SIEMENS

SIMATIC

STEP 7 und WinCC Engineering V15.1

Systemhandbuch

Ausdruck der Onlinehilfe

10/2018

Systemübersicht STEP 7 und WinCC 1

Was ist neu im TIA Portal 2

Liesmich 3

Installation 4

Projekte und Programme migrieren 5

Einführung in das TIA Portal 6

Projekte bearbeiten 7

Projektdaten bearbeiten 8

Benutzerverwaltung verwenden 9

Bibliotheken verwenden 10

Geräte und Netze bearbeiten 11

PLC programmieren 12

Prozesse visualisieren 13

Technologiefunktionen einsetzen 14

Online- und Diagnosefunktionen nutzen 15

Energiedaten verarbeiten 16

Team Engineering einsetzen 17

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 11/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

SIEMENS

Skalierung von STEP 7 und WinCC im TIA Portal 1

Optionen für STEP 7 und WinCC Engineering 2

Optionen für S7-1500 Runtime 3

Optionen für STEP 7 Engineeringsystem 4

Optionen für WinCC Engineering- und Runtimesysteme 5

Systemübersicht STEP 7 und WinCC

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Skalierung von STEP 7 und WinCC im TIA Portal.....5
2 Optionen für STEP 7 und WinCC Engineering.....7
3 Optionen für S7-1500 Runtime.....9
4 Optionen für STEP 7 Engineeringsystem.....11
5 Optionen für WinCC Engineering- und Runtimesysteme.....13

Skalierung von STEP 7 und WinCC im TIA Portal

1

Leistungsumfang der Produkte

Die folgende Grafik zeigt, welchen Leistungsumfang die einzelnen Produkte von STEP 7 und WinCC haben:

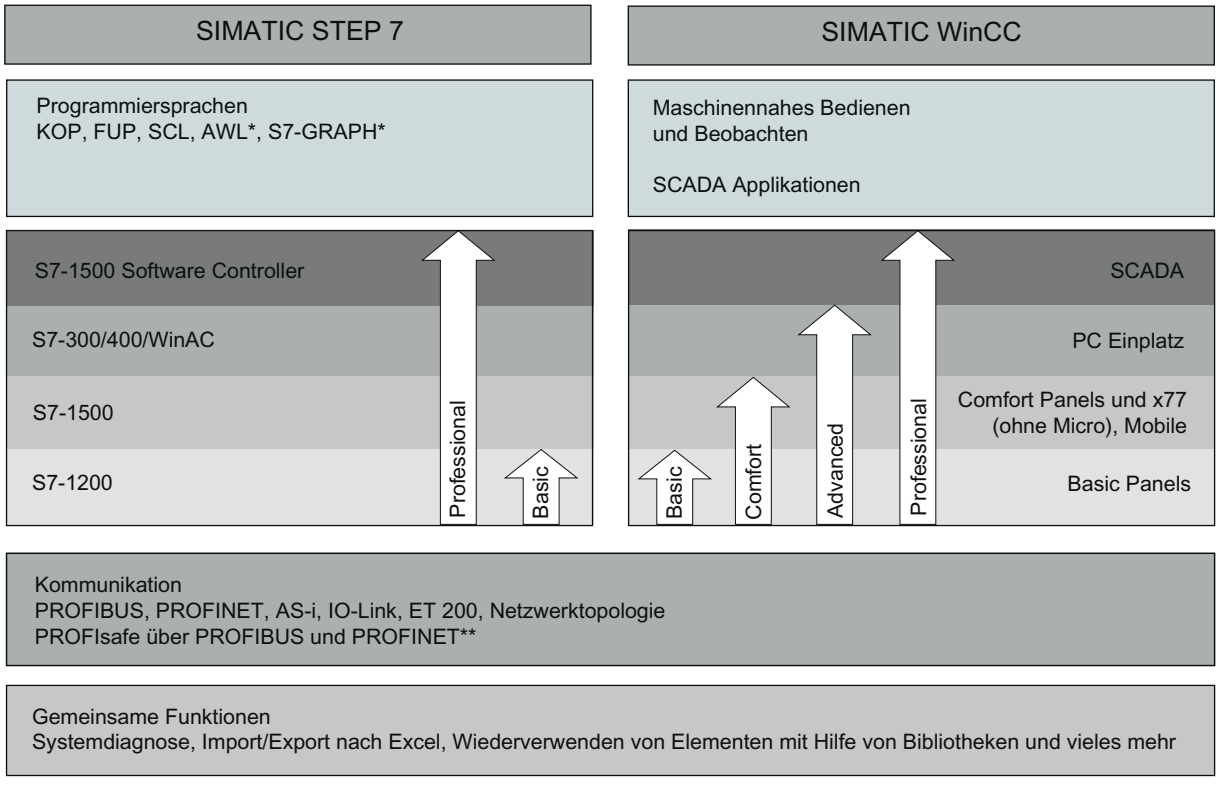

* Nur bei STEP 7 Professional für S7-300/400/WinAC und S7-1500
** Mit installiertem Optionspaket "STEP 7 Safety Advanced"

STEP 7

STEP 7 (TIA Portal) ist die Engineering-Software zur Projektierung der SIMATIC Steuerungsfamilien S7-1200, S7-1500, S7-300/400 und Software-Controller (WinAC). STEP 7 (TIA Portal) ist in zwei Editionen erhältlich, abhängig von den projektierbaren Steuerungsfamilien:

- STEP 7 Basic zur Projektierung der S7-1200
- STEP 7 Professional zur Projektierung von S7-1200, S7-1500, S7-300/400 und Software-Controller (WinAC)

WinCC

WinCC (TIA Portal) ist eine Engineering-Software zur Projektierung von SIMATIC Panels, SIMATIC Industrie PC und Standard PC mit der Visualisierungssoftware WinCC Runtime Advanced oder dem SCADA System WinCC Runtime Professional.

WinCC (TIA Portal) ist erhältlich in vier Editionen, abhängig von den projektierbaren Bediensystemen:

- WinCC Basic zur Projektierung der Basic Panels
  WinCC Basic ist immer auch Bestandteil der Editionen STEP 7 Basic und STEP 7 Professional.
- WinCC Comfort zur Projektierung aller Panels (inklusive Comfort Panels, Mobile Panels)
- WinCC Advanced zur Projektierung aller Panels sowie PCs mit der Visualisierungssoftware WinCC Runtime Advanced
  WinCC Runtime Advanced ist eine Visualisierungssoftware für PC-basierte Einzelplatzsysteme. WinCC Runtime Advanced ist erhältlich mit Lizenzen für 128, 512, 2k, 4k, 8k sowie 16k PowerTags (Variablen mit Prozessanbindung).
- WinCC Professional zur Projektierung von Panels sowie PCs mit WinCC Runtime Advanced oder dem SCADA-System WinCC Runtime Professional. WinCC Professional ist in den Editionen WinCC Professional für 512 und 4096 Powertags sowie als "WinCC Professional max. Powertags" erhältlich.
  WinCC Runtime Professional ist ein SCADA-System zum Aufbau von Konfiguration mit Einzelplatzsystemen bis zu Mehrplatzsystemen mit Standard- oder Web-Clients. WinCC Runtime Professional ist erhältlich mit Lizenzen für 128, 512, 2k, 4k, 8k, 64k, 100k, 150k sowie 256k PowerTags (Variablen mit Prozessanbindung).

Optionen für STEP 7 und WinCC Engineering 2

Für das STEP 7 und WinCC Engineering stehen folgende produktübergreifende Optionen zur Verfügung:

- TIA Portal Multiuser (Gemeinsames Arbeiten an einem TIA Portal Projekt)
- TIA Portal Cloud Connector (Zugriff auf lokale Schnittstellen bei RDP-Nutzung)
- TIA Portal Teamcenter Gateway (Anbindung an Teamcenter)
- SIMATIC Visualization Architect (Erstellung von HMI-Inhalten auf Basis von STEP7-Projekten)
- SIMATIC Energy Suite (Energiemanagement)
- SIMATIC Safe Kinematics (Sichere Zonen- und Geschwindigkeitsüberwachung von Kinematikbewegungen)

Optionen für S7-1500 Runtime 3

Das Engineering folgender Funktionen ist in den Engineering Paketen von SIMATIC STEP 7 Professional bzw. SIMATIC Energy Suite integriert. Die Freischaltung erfolgt im CPU Kontext:

- SIMATIC ProDiag (Maschinen- und Anlagendiagnose für S7-1500 und SIMATIC HMI)
- SIMATIC OPC UA S7-1500 (Anbindung beliebiger Fremdgeräte an die S7-1500)
- SIMATIC Energy Suite S7-1500 (Energiemanagement)

Optionen für STEP 7 Engineeringsystem

4

Für das STEP 7 Engineering stehen folgende Optionen zur Verfügung:

- SIMATIC STEP 7 Safety Basic/Advanced (Sicherheitsprogramme für F-CPUs)
- SIMATIC S7-PLCSIM Advanced (Virtueller Controller S7-1500)
- SIMATIC Target 1500S™ for Simulink® (Add-On für Simulink®)

Optionen für WinCC Engineering- und Runtimesysteme

5

Die SIMATIC Panels sowie WinCC Runtime Advanced und WinCC Runtime Professional beinhalten alle wesentlichen Funktionen für das Bedienen und Beobachten von Maschinen oder Anlagen. Für einen erweiterten Aufgabenbereich kann die Funktionalität teilweise durch zusätzliche Optionen ergänzt werden.

Option für Basic Panel

Für Basic Panels gibt es die folgende Erweiterungsmöglichkeit:

- WinCC Sm@rtServer (Remote-Bedienung)

Optionen für Comfort Panels, Mobile Panels

Für Comfort Panels und Mobile Panels gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC Audit (Audit Trail und Elektronische Unterschrift für regulierte Applikationen)
- SIMATIC ProDiag (Maschinen- und Anlagendiagnose für S7-1500 und SIMATIC HMI)

Optionen für WinCC Runtime Advanced

Für WinCC Runtime Advanced gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC SmartServer (Remote-Bedienung)
- WinCC Recipes (Rezeptursystem)
- WinCC Logging (Archivierung von Prozesswerten und Meldungen)
- WinCC Audit (Audit Trail für regulierte Applikationen)
- SIMATIC ProDiag (Maschinen- und Anlagendiagnose für S7-1500 und SIMATIC HMI)

Hinweis

Im Vergleich zu WinCC flexible 2008 sind Funktionen aus den Optionen WinCC flexible /Sm@rtService, WinCC flexible /Sm@rtAccess sowie die Option WinCC flexible /OPC-Server in die Basisfunktionalität eingeflossen.

Optionen für WinCC Runtime Professional

Für WinCC Runtime Professional gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC Client (Standard-Client zum Aufbau von Mehrplatzsystemen)
- WinCC Server (Ergänzung der WinCC Runtime um Server-Funktionalität)
- WinCC Recipes (Rezeptursystem, bisher WinCC/UserArchives)

- WinCC WebNavigator (Bedienen und Beobachten über Web)
- WinCC DataMonitor (Anzeige und Auswertung von Prozesszuständen und historischen Daten)
- WebUX (Plattform- und browserunabhängiges Bedienen und Beobachten über das Web)
- SIMATIC Information Server 2014 (Web-basierte und browserunabhängige Analyse und Berichte historischer Prozessdaten)
- SIMATIC Process Historian 2014 (Anlagenweiter Archivserver für Meldungen und Prozessdaten)
- Industrial Data Bridge (Konfigurierbare Verbindungen zu Datenbanken und IT-Systemen)
- Redundancy (Erhöhte Verfügbarkeit durch redundante Server)
- SIMATIC ProDiag (Maschinen- und Anlagendiagnose für S7-1500 und SIMATIC HMI)

Hinweis

Im Vergleich zu WinCC V7 sind Funktionen aus den Optionen WinCC /OPC-Server, WinCC / ConnectivityPack in die Basisfunktionalität eingeflossen. Ebenfalls in der Basisfunktionalität enthalten sind die Runtime API aus WinCC /ODK.

SIEMENS

Was ist neu im TIA Portal V15.1 1

Was ist neu im TIA Portal V15 2

Was ist neu im TIA Portal V14 SP1 3

Was ist neu im TIA Portal V14 4

Was ist neu im TIA Portal

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Was ist neu im TIA Portal V15.1 5

1.1 SIMATIC STEP 7 5

1.2 SIMATIC WinCC 9

1.3 SINAMICS Startdrive 10

1.4 Engineeringoptionen 12

1.5 Runtimeoptionen 18

2 Was ist neu im TIA Portal V15 21

2.1 Einleitung 21

2.2 SIMATIC STEP 7 22

2.3 SIMATIC WinCC 26

2.4 SINAMICS Startdrive 27

2.5 SIMOCODE ES 28

2.6 Engineeringoptionen 29

2.7 Runtimeoptionen 34

3 Was ist neu im TIA Portal V14 SP1 37

3.1 Einleitung 37

3.2 SIMATIC STEP 7 38

3.3 SIMATIC WinCC 42

3.4 SINAMICS Startdrive 44

3.5 Engineeringoptionen 45

3.6 Runtimeoptionen 48

4 Was ist neu im TIA Portal V14 49

4.1 Einleitung 49

4.2 SIMATIC STEP 7 50

4.3 SIMATIC WinCC 53

4.4 SINAMICS Startdrive 55

4.5 SIMOTION SCOUT TIA 56

4.6 SIMOCODE ES V14 57

4.7 Engineeringoptionen 58

4.8 Runtimeoptionen 61

Was ist neu im TIA Portal V15.1 1

1.1 SIMATIC STEP 7

Alle wichtigen Neuheiten von STEP 7 sind hier zusammengefasst. Weitere Details zu den einzelnen Themen können Sie den Kapiteln der Produktdokumentationen entnehmen.

Software Units

Software Units ermöglichen die Unterteilung des Anwenderprogramms in separat ladbare Einheiten. Damit können Sie Änderungen an verschiedenen Software Units unabhängig voneinander in die Steuerung laden, insbesondere auch dann, wenn die Änderungen von verschiedenen Nutzern vorgenommen wurden.

Die modulare Bearbeitung des PLC-Programms durch mehrere Benutzer wird dadurch deutlich vereinfacht.

Systemfunktionen

Für viele Funktionen des TIA Portals sind Tastenkombinationen im System definiert. Ab V15.1 können Sie alternativ zu den systemdefinierten Tastenkombinationen auch eigene Tastenkombinationen verwenden. Eine Übersicht aller Tastenkombinationen und die Möglichkeit, benutzerdefinierte Tastenkombinationen zuzuweisen, finden Sie in den Einstellungen des TIA Portals.

Editoren für Programmiersprachen

- Textuelle Bausteinschnittstelle in SCL
  Beim Anlegen neuer SCL-Bausteine haben Sie die Wahl zwischen tabellarischer und textueller Darstellung der Bausteinschnittstelle. Mit der textuellen Darstellung steht wieder die gewohnte Programmierumgebung aus STEP 7 V5.x zur Verfügung. Damit genießen Sie Vorteile, wie z. B. Kommentarabschnitte in der Bausteinschnittstelle anzulegen oder einfache Kopieraktionen mit anderen Texteditoren auszuführen.

- Aktualparameter in SCL
  Für eine bessere Lesbarkeit von Programmen in SCL-Bausteinen haben Sie die Möglichkeit, die Aktualparameter bei Bausteinaufrufen linksbündig auszurichten.

- Einstellwerte in PLC-Datentypen (UDTs)
  Die in einem anwenderdefinierten PLC-Datentyp (UDT) vordefinierten Einstellwerte können Sie in der verwendeten Instanz an- bzw. abwählen.

- Momentaufnahme von Aktualwerten
  Eine Momentaufnahme von Aktualwerten bleibt nun auch bei strukturellen Änderungen des Datenbausteins erhalten.

- Beobachten von Variablen
  Die Tastenkombination "Strg+T" zum Aktivieren/Deaktivieren des Beobachtens von Variablen ist nun in allen STEP 7-Editoren (DB-Editor, Beobachtungstabelle, Variablentabelle, etc.) verfügbar.

- Beobachten von PLC-Datentypen (UDTs)
  Beim Beobachten können Sie einen Ein- bzw. Ausgangsparameter vom Datentyp eines UDTs im Programm markieren und dazu die Aktualwerte im Inspektorfenster einblenden.

- Beobachten von Bausteinaufrufen
  Beim Beobachten von Bausteinaufrufen werden nun auch Aktualwerte für nicht verschaltete Ausgänge angezeigt.

Hardware-Konfiguration

- S7-1500 CPUs für die Konfiguration von redundanten Automatisierungssystemen
  Für die Konfiguration von redundanten und hochverfügbaren Automatisierungssystemen stehen Ihnen folgende S7-1500 CPUs zur Verfügung:
  - CPU 1513R-1 PN
  - CPU 1515R-2 PN
  - CPU 1517H-3 PN

- Taktsynchronität am zentralen Rückwandbus
  Zentral gesteckte Module in einer S7-1500-Station lassen sich taktsynchron betreiben. Eine taktsynchrone Kopplung zentral gesteckter Module mit dezentral gesteckten Modulen ist ebenfalls möglich.

- MRP-Domain-Konfiguration über Projektgrenzen hinweg
  Redundanzmanager und Redundanz-Clients einer MRP-Domain dürfen sich in unterschiedlichen Projekten befinden.

- Schnelles Ändern der Firmware-Version für mehrere Module
  Die konfigurierte Firmware-Version von Modulen in einem IO-Device können Sie durch Mehrfachauswahl einer Gruppe von Modulen in einem Schritt ändern.

- Schnelles Umschalten zwischen E/A-Variable im Programmiereditor und betreffendem Ein-/Ausgang in der Gerätesicht
  Das Kontextmenü "Gehe zu > Gerätesicht" ermöglicht eine schnelle Navigation von der programmierten E/A-Adresse aus den Programmeditoren heraus zum konfigurierten Ein- oder Ausgang in der Gerätesicht.

- Vereinfachte Konfiguration von Serienmaschinen-Projekten
  Für die Projektierung von optionalen IO-Devices in einem Serienmaschinen-Projekt entfällt die Notwendigkeit einer Portverschaltung (Topologie-Projektierung).

S7-1500 Motion Control

Die Technologieversion V4.0 enthält folgende Neuerungen:

- Austausch von Momentendaten mit dem Antrieb in den technologischen Einheiten des Technologieobjekts:
  - Additives Sollmoment
  - Aktuelles Istmoment
  - Zulässiger Momentenbereich
- Zeitliche Anforderungen für das Technologieobjekt „Messaufträge“ über "MC_MeasuringInput"
- Erweiterung der Datenstruktur von Positionierachse und Gleichlaufachse zur Verwendung des Technologieobjekts Kinematik
- Verwendung optimierter Datenbausteine (Antriebs-/Geberanbindung)

Die Technologieversion V4.0 enthält für Technologie-CPUs (S7-1500T) folgende zusätzliche Neuerungen:

- Technologieobjekt Kinematik (S7-1500T)
- Bewegungsvorgabe über "MotionIn"-Anweisungen (S7-1500T)
- Direkt synchron setzen mit "MC_CamIn" V4.0 (S7-1500T)

Die Technologieversion V4.0 enthält für diese Funktionen neue bzw. erweiterte Motion Control-Anweisungen.

Trace

Vereinfachte Handhabung von Diagrammkonfigurationen:

- Für die Anzeige relevante Einstellungen können bereits während der Konfiguration erfolgen. Das betrifft die Signalgruppierung, Farbauswahl, Anzeigeformat und das Anlegen von Formeln.

- Online gemachte Änderungen können durch „Übertragen der Tracekonfiguration aus dem Gerät“ übernommen werden. Dadurch bleiben die Einstellungen im Register "Diagramm" erhalten, auch wenn Sie bei der Konfiguration Änderungen vornehmen und beispielsweise Signale hinzufügen oder die Anzahl der Messpunkte verändern.

1.2 SIMATIC WinCC

WinCC RT Professional

- OPC UA-Server
  - Unterstützung OPC UA-Server-Alarm und -Condition
  - WinCC Alarms können über OPC UA-Server an eine 3rd party-Applikation übertragen werden.

- ProDiag-Control
  - Anzeige des gesamten Call Interface Bausteins

- Weitere Funktionalitätserweiterungen
  - Einstellung der WebUX / WebNavigator Benutzerrechte in Runtime
  - Automatisches Login für Operator-Rolle
  - HMI Compiler bringt zusätzliche Informationen für Benutzer (z. B. Anzahl der kompilierten HMI-Objekte)
  - Software Control mit WinCC Professional auf einen gemeinsamen Rechner

Mobile Panel

- KTP Mobile V15-Images und SmartServer-Option für F-Geräte
  https://support.industry.siemens.com/cs/ww/de/view/109758056 (https://support.industry.siemens.com/cs/ww/de/view/109758056)

- Option+ V2 für Comfort und Mobile Panel
  - Unterstützung der Mobile Panel 2nd Generation
  - Code wird direkt in eine Variable geschrieben mit QR Reader MV320/325
  - Anzeige der Panel CPU Last direkt in der Runtime
  - Optimierung des Zertifikats-Handling von SIMATIC Logon
  - Ausblenden einzelner Desktop Icons
  - Erweiterung der Service-Datei für technischen Support
  - Umstellung der Kommunikation zwischen Option+ und Panel Runtime von SOAP auf OPC UA

- Mobile Panel 2nd Gen., Comfort Panel undBasic Panel 2nd Gen.
  Konfiguration des Meldepuffers

1.3 SINAMICS Startdrive

Alle Antriebsfamilien (G120, S120, S210)

- Erweiterungen der Openness Schnittstelle:
  - Funktionale Schnittstellen für wichtige Anwendungsfälle (z. B. Motordaten eingeben)
  - Für G120 und S210: Generierung des TIA Portal-Projektes auf Basis eines AML-basierten Imports von EPLAN
- Startdrive Advanced V15.1: Safety Integrated Abnahmetest auch für S210 und S120
  - Geführter Abnahmetest-Assistent für alle drive-based Safety Integrated Functions (Basic, Extended und Advanced)
  - Automatische und Safety Integrated-funktionsspezifische Erstellung von Traces zur Analyse des Maschinenverhaltens
  - Generierung eines Abnahmeprotokolls als Excel-Datei (xlsx-Format, auch mit OpenOffice nutzbar)

SINAMICS G120

- Anwenderdefinierbare Parameterlisten
- Datenabgleich zwischen G120 und S7-1500 Technologieobjekten auch offline (im Projekt) möglich
- Usability Verbesserungen:
  - Übersichtsmaske für G120-Konfiguration
  - Vereinfachte Geräteidentifikation über LED-Blinken im Dialog “Online Verbinden”

Integration von SINAMICS S210 und 1FK2

- Integration der S210-Antriebe mit 200V und 400V ab SINAMICS Firmware V5.2
- Unterstützung von 1FK2-Motoren
- Usability: Gegenüber S120 vereinfachter Projektierungsablauf
- Datenabgleich zwischen S210 und S7-1500-Technologieobjekten online und offline (im Projekt)
- Trace (inkl. Vorschlag typischer Parameter)
- Firmware-Update über TIA Portal
- Automatische Regleroptimierung über One-Button-Tuning
- Safety Integrated-Projektierung

SINAMICS S120 (CU320-basierte Antriebe)

- Unterstützung der SINAMICS-Firmware V5.1 SP1
- Unterstützung von 1FK2-Motoren

- Betrieb an S7-1500 Open-/Software-Controllern
- Datenabgleich zwischen S120 und S7-1500-Technologieobjekten auch offline (im Projekt)
- Anbindung an S7-1500 Technologie-Objekt Typ „Messtaster“
- Unterstützung des Technologiedatenblock-Telegramms 750
- Unterstützung von Isochronous Safe Position (zur Unterstützung der PLC-Funktion "kartesisches Safety")
- Parameterliste mit funktionaler Struktur (Parametergruppen)
- Automatische Regleroptimierung über One-Button-Tuning

1.4 Engineeringoptionen

TIA Portal Multiuser Engineering

- Verbesserung der Performance bei geringen Netzwerk-Bandbreiten
  Es wurden neue Netzwerkprofile für eine verbesserte Konnektivität zwischen der Engineering-Station und dem Multiuser-Server hinzugefügt.

- Funktionserweiterungen bei den Multiuser-Powertools (Kommandozeilen-Tools)
  Es gibt jetzt Export- und Importfunktionen für Multiuser-Serverprojekte und lokale Sessions.

TIA Portal Multiuser Commissioning

Multiuser Engineering unterstützt den Anwender bei der Inbetriebnahme durch einen geführten Commissioning-Workflow. Im Commissioning-Modus wird das Laden ins Gerät automatisch synchronisiert. Dies ermöglicht einen konsistenten Stand zwischen dem Gerät und dem Multiuser-Serverprojekt den zugehörigen lokalen Sessions.

PLCSIM Advanced

- Control Panel-Erweiterungen
  - Durch einen Doppelklick kann das Control Panel als freischwebendes Fenster geöffnet werden. Dieses lässt sich beliebig verschieben und bei Bedarf immer im Vordergrund anzeigen.
  - In das freischwebende Control Panel können bereits in der Vergangenheit angelegte Instanzen durch Auswahl der virtuellen Memory Card im Explorer via Drag & Drop vom Explorer auf das Control Panel angelegt werden.

- API-Erweiterungen
  Die maximale Zykluszeit des virtuellen Controllers kann über die API frei definiert werden, ohne eine Änderung der maximalen Zykluszeit im TIA Portal-Projekt vornehmen zu müssen.

SIMATIC TIA Portal Openness

- Openness-DLLs von V14 SP1, V15 und V15.1 im Lieferumfang
  Da die Openness-DLLs von V14 SP1, V15 und V15.1 im Lieferumfang enthalten sind, laufen Applikationen, die auf V14S P1 oder V15 basieren, ohne Änderung auch in V15.1. Um die Funktionen der V15.1 nutzen zu können, müssen Sie die DLL der V15.1 einbinden und die Applikation neu übersetzen.

- Zugriff auf Projekte
  - Über Openness können mehrere Projekte in einer Instanz des TIA Portals geöffnet werden.
  - Projekte können über Openness archiviert und wiederhergestellt werden.
  - Bei UMAC-geschützten Projekten ermöglicht Openness lesende Zugriffe auf Objekte. Für die Zugriffe gelten die gleichen Einschränkungen wie für einen Anwender mit der Berechtigung "Nur Lesen".

- Speichern unter
  Projekte und globale Bibliotheken können über Openness mit einem anderen Namen gespeichert werden.

- Online-Offline Vergleich
  Über Openness ist der Vergleich von Online- und Offline-Daten möglich.

- Zugriff auf Schutzstufen
  Bei Bausteinen kann eine Schutzstufe gesetzt oder entfernt werden.

- Zugriff auf Fingerprints
  Bei Bausteinen und PLC-Datentypen (UDTs) kann der Fingerprint abgefragt werden.

- Zugriff auf Namen
  Bei Bausteinen, Datenbausteinen und UDTs kann der Name gesetzt werden.

- Fehlertoleranter Import
  Neben dem strikten Import, der weiterhin genutzt werden kann, steht jetzt auch ein fehlertoleranter Import zur Verfügung. Der Import ist auch dann möglich, wenn z. B. verknüpfte Anwenderdatentypen oder aufgerufene Bausteine nicht zueinander passen.

- Export und Import
  - Sowohl Beobachtungstabellen als auch Force-Tabellen können exportiert und importiert werden.
  - Von einem Offline-DB können Momentaufnahmen der Aktualwerte als XML exportiert werden. Somit können verschiedene Momentaufnahmen über die XML-Dateien verglichen werden.

- ET200SP
  Für die meiste Attribute der Module einer ET200SP ist lesender und schreibender Zugriff möglich.

- R/H-Systeme
  Bei R/H-Systemen ist das Laden auf die primäre PLC und die Backup-PLC möglich.

- Fehlersicher PLC
  - Der Upload ist jetzt auch indirekt über NAT-Router möglich.
  - Beim Upload werden auch Rezepturen, Archive, Benutzerdateien und die Kennwörter der Schutzstufen berücksichtigt.

SIMATIC STEP 7 Safety

- Neue fehlersichere CPU-CPU Kommunikation „Flexible F-Link“
  Mit STEP 7 Safety V15.1 steht für die fehlersicheren S7-1200 und S7-1500 CPUs eine neue fehlersichere CPU-CPU-Kommunikation „Flexible F-Link“ zur Verfügung. Fehlersichere Daten können hiermit einfach, als fehlersicherer ARRAY, mittels Standard-Kommunikationsmechanismen zwischen zwei fehlersicheren CPUs ausgetauscht werden.

- Fehlersichere Übertragung großer Mengen fehlersicherer Daten über Standard-Kommunikationsbausteine
  Flexible F-Link unterstützt die fehlersichere Übertragung großer Mengen fehlersicherer Daten mittels Standard-Kommunikationsbausteinen auch über Netzwerkgrenzen (Layer 3-Kommunikation) hinweg. Voraussetzung ist die konsistente und deterministische Übertragung der Daten.
  Die zu übertragenden Daten werden in benutzerdefinierten F-konformen PLC-Datentypen (UDTs) zusammengefasst. Dabei können beliebige fehlersichere Datentypen (Bool, Word, INT, DINT, TIME) verwendet werden. Es werden bis zu 100 Byte pro UDT unterstützt.

- Einfache Parametrierung und automatische Generierung von fehlersicheren Kommunikations-DBs
  Die Parametrierung der Kommunikationsverbindung erfolgt über den Safety Administration Editor (SAE), in dem jede Kommunikationsverbindung über Name, UDT, Überwachungszeit, Sende- oder Empfangsrichtung angelegt wird und vom System automatisch ein Kommunikations-DB erzeugt wird. Dieser Kommunikations-DB stellt die zu übertragenden Daten in Form eines kodierten F-ARRAY zur Verfügung, das dann einfach über Standard-Kommunikationsbausteine wie z. B. TSEND oder USEND übertragen werden kann.

- Fehlersichere Ablaufgruppenkommunikation
  Flexible F-Link bietet ebenso die Möglichkeit, auf einfachem Wege fehlersichere Ablaufgruppenkommunikation zu realisieren. Hierzu können die fehlersicheren Daten beispielsweise über die Anweisung „UMOVE_BLK“ zwischen den F-Ablaufgruppen übertragen werden.

- Netzweit eindeutige Kommunikations-ID (UUID)
  Auch auf die Frage, wie die eindeutige Identifikation der Kommunikationspartner auch über Netzwerkgrenzen sichergestellt wird, liefert Flexible F-Link eine Antwort. Hierzu kann mit STEP 7 Safety eine eindeutige Kommunikations-ID (Universally Unique Identifier UUID) erzeugen werden, und so die Eindeutigkeit gewährleistet werden.

- Kommunikationssignatur
  Eine separate Kommunikationssignatur, unabhängig von der Hardware- und Software-Signatur, ermöglicht es, Änderungen an der UUID klar von Programm- und Hardwareänderungen zu unterscheiden.

- Pre- und Postprocessing
  Zur Vereinfachung der Projektierung und für ein einfach nachvollziehbares Zeitverhalten speziell im Kontext der fehlersicheren Kommunikation, bietet STEP 7 Safety in der Version V15.1 die Möglichkeit, Standardbausteine zur Vor- und Nachverarbeitung von Prozesswerten um die F-Ablaufgruppen zu platzieren.

- Variable Kommunikations ID (DP_DP_ID)
  Ab Version V15.1 bietet STEP 7 Safety die Möglichkeit, die für die Bausteine SENDDP und RCVDP benötigten Kommunikations-IDs (DP_DP_IDs), anstelle von Konstanten, mit Variablen zu belegen. Hierdurch wird es dem Anwender beispielsweise ermöglicht, in Anlagen mit einer Vielzahl von identischen iDevices (Hochregallager, AGVs, Einschienenhängebahnen, etc.), Sicherheitsprogramme mit gleicher F-Gesamtsignatur einzusetzen. Die variable Kommunikations-ID ermöglicht so eine vereinfachte Inbetriebsetzung, Wartung und Projektverwaltung.
  Achtung: Werden variable Kommunikations-IDs (DP_DP_ID) verwendet, muss der Anwender applikativ dafür Sorge tragen, dass die Kommunikations-IDs netzweit und zu jedem Zeitpunkt eindeutig sind.

- Openness
  Über die Openness-Schnittstelle werden mit STEP 7 Safety V15.1 weitere Funktionen bereitgestellt
  - Lesen und Konfigurieren von i-Parametern der ET200SP F-IOs
  - Station Upload
  - Hardware- und Software-Vergleich (offline/offline)
  - Import und Export von konsistenten (unveränderten) F-Bausteinen

TIA Portal Teamcenter Gateway

- Multiuser Engineering
  Mit Hilfe der Optionen "TIA Portal Teamcenter Gateway" und "Multiuser Engineering" können jetzt mehrere Personen gleichzeitig an einem TIA Portal-Projekt arbeiten.

- Referenzprojekte aus Teamcenter öffnen
  Sie können Projekte aus Teamcenter direkt als Referenzprojekte öffnen, um z. B. das Vergleichen und Zusammenführen von verschiedenen Revisionen zu ermöglichen.

- Speichern von Projekten unter existierende Teamcenter-Elemente
  Sie können TIA Portal-Projekte und -Bibliotheken aus dem TIA Portal heraus unter bereits bestehende Teamcenter-Elemente speichern.

- Kompatibilität
  Es werden die Teamcenter-Versionen V11.2/11.3/11.4/11.5 unterstützt.

SIMATIC Energy Suite

- Energy Screens
  Jetzt Bestandteil des Lieferumfangs von Energy Suite V15.1

- Reports
  Neu: Kostenstellenbericht inkl. Tarife

- SINAMICS
  Die Konfiguration von SINAMICS-Geräten mittels MDD wird unterstützt (bisher wurde ausschließlich GSDML unterstützt)

- Usability-Verbesserungen
  Verschiedene kleinere Usability-Verbesserungen

SIMATIC Visualization Architect

- Zugriffsschutz auf die SiVArc-Regeleditoren durch UMAC

- X/Y – Eigenschaften können durch SiVArc-Expressions gesetzt werden.

- Support von STEP 7-Bausteinen, die in SCL geschrieben sind

- Generieren auf Template Screens

- SiVArc – Openness:
  - Kopieren von einzelnen Regeln oder ganze Regel-Gruppen aus der Bibliothek
  - Starten der SiVArc-Generierung

1.5 Runtimeoptionen

SIMATIC ProDiag

- Textexport/-import für Übersetzungen
  Alle ProDiag- bzw. S7-GRAPH-relevanten Texte können mit einer Aktion z. B. für Übersetzungen exportiert werden.
  Folgende Meldetexte können exportiert werden:
  - Anwenderdefinierte Textlisten für Überwachungen
  - Anzeigenamen für Schritte und Transitionen in GRAPH
  - Kommentare aller überwachten Variablen
  - Kommentare aller überwachten Parameter eines Bausteins
  - Instanzname und Instanzkommentar von Multiinstanzen

  Folgende Texte für de HMI-Anzeige können exportiert werden:
  - Variablenkommentare in Transition und Interlock
  - Titel und Kommentare überwachter Netzwerke oder Variablen

- Anzeige des ganzen Bausteins (KOP, FUP) im HMI PLC-Code Viewer
  Die Nutzung des PLC-Code Viewers ist auch möglich, wenn Static-Parameter eines Bausteins überwacht werden (Anwenderbausteine mit eigener Überwachungslogik)

- Bausteinüberwachungen im ProDiag-Übersichtseditor editieren
  Vorhandene Überwachungen im Register "Überwachungsdefinitionen am FB" können editiert oder gelöscht werden.

- Einsprung in das TIA Portal aus WinCC Advanced
  - Systemunterstützter Einsprung in das TIA Portal im Kontext einer GRAPH- bzw. ProDiag-Meldung
  - Relevanter Baustein wird automatisch geöffnet
  - Lesemodus oder Schreib-/Lesemodus möglich
  - Automatische Statusanzeige

- Berücksichtigung hierarchischer Kommentare in ProDiag-Meldungen
  - Leichte Abbildung der technologischen Sicht auf die Maschine / Anlage über hierarchische Kommentare, wobei der Pfad automatisch generiert wird.

- Unterkategorien in GRAPH
  - Identischer Meldungsaufbau wie bei ProDiag konfigurierbar.
  - Insbesondere für Leitsysteme relevant, bei denen eine Folgeaktion über Zusatzinformationen im Meldetext abgeleitet werden.

OPC UA

Für S7-1500 CPUs ab Firmware V2.6 können Sie mit einer entsprechenden OPC UA Runtime-Lizenz von folgender Erweiterung profitieren:

Zusätzlich zum OPC UA-Server ist ein OPC UA-Client in der CPU integriert und bietet über entsprechende OPC UA-Kommunikationsanweisungen folgende Funktionen:

- Methodenaufrufe
- Lesen und Schreiben von Daten

Der Client erschließt Ihnen z. B. folgende Anwendungsbereiche:

- Vertikale Kommunikation zu MES-Systemen oder zu Cloud-Diensten
- Controller-Controller-Kommunikation

Was ist neu im TIA Portal V15

2

2.1 Einleitung

Beschreibung

Alle wichtigen Neuheiten gegenüber den bestehenden Versionen sind hier zusammengefasst aufgeführt. Weitere Details zu den einzelnen Themen können Sie den einzelnen Kapiteln der Produktdokumentationen entnehmen.

2.2 SIMATIC STEP 7

Beschreibung

Alle wichtigen Neuheiten von STEP 7 sind anhand des Engineering-Workflows zusammengefasst aufgeführt.

Hardware-Konfiguration

- Die folgenden neuen CPUs der Familie S7-1500 runden das Portfolio ab:
  - CPU 1518(F)-4 PN/DP MFP (Multifunktionale Plattform)
  - CPU 1516T(F)
- Hardware-Erkennung von real vorhandenen PROFINET IO-Geräten
  Die Online-Funktion "Hardware-Erkennung" erkennt IO-Devices am angeschlossenen PROFINET-Subnetz. Ein erkanntes Gerät können Sie in Ihr Projekt übernehmen: STEP 7 fügt das IO-Device mit allen Modulen und Submodulen ein und erspart so das manuelle Einfügen von IO-Geräten und -Modulen aus dem Hardware-Katalog.
- Die Funktion "Gehe zur Gerätesicht" ermöglicht die schnelle Navigation aus der PLC-Variablentabelle zum konfigurierten Ein- oder Ausgang in der Gerätesicht.
- Erweiterte Meldungsanzeige mit neuen Filterfunktionen.

Editoren für Programmiersprachen

- Mehrsprachige Projekttexte, z. B. Baustein- und Netzwerktitel oder Kommentare, können Sie in allen verfügbaren Sprachen direkt in den Programmierditoren anzeigen und bearbeiten. Dort können Sie die Texte auch zur externen Übersetzung exportieren und importieren. Folgende Editoren unterstützen die lokale Anzeige von Projekttexten: PLC-Variablentabelle, Programmiereditoren (KOP, FUP, SCL, AWL, GRAPH), Datenbausteine und PLC-Datentypen.
- In SCL wurde die Statuszeile erweitert: Die aktuelle Cursorposition wird nun angezeigt. Durch Doppelklick auf die Zeilennummer können Sie die Funktion "Gehe zu Zeilennummer" aufrufen, um zu weiteren Zeilen zu navigieren. Außerdem wird der Editiermodus (Einfügen/Überschreiben) in der Statuszeile angezeigt. Durch Doppelklick auf das Anzeigefeld können SIe den Editiermodus umschalten.

Neue Anweisungen

- Mit den Anweisungen "FileReadC" und "FileWriteC" können Sie bei S7-1500-CPUs Daten aus einer ASCII-Datei von der SIMATIC Memory Card lesen bzw. Daten in eine ASCII-Datei auf die SIMATIC Memory Card schreiben.
- Mit der Anweisung "GetClockStatus" können Sie bei S7-1500-CPUs folgende Informationen über die CPU-interne Uhr auslesen:
  - Informationen zur Uhrzeitsynchronisation über NTP-Server
  - Information, ob die automatische Anpassung an Sommer- und Winterzeit aktiviert ist

- Mithilfe der Anweisung "EQ_TypeOfDB" (SCL: "TypeofDB") kann der Datentyp eines Datenbausteins ermittelt werden, der über eine DB_ANY Variable adressiert wird.
- Die bestehenden Anweisungen "SCATTER" und "GATHER" wurden erweitert und unterstützen jetzt auch einen anonymen STRUCT-Datentyp und PLC-Datentypen mit ausschließlich booleschen Elementen.
- Es stehen zwei neue PID-Hilfsfunktionen zur Verfügung. Diese reduzieren den Programmieraufwand bei Aufgaben in der Regelungs- und Steuerungstechnik.
  - "SplitRange"
    Dient zur Aufteilung der Stellgröße auf mehrere Stellglieder. Damit können in Abhängigkeit vom Reglerausgang unterschiedliche Stellglieder angesteuert werden
  - "RampFunction"
    Damit lassen sich die Änderungsgeschwindigkeit und die Maximalwerte von Signalen begrenzen. Die Anstiegsgeschwindigkeit kann individuell in vier Betriebsbereichen begrenzt werden: Für positive oder negative Signale, für steigende oder fallende Signale.

Sprachinnovationen

- Mit "Referenzen" steht eine neue Art von Zeiger zur Verfügung. Referenzen sind typisierte Zeiger, die auf einen konkreten Datentyp verweisen. Wenn Sie Referenzen verwenden, definieren Sie den Datentyp bereits bei der Programmerstellung. Da der Datentyp nicht zur Laufzeit ermittelt werden muss, wird das Programm performanter und übersichtlicher. Durch "Dereferenzieren" können Sie direkt schreibend oder lesend auf die referenzierte Variable zugreifen, ohne die Variable zuvor zu kopieren oder zusätzliche Anweisungen in Ihr Programm einzubinden.
  Voraussetzung für den Einsatz von Referenzen ist eine CPU der Baureihe S7-1500 ab FW V2.5.
- Die Verwendung von Konstanten vom Typ BOOL an Eingängen von Anweisungen ist in KOP/FUP möglich. Das Testen bzw. "Überbrücken" von Strompfaden wird dadurch erleichtert.
  Voraussetzung für diese Funktion ist eine CPU der Baureihe S7-1200 ab FW V4.0 oder S7-1500 ab FW V1.8.

Systemfunktionen

- Das Informationssystem bietet die Möglichkeit, Hardware-Handbücher integriert anzuzeigen. Das hat den Vorteil, dass Sie die Inhalte durchsuchen, filtern oder als Favoriten verwenden können. Einige Hardware-Handbücher sind bereits in der Installation des TIA Portals V15 enthalten. Bei Bedarf können Sie weitere Hardware-Handbücher, die als Support Package verfügbar sind, herunterladen. Wählen Sie hierzu die Funktion "Nach Updates suchen" in der Funktionsleiste des Inhaltsverzeichnisses.
- Lokale Benutzer- und Rechteverwaltung:
  - Verwaltung von Projektbenutzern
  - Neue Funktionsrechte Projekt lesend öffnen / Projekt lesend+schreibend öffnen / Benutzer und Rollen anlegen
  - Verwaltung von Projektrollen
  - Zuweisung von Benutzern zu Projektrollen
  - Ein Baustein für Industrial Security mit TIA Portal
  - Effiziente Benutzeradministration für Produkte in einem TIA Portal Projekt

Bibliotheken

- Globale Bibliotheken können als schreibgeschützte Bibliotheken exportiert werden. Die enthaltenen Bibliothekstypen erhalten einen permanenten Schreibschutz und sind somit vor Änderungen geschützt. Der Schreibschutz der Bibliothekstypen bleibt auch bei der Verwendung als Typ-Instanz erhalten. Der Schreibschutz einmal geschützter globaler Bibliotheken kann nicht aufgehoben werden. Schreibgeschützte Bibliotheken sind verfügbar für PLC- und HMI-Typen.

Neue Funktionalität beim Steuern von Operanden

- Boolesche Operanden können durch einen Doppelklick auf den Beobachtungswert gesteuert werden. Die Aktualwerte können damit einfach und schnell getoggelt werden.
- Nicht boolesche Operanden können ebenfalls einfach und schnell über den hinterlegten Dialog gesteuert werden.
- Unterstützt werden sowohl globale Variablen als auch Instanzvariablen im DB.

Querverweise für verwendete Anweisungen

- Mit Hilfe von anwenderdefinierten Filtern vom Typ "Instruction" können bei den Querverweisen alle in einer CPU verwendeten Anweisungen mit den jeweiligen Anweisungsversionen angezeigt werden.

Laden von PLC-Variablentabellen

- PLC-Variablentabellen können in der vorgegebenen Struktur ins Gerät und vom Gerät geladen werden und ermöglichen ein verbessertes Team Engineering auf der CPU.
- Die auf der CPU vorhandenen PLC-Variablentabellen werden auch bei den "Erreichbaren Teilnehmern" und auf der Memory Card angezeigt.
- Durch den Online/Offline-Vergleich erhalten Sie einen detaillierten Überblick über die jeweils Online und Offline vorhandenen Variablen in den einzelnen PLC-Variablentabellen.

Haltepunkte für S7-1500 ab Firmwarestand V2.5

- Das Testen mit Haltepunkten ist für SCL- und AWL-Programme auch in gemischten KOP/FUP-Bausteinen möglich.
- Beim Erreichen eines Haltepunktes geht die CPU in den Betriebszustand "HALT". Variablen können auch bei erreichtem Haltepunkt beobachtet und gesteuert werden.
- Damit kann der AWL- und SCL-Programmcode mit Hilfe von Haltepunkten schrittweise getestet werden, um logische Fehler schon bei der Programmerstellung zu finden und zu beheben. Dies ermöglicht ein einfaches und schnelles Analysieren komplexer Programme schon vor der eigentlichen Inbetriebnahme.

PLCSIM

- Know-how geschützte Bausteine einer S7-1500 CPU können mit PLCSIM V15 simuliert werden (S7-1200 CPUs werden aktuell nicht unterstützt).
- Mit Schieberegler für analoge Werte und Push-Buttons für boolesche Werte können Sie sehr einfach Änderungen von Werten innerhalb der SIM-Tabelle für einen schnellen Test des STEP 7 Anwenderprogramms durchführen.
- PLCSIM V15 und PLCSIM Advanced V2.0 können auf demselben PC installiert werden. Eine gleichzeitige Nutzung beider Simulations-Tools ist jedoch ausgeschlossen

Trace

- Erzeugung virtueller Signale anhand von mathematischen Funktionen aus den aufgezeichneten Signalen. Unterstützt werden folgende Funktionen:
  - Grundrechenarten
  - Betragsbildung, Wurzel, Quadrat, 1/x, Modulo-Operation
  - Integral, Differentiation
  - Filterfunktion
- Berechnung des Mittelwerts, Effektivwerts und Integrals des selektierten Signals im Bereich der Messcursor
- Zeitvariablen vom Typ TIME, LTIME, TOD, LTOD, DATE und LDT können als Signal aufgezeichnet und zur Vorgabe von Triggerbedingungen verwendet werden
- Import und Export von überlagerten Messungen
- Verschieben der Messcursor mithilfe der Pfeiltasten

2.3 SIMATIC WinCC

Bediengeräte

Folgende Bediengeräte sind im TIA Portal zusätzlich projektierbar:

- TP1200 Comfort PRO
- TP1500 Comfort PRO
- TP1900 Comfort PRO
- TP2200 Comfort PRO

Bediengeräteversionen

Um die Größe der Standard-Installation zu optimieren, sind ältere Versionen der Bediengeräte-Images auf eine zusätzliche DVD ausgelagert. Betroffen sind die Geräteversionen:

- V13
- V13 SP1
- V13 SP2
- V14

SVG-Grafiken

In WinCC wird für Grafiken zusätzlich das Format SVG unterstützt.

Erweiterung der Funktionalität von ProDiag

Zur verbesserten Prozessdiagnose ist die Kriterienanalyse-Anzeige projektierbar. Damit können fehlerhafte Operanden angezeigt werden, die im Anwenderprogramm eine ausgewählte ProDiag- oder GRAPH-Meldung ausgelöst haben.

OPC UA Client in WinCC RT Professional

Die Kommunikation über OPC UA wurde erweitert:

- Es werden Arrays und Arrayvariablen unterstützt.
- Die Kommunikation über eine OPC UA-Verbindung kann durch ein Kennwort geschützt werden.

Anzahl der Verbindungen bei WinCC RT Professional

Die Anzahl der möglichen integrierten Verbindungen zu S7-1200 und S7-1500 wurde auf 128 erhöht.

2.4 SINAMICS Startdrive

SINAMICS Startdrive Basic (ehemals SINAMICS Startdrive)

- Erweiterungen für CU320-2 PN basierte Antriebsgeräte:
  - Unterstützung von G130, G150, S150 und Sinamics MV (bis 85.00 kW)
  - Unterstützung von Chassis und Cabinet Modules (nicht Blocksize Modules)
  - Unterstützung von SIMOTICS Asynchronmotoren und Fremdmotoren
  - Vektor-Regelung
  - Parametervergleich (Online/Offline, gegen Werkseinstellung)
- Erweiterungen für SINAMICS G120 Familie:
  - Unterstützung der SINAMICS Firmware Version V4.7 SP9
  - Weitere Optimierung und Ausbau des Inbetriebnahmeassistenten:
    - Konfiguration der Motorhaltebremse
    - Online abbrechen
    - CU250D-2: SSI-Geber als Motorgeber
  - PROFINET-Namensvergabe ohne Neustart der G120 Control Unit auch in der Liste der erreichbaren Teilnehmer (Accessible devices)
  - Unterstützung der CU240D-2/CU250D-2 mit Polymeric Optical Fiber (POF)
- Openness für Antriebsgeräte:
  - Antriebsgeräte und -komponenten anlegen (inklusive Telegrammkonfiguration)
  - Einstellung von ausgewählten Antriebsparametern (offline und online, lesen und schreiben)
  - Download ins Gerät (kein Upload)

SINAMICS Startdrive Advanced

- Einführung der Startdrive Advanced Lizenz zur Nutzung zusätzlicher Engineering-Funktionen mit hohem Mehrwert
- Nur Licence Key erforderlich, keine zusätzliche Installation
- Trial-Lizenz kostenfrei ohne Licence Key (21 Tage)
- Funktionen in V15: Safety Abnahmetest für G120-Familie
  - Geführter Abnahmetest-Assistent für alle Safety Integrated Funktionen (Basic und Extended Safety)
  - Automatische und Safety-funktionsspezifische Erstellung von Traces
  - Generierung eines Abnahmeprotokolls als Excel-Datei

2.5 SIMOCODE ES

Beschreibung

Folgende Neuheiten stehen bei SIMOCODE ES im TIA Portal zur Verfügung:

- Anwenderkonfigurierbare Darstellung von Geräteparametern
  - Im Parameter Editor ist eine neue Sicht auf die Geräteparameter verfügbar, die sogenannte Expertenliste.
  - Darin können Gruppen von Parametern, aber auch einzelne Parameter als Favoriten gekennzeichnet werden.
  - Auf Basis dieser Favoriten ist eine gefilterte Parameter-Sicht möglich.
  - Export und Import dieser Favoriten-Auswahl (ohne Werte)
- Massen-Engineering
  - Leistungsfähige Funktion für das Massen-Engineering von Geräten innerhalb eines Projekts (auch für unterschiedliche SIMOCODE pro Geräte)
  - Ausgewählte und entsprechend eingestellte Parameter können in einem Arbeitsschritt auf eine Gruppe von Geräten übertragen werden.
- Parameter-Assistent
  - Schrittweise Einstellung aller unbedingt notwendigen Geräteparameter
  - Einstellungen zu Feldbusschnittstelle, Gerätekonfiguration, Motorschutz und Überwachungsfunktionen werden abgefragt und gespeichert
  - Anwender kann den Wizard direkt aus dem Parameter Editor zum beliebigen Zeitpunkt aufrufen.
- Automatische Erzeugung von PLC-Variablen
  - Wenn STEP 7 zusätzlich installiert und ein SIMOCODE pro Geräte mit einer PLC vernetzt ist, werden für alle Eingänge und Ausgänge, die mit einer PLC über PROFIBUS oder PROFINET ausgetauscht werden, automatisch PLC-Variablen erzeugt.
  - Dank dieser Funktion kann der Anwender über symbolische Adressierung direkt auf diese Daten im Anwenderprogramm zugreifen.
- Übersichtliche automatische Anordnung der grafischen Blöcke
  - Platzierung der Blöcke in festen Spalten ähnlich dem EVA-Prinzip (EVA = Eingabe, Verarbeitung, Ausgabe)
- Im Grafikeditor können nun alle Parameter bearbeitet werden und nicht nur die Verbindungen
  - Direkt am Funktionsbock können Werte mit der Tastatur eingegeben werden

2.6 Engineeringoptionen

PLCSIM Advanced

- Synchronisation von PLCSIM Advanced mit Co-Simulations-Tools an Teilprozessabbildern zyklischer OBs (z. B. Weckalarm OBs)
- Unterstützung von azyklischen Diensten (RDREC/WRREC) und Alarmen (z. B. Prozessalarmen)
- Im TIA Portal projektierte Prozessalarme können über die API ausgelesen werden
- Einfaches Sichern und Wiedereinspielen der Software und Hardware-Konfiguration von PLCSIM Advanced Instanzen
- Der Speicherpfad der virtuellen SIMATIC Memory Card ist frei wählbar
- GUI Erweiterungen:
  - Autovervollständigung
  - Buttons für RUN / STOP und Memory Reset in der GUI von PLCSIM Advanced integriert
- Rechnerübergreifender Transfer von SIMATIC Memory Cards über die API
- Performanceverbesserung bei symbolischer Adressierung
- Parallel-Installation von PLCSIM und PLCSIM Advanced auf einem PC

User Management Component

- Zentrale, projektübergreifende Verwaltung von Benutzern in der Anlage
- Verwaltung von Benutzergruppen
- Import von Windows-Benutzern und Benutzergruppen ist möglich
- Effiziente Administration von Benutzern/-gruppen in einer Anlage
- Ausfallsicherheit durch redundante Auslegung einer UMC Domain
- Lastverteilung von Login Anforderungsschwallen durch mehrere UMC Stationen in einer UMC Domain
- Lizenzierung über die Anzahl von Benutzern

Multiuser Engineering

- Multiuser-Objekte werden beim Bearbeiten nun automatisch markiert. Das manuelle Markieren steht weiterhin zur Verfügung. Das automatische Markieren erhöht die Sicherheit beim gemeinsamen Arbeiten mit Multiuser Engineering.
- Das Arbeiten mit Multiuser Engineering ist jetzt auch offline und ohne direkte Serververbindung möglich.
- Die Multiuser-Editoren wurden durch Kommentarfunktionen erweitert zur Dokumentation von Art und Umfang der durchgeführten Änderungen.

- Beim Einchecken steht ein Filter für das Anzeigen von Objekten mit "Konflikten" zur Verfügung.
- Der Multiuser-Server enthält eine erweiterte Projekthistorie mit Wiederherstellungsfunktion. Dies ermöglich die Nachvollziehbarkeit des Projektverlaufs am Multiuser-Server. Die Meilensteine eines Projekts können kommentiert und gespeichert werden. Die Projekthistorie ist für eine erweiterte Auswertung exportierbar.
- Die Multiuser-Server V14 und V15 sind nebeneinander einsetzbar. Der Multiuser-Server V15 unterstützt auch TIA Portal-Projekte aus der V14, mit entsprechendem Funktionsumfang.
- Die externen Multiuser-Tools für die Konfiguration und Administration des Multiuser-Servers sind in jetzt allen Sprachen des TIA Portals verfügbar.

SIMATIC Visualization Architect

- Generierung von Meldungen, Meldeklassen und Meldegruppen
- Unterstützung der HMI-Objekte "Kurvenanzeige", "f(x)-Kurvenanzeige" und "f(t)-Kurvenanzeige"
- Unterstützung der Bildvorlagen

SIMATIC Energy Suite

- Visualisierung von Energiedaten in Energieobjekten erfordert keine zusätzlichen PowerTags in WinCC RT Professional.
- S7 Energieeffizienz-Monitor für Maschinen (neue S7-Anweisung in STEP 7: „EnS_EEm_Calc“ und „EnS_EEm_Report):
  - Zur produktionsbezogenen und standardisierten Ermittlung von Energieverbräuchen in Maschinen (gemäß VDMA 34179)
  - Einfache Integration in Maschinensteuerung (S7-1200/1500) und Vor-Ort-Visualisierung des Effizienzstatus
  - Automatische Langzeitmessungen (z. B. Charge, Schicht)
  - Effizienzprotokoll (.csv) zur Detailauswertung und Dokumentation
- Unterstützung von Sentron PAC 3200/4200 konfiguriert über den Hardware-Katalog (MDD) – bisher wurde ausschließlich GSDML unterstützt.

SIMATIC TIA Portal Openness

- Openness DLLs von V14SP1 und V15 im Lieferumfang
  Da die Openness DLLs von V14SP1 und V15 im Lieferumfang enthalten sind, laufen Applikationen, die auf V14SP1 basieren, ohne Änderung auch in V15. Um die Funktionen der V15 nutzen zu können, müssen Sie die DLL der V15 einbinden und die Applikation neu übersetzen
- Export und Import von SCL-Bausteinen
  Sowohl SCL-Bausteine als auch KOP- bzw. FUP-Bausteine mit SCL-Netzwerken können als XML exportiert und importiert werden.
- PLC Download
  Standard PLC S7-1500 können automatisiert geladen werden. Dabei wird ein Stoppen und Starten der PLC implizit durchgeführt. Schutzstufen-Kennwörter und Binding-Kennwörter können von der Appplikation übergeben werden.
- Prüfsumme lesen
  Die Prüfsumme des PLC-Standardprogramms kann von einer offline PLC gelesen werden.
- ProDiag
  Es können FBs, Instanzen und Zuweisungen für ProDIAG angelegt werden.
- Systemordner für UDTs
  Auf System-UDTs kann in der Systemgruppe innerhalb der Anwenderdatentypen zugegriffen werden.
- Bausteinnummerierung
  Die automatische Bausteinnummerierung kann ein- und ausgeschaltet werden. Weiterhin können Bausteinnummern geändert werden.
- Technologieobjekte
  Openness enthält Erweiterungen zum Laden und Speichern von Daten des TO-Kurvenschreibers und für neue TO Features.
- Startdrive
  DriveObjects und Telegramme können für die Antriebe SINAMICS G120 und SINAMICS S120 angelegt werden. Ausgewählte Drive-Parameter können online und offline eingestellt werden. Das Laden auf das Gerät ist möglich.
- SINAMICS in AutomationML
  V90 Slaves und SINAMICS G120 Profinet Kopfmodule können über AutomationML exportiert und importiert werden.

SIMATIC STEP 7 Safety

SIMATIC STEP 7 Safety Basic / Advanced V15, das leistungsfähige Optionspaket zur Programmierung der fehlersicheren S7-Controller für das Totally Integrated Automation Portal V15.

- Fehlersichere Arrays (Lesend) für Datentypen INT und DINT
  Mit Safety V15 stehen neue Anweisungen zum Lesen von fehlersicheren Integer- und Doubleinteger-Werten aus fehlersicheren Arrays zur Verfügung. Die Arrays können in globalen fehlersicheren Datenbausteinen (F-Global-DBs) angelegt werden und können bis zu 10.000 Elemente vom Datentyp INT oder DINT enthalten.
  Der Zugriff auf die Array-Elemente erfolgt durch die Anweisungen RD_ARRAY_I bzw. RD_ARRAY_DI. Die Anweisungen stehen für die fehlersicheren S7-1500 CPUs zur Verfügung.
- Hardware- und Software-Signatur
  Die F-Gesamtsignatur kennzeichnet eindeutig einen bestimmten Stand des Sicherheitsprogramms und der sicherheitsrelevanten Parameter der F-CPU und F-Peripherie.
  Neben der F-Gesamtsignatur erhalten Sie mit Safety V15, für die fehlersicheren S7-1200 und S7-1500 CPUs, zusätzliche Informationen hinsichtlich der fehlersicheren Software- und Hardwaresignatur. Diese zusätzlichen Signaturen ermöglichen eine bessere Differenzierung zwischen hardware- und softwarerelevanten Änderungen und vereinfachen so die Änderungsabnahme.
  Die F-Signaturen werden im Safety-Administration-Editor (SAE) angezeigt und sind Bestandteil des Sicherheitsausdrucks.
- Überlauferkennung (Overflow-Handling)
  Bei der Verwendung von Konvertern oder mathematischen Funktionen wie Additionen, Multiplikationen, etc. besteht die Möglichkeit eines Bereichsüberlaufs, welcher durch die fehlersichere Verarbeitung erkannt wird und einen STOP der CPU zur Folge hat.
  Dieser STOP kann durch eine sichere Überlauferkennung vermieden werden.
  Fehlersichere S7-300 / S7-400 CPUs erfordern hierbei die Platzierung einer entsprechenden Overflow-Anweisung „Statusbit OV abfragen“ im nachfolgenden Netzwerk.
  Für die fehlersicheren S7-1200 / S7-1500 CPUs kann eine Überlauferkennung wesentlich komfortabler und analog zum STEP 7-Standard programmiert werden.
  Allein durch die Beschaltung des Freigabeausgangs ENO wird eine Überlauferkennung programmiert und ein STOP der CPU im Falle eines Überlaufs kann vermieden werden.
  Liegt das Ergebnis der Anweisung außerhalb des für den Datentyp zulässigen Bereichs, dann liefert der Freigabeausgang ENO den Signalzustand "0". Das Ergebnis der Anweisung verhält sich dann, wie bei der entsprechenden Anweisung in einem Standard-Baustein.
  Die Überlauferkennung wird für folgende Anweisungen unterstützt:
  - ADD: Addieren
  - SUB: Subtrahieren
  - MUL: Multiplizieren
  - DIV: Dividieren
  - NEG: Zweierkomplement erzeugen
  - ABS: Absolutwert bilden (S7-1200, S7-1500)
  - CONVERT: Wert konvertieren

- Taktsynchroner Fail-safe OB
  Safety V15 ermöglicht durch taktsynchrone F-OBs die Anbindung von taktsynchronen PROFIsafe-Geräten wie z. B. SINAMICS S120 (ab FW V5.1) für minimale Reaktionszeiten und Jitter.
  Voraussetzung: F-CPUs S7-1500 ab Firmware-Version 2.0, die IRT unterstützen.
- Usability-Verbesserungen
  Safety Basic / Advanced bietet mit V15 einige Usability-Verbesserungen, welche die Erstellung von fehlersicheren Anwenderprogrammen erleichtern:
  - Rücklesen von fehlersicheren F-FB Out-Variablen auch auf S7-1200 und S7-1500 CPUs
  - Schreiben von F-FB Input Variablen analog zu STEP 7 Standard
  - Startwerte von Instanz-DBs können geändert werden
- Weitere Neuerungen
  - Die Überwachung von systemintegrierten F-IO-DBs und F-Ablaufgruppeninfo-DBs mittels ProDiag wird unterstützt.
  - DINT -> INT Konverter (S7-1200, S7-1500)
  - ABS: Absolutwert bilden (S7-1200, S7-1500)

Siehe auch

SIMOCODE ES (Seite 28)

2.7 Runtimeoptionen

Beschreibung

Folgende neue Optionen werden mit dem TIA Portal ausschließlich als Runtimeoption angeboten. Das Produkt ProDiag ist bereits in die bestehenden Produkte STEP 7 und WinCC integriert und muss ausschließlich bei Verwendung auf der realen Hardware lizenziert werden. Im Vergleich zu den Engineeringoptionen sind die Runtimeoptionen versionsunabhängig.

SIMATIC ProDiag

- S7-GRAPH Bausteine ab Version 5.0 ermöglichen das Übersetzen von GRAPH-Schritt- und GRAPH-Transitionsnamen sowie das Zurücksetzen der Erstwerte von Interlocks und Transitionen.
- Verbesserungen der Mehrfachauswahl in globalen Datenbausteinen, Funktionsbausteinen und Variablentabellen
  - Neue Überwachungen hinzufügen, auch wenn die Mehrfachauswahl nicht-boolesche Elemente enthält
  - Beim Löschen werden alle Überwachungen gelöscht, auch wenn ein Element mehr als eine Überwachung enthalten hat
  - Beim Kopieren einer Überwachung werden alle Eigenschaften der Überwachung mitkopiert. Beim Einfügen werden neue Überwachungen mit den kopierten Eigenschaften angelegt bzw. bereits bestehende Überwachungen werden überschrieben. Wenn ein Element mehrere Überwachungen enthält, werden diese gelöscht und eine oder mehrere neue Überwachungen mit den kopierten Eigenschaften angelegt.
  - Beim automatischen Ausfüllen der Zellen werden in den markierten Zellen eines globalen Datenbausteins oder eines Funktionsbausteins neue Überwachungen angelegt und die Eigenschaften der ausgewählten Überwachung(en) werden übernommen.
- Innerhalb der Variablentabelle, der FB-Bausteinschnittstelle oder globalen Datenbausteinen wird in der Spalte "Überwachung" angezeigt, wie viele Überwachungen der Operand enthält.
- Die globale Suche findet jetzt auch die ProDiag Texte der angelegten Überwachungen aus dem Inspektorfenster, aus den ProDiag-Funktionsbausteinen und aus den ProDiag-Überwachungseinstellungen.

OPC UA

Für S7-1500 CPUs ab Firmware V2.5 können Sie mit einer entsprechenden Runtime-Lizenz von folgenden Erweiterungen des integrierten OPC UA-Servers profitieren:

- Methoden können über das Anwenderprogramm einer S7-1500 CPU bereitgestellt werden. Diese Methoden nutzen OPC UA-Clients, um z. B. einen Fertigungsauftrag über den Methodenaufruf von der S7-1500 CPU abarbeiten zu lassen.
- Importieren von standardisierten OPC UA-Informationsmodellen (Companion Spezifikationen). Damit nutzen Sie einheitliche Schnittstellen für eine große Auswahl von Geräten oder Maschinen - z. B. für die Modellierung von RFID-Readern oder Spritzgießmaschinen. Beispiele sind hier die Companion Spezifikationen AutoID und Euromap 77.

Was ist neu im TIA Portal V14 SP1

3

3.1 Einleitung

Beschreibung

Alle wichtigen Neuheiten gegenüber der bestehenden Version V14 sind hier zusammengefasst aufgeführt. Weitere Details zu den einzelnen Themen können Sie den einzelnen Kapiteln der Produktdokumentationen entnehmen.

3.2 SIMATIC STEP 7

Beschreibung

Alle wichtigen Neuheiten von STEP 7 sind anhand des Engineering-Workflows zusammengefasst aufgeführt.

Hardware-Konfiguration

- Neue Systemstromversorgung PS 60W 24/48/60VDC HF für die S7-1500 mit Energiepufferung zur Datensicherung von bis zu 20 MB in der CPU
- Tabellarische Konfiguration der IO-Kanäle für Module der ET 200MP DI HF und ET 200MP DO HF inkl. Copy & Paste und Auto-Ausfüllfunktion
- Bearbeiten von Gerätenamen in den grafischen Sichten
- Verbessertes Zoomverhalten und optimierte (Multi-) Selektion in der grafischen Netz- und Topologiesicht
- Übersicht von IO-Variablen eines ganzen Gerätes (z. B. einer ET 200SP-Station)
- Einfaches Entfernen ungenutzter GSD-Dateien im Projekt
- Wizard-Erweiterung für UDP Multicast bei der Open User Communication (OUC) mit S7-1500

Editoren für Programmiersprachen

- Im SCL-Programmcode können strukturierte Variablen und ARRAY-Elemente per Drag&Drop durch andere Variablen ersetzt werden.
- Im SCL-Editor kann zwischen Einfüge- und Überschreibmodus gewechselt werden.
- In SCL können selektierte Programcodesegmente einfach mit bestimmten Strukturelementen, wie z. B. "IF..Then", umschlossen werden.
- Der Detailvergleich einzelner Bausteine kann direkt aus der Projektnavigation gestartet werden (Offline/Offline, aber auch Online/Offline).
- In KOP/FUP kann der Variablenkommentar direkt an der Verwendungsstelle eingeblendet werden.
- Die F7-Funktion ("Baustein/PLC-Datentyp öffnen…") kann auch mit Fokus innerhalb von Editoren gestartet werden.
- Externe Quelldatei aus Bausteinen generieren: Der Export von Objekten als externe Quelle berücksichtigt in der Bausteinschnittstelle definierte PLC-Datentypen (UDTs), Multiinstanzen und alle aufgerufenen Bausteine.
- In der Task Card "Anweisungen" können alle verwendeten Anweisungen einer Steuerung auf den aktuellsten Versionsstand gebracht werden.

Neue Anweisungen

- Mit den neuen Anweisungen "SCATTER" und "GATHER" für die S7-1200/1500 können die Werte zwischen Variablen vom Datentyp BYTE, WORD, DWORD, LWORD und Variablen-ARRAYs vom Datentyp BOOL umkopiert werden. Damit können z. B. Steuer- und Statuswörter einzelner Anlagenteile einfach in die einzelnen Bitsignale zerlegt und wieder zusammengesetzt werden.
- Mit Hilfe der Anweisung "GetSMCInfo" kann man im Anwenderprogramm Informationen über die gesteckte SIMATIC Memory Card, wie die Speichergröße, den belegten Speicherplatz oder die Anzahl der bereits durchgeführten Schreib-/Löschvorgänge (Prozentangabe), auslesen.
- Die Anweisung "Polyline" bildet den Eingangswert über eine Kennlinie auf den Ausgangswert ab. Die Kennlinie wird als Polygonzug (mit bis zu 50 Stützpunkten) definiert.

Sprachinnovationen

- PLC-Datentypen (UDT)- und ARRAY-Kommentare können instanzspezifisch geändert werden.
- Die Zuweisung verschiedener PLC-Datentypen (UDTs) mit gleichem Aufbau, aber unterschiedlichen symbolischen Namen, wird unterstützt.
- S7-GRAPH Bausteine ab Version 4.0 ermöglichen die Erstwerterfassung der gestörten Operanden. Die Aktualwerte der Signalzustände von booleschen Operanden und die Ergebnisse von Vergleichern in Interlocks und Transitionen werden ab dem Zeitpunkt der Aktivierung der Erstwerterfassung kontinuierlich aufgezeichnet, auch wenn kein Fehler auftritt. Die Signalzustände werden pro Zyklus im GRAPH-Instanzdatenbaustein gespeichert. Es können max. 32 Signalzustände pro Interlock bzw. pro Transition eines GRAPH-Schritts aufgezeichnet werden.

Bibliotheken

- Projektbibliotheken und globale Bibliotheken können jetzt separat in weitere Fremdsprachen übersetzt werden.
- Vereinfachtes Hochrüsten von Typen (Bausteine, PLC-Datentypen) in einer bestehenden Projektbibliothek.
- Die Anwenderdefinierte Hilfe kann jetzt auch im Kontext der Projektsprache gestartet werden. Bis V14 war das nur im Kontext der TIA Portal-Sprache möglich.

Meldungen / Diagnose

- Die Darstellung der Meldungsanzeige im Inspektorfenster wurde optimiert und mit neuen Funktionen, wie "Meldungen empfangen" und "Meldungen einfrieren", funktional erweitert.

Migration

- Die in SCL-Quellen eingesetzten lokalen Konstanten für ARRAY-Grenzen bleiben bei der Migration von STEP 7 V5.x nach TIA Portal erhalten.

Trace

- Umschaltung des Displayformats (z. B.: Ein WORD kann hexadezimal, als signed oder unsigned INT, dargestellt werden)
- Die Scrollbar der X-Achse erhält ein kleines Vorschaufenster ("Zeitbereichsanzeige"), mit dem man erkennen kann, welchen Bereich man gerade gezoomt hat.

Systemfunktionen

- CAx-Daten auf Basis von XML, zum Austausch von z. B. ECAD-Systemen, können exportiert und auch wieder importiert werden.
- Frei durch den Anwender definierbare Filter für Querverweise, um z. B. Mehrfachzugriffe auf Variablen herauszufinden, werden unterstützt.
- Über einen Passwort-Provider kann das TIA Portal an eine externe Passwortverwaltung angebunden werden. Dies bietet den Vorteil, dass Passwörter nicht preisgegeben werden müssen. Somit wird die Sicherheit des Programmcodes erhöht.

SIMATIC Ident

Das Technologieobjekt (TO) für SIMATIC Ident verbindet die Hardware-Projektierung von Kommunikationsmodulen und Readern mit der Programmierung einer S7-1200 und S7-1500 Steuerung. Dabei verhindert das TO falsche Einstellungen und erkennt Fehler in der Parametrierung.

Hardware-Konfiguration:

- Jedes Kommunikationsmodul/Reader erhält vom Anlagenbauer einen "eindeutigen Namen" mit dem er von Inbetriebnehmern und Programmierern gleichermaßen identifizierbar ist.
- Die Parametrierung jedes Kommunikationsmoduls, Readers oder Lesestelle wird über das TO eingestellt.
- Mit der Schaltfläche "Auto-Konfiguration" können die Ident-Geräte mit einem Klick für den Betrieb mit dem TO konfiguriert werden.

Neue Anweisungen:

- Alle Anweisungen für SIMATIC Ident wurden für die Zusammenarbeit mit dem TO überarbeitet. Sie finden die Anweisungen im TIA Portal unter "Optionale Pakete" > "SIMATIC Ident".
- Die Hardware-Adressierung eines Readers an den Anweisungen ist so geändert, dass die Reader über den "eindeutigen Namen" ausgewählt werden, der im TO festgelegt wurde. Das bisherige fehlerträchtige herausfinden, notieren und übertragen von HW-Identifier und logischer Adresse entfällt somit.
- Optionale Eingänge an den Anweisungen sind ausgeblendet. Somit erkennt der Anwender sofort die wichtigen Parameter, die für eine Funktion notwendig sind.
- Der Programmierer muss sich nicht mehr um die Parametrierung der Hardware kümmern. Unabhängig vom angeschlossenen Reader wird beim PLC-Anlauf nur noch die Anweisung RESET_READER, welche keine weiteren Input-Parameter besitzt, benötigt. Die notwendigen Parameter holt sich die Anweisung aus einem Datenbereich, der mit dem TO parametriert wurde.

Diagnose:

- Mit dem TO können die letzten Meldungen, welche von den Anweisungen an die Applikation gemeldet wurden, angezeigt werden.
- Ein Zeitstempel und der genaue Fehlertext helfen für eine schnelle Analyse des Fehlers.

3.3 SIMATIC WinCC

Funktionserweiterung – Openness

- Unterstützung von HMI Bildbausteinen
- Unterstützung von HMI Slide-in-Bildern
- Unterstützung von HMI Pop-up-Bildern

Erweiterungen für SIMATIC HMI Panels

- SIMATIC HMI BASIC Panels:
  - Unicode CSV-Export von Rezepten für Basic Panels
  - IP-Adressen von Steuerungen und zusätzlichen SIMATIC Komponenten (z. B. CPs) über das Bediengerät ändern.
  - Unterstützung von chinesischen, japanischen, koreanischen, russischen und arabischen Schriftzeichen im Browser.
- Aktualisierung der getesteten Netzwerk-Kameras (AXIS M1013, AXIS M1011, D-Link DCS-942L, D-LINK 4701E, Siemens CCMS2025).
- Installieren/deinstallieren von Word, Excel, Win Mediaplayer, IE über ProSave.
- Unbegrenzte Anzahl an verwendbaren Skripten für Comfort.
- Konfiguration für Startup (zusätzlicher Wert "60s" für Startzeit der RT und Systemfunktion für Restart Panel).
- Zusätzliche Systemfunktionen im Editor "Zonen" für Mobile Panels.

Erweiterungen für SIMATIC Runtime Advanced

- Kamera-Anzeige zur Darstellung von Live Video Streams von Netzwerk-Kameras
- Erweiterungen der PDF-Anzeige (Suchen und Kopieren)
- Lizenzfreie Nutzung des Sm@rtServer auf Runtime Advanced (analog zu Comfort Panels)

Integration von SINUMERIK Operate (GUI und Daten) in Runtime Advanced

- Zugriff auf SINUMERIK Operate-Bedienoberfläche über neue NC-Bedienanzeige in Bildern von RT Advanced (auch Multitouch-Bedienung)
- Zugriff auf Daten der SINUMERIK NCU über separaten SINUMERIK-Kanal

Erweiterungen für SIMATIC Runtime Professional

- Unterstützung OPC UA DA Client
- Neue VBS-Methode für Bildfenster. Damit haben Sie die Möglichkeit, die Bedienbarkeit eines Bildfensters temporär auszuschalten.
- Individualisierung der Rezepturanzeige. Damit haben Sie die Möglichkeit, Eigenschaften der Controls (Schaltflächen oder Tabellenelemente) anzupassen.

3.4 SINAMICS Startdrive

Startdrive für G120

- Optimierter und erweiterter Inbetriebnahme-Assistent:
  - Zusammenfassung der beiden bisherigen Assistenten für die Standard-Inbetriebnahme und den Betrieb unter einer SIMATIC Motion Control Achse in einem Assistenten
  - Integration der Auswahl von Telegramm, Bremswiderstand, motorseitigem Filter, Temperatursensor sowie Auswahl des Motors auch über Motor-Code-Nummer
- Online/Offline Parameter-Vergleich
- Unterstützung von Windows 10

Startdrive Option für S120

- Automatische Hardware-Konfiguration
- Unterstützung von Drive-based Extended Safety mit grafischen Parametriermasken
- Unterstützung von Torque-Motoren (1FW3, 1FW6)
- Vernetzung mit ET200 SP CPU
- Unterstützung von Windows 10

3.5 Engineeringoptionen

SIMATIC STEP 7 Safety

SIMATIC STEP 7 Safety Basic/Advanced V14 SP1, das leistungsfähige Optionspaket zur Programmierung der fehlersicheren S7-Controller für das Totally Integrated Automation Portal V14 SP1.

- Stationsupload
  Der konsistente Hardware- und Software-Upload ermöglicht ein Laden der Projektdaten (inkl. sicherheitsrelevanter Projektdaten) von einer fehlersicheren S7-1500 CPU*) in ein PG/PC. Die Projektdaten werden ohne zusätzlichen Generierungsschritt geladen und stehen konsistent im TIA Portal zur direkten Weiterarbeit zur Verfügung.
  *) ab Firmware Version V2.1
- Konfigurationssteuerung
  Mit Safety V14 SP1 wird die Konfigurationssteuerung (Optionenhandling) für SIMATIC F-Peripherien in folgenden Konfigurationen unterstützt:
  - dezentral an einer F-CPU S7-300/400/1200/1500
  - zentral an einer F-CPU S7-1500 (inkl. fehlersichere ET200SP CPUs)
- Projektübergreifende IO-Controller-I-Device-Kommunikation
  Safety V14 SP1 ermöglicht eine projektübergreifende Kommunikation von fehlersicheren S7-1200/1500/1500 Software Controller zu S7-300/400/1200/1500 I-Devices. Das I-Device-Projekt kann einfach mit STEP 7 Safety V13 (oder höher) oder auch S7 Distributed Safety V5.4 erstellt, als GSD-Datei exportiert, und in das IO-Controller-Projekt importiert werden.

- Globaler F-IO-Status-Baustein
  Fehlersichere F-Peripherien bieten eine Vielzahl von Status-Informationen, über den Kanal- und Baugruppenzustand, mittels der Variablen QBAD und den Wertstatusbits.
  Um auch in Automatisierungssystemen mit sehr großen Mengengerüsten einen möglichen Kanal-, Kommunikations- oder F-Peripheriefehler zentral zu erfassen und weiterzuverarbeiten, haben Sie im Safety Administration Editor die Möglichkeit, für jede F-Ablaufgruppe einen Standard-FB anlegen zu lassen. Im angelegten Standard-FB werden die Wertstatusbits und die Variablen QBAD der in der jeweiligen F-Ablaufgruppe verwendeten F-Peripherie ausgewertet und das Ergebnis der Auswertung am Ausgang "QSTATUS" zur Verfügung gestellt.
  Den Standard-FB können Sie im Standard-Programm verwenden.
- Openness
  Sie können in STEP 7 Safety, wie im Standard, den Openness-Befehlssatz nutzen. Voraussetzung ist, dass kein Passwort für die sicherheitsrelevanten Projektdaten eingerichtet ist. Der Einsatz von Openness im Produktivbetrieb ist nicht zulässig.
  Unter anderem wird Folgendes unterstützt:·
  - Arbeiten mit Bibliotheken (Elemente löschen, kopieren, instanziieren, upgraden, etc.)
  - Import von Variablen, Variablentabellen, PLC-Datentypen (UDTs), etc.
  - Übersetzen von SW, Blöcke, PLC-Datentypen (UDTs).
  - F-CPUs und F-Peripherie stecken/löschen
  - F-CPUs und F-Peripherie aus Kopiervorlagen kopieren/löschen
  - Netze konfigurieren
  - Software übersetzen (inkl. Sicherheitsprogramm)·
  - F-Parameter der F-CPU lesen/projektieren
  - F-Parameter der F-Peripherie lesen/projektieren
  - Fehlersichere Variablen in der PLC-Variablentabelle lesen, deklarieren oder löschen
  - Projekt auf die neuesten Typ-Versionen von F-Bausteinen aktualisieren

SINUMERIK STEP 7 Toolbox

Die SINUMERIK STEP 7 Toolbox V14 SP1 ist ein Optionspaket zu SIMATIC STEP 7 Professional V14 SP1 (TIA Portal) mit zusätzlichem Setup.

Die SINUMERIK STEP 7 Toolbox V14 SP1 ermöglicht die Projektierung der SINUMERIK Hardware im TIA Portal durch folgende Funktionalitäten und Tools:

- Ergänzung des Hardware-Katalogs durch folgende Baugruppen der SINUMERIK 840D sl (ab Firmware V4.5 SP2):
  - NCU 710.3B
  - NCU 720.3B
  - NCU 730.3B
  - NX10.3
  - NX15.3
- Ergänzung des Hardware-Katalogs durch die Baugruppe ADI4
- SINUMERIK-PLC-Grundprogramm für die Versionen
  - V4.5.x.x
  - V4.7.x.x
  - V4.8.x.x
    Mit der SINUMERIK STEP 7 Toolbox V14 SP1 wird automatisch das PLC-Grundprogramm in Form der Systembibliothek "SINUMERIK 840D sl PLC Basic Program" installiert.
- Unterstützung der Hardware-Migration
- Erstellung von SINUMERIK PLC-Inbetriebnahmearchiven, PLC Hardwarehochrüstarchiven und Nachladearchiven
- Exportieren von PLC-Symbolen für SINUMERIK Operate
- Importieren von SINUMERIK Anwenderalarmtexten
- Erstellen von SINUMERIK-PLC-Archiven
- Unterstützung von PROFINET IO IRT für NCK
- Unterstützung von SINUMERIK Safety Integrated und Safety Integrated plus
- NC-VAR-Selector (externes Tool)

3.6 Runtimeoptionen

Beschreibung

Folgende neue Optionen werden mit TIA Portal V14 SP1 ausschließlich als Runtimeoption angeboten. Das Produkt ProDiag ist bereits in die bestehenden Produkte STEP 7 und WinCC integriert und müssen ausschließlich bei Verwendung auf der realen Hardware lizenziert werden. Im Vergleich zu den Engineeringoptionen sind die Runtimeoptionen versionsunabhängig.

SIMATIC ProDiag

- S7-GRAPH Bausteine ab Version 4.0 ermöglichen die Erstwerterfassung der gestörten Operanden.
- Mithilfe des Objekts "PLC-Code-Anzeige" können die Erstwerte auf dem HMI angezeigt und im Fehlerfall die fehlerhaften Operanden sofort erkannt werden (Kriterienanalyse). Die Erstwerte bleiben solange gespeichert, bis sie bei einem erneuten Fehler überschrieben werden.
- Für die Erstwerterfassung werden alle Anweisunge unterstützt, die bei einem S7-GRAPH-Funktionsbaustein für Transitionen und Interlocks unterstützt werden. Bei den Bitverknüpfungen wird der Zustand des Operanden und bei den Vergleichern das Ergebnis des Vergleichs gespeichert.

SIMATIC Energy Suite S7-1500

SIMATIC Energy Suite als integrierte Option für das TIA Portal verknüpft erstmals Energiemanagement effizient mit der Automatisierung und bringt somit Energietransparenz in die Produktion. Zusätzlich wird durch die vereinfachte Projektierung von energiemessenden Komponenten der Projektierungsaufwand deutlich reduziert.

Funktionen:

- Integriertes und intuitives Konfigurieren des Energiemanagements
- Automatische Erzeugung des PLC-Energieprogramms für S7-1500 für Basis Energiedaten (Energie und Leistung)
- Archivierung auf WinCC Runtime Professional oder auf PLC-interner SIMATIC Memory Card
- Nahtlose Anbindung an SIMATIC Energy Manager PRO

Neu in V14 SP1:

- Erweiterung der automatischen PLC-Programmerzeugung um zusätzliche Energiedaten (Ströme, Spannungen, Frequenz)
- Neben dem manuellen Energiedatenexport wird nun auch der automatische Export angeboten (z. B. Start des Exports der Energiedaten der letzten Woche jeden Montag um 00:15 Uhr)
- Programmgenerator: Detailliertere Darstellung des Generierfortschritts und Meldungen bei Konfigurationsfehlern.

Was ist neu im TIA Portal V14

4

4.1 Einleitung

Beschreibung

Alle wichtigen Neuheiten gegenüber der bestehenden Version V13 SP1 sind hier zusammengefasst aufgeführt. Weitere Details zu den einzelnen Themen können Sie den einzelnen Kapiteln der Produktdokumentationen entnehmen.

4.2 SIMATIC STEP 7

Beschreibung

Alle wichtigen Neuheiten von STEP 7 sind anhand des Engineering-Workflows zusammengefasst aufgeführt.

Hardware-Konfiguration

- Unterstützung der S7-1500 T-CPU Familie, mit den Motion Control Funktionen "absoluter Gleichlauf" und "Kurvenscheibe".
- Unterstützung der neuen CPU 1518(F)-4 PN/DP ODK (Open Development Kit), welche es erlaubt, über das SIMATIC ODK 1500S erstellte C/C++ Funktionen in das Anwenderprogramm einzubinden
- Unterstützung der CPU 1516pro (F)-2 PN
- Unterstützung des neuen S7-1500 Failsafe Software Controllers
- Das Gruppieren von IO-Geräten / Devices in der Projektnavigation zum schnellen Auffinden und technologischen Zusammenführen von Geräten ist möglich.
- Die Anordnung der Geräte in der topologischen Sicht kann mit der Netzwerksicht synchronisiert werden.
- Für einen einfachen und schnellen Zugang zu oft benutzten Modulen können nun eigene Profile im Hardware- Katalog definiert werden.

Editoren für Programmiersprachen

- Für die neuen Technologieobjekte Nocken, Nockenspur und Messtaster stehen eigene Editoren zur Verfügung. Zusätzlich gibt es für die S7-1500T Steuerungen einen komfortablen Kurvenscheibeneditor.
- In KOP / FUP-Baustein können SCL-Netzwerke eingefügt werden.
- Im SCL-Programmcode können "Regions" zur besseren Strukturierung und Navigation genutzt werden.
- Bausteinparameter können beim Aufruf des Bausteins in KOP oder FUP ausgeblendet werden.
- Der Vergleich von PLC-Programmen wurde grundlegend überarbeitet und ermöglicht einen schnellen und komfortablen Quellcodevergleich mit vielen Filtermöglichkeiten.
- Netzwerke in den Programmiereditoren für KOP / FUP / AWL werden automatisch so dargestellt, wie sie beim letzten Schließen eines Bausteins angezeigt wurden.

Sprachinnovationen

- Mit ARRAY[*] können ARRAYs variabler Länge übergeben werden.
- In der Bausteinschnittstelle können Multiinstanzen in ARRAYs deklariert und in Schleifen aufgerufen werden. Dies hilft, um den Programmcode stark zu reduzieren und eine bessere Lesbarkeit und effizientere Programmierung bei der Abarbeitung gleicher Programmobjekte zu realisieren.
- Der Datentyp DB_ANY kann auch für Technologieobjekte verwendet werden. Damit lassen sich Technologieobjekte flexibler im Programm hantieren. Ein "ARRAY of DB_ANY" kann beispielsweise eine Liste von Positionierachsen darstellen.
- Instanzen können als Parameter in der Bausteinschnittstelle im Abschnitt "InOut" übergeben werden, sodass erst beim Aufruf eine dedizierte Instanz ausgewählt werden muss.
- Meldetexte und Kommentare für bis zu 3 Sprachen können auf die PLC S7-1500 geladen und online gewechselt werden.

Meldungen / Diagnose

- Mit Hilfe der neuen Anweisung "Get_Alarm" können Meldungen direkt aus dem Meldeserver der S7-1500 CPU ausgelesen und z. B. an ein übergeordnetes Meldesystem (z. B. ein Leitsystem) gesendet werden.
- Bei der Anweisung "GET_DIAG" werden im Mode 1 die Diagnosedaten in der Struktur DIS ausgegeben. Das Bit 15 des Parameters IOState dieser Struktur liefert den Signalzustand "1", wenn ein Netzwerk- / Hardware-Fehler aufgetreten ist.

Trace

- Mit der Tracefunktion ist es möglich, unterschiedliche Messungen in einem gemeinsamen Diagramm anzuzeigen ("überlagerte Messungen").
- Außerdem können beim Trace Messungen nach Abschluss netzausfallsicher auf der Speicherkarte abgelegt und bei Bedarf automatisch neu gestartet werden.

Onlinefunktionen

- Neustrukturierung der Toolbar im DB-Editor mit der Möglichkeit, komplette Datenbausteine mit den definierten Startwerten zu reinitialisieren.
- Strukturen im Baustein können mit Ihrer Aufrufumgebung direkt im Inspektorfenster beobachtet werden.

Systemfunktionen

- Die neue globale Suche erlaubt ein projektübergreifendes Suchen von PLC- und HMI-Objekten. Zusätzlich kann das Suchergebnis per Filter eingeschränkt werden. Auch eine "GoTo"-Funktion steht zur Verfügung.
- TIA Portal Einstellungen können ex- bzw. importiert und auf einem zentralen Server abgelegt werden. Ist z. B. ein Serverpfad vorhanden, werden bei jedem Neustart des TIA Portals automatisch die abgelegten Einstellungen als "Voreinstellung" übernommen.

- Innoviertes Hilfe-/Informationssystem mit Registertechnologie, verbesserter Suchfunktion und der Möglichkeit, Informationen nach Gerätefamilien zu filtern.
- Die neue Querverweisliste bietet einen schnellen Überblick über die Verwendung von Objekten und Geräten innerhalb des Projektes:
  - Unterstützung von überlappenden Zugriffen (PLC-Variablen)
  - Unterstützung von Know-How-geschützten Bausteinen
  - Zugriff auf einzelne Elemente eines ARRAYs
  - Darstellung von Objekten, die sowohl in HMI als auch in PLC-Geräten vorkommen
  - Freie Eingabe von zu suchenden Objekten in der Querverweisliste möglich
  - Diverse Filtermöglichkeiten
  - Updates in Firmennetzwerken komfortabel und zentral zur Verfügung stellen.
  - Mit dem TIA Automation Corporate Configuration Tool kann ein unternehmenseigener Server konfiguriert werden. Auf dem Server werden die verfügbaren Updates / Support Packages abgelegt und stehen so den Anwendern zur Verfügung.

S7-PLCSIM

- Schneller Start von S7-PLCSIM im Kompakt-Mode und ohne Simulationsprojekt.
- Geräte können im Simulationsprojekt getauscht werden, sodass bestehende SIM-Tabellen und Sequenzeinträge ganz einfach weiter benutzt werden können.
- Sequenzen können auf Basis einer Triggerbedingung automatisch gestartet werden.
- Sequenzen können temporär deaktiviert werden.
- Dezentrale IO-Geräte können in der Geräteansicht simuliert werden.

Motion Control-Anwendungen

- Vorkonfigurierte und integrierte Anbindung und Projektierung des Antriebs SINAMICS V90 PN an ein Motion Control-Technologieobjekt der SIMATIC S7-1500.
- Automatischer Datenabgleich zwischen der Steuerung und dem Antrieb.

Siehe auch

Engineeringoptionen (Seite 58)

Runtimeoptionen (Seite 61)

4.3 SIMATIC WinCC

Bilder projektieren

- Neue Ansicht auf Distributed I/Os des Systemdiagnose Controls.
- Verbesserungen bei den Controls:
  - Neue Scroll-Balken mit konfigurierbaren Farben.
  - Dynamische Anzahl der Einträge im Dropdown Menü beim Rezeptur Control.
  - Erweiterung des Slider, Bar, Gauge Controls mit bis zu vier Grenzwerten.
- Neue Schriftart "Siemens TIA Portal Icons".
- Grafikbibliothek wurde um neue Grafiken ergänzt und Struktur komplett überarbeitet.

Arbeiten mit Variablen

- Symbolisch Adressmultiplexen von PlcUDT Instanzen innerhalb einer ARRAY-Variablen eines Datenbausteins.
- Hierarchisches IntelliSense bei der Anbindung von PLC-Daten an HMI-Variablen.

SIMATIC Panels und SIMATIC WinCC Runtime Advanced

- Transfer der Projektierung per USB Stick / SD Karte für Basic Panels 2ndGeneration, Comfort Panels, Outdoor Panels, Mobile Panels und WinCC Runtime Advanced.
- Sm@rt Server Option für die Basic Panel 2. Generation.
- Die Sicherheit von folgenden Funktionen wurde verbessert:
  - Folgende Komponenten werden signiert:
    Comfort und Mobile Panel: Runtime, ProSave Optionen und Runtime AddOns.
    Basic Panels 2. Generation: Image des Panels
  - Sicherheit der in der Runtime verwendeten Passwörter
  - Webbrowser Verbindung zu einem Sm@rtServer über HTTPS Protokoll möglich und sichere Verbindung von einem Web SmartClient zum Sm@rtServer möglich.
- Die maximale Anzahl von Powertags in WinCC Runtime Advanced wurde auf 16K erhöht.

WinCC Runtime Professional

- WebUX Option
  Plattform- und browserunabhängiges Bedienen und Beobachten über das Web.
- Prozess Diagnose Option
  Die Prozessdiagnose gibt beim Auftreten einer Prozessstörung Informationen zu Fehlerort und Fehlerursache und unterstützt bei der Fehlerbehebung.
- Erweiterung: Mengengerüste 1200/1500er PLCs
  Die Anzahl der Verbindungen zur S7-1200/1500 beträgt nun 64 Verbindungen.

- Erweiterung: Multi touch für bestimmte Controls
  Die Geste zum Verschieben im Anzeigebereich von Controls ist nun anwendbar (Trend-based Controls, Grid-based Controls, Axis-Control).
- Vereinheitlichung: Lizenzen für WEBNAVIGATOR, DATAMONITOR und WebUX
  Der Lizenztyp der Weboptionen wird durchgängig Coutable. Zukünftig werden diese SCADA-Optionen ab V7.4 bzw. V14 einheitliche MLFBs haben und deshalb wird es für diese Produkte keine eigenen MLFBs für Runtime Professional mehr geben.

Siehe auch

Engineeringoptionen (Seite 58)

Runtimeoptionen (Seite 61)

4.4 SINAMICS Startdrive

Vereinfachte Inbetriebnahme und vereinfachter (Tele-)Service

Erweiterte Routing-Funktionalität für die Kommunikation zwischen Startdrive und Antrieb:

- Routing über verschiedene PROFINET-Systeme hinweg
- Routing von PROFIBUS auf PROFINET

Einfaches Anpassen eines Projekts an die Maschine

Upgrade G120 Firmware-Version im Startdrive-Projekt (beginnend mit Firmware-Version 4.4)

Wiederverwendung der Parametrierung über verschiedene Leistungsbereiche hinweg

Auch für G120C

Schneller Zugang zum Startdrive Download-Paket in SIOS

- Keine Wartezeit für die Exportkontrolle
- Beginnend mit Startdrive V13 SP1 entfällt die Exportbeschränkung (Exportkennzeichen: AL=N, ECCN=N)

Effizientere Projektierung

Optimierte Übersetzungszeit der S7-CPU bei vernetzten mit Startdrive projektierten Antrieben

Veränderbarkeit der mitgelieferten S7-Datenbausteine zur Antriebsansteuerung

Drive Library jetzt ohne Know-how-Schutz

Vereinfachte Inbetriebnahme

Die Vergabe des PROFINET-Gerätenamens ist in der Projektnavigation und der HWCN-Netzsicht über das Kontextmenü des Antriebs einfach durchführbar.

Vereinfachter Update-Prozess

Automatische Installation von Startdrive HSPs und Updates über den TIA Portal Updater

Unterstützung neuer Hardware

Neue 1FP1 Synchron Reluktanz-Motoren werden unterstützt.

Keine manuellen Updates erforderlich

HSPs für Sinamics Firmware V4.7.3 und V4.7.6 sind integriert.

4.5 SIMOTION SCOUT TIA

Beschreibung

Die SIMOTION P – Plattform (PC-Based) steht nun auch im TIA Portal zur Verfügung. (Geplant ca. 3 Monate nach Freigabe von STEP 7.)

- Ein Upload der Hardwarekonfiguration für SIMOTION Geräte ist möglich.
- Ergänzende PROFINET-Funktionen für SIMOTION Geräte:
  - Shared-I-Device
  - Shared-Device
  - Seamless Media Redundancy (MRPD)
  - "Check of VendorID/DeviceID" und "Overwriting of the NameOfStation" (Automatische Konfiguration der Topologie)
  - Addresstailoring (Modulare Maschinen)

4.6 SIMOCODE ES V14

Routing über Ethernet - PROFINET

Mit SIMOCODE ES V14 wird das Routing zu SIMOCODE pro Geräten mit PROFINET unterstützt. Geräte werden jetzt auch von einer Engineering Station mit SIMOCODE ES V14 Premium über Ethernet erreicht, wenn diese nicht direkt mit dem PROFINET Netzwerk verbunden ist, an dem die SIMOCODE pro Geräte angeschlossen sind. Dadurch verbessern sich die Einsatzmöglichkeiten der Software in Anlagen, bei denen SIMOCODE pro Geräte mit PROFINET zum Einsatz kommen.

Migration von SIMOCODE ES 2007 Gruppendateien

Mit SIMOCODE ES 2007 erstellte Projektierungen, in denen Parameterdateien für die SIMOCODE pro Geräte zu sogenannten Gruppendateien zusammengefasst wurden, können nunmehr direkt und ohne zusätzliche Projektierungsschritte in ein und dasselbe TIA Portal V14 Projekt migriert werden. Durch diese Funktion verbessert sich die Handhabung der Software bei der Migration von SIMOCODE ES 2007 nach SIMOCODE ES V14 TIA Portal.

Verbesserte Editor-Funktionen für Pläne mit SIMOCODE pro Verschaltungen

Mit SIMOCODE ES V14 Standard und Premium können CFC-basierte Pläne ganz oder auszugsweise kopiert und als Bibliothekselemente gespeichert und wiederverwendet werden. Durch diese Funktion wird die Handhabung häufig wiederkehrender Projektierungen für typische Konfigurationen vereinfacht.

Übersichtliche Darstellung aller wichtigen Parametereinstellungen im Parametereditor

Alle für die schnelle Inbetriebnahme wichtigen Parametereinstellungen eines Motorabzweigs können in SIMOCODE ES V14 in einer neuen Übersicht dargestellt und editiert werden, was erheblich zur Verbesserung der Übersichtlichkeit beiträgt.

Einheitliche Darstellung von aufgezeichneten Kurvenverläufen für Messwerte

Die Anzeige von mit den SIMOCODE ES-Funktionen "Live-Trend" und "Analogwertaufzeichnung" online und offline aufgezeichneten Messwerte und Daten erfolgt mithilfe der TIA Portal-Funktion "Traces". Die Bedienung dieser Funktion verbessert sich dank der TIA Portal-weiten Vereinheitlichung maßgeblich.

4.7 Engineeringoptionen

TIA Portal Multiuser

TIA Portal Multiuser Engineering ermöglicht das Arbeiten mit mehreren Benutzern gemeinsam und gleichzeitig an einem Projekt. Dadurch werden die Projektierungszeiten wesentlich reduziert und Projekte können schneller in Betrieb genommen werden.

Das Grundprinzip:

- Für die Projektverwaltung übernimmt eine eigenständige Server Applikation. Die Server Applikation kann unabhängig von einem TIA Portal installiert werden
  oder
  die Projektverwaltung liegt auf einem Server. Der Server kann unabhängig von einem TIA Portal installiert werden.
- Verschiedene Anwender arbeiten in lokalen Sessions, basierend auf den vom Server verwalteten Projekten.
- Das Arbeiten in lokalen Sessions ist unabhängig voneinander, Änderungen von anderen Bearbeitern werden angezeigt und können übernommen werden.
- Die Änderungen aus den lokalen Sessions werden durch das Einchecken ins Serverprojekt übertragen.

TIA Portal Teamcenter Gateway

Das TIA Portal Teamcenter Gateway ermöglicht das Speichern und Verwalten von TIA Portal-Projekten und globalen Bibliotheken in Teamcenter. Die Bedienung erfolgt integriert im TIA Portal.

TIA Portal Cloud Connector

Der TIA Portal Cloud Connector ermöglicht es auf lokale PC-Schnittstellen und daran angeschlossene SIMATIC Hardware im TIA Portal Engineering zuzugreifen, obwohl das Engineering selbst per Remote Desktop in einer Private Cloud betrieben wird.

SIMATIC Energy Suite

SIMATIC Energy Suite als integrierte Option für das TIA Portal verknüpft erstmals Energiemanagement effizient mit der Automatisierung und bringt somit Energietransparenz in die Produktion. Zusätzlich wird durch die vereinfachte Projektierung von energiemessenden Komponenten der Projektierungsaufwand deutlich reduziert.

- Integriertes und intuitives Konfigurieren des Energiemanagements
- Automatische Erzeugung des PLC-Energieprogramms für S7-1500
- Archivierung auf WinCC Runtime Professional oder auf PLC-interner SIMATIC Memory Card
- Nahtlose Anbindung an SIMATIC Energy Manager PRO

SIMATIC S7-PLCSIM Advanced

S7-PLCSIM Advanced ist für die Simulation einer S7-1500 PLC geeignet. Im Vergleich zum Standard S7-PLCSIM bietet S7-PLCSIM Advanced folgende zusätzliche Funktionen:

- API zur Anbindung an Co-Simulationen
- Mehrere, verteilte Instanzen möglich
- Einstellbare, virtuelle Zeit
- Webserver- und OPC UA-Zugriff

SIMATIC ODK 1500S

Das SIMATIC ODK 1500S ermöglicht die Entwicklung von Funktionsbibliotheken mit den Hochsprachen C/C++ für ODK-fähige SIMATIC S7-1500 PLCs. Diese Funktionsbibliotheken können direkt aus dem Anwenderprogramm geladen und ausgeführt werden. ODK-fähige S7-1500 PLC sind:

- S7-1500 Software Controller (CPU 1507S (F))
- ET 200SP Open Controller (CPU 1515SP PC (F))
- S7-1500 Advanced Controller (CPU 1518(F)-4 PN/DP ODK)

SIMATIC Target 1500S™ for Simulink®

Das SIMATIC Target 1500S™ für Simulink® ist ein Add-On für Simulink der Firma MathWorks. Es ermöglicht die Einbindung von Simulink-Modellen direkt in den Programmzyklus einer ODK-fähigen S7-1500 PLC.

- Ermöglicht modellbasierte Entwicklung mit MATLAB® und Simulink für SIMATIC-Steuerungen
- Nahtlose Integration in Simulink als sog. System target file
- Kein C/C++ oder ODK Know-How notwendig
- Unterstützung des Simulink External Mode zum direkten Beobachten und Steuern von Modell-Parametern in der Steuerung aus Simulink heraus
- Der generierte Programmcode ist ablauffähig auf den CPUs 1507S(F) V2.0, CPU 1515SP PC (F) V2.0 und CPU 1518(F)-4 PN/DP ODK.

SIMATIC Visualization Architect

Der SIMATIC Visualization Architect (SiVArc) ermöglicht die einfache, schnelle und flexibel automatisierte Erstellung von HMI-Inhalten, basierend auf dem STEP 7-Anwenderprogramm.

- Unterstützte HMI-Geräte:
  Comfort Panel, Mobile Panel 2nd Generation, WinCC RT Professional, WinCC RT Advanced
- Unterstütze HMI-Objekte:
  Screens, Textlisten, Power-Tags, Caption, Button, E/A Field, Symbolic I/O Field, Graphic I/O Field, Switch, Round Button

- Konfigurierung über Regeleditoren
- Vorlagen werden über Bibliotheken verwaltet

4.8 Runtimeoptionen

Beschreibung

Folgende neue Optionen werden mit TIA Portal V14 ausschließlich als Runtimeoption angeboten. Die Produkte ProDiag und OPC UA S7-1500 sind bereits in die bestehenden Produkte STEP 7 und WinCC integriert und müssen ausschließlich bei Verwendung auf der realen Hardware lizenziert werden. Im Vergleich zu den Engineeringoptionen sind die Runtimeoptionen versionsunabhängig.

SIMATIC ProDiag

SIMATIC ProDiag ermöglicht eine gezielte und schnelle Maschinen- und Anlagendiagnose für SIMATIC S7-1500 und SIMATIC HMI:

- Zentrale zyklusgenaue Zeitstempelung der Störmeldungen
- Automatische Generierung der Überwachungslogik und der Meldungsaufrufe
- Automatisches Update der SIMATIC HMI bei Änderung der Meldeprojektierung für 3 Sprachen
- HMI Systeme müssen bei Änderungen den Runtime-Betrieb nicht verlassen
- Steht in den Spracheditoren KOP, FUP, SCL und AWL direkt zur Verfügung
- Überwachungen können nachträglich an F-Bausteinen und know-how-geschützten Bausteinen parametriert werden
- Projektzentrale Festlegung des Meldeaufbaus
- Meldetexte werden automatisch von im Projekt vorliegenden Informationen abgeleitet
- Einfache Visualisierung auf dem HMI durch vorbereitete Controls

SIMATIC OPC UA S7-1500

Die Option OPC UA S7-1500 ermöglicht über den in die S7-1500-CPU integrierten OPC UA Server die einfache Anbindung beliebiger Fremdgeräte an die S7-1500:

- OPC UA Server direkt in der S7-1500
- OPC UA Data Access, Read/Write, Subscription auf Wertänderungen
- Security
- XML-Export zur offline-Projektierung von OPC UA Clients

SIMATIC Energy Suite S7-1500

Mit der SIMATIC Energy Suite Engineering werden 10 Energieobjekte mitgeliefert. Zusätzliche Energieobjekte können durch die Runtime-Pakete erweitert werden.

WinCC / WebUX

WinCC / WebUX erlaubt es, Anlagenprozesse mittels mobiler Endgeräte über das Internet oder Intranet zu beobachten und bei Bedarf auch zu steuern. Mit der Software steht eine Visualisierungslösung zur Verfügung, die unabhängig vom Betriebssystem für HTML5 und SVG-fähige Browser geeignet ist – und keinen Engineering-Aufwand erfordert.

SIEMENS

Hinweise zum TIA Portal 1

STEP 7 2

WinCC 3

Liesmich

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Hinweise zum TIA Portal....5

1.1 Allgemeine Hinweise....5

1.2 Hinweise zu Bibliotheken....8

1.3 Hinweise zu Memory Cards....9

1.4 Hinweise zur Hardwarekonfiguration....10

1.5 Hinweise zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud)....11

1.6 Hinweise zur Benutzung des TIA Portal Teamcenter Gateway....12

1.7 Hinweise zum Arbeiten mit Multiuser Engineering....15

1.8 Hinweise zu Engineering-Optionen....17

2 STEP 7....19

2.1 Security-Hinweise....19

2.2 Hinweise zur Benutzung....21

2.3 Geräte und Netze bearbeiten....23
2.3.1 Allgemeine Hinweise zu Geräten und Netzen....23
2.3.2 Hinweise zu OPC UA-Funktionen....23
2.3.3 Hardware-Erkennung von IO-Devices....26
2.3.4 HW-ID des Typs "Hw_SubModule" für ein HMI Panel IO-Device....27
2.3.5 Verwendung von Modulen an der S7-1200....27
2.3.6 CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX....28
2.3.7 Laden von S7-1500 CPUs (SIPLUS)....29
2.3.8 ET 200MP (IM 155-5 DP ST) an einer WinLC RTX....30
2.3.9 ET 200SP HF ab V2.1, ET 200SP HS V4.0 übersetzen....30
2.3.10 Hinweise zu Online und Diagnose....31
2.3.11 Netzwerk-Komponenten....32
2.3.11.1 Netzwerk- und Kommunikationsmodule....32
2.3.12 Hinweise zum redundanten System S7-1500R/H....37

2.4 PLC programmieren....38
2.4.1 Allgemeine Hinweise zur PLC-Programmierung....38
2.4.2 Hinweise zu Anweisungen....42
2.4.3 Hinweise zu Software Units....42
2.4.4 Anwenderprogramm testen....43
2.4.4.1 Testen mit der Beobachtungstabelle....43
2.4.4.2 Testen mit der Forcetabelle....44
2.4.5 Kompatibilität von PLC-Programmen aus älteren Versionen....44
2.4.5.1 Kompatibilität von PLC-Programmen aus Versionen vor V15.1....44
2.4.5.2 Kompatibilität von PLC-Programmen aus Versionen vor V15....46
2.4.5.3 Kompatibilität von PLC-Programmen aus Versionen vor V14 SP1....50
2.4.5.4 Kompatibilität von PLC-Programmen aus Versionen vor V14....52
2.4.5.5 Kompatibilität von PLC-Programmen aus Versionen vor V13 SP1....57

2.5 Inter Project Engineering (IPE)....63
2.5.1 Hinweise zu IPE....63

2.6 Technologiefunktionen....64
2.6.1 Hinweise zu S7-1500 Motion Control....64
2.6.2 Hinweise zu weiteren Technologiefunktionen....66
2.6.3 Hinweise zu Technologiefunktionen....67
2.6.4 Hinweise zu Technologiefunktionen....68
2.6.5 SIMATIC STEP 7 Easy Motion Control....69
2.6.5.1 Hinweise zu SIMATIC STEP 7 Easy Motion Control....69
2.6.6 SIMATIC STEP 7 PID Professional....69
2.6.6.1 Hinweise zu SIMATIC STEP 7 PID Professional....69

3 WinCC....71

3.1 Security-Hinweise....71

3.2 Hinweise zur Benutzung....78

3.3 Migration....81

3.4 Engineering System....83
3.4.1 Bilder und Bildobjekte....83
3.4.2 Variablen und Verbindungen....88
3.4.3 Meldesystem und Meldeanzeigen....89
3.4.4 Systemfunktionen und Skripte....91
3.4.5 Protokolle....93
3.4.6 Rezepturen....94
3.4.7 Benutzerverwaltung....94
3.4.8 Kommunikation....94

3.5 Übergreifende Funktionen....100

3.6 Übersetzen und Laden....101

3.7 Runtime....104
3.7.1 Hinweise zum Betrieb in Runtime....104
3.7.2 Hinweise zum Betrieb von Panels in Runtime....106
3.7.3 Hinweise zum Betrieb von Runtime Advanced....107
3.7.4 Hinweise zum Betrieb von Runtime Professional....109

3.8 Bediengeräte....113
3.8.1 Hinweise zu Bediengeräten....113

3.9 Optionen....119
3.9.1 Sm@rtServer....119
3.9.2 Audit....119
3.9.3 DataMonitor....119
3.9.4 WebNavigator....120
3.9.5 WebUX....122
3.9.6 Redundancy....122
3.9.7 SIMATIC Information Server....123

Index....125

Hinweise zum TIA Portal 1

1.1 Allgemeine Hinweise

Die Hinweise der Liesmich sind Aussagen anderer Dokumente in der Verbindlichkeit übergeordnet.

Lesen Sie die folgenden Hinweise sorgfältig durch, da sie wichtige Informationen für die Installation und Nutzung enthalten. Lesen Sie diese Hinweise vor der Installation.

Windows 7 / Windows Server 2008 R2: Ungültige Zertifikatsvalidierung

Die neuen, zur Signierung verwendeten Zertifikate werden auf den Betriebssystemen Windows 7 und Windows Server 2008 R2 möglicherweise nicht validiert.

Wenn Sie keine Möglichkeit haben, den Zertifikatsspeicher online zu aktualisieren, wird gegebenenfalls die Signatur der Produktdateien als ungültig angezeigt. Aktualisieren Sie in diesem Fall auf den Offline-Systemen die von Microsoft bereitgestellten Root Zertifikate. Folgen Sie dafür den Anweisungen im FAQ 109748887 (https://support.industry.siemens.com/cs/ww/de/view/109748887).

Anzeige asiatischer Schriftzeichen im TIA Portal

Durch eine Verhaltensänderung in Microsoft Windows kann es vorkommen, dass Texte im TIA Portal nicht korrekt angezeigt werden, wenn Sie ein chinesisches TIA Portal auf ein anderes asiatisches Betriebssystem (z. B. Koreanisch) installieren. Um die Texte im TIA Portal korrekt anzuzeigen, wählen Sie im Windows Control Panel unter "Sprache für Unicode-inkompatible Programme" die Einstellung "Englisch" aus. Beachten Sie dabei jedoch, dass dadurch möglicherweise Anzeigeprobleme in anderen Programmen entstehen können.

Installation von neuen .Net Versionen oder .Net Service Packs

- Beenden Sie das TIA Portal, bevor Sie eine neue .Net Version oder ein neues .Net Service Pack auf Ihrem PG/PC installieren.
- Starten Sie das TIA Portal erst nach der erfolgreichen Installation der neuen .Net Version oder des neuen .Net Service Packs.

Hinweise zur Bedienung

- Wenn sich ein Projekt in der Liste der zuletzt verwendeten Projekte auf einem nicht verbundenen Netzlaufwerk befindet, kann es beim Öffnen des Menüs "Projekt" zu Verzögerungen kommen.
- Beim Einfügen einer CPU kann es zu einer längeren Wartezeit kommen, wenn gleichzeitig der Projekttext-Editor geöffnet ist. Insbesondere wenn es sich um die erste CPU in einem neu angelegten Projekt handelt, müssen Sie mit einer längeren Wartezeit rechnen. Um ein gutes Arbeitstempo zu gewährleisten, schließen Sie den Projekttext-Editor, bevor Sie eine CPU einfügen.

- Unter Windows 7 kann es bei länger dauernden Funktionen (z. B. beim Laden der CPU) zur Anzeige der Meldung "Applikation reagiert nicht" kommen. In diesem Fall warten Sie bitte ab, bis die Funktion ordnungsgemäß beendet ist.
- Wenn Sie eine Microsoft-Maus mit IntelliPoint installiert haben, kann es zu Überlagerungen der Schaltflächen der Titelleiste kommen. Deinstallieren Sie in diesem Fall die IntelliPoint-Software von Microsoft.
- Die Aktivierung der Option "Virtual Desktop" bei NVIDIA-Grafikkarten kann zu Problemen führen. Deaktivieren Sie in diesem Fall den "nView virtual desktop manager" Ihres NVIDIA-Grafiktreibers.

Arbeiten mit automatisch synchronisierten Netzlaufwerken

Ein TIA Portal-Projekt besteht aus mehreren Dateien, die zusammen in einem Verzeichnis gespeichert werden. Falls Sie ein Projekt auf einem Netzlaufwerk oder in einem Cloud-Verzeichnis (z. B. Dropbox, Syncplicity oder GoogleDrive) ablegen, das automatisch synchronisiert wird, kann dies zu einem Datenverlust führen, falls die Synchronisation nur teilweise bzw. asynchron erfolgt. Daher wird nicht empfohlen, TIA Portal-Projekte auf synchronisierten Netzlaufwerken oder in Cloud-Verzeichnissen direkt zu bearbeiten. Schließen Sie das TIA Portal-Projekt grundsätzlich vor der Sychronisation und stellen Sie sicher, dass alle Verzeichnisse und Dateien aus dem Projektverzeichnis gemeinsam und vollständig synchronisiert werden. Schalten Sie die automatische Synchronisation eventuell während des Arbeitens mit dem TIA Portal ab. Die Synchronisation selbst muss so erfolgen, dass die aktuellen (lokalen) Projektdaten die Projektdaten auf dem Netzlaufwerk bzw. im Cloud-Verzeichnis ersetzen.

Eingabe von Dezimalstellen

Bei bestimmten Windows-Spracheinstellungen kann es vorkommen, dass die Eingabe von Werten mit Komma nicht erkannt wird (Eingabe von "1,23" führt zu einem Fehler). Verwenden Sie stattdessen die internationale Schreibweise ("1.23").

Informationen zum TIA Portal im Online Support

Übersicht zu den wichtigsten technischen Informationen und Lösungen für das TIA Portal im Siemens Industry Online Support.

Internet-Link: TIA Portal im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/65601780)

Alle Informationen zu Service und Support im Siemens Industry Online Support:

Internet-Link: Service und Support im Siemens Industry Online Support (http://support.automation.siemens.com)

Hier können Sie auch den Newsletter abonnieren, der Sie ständig mit den aktuellen Informationen zu Ihren Produkten versorgt.

Projekte öffnen bei fehlenden Software-Komponenten

Wenn sich im TIA Portal ein Projekt aufgrund fehlender Software-Komponenten nicht öffnen lässt, können Sie diese im Siemens Industry Online Support als Trial-Software herunterladen und installieren. Anschließend können Sie das Projekt öffnen. Beachten Sie, dass sich die Trial-Software nur für einen begrenzten Zeitraum nutzen lässt.

Über den folgenden Link können Sie die Seite im Siemens Industry Online Support öffnen, auf der Ihnen die verschiedenen Softwareprodukte als Trial-Software zur Verfügung stehen: www.siemens.de/tia-portal-trial (www.siemens.de/tia-portal-trial)

TIA Portal starten

Beim Start des TIA Portals versucht Windows die Certificate Revocation List (CRL) von "windowsupdate.com" zu aktualisieren.

Bei fehlendem Internetzugang und mehreren DNS-Servern kann es zu Zeitüberschreitungen kommen, die den Start des TIA Portals verzögern.

1.2 Hinweise zu Bibliotheken

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Bibliothekselemente vergleichen

Wenn Kopiervorlagen und Typen den gleichen Namen haben, werden die entsprechenden Objekte im Projekt durch Anwenden der Aktion "Kopieren" überschrieben. Beachten Sie, dass dies ohne weitere Rückfrage erfolgt. Das gleiche Verhalten tritt auf, wenn zwar der Name der Kopiervorlage vom Name des Typen verschieden ist, aber ein Objekt innerhalb der Kopiervorlage den gleichen Namen hat wie der Typ.

1.3 Hinweise zu Memory Cards

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Hinweise zu SIMATIC Memory Cards

Die SIMATIC Memory Cards sind von Siemens für den Einsatz auf S7-1200- und S7-1500-Baugruppen formatiert und eingerichtet. Dieses Format darf nicht überschrieben werden, da sonst die Karte von den Baugruppen nicht mehr akzeptiert wird. Das Formatieren mit Windows-Mitteln ist deshalb nicht zulässig.

Verhalten bei offenem Force-Auftrag

Bitte beachten Sie, dass ein aktiver Force-Auftrag auch nach dem Laden eines neuen Projektes auf die SIMATIC Memory Card bestehen bleibt. Daher sollten Sie vor dem Ziehen einer SIMATIC Memory Card aus einer CPU zunächst den aktiven Force-Auftrag löschen, bevor Sie die Karte im PC mit einem neuen Projekt beschreiben. Wenn Sie eine SIMATIC Memory Card mit unbekanntem Inhalt verwenden, sollten Sie die SIMATIC Memory Card vor dem erneuten Laden formatieren.

Zugriffsschutz für Memory Cards in USB-Kartenleser

Durch die Verbesserung der Sicherheitsmechanismen beim Online-Zugriff und Engineering von S7-1500-CPUs, wurde auch die Datenspeicherung auf Memory Cards verändert. Aus diesem Grund kann diese Version von STEP 7 beim Lesen von Projektdaten von Memory Cards, auf die über einen USB-Kartenleser zugegriffen wird, die Passwörter der konfigurierten Schutzstufe nicht auswerten. Das geänderte Verhalten betrifft die Memory Cards für CPUs der Baureihen S7-1200/1500. Verwenden Sie daher physikalische Schutzmechanismen, um kritische Projektdaten auf Memory Cards dieser Geräte zu schützen.

Hinweis

Diese Einschränkung steht in keinem Zusammenhang mit dem Online-Zugriff auf Geräte oder mit dem Know-how-Schutz von Programmbausteinen.

1.4 Hinweise zur Hardwarekonfiguration

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Editieren einer Geräte-IP-Adresse

Verwenden Sie beim Editieren einer Geräte-IP-Adresse nicht den Adressbereich von 192.168.x.241 bis 192.168.x.250. Dieser Adressbereich wird ggf. vom System automatisch einem PG zugewiesen. Dies gilt analog in Abhängigkeit der Subnetz-Maske für alle Netzklassen.

Treiber für Kommunikationsprozessor CP 5512

Die Treiber für den CP 5512 sind in dieser Softwareversion nicht mehr enthalten. Soll der CP 5512 verwendet werden, dann muss vor der Installation dieser Software folgendes installiert sein:

http://support.automation.siemens.com/WW/view/78453460 (http://support.automation.siemens.com/WW/view/de/78453460)

Bitte beachten Sie, dass der CP 5512 maximal unter Windows 7 32 Bit verwendet werden kann. Ab Windows 8 oder mit 64 Bit Betriebssystemen ist der CP 5512 nicht kompatibel.

Hinweis

Seit 2009 steht als funktionaler Nachfolger die USB 2.0 Baugruppe CP 5711 zur Verfügung.

1.5 Hinweise zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud)

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Anleitung zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud)

Eine Anleitung zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud) finden Sie auf dem Installationsdatenträger im Verzeichnis "Documents\Readme \<Sprachverzeichnis>". Öffnen Sie hier das PDF-Dokument "TIAPortalCloudConnectorHowTo<Sprachkennung>.pdf".

Der TIA Portal Cloud Connector kann mit den Betriebssystemen Windows 7 (64 Bit) und Windows 10 (64 Bit) genutzt werden.

Umgebungsvariablen "TiaUserSettingsPath" und "TiaDefaultProjectPath"

Mit den Umgebungsvariablen "TiaUserSettingsPath" und "TiaDefaultProjectPath" kann der Speicherort für die Benutzer- und Projekteinstellung abweichend von der Systemvorgabe gesetzt werden. Dies kann z. B. dazu genutzt werden, um die Benutzer- und Projekteinstellungen zentral zu speichern.

Weitere Informationen zur Verwendung der Umgebungsvariablen finden Sie in der Anleitung zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud).

Hinweis zur Nutzung von selbstsignierten Zertifikaten bei Nutzung von HTTPS als Kommunikationsprotokoll

Microsoft überprüft regelmäßig, ob die Zertifikate im Windows-Zertifikatsspeicher aus einer vertrauenswürdigen Quelle stammen. Zertifikate aus nicht vertrauenswürdigen Quellen werden dabei gelöscht. Achten Sie daher bei Ihren selbstsignierten Zertifikaten darauf, dass sie mit einem sicheren Zertifikat signiert sind.

1.6 Hinweise zur Benutzung des TIA Portal Teamcenter Gateway

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Hinweis zur Installation des TIA Portal Teamcenter Gateway

Die Anleitung zur Installation des TIA Portal Teamcenter Gateway finden Sie in der Online-Hilfe zum TIA Portal Teamcenter Gateway unter "TIA Portal Teamcenter Gateway installieren und deinstallieren".

Hinweis zur Benutzung des TIA Portal Teamcenter Gateway

Die vorliegende Liesmich-Datei gibt Ihnen Hinweise zur Benutzung des TIA Portal Teamcenter Gateway.

Die Installation des TIA Portal Teamcenter Gateway ermöglicht Ihnen, die ab dem TIA Portal V14 zur Verfügung stehende Funktion des TIA Portal Teamcenter Gateway zu nutzen.

Hinweis zur Installation des TIA Portal Teamcenter Gateway - Datenmodells

Die Beschreibung zur Installation des TIA Portal Teamcenter Gateway - Datenmodells finden Sie auf dem Installationsdatenträger im Verzeichnis "\Support \Teamcenter_11\ServerSetupDocument\" auf der STEP 7/WinCC DVD2. Öffnen Sie das PDF-Dokument "Datamodel_Installation_enUS". Die Installationsanleitung des TIA Portal Teamcenter Gateway - Datenmodells steht nur in englischer Sprache zur Verfügung.

Hinweis zur Aktualisierung des bestehenden TIA Portal Teamcenter Gateway - Datenmodells

Beachten Sie Folgendes:

- Im TIA Portal Teamcenter Gateway V15 wurden keine Änderungen an dem bestehend TIA Portal Teamcenter Gateway - Datenmodell und auch keine Teamcenter spezifischen Änderungen vorgenommen.

Hinweis zu Teamcenter

Beachten Sie Folgendes:

- Um mit dem TIA Portal Teamcenter Gateway zu arbeiten, muss Teamcenter Rich Application Client (RAC) oder Teamcenter Client Communication System (TCCS) Version 11.2 oder höher auf Ihrem PG/PC installiert sein.

Hinweis zu TCCS (Teamcenter Client Communication System)

Beachten Sie Folgendes:

- Falls Sie die Installation von TCCS (Teamcenter Client Communication System) als "Standalone"-Installation ausgeführt haben, müssen Sie auch "Microsoft Visual C++ 2013 – Redistributable" installieren.

Hinweis zu UTF-8

Beachten Sie Folgendes:

- Um im TIA Portal Daten aus Teamcenter mit "Nicht-englische-Zeichensätze z. B. für asiatische Sprachen anzuzeigen, müssen im Teamcenter Server Unicode-Zeichensätze verwendet werden.

Hinweis zu Cloud Computing

Beachten Sie Folgendes:

- Falls Sie das TIA Portal in einer Cloud betreiben (z. B. virtuelle Maschine), muss dieses System in die Teamcenter-Umgebung eingebunden sein. Eine direkte Anbindung über eine Remote Verbindung wird vom TIA Portal Teamcenter Gateway nicht unterstützt.

Hinweis zu den Grundlagen zum Arbeiten mit dem TIA Portal Teamcenter Gateway

Ab der Version TIA Portal Teamcenter Gateway V15, steht Ihnen die Teamcenter-Operation "Teamcenter > Export..." für TIA Portal-Projekte und globale Bibliotheken zur Verfügung.

Mit dieser Operation erzeugen Sie eine Kopie des aktuellen Projektes, welche völlig frei von jeglichen Teamcenter Informationen ist. Verwenden Sie, falls eine Trennung von Teamcenter gewünscht ist, diese Operation anstelle von "Save as", da bei "Save as" Teamcenter Informationen an dem Projekt bestehen bleiben.

TIA Portal Teamcenter Gateway Cache Pfad festlegen

Standardmäßig, wird der Wert, der unter "Extras" > "Einstellungen" > "Allgemein" > "Teamcenter Gateway" > im Bereich "TIA Portal Teamcenter Gateway Cache Pfad" eingetragen ist, als "TIA Portal Teamcenter Gateway Cache Pfad" genommen.

Dieser Wert, verweist standardmäßig, auf den Projektverzeichnispfad.

Wenn Sie den Wert des "TIA Portal Teamcenter Gateway Cache Pfad" ändern, prüft Teamcenter Gateway die Gültigkeit der standardmäßig eingetragenen Werte und der geänderten Werte.

Hinweis

Der standardmäßig eingetragene Wert nicht leer sein darf.

Hinweis zu TIA Portal Teamcenter Gateway Operationen während einer gestarteten WinCC Professional Simulation

Führen Sie keine Teamcenter Gateway Operationen, wie z.B. "Einchecken" oder "Auschecken rückgängig machen", während einer gestarteten WinCC Professional Simulation aus.

1.7 Hinweise zum Arbeiten mit Multiuser Engineering

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften von Multiuser Engineering und zum Multiuser-Server.

Die Installation des Multiuser-Servers ermöglicht Ihnen, die mit dem entsprechenden TIA Portal zur Verfügung stehende Funktion des "Multiuser Engineering" zu nutzen.

Microsoft System-Account für Multiuser-Server

Bei der Installation des Multiuser-Servers wird ein Microsoft System-Account für den Multiuser-Server erzeugt. Für diesen Account tauscht Microsoft zyklisch das Passwort aus.

Falls das Passwort für diesen Account abgelaufen ist, muss das Passwort erneuert werden und der Multiuser-Server danach erneut gestartet werden.

Fehlermeldung beim Starten des Multiuser-Servers

Wenn beim oder nach dem Installieren des Multiuser-Servers eine Fehlermeldung angezeigt wird, dass der Dienst für den Multiuser-Server nicht gestartet werden kann, wird folgende Vorgehensweise empfohlen:

1. Öffnen Sie die Windows-Ereignisanzeige und überprüfen Sie, ob ein Fehlereintrag mit der Fehlernummer "1069" vorhanden ist.
2. Ist dies der Fall, finden Sie nachfolgend Hinweise für die weitere Vorgehensweise.
   Internet-Link: TIA Portal im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/65601780)
3. Alternativ können Sie auch direkt den Customer Support kontaktieren.
   Internet-Link zum Customer Support: FAQ mit ID 109739926 (https://support.industry.siemens.com/cs/ww/de/view/109739926)

Arbeiten in der lokalen Session mit aktiven Beobachtungs- und Forceaufträgen

Vor einem Einchecken oder Aktualisieren in der lokalen Session müssen aktive Beobachtungs- und Forceaufträge beendet werden. Das Einchecken oder Aktualisieren in der lokalen Session ist mit aktiven Beobachtungs- oder Forceaufträgen nicht möglich.

Arbeiten in der lokalen Session mit der Option "Lokale Session erhalten"

Wird nach dem Einchecken die Option "Lokale Session erhalten" gewählt, so erscheinen alle eingecheckten Objekte in der lokalen Session als veraltet, auch wenn diese inhaltlich identisch sind.

Arbeiten mit verschiedenen TIA Portal-Versionen

Lokale Sessions, die mit einer Version <x.y> des TIA Portals erstellt wurden, können auch nur mit dieser Version des TIA Portals bearbeitet werden.

Beispiel: Eine mit dem TIA Portal V15.1 zu einem V15.1-Projekt erstellte lokale Session kann nicht mit einer früheren TIA Portal-Version bearbeitet werden.

Arbeiten mit Multiuser Commissioning

- Beim Arbeiten mit Multiser Commissioning wird nach jedem Laden in die CPU eine neue Revision des Serverprojekts erstellt. Es wird jedoch keine neue Revision erstellt, wenn in der lokalen Session Objekte bearbeitet wurden, die **nicht** markierbar sind. Die Änderungen an nicht markierbaren Objekten gehen beim Einchecken verloren und werden nicht in die CPU geladen.
- Wenn Sie im Commissioning Modus das Laden in Gerät aus der Projektnavigation heraus über den Menübefehl „Laden in Gerät > Hardwarekonfiguration“ starten, wird immer der Dialog angezeigt, dass unterschiedliche Daten auf der CPU und im Serverprojekt vorhanden sind.
- Beim Arbeiten im Commissioning-Modus werden nur S7-1500 CPUs voll unterstützt. Bei Verwenden einer S7300/400-CPU wird im Commissioning-Modus beim Laden in die CPU immer der Dialog angezeigt, dass unterschiedliche Daten auf der CPU und im Serverprojekt vorhanden sind. Der Grund dafür ist, dass S7-300/400-CPUs keine Synchronisation unterstützen. Um die Anzeige dieser Meldung zu vermeiden, sollten Sie im Administration Tool die Option "Vor·dem·Laden·auf·unterschiedliche·Daten·überprüfen·(empfohlen)" deaktivieren.

Siehe auch

TIA Portal im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/65601780)

FAQ mit ID 109739926 (https://support.industry.siemens.com/cs/ww/de/view/109739926)

1.8 Hinweise zu Engineering-Optionen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Was ist neu bei SINAMICS in AutomationML

Entgegen der Aussage unter "Was ist neu im TIA Portal V15" im Abschnitt "SIMATIC TIA Portal Openness" können V90 Slaves und SINAMICS G120 Profinet Kopfmodule nicht über AutomationML exportiert und importiert werden.

STEP 7 2

2.1 Security-Hinweise

Security-Hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und entsprechende Schutzmaßnahmen (z. B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Industrial Security finden Sie unter:

https://www.siemens.com/industrialsecurity (https://www.siemens.com/global/de/home/unternehmen/themenfelder/zukunft-der-industrie/industrial-security.html)

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter:

https://www.siemens.com/industrialsecurity (https://www.siemens.com/global/de/home/unternehmen/themenfelder/zukunft-der-industrie/industrial-security.html)

Netzwerkeinstellungen

Die folgenden Tabellen zeigen die Netzwerkeinstellungen des jeweiligen Produkts, die Sie zur Analyse der Netzwerksicherheit und zur Konfiguration von externen Firewalls benötigen:

STEP 7 Professional					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
ALM	4410*	TCP	In-/Outbound	Lizenz-Service	Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet.

STEP 7 Professional					
RFC 1006	102	TCP	Outbound	S7-Kommunikation	Kommunikation zum S7-Controller über Ethernet / PROFINET für Programmier- und Diagnosezwecke.
DCP	---	Ethernet	Outbound	PROFINET	Das DCP-Protokoll (Discovery and basic Configuration Protocol) wird von PROFINET verwendet und stellt die Basisfunktionalität zum Auffinden und Konfigurieren von PROFINET-Geräten zur Verfügung.
SNMP	161	UDP	Outbound	PROFINET	Die SNMP Client-Funktionalität wird von STEP 7 verwendet, um Statusinformationen von PROFINET-Geräten auszulesen.
* Standard-Port, der durch Anwenderkonfiguration geändert werden kann					

WinCC ES Basic (ohne Simulation)					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
ALM	4410*	TCP	In-/Outbound	Lizenz-Service	Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet.
HMI Load	1033	TCP	Outbound	HMI Load (RT Basic)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.
* Standard-Port, der durch Anwenderkonfiguration geändert werden kann					

Simulation RT Basic					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
HMI Load	1033	TCP	Inbound	HMI Load (RT Basic)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.
EtherNet/IP	44818	TCP	Outbound	Ethernet/IP-Kanal	Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.
	2222	UDP	Inbound	Ethernet/IP-Kanal	Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.
Modbus TCP	502	TCP	Outbound	Modbus TCP-Kanal	Das Modbus TCP-Protokoll wird für Verbindungen zu Schneider PLCs verwendet.
RFC 1006	102	TCP	Outbound	S7-Kanal	Kommunikation zum S7-Controller über Ethernet / PROFINET
Mitsubishi MC	5002	TCP	Outbound	Mitsubishi MC-Kanal	Das Mitsubishi-Protokoll wird für Verbindungen zu Mitsubishi PLCs verwendet.

2.2 Hinweise zur Benutzung

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Online-Betrieb

Der parallele Online-Betrieb von STEP 7 V5.6 oder älter und STEP 7 Professional V15.1 ist nicht freigegeben.

Gleichzeitige Online-Verbindungen auf eine S7-1200-CPU (S7-1200)

Es ist nicht möglich, gleichzeitig von mehreren Instanzen des TIA Portals aus eine Online-Verbindung auf dieselbe S7-1200-CPU herzustellen.

Parallele Nutzung

Sollte die Software im Namen oder der Licence-Key nach der Installation den Zusatz "Combo" tragen, ist Ihnen gemäß Ziffer 1.6 der Allgemeinen Bedingungen (siehe auch Setup-Text) die Nutzung folgender Produkte/Versionen erlaubt:

- STEP 7 Professional 2017
- STEP 7 Professional 2010
- STEP 7 Professional 2006

PLC-Migration

Während der PLC-Migration darf kein Anweisungsprofil aktiviert sein.

Deaktivieren Sie Ihr Anweisungsprofil, bevor Sie die PLC-Migration durchführen. Anschließend können Sie das Profil wieder aktivieren.

Ziehen/Stecken der Memory Card

Führen Sie nach dem Ziehen oder Stecken einer Memory Card immer ein Urlöschen auf der CPU durch, um die CPU wieder in einen funktionsfähigen Zustand zu versetzen.

Ethernet-Baugruppen ziehen und stecken

Wenn Ethernet-Baugruppen während des Betriebs gezogen und wieder gesteckt werden, dann müssen Sie den PC booten, da die Funktionalität "Erreichbare Teilnehmer" in STEP 7 oder NCM PC ansonsten nicht alle Teilnehmer anzeigt. Während der PC gebootet wird, müssen Ethernet-Baugruppen aktiviert sein.

Projektdaten mit TIA Portal V12 und V13 laden (S7-1200)

Wenn Sie die Projektdaten einer S7-1200-CPU mit dem TIA Portal V13 geladen haben, können Sie mit dem TIA Portal V12 auf diese Daten nicht mehr zugreifen. Setzen Sie dazu die CPU zuerst auf die Werkseinstellungen zurück. Beachten Sie dabei die Informationen in der Online-Hilfe unter "Eine CPU auf Werkseinstellungen zurücksetzen".

Kompatibilität (S7-1200)

Gerätekonfiguration und Programm einer S7-1200-CPU müssen grundsätzlich mit der gleichen STEP 7-Version konfiguriert werden. Im Normalfall stellt das TIA Portal durch entsprechende Hinweise beim Laden ins Gerät sicher, dass es nicht zu Versionskonflikten kommt.

Bei S7-1200-CPUs mit der Firmware-Version V1.x ist diese automatische Überprüfung nicht möglich und daher vom Anwender sicherzustellen.

2.3 Geräte und Netze bearbeiten

2.3.1 Allgemeine Hinweise zu Geräten und Netzen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

S7-PCT IO-Link (S7-1200)

Das S7-Port Configuration Tool kann unter folgendem Link kostenlos heruntergeladen werden:

http://support.automation.siemens.com/WW/view/37936752 (http://support.automation.siemens.com/WW/view/de/37936752)

Port Configuration Tool PCT

Falls Sie das PCT-Tool einsetzen, stellen Sie sicher, dass Sie stets die aktuelle Version verwenden, die unter folgendem Link kostenlos bereitgestellt wird:

http://support.automation.siemens.com/WW/view/32469496 (http://support.automation.siemens.com/WW/view/de/32469496)

Das PCT-Tool steht zur Installation auf der DVD im Ordner "Support" zur Verfügung.

2.3.2 Hinweise zu OPC UA-Funktionen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verwendete Schema-Dateien für OPC UA-Funktionalität

Die OPC Foundation stellt ihren Nutzern Schemata zur Verfügung, die Hersteller für ihre jeweiligen Server-/Client-Implementierungen nutzen können. Diese Dateien werden z. B. genutzt, um importierte Informationsmodelle / Nodesets zu validieren.

Für die TIA Portal Version V15.1 bzw. S7-1500 CPU Firmware Version V2.6 werden folgende Schema-Dateien genutzt:

Schema	Erläuterung
https://opcfoundation.org/UA/schemas/1.04/UANodeSet.xsd	XML Schema für NodeSet-Dateien
https://opcfoundation.org/UA/schemas/1.04/Opc.Ua.NodeSet2.xml	OPC UA ("CORE") Model von 2018-06-20
https://opcfoundation.org/UA/schemas/DI/1.1/Opc.Ua.Di.NodeSet2.xml	OPC UA for Devices ("DI") Model von 2018-06-06

PLC-Objekt nicht im Server-Adressraum sichtbar

Bei sehr großen Projekten kann es in seltenen Fällen dazu kommen, dass trotz erfolgreicher Übersetzung des Projekts und erfolgreichem Laden in die CPU der CPU-Knoten bzw. das PLC-Objekt nicht im Adressraum des OPC UA-Servers sichtbar ist. In diesem Fall reicht der für OPC UA reservierte Speicher in der CPU nicht aus.

Diagnose z. B. durch UaExpert liefert den ServerStatus "Failed".

Abhilfe

Deaktivieren Sie die Standard-SIMATIC-Server-Schnittstelle in den CPU-Eigenschaften, falls sie nicht benötigt wird.

Reduzieren Sie die Anzahl bzw. Größe der Elemente, die zum Verbrauch von OPC UA-Ressourcen beitragen. Reduzieren Sie die Dateigröße auf weniger als 250 MB und die Anzahl der Knoten auf weniger als 30000.

Beispiele:

- Reduzieren Sie die Anzahl von Variablen im Adressraum des OPC UA-Servers, insbesondere von Variablen mit Datentyp Struktur.
  Um Variablen aus dem Adressraum herauszunehmen, deaktivieren Sie in der PLC-Variablentabelle oder in Datenbausteinen die Option "Erreichbar aus HMI/OPC UA".
  Bei Datenbausteinen haben Sie auch die Möglichkeit, den DB mit allen Variablen komplett aus dem Adressraum herauszunehmen. Deaktivieren Sie dazu die Option "DB erreichbar aus OPC UA" in den Eigenschaften des DBs.
- Reduzieren Sie die Länge von Namen der Elemente bei umfangreichen Datenstrukturen.
- Kürzen Sie die Descriptions der Knoten.
- Reduzieren Sie die Anzahl implementierter Methoden.

Ressourcen-Engpass beim Verbindungsaufbau bei schnellem Auf- und Abbau von Verbindungen

Der Fehler 0x80810000 (OpcUa_BadTcpNotEnoughResources) kann beim Verbindungsaufbau (OPC_UA_Connect) der Client-CPU auftreten, obwohl temporär noch freie Verbindungsressourcen angezeigt werden.

Mögliche Ursache

Während des Abbaus einer OPC UA-Verbindung (OPC_UA_Disconnect) bleibt die für diese Verbindung genutzte Verbindungsressource noch für ca. 60 s belegt.

Dieses Verhalten wird auch als "Graceful Shutdown" bezeichnet und ist konform zu TCP RFC 793.

Für die Zeitspanne des "Graceful Shutdown" gilt Folgendes:

- Das TIA Portal (Bereich "Diagnose > Verbindungsinformation") zählt diese noch belegten Ressourcen bereits als freie Ressourcen, so dass Sie möglicherweise keinen Ressourcenengpass diagnostizieren, obwohl ein Ressourcenengpass vorliegt.
- Die Anweisung zum Abbau einer OPC UA-Verbindung (OPC_UA_Disconnect) zeigt mit DONE=1 an, dass der Verbindungsabbau beendet wurde.
  Allerdings bleibt ab dem Übergang DONE=0 => DONE=1 die Verbindungsressource in der CPU noch für ca. 60 s belegt.

Abhilfe

Das beschriebene Verhalten wirkt sich beim Auf- und Abbau von Verbindungen in schneller Folge aus: Der ordnungsgemäße Abbau einer Verbindung bindet noch für 60 s die dafür genutzte Verbindungsressource.

Wenn ein Client in dieser Phase erneut einen Verbindungsaufbau initiiert, wird eine zusätzliche Ressource verbraucht, da die Ressource der abgebauten Verbindung noch belegt ist. Falls bei Ihrer Anwendung Verbindungsaufbau und Verbindungsabbau in schneller Folge vorkommen und die oben genannte Meldung auftritt, ist daher nach Möglichkeit die Frequenz dieser Folge zu verringern.

Beschreibung der OPC UA-Anweisugen OPC_UA_ServerMethodPre und OPC_UA_ServerMethodPost

Deklaration der Variablen in der Beschreibung zur Anweisung "OPC_UA_ServerMethodPre": Vorbereitung des Server-Methodenaufrufs

Entgegen der beschriebenen Variablendeklaration (Tabelle) in der Hilfe zu dieser Server-Anweisung dürfen Sie keine InOut-Variable "UAMethod_OutParameters" deklarieren. Der Aufruf der Methode wird sonst mit einer Fehlermeldung abgewiesen.

Deklaration der Variablen in der Beschreibung zur Anweisung "OPC_UA_ServerMethodPost": Nachbereitung des Server-Methodenaufrufs

Entgegen der beschriebenen Variablendeklaration (Tabelle) in der Hilfe zu dieser Server-Anweisung dürfen Sie keine InOut-Variable "UAMethod_OutParameters" deklarieren. Der Aufruf der Methode wird sonst mit einer Fehlermeldung abgewiesen.

Beispielprogramm für OPC_UA_TranslatePathList

Im Beispielprogramm für OPC_UA_TranlatePathList ist der Aufruf der Anweisung "OPC_UA_NamespaceGetIndexList" (#OPC_UA_NamespaceGetIndexList_Instance (...)) fehlerhaft, da nicht alle erforderlichen Variant-Parameter versorgt sind.

Für alle Client-Anweisungen gilt: Alle erforderlichen Variant-Parameter müssen versorgt werden.

Im Fall der verwendeten Anweisung "OPC_UA_NamespaceGetIndexList" sind das die Parameter "NamespaceUris" und "NamespaceIndexes".

Neustart des OPC UA-Servers

Der OPC UA-Server wird bei jedem Laden in die CPU gestoppt (z. B. nach dem Laden einer Konfiguration oder eines Bausteins) und startet anschließend neu.

Beachten Sie, dass OPC UA-Clients in diesem Fall die laufenden Dienste für diesen Server nicht weiter nutzen können und deshalb die Ressourcen hierfür wieder freigeben müssen.

Anschließend sind diese Dienste erneut anzufordern:

- Verbindungsaufbau erneut einleiten
- Subscriptions erneut erzeugen
- Lese/Schreibaufträge neu anlegen

2.3.3 Hardware-Erkennung von IO-Devices

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Unvollständige Hardware-Erkennung bei ET 200SP-Stationen

Mit STEP 7 V15 können Sie über die Online-Funktion "Hardware-Erkennung" den Stationsaufbau von IO-Devices ermitteln und in Ihr Projekt übernehmen inklusive aller gesteckten Module.

Wenn Sie eine ET 200SP über die Hardware-Erkennung in Ihr Projekt übernehmen, dann müssen Sie den ermittelten Stationsaufbau manuell vervollständigen:

- Überprüfen und korrigieren Sie für jeden Steckplatz die eingestellte Potenzialgruppen-Eigenschaften (helle oder dunkle BaseUnit). Welche BaseUnit tatsächlich auf welchem Steckplatz vorhanden ist, kann STEP 7 durch die Hardware-Erkennung nicht ermitteln.
- Überprüfen Sie beim Interfacemodul, ob der passende Busadapter in der Hardware-Konfiguration ermittelt wurde (z. B. BA 2xRJ45 statt real vorhandenem BA 2xFC).
- Bei einem Mischaufbau ET 200SP/ET 200AL mit Busadapter BA-Send werden die ET 200AL-Peripheriemodule nicht erkannt. Vervollständigen Sie den Stationsaufbau manuell, d. h. ergänzen Sie die fehlenden Module aus dem Hardware-Katalog.

Module/Submodule bei ET 200S (GSDML) werden nicht erkannt

Die nachfolgend genannten IO-Devices der ET 200S (GSDML) werden zwar bei der Hardware-Erkennung berücksichtigt, allerdings werden deren gesteckte Module / Submodule nicht erkannt. Wenn solche IO-Devices dem Projekt hinzugefügt werden, müssen eventuell verwendete Module / Submodule nachträglich aus dem HW-Katalog geholt und parametriert werden.

Betroffen sind:

- 6ES7 151-3AA00-0AB0 (mit Firmware V1.0)
- 6ES7 151-3AA10-0AB0 (mit Firmware V2.0)
- 6ES7 151-3BA00-0AB0 (mit Firmware V2.0)
- 6ES7 151-3AA20-0AB0 (mit Firmware V3.0)
- 6ES7 151-3BA20-0AB0 (mit Firmware V3.0)

2.3.4 HW-ID des Typs "Hw_SubModule" für ein HMI Panel IO-Device

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Ermittlung der HW-ID mit dem Typ "Hw_SubModule" für ein HMI Panel IO-Device

Der Parameter "ID" des PROFIenergy-Programmbausteins "PE_START_END" muss mit der HW-Kennung (HW-ID) mit dem Typ "Hw_SubModule" des IO-Devices versorgt werden. Die HW-ID mit dem Typ "Hw_SubModule" des HMI Panel IO-Devices wird in einer S7-1500/ S7-1200/ET200SP CPU nicht in den Systemkonstanten angezeigt, kann aber ermittelt werden. Hierzu wird in der Liste der PLC-Variablen im Register "Systemkonstanten" das Symbol für das IO-Device im HMI Panel mit dem Typ "Hw_Device" ermittelt und um eins verringert, z.B. "HMI_1_IE_CP_1~PROFINET_Interface_1~IODevice -1".

2.3.5 Verwendung von Modulen an der S7-1200

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verwendung von Modulen an der S7-1200

Die folgend aufgeführten Module werden an der S7-1200 nicht unterstützt:

Familie	Baugruppe	Bestellnummer
S7-300 FMs	SM 338	6ES7 338-4BC01-0AB0
	FM 350-1	6ES7 350-1AH03-0AE0
	FM 350-2	6ES7 350-2AH00-0AE0, 6ES7 350-2AH01-0AE0
	FM 351	6ES7 351-1AH01-0AE0, 6ES7 351-1AH02-0AE0
	FM 352	6ES7 352-1AH02-0AE0
	FM 355 S	6ES7 355-1VH10-0AE0
	FM 355 C	6ES7 355-0VH10-0AE0
	FM 355-2 C	6ES7 355-2CH00-0AE0
	FM 355-2 S	6ES7 355-2SH00-0AE0
S7-300 PtP-CP	CP 340	6ES7 340-1AH02-0AE0, 6ES7 340-1BH02-0AE0, 6ES7 340-1CH02-0AE0
	CP 341	6ES7 341-1AH01-0AE0, 6ES7 341-1AH02-0AE0, 6ES7 341-1BH01-0AE0, 6ES7 341-1BH02-0AE0, 6ES7 341-1CH01-0AE0, 6ES7 341-1CH02-0AE0
Netzkomponente	Diagnose-Repeater	6ES7 972-0AB01-0XA0
ET 200M	SIWAREX	7MH4900-2AA01, 7MH4900-3AA01, 7MH4950-1AA01, 7MH4950-2AA01

Modulkommentare der S7-1200 in PG/PC laden

Bei zentralen Konfigurationen mit S7-1200 werden Kommentare der Module, Submodule und Signalboards nicht geladen. Bei CPs/CMs werden nur die Kommentare der IE-Schnittstelle bzw. DP-Schnittstelle geladen. Bei dezentralen Konfigurationen mit ET 200SP oder ET 200MP wird von den E/A-Modulen nur der Kommentar der Kanäle geladen.

2.3.6 CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX

Das Modul AS-Interface CP 343-2 (Artikel-Nr.: 6GK7 343-2AH01) kann zwar in einen Erweiterungsbaugruppenträger des SIMATIC S7 Embedded Controller EC31-RTX (Artikel-Nr.: 6ES7 677-1DDxx-0BB0) gesteckt werden, aber der CP 343-2 darf nicht mit dem EC31-RTX betrieben werden.

2.3.7 Laden von S7-1500 CPUs (SIPLUS)

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verhalten einer S7-1500 CPU SIPLUS (6AG1...) FW <V2.5 mit formatierter Memory Card

Beteiligte Komponenten:

- CPU 151x SIPLUS (6AG1...) mit Firmware-Version kleiner V2.5
- SIMATIC Memory Card

Wenn Sie auf eine formatierte SIMATIC Memory Card eine SIPLUS-CPU laden und dann die CPU in RUN versetzen, läuft die CPU ohne Fehler. Nachdem aber die Spannung aus- und wieder eingeschaltet wird oder ein Memory Reset (MRES) durchgeführt wird, geht die SIPLUS-CPU mit einer Fehlermeldung in den STOP-Zustand. In der Fehlermeldung wird darauf hingewiesen, dass die Memory Card nicht erkannt wird, da sie fehlt, ein falscher Typ ist, falsche Daten enthält oder geschützt ist.

Das Verhalten tritt nicht auf, bei einer CPU 151x SIPLUS (6AG1...) mit einer Firmware-Version V2.5 oder größer oder wenn auf der Memory Card bereits eine Konfiguration mit einer S7-1500 SIMATIC-CPU (6ES7...) vorhanden ist und durch eine Konfiguration mit typgleicher SIPLUS-CPU (6AG1...) überschrieben wird.

Abhilfe:

1. Auf der SIMATIC Memory Card muss eine Konfiguration mit SIMATIC-CPU geladen sein. Es reicht das Vorhandensein einer CPU entsprechenden Typs, das Vorhandensein der Peripherie ist hier nicht erforderlich.
2. Die SIPLUS-CPU wird auf die Karte geladen. Die vorhandene SIMATIC-Konfiguration auf der Memory Card wird mit der SIPLUS-Konfiguration überschrieben.

Mit dieser Vorgehensweise wird eine Fehlermeldung des TIA-Portals vermieden.

Hinweis

Dieses Verfahren funktioniert NUR beim Online-Laden auf die CPU! Wenn die Karte über einen externen Card Reader programmiert wird, bleibt der Fehler bestehen!

Wenn Sie beispielsweise eine CPU 1511-1 PN SIPLUS (6AG1 511-1AK00...) laden möchten, gehen Sie folgendermaßen vor.

1. Stecken Sie eine ausreichend große, leere Memory Card in die SIPLUS-CPU.
2. Konfigurieren Sie eine typgleiche SIMATIC-CPU - in diesem Fall eine S7-1511-1 PN (6ES7 511-1AK00...). Peripheriemodule sind für den ersten Ladevorgang nicht erforderlich.
3. Laden Sie die Konfiguration in die SIPLUS-CPU.

4. Ersetzen Sie die SIMATIC-CPU in STEP 7 durch die SIPLUS-CPU (6AG1 511-1AK00...) und vervollständigen Sie die Konfiguration mit den geplanten Peripheriemodulen und dezentralen Peripheriegeräten.
5. Laden Sie die geänderte Konfiguration in die SIPLUS-CPU.

2.3.8 ET 200MP (IM 155-5 DP ST) an einer WinLC RTX

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

ET 200MP (IM 155-5 DP ST) aktuell nicht über PROFIBUS an einer WinLC RTX betreibbar

Beteiligte Komponenten:

- PC-Systeme mit Logic Controller Funktionalität WinLC RTX als DP-Master (alle Varianten: EC31-RTX (F), WinAC RTX (F) Software PLC) - im folgenden kurz "WinLC RTX" genannt
- ET 200MP (IM 155-5 DP ST) als DP-Slave

Wenn Sie Standard-Module oder F-Module in einer ET 200MP konfigurieren und als DP-Master eine WinLC RTX einsetzen, dann werden diese Module in der ET 200MP nicht vom DP-Master parametriert. Nach dem Laden der Konfiguration und dem Hochlauf der beteiligten Komponenten signalisieren die Module in der ET 200MP die fehlende Parametrierung durch blinkende grüne LEDs.

STEP 7 zeigt die fehlende Betriebsbereitschaft der Module online nicht an (Status "OK").

Abhilfe:

- Empfehlung: Verwenden Sie ein Interfacemodul IM 155-5 PN, um die ET 200MP über PROFINET IO an eine WinLC RTX anzubinden
  oder
- Wenn keine F-Module in der ET 200MP stecken: Verwenden Sie zur Projektierung der Standard-Module die GSD-Variante der ET 200MP an einer WinLC RTX

2.3.9 ET 200SP HF ab V2.1, ET 200SP HS V4.0 übersetzen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Übersetzungsfehler bei ET 200SP HF ab V2.1, ET 200SP HS V4.0

Betroffene Komponenten:

- ET 200SP HF ab V2.1
- ET 200SP HS V4.0

Beim taktsynchronen Betrieb der ET200SP (ET 200SP HF ab V2.1, ET 200SP HS V4.0) mit der Einstellung "Von OB" kann es trotz gültiger Einstellungen zu einem Übersetzungsfehler kommen. Die Fehlermeldung lautet typischerweise: "Der spezifische Ti Wert ist ungültig" oder "Der spezifische To Wert ist ungültig". Es sind aber auch andere Fehlermeldungen möglich.

Abhilfe:

Rüsten Sie in diesem Fall die Modulbeschreibung des Kopfmoduls ET200SP hoch. In der Netzsicht oder in der Gerätesicht können Sie im Inspektorfenster über die Funktion "Modulbeschreibung aktualisieren" die Modulbeschreibung des Kopfmoduls ET 200SP hochrüsten.Mit der aktuellen Modulbeschreibung kann der Fehler nach der ersten Übersetzung weiterhin auftreten. Wenn Sie gültige Einstellungen gewählt haben, tritt der Fehler mit der nachfolgenden Übersetzung nicht mehr auf.

2.3.10 Hinweise zu Online und Diagnose

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Hardware-Erkennung und anschließendes Online verbinden

Wenn Sie bei einer nicht spezifizierten CPU den Befehl "Online > Hardware-Erkennung" ausführen, wird die Online-Konfiguration dabei nicht aus der CPU geladen. Falls Sie die über die Hardware-Erkennung erhaltene Konfiguration nicht auf die CPU laden, hat dies zur Folge, dass in der Geräte- und Netzsicht immer ein Unterschied zwischen Offline- und Online-Konfiguration angezeigt wird. In der Online- und Diagnosesicht ist ersichtlich, dass eine unterschiedliche Projektierung vorliegt, obwohl die MLFBs bei der tatsächlich vorhandenen und der Offline-CPU identisch sind.

Messungen im Gerät (Speicherkarte) löschen (S7-1200)

Löschen Sie keine Messungen im Gerät, solange noch ein Trace im Gerät neue Messungen auf der Speicherkarte erzeugt.

2.3.11 Netzwerk-Komponenten

2.3.11.1 Netzwerk- und Kommunikationsmodule

Online-Verbindungen für Spezialdiagnose und Firmware-Lader

Spezialdiagnose: Online-Verbindung herstellen

Einige Netzkomponenten (S7-300/400-CPs, PC-CPs, Netzübergänge) stellen erweiterte Diagnosedaten in der Spezialdiagnose zur Verfügung.

Voraussetzung:

Eine PG/PC-Schnittstelle ist physikalisch mit dem Ziel-Gerät verbunden, z. B. über ein Ethernet-Kabel.

Vorgehen:

Gehen Sie zum Start der Spezialdiagnose folgendermaßen vor:

1. Öffnen Sie die Online- und Diagnosesicht der Baugruppe.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Spezialdiagnose".
3. Klicken Sie im Bereich "Spezialdiagnose" auf die Schaltfläche "Starte Spezialdiagnose".

Ab STEP 7 Professional V14 kann von der Engineering-Station für einige CPs nur eine einzige Online-Verbindung über den CP aufgebaut werden. Diese Verbindung kann entweder mit STEP 7 oder der Spezialdiagnose aufgebaut werden.

Wenn die Spezialdiagnose über die oben genannten Schritte nicht gestartet und eine Fehlermeldung ausgegeben wird, dann gehen Sie folgendermaßen vor:

1. Trennen Sie die bestehende Online-Verbindung zur Station.
2. Öffnen Sie die Online- und Diagnosesicht der Baugruppe erneut wie oben beschrieben und starten Sie die Spezialdiagnose.

32-Bit-Applikationen: NCM S7-Diagnose, Firmware-Lader

Wenn Sie von einer 32-Bit-Applikation aus eine Verbindung mit einem PC-CP aufgebaut haben und eine weitere Online-Verbindung über dieselbe Schnittstelle des PC-CP aufbauen möchten, dann müssen Sie vorher die Verbindung der 32-Bit-Applikation beenden.

Spezialdiagnose und Firmware-Lader: Chinesische GUI

Wenn Sie die Spezialdiagnose oder den Firmware-Lader unter chinesischer GUI öffnen, dann müssen Sie die Sprache für nicht-Unicode-Applikationen im Betriebssystem auf Chinesisch stellen.

S7-CM/CP

Kein S7-Routing mit CP 300/400 < V3 als Router

Beim S7-Routing zwischen S7-1200- oder S7-1500-Stationen wird keine Verbindung aufgebaut, wenn als Router ältere CPs der S7-300/400 zwischengeschaltet sind. Betroffen sind folgende Kommunikationsarten, die über ISO-Transport- oder ISO-on-TCP-Verbindungen laufen:

- S7-Kommunikation
- PG-Kommunikation
- HMI-Kommunikation

Das Verhalten gilt für S7-300/400-CPs mit einer Firmware-Version < V3.0, bei denen die TSAP-Länge auf 2 Byte begrenzt ist.

S7-300/400-CPs ab V3.0 dagegen unterstützen wie die S7-1200/1500 lange TSAPs und damit auch das S7-Routing.

CM 1542-1: Remanente Speicherung der IP-Adresse bei Doppeladressierung entfernen

Die IP-Adresse und der Gerätename des CM 1542-1 bleiben remanent gespeichert:

Wenn das CM z. B. bei Anlauf in einem anderen Netzwerk eine Doppeladressierung erkennt, wird das CM nicht in das Netzwerk eingebunden. Das CM geht in RUN und ist über die Ethernet-Schnittstelle nicht erreichbar.

Um das CM in das Netzwerk einzubinden, entfernen Sie die remanent gespeicherte IP-Adresse auf folgendem Weg:

1. Entfernen Sie die Speicherkarte der CPU.
2. Setzen Sie über DCP mit dem Primary Setup Tool (PST) bei einem STOP der CPU die IP-Adresse des CM ohne Projektierung auf 0.0.0.0.
   Sie haben die remanent gespeicherte IP-Adresse des CM entfernt. Der CM kann in das Netzwerk eingebunden werden.
3. Setzen Sie die Speicherkarte der CPU wieder ein.

PC-CPs und Applikationen

PC-CPs für VMware vSphere Hypervisior ESXi

Nachfolgende Kommunikationsprozessoren sind unter VMware vSphere Hypervisior ESXi für STEP 7 Professional V14 freigegeben:

- CP 5711
- VMware E1000-Netzwerk-Adapter

Nachfolgende Kommunikationsprozessoren sind nicht unter VMware vSphere Hypervisior ESXi freigegeben:

- CP 5611
- CP 5612
- CP 5621
- CP 5622
- CP 5512

Weitere Kommunikationsprozessoren wie CP 1623 und CP 1628 sind zusätzlich über die Installation der SIMATIC NET PC-Software freigegeben. Informationen darüber finden Sie in der Liesmich-Datei der SIMATIC NET PC Software.

Laden in PC-Stationen

Wenn Sie Projektdaten in eine PC-Station mit PC-CP laden, dann stellen Sie sicher, dass der CP erreichbar ist.

Aktivieren Sie im Dialog "Erweitertes Laden" nicht das Optionskästchen für die Suche der kompatiblen Teilnehmer.

Wenn ein anderer CP geladen werden soll als derjenige, der zuvor im Projekt selektiert wurde, dann ändern Sie die IP-Adresse des CP im Dialog "Erweitertes Laden" in der Liste der konfigurierten Zugriffsknoten manuell.

PROFIBUS-S7-Verbindungen mit OPC-Servern

Die Projektierung einer S7-Verbindung über PROFIBUS ist nicht möglich zwischen einem OPC-Server ab V12 und einer S7-1200 ab V4 bzw. einer S7-1500.

Abhilfe: Verwenden Sie einen OPC-Server / PC-Applikation < V12 oder verwenden Sie als S7-Station eine S7-1200 < V4 oder eine S7-300 bzw. S7-400.

Sichtbarkeit von PLC-Variablen für die OPC UA-Kommunikation

Wenn Sie bei der S7-Kommunikation zwischen einer PC-Station und einer S7-Station PLC-Variablen mit deaktivierter Option "Erreichbar aus HMI/OPC UA" projektieren, dann hat dies keine Auswirkung auf die Sichtbarkeit im OPC UA-Server der PC-Station.

Zugriffsrechte eines OPC-Client für den Zugriff auf PLC-Variablen

Wenn Sie PLC-Variablen von S7-Stationen, die über S7-Kommunikation im OPC UA-Server einer PC-Station bereitgestellt werden, von einem OPC UA-Client aus schreiben möchten, dann stellen Sie bei der Projektierung der Variablen folgendes sicher:

- Bei den PLC-Variablen ist die Option "Schreibbar aus HMI/OPC UA" aktiviert.
- In der Symbol-Konfiguration der OPC UA-Server-Applikation der PC-Station ist das Zugriffsrecht "Read/Write" aktiviert.

Security-Module

Migration von Projekten mit Ethernet-CPs und aktivierten Security-Funktionen

In STEP 7 V5.5-Projekten, welche Industrial Ethernet-CPs mit aktivierten Security-Funktionen enthalten, werden die Security-Einstellungen bei der Migration nach STEP 7 Professional deaktiviert.

Gehen Sie bei Bedarf nach der Migration wie folgt vor:

1. Aktivieren Sie die Security-Funktionalität.
2. Projektieren Sie die erforderlichen Security-Einstellungen.

Migration der IP-Zugriffsschutzliste bei Aktivierung der Security-Funktionen

Eine aktive IP-Zugriffsschutzliste wird beim Aktivieren der Security-Funktionen in Firewall-Regeln umgesetzt. Diese sind im erweiterten Firewall-Modus sichtbar und können dort angepasst werden. Der erweiterte Firewall-Modus wird automatisch aktiviert.

Security-Online-Diagnose von S7-CPs

Die Security-Online-Diagnose eines Security-fähigen CP ist nur dann möglich, wenn die Online-Verbindung direkt über den CP aufgebaut wird. Wenn die Online-Verbindung von STEP 7 über die CPU zur Station aufgebaut wurde, können Sie mittels der Schaltfläche "Online verbinden", die Sie auf der Security-Diagnoseseite "Security" > "Zustand" am Security-CP finden, eine direkte Verbindung zum CP aufbauen, um die Security-Online-Diagnose durchzuführen. Alternativ können Sie auch die Online-Verbindung zur CPU trennen und in der Online-Diagnose unter dem Eintrag "Online-Zugänge" die IP-Adresse des CP in das Eingabefeld "Geräteadresse" eintragen.

IKE-Modus

Bei der Schlüsselaushandlung in Phase 1 ist der IKE-Modus "Main" zu bevorzugen. Dieser Modus bietet gegenüber dem Modus "Aggressive" ein in der Regel sichereres Verfahren. Ein Grund für den Einsatz der Einstellung "Aggressive" ist die Verwendung von VPN-Gruppen mit unterschiedlichen Pre-shared keys.

Der IKE-Modus "Aggressive" sollte nicht in Verbindung mit Zertifikaten verwendet werden. Verwenden Sie im IKE-Modus "Aggressive" ausschließlich Pre-shared keys.

Ein Security-Modul darf nicht gleichzeitig in VPN-Gruppen enthalten sein, die unterschiedliche IKE-Modi verwenden.

Projektierungsdaten in S7-300/400 über VPN-Tunnel laden

Beim Laden von Projektierungsdaten über die Gigabit-Schnittstelle eines CP x43-1 Advanced auf eine S7-300-/S7-400-Station wird der Weg, über den geladen wird, im Projekt gespeichert. Wenn das Projekt im Anschluss über einen VPN-Tunnel, der zwischen einem SCALANCE S-Modul und dem CP x43-1 Advanced aufgebaut ist, geladen wird, dann schlägt der Ladevorgang aufgrund des geänderten Weges fehl.

Um den Ladevorgang über den VPN-Tunnel durchzuführen, gehen Sie wie folgt vor:

1. Verbinden Sie die Engineering-Station über die Schaltfläche "Online verbinden" mit der Gigabit-Schnittstelle des CP x43-1 Advanced.
2. Trennen Sie die Online-Verbindung zum CP x43-1 Advanced.
3. Laden Sie das Projekt über die Gigabit-Schnittstelle des CP x43-1 Advanced auf die Station.

VPN-Tunnelaufbau durch VPN-fähige CP 1200/1500

Der Aufbau von VPN-Tunnelverbindungen durch VPN-fähige CP 1200/1500 ist mit der Authentifizierungsmethode "Pre-shared key" nur dann möglich, wenn es sich bei dem VPN-Verbindungspartner ebenfalls um einen VPN-fähigen CP 1200/1500 handelt. Der Aufbau von VPN-Tunnelverbindungen durch einen VPN-fähigen CP 1200/1500 zu allen anderen VPN-Verbindungspartnern ist nur mit der Authentifizierungsmethode "Zertifikat" möglich.

Security-Konfigurationen aus Altprojekten

Bevor eine Security-Konfiguration aus einer älteren STEP 7 Version in STEP 7 V14 verwendet werden kann, muss sich ein Benutzer mit Administratorrechten in STEP 7 V13 SP1 an dieser Security-Konfiguration anmelden und das Projekt speichern. Nach dem Öffnen des Projekts in STEP 7 V14 muss sich erneut ein Benutzer mit Administratorrechten an der Security-Konfiguration anmelden. Danach ist das Verwenden dieser Security-Konfiguraiton in STEP 7 V14 möglich.

Kopieren von Security-Modulen zwischen STEP 7 Instanzen

Das Kopieren von Security-Modulen zwischen verschiedenen Instanzen von STEP 7 wird nicht unterstützt.

Gerätezertifikate für CP 1543-1 V2.0 und CP 1543SP-1

Für CP 1543-1 V2.0 und CP 1543SP-1 werden nur diejenigen Gerätezertifikate auf die Baugruppe geladen, die der Baugruppe über den lokalen Zertifikatsmanager als Gerätezertifikate zugeordnet wurden. Diese Zuordnung erfolgt in den lokalen Security-Einstellungen der Baugruppe im Eintrag „Zertifikatsmanager“ über den Tabelleneditor „Gerätezertifikate“. Bei der Zertifikatszuordnung stehen die Zertifikate des globalen Zertifikatsmanagers zur Verfügung.

Keine Fallback-VPN-Profile für CP 1200/1500

Fallback-VPN-Profile werden von CP 1200/1500 nicht unterstützt. Projektierte Fallback-VPN-Profile werden deshalb nicht auf CP 1200/1500 geladen.

Medienkonverter

Wenn Sie ein Projekt migrieren, müssen Sie Medienkonverter neu projektieren.

2.3.12 Hinweise zum redundanten System S7-1500R/H

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Redundante Systeme S7-1500R/H

GRAPH-Bausteine, ProDiag-Bausteine und die Anweisung "Program_Alarm" werden in V15.1 von S7-1500R/H-CPUs nicht unterstützt. Falls Ihr Programm eines dieser Elemente enthält, werden Sie durch eine Übersetzungsmeldung darauf hingewiesen.

Weitere Details finden Sie unter "Funktionsweise des redundanten Sysems S7-1500R/H" und "Programmierung von S7-1500R/H CPUs" im Informationssystem sowie im Kapitel "Grundlagen zur Programmbearbeitung" des Systemhandbuchs S7-1500R/H.

2.4 PLC programmieren

2.4.1 Allgemeine Hinweise zur PLC-Programmierung

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Übersetzen von Bausteinen

Wird die Meldung "Das Netzwerk wird in dieser Version nicht unterstützt." ausgegeben, verteilen Sie Mehrfachzuweisungen, die nicht am ersten boolschen Eingang der Anweisung angeschlossen sind, mithilfe einer TEMP-Variablen vom Datentyp BOOL auf mehrere Netzwerke.

Instanz-spezifische Attribute beim Importieren von Bausteinen über TIA Openness

Die Import-Regeln können in bestimmten Situationen den Verlust instanz-spezifischer Attribute, wie z. B. Startwerte, bedeuten.

Anzeige des CPU-Namens bei Systemmeldungen

Ab V15.1 wird der Name der CPU (oder bei H- oder R-CPUs der Name der Station) in der Spalte "Zusatztext 1" der Systemmeldungen angezeigt. Hierzu müssen Sie die Hardware des betreffenden Geräts übersetzen. Wenn der CPU-Name nicht über die Option "Übersetzen > Hardware (nur Änderungen)" angezeigt wird, verwenden Sie die Option "Übersetzen > Hardware (komplett übersetzen)".

Stellen Sie sicher, dass das Laden in das Gerät tatsächlich durchgeführt wird; die Information, dass die Hardware aktuell ist, ist nicht ausreichend!

Applikationszyklus eines Taktsynchronalarm-OBs ändern

Wenn Sie den Parameter "Applikationszyklus (ms)" eines Taktsynchronalarm-OBs ändern, ändert sich implizit auch die Hardware-Konfiguration. Damit der geänderte Wert für die Applikationszyklusdauer nach dem nächsten Ladevorgang auf der CPU wirksam wird, müssen Sie die Hardware-Konfiguration explizit mitladen. Wählen Sie dazu entweder den Menübefehl "Laden in Gerät > Hardware und Software" oder den Befehl "Laden in Gerät > Hardwarekonfiguration".

Hinweise für das Setzen von Haltepunkten im Standard-Anwenderprogramm bei S7-1500 F-CPUs

Wenn ein Haltepunkt aktiviert ist und erreicht wird, geht die CPU nach dem "HALT" unmittelbar in STOP. Sollten Sie nach dem "HALT" wieder in RUN wechseln wollen, um Ihr Standard-Anwenderprogramm weiter zu testen, können Sie dies mit S7-PLCSIM simulieren.

Weitere Hinweise zum Testen mit Haltepunkten bei F-CPUs finden Sie im Handbuch "SIMATIC Safety - Projektieren und Programmieren".

Verwendung des Systemdatentyps "TDiag_StatusExt" (S7-1200/1500)

Alternativ zum Systemdatentyp "TDiag_Status" kann auch der erweiterte Systemdatentyp "TDiag_StatusExt" verwendet werden, um erweiterte Informationen zur Ursache eines Verbindungsabbruches zu erhalten. Speziell das Auslesen des Strukturelements "LastConnErrTimeStamp" führt zum Wechsel der CPU in den Systemzustand "Defect" und darf aktuell nicht verwendet werden.

Gesamtgröße der Begleitwerte für ProDiag (S7-1500)

Das Meldeverfahren der S7-1500-CPU stellt maximal 512 Byte für Begleitwerte zur Verfügung. ProDiag benötigt für interne Informationen einen Teil dieser Gesamtgröße der Begleitwerte. Damit stehen für ProDiag für die drei möglichen Begleitwerte im Spezifischen Textfeld maximal 400 Byte zur Verfügung.

Diese Größe wurde im TIA Portal V14 beim Übersetzen nicht überprüft. Ab TIA Portal V14 SP1 erfolgt eine entsprechende Überprüfung, die einen Fehler beim Überschreiten dieser Größe ausgibt.

Dies ist bei der Migration von Projekten von der Version 14 zur Version 14 SP1 entsprechend zu berücksichtigen.

Hochrüstung von know-how-geschützten Bausteinen (S7-1200/1500)

Know-how-geschützte Bausteine, die mit einer Version <V15 eingerichtet wurden, können nach dem Upgrade von V13 SP1 auf >=V15 ohne Eingabe von Passwort in die Steuerung geladen werden und sind ablauffähig. Zum Bearbeiten eines know-how-geschützten Bausteins in >=V15 entfernen Sie den Know-how-Schutz und setzen ihn erneut. Anschließend kann der Baustein auch im know-how-geschützten Zustand umnummeriert und umbenannt werden.

- Für know-how-geschützte Bausteine, die mit V13 SP1 eingerichtet wurden:
  - Zum Bearbeiten des know-how-geschützten Bausteins in >=V15 entfernen Sie den Know-how-Schutz und setzen ihn erneut. Anschließend kann der Baustein auch im know-how-geschützten Zustand umnummeriert und umbenannt werden.
- Für know-how-geschützte Bausteine, die VOR V13 SP1 eingerichtet wurden:
  - Zum Bearbeiten entfernen Sie den Know-how-Schutz in der V13 SP1 und setzen ihn erneut.

Hochrüsten von GRAPH-Bausteinen (S7-300, S7-400, S7-1500)

Prüfen Sie nach dem Hochrüsten eines Projekts aus eine Version kleiner V15 auf V15.1, ob im Protokoll reparierte GRAPH-Bausteine aufgelistet sind und testen Sie die Ablauffähigkeit dieser Bausteine in Ihrem Programm.

Wird ein Projekt, das GRAPH-Bausteine enthält, von V15.0 nach V15.1 hochgerüstet, müssen die Programmbausteine zuerst übersetzt werden, bevor sie beobachtet werden können.

Informationen zur Netzwerksicherheit

Die Kommunikationszugriffe zwischen TIA Portal und CPU bzw. zwischen HMI (außer, wenn HMI-Zugriffe über die "GET/PUT-Kommunikation" verwendet werden) und CPU bieten integrierte Sicherheitsfunktionen. Diese beinhalten einen größeren Schutz vor Manipulationen und einen höheren Zugriffsschutz. Zum Schutz vor unbefugtem Netzwerkzugriff auf eine CPU mit standardisierten Kommunikationszugriffen, wie z. B. "GET/PUT", "TSEND/TRCV", "Modbus", "FETCH/WRITE", sollten Sie zusätzlich geeignete Schutzmaßnahmen ergreifen (z. B. Zellenschutzkonzept).

Funktionen aus der globalen Bibliothek “Long Functions“ mit PLCSIM (S7-1500)

Durch den Übersetzungsvorgang werden Funktionen der CPU-Familie S7-1500 an die CPU gebunden, um sie gegen Manipulationen zu schützen. In der CPU verwendete Funktionen aus der globalen Bibliothek „Long Functions“, die eine solche Bindung besitzen, können nicht in eine Simulation geladen werden. Es besteht jedoch die Möglichkeit solche Elemente gegen ihre Originale aus der globalen Bibliothek auszutauschen. Diese haben keine Bindung an eine CPU.

Inkonsistente Programme in eine S7-1500-CPU laden (S7-1500)

Im TIA Portal ist es nicht möglich, inkonsistente Programme ohne Konsistenzprüfung in eine S7-1500-CPU zu laden. Während des Ladens werden implizit alle Bausteine des Programms geprüft und bei Inkonsistenzen erneut übersetzt. Wenn sich auf Ihrer CPU jedoch Programme befinden, die mit älteren Versionen von STEP 7 geladen wurden, können diese Programme Inkonsistenzen aufweisen.

Beachten Sie in diesem Fall folgenden Hinweis:

Wenn Sie ein inkonsistentes Programm von einem Gerät laden, können Sie das Programm anschließend nicht unverändert zurück in eine S7-1500-CPU laden, da während des Ladens zwangsläufig eine Konsistenzprüfung durchgeführt und bestehende Inkonsistenzen behoben werden.

Geänderte Zugriffsberechtigungen für die Anzeige des Vergleichsstatus (S7-1200/1500)

Gültig für:

- **CPU 1500 < V2.0 (sowie ET200 SP und Software Controller)**
- **CPU 1200 V4.0 und V4.1**

Ab dem TIA Portal V14 haben sich für die genannten CPUs aus der Familie S7-1200/1500 die erforderlichen Zugriffsrechte für das Anzeigen des Vergleichsstatus von Bausteinen geändert.

Bisher benötigten Sie bei den genannten CPUs lediglich die Zugriffsstufe "HMI-Zugriff" für die Anzeige des Vergleichsstatus, ab der V14 benötigen Sie für die Anzeige des Vergleichsstatus die Zugriffsstufe "Lesezugriff".

Diese Änderung bewirkt, dass Sie beim Online-Gehen nach dem Passwort für "Lesezugriff" gefragt werden.

Falls Sie nur das Passwort für "HMI-Zugriff" zur Verfügung haben, können Sie trotzdem eine Online-Verbindung herstellen.

Klicken Sie in diesem Fall bei der Passwortabfrage für den Lesezugriff auf "Abbrechen" und tragen Sie im nachfolgenden Dialog das Passwort für den "HMI-Zugriff" ein. Die Online-Verbindung wird dann mit den vorhandenen "HMI-Zugriffsrechten" erstellt, allerdings wird kein Vergleichsstatus zu den Bausteinen angezeigt.

Anstelle der Symbole für den Vergleichsstatus werden nur Fragezeichen angezeigt.

Gültig für:

- **CPU 1200 <=V3.x.**

Ab dem TIA Portal V14 erfolgt für die oben genannten CPUs neuerdings beim Online-Gehen eine Passwort-Abfrage für das Anzeigen des Vergleichsstatus von Bausteinen, sofern diese CPUs mit einem Schreibschutz und/oder Leseschutz projektiert sind.

Falls Sie das Passwort für den Zugriff auf lesegeschützte Bausteine auf einer geschützten CPU nicht kennen, können Sie trotzdem eine Online-Verbindung herstellen.

Verlassen Sie in diesem Fall den angezeigten Dialog für die Passwortabfrage mit "Abbrechen".

Die Online-Verbindung wird trotzdem erstellt, allerdings wird kein Vergleichsstatus zu den Bausteinen angezeigt.

Anstelle der Symbole für den Vergleichsstatus werden nur Fragezeichen angezeigt.

Datenaustausch zwischen Standard- und F-Programm beim Einsatz von Team Engineering

Um die Datenkonsistenz zwischen Standard- und Sicherheitsprogramm zu gewährleisten, sollten Sie:

- Daten nicht über Merker, sondern über Globale Datenbausteine austauschen.
- Zugriffe zwischen Sicherheitsprogrammen und Standard-Anwenderprogrammen auf zwei Standard-Datenbausteine begrenzen.

Zusätzliche Informationen dazu finden Sie im Kapitel 5.13 des Programmierleitfadens unter: https://support.industry.siemens.com (https://support.industry.siemens.com/cs/document/81318674/programmierleitfaden-und-programmierstyleguide-f%C3%BCr-s7-1200-und-s7-1500?dti=0&lc=de-WW)

Offline/Online-Vergleich

In der aktuellen Version wurden Bereinigungen der internen Projektdaten vorgenommen, um die Datenintegrität zu erhöhen. Nach der Installation können dadurch vereinzelt Bausteine beim Öffnen automatisch korrigiert werden und der Offline/Online-Vergleich zeigt folglich unterschiedliche Prüfsummen an.

Online-/Offline-Unterschiede in der Projektnavigation (S7-1200 FW V2.0 und V2.1)

Wenn Sie mithilfe der Anweisung "WRIT_DBL" einen Datenbaustein ändern, wird der dadurch entstandene Unterschied zwischen Online- und Offline-Baustein zunächst nicht korrekt durch die Symbole in der Projektnavigation angezeigt. Der Unterschied wird erst dargestellt, wenn Sie die Online-Verbindung trennen und anschließend erneut online gehen.

2.4.2 Hinweise zu Anweisungen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Anweisung "TSEND_C: Verbindung aufbauen und Daten senden" für S7-1200- und S-1500-CPUs

Entgegen der Aussage in der Online Hilfe hat der Wert 7002 des Parameters „STATUS“ die folgende Bedeutung:

- STATUS=7002: Verbindung wird gerade aufgebaut (REQ irrelevant)

Anweisung "TRCV_C: Verbindung aufbauen und Daten empfangen" für S7-1200- und S-1500-CPUs

Entgegen der Aussage in der Online Hilfe hat der Wert 7002 des Parameters „STATUS“ die folgende Bedeutung:

- STATUS=7002: Verbindung wird gerade aufgebaut (REQ irrelevant)

Anweisung "GetSMCinfo: Informationen über die SIMATIC Memory Card auslesen"

Zusätzlich zu den in der Online Hilfe beschriebenen Werten kann de Parameter "Status" den folgenden Wert annehmen:

- 8091: Der gewählte Wert des Parameters "Mode" ist nur bei S7-1500-R/H-CPUs zulässig.

Anweisung "RH_CTRL: Abläufe bei R/H-Systemen beeinflussen"

Entgegen der Aussage in der Online Hilfe haben die Werte 3 und 4 des Parameters "Mode" die folgende Bedeutung:

- Mode=3: Durchführung des SYNCUP sperren
- Mode=4: Durchführung des SYNCUP wieder freigeben

2.4.3 Hinweise zu Software Units

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Software Units für die gemeinsame Inbetriebnahme nutzen (S7-1500)

Beachten Sie bei der gemeinsamen Inbetriebnahme, dass nur Bausteine synchronisiert werden können. Relationen werden beim Synchronisationsvorgang nicht berücksichtigt.

Manueller Vergleich von Software Units (S7-1500)

Beim manuellen Vergleich von Software Units zeigt das Übersichtssymbol des Eigenschaftsvergleichs nicht immer das korrekte Ergebnis an. Die Vergleichsergebnisse der einzelnen Quell- und Zieldaten sind jedoch korrekt.

2.4.4 Anwenderprogramm testen

2.4.4.1 Testen mit der Beobachtungstabelle

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Mehrfachzugriffe auf dieselbe CPU (S7-300/400/1200)

Es darf von einem PG/PC aus nur mit einem geöffneten TIA Portal online auf eine CPU (S7-300/400/1200) zugegriffen werden. Mehrfachzugriffe auf dieselbe CPU aus dieser Familie sind nicht erlaubt und können zu Fehlern führen.

Laden von Datenbausteinen bei laufendem Steuerauftrag

Hinweis

Das Laden von geänderten Datenbausteinen bei laufendem Steuerauftrag kann zu unvorhergesehenen Betriebszuständen führen. Der Steuerauftrag steuert weiterhin die angegebene Adresse, während sich die Adressbelegung im Datenbaustein möglicherweise geändert hat. Beenden Sie laufende Steueraufträge vor dem Laden von Datenbausteinen.

Funktion "Peripherieausgänge freischalten"

Die Funktion "Peripherieausgänge freischalten" steht im TIA Portal für CPUs aus der Baureihe S7-1500 nicht zur Verfügung.

Diese Funktion kann im TIA Portal ausschließlich mit einer CPU S7-300, S7-400 oder S7-1200 ausgeführt werden.

Testen von Programmen, die aus älteren STEP 7-Versionen konvertiert wurden

Um ein aus einer älteren STEP 7-Version konvertiertes Programm zu beobachten und zu testen, müssen Sie das Programm zunächst einmal mit der aktuellen STEP 7-Version übersetzen und laden.

Geänderte Anzeigeformate in der Beobachtungstabelle

Die Anzahl der Anzeigeformate wurde reduziert, so dass verwendete, aber nicht mehr unterstütze Anzeigeformate bei einem Projektupgrade auf ihre Voreinstellungen zurückgesetzt werden.

Beispiel: Zeiger können nur noch im Format "Zeiger" oder im Format "Hexadezimal" angezeigt werden.

2.4.4.2 Testen mit der Forcetabelle

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Forcen von Variablen bei direktem Peripheriezugriff (S7-300)

Wenn Sie in Ihrem Anwenderprogramm einen direkten Peripheriezugriff für eine S7-300 CPU verwenden, so ist ein Forcen dieser Peripherieadresse nicht erlaubt.

Beispiel

Erfolgt im Anwenderprogramm ein Peripheriezugriff auf die Adresse "EB0:P", so dürfen folgende Peripherieadressbereiche nicht geforct werden: E0.0:P, EB0:P, EW0:P und ED0:P.

Geänderte Anzeigeformate in der Forcetabelle

Die Anzahl der Anzeigeformate wurde reduziert, so dass verwendete, aber nicht mehr unterstütze Anzeigeformate bei einem Projektupgrade auf ihre Voreinstellungen zurückgesetzt werden.

Beispiel: Zeiger können nur noch im Format "Zeiger" oder im Format "Hexadezimal" angezeigt werden.

2.4.5 Kompatibilität von PLC-Programmen aus älteren Versionen

2.4.5.1 Kompatibilität von PLC-Programmen aus Versionen vor V15.1

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Kompatibilität

Grundsätzlich können Sie alle PLC-Programme, die mit älteren Versionen des TIA Portals erstellt wurden, in V15.1 weiter verwenden. Da jedoch in V15.1 Verbesserungen und Fehlerbehebungen am Compiler durchgeführt wurden, kann es in wenigen Fällen vorkommen, dass sich das Programm nach dem Hochrüsten anders verhält oder Sie den Programmcode manuell anpassen müssen. Diese Fälle sind nachfolgend detailliert beschrieben.

Abarbeitungsreihenfolge von Funktionsaufrufen in SCL (S7-1200/S7-1500)

In älteren Versionen des TIA Portals war die Abarbeitungsreihenfolge von Zuweisungen, die Ausdrücke oder Funktionsaufrufe beinhalteten, nicht immer eindeutig. Die Bearbeitung wurde nicht konsequent von rechts nach links durchgeführt, sondern es wurden in bestimmten Fällen Teile des Ausdrucks auf der linken Seite zuerst ausgewertet.

Dieses Verhalten wurde in V15.1 korrigiert, so dass nun immer zuerst die rechte Seite der Zuweisung berechnet und dann die linke Seite bestimmt wird.

Beispiel:

```
#myArr["FC_1"()]:= #x + #y;
```

In früheren Versionen wurde hier zunächst der „FC1“ ausgeführt, um ein ARRAY-Element zu bestimmen. Anschließend wurde #x + #y gerechnet.

Ab V15.1 wird das Programm nun korrekt von rechts nach links bearbeitet: Zunächst wird #x + #y gerechnet und anschließend der „FC1“ ausgeführt.

Diese Korrektur kann dazu führen, dass Ihr Programm sich nach dem Hochrüsten anders verhält, z. B. wenn im „FC1“ die Variablen #x oder #y gesetzt werden. Prüfen Sie deshalb Ihr Programm nach dem Hochrüsten.

Vergleicher (SCL S7-1200 >= 4.2 und S7-1500 >= 2.0)

Beim Vergleich von zwei Instanzdatenbausteinen über Vergleichsoperationen in SCL wird nicht die Struktur der Bausteine, sondern lediglich die Bausteinnummer verglichen. Ab V15.1 werden Sie durch eine Fehlermeldung beim Übersetzen darauf hingewiesen, dass diese Operation möglicherweise nicht das gewünschte Ergebnis liefert (im Regelfall wird FALSE geliefert, da die Nummern der Bausteine verschieden sind).

Beispiel:

Das Ergebnis der folgenden Vergleichsoperation ist FALSE, auch wenn die Struktur beider Bausteine identisch ist.

```
IF "MyDBofFB" = "MyDBofSameFB" THEN …
```

Import von IEC-Zeiten (S7-1200/S7-1500)

IEC-Zeiten enthalten schreibgeschützte Parameter, z. B. ET (Aktueller Zeitwert). In älteren Versionen des TIA Portals wurden beim Import von externen Quelldateien die Startwerte dieser schreibgeschützten Parameter in bestimmten Fällen überschrieben. Dieses Verhalten wurde mit V15.1 korrigiert: Startwerte für schreibgeschützte Parameter werden nun beim Import von Quelldateien ignoriert. Stattdessen werden die Defaultwerte des Datentyps verwendet. Prüfen Sie deshalb Ihr Programm nach einem Import.

Ersetzen ungültiger Unicode-Zeichen beim Hochrüsten

Beim Hochrüsten auf V15.1 werden ungültige Unicode-Zeichen in Ihrem Programm korrigiert, z. B. werden ungültige Zeichen in String-Konstanten mit den Escape-Sequenzen ersetzt, die per IEC definiert sind. Dadurch kann sich die Prüfsumme des Programms ändern und es werden Unterschiede zwischen Online- und Offline-Programm angezeigt. Übersetzen und laden Sie Ihr Programm in diesem Fall neu.

Anweisungen “SCATTER” / “SCATTER_BLK” und “GATHER“ / „GATHER_BLK“ (S7-1200/1500)

Mit den Anweisungen „SCATTER“ / „SCATTER_BLK“ können Sie eine Bitfolge in einzelne Bits zerlegen. Mit den Befehlen „GATHER“ / „GATHER_BLK“ können Sie einzelne Bits zu einer Bitfolge zusammenfassen.

Die Werte, mit denen die Anweisungen arbeiten, dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen, da in diesen Bereichen die Datenkonsistenz nicht gewährleistet ist.

Ab V15.1 wird diese Regel durch eine genauere Syntaxprüfung überprüft. Falls Sie in Ihrem Programm die o.g. Anweisungen mit unzulässigen Eingangsparametern versorgt haben, wird beim Übersetzen nun ein Fehler gemeldet.

Als Abhilfe können Sie die Variablen zunächst auf einen temporären Speicherbereich kopieren und von dort an die Anweisungen „SCATTER“ oder „GATHER“ übergeben.

Anweisungen "BLKMOV" / "UBLKMOV" und "FILL" / "UFILL"

Mit der Anweisung "BLKMOV" / "UBLKMOV" und "FILL" / "UFILL" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Wenn diese Anweisungen auf einen nicht vorhandenen Speicherbereich zugreifen, wird ein Programmierfehler ausgegeben. In vorherigen Versionen wurde in seltenen Fällen ein Fehlercode am Parameter "RET_VAL" anstelle eines Programmierfehlers erzeugt. Dieses Verhalten wurde für S7-1500-CPUs ab der Firmware-Version 2.6 korrigiert.

Anweisung “RUNTIME” (SCL/AWL S7-1200/S7-1500)

Der Parameter „Mem“ der Anweisung „RUNTIME“ ist vom Datentyp „LREAL“ und darf nur mit einer Variable dieses Datentyps verwendet werden. Eine implizite Konvertierung ist hier nicht möglich. In V15.1 wird diese Regel durch eine genauere Syntaxprüfung überprüft. Falls Sie in Ihrem Programm den Parameter „Mem“ der Anweisung „RUNTIME“ mit einem unzulässigen Datentypen versorgt haben, wird beim Übersetzen nun ein Fehler gemeldet.

2.4.5.2 Kompatibilität von PLC-Programmen aus Versionen vor V15

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Kompatibilität

Grundsätzlich können Sie alle PLC-Programme, die mit älteren Versionen des TIA Portals erstellt wurden, in V15 weiter verwenden. Da jedoch in V15 Verbesserungen und Fehlerbehebungen am Compiler durchgeführt wurden, kann es in wenigen Fällen vorkommen, dass sich das Programm nach dem Hochrüsten anders verhält oder Sie den Programmcode manuell anpassen müssen. Diese Fälle sind nachfolgend detailliert beschrieben.

S7-GRAPH: Implizite Konvertierungen "Hardware-Datentypen <> INT"

Ab TIA Portal V15 ist es, wie bereits bei KOP, FUP, AWL und SCL, nicht mehr möglich implizite Konvertierungen von Hardware-Datentypen (z. B. INT -> DB_ANY oder DB_ANY -> INT) zu programmieren.

Bitte verwenden Sie stattdessen entweder den Datentyp UINT oder eine explizite Konvertierung.

Unterkategorien in ProDiag (S7-1500)

Ab TIA Portal V14 SP1 Update 2 werden die einzelnen Unterkategorien für ProDiag nicht mehr manuell hinzugefügt, sondern in der Spalte "Aktivierung" aktiviert bzw. deaktiviert.

Das führt dazu, dass nach dem Hochrüsten eines Projekts <V14 SP1 Update 2, das bereits vorhandene Unterkategorien in mindestens einer Überwachung verwendet, zuerst die Schaltfläche "Reparieren" ausgeführt werden muss, bevor Sie neue Unterkategorien anlegen können.

Anweisungen "(U)MOVE_BLK" und "(U)FILL_BLK" (S7-1500)

Die Anweisungen "(U)MOVE_BLK" und "(U)FILL_BLK" haben im TIA Portal <= V15 beim direkten Zugriff auf die Peripherie nur auf das Prozessabbild zugegriffen.

Dieses Verhalten wurde korrigiert und führt jetzt zu einem Laufzeitfehler, da direkte Peripheriezugriffe bei BLK-Befehlen nicht zulässig sind.

Programmschleifen in SCL (S7-1200/S7-1500)

In S7-1200/S7-1500 ist es mit SCL nicht möglich, den Index einer Programmschleife aus der Schleife heraus zu ändern.

Das folgende Programmbeispiel wäre also ungültig:

```
FOR #i := 1 TO 10 DO
#i := #i + 1;
END_FOR;
```

Ab TIA Portal V15 wird beim Übersetzen diesbezüglich eine genauere Syntaxprüfung durchgeführt und ein Übersetzungsfehler ausgegeben.

Rückgabewerte von Funktionen in SCL

Bei Funktionen, deren Rückgabewert einen strukturierten Datentyp hat, wurde das Verhalten angeglichen. Strukturierte Datentypen sind z. B. PLC-Datentypen oder STRING.

Abhängig davon, ob der ENO Ausgang der Funktion genutzt wurde oder nicht, konnte es bisher vorkommen, dass eine temporäre Arbeitskopie des Funktionswerts angelegt wurde, obwohl beim sonst gleichen Aufruf ohne ENO diese Arbeitskopie nicht angelegt wurde. Die Zuweisung dieser temporären Kopie auf die verschaltete Variable ist dann bei Beendigung der Funktion erfolgt.

Dieses Verhalten wurde folgendermaßen angeglichen:

Die Verwendung des Ausgangsparameters ENO hat nun keinen Einfluss mehr darauf, ob der Rückgabewert per Kopie oder per Referenz übergeben wird. Wenn der Rückgabewert einen strukturierten Datentyp hat, wird er in der Regel per Referenz übergeben. Die Übergaberegeln sind im Detail im Kapitel "Parameter als Kopie oder als Pointer übergeben" beschrieben.

LDT_TO_DATE in SCL (S7-1500)

Das Verhalten von LDT_TO_DATE bei einem Konvertierungsfehler zur Laufzeit wurde korrigiert.

Vergleich eines ARRAY-Elements mit einer Variable vom Datentyp "VARIANT" in SCL (S7-1200/1500)

Der Vergleich eines variabel indizierten ARRAY-Elements mit einem VARIANT wurde bisher in einigen Fällen nicht korrekt durchgeführt. Statt des ARRAY-Elements wurde das gesamte ARRAY für den Vergleich herangezogen.

Dieses Verhalten wurde in V15 korrigiert: Für den Vergleich wird nun das indizierte ARRAY-Element ausgewertet. Wenn Sie in Ihrem Programm solche Vergleiche verwenden, müssen Sie den betroffenen Baustein nach dem Hochrüsten prüfen.

Beispiel:

```
IF (#my_Array[#1] = #my_variant) THEN…
```

Bisher wurde "my_variant" mit "my_Array" verglichen. Ab V15 wird der Vergleich korrekt durchgeführt und "my_variant" wird mit Element #1 von "my_Array" verglichen.

Anweisung "GetSymbolPath" (S7-1200/S7-1500)

Verschaltet ein FB eine statische Variable an einen Formalparameter eines Aufruf und ein unterlagerter Baustein verwendet "GetSymbolPath" V1.0 oder V1.1 für den Parameter, so führt die Übersetzung dieses Bausteins zur zusätzlichen Anzeige des DB-Namens vor dem Namen der statischen Variable, allerdings ohne Punkt als Trennzeichen und ohne Ersetzen des #. Die Version V1.2 von "GetSymbolPath" zeigt den Pfad korrekt mit Punkt als Trennzeichen und ohne # an.

Falls Sie die Anweisung "GetSymbolPath" in Ihrem Programm einsetzen, sollten Sie nach dem Hochrüsten auf >=V15 die Anweisung "GetSymbolPath" V1.2 verwenden.

Anweisung "GetInstancePath" (S7-1200/S7-1500)

Verschaltet ein FB eine statische Variable an einen Formalparameter eines Aufruf und ein unterlagerter Baustein verwendet "GetInstancePath", so führt die Übersetzung dieses Bausteins zur zusätzlichen Anzeige des DB-Names vor dem Namen der statischen Variable.

Additionen und Subtraktionen mit DTL/LDT/DT

In Additionen und Subtraktionen mit den Datentypen DTL/LDT/DT können keine untypisierten Konstanten mehr verwendet werden. Es wird ein Syntaxfehler gemeldet.

Folgende arithmetische Operationen sind nun beispielsweise nicht mehr möglich:

```
#myDTL + 1
#myLDT + 1
#myDate_And_Time +1
```

Verwenden Sie stattdessen Konstanten vom Datentyp "Time" oder "LTime"

```
#myDTL + TIME#1d
#myLDT + TIME#1ms
#myDate_And_Time + LTIME#1ns
```

Laufzeit-Informationen (S7-1200/S7-1500)

Die erweiterten Anweisungen "GetInstanceName", "GetInstancePath", "GetSymbolName", "GetSymbolPath" und "GetBlockName" verhalten sich nun in allen Programmiersprachen gleich: Falls die auszugebende Zeichenkette aufgrund von Größenbeschränkungen nicht in den Ausgangsparameter "OUT" passt, wird der Name abgeschnitten und das Zeichen "..." am Ende der Zeichenkette angezeigt.

ARRAYs von Multiinstanzen

ARRAYs von Multiinstanzen konnten bisher in manchen Fällen nicht korrekt adressiert werden, wenn über variablen Index eine Multiinstanz adressiert wurde, die in einer weiteren Multiinstanz instanziiert war, die ihrerseits ein Element eines ARRAYs von Multiinstanzen war.

Beispiel:

```
#MyMultiArray[#index].MyMultiMember := 10;
```

Dieses Verhalten wurde in V15 korrigiert.

Wenn Sie in Ihrem Programm solche Zugriffe verwendet haben, müssen Sie den betroffenen Baustein nach dem Hochrüsten prüfen und möglicherweise neu übersetzen.

Bausteinparameter vom Datentyp "VARIANT" (S7-1200/1500)

Wie bereits in KOP, FUP, AWL und GRAPH, ist es nun auch in SCL nicht mehr möglich, einem Formalparameter vom Datentyp "VARIANT" einen Instanz-DB als Aktualparameter zuzuweisen.

Eine Ausnahme bilden Instanz-DBs, die von einem PLC-Datentyp oder Systemdatentyp abgeleitet sind. Diese können Sie in allen Programmiersprachen einem Bausteinparameter vom Datentyp "VARIANT" zuweisen.

Es ist nicht mehr möglich, eine schreibgeschützte Variable einem Bausteinparameter vom Datentyp "VARIANT" als Aktualparameter zuzuweisen.

Entfernen Sie wenn möglich den Schreibschutz der Variable. Wenn das nicht möglich ist, prüfen Sie, wie Sie Ihr Programm ändern können, um den Zugriff auf die schreibgeschützte

Variable zu vermeiden. Wenn Sie z. B. nur lesend zugreifen möchten, können Sie den Inhalt der schreibgeschützten Variable im aufrufenden Baustein in eine andere Variable umkopieren, die keinen Schreibschutz trägt. Anschließend können Sie diese Variable als Aktualparameter übergeben.

Externe Quellen importieren

Beim Import einer externen Quelle, in der eine Anweisung als Multi- oder Einzelinstanz verwendet wird, wurde bisher das gesamte Programm auf die in der Quelle verwendete Anweisungsversion umgestellt. Ab Version V15 des TIA Portals ist dieses Verhalten korrigiert: Die bereits im Projekt enthaltenen Anweisungen werden nicht mehr umgestellt, sondern behalten die von Ihnen projektierte Anweisungsversion.

Wenn das Projekt nach dem Import verschiedene Versionen einer Anweisung enthält, wird ein Syntaxfehler gemeldet. Übersetzen Sie in diesem Fall das gesamte Programm erneut. Markieren Sie dazu in der Projektnavigation den Ordner "Programmbausteine" und wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)" oder "Übersetzen > Software (komplett übersetzen)".

Größe von ARRAY of BOOL/BYTE/CHAR (S7-1500)

Bisher hatten ARRAYs der Datenypen BOOL, BYTE oder CHAR unterschiedliche Größen, abhängig davon, ob sie innerhalb einer Struktur verwendet wurden oder nicht. Ab V15 wurde die Größe der ARRAYs vereinheitlicht. Wenn Sie in Ihrem Programm absolut, z. B. über einen ANY-Zeiger, auf ein ARRAY von BOOL, BYTE oder CHAR zugreifen, müssen Sie das Programm nach dem Hochrüsten prüfen.

2.4.5.3 Kompatibilität von PLC-Programmen aus Versionen vor V14 SP1

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Kompatibilität

Grundsätzlich können Sie alle PLC-Programme, die mit älteren Versionen des TIA Portals erstellt wurden, in V14 SP1 weiter verwenden. Da jedoch in V14 SP1 Verbesserungen und Fehlerbehebungen am Compiler durchgeführt wurden, kann es in wenigen Fällen vorkommen, dass sich das Programm nach dem Hochrüsten anders verhält oder Sie den Programmcode manuell anpassen müssen. Diese Fälle sind nachfolgend detailliert beschrieben.

Daneben haben Sie auch die Möglichkeit, das Projekt im Kompatibilitätsmodus zu bearbeiten. Weitere Informationen zur Kompatibilität finden Sie unter "Kompatibilität von Projekten".

Gesamtgröße der Begleitwerte für Programmmeldungen

Das Meldeverfahren der S7-1500-CPU stellt maximal 512 Bytes für Begleitwerte (SD-Parameter) der Anweisung "Program_Alarm" zur Verfügung.

Auf diese Größe wurde im TIA Portal <= V14 beim Übersetzen lediglich mit einer Warnung hingewiesen. Ab TIA Portal V14 SP1 erfolgt eine strengere Überprüfung. Beim Überschreiten dieser Größe wird bereits beim Übersetzen ein Fehler ausgegeben, um während der Laufzeit Fehlerszenarien auszuschließen. Diese Überprüfung bezieht sich fortan nicht mehr auf die tatsächliche Größe der SD-Parameter zum Zeitpunkt des Aufrufs der Anweisung "Program_Alarm", sondern auf die maximal mögliche Größe der SD-Parameter. Sie müssen sich also im Vorfeld überlegen, mit welcher Länge Sie Ihre zu übergebenden Zeichenfolgen-Variablen deklarieren, da diese besonders viele Bytes in Anspruch nehmen.

Dies ist bei der Migration von Projekten von der Version <= V14 zur Version V14 SP1 Update 1 oder neuer entsprechend zu berücksichtigen.

Programmschleifen in SCL (S7-1200/S7-1500)

In S7-1200/S7-1500 ist es mit SCL nicht möglich, den Index einer Programmschleife aus der Schleife heraus zu ändern.

Das folgende Programmbeispiel wäre also ungültig:

```
FOR #i := 1 TO 10 DO
#i := #i + 1;
END_FOR;
```

Ab TIA Portal V14.0.1 wird beim Übersetzen diesbezüglich eine genauere Syntaxprüfung durchgeführt und ein Übersetzungsfehler ausgegeben.

CASE-Anweisungen in SCL

Innerhalb einer Case-Anweisung sind Labels ohne zugehörige "GOTO"-Anweisung nicht mehr zulässig. Es wird ein Syntaxfehler gemeldet.

ARRAYs von Multiinstanzen

ARRAYs von Multiinstanzen konnten in V14 in manchen Fällen nicht korrekt adressiert werden, wenn einzelne Variablen innerhalb einer Multiinstanz die Remanenzeinstellung „Im IDB setzen“ hatten. Dieses Verhalten wurde in V14 SP1 korrigiert.

Wenn Sie in Ihrem Programm solche Variablen verwendet haben, müssen Sie den betroffenen Baustein nach dem Hochrüsten möglicherweise neu übersetzen.

Parametertyp “Block_DB” zur Angabe des Instanz-DBs

In KOP und FUP können Sie den Instanz-DB einer Anweisung nicht über einen Eingang des Datentyps „Block_DB“ angeben. Dieses Verhalten wurde für alle CPU-Familien und alle Bausteinarten vereinheitlicht.

Ab TIA Portal V14 SP1 wird diesbezüglich ein Syntaxfehler ausgegeben.

Wenn Sie in Ihrem Programm Instanz-DBs mithilfe des Datentyps "Block_DB" übergeben haben, müssen Sie Ihr Programm ändern. Verwenden Sie stattdessen eine Parameterinstanz, um die Instanz zur Laufzeit zu übergeben.

Peripheriezugriffe als Eingangsparameter von Bausteinen (S7-1500 ab FW V2.1)

Peripheriedirektzugriffe, die als Eingangsparameter an Bausteinen verschaltet sind, führen beim Auftreten eines Peripheriezugriffsfehlers nicht mehr dazu, dass der Baustein nicht durchlaufen wird. Stattdessen wird mit dem Ersatzwert des Signals im Baustein gearbeitet.

Für CPUs S7-1500 mit FW kleiner V2.1 und alle CPUs S7-1200 ist das Systemverhalten unverändert:

Der Baustein wird aufgrund des Peripheriezugriffsfehlers nicht aufgerufen. Die Programmausführung wird nach dem Bausteinaufruf fortgesetzt. Wenn ein OB 122 vorhanden ist, oder die lokale Fehlerbehandlung aktiviert ist, werden diese ausgeführt.

2.4.5.4 Kompatibilität von PLC-Programmen aus Versionen vor V14

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Kompatibilität

Grundsätzlich können Sie alle PLC-Programme, die mit älteren Versionen des TIA Portals erstellt wurden, in V14 weiter verwenden. Da jedoch in V14 Verbesserungen und Fehlerbehebungen am Compiler durchgeführt wurden, kann es in wenigen Fällen vorkommen, dass sich das Programm nach dem Hochrüsten anders verhält oder Sie den Programmcode manuell anpassen müssen. Diese Fälle sind nachfolgend detailliert beschrieben.

AWL: Anweisungen "SRW", "SLW" und "SSI" (S7-300, S7-400, S7-1500)

Der erlaubte Wertebereich der Schiebezahl dieser Anweisungen hat sich vom TIA Portal V13 SP1 zum TIA Portal V14 verändert.

In der Version V13 SP1 ist es möglich, auf einer CPU der Baureihen S7-1200/-1500 als Schiebezahl eine Zahl im Wertebereich von 0 bis 31 und auf einer CPU der Baureihen S7-300/-400 als Schiebezahl eine Zahl im Wertebereich von 0 bis 15 anzugeben.

Ab der Version V14 wurden die Wertebereiche für alle CPU-Baureihen (S7-300/400/1200/1500) auf den einheitlichen Wert von 0 bis 15 festgelegt.

Darstellung des BCD-Formats

Die Darstellung des BCD-Formats hat sich in Bezug auf das Vorzeichen vom TIA Portal V13 SP1 zum TIA Portal V14 verändert.

In der Version V13 SP1 werden die Werte im BCD-Format ohne Vorzeichen dargestellt.

Ab der Version V14 werden die Werte im BCD-Format mit Vorzeichen dargestellt. Dadurch kann es zu einer veränderten Darstellung der Werte im BCD-Format kommen.

Ganzzahlen (Dezimalsystem)	Hexadezimalzahlen	Darstellung in V13 SP1	Darstellung in V14
0	16#0000	BCD#0	BCD#0
-26215	16#9999	BCD#9999	BCD#-999
1365	16#0555	BCD#555	BCD#555
21845	16#5555	BCD#5555	BCD#555
4096	16#1000	BCD#1000	BCD#0
-28672	16#9000	BCD#9000	BCD#0

Anweisungen "SET_BF: Bitfeld setzen" und "RESET_BF: Bitfeld rücksetzen" (S7-1200, S7-1500)

Das Verhalten der Anweisung hat sich in Bezug auf Strukturen vom TIA Portal V13 SP1 zum TIA Portal V14 verändert.

In der Version V13 SP1 werden immer so viele Bits gesetzt bzw. zurückgesetzt, wie am Eingang angegeben sind. Das gilt auch, wenn Sie z. B. eine Struktur vom Datentyp PLC-Datentyp, STRUCT oder ARRAY angegeben haben. Wenn Sie z. B. 10 Bits setzen oder zurücksetzen wollen und die Struktur nur 5 Bits enthält, dann werden die nachfolgenden 5 Bits innerhalb der Adressreihenfolge ebenfalls gesetzt oder zurückgesetzt.

Ab der Version V14 werden bei Strukturen vom Datentyp PLC-Datentyp, STRUCT oder ARRAY maximal so viele Bits gesetzt, wie die Struktur Bits enthält: Wenn Sie z. B. den Wert "20" angeben und die Struktur nur 10 Bits enthält, dann werden auch nur diese 10 Bits gesetzt. Wenn Sie z. B. den Wert "5" angeben und die Struktur 10 Bits enthält, dann werden genau 5 Bits gesetzt.

Anweisungen "SCALE: Skalieren" und "UNSCALE: Deskalieren" (S7-1500)

Das Verhalten der Anweisung hat sich in Bezug auf die Angabe der Grenzwerte (LO_LIM > HI_LIM) vom TIA Portal V13 SP1 zum TIA Portal V14 verändert.

In der Version V13 SP1 erhielten Sie eine Fehlermeldung, wenn der untere Grenzwert größer als der obere Grenzwert war.

Ab der Version V14 ist diese Wertangabe zulässig und das Ergebnis wird umgekehrt proportional zum Eingabewert skaliert.

Auf den CPUs der Baureihen S7-300/400 hat die Anweisung schon immer das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Nicht genutzte Bits von PLC-Datentypen (UDT) bei Firmware >= V1.8.1

Die nicht genutzten Bits von PLC-Datentypen in Standard Speicherbereichen werden belegt bzw. überschrieben, z. B. bei einem PLC-Datentyp, der 4 Bit beinhaltet.

Bei Firmware Versionen < V1.8.1 konnten Sie die nicht genutzten Bits eines PLC-Datentyps anderweitig verwenden.

Ab der Firmware Version >= V1.8.1 werden alle Bits, auch wenn nur 4 Bits benutzt werden, belegt bzw. überschrieben.

Hinweis

Adresszuweisungen

Achten sie darauf, dass Sie unterschiedlichen symbolischen Adressierungen nicht zweimal die gleiche absolute Adresse zuweisen.

Explizite Datentypkonvertierung in SCL (S7-1200) bei Firmware >= V4.2

Bei Firmware Versionen < V4.2 wurde bei der expliziten Datentypkonvertierung in SCL von SINT/INT/DINT/REAL_TO_STRING/WSTRING die Zeichenkette rechtsbündig übertragen und mit führenden Leerzeichen aufgefüllt.

Beispiel: REAL_TO_WSTRING(12) = ' 1.200000E+1'

Ab dem TIA Portal V13 wird bei der expliziten Datentypkonvertierung in SCL von SINT/INT/DINT/REAL_TO_STRING/WSTRING die Zeichenkette mit einem führenden Vorzeichen dargestellt und linksbündig übertragen.

Beispiel: REAL_TO_WSTRING(12) = '+1.200000E+1'

SCL: EN-/ENO-Mechanismus bei Bausteinparametern vom Datentyp (W)STRING (S7-1200/1500)

Ab TIA Portal V14 wird geprüft, ob Bausteinparameter vom Datentyp (W)STRING bei der Parameterübergabe zur Laufzeit abgeschnitten werden. Dies kann vorkommen, wenn Formal- und Aktualparameter unterschiedliche deklarierte Längen haben. Wenn die deklarierte Länge des Zielparameters zur Laufzeit nicht ausreicht, um den (W)STRING aufzunehmen, wird der (W)STRING abgeschnitten und der Freigabeausgang ENO auf "FALSE" gesetzt.

Wenn Sie den Freigabeausgang ENO in Ihrem Programm auswerten, ändert sich möglicherweise die Semantik Ihres Programms.

Weiterreichen von STRING-Parametern zwischen optimierten Bausteinen und Standardbausteinen (S7-1200/1500)

Vor TIA Portal V14 wurden bei der Parameterübergabe zwischen optimierten und nicht-optimierten Bausteinen keine Längeninformationen für STRINGs an den aufgerufenen Baustein übergeben. Somit konnten bei der Weiterbearbeitung des STRINGs Zugriffsfehler auftreten, die nicht vom ENO-Mechanismus erfasst wurden. Der Freigabeausgang "ENO" blieb trotz Zugriffsfehler auf "TRUE".

Ab TIA Portal V14 werden STRING-Längeninformationen übergeben und Zugriffsfehler durch den ENO-Mechanismus überwacht. Wenn Sie den Freigabeausgang ENO in Ihrem Programm auswerten, ändert sich möglicherweise die Semantik Ihres Programms.

Längendeklaration bei Konstanten vom Datentyp "STRING" oder "WSTRING"

Bei lokalen und globalen (W)STRING-Konstanten sind keine Längendeklarationen mehr zulässig. Wenn Ihr Programm Längendeklarationen, z.B."MyStringConst [7]", enthält, werden diese beim Hochrüsten automatisch entfernt. Diese Änderung hat jedoch keine semantische Auswirkung auf Ihr Programm.

Änderung des Basisdatentyps für HW_ANY von WORD auf UINT

Der Hardware-Datentyp "HW_ANY" dient zur Identifikation einer beliebigen Hardware-Komponente, z. B. einer Baugruppe. In V14 wurde der Basisdatentyp von "HW_ANY" von WORD auf UINT geändert. Nach der Konvertierung nach V14 kann es deshalb zu Fehlern beim Übersetzen kommen, wenn Sie an einem Parameter vom Datentyp "HW_ANY" als Aktualwert eine explizite WORD-Konstanten (z. B. W#16#1) oder eine Variable vom Datentyp WORD angegeben haben.

Ändern Sie in diesem Fall die Notation der Konstante (z. B. auf 16#1) oder ändern Sie den Datentyp der Variablen auf UINT.

Länge von Bausteinkommentaren

Ab TIA Portal V14 sollten die Bausteinkommentare eine Länge von 32767 Unicode-Zeichen nicht überschreiten.

Syntaxprüfung der Anweisungen „SR: Flipflop setzen/rücksetzen“ und „RS: Flipflop rücksetzen/setzen“

Am Eingang „S“ dürfen Eingänge, Ausgänge, Merker sowie statische oder lokale Daten als Operanden verwendet werden. Die Verwendung von Konstanten ist nicht zulässig. Bisher wurde jedoch die Verwendung der Konstanten "0" und "1" nicht durch die Syntaxprüfung abgelehnt. Ab TIA Portal V14 wurde die Syntaxprüfung verbessert, so dass nun auch die Verwendung von "0" oder "1" am Eingang „S“ abgelehnt wird. Möglicherweise erhalten Sie beim Übersetzen eines bisher fehlerfrei übersetzbaren Programms nun eine Fehlermeldung. Ändern Sie in diesem Fall Ihr Programm und verwenden Sie einen zulässigen Operanden.

Any-Zeiger: Zugriff auf optimierte Daten

Der ANY-Zeiger darf auch bei einer CPU der Baureihe S7-1500 nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen. Der Zugriff auf optimierte Daten über einen ANY-Zeiger ist nicht zulässig.

Ab TIA Portal V14 findet eine genauere Syntaxprüfung bei ANY-Zeigern statt. Der Zugriff auf optimierte Daten wird nun abgelehnt. Möglicherweise erhalten Sie beim Übersetzen eines bisher fehlerfrei übersetzbaren Programms nun eine Fehlermeldung.

Datentyp "(L)REAL"

Auf 64-Bit-Windows-Versionen wird der Datentyp LREAL unter Umständen anders ausgewertet als in den bisherigen 32-Bit-Windows-Versionen. Ein Unterschied kann bei Rundungen und (L)Real-Werten von Konstanten im Least Significant Bit auftreten.

Die mögliche Differenz liegt in der letzten Stelle der Mantisse und hat somit in den meisten Fällen nur geringfüge Auswirkungen auf die Genauigkeit.

Falls Sie die letzte Stelle der Mantisse auswerten, kann sich die Semantik Ihres STEP 7-Programms nach der Konvertierung ändern.

Parameter "ST" bei IEC-Zeiten

Der Parameter "ST" einer IEC-Zeit wird intern genutzt und darf nicht geschrieben werden. Deshalb ist dieser Parameter ab TIA Portal V14 nicht mehr sichtbar. Wenn Sie in Ihrem Programm schreibend auf den Parameter "ST" zugreifen, wird nach dem Hochrüsten eine Fehlermeldung beim Übersetzen ausgegeben.

Wenn Sie Daten mit Inter Project Engineering austauschen, und mit HMI auf den Parameter "ST" zugegriffen haben, müssen Sie das Projekt zunächst exportieren und in der Proxy-CPU neu importieren. Erst dann verschwindet der Parameter "ST" in der HMI-Projektierung.

Begleitwerte für Meldungen (S7-1500)

Mit einer älteren Version des TIA Portals erstellte Projekte können eingebettete Begleitwerte enthalten, die in den verschiedenen Projektsprachen unterschiedlich strukturiert sind. Die Begleitwerte können beispielsweise eine andere Reihenfolge haben oder fehlen in einer anderen Sprache. Beim Hochrüsten eines solchen Projekts wird die Reihenfolge der Begleitwerte vereinheitlicht. Sie richtet sich nach der im Ursprungsprojekt eingestellten Referenzsprache. In diesem Fall ist es empfehlenswert, die übersetzten Texte prüfen und korrigieren zu lassen.

Implizite Datentypkonvertierung bei Bausteinparametern vom Datentyp "DB_ANY"

Beim Zugriff auf einen Bausteinparameter vom Datentyp "DB_ANY" mit der Syntax DB_ANY.%DB(B|W|D) wurde bisher keine implizite Datentypkonvertierung durchgeführt. Das Bitmuster des Quellwerts wurde einfach in den Zieldatentyp verschoben. Wenn der Zieldatentyp kleiner war als der Quelldatentyp, kam es zu einem Überlauf und der geschriebene Wert war möglicherweise falsch oder ungenau. Bei Genauigkeitsverlusten oder Laufzeitfehlern wurde der Freigabeausgang ENO nicht auf "FALSE" gesetzt.

Ab V14 wird bei einer Parameterübergabe mit der Syntax DB_ANY.%DB(B|W|D) nach den Regeln der impliziten Datentypkonvertierung konvertiert. Beachten Sie, dass dadurch nach dem Hochrüsten möglicherweise ein anderer Wert geschrieben wird und die Semantik Ihres Programms sich verändern kann. Außerdem wird beim Auftreten eines Laufzeitfehlers der Freigabeausgang ENO auf "FALSE" gesetzt.

Schnittstelle von Organisationsbausteinen mit Standardzugriff

Die Schnittstelle von Organisationsbausteinen mit Standardzugriff muss eine Mindestgröße von 20 Byte haben. In älteren Versionen des TIA Portals wurde während des Übersetzungslaufs nur die Schnittstelle des OB1 hinsichtlich der Mindestgröße überprüft. Ab V14 werden die Schnittstellen aller Organisationsbausteine geprüft. Möglicherweise erhalten Sie beim Übersetzen eines bisher fehlerfrei übersetzbaren Programms nun eine Fehlermeldung.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME (S7-1500)

Ab TIA Portal V14 werden beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig vom gewählten Vergleichsausdruck durchgeführt.

2.4.5.5 Kompatibilität von PLC-Programmen aus Versionen vor V13 SP1

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Kompatibilität

Grundsätzlich können Sie alle PLC-Programme, die mit TIA Portal V12 SP1 oder V13 erstellt wurden, in V13 SP1 weiter verwenden. Da jedoch in V13 SP1 Verbesserungen und Fehlerbehebungen am Compiler (Übersetzung des Programmcodes) durchgeführt wurden, kann es in wenigen Fällen vorkommen, dass sich das Programm nach dem Hochrüsten anders verhält oder Sie den Programmcode manuell anpassen müssen. Diese Fälle sind nachfolgend detailliert beschrieben.

Nicht genutzte Bits von PLC-Datentypen (UDT) bei Firmware >= V1.8.1

Die nicht genutzten Bits von PLC-Datentypen in Standard Speicherbereichen werden belegt bzw. überschrieben, z. B. bei einem PLC-Datentyp, der 4 Bit beinhaltet.

Bei Firmware Versionen < V1.8.1 konnten Sie die nicht genutzten Bits eines PLC-Datentyps anderweitig verwenden.

Ab der Firmware Version >= V1.8.1 werden alle Bits, auch wenn nur 4 Bits benutzt werden, belegt bzw. überschrieben.

Hinweis

Adresszuweisungen

Achten sie darauf, dass Sie unterschiedlichen symbolischen Adressierungen nicht zweimal die gleiche absolute Adresse zuweisen.

Explizite Datentypkonvertierung in SCL (S7-1200) bei Firmware >= V4.2

Bei Firmware Versionen < V4.2 wurde bei der expliziten Datentypkonvertierung in SCL von SINT/INT/DINT/REAL_TO_STRING/WSTRING die Zeichenkette rechtsbündig übertragen und mit führenden Leerzeichen aufgefüllt.

Beispiel: REAL_TO_WSTRING(12) = ' 1.200000E+1'

Ab dem TIA Portal V13 wird bei der expliziten Datentypkonvertierung in SCL von SINT/INT/DINT/REAL_TO_STRING/WSTRING die Zeichenkette mit einem führenden Vorzeichen dargestellt und linksbündig übertragen.

Beispiel: REAL_TO_WSTRING(12) = '+1.200000E+1'

Lesen eines ungültigen Peripherieeingangs

Im TIA Portal ab der Version V12 wird der Fehlercode 16#2942 für das Lesen eines ungültigen Peripherieeingangs nur noch dann ausgegeben, wenn Sie den fehlerhaften Zugriff so programmiert haben, dass dieser auch wirksam wird, z. B. %MW10 := "UngültigerWortZugriff":P, und nicht innerhalb einer irrelevanten Sequenz liegt. Das wäre z. B. #tmp := "UngültigerWortZugriff":P, wenn die Variable #tmp im restlichen Programmbaustein nicht verwendet wird.

Anweisung "S_CONV: Zeichenkette konvertieren"

Der EN-/ENO-Mechanismus verhält sich im TIA Portal V13 SP1 unterschiedlich zum TIA Portal V13.

In der Version V13 liefert der ENO-Freigabeausgang im Fehlerfall den Signalzustand "0", auch wenn Sie den ENO-Freigabeausgang deaktiviert haben. Wenn Sie eine weitere Anweisung an den ENO-Freigabeausgang geschaltet haben, dann wird diese nicht ausgeführt.

In der Version V13 SP1 liefert der ENO-Freigabeausgang im Fehlerfall den Signalzustand "1", wenn Sie den ENO-Freigabeausgang deaktiviert haben. Wenn Sie eine weitere Anweisung an den ENO-Freigabeausgang geschaltet haben, dann wird diese erwartungsgemäß ausgeführt.

Anweisung "SET: Bereich bitweise setzen" (S7-300/S7-400)

Im TIA Portal können Sie am Parameter N der Anweisung SET auch ein Element eines Datenbausteins verschalten.

Anweisung "MUX: Multiplexen" (SCL)

Bis einschließlich TIA Portal V13 wurde der Wert der Variablen am Eingangsparameter unverändert als Funktionswert ausgegeben, auch wenn der Parameter K eine negative Ganzzahl hatte. Dieses Verhalten hat sich im TIA Portal V13 SP1 geändert.

Wenn Sie im TIA Portal V13 SP1 an den Eingangsparametern der Anweisung MUX eine Variable mit gültigem Datentyp angeben und der Parameter K eine negative Ganzzahl hat, dann wird der Wert der Variable verändert. Die gültigen Datentypen finden Sie in der Beschreibung der Anweisung MUX im Informationssystem.

Anweisung "DEMUX: Demultiplexen" (SCL)

Bis einschließlich TIA Portal V13 wurde am Ausgangsparameter OUTELSE kein Wert ausgegeben, wenn der Wert am Parameter K < 0 war. Dagegen wurde am Ausgangsparameter OUTELSE der Wert des Eingangsparameters IN ausgegeben, wenn der Wert am Parameter K > verfügbare Ausgänge war. Dieses Verhalten hat sich im TIA Portal V13 SP1 geändert.

Wenn Sie im TIA Portal V13 SP1 am Parameter K einen Wert angeben, der außerhalb der verfügbaren Ausgänge liegt (K < 0 oder K > verfügbare Ausgänge), dann wird am Ausgangsparameter OUTELSE der Wert des Eingangsparameters IN ausgegeben.

Anweisung "MOVE: Wert kopieren" (AWL) (S7-1500)

Ab TIA Portal V13 SP1 gilt bei der Anweisung "MOVE" in AWL eine strengere Syntaxregel:

Bis einschließlich TIA Portal V13 war es möglich, die untypisierte Konstanten, wie z. B. "0" am Eingangsparameter IN anzugeben.

Wenn Sie im TIA Portal V13 SP1 z. B. den Wert "0" kopieren möchten, dann müssen Sie eine typisierte Konstante, entsprechend des Datentyps des Ausgangsparameters, angeben (z. B. INT#0, DINT#0, etc.).

Anweisung "Program_Alarm: Programmmeldung mit Begleitwerten erzeugen" (S7-1500)

Ab TIA Portal V13 SP1 darf der Datentyp INT für die Begleitwerte bzw. für eine Indizierungsvariable einer Textliste nicht mehr verwendet werden. Verwenden Sie stattdessen den Datentyp UINT.

Anweisungen "READ_DBL: Aus Datenbaustein im Ladespeicher lesen" und "WRIT_DBL: In Datenbaustein im Ladespeicher schreiben"

Das TIA Portal V13 SP1 meldet einen Übersetzungsfehler, wenn Sie "READ_DBL" V1.0 oder "WRIT_DBL" V1.0 verwenden und mit dem Parameter DSTBLK auf einen Datenbaustein mit dem Attribut "Nur im Ladespeicher ablegen" zugreifen.

Verwenden Sie stattdessen die Version 1.2 der beiden Anweisungen.

Anweisungen "READ_DBL: Aus Datenbaustein im Ladespeicher lesen" und "WRIT_DBL: In Datenbaustein im Ladespeicher schreiben" (SCL)

Ab TIA Portal V13 SP1 gilt bei den Anweisungen "READ_DBL" und "WRIT_DBL" in SCL die gleiche strenge Syntaxregel wie in allen anderen Programmiersprachen:

Im TIA Portal V13 war es möglich, an den Parametern SRCBLK und DSTBLK „Standard“-Datenbausteine anzugeben, deren enthaltene Variablen unterschiedliche Datentypen hatten.

Ab TIA Portal V13 SP1 müssen die einzelnen enthaltenen Variablen sowohl bei „Standard“- als auch bei „Optimierten“-Datenbausteinen den gleichen Datentyp haben, da ansonsten ein Übersetzungsfehler gemeldet wird.

Funktionswerte (Return)

Ab TIA Portal V13 SP1 gelten beim Funktionsaufruf strengere Syntaxregeln:

Es wird geprüft, dass der Funktionswert (Return) auf jeden Fall geschrieben wird, auch wenn in der Funktion mehrere mögliche Programmpfade durchlaufen werden können, z. B. bei Verwendung von Sprungbefehlen oder IF-Schleifen. Die Gefahr, dass Funktionswerte versehentlich zur Laufzeit nicht geschrieben werden, besteht somit nicht mehr.

Möglicherweise erhalten Sie beim Übersetzen jedoch Syntaxfehler in Funktionen, die sich mit V13 noch fehlerfrei übersetzen ließen. Ändern Sie dann den Programmcode, so dass der Funktionswert in allen möglichen Programmpfaden geschrieben wird.

Beispiel:

SCL

```
IF #MyIn1 = #MyIn2 THEN
   #Block_3 := #MyIn1 + 1;
END_IF;
```

In diesem Beispiel wird der Funktionswert von "Block_3" nicht geschrieben, wenn die Bedingung der IF-Anweisung nicht erfüllt ist. Der Funktionswert enthält dann einen undefinierten Wert.

SCL

```
#Block_3 := #MyIn1;
IF #MyIn1 = #MyIn2 THEN
   #Block_3 := #MyIn1 + 1;
END_IF;
```

In diesem Beispiel wird der Funktionswert von "Block_3" auf jeden Fall geschrieben, da er mit "MyIn1" vorbelegt wird, bevor die IF-Schleife durchlaufen wird.

Vergleich der beiden Hardware-Datentypen HW_IO und HW_DEVICE

Ab TIA Portal V13 SP1 gilt beim Vergleich der Datentypen HW_IO und HW_DEVICE eine strengere Syntaxregel:

Bis einschließlich TIA Portal V13 war es möglich, die beiden Datentypen HW_IO und HW_DEVICE direkt miteinander zu vergleichen.

Wenn Sie im TIA Portal V13 SP1 die beiden Datentypen vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Implizite Konvertierung von Gleitpunktzahlen nach DWORD (GRAPH) (S7-1500)

Ab TIA Portal V13 SP1 gelten bei der impliziten Konvertierung von Gleitpunktzahlen nach DWORD strengere Syntaxregeln:

Bis einschließlich TIA Portal V13 war es möglich, eine untypisierte Konstante (z. B. 1.0) implizit in den Datentyp DWORD zu konvertieren, wenn der Wert kleiner als 32 Bit war.

Wenn Sie im TIA Portal V13 SP1 eine Gleitpunktzahl implizit in den Datentyp DWORD konvertieren wollen, dann dürfen Sie nur noch typisierte Konstanten (z. B. REAL#1.0) verwenden, da eine untypisierte Konstante (z. B. 1.0) als Datentyp LREAL interpretiert wird und somit nicht mehr implizit nach DWORD konvertiert werden kann.

Zuweisung eines Wertes an eine LTIME-Konstante (S7-1500)

Ab TIA Portal V13 SP1 gilt bei der Definition eines Wertes an eine LTIME-Konstante eine strengere Syntaxregel:

Bis einschließlich TIA Portal V13 war es möglich, einen TIME-Wert (z.B. T#10s) einer LTIME-Konstanten zuzuweisen.

Wenn Sie im TIA Portal V13 SP1 einen Wert einer LTIME-Konstanten zuweisen möchten, dann müssen Sie eine LTIME-Syntax verwenden. (z. B. LT#10s).

Bitte beachten Sie, dass entsprechende TIA Portal-Projekte aus Vorgängerversionen bereits vor dem Upgrade auf TIA Portal V13 SP1 angepasst werden müssen.

Indirektes Indizieren von ARRAY-Komponenten vom Datentyp Bitfolge in SCL (S7-1200/S7-1500)

Im TIA Portal V13 haben Sie in SCL auf einer CPU der Baureihen S7-1200/1500 die Möglichkeit, zur Adressierung der Komponenten eines ARRAYs als Index neben einer Variablen vom Datentyp Ganzzahl auch eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD anzugeben, wenn die IEC-Prüfung nicht eingestellt ist. Auf einer CPU der Baureihen S7-300/400 ist das nicht möglich.

Dadurch ergibt sich eine Inkompatibilität zum TIA Portal V12 SP1, bei dem das indirekte Indizieren mit dem Datentyp Bitfolge auch auf einer CPU der Baureihen S7-300/400 zulässig ist.

EN-/ENO-Mechanismus bei STRING Konvertierungen (S7-1200/S7-1500)

Konvertierung	Erklärung
Strg_TO_Chars: Zeichenkette in Array of CHAR konvertieren	Der ENO-Freigabeausgang liefert den Signalzustand "0", auch wenn Sie den ENO-Freigabeausgang deaktiviert haben: • Bei einem ungültigen Zeichen am Parameter CHARS • Bei einem ungültigen ARRAY-Index am Parameter PCHARS • Wenn die Summe der Parameter PCHARS und STRG die Länge des Ziel-ARRAYs übersteigt.
Chars_TO_Strg: Array of CHAR in Zeichenkette konvertieren	Der ENO-Freigabeausgang liefert den Signalzustand "0", auch wenn Sie den ENO-Freigabeausgang deaktiviert haben: • Wenn die Summe der Parameter PCHARS und CNT die Länge des Quell-ARRAYs übersteigt.

Überschreitung des Lokaldaten-Stacks (S7-300/S7-400)

Ab TIA Portal V13 SP1 wird geprüft, ob die Größe des Lokaldaten-Stacks, die Sie in der Hardware-Konfiguration für die OB-Prioritäten definiert haben, überschritten wird.

Dadurch erhalten Sie möglicherweise beim Übersetzen Fehlermeldungen in Projekten, die sich mit V13 noch fehlerfrei übersetzen ließen. Ändern Sie dann in der Hardware-Konfiguration die maximale Größe des Lokaldaten-Stacks.

2.5 Inter Project Engineering (IPE)

2.5.1 Hinweise zu IPE

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Systemdiagnose im Geräte-Proxy einsetzen

Um die Funktion "Systemdiagnose" in einem IPE Geräte-Proxy verwenden zu können, z. B. für eine Systemdiagnose-Anzeige, fügen Sie die "PLC-Meldungen" als Inhalt eines Geräte-Proxys ein.

Verbindungen zum SIMATIC S7-1500 Software Controller

Verbindungen zum SIMATIC S7-1500 Software Controller als Geräte-Proxy sind nur über die Schnittstellen möglich, die der Software PLC zugeordnet sind.

Unterstützung von Alarm_S Meldungen

Die Funktionalität Integriertes Projektieren mit WinCC und SIMATIC Manager unterstützt die Alarm_S Meldungen, die über Datenbausteine und Continuous Function Chart (CFC) erreichbar sind.

Kompatibilität

Mit TIA Portal V15.1 können Sie die IPE-Dateien der gleichen Version und früherer Versionen ab V13 SP1 öffnen.

Die IPE-Dateien, die mit der TIA Portal Version V15.1 erstellt wurden, können mit früheren Versionen nicht geöffnet werden.

2.6 Technologiefunktionen

2.6.1 Hinweise zu S7-1500 Motion Control

Ausnahmefehler beim Aktivieren der Festanschlagserkennung mit MC_TorqueLimiting

Wenn Sie einen "MC_TorqueLimiting"-Auftrag mit Parameter "Mode" = 1 starten, ohne dass das Technologieobjekt freigegeben ist, kann es zu einem Ausnahmefehler im Ablaufsystem der CPU kommen. Stellen Sie sicher, dass das Technologieobjekt freigegeben ist (MC_Power.Status = TRUE), bevor Sie einen "MC_TorqueLimiting"-Auftrag starten.

Konfiguration TM PosInput 1/TM PosInput 2 für die Positionserfassung mit dem Technologieobjekt Messtaster

- Technologieobjekt Messtaster
  Für den Einsatz mit Motion Control und einem Technologieobjekt Messtaster, wählen Sie unter "TO_Messtaster > Hardware-Schnittstelle > Messtastereingang" den Messtastertyp "Messen über PROFIdrive Telegramm (Antrieb oder Externer Geber)" aus.
- TM PosInput 1/TM PosInput 2
  Im Betriebsmodus "Positionserfassung für Motion Control" stellen Sie unter "Modulparameter" die Parameter für die Gebersignale des Kanals ein. Die Parameter sind abhängig vom verwendeten Geber einzustellen.

Hinweis

Wenn ein Singleturn Absolutwertgeber verwendet wird und zwei Flanken gemessen werden sollen ("MC_MeasuringInput.Mode" = 2, 3 oder 4), muss der Abstand zwischen den erfassten Flanken des Messeingangs < 1 Geberumdrehung sein. Anderenfalls verwenden Sie einen Multiturn Absolutwertgeber.

Darstellung der Geometrie des Kinematiktyps "Zylindrischer Roboter" (S7-1500T)

Im Gegensatz zur dokumentierten Darstellung des Kinematiktyps "Zylindrischer Roboter" stellt die Länge L1 den Abstand zwischen dem Kinematiknullpunkt (KNP) und der Unterkante des Gelenkarms dar. Die Flanschlänge LF stellt den Abstand zwischen der Unterkante des Gelenkarms und dem Flanschkoordinatensystem (FCS) dar. Dies gilt sowohl für die Variante "3D" als auch für die Variante "3D mit Orientierung".

Die folgende Grafik zeigt beispielhaft den Kinematiktyp "Zylindrischer Roboter 3D" in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellungen der Achsen A1 und A2
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

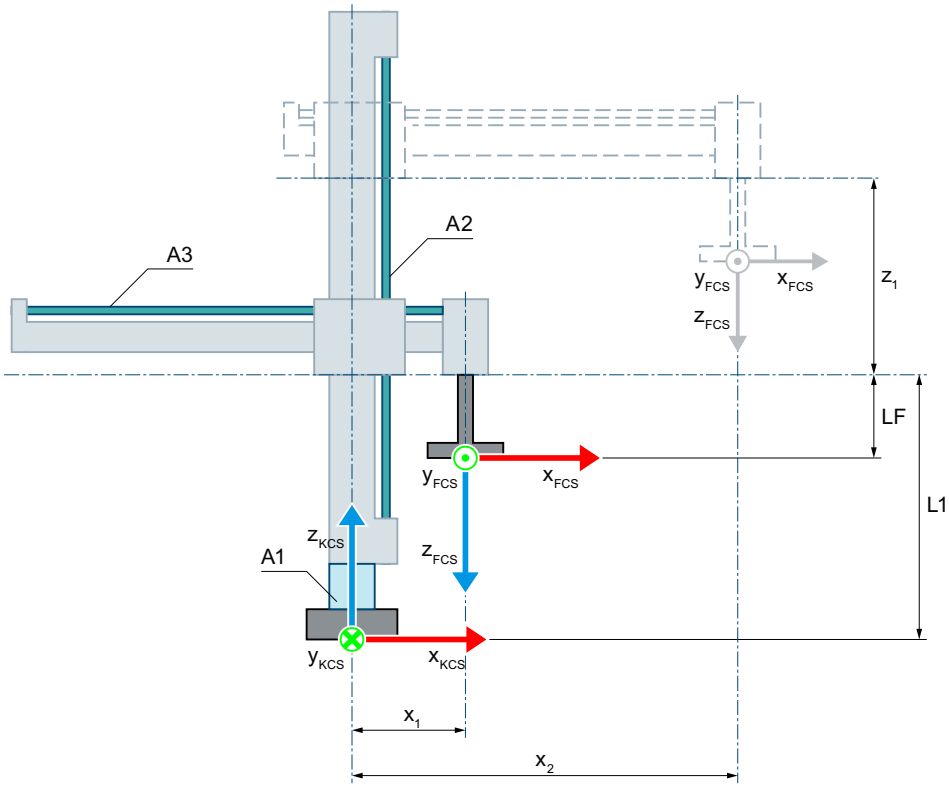

Nullstellung der Achsen A1 und A2

L1 Bei Nullstellung der Achse A2:

Abstand des FCS zum Kinematiknullpunkt (KNP) und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

x_1 Positive Auslenkung der Achse A3

In der Nullstellung der Achse A3 liegt die z-Achse des FCS auf der z-Achse des KCS. Die dargestellte Kinematik kann die Nullposition der Achse A3 mechanisch bedingt nicht anfahren.

Auslenkung der Kinematik

x_2 Positive Auslenkung der Achse A3

z_1 Positive Auslenkung der Achse A2

Bewegungsausführung mit einem "MC_GroupInterrupt"-Auftrag unterbrechen (S7-1500T)

Im Gegensatz zum dokumentierten Verhalten können Sie die Bewegungsausführung am Technologieobjekt Kinematik nur während einer aktiven Kinematikbewegung mit einem "MC_GroupInterrupt"-Auftrag unterbrechen. Wenn sich die Kinematik bereits im Stillstand befindet, ist die Bewegungsausführung für folgende Bewegungsaufträge nicht unterbrochen. Die Kinematik befindet sich nicht im Zustand "Interrupted" (StatusWord.X17 (Interrupted)).

Indem Sie für den Geschwindigkeits-Override den Wert "0.0" angeben, können Sie die Bewegung so lange zurückhalten, bis alle gewünschten Aufträge in die Auftragskette eingetragen sind. Anschließend können Sie mit einem Geschwindigkeits-Override von "100.0" die Bewegung fortsetzen.

Wirkung des Geschwindigkeits-Override auf Kinematikbewegungsaufträge (S7-1500T)

Wenn Sie den Geschwindigkeits-Override ändern, treten bei Bewegungsaufträgen des Technologieobjekts Kinematik, bei denen die Orientierungsbewegung die Dauer der Gesamtbewegung bestimmt, unterschiedliche Ergebnisse auf. Dies ist abhängig davon, ob Sie den Geschwindigkeits-Override vor oder nach dem Absetzen des Bewegungsauftrags ändern.

Bei Bewegungsaufträgen, die sich bereits in der Auftragskette ("Busy" = TRUE) oder in der Ausführung ("Active" = TRUE) befinden, ist die Kinematikbewegung bereits auf die mögliche Orientierungsgeschwindigkeit reduziert. Somit wirkt sich die Änderung des Geschwindigkeits-Override nur auf die Bahngeschwindigkeit aus.

Bei Bewegungsaufträgen, die nach der Änderung des Geschwindigkeits-Override der Auftragskette hinzugefügt werden, wirkt der geänderte Geschwindigkeits-Override auch auf die Orientierungsgeschwindigkeit.

Ändern Sie deshalb den Wert des Geschwindigkeits-Override, bevor Sie Bewegungsaufträge absetzen.

Geschwindigkeits-Override mit der Kinematiksteuertafel steuern (S7-1500T)

Im Gegensatz zum dokumentierten Verhalten können Sie den Schieberegler zum Einstellen der Geschwindigkeit nur im Weltkoordinatensystem (WCS) und Objektkoordinatensystem (OCS) verwenden. Kinematikbewegungen im Maschinenkoordinatensystem (MCS) werden mit dem Geschwindigkeits-Override (<TO>.Override.Velocity) der einzelnen verschalteten Achsen ausgeführt.

"MC_GroupStop" nach "MC_GroupInterrupt" ausführen (S7-1500T)

Wenn Sie nach einem "MC_GroupInterrupt"-Auftrag einen "MC_GroupStop"-Auftrag ausführen, müssen Sie den Parameter "Execute" am "MC_GroupStop" so lange auf TRUE lassen, bis die Auftragskette leer ist (StatusMotionQueue.NumberOfCommands = 0). Wenn die Auftragskette nicht leer ist, werden die verbleibenden Bewegungsaufträge aus der Auftragskette ausgeführt.

2.6.2 Hinweise zu weiteren Technologiefunktionen

Parameter der Technologieobjekte in TIA Portal Openness

Sie finden eine Auflistung der verfügbaren Technologieobjekt-Parameter in der Produktinformation "Parameter der Technologieobjekte in TIA Portal Openness" im Internet (https://support.industry.siemens.com/cs/ww/de/view/109744932).

S_USSI an S7-1500

Beachten Sie folgendes, falls Sie die Anweisung S_USSI der dezentralen Peripherie ET 200S 1SI an einer S7-1500 einsetzen.

Der Parameter ANZ zeigt auch im Fehlerfall den Wert 0.

Prüfen Sie, falls die CPU durch die Anweisung S_USSI in STOP geschaltet wird, alle beim Parameter ANZ aufgelisteten Fehlermöglichkeiten.

PtP-Module mit CM 1243-5 PROFIBUS Master

Beim Einsatz der PtP-Module CM PtP RS232 BA, CM PtP RS422/485 BA, CM PtP RS232 HF, CM PtP RS422/485 HF und CM PtP mit einem CM 1243-5 PROFIBUS Master mit einer Firmware ≤ V1.3.4 müssen Sie im Instanz-DB der Anweisungen folgende Einstellungen treffen:

- Send_P2P
  max_record_len = 240
- Modbus_Master
  Send_P2P.max_record_len = 240
- Modbus_Slave
  Send_P2P.max_record_len = 240

PID_Temp auf CPU 1500 V1.7

Beachten Sie beim Einsatz von PID_Temp auf einer CPU 1500 mit Firmware-Version V1.7 folgendes:

- Die in PID_Temp integrierte Totzone darf auf CPU 1500 V1.7 nicht genutzt werden. Ändern Sie die Vorbelegung 0.0 der zugehörigen Parameter Retain.CtrlParams.Heat.DeadZone und Retain.CtrlParams.Cool.DeadZone nicht.
- Ist bei einem Regler mit aktivierter Kühlung und PID-Parameterumschaltung (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) der Kühlausgang aktiv (PidOutputSum < 0.0, OutputCool <> 0.0) während eine "Erstoptimierung Kühlen" gestartet wird (Mode = 1, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE), bricht diese mit Fehler ErrorBits = 16#0040_0000 ab.
  - Verwenden Sie in diesem Fall die "Nachoptimierung Kühlen" (Mode = 2, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE)

  oder

  - wechseln Sie vor dem Start der Erstoptimierung in den Handbetrieb mit einem Handwert größer 0.0 (Mode = 4, ManualValue ≥ 0.0).

Beide Einschränkungen bestehen ab Firmware-Version V1.8 nicht mehr.

2.6.3 Hinweise zu Technologiefunktionen

Anzeigefehler im Online/Offline-Vergleich bei S7-1200 Motion Control

Wenn Sie ein TIA Portal Projekt V15 auf V15.1 hochrüsten und eine Online-Verbindung zur CPU aufbauen, auf die das Projekt mit TIA Portal V15 geladen wurde, dann werden Unterschiede im Online / Offline-Vergleich angezeigt, die nicht existieren. Das können auch ungültige Konfigurationen sein.

Laden Sie das hochgerüstete Projekt in die CPU. Der Online/Offline-Vergleich zeigt wieder korrekte Ergebnisse an.

Parameter der Technologieobjekte in TIA Portal Openness

Sie finden eine Auflistung der verfügbaren Technologieobjekt-Parameter in der Produktinformation "Parameter der Technologieobjekte in TIA Portal Openness" im Internet (https://support.industry.siemens.com/cs/ww/de/view/109744932).

PtP-Module mit CM 1243-5 PROFIBUS Master

Beim Einsatz der PtP-Module CM PtP RS232 BA, CM PtP RS422/485 BA, CM PtP RS232 HF, CM PtP RS422/485 HF und CM PtP mit einem CM 1243-5 PROFIBUS Master mit einer Firmware ≤ V1.3.4 müssen Sie im Instanz-DB der Anweisungen folgende Einstellungen treffen:

- **Send_P2P**
  max_record_len = 240
- **Modbus_Master**
  Send_P2P.max_record_len = 240
- **Modbus_Slave**
  Send_P2P.max_record_len = 240

2.6.4 Hinweise zu Technologiefunktionen

Parametrierung FM x51 und FM x52

Entgegen den Angaben in der Online-Hilfe werden bei einer Änderung des Maßsystems die bereits eingegebenen Parameterwerte nicht in das neue Maßsystem umgerechnet.

Programmierung FM x50-1, FM x51 und FM x52

Die in der Online-Hilfe zu den Anweisungen angegebenen "Anfangswerte" in den Parametertabellen entsprechen dem "Defaultwert" in der SW-Oberfläche der Anwenderprogrammierung.

Migration IM 174

Bei der Migration von Projekten mit IM 174 wird die Parametrierung der IM 174 auf Defaultwerte gesetzt. Die IM 174 muss nach der Migration in jedem Fall neu parametriert werden.

2.6.5 SIMATIC STEP 7 Easy Motion Control

2.6.5.1 Hinweise zu SIMATIC STEP 7 Easy Motion Control

Es liegen keine Hinweise zu SIMATIC STEP 7 Easy Motion Control vor.

2.6.6 SIMATIC STEP 7 PID Professional

2.6.6.1 Hinweise zu SIMATIC STEP 7 PID Professional

Es liegen keine Hinweise zu SIMATIC STEP 7 PID Professional vor.

WinCC 3

3.1 Security-Hinweise

Security-Hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Lösungen, Maschinen, Geräten und/oder Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und entsprechende Schutzmaßnahmen (z.B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weitergehende Informationen über Industrial Security finden Sie unter:

http://www.siemens.com/industrialsecurity (http://www.industry.siemens.com/topics/global/de/industrial-security/Seiten/Default.aspx)

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter:

http://www.siemens.com/industrialsecurity (http://www.industry.siemens.com/topics/global/de/industrial-security/Seiten/Default.aspx)

Kennwörter

In WinCC sind verschiedene Kennwörter voreingestellt. Aus Sicherheitsgründen sollten Sie diese Kennwörter ändern.

- Bei Bediengeräten der Version 12 ist für den Sm@rtServer und für den integrierten Webserver das Kennwort "100" voreingestellt. Bei Bediengeräten der Version 13 ist kein Kennwort voreingestellt.
- Für den Benutzer "Administrator" ist das Kennwort "administrator" voreingestellt.

Integrierte Webserver

Auf einem PC mit ist es in Runtime möglich auf die HTML-Seiten zuzugreifen, obwohl die Option "HTML-Seiten" deaktiviert ist. Durch die Installation werden auf dem PC immer die Standardseiten des Webservers abgelegt. Vergeben Sie ein Administrator-Kennwort, das den unbefugten Zugriff auf die Seiten verhindert.

Kommunikation über Ethernet

Bei Ethernet-basierter Kommunikation ist der Endanwender für die Sicherheit seines Datennetzes selbst verantwortlich, da z. B. bei gezielten Angriffen, die zur Überlast des Gerätes führen, die Funktionsfähigkeit nicht garantiert ist.

Verwendung von SSL 3.0

Aus Security-Gründen ist die Verwendung des Protokolls SSL 3.0 auf Comfort Panels und in Runtime Advanced nicht empfohlen. Auf Comfort Panels ist die Verwendung des Protokolls SSL 3.0 standardmäßig deaktiviert. Wenn Sie dennoch die Verwendung von SSL 3.0 aktiveren möchten, wählen Sie im Internet Explorer oder im "Start Center > Settings" unter „Internet Options > Advanced > Use SSL 3.0“.

Für RT Advanced kann die Verwendung von SSL 3.0 im Internet Explorer oder über Systemsteuerung unter „Internetoptionen > Erweitert“ durch Deaktivieren der Option „SSL 3.0 verwenden“ abgeschaltet werden.

Netzwerkeinstellungen

Die folgenden Tabellen zeigen die Netzwerkeinstellungen des jeweiligen Produkts, die Sie zur Analyse der Netzwerksicherheit und zur Konfiguration von externen Firewalls benötigten:

WinCC Professional (ohne Simulation)					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
ALM	4410*	TCP	Inbound, Outbound	Lizenz-Service	Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet.
HMI Load	1033	TCP	Outbound	HMI Load (RT Basic)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.
HMI Load	2308	TCP	Outbound	HMI Load (RT Advanced)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Panels zu übertragen.
RPC	**	UDP	Inbound, Outbound	Client / server & ES communication (CCAgent)	Dieser Dienst wird von WinCC Professional und WinCC Runtime Professional verwendet.
* Standard-Port, der durch Anwenderkonfiguration geändert werden kann ** Port wird automatisch vergeben					

WinCC Simulation für Basic Panels					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
HMI Load	1033	TCP	Inbound	HMI Load (RT Basic)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.
EtherNet/IP	44818	TCP	Outbound	Ethernet/IP-Kanal	Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.
	2222	UDP	Inbound	Ethernet/IP-Kanal	Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.
Modbus TCP	502	TCP	Outbound	Modbus TCP-Kanal	Das Modbus TCP-Protokoll wird für Verbindungen zu Schneider PLCs verwendet.
RFC 1006	102	TCP	Outbound	S7-Kanal	Kommunikation zum S7-Controller über Ethernet / PROFINET
Mitsubishi MC	5002	TCP	Outbound	Mitsubishi MC-Kanal	Das Mitsubishi-Protokoll wird für Verbindungen zu Mitsubishi PLCs verwendet.

WinCC Simulation für Panels und Runtime Advanced					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
DCP	---	Ethernet	Outbound	PROFINET	Das DCP-Protokoll (Discovery and basic Configuration Protocol) wird von PROFINET verwendet und stellt die Basisfunktionalität zum Auffinden und Konfigurieren von PROFINET-Geräten zur Verfügung.
LLDP	---	Ethernet	Inbound, Outbound	PROFINET	Das LLDP-Protokoll (Link Layer Discover Protocol) wird von PROFINET zur Topologie-Erkennung verwendet.
SMTP	25	TCP	Outbound	SMTP Communication	Dieser Dienst wird von WinCC Runtime Advanced zum Versenden von E-Mails verwendet.
HTTP	80*	TCP	Inbound	Sm@rtServer	Der Webserver ist nur vorhanden, wenn Sm@rtService aktiviert wird. Der verwendete Port kann sich aufgrund von automatisch gewählten Einstellungen unterscheiden.
RFC 1006	102	TCP	Outbound	S7-Kanal	Kommunikation zum S7-Controller über Ethernet / PROFINET
NTP	123	UDP	Outbound	Uhrzeit-synchronisation	Das NTP-Protokoll (Network Time Protocol) wird in IP-basierten Netzwerken zur Uhrzeitsynchronisation verwendet.
SNMP	161	UDP	Outbound	PROFINET	Die SNMP Client-Funktionalität wird von STEP 7 verwendet, um Statusinformationen von PROFINET-Geräten auszulesen.
HMI Load	2308	TCP	Outbound	HMI Load (RT Advanced)	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Panels zu übertragen.

WinCC Simulation für Panels und Runtime Advanced					
HTTPS	443*	TCP	Inbound	Sm@rtServer	Der Webserver mit HTTPS-Protokoll ist nur vorhanden, wenn Sm@rtService aktiviert wird. Der verwendete Port kann sich aufgrund von automatisch gewählten Einstellungen unterscheiden.
VNC Server	5900*	TCP	Inbound	Sm@rtServer	Dieser Dienst ist nur vorhanden, wenn Sm@rtService aktiviert wird.
	5800*	TCP	Inbound	Sm@rtServer	Dieser Dienst ist nur vorhanden, wenn Sm@rtService aktiviert wird.
VNC Client	5500	TCP	Outbound	Sm@rtServer	Dieser Dienst ist nur vorhanden, wenn Sm@rtService aktiviert wird.
* Standard-Port, der durch Anwenderkonfiguration geändert werden kann					

WinCC Simulation für Runtime Professional					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
RPC	**	UDP	Inbound, Outbound	Client / server & ES communication (CCAgent)	Dieser Dienst wird von WinCC Professional und WinCC RT Professional verwendet.
RPC	**	UDP	Inbound, Outbound	Client / server communication (CCEServer / CCEClient)	Dieser Dienst wird von WinCC Runtime Professional verwendet.
HTTP	80	TCP	Inbound, Outbound	Client / server communication (CCEServer / CCEClient)	Dieser Dienst wird von WinCC Runtime Professional verwendet.
RFC 1006	102	TCP	Outbound	S7-Kanal	Kommunikation zum S7-Controller über Ethernet / PROFINET
OPC UA	4840	TCP	Inbound	OPC UA Server	Dieser Dienst wird zur primären Kommunikation über OPC UA benötigt. Er wird bei der Installation aktiviert und konfiguriert.
OPC UA discovery	52601	TCP	Inbound	OPC UA Server	Dieser Dienst liefert Informationen über installierte OPC-Server. Er wird durch den OPC UA Server installiert und konfiguriert.
DCOM	135	TCP	Inbound	OPC Server	Dieser Dienst ist Teil des Betriebssystems Windows. Da die Kommunikation über OPC (DA) auf DCOM basiert, wird dieser Dienst zur Initialisierung von OPC (DA) Verbindungen benötigt.
DCOM	**	TCP	Inbound	OPC Server	Die Kommunikation über OPC (DA) basiert auf DCOM und verwendet unspezifizierte vom System vergebene Ports. Dies sollte bei der Verwendung von OPC (DA) und der Erstellung von Regeln für die Firewall berücksichtigt werden.
HTTP	80	TCP	Inbound	OPC Server	Dieser Dienst wird zur primären Kommunikation über OPC XML benötigt. Er wird bei der Installation aktiviert und konfiguriert.

WinCC Simulation für Runtime Professional					
NetBIOS	137	UDP	Inbound	OPC Server	Dieser Dienst ist Teil des Betriebssystems Windows. Der Zugriff auf diesen Dienst wird z. B. von OPC-Scout zum Browsen benötigt.
NetBIOS	138	UDP	Inbound	OPC Server	Dieser Dienst ist Teil des Betriebssystems Windows. Der Zugriff auf diesen Dienst wird z. B. von OPC-Scout zum Browsen benötigt.
SNMP	161	UDP	Outbound	SNMP OPC Server	Dieser Dienst wird vom SNMP OPC Server verwendet, um Daten auf z. B. Netzlaufwerken zu ändern oder abzufragen.
SNMP Traps	162	UDP	Inbound	SNMP OPC Server	Dieser Dienst wird vom SNMP OPC Server verwendet, um Ereignisse von z. B. Netzlaufwerken abzufragen.
** Port wird automatisch vergeben					

PROFINET-Protokolle für Panels und Runtime Advanced					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
DCP	---	Ethernet	Outbound	Lifelist, PROFINET Discovery and configuration	Das DCP-Protokoll (Discovery and basic Configuration Protocol) wird von PROFINET verwendet und stellt die Basisfunktionalität zum Auffinden und Konfigurieren von PROFINET-Geräten zur Verfügung.
LLDP	---	Ethernet	Inbound, Outbound	PROFINET Link Layer Discovery protocol	Das LLDP-Protokoll (Link Layer Discover Protocol) wird von PROFINET zur Topologie-Erkennung verwendet.
MRP	---	Ethernet	Outbound	PROFINET medium redundancy	Das MRP-Protokoll (Medium redundancy protocol) ermöglicht die Steuerung von redundanten Übertragungswegen durch eine Ringtopologie.
PROFINET IO Data	---	Ethernet	Inbound, Outbound	PROFINET Cyclic IO data transfer	Der zyklische Datenaustausch wird von Panels für Direkttasten und LEDs verwendet.
NARE	---	Ethernet	Inbound, Outbound	Name Addres Resolution	Dieses Protokoll wird verwendet, um Netzwerknamen aufzulösen und IP-Adressen zuzuweisen.
PROFINET Context Manager	34964	UDP	Inbound, Outbound	PROFINET connection less RPC	Der PROFINET Context Manager stellt einen Endpoint-Mapper zur Verfügung, um eine Applikationsbeziehung (PROFINET AR) herzustellen.

Kommunikationsverbindungen für Panels und WinCC Runtime Advanced					
Name	**Port-Nummer**	**Transport-Protokoll**	**Richtung**	**Funktion**	**Beschreibung**
Telnet	23	TCP	Inbound	Telnet	Dieser Dienst kann für Wartung verwendet werden.
SMTP	25	TCP	Outbound	SendEMail	Dieser Dienst wird von Windows CE / PC Runtime zum Versenden von E-Mails verwendet.

Kommunikationsverbindungen für Panels und WinCC Runtime Advanced					
HTTP	80*	TCP	Intbound	Hypertext Transfer Protocol	Der HTTP Protokoll wird für die Kommunikation mit dem internen Webserver verwendet.
RFC 1006	102	TCP	Outbound	S7-Kanal	Kommunikation zum S7-Controller über Ethernet / PROFINET.
NTP	123	UDP	Outbound	Uhrzeit-synchronisation	Das NTP-Protokoll (Network Time Protocol) wird in IP-basierten Netzwerken zur Uhrzeitsynchronisation verwendet.
DCOM***	135	TCP	Inbound	OPC Server	Dieser Dienst ist ein Bestandteil des Betriebssystems Microsoft Windows. Kommunikation über OPC (DA) basiert auf DCOM. Deshalb wird dieser Dienst benötigt, um OPC (DA) Verbindungen zu initialisieren.
DCOM***	**	TCP	Inbound	OPC Server	Die Kommunikation über OPC (DA) basiert auf DCOM und verwendet unspezifizierte vom System vergebene Ports. Dies sollte bei der Verwenundung von OPC (DA) und der Erstellung von Regeln für die Firewall berücksichtigt werden.
NetBIOS over TCP/IP	137	UDP	Outbound	Mit Verwendung von Remote-Fileshare	Anmeldung/Einloggen auf einem Remote-Server.
NetBIOS over TCP/IP	138	UDP	Outbound	Mit Verwendung von Remote-Fileshare	Anmeldung/Einloggen auf einem Remote-Server.
SNMP	161	UDP	Outbound	Simple Network Management Protocol	Die SNMP Client-Funktionalität wird von STEP 7 verwendet, um Statusinformationen von PROFINET-Geräten auszulesen.
HTTPS	443*	TCP	Inbound	Secure Hypertext Transfer Protovol	Das HTTP-Protokoll wird zur Kommunikation mit dem CPU-internen Webserver über Secure Socket Layer (SSL) verwendet.
Modbus TCP	502*	TCP	Outbound	Modbus TCP-Kanal	Das Modbus TCP-Protokoll wird für Verbindungen zu Schneider PLCs verwendet.
Mitsubishi MC	1025*	TCP	Outbound	Mitsubishi MC-Kanal	Das Mitsubishi-Protokoll wird für Verbindungen zu Mitsubishi PLCs verwendet.
Printing	1032	TCP	Outbound	Printing	Drucken auf dem Bedienfeld (über das Ethernet).
HMI Load	2308	TCP	Outbound	Transfer	Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Panels zu übertragen. Auf Comfort Panels wird dieser Dienst ab V13 durch DeviceManager und SCS ersetzt.Dieser Dienst wird verwendet, um Konfigurationsdaten zu WinCC Runtime Advanced zu übertragen.

Kommunikationsverbindungen für Panels und WinCC Runtime Advanced					
HMI Load	50523	TCP	Outbound	Transfer	Dieser Port wird verwendet, wenn Port 2308 nicht verfügbar ist. Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Panels zu übertragen. Auf Comfort Panels wird dieser Dienst ab V13 durch DeviceManager und SCS ersetzt. Dieser Dienst wird verwendet, um Konfigurationsdaten zu WinCC Runtime Advanced zu übertragen.
ALM	4410*	TCP	Inbound, Outbound	Application License Manager	Dieser Dienst von RT Advanced stellt die komplette Funktionalitäten für Software-Lizenzen zur Verfügung und wird vom Automation License Manager verwendet.
OPC UA	4870*	TCP	Inbound	OPC UA Server	Dieser Dienst wird zur Kommunikation über OPC UA benötigt.
HMI Load	5001	TCP	Outbound	Device Manager	Dieser Dienst wird verwendet, um Images und Runtime auf Panels zu übertragen.
HMI Load	5002	TCP	Outbound	SCS (System Configuration Server)	Dieser Dienst wird verwendet, um Konfigurationsdaten zu Panels zu übertragen.
VNC Client	5500	TCP	Outbound	Sm@rtServer	VNC Client Verbindung
VNC Server	5800*	TCP	Inbound	Sm@rtServer	VNC Server Verbindung HTTP
	5900*	TCP	Inbound	Sm@rtServer	VNC Server Verbindung
SIMATIC Logon	16389*	TCP	Outbound	UMAC (User Management ans Access Control)	Anmeldung/Einloggen auf einem Remote-Server.
Allen Bradley Ethernet/IP	44818	TCP	Outbound	Ethernet/IP-Kanal	Das Ethernet/IP Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.
Reserved	49152 ... 65535	TCP/UDP	Outbound		Dynamisches Portbereich wird z. B. zum Verbinden zum Remote-Filesharing verwendet.

* Standard-Port, der durch Anwenderkonfiguration geändert werden kann

** Port wird automatisch vergeben.

*** Wird nur von WinCC Runtime Advanced unterstützt.

3.2 Hinweise zur Benutzung

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verhaltensänderung nach dem Hochrüsten

Die neue WinCC Version enthält neue Funktionen und Verbesserungen gegenüber der Vorgängerversion. Dadurch können die Bezeichnungen oder das Verhalten von Eigenschaften und Funktionen nach dem Hochrüsten auf die neue Version kleine Änderungen aufzeigen. Auch die Standard-Eigenschaften von neu angelegten Objekten können gegenüber früheren Versionen verändert sein.

Diese Verhaltensänderungen können auch zu Fehlern beim Übersetzen des Projekts führen.

Hochrüsten mit anschließendem Gerätewechsel (RT Advanced)

Wenn Sie ein Projekt mit einem Gerät "Runtime Advanced" hochrüsten und anschließend einen Gerätetausch auf ein Panel mit Bediengeräteversion 14.0.1.0 oder 15.0.0.0 durchführen, können die projektierten Daten verloren gehen. Es wird empfohlen, dass Sie nach dem Hochrüsten entweder

- zuerst einen Gerätewechsel auf einen IPC ohne Bildanpassung durchführen und anschließend wieder zu einer PC-Station mit Runtime Advanced zu wechseln. Danach können Sie den gewünschten Gerätetausch auf ein Panel ohne Datenverlust durchführen.
- zuerst einen Gerätewechsel auf eine Panel mit einer Bediengeräteversion kleiner als 14.0.1.0 durchzuführen. Anschließend können Sie den Gerätewechsel auf ein Panel mit der gewünschten Geräteversion ohne Datenverlust durchführen.

Arbeiten mit Standard-Benutzerrechten

Wenn Sie unter Windows 7 mit Standard-Benutzerrechten arbeiten, darf die "Benutzerkontensteuerung" (UAC) nicht deaktiviert sein.

Standardmäßig ist die "Benutzerkontensteuerung" unter Windows 7 aktiviert.

Weitere Informationen zur "Benutzerkontensteuerung" entnehmen Sie bitte aus der Onlinehilfe von Windows 7.

Bildschirmtastatur (RT Advanced)

Wenn Sie das TIA Portal geöffnet haben, dann können Sie die Bildschirmtastatur nicht mehr aufrufen.

Verwenden Sie für den Aufruf der Bildschirmtastatur in Windows: „Start > Alle Programme > Zubehör > Erleichterte Bedienung > Bildschirmtastatur“.

Installation auf einem Computer mit regionaler Einstellung "Türkisch" (RT Professional)

Wenn zum Zeitpunkt der Installation der Computer mit der regionalen Einstellung "Türkisch" betrieben wird, kann WinCC Runtime Professional nicht gestartet werden.

Gerätewechsel - Zeichensätze (RT Professional)

Wenn Sie an Bildobjekten für mehrsprachige Texte in den verschiedenen Projektsprachen unterschiedliche Zeichensätze projektiert haben, sollten Sie nach einem Bediengerätewechsel von einem Panel auf eine PC-basierte Runtime die Bilder überprüfen. Es kann vorkommen, dass die projektierten Zeichensätze durch den Gerätewechsel verändert wurden.

Texte in der Simulation (RT Professional)

Wenn in der Simulation eine Runtime-Sprachen verwendet wird, die nicht einer installierten Produktsprache entspricht, so werden Systemtexte in einer installierten Produktsprache angezeigt. Abhängig vom Betriebssystem und den installierten Produktsprachen ist dies entweder Englisch oder Chinesisch.

Wenn Sie in Japanisch, Koreanisch oder Taiwanesisch in der Simulation alle Texte in der jeweiligen Sprache sehen möchten, sollten Sie WinCC zunächst ohne Simulation installieren. Anschließend installieren Sie WinCC Runtime Professional mit den benötigten Sprachen.

Online-Deltaload mit benutzerdefinierten Zyklen (RT Professional)

Wenn Sie im Editor "Zyklen" eigene Zyklen neu projektiert haben, wählen Sie beim Laden des Projekts den Befehl "Laden > Software (komplett übersetzen)". Ein Online-Deltaladen ist nicht möglich.

Lizenztransfer über S7USB

Der Lizenztransfer zu einem Panel über S7USB ist nur aus WinCC heraus möglich.

Lizenztransfer auf ein Panel unter 64-Bit-Betriebssystemen

Wenn auf einem 64-Bit-Betriebssystem im Automation License Manager der Menübefehl "Bearbeiten > Zielsysteme verbinden > Bediengerät verbinden" nicht zur Verfügung steht, öffnen Sie bitte die Kommandozeile und führen Sie, mit Administrator-Rechten, folgenden Befehl aus:

"%WINDIR%\system32\RegSvr32.exe" "%CommonProgramFiles%\siemens\AlmPanelPlugin\ALMPanelParam.dll"

Ostasiatische Projektsprachen projektieren auf einem PC ohne asiatisches Betriebssystem (Basic)

Wenn Sie auf Ihrem PC kein asiatisches Betriebssystem installiert haben, aber in der Projektierung eine ostasiatische Projektsprache wählen, wird im Editor "Runtime-Einstellungen > Sprache & Schriftart" die Standardschrift als ungültig markiert.

Um das Problem zu beheben, aktivieren Sie in der Systemsteuerung "Regions- und Sprachoption > Sprachen" die Option "Dateien für ostasiatische Sprachen installieren".

Betrieb auf einem Netzwerk Server (RT Professional)

Der Betrieb von WinCC Professional auf einem Netzwerk Server (z. B. Domain Controller, File- and Name Utility Server, Router, Software Firewall, Media Server, Exchange Server) ist nicht zulässig.

File Browser auf einem Windows 10 PC mit Touch-Screen

Auf einem Windows 10 PC mit einem Touch-Screen kann der File Browser Dialog nur mit der Maus, Tastatur oder Bildschirmtatstatur (ohne Verwendung von Touch-Funktion) bedient werden. Es wird empfohlen, auf einem Touch-Screen PC mit Windows 10 den File Browser Dialog von Windows Betriebssystem mithilfe eines Skripts zu nutzen.

Postscript-Zeichensätze

Im TIA Portal können nur Schriftarten verwendet werden, die nur True Type Daten enthalten. PostScript-Schriftarten und Open-Type-Schriftarten, die PostScript-Daten enthalten, werden nicht unterstützt.

3.3 Migration

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Migration eines integrierten Projekts mit ProTool-Objekten

Wenn Sie ein in STEP 7 integriertes WinCC flexible Projekt migrieren und die Fehlermeldung "Optionspaket(e) PROTOOL in STEP 7 fehlen" auftritt, haben Sie WinCC flexible 2008 SP3 oder neuer installiert. Zudem enthält das Projekt noch Objekte, die mit ProTool projektiert worden sind. Öffnen Sie das Projekt nicht mit WinCC flexible. Um das Projekt zu migrieren, gehen Sie wie folgt vor:

1. Öffnen Sie das Projekt im SIMATIC Manager.
2. Führen Sie im Menü "Datei" den Befehl "Speichern unter" aus.
3. Aktivieren Sie im Dialog "Projekt speichern untern" die Option "Mit Reorganisation".
4. Klicken Sie auf OK.
5. Kopieren Sie das Projekt wieder auf den ursprünglichen Rechner.
6. Starten Sie die Migration erneut.

Migration eines WinCC V7 Projekts: Rahmenlinie von Rechtecken (RT Professional)

Sie haben in einem WinCC V7 Projekt ein Rechteck mit den Einstellungen "Linienstärke=1" und "Rahmen innen zeichnen= ja" projektiert.

Sie migrieren das WinCC V7 Projekt nach WinCC V15. Damit das Rechteck korrekt dargestellt wird, gehen Sie wie folgt vor.

1. Öffnen Sie das Inspektorfenster des Rechtecks.
2. Öffnen Sie die Eigenschaftsliste.
3. Deaktivieren Sie "Rahmenlinie nach innen verbreitern".

Fortschrittsanzeige

Wenn die Fortschrittsanzeige 100% anzeigt, werden von der Software noch verschiedene Restarbeiten, z. B. das Schließen von Referenzen, durchgeführt. In dieser Zeit wird die Software nicht auf Eingaben im User Interface reagieren.

Unterstützte Projektsprachen in WinCC V7 (RT Professional)

Bedingt durch unterschiedliche Windows-DotNet-Versionen kann dieselbe Sprache unterschiedliche Bezeichnungen in WinCC V7 und im TIA Portal haben. Dadurch können sich in demselben Projekt unterschiedliche Sortierreihenfolgen ergeben.

Folgende Tabelle zeigt einige Beispiele in englischer Schreibweise:

WinCC V7.x	TIA Portal
Bengali (Bangladesh)	Bangla (Bangladesh)
Bengali (India)	Bangla (India)
Chinese (Simplified, PRC)	Chinese (People's Republic of China)
Chinese (Simplified, Singapore)	Chinese (Singapore)
Chinese (Traditional, Hong Kong S.A.R)	Chinese (Hong Kong S.A.R)
Chinese (Traditional, Macao S.A.R)	Chinese (Macao S.A.R)
Chinese (Traditional, Taiwan)	Chinese (Taiwan)
French (Monaco)	French (Principality of Monaco)
Oriya (India)	Odia (India)
Sami (Inari, Finland)	Sami, Inari (Finland)
Sami (Skolt, Finland)	Sami, Skolt (Finland)
Sami, Northern (Norway)$Sami	Sami, Northern (Norway)
Spanish (Bolivarian Republic of Venezuela)	Spanish (Venezuela)
Spanish (Spain, Traditional Sort)	Spanish (Spain)
Spanish (Spain, International Sort)	Spanish (Spain)
Yakut (Russia)	Sakha (Russia)

3.4 Engineering System

3.4.1 Bilder und Bildobjekte

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Anzeigeobjekte nach Bediengerätewechsel

Wenn Sie ein Gerät auf die neuen Bediengeräteversionen hochrüsten, sollten Sie die im Projekt enthaltenen Bilder überprüfen. Aufgrund des neuen Aussehens und der verbesserten Bedienmöglichkeit kann es vorkommen, dass z. B bei symbolischen EA-Feldern die Texte nicht vollständig lesbar sind und durch Bedienelemente überlagert werden.

Gruppierung von Bildobjekten

Wenn Sie Bildobjekte in WinCC gruppieren, dann kann es bei tieferen Verschachtelungen zu Performance-Problemen von WinCC kommen.

ActiveX- und .NET-Controls

Aus Gründen der Stabilität und Performance sind in WinCC nicht alle registrierten ActiveX-Controls projektierbar. Nur die im Auswahldialog sichtbaren Controls können in WinCC verwendet werden.

In Runtime werden ActiveX- und .NET-Controls immer im Vordergrund positioniert.

Die Projektierung von ActiveX- und .NET-Controls in Ebenen wird nicht unterstützt.

Nutzung von WinCC Control Development

WinCC Control Development kann ohne Lizenz genutzt werden.

Die aktuelle Version von WinCC Control Development zum Herunterladen finden Sie im Siemens Industry Online Support im Beitrag 109760182 "Control Development Kit" (https://support.industry.siemens.com/cs/us/en/view/109760182).

Rahmenlinie von Rechtecken (RT Professional)

Sie haben in einem WinCC V7 Projekt ein Rechteck mit den Einstellungen "Linienstärke=1" und "Rahmen innen zeichnen= ja" projektiert.

Sie migrieren das WinCC V7 Projekt nach WinCC. Damit das Rechteck korrekt dargestellt wird, gehen Sie wie folgt vor.

1. Öffnen Sie das Inspektorfenster des Rechtecks.
2. Öffnen Sie die Eigenschaftsliste.
3. Deaktivieren Sie "Rahmenlinie nach innen verbreitern".

Vorschau im Bildfenster (RT Professional)

Sie verwenden eigene Designs mit Schatten bei Bildobjekten. Die Bildobjekte lassen Sie in einem Bildfenster anzeigen.
Der Schatten der Bildobjekte wird in der Vorschau des Bildfensters nicht angezeigt. Das Verhalten tritt nur im Engineering System auf. In Runtime wird es korrekt dargestellt.

Viele visuell unterschiedliche Bildobjekte

Die Verwendung von Bildobjekten mit vielen visuell unterschiedlichen Eigenschaften (z.B. sehr viele verschiedene Stile) kann zur Verringerung der Performance in Runtime führen und den Verbrauch von Speicherplatz erhöhen. Vermeiden Sie z.B. sehr viele unterschiedliche Eckenrundungen: 0 Pixel, 1 Pixel, 2 Pixel, 3 Pixel, usw.

Dynamisierung von gruppierten Objekten (RT Professional)

Bei mehrfach geschachtelten Gruppierungen (Gruppe in Gruppe, Bildbaustein in Gruppe, Gruppe in Bildbaustein, ...) können bei der Dynamisierung mit Systemfunktionen nur die Ereignisse der äußersten Gruppe und der innersten Objekte verwendet werden. Systemfunktionen, die an die Ereignisse der unterlagerten Gruppe bzw. des unterlagerten Bildbausteins projektiert sind, werden nicht ausgeführt.

Die Projektierung von gleichzeitiger Bewegungsanimation einer Objektgruppe und Objekte innerhalb dieser Gruppe kann zur Laufzeit anders ausgeführt werden, als erwartet. Überprüfen Sie die Reihenfolge der Ausführung in Runtime und passen Sie die Projektierung gegebenenfalls an.

Anzahl von Zeichen in Textfeldern, Listen und Meldetexten

Die Anzahl der nutzbaren Zeichen im Text eines Bildobjekts ist nicht konstant. Bei der Eingabe von Textdaten werden Steueranweisungen und Formatierung mitberücksichtigt und die maximale Anzahl der Zeichen verringert sich entsprechend.

Transparenz in WinCC ab V13

Transparente Grafiken können ohne Verluste in Runtime dargestellt werden. Dies gilt für alle Comfort Panels und WinCC Runtime Advanced ab der Version 13.0.
Um die Transparenz in einer Grafikanzeige oder einem Grafischen E/A-Feld zu nutzen, muss die Eigenschaft „Füllmuster“ auf "Transparent" gestellt und die Eigenschaft „Transparente Farbe verwenden“ deaktiviert werden.

Falls die Eigenschaft "Transparente Farbe verwenden" in einem Gerät ab der Version 13.0 aktiviert ist, geht die Transparenz der Grafik verloren und transparente Pixel werden schwarz

dargestellt. Nach einer Hochrüstung von bestehenden Projekten auf die Geräteversion 13.0 oder größer ist es empfohlen, die Eigenschaft "Transparente Farbe verwenden" bei den Verwendungsstellen von transparenten Grafiken zu prüfen. Falls die Grafiken nicht richtig dargestellt werden, deaktivieren Sie diese Eigenschaft.

SIMOTION PLC Webseiten im HTML-Browser anzeigen

Falls die Websiten von SIMOTION PLC im HTML-Browser nicht richtig angezeigt werden, fügen Sie nach der Webseiten-URL "/basic" ein, um die Webseiten im Basismodus anzuzeigen.

Darstellungsunterschiede zwischen der Projektierung und der Anzeige am Bediengerät

Die Darstellung des in einem Bildobjekt projektierten Texts kann sich auf dem Bediengerät aufgrund der Display-Konfiguration unterscheiden. Falls Sie in der Projektierung die Möglichkeiten der automatischen Größenanpassung nutzen, überprüfen Sie die Darstellung auf Ihrem Bediengerät in jeder Sprache.

Falls Texte, die mit der Option "Objekte an Inhalt anpassen" projektiert sind, nicht vollständig angezeigt werden können, werden sie auf dem Bediengerät leicht verkleinert. Wenn durch die Verkleinerung Texte verzerrt dargestellt werden, deaktivieren Sie die Option "Objekte an Inhalt anpassen" und erweitern Sie den Text mit zusätzlichen Leerzeichen. Optional vergrößern Sie das Objekt in der Breite oder formulieren Sie den Text kürzer um.

GRAPH-Übersicht (RT Professional)

Beim Hochrüsten auf V14 wird die Signatur der C-Funktion am Ereignis "Klicken" geändert. Der in früheren Versionen vorhandene Parameter propertyName entfällt. Funktionen, die auf diesen Parameter zugreifen, werden nach dem Hochrüsten als fehlerhaft markiert.

Dynamisierung in ProDiag-Übersicht (RT Professional)

Die Dynamisierung der Eigenschaft "Bedienung zulassen" mit einer Variablen ist im Objekt "ProDiag-Übersicht" nicht möglich.

Erstwerterfassung und Kriterienanalyse: Unterstützte Anweisungen

Für die Erstwerterfassung und Kriterienanalyse werden auch die FUP-Anweisungen "AND" und "OR" unterstützt.

Folgende Anweisungen werden bei der Erstwerterfassung und Kriterienanalyse bis einschließlich der Anweisungsbox unterstützt:

- "Flipflop setzen/rücksetzen"
- "Flipflop rücksetzen/setzen"
- Zeiten TP, TOF und TONR
- Zähler CTU, CTD, CTUD

Bildbausteine mit den UDTs vom Typ "IEC-Zeit" aktualisieren

Aus den benutzerdefinierten PLC-Datentypen (UDT), die den Datentyp "IEC-Zeit" verwenden, wurde das Element "ST" automatisch entfernt. Passen Sie die Bildbausteine, die diese UDTs beinhalten, auf eine neue Version des UDTs ohne ST-Element an.

Eine detaillierte Beschreibung dazu finden Sie im FAQ mit der Beitragsnummer "109740393" im SIMATIC Customer Online Support.

Überlagerte Controls mit Sichtbarkeitsanimationen

Die überlagerte Projektierung von tabellenbasierten Controls mit Sichtbarkeitsanimationen kann auf den Geräten mit dem Betriebssystem Windows CE/Embedded Compact zu Problemen bei der Touch-Bedienung führen.

SVG-Grafiken

Die SVG-Grafiken werden in Siemens SVG-Standard umgewandelt. Beachten Sie bei der Verwendung von SVG-Grafiken folgende Einschränkungen:

- Die CSS-Definitionen werden zu Inline-Attributen umgewandelt.
- Die eingebettete Skripte und nicht-lokale URL-Links werden in den SVG-Grafiken nicht unterstützt und aus den ursprünglichen Grafiken bei der Umwandlung entfernt.
- Die Verwendung von SVG-Grafiken mit eingebetteten Grafiken und Animationen wird nicht unterstützt.
- Die Verwendung von großen SVG-Grafiken beeinflusst die Performance durch erhöhten Zeichenaufwand.

Die Darstellung von SVG-Grafiken hängt von dem verwendeten Browser ab.

SVG Grafiken bearbeiten

Das Öffnen von SVG-Grafiken in einem externen Editor mithilfe des Befehls "Bearbeiten" ist nicht möglich.

System-Diagnoseanzeige: Zugriffsschutz "HMI-Zugriff"

Bei Verwendung der Schutzstufe "HMI-Zugriff" in Verbindung mit der SIMATIC S7-1200 PLC kann eine richtige Anzeige der Diagnosedaten erst ab Firmware-Version 4.1 oder höher sichergestellt werden.

Schriftarten in der Überschrift von Fensterobjekten

Die Schriftart, die Sie in den Runtime-Einstellungen des Geräts konfigurieren, wird in den Überschriften von Meldefenster, System-Diagnosefenster und Dialogüberschriften nicht verwendet.

Ungültige Grafiktypen in den Grafiklisten nach dem Hochrüsten (WinCC Runtime Professional)

Wenn Sie Ihr Projekt auf die TIA Portal Version V15 hochrüsten, resultieren aus ungültigen Grafiktypen der Grafikliste Fehler beim Übersetzen, die in V14 als Warnungen angezeigt wurden.

Um Fehlermeldungen beim Übersetzen zu vermeiden, müssen Sie bei Bedarf den Grafiktyp in der Grafikliste manuell korrigieren.

Mehrsprachige Grafiken in Bildbausteinen

In einem Bildbaustein fügen Sie in ein Bildobjekt eine Grafik ein. Diese Grafik wird in allen festgelegten Projektsprachen verwendet.

Wenn Sie die Grafik mithilfe des Kontextmenü-Befehls „Editieren“ mit der Option "Grafik zentralbearbeiten" öffnen, wird die Grafik nach dem Bearbeiten und Speichern in einem externen Editor, z. B. MS Paint, nur für die aktuell festgelegte Projektsprache geändert. Die Änderung der entsprechenden Grafik wird in allen Objekten sichtbar, die diese Grafik verwenden.

Verwendung von mehrsprachigen Grafiken (RT Professional)

Grafiken werden im Runtime Projekt in der Sprache abgelegt, die Sie in den Runtime-Einstellungen unter "Sprache & Schriftart > Sprachauswahl > Runtime-Sprache für nicht-multilinguale Objekte " festlegen.

Die Verwendung von unterschiedlichen Grafiken für einzelne Runtime-Sprachen hat in Runtime bei der Sprachumschaltung keine Wirkung.

Abhilfe: Projektieren Sie einzelne Grafikanzeigen mit der gewünschten Grafik in der eingestellten Runtime-Sprache für nicht-multilinguale Objekte und schalten Sie diese entsprechend der Runtime-Sprache im Skript sichtbar bzw. unsichtbar.

PLC Code Anzeige (RT Professional)

Die Schaltfläche "Gesamten Funktionsbaustein anzeigen" der PLC Code Anzeige kann nur korrekt ausgeführt werden, wenn die Namen der folgenden Objekte keine Sonderzeichen, wie z.B. ein Komma, enthalten:

- PLC Name
- DB/FB Name
- DB/FB Interface Element Name

Netzwerkänderungen in der PLC-Code-Anzeige (RT Professional)

Bei anstehender ProDiag-Meldung werden Netzwerkänderungen in der PLC-Code-Anzeige in Runtime aktualisiert.

Damit Netzwerkänderungen in der PLC-Code-Anzeige bezüglich Kriterienanalyse korrekt angezeigt werden können, muss die Meldung neu ausgelöst werden.

PDF-Anzeige (RT Advanced)

Mit der PDF-Anzeige können keine Poup-Notizen (Sticky Notes) innerhalb von PDF-Dateien angezeigt werden.

Siehe auch

http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/de/62101921)

3.4.2 Variablen und Verbindungen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Multiplexen von Variablen (Basic Panels)

Wenn Sie auf einem Basic Panel eine Variable mit einer externen Variablen multiplexen, wird in Runtime beim ersten Lesezyklus die Adresse aus der PLC gelesen. Erst beim zweiten Lesezyklus steht der Wert der gelesenen Adresse zur Verfügung.

Variablennamen in Bildbausteinen (RT Professional)

Die Verwendung von Zeichen "." und "@" in Namen von Variablen in Bildbaustein-Typen ist unzulässig. Verwenden Sie diese Sonderzeichen nicht in den Variablennamen in Bildbausteinen.

Variablen mit symbolischer Adressierung und Datentyp "Char Array" (RT Professional)

Variablen mit symbolischer Adressierung und dem Datentyp "WString" sind für die Kommunikation von RT Professional und SIMATIC S7-1200 V3 nicht freigegeben.

Längenangabe für Variablen vom Typ "String" (RT Professional)

Im Editor "Variablen" ist die Längenangabe für die internen HMI-Variablen vom Typ "String" unwirksam. Wenn Sie die Variablenlänge, z. B. für ein Eingabefeld oder Ausgabefeld, einschränken möchten, begrenzen Sie die Länge mithilfe VB-Scripting oder C-Scripting.

Variable vom Typ "String" für Verbindungen mit SIMOTION-Steuerungen (Panels, RT Advanced)

Für SIMOTION-Steuerungen darf die Länge der String-Variablen 210 Zeichen nicht überschreiten.

Es ist möglich, im Engineering System eine String-Variable mit mehr als 210 Zeichen anzulegen. Jedoch wird die betreffende Variable in Runtime nicht geschrieben. Es erscheint die Meldung, dass die Wertgrenze außerhalb der Variable liegt.

Passen Sie in diesem Fall die Länge der String-Variable an.

Variablen unter "Variable Box-ID"

Die Variable, die Sie unter "Runtime-Einstellungen > Allgemein" unter "Variable Box-ID" projektieren, darf nur im schreibgeschützten Modus verwendet werden.

In diese Variable dürfen keine Werte geschrieben werden, jedoch darf sie z. B. in Skripten gelesen oder in Bildern ausgegeben werden.

Array-Elemente in WinCC

Wenn Sie eine HMI-Variable mit einem Array aus einem STEP 7 Datenbaustein verbunden haben, das nicht mit einer Untergrenze von 0 beginnt, werden die Array-Elemente in WinCC auf die Untergrenze von 0 abgebildet.

Damit Sie beim Zugriff auf die einzelnen Arrayelemente nicht zwischen den STEP 7-Indizes und den WinCC-Indizes umdenken müssen, sollten Sie die Untergrenzen von Arrays auch in STEP 7 bei 0 beginnen lassen.

Verwendung von 64-Bit Datentypen

Bei der Verwendung von 64-Bit Datentypen kann es zu einem geringen Genauigkeitsverlust kommen, da diese Datentypen im HMI-Kanal auf den Datentyp Double abgebildet werden. Somit kann es vorkommen, dass ganzzahlige Datentypen mit Nachkommastellen angezeigt werden.

Systemdiagnose von PLCs im R/H-System im nicht-redundanten Betrieb

Zwei PLCs und zwei entsprechende Verbindungen sind in einem R/H-System projektiert. Wenn das R/H-System nicht im redundanten Betrieb ist, kann es vorkommen, dass die Systemdiagnose-Anzeigen der beiden Verbindungen für die gleiche PLC unterschiedliche Diagnosestatus-Symbole anzeigen.

3.4.3 Meldesystem und Meldeanzeigen

Inhalt

Hinweise, die nicht mehr in der Onlinehilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Boolesche Variablen in Meldearchiven

Variablen vom Typ Bool werden im Meldearchiv als 0 und -1 erkannt. Wenn Sie in einem Meldearchiv eine Textliste verwenden, die von einer Variable vom Typ Bool gesteuert wird, dann ergänzen Sie die entsprechende Textliste um den Eintrag für den Wert -1.

PLC-Meldungen

Eine PLC-Meldung ist nur dann im Editor "Meldungen" eines HMI-Gerätes sichtbar, wenn auch ein Instanz-Datenbaustein für diesen FB existiert.

Steuerungsmeldungen in Runtime (WinCC Professional)

Wenn Sie ein Projekt von den Versionen V13SP1 oder V13 SP2 auf die Versionen V14, V14 SP1, V15.x hochrüsten, müssen Sie das Projekt übersetzen und in die Steuerung und das Bediengerät laden. Ansonsten werden die Steuerungsmeldungen in der Meldeanzeige durchgestrichen dargestellt.

Steuerungsmeldungen in Runtime (Panels und RT Advanced)

Wenn für eine Steuerungsmeldung die Eigenschaft "Meldung > Grundeinstellungen > Nur Information" aktiviert wurde, wird diese Steuerungsmeldung in der Meldeanzeige in WinCC nicht angezeigt.

"Automatische Aktualisierung" von Steuerungsmeldungen am Bediengerät aktiviert (Panels und RT Advanced)

Für Steuerungsmeldungen, die im Meldearchiv angezeigt werden, gilt Folgendes:

Die Infotexte der Meldungen werden nicht archiviert und zeigen bei der Anzeige eines Meldearchivs in Runtime "###Text missing###".

Kopieren von Projekten mit integrierten Verbindungen

Wenn Sie aus einem Projekt die Geräte mit ihren integrierten Verbindungen (zwischen Bediengeräten und Steuerungen) in eine globale Bibliothek kopieren und von dort in ein neues Projekt einfügen, dann können die Informationen zu Steuerungsmeldungen am Bediengerät verloren gehen. Um diesen Informationsverlust zu vermeiden, kopieren Sie zuerst alle Steuerungen und dann erst alle Bediengeräte in das neue Projekt

Meldung 20013 zu Globalen Skripten (Panels und RT Advanced)

Anders als in der Hilfe beschrieben, wird die Meldung mit der Nummer 20013 nur beim Runtime-Start ausgegeben, wenn die Skript-Engine noch nicht initialisiert wurde.

Diese Meldung hat auf die Funktionsweise der VB-Skripte in Runtime keine Auswirkung.

Archivierung von Meldetexten mit mehr als 6 Werten (Panels und RT Advanced)

Wenn ein Meldetext mehr als 6 Werte enthält, werden nach dem Archivieren nur die ersten 6 Werte in der Archivdatei gelistet.

Eine Meldeanzeige, die für die Anzeige der Archive konfiguriert ist, zeigt alle im Meldetext enthaltenen Werte.

3.4.4 Systemfunktionen und Skripte

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Skripte bei Service-Projekten (RT Professional)

Da in Service-Projekten normalerweise kein interaktiver Benutzer angemeldet ist, führen C-Skripte und VB-Skripte z. B. in folgenden Fällen zu Problemen:

- Wenn die Skripte Interaktionen, z. B. Eingaben erfordern.
- Wenn die Skripte Message-Boxen anzeigen.

Im Service Mode besteht kein gemeinsamer Datenbereich für C-Scripting. Daher können beispielsweise keine globalen C-Variablen zwischen "Aufgabenplaner" und "Bildern" ausgetauscht werden.

Triggervariablen von anwenderdefinierten Funktionen (RT Professional)

Wenn Sie an einem Bild einen HMI- oder einen PLC-Anwenderdatentyp projektiert haben, können Sie in neu erstellten oder geänderten benutzerdefinierten Anwenderfunktionen die Elemente des Anwenderdatentyps nicht als Triggervariable einsetzen.

Suchen und Ersetzen in lokalen Skripten (RT Professional)

Sie suchen und ersetzen Texte innerhalb von Editoren mithilfe der Funktion "Suchen und ersetzen". Suchen und ersetzen von Texten in lokalen Skripten ist nicht möglich.

Anwenderdefinierte Funktionen in Bildbausteinen

Wenn Sie in einem Bildbaustein mit einer anwenderdefinierten Funktion auf Variablen zugreifen wollen, können Sie den Namen der Variablen nicht dynamisch zusammensetzen.

Systemfunktion "SimuliereVariable" (Basic Panels)

Wenn Sie auf einem Basic Panel die Systemfunktion "SimuliereVariable" in Verbindung mit einer kleinen Zykluszeit verwenden, kann das Bediengerät überlastet werden.

Systemfunktionen der Bitbearbeitung (Mobile Panels und RT Advanced)

Es ist nicht möglich, aufgrund von Ereignissen Bitarrays zu manipulieren. Systemfunktionen wie z.B. InvertBit, SetBit usw. dürfen keine Elemente eines Bitarrays als Argument enthalten.

Nicht weiterverwendete Eigenschaften (Comfort Panels und RT Advanced)

In WinCC V13 wurden viele neue visuelle Eigenschaften eingebracht und viele bestehende Eigenschaften harmonisiert.

Im Zuge der Überarbeitung der Objekteigenschaften kann die Verwendung mancher dieser Eigenschaften, die in früheren Versionen von WinCC in VB Skripten nutzbar waren, in WinCC V13 nicht mehr unterstützt werden. Skripte, die nicht mehr unterstützte Eigenschaften nutzen, bleiben gültig, die betreffenden Aufrufe innerhalb der Skripte sind jedoch zur Laufzeit ohne Funktion. Diese Einschränkungen betreffen ausschließlich Comfort Panels und RT Advanced in der Geräteversion V13.0.

Die folgende Tabelle zeigt die nicht weiterverwendeten Eigenschaften der Bildobjekte:

Eigenschaft	Bildobjekt
BorderBrightColor3D	Schieberegler
BorderColor	Textfeld, E/A-Feld, Symbolisches E/A-Feld, Datum/Uhrzeit-Feld
BorderInnerWidth3D	Schieberegler
BorderOuterWidth3D	Schieberegler
BorderShadeColor3D	Schieberegler
BorderStyle3D	Textfeld, E/A-Feld, Schaltfläche, Symbolisches E/A-Feld, Grafisches E/A-Feld, Datum/Uhrzeit-Feld, Balken, Schalter
BorderWidth	Textfeld, Symbolisches E/A-Feld
CenterColor	Zeigerinstrument
DialColor	Zeigerinstrument
EdgeStyle	Textfeld, E/A-Feld

Debuggen von VB-Skripten

Mit dem Objekt "Debug" geben Sie benutzerdefinierte Texte in einem Debugger aus. Das Objekt besitzt die Methoden "Write" und "WriteLine". Weiterführende Informationen zum Objekt "Debug" finden Sie im Internet (https://msdn.microsoft.com/en-us/library/wk3y866c(v=vs.84).aspx).

Mit dem Objekt "Err" reagieren Sie auf Fehlermeldungen, die von einem anderen Objekt erzeugt werden. Das Objekt besitzt die Methoden "Raise" und "Clear".

Weiterführende Informationen zum Objekt "Err" finden Sie im Internet (https://msdn.microsoft.com/en-us/library/sbf5ze0e(v=vs.84).aspx).

Übergabe von Parametern an VB-Skripte (Panels und RT Advanced)

In WinCC können Sie definieren, ob die Parameter von VB-Skripten "ByRef" (Adresse des Parameters) oder "ByVal" (Wert des Parameters) übergeben werden.

Eine Übergabe der Adresse des Parameters mithilfe von "ByRef" (Call by Reference) ist nur für die internen Variablen des VB-Skripts möglich. Bei HMI-Variablen wird immer der Wert des Parameters übergeben, unabhängig davon ob der Parameter als „ByRef“ oder als „ByVal“ definiert wurde.

Systemfunktionen mit Parameter "Bearbeitungsstatus"

Bei Systemfunktionen, wie z. B. "SaveDataRecord", können für den Parameter "Bearbeitungsstatus" nur HMI-Variablen angegeben werden.

Funktionen der PLC Code Anzeige (RT Professional)

Die Funktion der PLC Code Anzeige ""ShowCodeViewerFromAlarm" können nur korrekt ausgeführt werden, wenn die Namen der folgenden Objekte keine Sonderzeichen, wie z.B. ein Komma, enthalten:

- PLC Name
- DB/FB Name
- DB/FB Interface Element Name

Zugriff in Runtime für die Eigenschaften "Height" und "Width" einiger VBS-Objekte

Anders als in der Hilfe beschrieben, haben Sie bei folgenden VBS-Objekten auf die Eigenschaften "Height" und "Width" lesenden Zugriff:

AlarmView, Bar, BatteryView, Button, CameraControl, Clock, DateTimeField, FunctionTrendControl, Gauge, GraphicIOField, GraphicView, HTMLBrowser, IOField, Line, MediaPlayer, PDFView, PLCCodeViewer, Polygon, Polyline, ProDiagOverview, ProtectedAreaNameView, RangeLabelView, RangeQualityView, RecipeView, S7GraphOverview, Slider, SmartClientView, StatusForce, Switch, SymbolicIOField, SymbolLibrary, SysDiagControl, TextField, TrendView, UserView, WLanQualityView, ZoneLabelView, ZoneQualityView

Bei folgenden Objekten habe Sie für Runtime Advanced auf die Eigenschaft "BorderColor" keinen Zugriff:

TextField

Weiterführende Informationen zum VBS-Objektmodell finden Sie finden Sie im FAQ mit der Beitragsnummer "53752382" im SIMATIC Customer Online Support:

https://support.industry.siemens.com/cs/ww/de/view/53752382 (https://support.industry.siemens.com/cs/ww/de/view/53752382)

3.4.5 Protokolle

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Drucken von Protokollen (Panels)

Abhängig vom verwendeten Drucker können die in den Protokollen am Rand platzierten Objekte beim Drucken abgeschnitten werden.

Vergewissern Sie sich, dass Ihr Protokoll eine passende Größe hat.

3.4.6 Rezepturen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Arrays in Rezepturelementen

Wenn Sie an Rezepturelementen einer Rezeptur sowohl ein Array als auch Elemente dieses Arrays projektiert haben, wird das Laden mit Datensätzen mit folgender Fehlermeldung abgebrochen: "290055: Import der Datensätze mit Fehler abgebrochen"

Verwenden Sie für Rezepturelemente der gleichen Rezeptur entweder nur die Arrays oder nur die Arrayelemente.

3.4.7 Benutzerverwaltung

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

3.4.8 Kommunikation

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verbindungsaufbau mit WinCC Runtime Advanced

Bitte beachten Sie, dass die im Engineering System projektierte Geräteversion mit der Version der installierten Runtime übereinstimmt, da ansonsten die Kommunikation beim Laden des Projektes nicht aufgebaut werden kann oder verzögert erfolgt.

Wenn die Geräteversion nicht mit der Version der installierten Runtime übereinstimmt, haben sie folgende Möglichkeiten:

1. Ändern Sie die im Engineering System projektierte Geräteversion.
2. Installieren sie die zur projektierten Geräteversion passende Version von WinCC Runtime Advanced.

Verbindungswechsel bei nicht-integrierten Verbindungen

Wenn Sie die Kommunikation von einer nicht-integrierten Verbindung auf eine andere nciht-integrierte Verbindung umstellen wollen, z. B. von Omron zu Mitsubishi, müssen Sie sicherstellen, dass die Datentypen der Variablen auch in der neuen Verbindung unterstützt werden.

Verbindungsunterbrechungen bei Mitsubishi Steuerungen

Nach mehrfachen Verbindungsunterbrechungen kann es zu der Situation kommen, dass in der Mitsubishi-Steuerung alle Verbindungsressourcen belegt sind und die Verbindung nicht mehr aufgebaut werden kann. Es wird empfohlen, im SPS-Programm der Steuerung diese Verbindungsressourcen zu kontrollieren und sie auch wieder freizugeben.

Selbst signierte Zertifikate (RT Professional)

Selbst signierte Zertifikate werden nicht mehr unterstützt.

Wenn Server-seitig ein selbst signiertes Zertifikat erkannt wird, findet die Authentifizierung über den zuvor verteilten PSK-Schlüssel (Pre-shared key) statt. Wenn der PSK-Schlüssel zwischen den Kommunikationspartnern nicht übereinstimmt, kommt keine Kommunikation zustande.

Kommunikation zwischen WebClient und Sm@rtServer

Java wird von zukünftigen Browserversionen nicht mehr unterstützt.

In diesen Fällen nutzen Sie die folgenden Alternativen als Sm@rtClient:

- Sm@rtClient.exe
- Sm@rtClient App für Smartphone

Genauigkeit des Datentyps "DTL"

Der Datentyp "DTL" unterstützt Zeitangaben bis im Bereich von Nanosekunden. Da Panels Zeitangaben lediglich bis in den Bereich von Millisekunden unterstützen, ergeben sich bei der Verwendung an den Bereichszeigern folgende Einschränkungen:

- Bereichszeiger "Datum/Uhrzeit"
  Bei der Übertragung der Zeitangaben vom einem Panel zur Steuerung gilt als kleinste Zeiteinheit 1 Millisekunde. Der Wertebereich von Mikrosekunden bis Nanosekunden des Datentyps "DTL" wird mit Nullen befüllt.
- Bereichszeiger "Datum/Uhrzeit Steuerung"
  Bei der Übertragung der Zeitangaben von einer Steuerung zu einem Panel wird der Bereich von Mikrosekunden bis Nanosekunden ignoriert. Auf dem Panel wird die Zeitangabe bis einschließlich der Millisekunden weiter verarbeitet.

Beschränkte Anzahl von möglichen HMI-Verbindungen

Beim Übersetzen eines Geräts tritt eine Fehlermeldung auf, dass die Konfiguration der HMI-Verbindung im Editor "Geräte & Netze" ungültig ist. Die Ursache kann darin liegen, dass die maximale Anzahl der möglichen Verbindungen des HMI-Geräts bzw. der PLC überschritten ist.

Prüfen Sie die maximale Anzahl der möglichen Verbindungen. Informieren Sie sich bitte im Gerätehandbuch der von Ihnen verwendeten Geräte.

Verwendung von PROFINET IO bei Panel-Bediengeräten

Bei der Verwendung von PROFINET IO zur Verbindung der Direkttasten und LEDs von Bediengeräten mit der Steuerung können Sie bei der Projektierung in HW Konfig einen Offset für den Adressbereich der Eingänge und der Ausgänge festlegen.

Bei der Verwendung einer PROFINET IO-fähigen CPU der 400er Baureihe in Verbindung mit einem der unten aufgeführten Bediengeräte gilt folgende Einschränkung:

Der Offset für den Anfang des Adressbereichs der Eingänge darf nicht größer sein als der Offset für den Anfang des Adressbereichs der Ausgänge.

Die Einschränkung gilt für folgende Bediengeräte:

- Mobile Panel 177

Für die Projektierung der Adressparameter öffnen Sie die Steuerung mit der CPU der 400er Baureihe in HW Konfig. Selektieren Sie im Stationsfenster von HW Konfig das über PROFINET IO verbundene Bediengerät. Im unteren Bereich des Stationsfensters wird in der Detailansicht eine Tabelle mit den Eigenschaften des Bediengeräts angezeigt. Selektieren Sie in der Tabelle die Zeile mit den Adressen des Bediengeräts und öffnen Sie die Objekteigenschaften über das Kontextmenü.

Aktivieren Sie im Dialog "Objekteigenschaften" die Registerkarte "Adressen". Unter "Eingänge > Anfang" projektieren Sie den Offset für die Eingänge. Unter "Ausgänge > Anfang" projektieren Sie den Offset für die Ausgänge.

Kommunikation der RT Advanced über den Station Manager (SIMATIC NET) mit einer S7-1200

Die Kommunikation von einer SIMATIC S7-1200 mit einem PC mit WinCC RT Advanced über einen Router gelten folgende Einschränkungen für den PC:

- Windows 7: Nur mit installiertem SIMATIC NET 8.1

Diese Einschränkungen gelten auch, wenn Sie den Station Manager verwenden. Verbindungen mit Hilfe des Station Managers von der Runtime Advanced werden immer wie geroutete Verbindungen behandelt.

IP-Einstellungen und Gerätename einer Steuerung im Control Panel des Bediengeräts ändern (Basic Panels)

Auf dem Bediengerät ist das Control Panel im Menü "Service and Commissoning > IP-Adaptation" geöffnet. Wenn Sie die IP Einstellungen oder den Gerätenamen einer Steuerung ändern wollen, beachten Sie Folgendes:

Im Engineering System haben Sie zuvor im Inspektorfenster der Steuerung unter "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen" folgende Optionen aktiviert:

- "IP-Adresse auf anderem Weg beziehen"
- "PROFINET-Gerätenamen auf anderem Weg beziehen"

HMI-Verbindungen zu Steuerungen SIMATIC S7-1200

HMI-Verbindungen zu Steuerungen SIMATIC S7-1200 mit der Firmware-Version kleiner als V2.0 sind nicht möglich.

"Setzen der IP-Suite (Adresse) der PLC im Control Panel" bei SIMATIC S7-1200 V1 (Basic Panels)

Die Funktion "Setzen der IP Suite (Adresse) der PLC im Control Panel" ist für folgende Steuerungen nicht freigegeben:

- SIMATIC S7-1200 V1

Verbindungen über PROFIBUS DP

Wenn eine Verbindung über PROFIBUS DP zwischen einer Steuerung und einem Bediengerät unterbrochen und dann wieder aufgebaut wird, dann werden sporadisch alle weiteren PROFIBUS DP-Verbindungen im Kommunikationsnetz abgebrochen und wieder aufgebaut.

Schalten Sie den getrennten Teilnehmer spannungslos, bevor Sie den Teilnehmer wieder verbinden.

Wechsel einer Verbindung

Bei einem Wechsel der Verbindung von einem Bediengerät zu einer Steuerung SIMATIC S7-300/400, zu einer Steuerung SIMATIC S7-1500 oder SIMATIC S7-1200 kann es zum Abbruch der Verbindung kommen.

Beachten Sie folgende Einstellungen in den Steuerungen SIMATIC S7-1500 oder SIMATIC S7-1200:

- Absolute Adressierung der Variablen
- Option "PUT/GET-Kommunikation deaktivieren" muss ausgewählt sein
- Schutzstufe "Kompletter Schutz" darf nicht projektiert sein

Kommunikation der Runtime Professional mit SIMATIC S7-1200

Im Produktivbetrieb ist die Kommunikation von WinCC Runtime Professional mit SIMATIC S7-1200 nur für Einplatzsysteme freigegeben.

RT Advanced im Station Manager

Wenn RT Advanced und WinAC RTX denselben Kommunikationsprozessor eines PCs verwenden, dann ist eine HMI-Kommunikation zu SIMATIC S7-1200 und SIMATIC S7-1500 nicht möglich.

Rohdatenkommunikation in redundanten Projekten (RT Professional)

Simatic.NET, Named Connections und diverse Kommunikationsbausteine, wie z.B. BSEND/BRCV, können bei einer redundant konfigurierten PC-Station nur eingeschränkt verwendet werden, weil die Verbindungsparameter für den redundanten Partnerserver nicht projektiert werden können.

Nicht integrierte Verbindung zu einem SIMATIC S7-1500 Software Controller (RT Advanced)

Eine nicht integrierte Verbindung zwischen einem HMI-Gerät und einem SIMATIC S7-1500 Software Controller wird in WinCC nicht unterstützt.

Transferbereiche für die Betriebsart IO-Device und DP-Slave der HMI-Geräte

Wenn Sie für HMI-Bediengeräte die Betriebsart "IO-Device" bzw. "DP-Slave" aktiviert haben, sollten in den Eigenschaften des Bediengeräts keine Transferbereiche hinzugefügt oder gelöscht werden. Wenn Sie versehentlich einen Transferbereich gelöscht oder hinzugefügt haben, so entfernen und verbinden Sie den Controller erneut.

Ermittlung von Diagnosedaten durch HMI-Geräte

Entgegen der Beschreibung der Diagnosesymbole für Baugruppen und Geräte im Informationssystem hat für HMI-Geräte das folgende Symbol die hier angegebene Bedeutung:

Symbol	Bedeutung
	Es sind keine Diagnosedaten verfügbar.

PROFIEnergy-Kommunikation

Um die PROFIEnergy-Kommunikation einzurichten, kontaktieren Sie den Customer Support.

3.5 Übergreifende Funktionen

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Systemdiagnose im Geräte-Proxy einsetzen

Um die Funktion Systemdiagnose in einem IPE Geräte-Proxy verwenden zu können, z. B. eine Systemdiagnose-Anzeige, fügen Sie die PLC-Meldungen als Inhalt eines Geräte-Proxys ein.

Geräte-Proxy mit Daten aus einem Projekt der Vorgängerversion initialisieren

In einem Projekt mit der Service Pack Version kann ein Geräte-Proxy nicht mit den Daten aus einem Projekt der zugehörigen Vollversion initialisiert werden.
Rüsten Sie das Quellprojekt auf die Service Pack Version hoch, um den Geräte-Proxy im Zielprojekt mit den Daten aus dem Quellprojekt initialisieren zu können.

Import und Export von Bibliothekstexten

Folgende Objekte werden beim Import und Export von Bibliothekstexten nicht unterstützt:

- Import: Rezepturen, Rezepturelemente, Rezepturdatensätze, HMI-Meldungen, Textlisteneinträge, Bilder, Protokolle.
- Export: Textlisteneinträge, Bilder, Protokolle.
- Import und Export von Bibliothekstypen "Bild".

3.6 Übersetzen und Laden

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Übersetzen und Laden

Wenn beim Übersetzen interne Fehler oder Warnungen auftreten, übersetzen Sie das Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.

Bevor Sie mit Ihrem Projekt in den Produktivbetrieb gehen, übersetzen Sie das Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem Bediengerät auch alle geänderten Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

Fehlerhafte Installation von ProSave

Wenn Sie beim Laden der Daten in ein Zielgerät oder bei der Bediengerätewartung eine Fehlermeldung bezüglich der Installation von ProSave erhalten, können Sie den Fehler nicht mit der Reparaturfunktion des Setup beheben. Deinstallieren Sie ProSave über die Systemsteuerung. Starten Sie anschließend das Setup und installieren Sie die Komponente "ProSave" erneut.

Prüfen der Adressparameter

Beim Übersetzen eines Bediengerätes in der Projektnavigation über den Befehl "Übersetzen > Hardware und Software (nur Änderungen)" im Kontextmenü werden die Adressparameter des Bediengeräts, wie z. B. die IP-Adresse, nicht überprüft. Um sicherzugehen, dass auch die Adressparameter überprüft werden, müssen Sie das Bediengerät im Editor "Geräte & Netze" über die Schaltfläche "Übersetzen" in der Symbolleiste übersetzen.

Fehlermeldung beim Laden von Daten auf die PLC

Ein Panel und eine PLC sind verbunden und kommunizieren miteinander.

Wenn beim Laden von Daten auf die PLC vom Panel aus auf eine Variable zugegriffen wird, wird am Panel eine Fehlermeldung ausgegeben.

Verzögerte Reaktion im Dialog "Erweitertes Laden"

Wenn im Dialog "Erweitertes Laden" die Einstellungen für "Typ der PG/PC Schnittstelle" und "PG/PC-Schnittstelle" nicht mit den Einstellungen auf dem Bediengerät übereinstimmen, dann kann dies dazu führen, dass die Applikation bis zu eine Minute nicht reagiert.

Erweitertes Laden bei einer S7-1200 und einem Comfort Panel

Eine PLC S7-1200 und ein Comfort Panel befinden sich im gleichen physikalischen Netz wie der PG/PC. Sie öffnen den Dialog "Erweitertes Laden" für das Comfort Panel.

Wenn Sie die Option "Alle erreichbaren Teilnehmer anzeigen" aktivieren, kann es vorkommen, dass die Applikation nicht mehr reagiert.

Projekttransfer über USB

Wenn Sie mehr als ein Bediengerät an ihren Projektierungs-PC über USB angeschlossen haben, dann ist ein Projekttransfer nur auf das zuletzt angeschlossene Bediengerät möglich.

Comfort Panels ab Geräteversion 13.0: Sicherung der Daten beim Laden von Projekten

Bei Comfort Panels mit der Geräteversion 13.0 oder höher sichert WinCC beim Abbruch der Übertragung automatisch, dass keine Daten verloren gehen und dass die vorhandenen Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Anzeige von Zeichen in Transfer-Meldungen

Falls beim Transfer auf Bediengeräte mit der Geräteversion V12 oder älter die Zeichen in den Transfer-Meldungen nicht richtig dargestellt werden, überprüfen Sie das Gebietsschema in Windows. Stellen Sie die entsprechende Sprache unter "Sprache für Unicode-inkompatible Programme" ein.

Fehlende Schriftarten (Panels)

Folgende Schriftarten gehören unter Windows 10 nicht mehr zum Standard-Installationspaket und sind nicht vorinstalliert:

- MS PGothic (wird standardmäßig für die Darstellung der japanischen Zeichen verwendet)
- Gulim (wird standardmäßig für die Darstellung der koreanischen Zeichen verwendet)
- MingLiU für Chinesisch (wird standardmäßig für die Darstellung der traditionellen chinesischen Zeichen für Taiwan, Hong Kong und Macao verwendet)

Wenn eine Schriftart nicht installiert ist, wird im Engineering System eine Fehlermeldung ausgegeben. Die Fehlermeldung erscheint in jeder TIA Portal-Session, wenn Sie zu "Runtime-Einstellungen > Sprache und Schriftart" navigieren oder Bildobjekte öffnen, die die fehlende Schriftart verwenden. Die fehlende Schriftart wird vom TIA Portal durch eine geeignete Schriftart ersetzt.

Fehlende Schriftarten verursachen Probleme beim Übersetzen und in Runtime.

Wenn Sie, zum Beispiel, ein fehlerfreies V13 SP1-Projekt, das auf einem Projektierungs-PC mit Windows 7 erstellt wurde, auf einem Projektierungs-PC der V14 SP1 mit dem Betriebssystem Windows 10 öffnen, können beim Übersetzen Fehlermeldungen wegen der nicht installierten Schriftarten ausgegeben werden.

Weiterführende Informationen zu Schriftarten in Windows 10 finden sie unter https://support.microsoft.com/en-us/kb/3083806 (https://support.microsoft.com/en-us/kb/3083806)

Abhilfe

1. Übersetzen Sie das Projekt, bevor Sie es auf Ihr Bediengerät übertragen. So stellen Sie fest, ob Ihnen die benötigten Schriftarten fehlen.
2. Installieren Sie diese Schriftarten bzw. das entsprechende Windows-Sprachpaket selbst.

Alternativ können Sie das Projekt auch mit Ersatzschriften übersetzen.

Die fehlende Schriftart ist in der Projektierung gespeichert. Sobald diese Schriftart wieder verfügbar ist, wird sie angezeigt und in der Projektierung verwendet.

Projektgröße verringern

Beim Übersetzen Ihres Bediengeräts weist eine Meldung Sie darauf hin, dass die Größe Ihres Projekts für das entsprechende Bediengerät an die Systemgrenze kommt. In diesem Fall kann das komplette Übersetzen der Software die Projektgröße verkleinern. Führen Sie dafür den Befehl "Übersetzen > Software (komplett übersetzen)" aus.

3.7 Runtime

3.7.1 Hinweise zum Betrieb in Runtime

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Fokus in Runtime

Wenn Sie für ein Bediengerät mit der Geräteversion 12.0.0 oder kleiner eine kontrastarme Kombination von Fokus-Farbe und Rahmenfarbe projektiert haben, kann es nach einem Wechsel der Geräteversion im TIA Portal anschließend in Runtime vorkommen, dass der Fokus nicht mehr erkennbar ist. Ändern Sie eine der beiden Farben.

Sprachenverhalten - Layout der Bildschirmtastatur

Das Layout der Bildschirmtastatur wird nicht umgeschaltet, wenn Sie zu einer Runtime-Sprache wechseln, die unter Windows für das Tastatur-Layout nicht installiert ist.

In diesem Fall bleibt die Spracheinstellung der Tastatur auf der letzten gültigen Sprache eingestellt bzw. es wird die eingestellte Sprache des Standard-Tastatur-Layouts von Windows verwendet.

Variablenwerte überschreiten die maximale Länge

Sie geben eine Zeichenkette in eine String Variable über ein E/A-Feld ein. Wenn die Zeichenfolge die projektierte Länge der Variablen überschreitet, wird die Zeichenkette auf die projektierte Länge gekürzt.

Leere Meldetexte

Die Runtime läuft mit einem Projekt. Das Projekt ist auf einem Netzlaufwerk gespeichert.

Bei Unterbrechungen zum Netzlaufwerk kommt es gegebenenfalls vor, dass die Runtime versucht Meldetexte vom Netzlaufwerk zu laden.

Wenn die Verbindung getrennt ist, bleibt das Meldefenster bzw. die Meldeanzeige leer.

Um das Verhalten zu vermeiden, kopieren Sie das Projekt auf ein lokales Laufwerk bevor Sie das Projekt in der Runtime starten.

Dauer der Initialisierung der Archive (Panels, RT Advanced)

Initialisierung der Archive auf manchen Speichermedien kann bis zu 5 Minuten dauern. Der erfolgreiche Abschluss der Initialisierung wird durch eine Systemmeldung bestätigt, sobald diese abgeschlossen wurde. Bei einem eventuell fehlendem Speichermedium für die Archivierung kann beim Runtime-Start das Anzeigen der Systemmeldung ebenfalls bis zu 5 min dauern.

Große Archive verlangsamen das Beenden der Runtime (Basic Panels 2nd Generation)

Bei der Verwendung von sehr großen Archiven kann das Beenden der Runtime längere Zeit in Anspruch nehmen. Verwenden Sie segmentierte Archive als Alternative zu sehr großen Umlaufarchiven.

Langsame Reaktion von SmartServer

Unter Windows 7 starten und reagieren folgende Programme möglicherweise sehr langsam:

- HMI TouchInputPC
- SmartServer: Tastenkombination <Strg+Alt+Del> beim Anmeldedialog

Die Verzögerung entsteht durch die Internet-Zertifikatprüfung auf Rückruf.

Abhilfe:

Auf der Produkt-DVD finden Sie unter:
Support\Windows7\CRL_Check bzw. CD_RT\ Support\Windows7\CRL_Check\
die folgenden Dateien:

- DisableCRLCheck_LocalSystem.cmd
- DisableCRLCheck_CurrentUser.cmd

1. Führen Sie die Datei "DisableCRLCheck_LocalSystem.cmd" mit Administratorrechten aus. Rufen Sie im Kontextmenü der Datei den Befehl "Run as administrator" auf.
2. Starten Sie den PC neu.

Wenn das Problem damit nicht behoben ist, gehen Sie wie folgt vor:

1. Doppelklicken Sie auf die Datei und führen Sie die Datei "DisableCRLCheck_CurrentUser.cmd" mit Benutzerrechten aus.
2. Starten Sie den PC neu

Hinweis

Die Zertifikatprüfung auf Rückruf wird benutzer- oder PC-weit deaktiviert. Um den Ausgangszustand wiederherzustellen, führen Sie die folgenden Dateien aus:

- RestoreDefaults_LocalSystem.cmd
- RestoreDefaults_CurrentUser.cmd

Die Dateien finden Sie im folgenden Verzeichnis der Produkt-DVD:

- Support\Windows7\CRL_Check bzw. CD_RT\Support\Windows7\CRL_Check\

Bildschirmschoner auf dem Sm@rtServer beenden

Wenn der Bildschirmschoner auf dem Sm@rtServer am Server-Bediengerät aktiv ist, benötigen Sie einen schreibenden Zugriff auf der Sm@rtClient-Seite, um den Bildschirmschoner am Server-Bediengerät zu beenden.

Beschädigte Dateien bei Stromausfall vermeiden

Wenn bei Windows Systemen ein Stromausfall auftritt, während das WinCC System aktiv ist, können Dateien beschädigt oder zerstört werden. Der Betrieb mit dem Dateisystem NTFS bietet höhere Sicherheit.

Einen gesicherten Dauerbetrieb gewährleistet nur der Einsatz einer unterbrechungsfreien Stromversorgung (USV).

3.7.2 Hinweise zum Betrieb von Panels in Runtime

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Verwendung des Mausrades in Runtime

Auf allen Panels wird die Verwendung des Mausrades in Runtime nicht unterstützt.

Backup auf der Speicherkarte der PLC (Basic Panels)

Sie erstellen die Backup-Datei "A.psb" auf der Speicherkarte der PLC. Beim Erstellen des Backups tritt ein Fehler auf z. B. ein Verbindungsabbruch.
Dadurch entsteht eine fehlerhafte Datei auf der Speicherkarte der PLC. Eine solche Datei hat als Präfix "~$".
Löschen Sie die Datei mit dem Präfix "~$", wenn Sie ein Backup unter dem gleichen Namen "A.psb" erneut speichern wollen.

Panel Data Storage und S7-1500F (Basic Panels)

Die Funktion "Panel Data Storage" PDS kann auf Basic Panels im Zusammenspiel mit S7-1500F nicht verwendet werden, wenn das Kennwort für die Schutzstufe "Full access incl. fail-safe" verwendet wird.

Funktion "Panel Data Storage" (Basic Panels)

Die Funktion "Panel Data Storage" (PDS), die Ihnen auf Basic Panels zur Verfügung steht, unterstützt nur SIMATIC S7-1200 ab Firmware V4.0 und SIMATIC S7-1500. Für die PDS-Funktion muss das Panel direkt mit der CPU verbunden sein und darf nicht über den CP verbunden sein.

Schriftarten mit Dateikennung ".ttx"

Die Schriftarten mit der Dateikennung ".ttx" sind auf allen Panels und in Runtime Advanced verfügbar und können auf allen Geräten richtig dargestellt werden, z. B. die Schriftart "WinCC_flexible_smart".

Sprachen in Runtime (Comfort Panels)

Wenn auf dem Bediengerät die unter "Region und Sprache" eingestellte Sprache in Runtime nicht unterstützt wird, werden alle Dialoge in der Standardsprache "Englisch" dargestellt.

Massendatentransfer über USB

Wenn Sie auf dem Projektierungs-PC Windows 10 installieren, ist der Transferkanal "USB" nicht verfügbar.

Diese Einschränkung gilt für die folgenden Geräte und Tools:

- Basic Panels: KP300 Basic, KP400 Basic, TP1500 Basic
- Mobile Panels: 170er Serie und 270er Serie
- WinCC Engineering System
- Service-Tool "ProSave"

Hinweis

Der Transferkanal "USB" bleibt verfügbar, wenn auf dem Projektierungs-PC Windows 7 installiert sind.

Abhilfe

Verwenden Sie andere Transferkanäle für die Übertragung der Projektdaten, z. B. den Kanal "Industrial Ethernet" oder "PROFIBUS DP".

Ändern der IP-Adresse und des Gerätenamens bei Panels

Auf Comfort Panels kann die IP-Adresse oder der Gerätename auch geändert werden, wenn Runtime gestartet ist.

Auf Basic Panels ist das Ändern der IP-Adresse oder des Gerätenamens bei laufender Runtime nicht möglich.

3.7.3 Hinweise zum Betrieb von Runtime Advanced

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Runtime starten

Im TIA Portal V15 kann nur WinCC Runtime V15 gestartet werden. WinCC Runtime V11.02, V12, V13, V13 SP1, V14 und V14 SP1 können im TIA Portal V14 SP1 simuliert werden.

Bildschirmschoner auf Rechnern mit Windows 10

Auf einem Rechner mit Windows 10 wird ein aktivierter Bildschirmschoner durch das Auftreten einer Meldung der Klasse "Fehler" nicht mehr beendet.

Windows 10 im Tabletmodus

Es wird nicht empfohlen, Windows 10 im Tabletmodus zu nutzen.

Wenn Sie unter Windows 10 im Tabletmodus Anwendungen, z. B. Runtime Advanced, Sinumerik Operate usw. betreiben, werden diese Anwendungen nach dem Starten minimiert.

Berechtigung für den Start der Runtime (RT Advanced)

Auf einem Rechner mit der 32-Bit-Version von Windows 7 kann WinCC Runtime Advanced nur gestartet werden, wenn ein Benutzer der automatisch angelegten Gruppe "Siemens TIA Engineer" zugeordnet ist.

.Net-Controls in Runtime

Wenn Sie in Ihrem Projekt ein .Net-Control als "Spezifisches .Net-Control" eingebunden haben, müssen Sie die zu diesen Controls gehörenden Dateien in das Installations-Verzeichnis der WinCC Runtime, z. B. "C:\ProgramFiles\Siemens\Automation\WinCC RT Advanced", kopieren. Ansonsten kann das Control in Runtime nicht geladen werden.

Automatische Überprüfung der Software-Aktualisierungen deaktivieren

Wenn das Engineering System zusammen mit der Runtime auf einem PC installiert ist, bekommt der Bediener Benachrichtigungen über die Software-Aktualisierungen. Damit das System auf einem Mehrplatzsystem stabil läuft, muss auf allen Rechner dieselbe Software-Version installiert sein.

Es ist möglich, die automatische Überprüfung der Software-Aktualisierungen zu deaktivieren und die Performance damit zu verbessern.

Sie deaktivieren die automatische Überprüfung der Software-Aktualisierungen unter "Einstellungen > Allgemein > Software-Aktualisierungen", indem Sie das Optionskästchen "Täglich auf Aktualisierungen überprüfen" deaktivieren.

3.7.4 Hinweise zum Betrieb von Runtime Professional

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Benutzerberechtigungen im Betriebssystem

1. Alle Benutzer müssen in die Benutzergruppe "SIMATIC HMI" aufgenommen sein. Dies gilt auch für Benutzer, die WinCC-Projekte remote öffnen wollen.
2. Der Ablageordner der Projekte muss die NTFS Berechtigungen "SIMATIC HMI" mit Vollzugriff sowie "SIMATIC HMI Viewer" mit Leserechten versehen sein. Die Berechtigungen müssen für alle untergeordnete Objekte vererbt werden.
3. Mitglieder der Windows-Benutzergruppe "SIMATIC HMI" sollten nicht gleichzeitig Mitglieder der Windows-Benutzergruppe "SQLServerMSSQLUser$<Computername> $WINCC" sein. Die Mitglieder dieser Gruppe haben Administratorrechte auf dem SQL Server. Entfernen Sie daher alle Windows-Benutzer aus dieser Gruppe, für die ein eingeschränkter Zugriff auf die WinCC-Datenbank ausreicht.

Windows 7: Tastenkombinationen sperren

Wenn Sie unter Windows 7 Tastenkombinationen sperren wollen, müssen Sie in der Verwaltung des Betriebssystems die Gruppenrichtlinien anpassen.

Eine detaillierte Beschreibung dazu finden Sie im FAQ mit der Beitragsnummer "44027453" im SIMATIC Customer Online Support:

Internet: WinCC FAQ 44027453 (http://support.automation.siemens.com/WW/view/de/44027453)

Nicht fixierte Symbolleisten

Wenn für die Windows Taskleiste die Einstellung "Immer im Vordergrund" verwendet wird, können in Runtime nicht fixierte Symbolleisten hinter die Windows Taskleiste verschoben werden. Um die Symbolleisten wieder sichtbar zu machen, gehen Sie wie folgt vor:

1. Wählen Sie im Kontextmenü der Taskleiste den Befehl "Eigenschaften".
2. Deaktivieren Sie "Taskleiste immer im Vordergrund halten".

Störung der Verbindung zwischen Server und Client

Wenn die Verbindung zwischen Server und Client gestört ist, überprüfen sie Einstellungen der PG/PC-Schnittstelle. Für die "Benutzte Schnittstellenparametrierung" sollte nicht TCP/IP(Auto) verwendet werden. Verwenden Sie stattdessen feste IP-Adressen.

Störung der Verbindung zwischen Server und Client

Wenn der Rechner mit dem Engineering System gleichzeitig als Server verwendet wird und ein Client die Verbindung zum Server nicht aufbauen kann, sollten Sie die auf dem Server gesetzten Freigaben überprüfen.

1. Benden Sie auf dem Server die Runtime.
2. Wählen Sie im Kontextmenü der Netzwerkumgebung auf dem Desktop des Servers den Befehl "Computer suchen...".
3. Geben sie den Namen des Servers als den zu suchenden Computer ein.
4. Öffnen Sie den gefundenen Computer um die freigegebenen Verzeichnisse zu erkennen.
5. Enternen Sie alle Freigaben die mit "WinCC_Project_HMI" beginnen. Informationen dazu finden Sie in der Dokumentation des Betriebssystems.

Das Fehlverhalten wird verursacht durch die Verwendung des Befehls "Speichern unter..." in Verbindung mit dem Starten der Runtime dieses neuen Projektes. Zu Sicherungszwecken können Sie mit "Speichern unter..." Kopien der Projekte erzeugen. Sie sollten jedoch mit dem ursprünglichen Projekt weiterarbeiten.

Starten von WinCC Runtime Professional

Wenn auf einem Rechner Engineering System und Runtime betrieben werden, sollte Runtime oder Simulation zu einem im Engineering System geöffneten Projekt nur über das TIA Portal gestartet und beendet werden. Andere Möglichkeiten, z. B. das Symbol im Infobereich der Taskleiste, sollten nicht genutzt werden.

Beim Konfigurieren des Autostarts am Client ist für jedes eingetragene Server-Projekt die Eingabe von Login und Passwort zwingend notwendig. Um jedem Server ein Login zuordnen zu können, muss die Eingabe des jeweiligen Projektpfades mit der Taste „Anwenden“ bestätigt werden. Dies führt dazu, dass Sie bei redundanten oder alternativen Projekten Login und Passwort zweimal eingeben müssen.

Einstellung der Dienste für den SQL-Server

Um die volle Funktionalität des SQL-Servers für WinCC und WinCC Runtime sicherzustellen, müssen Sie Einstellungen des SQL-Servers überprüfen.

1. Starten Sie im Startmenü "Programme > Microsoft SQL Server 2014 > Configuration Tools > SQL Server 2014 Configuration Manager".
2. Klicken Sie im Baum auf "SQL Server Services".
3. Überprüfen Sie die Dienste "SQL Server (WinCC)", "SQL Server (WINCCPLUSMIG)" und "SQL Server Browser". Für "Start Mode" muss "Automatic" eingetragen sein. Für "Log On As" muss "LocalSystem" eingetragen sein. Ändern Sie gegebenenfalls die Einstellungen.
4. Klicken Sie im Baum auf "SQL Server Network Configuration".
5. Klicken Sie auf "Protocols for WinCC".
6. Überprüfen Sie das Protokoll "TCP/IP". Bei "Status" muss "Enabled" eingetragen sein. Ändern Sie gegebenenfalls die Einstellungen.

"Systemfehler melden" in WinCC Runtime Professional

Die Funktionalität "Systemfehler melden" ist in der WinCC Runtime Professional in Zusammenarbeit mit einer S7-300 bzw. S7-400 eingeschränkt. In einer Meldeanzeige wird zu einer Diagnoseart immer nur eine Meldung angezeigt. Meldungen zu weiteren Fehlern der gleichen Diagnoseart werden nicht angezeigt.

WinCC-Schnittstellen und 64-Bit Betriebssystem

Die offenen Schnittstellen von WinCC Runtime Professional bieten keine native 64-Bit Unterstützung. Davon sind vor allem Runtime API, VBS und der WinCC OleDB-Provider betroffen. Um die Schnittstellen von WinCC unter einem 64-Bit Betriebssystem nutzen zu können, müssen Sie folgendes beachten:
- VB-Skripte können Sie nicht einfach per Doppelklick starten. Sie müssen explizit die 32-Bit-Version unter "syswow64\wscript.exe" verwenden.
- .NET-Applikationen, die die WinCC-API verwenden, müssen explizit als 32-Bit Applikationen übersetzt werden. Nicht mit "AnyCPU" sondern mit "x86".
- C++ Applikationen dürfen nicht als 64-Bit Applikationen übersetzt werden.

Dynamische Grafiken im WinCCViewer RT

Wenn Sie dynamische Grafiken aus einer Grafiksammlung, z.B. an Skripte gebunden verwenden, dann werden diese Grafiken im WinCCViewer RT nicht aktualisiert.

Wählen Sie unter "Runtime-Einstellungen > Grafikeinstellungen" die Option "Benutzerdefiniert" aus.

Runtime Professional: Laden eines Projekts ohne Verbindung zum Projektierungs-PC

Bevor Sie mit dem Laden eines Projekts auf der Bedienstation starten, speichern Sie die Daten wie "Benutzerverwaltung" und "Rezepturdaten".

1. Daten der Benutzerverwaltung werden standardmäßig überschrieben.
   Projektieren Sie daher zwei Schaltflächen in Ihrem Projekt für den Export bzw. Import der Benutzerverwaltung mit der Systemfunktion „ExportiereBenutzerverwaltung“. Der Export der Daten ist nur erforderlich, wenn Sie in Runtime Änderungen in der Benutzeranzeige vorgenommen haben.
2. Beim Laden des Projekts bestimmen Sie im Dialog „Laden Vorschau“, ob Rezepturdaten überschrieben werden.
   Rezepturdaten können Sie in der Rezepturanzeige über die Bedienelemente „Archiv exportiere“ und „Archiv importieren“ speichern.

Beachten Sie, dass die zuletzt importierten bzw. geladenen Daten in Runtime verwendet werden.

Betrieb auf einem Netzwerk Server

Der Betrieb von WinCC Runtime Professional auf einem Netzwerk Server (z. B. Domain Controller, File-Server, Namensdienst-Server, Software Firewall, Media Server, Exchange Server) ist nicht zulässig.

Verbindung mit S7-1200 in Systemdiagnose

Bei Verwendung von Systemdiagnose kann nur ein WinCC Runtime Professional HMI System mit einer S7-1200 (bis V3) verbunden werden.

Automatische Überprüfung der Software-Aktualisierungen deaktivieren

Wenn das Engineering System zusammen mit der Runtime auf einem PC installiert ist, bekommt der Bediener Benachrichtigungen über die Software-Aktualisierungen. Damit das System auf einem Mehrplatzsystem stabil läuft, muss auf allen Rechner dieselbe Software-Version installiert sein.

Es ist möglich, die automatische Überprüfung der Software-Aktualisierungen zu deaktivieren und die Performance damit zu verbessern.

Sie deaktivieren die automatische Überprüfung der Software-Aktualisierungen unter "Einstellungen > Allgemein > Software-Aktualisierungen", indem Sie das Optionskästchen "Täglich auf Aktualisierungen überprüfen" deaktivieren.

Einschränkung der Online-Delta-Ladefähigkeit

Wenn Sie ein Design ersetzen, geht die Online-Delta-Ladefähigkeit des Projekts verloren. TIA Portal gibt in diesem Fall keine Fehlermeldung aus.

Bildobjekte in Protokollen als PDF

Die Qualität der Bildobjekte in den Protokollen, die Sie in Runtime als PDF-Datei erzeugen, hängt von den verwendeten PDF-Druckertreibern ab.

Projekte in Runtime nach Windows 10-Update

Sie haben auf einem PC mit WinCC Professional V15 und einem PC mit WinCC Professional RT V15 die Windows 10-Version 1709 installiert. Auf dem Runtime-PC haben Sie ein Windows 10-Update auf die Version 1803 durchgeführt. Nach dem Windows 10-Update ist es in Runtime unter Umständen nicht mehr möglich, ein vom Engineering System-PC geladenes Projekt manuell zu öffnen.

Abhilfe: Stellen Sie den Starttyp des Diensts der WinCC-SQL-Instanz manuell auf "Automatisch (Verzögerter Start)" und starten Sie das System neu.

3.8 Bediengeräte

3.8.1 Hinweise zu Bediengeräten

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Nicht unterstützte Bediengeräte (ab TIA Portal V15)

Alle Bediengeräte mit der Version 11.x werden ab TIA Portal V15 nicht mehr unterstützt.

Folgende Bediengeräte werden ab TIA Portal V15 nicht mehr unterstützt:

- WinAC MP 177, WinAC MP 277, WinAC MP 377
- Panels: OP und TP der 70er Serie, 170er Serie und 270er Serie
- Multi Panels: Multi Panels der 170er Serie, 270er Serie und 370er Serie

Um im TIA Portal V15 andere Geräte, z. B. Basic, Comfort oder Mobile Panels, nutzen zu können, die mit der Version 11.x projektiert wurden, ändern Sie die Geräteversion des betroffenen Geräts.

Auch die Ablagestruktur im Installationsverzeichnis enthält keine Daten für 11.0 mehr. Auf diese Weise wird die Größe des TIA Portal-Installationspakets im Vergleich zu den vorhergehenden Versionen reduziert.

Nicht unterstützte Bediengeräte (ab TIA Portal V15.1)

Zusätzlich zu den Geräten, die im TIA Portal V15 nicht unterstützt werden, werden ab TIA Portal V15.1 die folgenden Geräte nicht mehr unterstützt:

- WinCC Runtime Advanced V11

Mehrtastenbedienung

Durch eine Mehrtastenbedienung können unbeabsichtigte Aktionen ausgelöst werden:

- Wenn Sie ein Tasten-Gerät verwenden, dürfen Sie nicht mehr als zwei Funktionstasten gleichzeitig betätigen.
- Wenn Sie ein Touch-Gerät, einen Standard-PC oder einen Panel-PC verwenden, dürfen Sie immer nur eine Funktionstaste oder Schaltfläche gleichzeitig bedienen.

Bediengeräte mit hoher Kommunikationslast

Wenn bei einem Panel viele Verbindungen zu Steuerungen oder anderen Bediengeräten projektiert sind, sollte die S7-Diagnose deaktiviert sein. Andernfalls besteht die Gefahr der Überlastung des Panel.

TS Adapter mit Ethernet Schnittstelle

Wenn ein Bediengerät über Ethernet und einen TS Adapter verbunden ist, kann es nicht auf Werkseinstellungen zurückgesetzt werden.

Simulation mit realer Steuerungsanbindung

Der von der Simulation verwendete Zugangspunkt ist unabhängig von den Einstellungen im Engineering System und kann nur über das Tool " PG/PC Interface einstellen" in der Systemsteuerung verändert werden. Wenn die Verbindung zu einer Steuerung direkt nach dem Start der Simulation mit Meldung 140001 wieder abgebaut wird, sollten Sie den von der Simulation verwendeten Zugangspunkt mit " PG/PC Interface einstellen" überprüfen.

1. Doppelklicken Sie in der Systemsteuerung auf "PG/PC Interface einstellen". Ein Dialog wird geöffnet.
2. Wählen Sie im Bereich "Zugangspunkt der Applikation" "S7ONLINE" standardmäßig für HMI.
3. Wählen Sie im Bereich "Benutzte Schnittstellenparametrierung" die Schnittstelle.
4. Beenden Sie den Dialog "PG/PC Interface einstellen" mit OK.

Basic Panels auf Geräteversion V12 hochrüsten

Bevor Sie Basic Panels von der Geräteversion V11 auf die Geräteversion V12 hochrüsten, übertragen Sie das Image der V11 SP2 Updates 5 oder höher auf die Geräte.
Wählen Sie im Dialog "SIMATIC ProSave [OS-Update]" die Einstellung "Rücksetzen auf Werkseinstellungen".

Damit starten Sie immer eine funktionsfähige Aktualisierung des Images.

Betroffene Geräte:

- KP300 Basic mono PN
- KP400 Basic color PN
- KTP400 Basic color PN

Simulation der Basic Panels

Für die Ausgabe einer externen Variablen verwenden Sie ein Ausgabefeld in einem Meldetext. Bei der Simulation wird dann der Inhalt dieses Ausgabefelds immer mit "0" angezeigt.

Basic Panels 2nd Generation

Basic Panelds 2nd Generation unterstützen die Option Sm@rtServer.

Wenn Sie keinen USB-Hub verwenden, wählen Sie den USB-Port USB_X60.1 als Ablagepfad.

Bediengeräte mit Betriebssystem Windows CE 5.0 und höher

Aufgrund einer geänderten Sicherheitseinstellung bei der Client-Server-Kommunikation darf der Zeitunterschied zwischen Bediengerät (Client) und einem PC (Server) nicht größer als 1 Tag sein. Wenn Sie z. B. Rezepturdaten vom Bediengerät auf ein Netzlaufwerk sichern, stellen Sie sicher, dass die Uhrzeit auf PC (Server) und Bediengerät (Client) korrekt eingestellt ist.

Verbindungswechsel im Control Panel bei Basic Panels

Wenn Sie die Funktion "Override projected connection information" verwenden, gilt folgende Einschränkung:

Der Verbindungswechsel im Control Panel eines Basic Panels von einer Steuerung ohne Schutzstufe zu einer Steuerung mit der Schutzstufe "Kompletter Schutz" ist nicht möglich.

Mobile Panels 277F IWLAN

Mit WinCC V13 SP1 Update 4 sind Mobile Panel 277F IWLAN für die Verwendung mit CPUs vom Typ S7-1500F freigegeben.

Falls sich nach dem Einschalten des Panels oder nach einem Abbruch der Failsafe-Verbindung zu einer CPU S7-1500F diese Verbindung nicht von selbst aufbaut, muss an der CPU ein STOP-RUN-Übergang durchgeführt werden.

Betroffene MLFBs:

- 6AV6645-0EB01-0AX1
- 6AV6645-0EC01-0AX1
- 6AV6645-0EF01-0AX1
- 6AV6645-0GB01-0AX1
- 6AV6645-0GC01-0AX1
- 6AV6645-0GF01-0AX1
- 6AV6645-0EF02-0AX1
- 6AV6645-0EF01-0AX1

Einsatz des Mobile Panel 277F IWLAN im Industriebereich

Um Kommunikationsstörungen zwischen Bediengerät und Access Point zu vermeiden, beachten Sie folgende Hinweise:

IWLAN-Planung: Funkverbindung zwischen Bediengerät und Access Point

- Es ist eine Funkfeldplanung durchzuführen. Bei Kanälen, die Dynamic Frequency Selection (DFS) erfordern, müssen dabei Radareinflüsse berücksichtigt werden.
- Vermeiden Sie Funkstörungen durch andere Geräte, z. B. Handheld Devices mit Bluetooth oder WLAN-Funktion.
- Verwenden Sie für räumlich benachbarte Access Points Datenkanäle, die mindestens einen Funkkanal auseinanderliegen (z. B. nicht 36 und 40, sondern 36 und 44).

- Platzieren Sie mehrere Access Points nicht in unmittelbarer Nähe zueinander. Empfehlenswert ist ein Abstand von mindestens 5 m.
- Verwenden Sie für das Mobile Panel 277F IWLAN die „Software Retry-number“ 15, statt den Default-Wert 3.

Verwenden der Funktion „rapid roaming“ mit iPCF-MC

- Verwenden Sie für die PROFIsafe-Kommunikation über PROFINET immer iPCF bzw. iPCF-MC (rapid roaming).
- Verwenden Sie für alle Access Points dieselbe SSID.
- Der „Background scan“ sollte beim Bediengerät ausgeschaltet sein (“Background scan interval” für das Bediengerät = 1 (Default-Wert)).
- Verwenden Sie für alle Access Points denselben „Management channel“ und unterschiedliche überlappungsfreie „Data channels“.
- Parametrieren Sie die „PN-IO cycle time“ größer der „iPCF scan cycle time“.
- Beachten sie bei der PROFIsafe-Parametrierung die eingestellten Timeouts.

Variable für "Zonen-ID/Anschlusspunkt-ID" eines Mobile Panel 277 IWLAN V2

Die für "Zonen-ID/Anschlusspunkt-ID" verwendete Variable muss bei Mobile Panel 277 IWLAN V2 Geräten den Datentyp INT besitzen. Achten Sie bei der Migration eines Projektes darauf den Datentyp gegebenenfalls anzupassen.

Mobile Panels 277F IWLAN (RFID Tag): F_DB_STATES

Der Baustein F_DB_STATES wird mit WinCC V14 nicht mehr geliefert. Der Baustein F_DB_STATES diente lediglich dem Datenaustausch. Um die Funktion des F_DB_STATES nachzubilden, können Sie einen F_DB projektieren. Beachten Sie die Beschreibung des F_DB_STATES in der Betriebsanleitung Ihres Bediengeräts oder im Informationssystem.

IPC427D

Wenn in einem V12 oder V13 Projekt ein IPC427D projektiert ist, bei dem IE1 der PC-Station zugewiesen ist und nach dem Hochrüsten auf V14 ein Wechsel der Geräteversion der PC-Station von V1 nach V2 durchgeführt, muss vor dem Übersetzen des Projekts die Schnittstellenzuweisung von CP_IE_1 zunächst nach "Keine oder eine andere Windows-Einstellung" und anschließend zu "SIMATIC PC Station" geändert werden. Anschließend kann das Projekt fehlerfrei übersetzt werden.

Meldungen drucken (Mobile Panels 2nd Generation)

Es ist nicht möglich, Meldungen als PDF oder HMTL zu drucken, wenn als "Storage Location" der interne Speicher "/flash" ausgewählt ist. Um Meldungen als PDF oder HMTL zu drucken, wählen Sie als Ablageort eine SD-Speicherkarte oder einen USB-Stick aus.

DHCP auf KTP 400 PN, KTP 700 PN und KTP 900 PN

Für die Geräte KTP 400 PN, KTP 700 PN und KTP 900 PN ist der Betrieb mit DHCP nicht empfohlen.

Unter Umständen kann es vorkommen, dass das Gerät nicht mehr bootet. In diesem Fall entfernen Sie das Netzwerkkabel und booten Sie das Gerät dann erneut. Nach dem erfolgreichen Hochlauf vergeben Sier bitte eine feste IP Adresse.

Mobile Panels 2nd Generation

Mobile Panels 2nd Generation können mit TIA Portal V15.1 in der Geräteversion V15 projektiert werden. Allerdings kann das übersetzte Projekt nicht auf diese Geräte geladen werden.

Um mit Mobile Panels 2nd Generation im TIA Portal V15.1 zu arbeiten, ist eine Hochrüstung auf Geräteversion V15.1 erforderlich.

Sm@rtServer mit Mobile Panels 2nd Generation

Ab WinCC V15.1 ist die Option Sm@rtServer lizenzfrei für alle Mobile Panels 2nd Generation verfügbar und projektierbar. Beachten Sie die wichtigen Hinweise im Siemens Industry Online Support mit der Beitrags-ID 109758056 (https://support.industry.siemens.com/cs/ww/de/view/109758056).

Sm@rtServer mit KTP Mobile Panels

Die Auswertung der Sicherheitsbedienung über PROFIsafe ist nur gewährleistet, wenn der Sm@rtServer auf dem Panel deaktiviert ist.

Die Benutzung von Sm@rtServer auf Geräten ohne Sicherheitsfunktion ist nicht empfohlen.

Simulation von Basic Panels 2nd Generation

Für die Simulation der Basic Panels 2nd Generation sollte die 3D Hardwarebeschleunigung der Grafikkarte aktiviert sein.

Installation von Schriftarten

Wenn Sie mit Prosave neue Zeichensätze auf einem Panel installieren haben, ist ein Neustart des Panels erforderlich. Die Dauer des Startvorgangs der Runtime ist abhängig von Anzahl und Größe der Schriftarten.

Unterbrechung der DH485-Verbindung

Wenn Sie bei Comfort-Panels das Ethernet-Kabel an den Anschluss "PN_X1" des HMI-Panels anschließen, kann es zu einer Unterbrechung der DH485-Verbindung kommen.

Abhilfe für Comfort Redesign/PRO-Panels: Verwenden Sie beim Ausführen des DH485-Projekts auf dem HMI-Panel den Anschluss "PN_X3(Gigabit)" für die Ethernet-Kommunikation.

Sm@rtServer mit fehlersicheren Mobile Panels KTP400F/700F/900F Mobile

Für eine korrekte Auswertung einer sicherheitsgerichteten Bedienung stellen Sie sicher, dass Sm@rtServer auf dem Bediengerät deaktiviert ist.

Sichere E-Mail-Benachrichtigung aus Runtime projektieren

Ab TIA Portal V14 kann auf Comfort Panels und KTP Mobile Panels der verschlüsselte Versand von Emails mit TLS1.2 erfolgen.

3.9 Optionen

3.9.1 Sm@rtServer

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Dateigröße für File Broswer

Die maximale Größe für den Upload oder Download von Dateien beträgt 2GB.

SmartClient *.sac-Dateien unter Windows 10 öffnen

Sie speichern die Daten der aktuellen SmartServer-Verbindung mithilfe der ausführbaren Datei SmartClient.exe mit dem Format *.sac. Um die so erzeugte Dateien unter Windows 10 automatisch zu öffnen, muss eine *.sac-Datei mit SmartClient.exe einmal manuell über die Auswahlschaltfläche und Dialogfeld in der SmartClient.exe geöffnet werden.

3.9.2 Audit

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

WinCC Audit Viewer 7.2 Update

Für den optional installierbaren WinCC Audit Viewer 7.2 steht Ihnen ein Update zur manuellen Installation zur Verfügung.

Installieren Sie das Update zum WinCC Audit Viewer 7.2, indem Sie die Datei "AuditViewer_V72_00_Upd1.exe" aus dem Support-Verzeichnis der Produkt-DVD ausführen.

3.9.3 DataMonitor

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Deinstallation von WinCC Runtime

Wenn Sie DataMonitor auch nach einer Deinstallation von WinCC Runtime Professional weiter verwenden wollen, müssen Sie eine Reperatur der Installation durchführen Gehen Sie dazu wie folgt vor:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie "Software".
3. Wählen Sie in der Liste der installierten Programme "SIMATIC WinCC/DataMonitor Client"
4. Klicken Sie die Schaltflächen "Ändern/Entfernen".
5. Wählen Sie im Setup von WinCC/DataMonitor die Einstellung "Reperatur".

Bezeichnung der Benutzergruppen

Die Bezeichnung der Benutzergruppen unterscheidet sich an einigen Stellen der Online-Hilfe von den tatsächlich angelegten Benutzergruppen. Es gilt folgende Zuordnung:

SIMATIC Report Administrators entspricht DM_Admin

SIMATIC Report Users entspricht DM_User

Excel-Workbook auf einem PC mit Windows 7

Wenn Sie Excel-Workbook auf einem PC mit Windows 7 betreiben, müssen Sie die Darstellung "Aero Glass" deaktivieren.

Import von Projektierungsdaten

Wenn Sie ein Projekt über einen externen Datenträger auf ein Gerät laden, dann kann der DataMonitor auf diesem Gerät keine korrekte XML-Datei exportieren.

Die exportierte XML-Datei kann danach zum Import von Projektierungsdaten nicht mehr verwendet werden.

Kanaldiagnose-Anzeige

Das Objekt "Kanaldiagnose-Anzeige" wird von der Option WinCC DataMonitor nicht unterstützt.

3.9.4 WebNavigator

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Deinstallation von WinCC Runtime

Wenn Sie WebNavigator auch nach einer Deinstallation von WinCC Runtime Professional weiter verwenden wollen, müssen Sie eine Reperatur der Installation durchführen Gehen Sie dazu wie folgt vor:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie "Software".
3. Wählen Sie in der Liste der installierten Programme "SIMATIC WinCC/WebNavigator Client"
4. Klicken Sie die Schaltflächen "Ändern/Entfernen".
5. Wählen Sie im Setup von WinCC/WebNavigator die Einstellung "Reperatur".

WebNavigator-Client: Manuelles Abmelden

Benutzer können nun auch mit der ODK-Funktion "PWRTLogout()" manuell vom WebNavigator-Client abgemeldet werden. Eine Beschreibung der Funktion finden Sie in der Runtime API-Dokumentation unter "Funktionen der Benutzerverwaltung > Funktionen zum Anmelden, Abmelden".

Virenscanner Trend Micro OfficeScan

Die Verwendung des VirenScanners Trend Micro OfficeScan in Verbindung mit dem WebNavigator ist nicht empfehlenswert, da dieser Virenscanner zu sehr langen Bildaufschlagszeiten führen kann.

ProDiag-Übersicht, GRAPH-Übersicht und PLC-Code-Anzeige

Die Verwendung von WinCC Objekten "GRAPH-Übersicht", "ProDiag-Übersicht" und "PLC-Code-Anzeige" wird in der WinCC Option Web-Navigator nicht unterstützt.

Zeitzonen von Webparts

Wenn Sie im Webcenter ein Webpart unter einer bestimmten Zeitzone erstellen und konfigurieren, wird die Zeitzone im Webcenter gespeichert.

Eine Änderung der Zeitzone im Betriebssystem kann dazu führen, dass die Zeitdarstellung im Webpart nicht korrekt ist.

Nicht erlaubte Zeichen im WinCC WebNavigator

Verwenden Sie in den Bildnamen und Projektnamen die Zeichen des ASCII-Zeichensatzes. Verwenden Sie keine Sonderzeichen und nationalen Sonderzeichen.

Kanaldiagnose

Das Objekt "Kanaldiagnose" wird von der Option WinCC WebNavigator nicht unterstützt.

Automatischer Trigger bei SmartTags in Skripten

Wenn Sie SmartTags in Skripten in den WebNavigator-Bildern nutzen, verwenden Sie den automatischen Trigger.

Verzögerung bei der Ausgabe der WebNavigator-Bilder

Durch die WebServer-Technologie kann es vorkommen, dass aktualisierte WebNavigator-Bilder in Runtime verzögert ausgegeben werden.

Neue Funktion für WebNavigator-Client

Ab der TIA Portal Version V15.1 haben sie die Möglichkeit, im ActiveX-Control UserAdminControl für WebNavigator-Client ein Startbild und die Startsprache festzulegen.

3.9.5 WebUX

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Aktualisierungszyklen

Bei einem Zugriff über WebUX erfolgt die Aktualisierung von Dynamiken, die den Bildzyklus verwenden, immer mit 2 Sekunden, auch wenn ein davon abweichender Bildzyklus projektiert ist.

Automatisches Login

Wenn das automatische Login nicht mehr gewünscht ist, müssen Sie die Browser-Historie im jeweiligen Browser löschen.

Neue Funktion für WebUX-Client

Ab der TIA Portal Version V15.1 haben sie die Möglichkeit, im ActiveX-Control UserAdminControl für WebUX-Client ein Startbild und die Startsprache festzulegen.

3.9.6 Redundancy

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Namen in der Konfiguration eines redundaten Systems

Für die Konfiguration eines redundanten Systems sind ausschließlich die Computernamen der Bediengeräte relevant. Den Computernamen eines Bediengeräts definieren Sie in dessen Betriebssystem. Der Name des Bediengeräts aus der Projektnavigation dient nur zur Identifikation des Bediengeräts im Projekt.

3.9.7 SIMATIC Information Server

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Installation von SIMATIC Information Server und WinCC Runtime Professional auf einem PC

Um den SIMATIC Information Server und WinCC Runtime Professional auf einem PC einzurichten, halten Sie die folgende Installationsreihenfolge ein:

1. Installieren Sie WinCC Runtime Professional.
2. Installieren Sie Information Server.

Wenn Sie die Installationsreihenfolge nicht einhalten, können Verbindungsprobleme auftreten.

Index

.

.Net Service Packs, 5
.Net Versionen, 5

A

Asiatisch, 5

B

Baugruppe
    stecken, 21
    ziehen, 21
Beobachtungstabelle
    Geänderte Anzeigeformate, 44
    Laden von Datenbausteinen bei laufendem Steuerauftrag, 43
    Mehrfachzugriffe auf dieselbe CPU, 43
Bibliothek, 8
Bibliothekselemente vergleichen, 8

C

CP 343-2, 28
CP 5512, 10

D

deaktivieren, 62
DEMUX, 59
Dezimalstellen, 6

E

EC31-RTX, 28
ENO, 62
Ethernet-Baugruppe
    Ziehen/Stecken, 21

F

Firmware-Lader - chinesische GUI, 32
Force-Auftragauf SD-Karte, 9
Forcetabelle
    Geänderte Anzeigeformate, 44
Funktionswert, 60

H

Hardware
    Erkennung, 31

I

IP-Adresse, 10
ISO-on-TCP, 33
ISO-Transport, 33

K

Kompatibilität, 22, 57
Konvertierungen, 62

L

Laden
    in Gerät, 22
    Modulkommentare, 28
    von einem Gerät, 22, 40

M

Memory Card, 9
    Ziehen/Stecken, 21
Migration der IP-Zugriffsschutzliste bei der Aktivierung von Security, 35
Migration von Projekten mit Ethernet-CPs und Security, 35
Module der S7-1200, 28
Modulkommentare, 28
MUX, 58

N

Netzlaufwerk, 6
Netzwerk, 40

O

offener Force-Auftrag, 9
Online verbinden
    mehrere TIA Portal Instanzen, 21
Online-Betrieb, 21
Online-Verbindung
    mehrere TIA Portal Instanzen, 21

P

Parameter K, 58, 59
Peripherie, 43, 44
PLC-Migration
    Anweisungsprofil, 21
PLCSIM, 40
Private Cloud, 11
Program_Alarm, 59

R

Return, 60

S

S_CONV, 58
S7-1200, 28
S7-Port Configuration Tool, 23
S7-Routing, 33
Schriftzeichen, 5
SD-Karte, 9
Security-Online-Diagnose von S7-CPs, 35
SET, 58
Sicherheit, 40
SIMATIC S7 Embedded Controller, 28
Spezialdiagnose, 32
Spezialdiagnose - chinesische GUI, 32
STRING, 62

T

TiaDefaultProjectPath, 11
TiaUserSettingsPath, 11
Trace
    Löschen von Messungen, 31
    Messungen auf der Speicherkarte löschen, 31
    Messungen im Gerät löschen, 31
TSAP-Länge, 33

U

USB-Kartenleser, 9

V

Virtuelle Umgebung, 11
VMware - PC-CPs, 33

Z

Zeichenkette, 58
Zeichenkette konvertieren, 58

SIEMENS

Installation

Programmier- und Bedienhandbuch

Hinweise zur Installation 1

Parallelinstallationen 2

Lizenzierung 3

Systemvoraussetzungen für die Installation 4

Installationsprotokoll 5

Installation starten 6

Installierte Software anzeigen 7

Installierte Produkte modifizieren oder hochrüsten 8

Installierte Produkte reparieren 9

Deinstallation starten 10

Updates und Support Packages installieren 11

Support Packages automatisiert installieren 12

Migrations-Tool installieren und deinstallieren 13

Multiuser-Server installieren und deinstallieren 14

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Hinweise zur Installation....5

2 Parallelinstallationen....7

3 Lizenzierung....9

3.1 Hinweise zu Lizenzen....9

3.2 Lizenzierung von STEP 7 und WinCC....11

3.3 Umgang mit Lizenzen und License Keys....17

3.4 Lizenzierung von Bediengeräten....22

3.5 Lizenzierung von WinCC Runtime auf PC-basierten Bediengeräten....23

3.6 Lizenzierung WinCC Runtime Professional asiatisch....24

3.7 STEP 7 ohne Lizenz für PC-CP....25

4 Systemvoraussetzungen für die Installation....27

4.1 Hinweise zu den Systemvoraussetzungen....27

4.2 Allgemeine Software- und Hardwarevoraussetzungen....29

4.3 Produktspezifische Besonderheiten....33

4.3.1 Installation von WinCC Runtime Professional....33

4.3.2 Deinstallation einer SQL-Instanz....33

4.3.3 Hinweise zur Installation des TIA Portal Teamcenter Gateway....36

5 Installationsprotokoll....39

6 Installation starten....41

7 Installierte Software anzeigen....45

8 Installierte Produkte modifizieren oder hochrüsten....47

9 Installierte Produkte reparieren....49

10 Deinstallation starten....51

11 Updates und Support Packages installieren....53

11.1 Verfügbarkeit von Updates und Support Packages prüfen und installieren....53

11.2 Arbeiten mit einem unternehmenseigenen Server....58

11.2.1 Eigenschaften und Vorteile eines unternehmenseigenen Servers....58

11.2.2 Unternehmenseigenen Server für Updates konfigurieren....59

11.2.3 Updates auf einem unternehmenseigenen Server bereitstellen....63

12 Support Packages automatisiert installieren....65

12.1 Support Packages automatisiert installieren....65

12.2 Rückgabewerte des Installationsprozesses....67

12.3 Protokolldatei....68

13 Migrations-Tool installieren und deinstallieren....69
13.1 Systemvoraussetzungen....69
13.2 Migrations-Tool installieren....70
13.3 Migrations-Tool deinstallieren....71
14 Multiuser-Server installieren und deinstallieren....73
14.1 Hinweise zur Installation des Multiuser-Servers....73
14.2 Multiuser-Server installieren....76
14.3 Multiuser-Server deinstallieren....78
Index....79

Hinweise zur Installation

1

Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Installation in einer virtuellen Umgebung (Private Cloud)

Eine Anleitung zur Installation des TIA Portals in einer virtuellen Umgebung (Private Cloud) finden Sie auf dem Installationsdatenträger im Verzeichnis "Documents\Readme \<Sprachverzeichnis>". Öffnen Sie hier das PDF-Dokument "TIAPortalCloudConnectorHowTo<Sprachkennung>.pdf".

Automatisierte Installation

Eine Beschreibung zur automatisierten Installation finden Sie auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Verwendung gleicher Versionen der TIA Portal Produkte bei der Installation

Achten Sie bei der Installation von verschiedenen TIA Portal Produkten darauf, dass Sie für die Installation die gleichen Stände der Service Packs und Updates verwenden. Wenn Sie also z. B. bei STEP 7 V13 das SP1 installiert haben, dann müssen Sie auch für WinCC V13 das SP1 installieren. Die Installation der Service Packs und Updates muss für alle Produkte zu einem gemeinsamen Zeitpunkt erfolgen. Starten Sie das TIA Portal nicht, bevor Sie nicht alle Produkte hochgerüstet haben.

Die Servicepacks können Sie im Internet unter Siemens Industry Online Support (https://support.industry.siemens.com/cs/de/de/) herunterladen.

Zielverzeichnis der Installation

Verwenden Sie keine UNICODE-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

Der Installationspfad kann nach der Installation von einem der TIA Portal-Produkte nicht mehr geändert werden.

Sicherheitseinstellungen bei der Installation von WinCC

Wenn Sie WinCC installieren, werden in Ihrem Betriebssystem Sicherheitseinstellungen verändert. Diese Sicherheitseinstellungen werden während der Installation aufgelistet.

Die Änderungen der Sicherheitseinstellungen müssen Sie bestätigen.

Wenn Sie nach der Installation an Ihrem Betriebssystem Änderungen vornehmen, können dadurch die Änderungen der Sicherheitseinstellungen durch die Installation des TIA Portals verändert werden.

Die Änderungen der Sicherheitseinstellungen durch die Installation des TIA Portals können Sie folgenderweise wiederherstellen:
"Start > Alle Programme > Siemens Automation > Security Controller > Einstellungen wiederherstellen“.

Verwendung von Virenscannern

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscanner verhindern dies. Daher wird empfohlen, während der Installation von TIA Portal Virenscanner zu deaktivieren und danach wieder zu aktivieren.

Kompatibilität mit V12 SP1 und V13

In das Installationsverzeichnis unter ..\<INSTALLDIR>\SampleProjects werden leere Projekte der Vorgängerversionen installiert, damit das aktuelle TIA Portal im Kompatibilitätsmodus geöffnet werden kann. Diese Projekte müssen in ein lokales Verzeichnis mit Vollzugriff kopiert werden, bevor sie verwendet werden können. Weitergehende Informationen finden Sie unter FAQ ID 82169157.

FAQs zum TIA Portal

FAQs zum TIA Portal finden Sie unter FAQs (http://support.automation.siemens.com/WW/view/de/28919804/133000).

Parallelinstallationen

2

Parallelinstallation von WinCC V15.1 und anderen SIMATIC HMI Produkten

Die parallele Installation von WinCC V15.1 mit Versionen von WinCC flexible vor WinCC flexible 2008 SP1 ist nicht zulässig.

Die parallele Installation von WinCC V15.1 mit Versionen von WinCC vor WinCC V7.0 SP2 ist nicht zulässig. Die parallele Installation von WinCC V15 mit WinCC V7.0 SP2 oder WinCC V7.0 SP3 ist nicht zulässig für:

- WinCC V15.1 Professional
- WinCC V15.1 Runtime Professional

Parallele Nutzung

Sollte die Software im Namen oder der Licence-Key nach der Installation den Zusatz "Combo" tragen, ist Ihnen gemäß Ziffer 4.3 der Allgemeinen Bedingungen (siehe auch Setup-Text) die Nutzung folgender Produkte/Versionen erlaubt:

- Mit der Lizenz "STEP 7 Professional Combo": ältere Versionen von STEP 7 (STEP 7 Professional 2006 - STEP 7 Professional V15)
- Mit der Lizenz "WinCC V15.1 Comfort Combo": WinCC flexible 2008 Standard
- Mit der Lizenz "WinCC V15.1 Advanced Combo": WinCC flexible 2008 Advanced

Lizenzierung 3

3.1 Hinweise zu Lizenzen

Bereitstellung der Lizenzen

Die Lizenzen für die Produkte des TIA Portals werden in der Regel auf einem Installationsdatenträger ausgeliefert und über den Automation License Manager während des Installationsprozesses des TIA Portals automatisch übertragen.

Bevor Sie das TIA Portal deinstallieren, müssen Sie noch benötigte Lizenzen transferieren und sichern. Verwenden Sie dazu den Automation License Manager.

Bereitstellung des Automation License Managers

Der Automation License Manager wird auf dem Installationsdatenträger mitgeliefert und während des Installationsprozesses automatisch übertragen.

Wenn Sie das TIA Portal deinstallieren, bleibt der Automation License Manager weiterhin auf Ihrem System installiert.

Arbeiten mit dem Automation License Manager

Der Automation License Manager ist ein Produkt der Siemens AG und wird zur Handhabung von License Keys (technische Repräsentanten von Lizenzen) eingesetzt.

Software-Produkte, für deren Betrieb License Keys notwendig sind, wie z. B. das TIA Portal, melden den Bedarf an License Keys automatisch beim Automation License Manager an. Wenn der Automation License Manager für diese Software einen gültigen License Key findet, so kann die Software entsprechend der mit diesem License Key verbundenen Lizenz-Nutzungsbedingungen verwendet werden.

Hinweis

Weiterführende Informationen, wie Sie mit dem Automation License Manager Ihre Lizenzen verwalten können, finden Sie in der mit dem Automation License Manager mitgelieferten Dokumentation.

Siehe auch

Hinweise zu den Systemvoraussetzungen (Seite 27)

Installation starten (Seite 41)

Installierte Software anzeigen (Seite 45)

Installierte Produkte modifizieren oder hochrüsten (Seite 47)

Installierte Produkte reparieren (Seite 49)
Deinstallation starten (Seite 51)
Installationsprotokoll (Seite 39)

3.2 Lizenzierung von STEP 7 und WinCC

Einführung

Für die Lizenzierung der folgenden Editionen von STEP 7 benötigen Sie einen License Key:

- STEP 7 Basic
- STEP 7 Professional

Sie können den entsprechenden License Key für STEP 7 nach der Installation mit dem Automation License Manager übertragen.

In WinCC benötigen Sie für Folgendes einen License Key:

- WinCC Engineering System
- WinCC Runtime
- Optionen für WinCC Engineering System

Lizenzen für WinCC und die WinCC Optionen übertragen Sie nach der Installation mit dem Automation License Manager.

Hinweis

Der Lizenznehmer bestätigt weiterhin, dass die Software (SW) von der Microsoft Corporation oder Tochtergesellschaften lizenzierte Software beinhaltet. Hiermit stimmt der Lizenznehmer zu, an die Geschäftsbedingungen des beigefügten Lizenzvertrags zwischen Microsoft SQL Server und Endabnehmer gebunden zu sein und diese zu erfüllen.

Lizenzen

Folgende Lizenzen mit den entsprechenden License Keys stehen Ihnen zur Verfügung:

- STEP 7 Basic
- STEP 7 Professional
- STEP 7 Professional Combo
- WinCC Basic
- WinCC Comfort
- WinCC Comfort Combo
- WinCC Advanced
- WinCC Advanced Combo
- WinCC Professional
- WinCC Professional Combo
- WinCC RT Advanced
- WinCC RT Professional

Gültigkeit der License Keys für ältere Versionen von STEP 7

Mit einem gültigen License Key für die V15.x von STEP 7 Professional und STEP 7 Professional Combo können auch ältere Versionen von STEP 7 ohne Einschränkungen betrieben werden. Genauere Informationen dazu entnehmen Sie der nachfolgenden Tabelle:

Edition	Lizenz	Gültig für
STEP 7 Basic V15.x	STEP 7 Basic	• STEP 7 Basic V15.x • STEP 7 Basic V14.x • STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5
STEP 7 Professional V15.x	STEP 7 Professional	• STEP 7 Professional V15.x • STEP 7 Professional V14.x • STEP 7 Professional V13.x • STEP 7 Professional V12.x • STEP 7 Professional V11.x • STEP 7 Basic V14.x • STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5
STEP 7 Professional V15.x	STEP 7 Professional Combo	• STEP 7 Professional V15.x • STEP 7 Professional V14.x • STEP 7 Professional V13.x • STEP 7 Professional V12.x • STEP 7 Professional V11.x • STEP 7 Basic V14.x • STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5 • STEP 7 V5.6 • STEP 7 V5.5 • STEP 7 V5.4 • STEP 7 Professional 2017 • STEP 7 Professional 2010 • STEP 7 Professional 2006

Gültigkeit der License Keys für ältere Versionen von WinCC

Mit einem gültigen License Key für die WinCC Professional V15.x können auch ältere Versionen von WinCC ohne Einschränkungen betrieben werden.

Genauere Informationen entnehmen Sie der nachfolgender Tabelle:

WinCC Engineering System

Edition	Lizenz	Gültig für
WinCC Basic V15.x	WinCC Basic	• WinCC Basic V11.x • WinCC Basic V12.x • WinCC Basic V13.x • WinCC Basic V14.x • WinCC Basic V15.x
WinCC Comfort/Advanced V15.x	WinCC Comfort	• WInCC Basic V11.x • WinCC Comfort V10.x • WinCC Comfort V11.x • WinCC Basic, Comfort V12.x • WinCC Basic, Comfort V13.x • WinCC Basic, Comfort V14.x • WinCC Basic, Comfort V15.x
	WinCC Comfort Combo	• WinCC flexible Compact, Standard >= 2005 • WInCC Basic V11.x • WinCC Comfort V10.x • WinCC Comfort V11.x • WinCC Basic, Comfort V12.x • WinCC Basic, Comfort V13.x • WinCC Basic, Comfort V14.x • WinCC Basic, Comfort V15.x
	WinCC Advanced	• WInCC Basic V11.x • WinCC Comfort, Advanced V10.x • WinCC Comfort, Advanced V11.x • WinCC Basic, Comfort, Advanced V12.x • WinCC Basic, Comfort, Advanced V13.x • WinCC Basic, Comfort, Advanced V14.x • WinCC Basic, Comfort, Advanced V15.x
	WinCC Advanced Combo	• WinCC flexible Compact, Standard, Advanced >= 2005 • WInCC Basic V11.x • WinCC Comfort, Advanced V10.x • WinCC Comfort, Advanced V11.x • WinCC Basic, Comfort, Advanced V12.x • WinCC Basic, Comfort, Advanced V13.x • WinCC Basic, Comfort, Advanced V14.x • WinCC Basic, Comfort, Advanced V15.x

Edition	Lizenz	Gültig für
WinCC Professional V15.x	WinCC Professional (512) WinCC Professional (4096) WinCC Professional (max.)	• WinCC Comfort, Advanced, Professional V10.x • WinCC Comfort, Advanced, Professional V11.x • WinCC Basic, Comfort, Advanced, Professional V12.x • WinCC Basic, Comfort, Advanced, Professional V13.x • WinCC Basic, Comfort, Advanced, Professional V14.x • WinCC Basic, Comfort, Advanced, Professional V15.x
	WinCC Professional (512) Combo WinCC Professional (4096) Combo WinCC Professional (max.) Combo	• WinCC flexible Compact, Standard, Advanced >= 2008 SP3 • WinCC SCADA V7.0 SP3 ES • WinCC Comfort, Advanced, Professional V10.x • WinCC Comfort, Advanced, Professional V11.x • WinCC Basic, Comfort, Advanced, Professional V12.x • WinCC Basic, Comfort, Advanced, Professional V13.x • WinCC Basic, Comfort, Advanced, Professional V14.x • WinCC Basic, Comfort, Advanced, Professional V15.x

WinCC Runtime Advanced and Runtime Professional

Edition	Lizenz	Gültig für
WinCC RT Advanced V15.x	WinCC RT Advanced (128) WinCC RT Advanced (512) WinCC RT Advanced (2048) WinCC RT Advanced (4096) WinCC RT Advanced (8192) WinCC RT Advanced (16384)	• WinCC RT Advanced V11.x • WinCC RT Advanced V12.x • WinCC RT Advanced V13.x • WinCC RT Advanced V14.x • WinCC RT Advanced V15.x
WinCC RT Professional V15.x	WinCC RT Professional (128) WinCC RT Professional (512) WinCC RT Professional (2048) WinCC RT Professional (4096) WinCC RT Professional (8192) WinCC RT Professional (65536) WinCC RT Professional (102400) WinCC RT Professional (153600) WinCC RT Professional (262144)	• WinCC RT Advanced V11.x • WinCC RT Advanced V12.x • WinCC RT Advanced V13.x • WinCC RT Advanced V14.x • WinCC RT Professional V11.x • WinCC RT Professional V12.x • WinCC RT Professional V13.x • WinCC RT Professional V14.x • WinCC RT Professional V15.x

Gültigkeit der Lizenzen für WinCC Optionen

WinCC Optionen besitzen in der Regel versionsunabhängige License Keys. Sie können WinCC Optionen V15.x mit V11, V12, V13 und V14 License Keys verwenden und umgekehrt.

Starten ohne gültigen License Key

Wenn Sie das TIA Portal ohne gültigen License Key starten, weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin. Sie haben die Möglichkeit, einmalig eine Trial License zu aktivieren. Diese hat jedoch nur eine begrenzte Gültigkeit und läuft nach 21 Tagen ab.

Wenn die Trial License abgelaufen ist, können folgende Fälle auftreten:

- Auf dem betreffenden PC war das TIA Portal noch nie lizenziert:
  - Im TIA Portal lassen sich keine lizenzpflichtigen Aktionen mehr durchführen.
- Auf dem betreffenden PC war das TIA Portal bereits lizenziert:
  - Der nicht lizenzierte Modus wird alle 10 Minuten und bei jeder lizenzpflichtigen Aktion durch ein quittierpflichtiges Fenster angezeigt.

Lizenzbedarf bei Simulation

Wenn Sie in STEP 7 die Simulation über den Menübefehl "Online > Simulation" starten, benötigen Sie keine weiteren Lizenzen.

Wenn folgende Bedingungen erfüllt sind, benötigen Sie auch bei der Simulation die entsprechenden Lizenzen für von Ihnen installierte Edition von STEP 7:

- Die Engineering Station ist mit einer PLC verbunden.
- Die Verbindung zur PLC ist projektiert und aktiv.

Wenn Sie in WinCC die Simulation über den Menübefehl "Online > Simulation > Mit Variablen-Simulator" starten, benötigen Sie keine Lizenzen für WinCC Runtime oder für lizenzpflichtige Optionen.

Wenn folgende Bedingungen erfüllt sind, benötigen Sie auch bei der Simulation die entsprechenden Lizenzen für WinCC Runtime und für lizenzpflichtige Optionen:

- Die Engineering Station ist mit einer PLC verbunden.
- Die Verbindung zur PLC ist projektiert und aktiv.
- Sie starten den Simulator über den Menübefehl "Online > Simulation > starten".

Lizenzen für den TIA Portal Cloud Connector

Für das Arbeiten mit dem TIA Portal Cloud Connector benötigen Sie auf jedem Gerät, das Sie im TIA Portal Cloud Connector als "Benutzergerät" festlegen, einen gültigen License Key. Der License Key ist auch dann erforderlich, falls auf diesem Gerät das TIA Portal installiert ist. Für Geräte, die Sie als "Remotegerät" verwenden, ist kein License Key erforderlich.

Siehe auch

Umgang mit Lizenzen und License Keys (Seite 17)

3.3 Umgang mit Lizenzen und License Keys

Einleitung

Zur Nutzung von STEP 7 Basic, STEP 7 Professional sowie WinCC Engineering System, Optionen für WinCC Engineering System und WinCC Runtime benötigen Sie jeweils einen gültigen License Key.

ACHTUNG
Zerstörung von License Keys durch Kopieren Ein License Key ist nicht kopierbar. Das verwendete Kopierschutzverfahren verhindert das Kopieren der License Keys. Beim Versuch, einen License Key zu kopieren, wird dieser zerstört!

Trial-Lizenz

Wenn Sie das TIA Portal ohne gültigen License Key starten, weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin. Sie haben die Möglichkeit, einmalig eine Trial-Lizenz zu aktivieren. Die Trial-Lizenz läuft nach 21 Tagen ab.

Wenn die Trial-Lizenz abgelaufen ist, können folgende Fälle auftreten:

- Auf dem betreffenden PC war das TIA Portal noch nie lizenziert.
  Das TIA Portal lässt sich nicht mehr starten.
- Auf dem betreffenden PC war das TIA Portal bereits lizenziert.
  Das TIA Portal lässt sich starten. Der nicht lizenzierte Modus wird alle 10 Minuten durch ein quittierpflichtiges Fenster angezeigt.

Installation von License Keys für STEP 7

Bei der Installation von STEP 7 Basic wird der benötigte License Key vom Setup automatisch mit installiert. Bei der Installation von STEP 7 Professional werden Sie am Ende des Setups aufgefordert, die Lizenz mithilfe des Automation License Manager vom mitgelieferten Datenträger auf Ihren PC zu übertragen.

Falls Sie zusätzliche License Keys installieren möchten, müssen Sie dazu den Automation License Manager nutzen.

Wenn Sie eine Lizenz installieren, wird der zugehörige License Key vom Ablageort der License Keys entfernt.

Hinweis

Behalten Sie nach der Installation auf jeden Fall den Original-Ablageort der License Keys.

Installation von License Keys für WinCC

Bei der Installation von WinCC Basic wird der benötigte License Key vom Setup automatisch mit installiert. Bei der Installation von WinCC Professional werden Sie am Ende des Setups aufgefordert, die Lizenz mithilfe des Automation License Manager vom mitgelieferten Datenträger auf Ihren PC zu übertragen.

Zur Nutzung von WinCC Runtime oder der Simulation auf dem Engineering-PC über den Menübefehl "Online > Simulation > starten" benötigen Sie zusätzliche License Keys. Sie müssen dazu den Automation License Manager nutzen.

Wenn Sie eine Lizenz installieren, wird der zugehörige License Key vom Ablageort der License Keys entfernt.

Hinweis

Behalten Sie nach der Installation auf jeden Fall den Original-Ablageort der License Keys.

Übertragen von License Keys auf Bediengeräte

Für den Betrieb von WinCC müssen Sie die License Keys auf das Bediengerät übertragen.

Sie übertragen einen License Key in folgenden Fällen auf das Bediengerät:

- Zur Nutzung vom WinCC Runtime
- Zur Nutzung von Optionen von WinCC Runtime

Wenn Sie eine Lizenz auf ein Bediengerät übertragen, wird der zugehörige License Key vom Ablageort der License Keys entfernt.

Wenn Sie die Lizenz nicht mehr benötigen oder Daten sichern möchten, müssen Sie die License Keys vom Bediengerät übertragen. Sie können diese Lizenz an einem anderen PC oder Bediengerät verwenden.

Deinstallation von License Keys

Die Deinstallation von License Keys erfolgt immer mit dem Automation License Manager. Sie deinstallieren einen License Key in folgenden Fällen:

- Wenn Sie Daten sichern.
- Wenn Sie die Lizenz nicht mehr benötigen.

Sie können eine gültige Lizenz auch an einem anderen PC verwenden.

Modifikation an den Datenträgern des Engineering Systems

Durch Modifikationen der Datenträger des Engineering Systems kann der License Key zerstört werden. Rufen Sie vor derartigen Modifikationen den Automation License Manager auf und sichern Sie den zu deinstallierenden License Key auf einen anderen Ablageort.

ACHTUNG
Zerstörung von License Keys auf PCs Deinstallieren Sie zuerst alle License Keys, wenn einer der folgenden Fälle zutrifft: • Formatieren der Festplatte • Komprimieren der Festplatte • Restaurieren der Festplatte • Starten eines Optimierungsprogramms, das feste Blöcke verschiebt • Installation eines neuen Betriebssystems Lesen Sie die Beschreibung zu Automation License Manager ("Start > Siemens Automation > Dokumentation"). Beachten Sie alle Warnhinweise und Hinweise.

Datensicherung von Bediengeräten (Backup/Restore)

Zur Datensicherung am Bediengerät deinstallieren Sie die License Keys am Bediengerät. Rufen Sie dazu den Automation License Manager auf und sichern Sie den zu deinstallierenden License Key auf einen anderen Ablageort.

ACHTUNG
Zerstörung von License Keys auf nicht PC-basierten Bediengeräten Bei folgenden Bediengeräten werden übertragene License Keys durch Backup-/Restore-Vorgänge zerstört: • 270er Reihe • 370er Reihe Bevor Sie mit Wiederherstellen beginnen, führen Sie Folgendes aus: • Überprüfen Sie mit dem Automation License Manager und ProSave, ob auf dem Bediengerät License Keys vorhanden sind. • Übertragen Sie die vorhandenen License Keys vom Bediengerät Nach dem Wiederherstellen übertragen Sie die License Keys wieder auf das Bediengerät.

Ungültige Lizenz nach Zeitzonenwechsel

Die installierten Lizenzen funktionieren nicht mehr, wenn Sie auf einem PC die Zeitzone folgendermaßen umstellen: Von einer auf die volle Stunde bezogenen Zeitzone auf eine Zeitzone, die sich nicht auf eine volle Stunde bezieht.

Um dieses Verhalten zu umgehen, deinstallieren Sie den License Key mit dem Automation License Manager mit einer Zeitzoneneinstellung, die bei der Installation des License Keys eingestellt war.

Beispiel: Sie möchten das Bediengerät aus der Zeitzone "GMT +3:00" in der Zeitzone "GMT +3:30" einsetzen.

1. Deinstallieren Sie den License Key mit der Zeitzoneneinstellung "GMT +3:00".
2. Ändern Sie die Zeitzoneneinstellungen entsprechend auf Ihrem PC.
3. Installieren Sie den License Key.

Das Verhalten betrifft nicht die Trial License.

Defekte Lizenz

Eine Lizenz ist in folgenden Fällen defekt:

- Wenn der License Key auf dem Ablageort nicht mehr zugänglich ist.
- Wenn der License Key beim Übertragen auf das Ziellaufwerk verschwindet.

Hinweis

Wenn Sie das Systemdatum auf einen früheren Zeitpunkt zurücksetzen, werden alle Lizenzen defekt.

Sie können die defekte Lizenz mithilfe von Automation License Manager reparieren lassen. Benutzen Sie dazu die Funktion "Wiederherstellen" bzw. den "Wiederherstellen-Assistenten" des Automation License Manager. Für die Wiederherstellung nehmen Sie Kontakt mit dem Customer Support auf. Nähere Hinweise hierzu finden Sie im Internet unter: http://support.automation.siemens.com (http://support.automation.siemens.com)

Hinweis

Ein störungsfreier Betrieb von Runtime ist auch bei fehlender bzw. defekter Lizenz sichergestellt. In kurzen Zeitabständen weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin.

Microsoft SQL Server 2014

Für die Nutzung der Datenbank Microsoft SQL Server ist eine Lizenz erforderlich. Diese Lizenz liegt bei einer lizenzrechtlich ordnungsgemäßen Installation von WinCC vor.

Der mit der Installation von WinCC lizenzierte SQL Server dart nur in Verbindung mit WinCC eingesetzt werden.

Der Einsatz für andere Zwecke erfordert eine zusätzliche Lizenz. Dazu gehören z. B.:

- Einsatz für eigene Datenbanken
- Einsatz für Fremdapplikationen
- Einsatz von SQL Zugriffsmechanismen, die nicht über WinCC zur Verfügung gestellt werden

Deinstallation

Nach der Deinstallation von WinCC müssen Sie die SQL Server-Instanz "WinCC" deinstallieren. Wählen Sie unter “Systemsteuerung > Software" den Eintrag "Microsott SQL Server 2014” zur Deinstallation.

Siehe auch

Lizenzierung von STEP 7 und WinCC (Seite 11)

3.4 Lizenzierung von Bediengeräten

Nicht PC-basierte Bediengeräte werden mit Runtime-Lizenzen ausgestattet. Ein License Key für den Runtimebetrieb ist nicht erforderlich.

Zu einer Option für nicht PC-basierte Bediengeräte kann eine Lizenz erforderlich sein. Der License Key der jeweiligen Lizenz schaltet jeweils eine Option zur Nutzung frei.

License Key

Um nicht PC-basierte Bediengeräte mit License Keys lizenzieren zu können, ist das Add-On "SIMATIC HMI License Manager Panel Plugin" notwendig.

Das WinCC Setup installiert standardmäßig dieses Add-On. Das License Manager Panel Plugin öffnen Sie im Automation License Manager über den Menübefehl "Bearbeiten > Zielsystem verbinden > Bediengerät verbinden".

Wenn WinCC nicht installiert ist, ist eine Installation von ProSave 7.2 oder höher notwendig.

Hinweise zum Umgang mit Lizenzen

- Weitere Informationen zur Handhabung der Lizenzen finden Sie in der Hilfe des Automation License Manager.
- Vergewissern Sie sich, bevor Sie mit Lizenzierung beginnen, dass das Bediengerät mindestens mit einem aktuellen Stand des Betriebssystems ausgestattet ist. Gegebenenfalls aktualisieren Sie das Betriebssystem mit ProSave.
- Der Lizenznehmer bestätigt weiterhin, dass die Software (SW) von der Microsoft Corporation oder Tochtergesellschaften lizenzierte Software beinhaltet. Hiermit stimmt der Lizenznehmer zu, an die Geschäftsbedingungen des beigefügten Lizenzvertrags zwischen Microsoft SQL Server und Endabnehmer gebunden zu sein und diese zu erfüllen.

Nicht lizenzierter Modus

Die Runtime-Optionen sind ohne gültige Lizenz vorübergehend ohne Einschränkung der Funktionalität nutzbar. Der nicht lizenzierte Modus wird alle 10 Minuten durch ein quittierpflichtiges Fenster angezeigt.

3.5 Lizenzierung von WinCC Runtime auf PC-basierten Bediengeräten

License Key

Bei PC-basierten Bediengeräten benötigen Sie für Folgendes einen License Key:

- WinCC Runtime, z. B. mit 128 Variablen
- WinCC Optionen

Sie können den License Key für WinCC Runtime gleich bei der Installation von WinCC mitübertragen. Lizenzen für die WinCC Optionen übertragen Sie nach der Installation mit dem Automation Licence Manager.

Hinweis

Nur das Certificate of License für WinCC Runtime V15.1 berechtigt zum Betrieb von WinCC Runtime V15.1.

Um WinCC Runtime zu lizenzieren, benötigen Sie die entsprechende, neue Lizenz.

Die Runtime-Lizenzen von WinCC flexible 2008 und von WinCC V7 können Sie mit einem Upgrade auf WinCC Runtime V15.1 hochrüsten.

Nur mit einem gültigen Certificate of License, entsprechend der darin genannten Version, ist der produktive Betrieb der Software zulässig.

Hinweise zum Umgang mit Lizenzen

- Wenn Sie von einer älteren Version auf die Version V15 hochrüsten, benötigen Sie neue Lizenzen für WinCC Runtime Professional.
- Der Lizenznehmer bestätigt weiterhin, dass die Software (SW) von der Microsoft Corporation oder Tochtergesellschaften lizenzierte Software beinhaltet. Hiermit stimmt der Lizenznehmer zu, an die Geschäftsbedingungen des beigefügten Lizenzvertrages zwischen Microsoft SQL Server und Endabnehmer gebunden zu sein und diese zu erfüllen.

Nicht lizenzierter Modus

WinCC Runtime und die Runtime-Optionen sind ohne gültige Lizenz uneingeschränkt nutzbar. Der nicht lizenzierte Modus wird alle 10 Minuten durch ein quittierpflichtiges Fenster angezeigt.

3.6 Lizenzierung WinCC Runtime Professional asiatisch

Einleitung

Wenn Sie WinCC Runtime Professional in einer asiatischen Sprache verwenden, wird für die Gültigkeitsprüfung der erforderlichen Lizenz zusätzlich ein Dongle eingesetzt. Der Dongle wird in Form eines zertifizierten USB-Sticks zusammen mit dem erforderlichen Licence Key für WinCC Runtime Professional asiatisch ausgeliefert. Der License Key ist auf dem zertifizierten USB-Stick gespeichert.

Folgende asiatischen Sprachen sind durch einen Dongle geschützt:

- Chinesisch vereinfacht
- Chinesisch traditionell
- Japanisch
- Koreanisch

Einsatz des Dongle für eine asiatische Runtime Professional

Um ein WinCC-Projekt in einer asiatischen Sprache erfolgreich zu lizenzieren, übertragen Sie den erforderlichen License Key mithilfe des "Automation License Manager" von dem zertifizierten USB-Stick auf das Bediengerät. Stecken Sie den USB-Stick in den USB-Anschluss des Bediengeräts. Der USB-Stick wirkt während Runtime als Dongle. Der Dongle muss in Runtime immer im USB-Anschluss gesteckt bleiben.

Wenn Sie den Dongle in Runtime entfernen, wechselt WinCC Runtime in den Demo-Modus.

Die für die asiatischen Märkte gelieferten USB-Sticks sind für den Dauereinsatz in Produktionsanlagen zertifiziert. Den Inhalt können Sie daher nicht auf einen anderen USB-Datenträger kopieren oder transferieren. Bei Verlust oder Defekt eines zertifizierten USB-Sticks benötigen Sie einen neuen, zertifizierten USB-Stick.

Simulation eines WinCC-Projekts in einer asiatischen Sprache

Für die Simulation benötigen Sie dann ebenfalls einen zertifizierten USB-Stick als Dongle. Wenn kein Dongle verfügbar ist, läuft die Simulation im Demo-Modus.

3.7 STEP 7 ohne Lizenz für PC-CP

Funktionen von STEP 7 für PC-CP

STEP 7 für PC-CP ist eine vollständige STEP 7Version, die ohne Lizenz installiert wurde. In dieser Version stehen Ihnen nur Konfigurations- und Diagnose-Möglichkeiten für folgende Komponenten zur Verfügung:

- PC-CPs und Applikationen (OPC-Server) für PC-Stationen
- Switches SCALANCE X
- Router SCALANCE W / SCALANCE M
- Security-Module SCALANCE S
- PROFINET IO-Devices und PROFIBUS DP-Slaves
- IE/PB LINK PN IO, IWLAN/PB LINK PN IO
- Reader (RFID) und optische Lesegeräte (MV)

SIMATIC-Stationen und WinAC-Komponenten stehen nur in der Netz- und Topologie-Ansicht zur Verfügung (keine Geräte-Ansicht). Programmierfunktionen stehen nicht zur Verfügung.

Informationssystem

Das in der STEP 7-Version für PC-CP vorhandene Informationssystem beschreibt den vollen Funktionsumfang von STEP 7. Beachten Sie daher die Beschränkung auf die Funktionen zur Konfiguration der zuvor genannten Gerätegruppen.

Auf Funktionsumfang von STEP 7 Professional hochrüsten

Mit der Installation von STEP 7 für PC-CP ist bereits der volle Funktionsumfang von STEP 7 auf Ihrer Engineering-Station vorhanden, aber nur eingeschränkt nutzbar.

Durch eine nachträgliche Lizenzinstallation können Sie jederzeit auf die volle Funktionalität von STEP 7 Professional hochrüsten. Eine erneute Installation ist nicht erforderlich. Zuvor in STEP 7 für PC-CP angelegte Projekte können nach der Installation einer Lizenz ohne Einschränkung weiterverwendet werden.

Systemvoraussetzungen für die Installation 4

4.1 Hinweise zu den Systemvoraussetzungen

Systemvoraussetzungen für einzelne Produkte

Je nachdem, welche Produkte Sie installieren möchten, gelten unterschiedliche Systemvoraussetzungen. Beachten Sie daher die individuellen Systemvoraussetzungen Ihrer Produkte.

Möchten Sie mehrere Produkte installieren, berücksichtigen Sie bitte die Systemvoraussetzungen des Produkts mit den höchsten Anforderungen.

Anzeigen von PDF-Dateien

Um mitgelieferte PDF-Dateien zu lesen, benötigen Sie einen PDF Reader, der das PDF-Format 1.7 unterstützt, z. B. Adobe (R) Reader Version 9.

Anzeigen der Welcome Tour

Um die Welcome Tour zum TIA Portal zu starten, benötigen Sie den Adobe (R) Flashplayer ab Version 9.

Windows 7 und Windows Server 2008 R2: Ungültige Zertifikatsvalidierung

Die neuen, zur Signierung verwendeten Zertifikate werden auf den Betriebssystemen Windows 7 und Windows Server 2008 R2 möglicherweise nicht validiert.

Wenn Sie keine Möglichkeit haben, den Zertifikatsspeicher online zu aktualisieren, wird gegebenenfalls die Signatur der Produktdateien als ungültig angezeigt. Aktualisieren Sie in diesem Fall auf den Offline-Systemen die von Microsoft bereitgestellten Root Zertifikate. Folgen Sie dafür den Anweisungen im FAQ 109748887 (https://support.industry.siemens.com/cs/ww/de/view/109748887).

Hinweis für Installationen unter Windows 7

Das TIA Portal ist nur auf Windows-Systemen lauffähig, die den sicheren Hash-Algorithmus "SHA-2" unterstützen. Die Unterstützung von "SHA-2" unter Windows 7 ist erst nach der Installation eines Sicherheitsupdates (KB 3033929) gegeben. Stellen Sie daher sicher, dass die neuesten Sicherheitsupdates für Windows auf Ihrem System installiert sind.

Siehe auch

Hinweise zu Lizenzen (Seite 9)

Installation starten (Seite 41)

Installierte Software anzeigen (Seite 45)
Installierte Produkte modifizieren oder hochrüsten (Seite 47)
Installierte Produkte reparieren (Seite 49)
Deinstallation starten (Seite 51)

4.2 Allgemeine Software- und Hardwarevoraussetzungen

Voraussetzungen für die Installation

Die folgende Tabelle zeigt die minimalen Software- und Hardwarevoraussetzungen, die für die Installation mindestens erfüllt sein müssen:

Hardware/Software	Voraussetzung
Prozessor	Intel® Core™ i3-6100U, 2.30 GHz
RAM	8 GB
Festplatte	S-ATA mit mindestens 20 GB verfügbarem Speicherplatz
Netzwerk	Ab 100 Mbit
Bildschirmauflösung	1024 x 768
Betriebssysteme *	**Windows 7 (64 Bit)** • Windows 7 Home Premium SP1 ** • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 10 (64 Bit)** • Windows 10 Home Version 1709 ** • Windows 10 Home Version 1803 ** • Windows 10 Professional Version 1709 • Windows 10 Professional Version 1803 • Windows 10 Enterprise Version 1709 • Windows 10 Enterprise Version 1803 • Windows 10 Enterprise 2016 LTSB • Windows 10 IoT Enterprise 2015 LTSB • Windows 10 IoT Enterprise 2016 LTSB **Windows Server (64 Bit)** • Windows Server 2012 R2 StdE (Vollinstallation) • Windows Server 2016 Standard (Vollinstallation)

* Inklusive aller Security Updates bis zum Microsoft Security Bulletin Summary für September 2018 (MS18-Sep). Weitere Informationen zu den Betriebssystemen finden Sie in der Hilfe zu Microsoft Windows oder auf der Homepage von Microsoft.

** Nur für Basic-Editionen

Empfohlene PC-Hardware

Die folgende Tabelle zeigt die empfohlenen Hardwarevoraussetzungen:

Hardware/Software	Voraussetzung
Rechner	Ab SIMATIC FIELD PG M5 Advanced (oder vergleichbarer PC)
Prozessor	Intel® Core™ i5-6440EQ (bis zu 3.4 GHz)
RAM	16 GB oder mehr (für große Projekte 32 GB)

Hardware/Software	Voraussetzung
Festplatte	SSD mit mindestens 50 GB verfügbarem Speicherplatz
Netzwerk	1 GBit (für Multiuser)
Bildschirm	15,6" Full HD Display (1920 x 1080 oder größer)

Unterstützte Virtualisierungsplattformen

Sie können die Softwarepakete "SIMATIC STEP 7" und "SIMATIC WinCC" in einer virtuellen Maschine installieren. Verwenden Sie dazu eine der folgenden Virtualisierungsplattformen in der angegebenen oder einer neueren Version:

- VMware vSphere Hypervisor (ESXi) 6.5
- VMware Workstation 12.5.7
- VMware Player 12.5.7
- Microsoft Hyper-V Server 2016

Für diese Virtualisierungsplattformen werden die folgende Betriebssysteme als Host-Betriebssystem empfohlen:

Betriebssystem	VMware vSphere Hypervisor (ESXi)	VMware Workstation	VMware Player	Microsoft Hyper-V
Windows Server 2016 Standard (Vollinstallation, 64-Bit)	-	-	-	x
Windows 7 Professional SP1	x	x	x	-
Windows 7 Enterprise SP1	x	x	x	-
Windows 7 Ultimate SP1	x	x	x	-
Windows 10 Professional Version 1703	x	x	x	-
Windows 10 Enterprise Version 1703	x	x	x	-
Windows 10 Enterprise 2016 LTSB	x	x	x	-

x: als Host-Betriebssystem verwendbar

-: nicht als Host-Betriebssystem verwendbar

Innerhalb der gewählten Virtualisierungsplattform werden die folgenden Gast-Betriebssysteme empfohlen:

Betriebssystem	VMware vSphere Hypervisor (ESXi)	VMware Workstation	VMware Player	Microsoft Hyper-V
Windows Server 2012 R2 StdE (Vollinstallation, 64-Bit)	-	-	-	x
Windows Server 2016 Standard (Vollinstallation, 64-Bit)	-	-	-	x
Windows 7 Professional SP1	x	x	x	x
Windows 7 Enterprise SP1	x	x	x	x
Windows 7 Ultimate SP1	x	x	x	x
Windows 10 Professional Version 1703	x	x	x	x

Betriebssystem	VMware vSphere Hypervisor (ESXi)	VMware Workstation	VMware Player	Microsoft Hyper-V
Windows 10 Enterprise Version 1703	x	x	x	x
Windows 10 Enterprise 2016 LTSB	x	x	x	x

x: als Gast-Betriebssystem verwendbar

-: nicht als Gast-Betriebssystem verwendbar

Hinweis

- Für die Gast-Betriebssysteme gelten dieselben Hardware-Anforderungen wie für die jeweiligen TIA-Produkte selbst.
- Der Betreiber der Anlage muss dafür Sorgen, dass den Gast-Betriebssystemen ausreichend Systemressourcen zur Verfügung gestellt werden.
- Für den Einsatz von HyperV-Server und ESXi wird die von den Herstellern zertifizierte Hardware empfohlen.
- Bei der Verwendung von Microsoft Hyper-V ist das Anzeigen der erreichbaren Teilnehmer nicht möglich.

Unterstützte Sicherheitsprogramme

Folgende Sicherheitsprogramme sind kompatibel zu "SIMATIC STEP 7 und "SIMATIC WinCC":

- Virenscanner:
  - Symantec Endpoint Protection 14
  - Trend Micro Office Scan Corporate Edition 12.0
  - McAfee VirusScan Enterprise 8.8
  - Kaspersky Anti-Virus 2017
  - Windows Defender
  - Qihoo 360 "Safe Guard 11.0" + "Virus Scanner 5.0"

  Hinweis

  Achten Sie darauf, dass Ihr Virenscanner und dessen Datenbanken immer auf dem aktuellen Stand sind.

  Die letzte Aktualisierung der Viren-Pattern erfolgte am 01.10.2018.
- Verschlüsselungssoftware:
  - Microsoft Bitlocker
- Host-basiertes Intrusion Detection System:
  - McAfee Application Control 8.0

Siehe auch

Hinweise zu Lizenzen (Seite 9)
Installation starten (Seite 41)
Verfügbarkeit von Updates und Support Packages prüfen und installieren (Seite 53)
Installierte Software anzeigen (Seite 45)
Installierte Produkte modifizieren oder hochrüsten (Seite 47)
Installierte Produkte reparieren (Seite 49)
Deinstallation starten (Seite 51)

4.3 Produktspezifische Besonderheiten

4.3.1 Installation von WinCC Runtime Professional

Installation in Domänen und Arbeitsgruppen

WinCC ist grundsätzlich für den Betrieb in einer Domäne oder Arbeitsgruppe freigegeben.

Beachten Sie jedoch, dass Domänen-Gruppenrichtlinien und Einschränkungen der Domäne die Installation behindern können. Entfernen Sie in diesem Fall den Rechner vor der Installation von Microsoft Message Queuing, Microsoft SQL Server und WinCC aus der Domäne. Melden Sie sich an dem betreffenden Rechner lokal mit Administratorrechten an. Führen Sie die Installation durch. Nach der erfolgreichen Installation können Sie den WinCC-Rechner wieder in die Domäne aufnehmen. Wenn die Domänen-Gruppenrichtlinien und Einschränkungen der Domäne die Installation nicht behindern, muss der Rechner während der Installation nicht aus der Domäne entfernt werden.

Beachten Sie, dass Domänen-Gruppenrichtlinien und Einschränkungen der Domäne auch den Betrieb behindern können. Wenn Sie diese Einschränkungen nicht aufheben können, betreiben Sie die WinCC-Rechner in einer Arbeitsgruppe.

Wenden Sie sich gegebenenfalls an den Domänen-Administrator.

Windows Rechnername

Ändern Sie den Windows Rechnernamen nach der WinCC-Installation nicht.

Wenn Sie den Windows Rechnernamen ändern, müssen Sie den SQL-Server deinstallieren und neu installieren.

Folgende Zeichen sind im Rechnernamen nicht zulässig:

- . , ; : ! ? " ' ^ ´ ` ~ _
- + = / \ ¦ @ * # $ % & § °
- ( ) [ ] { } < >
- Leerzeichen

Bei der Vergabe der Windows Rechnernamen beachten Sie Folgendes:

- Nur Großschreibung darf verwendet werden.
- Erstes Zeichen muss ein Buchstabe sein.
- Die ersten 12 Zeichen des Rechnernamens müssen eindeutig sein.

4.3.2 Deinstallation einer SQL-Instanz

Inhalt

Mit der Installation von WinCC V15.1 wird eine neue WINCC SQL 2014 Instanz installiert.

Wenn Sie bereits eine Version von WinCC Runtime Professional V13 oder WinCC Runtime Professional V13 SP1 installiert hatten, müssen Sie vor der Installation von WinCC Professional V15.1 entweder

- zuerst WinCC Runtime Professional V15.1 installieren, oder
- die Instanz "WINCC" des SQL Server 2008 R2 deinstallieren.

Wenn Sie eine ältere Version als V13 von WinCC Runtime Professional installiert hatten, müssen Sie die WinCC-Instanz des SQL Server 2008 R2 bzw. die WinCC-Instanz des SQL Server 2005 deinstallieren.

Hinweis

Durch eine Deinstallation von WinCC Runtime Professional oder von WinCC Professional wird die Instanz des SQL-Servers nicht deinstalliert.

Deinstallation einer SQL-Instanz

Um eine ältere SQL-Instanz von WinCC zu deinstallieren, gehen Sie folgendermaßen vor

1. Öffnen Sie die Systemsteuerung.
2. Klicken Sie auf Programm deinstallieren.
3. Wählen Sie in der Lister der installierten Programme "Microsoft SQL Server 2008 R2" und klicken Sie auf "Deinstallieren/ändern".
4. Klicken Sie im Dialog "SQL Server 2008 R2" auf "Remove".
5. Klicken Sie im Dialog "Setup Support Rules" auf "OK".

6. Wählen Sie im Dialog "Select Instance" die Instanz "WINCC" und klicken Sie auf "Next".

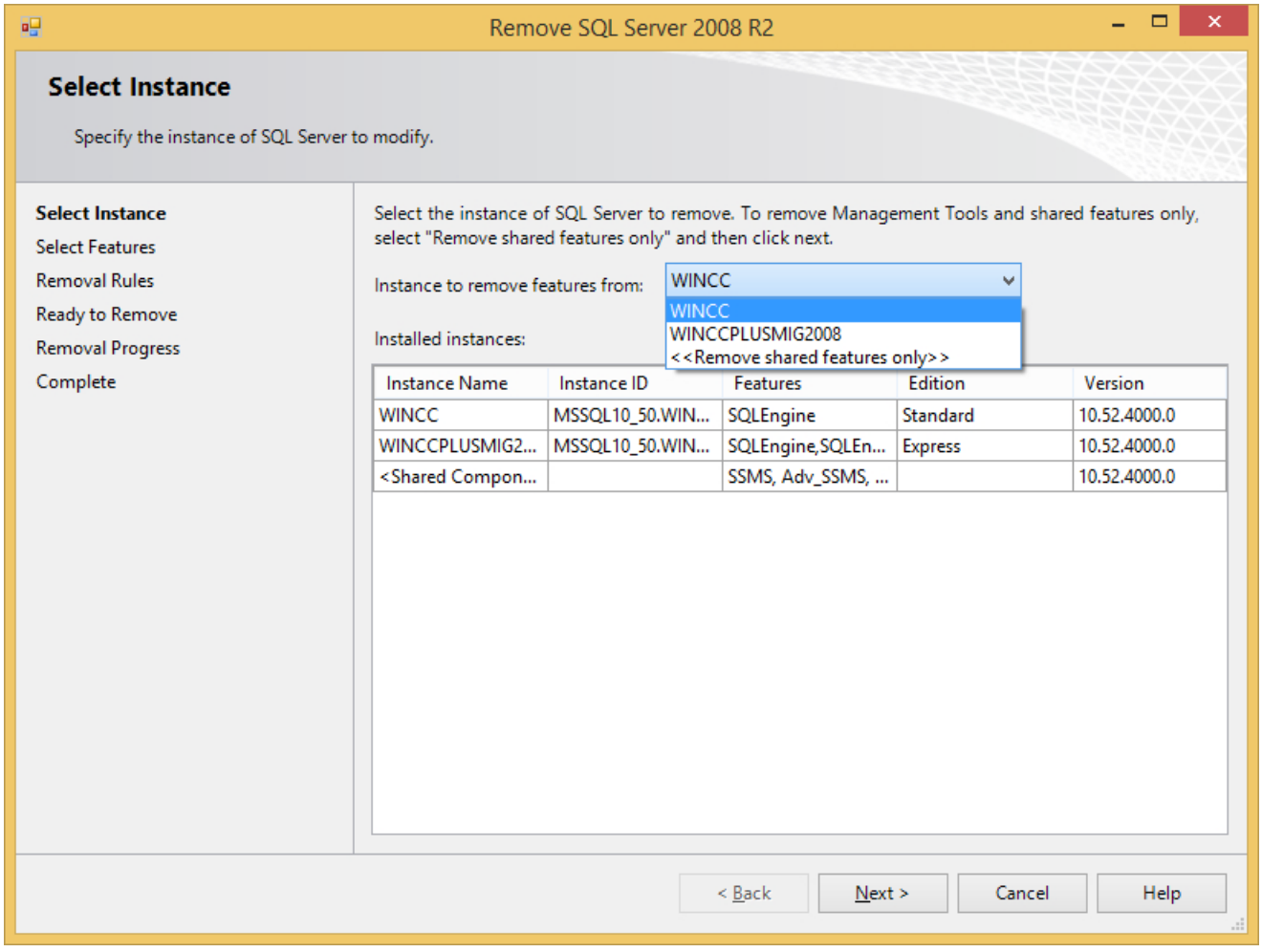

7. Klicken Sie im Dialog "Select Features" auf "Select All" und anschließend auf "Next".
8. Klicken Sie im Dialog "Removal Rules" auf "Next".
9. Klicken Sie im Dialog "Ready to Remove" auf "Remove".
10. Klicken Sie nach Abschluss der Deinstallation im Dialog "Complete" auf "Close"

Die Vorgehensweise zur Deinstallation einer Instanz des SQL Server 2005 ist vergleichbar.

4.3.3 Hinweise zur Installation des TIA Portal Teamcenter Gateway

Einführung

Sie können das TIA Portal Teamcenter Gateway wie folgt installieren:

- Zusammen mit der Installation des TIA Portals

Hinweis

Für die Serverinstallation werden Administratorrechte benötigt.

Voraussetzungen für die Installation

Auf Ihrem Computer muss/müssen

- Teamcenter Rich Application Client (nachfolgend "RAC" genannt) oder Teamcenter Client Communication System (TCCS) Version 11.2 oder höher installiert sein.

Hinweis

Wenn Sie die Installation von TCCS als "Standalone"-Installation durchgeführt haben, müssen Sie auch "Microsoft Visual C++ 2013 – Redistributable" installieren. Damit stellen Sie sicher, dass Sie eine Verbindung zwischen dem TIA Portal und TIA Portal Teamcenter Gateway möglich ist.

- eine Verbindung von FCC/FMS (FileClientCache/FileManagementSystem) zum "Teamcenter Server 11.2 oder höher" bestehen.

Hinweis

- Prüfen Sie nach einer Nachinstallation oder einem Neustart, ob eine Verbindung von FCC/FMS (FileClientCache/FileManagementSystem) zum "Teamcenter Server 11.2 oder höher" besteht.
- Prüfen Sie nach einer Nachinstallation oder einem Neustart die Registrierungsschlüssel. Weitere Informationen zur Überprüfung der Registrieungsschlüssel finden Sie in der Online-Hilfe "TIA Portal Teamcenter Gateway" im Abschnitt "TIA Portal Teamcenter Gateway installieren und deinstallieren" > "TIA Portal Teamcenter Gateway Installation prüfen".
- Prüfen Sie nach einer Nachinstallation oder einem Neustart, ob Sie einen Elementtypen in Teamcenter speichern können und ob Sie der Element-Revision ein Dataset hinzufügen können. Weitere Informationen zum Speichern von Elementtypen finden Sie in der Online-Hilfe "TIA Portal Teamcenter Gateway" im Kapitel "TIA Portal-Projekte mit dem TIA Portal Teamcenter Gateway verwalten" im Abschnitt "TIA Portal-Projekt als neues Element in Teamcenter speichern".
- Prüfen Sie nach einer Nachinstallation oder einem Neustart, ob ein Dateitransfer vom Teamcenter Rich Application Client zum Teamcenter Server möglich ist.

- Auf dem Teamcenter Server, muss das mitgelieferte Datenmodell, bestehend aus mehreren Dateien, installiert sein.

Installation des TIA Portal Teamcenter Gateway - Datenmodells

Die Installation des mitgelieferten TIA Portal Teamcenter Gateway - Datenmodells muss mit dem "Environment Manager" erfolgen.

Die Beschreibung zur Installation finden Sie auf der Produkt-DVD im Verzeichnis "\Support \Teamcenter_11\ServerSetupDocument\".

Gleiche TIA Portal-Versionen bei der Installation

Achten Sie bei der Installation von verschiedenen TIA Portal-Produkten darauf, dass Sie für die Installation die gleichen Stände der Service Packs und Updates verwenden.

Wenn Sie also z. B. TIA Portal V14 oder höher installiert haben, müssen Sie auch das TIA Portal Teamcenter Gateway V14 oder höher installieren.

Installationspfad

Verwenden Sie keine Unicode-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

Virenscanner

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscanner verhindern dies. Daher wird empfohlen, während der Installation des TIA Portal Teamcenter Gateway den Virenscanner zu deaktivieren und erst danach wieder zu aktivieren.

Installationsprotokoll 5

Funktion des Installationsprotokolls

Während folgender Installationsprozesse werden die Fortschritte in einer Datei protokolliert:

- Produkte installieren
- Bereits installierte Produkte modifizieren oder hochrüsten
- Eine bereits vorhandene Installation reparieren
- Produkte deinstallieren

Wenn während des Installationsprozesses Fehler auftreten oder Warnungen ausgegeben werden, können diese mithilfe der Protokolldatei ausgewertet werden. Dies können Sie entweder selbst tun oder Sie wenden sich an den Produkt-Support.

Ablageort der Installationsprotokolle

Die Protokolldatei ist die neueste Datei mit der Dateinamenerweiterung ".log", deren Name mit "SIA" beginnt.

Der Ablageort der Protokolldatei ist in der Umgebungsvariablen "%autinstlog%" gespeichert. Diese Umgebungsvariable können Sie in die Adressleiste des Windows Explorers eingeben, um den Ordner mit den Protokolldateien zu öffnen. Alternativ können Sie mit "CD %autinstlog %" über die Kommandozeile zum entsprechenden Verzeichnis navigieren.

Der Ablageort ist abhängig vom Betriebssystem, z. B. "C:\Programdata\Siemens\Automation \Logfiles\Setup" in englischem Windows.

Setup_Report (CAB-Datei)

Damit Sie alle notwendigen Dateien an den Produkt-Support liefern können, wird zusätzlich eine Archiv-Datei im CAB-Format abgelegt, die das Installationsprotokoll und alle anderen benötigten Dateien enthält. Dieses Archiv finden Sie unter "%autinstlog%\Reports \Setup_report.cab". Übermitteln Sie diese CAB-Datei dem Produkt-Support, wenn Sie Unterstützung bei der Installation benötigen. Damit kann der Produkt-Support schnell feststellen, ob die Installation einwandfrei durchgelaufen ist. CAB-Dateien, die bei früheren Installationsprozessen erzeugt wurden, werden mit einer Datumskennung im Verzeichnis "Reports" gespeichert.

Siehe auch

Hinweise zu Lizenzen (Seite 9)

Installation starten (Seite 41)

Verfügbarkeit von Updates und Support Packages prüfen und installieren (Seite 53)

Installierte Software anzeigen (Seite 45)

Installierte Produkte modifizieren oder hochrüsten (Seite 47)

Installierte Produkte reparieren (Seite 49)

Deinstallation starten (Seite 51)

Installation starten

6

Einführung

Die Installation der Softwarepakete wird durch ein Setup-Programm automatisch durchführt. Das Setup-Programm startet nach dem Einlegen des Installationsdatenträgers in das entsprechende Laufwerk.

Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

Vorgehen

Um die Softwarepakete zu installieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Produktsprachen wird geöffnet.
6. Wählen Sie die Sprachen für die Produktoberfläche und klicken Sie auf die Schaltfläche "Weiter".

   Hinweis

   Die Produktsprache "Englisch" wird als Basis immer installiert.

   Der Dialog zur Auswahl der Produktkonfiguration wird geöffnet.

7. Wählen Sie die Produkte, die Sie installieren möchten:
   - Wenn Sie das Programm mit einer minimalen Programmzusammenstellung installieren möchten, klicken Sie auf die Schaltfläche "Minimal".
   - Wenn Sie das Programm mit produkttypischen Programmzusammenstellungen installieren möchten, klicken Sie auf die Schaltfläche "Typisch".
   - Wenn Sie selbst auswählen möchten, welche Produkte installiert werden sollen, klicken Sie auf die Schaltfläche "Benutzerdefiniert". Aktivieren Sie anschließend die Optionskästchen für die Produkte, die Sie installieren möchten.
8. Wenn eine Verknüpfung auf dem Desktop angelegt werden soll, aktivieren Sie das Optionskästchen "Desktop-Verknüpfung anlegen".
9. Klicken Sie auf die Schaltfläche "Durchsuchen", wenn Sie das Zielverzeichnis für die Installation ändern möchten.

   Hinweis

   Beachten Sie die folgenden Hinweise:
   - Die Länge des Installationspfades darf 89 Zeichen nicht überschreiten.
   - Sie können den Installationspfad nur ändern, wenn noch kein Produkt des Software-Pakets installiert ist, das Sie installieren möchten.
   - Um die Sicherheit zu erhöhen, wählen Sie als Zielverzeichnis ein Verzeichnis, das durch Administratorrechten geschützt ist.

10. Klicken Sie auf die Schaltfläche "Weiter".
    Der Dialog zu den Lizenzbedingungen wird geöffnet.
11. Um die Installation fortzusetzen, lesen und akzeptieren Sie alle Lizenzvereinbarungen und klicken Sie auf "Weiter".
    Für den Fall, dass für die Installation des TIA Portals Sicherheits- und Rechteeinstellungen geändert werden müssen, wird der Dialog zu den Sicherheitseinstellungen geöffnet.
12. Um die Installation fortzusetzen, akzeptieren Sie die Änderungen der Sicherheits- und Rechteeinstellungen und klicken Sie auf die Schaltfläche "Weiter".
    Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
13. Überprüfen Sie die gewählten Installationseinstellungen. Wenn Sie Änderungen vornehmen möchten, klicken Sie auf die Schaltfläche "Zurück", bis Sie die zu ändernde Stelle im Dialog erreicht haben. Wenn Sie die gewünschten Änderungen vorgenommen haben, kehren Sie mit "Weiter" wieder zur Übersicht zurück.
14. Klicken Sie auf die Schaltfläche "Installieren".
    Die Installation wird gestartet.

    Hinweis

    Wenn während der Installation kein License Key gefunden wird, erhalten Sie die Möglichkeit diesen auf Ihren PC zu übertragen. Wenn Sie den Lizenztransfer überspringen, können Sie dies später mit dem Automation License Manager nachholen.

    Nach der Installation erhalten Sie eine Meldung darüber, ob die Installation erfolgreich durchgeführt wurde.

15. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
16. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Auf Ihrem Computer wurden das TIA Portal zusammen mit den erworbenen Produkten und Lizenzen und der Automation License Manager installiert.

Siehe auch

Installationsprotokoll (Seite 39)

Hinweise zu den Systemvoraussetzungen (Seite 27)

Hinweise zu Lizenzen (Seite 9)

Installierte Software anzeigen (Seite 45)

Installierte Produkte modifizieren oder hochrüsten (Seite 47)

Installierte Produkte reparieren (Seite 49)

Deinstallation starten (Seite 51)

Installierte Software anzeigen 7

Sie können sich jederzeit darüber informieren, welche Software installiert ist. Zusätzlich können Sie sich weitere Informationen zur installierten Software anzeigen lassen.

Vorgehen

Um eine Übersicht über die installierte Software aufzurufen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Hilfe" auf "Installierte Software".
   Der Dialog "Installierte Software" wird geöffnet. Innerhalb des Dialogs sehen Sie die installierten Software-Produkte. Klappen Sie die Einträge auf, um die jeweils installierte Version zu sehen.
2. Wenn Sie sich weitere Informationen zu der installierten Automatisierungssoftware anzeigen lassen möchten, klicken Sie auf den Link innerhalb des Dialogs "Weitere Informationen zu installierter Software".
   Der Dialog "Detailinformation" wird geöffnet.
3. Wählen Sie in der Bereichsnavigation das Thema, zu dem Sie sich weitere Informationen ansehen möchten.

Siehe auch

Hinweise zu den Systemvoraussetzungen (Seite 27)

Hinweise zu Lizenzen (Seite 9)

Installation starten (Seite 41)

Installierte Produkte modifizieren oder hochrüsten (Seite 47)

Installierte Produkte reparieren (Seite 49)

Deinstallation starten (Seite 51)

Installationsprotokoll (Seite 39)

Installierte Produkte modifizieren oder hochrüsten 8

Sie haben die Möglichkeit, installierte Produkte über das Setup-Programm zu ändern oder auf eine neue Version hochzurüsten (Update).

Bausteine mit Know-how-Schutz aus früheren Versionen des TIA Portals werden nicht automatisch mit dem Projekt hochgerüstet. Entfernen Sie den Know-how-Schutz der Bausteine, bevor Sie das TIA Portal aktualisieren. Richten Sie den Know-how-Schutz anschließend mit der aktuellen Version des TIA Portals wieder ein. Weitere Informationen finden Sie im Informationssystem.

Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

Vorgehen

Um installierte Produkte zu ändern oder hochzurüsten, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Modifizieren/Hochrüsten" und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Produktsprachen wird geöffnet.
7. Aktivieren Sie die Optionskästchen der Produktsprachen, die Sie installieren möchten.
   Bereits installierte Produktsprachen können Sie durch Deaktivieren der entsprechenden Optionskästchen deinstallieren.

   Hinweis

   Beachten Sie, dass Sie die Produktsprache "Englisch" nicht deinstallieren können.

8. Klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Produktkonfiguration wird geöffnet.
9. Aktivieren Sie die Optionskästchen der Bestandteile, die Sie installieren möchten. Bereits installierte Bestandteile können Sie durch Deaktivieren der entsprechenden Optionskästchen deinstallieren.
10. Klicken Sie auf die Schaltfläche "Weiter".

    Hinweis

    Beachten Sie, dass Sie das Zielverzeichnis nicht ändern können, da die vorhandene Installation verändert wird.

    Für den Fall, dass für die Installation des TIA Portals Sicherheits- und Rechteeinstellungen geändert werden müssen, wird der Dialog zu den Sicherheitseinstellungen geöffnet.
11. Um die Installation fortzusetzen, akzeptieren Sie die Änderungen der Sicherheits- und Rechteeinstellungen und klicken Sie auf die Schaltfläche "Weiter".
    Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
12. Klicken Sie auf die Schaltfläche "Modifizieren".
    Die Installation der zusätzlichen Bestandteile wird gestartet.

    Hinweis

    Nach der Installation erhalten Sie eine Meldung darüber, ob die bestehende Installation erfolgreich geändert wurde.
13. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
14. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Auf Ihrem Computer wurde die bestehende Installation geändert.

Siehe auch

Hinweise zu den Systemvoraussetzungen (Seite 27)

Hinweise zu Lizenzen (Seite 9)

Installation starten (Seite 41)

Installierte Software anzeigen (Seite 45)

Installierte Produkte reparieren (Seite 49)

Deinstallation starten (Seite 51)

Installationsprotokoll (Seite 39)

Installierte Produkte reparieren

9

Sie haben die Möglichkeit, installierte Produkte mithilfe des Setup-Programms durch eine vollständige Reinstallation zu reparieren.

Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

Vorgehen

Um installierte Produkte zu reparieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Reparieren" und klicken Sie auf die Schaltfläche "Weiter".
   Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
7. Klicken Sie auf die Schaltfläche "Reparieren".
   Das Reparieren der bestehenden Installation wird gestartet.

   Hinweis

   Nach der Installation erhalten Sie eine Meldung darüber, ob die installierten Produkte erfolgreich repariert wurden.

8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Die installierten Produkte wurden erneut installiert.

Siehe auch

Hinweise zu den Systemvoraussetzungen (Seite 27)
Hinweise zu Lizenzen (Seite 9)
Installation starten (Seite 41)
Installierte Software anzeigen (Seite 45)
Installierte Produkte modifizieren oder hochrüsten (Seite 47)
Deinstallation starten (Seite 51)
Installationsprotokoll (Seite 39)

Deinstallation starten 10

Einführung

Die Deinstallation der Softwarepakete wird durch ein Setup-Programm automatisch durchgeführt. Nach seiner Ausführung führt Sie das Setup-Programm Schritt für Schritt durch den gesamten Vorgang der Deinstallation.

Sie haben zwei Möglichkeiten für die Deinstallation:

- Deinstallation von ausgewählten Komponenten über die Systemsteuerung
- Deinstallation eines Produkts mithilfe des Installationsdatenträgers

---

Hinweis

Der Automation License Manager wird mit der Deinstallation der Softwarepakete nicht automatisch entfernt, da dieser für die Verwaltung der License Keys mehrerer Produkte der Siemens AG eingesetzt wird.

Deinstallation von ausgewählten Komponenten über die Systemsteuerung

Um ausgewählte Softwarepakete zu deinstallieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Systemsteuerung über "Start > Systemsteuerung".
2. Klicken Sie unter "Programme" auf "Programm deinstallieren".
   Der Dialog "Programm deinstallieren oder ändern" wird geöffnet.
3. Wählen Sie die zu deinstallierenden Softwarepakete und klicken Sie auf die Schaltfläche "Deinstallieren".
   Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
4. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten, und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Produkte, die Sie deinstallieren möchten, wird geöffnet.
5. Aktivieren Sie die Optionskästchen für die Produkte, die Sie deinstallieren möchten, und klicken Sie auf die Schaltfläche "Weiter".
   Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
6. Überprüfen Sie die Liste mit den Produkten, die deinstalliert werden. Wenn Sie Änderungen vornehmen möchten, klicken Sie auf die Schaltfläche "Zurück".
7. Klicken Sie auf die Schaltfläche "Deinstallieren".
   Die Deinstallation wird gestartet.
8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

Deinstallation eines Produkts mithilfe des Installationsdatenträgers

Um alle Softwarepakete zu deinstallieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Produktinformation lesen" bzw. "Installationshinweise lesen".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Deinstallieren" und klicken Sie auf die Schaltfläche "Weiter".
   Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
7. Klicken Sie auf die Schaltfläche "Deinstallieren".
   Die Deinstallation wird gestartet.
8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

Siehe auch

Installationsprotokoll (Seite 39)

Hinweise zu den Systemvoraussetzungen (Seite 27)

Hinweise zu Lizenzen (Seite 9)

Installation starten (Seite 41)

Installierte Software anzeigen (Seite 45)

Installierte Produkte modifizieren oder hochrüsten (Seite 47)

Installierte Produkte reparieren (Seite 49)

Updates und Support Packages installieren 11

11.1 Verfügbarkeit von Updates und Support Packages prüfen und installieren

Per Voreinstellung prüft das TIA Portal automatisch, ob neue Software Updates oder Support Packages verfügbar sind, z. B. Hardware Support Packages (HSPs). Die automatische Suche nach Updates findet nach jedem Neustart des Rechners und danach zyklisch alle 24 Stunden statt. Sie können die automatische Suche jederzeit deaktivieren bzw. erneut aktivieren. Zusätzlich können Sie manuell nach Updates suchen.

Falls Updates gefunden werden, können Sie diese herunterladen und installieren.

Hinweis

Es werden Updates und Support Packages ab TIA Portal V13 unterstützt.

Automatische Suche nach Software Updates deaktivieren oder aktivieren

Um die automatische Suche nach Software Updates zu deaktivieren oder erneut zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein > Software Updates".
3. Deaktivieren Sie das Optionskästchen "Täglich nach Updates suchen", wenn Sie die automatische Suche nach Software Updates deaktivieren möchten.
4. Aktivieren Sie das Optionskästchen "Täglich nach Updates suchen", wenn Sie die automatische Suche nach Software Updates erneut aktivieren möchten.

Manuell nach Software Updates suchen

Wenn Sie manuell nach Software Updates suchen möchten, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Hilfe" auf "Installierte Software".
   Der Dialog "Installierte Software" wird geöffnet.
2. Klicken Sie auf "Nach Updates suchen".
   Der TIA Updater wird geöffnet und die verfügbaren Updates werden angezeigt.

Oder:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein > Software Updates".
3. Klicken Sie auf "Jetzt nach Updates suchen".
   Der TIA Updater wird geöffnet und die verfügbaren Updates werden angezeigt.

Oder:

1. Öffnen Sie den TIA Updater über "Start > Alle Programme > Siemens Automation > Totally Integrated Automation UPDATER".
2. Klicken Sie auf "Nach Updates suchen".
   Die verfügbaren Updates werden angezeigt.

Server einstellen

Abhängig davon, ob Sie die Updates bzw. Support Packages vom TIA Automation Update Server oder von einem unternehmenseigenen Server herunterladen möchten, müssen Sie den entsprechenden Server einstellen. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie den TIA Updater.
2. Klicken Sie auf "Einstellungen" und wählen Sie im aufgeblendeten Dialog im Bereich "Server auf dem nach Updates gesucht werden soll" die Option "TIA Automation Software Update Server".
   Die Software sucht auf dem Server des Herstellers nach verfügbaren Updates.

Oder:

1. Öffnen Sie den TIA Updater.
2. Klicken Sie auf "Einstellungen" und wählen Sie im aufgeblendeten Dialog im Bereich "Server auf dem nach Updates gesucht werden soll" die Option "Anwender-definierter Server".
3. Geben Sie die Server-URL ein, die Sie von Ihrem Administrator bekommen haben. Die URL muss entweder im Format https://[URL des Servers]/[Pfad auf Ablageverzeichnis] oder im Format https://[IP-Adresse des Servers]/[Pfad auf Ablageverzeichnis] eingegeben werden. Sollten Sie sich nicht sicher sein, welches Format verwendet wird, wenden Sie sich an Ihren Administrator!
   Die Software sucht auf dem Server Ihres Unternehmens nach verfügbaren Updates.
4. Prüfen Sie den Namen der Fertigungsline und ändern Sie diesen bei Bedarf.
   Ausführliche Informationen zum Erstellen von Projekten und Arbeiten mit Fertigungslinien entnehmen Sie der Hilfe zum TIA Updater Corporate Configuration Tool.

Sie können die Optionen, auf welchem Server nach Updates gesucht werden soll, jederzeit umstellen. Während eines Download-Vorgangs ist die Umstellung jedoch blockiert. Nach der Umstellung des Servers werden alle bereits heruntergeladenen Updates und Support Packages angezeigt, auch wenn sie auf dem aktuell eingestellten Server nicht verfügbar sind.

Software Updates separat herunterladen und installieren

Um verfügbare Software Updates und Support Packages herunterzuladen und zu installieren, gehen Sie folgendermaßen vor:

1. Falls der TIA Updater nicht angezeigt wird, öffnen Sie ihn über einen der folgenden Wege:
   - Wenn die automatische Suche Updates gefunden hat, erhalten Sie eine Benachrichtigung im Infobereich der Taskleiste von Windows und das Symbol für den TIA Updater wird im Infobereich angezeigt. Klicken Sie entweder auf die Benachrichtigung oder auf das Symbol, um den TIA Updater zu öffnen.
   - Klicken Sie im Menü "Hilfe" auf "Installierte Software" und anschließend auf "Nach Updates suchen".
   - Klicken Sie im Startmenü von Windows auf "Start > Alle Programme > Siemens Automation > Automation Software Updater".
2. Klicken Sie in der Zeile des Updates oder Support Packages, das Sie installieren möchten, auf "Download".
   Das Update oder Support Package wird heruntergeladen. Sobald der Download-Vorgang beendet ist, wird die zugehörige Schaltfläche "Installieren" aktiv.

   Hinweis

   Beachten Sie folgende Hinweise:
   1. Sie können gleichzeitig mehrere Download-Vorgänge anstoßen.
   2. Während eines laufenden Download-Vorgangs können Sie sich abmelden oder den Rechner auch herunterfahren. Der Download-Vorgang wird in diesen Fällen im Hintergrund fortgesetzt, sobald Sie sich erneut anmelden.
   3. In einigen Fällen wird statt der Schaltfläche "Download" ein Link auf eine externe Web-Seite angezeigt. Laden Sie das Software Update dann von dieser Seite herunter und installieren Sie es manuell.
3. Schließen Sie das TIA Portal, falls es noch geöffnet ist.
4. Klicken Sie im TIA Updater auf die Schaltfläche "Installieren" des Software-Pakets, das Sie installieren möchten.
   Der Installationsdialog wird angezeigt.

   Hinweis

   Beachten Sie folgende Hinweise:
   1. Die Installation mehrerer Updates zur gleichen Zeit ist nicht möglich.
   2. Melden Sie sich während einer Installation nicht ab und fahren Sie den Rechner nicht herunter. Dadurch vermeiden Sie inkonsistente Stände der Software auf Ihrem Rechner.
5. Klicken Sie auf "Weiter".
   Das ausgewählte Produkt wird installiert.

Mehrere Support Packages in einem Schritt herunterladen und installieren

Um verfügbare Support Packages in einem Schritt herunterzuladen und zu installieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie die gewünschten Zeilen. Hierzu haben Sie mehrere Möglichkeiten:
   - Um alle angezeigten Support Packages herunterzuladen und zu installieren, aktivieren Sie das Häkchen im Titel der Liste.
   - Um mehrere Support Packages herunterzuladen und zu installieren, markieren Sie das erste gewünschte Support Package, halten die <Ctrl>-Taste gedrückt und markieren Sie dann alle weiteren gewünschten Support Packages. Setzen Sie dann das Häkchen in eine der markierten Zeilen.
   - Um mehrere Support Packages in einer bestimmten Reihenfolge herunterzuladen und zu installieren, markieren Sie das erste gewünschte Support Package, halten die <Shift>-Taste gedrückt und markieren Sie dann das letzte gewünschte Support Package.
2. Klicken Sie auf die Schaltfläche "Herunterladen und installieren".
3. Klicken Sie nach dem Herunterladen im aufgeblendeten Dialog auf "Weiter", um die Installation zu starten. Danach wird ein Dialog angezeigt, der Sie über den Fortschritt der Installation informiert.

Hinweis

Es können nur Support Packages heruntergeladen und installiert werden, die keinen Exporteinschränkungen unterliegen.

Alternative Vorgehensweise für die Installation von Support Packages

Für die Installation eines Support Packages steht Ihnen eine zusätzliche Vorgehensweise zur Verfügung. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie im TIA Portal im Menü "Extras" auf "Support Packages".
   Der Dialog "Detailinformation" wird geöffnet. In einer Tabelle werden alle Support Packages aus dem Verzeichnis aufgelistet, das Sie als Speicherort für Support Packages in den Einstellungen festgelegt haben.
2. Falls Sie ein Support Package installieren möchten, das in der Liste nicht aufgeführt wird, haben Sie folgende Möglichkeiten:
   - Wenn sich das Support Package bereits auf Ihrem Computer befindet, können Sie es über "Aus dem Dateisystem hinzufügen" in die Liste aufnehmen.
   - Wenn Sie ein Support Package von der "Service & Support"-Seite im Internet hinzufügen möchten, laden Sie es zunächst mit "Aus dem Internet laden" herunter. Anschließend können Sie es aus dem Dateisystem hinzufügen.
3. Selektieren Sie das Support Package, das Sie installieren möchten.
4. Klicken Sie auf "Installieren".
5. Beenden Sie das TIA Portal und starten Sie es erneut.

Siehe auch

Installationsprotokoll (Seite 39)

11.2 Arbeiten mit einem unternehmenseigenen Server

11.2.1 Eigenschaften und Vorteile eines unternehmenseigenen Servers

Einleitung

Mithilfe eines unternehmenseigenen Servers können Sie ausgewählte Updates/Support Packages auf einem lokalen Server ablegen und den Anwendern zur Verfügung stellen. Dies hat den Vorteil, dass die Anwender nicht auf das Internet zugreifen müssen, sondern die Updates über das Intranet, eine externe Festplatte etc. installieren können. Da die Anwender keinen direkten Zugriff ins Internet benötigen, wird der Schutz vor Trojanern oder schädlicher Software, die Kontakt aus dem internen Firmennetzwerk ins Internet aufnimmt, deutlich erhöht.

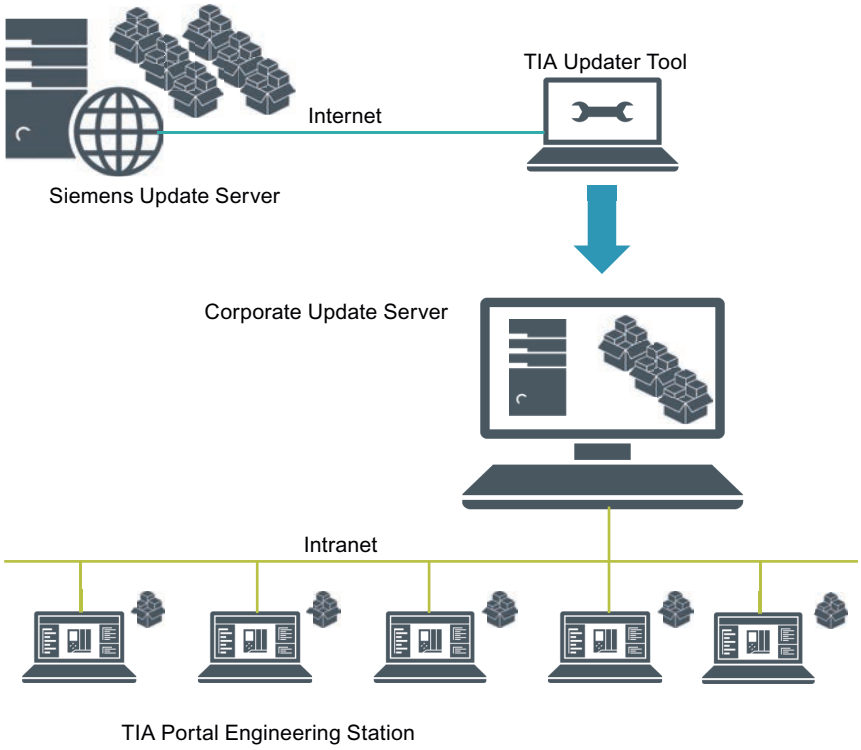

Konfigurieren des Servers und Erstellen von Projekten

In der ersten Phase wird der unternehmenseigene Server durch einen Server-Administrator konfiguriert, und mittels TIA Updater Corporate Configuration Tool werden die Updates/ Support Packages bereitgestellt. Zusätzlich können Projekte für verschiedene Fertigungslinien erstellt werden, über die der Anwender die von ihm benötigten Updates erhalten kann. Die Anwender müssen Zugriff auf den Serverbereich und ggf. den Namen der Fertigungslinie haben und über das Ablageverzeichnis in Kenntnis gesetzt werden.

Ausführliche Informationen zum Erstellen von Projekten und Arbeiten mit Fertigungslinien entnehmen Sie der Hilfe zum TIA Updater Corporate Configuration Tool.

Arbeiten mit dem Server

Mit dem TIA Updater können die Anwender die bereitgestellten und für sie relevanten Updates/ Support Packages herunterladen und installieren. Es können gleichzeitig mehrere Download-Vorgänge angestoßen werden; die Installation der Updated/Support Packages muss jedoch hintereinander erfolgen.

11.2.2 Unternehmenseigenen Server für Updates konfigurieren

Einfühung

Um den Anwendern zentral die verfügbaren Updates und Support Packages zur Verfügung zu stellen, müssen Sie einen unternehmenseigenen Server konfigurieren. Verwenden Sie hierzu den Microsoft Server Manager.

Nachfolgend sind beispielhaft die Schritte beschrieben, die zum Erstellen und Konfigurieren des Servers zur Nutzung des TIA Updater Corporate Configuration Tools (mit dem Betriebssystem Microsoft Server 2016) notwendig sind. Weitere Einstellungen, die evtl. zum Betrieb in Ihrem Unternehmen notwendig sind, sind **nicht** Bestandteil dieser Beschreibung!

Hinweis

Bitte beachten Sie, dass der Aufbau des Startmenüs und der Ablageort der Programme unter den verschiedenen Betriebssystemen unterschiedlich sein kann.

Genaue Informationen zur Konfiguration und Bedienung des Server Managers finden Sie in der Microsoft-Hilfe.

Voraussetzung

Sie müssen Administratorrechte besitzen.

Webserverrolle (IIS) installieren

Um die notwendige Webserverrolle zu installieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Startmenü und wählen Sie "Server Manager" aus.
2. Klicken Sie im Dashboard auf "Rollen und Features hinzufügen".
3. Verwenden Sie den Assistenten "Rollen und Features hinzufügen", um die Webserverrolle hinzuzufügen.
   Vor dem Start des Assistenten wird verifiziert, dass der Administrator ein komplexes Passwort vergeben hat, dass die Netzwerkeinstellungen entsprechend konfiguriert sind und dass die neusten Windows Sicherheits-Updates installiert sind.
4. Wählen Sie als Installationstyp "Rollenbasierte oder featurebasierte Installation" und klicken Sie auf "Weiter".
5. Wählen Sie dann den Ziel-Server aus und klicken Sie auf "Weiter".

6. Beachten Sie die vorausgewählten Rollen, die standardmäßig installiert werden, und wählen Sie dann die zusätzliche Rolle "Webserver (IIS)" aus.
7. Klicken Sie auf "Features hinzufügen" und dann auf "Weiter". Die Features für den Web Server werden angezeigt.
8. Klicken Sie auf "Role 'Services" und prüfen Sie, dass die folgenden Features ausgewählt sind, bzw. wählen Sie diese aus:
   Bereich "Webserver"
   - Standarddokument
   - Verzeichnis durchsuchen
   - HTTP-Fehler
   - Statischer Inhalt

   Bereich "Systemzustand und Diagnose"
   - HTTP-Protokollierung
   - Anforderungsüberwachung

   Bereich "Leistung"
   - Komprimierung statischer Inhalte

   Bereich "Sicherheit"
   - Anforderungsfilterung

   Bereich "Anwendungsentwicklung"
   - .NET-Erweiterbarkeit 3.5
   - .NET-Erweiterbarkeit 4.6
   - ASP.NET 3.5
   - ASP.NET 4.6
   - ISAPI-Erweiterungen
   - ISAPI-Filter
9. Klicken Sie auf "Weiter".
10. Überprüfen Sie Ihre Auswahl im Dialog "Installationsauswahl bestätigen" und klicken Sie auf "Installieren".

Im Dashboard wird "IIS" hinzugefügt und kann weiter konfiguriert werden.

Website erstellen

1. Klicken Sie im Navigationsbereich auf "IIS" und im Bereich "Server" klicken Sie mit der rechten Maustaste und wählen Sie "Internetinformationsdienste (IIS)-Manager".
2. Klicken Sie im Bereich "Verbindungen" auf "Sites" und klicken Sie dann auf "Default Web Site".
3. Wählen Sie mit einem rechten Mausklick die Option "Virtuelles Verzeichnis hinzufügen...".
4. Geben Sie im aufgeblendeten Dialog einen Anzeigenamen in das Feld "Alias" ein, z. B. "TIAPortalUpdates".

5. Geben Sie im Feld "Physikalischer Pfad" den physikalischen Pfad des Ordners ein, in dem sich die Website befindet, oder klicken Sie auf die Schaltfläche zum Durchsuchen (...), um den Ordner im Dateisystem zu suchen.
6. Klicken Sie auf "Einstellungen testen...", um zu prüfen, ob die Einstellungen korrekt sind.
7. Klicken Sie auf "OK".

Virtuelles Verzeichnis aktivieren

1. Markieren Sie im Bereich "Verbindungen > Sites > Default Web Ssite" das von Ihnen erstellte virtuelle Verzeichnis. Doppelklicken Sie in der Ansicht "Features" den Eintrag "Verzeichnis durchsuchen".
2. Klicken Sie im Bereich "Aktionen" auf "Aktivieren".

MIME-Typ hinzufügen und konfigurieren

1. Markieren Sie im Bereich "Verbindungen" die von Ihnen erstellte Website.
2. Doppelklicken Sie in der Ansicht "Features" auf "MIME-Typ".
3. Klicken Sie im Bereich "Aktionen" auf "Hinzufügen".
4. Geben Sie im Dialog "MIME-Typ hinzufügen" im Textfeld "Dateinamenerweiterung" .* ein.
5. Geben Sie im Textfeld "MIME-Typ" "all/files" ein.
6. Klicken Sie auf "OK".

BITS-IIS-Servererweiterung aktivieren

1. Wechseln Sie zum virtuellen Verzeichnis im IIS-Manager.
2. Doppelklicken Sie in der Featureansicht des virtuellen Verzeichnisses auf "BITS-Uploads".
3. Aktivieren Sie das Optionskästchen "Clients können Dateien hochladen" und klicken Sie dann auf "Übernehmen".

Selbstsigniertes Serverzertifikat erstellen

1. Navigieren Sie im Bereich "Verbindungen" zu der Ebene, die Sie verwalten möchten.
2. Doppelklicken Sie in der Ansicht "Features" auf "Serverzertifikate".
3. Klicken Sie im Bereich "Aktionen" auf "Selbstsigniertes Zertifikat erstellen".
4. Geben Sie im Dialog "Selbstsigniertes Zertifikat erstellen" im Feld "Anzeigename für das Zertifikat" einen Anzeigenamen für das Zertifikat ein und klicken Sie auf "OK".

SSL-Bindung erstellen

1. Erweitern Sie im Bereich "Verbindungen" den Eintrag "Sites" und klicken Sie anschließend auf die Site, der Sie eine Bindung hinzufügen möchten.
2. Klicken Sie im Bereich "Aktionen" auf "Bindungen".

3. Klicken Sie im Dialog "Sitebindungen" auf "Hinzufügen".
4. Wählen Sie im Dialog "Sitebindung hinzufügen" unter "Typ" "https" und unter "SSL-Zertifikat" das von Ihnen erstellte Zertifikat aus und klicken Sie dann auf "OK".

SSL-Bindung verifizieren

1. Klicken Sie im Bereich "Aktionen" unter "Website durchsuchen" auf die zuvor erstellte Bindung.
   Im Internet Explorer wird eine Fehlerseite geöffnet, weil das selbstsignierte Zertifikat von Ihrem Computer erstellt wurde.
2. Klicken Sie auf "Laden dieser Website fortsetzen (nicht empfohlen)".
   Wenn Sie das Zertifikat in den Zertifikatspeicher "Vertrauenswürdige Stammzertifizierungsstellen" hinzufügen, wird diese Meldung nicht mehr angezeigt.

SSL-Einstellungen konfigurieren

1. Doppelklicken Sie in der Ansicht "Features" auf "SSL-Einstellungen".
2. Verwenden Sie eines der folgenden Verfahren im Dialog "SSL-Einstellungen" im Bereich "Clientzertifikate":
   - Wählen Sie "Ignorieren", wenn auch dann keine Clientzertifikate akzeptiert werden sollen, wenn ein Client ein Zertifikat vorweisen kann.
   - Wählen Sie "Akzeptieren", wenn Clientzertifikate akzeptiert werden sollen.
   - Wählen Sie "Erforderlich", wenn Clientzertifikate erforderlich sein sollen. Damit die Option "Clientzertifikate erforderlich" verwendet werden kann, muss die Option "SSL erforderlich" aktiviert sein.
3. Klicken Sie im Bereich "Aktionen" auf "Übernehmen".

Updates auf unterschiedliche Bereiche verteilen

Es kann vorkommen, dass verschiedene Abteilungen unterschiedliche Updates oder Support Packages benötigen. In diesem Fall ist es empfehlenswert, mehrere Server für Updates zu betreiben, die unterschiedliche Updates und Support Packages anbieten. Gehen Sie dazu folgendermaßen vor:

1. Erstellen Sie unterschiedliche unternehmenseigene Server wie oben beschrieben.

oder

1. Installieren Sie die Webserverrolle.
2. Erstellen Sie mehrere Websites.
   Beachten Sie, dass Sie eindeutige Namen für diese Websites und die physikalischen Pfade vergeben, damit es nicht zu Verwechselungen kommt!
3. Stellen Sie die oben beschriebenen Features für den Webserver bzw. die Websites ein.

Sie können nun die erforderlichen Updates und Support Packages für die verschiedenen Abteilungen in den definierten Verzeichnissen ablegen.

11.2.3 Updates auf einem unternehmenseigenen Server bereitstellen

Einführung

Im TIA Updater Corporate Configuration Tool können Sie einen unternehmenseigenen Server einstellen, auf dem Sie die verfügbaren Updates/Support Packages ablegen und den Anwendern zur Verfügung stellen können.

Voraussetzung

Sie müssen Administratorrechte besitzen.

Ein unternehmenseigener Server kann nur auf einem Microsoft Server Betriebssystem aufgesetzt werden, da zum Betrieb der BITS IIS-Dienst und ein laufender IIS-Sever benötigt wird.

Updates vom TIA Automation Software Update Server hinzufügen

Um Updates vom TIA Automation Software Update Server hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das TIA Updater Corporate Configuration Tool.
   Die verfügbaren Updates werden angezeigt.
2. Klicken Sie auf die Schaltfläche "Update hinzufügen" und aktivieren Sie im aufgeblendeten Dialog das Optionskästchen "Vom TIA Automation Software Update Server hinzufügen".
3. Wählen Sie im Dialog "Update vom TIA Automation Software Update Server hinzufügen" die gewünschten Updates (Software und Support Packages) aus und klicken Sie auf "Hinzufügen".
   Wenn sich bereits Updates auf dem unternehmenseigenen Server befinden, so werden diese gegraut dargestellt. Sie können nicht erneut geladen werden. Während des Download-Vorgangs werden der Status und die verbleibende Zeit im Dialog angezeigt.

Abbrechen des Download-Vorgangs

Um den Vorgang abzubrechen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Download abbrechen".
2. Bestätigen Sie den aufgeblendeten Dialog mit "Ja".
   Der Download des Updates wurde abgebrochen und es wurde aus der Liste gelöscht.

Updates vom unternehmenseigenen Server löschen

Um Updates zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie die gewünschten Updates aus und klicken Sie auf die Schaltfläche "Entfernen".
2. Bestätigen Sie den aufgeblendeten Dialog mit "Ja".
   Die ausgewählten Updates wurden aus der Liste gelöscht.

Server-Optionen

Klicken Sie im Dialog " TIA Automation Software Update Server" auf "Einstellungen", um Folgendes festzulegen:

1. Geben Sie unter "Server-Pfad" den Ordner an, in dem die Downloads gespeichert werden sollen. Sie können sowohl ein lokales Laufwerk als auch ein Netzlaufwerk auswählen.

   Hinweis

   Wenn Sie direkt auf dem Server arbeiten, ist das Zielverzeichnis gleich dem Bereitstellungsverzeichnis auf dem Server. Alle Änderungen werden direkt übernommen. Diese Vorgehensweise wird nicht empfohlen, da Konflikte mit aktiven Downloads auftreten können!

   Es wird empfohlen, auf einem sonstigen Rechner arbeiten; das Zielverzeichnis ist ein beliebiges Verzeichnis. Der Inhalt dieses Verzeichnisses muss dann noch in das Bereitstellungsverzeichnis kopiert werden. Stellen Sie hierbei ebenfalls sicher, dass keine Konflikte mit aktiven Downloads auftreten!

2. Aktivieren Sie das Optionskästchen "Server immer im Hintergrund laufen lassen (Symbol in der Taskleiste wird angezeigt)", wenn der Server immer im Hintergrund laufen soll.
3. Bestätigen Sie Ihre Eingaben mit "OK".

Siehe auch

Eigenschaften und Vorteile eines unternehmenseigenen Servers (Seite 58)

Unternehmenseigenen Server für Updates konfigurieren (Seite 59)

Support Packages automatisiert installieren 12

12.1 Support Packages automatisiert installieren

Einleitung

Mit dem Support Package Installer können Sie ab V15.1 alle Support Packages wie z. B. HSPs (Hardware Support Packages), die kompatibel zur installierten Version des TIA Portals sind (z. B. isp.15_1), automatisiert über die Kommandozeile installieren bzw. hochrüsten.

Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

Vorgehen

Um die Installation direkt über die Kommandozeile mit den gewünschten Optionen zu starten, gehen Sie folgendermaßen vor:

1. Starten Sie die Eingabeaufforderung von Windows über "Start > Ausführen > cmd".
2. Wechseln Sie in das Verzeichnis, das die Datei "Siemens.Automation.SupportPackageInstaller.exe" enthält. Dies ist das Installationsverzeichnis Ihres installierten TIA Portals.
3. Geben Sie in der Eingabeaufforderung den folgenden Befehl ein:
   %Installationsverzeichnis% > Siemens.Automation.SupportPackageInstaller.exe <Support Package(HSP)-Ablageverzeichnis> [-l <Name der Protokolldatei>] [-warnaserror]

<Support Package-Ablageverzeichnis>
Zwingend erforderliches Verzeichnis, in dem die Support Package-Dateien (*.isp) liegen. Dies kann sowohl ein lokaler Ordner oder ein Remote Share sein.

-l <Ablageort der Protokolldatei>
Dieser Parameter ist optional und gibt den Pfad zur Protokolldatei an. Wenn er nicht gesetzt ist, wird die Protokolldatei im Binärverzeichnis erstellt. Sie müssen über Schreibrechte im Ablageordner verfügen!

-warnaserror
Dieser Parameter ist optional und gibt an, dass jede Nachricht, die in der Regel als Warnungen gemeldet worden wäre, stattdessen als Fehler gemeldet wird. Wenn beispielsweise ein HSP nicht den notwendigen Voraussetzungen entspricht, wird ein entsprechender Rückgabe-/Abbruchwert an die Kommandozeile ausgegeben und das betreffende HSP wird übersprungen.

-?
Dieser Parameter ist optional und zeigt die Hilfe an.

Hinweis

- Wenn das Ablageverzeichnis oder der Name der Protokolldatei Leerzeichen enthält, müssen Sie die Bezeichnungen in Anführungszeichen setzen.
- Es werden nur Support Packages berücksichtigt, die direkt im Ablageverzeichnis liegen; Unterverzeichnisse werden ignoriert. D. h. die *.isp-files müssen unmittelbar im Ablageverzeichnis liegen.

Siehe auch

Rückgabewerte des Installationsprozesses (Seite 67)

Protokolldatei (Seite 68)

12.2 Rückgabewerte des Installationsprozesses

Kernaussage

Die folgende Tabelle zeigt die Rückgabewerte und ihre Beschreibungen:

Code	Beschreibung
0	Die Installation war erfolgreich.
1	Die Installation war erfolgreich, es ist ein Neustart erforderlich.
-1	Die Installation ist fehlgeschlagen.
-2	Die Installation ist fehlgeschlagen: Das TIA Portal läuft noch.
-3	Die Installation ist fehlgeschlagen: Eine andere Installation läuft bereits.
-4	Die Installation ist fehlgeschlagen: Sie haben nicht genügend Rechte. Für die Installation sind Administrationsrechte erforderlich.
-6	Die Installation ist fehlgeschlagen: Unzureichender Speicherplatz.
-7	Die Installation ist fehlgeschlagen: Es ist ein Neustart erforderlich bevor die Installation starten kann.

Siehe auch

Support Packages automatisiert installieren (Seite 65)

Protokolldatei (Seite 68)

12.3 Protokolldatei

Inhalt der Protokolldatei

Die Protokolldatei enthält detaillierte Informationen über jede Installation. Wenn die Protokolldatei bereits existiert, wird sie ergänzt. Für jede Installation werden Kopf- und Fußzeilen erstellt, zwischen denen Sie die gewünschten Informationen finden.

Für die folgenden Ereignisse wird ein Eintrag erzeugt:

- Support Packages, die zwar im Ablageverzeichnis aufgeführt und bei der Installation verfügbar sind, aber aufgrund fehlender Produkte nicht installiert werden konnten. Dem Eintrag in der Protokolldatei ist zu entnehmen, welche Produkte und welche Versionen dieser Produkte fehlen.
- Support Packages, die nicht erfolgreich installiert werden konnten.
- Support Packages, die erfolgreich installiert werden konnten.

Für jeden Eintrag wird der Dateiname des HSPs einschließlich der Dateiendung protokolliert.

Siehe auch

Support Packages automatisiert installieren (Seite 65)

Rückgabewerte des Installationsprozesses (Seite 67)

Migrations-Tool installieren und deinstallieren 13

13.1 Systemvoraussetzungen

Systemvoraussetzungen für das Migrations-Tool

Zur Verwendung des Migrations-Tools gelten folgende Systemvoraussetzungen:

- Es sind alle Produkte installiert, mit denen das Quellprojekt erstellt wurde. Folgende Produkte werden unterstützt:
  - Ab STEP 7 V5.4 SP5
  - WinCC V7.4 SP1 mit den neuesten Updates
  - Ab WinCC flexible 2008 SP2
  - Integrierte Projekte aus STEP 7 V5.4 und den oben genannten WinCC-Produkten
  - STEP 7 Distributed Safety V5.4 SP5
  - F-Configuration-Pack V5.4 SP5 bis V5.5 SP12
  - SINUMERIK STARTER und Startdrives
  - Ab SIMOTION SCOUT V4.4
    Für die Migration von SIMOTION SCOUT-Projekten wird das SCOUT Migration Tool PlugIn in der zum SCOUT TIA entsprechenden Version benötigt.
- Es sind alle Optionspakete installiert, welche für das Bearbeiten des STEP 7-Projekts notwendig sind. Dies umfasst beispielsweise alle HSPs für alle im Quellprojekt verwendeten Geräte.
- Microsoft .Net Framework 4.6.2 ist installiert. Wenn Microsoft .Net Framework nicht installiert ist, wird die benötigte Version mit dem Migrations-Tool installiert.

13.2 Migrations-Tool installieren

Vertrieb des Migrations-Tools

Das Migrations-Tool befindet sich auf der Installations-DVD des TIA Portals im Verzeichnis "Support". Alternativ steht es im Siemens Industry Online Support zum Download bereit. Für manche Produkte sind zusätzliche Plugins für das Migrations-Tool erforderlich, z. B. für SIMATIC Failsafe oder SIMOTION. Die Plugins können Sie ebenfalls im Siemens Industry Online Support herunterladen oder von der Installations-DVD der jeweiligen Produkte installieren.

In der Regel wird das Migrations-Tool ohne das TIA Portal installiert. Da das TIA Portal eine eigene Migrations-Funktion bereits eingebunden hat, ist eine separate Installation des Migrations-Tools nicht notwendig.

Vorgehen

Um das Migrations-Tool zu installieren, gehen Sie folgendermaßen vor:

1. Laden Sie die Installations-Datei im Siemens Industry Online Support herunter oder verwenden Sie für die Installation die Installations-Datei aus dem Verzeichnis "Support" der Installations-DVD des TIA Portals.
2. Starten Sie die Installations-Datei.
   Das Setup-Programm für das Migrations-Tool wird geöffnet.
3. Wählen Sie im ersten Schritt, in welcher Sprache das Setup angezeigt werden soll und klicken Sie auf die Schaltfläche "Weiter".
   Die Seite zur Auswahl der Software-Sprache wird angezeigt.
4. Da das Migrations-Tool ausschließlich in englischer Sprache geliefert wird, können Sie keine weitere Sprache zur Installation wählen. Klicken Sie daher auf "Weiter", um zum nächsten Schritt zu gelangen.
   Die Seite zur Auswahl der Produkte wird angezeigt.
5. Das Migrations-Tool besteht lediglich aus einer Software-Komponente. Daher ist das Migrations-Tool bereits ausgewählt. Klicken Sie auf die Schaltfläche "Weiter".
   Die Seite zur Bestätigung der Lizenzbestimmungen wird geladen.
6. Klicken Sie auf die Einträge in der Liste der Lizenzbedingungen, um die Lizenzbedingungen und Securityhinweise zu lesen. Sind Sie mit allen Lizenzbedingungen einverstanden, aktivieren Sie die Optionskästchen "Ich akzeptiere sämtliche Bedingungen der aufgelisteten Lizenzvereinbarung(en)" und "Ich bestätige hiermit, dass ich die Securityhinweise zum sicheren Betrieb der Produkte gelesen und verstanden habe".
7. Klicken Sie anschließend auf die Schaltfläche "Weiter".
   Eine Übersicht zur Installation wird angezeigt.
8. Klicken Sie auf die Schaltfläche "Installieren".
   Die Installation wird mit den angezeigten Einstellungen durchgeführt.
   Nach der Installation des Migrations-Tools ist ein Neustart des Rechners erforderlich.

13.3 Migrations-Tool deinstallieren

Das Migrations-Tool kann über die Systemsteuerung deinstalliert werden.

Vorgehen

Um das Migrations-Tool zu deinstallieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie in der Systemsteuerung auf den Eintrag "Programme und Funktionen".
   Der Dialog "Programm deinstallieren oder ändern" wird geöffnet.
3. Wählen Sie im Dialog "Programm deinstallieren oder ändern" den Eintrag für das Migrations-Tool aus und klicken Sie auf die Schaltfläche "Deinstallieren".
   Es erscheint eine Sicherheitsabfrage.
4. Bestätigen Sie die Sicherheitsabfrage, indem Sie auf die Schaltfläche "Ja" klicken.
   Das Setup-Programm startet.
5. Wählen Sie die Installationssprache aus und klicken Sie auf "Weiter".
6. Wählen Sie das Migrations-Tool zur Deinstallation aus und klicken Sie auf "Weiter".
7. Klicken Sie auf "Deinstallieren".
   Das Migrations-Tool wird deinstalliert.
8. Um das Setup-Programm zu beenden, klicken Sie auf "Beenden".

Multiuser-Server installieren und deinstallieren 14

14.1 Hinweise zur Installation des Multiuser-Servers

Einführung

Die Installation des Multiuser-Servers ermöglicht Ihnen, die Funktionalität des Multiuser Engineering zu nutzen.

Sie können den Multiuser-Server wie folgt installieren:

- Zusammen mit der Installation des TIA Portals.
- Als Standalone-Installation über einen eigenständigen Installationsprozess ohne TIA Portal.

Hinweis

Für die Serverinstallation werden Administratorrechte benötigt.

Installation des Multiuser-Servers mit dem TIA Portal

Die Installation des Multiuser-Servers erfolgt zusammen mit der Installation folgender TIA Portal-Produkte:

- SIMATIC STEP 7 Basic
- SIMATIC STEP 7 Professional
- SIMATIC WinCC Basic
- SIMATIC WinCC Comfort/Advanced
- SIMATIC WinCC Professional

Der Multiuser-Server ist in der Produktauswahl teilweise per Voreinstellung aktiviert und wird dann mitinstalliert. Überprüfen Sie die Einstellung während der Installation des TIA Portals. Sie finden die jeweilige Voreinstellung für den Multiuser Server während des Setups bei der Produktauswahl im Verzeichnis "Options". Wählen Sie hier durch das Setzen oder entfernen des Häkchens aus, ob der Multiuser-Server während der Installation des TIA Portals mitinstalliert werden soll oder nicht.

Installation des Multiuser-Servers als Standalone-Installation

Die Installation des Multiuser-Servers kann auch als Standalone-Installation durchgeführt werden.

Eine Beschreibung zur automatisierten Installation finden Sie auch auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Gleiche TIA Portal-Versionen bei der Installation

Achten Sie bei der Installation von TIA Portal-Produkten auf den verschiedenen Clients darauf, dass alle Teilnehmer für die Installation die gleichen Stände der Software-Produkte verwenden.

Wenn Sie also z. B. STEP 7 Professional, WinCCAdvanced und Safety sowie den Multiuser Server mit der Version Vx.y installiert haben, müssen alle anderen Teilnehmer von Team Engineering ebenfalls diese Software in identischen Versionen installiert haben.

Die Installation von Service Packs und Updates muss für alle Produkte zu einem gemeinsamen Zeitpunkt erfolgen.

Installationspfad

Verwenden Sie keine Unicode-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

Installation eines neueren Multiuser-Servers

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Virenscanner

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscanner verhindern dies. Daher wird empfohlen, während der Installation des Multiuser-Servers Virenscanner zu deaktivieren und erst danach wieder zu aktivieren.

Unterstützte Virtualisierungsplattformen

Sie können den Multiuser-Server ebenso wie das TIA Portal auch in einer virtuellen Maschine installieren.

Verwenden Sie dazu eine der folgenden Virtualisierungsplattformen in der angegebenen oder einer neueren Version:

- VMware vSphere Hypervisor (ESXi) 6.5
- VMware Workstation 12.5.7
- VMware Player 12.5.7
- Microsoft Hyper-V Server 2016

Für die Installation einer Visualisierungsplattform gelten die gleichen Voraussetzungen wie für die Installation des TIA Portals.

Siehe auch

Multiuser-Server installieren (Seite 76)
Multiuser-Server deinstallieren (Seite 78)

14.2 Multiuser-Server installieren

Einleitung

Der Multiuser-Server kann im Rahmen der Installation des TIA Portals oder auch standalone installiert werden. Wird der Multiuser-Server mit dem TIA Portal mitinstalliert, so wird immer auch die dem TIA Portal entsprechende Version des Multiuser-Servers installiert.

Eine Beschreibung zur automatisierten Installation finden Sie auch auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Hinweis

Neuinstallation vor Deinstallation

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Regeln für die Installation

Auf einem Computer können mehrere Versionen des Multiuser-Servers installiert sein, die unterschiedliche Projektversionen des TIA Portals nutzen können.

Nutzbare Projektversionen:

Multiuser-Server-Version	TIA Portal-Projektversion
Server V15.1	TIA Portal V15.1, V15 und V14
Server V15	TIA Portal V15 und V14
Server V14	TIA Portal V14

Beachten Sie jedoch, dass jede Version des TIA Portals und des Multiuser-Servers jeweils nur über die in dieser Version enthaltene Funktionalität verfügt.

Multiuser Engineering führt kein Upgrade der Projektversionen durch.

Voraussetzungen für die Installation

Für die Installation des Multiuser-Servers gelten die gleichen Voraussetzungen wie für die Installation des TIA Portals.

Multiuser-Server mit dem TIA Portal installieren

Für die Installation gehen Sie folgendermaßen vor:

1. Beachten Sie die Voraussetzungen für die Installation des ausgewählten Softwarepakets.
2. Starten Sie das Setup-Programm des gewünschten Produkts durch Einlegen des Installationsdatenträgers in das entsprechende Laufwerk.

3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Vergewissern Sie sich, dass der Multiuser-Server in der Produktauswahl angewählt ist und folgen Sie den Hinweisen in den Setup-Dialogen.

Ergebnis

Der Multiuser-Server wurde auf Ihrem Computer zusammen mit dem TIA Portal installiert.

Multiuser-Server standalone installieren

Für die Installation gehen Sie folgendermaßen vor:

1. Beachten Sie die Voraussetzungen für die Installation des Multiuser-Servers.
2. Starten Sie den Setup für die Standalone-Installation des Multiuser-Servers durch Doppelklick auf die selbstextrahierende Exe-Datei "TIA_Portal_Multiuser_Server_V<x.y>.exe". Sie finden diese Datei im "Support"-Verzeichnis auf der jeweiligen Produkt-DVD.
3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Folgen Sie den Anweisungen in den Setup-Dialogen.

Ergebnis

Der Multiuser-Server wurde auf Ihrem Computer installiert.

Siehe auch

Hinweise zur Installation des Multiuser-Servers (Seite 73)

Multiuser-Server deinstallieren (Seite 78)

14.3 Multiuser-Server deinstallieren

Einleitung

Der Multiuser-Server kann über die Systemsteuerung deinstalliert werden.

Hinweis

Neuinstallation vor Deinstallation

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Voraussetzungen für die Deinstallation

Auf Ihrem Computer ist eine Multiuser-Serverinstallation vorhanden.

Multiuser-Server deinstallieren

Für die Deinstallation gehen Sie vor wie folgt:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie in der Systemsteuerung auf den Eintrag "Software".
   Der Dialog "Software" wird geöffnet.
3. Wählen Sie im Dialog "Software" den Eintrag für den "Multiuser-Server" aus und klicken Sie auf die Schaltfläche "Entfernen".
   Es erscheint eine Sicherheitsabfrage.
4. Bestätigen Sie die Sicherheitsabfrage, indem Sie auf die Schaltfläche "Deinstallieren" klicken.

Ergebnis

Der Multiuser-Server wurde auf ihrem Computer deinstalliert.

Siehe auch

Hinweise zur Installation des Multiuser-Servers (Seite 73)

Multiuser-Server installieren (Seite 76)

Index

B

Bediengerät, 22
Betriebssystem, 29

D

Datensicherung, 19
defekte Lizenz, 20
Deinstallation starten, 51
Deinstallation von License Keys, 18

H

Hardware Support Package
    installieren, 55
Hardwarevoraussetzungen, 29
Hardware-Voraussetzungen, 29
HSP, (Siehe Support Package)

I

Installation
    automatisiert durchführen, 5
    Lizenzen, 9
    Produkte hochrüsten, 47
    Produkte modifizieren, 47
    Produkte reparieren, 49
    Protokoll, 39
    Software anzeigen, 45
    starten, 41
    Support Package, 55
    Systemvoraussetzungen, 27
    Updates, 55
    Virtuelle Umgebung (Private Cloud), 5
    Zielverzeichnis, 5
Installation von License Keys, 17, 18
Intrusion Detection System (IDS), 31

L

License Key, 18
    Umgang mit License Keys, 18
License Manager Panel Plugin, 22
Lizenz, 24
    defekt, 20
    defekte Lizenz, 20
    Starten ohne gültige Lizenz, 15
    WinCC Runtime Professional asiatisch, 24
Lizenz für PC-CPs, 25
Lizenzen, 9
Lizenzierung Bediengerät
    nicht PC-basiert, 22

M

Migrations-Tool
    Deinstallation, 71
    Systemvoraussetzungen, 69
    Vertrieb und Bezugsquellen, 70
Multiuser Engineering
    Deinstallation des Multiuser-Servers, 78
    Einführung, 73
    Installation des Multiuser-Servers als Standalone-Installation, 73
    Installation des Multiuser-Servers mit dem TIA Portal, 73
    Installation des Multiuser-Servers starten, 76
    Regeln und Voraussetzung für die Installation, 76

N

Nicht lizenzierter Modus
    Bediengeräte, 22
    Engineering-System, 15
    Runtime, 23

P

PDF-Dateien anzeigen, 27

R

Restore-Vorgang
    License Key, 19

S

Sicherheitsprogramm, 31
Software-Voraussetzungen, 29
Support Package installieren, 55
Systemvoraussetzungen, 29

U

ungültige Lizenz
    bei Zeitzonenwechsel, 19
Update installieren, 55

V

Verschlüsselungssoftware, 31
Virenscanner, 6, 31
Virtuelle Maschine (VM)
    unterstützte Virtualisierungsplattformen, 30

W

Welcome Tour anzeigen, 27
WinCC Runtime Professional asiatisch
    Lizenzierung, 24
Windows, 29

Z

Zerstörung von License Keys, 17

SIEMENS

Übersicht über die Migrationsmöglichkeiten 1

Projekte in ein TIA Portal-Projekt migrieren 2

PLC-Programme in eine S7-1500-CPU / ET 200SP migrieren 3

S7-1200 auf Firmware ab V4 migrieren 4

Projekte und Programme migrieren

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Übersicht über die Migrationsmöglichkeiten....7

2 Projekte in ein TIA Portal-Projekt migrieren....9

2.1 Migration von Projekten mit dem TIA Portal....9

2.2 Migrationsfähigkeit von Hardware-Komponenten prüfen....12

2.3 Vorbereitung von Projekten mit dem Migrations-Tool....13
2.3.1 Migration von Projekten mit dem Migrations-Tool....13
2.3.2 Migrations-Tool aufrufen....14
2.3.3 Migrationsdatei erstellen....15

2.4 Projekte migrieren....16

2.5 Verlauf der Migration anzeigen....18

2.6 Protokoll der Migration anzeigen....19

2.7 STEP 7-Projekte migrieren....20
2.7.1 Migration von STEP 7-Projekten....20
2.7.2 Voraussetzungen für die Migration von STEP 7-Projekten....21
2.7.3 STEP 7-Projekte auf ihre Migrationsfähigkeit prüfen....22
2.7.4 Konsistenzprüfung im Ausgangsprojekt....23
2.7.5 Entfernen nicht unterstützter Hardware-Komponenten....24
2.7.6 Besonderheiten bei der Migration....25
2.7.6.1 Gerätenamen....25
2.7.6.2 PROFINET IO-Konfigurationen....25
2.7.6.3 PROFIBUS-Konfigurationen....27
2.7.6.4 GSD- und GSDML-Geräte....27
2.7.6.5 Verbindungen....27
2.7.6.6 Multiprojekte....28
2.7.6.7 Migration von Programmbausteinen....28
2.7.6.8 Migration von KOP-/FUP-/AWL-Programmen....33
2.7.6.9 Migration von GRAPH-Programmen....34
2.7.6.10 Migration von SCL-Programmen....35
2.7.6.11 Meldungen....42
2.7.6.12 Systemfehler melden....42
2.7.6.13 Migration von Projekttexten....42

2.8 WinCC-Projekte migrieren....44
2.8.1 WinCC flexible 2008 Projekte migrieren....44
2.8.1.1 Grundlagen (WinCC flexible)....44
2.8.1.2 Engineering-Daten migrieren (WinCC flexible)....50
2.8.1.3 Runtime-Daten migrieren (WinCC flexible)....72
2.8.1.4 Integrierte Projekte migrieren (WinCC flexible)....76
2.8.1.5 Referenz (WinCC flexible)....80
2.8.2 WinCC V7 Projekte migrieren....95
2.8.2.1 Grundlagen (WinCC V7)....95
2.8.2.2 Engineering-Daten migrieren (WinCC V7)....103
2.8.2.3 Kopieren der Runtime-Daten (WinCC V7)....135

2.8.2.4 Referenz (WinCC V7)....135
2.8.2.5 Integrierte Projekte....137

2.9 Integrierte Projekte migrieren....142
2.9.1 Migration eines integrierten Projekts....142
2.9.2 Integrierte Projekte nacharbeiten....144
2.9.3 Unspezifizierte CPUs in spezifizierte CPUs umwandeln....145
2.9.4 Integrierte HMI-Verbindung erstellen....147
2.9.5 HMI-Variablen neu verknüpfen....149
2.9.6 Nicht integrierte HMI-Verbindung löschen....150

3 PLC-Programme in eine S7-1500-CPU / ET 200SP migrieren....153

3.1 Grundlagen zur Migration von PLC-Programmen....153

3.2 Migration durchführen....156

3.3 Protokolldatei anzeigen....158

3.4 Besonderheiten bei der Migration von PLC-Programmen....159
3.4.1 Hinweise zur Migration von PLC-Programmen....159
3.4.1.1 Migration von Organisationsbausteinen....159
3.4.1.2 Migration von Hardware-Kennungen....160
3.4.1.3 Migration von IEC-Zeiten und IEC-Zählern....160
3.4.1.4 Migration von CPU-Datenbausteinen....161
3.4.1.5 Migration von Diagnosefunktionen....162
3.4.1.6 Migration von Absolutzugriffen auf Lokaldaten....162
3.4.1.7 Bausteinparameter in S7-1500....164
3.4.1.8 Multiinstanzfähigkeit in S7-1500....165
3.4.1.9 Gleitpunktzahlen in S7-1500....165
3.4.1.10 Schreiben einzelner Zeichen eines STRING in S7-1500....166
3.4.1.11 Zugriffe auf das Statuswort in S7-1500....166
3.4.1.12 Laden von Softwareänderungen in S7-1500....167
3.4.2 Migration von Anweisungen nach S7-1500....167
3.4.2.1 Hinweise zur Migration von Anweisungen....167
3.4.2.2 Übersicht der nicht migrierbaren Anweisungen....170
3.4.2.3 Migration von Datenbausteinanweisungen....179
3.4.2.4 Migration der Anweisung RDSYSST....181
3.4.2.5 Migration der Anweisung T_CONV....183
3.4.2.6 Migration von Meldungen mit Begleitwerten....183
3.4.2.7 Migration von Tabellenanweisungen....187
3.4.2.8 Migration von Kommunikationsanweisungen....191
3.4.3 Migration von KOP/FUP-Programmen nach S7-1500....197
3.4.3.1 Hinweise zur Migration von KOP/FUP-Programmen....197
3.4.3.2 Teilqualifizierte Adressen in KOP/FUP....198
3.4.3.3 Zugriffe auf Statusbits in KOP/FUP....199
3.4.4 Migration von AWL-Programmen nach S7-1500....201
3.4.4.1 Hinweise zur Migration von AWL-Programmen....201
3.4.4.2 Verknüpfungsketten und Sprünge....203
3.4.4.3 Werteübergabe über Register bei Sprachwechsel....204
3.4.4.4 Werteübergabe über Register oder das Statuswort beim Bausteinaufruf....205
3.4.4.5 Werteübergabe über Register mit den Anweisungen CC und UC....206
3.4.4.6 Vollqualifizierte Adressen in AWL....208
3.4.4.7 Teilqualifizierte Adressen in AWL....209
3.4.4.8 Zugriffe auf den Instanz-DB in AWL....211

3.4.5 Migration von SCL-Programmen nach S7-1500....213
3.4.5.1 Hinweise zur Migration von SCL-Programmen....213
3.4.5.2 Indirekte Adressierung in SCL....214
3.4.5.3 Anweisungen in SCL....216
3.4.6 Migration von GRAPH-Programmen nach S7-1500....218
3.4.6.1 Hinweise zur Migration von GRAPH-Programmen....218
3.4.6.2 Änderungen im GRAPH-Programm....220
3.4.6.3 Anpassung der Schnittstelle von GRAPH-Bausteinen....222

4 S7-1200 auf Firmware ab V4 migrieren....225

4.1 Grundlagen zum Umstieg auf V4....225

4.2 Migration auf V4 durchführen....228

4.3 Besonderheiten nach dem Umstieg auf V4....230

Index....233

Übersicht über die Migrationsmöglichkeiten

1

Migrationswege

Wenn Sie bestehende Projekte oder Programme auf der neuesten Version des TIA Portals und der S7-1500 weiter verwenden wollen, haben Sie verschiedene Möglichkeiten, die Projekte zu migrieren.

Die folgende Grafik gibt einen Überblick über die Migrationsmöglichkeiten:

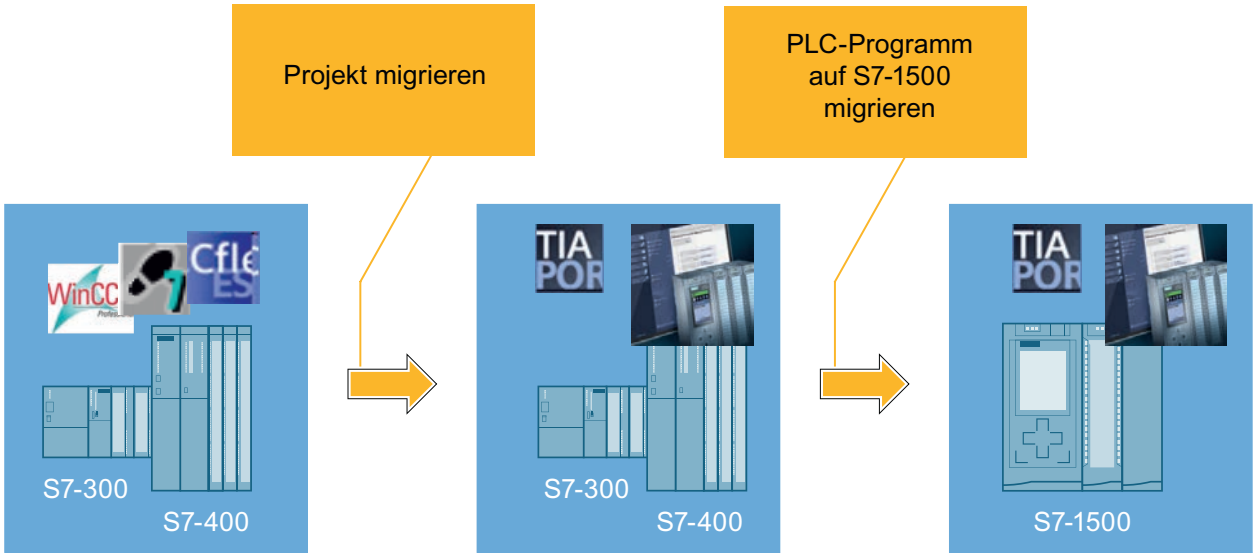

Projekte in das TIA Portal migrieren

Mit der Funktion "Projekt migrieren" migrieren Sie Projekte, die außerhalb des TIA Portals z. B. mit STEP 7, WinCC, WinCC flexible oder mit SINUMERIK erstellt wurden.

Das Ergebnis der Projektmigration ist ein TIA Portal-Projekt, das Sie mit Ihrer bestehenden Hardware der Familie S7-300/400 und den vorhandenen HMI-Geräten nutzen können.

PLC-Programme von S7-300/400 auf S7-1500 migrieren

Um ein PLC-Programm innerhalb des TIA Portals von einem Gerät der Baureihe S7-300/400 in eine S7-1500 zu migrieren, nutzen Sie die PLC-Migration.

Das Ergebnis der PLC-Migration ist ein ablauffähiges PLC-Programm, das automatisch soweit wie möglich an die neue Systemarchitektur der S7-1500 angepasst wurde.

PLC-Programm für S7-1500 optimieren

Optional haben Sie die Möglichkeit, Ihr Programm für die S7-1500 zu optimieren, indem Sie die neu eingeführten Programmiertechniken einsetzen. Durch die Optimierung nutzen Sie die höhere Performance, die innovierte Speichertechnologie und die neuen Funktionen des S7-1500-Systems effektiv.

Projekte hochrüsten

Projekte aus früheren Versionen des TIA Portals können Sie ebenfalls weiterverwenden. Diese Projekte müssen Sie jedoch nicht migrieren. Projekte aus früheren Versionen rüsten Sie auf die aktuelle Produktversion hoch. Weitere Informationen zur Kompatibilität sowie zum Hochrüsten von Projekten finden Sie im Kapitel "AUTOHOTSPOT".

Hinweis

Weitere Unterstützung zur Migration

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur PLC-Migration:

Migration Gesamtsysteme:

http://support.automation.siemens.com/WW/view/de/83558085 (http://support.automation.siemens.com/WW/view/de/83558085)

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Migration Visualisierung:

http://support.automation.siemens.com/WW/view/de/76878921 (http://support.automation.siemens.com/WW/view/de/76878921)

Migration Kommunikation:

http://support.automation.siemens.com/WW/view/de/83558087 (http://support.automation.siemens.com/WW/view/de/83558087)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Projekte in ein TIA Portal-Projekt migrieren 2

2.1 Migration von Projekten mit dem TIA Portal

Migration von bereits bestehenden Projekten

Sie können Projekte aus früheren Automatisierungslösungen in das TIA Portal migrieren. Bei jeder Migration wird ein neues Projekt für die migrierten Daten angelegt, mit dem Sie anschließend weiterarbeiten können. Eventuell bereits geöffnete TIA Portal-Projekte werden zuvor geschlossen.

Die Migration wird anschließend in der Tabelle zum Projektverlauf angezeigt. Von dort aus haben Sie Zugriff auf das Migrationsprotokoll, das für die Migration automatisch angelegt wird.

Prüfen Sie nach der Migration von Hardware-Konfigurationen und Programmbausteinen aus früheren Automatisierungslösungen zuerst die Funktionsfähigkeit des migrierten Projektes, bevor Sie es im Produktivbetrieb verwenden.

Unterstützte Produkte für die Migration

Im Kapitel "Systemübersicht STEP 7 und WinCC" erfahren Sie, welche Produkte für das TIA Portal erhältlich sind. Alle dort gelisteten Produkte werden grundsätzlich vom TIA Portal bei der Migration unterstützt.

Welche Voraussetzungen zusätzlich erfüllt sein müssen, ist von den verwendeten Ausgangsprodukten und von den aktuell installierten Produkten abhängig. Nähere Informationen zu den Migrationsmöglichkeiten für Ihre Produkte erhalten Sie beispielsweise im Siemens Industry Online Support und in der Dokumentation zu Ihren Softwareprodukten.

Siehe auch: AUTOHOTSPOT

Vorgehen bei der Migration

Der Migrationsprozess gliedert sich grundsätzlich in die folgenden Schritte:

1. Vorbereiten des Ausgangsprojekts
   Wenn auf dem PG/PC mit dem TIA Portal die Software zum Bearbeiten des Ausgangsprojekts nicht oder nicht in vollem Umfang installiert ist, oder wenn es sich bei dem Ausgangsprojekt um ein integriertes Projekt handelt, dann müssen Sie das Ausgangsprojekt zunächst in eine Migrationsdatei wandeln. Installieren Sie dafür das Migrations-Tool auf einem PG/PC, auf dem die notwendige Software zur Bearbeitung des Ausgangsprojekts installiert ist. Wandeln Sie anschließend das Ausgangsprojekt mithilfe des Migrations-Tools um und kopieren Sie die Datei auf den PG/PC mit der Installation des TIA Portals. Sie können diesen Schritt auslassen, wenn sich das Ausgangsprojekt und die dazugehörige Ausgangssoftware auf demselben PG/PC wie das TIA Portal befindet und wenn es sich bei dem Ausgangsprojekt nicht um ein integriertes Projekt handelt.
2. Migration durchführen
   Führen Sie die Migration innerhalb des TIA Portals durch. Geben Sie bei der Migration entweder die Migrationsdatei, die Sie mit dem Migrations-Tool erstellt haben, als Quelle an, oder geben Sie das Ausgangsprojekt an, wenn alle notwendige Software installiert ist.
3. Kontrollieren des Migrationsprotokolls
   Zu jeder Migration wird ein Migrationsprotokoll angelegt. Darin finden Sie Hinweise auf veränderte Projektbestandteile. Das Protokoll können Sie in der Projektnavigation unter "Gemeinsame Daten > Protokolle" oder über den Projektverlauf aufrufen. Nach dem Abschluss der Migration wird das Migrationsprotokoll im TIA Portal angezeigt. Kontrollieren Sie das Protokoll nach Abschluss der Migration.
   Wenn die Migration fehlgeschlagen ist, dann wird im Projektverzeichnis unter "\Logs" eine XML-Datei als Protokoll angelegt. Diese können Sie mit einem beliebigen XML-Editor öffnen und die Ursachen für die fehlgeschlagene Migration ersehen.
4. Nacharbeiten des migrierten Projekts
   Da die Konfigurationen des Ausgangsprojekts nicht immer vollständig mit dem TIA Portal kompatibel sein können, werden im migrierten Projekt gegebenenfalls nicht alle Konfigurationen identisch übernommen. Arbeiten Sie daher im Migrationsprotokoll verzeichnete Hinweise systematisch nach. Wenn Sie die Hardware-Konfiguration nicht in die Migration eingeschlossen haben, müssen Sie zusätzlich die unspezifizierten Geräte in die passende Hardware umwandeln.

Einschließen der Hardware-Konfiguration in die Migration

Standardmäßig werden bei der Migration nur die Software-Bestandteile des Projekts berücksichtigt. Für die im Ausgangsprojekt enthaltenen Geräte wird jeweils ein unspezifiziertes Gerät im migrierten Projekt erzeugt. Die Hardware- und Netzkonfigurationen sowie Verbindungen werden nicht migriert. Wandeln Sie daher im Anschluss an die Migration die unspezifizierten Geräte wieder in ein passendes Gerät um und stellen Sie manuell etwaige Netzkonfigurationen und Verbindungen wieder her.

Wenn Sie sich sicher sind, dass die verwendete Hardware im Ausgangsprojekt ein passendes Äquivalent im TIA Portal besitzt, dann können Sie die Hardware-Konfiguration bei der Migration einschließen. In diesem Fall wird sowohl die Hardware-Konfiguration als auch die Software migriert. Welche Hardware-Komponenten unterstützt werden, können Sie mit einem Tool (Seite 12) prüfen.

Siehe auch

Protokoll der Migration anzeigen (Seite 19)

2.2 Migrationsfähigkeit von Hardware-Komponenten prüfen

Einführung

Siemens bietet Ihnen ein Tool, mit dem Sie prüfen können, ob die verwendete Hardware-Konfiguration in einem Ausgangsprojekt für die Migration in das TIA Portal geeignet ist.

Über GSD- oder GSDML-Dateien eingebundene Komponenten können nicht geprüft werden. Prüfen Sie für solche Baugruppen manuell im TIA Portal nach, ob die Baugruppen im Hardware-Katalog vorhanden sind. Falls die Baugruppen nicht vorhanden sind, installieren Sie die notwendigen GSD- oder GSDML-Dateien im TIA Portal. Die notwendigen Dateien erhalten Sie vom Hersteller der Komponenten.

Im Ergebnis der Prüfung sehen Sie, welche Software-Produkte und Lizenzen auf dem PG/PC mit der Installation des TIA Portals verfügbar sein müssen, um eine Migration durchzuführen. Weiterhin sehen Sie, ab welcher Firmware-Version die einzelnen Module des Ausgangsprojekts im TIA Portal unterstützt werden. Das Ergebnis der Prüfung können Sie in eine Microsoft Excel- oder PDF-Datei exportieren.

Download

Das Tool zur Prüfung der Migrationsfähigkeit finden Sie zum Download in den FAQs des Siemens Industry Online Supports unter der Beitragsnummer 60162195 (http://support.automation.siemens.com/WW/view/de/60162195).

Quelldateien für die Prüfung

Zur Prüfung benötigen Sie eine der folgenden Quelldateien, welche die Artikelnummer der im Ausgangsprojekt verwendeten Hardware enthält:

- .cfg-Datei
  Sie können die .cfg-Datei aus HW Konfig (STEP 7) über den Menübefehl "Exportieren als .cfg-Datei" im Menü "Station" exportieren. Die .cfg-Datei enthält alle MLFBs der in der aktuell geöffneten Station verwendeten Geräte.
- Microsoft Excel-Datei (im .xls-Dateiformat)
  Sie können unabhängig vom verwendeten Ausgangsprojekt eine Microsoft Excel-Liste erstellen, in der alle MLFBs der Geräte enthalten sind, die Sie migrieren möchten.
- Datei im .csv-Format
  Anstatt einer Microsoft Excel-Liste können Sie zu prüfenden Artikelnummern ebenso in einer .csv-Datei speichern. Verwenden Sie dazu einen einfachen Text-Editor und geben Sie die MLFBs durch Komma getrennt und ohne Leerzeichen nach dem Komma ein. Speichern Sie die Text-Datei mit der Dateiendung ".csv" ab.

Siehe auch

Tool zur Prüfung der Migrationsfähigkeit (http://support.automation.siemens.com/WW/view/de/60162195)

2.3 Vorbereitung von Projekten mit dem Migrations-Tool

2.3.1 Migration von Projekten mit dem Migrations-Tool

Vorbereitung der Migration

In vielen Fällen wird sich ein Projekt, das Sie migrieren möchten, nicht auf demselben PG/PC befinden, auf dem die aktuelle Version des TIA Portals installiert ist. Das Ausgangsprojekt muss daher für die Migration zunächst in ein kompatibles Format umgewandelt werden. Dasselbe gilt für integrierte Projekte.

Nach dem Erstellen der Migrationsdatei kopieren Sie die Migrationsdatei auf den PG/PC mit der aktuellen Installation des TIA Portals. Im TIA Portal geben Sie die Migrationsdatei als Quelle für die Migration an und können ein Projekt im aktuellen Dateiformat des TIA Portals erstellen.

Vorgehen bei der Migration mit dem Migrations-Tool

Um eine Migration mit dem Migrations-Tool vorzubereiten, sind die folgenden Schritte notwendig:

1. Installieren Sie auf dem PG/PC mit dem Quellprojekt das Migrations-Tool. Laden Sie dazu die Installationsdatei aus dem Siemens Industry Online Support herunter oder installieren Sie das Migrations-Tool von der Setup-DVD des TIA Portals.
2. Starten Sie das Migrations-Tool und wandeln Sie mithilfe des Migrations-Tools das Quellprojekt in das Migrationsdateiformat mit der Dateiendung ".am15" um.
   Achten Sie bei diesem Schritt besonders darauf, dass die gesamte, für die Bearbeitung des Quellprojekts notwendige Software auf dem PG/PC installiert ist. Dazu gehören auch alle notwendigen Servicepacks, Hardware-Support-Packages und die gesamte Erweiterungssoftware, die zur Bearbeitung des Ausgangsprojekts notwendig ist. Sind einzelne Produkte nicht installiert, kann die Migration nicht oder nur unvollständig durchgeführt werden.
3. Kopieren Sie die Migrationsdatei auf das Zielsystem mit einer aktuellen Installation des TIA Portals.
   Beachten Sie, dass auf dem Zielsystem sämtliche Software installiert sein muss, die zur Projektierung aller in der Migrationsdatei enthaltenen Geräte notwendig ist.
4. Führen Sie die Migration innerhalb des TIA Portals durch und geben Sie als Quelle die Migrationsdatei mit der Endung ".am15" an.
5. Kontrollieren Sie nach der Migration das Migrationsprotokoll und arbeiten Sie darin verzeichnete Hinweise im neu erstellten Projekt systematisch nach. Beachten Sie nach dem ersten Übersetzen der Konfiguration insbesondere auch die Hinweise im Inspektorfenster.

Einschließen der Hardware-Konfiguration in die Migration

Standardmäßig werden bei der Migration nur die Software-Bestandteile des Projekts berücksichtigt. Für die im Ausgangsprojekt enthaltenen Geräte wird jeweils ein unspezifiziertes Gerät im migrierten Projekt erzeugt. Die Hardware- und Netzkonfigurationen sowie Verbindungen werden nicht migriert. Wandeln Sie daher im Anschluss an die Migration die unspezifizierten Geräte wieder in ein passendes Gerät um und stellen Sie manuell etwaige Netzkonfigurationen und Verbindungen wieder her.

Wenn Sie sich sicher sind, dass die verwendete Hardware im Ausgangsprojekt ein passendes Äquivalent im TIA Portal besitzt, dann können Sie die Hardware-Konfiguration bei der Migration einschließen. In diesem Fall wird sowohl die Hardware-Konfiguration als auch die Software migriert. Welche Module unterstützt werden, können Sie mit einem Tool prüfen.

Siehe auch

Migration von Projekten mit dem TIA Portal (Seite 9)

Projekte migrieren (Seite 16)

Migrations-Tool aufrufen (Seite 14)

Migrationsdatei erstellen (Seite 15)

2.3.2 Migrations-Tool aufrufen

Migrations-Tool starten

Bei der Installation wird im Startmenü unter "Siemens Automation > Migration Tool" standardmäßig eine Verknüpfung "Migration to TIA Portal V15.1" angelegt. Rufen Sie diese Verknüpfung auf.

Alternativ können Sie das Migrations-Tool direkt über den Windows-Explorer aufrufen. Standardmäßig wird das Migrations-Tool bei der Installation in einem der folgenden Verzeichnisse gespeichert:

- Auf einem 64 Bit-Betriebssystem:
  C:\Program Files (x86)\Siemens\Automation\MIGTOOL_V15_1\Bin
- Auf einem 32 Bit-Betriebssystem:
  C:\Program Files\Siemens\Automation\MIGTOOL_V15_1\Bin

Um das Migrations-Tool zu starten, rufen Sie die Datei "Siemens.Automation.MigrationApplication.exe" in einem der Verzeichnisse auf.

Siehe auch

Migrationsdatei erstellen (Seite 15)

2.3.3 Migrationsdatei erstellen

Im Folgenden erfahren Sie, wie Sie das Ausgangsprojekt mithilfe des Migrations-Tools in eine für das TIA Portal lesbare Migrationsdatei umwandeln. Diese Datei wird nach dem Umwandeln auf das Zielsystem übertragen und dort migriert.

Sie können festlegen, ob die Migrationsdatei das gesamte Projekt inklusive der kompletten Hardware-Konfiguration und der zugehörigen Software enthalten soll, oder ob Sie nur die Software migrieren möchten.

Voraussetzung

- Es ist für alle im Ausgangsprojekt verwendeten Konfigurationen die passende, ursprüngliche Software mit einer gültigen Lizenz installiert.
- Das Ausgangsprojekt ist nicht mit einem Zugriffsschutz versehen.
- Das Ausgangsprojekt muss sich in einem konsistenten Zustand befinden, da eine einwandfreie Migration sonst nicht sichergestellt ist.

Vorgehen

Um die Migrationsdatei zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Feld "Storage Location (Path)" den Pfad der Quelldatei für die Migration.
2. Geben Sie an, welche Projektanteile migriert werden:
   - Aktivieren Sie das Optionskästchen "Include HW and Network data during the migration", um nicht nur die Software, sondern auch die vollständigen Hardware-Anteile und die Netzkonfiguration des Projekts zu migrieren.
   - Aktivieren Sie das Optionskästchen "Copy SCADA runtime data", wenn Sie zusätzlich zu den Daten des Engineering-Systems auch die Runtime-Daten, wie z. B. Meldearchive, Variablenarchive oder Anwenderarchive, migrieren möchten.
3. Wählen Sie im Feld "Intermediate file" den Pfad und den Dateinamen für die Migrationsdatei.
4. Klicken Sie auf die Schaltfläche "Migrate".

Ergebnis:

Eine Migrationsdatei wird erstellt. Kopieren Sie im Anschluss diese Datei auf das Zielsystem und migrieren Sie diese Datei im TIA Portal.

Siehe auch

Projekte migrieren (Seite 16)

Migrations-Tool aufrufen (Seite 14)

Migration von Projekten mit dem Migrations-Tool (Seite 13)

2.4 Projekte migrieren

Voraussetzung

- Es liegt eine bereits in das Format ".am15" gewandelte Datei vor oder es ist für alle im Ausgangsprojekt verwendeten Konfigurationen die passende, ursprüngliche Software mit einer gültigen Lizenz installiert.
- Das Ausgangsprojekt ist nicht mit einem Zugriffsschutz versehen.
- Das Ausgangsprojekt muss sich in einem konsistenten Zustand befinden, da eine einwandfreie Migration sonst nicht sichergestellt ist.

Beachten Sie zusätzliche Hinweise zu den Voraussetzungen in der Hilfe zu den jeweiligen, installierten Produkten.

Hinweis

Ruhezustand des Systems während der Migration

Das System sollte während einer laufenden Migration nicht in Standby oder in den Ruhezustand versetzt werden. Andernfalls wird die Migration abgebrochen.

Vorgehen

Um ein Projekt zu migrieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Projekt migrieren".
   Der Dialog "Projekt migrieren" wird geöffnet.
2. Geben Sie im Feld "Quellpfad" den Pfad und den Dateinamen für das zu migrierende Projekt an. Wählen Sie entweder ein Projekt im Migrations-Format ".am15" oder im Format des Ausgangsprojekts.
3. Aktivieren Sie das Optionskästchen "Hardware-Konfiguration einschließen", um die Hardware-Konfiguration in die Migration einzuschließen.
   Wenn Sie eine Migrationsdatei gewählt haben, die mit dem Migrations-Tool erstellt wurde, ist das Optionskästchen nicht aktivierbar. In diesem Fall müssen Sie bereits bei der Umwandlung mit dem Migrations-Tool angeben, ob Sie die Hardware-Konfiguration bei der Migration einschließen möchten.
4. Aktivieren Sie das Optionskästchen "Daten von WinCC Runtime Professional kopieren", wenn Sie zusätzlich zu den Daten des Engineering-Systems auch die Runtime-Daten, wie z. B. Meldearchive, Variablenarchive oder Anwenderarchive, migrieren möchten.
   Wenn Sie eine Migrationsdatei gewählt haben, die mit dem Migrations-Tool erstellt wurde, ist das Optionskästchen nicht aktivierbar. In diesem Fall müssen Sie bereits bei der Umwandlung mit dem Migrations-Tool angeben, ob Sie die SCADA Runtime-Daten bei der Migration einschließen möchten.
5. Wählen Sie im Feld "Projektname" einen Namen für das neue Projekt.
6. Wählen Sie im Feld "Zielpfad" einen Pfad, unter dem das neue Projekt angelegt wird.
7. Geben Sie im Feld "Autor" Ihren Namen oder den Namen eines anderen Projektverantwortlichen an.

8. Tragen Sie ggf. einen Kommentar in das Feld "Kommentar" ein.
9. Klicken Sie auf "Migrieren".

Ergebnis

Das Ausgangsprojekt wird umgewandelt und nach dem Fertigstellen wird eine Meldung angezeigt. Anschließend wird das neu angelegte Projekt in der Projektansicht geöffnet und das Migrationsprotokoll wird im TIA Portal geöffnet.

Sollte die Migration fehlgeschlagen sein, wird dennoch ein Projektverzeichnis angelegt und darin ein Migrations-Protokoll in Form einer XML-Datei erzeugt. In der Abschlussmeldung nach der Migration finden Sie einen Link zu dieser XML-Datei. Klicken Sie auf den Link, um die XML-Datei zu öffnen. Alternativ finden Sie die XML-Datei im Projektverzeichnis unter "\Logs".

Siehe auch

Integrierte Projekte nacharbeiten (Seite 144)

Protokoll der Migration anzeigen (Seite 19)

Migration von Projekten mit dem Migrations-Tool (Seite 13)

Migrationsdatei erstellen (Seite 15)

2.5 Verlauf der Migration anzeigen

Ist ein Projekt durch eine Migration entstanden, dann wird die Migration in der Tabelle zum Projektverlauf aufgeführt. In der Tabelle kann das Migrationsprotokoll aufgerufen werden. Außerdem wird der Zeitpunkt der Migration angezeigt.

Vorgehen

Um eine Übersichtstabelle mit der Migration anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das geöffnete Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Eigenschaften".
   Der Dialog mit den Eigenschaften des Projekts wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die Gruppe "Projektverlauf ".
   Die Übersichtstabelle wird geöffnet.

2.6 Protokoll der Migration anzeigen

Zu jeder erfolgreichen Migration wird ein Protokoll angelegt. Das Protokoll enthält folgende Informationen:

- Migrierte Objekte
- Änderungen an Objekten, die während der Migration vorgenommen wurden
- Fehler, die bei der Migration aufgetreten sind
- Gegebenenfalls einen Link zu weiterer Hilfe bei bestimmten Ereignissen.
  Weitere Hilfe erhalten Sie in diesem Fall durch einen Klick auf das Fragezeichen.

Vorgehen

Um die Protokolldatei zur Migration anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Gemeinsame Daten > Protokolle" in der Projektnavigation.
2. Doppelklicken Sie das gewünschte Protokoll in der Liste.
   Die Inhalte des Protokolls werden im Arbeitsbereich angezeigt.

Siehe auch

Migration von Projekten mit dem TIA Portal (Seite 9)

2.7 STEP 7-Projekte migrieren

2.7.1 Migration von STEP 7-Projekten

Einführung

Sie können ein Projekt aus SIMATIC STEP 7 V5.4 SP5 migrieren. Sie können das Projekt anschließend im TIA Portal weiterverwenden. Wenn Sie Projekte aus älteren STEP 7-Versionen migrieren möchten, müssen Sie diese zunächst in STEP 7 V5.4 SP5 öffnen, übersetzen und abspeichern. Auch die Migration integrierter Projekte mit Geräten aus anderen Softwareprodukten ist möglich.

Migrationsumfang

Generell können Sie alle Konfigurationen und Objekte aus SIMATIC STEP 7 V5.4 SP5 migrieren, die auch die installierte Version des TIA Portals unterstützt. Dazu zählen beispielsweise die folgenden Geräte und Konfigurationen:

- Geräte der S7-300 und S7-400-Familie.
- PROFIBUS-Konfigurationen mit angeschlossener dezentraler Peripherie. Dazu zählen auch GSD-basierte Slaves, I-Slaves und direkter Datenaustausch.
- PROFINET-Konfigurationen mit dezentraler Peripherie. Dazu zählen auch GSD-basierte Geräte und I-Devices.
- Netzwerkkonfigurationen
- Verbindungen
- Bausteine, die in den Sprachen KOP, FUP oder AWL, S7-SCL, S7-GRAPH erstellt wurden
- PLC-Variablen
- Anwenderdefinierte Datentypen (UDT)
- Meldungen und Meldeklassen
- Alarme
- Anwenderdefinierte Attribute (UDA)
- Anwender-Textbibliotheken

Einschließen der Hardware-Konfiguration in die Migration

Standardmäßig werden bei der Migration nur die Software-Bestandteile des Projekts berücksichtigt. Für die im Ausgangsprojekt enthaltenen Geräte wird jeweils ein unspezifiziertes Gerät im migrierten Projekt erzeugt. Die Hardware- und Netzkonfigurationen sowie Verbindungen werden nicht migriert. Wandeln Sie daher im Anschluss an die Migration die unspezifizierten Geräte wieder in ein passendes Gerät um und stellen Sie manuell etwaige Netzkonfigurationen und Verbindungen wieder her.

Wenn Sie sich sicher sind, dass die verwendete Hardware im Ausgangsprojekt ein passendes Äquivalent im TIA Portal besitzt, dann können Sie die Hardware-Konfiguration bei der

Migration einschließen. In diesem Fall wird sowohl die Hardware-Konfiguration als auch die Software migriert.

Hinweis

Weitere Unterstützung zur Migration

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur Migration:

Migration Steuerungen (http://support.automation.siemens.com/WW/view/de/83557459)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich an den SIMATIC Customer Support.

Siehe auch

Systemfehler melden (Seite 42)

Migration von Programmbausteinen (Seite 28)

2.7.2 Voraussetzungen für die Migration von STEP 7-Projekten

Für die Migration gelten verschiedene Anforderungen an die installierte Software auf dem ursprünglichen PG/PC und an das Ausgangsprojekt.

Anforderungen an den ursprünglichen PG/PC

Für den PG/PC gelten die folgenden Anforderungen:

- Eine lizenzierte Version von STEP 7 V5.4 SP5 oder höher (V5.5 SP4 empfohlen) muss installiert sein. Bei Projekten, die ausschließlich PC-Stationen enthalten, ist es ausreichend, wenn die SIMATIC NET PC-Software installiert ist.
- Für alle im Projekt verwendeten Konfigurationen muss die entsprechende Zusatzsoftware mit einer gültigen Lizenz installiert sein, z. B. Optionspakete
- Alle HSPs zu Baugruppen, die nicht im Hardware-Katalog enthalten sind, müssen installiert sein.
- Alle im Projekt verwendeten GSD-Dateien müssen den Spezifikationen entsprechen.
- Sie müssen mit Administrationsrechten am Betriebssystem angemeldet sein.
- Es muss eines der folgenden Software-Produkte installiert sein:
  - Das TIA Portal mit STEP 7 Professional ab V13 SP1
    Wenn Sie nicht mit der aktuellsten Version des TIA Portals auf dem ursprünglichen PG/PC migrieren, ist der Funktionsumfang des Projekts in neueren Versionen des TIA Portals zunächst beschränkt. Um den vollen Funktionsumfang von neueren Versionen nutzen zu können, müssen Sie das Projekt zuerst hochrüsten.
  - Das Migrations-Tool

Anforderungen an das Ausgangsprojekt

- Das Ausgangsprojekt darf nicht mit einem Zugriffsschutz versehen sein.
- Die Hardware und die Software müssen konsistent sein.
- Die Meldenummernvergabe muss auf CPU-weit eingestellt sein.
- Das Projekt darf keine geschützten Bausteine mit Zeitstempelkonflikt enthalten.
- Alle Programme und deren Quellen müssen fehlerfrei übersetzbar sein.
- Alle aufgerufenen Bausteine müssen im Bausteinordner vorhanden sein.
- Nicht aufgerufene Bausteine, insbesondere Instanzdatenbausteine, dürfen nicht im Bausteinordner liegen.
- Es ist empfehlenswert Bausteine unverschlüsselt zu migrieren, da mit S7-Block Privacy verschlüsselte Bausteine im TIA Portal nicht lesbar und nicht änderbar sind.

Anforderungen an den PG/PC mit STEP 7 Professional

- Es muss eines der folgenden Software-Produkte installiert sein:
  - STEP 7 Professional V13 SP1 oder neuer
  - STEP 7 Professional V13 SP1 oder neuer (ohne Lizenz), wenn das zu migrierende Projekt nur Komponenten enthält, die mit der SIMATIC NET PC-Software konfiguriert wurden.
- Für alle im Projekt verwendeten Konfigurationen muss die entsprechende Zusatzsoftware mit einer gültigen Lizenz installiert sein, z. B. Optionspakete.
- Alle HSPs zu Baugruppen, die nicht im Hardwarekatalog enthalten sind, und alle im Projekt verwendeten GSD-Dateien müssen installiert sein.

2.7.3 STEP 7-Projekte auf ihre Migrationsfähigkeit prüfen

Bevor Sie mit der Migration beginnen, prüfen Sie, ob alle notwendigen Voraussetzungen für die Migration im Ausgangsprojekt gegeben sind.

Hinweis

Beachten Sie, dass beim Durchführen der folgenden Schritte die Aktualwerte im Ausgangsprojekt zurückgesetzt werden. Beachten Sie weiterhin, dass das Ausführen der Bausteinkonsistenzprüfung die Startwerte in Instanz-DBs auf die Defaultwerte zurücksetzt. Wenn Sie Technologiefunktionen einsetzen, ist dieses Verhalten unerwünscht.

Vorgehen

Um ein Projekt auf seine Migrationsfähigkeit zu prüfen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Ausgangsprojekt in SIMATIC STEP 7 V5.4 SP5. Für einen korrekten Abgleich dürfen ausschließlich die im TIA Portal vorhandenen Optionspakete und Hardware-Komponenten, einschließlich HSPs und GSD-Dateien, installiert sein.
2. Öffnen Sie die einzelnen Stationen. Wenn beim Öffnen der Stationen keine Meldungen erscheinen, die auf fehlende Komponenten hinweisen, dann sind alle für die Migration notwendigen Komponenten vorhanden.
3. Führen sie für jeden im Projekt vorhandenen Bausteincontainer eine Bausteinkonsistenzprüfung durch.
   Weitere Informationen zur Konsistenzprüfung finden Sie im Kapitel "Konsistenzprüfung im Ausgangsprojekt (Seite 23)".
4. Übersetzen Sie das gesamte Projekt. Wenn beim Übersetzen kein Fehler angezeigt wird, dann sind die Programme migrierbar.
5. Prüfen Sie, ob alle Baugruppen mit einer identischen MLFB und demselben Firmwarestand im Hardware-Katalog des TIA Portals enthalten sind.
6. Prüfen Sie, ob für jede CPU die CPU-weite Meldenummernvergabe aktiviert ist.
7. Übersetzen Sie das Projekt in NetPro. Auch die Übersetzung in NetPro muss fehlerfrei sein.

Siehe auch

Konsistenzprüfung im Ausgangsprojekt (Seite 23)

FAQ-Eintrag zur Konsistenzprüfung (http://support.automation.siemens.com/WW/view/de/5416540)

2.7.4 Konsistenzprüfung im Ausgangsprojekt

Konsistentes Ausgangsprojekt

Die Voraussetzung für die erfolgreiche Migration eines Projekts in das TIA Portal, ist ein konsistentes Ausgangsprojekt. Sie sollten daher vor der Migration eine Konsistenzprüfung im Ausgangsprojekt durchführen. Eine Anleitung zur Konsistenzprüfung in SIMATIC STEP 7 finden Sie in einem FAQ-Eintrag (http://support.automation.siemens.com/WW/view/de/5416540) im Siemens Industry Online Support oder in der Hilfe zu SIMATIC STEP 7.

Startwerte in Instanz-DBs für Technologiebausteine migrieren

Die Übersetzung in der Bausteinkonsistenzprüfung führt dazu, dass Variablen in Instanz-DBs auf die Defaultwerte zurückgesetzt werden. Die eingestellten Startwerte gehen dabei verloren.

Um die Startwerte dennoch zu migrieren, ist eines der folgenden Vorgehen empfehlenswert:

- Für kleine Projekte: Sichern der Instanz-DBs außerhalb des Bausteinordners
  Sichern Sie die Instanz-DBs, bevor Sie die Konsistenzprüfung starten. Im Anschluss an die Konsistenzprüfung kopieren Sie die Instanz-DBs wieder in den Bausteinordner zurück.
- Für umfangreiche Projekte: Erstellen einer Kopie des Projekts
  Erstellen Sie eine Kopie des Projekts und führen Sie die Prüfung mit der Kopie des Projekts durch. Eventuell bei der Konsistenzprüfung gefundene Fehler beheben Sie im Ausgangsprojekt.

Anschließend können Sie das Projekt mit Ihren projektierten Werten migrieren. Damit bleibt auch die Projektierung von Technologiefunktionen erhalten.

2.7.5 Entfernen nicht unterstützter Hardware-Komponenten

Wenn Sie im Ausgangsprojekt Konfigurationen erstellt haben, für die nicht die gesamte passende Software installiert ist, dann können Sie das Projekt trotzdem migrieren, indem Sie die nicht unterstützten Konfigurationen manuell aus dem Projekt löschen. Alternativ können Sie die Funktion "Reorganisieren" verwenden.

Vorgehen

Um das Ausgangsprojekt zu bereinigen, gehen Sie folgendermaßen vor:

1. Verwenden Sie eine Installation von SIMATIC STEP 7 V5.4 SP5, die nur die im TIA Portal vorhandenen Optionspakete und alle benötigten Baugruppen enthält.
2. Speichern Sie damit das Projekt neu ab und wählen beim Speichern die Option "Mit Reorganisation". Dabei werden alle nicht unterstützten Konfigurationen aus dem Projekt entfernt.

Für Baugruppen, die im TIA Portal nur in einer neueren Version oder mit einem neueren Firmwarestand verfügbar sind, verwenden Sie in der Station die Funktion "Objekt tauschen", um eine ältere Baugruppe gegen eine migrierbare auszutauschen.

2.7.6 Besonderheiten bei der Migration

2.7.6.1 Gerätenamen

Gerätenamen von PROFINET IO-Devices

In STEP 7 V5.5 wird der Gerätename eines PROFINET IO-Devices als Name des IO-Devices verwendet. Nach der Migration zu einem TIA Portal-Projekt wird der Gerätename des PROFINET IO-Devices aus dem Ausgangsprojekt folgendermaßen verwendet:

- Als PROFINET-Gerätename
- Als Name für das Interfacemodul
- Als Name der Station
  Für den Namen der Station sind im TIA Portal nicht mehr als 24 Zeichen zulässig. Daher wird der Name der Station bei Bedarf auf einen Namen mit nicht mehr als 24 Zeichen gekürzt. Wenn das Kürzen des Stationsnamens zu mehreren gleichnamigen Stationen führen würde, dann werden die letzten Zeichen des Stationsnamens zur Durchnummerierung der Stationen verwendet.

Namen von Modulen und Racks

In SIMATIC STEP 7 V5.5 müssen Namen von Modulen und Racks nicht eindeutig sein. Im TIA Portal hingegen ist eine eindeutige Benennung erforderlich. Daher werden die Namen der Module und Racks bei Bedarf während der Migration hochgezählt, um eine eindeutige Namensgebung sicherzustellen. Davon ausgenommen sind PROFINET IO-Devices und PROFINET CPs. Bei diesen müssen Sie selbst für eindeutige Namen sorgen.

2.7.6.2 PROFINET IO-Konfigurationen

Eindeutige Namensgebung erforderlich

Die Namen von PROFINET IO-Devices müssen innerhalb desselben Subnetzes eindeutig sein. Dies gilt auch für die Namen von PROFINET CPs. Stellen Sie daher vor der Migration im Ausgangsprojekt eine eindeutige Namensgebung sicher, oder sorgen Sie nach der Migration vor dem ersten Übersetzen für eindeutige Namen.

Medienredundanz bei Sync-Domains

Wenn im Ausgangsprojekt eine Sync-Domain mit aktivierter Medienredundanz vorhanden ist, dann wird die Sync-Domain-Konfiguration bei der Migration übernommen. Sind hingegen mehrere Sync-Domains mit aktivierter Medienredundanz im Ausgangsprojekt vorhanden, dann werden diese entfernt und das Projekt wird ohne die Sync-Domains migriert. In jedem Fall erhalten Sie eine Meldung, welche Sync-Domains übernommen wurden.

Verwenden Sie zur Projektierung der Medienredundanz im Ausgangsprojekt in jedem Fall SIMATIC STEP 7 V5.5.

Taktsynchronität bei PROFINET IO

Beachten Sie bei PROFINET IO-Konfigurationen mit Taktsynchronität Folgendes:

- IRT mit hoher Performance
  Wenn Sie im Ausgangsprojekt bei der Topologieplanung IRT mit hoher Performance projektiert haben, wird die Option bei der Migration auf "RT" umgestellt. Sie erhalten eine Meldung, welche Sie über die Änderung informiert.
- Verwendung des OB 61
  Ist in der ursprünglichen Konfiguration im STEP 7-Projekt der taktsynchrone Modus für PROFINET IO mit OB 61 aktiviert, wird dieser von der Migration deaktiviert. Die Geräte werden trotzdem übernommen. PROFINET IRT-Konfigurationen mit der Option "hohe Flexibilität" ohne Verwendung des OB 61 können migriert werden.

CBA bei IO-Controllern

Haben Sie im Ausgangsprojekt den zyklischen Dienst CBA (Component based Automation) verwendet, wird dieser durch die Migration deaktiviert. Sie erhalten eine Meldung, welche Sie über die Änderung informiert.

Verwendung von Shared Devices

Sind im Ausgangsprojekt Shared Devices projektiert, dann werden diese bei der Migration entfernt. Sie erhalten eine Meldung, welche Sie über das Entfernen informiert.

Transferbereiche

Wenn im Ausgangsprojekt Transferbereiche für den Datenaustausch zwischen IO-Controller und I-Device projektiert sind, dann werden diese bei der Migration entfernt. Sie werden mit einer Meldung informiert, wenn ein Transferbereich durch die Migration entfernt wird.

Update-Gruppen für PROFINET IO

Entfernen Sie gegebenenfalls Update-Gruppen aus dem STEP 7-Projekt, bevor Sie die Migration starten. Sind noch Update-Gruppen projektiert, dann kann die Migration nicht durchgeführt werden.

I-Slaves und I-Devices

Nur verbundene und konsistente I-Slave- und I-Device-Konfigurationen können migriert werden. Ist die Konfiguration inkonsistent, werden die betroffenen Konfigurationen von der Migration ausgenommen.

Nicht verbundene PROFINET IO-Systeme

Stationen mit nicht verbundenen PROFINET IO-Systemen können migriert werden. Die IO-Devices und die PROFINET IO-Systeme liegen nach der Migration unverbunden in der Netzsicht vor. Verbinden Sie diese IO-Devices im Anschluss an die Migration mit einem IO-Controller oder löschen Sie nicht benötigte IO-Devices.

2.7.6.3 PROFIBUS-Konfigurationen

Taktsynchroner PROFIBUS

Die Migration von taktsynchron laufenden PROFIBUS-Konfigurationen ist möglich. Sie sollten jedoch nach der Migration das Projekt neu übersetzen, um eventuell auftretende Inkonsistenzen auszuschließen. Achten Sie ebenso auf geringfügig veränderte SDBs.

Verwaiste Mastersysteme

Stationen mit verwaisten DP-Mastersystemen können migriert werden. Die DP-Slaves des verwaisten DP Mastersystems liegen nach der Migration unverbunden in der Netzsicht vor. Verbinden Sie diese DP-Slaves im Anschluss an die Migration mit einem DP-Master oder löschen Sie nicht benötigte DP-Slaves.

2.7.6.4 GSD- und GSDML-Geräte

Für Geräte, die mit PROFIBUS GSD- oder PROFINET GSD-Dateien konfiguriert wurden, müssen die entsprechenden GSD-Dateien ebenfalls im TIA Portal installiert sein. Wenn GSD-Dateien im TIA Portal noch nicht installiert sind, dann werden die GSD-Dateien während der Migration automatisch im TIA Portal installiert. Die GSD-Dateien müssen den Spezifikationen entsprechen (z. B. GSDML-Specification for PROFINET IO, siehe www.profinet.com (http://www.profinet.com)), um das Projekt erfolgreich zu migrieren. Wenn GSD-Dateien eines Geräteherstellers nicht der Spezifikation entsprechen, kann das Projekt nicht migriert werden. Im Migrationsprotokoll erhalten Sie Meldungen, die auf nicht normgerechte GSD-Dateien hinweisen.

Wenn die GSD-Dateien nicht den Spezifikationen entsprechen, prüfen Sie, ob der Hersteller eine neuere Version der GSD-Datei zur Verfügung stellt. Installieren Sie die aktuelle Version. Wenn keine den Spezifikationen entsprechende Fassung der GSD-Datei vorliegt, wenden Sie sich an den Gerätehersteller.

2.7.6.5 Verbindungen

Grundsätzlich lassen sich Verbindungen aus SIMATIC STEP 7-Projekten migrieren, sofern diese auch von der installierten Version des TIA Portals unterstützt werden.

Migrierbare Verbindungstypen

Die folgenden Verbindungstypen lassen sich migrieren:

- S7-Verbindung einseitig
- S7-Verbindung beidseitig
- S7-Verbindung fehlersicher
- TCP-Verbindung
- ISO-on-TCP-Verbindung
- ISO-Verbindung

- UDP-Verbindung
- eMail-Verbindung
- FDL-Verbindung
- Point-to-Point-Verbindung

FMS-Verbindungen am PROFIBUS werden bei der Migration nicht übernommen. Passen Sie die Verbindungen an oder ändern Sie das Anwenderprogramm.

2.7.6.6 Multiprojekte

Multiprojekte in SIMATIC STEP 7

In SIMATIC STEP 7 können Sie Projekte zu einem sogenannten Multiprojekt aufteilen, beispielsweise wenn ein Projekt besonders groß ist, oder wenn mehrere Personen daran arbeiten. Die Teilprojekte können projektübergreifende Referenzen enthalten, wie z. B. Verbindungen. Alle Teilprojekte eines Multiprojekts und die dazugehörigen Bibliotheken liegen im selben Verzeichnis.

Migration von Teilen eines Multiprojekts

Sie können die Teilprojekte eines Multiprojekts migrieren. Wählen Sie dazu bei der Migration eines der Teilprojekte aus. Alle im jeweiligen Teilprojekt enthaltenen Geräte und Konfigurationen werden bei der Migration übernommen.

Beachten Sie bei der Migration von Teilprojekten Folgendes:

- Projektübergreifende Verbindungen
  Projektübergreifende Verbindungen zwischen Teilprojekten werden als unspezifizierte Verbindung angelegt.
- Projektübergreifende Vernetzung
  Informationen zur Vernetzung von Geräten über Projektgrenzen hinweg gehen bei der Migration verloren. Vernetzungen zwischen Geräten im selben Teilprojekt bleiben hingegen erhalten.

Zusammenführen eines Multiprojekts zur Migration

Die Migration eines gesamten Multiprojekts mit allen zugehörigen Projektteilen ist nur dann möglich, wenn Sie vor der Migration die verschiedenen Projektteile wieder manuell zusammenführen. Um ein Multiprojekt zusammenzuführen, kopieren Sie beispielsweise alle Geräte und fügen Sie die Geräte in einem einzigen Projekt wieder ein.

2.7.6.7 Migration von Programmbausteinen

Sie können grundsätzlich alle Bausteine migrieren, die in KOP, FUP, SCL, AWL und GRAPH erstellt wurden. Beachten Sie jedoch folgende Hinweise:

Programme mit absoluter Adressierung

Im TIA Portal werden automatisch für alle absoluten Adressen symbolische Operanden deklariert. Wenn in dem zu migrierenden STEP 7-Programm mit absoluter Adressierung gearbeitet wurde, werden während der Migration symbolische Operanden für die absoluten Adressen deklariert. Diese automatische Deklaration kann zu Datentypkonflikten führen, insbesondere wenn die IEC-Prüfung im TIA Portal aktiviert ist. Ein Datentypkonflikt könnte z. B. entstehen, wenn ein 32 Bit-Datenwert durch die automatische Deklaration als Variable vom Typ DWORD interpretiert wird, im Programm aber eine Variable vom Datentyp REAL erwartet wird.

In diesem Fall genügt es nicht, die IEC-Prüfung auszuschalten. Sie müssen stattdessen die Deklaration in der PLC-Variablentabelle korrigieren.

Datentypkonflikte können ebenfalls entstehen, wenn Sie im Programm mehrfach auf dieselbe absolute Adresse zugreifen, für die verschiedenen Zugriffe aber unterschiedliche Datentypen verwenden. Das folgende Bild zeigt ein Beispiel für solche Mehrfachzugriffe:

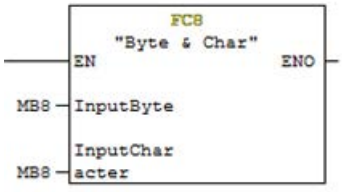

Während der Migration wird in diesen Fällen automatisch für jeden Datentyp eine eigene Variable deklariert. Da alle Variablen auf dieselbe absolute Adresse verweisen, werden in der PLC-Variablentabelle Warnungen ausgegeben. Diese Adressüberlappungen müssen Sie manuell in der PLC-Variablentabelle korrigieren.

Migration von Kommunikationsbausteinen

Bei der Migration von Kommunikationsanweisungen aus STEP 7 V5.x werden Zeiger vom Datentyp ANY durch PLC-Variablen innerhalb der eigenen S7-Station ersetzt.

- Nach der Migration wird eine Warnung an den betroffenen Parametern angezeigt.
- Ersetzen Sie an den betroffenen Parametern die PLC-Variablen durch einen Zeiger.
  Bei der absoluten Adressierung eines Datenbausteins mit dem Datentyp ANY muss der Datenbaustein mit Standardzugriff erstellt worden sein, da die symbolische Adressierung mit dem Datentyp ANY nicht möglich ist.
  Beispiel: Verwenden Sie zur Werteingabe an den entsprechenden Parametern der Anweisung P#M10.0 BYTE 4 anstelle von %MD10.

Symbolische Namen für Bausteine und anwenderdefinierte Datentypen (UDT)

Im TIA Portal hat jeder Baustein oder UDT eine Nummer und einen Namen. Es ist deshalb nicht mehr notwendig, symbolische Namen zu deklarieren. Wenn das migrierte Programm Symboldeklarationen für Bausteine oder UDTs enthält, werden diese nach der Migration als Namen verwendet.

Bausteine aus STEP 7-Bibliotheken

Im TIA Portal werden Standardfunktionen und -funktionsbausteine nicht in Bibliotheken bereitgestellt, sondern stehen als Anweisungen in der Task Card "Anweisungen" zur Verfügung. Die Anweisungen sind nach Funktion sortiert und unter ihrem symbolischen Namen auffindbar.

Wenn in dem migrierten Programm Aufrufe von Standardfunktionen oder -funktionsbausteinen enthalten sind, werden diese während der Migration durch die Anweisungen ersetzt, die der ursprünglichen Standardfunktion oder dem ursprünglichen Standardfunktionsbaustein entsprechen.

Falls der Bibliotheksbaustein im TIA Portal nicht mehr unterstützt wird, gibt es folgende Alternativen:

- Der Bibliotheksbaustein wird durch eine kompatible Anweisung ersetzt. In diesem Fall erhalten Sie eine Meldung, die Sie darauf hinweist, dass Sie das Programm nach der Migration übersetzen müssen. Beachten Sie, dass der aufrufende Baustein keinen Know-how-Schutz tragen darf, da der Bausteinaufruf während des Übersetzens angepasst werden muss.
- Wenn es keine kompatible Anweisung gibt, wird der Baustein als Anwenderbaustein migriert. Er liegt dann als know-how-geschützter Baustein im Ordner "Programmbausteine" und hat den Namenszusatz "_LF" (legacy function).

Anwenderdefinierte Bausteinbibliotheken

Anwenderdefinierte Bausteinbibliotheken gibt es im TIA Portal nicht mehr in der bekannten Form. Sie können anwenderdefinierte Bibliotheken jedoch migrieren, indem Sie sie vor der Migration in ein Projekt einfügen und dieses migrieren. Danach können Sie sie im TIA Portal aus diesem Projekt kopieren und verwenden.

Anweisungen in geänderter Darstellung

Die Darstellung einiger KOP-, FUP-, AWL-, SCL- und GRAPH-Befehle im TIA Portal unterscheidet sich von der Darstellung in früheren STEP 7-Versionen. So sind z. B. die mathematischen Funktionen und die Vergleicher nicht mehr datentypspezifisch. Es gibt stattdessen eine zentrale Anweisung, die für alle Datentypen anwendbar ist. Der Befehl "ADD_I" ist z. B. nicht mehr zulässig, stattdessen wird die Anweisung "ADD" verwendet.

Auch einige weitere Befehle haben im TIA Portal eine neue Darstellung, z. B. Flankenbefehle, Wortverknüpfungen, Konvertierungen, IEC-Zeiten, IEC-Zähler etc.

Während der Migration werden die Befehle in die neue Darstellung umgewandelt.

Strengere IEC-Prüfung

Im TIA Portal werden strengere Regeln zur Prüfung der Datentypkompatibilität angewandt. Außerdem werden Fehler, die zu Laufzeitfehlern führen können, bereits bei der Syntaxprüfung erkannt. So ist es in Funktionen (FC) nicht mehr zulässig, auf eigene Eingangsparameter zu schreiben oder eigene Ausgangsparameter zu lesen. Um Syntaxfehler zu vermeiden, beachten Sie die folgenden Regeln:

- Verwenden Sie bei der Parameterübergabe nur Variablen mit kompatiblen Datentypen.
- Verwenden Sie in Vergleichsanweisungen oder arithmetischen Anweisungen nur Variablen mit kompatiblen Datentypen.
- Auf Eingangsparameter darf nicht geschrieben und von Ausgangsparametern nicht gelesen werden.

Weitere Informationen zur Bausteinschnittstelle und zur Datentypkonvertierung mit IEC-Prüfung finden Sie in der Hilfe zu Datentypen:

Datentypen

Strengere Datentypprüfung bei den Anweisungen "MOVE" und "FILL"

Bei den Anweisungen "MOVE" und "FILL" wird die Kompatibilität der Quell- und Zielbereiche nach strengeren Regeln geprüft. Es ist z. B. nicht mehr möglich, mit "MOVE" eine Variable vom Datentyp "Integer" in einen Zielbereich vom Datentyp "WORD" zu verschieben. Nutzen Sie im TIA Portal stattdessen die Anweisungen "CONVERT", "T_CONV" oder "S_CONV".

Genauere Angaben zu den kompatiblen Datentypen für "MOVE" und "FILL" finden Sie in der Hilfe zu den Anweisungen:

AUTOHOTSPOT

AUTOHOTSPOT

Programme, die in KOP oder in FUP erstellt wurden, und unzulässige Datentypen an den Anweisungen "MOVE" oder "FILL" verwenden, werden nach der Migration in AWL dargestellt. In den Bausteineigenschaften ist die Sprache aber weiterhin als KOP oder FUP eingestellt.

Stellen Sie in den Bausteineigenschaften das Programm noch einmal nach AWL um. Anschließend stellen Sie die Sprache wieder zurück nach KOP oder FUP.

Peripheriezugriffskennung ":P"

Im TIA Portal wird für die direkte Adressierung der Peripherie die Zugriffskennung ":P" verwendet. Die folgende Notation ist z. B. nicht zulässig:

```
%PEW3
```

Stattdessen wird die folgende Notation verwendet:

```
%EW3:P //absolute Darstellung
MyTag:P //symbolische Darstellung
```

Während der Migration werden die Zugriffe in die neue Notation umgewandelt. Allerdings können symbolische Namen, die im Ursprungsprogramm für Peripherievariablen deklariert wurden, nicht übernommen werden. Stattdessen werden neue Deklarationen angelegt. Weitere Informationen zum Peripheriezugriff finden Sie in der Hilfe zur Adressierung der Peripherie:

Peripherie adressieren

Groß-/Kleinschreibung wird bei Sprungmarken nicht unterschieden

Bei Sprungmarken wird im TIA Portal nicht zwischen Groß- und Kleinschreibung unterschieden. Sind im Ursprungsprogramm Sprungmarken enthalten, die sich nur durch Groß-/Kleinschreibung unterscheiden, werden sie in eindeutige Sprungmarken umgewandelt. Die Protokolldatei gibt Aufschluss über die geänderten Sprungmarken.

Deutsches Sonderzeichen "ß" wird nicht von "ss" unterschieden

Bei Variablennamen wird im TIA Portal das deutsche Sonderzeichen "ß" nicht von "ss" unterschieden. Sind im Ursprungsprogramm z. B. Variablennamen enthalten, die sich nur durch diese beiden Zeichen unterscheiden, werden sie in eindeutige Variablennamen umgewandelt.

Migration von REAL- und STRING-Werten

Im TIA Portal wird geprüft, ob die Startwerte von REAL- und STRING-Variablen sich im zulässigen bzw. deklarierten Bereich befinden. Es gelten folgende Regeln:

- REAL-Variablen müssen einen Wert im Bereich von +1,175494E-38 bis +3,402823E+38 haben.
- Die maximale Länge des STRING-Werts darf die deklarierte Länge des STRINGS nicht überschreiten.

Wenn diese Regeln nicht erfüllt sind, wird das Programm zwar erfolgreich migriert, lässt sich jedoch anschließend nicht übersetzen.

Als Abhilfe können Sie in STEP 7 (TIA Portal) die entsprechenden Datenbausteine öffnen und die Startwerte ändern. Anschließend übersetzen Sie das Programm mit dem Menübefehl "Software (Bausteine komplett übersetzen)".

Migration von globalen Datenbausteinen

Die Darstellung und Hantierung von Datenwerten in Datenbausteinen ist im TIA Portal anders als in SIMATIC STEP 7. In globalen Datenbausteinen, die nicht von einem übergeordneten Objekt, z. B. einem UDT, abgeleitet sind, erhalten die Variablen als Defaultwert immer den Defaultwert des Datentyps, z. B. FALSE für den Datentyp BOOL. Dieser Defaultwert ist nicht editierbar. Falls im Ausgangsprojekt Offline-Anfangswerte in der Deklarationssicht vergeben wurden, werden diese bei der Migration nicht übernommen. Wenn Sie anwenderspezifische Defaultwerte im Programm benötigen, nutzen Sie einen globalen Datenbaustein, der auf einem PLC-Datentyp basiert. In einem PLC-Datentyp können Sie anwenderspezifische Defaultwerte für Variablenwerte vergeben.

Weitere Informationen zu Datenwerten in Datenbausteinen finden Sie in der Hilfe zu Datenbausteinen:

Datenbausteine programmieren

Variablentabellen

Die Variablentabellen aus dem Ursprungsprogramm werden nicht in das TIA Portal migriert. Um im TIA Portal Variablen online zu beobachten, erzeugen Sie sogenannte "Beobachtungstabellen". Diese haben dieselbe Funktion wie die Variablentabellen in SIMATIC STEP7.

Hinweis

Weitere Unterstützung zur Migration

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur Migration:

Migration Steuerungen (http://support.automation.siemens.com/WW/view/de/83557459)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

„Ret_val“ als Variablenname in der Bausteinschnittstelle

Wenn im Ausgangsprogramm in einer Bausteinschnittstelle das Schlüsselwort „Ret_val“, als Variablenname verwendet wird, wird der Baustein nicht korrekt migriert. Korrigieren Sie in diesem Fall das Ausgangsprogramm und migrieren Sie es erneut.

2.7.6.8 Migration von KOP-/FUP-/AWL-Programmen

KOP-/FUP- und AWL-Bausteine können nach der Migration ohne erneutes Übersetzen oder Laden ablaufen.

AWL-Quellen

AWL-Quellen können nicht in das TIA Portal migriert werden. Erzeugen Sie vor der Migration AWL-Bausteine aus Ihren Quellen. Diese können Sie anschließend im TIA Portal wieder als Quelle exportieren. Nutzen Sie dazu in der Projektnavigation den Kontextmenübefehl "Quelle aus Bausteinen generieren".

Siehe auch: AUTOHOTSPOT

Know-how-Schutz

Bausteine, die vor der Migration know-how-geschützt waren, bleiben auch nach der Migration know-how-geschützt. Da die Quellen nicht migriert werden, kann der Know-how-Schutz nicht aufgehoben werden. Die Bausteine können somit nicht geöffnet oder bearbeitet werden. Es ist jedoch möglich, die Bausteine in die CPU zu laden und zum Ablauf zu bringen.

Um den Know-how-Schutz eines Bausteins aufzuheben, haben Sie zwei Möglichkeiten:

- Entfernen Sie vor der Migration das Attribut KNOW_HOW_PROTECT aus der Quelle, erzeugen Sie daraus einen Baustein ohne Know-how-Schutz. Migrieren Sie dann diesen Baustein.
- Importieren Sie die Quelle in das TIA Portal und erzeugen daraus einen Baustein ohne Know-how-Schutz. Versehen Sie anschließend den Baustein über den Menübefehl "Bearbeiten > Know-how-Schutz" mit einem Passwort.

Umschaltung von KOP/FUP nach AWL

Im TIA Portal werden an einigen Stellen genauere Syntaxprüfungen durchgeführt als in älteren Versionen von STEP 7. Die Migration sorgt jedoch wenn möglich dafür, dass Ihr Programm übersetzbar bleibt: Wenn ein KOP-/FUP-Baustein Operationen enthält, die im TIA Portal zu einem Syntaxfehler führen würden, werden die betroffenen Netzwerke nach der Migration in AWL dargestellt. Die Semantik des Ausgangsprogramms bleibt jedoch unverändert.

Eine Meldung im Migrationsprotokoll weist Sie auf die Umstellung der Programmiersprache einzelner Netzwerke hin.

2.7.6.9 Migration von GRAPH-Programmen

Voraussetzung

Voraussetzung für die Migration von GRAPH-Bausteinen ist ein installiertes Optionspaket "S7-GRAPH, Version 5.3 SP6" oder höher auf dem Ursprungsgerät.

GRAPH-spezifische Bausteineinstellungen

Die GRAPH-spezifischen Bausteineinstellungen wurden im TIA Portal deutlich reduziert. Die folgende Tabelle gibt einen Überblick darüber, wie die ursprünglichen Bausteineinstellungen in das TIA Portal übertragen werden.

Einstellung im Ausgangsprojekt		Einstellung im TIA Portal
FB-Parameter	Minimal	Die vorhandenen Parameter werden migriert
	Standard	
	Maximal	
	Definierbar	
Schnittstellenbeschreibung	Speicherplatzminimiert	Speicherplatzminimierten DB erzeugen
	Strukturfelder	Es werden immer individuelle Strukturen angelegt
	Individuelle Strukturen	
Ablauffähigkeit	FC70/71	FC 73, wenn die Option "Speicherplatzminimierten DB erzeugen" aktiv ist, ansonsten FC 72
	FC72	
	FC73	
Kriterienanalyse im DB		Einstellung im TIA Portal nicht relevant
Schritte überspringen		Schritte überspringen
Quittierpflicht bei Fehler		Quittierpflicht bei Supervisionsfehlern
Synchronisation		Im TIA Portal immer aktiviert
Permanente Bearbeitung aller Interlocks im Handbetrieb		Permanente Bearbeitung aller Interlocks im Handbetrieb
Sperre der Betriebsartenwahl		Sperre der Betriebsartenwahl
Gesichertes Schaltverhalten		Im TIA Portal immer aktiviert
Warnungen	Keine	Im TIA Portal immer aktiviert
	Alle	

Einstellung im Ausgangsprojekt	Einstellung im TIA Portal
Meldungen aktivieren	Meldungen aktivieren
Meldungen mit WR_USMSG	Im TIA Portal Meldungen immer mit ALARM_SQ/ALARM_S

Schrittnamenserweiterung

Schrittnamenserweiterungen werden im TIA Portal nicht unterstützt. Sie werden während der Migration in schrittspezifischen Text umgewandelt.

Übersetzen des migrierten Bausteins

GRAPH-Bausteine müssen nach der Migration erneut übersetzt und geladen werden.

Beim Übersetzen kann es unter Umständen zu folgenden Fehlern kommen:

- Aufgrund der strengeren IEC-Prüfung kann es zu Datentypkonflikten kommen. Ändern Sie in diesem Fall die Deklaration.
- Da die Einstellung "Strukturfelder" für die Schnittstelle im TIA Portal nicht mehr verfügbar ist, können Schnittstellenelemente nicht mehr z. B. mit G7S[1].X adressiert werden. Der Zugriff auf das Element muss daher korrigiert werden, z. B. nach STEP1.X.

Änderung der Schnittstelle des GRAPH-Bausteins

Die geänderten Einstellungen können zu Schnittstellenänderungen des Bausteins führen. Möglicherweise müssen Sie deshalb den Instanz-DB neu erzeugen oder Bausteinaufrufe aktualisieren.

2.7.6.10 Migration von SCL-Programmen

Voraussetzung für die Migration von SCL-Bausteinen ist ein installiertes Optionspaket "S7-SCL, Version 5.3 SP5" oder höher auf dem Ursprungsgerät.

SCL-Bausteine müssen nach der Migration erneut übersetzt und geladen werden.

Grundsätzliche Vorgehensweise bei der Migration

Eine vollständige Migration von SCL-Bausteinen wird nur durchgeführt, wenn die zugehörigen Quellen im Ausgangsprojekt vorhanden sind.

Die folgende Tabelle gibt einen Überblick über die grundsätzliche Vorgehensweise:

Im Ausgangsprojekt vorhanden	Nach der Migration vorhanden
SCL-Bausteine mit Quelle	Editierbare SCL-Bausteine
Know-how-geschützte SCL-Bausteine mit Quelle	Editierbare SCL-Bausteine. Nach der Migration haben die Bausteine keinen Know-how-Schutz mehr, können aber bei Bedarf wieder geschützt werden.

Im Ausgangsprojekt vorhanden	Nach der Migration vorhanden
SCL-Bausteine ohne Quelle	Know-how-geschützte SCL-Bausteine
Know-how-geschützte SCL-Bausteine ohne Quelle	Know-how-geschützte SCL-Bausteine

SCL-spezifische Compiler-Optionen

Nur die Compiler-Optionen, die direkt in einer SCL-Quelle definiert sind, werden bei der Migration ins TIA-Portal als Bausteineigenschaft übernommen. Wenn in der ursprünglichen SCL-Quelle keine Compiler-Optionen definiert sind, werden die Optionen in den Eigenschaften des migrierten Bausteins deaktiviert.

Compiler-Optionen, die als Einstellung im SCL-Editor oder in einer SCL-Übersetzungssteuerdatei definiert sind, werden nicht migriert.

Die folgende Tabelle gibt einen Überblick über die Compiler-Optionen, die als Bausteineigenschaft in das TIA-Portal übertragen werden.

Compiler-Option in der SCL-Quelle	Bausteineigenschaft im TIA Portal
scl_monitorarraylimits	ARRAY-Grenzen prüfen
scl_createdebuginfo	Erweiterte Statusinformation erstellen
scl_setokflag	ENO automatisch setzen

Aufruf von Funktionsbausteinen

Ein Bausteinaufruf mit Angabe des Funktionsbausteins, z.B. "DBX"."FBX()" ist im TIA Portal nicht mehr zulässig. Die Aufrufsyntax wird während der Migration in die Notation "DBX()" umgewandelt.

Neuer EN-/ENO-Mechanismus

SCL nutzt den EN-/ENO-Mechanismus des TIA Portals. Aus diesem Grund werden bei der Migration alle Verwendungen des OK-Flags durch ENO ersetzt. Die Stellen, an denen im Ursprungsprogramm ENO verwendet wurde, werden markiert. Diese Stellen müssen Sie nach der Migration überprüfen und an den neuen Mechanismus anpassen.

Siehe auch:

AUTOHOTSPOT

Operator "DIV"

Der Operator "DIV" ist im TIA Portal nicht mehr verfügbar. Alle Verwendungen von "DIV" werden während der Migration in die normkonforme Notation "/" umgewandelt.

Lokale Konstanten

Im TIA Portal werden lokale, symbolische Konstanten im Abschnitt "CONST" der Bausteinschnittstelle deklariert. Sie haben immer einen Datentyp. Sind im Ursprungsprogramm symbolische Konstanten ohne Datentyp deklariert, wird ihnen während der Migration ein passender Datentyp zugewiesen.

In Classic-SCL-Programmen war es möglich, lokale Konstanten als ARRAY-Grenzen zu verwenden. Bei der Migration bleiben diese Deklarationen erhalten.

Beispiel für die Deklaration von ARRAY-Grenzen mit lokalen Konstanten in einem Classic-SCL-Programm:

```
VAR_INPUT
MyArray : Array[MyConst1 .. MyConst2] of Int;
END_VAR
```

Diese Deklaration sieht nach der Migration wie folgt aus:

1	Input	
2	MyArray	Array[#MyConst1..#MyConst2] of Int

In der Bausteinschnittstelle des TIA Portals ist es allerdings nicht möglich, Ausdrücke mit lokalen Konstanten zu bilden und diese zur Deklaration von ARRAY-Grenzen oder zur Initialisierung von Werten zu verwenden. Bei der Migration wird stattdessen das Ergebnis des Ausdrucks gebildet und als Wert in die Deklaration eingetragen. Der Ausdruck aus dem Ursprungsprogramm wird als Kommentar hinterlegt.

Beispiel für die Deklaration von ARRAY-Grenzen mit Hilfe von Ausdrücken in einem Classic-SCL-Programm:

```
CONST
MyConst1Value10 := int#10;
MyConst2Value10 := int#10;
MyConst20 := MyConst1Value10 + MyConst2Value10 ;
MyConst0 := MyConst1Value10 - MyConst2Value10 ;
MyConst100 := MyConst1Value10 * MyConst2Value10 ;
END_CONST
```

Diese Deklaration sieht nach der Migration wie folgt aus:

Constant			
MyConst1Value10	Int	10	
MyConst2Value10	Int	10	
MyConst20	Int	20	MigChange: MyConst1Value10 + MyConst2Value10
MyConst0	Int	0	MigChange: MyConst1Value10 - MyConst2Value10
MyConst100	Int	100	MigChange: MyConst1Value10 * MyConst2Value10

Siehe auch:

AUTOHOTSPOT

AUTOHOTSPOT

Geschachtelte ARRAYs

Im TIA Portal ist es nicht möglich, ARRAYs zu schachteln. Die folgende Deklaration wäre z. B. nicht zulässig:

```
ARRAY[1..5] OF ARRAY[0..3] OF INT
```

Geschachtelte ARRAYs werden während der Migration in mehrdimensionale ARRAYs umgewandelt. Das Beispiel würde also nach der Migration wie folgt aussehen:

```
ARRAY[1..5, 0..3] OF INT
```

Deklaration von Sprungmarken (LABEL)

Die Deklaration von Sprungmarken ist im TIA Portal nicht möglich. Die folgende Deklaration würde z. B. nicht aus dem Ausgangsprojekt übernommen:

```
LABEL
MARKE1, MARKE2, MARKE3 ;
END_LABEL
```

Gesetzte Sprungmarken bleiben jedoch im Programmcode enthalten und können für GOTO-Anweisungen genutzt werden.

Indizierte Peripheriezugriffe

Im TIA Portal wird für indizierte Peripheriezugriffe die Syntax mit runden Klammern verwendet. Die folgende Notation ist z. B. nicht zulässig:

```
PEB[1]
```

Stattdessen wird die folgende Notation verwendet:

```
EB(1):P
```

Während der Migration werden die Zugriffe in die neue Notation umgewandelt.

Siehe auch:

AUTOHOTSPOT

Indizierte Speicherzugriffe

Im TIA Portal wird für indizierte Speicherzugriffe die Syntax mit runden Klammern verwendet. Die folgende Notation ist z. B. nicht zulässig:

```
EB[2]
```

Stattdessen wird die folgende Notation verwendet:

```
EB(2)
```

Während der Migration werden die Zugriffe in die neue Notation umgewandelt. Weitere Informationen zur Adressierung finden Sie unter "Siehe auch".

Indizierter Zugriff auf Datenbausteine

Im TIA Portal wird für indizierte Datenbausteinzugriffe die Syntax mit runden Klammern verwendet. Die folgende Notation ist z. B. nicht zulässig:

```
%DB100.DW[5]
```

Stattdessen wird die folgende Notation verwendet:
`%DB100.DW(5)`

Während der Migration werden die Zugriffe in die neue Notation umgewandelt. Weitere Informationen zur Adressierung finden Sie unter "Siehe auch".

Absoluter Zugriff auf Datenbausteine

Bei einem absoluten Zugriff muss der absolute Bezeichner des Datenbausteins verwendet werden, der Zugriff mit dem symbolischen Bezeichner ist im TIA Portal nicht zulässig.

Die folgende Notation ist z. B. nicht zulässig:
`DB100.DW3`

Stattdessen wird die folgende Notation verwendet:
`%DB100.DW3`

Bei der Migration wird bei einem erkannten absoluten Datenbausteinzugriff der Kennzeichner "%" hinzugefügt. Weitere Informationen zur Adressierung finden Sie unter "Siehe auch".

Logarithmische Funktionen "EXPD" und "LOG"

Die logarithmischen Funktionen "EXPD" und "LOG" sind im TIA Portal nicht mehr verfügbar. Alle Verwendungen von "EXPD" werden während der Migration in die normkonforme Notation "10**(<Ausdruck>)" umgewandelt. Alle Verwendungen von "LOG" werden während der Migration in die normkonforme Notation "LN(<Ausdruck>)" umgewandelt.

Mathematische Anweisung "MAX"

Im TIA Portal kann die Anweisung "MAX" keine Zeitdatentypen verarbeiten. Sie können jedoch stattdessen Vergleicher verwenden.

Das folgende Beispiel zeigt, wie Sie mit Hilfe von Vergleichern den "größeren" von zwei Zeitdatentypen ermitteln können:

SCL

```
#xs_tod := #xi_tod + #xi_time;
   // statt #xstod1 := MAX(IN1:=#xtod1, IN2:=#xstod1);
   IF #xs_tod < #xi_tod THEN
         #xs_tod := #xi_tod;
   END_IF;

   // statt #xsd1 := MAX(IN1:=#xd1, IN2:=#xsd1);
   IF #xi_date <#xs_date THEN
         #xs_date := #xi_date;
   END_IF;
#xs_date_time := #xi_date_time + #xi_time;
   // statt #xs_date_time := MAX(IN1:=#xi_date_time, IN2:=#xs_date_time);
   IF #xi_date_time <#xs_date_time THEN
          #xs_date_time := #xi_date_time;
   END_IF;
```

SCL

Gleitpunktzahlen - Exponentialdarstellung

Die exponentielle Schreibweise von Gleitpunktzahlen ohne Dezimalpunkt ist im TIA Portal nicht mehr erlaubt. Die folgende Notation ist z. B. nicht zulässig:

```
3E10
```

Stattdessen wird die folgende Notation verwendet:

```
3.0E10
```

Während der Migration werden alle Verwendungen von "3E10" in die normkonforme Notation "3.0E10" umgewandelt.

Siehe auch:

AUTOHOTSPOT

Bausteine aus STEP 7-Bibliotheken

Die Funktionen FC1 - FC40 aus der STEP 7 Classic-Bibliothek "IEC-Standard-Funktionen" sind im TIA Portal nicht mehr verfügbar. Aufrufe der Standardfunktionen werden während der Migration in Aufrufe der entsprechenden Erweiterten Anweisungen umgewandelt. Wenn keine eindeutige Umwandlung möglich ist, wird ein Syntaxfehler ausgegeben.

Ändern Sie den SCL-Baustein in diesem Fall nach der Migration und verwenden Sie SCL-eigene Anweisungen oder Operatoren anstelle von IEC-Standard-Funktionen.

Beispiele:

Verwenden Sie S5TIME_TO_TIME anstelle von S5TI_TIM (FC33)

```
"TAG_TIME" := S5TIME_TO_TIME ("TAG_S5TIME");
```

Verwenden Sie den Vergleichsoperator **<>** anstelle von NE_DT (FC 9):

```
IF #D1 <> #D2 THEN "MyOutput" : = 10;
END_IF;
```

Verwenden Sie den Vergleichsoperator **=** anstelle von EQ_STRNG (FC 10):

```
IF #My_String1 = #MyString2 THEN "MyOutput" : = 10;
END_IF;
```

Unterbrechung von Strings mit "$"

In STEP 7 Classic konnten Strings in SCL durch Sonderzeichen, z. B. Steuerzeichen oder nichtdruckbare Zeichen unterbrochen werden. Diese Zeichen wurden durch die Unterbrechungssequenzen "$>" und "$<" gekennzeichnet.

Beispiel:

```
#myString := 'Test1$>     $<Test1';
```

Die Möglichkeit besteht im TIA Portal nicht mehr. Stringunterbrechungen werden bei der Migration gelöscht.

NULL-Pointer

Null-Pointer werden im TIA Portal nicht mehr mit "NIL", sondern mit "NULL" angegeben. Die Notation wird bei der Migration automatisch ersetzt.

Kommentare

Kommentare aus der migrierten Quelle werden wenn möglich übernommen. Dabei gelten folgende Regeln:

- Kommentare, die im Ausgangsbaustein direkt nach der Bausteindefinition stehen, werden als Bausteinkommentare umgesetzt.
- Kommentare, die als Zeilenkommentar nach einer Variablendeklaration stehen, werden in die Spalte "Kommentar" in der Bausteinschnittstelle übernommen.
- Alle anderen Kommentare werden nicht übernommen.

Beispiel:

```
// Kommentar, der nicht übernommen wird.
FUNCTION_BLOCK FB57
// Kommentar, der in die Bausteineigenschaften übernommen wird.
(*Kommentar, der in die Bausteineigenschaften übernommen wird. *)
VAR_TEMP
    myVar1 : INT;// Kommentar, der in die Schnittstelle übernommen
wird.
    myVar2 : INT;(* Kommentar, der nicht übernommen wird, da es sich
um einen Kommentarabschnitt handelt *)
END_VAR
// Kommentar, der nicht übernommen wird.
BEGIN
    ;
END_FUNCTION_BLOCK
```

Bausteinaufrufe in S7-SCL-Quellen

In den S7-SCL-Quellen bei STEP 7 V5.x konnten Sie die Bausteinparameter vor oder nach dem eigentlichen Aufruf des Bausteins setzen. Das heißt, der Baustein konnte mit unvollständigen Parametern aufgerufen und die In/Out-Parameter der Bausteine an einer anderen Stelle innerhalb der S7-SCL-Quelle mit Variablen oder Werten versorgt werden. Diese Art der Programmierung ist in STEP 7 (TIA Portal) systembedingt nicht möglich und wird auch in Zukunft nicht unterstützt. Sie können das STEP 7 V5.x Projekt zwar in TIA Portal fehlerfrei migrieren und übersetzen, doch ist die Funktion dieses Bausteinaufrufes innerhalb des S7-SCL-Programmes nicht mehr möglich.

Bevor Sie Ihre S7-SCL-Quelle in STEP 7 V5.x übersetzen und das Projekt in TIA Portal migrieren, beachten Sie die folgenden Punkte:

- Alle Parameter des aufgerufenen Bausteins müssen innerhalb der Aufrufanweisung versorgt werden.
- Es dürfen keine Parameter des aufgerufenen Bausteins gelöscht und an einer anderen Stelle der S7-SCL-Quelle eingefügt und parametriert werden.

2.7.6.11 Meldungen

PLC-Meldungen werden mit allen Parametern und inklusive aller Meldeklassen migriert. Meldungstypen und Meldungsinstanzen werden ebenfalls migriert und unter den PLC-Meldungen im TIA Portal abgelegt.

Voraussetzung

Voraussetzung ist, dass im Ausgangsprojekt die CPU-weite Vergabe der Meldenummern gewählt ist. Aktivieren Sie vor der Migration im Ausgangsprojekt daher gegebenenfalls die Option "Meldenummern immer CPU-weit eindeutig vergeben" und führen Sie ein Speichern unter > Mit Reorganisation aus.

Einschränkungen

- Symbolbezogene Meldungen können nicht migriert werden.
- Die Schlüsselwörter "$$AKZ$$", "$$BlockComment$$" und "$$Area$$" werden übernommen, jedoch findet keine Ersetzung zur Laufzeit statt.

2.7.6.12 Systemfehler melden

Systemfehlermeldungen, die Sie mit SIMATIC STEP 7 projektiert haben, können Sie prinzipiell migrieren.

Einschränkungen

Bei der Migration der Applikation "Systemfehler melden" werden nur die Einstellungen, die vom Anwender im SIMATIC STEP 7-Projekt vorgenommen wurden, migriert (z. B. Grundeinstellungen, Bausteinnummern, Bausteinnamen).

Nicht migriert werden die folgenden Elemente:

- SFM-Bausteine
- SFM-Meldungen und die Einstellungen, die zum Konfigurieren der Meldetexte vorgenommen wurden
- System-Textbibliotheken

Nach der Migration können die Einstellungen erneut manuell angepasst werden. Die fehlenden Elemente werden beim ersten Übersetzen der Hardware-Konfiguration neu erzeugt.

2.7.6.13 Migration von Projekttexten

Bei der Migration des Ausgangsprojekts werden Projekttexte, wie z. B. Bausteintitel und Bausteinkommentare, ins TIA Portal übernommen. Jedoch werden nur Projekttexte in der im Ausgangsprojekt eingestellten Editiersprache migriert. Projekttexte in anderen Sprachen werden nicht migriert.

Siehe auch

Migration von STEP 7-Projekten (Seite 20)

2.8 WinCC-Projekte migrieren

2.8.1 WinCC flexible 2008 Projekte migrieren

2.8.1.1 Grundlagen (WinCC flexible)

Migration (WinCC flexible)

Einleitung

In WinCC können Sie Projekte aus WinCC flexible weiter verwenden. Folgende Versionen von WinCC flexible werden unterstützt:

- WinCC flexible 2008 SP2
- WinCC flexible 2008 SP3
- WinCC flexible 2008 SP5

In den folgenden Kapiteln wird beschrieben, welche Bediengeräte unterstützt werden und welche Randbedingungen für eine erfolgreiche Migration gelten.

Projekte aus ProTool Pro und aus früheren WinCC flexible Versionen können Sie nicht direkt nach WinCC migrieren. Wenn Sie solche Projekte in WinCC weiter verwenden wollen, müssen Sie diese erst in eine unterstützte Version von WinCC flexible migrieren und den Bediengerätetyp ändern.

Wenn das zu migrierende Projekt Anteile eines unterstützten Optionspakets enthält, dann muss das Optionspaket für eine erfolgreiche Migration in WinCC installiert sein. Wenn das Optionspaket nicht installiert ist, wird die Migration abgebrochen. Folgende Optionspakete sind betroffen:

- SINUMERIK

Folgende Optionspakete werden von der Migration nicht unterstützt:

- ProAgent
- Open Platform Program - OPP

Siehe auch

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Projekte aus WinCC flexible migrieren (WinCC flexible) (Seite 47)

Übersetzen und Laden eines migrierten Projekts (WinCC flexible) (Seite 49)

Unterstützte Bediengeräte (WinCC flexible) (Seite 50)

Objektunterstützung bei der Migration (WinCC flexible) (Seite 55)

Migration von Verbindungen (WinCC flexible) (Seite 59)

Migration von Variablen (WinCC flexible) (Seite 62)

Migration von Runtime-Daten (WinCC flexible) (Seite 72)

Migration integrierter Projekte (WinCC flexible) (Seite 76)

Migration der Datentypen (WinCC flexible) (Seite 80)

Grundlagen zur Migration (WinCC flexible)

Einleitung

Bei der Migration werden die Projektdaten aus einem WinCC flexible Projekt in das neue Datenformat von WinCC umgesetzt. Dabei wird nicht geprüft, ob die Daten in dem zu migrierenden Projekt konsistent sind. Wenn in einem Quellprojekt beim Übersetzen Fehler oder Warnungen ausgegeben werden, dann werden diese bei der Migration nicht behoben. Daher sollte das zu migrierende Projekt vor der Migration fehlerfrei übersetzbar sein. Beachten Sie bei der Migration den Umfang eines Projekts. Für die Migration gelten die Leistungsmerkmale von WinCC. Nähere Hinweise hierzu finden Sie in der Online-Hilfe im Kapitel "Prozesse visualisieren > Leistungsmerkmale > Engineering System".

Eindeutigkeit von Objektnamen

In WinCC flexible wird die Eindeutigkeit der Objekte durch den enthaltenden Ordner bestimmt. Bei Bildobjekten in Gruppen wird die Eindeutigkeit durch den Gruppennamen bestimmt.

In WinCC muss ein Objektname innerhalb eines Bediengeräts eindeutig sein. Bei Bildobjekten muss der Name innerhalb eines Bilds eindeutig sein.

Bei der Migration wird die Eindeutigkeit der Namen geprüft. Wenn ein Name nach der neuen Regelung nicht eindeutig ist, wird das betreffende Objekt umbenannt. Ein umbenanntes Objekt erhält das Suffix "#Mign", wobei "n" eine fortlaufende Nummer ist.

Hinweis

Geänderter OPC-DA-Servername

Wenn Sie im WinCC flexible-Projekt einen OPC-DA-Server projektiert haben, ist der OPC-DA-Server nach der Migration nicht mehr unter dem alten Namen erreichbar.

Ändern Sie im migrierten Projekt bei den betroffenen OPC-Clients den Namen des OPC-DA-Servers auf folgenden Wert: "OPC.SimaticHMI.CoRtHmiRTm".

Besonderheiten bei der Umbenennung von Variablen

Wenn Sie in WinCC flexible Variablen in Ordnern strukturiert haben, wird durch die Migration der Name der Variablen gebildet aus dem Ordnernamen und dem Variablennamen. Die Namen der Ordner und der Variablen werden durch das Zeichen \ voneinander getrennt. Der Name der Variablen nach der Migration lautet dann z. B.: Plant1\Line3\Tag17.

Wenn der Name nach der Migration länger als 128 Zeichen werden würde, wird der Name gebildet aus der Zeichenfolge #mig, einer laufenden Nummer, dem Zeichen # und dem Variablennamen aus WinCC flexible, z. B. #mig2#Tag17.

Wenn Sie in Skripten Variablennamen dynamisch zusammensetzen, müssen Sie die Variablen überprüfen, deren Name bei der Migration mit "#mig" verändert wurde.

Betroffene Objekte

Folgende Objekte werden bei Bedarf umbenannt:

- Bilder
- Bildbausteine
- Bildobjekte
- Grafiken
- Rezepturen
- Strukturen
- Strukturelemente
- Meldearchive
- Variablen
- Variablenarchive
- Verbindungen

Abbruch der Migration

In folgenden Fällen wird die Migration abgebrochen:

- Wenn das zu migrierende Projekt im Engineering-System oder in Runtime geöffnet ist.
- Wenn nicht genügend freier Speicherplatz auf der Festplatte zur Verfügung steht, um eine Kopie für die Migration des Projekts anzulegen.
- Wenn die Migration wegen Problemen mit dem installierten SQL-Server die Projektdatenbank nicht ansprechen kann.
- Wenn die Migration wegen fehlender Benutzerberechtigung die Projektdatenbank nicht ansprechen kann.
- Wenn Sie in einem integrierten Projekt die "*.hmi" Datei für die Migration auswählen. In einem integrierten Projekt müssen Sie für die Migration die Datei "*.s7" auswählen.
- Wenn das Projekt mit einer von der Migration nicht unterstützten Version erstellt wurde.
- Wenn im Projekt nicht unterstütze Geräte projektiert sind.

Projekt im Migrationsformat speichern

Die Migration eines WinCC flexible-Projekts müssen Sie nicht vollständig auf dem PC durchführen, auf dem das Projekt vorhanden ist. Sie können die Migration vorbereiten, indem Sie das Projekt im Migrationsformat speichern. Für die Speicherung eines WinCC flexible-Projekts im Migrationsformat steht Ihnen das Migrationstool zur Verfügung. Das Migrationstool exportiert die Engineering-Daten aus dem WinCC flexible-Projekt und speichert die Daten im Migrationsformat "*.am*".

Für die eigentliche Migration kopieren Sie die Daten im Migrationsformat auf einen PC, auf dem das TIA Portal installiert ist.

Nähere Hinweise finden Sie in der Dokumentation zum Migrationstool.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Bediengerätewechsel durch die Migration (WinCC flexible) (Seite 52)

Projekte aus WinCC flexible migrieren (WinCC flexible)

Einleitung

Bei der Migration eines Projekts werden Daten aus einem WinCC flexible-Projekt in ein neues Projekt für WinCC geladen. Deshalb wird für die Projektmigration automatisch ein neues Projekt angelegt. Sie können nicht in ein bestehendes Projekt migrieren.

Die Migration lässt sich sowohl in der Portalansicht als auch in der Projektansicht aufrufen.

Migrieren Sie ein Projekt nur in einem neu gestarteten TIA Portal.

Hinweise zur Migration eines integrierten Projekts finden Sie im Kapitel Migration integrierter Projekte (WinCC flexible) (Seite 76).

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible) (Seite 45).

Hinweis

Weitere Unterstützung zur Migration

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur Migration:

Migration Visualisierung (http://support.automation.siemens.com/WW/llisapi.dll?query=Visualisierung&func=cslib.cssearch&content=adsearch%2Fadsearch.aspx&lang=de&siteid=csius&objaction=cssearch&searchinprim=0&nodeid0=4000024&redir=false)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich an den SIMATIC Customer Support.

Voraussetzung

- Ein Projekt aus WinCC flexible ist vorhanden.
- Das Projekt ist nicht in WinCC flexible geöffnet.

Vorgehen

Ein Projekt migrieren Sie in der Portalansicht wie folgt:

1. Wählen Sie die Aktion "Start > Projekt migrieren".

2. Navigieren Sie im Feld "Quellpfad" zu dem Projekt, das Sie migrieren wollen.
3. Selektieren Sie die WinCC flexible-Projektdatei "*.hmi" oder "*.am*".
4. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.
5. Klicken Sie auf "Migrieren".
   Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
   - Die Projektansicht wird geöffnet.
   - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
   - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
   - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
   - Nach Abschluss der Migration wird das Projekt gespeichert und eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte Bediengerät ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten, z. B. Bilder, Meldungen und Variablen.

Migrationsprotokoll zu einem späteren Zeitpunkt öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Sie können das Protokoll zu einem späteren Zeitpunkt ansehen. Die Protokolldatei öffnen Sie folgendermaßen:

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten > Protokolle". Er enthält die Protokolle aller bisher durchgeführten Migrationen.
2. Doppelklicken Sie auf das gewünschte Migrationsprotokoll.
   Das Protokoll wird geöffnet.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Migration integrierter Projekte (WinCC flexible) (Seite 76)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Übersetzen und Laden eines migrierten Projekts (WinCC flexible)

Übersetzen eines migrierten Projekts

Wenn Sie ein WinCC flexible-Projekt erfolgreich migriert haben, müssen Sie es vor dem Laden auf das Bediengerät neu übersetzen. Das Projekt wird nur dann erfolgreich übersetzt, wenn es vor der Migration fehlerfrei übersetzbar war.

Wenn beim Übersetzen des migrierten Projekts Fehler auftreten, müssen Sie diese beseitigen.

Nachdem das Projekt fehlerfrei übersetzt wurde, laden Sie es auf das Bediengerät.

Einstellungen zum Laden auf das Bediengerät

Bei der Migration werden die Einstellungen für das Laden auf das Bediengerät nicht übernommen. Wenn Sie das Projekt migriert haben, müssen Sie die Einstellungen für das Laden neu projektieren.

Selektieren Sie das Bediengerät in der Projektnavigation und wählen Sie den Kontextmenübefehl "Laden in Gerät > Software (komplett laden)". Der Dialog "Erweitertes Laden" wird geöffnet. Projektieren Sie die erforderlichen Einstellungen für die Schnittstelle. Klicken Sie auf die Schaltfläche "Laden". Das Projekt wird neu übersetzt und der Dialog "Vorschau laden" wird geöffnet.

Expandieren Sie den Eintrag "Überschreiben" und kontrollieren Sie die Einstellungen für die folgenden Optionen:

- Wollen Sie die bestehenden Benutzerverwaltungsdaten dieses Geräts überschreiben
- Wollen Sie die bestehenden Rezepturdaten auf dem Bediengerät überschreiben

Projektieren Sie die Optionen so, wie Sie sie im Projekt künftig verwenden wollen. Laden Sie anschließend das Projekt auf das Bediengerät.

Siehe auch

Migration (WinCC flexible) (Seite 44)

2.8.1.2 Engineering-Daten migrieren (WinCC flexible)

Bediengeräte (WinCC flexible)

Unterstützte Bediengeräte (WinCC flexible)

Einleitung

Bei der Migration von Projekten aus WinCC flexible müssen Sie beachten, dass WinCC nicht alle Bediengeräte unterstützt. Folgende Fälle müssen Sie unterscheiden:

- Bediengerät wird von WinCC unterstützt.
  Das Projekt wird 1:1 migriert und enthält nach der Migration das gleiche Bediengerät wie vor der Migration.
- Bediengerät wird durch ein kompatibles Nachfolgemodell ersetzt.
  Das Projekt wird migriert. Die Migration ersetzt das Bediengerät durch ein kompatibles Nachfolgemodell. Nähere Hinweise hierzu finden Sie unter Bediengerätewechsel durch die Migration (WinCC flexible) (Seite 52).
- Bediengerät wird nicht unterstützt.
  Wenn Ihr WinCC flexible Projekt ein Bediengerät enthält, das von WinCC nicht unterstützt wird, wird der Migrationsvorgang abgebrochen. Um das Projekt zu migrieren, müssen Sie in WinCC flexible das Bediengerät in einen von WinCC unterstützten Bediengerätetyp ändern.
  Wenn das nicht unterstützte Bediengerät in der Bibliothek enthalten ist, wird die Migration ausgeführt. Im migrierten Projekt im TIA Portal werden in der Projektbibliothek entsprechende Kopiervorlagen für die nicht unterstützten Geräte angelegt.

Die folgenden Bediengerätetypen werden sowohl von WinCC flexible als auch von WinCC unterstützt:

Basic Panels

- KTP400 Basic mono PN
- KTP400 Basic mono PN Portrait
- KTP600 Basic DP
- KTP600 Basic DP Portrait
- KTP600 Basic PN
- KTP600 Basic PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic mono PN Portrait
- KTP1000 Basic DP
- KTP1000 Basic PN
- TP1500 Basic PN

Mobile Panels

- Mobile Panel 177 6" DP
- Mobile Panel 177 6" PN
- Mobile Panel 277 8"
- Mobile Panel 277 8" IWLAN V2
- Mobile Panel 277F 8" IWLAN V2
- Mobile Panel 277F 8" IWLAN (RFID Tag)
- Mobile Panel 277 10"

SINUMERIK PC

- OP 010 Key
- OP 012 Key
- OP 015 Key
- OP 015A Key
- TP 015A Touch+Key

HMI Applikationen

- WinCC flexible Runtime

In WinCC werden nur die Funktionalitäten unterstützt, die diese Bediengerätetypen anbieten.

Weitere Funktionalitäten, die wegen der eingeschränkten Geräteauswahl nicht migriert werden, sind in den nachfolgenden Kapiteln dokumentiert.

Anpassungen vor der Migration

Wenn in dem zu migrierenden Projekt das Bediengerät in ein Bediengerät mit anderer Bildgröße geändert wurde, müssen Sie das Projekt vor der Migration in WinCC flexible neu kompilieren und speichern. Durch den Kompiliervorgang werden die Bilder und Bildelemente in ihrer Größe angepasst.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Bediengerätewechsel durch die Migration (WinCC flexible) (Seite 52)

Projektierungsänderung nach Bediengerätewechsel (WinCC flexible) (Seite 54)

Objektunterstützung bei der Migration (WinCC flexible) (Seite 55)

Migration von Verbindungen (WinCC flexible) (Seite 59)

Bediengerätewechsel durch die Migration (WinCC flexible)

Einleitung

WinCC flexible unterstützt einige Bediengeräte, die künftig nicht fortgeführt werden. Diese Bediengeräte werden von WinCC nicht mehr unterstützt. Bei der Migration eines WinCC flexible-Projekts wird ein nicht unterstütztes Bediengerät durch ein kompatibles Nachfolgegerät ersetzt.

Während der Migration wird lediglich der Bediengerätetyp geändert. Bediengerätspezifische Daten werden durch die Migration nicht geändert.

Durch einen Bediengerätewechsel können Inkonsistenzen im Projekt auftreten. Bei einem Projekt, das vor der Migration übersetzbar war, können nach dem Bediengerätewechsel Fehler bei der Übersetzung des Projekts entstehen. Z. B. weil das geänderte Bediengerät andere Speichermedien unterstützt als das vorige Bediengerät.

Bediengerätewechsel durch die Migration

Die folgende Tabelle zeigt, durch welche Nachfolger die nicht unterstützten Bediengeräte ersetzt werden.

Nicht unterstützte Bediengeräte	Nachfolger
HT 8	PCU50.3 -C (Auflösung 640*480)
Mobile Panel 170 6"	Mobile Panel 177 6" DP
Mobile Panel 277 8" IWLAN	Mobile Panel 277 8" IWLAN V2
Mobile Panel 277F 8" IWLAN	Mobile Panel 277F 8" IWLAN V2
OP010 Key	PCU50.3B -C (Auflösung 640*480)
OP012 Key	PCU50.3B -C (Auflösung 800*600)
OP015 Key	PCU50.3B -C (Auflösung 1024*768)
OP 08T	PCU50.3 -C (Auflösung 640*480)
TP015A Touch Key	PCU50.3B (Auflösung 1024*768)
PC 477 12" Key	PC 477B 12" Key PB
PC 477 12" Touch	PC 477B 12" Touch PB
PC 477 15" Key	PC 477B 15" Key PB
PC 477 15" Touch	PC 477B 15" Touch PB
PC 477 19" Touch	PC 477B 19" Touch PB
PC 577 12" Key	IPC 577C 12" Key PB
PC 577 12" Touch	IPC 577C 12" Touch PB
PC 577 15" Key	IPC 577C 15" Key PB
PC 577 15" Touch	IPC 577C 15" Touch PB
PC 577 19" Touch	IPC 577C 19" Touch PB
PC 670 10" Key	HMI IPC677C 12" Key PB
PC 670 12" Key	PC 677B 12" Key PB
PC 670 12" Touch	PC 677B 12" Touch PB
PC 670 15" Key	PC 677B 15" Key PB
PC 670 15" Touch	PC 677B 15" Touch PB

Nicht unterstützte Bediengeräte	Nachfolger
PC 677 12" Key	PC 677B 12" Key PB
PC 677 12" Touch	PC 677B 12" Touch PB
PC 677 15" Key	PC 677B 15" Key PB
PC 677 15" Touch	PC 677B 15" Touch PB
PC 677 17" Touch	PC 677B 17" Touch PB
PC 677 19" Touch	PC 677B 19" Touch PB
PC 870 12" Key	HMI IPC677C 12" Key PB
PC 870 15" Key	HMI IPC677C 15" Key PB
PC 870 15" Touch	HMI IPC677C 15" Touch PB
PC 877 12" Key	HMI IPC677C 12" Key PB
PC 877 15" Key	HMI IPC677C 15" Key PB
PC 877 15" Touch	HMI IPC677C 15" Touch PB
PC 877 19" Touch	HMI IPC677C 19" Touch PB
PC IL 70 12" Touch	HMI IPC577C 12" Touch PB
PC IL 70 15" Touch	HMI IPC577C 15" Touch PB
PC IL 77 12" Touch	HMI IPC577C 12" Touch PB
PC IL 77 15" Touch	HMI IPC577C 15" Touch PB
PC IL 77 19" Touch	HMI IPC577C 19" Touch PB
PC IL 77 12" Key	HMI IPC577C 12" Key PB
PC IL 77 15" Key	HMI IPC577C 15" Key PB
P012T Touch	IPC 677C 12" Touch PB (wenn eine P350 verbunden ist)
P012T Touch	IPC 477C 12" Touch PB (wenn eine P320 verbunden ist)

Folgende Geräte werden nicht unterstützt und somit auch nicht migriert:

- OP 73 und OP 73 micro
- TP 170 micro 6"
- TP 177 micro 6" und TP 177 micro 6" Portrait
- C7-635 6" Key und C7-635 6" Touch
- C7-636 6" Key
- C7-636 10" Touch
- OP 77A und OP 77B
- TP 170A 6"
- TP 177A 6" und TP 177A 6" Portrait
- TP 170B 6" mono und TP 170B 6" color
- TP 177B 6" mono DP und TP 177B 6" color PN/DP
- OP 170B 6" mono
- OP 177B 6" mono DP und OP 177B 6" color PN/DP
- TP 177B 4" color PN/DP

- TP 270 6"
- TP 270 10"
- OP 270 6"
- OP 277 6"
- OP 270 10"
- TP 277 6"
- MP 177 6" Touch
- MP 270 6" Touch
- MP 270 10" Key und MP 270 10" Touch
- MP 277 8" Key V1 und V2, MP 277 8" Touch V1 und V2
- MP 277 10" Key V1 und V2, MP 277 10" Touch V1 und V2
- MP 370 12" Key und MP 370 12" Touch
- MP 370 15" Touch
- MP 377 12" Key und MP 377 12" Touch
- MP 377 15" Touch und MP 377 15" Touch Daylight
- MP 377 19" Touch
- TP 012 Touch
- PC-R Key
- PC-R Touch und PC-R Key

Bediengerätewechsel durch den Anwender

Wenn ein Projekt ein Bediengerät enthält, das von WinCC im TIA Portal nicht unterstützt wird und kein kompatibles Nachfolgemodell vorhanden ist, wird die Migration abgebrochen. Wenn Sie das Projekt migrieren wollen, müssen Sie das Bediengerät vor der Migration in WinCC flexible selbst wechseln. Verwenden Sie dafür ein Bediengerät, das sowohl von WinCC flexible als auch von WinCC unterstützt wird.

Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 50)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Projektierungsänderung nach Bediengerätewechsel (WinCC flexible)

Hinweis

Wenn Sie ein Projekt mit eingebetteten Bildern migrieren, kann dadurch die Projektdatei größer als die alte Projektdatei werden.

Bediengerätewechsel durch die Migration

Wenn die Migration einen Bediengerätewechsel durchführt, ersetzt sie das bestehende Bediengerät durch ein kompatibles Nachfolgemodell. Die Nachfolgemodelle sind weiter entwickelt und damit leistungsfähiger als ihre Vorgängermodelle. Die neuen Modelle unterstützen damit alle Eigenschaften ihrer Vorgänger. Daher sind durch den Bediengerätewechsel nur wenige Nacharbeiten am Projekt zu erwarten.

Bediengerätewechsel durch den Anwender

Wenn Sie ein Bediengerät vor der Migration selbst wechseln, müssen Sie sicherstellen, dass das eingesetzte Bediengerät alle Eigenschaften unterstützt, die im Projekt verwendet werden. Übersetzen Sie das Projekt nach dem Bediengerätewechsel in WinCC flexible. Das Projekt muss vor der Migration fehlerfrei übersetzbar sein. Migrieren Sie anschließend das Projekt in WinCC.

Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 50)

Objektunterstützung bei der Migration (WinCC flexible)

Einleitung

Bei der Migration von Projekten aus WinCC flexible werden alle Projektierungsdaten migriert, die ein von WinCC unterstütztes Bediengerät enthält. Grundsätzlich werden alle Objekttypen und Funktionen vollständig migriert, die in der neuen Projektumgebung zur Verfügung stehen und abgebildet werden können.

Einige globale Objekttypen werden nicht migriert, z. B. Wörterbücher und globale Bibliotheken.

Unterstützte Objekttypen

Folgende Objekttypen werden von der Migration unterstützt:

- Animationen
  Deaktivierte Animationen werden nicht migriert.
- Audit-Einstellungen
- Audit-Protokolle
- Aufgabenplaner
- Benutzerverwaltung
- Bereichszeiger
- Bildbausteine
- Bilder
- Bildvorlage
- Datentypen

- Funktionslisten
- Grafiklisten
- Anzeige- und Bedienobjekte
  Von der Migration werden alle Anzeige- und Bedienobjekte unterstützt, die auf den unterstützten Bediengeräten verfügbar sind.
- Meldungen
- Meldeklassen
- Meldegruppen
- Projektbibliothek
- Projektsprachen
- Protokolle
- Rezepturen
- Runtime-Sprachen
- Runtime-Schriften
- Sm@rtAccess/Sm@rtService
- Strukturen
- Systemereignisse
- Systemfunktionen
- Texte
- Textlisten
- Variablen
- Verbindungen
- Wirkbereiche
- Zonen
- Zyklen

Nicht unterstützte Objekttypen

Folgende Funktionalitäten und Objekttypen werden von der Migration nicht unterstützt:

- Globale Bibliotheken
- Wörterbücher
- Projektversionen
- Änderungslogbuch

Abbildung der Bildnavigation

WinCC unterstützt nicht die Bildnavigation aus WinCC flexible. Die Daten der Bildhierarchie aus WinCC flexible werden nicht migriert. Um die Funktionalität einer Bildnavigation aus WinCC flexible abzubilden, werden die Navigationsschaltflächen auf die in WinCC verfügbare Schaltfläche migriert. Die auf die Systemfunktion "AktiviereBild" migrierten Systemfunktionen werden an das Ereignis "Klicken" der jeweiligen Schaltfläche migriert. Folgende in WinCC flexible verwendeten Systemfunktionen für die Bildnavigation sind in WinCC nicht verfügbar:

- AktiviereGrundbild
- AktiviereLinkesBild
- AktiviereRechtesBild
- AktiviereUebergeordnetesBild
- AktiviereErstesUntergeordnetesBild

Diese Systemfunktionen werden migriert auf die Systemfunktion "AktiviereBild". Der Parameter "Bildname" wird aus den Daten der Bildhierarchie entnommen. Wenn eine der genannten Systemfunktionen aus einem Bild aufgerufen wird, das nicht in der Bildhierarchie enthalten ist, dann wird die Systemfunktion "AktiviereBild" ohne Parameter erzeugt. An diese Systemfunktion müssen Sie nach der Migration das gewünschte Bild projektieren.

Tab-Reihenfolge in Bildern mit Bildbausteinen

In Bildern mit Bildbausteinen ändert sich durch die Migration die Tab-Reihenfolge sowohl im Bild als auch im Bildbaustein.

Migration der Bildvorlage

In WinCC steht Ihnen ein erweitertes Konzept zur Arbeit mit Bildvorlagen zur Verfügung. In WinCC stehen Ihnen ein globales Bild und mehrere Vorlagen für jedes Gerät zur Verfügung. Bei der Migration einer Vorlage aus WinCC flexible werden die enthaltenen Objekte und die in der Vorlage projektierten Eigenschaften in das erweiterte Konzept der Bildvorlagen von WinCC migriert.

Folgende Objekte werden in das "Globale Bild" von WinCC migriert:

- Meldefenster
- Meldeindikator
- Hilfeindikator
- Funktionstasten von Bediengeräten mit Funktionstasten

Alle weiteren Objekte und Eigenschaften werden in eine Vorlage von WinCC migriert.

Die Verbindung der Objekte und Eigenschaften zur jeweiligen Vorlage wird von der Migration automatisch angepasst.

Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 50)

Migration (WinCC flexible) (Seite 44)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC flexible) (Seite 58)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC flexible)

Einleitung

Aufgrund der Vereinheitlichung von Objekteigenschaften aus WinCC V7 und WinCC flexible sind im Verlauf der Migration an einigen Objekteigenschaften Änderungen erforderlich. Die Migration berechnet die Änderungen so, dass die Darstellung der Objekte nach der Migration gleich ist, wie vor der Migration. Durch die Änderungen während der Migration ergeben sich für einige Objekteigenschaften andere Maßeinheiten und Werte in der Projektierung.

Migration der Schrifteinstellungen eines Objekts

In WinCC V7 und in WinCC flexible wird für die Größe der verwendeten Schriften an einem Objekt die Maßeinheit "Punkt" verwendet. In WinCC wird für die Größe der verwendeten Schriften an einem Objekt die Maßeinheit "Pixel" verwendet. Bei der Migration wird die Schriftgröße entsprechend umgerechnet, so dass die Darstellung der Schrift bei der Zoomstufe 100% gleich groß ist. Aufgrund der unterschiedlichen Maßeinheiten ändern sich nach der Migration die Zahlenwerte für die Schriftgrößen.

Beispiel:

Schriftart vor der Migration	Schriftart nach der Migration
Arial 10 Punkt	Arial 13 Pixel
Arial 16 Punkt	Arial 21 Pixel
Tahoma 10 Punkt	Tahoma 13 Pixel
Tahoma 16 Punkt	Tahoma 21 Pixel

Migration von Objekträndern

In WinCC flexible gestatten einige Objekte bei der Projektierung der Darstellung die Eingabe von Werten <0 und >127 für die Einrichtung der Objektränder. In WinCC ist der Wertebereich für Objektränder begrenzt auf Werte zwischen 0 und 127. Die Migration ändert Werte <0 auf den Wert "0" und Werte >127 auf den Wert "127".

Siehe auch

Objektunterstützung bei der Migration (WinCC flexible) (Seite 55)

Verbindungen (WinCC flexible)

Migration von Verbindungen (WinCC flexible)

Einleitung

Wenn Sie ein Projekt migrieren, in dem ein unterstützter Kommunikationstreiber verwendet wird, dann wird dieser Treiber in WinCC weiter verwendet. Objekte, die über diesen Treiber kommunizieren, werden 1:1 migriert. Nacharbeiten sind nicht erforderlich.

In WinCC werden nicht alle Kommunikationstreiber unterstützt, die in WinCC flexible zur Verfügung stehen. Wenn in dem zu migrierenden Projekt ein nicht unterstützter Treiber verwendet wird, gibt es 2 mögliche Szenarien:

1. Für den verwendeten Treiber ist ein kompatibler Ersatztreiber verfügbar.
2. Für den verwendeten Treiber ist kein kompatibler Ersatztreiber verfügbar.

Kompatibler Ersatztreiber ist verfügbar

Wenn in WinCC ein Treiber verfügbar ist, der die verwendete Steuerung oder eine kompatible Steuerung adressiert, dann wird der Treiber während der Migration automatisch ersetzt.

Wenn der verwendete Treiber ersetzt wird, erhalten Sie eine entsprechende Warnmeldung.

Prüfen Sie in diesem Fall nach der Migration mithilfe des Migrationsprotokolls, ob noch alle externen Variablen und Bereichszeiger gültig sind.

Wenn der verwendete Treiber ersetzt wird, werden alle Verbindungsparameter auf die Standardwerte zurückgesetzt. Der CPU-Typ wird der entsprechenden Steuerung angepasst. Die Eigenschaften der verbundenen Variablen und Bereichszeiger werden nicht verändert.

Kompatibler Ersatztreiber ist nicht verfügbar

Wenn in WinCC kein Treiber verfügbar ist, der die verwendete Steuerung oder eine kompatible Steuerung adressiert, dann wird die projektierte Verbindung nicht migriert. Alle externen Variablen, die über diesen Treiber mit der Steuerung verbunden waren, werden in interne Variablen konvertiert. Externe Variableneigenschaften gehen bei der Konvertierung verloren, z. B. die Adresse in der Steuerung. Alle Änderungen an den Variablen werden im Migrationsprotokoll festgehalten.

Nach der Migration und dem Projektieren einer Verbindung müssen Sie die Variablen wieder mit der Steuerung verbinden.

Wenn der Treiber nicht migriert wird, erhalten Sie eine entsprechende Warnmeldung.

Umsetzung der Kommunikationstreiber

Die nachfolgenden Tabellen zeigen die Abbildung der Kommunikationstreiber von WinCC flexible auf WinCC.

Unterstützte Kommunikationstreiber

WinCC flexible	WinCC
Allen Bradley DF1	Allen Bradley DF1
Mitsubishi FX	Mitsubishi FX
Modicon MODBUS	Modicon MODBUS RTU
Modicon MODBUS TCP/IP	Modicon MODBUS TCP/IP
Omron Hostlink/Multilink	Omron HostLink
OPC	OPC
SIMATIC HMI HTTP Protocol	SIMATIC HMI HTTP Protocol
SIMATIC S7 200	SIMATIC S7 200
SIMATIC S7 300/400	SIMATIC S7 300/400

Kompatible Kommunikationstreiber

WinCC flexible	WinCC
Allen Bradley DH485	Allen Bradley DF1
Allen Bradley E/IP C.Logix	Allen Bradley EtherNet/IP
Mitsubishi Protocol 4	Mitsubishi MC TCP/IP [1)]
1) Wenn Sie eine Steuerung FX2N verwenden, müssen nach der Migration den Kommunikationstreiber Mitsubishi FX einstellen.	

Nicht unterstützte Kommunikationstreiber

WinCC flexible	WinCC
GE Fanuc SNP	nicht unterstützt
ISAC	nicht unterstützt
LG GLOFA-GM	nicht unterstützt
SIMATIC 500/505 DP	nicht unterstützt
SIMATIC 500/505 seriell	nicht unterstützt
SIMATIC S5 AS511	nicht unterstützt
SIMATIC S5 DP	nicht unterstützt
Telemecanique Uni-Telway	nicht unterstützt

Siehe auch

Migration (WinCC flexible) (Seite 44)

Unterstützte Bediengeräte (WinCC flexible) (Seite 50)

Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC flexible) (Seite 61)

Migration von Bereichszeigern (WinCC flexible) (Seite 62)

Migration von Variablen (WinCC flexible) (Seite 62)

Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC flexible)

Einleitung

Wenn eine Verbindung nicht migriert wird, haben Sie folgende Möglichkeiten:

- Ändern der Projektierung in WinCC flexible vor der Migration
- Ändern der Projektierung nach der Migration

Ändern einer Verbindung vor der Migration

Wenn der für die Verbindung gewählte Kommunikationstreiber von der Migration nicht unterstützt wird, müssen Sie einen Treiber wählen, der von der Migration unterstützt wird. Das Bediengerät muss den gewählten Kommunikationstreiber ebenfalls unterstützen.

Falls für das verwendete Bediengerät kein passender Treiber verfügbar ist, müssen Sie ein geeignetes Bediengerät verwenden. Passen Sie anschließend die Projektierung in WinCC flexible an und übersetzen Sie das Projekt neu. Nach erfolgreicher Anpassung und Übersetzung migrieren Sie das Projekt. Die Verbindung wird dann ebenfalls erfolgreich migriert.

Ändern einer Verbindung nach der Migration

Wenn Sie ein WinCC flexible-Projekt migrieren, bei dem die Verbindung nicht migriert wird, werden alle externen Variablen dieser Verbindung auf interne Variablen abgebildet. Für jede betroffene Variable erhalten Sie einen entsprechenden Eintrag im Migrationsprotokoll.

Nach der Migration müssen Sie eine neue Verbindung projektieren. Für diese Verbindung wählen Sie einen Kommunikationstreiber, der das verwendete Bediengerät unterstützt. Falls für das verwendete Bediengerät kein passender Treiber verfügbar ist, müssen Sie ein geeignetes Bediengerät verwenden.

Wenn Sie die neue Verbindung projektiert haben, müssen Sie alle Variablen neu verbinden, die von der Migration auf interne Variablen abgebildet wurden. Um die Nachprojektierung zu vereinfachen, öffnen Sie das Migrationsprotokoll. Projektieren Sie die Variablen entsprechend dem Protokolleintrag nach.

Wenn in dem Projekt Bereichszeiger enthalten waren, müssen Sie diese ebenfalls nachprojektieren. Entsprechende Einträge finden Sie auch für die Bereichszeiger im Migrationsprotokoll.

Nähere Hinweise zum Migrationsprotokoll finden Sie im Kapitel Projekte aus WinCC flexible migrieren (WinCC flexible) (Seite 47).

Siehe auch

Migration von Verbindungen (WinCC flexible) (Seite 59)

Projekte aus WinCC flexible migrieren (WinCC flexible) (Seite 47)

Migration von Bereichszeigern (WinCC flexible)

Einleitung

Die Migration der Bereichszeiger ist abhängig vom verwendeten Kommunikationstreiber.

- Wenn der verwendete Kommunikationstreiber durch die Migration unterstützt wird, dann werden Bereichszeiger aus WinCC flexible bei der Migration unverändert übernommen.
- Wenn der verwendete Kommunikationstreiber durch die Migration nicht unterstützt wird, dann werden Bereichszeiger nicht migriert.

Migration der Bereichszeiger

Wenn die durch die Bereichszeiger verwendete Verbindung vollständig migriert wird, dann werden auch die Bereichszeiger vollständig migriert. Die Parameter der Bereichszeiger werden unverändert übernommen.

Wenn Sie den Kommunikationstreiber der Verbindung vor der Migration ersetzen, dann prüfen Sie nach der Migration, ob alle Parameter der Bereichszeiger weiterhin gültig sind.

Wenn die durch den Bereichszeiger verwendete Verbindung nicht migriert wird, dann werden auch die Bereichszeiger nicht migriert. Nähere Hinweise hierzu finden Sie im Kapitel Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC flexible) (Seite 61).

Siehe auch

Migration von Verbindungen (WinCC flexible) (Seite 59)

Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC flexible) (Seite 61)

Migration von Variablen (WinCC flexible)

Einleitung

Bei der Migration von Variablen müssen Sie einige Besonderheiten beachten. Folgende Aspekte sind zu unterscheiden:

- Migration der Datentypen von Variablen
- Migration interner Variablen
- Migration externer Variablen
- Namen von Variablen
- Grenzwerte von Variablen

Migration der Datentypen

WinCC verfügt teilweise über andere Datentypen und verwendet andere Datentypnamen als WinCC flexible. Bei der Migration einer betreffenden Variable wird der Datentyp aus WinCC flexible auf den entsprechenden Datentyp in WinCC abgebildet. Nähere Hinweise hierzu finden Sie im Kapitel Migration der Datentypen (WinCC flexible) (Seite 80).

Migration interner Variablen

Interne Variablen werden immer vollständig migriert. Lediglich die Datentypnamen und Variablennamen können sich durch die Migration ändern.

Migration externer Variablen

Wenn die durch externe Variablen verwendete Verbindung migriert wird, dann werden auch die externen Variablen migriert. Die Migration externer Variablen ist davon abhängig, ob der verwendete Kommunikationstreiber von der Migration unterstützt wird. Nähere Hinweise hierzu finden Sie im Kapitel Migration von Verbindungen (WinCC flexible) (Seite 59).

Wenn die durch externe Variablen verwendete Verbindung nicht migriert wird, dann werden auch die externen Variablen nicht migriert. Die externen Variablen werden dann auf interne Variablen abgebildet. Nach der Migration müssen Sie eine neue Verbindung projektieren und die Variablen wieder mit den Variablen der Steuerung verbinden.

Migration der Namen von Variablen

In WinCC flexible dürfen Variablen, die in verschiedenen Ordnern abgelegt werden, den gleichen Namen haben. In WinCC muss der Variablenname auf dem projektierten Bediengerät eindeutig sein. Bei der Migration werden daher gleichnamige Variablen aus verschiedenen Ordnern umbenannt. Nähere Hinweise hierzu finden Sie im Kapitel Grundlagen zur Migration (WinCC flexible) (Seite 45).

Siehe auch

Migration von Verbindungen (WinCC flexible) (Seite 59)

Migration (WinCC flexible) (Seite 44)

Migration von Meldeklassen und Meldegruppen (WinCC flexible) (Seite 64)

Migration von Skripten (WinCC flexible) (Seite 67)

Migration von sprachabhängigen Inhalten (WinCC flexible) (Seite 69)

Migration von Bibliotheken (WinCC flexible) (Seite 71)

Migration der Datentypen (WinCC flexible) (Seite 80)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Migration von Strukturen (WinCC flexible)

Der Datentyp "StringChar" wird in HMI-Anwenderdatentypen in WinCC nicht unterstützt.

Wenn Sie in einem WinCC flexible-Projekt diesen Datentyp in einer Struktur verwendet haben, wird durch die Migration im HMI-Anwenderdatentyp ein ungültiges Element erzeugt.

Sie müssen nach der Migration diesen Anwenderdatentyp in WinCC nacharbeiten und den Anwenderdatentyp freigeben.

Überprüfen Sie dabei auch die Offsets der nachfolgenden Elemente sowie eventuell vorhandene Verschaltungen zwischen einem Bildbaustein und den Elementen des Anwenderdatentyps. Passen Sie die Elemente gegebenenfalls an.

Migration von Archiven

Ablageort von Archiven

In WinCC flexible bestand die Möglichkeit als Ablageort eines Archives eine Datenbank zu verwenden, die bei der Installation von WinCC flexible automatisch eingerichtet wird. (Einstellung "Systemdefinierte Datenquelle"). Diese Möglichkeit steht in WinCC nicht zur Verfügung.

Wenn Sie in ihrem WinCC flexible Projekt diese Einstellung verwendet haben, so wird sie bei der Migration auf "Benutzerdefinierter Name der Datenquelle" geändert. Bevor sie die Archivierung in eine Datenbank nutzen können, müssen sie über die Systemsteuerung von Windows eine ODBC-Datenquelle konfigurieren und den dort angegebenen Namen der Benutzerdatenquelle als "Namen der Datenquelle" beim Archiv in WinCC projektieren.

Migration von Meldeklassen und Meldegruppen (WinCC flexible)

Änderung der Namen von Meldeklassen

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

WinCC flexible	WinCC
Fehler	Alarms
System	System
Warnungen	Events

Nach der Migration können Sie die Namen der Meldeklassen beliebig verändern.

Migration von Meldegruppen

Bei der Migration werden nur die tatsächlich benutzten Meldegruppen migriert.

Meldegruppen mit einer ID von 1-31 werden 1:1 migriert.

In WinCC wird systemseitig pro Meldeklasse eine entsprechende Meldegruppe angelegt. Diese vom System angelegten Meldegruppen erhalten IDs ab Nummer 32 und werden fortlaufend hochgezählt. Die in jedem WinCC-Projekt vorhandenen 4 vordefinierten Meldeklassen belegen durch ihre Meldegruppen automatisch die IDs 32-35. Für jede benutzerdefinierte Meldeklasse wird eine weitere Meldegruppe angelegt und dafür eine weitere ID vergeben. Daher können sich bei Meldegruppen mit IDs > 31 die IDs nach der Migration ändern. Dadurch ändert sich auch die Zuordnung der Meldegruppennamen zu den IDs.

Beispiel:

In dem Beispiel sehen Sie die Vergabe der IDs in WinCC durch die Migration.

Meldegruppen	ID in WinCC flexible	ID in WinCC	
Meldegruppe 1-16	1-16	1-16	Vorbelegt für die Meldegruppen von Systemmeldungen
Meldegruppe 17-31	17-31	17-31	Benutzerdefinierte Meldegruppen
		32-35	In WinCC vorbelegt für die Meldegruppen der vordefinierten Meldeklassen.
Meldegruppe 32	32	36	Geänderte Zuordnung von ID zu Meldegruppe in WinCC
Meldegruppe 33	33	37	Geänderte Zuordnung von ID zu Meldegruppe in WinCC

Beachten Sie außerdem:

Bei der Migration von Meldegruppen mit vermeintlich gleichem Gruppennamen passt die Migration die Namen an. Dieser Fall tritt ein, wenn ein Gruppenname z. B. ein Leerzeichen am Namensende enthält. Die Migration löscht alle vorhandenen Leerzeichen am Namensende. Wenn 2 Gruppen durch den Löschvorgang den gleichen Gruppennamen erhalten, erweitert die Migration den Gruppennamen nachfolgender Meldegruppen mit dem Suffix "#Mign", wobei "n" eine fortlaufende Nummer ist.

Beispiel:

In WinCC flexible existieren folgende Meldegruppen:

"AlarmGroup_18"

"AlarmGroup_18 " - Gruppenname enthält 1 Leerzeichen

"AlarmGroup_18 " - Gruppenname enthält 2 Leerzeichen

Die Meldegruppe "AlarmGroup_18" ist die Meldegruppe mit der höchsten Nummer.

Ergebnis nach der Migration:

"AlarmGroup_18"

"AlarmGroup_18#Mig1"

"AlarmGroup_18#Mig1.1"

Änderung der Namen von Meldeklassen

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

WinCC flexible	WinCC
Fehler	Errors
System	System
Warnungen	Warnings

Nach der Migration können die Namen der Meldeklassen beliebig verändert werden.

Anzeige von ALARM_S Meldungen und SIMATIC SFM Meldungen

In WinCC flexible können Sie in integrierten Projekten die Anzeigeklassen für ALARM_S-Meldungen aktivieren. Die Anzeige von SIMATIC SFM-Meldungen aktivieren Sie in WinCC flexible über eine separate Einstellung. In WinCC entfällt die separate Einstellung für die Aktivierung der Anzeige von SIMATIC SFM-Meldungen. Die Anzeige von SIMATIC SFM-Meldungen steuern Sie, wie auch die Anzeige von ALARM_S-Meldungen, in WinCC ausschließlich über die Aktivierung der entsprechenden Anzeigeklasse.

Durch das geänderte Konzept kann sich die Anzeige der Meldungen nach der Migration ändern.

Wenn im WinCC flexible-Projekt alle Anzeigeklassen für ALARM_S-Meldungen aktiviert und die Anzeige von SIMATIC SFM-Meldungen deaktiviert sind, werden nach der Migration ALARM_S-Meldungen und SIMATIC SFM-Meldungen angezeigt.

Damit nach der Migration nur ALARM_S-Meldungen angezeigt werden, müssen Sie nach der Migration in STEP 7 die SIMATIC SFM-Meldungen einer nicht verwendeten Anzeigeklasse zuordnen. In WinCC müssen Sie dann diese Anzeigeklasse deaktivieren.

Wenn im WinCC flexible-Projekt alle Anzeigeklassen für ALARM_S-Meldungen deaktiviert und die Anzeige von SIMATIC SFM-Meldungen aktiviert sind, werden nach der Migration keine ALARM_S-Meldungen und keine SIMATIC SFM-Meldungen angezeigt.

Damit nach der Migration nur SIMATIC SFM-Meldungen angezeigt werden, müssen Sie nach der Migration in STEP 7 die SIMATIC SFM-Meldungen einer nicht verwendeten Anzeigeklasse zuordnen. In WinCC müssen Sie dann diese Anzeigeklasse aktivieren.

Die Anzeigeklasse ist abhängig von den Einstellungen in STEP 7. Die Standardeinstellung für SIMATIC SFM-Meldungen in Step 7 ist die Anzeigeklasse "0". Um die Anzeige in WinCC zu aktivieren, muss die Anzeigeklasse "0" aktiviert werden.

Die Anzeigeklassen aktivieren Sie in WinCC in den Runtime-Einstellungen des jeweiligen Bediengeräts in der Kategorie "Meldungen".

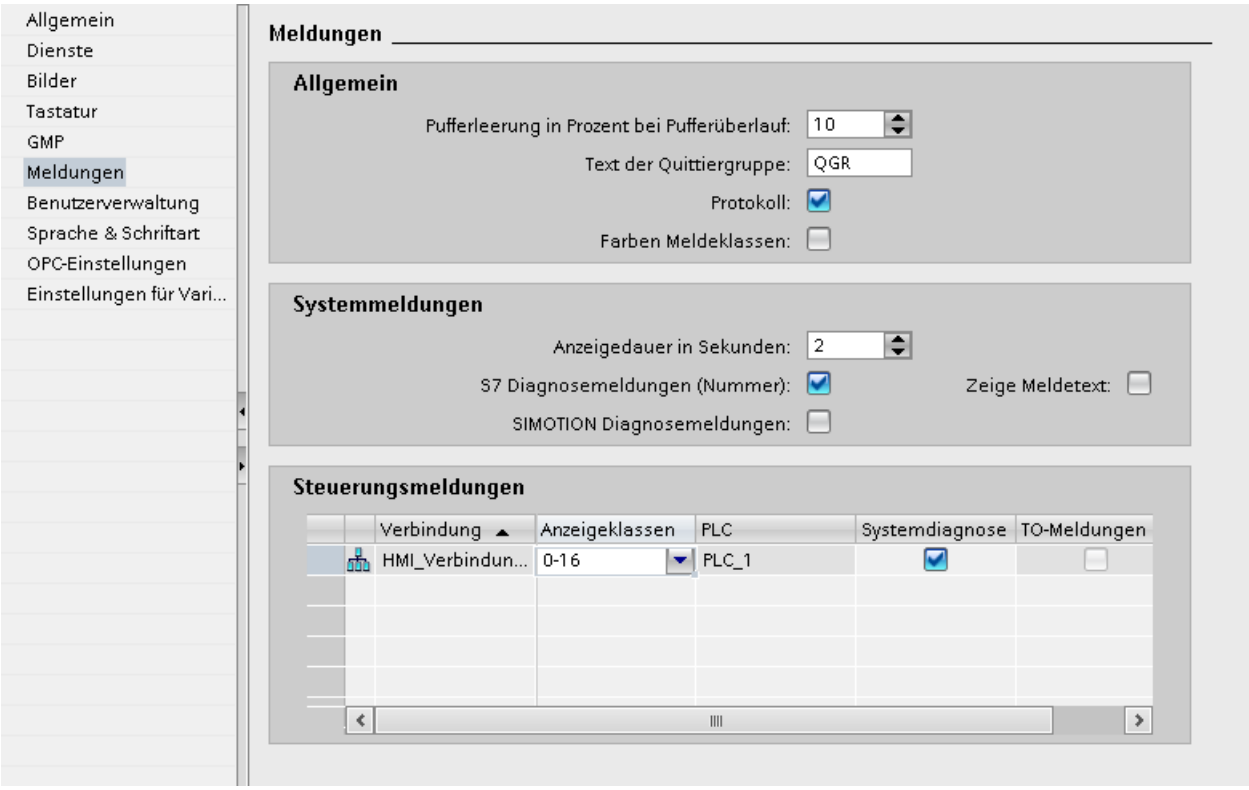

Siehe auch

Migration von Variablen (WinCC flexible) (Seite 62)

Migration von Skripten (WinCC flexible)

Einleitung

Die Migration unterstützt VB-Skripte, die in WinCC flexible erstellt wurden. Damit ein VB-Skript erfolgreich migriert wird, muss es bereits in WinCC flexible funktionsfähig sein.

Hinweis

Script-Fehler

Vorhandene Script-Fehler können beim ersten Testlauf nach der Migration am schnellsten durch einen installierten Script Debugger und die Verwendung des Diagnose-Controls gefunden werden.

Migration eines VB-Skripts

Ein Skript wird bei der Migration analysiert und bei Bedarf an das Systemverhalten von WinCC angepasst. Folgendes wird angepasst:

- Systemfunktionen, deren Namen sich geändert haben, werden umbenannt.
  Folgende Systemfunktionen sind betroffen:

Funktionsname in WinCC flexible	Funktionsname in WinCC
IncreaseValue	IncreaseTag
DecreaseValue	DecreaseTag
SetBrightness	SetAndGetBrightness
SetValue	SetTag

- Der Zugriff auf Variablen als Parameter von Systemfunktionen wird an das Systemverhalten von WinCC angepasst. In WinCC werden Skripte nicht mehr transformiert wie in WinCC flexible. Die Skripte werden direkt als Quellcode ausgeführt. Daher gelten die strengeren Regeln für die VBS-Syntax. Der Aufruf einer Variable wird immer mit Anführungszeichen migriert. Wenn für eine Systemfunktion mehrere Parametertypen erlaubt sind, werden diese mit dem Schlüsselwort "SmartTags" migriert.
  Beispiel 1:
  Der Wert der Variable "temperature" soll um den Wert "1" erhöht werden.
  Gültige Ausdrücke in WinCC flexible:
  IncreaseValue temperature, 1
  IncreaseValue "temperature", 1
  IncreaseValue SmartTags("temperature"), 1
  Gültiger Ausdruck in WinCC:
  IncreaseTag "temperature", 1
  Beispiel 2:
  Sie verwenden eine Systemfunktion, an der mehrere Parameter erlaubt sind. Der Wert der Variable "temperature" soll um den Wert der Variable "heatcontrol" erhöht werden.
  Gültiger Ausdruck für WinCC flexible:
  IncreaseValue "temperature", "heatcontrol"
  Gültiger Ausdruck für WinCC:
  IncreaseTag "temperature", SmartTags("heatcontrol")
- Von der Migration umbenannte Objekte werden bei Verwendung in einem Skript auch im Skript umbenannt. Nähere Hinweise zur Umbenennung von Objekten finden Sie im Kapitel Grundlagen zur Migration (WinCC flexible) (Seite 45). Bei der Umbenennung müssen Sie folgendes beachten:
  Wenn Sie mithilfe eines Skripts Objekte ansprechen, deren Namen durch das Skript dynamisch erzeugt werden, können die Objektnamen im Skript durch die Migration nicht automatisch angepasst werden. In einem solchen Fall müssen Sie die Erzeugung der Objektnamen im Skript nach der Migration korrigieren.

Siehe auch

Migration von Variablen (WinCC flexible) (Seite 62)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Migration von sprachabhängigen Inhalten (WinCC flexible)

Einleitung

In WinCC stehen Ihnen für die Projektierung in unterschiedlichen Sprachen die gleichen Möglichkeiten zur Verfügung wie in WinCC flexible. Bei der Migration eines Projekts werden alle Sprachen migriert, die WinCC unterstützt.

Migration sprachabhängiger Inhalte

Bei der Migration werden folgende sprachabhängigen Inhalte migriert:

- Projektsprachen
- Projekttexte
- Schriftarten für die Darstellung in Runtime
- Sprachabhängige Grafiken

Bei der Migration sprachabhängiger Inhalte müssen Sie folgende Dinge beachten:

- Auf dem PC für die Migration muss das Betriebssystem die im Projekt verwendeten Sprachen unterstützen.
- Auf dem PC für die Migration müssen die Schriftarten installiert sein, die Sie für die Darstellung in Runtime verwenden.
- Wörterbücher werden von der Migration nicht unterstützt.

Editiersprache integrierter Projekte nach der Migration

Bei der Migration eines integrierten Projekts bringen die zu migrierenden Projektanteile aus STEP 7 und WinCC flexible ihre jeweiligen Einstellungen für die Editiersprache mit. In WinCC gibt es nur noch eine Editiersprache für alle Projektanteile. Die Migration aktiviert für das migrierte Projekt die Editiersprache, die in STEP 7 vor der Migration eingestellt war. Wenn diese Einstellung nicht mit der Einstellung aus WinCC flexible übereinstimmt, dann sind die projektierten Texte in WinCC nicht mehr sichtbar. An den Verwendungsstellen der Texte wird kein Text angezeigt oder es ist nur noch der Eintrag "Text" zu sehen. Um die Texte sichtbar zu machen, müssen Sie die Editiersprache wechseln. Klicken Sie am rechten Rand des TIA Portals auf die Taskcard "Aufgaben" und wählen Sie im Bereich "Sprachen & Ressourcen" die richtige Editiersprache aus.

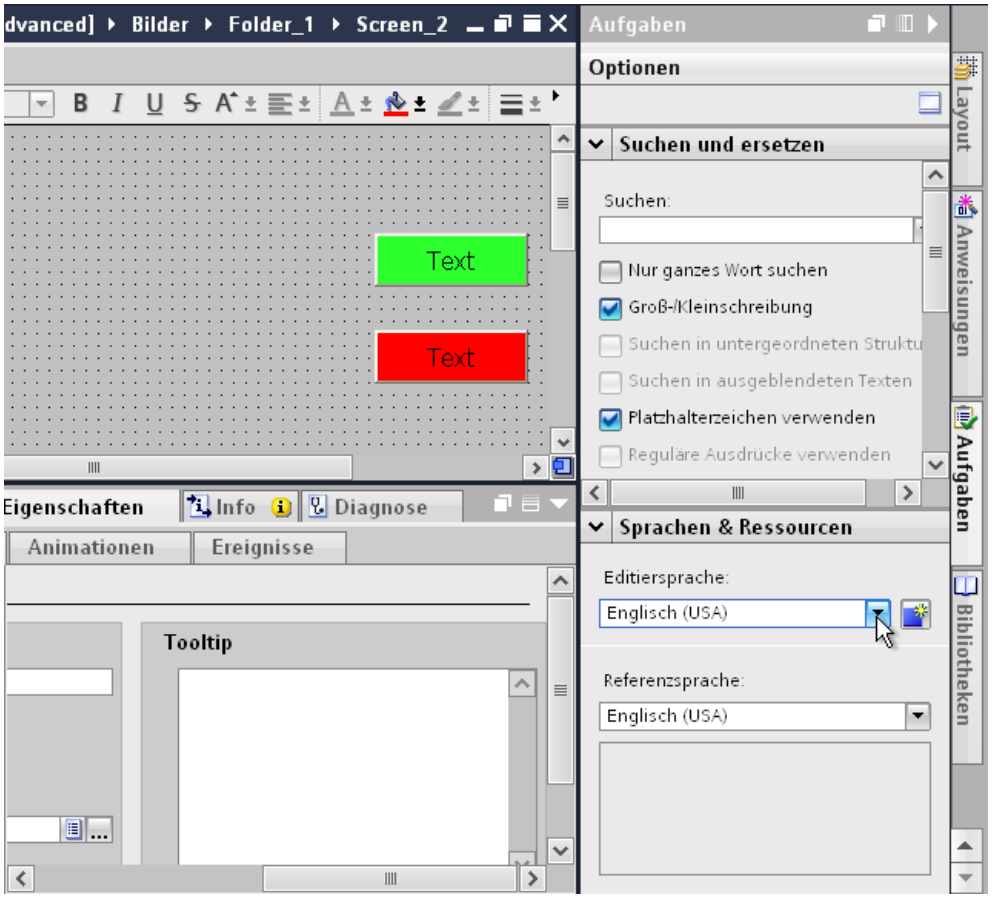

Nicht unterstützte Sprachen

Die Migration von sprachabhängigen Inhalten ist abhängig davon, ob WinCC die jeweilige Sprache unterstützt.

Wenn ein Projekt nur Projektsprachen enthält, die WinCC nicht unterstützt, wird das Projekt nicht migriert.

Wenn ein Projekt unterstützte und nicht unterstützte Projektsprachen enthält, werden nur die unterstützten Sprachen migriert. Die Bearbeitungssprache und die Referenzsprache werden auf eine unterstützte Sprache gesetzt.

Folgende Sprachen werden von WinCC nicht unterstützt:

- Arabisch
- Hebräisch
- Dhivehi
- Gujara
- Kannada
- Tamil
- Telugu
- Urdu
- Punjabi
- Persisch
- Syrisch

Siehe auch

Migration von Variablen (WinCC flexible) (Seite 62)

Migration von Bibliotheken (WinCC flexible)

Einleitung

Bei der Migration von Bibliotheken müssen Sie zwei Fälle unterscheiden:

1. Migration der Projektbibliothek
2. Migration einer globalen Bibliothek

Migration der Projektbibliothek

Eine Projektbibliothek wird gemeinsam mit den Projektdaten in der Projektdatei gespeichert. Daher wird eine Projektbibliothek mit den gleichen Einschränkungen migriert wie die Projektdaten.

Migration einer globalen Bibliothek

Globale Bibliotheken werden von der Migration nicht unterstützt. Die im Projekt verwendeten Bibliotheksobjekte werden jedoch migriert. Die Bibliotheksobjekte werden bei der Verwendung in das Projekt kopiert und haben danach keine Verbindung mehr zur Bibliothek.

Um eine globale Bibliothek zu migrieren, müssen Sie die in der Bibliothek enthaltenen Objekte in die Projektbibliothek verschieben oder kopieren. Bei der Migration werden die Objekte dann migriert. In WinCC verschieben Sie die migrierten Objekte in eine neu zu erstellende, globale Bibliothek. Sie können sowohl einzelne Objekte als auch ganze Bibliothekskategorien verschieben oder kopieren.

Siehe auch

Migration von Variablen (WinCC flexible) (Seite 62)

2.8.1.3 Runtime-Daten migrieren (WinCC flexible)

Migration von Runtime-Daten (WinCC flexible)

Einleitung

Bei der Migration eines Projekts werden von der Migration lediglich die Projektierungsdaten migriert. Die Runtime-Daten sind davon nicht betroffen. Die Runtime-Daten müssen Sie nach der Migration aktualisieren.

Die Runtime-Daten setzen sich wie folgt zusammen:

- Runtime-Projekt
  Das Runtime-Projekt enthält die übersetzten Projektdaten.
- Rezepturdaten und die Benutzerverwaltung
  Die Rezepturdaten und die Benutzerverwaltung sind Daten, die in Runtime verändert werden können.
- Archivdaten
  Die Daten aus Variablenarchiven und Meldearchiven werden in Runtime erfasst und archiviert.

Migration von Rezepturdaten und Benutzerverwaltung

Wenn die Rezepturdaten und die Benutzerverwaltung in Runtime geändert wurden, müssen Sie diese Daten vom Bediengerät sichern, bevor Sie das migrierte Projekt laden.

Abhängig vom verwendeten Bediengerät haben Sie verschiedene Möglichkeiten, die oben genannten Daten zu sichern.

Wenn das Bediengerät externe Speichermedien unterstützt und die Rezepturdaten dort gespeichert sind, verbleiben die Daten auf dem Speichermedium. Externe Speichermedien sind z. B. eine StorageCard oder ein Netzlaufwerk. Die Daten verwenden Sie nach der Migration weiter. Die Benutzerverwaltung wird nicht auf einem externen Speichermedium gespeichert, daher müssen Sie diese Daten sichern, z. B. mit ProSave.

Wenn die Rezepturdaten im internen Speicher des Bediengeräts gespeichert sind, sichern Sie diese Daten auf einem externen Speichermedium. Für die Sicherung der Rezepturdaten verwenden Sie ProSave. Wenn Sie in Ihrem Projekt bereits entsprechende Systemfunktionen projektiert haben, verwenden Sie die Systemfunktionen. Folgende Systemfunktionen stehen für die Rezepturdaten zur Verfügung:

- "ExportiereDatensätze" für die Sicherung
- "ImportiereDatensätze" für die Wiederherstellung

Das Runtime-Projekt aktualisieren Sie, indem Sie das Projekt in WinCC neu übersetzen und auf das Bediengerät laden.

Nachdem Sie das migrierte Projekt auf das Bediengerät geladen haben, transferieren Sie die Rezepturdaten und die Benutzerverwaltung wieder auf das Bediengerät.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Rezepturdaten und Benutzerverwaltung sichern (WinCC flexible) (Seite 73)

Rezepturdaten und Benutzerverwaltung wiederherstellen (WinCC flexible) (Seite 74)

Archivdaten sichern (WinCC flexible) (Seite 75)

Rezepturdaten und Benutzerverwaltung sichern (WinCC flexible)

Einleitung

Um die Rezepturdaten und die Benutzerverwaltung in einem migrierten Projekt weiter zu verwenden, müssen Sie diese Daten zuerst vom Bediengerät sichern. Anschließend laden Sie die Daten in das migrierte WinCC-Projekt. Zur Sicherung der Daten verwenden Sie ProSave.

Voraussetzung

- Das WinCC flexible-Projekt läuft auf dem Bediengerät in Runtime.
- Das Bediengerät ist mit einem PC verbunden, auf dem ProSave installiert ist.

Vorgehen

Um die Rezepturdaten und die Benutzerverwaltung zu sichern, gehen Sie folgendermaßen vor:

1. Starten Sie ProSave.
2. Wählen Sie auf der Registerkarte "Allgemein" den Gerätetyp und die Verbindungsparameter aus.
3. Aktivieren Sie die Registerkarte "Backup".
4. Wählen Sie im Feld "Datenart" den Eintrag "Rezepturen aus dem Gerätespeicher" aus. Wählen Sie nicht "Komplett Backup", da Sie sonst bei der Wiederherstellung die Rezepturdaten nicht mehr separat auswählen können.
5. Navigieren Sie im Feld "Speichern unter" zum gewünschten Ablageort und klicken Sie anschließend auf die Schaltfläche "Start Backup".
   Die Rezepturdaten werden gesichert.
6. Wählen Sie im Feld "Datenart" den Eintrag "Benutzerverwaltung" aus und klicken Sie auf die Schaltfläche "Start Backup".
   Die Benutzerverwaltung wird gesichert.

Nähere Hinweise hierzu finden Sie in der Online-Hilfe von ProSave.

Alternative Vorgehensweise

ProSave wird standardmäßig mit WinCC flexible installiert. Auf dem Projektierungs-PC steht Ihnen innerhalb von WinCC flexible über den Menübefehl "Projekt > Transfer" der gesamte Funktionsumfang von ProSave zur Verfügung.

Alternativ sichern Sie die Rezepturdaten und die Benutzerverwaltung über das in WinCC flexible integrierte ProSave. Starten Sie WinCC flexible und wählen Sie den Menübefehl "Projekt > Transfer > Sicherung". Sichern Sie die Rezepturdaten und die Benutzerverwaltung, wie in den Schritten 4-6 beschrieben.

Siehe auch

Migration von Runtime-Daten (WinCC flexible) (Seite 72)

Rezepturdaten und Benutzerverwaltung wiederherstellen (WinCC flexible)

Einleitung

Um die gesicherten Rezepturdaten und die Benutzerverwaltung nach der Migration weiter zu verwenden, müssen Sie zuerst das migrierte Projekt übersetzen und auf das Bediengerät laden. Anschließend transferieren Sie die gesicherten Daten auf das Bediengerät. Zur Wiederherstellung der Daten verwenden Sie ProSave.

Voraussetzung

- Das migrierte Projekt ist auf das Bediengerät transferiert und läuft in Runtime.
- Das Bediengerät ist mit einem PC verbunden, auf dem ProSave installiert ist.

Vorgehen

Um die gesicherten Rezepturdaten und die Benutzerverwaltung auf das Bediengerät zu laden, gehen Sie folgendermaßen vor:

1. Starten Sie ProSave.
2. Wählen Sie auf der Registerkarte "Allgemein" den Gerätetyp und die Verbindungsparameter aus.
3. Aktivieren Sie die Registerkarte "Restore".
4. Navigieren Sie im Feld "Öffnen von ..." zum Speicherort der gesicherten Rezepturdaten und wählen Sie die Datei aus.
5. Klicken Sie auf die Schaltfläche "Start Restore".
   Die Rezepturdaten werden auf das Bediengerät transferiert.
6. Wiederholen Sie die Schritte 4-5 zur Wiederherstellung der Benutzerverwaltung.
   Die Benutzerverwaltung wird auf das Bediengerät transferiert.

Nähere Hinweise hierzu finden Sie in der Online-Hilfe von ProSave.

Alternative Vorgehensweise

ProSave wird standardmäßig mit WinCC installiert. Auf dem Projektierungs-PC steht Ihnen innerhalb von WinCC über den Menübefehl "Projekt > Transfer" der gesamte Funktionsumfang von ProSave zur Verfügung.

Die Rezepturdaten und die Benutzerverwaltung stellen Sie alternativ über das in WinCC integrierte ProSave wieder her. Starten Sie WinCC und wählen Sie den Menübefehl "Online > Bediengeräte Wartung > Wiederherstellen". Stellen Sie die Rezepturdaten und die Benutzerverwaltung wieder her, wie in den Schritten 4-6 beschrieben.

Siehe auch

Migration von Runtime-Daten (WinCC flexible) (Seite 72)

Archivdaten sichern (WinCC flexible)

Einleitung

Wenn ein Bediengerät externe Speichermedien unterstützt und die Archivdaten dort gespeichert sind, verbleiben die Daten auf dem Speichermedium. Externe Speichermedien sind z. B. eine StorageCard oder ein Netzlaufwerk. Wenn ein Archiv auf einem externen Speichermedium gespeichert ist, greift das migrierte Projekt nach der Migration wieder auf diesen Speicherort zu. In diesem Fall müssen die Archivdaten nicht gesichert werden.

Archivdaten sichern

Wenn Sie Archivdaten vor der Migration extern sichern wollen, haben Sie folgende Möglichkeiten:

- Sicherung mit der Systemfunktion "ArchiviereProtokolldatei".
  Wenn Sie im WinCC flexible-Projekt bereits die Systemfunktion "ArchiviereProtokolldatei" projektiert haben, benutzen Sie diese Funktion zur Sicherung der Daten.
- Kopieren der Archivdateien per Copy&Paste mit dem Windows Explorer auf ein externes Speichermedium oder ein Netzlaufwerk.

Siehe auch

Migration von Runtime-Daten (WinCC flexible) (Seite 72)

2.8.1.4 Integrierte Projekte migrieren (WinCC flexible)

Migration integrierter Projekte (WinCC flexible)

Einleitung

In einem in STEP 7 integrierten Projekt sind enthaltene Steuerungen und Bediengeräte durch die Projektierung miteinander verbunden. Auch die Projektierungsdaten von WinCC flexible und STEP 7 haben Verbindungen zueinander. Bei der Migration eines integrierten Projekts wird das komplette Projekt mit den Anteilen aus WinCC flexible und STEP 7 migriert. Die Verbindungen bleiben erhalten.

Hinweis

Bevor sie ein integriertes Projekt migrieren, sollten sie dieses Projekt in WinCC flexible übersetzen und speichern. Eine fehlerfreie Übersetzung stellt sicher, dass die Daten in WinCC flexible und in STEP 7 synchronisiert sind.

Migration eines integrierten Projekts

Bei der Migration eines integrierten Projekts gelten für den WinCC flexible-Anteil die gleichen Voraussetzungen wie für die Migration eines nicht integrierten WinCC flexible-Projekts. Die im WinCC flexible-Anteil enthaltenen Objekte und Eigenschaften müssen von WinCC unterstützt werden, z. B. das Bediengerät oder der Kommunikationstreiber. An einer projektierten Verbindung muss die Eigenschaft "Online" aktiviert sein. Eine Verbindung, deren Eigenschaft "Online" deaktiviert ist, wird nicht migriert.

Neben den Voraussetzungen für den WinCC flexible-Anteil gelten weitere Voraussetzungen für den STEP 7-Anteil des integrierten Projekts. Die im STEP 7 V5.4 SP5 oder V5.5-Anteil enthaltenen Objekte und Eigenschaften müssen in STEP 7 unterstützt werden. Nähere Hinweise hierzu finden Sie in der Dokumentation zu STEP 7.

Um ein integriertes Projekt vollständig zu migrieren und nachbearbeiten zu können, müssen auf dem PC für die Migration folgende Komponenten installiert sein:

- STEP 7 V5.4 SP5 oder STEP 7 V5.5
- WinCC flexible 2008 SP2 oder WinCC flexible 2008 SP3 oder WinCC flexible 2008 SP5
- STEP 7

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible) (Seite 45).

Ein integriertes Projekt wird immer vollständig migriert. Wenn Sie lediglich das darin enthaltene WinCC flexible-Projekt migrieren wollen, müssen Sie es vor der Migration aus dem STEP 7-Projekt lösen. Um das Projekt aus der Integration zu lösen, öffnen Sie das Projekt in STEP 7 V5.4 SP5 oder V5.5. Öffnen Sie im SIMATIC-Manager das WinCC flexible-Projekt. Das Projekt wird mit WinCC flexible geöffnet. Wählen Sie in WinCC flexible den Menübefehl "Projekt >

Kopiere Projekt aus STEP 7". WinCC flexible speichert eine nicht integrierte Kopie des Projekts.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Integriertes Projekt migrieren (WinCC flexible) (Seite 77)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

Integriertes Projekt migrieren (WinCC flexible)

Einleitung

Bei der Migration eines integrierten Projekts werden sowohl die Anteile aus dem WinCC flexible-Projekt als auch aus dem STEP 7-Projekt migriert. Deshalb müssen Sie für die Migration die Projektdatei mit der Dateiendung "*.s7p" wählen. Bei der Migration werden die Daten aus dem bestehenden Projekt kopiert und in ein neues Projekt migriert. Sie können nicht in ein bestehendes Projekt migrieren.

Die Migration lässt sich sowohl in der Portalansicht als auch in der Projektansicht aufrufen.

Migrieren Sie ein Projekt nur in einem neu gestarteten TIA Portal.

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible).

Voraussetzung

- STEP 7 V5.4 SP5 oder STEP 7 V5.5 und alle verwendeten Optionspakete sind installiert.
- STEP 7 und alle verwendeten Optionspakete sind installiert.
- Das TIA Portal ist neu gestartet.
- In WinCC ist kein Projekt geöffnet.
- Ein integriertes Projekt ist vorhanden.
- Das integrierte Projekt ist nicht geöffnet.

Vorgehen

Um ein integriertes Projekt in der Portalansicht zu migrieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die Aktion "Start > Projekt migrieren".

2. Navigieren Sie im Feld "Quellpfad" zu dem Projekt, das Sie migrieren wollen.

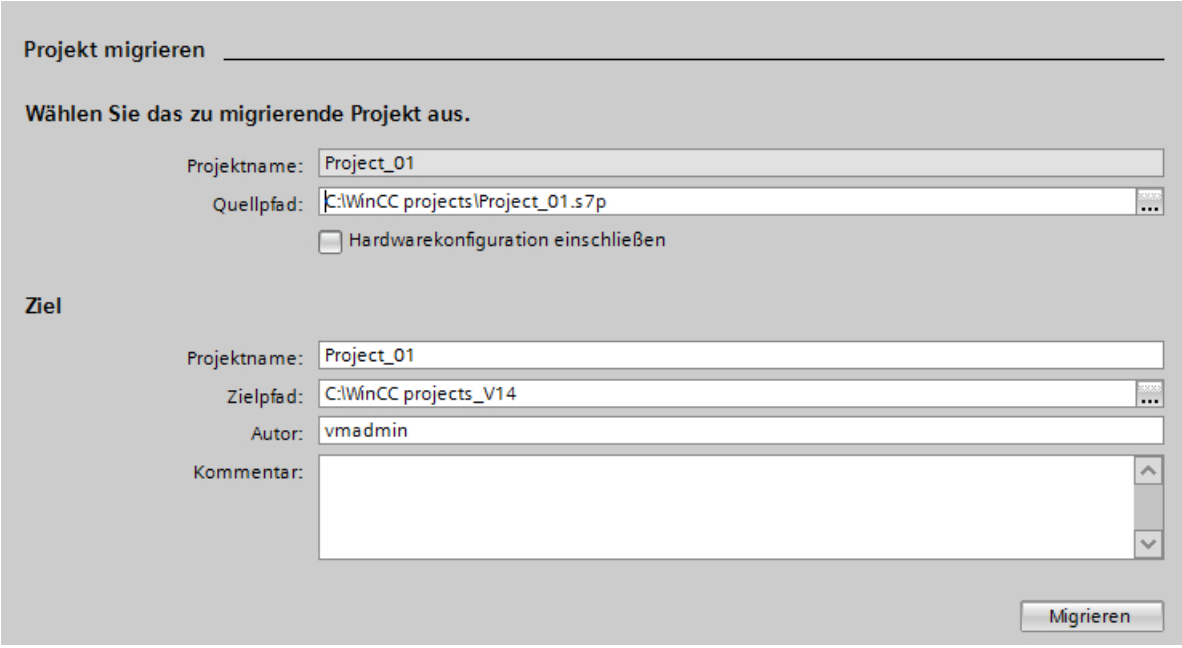

3. Selektieren Sie die Projektdatei "*.s7p".
4. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.
5. Um das Projekt mit Hardwarekonfiguration zu migrieren, aktivieren Sie "Hardware-Konfiguration einschließen".

6. Klicken Sie auf "Migrieren".
   Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
   - Die Projektansicht wird geöffnet.
   - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
   - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
   - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
   - Nach Abschluss der Migration wird eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.
7. Speichern Sie das Projekt nach Ende der Migration.

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte Bediengerät und für jede migrierte Steuerung ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten.

Hinweis

Integrierte Projekte mit Meldeanzeigen

Bei der Migration des Projekts können die Meldeklassen in der Meldeanzeige deaktiviert werden. Prüfen Sie nach der Migration des Projekts die Einstellungen in der Meldeanzeige. Aktivieren Sie gegebenenfalls im Inspektorfenster der Meldeanzeige unter "Eigenschaften > Allgemein" die erforderlichen Meldeklassen.

Migrationsprotokoll zu einem späteren Zeitpunkt öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Sie können das Protokoll zu einem späteren Zeitpunkt ansehen. Die Protokolldatei öffnen Sie folgendermaßen:

1. Öffnen Sie in der Projektnavigation "Gemeinsame Daten > Protokolle".
2. Doppelklicken Sie die Protokolldatei. Das Migrationsprotokoll wird geöffnet.

Siehe auch

Migration integrierter Projekte (WinCC flexible) (Seite 76)

Grundlagen zur Migration (WinCC flexible) (Seite 45)

2.8.1.5 Referenz (WinCC flexible)

Migration der Datentypen (WinCC flexible)

Einleitung

Zur Harmonisierung der verwendeten Datentypen von Steuerungen und HMI-Systemen werden einige Datentypen der internen HMI-Variablen umbenannt. Die Namensgebung erfolgt IEC konform. Da sich lediglich die Namen ändern, ergeben sich für die Projektierung bei internen Variablen keine Änderungen.

Die nachfolgende Tabelle beschreibt die Abbildung der internen Datentypen aus WinCC flexible auf die Datentypen in WinCC.

Migration der internen Datentypen

Die internen Datentypen werden bei der Migration folgendermaßen aufeinander abgebildet:

Interne Datentypen WinCC flexible	Interne Datentypen WinCC
Bool	Bool
Char	SInt
Byte	USInt
Int	Int
UInt	UInt
Long	DInt
ULong	UDInt
Float	Real
Double	LReal
String	WString
DateTime	DateTime

Migration der externen Datentypen

Wie die Datentypen der verfügbaren Kommunikationstreiber abgebildet werden, finden Sie auf den nachfolgenden Seiten.

Siehe auch

Migration (WinCC flexible) (Seite 44)

Migration der Datentypen von Allen-Bradley DF1 (WinCC flexible) (Seite 81)

Migration der Datentypen von Allen-Bradley DH485 (WinCC flexible) (Seite 82)

Migration der Datentypen von Allen-Bradley Ethernet IP (WinCC flexible) (Seite 82)

Migration der Datentypen von GE Fanuc SNP (WinCC flexible) (Seite 83)

Migration der Datentypen von LG GLOFA GM (WinCC flexible) (Seite 83)

Migration der Datentypen von Mitsubishi FX (WinCC flexible) (Seite 84)
Migration der Datentypen von Modicon Modbus (WinCC flexible) (Seite 85)
Migration der Datentypen von Modicon Modbus TCP/IP (WinCC flexible) (Seite 86)
Migration der Datentypen von Omron Hostlink/Multilink (WinCC flexible) (Seite 86)
Migration der Datentypen von OPC (WinCC flexible) (Seite 87)
Migration der Datentypen von SIMATIC 500/505 DP (WinCC flexible) (Seite 88)
Migration der Datentypen von SIMATIC 500/505 seriell (WinCC flexible) (Seite 88)
Migration der Datentypen von SIMATIC S5 AS511 (WinCC flexible) (Seite 89)
Migration der Datentypen von SIMATIC S5 DP (WinCC flexible) (Seite 90)
Migration der Datentypen von SIMATIC S7 200 (WinCC flexible) (Seite 91)
Migration der Datentypen von SIMATIC S7 300/400 (WinCC flexible) (Seite 91)
Migration der Datentypen von Telemecanique Uni-Telway (WinCC flexible) (Seite 94)
Migration der Datentypen von Mitsubishi Protocol 4 (WinCC flexible) (Seite 85)
Migration der Datentypen von SIMATIC HMI HTTP Protocol (WinCC flexible) (Seite 89)

Migration der Datentypen von Allen-Bradley DF1 (WinCC flexible)

Migration der Datentypen Allen-Bradley DF1

Die Datentypen des Kommunikationstreibers Allen-Bradley DF1 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
ASCII	ASCII
BCD4	UInt
BCD8	UDInt
Bit	Bool
Int	Int
Long	DInt
Real	Real
UInt	UInt
ULong	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Allen-Bradley DH485 (WinCC flexible)

Migration der Datentypen Allen-Bradley DH485

Der Kommunikationstreiber Allen-Bradley DH485 wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Allen-Bradley DF1 ersetzt. Die Datentypen des Kommunikationstreibers Allen-Bradley DH485 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
ASCII	ASCII
Bit	Bool
Int	Int
Long	DInt
Real	Real
UInt	UInt
ULong	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Allen-Bradley Ethernet IP (WinCC flexible)

Migration der Datentypen Allen-Bradley Ethernet IP

Die Datentypen des Kommunikationstreibers Allen-Bradley Ethernet IP werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	Bool
DInt	DInt
Int	Int
Real	Real
SInt	SInt
String	String
UDInt	UDInt
UInt	UInt
USInt	USInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von GE Fanuc SNP (WinCC flexible)

Migration der Datentypen GE Fanuc SNP

Der Kommunikationstreiber GE Fanuc SNP wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers GE Fanuc SNP werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
BCD4	UInt
BCD8	UDInt
Bit	Bool
Byte	USInt
DInt	DInt
Word	UInt
Int	Int
Real	Real
UInt	UInt
DWord	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von LG GLOFA GM (WinCC flexible)

Migration der Datentypen LG GLOFA GM

Der Kommunikationstreiber LG GLOFA GM wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers LG GLOFA GM werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	Bool
Byte	USInt
DInt	DInt
DWord	UDInt
Int	Int
SInt	SInt
String	WString
Time	UDInt
UDInt	UDInt
UInt	UInt

Datentyp in WinCC flexible	Datentyp in WinCC
USInt	USInt
Word	UInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Mitsubishi FX (WinCC flexible)

Migration der Datentypen Mitsubishi FX

Die Datentypen des Kommunikationstreibers Mitsubishi FX werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
12 Bit Block	12-Bit Block
16 Bit Block	16-Bit Block
20 Bit Block	20-Bit Block
24 Bit Block	24-Bit Block
28 Bit Block	28-Bit Block
32 Bit Block	32-Bit Block
4 Bit Block	4-Bit Block
8 Bit Block	8-Bit Block
Bit	Bool
Double	DWord
IEEE-Float	Real
String	String
Word	Word

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Mitsubishi Protocol 4 (WinCC flexible)

Migration der Datentypen Mitsubishi Protocol 4

Der Kommunikationstreiber Mitsubishi Protocol 4 wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Mitsubishi MC TCP/IP ersetzt. Die Datentypen des Kommunikationstreibers Mitsubishi Protocol 4 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
4 Bit Block	4-Bit Block
8 Bit Block	8-Bit Block
12 Bit Block	12-Bit Block
16 Bit Block	16-Bit Block
20 Bit Block	20-Bit Block
24 Bit Block	24-Bit Block
28 Bit Block	28-Bit Block
32 Bit Block	32-Bit Block
Bit	Bool
DInt	DInt
DWord	DWord
Int	Int
Real	Real
String	String
Word	Word

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Modicon Modbus (WinCC flexible)

Migration der Datentypen Modicon Modbus

Der Kommunikationstreiber Modicon Modbus wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Modicon Modbus RTU ersetzt. Die Datentypen des Kommunikationstreibers Modicon Modbus werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-Double	+/- Double
+/-Int	+/- Int
16 Bit Group	16 Bit Group
ASCII	ASCII
Bit	Bit

Datentyp in WinCC flexible	Datentyp in WinCC
Double	Double
Float	Float
Int	Int

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Modicon Modbus TCP/IP (WinCC flexible)

Migration der Datentypen Modicon Modbus TCP/IP

Die Datentypen des Kommunikationstreibers Modicon Modbus TCP/IP werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-Double	+/- Double
+/-Int	+/- Int
16 Bit Group	16 Bit Group
ASCII	ASCII
Bit	Bit
Double	Double
Float	Float
Int	Int

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Omron Hostlink/Multilink (WinCC flexible)

Migration der Datentypen Omron Hostlink/Multilink

Der Kommunikationstreiber Omron Hostlink/Multilink wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Omron Host Link ersetzt. Die Datentypen des Kommunikationstreibers Omron Hostlink/Multilink werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-DEC	Int
+/-LDEC	DInt
ASCII	String
BIN	Bool

Datentyp in WinCC flexible	Datentyp in WinCC
BYTE	Byte
DEC	UInt
IEEE	Real
LDEC	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von OPC (WinCC flexible)

Migration der Datentypen OPC

Die Datentypen des Kommunikationstreibers OPC werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	VT_BOOL
Byte	VT_UI1
Char	VT_I1
Date	VT_DATE
Double	VT_R8
DWord	VT_UI4
Float	VT_R4
Long	VT_I4
Short	VT_I2
String	VT_BSTR
Word	VT_UI2

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC 500/505 DP (WinCC flexible)

Migration der Datentypen SIMATIC 500/505 DP

Der Kommunikationstreiber SIMATIC 500/505 DP wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers SIMATIC 500/505 DP werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-Double	DInt
+/-Int	Int
ASCII	WString
Bit	Bool
Double	UDInt
Int	UInt
Real	Real

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC 500/505 seriell (WinCC flexible)

Migration der Datentypen SIMATIC 500/505 seriell

Der Kommunikationstreiber SIMATIC 500/505 seriell wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers SIMATIC 500/505 seriell werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-Double	DInt
+/-Int	Int
ASCII	WString
Bit	Bool
Double	UDInt
Int	UInt
Real	Real

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC HMI HTTP Protocol (WinCC flexible)

Migration der Datentypen SIMATIC HMI HTTP Protocol

Die Datentypen des Kommunikationstreibers SIMATIC HMI HTTP Protocol werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	Bool
Byte	USInt
Char	SInt
DateTime	DateTime
Double	LReal
Float	Real
Int	Int
Long	DInt
String	WString
UInt	UInt
ULong	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC S5 AS511 (WinCC flexible)

Migration der Datentypen SIMATIC S5 AS511

Der Kommunikationstreiber SIMATIC S5 AS511 wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers SIMATIC S5 AS511 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bit in D	Bool
Bit in W	Bool
DF	DInt
DH	UDInt
KC	WString
KF	Int
KG	Real
KH	UInt
KM	UInt
KT	UDInt

Datentyp in WinCC flexible	Datentyp in WinCC
KY	UInt
KZ	UInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC S5 DP (WinCC flexible)

Migration der Datentypen SIMATIC S5 DP

Der Kommunikationstreiber SIMATIC S5 DP wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers SIMATIC S5 DP werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bit in D	Bool
Bit in W	Bool
DF	DInt
DH	UDInt
KC	WString
KF	Int
KG	Real
KH	UInt
KM	UInt
KT	UDInt
KY	UInt
KZ	UInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC S7 200 (WinCC flexible)

Migration der Datentypen SIMATIC S7 200

Die Datentypen des Kommunikationstreibers SIMATIC S7 200 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	Bool
Byte	Byte
Char	Char
DInt	DInt
DWord	DWord
Int	Int
Real	Real
StringChar	StringChar
Timer	Timer
Word	Word

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von SIMATIC S7 300/400 (WinCC flexible)

Migration der Datentypen SIMATIC S7 300/400

Die Datentypen des Kommunikationstreibers SIMATIC S7 300/400 werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
Bool	Bool
Byte	Byte
Char	siehe unten
Counter	siehe unten
Date	Date
Date and Time	Date_And_Time
DInt	DInt
DWord	DWord
Int	Int
Real	Real
String	String
StringChar	siehe unten
Time	Time
Time of Day	Time_Of_Day

Datentyp in WinCC flexible	Datentyp in WinCC
Timer	siehe unten
Word	Word

Besonderheiten bei einigen Datentypen

Bei der Migration von externen Variablen, die Datentypen einer SIMATIC S7-300/400 Steuerung enthalten, müssen Sie einige Besonderheiten beachten.

Abbildung des S7-Datentyps "Char"

Der S7-Datentyp "Char" ist gemäß Spezifikation ein Datentyp zur Abbildung von Zeichen. Da dieser Datentyp aber oft zum Lesen und Schreiben von Zahlenwerten verwendet wurde, wird er in WinCC abgebildet auf den S7-Datentyp "Byte". Wenn dieser Fall bei der Migration eintritt, wird im Ausgabefenster eine Meldung ausgegeben.

Wenn der S7-Datentyp "Char" für Zahlenwerte verwendet wird und an der Verwendungsstelle negative Zahlen projektiert wurden, führt dies bei der Abbildung auf den S7-Datentyp "Byte" zu einem Fehler. Der S7-Datentyp "Byte" kann keine negativen Zahlen abbilden. Um den Fehler zu beheben, müssen Sie die Projektierung entsprechend anpassen. Verwenden Sie für die Verarbeitung von positiven und negativen Zahlenwerten einen vorzeichenbehafteten Datentyp, z. B. den Datentyp "Int".

Wenn der S7-Datentyp "Char" doch zur Abbildung von Zeichen genutzt wird, müssen Sie nach der Migration die Projektierung ändern. Zur Darstellung von Zeichen benutzen Sie den Datentyp "String".

Bei der Migration eines integrierten Projekts wird der Datentyp "Char" in WinCC ebenfalls auf den Datentyp "Byte" migriert. Bei einer verbundenen PLC-Variablen wird der Datentyp "Char" allerdings auf "Char" belassen. Durch die Änderung des Datentyps der HMI-Variable wird die symbolische Adressierung der betreffenden Variablen nicht migriert. Die Variablen sind nach der Migration durch absolute Adressen miteinander verbunden und funktionieren weiterhin. Wenn Sie die symbolische Adressierung wieder herstellen wollen, müssen Sie die Projektierung nach der Migration entsprechend ändern.

Abbildung eines Arrays des S7-Datentyps "Char"

Ein Array vom S7-Datentyp "Char" wird bei der Migration auf ein Array des Datentyps "Byte" abgebildet.

Wenn ein Array vom S7-Datentyp "Char" für Zahlenwerte verwendet wird und an der Verwendungsstelle negative Zahlen projektiert wurden, führt dies bei der Abbildung auf ein Array vom S7-Datentyp "Byte" zu einem Fehler. Der S7-Datentyp "Byte" kann keine negativen Zahlen abbilden. Um den Fehler zu beheben, müssen Sie die Projektierung entsprechend anpassen. Verwenden Sie für die Verarbeitung von positiven und negativen Zahlenwerten einen vorzeichenbehafteten Datentyp, z. B. den Datentyp "Int".

Abbildung des S7-Datentyps "Counter"

Eine externe Variable mit dem S7-Datentyp "Counter" mit Counter-Adresse wird auf den S7-Datentyp "Counter" abgebildet. Die Adresse wird beibehalten.

Wenn eine externe Variable mit dem S7-Datentyp "Counter" einen Datenbaustein oder einer Merker-Adresse adressiert, wird auf den S7-Datentyp "Word" abgebildet. Die Adresse wird beibehalten. Die Migration setzt die Codierung auf "SimaticBCDCounter".

Der S7-Datentyp "Counter" hat einen Wertebereich von 0-999. Bei der Versorgung über den S7-Datentyp "Word" ist eine Überschreitung des Wertebereichs auf der Steuerungsseite möglich. Stellen Sie sicher, dass der Wertebereich eingehalten wird.

Beispiel:

WinCC flexible

Variable	S7-Datentyp	Adresse	Kommentar
Counter_Actual_Value	Counter	C10	BCD codierter Zählerwert
Counter_Setpoint_Value	Counter	DB10.DBW200	BCD codierter Zählerwert
Counter_Setpoint_Value#2	Counter	MW20	BCD codierter Zählerwert

WinCC

Variable	S7-Datentyp	Adresse	Codierung	Kommentar
Counter_Actual_Value	Counter	%C10	<Standard>	BCD codierter Zählerwert
Counter_Setpoint_Value	Word	%DB10.%DBW200	SimaticBCDCounter	BCD codierter Zählerwert
Counter_Setpoint_Value#2	Word	%MW20	SimaticBCDCounter	BCD codierter Zählerwert

Abbildung des Datentyps "StringChar"

In WinCC gibt es keinen entsprechenden Datentyp, auf den der Datentyp "StringChar" abgebildet werden kann. Die Abbildung in WinCC ist abhängig von der Eigenschaft "Länge" des S7-Datentyps.

Eine Variable vom Datentyp "StringChar" mit der Eigenschaft "Länge" > 1 wird auf ein Array vom S7-Datentyp "Char" migriert. Die Länge des Arrays entspricht der Länge des ursprünglich projektierten Datentyps "StringChar".

Wenn die Eigenschaft "Länge" = 1 ist, wird der Datentyp in WinCC auf ein Array vom S7-Datentyp "Char" mit der Länge = 1 migriert. Der Ausdruck für ein Array mit einem Element lautet "Array[0 ..0] of Char".

Abbildung des S7-Datentyps "Timer"

Eine externe Variable mit dem S7-Datentyp "Timer" mit Timer-Adresse wird auf den S7-Datentyp "Timer" abgebildet. Die Adresse wird beibehalten.

Wenn eine externe Variable mit dem S7-Datentyp "Timer" einen Datenbaustein oder eine Merker-Adresse adressiert, wird auf den S7-Datentyp "S5 Time" abgebildet. Die Adresse wird beibehalten.

Beispiel:

WinCC flexible

Variable	S7-Datentyp	Adresse	Kommentar
Timer_Actual_Value	Timer	T10	BCD codierter Timer-Wert
Timer_Setpoint_Value	Timer	DB10.DBW200	BCD codierter Timer-Wert
Timer_Setpoint_Value#2	Timer	MW20	BCD codierter Timer-Wert

WinCC

Variable	S7-Datentyp	Adresse	Kommentar
Timer_Actual_Value	Timer	%T10	BCD codierter Timer-Wert
Timer_Setpoint_Value	S5Time	%DB10.%DBW200	BCD codierter Timer-Wert
Timer_Setpoint_Value#2	S5Time	%MW20	BCD codierter Timer-Wert

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

Migration der Datentypen von Telemecanique Uni-Telway (WinCC flexible)

Migration der Datentypen Telemecanique Uni-Telway

Der Kommunikationstreiber Telemecanique Uni-Telway wird von WinCC nicht unterstützt, die Datentypen werden auf die internen Datentypen von WinCC abgebildet. Die Datentypen des Kommunikationstreibers Telemecanique Uni-Telway werden bei der Migration nach WinCC folgendermaßen abgebildet:

Datentyp in WinCC flexible	Datentyp in WinCC
+/-Int	Int
+/-Long	DInt
ASCII	WString
Bool	Bool
Float	Real
Int	UInt
Long	UDInt

Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 80)

2.8.2 WinCC V7 Projekte migrieren

2.8.2.1 Grundlagen (WinCC V7)

Migration (WinCC V7)

Unterstützte Versionen

Projekte aus WinCC V7.4 SP1 können Sie nach WinCC V15.1 migrieren.

Projekte aus früheren WinCC-Versionen können Sie nicht direkt migrieren. Wenn Sie solche Projekte weiter verwenden wollen, müssen Sie diese erst nach WinCC V7.4 SP1 migrieren.

1. Öffnen Sie das Projekt in WinCC V7.4 SP1
2. Bestätigen Sie die Abfrage zur Migration oder konvertieren Sie im WinCC Explorer die Projektdaten über den Menübefehl "Extras". Alle Objekte müssen auf die aktuelle Version konvertiert werden.

Uneingeschränkt migrierbare Projekte

Folgende Projektarten werden unterstützt:

- Einzelplatz-Projekte
- Mehrplatz-Projekte

Projekte aus WinCC V7 werden zu WinCC Runtime Professional-Geräten migriert.

Migration von Projekten mit redundanten Systemen

Wenn in WinCC V7 ein redundantes System projektiert ist, wird die Projektierung der redundanten Partner migriert.

Nicht migrierbare Projekte

WinCC V7-Projekte mit folgenden Geräten oder Funktionen können nicht migriert werden:

- Multi-Client
- CAS
- Connectivity Station
- Maintenance Station

Die Migration wird abgebrochen und es wird eine Meldung ausgegeben.

Siehe auch

Grundlagen zur Migration (WinCC V7) (Seite 96)

Projekte aus WinCC V7 migrieren (WinCC V7) (Seite 98)

Simulation und Runtime starten (WinCC V7) (Seite 100)
Änderung des Projektnamens durch die Migration (WinCC V7) (Seite 101)
Objektunterstützung bei der Migration (WinCC V7) (Seite 103)
Migration von Verbindungen (WinCC V7) (Seite 106)
Migration von Variablen (WinCC V7) (Seite 109)
Migration von Anwenderobjekten (WinCC V7) (Seite 115)
Migration von Meldungen und Meldegruppen (WinCC V7) (Seite 122)
Migration von Funktionen und Aktionen (WinCC V7) (Seite 124)
Grundlagen zur Migration von Protokollen (WinCC V7) (Seite 128)
Kopieren der Runtime-Daten (WinCC V7) (Seite 135)
Migration der Datentypen (WinCC V7) (Seite 135)
Filterung der Projektsprachen bei der Migration (Seite 102)

Grundlagen zur Migration (WinCC V7)

Einleitung

Bei der Migration werden die Projektdaten aus einem WinCC V7-Projekt in das neue Datenformat von WinCC umgesetzt. Dabei wird nicht geprüft, ob die Daten in dem zu migrierenden Projekt konsistent sind. Wenn in einem Quellprojekt Fehler enthalten sind, dann werden diese bei der Migration nicht behoben. Daher sollte das zu migrierende Projekt vor der Migration fehlerfrei sein. Beachten Sie bei der Migration den Umfang eines Projekts. Für die Migration gelten die Leistungsmerkmale von WinCC. Nähere Hinweise hierzu finden Sie in der Online-Hilfe im Kapitel "Prozesse visualisieren > Leistungsmerkmale > Leistungsmerkmale der Geräte > WinCC Runtime Professional (RT Professional)".

Hinweis

Verwendung der Schriftzeichen "ss" und "ß" im TIA Portal

Die Namen der HMI-Objekte (Variablen, Bilder, Rezepturen, Verbindungen, Textlisten usw.) müssen im TIA Portal eindeutig sein. Zwischen den Zeichen "ss" und "ß" wird nicht unterschieden. Deshalb ist es nicht möglich, in ansonsten identischen Namen der HMI-Objekte die beiden Zeichenabfolgen zur Unterscheidung zu verwenden.

Voraussetzungen für die Migration

Damit die Migration durchgeführt werden kann, sind einige grundlegende Voraussetzungen zu beachten.

- Das zu migrierende Projekt muss in WinCC V7.4 SP1 gespeichert worden sein
- Das zu migrierende Projekt darf nicht geöffnet sein, weder in WinCC V7 noch in Runtime.
- Auf dem PC für die Migration muss genügend freier Speicherplatz auf der Festplatte vorhanden sein. Während der Migration kann der Bedarf an notwendigem Festplattenspeicher die 10-fache Größe des WinCC V7-Projekts ohne Runtime-Daten betragen.

Abbruch der Migration

In folgenden Fällen wird die Migration abgebrochen:

- Wenn das zu migrierende Projekt im Engineering-System oder in Runtime geöffnet ist.
- Wenn nicht genügend freier Speicherplatz auf der Festplatte zur Verfügung steht, um eine Kopie für die Migration des Projekts anzulegen.
- Wenn die Migration wegen Problemen mit dem installierten SQL-Server die Projekt-Datenbank nicht ansprechen kann.
- Wenn die Migration wegen fehlender Benutzerberechtigung die Projekt-Datenbank nicht ansprechen kann.
- Wenn das Projekt mit einer von der Migration nicht unterstützten Version erstellt wurde.
- Wenn im Projekt ein Control verwendet wird, das nicht im Betriebssystem registriert ist.

Projekt mit dem Migrations-Tool vorbereiten

Die Migration eines WinCC V7-Projekts müssen Sie nicht auf dem PC durchführen, auf dem WinCC installiert ist.

Auf einem PC, auf dem WinCC V7.4 SP1 installiert ist, können Sie die Migration vorbereiten, indem Sie das Migrations-Tool verwenden. Das Migrations-Tool exportiert die Engineering-Daten aus dem WinCC V7-Projekt und speichert die Daten im Format "*.am*".

Für die eigentliche Migration kopieren Sie die vom Migrations-Tool erzeugte Datei auf einen PC, auf dem das TIA Portal installiert ist.

Nähere Hinweise zum Migrations-Tool finden Sie im Kapitel "Vorbereitung von Projekten mit dem Migrations-Tool".

Siehe auch

Migration (WinCC V7) (Seite 95)

Projekte aus WinCC V7 migrieren (WinCC V7)

Einleitung

Bei der Migration eines Projekts werden Daten aus einem WinCC V7 Projekt in ein neues Projekt für WinCC übernommen. Deshalb wird für die Projektmigration automatisch ein neues Projekt angelegt. Sie können nicht in ein bestehendes Projekt migrieren.

Die Migration lässt sich sowohl in der Portalansicht als auch in der Projektansicht aufrufen.

Voraussetzung

- Ein Projekt aus WinCC V7.4 SP1 ist vorhanden.
- Das Projekt ist nicht in WinCC V7 geöffnet.

Vorgehen

Ein Projekt migrieren Sie in der Portalansicht wie folgt:

1. Wählen Sie die Aktion "Start > Projekt migrieren".

2. Navigieren Sie im Feld "Quellpfad" zu dem Projekt, das Sie migrieren wollen.
3. Selektieren Sie die WinCC V7-Projektdatei "*.mcp".
4. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.

5. Um alle Runtime-Daten des Projekts zu migrieren, aktivieren Sie "Daten von WinCC Runtime Professional kopieren".
   Diese Runtime-Daten sind erst nach dem Laden auf dem Ziel-Gerät sichtbar.
6. Klicken Sie auf "Migrieren".
   Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
   - Die Projektansicht wird geöffnet.
   - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
   - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
   - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
   - Nach Abschluss der Migration wird das Projekt gespeichert und eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte PC-System ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten, z. B. Bilder, Meldungen und Variablen. In einem Mehrplatz-Projekt wird für jeden Client und für den Server ein Gerät in der Projektnavigation von WinCC angelegt.

Wenn das Projekt erfolgreich migriert wurde, wird ein Hinweis mit einer Erfolgsmeldung angezeigt.

Migrationsprotokoll zu einem späteren Zeitpunkt öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Um das Protokoll zu einem späteren Zeitpunkt anzusehen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten > Protokolle"..
2. Doppelklicken Sie auf das Migrationsprotokoll. Das Protokoll wird geöffnet.

Hinweis

Meldungen als Baumstruktur

Ab TIA Portal V13.0 sind Meldungen im Migrationsprotokoll in eine Baumstruktur einsortiert. Somit sind alle Meldungen, die zu einem bestimmten Subsystem gehören, in einem Ordner abgelegt. Um den aktuell ausgewählten Ordner und alle Unterordner in einem einzigen Vorgang aufzuklappen, drücken Sie die Tastenkombination <NUM + x(Multiplikation)>.

Siehe auch

Migration (WinCC V7) (Seite 95)

Projekt anpassen (WinCC V7)

Ursachen

Wenn Sie ein Projekt migriert haben, müssen Sie es anpassen. Die erforderliche Nachprojektierung wird verursacht durch:

- Meldungen, die während der Migration aufgetreten sind und im Migrationsprotokoll eingetragen wurden.
- Daten, die in dem zu migrierenden Projekt nicht konsistent oder ungültig waren. WinCC im TIA Portal verfügt über strengere Kriterien zur Überprüfung von z. B. der Gültigkeit von Variablenadressen oder der Länge von Objektnamen. Durch die Migration werden ungültige Angaben nicht korrigiert.
- Ordner und Dateien, die Sie im WinCC V7 Projektverzeichnis angelegt haben, werden bei der Migration nicht in das neue Projektverzeichnis kopiert. Skripte, die auf solche Verzeichnisse und Dateien zugreifen, müssen Sie nach der Migration entsprechend anpassen.

Kriterien für Meldungen während der Migration

Während der Migration werden Meldungen im Migrationsprotokoll ausgegeben:

- Wenn durch die Migration eine Eigenschaft geändert wird.
- Wenn eine Eigenschaft nach der Migration den Zustand "ungültig" erhält, weil z. B. der Startwert einer Variablen außerhalb des Wertebereichs liegt, wird keine Meldung im Migrationsprotokoll ausgegeben. Den Zustand "ungültig" erkennen Sie im Engineering System z. B. an farbig hinterlegten Eingabefeldern. Wenn Sie ein Projekt übersetzen, werden für ungültige Angaben im Inspektorfenster Meldungen ausgegeben.

Projekt anpassen

1. Bearbeiten Sie die Meldungen im Migrationsprotokoll.
2. Klicken Sie in der Projektnavigation auf das migrierte Projekt.
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software".
4. Bearbeiten Sie die Fehlermeldungen im Inspektorfenster.

Simulation und Runtime starten (WinCC V7)

Simulation starten

Nach dem fehlerfreien Übersetzen des Projekts können sie auf einem PC mit installiertem Engineering System die Runtime simulieren. Sie können so die Funktionsfähigkeit der notwendigen Anpassungen im Projekt nach der Migration prüfen, ohne einen Download auf das Zielsystem durchführen zu müssen. Wählen Sie dazu den Menübefehl "Online > Simulation > Starten".

Laden des Projekts auf einen PC mit WinCC Runtime Professional

WinCC bietet darüber hinaus die Möglichkeit, das Projekt auf einen PC zu übertragen, auf dem nur WinCC Runtime Professional installiert ist. Sie müssen dazu das Projekt auf den betreffenden Zielcomputer laden. In einem WinCC V7-Projekt sind keine Informationen zum Laden auf einen anderen Zielcomputer verfügbar. Die Migration kann daher keine Daten für die Kommunikation mit einem anderen Zielcomputer migrieren. Um ein migriertes Projekt auf einen anderen Zielcomputer zu laden, müssen Sie in WinCC die erforderlichen Einstellungen projektieren.

Siehe auch

Migration (WinCC V7) (Seite 95)

Änderung des Projektnamens durch die Migration (WinCC V7)

Einleitung

Wenn Sie ein WinCC V7-Projekt migriert haben, wird das migrierte Projekt in der Projektnavigation von WinCC als neues Gerät mit dem Namen des WinCC V7-Projekts angezeigt.

Wenn Sie das migrierte Projekt übersetzen, erzeugt WinCC für das Runtime-Projekt und den enthaltenden Ordner einen neuen Namen. Dadurch wird die unter Windows erlaubte Pfadlänge von 256 Zeichen nicht überschritten. Das Runtime-Projekt hat damit einen anderen Namen als das Gerät in der Projektnavigation.

Runtime auf dem Zielsystem starten

Wenn Sie Runtime auf dem Projektierungs-PC simulieren, müssen Sie den neuen Namen nicht beachten. Öffnen Sie das Projekt in WinCC und starten Sie die Simulation über den Menübefehl "Online > Simulation > Starten".

Wenn Sie die Runtime auf einem PC, auf dem kein WinCC Engineering System installiert ist, starten wollen, gelten folgende Bedingungen:

- Für den ersten Ladevorgang müssen Sie den Menübefehl "Erweitertes Laden in Gerät" verwenden.
- WinCC Runtime legt bei der Installation das Freigabeverzeichnis "WinCCProjects" an. Im Dialog "Erweitertes Laden" tragen Sie im Feld "Zielpfad" den Pfad zum Freigabeverzeichnis auf dem Zielgerät ein. Alternativ navigieren Sie mithilfe der Navigationsschaltfläche zum Freigabeverzeichnis.
- Wenn Sie anschließend das Projekt laden, wird es im Freigabeverzeichnis "WinCCProjects" gespeichert.
- Wenn Sie das Projekt starten wollen, benutzen Sie das "WinCC StartTool".

Skripte anpassen

Wenn Sie ein ausführbares Programm erstellt haben, das ein migriertes Projekt über den Namen referenziert, müssen Sie dieses Skript entsprechend anpassen. Der neue Projektname ist die Zahlenfolge innerhalb der geschweiften Klammern in der Pfadangabe der oben beschriebenen MCX-Datei.

Siehe auch

Migration (WinCC V7) (Seite 95)

Filterung der Projektsprachen bei der Migration

Einleitung

Für die optimale Migration der Projekte werden vom bestehenden WinCC V7-Projekt automatisch die Projektsprachen herausgefiltert, die nicht in der Textbibliothek aufgeführt sind und für die keine vom Default-Wert abweichenden Texte gefunden werden. Diese Projektsprachen werden nicht migriert.

Kriterien für Filterung während der Migration

Die Filterung bezieht sich nur auf Projektsprachen, die nicht explizit in der Textbibliothek enthalten sind.

Wenn eine Projektsprache nicht in der Textbibliothek enthalten ist und die Texte ausschließlich die folgenden Textstrings enthalten, wird die Projektsprache nicht migriert:

- Textstring mit sechs Fragezeichen
- Leerer Textstring
- Textstring mit automatisch generiertem Inhalt "Text"
- Textstring für Optionskästchen oder Optionsfelder, der mit sechs Fragezeichen oder automatisch generiertem Inhalt "Text" anfängt und mit einer Zahl endet, z. B. "??????5" oder "Text5"

Hinweis

Projektsprachen, die an Controls verwendet werden, beeinflussen nicht das Filterverhalten.

Hinweis

Wenn eine Projektsprache nicht migriert wird, wird eine Meldung im Migrationsprotokoll ausgegeben.

Siehe auch

Migration (WinCC V7) (Seite 95)

Migration von Texten in Spanisch (WinCC V7)

Texte in Spanisch werden in Abhängigkeit von ihrer Sortierung unterschiedlich migriert.

Migration von Texten mit traditioneller und internationaler Sortierung

Wenn ein WinCC V7-Projekt Texte in Spanisch (traditionelle Sortierung) enthält, werden diese Texte als Spanisch (Spain) migriert.

Wenn ein WinCC V7-Projekt Texte in Spanisch (internationale Sortierung) enthält, werden diese Texte als Spanisch (Spain) migriert.

Wenn ein WinCC V7-Projekt sowohl Texte in Spanisch (internationale Sortierung) als auch Spanisch (traditionelle Sortierung) enthält, werden nur die Texte aus Spanisch (traditionelle Sortierung) als Spanisch (Spain) migriert. Die Texte aus Spanisch (internationale Sortierung) bleiben dann unberücksichtigt.

2.8.2.2 Engineering-Daten migrieren (WinCC V7)

Objektunterstützung (WinCC V7)

Objektunterstützung bei der Migration (WinCC V7)

Einleitung

Bei der Migration von Projekten aus WinCC V7 werden alle Objekttypen und Funktionen vollständig migriert, die in der neuen Projektumgebung zur Verfügung stehen und abgebildet werden können.

Unterstützte Objekttypen

Folgende Objekttypen werden von der Migration unterstützt:

- Bilder
- Anzeige- und Bedienobjekte
- Bildbausteine
- Variablen
- Datentypen
- Strukturen
- Verbindungen
- Zyklen
- Meldungen
- Meldeklassen
- Meldegruppen

- Meldearten
- Anwenderarchive
- Meldearchive
- Variablenarchive
- Protokolle
- Druckaufträge
- Layouts
- Projekt-Funktionen
- Standard-Funktionen
- Aktionen
- Benutzerverwaltung
- Texte
- Projektsprachen
- Runtime-Sprachen
- Runtime-Schriften
- ODK-Funktionen des RT

Einschränkungen und bedingt unterstützte Objekte sind in den nachfolgenden Kapiteln beschrieben.

Nicht unterstützte Objekttypen

Folgende Objekttypen werden von der Migration nicht unterstützt:

- 3D-Balken
- Bildhierarchie
- Erweiterte Analoganzeige
- Erweiterte Zustandsanzeige
- Eingefügtes Layout
- Globale Bibliotheken
- Projektbibliothek
- Hörmelder
- Sammelanzeige
- Server-Server-Kommunikation
- XAML-Controls, z. B. WPF-Controls
- ODK-Funktionen des CS
- Default-Bilder der PCS7 Standard/Advanced Bibliotheken

Nicht unterstützte Sprachen

In WinCC V7 stehen über 200 Sprachen und Dialekte für die Projektierung zur Verfügung. Im TIA Portal werden ca. 100 Sprachen und Dialekte unterstützt, die im Editor "Projektsprachen" unter "Sprachen&Ressourcen" aufgelistet sind. Alle Texte eines WinCC V7-Projekts in den im TIA Portal nicht unterstützten Sprachen gehen bei der Migration verloren. Dazu werden im Migrationsprotokoll Meldungen ausgegeben.

Die programmierten Runtime-Sprachumschaltungen in den nicht unterstützten Sprachen werden im TIA Portal-Projekt ungültig. Vor der Inbetriebnahme des Projekts müssen Sie diese Sprachumschaltungen entfernen. Wenn ein WinCC V7-Projekt nur Texte in nicht unterstützten Sprachen enthält, werden keine Texte migriert. Im Editor "Projektsprachen" unter "Sprachen&Ressourcen" wird in diesem Fall als Ersatz die Standardsprache "Englisch (USA)" eingestellt.

Nicht erlaubte Zeichen

WinCC unterstützt für Objektnamen nicht exakt den gleichen Zeichenvorrat wie WinCC V7. Zum Beispiel sind in WinCC für Variablen folgende Zeichen nicht erlaubt: ? " %

Durch die Migration werden die Objektnamen nicht verändert, auch wenn der Name ein Zeichen enthält, das in WinCC nicht zulässig ist. Nicht erlaubte Zeichen führen dazu, dass der Objektname ungültig ist und beim Übersetzen eine Meldung ausgegeben wird. In WinCC können Sie Objektnamen zentral ändern. Es werden immer auch alle Verwendungsstellen mitgeändert.

Siehe auch

Geänderte Objekte (WinCC V7) (Seite 105)

Migration (WinCC V7) (Seite 95)

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Geänderte Objekte (WinCC V7)

Einleitung

Einige Objekte, die in WinCC V7 verfügbar sind, werden von WinCC nicht mehr unterstützt. Die Migration bildet diese Objekte auf die in WinCC vorhandenen Objekte ab. Da nicht alle Eigenschaften abgebildet werden können, sind teilweise Nacharbeiten erforderlich. Im Migrationsprotokoll werden geänderte und nicht migrierte Eigenschaften ausgegeben.

Geänderte Objekte

Folgende Objekte werden von der Migration auf andere Objekte abgebildet oder haben teilweise andere Eigenschaften:

- OLE-Objekt
- Push Button Control
- Textliste

- Zustandsanzeige
- WinCC Controls vor WinCC V7
- SVG-Grafiken
- Anwenderobjekte
- Bildbausteine
- Objektgruppen
- Dynamisierungen

Nähere Hinweise hierzu finden Sie im Kapitel Bilder und Bildobjekte (WinCC V7) (Seite 110).

Siehe auch

Objektunterstützung bei der Migration (WinCC V7) (Seite 103)

Bilder und Bildobjekte (WinCC V7) (Seite 110)

Verbindungen (WinCC V7)

Migration von Verbindungen (WinCC V7)

Einleitung

Bei der Migration von Verbindungen sind folgende Fälle zu unterscheiden:

- Kommunikationstreiber wird in WinCC unterstützt.
- Kommunikationstreiber wird in WinCC nicht unterstützt.

Wie die einzelnen Kommunikationstreiber migriert werden, entnehmen Sie der nachfolgenden Beschreibung.

Migration mit unterstütztem Kommunikationstreiber

Wenn WinCC den verwendeten Kommunikationstreiber unterstützt, wird die Verbindung mit diesem Kommunikationstreiber migriert. Folgende Kommunikationstreiber werden unterstützt:

WinCC V7	WinCC
SIMATIC S7 PROTOCOL SUITE	SIMATIC S7-300/400
SIMATIC S7-1200	SIMATIC S7-1200
SIMATIC S7-1500	SIMATIC S7-1500
ALLEN BRADLEY - ETHERNET IP	Allen Bradley EtherNet/IP
MITSUBISHI ETHERNET	Mitsubishi MC TCP/IP
MODBUS TCPIP	Modicon MODBUS TCP/IP
OPC	OPC
OPC UA DA	OPC UA DA

Die Kanal-Units "Industrial Ethernet (I + II)" der S7 Protocol Suite werden migriert zur Kanal-Unit Industrial Ethernet des Treibers SIMATIC S7-300/400.

Die Kanal-Units "PROFIBUS (I + II)" der S7 Protocol Suite werden migriert zur Kanal-Unit PROFIBUS des Treibers SIMATIC S7-300/400.

Die Kanal-Unit "Named Connection" wird nicht migriert.

Migration eines nicht unterstützten Kommunikationstreibers

In WinCC werden nicht alle Kommunikationstreiber unterstützt, die in WinCC V7 zur Verfügung stehen. Wenn ein Kommunikationstreiber nicht unterstützt wird, dann wird die projektierte Verbindung nicht migriert.

Wenn Sie ein WinCC V7-Projekt migrieren, das eine nicht unterstützte Verbindung enthält, werden alle externen Variablen dieser Verbindung auf interne Variablen in einer Variablen-Tabelle abgebildet. Die Adresse der Steuerung wird als Kommentar zur Variablen abgelegt Für jede betroffene Verbindung erhalten Sie eine Variablen-Tabelle und einen Eintrag im Migrationsprotokoll.

Nach der Migration und dem Projektieren einer neuen Verbindung müssen Sie die konvertierten Variablen wieder mit der Steuerung verbinden.

Wenn der Treiber nicht migriert wird, erhalten Sie eine entsprechende Warnmeldung.

Folgende Kommunikationstreiber werden von WinCC nicht unterstützt:

WinCC V7	WinCC
Profibus DP	nicht unterstützt
Profibus FMS	nicht unterstützt
SIMATIC TI Serial	nicht unterstützt
SIMATIC 505 TCP IP	nicht unterstützt
SIMATIC S5 Ethernet Layer 4	nicht unterstützt
SIMATIC S5 Ethernet TF	nicht unterstützt
SIMATIC S5 Profibus FDL	nicht unterstützt
SIMATIC S5 Programmers Port AS 511	nicht unterstützt
SIMATIC S5 Serial 3964R	nicht unterstützt
SIMATIC TI Ethernet Layer 4	nicht unterstützt
SIMATIC TI Serial	nicht unterstützt
System Info	nicht unterstützt
Windows DDE	nicht unterstützt

Bestehende Funktionalitäten aus Windows DDE können Sie in WinCC mithilfe von OPC umsetzen. Die Umsetzung müssen Sie selbst vornehmen.

Siehe auch

Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC V7) (Seite 108)

Migration (WinCC V7) (Seite 95)

Migration von Variablen (WinCC V7) (Seite 109)

Anpassen der Projektierung bei nicht migrierter Verbindung (WinCC V7)

Einleitung

Wenn eine Verbindung nicht migriert wird, haben Sie folgende Möglichkeiten:

- Ändern der Projektierung in WinCC V7 vor der Migration
- Ändern der Projektierung nach der Migration

Ändern einer Verbindung vor der Migration

Wenn der für die Verbindung gewählte Kommunikationstreiber von der Migration nicht unterstützt wird, müssen Sie einen Treiber wählen, der von der Migration unterstützt wird.

Falls kein passender Treiber verfügbar ist, müssen Sie geeignete Komponenten für die Kommunikation verwenden. Passen Sie anschließend die Projektierung in WinCC V7 an. Nach erfolgreicher Anpassung migrieren Sie das Projekt. Die Verbindung wird dann ebenfalls erfolgreich migriert.

Ändern einer Verbindung nach der Migration

Wenn Sie ein WinCC V7-Projekt migrieren, das eine nicht unterstützte Verbindung enthält, werden alle externen Variablen dieser Verbindung auf interne Variablen in einer Variablen-Tabelle abgebildet. Die Adresse der Steuerung wird als Kommentar zur Variablen abgelegt Für jede betroffene Verbindung erhalten Sie eine Variablen-Tabelle und einen Eintrag im Migrationsprotokoll.

Nach der Migration müssen Sie eine neue Verbindung projektieren. Für diese Verbindung wählen Sie einen Kommunikationstreiber, der den verwendeten Kommunikationspartner unterstützt. Falls für den Kommunikationspartner kein passender Treiber verfügbar ist, müssen Sie geeignete Komponenten für die Kommunikation verwenden.

Wenn Sie die neue Verbindung projektiert haben, müssen Sie alle Variablen neu verbinden, die von der Migration auf interne Variablen abgebildet wurden. Um die Nachprojektierung zu vereinfachen, öffnen Sie das Migrationsprotokoll. Projektieren Sie die Variablen entsprechend dem Protokolleintrag nach.

Siehe auch

Projekte aus WinCC V7 migrieren (WinCC V7) (Seite 98)

Migration von Verbindungen (WinCC V7) (Seite 106)

Migration von Variablen (WinCC V7)

Einleitung

Bei der Migration von Variablen müssen Sie einige Besonderheiten beachten. Folgende Aspekte sind zu unterscheiden:

- Migration der Datentypen von Variablen
- Migration interner Variablen
- Migration externer Variablen

Migration der Datentypen

WinCC verfügt über die gleichen Datentypen, verwendet aber teilweise andere Datentypnamen als WinCC V7. Bei der Migration von externen Variablen, die Datentypen einer SIMATIC S7-Steuerung enthalten, sind einige Besonderheiten zu beachten. Nähere Hinweise hierzu finden Sie im Kapitel Migration der Datentypen (WinCC V7) (Seite 135).

Migration interner Variablen

Interne Variablen werden immer vollständig migriert. Lediglich der Datentypname kann sich durch die Migration ändern. Nähere Hinweise finden Sie im Kapitel Migration der Datentypen (WinCC V7) (Seite 135)

Migration externer Variablen

Wenn die durch externe Variablen verwendete Verbindung migriert wird, dann werden auch die externen Variablen migriert. Die Migration externer Variablen ist davon abhängig, ob der verwendete Kommunikationstreiber von der Migration unterstützt wird. Nähere Hinweise hierzu finden Sie im Kapitel Migration von Verbindungen (WinCC V7) (Seite 106).

Wenn die durch externe Variablen verwendete Verbindung nicht migriert wird, dann werden auch die externen Variablen nicht migriert. Die externen Variablen werden dann auf interne Variablen abgebildet. Nach der Migration müssen Sie eine neue Verbindung projektieren und die Variablen wieder mit den Variablen der Steuerung verbinden.

Siehe auch

Migration der Datentypen (WinCC V7) (Seite 135)

Migration von Verbindungen (WinCC V7) (Seite 106)

Migration von Strukturen (WinCC V7) (Seite 110)

Migration (WinCC V7) (Seite 95)

Migration von Strukturen (WinCC V7)

Migration von Strukturen

In WinCC V7 werden strukturierte Variablen mithilfe von Strukturtypen und Strukturelementen realisiert. Die Strukturtypen sind in WinCC V7 für das Gerät gültig, in dem sie angelegt werden. In WinCC werden Strukturtypen in der Projektbibliothek angelegt und sind für alle Geräte eines Projekts gültig. Der Aufbau der Strukturtypen in WinCC und WinCC V7 ist gleich.

Siehe auch

Migration von Variablen (WinCC V7) (Seite 109)

Bilder und Bildobjekte (WinCC V7)

Migration von Bildern und Bildobjekten (WinCC V7)

Einleitung

Die Mehrzahl der in WinCC V7 verfügbaren Bildobjekte wird durch die Migration ohne Einschränkungen nach WinCC übernommen.

Bei einigen Bildobjekten, z. B. Textliste und Zustandsanzeige, stehen in WinCC verbesserte Möglichkeiten zur Verfügung. Durch die Migration werden diese Objekte so umgesetzt, dass keine Nacharbeiten erforderlich sind.

Einzelne Bildobjekte, wie das WinCC Push Button Control, werden von WinCC nicht mehr unterstützt. Durch die Migration werden diese Objekte so weit wie möglich auf die in WinCC verfügbaren Objekte abgebildet.

Bei der Migration sind folgende Besonderheiten zu beachten:

In WinCC seit V7.3 neu eingeführte Eigenschaften

Die in WinCC seit V7.3 eingeführten neuen Eigenschaften werden nicht nach WinCC migriert. Wenn eine der neuen Eigenschaften dynamisiert ist, wird über die nicht migrierten dynamisierten Eigenschaften im Migrationsprotokoll eine Meldung ausgegeben. Im Gegensatz dazu wird über die nicht migrierten statischen Eigenschaften keine Meldung ausgegeben.

Tab-Reihenfolgen in Bildern mit Bildbausteinen oder Objektgruppen

In Bildern mit Bildbausteinen oder Objektgruppen ändern sich durch die Migration die Tab-Reihenfolgen sowohl im Bild als auch im Bildbaustein bzw. in der Objektgruppe.

Umbenennen von Ereignissen

Einige Ereignisse erhalten in WinCC neue Namen, z. B. heißt das Ereignis "Mausklick" in WinCC "Klicken". Durch die Migration wird diese Umbenennung vollständig umgesetzt. Nacharbeiten sind nicht erforderlich.

Migration von Triggern

Trigger aus WinCC V7 vom Typ Fenster-Zyklus werden migriert zu dem in WinCC gültigen Aktualisierungszyklus eines Bildes.

Migration einer Textliste

In WinCC V7 ermöglicht das Objekt "Textliste" die Ausgabe von Texten abhängig vom Wert einer Variablen. Die auszugebenden Texte mussten dabei bei jedem Objekt neu festgelegt werden. In WinCC steht eine von der Verwendungsstelle unabhängige Textliste zur Verfügung. Das Objekt "Symbolisches EA-Feld" dient der Anzeige von Texten aus einer Textliste.

Die in WinCC V7 verwendeten Texte werden in eine Textliste migriert und mit dem Symbolischen EA-Feld verbunden. Die verwendeten Texte werden als Projekttext abgelegt. Projekttexte finden Sie in der Projektnavigation unter "Sprachen & Ressourcen".

Migration einer Zustandsanzeige

In WinCC V7 ermöglicht das Objekt "Zustandsanzeige" die Ausgabe von Grafiken abhängig vom Wert einer Variablen. Die auszugebenden Grafiken mussten dabei bei jedem Objekt neu festgelegt werden. In WinCC steht eine von der Verwendungsstelle unabhängige Grafikliste zur Verfügung. Das Objekt "Grafisches EA-Feld" dient der Anzeige von Grafiken aus einer Grafikliste.

Die in der Zustandsanzeige verwendeten Bilder werden in eine Grafikliste migriert und mit dem Grafischen EA-Feld verbunden. Die verwendeten Bilder werden in der Grafiksammlung von WinCC abgelegt. Die Grafiksammlung finden Sie in der Projektnavigation unter "Sprachen & Ressourcen".

Migration eines Bildbausteins

In WinCC weisen Bildbausteine einige Unterschiede gegenüber den Bildbausteinen in WinCC V7 auf. Bei der Migration eines Bildbausteins sind daher folgende Besonderheiten zu beachten:

- In einem Bildbaustein kann eine Eigenschaft eines inneren Objekts nur auf jeweils eine Eigenschaft des Bildbausteins verschaltet werden. Wenn eine Eigenschaft eines inneren Objekts auf mehrere Eigenschaften des Bildbausteins verschaltet ist, wird nur die Verbindung zu der zuerst projektierten Eigenschaft migriert. Für die nicht migrierten Eigenschaften wird im Migrationsprotokoll eine Meldung ausgegeben.
- Wenn ein Ereignis eines inneren Objekts mehrfach an die Ereignisschnittstelle des Bildbausteins angebunden ist, dann hat nach der Migration nur das letzte Ereignis eine Verbindung zum inneren Objekt. Die übrigen Ereignisse der Ereignisschnittstelle werden ohne Verbindung zum inneren Objekt migriert. Wenn an diese Ereignisse Dynamisierungen projektiert sind, dann werden diese migriert. Die Anbindung der Dynamisierung an die inneren Objekte geht verloren.

- Wenn eine Objekteigenschaft innerhalb eines Bildbaustein-Typs dynamisiert ist und diese Eigenschaft gleichzeitig an die Ereignisschnittstelle des Bildbausteins geführt ist, wird bei der Migration die Verbindung zur Ereignisschnittstelle verworfen. Die innere Dynamisierung wird migriert. Die Eigenschaft an der Ereignisschnittstelle wird ebenfalls migriert, lediglich die Verbindung zu der Eigenschaft des inneren Objekts geht verloren.
  Beispiel:
  Ein Bildbaustein-Typ hat ein inneres Objekt "Kreis". Die Eigenschaft "Hintergrundfarbe" des Kreises ist dynamisiert mit einem inneren Ereignis, z. B. durch eine Bildbaustein-Variable. Die Eigenschaft "Hintergrundfarbe" des Kreises ist nach außen an die Ereignisschnittstelle des Bildbausteins geführt und wird in der Bildbaustein-Instanz zusätzlich von außen dynamisiert. Diese Projektierung ist in WinCC nicht mehr erlaubt. Bei der Migration wird die innere Dynamisierung migriert. Die äußere Dynamisierung und die Eigenschaft an der Ereignisschnittstelle des Bildbausteins werden ebenfalls migriert. Die Verbindung zwischen der äußeren Eigenschaft und der inneren Eigenschaft "Hintergrundfarbe" des Kreises wird aufgelöst.
- Wenn ein Bildbaustein-Typ ein EA-Feld enthält, kann dessen Eigenschaft "Ausgabeformat" nicht an die Ereignisschnittstelle des Bildbaustein-Typs angebunden werden.
- Bei Bildbausteinen in WinCC V7 können interne Variablen direkt mit der Schnittstelle verschaltet werden. Dies ist bei Bildbausteinen in WinCC nicht möglich. Um bei Bildbausteinen das gleiche Verhalten zu erzielen, wird durch die Migration ein unsichtbares EA-Feld als Hilfsobjekt erzeugt. An diesem EA-Feld werden pro interner Variable zwei VB-Funktionen generiert und mit einer Schnittstelleneigenschaft verschaltet. Über diese Schnittstelleneigenschaft erreichen Sie die interne Variable. Allerdings können nur maximal 23 mit der Schnittstelle verschaltete Variablen migriert werden.

Migration eines OLE-Objekts

Das OLE-Objekt wird in WinCC nicht unterstützt. Die Funktionen des OLE-Objekts werden durch die Migration auf eine Grafikanzeige abgebildet. Im Migrationsprotokoll wird eine Meldung ausgegeben.

Wenn das OLE-Objekt mit einer textorientierten Datei z. B einer Excel-Tabelle verbunden ist, wird durch die Migration ein Screenshot der Excel-Tabelle erzeugt und mit der Grafikanzeige verbunden.

Migration eines WinCC Push Button Control

Das WinCC Push Button Control wird in WinCC nicht unterstützt. Die Funktionen des Push Button Control werden durch die Migration auf eine Schaltfläche abgebildet.

Folgende Eigenschaften des Push Button Control sind an einer Schaltfläche nicht verfügbar und werden daher nicht migriert:

- Outline
- FontStrikeThru
- FocusRect
- AutoSize
- Transparent

Wenn Sie eine Dynamik an eine dieser Eigenschaften projektiert haben, geht die Dynamisierung bei der Migration verloren. Wenn eine Eigenschaft oder eine Dynamik des Objekts nicht migriert werden kann, erfolgt ein entsprechender Eintrag im Migrationsprotokoll.

Da eine Schaltfläche andere Ereignisse unterstützt, sind bei der Migration folgende Besonderheiten zu beachten:

- Das Ereignis "Click" des Push Button Control wird ohne Einschränkungen auf das Ereignis "Klicken" der Schaltfläche migriert.
- Die Ereignisse "KeyDown" und "KeyUp" des Push Button Control werden auf die Ereignisse "Taste auf Tastatur drücken" und "Taste auf Tastatur loslassen" migriert. Dabei können aber die Parameter nicht eindeutig abgebildet werden. Falls Sie diese Parameter in einem Skript verwenden, wird das Skript beim Übersetzen eine Meldung verursachen.
- Das Ereignis "MouseDown" des Push Button Control wird auf die beiden Ereignisse "Linke Maustaste drücken" und "Rechte Maustaste drücken" migriert. Das Ereignis "MouseUp" des Push Button Control wird auf die beiden Ereignisse "Linke Maustaste loslassen" und "Rechte Maustaste loslassen" migriert. Dabei können aber die Parameter nicht eindeutig abgebildet werden. Falls Sie diese Parameter in einem Skript verwenden, wird das Skript beim Übersetzen eine Meldung verursachen.
- Die Ereignisse "KeyPress" und "MouseMove" des Push Button Control können nicht migriert werden. Bei der Migration geht die Projektierung an den beiden Ereignissen verloren. Im Migrationsprotokoll wird eine Meldung ausgegeben.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC V7) (Seite 113)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC V7)

Einleitung

Aufgrund der Vereinheitlichung von Objekteigenschaften aus WinCC V7 und WinCC flexible sind im Verlauf der Migration an einigen Objekteigenschaften Änderungen erforderlich. Die Migration berechnet die Änderungen so, dass die Darstellung der Objekte nach der Migration gleich ist wie vor der Migration. Durch die Änderungen während der Migration ergeben sich für einige Objekteigenschaften andere Maßeinheiten und Werte in der Projektierung.

Migration der Schrifteinstellungen eines Objekts

In WinCC V7 und in WinCC werden für die Größe der verwendeten Schriften teilweise unterschiedliche Maßeinheiten verwendet. Bei der Migration wird die Schriftgröße umgerechnet, so dass die Darstellung der Schrift bei der Zoomstufe 100% gleich groß ist. Aufgrund der unterschiedlichen Maßeinheiten ändern sich nach der Migration die Zahlenwerte für die Schriftgrößen.

Beispiel:

Schriftart vor der Migration	Schriftart nach der Migration
Arial 10 Punkt	Arial 13 Pixel
Arial 16 Punkt	Arial 21 Pixel
Tahoma 10 Punkt	Tahoma 13 Pixel
Tahoma 16 Punkt	Tahoma 21 Pixel

Migration von Größenangaben eines Objekts.

An Objekten des Report Designer in WinCC V7 erfolgen Positionsangaben und Größenangaben entsprechend der Voreinstellung im Report Designer, z. B. in cm, inch oder Punkt. In WinCC werden Positionsangaben und Größenangaben in Pixel angegeben. Bei der Migration werden die Größenangaben so umgerechnet, dass die Darstellung der Objekte vor und nach der Migration identisch ist. Durch die Umrechnung ergeben sich nach der Migration andere Zahlenwerte für die Angaben in den Objekteigenschaften. Betroffene Objekteigenschaften sind z. B. Position X, Position Y, Breite und Höhe.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Bildern und Bildobjekten (WinCC V7) (Seite 110)

Migration von Objektgruppen (WinCC V7)

Objektgruppe mit Objekten in mehreren Ebenen

In WinCC V7 können die Objekte einer Objektgruppe in verschiedenen Ebenen des Bildes liegen und unterliegen der Ebenenverwaltung des Bildes. Die Objektgruppe selbst hat keine Ebenenverwaltung. In WinCC V7 ist es daher möglich, ein Objekt in einer Ebene zwischen zwei Objekten einer Objektgruppe zu platzieren. Die Z-Ordnung gilt für alle Objekte innerhalb des Bildes. Daher ist es in WinCC V7 möglich, ein Objekt in einer Objektgruppe unsichtbar zu machen, indem eine Ebene deaktiviert wird.

In WinCC liegt eine Objektgruppe und die darin enthaltenen Objekte in der gleichen Ebene des Bildes. In WinCC hat eine Objektgruppe eine eigene Ebenenverwaltung. Die Z-Ordnung der Objekte gilt in WinCC nur innerhalb der Objektgruppe. Wenn ein Objekt über eine Objektgruppe platziert wird und eine Ebene nach hinten verschoben wird, liegt dieses Objekt hinter der Objektgruppe und ist nicht mehr sichtbar. Das Objekt wird auch nicht in die Z-Ordnung der Objektgruppe aufgenommen. Daher ist es in WinCC nicht möglich, ein Objekt in einer Objektgruppe unsichtbar zu machen, indem eine Ebene deaktiviert wird. Wenn die Migration die Anordnung einer Objektgruppe nicht vollständig abbilden kann, wird im Migrationsprotokoll eine Meldung ausgegeben. Prüfen Sie nach der Migration, ob die Objektgruppe korrekt dargestellt wird.

Objektgruppe mit nicht unterstützten Objekten

Eine Objektgruppe besteht in WinCC aus mindestens 2 Objekten. Wenn WinCC ein Objekt einer Gruppe nicht unterstützt, wird die Gruppe trotzdem migriert. Das nicht unterstützte Objekt wird nicht migriert und fehlt nach der Migration in der Gruppe. Enthält die Objektgruppe vor der Migration lediglich 2 Objekte und wird eines davon nicht migriert, entsteht eine ungültige Objektgruppe. Die Objektgruppe wird in WinCC als ungültiges Objekt gekennzeichnet. Selektieren Sie das ungültige Objekt und heben Sie die Gruppierung auf. Das gültige Objekt wird angezeigt.

Wenn nicht alle Objekte einer Gruppe migriert werden können, dann werden die Koordinaten und die Größe der Gruppe neu berechnet und entsprechend angepasst.

In beiden oben beschriebenen Fällen erfolgt ein Eintrag im Migrationsprotokoll.

Migration von Eigenschaften einer Objektgruppe

Statische Werte und Dynamisierungen der Eigenschaften der gruppierten Objekte werden ohne Einschränkungen migriert.

Objektgruppen in WinCC V7 besitzen eigene Eigenschaften: X-Position, Y- Position, Breite, Höhe, Objektname, Ebene, Tooltip-Text, Berechtigungen, Bedienfreigabe und Sichtbarkeit. Statische Werte und Dynamisierungen dieser Eigenschaften werden ohne Einschränkungen migriert.

Neben den vorgenannten Eigenschaften besitzt eine Objektgruppe auch eine Sammlung aller Eigenschaften der gruppierten Objekte. Über diese gesammelten Eigenschaften ist es in WinCC V7 möglich, z. B. die Hintergrundfarbe aller gruppierten Objekte zu beeinflussen. Änderungen werden sowohl an der Objektgruppe als auch an den gruppierten Objekten gespeichert. Die Änderungen an der Objektgruppe werden nicht migriert. Auch die Dynamisierungen an den gesammelten Eigenschaften geht bei der Migration verloren.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Anwenderobjekten (WinCC V7)

Einleitung

Anwenderobjekte werden in WinCC nicht mehr unterstützt. Die Migration bildet Anwenderobjekte auf Bildbausteine ab. Identische Anwenderobjekte werden zu einem Bildbaustein-Typ zusammengefasst. Jedes Anwenderobjekt wird durch eine Bildbaustein-Instanz ersetzt.

Anwenderobjekte

Ein Bildbaustein bietet nicht exakt die gleichen Merkmale wie ein Anwenderobjekt, deshalb sind bei der Migration folgende Besonderheiten zu beachten:

- In einem Bildbaustein kann eine Eigenschaft eines inneren Objekts nur auf eine Eigenschaft des Bildbausteins verschaltet werden. Wenn eine Eigenschaft eines inneren Objekts auf mehrere Eigenschaften des Anwenderobjekts verschaltet ist, wird die Verbindung zur ersten Eigenschaft migriert. Für die nicht migrierten Eigenschaften wird im Migrationsprotokoll eine Meldung ausgegeben.
- Die Koordinaten für die Position eines Objekts innerhalb des Anwenderobjekts können nicht auf die Eigenschaften eines Bildbausteins verschaltet werden. Die Eigenschaft des Anwenderobjekts, an das eine Position projektiert ist, wird migriert. Wenn an dieser Eigenschaft eine Dynamik projektiert ist, wird diese ebenfalls migriert. Allerdings sind die Eigenschaft und die Dynamik im Bildbaustein nicht mehr mit der Objektposition des inneren Objekts verschaltet.
- Bildbausteine können nicht gedreht oder gespiegelt werden. Wenn das Projekt gedrehte oder gespiegelte Anwenderobjekte enthält, wird im Migrationsprotokoll eine Meldung ausgegeben.
- Wenn ein Anwenderobjekt Objekte enthält, die von der Migration nicht unterstützt werden, dann fehlen diese Objekte im resultierenden Bildbaustein nach der Migration. Der Bildbaustein enthält dann nur die von der Migration unterstützten Objekte. Wenn an ein nicht unterstütztes Objekt eine Dynamik projektiert ist, geht diese Dynamik verloren. Wenn ein inneres Objekt eines Anwenderobjekts nicht migriert wird, ändert sich gegebenenfalls die Größe des resultierenden Bildbausteins. Die Objektposition des Anwenderobjekts wird beibehalten und auf den Bildbaustein angewendet.
- Die Position eines inneren Objekts ist im Anwenderprojekt bezogen auf das Koordinatensystem des Bilds, in dem das Anwenderobjekt platziert ist. Ein Bildbaustein hat ein eigenes Koordinatensystem, das unabhängig ist vom Bild, in dem es platziert ist. Bei der Migration ändern sich daher die Koordinaten der inneren Objekte. Wenn das Projekt ein Skript enthält, das die Koordinaten eines inneren Objekts benutzt oder verändert, müssen Sie das Skript entsprechend anpassen. Die Änderung der Koordinaten eines inneren Objekts wird nicht protokolliert.
- Die Verwendung des Objekts "Verbinder" ist in einem Bildbaustein nicht erlaubt. Deshalb wird ein Verbinder innerhalb eines Anwenderobjekts nicht migriert.
- Die Verwendung von WinCC Controls ist in einem Bildbaustein nicht erlaubt. Deshalb wird ein Control innerhalb eines Anwenderobjekts nicht migriert. Eine Ausnahme bilden folgende Controls, diese werden auch als innere Objekte eines Anwenderobjekts migriert:
  - WinCC Digital/Analog Clock Control
  - WinCC Gauge Control
  - Slider Objekt

Wenn die Migration wegen der beschriebenen Besonderheiten Änderungen an den Projektierungsdaten vornimmt, werden diese im Migrationsprotokoll eingetragen.

Siehe auch

Migration (WinCC V7) (Seite 95)
Objektunterstützung bei der Migration (WinCC V7) (Seite 103)
Migration von Bildern und Bildobjekten (WinCC V7) (Seite 110)
Migration von Objektgruppen (WinCC V7) (Seite 114)
Migration von Dynamisierungen per Dynamik-Dialog (WinCC V7) (Seite 117)
Migration von Dynamisierungen per Skript (WinCC V7) (Seite 118)
Migration von Controls (WinCC V7) (Seite 120)
Migration der Farbeinstellungen (WinCC V7) (Seite 121)
Migration einer Dynamisierung per Direktverbindung (WinCC V7) (Seite 119)
Wertänderungen an Objekteigenschaften durch die Migration (WinCC V7) (Seite 113)

Migration von Dynamisierungen per Dynamik-Dialog (WinCC V7)

Einleitung

Für die Animation von Bildobjekten steht Ihnen in WinCC V7 der Dynamik-Dialog zur Verfügung. In WinCC werden die Funktionen des Dynamik-Dialogs als Animation einer Eigenschaft abgebildet. Diese Animation bietet Ihnen ähnliche Funktionalitäten zur Dynamisierung wie der Dynamik-Dialog in WinCC V7. Wenn Sie in einem WinCC V7-Projekt eine Dynamisierung an eine Eigenschaft projektiert haben, wird diese Dynamisierung von der Migration in eine Animation von WinCC konvertiert.

Hinweis

Die in WinCC seit V7.3 eingeführten neuen Eigenschaften werden nicht nach WinCC migriert. Wenn eine der neuen Eigenschaften dynamisiert ist, wird über die nicht migrierten dynamisierten Eigenschaften im Migrationsprotokoll eine Meldung ausgegeben.

Migration von Dynamisierungen per Dynamik-Dialog

Der Ausdruck einer Dynamisierung, die mithilfe des Dynamik-Dialogs aus WinCC V7 erstellt ist, basiert auf der Syntax von C-Skript. Die Migration versucht den Ausdruck einer Dynamisierung aus WinCC V7 auf die Syntax von VB-Skript zu konvertieren. Diese Form der Konvertierung kann meist dann durchgeführt werden, wenn die Dynamisierung folgende C-Operatoren verwendet:

- Arithmetische Operatoren [+, -, *, /, %, ^]
- Logische Operatoren [&&, ||]
- Vergleichende Operatoren [<, >, !=, <=, >=, ==]
- Nummerische Konstanten

Wenn die Konvertierung auf die Syntax von VB-Skript nicht möglich ist, wird die Dynamisierung auf ein C-Skript abgebildet. Die Abbildung auf ein C-Skript wird in folgenden Fällen durchgeführt:

- Wenn der Ausdruck des Dynamik-Dialogs keine Variable oder keinen zyklischen Trigger benutzt. Wenn eine Variable und eine Trigger-Variable verwendet werden, so müssen diese übereinstimmen. Bei Verwendung von Variablen im Ausdruck ist kein zyklischer Trigger erlaubt.
- Wenn das Ergebnis des Ausdrucks im Dynamik-Dialog vom Datentyp "Analog" ist, und im Feld "Ergebnis des Ausdrucks" der Geltungsbereich "sonst" einen anderen Wert hat als der Wert "Statik" der dynamisierten Eigenschaft.
- Wenn im Ausdruck des Dynamik-Dialogs mehrere Variablen benutzt werden und nicht alle Variablen die gleiche Konfiguration für den Quality Code haben.
- Wenn im Ausdruck des Dynamik-Dialogs eine Variable benutzt wird, deren Variablenstatus verwendet wird.
- Wenn der Ausdruck des Dynamik-Dialogs nicht in eine Entsprechung in VB-Skript konvertiert werden kann.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Dynamisierungen per Skript (WinCC V7)

Einleitung

Damit VB-Skripte und C-Skripte aus WinCC V7 funktionsfähig bleiben, führt die Migration verschiedenen Maßnahmen durch. Diese Maßnahmen gelten auch für Aktionen an Ereignissen.

Migration von VB-Skripten für die Dynamisierung

In WinCC V7 können Sie in einer Dynamisierung per VB-Skript mehrere Funktionen einsetzen, dies ist in WinCC nicht mehr erlaubt. Wenn die Migration mehrere Funktionen in einem Skript findet, dann wird die Funktion migriert, die bei Triggerung des Ereignisses aufgerufen wird. Die übrigen Funktionen werden in den Deklarationsbereich migriert. Beim Übersetzen des Projekts führt dies zu einer Meldung.

In WinCC ändern sich teilweise die Funktionsnamen von VB-Skripten gegenüber WinCC V7. Der Funktionsname "LeftTrigger" ändert sich z. B. in "LeftValue". Wenn die Funktion einen Wert zurückgeben muss, dann enthält der Quellcode ein Statement, in dem der Wert dem Funktionsnamen zugewiesen wird. Die Migration sucht und ersetzt den Funktionsnamen innerhalb der Funktion und dem Funktionskopf. Wenn der neue Funktionsname bereits als Name einer Funktion, einer Variable oder eines Parameters verwendet wird, dann wird im Migrationsprotokoll eine Meldung ausgegeben. Passen Sie die Projektierung nach der Migration selbst an.

Migration von C-Skripten für die Dynamisierung

In WinCC V7 haben einige Parameter für C-Skripte andere Namen als in WinCC. Um ein Skript funktionsfähig zu halten, fügt die Migration für jeden geänderten Parameternamen eine Variable mit dem bisherigen Parameternamen ein.
Beispiel:
Der Parametername "lpszPictureName" wird ersetzt durch den Parameternamen "ScreenName". Die Migration fügt im Skript die folgende Variable ein: "char* lpszPictureName = ScreenName".
Wenn bereits eine Variable oder ein Parameter mit diesem Namen existiert, wird im Migrationsprotokoll eine Meldung ausgegeben.
Folgende Parameternamen sind betroffen:

WinCC V7	WinCC
lpszPictureName	ScreenName
lpszObjectName	ObjectName
lpszPropertyName	PropertyName

In WinCC ändern sich die Funktionsnamen von C-Skripten gegenüber WinCC V7. Der Funktionsname einer Dynamik erhält den Ausdruck "_main". In WinCC ändert sich der Ausdruck nach "<Name der Eigenschaft>Value". Die Migration korrigiert die Namen wie die Änderung der Funktionsnamen von VB-Skripten im vorigen Absatz.

Migration von Triggern

Trigger aus WinCC V7 vom Typ Fenster-Zyklus werden migriert zu dem in WinCC gültigen Bildzyklus.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration einer Dynamisierung per Direktverbindung (WinCC V7)

Migration einer Direktverbindung

Eine Dynamisierung per Direktverbindung wird als Funktionsliste migriert.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Controls (WinCC V7)

Einleitung

Ein WinCC V7-Projekt kann folgende Controls enthalten:

- WinCC Controls vor V7
  Die Controls vor WinCC V7 werden in WinCC V7 noch unterstützt. In WinCC entfällt die Unterstützung dieser Controls. Wenn ein Projekt Controls vor WinCC V7 enthält, dann werden diese Controls in die Version ab WinCC V7 migriert. Dabei können sich auch die Namen von Eigenschaften, Schaltflächen oder der Symbolleiste ändern.
  In Runtime können sich Änderungen in der Darstellung ergeben.
  Skripte, die auf Controls vor WinCC V7 gewirkt haben, müssen Sie nach der Migration anpassen.
- WinCC Controls ab V7
  Die unten aufgeführten Controls ab WinCC V7 werden von WinCC unterstützt.
- Controls von Drittanbietern
  Die Migration unterstützt auch ActiveX-Controls und einfache .Net-Controls von Drittanbietern. XAML Controls, z. B. WPF (Windows Presentation Foundation) Controls werden nicht unterstützt. Die ActiveX-Controls müssen im Betriebssystem registriert sein. Wenn ein ActiveX-Control nicht registriert ist, wird die Migration abgebrochen.
  Wenn Sie ein Projekt mit dem Migrationstool speichern und die Migration selbst auf einem anderen PC durchführen, dann müssen die ActiveX-Controls auch auf diesem PC registriert sein.

Unterstützte WinCC Controls

WinCC unterstützt folgende Controls aus WinCC V7:

- Siemens HMI Symbol Library 1.4.1
- WinCC Channel Diagnosis Control
- WinCC Digital/Analog ClockControl
- WinCC Gauge Control
- WinCC Slider Control
- WinCC Ruler Control
- WinCC Media Control
- WinCC AlarmControl
- WinCC FunctionTrendControl
- WinCC OnlineTableControl
- WinCC OnlineTrendControl
- WinCC UserArchiveControl
- CCDiskSpace.Diskspace
- WinCC_WB.WinCCWebBrowser

Besonderheiten bei WinCC OnlineTableControl

In WinCC V7 kann WinCC OnlineTableControl Überschreitungen von Werten in verschiedenen Farben anzeigen. Dafür können wenigstens zwei obere und zwei untere Grenzwerte sowie zugehörige Farben konfiguriert werden.

WinCC unterstützt diese Eigenschaften nicht. Im Migrationsprotokoll wird eine Meldung ausgegeben, wenn entsprechende Werte konfiguriert sind.

Besonderheiten bei WinCC AlarmControl

Der neue Meldeblock "Class Priority" im WinCC AlarmControl von WinCC V7 wird von WinCC nicht unterstützt. Im Migrationsprotokoll wird eine Meldung ausgegeben, wenn der Meldeblock konfiguriert war.

Nicht unterstützte WinCC Controls

Folgende Controls werden von WinCC nicht unterstützt:

- WinCC Push Button Control
  Das WinCC Push Button Control wird in WinCC auf eine Schaltfläche abgebildet. Nähere Hinweise hierzu finden Sie unter Migration von Bildern und Bildobjekten (WinCC V7) (Seite 110).
- XAML Controls, z. B. WPF-Controls
  Wenn ein Projekt XAML Controls enthält, dann werden diese Controls nicht migriert. Für jedes nicht migrierte Control wird im Migrationsprotokoll eine Meldung ausgegeben.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Bildern und Bildobjekten (WinCC V7) (Seite 110)

Migration der Farbeinstellungen (WinCC V7)

Migration der Farbeinstellungen eines Objekts

In WinCC V7 ist es möglich, die Farbeinstellungen von Objekten über die zentrale Farbpalette zu projektieren. In WinCC entfällt die zentrale Farbpalette. Die Migration übernimmt die Farbeinstellungen für ein Objekt aus der zentralen Farbpalette und legt die Farben direkt am jeweiligen Objekt fest. Im migrierten Projekt ist eine zentrale Änderung der Farbeinstellungen nicht mehr möglich.

Siehe auch

Migration von Anwenderobjekten (WinCC V7) (Seite 115)

Migration von Meldungen und Meldegruppen (WinCC V7)

Einleitung

Das Meldesystem von WinCC unterscheidet sich deutlich in seinem Aufbau gegenüber dem Meldesystem von WinCC V7.

WinCC V7 bietet bis zu 18 Meldeklassen mit 16 Meldearten pro Meldeklasse.

WinCC bietet nur Meldeklassen, aber keine Meldearten.

Bei der Migration eines Projekts werden die Meldearten im migrierten Projekt zu Meldeklassen konvertiert. Die Meldearten entfallen vollständig.

Migration von Meldegruppen

Die Meldeklassen und Meldearten werden nach folgendem Schema migriert:

Die Migration vergibt durchgehend IDs für alle Meldeklassen und deren Meldearten. Die IDs 1-63 sind in WinCC systemseitig reserviert. Die Migration beginnt die Vergabe der IDs ab der ID 64.

Meldeklasse WinCC V7	Meldeart WinCC V7	Meldeklasse WinCC
ID 1	ID 1-16	ID 64-79
ID 2	ID 17-32	ID 80-95
...	...	...
ID 16	ID 241-256	ID 304-319
ID 17	ID 257-272 davon benutzt, 257-258	ID 320-335
ID 18	ID 273-288 davon benutzt, 273-274	ID 336-351

Die Migration berechnet neue IDs für alle Meldeklassen und Meldearten, die in dem zu migrierenden Projekt enthalten sind.

Nach der Vergabe der IDs werden die im Projekt existierenden Meldeklassen und Meldearten migriert. Um eindeutige Namen für die Meldeklassen zu gewährleisten, verknüpft die Migration die Namen der Meldeklassen mit den darin enthaltenen Meldearten.

Beispiel:

In WinCC V7 existiert die Meldeklasse "Tolerance" mit der Meldeart "ToleranceHigh". Nach der Migration wird daraus die Meldeklasse "Tolerance_ToleranceHigh".

Mit Ausnahme von Skripten und ODK-Funktionen werden alle Verwendungsstellen im Projekt entsprechend angepasst. Z. B. werden alle statischen Filter einer Meldeanzeige auf die neue Meldeklasse angepasst.

Die Migration korrigiert keine Skripte oder ODK-Funktionen, die eine Meldeart verarbeiten. Wenn die Migration die Verwendung einer Meldeart in einem Skript feststellt, wird im Migrationsprotokoll eine Meldung ausgegeben. Passen Sie ein solches Skript nach der Migration selbst an.

Migration von Meldungen

In WinCC V7 besitzen Systemmeldungen IDs aus dem Bereich von 1000000 bis 1019999. In diesem Zahlenbereich konnten aber auch Meldungen frei projektiert werden.

In WinCC ist Bereich von 1000000 bis 1019999 für Systemmeldungen reserviert.

WinCC V7-Systemmeldungen werden migriert. Anwenderdefinierte Meldungen mit IDs in diesem Bereich werden nicht migriert.

Wenn Anwendermeldungen nicht unterstützte ASCII-Zeichen enthalten, wird jedes nicht unterstützte ASCII-Zeichen durch folgenden Ausdruck ersetzt: "#<ASCII-Code>".

Einstellungen zur Archivierung

In WinCC V7 können Sie die Einstellungen für die Archivierung für jede Meldung separat festlegen. In WinCC können Sie die Einstellungen für die Archivierung nur für eine Meldeklasse und nicht für Einzelmeldungen festlegen. Wenn an einer Meldung die Archivierung aktiviert ist, übernimmt die Migration diese Einstellung für die gesamte Meldeklasse. Damit wird sichergestellt, dass keine Archivierungseinstellungen verloren gehen. Damit werden nach der Migration alle Meldungen archiviert, die in dieser Meldeklasse enthalten sind, auch wenn diese vor der Migration nicht archiviert wurden. Für jede betroffene Meldung wird im Migrationsprotokoll eine Meldung ausgegeben.

Siehe auch

Migration (WinCC V7) (Seite 95)

Migration von Anwenderarchiven (WinCC V7) (Seite 123)

Migration von Anwenderarchiven (WinCC V7)

Migration von Anwenderarchiven

Anwenderarchive aus WinCC V7 werden in WinCC Rezepturen genannt. Datenfelder eines Anwenderarchivs werden als Rezepturelemente bezeichnet.

Bei der Migration von Anwenderarchiven werden die Projektierungsdaten auf Rezepturen abgebildet. Sichten werden auf Rezepturabfragen abgebildet. Bei der Migration müssen Sie folgende Regel beachten:

- Wenn Sie in einer Sicht eine Relation projektiert haben, dann darf diese Relation nicht "NULL" sein.
  Wenn die Relation "NULL" ergibt, dann wird beim Übersetzen des migrierten Projekts eine Fehlermeldung ausgegeben. Um den Fehler zu beheben, müssen Sie die Relation in der Rezepturanzeige so nachprojektieren, dass eine gültige Relation entsteht.
- In WinCC V7 konnten Steuervariablen unabhängig von den Festlegungen für die Kommunikation projektiert werden. Steuervariablen werden nur migriert, wenn die Kommunikation über WinCC Variablen erfolgt.

- In WinCC V7 wird bei der Kommunikation über Rohdatenvariablen die PLCID als eindeutiger Name des Archivs verwendet. In WinCC wird zur eindeutigen Kennzeichnung einer Rezeptur die Rezepturnummer verwendet. Angaben zur PLCID werden nicht migriert.
- Wenn in WinCC V7 für ein Anwenderarchiv ein Alias projektiert ist, wird dieser zum Anzeigenamen der Rezeptur migriert.

Die Runtime-Daten von Anwenderarchiven können bei der Migration kopiert werden.

Siehe auch

Kopieren der Runtime-Daten (WinCC V7) (Seite 135)

Migration von Meldungen und Meldegruppen (WinCC V7) (Seite 122)

Skripte und Standardfunktionen (WinCC V7)

Migration von Funktionen und Aktionen (WinCC V7)

Einleitung

Wie WinCC V7 unterstützt auch WinCC sowohl ANSI-C als auch VBS. Durch die systemseitigen Änderungen sind bei der Migration von Funktionen und Aktionen einige Dinge zu beachten. Nachfolgend werden die Randbedingungen zur Migration beschrieben.

Migration von Funktionen und Aktionen

Beachten Sie bei der Migration von Funktionen und Aktionen folgende Punkte:

- VBA-Skripte werden in WinCC nicht mehr unterstützt, daher werden VBA-Skripte nicht migriert.
- Einige Funktionen erhalten in WinCC neue Namen. Während der Migration werden keine Funktionsnamen angepasst. Skripte sind ohne Anpassung funktionsfähig, denn in Skripten können die bisherigen Namen weiter verwendet werden. Die bisherigen Namen werden aber durch die Autovervollständigung und durch die Hilfe nicht mehr unterstützt.

Besonderheiten bei VB-Skripten

- In WinCC V7 sind VB-Skripte strukturiert als Module, die eine oder mehrere Prozeduren enthalten können. In WinCC darf jedes Skript nur eine Prozedur enthalten. Die Migration spaltet Module mit mehreren Prozeduren auf und legt für jede Prozedur ein neues Skript an.
  In einem Skript aus WinCC V7 können Deklarationen für mehrere globale Variablen enthalten sein. Die globalen Variablen werden beim Aufruf des Skripts initialisiert und sind für alle Prozeduren des Skripts verfügbar. Die globalen Variablen werden bei der Aufspaltung eines Skripts in das Skript migriert, das der Variablendeklaration folgt. Nach der Migration müssen Sie dafür sorgen, dass dieses erste Skript auch zuerst aufgerufen wird, damit die globalen Variablen initialisiert werden. Erst nach der Initialisierung sind die globalen Variablen für die abgespalteten Skripte verfügbar.
- In WinCC wird die Deklaration von VB-Klassen nicht unterstützt.
- In WinCC müssen die Parameter in einem VB-Skript grundsätzlich mit den Schlüsselworten "ByRef" oder "ByVal" übergeben werden. In WinCC V7-Skripten gibt es keine definierte Übergabe mit den Schlüsselworten. Bei der Migration wird die Übergabe der Parameter mit dem Schlüsselwort "ByRef" ergänzt.

Strengere Kriterien bei der Überprüfung von Funktionen können dazu führen, dass Skripte und Funktionen nach der Migration nachbearbeitet werden müssen. Fehler und Warnungen werden in WinCC unter anderem verursacht durch:

- Objekte, wie Variablen oder Bildobjekte, die in Funktionen referenziert werden, aber nicht existieren.
- Falsche Nutzung des VB-Objektmodells, z. B. HmiRuntime.ActivateScreen.Item.
- Unterschiedliche Anzahl der Parameter bei Definition und Aufruf einer Funktion.
- VB-Funktionen, die keinen Rückgabewert liefern.
- Wenn innerhalb einer VB-Funktion der Rückgabewert einer anderen Funktion gesetzt wird.
- Variablen, die in einer VB-Funktion verwendet werden, aber nicht in dieser Funktion definiert sind.

Besonderheiten bei C-Funktionen

In WinCC V7 wird bei den Namen von C-Funktionen die Groß- und Kleinschreibung berücksichtigt. In WinCC sind Namen, die sich nur durch die Groß- und Kleinschreibung unterscheiden, nicht erlaubt. Bei der Migration bleibt die Groß- und Kleinschreibung von Funktionsnamen unverändert erhalten. Wenn in einem migrierten Projekt Funktionsnamen enthalten sind, die sich nur durch Groß- und Kleinschreibung unterscheiden, wird eine der Funktionen im Engineering System als ungültig gekennzeichnet. Beim Übersetzen wird eine entsprechende Fehlermeldung ausgegeben.

Strengere Kriterien bei der Überprüfung von Funktionen können dazu führen, dass Skripte und Funktionen nach der Migration nachbearbeitet werden müssen. Fehler und Warnungen werden in WinCC unter anderem verursacht durch:

- Objekte, wie Variablen oder Bildobjekte, die in Funktionen referenziert werden, aber nicht existieren.
- Unterschiedliche Anzahl der Parameter bei Definition und Aufruf einer Funktion.

- Wenn die Extern-Definition einer verwendeten Funktion nicht mit der Funktionsdefinition übereinstimmt. In diesem Fall können Sie entweder die Extern-Definition entfernen oder die Definition anpassen.
- Wenn die Namen von Funktionen oder Variablen in den Defines nicht erlaubte Sonderzeichen enthalten

ODK-Funktionen

In WinCC V7 werden ODK-Funktionen sowohl für die Änderung von CS-Daten als auch für die Änderung von Runtime-Daten unterstützt. In WinCC werden nur noch Funktionen für die Änderung von Runtime-Daten unterstützt. Die ODK-Funktionen für die Änderung von CS-Daten werden nicht mehr unterstützt. Funktionen und Anwendungen, die diese ODK-Funktionen aufrufen, funktionieren nach der Migration unter Umständen nicht mehr wie gewünscht. Nähere Hinweise zur Verwendung von ODK-Funktionen finden Sie in der Online-Hilfe zu Runtime API.

Siehe auch

Migration (WinCC V7) (Seite 95)

Migration von Header-Dateien (WinCC V7) (Seite 126)

Migration von Aktionen (WinCC V7) (Seite 127)

Migration von Standardfunktionen (WinCC V7) (Seite 127)

Änderung von Funktionsnamen (WinCC V7) (Seite 137)

Migration von Header-Dateien (WinCC V7)

Einleitung

Durch die systemseitigen Änderungen sind bei der Migration von Header-Dateien einige Dinge zu beachten. Nachfolgend werden die Änderungen durch die Migration beschrieben.

Migration von Header-Dateien

In WinCC V7 werden Header-Dateien standardmäßig im Projekt abgelegt. Die im Projekt abgelegten Header-Dateien werden migriert und in WinCC in der Projektnavigation im jeweiligen Gerät unter "Funktionen > C-Header" abgelegt. Wenn Sie in WinCC V7 Header-Dateien an anderer Stelle abgelegt und in das Projekt eingebunden haben, dann werden diese Header-Dateien nicht migriert. Die nicht migrierten Header-Dateien müssen Sie selbst wieder in das Projekt einbinden.

Siehe auch

Migration von Funktionen und Aktionen (WinCC V7) (Seite 124)

Migration von Aktionen (WinCC V7)

Einleitung

Aktionen in WinCC V7 werden in WinCC zu Aufgaben migriert. Aufgaben verwalten Sie in WinCC im Editor "Aufgaben". Durch die systemseitigen Änderungen sind bei der Migration von Aktionen einige Punkte zu beachten.

Migration von Aktionen

Einer Aufgabe in WinCC ist immer nur ein Trigger zugeordnet. In WinCC V7 haben Sie die Möglichkeit, an einer Aktion mehrere Trigger zur Auslösung der Aktion zu projektieren, z. B. einen Variablentrigger und einen zyklischen Trigger. Bei der Migration wird eine Aktion mit mehreren Triggern aufgespaltet in mehrere Aufgaben. Für jeden Trigger wird eine neue Aufgabe angelegt.

In WinCC V7 gibt es globale Aktionen und lokale Aktionen. In WinCC gibt es nur globale Aufgaben. Eine globale Aktion wird zu einer globalen Aufgabe migriert. Eine lokale Aktion wird ebenfalls zu einer globalen Aufgabe migriert. Allerdings wird das Skript einer globalen Aufgabe erweitert. In der Erweiterung des Skripts wird die Systemvariable "@LocalMachineName" ausgewertet, damit die Aufgabe nur auf dem PC abläuft, der für die lokale Aktion vorgesehen ist. Überprüfen Sie deswegen im migrierten Skript den Namen des PC und passen Sie diesen gegebenenfalls an.

Wenn ein Skript erweitert wird, wird im Migrationsprotokoll eine Meldung ausgegeben.

Migration von kennwortgeschützten Aktionen

In WinCC V7 konnte eine Aktion mit einem Kennwort geschützt werden. Bei der Migration wird daraus eine Aufgabe und ein kennwortgeschütztes Skript erzeugt.

Siehe auch

Migration von Funktionen und Aktionen (WinCC V7) (Seite 124)

Migration von Standardfunktionen (WinCC V7)

Migration von Standardfunktionen

Standardfunktionen sind Teil der Runtime-Installation von WinCC, daher werden die Standardfunktionen nicht migriert. Wenn Sie in Ihrem WinCC V7-Projekt den Quellcode einer Standardfunktion geändert haben, geht die Änderung bei der Migration verloren.

Standardfunktion " OnDeactivateExecute "

Die Standardfunktion "OnDeactivateExecute" wird zu einer Aufgabe mit dem Trigger "Runtime-Stopp" migriert.

Anwenderdefinierte Standardfunktionen

Anwenderdefinierte Standardfunktionen werden in WinCC V7 im Ordner "APLIB" abgelegt. Um sicherzustellen, dass anwenderdefinierte Standardfunktionen migriert werden, sollten Sie diese Funktionen in das Projektverzeichnis kopieren.

Anwenderdefinierte C-Standardfunktionen finden Sie im Installationsverzeichnis von WinCC V7 im Ordner "wincc\aplib". Kopieren Sie diese Funktionen in das Projektverzeichnis in den Ordner "library".

Anwenderdefinierte C-Standardfunktionen finden Sie im Installationsverzeichnis von WinCC V7 im Ordner "wincc\aplib\ScriptLibStd". Kopieren Sie diese Funktionen in das Projektverzeichnis in den Ordner "ScriptLib".

Siehe auch

Migration von Funktionen und Aktionen (WinCC V7) (Seite 124)

Migration von Protokollen (WinCC V7)

Grundlagen zur Migration von Protokollen (WinCC V7)

Einleitung

Das Protokollsystem von WinCC V7 kann sowohl Projektierungsdaten als auch Runtime-Daten verarbeiten und ausgeben. WinCC unterstützt lediglich die Dokumentation von Runtime-Daten.

Die Festlegung, welche Daten dokumentiert werden, ist in WinCC V7 in Layout-Dateien hinterlegt. Die Ausgabe der Inhalte wird über Druckaufträge gesteuert.

Migration von Protokollen

Die Migration erkennt die Layout-Dateien für die Runtime-Dokumentation an deren Inhalt. Wenn eine Layout-Datei Objekte für die Projektdokumentation enthält, wird diese Layout-Datei nicht migriert. Wenn eine Layout-Datei eingebettete Layouts enthält, die ebenfalls für die Projektdokumentation benutzt werden, dann wird diese Layout-Datei nicht migriert. Wenn ein Druckauftrag eine Layout-Datei für die Projektdokumentation referenziert, dann wird dieser Druckauftrag nicht migriert. Für jede Layout-Datei, die nicht migriert wird, wird im Migrationsprotokoll eine Meldung ausgegeben.

Hinweis

Wenn ein WinCC V7-Projekt bereits mit früheren Versionen von WinCC, z.B. V5.0, erstellt wurde, können in einem Protokoll Objekte enthalten sein, die in Runtime durch Provider aus der früheren Version mit Daten versorgt werden. Diese Objekte werden nicht migriert.

Sprachabhängige Inhalte

Das Protokollsystem von WinCC V7 unterstützt keine mehrsprachigen Inhalte. Daher benötigt das Protokollsystem eine separate Layout-Datei für jede Runtime-Sprache. Die

Sprachzuordnung der Dateien wird durch Namenskonventionen der enthaltenden Ordner oder Dateinamen festgelegt. Bei der Migration werden die sprachabhängigen Layout-Dateien in einer mehrsprachigen Protokolldatei zusammengefasst. Um ein mehrsprachiges Protokoll zu erzeugen, muss die Migration daher verschieden Dateien berücksichtigen. Bei der Zusammenführung der Layout-Dateien für unterschiedliche Sprachen gelten folgende Regeln:

- Für die Migration der Protokollobjekte wird jeweils die zuerst gefundene Layout-Datei verwendet. Unterschiede im Layout der verschiedenen Sprachen gehen daher bei der Migration verloren.
- Um die sprachabhängigen Inhalte zu extrahieren, werden alle sprachabhängigen Dateien eines Protokolls benutzt.
- Die Texteinträge eines sprachunabhängigen Protokolls werden für alle projektierten Runtime-Sprachen übernommen.

Migration von Druckaufträgen

In WinCC V7 werden die Startparameter für einen Druckauftrag in den Eigenschaften des jeweiligen Druckauftrags festgelegt. In WinCC können Sie an einem Druckauftrag keine Startparameter festlegen. Für den Start von Druckaufträgen steht in WinCC der Aufgabenplaner zur Verfügung. Im Aufgabenplaner gibt es eine Aufgabe vom Typ "Druckauftrag".

Bei der Migration eines zyklischen Druckauftrags wird der Druckauftrag in zwei Teile aufgespaltet. Die Migration erstellt im Aufgabenplaner eine Aufgabe vom Typ "Druckauftrag". Außerdem erstellt die Migration einen Druckauftrag im Protokollsystem und verbindet diesen mit der neu erstellten Aufgabe.

Migration von System-Layouts

Folgende System-Layouts aus WinCC V7 werden migriert:

WinCC V7 Druckauftrag	WinCC Systemdruckauftrag	WinCC Systemprotokoll
AlarmControl - Picture	Alarm Control - Picture	@Alarm Control - Picture
AlarmControl - Table	Alarm Control - Table	@Alarm Control - Table
FunctionTrendControl	Function Trend Control - Picture	@Function Trend Control - Picture
OnlineTrendControl - Picture	Online Trend Control - Picture	@Online Trend Control - Picture
OnlineTableControl - Picture	Online Table Control - Picture	@Online Table Control - Picture
OnlineTableControl - Table	Online Table Control - Table	@Online Table Control - Table
UserArchiveControl - Picture	Recipe Control - Picture	@Recipe Control - Picture
UserArchiveControl - Table	Recipe Control - Table	@Recipe Control - Table
RulerControl - Table	Ruler Control - Table	@Ruler Control - Table
RulerControl - Picture	Ruler Control - Picture	@Ruler Control - Picture

Änderungen in einem dieser System-Layouts sind auch nach der Migration verfügbar.

Siehe auch

Migration (WinCC V7) (Seite 95)

Migration von Protokoll-Layouts (WinCC V7) (Seite 130)

Migration des Objekts "Variable" (WinCC V7) (Seite 131)

Migration eines Meldefolgeprotokolls (WinCC V7) (Seite 131)

Migration von Protokoll-Layouts (WinCC V7)

Einleitung

Die Layout-Datei für ein Protokoll von WinCC V7 ist in mehrere Abschnitte aufgeteilt. Der Aufbau eines Layouts in WinCC unterscheidet sich in einigen Details. Daher kann die Ausgabe eines Protokolls sich nach der Migration etwas von der Ausgabe in WinCC V7 unterscheiden.

Migration von Protokoll-Layouts

Die folgende Tabelle zeigt den unterschiedlichen Aufbau der Layouts aus WinCC V7 und WinCC:

WinCC V7	WinCC
Statische Ebene des Deckblatts	Wasserzeichen Titelseite
Deckblatt	Titelseite
Statische Ebene der Kopfzeile	Kopfzeile Detailseite
Statische Ebene des Protokoll-Inhalts	Wasserzeichen Detailseite
Protokollinhalt	Detailseite
Statische Ebene der Fußzeile	Fußzeile Detailseite
Schluss-Seite	Rückseite
Zeilenlayout	Zeilenprotokoll

Wenn ein Protokoll-Layout Seitenumbrüche enthält, generiert die Migration für jeden Seitenumbruch eine neue Detailseite im Protokoll von WinCC.

Welche Protokollobjekte in einem bestimmten Abschnitt eines Protokolls erlaubt sind, ist in WinCC V7 und WinCC unterschiedlich. Wenn ein Protokollobjekt aus einem WinCC V7-Protokoll in WinCC nicht erlaubt ist, wird dieses Objekt nicht migriert. Wenn ein Protokollobjekt nicht migriert wird, wird im Migrationsprotokoll eine Meldung ausgegeben.

Siehe auch

Grundlagen zur Migration von Protokollen (WinCC V7) (Seite 128)

Migration des Objekts "Variable" (WinCC V7)

Einleitung

Einzelne Protokoll-Objekte, die in WinCC V7 verfügbar sind, werden von WinCC nicht mehr unterstützt. Nicht mehr verfügbare Objekte werden durch die Migration so weit wie möglich auf die in WinCC verfügbaren Objekte abgebildet.

Statisches Metafile

Wenn eine Kopie des WinCC V7-Projektes migriert wird, kann durch die Migration das Ablageverzeichnis der zugeordneten *.emf-Datei nicht aufgelöst werden. In das Migrationsprotokoll wird ein entsprechender Hinweis eingetragen.

EA-Feld eines Protokolls

Das Protokollobjekt "Variable" aus WinCC V7 steht in WinCC nicht mehr zur Verfügung. Das Objekt wird in WinCC ersetzt durch ein EA-Feld. Bei der Abbildung des Objekts "Variable" auf das EA-Feld werden die Eigenschaften "Berechnung" und "Ausgabewert" ersetzt durch die Eigenschaft "Prozesswert". Die Eigenschaft "Berechnung" hat Vorrang vor der Eigenschaft "Ausgabewert". Wenn in dem zu migrierenden Projekt die beiden Eigenschaften verwendet sind, wird der Wert der Eigenschaft "Berechnung" migriert. Der Wert der Eigenschaft "Ausgabewert" wird nicht migriert.

Die Migration erfolgt wie in der nachfolgenden Tabelle dargestellt:

WinCC V7		WinCC
Berechnung	**Ausgabewert**	**Prozesswert**
nicht gesetzt	nicht gesetzt	nicht gesetzt
Skript	nicht gesetzt	Skript
nicht gesetzt	Variable	Variable
Skript	Variable	Skript

Wenn der Wert einer Eigenschaft nicht migriert wird, wird im Migrationsprotokoll eine Meldung ausgegeben.

Siehe auch

Grundlagen zur Migration von Protokollen (WinCC V7) (Seite 128)

Migration eines Meldefolgeprotokolls (WinCC V7)

Einleitung

In WinCC steht Ihnen ein Druckauftrag für ein Meldefolgeprotokoll zur Verfügung. Für diesen Druckauftrag können Sie ein Systemprotokoll für Zeilendrucker oder ein Systemprotokoll für Seitendrucker projektieren. In WinCC V7 steht ebenfalls nur ein Druckauftrag für ein Zeilenprotokoll zur Verfügung, allerdings können mehrere Layouts für die Ausgabe projektiert werden.

Migration eines Meldefolgeprotokolls

Da in WinCC für das Meldefolgeprotokoll nur ein Zeilenlayout oder ein Seitenlayout erlaubt ist, wird das am Druckauftrag projektierte Layout migriert. Wenn Sie nach der Migration einen anderen Typ von Meldefolgeprotokoll verwenden wollen, müssen Sie in WinCC im Kontextmenü von Protokolle den Befehl "Systemprotokolle hinzufügen" ausführen.

Wenn ein Protokoll-Layout nicht migriert wird, wird im Migrationsprotokoll eine Meldung ausgegeben.

Siehe auch

Grundlagen zur Migration von Protokollen (WinCC V7) (Seite 128)

Migration der Benutzerverwaltung (WinCC V7)

Migration der Benutzerverwaltung

Die Benutzerverwaltung von WinCC V7 unterscheidet sich in einigen Details von der in WinCC. Daher kann es bei der Migration der Benutzerverwaltung zu Änderungen kommen.

In WinCC V7 kann ein Benutzer Berechtigungen haben, die sich von den Berechtigungen seiner Benutzergruppe unterscheiden. In WinCC werden die Berechtigungen auf Ebene der Benutzergruppen vergeben.

Um alle Berechtigungen zu migrieren, legt die Migration zuerst für jede Benutzergruppe eine neue Benutzergruppe an. Zusätzlich wird für jeden Benutzer eine neue Gruppe angelegt, bei dem sich die Berechtigungen von der Benutzergruppe unterscheiden. Wenn mehrere Benutzer die gleiche Abweichung von der Gruppenberechtigung haben, werden sie alle Mitglieder der gleichen, neu angelegten Gruppe.

Beispiel 1

WinCC V7	WinCC
Group 1(Auth A, Auth B) User 1(Auth A, Auth B) User 2(Auth C) User 3(Auth C)	Group 1(Auth A, Auth B) User 1
	Group 1_1(Auth C) User 2 User 3

Beispiel 2

WinCC V7	WinCC
Group 2(Auth A, Auth D) User 4(Auth D) User 5(Auth A, Auth B)	Group 2(Auth A, Auth D) User empty
	Group 2_1(Auth D) User 4
	Group 2_2(Auth A, Auth B) User 5

Beispiel 3

WinCC V7	WinCC
Group 3(Auth C, Auth E) User empty	Group 3(Auth C, Auth E) User empty

Für jede neu angelegte Benutzergruppe wird im Migrationsprotokoll eine Meldung ausgegeben.

Migration von Kennwörtern

Kennwörter werden nicht migriert. Für den Administrator ist das Kennwort "administrator" voreingestellt. Den Benutzern sind keine Kennwörter zugewiesen. Sie müssen die Kennwörter in WinCC neu vergeben.

Migration von Zyklen (WinCC V7)

Migration von Zyklen

In WinCC V7 gibt es die folgenden zwei Arten von Zyklen:

- Aktualisierungszyklen
- Archivierungszyklen

Diese Arten von Zyklen werden in WinCC V7 in unterschiedlichen Editoren bearbeitet.

In WinCC gibt es nur einen Editor für die Bearbeitung aller Zyklen.

Die Zyklen eines Projekts werden folgendermaßen migriert:

- Die 10 Standard-Aktualisierungszyklen werden unverändert benutzt.
- Die Migration legt fünf anwenderdefinierte Aktualisierungszyklen an.
- Die Migration legt für jeden Archivierungszyklus einen neuen Zyklus an.

WinCC verwaltet intern, welche Zyklen zur Aktualisierung und welche zur Archivierung dienen.

Nicht unterstützte Einstellungen

Von WinCC werden nicht alle in WinCC V7 möglichen Zykluseinstellungen unterstützt. Bei der Migration wird eine nicht unterstützte Einstellung durch eine in WinCC vorhandene ersetzt und eine Meldung im Migrationsprotokoll ausgegeben.

Die Archivvariablen mit einem nicht unterstützten Erfassungszyklus werden daher in Runtime mit einem anderen als dem projektierten Zyklus archiviert.

Beispiel: Wenn ein Zyklus z. B. für "Montag", "Mittwoch" und "Freitag" konfiguriert war, wird die Variable nach der Migration "einmal täglich" archiviert.

Migration der Prozesswertarchivierung (WinCC V7)

Einleitung

Die Einstellungen zur Prozesswertarchivierung in WinCC V7 werden nach WinCC migriert. Dabei gibt es Besonderheiten.

Manuelle Eingabe

In WinCC V7 können in Archive und in einzelne Archivvariablen Werte manuell eingegeben werden. Zusätzlich gibt es die Funktion "Neuberechnung nach manueller Eingabe".

Die manuelle Eingabe und Neuberechnung werden in WinCC nicht unterstützt. Bei der Migration werden diese Einstellungen ignoriert und im Migrationsprotokoll wird eine Meldung ausgegeben.

Basic Process Control (WinCC V7)

Einleitung

Ein Projekt, in dem Sie Basic Process Control verwendet haben, können Sie grundsätzlich nach WinCC migrieren. Bei der Migration eines solchen Projekts gehen jedoch spezifische Projektierungen für Basic Process Control verloren.

Migration von Basic Process Control

Nicht migriert werden die Default-Bilder von Basic Process Control, sowie die von Basic Process Control generierten Bilder. Andere Bilder eines Projekts werden migriert. Diese Bilder können jedoch Objekte enthalten, die von WinCC nicht unterstützt werden. Diese Objekte werden nicht migriert. Folgende Objekte sind betroffen:

- Hörmelder
- 3D-Balken
- Erweiterte Analoganzeige
- Erweiterte Zustandsanzeige
- Sammelanzeige

Eine Bildhierarchie, die mit Basic Process Control erstellt wurde, wird nicht migriert. Es werden nur die Bilder migriert, die für die Anzeige im Arbeitsbereich projektiert sind. Bild in Bild Merkmale werden nicht migriert.

Nicht unterstützte Komponenten und Funktionen

Folgende Komponenten von Basic Process Control werden von WinCC nicht unterstützt:

- OS-Projekteditor
- Lifebeat Monitoring
- Picture Tree Manager

- Bausteinlisten Editor
- Bereichsspezifische Benutzerverwaltung
- Multi-VGA

Einträge in der Anlaufliste zu Prozessen aus Basic Process Control werden von der Migration nicht unterstützt.

Wenn Sie in einem Skript API-Funktionen von Basic Process Control aufrufen, erhalten Sie eine Fehlermeldung bei der Übersetzung des Projekts.

Im Migrationsprotokoll werden entsprechende Meldungen ausgegeben.

2.8.2.3 Kopieren der Runtime-Daten (WinCC V7)

Runtime-Daten

Bei der Migration eines Projekts werden die Projektierungsdaten migriert. Um Runtime-Daten weiterhin zu verwenden, können Sie bei der Migration die Runtime-Daten kopieren.

Die Runtime-Daten setzen sich wie folgt zusammen:

- Anwenderarchive aus User Archives
  Anwenderarchivdaten sind Prozessdaten, die in Runtime in eine Datenbank gespeichert werden.
- Archivdaten
  Die Daten aus Variablenarchiven und Meldearchiven werden in Runtime erfasst und archiviert.

Siehe auch

Migration (WinCC V7) (Seite 95)

2.8.2.4 Referenz (WinCC V7)

Migration der Datentypen (WinCC V7)

Einleitung

Die nachfolgende Tabelle beschreibt die Abbildung der Datentypen aus WinCC V7 auf die Datentypen in WinCC.

Migration der Datentypen

Die internen Datentypen werden bei der Migration folgendermaßen aufeinander abgebildet:

Interne Datentypen WinCC V7	Interne Datentypen WinCC
Bool	Bool
Char	SByte
Byte	UByte
Int	Short
UInt	UShort
Long	Long
ULong	ULong
Float	Float
Double	Double
String	WString
DateTime	DateTime

Abbildung des S7-Datentyps "Timer"

Eine externe Variable mit dem S7-Datentyp "Timer" mit Timer-Adresse wird auf den S7-Datentyp "Timer" abgebildet. Die Adresse wird beibehalten.

Eine externe Variable mit dem S7-Datentyp "Timer" mit Adressierung eines Datenbausteins oder einer Merker-Adresse wird auf den S7-Datentyp "S5 Time" abgebildet. Die Adresse wird beibehalten.

Beispiel:

WinCC V7

Variable	S7-Datentyp	Adresse	Kommentar
Timer_Actual_Value	Timer	T10	BCD codierter Timer-Wert
Timer_Setpoint_Value	Timer	DB10.DBW200	BCD codierter Timer-Wert
Timer_Setpoint_Value#2	Timer	MW20	BCD codierter Timer-Wert

WinCC

Variable	S7-Datentyp	Adresse	Kommentar
Timer_Actual_Value	Timer	%T10	BCD codierter Timer-Wert
Timer_Setpoint_Value	S5Time	%DB10.%DBW200	BCD codierter Timer-Wert
Timer_Setpoint_Value#2	S5Time	%MW20	BCD codierter Timer-Wert

Siehe auch

Migration (WinCC V7) (Seite 95)

Änderung von Funktionsnamen (WinCC V7) (Seite 137)

Änderung von Funktionsnamen (WinCC V7)

Geänderte Funktionsnamen

In WinCC ändern sich teilweise die Funktionsnamen von C-Funktionen und VB-Funktionen gegenüber WinCC V7. Folgende Funktionsnamen sind betroffen:

WinCC V7	WinCC
GetLinkedVariable	GetLinkedTag
GetLocalPicture	GetLocalScreen
GetParentPicture	GetParentScreen
GetParentPictureWindow	GetParentScreenWindow
PASSCheckLevelPermission	IsUserAuthorized
OpenHomePicture	ActivateStartScreen
OpenNextPicture	ActivateNextScreen
OpenPrevPicture	ActivatePreviousScreen
OpenStoredPicture	ActivateStoredScreen
StorePicture	StoreScreen
GetLanguage	GetLanguageByLocaleID
InquireLanguage	InquireLocale or InquireLocaleID
SetLanguage	SetLanguageByLocaleID
OpenPicture	ActivateScreen
PASSLoginDialog	ShowLogonDialog
ProgramExecute	StartProgram
DeactivateRTProject	StopRuntime
ExitWinCC	StopRuntime

Siehe auch

Migration von Funktionen und Aktionen (WinCC V7) (Seite 124)

2.8.2.5 Integrierte Projekte

Grundlagen zur Migration integrierter Projekte (WinCC V7)

Einleitung

In einem in STEP 7 integrierten Projekt sind enthaltene Steuerungen und HMI-Geräte durch die Projektierung miteinander verbunden. Auch die Projektierungsdaten von WinCC V7 und STEP 7 haben Bindungen zueinander. Bei der Migration eines integrierten Projekts wird das komplette Projekt mit den Anteilen aus WinCC V7 und STEP 7 migriert. Die Bindungen bleiben erhalten.

Beachten Sie für die Migration integrierter Projekte Folgendes:

- Vor der Migration eines integrierten WinCC V7 Projekts ins TIA Portal erstellen Sie mit dem Migrations-Tool die Migrationsdatei (Intermediate file).
- Es können nur Projekte migriert werden, die mit WinCC V7.4 SP1 erstellt wurden.
- Steuerungsmeldungen und integrierte Variablen werden nur für Verbindungen vom Typ TCP/IP, MPI und PROFIBUS erstellt. Named Connections werden nicht unterstützt.
- Damit eine Bedienstation (OS) migriert wird, muss sie sich im SIMATIC-Manager unterhalb einer PC-Station und der WinCC-Anwendung befinden.

Migration eines integrierten Projekts

Bei der Migration eines integrierten Projekts gelten für den WinCC V7-Anteil die gleichen Voraussetzungen wie für die Migration eines nicht integrierten WinCC V7-Projekts. Die im WinCC V7-Anteil enthaltenen Objekte und Eigenschaften müssen von WinCC unterstützt werden, z. B. der Kommunikationstreiber.

Neben den Voraussetzungen für den WinCC V7-Anteil gelten weitere Voraussetzungen für den STEP 7-Anteil des integrierten Projekts. Die im STEP 7-Anteil enthaltenen Objekte und Eigenschaften müssen in STEP 7 unterstützt werden. Nähere Hinweise hierzu finden Sie in der Dokumentation zu STEP 7.

Um ein integriertes Projekt vollständig migrieren zu können, müssen auf dem PC für die Migration folgende Komponenten installiert sein:

- STEP 7 V5.4 SP5 ... STEP 7 V5.6
- WinCC V7.4 SP1

Um ein integriertes Projekt vollständig nachbearbeiten zu können, müssen auf dem PC für die Nachbearbeitung folgende Komponenten installiert sein:

- STEP 7
- WinCC

Wenn Sie das WinCC V7-Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung.

Ein integriertes Projekt wird immer vollständig migriert. Das im integrierten Projekt enthaltene WinCC-Projekt können Sie nicht alleine migrieren. Sie können lediglich das enthaltene STEP 7-Projekt alleine migrieren. Wenn Sie das STEP 7-Projekt alleine migrieren wollen, müssen Sie im SIMATIC-Manager das integrierte WinCC-Projekt löschen. Anschließend migrieren Sie das STEP 7-Projekt ohne das WinCC-Projekt.

Multi-Clients

Multi-Clients werden in WinCC nicht unterstützt. Wenn Sie im SIMATIC-Manager einen WinCC-Client anlegen, wird automatisch ein Multi-Client erzeugt. Das Multi-Client-Projekt ist in STEP 7 integriert. Wenn die Migration einen Multi-Client erkennt, bricht die Migration ab. Um das STEP 7-Projekt trotzdem zu migrieren, müssen Sie den Multi-Client im SIMATIC-Manager löschen. Anschließend migrieren Sie das Projekt.

Speicherort eines integrierten WinCC-Projekts

Wenn Sie ein integriertes Projekt migrieren wollen, dann muss der darin enthaltene HMI-Anteil auf dem gleichen PC liegen, wie der STEP 7-Anteil des Projekts. Wenn der HMI-Anteil auf einem anderen PC liegt, dann wird nur der STEP 7-Anteil migriert, der HMI-Anteil des Projekts fehlt nach der Migration.

Nicht unterstützte Objekte

Folgende Komponenten werden von der Migration nicht unterstützt:

- STEP 7-Multi-Projekt
  Ein STEP 7-Multi-Projekt kann nicht migriert werden, die Migration bricht den Vorgang ab.
- Central Archive Server - CAS
  Wenn die Migration einen CAS erkennt, bricht die Migration ab.

Siehe auch

Integriertes Projekt migrieren (WinCC V7) (Seite 139)

Integriertes Projekt migrieren (WinCC V7)

Einleitung

Bei der Migration eines integrierten Projekts werden sowohl die Anteile aus dem WinCC V7-Projekt als auch aus dem STEP 7 Projekt migriert. Deshalb migrieren Sie die vom Migrations-Tool erzeugte Intermediate-Datei. Die Intermediate-Datei hat das Dateiformat "*.am*". Bei der Migration werden die Daten aus dem bestehenden Projekt kopiert und in ein neues Projekt migriert. Sie können nicht in ein bestehendes Projekt migrieren.

Voraussetzung

- STEP 7 Professional und alle im Projekt verwendeten Optionspakete sind installiert.
- WinCC Professional ist installiert.
- Das TIA Portal ist neu gestartet.
- Die Intermediate-Datei des zu migrierenden Projekts liegt auf Ihrem Laufwerk vor.

Vorgehen

Um ein integriertes Projekt zu migrieren, gehen Sie folgendermaßen vor:

1. Wählen Sie "Start > Projekt migrieren".

2. Navigieren Sie im Feld "Quellpfad" zur Intermediate-Datei.
3. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.
4. Klicken Sie auf "Migrieren".
   Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
   - Die Projektansicht wird geöffnet.
   - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
   - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
   - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
   - Nach Abschluss der Migration wird eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.

Alternativ starten Sie die Migration aus der Projektansicht über "Projekt > Projekt migrieren".

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte Bediengerät ein neu angelegtes Gerät. Für jede migrierte Steuerung finden Sie ebenfalls ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten.

Hinweis

Integrierte Projekte mit Meldeanzeigen

Bei der Migration des Projekts können die Meldeklassen in der Meldeanzeige deaktiviert werden. Prüfen Sie nach der Migration des Projekts die Einstellungen in der Meldeanzeige. Aktivieren Sie gegebenenfalls im Inspektorfenster der Meldeanzeige unter "Eigenschaften > Allgemein" die erforderlichen Meldeklassen.

Migrationsprotokoll öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Sie können das Protokoll zu einem späteren Zeitpunkt ansehen. Die Protokolldatei öffnen Sie folgendermaßen:

1. Klicken Sie in der Projektnavigation auf "Gemeinsame Daten".
2. Klicken Sie auf die "Protokolle".
3. Doppelklicken Sie das Migrationsprotokoll.

Siehe auch

Grundlagen zur Migration integrierter Projekte (WinCC V7) (Seite 137)

2.9 Integrierte Projekte migrieren

2.9.1 Migration eines integrierten Projekts

Einleitung

In integrierten Projekten verwenden Sie SIMATIC-Steuerungen und WinCC-Komponenten zusammen in einem Projekt. Bei der Migration eines integrierten Projekts wird das komplette Projekt mit den Anteilen aus WinCC und STEP 7 migriert. Projektierte Verbindungen zwischen Steuerung und Visualisierung bleiben erhalten.

Migration eines integrierten Projekts

Bei der Migration eines integrierten Projekts gelten für den STEP 7-Anteil die gleichen Voraussetzungen wie für die Migration eines nicht integrierten STEP 7-Projekts. Die im WinCC Anteil enthaltenen Objekte und Eigenschaften müssen ebenso in WinCC (TIA Portal) unterstützt werden.

Um eine Bedienstation (OS) zu migrieren, muss die Bedienstation im Projektnavigator des SIMATIC-Managers unterhalb einer PC-Station und der WinCC-Anwendung liegen. Das folgende Bild zeigt die Zuordnung der Bedienstation im Ausgangsprojekt:

Weitere Migrationsvoraussetzungen für integrierte Projekte finden Sie in der Dokumentation zu WinCC.

Beachten Sie zusätzlich, dass das Ausgangsprojekt vor der Migration übersetzt werden muss.

Um ein integriertes Projekt vollständig zu migrieren, müssen auf dem PG/PC für die Migration folgende Komponenten installiert sein:

- STEP 7 V5.4 SP5 oder höher
- WinCC V7.4 SP1 mit dem neuesten Update oder WinCC Flexible 2008 ab SP2

Um ein integriertes Projekt vollständig nachbearbeiten zu können, müssen auf dem PC für die Nachbearbeitung folgende Komponenten in der aktuellsten Version installiert sein:

- STEP 7 Professional
- WinCC Basic, WinCC Comfort/Advanced oder WinCC Professional, in Abhängigkeit der verwendeten Komponenten

Verwendung des Migrations-Tools

Unter den folgenden Voraussetzungen ist die Verwendung des Migrations-Tools erforderlich:

- Das Ausgangsprojekt befindet sich nicht auf demselben PG/PC wie die Installation des TIA Portals.
- Im Ausgangsprojekt befinden sich SCADA-Geräte. Diese können ausschließlich mit dem Migrations-Tool migriert werden.
- WinCC Professional V15 und STEP 7 mit WinCC V7.4 SP1 können nicht auf demselben PG/PC installiert sein. Daher müssen integrierte Projekte mit WinCC V7.4 SP1-Anteilen mithilfe des Migrations-Tools für die Migration aufbereitet werden.

Migration des STEP 7-Teils eines integrierten Projekts

Ein integriertes Projekt wird immer vollständig migriert. Einzelne Teile daraus können nicht alleine migriert werden. Sie können lediglich das enthaltene STEP 7-Projekt alleine migrieren, wenn Sie zuvor im SIMATIC-Manager alle HMI-Stationen löschen und anschließend in NetPro das Projekt neu übersetzen.

Alternativ können Sie das Projekt in einer Installation von STEP 7 V5.4 SP5 oder höher ohne eine Installation von WinCC öffnen. Speichern Sie das Projekt dann neu ab und wählen Sie beim Speichern die Funktion "Reorganisieren". Dabei werden beim Speichern der Kopie die WinCC-Anteile automatisch entfernt.

Anschließend migrieren Sie das STEP 7-Projekt ohne das WinCC-Projekt.

Migration eines integrierten Projekts mit der Hardware-Konfiguration

Bei integrierten Projekten werden die HMI-Geräte auch dann migriert, wenn Sie die Hardware-Konfiguration nicht in die Migration einschließen. Der STEP 7-Anteil der Hardware-Konfiguration, einschließlich der Netzkonfigurationen, Verbindungen und Alarme, wird nur dann migriert, wenn Sie die Hardware-Konfiguration in die Migration einschließen. Andernfalls werden für die STEP 7-Geräte unspezifizierte Baugruppen angelegt, die Sie nach der Migration wieder in eine passende Baugruppe umwandeln müssen.

HMI-Baugruppen, die in einer PC-Station gesteckt sind, werden bei der Migration zu einer eigenen Station umgewandelt. Wenn Sie die Migration unter Ausschluss der Hardware-Konfiguration durchführen, dann enthält das migrierte Projekt eine nicht spezifizierte SIMATIC PC-Station und eine SIMATIC PC-Station mit den HMI-Geräten. Bezüge zu HMI-Geräten werden bei der Migration nicht übernommen. Bei eingeschlossener Hardware-Konfiguration enthält das migrierte Projekt zwei separate Stationen, die HMI-Station und die PC-Station.

Speicherort eines integrierten WinCC-Projekts

Wenn Sie ein integriertes Projekt migrieren, muss der darin enthaltene HMI-Anteil auf demselben PG/PC liegen, wie der STEP 7-Anteil des Projekts. Liegt der HMI-Anteil auf einem anderen PG, wird nur der STEP 7-Anteil migriert.

Nicht unterstützte Objekte

Folgende Komponenten werden von der Migration nicht unterstützt:

- STEP 7-Multi-Projekt
  Ein STEP 7-Multi-Projekt kann nicht migriert werden. Die Migration bricht den Vorgang ab.
- Central Archive Server - CAS
  Wenn in einem integrierten Projekt ein CAS enthalten ist, dann wird die Migration zwar durchgeführt, die CAS-Daten werden jedoch nicht migriert.

Siehe auch

Integrierte Projekte nacharbeiten (Seite 144)

2.9.2 Integrierte Projekte nacharbeiten

Wenn Sie ein integriertes Projekt ohne Hardware-Konfiguration migriert haben, dann werden unspezifizierte CPUs anstelle der CPUs des usprünglichen Projekts verwendet. Da zwischen einer unspezifizierten CPU und einem HMI-Gerät keine Verbindung bestehen kann, werden Verbindungen aus dem Ausgangsprojekt ebenfalls nur unspezifiziert übernommen.

Vorgehen

Um ein integriertes Projekt nach der Migration weiter zu nutzen, gehen Sie folgendermaßen vor:

1. Wandeln Sie die unspezifizierten Geräte wieder in passende Geräte um.
2. Stellen Sie die integrierte HMI-Verbindung zwischen dem HMI-Gerät und der PLC wieder her.
3. Verbinden Sie alle HMI-Variablen mit der neu erstellten integrierten Verbindung.
4. Stellen Sie die Verbindung zwischen HMI-Variablen und PLC-Variablen wieder her.
5. Löschen Sie die nicht integrierte HMI-Verbindung.

Die einzelnen Schritte sind in den folgenden Kapiteln anhand eines Beispielprojekts genauer beschrieben.

Siehe auch

Unspezifizierte CPUs in spezifizierte CPUs umwandeln (Seite 145)

Integrierte HMI-Verbindung erstellen (Seite 147)

HMI-Variablen neu verknüpfen (Seite 149)

Nicht integrierte HMI-Verbindung löschen (Seite 150)

2.9.3 Unspezifizierte CPUs in spezifizierte CPUs umwandeln

Der erste Schritt nach der Migration ohne Hardware-Konfiguration ist das Umwandeln der unspezifizierten CPUs in spezifische CPUs. Unspezifizierte CPUs sind als Platzhalter für bestimmte CPUs aus dem Hardware-Katalog zu verstehen, welche aktuell noch nicht bekannt sind. Sie können allgemeine Parameter festlegen und die CPUs bereits im Anwenderprogramm referenzieren. Das Projekt ist allerdings nicht vollständig funktionsfähig, solange die unspezifizierte CPU noch nicht spezifiziert wurde.

Eine CPU über den Baugruppentausch spezifizieren

Um eine unspezifizierte CPU über den Baugruppentausch zu spezifizieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die unspezifizierte CPU in der Netz- oder Gerätesicht.
2. Wählen Sie im Kontextmenü den Befehl "Gerät tauschen".
   Der Dialog "Gerät tauschen" wird geöffnet.

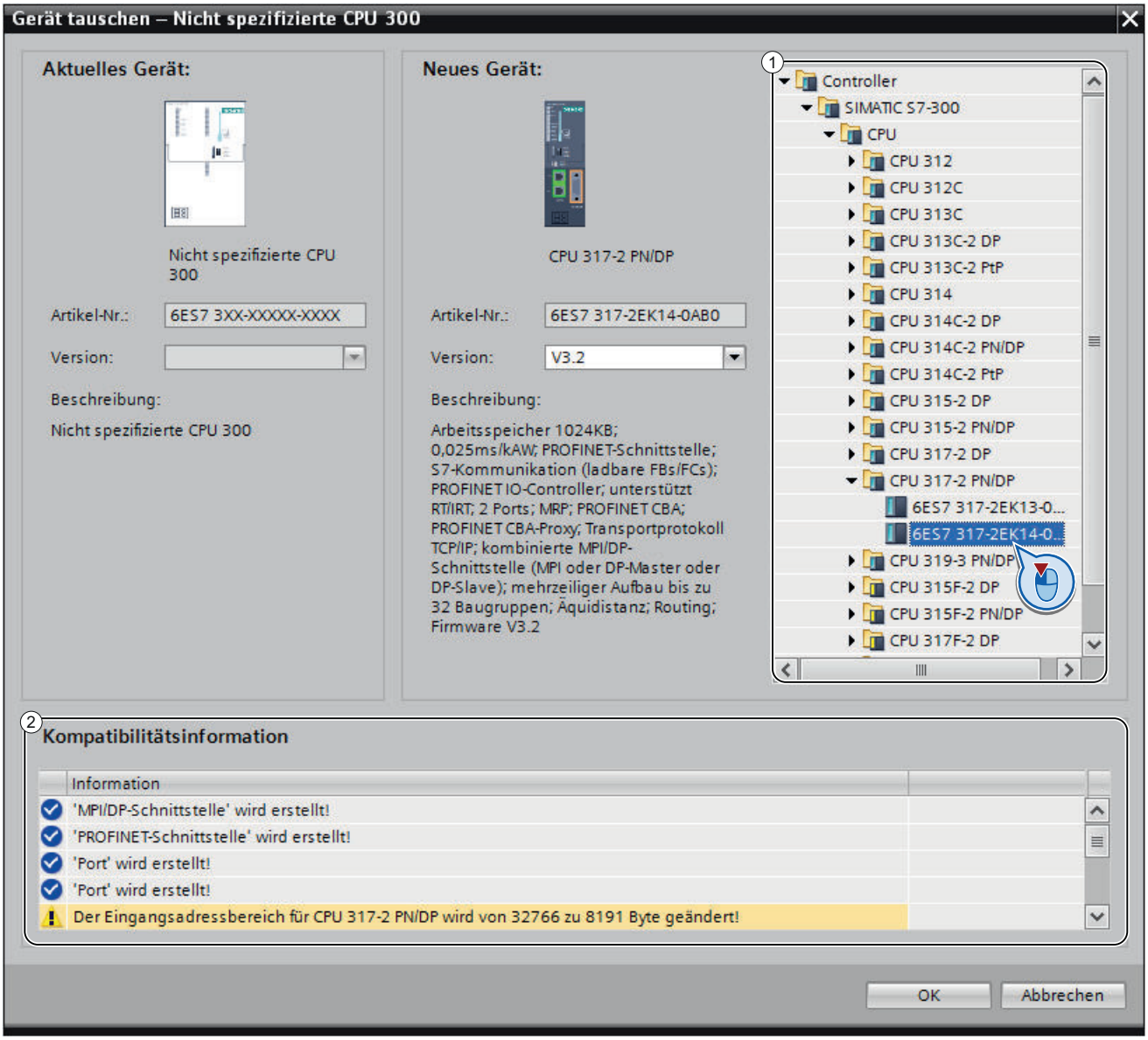

3. Selektieren Sie in der Baumstruktur unter "Neues Gerät" die Baugruppe, durch die Sie die unspezifizierte CPU ersetzen möchten. (Bereich 1)
   Unter "Kompatibilitätsinformationen" erhalten Sie Hinweise, inwieweit die gewählte CPU mit der Konfiguration im Ausgangsprojekt kompatibel ist. (Bereich 2)
4. Klicken Sie auf "OK".
5. Führen Sie die vorangegangen Schritte für alle unspezifizierten CPUs durch.

Siehe auch

Integrierte HMI-Verbindung erstellen (Seite 147)

2.9.4 Integrierte HMI-Verbindung erstellen

Nachdem Sie die unspezifizierte CPU spezifiziert haben, stellen Sie die Verbindung zum HMI-Gerät her.

Vorgehen

Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

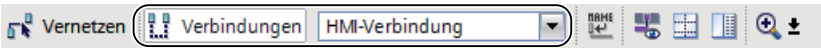

2. Wählen Sie den Verbindungstyp "HMI-Verbindung" in der danebenstehenden Klappliste. In der Netzsicht sind alle CPUs und HMI-Geräte, die für eine HMI-Verbindung in Frage kommen, farblich hervorgehoben.
3. Sie können nun den Verbindungsweg automatisch bestimmen lassen, oder explizit einen Verbindungsweg über bestimmte Schnittstellen wählen:
   - Verbindungsweg automatisch bestimmen lassen
     Selektieren Sie die CPU, von der eine Verbindung ausgehen soll. Ziehen Sie die Maus auf die Zielkomponente. Bestätigen Sie den Verbindungsendpunkt durch erneuten Mausklick.
     Alternativ: Selektieren Sie bei gedrückter Shift-Taste zusätzlich die Zielkomponente und wählen Sie mit der rechten Maustaste den Befehl "Neue Verbindung hinzufügen".
   - Expliziten Verbindungsweg von Schnittstelle zu Schnittstelle wählen
     Klicken Sie auf die Subnetzschnittstelle in dem Gerät, von dem eine Verbindung ausgehen soll. Ziehen Sie anschließend den Mauszeiger bei gedrückter Maustaste auf die gewünschte Schnittstelle im Zielgerät und lassen Sie die Maustaste los.

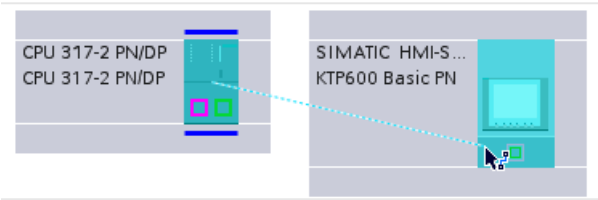

Ergebnis

Das folgende Bild zeigt den Zustand nach dem Anlegen der integrierten Verbindung:

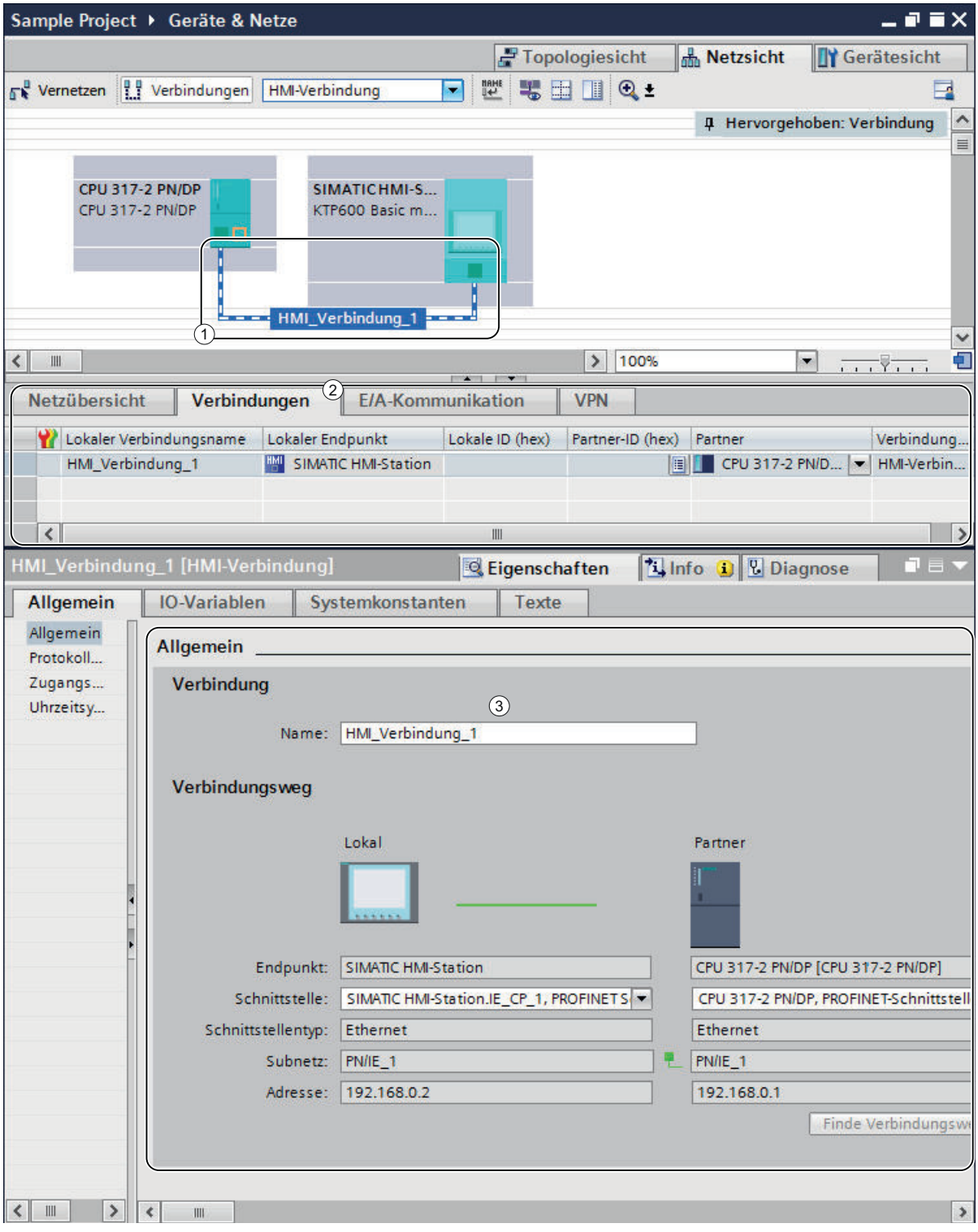

① Eine integrierte HMI-Verbindung ist angelegt und wird in der Netzsicht hervorgehoben dargestellt.

② Die Verbindung wird in der Verbindungstabelle der Komponenten angezeigt.
③ Die Verbindung kann in den Verbindungseigenschaften bearbeitet werden.

Siehe auch

HMI-Variablen neu verknüpfen (Seite 149)

2.9.5 HMI-Variablen neu verknüpfen

Nachdem Sie eine neue HMI-Verbindung zwischen CPU und HMI-Gerät aufgebaut haben, müssen Sie der neuen Verbindung die bestehenden HMI-Variablen zuordnen. Führen Sie die folgenden Schritte für jede Zeile in der betroffenen Variablentabelle durch.

Vorgehen

Um die HMI-Variablen neu zu verknüpfen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Projektnavigation zu den HMI-Variablen und doppelklicken Sie auf die entsprechende Variablentabelle, um diese im Arbeitsbereich anzuzeigen.
   Die Variablentabelle wird geöffnet.

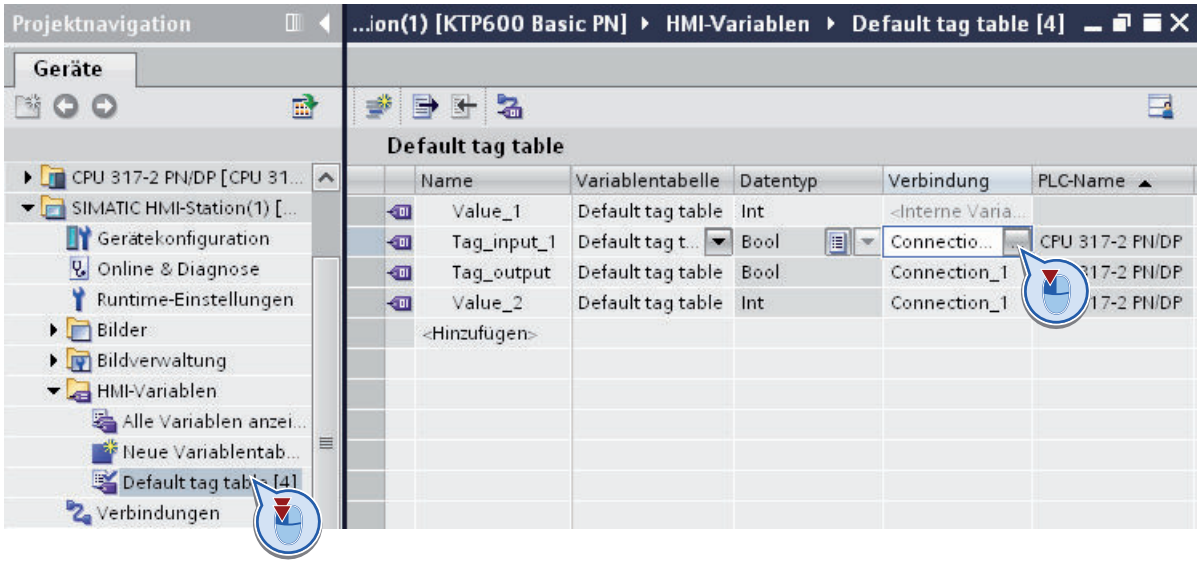

2. Klicken Sie auf die Schaltfläche "..." in der Spalte "Verbindung".
   Ein Dialog zur Auswahl der Verbindung wird geöffnet.

3. Selektieren Sie die neu angelegte HMI-Verbindung.

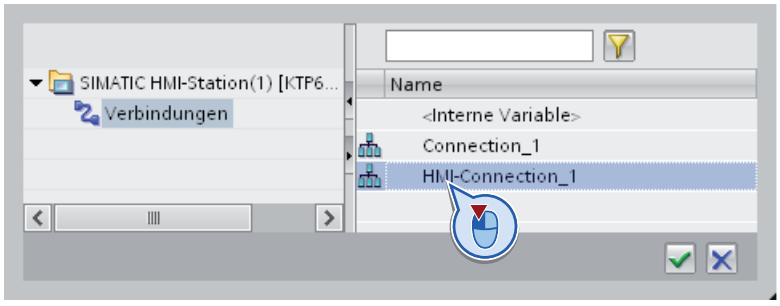

4. Klicken Sie auf die Schaltfläche "✓", um die ausgewählte Verbindung zu übernehmen.
5. Klicken Sie in der Funktionsleiste auf die "Schaltfläche PLC-Variable neu verbinden."

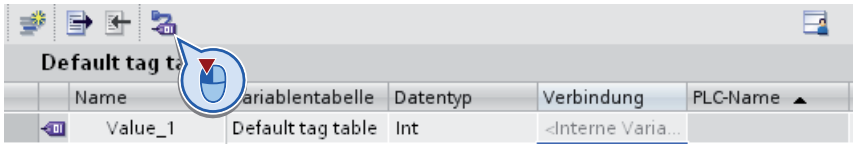

Siehe auch

Nicht integrierte HMI-Verbindung löschen (Seite 150)

2.9.6 Nicht integrierte HMI-Verbindung löschen

Zuletzt können Sie nicht integrierte HMI-Verbindungen, die noch aus dem Ausgangsprojekt verblieben sind, entfernen.

Vorgehen

Um die nicht integrierten HMI-Verbindungen zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation das HMI-Gerät und doppelklicken Sie auf den Eintrag "Verbindungen".
   Die Verbindungstabelle wird geöffnet.

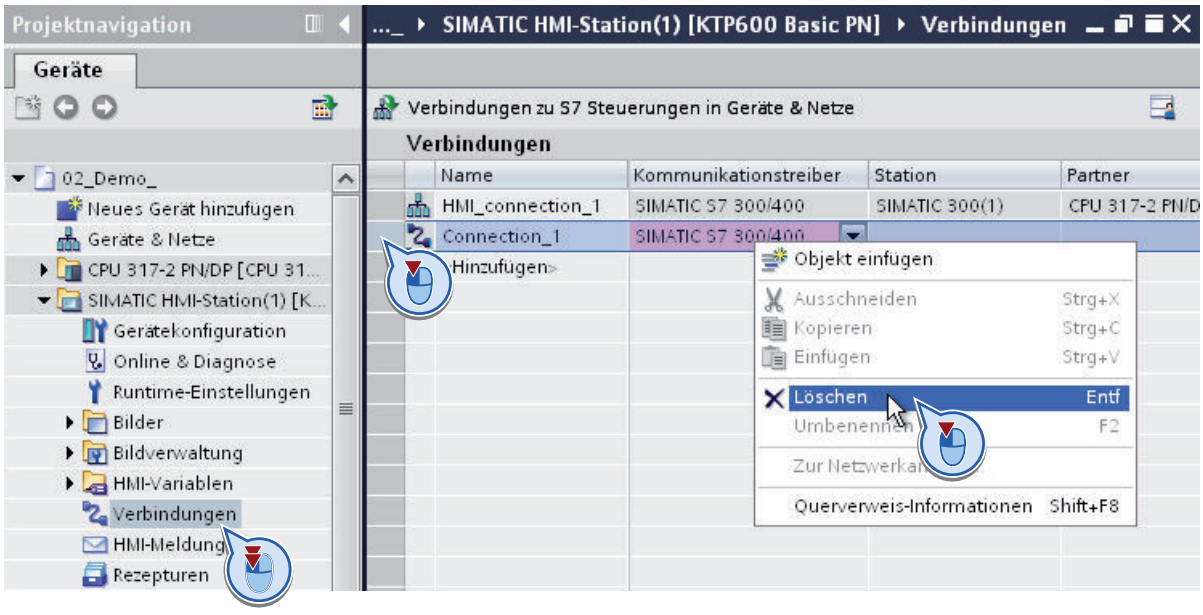

2. Selektieren Sie die Zeile mit der alten Verbindung in der Tabelle.
3. Wählen Sie den Befehl "Löschen" im Kontextmenü der Verbindungszeile.
4. Führen Sie die vorangegangenen Schritte für alle nicht integrierten HMI-Verbindungen des Ausgangsprojekts durch.

PLC-Programme in eine S7-1500-CPU / ET 200SP migrieren

3

3.1 Grundlagen zur Migration von PLC-Programmen

Einführung

Sie können PLC-Programme innerhalb des TIA Portals migrieren. Die Migration innerhalb des TIA Portals wird "PLC-Migration" genannt. Bei dieser Migration wird die neue Baugruppe im Projekt angelegt und das bestehende PLC-Programm in die neue Baugruppe kopiert. Die alte Baugruppe bleibt unverändert im Projekt bestehen.

Der Vorteil des Migrierens im Vergleich zum Kopieren besteht darin, dass das Programm automatisch so weit wie möglich an die neue CPU-Familie angepasst wird. Programmstrukturen, die nicht mehr zulässig sind, werden aktualisiert. Anweisungen, die auf der neuen Baugruppe nicht verfügbar sind, werden durch entsprechende Anweisungen der S7-1500 ersetzt.

Die folgende Tabelle gibt einen Überblick über die migrierbaren Baugruppen:

Ausgangsbaugruppe	Zielbaugruppe
S7-300/400 CPU	S7-1500 CPU
ET 200S CPU	ET200SP CPU
ET 200S CPU	S7-1500 CPU
ET 200S CPU	ET 200pro CPU
IM 154-8 CPU	CPU 1516-2 PN

Die Migration nach S7-1500 und die Migration nach ET 200SP laufen nach den gleichen Regeln ab. Die PLC-Programme werden bei beiden Migrationsvorgängen gleich behandelt.

Migrationsumfang

Bei der PLC-Migration werden alle Bestandteile des PLC-Programms in die neu angelegte Baugruppe kopiert. Dazu gehören beispielsweise die folgenden Objekte:

- Programmbausteine, deren Erstellsprache von S7-1500 unterstützt wird, und die keinen Know-how-Schutz tragen.
- PLC-Variablentabellen
- Beobachtungs- und Forcetabellen
- PLC-Datentypen
- Technologieobjekte
- Benutzerdefinierte Gruppen in der Projektnavigation

Die Hardware-Konfiguration der Ausgangs-Baugruppe wird nicht automatisch übertragen.

⚠ WARNUNG

Personen- und Sachschaden verhindern

Während der PLC-Migration werden in manchen Fällen Änderungen am Programm vorgenommen. Es ist deshalb unbedingt erforderlich, das Programm nach der Migration ausführlich in einer Testumgebung zu überprüfen, bevor Sie es in den laufenden Betrieb übernehmen.

Hinweis

Weitere Unterstützung zur PLC-Migration

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur PLC-Migration:

Migration Gesamtsysteme:

http://support.automation.siemens.com/WW/view/de/83558085 (http://support.automation.siemens.com/WW/view/de/83558085)

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Migration Visualisierung:

http://support.automation.siemens.com/WW/view/de/76878921 (http://support.automation.siemens.com/WW/view/de/76878921)

Migration Kommunikation:

http://support.automation.siemens.com/WW/view/de/83558087 (http://support.automation.siemens.com/WW/view/de/83558087)

FAQs zur Migration:

http://support.automation.siemens.com/WW/view/de/67858106 (http://support.automation.siemens.com/WW/view/de/67858106)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Hinweis

Weitere Informationen zum TIA Portal

Weitere Informationen rund um das TIA Portal finden Sie im TIA Portal im Siemens Industry Online Support:

http://support.automation.siemens.com/WW/view/de/65601780 (http://support.automation.siemens.com/WW/view/de/65601780)

Dort erhalten Sie auch Informationen zu Trainingsangeboten. Für Kenner der klassischen SIMATIC S7-Welt, die effizient auf SIMATIC S7-1500 mit dem TIA Portal umsteigen wollen, empfehlen wir das Training "TIA-SYSUP - SIMATIC TIA Portal System-Umsteigerkurs auf SIMATIC S7-1500".

3.2 Migration durchführen

Voraussetzung

- Ein konsistentes Ausgangsprogramm liegt vor.
- Alle Bausteine sind aktuell übersetzt. Dies gilt auch für Bausteine, die im Projekt enthalten sind, aber während der Programmbearbeitung nicht aufgerufen werden.
- Das Programm wurde mit TIA Portal V12 oder höher erstellt oder auf diese Version hochgerüstet.

Migration vorbereiten

Um das Programm für die Migration vorzubereiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation das Gerät, das das Ausgangsprogramm enthält.
2. Öffnen Sie den Ordner "Programmbausteine" und prüfen Sie, ob darin know-how-geschützte Bausteine enthalten sind.
3. Entfernen Sie den Know-how-Schutz von den Bausteinen.
4. Öffnen Sie den Ordner "Programmbausteine > Systembausteine" und prüfen Sie, ob darin Bausteine mit dem Namenszusatz "_LF" (legacy functions) enthalten sind.
   Es handelt sich dabei um Bibliotheksbausteine aus STEP 7, die im TIA Portal nicht mehr unterstützt werden. Bei der PLC-Migration werden diese Bausteine nicht in das neue Gerät übernommen, da sie know-how-geschützt sind.
5. Ersetzen Sie die Bausteine wenn möglich durch eine Anweisung aus der Task Card "Anweisungen".
6. Wenn das Programm Anweisungen zur Meldungsprojektierung enthält, folgen Sie der Anleitung zur Migration von Meldungen mit Begleitwerten.
   Siehe auch: Migration von Meldungen mit Begleitwerten (Seite 183)

Vorgehen

Um ein PLC-Programm innerhalb des TIA Portals zu migrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation das Gerät, das das Ausgangsprogramm enthält.
2. Öffnen Sie die Gerätekonfiguration.
3. Markieren Sie in der Geräte- oder in der Netzsicht die Baugruppe, die das Ausgangsprogramm enthält.
4. Wählen Sie im Kontextmenü den Befehl "Migrieren auf S7-1500".
   Der Dialog "Migrieren auf S7-1500" öffnet sich.
5. Wählen Sie im Bereich "Neues Gerät" das Gerät aus, in das Sie Ihr Programm migrieren möchten, und bestätigen Sie mit "OK".
   Ein Sicherheitshinweis erscheint, der Sie darüber informiert, dass das Programm während der Migration geändert wird.

6. Bestätigen Sie den Sicherheitshinweis.
   Ein Dialog erscheint, in dem Sie angeben können, welche Anweisungen zur Punkt-zu-Punkt-Kommunikation Sie im migrierten Programm verwenden wollen.
7. Wählen Sie den gewünschten Satz von Anweisungen aus.
   – Wählen Sie die Option "Neue PtP-Anweisungen für die integrierten Kommunikationsmodule der S7-1500 verwenden", um die neuen PtP-Anweisungen der S7-1500 zu verwenden.
   – Wählen Sie die Option "Weiterhin die PtP-Anweisungen für Kommunikationsprozessoren der S7-300/400 verwenden", um weiterhin die PtP-Anweisungen der CP 300/400 oder ET200 zu verwenden.

   Auch wenn Sie die Punkt-zu-Punkt-Kommunikation nicht nutzen, müssen Sie eine der beiden Optionen auswählen. Die Auswahl hat in diesem Fall jedoch keine Rückwirkung auf das Programm.
   Das PLC-Programm wird migriert. Anschließend erscheint ein Meldung, die Sie darüber informiert, ob Fehler währende der Migration aufgetreten sind. Die Meldung enthält auch einen Link auf das Migrationsprotokoll.
8. Öffnen sie das Migrationsprotokoll. Es enthält genaue Angaben zur Migration und informiert Sie über die Programmänderungen, die Sie durchführen müssen, um Ihr Programm auf dem neuen Gerät ablauffähig zu machen.
9. Bearbeiten Sie alle Hinweise, die im Migrationsprotokoll enthalten sind.
10. Übersetzen Sie das migrierte Programm.
    Siehe auch: Migration von Programmbausteinen PtP (Seite 195)

Ergebnis

Eine neues Gerät wurde neben dem ursprünglichen Gerät in der Projektnavigation angelegt. Es enthält das migrierte PLC-Programm.

Um die volle Funktionalität der S7-1500 nutzen zu können, empfehlen wir, den optimierten Bausteinzugriff für die migrierten Bausteine zu aktivieren. Weitere Hinweise hierzu finden Sie in der Hilfe zu Bausteinen mit optimiertem Zugriff:

Bausteine mit optimiertem Zugriff

Hinweis

Beachten Sie, dass durch die PLC-Migration nur das PLC-Programm umgewandelt wird. Die Hardware-Konfiguration des Ausgangsgeräts wird nicht automatisch auf das neue Gerät übertragen. Führen Sie die Hardware-Konfiguration des neuen Geräts nach der Migration manuell durch.

3.3 Protokolldatei anzeigen

Zu jeder Migration wird eine Protokolldatei angelegt. Die Protokolldatei enthält folgende Informationen:

- Änderungen an Objekten, die während der Migration vorgenommen wurden.
- Hinweise auf notwendige Anpassungen, die Sie am Programm vornehmen müssen.

Vorgehen

Um die Protokolldatei zur Migration anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten > Protokolle". Er enthält die Protokolle aller bisher durchgeführten Migrationen.
2. Doppelklicken Sie auf das gewünschte Migrationsprotokoll.
   Das Protokoll öffnet sich.
3. Meldungen, die sich auf eine bestimmte Stelle im Programm beziehen, sind mit einem Pfeil in der Spalte "Gehe zu" gekennzeichnet. Doppelklicken Sie auf den Pfeil, um an die relevante Programmstelle zu springen.
4. Meldungen, zu denen weiterführende Informationen vorhanden sind, sind mit einem Fragezeichen in der Spalte "?" gekennzeichnet. Um nähere Informationen zur Meldung zu erhalten, klicken Sie auf das Fragezeichen.

3.4 Besonderheiten bei der Migration von PLC-Programmen

3.4.1 Hinweise zur Migration von PLC-Programmen

3.4.1.1 Migration von Organisationsbausteinen

Organisationsbausteine

Bei der Migration von Organisationsbausteinen gelten folgende Regeln:

- Bausteinname und -nummer werden unverändert übernommen.
- Auch die Bausteinschnittstelle wird unverändert übernommen.
- Dem OB wird das zum OB-Typ passende Ereignis zugeordnet.
- Die OB-Parameter, z. B. die Priorität, werden unverändert übernommen. Wenn in der neuen CPU zusätzliche Parameter vergebbar sind, erhalten diese die voreingestellten Werte.

Einige Organisationsbausteine (OB) werden von S7-1500 nicht unterstützt und können deshalb nicht migriert werden. Die folgende Tabelle gibt eine Übersicht über diese Organisationsbausteine und enthält Hinweise, wie Sie die Funktionalität in Ihrem Programm nachbilden können:

Nicht migrierbarer OB	Hinweise
OB60 (Multicomputing-alarm)	Der OB kann nicht ersetzt werden. Multicomputing ist auf S7-1500 nicht verfügbar.
OB65 (Technologiesyn-chronalarm)	Prüfen Sie, ob der OB in Ihrem Programm durch den MC-Servo-OB oder den MC-Interpolator-OB ersetzbar ist.
OB7x (Redundanzfeh-ler)	Der OB kann nicht ersetzt werden. Redundanz ist auf S7-1500 nicht verfügbar.
OB81 (Stromversor-gungsfehler)	Prüfen Sie, ob der OB in Ihrem Programm durch den Diagnosealarm-OB ersetzbar ist.
OB84 (CPU-Hardware-fehler)	Prüfen Sie, ob der OB in Ihrem Programm durch den Diagnosealarm-OB ersetzbar ist.
OB85 (Programma-blauffehler)	Prüfen Sie, ob der OB in Ihrem Programm durch den Ziehen/Stecken-OB oder den Baugruppent-rägerfehler-OB ersetzbar ist.
OB87 (Kommunikati-onsfehler)	Prüfen Sie, ob der OB in Ihrem Programm durch den Diagnosealarm-OB ersetzbar ist.
OB88 (Bearbeitungsab-bruch)	Prüfen Sie, ob der OB in Ihrem Programm durch den Programmierfehler-OB ersetzbar ist.
OB90 (Hintergrundzyk-lus)	Der OB kann nicht ersetzt werden. Hintergrundbearbeitung ist auf S7-1500 nicht verfügbar.
OB101 (Wiederanlauf)	Der OB kann nicht ersetzt werden. Wiederanlauf ist auf S7-1500 nicht verfügbar.
OB102 (Kaltstart)	Prüfen Sie, ob der OB in Ihrem Programm durch den Anlauf-OB ersetzbar ist.

Siehe auch

FAQs zur Migration (http://support.automation.siemens.com/WW/view/de/67858106)

3.4.1.2 Migration von Hardware-Kennungen

Adressierung von Baugruppen über Hardware-Kennungen

Wenn Sie in Ihrem Programm Hardwaremodule adressieren, z. B. am Parameter "LADDR" oder "ID", sind diese nach der Migration auf die neue Hardware nicht mehr gültig. Die Adressen müssen ersetzt werden. Die Hardwaremodule der S7-1500 werden durch Hardware-Kennungen adressiert. Tragen Sie deshalb nach der Migration am Parameter "LADDR" oder "ID" die Hardware-Kennung des neuen Moduls ein.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration.
2. Markieren Sie das Modul, das Sie adressieren wollen.
3. Wählen Sie im Inspektorfenster das Register "Eigenschaften > Systemkonstanten". Die Tabelle enthält Konstanten für alle verwendeten Module mit den benötigten HW-Kennungen.
4. Markieren Sie die Konstante für das zu adressierende Modul und wählen Sie im Kontextmenü den Befehl "Kopieren".
5. Fügen Sie die Konstante am Parameter "LADDR" oder "ID" der migrierten Anweisung ein.

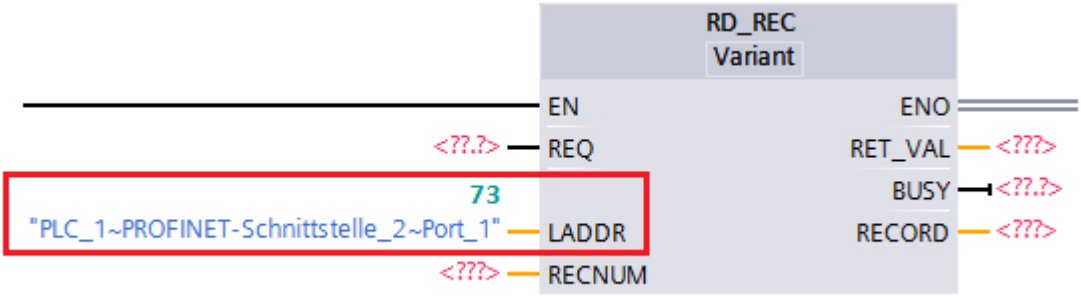

3.4.1.3 Migration von IEC-Zeiten und IEC-Zählern

Aktualisierung von IEC-Zeiten

Die Ausgänge "Q" und "ET" der IEC-Zeiten TP, TON und TOF werden in S7-300/400 zu dem Zeitpunkt bestimmt, an dem das Programm die IEC-Zeit ausführt. Danach bleibt der Zustand von "Q" und "ET" unverändert. Wenn Sie mehrmals im Programm auf diese Ausgänge zugreifen, erhalten Sie also immer denselben Wert.

Dieses Verhalten hat sich in S7-1500 geändert: Das Programm prüft hier bei jedem Zugriff den aktuellen Wert der Zeit und bestimmt die Ausgänge "Q" und "ET" erneut.

Nach der Migration kann sich das Verhalten Ihres Programms also ändern, falls Sie mehrfach auf "Q" oder "ET" zugreifen.

Damit das Programm das gleiche Verhalten zeigt wie in S7-300/400, können Sie den Wert von "Q" bzw. "ET" einer Variablen zuweisen. Im Programm referenzieren Sie dann diese Variable anstelle des Ausgangs.

IEC-Zeiten und IEC-Zähler mit EN/ENO-Beschaltung

Die Anweisungen wurden für S7-1500 an die Norm IEC 1131-3 angepasst. Sie sind nun über den Parameter "IN" mit dem Strompfad verbunden und verfügen deshalb nicht mehr über die Parameter "EN" und "ENO". Die Migration meldet einen Fehler, wenn das Ausgangsprogramm einen IEC-Zeiten oder -Zähler mit Vorverknüpfung enthält, die über den Eingangsparameter "EN" ausgewertet wird.

Fügen Sie zwischen der Vorverknüpfung und dem IEC-Zeiten oder -Zähler eine Sprunganweisung ein, um die IEC-Zeit oder den IEC-Zähler abhängig vom VKE aufzurufen. Folgende Sprunganweisungen stehen zur Verfügung:

- ---(JMP): Springen bei VKE = 1
- ---(JMPN): Springen bei VKE = 0

IEC-Zähler: Neue Parameter für den Zählerstatus in der Bausteinschnittstelle

Die Bausteinschnittstelle enthält nach der Migration anstelle des Ausgangsparameters "Q" nun die beiden Parameter "QU" und "QD" für den Zählerstatus eines IEC-Zählers. Abhängig von der Art des IEC-Zählers wird nur einer der beiden Parameter ausgelesen. Der jeweils andere Parameter wird nicht genutzt.

Falls Sie im Programmcode auf den Parameter "Q" zugegriffen haben, müssen Sie den Zugriff nach der Migration manuell anpassen. Verwenden Sie "QU", wenn Sie vorwärts zählen und "QD", wenn Sie rückwärts zählen.

3.4.1.4 Migration von CPU-Datenbausteinen

CPU-Datenbausteine

Bausteine, die durch die Anweisungen CREAT_DB oder CREATE_DB auf der CPU erzeugt wurden und nur online vorhanden sind, werden nicht migriert.

Siehe auch

Migration von Datenbausteinanweisungen (Seite 179)

3.4.1.5 Migration von Diagnosefunktionen

Systemzustandslisten

CPUs der Baureihe S7-1500 stellen keine Systemzustandslisten bereit. Es besteht jedoch die Möglichkeit, einige der Informationen über die Anweisungen "GET_DIAG", "Geo2Log", "DeviceStates" oder "ModuleStates" auszulesen.
Weitere Informationen zur Migration von Systemzustandslisten finden Sie unter: Migration der Anweisung RDSYSST (Seite 181)

Systemdiagnose mit "Report System Errors"

Die CPUs der Baureihe S7-1500 verfügen über eine integrierte Systemdiagnose. "Report System Errors" wird somit nicht mehr unterstützt. Wenn das Programm Bausteine für "Report System Errors" enthält, deaktivieren Sie im Ausgangsprogramm die Systemdiagnose vor der Migration und übersetzen Sie das Programm erneut. Die betroffenen Bausteine werden dabei gelöscht.

Die Systemdiagnose der S7-1500-CPUs ist per Voreinstellung aktiviert. Sie müssen hier keine weiteren Einstellungen vornehmen.
Um die Einstellungen zur Systemdiagnose zu ändern, wählen Sie in der Gerätekonfiguration die CPU an. Im Inspektorfenster werden daraufhin unter "Eigenschaften > Systemdiagnose" die Einstellungen zur Systemdiagnose angezeigt. Die Meldungen der Systemdiagnose sind aktiviert und können abgewählt werden, falls sie nicht benötigt werden.

PROFINET-/ PROFIBUS-Diagnose

Bei der Migration der PROFINET- / PROFIBUS-Diagnose gelten folgende Regeln:

- "Systemfehler Melden" wird durch die integrierte Systemdiagnose ersetzt.
- Der FB 126 wird nicht unterstützt und kann deshalb nicht migriert werden.
- Für die applikative Auswertung können Sie die Anweisungen "DeviceStates" und "ModuleStates" verwenden.

3.4.1.6 Migration von Absolutzugriffen auf Lokaldaten

Absolutzugriffe auf Lokaldaten

In S7-300/400 war es möglich, absolut auf die Lokaldaten eines Bausteins zuzugreifen, ohne ein Symbol für die Adresse zu deklarieren.

Da der rein absolute Zugriff in S7-1500 nicht mehr unterstützt wird, behandelt die Migration diese wie folgt:

- Ist für einen Bereich innerhalb der Lokaldaten eine temporäre Variable in der Bausteinschnittstelle deklariert, wird die absolute Adresse in eine symbolische Adresse umgewandelt.
- Existiert keine passende temporäre Variable, bleibt die absolute Adresse erhalten. Da während der Migration jedoch unter Umständen weitere lokale Variablen angelegt werden, können sich Adressen innerhalb der Lokaldaten verschieben. Prüfen Sie deshalb unbedingt, ob die von Ihnen verwendete absolute Adresse noch korrekt ist und das richtige Datum adressiert wird. Legen Sie wenn möglich eine Variable im Abschnitt "Temp" an und verwenden Sie diese für die Adressierung. Um Bereiche innerhalb deklarierter Variablen gezielt zu adressieren, können Sie bestehende Variablen mit AT überlagern oder einzelne Bereiche mit der Syntax .X, .B, .W oder .D adressieren.

Beispiele

Die folgenden Beispiele zeigen die Migration von Absolutzugriffen auf Lokaldaten.

In der ersten Tabelle sehen Sie einen Programmausschnitt vor der Migration:

AWL
`L %LW20`
`L %LW3`
`+I`
`T %LW5`

In der zweiten Tabelle sehen Sie den Programmausschnitt nach der Migration:

AWL
`L "MyTempVar1"`
`L "MyTempVar2"`
`+I`
`T %LW5`

- Für "LW20" und "LW3" waren bereits temporäre Variablen deklariert. Nach der Migration werden diese im Programm verwendet.
- Für "LW5" war keine Variable deklariert. Sie müssen nach der Migration prüfen, ob der Zugriff noch korrekt ist.

3.4.1.7 Bausteinparameter in S7-1500

Verwenden von Bausteinparametern

Für die Verwendung der Bausteinparameter innerhalb des Bausteins gelten in S7-1500 die folgenden Regeln:

- Eingangsparameter (Input) dürfen nur gelesen werden.
- Ausgangsparameter (Output) dürfen nur geschrieben werden.
- Durchgangsparameter (InOut) dürfen gelesen und geschrieben werden.
- Funktionswerte (Ret_Val) dürfen nur geschrieben werden.

Wenn Ihr Programm diesen Regeln nicht entspricht, wird bei der Übersetzung eine Warnung gemeldet. Wandeln Sie in diesem Fall den betroffenen Ein- oder Ausgangsparameter in einen Durchgangsparameter um.

Für die Parameterversorgung von Funktionen (FC) gelten weitere Regeln.

Siehe auch:

Parameterversorgung von Funktionen

Bausteinparameter als Flankenmerker in Funktionen (FC)

Auch bei der Programmierung von Flankenauswertungen in Funktionen (FC) müssen Sie die oben genannten Regeln zur Parameterversorgung beachten:

Als Flankenmerker benötigen Sie einen Datenwert, der sowohl gelesen als auch geschrieben werden darf und der über mehrere Zyklen hinweg erhalten bleibt. Da Eingangsparameter (Input) nur gelesen und Ausgangsparameter (Output) nur geschrieben werden dürfen, können Sie diese nicht als Flankenmerker einsetzen. Auch temporäre Lokaldaten (Temp) sind nicht als Flankenmerker verwendbar, da sie nur für einen Zyklus zur Verfügung stehen.

Verwenden Sie in Funktionen (FC) deshalb nur Durchgangsparameter (InOut) als Flankenmerker. Sie können auch ein einzelnes Bit eines Durchgangsparameters als Flankenmerker verwenden, wenn Sie dieses über einen Slice-Zugriff adressieren.

Das folgende Beispiel zeigt, wie Sie einen Durchgangsparameter als Flankenmerker einsetzen können.

AWL

```
FP #InOutFlagStore.x0
```

"InOutFlagStore" ist ein Durchgangsparameter vom Datentyp BYTE. Die Adresse "#InOutFlagStore.x0" adressiert die Bitadresse 0, die als Flankenmerker dient.

Automatische Initialisierung von Bausteinparametern in Funktionen (FC)

In S7-300/400 müssen temporäre Lokaldaten (Temp) einer Funktion zwingend mit einem Wert vorbelegt werden. Andernfalls konnte es vorkommen, dass das Programm mit undefinierten Werten arbeitet.

In S7-1500 ist die Gefahr des Weiterverarbeitens undefinierter Werte deutlich reduziert, da folgende Parameter beim Bausteinaufruf automatisch initialisiert werden:

- Temporäre Daten vom Datentyp STRING und WSTRING werden grundsätzlich mit der Maximallänge 254 und der tatsächlichen Länge 0 vorbelegt.
- Temporäre Daten mit elementarem Datentyp werden in Funktionen (FC) mit optimiertem Zugriff automatisch initialisiert. Sie werden dabei mit dem Wert belegt, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Elemente von PLC-Datentypen werden mit dem Defaultwert vorbelegt, der in der Deklaration des PLC-Datentyps (UDT) angegeben ist. Elemente vom Datentyp ARRAY, STRING oder WSTRING werden mit dem Wert "0" vorbelegt, auch wenn sie innerhalb eines PLC-Datentyps verwendet werden.

Ausgangsparameter (Output) in Funktionen mussten in S7-300/400 zwingend mit einem Wert beschrieben werden, da es ansonsten zu ungewollten Reaktionen im Programm kommen konnte. Beim Einsatz von Sprunganweisungen oder VKE-abhängigen Anweisungen musste jeder mögliche Programmpfad daraufhin geprüft werden.

In S7-1500 ist die Gefahr undefinierter Ausgangsparameter deutlich reduziert, da Ausgangsparameter elementaren Datentyps beim Bausteinaufruf automatisch initialisiert werden. Sie werden dabei mit dem Wert vorbelegt, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Strukturierte Ausgangsparameter werden jedoch nicht mit einem Wert vorbelegt. Sie werden beim Bausteinaufruf als Pointer übergeben und können somit nicht undefiniert sein.

3.4.1.8 Multiinstanzfähigkeit in S7-1500

Multiinstanzfähigkeit

In CPUs der Baureihe S7-1500 sind alle Funktionsbausteine multiinstanzfähig. Durch die Migration werden deshalb FBs, die im Ausgangsprogramm als nicht multiinstanzfähig gekennzeichnet waren, in multiinstanzfähige Bausteine umgewandelt. Es wird für jeden umgewandelten FB ein Hinweis im Migrationsprotokoll ausgegeben. Wenn Sie den FB nach der Migration weiterhin als Einzelinstanz verwenden, funktioniert Ihr Programm unverändert.

3.4.1.9 Gleitpunktzahlen in S7-1500

Ungültige Gleitpunktzahlen

Die Behandlung ungültiger Gleitpunktzahlen in S7-1500 unterscheidet sich von S7-300/400. Beispiel:

- S7-300/400: Das Ergebnis des Ausdrucks "ungültige Gleitpunktzahl <> 1.0" ist FALSE.
- S7-1500: Das Ergebnis des Ausdrucks "ungültige Gleitpunktzahl <> 1.0" ist TRUE.

Beachten Sie, dass die Anweisungen in Ihrem Programm aufgrund dieses Unterschieds andere Ergebnisse liefern können.

3.4.1.10 Schreiben einzelner Zeichen eines STRING in S7-1500

Einzelne Zeichen eines STRING schreiben

Wenn ein Zeichen oder ein Byte innerhalb eines STRING geschrieben werden soll, prüft eine CPU der Baureihe S7-1500, ob die Zieladresse innerhalb der tatsächlichen Länge des STRINGs liegt. Ansonsten wird das Zeichen oder Byte nicht geschrieben. Eine Ausnahme besteht nur, wenn das Zeichen direkt hinter die tatsächliche Länge des STRINGs geschrieben wird.

Das folgende Beispiel zeigt den STRING 'hello' mit der tatsächlichen Länge 5. Das 27. Zeichen des STRINGs liegt außerhalb der tatsächlichen Länge und kann deshalb nicht geschrieben werden. Der STRING bleibt unverändert, das Ergebnis der Zuweisung ist 'hello'.

SCL

```
MyDB.mystring := 'hello';
MyDB.mystring[27] :=
CHAR_TO_BYTE('!');
```

Das folgende Beispiel zeigt den genannten Ausnahmefall: Das Zeichen wird direkt hinter den STRING in das 6. Zeichen geschrieben. Das Ergebnis der Zuweisung ist 'hello!'.

SCL

```
MyDB.mystring := 'hello';
MyDB.mystring[6] :=
CHAR_TO_BYTE('!');
```

Verwenden Sie zur Hantierung von STRINGs wenn möglich die Anweisungen aus der Palette "Erweiterte Anweisungen > String + Char".

SCL

```
CONCAT(IN1 := 'hello', IN2 := '!');
```

3.4.1.11 Zugriffe auf das Statuswort in S7-1500

Statuswort

Das Statuswort wird in S7-1500 nicht mehr in vollem Umfang unterstützt. Es enthält eine geringere Anzahl an Informationen und ist nur noch in AWL auswertbar. KOP und FUP unterstützen die Auswertung des Statusworts nicht mehr. Hinweise zur Migration von Statuswortzugriffen in den einzelnen Programmiersprachen erhalten Sie in den nachfolgenden Kapiteln.

Siehe auch

Migration von KOP/FUP-Programmen nach S7-1500 (Seite 197)

Migration von AWL-Programmen nach S7-1500 (Seite 201)

3.4.1.12 Laden von Softwareänderungen in S7-1500

Laden von Softwareänderungen

CPUs der Baureihe S7-1500 verhalten sich beim Laden von Softwareänderungen anders als CPUs der Baureihen S7-300/400. Sie bieten zum Beispiel die Möglichkeit, im STOP und im RUN Änderungen zu laden, ohne die Aktualwerte bereits geladener Variablen zu beeinflussen.

3.4.2 Migration von Anweisungen nach S7-1500

3.4.2.1 Hinweise zur Migration von Anweisungen

Automatische Migration von Anweisungen

Während der Migration werden die im Ausgangsprogramm verwendeten Anweisungen in das neue Programm übernommen. Dabei werden notwendige Anpassungen soweit möglich automatisch durchgeführt, ohne die Semantik des Ausgangsprogramms zu verändern. Wenn mehrere Versionen einer Anweisung existieren, verwendet die Migration immer die neueste Anweisungsversion.

Anweisungen, die in S7-1500 nicht verfügbar sind, werden, soweit möglich, automatisch durch eine kompatible oder ähnliche Anweisung ersetzt.

Migrationsszenarien

Die folgende Tabelle zeigt die möglichen Migrationsszenarien für Anweisungen:

Klasse	Szenario	Systemreaktion	Manuelle Anpassung des Programms
1	Die Anweisung ist in beiden CPU-Familien identisch.	Die Anweisung wird migriert.	Keine Anpassung erforderlich. Das migrierte Programm ist semantisch unverändert und lässt sich übersetzen.
2	In S7-1500 gibt es die Anweisung nicht mehr, es existiert aber eine kompatible, neuere Anweisung.	Die Anweisung wird durch die neue Anweisung ersetzt. Möglicherweise werden weitere Anweisungen eingefügt, die notwendig sind, um die Semantik des Ausgangsprogramms zu erhalten. Das Migrationsprotokoll meldet die Ersetzung.	Keine Anpassung erforderlich. Das migrierte Programm ist semantisch unverändert und lässt sich übersetzen.
3	In S7-1500 gibt es die Anweisung nicht mehr, es existiert aber eine ähnliche Anweisung.	Die Anweisung wird durch die ähnliche Anweisung ersetzt. Das Migrationsprotokoll meldet die Ersetzung. Zu prüfende Stellen sind im Programm markiert. Sie erhalten Hinweise zu notwendigen Anpassungen als Kommentar direkt an der entsprechenden Programmstelle oder beim Übersetzen.	Sie müssen das Programm prüfen und möglicherweise anpassen.
4	In S7-1500 gibt es die Anweisung nicht mehr, es existiert auch keine ähnliche Anweisung.	Die Anweisung wird rot dargestellt. Das Programm lässt sich nicht übersetzen.	Sie müssen das Programm prüfen und möglicherweise anpassen.

Hinweise zu Anweisungen der Migrationsklasse 3

Einige Anweisungen, die in S7-300/400 zur Verfügung standen, werden in S7-1500 nicht mehr angeboten, da hier einfachere oder performantere Methoden zur Lösung bestimmter Aufgaben verfügbar sind. Die PLC-Migration unterstützt Sie bei der Migration, indem diese Anweisungen durch geeignete neue Anweisungen ersetzt werden. Die Anweisungen aus Fall 3 sind jedoch nicht komplett automatisch migrierbar, da zum Beispiel neue Formalparameter hinzugekommen sind. Die Stellen, die nicht automatisch migrierbar sind, werden im Programm markiert. Sie erhalten Hinweise zu notwendigen Anpassungen als Kommentar direkt an der entsprechenden Programmstelle, im Migrationsprotokoll oder beim Übersetzen.

Folgende manuelle Anpassungen können nötig sein:

- Wenn die neue Anweisung zusätzliche Formalparameter hat, versorgen Sie diese Parameter mit geeigneten Aktualparametern.
- Wenn die neue Anweisung geänderte Parameternamen hat, ändern Sie die Parameterversorgung.
- Wenn die Migration nicht eindeutig sicherstellen kann, dass die Parameterversorgung mit dem Ausgangsprogramm semantisch identisch ist, werden die zu prüfenden Aktualparameter gekennzeichnet. Prüfen Sie die Semantik dieser Programmstellen nach der Migration und korrigieren Sie sie wenn nötig.
- Wenn das Ausgangsprogramm HW-Kennungen enthält, z. B. am Parameter "LADDR", werden diese ebenfalls gekennzeichnet. Tragen Sie die neue HW-Kennung am Parameter "LADDR" ein.
- Wenn die neue Anweisung geänderte Fehlercodes hat, passen Sie die Fehlerbehandlung im Programm an.

Einige Beispiele für Anweisungen der Migrationsklasse 3 finden Sie im Kapitel "Migration von Datenbausteinanweisungen (Seite 179)".

Hinweise zu Anweisungen der Migrationsklasse 4

Einige Anweisungen werden in S7-1500 nicht mehr angeboten, da sich die Funktionalität von Grund auf geändert hat. Diese Anweisungen werden durch die Migration nicht ersetzt. Sie werden im Programm rot dargestellt und führen dazu, dass das Programm sich nicht übersetzen lässt. Sie erhalten Hinweise zu notwendigen Anpassungen als Kommentar direkt an der entsprechenden Programmstelle, im Migrationsprotokoll oder beim Übersetzen.

Eine Übersicht über die nicht migrierbaren Anweisungen der Migrationsklasse 4 sowie Informationen zu alternativen Lösungen finden Sie in den nachfolgenden Kapiteln.

Hinweis

Weitere Unterstützung

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur PLC-Migration:

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Falls Sie weitere Unterstützung zur PLC-Migration benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Siehe auch

FAQs zur Migration (http://support.automation.siemens.com/WW/view/de/67858106)

3.4.2.2 Übersicht der nicht migrierbaren Anweisungen

Einführung

Einige Anweisungen aus S7-300/400 lassen sich nicht nach S7-1500 migrieren. Dies ist z. B. der Fall, wenn die Anweisungen nur in Zusammenhang mit bestimmten Baugruppen verwendbar sind oder wenn die Funktionalität der Anweisung in S7-1500 anders implementiert werden muss.

Hinweise zur Ersetzung nicht migrierbarer Anweisung (Klasse 4)

Nachfolgend finden Sie Beispiele für nicht migrierbare Anweisungen und Hinweise zu deren Ersetzung:

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Weitere Anweisungen	SETP	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "SET_BF" ersetzbar ist.
	RESETP	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RESET_BF" ersetzbar ist.
	DRUM_X	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "DRUM" ersetzbar ist.
	TONR_X	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "TONR" ersetzbar ist.
	RESET	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RESET_BF" ersetzbar ist.
	SET	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "SET_BF" ersetzbar ist.
	WSR	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	SHRB	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	RESETI	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RESET_BF" ersetzbar ist.
	SETI	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "SET_BF" ersetzbar ist.
Laufzeitsteuerung	COMPRESS	Löschen Sie die Anweisung. In S7-1500 ist es nicht mehr notwendig, den Arbeitsspeicher der CPU oder den Ladespeicher zu komprimieren.
	PROTECT	Projektieren Sie Schutzstufe und Passwort über das Frontpanel der S7-1500 CPU oder in den Baugruppeneigenschaften der CPU.
	CIR	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Datum und Uhrzeit	SET_CLKS	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "WR_SYS_T" ersetzbar ist.
	LOC_TIME	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RD_SYS_T" ersetzbar ist.
	BT_LT	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RD_SYS_T" ersetzbar ist.
	LT_BT	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RD_SYS_T" ersetzbar ist.
	S_LTINT	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "SET_TINTL" ersetzbar ist.
	SET_SW	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "WR_SYS_T" oder "RD_SYS_T" ersetzbar ist.
	SET_SW_S	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "WR_SYS_T" oder "RD_SYS_T" ersetzbar ist.
	TIMESTMP	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "WR_SYS_T" oder "RD_SYS_T" ersetzbar ist.
	WS_RULES	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
Dezentrale Peripherie	D_PRAL	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	ASi_3422	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	SALRM	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
Modulparametrierung	PARM_MOD	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
Alarme	MP_ALM	Multicomputing ist auf S7-1500 nicht verfügbar.
	REPL_VAL	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Meldungen	EN_MSG	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	DIS_MSG	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	ALARM_SQ	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	ALARM_S	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	ALARM_SC	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	WR_USMSG	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	READ_SI	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	DEL_SI	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	ALARM_DQ	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	ALARM_D	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	NOTIFY_8P	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	ALARM	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
	ALARM_8	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	ALARM_8P	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	NOTIFY	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
	AR_SEND	Püfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "Program_Alarm" ersetzbar ist. Weitere Informationen erhalten Sie im Kapitel "Migration von Meldungen mit Begleitwerten (Seite 183)" und beim SIMATIC Customer Support.
Diagnose	RDSYSST	Die Anweisung "RDSYSST: Systemzustandsliste auslesen" steht in S7-1500 nicht mehr zur Verfügung, da die CPUs dieser Projektfamilie keine Systemzustandslisten bereitstellen. Es besteht jedoch die Möglichkeit, einige der Informationen über die Anweisungen "GET_DIAG", "Geo2Log", "DeviceStates" oder "ModuleStates" auszulesen. Weitere Informationen erhalten Sie im Kapitel "Migration der Anweisung RDSYSST (Seite 181)" und beim SIMATIC Customer Support.
	OB_RT	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "RD_SYS_T" ersetzbar ist.

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Tabellenfunktionen	ATT	Die Anweisungen können nicht ersetzt werden. Wenn Sie die Anweisungen in Ihrem Programm benötigen, müssen Sie sie selbst implementieren. Weitere Unterstützung erhalten Sie im Kapitel "Migration von Tabellenanweisungen (Seite 187)" und beim SIMATIC Customer Support.
	FIFO	
	TBL_FIND	
	LIFO	
	TBL	
	TBL_WRD	
	WRD_TBL	
	DEV	
	CDT	
	TBL_TBL	
	PACK	
Adressierung	LGC_GADR	Die Anweisungen können für die Baugruppen hinter Netzübergängen (z. B. IE/PB Link) nicht benutzt werden. Verwenden Sie stattdessen die Anweisungen "GEO2LOG" und "LOG2GEO". Für Baugruppen, die mit der CPU nicht über Netzübergang (z. B. IE/PB Link) verbunden sind, funktionieren die Anweisungen "LGC_GADR" und "GADR_LGC" immer noch.
	GADR_LGC	
iSlave	SET_ADDR	Die Anweisung kann nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
PID Control	TUN_EC	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch die Technologieobjekte "PID_Compact" oder "PID_3Step" ersetzbar sind.
	TUN_ES	

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Funktionsmodule	CNT_CTRL	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch das Technologieobjekt "High_Speed_Counter" in Verbindung mit den Zählermodulen "S7-1500 TM Count 2x24V" oder "TM PosInput 2" ersetzbar sind. Prüfen Sie alternativ, ob die gewünschte Funktionalität mit den Modulen "ET 200S 1Count24V" oder "1Count5V" umgesetzt werden kann.
	DIAG_INF	
	CNT_CTL1	
	CNT_CTL2	
	CNT2WRPN	
	CNT2RDPN	
	CNT2_CTR	
	DIAG_RD	
	ABS_INIT	Prüfen Sie, ob die Funktionen durch "S7-1500 Motion" oder "ET 200S 1PosU" abgelöst werden können. Prüfen Sie, ob die Funktionen mit Hilfe der Module "TM Count 2x24V" oder "TM PosInput 2" gelöst werden können.
	ABS_CTRL	
	ABS_DIAG	
	CAM_INIT	Prüfen Sie, ob die Gebererfassung über die Module "S7-1500 TM Count 2x24V" oder "TM PosInput 2" und der Verwendung der beiden DQs für schnelle Reaktionen möglich ist. Weitere DQs von Standard DQs können abhängig vom Geberwert aus dem Anwenderprogramm geschaltet werden.
	CAM_CTRL	
	CAM_DIAG	
	PID_FM	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch die Technologieobjekte "PID_Compact", "PID_3Step" oder PID Basisfunktionen ersetzbar sind.
	FUZ_355	
	FORCE355	
	READ_355	
	CH_DIAG	
	PID_PAR	
	CJ_T_PAR	
	FMT_PID	
	FMT_PAR	
	FMT_CJ_T	
	FMT_DS1	
	FMT_TUN	
	FMT_PV	
300C Funktionen	ANALOG_300C	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch die Anweisungen aus der Gruppe "Motion Control" ersetzbar sind.
	DIGITAL_300C	
	COUNT_300C	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch das Technologieobjekt "High_Speed_Counter" in Verbindung mit Zählermodulen ersetzbar sind.
	FREQUENC_300C	
	PULSE_300C	Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "CTRL_PWM" ersetzbar ist.
	SEND_RK_300C	Die Kommunikationsanweisungen der S7-1500 unterstützen RK 512 nicht. Wenn Sie RK-Funktionalität benötigen, müssen Sie sie selbst implementieren.
	FETCH_RK_300C	
	SERVE_RK_300C	

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Kommunikation mit iSlave/iDevice	I_GET	Die Anweisungen können nicht ersetzt werden. Bitte wenden Sie sich für weitere Unterstützung an den SIMATIC Customer Support.
	I_PUT	
	I_ABORT	
S7-Kommunikation	PRINT	Prüfen Sie, ob die Anweisungen in Ihrem Programm durch die Anweisungen der S7-Kommunikation für S7-1500 ersetzbar sind. Die Anweisungen "CONTROL" und "C_CNTRL" könnten z. B. durch die Anweisung "T_DIAG" ersetzbar sein.
	START	
	STOP	
	RESUME	
	STATUS	
	USTATUS	
	CONTROL	
	C_CNTRL	
Open User Communication	TCON_PAR	Diese Anweisungen sind für die Open User Communication in S7-1500 nicht erforderlich. Die Verbindungsparameter werden in den Bausteineigenschaften eingestellt.
	TADDR_PAR	
	TCP_conn_active	
	TCP_conn_passive	
	ISOonTCP_conn_active	
	SOonTCP_conn_passive	
	ISOonTCP_conn_CP_active	
	ISOonTCP_conn_CP_passive	
	UDP_local_open	
	UDP_rem_address_and_port	
Kommunikationsprozessor	Anweisungen der Gruppe SIMATIC NET-CPs Anweisungen der Gruppen PTP CP 340, PTP CP341, PTP CP 440, PTP CP 441	Ausführliche Informationen zu diesen Anweisungen erhalten Sie im Kapitel "Migration von Kommunikationsanweisungen (Seite 191)".
PROFINET / CBA	PN_IN	Die PROFINET / CBA-Kommunikation wird von S7-1500 nicht unterstützt. Verwenden Sie die S7-Komunikation oder die Open User Communication.
	PN_OUT	
	PN_DP	
MPI-Kommunikation	X_SEND	Die MPI-Kommunikation wird von S7-1500 nicht unterstützt. Verwenden Sie die S7-Komunikation oder die Open User Communication.
	X_RCV	
	X_GET	
	X_PUT	
	X_ABORT	
Globaldaten-Kommunikation	GD_SND	Die Globaldaten-Kommunikation wird von S7-1500 nicht unterstützt. Verwenden Sie die S7-Komunikation oder die Open User Communication.
	GD_RCV	

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
PtP-Kommunikation	P_PRINT	Die Druckfunktion wird von den PtP-Kommunikationsmodulen nicht unterstützt.
	P_PRINT341	Die Druckfunktion wird von den PtP-Kommunikationsmodulen nicht unterstützt.
	MODB_341	Diese Bausteine für CP 341 / CP 441 werden von S7-1500 nicht unterstützt. Verwenden Sie stattdessen die Anweisungen für MODBUS (RTU). • Modbus_Comm_Load • Modbus_Master • Modbus_Slave in Verbindung mit den Punkt-zu-Punkt-Modulen der S7-1500 • CM PtP RS232 HF • CM PtP RS422/485 HF Ausführliche Informationen zu diesen Anweisungen erhalten Sie im Kapitel "Migration von Programmbausteinen PtP (Seite 195)".
	MODB_441	
	S_MODB	Diese Anweisungen für ET 200S 1SI werden von S7-1500 nicht unterstützt. Verwenden Sie stattdessen die Anweisungen für MODBUS (RTU) bzw. für USS in Verbindung mit dem Punkt-zu-Punkt-Modul "ET 200SP CM PtP". Ausführliche Informationen zu diesen Anweisungen erhalten Sie im Kapitel "Migration von Programmbausteinen PtP (Seite 195)".
	S_USST	
	S_USSR	
	S_USSI	

Anweisungsgruppe	Nicht migrierbare Anweisung	Hinweise
Teleservice	PG_DIAL	Die MPI-Schnittstelle wird von S7-1500 nicht als Standardschnittstelle unterstützt. Die Anweisung ist nicht mehr verfügbar.
	AS_DIAL	Die MPI-Schnittstelle wird von S7-1500 nicht als Standardschnittstelle unterstützt. Die Anweisung ist nicht mehr verfügbar.
	SMS_SEND	Die MPI-Schnittstelle wird von S7-1500 nicht als Standardschnittstelle unterstützt. Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "TMAIL_C" ersetzbar ist.
	AS_MAIL	Die MPI-Schnittstelle wird von S7-1500 nicht als Standardschnittstelle unterstützt. Prüfen Sie, ob die Anweisung in Ihrem Programm durch die Anweisung "TMAIL_C" ersetzbar ist.
Anwenderbibliothek	PNIO_DIAG	Die Anweisung "PNIO_DIAG" konnte für S7-300/400 als Download in eine Anwenderbibliothek geladen und für Diagnosezwecke verwendet werden. "PNIO_DIAG" wurde in S7-1500 durch die integrierte Diagnose abgelöst und wird somit nicht mehr benötigt. In S7-1500 ist die Systemdiagnose standardmäßig aktiviert. Sie können die Systemdiagnose nutzen, ohne weitere Einstellungen vorzunehmen. Bei Bedarf können Sie die Einstellungen zur Systemdiagnose in der Gerätekonfiguration der S7-1500 im Register "Eigenschaften > Diagnose" bearbeiten. Dort können Sie auch die Alarme der Systemdiagnose projektieren. Die Alarme sind standardmäßig aktiviert. Im HMI Panel können Sie nach der Migration die in Zusammenhang mit "PNIO_DIAG" verwendete Diagnosebilder und die zugehörigen Tags löschen. Verwenden Sie stattdessen das Control "System-Diagnoseanzeige" in Ihren Diagnosebildern.

Hinweis

Unterstützung bei nicht migrierbaren Anweisungen

Falls Sie weitere Unterstützung zu nicht migrierbaren Anweisungen benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur PLC-Migration:

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Siehe auch

FAQs zur Migration (http://support.automation.siemens.com/WW/view/de/67858106)

3.4.2.3 Migration von Datenbausteinanweisungen

Vorgehen bei der Migration von Datenbausteinanweisungen (Migrationsklasse 3)

S7-1500 bietet einen geänderten Satz von Anweisungen der Kategorie "Datenbausteine". Diese Anweisungen werden bei der Migration automatisch ersetzt. Anschließend müssen Sie noch manuelle Anpassungen vornehmen.

Die folgende Tabelle gibt einen Überblick über die Migration der Datenbausteinanweisungen:

Anweisung in S7-300/400	Anweisung in S7-1500	Migrationsklasse
CREA_DBL	CREATE_DB	3
CREAT_DB	CREATE_DB	3
CREA_DB	CREATE_DB	3
DEL_DB	DELETE_DB	3
TEST_DB	ATTR_DB	3

ACHTUNG
Asynchrone Bearbeitung Die neuen Anweisungen arbeiten in S7-1500 asynchron, d. h. ihre Ausführung erstreckt sich über mehrere Programmzyklen.

Manuelle Anpassungen nach der Migration

Die Anweisungen haben in S7-1500 zusätzliche Formalparameter, die z. B. für die asynchrone Bearbeitung erforderlich sind. Diese zusätzlichen Parameter müssen Sie nach der Migration mit passenden Aktualparametern versorgen.

Die Bausteinnummer des erzeugten DB muss in S7-1500 zwischen 60000 und 60999 liegen. Passen Sie deshalb die Parameter "LOW_LIMIT" und "UP_LIMIT" an den neuen Nummernbereich an.

Eine detaillierte Beschreibung der Parameter finden Sie in der Referenzhilfe zu S7-1500.

Siehe auch:

AUTOHOTSPOT

AUTOHOTSPOT

Beispiel

Das folgende Beispiel zeigt die Migration der Anweisung "CREAT_DB" in die Anweisung "CREATE_DB".

Im ersten Bild sehen Sie den Aufruf von "CREAT_DB" vor der Migration:

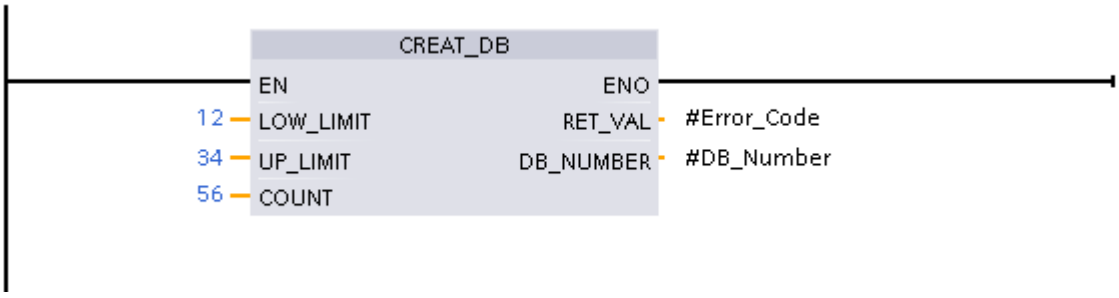

Im zweiten Bild sehen Sie den Aufruf von "CREATE_DB" nach der Migration:

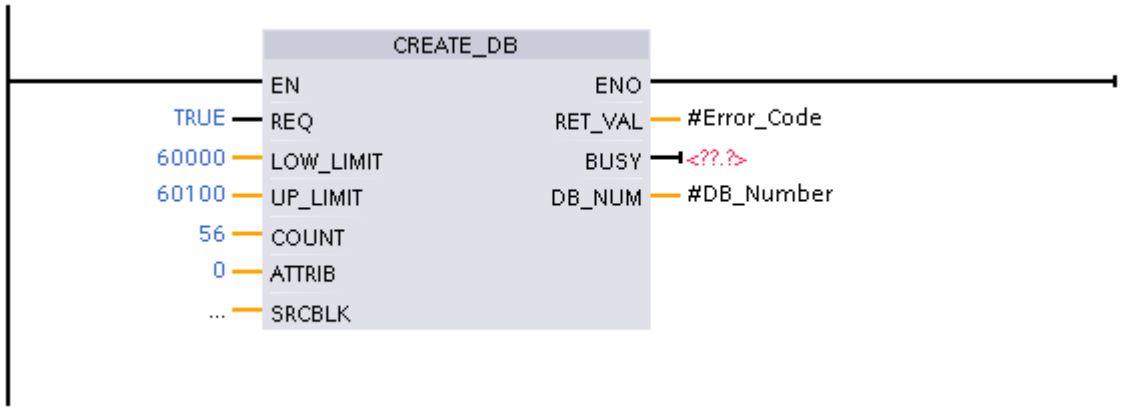

- Der Parameter "REQ" war bei "CREAT_DB" nicht vorhanden und wurde von der Migration ergänzt. Er erhält den voreingestellten Wert "TRUE".
- Die Parameter "LOW_LIMIT" und "UP_LIMIT" lag außerhalb des zulässigen Nummernbereichs von 60000 bis 60999. Während der Migration wurden die Werte "12" und "34" deshalb als ungültig gekennzeichnet. Anschließend wurden manuell die zulässigen Werte "60000" und "60100"eingegeben.
- Der Parameter "ATTRIB" war bei "CREAT_DB" nicht vorhanden und wurde von der Migration ergänzt. Er erhält den voreingestellten Wert "0". An diesem Parameter können Sie die Eigenschaften des zu erzeugenden DBs spezifizieren.
- Der Parameter "SRCBLK" war bei "CREAT_DB" nicht vorhanden und wurde von der Migration ergänzt. Hier können Sie den Datenbereich angeben, mit dessen Inhalt der zu erzeugende DB beschrieben werden soll.
- Der Parameter "BUSY" war bei "CREAT_DB" nicht vorhanden und wurde von der Migration ergänzt. Der Parameter führt so lange das Signal "1", bis der Vorgang der Datenbausteinerzeugung abgeschlossen ist. Um das Signal auszuwerten, müssen Sie den Ausgangsparameter mit einem Operanden verschalten.

3.4.2.4 Migration der Anweisung RDSYSST

Vorgehen bei der Migration von RDSYSST (Migrationsklasse 4)

Die Anweisung "RDSYSST: Systemzustandsliste auslesen" steht in S7-1500 nicht mehr zur Verfügung, da die CPUs dieser Projektfamilie keine Systemzustandslisten bereitstellen. Es besteht jedoch die Möglichkeit, einige der Informationen, die Sie bisher über die Systemzustandsliste erhalten haben, über eine der folgenden Anweisungen auszulesen:

- "RDREC" (Erweiterte Anweisungen > Dezentrale Peripherie)
- "GET_DIAG" (Erweiterte Anweisungen > Diagnose)
- "GEO2LOG" (Erweiterte Anweisungen > Diagnose)
- "DeviceStates" (Erweiterte Anweisungen > Diagnose)
- "ModuleStates" (Erweiterte Anweisungen > Diagnose)
- "RALRM" (Eweiterte Anweisungen > Dezentrale Peripherie)
- "LED" (Erweiterte Anweisungen > Diagnose)

Systemzustandsinformationen in S7-1500 auslesen

Die folgende Tabelle zeigt, welche Informationen über den Systemzustand Sie in S7-1500 auslesen können.

In der Spalte "SZL-ID" finden Sie die IDs, die Sie bisher mit der Anweisung RDSYSST gelesen haben. Relevant sind die beiden Endziffern. Der Platzhalter "xy" variiert je nach eingesetzter Baugruppe.

Die Spalte "Migration" gibt an, ob die Information in S7-1500 weiterhin zur Verfügung steht und wie Sie sie auslesen können.

Teilliste	SZL-ID	Migration
Baugruppen-Identifikation	W#16#xy11	Diese Information ist mit der Anweisung "RDREC" auslesbar. Lesen Sie am Parameter "INDEX" den Datensatz 16#AFF0.
CPU-Merkmale	W#16#xy12	In S7-1500 steht diese Information derzeit nicht zur Verfügung.
Anwenderspeicherbereiche	W#16#xy13	In S7-1500 steht diese Information derzeit nicht zur Verfügung.
Systembereiche	W#16#xy14	In S7-1500 steht diese Information derzeit nicht zur Verfügung.
Bausteintypen	W#16#xy15	In S7-1500 steht diese Information derzeit nicht zur Verfügung.
Identifikation einer Komponente	W#16#xy1C	Für zentrale Module und PROFINET IO ist diese Information mit der Anweisung "RDREC" auslesbar. Lesen Sie am Parameter "INDEX" den Datensatz 16#AFF0.

Teilliste	SZL-ID	Migration
Alarmstatus	W#16#xy22	In S7-1500 steht diese Information nicht zur Verfügung.
Zuordnung zwischen Teilprozessabbildern und OBs	W#16#xy25	In S7-1500 steht diese Information nicht zur Verfügung.
Kommunikation: Zustandsdaten	W#16#xy32	In S7-1500 steht diese Information nicht zur Verfügung.
H-CPU-Sammelinformation	W#16#xy71	Nicht relevant
Zustand der Baugruppen-LEDs	W#16#xy74	Diese Information ist mit der Anweisung "LED" auslesbar. Lesen Sie am Parameter "LED" den Datensatz W#16#xy74.
Geschaltete DP-Slaves im H-System	W#16#xy75	Nicht relevant
DP-Mastersystem-Information	W#16#xy90	In S7-1500 steht diese Information nicht zur Verfügung.
Baugruppenzustandsinfo	W#16#xy91	Diese Information ist mit den Anweisungen "GET_DIAG" oder "ModuleStates" auslesbar.
Baugruppenträger- /Stationszustandsinformation	W#16#xy92	Diese Information ist mit den Anweisungen "GET_DIAG" oder "DeviceStates" auslesbar.
Baugruppenträger- /Stationszustandsinformation	W#16#xy94	Diese Information ist mit den Anweisungen "GET_DIAG" oder "DeviceStates" auslesbar.
Erweiterte DP-Mastersystem-Information	W#16#xy95	In S7-1500 steht diese Information nicht zur Verfügung.
Baugruppenzustandsinformation PROFINET IO und PROFIBUS DP	W#16#xy96	Diese Information ist mit den Anweisungen "GET_DIAG", "ModuleStates" oder "DeviceStates" auslesbar.
Werkzeugwechslerinformationen (PROFINET IO)	W#16#xy9C	In S7-1500 steht diese Information nicht zur Verfügung.
Diagnosepuffer	W#16#xyA0	In S7-1500 steht diese Information nicht zur Verfügung.
Baugruppendiagnoseinfo (DS 0)	W#16#00B1	Diese Information ist mit den Anweisungen "GET_DIAG" oder "RDREC" auslesbar. Innerhalb des Alarm-OB erhalten Sie die Diagnoseinfo auch über "RALRM".
Baugruppendiagnoseinfo (DS 1), physikalische Adresse	W#16#00B2	Diese Information ist mit den Anweisungen "RDREC" oder "GEO2LOG" auslesbar. Innerhalb des Alarm-OB erhalten Sie die Diagnoseinfo auch über "RALRM".
Baugruppendiagnoseinfo (DS 1), logische Adresse	W#16#00B3	Diese Information ist mit den Anweisungen "GET_DIAG" oder "RDREC" auslesbar. Innerhalb des Alarm-OB erhalten Sie die Diagnoseinfo auch über "RALRM".
Diagnosedaten eines DP-Slaves	W#16#00B4	Diese Information ist teilweise durch die Anweisungen "ModuleStates" oder "GET_DIAG" auslesbar. Innerhalb des Alarm-OB erhalten Sie die Diagnoseinfo auch über "RALRM". Teilweise stehen die Informationen in S7-1500 nicht zur Verfügung.

Hinweis

Unterstützung bei nicht auslesbaren Informationen

Falls Sie weitere Unterstützung zu nicht auslesbaren Informationen benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur PLC-Migration:

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Siehe auch

FAQs zur Migration (http://support.automation.siemens.com/WW/view/de/67858106)

3.4.2.5 Migration der Anweisung T_CONV

Wochentage mit T_CONV aus DATE_AND_TIME extrahieren

Die Anweisung "T_CONV" steht sowohl in S7-300/400 als auch in S7-1500 zur Verfügung. Die Anweisung wird in den meisten Fällen automatisch migriert. Wenn Sie "T_CONV" verwendet haben, um den Wochentag aus dem Datentyp DATE_AND_TIME (DT) zu extrahieren, müssen Sie in S7-1500 jedoch folgendes beachten:

In S7-300/400 wurde zur Extraktion des Wochentags ein Ausgangsparameter "OUT" vom Datentyp INT erwartet. In S7-1500 muss der Ausgangsparameter stattdessen vom Datentyp DTL sein, wenn Sie einen Wochentag als Ergebnis erwarten.

Wenn der Ausgangsparameter "OUT" einen anderen Datentyp hat, wird bei der PLC-Migration ein Fehler im Migrationsprotokoll ausgegeben. Wandeln Sie in diesem Fall den Datentyp des Ausgangsparameters in DTL um und verwenden Sie die Komponente "Weekday".

Weitere Informationen finden Sie unter "Siehe auch".

3.4.2.6 Migration von Meldungen mit Begleitwerten

Meldungsprojektierung mit S7-1500

Alle Funktionen zur Meldungsprojektierung können Sie in S7-1500 zentral mit der Anweisung "Program_Alarm" erledigen.

"Program_Alarm" löst die folgenden Anweisungen der S7-300/400 ab:

Anweisung zur Meldungsprojektierung (S7-300/400)	Anweisung zur Meldungsprojektierung (S7-1500)
ALARM_SQ	Program_Alarm (Migrationsklasse 4)
ALARM_S	
ALARM_SC	
ALARM_DQ	
ALARM_D	
NOTIFY_8P	
ALARM	
ALARM_8	
ALARM_8P	
NOTIFY	
AR_SEND	

Vorgehen bei der Migration von Meldungen

Meldungen werden nicht automatisch migriert. Sie müssen die Meldungen nach der Migration manuell neu projektieren.

Im Folgenden erfahren Sie, wie Sie Ihr Programm so vorbereiten können, dass die Meldungstexte nach der Migration erhalten bleiben und die manuelle Nacharbeit minimal bleibt.

Die Vorgehensweise gilt nicht für Meldungsanweisungen, die die Datentypen "C_Alarm_8", "C_Alarm_8p", "C_Alarm_t"; "C_Notify_8p" oder "C_Ar_send" verwenden.

Um Meldungen zu migrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, der die Meldung enthält.
   Die Schnittstelle des Bausteins enthält einen Parameter, der einen Meldungsdatentypen hat (z. B. den Datentypen "C_Alarm_s", " C_Notify", etc.). Dieser Parameter definiert den Meldenummerneingang.
2. Verschieben Sie den Parameter für den Meldenummerneingang in den Abschnitt "Static".

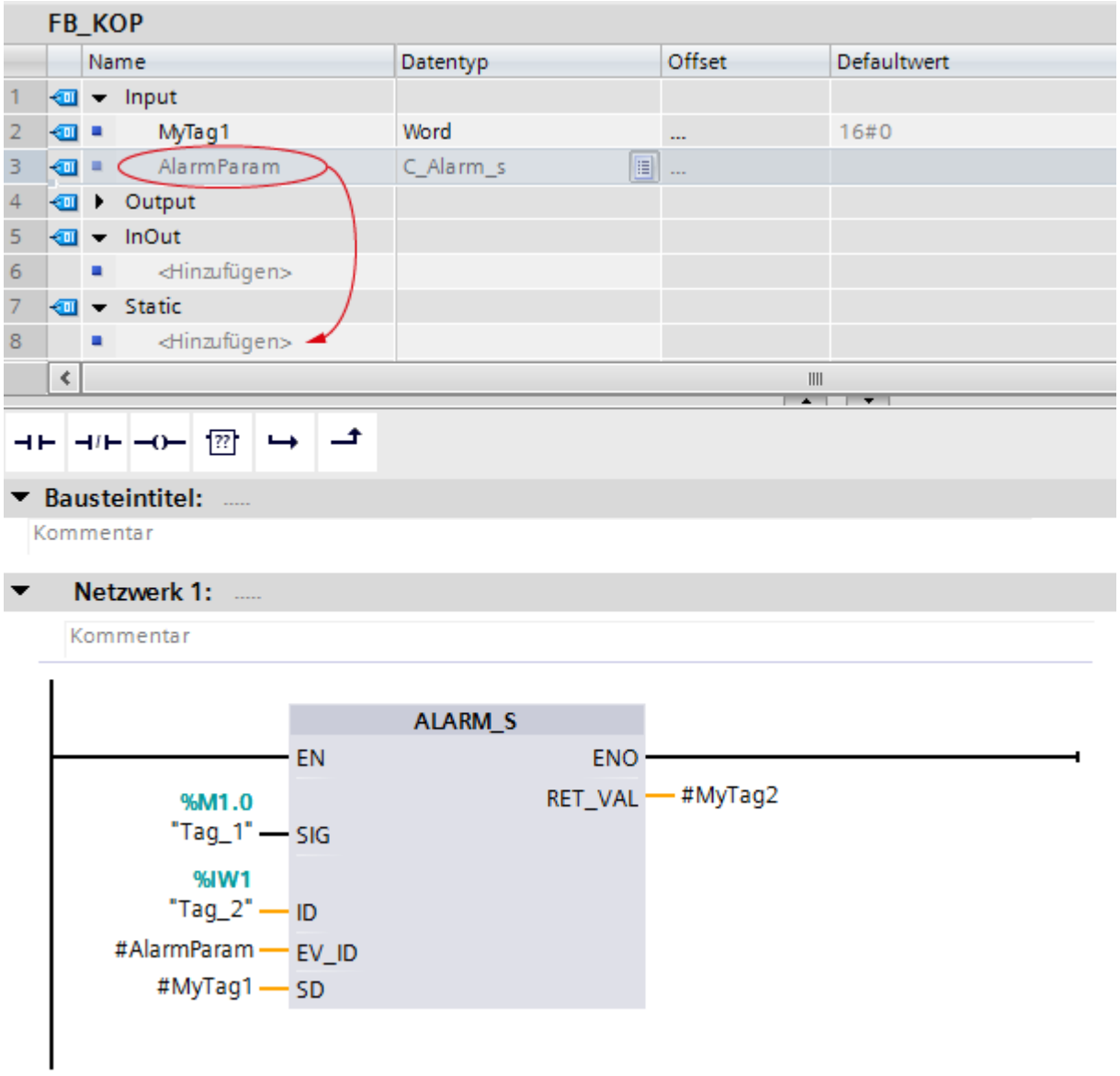

3. Speichern und übersetzen Sie den Baustein.
4. Führen Sie anschließend die PLC-Migration durch.
   Siehe auch: Migration durchführen (Seite 156)
5. Öffnen Sie in der neu angelegten CPU erneut den Baustein, der die Meldung enthielt.
   Die Anweisung zur Meldungsprojektierung konnte nicht migriert werden und wird rot dargestellt.

6. Ersetzen Sie in der Bausteinschnittstelle den Meldungsdatentypen (z.B. "C_Alarm_s") durch den neuen Datentypen "Program_Alarm". Wenn der Datentyp "Program_Alarm" nicht in der Auswahlliste angeboten wird, geben Sie den Namen über die Tastatur ein.
7. Öffnen Sie die Task Card "Anweisungen" und navigieren in der Palette "Erweiterte Anweisungen" zum Ordner "Meldungen".
8. Ziehen Sie die Anweisung "Program_Alarm" in Ihr Netzwerk.
9. Wählen Sie im Dialog "Aufrufoptionen" im Eingabefeld "Name in der Schnittstelle" die zuvor umprojektierte Meldungsvariable aus.
10. Öffnen Sie die Eigenschaften der Variable und wechseln Sie zum Register "Meldung". Der Meldetext aus dem Ausgangsprogramm ist eingetragen.
11. Löschen Sie das Zeichen für den Elementtyp aus den Begleitwerten (z. B. Y, W, X, I etc.). In S7-1500 enthalten Begleitwerte keine Angaben zum Elementtyp.
    Siehe auch: AUTOHOTSPOT
12. Löschen Sie den alten Meldebaustein aus dem Netzwerk.
    Die Meldungsprojektierung wurde migriert. Der Meldetext aus dem Ausgangsprogramm wurde übernommen.

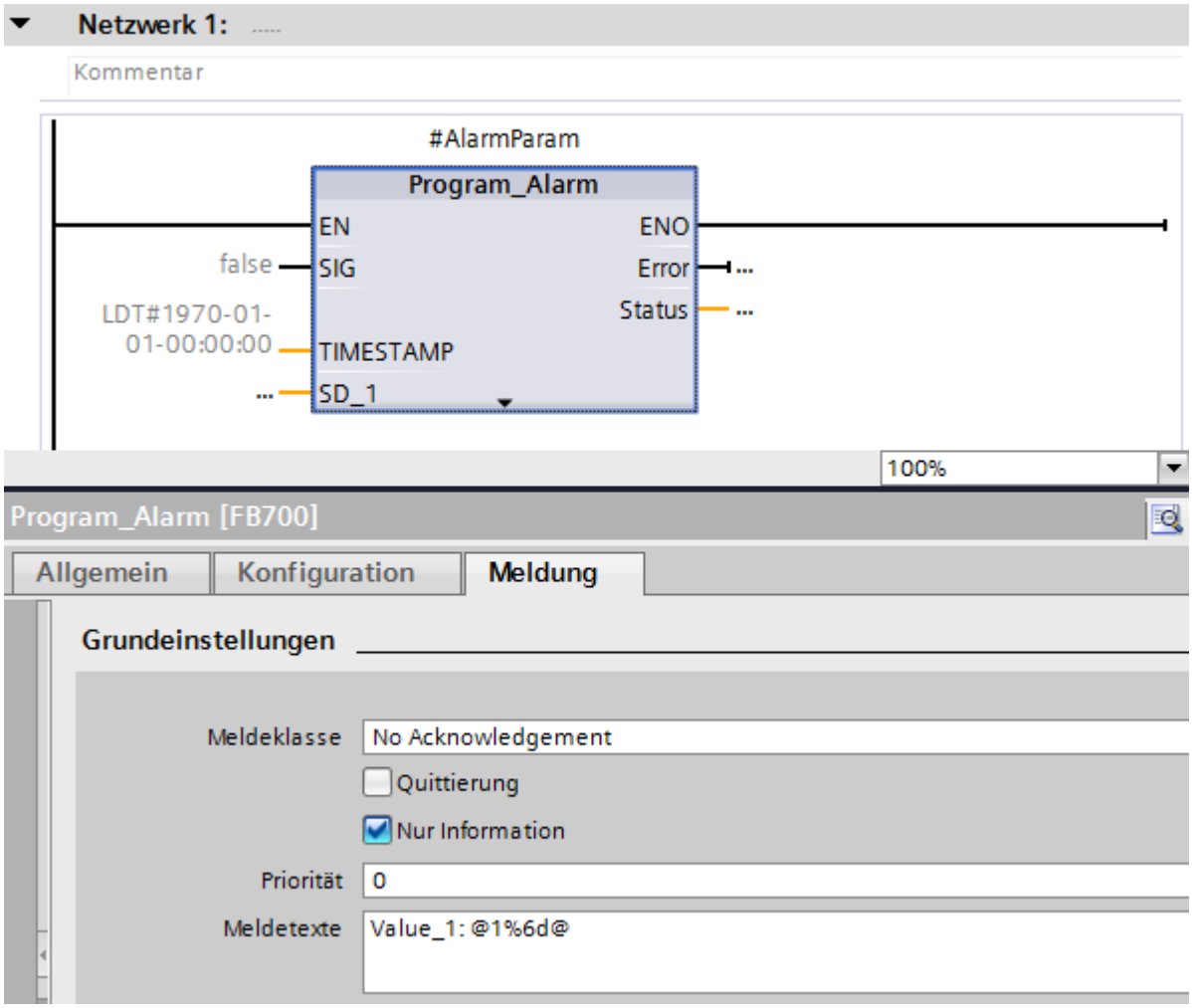

3.4.2.7 Migration von Tabellenanweisungen

Beispiel zur Migration von "ATT: Wert in Tabelle eintragen"

Vorgehen bei der Migration von Tabellenanweisungen (Migrationsklasse 4)

Die Anweisungen aus der Gruppe "Tabellenanweisungen" stehen in S7-1500 nicht mehr zur Verfügung und können durch die Migration nicht ersetzt werden, da sie auf einer rein absoluten Adressierung beruhen. In S7-1500 stehen Ihnen jedoch komfortable Funktionen zur Verfügung, mit denen Sie die Anweisungen selbst implementieren können. Im Folgenden sehen Sie ein Beispiel zur Umsetzung der Anweisung "ATT: Wert in Tabelle eintragen".

Umsetzung

Das folgende SCL-Programm zeigt die Umsetzung der Anweisung "ATT". Die Funktion im Beispiel wurde als FC in SCL implementiert.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Operand		Deklaration	Datentyp	Beschreibung
val		Input	INT	Eingegebener Wert
tab		InOut	PLC-Datentyp "Tabelle10"	Tabelle mit 10 Zeilen
	x	InOut	ARRAY [0..9] of INT	Jedes ARRAY-Element repräsentiert eine Zeile. Im Beispiel wird ein ARRAY of INT verwendet. Sie können jedoch ARRAYs mit jedem beliebigen Datentyp erstellen. Auch ARRAY of STRUCT oder ein ARRAY eines PLC-Datentyps ist möglich.
	lng	InOut	INT	Länge des Puffers. Der Wert richtet sich nach der Anzahl der ARRAY-Elemente von "tab.x".
	first	InOut	INT	Erster geschriebener Wert
	last	InOut	INT	Letzter geschriebener Wert
h		Temp	INT	Hilfsvariable
Ret_Val			BOOL	Funktionswert

Die folgende Tabelle zeigt den FC "Attend":

Die Funktion schreibt den Wert des Eingangsparameters "#val" in die letzte Tabellenzeile.

```
SCL
#h:=(#tab.last+1) MOD #tab.lng;    // Index hochzählen
IF #h <> #tab.first THEN           // Abfrage, ob sich die Indizes überholen
   #tab.x[#tab.last] := #val;      // Wert schreiben
   #tab.last := #h;               // Index des nächsten freien Elements speichern
   #Attend:= false;
ELSE
  #Attend:= true;  // Fehler >> Die Tabelle ist voll, der Wert kann nicht ein-
getragen werden
END_IF;
```

Beispiel zur Migration von "FIFO: Ersten Wert der Tabelle ausgeben"

Vorgehen bei der Migration von Tabellenanweisungen (Migrationsklasse 4)

Die Anweisungen aus der Gruppe "Tabellenanweisungen" stehen in S7-1500 nicht mehr zur Verfügung und können durch die Migration nicht ersetzt werden, da sie auf einer rein absoluten Adressierung beruhen. In S7-1500 stehen Ihnen jedoch komfortable Funktionen zur Verfügung, mit denen Sie die Anweisungen selbst implementieren können. Im Folgenden sehen Sie ein Beispiel zur Umsetzung der Anweisung "FIFO: Ersten Wert der Tabelle ausgeben".

Umsetzung

Das folgende SCL-Programm zeigt die Umsetzung der Tabellenfunktion "FIFO". Die Funktion im Beispiel wurde als FC in SCL implementiert.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Operand		Deklaration	Datentyp	Beschreibung
val		Output	INT	Ausgegebener Wert
tab		InOut	PLC-Datentyp "Tabelle10"	Tabelle mit 10 Zeilen
	x	InOut	ARRAY of INT [0..9]	Jedes ARRAY-Element repräsentiert eine Zeile. Im Beispiel wird ein ARRAY of INT verwendet. Sie können jedoch ARRAYs mit jedem beliebigen Datentyp erstellen. Auch ARRAY of STRUCT oder ein ARRAY eines PLC-Datentyps ist möglich.
	lng	InOut	INT	Länge des Puffers. Der Wert richtet sich nach der Anzahl der ARRAY-Elemente von "tab.x".

Operand		Deklaration	Datentyp	Beschreibung
	first	InOut	INT	Erster geschriebener Wert
	last	InOut	INT	Letzter geschriebener Wert
h		Temp	INT	Hilfsvariable
Ret_Val			BOOL	Funktionswert

Die folgende Tabelle zeigt den FC "Fifo":

Die Funktion liefert den ersten in die Tabelle geschriebenen Wert zurück (first in, first out). Der Funktionswert "true" zeigt an, dass die Tabelle leer ist.

SCL

```
#h:=(#tab.first+1) MOD #tab.lng; // Index hochzählen
  IF #h <> #tab.last THEN        // Abfrage, ob sich die Indizes
überholen
    #tab.first := #h;            // Index des nächsten Wertes
speichern
    #val := #tab.x[#h];      // gewünschten Wert ausgeben
    #Fifo:= false;
  ELSE
    #Fifo:= true; // Fehler >> Die Tabelle ist leer, es wird kein
Wert ausgegeben
    #val := #tab.x[#tab.first]; // letzten gültigen Wert erneut
ausgeben
  END_IF;
```

Beispiel zur Migration von "LIFO: Letzten Wert der Tabelle ausgeben"

Vorgehen bei der Migration von Tabellenanweisungen (Migrationsklasse 4)

Die Anweisungen aus der Gruppe "Tabellenanweisungen" stehen in S7-1500 nicht mehr zur Verfügung und können durch die Migration nicht ersetzt werden, da sie auf einer rein absoluten Adressierung beruhen. In S7-1500 stehen Ihnen jedoch komfortable Funktionen zur Verfügung, mit denen Sie die Anweisungen selbst implementieren können. Im Folgenden sehen Sie ein Beispiel zur Umsetzung der Anweisung "LIFO: Letzten Wert der Tabelle ausgeben".

Umsetzung

Das folgende SCL-Programm zeigt die Umsetzung der Tabellenfunktion "LIFO". Die Funktion im Beispiel wurde als FC in SCL implementiert.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Operand		Deklaration	Datentyp	Beschreibung
val		Output	INT	Ausgegebener Wert
tab		InOut	PLC-Datentyp "Tabelle10"	Tabelle mit 10 Zeilen
	x	InOut	ARRAY of INT [0..9]	Jedes ARRAY-Element repräsentiert eine Zeile. Im Beispiel wird ein ARRAY of INT verwendet. Sie können jedoch ARRAYs mit jedem beliebigen Datentyp erstellen. Auch ARRAY of STRUCT oder ein ARRAY eines PLC-Datentyps ist möglich.
	lng	InOut	INT	Länge des Puffers. Der Wert richtet sich nach der Anzahl der ARRAY-Elemente von "tab.x".
	first	InOut	INT	Erster geschriebener Wert
	last	InOut	INT	Letzter geschriebener Wert
h		Temp	INT	Hilfsvariable
Ret_Val			BOOL	Funktionswert

Die folgende Tabelle zeigt den FC "Lifo":

Die Funktion liefert den letzten in die Tabelle geschriebenen Wert zurück (last in, first out). Der Funktionswert "true" zeigt an, dass die Tabelle leer ist.

SCL

```
#h:=(#tab.last-1) MOD #tab.lng;           // Index herunter-
zählen
  IF #h<0 THEN #h:=#h + #tab.lng; END_IF; // nur positive In-
dizes zulässig
   IF #h <> #tab.first THEN              // Abfrage, ob sich
die Indizes überholen
    #tab.last := #h;                    // Index des nächsten
Wertes speichern
    #val := #tab.x[#h];               // gewünschten Wert
ausgeben
    #Lifo:= false;
  ELSE
    #Lifo:= true;  // Fehler >> Die Tabelle ist leer, es wird
kein Wert ausgegeben
    #val := #tab.x[#tab.last];           // letzten gültigen
Wert erneut ausgeben
  END_IF;
```

3.4.2.8 Migration von Kommunikationsanweisungen

SIMATIC NET CP/CM

Migration von Programmbausteinen

Einführung

S7-300/400 verwendet spezifische Programmbausteine für Kommunikationsfunktionen über SIMATIC NET CPs. S7-1500 bietet einen geänderten Satz von Anweisungen für diese Kommunikationsfunktionen.

Die nachfolgenden Tabellen geben eine Übersicht über die Migration der bei S7-300/400 verwendeten Programmbausteine.

Nähere Informationen zu der Angabe in der Spalte "Klasse" finden Sie im Kapitel Hinweise zur Migration von Anweisungen (Seite 167).

Programmbausteine für Industrial Ethernet / PROFINET

Kommunikationsdienst / Funktionsbereich	Typ der Anweisung	Bibliothek		Anweisung in S7-1500	Klasse
		CP 300	CP 400		
SEND / RECEIVE (Offene Kommunikationsdienste)	AG_SEND Migration AG_SEND/ AG_LSEND (Seite 193)	x	x	T_SEND	3
	AG_LSEND Migration AG_SEND/ AG_LSEND (Seite 193)	x	x	T_SEND	3
	AG_SEND (über UDP) Migration AG_SEND/ AG_LSEND (Seite 193)	x	x	TUSEND	3
	AG_SEND (E-Mail) Migration AG_SEND/ AG_LSEND (Seite 193)	x	x	TMAIL_C	3
	AG_RECV Migration AG_RECV/ AG_LRECV (Seite 195)	x	x	T_RCV	3
	AG_LRECV Migration AG_RECV/ AG_LRECV (Seite 195)	x	x	T_RCV	3
	AG_LOCK	x	x	-	4
	AG_UNLOCK	x	x	-	4
	AG_CNTRL	x	x	T_DIAG, T_RESET, TCON, TDISCON	3
Programmierte Kommunikationsverbindungen	IP_CONFIG	x	x	T_CONFIG	3
FTP	FTP_CMD	x	x	FTP_CMD	3

Kommunikationsdienst / Funktionsbereich	Typ der Anweisung	Bibliothek		Anweisung in S7-1500	Klasse
		CP 300	CP 400		
PROFINET IO	PNIO_SEND	x		-	4 *)
	PNIO_RECV	x		-	4 *)
	PNIO_RW_REC	x		-	4 *)
	PNIO_ALARM	x		-	4 *)

*) Es sind andere Zugriffsmechanismen anzuwenden.

Programmbausteine für PROFIBUS

Kommunikationsdienst / Funktionsbereich	Typ der Anweisung	Bibliothek		Anweisung in S7-1500	Fall
		CP 300	CP 400		
SEND / RECEIVE (Offene Kommunikationsdienste)	AG_SEND Migration AG_SEND/ AG_LSEND (Seite 193)	x	x	T_SEND	3
	AG_LSEND Migration AG_SEND/ AG_LSEND (Seite 193)	-	x	T_SEND	3
	AG_RECV Migration AG_RECV/ AG_LRECV (Seite 195)	x	x	T_RCV	3
	AG_LRECV Migration AG_RECV/ AG_LRECV (Seite 195)	-	x	T_RCV	3
PROFIBUS DP	DP_SEND	x	-	-	4 *)
	DP_RECV	x	-	-	4 *)
	DP_DIAG	x	-	-	4 *)
	DP_CTRL	x	-	-	4 *)

*) Es sind andere Zugriffsmechanismen anzuwenden.

Zusätzliche Formalparameter

Die Anweisungen haben in S7-1500 zusätzliche Formalparameter, die z. B. die für die asynchrone Bearbeitung erforderlich sind. Diese zusätzlichen Parameter müssen Sie nach der Migration mit passenden Aktualparametern versorgen.

Eine detaillierte Beschreibung der Parameter finden Sie in der Referenzhilfe zu S7-1500.

Siehe auch:

AUTOHOTSPOT

Beispiel

Ein prinzipielles Beispiel für die Darstellung einer migrierten Anweisung sehen Sie in Kapitel Migration von Datenbausteinanweisungen (Seite 179)

Migration AG_SEND/AG_LSEND

Migration der Aufrufparameter - Kommunikation über ISO-Transprt, ISO-on-TCP, TCP

Tabelle 3-1 Migration AG_SEND / AG_LSEND

AG_SEND / AG_LSEND		T_SEND		Beschreibung / Hinweise
Parameter	Typ	Parameter	Typ	
ACT	BOOL	REQ	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	
SEND	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
DONE	BOOL	DONE	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Zusätzliche Schritte

- Konfigurieren Sie den entsprechenden Verbindungstyp mit den relevanten Adressparametern.

Migration der Aufrufparameter - Kommunikation über UDP

Tabelle 3-2 Migration AG_SEND / AG_LSEND

AG_SEND / AG_LSEND		TUSEND		Beschreibung / Hinweise
Parameter	Typ	Parameter	Typ	
ACT	BOOL	REQ	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	
SEND	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
DONE	BOOL	DONE	BOOL	

AG_SEND / AG_LSEND		TUSEND		Beschreibung / Hinweise
Parameter	Typ	Parameter	Typ	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Zusätzliche Schritte

- Konfigurieren Sie den entsprechenden Verbindungstyp mit den relevanten Adressparametern.

Migration der Aufrufparameter - Kommunikation über E-MAIL

Tabelle 3-3 Migration AG_SEND / AG_LSEND

AG_SEND / AG_LSEND		TMAIL_C		Beschreibung / Hinweise
Parameter	Typ	Parameter	Typ	
ACT	BOOL	REQ	BOOL	
ID	INT	--	--	
LADDR	WORD	--	--	
SEND	ANY	TO_S	STRING	
SEND	ANY	CC	STRING	
SEND	ANY	SUBJECT	STRING	
SEND	ANY	TEXT	STRING	
SEND	ANY	ATTACHMENT	VARIANT	
--	--	ATTACHMENT _NAME	STRING	
--	--	MAIL_ADDR_PARAM	VARIANT	
LEN	INT	--	--	
DONE	BOOL	DONE	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Zusätzliche Schritte

- Konfigurieren Sie den entsprechenden Verbindungstyp mit den relevanten Adressparametern.

Migration AG_RECV/AG_LRECV

Migration der Aufrufparameter

Tabelle 3-4 Migration AG_RECV / AG_LRECV

AG_RECV / AG_LRECV		TRCV / TURCV		Beschreibung / Hinweise
Parameter	Typ	Parameter	Typ	
		EN_R	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	
--	--	LEN	UDINT	
RECV	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
NDR	BOOL	NDR	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Zusätzliche Schritte

- Konfigurieren Sie den entsprechenden Verbindungstyp mit den relevanten Adressparametern.

Punkt-zu-Punkt CP/CM

Migration von Programmbausteinen PtP

Einführung

S7-300/400 verwendet spezifische Programmbausteine für Kommunikationsfunktionen über Punkt-zu-Punkt-CPs.

S7-1500 bietet einen geänderten Satz von Anweisungen für diese Kommunikationsfunktionen.

Bei der Migration werden Sie gefragt, ob Sie den geänderten Satz von Anweisungen für die Kommunikationsmodule der S7-1500 oder die Programmbausteine für die Kommunikationsprozessoren der S7-300/400 verwenden wollen.

Die nachfolgende Tabelle gibt Ihnen eine Übersicht über die Migration der bei S7-300/400 verwendeten Programmbausteine auf Anweisungen der S7-1500.

Nähere Informationen zu der Angabe in der Spalte "Klasse" finden Sie im Kapitel Hinweise zur Migration von Anweisungen (Seite 167).

Programmbausteine für Punkt-zu-Punkt-Kommunikation

Kommunikationsdienst / Funktionsbereich	Baustein	Bibliothek			Anweisung in S7-1500	Klasse
		CP 300	CP 400	ET 200S		
Bausteine für CP 340	P_RCV	x	-	-	Receive_P2P	3 [1]
	P_SEND	x	-	-	Send_P2P	3 [1]
	P_RESET	x	-	-	Receive_Reset	3 [1]
	V24_STAT_340	x	-	-	Signal_Get	3 [1]
	V24_SET_340	x	-	-	Signal_Set	3 [1]
	P_PRINT	x	-	-	-	4
Bausteine für CP 341	P_RCV_RK	x	-	-	Receive_P2P	3 [1]
	P_SND_RK	x	-	-	Send_P2P	3 [1]
	V24_STAT	x	-	-	Signal_Get	3 [1]
	V24_SET	x	-	-	Signal_Set	3 [1]
	P-PRT341	x	-	-	-	4
Bausteine für CP 440	RECV_440	-	x	-	Receive_P2P	3 [1]
	SEND_440	-	x	-	Send_P2P	3 [1]
	RES_RECV	-	x	-	Receive_Reset	3 [1]
Bausteine für CP 441	V24_STAT_441	-	x	-	Signal_Get	3 [1]
	V24_SET_441	-	x	-	Signal_Set	3 [1]
Bausteine für Modbus	MODB_341	x	-	-	-	4 [2]
	MODB_441	-	x	-	-	4 [2]
Bausteine für ET 200S 1SI	S_RCV	-	-	x	Receive_P2P	3 [1]
	S_SEND	-	-	x	Send_P2P	3 [1]
	S_VSTAT	-	-	x	Signal_Get	3 [1]
	S_VSET	-	-	x	Signal_Set	3 [1]
	S_XON	-	-	x	Port_Config	3 [1]
	S_RTS	-	-	x	Port_Config	3 [1]
	S_V24	-	-	x	Send_Config	3 [1]
	S_MODB	-	-	x	-	4 [2]
	S_USST	-	-	x	-	4 [2]
	S_USSR	-	-	x	-	4 [2]
	S_USSI	-	-	x	-	4 [2]
Bausteine für Kompakt-CPUs S7-300	SEND_PTP	x	-	-	Send_P2P	3 [1]
	RCV_PTP	x	-	-	Receive_P2P	3 [1]
	RES_RCVB	x	-	-	Receive_Reset	3 [1]

[1] Informationen zur Migration dieser Anweisung finden Sie unter diesem Link (Seite 167).

[2] Diese Anweisungen für ET 200S 1SI werden von S7-1500 nicht unterstützt. Verwenden Sie stattdessen die Anweisungen für Modbus (RTU) bzw. für USS in Verbindung mit dem Punkt-zu-Punkt-Modul "ET 200SP CM PtP".

Zusätzliche Formalparameter

Die Anweisungen haben in S7-1500 zusätzliche Formalparameter. Diese zusätzlichen Parameter müssen Sie nach der Migration mit passenden Aktualparametern versorgen.

Hinweis

Weitere Unterstützung zur Migration von Kommunikationsanweisungen

Im Siemens Industry Online Support finden Sie aktuelle Informationen zur Migration von Kommunikationsanweisungen.

Migration Kommunikation:

http://support.automation.siemens.com/WW/view/de/83558087 (http://support.automation.siemens.com/WW/view/de/83558087)

Migration Steuerungen:

http://support.automation.siemens.com/WW/view/de/83557459 (http://support.automation.siemens.com/WW/view/de/83557459)

Falls Sie weitere Unterstützung benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Siehe auch

FAQs zur Migration (http://support.automation.siemens.com/WW/view/de/67858106)

3.4.3 Migration von KOP/FUP-Programmen nach S7-1500

3.4.3.1 Hinweise zur Migration von KOP/FUP-Programmen

---(SAVE): Verknüpfungsergebnis in BIE-Register laden

Die Anweisung ist auf S7-1500 nicht verfügbar. Die Migration behandelt die Anweisung wie folgt:

- Wenn die Anweisung "---(SAVE)" im letzten Netzwerk eines Bausteins verwendet wurde, wird sie durch die Anweisung "---(RET)" mit dem Parameter "RLO" ersetzt.
- Wenn die Anweisung "---(SAVE)" in einem Netzwerk verwendet wurde, das nicht das letzte Netzwerk des Bausteins ist, kann sie nicht ersetzt werden. Ändern Sie das Programm manuell ab. Speichern Sie z. B. das VKE in einer PLC-Variable, einer DB-Variable oder einer lokalen Variable und geben Sie diese am Ende des Bausteins mit der Anweisung --(RET) an den aufrufenden Baustein zurück.

---(CALL) Baustein aufrufen ohne Parameter

Die Anweisung ist auf S7-1500 nicht verfügbar. Die Migration passt das Programm an, so dass der Aufruf über eine Box realisiert wird.

Master Control Relais

Das Master Control Relais ist auf S7-1500 nicht verfügbar. Die Migration meldet einen Fehler. Ändern Sie das Programm manuell ab. Nutzen Sie z. B. temporäre Variablen, um Anweisungen oder Netzwerke bedingt auszuführen.

Sprachwechsel innerhalb eines Bausteins

Die Inhalte der Register, der Akkumulatoren und des Statusworts stehen nur innerhalb von AWL-Netzwerken zur Verfügung. Wenn ein KOP- oder FUP-Netzwerk auf ein AWL-Netzwerk folgt, können Sie aus dem KOP- oder FUP-Netzwerk heraus nicht auf die Registerinhalte zugreifen, die zuvor in AWL gesetzt wurden. In einem nachfolgenden AWL-Netzwerk stehen die Registerinhalte jedoch wieder zur Verfügung.Eine Ausnahme bildet das VKE-Bit: Es wird bei einem Sprachwechsel auf "undefiniert" gesetzt und steht in nachfolgenden Netzwerken nicht mehr zur Verfügung.

Siehe auch: Werteübergabe über Register bei Sprachwechsel (Seite 204)

3.4.3.2 Teilqualifizierte Adressen in KOP/FUP

Einführung

Die Adressierung von DB-Variablen ohne Angabe des DB-Namens wird als teilqualifizierte Adressierung bezeichnet. Teilqualifizierte Adressierung wird in S7-1500 für KOP/FUP nicht unterstützt. Die Anweisungen "---(OPN)" und "---(OPNI)", die häufig in Zusammenhang mit teilqualifizierter Adressierung verwendet wurden, werden ebenfalls in S7-1500 nicht unterstützt.

Migration von teilqualifizierten Adressen

Die Migration behandelt teilqualifizierte Adressen, wenn möglich, wie folgt:

- Wenn der Datenbaustein eindeutig bestimmt werden kann, ersetzt die Migration teilqualifizierte Adressen durch vollständige Adressen, mit Angabe des DBs.
- Wenn der DB indirekt über das Datenbausteinregister übergeben wurde, kann der Datenbaustein nicht eindeutig bestimmt werden. In diesem Fall fügt die Migration in der Bausteinschnittstelle einen Parameter vom Datentyp "DB_Any" ein. An diesen Parameter wird der Datenbausteinname übergeben. Die Migration ersetzt anschließend alle teilqualifizierten Adressen durch Adressen mit folgender Syntax: `#<Parametername>.%<absolute Adresse>.`

Beispiel

Das folgende Beispiel zeigt die Migration einer teilqualifizierten Adressierung. Im ersten Bild sehen Sie das Programm vor der Migration:

Netzwerk 1: Variablen werden teilqualifiziert adressiert.

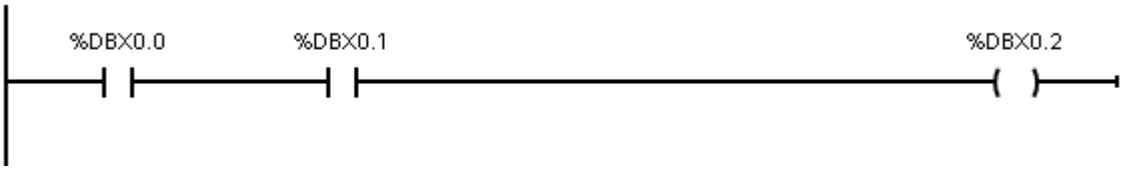

Das folgende Bild zeigt das Programm nach der Migration:

Netzwerk 1: Der DB, in dem die Variablen liegen, konnte eindeutig bestimmt werden. Das Programm wurde automatisch migriert. Die Variablen in "DB1" werden vollqualifiziert adressiert.

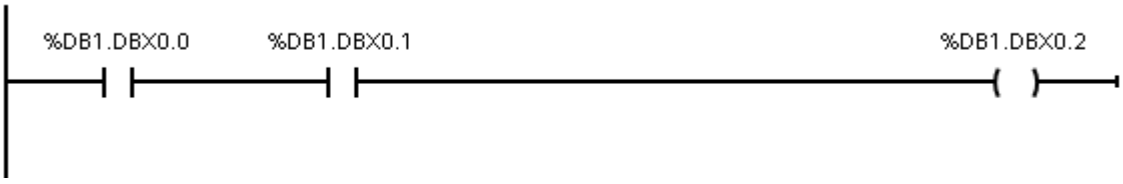

Hinweis

Migration von teilqualifizierten Bausteinparametern

Die Migration wandelt auch teilqualifizierte DB-Parameter in vollqualifizierte Parameter um. Beachten Sie, dass sich die Art der Parameterübergabe an den aufgerufenen Baustein dadurch ändern kann: Der aufgerufene Baustein greift über die vollqualifizierte Adresse möglicherweise nicht mehr direkt auf den Aktualparameter zu sondern arbeitet mit einer Kopie, die beim Bausteinaufruf übergeben wird.
Prüfen Sie deshalb, ob die Semantik des migrierten Programms noch mit der des Ausgangsprogramms übereinstimmt.
Weitere Informationen finden Sie unter: Zugriff auf Bausteinparameter während der Programmbearbeitung

Sollte diese Anpassung nicht gewünscht sein, können Sie zum Beispiel statt eines elementaren Bausteinparameters einen Parameter mit strukturiertem Datentyp verwenden. Definieren sie als Formalparameter PLC-Datentyp (UDT), und übergeben sie eine Variable dieses Typs bzw. einen DB, der von dem PLC-Datentypen (UDT) abgeleitet wurde.

3.4.3.3 Zugriffe auf Statusbits in KOP/FUP

Einführung

Um eine möglichst performante Programmbearbeitung auf CPUs der Baureihe S7-1500 zu erreichen, sind Wertübergaben zwischen Bausteinen prinzipiell nur über die Bausteinschnittstelle, über globale Datenbausteine oder über PLC-Variablen zulässig. Die Möglichkeit der Wertübergabe über das Statuswort besteht in KOP und FUP nicht.

Migration der Anweisungen "--| |--: Statusbit abfragen" / "--|/|--: Statusbit negiert abfragen"

Die Anweisungen sind auf S7-1500 nicht verfügbar. Die Migration unterscheidet folgende Fälle:

- Wenn eine Statusbitabfrage direkt nach einer mathematischen Anweisung im selben Netzwerk platziert ist, wird stattdessen ein Vergleicher eingesetzt.
- In allen anderen Fällen wird das Programm nach der Migration fehlerhaft. Sie müssen dann Ihr Programm ändern. Verwenden Sie beispielsweise anstelle der Abfrage des Statusbits OV eine Abfrage des Ausgangs ENO. Wenn Sie die Negation mehrerer ENO-Abfragen durch ODER miteinander verknüpfen, können Sie die Abfrage des Statusbits OS ersetzen.

Beispiel

Das folgende Beispiel zeigt die Migration einer Statusbitabfrage. Im ersten Bild sehen Sie das Programm vor der Migration.

Netzwerk 1: Die Anweisung "<=0" fragt das Statusbit "A1" ab. Es zeigt an, ob das Ergebnis der Multiplikation kleiner oder gleich Null ist.

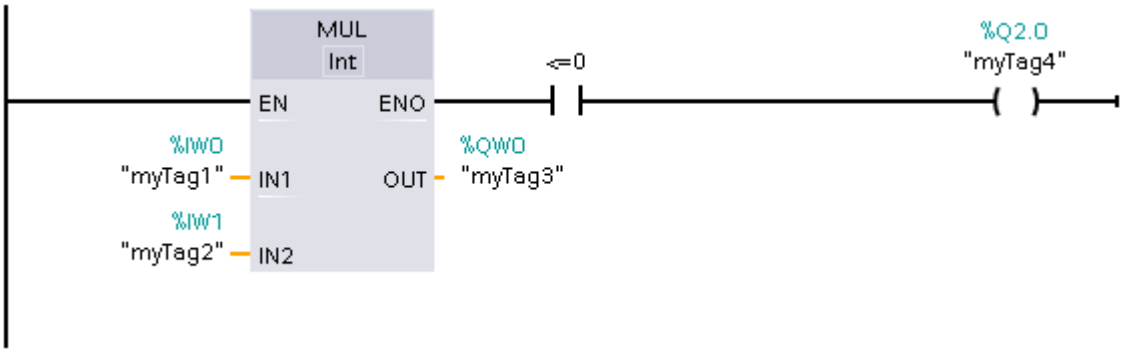

Das folgende Bild zeigt das Programm nach der Migration.

Netzwerk 1: Das Programm konnte automatisch migriert werden. Die Anweisung "<=0" wurde entfernt. Stattdessen wird das Ergebnis der Multiplikation mithilfe eines Vergleichers über die Variable "myTag3" abgefragt.

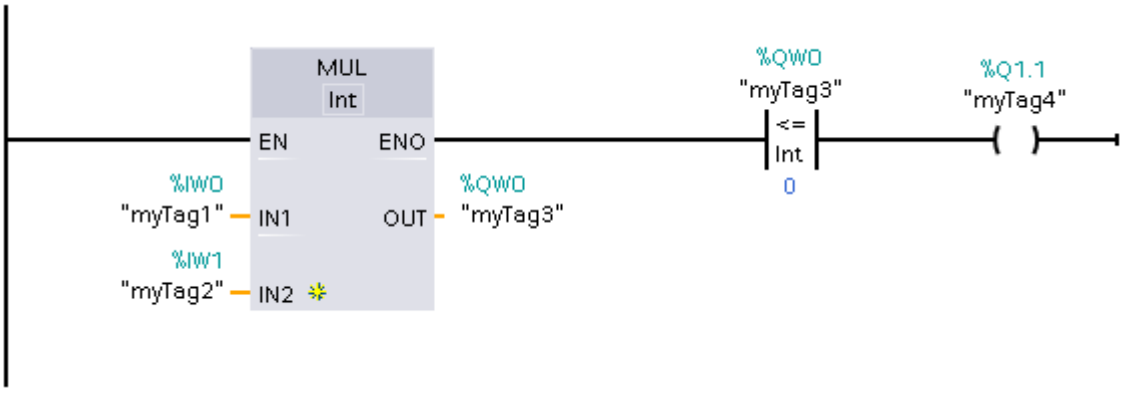

3.4.4 Migration von AWL-Programmen nach S7-1500

3.4.4.1 Hinweise zur Migration von AWL-Programmen

Übergabe von Werten über Register und Statuswort

Um eine möglichst performante Programmbearbeitung auf CPUs der Baureihe S7-1500 zu erreichen, sind Wertübergaben zwischen Bausteinen prinzipiell nur über die Bausteinschnittstelle, über globale Datenbausteine oder über PLC-Variablen zulässig.

Die Möglichkeit der Wertübergabe über Register (z. B. Akkumulatoren, AR1, AR2, DB, DI) oder das Statuswort besteht in KOP und FUP nicht. In AWL ist die Wertübergabe mit einigen Einschränkungen möglich. Beachten Sie jedoch, dass sich die Programmbearbeitung verlangsamt, wenn Sie diese Bereiche nutzen, um Werte zwischen verschiedenen Bausteinen zu übergeben.

Folgende Regeln gelten für AWL:

- Die Inhalte der Register, der Akkumulatoren und des Statusworts stehen nur innerhalb von AWL-Netzwerken zur Verfügung. Wenn ein KOP- oder FUP-Netzwerk auf ein AWL-Netzwerk folgt, können Sie aus dem KOP- oder FUP-Netzwerk heraus nicht auf die Registerinhalte zugreifen, die zuvor in AWL gesetzt wurden. In einem nachfolgenden AWL-Netzwerk stehen die Registerinhalte jedoch wieder zur Verfügung.
  Eine Ausnahme bildet das VKE-Bit: Es wird bei einem Sprachwechsel auf "undefiniert" gesetzt und steht in nachfolgenden Netzwerken nicht mehr zur Verfügung.
- Die Werte aus Registern, Akkumulatoren und dem Statuswort werden grundsätzlich nicht an aufgerufene Bausteine übergeben. Eine Ausnahme besteht nur bei den Anweisungen "CC" und "UC". Wenn Sie "UC" oder "CC" verwenden und dem aufgerufenen Baustein Parameter über Register, Statuswort oder Akkumulatoren übergeben möchten, müssen Sie in den Eigenschaften des aufgerufenen Bausteins die Option "Parameterversorgung über Register" aktivieren. Beachten Sie, dass diese Option nur für AWL-Bausteine mit Standardzugriff zur Verfügung steht und der Baustein keine Formalparameter haben darf. Wenn die Option aktiviert ist, können Sie Registerinhalte zwischen Bausteinen übergeben. Eine Ausnahme bildet auch hier das VKE-Bit: Es wird beim Verlassen des Bausteins auf "undefiniert" gesetzt und steht nach einem Bausteinaufruf nicht mehr zur Verfügung.
- Um eine Fehleraussage an den aufrufenden Baustein zu übergeben, können Sie das BIE-Bit nutzen. Die Fehleraussage müssen Sie zunächst im aufgerufenen Baustein mit der Anweisung "SAVE" im BIE-Bit ablegen. Anschließend können Sie das BIE-Bit im aufrufenden Baustein lesen.
- Nach jedem Zugriff auf einen Datenbaustein mit Angabe einer vollqualifizierten Adresse (z. B. `%DB10.DBW10`) wird das Datenbausteinregister DB auf "0" gesetzt. Ein nachfolgender teilqualifizierter Zugriff führt beim Übersetzen zu einem Fehler.
- Wenn Sie in S7-1500 einen lokalen Formalparameter aus der Bausteinschnittstelle eines FB symbolisch adressieren (z .B. mit der Anweisung L #myIn), greifen Sie immer auf den Datenbaustein zu, den Sie beim Bausteinaufruf als Instanz angegeben haben. Die Anweisungen AUF DI, L AR2, +AR2, TDB, TAR ändern zwar den Inhalt des DI- bzw. Adressregisters, die Register werden jedoch bei der Adressierung von lokalen Formalparametern nicht mehr ausgewertet.

Beispiele zur Programmierung finden Sie in den nachfolgenden Kapiteln.

Master Control Relais

Das Master Control Relais ist auf S7-1500 nicht verfügbar. Die Migration meldet einen Fehler. Ändern Sie das Programm manuell ab. Formulieren Sie Bedingungen, z. B. in Form von Bausteinparametern, um Anweisungen oder Netzwerke bedingt auszuführen.

LEAVE und ENT

Die Anweisungen "LEAVE" und "ENT" werden in S7-1500 nicht angeboten, da nur zwei Akkus zur Verfügung stehen. Die Migration meldet einen Fehler. Ändern Sie das Programm manuell ab. Nutzen Sie z. B. temporäre Variablen, um Zwischenergebnisse abzulegen.

Bausteinparameter vom Parametertyp "Block_DB"

Der Parametertyp "Block_DB" ist auf S7-1500 nicht verfügbar. Die Migration ändert diese Parameter ab und weist ihnen stattdessen den Datentyp "DB_Any" zu.

In S7-1500 ist es jedoch nicht möglich, Bibliotheksanweisungen beim Aufruf eine Instanz in Form eines Parameters vom Typ "DB_Any" zuzuordnen.

Das folgende Beispiel zeigt den Aufruf eines Bibliotheksbausteins mit einer variablen Instanz auf einer CPU der Familie S7-300/400. Diese Sequenz kann nicht nach S7-1500 migriert werden.

AWL	Erläuterung
CALL GET, #myBlock_DB	// Der Bibliotheksbaustein "GET" wird mit dem Instanzdatenbaustein aufgerufen, der aktuell am Bausteinparameter "myBlock_DB" anliegt.
REQ := #Start	
ID := W#16#100	
NDR := #Done	
ERROR := #Error	
STATUS := #Status	
ADDR_1 := P#DB10.DBX5.0 BYTE 10	
ADDR_2 :=	
ADDR_3 :=	
ADDR_4 :=	
RD_1 := P#DB10.DBX6.0 BYTE 10	
RD_2 :=	
RD_3 :=	
RD_4 :=	

3.4.4.2 Verknüpfungsketten und Sprünge

Einführung

In S7-300/400 ist es möglich, Sprungmarken in AWL an jede beliebige Anweisung zu setzen, um programmgesteuert an die markierte Stelle zu springen. Definieren Sie das Sprungziel jedoch innerhalb einer logischen Programmeinheit, an der das VKE keinen definierten Zustand hat, können während der Programmbearbeitung unerwartete Ergebnisse auftreten. Unter Umständen geht die CPU in den Zustand STOP.

Sie müssen also bei der Programmerstellung selbst darauf achten, Sprungmarken nur am Beginn einer Verknüpfungskette zu setzen. Vermeiden Sie auch Sprünge aus der Mitte einer Verknüpfungskette heraus.

Die CPU-Familien verhalten sich unterschiedlich, wenn ein Sprung in einer Verknüpfungskette auftritt:

- Bei der S7-300 wird das Erstabfrageverhalten einer Anweisung hinter einer Sprungmarke an Hand der linear davor ausgeführten Anweisungen bestimmt. Ob die Anweisung hinter der Sprungmarke linear oder mittels eines ausgeführten Sprungs erreicht wird, ist hierbei unerheblich.
- Bei der S7-400 wird das Erstabfrageverhalten einer Anweisung hinter einer Sprungmarke an Hand des tatsächlich ausgeführten Programmflusses bestimmt. Das Erstabfrageverhalten kann also davon abhängig sein, ob die Sprungmarke linear oder mittels eines ausgeführten Sprungs erreicht wird.
- In S7-1500 wird ein Übersetzungsfehler ausgegeben, wenn Programme Sprünge aus Verknüpfungsketten heraus oder in Verknüpfungsketten hinein enthalten, die bei Ausführung auf einer S7-300 bzw. S7-400 zu einem unterschiedlichen Laufzeitverhalten führen könnten. Das Programm muss an dieser Stelle angepasst werden, um das beabsichtigte Verhalten des Programms festzulegen.

Beispiel

Das folgende Beispiel zeigt einen Sprung in eine Verknüpfungskette hinein. Die Anweisung in Zeile 6 an der Sprungmarke n_OK hat kein eindeutig bestimmtes Erstabfrageverhalten. Somit ist die Bildung des VKE nicht eindeutig möglich. Diese Programmsequenz würde auf einer S7-1500 zu einem Übersetzungsfehler führen.

```
1          A     #Motor_Enabled
2          AN    #Motor_EmergencyStop
3          JU    n_OK
4          =     #Motor_Start
5          O     #Motor_SpeedHigh
6  n_OK:   ON    #Motor_BrakesEnabled
7          =     #Motor_Stop
```

Migration von Sprüngen in Verknüpfungsketten

Die Migration verändert die Sprungmarken Ihres AWL-Programms nicht. Beim Übersetzen wird jedoch ein Fehler gemeldet, wenn Sprungmarkenkonflikte erkannt werden. Überprüfen Sie an den gemeldeten Stellen die Programmlogik Ihres Bausteins. Verwenden Sie die Anweisungen SET oder CLR, um den Beginn einer neuen „logischen Einheit“ zu markieren und eine Erstabfrage zu forcieren.

Empfehlung

Vermeiden Sie Sprünge in Verknüpfungsketten, um z.B. den Programmcode zu optimieren. Der Code wird dadurch unübersichtlich und schwer zu pflegen.

3.4.4.3 Werteübergabe über Register bei Sprachwechsel

Einführung

Die Inhalte der Register, der Akkumulatoren und des Statusworts stehen nur innerhalb von AWL-Netzwerken zur Verfügung. Wenn ein KOP- oder FUP-Netzwerk auf ein AWL-Netzwerk folgt, können Sie aus dem KOP- oder FUP-Netzwerk heraus nicht auf die Registerinhalte zugreifen, die zuvor in AWL gesetzt wurden. In einem nachfolgenden AWL-Netzwerk stehen die Registerinhalte jedoch wieder zur Verfügung.

Eine Ausnahme bildet das VKE-Bit: Es wird bei einem Sprachwechsel auf "undefiniert" gesetzt und steht in nachfolgenden Netzwerken nicht mehr zur Verfügung.

Migration von Werteübergaben über Register bei Sprachwechsel

Wenn in einem migrierten KOP- oder FUP-Netzwerk auf Register zugegriffen wird, wird beim Übersetzen ein Fehler gemeldet. Ändern Sie das Programm ab, so dass Register nur in AWL-Netzwerken gesetzt und gelesen werden.

Beispiel

Das folgende Beispiel zeigt die Migration eines Akkumulatorzugriffs. In den ersten beiden Bildern sehen Sie das Programm vor der Migration:

Netzwerk 1: Der Operand "myIN1" wird in Akku 1 geladen. Anschließend wird "myIN2" in Akku 1 geladen, dadurch wird "myIN1" in Akku 2 transferiert. Nun werden beide Werte addiert. Das Ergebnis wird in Akku 1 abgelegt und von dort dem Operanden "myOUT" zugewiesen.

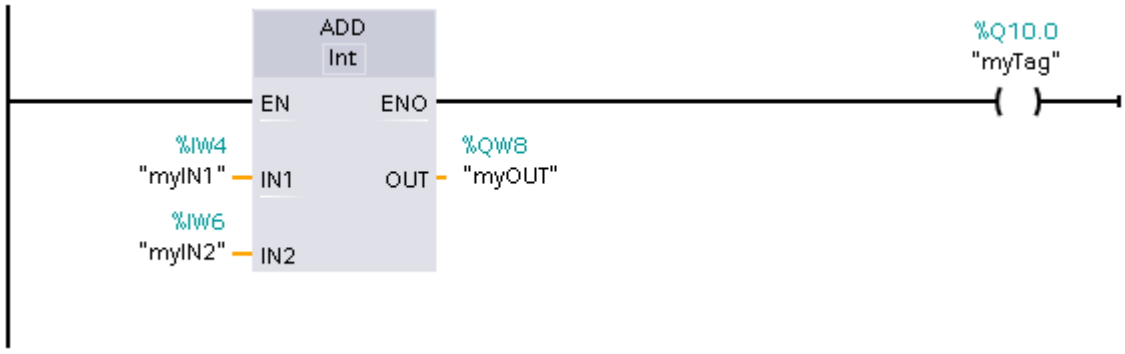

Netzwerk 2: Der Operand "myIN3" wird in Akku 1 geladen. Der dort noch vorhandene Operand "myOUT" wird dadurch in Akku 2 transferiert und kann sofort zu Operand "myIN3" addiert werden. Operand "myOUT" muss nicht explizit erneut geladen werden.

```
1    L    "myIN3"    %IW12
2    +I
3    T    "myResult"    %IW14
4
```

Nach der Migration wird Netzwerk 2 fehlerhaft, da durch das KOP-Netzwerk keine Werte mehr in die Akkumulatoren geschrieben werden. Das Programm muss manuell angepasst werden, so dass beide Werte explizit in AWL in die Akkumulatoren geladen werden.

Das folgende Bild zeigt Netzwerk 2 nach der Behebung des Fehlers.

Netzwerk 2: Beide Werte werden in AWL in die Akkumulatoren geladen, bevor sie addiert werden.

```
1    L    "myIN3"    %IW12
2    L    "myOUT"    %QW8
3    +I
4    T    "myResult"    %QW14
5
```

3.4.4.4 Werteübergabe über Register oder das Statuswort beim Bausteinaufruf

Migration von Werteübergaben über Register oder das Statuswort beim Bausteinaufruf

Die Werte aus Registern, Akkumulatoren und dem Statuswort werden beim Bausteinwechsel auf "0" gesetzt oder erhalten den Status "undefiniert". Somit können Sie nicht an aufgerufene Bausteine übergeben werden.

Eine Ausnahme besteht nur bei den Anweisungen "CC" und "UC". Wenn Sie "UC" oder "CC" verwenden und dem aufgerufenen Baustein Parameter über Register, Statuswort oder Akkumulatoren übergeben möchten, müssen Sie in den Eigenschaften des aufgerufenen Bausteins die Option "Parameterversorgung über Register" aktivieren. Beachten Sie, dass diese Option nur für AWL-Bausteine mit Standardzugriff zur Verfügung steht und der Baustein keine Formalparameter haben darf. Wenn die Option aktiviert ist, können Sie Registerinhalte

zwischen Bausteinen übergeben. Eine Ausnahme bildet das VKE-Bit: Es wird beim Bausteinübergang immer auf "undefiniert" gesetzt und steht nach einem Bausteinaufruf nicht mehr zur Verfügung.

Um eine Fehleraussage an den aufrufenden Baustein zu übergeben, können Sie das BIE-Bit nutzen. Die Fehleraussage müssen Sie zunächst im aufgerufenen Baustein BIE-Bit ablegen. Nutzen Sie dazu z. B. die Anweisungen "SAVE" oder "SPBNB". Anschließend können Sie das BIE-Bit im aufrufenden Baustein lesen.

Wenn Sie nach einem Bausteinaufruf auf Registerinhalte zugreifen, die im aufgerufenen Baustein gesetzt wurden, wird ein Fehler gemeldet. Ändern Sie dann Ihr Programm ab. Nutzen Sie z. B. Variablen in Datenbausteinen oder PLC-Variablen, um Werte an den aufrufenden Baustein zurückzugeben.

Beispiele

Das folgende Beispiel zeigt, wie Sie Ihr Programm ändern müssen, um Werte über Register an einen aufrufenden Baustein zu übergeben.

In der ersten Tabelle sehen Sie das Programm vor der Migration:

AWL	Erläuterung
CALL "MyFB", "MyFB_DB"	Das VKE des Bausteins "MyFB" wird nach dessen Bearbeitung dem Operanden "MyBit" zugewiesen.
= #MyBit	

In der zweiten Tabelle sehen Sie, wie Sie das Programm ändern müssen

AWL	Erläuterung
CALL "MyFB", "MyFB_DB"	Im aufgerufenen Baustein "MyFB" schreiben Sie an einer beliebigen Stelle das aktuell anstehende VKE mit der Anweisung "SAVE" in das BIE-Bit.
U BIE	Das BIE-Bit wird im aufrufenden Baustein gelesen.
= #MyBit	Der Wert des BIE-Bits wird dem Operanden "MyBit" zugewiesen.

3.4.4.5 Werteübergabe über Register mit den Anweisungen CC und UC

Einführung

In S7-300/400 ist es möglich, mithilfe der Anweisungen UC und CC Bausteinaufrufe zu programmieren. Die Parameterübergabe an den aufrufenden Baustein erfolgt dabei nicht über die Schnittstelle, sondern über Register, z. B. AR1, AR2, DB, DI, die Akkumulatoren oder das Statuswort.

Diese Aufrufe verlangsamen die Programmbearbeitung und sind deshalb in S7-1500 nicht mehr per Voreinstellung möglich. In S7-1500 wird stattdessen die Anweisung CALL verwendet. Mit der Anweisung CALL lassen sich jedoch keine indirekten Bausteinaufrufe realisieren.

Wenn Sie UC oder CC für indirekte Bausteinaufrufe verwenden möchten, müssen Sie in den Eigenschaften des aufgerufenen Bausteins die Option "Parameterversorgung über Register" aktivieren. Dann haben Sie die Möglichkeit, Registerinhalte zwischen verschiedenen Bausteinen zu übergeben. Eine Ausnahme bildet das VKE-Bit: Es wird beim Bausteinübergang auf "undefiniert" gesetzt und steht nach einem Bausteinaufruf nicht mehr zur Verfügung.

Beachten Sie, dass diese Option nur für AWL-Bausteine mit Standardzugriff zur Verfügung steht und der Baustein keine Formalparameter haben darf. Wenn die Option aktiviert ist, können Sie Registerinhalte zwischen Bausteinen übergeben. Eine Ausnahme bildet auch hier das VKE-Bit: Es wird beim Bausteinübergang auf "undefiniert" gesetzt und steht nach einem Bausteinaufruf nicht mehr zur Verfügung.

Bei der Erstellung know-how-geschützter Bibliothekselemente ist es nicht mehr notwendig, indirekte Bausteinaufrufe zu verwenden. Das TIA Portal unterstützt Sie hier durch einen Automatismus: Beim Einfügen eines Bausteins aus einer Bibliothek werden eventuell auftretende Nummernkonflikte mit Bausteinen, die sich bereits im Anwenderprogramm befinden, automatisch aufgelöst.

Migration von Bausteinaufrufen über "UC" oder "CC"

Die Migration behandelt Bausteinaufrufe über "UC" oder "CC" wie folgt:

- Die Anweisung "UC FC" mit Angabe einer Bausteinnummer wird durch die Anweisung "CALL" ersetzt.
- Die Anweisung "CC FC" mit Angabe einer Bausteinnummer wird durch die Anweisung "CALL" ersetzt. Zusätzlich wird ein Sprungbefehl eingefügt, der den bedingten Aufruf realisiert.
- Die Anweisungen "UC FC" und "CC FC" mit indirekter Angabe der Bausteinnummer bleiben unverändert.
- Die Anweisungen "UC FB" und "CC FB" mit direkter oder indirekter Angabe einer Bausteinnummer bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Migration von Bausteinaufrufen über "UC".

In der ersten Tabelle sehen Sie das Programm vor der Migration:

AWL
UC FC 10
UC FC[#temp0]
UC FB 10
UC FB [#temp0]

In der folgenden Tabelle sehen Sie das Programm nach der Migration:

AWL	Erläuterung
CALL FC 10	

AWL	Erläuterung
UC FC[#temp0]	Option "Parameterversorgung über Register" muss am aufgerufenen Baustein gesetzt sein.
UC FB 10	Option "Parameterversorgung über Register" muss am aufgerufenen Baustein gesetzt sein.
UC FB [#temp0]	Option "Parameterversorgung über Register" muss am aufgerufenen Baustein gesetzt sein.

3.4.4.6 Vollqualifizierte Adressen in AWL

Einführung

Die Adressierung von DB-Variablen mit Angabe des DB-Namens oder der DB-Nummer wird als vollqualifizierte Adressierung bezeichnet. Nach jedem Zugriff auf einen Datenbaustein mit Angabe einer vollqualifizierten Adresse wird das Datenbausteinregister auf "0" gesetzt. Wenn Sie nach einem vollqualifizierten Zugriff wieder auf das DB-Register zugreifen möchten, müssen Sie es zuerst wieder mit der Anweisung AUF DB mit einem Wert belegen.

Migration von vollqualifizierten Adressen

Die Migration fügt bei Bedarf nach einem vollqualifizierten Zugriff die Anweisung "AUF" ein, um den aktuellen Datenbaustein wieder in das Datenbausteinregister zu laden.

Beispiel

Das folgende Beispiel zeigt die Migration einer vollqualifizierten Adresse.

In der ersten Tabelle sehen Sie das Programm vor der Migration:

AWL	Erläuterung
AUF "MyDB"	"MyDB" wird in das Datenbausteinregister geladen.
L %DBW1	Datenelemente aus "MyDB" werden teilqualifiziert adressiert.
L "Global_DB".Data1	Daraufhin werden Elemente aus einem globalen DB vollqualifiziert adressiert. Dadurch wird implizit der "Global_DB" in das Datenbausteinregister geladen.
L "Global_DB".DBW2	
T DBW[AR1, P#0.0]	Ein nachfolgender Zugriff adressiert den "Global_DB", da er im DB-Register liegt.

In der folgenden Tabelle sehen Sie das Programm nach der Migration:

AWL	Erläuterung
AUF "MyDB"	

AWL	Erläuterung
L "MyDB".DBW1	Der teilqualifizierte Zugriff wird in einen vollqualifizierten Zugriff umgewandelt.
L "Global_DB".Data1	
L "Global_DB".DBW2	Nach dem vollqualifizierten Zugriff wird das DB-Register rückgesetzt.
AUF "Global_DB"	Der "Global_DB" muss mit AUF wieder in das DB-Register geladen werden.
T DBW[AR1, P#0.0]	

3.4.4.7 Teilqualifizierte Adressen in AWL

Einführung

Die Adressierung von DB-Variablen ohne Angabe des DB-Namens oder der DB-Nummer wird als teilqualifizierte Adressierung bezeichnet. Die teilqualifizierte Adressierung greift auf einen definierten Wert in dem Datenbaustein zu, der aktuell im DB-Register hinterlegt ist.

Für teilqualifizierte Adressen gelten in S7-1500 folgende Einschränkungen:

- In S7-1500 sind teilqualifizierte Adressen nur zulässig, wenn das DB-Register zuvor im aktuellen Baustein explizit gesetzt wurde. Das DB-Register setzen Sie z. B. mit der Anweisung "AUF". Sie können nur Variablen in Datenbausteinen mit Standardzugriff teilqualifiziert adressieren.
- Beim Bausteinaufruf werden in S7-1500 die Datenbausteinregister auf "0" gesetzt. Es ist somit nicht möglich, einen Datenbaustein in einem Baustein zu öffnen, und in einem unterlagerten Baustein Datenelemente aus dem Datenbaustein teilqualifiziert zu adressieren. Setzen Sie das DB-Register im aktuellen Baustein, bevor Sie eine DB-Variable teilqualifiziert adressieren.
- Auch bei jedem vollqualifizierten Zugriff (z. B. %DB10.DBW10) wird das Datenbausteinregister DB auf "0" gesetzt. Setzen Sie das DB-Register nach einem vollqualifizierten Zugriff erneut, bevor Sie eine DB-Variable teilqualifiziert adressieren.
- Durch die Verwendung teilqualifizierter Adressen verlangsamt sich die Programmbearbeitung zur Laufzeit.

Migration von teilqualifizierten Adressen

Die Migration behandelt teilqualifizierte Adressen in AWL wie folgt:

- Die teilqualifizierte Adresse wird möglich in eine vollqualifizierte Adresse umgewandelt.
- Wenn der Datenbaustein im aufrufenden Codebaustein geöffnet wurde und der DB nicht eindeutig bestimmbar ist, fügt die Migration in der Schnittstelle des aufgerufenen Bausteins einen Parameter vom Datentyp "DB_Any" ein. An diesem Parameter wird der Datenbausteinname übergeben. Die Migration fügt im aufgerufenen Baustein die Anweisung "AUF" ein, um den Datenbaustein zu öffnen.

Beispiel

Das folgende Beispiel zeigt die Migration von teilqualifizierten Adressen.

In der ersten Tabelle sehen Sie das Programm vor der Migration:

AWL	Erläuterung
L DBW10 L DBW12 +I T DBW14	Ein Datenbaustein wurde im aufrufenden Baustein aufgeschlagen und in das DB-Register übertragen. Im aktuellen Baustein werden die Werte "DBW10", "DBW12" und "DBW14" aus dem Datenbaustein, der aktuell im DB-Register vorliegt, übernommen.

In der folgenden Tabelle sehen Sie das Programm nach der Migration:

AWL	Erläuterung
AUF "PlcmigTempBlockDB" L DBW10 L DBW12 +I T DBW14	Die Migration fügt in der Schnittstelle des aufgerufenen Bausteins einen Parameter vom Datentyp "DB_Any" ein. An diesem Parameter wird der Datenbausteinname übergeben. Die Migration fügt im aufgerufenen Baustein die Anweisung "AUF" ein, um den an der Schnittstelle übergebenen Datenbaustein zu öffnen.

Hinweis

Migration von teilqualifizierten Bausteinparametern

Die Migration wandelt auch teilqualifizierte DB-Parameter in vollqualifizierte Parameter um. Beachten Sie, dass sich die Art der Parameterübergabe an den aufgerufenen Baustein dadurch ändern kann: Der aufgerufene Baustein greift über die vollqualifizierter Adresse möglicherweise nicht mehr direkt auf den Aktualparameter zu sondern arbeitet mit einer Kopie, die beim Bausteinaufruf übergeben wird.
Prüfen Sie deshalb, ob die Semantik des migrierten Programms noch mit der des Ausgangsprogramms übereinstimmt.
Weitere Informationen finden Sie unter: Zugriff auf Bausteinparameter während der Programmbearbeitung

Sollte diese Anpassung nicht gewünscht sein, können Sie zum Beispiel statt eines elementaren Bausteinparameters einen Parameter mit strukturiertem Datentyp verwenden. Definieren sie als Formalparameter PLC-Datentyp (UDT), und übergeben sie eine Variable dieses Typs bzw. einen DB, der von dem PLC-Datentypen (UDT) abgeleitet wurde.

Beispiele:

```
CALL  "MyFC"
          InStruct :="DBofUDT"
```

oder

```
CALL  "MyFC"
          InStruct := "DBArrayOfUDT".a[#i]
```

3.4.4.8 Zugriffe auf den Instanz-DB in AWL

Einführung

Die Anweisungen "AUF DI" oder "TDB" legen einen Datenbaustein in das DI-Register. In S7-300/400 gilt der Baustein, der dort aufgeschlagen ist, als Instanz-Datenbaustein. Die anschließende symbolische Adressierung eines lokalen Formalparameters aus der Bausteinschnittstelle eines FB (IN, OUT, InOut, Static) adressiert nicht mehr den Datenbaustein, der beim Bausteinaufruf als Instanz angegeben wurde, sondern den Datenbaustein, der im DI-Register liegt. Um anschließend einen lokalen Formalparameter aus der Bausteinschnittstelle symbolisch zu adressieren, muss der Instanz-DB in das DI-Register geladen werden.

Auch nach den Anweisungen "L AR2", "+ AR" und "TAR" ist es in S7-300/400 nicht möglich, einen Formalparameter aus der Bausteinschnittstelle symbolisch zu adressieren, da die Zugriffsbasis für den Parameterzugriff durch die Anweisungen zerstört wurde.

Dieses Verhalten wurde in S7-1500 korrigiert: Wenn Sie in S7-1500 einen lokalen Formalparameter aus der Bausteinschnittstelle symbolisch adressieren (z .B. mit der Anweisung L #myIn), greifen Sie immer auf den Datenbaustein zu, den Sie beim Bausteinaufruf als Instanz angegeben haben. Die Anweisungen AUF DI, L AR2, +AR2, TDB, TAR ändern zwar den Inhalt des DI- bzw. Adressregisters, die Register werden jedoch bei der Adressierung von lokalen Formalparametern nicht mehr ausgewertet.

Migration von Zugriffen auf lokale Variablen im Instanz-DB

Die Migration verändert die programmierten Zugriffe nicht. Wenn Sie im Ausgangsprogramm die Anweisungen AUF DI, L AR2, +AR2, TDB, TAR, etc. verwendet haben, ändert sich jedoch möglicherweise die Semantik des Programms.
Um die ursprüngliche Semantik wieder zu erreichen, müssen Sie das Programm manuell ändern. Häufig ist es nicht mehr notwendig, Daten über Register zu adressieren. Nutzen Sie stattdessen die Möglichkeit, ARRAYs innerhalb des Instanz-DB zu verwenden und die ARRAY-Elemente indirekt zu indizieren.

Beispiel 1

Das folgende Beispiel zeigt die geänderte Semantik der Anweisung "AUF DI":

AWL	Erläuterung
L #MyIn1 L #MyIn2 +I T #MyOut3	Die Anweisungen "L" und "T" adressieren lokale Variablen, die in der Bausteinschnittstelle deklariert wurden. Die Werte der Variablen liegen in dem Instanz-DB, der beim Bausteinaufruf angegeben wurde.
AUFDI "MyDB" L #MyIn1 L #MyIn2 +I T #MyOut3	Der globale Datenbaustein "MyDB" wird in das DI-Register geschrieben. Vor der Migration adressieren die Anweisungen "L" und "T" Variablen, die in "MyDB" deklariert wurden. Nach der Migration adressieren die Anweisungen "L" und "T" Variablen, die in der Bausteinschnittstelle deklariert wurden. Das DI-Register wird in S7-1500 für den Zugriff nicht ausgewertet.

Beispiel 2

Das folgende Beispiel zeigt die geänderte Semantik der Anweisung "LAR2":

AWL	Erläuterung
L P#M23.0	
LAR2	Vor der Migration wird mit der Zuweisung auf AR2 die Zugriffsbasis für Parameterzugriffe zerstört.

AWL	Erläuterung
L #MyIn1	Vor der Migration ist ein Zugriff auf "MyIn1" nicht möglich oder führt zu einem Fehler. Nach der Migration wird der Zugriff auf den Formalparameter "MyIn1" korrekt ausgeführt.
L EW [AR2, P#1.0]	Zugriff auf %EW24.0
U [AR2, P#0.4]	Zugriff auf %M23.4

Beispiel 3

Das folgende Beispiel zeigt, wie Sie in S7-1500 eine DB-Variable indirekt adressieren können, ohne die Adressregister zu verwenden:

AWL	Erläuterung
AUF "MyDB" L #index LAR1 L DBW [AR1 , P#10.0]	Vor der Migration wird die registerindirekte bereichsinterne Adressierung verwendet. Dabei wird ein variabler Wert (#index) in das Adressregister 1 geladen. Abhängig von diesem Wert wird ein Datenwort aus "MyDB" in den Akkumulator 1 geladen.
L "MyDB".MyArray1[#index]	Nach der Migration können Sie die Datenwerte in "MyDB" in einem ARRAY ablegen. Die einzelnen ARRAY-Elemente können variabel über den Eingangsparameter "#index" indiziert werden.

3.4.5 Migration von SCL-Programmen nach S7-1500

3.4.5.1 Hinweise zur Migration von SCL-Programmen

Darstellung von SCL-Bausteinen nach der Migration

SCL-Bausteine, die mit S7-300/400 erzeugt wurden, werden automatisch nach S7-1500 migriert, wenn ihre Semantik eindeutig interpretierbar ist. Programmstellen, die nicht eindeutig interpretierbar sind, werden nach der Migration mit Fragezeichen markiert, die zu Syntaxfehlern führen. Prüfen Sie die Semantik dieser Programmstellen nach der Migration und korrigieren Sie sie, wenn nötig.

In manchen Fällen wird der Programmcode durch die Migration verändert. Dann wird der entsprechende Codeabschnitt aus dem Ausgangsprogramm als Kommentarabschnitt in den migrierten Baustein übernommen. So können Sie die Änderungen, die durch die Migration vorgenommen wurden, leicht nachvollziehen.

Siehe auch

Grundlagen zur Migration von PLC-Programmen (Seite 153)

Hinweise zur Migration von PLC-Programmen (Seite 159)

3.4.5.2 Indirekte Adressierung in SCL

Einführung

In S7-1500 wurden die Möglichkeiten der indirekten Adressierung für alle Programmiersprachen vereinheitlicht. Indirekte Adressierungen, die in S7-300/400 möglich waren, werden deshalb teilweise während der Migration umgewandelt.

Die folgende Tabelle gibt einen Überblick über die Umwandlungsmöglichkeiten. Die Migration der einzelnen Sprachkonstrukte wird anschließend genauer erklärt.

Indirekte Adressierung	S7-300/400	S7-1500
Indirekte Adressierung eines DB	Datentyp "BLOCK_DB"	Datentyp "DB_ANY"
Indirekten Adressierung von DB-Variablen	#block.%DBW3	Anweisungen "PEEK"/"POKE"
	WORD_TO_BLOCK_DB(#myWord).%DBW3	Anweisungen "PEEK"/"POKE"
	#block.DW(IDX := #myInt)	Anweisungen "PEEK"/"POKE"
	WORD_TO_BLOCK_DB(#myWord).DW(IDX:=#myInt)	Anweisungen "PEEK"/"POKE"
	%DB1.DW(IDX :=#myInt)	Anweisungen "PEEK"/"POKE"
Indirekte Adressierung von Peripherie	QB(IDX :=#myInt):P	Anweisungen "PEEK"/"POKE"
Indirekte Adressierung von PLC-Variablen	IX(IDX :=#myInt1,Bit:=#myInt2)	Anweisungen "PEEK"/"POKE"
	QB(IDX :=#myInt)	Anweisungen "PEEK"/"POKE"
	MW(IDX :=#myInt)	Anweisungen "PEEK"/"POKE"

Migration des Datentyps "BLOCK_DB" nach "DB_ANY"

Der Datentyp "BLOCK_DB" steht in S7-1500 nicht mehr zur Verfügung. Variablen dieses Datentyps werden von der Migration in den Datentyp "DB_ANY" umgewandelt. Die Konvertierungsfunktionen "BLOCK_DB_TO_WORD" und "WORD_TO_BLOCK_DB" werden nach "UINT_TO_WORD(DB_ANY_TO_UINT)" und "UINT_TO_DB_ANY(WORD_TO_UINT)" migriert.

Die folgenden Beispiele zeigen die Migration des Datentyps "BLOCK_DB".

In der ersten Tabelle sehen Sie drei Programmausschnitte vor der Migration:

SCL

```
#myBlock  //Datentyp BLOCK_DB;
//...
#myWord :=BLOCK_DB_TO_WORD(#myBlock);
//...
#myBlockDB := WORD_TO_BLOCK_DB(#myWord);
```

In der folgenden Tabelle sehen Sie die drei Programmausschnitte nach der Migration:

SCL

```
#myBlock  //Datentyp DB_Any;
//...
#myWord := UINT_TO_WORD(DB_ANY_TO_UINT(#myBlock));
//...
#myDBANY:= UINT_TO_DB_ANY(WORD_TO_UINT(#myWord));
```

Migration der indirekten Adressierung von DB-Variablen

Indirekte Angaben von Datenbausteinen oder DB-Variablen werden in S7-1500 über die Anweisungen "PEEK"/"POKE" implementiert. Die Migration wandelt diese indirekten Adressierungen automatisch um.

Die folgenden Beispiele zeigen die Migration der indirekten Adressierung von DB-Variablen.

In der ersten Tabelle sehen Sie einige Programmausschnitte vor der Migration:

SCL

```
#myWord := %DB1.DW(IDX :=#myInt);
//...
#myBool := %DB1.DX(IDX :=#myByteOffset,Bit:=#myBitOffset);
//...
%DB1.DW(IDX := #myInt) := 12;
//...
```

In der folgenden Tabelle sehen Sie die Programmausschnitte nach der Migration:

SCL

```
#myWord := PEEK_WORD(area:=16#84, dbNumber:=1, byteOffset:=#myInt);
//...
#myBool:=PEEK_BOOL(area:=16#84, dbNumber:=1, byteOffset:=#myByteOffset, bitOffset:=#myBitOffset);
//...
POKE(area:=16#84,dbNumber:=1,byteOffset:=#myInt,value:=12);
//...
```

Migration der indirekten Adressierung von PLC-Variablen

Indirekte Angaben von PLC-Variablen werden in S7-1500 über die Anweisungen "PEEK"/"POKE" implementiert. Die Migration wandelt diese indirekten Adressierungen automatisch um.

Die folgenden Beispiele zeigen die Migration der indirekten Adressierung von DB-Variablen.

In der ersten Tabelle sehen Sie einen Programmausschnitt vor der Migration:

SCL

```
#myWord := MW(IDX := #myInt);
```

In der folgenden Tabelle sehen Sie die den Programmausschnitte nach der Migration:

SCL

```
#myWord := PEEK(area:=16#83,dbNumber:=0,byteOffset:=#myInt);
```

3.4.5.3 Anweisungen in SCL

Migration von Anweisungen

S7-1500 bietet einen leicht geänderten Satz von Anweisungen. Einige Anweisungen wurden innoviert oder durch neuere Anweisungen ersetzt.

Während der Migration werden die im Programm verwendeten Anweisungen soweit möglich übernommen. Falls die Anweisung in S7-1500 nicht verfügbar ist, wird versucht, sie durch eine kompatible oder ähnliche Anweisung zu ersetzen. Sie müssen dann das migrierte Programm noch einmal überprüfen und ggf. anpassen.

So können zum Beispiel folgende Änderungen vorkommen:

- Die neue Anweisung hat zusätzliche Formalparameter. Versorgen Sie die zusätzlichen Parameter.
- Ein Formalparameter der neuen Anweisung hat einen anderen Datentyp. In dem Fall ergänzt die Migration automatisch eine Anweisung zur Konvertierung der Datentypen.

Wenn die Migration nicht eindeutig sicherstellen kann, dass die Parameterversorgung mit dem Ausgangsprogramm semantisch identisch ist, werden die betreffenden Parameter auskommentiert. Prüfen Sie die Semantik dieser Programmstellen nach der Migration und korrigieren Sie sie wenn nötig.

Die folgende Beispiele zeigen die Migration der Anweisung "AG_SEND" in die Anweisung "TSEND".

In der ersten Tabelle sehen Sie den Aufruf von "AG_SEND" vor der Migration:

SCL

```
AG_SEND (ACT:=#myBool,
         ID:=#myInt1,
         LADDR:=#myWord,
         SEND:=#myAny,
         LEN:=#myInt2,
         DONE=>#myBool,
         ERROR=>#myBool,
         STATUS=>#myWord);
```

In der zweiten Tabelle sehen Sie den Aufruf von "TSEND" nach der Migration:

SCL

```
"TSEND_DB"((* ToReplace: REQ:=#myBool *)
         (* ToReplace: ID:=#myInt1 *)
         LEN:=INT_TO_UDINT(#myInt2),
         DONE=>#myBool,
         ERROR:=#myBool,
         STATUS:=#myWord,
         DATA:=#myAny;
```

Der Aktualparameter von "REQ" wurde vom Parameter "ACT" der Anweisung "AG_SEND" übernommen. Er ist auskommentiert, weil das Programm an dieser Stelle geprüft werden muss.

Der Aktualparameter von "ID" wurde von "AG_SEND" übernommen und muss ebenfalls geprüft werden.

Der Parameter "LEN" hatte bei "AG_SEND" einen anderen Datentyp. Die Migration fügt automatisch eine Konvertierung ein.

Migration von Anweisungen in Ausdrücken

In einigen Fällen erzeugt die Migration aus einer Anweisung mehrere Anweisungen, z. B. wenn ein Ausgangsparameter nach der Migration einen anderen Datentyp hat und der Formalparameter konvertiert werden muss. Die Konvertierungsanweisung wird nach der migrierten Anweisung eingefügt. Dadurch lässt sich jedoch die ursprüngliche Anweisung nicht mehr als Ausdruck verwenden. In diesem Fall ergänzt die Migration eine temporäre Variable in der Schnittstelle des migrierten Bausteins und weist ihr das Ergebnis der Anweisung zu. Diese temporäre Variable kann dann als Ausdruck verwendet werden. Bei komplexen Ausdrücken kann nicht immer eine automatische Migration durchgeführt werden. Ausdrücke, die nicht eindeutig interpretierbar sind, werden nach der Migration mit Fragezeichen markiert, die zu Syntaxfehlern führen. Prüfen Sie die Semantik dieser Programmstellen nach der Migration und korrigieren Sie sie wenn nötig.

Die folgenden Beispiele zeigen die Migration der Anweisung "READ_RTM" in die Anweisung "RTM".

In der ersten Tabelle sehen Sie den Aufruf von "READ_RTM" vor der Migration:

SCL

```
IF READ_RTM(NR:=#myByte, CQ=>#myBool, CV=>#myOutInt)= 1 THEN
...;
END_IF;
```

In der zweiten Tabelle sehen Sie den Aufruf von "RTM" nach der Migration.

SCL

```
(* classic code: IF READ_RTM(NR:=#myByte, CQ=>#myBool, CV=>#myOutInt)= 1 THEN*)
#SCL_MIGRA_TEMP_INT_1:=RTM(NR:=#myByte,
                           MODE:=0,
                           PV:=#Migra_PV,
                           CQ:=#myBool,
                           CV:=#Migra_CV);
#myOutInt:=DINT_TO_INT(#Migra_CV);

IF #SCL_MIGRA_TEMP_INT_1= 1 THEN;
...
END_IF;
```

Die Anweisung aus dem Ausgangsprogramm wird als Kommentarabschnitt in den migrierten Baustein übernommen. So können Sie die Änderungen, die durch die Migration vorgenommen wurden, leicht nachvollziehen.

Da CV bei "RTM" den Datentyp DINT hat, wird eine Konvertierungsanweisung eingefügt, die den Aktualparameter von INT nach DINT konvertiert. Dadurch ist das Ergebnis von RTM nicht mehr als Ausdruck innerhalb der IF-Anweisung verwendbar.

In der Schnittstelle des migrierten Bausteins wird die temporäre Variable "SCL_MIGRA_TEMP_INT_1" eingefügt. Das Ergebnis der Anweisung RTM wird der temporären Variablen zugewiesen.

Anstelle der Anweisung "RTM" wird die temporäre Variable als Ausdruck in der IF-Anweisung verwendet.

3.4.6 Migration von GRAPH-Programmen nach S7-1500

3.4.6.1 Hinweise zur Migration von GRAPH-Programmen

Einführung

GRAPH-Programme, die mit S7-300/400 erzeugt wurden, können automatisch nach S7-1500 migriert werden. Durch die PLC-Migration werden die Programme für den Ablauf auf einer CPU der Baureihe S7-1500 optimiert. Das Migrationsprotokoll meldet alle automatischen Programmänderungen, die während der Migration gemacht werden und informiert Sie über eventuell notwendige manuelle Änderungen, die Sie anschließend noch durchführen müssen.

Anpassung des GRAPH-Programms durch die PLC-Migration

Die folgende Liste zeigt einige Beispiele für automatische Anpassungen, die während der PLC-Migration durchgeführt werden. Alle Anpassungen werden in den nachfolgenden Kapiteln ausführlich beschrieben.

- Anpassung der Bausteinschnittstelle
  Die Schnittstelle der GRAPH-Bausteine wird leicht abgeändert. Dadurch wird es möglich, für GRAPH-Bausteine den optimierten Bausteinzugriff der S7-1500 zu nutzen.
- Durchgängige symbolische Adressierung
  SIMATIC S7-1500 zeichnet sich durch eine durchgängige symbolische Programmierung aus. Das Programm wird dadurch performanter und die Gefahr von Zugriffsfehlern wird reduziert.
  Wenn das Ausgangsprogramm Operanden verwendet, für die kein symbolischer Name deklariert ist, vergibt die Migration deshalb wenn möglich einen symbolischen Namen für den Operanden.
- Speichermodell des GRAPH-DB
  Aufgrund der Leistungsdaten der S7-1500 besteht nicht mehr die Notwendigkeit, speicherplatzminimierte GRAPH-DBs zu verwenden. GRAPH bietet deshalb keine unterschiedlichen Speicherplatzmodelle mehr an. In allen GRAPH-DBs steht Ihnen der volle Parametersatz und somit die volle Funktionalität zur Verfügung.
  Während der Migration werden speicherplatzminimierte DBs auf den vollen Parametersatz erweitert.

Neue Funktionen der S7-1500

Nach der Migration eines GRAPH-Programms können Sie die Funktionalität der S7-1500 voll ausnutzen. Die folgende Liste zeigt einige Beispiele der neuen Funktionen:

- Optimierter Bausteinzugriff
  Bei optimiertem Bausteinzugriff werden die deklarierten Datenelemente im verfügbaren Speicherbereich des Bausteins automatisch so angeordnet, dass seine Kapazität optimal ausgeschöpft wird. Die Daten werden auf eine Weise strukturiert und abgelegt, die für die verwendete CPU optimal ist. Dadurch steigern Sie die Performance der CPU.
  Zugriffsfehler, z. B. aus HMI heraus, sind nicht möglich.
  Das Attibut "optimierter Bausteinzugriff" ist für GRAPH-Bausteine in S7-1500 immer aktiviert und kann nicht abgewählt werden.
  Siehe auch: AUTOHOTSPOT
- Neue Anweisungen
  In der Task Card "Anweisungen" steht Ihnen der komplette Satz von Anweisungen der S7-1500 zur Verfügung. Zahlreiche Anweisungen wurden neu entwickelt oder innoviert. So finden Sie zum Beispiel in der Palette "Einfache Anweisungen" im Ordner "Programmsteuerung" die Anweisung "GET_ERROR" zur Programmierung einer lokalen Fehlerbehandlung im Baustein.
  Siehe auch: AUTOHOTSPOT

- KOP-/FUP-Funktionen
  In GRAPH für S7-300/400 gab es nur die Möglichkeit, am ersten booleschen Eingang einer Box eine Vorverknüpfung zu programmieren. In S7-1500 sind alle booleschen Eingänge vorverknüpfbar.
  Der Freigabeausgang ENO kann gezielt für einzelne Anweisungen aktiviert oder deaktiviert werden.
  Siehe auch: AUTOHOTSPOT
- Neue Datentypen
  S7-1500 bietet einen erweiterten Satz von Datentypen und neue Möglichkeiten der Datentypkonvertierung.
  Siehe auch: AUTOHOTSPOT
- Konstanten
  S7-1500 bietet zusätzliche Möglichkeiten der Konstantennotation. Damit können Sie auch Konstanten Datentypen zuweisen.
  Siehe auch: AUTOHOTSPOT

3.4.6.2 Änderungen im GRAPH-Programm

Einführung

Die PLC-Migration passt das GRAPH-Programm automatisch so weit wie möglich an die neue CPU-Familie an. Dabei werden sowohl die Aktionen als auch die KOP-/oder FUP-Netzwerke, die Sie zur Programmierung von Bedingungen eingesetzt haben, analysiert. Programmstrukturen, die nicht mehr zulässig sind, werden aktualisiert. Anweisungen, die auf der neuen CPU nicht verfügbar sind, werden durch entsprechende Anweisungen der S7-1500 ersetzt.

Symbolische Adressierung

SIMATIC S7-1500 zeichnet sich durch eine durchgängige symbolische Programmierung aus. Das Programm wird dadurch performanter und die Gefahr von Zugriffsfehlern wird reduziert.

Wenn das Ausgangsprogramm absolute Operanden verwendet, für die kein symbolischer Name deklariert ist, wird der absolute Operand als ungültig markiert. Sie müssen dann einen symbolischen Namen deklarieren, bevor Sie das migrierte Programm übersetzen können.

Wenn das Ausgangsprogramm absolute Operanden verwendet, für die ein symbolischer Name deklariert ist, setzt die Migration anstelle des absoluten Operanden den symbolischen Namen ein. Zugriffe auf einzelne Bits innerhalb eines deklarierten symbolischen Operanden werden durch Slice-Zugriffe ersetzt.

Das folgende Beispiel zeigt einen absoluten Zugriff vor der Migration:

```
%DB1.DBx1.1
```

Im Folgenden sehen Sie, wie die Migration den absoluten Zugriff ersetzt. Da in "DataBlock1" die Variable "FirstDataByte" für das erste Datenbyte deklariert war, kann der Zugriff auf das erste Bit innerhalb dieser Variable durch einen Slice-Zugriff ersetzt werden.

```
"DataBlock1".FirstDataByte.x1
```

Vollqualifizierte Adressierung

S7-300/400-Programme verwenden häufig teilqualifizierte Adressen, um indirekt auf Operanden zuzugreifen. Dabei wird nur die Adresse einer DB-Variablen innerhalb eines Datenbausteins angegeben. Die Angabe des DB-Namens oder der DB-Nummer ist nicht in der Adresse enthalten (z. B. `%DBX0.2`). Zur Laufzeit greift das Programm auf den Datenbaustein zu, der aktuell im DB-Register hinterlegt ist. Teilqualifizierte Adressierung wird in S7-1500 nicht unterstützt. Um DB-Variablen indirekt zu adressieren, verwenden Sie Bausteinparameter vom Datentyp "DB_ANY".

Falls Sie teilqualifizierte Adressen in GRAPH-Programmen verwendet haben, werden diese nach der Migration als Fehler gemeldet. Der Baustein kann nicht übersetzt werden. Ersetzen Sie dann die teilqualifizierte durch eine vollqualifizierte Adresse.

KOP- /FUP-Elemente in Bedingungen

Die KOP- oder FUP-Netzwerke in Transitionen, Interlock, Supervision und permanenten Anweisungen werden nach denselben Regeln migriert wie die Netzwerke in reinen KOP- oder FUP-Bausteinen.

Siehe auch: Migration von KOP/FUP-Programmen nach S7-1500 (Seite 197)

GRAPH-Aktionen

SIMATIC S7-1500 bietet einen leicht geänderten Satz von Anweisungen. Einige Anweisungen wurden innoviert oder durch neuere Anweisungen ersetzt.

Wenn Sie in GRAPH-Aktionen Anweisungen verwendet haben, prüft die Migration, ob diese auf S7-1500 weiterhin unterstützt werden. Die Anweisungen werden soweit möglich übernommen. Falls die Anweisung in S7-1500 nicht verfügbar ist, wird versucht, sie durch eine kompatible oder ähnliche Anweisung zu ersetzen.

Beispiele hierfür sind die Anweisungen "RLDA: Durch Statusbit A1 nach links rotieren" und "RRDA: Durch Statusbit A1 nach rechts rotieren". Sie werden durch die Anweisungen"ROR: Rechts rotieren" und "ROL: Links rotieren" ersetzt.

Wenn Anweisungen durch die Migration ersetzt werden, erhalten Sie einen Hinweis im Migrationsprotokoll. Sie müssen dann das migrierte Programm noch einmal überprüfen und ggf. anpassen.

Siehe auch: Migration von Anweisungen nach S7-1500 (Seite 167)

3.4.6.3 Anpassung der Schnittstelle von GRAPH-Bausteinen

Änderung der Bausteinschnittstelle während der Migration

Die Schnittstelle der GRAPH-Bausteine wird während der Migration angepasst. Durch die Anpassungen können Sie mit GRAPH das optimierte Speicherkonzept der S7-1500 nutzen. Der optimierte Bausteinzugriff ist für GRAPH-Bausteine auf S7-1500 immer aktiv und kann nicht abgewählt werden. Änderungen werden in folgenden Bereichen durchgeführt:

- Statische Parameter: "STATIC"
- Ausgangsparameter: "OUTPUT"
- Remanenzeinstellung bei allen internen Parametern

Siehe auch: Bausteine mit optimiertem Zugriff

Wenn Sie im Ausgangsprogramm Parameternamen verwendet haben, die in S7-1500 reserviert sind, werden diese während der Migration umbenannt.

Statische Parameter

Die statischen Parameter werden während der Migration wie folgt angepasst:

- Die Schnittstelle enthält für jeden Schritt und jede Transition des GRAPH-FB eine Struktur im Abschnitt "Static". In S7-300/400 wurden die Datentypen "GraphStep", "GraphStepMin", "GraphTransition" und "GraphTransitionMin" verwendet. In S7-1500 werden die neuen Datentypen "G7_StepPlus" und "G7_TransitionPlus" zur Abbildung der Schritt- und Transitionsstrukturen eingeführt.
  Die Migration verfährt hier wie folgt:
  - Die Parameter "SNO" und "TNO" werden mit ihren Defaultwerten aus den alten Strukturen übernommen, alle weiteren Defaultwerte werden nicht transferiert. GRAPH-Programme mit speicherplatzminimiertem DB beinhalten die Parameter "SNO" und "TNO" nicht. Wenn Sie einen speicherplatzminimierten DB migrieren, vergibt die Migration automatisch bei "1" beginnende Defaultwerte für diese Parameter.
  - Die Verwendungsstellen der geänderten Parameter im GRAPH-Programm werden nicht automatisch angepasst, sondern als Fehler markiert. Sie müssen die Verwendungsstellen nach der Migration manuell anpassen.
- Alle weiteren statischen Parameter werden in die Struktur "RT_DATA" transferiert. Die Defaultwerte werden soweit möglich übernommen und die Verwendungsstellen im Programm automatisch angepasst.

Ausgangsparameter

Da der Ausgangsparameter "S_CRITSUP" bei der Kriterienanalyse auf S7-1500 nicht mehr unterstützt wird, wird er während der Migration entfernt.

Die Verwendungsstellen im GRAPH-Programm werden nicht automatisch angepasst, sondern als Fehler markiert. Sie müssen die Verwendungsstellen nach der Migration manuell anpassen.

Remanenzeinstellung bei allen internen Parametern

Die Remanenzeinstellung der internen GRAPH-Parameter beeinflusst das Verhalten der Ablaufsteuerung nach einem Spannungsverlust:

- Ein GRAPH-Baustein, dessen Parameter remanent sind, läuft nach einem Spannungsverlust mit dem letzten aktiven Schritt weiter.
- Ein GRAPH-Baustein, dessen Parameter nicht remanent sind, startet nach einem Spannungsverlust wieder mit dem Initialschritt.

GRAPH-Bausteine, die mit S7-300/400 angelegt wurden, haben per Voreinstellung remanente Schnittstellenparameter.

Die Remanenz der internen Parameter wird nicht mitgeführt, wenn Sie den GRAPH-FB von einer S7-300/S7-400 CPU auf eine S7-1500 CPU migrieren. Die internen Parameter sind nach der Migration als nicht-remanent deklariert.

Um die internen Parameter eines GRAPH-FB wieder remanent zu schalten, gehen Sie nach der Migration folgendermaßen vor:

1. Öffnen Sie nach der Migration den GRAPH-FB.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Interne Parameter remanent".
   In der Bausteinschnittstelle des GRAPH-FBs ändern sich die Einstellungen für die internen Parameter in der Spalte "Remanenz" von "Nicht remanent" nach "Remanent".
3. Speichern und übersetzen Sie Ihr Projekt.
   Die Einstellungen für die Parameter werden dann für alle Instanz-DBs übernommen.

S7-1200 auf Firmware ab V4 migrieren 4

4.1 Grundlagen zum Umstieg auf V4

Einführung

Wenn Sie in Ihrem Projekt eine CPU mit dem Firmwarestand V3 verwendet haben und auf eine CPU mit einem Firmwarestand V4.0 oder höher umsteigen möchten, können Sie die Geräte auf einfache Weise austauschen.

Das TIA Portal bietet Ihnen hierzu die Funktion "Gerät tauschen". Beim Gerätetausch bleibt das Projekt unverändert erhalten. Die Programme, die Sie mit dem Firmwarestand V3 erstellt haben, können Sie weiter verwenden.

Regeln

Beim Gerätetausch gelten folgende grundsätzliche Regeln:

- Der Gerätetausch ist nur möglich, wenn das Projekt auf Basis einer CPU mit dem Firmwarestand V3.0 erstellt wurde. Falls Ihr Projekt auf Basis der Firmwarestände V1.0 oder V2.0 erstellt wurde, erzeugen Sie offline im Projekt eine neue CPU mit dem Firmwarestand V3.0 und kopieren Sie Ihr Programm in diese CPU.
- Der Austausch einer CPU V4 gegen eine CPU V3 ist nicht möglich. Wenn Sie die bisherige CPU V3 weiter verwenden wollen, erstellen Sie vor dem Gerätetausch eine Kopie dieser CPU.
- Das Programm kann nicht über eine Memory Card auf die neue CPU übertragen werden. Verwenden Sie stattdessen die Funktion "Gerät tauschen", die in den nachfolgenden Kapiteln beschrieben ist.

HMI-Panels

Projektierte HMI-Panels werden beim Gerätetausch abhängig von der Firmware-Version des Panels und der Kommunikationsart unterschiedlich behandelt.

Die folgende Tabelle zeigt, welche HMI-Verbindungen beim Umstieg unterstützt werden:

Firmware-Version des Panels	PUT/GET-Kommunikation	Umstieg auf V4
V11 oder älter	nein	S7-1200 unterstützt diese Konfiguration nicht. Rüsten Sie die Firmware des HMI-Panels auf V12.0 hoch. Übersetzen und laden Sie die Konfiguration anschließend.
V11 oder älter	ja	S7-1200 unterstützt diese Konfiguration. Die Verbindung wird automatisch aufgebaut, während Sie das Projekt nach dem Gerätetausch übersetzen und laden.
V12 oder höher	nein	S7-1200 unterstützt diese Konfiguration. Die Verbindung wird automatisch aufgebaut, während Sie das Projekt nach dem Gerätetausch übersetzen und laden.

Beim Übersetzen des Programms erhalten Sie gezielte Hinweise zur Migration der HMI-Panels.

Hinweis

HMI TP 177B 4"

Das HMI TP 177B 4" mit Firmware Version V11.0.2 kann nicht mit S7-1200 V4 betrieben werden. Ersetzen Sie das Panel bei Bedarf mit einem neueren Gerät.

S7-1200 Erweiterungsmodule

Wenn Sie folgende zentral gesteckten S7-1200-Module bereits in Ihrer Anlage verwenden, müssen Sie ein Firmware-Update für diese Module durchführen, um den Betrieb mit S7-1200 V4 zu gewährleisten.

- ASi - Master - CM 1243
- DP - Master - CM 1243-5
- WAN CP - CP1243-1

Neu ausgelieferte S7-1200-Module sind bereits ab Werk mit der neuesten Firmware ausgestattet.

Geschützte Bausteine

Bausteine, die einen Know-How-Schutz oder einen Kopierschutz tragen, können nicht auf V4 umgestellt werden. Wenn das Projekt geschützte Bausteine enthält, müssen Sie den Schutz vor dem Umstieg entfernen.

Handelt es sich um zugelieferte Bausteine, deren Passwort Sie nicht kennen, fordern Sie bei Ihrem Lieferanten entweder das Passwort oder einen V4-kompatiblen Baustein an.

⚠ WARNUNG

Personen- und Sachschaden verhindern

Während des Gerätetauschs werden in manchen Fällen Änderungen am Programm vorgenommen. Prüfen Sie das Programm nach dem Gerätetausch deshalb ausführlich in einer Testumgebung, bevor Sie es in den laufenden Betrieb übernehmen.

Hinweis

Weitere Unterstützung

Im Siemens Industry Online Support finden Sie aktuelle FAQs zum Umstieg auf S7-1200 V4:

http://support.automation.siemens.com/WW/view/de/82140966 (http://support.automation.siemens.com/WW/view/de/82140966)

Falls Sie weitere Unterstützung zum Umstieg auf S7-1200 V4 benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Siehe auch

Migration auf V4 durchführen (Seite 228)

Besonderheiten nach dem Umstieg auf V4 (Seite 230)

4.2 Migration auf V4 durchführen

Voraussetzung

- Eine CPU mit Firmwarestand V3 ist im Projekt vorhanden.
- Das Projekt enthält keine geschützten Bausteine.

Vorgehen

Um eine CPU auszutauschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU V3, die Sie austauschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Gerät tauschen".
   Das Dialogfeld "Gerät tauschen" wird geöffnet.
3. Selektieren Sie in der Baumstruktur unter "Neues Gerät" die CPU V4, durch die Sie Ihre aktuelle CPU V3 ersetzen möchten.
4. Klicken Sie auf "OK".
   Die vorhandene CPU wird durch die neue CPU ersetzt.
5. Selektieren Sie die neue CPU und wählen Sie im Kontextmenü den Befehl "Übersetzen > Hardware und Software (nur Änderungen)."
   Die Gerätekonfiguration und das Anwenderprogramm werden neu übersetzt.
6. Optional: Setzen Sie, falls erforderlich, einen Know-how-Schutz bzw. Kopierschutz für einzelne Bausteine im Programm.
7. Selektieren Sie die neue CPU und wählen Sie im Kontextmenü den Befehl "Laden in Gerät> Hardware und Software (nur Änderungen)."
   Die Gerätekonfiguration und das Anwenderprogramm werden in die neue CPU geladen.
   Damit ist der Gerätetausch abgeschlossen.

⚠ WARNUNG
Personen- und Sachschaden verhindern
Während des Gerätetauschs werden in manchen Fällen Änderungen am Programm vorgenommen. Prüfen Sie das Programm nach dem Gerätetausch deshalb ausführlich in einer Testumgebung, bevor Sie es in den laufenden Betrieb übernehmen.

Hinweis

Weitere Unterstützung

Im Siemens Industry Online Support finden Sie aktuelle FAQs zum Umstieg auf S7-1200 V4:

http://support.automation.siemens.com/WW/view/de/82140966 (http://support.automation.siemens.com/WW/view/de/82140966)

Falls Sie weitere Unterstützung zum Umstieg auf S7-1200 V4 benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Siehe auch

Grundlagen zum Umstieg auf V4 (Seite 225)

Besonderheiten nach dem Umstieg auf V4 (Seite 230)

4.3 Besonderheiten nach dem Umstieg auf V4

Funktionsänderungen der V4

S7-1200 V4 bietet einen deutlich erweiterten Funktionsumfang. Die wichtigsten Funktionsänderungen, die Sie nach dem Umstieg von V3 beachten müssen, sind im Folgenden kurz beschrieben.

Weitere Informationen zu S7-1200 finden Sie im "Systemhandbuch SIMATIC S7-1200 Automatisierungssystem":

https://support.industry.siemens.com/cs/ww/de/view/65601780 (http://support.automation.siemens.com/WW/view/de/65601780)

Organisationsbausteine

Mit S7-1200 V4 können Sie gezielt die Unterbrechbarkeit jedes verwendeten Organisationsbausteins einstellen. Beim Gerätetausch werden alle Organisationsbausteine als nicht unterbrechbar konfiguriert, um sicherzustellen, dass Ihr V3-Programm unverändert ablauffähig bleibt. Auch die OB-Prioritäten werden unverändert aus dem V3-Programm übernommen. Nach dem Umstieg können Sie bei Bedarf die Einstellungen zur Priorität und zur Unterbrechbarkeit ändern.

Das Verhalten von Diagnosealarmen hat sich in V4 wie folgt geändert:

In V3 enthielt die Startinformation immer Angaben zum auslösenden Modul einschließlich der Kanalnummer. In V4 werden diese beiden Angaben nur bei anstehendem Diagnoseereignis ausgegeben. Wenn kein Diagnoseereignis ansteht, z. B. weil die Störung bereits behoben ist, wird nur das auslösende Modul ausgegeben.

Zugriffsstufen

S7-1200 V4 bietet ein erweitertes Zugriffsstufen-Konzept an. Die folgende Tabelle zeigt, wie die Schutzstufen der Firmware V3 in V4 abgebildet werden:

V3 Schutzstufe	V4 Zugriffsstufe	Bedeutung
Kein Schutz	Vollzugriff (kein Schutz)	Uneingeschränkter Zugriff ohne Passwortschutz.
Schreibschutz	Lesezugriff	HMI-Zugriff und ungehinderte Kommunikation zwischen CPUs ohne Passwort-schutz. Ein Passwort ist für Änderungen (Schreibzugriffe) in der CPU und für den Wechsel des Betriebszustands der CPU (RUN/STOP) erforderlich.

V3 Schutzstufe	V4 Zugriffsstufe	Bedeutung
Schreib-/Lese-schutz	HMI-Zugriff	HMI-Zugriff und ungehinderte Kommunikation zwischen CPUs ohne Passwort-schutz. Ein Passwort ist zum Lesen der Daten in der CPU, für Änderungen (Schreiben) in der CPU und für den Wechsel des Betriebszustands der CPU (RUN/STOP) erforderlich.
-	Kein Zugriff (kom-pletter Schutz)	Kein Zugriff ohne Passwortschutz. Ein Passwort ist für den HMI-Zugriff, zum Lesen der Daten in der CPU, für das Ändern (Schreiben) von Daten in der CPU und für den Wechsel des Betriebszus-tands der CPU (RUN/STOP) erforderlich.

Anweisungsbibliotheken

Nach dem Umstieg auf S7-1200 V4 stehen die Anweisungen aus den Bibliotheken der Firmware-Version V3 weiterhin zur Verfügung. Damit ist gewährleistet, dass Sie Ihr Programm unverändert weiter verwenden können. Darüber hinaus bietet die S7-1200 V4 zahlreiche neue Anweisungen, die auch mit den Anweisungen der S7-1500 kompatibel sind.

Weitere Informationen zu den Anweisungsbibliotheken finden Sie im "Systemhandbuch SIMATIC S7-1200 Automatisierungssystem":

https://support.industry.siemens.com/cs/ww/de/view/65601780 (http://support.automation.siemens.com/WW/view/de/65601780)

Motion Control

Beim Gerätetausch werden die Motion Control-Objekte aus den Bibliotheken der Firmware-Versionen V1 und V2 gegen die entsprechenden Objekte aus den V3-Bibliotheken ausgetauscht. Die Objekte aus V3-Bibliotheken sind kompatibel, so dass Sie die Programme unverändert weiter verwenden können.

Die Bibliotheken der S7-1200 V4 bieten zahlreiche neue Motion Control-Funktionen, die auch mit den Funktionen der S7-1500 kompatibel sind. Wenn Sie V4-Bibliotheken nutzen möchten, wählen Sie diese nach dem Gerätetausch über die Task Card "Anweisungen" aus.

Weitere Informationen zu den neuen Motion Control-Funktionen finden Sie im "Systemhandbuch SIMATIC S7-1200 Automatisierungssystem".

https://support.industry.siemens.com/cs/ww/de/view/65601780 (http://support.automation.siemens.com/WW/view/de/65601780)

Webserver

Folgende Einstellungen zum Betrieb über Webserver werden beim Gerätetausch von der CPU V3 auf die CPU V4 übertragen:

- Webserver auf dieser Baugruppe aktivieren
- Zugriff nur über HTTPS zulassen

Wenn Sie die CPU V4 über Webserver betreiben wollen, müssen Sie über die Benutzerverwaltung Benutzerkonten mit zugeordneten Rechten einrichten. Als Standardnutzer ohne zusätzliche Rechte stehen Ihnen nur die Standard-Webseiten zur Verfügung.

Hinweis

Weitere Unterstützung

Im Siemens Industry Online Support finden Sie aktuelle FAQs zum Umstieg auf S7-1200 V4:

http://support.automation.siemens.com/WW/view/de/82140966 (http://support.automation.siemens.com/WW/view/de/82140966)

Falls Sie weitere Unterstützung zum Umstieg auf S7-1200 V4 benötigen, wenden Sie sich bitte an den SIMATIC Customer Support.

Kommunikation über PUT/GET

Nach dem Gerätetausch ist die Kommunikation über PUT/GET aktiviert. Beachten Sie, dass die neuen, integrierten Verbindungstypen einen höheren Sicherheitsstandard bieten als die PUT/GET-Kommunikation. Wenn Sie die PUT/GET-Kommunikation nicht nutzen, sollten Sie sie wieder deaktivieren.

Siehe auch

Grundlagen zum Umstieg auf V4 (Seite 225)

Migration auf V4 durchführen (Seite 228)

Index

A

Aktion
    migrieren, 124, 127
Allen-Bradley DF1
    Migration der Datentypen, 81
Allen-Bradley DH485
    Migration der Datentypen, 82
Allen-Bradley Ethernet IP
    Migration der Datentypen, 82
Anwenderarchiv
    migrieren, 123
Archivdaten
    migrieren, 72
AWL
    Migration, 33

B

Basic Process Control
    migrieren, 134
Baugruppentausch, 146
Bediengerätewechsel
    durch die Migration, 55
Benutzerverwaltung
    migirieren, 132
    Migration, 72, 135
Bereichszeiger
    Migration, 62
Bildbaustein
    migrieren, 111

C

Controls
    migrieren, 120

D

Datentypen
    Migration, 80, 136
Druckauftrag
    migrieren, 129
Dynamik
    migrieren, 117

E

Eindeutigkeit
    von Objektnamen, 45

F

Farbeinstellungen
    migrieren, 121
Filterung der Projektsprachen, 102
Funktion
    migrieren, 124
FUP
    Migration, 33

G

GE Fanuc SNP
    Migration der Datentypen, 83
Gerätename
    Änderung durch die Migration, 101
Globale Bibliothek
    migrieren, 71
GRAPH
    Migration, 34
Größenangaben
    migrieren, 114
Grundlagen
    Migration, 45, 96

H

Header-Datei
    migrieren, 126
HMI-Variablen verknüpfen, 149
HMI-Verbindung, 147

I

Integriertes Projekt
    HMI-Variablen verknüpfen, 149
    Integrierte HMI-Verbindung erstellen, 147
    migrieren, 76, 78, 138, 140, 142
    nacharbeiten, 144
    Unspezifizierte CPU wandeln, 146
    Unspezifizierte Verbindung löschen, 151

IX, 214

K

Konsistenprüfung vor der Migration, 23
KOP
    Migration, 33

L

LG GLOFA GM
    Migration der Datentypen, 83

M

Meldearchiv
    migrieren, 72
Meldegruppe
    Migration, 64
Meldegruppen
    migrieren, 122
Meldeklassen
    migrieren, 122
    Namensänderung durch die Migration, 65
Meldesystem
    migrieren, 122
Meldungen
    migrieren, 122
Meldungen nach S7-1500 migrieren, 183
Migration, 29, 33, 34, 35, 167, 170, 179, 181, 191, 195, 213, 214, 216
    Ablauf der Migration, 10
    Aktionen, 124, 127
    Änderung des Gerätenamens, 101
    Anpassungen vorher, 51
    Anwenderarchiv, 123, 135
    Bediengerätewechsel, 55
    Benutzerverwaltung, 132, 135
    Bereichszeiger, 62
    Bildbaustein, 111
    Controls, 120
    Datentypen Allen-Bradley DF1, 81
    Datentypen Allen-Bradley DH485, 82
    Datentypen Allen-Bradley Ethernet IP, 82
    Datentypen GE Fanuc SNP, 83
    Datentypen LG GLOFA GM, 83
    Datentypen Mitsubishi FX, 84
    Datentypen Mitsubishi Protocol 4, 85
    Datentypen Modicon Modbus, 85
    Datentypen Modicon Modbus TCP/IP, 86
    Datentypen Omron Hostlink/Multilink, 86
    Datentypen OPC, 87
    Datentypen SIMATIC 500/505 DP, 88
    Datentypen SIMATIC 500/505 seriell, 88
    Datentypen SIMATIC HMI HTTP Protocol, 89
    Datentypen SIMATIC S5 AS511, 89
    Datentypen SIMATIC S5 DP, 90
    Datentypen SIMATIC S7 200, 91
    Datentypen SIMATIC S7 300/400, 91
    Datentypen Telemecanique Uni-Telway, 94
    der Benutzerverwaltung, 72
    der Rezepturdaten, 72
    Druckauftrag, 129
    Dynamik, 117
    Einführung, 44
    Einführung zur Migration, 9
    Einschluss der Hardware-Konfiguration, 10
    ET200S migrieren, 153
    Farbeinstellungen, 121
    Funktionen, 124
    Globale Bibliothek, 71
    Größenangaben, 114
    Grundlagen, 45, 96
    Header-Datei, 126
    Historie anzeigen, 18
    Integrierte Projekte, 140
    Integriertes Projekt, 142
    Integriertes Projekt migrieren, 76, 138, 142
    mehrsprachiger Protokolle, 128
    Meldegruppen, 122
    Meldeklassen, 122
    Objektgruppe, 115
    OLE-Objekt, 112
    PLC-Programme nach S7-1500 migrieren, 153, 156, 158, 159, 160, 161, 162, 164, 165, 166, 167, 183, 187, 188, 189, 197, 198, 200, 201, 203, 204, 205, 206, 208, 209, 211, 218, 220, 222
    Positionsangaben, 114
    Projektbibliothek, 71
    Projekte migrieren, 48, 78, 98
    Protokoll, 128
    Protokolldatei anzeigen, 19
    Runtime-Daten, 135
    Schrifteinstellungen, 58, 113
    Skript, 68
    Skript für Dynamisierung, 118
    Sprachen, 105
    Unterstützte Bediengeräte, 50
    unterstützte Objekttypen, 103
    Unterstützte Produkte, 9
    Variablen, 109
    VBSkript, 68
    Verbindung, 106

Verbindung nachprojektieren, 61, 108
von Archivdaten, 72
von Basic Process Control, 134
von externen Variablen, 80, 136
von Meldegruppen, 64
von Projekttexten, 69
von Runtime-Daten, 72
von Sprachen, 69
von Standardfunktionen, 127
von Texten, 69
WinCC Controls, 120
WinCC Push Button Control, 112
Zeilenprotokoll, 132
Zyklen, 133
Migration von STEP 7-Projekten
Anforderungen an das Projekt, 22
CBA bei IO-Controllern, 26
Einführung, 20
Gerätenamen von PROFINET IO-Devices, 25
GSD- und GSDML-Geräte, 27
I-Slaves und I-Devices, 26
Medienredundanz, 25
Meldungen, 42
Migration von Projekttexten, 43
Migrationsumfang, 20
Migrierbare Verbindungstypen, 27
Multiprojekt, 28
Namen von Modulen und Racks, 25
Shared Devices, 26
Systemfehlermeldungen, 42
Taktsynchronität bei PROFINET IO, 26
Taktsynchronität PROFIBUS, 27
Transferbereiche, 26
Update-Gruppen PROFINET IO, 26
Voraussetzungen, 21
Migration vorbereiten, 24
Migration: Migrationsfähigkeit der Hardware "prüfen, 12
Migrationsfähigkeit prüfen, 23
Migrationstool, 46, 97
Migrations-Tool
Hardware-Konfiguration einschließen, 14
Migrationsdatei erstellen, 15
Migrations-Tool verwenden, 13
Migrationstool starten, 14
Migriertes Projekt
starten, 101
übersetzen, 49
Mitsubishi FX
Migration der Datentypen, 84
Mitsubishi Protocol 4
Migration der Datentypen, 85
Modicon Modbus
Migration der Datentypen, 85
Modicon Modbus TCP/IP
Migration der Datentypen, 86

N

Namen von Meldeklassen
Änderung durch die Migration, 65

O

Objektgruppe
migrieren, 115
Objektnamen
Eindeutigkeit, 45
Umbenennung bei Migration, 45
OLE-Objekt
Migration, 112
Omron Hostlink/Multilink
Migration der Datentypen, 86
OPC
Migration der Datentypen, 87

P

PLC-Programme nach S7-1500 migrieren, 167, 170, 179, 181, 191, 195, 213, 214, 216
Positionsangaben
migrieren, 114
Projekt
migrieren, 48, 98
Projekt für Migration bereinigen, 24
Projektbibliothek
migrieren, 71
Projekte migrieren
Voraussetzungen, 16
Vorgehen, 16
Projektname
Änderung durch die Migration, 101
Projekttexte
migrieren, 69
Protokoll
migrieren, 128

R

Rezepturdaten
Migration, 72, 135

Runtime-Daten
Migration, 72, 135

S

Schrifteinstellungen
migrieren, 58, 113
SCL
Migration, 35
SIMATIC 500/505 DP
Migration der Datentypen, 88
SIMATIC 500/505 seriell
Migration der Datentypen, 88
SIMATIC HMI HTTP Protocol
Migration der Datentypen, 89
SIMATIC S5 AS511
Migration der Datentypen, 89
SIMATIC S5 DP
Migration der Datentypen, 90
SIMATIC S7 200
Migration der Datentypen, 91
SIMATIC S7 300/400
Migration der Datentypen, 91
Skript
migrieren, 68
Skript für Dynamisierung
migrieren, 118
Sprachen
migrieren, 69
Starten
migriertes Projekt, 101

T

Telemecanique Uni-Telway
Migration der Datentypen, 94
Texte
migrieren, 69

U

Übersetzen
migriertes Projekt, 49
Unspezifizierte CPU umwandeln, 146
User Archive
migrieren, 123

V

Variablen
Migration, 80, 136
migrieren, 109
Variablenarchiv
migrieren, 72
VBSkript
migrieren, 68
Verbindung
ändern nach Migration, 61, 108
migrieren, 106
Verbindung löschen, 151

W

WinCC flexible
Integriertes Projekt migrieren, 78
WinCC flexible-Projekt
migrieren, 48
WinCC Push Button Control
Migration, 112
WinCC V7-Projekt
migrieren, 98

Z

Zeilenprotokoll
migrieren, 132
Zyklen
migrieren, 133

SIEMENS

Erste Schritte 1

Bedienoberfläche und Bedienung 2

Hilfe zum Informationssystem 3

Suchen und ersetzen 4

Aktionen rückgängig machen und wiederholen 5

Einführung in das TIA Portal

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Erste Schritte....7

1.1 Getting Started Dokumentationen....7

2 Bedienoberfläche und Bedienung....9

2.1 TIA Portal starten, einstellen und beenden....9
2.1.1 TIA Portal starten und beenden....9
2.1.2 Übersicht der Programmeinstellungen....10
2.1.3 Übersicht der Einstellungen für Software Updates....13
2.1.4 Übersicht der Skript- und Texteditor-Einstellungen....13
2.1.5 Übersicht der Druckeinstellungen....14
2.1.6 Übersicht der Bibliothekseinstellungen....15
2.1.7 Übersicht der Einstellungen zum Durchsuchen eines Projekts....16
2.1.8 Übersicht der Einstellungen zu Querverweisen....16
2.1.9 Übersicht der Einstellungen zum Verwalten von Fensterlayouts....16
2.1.10 Übersicht der Einstellungen zum Basis-Integritätsschutz....17
2.1.11 Übersicht der Online- und Diagnoseeinstellungen....17
2.1.12 Einstellungen ändern....18
2.1.13 Einstellungen exportieren und importieren....18
2.1.13.1 Export und Import von Einstellungen....18
2.1.13.2 Einstellungen exportieren....19
2.1.13.3 Einstellungen automatisch importieren....19
2.1.13.4 Einstellungen manuell importieren....20

2.2 Aufbau der Bedienoberfläche....21
2.2.1 Ansichten....21
2.2.2 Portalansicht....21
2.2.3 Projektansicht....23
2.2.4 Bibliotheksansicht....26
2.2.5 Projektnavigation....26
2.2.5.1 Funktion und Aufbau der Projektnavigation....26
2.2.5.2 Spalten ein- und ausblenden....30
2.2.6 Arbeitsbereich....30
2.2.6.1 Grundlagen zum Arbeitsbereich....30
2.2.6.2 Arbeitsbereich maximieren und minimieren....32
2.2.6.3 Arbeitsbereich teilen....33
2.2.6.4 Elemente des Arbeitsbereichs abdocken....34
2.2.6.5 Gruppierte Elemente des Arbeitsbereichs verwenden....35
2.2.6.6 Elemente des Arbeitsbereichs minimieren und maximieren....37
2.2.6.7 Zwischen den Elementen des Arbeitsbereichs wechseln....37
2.2.7 Inspektorfenster....38
2.2.8 Task Cards....40
2.2.8.1 Grundlagen zu den Task Cards....40
2.2.8.2 Palettenmodus ändern....42
2.2.9 Referenzprojekte....42
2.2.10 Detailansicht....43
2.2.11 Übersichtsfenster....46

2.2.11.1 Übersichtsfenster....46
2.2.11.2 Objekte im Übersichtsfenster vergleichen....48
2.2.11.3 Detailansicht des Übersichtsfensters sortieren....49
2.2.12 Bedienoberflächenlayout....50
2.2.12.1 Speichern des Bedienoberflächenlayouts....50
2.2.12.2 Fensterlayout speichern....50
2.2.12.3 Fensterlayout laden....51
2.2.12.4 Fensterlayouts verwalten....52
2.2.12.5 Anordnung von Editoren und Tabellen sichern....52
2.2.12.6 Bedienoberfächenlayout zurücksetzen....53

2.3 Externe Applikationen ausführen....55
2.3.1 Grundlagen zur Ausführung von externen Applikationen....55
2.3.2 Externe Applikationen konfigurieren....55
2.3.3 Externe Applikation starten....57

2.4 Tastaturbedienung im TIA Portal....59
2.4.1 Bedienung des TIA Portals mit der Tastatur....59
2.4.2 Übersicht aller Tastenkombinationen anzeigen....59
2.4.3 Grundfunktionen des TIA Portals....59
2.4.4 Nutzen projektbezogener Funktionen....61
2.4.5 Anordnung von Fenstern....61
2.4.6 Navigation durch die Programmoberfläche....63
2.4.7 Anpassen von Editoren....64
2.4.8 Bearbeiten von Objekten....66
2.4.9 Textbearbeitung....67
2.4.10 Bearbeiten von Tabellen....68
2.4.11 Nutzen von Online-Funktionen....70
2.4.12 Bildschirmtastatur verwenden....70
2.4.13 Benutzerdefinierte Tastenkombinationen verwenden....71
2.4.13.1 Benutzerdefinierte Tastenkombinationen zuweisen....71
2.4.13.2 Benutzerdefinierte Tastenkombinationen löschen....71
2.4.13.3 Benutzerdefinierte Tastenkombinationen auf Standardeinstellung zurücksetzen....72
2.4.13.4 Benutzerdefinierte Tastenkombinationen importieren und exportieren....72
2.4.13.5 Liste der Tastenkombinationen auswählen....73

2.5 Betriebssystemabhängige Besonderheiten....75
2.5.1 Einfluss von Benutzerrechten....75
2.5.2 Benutzerrechte erweitern....76

3 Hilfe zum Informationssystem....79

3.1 Allgemeines zum Informationssystem....79

3.2 Informationssystem öffnen....86

3.3 Informationssystem verwenden....87

3.4 Informationssystem durchsuchen....89
3.4.1 Grundlagen zur Suche....89
3.4.2 Hilfethemen suchen....90
3.4.3 Beispiele zur Suche im Informationssystem 92

3.5 Favoriten verwenden....95

3.6 Codebeispiele in das Programm übernehmen....97

3.7 Weiterführende Informationen im Siemens Industry Online Support anzeigen....98

3.8 Hardware-Handbücher anzeigen....99
3.9 Hilfethemen drucken....101
3.10 Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren....102
3.11 Individuelle Dokumentation zusammenstellen....103
3.12 Sicherheitshinweise....104
4 Suchen und ersetzen....107
4.1 Grundlagen zur Suche....107
4.2 Gesamtes Projekt durchsuchen....108
4.2.1 Grundlagen zur Suche im Projekt....108
4.2.2 Übersicht über den Sucheditor....110
4.2.3 Im gesamten Projekt suchen....111
4.2.4 Suchindex manuell neu erstellen....113
4.2.5 Suche im Projekt aktivieren und deaktivieren....113
4.2.6 Beispiele zur Suche im Projekt....114
4.3 Innerhalb eines Editors suchen und ersetzen....118
4.3.1 Grundlagen zur Suche in geöffneten Editoren....118
4.3.2 Übersicht über die Palette "Suchen und ersetzen"....119
4.3.3 Im Editor suchen und ersetzen....120
5 Aktionen rückgängig machen und wiederholen....123
5.1 Grundlagen zum Rückgängig machen und Wiederholen von Aktionen....123
5.2 Aktion rückgängig machen....125
5.3 Aktion wiederholen....126
Index....127

Erste Schritte 1

1.1 Getting Started Dokumentationen

Einstieg in das TIA-Portal über das TIA Portal Tutorial Center

Zum schnellen Einstieg in das TIA Portal stehen Ihnen im TIA Portal Tutorial Center Getting Started-Dokumente in Form von Videos zur Verfügung. Sie können sich einzelne Videos gezielt online anschauen oder sich das gesamte Media System herunterladen.

Die Videos haben die folgende Inhalte:

- Sie geben Ihnen einen Überblick über die Funktionen und Werkzeuge des TIA Portals.
- Sie geben Ihnen einen Überblick über das reibungslose Zusammenspiel der Projektier- und Programmiertools .
- Sie zeigen, wie Sie mit dem TIA Portal effektiv arbeiten können.
- Sie zeigen, wie Sie Ihre verschiedenen Automatisierungsaufgaben im TIA Portal realisieren.

Die Videos des TIA Portal Tutorial Center richten sich sowohl an Planer und Projekteure, als auch an Inbetriebsetzungs- und Servicepersonal. Die enthaltenen Videos vermitteln hauptsächlich technische Inhalte und bieten eine Hilfestellung in allen Phasen des Anlagenlebenszyklus.

Über den folgenden Link können Sie das TIA Portal Tutorial Center aufrufen:

http://www.siemens.de/tia-portal-tutorial-center (http://www.siemens.de/tia-portal-tutorial-center)

Siehe auch

Getting Started Basic 10.5 (http://support.automation.siemens.com/WW/view/de/40263542)

Service und Support (https://support.industry.siemens.com/cs/de/de/)

Getting Started Professional 11.0 (https://support.industry.siemens.com/cs/ww/de/view/54430386)

Getting Started Professional 12.0 (http://www.automation.siemens.com/salesmaterial-as/interactive-manuals/getting-started_simatic-s7-1500/_content/DE/content_de.html)

Bedienoberfläche und Bedienung 2

2.1 TIA Portal starten, einstellen und beenden

2.1.1 TIA Portal starten und beenden

TIA Portal starten

Um das TIA Portal zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie unter Windows den Befehl "Start > Alle Programme > Siemens Automation > TIA Portal V14".
   Das TIA Portal wird mit den zuletzt verwendeten Einstellungen geöffnet.

TIA Portal beenden

Um das TIA Portal zu beenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Beenden".
   Wenn das Projekt nicht gespeicherte Änderungen enthält, werden Sie gefragt, ob die Änderungen gespeichert werden sollen.
   - Wählen Sie "Ja", um die Änderungen im aktuellen Projekt zu speichern und das TIA Portal zu schließen.
   - Wählen Sie "Nein", um das TIA Portal zu schließen, ohne die letzten Änderungen im Projekt zu speichern.
   - Wählen Sie "Abbrechen", um das Beenden abzubrechen. In diesem Fall bleibt das TIA Portal geöffnet.

2.1.2 Übersicht der Programmeinstellungen

Überblick

Die folgende Tabelle zeigt die Anwendungseinstellungen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Allgemeine Einstellungen	Benutzername	Der Benutzername des Anwenders. Der Benutzername wird beim Anlegen eines neuen Projekts in den Projekteigenschaften hinterlegt.
	Oberflächensprache	Sprache für die Programmoberfläche
	Mnemonik	Legt die Mnemonik für die Programmierung fest: "Deutsch" verwendet die deutsche Mnemonik, z. B. "E1.0" "International" verwendet die internationale Mnemonik, z. B. "I1.0" Die Unterschiede in der Mnemonik der einzelnen Befehle entnehmen Sie den Sprachbeschreibungen der jeweiligen Programmiersprache.
	Liste zuletzt verwendeter Projekte anzeigen	Anzahl der Einträge in der Liste der zuletzt verwendeten Projekte im Menü "Projekt"
	Zuletzt geöffnetes Projekt beim Start laden	Das zuletzt geöffnete Projekt wird nach dem Start des TIA Portals automatisch geöffnet.
	Abgeschnittene Texte komplett anzeigen	Texte, die aufgrund ihrer Länge abgeschnitten werden, werden in einem Tooltip angezeigt.
	Tooltips anzeigen (kontextsensitive Hilfe ist verfügbar)	Tooltips werden angezeigt und Sie erhalten kontextbezogene Hilfe. Ist die Funktion deaktiviert, können Sie den Tooltip über <F1> öffnen. Die Einstellung wirkt sich ebenfalls auf Rollouts mit kontextbezogenen Syntaxfehlern und Eingabehilfen in AWL aus.
	Kaskade in den Tooltips automatisch öffnen	Die Tooltips klappen nach kurzer Zeit automatisch auf und eine Kaskade mit weiterer Hilfe wird angezeigt. Ist die Option deaktiviert, müssen die Tooltips manuell aufgeklappt werden.
	Banner minimiert anzeigen	Banner mit zusätzlichen Informationen im Editor werden standardmäßig minimiert angezeigt.
Auf Standard zurücksetzen	Alle Anwendungseinstellungen	Alle Änderungen, die Sie im TIA Portal nach der Installation durchgeführt haben, werden rückgängig gemacht.
	Layout der Editoren	Setzt das komplette Layout der Anwendung auf den Auslieferungszustand zurück.
	Alle Meldefenster anzeigen	Alle Meldefenster, deren Erscheinen manuell unterdrückt wurde, werden wieder angezeigt.
Startansicht	Zuletzt verwendete Ansicht	Startet das Programm immer in der zuletzt verwendeten Ansicht. Dies kann entweder die Portalansicht oder die Projektansicht sein.
	Portalansicht	Startet das TIA Portal immer in der Portalansicht, unabhängig davon, in welcher Ansicht zuletzt gearbeitet wurde.
	Projektansicht	Startet das TIA Portal immer in der Projektansicht, unabhängig davon, in welcher Ansicht zuletzt gearbeitet wurde.
Einstellungen importieren/ exportieren	Einstellungen in Datei exportieren	Öffnet einen Dialog, über den Sie ausgewählte Programmeinstellungen in eine Datei exportieren können.
	Einstellungen aus Datei importieren	Ermöglicht den Import von zuvor exportierten Programmeinstellungen aus einer Datei.

Gruppe	Einstellung	Beschreibung
Zentrale Einstellungen	Speicherort für zentrale Einstellungen	Hier legen Sie den Speicherort für die Datei fest, in der die Einstellungen zentral gespeichert sind.
Ansicht für Objekte in der Übersicht	Details	Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Detailansicht gestartet, z. B. im Übersichtsfenster.
	Liste	Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Listenansicht gestartet, z. B. im Übersichtsfenster.
	Symbole	Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Symbolansicht gestartet, z. B. im Übersichtsfenster.
Speichereinstellungen	Zuletzt verwendeten Speicherort verwenden	Beim ersten Speichern eines Projekts ist der zuletzt verwendete Pfad zum Speichern voreingestellt.
	Standardeinstellung für Speicherort festlegen	Ermöglicht das Festlegen von Speicherpfaden für: • Projekte • Bibliotheken • Konfigurationsdatei zu Unternehmensbibliotheken
	Speicherort für neue Projekte	Ermöglicht das Festlegen des Speicherpfads für Projekten. Neue Projekte werden automatisch an diesem Pfad angelegt.
	Speicherort für Bibliotheken	Ermöglicht das Festlegen des Speicherpfads für globale Bibliotheken. Beim Anlegen einer neuen Bibliothek wird diese automatisch an diesem Pfad gespeichert.
	Speicherort für die Konfigurationsdatei zu Unternehmensbibliotheken	Zeigt den Pfad für die Konfigurationsdatei zu Unternehmensbibliotheken an. Der Pfad ist nicht änderbar.
Speichereinstellungen für das Archivieren	Zuletzt verwendete Speichereinstellungen für das Archivieren	Beim Speichern ist der zuletzt verwendete Pfad voreingestellt.
	Standardeinstellungen für die Speicherorte zum Archivieren festlegen	Ermöglicht das Festlegen von Speicherpfaden für: • Projektarchive • dearchivierte Projekte • Bibliotheksarchive • dearchivierte Bibliotheken
	Speicherort für Projektarchive	Ermöglicht das Festlegen des Speicherpfads für Projektarchive. Projektarchive werden automatisch an diesem Pfad angelegt.
	Speicherort für dearchivierte Projekte	Ermöglicht das Festlegen des Speicherpfads für dearchivierte Projekte. Dearchivierte Projekte werden automatisch an diesem Pfad angelegt.
	Speicherort für Bibliotheksarchive	Ermöglicht das Festlegen des Speicherpfads für Bibliotheksarchive. Bibliotheksarchive werden automatisch an diesem Pfad angelegt.
	Speicherort für dearchivierte Bibliotheken	Ermöglicht das Festlegen des Speicherpfads für dearchivierte Bibliotheken. Dearchivierte Bibliotheken werden automatisch an diesem Pfad angelegt.

Gruppe	Einstellung	Beschreibung
Datenaustausch	Speicherort für Datenimport	An diesem Speicherpfad wird standardmäßig nach Importdateien gesucht.
	Speicherort für Datenexport	Dieser Speicherpfad ist für den Datenexport voreingestellt.
	Speicherort für Support Packages	Nach dem Laden von Support Packages werden diese im angegebenen Speicherpfad abgelegt und können dann von dort aus installiert werden.
	Speicherort für Protokolldateien	Protokolldateien werden am hier angegebenen Ort abgelegt.
Anwenderdokumentation	Anwenderdefinierte Dokumentation in der gewählten Editiersprache anzeigen	Die Anwenderdokumentation wird in der Projekt-Editiersprache aufgerufen (Shift + F1). Wenn die Anwenderdokumentation in der ausgewählten Editiersprache nicht vorhanden ist, dann wird per Voreinstellung die englische Fassung aufgerufen.
	Aufrufprotokoll zu anwenderdefinierter Dokumentation anzeigen	Zeigt Aufrufprotokolle zur Anwenderdokumentation an. Im Inspektorfenster werden Meldungen zum Aufruf von selbst erstellten Hilfedateien ausgegeben. So lassen sich Schwierigkeiten beim Aufruf der selbst erstellten Hilfe leichter identifizieren und beheben. Diese Funktion ist insbesonders während des Erstellungsprozesses eigener Hilfedateien nützlich.
	Suche nach anwenderdefinierter Dokumentation in einem zentralen Verzeichnis	Ermöglicht das Speichern von anwenderdefinierter Dokumentation in einem Verzeichnis außerhalb des aktuellen Projektverzeichnisses, z. B. um projektübergreifende Dokumentation zur Verfügung zu stellen.
	Zentrales Verzeichnis für anwenderdefinierte Dokumentation	Ermöglicht das Festlegen des Speicherpfads, an dem eigene Hilfedateien für die anwenderdefinierte Dokumentation abgelegt werden können.

Siehe auch

TIA Portal starten und beenden (Seite 9)

Bedienoberfächenlayout zurücksetzen (Seite 53)

Einstellungen ändern (Seite 18)

Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren (Seite 102)

2.1.3 Übersicht der Einstellungen für Software Updates

Überblick

Die folgende Tabelle zeigt die Einstellungen für Software Updates, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
TIA Portal Update	Täglich nach Updates suchen	Prüft nach jedem Neustart des Rechners und anschließend alle 24 Stunden, ob Updates verfügbar sind. Falls Updates gefunden werden, erhalten Sie eine Benachrichtigung in Infobereich der Taskleiste von Windows.
	Jetzt nach Updates suchen	Startet die Suche nach Updates und zeigt das Ergebnis im TIA Updater an.

2.1.4 Übersicht der Skript- und Texteditor-Einstellungen

Überblick

Die folgende Tabelle zeigt die Einstellungen für Skript- und Texteditoren, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Schriftart	Schriftart und Schriftgröße	Stellt die Schriftart und Schriftgröße für den Text in Texteditoren ein.
Schriftfarben	Farbeinstellungen	Sie können die Farben für einzelne Textelemente in den Texteditoren aus den jeweiligen Klapplisten wählen. Es stehen Einstellmöglichkeiten für die folgenden Textelemente zur Verfügung: • Text • Schlüsselwörter • Kommentare • Operatoren • Skripte • Standardfunktionen • Anweisungen/Systemfunktionen • String-Konstanten • Numerische Konstanten • Symbolische Konstanten • Variablen • Objektmodelle • Formalparameter
	Auf Standard zurücksetzen	Setzt alle Schriftfarben in Editoren auf Werkseinstellungen zurück.

Gruppe	Einstellung	Beschreibung
Tabulatoren	Tabulatorweite	Stellt die Weite von Tabulatoren ein.
	Tabulatoren verwenden	Aktiviert die Verwendung von Tabulatoren.
	Leerzeichen verwenden	Verwendet anstatt von Tabulatoren Leerzeichen.
Einrückung	Keine	Texteingaben werden nicht automatisch eingerückt.
	Absatz	Die Zeile bzw. der selektierte Absatz erhält einen automatischen Einzug.
	Smart	Die Zeile bzw. der selektierte Absatz erhält einen automatischen Einzug. Zusätzlich werden alle unnötigen Leerzeichen entfernt.
Ansicht	Leerzeichen anzeigen	Zeigt Steuerzeichen innerhalb eines Texts an.
AWL (Anweisungsliste)	Schriftart und Schriftgröße	Stellt die Schriftart und die Schriftgröße für AWL-Programmcode ein.
SCL (Structured Control Language)	Zeilennummern anzeigen	Zeigt in SCL-Programmen Zeilennummern an.

Siehe auch

Einstellungen ändern (Seite 18)

2.1.5 Übersicht der Druckeinstellungen

Überblick

Die folgende Tabelle zeigt die Einstellungen für den Druck, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Allgemein	Tabellendaten immer in Wertepaaren drucken	Tabellen werden nicht in tabellarischer Form, sondern als Liste gedruckt. Zu jeder Spalte werden die entsprechenden Werte aufgelistet. Aktivieren Sie diese Option beispielsweise, wenn Sie eine Tabelle drucken möchten, die zu groß für die Druckfläche ist.
	Zeitüberschreitung für Druckauftrag (Minuten)	Wenn Elemente nicht vollständig gedruckt werden können, wird nach der eingestellten Zeitüberschreitung in Minuten eine Meldung angezeigt.
Hardware-Konfiguration	Aktive grafische Sicht	Die Grafiken der Netz- und Gerätesicht werden mit ausgedruckt.
	Aktive Tabelle	Eine zu einem Editor gehörende Tabelle wird beim Drucken des Editors mit ausgegeben.
PLC-Programmierung	Zoomfaktor	Legt fest, in welcher Größe Bausteine ausgedruckt werden sollen.
	mit Schnittstelle	Die Schnittstellen von Bausteinen werden mit ausgedruckt.
	mit Kommentaren	Kommentare zu Bausteinen werden mit ausgedruckt.
	mit Zeilennummern	Bei textuellen Programmiersprachen werden die Zeilennummern des Programmcodes mit ausgedruckt.

Gruppe	Einstellung	Beschreibung
Motion & Technology	Dialogdarstellung/Grafik	Der Inhalt des Editors wird als Grafik ausgedruckt, wenn der Editor dies unterstützt.
	Tabelle	Die Parameter der Technologieobjekte werden in Tabellenform ausgedruckt.
HMI-Bilder	Tab-Reihenfolge anzeigen	Im Ausdruck wird die Reihenfolge angegeben, in der die Runtime Objekte mithilfe des Tabulators angewählt werden können.

Siehe auch

Einstellungen ändern (Seite 18)

2.1.6 Übersicht der Bibliothekseinstellungen

Überblick

Die folgende Tabelle zeigt die Bibliothekseinstellungen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Typ-Version	Bearbeitung der Typ-Versionsnummer deaktivieren	Wenn die Einstellung aktiviert ist, kann die Typ-Versionsnummer bei den Aktionen "Version freigeben" und "Version zuweisen" vom Anwender nicht mehr verändert werden.
Typverbindung	Kontextmenüeintrag "Verbindung zum Typ aufheben" deaktivieren	Wenn die Einstellung aktiviert ist, dann kann die Instanz eines Typs in der Projektnavigation nicht mehr mithilfe des Kontextmenüs vom Typ in einer Bibliothek getrennt werden. Der Kontextmenüeintrag "Verbindung zum Typ aufheben" wird deaktiviert.
Projekt aktualisieren	Beim Aktualisieren des Projekts aus der Bibliothek die nicht-verwendeten Typ-Instanzen nicht löschen	Wenn die Einstellung aktiviert ist, werden beim Aktualisieren des Projekts die nicht-verwendeten Typ-Instanzen nicht gelöscht. Wenn ein nicht verwendeter Typ keine neuere Typ-Instanz hat, wird diese Typ-Instanz nicht gelöscht. Wenn ein nicht verwendeter Typ eine neuere Typ-Instanz hat, wird diese Typ-Instanz nur dann gelöscht, wenn sie nach dem Hochrüsten des Projekts nicht verwendet wird.

2.1.7 Übersicht der Einstellungen zum Durchsuchen eines Projekts

Überblick

Die folgende Tabelle zeigt die Einstellungen zum Durchsuchen eines Projekts, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Projekt durchsuchen	"Projekt durchsuchen" aktivieren	Aktiviert oder deaktiviert die Suche im Projekt.
	Suchindex neu erstellen	Erzeugt den Suchindex für die Suche im Projekt neu.

2.1.8 Übersicht der Einstellungen zu Querverweisen

Überblick

Die folgende Tabelle zeigt die Einstellungen zu Querverweisen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Querverweisinformationen neu erstellen	Querverweisinformationen neu erstellen	Erstellt neue Querverweisinformationen zum ausgewählten Projekt.

2.1.9 Übersicht der Einstellungen zum Verwalten von Fensterlayouts

Überblick

Die folgende Tabelle zeigt die Einstellungen zum Verwalten von Fensterlayouts, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Alle Fensterlayouts verwalten	Aktiv	Ermöglicht die Auswahl eines Fensterlayouts, sofern zusätzlich zum Standard-Fensterlayout weitere Fensterlayouts angelegt wurden.

2.1.10 Übersicht der Einstellungen zum Basis-Integritätsschutz

Überblick

Die folgende Tabelle zeigt die Einstellungen für den Basis-Integritätsschutz, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Basis-Integritäts-schutz	Basis-Integritätsprüfung beim Öffnen eines Projekts aktivieren	Aktiviert standardmäßig die Option zur Durchführung der Basis Integritätsprüfung im Dialog "Projekt öffnen". Außerdem erfolgt die Basis Integritätsprüfung beim Öffnen von Projekten aus der Liste der zuletzt geöffneten Projekte, der Referenzprojekte als auch in Multiuser Engineering automatisch. Die Prüfung kann je nach Umfang und Anwendungsfall zeitaufwändig sein, insbesondere wenn Sie Multiuser Engineering nutzen.
	Basis-Integritätsprüfung beim Öffnen einer Bibliothek aktivieren	Aktiviert die automatische Basis Integritätsprüfung beim Öffnen von Bibliotheken. Die Prüfung kann je nach Umfang und Anwendungsfall zeitaufwändig sein.

Siehe auch

Einstellungen ändern (Seite 18)

2.1.11 Übersicht der Online- und Diagnoseeinstellungen

Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie für die Online- und Diagnosefunktionen vornehmen können:

Gruppe	Einstellung	Beschreibung
Voreingestellter Verbindungsweg für den Online-Zugriff	Typ der PG/PC-Schnittstelle	Legt den Typ der PG/PC-Schnittstelle fest, der als Voreinstellung in den Dialogen für den Online-Zugriff verwendet wird, z. B. im Dialog "Online verbinden".
	PG/PC-Schnittstelle	Legt eine bestimmte PG/PC-Schnittstelle fest, die als Voreinstellung in den Dialogen für den Online-Zugriff verwendet wird, z. B. im Dialog "Online verbinden".
	Voreingestellten Verbindungsweg bei der Online-Verbindung verwenden	Aktiviert oder deaktiviert die Voreinstellungen für die PG/PC-Schnittstelle. Wenn das Optionskästchen aktiviert ist, wird der in den Einstellungen festgelegte Verbindungsweg als Voreinstellung in den Dialogen für den Online-Zugriff verwendet.
Meldungsanzeige	Mehrzeilig	Zeigt die Meldungen im Inspektorfenster mehrzeilig an.
	Automatisch aktuelle Meldung anzeigen	Beim Auftreten einer neuen Meldungen wird automatisch zu dieser Meldung gescrollt.
	Archivgröße	Maximale Anzahl der Meldungen, die im Inspektorfenster angezeigt werden. Wird die eingestellte Zahl von Meldungen überschritten, werden ältere Meldungen automatisch gelöscht.

2.1.12 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein", um die in den vorangegangenen Kapiteln beschriebenen Einstellungen zu ändern. Oder klicken Sie auf einen der weiteren Einträge in der Bereichsnavigation, um Einstellungen für Ihre installierten Produkte zu ändern.
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird sofort übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht der Programmeinstellungen (Seite 10)

Übersicht der Skript- und Texteditor-Einstellungen (Seite 13)

Übersicht der Druckeinstellungen (Seite 14)

2.1.13 Einstellungen exportieren und importieren

2.1.13.1 Export und Import von Einstellungen

Sie können Einstellungen des TIA Portals in eine Datei exportieren. Wenn Sie die Einstellungen importieren, werden diese gesammelt übernommen. Diese Funktion ermöglicht es, Einstellungen an andere Anwender weiter zu geben. Ebenso können Systemeinstellungen zentral vorgegeben werden, um für mehrere Engineering-PCs eine einheitliche Umgebung zu schaffen. Das TIA Portal sucht beim Start nach einer Einstellungsdatei. Wenn eine Einstellungsdatei gefunden wird, dann werden die darin gespeicherten Einstellungen nach einem Neustart automatisch übernommen. Wenn eine neuere Version dieser Datei vorliegt, werden die lokalen Einstellungen aktualisiert.

Sie können alle Einstellungen exportieren, z. B.:

- Sprache der Bedienoberfläche
- Pfad der Unternehmensbibliotheken
- Pfad der anwenderdefinierten Dokumentation
- Fensterlayouts
- Profile des Hardware-Katalogs

Die Einstellungsdateien sind versionsgebunden und können nur innerhalb einer Produktversion automatisch importiert werden, nicht jedoch in anderen Produktversionen.

2.1.13.2 Einstellungen exportieren

Sie können Einstellungen des TIA Portals in eine Datei exportieren, z. B. um diese zu sichern oder an andere Anwender weiter zu geben.

Vorgehen

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen exportieren...".
   Der Dialog "Exporteinstellungen" wird geöffnet.
4. Wählen Sie die gewünschten Einstellungen für den Export aus.
   - Klicken Sie auf "Alles auswählen", um alle Einstellungen für den Export auszuwählen.
   - Klicken Sie auf "Alles abwählen", um anschließend einzelne Einstellungen für den Export auszuwählen. Dabei werden alle untergeordneten Einstellungen mit ausgewählt.
5. Wenn Sie die Auswahl der Einstellungen von früheren Exportvorgängen übernehmen wollen, z. B. bei unterschiedlichen Verwendungen der Einstellungsdateien, aktivieren Sie das Optionskästchen bei "Auswahl aus einer vorherigen Exportdatei verwenden" und tragen die Datei in das Feld ein.
6. Tragen Sie einen Dateinamen für die Einstellungsdatei in das Feld "Dateiname" ein.
7. Klicken Sie auf "Export".
   Die Einstellungen werden in eine Datei mit der Endung ".tps15_1" exportiert.

Hinweis

Leere Eingabefelder in der Exportdatei

In den Einstellungen gibt es Eingabefelder, z. B. "Zentrale Einstellungen". Wenn Sie eines dieser Eingabegelder nicht ausfüllen und die Einstellung trotzdem für den Export auswählen, dann wird eine leere Einstellung exportiert. Wenn Sie anschließend die Einstellungsdatei importieren, werden die entsprechenden lokalen Einstellungen durch die leeren Inhalte der importierten Einstellungsdatei überschrieben.

Wenn Sie nicht wünschen, dass diese Felder durch einen Import gelöscht werden, deaktivieren Sie bereits beim Export der Einstellungsdatei die nicht befüllten Eingabefelder.

2.1.13.3 Einstellungen automatisch importieren

Sie können beim Start des TIA Portals automatisch die Einstellungen aus einer Einstellungsdatei importieren, z. B. um eine zentral vorgegebene Umgebung zu übernehmen.

Vorgehen

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Zentrale Einstellungen" auf die Schaltfläche "Durchsuchen".
4. Wählen Sie die Einstellungsdatei mit der Endung ".tps15_1" aus.
5. Klicken Sie auf "Öffnen".
   Nach einem Neustart des TIA Portals werden die Einstellungen übernommen.

Hinweis

Priorität von zentralen Einstellungen

Zentrale Einstellungen haben Priorität vor manuell importierten Einstellungen und überschreiben diese.

Hinweis

Anzeige des Speicherpfads für zentrale Einstellungen

Wenn Sie in den Einstellungen im Bereich "Zentrale Einstellungen" einen Speicherpfad zu einer Einstellungsdatei eingetragen haben, wird dieser Pfad nach einem Neustart des TIA Portals nicht mehr angezeigt. Die Einstellungen werden nach einem Neustart dennoch geladen.

Um die zentralen Einstellungen für das System zu konfigurieren, ändern Sie daher zuerst den Pfad zum Speicherort der Einstellungsdatei und exportieren Sie anschließend die Einstellungen in die Einstellungsdatei.

2.1.13.4 Einstellungen manuell importieren

Sie können Einstellungen des TIA Portals manuell aus einer Einstellungsdatei importieren, z. B. um Einstellungen von anderen Anwendern zu übernehmen.

Vorgehen

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen importieren...".
4. Wählen Sie die Einstellungsdatei mit der Endung ".tps15_1" aus.
5. Klicken Sie auf "Öffnen".
   Nach einem Neustart des TIA Portals werden die Einstellungen übernommen.

2.2 Aufbau der Bedienoberfläche

2.2.1 Ansichten

Ansichten

Es stehen Ihnen drei verschiedene Ansichten für Ihr Automatisierungsprojekt zur Verfügung:

- Die Portalansicht ist eine aufgabenorientierte Sicht auf die Projektaufgaben.
- Die Projektansicht ist eine Sicht auf die Bestandteile des Projekts und auf die zugehörigen Arbeitsbereiche und Editoren.
- Die Bibliotheksansicht (Seite 26) zeigt die Elemente der Projektbibliothek und der geöffneten globalen Bibliotheken.

Sie können über einen Link zwischen den beiden Ansichten wechseln.

2.2.2 Portalansicht

Funktion der Portalansicht

Die Portalansicht bietet eine aufgabenorientierte Sicht der Werkzeuge. Hier können Sie schnell entscheiden, was Sie tun möchten und das Werkzeug für die jeweilige Aufgabe aufrufen. Falls erforderlich, wird für die ausgewählte Aufgabe automatisch zur Projektansicht (Seite 23) gewechselt.

Aufbau der Portalansicht

Das folgende Bild zeigt ein Beispiel für die Komponenten der Portalansicht:

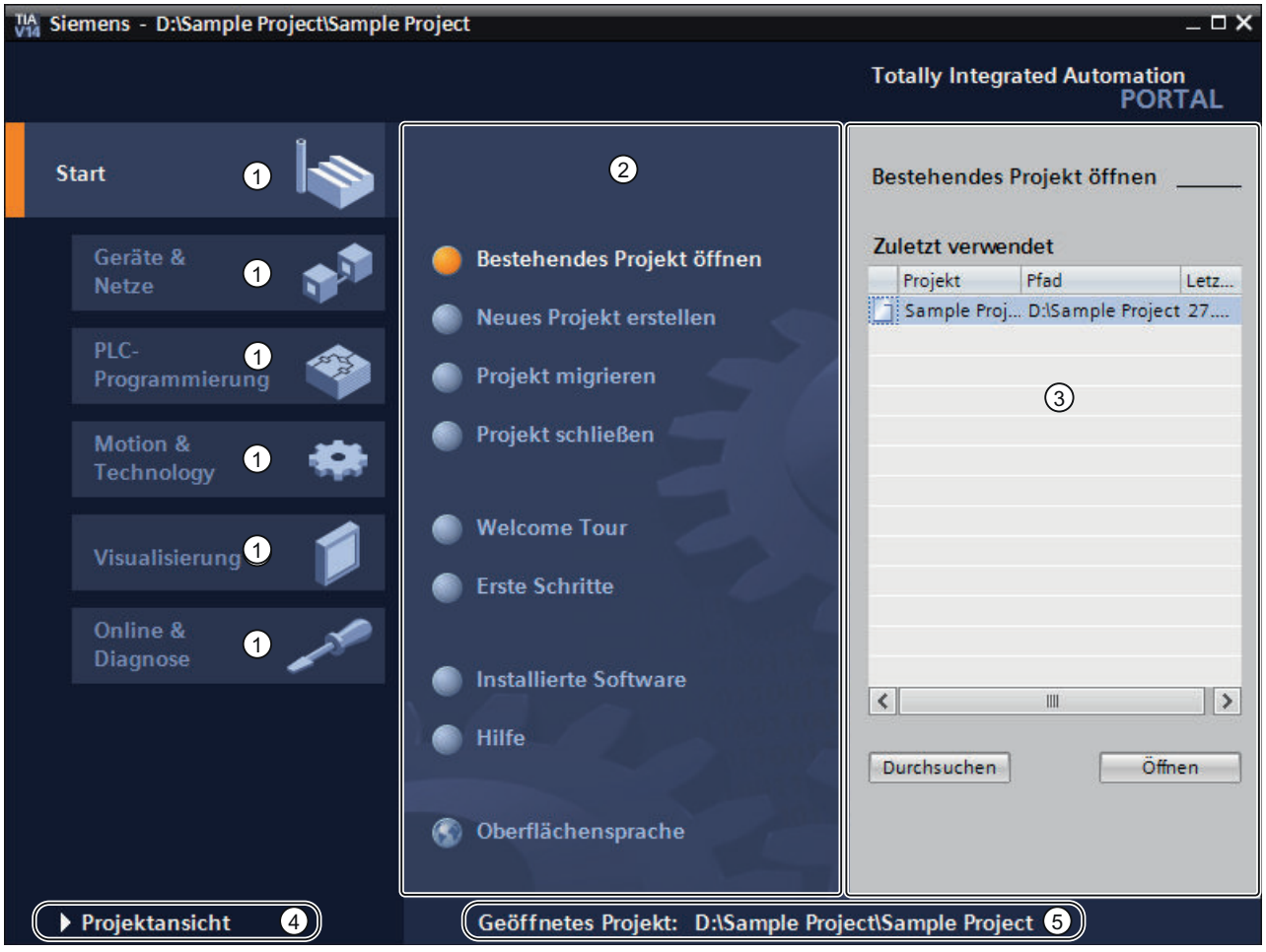

① Portale für die verschiedenen Aufgaben
② Aktionen zum selektierten Portal
③ Auswahlfenster zur selektierten Aktion
④ Zur Projektansicht wechseln
⑤ Anzeige des aktuell geöffneten Projekts

Portale

Die Portale stellen die grundlegenden Funktionen für die einzelnen Aufgabengebiete zur Verfügung. Welche Portale Ihnen in der Portalansicht angeboten werden, hängt von den installierten Produkten ab.

Aktionen zum selektierten Portal

Abhängig vom ausgewählten Portal werden Ihnen hier die Aktionen angeboten, die Sie in diesem Portal ausführen können. Der Aufruf der Hilfe steht Ihnen in jedem Portal kontextsensitiv zur Verfügung.

Auswahlfenster zur selektierten Aktion

Das Auswahlfenster steht Ihnen in allen Portalen zur Verfügung. Der Inhalt des Fensters passt sich der aktuellen Auswahl an.

Zur Projektansicht wechseln

Mithilfe des Links "Projektansicht" können Sie zur Projektansicht wechseln.

Anzeige des aktuell geöffneten Projekts

An dieser Stelle erhalten Sie die Information, welches Projekt aktuell geöffnet ist.

Siehe auch

Funktion und Aufbau der Projektnavigation (Seite 26)

Grundlagen zum Arbeitsbereich (Seite 30)

Inspektorfenster (Seite 38)

Grundlagen zu den Task Cards (Seite 40)

Detailansicht (Seite 43)

2.2.3 Projektansicht

Funktion der Projektansicht

Die Projektansicht ist eine strukturierte Sicht auf alle Bestandteile des Projekts.

Aufbau der Projektansicht

Das folgende Bild zeigt beispielhaft die Komponenten der Projektansicht:

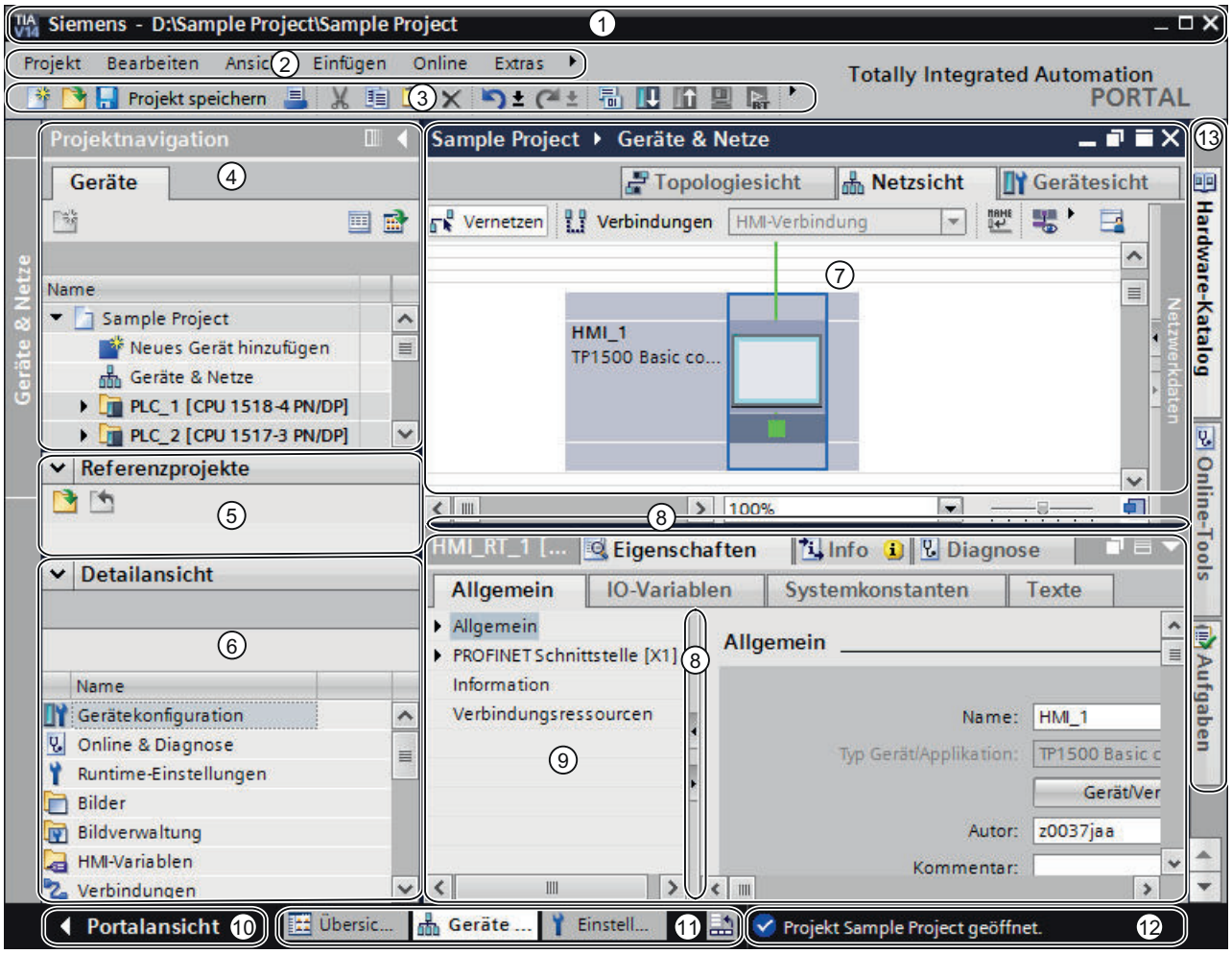

① Titelleiste
② Menüleiste
③ Funktionsleiste
④ Projektnavigation (Seite 26)
⑤ Referenzprojekte (Seite 42)
⑥ Detailansicht (Seite 43)
⑦ Arbeitsbereich (Seite 40)
⑧ Fensterteiler
⑨ Inspektorfenster (Seite 38)
⑩ Zur Portalansicht (Seite 21) wechseln
⑪ Editorleiste
⑫ Statusleiste mit Fortschrittsanzeige
⑬ Task Cards (Seite 40)

Titelleiste

In der Titelleiste wird der Name des Projekts angezeigt.

Menüleiste

In der Menüleiste finden Sie alle Befehle, die Sie für Ihre Arbeit benötigen.

Funktionsleiste

Die Funktionsleiste stellt Ihnen oft benötigte Befehle über Schaltflächen zur Verfügung. Dadurch können Sie auf diese Befehle schneller zugreifen.

Fensterteiler

Fensterteiler trennen einzelne Elemente der Programmoberfläche. Mit den Pfeilen auf den Fensterteilern lassen sich die angrenzenden Teile der Programmoberfläche ein- und ausblenden.

Zur Portalansicht wechseln

Mithilfe des Links "Portalansicht" können Sie zur Portalansicht wechseln.

Editorleiste

In der Editorleiste werden die geöffneten Editoren angezeigt. Dies ermöglicht es Ihnen, schnell zwischen den geöffneten Elementen zu wechseln. Falls Sie viele Editoren geöffnet haben, können Sie gleichartige Editoren als Gruppe anzeigen lassen.

Statusleiste mit Fortschrittsanzeige

In der Statusleiste finden Sie die Fortschrittsanzeige für aktuell laufende Hintergrundprozesse. Hierzu gehört auch eine Fortschrittsleiste, die den Fortschritt grafisch anzeigt. Verweilen Sie etwas länger mit dem Mauszeiger über der Fortschrittsleiste, so wird ein Tooltip eingeblendet, der Ihnen weitere Informationen über die laufenden Hintergrundprozesse gibt. Sie können die Hintergrundprozesse über die Schaltfläche neben der Fortschrittsleiste abbrechen.

Falls zurzeit keine Hintergrundprozesse ausgeführt werden, wird in der Statusleiste die zuletzt generierte Meldung angezeigt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

2.2.4 Bibliotheksansicht

Funktion der Bibliotheksansicht

Die Bibliotheksansicht bietet Ihnen eine Übersicht über die Elemente der Projektbibliothek und der geöffneten globalen Bibliotheken. Sie können über die Task Card "Bibliotheken" zur Bibliotheksansicht wechseln.

Siehe auch: Übersicht über die Bibliotheksansicht

2.2.5 Projektnavigation

2.2.5.1 Funktion und Aufbau der Projektnavigation

Funktion der Projektnavigation

Über die Projektnavigation haben Sie Zugang zu allen Komponenten und Projektdaten. Sie können in der Projektnavigation z. B. folgende Aktionen durchführen:

- Neue Komponenten hinzufügen
- Bestehende Komponenten bearbeiten
- Die Eigenschaften bestehender Komponenten abfragen und verändern

Sie können die Objekte der Projektnavigation entweder mit der Maus selektieren oder über die Tastatur, in dem Sie den Anfangsbuchstaben des gewünschten Objekts eingeben. Wenn mehrere Objekte mit dem gleichen Buchstaben beginnen, wird das nächst untere Objekt selektiert. Die Projektnavigation muss das aktuelle Oberflächenelement sein, damit Sie die Objekte über ihre Anfangsbuchstaben auswählen können.

Aufbau der Projektnavigation

Das folgende Bild zeigt beispielhaft die Bereiche der Projektnavigation:

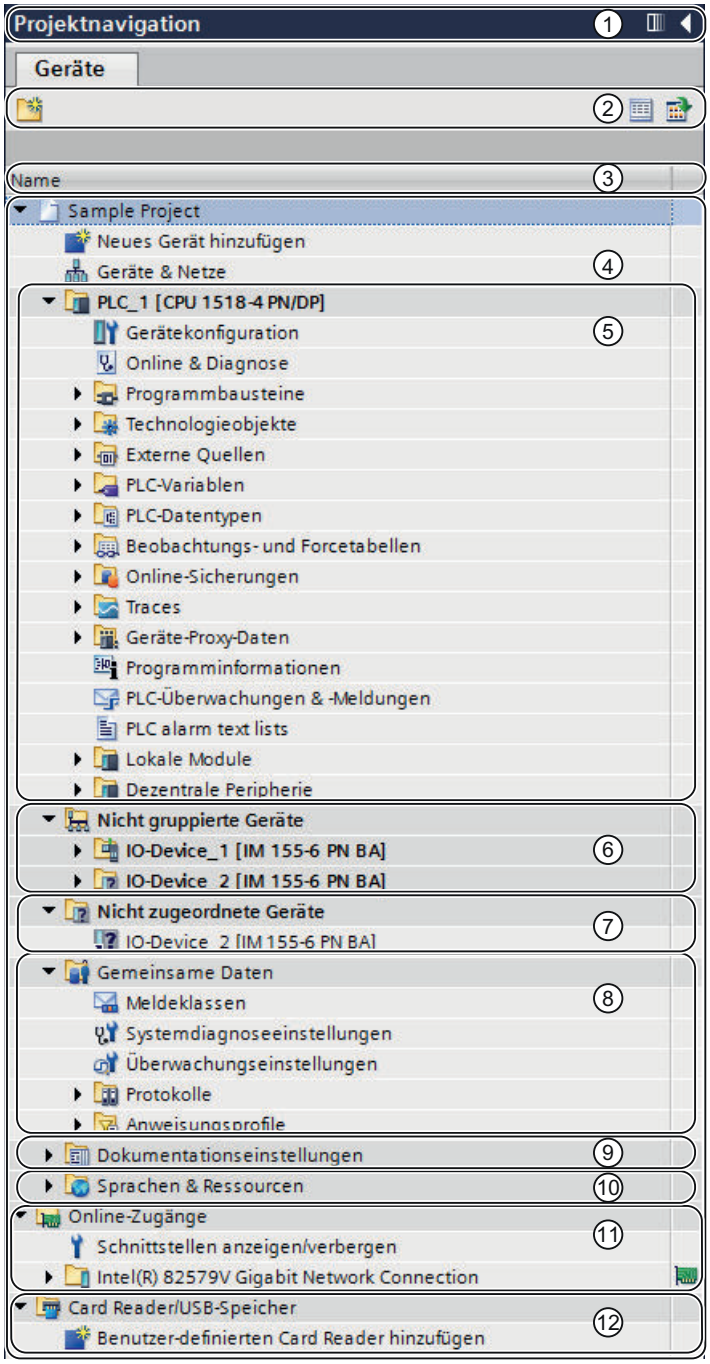

① Titelleiste
② Funktionsleiste
③ Spaltenkopf
④ Projekt

⑤ Geräte
⑥ Nicht gruppierte Geräte
⑦ Nicht zugeordnete Geräte
⑧ Gemeinsame Daten
⑨ Dokumentationseinstellungen
⑩ Sprachen & Ressourcen
⑪ Online-Zugänge
⑫ Card Reader/USB-Speicher

Titelleiste

In der Titelleiste der Projektnavigation finden Sie die Schaltflächen für das manuelle und automatische Zuklappen der Projektnavigation. Nach dem manuellen Zuklappen positioniert sich die Schaltfläche "Reduzieren" am linken Rand. Sie verändert sich von einem Pfeil-nach-links in einen Pfeil-nach-rechts und dient dem erneuten Aufklappen der Projektnavigation. Über die Schaltfläche "Automatisch reduzieren" können Sie die Projektnavigation automatisch zuklappen lassen, wenn Sie sie nicht benötigen.

Siehe auch: Arbeitsbereich maximieren und minimieren (Seite 32)

Funktionsleiste

Über die Funktionsleiste der Projektnavigation können Sie folgende Aktionen durchführen:

- Einen neuen Anwenderordner erstellen, z. B. zum Gruppieren von Bausteinen im Ordner "Programmbausteine" oder zum Gruppieren von Geräten
- Vorwärts zur Quelle eines Links und zurück zum Link navigieren
  Für Links in der Projektnavigation stehen zwei Schaltflächen zur Verfügung, über die Sie vom Link zur Quelle und zurück navigieren können.
- Übersicht zum selektierten Objekt im Arbeitsbereich einblenden
  Wenn Sie die Übersicht einblenden, werden die unterlagerten Objekte und Aktionen der Elemente in der Projektnavigation ausgeblendet.

Spaltenkopf

Standardmäßig wird die Spalte "Name" eingeblendet. Zusätzlich können Sie die Spalten "Typname" und "Version" einblenden. Wenn Sie die zusätzlichen Spalten einblenden, sehen Sie bei Instanzen von Typen aus der Bibliothek den Namen des jeweiligen Typen sowie die verwendete Version.

Projekt

Im Ordner "Projekt" finden Sie alle projektrelevanten Objekte und Aktionen, z. B.:

- Geräte
- Sprachen & Ressourcen
- Online-Zugänge

Gerät

Für jedes Gerät im Projekt gibt es einen eigenen Ordner mit dessen projektinternem Namen. Innerhalb dieser Ordner sind Objekte und Aktionen strukturiert, die zum Gerät gehören.

Nicht gruppierte Geräte

Im Ordner "Nicht gruppierte Geräte" werden alle dezentralen Peripheriegeräte des Projekts zusammengefasst.

Nicht zugeordnete Geräte

Dezentrale Peripheriegeräte, die keinem dezentralen Peripheriesystem zugeordnet sind, werden im Ordner "Nicht zugeordnete Geräte" als Verknüpfung angezeigt.

Gemeinsame Daten

In diesem Ordner finden Sie Daten, die Sie geräteübergreifend verwenden können, wie z. B. gemeinsame Meldeklassen, Protokolle und Skripte.

Dokumentationseinstellungen

In diesem Ordner legen Sie das Layout für die gedruckte Projektdokumentation fest.

Sprachen & Ressourcen

In diesem Ordner legen Sie die Projektsprachen und die Projekttexte fest.

Online-Zugänge

In diesem Ordner finden Sie alle Schnittstellen des PG/PC, auch wenn diese nicht zur Kommunikation mit einer Baugruppe verwendet werden.

Card Reader/USB-Speicher

In diesem Ordner werden alle an den PG/PC angeschlossenen Kartenleser und andere USB-Speichermedien verwaltet.

Siehe auch

Portalansicht (Seite 21)

Projektansicht (Seite 23)

Grundlagen zum Arbeitsbereich (Seite 30)

Inspektorfenster (Seite 38)

Grundlagen zu den Task Cards (Seite 40)

Detailansicht (Seite 43)

Spalten ein- und ausblenden (Seite 30)

2.2.5.2 Spalten ein- und ausblenden

Bei Bedarf blenden Sie in der Projektnavigation zusätzliche Spalten ein. In den zusätzlichen Spalten sehen Sie den Namen des zu einer Instanz gehörigen Typen sowie dessen Versionsnummer.

Vorgehen

Um zusätzliche Spalten für den zugehörigen Typen und dessen Versionsnummer ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste der Projektnavigation auf das Symbol "Spaltenüberschrift anzeigen/verbergen".
   Die Spaltenansicht der Projektnavigation wird aktiviert und ein Tabellenkopf wird eingeblendet.
2. Klicken Sie mit der rechten Maustaste auf den Tabellenkopf der Projektnavigation.
3. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen" und wählen Sie die Spalten aus, die Sie anzeigen möchten.
   Die gewünschten Spalten werden ein- oder ausgeblendet.

Siehe auch

Funktion und Aufbau der Projektnavigation (Seite 26)

2.2.6 Arbeitsbereich

2.2.6.1 Grundlagen zum Arbeitsbereich

Funktion des Arbeitsbereichs

Innerhalb des Arbeitsbereichs werden die Objekte angezeigt, die Sie zur Bearbeitung öffnen. Diese Objekte sind z. B.:

- Editoren und Sichten
- Tabellen

Sie können mehrere Objekte öffnen, von denen jedoch normalerweise nur eins im Arbeitsbereich zu sehen ist. Alle weiteren Objekte werden in der Editorleiste als Register angezeigt. Wenn Sie für bestimmte Aufgaben zwei Objekte gleichzeitig sehen möchten, können Sie den Arbeitsbereich vertikal oder horizontal teilen oder Elemente des Arbeitsbereichs abdocken. Falls Sie kein Objekt geöffnet haben, ist der Arbeitsbereich leer.

Aufbau des Arbeitsbereichs

Das folgende Bild zeigt beispielhaft einen vertikal geteilten Arbeitsbereich:

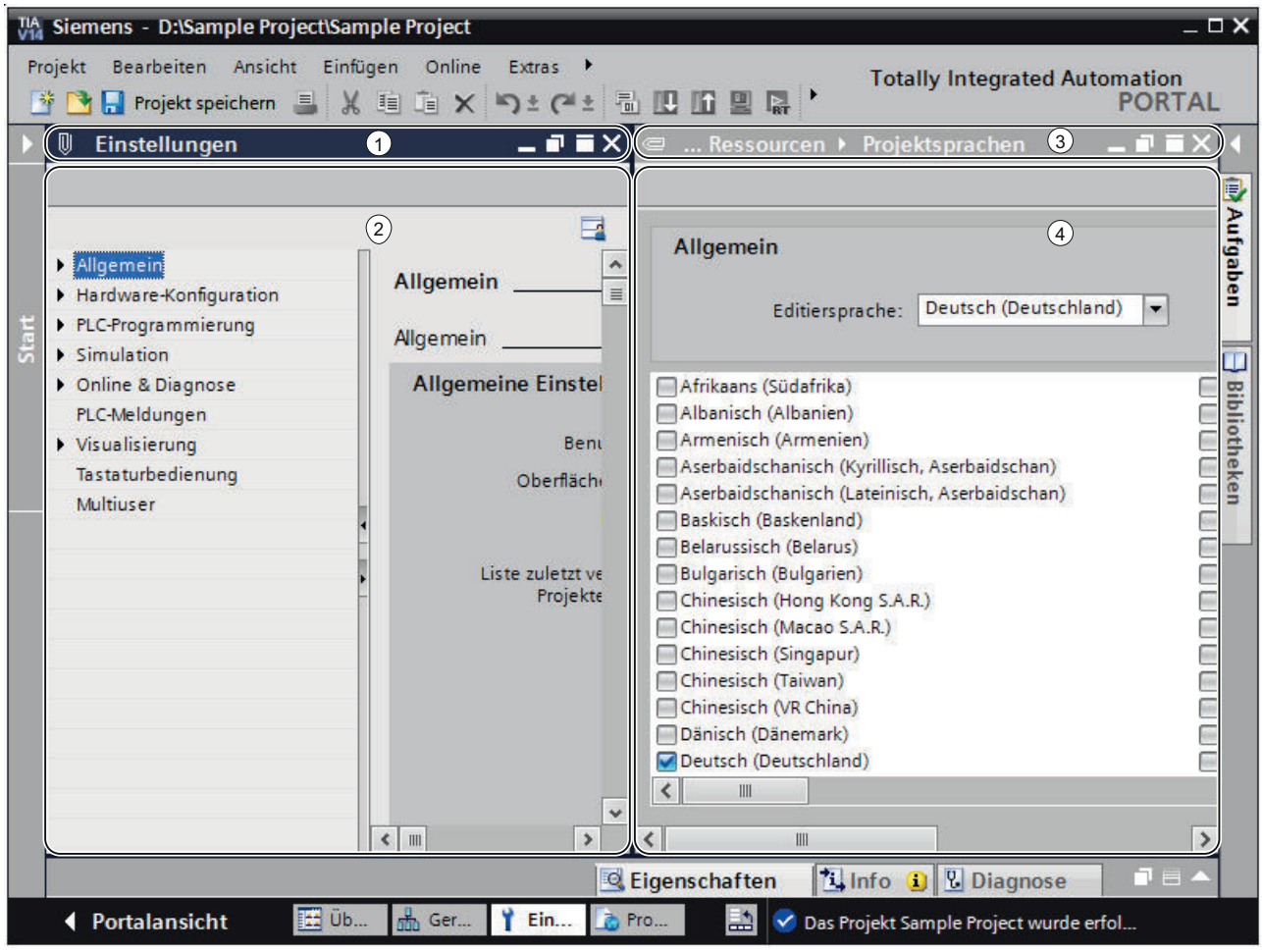

① Titelleiste linker Editor
② Arbeitsbereich linker Editor
③ Titelleiste rechter Editor
④ Arbeitsbereich rechter Editor

Siehe auch

Arbeitsbereich maximieren und minimieren (Seite 32)

Arbeitsbereich teilen (Seite 33)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

Speichern des Bedienoberflächenlayouts (Seite 50)

2.2.6.2 Arbeitsbereich maximieren und minimieren

Sie haben die Möglichkeit, den Arbeitsbereich so anzupassen, dass Ihnen eine möglichst große Arbeitsfläche zur Verfügung steht. Dazu können Sie folgende Funktion nutzen:

- Arbeitsbereich maximieren
  Sie können mit einem Klick die Task Cards, die Projektnavigation und das Inspektorfenster schließen. Dadurch wird die Fläche des Arbeitsbereichs vergrößert. Sie können den Arbeitsbereich jederzeit wieder minimieren, um zur vorherigen Ansicht zurückzukehren.
- Task Cards, Projektnavigation und Inspektorfenster automatisch zuklappen
  Sie können für die Task Cards, die Projektnavigation und das Inspektorfenster die Option "Automatisch reduzieren" verwenden. Dadurch klappen diese Fenster automatisch zu, wenn Sie sie nicht benötigen.

Arbeitsbereich maximieren und minimieren

Um den Arbeitsbereich zu maximieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie ein Element, z. B. einen Editor oder eine Tabelle.
   Das Element wird im Arbeitsbereich angezeigt.
2. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Maximieren".
   Die Task Cards, die Projektnavigation und das Inspektorfenster werden zugeklappt und der Arbeitsbereich wird mit maximaler Größe dargestellt.

Um den Arbeitsbereich wieder zu minimieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des angezeigten Elements auf die Schaltfläche "Einbetten".
   Sie erhalten wieder die Ansicht wie vor dem Maximieren des Arbeitsbereichs. Das heißt, wenn die Task Cards, die Projektnavigation oder das Inspektorfenster davor aufgeklappt waren, werden sie wieder aufgeklappt.

Task Cards, Projektnavigation und Inspektorfenster automatisch zuklappen

Um die Task Cards automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste der Task Cards auf "Automatisch reduzieren ".
   Die Task Cards klappen zu, wenn Sie mit der Maus auf eine Fläche außerhalb der Task Cards klicken.
2. Um die Task Cards zu verwenden, klicken Sie auf die zugeklappten Task Cards.
3. Die Task Cards klappen auf und Sie können sie verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um die Projektnavigation automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste der Projektnavigation auf "Automatisch reduzieren".
   Die Projektnavigation klappt zu, wenn Sie mit der Maus auf eine Fläche außerhalb der Projektnavigation klicken.
2. Um die Projektnavigation zu verwenden, klicken Sie auf die zugeklappte Projektnavigation.
   Die Projektnavigation klappt auf und Sie können sie verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um das Inspektorfenster automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Inspektorfensters auf "Automatisch reduzieren".
   Das Inspektorfenster klappt zu, wenn Sie mit der Maus auf eine Fläche außerhalb des Inspektorfensters klicken.
2. Um das Inspektorfenster zu verwenden, klicken Sie auf das zugeklappte Inspektorfenster.
   Das Inspektorfenster klappt auf und Sie können es verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um das automatische Zuklappen wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie im jeweiligen Fenster erneut auf "Dauerhaft erweitern".
   Die Option "Automatisch reduzieren" wird deaktiviert und das Fenster bleibt aufgeklappt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich teilen (Seite 33)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.6.3 Arbeitsbereich teilen

Sie können den Arbeitsbereich vertikal oder horizontal teilen.

Vorgehen

Um den Arbeitsbereich vertikal oder horizontal zu teilen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Editorbereich vertikal teilen" oder "Editorbereich horizontal teilen".
   Das angeklickte Element und das nächste Element der Editorleiste werden nebeneinander oder untereinander angezeigt.

Hinweis

Wenn kein Element im Arbeitsbereich geöffnet ist, stehen Ihnen die Funktionen "Editorbereich vertikal teilen" und "Editorbereich horizontal teilen" nicht zur Verfügung

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich maximieren und minimieren (Seite 32)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.6.4 Elemente des Arbeitsbereichs abdocken

Sie können folgende Elemente vom Arbeitsbereich abdocken und in einem eigenen Fenster öffnen:

- Editoren
- Tabellen
- Einstellungsfenster
- Task Cards
- Inspektorfenster

Abgedockte Elemente können Sie jederzeit wieder in den Arbeitsbereich einbetten.

Elemente des Arbeitsbereichs abdocken

Um Elemente aus dem Arbeitsbereich abzudocken, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Ablösen".
   Das Element wird aus dem Arbeitsbereich ausgelöst und als eigenes Fenster angezeigt. Sie können das Fenster nun frei platzieren. Wenn Sie das Fenster minimieren, können Sie es über die Editorleiste wiederherstellen.

Elemente im Arbeitsbereich einbetten

Um Elemente wieder im Arbeitsbereich einzubetten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Einbetten".
   Das Element wird wieder im Arbeitsbereich angezeigt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich maximieren und minimieren (Seite 32)

Arbeitsbereich teilen (Seite 33)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.6.5 Gruppierte Elemente des Arbeitsbereichs verwenden

Wenn Sie mehrere gleichartige Elemente geöffnet haben z. B. Editoren oder Tabellen, können Sie diese in der Editorleiste als Gruppe anzeigen lassen. Sie können diese Gruppen folgendermaßen verwenden:

- Einzelne Elemente einer Gruppe anzeigen
- Alle Elemente einer Gruppe in eigenen Fenstern anzeigen
- Alle angezeigten Elemente einer Gruppe in den Arbeitsbereich einbetten
- Alle angezeigten Elemente minimieren
- Alle Elemente einer Gruppe schließen

Sie können die Gruppierung von gleichartigen Elementen jederzeit wieder aufheben.

Gleichartige Elemente gruppieren oder Gruppierung aufheben

Um gleichartige Elemente in der Editorleiste zu gruppieren oder die Gruppierung wieder aufzuheben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste rechts auf die Schaltfläche "Editoren gruppieren/Gruppierung aufheben".
   Mit jedem Klick auf die Schaltfläche werden gleichartige Elemente gruppiert oder bestehende Gruppen werden aufgelöst. Das Symbol der Schaltfläche ändert sich dabei leicht, sodass Sie schnell erkennen können, in welchem Modus Sie arbeiten. Der eingestellte Modus bleibt auch nach dem Schließen der Elemente erhalten.

Einzelne Elemente einer Gruppe anzeigen

Um einzelne Elemente einer Gruppe anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste auf die Gruppe, in der sich das Element befindet, das Sie anzeigen möchten.
   Eine Liste mit allen verfügbaren Elementen der Gruppe wird angezeigt.
2. Klicken Sie auf das Element, das Sie anzeigen möchten.

Alle Elemente einer Gruppe in eigenen Fenstern anzeigen

Um alle Elemente einer Gruppe in eigenen Fenstern anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü "Gruppe wiederherstellen".
   Alle Elemente der Gruppe werden in eigenen Fenstern angezeigt, die sich überlagern. Verschieben Sie die Fenster, um die einzelnen Elemente zu sehen oder wählen Sie ein Element über die Gruppe in der Editorleiste.

Alle angezeigten Elemente einer Gruppe in den Arbeitsbereich einbetten

Um alle in eigenen Fenstern angezeigten Elemente einer Gruppe wieder in den Arbeitsbereich einzubetten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie einbetten möchten.
2. Wählen Sie im Kontextmenü "Gruppe einbetten".
   Alle Elemente der Gruppe werden wieder in den Arbeitsbereich eingebettet.

Alle angezeigten Elemente minimieren

Um alle Elemente einer Gruppe zu minimieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie minimieren möchten.
2. Wählen Sie im Kontextmenü "Gruppe minimieren".
   Alle Elemente der Gruppe werden minimiert. Die minimierten Elemente sind dennoch geöffnet und können in der Editorleiste über die Gruppe schnell wieder maximiert werden.

Alle Elemente einer Gruppe schließen

Um alle Elemente einer Gruppe zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie schließen möchten.
2. Wählen Sie im Kontextmenü "Gruppe schließen".
   Alle Elemente der Gruppe werden geschlossen. Die Gruppe wird entfernt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich maximieren und minimieren (Seite 32)

Arbeitsbereich teilen (Seite 33)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.6.6 Elemente des Arbeitsbereichs minimieren und maximieren

Sie können die geöffneten Elemente des Arbeitsbereichs, z. B. Editoren oder Tabellen, bei Bedarf minimieren. Ein minimiertes Element ist dennoch geöffnet und kann über die Editorleiste schnell wieder maximiert werden.

Elemente des Arbeitsbereichs minimieren

Um Elemente des Arbeitsbereichs zu minimieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Minimieren".
   Das Element wird minimiert und ist weiterhin über die Editorleiste verfügbar.

Um alle Elemente gleichzeitig zu minimieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Alle minimieren".

Elemente des Arbeitsbereichs maximieren

Um Elemente im Arbeitsbereich wieder zu maximieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste auf das gewünschte Element.
   Das Element wird wieder maximiert und im Arbeitsbereich angezeigt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich maximieren und minimieren (Seite 32)

Arbeitsbereich teilen (Seite 33)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.6.7 Zwischen den Elementen des Arbeitsbereichs wechseln

Sie können jederzeit zwischen den Elementen des Arbeitsbereichs wechseln.

Zwischen den Elementen des Arbeitsbereichs wechseln

Um zum vorherigen oder nächsten Editor zu wechseln, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Nächster Editor" oder "Vorheriger Editor".
   Der nächste oder der vorherige Editor wird angezeigt.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)
Arbeitsbereich maximieren und minimieren (Seite 32)
Arbeitsbereich teilen (Seite 33)
Elemente des Arbeitsbereichs abdocken (Seite 34)
Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)
Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)
Anordnung von Editoren und Tabellen sichern (Seite 52)

2.2.7 Inspektorfenster

Funktion des Inspektorfensters

Im Inspektorfenster werden zusätzliche Informationen zu einem selektierten Objekt oder zu ausgeführten Aktionen angezeigt.

Aufbau des Inspektorfensters

Die folgenden Bilder zeigen die Komponenten des Inspektorfensters:

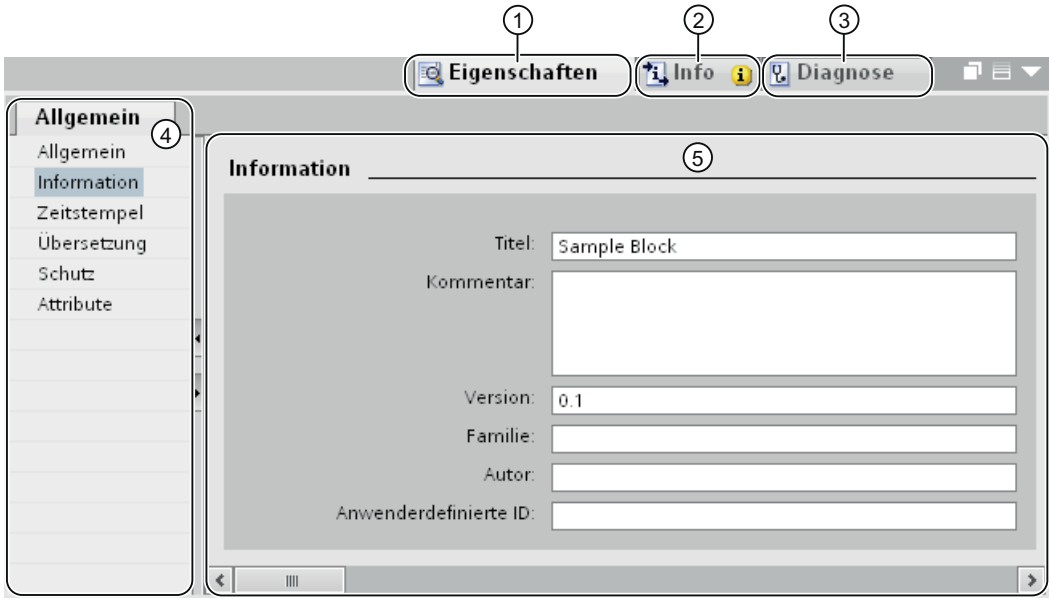

① Register "Eigenschaften"
② Register "Info"
③ Register "Diagnose"
④ Bereichsnavigation im Register "Eigenschaften"
⑤ Inhalt des Registers "Eigenschaften"

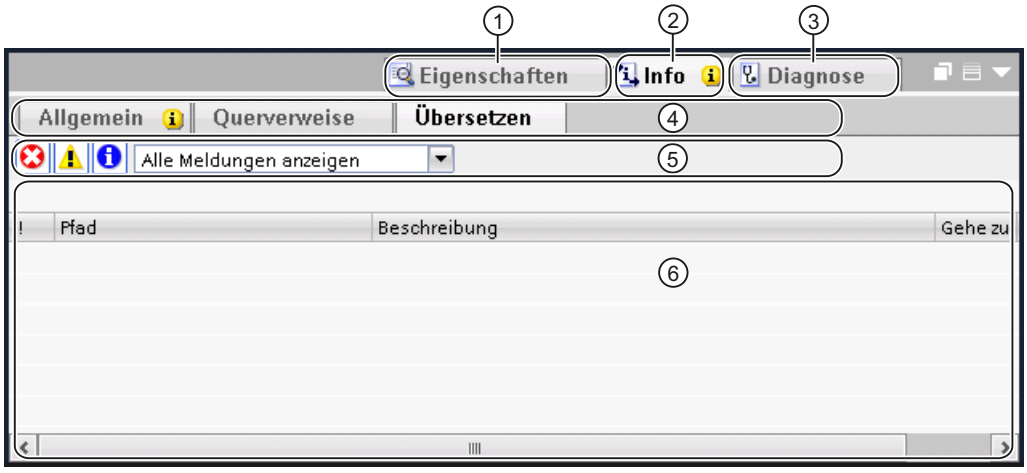

① Register "Eigenschaften"

② Register "Info"

③ Register "Diagnose"

④ Navigation innerhalb der Register durch weitere Register (nur in den Registern "Info" und "Diagnose" verfügbar)

⑤ Funktionsleiste (nur im Register "Info" in den untergeordneten Registern "Allgemein" und "Übersetzen" verfügbar)

⑥ Inhalt des Registers "Übersetzen" im Register "Info"

Register "Eigenschaften"

In diesem Register werden die Eigenschaften des selektierten Objekts angezeigt. Editierbare Eigenschaften können Sie hier verändern.

Register "Info"

In diesem Register werden weitere Informationen zum selektierten Objekt und Meldungen zu den ausgeführten Aktionen, z. B. Übersetzen, angezeigt.

Register "Diagnose"

In diesem Register erhalten Sie Informationen zu Systemdiagnose-Ereignissen, projektierten Meldungsereignissen und zur Verbindungsdiagnose.

Navigation innerhalb der Register

Über die Bereichsnavigation und die untergeordneten Register können Sie innerhalb der Register die gewünschten Informationen anzeigen.

Funktionsleiste

Über die Funktionsleiste in den Registern "Allgemein" und "Übersetzen" innerhalb des Registers "Info" können Sie festlegen, welche Arten von Meldungen angezeigt werden sollen. Sie können die Anzeige für die folgenden Meldungsarten getrennt aktivieren oder deaktivieren:

- Fehler
- Warnungen
- Informationen

Siehe auch

Funktion und Aufbau der Projektnavigation (Seite 26)

Grundlagen zum Arbeitsbereich (Seite 30)

Portalansicht (Seite 21)

Projektansicht (Seite 23)

Grundlagen zu den Task Cards (Seite 40)

Detailansicht (Seite 43)

2.2.8 Task Cards

2.2.8.1 Grundlagen zu den Task Cards

Funktion der Task Cards

Abhängig vom bearbeiteten oder selektierten Objekt stehen Ihnen Task Cards zur Verfügung, über die Sie weitere Aktionen ausführen können, z. B.:

- Objekte aus einer Bibliothek oder aus dem Hardware-Katalog auswählen
- Objekte im Projekt suchen und ersetzen
- Vordefinierte Objekte in den Arbeitsbereich ziehen

Die verfügbaren Task Cards finden Sie in einer Leiste am rechten Bildschirmrand. Sie können sie jederzeit auf- und zuklappen. Welche Task Cards Ihnen zur Verfügung stehen, hängt von den installierten Produkten ab. Komplexere Task Cards sind in Paletten unterteilt, die Sie ebenfalls auf- und zuklappen können.

Aufbau der Task Cards

Das folgende Bild zeigt beispielhaft die Leiste mit den Task Cards:

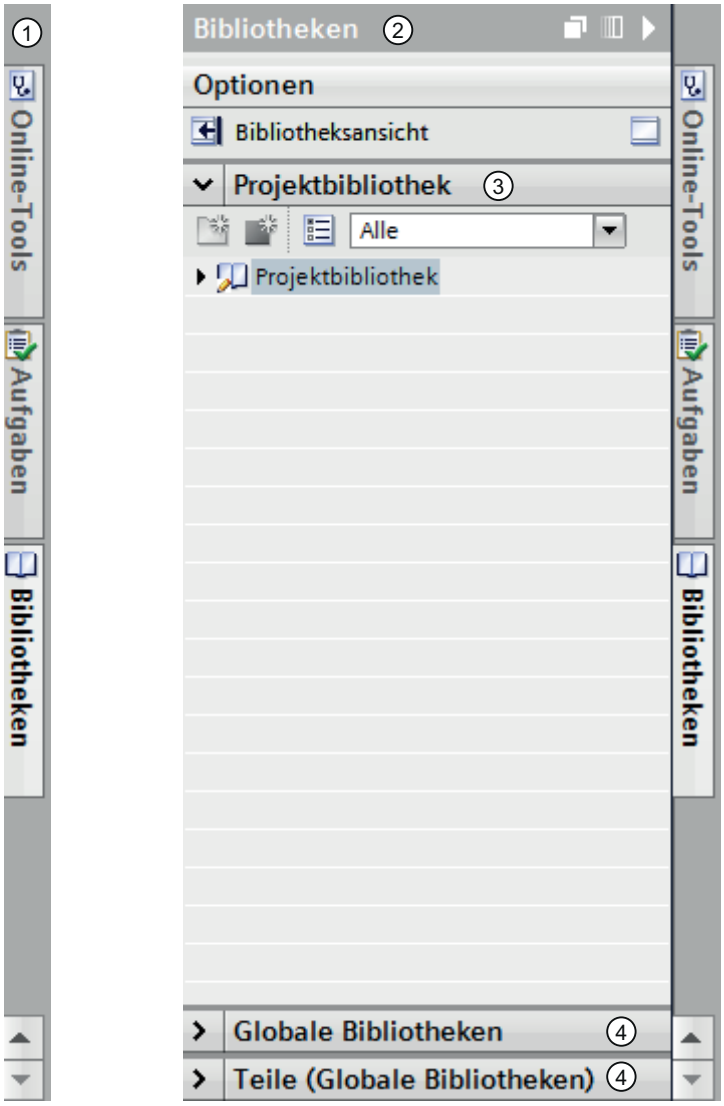

① Task Cards geschlossen
② Task Card geöffnet
③ Geöffnete Palette einer Task Card
④ Geschlossene Palette einer Task Card

Siehe auch

Palettenmodus ändern (Seite 42)

Funktion und Aufbau der Projektnavigation (Seite 26)

Grundlagen zum Arbeitsbereich (Seite 30)

Inspektorfenster (Seite 38)

Portalansicht (Seite 21)

Projektansicht (Seite 23)

Detailansicht (Seite 43)

2.2.8.2 Palettenmodus ändern

Sie können zwischen zwei Palettenmodi wählen:

- Einzelpalettenmodus:
  Es ist immer nur eine Palette geöffnet. Wenn Sie eine weitere Palette öffnen, wird die bisher geöffnete Palette automatisch geschlossen.
- Mehrpalettenmodus:
  Sie können mehrere Paletten gleichzeitig öffnen.

Vorgehen

Um den Palettenmodus zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb einer Task Card über den Paletten auf die Schaltfläche "Palettenmodus ändern".

Siehe auch

Grundlagen zu den Task Cards (Seite 40)

2.2.9 Referenzprojekte

Funktion von Referenzprojekten

In der Palette "Referenzprojekte" können Sie zusätzlich zum aktuellen Projekt weitere Projekte öffnen. Diese Referenzprojekte werden schreibgeschützt geöffnet und können nicht bearbeitet werden. Sie können die Objekte eines Referenzprojekts aber in Ihr aktuelles Projekt ziehen und dort weiter bearbeiten. Zusätzlich können Sie die Objekte eines Referenzprojekts mit den Objekten Ihres aktuellen Projekts vergleichen.

Aufbau der Palette "Referenzprojekte"

Das folgende Bild zeigt den Aufbau der Palette "Referenzprojekte":

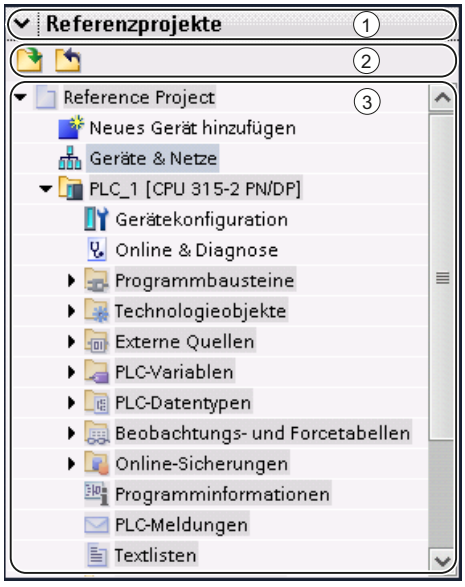

① Titelleiste
② Funktionsleiste
③ Geöffnete Referenzprojekte

Titelleiste

In der Titelleiste der Palette "Referenzprojekte" finden Sie den Pfeil für das Schließen der Palette. Nach dem Schließen verändert er sich von einem Pfeil-nach-unten in einen Pfeil-nach-rechts und dient dem Öffnen der Palette.

Funktionsleiste

In der Funktionsleiste finden Sie die Schaltflächen für das Öffnen und Schließen von Referenzprojekten.

Geöffnete Referenzprojekte

Geöffnete Referenzprojekte werden mit ihren Objekten und ihrer hierarchischen Struktur schreibgeschützt angezeigt.

2.2.10 Detailansicht

Funktion der Detailansicht

In der Detailansicht werden bestimmte Inhalte eines selektierten Objekts im Übersichtsfenster oder in der Projektnavigation angezeigt. Mögliche Inhalte sind beispielsweise Textlisten oder Variablen.

Aufbau der Detailansicht

Das folgende Bild zeigt ein Beispiel für die Detailansicht:

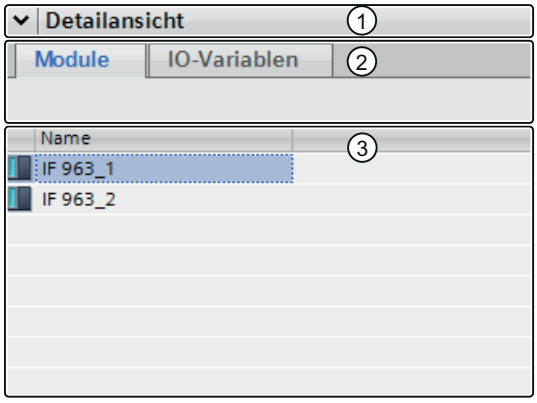

① Titelleiste
② Register
③ Inhalte des selektierten Objekts

Titelleiste

In der Titelleiste der Detailansicht finden Sie den Pfeil für das Schließen der Detailansicht. Nach dem Schließen verändert er sich von einem Pfeil-nach-unten in einen Pfeil-nach-rechts und dient dem Öffnen der Detailansicht.

Register

Abhängig vom selektierten Objekt werden in der Detailansicht Register eingeblendet.

- Das Register "Module" wird eingeblendet, wenn Sie ein Gerät, eine Geräteverknüpfung oder einen Ordner selektieren, der eines dieser Objekte enthält.
  Das Register "Module" zeigt die gesteckten Module oder Submodule eines Geräts in der Detailansicht an, wenn Sie das Gerät selbst oder eine Geräteverknüpfung in der Projektnavigation selektieren.
- Das Register "IO-Variablen" zeigt vorhandene IO-Variablen zu selektierten Geräten und Geräteverknüpfungen unterhalb des Ordners "Lokale Module" oder eines Geräteordners an. Neben dem Datentyp und der Adresse wird der Name der einfachen Variable oder des Datenelements eines PLC-Datentyps angezeigt.

  Hinweis

  Wenn für eine Kanaladresse eine einfache Variable definiert ist, diese aber auch im Adressbereich des Datenelements eines PLC-Datentyps liegt, wird nur der Name der einfachen Variable angezeigt.

- Das Register "Systemkonstanten" wird eingeblendet, wenn Sie unterhalb eines Geräteordners im Ordner "Dezentrale Peripherie" oder im Ordner "Nicht zugeordnete Geräte" eine Geräteverknüpfung selektieren.
  Das Register "Systemkonstanten" zeigt die vom Gerät verwendeten globalen Konstanten mit den HW-Kennungen, zum Beispiel der Module, Submodule und Schnittstellen an.

Objekte

Die angezeigten Inhalte variieren abhängig vom selektierten Objekt. Sie können die Objektinhalte per Drag & Drop direkt aus der Detailansicht an die gewünschte Verwendungsstelle ziehen.

Siehe auch

Funktion und Aufbau der Projektnavigation (Seite 26)

Grundlagen zum Arbeitsbereich (Seite 30)

Inspektorfenster (Seite 38)

Grundlagen zu den Task Cards (Seite 40)

Portalansicht (Seite 21)

Projektansicht (Seite 23)

2.2.11 Übersichtsfenster

2.2.11.1 Übersichtsfenster

Funktionen des Übersichtsfensters

Das Übersichtsfenster dient zur Ergänzung der Projektnavigation. Das Übersichtsfenster zeigt die Inhalte des aktuell in der Projektnavigation selektierten Ordners an.

Außerdem können Sie im Übersichtsfenster folgende Aktionen durchführen:

- Objekte öffnen
- Eigenschaften von Objekten im Inspektorfenster anzeigen und bearbeiten
- Objekte umbenennen
- Objektspezifische Aktionen über das Kontextmenü aufrufen
- Objekte in einer Gegenüberstellung vergleichen
- Verschiedene Objektoperationen durchführen, wie z. B. Objekte per Drag & Drop aus der Bibliothek einfügen, Objekte verschieben, kopieren, einfügen und löschen

Aufbau des Übersichtsfensters

Das folgende Bild zeigt die Komponenten des Übersichtsfensters:

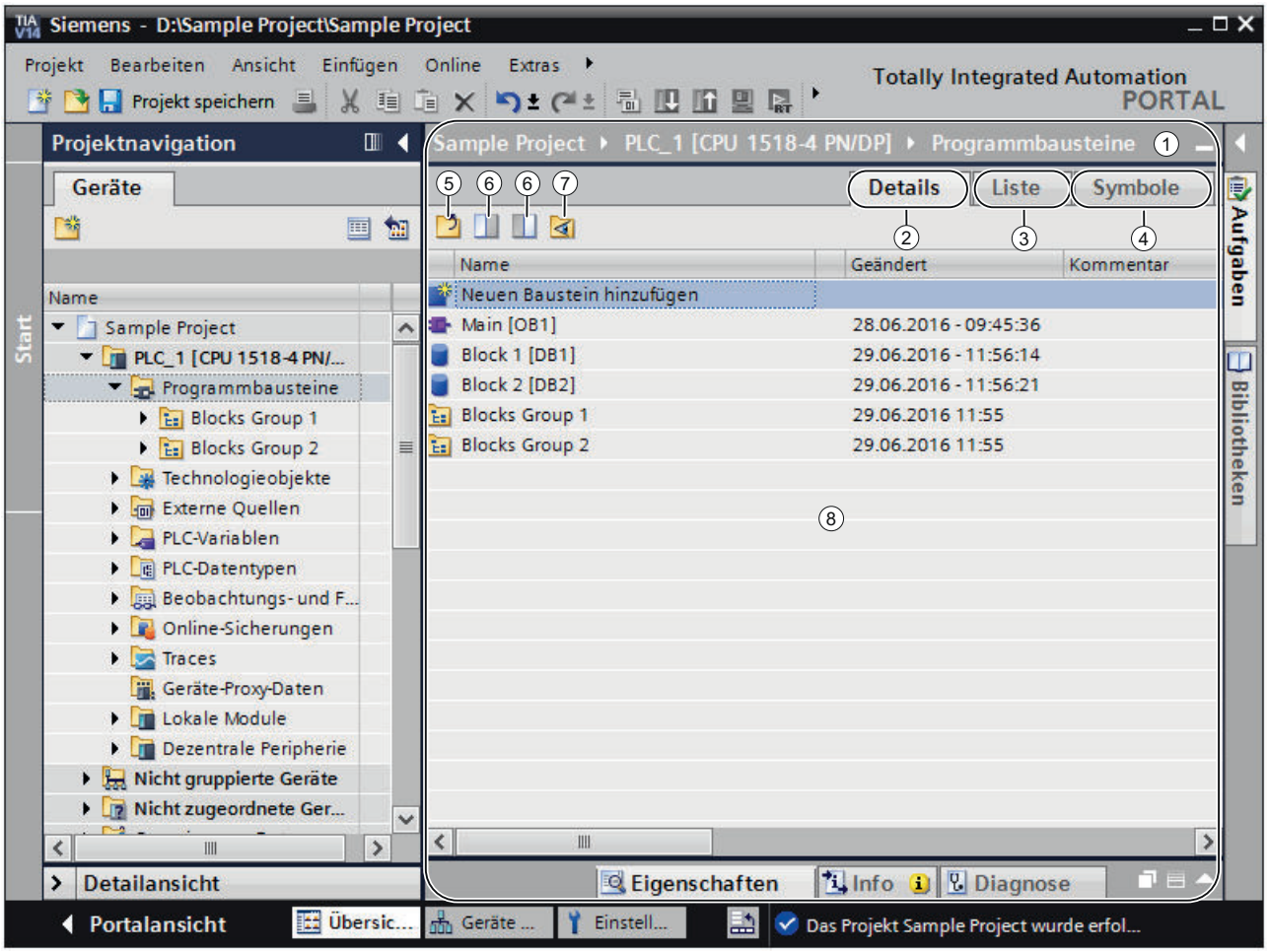

① Übersichtsfenster

② Umschalten in die Detailansicht

③ Umschalten in die Listenansicht

④ Umschalten in die Symbolansicht

⑤ Wechseln zur übergeordneten Ebene

⑥ Teilen des Übersichtsfensters in zwei Seiten. Es wird entweder die rechte oder linke Hälfte des Übersichtsfensters synchronisiert. Durch erneutes Klicken wird die Teilung wieder aufgehoben.

⑦ Es werden alle Elemente innerhalb eines selektierten Ordners angezeigt, auch wenn diese sich in unterlagerten Gruppen befinden. Diese Option ist nur in der Detailsicht verfügbar.

⑧ Inhalte des Objekts, welches in der Projektnavigation selektiert ist.

Darstellungsformen des Übersichtsfensters

Die Inhalte des Übersichtsfensters können folgendermaßen dargestellt werden:

- Detailansicht
  Die Objekte werden in einer Liste mit zusätzlichen Informationen, wie z. B. dem Änderungsdatum, angezeigt.
- Listenansicht
  Die Objekte werden in einer einfachen Liste angezeigt.
- Symbolansicht
  Die Objekte werden symbolisch nach Kategorien geordnet angezeigt.

Siehe auch

Objekte im Übersichtsfenster vergleichen (Seite 48)

Detailansicht des Übersichtsfensters sortieren (Seite 49)

2.2.11.2 Objekte im Übersichtsfenster vergleichen

Im Übersichtsfenster können Sie die Inhalte zweier Ordner oder Objekte gegenüberstellen und vergleichen. Das Übersichtsfenster wird in der Mitte geteilt und Sie können auf der linken und rechten Seite jeweils unterschiedliche Inhalte anzeigen.

Darüber hinaus ist es möglich, Objekte zwischen den geteilten Fenstern per Drag & Drop zu verschieben. Auf diese Weise können Sie beispielsweise Inhalte von einem Ordner in einen anderen verschieben.

Vorgehen

Um das Übersichtsfenster in der Mitte zu teilen, oder um die Teilung wieder aufzuheben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Linke Seite synchronisieren" oder "Rechte Seite synchronisieren", um das Übersichtsfenster zu teilen. Dabei wird entweder die linke oder die rechte Seite des Übersichtsfensters mit den Inhalten des selektierten Objekts in der Projektnavigation synchronisiert.
2. Um die Teilung wieder aufzuheben, klicken Sie erneut auf das zuvor gewählte Symbol.

Siehe auch

Übersichtsfenster (Seite 46)

2.2.11.3 Detailansicht des Übersichtsfensters sortieren

In der Detailansicht des Übersichtsfensters stehen Ihnen einige Möglichkeiten zur Verfügung, die Darstellung anzupassen:

- Einfügen zusätzlicher Spalten
  Einige Spalten sind standardmäßig ausgeblendet, um die Übersichtlichkeit zu erhöhen. Nicht angezeigte Spalten blenden Sie bei Bedarf ein. Welche Spalten zur Verfügung stehen, ist abhängig vom gewählten Objekt.
- Anzeigen von Ordnerinhalten in flacher Hierarchie
  Ordnerinhalte lassen sich in einer flachen Hierarchie anzeigen. Alle Inhalte werden auf einmal angezeigt, auch wenn sich diese in verschiedenen Gruppen befinden.
- Sortieren der Tabellenspalten
  Einzelne Spalten der Tabelle lassen sich auf- oder absteigend sortieren.

Spalten ein- oder ausblenden

Um zusätzliche Spalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Titelleiste der Tabelle.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen" und selektieren Sie die Spalten, die angezeigt werden sollen.

Ordnerinhalte in flacher Hierarchie anzeigen

Um die Inhalte eines Ordners in einer flachen Hierarchie anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gewünschten Ordner in der Projektnavigation oder in der Bibliotheksnavigation der Bibliotheksansicht.
2. Klicken Sie in der Funktionsleiste auf das Symbol "Unterlagerte Elemente anzeigen". Alle Elemente werden auf einmal in der Tabelle angezeigt, auch wenn diese sich in unterlagerten Ordnern befinden.

Tabelle auf- oder absteigend sortieren

Um die Tabelle nach einer Spalte auf- oder absteigend zu sortieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Tabellenkopf einer Spalte, wenn Sie diese aufsteigend sortieren möchten.
2. Klicken Sie erneut auf den Tabellenkopf derselben Spalte, um die Spalte absteigend zu sortieren.
3. Klicken Sie ein drittes Mal auf den Tabellenkopf derselben Spalte, um die Sortierung wieder aufzuheben.

Siehe auch

Übersichtsfenster (Seite 46)

2.2.12 Bedienoberflächenlayout

2.2.12.1 Speichern des Bedienoberflächenlayouts

Möglichkeiten das Bedienoberflächenlayout zu sichern

Wenn Sie eine Änderung an der Bedienoberfläche vornehmen, dann bleibt diese auch nach einem Neustart des TIA Portals erhalten. Zu einer Änderung am Bedienoberflächenlayout zählt beispielsweise das Verschieben eines Fensters oder das Anpassen der Größe eines Editors.

Neben dem automatischen Speichern des Bedienoberflächenlayouts haben Sie die Möglichkeit bestimmte Anordnungen manuell zu sichern:

- Speichern des Fensterlayouts
  Sie können die Anordnungen der Fenster und Editoren des TIA Portals manuell speichern und zu einem späteren Zeitpunkt wiederherstellen. Fünf Fensterlayouts lassen sich über eine Tastenkombination aufrufen.
  Verwenden Sie diese Funktion beispielsweise, wenn Sie mit einem Notebook arbeiten, welches Sie bei Bedarf um einen externen Monitor ergänzen. So können Sie sich jeweils ein Fensterlayout für den mobilen Einsatz auf dem Notebook-Display und eines für die Arbeit im Büro mit einem externen Monitor anlegen.
- Speichern der Anordnung innerhalb von Editoren
  Bei einigen Editoren können Sie die Darstellung anpassen. Sie können z. B. die Breite von Tabellen anpassen oder einzelnen Tabellenspalten ein- und ausblenden.

Siehe auch

Fensterlayout speichern (Seite 50)

Fensterlayout laden (Seite 51)

Fensterlayouts verwalten (Seite 52)

Anordnung von Editoren und Tabellen sichern (Seite 52)

Bedienoberfächenlayout zurücksetzen (Seite 53)

Grundlagen zum Arbeitsbereich (Seite 30)

2.2.12.2 Fensterlayout speichern

Sie können das aktuelle Fensterlayout speichern, um es zu einem späteren Zeitpunkt in derselben Form wieder aufzurufen.

Vorgehen

Um ein Fensterlayout zu speichern, gehen Sie folgendermaßen vor:

1. Ordnen Sie alle Fenster so an, wie Sie diese speichern möchten.
2. Wählen Sie im Menü "Fenster" den Befehl "Fensterlayout speichern unter".
   Der Dialog "Aktuelles Fensterlayout speichern unter" wird geöffnet.

3. Geben Sie in das Feld "Name" einen Namen für das Fensterlayout ein.
4. Geben Sie eine Beschreibung des Fensterlayouts in das Feld "Beschreibung" ein, um das Fensterlayout später leichter identifizieren zu können.
5. Klicken Sie auf die Schaltfläche "Speichern".

Ergebnis

Das neue Fensterlayout wird an letzter Position nach den bereits vorhandenen Fensterlayouts gespeichert. Die ersten fünf Fensterlayouts können über eine Tastenkombination aufgerufen werden.

Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 50)

2.2.12.3 Fensterlayout laden

Wenn Sie bereits ein Fensterlayout gespeichert haben, können Sie dieses laden und so Ihre Arbeitsumgebung schnell an die jeweiligen Gegebenheiten anpassen. Die ersten fünf Fensterlayouts können Sie im Schnellzugriff über das Menü "Fenster" oder über eine Tastenkombination laden.

Wenn Sie ein Fensterlayout geladen haben und anschließend Änderungen an der Anordnung der Fenster vorgenommen haben, können Sie das ursprünglich gespeicherte Fensterlayout wiederherstellen.

Fensterlayouts 1 bis 5 über einen Schnellzugriff laden

Um eines von den ersten fünf gespeicherten Fensterlayouts zu laden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" ein Fensterlayout aus oder wählen Sie die Tastenkombination <Alt+Shift+[1 ... 5]>.

Weitere Fensterlayouts laden

Um ein Fensterlayout zu wählen, welches sich nicht unter den ersten fünf Fensterlayouts befindet, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Weitere Fensterlayouts".
   Der Dialog "Alle Fensterlayouts verwalten" wird geöffnet.
2. Selektieren Sie das gewünschte Fensterlayout.
3. Klicken Sie auf "OK".

Fensterlayout wiederherstellen

Um zum gespeicherten Fensterlayout zurückzukehren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Fensterlayout wiederherstellen" oder wählen Sie die Tastenkombination <Alt+Shift+0>.

Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 50)

2.2.12.4 Fensterlayouts verwalten

Sie können bei bereits vorhandenen Fensterlayouts die folgenden Aktionen durchführen:

- Reihenfolge der Fensterlayouts ändern
  Die Reihenfolge der Fensterlayouts ist entscheidend, da sich die ersten fünf Fensterlayouts direkt über das Menü "Fenster" und eine Tastenkombination aufrufen lassen.
- Wählen eines Fensterlayouts
  Wenn sich ein Fensterlayout nicht unter den ersten fünf Fensterlayouts befindet, können Sie es über den Dialog "Alle Fensterlayouts verwalten" aufrufen.
- Löschen von Fensterlayouts

Vorgehen

Um die vorhandenen Fensterlayouts zu verwalten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Navigieren Sie in der Bereichsnavigation zum Eintrag "Allgemein > Alle Fensterlayouts verwalten".
3. Wählen Sie ein Fensterlayout aus.
   Das ausgewählte Fensterlayout wird sofort aktiviert.
4. Klicken Sie auf das Symbol "aufwärts " oder "abwärts", um das Fensterlayout nach vorne oder nach hinten zu bringen.
5. Klicken Sie auf das Symbol "löschen", um das selektierte Fensterlayout zu löschen.

Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 50)

2.2.12.5 Anordnung von Editoren und Tabellen sichern

Sie haben die Möglichkeit, Editoren und Tabellen an Ihre Bedürfnisse anzupassen. So können Sie z. B. in Tabellen die Spalten ausblenden, die Sie nicht benötigen. Die von Ihnen angepasste Ansicht können Sie anschließend sichern.

Vorgehen

Um die Anordnung von Editoren und Tabellen des Arbeitsbereichs zu sichern, gehen Sie folgendermaßen vor:

1. Passen Sie den Editor oder die Tabelle nach Ihren Bedürfnissen an.
2. Klicken Sie innerhalb des Editors oder der Tabelle auf die Schaltfläche "Anordnung merken".

Ergebnis

Die Anordnung wird gesichert. Wenn Sie den Editor oder die Tabelle erneut öffnen, wird diese Anordnung verwendet.

Siehe auch

Grundlagen zum Arbeitsbereich (Seite 30)

Arbeitsbereich maximieren und minimieren (Seite 32)

Arbeitsbereich teilen (Seite 33)

Elemente des Arbeitsbereichs abdocken (Seite 34)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 35)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 37)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 37)

Speichern des Bedienoberflächenlayouts (Seite 50)

2.2.12.6 Bedienoberfächenlayout zurücksetzen

Jede Änderung, die Sie am Layout der Bedienoberfläche vornehmen, wird gespeichert. Die Änderungen stehen somit auch nach dem Neustart des TIA Portals zur Verfügung. Ändern Sie beispielsweise die Höhe und Breite eines Texteditors, oder verändern die Aufteilung einer Tabelle, so bleiben Ihre Änderungen dauerhaft erhalten und Sie müssen Ihre individuellen Anpassungen nicht erneut vornehmen.

In manchen Fällen kann es jedoch sinnvoll sein, die ursprünglichen Layout-Einstellungen wiederherzustellen, z. B. wenn ein weiterer Nutzer eine andere Aufteilung der Bedienoberfläche bevorzugt.

Vorgehen

Um die Einstellungen der Bedienoberfläche auf Standardeinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Klicken Sie unter "Auf Standard zurücksetzen > Editoren-Layout" auf die Schaltfläche "Auf Standard zurücksetzen".

Ergebnis

Die Standardeinstellungen für die Bedienoberfläche werden wiederhergestellt.

Siehe auch

Übersicht der Programmeinstellungen (Seite 10)
Speichern des Bedienoberflächenlayouts (Seite 50)

2.3 Externe Applikationen ausführen

2.3.1 Grundlagen zur Ausführung von externen Applikationen

Einführung

Sie haben die Möglichkeit, bis zu zwanzig externe Applikationen zu konfigurieren, die Sie direkt aus dem TIA Portal heraus ausführen können. Für diese Applikationen können Sie bei der Konfiguration Argumente festlegen. Sie können dazu eigene Argumente definieren, die die Applikation versteht oder Sie können die folgenden Systemvariablen des TIA Portals als Argumente auswählen:

- ProjectFileName: Name des aktuell geöffneten TIA Portal-Projekts
- ProjectVersion: Version des aktuell geöffneten TIA Portal-Projekts
- ProjectPath: Pfad des aktuell geöffneten TIA Portal-Projekts
- ProjectDirectory: Projektverzeichnis des aktuell geöffneten TIA Portal-Projekts
- ProjectExtension: Dateinamenerweiterung des aktuell geöffneten TIA Portal-Projekts
- BinaryDirectory: Pfad zum Verzeichnis "Bin" der geöffneten TIA Portal-Instanz
- ProductVersion: Version des geöffneten TIA Portals

Sie können die eigenen Argumente auch mit den Systemvariablen mischen.

Zusätzlich können Sie den Startpfad für die Applikation festlegen. Hier können Sie entweder einen der folgenden beiden vordefinierten Werte verwenden oder einen eigenen Startpfad eintragen:

- ProjectDirectory: Projektverzeichnis des aktuell geöffneten TIA Portal-Projekts
- BinaryDirectory: Pfad zum Verzeichnis "Bin" der geöffneten TIA Portal-Instanz

Siehe auch

Externe Applikationen konfigurieren (Seite 55)

Externe Applikation starten (Seite 57)

2.3.2 Externe Applikationen konfigurieren

Bevor Sie externe Applikationen aus dem TIA Portal heraus aufrufen können, müssen Sie diese Applikationen zunächst konfigurieren. Für bereits vorhandene externe Applikationen können Sie die Konfiguration nachträglich ändern und die Reihenfolge bearbeiten, in der sie im Menü "Werkzeuge > Externe Applikationen" angezeigt werden. Externe Applikationen, die Sie nicht mehr aufrufen möchten, können Sie löschen.

Neue externe Applikation hinzufügen und konfigurieren

Um eine neue externe Applikation zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Werkzeuge > Externe Applikationen" auf den Befehl "Konfigurieren" oder öffnen Sie über das Menü "Extras > Einstellungen > Externe Applikationen" die Liste der Applikationen.
   Die Liste der Applikationen wird im Arbeitsbereich angezeigt.
2. Klicken Sie doppelt auf "Hinzufügen" oder wählen Sie im Kontextmenü den Befehl "Hinzufügen".
   Der Dialog "Neue externe Applikation hinzufügen" wird geöffnet.
3. Geben Sie den Namen der externen Applikation ein. Dieser Name darf noch nicht für eine zuvor konfigurierte Applikation verwendet werden und die Anzahl von 100 Zeichen nicht übersteigen.
4. Geben Sie den Befehl ein oder wählen Sie den Pfad zur ausführbaren Datei aus.
5. Optional: Geben Sie die Argumente ein, die beim Starten der externen Applikation übergeben werden sollen. Zusätzlich können Sie über die Schaltfläche "+" neben dem Feld "Argumente" eine Liste mit Systemvariablen des TIA Portals öffnen und diese als Argumente auswählen.

   Hinweis

   Falls beim Starten der externen Applikation ein Argument nicht übergeben werden kann, z. B. eine Systemvariable, wird dieses Argument als leere Zeichenkette übergeben.

6. Optional: Klicken Sie auf die Schaltfläche "+" neben dem Feld "Starten in" und wählen Sie die Startposition aus.
7. Klicken Sie auf "Hinzufügen".
   Die neue externe Applikation wird in die Liste der Applikationen aufgenommen und kann ab sofort über das Menü "Werkzeuge > Externe Applikationen" über den eingegebenen Namen gestartet werden. Die Reihenfolge der Anzeige im Menü "Werkzeuge > Externe Applikationen" entspricht der Reihenfolge in der Liste der Applikationen. Sie können diese Reihenfolge beliebig verändern.

Konfiguration einer vorhandenen externen Applikation ändern

Um eine vorhandene externe Applikation zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Werkzeuge > Externe Applikationen" auf den Befehl "Konfigurieren" oder öffnen Sie über das Menü "Extras > Einstellungen > Externe Applikationen" die Liste der Applikationen.
   Die Liste der Applikationen wird im Arbeitsbereich angezeigt.
2. Selektieren Sie die externe Applikation, die Sie ändern möchten.
3. Klicken Sie auf die Schaltfläche "Bearbeiten" oder wählen Sie im Kontextmenü den Befehl "Bearbeiten".
   Der Dialog "Externe Applikation ändern" wird geöffnet.
4. Verändern Sie die Konfiguration entsprechend Ihren Wünschen.
5. Klicken Sie auf "Ändern".

Reihenfolge der externen Applikationen in der Liste ändern

Um die Reihenfolge der externen Applikationen in der Liste und damit auch im Menü "Werkzeuge > Externe Applikationen" zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Werkzeuge > Externe Applikationen" auf den Befehl "Konfigurieren" oder öffnen Sie über das Menü "Extras > Einstellungen > Externe Applikationen" die Liste der Applikationen.
   Die Liste der Applikationen wird im Arbeitsbereich angezeigt.
2. Selektieren Sie die externe Applikation, deren Position Sie ändern möchten.
3. Klicken Sie auf die Schaltfläche "Nach oben bewegen", um die Applikation in der Liste weiter nach oben zu verschieben.
4. Klicken Sie auf die Schaltfläche "Nach unten bewegen", um die Applikation in der Liste weiter nach unten zu verschieben.
5. Wiederholen Sie dies für alle Applikationen, für die Sie die Reihenfolge ändern möchten.

Vorhandene externe Applikation löschen

Um eine vorhandene externe Applikation zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Werkzeuge > Externe Applikationen" auf den Befehl "Konfigurieren" oder öffnen Sie über das Menü "Extras > Einstellungen > Externe Applikationen" die Liste der Applikationen.
   Die Liste der Applikationen wird im Arbeitsbereich angezeigt.
2. Selektieren Sie die externe Applikation, die Sie löschen möchten.
3. Klicken Sie auf die Schaltfläche "Löschen" oder wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Dialog "Löschen bestätigen" wird geöffnet.
4. Klicken Sie auf "Ja", um die externe Applikation zu löschen.

Siehe auch

Grundlagen zur Ausführung von externen Applikationen (Seite 55)

Externe Applikation starten (Seite 57)

2.3.3 Externe Applikation starten

Voraussetzung

Sie haben eine externe Applikation konfiguriert.

Vorgehen

Um eine externe Applikation zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Werkzeuge > Externe Applikation" auf die externe Applikation, die Sie starten möchten.
   Wenn Sie die Applikation zum ersten Mal starten oder seit dem letzten Start die Konfiguration für diese externe Applikation verändert wurde, wird der Dialog "Applikation starten" angezeigt. Führen Sie dann noch Schritt 2 aus.
2. Klicken Sie auf "OK", wenn Sie die externe Applikation trotz möglicher Sicherheitsrisiken starten möchten. Ansonsten brechen Sie den Start mit einem Klick auf "Abbrechen" ab.

Siehe auch

Grundlagen zur Ausführung von externen Applikationen (Seite 55)

Externe Applikationen konfigurieren (Seite 55)

2.4 Tastaturbedienung im TIA Portal

2.4.1 Bedienung des TIA Portals mit der Tastatur

Sie können sich durch das TIA Portal mithilfe der Tastatur bewegen, wenn Sie zum Beispiel gerade keine Maus zur Verfügung haben. Für viele Funktionen sind zusätzlich Tastenkombinationen verfügbar und Sie können alternativ benutzerdefinierte Tastenkombinationen verwenden. Eine Übersicht aller Tastenkombinationen und die Möglichkeit, benutzerdefinierte Tastenkombinationen zuzuweisen, finden Sie in den Einstellungen des TIA Portals.

In den folgenden Kapiteln erfahren Sie, wie Sie sich mithilfe der Tastatur im TIA Portal bewegen, Objekte bearbeiten und das TIA Portal an Ihre Bedürfnisse anpassen können.

Siehe auch

Übersicht aller Tastenkombinationen anzeigen (Seite 59)

Benutzerdefinierte Tastenkombinationen verwenden (Seite 71)

2.4.2 Übersicht aller Tastenkombinationen anzeigen

Sie können sich eine Übersicht aller Tastenkombinationen anzeigen lassen.

Vorgehen

Um eine Übersicht aller möglichen Tastenkombinationen anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".
   Sie sehen eine Übersicht aller vorgegebenen und benutzerdefinierten Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.

2.4.3 Grundfunktionen des TIA Portals

Im Folgenden erfahren Sie, wie Sie grundlegende Funktionen des TIA Portals ausschließlich mit der Tastatur nutzen können.

Grundfunktionen des TIA Portals mit der Tastatur nutzen

Die folgende Tabelle zeigt, wie sie mit Tastenkombinationen auf Grundfunktionen des TIA Portals zugreifen können:

Funktion	Tastenkombination	Menübefehl
Zwischen Projektansicht und Portalansicht wechseln	<Alt+F7>	
Zwischen Projektansicht und Bibliotheksansicht wechseln	<Alt+F5>	
Öffnet die Online- und Diagnosesicht	<Strg+D>	
Runtime starten	<Strg+Shift+Z>	
Querverweise anzeigen	<Shift+Alt+F11>	
Querverweis-Informationen	<Shift+F11>	
Hilfe öffnen Wenn Sie Hilfe zum TIA Portal benötigen, drücken Sie <F1>.	<F1>	Hilfe > Hilfe anzeigen
Aktuelle Aktion abbrechen	<Esc>	
Suchen	<Strg+F>	
Objekt ersetzen Bei einer Suche im Editor können Sie gefundene Objekte ersetzen.	<Strg+H>	
Weitersuchen Wenn Sie eine Suche im Editor gestartet haben, können Sie mit <F3> zur nächsten Fundstelle springen.	<F3>	
Objekt drucken	<Strg+P>	Projekt > Drucken
Standardaktion des Objekts ausführen	<Return>	
Aktualisieren	<F5>	
Horizontal nach rechts scrollen	<Strg+Pfeil-nach-rechts>	
Horizontal nach links scrollen	<Strg+Pfeil-nach-links>	
Baustein/PLC-Datentyp öffnen	<F7>	

Menüs bedienen

Die folgende Tabelle zeigt, wie Sie sich mithilfe der Tastatur durch Menüs bewegen können:

Funktion	Tastenkombination
Tastaturbedienung im Menü starten Das Menü erreichen Sie mit der <Alt>-Taste und können anschließend weiter mit den Pfeiltasten durch das Menü navigieren. Die Auswahl eines Menübefehls bestätigen Sie mit <Return>.	<Alt>
Direkt zu einem bestimmten Menü springen Sie können direkt zu einem einzelnen Menüeintrag springen, indem Sie die <Alt>-Taste gedrückt halten. Bei jedem Menüeintrag wird ein Buchstabe unterstrichen. Drücken Sie zusätzlich zur <Alt>-Taste den unterstrichenen Buchstaben.	<Alt+unterstrichener Buchstabe im jeweiligen Menü>
Kontextmenü eines Objekts öffnen Mit der Kontextmenütaste (auf Microsoft Windows kompatiblen Tastaturen) öffnen Sie das Kontextmenü des selektierten Objekts. Alternativ benutzen Sie <Shift+F10>, wenn Sie keine Microsoft Windows kompatible Tastatur verwenden. Sie können mit den Pfeiltasten durch das Kontextmenü navigieren und einen Menübefehl mit <Return> wählen.	<Kontextmenütaste> Alternativ: <Shift+F10>

Ausklappbare Elemente bedienen

Die folgende Tabelle zeigt, wie Sie ausklappbare Elemente mit der Tastatur bedienen können:

Funktion	Tastenkombination
Ordner in einer Baumstruktur aufklappen Mit <Pfeil-nach-rechts> klappen Sie beispielsweise einen Ordner in der Projektnavigation auf.	<Pfeil-nach-rechts>
Ordner in einer Baumstruktur schließen Mit <Pfeil-nach-links> klappen Sie beispielsweise einen Ordner in der Projektnavigation zu.	<Pfeil-nach-links>
Klappliste öffnen Klapplisten öffnen Sie mit <F4> und können anschließend mit den Pfeiltasten durch die Klappliste navigieren. Bestätigen Sie am Schluss Ihre Auswahl mit <Return>.	<F4>
Autovervollständigung öffnen	<Strg+Leertaste> <Strg+I>
Objektauswahl anzeigen	<Strg+J>

2.4.4 Nutzen projektbezogener Funktionen

Projekt bearbeiten

Funktion	Tastenkombination	Menübefehl
Projekt öffnen	<Strg+O>	Projekt > Öffnen
Projekt schließen	<Strg+W>	Projekt > Schließen
Projekt speichern	<Strg+S>	Projekt > Speichern
Projekt unter anderem Namen speichern	<Strg+Shift+S>	Projekt > Speichern unter
Projekt löschen	<Strg+E>	Projekt > Projekt löschen
Projekt drucken	<Strg+P>	Projekt > Drucken
Letzte Aktion rückgängig machen	<Strg+Z>	Bearbeiten> Rückgängig
Letzte Aktion wiederholen	<Strg+Y>	Bearbeiten > Wiederholen

Hilfe aufrufen

Funktion	Tastenkombination	Menübefehl
Hilfe aufrufen	<F1> oder <Shift+F1>	Hilfe > Hilfe anzeigen

2.4.5 Anordnung von Fenstern

Im Folgenden erfahren Sie, wie Sie mit der Tastatur einzelne Fenster des TIA Portals öffnen und schließen können oder wie Sie mit Hilfe der Tastatur mit gespeicherten Fensterlayouts arbeiten können.

Fenster öffnen und schließen

Die folgende Tabelle zeigt, wie Sie mit Tastenkombinationen einzelne Fenster öffnen und schließen können:

Funktion	Tastenkombination	Menübefehl
Projektnavigation öffnen/schließen	<Strg+1>	Ansicht > Projektnavigation
Detailansicht öffnen/schließen	<Strg+4>	Ansicht > Detailansicht
Übersicht öffnen/schließen	<Strg+2>	Ansicht > Übersicht
Task Card öffnen/schließen	<Strg+3>	Ansicht > Task Card
Bibliotheken öffnen	<Strg+Shift+L>	
Hardware-Katalog öffnen Wenn Sie sich in der Geräte- oder Netzsicht befinden, wird der Hardware-Katalog geöffnet.	<Strg+Shift+C>	
Inspektorfenster öffnen/schließen	<Strg+5>	Ansicht > Inspektorfenster
Register "Eigenschaften" im Inspektorfenster öffnen	<Strg+6>	
Register "Info" im Inspektorfenster öffnen	<Strg+7>	
Register "Diagnose" im Inspektorfenster öffnen	<Strg+8>	
Referenzprojekte anzeigen oder ausblenden	<Strg+9>	
Bildschirmtastatur anzeigen Sie können eine Tastatur am Bildschirm anzeigen, z. B. für die Verwendung an Touchscreens.	<Strg+Shift+K>	
Alle Editoren schließen	<Strg+Shift+F4>	Fenster > Alle schließen

Gespeicherte Fensterlayouts verwenden

Sie können sich individuelle Fensteranordnungen speichern und diese zu einem späteren Zeitpunkt wiederherstellen. Die folgende Tabelle zeigt, wie Sie mithilfe von Tastenkombinationen auf gespeicherte Fensterlayouts zugreifen:

Funktion	Tastenkombination	Menübefehl
Aktives Fensterlayout wiederherstellen Wenn Sie ein gespeichertes Fensterlayout verwenden und Sie haben zwischenzeitlich Änderungen an der Programmoberfläche vorgenommen, dann können Sie mit <Alt+Shift+0> den Ursprungszustand des aktiven Fensterlayouts wiederherstellen.	<Shift+Alt+0>	Fenster > Fensterlayout wiederherstellen
Fensterlayout laden Sie können mit <Alt+Shift+[Nummer des Fensterlayouts]> eines von den ersten fünf gespeicherten Fensterlayouts aktivieren.	<Shift+Alt+[Nummer des Fensterlayouts]>	Fenster > Fensterlayout 1 bis 5

2.4.6 Navigation durch die Programmoberfläche

Das TIA Portal untergliedert sich in verschiedene Oberflächenbereiche, wie z. B. einzelne Fenster, Funktionsleisten und Editoren. Wenn Sie innerhalb eines Oberflächenbereichs mit der Tastatur arbeiten möchten, müssen Sie diesen zunächst in den Fokus nehmen. Im Folgenden erfahren Sie, wie Sie mit der Tastatur einzelne Oberflächenbereiche fokussieren können. Außerdem erfahren Sie, wie Sie sich mit der Tastatur innerhalb eines Oberflächenbereichs im TIA Portal bewegen.

Wechseln zwischen Oberflächenbereichen und Editoren

Die folgende Tabelle zeigt, wie Sie zwischen einzelnen Oberflächenbereichen des TIA Portals wechseln:

Funktion	Tastenkombination
Wechsel zwischen den Oberflächenbereichen im Uhrzeigersinn Mit der Taste <F6> können Sie zwischen den einzelnen Oberflächenbereichen des TIA Portals im Uhrzeigersinn wechseln. Der Oberflächenbereich, der aktuell im Fokus steht, ist mit einer blauen Titelleiste gekennzeichnet. Befinden Sie sich beispielsweise in der Projektnavigation und drücken die Taste <F6>, dann springen Sie in den aktuell geöffneten Editor. Drücken Sie erneut <F6> und die Task Cards sind im Fokus. Drücken Sie hingegen <Shift+F6>, um entgegen dem Uhrzeigersinn zwischen den Arbeitsbereichen zu wechseln.	<F6>
Wechsel zwischen den Oberflächenbereichen entgegen dem Uhrzeigersinn Mit <Shift+F6> wechseln Sie entgegen dem Uhrzeigersinn zwischen den Oberflächenbereichen des TIA Portals.	<Shift+F6>
Wechsel zum nächsten geöffneten Editor Mit <Strg+Alt+Pfeil-nach-rechts> wechseln Sie zum nächsten geöffneten Editor. Die geöffneten Editoren sehen Sie in der Editorleiste.	<Strg+Alt+Pfeil-nach-rechts> Alternative: <Strg+F6>
Wechseln zum vorherigen geöffneten Editor Mit <Strg+Alt+Pfeil-nach-links> wechseln Sie zum zuletzt geöffneten Editor.	<Strg+Alt+Pfeil-nach-links> Alternative: <Strg+Shift+F6>
Zum übergeordneten Teil des Oberflächenbereichs springen Mit <Shift+Esc> springen Sie zum übergeordneten Teil der Programmoberfläche. Wenn Sie beispielsweise ein Gerät in der Projektnavigation selektiert haben und Sie drücken <Shift+Esc>, dann wird die gesamte Projektnavigation in den Fokus genommen.	<Shift+Esc> Alternativ: <Alt+Pfeil-nach-oben>
Zum untergeordneten Teil des Oberflächenbereichs springen Mit <Return> wird der untergeordnete Teil der Programmoberfläche in den Fokus genommen. Wenn Sie z. B. gerade die Eigenschaften eines Geräts im Inspektorfenster geöffnet haben, um das Gerät zu parametrieren, dann gelangen Sie mit <Return> eine Ebene tiefer in der Programmoberfläche. Im Anschluss können Sie mit der Tabulator-Taste bis zum gewünschten Parameter navigieren.	<Return> Alternativ: <Alt+Pfeil-nach-unten>

Navigation innerhalb von Oberflächenbereichen und Editoren

Die folgende Tabelle zeigt, wie Sie sich innerhalb eines Oberflächenbereichs mithilfe der Tastatur bewegen:

Funktion	Tastenkombination
Springen zum nächsten Element innerhalb eines Oberflächenbereichs Mit der Tabulator-Taste können Sie von einem zum nächsten Element innerhalb eines Arbeitsbereichs springen. Wenn Sie beispielsweise die Eigenschaften eines Geräts geöffnet haben und möchten von einem Eingabefeld zum nächsten springen, drücken Sie die Tabulator-Taste. Dabei werden eventuelle Änderungen am aktuellen Eingabefeld übernommen.	<Tab>
Springen zum vorherigen Element innerhalb eines Oberflächenbereichs Mit <Shift+Tab> können Sie zum voranstehenden Element innerhalb eines Arbeitsbereichs springen, beispielsweise zu einem vorhergehenden Textfeld. Dabei werden eventuelle Änderungen am aktuellen Eingabefeld übernommen.	<Shift+Tab>
Springen zum übergeordneten Element innerhalb eines Oberflächenbereichs Mit <Shift+Pos1> können Sie zum übergeordneten Element eines Arbeitsbereichs springen, z. B. zum übergeordneten Ordner in der Projektnavigation.	<Shift+Pos1>
Wechsel zum nächsten Register innerhalb eines Oberflächenbereichs Wenn ein Oberflächenbereich in einzelne Register unterteilt ist, können Sie zwischen den Registern mit der Tastenkombination <Strg+Tab> umschalten. Wenn Sie sich beispielsweise im Inspektorfenster im Register "Eigenschaften" befinden und Sie möchten in das Register "Info" springen, drücken Sie die Tastenkombination <Strg+Tab>.	<Strg+Tab>
Zum vorhergehenden Register wechseln Mit <Strg+Shift+Tab> wechseln Sie zum zuletzt geöffneten Register innerhalb eines Oberflächenbereichs.	<Strg+Shift+Tab>
Springen zur Funktionsleiste eines Editors Mit der Taste <Alt+F10> springen Sie zur Funktionsleiste eines Editors. Wenn Sie beispielsweise die Druckvorschau geöffnet haben und Sie möchten in der Funktionsleiste zur nächsten Seite des Ausdrucks wechseln, drücken Sie <Alt+F10>. Anschließend navigieren Sie mit den Pfeiltasten zum entsprechenden Symbol in der Funktionsleiste und bestätigen die Auswahl mit <Return>.	<Alt+F10>
Pfeile auf dem Fensterteiler bedienen, um Oberflächenelemente ein- oder auszublenden Die Tabelle im Arbeitsbereich ist minimierbar und maximierbar. Navigieren Sie zunächst in den Arbeitsbereich und nehmen Sie mithilfe der Tabulator-Taste einen der kleinen Pfeile auf der Trennlinie oberhalb der Tabelle in den Fokus. Die Pfeile liegen im Fokus, sobald diese blau hinterlegt sind. Anschließend drücken Sie die Leertaste, um die Tabelle zu minimieren oder zu maximieren.	<Leertaste>

2.4.7 Anpassen von Editoren

Im Folgenden erfahren Sie, wie Sie mithilfe der Tastatur die Anordnung der Editoren ändern können. Außerdem erfahren Sie, wie Sie die Darstellungsgröße und den Ausschnitt innerhalb eines grafischen Editors wählen können.

Anordnen und anpassen von Editoren

Die folgende Tabelle zeigt, wie Sie geöffnete Editoren untereinander oder nebeneinander anordnen können, oder wie Sie einen geöffneten Editor schließen können:

Funktion	Tastenkombination	Menübefehl
Aktiven Editor schließen	<Strg+F4>	
Editorbereich vertikal teilen Haben Sie beispielsweise das Übersichtsfenster und die Netzsicht geöffnet und möchten diese nebeneinander darstellen, drücken Sie die Taste <F12>.	<F12>	Fenster > Editorbereich vertikal teilen
Editorbereich horizontal teilen Sie können zwei geöffnete Editoren im Arbeitsbereich untereinander darstellen.	<Strg+F12>	Fenster > Editorbereich horizontal teilen
Fensterteilung aufheben Wenn Sie zwei Editoren im geteilten Modus untereinander oder nebeneinander im Arbeitsbereich anzeigen, können Sie die Teilung mit <Alt+Shift+F12> wieder aufheben.	<Alt+Shift+F12>	Fenster > Teilung aufheben

Darstellung in einem Editor anpassen

Die folgende Tabelle zeigt, wie Sie die Darstellung innerhalb grafischer Editoren vergrößern oder verkleinern können und wie Sie den Ausschnitt eines Editors verschieben:

Funktion	Tastenkombination
Darstellung in einem Editor stufenweise vergrößern Mit <Strg> und der Taste <Plus> auf dem Ziffernblock der Tastatur vergrößern Sie die Darstellung des Editors.	<Strg+Plus> Alternativ: <Strg+aufwärts scrollen>
Darstellung in einem Editor stufenweise verkleinern Mit <Strg> und der Taste <Minus> auf dem Ziffernblock der Tastatur verkleinern Sie die Darstellung des Editors.	<Strg+Minus> Alternativ: <Strg+abwärts scrollen>
100 %-Ansicht im Editor einstellen In einem grafischen Editor vergrößern oder verkleinern Sie die aktuelle Ansicht auf 100 %, wenn Sie <Strg+0> drücken.	<Strg+0>
Ausschnitt des Editors verschieben Wenn Sie die Leertaste gedrückt halten, können Sie den angezeigten Ausschnitt eines Editors mit der Maus verschieben.	<Leertaste>

2.4.8 Bearbeiten von Objekten

Objekte selektieren

Die folgende Tabelle zeigt, wie Sie einzelne Objekte, wie z. B. Geräte in der Projektnavigation, selektieren können:

Funktion	Tastenkombination	Menübefehl
Ein links, rechts, oberhalb oder unterhalb liegendes Objekt selektieren	<Pfeiltasten>	
Zum ersten Objekt innerhalb des aktuellen Oberflächenbereichs springen Das erste Objekt innerhalb des aktuell im Fokus liegenden Oberflächenbereichs wird selektiert. In der Projektnavigation wäre dies z. B. der Projektknoten ganz oben.	<Pos1>	
Zum letzten Objekt innerhalb des aktuellen Oberflächenbereichs springen Das letzte Objekt innerhalb des aktuell im Fokus liegenden Oberflächenbereichs wird selektiert, z. B. der letzte Eintrag in der Projektnavigation.	<Ende>	
Alle Objekte eines Bereichs selektieren Alle Objekte, die innerhalb des aktuell im Fokus liegenden Arbeitsbereichs liegen, werden selektiert.	<Strg+A>	Bearbeiten > Alles auswählen
Mehrere Objekte auswählen Wenn Sie mehrere Objekte selektieren möchten, die nicht unmittelbar nebeneinander liegen, müssen Sie zunächst mit <Strg+Pfeiltasten> den Fokus (graue Umrandung eines Objekts) auf das nächste, gewünschte Objekt verschieben. Die aktuelle Selektion bleibt dabei erhalten. Im Anschluss drücken Sie die Leertaste, um das neu fokussierte Objekt ebenfalls zu selektieren. Diesen Vorgang wiederholen Sie, bis alle gewünschten Objekte selektiert sind.	<Strg+Pfeiltasten> + <Leertaste>	

Objekte bearbeiten

Die folgende Tabelle zeigt eine Übersicht aller Tastenkombinationen, die für das Bearbeiten von Objekten notwendig sind:

Funktion	Tastenkombination	Menübefehl
Neues Objekt einfügen Abhängig vom Kontext, in dem Sie sich gerade befinden, wird ein neues Objekt eingefügt. Wenn Sie sich beispielsweise in der Gerätesicht befinden, wird der Dialog "Gerät hinzufügen" geöffnet, um ein neues Gerät anzulegen.	<Strg+N>	
Objekt öffnen	<Return>	
Objekt umbenennen	<F2>	Bearbeiten > Umbenennen
Objekt kopieren	<Strg+C> Alternativ: <Strg+Einfg>	Bearbeiten > Kopieren

Funktion	Tastenkombination	Menübefehl
Objekt ausschneiden	<Strg+X> Alternativ: <Shift+Entf>	Bearbeiten > Ausschneiden
Objekt einfügen	<Strg+V> Alternativ: <Shift+Einfg>	Bearbeiten > Einfügen
Objekt löschen	<Entf>	Bearbeiten > Löschen
Objekt übersetzen	<Strg+B>	Bearbeiten > Übersetzen
Eigenschaften eines Objekts öffnen Viele Objekte im TIA Portal verfügen über editierbare Eigenschaften. Drücken Sie die Tastenkombination <Alt+Enter>, um die Eigenschaften eines Objekts anzuzeigen.	<Alt+Return>	-

2.4.9 Textbearbeitung

Im Folgenden erfahren Sie, wie Sie Textbearbeitungsfunktionen ausschließlich mit der Tastatur nutzen können.

Texte editieren

Die folgende Tabelle zeigt grundlegende Editierfunktionen für Text:

Funktion	Tastenkombination
Einfüge- bzw. Überschreibmodus umschalten	<Einfg>
Editiermodus beenden	<Esc>
Löschen	<Entf>
Löschen von Zeichen	<Backspace>
Eingabe in einem Eingabefeld bestätigen und Eingabefeld verlassen	<Return>
Zeilenumbruch in einem mehrzeiligen Eingabefeld In einem mehrzeiligen Eingabefeld müssen Sie die <Shift>-Taste gedrückt halten, um einen Zeilenumbruch zu erzeugen.	<Shift+Return>
Eingaben in einem Eingabefeld zurücksetzen Wenn Sie sich in einem Eingabefeld befinden und Sie drücken <Esc>, dann verlassen Sie das Eingabefeld und durchgeführte Änderungen werden nicht übernommen.	<Esc>

Navigation innerhalb eines Textbereichs

Die folgende Tabelle zeigt, wie Sie mit der Tastatur innerhalb eines Textbereichs navigieren:

Funktion	Tastenkombination
Zum Zeilenanfang springen	<Pos1>
Zum Zeilenende springen	<Ende>
Zum Textanfang springen	<Strg+Pos1>
Zum Textende springen	<Strg+Ende>
Zur vorhergehenden Seite springen	<Bild Auf>
Zur nachfolgenden Seite springen	<Bild Ab>

Funktion	Tastenkombination
Eingabe in einem Eingabefeld bestätigen und Eingabefeld verlassen	<Return>
Zeilenumbruch in einem mehrzeiligen Eingabefeld	<Shift+Return>
Eingaben in einem Eingabefeld zurücksetzen Wenn Sie sich in einem Eingabefeld befinden und Sie drücken <Esc>, dann verlassen Sie das Eingabefeld und durchgeführte Änderungen werden nicht übernommen.	<Esc>

Text selektieren

Die folgende Tabelle zeigt, wie Sie mit der Tastatur Text selektieren:

Funktion	Tastenkombination
Auswahl auf das links oder rechts liegende Wort erweitern Der Text oder die bestehende Textauswahl wird bis zum Ende des Wortes ausgewählt. Wenn Sie sich am Anfang oder Ende eines Wortes befinden, wird das vorhergehende oder nachfolgende Wort ausgewählt.	<Strg+Shift+Pfeil-nach-links oder Pfeil-nach-rechts>
Auswahl bis zum Zeilenanfang erweitern	<Shift+Pos1>
Auswahl bis zum Zeilenende erweitern	<Shift+Ende>
Auswahl bis zum Textanfang erweitern Der Text wird bis zum Anfang oder bis zum Ende hin ausgewählt.	<Strg+Shift+Pos1>
Auswahl bis zum Textende erweitern Der Text wird bis zum Anfang oder bis zum Ende hin ausgewählt.	<Strg+Shift+Ende>

2.4.10 Bearbeiten von Tabellen

Im Folgenden erfahren Sie, wie Sie mit der Tastatur in Tabellen navigieren, einzelne Felder bearbeiten und Teile von Tabellen selektieren können.

Allgemeine Tastaturbedienung in Tabellen

Die folgende Tabelle zeigt, wie Sie Tabellen ausschließlich mithilfe der Tastatur bearbeiten können:

Funktion	Tastenkombination
Zelle in den Editiermodus setzen	<F2> oder <Return>
Eingabe bestätigen und Editiermodus beenden	<Return>
Editieren abbrechen und Änderungen verwerfen	<Esc>
Klappliste in einer Zelle öffnen Mit <F4> öffnen Sie die Klappliste. Mit den Pfeiltasten wählen Sie anschließend den gewünschten Eintrag und bestätigen die Auswahl mit <Return>.	<F4>
Klappliste in einer Zelle schließen und Änderungen verwerfen	<Esc>

In Tabellen navigieren

Die folgende Tabelle zeigt, wie Sie sich mithilfe der Tastatur innerhalb einer Tabelle bewegen:

Funktion	Tastenkombination
Gehe zu Zeile	<Strg+G>
Zur nächsten Zelle	<Pfeiltasten>
Zur nächsten editierbaren Zelle nach rechts	<Tab>
Zur nächsten editierbaren Zelle nach links	<Shift+Tab>
Eine Bildschirmseite nach oben	<Bild Auf>
Eine Bildschirmseite nach unten	<Bild Ab>
Zur ersten Zelle in der Zeile	<Pos1>
Zur letzten Zelle in der Zeile	<Ende>
Zur ersten Zelle der Tabelle	<Strg+Pos1>
Zur letzten Zelle der Tabelle	<Strg+Ende>
Zur obersten Zelle in der Spalte	<Strg+Pfeil-nach-oben>
Zur untersten Zelle in der Spalte	<Strg+Pfeil-nach-unten>
Tabellenzeile unterhalb einfügen	<Alt+Einfg>
Zeilenumbruch in einem Eingabefeld einfügen / Tabellenzeile oberhalb einfügen	<Strg+Return>

Bereiche in Tabellen selektieren

Die folgende Tabelle zeigt, wie Sie mit der Tastatur Bereiche innerhalb einer Tabelle selektieren können:

Funktion	Tastenkombination
Spalte selektieren	<Strg+Leertaste>
Zeile selektieren	<Shift+Leertaste>
Alle Zellen selektieren	<Strg+A>
Selektion um eine Zelle erweitern	<Shift+Pfeiltasten>
Selektion eine Seite nach oben erweitern	<Shift+Bild Auf>
Selektion eine Seite nach unten erweitern	<Shift+Bild Ab>
Selektion bis zur ersten Zeile erweitern	<Strg+Shift+Pfeil-nach-oben>
Selektion bis zur letzten Zeile erweitern	<Strg+Shift+Pfeil-nach-unten>
Selektion bis zur ersten Zelle in der Zeile erweitern	<Strg+Shift+Pfeil-nach-links>
Selektion bis zur letzten Zelle in der Zeile erweitern	<Strg+Shift+Pfeil-nach-rechts>

2.4.11 Nutzen von Online-Funktionen

Online-Funktionen mit der Tastatur steuern

Die folgende Tabelle zeigt eine Übersicht der Tastenkombinationen, mit denen Sie die Online-Funktionen des TIA Portals nutzen können:

Funktion	Tastenkombination	Menübefehl
Online-Verbindung herstellen	<Strg+K>	Online > Online verbinden
Online-Verbindung trennen	<Strg+M>	Online > Online-Verbindung trennen
Projektdaten in das Gerät laden	<Strg+L>	Online > Laden in Gerät
Erreichbare Teilnehmer anzeigen Es wird ein Dialog geöffnet, in dem alle Geräte angezeigt werden, die mit der PG/PC-Schnittstelle des PG/PCs verbunden sind.	<Strg+U>	Online > Erreichbare Teilnehmer anzeigen
CPU starten Die CPU wird in den Betriebszustand "RUN" versetzt. Die CPU muss dazu online verbunden sein.	<Strg+Shift+E>	Online > CPU starten
CPU stoppen Die CPU wird in den Betriebszustand "STOP" versetzt. Die CPU muss dazu online verbunden sein.	<Strg+Shift+Q>	Online > CPU stoppen
Simulation starten Die Hardware und Software des Projektes kann in einer simulierten Online-Umgebung getestet werden, ohne dass die Baugruppen tatsächlich online verbunden sein müssen.	<Strg+Shift+X>	Online > Simulation > Starten

2.4.12 Bildschirmtastatur verwenden

Einführung

Für die Bedienung des TIA Portals steht Ihnen zusätzlich die Microsoft Bildschirmtastatur zur Verfügung.

Bildschirmtastatur anzeigen

Um die Bildschirmtastatur anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Ansicht" den Befehl "Bildschirmtastatur".

Bildschirmtastatur beenden

Um die Bildschirmtastatur zu beenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Datei" der Bildschirmtastatur den Befehl "Beenden".

2.4.13 Benutzerdefinierte Tastenkombinationen verwenden

2.4.13.1 Benutzerdefinierte Tastenkombinationen zuweisen

Sie können eigene Tastenkombinationen zur Bedienung des TIA Portals einstellen.

Vorgehen

Um benutzerdefinierte Tastenkombinationen zuzuweisen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".
   Sie sehen eine Tabelle mit einer Übersicht aller Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.
3. Klicken Sie auf eine Reihe in der Tabelle.
   In der Zelle "Benutzerdefinierte Tastenkombinationen" erscheint eine Schaltfläche.
4. Klicken Sie auf die Schaltfläche.
   Der Dialog "Benutzerdefinierte Tastenkombination bearbeiten" öffnet sich. Sie sehen im oberen Teil des Dialogs die aktuell ausgewählte Aktion, den Editor und die aktuell zugewiesene Tastenkombination.
5. Wählen Sie die Option "Neue Tastenkombination zuweisen" aus.
   Die Option ist voreingestellt.
6. Öffnen Sie die Klappliste unter der Option "Neue Tastenkombination zuweisen".
   Die verfügbaren, nicht im TIA Portal verwendeten Tastenkombinationen werden angezeigt.
7. Wählen Sie eine Tastenkombination aus.
8. Um Ihre Auswahl zu übernehmen und den Dialog zu schließen, klicken Sie auf "OK".
9. Damit die zugewiesene Tastenkombination angewendet wird, starten Sie das TIA Portal neu.
   Die zugewiesene Tastenkombination ist für neue benutzerdefinierte Tastenkombinationen nicht mehr verfügbar. Die vorher genutzte Tastenkombination ist für neue benutzerdefinierte Tastenkombinationen verfügbar.
   Benutzerdefinierte Tastenkombinationen überschreiben voreingestellte Tastenkombinationen.

2.4.13.2 Benutzerdefinierte Tastenkombinationen löschen

Wenn Sie für Funktionen im TIA Portal benutzerdefinierte Tastenkombinationen zugewiesen haben, können Sie diese Tastenkombinationen wieder löschen.

Vorgehen

Um benutzerdefinierte Tastenkombinationen zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".
   Sie sehen eine Tabelle mit einer Übersicht aller Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.
3. Klicken Sie auf eine Reihe in der Tabelle.
   In der Zelle "Benutzerdefinierte Tastenkombinationen" erscheint eine Schaltfläche.
4. Klicken Sie auf die Schaltfläche.
   Der Dialog "Benutzerdefinierte Tastenkombination bearbeiten" öffnet sich. Sie sehen im oberen Teil des Dialogs die aktuell ausgewählte Aktion, den Editor und die aktuell zugewiesene Tastenkombination.
5. Wählen Sie die Option "Tastenkombination entfernen" aus.
6. Um Ihre Auswahl zu übernehmen und den Dialog zu schließen, klicken Sie auf "OK".
7. Damit das Löschen der Tastenkombination angewendet wird, starten Sie das TIA Portal neu.
   Die gelöschte Tastenkombination ist für neue benutzerdefinierte Tastenkombinationen wieder verfügbar.

2.4.13.3 Benutzerdefinierte Tastenkombinationen auf Standardeinstellung zurücksetzen

Sie haben die Möglichkeit, alle benutzerdefinierten Tastenkombinationen zu verwerfen und auf die Standardeinstellung zurückzusetzen.

Vorgehen

Um benutzerdefinierte Tastenkombinationen auf die Standardeinstellung zurückzusetzen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".
   Sie sehen eine Tabelle mit einer Übersicht aller Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.
3. Klicken Sie rechts unterhalb der Übersicht der Tastenkombinationen auf die Schaltfläche "Auf Standardeinstellung zurücksetzen".
   Ein Bestätigungsdialog wird angezeigt.
   Die Tastenkombinationen werden auf die Standardeinstellung zurückgesetzt.

2.4.13.4 Benutzerdefinierte Tastenkombinationen importieren und exportieren

Um Ihre benutzerdefinierten Tastenkombinationen in einer anderen Installation oder in einem anderen Produkt zu verwenden, können Sie diese exportieren und importieren.

Benutzerdefinierte Tastenkombinationen importieren

Um benutzerdefinierte Tastenkombinationen zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen importieren".
4. Wählen Sie die Einstellungsdatei mit der Endung ".tps15" aus.
5. Klicken Sie auf "Öffnen".
   Nach einem Neustart des TIA Portals werden die Einstellungen übernommen.

Benutzerdefinierte Tastenkombinationen exportieren

Um benutzerdefinierte Tastenkombinationen zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen exportieren".
   Der Dialog "Exporteinstellungen" wird geöffnet.
4. Aktivieren Sie das Optionskästchen "Tastaturbedienung".
5. Tragen Sie einen Dateinamen für die Einstellungsdatei in das Feld "Dateiname" ein.
6. Klicken Sie auf "Export".
   Die Einstellungen werden in eine Datei mit der Endung ".tps15" exportiert.

2.4.13.5 Liste der Tastenkombinationen auswählen

Sie können entweder Ihre benutzerdefinierten Tastenkombinationen oder die systemdefinierten Tastenkombinationen verwenden. Dazu können Sie eine Auswahl in den Einstellungen treffen.

Vorgehen

Um eine Liste für Tastenkombinationen auszuwählen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".
   Sie sehen eine Tabelle mit einer Übersicht aller Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.
3. Wählen Sie in der Auswahlliste oberhalb der Übersicht der Tastenkombinationen eine der folgenden Listen aus:
   - Benutzerdefinierte Tastenkombinationen
   - Systemdefinierte Tastenkombinationen

Hinweis

Zuweisung von benutzerdefinierten Tastenkombinationen

Wenn die Liste "Systemdefinierte Tastenkombinationen" ausgewählt ist, können keine benutzerdefinierten Tastenkombinationen zugewiesen werden.

2.5 Betriebssystemabhängige Besonderheiten

2.5.1 Einfluss von Benutzerrechten

Einschränkungen bei limitierten Benutzerrechten

Die Software bietet einige Funktionen, welche direkten Zugriff auf die Hardware des PG/PC und damit auf das installierte Betriebssystem erfordern. Um den vollen Funktionsumfang nutzen zu können, muss die Software eng mit dem Betriebssystem zusammenarbeiten. Für eine reibungslose Interaktion sollten Sie daher mit ausreichenden Benutzerrechten am Betriebssystem angemeldet sein.

Insbesondere Funktionen, die eine Online-Verbindung erfordern oder die Einstellungen von Schnittstellenkarten ändern, sind möglicherweise nicht vollständig nutzbar, wenn Sie mit eingeschränkten Benutzerrechten arbeiten.

Erkennen von eingeschränkten Funktionen

Funktionen, die spezielle Rechte erfordern, können Sie folgendermaßen erkennen:

- Ein Schildsymbol wird neben der Funktion angezeigt.
  Die Funktion ist bedienbar, wird allerdings durch die Benutzerkontensteuerung reglementiert.
- Ein Feld ist gegraut und nicht bedienbar.
  Für das Feld benötigen Sie zwingend Administrationsrechte. In manchen Betriebssystemumgebungen können Sie kurzfristig Administrationsrechte durch die Eingabe eines Administratorkennworts erlangen.

Hinweis

Ist ein Feld gegraut, ist dies nicht in jedem Fall durch fehlende Rechte bedingt. Beachten Sie zusätzliche Informationen in den Tooltip-Kaskaden, um die Bedingungen für das Bearbeiten zu erfahren.

2.5.2 Benutzerrechte erweitern

Beschränkungen durch Benutzerrechte aufheben

Bestimmte Funktionen sind möglicherweise nicht bedienbar, wenn Sie nicht mit ausreichenden Benutzerrechten am Betriebssystem angemeldet sind. Um etwaige Beschränkungen aufzuheben, gibt es die folgenden Möglichkeiten:

- Freigeben von erweiterten Rechten mithilfe der Windows-Benutzerkontensteuerung
- Anmelden am Betriebssystem mit Administrationsrechten
- Nutzen vorübergehender Administrationsrechte

Erweiterte Rechte mithilfe der Windows-Benutzerkontensteuerung freigeben

Um eine Funktion zu bedienen, die mit dem Schild-Symbol der Windows-Benutzerkontensteuerung versehen ist, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Feld oder die Schaltfläche mit dem Schild-Symbol.
   Die Sicherheitsabfrage der Windows-Benutzerkontensteuerung wird geöffnet.
2. Folgen Sie den Anweisungen der Windows-Benutzerkontensteuerung und geben Sie auf Nachfrage gegebenenfalls ein Administratorkennwort ein.

Die Funktion ist nun einmalig ohne Einschränkungen nutzbar.

Am Betriebssystem mit Administrationsrechten anmelden

Um eine Funktion zu benutzen, die aufgrund fehlender Benutzerrechte deaktiviert ist, gehen Sie folgendermaßen vor:

1. Schließen Sie die Software.
2. Melden Sie sich am Betriebssystem ab.
3. Melden Sie sich mit Administrationsrechten am Betriebssystem an.
4. Starten Sie die Software erneut.

Nutzen vorübergehender Administrationsrechte

Um vorübergehend Administrationsrechte zu erhalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Einstellungen ändern". Die Schaltfläche ist in Dialogen zu finden, die das vorübergehende Zuweisen von Administrationsrechten erlauben.
   Ein Dialog des Betriebssystems zur Eingabe eines Administratorkennworts wird geöffnet.
2. Geben Sie ein Administratorkennwort ein.

Die Einstellungen können vorübergehend geändert werden. Bei erneutem Aufruf des Dialogs müssen Sie das Vorgehen wiederholen.

Hinweis

Diese Funktion wird nicht von jedem Betriebssystem unterstützt. Wenn keine Schaltfläche "Einstellungen ändern" vorhanden ist oder wenn die Schaltfläche gegraut ist, dann müssen Sie sich stattdessen am Betriebssystem mit Administrationsrechten anmelden.

Hilfe zum Informationssystem 3

3.1 Allgemeines zum Informationssystem

Schnelle Antworten auf Ihre Fragen

Das Informationssystem des TIA Portals unterstützt Sie bei der Lösung Ihrer Aufgaben und bietet Ihnen bei jedem Projektierungsschritt die benötigten Hilfethemen an.

Während der Arbeit mit dem Programm erhalten Sie folgende Unterstützung:

- Informationssystem mit allen Hintergrundinformationen, Schritt-für-Schritt-Anleitungen und Beispielen, die Sie beim Arbeiten mit dem TIA Portal benötigen.
- Tooltips für die Informationen zu Oberflächenelementen, wie z. B. Eingabefeldern, Schaltflächen und Symbolen. Tooltips sind zum Teil durch Kaskaden mit genaueren Informationen ergänzt.
- Hilfe zum aktuellen Kontext, z. B. zu Menübefehlen über die Taste <F1>.
- Hilfe zu Meldungen
- Roll-outs für korrekte Eingaben in Dialogfelder
- Banner an der Software-Oberfläche

Informationssystem

Das Informationssystem wird in einem eigenen Fenster geöffnet. Das folgende Bild zeigt das Informationssystem zum TIA Portal:

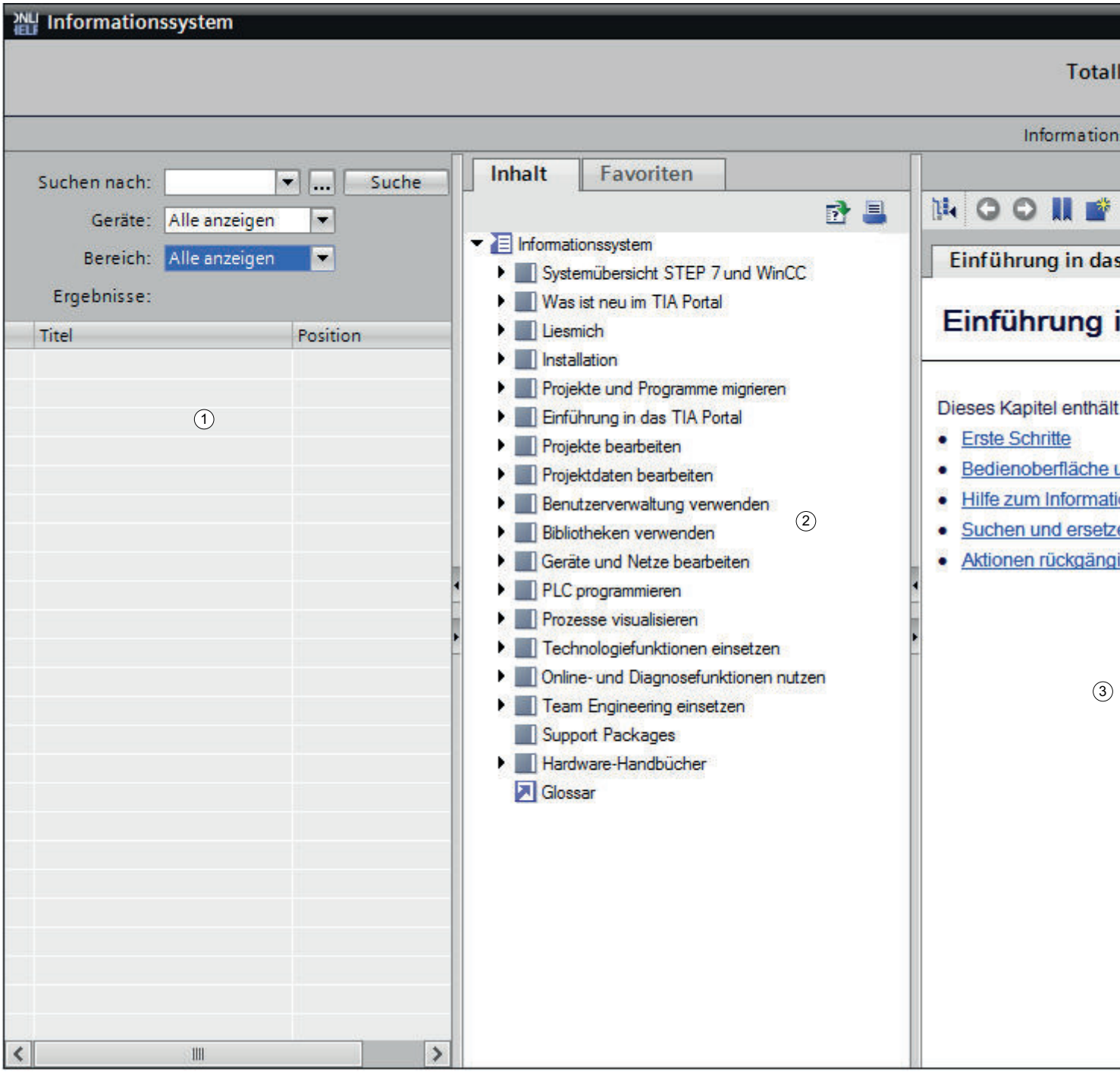

① Suchbereich

Im Suchbereich können Sie eine Volltextsuche über alle Hilfethemen durchführen.

② Navigationsbereich
Im Navigationsbereich finden Sie das Inhaltsverzeichnis und die Favoriten.

③ Inhaltsbereich
Im Inhaltsbereich werden die Hilfeseiten angezeigt. Sie können mehrere Register öffnen, um verschiedene Hilfeseiten gleichzeitig anzuzeigen.

Mit den Pfeilen auf den Fensterteilern lassen sich die einzelnen Bereiche ein- und ausblenden. So können Sie sowohl den Suchbereich als auch den Navigationsbereich zuklappen, um den Inhaltsbereich bei Bedarf zu vergrößern.

Schaltflächen im Navigationsbereich

Im Navigationsbereich des Informationssystems stehen folgende Schaltflächen zur Verfügung:

Schaltfläche	Funktion
	Nach Updates suchen Startet die Suche nach Hardware-Handbüchern, die als Update zur Verfügung stehen.
	Drucken Druckt auszuwählende Seiten oder Kapitel des Informationssystems aus.

Schaltflächen im Inhaltsbereich

Im Inhaltsbereich des Informationssystems stehen folgende Schaltflächen zur Verfügung:

Schaltfläche	Funktion
	Synchronisieren Zeigt die Position des geöffneten Hilfethemas im Inhaltsverzeichnis an.
	Vorwärts / Rückwärts Navigiert durch den Verlauf der bisher in diesem Register geöffneten Hilfethemen.
	Favoriten Fügt das aktuelle Hilfethema zu den Favoriten hinzu.
	Neues Register Öffnet ein neues, leeres Register.
	Startseite Zeigt die Startseite des Informationssystems an.
	Drucken Druckt das aktuelle Hilfethema aus.
	Hilfe Zeigt die Hilfe zum Informationssystem an.

Kennzeichnung der Themen nach dem Informationstyp

Im Informationssystem finden Sie ein breites Informationsangebot. Die Hilfethemen sind abhängig vom Informationstyp mit unterschiedlichen Symbolen gekennzeichnet. Anhand der Symbole können Sie auf einen Blick erkennen, welche Art von Information Sie gerade geöffnet haben.

Symbol	Informationstyp	Erklärung
	Handlungsanweisung	Beschreibt, wie Sie schrittweise vorgehen, um eine bestimmte Aufgabe durchzuführen.
	Beispiel	Enthält ein praxisorientiertes Anwendungsbeispiel, das die Lösung einer Automatisierungsaufgabe erläutert.
	Sachverhalt	Enthält Hintergrundinformationen zu den Funktionen des TIA Portals und vermittelt Grundlagenwissen.
	Referenz	Enthält detaillierte Referenzinformationen zu Anweisungen und Objekten zum Nachschlagen.

Kennzeichnung der Gültigkeit von Hilfethemen

Das Informationssystem beschreibt den vollen Funktionsumfang des installierten Produkts. Der für Sie nutzbare Funktionsumfang hängt jedoch von der erworbenen Lizenz und den eingesetzten Geräten ab.

Aus diesem Grund sind Hilfethemen, die nur für bestimmte Geräte oder Zielsysteme gültig sind, folgendermaßen gekennzeichnet:

- Angabe der Gültigkeit innerhalb des Inhaltsverzeichnisses
- Angabe der Gültigkeit in den Suchergebnissen in der Volltextsuche
- Angabe der Gültigkeit in der Titelleiste eines Kapitels oder Hilfethemas. Die Titelleiste enthält ein Symbol, dessen Tooltip die Gültigkeit angibt.

Das folgende Bild zeigt die Anzeige der Gültigkeiten in der Titelleiste eines Hilfethemas:

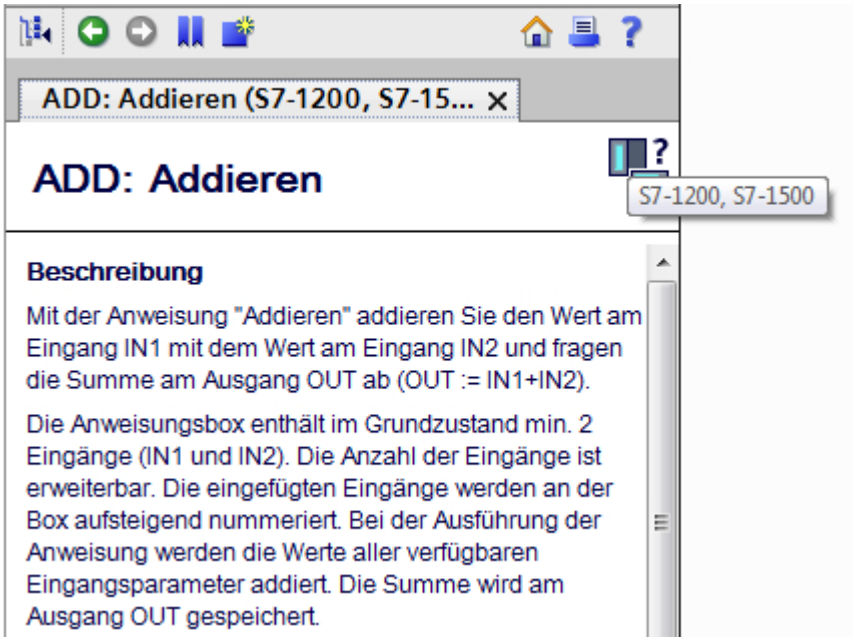

Darüber hinaus können Sie den Suchfilter "Geräte" verwenden, um die Suche auf ein bestimmtes Gerät oder eine Gerätefamilie zu beschränken.

Siehe auch: Hilfethemen suchen (Seite 90)

Roll-out

Einige Eingabefelder bieten eine Information, die "herausrollt" und Ihnen hilft, gültige Parameter und Werte einzugeben. Das Roll-out informiert Sie über die zulässigen Wertebereiche und Datentypen der Eingabefelder.

Das folgende Bild zeigt ein Roll-out (gelb) und eine Roll-out-Fehlermeldung (rot), die Sie auf einen ungültigen Wert hinweist:

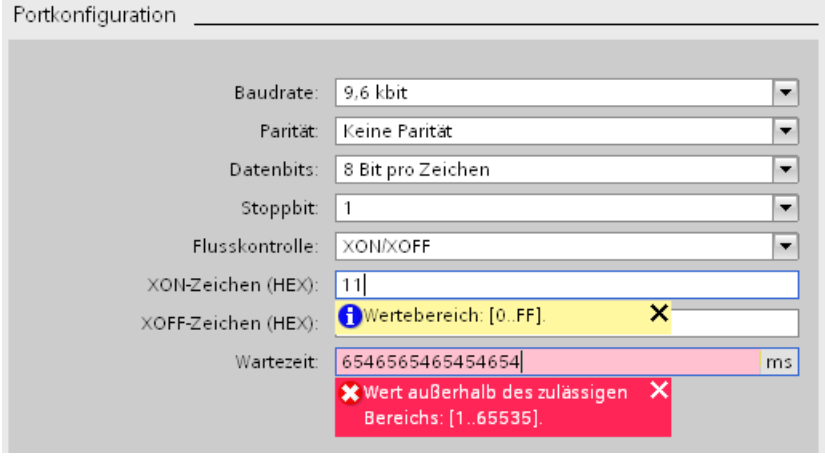

Der Roll-out wird geschlossen, sobald Sie das Feld verlassen oder auf das Kreuz in der rechten oberen Ecke klicken.

Tooltip

Oberflächenelemente bieten Ihnen Kurzinformationen in Form eines Tooltips an.

Tooltips, die links ein Pfeilsymbol haben, enthalten noch zusätzliche Informationen in Tooltip-Kaskaden. Verweilen Sie etwas länger mit dem Mauszeiger über dem Tooltip oder klicken Sie auf das Pfeilsymbol, so wird diese Information angezeigt. Die automatische Anzeige von Tooltip-Kaskaden kann deaktiviert werden.

Sind weiterführende Informationen vorhanden, erscheint ein Link zum entsprechenden Hilfethema in der Kaskade. Wenn Sie auf den Link klicken, dann wird das passende Thema im Informationssystem geöffnet.

Das folgende Bild zeigt einen Tooltip mit geöffneter Kaskade:

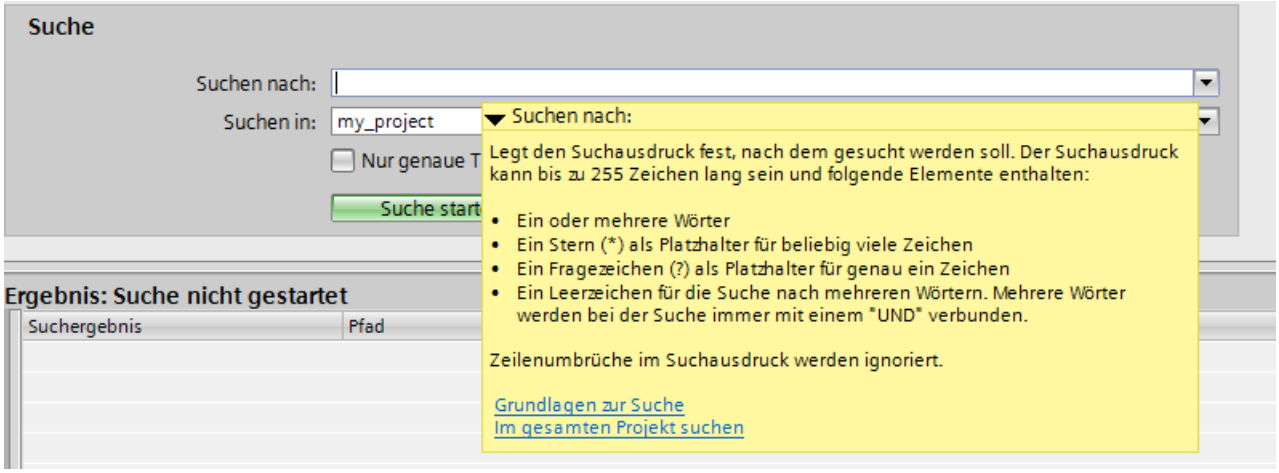

Hilfe zu Meldungen

Zahlreiche Aktionen werden im TIA Portal von Meldungen im Inspektorfenster begleitet. Die Meldungen geben Aufschluss darüber, ob eine Aktion erfolgreich war oder nicht. Außerdem sehen Sie, welche Änderungen im Projekt vorgenommen wurden. Zu einigen Meldungen existiert weiterführende Hilfe. Wenn weiterführende Hilfe zu einer Meldung vorhanden ist, dann erreichen Sie die Hilfe mit einem Klick auf das Fragezeichensymbol.

Das folgende Bild zeigt das Register "Info" im Inspektorfenster mit einigen Meldungen und einem Fragezeichensymbol:

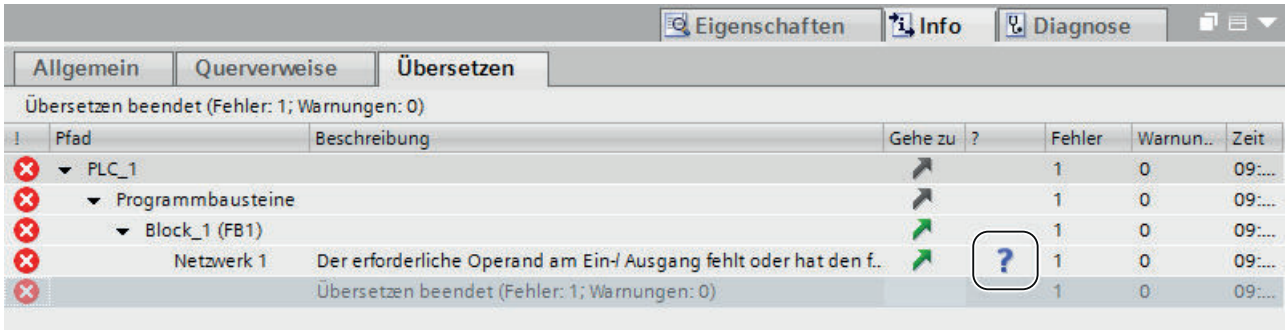

Banner an der Software-Oberfläche

In manchen Editoren erhalten Sie zusätzliche Unterstützung durch Banner am oberen Rand des Editors. Das folgende Bild zeigt einen Banner bei einem geöffneten Bibliotheks-Typen:

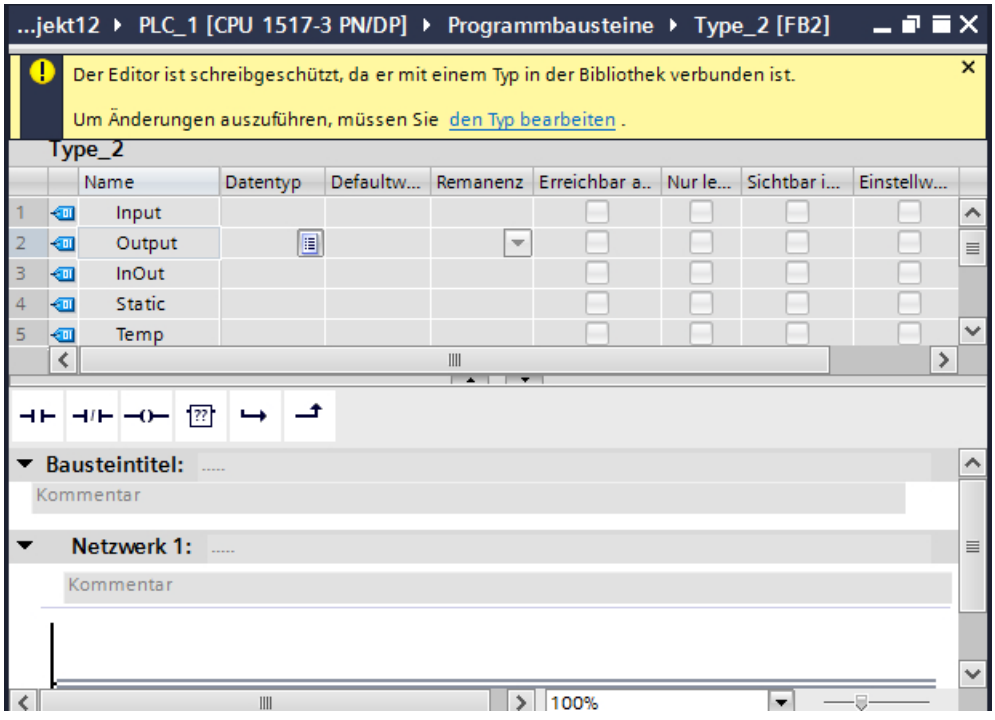

Um den Banner auszublenden, klicken Sie auf die senkrechte blaue Leiste am linken Rand des Banners oder auf das Kreuz in der rechten oberen Ecke. Um einen ausgeblendeten Banner einzublenden, klicken Sie auf das Hinweissysmbol am linken oberen Rand des geöffneten Editors. In den Einstellungen des TIA Portals können Sie einstellen, dass Banner standardmäßig ausgeblendet angezeigt werden.

Wenn in einem Editor mehrere Banner verfügbar sind, wird der relevanteste Banner zuerst angezeigt. Um zum nächsten Banner zu gelangen, blättern Sie in der rechten unteren Ecke weiter zum nächsten Banner.

Siehe auch

Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren (Seite 102)

3.2 Informationssystem öffnen

Informationssystem über das Menü öffnen

Um das Informationssystem auf der Startseite zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Hilfe" den Befehl "Hilfe anzeigen".

Die Startseite des Informationssystems wird geöffnet.

Informationssystem mit <F1> öffnen

Um das Informationssystem zu öffnen und Hilfe zum aktuellen Kontext anzuzeigen, gehen Sie folgendermaßen vor:

1. Markieren Sie ein Objekt, zu dem Sie Hilfe anzeigen möchten, z. B. einen Menübefehl oder ein Programmelement.
2. Drücken Sie <F1>.

Das Informationssystem wird geöffnet. Falls Informationen zum aktuellen Kontext vorhanden sind, wird das passende Thema angezeigt. Wenn keine Informationen zum aktuellen Kontext vorhanden sind, wird die Startseite des Informationssystems angezeigt.

Informationssystem über Tooltip-Kaskaden öffnen

Um das Informationssystem aus einer Tooltip-Kaskade heraus zu öffnen, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger über ein Objekt mit einer Tooltip-Kaskade.
   Die Tooltip-Kaskade wird geöffnet. Wenn weiterführende Informationen vorhanden sind, enthält die Tooltip-Kaskade Links zu den entsprechenden Hilfethemen.
2. Klicken Sie auf einen Link.

Das Informationssystem wird geöffnet und das weiterführende Hilfethema wird angezeigt.

Hilfe zu Fehlermeldungen öffnen

Zahlreiche Aktionen werden im TIA Portal von Meldungen im Inspektorfenster begleitet. Wenn Hilfe zu einer solchen Meldung vorhanden ist, wird hinter der Meldung ein blaues Fragezeichen angezeigt.

Um Hilfe zu einer Meldung anzuzeigen, gehen Sie folgendermaßen vor:

1. Zeigen Sie das Inspektorfenster an.
2. Klicken Sie auf das blaue Fragezeichen hinter einer Meldung.

Das Informationssystem wird geöffnet und die Hilfe zur Fehlermeldung wird angezeigt. Sie enthält eine genaue Beschreibung der Fehlerursache und Hinweise zur Fehlerbehebung.

Siehe auch

Informationssystem verwenden (Seite 87)

3.3 Informationssystem verwenden

Hilfethema im aktuellen Register öffnen

Um ein Hilfethema im aktuellen Register zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Hilfethema im Inhaltsverzeichnis.

Oder:

1. Doppelklicken Sie auf das Hilfethema in der Liste der Suchergebnisse.

Hilfethema in einem neuen Register öffnen

Um ein Hilfethema in einem neuen Register zu öffnen, gehen Sie folgendermaßen vor:

1. Markieren Sie das Hilfethema im Inhaltsverzeichnis oder in der Liste der Suchergebnisse.
2. Wählen Sie im Kontextmenü den Befehl "In neuer Registerkarte öffnen".

Oder:

1. Klicken Sie im Inhaltsbereich des Informationssystems auf die Schaltfläche "Neues Register".
   Ein neues, leeres Register wird geöffnet.
2. Klicken Sie auf das Hilfethema im Inhaltsverzeichnis oder doppelklicken Sie auf das Hilfethema in der Liste der Suchergebnisse.

Register schließen

Um ein Register zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Kreuz am rechten Rand des Registers.
   Das aktuelle Register wird geschlossen.

Oder:

1. Klicken Sie auf das Register.
2. Wählen Sie im Kontextmenü den Befehl "Andere Registerkarten schließen".
   Alle Register mit Ausnahme des aktuellen Registers werden geschlossen.

Verlauf anzeigen

Der Verlauf der bisher geöffneten Hilfethemen wird für jedes Register getrennt gespeichert.
Um ein Hilfethema aus dem Verlauf anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Inhaltsbereich des Informationssystems auf die Schaltflächen "Vorwärts" oder "Rückwärts", um chronologisch durch den Verlauf zu blättern.

Position eines Hilfethemas im Inhaltsverzeichnis anzeigen

Um die Position des geöffneten Hilfethemas im Inhaltsverzeichnis anzuzeigen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in dem Hilfethema.
2. Klicken Sie im Inhaltsbereich des Informationssystems auf die Schaltfläche "Synchronisieren".

Hilfe zum Informationssystem anzeigen

Um die Hilfe zum Informationssystem anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Inhaltsbereich des Informationssystems auf die Schaltfläche "Hilfe".

3.4 Informationssystem durchsuchen

3.4.1 Grundlagen zur Suche

Einführung

Sie haben die Möglichkeit, das Informationssystem nach Begriffen zu durchsuchen. Dazu steht Ihnen ein Suchbereich zur Verfügung.

Suchoptionen

Sie können nach den folgenden Texten suchen:

- Suche nach einem Wort:
  `<Suchbegriff>`
  Es werden alle Hilfeseiten gefunden, die den Suchbegriff entweder exakt oder leicht abgewandelt enthalten.
- Suche nach mehreren Wörtern:
  `<Suchbegriff_1 Suchbegriff_2 ... Suchbegriff_n>`
  Es werden alle Hilfeseiten gefunden, die alle Suchbegriffe entweder exakt oder leicht abgewandelt enthalten. Dabei spielt weder die Reihenfolge, noch der Abstand der gesuchten Wörter voneinander eine Rolle.
- Suche nach Phrasen
  `<"Suchbegriff_1 Suchbegriff_2 ... Suchbegriff_n">`
  Es werden alle Hilfeseiten gefunden, in denen alle Suchbegriffe exakt in der eingegebenen Reihenfolge enthalten sind. Setzen Sie für eine Phrasensuche Ihren Suchausdruck in Anführungszeichen.

Sie können Ihre Suche mit zusätzlichen Kriterien weiter spezifizieren:

- Boolesche Suche mit "AND":
  `<Suchbegriff AND Suchbegriff>`
  `<Suchbegriff + Suchbegriff>`
  Sie können Ihre Suchbegriffe mit dem booleschen Operator "AND" verknüpfen. Es werden dann alle Hilfeseiten gefunden, in denen alle Suchbegriffe entweder exakt oder leicht abgewandelt enthalten sind.
  Alternativ können Sie mehrere Suchbegriffe auch ohne die Operatoren "AND" oder "+" eingeben. Auch dann werden alle Hilfeseiten gefunden, die alle Suchbegriffe enthalten.
- Boolesche Suche mit "OR":
  `<Suchbegriff OR Suchbegriff>`
  `<Suchbegriff - Suchbegriff>`
  Sie können Ihre Suchbegriffe mit dem booleschen Operator "OR" verknüpfen. Es werden dann alle Hilfeseiten gefunden, in denen einer der Suchbegriffe entweder exakt oder leicht abgewandelt enthalten ist.

- Suche negieren mit "NOT"
  `<Suchbegriff_1 NOT Suchbegriff_2>`
  `<Suchbegriff_1 ! Suchbegriff_2>`
  Sie können einen Suchbegriff mit "NOT" negieren. Dazu ist jedoch die Eingabe von mindestens einem weiteren Suchbegriff erforderlich. Es werden dann alle Hilfeseiten gefunden, in denen "Suchbegriff_1" vorkommt. Seiten, in denen "Suchbegriff_2" vorkommt, werden nicht als Treffer aufgenommen, da dieser Suchbegriff negiert wurde.
- Unscharfe Suche mit "~"
  `<Suchbegriff~>`
  Sie können Ihrem Suchbegriff eine Tilde ("~") nachstellen. Dadurch werden alle Hilfeseiten gefunden, in denen der Suchbegriff in ungefährer Schreibweise vorkommt.
- Abstandssuche mit "~<Abstand>"
  `<"Suchbegriff_1 Suchbegriff_2" ~Abstand>`
  Die Abstandssuche ermöglicht Ihnen Hilfeseiten zu finden, in denen zwei Suchbegriffe nicht weiter als der definierte Abstand voneinander entfernt stehen. Geben Sie als Abstand eine positive Ganzzahl ein. Setzen Sie die beiden Suchbegriffe dazu in Anführungszeichen. Zwischen der Tilde ("~") und dem Abstand darf kein Leerzeichen stehen.

Suchfilter

Sie können die Suche beschränken, indem Sie Suchfilter angeben:

- Wenn Sie ein Gerät als Filterkriterium auswählen, wird der Suchbegriff nur in den Seiten oder Kapiteln gesucht, die für das angegebene Gerät gültig sind.
- Wenn Sie einen Bereich als Filterkriterium auswählen, wird der Suchbegriff entweder nur in der Software-Dokumentation oder nur in den integrierten Hardware-Handbüchern gesucht.

Die beiden Suchfilter können auch kombiniert werden.

Erweiterte Suche

Sie können auch die erweiterte Suche nutzen, um das Informationssystem zu durchsuchen. Sie bietet die gleichen Suchmöglichkeiten wie die Standard-Suchfunktion. Sie erhalten jedoch zusätzlich direkte Informationen zu den Eingabemöglichkeiten der Suchoptionen.

Siehe auch

Hilfethemen suchen (Seite 90)

Beispiele zur Suche im Informationssystem (Seite 92)

3.4.2 Hilfethemen suchen

Um die benötigten Informationen schnell zu finden, können Sie eine Volltextsuche über alle Hilfethemen durchführen.

Vorgehen

Um das Informationssystem nach bestimmten Begriffen zu durchsuchen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Suchbereich.
2. Geben Sie im Eingabefeld "Suchen nach" den Suchbegriff ein oder wählen Sie einen kürzlich gesuchten Begriff aus der Klappliste aus.
3. Optional: Um Informationen zu den Eingabemöglichkeiten von Suchbegriffen zu erhalten, klicken Sie auf die Schaltfläche "...".
   Der Dialog "Erweiterte Suche" wird geöffnet. Er zeigt eine Übersicht aller Optionen.
4. Optional: Wählen Sie im Auswahlfeld "Geräte" ein Gerät als Suchfilter aus.
5. Optional: Wählen Sie im Auswahlfeld "Bereich" den Bereich "Software-Dokumentation" oder "Hardware-Handbücher" aus.
6. Klicken Sie auf "Suchen".
   Die Suche wird durchgeführt und die Ergebnisse werden aufgelistet:
   - In der Spalte "Titel" wird der Titel des gefundenen Hilfethemas angezeigt.
   - In der Spalte "Position" wird das Kapitel angezeigt, in dem das Hilfethema enthalten ist. Kapitel aus Hardware-Handbüchern sind mit dem Präfix "HW" gekennzeichnet.
   - In der Spalte "Rang" wird die Relevanz des Hilfethemas bezüglich des Suchbegriffs angezeigt.

   Sie können die Sortierung ändern, indem Sie auf die Kopfzeile einer Spalte klicken.
7. Doppeklicken Sie auf ein Suchergebnis oder wählen Sie im Kontextmenü den Befehl "In neuer Registerkarte öffnen".

Ergebnis

Das Hilfethema wird im Inhaltsbereich geöffnet.

Um die Position des geöffneten Hilfethemas im Inhaltsverzeichnis anzuzeigen, klicken Sie auf die Schaltfläche "Synchronisieren".

Siehe auch

Grundlagen zur Suche (Seite 89)

Beispiele zur Suche im Informationssystem (Seite 92)

3.4.3 Beispiele zur Suche im Informationssystem

Beispiele für die Suche nach einem Wort

Bei der Suche nach einem Wort werden alle Hilfeseiten gefunden, die den Suchbegriff entweder exakt oder leicht abgewandelt enthalten. Begriffe, die Sonderzeichen enthalten, müssen in Anführungszeichen gesetzt werden:

- Suche nach `<PLC-Datentyp>`
  Es werden alle Seiten gefunden, die die Begriffe "PLC-Datentyp", "PLC-Datentyps", "PLC-Datentypen" oder ähnliche verwandte Begriffe enthalten.

Beispiele für die Suche nach mehreren Wörtern

Bei der Suche nach mehreren Wörtern werden alle Hilfeseiten gefunden, die alle Suchbegriffe entweder exakt oder leicht abgewandelt enthalten. Dabei spielt weder die Reihenfolge, noch der Abstand der gesuchten Wörter voneinander eine Rolle:

- Suche nach `<Variant Beispiel>`
  Es werden alle Seiten gefunden, die die Begriffe "Variant", "VARIANT", "Variant-Anweisung" und "Beispiel", "Beispiele", "Beispiels" oder ähnliche verwandte Begriffe enthalten.
- Suche nach `<LED Variable>`
  Es werden alle Seiten gefunden, die die Begriffe "LED Variable", "LED-Variable", "LED-Status der Variable" oder ähnliche verwandte Begriffe enthalten.

Beispiele für die Suche nach Phrasen

Bei der Suche nach Phrasen werden alle Hilfeseiten gefunden, in denen alle Suchbegriffe exakt in der eingegebenen Reihenfolge enthalten sind. Setzen Sie für eine Phrasensuche Ihren Suchausdruck in Anführungszeichen.

- Suche nach `<"Beispiele zum Einsatz von Rezepturen">`
  Es werden alle Seiten gefunden, die die Begriffe "Beispiele zum Einsatz von Rezepturen" oder ähnliche Begriffe in der angegebenen Reihenfolge enthalten.

Beispiele für die boolesche Suche mit "AND"

Bei der booleschen Suche mit "AND" werden alle Hilfeseiten gefunden, die alle Suchbegriffe entweder exakt oder leicht abgewandelt enthalten. Dabei spielt weder die Reihenfolge, noch der Abstand der gesuchten Wörter voneinander eine Rolle:

- Suche nach `<Benutzergruppen AND Berechtigung>`
  Es werden alle Seiten gefunden, die die Begriffe "Benutzergruppen", "Benutzergruppe" und "Berechtigung", "Berechtigungen" oder ähnliche verwandte Begriffe enthalten.
- Suche nach `<Beispiele AND "Einsatz von Rezepturen">`
  In diesem Beispiel ist die boolesche Suche mit "AND" mit der Phrasensuche kombiniert: Es werden alle Seiten gefunden, die exakt die Begriffe "Einsatz von Rezepturen" in der angegebenen Reihenfolge und "Beispiele" an einer beliebigen Position enthalten.

Beispiele für die boolesche Suche mit "OR"

Bei der booleschen Suche mit "OR" werden alle Hilfeseiten gefunden, in denen einer oder mehrere der angegebenen Suchbegriffe exakt oder leicht abgewandelt enthalten sind.

- Suche nach `<Baustein OR Funktion>`
  Es werden alle Seiten gefunden, die die Begriffe "Baustein", "Bausteine", "Bausteins" und / oder "Funktion", "Funktionen" oder ähnliche verwandte Begriffe enthalten.
- Suche nach `<Baustein OR Funktion OR Funktionsbaustein>`
  Es werden alle Seiten gefunden, die die Begriffe "Baustein", "Bausteine", "Bausteins" und / oder "Funktion", "Funktionen" und / oder "Funktionsbaustein", "Funktionsbausteine" oder ähnliche verwandte Begriffe enthalten.

Beispiele für die boolesche Suche mit "NOT"

Bei der booleschen Suche mit "NOT" werden alle Hilfeseiten gefunden, die die angegebenen Suchbegriffe, aber nicht den negierten Suchbegriff enthalten.

- Suche nach `<Meldung CPU NOT GRAPH>`
  Es werden alle Seiten gefunden, die die Begriffe "Meldungen", "Meldung" und "CPU", aber nicht das Wort "GRAPH" enthalten.
- Suche nach `<Meldung NOT GRAPH CPU>`
  Es werden alle Seiten gefunden, die die Begriffe "Meldungen", "Meldung" und "CPU", aber nicht das Wort "GRAPH" enthalten.
- Suche nach `<Meldung NOT CPU NOT GRAPH>`
  Es werden alle Seiten gefunden, die den Begriff "Meldung" aber nicht das Wort "GRAPH" und nicht das Wort "CPU" enthalten.

Beispiele für die unscharfe Suche mit "~"

Bei der unscharfen Suche werden alle Hilfeseiten gefunden, in denen der Suchbegriff in ungefährer Schreibweise vorkommt.

- Suche nach `<Kurven~>`
  Es werden alle Seiten gefunden, die die Begriffe "Kurve", "Kurvenbit", "Kurvenwert", "Kurvenanzeige", "Kurvenschreiber" oder ähnliche verwandte Begriffe enthalten.

Beispiele für die Abstandssuche mit "~<Abstand>"

Bei der Abstandssuche werden alle Hilfeseiten gefunden, in denen zwei Suchbegriffe nicht weiter als der definierte Abstand voneinander entfernt stehen.

- Suche nach `<Kurve Messung~10>`
  Es werden alle Seiten gefunden, in denen die Begriffe "Kurve" und "Messung" nicht weiter als 10 Worte voneinander entfernt stehen.

Beispiele für die Suche mit Filter

Bei der Suche mit Filter wird der Suchbegriff nur in den Seiten oder Kapiteln gesucht, die für das angegebene Gerät gültig sind.

- Suche nach `<Beispiel>` mit Suchfilter "S7-1500"
  Es werden alle Seiten gefunden, die die Begriffe "Beispiel", "Beispiele", "Beispiels" oder ähnliche verwandte Begriffe enthalten und für S7-1500 gültig sind.
- Suche nach `<SCL>` mit Suchfilter "S7-400"
  Es werden alle Seiten gefunden, die den Begriff "SCL" enthalten und für S7-400 gültig sind.

3.5 Favoriten verwenden

Favoriten verwenden

Sie können einzelne Hilfeseiten oder -kapitel als Favoriten abspeichern. Dadurch ersparen Sie sich eine erneute Suche nach dem Hilfethema.

Favoriten bleiben erhalten, auch wenn Sie das TIA Portal schließen. Nach dem erneuten Öffnen des Informationssystems stehen sie weiterhin zur Verfügung.

Favoriten speichern

Um eine Seite oder ein Kapitel als Favorit abzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Seite oder das Kapitel, das Sie als Favorit speichern möchten.
2. Klicken Sie auf die Schaltfläche "Zu Favoriten hinzufügen".
   Oder:
1. Selektieren Sie eine Seite oder ein Kapitel im Inhaltsverzeichnis.
2. Wählen Sie den Kontextmenübefehl "Zu Favoriten hinzufügen".

Das Hilfethema oder das Kapitel wird als Favorit gespeichert und steht Ihnen auch beim erneuten Öffnen der Hilfe zur Verfügung.

Favoriten aufrufen

Um eine Seite aus den Favoriten aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Register "Favoriten".
2. Doppelklicken Sie auf das Hilfethema, das Sie öffnen möchten.
   Oder:
1. Öffnen Sie das Register "Favoriten".
2. Wählen Sie den Kontextmenübefehl "In neuer Registerkarte öffnen".

Das Hilfethema wird im Inhaltsbereich im aktuellen Register oder in einem neuen Register geöffnet.

Zur Position im Inhaltsverzeichnis gehen

Um aus den Favoriten zur Position des Themas im Inhaltsverzeichnis zu navigieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in den Favoriten das Thema, zu dessen Position im Inhaltsverzeichnis Sie gehen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu".
   Der Menübefehl ist nur aktiv, wenn Sie ein Hilfethema anwählen, das im Inhaltsverzeichnis enthalten ist. Kontextsensitive Hilfethemen, z. B. Hilfe zu Syntaxfehlern, sind nicht im Inhaltsverzeichnis enthalten. Für sie ist die Funktion "Gehe zu" nicht ausführbar.

Das Inhaltsverzeichnis wird geöffnet und an der gewünschten Position aufgeklappt.

Favoriten löschen

Um einen Eintrag aus den Favoriten zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Register "Favoriten".
2. Wählen Sie das Thema aus der Liste aus, das Sie entfernen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Entfernen".

3.6 Codebeispiele in das Programm übernehmen

Zahlreiche Hilfethemen enthalten Codebeispiele, die Anweisungen und Funktionen zur Programmierung im TIA Portal veranschaulichen. Codebeispiele in den Programmiersprachen SCL und AWL können Sie kopieren und direkt in Ihrem Programm verwenden.

Beispiel

Ein kopierbares Beispiel erkennen Sie an der Schaltfläche "Kopieren" in der Überschrift des Programmbeispiels:

SCL

```
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

Vorgehen

Um ein Codebeispiel zu kopieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Hilfethema mit dem gewünschten Programmbeispiel.
2. Klicken Sie auf die Schaltfläche "Kopieren" in der Überschrift des Programmbeispiels.
   Der Programmcode wird in die Zwischenablage kopiert.
3. Öffnen Sie den Programmiereditor und fügen Sie den kopierten Code ein. Wählen Sie dazu im Kontextmenü den Befehl "Einfügen".
   Der kopierte Programmcode wird aus der Zwischenablage in den bearbeiteten Programmbaustein eingefügt.

3.7 Weiterführende Informationen im Siemens Industry Online Support anzeigen

Das Informationssystem bietet Ihnen einen Einstieg zum "TIA Portal Information Center" im Siemens Industry Online Support. Hier erhalten Sie aktuelle weiterführende Informationen rund um das TIA Portal, zum Beispiel Applikationsbeispiele, Trainingsangebote, FAQs, Hardware-Dokumentation und vieles mehr.

Hinweis

Internet-Verbindung erforderlich

Um die Informationen im Siemens Industry Online Support zu erreichen, benötigen Sie eine Internet-Verbindung.

Vorgehen

Um weiterführende Informationen im Siemens Industry Online Support zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Startseite des Informationssystems. Dazu haben Sie folgende Möglichkeiten:
   - Wählen Sie im Inhaltsverzeichnis den obersten Eintrag. Der Eintrag hat den Titel "Informationssystem".
   - Wählen Sie im Inhaltsbereich die Schaltfläche "Startseite".

   Die Startseite des Informationssystems wird geöffnet. Im unteren Bereich der Startseite wird der "TIA Portal Information Center" angezeigt.
2. Klicken Sie auf das Bild des "TIA Portal Information Center".
   Der "TIA Portal Information Center" wird geöffnet.

3.8 Hardware-Handbücher anzeigen

Einführung

Mit dem TIA Portal lässt sich eine große Menge unterschiedlicher Hardware projektieren. Die zugehörigen Informationen finden Sie im Kapitel "Hardware-Handbücher" des Informationssystems. Sie können die Handbücher auf zwei verschiedene Arten nutzen:

- Hardware-Handbücher im Informationssystem
  Sie haben die Möglichkeit, Hardware-Handbücher integriert im Informationssystem anzuzeigen. Das hat den Vorteil, dass Sie die integrierten Inhalte durchsuchen, filtern oder als Favoriten verwenden können. Integrierte Handbücher können Sie auch über die Druckfunktion des Informationssystems ausdrucken.
  Darüberhinaus haben Sie die Möglichkeit, weitere Hardware-Handbücher, die als Support Package zur Verfügung stehen, herunterzuladen.
- Hardware-Handbücher im Siemens Industry Online Support
  Alle aktuellen Handbücher, Betriebsanleitungen und FAQs sowie Updates zu Ihren Geräten finden Sie im Siemens Industry Online Support. Um den Online Support zu erreichen, benötigen Sie eine Internet-Verbindung.

Nach Updates zu weiteren Hardware-Handbüchern suchen

Um zu prüfen, ob weitere Hardware-Handbücher als Update zur Verfügung stehen und diese herunterzuladen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Inhaltsverzeichnis des Informationssystems.
2. Klicken Sie in der Funktionsleiste des Inhaltsverzeichnisses auf die Schaltfläche "Nach Updates suchen".
   Alle verfügbaren Support Packages werden angezeigt.
3. Prüfen Sie, ob ein Support Package zur Hardware-Dokumentation bereit steht.
4. Laden Sie das Support Package herunter.
   Siehe auch: AUTOHOTSPOT
   Nach dem nächsten Start des Informationssystems finden Sie die heruntergeladenen Hardware-Handbücher im Kapitel "Hardware-Handbücher".

Informationssystem nach Informationen aus Hardware-Handbüchern durchsuchen

Um gezielt nach Informationen aus integrierten Hardware-Handbüchern zu suchen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Suchbereich.
2. Geben Sie im Eingabefeld "Suchen nach" einen Suchbegriff ein.
3. Optional: Wählen Sie im Auswahlfeld "Geräte" einen Suchfilter aus, um die Suche auf ein bestimmtes Gerät oder eine Gerätefamilie beschränken.
4. Wählen Sie im Auswahlfeld "Bereich" den Suchfilter "Hardware-Handbücher" aus, um die Suche auf integrierte Hardware-Handbücher zu beschränken.

5. Klicken Sie auf "Suchen".
   Die Suche wird durchgeführt und die Ergebnisse werden aufgelistet.
   - In der Spalte "Titel" wird der Titel des gefundenen Hilfethemas angezeigt.
   - In der Spalte "Position" wird das Kapitel angezeigt, in dem das Hilfethema enthalten ist. Kapitel aus Hardware-Handbüchern sind mit dem Präfix "HW" gekennzeichnet.
   - In der Spalte "Rang" wird die Relevanz des Hilfethemas bezüglich des Suchbegriffs angezeigt.

   Sie können die Sortierung ändern, indem Sie auf die Kopfzeile einer Spalte klicken.
6. Doppeklicken Sie auf ein Suchergebnis oder wählen Sie im Kontextmenü die Befehle "Öffnen" oder "In neuer Registerkarte öffnen".
   Das Hilfethema wird im Inhaltsbereich geöffnet.

Siehe auch

Informationssystem durchsuchen (Seite 89)

3.9 Hilfethemen drucken

Sie haben folgende Möglichkeiten, Inhalte des Informationssystems zu drucken:

- Sie können ausgewählte Kapitel oder Seiten aus dem Inhaltsverzeichnis gesammelt ausdrucken. Dabei können Sie beliebige Inhalte des Informationssystems auswählen. Es muss sich nicht um Inhalte aus zusammenhängenden Kapiteln handeln. Die ausgewählten Inhalte werden in einem druck-optimierten Layout mit Seitennummerierung ausgedruckt.
- Sie können eine geöffnete Seite einzeln ausdrucken.

Kapitel oder Seiten aus dem Inhaltsverzeichnis ausdrucken

Um Inhalte aus dem Inhaltsverzeichnis zu drucken, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Inhaltsverzeichnis des Informationssystems.
2. Klicken Sie im Register "Inhalt" auf die Schaltfläche "Drucken".
   Der Dialog "Drucken" wird geöffnet.
3. Wählen Sie im Dialog "Drucken" die Kapitel und Seiten an, die Sie drucken möchten.
4. Klicken Sie auf die Schaltfläche "Drucken", um die markierten Informationen zu drucken.
   Der Standard-Druckdialog wird geöffnet.
5. Wählen Sie den Drucker aus, auf dem Sie die Informationen ausdrucken möchten.
6. Klicken Sie auf "Eigenschaften", falls Sie weitere Druckereinstellungen angeben möchten.
7. Bestätigen Sie Ihre Eingaben mit "OK".
   Die Informationen werden auf dem ausgewählten Drucker gedruckt.

Geöffnete Seite einzeln ausdrucken

Um eine einzelne Seite auszudrucken, gehen Sie folgendermaßen vor:

1. Öffnen Sie die zu druckende Seite im Inhaltsbereich des Informationssystems.
2. Wählen Sie im Inhaltsbereich die Schaltfläche "Drucken".
3. Der Standard-Druckdialog wird geöffnet.
4. Wählen Sie den Drucker aus, auf dem Sie die Informationen ausdrucken möchten.
5. Klicken Sie auf "Eigenschaften", falls Sie weitere Druckereinstellungen angeben möchten.
6. Bestätigen Sie Ihre Eingaben mit "OK".
   Die Seite auf dem ausgewählten Drucker gedruckt.

3.10 Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren

Konfigurationsmöglichkeiten für Tooltips und Tooltip-Kaskaden

Sie können die Anzeige von Tooltips und Tooltip-Kaskaden an Ihre Bedürfnisse anpassen. Sie haben folgende Einstellungsmöglichkeiten:

- Anzeige abgeschnittener Texte ein- oder ausblenden
  Manchmal können Texte zu lang für ein Textfeld sein. Die Texte werden dann in einem Tooltip vollständig angezeigt, wenn Sie mit der Maus über dem Textfeld verweilen. Sie können diese Funktion aktivieren oder deaktivieren.
- Tooltips aktivieren oder deaktivieren
  Tooltips geben genauere Informationen zu einem Oberflächenelement. Außerdem ermöglichen Sie die Anzeige von Tooltip-Kaskaden. Wenn Sie die Tooltips deaktivieren, werden daher auch die Kaskaden mit kontextbezogener Hilfe nicht mehr angezeigt. Sie haben jedoch die Möglichkeit, den Tooltip für das gerade aktive Oberflächenelement manuell anzuzeigen, indem Sie <F1> drücken.
  Die Einstellung wirkt sich ebenfalls auf Rollouts mit kontextbezogene Syntaxfehlern und Eingabehilfen in AWL aus.
- Das automatische Öffnen von Tooltip-Kaskaden aktivieren oder deaktivieren
  Nach einer kurzen Verweildauer mit der Maus über einem Tooltip werden eventuell vorhandene Kaskaden automatisch angezeigt. Sie können die automatische Anzeige der Kaskaden aktivieren oder deaktivieren. Bei deaktivierter automatischer Anzeige müssen Sie die Kaskade gegebenenfalls manuell öffnen. Klicken Sie dazu auf das Pfeilsymbol innerhalb des Tooltips.

Vorgehen

Um die Anzeige von Tooltips und Tooltip-Kaskaden zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Aktiveren oder deaktivieren Sie die einzelnen Optionskästchen im Bereich "Tooltips" entsprechend Ihren Bedürfnissen. Das Optionskästchen "Kaskade in den Tooltips automatisch öffnen" ist nur bedienbar, wenn Sie die Anzeige der Tooltips aktiviert haben.

Siehe auch

Allgemeines zum Informationssystem (Seite 79)

3.11 Individuelle Dokumentation zusammenstellen

Individualisierte Dokumentation

Im Siemens Industry Online Support können Sie sich in der Kategorie „mySupport-Dokumentation“ eine individuelle und an Ihre Bedürfnisse angepasste Dokumentation zusammenstellen. Dafür stehen Ihnen alle konfigurierbaren Handbücher und Betriebsanleitungen des Siemens Industry Online Supports zur Verfügung. Daraus entnehmen Sie die für Sie interessanten Teile und kombinieren diese zu einer persönlichen Dokumentation in einer Bibliothek. Die Dokumentation strukturieren Sie mit Hilfe von Ordnern in Ihrer persönlichen Bibliothek. Die Ordner werden später zu den einzelnen Kapiteln der individuellen Dokumentation.

"mySupport-Dokumentation" rufen Sie hier auf:

https://support.industry.siemens.com/My/ww/de/documentation (https://support.industry.siemens.com/My/ww/de/documentation)

Voraussetzung

- Die verwendeten Handbücher oder Betriebsanleitungen müssen konfigurierbar sein. Konfigurierbare Handbücher erkennen Sie am Namenszusatz "vollkonfigurierbar".
- Um den vollen Funktionsumfang nutzen zu können, müssen Sie im Siemens Industry Online Support registriert und angemeldet sein.

Dokumentation in verschiedenen Sprachen

Die zusammengestellte Dokumentation können Sie bei Bedarf in die Sprachen Englisch, Französisch, Spanisch, Italienisch und Chinesisch umschalten. So haben Sie beispielsweise die Möglichkeit, relevante Informationen für ein bestimmtes Projekt zusammenzutragen und diese anderssprachigen Kollegen zur Verfügung zu stellen.

Exportfunktion innerhalb der Dokumentation

In Ihrer Bibliothek können Sie an jeder beliebigen Stelle Teile in unterschiedlichen Formaten (PDF, XML, RTF) exportieren.

Hilfe zum Erstellen der Dokumentation

Weitere Hilfe zum Erstellen und zur Nutzung der individuellen Dokumentation erhalten Sie im Siemens Industry Online Support:

https://support.industry.siemens.com/cs/helpcenter/de/index.htm (https://support.industry.siemens.com/cs/helpcenter/de/index.htm)

3.12 Sicherheitshinweise

Sicherheitshinweise

In der Hilfe finden Sie Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten wird, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
mit Warndreieck bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
ohne Warndreieck bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Hinweis

bedeutet, dass ein unerwünschtes Ergebnis oder Zustand eintreten kann, wenn der entsprechende Hinweis nicht beachtet wird.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zugehörige Gerät/System darf nur in Verbindung mit dieser Dokumentation eingerichtet und betrieben werden. Inbetriebsetzung und Betrieb eines Gerätes/Systems dürfen nur von qualifiziertem Personal vorgenommen werden. Qualifiziertes Personal im Sinne der sicherheitstechnischen Hinweise dieser Dokumentation sind Personen, die die Berechtigung haben, Geräte, Systeme und Stromkreise gemäß den Standards der Sicherheitstechnik in Betrieb zu nehmen, zu erden und zu kennzeichnen.

Bestimmungsgemäßer Gebrauch

Beachten Sie Folgendes:

⚠ WARNUNG
Das Gerät darf nur für die im Katalog und in der technischen Beschreibung vorgesehenen Einsatzfälle und nur in Verbindung mit von Siemens empfohlenen bzw. zugelassenen Fremdgeräten und -komponenten verwendet werden. Der einwandfreie und sichere Betrieb des Produktes setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung und Montage sowie sorgfältige Bedienung und Instandhaltung voraus.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Hilfe auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Hilfe werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Suchen und ersetzen 4

4.1 Grundlagen zur Suche

Einführung

Innerhalb des TIA Portals können Sie die folgenden Suchmöglichkeiten nutzen:

- Gesamtes Projekt durchsuchen
- Innerhalb eines Editors suchen und ersetzen
- Hardware-Katalog durchsuchen

Darüber hinaus bieten einzelne Produkte im TIA Portal weitere Suchmöglichkeiten.

Gesamtes Projekt durchsuchen

Sie können das gesamte Projekt nach einem bestimmten Text durchsuchen. Dazu steht Ihnen ein Sucheditor zur Verfügung, in dem Sie z. B. die Suche weiter eingrenzen können. Die Objekte, die den gesuchten Text enthalten, werden übersichtlich in einer Tabelle dargestellt. Sie können jedes Objekt vom Sucheditor aus öffnen, um sich die entsprechende Stelle anzusehen.

Innerhalb eines Editors suchen und ersetzen

Sie haben die Möglichkeit, nach Texten innerhalb von Editoren zu suchen. Die Suchfunktion findet alle Texte innerhalb des gerade geöffneten Editors, die den Suchbegriff enthalten. Die Ergebnisse werden nacheinander im geöffneten Editor selektiert.

Weiterhin haben Sie folgende Möglichkeiten:

- Verfeinern der Suche durch Zusatzoptionen
- Ersetzen gefundener Texte

Welche Zusatzoptionen verfügbar sind und welche Arten von Texten gesucht werden können, ist von den installierten Produkten und dem geöffneten Editor abhängig.

Hardware-Katalog durchsuchen

Sie können im Hardware-Katalog nach einer bestimmten Hardware-Komponente suchen.

Siehe auch: Hardware-Katalog durchsuchen

Siehe auch

Im Editor suchen und ersetzen (Seite 120)

Im gesamten Projekt suchen (Seite 111)

4.2 Gesamtes Projekt durchsuchen

4.2.1 Grundlagen zur Suche im Projekt

Einführung

Sie können innerhalb des geöffneten Projekts nach einem bestimmten Text (Suchausdruck) suchen. Der Suchausdruck kann aus einem Wort oder mehreren Wörtern oder aus Teilen von Wörtern bestehen. Dabei werden nur Leerzeichen als Wortgrenzen berücksichtigt. Sie können Suchausdrücke mit einer Länge von bis zu 255 Zeichen verwenden. Über Platzhalterzeichen können Sie beliebig komplexe Suchausdrücke erzeugen. Zusätzlich kann die Suche auf bestimmte Bereiche, Objekte oder Objekteigenschaften eingegrenzt werden. Groß- und Kleinschreibung wird bei der Suche nicht berücksichtigt.

Für die Suche wird im Hintergrund automatisch ein Index über die Texte der durchsuchbaren Editoren im Projekt aufgebaut. Bei jeder Änderung wird der Suchindex nachgezogen. Solange die Indexierung nicht abgeschlossen ist, können Sie keine Suche starten. Sie können den Suchindex jederzeit neu erstellen lassen.

Beachten Sie, dass der Suchindex mit dem Projekt gespeichert wird. Dadurch kann die Dateigröße des Projekts ansteigen. Wenn Sie das Projekt weitergeben möchten, können Sie die Dateigröße reduzieren, in dem Sie das Projekt archivieren oder minimieren. Dabei wird der Suchindex entfernt. Sie können die Suche im Projekt bei Bedarf auch komplett deaktivieren.

Bei Änderungen von STEP 7-Objekten, z. B. Namensänderungen von Variablen, PLC-Datentypen oder Bausteinen, ist es erforderlich, dass Sie vor einer Suche im Projekt die Verwendungsstellen dieser Objekte aktualisieren, damit die Suche diese Änderungen berücksichtigen kann. Dazu können Sie z. B. Ihr Programm neu übersetzen.

Die Suche erfasst nur die Texte in der eingestellten Editiersprache. Wenn Sie die Editiersprache wechseln, wird der Suchindex neu erzeugt. Abhängig von der Anzahl der im Projekt vorhandenen Texte kann dies einige Zeit dauern.

Für die Suche steht Ihnen der Sucheditor zur Verfügung. In ihm können Sie die Suchoptionen festlegen und die Suche starten. Nach erfolgter Suche werden im Sucheditor die Objekte aufgelistet, die den gesuchten Text enthalten. Sie haben die Möglichkeit, die einzelnen Objekte direkt aus dem Sucheditor heraus zu öffnen. Wenn die Objekte anzeigbare Eigenschaften haben, können Sie sich diese im Inspektorfenster anzeigen lassen.

Suchoptionen

Für die Suche stehen Ihnen die folgenden Optionen zur Verfügung:

- Suche einschränken:
  - Suchen in: Sie können auswählen, welche Teilbereiche des Projekts durchsucht werden sollen.
  - Nur genaue Treffer finden: Sucht genau nach dem eingegebenen Text. Der Suchausdruck wird dabei als einzelnes Wort oder als Phrase gesucht. Bei Verwendung dieser Option können Sie im Suchausdruck keine Platzhalterzeichen verwenden, d. h. bei Eingabe von "*" oder "?" wird genau nach diesen Zeichen gesucht.
  - Suche einschränken auf: Sie können die Suche auf bestimmte Objekteigenschaften oder Objekttypen einschränken.
- Verwenden von "*" als Platzhalter für beliebig viele Zeichen und "?" als Platzhalter für exakt ein Zeichen innerhalb eines Wortes:
  Wenn Sie die Platzhalter "*" oder "?" innerhalb eines Suchausdrucks verwenden, erhalten Sie als Treffer ganze Wörter, die auf diesen Suchausdruck passen. Wenn Sie z. B. "Motor*start" eingeben, erhalten Sie Treffer wie "Motorstarter" oder "Motor1_starten", aber nicht "Motoren starten" oder "Motor1 starten".
  Grundsätzlich findet eine implizite Platzhaltersuche statt. Geben Sie z. B. "Motor" als Suchausdruck ein, werden auch die Texte "Motor_ON", "Motorstarter" und "123_Motor" gefunden. Dies entspricht der expliziten Platzhaltersuche mit dem Suchausdruck "*Motor*". Sie können die implizite Platzhaltersuche abschalten, indem Sie die Option "Nur genaue Treffer finden" aktivieren.

Texte im Projekt ersetzen

Sie haben die Möglichkeit, einzelne Texte im Projekt zu ersetzen. Dazu können Sie nach einer Suche die Fundstelle direkt über einen Doppelklick in der Trefferliste öffnen und den Text bearbeiten. Nach dem Ändern wird der Text automatisch neu indiziert. Wenn Sie danach den nächsten Treffer öffnen möchten, erhalten Sie eine Meldung angezeigt, die Ihnen die Möglichkeit bietet, die Trefferliste zu aktualisieren. Auf diese Weise können Sie einen Text im gesamten Projekt ersetzen, bis die Trefferliste keine Treffer mehr anzeigt.

Siehe auch

Grundlagen zur Suche (Seite 107)

Übersicht über den Sucheditor (Seite 110)

Im gesamten Projekt suchen (Seite 111)

Suchindex manuell neu erstellen (Seite 113)

Suche im Projekt aktivieren und deaktivieren (Seite 113)

Beispiele zur Suche im Projekt (Seite 114)

4.2.2 Übersicht über den Sucheditor

Funktion des Sucheditors

Der Sucheditor erlaubt es Ihnen, das gesamte Projekt nach bestimmten Texten zu durchsuchen und stellt die Suchergebnisse anschließend übersichtlich dar.

Aufbau des Sucheditors

Das folgende Bild zeigt die Komponenten des Sucheditors:

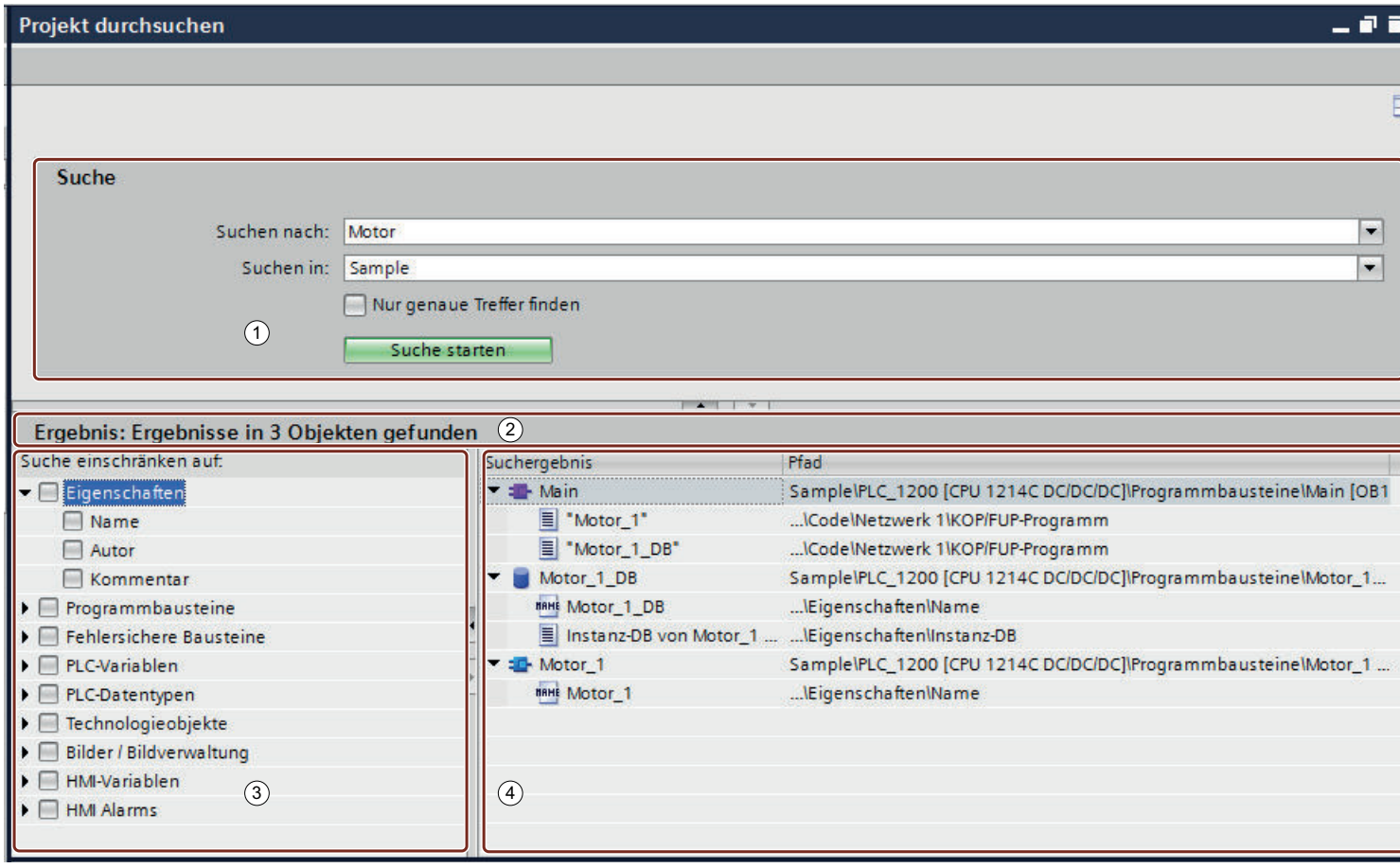

① Suchmaske

② Statusleiste

③ Suchfilter

④ Ergebnisliste

Suchmaske

Über die Suchmaske können Sie den Suchausdruck eingeben und den Suchbereich einschränken. Zudem haben Sie die Möglichkeit die Option "Nur genaue Treffer finden" zu aktivieren, wodurch im Projekt exakt nach dem eingegebenen Suchausdruck gesucht wird und ähnliche Textstellen ignoriert werden. In der Suchmaske starten Sie die Suche.

Statusleiste

In der Statusleiste erhalten Sie Informationen über das Suchergebnis.

Suchfilter

Über die Suchfilter können Sie die Suchergebnisse weiter einschränken. Wenn Sie keine Filter aktiviert haben, werden alle Objekte und Objekteigenschaften bei der Suche berücksichtigt. Aktivieren Sie einen Filter, werden nur die Suchergebnisse angezeigt, die dem Filterkriterium entsprechen.

Ergebnisliste

In der Ergebnisliste werden die Objekte angezeigt, die den Suchausdruck enthalten. Mit einem Doppelklick auf ein Suchergebnis wird der entsprechende Editor geöffnet und die Fundstelle markiert.

Siehe auch

Grundlagen zur Suche (Seite 107)

Grundlagen zur Suche im Projekt (Seite 108)

Im gesamten Projekt suchen (Seite 111)

Suchindex manuell neu erstellen (Seite 113)

Suche im Projekt aktivieren und deaktivieren (Seite 113)

Beispiele zur Suche im Projekt (Seite 114)

4.2.3 Im gesamten Projekt suchen

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um das aktuelle Projekt nach einem bestimmten Text zu durchsuchen, gehen Sie folgendermaßen vor:

1. Geben Sie in der Funktionsleiste des TIA Portals in das Textfeld für die Suche den Suchausdruck ein.
2. Klicken Sie auf "Projekt durchsuchen".
   Der Sucheditor wird geöffnet und eine Suche nach dem Suchausdruck wird über das gesamte Projekt durchgeführt. Die Objekte, die den gesuchten Text enthalten, werden in der Übersicht aufgelistet.

   Hinweis

   Bei dieser Suche können Sie keine Suchoptionen eingeben. Es wird das komplette Projekt durchsucht und alle durchsuchbaren Objekte werden in die Suche einbezogen. Wenn Sie die Suche weiter eingrenzen oder Suchoptionen verwenden möchten, können Sie die Suche erneut starten. Folgen Sie dazu der alternativen Anleitung unter "Oder" ab Schritt 2.

Oder:

1. Öffnen Sie den Sucheditor über einen der folgenden Wege:
   - Wählen Sie im Menü "Bearbeiten" den Befehl "Projekt durchsuchen".
   - Klicken Sie in der Projektnavigation mit der rechten Maustaste auf ein Objekt und wählen Sie im Kontextmenü den Befehl "Projekt durchsuchen". Das Objekt wird als Suchbereich im Feld "Suchen in" voreingestellt.
   - Stellen Sie sicher, dass sich der Fokus außerhalb eines Editors befindet, der eine editorspezifische Suche unterstützt, und drücken Sie die Tastenkombination <Strg+F>.
2. Geben Sie den Suchausdruck in das Feld "Suchen" ein oder wählen Sie in der Liste einen Ausdruck, nach dem Sie zuvor bereits gesucht haben.
3. Um exakt nach dem Suchausdruck zu suchen, aktivieren Sie das Optionskästchen "Nur genaue Treffer finden". Bei Verwendung dieser Option können Sie im Suchausdruck keine Platzhalterzeichen verwenden, d. h. bei Eingabe von "*" oder "?" wird genau nach diesen Zeichen gesucht.
4. Um den Suchbereich einzuschränken, aktivieren Sie unter "Suchen in" den Teilbereich, der durchsucht werden soll.
5. Um die Suche auf bestimmte Objekte oder Objekteigenschaften einzugrenzen, aktivieren Sie unter "Suche einschränken auf" die Objekte oder Objekteigenschaften, nach denen gesucht werden soll.
6. Klicken Sie auf "Suche starten".
   Die Objekte, die den gesuchten Text enthalten, werden in der Übersicht aufgelistet.
7. Klicken Sie doppelt auf ein Objekt, um es zu öffnen.

Siehe auch

Grundlagen zur Suche (Seite 107)

Grundlagen zur Suche im Projekt (Seite 108)

Übersicht über den Sucheditor (Seite 110)

Suchindex manuell neu erstellen (Seite 113)

Suche im Projekt aktivieren und deaktivieren (Seite 113)

Beispiele zur Suche im Projekt (Seite 114)

Im Editor suchen und ersetzen (Seite 120)

4.2.4 Suchindex manuell neu erstellen

Für die Suche wird im Hintergrund automatisch ein Index über die Texte der durchsuchbaren Editoren im Projekt aufgebaut. Bei jeder Änderung wird der Suchindex nachgezogen. Solange die Indexierung nicht abgeschlossen ist, können Sie keine Suche starten. Sie können das Erzeugen des Suchindexes aber auch jederzeit manuell anstoßen.

Vorgehen

Um den Suchindex neu erstellen zu lassen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein > Projekt durchsuchen".
3. Klicken Sie auf "Suchindex neu erstellen".

Ergebnis

Der Suchindex für das Projekt wird neu erstellt.

Siehe auch

Grundlagen zur Suche (Seite 107)

Grundlagen zur Suche im Projekt (Seite 108)

Übersicht über den Sucheditor (Seite 110)

Im gesamten Projekt suchen (Seite 111)

Suche im Projekt aktivieren und deaktivieren (Seite 113)

Beispiele zur Suche im Projekt (Seite 114)

4.2.5 Suche im Projekt aktivieren und deaktivieren

Sie können die Suche im Projekt zentral deaktivieren, wenn Sie sie nicht verwenden. Bei Bedarf können Sie sie jederzeit wieder aktivieren. Standardmäßig ist die Suche aktiviert.

Vorgehen

Um die Suche im Projekt zu deaktivieren oder zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein > Projekt durchsuchen".
3. Deaktivieren oder Aktivieren Sie das Optionskästchen "'Projekt durchsuchen' aktivieren".

Siehe auch

Grundlagen zur Suche (Seite 107)

Grundlagen zur Suche im Projekt (Seite 108)

Übersicht über den Sucheditor (Seite 110)

Im gesamten Projekt suchen (Seite 111)

Suchindex manuell neu erstellen (Seite 113)

Beispiele zur Suche im Projekt (Seite 114)

4.2.6 Beispiele zur Suche im Projekt

Beispiele zur Verwendung von Eigenschafts- und Objektfilter

Im Sucheditor stehen Ihnen im Bereich "Suche einschränken auf" Filter zur Verfügung, mit denen Sie die Suche auf bestimmte Eigenschaften oder Objekte einschränken können. Beachten Sie dabei den folgenden Mechanismus:

- Wenn Sie keine Filter aktivieren, wird der Suchausdruck innerhalb des eingestellten Suchbereichs in allen durchsuchbaren Objekten und Eigenschaften gesucht.
- Wenn Sie Filter aktivieren, werden nur noch die gewählten Eigenschaften bzw. Objekte durchsucht.

Beispiele für die Verwendung von Eigenschafts- und Objektfilter:

- Suche nach einem Objekt mit dem Namen "Motor1":
  Geben Sie den Suchausdruck ein und aktivieren Sie den Filter "Name". Es werden alle Objekte gefunden, bei denen "Motor1" im Namen enthalten ist.
- Suche nach einem Objekt, in dem der Text "Motor1" entweder im Namen oder im Kommentar vorkommt:
  Geben Sie den Suchausdruck ein und aktivieren Sie die Filter "Name" und "Kommentar". Es werden alle Objekte gefunden, bei denen "Motor1" im Namen oder im Kommentar enthalten ist.
- Suche nach einem Baustein mit dem Namen "Motor1":
  Geben Sie den Suchausdruck ein und aktivieren Sie die Filter "Name" und "Baustein". Es werden alle Bausteine gefunden, bei denen "Motor1" im Namen enthalten ist.

- Suche nach einem Baustein, in dem der Text "Motor1" entweder im Namen oder im Kommentar vorkommt:
  Geben Sie den Suchausdruck ein und aktivieren Sie die Filter "Name", "Kommentar" und "Baustein". Es werden alle Bausteine gefunden, bei denen "Motor1" im Namen oder im Kommentar enthalten ist.
- Suche nach einem Baustein oder einem Technologieobjekt mit dem Namen "Motor1":
  Geben Sie den Suchausdruck ein und aktivieren Sie die Filter "Name", "Baustein" und "Technologieobjekt". Es werden alle Bausteine oder Technologieobjekte gefunden, bei denen "Motor1" im Namen enthalten ist.

Beachten Sie beim Eingeben des Suchbegriffs auch die folgenden Beispiele für die Suche nach einem Begriff oder nach mehreren Begriffen.

Beispiele für die Suche nach einem Begriff

Wenn Sie einen Suchausdruck eingeben, der nur aus einem Begriff besteht, und die Option "Nur genaue Treffer finden" deaktiviert ist, findet immer eine implizite Platzhaltersuche statt. Dabei ergänzt das System vor und nach dem Suchausdruck das Platzhalterzeichen "*". Sie können auch selbst Platzhalter in den Suchausdruck einfügen, um eine explizite Platzhaltersuche durchzuführen. Möchten Sie hingegen genau den eingegebenen Begriff finden, aktivieren Sie die Option "Nur genaue Treffer finden". Groß- und Kleinschreibung wird bei der Suche nicht berücksichtigt.

Beispiele für die Suche nach einem Begriff:

- Suche nach "motor", Option "Nur genaue Treffer finden" ist deaktiviert
  Es findet eine implizite Platzhaltersuche statt. Der Suchausdruck wird dabei als "*motor*" interpretiert und es werden alle Begriffe gefunden, die "motor" enthalten, z. B. "motor", "motor_1", "Motorstart" oder "Start_Motor".
- Suche nach "m*tor" (explizite Platzhaltersuche), Option "Nur genaue Treffer finden" ist deaktiviert
  Der Suchausdruck wird bei dieser Platzhaltersuche als "*m*tor*" interpretiert. Daher werden z. B. auch die Begriffe "motor", "motor_1", "Start_Motor", "GLOB.Mode.motor[1]" oder "Mo_tor" gefunden.
- Suche nach "motor", Option "Nur genaue Treffer finden" ist aktiviert
  Es werden keine Platzhalter für die Suche verwendet, daher werden nur die Begriffe "motor" und "Motor" gefunden.
- Suche nach "m*tor" (explizite Platzhaltersuche), Option "Nur genaue Treffer finden" ist aktiviert
  Dem Zeichen "*" wird seine Platzhalterfunktion entzogen und es wird als einfaches Zeichen interpretiert. Daher werden bei der Suche nur die Zeichenfolgen "m*tor" und "M*tor" gefunden, aber z. B. nicht die Begriffe "motor", "Motor", "motor_1".

Anwendungsbeispiele:

- Wenn Sie ermitteln möchten, von wo aus auf das Element "GLOB_DB.MODE.FG[6].IN.ERROR_DROP_START” eines Datenbausteins zugegriffen wird, können Sie die Querverweisliste verwenden. Falls der Zugriff jedoch über indirekte Adressierung erfolgt (z. B. mit "#GLOB.MODE.FG[#FgNr].IN.ERROR_DROP_START := TRUE;"), liefert die Querverweisliste kein Ergebnis. Verwenden Sie in diesem Fall eine Platzhaltersuche nach "MODE.FG[*].IN.ERROR_DROP_START" über das gesamte Projekt.
- Sie möchten alle Stellen finden, an denen "ERROR_DROP_START" verwendet wird. Führen Sie dazu eine Suche über das gesamte Projekt nach “ERROR_DROP_START” durch. Dabei werden auch alle Elemente "ERROR_DROP_START" innerhalb des Arrays "FG[]" gefunden.

Beispiele für die Suche nach mehreren Begriffen

Wenn Sie einen Suchausdruck eingeben, der aus mehreren Begriffen besteht, und die Option "Nur genaue Treffer finden" deaktiviert ist, werden solche Objekte gefunden, in denen diese Begriffe alle enthalten sind. Die Texte können dabei in beliebiger Reihenfolge vorkommen. Sollen die Begriffe jedoch genau so in einem Objekt vorkommen, wie Sie sie eingegeben haben, aktivieren Sie die Option "Nur genaue Treffer finden". Groß- und Kleinschreibung wird bei der Suche nicht berücksichtigt.

Beispiele für die Suche nach mehreren Begriffen:

- Suche nach "motor start", Option "Nur genaue Treffer finden" ist deaktiviert
  Es werden alle Objekte gefunden, in denen die Begriffe "motor" und "start" vorkommen. Dabei spielt es keine Rolle wie weit sie voneinander entfernt stehen oder in welcher Reihenfolge sie vorkommen. Mögliche Treffer sind daher: "Motor start", "start my motor", "my motor is in start mode"

  Hinweis

  In Datenbausteinen und in Bausteinschnittstellen werden die Texte einer Spalte als ein Textobjekt behandelt. Daher werden bei der Suche auch die Datenbausteine und Bausteine gefunden, bei denen die Begriffe verteilt in unterschiedlichen Zeilen vorkommen.

- Suche nach "motor start*", Option "Nur genaue Treffer finden" ist deaktiviert
  Wie bei der Suche nach einem Wort wird auch bei derm Mehrwortsuche grundsätzlich eine implizite Platzhaltersuche für alle Wörter durchgeführt. Daher ist es nicht notwendig, ein Platzhalterzeichen explizit einzugeben. "motor start*" wird als "*motor* *start*" interpretiert und es werden die gleichen Treffer gefunden wie bei "motor start".

- Suche nach "motor start", Option "Nur genaue Treffer finden" ist aktiviert
  Wenn die Option "Nur genaue Treffer finden" aktiviert ist, werden nur die Objekte gefunden, in denen die beiden Begriffe nacheinander als Phrase vorkommen. Gefunden wird der Suchausdruck z. B. in Objekten in denen die Texte "push motor start button" oder "motor start on" vorhanden sind. Keine Treffer liefern jedoch "start motor", "motorstart", "start my motor", "my motor is in start mode".
- Suche nach "motor start*", Option "Nur genaue Treffer finden" ist aktiviert
  Dem Zeichen "*" wird seine Platzhalterfunktion entzogen und es wird als einfaches Zeichen interpretiert. Daher werden bei der Suche nur die Objekte gefunden, in denen die Zeichenfolge "motor start*" in beliebiger Groß-/Kleinschreibung vorkommt. "motor starting", "start my motor", "motor start", "Motor Start", "motor start_mode" liefern keine Treffer.

Anwendungsbeispiele:

- Sie möchten herausfinden, an welchen Stellen Sie die Funktion "Setbit" zum Setzen der Variablen "MotorOn" verwendet haben. Führen Sie dazu eine Suche nach "setbit motoron" durch.
- Sie möchten einen bestimmten Netzwerkkommentar finden, aber Sie wissen nur noch, dass die beiden Wörter "CPU" und "Cycle" darin enthalten sind. Starten Sie dazu eine Suche nach "CPU cycle".

Siehe auch

Grundlagen zur Suche (Seite 107)

Grundlagen zur Suche im Projekt (Seite 108)

Übersicht über den Sucheditor (Seite 110)

Im gesamten Projekt suchen (Seite 111)

Suchindex manuell neu erstellen (Seite 113)

Suche im Projekt aktivieren und deaktivieren (Seite 113)

4.3 Innerhalb eines Editors suchen und ersetzen

4.3.1 Grundlagen zur Suche in geöffneten Editoren

Einführung

Sie können eine Suche ausführen, die auf den aktuell im Arbeitsbereich geöffneten Editor beschränkt ist. Dabei können Sie die Suche folgendermaßen an Ihre Bedürfnisse anpassen:

- Sie können die Suche über Zusatzoptionen optimieren.
- Sie können den Suchbereich festlegen.
- Sie können die Suchrichtung festlegen.

Zusatzoptionen für die Suche

Sie können die Suche mithilfe der folgenden Zusatzoptionen verfeinern:

- Nur ganzes Wort suchen
  Bei der Suche werden ausschließlich ganze Wörter berücksichtig. Zusammengesetzte Wörter, die den Suchbegriff als Bestandteil enthalten, werden ignoriert.
- Groß-/Kleinschreibung
  Groß- und Kleinschreibung werden bei Ihrer Suche berücksichtigt.
- Suchen in untergeordneten Strukturen
  Bei der Suche werden auch Texte berücksichtigt, die in einem anderen Objekt enthalten sind.
- Suchen in ausgeblendeten Texten
  Bei der Suche werden auch Texte berücksichtigt, die einem anderen Text zwar zugeordnet, aber aktuell ausgeblendet sind.
- Platzhalterzeichen verwenden
  Geben Sie einen Stern als Platzhalter für beliebig viele Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Gerät" beginnen. Geben Sie also "Gerät*" in das Suchfeld ein. Wenn Sie nur ein einzelnes Zeichen auslassen möchten, geben Sie hingegen ein Fragezeichen als Platzhalter ein.

Welche Zusatzoptionen zur Verfügung stehen, ist von den installierten Produkten und dem geöffneten Editor abhängig.

Suchbereich und Suchrichtung

Sie haben die folgenden Möglichkeiten, den Suchbereich zu definieren:

- Gesamtes Dokument
  Unabhängig von der aktuellen Auswahl und Position wird der gesamte Editor durchsucht.
- Von der aktuellen Position
  Die Suche startet ab der aktuellen Position. Abhängig von der Suchrichtung werden alle Texte oberhalb oder links bzw. unterhalb oder rechts der aktuellen Position bei der Suche nicht berücksichtigt.
- Auswahl
  Der Suchbegriff wird nur innerhalb der aktuellen Selektion gesucht.

Unabhängig vom Suchbereich können Sie die Suchrichtung festlegen:

- Nach unten
  Die Suche im Editor erfolgt von oben nach unten oder von links nach rechts.
- Nach oben
  Die Suche im Editor erfolgt von unten nach oben oder von rechts nach links.

4.3.2 Übersicht über die Palette "Suchen und ersetzen"

Funktion der Palette "Suchen und ersetzen"

Die Palette "Suchen und ersetzen" innerhalb der Task Card "Aufgaben" ermöglicht Ihnen die Suche innerhalb eines geöffneten Editors. Sie enthält alle Optionen, die Sie für eine effiziente Suche benötigen.

Aufbau der Palette "Suchen und ersetzen"

Das folgende Bild zeigt die Komponenten der Palette "Suchen und ersetzen":

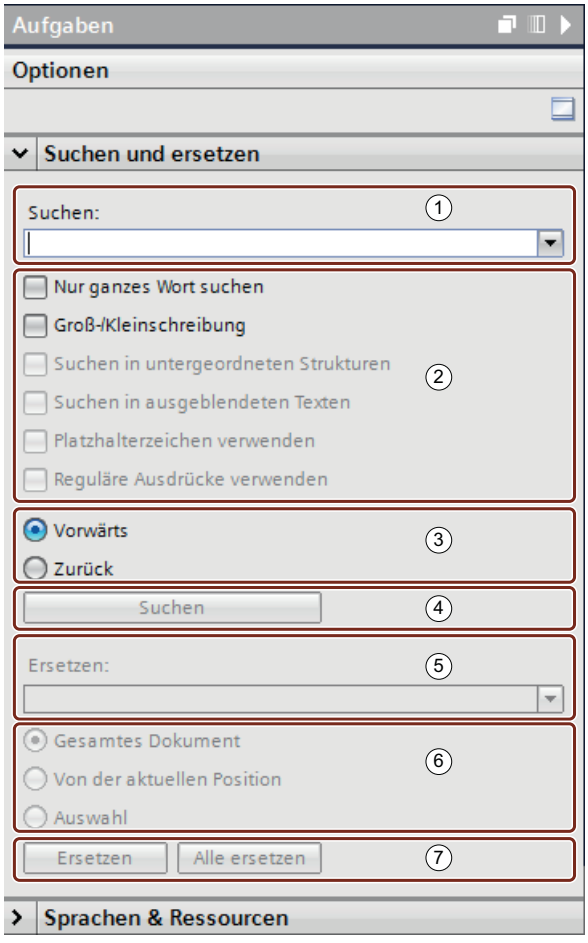

① Eingabe Suchbegriff

② Zusatzoptionen

③ Suchrichtung

④ Suche starten

⑤ Eingabe Ersatzbegriff

⑥ Ersatzbereich

⑦ Ersetzen

4.3.3 Im Editor suchen und ersetzen

Sie haben die Möglichkeit, innerhalb eines Editors nach Texten zu suchen oder diese zu ersetzen.

Suche starten

Um die Funktion "Suchen und ersetzen" zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Suchen und ersetzen" oder öffnen Sie in der Task Card "Aufgaben" die Palette "Suchen und ersetzen".
   Die Palette "Suchen und ersetzen" wird geöffnet.
2. Geben Sie einen Begriff in die Klappliste "Suchen" ein.
   Alternativ können Sie aus der Klappliste einen der letzten Suchbegriffe auswählen.
3. Aktivieren Sie die gewünschten Zusatzoptionen für die Suche.
4. Wählen Sie mithilfe der Optionsfelder den Ausgangspunkt für die Suche und die Suchrichtung.
5. Klicken Sie auf "Suchen".
   Der erste Suchtreffer wird im Editor markiert.
6. Um den nächsten Suchtreffer anzuzeigen, klicken Sie erneut auf "Suchen".
   Der nächste Suchtreffer wird im Editor markiert. Wiederholen Sie gegebenenfalls diesen Schritt so lange, bis Sie zum letzten Treffer gelangt sind.

Ersetzen des Suchbegriffs

Sie haben die Möglichkeit, einzelne Suchtreffer zu ersetzen oder automatisch alle gefundenen Texte ersetzen zu lassen, wenn der jeweilige Editor diese Funktionalität unterstützt. Um Begriffe zu ersetzen, gehen Sie folgendermaßen vor:

1. Geben Sie einen Begriff in die Klappliste "Suchen" ein.
   Alternativ wählen Sie aus der Klappliste einen der letzten Suchbegriffe aus.
2. Aktivieren Sie die gewünschten Zusatzoptionen für die Suche.
3. Klicken Sie auf die Schaltfläche "Suchen", um einen Suchlauf nach dem eingegebenen Begriff durchzuführen.
   Der erste Suchtreffer wird im Editor angezeigt.
4. Geben Sie den Text, durch den Sie den Suchtreffer ersetzen möchten, in die Klappliste "Ersetzen" ein.
   Alternativ wählen Sie aus der Klappliste einen der zuletzt eingegebenen Texte aus.
5. Um den markierten Suchtreffer durch den vorgegebenen Text zu ersetzen, klicken Sie auf die Schaltfläche "Ersetzen".
   Der gefundene Text wird ersetzt und der nächste Suchtreffer wird im Editor markiert. Wiederholen Sie diesen Schritt so lange, bis Sie alle gewünschten Suchtreffer ersetzt haben. Um zum nächsten Treffer zu springen, ohne das markierte Wort zu ersetzen, klicken Sie anstatt auf die Schaltfläche "Ersetzen" auf "Suchen".
6. Um alle Suchtreffer automatisch auf einmal zu ersetzen, klicken Sie auf die Schaltfläche "Alle ersetzen".

Siehe auch

Grundlagen zur Suche (Seite 107)

Im gesamten Projekt suchen (Seite 111)

Aktionen rückgängig machen und wiederholen 5

5.1 Grundlagen zum Rückgängig machen und Wiederholen von Aktionen

Funktion

Sie können durchgeführte Aktionen jederzeit rückgängig machen. Dazu wird jede Aktion, die Sie durchführen, in einem Aktionenstapel gespeichert. Beim Rückgängig machen wird der Stapel von oben nach unten abgearbeitet. Wenn Sie also eine Aktion rückgängig machen, die im Stapel weiter unten liegt, dann werden auch alle darüber liegenden Aktionen automatisch rückgängig gemacht.

Sie können rückgängig gemachte Aktionen so lange wiederholen, bis Sie eine neue Aktion ausführen. Wenn Sie eine neue Aktion ausführen, ist das Wiederholen rückgängig gemachter Aktionen nicht mehr möglich.

Besonderheiten beim Rückgängig machen

Es gibt einige Aktionen, die den Aktionenstapel leeren. Sie können diese Aktionen und auch die vor diesen Aktionen durchgeführten Aktionen nicht rückgängig machen. Folgende Aktionen leeren den Aktionenstapel:

- Speichern
- Projektverwaltung (neues Projekt anlegen, Projekt öffnen, Projekt schließen, Projekt löschen)
- Ungenutzte GSD-Dateien löschen
- Online-Verbindung herstellen

Beachten Sie, dass beim Herstellen einer Online-Verbindung der Aktionenstapel geleert wird. Das bedeutet, dass Sie nach dem Herstellen einer Online-Verbindung die offline durchgeführten Aktionen nicht mehr rückgängig machen können. Alle Aktionen, die Sie anschließend online durchführen, können Sie auch nach dem Trennen der Online-Verbindung rückgängig machen.

Anzeige des Aktionenstapels

Sobald Sie eine Aktion durchführen, die rückgängig gemacht werden kann, wird in der Funktionsleiste die Schaltfläche "Rückgängig" aktiv. Diese Schaltfläche ist zweigeteilt und über den Pfeil-nach-unten können Sie eine Klappliste öffnen. In dieser Klappliste werden Ihnen alle Aktionen des Aktionenstapels angezeigt, die Sie rückgängig machen können. Falls Sie Aktionen in einem anderen als dem aktuell angezeigten Editor ausgeführt hatten, wird auch der entsprechende Editor als Zwischenüberschrift angezeigt. Dadurch können Sie immer erkennen, an welcher Stelle sich das Rückgängig machen auswirkt. Die Zwischenüberschriften werden aus der Liste entfernt, wenn für den entsprechenden Editor keine Aktionen mehr rückgängig gemacht werden können.

Rückgängig gemachte Aktionen werden in den Aktionenstapel für das Wiederholen eingefügt. Hier können Sie die rückgängig gemachten Aktionen wiederholen. Die Anzeige der Aktionen, die Sie wiederholen können, ist analog zur Anzeige der Aktionen, die Sie rückgängig machen können.

Beispiel für das Rückgängig machen von Aktionen

Das folgende Bild zeigt das Rückgängig machen von Aktionen aus verschiedenen Editoren und Tabellen:

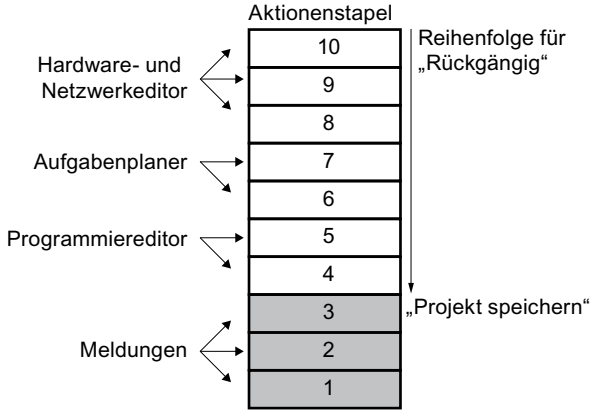

In diesem Beispiel können Sie die Aktionen 1 bis 3 nicht rückgängig machen, da das Projekt gespeichert wurde. Die Aktionen 4 bis 10 können Sie in Pfeilrichtung rückgängig machen. Das heißt, Sie müssen Aktion 10 zuerst rückgängig machen. Nachdem Sie Aktion 8 rückgängig gemacht haben, können Sie anschließend nicht die Aktion 5 rückgängig machen. Zunächst müssen Sie die Aktionen 7 und 6 rückgängig machen. Als Letztes können Sie Aktion 4 rückgängig machen. Sie haben aber die Möglichkeit mehrere Aktionen in einem Schritt rückgängig zu machen, in dem Sie eine untere Aktion des Aktionenstapels rückgängig machen. Alle darüber stehenden Aktionen werden automatisch rückgängig gemacht.

Das gleiche Prinzip gilt auch für das Wiederholen von Aktionen.

Siehe auch

Aktion rückgängig machen (Seite 125)

Aktion wiederholen (Seite 126)

5.2 Aktion rückgängig machen

Sie haben beim Rückgängig machen von Aktionen folgende Möglichkeiten:

- Nur die letzte Aktion rückgängig machen
  Es wird nur die zuletzt durchgeführte Aktion rückgängig gemacht.
- Eine beliebige Anzahl von Aktionen rückgängig machen
  Es werden mehrere Aktionen des Aktionenstapels in einem Schritt rückgängig gemacht.

Nur die letzte Aktion rückgängig machen

Um die zuletzt durchgeführte Aktion rückgängig zu machen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Rückgängig".
   - Falls die Aktion nicht in dem aktuell angezeigten Editor durchgeführt wurde, wird eine Sicherheitsabfrage eingeblendet.
   - Falls für das Rückgängig machen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
2. Bestätigen Sie ggf. die Sicherheitsabfrage mit "Ja".
3. Geben Sie ggf. das Passwort ein.
   Der Editor, in dem die Aktion durchgeführt wurde, wird angezeigt und die Aktion wird rückgängig gemacht.

Eine beliebige Anzahl von Aktionen rückgängig machen

Um mehrere Aktionen des Aktionenstapels in einem Schritt rückgängig zu machen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Rückgängig" auf den Pfeil-nach-unten.
   Eine Klappliste wird geöffnet, in der alle Aktionen aufgeführt werden, die Sie rückgängig machen können. Aktionen anderer Editoren werden durch die Editornamen als Zwischenüberschriften gekennzeichnet.
2. Klicken Sie auf die Aktion, die Sie rückgängig machen möchten.
   Die gewählte Aktion und alle über der gewählten Aktion stehenden Aktionen werden rückgängig gemacht. Falls für das Rückgängig machen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
3. Geben Sie ggf. die erforderlichen Passwörter ein.
   Die Editoren, in denen die Aktionen durchgeführt wurden, werden angezeigt und die Aktionen werden rückgängig gemacht.

Siehe auch

Grundlagen zum Rückgängig machen und Wiederholen von Aktionen (Seite 123)

Aktion wiederholen (Seite 126)

5.3 Aktion wiederholen

Sie haben die Möglichkeit, eine rückgängig gemachte Aktion zu wiederholen, sodass Sie den Stand vor der Aktion "Rückgängig" erhalten. Dies ist jedoch nur so lange möglich, bis Sie eine neue Aktion durchführen. Beim Wiederholen von Aktionen haben Sie folgende Möglichkeiten:

- Nur die zuletzt rückgängig gemachte Aktion wiederholen
  Es wird nur die zuletzt rückgängig gemachte Aktion wiederholt.
- Eine beliebige Anzahl von rückgängig gemachten Aktionen wiederholen
  Es werden mehrere rückgängig gemachten Aktionen des Aktionenstapels in einem Schritt wiederholt.

Nur die zuletzt rückgängig gemachte Aktion wiederholen

Um die zuletzt rückgängig gemachte Aktion zu wiederholen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Wiederholen".
   - Falls die Aktion nicht in dem aktuell angezeigten Editor wiederholt wird, wird eine Sicherheitsabfrage eingeblendet.
   - Falls für das Wiederholen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
2. Bestätigen Sie ggf. die Sicherheitsabfrage mit "Ja".
3. Geben Sie ggf. das Passwort ein.
   Der Editor, in dem die Aktion rückgängig gemacht wurde, wird angezeigt und die Aktion wird wiederholt.

Eine beliebige Anzahl von rückgängig gemachten Aktionen wiederholen

Um mehrere rückgängig gemachte Aktionen des Aktionenstapels in einem Schritt zu wiederholen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Wiederholen" auf den Pfeil-nach-unten.
   Eine Klappliste wird geöffnet, in der alle Aktionen aufgeführt werden, die Sie rückgängig gemacht haben. Aktionen anderer Editoren werden durch die Editornamen als Zwischenüberschriften gekennzeichnet.
2. Klicken Sie auf die Aktion, die Sie wiederholen möchten.
   Die gewählte Aktion und alle über der gewählten Aktion stehenden Aktionen werden wiederholt. Falls für das Wiederholen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
3. Geben Sie ggf. die erforderlichen Passwörter ein.
   Die Editoren, in denen die Aktionen rückgängig gemacht wurden, werden angezeigt und die Aktionen werden wiederholt.

Siehe auch

Grundlagen zum Rückgängig machen und Wiederholen von Aktionen (Seite 123)

Aktion rückgängig machen (Seite 125)

Index

A

Aktionen
- Grundlagen zum Rückgängig machen, 123
- Grundlagen zum Wiederholen von Aktionen, 123
- rückgängig machen, 125
- wiederholen, 126

Arbeitsbereich
- Abgedockte Elemente einbetten, 34
- Anordnung von Editoren und Tabellen sichern, 53
- Elemente abdocken, 34
- Elemente maximieren, 37
- Elemente minimieren, 37
- Funktion, 30
- Gruppierte Elemente verwenden, 35
- maximieren, 32
- minimieren, 32
- teilen, 33
- zwischen den Elementen wechseln, 37

B

Bedienoberfläche
- Ansichten, 21
- Arbeitsbereich, 30
- Arbeitsbereich maximieren, 32
- Arbeitsbereich minimieren, 32
- Detailansicht, 43
- Inspektorfenster, 38
- Palette "Referenzprojekte", 42
- Portalansicht, 21
- Projektansicht, 23
- Projektnavigation, 26
- Task Card, 40
- Übersichtsfenster, 46

Bedienoberfläche sichern, 50

Bedienoberfläche speichern
- Fensterlayout speichern, 50

Bedienoberflächenlayout zurücksetzen, 53

Benutzeroberfläche sichern
- Anordnung in Editoren sichern, 50

Benutzerrechte erwerben
- Am Betriebssystem mit Administratorrechten anmelden, 76
- Mit der Windows Benutzerkontensteuerung, 76

Beschränkungen durch Benutzerrechte, 75, 76
- erkennen, 75

Bildschirmtastatur verwenden, 70

D

Detailansicht, 43

Dokumentation individuell erstellen, 103

E

Einstellungen
- Allgemein, 10, 13, 14, 16, 17
- ändern, 18
- Online- und Diagnosefunktionen, 17

F

Fensterlayout laden
- Laden über Schnellzugriff, 51
- Weitere Fensterlayouts laden, 51

Fensterlayout speichern, 50

Fensterlayout wiederherstellen, 52

Fensterlayouts
- Fensterlayouts löschen, 52
- Reihenfolge ändern, 52

H

Handbücher zusammenstellen, 103

Hilfe
- durchsuchen, 91
- Hilfe zu Meldungen, 84
- Hilfethemen drucken, 101
- Kennzeichnung von Hilfethemen, 82
- Komponenten des Informationssystems, 79
- Navigation, 87
- öffnen, 86, 87
- Roll-out, 83
- Themen aus Favoriten aufrufen, 95
- Themen aus Favoriten löschen, 96
- Themen in Favoriten speichern, 95
- Tooltip, 84
- Tooltip-Kaskaden automatisch aufklappen, 102
- Volltextsuche, 91

I

Informationssystem
- Hilfe zu Meldungen, 84
- Komponenten des Informationssystems, 79
- Roll-out, 83
- Tooltip, 84

Inspektorfenster
- Aufbau, 38
- Automatisch reduzieren, 32

M

MDM, 103
My Documentation Manager, 103

P

Palettenmodus, 42
Portalansicht, 21
Programmiereditor
- Allgemeine Einstellungen, 10, 13, 14, 17

Projektansicht, 23
Projektnavigation
- Automatisch reduzieren, 32
- Funktion, 26
- Zusätzliche Spalten einblenden, 30

R

Rückgängig machen von Aktionen
- Grundlagen zum Rückgängig machen, 123
- letzte Aktion rückgängig machen, 125
- mehrere Aktionen rückgängig machen, 125

S

Sicherheitshinweise, 104
Suche
- Beispiele zur Suche im Projekt, 114
- Ersetzen von Suchbegriffen im Editor, 121
- Grundlagen zur Suche, 107
- Grundlagen zur Suche im Projekt, 108
- Im Editor suchen und ersetzen, 121
- Im gesamten Projekt suchen, 112
- Palette Suchen und Ersetzen, 119
- Sucheditor, 110
- Zusatzoptionen für die Suche im Editor, 118

T

Tabellenanordnung sichern, 50
Task Card
- Automatisch reduzieren, 32
- Funktion, 40
- Palettenmodus ändern, 42

Tastaturbedienung
- Anpassen von Editoren, 65
- Editieren von Texten, 67
- Fenster, 62
- Grundfunktionen des TIA Portals, 60
- Navigation im TIA Portal, 63
- Objekte bearbeiten, 66
- Objekte selektieren, 66
- Online-Funktionen, 70
- Projektbearbeitung, 61
- Tabellen, 68

TIA Portal
- beenden, 9
- starten, 9

U

Übersichtsfenster, 46
- Alle Bausteine anzeigen, 49
- Detailansicht sortieren, 49
- Objekte vergleichen, 48
- Spalten einblenden und ausblenden, 49

W

Wiederholen von Aktionen
- Aktionen wiederholen, 126
- Grundlagen zum Wiederholen von Aktionen, 123

SIEMENS

Projekte anlegen und verwalten 1

Kompatibilität von Projekten 2

Projekte archivieren und dearchivieren 3

Referenzprojekte verwenden 4

TIA Portal Teamcenter Gateway 5

Projekte bearbeiten

Funktionshandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Projekte anlegen und verwalten....7
1.1 Grundlagen zu Projekten....7
1.2 Projekt erstellen....9
1.3 Projekte öffnen....10
1.4 Projekte speichern....12
1.5 Projekte schließen....13
1.6 Projekte aus der Projektliste entfernen....14
1.7 Projekte löschen....15
1.8 Eigenschaften des Projekts anzeigen....16
1.9 Protokolle verwenden....18
1.10 Sicherheit von Industrieanlagen....19
1.11 CAx-Daten exportieren und importieren....21
2 Kompatibilität von Projekten....25
2.1 Kompatibilität zwischen TIA Portal Versionen....25
2.2 Mit optionaler Software erstellte Projekte verwenden....26
2.3 Projekte hochrüsten....27
2.3.1 Allgemeines zum Hochrüsten von Projekten....27
2.3.2 Projekte hochrüsten....28
2.3.3 Multiuser Engineering Projekte hochrüsten....30
2.3.4 Know-how-geschützte Bausteine hochrüsten....31
2.4 Kompatibilität mit dem Anlagenaufbau....32
3 Projekte archivieren und dearchivieren....35
3.1 Arbeiten mit Projektarchiven....35
3.2 Projektarchiv erstellen....36
3.3 Komprimiertes Projekt dearchivieren....38
4 Referenzprojekte verwenden....39
4.1 Grundlagen zu Referenzprojekten....39
4.2 Referenzprojekt öffnen und schließen....40
4.3 Referenzprojekte vergleichen....41
5 TIA Portal Teamcenter Gateway....43
5.1 Einführung in das TIA Portal Teamcenter Gateway....43
5.1.1 Grundlagen zum Arbeiten mit dem TIA Portal Teamcenter Gateway....43
5.1.2 Einsatzmöglichkeiten von TIA Portal Teamcenter Gateway....44

5.2 Systemvoraussetzungen....45
5.2.1 Systemvoraussetzungen für das TIA Portal Teamcenter Gateway....45

5.3 TIA Portal Teamcenter Gateway installieren und deinstallieren....47
5.3.1 Hinweise zur Installation des TIA Portal Teamcenter Gateway....47
5.3.2 Hinweise zur Kompatibilität des TIA Portal Teamcenter Gateway....48
5.3.3 Parallelinstallationen....48
5.3.4 TIA Portal Teamcenter Gateway Installation starten....49
5.3.5 TIA Portal Teamcenter Gateway installierte Software anzeigen....49
5.3.6 TIA Portal Teamcenter Gateway Installation prüfen....50
5.3.7 TIA Portal Teamcenter Gateway modifizieren oder hochrüsten....51
5.3.8 TIA Portal Teamcenter Gateway Installation reparieren....53
5.3.9 TIA Portal Teamcenter Gateway Deinstallation starten....54

5.4 Lizenzen für das TIA Portal Teamcenter Gateway....56
5.4.1 Lizenzen....56
5.4.2 Hinweise zu Lizenzen....56

5.5 Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen....58
5.5.1 Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen....58

5.6 Grundlagen zum Arbeiten mit TIA Portal Teamcenter Gateway....59

5.7 TIA Portal Teamcenter Gateway Cache Pfad festlegen....60

5.8 Einchecken beim Schließen des Projekts/der Bibliothek bestätigen....61

5.9 Verbindung zwischen dem TIA Portal und Teamcenter herstellen und trennen....62
5.9.1 Verbindung zwischen dem TIA Portal und Teamcenter herstellen....62
5.9.2 Verbindung zwischen dem TIA Portal und Teamcenter trennen....64
5.9.3 Verbindungsstatus zwischen TIA Portal und Teamcenter anzeigen....65

5.10 Unterscheidung zwischen TIA Portal-Projekten und globalen Bibliotheken....67
5.10.1 Unterschiedliche Datasettypen in eine neue Revision kopieren....67

5.11 TIA Portal-Projekte mit dem TIA Portal Teamcenter Gateway verwalten....69
5.11.1 TIA Portal-Projekt als neues Element in Teamcenter speichern....69
5.11.1.1 Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)....72
5.11.1.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)....72
5.11.1.3 Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)....73
5.11.1.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)....74
5.11.1.5 Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)....75
5.11.2 TIA Portal-Projekt auschecken....76
5.11.3 TIA Portal-Projekt einchecken....78
5.11.4 Änderungen an einem mit Teamcenter verbunden TIA Portal-Projekt speichern....79
5.11.5 Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen....81
5.11.5.1 Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)....83
5.11.5.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)....83
5.11.5.3 Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)....84

5.11.5.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)....84
5.11.5.5 Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)....85
5.11.6 Auschecken eines TIA Portal-Projekts rückgängig machen....87
5.11.7 TIA Portal-Projekt exportieren....88
5.11.7.1 Register "Info" im Dialog "Export..." - (TIA Portal Projekte)....89
5.11.8 TIA Portal-Projekt aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern....89
5.11.8.1 Prüfungen bei Verwendung der Funktion "Speichern zu Element..." - (TIA Portal-Projekte)....89
5.11.8.2 TIA Portal-Projekt aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern....90
5.11.8.3 Aufbau Dialog "Speichern zu Element..."....93
5.11.9 TIA Portal-Projekt das im Teamcenter-Cache Ordner gespeichert ist wieder in das zugehörige Element speichern....94
5.11.10 Ein mit Teamcenter Gateway verbundenes TIA Portal-Projekt als Referenzprojekt öffnen....95

5.12 Arbeitsabläufe zwischen dem TIA Portal Teamcenter Gateway und TIA Portal Multiuser-Engineering - TIA Portal-Projekte....98
5.12.1 Ein mit Teamcenter verbundenes TIA Portal-Projekt als Multiuser-Serverprojekt speichern....98
5.12.1.1 Serverprojekt-Ansicht eines Multiuser-Serverprojekts....99
5.12.1.2 Lokale Session eines spezifischen Multiuser-Serverprojekt erstellen....99
5.12.1.3 Beispiel für die Behandlung von Inkonsistenzen zwischen TIA Portal-Projekten in Teamcenter und im Multiuser-Server....100
5.12.2 Auschecken (nur gesperrte)....101
5.12.3 Einchecken (nur entsperrte)....103
5.12.4 Änderungen an einem Multiuser-Serverprojekt nach Teamcenter speichern....103
5.12.4.1 Beispiele für den Datenabgleich zwischen den verschiedenen Dataset Versionen im Teamcenter Cache und im Multiuser-Serverprojekts 107
5.12.5 Multiuser-Serverprojekt als neues Element in Teamcenter speichern....110
5.12.6 Multiuser-Serverprojekt als neue Revision in Teamcenter speichern....115
5.12.7 TIA Portal-Projekt (Multiuser-Serverprojekt) aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern....119
5.12.7.1 TIA Portal-Projekt in ein in Teamcenter vorhandenes Element in der Serverprojekt-Ansicht speichern....119
5.12.8 Ein mit Teamcenter Gateway verbundenes und dem Multiuser-Server hinzugefügtes TIA Portal-Projekt als Referezprojekt öffnen....122
5.12.8.1 Ein mit Teamcenter Gateway verbundenes und dem Multiuser-Server hinzugefügtes TIA Portal-Projekt als Referezprojekt öffnen....122

5.13 Globale Bibliotheken mit dem TIA Portal Teamcenter Gateway verwalten....125
5.13.1 Globale Bibliothek als neues Element in Teamcenter speichern....125
5.13.1.1 Register "Allgemein" im Dialog "Speichern als neues Element" (globale Bibliotheken). 128
5.13.1.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" (globale Bibliotheken)....128
5.13.1.3 Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (globale Bibliothek)....129
5.13.1.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (globale Bibliothek)....130
5.13.1.5 Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (globale Bibliotheken)....131

5.13.2 Globale Bibliothek auschecken....132
5.13.3 Globale Bibliothek einchecken....134
5.13.4 Änderungen an einer mit Teamcenter verbunden globalen Bibliothek speichern....135
5.13.5 Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen....136
5.13.5.1 Register "Allgemein" im Dialog "Speichern als neue Revision" (globale Bibliotheken)....138
5.13.5.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" (globale Bibliotheken)....139
5.13.5.3 Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (globale Bibliothek)....139
5.13.5.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (globale Bibliothek)....140
5.13.5.5 Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)....141
5.13.6 Auschecken einer globalen Bibliothek rückgängig machen....142
5.13.7 Globale Bibliothek exportieren....143
5.13.7.1 Register "Info" im Dialog "Export" - (globale Bibliothek)....144
5.13.8 Globale Bibliothek aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern....145
5.13.8.1 Prüfungen bei der Verwendung der Funktion "Speichern zu Element..." - (Globale Bibliotheken)....145
5.13.8.2 Globale Bibliothek aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern....146
5.13.8.3 Aufbau Dialog "Speichern zu Element...."....149
5.13.9 Globale Bibliothek die im Teamcenter-Cache Ordner gespeichert ist wieder in das zugehörige Element speichern....150

5.14 TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen....152
5.14.1 Öffnen von TIA Portal-Projekten aus dem Suchergebnis....156
5.14.1.1 Öffnen von TIA Portal-Projekten aus dem Suchergebnis....156
5.14.1.2 Aufbau des Registers "Eigenschaften" im Dialog "TIA Portal-Projekt aus Teamcenter öffnen"....158
5.14.2 Öffnen von TIA Portal-Projekten aus der TIA Portal-Projektliste....160
5.14.3 Öffnen von globalen Bibliotheken aus dem Suchergebnis....161
5.14.3.1 Öffnen von globalen Bibliotheken aus dem Suchergebnis....161
5.14.3.2 Aufbau der Register "Eigenschaften" im Dialog "globale Bibliothek aus Teamcenter öffnen"....163
5.14.4 Dataset Status im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" anzeigen....165
5.14.4.1 Anzeigen des Dataset Status....165
5.14.4.2 Anzeigen des Element-Revision Status....165
5.14.4.3 Anzeigen des Cache Status....166

5.15 Manueller Vergleich von zwei TIA Portal-Projekten (Dataset Versionen) mit derselben Revision....167

5.16 Symbole....168
5.16.1 Objektstatus im Dialog Projekt aus Teamcenter öffnen....168
5.16.2 Objektstatus im Dialog Globale Bibliothek aus Teamcenter öffnen....169
5.16.3 Symbole die den Verbindungsstatus zwischen dem TIA Portal und Teamcenter anzeigen....170

Index....171

Projekte anlegen und verwalten 1

1.1 Grundlagen zu Projekten

Projekte enthalten Daten und Programme, die bei der Erstellung einer Automatisierungslösung entstehen. Die in einem Projekt zusammengefassten Daten umfassen zum Beispiel:

- Konfigurationsdaten über den Hardware-Aufbau und Parametrierungsdaten für Baugruppen
- Projektierungsdaten für die Kommunikation über Netze
- Projektierungsdaten für die Geräte
- Protokolle für wichtige Ereignisse im Lebenszyklus des Projekts

Projektversion

Die Projektversion definiert, mit welcher Version des TIA Portals ein Projekt kompatibel ist. Ein Projekt, das mit der aktuellen Version des TIA Portals kompatibel ist, liegt in der Projektversion V15.1 des TIA Portals vor.

Die Projektversion eines gerade geöffneten Projekts sehen Sie in den Eigenschaften des Projekts. Wählen Sie dazu den Eintrag "Softwareprodukte im Projekt" im Register "Allgemein" des Eigenschaftsdialogs. In der Tabelle sehen Sie alle Produkte und deren Versionen, die zum Bearbeiten des geöffneten Projekts notwendig sind.

Projekthierarchie

Daten werden in einem Projekt in Form von Objekten abgelegt. Die Objekte sind innerhalb eines Projekts in einer Baumstruktur (Projekthierarchie) angeordnet.

Die Basis der Projekthierarchie bilden die Geräte und Stationen und deren zugehörige Projektierungsdaten und Programme.

Zusätzlich werden in der Projektnavigation z. B. gemeinsame Daten des Projekts und die Online-Zugänge dargestellt.

Basis-Integritätsprüfung

Um Veränderungen an den wesentlichen benutzerdefinierten Projektdaten (z. B. Bausteine, Hardware-Projektierung) zu erkennen, verfügt die Engineering-Software über einen Basis-Schutzmechanismus. Ursachen für Veränderungen an den Projektdaten können beispielsweise eine Manipulation außerhalb der Engineering-Software oder Datenträgerdefekte sein.

Die Prüfung der Projektierungsdaten erfolgt auf Anforderung optional, da diese je nach Umfang und Anwendungsfall zeitaufwändig sein kann, insbesondere wenn Sie Multiuser Engineering nutzen. In den Einstellungen der Engineering-Software ändern Sie Voreinstellungen zur Prüfung.

Wenn bei einer Prüfung eine Veränderung festgestellt wird, kontaktieren Sie den Siemens Customer Support. Sie können mit den Projektdaten weiterarbeiten. Die Weiterarbeit geschieht auf eigene Gefahr.

Hinweis

Verringerte Arbeitsgeschwindigkeit bei aktivierter Basis-Integritätsprüfung

Wenn Sie die Basis-Integritätsprüfung in den Einstellungen der Engineering-Software aktivieren, dann wird die Arbeitsgeschwindigkeit der Engineering-Software herabgesetzt.

Projektschutz

Sie können für ein Projekt die Benutzerverwaltung aktivieren und dadurch Ihr Projekt schützen. Wenn für ein Projekt der Schutz aktiv ist, können sich nur noch berechtigte Benutzer im Projekt anmelden und damit arbeiten.

Siehe auch: Grundlagen zur Benutzerverwaltung im TIA Portal

Siehe auch

Protokolle verwenden (Seite 18)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

1.2 Projekt erstellen

Vorgehen

Um ein neues Projekt zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Neu".
   Der Dialog "Neues Projekt erstellen" wird geöffnet.
2. Geben Sie einen Projektnamen und Pfad ein oder übernehmen Sie die vorgeschlagenen Daten.
3. Klicken Sie auf die Schaltfläche "Erstellen".

Ergebnis

Das neue Projekt wird erstellt und in der Projektnavigation angezeigt.

Siehe auch

Kompatibilität zwischen TIA Portal Versionen (Seite 25)

Grundlagen zu Projekten (Seite 7)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

1.3 Projekte öffnen

Projekte des TIA Portals erkennen Sie an der Dateinamenserweiterung ".ap[Versionsnummer]". Projekte der aktuellen Version besitzen die Dateinamenserweiterung ".ap15_1". Projekte in der Projektversion V15.1 lassen sich direkt öffnen. Projekte mit der Projektversion V13 SP1, V14, V14 SP1 und V15 rüsten Sie vor dem Öffnen hoch.

Vorgehen

Um ein bestehendes Projekt zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".
   Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird angezeigt.
2. Selektieren Sie ein Projekt aus der Liste und klicken Sie auf "Öffnen".
3. Falls das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.
4. Um zu überprüfen, ob das Projekt unzulässig verändert wurde, aktivieren Sie das Optionskästchen "Basis-Integritätsprüfung aktivieren". Die Basis-Integritätsprüfung kann einige Zeit in Anspruch nehmen.
5. Klicken Sie auf die Schaltfläche "Öffnen".
   Falls für das selektierte Projekt ein Schutz aktiv ist, wird der Dialog "Anmelden" geöffnet. Führen Sie dann die folgenden Schritte aus:
   - Wählen Sie den Benutzertyp aus.
   - Geben Sie Ihren Benutzernamen ein.
   - Geben Sie Ihr Passwort ein.
   - Klicken Sie auf "OK".

   Das Projekt wird geöffnet, wenn entweder kein Projektschutz aktiv ist oder wenn Sie sich mit einem Benutzerkonto angemeldet haben, das über die entsprechenden Rechte verfügt. Weitere Informationen zur Benutzerverwaltung finden Sie unter "Grundlagen zur Benutzerverwaltung im TIA Portal".
   Wenn Sie ein Projekt aus der Version V13 SP1, V14, V14 SP1 oder V15 des TIA Portals gewählt haben, wird der Dialog "Projekt hochrüsten" geöffnet. Weitere Informationen zum Hochrüsten von Projekten finden Sie unter "Projekte hochrüsten (Seite 28)".

Prüfung auf fehlende Software-Komponenten

Beim Öffnen von bereits bestehenden Projekten wird automatisch geprüft, ob für alle im Projekt verwendeten Baugruppen die entsprechende Software installiert ist. Versuchen Sie ein Projekt mit Baugruppen zu öffnen, die vom aktuellen Installationsumfang des TIA Portals nicht unterstützt werden, so erscheint bereits beim Öffnen des Projekts ein Hinweis auf die fehlenden Software-Komponenten. Sind die Software-Komponenten nicht zwingend für das Öffnen des Projekts erforderlich, lässt sich das Projekt dennoch öffnen.

Wenn sich im TIA Portal ein Projekt aufgrund fehlender Software-Komponenten nicht öffnen lässt, können Sie diese im Siemens Industry Online Support als Trial-Software herunterladen

und installieren. Anschließend können Sie das Projekt öffnen. Beachten Sie, dass sich die Trial-Software nur für einen begrenzten Zeitraum nutzen lässt.

Über den folgenden Link können Sie die Seite im Siemens Industry Online Support öffnen, auf der Ihnen die verschiedenen Softwareprodukte als Trial-Software zur Verfügung stehen: www.siemens.de/tia-portal-trial (www.siemens.de/tia-portal-trial)

Verhalten bei fehlenden Software-Komponenten

Nicht unterstützte Baugruppen eines Projekts werden folgendermaßen behandelt:

- Anzeige der Baugruppen an der Benutzeroberfläche
  - Die nicht unterstützten Baugruppen werden in der Projektnavigation mit all ihren unterlagerten Objekten angezeigt, die Baugruppen selbst können jedoch nicht in Editoren oder im Inspektorfenster bearbeitet werden. Falls möglich, wird eine Ersatzbaugruppe verwendet, die am besten zur ursprünglichen Baugruppe passt. Ersatzbaugruppen sind mit einem Ausrufezeichen-Symbol gekennzeichnet.
  - Die Anzeige von Eigenschaften in Tabellen ist eingeschränkt. Dies betrifft insbesondere die Darstellung von Netzparametern, wie beispielsweise der IP-Adresse.
- Funktionale Einschränkungen
  - Nicht unterstützte Baugruppen können nicht gedruckt und nicht übersetzt werden.
  - Es ist nicht möglich, eine Online-Verbindung zu den Baugruppen aufzubauen. Somit ist ein Download ebenfalls nicht möglich.
  - Um den Gerätetyp zu ändern, müssen Sie das Gerät zunächst löschen und neu einfügen. Die Funktion "Gerätetyp ändern" wird nicht unterstützt.
  - Das Kopieren und Einfügen von unterlagerten Objekten, wie z. B. Bausteinen, ist möglich, das Kopieren und Einfügen des Geräts selbst ist nicht möglich.
  - Das Ändern der Netzwerkkonfiguration innerhalb der Netzsicht ist mit Ersatzbaugruppen nicht möglich.
  - Querverweise können angezeigt werden. Die Querverweise geben jedoch nur den zuletzt im Projekt gespeicherten Zustand wider, da ein Online-Abgleich mit der Original-Baugruppe nicht durchgeführt werden kann.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

1.4 Projekte speichern

Sie können ein Projekt zu jedem beliebigen Zeitpunkt entweder unter dem gleichen oder einem anderen Namen speichern. Das Speichern ist auch dann möglich, wenn das Projekt noch fehlerhafte Elemente enthält.

Projekt speichern

Um ein Projekt zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern".
   Alle Änderungen im Projekt werden unter dem aktuellen Projektnamen gespeichert.

Projekt speichern unter

Um das Projekt unter einem anderen Namen zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern unter".
   Der Dialog "Aktuelles Projekt speichern unter" wird geöffnet.
2. Wählen Sie im Feld "Speichern in" den gewünschten Projektordner.
3. Geben Sie im Feld "Dateiname" den neuen Projektnamen ein.
4. Bestätigen Sie Ihre Eingabe mit "Speichern".
   Das Projekt wird unter dem neuen Namen gespeichert und geöffnet.

Hinweis

Aktionen rückgängig machen

Beachten Sie, dass Sie keine Aktionen rückgängig machen können, nachdem Sie das Projekt gespeichert haben.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

1.5 Projekte schließen

Vorgehen

Um ein Projekt zu schließen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Schließen".
   Falls Sie seit dem letzten Speichern Änderungen am Projekt vorgenommen haben, wird eine Meldung eingeblendet.
2. Geben Sie an, ob die Änderungen gespeichert werden sollen.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

1.6 Projekte aus der Projektliste entfernen

Sie können Projekte aus der Liste der zuletzt verwendeten Projekte entfernen. Die Projektdaten bleiben dabei auf dem Speichermedium erhalten.

Vorgehen

Um ein Projekt aus der Liste der zuletzt verwendeten Projekte zu entfernen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Projekt löschen".
   Der Dialog "Projekt löschen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird darin angezeigt.
2. Selektieren Sie ein Projekt aus der Liste.
3. Klicken Sie auf die Schaltfläche "Entfernen".
4. Bestätigen Sie die Sicherheitsabfrage mit "Ja", um das Projekt aus der Liste zu entfernen.

Ergebnis

Das Projekt wird nicht mehr in der Liste der zuletzt verwendeten Projekte angezeigt. Wenn Sie das Projekt erneut öffnen, wird es wieder in die Liste aufgenommen.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte löschen (Seite 15)

1.7 Projekte löschen

Mit dem Löschen werden die gesamten Projektdaten vom Speichermedium entfernt.

Voraussetzung

Das zu löschende Projekt ist nicht geöffnet.

Vorgehen

Um ein bestehendes Projekt zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Projekt löschen".
   Der Dialog "Projekt löschen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird darin angezeigt.
2. Selektieren Sie ein Projekt aus der Liste.
   Wenn das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.
3. Klicken Sie auf die Schaltfläche "Löschen".
4. Bestätigen Sie die Sicherheitsabfrage mit "Ja", um das Projekt endgültig zu löschen.

Ergebnis

Der komplette Projektordner wird aus dem Dateisystem gelöscht.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

1.8 Eigenschaften des Projekts anzeigen

Sie können sich die Eigenschaften eines Projekts anzeigen lassen. Im Register "Allgemein" finden Sie in der Bereichsnavigation die folgenden Projekteigenschaften:

- Meta-Daten zum Projekt
  Hier erhalten Sie beispielsweise Informationen zum Erstellzeitpunkt, zum Autor, den Speicherpfad, der Projektgröße, Hinweise zum Copyright, zu den verwendeten Projektsprachen, usw. Viele der Eigenschaften sind änderbar.
- Projektverlauf
  Der Projektverlauf enthält eine Übersicht mit wichtigen Ereignissen im Lebenszyklus eines Projekts. Hier können Sie z. B. sehen, mit welcher Version des TIA Portals ein Projekt erstellt wurde und ob es zwischenzeitlich hochgerüstet wurde. Ist ein Projekt beispielsweise durch eine Migration entstanden, wird dies in der Projektverlaufstabelle ebenfalls mit dem Datum und der Uhrzeit des Migrationszeitpunkts angezeigt. Wenn zu einem Ereignis ein Protokoll angelegt wurde, können Sie außerdem das Protokoll direkt aufrufen.
- Support-Packages im Projekt
  Es wird eine Übersicht zusätzlicher Software angezeigt, die notwendig ist, um alle Geräte des Projekts bearbeiten zu können. Außerdem werden installierte GSD-Dateien (Gerätebeschreibungsdateien für weitere Geräte im Hardware-Katalog) aufgelistet.
- Software-Produkte im Projekt
  Sie können eine Übersicht aller installierten und für das Projekt notwendigen Software-Produkte anzeigen lassen. In der Spalte "Version" sehen Sie, in welcher Projektversion der jeweiligen Software das Projekt gespeichert ist.

Im Register "Schutz" legen Sie fest, dass Bausteine nach dem Übersetzen mit SIMATIC S7-PLCSIM Advanced simuliert werden können. Aktivieren Sie dazu das Optionskästchen "Beim Übersetzen von Bausteinen Simulierbarkeit unterstützen". Standardmäßig ist die Funktion deaktiviert.

Vorgehen

Um die Projekteigenschaften anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das geöffnete Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Eigenschaften".
   Der Dialog mit den Eigenschaften des Projekts wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die gewünschten Projekteigenschaften aus, die angezeigt werden sollen.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)
Projekte aus der Projektliste entfernen (Seite 14)
Projekte löschen (Seite 15)

1.9 Protokolle verwenden

Bei einigen Operationen innerhalb des TIA Portals werden im Hintergrund automatisch Protokolle angelegt, die Änderungen am Projekt dokumentieren. Protokolle werden beispielsweise bei der Migration von Projekten und Programmen oder der Aktualisierung von Instanzen aus der Bibliothek automatisch angelegt.

Protokolle werden in der Projektnavigation im Ordner "Gemeinsame Daten" angezeigt. Sie werden in Zip-Dateien zusammen mit dem Projekt im Projektordner abgespeichert und sind daher unabhängig vom verwendeten PG/PC lesbar, sobald Sie das Projekt geöffnet haben. Das Protokoll lässt sich nach Fehlern, Warnungen und Informationen filtern.

Neben der Anzeige im TIA Portal ist es möglich, die Protokolle auszudrucken.

Protokolle anzeigen

Um ein Protokoll aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Gemeinsame Daten > Protokolle" in der Projektnavigation.
2. Doppelklicken Sie das gewünschte Protokoll in der Liste.
   Die Inhalte des Protokolls werden im Arbeitsbereich angezeigt.
3. Optional: Um eine bestimmte Kategorie von Meldungen ein- oder auszublenden, aktivieren oder deaktivieren Sie in der Funktionsleiste das Symbol für "Fehler", "Warnungen" oder "Informationen".

Protokoll löschen

Um ein Protokoll zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Protokoll in der Projektnavigation.
2. Drücken Sie die Taste <Entf>.
   Das selektierte Protokoll wird aus dem Projektverzeichnis gelöscht und aus der Projektnavigation entfernt.

1.10 Sicherheit von Industrieanlagen

Die voranschreitende Digitalisierung der industriellen Automatisierung bringt neben zahlreichen Vorteilen auch eine erhöhte Verwundbarkeit von Industrieanlagen mit sich. Das TIA Portal und die SIMATIC-Produkte bringen bereits etliche Funktionen mit, die einen sicheren Betrieb Ihrer Anlage gewährleisten, zum Beispiel:

- Know-how-Schutz von Bausteinen
- Basis-Integritätsschutz für Projekte, um Projekte vor Manipulationen zu schützen
- Die Schutz- und Security-Funktionen von CPUs

Weitergehende Informationen und die Möglichkeit Sicherheitsprobleme zu melden, finden Sie im Internet unter folgendem Link:

Whitepapers und weiterführende Informationen (https://www.industry.siemens.com/topics/global/de/industrial-security/support/Seiten/whitepaper.aspx)

Industrial Security

Industrial Security geht über die Sicherheitsvorschriften und Sicherheitsmaßnahmen in Büroumgebungen hinaus. Industrial Security ist ein ganzheitlicher Ansatz, um Industrieanlagen zu schützen. Der Ansatz dient dem Schutz vor unberechtigten Zugriffen, Sabotage, Spionage und böswilliger Manipulation. Industrial Security stellt die folgenden Faktoren sicher:

- Verfügbarkeit
  Bei Industrieanlagen hat Verfügbarkeit höchste Priorität. Ausfallzeiten müssen so gering wie möglich sein.
- System- und Datenintegrität
  Dritte dürfen nicht in der Lage sein, die Anlage und sämtliche Daten unrechtmäßig zu verändern.
- Vertraulichkeit
  Informationen über Industrieanlage und Daten dürfen nicht in die Hände von unberechtigten Dritten gelangen.

Ganzheitlicher Ansatz

Neben den in das TIA Portal integrierten Sicherheitsfunktionen erfordert Industrial Security weitere ganzheitliche Maßnahmen. Beachten Sie die folgenden grundlegenden Sicherheitsaspekte:

- Anlagensicherheit
  - Zugangsschutz gegen unautorisierte Personen
  - Physikalischer Zugangsschutz zu kritischen Komponenten
- Netzwerksicherheit
  - Kontrollierte Schnittstellen zwischen Büro- und Anlagennetzwerk z. B. über Firewalls
  - Segmentierung des Anlagennetzwerks
- Systemintegrität
  - Einsatz von Antivirus- und Whitelisting-Software
  - Einhaltung von Wartungs- und Updateprozessen
  - Nutzerauthentifizierung für Maschinen- oder Anlagenbetreiber
  - Integrierter Zugriffsschutz in Automatisierungskomponenten

1.11 CAx-Daten exportieren und importieren

CAx-Dateien sind XML-Dateien, die Sie im TIA Portal erstellen und zwischen verschiedenen Installationen oder mit anderen Werkzeugen wie z. B. ECAD-Systemen austauschen können. Die Dateien enthalten die folgenden Informationen:

- Hardware-Konfiguration eines einzelnen Geräts oder eines ganzen Projekts (ohne Parametrierung)
- Ordnerhierarchien (Gruppen)
  Die Ordnerhierarchie wird beim Import wieder hergestellt.
- Zugehörigkeit eines Geräts zu einem Subnetz
  Beim Import werden die in der CAx-Datei gespeicherten Geräte wieder mit demselben Subnetz verbunden.
- Adressen der Geräte
- Adressbereiche der Eingänge und Ausgänge
- Profinet I/O-Systeme und PROFIBUS DP Master-Systeme

Voraussetzungen

- TIA Openness ist installiert.
  TIA Openness wird vom Setup-Programm des TIA Portals automatisch installiert. Dazu muss im Setup-Programm unter "Optionen" im Menü "Konfiguration" das Kontrollkästchen für TIA Openness aktiviert sein.
- Ein Projekt ist geöffnet.

CAx-Datei eines Projekts exportieren

1. Klicken Sie im Menü "Werkzeuge" auf den Eintrag "CAx-Daten exportieren...".
2. Wählen Sie im Dialog "CAx Export" ein Verzeichnis für die zu exportierende Datei aus.
3. Ändern Sie gegebenenfalls den Dateinamen und bestätigen Sie mit "Speichern".
   Ein Dialog zeigt den Fortschritt des Exports an. Um den Export gegebenfalls abzubrechen, klicken Sie auf "Abbrechen".
   Im Inspektorfenster im Register "Info > Allgemein" sehen Sie den Status des Exports. Zusätzlich wird eine Protokolldatei in dem Verzeichnis erstellt, das Sie unter "Einstellungen > Allgemein > Datenaustausch" als Speicherort für Protokolldateien eingetragen haben. Die exportierte Datei erhält die Dateiendung ".aml".

CAx-Datei eines Geräts exportieren

1. Markieren Sie die zu exportierenden Geräte in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "CAx-Daten exportieren...".

3. Wählen Sie im Dialog "CAx Export" ein Verzeichnis für die zu exportierende Datei aus.
4. Ändern Sie gegebenenfalls den Dateinamen und bestätigen Sie mit "Speichern".
   Ein Dialog zeigt den Fortschritt des Exports an. Um den Export gegebenfalls abzubrechen, klicken Sie auf "Abbrechen".
   Im Inspektorfenster im Register "Info > Allgemein" sehen Sie den Status des Exports. Zusätzlich wird eine Protokolldatei in dem Verzeichnis erstellt, das Sie unter "Einstellungen > Allgemein > Datenaustausch" als Speicherort für Protokolldateien eingetragen haben. Die exportierte Datei erhält die Dateiendung ".aml".

CAx-Datei eines Geräts oder eines Projekts importieren

1. Klicken Sie im Menü "Werkzeuge" auf den Eintrag "CAx-Daten importieren...".
2. Wählen Sie im Dialog "CAx Import" eine CAx-Datei in einem Verzeichnis aus.
3. Klicken Sie auf die Schaltfläche "Öffnen".
   Ein Dialog zeigt den Fortschritt des Imports an. Um den Import gegebenfalls abzubrechen, klicken Sie auf "Abbrechen".
   Sie erhalten im Inspektorfenster eine Rückmeldung im Register "Info > Allgemein" über den Status des Imports von Geräten, z. B. Fehlermeldungen. Zusätzlich wird eine Protokolldatei in dem Verzeichnis erstellt, das Sie unter "Einstellungen > Allgemein > Datenaustausch" als Speicherort für Protokolldateien eingetragen haben.

Zulässige Austauschformate für CAx-Daten

Zum Austausch von CAx-Daten wird das standardisierte Datenaustauschformat "AutomationML" verwendet. Die Austauschdateien erhalten die Erweiterung "*.aml". Der Export und Import von Dateien mit der Erweiterung "*.xml" wird nicht unterstützt.

Die beim Datenaustausch erzeugten Protokolldateien erhalten die Erweiterung "*.log". Protokolldateien mit der Erweiterung ".txt" werden nicht unterstützt.

Austausch von CAx-Daten mit anderen Anwendungen

Gesteckte Submodule wie z. B. Busadapter können im TIA Portal und in anderen Anwendungen eine abweichende hierarchische Struktur haben. Die AML-Dateien zum Austausch von CAx-Daten stellen die Submodule daher in einer angepassten Hierarchie dar, die programmübergreifend lesbar ist. Beim Import hat die Hierarchie der Submodule in der AML-Datei keinen Einfluss auf die Hierarchie der Submodule im TIA Portal.

Hinweis

Kompatibilität zwischen verschiedenen Produktversionen

Wenn Sie Dateien mit Geräten austauschen wollen, die Busadapter enthalten, beachten Sie folgende Hinweise zur Kompatibilität:

AML-Dateien, die mit einer älteren Version des TIA Portals erstellt wurden, lassen sich mit der aktuellen Version importieren.

AML-Dateien, die mit der aktuellen Version des TIA Portals erstellt wurden, lassen sich mit älteren Versionen nicht importieren.

Importoptionen bei Namenskonflikten von Stationen

Wenn eine zu importierende CAx-Datei und das aktuell geöffnete TIA Portal-Projekt gleichnamige Stationen enthalten, entsteht ein Konflikt. Ein Dialog zur Einstellung der Importoptionen öffnet sich. Bevor Sie die Datei importieren, wählen Sie aus, wie die gleichnamigen Stationen zusammengeführt werden sollen. Die Stationen aus der CAx-Datei können wahlweise umbenannt und in einem neuen Ordner zusammengefasst werden.

Hinweis

Standardname des Ordners

Der Ordner für die zu importierenden Stationen wird standardmäßig mit dem Namen "ParkingLot" benannt.

Sie können den Namen des neuen Ordners auch selbst vorgeben. Öffnen Sie dazu die Einstellungen und navigieren zum Punkt "CAx". Geben Sie den Namen für den Ordner bei "Einstellungen zur Konfliktlösung" in das Feld "Name der Ablage für Geräte, die bereits im Projekt existieren" ein.

Wahlweise können Sie auch vorgeben, dass die konfliktauslösenden Stationen entweder nicht importiert werden oder die namensgleichen Stationen im aktuell geöffneten TIA Portal-Projekt ersetzen.

Kompatibilität von Projekten 2

2.1 Kompatibilität zwischen TIA Portal Versionen

Mit dem TIA Portal V15.1 lassen sich Projekte öffnen, die mit dieser Version des TIA Portals erstellt wurden. Projekte aus früheren Versionen des TIA Portals müssen Sie vor dem Öffnen auf den aktuellen Versionsstand hochrüsten.

Projekte aus früheren Produktversionen öffnen

Die folgende Tabelle zeigt das Verhalten des TIA Portals beim Öffnen von Projekten aus einer früheren Produktversion:

Version des TIA Portals (Dateiendung der jeweiligen Version)	Verhalten beim Öffnen mit der aktuellen Version des TIA Portals
V10.5 (.ap10) V11.x (.ap11) V12 (.ap12) V12 SP1 (.ap12) V13 (.ap13)	Projekte aus Vorgängerversionen von V13 SP1 des TIA Portals lassen sich nicht direkt mit dem TIA Portal V15.1 öffnen. Sie erhalten daher beim Öffnen des Projekts den Hinweis, das Projekt mit der Version V13 SP1 zu öffnen. Wenn Sie das Projekt mit der TIA Portal Version V13 SP1 öffnen, dann wird das Projekt nach Ihrer Bestätigung auf die Projektversion V13 SP1 hochgerüstet und erhält die Dateiendung ".ap13".
V13 SP1 (.ap13) V14 (.ap14) V14 SP1 (.ap14) V15 (.ap15)	Das Projekt wird beim Öffnen nach Ihrer Bestätigung auf die Projektversion V15.1 hochgerüstet und erhält die Dateiendung ".ap15_1".

Rückwärtskompatibilität der aktuellen Version des TIA Portals

Projekte, die Sie in der aktuellen Projektversion V15.1 speichern, sind aufgrund der erweiterten Funktionalität gegenüber früheren Versionen nicht rückwärts kompatibel. Projekte in der Projektversion V15.1 lassen sich ausschließlich mit dem TIA Portal V15.1 oder neuer öffnen.

Projekte mit zugewiesenen Funktionsrechten öffnen

Wenn in einem Projekt bestimmte Funktionsrechte über die Benutzerverwaltung eingerichtet wurden, lässt sich die Benutzerverwaltung für dieses Projekt nur in Produktversionen öffnen, die diese Funktionsrechte unterstützen.

2.2 Mit optionaler Software erstellte Projekte verwenden

Für das TIA Portal ist zahlreiche optionale Software erhältlich. Dazu zählen zum Beispiel Hardware Support Packages. Wenn Sie ein Projekt öffnen, das mit optionaler Software erstellt wurde, die optionale Software ist jedoch nicht installiert, dann können die folgenden Fälle auftreten:

- Software-Komponenten, die nicht zwingend erforderlich sind, fehlen:
  Ein Dialog erscheint, in dem die fehlenden Software-Komponenten aufgelistet sind. Nach dem Öffnen des Projekts werden die Eigenschaften des Projekts angezeigt. An dieser Stelle haben Sie die Möglichkeit, fehlende Komponenten nachzuinstallieren. Auch ohne die Nachinstallation der fehlenden Komponenten stehen alle im Projekt enthaltenen Geräte zur Verfügung. Sie können jedoch nur die Geräte bearbeiten, die von der aktuell installierten Software unterstützt werden.
  Geräte, deren Bearbeitung aufgrund von fehlenden Software-Komponenten nicht unterstützt wird, sind mit folgendem Symbol in der Projektnavigation gekennzeichnet:

- Software-Komponenten, die zwingend erforderlich sind, fehlen:
  Ein Dialog erscheint, in dem die fehlenden Software-Komponenten aufgelistet sind. Die zwingend erforderlichen Software-Komponenten sind gekennzeichnet. Das Projekt lässt sich nur öffnen, wenn Sie die fehlenden Software-Komponenten installieren.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Projekte öffnen (Seite 10)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

2.3 Projekte hochrüsten

2.3.1 Allgemeines zum Hochrüsten von Projekten

Ein Projekt aus der Version V13 SP1 des TIA Portals lässt sich erst öffnen, nachdem sie das Projekt auf die aktuelle Version des TIA Portals hochgerüstet haben. Beim Öffnen des Projekts erhalten Sie eine Aufforderung, das Projekt hochzurüsten.

Welche Möglichkeiten Sie bei anderen Produktversionen des Projekts haben, lesen Sie im Kapitel "Kompatibilität zwischen TIA Portal Versionen (Seite 25)".

Bausteine mit Know-how-Schutz hochrüsten

Bausteine mit Know-how-Schutz werden nicht automatisch mit dem Projekt hochgerüstet. Entfernen Sie den Know-how-Schutz der Bausteine, bevor Sie das Projekt hochrüsten:

- Bei V13 SP1-Projekten
  Entfernen Sie den Know-how-Schutz mit dem TIA Portal V13 SP1.
- Bei V14-Projekten
  Entfernen Sie den Know-how-Schutz mit dem TIA Portal V14 oder mit dem TIA Portal V14 SP1 im Kompatibilitätsmodus.
- Bei V14 SP1-Projekten
  Entfernen Sie den Know-how-Schutz mit dem TIA Portal V14 SP1.
- Bei V15-Projekten
  Entfernen Sie den Know-how-Schutz mit dem TIA Portal V15.

Richten Sie den Know-how-Schutz anschließend mit der aktuellen Version des TIA Portals wieder ein. Weitere Informationen finden Sie im Kapitel "Know-how-geschützte Bausteine hochrüsten (Seite 31)".

Geschützte Projekte hochrüsten

Wenn Sie ein mit der Benutzerverwaltung geschütztes Projekt aus einer früheren Produktversion hochrüsten wollen, müssen Sie in diesem Projekt als Projektadministrator angelegt sein und benötigen die Zugangsdaten zum Projekt.

Beim Hochrüsten von geschützten Projekten wird der Dialog "Anmelden" geöffnet. Damit Sie den Vorgang fortsetzen können, wählen Sie den Benutzertyp aus und geben Ihren Benutzernamen und Ihr Passwort ein.

Weitere Informationen finden Sie im Kapitel "Grundlagen zur Benutzerverwaltung im TIA Portal".

Globale Bibliotheken hochrüsten

Globale Bibliotheken werden nicht automatisch zusammen mit dem Projekt hochgerüstet, da diese unabhängig von Projekten sind. Wenn Sie globale Bibliotheken aus früheren Versionen des TIA Portals weiterverwenden möchten, rüsten Sie die globalen Bibliotheken ebenfalls hoch. Weitere Informationen zum Hochrüsten von globalen Bibliotheken finden Sie im Kapitel "AUTOHOTSPOT".

Anweisungsversionen

Die Anweisungen, die Sie zum Erstellen Ihres Anwenderprogramms verwenden, können in verschiedenen Versionen vorliegen. Unterschiedliche Produktversionen des TIA Portals können verschiedene Anweisungsversionen enthalten. Sie haben die Möglichkeit, das Projekt und die Anweisungen auf die neueste Version hochzurüsten. Anschließend stehen Ihnen die neuen Anweisungsversionen zur Verfügung. Nach dem Hochrüsten werden die neuen Anweisungsversionen jedoch nicht automatisch in Ihrem Programm verwendet. Sie können Ihr Programm aktualisieren, um die neuen Versionen zu verwenden. Mehr zum Thema "Anweisungsversionen" finden Sie im Kapitel "Programmiereditor".

Multiuser Engineering

Um Multiuser Engineering Projekte hochzurüsten, beachten Sie das im Kapitel "Multiuser Engineering Projekte hochrüsten (Seite 30)" beschriebene Vorgehen.

Projekte mit fehlender Lizenz hochrüsten

Zum Hochrüsten von Projekten ist keine Lizenz erforderlich. Wenn Sie über keine oder über eine nicht passende Lizenz verfügen, können Sie nicht den vollen Funktionsumfang der Produkte nutzen.

So können Sie z. B. mit einer Lizenz von STEP 7 Basic keine Funktionen nutzen, die an eine Lizenz von STEP 7 Professional gebunden sind.

Siehe auch

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte hochrüsten (Seite 28)

2.3.2 Projekte hochrüsten

Im Folgenden erfahren Sie, wie Sie Projekte aus früheren Versionen des TIA Portals hochrüsten.

Projekte aus dem TIA Portal V14 oder höher hochrüsten

Um ein Projekt aus dem TIA Portal V14 oder höher hochzurüsten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".
   Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird angezeigt.
2. Selektieren Sie ein Projekt aus der Liste und klicken Sie auf "Öffnen".
3. Falls das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.
   Der Dialog "Projekt öffnen" wird geöffnet.

4. Klicken Sie auf "Hochrüsten".
   Der Dialog "Projekt/Bibliothek hochrüsten" wird geöffnet.
5. Klicken Sie auf "OK".
   Das Projekt wird auf die aktuelle Projektversion hochgerüstet und geöffnet.

Projekte aus dem TIA Portal V13 SP1 hochrüsten

Um ein Projekt aus dem TIA Portal V13 SP1 hochzurüsten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".
   Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird angezeigt.
2. Selektieren Sie ein Projekt aus der Liste und klicken Sie auf "Öffnen".
3. Falls das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.
   Der Dialog "Projekt öffnen" wird geöffnet.
4. Klicken Sie auf "Hochrüsten".
   Der Dialog "Projekt/Bibliothek hochrüsten" wird geöffnet.
5. Klicken Sie auf "OK".
6. Bestätigen Sie das Hinweisfenster mit "OK".
   Das Projekt wird auf die aktuelle Projektversion hochgerüstet und geöffnet.
7. Übersetzen Sie die Hardware und Software aller Geräte im Projekt.

TIA Portal V13 SP1-Projekte mit Kopiervorlagen aus dem TIA Portal V10.5 hochrüsten

Kontrollieren Sie vor dem Hochrüsten eines Projekts, ob in dessen Projektbibliothek mit dem TIA Portal V10.5 erstellte Kopiervorlagen enthalten sind. Dies ist beispielsweise dann notwendig, wenn Sie beim Hochrüsten eine Meldung über eine im Projekt verwendete S7-1200 CPU mit der Firmware-Version V1.0 erhalten, obwohl Sie keine solche CPU projektiert haben. Die CPU befindet sich in diesem Fall als Kopiervorlage in der Projektbibliothek.

Gehen Sie deshalb vor dem Hochrüsten eines Projekts mit Kopiervorlagen aus dem TIA Portal V10.5 folgendermaßen vor:

1. Verwenden Sie jede Kopiervorlage, die mit dem TIA Portal V10.5 erstellt wurde, einmal im Projekt.
2. Löschen Sie die betroffenen Kopiervorlagen aus der Projektbibliothek.
3. Erstellen Sie die Kopiervorlage mit den Objekten aus dem Projekt neu.
4. Speichern Sie das Projekt im TIA Portal V13 SP1.

Anschließend können Sie das Projekt auf die aktuelle Version hochrüsten.

Projekte aus dem TIA Portal V13 oder früher hochrüsten

Mit dem TIA Portal lassen sich keine Projekte der Produktversion V13 oder früher hochrüsten und öffnen. Rüsten Sie Projekte aus dem TIA Portal V13 oder früher zunächst mit der Produktversion V13 SP1 hoch. Anschließend rüsten Sie die Projekte mit der aktuellen Version des TIA Portals erneut auf die Projektversion V15.1 hoch.

Weitere Informationen zum Hochrüsten von Projekten aus früheren Produktversionen finden Sie im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/109476392).

Siehe auch

Know-how-geschützte Bausteine hochrüsten (Seite 31)

Allgemeines zum Hochrüsten von Projekten (Seite 27)

Multiuser Engineering Projekte hochrüsten (Seite 30)

Grundlagen zu Projekten (Seite 7)

Projekt erstellen (Seite 9)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte öffnen (Seite 10)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte speichern (Seite 12)

Projekte schließen (Seite 13)

Projekte aus der Projektliste entfernen (Seite 14)

Projekte löschen (Seite 15)

2.3.3 Multiuser Engineering Projekte hochrüsten

Im Folgenden erfahren Sie, wie Sie Multiuser Engineering Projekte hochrüsten.

Vorgehen

Um Multi User Engineering Projekte hochzurüsten, gehen Sie folgendermaßen vor:

1. Speichern Sie vor dem Hochrüsten alle vorhandenen lokalen Sessions im Multiuser-Serverprojekt.
2. Speichern Sie das Multiuser-Serverprojekt als Single-User-Projekt.
3. Rüsten Sie das Single-User-Projekt hoch.
4. Erstellen Sie nach dem Hochrüsten aus dem Single-User-Projekt wieder ein Multiuser-Projekt.
5. Legen Sie vom hochgerüsteten Multiuser-Projekt für alle Bearbeiter die lokalen Sessions erneut an.

Siehe auch

Projekte hochrüsten (Seite 28)

2.3.4 Know-how-geschützte Bausteine hochrüsten

Aus Sicherheitsgründen ist der Know-how-Schutz von Bausteinen an die Version des TIA Portals gebunden, mit der der Know-how-Schutz eingerichtet wurde. Bausteine, die in einer früheren Produktversion mit einem Know-how-Schutz versehen wurden, lassen sich nach dem Hochrüsten mit dem bisherigen Passwort nicht öffnen und bearbeiten. Die Bausteine lassen sich jedoch in die Steuerung laden und sind ablauffähig. Falls Sie den Baustein nach dem Hochrüsten öffnen und bearbeiten möchten, dann entfernen Sie den Know-how-Schutz der Bausteine und richten ihn erneut ein.

Vorgehen

Um know-how-geschützte Bausteine hochzurüsten, gehen Sie folgendermaßen vor:

1. Entfernen Sie in der früheren Produktversion den Know-how-Schutz der Bausteine im Menü "Bearbeiten" über den Befehl "Know-how-Schutz".
   Weitere Informationen finden Sie im Kapitel "Know-how-Schutz für Bausteine entfernen".
2. Rüsten Sie das Projekt hoch, wie im Kapitel "Projekte hochrüsten (Seite 28)" beschrieben.
3. Richten Sie in der aktuellen Produktversion den Know-how-Schutz der Bausteine im Menü "Bearbeiten" über den Befehl "Know-how-Schutz" wieder ein.
   Weitere Informationen finden Sie im Kapitel "Know-how-Schutz für Bausteine einrichten".

2.4 Kompatibilität mit dem Anlagenaufbau

Um alle Online-Funktionen nutzen zu können, muss der Anlagenaufbau mit dem TIA Portal V15.1 und der Projektversion des geöffneten Projekts kompatibel sein. Beim Anlagenaufbau sind insbesondere folgende Faktoren für die Kompatibilität relevant:

- Hardware und deren Firmware-Version
  Die einzelnen Geräte und deren Firmware-Version müssen vom TIA Portal V15.1 und innerhalb des geöffneten Projekts unterstützt werden.
- Das Anwenderprogramm auf den jeweiligen Geräten
  Alle Bausteine und Anweisungen auf den Geräten müssen im TIA Portal V15.1 verfügbar und mit der Projektversion des geöffneten Projekts nutzbar sein.

Einfluss der Projektversion auf die Kompatibilität

Mit dem TIA Portal V15.1 können Sie Projekte für die Projektversion V15.1 bearbeiten. Sie können nur Hardware-Konfiguration und Software von der Anlage laden, die mit Ihrer Version des TIA Portals kompatibel ist.

Bei CPUs der S7-1200 und S7-1500 Produktfamilie sehen Sie auf dem Display, im Webserver und in der Online- und Diagnosesicht des TIA Portals die Projektversion, mit der die Bausteine auf der CPU kompatibel sind. Beachten Sie jedoch, dass alleine eine kompatible Projektversion nicht ausreicht, um mit der CPU zu arbeiten. Auch die CPU selbst und deren Firmware-Stand müssen mit der installierten Version des TIA Portals und den einzelnen installierten Produkten kompatibel sein.

Die Projektversion des gerade geöffneten Projekts sehen Sie in den Eigenschaften des Projekts.

Online-Verbindung nach Hochrüsten eines Projekts

Wenn Sie ein Projekt aus der Version V14, V14 SP1 oder V15 auf die Version V15.1 hochrüsten, können Sie anschließend Geräte online verbinden, die mit Version V14, V14 SP1 oder V15 geladen wurden. Die Bausteine dieser Geräte bleiben funktionsfähig und müssen nicht neu übersetzt werden.

Somit können Sie z. B. den Anlagenaufbau erweitern oder Geräte mit neuer Funktionalität ausstatten, ohne dass sich die Änderungen auf andere Geräte im Anlagenaufbau auswirken.

Der Unterschied der Projektversion wird in der Projektnavigation und im Vergleichseditor durch ein Overlay-Symbol am Gerät angezeigt.

Das folgende Symbol zeigt den Unterschied der Projektversion an, wenn keine Programmbausteine modifiziert wurden:

Wenn keine Bausteine modifiziert wurden, wird bei den Bausteinen kein Online-/Offline-Unterschied angezeigt.

Das folgende Symbol zeigt den Unterschied der Projektversion an, wenn Bausteine modifiziert wurden:

Wenn Bausteine modifiziert wurden, wird bei den Bausteinen ein Online-/Offline-Unterschied angezeigt. Sie können diese Bausteine nicht vom Gerät laden, da sich ins TIA Portal V15.1 nur Bausteine der gleichen Version laden lassen.

Inkompatible Bausteine

Bei einer bestehenden Online-Verbindung wird für die projektierten Geräte ein Offline-/Online-Vergleich durchgeführt. Wenn die online verfügbaren Bausteine nicht mit der Projektversion des geöffneten Projekts kompatibel sind, dann werden nicht kompatible Bausteine in der Projektnavigation mit folgendem Symbol als nicht kompatibel gekennzeichnet:

Hardware-Konfiguration und Software auf ein Gerät laden

Wenn Sie die Hardware-Konfiguration oder Software auf ein Gerät laden möchten, das zuletzt mit dem TIA Portal V15 oder einer Vorgängerversion geladen wurde, dann laden Sie die gesamte Hardware-Konfiguration und Software neu auf das Gerät. Nur die Änderungen zu laden, ist in diesem Fall nicht möglich. Auf Geräte, die zuletzt mit dem TIA Portal V15.1 oder neuer geladen wurden, können Sie auch nur die Änderungen an der Hardware-Konfiguration und Software laden.

Siehe auch

Grundlagen zu Projekten (Seite 7)

Allgemeines zum Hochrüsten von Projekten (Seite 27)

Projekte hochrüsten (Seite 28)

Eigenschaften des Projekts anzeigen (Seite 16)

Projekte archivieren und dearchivieren 3

3.1 Arbeiten mit Projektarchiven

Archivierung und Weitergabe von Projekten

Wenn Sie längere Zeit mit einem Projekt arbeiten, kann dies insbesondere bei umfangreichen Hardware-Aufbauten zu großen Dateien führen. Daher können Sie die Projektgröße reduzieren, wenn Sie ein Projekt beispielsweise auf einer externen Festplatte archivieren möchten, oder wenn Sie ein Projekt per E-Mail verschicken möchten und daher auf eine geringe Dateigröße angewiesen sind.

Um die Projektgröße zu reduzieren, erstellen Sie ein Projektarchiv. TIA Portal-Projektarchive sind komprimierte oder unkomprimierte Dateien, die jeweils ein ganzes Projekt inklusive der gesamten Ordnerstruktur des Projekts enthalten. Bevor das Projektverzeichnis zur Archivdatei gepackt wird, werden sämtliche Dateien auf ihre wesentlichen Bestandteile reduziert, um die Projektgröße zusätzlich zu verringern.

Projektarchive besitzen die Dateiendung ".zap[Versionsnummer des TIA Portals]". Projekte, die mit dem TIA Portal V15.1 erstellt wurden, haben die Dateiendung ".zap15_1".

Um ein Projektarchiv zu öffnen, dearchivieren Sie das Projektarchiv. Mit dem Dearchivieren wird die Archivdatei zur ursprünglichen Projektverzeichnisstruktur mit den darin enthaltenen Projektdateien entpackt.

Siehe auch

Komprimiertes Projekt dearchivieren (Seite 38)

Projektarchiv erstellen (Seite 36)

3.2 Projektarchiv erstellen

Projekte der aktuellen Produktversion lassen sich als komprimierte oder unkomprimierte Dateien archivieren. Die zu archivierenden Projekte müssen dafür nicht im TIA Portal geöffnet sein.

Der Speicherplatzbedarf von Projekten lässt sich reduzieren, indem Sie Projekte in komprimierten Dateien archivieren.

Hinweis

Beim Archivieren eines geöffneten Projekts wird der zuletzt gespeicherte Stand des Projekts verwendet. Speichern Sie daher das Projekt ab, bevor Sie die Archivieren-Funktion nutzen. So sind auch Ihre letzten Änderungen im archivierten Projekt enthalten.

Vorgehen

Um ein Projekt zu archivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Archivieren...".
   Der Dialog "Archivieren" wird geöffnet.
2. Wählen Sie im Feld "Quellpfad" eine Projektdatei mit der Endung ".ap15_1" aus.
3. Um eine komprimierte Archivdatei zu erzeugen, wählen Sie die Option "Als komprimierte Datei archivieren" aus.
4. Wenn Sie den Suchindex und das HMI Compile-Ergebnis nicht archivieren wollen, wählen Sie die Option "Wiederherstellbare Daten verwerfen" aus.
   Die verworfenen Daten können Sie bei Bedarf wiederherstellen.
5. Um Datum und Zeit automatisch hinzuzufügen, wählen Sie die Option "Datum und Zeit zum Zielnamen hinzufügen" aus.
6. Wählen Sie im Feld "Zielpfad" den Speicherort, an dem Sie die Archivdatei oder das neue Verzeichnis des Projekts speichern möchten.
   Das Standardverzeichnis können Sie unter "Extras > Einstellungen > Allgemein > Archivierungseinstellungen > Speicherort für Projektarchive" einstellen.
7. Klicken Sie auf "Archivieren".

Ergebnis

Ein Archiv des Projekts wird erzeugt. Komprimierte Projekte erhalten die Dateiendung ".zap15_1". Die Archivdatei enthält das komplette Projektverzeichnis. Die einzelnen Dateien von komprimierten Archiven wurden zusätzlich auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

Bei unkomprimierten Projekten wird lediglich eine Kopie des ursprünglichen Projektverzeichnisses am gewünschten Speicherort erstellt.

Siehe auch

Arbeiten mit Projektarchiven (Seite 35)
Komprimiertes Projekt dearchivieren (Seite 38)

3.3 Komprimiertes Projekt dearchivieren

Projektarchive des TIA Portals entpacken Sie mit der Funktion "Dearchivieren". Dabei wird die enthaltene Projektverzeichnisstruktur inklusive aller Projektdateien wiederhergestellt.

Voraussetzung

Kein Projekt ist geöffnet.

Vorgehen

Um ein Projektarchiv zu entpacken, gehen Sie folgendermaßen vor:

1. Wählen Sie den Befehl "Dearchivieren" im Menü "Projekt".
   Der Dialog "Archiviertes Projekt dearchivieren" wird geöffnet.
2. Wählen Sie das Projektarchiv aus.
3. Klicken Sie auf Öffnen.
4. Der Dialog "Ordner suchen" wird geöffnet.
5. Wählen Sie das Zielverzeichnis aus, in dem das archivierte Projekt entpackt werden soll.
6. Klicken Sie auf "OK".

Ergebnis

Das Projekt wird im gewählten Verzeichnis entpackt und sofort geöffnet. Wenn Sie ein Projektarchiv entpacken, das ein Projekt aus der Produktversion V13 SP1 enthält, müssen Sie das Projekt hochrüsten. Eine entsprechende Aufforderung erhalten Sie automatisch, sobald das Projekt geöffnet wird. Es gelten dieselben Regeln, wie im Kapitel "Mit optionaler Software erstellte Projekte verwenden (Seite 26)" beschrieben. Suchindex und HMI Compile-Ergebnis werden automatisch im Hintergrund wiederhergestellt.

Siehe auch

Arbeiten mit Projektarchiven (Seite 35)

Projekte öffnen (Seite 10)

Mit optionaler Software erstellte Projekte verwenden (Seite 26)

Projekte hochrüsten (Seite 28)

Referenzprojekte verwenden 4

4.1 Grundlagen zu Referenzprojekten

Einführung

Sie haben die Möglichkeit zusätzlich zum aktuellen Projekt weitere Projekte als Referenz zu öffnen. Diese Referenzprojekte können Sie folgendermaßen verwenden:

- Sie können einzelne Objekte aus einem Referenzprojekt in das aktuelle Projekt ziehen und anschließend bearbeiten.
- Sie können bestimmte Objekte, z. B. Codebausteine, aus einem Referenzprojekt schreibgeschützt öffnen. Dies ist jedoch nicht für alle Elemente möglich.
- Sie können Geräte des Referenzprojekts mithilfe eines Offline/Offline-Vergleichs mit Geräten aus dem aktuellen Projekt vergleichen.

Beachten Sie, dass die Referenzprojekte schreibgeschützt sind. Sie können die Objekte eines Referenzprojekts also nicht verändern. Daher können schwebende Änderungen, z. B. Namensänderungen von Objekten, erst auf das referenzierende Objekt angewendet werden, wenn das Projekt als normales Projekt geöffnet wird. Nachdem die Änderungen angewendet wurden, können Sie das Projekt als Referenzprojekt weiterverwenden.

Sie können auch Projekte als Referenzprojekte öffnen, die mit einer früheren Version des TIA Portals oder mit einem anderen Installationsumfang erstellt wurden. Dabei gelten die gleichen Kompatibilitätsregeln wie beim normalen Öffnen eines Projekts aus einer früheren Version des TIA Portals.

Siehe auch: Kompatibilität von Projekten (Seite 26)

Siehe auch

Referenzprojekte vergleichen (Seite 41)

Referenzprojekt öffnen und schließen (Seite 40)

4.2 Referenzprojekt öffnen und schließen

Referenzprojekt öffnen

Um ein Referenzprojekt zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Menü "Ansicht" das Optionskästchen "Referenzprojekte".
   Die Palette "Referenzprojekte" wird in der Projektnavigation angezeigt.
2. Klicken Sie in der Projektnavigation in der Palette "Referenzprojekte" in der Funktionsleiste auf "Referenzprojekt öffnen".
   Der Dialog "Referenzprojekt öffnen" wird geöffnet.
3. Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei. Projekte des TIA Portals V15.1 besitzen die Endung ".ap15_1". Ältere Projekte des TIA Portals haben die Endung ".ap[Versionsnummer]".
4. Klicken Sie auf "Öffnen".
   Das gewählte Projekt wird schreibgeschützt als Referenzprojekt geöffnet.

Referenzprojekt schließen

Um ein Referenzprojekt zu schließen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation in der Palette "Referenzprojekte" das Referenzprojekt, das Sie schließen möchten.
2. Klicken Sie in der Funktionsleiste auf "Referenzprojekt schließen".
   Das gewählte Referenzprojekt wird geschlossen.

Siehe auch

Grundlagen zu Referenzprojekten (Seite 39)

Referenzprojekte vergleichen (Seite 41)

4.3 Referenzprojekte vergleichen

Einführung

Sie können Geräte aus Referenzprojekten sowohl mit Geräten aus dem aktuellen Projekt als auch aus dem gleichen oder einem anderen Referenzprojekt oder auch aus einer Bibliothek vergleichen.

Hinweis

Beachten Sie folgende Hinweise:

- Sie können für die Vergleichsobjekte keine Aktionen festlegen, da die Referenzprojekte schreibgeschützt sind.
- Sie können für die Vergleichsobjekte einen Detailvergleich durchführen, wenn der Typ des Vergleichsobjekts einen Detailvergleich generell erlaubt.
- Sie können auch beim Vergleichen von Referenzprojekten jederzeit zwischen automatischem und manuellem Vergleich wechseln.

Vorgehen

Um die Objekte eines Referenzprojekts mit den Gerätedaten des aktuellen Projekts zu vergleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, dessen Daten Sie mit den Daten eines Referenzprojekts vergleichen möchten und das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Öffnen Sie in der Projektnavigation die Palette "Referenzprojekte".
4. Selektieren Sie das Gerät eines Referenzprojekts, das Sie mit den Gerätedaten aus dem aktuellen Projekt vergleichen möchten.

5. Ziehen Sie das Gerät aus dem Referenzprojekt in die rechte Drop-Fläche des Vergleichseditors.
   Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.
   Sie können jederzeit aus dem aktuellen Projekt, einer Bibliothek oder aus einem Referenzprojekt andere Geräte in die Drop-Flächen ziehen und somit einen neuen Vergleich starten. Dabei ist es unerheblich, welches Gerät Sie in welche Drop-Fläche ziehen.
6. Führen Sie bei Bedarf einen Detailvergleich für Objekte aus, für die diese Vergleichsart möglich ist.

Hinweis

Für Bausteine, PLC-Variablen und PLC-Datentypen können Sie den Detailvergleich direkt aus dem Referenzprojekt starten. Weitere Informationen finden Sie im Kapitel "PLC-Programme vergleichen".

Siehe auch

Grundlagen zu Referenzprojekten (Seite 39)

Referenzprojekt öffnen und schließen (Seite 40)

TIA Portal Teamcenter Gateway 5

5.1 Einführung in das TIA Portal Teamcenter Gateway

5.1.1 Grundlagen zum Arbeiten mit dem TIA Portal Teamcenter Gateway

Einführung

Im TIA Portal haben Sie ab der Version V14 und in WinCC ab der Version V15 die Möglichkeit, mit der Option TIA Portal Teamcenter Gateway zu arbeiten. Das hat den Vorteil, dass Sie Ihre TIA Portal-Projekte und globalen Bibliotheken in Teamcenter speichern und verwalten können.

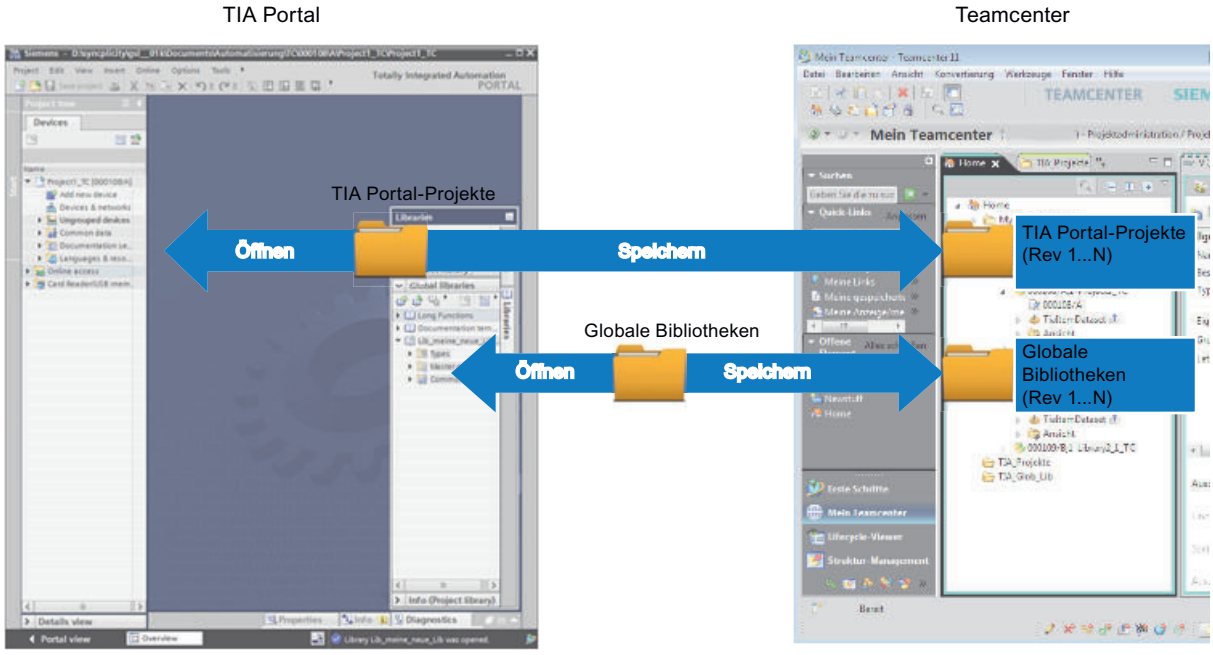

Vorteile

Durch den Einsatz der Option TIA Portal Teamcenter Gateway ergeben sich für Sie folgende Vorteile:

- Sie können TIA Portal-Projekte und globale Bibliotheken in Teamcenter speichern.
- Sie können einem TIA Portal-Projekt und einer globale Bibliothek einen Elementtypen zuweisen.
- Sie können TIA Portal-Projekte und globale Bibliotheken, die in Teamcenter gespeichert sind, revisionieren/versionieren.

- Sie können TIA Portal-Projekte und globale Bibliotheken, die in Teamcenter gespeichert sind, im TIA Portal zur weiteren Bearbeitung öffnen.
- Sie können TIA Portal-Projekte, die mit Teamcenter verbunden sind, dem Multiuser-Server als Multiuser-Serverprojekte hinzufügen. Änderungen, die Sie an Multiuser-Serverprojekten vorgenommene haben, können Sie als neue Version (Dataset Version) nach Teamcenter speichern.
- Sie können TIA Portal Projekte oder globale Bibliotheken exportieren, um die Teamcenter Metainformationen zu entfernen und das geänderte Projekt oder die globale Bibliothek in den ausgewählten Ordner speichern.

5.1.2 Einsatzmöglichkeiten von TIA Portal Teamcenter Gateway

Einsatzmöglichkeiten von TIA Portal Teamcenter Gateway

Das TIA Portal Teamcenter Gateway bietet folgende Einsatzmöglichkeiten:

- TIA Portal-Projekte oder globale Bibliotheken als neues Element in Teamcenter speichern.
- TIA Portal-Projekten oder globale Bibliotheken, die Sie als neues Element in Teamcenter speichern möchten, einen Elementtypen oder abhängigen Typen zuweisen.
- Änderungen an einem TIA Portal-Projekt oder einer globalen Bibliothek in Teamcenter speichern.
- TIA Portal-Projekte oder globale Bibliotheken als neue Revision in Teamcenter speichern.
- Speichert Element-Typen von TIA Portal-Projekten oder globalen Bibliotheken oder abhängige Typen als neue Revision in Teamcenter.
- TIA Portal-Projekte oder globale Bibliotheken auschecken.
- Auschecken eines TIA Portal-Projekts oder einer globalen Bibliothek rückgängig machen.
- TIA Portal-Projekte oder globale Bibliotheken einchecken.
- TIA Portal-Projekte oder globale Bibliotheken, die in Teamcenter gespeichert sind, zur weiteren Bearbeitung im TIA Portal öffnen.
- TIA Portal Projekte oder globale Bibliotheken exportieren, um die Teamcenter Metainformationen zu entfernen und das geänderte Projekt oder die globale Bibliothek in den ausgewählten Ordner zu speichern.
- Sie können TIA Portal-Projekte, die mit Teamcenter verbunden sind, dem Multiuser-Server als Multiuser-Serverprojekte hinzufügen.

5.2 Systemvoraussetzungen

5.2.1 Systemvoraussetzungen für das TIA Portal Teamcenter Gateway

Systemvoraussetzung

Für das Arbeiten mit TIA Portal Teamcenter Gateway ab der V14 gelten die gleichen Hard- und Software-Voraussetzungen wie für das Arbeiten mit dem TIA Portal V14 oder höher.

Hinweis

Identische Software-Installation als Voraussetzung für das Arbeiten mit TIA Portal Teamcenter Gateway

Um ohne Einschränkungen mit TIA Portal Teamcenter Gateway arbeiten zu können, ist es wichtig, dass die Software-Produkte des TIA Portals und des TIA Portal Teamcenter Gateway mit identischen Ausgabeständen installiert sind.

Systemvoraussetzungen für die Installation des TIA Portal Teamcenter Gateway

- Sie haben Administrationsrechte auf Ihrem Computer.
- Sie haben Teamcenter Rich Application Client (nachfolgend "RAC" genannt) oder Teamcenter Client Communication System (TCCS) Version 11.2 oder höher ist auf Ihrem PG/PC installiert.

  Hinweis

  Wenn Sie die Installation von TCCS als "Standalone"-Installation durchgeführt haben, müssen Sie auch "Microsoft Visual C++ 2013 – Redistributable" installieren. Damit stellen Sie sicher, dass Sie eine Verbindung zwischen dem TIA Portal und TIA Portal Teamcenter Gateway herstellen können.

- Sie haben sichergestellt, dass eine Verbindung von FCC/FMS (FileClientCache/FileManagementSystem) zum "Teamcenter Server 11.2 oder höher" besteht.

  Hinweis

  Prüfen Sie auch nach einer Nachinstallation oder einem Neustart, dass eine Verbindung von FCC/FMS (FileClientCache/FileManagementSystem) zum "Teamcenter Server 11.2 oder höher" besteht.

- Sie haben, das im Lieferumfang von TIA Portal Teamcenter Gateway enthaltene Teamcenter Datenmodell, auf dem Teamcenter Server installiert.
  Eine detaillierte Beschreibung zur Installation des Teamcenter Datenmodells finden Sie im Installationshandbuch "TIA Portal Teamcenter Gateway - Datenmodell installieren".
- Sie haben, die im Lieferumfang von TIA Portal Teamcenter Gateway enthaltenen Teamcenter Suchabfragen "__T4Tia_Item.xml", "__T4Tia_ItemRevision.xml", "__T4Tia_ItemWithRevision.xml" und "__T4Tia_RevisionOrderbyCreatedDateTime.xml" in Teamcenter importiert.
  Eine detaillierte Beschreibung zum Importieren von Teamcenter Suchabfragen finden Sie im Installationshandbuch "TIA Portal Teamcenter Gateway - Datenmodell installieren" im Kapitel "Suchabfragen in Teamcenter importieren".

Lizenzen für das TIA Portal Teamcenter Gateway

Für das Arbeiten mit dem TIA Portal Teamcenter Gateway benötigen Sie auf jedem Gerät, auf dem Sie TIA Portal Teamcenter Gateway installieren einen gültigen License Key.

Sie können den License Key entweder bei der Installation gleich mitinstallieren oder nach der Installation mit dem Automation License Manager übertragen. Beim Arbeiten, mit dem TIA Portal in Verbindung mit TIA Portal Teamcenter Gateway wird das Vorhandensein des gültigen License Keys geprüft.

Installation des TIA Portal Teamcenter Gateway in Verbindung mit Windows10

Beachten Sie Folgendes:

- Starten Sie nach der Installation des TIA Portal Teamcenter Gateway, in Verbindung mit Windows10, den Computer neu.

Voraussetzungen für das Arbeiten zwischen TIA Portal Teamcenter Gateway und Multiuser Engineering

- Sie haben Zugriff auf den Multiuser-Server.
- Sie haben Zugriff auf das Multiuser-Serverprojekt.

5.3 TIA Portal Teamcenter Gateway installieren und deinstallieren

5.3.1 Hinweise zur Installation des TIA Portal Teamcenter Gateway

Einführung

Sie können das TIA Portal Teamcenter Gateway wie folgt installieren:

- Zusammen mit der Installation des TIA Portals.

Hinweis

Für die Serverinstallation werden Administratorrechte benötigt.

Installation des TIA Portal Teamcenter Gateway mit dem TIA Portal

Das TIA Portal Teamcenter Gateway ist in der Produktauswahl teilweise per Voreinstellung aktiviert und wird dann mitinstalliert. Überprüfen Sie die Einstellung während der Installation des TIA Portals. Sie finden die jeweilige Voreinstellung für das TIA Portal Teamcenter Gateway während des Setups bei der Produktauswahl im Verzeichnis "Options". Wählen Sie hier durch das Setzen oder entfernen des Häkchens aus, ob das TIA Portal Teamcenter Gateway während der Installation des TIA Portals mitinstalliert werden soll oder nicht.

Gleiche TIA Portal-Versionen bei der Installation

Achten Sie bei der Installation von verschiedenen TIA Portal-Produkten darauf, dass Sie für die Installation die gleichen Stände der Service Packs und Updates verwenden.

Wenn Sie also z. B. TIA Portal V14 oder höher installiert haben, müssen Sie auch das TIA Portal Teamcenter Gateway V14 oder höher installieren.

Installationspfad

Verwenden Sie keine Unicode-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

Virenscanner

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscanner verhindern dies. Daher wird empfohlen, während der Installation des TIA Portal Teamcenter Gateway den Virenscanner zu deaktivieren und erst danach wieder zu aktivieren.

Hinweise zur Virtualisierung

Das TIA Portal Teamcenter Gateway wurde mit folgender VMware (virtuellen Maschine) getestet:

- VMware Workstation 12 Pro

5.3.2 Hinweise zur Kompatibilität des TIA Portal Teamcenter Gateway

Kompatibilität des TIA Portal Teamcenter Gateway mit anderen Software-Produkten

Das TIA Portal Teamcenter Gateway wurde mit folgenden Software-Produkten auf Verträglichkeit getestet:

TIA Portal Installationen

- WinCC Professional
- WinCC UAES
- WinCC Advanced

Optionspakete

- STEP 7 Safety
- Multiuser-Server
- Cloud Connector

5.3.3 Parallelinstallationen

Hinweis zu Parallelinstallationen

Die folgenden zwei Parallelinstallationen sind getestet:

- TIA Portal V14 SP1 und TIA Portal Teamcenter Gateway V14 SP1 mit TIA Portal V13 SP1 und
- TIA Portal V14SP1 mit Teamcenter Gateway V14 SP1 mit STEP 7 Classic 5.5
- TIA Portal V15 und TIA Portal Teamcenter Gateway V15 mit TIA Portal V13 SP2 und
- TIA Portal V15 mit Teamcenter Gateway V15 mit STEP 7 Classic 5.6
- TIA Portal V15.1 und TIA Portal Teamcenter Gateway V15.1 mit TIA Portal V14 SP1 und
- TIA Portal V15.1 und TIA Portal Teamcenter Gateway V15.1 mit STEP 7 Classic 5.6

5.3.4 TIA Portal Teamcenter Gateway Installation starten

Einleitung

Das TIA Portal Teamcenter Gateway kann im Rahmen der Installation des TIA Portals installiert werden. Wird das TIA Portal Teamcenter Gateway mit dem TIA Portal mitinstalliert, so wird immer auch die dem TIA Portal entsprechende Version des TIA Portal Teamcenter Gateway installiert.

Eine Beschreibung zur automatisierten Installation finden Sie auch auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Voraussetzungen für die Installation des TIA Portal Teamcenter Gateway

Für die Installation des TIA Portal Teamcenter Gateway gelten die folgenden Voraussetzungen:

Siehe hierzu: Systemvoraussetzungen für das TIA Portal Teamcenter Gateway (Seite 45)

TIA Portal Teamcenter Gateway mit dem TIA Portal installieren

Für die Installation gehen Sie folgendermaßen vor:

1. Beachten Sie die Voraussetzungen für die Installation des ausgewählten Softwarepakets.
2. Starten Sie das Setup-Programm des gewünschten Produkts durch Einlegen des Installationsdatenträgers in das entsprechende Laufwerk.
3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Vergewissern Sie sich, dass das TIA Portal Teamcenter Gateway in der Produktauswahl angewählt ist und folgen Sie den Hinweisen in den Setup-Dialogen.

Ergebnis

Das TIA Portal Teamcenter Gateway wurde auf Ihrem Computer zusammen mit dem TIA Portal installiert.

Die Installation von TIA Portal Teamcenter Gateway fügt in der Projektansicht im Menü "Projekt" das Untermenü "Teamcenter Projekt öffnen..." ein.

Weiterhin wird im Hauptmenü des TIA Portals im Menü "Extras" das Untermenü "Teamcenter" eingefügt.

5.3.5 TIA Portal Teamcenter Gateway installierte Software anzeigen

Sie können sich jederzeit darüber informieren, welche Software installiert ist. Zusätzlich können Sie sich weitere Informationen zur installierten Software anzeigen lassen.

Vorgehen

Um eine Übersicht über die installierte Software aufzurufen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Hilfe" auf "Installierte Software".
   Der Dialog "Installierte Software" wird geöffnet. Innerhalb des Dialogs sehen Sie die installierten Software-Produkte. Klappen Sie die Einträge auf, um die jeweils installierte Version zu sehen.
2. Wenn Sie sich weitere Informationen zu der installierten Software anzeigen lassen möchten, klicken Sie auf den Link innerhalb des Dialogs "Weitere Informationen zu installierter Software".
   Der Dialog "Detailinformation" wird geöffnet.
3. Wählen Sie in der Bereichsnavigation das Thema, zu dem Sie sich weitere Informationen ansehen möchten.

5.3.6 TIA Portal Teamcenter Gateway Installation prüfen

Vorgehen

Stellen Sie sicher, dass

- entweder Teamcenter Rich Application Client (RAC) 11.2 oder höher
- oder Teamcenter Client Communication System (TCCS) 11.2 oder höher

korrekt auf Ihrem TIA Portal Rechner installiert ist.

Die folgende Tabelle zeigt die entsprechenden Registrierungsschlüssel. Für die dargestellten Szenarien, ist die Verwendung des Betriebssystems Windows 7 (64 Bit) auf Ihrem TIA Portal Rechner erforderlich.

Produkt	64 Bit Installation Registrierungsschlüssel	Wert Name	Wert Daten
TC RAC 11.2	HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Teamcenter**11000_2** [1)]	-	-
TC RAC 11.3	HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Teamcenter**11000_3** [1)]	-	-
TC RAC 11.4	HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Teamcenter**11000_4** [1)]	-	-
TC RAC 11.5	HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Teamcenter**11000_5** [1)]	-	-
TCCS 11.2	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Teamcenter client communication system	"DisplayVersion"	11.2.* [2)]
TCCS 11.3	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Teamcenter client communication system	"DisplayVersion"	11.3* [2)]

Produkt	64 Bit Installation Registrierungsschlüssel	Wert Name	Wert Daten
TCCS 11.4	HKEY_LOCAL_MACHINE\SOFTWARE \Microsoft\Windows\CurrentVersion\ Uninstall\Teamcenter client communication system	"DisplayVersion"	11.4* [2]
TCCS 11.5	HKEY_LOCAL_MACHINE\SOFTWARE \Microsoft\Windows\CurrentVersion\ Uninstall\Teamcenter client communication system	"DisplayVersion"	11.5* [2]

* Es können weitere spezifische Werte vorhanden sein, die Sie ignorieren können. Setzen Sie dafür ein Platzhalterzeichen ein.

[1] TIA Portal Teamcenter Gateway prüft, ob der vorhandene Registrierungsschlüssel mit "11000_" beginnt. Alle früheren Versionen von TC RAC sind ungültig.

[2] TIA Portal Teamcenter Gateway prüft, ob der Wert der angezeigten Version mit "11." beginnt. Alle früheren Versionen von TCCS sind ungültig.

5.3.7 TIA Portal Teamcenter Gateway modifizieren oder hochrüsten

Sie haben die Möglichkeit, installierte Produkte über das Setup-Programm zu ändern oder auf eine neue Version hochzurüsten (Update).

Voraussetzung

- TIA Portal V14 oder höher und TIA Portal Teamcenter Gateway V14 oder höher ist auf Ihrem PG/PC installiert.
- Teamcenter Rich Application Client (nachfolgend "RAC" genannt) oder Teamcenter Client Communication System (TCCS) Version 11.2 oder höher ist auf Ihrem PG/PC installiert.
- Verbindung von FCC/FMS (FileClientCache/FileManagementSystem) zum "Teamcenter Server 11.2" besteht.
- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle aktiven TIA Portal Prozesse sind beendet.

Vorgehen

Um das TIA Portal Teamcenter Gateway zu modifizieren oder hochzurüsten, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog "Installationssprache" zur Auswahl der Installationssprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Installationshinweis lesen" bzw. "Produktinformation lesen".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog "Eine vorhandenen Installation wurde erkannt. Bitte wählen Sie:" wird geöffnet.
6. Aktivieren Sie auf das Optionskästchen "Modifizieren/Hochrüsten" und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog "Bitte wählen Sie die gewünschte Produktkonfiguration. Wenn Sie bereits installierte Bestandteile abwählen, werden diese deinstalliert." wird geöffnet.
7. Wählen Sie die Produkte, die Sie "Modifizieren/Hochrüsten" möchten und klicken Sie auf die Schaltfläche "Weiter".

   Hinweis

   Beachten Sie, dass Sie das Zielverzeichnis nicht ändern können, da die vorhandene Installation verändert wird.

8. Um die Installation fortzusetzen, lesen und akzeptieren Sie alle Lizenzvereinbarungen und klicken Sie auf die Schaltfläche "Weiter".
   Im Dialog "Übersicht" werden die Installationseinstellungen angezeigt.
9. Klicken Sie auf die Schaltfläche "Modifizieren/Hochrüsten".
   Das "Modifizieren/Hochrüsten" der bestehenden Installation wird gestartet und der Dialog "Lizenztransfer" wird geöffnet.

   Hinweis

   Wenn während der Installation kein License Key gefunden wird, erhalten Sie die Möglichkeit diesen auf Ihren PC zu übertragen. Wenn Sie den Lizenztransfer überspringen, können Sie dies später mit dem Automation License Manager nachholen.

   Nach dem "Modifizieren/Hochrüsten" erhalten Sie die Meldung "Das Setup wurde erfolgreich ausgeführt".

10. Um die Installation abzuschliessen, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Das auf Ihrem Computer installierte TIA Portal Teamcenter Gateway wurde erneut installiert.

5.3.8 TIA Portal Teamcenter Gateway Installation reparieren

Sie haben die Möglichkeit, installierte Produkte mithilfe des Setup-Programms durch eine vollständige Reinstallation zu reparieren.

Voraussetzung

- TIA Portal V14 oder höher und TIA Portal Teamcenter Gateway V14 oder höher ist auf Ihrem PG/PC installiert.
- Teamcenter Rich Application Client (RAC) oder Teamcenter Client Communication System (TCCS) Version 11.2 oder höher ist auf Ihrem PG/PC installiert.
- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle aktiven TIA Portal Prozesse sind beendet.

Vorgehen

Um die Option TIA Portal Teamcenter Gateway zu reparieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog "Installationssprache" zur Auswahl der Installationssprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Installationshinweis lesen" bzw. "Produktinformation lesen".
   Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog "Eine vorhandene Installation wurde erkannt. Bitte wählen Sie:" wird geöffnet.
6. Aktivieren Sie auf das Optionskästchen "Reparieren" und klicken Sie auf die Schaltfläche "Weiter".
7. Wählen Sie die Produkte, die Sie "Reparieren" möchten und klicken Sie auf die Schaltfläche "Weiter".
8. Um die Installation fortzusetzen, lesen und akzeptieren Sie alle Lizenzvereinbarungen und klicken Sie auf die Schaltfläche "Weiter".
9. Der Dialog "Übersicht" wird geöffnet.

10. Klicken Sie auf die Schaltfläche "Reparieren".
    Das "Reparieren" der bestehenden Installation wird gestartet.

    Hinweis

    Nach dem "Reparieren" erhalten Sie die Meldung "Das Setup wurde erfolgreich ausgeführt".

11. Um die Installation abzuschließen, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Das auf Ihrem Computer installierte TIA Portal Teamcenter Gateway wurde erneut installiert.

5.3.9 TIA Portal Teamcenter Gateway Deinstallation starten

Das TIA Portal Teamcenter Gateway kann über den Installationsdatenträger deinstalliert werden.

Voraussetzung

- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle aktiven TIA Portal Prozesse sind beendet.

Deinstallation des TIA Portal Teamcenter Gateway mithilfe des Installationsdatenträgers

Um das TIA Portal Teamcenter Gateway zu deinstallieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.
   Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".
   Der Dialog zur Auswahl der Installationssprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms deinstallieren möchten und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog "Eine vorhandene Installation wurde erkannt. Bitte wählen Sie:" wird geöffnet.
4. Aktivieren Sie das Optionskästchen "Deinstallieren" und klicken Sie auf die Schaltfläche "Weiter".
   Der Dialog "Übersicht" wird geöffnet.
5. Klicken Sie auf die Schaltfläche "Deinstallieren".
   Die Deinstallation wird gestartet.

   Hinweis

   Nach der Deinstallation erhalten Sie die Meldung "Das Setup wurde erfolgreich ausgeführt".

6. Um die Deinstallation abzuschließen, klicken Sie auf die Schaltfläche "Beenden".

Ergebnis

Das auf Ihrem Computer installierte TIA Portal Teamcenter Gateway wurde deinstalliert.

5.4 Lizenzen für das TIA Portal Teamcenter Gateway

5.4.1 Lizenzen

Lizenzierung des TIA Portal Teamcenter Gateway

Für das Arbeiten mit dem TIA Portal Teamcenter Gateway benötigen Sie auf jedem Gerät, auf dem Sie TIA Portal Teamcenter Gateway installieren einen gültigen License Key. Der License Key ist auch dann erforderlich, falls auf diesem Gerät das TIA Portal installiert ist.

Sie können den License Key entweder bei der Installation gleich mitinstallieren oder nach der Installation mit dem Automation License Manager übertragen. Beim Arbeiten mit dem TIA Portal in Verbindung mit TIA Portal Teamcenter Gateway wird das Vorhandensein des gültigen License Key geprüft.

Siehe auch

Hinweise zu Lizenzen (Seite 56)

5.4.2 Hinweise zu Lizenzen

Lizenzen für TIA Portal Teamcenter Gateway

Die Lizenzen für das TIA Portal Teamcenter Gateway werden in der Regel auf einem Installationsdatenträger ausgeliefert und können während des Installationsprozesses automatisch übertragen werden.

1. Wenn Sie im TIA Portal im Menü "Extras > Teamcenter" den Befehl "Verbinden" ausführen, wird geprüft, ob der Typ der erforderlichen Lizenz für TIA Portal Teamcenter Gateway vorhanden ist.
   Die folgenden Lizenztypen werden unterschieden:

License Key	Produktname	Standard Lizenztyp	Lizenztyp	Laufzeit
SIFLTEAGAW01500	TIA Portal Teamcenter Gateway	Floating License	unbegrenzt	unbegrenzt
SITTTEAGAW1500	TIA Portal Teamcenter Gateway	Single License	Trial License	21 Tage
SIUPTEAGAW1500	TIA Portal Teamcenter Gateway	Upgrade License	---	unbegrenzt
SIFLTEAGAW01400	TIA Portal Teamcenter Gateway	Dieser License Key ist Voraussetzung für die Hochrüstung auf neuere Versionen.		

2. Wird kein gültiger License Key gefunden, erhalten Sie eine Fehlermeldung.

3. Wird ein gültiger License Key gefunden, so wird der Dialog "Verbinden mit Teamcenter" geöffnet und Sie können eine Verbindung zwischen dem TIA Portal und Teamcenter herstellen.
   Der License Key ist dadurch benutzt.
4. Wenn Sie im TIA Portal im Menü "Extras > Teamcenter" den Befehl "Trennen" ausführen oder wenn das TIA Portal geschlossen wird, sollte der License Key über den Automation License Manager zurückgeschrieben werden.

Hinweis

Weiterführende Informationen, wie Sie mit dem Automation License Manager Ihre Lizenzen verwalten können, finden Sie in der mit dem Automation License Manager mitgelieferten Dokumentation.

5.5 Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen

5.5.1 Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen

Die nachfolgenden Kombinationen von Teamcenter-Server-Versionen und Teamcenter-Client-Versionen werden von TIA Portal Teamcenter Gateway unterstützt.

Tabelle 5-1 Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen

TIA Portal Teamcenter Gateway	Teamcenter Client (RAC oder TCCS)	Teamcenter-Server
V14.0	11.2	11.2
V14 SP1	11.2	11.2
V14 SP1 Update 1	11.2	11.2
V15.0	11.2	11.2
V15.0	11.3	11.3
V15.1	11.2	11.2
V15.1	11.3	11.3
V15.1	11.4	11.4
V15.1	11.5	11.5

Hinweis

Unterstützte Teamcenter-Server-Versionen und Teamcenter-Client-Versionen

Nur die in der Tabelle aufgeführten Kombinationen werden von TIA Portal Teamcenter Gateway unterstützt.

Alle anderen Kombinationen von Teamcenter-Server-Versionen und Teamcenter-Client-Versionen (RAC, TCCS) sind **nicht getestet** und werden von TIA Portal Teamcenter Gateway **nicht unterstützt**.

5.6 Grundlagen zum Arbeiten mit TIA Portal Teamcenter Gateway

Einführung

TIA Portal Teamcenter Gateway bietet Ihnen folgende Einsatzmöglichkeiten:

- Sie können eine Verbindung zwischen dem TIA Portal und Teamcenter herstellen.
- Sie können im TIA Portal, Teamcenter spezifische Operationen durchführen.
- Sie können TIA Portal-Projekte, die mit Teamcenter verbunden sind, dem Multiuser-Server als Multiuser-Serverprojekte hinzufügen.

Die Verwendung von TIA Portal Teamcenter Gateway zusammen mit dem TIA Portal bietet Ihnen folgende Vorteile:

- Speichern von TIA Portal-Projekten oder globale Bibliotheken in Teamcenter.
- Beim Speichern von TIA Portal-Projekten oder globale Bibliotheken in Teamcenter, können Sie diesen Elementtypen oder abhängige Typen zuweisen.
- Änderungen an mit Teamcenter verknüpften TIA Portal-Projekten oder globalen Bibliotheken speichern.
- Revisionieren von mit Teamcenter verknüpften TIA Portal-Projekten oder globalen Bibliotheken.
- Globale Bibliotheken oder TIA Portal-Projekte, die in Teamcenter gespeichert sind, im Modus "nur lesend" oder "schreibend" öffnen.
- Änderungen, die Sie an Multiuser-Serverprojekten vorgenommene haben, können Sie als neue Version (Dataset Version) nach Teamcenter speichern.

5.7 TIA Portal Teamcenter Gateway Cache Pfad festlegen

Einleitung

Wenn Sie mit TIA Portal Teamcenter Gateway arbeiten, können Sie in den Einstellungen vom TIA Portal den "TIA Portal Teamcenter Gateway Cache Pfad" einstellen.

Voraussetzung

Sie haben das TIA Portal ab V14 und TIA Portal Teamcenter Gateway installiert.

Teamcenter Cache Pfad einstellen

Um den "TIA Portal Teamcenter Gateway Cache Pfad" einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie dazu im TIA Portal unter "Extras" > "Einstellungen" > "Allgemein" > "Teamcenter Gateway" > im Bereich "TIA Portal Teamcenter Gateway Cache Pfad" auf die Schaltfläche "Durchsuchen".
   Der Dialog "Ordner suchen" wird angezeigt.
2. Wählen Sie den entsprechenden Ordner aus, bestätigen Sie die Auswahl durch klicken auf die Schaltfläche "OK".
   Der Dialog "TIA Portal Teamcenter Gateway Cache Pfad" wird angezeigt.
   – Klicken auf die Schaltfläche "OK" bewirkt, dass der ausgewählte Ordner im Textfeld angezeigt wird.
   – Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.

   Alternativ können Sie den absoluten Verzeichnispfad auch "händisch" in das Textfeld eintragen.

   Hinweis

   Szenarien, bei denen zwei TIA Portal-Instanzen denselben Teamcenter Cache Pfad verwenden, werden nicht unterstützt.

5.8 Einchecken beim Schließen des Projekts/der Bibliothek bestätigen

Einchecken (automatisches Einchecken)

Beim Schließen eines TIA Portal-Projekts oder einer globalen Bibliothek, die in Teamcenter gespeichert ist, erhalten Sie eine entsprechende Meldung das TIA Portal-Projekt oder die globale Bibliothek einzuchecken. Die Anzeige der Meldung ist davon abhängig, ob das Optionskästchen "Einchecken beim Schließen des Projekts/der Bibliothek bestätigen" aktiviert oder deaktiviert ist. Standardmäßig ist das Optionskästchen "Einchecken beim Schließen des Projekts/der Bibliothek bestätigen" deaktiviert.

Vorgehen

Um das Einchecken eines TIA Portal-Projekts oder einer globalen Bibliothek zu deaktivieren oder zu aktivieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie unter Extras" > "Einstellungen" > "Allgemein" > "Teamcenter Gateway" > im Bereich "Einchecken" das Optionskästchen "Einchecken beim Schließen von Projekt/ Bibliothek bestätigen".

- Standardmäßig ist das Optionskästchen "Einchecken beim Schließen des Projekts/der Bibliothek bestätigen" deaktiviert, dadurch wird das TIA Portal-Projekt oder die globale Bibliothek ohne vorherige Abfrage eingecheckt.
- Wenn das Optionskästchen "Einchecken beim Schließen des Projekts/der Bibliothek bestätigen" aktiviert ist, wird eine Meldung angezeigt:
  - Klicken auf die Schaltfläche "Ja" bewirkt, dass das TIA Portal-Projekt oder die globale Bibliothek eingecheckt wird.
  - Klicken auf die Schaltfläche "Nein" bewirkt, dass das TIA Portal-Projekt oder die globale Bibliothek nicht eingecheckt wird.

Ergebnis

Das TIA Portal-Projekt oder die globale Bibliothek wird eingecheckt.

5.9 Verbindung zwischen dem TIA Portal und Teamcenter herstellen und trennen

5.9.1 Verbindung zwischen dem TIA Portal und Teamcenter herstellen

Um mit dem TIA Portal Teamcenter Gateway zu arbeiten, müssen Sie sich mit Ihrem gültigen Teamcenter Benutzer-Login, an einem spezifischen Teamcenter Server anmelden.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass:

- Sie das TIA Portal gestartet haben.

Vorgehen

Um eine Verbindung zwischen dem TIA Portal und Teamcenter herzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Hauptmenü des TIA Portal im Menü "Extras > Teamcenter" den Befehl "Verbinden".
2. Der Dialog "Verbinden mit Teamcenter" wird geöffnet, geben Sie die folgenden Anmeldedaten ein:
   - Benutzername: Geben Sie Ihren "Teamcenter Benutzernamen" ein.
   - Kennwort: Geben Sie Ihr "Teamcenter Kennwort" ein.
     Geben Sie in die Felder "Benutzername" und "Kennwort" Ihre Teamcenter-Anmeldedaten ein.
   - Gruppe: Geben Sie Ihre spezifische Teamcenter Gruppe ein, die für die aktuelle Anwendungssitzung benutzt werden soll.
     Wenn Sie dieses Feld 'leer' lassen, werden die in Teamcenter vorbelegten Einstellungen übernommen.
   - Rolle: Geben Sie Ihre spezifische Teamcenter Rolle ein, die für die aktuelle Anwendungssitzung benutzt werden soll.
     Wenn Sie dieses Feld 'leer' lassen, werden die in Teamcenter vorbelegten Einstellungen übernommen.
   - Host: Geben Sie den Server Namen im folgenden Format ein:
     http://<Server Name>:<Port>
   - Instanz: Geben Sie den Namen der Teamcenter Instanz ein (z.B. tc).
3. Klicken Sie auf die Schaltfläche "OK" um die Verbindung zu Teamcenter herzustellen.
   Während des Verbindungsaufbaus wird der Dialog "Verbindung wird herstellt..." angezeigt. Dieser Hintergrundprozess kann nicht abgebrochen werden.
   Sobald die Verbindung zu Teamcenter hergestellt ist, öffnet sich ein Dialog, der die erfolgreiche Verbindung anzeigt.
4. Bestätigen Sie diesen Dialog durch klicken auf "OK".

5. Bei einer fehlgeschlagenen Authentifizierung wird ein Dialog mit möglichen Fehlermeldungen ausgegeben. Diese Fehlermeldung enthält Hinweise auf die Fehlerbehebung.
   Mögliche Fehler:
   – Die Kombination von Benutzername und Password ist nicht gültig.

   Wählen Sie erneut im Hauptmenü des TIA Portal im Menü "Extras > Teamcenter" den Befehl "Verbinden". Beheben Sie die Fehler im Dialog "Verbinden mit Teamcenter".
6. Während des Verbindungsaufbaus werden die "Zusätzlichen Eigenschaften" der Projekt- / Bibliothek-Elementtypen gelesen. Wenn beim Lesen der "Zusätzlichen Eigenschaften" ein Fehler auftritt, wird ein Dialog mit möglichen Fehlermeldungen ausgegeben.
7. Bestätigen Sie diesen Dialog durch klicken auf "OK".

Hinweis

Wenn ein erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Verbinden" deaktiviert.

Wenn ein nicht erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Trennen" deaktiviert.

Lizenzprüfung

Hinweis

Wenn Sie im Hauptmenü, des TIA Portal im Menü "Extras > Teamcenter" den Befehl "Verbinden" ausführen, wird geprüft, ob die passenden und gültigen Lizenzen für TIA Portal Teamcenter Gateway vorhanden sind.

Weitere Informationen zu passenden und gültigen Lizenzen finden Sie im Kapitel "Lizenzen für das TIA Portal Teamcenter Gateway" unter Hinweise zu Lizenzen (Seite 56).

Ergebnis

Die Verbindung zwischen dem TIA Portal und Teamcenter ist hergestellt.

Verbindung als Teamcenter Mitglied mit zwei verschiedenen Teamcenter Benutzergruppen- oder Benutzerrollen herstellen

Hinweis

Ein Wechsel der aktuellen Benutzergruppen- oder Benutzerrolle ist während einer aktiven Anwendungssitzung nicht möglich.

Um einen Wechsel Ihrer Benutzergruppe oder Benutzerrolle vorzunehmen, gehen Sie folgendermaßen vor:

1. Beenden Sie die aktuelle Anwendungssitzung, mit dem Menü "Extras > Teamcenter" und dem Befehl "Trennen", zwischen dem TIA Portal und Teamcenter.
2. Stellen Sie dann im Hauptmenü des TIA Portals mit dem Menü "Extras > Teamcenter" und dem Befehl "Verbinden" eine neue Verbindung zwischen dem TIA Portal und Teamcenter her.
3. Nehmen Sie im Dialog "Verbinden mit Teamcenter" die gewünschten Einstellungen vor.

5.9.2 Verbindung zwischen dem TIA Portal und Teamcenter trennen

Vorgehen

Um die Verbindung zwischen dem TIA Portal und Teamcenter zu trennen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Hauptmenü des TIA Portal im Menü "Extras > Teamcenter" den Befehl "Trennen".

Ergebnis

Die Verbindung zwischen dem TIA Portal und Teamcenter ist getrennt. Es können keine weiteren Teamcenter spezifischen Operationen ausgeführt werden.

5.9.3 Verbindungsstatus zwischen TIA Portal und Teamcenter anzeigen

Verbindungsstatus anzeigen

Folgende Darstellungen zur Anzeige des Verbindungsstatus zwischen dem TIA Portal und Teamcenter sind möglich:

- ein erfolgreicher Verbindungsaufbau, zwischen dem TIA Portal und Teamcenter, wird im TIA Portal in der Statuszeile, am rechten unteren Rand, durch ein Teamcenter Symbol dargestellt. Das Teamcenter Symbol wird farbig dargestellt, sobald der Verbindungsaufbau hergestellt wurde.
  Wenn Sie mit dem Mauszeiger etwas länger über dem Teamcenter Symbol verweilen, wird ein Tooltip eingeblendet, der Ihnen die Verbindungsinformation im Klartext anzeigt.
- ein nicht erfolgreicher Verbindungsaufbau, zwischen dem TIA Portal und Teamcenter, wird im TIA Portal in der Statuszeile, am rechten unteren Rand, durch ein Teamcenter Symbol dargestellt. Das Teamcenter Symbol wird schwarz-weiß dargestellt, wenn kein Verbindungsaufbau hergestellt wurde.
  Wenn Sie mit dem Mauszeiger etwas länger über dem Teamcenter Symbol verweilen, wird ein Tooltip eingeblendet, der Ihnen die Verbindungsinformation im Klartext anzeigt.
- wenn der Teamcenter-Server nicht für UTF-8 aktiviert ist, wird das im TIA Portal in der Statuszeile, am rechten unteren Rand, durch das folgende Teamcenter Symbol dargestellt.

Verbindungsstatus im Register "Info" anzeigen

Im Register "Info" wird der erfolgreiche Verbindungsaufbau zwischen dem TIA Portal und Teamcenter in folgendem Format angezeigt:

- http://hostname [Benutzername] - Gruppe / Rolle
  Beispiel: http://WIN-MLMCU5LQTP6:8080 [Benutzername] - Gruppe / Rolle

Im Register "Info" wird ein nicht erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter in folgendem Format angezeigt:

- Es besteht keine Verbindung mit Teamcenter.

Verbindungsstatus in der TIA Portal Statuszeile anzeigen

In der TIA Portal Statuszeile, wird der erfolgreiche Verbindungsaufbau zwischen dem TIA Portal und Teamcenter in folgendem Format angezeigt:

- http://hostname [Benutzername] - Gruppe / Rolle
  Beispiel: http://WIN-MLMCU5LQTP6:8080 [Benutzername] - Gruppe / Rolle

In der TIA Portal Statuszeile, wird ein nicht erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter in folgendem Format angezeigt:

- Es besteht keine Verbindung mit Teamcenter.

Hinweis

Wenn ein erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Verbinden" deaktiviert.

Wenn ein nicht erfolgreicher Verbindungsaufbau zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Trennen" deaktiviert.

5.10 Unterscheidung zwischen TIA Portal-Projekten und globalen Bibliotheken

Einleitung

TIA Portal-Projekte und globale Bibliotheken werden in Teamcenter über den Namen des Datasets und die Dateierweiterung identifiziert.

Name des Datasets

Beachten Sie bitte, dass:

- Beim Speichern eines TIA Portal-Projekt mit der Teamcenter-Operation "Speichern als neues Element" und "Speichern als neue Revision" in Teamcenter der Name des TIA Portal-Projekts als Datasetname (z.B. "TIAProjectDataset") verwendet wird.
- Beim Speichern einer globalen Bibliothek mit der Teamcenter-Operation "Speichern als neues Element" und "Speichern als neue Revision" in Teamcenter der Name der globalen Bibliothek als Datasetname (z.B. "TIALibraryDataset") verwendet wird.

Dateierweiterungen

Beachten Sie bitte, dass:

- Beim Speichern eines TIA Portal-Projekts mit der Teamcenter-Operation "Speichern als neues Element" und "Speichern als neue Revision" das TIA Portal-Projekt in Teamcenter mit der Dateierweiterung *.zap* gespeichert wird.
- Beim Speichern einer globalen Bibliothek mit der Teamcenter-Operation "Speichern als neues Element" und "Speichern als neue Revision" die globale Bibliothek in Teamcenter mit der Dateierweiterung *.zal* gespeichert wird.

5.10.1 Unterschiedliche Datasettypen in eine neue Revision kopieren

TIA Portal-Projekte und globale Bibliotheken mit der Teamcenter-Operation "Speichern als neue Revision..." in TIA Portal Teamcenter Gateway speichern

Mit der Teamcenter-Operation "Speichern als neue Revision..." in TIA Portal Teamcenter Gateway, legen Sie während des Speicherns fest, welche angehängten Datasets (non-TIA Datasets) von der alten in die neue Revision übertragen werden.

Dieses Verhalten ist von den Kopierregeln "Deep Copy Rules" abhängig.

Weitere Informationen zu den Kopierregeln "Deep Copy Rules", finden Sie in der Bedienungsanleitung "Teamcenter 11 – Business Modeler IDE" im Abschnitt "Deep Copy Rules".

Arbeiten mit der Funktion "Teamcenter Deep Copy Rule"

Die Funktion "Teamcenter Deep Copy Rule" unterstützt, beim Verwenden der Teamcenter-Operation "Speichern als neue Revision....", die folgenden Aktionen:

- CopyAsObject
- CopyAsReference
- NoCopy
- RelatedToLatest
- ReviseAndReleateToLatest
- SystemCopy
- ReviseObject

5.11 TIA Portal-Projekte mit dem TIA Portal Teamcenter Gateway verwalten

5.11.1 TIA Portal-Projekt als neues Element in Teamcenter speichern

Voraussetzung

Im Folgenden wird vorausgesetzt, dass:

- zwischen dem TIA Portal und Teamcenter eine aktive Verbindung besteht.
- keine Multiselektion von mehreren Projekten unterstützt wird.

Vorgehen

Um ein TIA Portal-Projekt zum ersten Mal in Teamcenter zu speichern, gehen Sie folgendermaßen vor:

1. Öffnen Sie im TIA Portal das TIA Portal-Projekt, das Sie als neues Element in Teamcenter speichern wollen.
2. Selektieren Sie in der Sicht "Projektnavigation" das TIA Portal-Projekt zum Speichern in Teamcenter.
3. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neues Element...".
   Der Dialog "Eigenschaften neues Element" wird geöffnet.
   Das Register "Allgemein" wird standardmäßig angezeigt.
4. Geben Sie im Register "Allgemein", im Feld "Name", einen Namen ein oder übernehmen Sie die vorgeschlagenen Daten. Standardmäßig wird der Name des ausgewählten TIA Portal-Projekts angezeigt.
   Maximal eingebbar sind 94 Zeichen.
5. Wählen Sie im Register "Allgemein", im Feld "Elementtyp", einen Elementtyp, für das TIA Portal-Projekt, das Sie als neues Element in Teamcenter speichern wollen aus.
   – Wählen Sie im Feld "Elementtyp" einen abhängigen Typen aus.
     Standardmäßig ist der Typ "TIA Project" ausgewählt.
     Die Klappliste ist nur editierbar, wenn abhängige Typen verfügbar sind.
6. Weisen Sie im Register "Allgemein" dem Feld "Element-ID / Revision-ID" eine bereits gültige oder beliebige Teamcenter Element-ID / Revision-ID zu.
   Wenn Sie das Feld "Element-ID / Revision-ID" 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID zugewiesen.

7. Durch klicken auf die Schaltfläche “Zuweisen” wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID dem ausgewählten Projekt zugewiesen.
   Ergebnis:
   - Die zugewiesene Element-ID / Revision-ID wird im jeweiligen Textfeld angezeigt.
   - Die Textfelder sind nicht editierbar und die Schaltfläche "Zuweisen" ist deaktiviert.

   Hinweis

   Anpassung der Elementtypen

   Die Anpassung, der Elementtypen, hat auf die Felder "Element-ID / Revision-ID" und die Schaltfläche "Zuweisen" folgende Auswirkungen:
   1. die den Feldern "Element-ID / Revision-ID" zugewiesenen IDs werden zurückgesetzt.
   2. die Schaltfläche "Zuweisen" wird auf bedienbar gesetzt.

8. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Projekt", das Teamcenter Projekt aus, dem Sie das TIA Portal-Projekt zuweisen wollen.
   - Klicken Sie auf die Schaltfläche "Durchsuchen" (...)..
     Der Dialog "Projekte auswählen" wird geöffnet.
   - Wählen Sie die entsprechenden Teamcenter Projekte aus und bestätigen Sie jede Auswahl mit klicken auf die Schaltfläche "OK".
     Klicken auf die Schaltfläche "OK" bewirkt, dass die ausgewählten Teamcenter Projekte getrennt durch Komma im Feld "Teamcenter Projekt" angezeigt werden.
     Wenn Sie das Feld "Teamcenter Projekt" 'leer' lassen, wird das TIA Portal-Projekt keinem Teamcenter Projekt zugewiesen.
9. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Ordner" den Ordner aus, dem das neu erstelle Element zugewiesen werden soll.
   - Klicken Sie auf die Schaltfläche "Durchsuchen" (...)..
     Der Dialog "Teamcenter Ordner auswählen" wird geöffnet.
     Die Vorbelegung für diesen Ordner ist *"Home*”.
     Alle neu erstellen Elemente werden standardmäßig in Ihrem persönlichen Verzeichnis *"Home"* abgelegt.
   - Wählen Sie den Ordner "*Home*” aus, bestätigen Sie die Auswahl durch klicken auf die Schaltfläche "OK".
     Klicken auf die Schaltfläche "OK" bewirkt, dass der ausgewählte Ordner im Textfeld angezeigt wird.
     Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.
10. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
    Maximal eingebbar sind 240 Zeichen.

11. Bestätigen Sie Ihre Eingabe mit "OK".
    Das TIA Portal-Projekt wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert.
    Sobald das TIA Portal-Projekt in Teamcenter gespeichert wurde, wird der Dialog "Das Projekt wurde erfolgreich in Teamcenter gespeichert" geöffnet.
    – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".
      Ein Überlagerungssymbol zeigt im TIA Portal in der "Projektnavigation" an, dass das TIA Portal-Projekt mit Teamcenter verbunden ist.
12. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass der Dialog "Eigenschaften neues Element" geschlossen wird.

Ergebnis

Nachdem das TIA Portal-Projekt in Teamcenter gespeichert wurde, steht Ihnen im Ordner "Home", ein neues Element inkl. Element-Revision mit der vergebenen Element-ID / Revision-ID zur Verfügung. Das Dataset des TIA Portal-Projekts ist in Teamcenter als Zap-Datei unter der jeweiligen Revision-ID gespeichert.

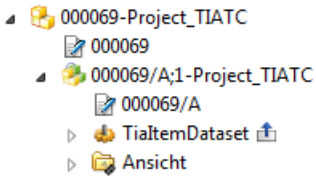

Das zugehörige Dataset ist "Ausgecheckt" und mit dem Symbol gekennzeichnet. Das können Sie in der Ordneransicht von Teamcenter sehen.

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen für das Arbeiten mit dem TIA Portal-Projekt zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element... .
- Einchecken.
- Auschecken rückgängig machen.

Siehe auch

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

TIA Portal-Projekt einchecken (Seite 78)

Auschecken eines TIA Portal-Projekts rückgängig machen (Seite 87)

Änderungen an einem mit Teamcenter verbunden TIA Portal-Projekt speichern (Seite 79)

Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 73)

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 72)

Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 72)

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 74)

Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 75)

5.11.1.1 Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)

Im Register "Allgemein" des Dialogs "Speichern als neues Element" definieren Sie die Eigenschaften eines TIA Portal-Projekts das Sie in Teamcenter speichern wollen.

Siehe auch

Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 73)

5.11.1.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)

Im Register "Zusätzliche Eigenschaften" bearbeiten Sie die Werte der hochgeladenen zusätzlichen Eigenschaften eines TIA Portal-Projekts.

Im Register "Zusätzliche Eigenschaften" haben Sie die Möglichkeit, Werte von hochgeladene Elementtypen oder abhängigen Typen einzugeben. Der Inhalt des Registers "Zusätzliche Eigenschaften" ist von den verwendeten Elementtypen, den abhängigen Typen oder den Eigenschaften von geerbten Typen abhängig.

Siehe auch

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 74)

Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 75)

5.11.1.3 Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)

Aufbau des Registers "Allgemein" im Dialog "Speichern als neues Element"

Die Ansicht des Registers "Allgemein" besteht aus folgenden Feldern und Schaltflächen:

Feld	Beschreibung
Name:	Eingabefeld für den Namen unter dem Sie das TIA Portal-Projekt als neues Element in Teamcenter speichern wollen. Standardmäßig wird der Name des ausgewählten TIA Portal-Projekts angezeigt.
Elementtyp:	Eingabefeld für den Elementtyp des TIA Portal-Projekts. Standardmäßig ist der Typ TIA Project ausgewählt.
Element-ID / Revision-ID:	Eingabefeld für die Element-ID / Revision-ID die Sie dem neuen Element zuweisen wollen.
Teamcenter Projekt:	Auswahlfeld zur Auswahl des Teamcenter Projekts, dem Sie das neue Element zuweisen wollen.
Teamcenter Ordner:	Auswahlfeld zur Auswahl des Teamcenter Ordners, in den Sie das neue Element speichern wollen.
Kommentar:	Eingabefeld für einen Kommentar.

Schaltfläche	Beschreibung
Klappliste	Durch klicken auf die Klappliste können Sie abhängige Typen auswählen und hinzufügen. Die Klappliste ist nur editierbar, wenn abhängige Typen definiert sind.
Zuweisen	Durch klicken auf diese Schaltfläche weisen Sie dem TIA Portal-Projekt, automatisch eine Element-ID / Revision-ID zu.
"Durchsuchen" (...) - (Teamcenter Projekt)	Durch klicken auf diese Schaltfläche wird der Dialog "Projekte auswählen" geöffnet.
"Durchsuchen" (...) - (Teamcenter Ordner)	Durch klicken auf diese Schaltfläche wird der Dialog "Teamcenter Ordner auswählen" geöffnet.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Siehe auch

Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 72)

5.11.1.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte)

Aufbau des Registers "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Die im Register "Zusätzliche Eigenschaften" tabellarisch angezeigten Informationen, sind davon abhängig:

- welchen Elementtypen oder abhängige Typen Sie ausgewählt haben,
- welche Eigenschaften die ausgewählten Elementtypen und abhängigen Typen geerbt haben, und welche Eigenschaften in der Attribut-Mapping-Datei abgebildet sind.

Eigenschaften des Registers "Zusätzliche Eigenschaften"

Im Register "Zusätzliche Eigenschaften" werden die Eigenschaften von Elementtypen und abhängigen Typen im Modus lesend/schreibend angezeigt.

Die Ansicht des Registers "Zusätzliche Eigenschaften" besteht aus folgenden Spalten und Schaltflächen:

Spalte	Bedeutung
Name	Name der zusätzlichen Eigenschaften.
Typ	Art der zusätzlichen Eigenschaften.
Wert	Wert der zusätzlichen Eigenschaften.

Schaltfläche	Bedeutung
Klappliste	Durch klicken auf die Klappliste können Sie Pflichtattribute hinzufügen.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Hinweis zur Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Folgende Regeln gelten für die Bedienbarkeit der Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element":

- Wenn die erforderlichen Pflichtattribute nicht befüllt sind, ist die Schaltfläche "OK" gegraut und nicht bedienbar.
- Wenn keine Pflichtattribute vorhanden sind oder wenn alle Pflichtattribute befüllt sind, ist die Schaltfläche "OK" bedienbar.

Siehe auch

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 72)

Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 75)

5.11.1.5 Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)

In der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" werden, die in den Teamcenter Server, hochgeladene Attribute eines TIA Portal-Projekts im Modus lesend/ schreibend angezeigt.

Voraussetzungen um die Werte von hochgeladenen Attributen eines TIA Portal-Projekts zu bearbeiten

Um die Werte der hochgeladenen Attribute eines TIA Portal-Projekts zu bearbeiten, wird vorausgesetzt, dass:

- die Attribute des ausgewählten TIA Portal-Projekt Elementtyps hochgeladen sind.
- die Attribute der ausgewählten Element-Revision des TIA Portal-Projekt Elementtyps hochgeladen sind.
- die hochgeladenen Attribute des Element Masterformulars mit dem TIA Portal-Projekt Elementtyp verbunden sind.
- die hochgeladenen Attribute des Element-Revision Masterformulars mit der TIA Portal-Projekt Element-Revision verbunden sind.

Hinweis

Beachten Sie, dass

nur die oben genannten Szenarien ab TIA Portal Teamcenter Gateway V14SP1 unterstützt werden. Alle anderen Szenarien werden ignoriert.

Bitte beachten Sie:

- wenn keine Attribute in den Teamcenter Server hochgeladen sind, wird im Register “Zusätzliche Eigenschaften” die folgende Information angezeigt:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.
- wenn das Lesen von hochgeladenen Attributen während des Verbindungsaufbaus mit TIA Portal Teamcenter Gateway fehlgeschlagen ist, in der Registerkarte “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen

Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften"

Beachten Sie Folgendes:

Bei Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften", wird immer nur der Eintrag des Attributs aus dem Register "Allgemein" in Teamcenter gespeichert.

Mehrfachverwendung von Attributen im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Beachten Sie Folgendes:

Bei Mehrfachverwendung desselben Attributs, im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element", wird immer nur der erste Eintrag des Attributs aus der tabellarischen Ansicht in Teamcenter gespeichert.

Verwendung von Sonderzeichen für den Datentyp 'Char'

Nachfolgend werden die Namenskonventionen für den Datentyp 'Char', die Sie beim "Speichern als neues Element" beachten müssen beschrieben.

Beachten Sie Folgendes:

- Verwenden Sie in den Eingabefeldern des Datentyps 'Char', im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element", nur die ASCII-Werte 32-126.

Siehe auch

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 72)

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 74)

5.11.2 TIA Portal-Projekt auschecken

Um ein oder mehrere TIA Portal-Projekte exklusiv für sich zur Bearbeitung zu reservieren, können Sie die TIA Portal-Projekte auschecken. Dadurch sind die TIA Portal-Projekte für andere Benutzer gesperrt.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal-Projekt mit Teamcenter verbunden ist.
- das TIA Portal-Projekt im Modus "nur lesend" geöffnet ist.
- keine Multiselektion von mehreren TIA Portal-Projekten unterstützt wird.

Vorgehen

Um ein TIA Portal-Projekt zu bearbeiten, müssen Sie dieses "Auschecken", gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das entsprechende TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Auschecken".

Hinweis

TIA Portal-Projekt auschecken

Beachten Sie: wenn das TIA Portal-Projekt lokal geändert wurde, oder die Dataset-Version älter ist, als die aktuell verfügbare Dataset-Version in Teamcenter, der Dialog "Bestätigen Sie das Auschecken" geöffnet wird.

1. Klicken auf die Schaltfläche "Ja" bewirkt, dass die aktuell verfügbare Dataset-Version überschrieben wird.
2. Klicken auf die "Schaltfläche "Nein" bewirkt, dass die aktuell verfügbare Dataset-Version als lokale Kopie geöffnet wird.
3. Durch klicken auf die Schaltfläche "Abbrechen" wird das Auschecken abgebrochen.

Wenn Sie das "Auschecken" fortsetzen, wird die Dataset-Version und die zugehörige Revision des TIA Portal-Projekts in Teamcenter ausgecheckt. Die Dataset-Version ist für die Bearbeitung durch andere Benutzer gesperrt.

Ergebnis

Die TIA Portal-Projekt wird wieder im Modus "Schreiben" geöffnet.

Die Dataset-Version des TIA Portal-Projekts ist ausgecheckt und in Teamcenter durch das folgende Symbol gekennzeichnet.

Die Dataset-Version ist für die Bearbeitung durch andere Benutzer gesperrt.

Ergebnis

Im TIA Portal stehen Ihnen nach erfolgreichem "Auschecken" folgende Teamcenter-Operationen für das Arbeiten mit dem TIA Portal-Projekt zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.

Siehe auch

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69)

Auschecken eines TIA Portal-Projekts rückgängig machen (Seite 87)

TIA Portal-Projekt einchecken (Seite 78)

5.11.3 TIA Portal-Projekt einchecken

TIA Portal-Projekte, die Sie nach Teamcenter gespeichert haben, befinden sich im Zustand "ausgecheckt". Dadurch sind diese TIA Portal-Projekte für andere Benutzer gesperrt.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass:

- das TIA Portal-Projekt mit Teamcenter verbunden ist.
- das Dataset der zugehörigen Revision des TIA Portal-Projekts durch Sie ausgecheckt ist.
- keine Multiselektion von mehreren TIA Portal-Projekten unterstützt wird.

Vorgehen

Um die Bearbeitungssperre des TIA Portal-Projekts aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)", das entsprechende TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Einchecken".

Automatisches einchecken

Das automatisches einchecken erfolgt beim Schließen des TIA Portal-Projekts im TIA Portal unter folgenden Bedingungen:

- Wenn die "VersionInfo.xml-Datei" für ein TIA Portal-Projekt vorhanden ist.
- Wenn das TIA Portal-Projekt im Modus "Schreiben" geöffnet ist.

Das automatisches einchecken erfolgt beim Schließen des TIA Portal-Projekts im TIA Portal unter folgenden Bedingungen **nicht**:

- Wenn die "VersionInfo.xml-Datei" für ein TIA Portal-Projekt nicht vorhanden ist.
- Wenn das TIA Portal-Projekt im Modus "Lesen/Schreiben" geöffnet ist.
- Wenn das TIA Portal-Projekt in RAC ausgecheckt ist.

Ergebnis

Das TIA Portal-Projekt wird wieder im Modus "Lesen" geöffnet.

Das TIA Portal-Projekt ist in Teamcenter eingecheckt und in der Ordneransicht von Teamcenter wird das folgende Symbol entfernt.

Die Bearbeitungssperre des TIA Portal-Projekts ist aufgehoben.

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen für das Arbeiten mit TIA Portal-Projekt zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Auschecken.

Siehe auch

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69)

TIA Portal-Projekt auschecken (Seite 76)

5.11.4 Änderungen an einem mit Teamcenter verbunden TIA Portal-Projekt speichern

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal mit Teamcenter verbunden ist.
- sich die Dataset Version von der lokalen Kopie unterscheidet.
- das ausgewählte TIA Portal-Projekt mit Teamcenter verbunden ist und Änderungen vorgenommen wurden.
- keine Multiselektion von mehreren TIA Portal-Projekten unterstützt wird.

Hinweis

Beachten Sie folgendes:

- Wenn keine Verbindung zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Teamcenter > Speichern" deaktiviert.

Vorgehen

Um Änderungen an einem mit Teamcenter verbundenen TIA Portal-Projekt zu speichern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern".
   Die neue Dataset-Version des TIA Portal-Projekts ist in Teamcenter als Zap-Datei unter der jeweiligen Revision-ID exportiert.

Einschränkungen

Folgende Einschränkungen sind zu beachten:

In Teamcenter kann die Anzahl der gespeicherten Dataset-Versionen durch einen Teamcenter Administrator konfiguriert werden (default = 3). Wenn das Maximum erreicht ist, wird die älteste Dataset-Version gelöscht und die neue Dataset-Version erstellt.

Ergebnis

In Teamcenter wurde eine neue Dataset-Version des TIA Portal-Projekts erstellt.

Im TIA Portal stehen Ihnen nach dem "Speichern" folgende Teamcenter-Operationen für das Arbeiten mit TIA Portal-Projekt zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.
- Export....

Siehe auch

Änderungen an einer mit Teamcenter verbunden globalen Bibliothek speichern (Seite 135)

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69)

TIA Portal-Projekt einchecken (Seite 78)

Auschecken eines TIA Portal-Projekts rückgängig machen (Seite 87)

5.11.5 Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal-Projekt mit Teamcenter verbunden ist.
- keine Multiselektion von mehreren TIA Portal-Projekten unterstützt wird.

Vorgehen

Um von einem TIA Portal-Projekt, das nach Teamcenter gespeichert ist, eine neue Revision zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das entsprechende "TIA Portal-Projekt".
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neue Revision...".
   Bei dieser Funktion wird das aktuell geöffnete TIA Portal-Projekt als Basis für das Erstellen der neuen Revision verwendet.
   - Der Dialog "Speichern als neue Revision" wird geöffnet.
   - Das Register "Allgemein" wird standardmäßig angezeigt.
   - Im Feld "Name" wird der Name des ausgewählten TIA Portal-Projekts, (z.B. t4tia_object_name) nur lesend angezeigt.
   - Im Feld "Element-ID" wird die "Element-ID" des ausgewählten TIA Portal-Projekts nur lesend angezeigt.
3. Optional: Weisen Sie im Register "Allgemein" im Feld "Revision-ID" dem TIA Portal-Projekt eine gültige Teamcenter Revision-ID zu.
   Maximal eingebbar sind 32 Zeichen.
4. Durch klicken auf die Schaltfläche “Zuweisen” wird durch Teamcenter automatisch die nächstmögliche Revision-ID dem TIA Portal-Projekt zugewiesen.
   Ergebnis:
   - Die zugewiesene Revision-ID wird im Feld "Revision-ID" angezeigt.
   - Das Feld "Revision-ID" ist nicht mehr editierbar.
   - Die Schaltfläche "Zuweisen" ist deaktiviert.

   Wenn Sie das Feld 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Revision-ID vergeben.
5. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
   Maximal eingebbar sind 240 Zeichen.
   Standardmäßig werden die Kommentare der Vorgängerversion übernommen. Sie können die Kommentare aus der Vorgängerversion bearbeiten oder unverändert beibehalten

6. Bestätigen Sie Ihre Eingabe mit "OK".
   Das TIA Portal-Projekt wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert.
   Sobald das TIA Portal-Projekt in Teamcenter gespeichert wurde, öffnet sich der Dialog "das TIA Portal-Projekt wurde erfolgreich in Teamcenter gespeichert".
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".
     Ein Überlagerungssymbol zeigt im TIA Portal in der "Projektnavigation" an, dass das TIA Portal-Projekt mit Teamcenter verbunden ist
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass der Dialog "Speichern als neue Revision..." geschlossen wird.

Ergebnis

Die neue Revision, des TIA Portal-Projekts, wird in Teamcenter erstellt und als neues Objekt unterhalb der letzten Revision angezeigt.

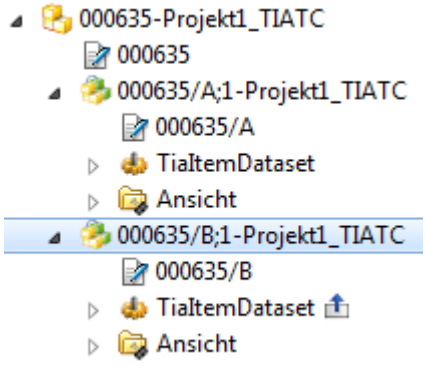

Beim Revisionieren des TIA Portal-Projekts wird das Dataset des aktuell geöffneten TIA Portal-Projekts übernommen.

Das zur neuen Revision zugehörige Dataset des TIA Portal-Projekts ist "Ausgecheckt" und in Teamcenter mit dem folgenden Symbol gekennzeichnet.

Das Dataset der älteren Revision wird eingecheckt, in Teamcenter wird das folgende Symbol entfernt.

Das können Sie in der Ordneransicht von Teamcenter sehen.

Im TIA Portal wird die ältere Revision des TIA Portal-Projekts geschlossen. Die neue Revision des TIA Portal-Projekts wird im TIA Portal im Modus "Schreiben" geöffnet.

Nach dem "Speichern als neue Revision..." stehen Ihnen im TIA Portal folgende Teamcenter-Operationen zum Arbeiten mit TIA Portal-Projekten zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....

- Einchecken.
- Auschecken rückgängig machen.

Siehe auch

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

TIA Portal-Projekt einchecken (Seite 78)

Auschecken eines TIA Portal-Projekts rückgängig machen (Seite 87)

TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69)

Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 83)

Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84)

5.11.5.1 Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)

Im Register "Allgemein" des Dialogs "Speichern als neue Revision" definieren Sie die Eigenschaften eines TIA Portal-Projekts das Sie als neue Revision in Teamcenter speichern wollen.

Siehe auch

Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84)

5.11.5.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)

Im Register "Zusätzliche Eigenschaften" bearbeiten Sie die Werte der hochgeladenen zusätzlichen Eigenschaften von TIA Portal-Projekten das Sie als neue Revision nach Teamcenter speichern wollen.

Im Register "Zusätzliche Eigenschaften" haben Sie die Möglichkeit, Werte von hochgeladenen Elementtypen oder abhängigen Typen einzugeben. Der Inhalt des Registers "Zusätzliche Eigenschaften" ist von den verwendeten Elementtypen, den abhängigen Typen oder den Eigenschaften von geerbten Typen abhängig.

Siehe auch

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84)

Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 85)

5.11.5.3 Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)

Aufbau des Registers "Allgemein" im Dialog "Speichern als neue Revision..."

Die Ansicht des Registers "Allgemein" besteht aus folgenden Feldern und Schaltflächen:

Feld	Beschreibung
Name (nur lesend):	Im Feld "Name" wird der Name der ausgewählten globalen Bibliothek (z.B. t4tia_object_name) lesend angezeigt
Element-ID: (nur lesend)/	Im Feld "Element-ID" wird die "Element-ID" der ausgewählten globalen Bibliothek lesend angezeigt
Revision-ID:	Eingabefeld für eine gültige Teamcenter Revision-ID, die Sie der neuen Revision zuweisen wollen.
Kommentar:	Eingabefeld für einen Kommentar.

Schaltfläche	Beschreibung
Zuweisen	Durch klicken auf diese Schaltfläche weisen Sie dem TIA Portal-Projekt, automatisch eine Revision-ID zu.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Siehe auch

Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 83)

5.11.5.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte)

Aufbau des Registers "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Die im Register "Zusätzliche Eigenschaften" tabellarisch angezeigten Informationen sind davon abhängig:

- welche Elementtypen oder abhängige Typen Sie ausgewählt haben,
- welche Eigenschaften die ausgewählten Elementtypen und abhängigen Typen geerbt haben.

Eigenschaften des Registers "Zusätzliche Eigenschaften"

Im Register "Zusätzliche Eigenschaften" werden die Eigenschaften von Elementtypen und abhängigen Typen im Modus lesend/schreibend angezeigt.

Die Ansicht des Registers "Zusätzliche Eigenschaften" besteht aus folgenden Spalten und Schaltflächen:

Spalte	Bedeutung
Name	Name der zusätzlichen Eigenschaften.
Typ	Art der zusätzlichen Eigenschaften.
Wert	Wert der zusätzlichen Eigenschaften.

Schaltfläche	Bedeutung
Klappliste	Durch klicken auf die Klappliste können Sie Pflichtattribute hinzufügen.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Hinweis zur Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Folgende Regeln gelten für die Bedienbarkeit der Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

- Wenn die erforderlichen Pflichtattribute nicht befüllt sind, ist die Schaltfläche "OK" gegraut und nicht bedienbar.
- Wenn keine Pflichtattribute vorhanden sind, oder wenn alle Pflichtattribute befüllt sind, ist die Schaltfläche "OK" bedienbar.

Siehe auch

Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 85)

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 83)

5.11.5.5 Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)

In der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" werden, die in den Teamcenter Server, hochgeladene Attribute eines TIA Portal-Projekts im Modus lesend/schreibend angezeigt.

Voraussetzungen um die Werte von hochgeladenen Attributen eines TIA Portal-Projekts zu bearbeiten

Um die Werte der hochgeladenen Attribute eines TIA Portal-Projekts zu bearbeiten, wird vorausgesetzt, dass:

- die Attribute des ausgewählten TIA Portal-Projekt Elementtyps hochgeladen sind.
- die Attribute der ausgewählten Element-Revision des TIA Portal-Projekt Elementtyps hochgeladen sind.
- die hochgeladenen Attribute des Element Masterformulars mit dem TIA Portal-Projekt Elementtyp verbunden sind.
- die hochgeladenen Attribute des Element-Revision Masterformulars mit der TIA Portal-Projekt Element-Revision verbunden sind.

Hinweis

Beachten Sie, dass

nur die oben genannten Attribute ab TIA Portal Teamcenter Gateway V14SP1 unterstützt werden. Alle anderen Attribute werden ignoriert.

Bitte beachten Sie:

- wenn das Lesen von hochgeladenen Attributen während des Verbindungsaufbaus mit TIA Portal Teamcenter Gateway fehlgeschlagen ist, im Register “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen
- wenn keine Attribute in den Teamcenter (Server) hochgeladen sind, im Register “Zusätzliche Eigenschaften” die folgende Information angezeigt:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.

Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften"

Beachten Sie Folgendes:

Bei Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften", wird immer nur der Eintrag des Attributs aus dem Register "Allgemein" in Teamcenter gespeichert.

Mehrfachverwendung von Attributen im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Beachten Sie Folgendes:

Bei Mehrfachverwendung desselben Attributs, im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision", wird immer nur der erste Eintrag des Attributs aus der tabellarischen Ansicht in Teamcenter gespeichert.

Verwendung von Sonderzeichen für den Datentyp 'Char'

Nachfolgend werden die Namenskonventionen für den Datentyp 'Char', die Sie beim "Speichern als neue Revision" beachten müssen beschrieben.

Beachten Sie Folgendes:

- Verwenden Sie in den Eingabefeldern des Datentyps 'Char', im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision", nur die ASCII-Werte 32-126.

Siehe auch

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84)

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 83)

5.11.6 Auschecken eines TIA Portal-Projekts rückgängig machen

Auschecken rückgängig machen

Um vorgenommene Änderungen an einem ausgecheckten TIA Portal-Projekt zu verwerfen, haben Sie die Möglichkeit, das Auschecken rückgängig zu machen.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal-Projekt mit Teamcenter verbunden ist.
- das TIA Portal-Projekt im Modus "Schreiben" geöffnet ist.
- keine Multiselektion von mehreren TIA Portal-Projekten unterstützt wird.

Vorgehen

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das entsprechende TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Auschecken rückgängig machen".
   Der Dialog "Auschecken rückgängig machen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "OK", um das Auschecken rückgängig zu machen.
4. Klicken Sie auf die Schaltfläche "Abbrechen", wenn Sie "Auschecken rückgängig machen" abbrechen möchten.

Hinweis

Alle Änderungen, die seit dem letzten Auschecken vorgenommen wurden, werden verworfen.

Ergebnis

Das Auschecken des TIA Portal-Projekts wird rückgängig gemacht. In der Ordneransicht von Teamcenter wird das Symbol entfernt.

Die Bearbeitungssperre des TIA Portal-Projektes ist aufgehoben.

Im TIA Portal-Portal stehen Ihnen nach "Auschecken rückgängig machen" folgende Teamcenter-Operationen für das Arbeiten mit dem TIA Portal-Projekt zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Auschecken.

Siehe auch

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69)

TIA Portal-Projekt auschecken (Seite 76)

5.11.7 TIA Portal-Projekt exportieren

Das Teamcenter Kontextmenü "Speichern" und "Export..." stehen in gegenseitiger Abhängigkeit. Der Zustand des Kontextmenüs "Speichern" und "Export..." ist von folgenden Bedingungen abhängig.

Teamcenter Kontextmenü "Export..."

Das Teamcenter Kontextmenü "Export..." ist aktiviert, wenn

- das ausgewählte TIA Portal-Projekt mit Teamcenter verbunden ist und keine Änderungen vorgenommen wurden.

Vorgehen

Um ein TIA Portal-Projekt zu exportieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das entsprechende TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Export...".

Hinweis

Beachten Sie Folgendes:

Wenn der Befehl "Teamcenter > Export..." für das TIA Portal-Projekt nicht aktiv ist, wurden in dem TIA Portal-Projekt Änderungen vorgenommen. Speichern Sie das TIA Portal-Projekt mit dem Befehl "Teamcenter > Speichern". Danach ist der Befehl "Teamcenter > Export..." aktiv.

Ergebnis

Der Befehl "Teamcenter > Export..." entfernt die Teamcenter-Metainformationen aus dem TIA Portal-Projekt und speichert das geänderte TIA Portal-Projekt in den von Ihnen ausgewählten Ordner.

Hinweis

Beachten Sie Folgendes:

Ein bereits vorhandenes TIA Portal-Projekt kann nicht an denselben Speicherort unter demselben Namen erneut exportiert werden.

Bitte beachten Sie:

1. Das Teamcenter Kontextmenü "Speichern" ist deaktiviert, wenn zwischen dem TIA Portal und Teamcenter keine Verbindung besteht.
2. Das Teamcenter Kontextmenü "Export..." ist deaktiviert, wenn in dem ausgewählten TIA Portal-Projekt Änderungen vorgenommen wurden.

5.11.7.1 Register "Info" im Dialog "Export..." - (TIA Portal Projekte)

Im Register "Info" werden nach dem Export die Informationen zum Speicherort des TIA Portal-Projekts angezeigt.

5.11.8 TIA Portal-Projekt aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern

5.11.8.1 Prüfungen bei Verwendung der Funktion "Speichern zu Element..." - (TIA Portal-Projekte)

Mit der Verwendung der Funktion "Speichern zu Element..." führt Teamcenter folgende Prüfungen durch.

1. Mit der Element-ID / Revision-ID wird geprüft, ob die Element-ID / Revision-ID in Teamcenter vorhanden ist.
   Wenn die Element-ID / Revision-ID nicht in Teamcenter vorhanden ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
2. Mit der Element-ID / Revision-ID wird geprüft, ob der Elementtyp in Teamcenter mit dem "T4TiaProject" Elementtyp und seinem abhängigen Typen übereinstimmt.
   Wenn der Elementtyp in Teamcenter mit dem "T4TiaProject" Elementtyp und seinem abhängigen Typen nicht übereinstimmt, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.

3. Mit der Element-ID / Revision-ID wird geprüft, ob in dem Element und der zugehörigen Revision ein gültiger TIA Dataset vorhanden ist.
   Wenn in dem Element und der zugehörigen Revision kein gültiger TIA Dataset vorhanden ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
4. Mit der Element-ID / Revision-ID wird geprüft, ob der Name des TIA Portal Objekts (Name des TIA Portal-Projekts) mit dem Namen des Datasets übereinstimmt.
   Wenn der Name des TIA Portal-Projekts nicht mit dem Namen des Datasets übereinstimmt, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
5. Mit der Element-ID / Revision-ID wird geprüft, ob das Dataset des Elements und die zugehörige Revision in Teamcenter durch einen anderen Benutzer ausgecheckt ist.
   Wenn das Dataset des Elements und die zugehörige Revision in Teamcenter durch einen anderen Benutzer ausgecheckt ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
6. Mit der Element-ID / Revision-ID wird geprüft, ob sich die Version des TIA Portal-Projekts von der Version von Teamcenter unterscheidet.
   Falls sich die Version des TIA Portal-Projekts von der Version in Teamcenter unterscheidet, können Sie im Dialog "Kritische Meldung" entscheiden, ob Sie die Version in Teamcenter mit Ihren Änderungen überschreiben möchten oder nicht.
   – Um die Version des Projekts in Teamcenter mit der neuen oder geänderten Version zu überschreiben, klicken Sie auf die Schaltfläche "Ja".
   – Um das "Speichern zu Element..." abzubrechen, klicken Sie auf die Schaltfläche "Nein".
7. Mit der Element-ID / Revision-ID wird geprüft:
   – Ob der Benutzer über die nötigen Zugriffsrechte verfügt.
   – Welchen Freigabe Status die Element-ID / Revision-ID hat.
     Wenn der Benutzer nicht über die nötigen Zugriffsrechte verfügt und die Element-ID / Revision-ID den Freigabe Status "Released" hat, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.

5.11.8.2 TIA Portal-Projekt aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern

Mit der Funktion "Speichern zu Element..." verbinden Sie ein TIA Portal-Projekt, das außerhalb des Cache-Servers von Teamcenter bearbeitet wurde, wieder mit Teamcenter.

Bei Änderungen des TIA Portal-Projekts außerhalb des Cache-Servers von Teamcenter, wird die Datei "versionInfo.xml" aktualisiert.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.

Vorgehensweis

Um ein TIA Portal-Projekt wieder mit Teamcenter zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern zu Element...".
   Der Dialog "Speichern zu Element..." wird geöffnet.
3. Im Feld "Name", wird standardmäßig der Name des ausgewählten TIA Portal-Projekts angezeigt.
   Ändern Sie gegebenenfalls den Namen im Feld "Name".
4. Im Feld "Element-ID", wird standardmäßig die zum TIA Portal-Projekt zugehörige Element-ID aus der Datei "versionInfo.xml" angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Element-ID" die vorhandene Element-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Element-ID" leer. Weisen Sie dem Feld "Element-ID" eine bereits gültige oder beliebige Teamcenter Element-ID zu.
     Maximal eingebbar sind 128 Zeichen.
5. Im Feld "Revision-ID" wird standardmäßig die zum TIA Portal-Projekt zugehörige Revision-ID aus der Datei "versionInfo.xml " angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Revision-ID" die vorhandene Revision-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Revision-ID" leer. Weisen Sie dem Feld "Revision-ID" eine bereits gültige oder beliebige Teamcenter Revision-ID zu.
     Maximal eingebbar sind 32 Zeichen.
6. Bestätigen Sie Ihre Eingabe mit "OK".
   Das TIA Portal-Projekt wird unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass die Operation "Speichern zu Element..." abgebrochen wird.

Hinweis

Beachten Sie:

Die Felder Element-ID / Revision-ID sind obligatorische Felder. Die Schaltfläche "OK" ist nur dann editierbar, wenn in den Feldern Element-ID / Revision-ID ein Wert eingetragen ist.

Vorabprüfung für das "Speichern zu Element..."

Bei Verwendung der Funktion "Speichern zu Element..." führt Teamcenter folgende Prüfungen durch:

Siehe hierzu: Prüfungen bei Verwendung der Funktion "Speichern zu Element..."

Beim Speichervorgang wird Folgendes überprüft:

1. Ob die Teamcenter gespeicherte Dataset-Version und die zugehörige Revision-ID durch den aktuell angemeldeten Benutzer oder einen anderen Benutzer ausgecheckt ist.
2. Ob die Dataset-Version durch einen anderen Benutzer ausgecheckt ist.
   Wenn die Dataset-Version durch keinen andern Benutzer ausgecheckt, wird beim Ausführen der Funktion "Speichern zu Element..." die Dataset-Version implizit ausgecheckt.
3. Ob das TIA Portal- Projekt exportiert und mit der Dateikennung "*.zap" archiviert wurde.
   Zu dem archivierten Projekt darf keine Datei "versionInfo.xml" vorhanden sein.
   Falls es während des Export- und Archivierungsvorgangs zu Fehlern kommt, wird eine entsprechende Meldung angezeigt, dass der Export- und Archivierungsvorgang fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
   Um die Meldung zu bestätigen, gehen Sie wie folgt vor:.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung:

   Hinweis

   Abhilfe beim Abbruch des Export- und Archivierungsvorgangs:

   1. Erstellen Sie in Teamcenter eine neue Dataset-Version.
   2. Laden Sie die archivierte Projektdatei in das Dataset hoch.
      Dabei wird das Dataset ausgecheckt.
      Falls das Erstellen der Element-Revision nicht möglich ist, wird eine Fehlermeldung angezeigt, dass das Erstellen der Element-Revision fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
      Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
      – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.
        Es wird keine Kopie des TIA Portal-Projekts im TcCache gespeichert.
        Es wird kein TcCache-Pfad für das TIA Portal-Projekt erstellt.

4. Übertragen Sie die Projektänderungen nach Teamcenter.
   Falls die Übertragung, der Projektänderungen nach Teamcenter nicht möglich ist, wird eine entsprechende Meldung angezeigt.
   Um die Meldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.

Aktionen beim "Speichern zu Element..."

Nach Abschluss aller Prüfungen werden folgende Aktionen ausgeführt:

1. Das TIA Portal-Projekt wird archiviert und unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
2. Das zugehörige Dataset befindet sich im Status "Ausgecheckt" und ist mit dem folgenden Symbol gekennzeichnet.

   Das können Sie in der Ordneransicht von Teamcenter sehen.
3. Nach Abschluss der Aktionen, wird der Dialog "Das Projekt wurde erfolgreich nach Teamcenter gespeichert" angezeigt.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

"Rollback"-Funktionen

1. Falls der Export- und Archivierungsvorgang fehlgeschlagen ist, wird auf die ältere Version der Datei "versionInfo.xml" zugegriffen.
2. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie die exportierten und archivierten Dateien von der lokalen Festplatte.
3. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version) entfernen. Damit wird im TIA Portal-Projekt auf die ursprüngliche Version der Datei "versionInfo.xml" zugegriffen. Sobald das TIA Portal-Projekt mit der Funktion "Speichern zu Element..." in Teamcenter gespeichert wurde, wird ein Dialog mit einer entsprechenden Meldung angezeigt. Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Ergebnis

Nach erneutem Öffnen der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter Kontextmenüeinträge zur Verfügung:

- Speichern als neues Element...
- Speichern als neue Revision...
- Speichern zu Element...
- Einchecken.
- Auschecken rückgängig machen.
- Export...

5.11.8.3 Aufbau Dialog "Speichern zu Element..."

Im Dialog "Speichern zu Element..." definieren Sie die Eigenschaften eines TIA Portal-Projekts das Sie in Teamcenter speichern wollen.

Aufbau des Dialogs "Speichern zu Element..."

Die Ansicht des Dialogs "Speichern zu Element..." besteht aus folgenden Feldern und Schaltflächen.

Feld	Beschreibung
Name:	Eingabefeld für den Namen unter dem Sie das TIA Portal-Projekt in Teamcenter speichern wollen. Standardmäßig wird der Name des ausgewählten TIA Portal-Projekts angezeigt. Das Feld "Name" ist nicht editierbar.
Element-ID / Revision-ID:	Eingabefeld für die Element-ID / Revision-ID die Sie dem TIA Portal-Projekt zuweisen wollen.

Schaltfläche	Bemerkung
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

5.11.9 TIA Portal-Projekt das im Teamcenter-Cache Ordner gespeichert ist wieder in das zugehörige Element speichern

Mit der Funktion "Speichern zu Element..." speichern Sie ein TIA Portal-Projekt aus dem Teamcenter-Cache Ordner heraus wieder in das zugehörige Element.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.

Vorgehensweis

Um ein TIA Portal-Projekt wieder mit Teamcenter zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern zu Element...".
   Der Dialog "Speichern zu Element..." wird geöffnet.
3. Im Feld "Name", wird standardmäßig der Name des ausgewählten TIA Portal-Projekts angezeigt.
   Das Feld "Name" ist nicht editierbar.
4. Weisen Sie im Feld "Element-ID" eine Element-ID zu.
   – Maximal eingebbar sind 128 Zeichen.
5. Geben Sie im Feld "Revision-ID" eine Revision-ID ein.
   – Maximal eingebbar sind 32 Zeichen.

6. Bestätigen Sie Ihre Eingabe mit "OK".
   Das TIA Portal-Projekt wird unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass die Operation "Speichern zu Element..." abgebrochen wird.

Hinweis

Die Felder Element-ID / Revision-ID sind obligatorische Felder. Die Schaltfläche "OK" ist nur dann editierbar, wenn in den Feldern Element-ID / Revision-ID ein Wert angezeigt wird.

Aktionen beim "Speichern zu Element..."

Beim Ausführen der Funktion "Speichern zu Element..." folgende Aktionen ausgeführt:

1. Das TIA Portal-Projekt wird archiviert und unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
2. Das zugehörige Dataset befindet sich im Status "Ausgecheckt" und ist mit dem folgenden Symbol gekennzeichnet.

   Das können Sie in der Ordneransicht von Teamcenter sehen.
3. Nach Abschluss der Aktionen, wird der Dialog "Das TIA Portal-Projekt wurde erfolgreich nach Teamcenter gespeichert" geöffnet.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Ergebnis

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen zum Arbeiten mit TIA Portal-Projekten zur Verfügung:

- Speichern als neues Element...
- Speichern als neue Revision...
- Speichern zu Element...
- Einchecken.
- Auschecken rückgängig machen.
- Export...

5.11.10 Ein mit Teamcenter Gateway verbundenes TIA Portal-Projekt als Referenzprojekt öffnen

Die Option "Als Referenzprojekt öffnen" bietet Ihnen im Dialog "Projekt aus Teamcenter öffnen" die Möglichkeit ein TIA Portal-Projekt als Referenzprojekt zu öffnen. Standardmäßig ist das Optionskästchen "Als Referenzprojekt öffnen" deaktiviert.

Vorgehen

Um ein TIA Portal-Projekt im Dialog "Projekt aus Teamcenter öffnen" als Referenzprojekt zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektansicht im Menü "Projekt" den Befehl "Projekt aus Teamcenter öffnen...".
   Der Dialog "Projekt aus Teamcenter öffnen" wird geöffnet.
   Weitere Informationen zum Dialog "Projekt aus Teamcenter öffnen" erhalten Sie unter: TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen (Seite 152).
2. Wählen Sie ein entsprechendes TIA Portal-Projekt aus dem Suchergebnis aus.
   Weitere Informationen hierzu finden Sie unter: Öffnen von TIA Portal-Projekten aus dem Suchergebnis (Seite 156).
3. Aktivieren Sie das Optionskästchen "Als Referenzprojekt öffnen".
   Das Optionskästchen "Als Referenzprojekt öffnen" ist standardmäßig deaktiviert.

   Hinweis

   Wenn Sie das Optionskästchen "Als Referenzprojekt öffnen" aktivieren, wird das Optionskästchen "Auschecken" deaktiviert.

4. Klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

Beim "Öffnen" des TIA Portal-Projekts als Referenzprojekt werden die folgenden Aktionen ausgelöst:

1. Ein temporärer "RF"-Ordner (Referenzprojekt-Ordner) wird parallel zum Teamcenter Cache-Ordner erstellt.
   Der "RF"-Ordner (Referenzprojekt-Ordner) muss 2 Zeichen lang sein und parallel zum TcCache-Ordner.
   Beispiel:

aktueller Teamcenter Cache-Ordner	<TIA default projectlocation>\TC\<ItemID>\<RevID>\ProjectFolder
"RF"-Ordner (Referenzprojekt-Ordner)	<TIA default projectlocation>\RF\<ItemID>\<RevID>\ProjectFolder

2. Für die Pfadlänge des temporären "RF"-Ordners (Referenzprojekt-Ordner), gelten die Regeln, die auch für die Pfadlänge des Teamcenter Cache-Ordners gelten.
3. Das ausgewählte TIA Portal-Projekt wird in den temporären "RF"-Ordner (Referenzprojekt-Ordner) heruntergeladen, und in der Palette "Referenzprojekte" geöffnet.
   - In der Palette Referenzprojekte wird das TIA Portal-Projekt mit der Namenserweiterung und dem Teamcenter Overlay-Symbol angezeigt.
   - Das TIA Portal-Projekt wird unabhängig davon, ob es durch den aktuellen Benutzer, oder einen andern Benutzer ausgecheckt ist, geöffnet.
4. Wenn Sie eine ältere Version eines TIA Portal-Projekts als Referenzprojekt öffnen, erhalten Sie eine entsprechende Meldung das TIA Portal-Projekt zu aktualisieren.

5. Wenn Sie das TIA Portal-Projekt aktualisieren, wird in der Palette Referenzprojekte, der Name mit der Namenserweiterung und dem Teamcenter Overlay-Symbol angezeigt.
6. Wenn es beim Herunterladen eines TIA Portal-Projekts von Teamcenter zu Fehlern kommt, erhalten Sie eine entsprechende Meldung mit detaillierten Fehlerinformationen.
7. Empfehlung: Vergleichen Sie mit dem Vergleichseditor die benötigten Objekte des TIA Portal-Projekts aus der Projektnavigation mit dem Referenzprojekt aus der Palette Referenzprojekte.
8. Wenn Sie ein TIA Portal-Projekt in der Palette Referenzprojekte schließen, wird der Revisions-Ordner mit dem TIA Portal-Projekt, aus dem temporären Referenzprojektordner gelöscht. Gleichzeitig wird das TIA Portal-Projekt auch aus dem temporären Ordner gelöscht.
9. Beachten Sie, dass das TIA Portal-Projekt nicht aus dem temporären Ordner gelöscht wird, wenn Sie das TIA Portal schließen.

Hinweis

Öffnen Sie immer zuerst die Palette "Referenzprojekte", bevor Sie das TIA Portal-Projekt aus Teamcenter heraus öffnen.

5.12 Arbeitsabläufe zwischen dem TIA Portal Teamcenter Gateway und TIA Portal Multiuser-Engineering - TIA Portal-Projekte

5.12.1 Ein mit Teamcenter verbundenes TIA Portal-Projekt als Multiuser-Serverprojekt speichern

Dieses Kapitel enthält Informationen wie Sie ein TIA Portal-Projekt, das nach Teamcenter gespeichert ist, als neues Multiuser-Serverprojekt speichern.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- zwischen dem TIA Portal und Teamcenter eine aktive Verbindung besteht.
- Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet, installiert haben.
- im TIA Portal unter "Extras" > "Einstellungen" > "Allgemein" > "Teamcenter Gateway" das Optionskästchen "Automatisch einchecken" aktiviert ist.

Vorgehen

Um ein TIA Portal-Projekt, das nach Teamcenter gespeichert ist, als neues Multiuser-Serverprojekt zu speichern, gehen Sie folgendermaßen vor:

1. Öffnen Sie das entsprechende TIA Portal-Projekt.
2. Wählen Sie dazu in der Projektansicht im Menü "Projekt" den Befehl "Projekt aus Teamcenter öffnen...".
   Der Dialog "Projekt aus Teamcenter öffnen" wird angezeigt.
3. Suchen Sie im Dialog "Projekt aus Teamcenter öffnen" nach dem entsprechenden TIA Portal-Projekt, um es zu öffnen.
4. Um das TIA Portal-Projekt dem Multiuser-Server hinzuzufügen, klicken Sie auf den Menübefehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Während das TIA Portal-Projekt dem Multiuser-Server hinzugefügt wird, wird das TIA Portal-Projekt in der Projektnavigation geschlossen. Wenn im TIA Portal, unter "Extras" > "Einstellungen" > "Allgemein" > "Teamcenter Gateway" das Optionskästchen "Automatisch einchecken" aktiviert ist, wird das TIA Portal-Projekt automatisch in Teamcenter eingecheckt. Das automatische Einchecken wird durch eine Meldung in einem Dialog protokolliert.

5.12.1.1 Serverprojekt-Ansicht eines Multiuser-Serverprojekts

Serverprojekt-Ansicht eines Multiuser-Serverprojekts

In der Serverprojekt-Ansicht wird der Name, des dem Multiuser-Server hinzugefügten TIA Portal-Projekts angezeigt. Der angezeigte Name besteht aus:

- Projektname
- Namenserweiterung
- Element-ID / Revision-ID.
- Teamcenter Overlay-Symbol
  Das Teamcenter Overlay-Symbol zeigt an, dass das TIA Portal-Projekt auf dem Multiuser-Server gespeichert wurde und mit Teamcenter verknüpft ist.

Befehle im Kontextmenü der Serverprojekt-Ansicht

In der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter-Operationen für das Arbeiten mit dem Multiuser-Serverprojekt zur Verfügung:

- Speichern.
- Speichern als neue Revision....
- Speichern als neues Element....
- Auschecken (nur gesperrte).
- Einchecken (nur entsperrte).

5.12.1.2 Lokale Session eines spezifischen Multiuser-Serverprojekt erstellen

Dieses Kapitel enthält Informationen wie Sie von einem TIA Portal-Projekt, das nach Teamcenter gespeichert ist und dem Multiuser-Server hinzugefügt wurde, eine lokale Session erstellen.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal-Projekt nach Teamcenter gespeichert ist und dem Multiuser-Server hinzugefügt wurde.

Lokale Session eines spezifischen Multiuser-Serverprojekts erstellen

Um von einem TIA Portal-Projekt, das nach Teamcenter gespeichert ist und dem Multiuser-Server hinzugefügt wurde eine lokale Session zu erstellen, gehen Sie folgendermaßen vor:

- Zum Erstellen einer lokalen Session eines spezifischen Multiuser-Serverprojekts, verwenden Sie den standardmäßigen Multiuser Arbeitsablauf.
  Weitere Informationen finden Sie in der Online-Hilfe von "Multiuser Engineering einsetzen" unter "Lokale Session erstellen".

In der lokalen Session wird der Name des TIA Portal-Projekts angezeigt, bestehend aus:

- Projektname
- Namenserweiterung
- Element-ID / Revision-ID.
- Teamcenter Overlay-Symbol
  Das Teamcenter Overlay-Symbol zeigt an, dass das TIA Portal-Projekt mit Teamcenter verknüpft ist.

Lokale Session speichern

Änderungen, die Sie auf dem Multiuser-Server an einer lokalen Session vorgenommen haben, speichern Sie mit dem standardmäßigen Multiuser Arbeitsablauf.

Weitere Informationen finden Sie in der Online-Hilfe von "Multiuser Engineering einsetzen" unter "Lokale Session erstellen".

Teamcenter spezifische Operationen müssen in der Serverprojekt-Ansicht ausgeführt werden.

5.12.1.3 Beispiel für die Behandlung von Inkonsistenzen zwischen TIA Portal-Projekten in Teamcenter und im Multiuser-Server

Dieses Kapitel enthält Beispiele, wie Sie Inkonsistenzen zwischen TIA Portal-Projekten in Teamcenter und im Multiuser-Server beheben.

Beispiele

Nachfolgend werden Beispiele erläutert, die dazu führen, dass Inkonsistenzen zwischen TIA Portal-Projekten in Teamcenter und im Multiuser-Server auftreten.

Beispiel 1:

- Die Dataset-Version A in Teamcenter wurde mit der letzten Revision des Multiuser-Servers mit dem Befehl "Teamcenter > Speichern" aktualisiert.
- Die Dataset-Version A wurde außerhalb des Multiuser-Servers geändert, dadurch ist in Teamcenter die neue Dataset-Version B entstanden.

Beispiel 2:

- Die Dataset-Version A in Teamcenter wurde mit der letzten Version des Multiuser-Servers mit dem Befehl "Teamcenter > Speichern" aktualisiert.
- Änderungen, die auf dem Multiuser-Server in einer lokalen Session vorgenommen wurden, wurden parallel auf dem Multiuser-Server eingecheckt.

Abhilfe

Aktualisieren Sie das TIA Portal-Projekt im Multiuser-Server durch einen manuellen Vergleich der Inhalte der Dataset-Version B. Um die Dataset-Version B des TIA Portal-Projekts in Teamcenter mit der Dataset-Version B im Multiuser-Server zu vergleichen müssen Sie die Dataset-Version in der Referenzprojekt-Ansicht öffnen.

Siehe auch: AUTOHOTSPOT

5.12.2 Auschecken (nur gesperrte)

Um ein oder mehrere TIA Portal-Projekte die Sie dem Multiuser-Server hinzugefügt haben exklusiv für sich zur Bearbeitung zu reservieren, können Sie die TIA Portal-Projekte in der Multiuser-Server-Ansicht auschecken. Dadurch sind die TIA Portal-Projekte in Teamcenter für andere Benutzer gesperrt.

Voraussetzung

Im folgenden wird vorausgesetzt, dass

- Sie das TIA Portal-Projekt in der Projektansicht im Menü "Projekt" mit dem Befehl "Projekt aus Teamcenter öffnen..." geöffnet haben.
- Sie das TIA Portal-Projekt dem Multiuser-Server hinzugefügt haben.
- Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet, installiert haben.

Vorgehen

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das entsprechende "TIA Portal-Projekt".
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Auschecken (nur gesperrte)".

   Hinweis

   Auschecken (nur gesperrte) bewirkt, dass

   - die Dataset-Version des TIA Portal-Projekts ausgecheckt wird, aber nicht auf den Multiuser-Server heruntergeladen wird.
   - die folgende Meldung "Das Projekt ist bereits durch einen anderen Benutzer ausgecheckt: <Benutzer Name>." angezeigt wird:
     - wenn das TIA Portal-Projekt bereits durch einen anderen Benutzer ausgecheckt ist,
     - und der Status des TIA Portal-Projekts in der Multiuser-Server Ansicht nicht verändert werden soll.
   - geprüft wird, ob die Dataset-Version bereits vom aktuellen Benutzer ausgecheckt ist, wenn ja, dann erscheint die Meldung "Das Projekt wurde bereits vom aktuellen Benutzer ausgecheckt".

   Hinweis

   Auschecken (nur gesperrte) prüft, ob die aktuelle Dataset-Version des TIA Portal-Projekts im Multiuser-Server von Teamcenter Server abweicht.

   - Wenn die aktuelle Dataset-Version des TIA Portal-Projekts im Multiuser-Server vom Teamcenter Server abweicht, wird der folgende Meldung angezeigt:
     "Die Dataset-Version des TIA Portal-Projekts im Multiuser-Server unterscheidet sich vom Teamcenter-Server. Möchten Sie mit dem Auschecken der Dataset-Version fortfahren?"
     - Klicken auf die Schaltfläche "Ja" bewirkt, dass die Dataset-Version ausgecheckt wird.
     - Klicken auf die Schaltfläche "Nein" bewirkt, dass die Operation "Auschecken (nur gesperrte) abgebrochen wird.
   - Wenn die aktuelle Dataset-Version des TIA Portal-Projekts im Multiuser-Server nicht von der im Teamcenter Server abweicht, wird die Dataset-Version des TIA Portal-Projekts ausgecheckt und für die Bearbeitung durch andere Benutzer gesperrt.

   Die Meldung "Das Projekt wurde erfolgreich aus Teamcenter ausgecheckt" wird angezeigt.
3. Klicken Sie auf die Schaltfläche "OK", um das Auschecken zu bestätigen.

Ergebnis

Die Dataset-Version des TIA Portal-Projekts ist ausgecheckt.

Die Dataset-Version ist für die Bearbeitung durch andere Benutzer gesperrt. Dadurch wird verhindert, dass während der Bearbeitung des TIA Portal-Projekts auf dem Multiuser-Server Änderungen in die Dataset-Version einfließen.

5.12.3 Einchecken (nur entsperrte)

Um ein TIA Portal-Projekt, das Sie dem Multiuser-Server zur Bearbeitung hinzugefügt haben, frei zu geben, müssen Sie das TIA Portal-Projekt in der Multiuser-Server-Ansicht einchecken. Dadurch wird das TIA Portal-Projekt für andere Benutzer in Teamcenter frei gegeben.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- Das TIA Portal-Projekt bereits in Teamcenter gespeichert ist und dem Multiuser-Server hinzugefügt wurde.
- Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet, installiert haben.
- Das Dataset ausgecheckt ist.

Vorgehen

Um die Bearbeitungssperre des TIA Portal-Projekts, das Sie dem Multiuser-Server hinzugefügt haben, aufzuheben, gehen Sie folgendermaßen vor:

- Selektieren Sie in der Serverprojekt-Ansicht das entsprechende "TIA Portal-Projekt".
- Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Einchecken (nur entsperrte)".

Hinweis

Verwendung des Menübefehls "Einchecken (nur entsperrte)":

- Wenn bei Verwendung des Menübefehls "Einchecken (nur entsperrte)" die Dataset-Version nicht ausgecheckt ist, wird die folgende Warnung angezeigt:
  - "Das Projekt ist nicht durch den aktuellen Benutzer ausgecheckt.".

Hinweis

Verwendung des Menübefehls "Einchecken (nur entsperrte)":

- Bei Verwendung des Menübefehls "Einchecken (nur entsperrte)" werden keine Daten vom Multiuser-Server an Teamcenter übertragen.
- Bei Verwendung des Menübefehls "Einchecken (nur entsperrte)" erfolgt keine Konsistenzprüfung zwischen der Dataset-Version-ID des Multiuser-Serverprojekts und der Dataset-Version-ID in Teamcenter.

5.12.4 Änderungen an einem Multiuser-Serverprojekt nach Teamcenter speichern

Dieses Kapitel beschreibt, wie Sie Änderungen an einem TIA Portal-Projekt (Multiuser-Serverprojekt), das bereits in Teamcenter gespeichert ist und dem Multiuser-Server hinzugefügt wurde, nach Teamcenter zurückspeichern.

Verwenden Sie in der Serverprojekt-Ansicht im Kontextmenü "Teamcenter" den Befehl "Speichern" um das Multiuser-Serverprojekt als neue Version (Dataset-Version) nach Teamcenter zurück zu speichern.

In der Serverprojekt-Ansicht ist im Kontextmenü "Teamcenter" der Befehl "Speichern" unter den nachfolgend genannten Voraussetzungen aktiviert.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.
- Die Revision-ID des in der Datei "versionInfo.xml" gespeicherten Multiuser-Serverprojekts und die aktuelle Revision-ID des Multiuser-Serverprojekts sind unterschiedlich. Damit ist gemeint, dass das TIA Portal-Projekt in Teamcenter und das TIA Portal-Projekt im Multiuser-Server unterschiedlich sind.
  Wenn die Revision-IDs unterschiedlich sind, ist das Speichern über das Kontextmenü möglich.

Ursachen und Anzeige für unterschiedliche Revision-IDs

Folgende Ursachen für unterschiedliche Revision-IDs des TIA Portal-Projekts in Teamcenter und des TIA Portal-Projekts im Multiuser-Serverprojekt sind möglich:

- Die Revision-ID des TIA Portal-Projekts im Multiuser-Serverprojekt wurde geändert, z.B. durch Einchecken Ihrer "lokalen Session" oder einer anderen "lokalen Session".
- Änderungen an einem TIA Portal-Projekt wurden in der Serverprojekt-Ansicht mit der Option "Änderungen Speichern" gespeichert.

Hinweis

Beachten Sie bitte, dass jedes Multiuser-Serverprojekt seine eigene Revision-ID hat, die Bestandteil der Datei "versionInfo.xml" ist.

Falls ein Multiuser-Serverprojekt bearbeitet wird, und die Änderungen nicht gespeichert werden, wird das Objekt in der Serverprojekt-Ansicht durch ein Fähnchen markiert und im Kontextmenü Teamcenter wird der Befehl "Speichern" aktiviert.

Vorabprüfung für das "Speichern"

Beim Speichervorgang wird Folgendes überprüft:

1. Ob die in Teamcenter gespeicherte Dataset-Version und die zugehörige Revision durch den aktuell angemeldeten Benutzer ausgecheckt ist.
   Falls die Dataset-Version und die zugehörige Revision nicht durch den aktuell angemeldeten Benutzer ausgecheckt ist, erscheint eine Meldung und die Operation "Speichern" wird nicht ausgeführt.
2. Ob sich die Dataset-Version ID des Multiuser-Serverprojekts von der Dataset-Version ID in Teamcenter unterscheidet.
   Falls sich die Dataset-Version ID des Multiuser-Serverprojekts von der Dataset-Version ID in Teamcenter unterscheidet, können Sie im Dialog "Kritische Meldung" entscheiden, ob Sie die Dataset-Version ID in Teamcenter mit Ihren Änderungen überschreiben möchten oder nicht.
   Im Dialog "Kritische Meldung" wird folgende Meldung angezeigt: "Die Dataset-Version ID des Projekts in Teamcenter unterscheidet sich von der Dataset-Version ID des Projekts, die Sie in Teamcenter speichern möchten. Möchten Sie die Dataset-Version ID in Teamcenter mit Ihren Änderungen überschreiben?".
   Um die Dataset-Version ID in Teamcenter mit der Dataset-Version ID aus der Serverprojekt-Ansicht zu überschreiben, klicken Sie auf die Schaltfläche "Ja".
   Um das "Speichern" abzubrechen, klicken Sie auf die Schaltfläche "Nein".

Weitere Informationen finden Sie unter Beispiele für den Datenabgleich zwischen den verschiedenen Dataset Versionen im Teamcenter Cache und im Multiuser-Serverprojekts (Seite 107).

Aktionen beim Speichern

Nach Abschluss aller Vorabprüfungen werden die folgenden Aktionen durchgeführt:

1. Die Datei "versionInfo.xml" wird aktualisiert und als Teil des Multiuser-Serverprojekts gespeichert.
   Während des Speichervorgangs wird ein Ladesymbol angezeigt.
   Durch das automatische Einchecken werden Sie aufgefordert, das Dataset einzuchecken.
   Sie können die Aufforderung das Dataset einzuchecken auch ablehnen.
   Nach dem Speichern des Multiuser-Server Projekts in Teamcenter wird die Serverprojekt-Ansicht geschlossen.
   Das neu erstellte Element des Multiuser-Server Projekts wird in einer lokalen Session geöffnet.
2. Das Multiuser-Server Projekt wird exportiert und mit der Dateikennung "*.zap" archiviert.
   Bei dem exportierten und archivierten Projekt handelt es sich um ein normales TIA Portal-Projekt. Zu dem archivierten Projekt darf keine Datei "versionInfo.xml" vorhanden sein.
   Falls es während des Export- und Archivierungsvorgangs zu Fehlern kommt, wird eine Fehlermeldung angezeigt, dass der Export- und Archivierungsvorgang fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
   Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.

   Hinweis

   Abhilfe beim Abbruch des Export- und Archivierungsvorgangs:

   1. Erstellen Sie in Teamcenter eine neue Dataset-Version.
   2. Laden Sie die archivierte Projektdatei in das Dataset hoch.
      Dabei wird das Dataset ausgecheckt.
      Falls das Erstellen der Element-Revision nicht möglich ist, wird eine Fehlermeldung angezeigt, dass das Erstellen der Element-Revision fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
      Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
      – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.
        Es wird keine Kopie des TIA Portal-Projekts im TcCache gespeichert.
        Es wird kein TcCache-Pfad für das TIA Portal-Projekt erstellt.
3. Übertragen Sie die Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server.
   Falls die Übertragung, der Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server nicht möglich ist, wird eine Fehlermeldung angezeigt.
   Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.

Folgende "Rollback"-Funktionen stehen Ihnen zur Verfügung, falls eine der zuvor genannten Aktionen durch eine Fehlermeldung abgebrochen wurde:

1. Falls der Export- und Archivierungsvorgang des Multiuser-Server Projekts fehlgeschlagen ist, wird im Multiuser-Server Projekt auf die ältere Version der Datei "versionInfo.xml" zugegriffen.
2. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie die exportierten und archivierten Dateien von der lokalen Festplatte.
3. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version) entfernen. Damit wird im Multiuser-Server Projekt zur ursprünglichen Version der Datei "versionInfo.xml" zugegriffen.

Empfehlung

Führen Sie im Multiuser-Server eine Aktualisierung der lokalen Session durch, um die Objekte der lokalen Session mit den aktuellen Daten des Multiuser-Server zu aktualisieren.

In der lokalen Session wird die Namenserweiterung angezeigt.

Kontextmenüeinträge

Nach erneutem Öffnen der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter Kontextmenüeinträge zur Verfügung:

- Speichern
- Speichern als neues Element
- Speichern als neue Revision
- Auschecken (nur gesperrte)
- Einchecken (nur entsperrte)
- Export...

5.12.4.1 Beispiele für den Datenabgleich zwischen den verschiedenen Dataset Versionen im Teamcenter Cache und im Multiuser-Serverprojekts

Beispiele für den Datenabgleich

Die folgenden Beispiele zeigen Dataset-Versionen die in Teamcenter, im Teamcenter Cache und im Multiuser-Server unterschiedlich sind.

Beispiel 1

Beispiel 1, zeigt den Status der Dataset-Version des TIA Portal-Projekts (Projekt 1) das vom Teamcenter Server heruntergeladen und dem Multiuser-Server hinzugefügt wurde.

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	1
TcCache	Projekt1	1
MUServer	Projekt1	1

Kontextmenü "Teamcenter" Befehl "Speichern" im Multiuser-Server (Datenabgleich)

Zum Speichern des Multiuser-Serverprojekts (Projekt 1) verwenden Sie im Kontextmenü "Teamcenter" den Befehl "Speichern".

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	2
TcCache	Projekt1	1
MUServer	Projekt1	2

Hinweis

Um die Dataset-Version im TcCache zu aktualisieren, öffnen Sie das TIA Portal-Projekt über den Dialog "Öffnen von TIA Portal-Projekten aus der Suchergebnisliste".

Beispiel 2

Beispiel 2, zeigt den Status der Dataset-Version des TIA Portal-Projekts (Projekt 1), das dem Multiuser-Server hinzugefügt wurde, und über die Projektnavigation geändert wurde. Um Änderungen, die Sie in der Projektnavigation durchgeführt haben zu speichern, wählen Sie auf dem TIA Portal-Projekt, im Kontextmenü "Teamcenter" den Befehl "Speichern". Der Befehl "Speichern" bewirkt folgendes:

- die Dataset-Version in Teamcenter und im TcCache wird aktualisiert.

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	2
TcCache	Projekt1	2
MUServer	Projekt1	1

Kontextmenü "Teamcenter" Befehl "Speichern" im Multiuser-Server (Datenabgleich)

Mit der Verwendung des Kontextmenüs "Teamcenter" mit dem Befehl "Speichern" können Sie die unterschiedlichen Dataset-Versionen zwischen dem Teamcenter-Server und dem Multiuser-Server abgleichen.

Gehen Sie dazu folgendermaßen vor:

- Speichern Sie das TIA Portal-Projekt in Teamcenter. Dabei wird eine neue Dataset-Version erstellt.
  oder
- Brechen Sie das "Speichern" ab.
- Führen Sie einen manuellen Datenabgleich durch.
- Wiederholen Sie den Speichervorgang.
  Die Meldung, dass die Dataset-Versionen unterschiedlich sind, wird erneut angezeigt.
  Wenn Sie mit dem "Speichern" fortfahren, wird eine neue Dataset-Version erstellt.

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	3
TcCache	Projekt1	2
MUServer	Projekt1	3

Beispiel 3

Wenn ein TIA Portal-Projekt das zum Multiuser-Server hinzugefügt wurde geändert wird (z.B. durch einen anderen Benutzer) und anschließend in der Projektnavigation im Kontextmenü "Teamcenter" mit dem Befehl "Speichern" gespeichert wird, werden die Dataset-Versionen in Teamcenter und im TcCache aktualisiert.

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	3
TcCache	Projekt1	2
MUServer	Projekt1	1

Kontextmenü "Teamcenter" Befehl "Speichern" im Multiuser-Server (Datenabgleich)

Mit der Verwendung des Kontextmenüs "Teamcenter" mit dem Befehl "Speichern" können Sie die unterschiedlichen Dataset-Versionen zwischen dem Teamcenter-Server und dem Multiuser-Server abgleichen.

Gehen Sie dazu folgendermaßen vor:

- Speichern Sie das TIA Portal-Projekt in Teamcenter. Dabei wird eine neue Dataset-Version erstellt.
  oder
- Brechen Sie das "Speichern" ab.
- Führen Sie einen manuellen Datenabgleich durch.
- Wiederholen Sie den Speichervorgang.
  Die Meldung, dass die Dataset-Versionen unterschiedlich sind, wird erneut angezeigt.
  Wenn Sie mit dem "Speichern" fortfahren, wird eine neue Dataset-Version erstellt.

Ablageort des Projekts	Projekt Name	Dataset Version
TcServer	Projekt1	4
TcCache	Projekt1	2
MUServer	Projekt1	4

Beispiel 4

Beispiel 4 zeigt den Status einer Dataset-Version eines TIA Portal-Projekts, das im lokalen Cache geändert wurde, nachdem es zum Multiuser-Server hinzugefügt wurde und die TcCache-Änderungen nicht in Teamcenter gespeichert wurden.

Ablageort des Projekts	Projekt Name	Dataset Version	beschädigt?
TcServer	Projekt1	1	-
TcCache	Projekt1	1	Ja
MUServer	Projekt1	1	-

Hinweis

Beachten Sie, dass die Übertragung der Projektänderungen vom lokalen Cache zum Multiuser-Server nicht möglich ist.

Beachten Sie, dass es nicht erlaubt ist, dem Multiuser-Server ein Projekt mit demselben Namen hinzuzufügen.

Für weitere Unterstützung wenden Sie sich an Ihren Multiuser Administrator.

5.12.5 Multiuser-Serverprojekt als neues Element in Teamcenter speichern

Dieses Kapitel beschreibt, wie Sie ein Multiuser-Serverprojekt auf der Serverprojekt-Ansicht als neues Element in Teamcenter speichern. Dazu muss in der Serverprojekt-Ansicht das Kontextmenü "Teamcenter" aktiv sein. Das Kontextmenü "Teamcenter > Speichern als neues Element..." ist in der Serverprojekt-Ansicht unter folgenden Voraussetzungen aktiviert.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass:

- Zwischen dem TIA Portal und Teamcenter eine aktive Verbindung besteht.

Hinweis

Bitte beachten Sie, dass

das Kontextmenü "Teamcenter > Speichern als neues Element..." für folgende Multiuser-Serverprojekte aktiv ist:

- Multiuser-Serverprojekte die in Teamcenter gespeichert sind.
- Multiuser-Serverprojekte die noch nicht in Teamcenter gespeichert sind.

Vorgehen

Um ein Multiuser-Serverprojekt in der Serverprojekt-Ansicht als neues Element in Teamcenter zu speichern, gehen Sie Folgendermaßen vor:

1. Selektieren Sie in der Serverprojekt-Ansicht das entsprechende "TIA Portal-Projekt".
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neues Element...".
   - Der Dialog "Eigenschaften neues Element" wird geöffnet.
   - Das Register "Allgemein" wird standardmäßig angezeigt.
   - Im Feld "Name" wird der Name des ausgewählten TIA Portal-Projekts nur lesend angezeigt und kann nicht geändert werden.
     Maximal eingebbar sind 94 Zeichen.
3. Wählen Sie im Register "Allgemein", im Feld "Elementtyp", einen Elementtyp, für das TIA Portal-Projekt, das Sie als neues Element in Teamcenter speichern wollen aus.
   - Wählen Sie im Feld "Elementtyp" einen abhängigen Typen aus.
     Standardmäßig ist der Typ "TIA Project" ausgewählt.
     Die Klappliste ist nur editierbar, wenn abhängige Typen verfügbar sind.
4. Weisen Sie im Register "Allgemein" dem Feld "Element-ID / Revision-ID" eine bereits gültige oder beliebige Teamcenter Element-ID / Revision-ID zu.
   Wenn Sie das Feld "Element-ID / Revision-ID" 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID zugewiesen.

5. Durch klicken auf die Schaltfläche “Zuweisen” wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID dem ausgewählten Projekt zugewiesen.
   Ergebnis:
   – Die zugewiesene Element-ID / Revision-ID wird im jeweiligen Textfeld angezeigt.
   – Die Textfelder sind nicht editierbar und die Schaltfläche "Zuweisen" ist deaktiviert.

Hinweis

Anpassung der Elementtypen

Die Anpassung, der Elementtypen, hat auf die Felder "Element-ID / Revision-ID" und die Schaltfläche "Zuweisen" folgende Auswirkungen:

1. Die den Feldern "Element-ID / Revision-ID" zugewiesenen IDs werden zurückgesetzt.
2. Die Schaltfläche "Zuweisen" wird auf bedienbar gesetzt.
3. Wenn Sie beabsichtigen, den Namen des TIA Portal-Projekts zu ändern, müssen Sie das TIA Portal-Projekt über die Projektansicht mit dem Kontextmenübefehl "Teamcenter > Speichern als neues Element..." speichern. Siehe auch: TIA Portal-Projekt als neues Element in Teamcenter speichern (Seite 69).

Zeichenlänge des TIA Portal Teamcenter Gateway Cache Pfads:

Hinweis

Beachten Sie, dass in Register "Allgemeine" die Zeichenlänge des TIA Portal Teamcenter Gateway Cache Pfads, bestehend aus dem Namen des Projekts und der Element-ID / Revision-ID geprüft wird. Die Zeichenläge von 143 Zeichen darf hier nicht überschritten werden.

Weitere Informationen zum Register "Allgemein" finden Sie unter Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (TIA Portal Projekte) (Seite 73).

Weitere Informationen zum Register "Zusätzliche Informationen" finden Sie unter Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84).

---

6. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Projekt", das Teamcenter Projekt aus, dem Sie das TIA Portal-Projekt zuweisen wollen.
   – Klicken Sie auf die Schaltfläche "Durchsuchen" (...)..
     Der Dialog "Projekte auswählen" wird geöffnet.
   – Wählen Sie die entsprechenden Teamcenter Projekte aus und bestätigen Sie jede Auswahl mit klicken auf die Schaltfläche "OK".
     Klicken auf die Schaltfläche "OK" bewirkt, dass die ausgewählten Teamcenter Projekte getrennt durch Komma im Feld "Teamcenter Projekt" angezeigt werden.
     Wenn Sie das Feld "Teamcenter Projekt" 'leer' lassen, wird das TIA Portal-Projekt keinem Teamcenter Projekt zugewiesen.

7. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Ordner" den Ordner aus, dem das neu erstelle Element zugewiesen werden soll.
   - Klicken Sie auf die Schaltfläche "Durchsuchen" (...)..
     Der Dialog "Teamcenter Ordner auswählen" wird geöffnet.
     Die Vorbelegung für diesen Ordner ist *"Home*".
     Alle neu erstellen Elemente werden standardmäßig in Ihrem persönlichen Verzeichnis *"Home"* abgelegt.
   - Wählen Sie den Ordner "*Home*" aus, bestätigen Sie die Auswahl durch klicken auf die Schaltfläche "OK".
     Klicken auf die Schaltfläche "OK" bewirkt, dass der ausgewählte Ordner im Textfeld angezeigt wird.
     Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.
8. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
   Maximal eingebbar sind 240 Zeichen.
9. Bestätigen Sie Ihre Eingabe mit "OK".
   Das Multiuser-Serverprojekt wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert:
   Während das Multiuser-Serverprojekt in Teamcenter gespeichert wird, werden die folgenden Aktionen ausgelöst.
   - Die Datei "versionInfo.xml" wird erstellt und als Bestandteil des Multiuser-Server Projekts gespeichert.
   - Das Multiuser-Server Projekt wurde exportiert und mit der Dateikennung "*.zap" archiviert. Bei dem exportierten und archivierten Projekt handelt es sich um ein normales TIA Portal-Projekt. Zu dem archivierten Projekt darf keine Datei "versionInfo.xml" vorhanden sein.
     Wenn es während des Export- und Archivierungsvorgang zu Fehlern kommt, wird eine Meldung angezeigt, dass der Export- und Archivierungsvorgang fehlgeschlagen ist, und dass der Speichervorgang abgebrochen wurde.
     Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
     - Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung

   Sobald das Multiuser-Serverprojekt in Teamcenter gespeichert wurde, wird der Dialog "Das Projekt wurde erfolgreich in Teamcenter gespeichert" geöffnet.
10. Erstellen Sie in Teamcenter ein neues Element mit Ihren spezifischen Eigenschaften und laden Sie die archivierte Projektdatei in den Dataset hoch. Dabei darf das Dataset **nicht** ausgecheckt.
    Falls das Erstellen des neuen Elements nicht möglich ist, wird eine Meldung angezeigt, dass das Erstellen des neuen Elements fehlgeschlagen ist, und dass der Speichervorgang abgebrochen wird.
    Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
    - Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung.

11. Wenn Sie beabsichtigen nicht in der Multiuser-Server Umgebung zur arbeiten, muss das TIA Portal-Projekt in der Projektnavigation im Menü "Projekt" mit dem Befehl "Projekt aus Teamcenter öffnen..." geöffnet werden.

    Hinweis

    Bitte beachten Sie, dass:

    - keine Kopie des TIA Portal-Projekts im TcCache gespeichert wird.
    - kein TcCache-Pfad für das TIA Portal-Projekt erstellt wird.

12. Übertragen Sie die Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server.
    Falls das Übertragen von Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server nicht möglich ist, wird eine Fehlermeldung angezeigt, dass die Übertragung der Projektänderungen an den Multiuser-Server fehlgeschlagen ist, und der Speichervorgang gestoppt wurde.
    Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:

    – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung.

"Rollback"-Funktionen

Folgende "Rollback"-Funktionen stehen Ihnen zur Verfügung, wenn eine der zuvor genannten Aktionen durch eine Fehlermeldung abgebrochen wurde:

1. Wenn der Export- und Archivierungsvorgang des Multiuser-Server Projekts fehlgeschlagen ist, wird im Multiuser-Server Projekt auf einen älteren Stand der Datei "versionInfo.xml" zugegriffen.
2. Wenn das Erstellen eines neuen Elements fehlgeschlagen ist, müssen die exportierten- und archivierten Dateien von der lokalen Festplatte entfernt werden.
3. Wenn das Erstellen eines neuen Elements fehlgeschlagen ist, müssen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version) entfernen, um im Multiuser-Server Projekt zur ursprünglichen Version der Datei "versionInfo.xml" zurückzukehren.

   Hinweis

   Bitte beachten Sie:

   Wenn Sie in der Serverprojekt-Ansicht den Befehl "Teamcenter > Speichern als neues Element..." für ein Multiuser-Server Projekt das noch nicht in Teamcenter gespeichert ist verwenden, muss die Datei "versionInfo.xml" aus dem Multiuser-Server Projekt entfernt werden.

   Sobald das TIA Portal-Projekt in Teamcenter gespeichert wurde, wird der Dialog "Das Projekt wurde erfolgreich in Teamcenter gespeichert" geöffnet.

   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Beachten Sie

1. Während des Speichervorgangs des neuen Elements wird ein Ladesymbol angezeigt.
2. Wenn Sie in der Serverprojekt-Ansicht, mit dem Befehl "Teamcenter > Speichern als neues Element...", von einem Multiuser-Server Projekt das in Teamcenter gespeichert ist, ein neues Element erstellen, werden Sie dazu aufgefordert, das Dataset einzuchecken. Sie können die Aufforderung das Dataset einzuchecken auch ablehnen.
3. Nach dem Speichern des Multiuser-Server Projekts in Teamcenter wird die Serverprojekt-Ansicht geschlossen.
4. Das neu erstellte Element des Multiuser-Server Projekts wird in einer lokalen Session geöffnet.

Empfehlung

Führen Sie in der lokalen Session eine Aktualisierung durch, um die Objekte der lokalen Session mit den aktuellen Daten vom Multiuser-Server zu aktualisieren.

In der lokalen Session wird die Dateikennung und die neue Revision-ID angezeigt. Das Teamcenter Overlay-Symbol zeigt an, dass das neu erstellte Element des Multiuser-Serverprojekts auf dem Multiuser-Server gespeichert wurde und mit Teamcenter verknüpft ist.

Hinweis

Nach erneutem Öffnen der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter Kontextmenüeinträge zur Verfügung:

- Speichern
- Speichern als neues Element...
- Speichern als neue Revision...
- Auschecken (nur gesperrte)
- Einchecken (nur entsperrte)

5.12.6 Multiuser-Serverprojekt als neue Revision in Teamcenter speichern

Dieses Kapitel beschreibt, wie Sie ein Multiuser-Serverprojekt auf der Serverprojekt-Ansicht als neue Revision in Teamcenter speichern. Dazu muss in der Serverprojekt-Ansicht das Kontextmenü "Teamcenter" aktiv sein. Das Kontextmenü "Teamcenter > Speichern als neue Revision..." ist in der Serverprojekt-Ansicht unter folgenden Voraussetzungen aktiviert.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal-Projekt mit Teamcenter verbunden ist.
- das Multiuser-Serverprojekt in Teamcenter gespeichert ist.

Vorgehen

Um ein Multiuser-Serverprojekt in der Serverprojekt-Ansicht als neue Revision in Teamcenter zu speichern, gehen Sie Folgendermaßen vor:

1. Selektieren Sie in der Serverprojekt-Ansicht das entsprechende "TIA Portal-Projekt".
2. Wählen Sie auf dem Multiuser-Serverprojekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neue Revision...".
   Bei dieser Funktion wird das aktuell geöffnete Multiuser-Serverprojekt als Basis für das Erstellen der neuen Revision verwendet.
   - Der Dialog "Speichern als neue Revision" wird geöffnet.
   - Das Register "Allgemein" wird standardmäßig angezeigt.
   - Im Feld "Name" wird der Name des ausgewählten TIA Portal-Projekts, (z.B. t4tia_object_name) nur lesend angezeigt.
   - Im Feld "Element-ID" wird die "Element-ID" des ausgewählten TIA Portal-Projekts nur lesend angezeigt.
3. Optional: Weisen Sie im Register "Allgemein" im Feld "Revision-ID" dem TIA Portal-Projekt eine gültige Teamcenter Revision-ID zu.
   Maximal eingebbar sind 32 Zeichen.
4. Durch klicken auf die Schaltfläche “Zuweisen” wird durch Teamcenter automatisch die nächstmögliche Revision-ID dem TIA Portal-Projekt zugewiesen.
   Ergebnis:
   - Die zugewiesene Revision-ID wird im Feld "Revision-ID" angezeigt.
   - Das Feld "Revision-ID" ist nicht mehr editierbar.
   - Die Schaltfläche "Zuweisen" ist deaktiviert.

   Wenn Sie das Feld 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Revision-ID vergeben.
   Zeichenlänge des TIA Portal Teamcenter Gateway Cache Pfads:

   Hinweis

   Beachten Sie, dass im Register "Allgemeine" die Zeichenlänge des TIA Portal Teamcenter Gateway Cache Pfads, bestehend aus dem Namen des Projekts und der Element-ID / Revision-ID geprüft wird. Die Zeichenläge von 143 Zeichen darf hier nicht überschritten werden.

   Weitere Informationen zum Register "Allgemein" finden Sie unter Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84).

   Weitere Informationen zum Register "Zusätzliche Informationen" finden Sie unter Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (TIA Portal Projekte) (Seite 84).

5. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
   Maximal eingebbar sind 240 Zeichen.
   Standardmäßig werden die Kommentare der Vorgängerversion übernommen. Sie können die Kommentare aus der Vorgängerversion bearbeiten oder unverändert beibehalten

6. Bestätigen Sie Ihre Eingabe mit "OK".
   Das Multiuser-Serverprojekt wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert.
   Während das Multiuser-Serverprojekt in Teamcenter gespeichert wird, werden die folgenden Aktionen ausgelöst:
   – Die Datei "versionInfo.xml" wird erstellt und als Bestandteil des Multiuser-Server Projekts gespeichert.
   – Das Multiuser-Server Projekt wurde exportiert und mit der Dateikennung "*.zap" archiviert. Bei dem exportierten und archivierten Projekt handelt es sich um ein normales TIA Portal-Projekt. Zu dem archivierten Projekt darf keine Datei "versionInfo.xml" vorhanden sein.
     Wenn es während des Export- und Archivierungsvorgang zu Fehlern kommt, wird eine Meldung angezeigt, dass der Export- und Archivierungsvorgang fehlgeschlagen ist, und dass der Speichervorgang abgebrochen wurde.
     Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
     - Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung

   Sobald das Multiuser-Serverprojekt in Teamcenter gespeichert wurde, wird der Dialog "Das TIA Portal-Projekt wurde erfolgreich in Teamcenter gespeichert" geöffnet.
7. Erstellen Sie in Teamcenter eine neue Revision mit Ihren spezifischen Eigenschaften und laden Sie die archivierte Projektdatei in den Dataset hoch. Dabei darf das Dataset **nicht** ausgecheckt.
   Falls das Erstellen der neuen Revision nicht möglich ist, wird eine Meldung angezeigt, dass das Erstellen der neuen Revision fehlgeschlagen ist, und dass der Speichervorgang abgebrochen wird.
   Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung.
8. Wenn Sie beabsichtigen nicht in der Multiuser-Server Umgebung zur arbeiten, muss das TIA Portal-Projekt in der Projektnavigation im Menü "Projekt" mit dem Befehl "Projekt aus Teamcenter öffnen..." geöffnet werden.

   Hinweis

   Bitte beachten Sie, dass:
   - keine Kopie des TIA Portal-Projekts im TcCache gespeichert wird.
   - kein TcCache-Pfad für das TIA Portal-Projekt erstellt wird.

9. Übertragen Sie die Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server.
   Falls das Übertragen von Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server nicht möglich ist, wird eine Fehlermeldung angezeigt, dass die Übertragung der Projektänderungen an den Multiuser-Server fehlgeschlagen ist, und der Speichervorgang gestoppt wurde.
   Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Fehlermeldung.

Folgende "Rollback"-Funktionen stehen Ihnen zur Verfügung, wenn eine der zuvor genannten Aktionen durch eine Fehlermeldung abgebrochen wurde

1. Wenn der Export- und Archivierungsvorgang des Multiuser-Server Projekts fehlgeschlagen ist, wird im Multiuser-Server Projekt auf einen älteren Stand der Datei "versionInfo.xml" zugegriffen.
2. Wenn das Erstellen einer neuen Revision fehlgeschlagen ist, müssen die exportierten- und archivierten Dateien von der lokalen Festplatte entfernt werden.
3. Wenn das Erstellen einer neuen Revision fehlgeschlagen ist, müssen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version) entfernen, um im Multiuser-Server Projekt zur ursprünglichen Version der Datei "versionInfo.xml" zurückzukehren.

Hinweis

Bitte beachten Sie:

Wenn Sie in der Serverprojekt-Ansicht den Befehl "Teamcenter > Speichern als neue Revision..." für ein Multiuser-Server Projekt das noch nicht in Teamcenter gespeichert ist verwenden, muss die Datei "versionInfo.xml" aus dem Multiuser-Server Projekt entfernt werden.

Sobald das TIA Portal-Projekt in Teamcenter gespeichert wurde, wird der Dialog "Das Projekt wurde erfolgreich in Teamcenter gespeichert" geöffnet.

– Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Beachten Sie:

1. Während des Speichervorgangs der neuen Revision wird ein Ladesymbol angezeigt.
2. Wenn Sie in der Serverprojekt-Ansicht, mit dem Befehl "Teamcenter > Speichern als neue Revision...", von einem Multiuser-Server Projekt das in Teamcenter gespeichert ist, ein neue Revision erstellen, werden Sie dazu aufgefordert, das Dataset einzuchecken. Sie können die Aufforderung das Dataset einzuchecken auch ablehnen.
3. Nach dem Speichern des Multiuser-Server Projekts in Teamcenter wird die Serverprojekt-Ansicht geschlossen.
4. Die neu erstellte Revision des Multiuser-Server Projekts wird in einer lokalen Session geöffnet.
5. Führen Sie in der lokalen Session eine Aktualisierung durch, um die Objekte der lokalen Session mit den aktuellen Daten vom Multiuser-Server zu aktualisieren.
   In der lokalen Session wird die Dateikennung und die neue Revision-ID angezeigt. Das Teamcenter Overlay-Symbol zeigt an, dass die neu erstellte Revision des Multiuser-Serverprojekts auf dem Multiuser-Server gespeichert wurde und mit Teamcenter verknüpft ist.

Hinweis

Nach erneutem Öffnen der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter Kontextmenüeinträge zur Verfügung:

- Speichern
- Speichern als neues Element...

- Speichern als neue Revision...
- Auschecken (nur gesperrte)
- Einchecken (nur entsperrte)

Siehe auch

Neue Revision eines mit Teamcenter verbundenen TIA Portal-Projekts erstellen (Seite 81)

5.12.7 TIA Portal-Projekt (Multiuser-Serverprojekt) aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern

5.12.7.1 TIA Portal-Projekt in ein in Teamcenter vorhandenes Element in der Serverprojekt-Ansicht speichern

Mit der Funktion "Speichern zu Element..." verbinden Sie ein TIA Portal-Projekt (Multiuser-Serverprojekt), das außerhalb des Cache-Servers von Teamcenter bearbeitet wurde, wieder mit Teamcenter.

Bei Änderungen des TIA Portal-Projekts (Multiuser-Serverprojekt) außerhalb des Cache-Servers von Teamcenter, wird die Datei "versionInfo.xml" aktualisiert.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.
- Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet, installiert haben.

Vorgehensweis

Um ein TIA Portal-Projekt in der Serverprojekt-Ansicht wieder mit Teamcenter zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Sicht "Projektnavigation (PNV)" das TIA Portal-Projekt.
2. Wählen Sie auf dem TIA Portal-Projekt, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern zu Element...".
   Der Dialog "Speichern zu Element..." wird geöffnet.
3. Im Feld "Name", wird standardmäßig der Name des ausgewählten TIA Portal-Projekts angezeigt.
   Ändern Sie gegebenenfalls den Namen im Feld "Name".

4. Im Feld "Element-ID", wird standardmäßig die zum TIA Portal-Projekt zugehörige Element-ID aus der Datei "versionInfo.xml" angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Element-ID" die vorhandene Element-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Element-ID" leer. Weisen Sie dem Feld "Element-ID" eine bereits gültige oder beliebige Teamcenter Element-ID zu.
     Maximal eingebbar sind 128 Zeichen.
5. Im Feld "Revision-ID" wird standardmäßig die zum TIA Portal-Projekt zugehörige Revision-ID aus der Datei "versionInfo.xml " angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Revision-ID" die vorhandene Revision-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Revision-ID" leer. Weisen Sie dem Feld "Revision-ID" eine bereits gültige oder beliebige Teamcenter Revision-ID zu.
     Maximal eingebbar sind 32 Zeichen.
6. Bestätigen Sie Ihre Eingabe mit "OK".
   Das TIA Portal-Projekt wird unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass die Operation "Speichern zu Element..." abgebrochen wird.

Hinweis

Beachten Sie:

Die Felder Element-ID / Revision-ID sind obligatorische Felder. Die Schaltfläche "OK" ist nur dann editierbar, wenn in den Feldern Element-ID / Revision-ID ein Wert eingetragen ist.

Vorabprüfung für das "Speichern zu Element..."

Bei Verwendung der Funktion "Speichern zu Element..." führt Teamcenter folgende Prüfungen durch:

- Die Prüfung auf das Vorhandensein des Elements.
- Die Prüfung des Freigabestatus des Datasets.
- Die Prüfung der Altdaten.

Siehe hierzu: Prüfungen bei Verwendung der Funktion "Speichern zu Element..." (Seite 89)

Beim Speichervorgang wird Folgendes überprüft:

1. Ob die im Multiuser-Server gespeicherte Dataset-Version und die zugehörige Revision-ID durch den aktuell angemeldeten Benutzer oder einen anderen Benutzer ausgecheckt ist.
2. Ob die Dataset-Version durch einen Benutzer ausgecheckt ist.
   Wenn die Dataset-Version durch keinen andern Benutzer ausgecheckt, wird beim Ausführen der Funktion "Speichern zu Element..." die Dataset-Version implizit ausgecheckt.

3. Ob das Multiuser-Server Projekt exportiert und mit der Dateikennung "*.zap" archiviert wurde. Zu dem archivierten Projekt darf keine Datei "versionInfo.xml" vorhanden sein. Falls es während des Export- und Archivierungsvorgangs zu Fehlern kommt, wird eine entsprechende Meldung angezeigt, dass der Export- und Archivierungsvorgang fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde. Um die Meldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung:

   Hinweis

   Abhilfe beim Abbruch des Export- und Archivierungsvorgangs:

   1. Erstellen Sie in Teamcenter eine neue Dataset-Version.
   2. Laden Sie die archivierte Projektdatei in das Dataset hoch.
      Dabei wird das Dataset ausgecheckt.
      Falls das Erstellen der Element-Revision nicht möglich ist, wird eine Fehlermeldung angezeigt, dass das Erstellen der Element-Revision fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
      Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
      – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.
        Es wird keine Kopie des TIA Portal-Projekts im TcCache gespeichert.
        Es wird kein TcCache-Pfad für das TIA Portal-Projekt erstellt.

4. Übertragen Sie die Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server.
   Falls die Übertragung, der Projektänderungen von der Serverprojekt-Ansicht auf den Multiuser-Server nicht möglich ist, wird eine entsprechende Meldung angezeigt.
   Um die Meldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.

Aktionen beim "Speichern zu Element..."

Nach Abschluss aller Prüfungen werden folgende Aktionen ausgeführt:

1. Das TIA Portal-Projekt wird archiviert und unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
2. Das zugehörige Dataset befindet sich im Status "Ausgecheckt" und ist mit dem folgenden Symbol gekennzeichnet.

   Das können Sie in der Ordneransicht von Teamcenter sehen.
3. Nach Abschluss der Aktionen, wird der Dialog "Das Projekt wurde erfolgreich nach Teamcenter gespeichert" angezeigt.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

"Rollback"-Funktionen

1. Falls der Export- und Archivierungsvorgang des Multiuser-Server Projekts fehlgeschlagen ist, wird im Multiuser-Server Projekt auf die ältere Version der Datei "versionInfo.xml" zugegriffen.
2. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie die exportierten und archivierten Dateien von der lokalen Festplatte.
3. Falls das Erstellen einer neuen Element-Revision fehlgeschlagen ist, entfernen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version) entfernen. Damit wird im Multiuser-Server Projekt zur ursprünglichen Version der Datei "versionInfo.xml" zugegriffen.

Empfehlung

Führen Sie im Multiuser-Server eine Aktualisierung der lokalen Session durch, um die Objekte der lokalen Session mit den aktuellen Daten des Multiuser-Servers zu aktualisieren.

In der lokalen Session wird die Namenserweiterung angezeigt.

In der lokalen Session wird die Dateikennung und die neue Revision-ID angezeigt. Das Teamcenter Overlay-Symbol zeigt an, dass das neu erstellte Element des Multiuser-Serverprojekts auf dem Multiuser-Server gespeichert wurde und mit Teamcenter verknüpft ist.

Ergebnis

Nach erneutem Öffnen der Serverprojekt-Ansicht stehen Ihnen folgende Teamcenter Kontextmenüeinträge zur Verfügung:

- Speichern als neues Element...
- Speichern als neue Revision...
- Speichern zu Element...
- Einchecken.
- Einchecken (nur entsperrte)
- Auschecken (nur gesperrte)
- Auschecken rückgängig machen.
- Export...

5.12.8 Ein mit Teamcenter Gateway verbundenes und dem Multiuser-Server hinzugefügtes TIA Portal-Projekt als Referezprojekt öffnen

5.12.8.1 Ein mit Teamcenter Gateway verbundenes und dem Multiuser-Server hinzugefügtes TIA Portal-Projekt als Referezprojekt öffnen

Die Option "Als Referenzprojekt öffnen" bietet Ihnen im Dialog "Projekt aus Teamcenter öffnen" die Möglichkeit ein TIA Portal-Projekt (Multiuser-Serverprojekt) als Referenzprojekt zu öffnen. Standardmäßig ist das Optionskästchen "Als Referenzprojekt öffnen" deaktiviert.

Vorgehen

Um ein TIA Portal-Projekt im Dialog "Projekt aus Teamcenter öffnen" als Referenzprojekt zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektansicht im Menü "Projekt" den Befehl "Projekt aus Teamcenter öffnen...".
   Der Dialog "Projekt aus Teamcenter öffnen" wird geöffnet.
   Weitere Informationen zum Dialog "Projekt aus Teamcenter öffnen" erhalten Sie unter: TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen (Seite 152).
2. Wählen Sie das entsprechende TIA Portal-Projekt aus dem Suchergebnis aus.
   Weitere Informationen hierzu finden Sie unter: Öffnen von TIA Portal-Projekten aus dem Suchergebnis (Seite 156).
3. Aktivieren Sie das Optionskästchen "Als Referenzprojekt öffnen".

   Hinweis

   Wenn Sie das Optionskästchen "Als Referenzprojekt öffnen" aktivieren, wird das Optionskästchen "Auschecken" deaktiviert.

4. Klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

Beim "Öffnen" des TIA Portal-Projekts (Multiuser-Serverprojekt) als Referenzprojekt werden die folgenden Aktionen ausgelöst:

1. Ein temporärer "RF"-Ordner (Referenzprojekt-Ordner) wird parallel zum Teamcenter Cache-Ordner erstellt.
   Der "RF"-Ordner (Referenzprojekt-Ordner) muss 2 Zeichen lang sein und parallel zum TcCache-Ordner.
   Beispiel:

aktueller Teamcenter Cache-Ordner	<TIA default projectlocation>\TC\<ItemID>\<RevID>\ProjectFolder
"RF"-Ordner (Referenzprojekt-Ordner)	<TIA default projectlocation>\RF\<ItemID>\<RevID>\ProjectFolder

2. Für die Pfadlänge des temporären "RF"-Ordners (Referenzprojekt-Ordner), gelten die Regeln, die auch für die Pfadlänge des Teamcenter Cache-Ordners gelten.
3. Das ausgewählte TIA Portal-Projekt wird in den temporären "RF"-Ordner (Referenzprojekt-Ordner) heruntergeladen, und in der Palette "Referenzprojekte" geöffnet.
   - In der Palette Referenzprojekte wird das TIA Portal-Projekt mit der Namenserweiterung und dem Teamcenter Overlay-Symbol angezeigt.
   - Das TIA Portal-Projekt wird unabhängig davon, ob es durch den aktuellen Benutzer, oder einen andern Benutzer ausgecheckt ist, geöffnet.
4. Wenn Sie eine ältere Version eines TIA Portal-Projekts als Referenzprojekt öffnen, erhalten Sie eine entsprechende Meldung das TIA Portal-Projekt zu aktualisieren.
5. Wenn Sie das TIA Portal-Projekt aktualisieren, wird in der Palette Referenzprojekte, der neue Name mit der Namenserweiterung und dem Teamcenter Overlay-Symbol angezeigt.

6. Wenn es beim Herunterladen eines TIA Portal-Projekts von Teamcenter zu Fehlern kommt, erhalten Sie eine entsprechende Meldung mit detaillierten Fehlerinformationen.
7. Empfehlung: Vergleichen Sie mit dem Vergleichseditor die benötigten Objekte des TIA Portal-Projekts aus der Projektnavigation mit dem Referenzprojekt aus der Palette Referenzprojekte.
8. Wenn Sie das TIA Portal-Projekt in der Palette Referenzprojekte schließen, wird der Revisions-Ordner mit dem TIA Portal-Projekt, aus dem temporären Referenzprojektordner gelöscht. Gleichzeitig wird das TIA Portal-Projekt auch aus dem temporären Ordner gelöscht.
9. Beachten Sie, dass das TIA Portal-Projekt nicht aus dem temporären Ordner gelöscht wird, wenn Sie das TIA Portal schließen.

Hinweis

Öffnen Sie immer zuerst die Palette "Referenzprojekte", bevor Sie das TIA Portal-Projekt aus Teamcenter heraus öffnen.

5.13 Globale Bibliotheken mit dem TIA Portal Teamcenter Gateway verwalten

5.13.1 Globale Bibliothek als neues Element in Teamcenter speichern

Voraussetzung

Im Folgenden wird vorausgesetzt, dass:

- zwischen dem TIA Portal und Teamcenter eine aktive Verbindung besteht.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Vorgehen

Um eine globale Bibliothek zum ersten Mal in Teamcenter zu speichern, gehen Sie folgendermaßen vor:

1. Öffnen Sie im TIA Portal die globale Bibliothek, die Sie als neues Element in Teamcenter speichern möchten.
2. Selektieren Sie in der Palette "Globale Bibliothek" die globale Bibliothek zum Speichern nach Teamcenter.
3. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neues Element...".
   Der Dialog "Speichern als neues Element" wird geöffnet.
   Das Register "Allgemein" wird standardmäßig angezeigt.
4. Geben Sie im Register "Allgemein" im Feld "Name" einen Namen ein oder übernehmen Sie die vorgeschlagenen Daten. Standardmäßig wird der Name der ausgewählten globalen Bibliothek angezeigt.
   Maximal eingebbar sind 94 Zeichen.
5. Wählen Sie im Register "Allgemein" im Feld "Elementtyp" einen Elementtyp für die globale Bibliothek die Sie als neues Element in Teamcenter speichern wollen aus.
   - Wählen Sie im Feld "Elementtyp" einen abhängigen Typen aus.
     Standardmäßig ist der Typ "T4TiaProject" ausgewählt.
     Die Klappliste ist nur editierbar, wenn abhängige Typen verfügbar sind.
6. Weisen Sie im Register "Allgemein" dem Feld "Element-ID / Revision-ID" eine bereits gültige oder beliebige Teamcenter Element-ID / Revision-ID zu.
   Wenn Sie das Feld "Element-ID / Revision-ID" 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID zugewiesen.

7. Durch klicken auf die Schaltfläche “Zuweisen” wird von Teamcenter automatisch die nächstmögliche Element-ID / Revision-ID der ausgewählten Bibliothek zugewiesen.
   Ergebnis:
   - Die zugewiesene Element-ID / Revision-ID wird im jeweiligen Textfeld angezeigt.
   - Die Textfelder sind nicht editierbar und die Schaltfläche "Zuweisen" ist deaktiviert.

   Hinweis

   Anpassung der Element-Typen

   Die Anpassung, der Element-Typen, hat auf die Felder "Element-ID / Revision-ID" und die Schaltfläche "Zuweisen" folgende Auswirkungen:
   1. Die den Feldern "Element-ID / Revision-ID" zugewiesenen IDs werden zurückgesetzt.
   2. Die Schaltfläche "Zuweisen" wird auf bedienbar gesetzt.

8. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Projekt", die Teamcenter Projekte aus, dem Sie die globale Bibliothek zuweisen möchten.
   - Klicken Sie auf die Schaltfläche "Durchsuchen" (...).
     Der Dialog "Projekte auswählen" wird geöffnet.
   - Wählen Sie die entsprechenden Teamcenter Projekte aus und bestätigen Sie jede Auswahl mit klicken auf die Schaltfläche "OK". Klicken auf die Schaltfläche "OK" bewirkt, dass die ausgewählten Teamcenter Projekte getrennt durch Komma im Textfeld angezeigt werden.
     Wenn Sie das Feld 'leer' lassen, wird die globale Bibliothek keinem Teamcenter Projekt zugewiesen.
9. Optional: Wählen Sie im Register "Allgemein" im Feld "Teamcenter Ordner" den Ordner aus, dem Sie das neu erstellte Element zuweisen möchten.
   - Klicken Sie auf die Schaltfläche "Durchsuchen" (...).
     Der Dialog "Teamcenter Ordner auswählen" wird geöffnet.
     Die Vorbelegung für diesen Ordner ist *"Home"*.
     Alle neu erstellen Elemente werden standardmäßig in Ihrem persönlichen Verzeichnis *"Home"* abgelegt.
   - Wählen Sie den Ordner *"Home"* aus und bestätigen Sie die Auswahl durch klicken auf die Schaltfläche "OK".
     Klicken auf die Schaltfläche "OK" bewirkt, dass der ausgewählte Ordner im Textfeld angezeigt wird.
     Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.
10. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
    Maximal eingebbar sind 240 Zeichen.

11. Bestätigen Sie Ihre Eingabe mit "OK".
    Die globale Bibliothek wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert.
    Sobald die globale Bibliothek in Teamcenter gespeichert wurde, öffnet sich der Dialog "Die globale Bibliothek wurde erfolgreich in Teamcenter gespeichert".
    – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".
      Ein Überlagerungssymbol zeigt im TIA Portal in der Palette "Globale Bibliothek" an, dass die globale Bibliothek mit Teamcenter verbunden ist.
12. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass der Dialog "Eigenschaften neues Element" geschlossen wird.

Ergebnis

Sobald die globale Bibliothek in Teamcenter gespeichert wurde, steht Ihnen in Teamcenter, im Ordner *"Home"*, das neue Element inkl. Element-ID / Revision-ID zur Verfügung. Das Dataset der globalen Bibliothek ist in Teamcenter als Zip-Datei unter der jeweiligen Revision-ID gespeichert.

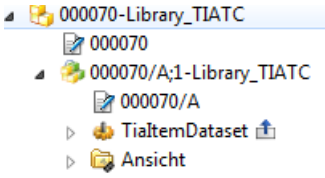

Das zugehörige Dataset ist "Ausgecheckt" und mit dem Symbol gekennzeichnet. Das können Sie in der Ordneransicht von Teamcenter sehen.

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen zum Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.

Globale Unternehmensbibliotheken

Hinweis

Achten Sie beim Speichern von globalen Bibliotheken darauf, dass Sie keine als "Unternehmensbibliotheken" gekennzeichnete globale Bibliothek in Teamcenter speichern. "Unternehmensbibliotheken" werden im "TC Cache" nur temporär zwischengespeichert, was zu unspezifischem Fehlverhalten zwischen dem TIA Portal und Teamcenter führen kann.

Siehe auch

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

Globale Bibliothek einchecken (Seite 134)

Auschecken einer globalen Bibliothek rückgängig machen (Seite 142)

Änderungen an einer mit Teamcenter verbunden globalen Bibliothek speichern (Seite 135)

Register "Allgemein" im Dialog "Speichern als neues Element" (globale Bibliotheken) (Seite 128)

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" (globale Bibliotheken) (Seite 128)

Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 129)

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 130)

5.13.1.1 Register "Allgemein" im Dialog "Speichern als neues Element" (globale Bibliotheken)

Im Register "Allgemein" des Dialogs "Speichern als neues Element" definieren Sie die Eigenschaften einer globalen Bibliothek die Sie in Teamcenter speichern wollen.

Siehe auch

Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 129)

5.13.1.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" (globale Bibliotheken)

Im Register "Zusätzliche Eigenschaften" bearbeiten Sie die Werte der hochgeladenen zusätzlichen Eigenschaften einer globalen Bibliothek.

Im Register "Zusätzliche Eigenschaften" haben Sie die Möglichkeit, Werte von hochgeladenen Elementtypen oder abhängigen Typen zu bearbeiten. Der Inhalt des Registers "Zusätzliche Eigenschaften" ist von den verwendeten Elementtypen, den abhängigen Typen oder den Eigenschaften von geerbten Typen abhängig.

Siehe auch

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 130)

Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (globale Bibliotheken) (Seite 131)

5.13.1.3 Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (globale Bibliothek)

Aufbau des Registers "Allgemein" im Dialog "Speichern als neues Element"

Die Ansicht des Registers "Allgemein" besteht aus folgenden Feldern und Schaltflächen:

Feld	Beschreibung
Name:	Eingabefeld für den Namen unter dem Sie die globale Bibliothek als neues Element in Teamcenter speichern wollen. Standardmäßig wird der Name der ausgewählten globalen Bibliothek angezeigt.
Elementtyp:	Eingabefeld für den Elementtyp der globalen Bibliothek. Standardmäßig ist der Typ T4TiaLibrary ausgewählt.
Element-ID / Revision-ID:	Eingabefeld für die Element-ID / Revision-ID die Sie dem neuen Element zuweisen wollen.
Teamcenter Projekt:	Auswahlfeld zur Auswahl des Teamcenter Projekts, dem Sie das neue Element zuweisen wollen.
Teamcenter Ordner:	Auswahlfeld zur Auswahl des Teamcenter Ordners, in den Sie das neue Element speichern wollen.
Kommentar:	Eingabefeld für einen Kommentar.

Schaltfläche	Beschreibung
Klappliste	Durch klicken auf die Klappliste können Sie abhängige Typen auswählen und hinzufügen. Die Klappliste ist nur editierbar, wenn abhängige Typen definiert sind.
Zuweisen	Durch klicken auf diese Schaltfläche weisen Sie der globalen Bibliothek, automatisch eine Element-ID / Revision-ID zu.
"Durchsuchen" (...) - (Teamcenter Projekt)	Durch klicken auf diese Schaltfläche wird der Dialog "Projekte auswählen" geöffnet.
"Durchsuchen" (...) - (Teamcenter Ordner)	Durch klicken auf diese Schaltfläche wird der Dialog "Teamcenter Ordner auswählen" geöffnet.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Siehe auch

Register "Allgemein" im Dialog "Speichern als neues Element" (globale Bibliotheken) (Seite 128)

5.13.1.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (globale Bibliothek)

Aufbau des Registers "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Die im Register "Zusätzliche Eigenschaften" tabellarisch angezeigten Informationen, sind davon abhängig:

- welchen Elementtypen oder welchen seiner abhängigen Typen Sie ausgewählt haben,
- welche Eigenschaften die ausgewählten Elementtypen und ihre abhängigen Typen geerbt haben, und welche Eigenschaften in der Attribut-Mapping-Datei abgebildet sind.

Eigenschaften des Registers "Zusätzliche Eigenschaften"

Im Register "Zusätzliche Eigenschaften" werden die Eigenschaften von Elementtypen und deren abhängigen Typen im Modus Lesend/Schreibend angezeigt.

Die Ansicht des Registers "Zusätzliche Eigenschaften" besteht aus folgenden Spalten und Schaltflächen:

Spalte	Bedeutung
Name	Name der zusätzlichen Eigenschaften.
Typ	Art der zusätzlichen Eigenschaften.
Wert	Wert der zusätzlichen Eigenschaften.

Schaltfläche	Bedeutung
Klappliste	Durch klicken auf die Klappliste können Sie Pflichtattribute hinzufügen.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Hinweis zur Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Folgende Regeln gelten für die Bedienbarkeit der Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element":

- Wenn die erforderlichen Pflichtattribute nicht befüllt sind, ist die Schaltfläche "OK" gegraut und nicht bedienbar.
- Wenn keine Pflichtattribute vorhanden sind, oder wenn alle Pflichtattribute befüllt sind, ist die Schaltfläche "OK" bedienbar.

Siehe auch

Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (globale Bibliotheken) (Seite 131)

Aufbau Register "Allgemein" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 129)

5.13.1.5 Werte in der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (globale Bibliotheken)

In der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" werden, die in den Teamcenter (Server), hochgeladenen Attribute einer globalen Bibliothek im Modus lesend/ schreibend angezeigt.

Voraussetzungen um die Werte von hochgeladenen Attributen einer globalen Bibliothek zu bearbeiten

Um die Werte der hochgeladenen Attribute einer globalen Bibliothek zu bearbeiten, wird vorausgesetzt, dass:

- die Attribute des ausgewählten Elementtyps einer globalen Bibliothek hochgeladen sind.
- die Attribute der ausgewählten Element-Revision des Elementtyps einer globalen Bibliothek hochgeladen sind.
- die hochgeladenen Attribute des Element Masterformulars mit dem TIA Portal-Projekt Elementtyp verbunden sind.
- die hochgeladenen Attribute des Element-Revision Masterformulars mit der Element-Revision der globalen Bibliothek verbunden sind.

Hinweis

Beachten Sie, dass

nur die oben genannten Szenarien ab TIA Portal Teamcenter Gateway V14SP1 unterstützt werden. Alle anderen Szenarien werden ignoriert.

Beachten Sie:

- wenn keine Attribute in den Teamcenter (Server) hochgeladen sind, in der Registerkarte “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.
- wenn das Lesen von hochgeladenen Attributen während des Verbindungsaufbaus mit TIA Portal Teamcenter Gateway fehlgeschlagen ist, in der Registerkarte “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.

Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften"

Beachten Sie Folgendes:

Bei Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften", wird immer nur der Eintrag des Attributs aus dem Register "Allgemein" in Teamcenter gespeichert.

Mehrfachverwendung von Attributen im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element"

Beachten Sie Folgendes:

Bei Mehrfachverwendung desselben Attributs, im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element", wird immer nur der erste Eintrag des Attributs aus der tabellarischen Ansicht in Teamcenter gespeichert.

Verwendung von Sonderzeichen für den Datentyp 'Char'

Nachfolgend werden die Namenskonventionen für den Datentyp 'Char', die Sie beim "Speichern als neues Element" beachten müssen beschrieben.

Beachten Sie Folgendes:

- Verwenden Sie in den Eingabefeldern des Datentyps 'Char', im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element", nur die ASCII-Werte 32-126.

Siehe auch

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" (globale Bibliotheken) (Seite 128)

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neues Element" - (globale Bibliothek) (Seite 130)

5.13.2 Globale Bibliothek auschecken

Um ein oder mehrere globale Bibliotheken exklusiv für sich zur Bearbeitung zu reservieren, können Sie die globalen Bibliotheken auschecken. Dadurch sind die globalen Bibliotheken für andere Benutzer gesperrt.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- die globale Bibliothek mit Teamcenter verbunden ist.
- die globale Bibliothek im Modus "Nur Lesen" geöffnet ist.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Vorgehen

Um eine globale Bibliothek zu bearbeiten, müssen Sie die globale Bibliothek auschecken, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Auschecken".

Hinweis

Globale Bibliothek auschecken

Beachten Sie: wenn die globale Bibliothek lokal geändert wurde, oder die Dataset-Version älter ist, als die aktuell verfügbare Dataset-Version in Teamcenter, der Dialog "Teamcenter Bibliothek öffnen" angezeigt wird.

1. Klicken auf die Schaltfläche "Ja" bewirkt, dass die aktuell verfügbare Dataset-Version überschrieben wird.
2. Klicken auf die "Schaltfläche "Nein" bewirkt, dass die aktuell verfügbare Dataset-Version als lokale Kopie geöffnet wird.
3. Durch klicken auf die Schaltfläche "Abbrechen" wird das Auschecken abgebrochen.

Wenn Sie das Auschecken fortsetzen, wird die Dataset-Version der globalen Bibliothek in Teamcenter ausgecheckt. Die Dataset-Version ist für die Bearbeitung durch andere Benutzer gesperrt.

Ergebnis

Die globale Bibliothek wird wieder im Modus "Schreiben" geöffnet.

Die Dataset-Version der globalen Bibliothek ist ausgecheckt und in Teamcenter mit dem Symbol gekennzeichnet.

Die Dataset-Version ist für die Bearbeitung durch andere Benutzer gesperrt.

Im TIA Portal stehen Ihnen nach erfolgreichem "Auschecken" folgende Teamcenter-Operationen für das Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.

Siehe auch

Globale Bibliothek als neues Element in Teamcenter speichern (Seite 125)

Globale Bibliothek einchecken (Seite 134)

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

Auschecken einer globalen Bibliothek rückgängig machen (Seite 142)

5.13.3 Globale Bibliothek einchecken

Globale Bibliotheken, die Sie nach Teamcenter gespeichert haben, befinden sich im Zustand "ausgecheckt". Dadurch sind diese globalen Bibliotheken für andere Benutzer gesperrt.

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- die globale Bibliothek mit Teamcenter verbunden ist.
- das Dataset der zugehörigen Revision der globalen Bibliothek durch Sie ausgecheckt ist.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Vorgehen

Um die Bearbeitungssperre der globalen Bibliotheken aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Einchecken".

Automatisches einchecken

Automatisches einchecken erfolgt beim Schließen der globalen Bibliothek im TIA Portal unter folgenden Bedingungen:

- Wenn die "VersionInfo.xml-Datei" für eine globale Bibliothek vorhanden ist.
- Wenn die globale Bibliothek im Modus "Schreiben" geöffnet ist.

Automatisches einchecken erfolgt beim Schließen der globalen Bibliothek unter folgenden Bedingungen **nicht**:

- Wenn die "VersionInfo.xml-Datei" für eine globale Bibliothek **nicht** vorhanden.
- Wenn die globale Bibliothek im Modus "Lesen/Schreiben" geöffnet ist.
- Wenn die globale Bibliothek in RAC ausgecheckt ist.

Ergebnis

Die globale Bibliothek wird wieder im Modus "Lesen" geöffnet.

Die globale Bibliothek ist in Teamcenter eingecheckt und in der Ordneransicht von Teamcenter wird das Symbol entfernt. Die Bearbeitungssperre der globalen Bibliothek ist aufgehoben.

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen für das Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Auschecken.

Siehe auch

Globale Bibliothek als neues Element in Teamcenter speichern (Seite 125)

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

Globale Bibliothek auschecken (Seite 132)

5.13.4 Änderungen an einer mit Teamcenter verbunden globalen Bibliothek speichern

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- das TIA Portal mit Teamcenter verbunden ist.
- die globale Bibliothek im Modus "Lesen/Schreiben" geöffnet ist.
- die ausgewählte globale Bibliothek mit Teamcenter verbunden ist und Änderungen vorgenommen wurden.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Hinweis

Beachten Sie folgendes:

- Wenn keine Verbindung zwischen dem TIA Portal und Teamcenter besteht, ist der Befehl "Teamcenter > Speichern" deaktiviert.

Vorgehen

Um Änderungen an einer mit Teamcenter verbunden globalen Bibliothek und dem zugehörigen Dataset zu speichern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "globale Bibliothek" die entsprechende globale Bibliothek aus.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern".
   Die neue Dataset-Version der globalen Bibliothek ist in Teamcenter als Zip-Datei unter der jeweiligen Revision-ID gespeichert.

Einschränkungen

Folgende Einschränkungen sind zu beachten:

In Teamcenter kann die Anzahl der gespeicherten Dataset-Versionen durch einen Teamcenter Administrator konfiguriert werden (default = 3). Wenn das Maximum erreicht ist, wird die älteste Dataset-Version gelöscht und die neue Dataset-Version erstellt.

Ergebnis

In Teamcenter wurde eine neue Dataset-Version der globalen Bibliothek erstellt

Im TIA Portal stehen Ihnen nach dem "Speichern" folgende Teamcenter-Operationen für das Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.
- Export...

Siehe auch

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

Globale Bibliothek als neues Element in Teamcenter speichern (Seite 125)

Auschecken einer globalen Bibliothek rückgängig machen (Seite 142)

Globale Bibliothek einchecken (Seite 134)

5.13.5 Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen

Voraussetzung

Im Folgenden wird vorausgesetzt, dass

- die globale Bibliothek mit Teamcenter verbunden ist.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Vorgehen

Um von einer globalen Bibliothek, die in Teamcenter gespeichert ist, eine neue Revision zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern als neue Revision...".
   Bei dieser Funktion wird die aktuell geöffnete Revision der globalen Bibliothek, als Basis für das Erstellen der neuen Revision verwendet.
   - Der Dialog "Speichern als neue Revision" wird geöffnet.
   - Das Register "Allgemein" wird standardmäßig angezeigt.
   - Im Feld "Name" wird der Name der ausgewählten globalen Bibliothek, (z.B. t4tia_object_name), nur lesend angezeigt.
   - Im Feld "Element-ID" wird die "Element-ID" der ausgewählten globalen Bibliothek nur lesend angezeigt.
3. Optional: Weisen Sie im Register "Allgemein" im Feld "Revision-ID" der globalen Bibliothek eine gültige Teamcenter Revision-ID zu.
   Maximal eingebbar sind 32 Zeichen.
4. Durch klicken auf die Schaltfläche “Zuweisen” wird durch Teamcenter automatisch die nächstmögliche Revision-ID der globalen Bibliothek zugewiesen.
   Ergebnis:
   - Die zugewiesene Revision-ID wird im Feld "Revision-ID" angezeigt.
   - Das Feld "Revision-ID" ist nicht mehr editierbar.
   - Die Schaltfläche "Zuweisen" ist deaktiviert.

   Wenn Sie das Feld 'leer' lassen, wird von Teamcenter automatisch die nächstmögliche Revision-ID zugewiesen.
5. Optional: Geben Sie im Register "Allgemein" im Feld "Kommentar" wenn erforderlich, einen Kommentar ein.
   Maximal eingebbar sind 240 Zeichen.
   Standardmäßig werden die Kommentare der Vorgängerversion übernommen. Sie können die Kommentare aus der Vorgängerversion bearbeiten oder unverändert beibehalten.
6. Bestätigen Sie Ihre Eingabe mit "OK".
   Die globale Bibliothek wird in der angegebenen Ablagestruktur unter dem angegebenen Namen in Teamcenter gespeichert.
   Sobald die globale Bibliothek in Teamcenter gespeichert wurde, öffnet sich der Dialog "Die globale Bibliothek wurde erfolgreich in Teamcenter gespeichert".
   - Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".
     Ein Überlagerungssymbol zeigt im TIA Portal in der Palette "Globale Bibliothek" an, dass die globale Bibliothek mit Teamcenter verbunden ist.
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass der Dialog "Speichern als neue Revision..." geschlossen wird.

Ergebnis

Die neue Revision, der globalen Bibliothek, wird in Teamcenter erstellt und als neues Objekt unterhalb der letzten Revision angezeigt.

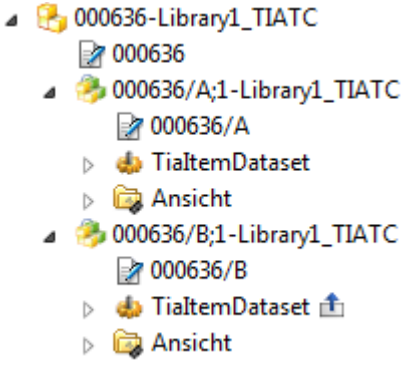

Beim Revisionieren der globalen Bibliothek wird das Dataset der aktuell geöffneten globalen Bibliothek übernommen.

Das zur neuen Revision zugehörige Dataset der globalen Bibliothek ist "Ausgecheckt" und in Teamcenter mit dem Symbol gekennzeichnet.

Das Dataset der älteren Revision wird eingecheckt, in Teamcenter wird das Symbol entfernt.

Das können Sie in der Ordneransicht von Teamcenter sehen.

Im TIA Portal wird die ältere Revision der globalen Bibliothek geschlossen. Die neue Revision der globalen Bibliothek wird im TIA Portal im Modus "Schreiben" geöffnet.

Nach dem "Speichern als neue Revision" stehen Ihnen im TIA Portal folgende Teamcenter-Operationen zum Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Einchecken.
- Auschecken rückgängig machen.

Siehe auch

Globale Bibliothek als neues Element in Teamcenter speichern (Seite 125)

Globale Bibliothek einchecken (Seite 134)

Auschecken einer globalen Bibliothek rückgängig machen (Seite 142)

5.13.5.1 Register "Allgemein" im Dialog "Speichern als neue Revision" (globale Bibliotheken)

Im Register "Allgemein" des Dialogs "Speichern als neue Revision" definieren Sie die Eigenschaften einer globalen Bibliothek die Sie als neue Revision in Teamcenter speichern wollen.

5.13.5.2 Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" (globale Bibliotheken)

Im Register "Zusätzliche Eigenschaften" bearbeiten Sie die Werte, der hochgeladenen zusätzlichen Eigenschaften, einer globalen Bibliothek, die Sie als neue Revision nach Teamcenter speichern wollen.

Im Register "Zusätzliche Eigenschaften" haben Sie die Möglichkeit, Werte von hochgeladenen Elementtypen oder abhängigen Typen zu bearbeiten. Der Inhalt des Registers "Zusätzliche Eigenschaften" ist von den verwendeten Elementtypen, den abhängigen Typen oder den Eigenschaften geerbter Typen abhängig.

Siehe auch

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (globale Bibliothek) (Seite 140)

5.13.5.3 Aufbau Register "Allgemein" im Dialog "Speichern als neue Revision" - (globale Bibliothek)

Aufbau des Registers "Allgemein" im Dialog "Speichern als neue Revision"

Die Ansicht des Registers "Allgemein" besteht aus folgenden Feldern und Schaltflächen:

Feld	Beschreibung
Name (nur lesend):	Im Feld "Name" wird der Name des ausgewählten TIA Portal-Projekts (z.B. t4tia_object_name) lesend angezeigt
Element-ID (nur lesend):	Im Feld "Element-ID" wird die "Element-ID" des ausgewählten TIA Portal-Projekts lesend angezeigt
Revision-ID:	Eingabefeld für eine gültige Teamcenter Revision-ID, die Sie der neuen Revision zuweisen wollen.
Kommentar:	Eingabefeld für einen Kommentar.

Schaltfläche	Beschreibung
Zuweisen:	Durch klicken auf diese Schaltfläche weisen Sie der globalen Bibliothek, automatisch eine Revision-ID zu.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Siehe auch

Register "Allgemein" im Dialog "Speichern als neue Revision" (globale Bibliotheken) (Seite 138)

5.13.5.4 Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (globale Bibliothek)

Aufbau des Registers "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Die im Register "Zusätzliche Eigenschaften" tabellarisch angezeigten Informationen, sind davon abhängig:

- welchen Elementtypen oder welchen seiner abhängigen Typen Sie ausgewählt haben,
- welche Eigenschaften die ausgewählten Elementtypen und ihre abhängigen Typen geerbt haben.

Eigenschaften des Registers "Zusätzliche Eigenschaften"

Im Register "Zusätzliche Eigenschaften" werden die Eigenschaften von Elementtypen und deren abhängigen Typen im Modus Lesend/Schreibend angezeigt.

Die Ansicht des Registers "Zusätzliche Eigenschaften" besteht aus folgenden Spalten und Schaltflächen:

Spalte	Bedeutung
Name	Name der zusätzlichen Eigenschaften.
Typ	Art der zusätzlichen Eigenschaften.
Wert	Wert der zusätzlichen Eigenschaften.

Schaltfläche	Bedeutung
Klappliste	Durch klicken auf die Klappliste können Sie Pflichtattribute hinzufügen.
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

Hinweis zur Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Folgende Regeln gelten für die Bedienbarkeit der Schaltfläche "OK" im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision":

- Wenn die erforderlichen Pflichtattribute nicht befüllt sind, ist die Schaltfläche "OK" gegraut und nicht bedienbar.
- Wenn keine Pflichtattribute vorhanden sind, oder wenn alle Pflichtattribute befüllt sind, ist die Schaltfläche "OK" bedienbar.

Siehe auch

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" (globale Bibliotheken) (Seite 139)

Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte) (Seite 141)

5.13.5.5 Werte in der tabellarische Ansicht des Registers "Zusätzliche Eigenschaften" bearbeiten - (TIA Portal-Projekte)

In der tabellarischen Ansicht des Registers "Zusätzliche Eigenschaften" werden, die in den Teamcenter Server, hochgeladene Attribute eines TIA Portal-Projekts im Modus lesend/ schreibend angezeigt.

Voraussetzungen um die Werte von hochgeladenen Attributen eines TIA Portal-Projekts zu bearbeiten

Um die Werte der hochgeladenen Attribute eines TIA Portal-Projekts zu bearbeiten, wird vorausgesetzt, dass:

- die Attribute des ausgewählten TIA Portal-Projekt Elementtyps hochgeladen sind.
- die Attribute der ausgewählten Element-Revision des TIA Portal-Projekt Elementtyps hochgeladen sind.
- die hochgeladenen Attribute des Element Masterformulars mit dem TIA Portal-Projekt Elementtyp verbunden sind.
- die hochgeladenen Attribute des Element-Revision Masterformulars mit der TIA Portal-Projekt Element-Revision verbunden sind.

Hinweis

Beachten Sie, dass

nur die oben genannten Szenarien ab TIA Portal Teamcenter Gateway V14SP1 unterstützt werden. Alle anderen Szenarien werden ignoriert.

Beachten Sie:

- wenn keine Attribute in den Teamcenter (Server) hochgeladen sind, in der Registerkarte “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.
- wenn das Lesen von hochgeladenen Attributen während des Verbindungsaufbaus mit TIA Portal Teamcenter Gateway fehlgeschlagen ist, in der Registerkarte “Zusätzliche Eigenschaften” die folgende Information angezeigt wird:
  Es wurden keine "Zusätzlichen Eigenschaften" hochgeladen.

Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften"

Beachten Sie Folgendes:

Bei Verwendung desselben Attributs im Register "Allgemein" und im Register "Zusätzliche Eigenschaften", wird immer nur der Eintrag des Attributs aus dem Register "Allgemein" in Teamcenter gespeichert.

Mehrfachverwendung von Attributen im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision"

Beachten Sie Folgendes:

Bei Mehrfachverwendung desselben Attributs, im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision", wird immer nur der erste Eintrag des Attributs aus der tabellarischen Ansicht in Teamcenter gespeichert.

Verwendung von Sonderzeichen für den Datentyp 'Char'

Nachfolgend werden die Namenskonventionen für den Datentyp 'Char', die Sie beim "Speichern als neue Revision" beachten müssen beschrieben.

Beachten Sie Folgendes:

- Verwenden Sie in den Eingabefeldern des Datentyps 'Char', im Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision", nur die ASCII-Werte 32-126.

Siehe auch

Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" (globale Bibliotheken) (Seite 139)

Aufbau Register "Zusätzliche Eigenschaften" im Dialog "Speichern als neue Revision" - (globale Bibliothek) (Seite 140)

5.13.6 Auschecken einer globalen Bibliothek rückgängig machen

Um vorgenommene Änderungen an einer ausgecheckten globalen Bibliothek zu verwerfen, haben Sie die Möglichkeit, das Auschecken rückgängig zu machen.

Voraussetzung

im Folgenden wird vorausgesetzt, dass

- die globale Bibliothek mit Teamcenter verbunden ist.
- die globale Bibliothek im Modus "Schreiben" geöffnet ist.
- keine Multiselektion von mehreren globalen Bibliotheken unterstützt wird.

Vorgehen

Um das Auschecken einer globalen Bibliothek rückgängig zu machen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Auschecken rückgängig machen".
   Der Dialog "Auschecken rückgängig machen" wird geöffnet.

3. Klicken Sie auf die Schaltfläche "OK", um das Auschecken rückgängig zu machen.
4. Klicken Sie auf die Schaltfläche "Abbrechen", wenn Sie "Auschecken rückgängig machen" abbrechen möchten.

Hinweis

Alle Änderungen, die seit dem letzten Auschecken vorgenommen wurden, werden verworfen.

Ergebnis

Das Auschecken der globalen Bibliothek wird rückgängig gemacht. In der Ordneransicht von Teamcenter wird das Symbol entfernt.

Die Bearbeitungssperre der globalen Bibliothek ist aufgehoben.

Im TIA Portal stehen Ihnen nach "Auschecken rückgängig machen" folgende Teamcenter-Operationen für das Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neue Revision....
- Speichern als neues Element....
- Auschecken.

Siehe auch

Globale Bibliothek auschecken (Seite 132)

Globale Bibliothek als neues Element in Teamcenter speichern (Seite 125)

Neue Revision einer mit Teamcenter verbundenen globalen Bibliothek erstellen (Seite 136)

5.13.7 Globale Bibliothek exportieren

Das Teamcenter Kontextmenü "Speichern" und "Export..." steht in gegenseitiger Abhängigkeit. Der Zustand des Kontextmenüs "Speichern" und "Export..." ist von folgenden Bedingungen abhängig.

Teamcenter Kontextmenü "Export..."

Das Teamcenter Kontextmenü "Export..." ist aktiviert, wenn

- die ausgewählte globale Bibliothek mit Teamcenter verbunden ist und keine Änderungen vorgenommen wurden.

Vorgehen

Um ein globale Bibliothek zu exportieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "Globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Export...".

Hinweis

Beachten Sie Folgendes:

Wenn der Befehl "Teamcenter > Export..." für die globale Bibliothek nicht aktiv ist, wurden in der globalen Bibliothek Änderungen vorgenommen. Speichern Sie die globale Bibliothek mit dem Befehl "Teamcenter > Speichern". Danach ist der Befehl "Teamcenter > Export..." aktiv.

Ergebnis

Der Befehl "Teamcenter > Export..." entfernt die Teamcenter-Metainformationen aus der globalen Bibliothek und speichert die geänderte globale Bibliothek in den von Ihnen ausgewählten Ordner.

Hinweis

Beachten Sie Folgendes:

Eine bereits vorhandene globale Bibliothek kann nicht an denselben Speicherort unter demselben Namen erneut exportiert werden.

Bitte beachten Sie:

1. Das Teamcenter Kontextmenü "Speichern" ist deaktiviert, wenn zwischen dem TIA Portal und Teamcenter keine Verbindung besteht.
2. Das Teamcenter Kontextmenü "Export..." ist deaktiviert, wenn in der ausgewählten globalen Bibliothek Änderungen vorgenommen wurden.

5.13.7.1 Register "Info" im Dialog "Export" - (globale Bibliothek)

Im Register "Info" werden nach dem Export die Informationen zum Speicherort der globalen Bibliothek angezeigt.

5.13.8 Globale Bibliothek aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern

5.13.8.1 Prüfungen bei der Verwendung der Funktion "Speichern zu Element..." - (Globale Bibliotheken)

Mit der Verwendung der Funktion "Speichern zu Element..." führt Teamcenter folgende Prüfungen durch.

1. Mit der Element-ID / Revision-ID wird geprüft, ob die Element-ID / Revision-ID in Teamcenter vorhanden ist.
   Wenn die Element-ID / Revision-ID nicht in Teamcenter vorhanden ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
2. Mit der Element-ID / Revision-ID wird geprüft, ob der Elementtyp in Teamcenter mit dem "T4TiaProject" Elementtyp und seinem abhängigen Typen übereinstimmt.
   Wenn der Elementtyp in Teamcenter mit dem "T4TiaProject" Elementtyp und seinem abhängigen Typen nicht übereinstimmt, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
3. Mit der Element-ID / Revision-ID wird geprüft, ob in dem Element und der zugehörigen Revision ein gültiger TIA Dataset vorhanden ist.
   Wenn in dem Element und der zugehörigen Revision kein gültiger TIA Dataset vorhanden ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
4. Mit der Element-ID / Revision-ID wird geprüft, ob der Name des TIA Portal Objekts (Name der globalen Bibliothek) mit dem Namen des Datasets übereinstimmt.
   Wenn der Name der globalen Bibliothek nicht mit dem Namen des Datasets übereinstimmt, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.
5. Mit der Element-ID / Revision-ID wird geprüft, ob das Dataset des Elements und die zugehörige Revision in Teamcenter durch einen anderen Benutzer ausgecheckt ist.
   Wenn das Dataset des Elements und die zugehörige Revision in Teamcenter durch einen anderen Benutzer ausgecheckt ist, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.

6. Mit der Element-ID / Revision-ID wird geprüft, ob sich die Version der globalen Bibliothek von der Version von Teamcenter unterscheidet.
   Falls sich die Version der globalen Bibliothek von der Version in Teamcenter unterscheidet, können Sie im Dialog "Kritische Meldung" entscheiden, ob Sie die Version in Teamcenter mit Ihren Änderungen überschreiben möchten oder nicht.
   - Um die Version der globalen Bibliothek in Teamcenter mit der neuen oder geänderten Version zu überschreiben, klicken Sie auf die Schaltfläche "Ja".
   - Um das "Speichern zu Element..." abzubrechen, klicken Sie auf die Schaltfläche "Nein".
7. Mit der Element-ID / Revision-ID wird geprüft:
   - Ob der Benutzer über die nötigen Zugriffsrechte verfügt.
   - Welchen Freigabe Status die Element-ID / Revision-ID hat.
     Wenn der Benutzer nicht über die nötigen Zugriffsrechte verfügt und die Element-ID / Revision-ID den Freigabe Status "Released" hat, wird von Teamcenter ein Dialog mit einer entsprechenden Meldung angezeigt.
   - Klicken Sie auf die Schaltfläche "OK" oder schließen Sie den Dialog.

5.13.8.2 Globale Bibliothek aus dem TIA Portal heraus in ein in Teamcenter vorhandenes Element speichern

Mit der Funktion "Speichern zu Element..." verbinden Sie eine globale Bibliothek, die außerhalb des Cache-Servers von Teamcenter bearbeitet wurde, wieder mit Teamcenter.

Bei Änderungen einer globalen Bibliothek außerhalb des Cache-Servers von Teamcenter, wird die Datei "versionInfo.xml" aktualisiert.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.

Vorgehensweis

Um die globale Bibliothek wieder mit Teamcenter zu verbinden, gehen Sie folgendermaßen vor:

1. Öffnen Sie im TIA Portal die globale Bibliothek, die Sie als neues Element in Teamcenter speichern möchten.
2. Selektieren Sie in der Palette "Globale Bibliothek" die entsprechende globale Bibliothek.
3. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern zu Element...".
   Der Dialog "Speichern zu Element..." wird geöffnet.
4. Im Feld "Name" wird standardmäßig der Name der ausgewählten globalen Bibliothek angezeigt.

5. Im Feld "Element-ID" wird standardmäßig die zur globalen Bibliothek zugehörige Element-ID aus der Datei "versionInfo.xml " angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Element-ID" die vorhandene Element-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Element-ID" leer. Weisen Sie dem Feld "Element-ID" eine bereits gültige oder beliebige Teamcenter Element-ID zu.
     Maximal eingebbar sind 128 Zeichen.
6. Im Feld "Revision-ID" wird standardmäßig die zur globalen Bibliothek zugehörige Revision-ID aus der Datei "versionInfo.xml " angezeigt.
   - Optional: Wenn erforderlich, ändern Sie im Feld "Revision-ID" die vorhandene Revision-ID.
   - Wenn die Datei "versionInfo.xml" nicht vorhanden ist, ist das Feld "Revision-ID" leer. Weisen Sie dem Feld "Revision-ID" eine bereits gültige oder beliebige Teamcenter Revision-ID zu.
     Maximal eingebbar sind 32 Zeichen.
7. Bestätigen Sie Ihre Eingabe mit "OK".
   Die globale Bibliothek wird unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert
8. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass die Operation "Speichern zu Element..." abgebrochen wird.

Hinweis

Bitte beachten Sie:

Die Felder Element-ID / Revision-ID sind obligatorische Felder. Die Schaltfläche "OK" ist nur dann editierbar, wenn in den Feldern Element-ID / Revision-ID ein Wert angezeigt wird.

Vorabprüfung für das "Speichern zu Element..."

Bei Verwendung der Funktion "Speichern zu Element..." führt Teamcenter folgende Prüfungen durch:

- Die Prüfung auf das Vorhandensein des Elements.
- Die Prüfung des Freigabestatus des Datasets.
- Die Prüfung der Altdaten.

Siehe hierzu: Prüfungen bei Verwendung der Funktion "Speichern zu Element..." (Seite 89)

Beim Speichervorgang wird Folgendes überprüft:

1. Ob die in Teamcenter gespeicherte Dataset-Version und die zugehörige Revision durch den aktuell angemeldeten oder einen anderen Benutzer ausgecheckt ist.
   Falls die Dataset-Version nicht ausgecheckt ist, wird beim Ausführen der Funktion "Speichern zu Element..." die Dataset-Version implizit ausgecheckt.
2. Ob die globale Bibliothek außerhalb des Cache-Servers von Teamcenter geöffnet wurde.
   Falls die globale Bibliothek, außerhalb des Teamcenter Cache Pfads geöffnet wurde, müssen Sie die globale Bibliothek in den Teamcenter Cache Pfad speichern.
3. Ob die globale Bibliothek außerhalb des Cache-Servers von Teamcenter geöffnet wurde oder ob im Cache-Server von Teamcenter eine globale Bibliothek unter derselben Element-ID / Revision-ID vorhanden ist.
   Falls die globale Bibliothek, außerhalb des Cache-Servers von Teamcenter geöffnet wurde, und im Cache-Server von Teamcenter eine globale Bibliothek unter derselben Element-ID / Revision-ID vorhanden ist, wird im Dialog "Kritische Meldung" eine entsprechende Meldung angezeigt:
   – "Eine Dataset-Version der globalen Bibliothek befindet sich im Cache-Server von Teamcenter. Möchten Sie die Dataset-Version überschreiben?"
     Um die Dataset-Version im Cache-Server von Teamcenter zu überschreiben, klicken Sie auf die Schaltfläche "Ja".
     Um die Funktion "Speichern zu Element..." abzubrechen, klicken Sie auf die Schaltfläche "Nein" oder schließen Sie den Dialog.

   Falls es während des Export- und Archivierungsvorgangs zu Fehlern kommt, wird ein Dialog mit einer entsprechenden Meldung angezeigt.

Hinweis

Abhilfe beim Abbruch des Export- und Archivierungsvorgangs:

1. Erstellen Sie in Teamcenter eine neue Dataset-Version.
2. Laden Sie die globale Bibliothek in das Dataset hoch.
   Dabei wird das Dataset ausgecheckt.
   Falls das Erstellen der Element-Revision nicht möglich ist, wird eine Fehlermeldung angezeigt, dass das Erstellen der Element-Revision fehlgeschlagen ist und dass der Export- und Archivierungsvorgang abgebrochen wurde.
   Um die Fehlermeldung zu bestätigen, gehen Sie wie folgt vor:
   – Klicken Sie auf die Schaltfläche "OK" oder schließen Sie die Meldung.
     Es wird keine Kopie der globalen Bibliothek im TcCache gespeichert.
     Es wird kein TcCache-Pfad für die globale Bibliothek erstellt.

Aktionen beim "Speichern zu Element..."

Nach Abschluss aller Prüfungen werden folgende Aktionen ausgeführt:

1. Die globale Bibliothek wird archiviert und unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
2. Das zugehörige Dataset befindet sich im Status "Ausgecheckt" und ist mit dem folgenden Symbol gekennzeichnet.

   Das können Sie in der Ordneransicht von Teamcenter sehen.
3. Nach Abschluss der Aktionen, wird der Dialog "Die globale Bibliothek wurde erfolgreich nach Teamcenter gespeichert" geöffnet.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

"Rollback"-Funktionen

1. Falls der Export- und Archivierungsvorgang fehlgeschlagen ist, wird auf die ältere Version der Datei "versionInfo.xml" zugegriffen.
2. Falls der Export- und Archivierungsvorgang fehlgeschlagen ist, entfernen Sie die exportierten und archivierten Dateien von der lokalen Festplatte.
3. Falls der Export- und Archivierungsvorgang fehlgeschlagen ist, entfernen Sie in Teamcenter alle erzeugten Objekte (Dataset-Version). Damit wird auf die ursprünglichen Version der Datei "versionInfo.xml" zugegriffen.
   Sobald die globale Bibliothek mit der Funktion "Speichern zu Element..." in Teamcenter gespeichert wurde, wird der Dialog "Die globale Bibliothek wurde erfolgreich nach Teamcenter gespeichert" geöffnet.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Ergebnis

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen zum Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neues Element...
- Speichern als neue Revision...
- Speichern zu Element...
- Einchecken.
- Auschecken rückgängig machen.
- Export...

5.13.8.3 Aufbau Dialog "Speichern zu Element..."

Im Dialog "Speichern zu Element..." definieren Sie die Eigenschaften einer globalen Bibliothek die Sie in Teamcenter speichern wollen.

Aufbau des Dialogs "Speichern zu Element..."

Die Ansicht des Dialogs "Speichern zu Element..." besteht aus folgenden Feldern und Schaltflächen.

Feld	Beschreibung
Name:	Eingabefeld für den Namen unter dem Sie die globale Bibliothek in Teamcenter speichern wollen. Standardmäßig wird der Name der ausgewählten globalen Bibliothek angezeigt.
Element-ID / Revision-ID:	Eingabefeld für die Element-ID / Revision-ID die Sie der globalen Bibliothek zuweisen wollen.

Schaltfläche	Bemerkung
OK	Durch klicken auf die Schaltfläche "OK" bestätigen Sie Ihre Eingaben. Der Dialog wird geschlossen.
Abbrechen	Durch klicken auf die Schaltfläche "Abbrechen" verlassen Sie den Dialog. Ihre Eingaben werden nicht übernommen.

5.13.9 Globale Bibliothek die im Teamcenter-Cache Ordner gespeichert ist wieder in das zugehörige Element speichern

Mit der Funktion "Speichern zu Element..." speichern Sie eine globale Bibliothek aus dem Teamcenter-Cache Ordner heraus wieder in das zugehörige Element.

Voraussetzung

- Zwischen dem TIA Portal und Teamcenter besteht eine aktive Verbindung.

Vorgehensweis

Um die globale Bibliothek wieder mit Teamcenter zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Palette "Globale Bibliothek" die entsprechende globale Bibliothek.
2. Wählen Sie auf der globalen Bibliothek, mit der rechten Maustaste, den Befehl "Teamcenter > Speichern zu Element...".
   Der Dialog "Speichern zu Element..." wird geöffnet.
3. Weisen Sie im Feld "Name" einen Namen zu.
4. Weisen Sie im Feld "Element-ID" eine Element-ID zu.
   – Maximal eingebbar sind 128 Zeichen.
5. Geben Sie im Feld "Revision-ID" eine Revision-ID ein.
   – Maximal eingebbar sind 32 Zeichen.

6. Bestätigen Sie Ihre Eingabe mit "OK".
   Die globale Bibliothek wird unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert
7. Klicken auf die Schaltfläche "Abbrechen" bewirkt, dass die Operation "Speichern zu Element..." abgebrochen wird.

Hinweis

Bitte beachten Sie:

Die Felder Element-ID / Revision-ID sind obligatorische Felder. Die Schaltfläche "OK" ist nur dann editierbar, wenn in den Feldern Element-ID / Revision-ID ein Wert angezeigt wird.

Aktionen beim "Speichern zu Element..."

Beim Ausführen der Funktion "Speichern zu Element..." folgende Aktionen ausgeführt:

1. Die globale Bibliothek wird archiviert und unter der angegebenen Element-ID / Revision-ID in Teamcenter gespeichert.
2. Das zugehörige Dataset befindet sich im Status "Ausgecheckt" und ist mit dem folgenden Symbol gekennzeichnet.

   Das können Sie in der Ordneransicht von Teamcenter sehen.
3. Nach Abschluss der Aktionen, wird der Dialog "Die globale Bibliothek wurde erfolgreich nach Teamcenter gespeichert" geöffnet.
   – Bestätigen Sie diesen Dialog durch klicken auf die Schaltfläche "OK".

Ergebnis

Im TIA Portal stehen Ihnen folgende Teamcenter-Operationen zum Arbeiten mit globalen Bibliotheken zur Verfügung:

- Speichern als neues Element...
- Speichern als neue Revision...
- Speichern zu Element...
- Einchecken.
- Auschecken rückgängig machen.
- Export...

5.14 TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen

Voraussetzungen

Im Folgenden wird vorausgesetzt, dass

- eine aktive Verbindung zwischen dem TIA Portal und Teamcenter hergestellt ist.

Vorgehen

Um ein mit Teamcenter verbundenes TIA Portal-Projekt mit dem TIA Portal Teamcenter Gateway zu suchen und zur weiteren Bearbeitung zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektansicht im Menü "Projekt" den Befehl "Projekt aus Teamcenter öffnen...".
   Der Dialog "Projekt aus Teamcenter öffnen" wird geöffnet.
2. Geben Sie im Feld "Projektname" den Projektnamen des TIA Portal-Projekts ein, nach dem Sie suchen möchten.
   Maximal eingebbar sind 128 Zeichen.

   Hinweis

   Suchen nach TIA Portal-Projekt Elementen und deren abhängigen Typen

   Die Suche vergleicht die angegebenen Suchkriterien. Wenn sowohl der Elementtyp eines TIA Portal-Projekts "T4TiaProject“, als auch dessen abhängiger Typ "T4TiaSubProject” definiert ist, wird in der Suchergebnisliste der Elementtyp "T4TiaProject" und der abhängige Typ "T4TiaSubProject" aufgelistet.

3. Optional: Geben Sie im Feld "Element-ID" die Element-ID des TIA Portal-Projekts ein, nach dem Sie suchen möchten.
   Maximal eingebbar sind 94 Zeichen.
4. Optional: Geben Sie im Feld "Elementname" den Elementnamen des TIA Portal-Projekts ein, nach dem Sie suchen möchten.
   Maximal eingebbar sind 94 Zeichen.

5. Optional: Geben Sie im Feld "Revision" die Revision-ID des TIA Portal-Projekts ein, nach dem Sie suchen möchten.
   Maximal eingebbar sind 32 Zeichen.

   Hinweis

   Alle zuvor genannten Felder sind optional. Um die Suche zu starten, muss jedoch mindestens eins der folgenden Suchkriterien definiert sein:
   - Projektname:
   - Element-ID:
   - Elementname:

   Sie können bei der Suche auch Platzhalterzeichen verwenden.

Platzhalterzeichen	Bedeutung
*	Geben Sie einen Stern als Platzhalter für beliebig viele Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Pro" beginnen. Geben Sie also "Pro*" in das Feld ein.
?	Geben Sie ein Fragezeichen als Platzhalter für ein beliebiges Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Proj" beginnen. Geben Sie also "Pro?" in das Feld ein.
;	Geben Sie ein Semikolon als Platzhalter innerhalb eines Suchkriteriums ein. Beispiel: Sie suchen nach Wörtern, die mit "Pro*; auto" beginnen. Geben Sie also "Pro*" ; "auto*" in das Feld ein.
Bei der Suche wird die Groß- und Kleinschreibung nicht berücksichtigt, sodass Sie die Schlüsselwörter in beliebiger Schreibweise angeben können. Beispiel: Die Suche nach "PROjekt1" und "Projekt1" soll identische Suchergebnisse liefern.	

6. Durch klicken auf die Schaltfläche "Suche starten" wird die Suche gestartet.
   Der Dialog "Suche wird ausgeführt..." wird geöffnet. Die Suche erfolgt im Hintergrund und kann nicht abgebrochen werden.
7. Durch klicken auf die Schaltfläche "Zurücksetzen" werden alle eingetragenen Suchkriterien gelöscht und das Suchergebnis zurücksetzen.

Ergebnis

Die in Teamcenter gefundenen TIA Portal-Projekte werden im Dialog "Projekt aus Teamcenter öffnen" angezeigt.

Siehe auch: Öffnen von TIA Portal-Projekten aus dem Suchergebnis (Seite 156).

Vorgehen

Um eine mit Teamcenter verbundene globale Bibliothek mit dem TIA Portal Teamcenter Gateway zu suchen und zur weiteren Bearbeitung zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Palette "Globale Bibliotheken" auf einen freien Bereich und wählen Sie mit der rechten Maustaste Befehl "Globale Bibliothek aus Teamcenter öffnen...".
   Oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Globale Bibliothek aus Teamcenter öffnen...".
   Der Dialog "Globale Bibliothek aus Teamcenter öffnen" wird geöffnet.
2. Geben Sie im Feld "Bibliothek Name" den Namen der globalen Bibliothek ein, nach der Sie suchen möchten.
   Maximal eingebbar sind 128 Zeichen.

   Hinweis

   Suchen nach globalen Bibliothek und deren abhängigen Typen

   Die Suche vergleicht die angegebenen Suchkriterien. Wenn sowohl der Elementtyp einer globalen Bibliothek "T4TiaLibrary“, als auch deren abhängiger Typ "T4TiaSubLibrary” definiert ist, wird in der Suchergebnisliste der Elementtyp "T4TiaLibrary" und der abhängiger Typ "T4TiaSubLibrary" aufgelistet.

3. Optional: Geben Sie im Feld "Element-ID" die Element ID der globalen Bibliothek ein, nach der Sie suchen möchten.
   Maximal eingebbar sind 94 Zeichen.
4. Optional: Geben Sie im Feld "Elementname" den Elementnamen der globalen Bibliothek ein, nach dem Sie suchen möchten.
   Maximal eingebbar sind 94 Zeichen.

5. Optional: Geben Sie im Feld "Revision" die Revision-ID der globalen Bibliothek ein, nach der Sie suchen möchten.
   Maximal eingebbar sind 32 Zeichen.

   Hinweis

   Alle zuvor genannten Felder sind optional. Um die Suche zu starten, muss jedoch mindestens eins der folgenden Suchkriterien definiert sein:
   - Name Bibliothek
   - Element-ID
   - Elementname

   Sie können bei der Suche auch Platzhalterzeichen verwenden.

Platzhalterzeichen	Bedeutung
*	Geben Sie einen Stern als Platzhalter für beliebig viele Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Bib" beginnen. Geben Sie also "Bib*" in das Feld ein.
?	Geben Sie ein Fragezeichen als Platzhalter für ein beliebiges Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Bib" beginnen. Geben Sie also "Bib?" in das Feld ein.
;	Geben Sie ein Semikolon als Platzhalter innerhalb eines Suchkriteriums ein. Beispiel: Sie suchen nach Wörtern, die mit "Bib; auto" beginnen. Geben Sie also "Bib*" ; "auto*" in das Feld ein.
Bei der Suche wird die Groß- und Kleinschreibung nicht berücksichtigt, sodass Sie die Schlüsselwörter in beliebiger Schreibweise angeben können. Beispiel: Die Suche nach "BIiothek1" und "Bibliothek1" soll identische Suchergebnisse liefern.	

6. Durch klicken auf die Schaltfläche "Suche starten" wird die Suche gestartet.
   Der Dialog "Suche wird ausgeführt..." wird geöffnet. Die Suche erfolgt im Hintergrund und kann nicht abgebrochen werden.
7. Durch klicken auf die Schaltfläche "Zurücksetzen" werden alle eingetragenen Suchkriterien gelöscht und das Suchergebnis zurückgesetzt.

Ergebnis

Die in Teamcenter gefundenen globalen Bibliotheken werden im Dialog "Globale Bibliothek aus Teamcenter öffnen" angezeigt.

Siehe auch: Öffnen von globalen Bibliotheken aus dem Suchergebnis (Seite 161).

5.14.1 Öffnen von TIA Portal-Projekten aus dem Suchergebnis

5.14.1.1 Öffnen von TIA Portal-Projekten aus dem Suchergebnis

Vorgehen

Um das TIA Portal-Projekt aus dem Suchergebnis zu öffnen, gehen Sie folgendermaßen vor:

1. Um sich die verfügbaren Revisionen anzeigen zu lassen, klicken Sie dazu auf das Pfeilsymbol, links neben dem entsprechenden TIA Portal-Projekt.

Um die gewünschte Revision des TIA Portal-Projekts zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte Revision des TIA Portal-Projekts.
   Beachten Sie beim "Öffnen von TIA Portal-Projekten aus der Suchergebnisliste" auch das Optionskästchen "Auschecken" (das Optionskästchen "Auschecken" ist standardmäßig aktiviert).
2. Klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

- Das Dataset des TIA Portal-Projekts ist ausgecheckt und in Teamcenter mit dem Symbol gekennzeichnet.
- Die ausgewählte Revision des TIA Portal-Projekts wird im TIA Portal im Modus "schreibend" geöffnet.
- Der Dialog "TIA Portal-Projekt aus Teamcenter öffnen" wird geschlossen.

Hinweise beim Öffnen von TIA Portal-Projekten aus der Suchergebnisliste

1. Öffnen eines TIA Portal-Portal-Projekts aus der Suchergebnisliste **mit** der Option "Auschecken".
   Wenn die zugehörige Revision des TIA Portal-Projekts bereits durch einen anderen Benutzer ausgecheckt ist, erhalten Sie folgende Meldung:
   "Das Projekt ist bereits durch einen anderen Benutzer ausgecheckt: Benutzername. Das Projekt wird schreibgeschützt geöffnet.".
2. Öffnen eines TIA Portal-Portal-Projekts aus der Suchergebnisliste **ohne** die Option "Auschecken",
   Die Revision des TIA Portal-Portal-Projekts wird im Modus "nur lesend" geöffnet.
3. Öffnen eines TIA Portal-Portal-Projekts aus der Suchergebnisliste (**mit** oder **ohne** die Option "Auschecken").
   Wenn die zugehörige Revision des TIA Portal-Projekts bereits durch Sie geöffnet ist, erhalten Sie folgende Meldung: "Das TIA Portal-Projekt das Sie versuchen zu öffnen, ist bereits geöffnet.".

Öffnen von TIA Portal-Projekten über die Suchergebnisliste

Hinweis

Beachten Sie Folgendes:

Ein TIA Portal-Projekt, das im TIA Portal zur gleichen Zeit als Referenzprojekt geöffnet ist, sollte nicht über den Dialog "Öffnen von TIA Portal-Projekten aus dem Suchergebnis" mit der Option "Auschecken" geöffnet werden.

Öffnen von älteren TIA Portal-Projekten über die Suchergebnisliste

Hinweis

Beachten Sie Folgendes:

Beim Öffnen von älteren TIA Portal-Projekten über die Suchergebnisliste wird ein Dialog angezeigt, der Sie darauf hinweist, dass Sie das TIA Portal-Projekt aktualisieren können.

1. Durch klicken auf die Schaltfläche "OK" wird das aktualisierte TIA Portal-Projekt in das ursprüngliche TIA Portal-Projekt umbenannt, indem die vorherige Version im Teamcenter Cache überschrieben wird.
2. Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.

Verhalten beim Öffnen von TIA Portal-Projekten über die Suchergebnisliste oder aus dem Teamcenter Cache

Hinweis

TIA Portal-Projekte, die Sie über die Suchergebnisliste, oder aus dem Teamcenter Cache öffnen, werden abhängig vom aktivieren/deaktivieren des Optionskästchens "Auschecken" im Modus "schreiben" oder "nur lesen" geöffnet.

Bei TIA Portal-Projekten, die Sie über die Suchergebnisliste von Teamcenter öffnen, ist der Einchecken/Auschecken Status mit Teamcenter synchronisiert.

Dabei wird angenommen, dass das TIA Portal-Projekt vom aktuellen Benutzer ausgecheckt ist. Diese Annahme ist unabhängig vom Einchecken/Auschecken Status des TIA Portal-Projekts auf dem Teamcenter Server.

Siehe auch

TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen (Seite 152)

Aufbau des Registers "Eigenschaften" im Dialog "TIA Portal-Projekt aus Teamcenter öffnen" (Seite 158)

5.14.1.2 Aufbau des Registers "Eigenschaften" im Dialog "TIA Portal-Projekt aus Teamcenter öffnen"

Aufbau

Das Register "Eigenschaften" befindet sich im unteren Bereich des Dialogs "Projekt aus Teamcenter öffnen". Um das Register "Eigenschaften" anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Suchergebnisliste das entsprechende TIA Portal-Projekt.
2. Klicken Sie im unteren Bereich des Suchergebnisses innerhalb des Fensterteilers auf den Pfeil-nach-oben oder Pfeil-nach-unten, um die angrenzenden Teile ein- und auszublenden.
3. Klicken Sie auf das Register "Eigenschaften", um die Eigenschaften des ausgewählten TIA Portal-Projekts anzuzeigen. Die Eigenschaften werden tabellarisch dargestellt.

Aufbau des Registers "Eigenschaften" im Dialog "Projekt aus Teamcenter öffnen"

Eigenschaften Teamcenter Element

Die folgende Tabelle zeigt die Eigenschaften des ausgewählten Elements an:

Eigenschaft	Beschreibung
	Wert
Typ	Abhängig von der ausgewählten Element-Revision werden unterschiedliche Informationen angezeigt.
Eigentümer	"Eigentümer" des Elements (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx).
Zugriff	"Lesend" oder "Schreibend" abhängig von den Zugriffsrechten des aktuellen Benutzers.
Letzte Änderung	"Letzte Änderung" zeigt den Zeitpunkt an, an dem das Element zuletzt geändert wurde.
Status	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Element ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.
Beschreibung	Zeigt den Beschreibungstext (definiert durch das T4tia_Beschreibungs_Attribut) des Elements an, der durch das Speichern des Projekts oder der globalen Bibliothek als neues Element nach Teamcenter erstellt wurde.

Eigenschaften Teamcenter Dataset

In Register "Eigenschaften" werden die Eigenschaften des ausgewählten Datasets angezeigt.

Die folgende Tabelle zeigt die Eigenschaften des ausgewählten Datasets:

Eigenschaft	Beschreibung	
	Wert (Revision)	Wert (Dataset)
Typ	Abhängig von der ausgewählten Element-Revision werden unterschiedliche Informationen angezeigt.	Abhängig von dem ausgewählten Datasettyp werden unterschiedliche Informationen angezeigt.
Eigentümer	"Eigentümer" des Elements (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx).	"Eigentümer" des Dataset (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx). Hinweis: Der Eigentümer des Dataset und der Element-Revision, muss für die nach Teamcenter zu speichernden TIA Projekte und globalen Bibliotheken identisch sein.
Zugriff	"Lesen" oder "Schreiben" abhängig von den Zugriffsrechten des aktuellen Benutzers.	"Lesen" oder "Schreiben" abhängig von den Zugriffsrechten des aktuellen Benutzers.
Letzte Änderung (Änderungsdatum)	"Änderungsdatum" zeigt den Zeitpunkt an, an dem das Element zuletzt geändert wurde.	"Änderungsdatum" zeigt den Zeitpunkt an, an dem das Dataset zuletzt geändert wurde.
Status	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Element ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Dataset ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.

Hinweis

Beachten Sie, dass

- die "Eigenschaften" schreibgeschützt angezeigt werden und von Ihnen nicht geändert werden können.

Bitte beachten Sie:

- wenn Sie in der Suchergebnisansicht nichts ausgewählt haben,
- wenn die Suchergebnisansicht leer ist,
- wenn Sie auf die Schaltfläche “Rücksetzen” geklickt haben,
  im Register “Eigenschaften” die folgende Information angezeigt wird:
  Es sind keine 'Eigenschaften' verfügbar.
  Momentan können keine 'Eigenschaften' angezeigt werden. Es ist kein Objekt ausgewählt.

Siehe auch

TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen (Seite 152)

Dataset Status im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" anzeigen (Seite 165)

5.14.2 Öffnen von TIA Portal-Projekten aus der TIA Portal-Projektliste

Öffnen von TIA-Projekten aus der Liste der zuletzt verwendeten Projekte

Wenn Sie ein TIA Portal-Projekt in Teamcenter gespeichert haben und dieses anschließend im TIA Portal aus der Liste der zuletzt verwendeten Projekte öffnen, sind die Teamcenter-Operationen "Speichern" und "Auschecken" deaktiviert.

Ergebnis

Das TIA Portal-Projekt wird im Modus "nur lesend" geöffnet.

Im TIA Portal stehen Ihnen nach dem Öffnen eines TIA Portal-Projekts aus der Liste der zuletzt verwendeten Projekte folgende Teamcenter-Operationen für das Arbeiten mit dem TIA Portal-Projekt zur Verfügung:

- Speichern als neues Element....

Abhilfe

1. Schließen Sie das aus der Liste der zuletzt verwendeten Projekte geöffnete TIA Portal-Projekt.
2. Öffnen Sie das TIA Portal-Projekt erneut aus Teamcenter.
3. Wählen Sie dazu im Menü "Projekt" den Befehl "Teamcenter Projekt öffnen..."
   Der Dialog "Projekt aus Teamcenter öffnen" wird geöffnet.
4. Geben Sie im Feld "Projektname" den Projektnamen des TIA-Portal-Projekts ein.
5. Klicken Sie auf die Schaltfläche "Suche starten".
   Der Dialog "Suche wird ausgeführt..." wird geöffnet. Die Suche wird im Hintergrund ausgeführt und kann nicht abgebrochen werden.
6. Öffnen Sie anschließend das TIA Portal-Projekt aus der Suchergebnisliste.
   Siehe auch: Öffnen von TIA Portal-Projekten aus dem Suchergebnis (Seite 156).

5.14.3 Öffnen von globalen Bibliotheken aus dem Suchergebnis

5.14.3.1 Öffnen von globalen Bibliotheken aus dem Suchergebnis

Vorgehen

Um globale Bibliotheken aus dem Suchergebnis zu öffnen, gehen Sie folgendermaßen vor:

1. Um sich die verfügbaren Revisionen anzeigen zu lassen, klicken Sie auf das Pfeilsymbol, links neben der entsprechenden globalen Bibliothek.

Um die gewünschte Revision der globalen Bibliothek zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte Revision der globalen Bibliothek.
   Beachten Sie beim "Öffnen von globalen Bibliotheken aus der Suchergebnisliste" auch das Optionskästchen "Auschecken" (das Optionskästchen "Auschecken" ist standardmäßig aktiviert).
2. Klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

- Das Dataset der globalen Bibliothek ist ausgecheckt und in Teamcenter mit dem Symbol gekennzeichnet.
- Die ausgewählte Revision der globalen Bibliothek wird im TIA Portal im Modus "schreibend" geöffnet.
- Der Dialog "Globale Bibliothek aus Teamcenter öffnen" wird geschlossen.

Hinweise beim Öffnen von globalen Bibliotheken aus der Suchergebnisliste

1. Öffnen einer globalen Bibliothek aus der Suchergebnisliste **mit** der Option "Auschecken"
   Wenn die zugehörige Revision der globalen Bibliothek bereits durch einen anderen Benutzer ausgecheckt ist, erhalten Sie die Meldung:
   "Die Bibliothek ist bereits durch einen anderen Benutzer ausgecheckt: Benutzername. Die Bibliothek wird schreibgeschützt geöffnet.".
2. Öffnen einer globalen Bibliothek aus der Suchergebnisliste **ohne** die Option "Auschecken".
   Die Revision der globalen Bibliothek wird im Modus "nur lesend" geöffnet.
3. Öffnen einer globalen Bibliothek aus der Suchergebnisliste (**mit** oder **ohne** die Option "Auschecken").
   Wenn die zugehörige Revision der globalen Bibliothek bereits durch Sie geöffnet ist, erhalten Sie folgende Meldung: "Die globalen Bibliothek die Sie versuchen zu öffnen, ist bereits geöffnet.".

Öffnen von älteren globalen Bibliotheken über die Suchergebnisliste

Hinweis

Beachten Sie Folgendes:

Beim Öffnen von älteren globalen Bibliotheken über die Suchergebnisliste wird ein Dialog angezeigt, der Sie darauf hinweist, dass Sie die globale Bibliothek aktualisieren können.

1. Durch klicken auf die Schaltfläche "OK" wird die aktualisierte globalen Bibliothek in die ursprüngliche globalen Bibliothek umbenannt, indem die vorherige Version im Teamcenter Cache überschrieben wird
2. Klicken auf die Schaltfläche "Abbruch" bewirkt, dass dieser Vorgang abgebrochen wird.

Verhalten beim Öffnen von globalen Bibliotheken über die Suchergebnisliste oder aus dem Teamcenter Cache

Hinweis

Globale Bibliotheken, die Sie über die Suchergebnisliste oder aus dem Teamcenter Cache öffnen, werden abhängig vom aktivieren/deaktivieren des Optionskästchens "Auschecken" im Modus "schreiben" oder "nur lesen" geöffnet.

Bei globalen Bibliotheken, die Sie über die Suchergebnisliste von Teamcenter öffnen, ist der Einchecken/Auschecken Status mit Teamcenter synchronisiert.

Dabei wird angenommen, dass die globale Bibliothek vom aktuellen Benutzer ausgecheckt ist. Diese Annahme ist unabhängig vom Einchecken/Auschecken Status der globalen Bibliothek auf dem Teamcenter Server.

Siehe auch

TIA Portal-Projekte bzw. globale Bibliotheken mit dem TIA Portal Teamcenter Gateway suchen und öffnen (Seite 152)

Aufbau des Registers "Eigenschaften" im Dialog "globale Bibliothek aus Teamcenter öffnen" (Seite 163)

5.14.3.2 Aufbau der Register "Eigenschaften" im Dialog "globale Bibliothek aus Teamcenter öffnen"

Aufbau des Registers "Eigenschaften" im Dialog "globale Bibliothek aus Teamcenter öffnen"

Aufbau

Das Register "Eigenschaften" befindet sich im unteren Bereich des Dialogs "Globale Bibliothek aus Teamcenter öffnen". Um das Register "Eigenschaften" anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Suchergebnisliste die entsprechende globale Bibliothek.
2. Klicken Sie im unteren Bereich des Suchergebnisses innerhalb des Fensterteilers auf den Pfeil-nach-oben oder Pfeil-nach-unten, um die angrenzenden Teile ein- und auszublenden.
3. Klicken Sie auf das Register "Eigenschaften", um die Eigenschaften der ausgewählten globalen Bibliothek anzuzeigen. Die Eigenschaften werden tabellarisch dargestellt.

Aufbau des Registers "Eigenschaften" im Dialog "Globale Bibliothek aus Teamcenter öffnen"

Eigenschaften Teamcenter Element

Die folgende Tabelle zeigt die Eigenschaften des ausgewählten Elements an:

Eigenschaft	Beschreibung
	Wert
Typ	Abhängig von der ausgewählten Element-Revision werden unterschiedliche Informationen angezeigt.
Eigentümer	"Eigentümer" des Elements (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx).
Zugriff	"Lesend" oder "Schreibend" abhängig von den Zugriffsrechten des aktuellen Benutzers.
Letzte Änderung	"Letzte Änderung" zeigt den Zeitpunkt an, an dem das Element zuletzt geändert wurde.
Status	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Element ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.
Beschreibung	Zeigt den Beschreibungstext (definiert durch das T4tia_Beschreibungs_Attribut) des Elements an, der durch das Speichern des Projekts oder der globalen Bibliothek als neues Element nach Teamcenter erstellt wurde.

Eigenschaften Teamcenter Dataset

In Register "Eigenschaften" werden die Eigenschaften des ausgewählten Datasets angezeigt.

Die folgende Tabelle zeigt die Eigenschaften des ausgewählten Datasets:

Eigenschaft	Beschreibung	
	Wert (Revision)	Wert (Dataset)
Typ	Abhängig von der ausgewählten Element-Revision werden unterschiedliche Informationen angezeigt.	Abhängig von dem ausgewählten Datasettyp werden unterschiedliche Informationen angezeigt.
Eigentümer	"Eigentümer" des Elements (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx).	"Eigentümer" des Dataset (z.B. Benutzername und aktive Kennung, z.B. Name, Vorname, z002vdyx). Hinweis: Der Eigentümer des Dataset und der Element-Revision, muss für die nach Teamcenter zu speichernden TIA Projekte und globalen Bibliotheken identisch sein.
Zugriff	"Lesen" oder "Schreiben" abhängig von den Zugriffsrechten des aktuellen Benutzers.	"Lesen" oder "Schreiben" abhängig von den Zugriffsrechten des aktuellen Benutzers.
Letzte Änderung (Änderungsdatum)	"Änderungsdatum" zeigt den Zeitpunkt an, an dem das Element zuletzt geändert wurde.	"Änderungsdatum" zeigt den Zeitpunkt an, an dem das Dataset zuletzt geändert wurde.
Status	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Element ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.	"Ausgecheckt von (<Benutzer-ID>)" - zeigt den Benutzer und die Benutzer-ID an, durch den das Dataset ausgecheckt wurde. "Released" - das Element hat den Status "Released". "Kein" - das Element nimmt weder den Status "Ausgecheckt" noch den Status "Released" ein.

Hinweis

Beachten Sie, dass

- die "Eigenschaften" schreibgeschützt angezeigt werden und von Ihnen nicht geändert werden können.

Bitte beachten Sie:

- wenn Sie in der Suchergebnisansicht nichts ausgewählt haben,
- wenn die Suchergebnisansicht leer ist,
- wenn Sie auf die Schaltfläche “Rücksetzen” geklickt haben,
  im Register “Eigenschaften” die folgende Information angezeigt wird:
  Es sind keine 'Eigenschaften' verfügbar.
  Momentan können keine 'Eigenschaften' angezeigt werden. Es ist kein Objekt ausgewählt.

5.14.4 Dataset Status im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" anzeigen

5.14.4.1 Anzeigen des Dataset Status

Dataset Status im Suchergebnis anzeigen

Die Suchergebnisse werden im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" tabellarisch dargestellt. Die folgende Tabelle zeigt, in welchem 'Dataset Status' sich das ausgewählte Dataset befindet.

Symbol	Bedeutung
	Durch den aktuellen Benutzer ausgecheckt.
	Durch einen anderen Benutzer ausgecheckt.
	Das Dataset befindet sich im Status "Released"

Hinweis

Bitte beachten Sie, dass der angezeigte Status nicht automatisch aktualisiert wird. Um den Status zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Suchergebnis auf das Pfeilsymbol, links neben dem entsprechenden Objekt, um das Objekt zu schließen und wieder zu öffnen.
   Das Schließen und Öffnen des Objekts bewirkt, dass die aktuellen Statusinformationen angezeigt werden.

5.14.4.2 Anzeigen des Element-Revision Status

Element-Revision Status im Suchergebnis anzeigen

Die Suchergebnisse werden im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" tabellarisch dargestellt. Die folgende Tabelle zeigt, in welchem 'Element-Revision Status' sich das ausgewählte Dataset befindet.

Symbol	Bedeutung
	Durch den aktuellen Benutzer ausgecheckt.
	Durch einen anderen Benutzer ausgecheckt.
	Das Dataset befindet sich im Status "Released"

Hinweis

Bitte beachten Sie, dass der angezeigte Status nicht automatisch aktualisiert wird. Um den Status zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Suchergebnis auf das Pfeilsymbol, links neben dem entsprechenden Objekt, um das Objekt zu schließen und wieder zu öffnen.
   Das Schließen und Öffnen des Objekts bewirkt, dass die aktuellen Statusinformationen angezeigt werden.

5.14.4.3 Anzeigen des Cache Status

TC Cache Status im Suchergebnis anzeigen

Die Suchergebnisse werden im Dialog "Projekt aus Teamcenter öffnen" bzw. "Globale Bibliothek aus Teamcenter öffnen" tabellarisch dargestellt. Die folgende Tabelle zeigt, in welchem 'TC Cache Status' sich das ausgewählte Dataset befindet.

Symbol	Bedeutung
	Das Dataset ist mit Teamcenter synchronisiert.
	Das Dataset wurde im Cache verändert.
	Veraltete Dataset-Version im Cache.

Hinweis

Bitte beachten Sie, dass der angezeigte Status nicht automatisch aktualisiert wird. Um den Status zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Suchergebnis auf das Pfeilsymbol, links neben dem entsprechenden Objekt, um das Objekt zu schließen und wieder zu öffnen.
   Das Schließen und Öffnen des Objekts bewirkt, dass die aktuellen Statusinformationen angezeigt werden.

5.15 Manueller Vergleich von zwei TIA Portal-Projekten (Dataset Versionen) mit derselben Revision

So vergleichen Sie zwei TIA Portal-Projekte (Dataset Versionen) derselben Revision miteinander

Um zwei TIA Portal-Projekte (Dataset Versionen) derselben Revision miteinander zu vergleichen, gehen Sie folgendermaßen vor:

1. Exportieren Sie mit "RAC" das Dataset "Version 1" des entsprechenden TIA Portal-Projekts auf ein lokales Verzeichnis.
2. Entpacken Sie die ZIP-Datei.

   Hinweis

   Achten Sie darauf, dass Sie die Datei nicht in den TIA Portal Teamcenter Gateway cache path speichern.

3. Öffnen Sie im TIA Portal das Dataset "Version 1" als Referenzprojekt.
4. Öffnen Sie im TIA Portal das Dataset "Version 2" derselben Revision. Verwenden Sie dazu, in der Projektansicht im Menü "Projekt" den Befehl "Projekt aus Teamcenter öffnen..." ("Öffnen und Suchen").

5.16 Symbole

5.16.1 Objektstatus im Dialog Projekt aus Teamcenter öffnen

Symbole im Dialog "Projekt aus Teamcenter öffnen"

Spalte "Dataset Status"

Abhängig vom Status des Datasets werden in der Spalte "Datasets Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "Ausgecheckt durch aktuellen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt durch den aktuellen Benutzer ausgecheckt ist.
	Status "Ausgecheckt durch anderen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt durch einen anderen Benutzer ausgecheckt ist.
	Status "TCM Released"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt den Status "RELEASED" erhalten hat.

Spalte "Element-Revision Status"

Abhängig vom Status des Elements werden in der Spalte "Element-Revision Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "Ausgecheckt durch aktuellen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt durch den aktuellen Benutzer ausgecheckt ist.
	Status "Ausgecheckt durch anderen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt durch einen anderen Benutzer ausgecheckt ist.
	Status "TCM Released"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt den Status "RELEASED" erhalten hat.

Spalte "Cache Status"

Abhängig vom Status des TC cache werden in der Spalte "Cache Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "In Sync"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt mit Teamcenter synchronisiert ist.
	Status "Locally modified"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt im lokalen Cache von Teamcenter geändert ist.
	Status "Stale Dataset"	Das Symbol kennzeichnet hinter einem Objekt, dass sich eine veraltete Dataset-Version des Objekts im Cache befindet.

5.16.2 Objektstatus im Dialog Globale Bibliothek aus Teamcenter öffnen

Symbole im Dialog "Globale Bibliothek aus Teamcenter öffnen"

Spalte "Dataset Status"

Abhängig vom Status des Elements werden in der Spalte "Dataset Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "Ausgecheckt durch aktuellen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt ausgecheckt ist.
	Status "Ausgecheckt durch anderen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt von einem anderen Benutzer ausgecheckt ist.
	Status "TCM Released"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt den Status "RELEASED" erhalten hat.

Spalte "Element-Revision Status"

Abhängig vom Status des Elements werden in der Spalte "Element-Revision Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "Ausgecheckt durch aktuellen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt ausgecheckt ist.
	Status "Ausgecheckt durch anderen Benutzer"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt von einem anderen Benutzer ausgecheckt ist.
	Status "TCM Released"	Das Symbol kennzeichnet hinter einem Objekt, dass das Objekt den Status "RELEASED" erhalten hat.

Spalte "Cache Status"

Abhängig vom Status des Elements werden in der Spalte "Cache Status" folgende Symbole angezeigt:

Symbol	Bezeichnung	Beschreibung
	Status "In Sync"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt mit Teamcenter synchronisiert ist.
	Status "Locally modified"	Das Symbol kennzeichnet hinter einem Objekt, dass ein Objekt im lokalen Cache von Teamcenter geändert ist.
	Status "Stale Dataset"	Das Symbol kennzeichnet hinter einem Objekt, dass sich eine veraltete Dataset-Version des Objekts im Cache befindet.

5.16.3 Symbole die den Verbindungsstatus zwischen dem TIA Portal und Teamcenter anzeigen

Verbindungsstatus anzeigen

Nachfolgend sind die Symbole, die den Verbindungsstatus zwischen dem TIA Portal und Teamcenter anzeigen aufgelistet.

Symbol	Beschreibung
	Dieses Symbol zeigt, dass zwischen dem TIA Portal und Teamcenter **eine** Verbindung besteht.
	Dieses Symbol zeigt, dass zwischen dem TIA Portal und Teamcenter **keine** Verbindung besteht.
	Dieses Symbol zeigt an, dass der Teamcenter-Server nicht für UTF-8 aktiviert ist.

Index

A

Archivieren von Projekten, 35, 36

D

Dearchivieren von Projekten, 38

K

Kompatibilität
    Projekte aus früheren Programmversionen, 25
    Projekte mit Funktionsrechten, 25
    Rückwärtskompatiblität von Projekten, 25

M

Minimieren von Projekten, 35
Multiuser Engineering
    Einführung, 47

P

Projekt
    Eigenschaften anzeigen, 16
    entfernen, 14
    erstellen, 9
    löschen, 15
    schließen, 13
    speichern, 12
Projekt öffnen, 10
Projektarchive, 35
Protokolle
    Protokolle anzeigen, 18
    Protokolle löschen, 18

R

Referenzprojekt
    Grundlagen, 39
    öffnen, 40
    schließen, 40
    vergleichen, 41

SIEMENS

Projektdaten bearbeiten

Programmier- und Bedienhandbuch

Projektdaten übersetzen und laden 1

Projektdaten vergleichen 2

Projektdaten schützen 3

Projektinhalte drucken 4

Mit mehrsprachigen Projekten arbeiten 5

Mit Textlisten arbeiten 6

Memory Cards verwenden 7

Querverweise nutzen 8

Anwenderdefinierte Dokumentation bereitstellen 9

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Projektdaten übersetzen und laden....7

1.1 Projektdaten übersetzen....7
1.1.1 Allgemeine Informationen zum Übersetzen von Projektdaten....7
1.1.2 Projektdaten übersetzen....8

1.2 Projektdaten laden....9
1.2.1 Allgemeine Informationen zum Laden....9
1.2.2 Projektdaten in ein Gerät laden....11
1.2.3 Projektdaten auf eine Memory Card laden....13
1.2.4 Projektdaten von einem Gerät laden....15
1.2.5 Projektdaten von einer Memory Card laden....17

2 Projektdaten vergleichen....19

2.1 Grundlagen zum Projektdatenvergleich....19

2.2 Offline/Online-Vergleich durchführen....23

2.3 Offline/Offline-Vergleich durchführen....24

2.4 Vergleichseditor verwenden....26
2.4.1 Übersicht über den Vergleichseditor....26
2.4.2 Tabellenspalten ein- und ausblenden....32
2.4.3 Ansicht im Vergleichseditor filtern....33
2.4.4 Detailvergleich durchführen....35
2.4.5 Vergleichsergebnisse aktualisieren....36
2.4.6 Vergleichskriterien festlegen....37
2.4.7 Nicht identische Objekte synchronisieren....38
2.4.7.1 Aktionen festlegen....38
2.4.7.2 Objekte synchronisieren....39
2.4.8 Ansicht ändern....40

3 Projektdaten schützen....43

3.1 Schutzkonzept für Projektdaten....43

3.2 Zugriffsrechte für Geräte zurücknehmen....45

4 Projektinhalte drucken....47

4.1 Projektdokumentation drucken....47
4.1.1 Dokumentationsfunktion....47
4.1.2 Ausdruck von Projektinhalten....48
4.1.3 Einstellungen für den Druck ändern....49
4.1.4 Festlegen des Drucklayouts....50
4.1.5 Dokument-Informationen eingeben....51
4.1.6 Deckblätter und Rahmen verwalten....52
4.1.6.1 Einsetzen von Deckblättern und Rahmen....52
4.1.6.2 Rahmen anlegen....53
4.1.6.3 Deckblatt anlegen....54
4.1.6.4 Vorgefertigte Rahmen und Deckblätter nutzen....54

4.1.7 Deckblätter und Rahmen gestalten....55
4.1.7.1 Deckblätter und Rahmen bearbeiten....55
4.1.7.2 Allgemeine Bedienung des Dokumentationseditors....56
4.1.7.3 Druckfläche festlegen....59
4.1.7.4 Platzhalter für Meta-Daten einfügen....60
4.1.8 Druckvorschau anzeigen....61
4.1.8.1 Druckvorschau erstellen....61
4.1.8.2 Bedienung innerhalb der Druckvorschau....63
4.1.9 Projektdaten drucken....64

4.2 Modulbeschriftungsstreifen drucken....67
4.2.1 Druckfunktion für Modulbeschriftungsstreifen....67
4.2.2 Beschriftungsstreifen drucken....69
4.2.3 Beschriftungsdaten als XML exportieren....70
4.2.4 XML-Schema der Exportdatei....71
4.2.5 Verschiebung des Druckbereichs ermitteln....74

5 Mit mehrsprachigen Projekten arbeiten....77

5.1 Grundlagen zu Projekttexten....77

5.2 Kategorien von Anwendertexten....80

5.3 Projektsprachen auswählen....81

5.4 Editiersprache festlegen....82

5.5 Alle Projekttexte tabellarisch übersetzen....83

5.6 Projekttexte übersetzen....84

5.7 Projekttexte mithilfe von Referenztexten übersetzen....85

5.8 Projekttexte exportieren....86

5.9 Projekttexte importieren....89

5.10 Anwendungsbeispiele für mehrsprachige Projekte....90

6 Mit Textlisten arbeiten....93

6.1 Textlisten....93

6.2 Anwenderdefinierte Textlisten anlegen....95

6.3 Anwenderdefinierte Textlisten bearbeiten....96

6.4 Systemdefinierte Textlisten bearbeiten....97

6.5 Textlisten exportieren....98

6.6 Textlisten importieren....99

7 Memory Cards verwenden....101

7.1 Grundlagen zu Memory Cards....101

7.2 Benutzerdefinierten Kartenleser hinzufügen....102

7.3 Auf Memory Cards zugreifen....103

7.4 Eigenschaften von Memory Cards anzeigen....104

7.5 Daten auf der Memory Card eines Controllers anzeigen....105

8 Querverweise nutzen....107
8.1 Querverweise nutzen....107
9 Anwenderdefinierte Dokumentation bereitstellen....109
9.1 Anwenderdefinierte Dokumentation verwenden....109
9.2 Einstellungen im TIA Portal festlegen....112
9.3 Einstellungen mit einer XML-Datei festlegen....113
9.4 Einstiegsseite erstellen....116
9.5 Konventionen für das Erstellen....117
9.6 Anwenderdefinierte Dokumentation aufrufen....120
9.7 Aufrufprotokoll anzeigen....121
9.8 Anwenderdefinierte Dokumentation erstellen....122
Index....125

Projektdaten übersetzen und laden 1

1.1 Projektdaten übersetzen

1.1.1 Allgemeine Informationen zum Übersetzen von Projektdaten

Übersetzen von Projektdaten

Beim Übersetzen werden die Projektdaten so umgewandelt, dass sie vom Gerät lesbar sind. Hardware-Konfigurationsdaten und Programmdaten können getrennt oder gemeinsam übersetzt werden. Sie können die Projektdaten gleichzeitig für ein oder mehrere Zielsysteme übersetzen.

Folgende Projektdaten müssen vor dem Laden übersetzt werden:

- Hardware-Projektdaten, z. B. Konfigurationsdaten der Geräte oder Netze und Verbindungen
- Software-Projektdaten, z. B. Programmbausteine oder Prozessbilder

---

Hinweis

Während ein Gerät übersetzt wird, kann kein weiterer Übersetzungsvorgang gestartet werden. Beachten Sie dabei, dass ein Übersetzungsvorgang nicht nur manuell durch Sie, sondern bei Bedien- und Beobachtungsgeräten auch automatisch angestoßen werden kann.

Umfang der Übersetzung

Wenn Sie Projektdaten übersetzen, haben Sie abhängig vom Gerät folgende Möglichkeiten:

- Hardware und Software (nur Änderungen)
- Hardware (nur Änderungen)
- Hardware (komplett übersetzen)
- Software (nur Änderungen)
- Software (komplett übersetzen)
- Software (Speicherreserve zurücksetzen)

Siehe auch

Projektdaten übersetzen (Seite 8)

1.1.2 Projektdaten übersetzen

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Übersetzens von Projektdaten innerhalb der Projektnavigation. Wie das Übersetzen bestimmter Objekte funktioniert und welche Besonderheiten es dabei zu beachten gilt, finden Sie in der Hilfe zum Produkt.

Vorgehen

Um Projektdaten zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Geräte, deren Projektdaten Sie übersetzen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

   Hinweis

   Beachten Sie, dass es vom selektierten Gerät abhängt, welche Optionen Ihnen zur Verfügung stehen.

   Die Projektdaten werden übersetzt. Ob das Übersetzen erfolgreich durchgeführt wurde, können Sie im Inspektorfenster unter "Info > Übersetzen" kontrollieren.

Siehe auch

Allgemeine Informationen zum Übersetzen von Projektdaten (Seite 7)

1.2 Projektdaten laden

1.2.1 Allgemeine Informationen zum Laden

Einführung

Für das Einrichten Ihres Automatisierungssystems ist es notwendig, dass Sie die Projektdaten, die Sie offline erzeugt haben, in die angeschlossenen Geräte laden. Diese Projektdaten entstehen z. B. bei der Konfiguration von Hardware, Netzen und Verbindungen, bei der Programmierung des Anwenderprogramms oder beim Anlegen von Rezepturen.

Beim erstmaligen Laden werden die Projektdaten vollständig geladen. Bei weiteren Ladevorgängen werden nur noch Änderungen geladen.

Sie können die Projektdaten in Geräte und auf Memory Cards laden.

Hinweis

Ab TIA Portal V15 werden mit einer CPU S7-1500 ab Firmwarestand V2.5 beim Laden in ein Gerät oder auf eine Memory Card und beim Laden von einem Gerät oder von einer Memory Card auch Variablentabellen mitgeladen. Somit sind die Variablentabellen auch online auf Ihrer CPU verfügbar. Die PLC-Variablen werden entsprechend den offline konfigurierten Variablentabellen abgelegt. Es kann ein Online-/Offline-Vergleich durchgeführt werden und Variablentabellen können vom Gerät geladen werden.

Hinweis

Während ein Gerät übersetzt wird, kann kein Ladevorgang gestartet werden. Beachten Sie dabei, dass ein Übersetzungsvorgang nicht nur manuell durch Sie, sondern bei Bedien- und Beobachtungsgeräten auch automatisch angestoßen werden kann.

Mögliche Optionen für das Laden

Abhängig vom Objekt, das Sie laden möchten, haben Sie folgende Optionen:

- Hardware und Software (nur Änderungen)
  Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen.
- Hardwarekonfiguration
  Es wird nur die Hardware-Konfiguration in das Ziel geladen.
- Software (nur Änderungen)
  Es werden nur die Objekte in das Ziel geladen, die online und offline unterschiedlich sind.
- Software (komplett laden)
  Es wird das PLC-Programm inklusive aller Bausteine, PLC-Datentypen und PLC-Variablen in das Ziel geladen und alle Werte auf ihre Startwerte zurückgesetzt. Beachten Sie, dass dies auch für die remanenten Werte der Fall ist.

Sie können aber auch Projektdaten, die sich bereits in einem Gerät befinden, wieder in Ihr Projekt laden. Dabei haben Sie folgende Möglichkeiten:

- Komplettes Gerät als neue Station laden
  Die Hardware-Konfiguration des Geräts und die auf dem Gerät vorhandene Software werden in das Projekt geladen.
  Das Gerät wird mit allen relevanten Daten in das Projekt geladen.
- Software eines Geräts laden
  Es werden nur die Bausteine und die Parameter aus dem Gerät in eine vorhandene CPU im Projekt geladen.

In beiden Fällen werden beim Laden alle Instanzen von Bibliothekstypen wieder mit der passenden Version des Typen in der Projektbibliothek verbunden. Wenn zu einer geladenen Instanz noch kein passender Typ oder nicht die richtige Version des Typen in der Projektbibliothek vorhanden ist, wird der Typ oder die Version in der Projektbibliothek ergänzt.

Laden mit Synchronisierung

Im Rahmen des Team Engineering ist es möglich, dass mehrere Anwender mit mehreren Engineering Systemen parallel an einem Projekt arbeiten und auf eine CPU S7-1500 zugreifen. Um die Konsistenz innerhalb des gemeinsamen bearbeiteten Projektes sicher zu stellen, ist es notwendig, die jeweils geänderten Daten vor dem Laden zu synchronisieren, damit es nicht zu ungewollten Überschreibungen kommt.

Wenn also beim Laden Unterschiede zwischen der Online- und der Offline-Datenhaltung innerhalb des gemeinsamen Projektes festgestellt werden, die von einem anderen Engineering System verursacht wurden, wird beim Ladevorgang automatisch eine Synchronisierung der zu ladenden Daten angeboten.

In diesem Fall werden im Dialog "Synchronisierung" die zu synchronisierenden Daten mit dem aktuellen Status (Online-Offline-Vergleich) und den möglichen Aktionen angezeigt.

Anwendungsfall	Empfehlung	Synchronisierung
Ein oder mehrere Bausteine sind auf der CPU (online) neuer als im Engineering System (offline).	Diese Bausteine sollten vor dem Laden zuerst von der CPU in das Engineering System geladen werden.	Automatische Synchronisierung ist möglich: Die Bausteine im Engineering System werden vor dem Laden aktualisiert.
Ein oder mehrere Bausteine sind neu erstellt und nur in der CPU (online) vorhanden.	Diese Bausteine sollten vor dem Laden zuerst von der CPU in das Engineering System geladen werden.	Automatische Synchronisierung ist möglich: Die neuen Bausteine werden vor dem Laden im Engineering System hinzugefügt.
Ein oder mehrere Bausteine auf der CPU wurden gelöscht.	Diese Bausteine sollten vor dem Laden im Engineering System ebenfalls gelöscht werden.	Automatische Synchronisierung ist nicht möglich. Die auf der CPU gelöschten Bausteine sollten im Offline-Projekt im Engineering System manuell gelöscht werden.

Anwendungsfall	Empfehlung	Synchronisierung
Ein oder mehrere Bausteine sind auf der CPU und im Engineering System unterschiedlich. Dieser Fall tritt dann auf, wenn ein anderer Anwender Bausteine verändert und bereits in die CPU geladen hat, an denen auch Sie Korrekturen vorgenommen haben.	Diese Bausteine mit konkurrierenden Änderungen müssen manuell angepasst werden. Entscheiden Sie in diesem Fall, welche Änderungen Sie übernehmen wollen. Sollen die Bausteine auf der CPU erhalten bleiben, so sollten Sie diese Bausteine vor dem Laden von der CPU in Ihr Engineering System übernehmen. Sollen die von Ihnen geänderten Bausteine übernommen werden, können Sie mit dem Laden ohne Synchronisierung fortfahren.	Automatische Synchronisierung ist nicht möglich: Die betroffenen Bausteine auf der CPU oder im Engineering System müssen manuell angepasst werden. Eine der vorhandenen Baustein-Versionen (online oder offline) wird dabei überschrieben.
Es werden Unterschiede in der Hardware-Konfiguration auf der CPU (online) und im Engineering System (offline) festgestellt.	Unterschiede in der Hardware-Konfiguration müssen manuell angepasst werden. Entscheiden Sie in diesem Fall, welche Hardware-Konfiguration Sie übernehmen wollen. Soll die bereits auf der CPU vorhandene Hardware-Konfiguration erhalten bleiben, so sollten Sie diese vor dem Laden in Ihr Engineering System übernehmen. Soll die von Ihnen geänderte Hardware-Konfiguration übernommen werden, kommen Sie mit dem Laden ohne Synchronisierung fortfahren.	Automatische Synchronisierung ist nicht möglich: Die Hardware-Konfiguration muss manuell angepasst werden. Eine der vorhandenen Hardware-Konfigurationen (online oder offline) wird dabei überschrieben.

Über den Befehl "Laden in Gerät erzwingen" können Sie Bausteine auch ohne Synchronisierung laden, falls gewünscht.

Siehe auch

Projektdaten in ein Gerät laden (Seite 11)

Projektdaten auf eine Memory Card laden (Seite 13)

Projektdaten von einem Gerät laden (Seite 15)

1.2.2 Projektdaten in ein Gerät laden

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Ladens von Projektdaten in ein Gerät. Wie das Laden bestimmter Objekte funktioniert und welche Besonderheiten es dabei zu beachten gilt, finden Sie in der Hilfe zum Produkt.

Voraussetzung

- Die Projektdaten sind konsistent.
- Jedes zu ladende Gerät ist über einen Online-Zugang erreichbar.

Vorgehen

Um Projektdaten in ausgewählte Geräte zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein oder mehrere Geräte.
2. Klicken Sie mit der rechten Maustaste auf ein selektiertes Element.
   Das Kontextmenü wird geöffnet.
3. Wählen Sie im Kontextmenü im Untermenü "Laden in Gerät" die gewünschte Option.

   Hinweis

   Beachten Sie, dass es vom selektierten Gerät abhängt, welche Optionen Ihnen zur Verfügung stehen.

   Wenn erforderlich, werden die Projektdaten übersetzt.

   – Falls Sie bereits eine Online-Verbindung hergestellt hatten, wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
   – Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet und Sie müssen zunächst die Schnittstellen auswählen, über die Sie die Online-Verbindung zum Gerät herstellen möchten. Sie haben die Möglichkeit, sich alle kompatiblen Teilnehmer anzeigen zu lassen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken.
     Siehe auch: AUTOHOTSPOT
4. Kontrollieren Sie die Meldungen im Dialog "Vorschau laden" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

   ⚠ **WARNUNG**

   Personen- und Sachschaden verhindern

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren.

   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt. Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
6. Klicken Sie auf die Schaltfläche "Fertig stellen".

Ergebnis

Die gewählten Projektdaten wurden in die Geräte geladen.

Siehe auch

Allgemeine Informationen zum Laden (Seite 9)

Projektdaten auf eine Memory Card laden (Seite 13)

Projektdaten von einem Gerät laden (Seite 15)

1.2.3 Projektdaten auf eine Memory Card laden

Sie haben die Möglichkeit, Projektdaten auf eine Memory Card zu laden. Für CPUs der Baureihe S7-300/400 können Sie auch explizit das Anwenderprogramm auf eine Memory Card laden, die in der CPU steckt.

Um Projektdaten auf eine Memory Card zu laden, haben Sie die folgenden Möglichkeiten:

- Projektdaten per Drag & Drop auf eine Memory Card laden
- Projektdaten auf eine Memory Card schreiben
- Anwenderprogramm auf eine Memory Card laden, die in einer CPU der Baureihe S7-300/400 steckt

Voraussetzung

Eine Memory Card wird angezeigt.

Siehe auch: Auf Memory Cards zugreifen (Seite 103)

Projektdaten auf eine Memory Card laden

Um Projektdaten auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation die Projektdaten, die Sie laden möchten, auf die Memory Card.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Oder:

1. Selektieren Sie in der Projektnavigation die Projektdaten, die Sie laden möchten.
2. Klicken Sie mit der rechten Maustaste auf die Selektion und wählen Sie im Kontextmenü den Befehl "Kopieren". Alternativ können Sie auch die Tastenkombination <Strg+C> verwenden.
3. Klicken Sie mit der rechten Maustaste auf die Memory Card und wählen Sie im Kontextmenü den Befehl "Einfügen". Alternativ können Sie auch die Tastenkombination <Strg+V> verwenden.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Oder:

1. Selektieren Sie in der Projektnavigation die Projektdaten, die Sie laden möchten.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben".
   Der Dialog "Memory Card auswählen" wird geöffnet.
3. Wählen Sie eine Memory Card aus, die mit der CPU kompatibel ist.
   Im unteren Bereich des Dialogs wird eine Schaltfläche mit einem grünen Häkchen aktiv.
4. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Anwenderprogramm auf eine Memory Card in der CPU laden (nur S7-300/400)

Um das Anwenderprogramm auf eine Memory Card in einer CPU der Baureihe S7-300/400 zu laden, gehen Sie folgendermaßen vor.

1. Selektieren Sie in der Projektnavigation eine CPU der Baureihe S7-300/400.
2. Wählen Sie im Menü "Online" den Befehl "Anwenderprogramm auf Memory Card laden".
   Der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

4. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt und der Dialog "Ergebnisse des Ladevorgangs" wird angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und mögliche Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
6. Klicken Sie auf die Schaltfläche "Fertig stellen".

Siehe auch

Allgemeine Informationen zum Laden (Seite 9)

Projektdaten in ein Gerät laden (Seite 11)

Projektdaten von einem Gerät laden (Seite 15)

1.2.4 Projektdaten von einem Gerät laden

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Ladens von Projektdaten von einem Gerät. Welche Projektdaten Sie von einem Gerät laden können, ist von den installierten Produkten abhängig.

Beim Laden von Projektdaten von einem Gerät in Ihr Projekt haben Sie die folgenden Möglichkeiten:

- Gerät als neue Station laden
  Mit dieser Option laden Sie die vorhandenen Projektdaten eines Geräts als neue Station in Ihr Projekt.
- Projektdaten eines Geräts laden
  Mit dieser Option laden Sie die Projektdaten aus dem Gerät in eine vorhandene CPU im Projekt. Welche Projektdaten geladen werden, finden Sie in der Hilfe zum Produkt.

In beiden Fällen werden beim Laden alle Instanzen von Bibliothekstypen wieder mit der passenden Version des Typen in der Projektbibliothek verbunden. Wenn zu einer geladenen Instanz noch kein passender Typ oder nicht die richtige Version des Typen in der Projektbibliothek vorhanden ist, wird der Typ oder die Version in der Projektbibliothek ergänzt. Auf dem Gerät gespeicherte Bibliothekstypen im Zustand „in Test“ können Sie nur dann in das Projekt laden, wenn von der Testversion die zugrundeliegende freigegebene Typversion in der Projektbibliothek vorhanden ist.

Voraussetzung

- Ein Projekt ist geöffnet.
- Die zu ladende Hardware-Konfiguration und Software muss mit dem TIA Portal kompatibel sein. Stellen Sie die Kompatibilität insbesondere dann sicher, wenn die Daten auf dem Gerät mit einer früheren Programmversion oder mit einer anderen Projektierungssoftware erstellt wurden.

Gerät als neue Station laden

Um das komplette Gerät in Ihr Projekt zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation den Projektnamen.
2. Wählen Sie im Menü "Online" den Befehl "Laden des Geräts als neue Station (Hardware und Software)".
   Der Dialog "Gerät in PG/PC laden" wird geöffnet.
3. Wählen Sie in der Klappliste "Typ der PG/PC-Schnittstelle" den Typ der Schnittstelle aus, die Sie für den Ladevorgang verwenden möchten.
4. Wählen Sie in der Klappliste "PG/PC-Schnittstelle" die Schnittstelle aus, die verwendet werden soll.
5. Klicken Sie rechts neben der Klappliste "PG/PC-Schnittstelle" auf die Schaltfläche "Schnittstelle konfigurieren", um die Einstellungen für die gewählte Schnittstelle anzupassen.
   Siehe auch: AUTOHOTSPOT
6. Lassen Sie sich alle kompatiblen Teilnehmer anzeigen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken. Wählen Sie in der Tabelle der erreichbaren Teilnehmer den Teilnehmer aus, von dem Sie die Projektdaten laden möchten.
7. Klicken Sie auf "Laden".
   Abhängig vom ausgewählten Gerät erscheint ein Dialog, in dem Sie weitere Informationen eingeben müssen, wie z. B. die Position auf dem Baugruppenträger.
   Die Projektdaten des Geräts werden in das Projekt geladen. Sie können es offline bearbeiten und anschließend erneut in das Gerät laden.

Projektdaten eines Geräts laden

Um nur die Projektdaten von einem Gerät in Ihr Projekt zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zum Gerät her, von dem Sie Projektdaten laden möchten.
   Siehe auch: AUTOHOTSPOT
2. Selektieren Sie in der Projektnavigation das Gerät.
   Im Menü "Online" wird der Befehl "Laden von Gerät (Software)" aktiv.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät (Software)".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet.
4. Kontrollieren Sie die Meldungen im Dialog "Vorschau für das Laden von Gerät" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Laden von Projektdaten aus einer anderen Produktversion von einem Gerät

Beim Ausführen der Befehle "Laden von Gerät (Software)" und "Laden des Geräts als neue Station (Hardware und Software)" im Menü "Online" wird geprüft, ob die Projektdaten auf dem Gerät mit einer anderen als der geöffneten Version des TIA Portals erstellt wurden.

Wenn die Projektdaten mit einer anderen Produktversion erstellt wurden, erhalten Sie im Dialog "Vorschau für das Laden von Gerät" Informationen darüber, ob alle Voraussetzungen für das Laden erfüllt sind.

Beachten Sie die Hinweise zu den Voraussetzungen für das Laden und wählen Sie gegebenenfalls einen Befehl im Auswahlmenü in der Spalte "Aktion" aus.

Sobald die Ansicht aktualisiert wurde und das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.

Besonderheiten bei der Verwendung von Datenbausteinen aus anderen Produktversionen

Datenbausteine, die mit einer anderen Produktversion des TIA Portals erstellt wurden, können mit der aktuellen Produktversion nicht geöffnet und editiert werden. Weitere Informationen finden Sie im Kapitel "AUTOHOTSPOT".

Siehe auch

Allgemeine Informationen zum Laden (Seite 9)

Projektdaten in ein Gerät laden (Seite 11)

Projektdaten auf eine Memory Card laden (Seite 13)

1.2.5 Projektdaten von einer Memory Card laden

Beim Laden von Projektdaten von einer Memory Card in das Projekt haben Sie die folgenden Möglichkeiten:

- Projektdaten der Memory Card als neue Station laden
  Mit dieser Option laden Sie die Projektdaten einer Memory Card als neue Station in Ihr Projekt.
- Projektdaten der Memory Card in ein vorhandenes Gerät laden
  Mit dieser Option laden Sie die Projektdaten einer Memory Card in ein vorhandenes Gerät im Projekt. Welche Projektdaten geladen werden, finden Sie in der Hilfe zum Produkt.

In beiden Fällen werden beim Laden alle Instanzen von Bibliothekstypen wieder mit der passenden Version des Typs in der Projektbibliothek verbunden. Wenn zu einer geladenen Instanz noch kein passender Typ oder nicht die richtige Version des Typs in der Projektbibliothek vorhanden ist, wird der Typ oder die Version in der Projektbibliothek ergänzt.

Voraussetzung

- Ein Projekt ist geöffnet.
- Die Memory Card wird angezeigt.
  Siehe auch: Auf Memory Cards zugreifen (Seite 103)
- Die zu ladende Hardware-Konfiguration und Software muss mit dem TIA Portal kompatibel sein. Stellen Sie die Kompatibilität insbesondere dann sicher, wenn die Daten auf der Memory Card mit einer früheren Programmversion oder mit einer anderen Projektierungssoftware erstellt wurden.

Projektdaten als neue Station laden

Um die Projektdaten einer Memory Card als neue Station in das Projekt zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Memory Card, von der Sie die Projektdaten laden möchten.
2. Wählen Sie im Menü "Online" den Befehl "Laden des Geräts als neue Station (Hardware und Software)".

Oder:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf das Projekt.

Oder:

1. Klicken Sie mit der rechten Maustaste auf die Memory Card.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Klicken Sie mit der rechten Maustaste auf das Projekt.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Projektdaten in ein vorhandenes Gerät laden

Um die Projektdaten einer Memory Card in ein vorhandenes Gerät zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf ein Gerät im Projekt oder kopieren Sie die Memory Card und fügen die Daten in ein Gerät ein.
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet.
2. Kontrollieren Sie die Meldungen im Dialog "Vorschau für das Laden von Gerät" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Projektdaten vergleichen 2

2.1 Grundlagen zum Projektdatenvergleich

Funktion

Sie können Projektdaten des gleichen Typs miteinander vergleichen, um mögliche Unterschiede zu ermitteln. Grundsätzlich stehen Ihnen die folgenden Vergleichsarten zur Verfügung:

- Offline/Online-Vergleich
  Bei dieser Vergleichsart können Sie einen Software-Vergleich von Objekten eines Geräts mit den Objekten eines Projekts durchführen. Dazu muss eine Online-Verbindung zum Gerät aufgebaut werden.
- Offline/Offline-Vergleich
  Bei dieser Vergleichsart können Sie entweder einen Software- oder einen Hardware-Vergleich durchführen. Beim Software-Vergleich können Sie Objekte vergleichen, die aus Projekten oder Bibliotheken stammen. Der Hardware-Vergleich steht Ihnen für Geräte aus dem aktuell geöffneten Projekt oder aus Referenzprojekten zur Verfügung. Sowohl für den Software- als auch für den Hardware-Vergleich können Sie entscheiden, ob der Vergleich automatisch für alle Objekte durchgeführt wird oder ob Sie einzelne Objekte manuell vergleichen möchten.
- Detailvergleich
  Für einige Objekte, z. B. Bausteine, können Sie über den Offline/Online- und Offline/Offline-Vergleich hinaus einen Detailvergleich durchführen. Dabei werden die verglichenen Objekte nebeneinander geöffnet und die Unterschiede werden gekennzeichnet.

Ein einfacher Offline/Online-Vergleich findet bereits statt, wenn Sie eine Online-Verbindung herstellen. Dabei werden vergleichsfähige Objekte in der Projektnavigation mit Symbolen gekennzeichnet, die das Vergleichsergebnis repräsentieren. Darüber hinaus können Sie einen weitergehenden Offline/Online- und Offline/Offline-Vergleich im Vergleichseditor durchführen.

Wenn Sie Software vergleichen, können Sie dabei auch Aktionen für nicht identische Objekte festlegen.

Hinweis

- Nicht alle Objekte ermöglichen jede Vergleichsart. Für welche Projektdaten Sie welchen Vergleich durchführen können, ist von den installierten Produkten abhängig.
- Übersetzen Sie Ihr Anwenderprogramm, bevor Sie einen Vergleich oder Detailvergleich starten. Nach jeder Änderung des Programms während eines Vergleichs ist dies zu wiederholen, bevor Sie das Vergleichsergebnis aktualisieren. Damit stellen Sie sicher, dass der Vergleich den aktuellen Status anzeigt.
- Beachten Sie beim Beobachten eines Bausteins, dass bei einem Offline/Online-Vergleich ein Unterschied der Zeitstempel auch dann vorhanden sein kann, wenn Sie den Baustein ohne Codeänderung neu übersetzen. Die Prüfsumme bleibt dabei unverändert. Laden Sie in diesem Fall den Baustein in das Gerät und starten Sie anschließend das Beobachten erneut.
- In den folgenden Fällen werden die Zeitstempel der Objekte als Vergleichskriterien verwendet:
  - Export und erneuter Import über die Openness-Schnittstelle
  - Bausteinaufrufe
  - Zugriffe auf die globalen Daten eines Datenbausteins (DB)

  Dies kann zu unterschiedlichen Prüfsummen führen, obwohl die Objekte identisch sind.

Vergleich mithilfe von Prüfsummen

Für den Offline/Online- und Offline/Offline-Vergleich von Software-Objekten werden Prüfsummen verwendet, die für bestimmte Daten der Objekte erzeugt werden. Die Objekte sind nur gleich, wenn auch ihre Prüfsummen identisch sind.

Hinweis

Für CPUs der Baureihen S7-300/400 werden für den Offline/Online/-Vergleich weiterhin Zeitstempel verwendet.

Die Daten eines Objekts unterteilen sich in die folgenden beiden Kategorien:

- Quelldaten
  Quelldaten sind alle Daten eines Objekts, auf die Sie direkt Einfluss nehmen können, z. B. Name des Objekts, Quellcode, Kommentare oder Programmiersprache. Abhängig vom Objekt unterscheiden sich hier die Eigenschaften, die Sie editieren können.
- Zieldaten
  Zieldaten sind alle Übersetzungs- und Laufzeitdaten. Auf diese Daten haben Sie keinen direkten Einfluss, da sie auf der Grundlage ihrer Quelldaten vom System erzeugt werden. Auch hier unterscheiden sich die Eigenschaften abhängig vom Objekt.

Die folgende Tabelle gibt einen Überblick über die Bereiche des Eigenschaftsvergleichs:

Kategorie	Bereich	Beschreibung
Quelldaten	Safety	Prüfsumme für Safety-Objekte Weitere Informationen zum Vergleich von Safety-Objekten finden Sie in der Online-Hilfe zu SIMATIC Safety im Kapitel „Sicherheitsprogramme vergleichen“.
	Schnittstelle ohne Kommentare	Prüfsumme über alle Variablen der Bausteinschnittstelle der ersten Ebene. Kommentare der Variablen sind nicht Bestandteil der Prüfsumme.
	Schnittstelle von veröffentlichten Bausteinen ohne Kommentare (Software Units)	Prüfsumme über alle Schnittstellen-Prüfsummen von veröffentlichten Objekten einer Software Unit. Kommentare sind nicht Bestandteil der Prüfsumme.
	Textuelle Bausteinschnittstelle	Prüfsumme über alle Variablen, Kommentare und Formatierungen in den textuellen Bausteinschnittstellen von SCL-Bausteinen.
	Code ohne Kommentare	Prüfsumme über den Quellcode. Baustein- und Netzwerkkommentare gehen nicht in die Prüfsumme ein. Nicht mehrsprachige Kommentare, z. B. in AWL, und das Umhängen von freien Kommentaren an andere Boxen in KOP und FUP gehen jedoch in die Prüfsumme ein.
	Kommentare (mehrsprachig)	Prüfsumme über alle Baustein- und Netzwerkkommentare
	Sprachkonfiguration	Offline/Online-Vergleich: Zeigt die Sprachen, die beim Ladevorgang mit ins Gerät geladen werden. Offline/Offline-Vergleich: Zeigt die aktivierten Sprachen im Projekt.
	Ereignis	Prüfsumme über alle ereignisrelevanten Daten eines Organisationsbausteins (OB).
	Eigenschaften	Prüfsumme über alle einstellbaren Eigenschaften eines Objekts.
	Erweiterte Eigenschaften von Programmelementen in Software Units	Prüfsumme über den Veröffentlichungsstatus von Objekten in Software Units.
	Konfiguration von Technologieobjekten	Prüfsumme über die technologiespezifischen Einstellungen eines Technologieobjekts.
	Meldungskonfiguration	Prüfsumme über alle meldespezifischen Einstellungen eines Objekts.
	Supervisionskonfiguration	Prüfsumme über alle Supervisions-Einstellungen eines Objekts.
	Textlisten	Prüfsumme über alle Textlisten eines Objekts.
	Variablen ohne Kommentare	Prüfsumme über alle Variablen der selektierten PLC-Variablentabelle. Die Spalten "Remanenz" und "Überwachung" sowie die Kommentare der Variablen sind nicht Bestandteil der Prüfsumme.
	Konstanten ohne Kommentare	Prüfsumme über alle Anwenderkonstanten der selektierten PLC-Variablentabelle. Systemkonstanten und die Kommentare der Anwenderkonstanten sind nicht Bestandteil der Prüfsumme.

Kategorie	Bereich	Beschreibung
Zieldaten	Übersetzungs- und Laufzeitdaten	Prüfsumme über übersetzungs- und laderelevante Daten.
	Zeitstempel	Zeitstempel für S7-300/400-CPUs, die vom System beim Übersetzen und Laden erstellt werden. Nur verfügbar, wenn eine Online-Verbindung besteht.

Hinweis

- Beachten Sie, dass es vom Objekt abhängt, welche Prüfsummen für den Vergleich tatsächlich verwendet werden.
- Die Werteeinstellungen von Technologieobjekten gehen auch in die Prüfsumme der Bausteinschnittstelle ein.
- Bei know-how-geschützten Bausteinen, welche in einer älteren Version erstellt wurden, werden unter "Quelldaten" die Prüfsummen "Code ohne Kommentare" und "Kommentare (mehrsprachig)" nicht angezeigt.

Siehe auch

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Detailvergleich durchführen (Seite 35)

Vergleichseditor verwenden (Seite 26)

2.2 Offline/Online-Vergleich durchführen

Der Offline/Online-Vergleich erlaubt Ihnen den Vergleich von Objekten eines Geräts mit den Objekten eines Projekts.

Voraussetzung

Die Projektnavigation ist geöffnet.

Vorgehen

Um einen Offline/Online-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Online-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".
3. Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".
   Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.

Ergebnis

Alle Objekte, die online und offline vorhanden sind, werden angezeigt. Den Status der Objekte können Sie an den Symbolen im Vergleichseditor und in der Projektnavigation ablesen. Im Vergleichseditor können Sie nun abhängig vom Status der Objekte bestimmte Aktionen für die Objekte festlegen.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Offline-Vergleich durchführen (Seite 24)

Vergleichseditor verwenden (Seite 26)

Detailvergleich durchführen (Seite 35)

2.3 Offline/Offline-Vergleich durchführen

Beim Offline/Offline-Vergleich besteht die Möglichkeit, die Projektdaten zweier Geräte zu vergleichen. Dabei ist sowohl ein Software- als auch ein Hardware-Vergleich möglich. Beim Software-Vergleich können Sie Objekte vergleichen, die aus Projekten oder Bibliotheken stammen. Der Hardware-Vergleich steht Ihnen für Geräte aus dem aktuell geöffneten Projekt oder aus Referenzprojekten zur Verfügung. Sie können entscheiden, ob der Vergleich automatisch für alle Objekte durchgeführt wird oder ob Sie einzelne Objekte manuell vergleichen möchten. Beim Software-Vergleich stehen Ihnen zusätzlich die folgenden Optionen zur Verfügung:

- Sie können für nicht identische Objekte Aktionen festlegen, um die Unterschiede zu beseitigen.
- Sie können die Kriterien festlegen, die für den Vergleich verwendet werden sollen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Voraussetzung

Die Projektnavigation ist geöffnet.

Offline/Offline-Vergleich für Software durchführen

Um einen Offline/Offline-Vergleich für Software durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich.
   Alle vorhandenen Objekte der ausgewählten Geräte werden abhängig von den Einstellungen im Register "Software" des Vergleichseditors angezeigt und es wird ein automatischer Vergleich durchgeführt. Den Status der Objekte können Sie an den Symbolen im Vergleichseditor ablesen. Legen Sie bei Bedarf für nicht identische Objekte eine Aktion fest, um die Unterschiede zu beseitigen.
4. Legen Sie bei Bedarf die Vergleichskriterien fest.
5. Selektieren Sie ein Objekt, um im Eigenschaftsvergleich die Details anzuzeigen.
6. Wenn Sie einen manuellen Vergleich durchführen möchten, klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich. Selektieren Sie anschließend die Objekte, die Sie miteinander vergleichen möchten.
   Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen.

Offline/Offline-Vergleich für Hardware durchführen

Um einen Offline/Offline-Vergleich für Hardware durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich.
4. Öffnen Sie das Register "Hardware".
5. Wenn Sie einen manuellen Vergleich durchführen möchten, klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich. Selektieren Sie anschließend die Objekte, die Sie miteinander vergleichen möchten.
   Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Vergleichseditor verwenden (Seite 26)

Detailvergleich durchführen (Seite 35)

2.4 Vergleichseditor verwenden

2.4.1 Übersicht über den Vergleichseditor

Funktion

Der Vergleichseditor stellt die Ergebnisse eines Vergleichs in einer tabellarischen Übersicht dar. Dabei variiert das Aussehen leicht bei einem Offline/Online-Vergleich bzw. einem Hardware-/Software-Vergleich.

Aufbau des Vergleichseditors

Das folgende Bild zeigt den Aufbau des Vergleichseditors bei einem Offline/Online-Vergleich:

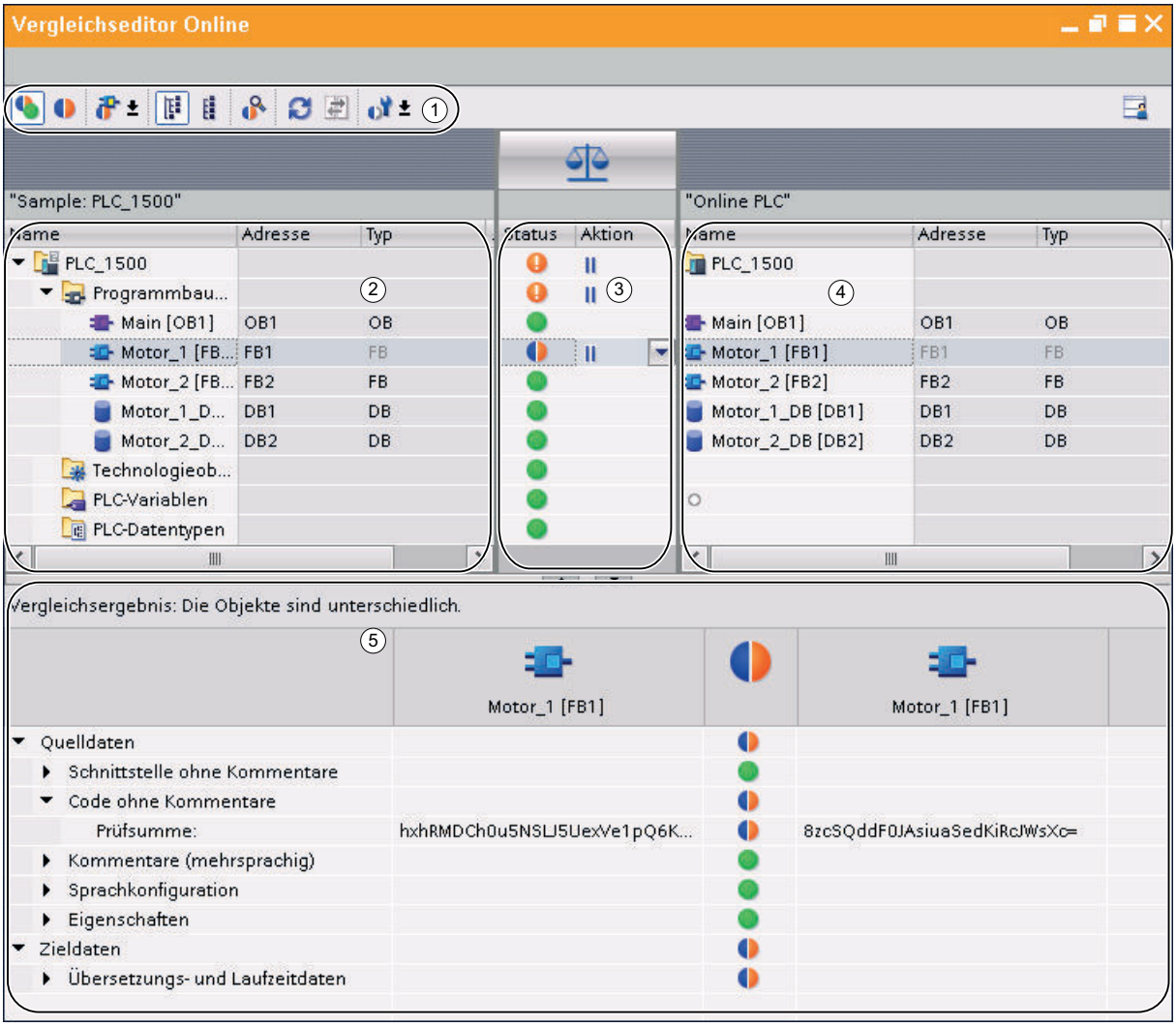

① Funktionsleiste des Vergleichseditors

② Linke Vergleichstabelle

③ Status- und Aktionsbereich

④ Rechte Vergleichstabelle

⑤ Eigenschaftsvergleich

Das folgende Bild zeigt den Aufbau des Vergleichseditors bei einem Offline/Offline-Vergleich (Software):

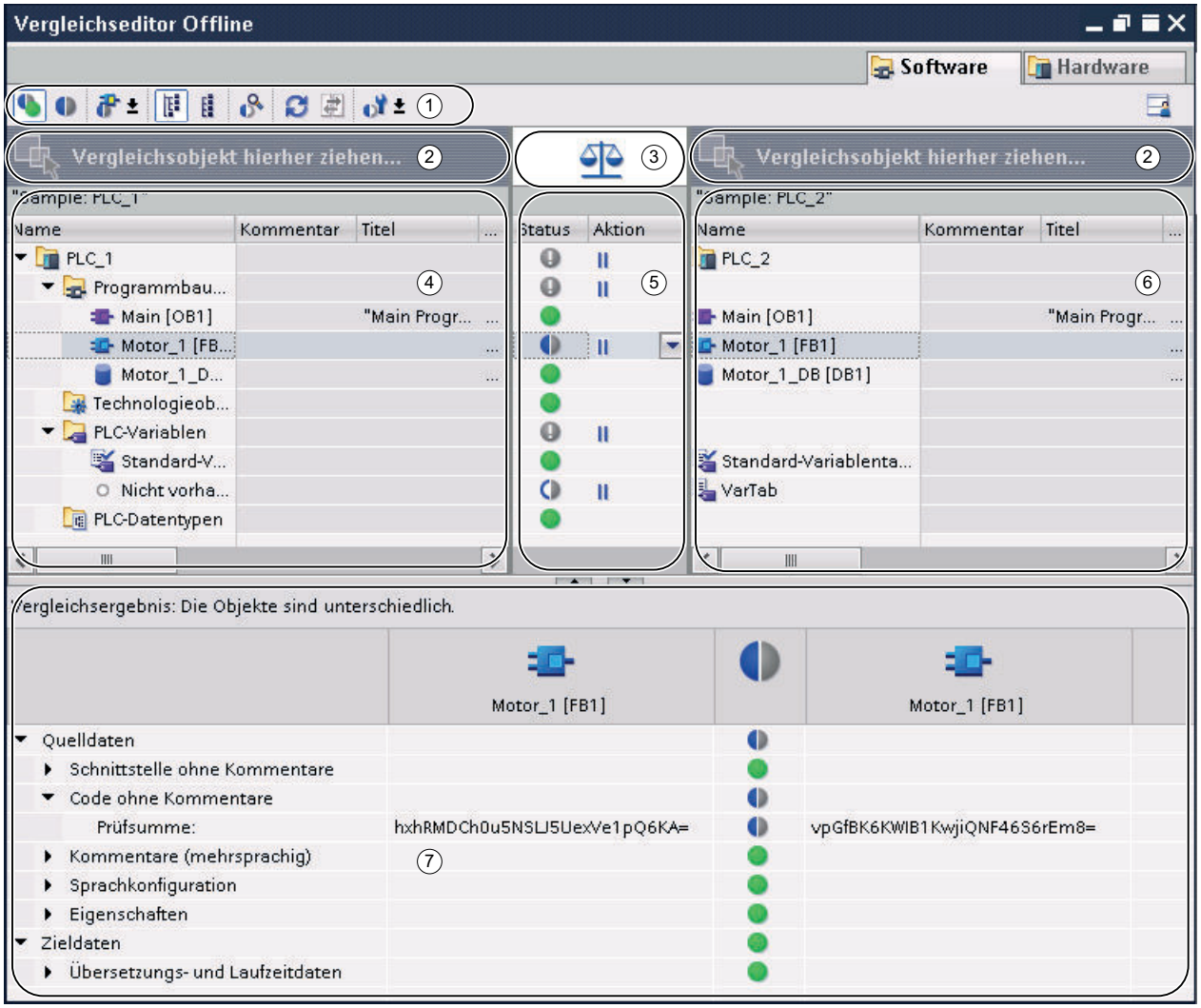

① Funktionsleiste des Vergleichseditors

② Drop-Flächen

③ Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich

④ Linke Vergleichstabelle

⑤ Status- und Aktionsbereich

⑥ Rechte Vergleichstabelle

⑦ Eigenschaftsvergleich

Das folgende Bild zeigt den Aufbau des Vergleichseditors bei einem Offline/Offline-Vergleich (Hardware):

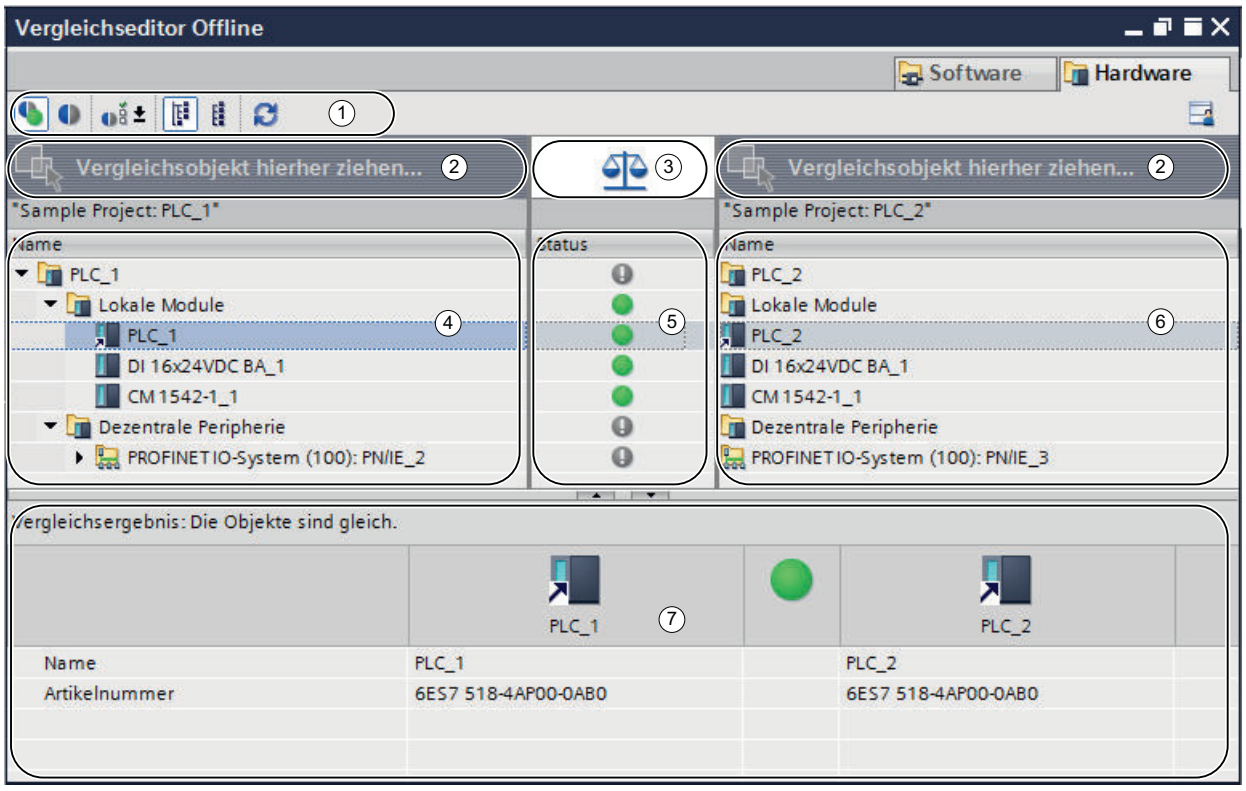

① Funktionsleiste des Vergleichseditors

② Drop-Flächen

③ Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich

④ Linke Vergleichstabelle

⑤ Statusbereich

⑥ Rechte Vergleichstabelle

⑦ Eigenschaftsvergleich

Funktionsleiste des Vergleichseditors

Über die Funktionsleiste haben Sie Zugriff auf folgende Funktionen des Vergleichseditors:

- Identische und unterschiedliche Objekte anzeigen
  Sie können identische Objekte einblenden, wenn Sie den Vergleich vollständig angezeigt bekommen möchten.
- Nur unterschiedliche Objekte anzeigen
  Sie können identische Objekte ausblenden, um die Übersichtlichkeit zu verbessern.
- Weitere verfügbare Filter anzeigen (nur Offline/Online-Vergleich und Offline/Offline-Vergleich für Software)
  Sie können festlegen, welche Objekte verglichen werden sollen.

- Verfügbare Zuordnungskriterien anzeigen (nur Offline/Offline-Vergleich für Hardware)
  Sie können festlegen, nach welchem Kriterium die Baugruppen für den Vergleich einander zugeordnet werden sollen.
- Ansicht ändern
  Sie können zwischen einer hierarchischen und einer flachen Ansicht wählen. In der hierarchischen Ansicht werden die Geräte in ihrer Struktur dargestellt, in der flachen Ansicht werden die Objekte der Geräte ohne Struktur aufgelistet.
- Detailvergleich starten (nur Offline/Online-Vergleich und Offline/Offline-Vergleich für Software)
  Sie können für Objekte einen Detailvergleich starten, um sich die einzelnen Unterschiede anzeigen zu lassen. Diese Funktion steht jedoch nicht für alle Objekte zur Verfügung.
- Anzeige aktualisieren
  Nachdem Sie Objekte geändert haben, können Sie mit dieser Funktion die Vergleichsergebnisse aktualisieren.
- Aktionen ausführen (nur Offline/Online-Vergleich und Offline/Offline-Vergleich für Software)
  Sie können nicht identische Objekte durch bestimmte Aktionen synchronisieren.
- Vergleichskriterien festlegen (nur Offline/Online-Vergleich und Offline/Offline-Vergleich für Software)
  Sie können festlegen, welche Kriterien für den Vergleich ausgewertet werden. Deaktivierte Kriterien haben keinen Einfluss auf das Gesamtergebnis des Vergleichs.

Drop-Flächen

Bei einem Offline/Offline-Vergleich können Sie in die Drop-Flächen die Geräte ziehen, die Sie vergleichen möchten. Bei einem Software-Vergleich können die zu vergleichenden Geräte aus dem geöffneten Projekt, aus Referenzprojekten, aus der Projektbibliothek oder aus globalen Bibliotheken stammen. Beachten Sie jedoch, dass Sie komplette Bibliotheken nur in die rechte Drop-Fläche ziehen können. Bei einem Hardware-Vergleich können Sie Geräte aus dem geöffneten Projekt oder aus Referenzprojekten vergleichen.

Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich

Bei einem Offline/Offline-Vergleich können Sie zwischen automatischem und manuellem Vergleich umschalten. Beim automatischen Vergleich werden die zu vergleichenden Objekte automatisch einander zugeordnet, beim manuellen Vergleich können Sie selbst die Objekte wählen, die verglichen werden sollen.

Vergleichstabellen

In den Vergleichstabellen werden die Objekte der Geräte angezeigt, die miteinander verglichen werden.

Die folgende Tabelle zeigt die Bedeutung der Spalten der Vergleichstabellen:

Spalte	Erläuterung
Name	Name des Vergleichsobjekts
Kommentar	Kommentar zum Vergleichsobjekt
Titel	Titel des Vergleichsobjekts

Spalte	Erläuterung
Adresse	Adresse des Vergleichsobjekts
Nummerierung	Art der Nummerierung für das Vergleichsobjekt
Typ	Typ des Vergleichsobjekts
Sprache	Programmiersprache, die für das Vergleichsobjekt eingestellt ist.
Zeitstempel Schnittstelle	Zeitpunkt der letzten Änderung der Bausteinschnittstelle
Zeitstempel Code	Zeitpunkt der letzten Änderung des Quellcodes
Autor	Name des Autors des Vergleichsobjekts
Version	Version des Vergleichsobjekts
Familie	Name der Objektfamilie
Ladespeicher	Speicherauslastung des Ladespeichers des Vergleichsobjekts
Arbeitsspeicher	Speicherauslastung des Arbeitsspeichers des Vergleichsobjekts
Datum der letzten Änderung	Zeitpunkt der letzten Änderung
Optimierter Bausteinzugriff	Gibt an, ob für einen Baustein die Option "Optimierter Bausteinzugriff" aktiviert ist.
Signatur	Signatur des Vergleichsobjekts (SIMATIC Safety*)
Interfacesignatur	Signatur der Bausteinschnittstelle des Vergleichsobjekts (SIMATIC Safety*)

* Weitere Informationen zum Vergleich von Safety-Objekten finden Sie in der Online-Hilfe zu SIMATIC Safety im Kapitel „Sicherheitsprogramme vergleichen“.

Es sind nicht alle Spalten in jeder Vergleichsart verfügbar, beim Hardware-Vergleich z. B. enthalten die Vergleichstabellen nur die Spalte "Name".

Standardmäßig sind nicht alle Spalten eingeblendet. Sie können jedoch wie in allen Tabelleneditoren die Spalten nach Bedarf ein- oder ausblenden und nach den einzelnen Spalten sortieren.

Status- und Aktionsbereich

Der Status- und Aktionsbereich bietet Ihnen folgende Möglichkeiten:

- Sie können sich die Ergebnisse eines automatischen Vergleichs ansehen. Die Ergebnisse werden mit Symbolen angezeigt.
- Bei einem Offline/Online-Vergleich und einem Offline/Offline-Vergleich für Software können Sie Aktionen für nicht identische Objekte festlegen.

Status- und Aktionssymbole

Die folgende Tabelle zeigt die Symbole für die Vergleichsergebnisse eines Offline/Online-Vergleichs:

Symbol	Erläuterung
	Ordner enthält Objekte, deren Online- und Offline-Version verschieden sind
	Vergleichsergebnis ist entweder nicht bekannt oder kann aus den folgenden Gründen nicht angezeigt werden: • Die Rechte für den Zugriff auf eine geschützte CPU sind nicht vorhanden. • Der Ladevorgang auf die CPU wurde mit einer TIA Portal-Version kleiner V14 durchgeführt.
	Online- und Offline-Version des Objekts sind identisch
	Online- und Offline-Version des Objekts sind verschieden
	Objekt nur offline vorhanden
	Objekt nur online vorhanden
	Dieses Vergleichskriterium ist deaktiviert und die zugehörige Prüfsumme fließt nicht ins Vergleichergebnis ein.

Die folgende Tabelle zeigt die Symbole für die Vergleichsergebnisse eines Offline/Offline-Vergleichs:

Symbol	Erläuterung
	Ausgangsprogramm
	Verglichene Version
	Ordner enthält Objekte, deren verglichene Versionen verschieden sind
	Ergebnis des Offline/Offline-Vergleichs ist nicht bekannt
	Die verglichenen Versionen des Objekts sind identisch
	Die verglichenen Versionen des Objekts sind verschieden
	Objekt nur im Ausgangsprogramm vorhanden
	Objekt nur in der verglichenen Version vorhanden
	Nur Hardware-Vergleich: Die unterlagerten Objekte des Containers sind identisch, es gibt jedoch Unterschiede zwischen den Containern selbst. Ein solcher Container kann z. B. ein Baugruppenträger sein.
	Nur Hardware-Vergleich: Die unterlagerten Objekte des Containers sind unterschiedlich. Zusätzlich gibt es Unterschiede zwischen den Containern. Ein solcher Container kann z. B. ein Baugruppenträger sein.
	Dieses Vergleichskriterium ist deaktiviert und die zugehörige Prüfsumme fließt nicht ins Vergleichergebnis ein.

Die folgende Tabelle zeigt die Symbole für die möglichen Aktionen bei einem Software-Vergleich:

Symbol	Erläuterung
	Keine Aktion
	Objekt der verglichenen Version mit dem Objekt aus dem Ausgangsprogramm überschreiben
	Objekt des Ausgangsprogramms mit dem Objekt aus der verglichenen Version überschreiben
	Unterschiedliche Aktionen für die Vergleichsobjekte innerhalb des Ordners

Eigenschaftsvergleich

Der Eigenschaftsvergleich vergleicht die Eigenschaften der selektierten Vergleichsobjekte. Das Ergebnis wird mit Symbolen angezeigt. Beim manuellen Vergleich erfolgt nur ein Eigenschaftsvergleich, sodass der Status- und Aktionsbereich leer bleibt. Beim automatischen Vergleich können Sie den Eigenschaftsvergleich zusätzlich zum Vergleich in den Vergleichstabellen verwenden.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Detailvergleich durchführen (Seite 35)

Vergleichsergebnisse aktualisieren (Seite 36)

Vergleichskriterien festlegen (Seite 37)

Nicht identische Objekte synchronisieren (Seite 38)

Ansicht ändern (Seite 40)

2.4.2 Tabellenspalten ein- und ausblenden

Bei einem Software-Vergleich können Sie die Spalten der Vergleichstabellen bei Bedarf ein- oder ausblenden.

Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
5. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.

Ergebnis

Die Spalten werden sowohl in der linken als auch in der rechten Vergleichstabelle ein- oder ausgeblendet.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Ansicht im Vergleichseditor filtern (Seite 33)

Detailvergleich durchführen (Seite 35)

Vergleichsergebnisse aktualisieren (Seite 36)

Vergleichskriterien festlegen (Seite 37)

Nicht identische Objekte synchronisieren (Seite 38)

Ansicht ändern (Seite 40)

2.4.3 Ansicht im Vergleichseditor filtern

Um die Übersichtlichkeit des Vergleichseditors zu verbessern, können Sie die Ansicht über folgende Filtermechanismen einschränken:

- Identische Vergleichsobjekte ausblenden
  Sie können Vergleichsobjekte, deren Offline/Online- bzw. Offline/Offline-Versionen identisch sind, ausblenden. Umgekehrt können Sie solche Vergleichsobjekte, wenn Sie ausgeblendet sind, jederzeit wieder einblenden.
- Angezeigte Objekte auswählen
  Bei einem Offline/Online-Vergleich oder einem Offline/Offline-Vergleich für Software können Sie festlegen, für welche Objekte die Vergleichsergebnisse angezeigt werden sollen.

Voraussetzung

Der Vergleichseditor ist geöffnet.

Identische Vergleichsobjekte ausblenden

Um identische Objekte auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Nur unterschiedliche Objekte anzeigen".
   Es werden nur die Elemente angezeigt, die online und offline unterschiedlich sind.

Identische Vergleichsobjekte einblenden

Um die identischen Objekte wieder einzublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Identische und unterschiedliche Objekte anzeigen".
   Es werden alle Elemente angezeigt.

Angezeigte Objekte auswählen

Um die Objekte auszuwählen, für die die Vergleichsergebnisse angezeigt werden sollen, gehen Sie folgendermaßen vor:

1. Führen Sie einen Offline/Online-Vergleich oder einen Offline/Offline-Vergleich für Software durch.
2. Klicken Sie in der Funktionsleiste auf den Pfeil der Schaltfläche "Weitere verfügbare Filter anzeigen".
3. Wählen Sie den gewünschten Filter aus.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Detailvergleich durchführen (Seite 35)

Vergleichsergebnisse aktualisieren (Seite 36)

Vergleichskriterien festlegen (Seite 37)

Nicht identische Objekte synchronisieren (Seite 38)

Ansicht ändern (Seite 40)

2.4.4 Detailvergleich durchführen

Hinweis

Beachten Sie die folgenden Hinweise:

- Nicht alle Objekte ermöglichen einen Detailvergleich. Für welche Projektdaten Sie einen Detailvergleich durchführen können, ist von den installierten Produkten abhängig. Beim Hardware-Vergleich ist kein Detailvergleich der Hardware-Komponenten möglich.
- Für Bausteine, PLC-Variablen und PLC-Datentypen können Sie den Detailvergleich direkt aus der Projektnavigation starten. Weitere Informationen finden Sie im Kapitel "PLC-Programme vergleichen".

Vorgehen

Um einen Detailvergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Führen Sie zunächst einen Offline/Online- oder einen Offline/Offline-Vergleich für Software durch.
   Der Vergleichseditor wird geöffnet.

   Hinweis

   Einen Detailvergleich können Sie nur für Objekte durchführen, die sowohl in der linken als auch in der rechten Vergleichstabelle vorhanden sind.

2. Selektieren Sie im Vergleichseditor das Objekt, für das Sie einen Detailvergleich durchführen möchten.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".

Ergebnis

Das Objekt und sein zugehöriger Vergleichspartner werden in ihren entsprechenden Editoren geöffnet. Die Editoren werden nebeneinander angeordnet, sodass Sie sich schnell einen Überblick über die Unterschiede verschaffen können. Weitere Informationen finden Sie bei der Beschreibung des jeweiligen Objektvergleichs.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Vergleichsergebnisse aktualisieren (Seite 36)

Vergleichskriterien festlegen (Seite 37)

Nicht identische Objekte synchronisieren (Seite 38)

Ansicht ändern (Seite 40)

2.4.5 Vergleichsergebnisse aktualisieren

Sobald Sie ein Objekt verändern, sind die Vergleichsergebnisse nicht mehr gültig und müssen aktualisiert werden.

Hinweis

Bei einem Offline/Online-Vergleich kann es vorkommen, dass durch Änderungen im Gerät der Vergleichseditor vom System automatisch aktualisiert wird, falls Objekte des Vergleichs von der Änderung betroffen sind. Dies kann folgende Auswirkungen haben:

- Eventuell werden einige der von Ihnen festgelegten Aktionen ungültig, z. B. falls das Objekt im Gerät nicht mehr existiert. Objekte mit solchen ungültigen Aktionen werden hervorgehoben, damit Sie eine neue, gültige Aktion festlegen können.
- Möglicherweise wird auch Ihre Auswahl aufgehoben, die Sie vor dem automatischen Aktualisieren getroffen haben.

Voraussetzung

Der Vergleichseditor ist geöffnet.

Vorgehen

Um die Vergleichsergebnisse zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Anzeige aktualisieren".
   Die Vergleichsergebnisse werden aktualisiert.

   Hinweis

   Beachten Sie, dass Ihnen die Schaltfläche "Anzeige aktualisieren" nicht zur Verfügung steht, während der Vergleichseditor Inhalte lädt oder synchronisiert.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Detailvergleich durchführen (Seite 35)

Vergleichskriterien festlegen (Seite 37)

Nicht identische Objekte synchronisieren (Seite 38)

Ansicht ändern (Seite 40)

2.4.6 Vergleichskriterien festlegen

Sie haben die Möglichkeit festzulegen, welche Kriterien für einen Vergleich berücksichtigt werden sollen. Die festgelegten Kriterien werden für alle Objekte eines Vergleichs herangezogen, soweit sie für ein Objekt zur Verfügung stehen. Steht ein Kriterium für ein Objekt nicht zur Verfügung, z. B. "Code ohne Kommentare" für Datenbausteine, wird das Vergleichsergebnis für diese Prüfsumme als "Gleich" angesehen.

Für ein Kriterium, das Sie deaktiviert haben, können Sie dennoch die Prüfsummen einsehen, für das Gesamtvergleichsergebnis wird es jedoch nicht verwendet. Deaktivierte Vergleichskriterien können Sie im Eigenschaftsvergleich am Filtersymbol erkennen. Das Filtersymbol kann auch im oberen Bereich des Vergleichseditors angezeigt werden, wenn durch die Auswahl der Vergleichskriterien keine Prüfsummen mehr in das Vergleichsergebnis einfließen.

Hinweis

Bei einem Offline/Online-Vergleich haben die Vergleichskriterien keinen Einfluss auf das Vergleichsergebnis in der Projektnavigation. Beachten Sie, dass es dadurch zu einem unterschiedlichen Ergebnis in der Projektnavigation und im Vergleichseditor kommen kann.

Vorgehen

Um festzulegen, welche Kriterien beim Vergleich berücksichtigt werden sollen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil der Schaltfläche "Weitere Vergleichskriterien anzeigen".
2. Aktivieren Sie alle Kriterien, die beim Vergleich berücksichtigt werden sollen.
3. Deaktivieren Sie alle Kriterien, die beim Vergleich nicht berücksichtigt werden sollen.
4. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen im unteren Bereich der Klappliste.
5. Mit einem Klick auf die Schaltfläche mit dem Kreuz schließen Sie die Klappliste, ohne die Änderungen zu übernehmen.

Ergebnis

Es werden nur noch die aktivierten Kriterien für den Vergleich berücksichtigt.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)
Übersicht über den Vergleichseditor (Seite 26)
Tabellenspalten ein- und ausblenden (Seite 32)
Ansicht im Vergleichseditor filtern (Seite 33)
Detailvergleich durchführen (Seite 35)
Vergleichsergebnisse aktualisieren (Seite 36)
Nicht identische Objekte synchronisieren (Seite 38)
Ansicht ändern (Seite 40)

2.4.7 Nicht identische Objekte synchronisieren

2.4.7.1 Aktionen festlegen

Wenn Sie einen Vergleich durchgeführt haben, können Sie im Vergleichseditor festlegen, welche Aktionen auf nicht identische Objekte angewendet werden sollen. Für identische Objekte können Sie keine Aktionen wählen. Beachten Sie, dass beim Hardware-Vergleich keine Aktionen ausgeführt werden können.

Bei einem Offline/Online-Vergleich sind nur Synchronisationsaktionen in eine Richtung zulässig, damit die Programmkonsistenz erhalten bleibt. So können Sie z. B. mehrere Bausteine in ein Gerät oder von einem Gerät laden, aber Sie können in einer Synchronisationsaktion keine gemischten Ladeaktionen durchführen. Dabei bestimmt die erste Aktion, die Sie im Vergleichseditor einstellen, die Synchronisationsrichtung. Stellen Sie also z. B. bei einem Baustein ein, dass der Offline-Baustein in das Gerät geladen werden soll, können Sie über eine Synchronisierung die anderen Objekte auch nur in das Gerät laden. Um doch wieder Objekte aus dem Gerät zu laden, stellen Sie zunächst die Option "Keine Aktion" ein, anschließend können Sie die Aktion wieder nach Bedarf einstellen. Oder Sie führen einen neuen Vergleich durch.

Hinweis

Beachten Sie beim Festlegen von Aktionen folgende CPU-spezifische Besonderheiten:

- S7-300/400: Sie können Aktionen für den Ordner "Programmbausteine", für Ordner, die Sie selbst erstellt haben, oder für einzelne Bausteine festlegen.
- S7-1200/1500: Sie können Aktionen für den Ordner "Programmbausteine", für Ordner, die Sie selbst erstellt haben, oder für einzelne Bausteine festlegen. Wenn Sie einen Offline/Online-Vergleich durchgeführt haben und als Aktion das Laden in das Gerät wählen, wird ein konsistentes Laden durchgeführt. Wenn Sie dagegen als Aktion das Objekt vom Gerät in das Projekt laden, können auch einzelne Bausteine geladen werden.

Voraussetzung

Der Vergleichseditor ist geöffnet.

Vorgehen

Um für ein nicht identisches Objekt eine Aktion auszuwählen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Status- und Aktionsbereich in der Spalte "Aktion" in die Zelle des Objekts, für das Sie eine Aktion festlegen möchten.
   Die Zelle ändert sich in eine Klappliste.
2. Klicken Sie in die Klappliste.
3. Wählen Sie die gewünschte Aktion aus.
   Für das Objekt wird beim nächsten Synchronisieren die eingestellte Aktion ausgeführt. Wenn Sie die zuvor eingestellte Auswahl der Aktion unbeabsichtigt geändert haben, können Sie die Änderung bis zur nächsten Synchronisation rückgängig machen.
4. Um die zuvor eingestellte Auswahl der Aktion wiederherzustellen, klicken Sie im Status- und Aktionsbereich mit der rechten Maustaste auf die Aktion, die Sie wiederherstellen möchten.
5. Wählen Sie im Kontextmenü den Befehl "Letzte Auswahl wieder herstellen".

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Vergleichsergebnisse aktualisieren (Seite 36)

Objekte synchronisieren (Seite 39)

2.4.7.2 Objekte synchronisieren

Durch das Synchronisieren werden die Aktionen ausgeführt, die Sie für nicht identische Objekte festgelegt haben. Beachten Sie dabei jedoch, dass Sie bei einem Offline/Online-Vergleich in einer Synchronisationsaktion nur Aktionen in eine Richtung ausführen können.

Voraussetzung

- Der Vergleichseditor ist geöffnet.
- Die gewünschten Aktionen sind ausgewählt.

Vorgehen

Um die Objekte zu synchronisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Aktionen ausführen".

Ergebnis

Die Aktionen, die Sie für die Objekte festgelegt haben, werden ausgeführt.

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Vergleichsergebnisse aktualisieren (Seite 36)

Aktionen festlegen (Seite 38)

2.4.8 Ansicht ändern

Sie können für die linke Vergleichstabelle zwischen einer hierarchischen und einer flachen Ansicht wählen. In der hierarchischen Ansicht werden die Geräte in ihrer Struktur dargestellt, in der flachen Ansicht werden die Objekte der Geräte ohne Struktur aufgelistet. In der rechten Vergleichstabelle werden die Objekte immer flach dargestellt.

Hierarchische Ansicht einstellen

Um die hierarchische Ansicht einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Vergleichseditors auf die Schaltfläche "Hierarchisch anzeigen".

Flache Ansicht einstellen

Um die flache Ansicht einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Vergleichseditors auf die Schaltfläche "Flach anzeigen".

Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 19)

Offline/Online-Vergleich durchführen (Seite 23)

Offline/Offline-Vergleich durchführen (Seite 24)

Übersicht über den Vergleichseditor (Seite 26)

Tabellenspalten ein- und ausblenden (Seite 32)

Ansicht im Vergleichseditor filtern (Seite 33)

Detailvergleich durchführen (Seite 35)
Vergleichsergebnisse aktualisieren (Seite 36)
Vergleichskriterien festlegen (Seite 37)
Nicht identische Objekte synchronisieren (Seite 38)

Projektdaten schützen 3

3.1 Schutzkonzept für Projektdaten

Einführung

Sie haben die Möglichkeit Ihre Projektdaten vor unberechtigtem Zugriff zu schützen. Dazu gehören z. B.:

- Zugriffsschutz für Geräte
- Kopier- und Anzeigeschutz von Objekten
- Einschränkungen beim Ausdruck von know-how-geschützten Objekten

Wenn Sie ein Objekt mit Know-how-Schutz versehen, dann bleibt dieser Schutz auch nach dem Einfügen in eine Bibliothek bestehen. Beachten Sie, dass nicht für alle Objekte jeder Schutzmechanismus zur Verfügung steht. Wie Sie bestimmte Objekte schützen können, finden Sie in der Hilfe zum Produkt.

Beim Schützen von Bausteinen können Unternehmen auch die Verwendung eines Passwort-Providers festlegen. Dazu müssen die zentralen Einstellungen angepasst werden und alle Anwender müssen die entsprechenden Passwort-Provider in das TIA Portal einbinden. Anschließend können die durch einen Passwort-Provider bereitgestellten Passwörter genutzt werden. Weitere Informationen finden Sie im Kapitel "Bausteine schützen".

Zugriffsrechte für Geräte zurücknehmen

Wenn Sie eine Funktion ausführen möchten, die über die Schutzstufe des Geräts passwortgeschützt ist, erhalten Sie einen Dialog zur Passwortabfrage. Bei korrekter Eingabe des Passworts können Sie die gewünschte Funktion ausführen. Das Zugriffsrecht auf das Gerät bleibt bestehen, bis Sie das TIA Portal schließen.

Wenn Sie den Passwortschutz bei geöffnetem TIA Portal wieder aktivieren möchten, können Sie die Zugriffsrechte für ein Gerät explizit wieder zurücknehmen. Dadurch können bestimmte Funktionen für das geschützte Gerät erst nach einer erneuten Eingabe des korrekten Passworts ausgeführt werden. Für welche Funktionen ein Passwort eingegeben werden muss, legen Sie bei der Parametrierung der Schutzstufe des Geräts fest.

Geschützte Projektdaten weitergeben

Geben Sie geschützte Projektdaten immer als Projekt- oder Bibliotheksarchiv weiter. Dadurch stellen Sie sicher, dass die Schutzmaßnahmen nicht umgangen werden können.

⚠ WARNUNG

Umgang mit know-how-geschützten Projektdaten

Wenn Sie den Know-how-Schutz eines Objekts entfernen und das TIA Portal vor dem nächsten Speichern unerwartet beendet wird, werden die entsprechenden Projektdaten ungeschützt im „CrashDumpReport“ gespeichert und könnten nicht autorisierten Personen zugänglich sein.

Um den Zugriff auf eigentlich geschützte Daten zu unterbinden, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Verzeichnis ..\Users\<user_name>\AppData\Roaming\Siemens \Automation\<tia_version>\UserDataDumps\<project_name>
2. Löschen Sie alle Dateien mit der Namenserweiterung “*.UDD“.

Siehe auch

Projektdaten drucken (Seite 64)

3.2 Zugriffsrechte für Geräte zurücknehmen

Voraussetzung

- Für das Gerät wurde eine Schutzstufe eingestellt.
- Eine geschützte Funktion für das Gerät wurde über eine Passworteingabe freigegeben.

Vorgehen

Um die Zugriffsrechte für das Gerät wieder zurückzunehmen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, für das Sie die Zugriffsrechte zurücknehmen möchten.
2. Wählen Sie im Menü "Online" den Befehl "Zugriffsrechte löschen".

Ergebnis

Die Zugriffsrechte werden zurückgenommen und für alle passwortgeschützten Funktionen des Geräts wird ab jetzt wieder der Dialog zur Passworteingabe geöffnet. Die Funktion kann nur durch die Eingabe des korrekten Passworts ausgeführt werden.

Falls für das Gerät eine Online-Verbindung besteht, wird die Online-Verbindung getrennt.

Siehe auch

Schutzkonzept für Projektdaten (Seite 43)

Projektinhalte drucken 4

4.1 Projektdokumentation drucken

4.1.1 Dokumentationsfunktion

Einführung

Nach dem Erstellen eines Projekts lassen sich die Inhalte in übersichtlicher Form drucken. Sie können entweder das gesamte Projekt drucken oder einzelne Objekte. Ein sinnvoll strukturierter Ausdruck erleichtert sowohl die weitere Bearbeitung des Projekts als auch Service-Arbeiten. Der Ausdruck kann ebenso als Präsentation für Ihre Kunden dienen oder als komplette Anlagendokumentation.

Sie können das Projekt in Form von standardisierten Schaltbüchern aufbereiten und mit einem einheitlichen Layout ausdrucken. Den Umfang des Drucks können Sie eingrenzen. Entweder Sie drucken das gesamte Projekt, einzelnen Objekten mit ihren Eigenschaften oder einen kompakten Überblick des Projekts. Außerdem können Sie die Inhalte eines geöffneten Editors drucken.

Den Ausdruck mit Rahmen und Deckblättern verbessern

Sie haben die Möglichkeit, das Aussehen des Drucks nach eigenen Vorgaben zu gestalten, z. B. um Ihr eigenes Firmenlogo einzufügen oder um das Corporate Design Ihres Unternehmens in der Projektdokumentation umzusetzen. Sie können beliebig viele Design-Varianten als Rahmen und Deckblätter anlegen. Die Rahmen und Deckblätter werden in der Projektnavigation unter dem Eintrag "Dokumentationseinstellungen" abgelegt und sind Teil des Projekts. Innerhalb der Rahmen und Deckblätter können Sie Platzhalter für Daten aus zuvor hinterlegten Dokument-Informationen einfügen. Diese werden beim Druck automatisch mit den entsprechenden Meta-Daten befüllt.

Möchten Sie auf die freie Gestaltung verzichten, stehen fertige Rahmen und Deckblätter zur Verfügung. Darunter befinden sich auch Vorlagen nach dem ISO-Standard für technische Produktdokumentation.

Modularer Aufbau eines Ausdrucks

Ein Ausdruck besteht allgemein aus folgenden Bestandteilen:

- Deckblatt (nur beim Druck aus der Projektnavigation)
- Inhaltsverzeichnis (nur beim Druck aus der Projektnavigation)
- Name und Pfad eines Objekts innerhalb der Projektnavigation
- Objektdaten

Der Ausdruck des Deckblattes oder des Inhaltsverzeichnisses kann im Dialog "Drucken" deaktiviert werden.

Siehe auch

Rahmen anlegen (Seite 53)

Deckblatt anlegen (Seite 54)

Deckblätter und Rahmen bearbeiten (Seite 55)

Dokument-Informationen eingeben (Seite 51)

Druckfunktion für Modulbeschriftungsstreifen (Seite 67)

4.1.2 Ausdruck von Projektinhalten

Verfügbarkeit der Druckfunktion

Folgende Inhalte können gedruckt werden:

- Ein gesamtes Projekt in der Projektnavigation
- Ein oder mehrere einem Projekt zugehörige Objekte in der Projektnavigation
- Inhalte eines Editors
- Tabellen
- Bibliotheken
- Diagnosesicht des Inspektorfensters

In folgenden Bereichen ist der Druck nicht möglich:

- Portalansicht
- Detailsicht
- Übersichtsfenster
- Vergleichseditor
- Alle Register des Inspektorfensters außer der Diagnosesicht
- Alle Task Cards außer den Bibliotheken
- Überwiegende Zahl der Dialoge
- Nicht zum Projekt gehörige Eigenschaften und Geräte des PG/PC, wie z. B. die Online-Zugänge und angeschlossene Card Reader.

Umfang des Ausdrucks

Um drucken zu können, muss mindestens ein druckbares Element selektiert sein.

Wird ein selektiertes Objekt gedruckt, so werden alle unterlagerten Objekte mitgedruckt. Wird beispielsweise ein Gerät in der Projektnavigation selektiert, werden alle dazugehörigen Daten

ebenfalls mit ausgedruckt. Wenn Sie das gesamte Projekt in der Projektnavigation für den Druck auswählen, dann werden alle Projektinhalte außer den grafischen Sichten gedruckt. Diese müssen separat gedruckt werden. Einträge der Projektnavigation, die nicht zum Projekt gehören, können nicht gedruckt werden. Dazu zählen beispielsweise die Online-Zugänge sowie die angeschlossenen Card Reader und USB-Speicher.

Beim Drucken von Tabelleninhalten werden alle Zeilen einer Tabelle gedruckt, in denen eine Zelle markiert ist. Um eine oder mehrere Tabellenspalten zu drucken, müssen die gewünschten Spalten markiert sein. Sind keine einzelnen Zellen oder Spalten markiert, wird die gesamte Tabelle gedruckt.

Beschränkungen beim Druck

Generell können alle Objekte gedruckt werden, die auch an der Benutzeroberfläche sichtbar gemacht werden können. Umgekehrt heißt dies, dass Objekte, auf die Sie keinen Zugriff haben, auch nicht gedruckt werden können. Mögliche Gründe für das Scheitern eines Drucks können beispielsweise sein:

- Es ist keine gültige Lizenz zur Anzeige eines Objekts vorhanden.
- Es liegt keine Gerätebeschreibung für ein Objekt vor.
- Eine notwendige Softwarekomponente zur Anzeige eines Objekts ist nicht installiert.

Siehe auch

Projektdaten drucken (Seite 64)

4.1.3 Einstellungen für den Druck ändern

Einstellungen für den Druck ändern

Sie können generelle Einstellungen für den Druck vornehmen, welche auch nach dem Schließen und erneuten Öffnen des TIA Portals erhalten bleiben. Einige Einstellungen sind abhängig von den installierten Produkten. Folgende Einstellungen sind jedoch immer möglich:

Tabellendaten immer in Wertpaaren drucken

Ist diese Funktion aktiviert, werden Tabellen nicht in tabellarischer Form gedruckt, sondern als Paar zwischen Schlüssel und Wert.

Beispiel:

Objektname	Eigenschaft 1	Eigenschaft 2
Objekt A	Wert A1	Wert A2
Objekt B	Wert B1	Wert B2

In diesem Fall sieht der Ausdruck wie folgt aus:

Objekt A

Eigenschaft 1: Wert A1

Eigenschaft 2: Wert A2

Objekt B

Eigenschaft 1: Wert B1

Eigenschaft 2: Wert B2

Maskeneditoren drucken

- Daten immer in Tabellen drucken
  Alle Parameter von Technologieobjekten werden in Form einer Tabelle gedruckt.
- Maskengrafiken drucken, wenn möglich
  Wenn der verwendete Editor diese Funktion unterstützt, werden die Inhalte des Editors nicht als Tabelle, sondern als vollständige Grafik wie auf dem Bildschirm gedruckt.

Vorgehen

Um die Druckeinstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein".
3. Aktivieren Sie im Bereich "Druckeinstellungen" die gewünschten Voreinstellungen.
   Die Änderungen werden sofort übernommen und bleiben auch nach dem Schließen des TIA Portals projektübergreifend erhalten.

4.1.4 Festlegen des Drucklayouts

Drucklayout festlegen

Wenn Sie nicht auf vorgefertigte Druckvorlagen zurückgreifen möchten, können Sie selbst ein Deckblatt oder das Layout der einzelnen Seiten festlegen. Ihre Entwürfe werden zusammen mit dem jeweiligen Projekt gespeichert.

Unter der Gruppe "Dokument-Information" in der Projektnavigation finden Sie Ihre Entwürfe für das Deckblatt und Vorlagen für das Seitenlayout. Darüber hinaus finden Sie dort unter dem Eintrag "Dokument-Information" Meta-Daten zum Projekt. Beim anschließenden Druck können Sie im Dialog "Drucken" aus den gespeicherten Deckblättern, Vorlagen für das Seitenlayout und den Meta-Daten ein individuelles Erscheinungsbild zusammenstellen.

Deckblatt gestalten

Das Deckblatt lässt sich individuell gestalten. Sie können eine Hintergrundgrafik einfügen und Platzhalter für Text auf der Seite vorsehen. Die Platzhalter werden beim Druck automatisch mit Daten aus Dokument-Informationen befüllt.

Die Deckblätter befinden sich in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter".

Inhaltsseite gestalten

Die laufenden Seiten eines Ausdrucks können folgende Elemente enthalten:

- Rahmen mit statischem Inhalt, z. B. Firmenlogo
- Platzhalter für Text, z. B. den Namen des Projekts, die Seitennummer oder den Startzeitpunkt des Drucks
  Etliche Werte für die einzelnen Platzhalter können in einer Dokument-Information festgelegt werden. Andere Werte, wie z. B. der Projektname, sind vorgegeben und werden automatisch beim Druck eingefügt.
- Fußnote
  Die Fußnote wird immer unterhalb des Bereichs für den Inhalt ausgegeben.
- Bereich für den Inhalt
  Sie können einen Bereich festlegen, innerhalb dessen der Druckinhalt eingebettet wird.

Die Gestaltung der Inhaltsseiten wird in Rahmen gespeichert. Die einzelnen Rahmen befinden sich in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen".

4.1.5 Dokument-Informationen eingeben

Sie können zu jedem Projekt Meta-Daten in Dokument-Informationen eingeben. Außerdem werden in den Dokument-Informationen ein Rahmen für den Druck und ein Deckblatt festgelegt. Sie können bei Bedarf verschiedene Dokument-Informationen anlegen, um beim Druck schnell zwischen verschiedenen Dokument-Informationen mit unterschiedlichen Informationen, Rahmen, Deckblättern, Seitengrößen und Seitenausrichtungen wechseln zu können. Dies ist beispielsweise auch dann hilfreich, wenn Sie in verschiedenen Sprachen drucken möchten und für jede Sprache unterschiedliche Dokument-Informationen vorgesehen haben.

Im Dokumentationseditor können Sie Platzhalter auf dem Deckblatt oder im Rahmen der laufenden Seiten festlegen. Diese Platzhalter können beim Druck automatisch mit den Meta-Daten aus Dokument-Informationen befüllt werden.

Die verschiedenen Dokument-Informationen sind somit ein Bestandteil der Druckfunktion und bestimmen das Drucklayout sowie die Druckinhalte.

Vorgehen

Um Meta-Daten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter "Dokument-Information > Dokument-Information" auf "Neue Dokument-Information hinzufügen", um neue Dokument-Informationen anzulegen.
   Die neuen Dokument-Informationen werden angelegt und sofort geöffnet.
2. Geben Sie im Feld "Name" einen Namen für das Set ein.
3. Befüllen Sie die einzelnen Felder mit den Meta-Daten zum Projekt.

4.1.6 Deckblätter und Rahmen verwalten

4.1.6.1 Einsetzen von Deckblättern und Rahmen

Nutzen von Deckblättern

Sie können dem Druck Ihrer Anlagendokumentation ein professionelles Aussehen verleihen, indem Sie ihr ein Deckblatt voranstellen. Das Deckblatt können Sie frei gestalten oder auf vorgefertigte Deckblätter zurückgreifen. Vorgefertigte Deckblätter lassen sich anpassen und ebenfalls wieder als Vorlage ablegen.

Deckblätter können in globalen Bibliotheken gespeichert werden und stehen so projektübergreifend zur Verfügung.

Deckblätter sind immer als rechte Druckseite vorgesehen.

Nutzen von Rahmen

Die laufenden Seiten der Anlagendokumentation können Sie in einen stets gleichbleibenden Seitenrahmen einbetten. Der Rahmen kann Platzhalter für Meta-Daten zum Projekt enthalten, die in Dokument-Informationen gespeichert sind. Er kann aber auch frei gestaltbare grafische Elemente enthalten.

Sie können eigene Rahmen anlegen, aber auch auf vorgefertigte Seitenrahmen zurückgreifen. Vorgefertigte Seitenrahmen können Sie anpassen und im Anschluss selbst wieder als neuen Rahmen abspeichern.

Rahmen können genauso wie Deckblätter in globalen Bibliotheken gespeichert werden und stehen so projektübergreifend zur Verfügung.

Rahmen sind immer als rechte Druckseiten vorgesehen.

Deckblätter und Vorlagen in der Projektnavigation

Zum Projekt gehörige Deckblätter und Rahmen werden in der Projektnavigation unter dem Eintrag "Dokument-Information" abgelegt. Darunter liegen jeweils Ordner für Rahmen und Deckblätter.

Folgende Operationen stehen Ihnen für Deckblätter und Rahmen in der Projektnavigation zur Verfügung:

- Erstellen eigener Unterordner
- Kopieren und Einfügen
- Einfügen von Deckblättern und Rahmen aus der Systembibliothek "Dokumentationsvorlagen"
- Kopieren von Deckblättern und Vorlagen in eine globale Bibliothek

Deckblätter und Vorlagen in Bibliotheken

In der Systembibliothek "Dokumentationsvorlagen" sind einige Deckblätter und Vorlagen gespeichert, die in jedem Projekt zur Verfügung stehen. Die Deckblätter und Vorlagen lassen sich von dort per Drag & Drop in die Projektnavigation ziehen. In der Projektnavigation können Sie die Deckblätter und Vorlagen anschließend auf das Projekt anpassen.

Deckblätter und Vorlagen lassen sich von der Projektnavigation in eine globale Bibliothek ziehen. Danach stehen sie in jedem Projekt zur Verfügung.

Siehe auch

Vorgefertigte Rahmen und Deckblätter nutzen (Seite 54)

Deckblätter und Rahmen gestalten (Seite 55)

4.1.6.2 Rahmen anlegen

Sie können zu jedem Projekt eine beliebige Anzahl von Rahmen anlegen. Die Rahmen werden in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen" abgelegt. Jeder Dokument-Information können Sie einen Rahmen zuweisen. Wenn Sie sich beim Druck für eine Dokument-Information entscheiden, wird der zugehörige Rahmen verwendet.

Vorgehen

Um einen neuen Rahmen anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen" auf den Eintrag "Neuen Rahmen hinzufügen".
   Der Dialog "Rahmen anlegen" wird geöffnet.
2. Geben Sie einen Namen für den Rahmen im Feld "Name" ein.
3. Wählen Sie die Papiergröße aus der Klappliste "Papierformat".
4. Wählen Sie in der Klappliste "Orientierung", ob die Seite im Hoch- oder Querformat angelegt werden soll.

Klicken Sie auf die Schaltfläche "Hinzufügen".

Ergebnis

Ein neuer Rahmen wird angelegt. Der Rahmen wird anschließend automatisch im Dokumentationseditor geöffnet und kann dort bearbeitet werden.

Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 55)

Deckblatt anlegen (Seite 54)

4.1.6.3 Deckblatt anlegen

Sie können zu jedem Projekt eine beliebige Anzahl von Deckblättern für den Ausdruck anlegen. Die Deckblätter werden in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter" abgelegt. Jeder Dokument-Information können Sie ein Deckblatt zuweisen. Wenn Sie sich beim Druck für bestimmte Dokument-Informationen entscheiden, wird das zugehörige Deckblatt verwendet.

Vorgehen

Um ein neues Deckblatt anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter" auf den Eintrag "Neues Deckblatt hinzufügen".
   Der Dialog "Neues Deckblatt hinzufügen" wird geöffnet.
2. Geben Sie einen Namen für das Deckblatt im Feld "Name" ein.
3. Wählen Sie die Papiergröße aus der Klappliste "Papierformat".
4. Wählen Sie in der Klappliste "Orientierung", ob die Seite im Hoch- oder Querformat angelegt werden soll.

Klicken Sie auf die Schaltfläche "Hinzufügen".

Ergebnis

Ein neues Deckblatt wird angelegt. Das Deckblatt wird anschließend automatisch im Dokumentationseditor geöffnet und kann dort bearbeitet werden.

Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 55)

Rahmen anlegen (Seite 53)

4.1.6.4 Vorgefertigte Rahmen und Deckblätter nutzen

Mit dem TIA Portal werden bereits einige Rahmen und Deckblätter mitgeliefert. Diese können Sie nach Ihren Wünschen ändern.

Vorgehen

Um mitgelieferte Rahmen und Deckblätter einzufügen und zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Task Card "Bibliotheken" die Palette "Globale Bibliotheken".
2. Öffnen Sie im Ordner "Kopiervorlagen" den Ordner "Cover Pages" für Deckblätter oder "Frames" für Rahmen.

3. Ziehen Sie aus einem der beiden Ordner ein Deckblatt oder einen Rahmen per Drag & Drop in die Projektnavigation in einen der folgenden Ordner:
   – Für Rahmen: "Dokument-Information > Rahmen"
   – Für Deckblätter: "Dokument-Information > Deckblätter".

   Der vorgefertigte Rahmen oder das vorgefertigte Deckblatt kann jetzt im Projekt verwendet werden.
4. Doppelklicken Sie in der Projektnavigation auf den neuen Eintrag, um den Rahmen oder das Deckblatt zu bearbeiten.

Siehe auch

Einsetzen von Deckblättern und Rahmen (Seite 52)

Deckblätter und Rahmen bearbeiten (Seite 55)

4.1.7 Deckblätter und Rahmen gestalten

4.1.7.1 Deckblätter und Rahmen bearbeiten

Der Dokumentationseditor ist ein grafischer Editor und dient der Gestaltung von Rahmen und Deckblättern für die Anlagendokumentation. Im Dokumentationseditor können Sie Bilder oder Textelemente auf den Rahmen und Deckblättern platzieren. Die Textelemente sind entweder statisch, oder sie werden beim Druck automatisch mit den Daten aus den Dokument-Informationen befüllt, die Sie im Druck-Dialog gewählt haben.

Vorgehen

Um ein Deckblatt oder einen Rahmen im Dokumentationseditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation in der Gruppe "Dokument-Information > Rahmen" oder "Dokument-Information > Deckblätter" auf den Eintrag für ein bereits vorhandenes Deckblatt oder einen Rahmen.
   Der Dokumentationseditor wird geöffnet.
2. Gestalten Sie das Deckblatt oder den Rahmen entsprechend Ihren Vorstellungen.
3. Schließen Sie den Dokumentationseditor.
   Die Änderungen am Deckblatt oder am Rahmen werden automatisch übernommen.

Siehe auch

Deckblatt anlegen (Seite 54)

Rahmen anlegen (Seite 53)

Allgemeine Bedienung des Dokumentationseditors (Seite 56)

4.1.7.2 Allgemeine Bedienung des Dokumentationseditors

Komponenten des Dokumentationseditors

Das folgende Bild zeigt einen Überblick über die Komponenten des Dokumentationseditors:

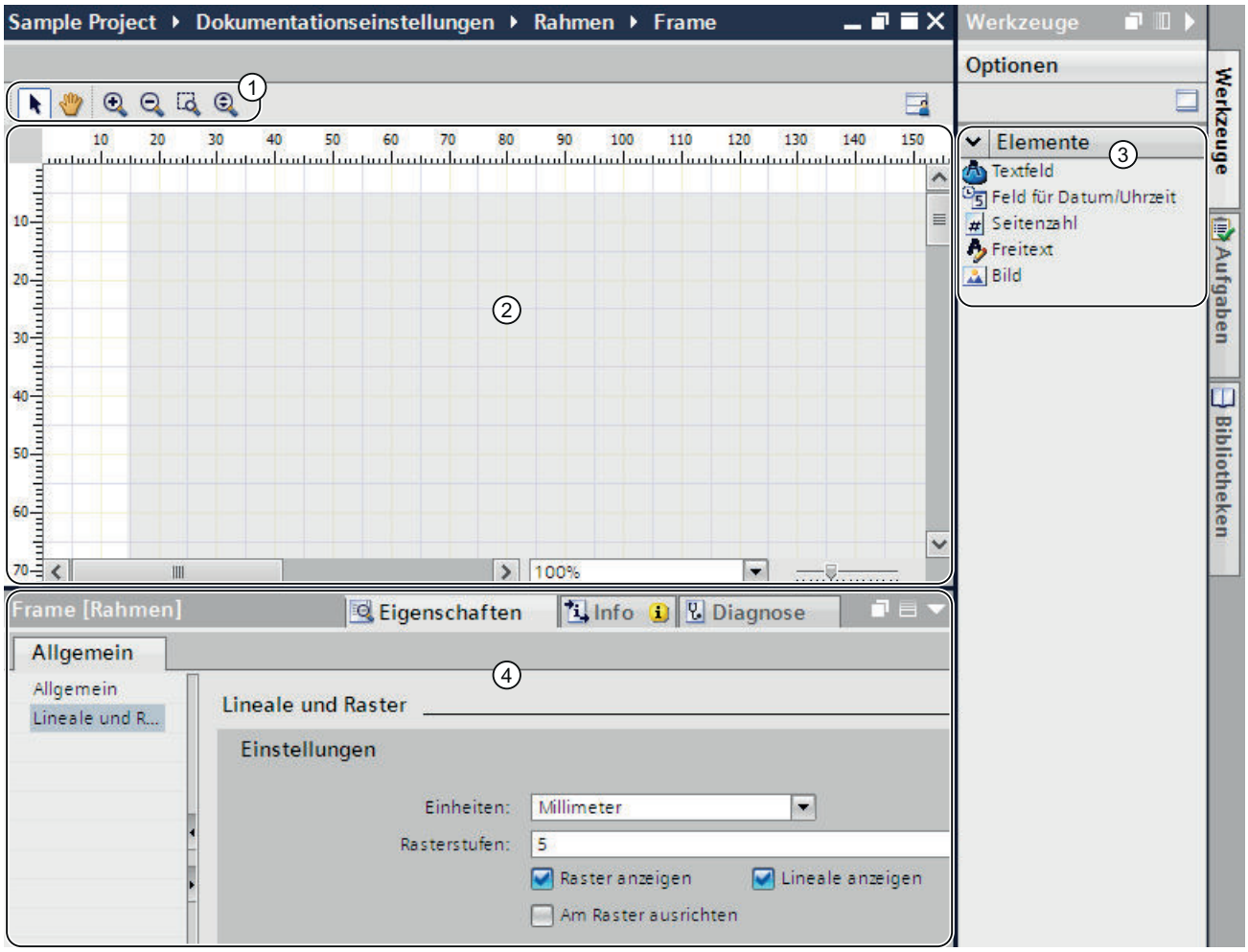

① Funktionsleiste

In der Funktionsleiste finden Sie folgende Werkzeuge (von links nach rechts):

- Pfeil-Werkzeug
  Ermöglicht das Selektieren eines Objekts.
- Navigationswerkzeug
  Erlaubt das Verschieben des Seitenausschnitts.
- Vergrößern-Schaltfläche
  Vergrößert die Seitendarstellung schrittweise.
- Verkleinern-Schaltfläche
  Verkleinert die Seitendarstellung schrittweise.
- Zoomfaktor auswählen
  Passt die Seitengröße an einen ausgewählten Arbeitsbereich an.
- Dynamischer Zoom
  Passt die Seitenbreite an den Arbeitsbereich an.

② Arbeitsbereich

Im Arbeitsbereich können Sie das Deckblatt oder den Rahmen gestalten.

③ Task Card "Werkzeuge"

In der Task Card "Werkzeuge" stehen die verschiedenen Typen von Platzhaltern, die Sie auf dem Deckblatt oder dem Rahmen verwenden können, zur Verfügung. Die Platzhalter können per Drag & Drop im Arbeitsbereich platziert werden.

④ Eigenschaften im Inspektorfenster

Im Register "Eigenschaften" des Inspektorfensters können Sie die Eigenschaften des aktuell selektierten Objekts anzeigen und ändern. Sie können beispielsweise die Eigenschaften der Seite ändern, Text formatieren, die Position von Objekten auf der Seite genau festlegen, usw.

Bedienung im Dokumentationseditor

Der Dokumentationseditor besitzt die folgenden grundlegenden Funktionen:

- Drag & Drop-Funktionalität
  Der Dokumentationseditor ist ein grafischer Editor. Das bedeutet, Sie können Objekte frei mit der Maus platzieren. Im Arbeitsbereich wird ein Abbild der Seite gezeigt, welches identisch mit dem späteren Drucklayout ist.
  Möchten Sie Objekte auf der Seite selektieren, um sie zu verschieben oder die Eigenschaften der Objekte zu ändern, muss das Pfeil-Werkzeug in der Funktionsleiste aktiviert sein.
- Zoomfunktion
  Mithilfe der Zoomfunktion können Sie die Darstellungsgröße der Seite ändern. Sie haben zwei Möglichkeiten, um die Seitengröße anzupassen:
  - Über die Symbole in der Funktionsleiste
    Aktivieren Sie das Lupen-Symbol "Zoomfaktor vergrößern" oder "Zoomfaktor verkleinern" in der Funktionsleiste des Dokumentationseditors. Klicken Sie anschließend auf die Seite, um sie stufenweise zu vergrößern oder zu verkleinern.
    Um einen bestimmten Bereich zu vergrößern, wählen Sie das Werkzeug "Zoomfaktor auswählen" und ziehen Sie mit der Maus einen Rahmen um den Bereich, den Sie in den Fokus rücken möchten.
    Um den Arbeitsbereich stufenlos zu vergrößern oder zu verkleinern, verwenden Sie das Werkzeug "Dynamischer Zoom". Klicken Sie auf eine Stelle im Arbeitsbereich und ziehen Sie mit gedrückter Maustaste die Maus nach oben, um die Seitendarstellung zu vergrößern. Um die Seitendarstellung zu verkleinern, ziehen Sie die Maus nach unten.
  - Über die Zoom-Leiste
    Über die Zoomleiste in der rechten unteren Ecke des Arbeitsbereichs können Sie ebenfalls die Darstellungsgröße ändern. Wählen Sie einen Prozentwert aus der Klappliste aus oder geben Sie einen Prozentwert ein. Alternativ können Sie die Darstellungsgröße über den Schieberegler steuern.
- Navigation über die Seite
  Neben dem Bildlauf gibt es die Möglichkeit, mit dem Navigationswerkzeug den Seitenausschnitt zu verändern. Um den Ausschnitt mit dem Navigationswerkzeug zu ändern, aktivieren Sie das Hand-Symbol in der Funktionsleiste. Klicken Sie anschließend mit der Maus auf die Seite, halten Sie die Maustaste gedrückt und verschieben Sie die Seite in die gewünschte Position.

Positionierungshilfsmittel nutzen und anpassen

Für die Positionierung von Elementen auf der Seite stehen Ihnen verschiedene Hilfsmittel zur Verfügung:

- Lineale
  An den Seitenrändern des Arbeitsbereichs sind Lineale angebracht.
- Seitenraster
  Der Seitenhintergrund im Arbeitsbereich ist mit einem Raster hinterlegt.

Sie können die Positionierungshilfsmittel im Inspektorfenster unter "Eigenschaften > Lineale und Raster" ein- und ausblenden oder anpassen. Sie haben folgende Einstellmöglichkeiten:

- Einheiten:
  Legen Sie die Maßeinheit für das Raster und für die Lineale fest.
- Rasterstufen:
  Legen Sie die Weite des Rasters fest.
- Raster anzeigen:
  Bestimmen Sie, ob das Raster angezeigt oder ausgeblendet wird.
- Am Raster ausrichten:
  Legen Sie fest, ob Objekte automatisch am Raster ausgerichtet werden. Ist die Option aktiviert, bekommen die Rasterlinien eine "magnetische" Funktion.
- Lineale anzeigen:
  Bestimmen Sie, ob die Lineale angezeigt werden.

Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 55)

Druckfläche festlegen (Seite 59)

Platzhalter für Meta-Daten einfügen (Seite 60)

4.1.7.3 Druckfläche festlegen

Innerhalb eines Rahmens ist ein Bereich für die eigentlichen Druckinhalte vorgesehen. Die Projektdaten werden dann innerhalb des festgelegten und stets gleich bleibenden Bereichs innerhalb des Rahmens eingefügt. Sie können die Größe des Druckbereichs anpassen.

Voraussetzung

Ein Rahmen ist im Dokumentationseditor geöffnet.

Vorgehen

Um einen Bereich für die Druckinhalte festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf etwas dunkler gehaltene Fläche innerhalb der Seitendarstellung im Dokumentationseditor, um den Bereich für die Druckinhalte zu selektieren.
   Die Eigenschaften der Druckfläche werden im Inspektorfenster geöffnet.
2. Geben Sie im Inspektorfenster die Position der Druckfläche auf der X- und der Y-Achse ein.
3. Geben Sie im Inspektorfenster die Breite und die Höhe der Druckfläche in cm an.

Alternativ können Sie die Breite und Position des Druckfeldes in der grafischen Darstellung der Seite ändern. Ziehen Sie dazu mit der Maus die Ränder der Druckfläche in die gewünschte Größe und Position.

Siehe auch

Rahmen anlegen (Seite 53)

Allgemeine Bedienung des Dokumentationseditors (Seite 56)

4.1.7.4 Platzhalter für Meta-Daten einfügen

Sie können auf dem Deckblatt und in einem Rahmen Platzhalter vorsehen. Die Platzhalter werden beim Druck automatisch mit Meta-Daten aus Dokument-Informationen befüllt, sofern es sich um Platzhalter für Text handelt. Alternativ können Sie unveränderliche Daten, wie beispielsweise Freitext oder ein Bild einfügen.

Alle Elemente sind in nummerierten Ebenen angeordnet. Wenn Objekte sich überlagern, können sie festlegen, in welcher Reihenfolge diese angeordnet sind.

Typen von Platzhaltern

Folgende Typen von Platzhaltern stehen zur Verfügung:

- Textfeld
  Das Textfeld steht als Platzhalter für ein Textelement aus einer Dokument-Information. Welcher Text aus einer Dokument-Information automatisch beim Druck eingesetzt werden soll, stellen Sie in den Eigenschaften des Textfeldes ein.
- Feld für Datum und Uhrzeit
  Anstelle des Platzhalters werden beim Druck ein Datum und eine Uhrzeit eingefügt. Dies kann beispielsweise das Erstellungsdatum sein, oder auch der Zeitpunkt der letzten Änderung am Projekt. In den Eigenschaften im Inspektorfenster legen Sie fest, um welches Datum oder welche Uhrzeit es sich handelt.
- Seitenzahl
  Beim Druck wird automatisch die entsprechende Seitenzahl eingesetzt.
- Freitext
  In den Eigenschaften des Textfeldes können Sie frei wählbaren Text eingeben. Der Text ist statisch und wird nicht von der beim Druck gewählten Dokument-Information beeinflusst.
- Bild
  Wählen Sie die Bilddatei in den Eigenschaften des Platzhalters im Inspektorfenster aus. Bilder in den Formaten BMP, JPEG, PNG, EMF oder GIF sind möglich.

Voraussetzung

Ein Deckblatt oder ein Rahmen ist im Dokumentationseditor geöffnet.

Vorgehen

Um Platzhalter für Meta-Daten auf dem Deckblatt oder in einen Rahmen einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus der Task Card "Werkzeuge > Elemente" per Drag & Drop ein Feld in den Arbeitsbereich des Dokumentationseditors.
   Der Platzhalter wird eingefügt. Die Eigenschaften des Platzhalters werden im Inspektorfenster angezeigt und können dort bearbeitet werden.
2. Wählen Sie aus der Klappliste "Text" im Inspektorfenster unter "Eigenschaften > Allgemein > Textfeld" das Meta-Datum aus, welches beim Druck eingefügt werden soll. Oder geben Sie alternativ Freitext ein, oder wählen Sie ein Bild, je nachdem um welche Art von Platzhalter es sich handelt.
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Position und Größe" die Position des Platzhalters auf der X- und Y-Achse fest und geben Sie die Breite und Höhe des Textfeldes in cm ein. Bestimmen Sie im Feld "Ebene" die Reihenfolge der Objekte, falls diese sich überlagern. Je kleiner der Wert, desto weiter hinten befindet sich ein Objekt.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Ansicht" die Schriftformatierung und die Orientierung des Textes sowie die Ausrichtung der Schrift. Bei Bildern können Sie diese Einstellung nicht vornehmen.

Siehe auch

Allgemeine Bedienung des Dokumentationseditors (Seite 56)

4.1.8 Druckvorschau anzeigen

4.1.8.1 Druckvorschau erstellen

Druckvorschau erstellen

Sie können eine Vorschau des Ausdrucks erstellen. Ebenso wie beim tatsächlichen Druck sind dafür Dokument-Informationen wählbar. So sehen Sie in der Vorschau bereits den gewählten Rahmen und gegebenenfalls das Deckblatt. Die Einstellungen bleiben für den späteren Druck erhalten.

Vorgehen

Um eine Druckvorschau zu erstellen und den Umfang des späteren Ausdrucks festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Druckvorschau".
   Der Dialog "Druckvorschau" wird geöffnet.
2. Wählen Sie das Rahmen-Layout, das Sie für den Ausdruck verwenden möchten.
   - Wählen Sie in der Klappliste "Dokument-Information" die Dokumentations-Informationen aus, die für den späteren Druck herangezogen werden sollen.
   - Aktivieren Sie das Optionskästchen "Deckblatt drucken", um das Deckblatt zu drucken, welches im gewählten Set von Dokument-Informationen festgelegt ist.
   - Aktivieren Sie das Optionskästchen "Inhaltsverzeichnis drucken", um dem Ausdruck ein Inhaltsverzeichnis voranzustellen.

   Die Optionskästchen zum Drucken des Deckblatts und des Inhaltsverzeichnisses sind nur aktivierbar, wenn Sie den Ausdruck über die Projektnavigation gestartet haben.
3. Wählen Sie unter "Objekte/Bereich drucken" aus, was gedruckt werden soll. Die Auswahl ist nur möglich, wenn Sie den Druck aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
   - Wählen Sie "Alle", um alle Inhalte des Editors auszudrucken.
   - Wählen Sie "Auswahl", um nur die aktuell im Editor selektierten Objekte zur drucken.
4. Wählen Sie den Druckumfang unter "Eigenschaften" aus.
   - Wählen Sie "Alle", um alle Konfigurationsdaten der selektierten Objekte zu drucken.
   - Wählen Sie "Sichtbar", um alle gerade am Bildschirm sichtbaren Informationen innerhalb eines Editors zu drucken. Die Option ist nur wählbar, wenn Sie den Druckauftrag aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
   - Wählen Sie "Kompakt", um eine verkürzte Ausgabe der Projektdaten auszudrucken.
5. Klicken Sie auf "Vorschau", um die Vorschau zu erzeugen.
   Im Arbeitsbereich wird eine Druckvorschau erstellt.

Hinweis

Wartezeit bei umfangreichen Dokumenten

Bei besonders umfangreichen Projekten kann das Erstellen der Druckvorschau bis zu mehreren Minuten dauern. Sie können auf ausreichend leistungsfähigen Systemen zwischenzeitlich normal weiterarbeiten. In der Statusleiste wird der Fortschritt der Druckvorschau angezeigt.

Siehe auch

Bedienung innerhalb der Druckvorschau (Seite 63)

4.1.8.2 Bedienung innerhalb der Druckvorschau

Funktionen innerhalb der Druckvorschau

Die Druckvorschau zeigt ein genaues Abbild des späteren Drucks an. Über die Symbole in der Funktionsleiste lässt sich die Anzeige der Druckvorschau ändern. Folgende Funktionen stehen zur Verfügung (von links nach rechts):

- Navigationsmodus
  Erlaubt das Verschieben des Seitenausschnitts.
  Um den Ausschnitt mit dem Navigationswerkzeug zu ändern, aktivieren Sie das Pfeil-Symbol. Klicken Sie anschließend mit der Maus auf die Seite, halten Sie die Maustaste gedrückt und verschieben Sie die Seite in die gewünschte Position.
- Zoomfunktion
  - "Vergrößern" und "Verkleinern"
    Vergrößert oder verkleinert die Seitendarstellung.
    Um die Darstellung schrittweise zu vergrößern oder zu verkleinern, aktivieren Sie das entsprechende Symbol. Klicken Sie anschließend auf die Seite, um sie stufenweise zu vergrößern oder zu verkleinern.
    Um einen bestimmten Bereich zu vergrößern, aktivieren Sie das Symbol "Zoomfaktor auswählen" und ziehen Sie mit der Maus einen Rahmen um den Bereich, den Sie in den Fokus rücken möchten.
    Um dynamisch durch die Seite zu zoomen, aktivieren Sie das Symbol "Dynamisch vergrößern/verkleinern". Ziehen Sie anschließend mit gedrückter Maustaste die Maus nach unten über die Seite, um die Darstellung zu verkleinern. Ziehen Sie stattdessen nach oben, um die Darstellung zur vergrößern.
  - Prozentwert in der Klappliste
    Legt die Darstellungsgröße der Seite in Prozent fest.
    Geben Sie einen Prozentwert ein oder wählen Sie einen Prozentwert aus der Klappliste aus. Alternativ wählen Sie die Optionen "An Seite anpassen" aus der Klappliste, um die Seitengröße an den Arbeitsbereich anzupassen. Oder wählen Sie "An Breite anpassen", um die Seitenbreite an den Arbeitsbereich anzupassen.

- "Vorwärts" und "Rückwärts"
  Jede Änderung des Seitenausschnitts, der Seitenzahl oder der Darstellungsgröße wird in einer Historie im Hintergrund gespeichert. Mit der Schaltfläche "Vorwärts" oder "Rückwärts" können Sie jeweils zur vorigen oder zur nächsten Ansicht zurückkehren.
- Seitennavigation
  - "Erste Seite"
    Springt zurück zur ersten Seite.
  - "Vorherige Seite"
    Geht eine Seite zurück.
  - Eingabefeld "Seitenzahl"
    Zeigt die aktuelle Seite an. Um direkt zu einer Seite zu springen, geben Sie die gewünschte Seitenzahl ein.
  - "Nächste Seite"
    Geht zur nächsten Seite.
  - "Letzte Seite"
    Springt zur letzten Seite.

Siehe auch

Druckvorschau erstellen (Seite 61)

4.1.9 Projektdaten drucken

Wenn Sie Projektdaten ausdrucken möchten, stehen Ihnen zwei Möglichkeiten zur Verfügung:

- Sofort-Druck mit Standardeinstellungen über die Schaltfläche "Drucken" in der Funktionsleiste.
  Die Schaltfläche ist nur aktiv, wenn ein druckbares Objekt selektiert ist.
- Ausdruck über den Menübefehl "Projekt > Drucken" mit erweiterten Einstellmöglichkeiten. Sie können beispielsweise einen anderen Drucker wählen, bestimmte Dokumentations-Informationen wählen, oder bestimmen, ob ein Deckblatt und Inhaltsverzeichnis gedruckt werden sollen. Außerdem können Sie den Druckumfang festlegen oder sich vor dem Ausdruck eine Druckvorschau anzeigen lassen.

Voraussetzung

- Mindestens ein Drucker ist konfiguriert.
- Die zu druckenden Objekte sind nicht geschützt.
  Sind Objekte geschützt, dann ist der Druckumfang eingeschränkt. Um die Objekte vollständig zu drucken, heben Sie den Know-how-Schutz auf.

Projektdaten drucken

Um Daten des aktuellen Projekts oder das gesamte Projekt mit erweiterten Einstellmöglichkeiten auszudrucken, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das gesamte Projekt, um das ganze Projekt auszudrucken. Um nur einzelne, einem Projekt zugehörige Elemente zu drucken, selektieren Sie diese in der Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Drucken".
   Der Dialog "Drucken" wird geöffnet.
3. Wählen Sie im Feld "Name" den Drucker aus.
4. Klicken Sie auf "Erweitert", um die Windows-Einstellungen des Druckers zu bearbeiten.
5. Wählen Sie das Rahmen-Layout, das Sie für den Ausdruck verwenden möchten.
   - Wählen Sie in der Klappliste "Dokument-Information" die Dokument-Informationen aus. Der in den Dokument-Informationen gespeicherte Rahmen wird für den Druck verwendet. Alle Platzhalter innerhalb des gewählten Rahmens werden mit den Meta-Daten aus den ausgewählten Dokument-Informationen befüllt.
   - Aktivieren Sie das Optionskästchen "Deckblatt drucken", um das Deckblatt zu drucken, welches in den gewählten Dokument-Informationen hinterlegt ist.
   - Aktivieren Sie das Optionskästchen "Inhaltsverzeichnis drucken", um dem Ausdruck ein Inhaltsverzeichnis voranzustellen.

   Die Optionskästchen zum Drucken des Deckblatts und des Inhaltsverzeichnisses sind nur aktivierbar, wenn Sie den Ausdruck über die Projektnavigation gestartet haben.
6. Wählen Sie unter "Objekte/Bereich drucken" aus, was gedruckt werden soll. Die Auswahl ist nur möglich, wenn Sie den Druck aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
   - Wählen Sie "Alle", um alle Inhalte des Editors auszudrucken.
   - Wählen Sie "Auswahl", um nur die aktuell im Editor selektierten Objekte zu drucken.
7. Wählen Sie den Druckumfang unter "Eigenschaften" aus.
   - Wählen Sie "Alle", um alle Konfigurationsdaten der selektierten Objekte zu drucken.
   - Wählen Sie "Sichtbar", um alle gerade am Bildschirm sichtbaren Informationen innerhalb eines Editors zu drucken. Die Option ist nur wählbar, wenn Sie den Druckauftrag aus einem Editor heraus gestartet haben.
   - Wählen Sie "Kompakt", um eine verkürzte Ausgabe der Projektdaten auszudrucken.
8. Klicken Sie auf "Vorschau", um vorab eine Druckvorschau zu erzeugen.
   Im Arbeitsbereich wird eine Druckvorschau erstellt.
9. Klicken Sie auf "Drucken", um den Ausdruck zu starten.

Hinweis

Umfang des Dialogs "Drucken"

Die angebotenen Optionen im Dialog "Drucken" variieren abhängig von den zu druckenden Elementen.

Ergebnis

Die Projektdaten werden im Hintergrund für den Druck vorbereitet und anschließend auf dem ausgewählten Drucker gedruckt. Der Fortschritt des Drucks wird in der Statusleiste angezeigt. Während der Druck vorbereitet wird können Sie normal weiterarbeiten.

Die Ergebnisse des Drucks sowie mögliche Fehler und Warnungen werden nach Abschluss des Druckauftrages im Inspektorfenster unter "Info" aufgelistet.

Druckauftrag abbrechen

Um einen laufenden Druckauftrag abzubrechen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Statusleiste auf das blaue Kreuz neben der Fortschrittsanzeige für den Druck.
   Der Ausdruck wird abgebrochen.

Siehe auch

Schutzkonzept für Projektdaten (Seite 43)

Zugriffsrechte für Geräte zurücknehmen (Seite 45)

Ausdruck von Projektinhalten (Seite 48)

Deckblätter und Rahmen gestalten (Seite 55)

4.2 Modulbeschriftungsstreifen drucken

4.2.1 Druckfunktion für Modulbeschriftungsstreifen

Drucken von Beschriftungsstreifen für Hardware-Module

Mithilfe des TIA Portals können Sie Beschriftungsstreifen für die Module Ihres Projekts drucken. Die Beschriftungsstreifen sind passgenau für die jeweils verwendeten Module und können mit den folgenden Informationen bedruckt werden:

- Symbolischer Name des Ein- oder Ausgangs
- Absolute Adresse des Ein- oder Ausgangs
- Symbolischer Name und zusätzlich die absolute Adresse des Ein- oder Ausgangs. Die Reihenfolge ist einstellbar.

In der Gerätesicht werden die Module grafisch dargestellt. Wenn Sie die Vergrößerung in der Gerätesicht auf mindestens 200 % einstellen, werden dort die Beschriftungen für die einzelnen Module sichtbar. Der Ausdruck auf dem Beschriftungsstreifen entspricht der Darstellung der Beschriftung in der Gerätesicht.

Das folgende Bild zeigt beispielhaft zwei Module in der Gerätesicht, an denen die Beschriftung der Ein- und Ausgänge sichtbar ist:

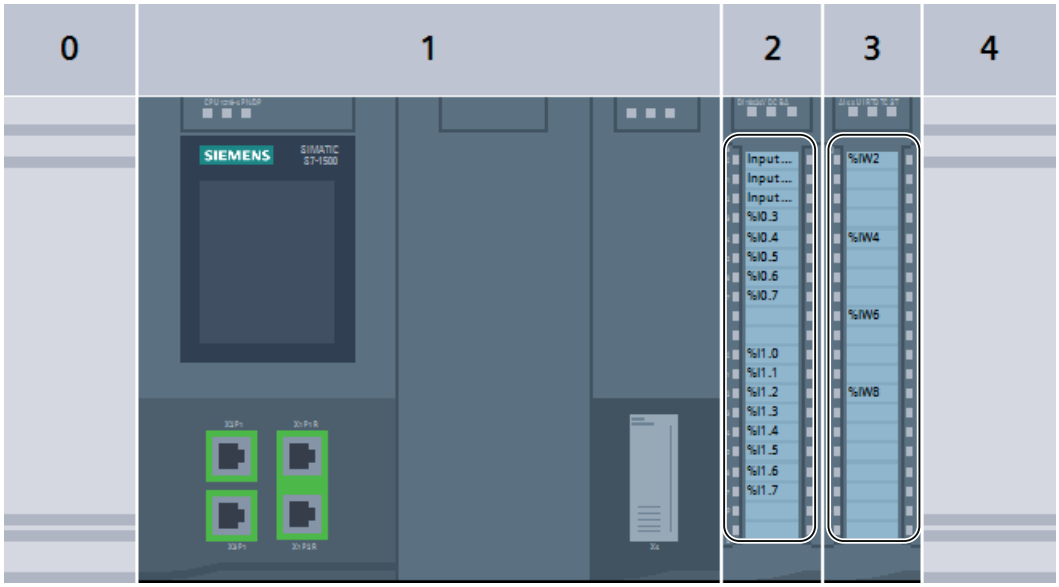

Export und Weiterbearbeitung als Microsoft Word-Datei

Vor dem Druck werden die Beschriftungsstreifen zunächst als Microsoft DOCX-Datei exportiert. Die Datei lässt sich mit gängigen Textverarbeitungsprogrammen, wie z. B. Microsoft Word, weiterbearbeiten. Die einzelnen Beschriftungsstreifen sind in der DOCX-Datei als Tabelle repräsentiert.

Standardmäßig wird die Laufweite des Texts innerhalb der Tabelle so angepasst, dass Texte nicht abgeschnitten werden. Wenn Sie verhindern möchten, dass der Text dadurch zu sehr gedehnt oder gestaucht wird, ändern Sie die Laufweite des Texts in den Eigenschaften der Tabellenzelle.

Ausgabe der Beschriftungsdaten in eine XML-Datei

Als Alternative zum Druck der Beschriftungsstreifen können Sie die Adressen der Ein- und Ausgänge eines Moduls in eine XML-Datei ausgeben. Den Export in eine XML-Datei nutzen Sie für Geräte, zu denen keine vorgefertigten Beschriftungsstreifen erhältlich sind. Ebenso können Sie den Export in eine XML-Datei nutzen, um die Beschriftungsstreifen mit einem anderen Programm zu erstellen. Das Programm muss die Daten der XML-Datei in ein für das Beschriftungssystem geeignetes Eingabeformat transformieren können. Das Schema der XML-Datei entnehmen Sie dem Kapitel "XML-Schema der Exportdatei (Seite 71)".

Druckmedien

Die Beschriftungsstreifen können Sie entweder auf vorgefertigte Druckbögen oder auf gewöhnlichem DIN-A4-Papier drucken. Aus den vorgefertigten Druckbögen können Sie die einzelnen Beschriftungsstreifen herausbrechen und in die vorgesehenen Beschriftungsfelder Ihrer Module stecken. Wenn Sie auf gewöhnlichem Papier drucken, schneiden Sie die einzelnen Beschriftungsstreifen selbst aus. Dabei helfen Ihnen Schnittmarken, die automatisch auf dem Ausdruck angebracht werden.

Da sich die Papiereinzüge an verschiedenen Druckern geringfügig unterscheiden, kann der Ausdruck leicht verschoben auf dem Papier sein. Beim Druck der Beschriftungsstreifen auf die vorgefertigten Druckbögen kommt es jedoch auf millimetergenauen Druck an, da der Text sonst nicht passgenau innerhalb des Stanzbereichs liegt. Außerdem kann bei ungenauem Druck die Beschriftung eines Ein- oder Ausgangs nicht mehr deckungsgleich mit den Kanalstatusanzeigen des Moduls sein. Um einen präzisen Druck zu gewährleisten, können Sie im TIA Portal einen Wert für die Verschiebung Ihres Druckers eingeben. Wie Sie den geeigneten Wert für die Verschiebung Ihres Druckers ermitteln, erfahren Sie im Kapitel "Verschiebung des Druckbereichs ermitteln (Seite 74)".

Siehe auch

Beschriftungsdaten als XML exportieren (Seite 70)

XML-Schema der Exportdatei (Seite 71)

Beschriftungsstreifen drucken (Seite 69)

Verschiebung des Druckbereichs ermitteln (Seite 74)

Dokumentationsfunktion (Seite 47)

4.2.2 Beschriftungsstreifen drucken

Sie können für die Module in Ihrem Projekt Beschriftungsstreifen drucken, wenn für die verwendeten Module das Anbringen von Beschriftungsstreifen vorgesehen ist. Die Beschriftungsstreifen werden zunächst in ein Microsoft Word DOCX-Dateien exportiert. Für jede Baugruppenfamilie wird eine eigene DOCX-Datei erstellt (z. B. für alle gewählten S7-1500-Module). Der Druck erfolgt schließlich aus dem Textverarbeitungsprogramm heraus.

Voraussetzung

Für den Druck von Beschriftungsstreifen gelten die folgenden Voraussetzungen:

- Die gewählten Module müssen den Druck von Beschriftungsstreifen unterstützen. Andernfalls ist nur die Ausgabe der Daten in eine XML-Datei möglich.
- Ein Textverarbeitungsprogramm muss installiert sein, das Microsoft Word DOCX-Dateien unterstützt, z. B. Microsoft Word 2010 oder aktueller.
- Sie benötigen die vorgefertigten Beschriftungsstreifen für Ihre Module oder handelsübliches DIN-A4-Papier.

Vorgehen

Um Beschriftungsstreifen für Hardware-Module zu drucken, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation die Module aus, für die Sie Beschriftungsstreifen drucken möchten.
   - Sie können ein oder mehrere Stationen auswählen, um für alle darin gesteckten Module Beschriftungsstreifen auszudrucken.
   - Oder wählen Sie die gewünschten Module unterhalb der Stationen im Ordner "Lokale Baugruppen" aus.
2. Klicken Sie mit der rechten Maustaste auf eines der Geräte und wählen Sie im Kontextmenü den Befehl "Beschriftungsstreifen exportieren".
   Der Dialog "Beschriftungsstreifen exportieren" wird geöffnet.
3. Wählen Sie im Bereich "Inhalt des Beschriftungsstreifens" aus, welche Daten auf den Beschriftungsstreifen gedruckt werden:
   - Wählen Sie "Symbolische Namen", um den symbolischen Namen des Ein- oder Ausgangs zu drucken (entspricht den Inhalten der Spalte "Name" in der IO-Variablentabelle).
   - Wählen Sie "Absolute Adresse", um die absolute Adresse des jeweiligen Ein- oder Ausgangs zu drucken (entspricht den Inhalten der Spalte "Adresse" in der IO-Variablentabelle).
   - Wählen Sie "Absolute und symbolische Adresse" oder "Symbolische und absolute Adresse", um beide Adressen zu drucken. Die Druckreihenfolge entspricht der genannten Reihenfolge.

4. Wählen Sie im Bereich "Exportformat" aus, wie die Beschriftungsdaten ausgegeben werden.
   - Wählen Sie "Druck auf einen SIEMENS-Beschriftungsbogen", wenn Sie auf einem vorgefertigten Beschriftungsbogen für Ihre Module drucken.
   - Wählen Sie "Druck auf leere A4-Seite", wenn Sie auf normales DIN-A4-Papier drucken.
5. Wählen Sie im Bereich "Verschiebung des Druckbereichs" bei Bedarf Korrekturwerte für Ihren Drucker. Die Korrekturwerte dienen zur korrekten Ausrichtung des Druckbereichs. Korrekturwerte sind nur notwendig, wenn Sie auf vorgefertigte Beschriftungsstreifen drucken.
   - Geben Sie in das Feld "Verschiebung vertikal" einen Korrekturwert in Millimeter ein. Ein negativer Wert verschiebt den Druckbereich nach oben. Ein positiver Wert verschiebt den Druckbereich nach unten.
   - Geben Sie in das Feld "Verschiebung horizontal" einen Korrekturwert in Millimeter ein. Ein negativer Wert verschiebt den Druckbereich nach links. Ein positiver Wert verschiebt den Druckbereich nach rechts.
6. Wählen Sie im Feld "Pfad" einen Pfad aus, an dem die exportierten Dateien abgelegt werden.
7. Klicken Sie auf die Schaltfläche "Export", um den Export zu starten.
   Die Exportdateien werden erstellt.
8. Öffnen Sie die DOCX-Dateien mit einem gängigen Textverarbeitungsprogramm, z. B. mit Microsoft Word, und ändern Sie bei Bedarf die Gestaltung der Beschriftungsstreifen.
9. Drucken Sie die Beschriftungsstreifen aus Ihrem Textverarbeitungsprogramm heraus aus. Verwenden Sie dazu das Papier, das Sie im Exportdialog angegeben haben.
10. Brechen Sie die Beschriftungsstreifen an den vorgesehenen Stanzungen aus, wenn Sie das vorgefertigte Papier verwendet haben. Bei handelsüblichem DIN-A4-Papier schneiden Sie die Beschriftungsstreifen aus.

Siehe auch

Verschiebung des Druckbereichs ermitteln (Seite 74)

Beschriftungsdaten als XML exportieren (Seite 70)

4.2.3 Beschriftungsdaten als XML exportieren

Das TIA Portal unterstützt eine große Zahl unterschiedlicher Module und ist z. B. über Hardware Support Packages ständig erweiterbar. Nicht für jedes der unterstützten Module sind vorgefertigte Beschriftungsstreifen verfügbar. Sie können die Ein- und Ausgänge von nicht unterstützten Modulen jedoch trotzdem mit Hilfe des TIA Portals beschriften. Zunächst exportieren Sie die absoluten und symbolischen Adressen der Ein- und Ausgänge in eine standardisierte XML-Datei. Die XML-Datei importieren Sie anschließend in ein externes Programm zum Druck von Etiketten. In diesem Programm bereiten Sie die Daten passgenau für Ihre Module auf und drucken die Etiketten schließlich aus.

Vorgehen

Um Beschriftungsdaten für Hardware-Module als XML zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation oder in der Netzsicht die Module aus, für die Sie Beschriftungsstreifen benötigen.
   - Sie können ein oder mehrere Stationen auswählen, um die Ein- und Ausgangsadressen aller darin gesteckten Module zu exportieren.
   - Oder wählen Sie die gewünschten Module unterhalb der Stationen im Ordner "Lokale Baugruppen" aus.
2. Klicken Sie mit der rechten Maustaste auf eines der Geräte und wählen Sie im Kontextmenü den Befehl "Beschriftungsstreifen für Module exportieren".
   Der Dialog "Beschriftungsstreifen exportieren" wird geöffnet.
3. Wählen Sie im Bereich "Exportformat" die Option "In XML-Datei exportieren".
4. Wählen Sie im Feld "Pfad" einen Pfad aus, an dem die XML-Datei abgelegt wird.
5. Klicken Sie auf die Schaltfläche "Export", um den Export in eine XML-Datei zu starten.
   Die XML-Datei wird mit dem Namen "<Projektname>_IO_Channels.xml" erstellt.

Siehe auch

XML-Schema der Exportdatei (Seite 71)

Druckfunktion für Modulbeschriftungsstreifen (Seite 67)

Beschriftungsstreifen drucken (Seite 69)

4.2.4 XML-Schema der Exportdatei

Das XML-Schema einer Exportdatei

Die XML-Datei für Baugruppenbeschriftungsstreifen ist nach folgendem Schema aufgebaut:

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace="http://tempuri.org/XMLSchema.xsd"
elementFormDefault="qualified"  xmlns:mstns="http://tempuri.org/
XMLSchema.xsd"  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Stations">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Station" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
        <xs:element name="Rack" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
          <xs:element name="Module" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
```

```
            <xs:element name="IOChannel" maxOccurs="unbounded">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Address" type="xs:string"></
xs:element>
               <xs:element name ="Tag" type="xs:string"></xs:element>
              </xs:sequence>
            <xs:attribute name="Number" type="xs:int"></xs:attribute>
            </xs:complexType>
            </xs:element>
            </xs:sequence>
            <xs:attribute name="Name"></xs:attribute>
          </xs:complexType>
          </xs:element>
          </xs:sequence>
          <xs:attribute name="Name" type="xs:string"></xs:attribute>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        <xs:attribute name="Name" type="xs:string"></xs:attribute>
      </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  </xs:element>
</xs:schema>
```

Beispiel für eine XML-Datei

Das folgende Beispiel zeigt eine XML-Datei, die Beschriftungsdaten für eine S7-1500 CPU mit einem Digitaleingangsmodul und einem Analogeingangsmodul enthält:

```
<?xml version="1.0" encoding="UTF-8"?>
<Stations>
  <!-- Zunächst wird die CPU aufgeführt -->
  <Station Name="S71500/ET200MP-Station_1">
    <Rack Name="Rack_0"> <!-- Name des Baugruppenträgers -->
      <Module Name="Sample S7-1500" /> <!-- Name der CPU -->
      <Module Name="DI 16x24VDC BA_1"> <!-- Name des
Digitaleingangsmoduls -->
        <!-- Die einzelnen Kanäle des Digitaleingangsmoduls werden
aufgeführt -->
        <IOChannel Number="0">
          <Address>%I0.0</Address>
          <Tag>Input Value 1</Tag> <!-- Symbolische Adresse des
Eingangs 0 -->
        </IOChannel>
        <IOChannel Number="1">
          <Address>%I0.1</Address>
          <Tag>Input Value 2</Tag>
        </IOChannel>
        <IOChannel Number="2">
          <Address>%I0.2</Address>
```

```
          <Tag>Input Value 3</Tag>
        </IOChannel>
        <!-- Alle weiteren Kanäle folgen -->
      </Module>
      <Module Name="AI 4xU/I/RTD/TC ST_1"> <!-- Name des
Analogeingangsmoduls -->
        <!-- Die einzelnen Kanäle des Analogeingangsmoduls werden
aufgeführt -->
        <IOChannel Number="0">
          <Address>%IW2</Address>
          <Tag> <!-- Für das Analogeingangsmodul sind keine
symbolischen Adressen festgelegt. -->
          </Tag>
        </IOChannel>
        <IOChannel Number="1">
          <Address>%IW4</Address>
          <Tag>
          </Tag>
        </IOChannel>
        <IOChannel Number="2">
          <Address>%IW6</Address>
          <Tag>
          </Tag>
        </IOChannel>
        <IOChannel Number="3">
          <Address>%IW8</Address>
          <Tag>
          </Tag>
        </IOChannel>
      </Module>
      <Module Name="Sample S7-1500" />
      <Module Name="DI 16x24VDC BA_1">
        <IOChannel Number="0">
          <Address>%I0.0</Address>
          <Tag>Input Value 1</Tag>
        </IOChannel>
        <IOChannel Number="1">
          <Address>%I0.1</Address>
          <Tag>Input Value 2</Tag>
        </IOChannel>
        <IOChannel Number="2">
          <Address>%I0.2</Address>
          <Tag>Input Value 3</Tag>
        </IOChannel>
        <!-- Alle weiteren Kanäle folgen -->
      </Module>
      <Module Name="AI 4xU/I/RTD/TC ST_1">
        <IOChannel Number="0">
          <Address>%IW2</Address>
          <Tag>
          </Tag>
        </IOChannel>
```

```
            <IOChannel Number="1">
              <Address>%IW4</Address>
              <Tag>
              </Tag>
            </IOChannel>
            <IOChannel Number="2">
              <Address>%IW6</Address>
              <Tag>
              </Tag>
            </IOChannel>
            <IOChannel Number="3">
              <Address>%IW8</Address>
              <Tag>
              </Tag>
            </IOChannel>
          </Module>
        </Rack>
      </Station>
    </Stations>
```

Siehe auch

Beschriftungsdaten als XML exportieren (Seite 70)

4.2.5 Verschiebung des Druckbereichs ermitteln

Wenn Sie einen vorgefertigten Beschriftungsbogen verwenden, muss dieser passgenau bedruckt werden, damit der Text genau an den vorgestanzten Beschriftungsstreifen ausgerichtet ist und später zu den Kanalstatusanzeigen des Moduls passt. Die Papiereinzüge variieren jedoch leicht von Drucker zu Drucker. Deshalb müssen Sie im TIA Portal gegebenenfalls einen passenden Korrekturwert für Ihren Drucker eingeben. Der Druckbereich wird dann in der exportierten DOCX-Datei so verschoben, dass der Druck zu den vorgefertigten Beschriftungsbögen passt.

Die Einstellungen zur Verschiebung des Druckbereichs werden für den jeweiligen Windows-Benutzer gespeichert. Wenn Sie sich unter einem anderen Benutzernamen unter Windows anmelden, müssen Sie die Korrekturwerte erneut eingeben.

Im Folgenden erfahren Sie, wie Sie den Korrekturwert für Ihren Drucker ermitteln.

Voraussetzung

- Sie benötigen einen vorgefertigten Beschriftungsbogen.
- Sie müssen den gleichen Drucker zur Verfügung haben, auf dem später der Ausdruck erfolgen soll. Der Drucker muss druckbereit für den Druck auf handelsüblichem DIN-A4-Papier sein.

Vorgehen

Um den Korrekturwert für Ihren Drucker zu ermitteln, gehen Sie folgendermaßen vor:

1. Drucken Sie einen Beschriftungsbogen auf gewöhnlichem DIN-A4-Papier aus, wie im Kapitel "Beschriftungsstreifen drucken (Seite 69)" beschrieben.
2. Vergleichen Sie den Ausdruck auf dem DIN-A4-Papier mit dem vorgefertigten Beschriftungsbogen.
3. Wenn der Druckbereich verschoben ist, müssen Sie Korrekturwerte verwenden.
   - Messen Sie mithilfe eines Lineals den horizontalen Versatz zum vorgefertigten Beschriftungsbogen. Dieser wird im Export-Dialog für den Druck später in das Feld "Verschiebung horizontal" eingetragen. Ist der Druckbereich nach rechts verschoben, muss ein negativer Korrekturwert eingetragen werden. Ist der Druckbereich nach links verschoben, muss ein positiver Korrekturwert eingetragen werden.
   - Messen Sie mithilfe eines Lineals den vertikalen Versatz zum vorgefertigten Beschriftungsbogen. Dieser wird im Export-Dialog für den Druck später in das Feld "Verschiebung vertikal" eingetragen. Ist der Druckbereich nach unten verschoben, muss ein negativer Korrekturwert eingetragen werden. Ist der Druckbereich nach oben verschoben, muss ein positiver Korrekturwert eingetragen werden.

Mit mehrsprachigen Projekten arbeiten 5

5.1 Grundlagen zu Projekttexten

Texte in unterschiedlichen Sprachen im Projekt

Wenn Sie beim Bearbeiten eines Projekts Texte eingeben, werden Sie dies in der Regel in Ihrer Muttersprache tun. Geben Sie das Projekt im Anschluss an einen anderen Bearbeiter ab, der diese Sprache nicht beherrscht, benötigt er eine Übersetzung aller relevanten Texte in einer für ihn verständlichen Sprache. Alle Texte sind daher übersetzbar. So können Sie sicherstellen, dass jeder, der zu einem späteren Zeitpunkt mit den Texten im Projekt konfrontiert wird, diese Texte in der von ihm gewünschten Sprache angezeigt bekommt.

Projektsprache

Als Projektsprachen werden alle Sprachen bezeichnet, in denen ein Projekt später bearbeitet werden soll. Alle Texte können auf Grundlage der Editiersprache in die verschiedenen Projektsprachen übersetzt werden. Welche Sprachen im Projekt verfügbar sind, legen Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" fest.

Editiersprache

Jedes Projekt besitzt eine sogenannte Editiersprache. Wenn Sie Texte eingeben, werden diese immer in der Editiersprache angelegt. Achten Sie daher darauf, dass die eingestellte Editiersprache der Sprache entspricht, in der Sie Texte eingeben. So vermeiden Sie Probleme beim späteren Übersetzen der Texte.

Die Editiersprache ist unabhängig von der Sprache der Benutzeroberfläche. Sie können beispielsweise Englisch als Oberflächensprache eingestellt haben, als Editiersprache allerdings Italienisch verwenden. Machen Sie Texteingaben, so werden diese in diesem Fall in der Projektsprache "Italienisch" angelegt, obwohl die Benutzeroberfläche des TIA Portals englisch angezeigt wird.

Die Editiersprache stellen Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen > Editiersprache" ein.

Referenzsprache

Die Referenzsprache dient als Vorlage zur Übersetzung. In der Task Card "Aufgaben > Sprachen und Ressourcen" wird zu jedem Eingabefeld der Text in der Referenzsprache eingeblendet. So wissen Sie, welcher Text in ein Textfeld gehört, auch wenn in der aktuell eingestellten Editiersprache noch kein Text eingetragen ist.

Anwendertexte und Systemtexte

Um die Übersichtlichkeit zu verbessern, wird zwischen Anwendertexten und Systemtexten unterschieden:

- Anwendertexte sind Texte, die vom Nutzer angelegt worden sind.
- Systemtexte sind Texte, die automatisch in Abhängigkeit von der Konfiguration im Projekt angelegt werden.

Die Projekttexte verwalten Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Beispiele für mehrsprachige Projekttexte

Sie können z. B. die folgenden Projekttexte mehrsprachig verwalten:

- Bausteintitel und Bausteinkommentare
- Netzwerktitel und Netzwerkkommentare
- Kommentare in Tabellen
- Meldungstexte
- Bedienerrelevante Texte
- Textlisten
- Beschriftungen von Schaltflächen
- Anzeigenamen von Rezepturen

Übersetzen von Texten

Um Texte zu übersetzen, gibt es die folgenden Vorgehensweisen.

- Alle im Projekt verwendeten Texte tabellarisch übersetzen
  Sie können die Übersetzungen für die einzelnen Projektsprachen direkt in die Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".
- Einzelnen Objekten zugeordnete Texte im Inspektorfenster eingeben
  Im Inspektorfenster können Sie die Texte übersetzen, die den aktuell selektierten Objekten zugeordnet sind. In einer Tabelle werden Spalten für alle verfügbaren Projektsprachen angezeigt. In den Spalten können Sie die Übersetzungen für den jeweiligen Text eingeben.
- Texte mithilfe von Referenztexten übersetzen
  Für kleinere Textmengen können Sie die Editiersprache wechseln. Alle Textfelder sind anschließend wieder mit den Standardwerten befüllt und können in der aktuellen Sprache ausgefüllt werden. Zur Orientierung, was Sie zuletzt in das Feld eingegeben haben, können Sie sich den Text in der Referenzsprache anzeigen lassen. Dazu verwenden Sie die Task Card "Aufgaben" und öffnen die Palette "Sprachen & Ressourcen".
- Texte exportieren und extern übersetzen
  Bei großen Textmengen können Sie Texte in eine Office Open XML Datei exportieren und diese in einem gängigen Tabellenkalkulationsprogramm übersetzen. Die übersetzte Liste importieren Sie anschließend wieder in das TIA Portal.

Hinweis

Verwenden von ostasiatischen Projektsprachen

Zur Anzeige ostasiatischer Projektsprachen benötigen Sie Microsoft Windows mindestens in der Professional-Variante oder in einer höheren Ausstattungsvariante. Microsoft Windows in der Professional-Variante muss in der entsprechenden Landessprache installiert sein. In den Ausstattungsvarianten "Ultimate" oder "Enterprise" ist es ausreichend, wenn das passende Sprachpaket installiert ist.

Siehe auch

Anwendungsbeispiele für mehrsprachige Projekte (Seite 90)

5.2 Kategorien von Anwendertexten

Übersicht der Textkategorien

Im Folgenden finden Sie eine Übersicht der Kategorien der Anwendertexte, die Sie beim Exportieren auswählen können.

Textkategorien	Verwendung
Formatierter Meldetext	
HMI Bild	Text, der in einem HMI Bild erscheint
HMI Kommentar	Text, der in einem Kommentar in Runtime erscheint
HMI Runtime	
Kategorie für Textlisten	
Mehrsprachige Textkategorie	
Meldeklassentext	
Meldetext	Text, der in einer Meldeanzeige erscheint
SiVArc Configuration	
Weitere Textkategorie	

5.3 Projektsprachen auswählen

Alle Texte innerhalb eines Projekts können in derselben Sprache dargestellt werden, die Sie auch für Ihre Softwareoberfläche ausgewählt haben. Dafür ist es notwendig, dass alle Projekttexte in der entsprechenden Sprache vorliegen. Die im Projekt zur Verfügung stehenden Projektsprachen können Sie selbst festlegen.

Vorgehen

Um die Projektsprachen auszuwählen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".
   Im Arbeitsbereich sehen Sie eine Liste von Sprachen, die Sie auswählen können.
3. Aktivieren Sie die gewünschten Sprachen.

Ergebnis

Alle Texte können in den aktivierten Sprachen angezeigt werden, sofern bereits eine Übersetzung in den Sprachen vorliegt.

5.4 Editiersprache festlegen

Alle Texte im Projekt werden bei der Eingabe in der so genannten Editiersprache angelegt. Wenn Sie die Editiersprache ändern, werden alle zukünftigen Texteingaben in der neuen Editiersprache gespeichert.

Vorgehen

Um die Editiersprache zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".
   Im Arbeitsbereich werden die Einstellmöglichkeiten zu den Projektsprachen angezeigt.
3. Wählen Sie unter "Allgemein > Editiersprache" die Editiersprache aus.

5.5 Alle Projekttexte tabellarisch übersetzen

Sie können sämtliche Projekttexte, die im aktuell geöffneten Projekt verwendet werden, in einer Liste anzeigen und bearbeiten. Anwender- und Systemtexte sind zur besseren Übersicht in zwei unterschiedlichen Listen getrennt. Beide Listen enthalten für jede Projektsprache jeweils eine Spalte. In die jeweilige Spalte tragen Sie die Übersetzungen der Texte ein.

Voraussetzung

Sie haben mindestens eine weitere Projektsprache gewählt.

Vorgehen

Um Texte in der projektweiten Liste zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte".
   Im Arbeitsbereich erscheint eine Liste mit den Anwendertexten im Projekt.
3. Klicken Sie auf "Systemtexte", wenn Sie statt der Anwendertexte die Liste mit Systemtexten bearbeiten möchten.
4. Schaffen Sie Übersichtlichkeit in der Liste, wenn die Menge an Texten zu groß wird.
   - Um identische Texte zu gruppieren und sie auf einmal zu übersetzen, klicken Sie in der Funktionsleiste auf die Schaltfläche "Gruppierung ein/aus".
   - Um Texte mit fehlenden Übersetzungen auszublenden, klicken Sie in der Funktionsleiste auf die Schaltfläche "Leere Texte ausfiltern ein/aus".
   - Um die angezeigten Projekttexte weiter auf bestimmte Geräte einzugrenzen, wählen Sie in der Klappliste die Geräte aus, zu denen Sie Projekttexte anzeigen möchten.
5. Geben Sie die Übersetzung für die Projekttexte in der jeweiligen Spalte ein.

5.6 Projekttexte übersetzen

Wenn Sie die Texte einzelner Objekte bearbeiten wollen, wäre es zu umständlich die passenden Texte in der Tabelle mit allen Projekttexten herauszusuchen. Daher gibt es im Inspektorfenster eine Tabelle, in der ausschließlich die Texte angezeigt werden, die den aktuell selektierten Objekten zugeordnet sind. In der Tabelle können Sie fehlende Übersetzungen für einzelne Projektsprachen ergänzen oder bereits vorhandene Texte ändern.

Voraussetzung

Für die zu übersetzenden Texte ist mindestens in einer Projektsprache bereits ein Text eingegeben.

Vorgehen

Um die Texte der aktuell selektierten Objekte zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Objekte, deren Texte Sie bearbeiten möchten.
2. Öffnen Sie das Register "Eigenschaften" im Inspektorfenster.
3. Öffnen Sie das unterlagerte Register "Texte" im Inspektorfenster.
   Eine Tabelle mit allen Texten, die zu den selektierten Objekten gehören, wird angezeigt. Die Tabelle enthält jeweils eine Spalte für die aktuell eingestellte Editiersprache und die Referenzsprache sowie weitere Spalten für die zusätzlichen Projektsprachen.
4. Ergänzen oder ändern Sie in der Tabelle die Einträge für die einzelnen Projektsprachen.

Siehe auch

Anwendungsbeispiele für mehrsprachige Projekte (Seite 90)

5.7 Projekttexte mithilfe von Referenztexten übersetzen

Einführung

Nach dem Wechseln der Editiersprache werden alle Texte in Eingabefeldern in der neuen Editiersprache angezeigt. Sind für die neu eingestellte Sprache noch keine Übersetzungen vorhanden, sind die Eingabefelder leer oder mit den Standardwerten befüllt.

Wenn Sie in ein Eingabefeld Text eingeben, wird dieser in der aktuellen Editiersprache gespeichert. Im Anschluss existieren für dieses Eingabefeld Texte in zwei Projektsprachen - in der ehemaligen Editiersprache und in der aktuellen Editiersprache. Auf diese Weise ist es möglich, Texte in mehreren Projektsprachen anzulegen.

Sie können sich bei einem Eingabefeld bereits bestehende Übersetzungen in anderen Projektsprachen anzeigen lassen. Diese dienen als Vergleich für die Texteingabe in der aktuellen Editiersprache und werden als Referenzsprache bezeichnet.

Hinweis

Die Anzeige von Referenztexten ist abhängig von den installierten Produkten und wird nicht von jedem Editor unterstützt.

Voraussetzung

Für ein Eingabefeld existiert mindestens eine Übersetzung in einer weiteren Projektsprache.

Vorgehen

Um sich für ein Eingabefeld die Übersetzung in einer Referenzsprache anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Task Card "Aufgaben" die Palette "Sprachen & Ressourcen ".
2. Wählen Sie aus der Klappliste "Referenzsprache" eine Referenzsprache aus.

Ergebnis

Die Referenzsprache ist voreingestellt. Wenn Sie in ein Textfeld klicken, werden in der Task Card "Aufgaben > Sprachen & Ressourcen" Übersetzungen angezeigt, die bereits in anderen Projektsprachen vorhanden sind.

5.8 Projekttexte exportieren

Sie können Projekttexte zur Übersetzung exportieren und anschließend wieder importieren. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann in Microsoft Excel oder in vielen anderen Tabellenkalkulationsprogrammen bearbeitet werden.

Sie haben folgende Möglichkeiten für den Export:

- Einzelne Projekttexte exportieren
  Im Projekttexte-Editor können Sie einzelne Texte auswählen und ausschließlich die gewählten Texte exportieren.
- Projekttexte eines Geräts exportieren
  Wenn Sie ein Gerät selektiert haben, finden Sie im Register "Eigenschaften > Texte" des Inspektorfensters alle zum jeweiligen Gerät gehörigen Texte. An dieser Stelle können Sie alle zum jeweiligen Gerät gehörigen Texte exportieren.
- Alle Anwendertexte oder Systemtexte auf einmal exportieren
  Sie können entweder alle im Projekt vorhandenen Texte exportieren oder den Export zusätzlich nach Kategorien eingrenzen.

Hinweis

Zeilenbeschränkung bei Microsoft Excel

Beachten Sie, dass Tabellenkalkulationsprogramme möglicherweise nur eine bestimmte Anzahl an Zeilen verarbeiten können. Microsoft Excel 2003 unterstützt beispielsweise maximal 65536 Zeilen. Neuere Versionen von Microsoft Excel unterstützen erheblich mehr Zeilen.

Einzelne Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte".
   Der Projekttexte-Editor wird geöffnet.
3. Wählen Sie das Register "Anwendertexte" oder "Systemtexte" im Editor, je nachdem, welche Texte Sie exportieren möchten.
4. Selektieren Sie die Projekttexte, die Sie exportieren möchten.
5. Klicken Sie in der Funktionsleiste des Editors auf "Projekttexte exportieren".
   Der Dialog "Exportieren" wird geöffnet.
6. Wählen Sie aus der Klappliste "Quellsprache" die Referenzsprache, aus der Sie die Übersetzung vornehmen möchten.
7. Wählen Sie unter "Zielsprache(n)" die Editiersprachen, in die Sie die Texte übersetzen möchten. Im Dialog stehen die bereits festgelegten Projektsprachen zur Verfügung. Fehlt die gewünschte Sprache, legen Sie diese zunächst im Projektsprachen-Editor fest. Wenn Sie alle Editiersprachen exportieren wollen, wählen Sie die Option "Alles auswählen".

8. Geben Sie im Eingabefeld "Zu exportierende Datei auswählen" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
9. Klicken Sie auf "Exportieren".

Projekttexte eines Geräts exportieren

Um zu einem bestimmten Gerät gehörende Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Gerät und öffnen Sie die Eigenschaften des Geräts im Inspektorfenster.
2. Öffnen Sie das Register "Texte" im Inspektorfenster.
3. Wählen Sie aus der Klappliste in der Funktionsleiste die zu exportierenden Texte aus:
   - Alle Texte
   - ProDiag / GRAPH: Meldetexte
   - ProDiag / GRAPH: Texte für HMI-Anzeige
   - ProDiag / GRAPH: Alle Texte
4. Klicken Sie in der Funktionsleiste auf das Symbol "Projekttexte exportieren".
   Der Dialog "Exportieren" wird geöffnet.
5. Wählen Sie aus der Klappliste "Quellsprache" die Referenzsprache, aus der Sie die Übersetzung vornehmen möchten.
6. Wählen Sie unter "Zielsprache(n)" die Editiersprachen, in die Sie die Texte übersetzen möchten. Im Dialog stehen die bereits festgelegten Projektsprachen zur Verfügung.
   Fehlt die gewünschte Sprache, legen Sie diese zunächst im Projektsprachen-Editor fest.
   Wenn Sie alle Editiersprachen exportieren wollen, wählen Sie die Option "Alles auswählen".
7. Geben Sie im Eingabefeld "Zu exportierende Datei auswählen" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
8. Klicken Sie auf "Exportieren".

Alle System- oder Anwendertexte exportieren

Um alle Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Werkzeuge" den Befehl "Projekttexte exportieren".
   Der Dialog "Exportieren" wird geöffnet.
2. Wählen Sie aus der Klappliste "Quellsprache" die Referenzsprache, aus der Sie die Übersetzung vornehmen möchten.
3. Wählen Sie unter "Zielsprache(n)" die Editiersprachen, in die Sie die Texte übersetzen möchten. Im Dialog stehen die bereits festgelegten Projektsprachen zur Verfügung.
   Fehlt die gewünschte Sprache, legen Sie diese zunächst im Projektsprachen-Editor fest.
   Wenn Sie alle Editiersprachen exportieren wollen, wählen Sie die Option "Alles auswählen".
4. Aktivieren Sie unter "Inhalte auswählen" das Optionskästchen "Anwendertexte", um Anwendertexte zu exportieren. Wählen Sie "Systemtexte", um Systemtexte zu exportieren. Aktivieren Sie beide Optionskästchen, um sowohl Anwendertexte als auch Systemtexte zu exportieren.

5. Wählen Sie unter "Inhalte auswählen" die gewünschten Textkategorien für die Anwendertexte oder die Systemtexte aus.
6. Geben Sie im Eingabefeld "Exportdatei" einen Dateinamen für die Exportdatei an.
7. Wählen Sie im Eingabefeld "Pfad" den Pfad im Dateisystem aus, an dem die Exportdatei gespeichert werden soll.
8. Klicken Sie auf "Exportieren".

Siehe auch

Anwendungsbeispiele für mehrsprachige Projekte (Seite 90)

Projekttexte importieren (Seite 89)

5.9 Projekttexte importieren

Nach der externen Übersetzung in einem Tabellenkalkulationsprogramm importieren Sie die Projekttexte in das TIA Portal. Die Projekttexte können Sie an folgenden Stellen importieren:

- Das Menü "Werkzeuge"
- In der Funktionsleiste des Projekttexte-Editors
- In den Eigenschaften eines Geräts
  Wenn Sie ein Gerät selektiert haben, finden Sie im Register "Eigenschaften > Texte" des Inspektorfensters alle zum jeweiligen Gerät gehörigen Texte. An dieser Stelle können Sie die Texte des Geräts ebenfalls importieren.

Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Werkzeuge" den Befehl "Projekttexte importieren".
   Alternativen:
   – Klicken Sie in der Funktionsleiste des Projekttexte-Editors auf das Symbol "Projekttexte importieren".
   – Selektieren Sie ein Gerät und öffnen Sie dessen Eigenschaften im Inspektorfenster. Öffnen Sie das Register "Texte" und klicken Sie in der Funktionsleiste auf das Symbol "Projekttexte importieren".
2. Der Dialog "Importieren" wird geöffnet.
3. Wählen Sie aus dem Feld "Zu importierende Datei auswählen" den Pfad und den Dateinamen der Importdatei aus.
4. Aktivieren Sie das Optionskästchen "Quellsprache importieren", wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
5. Klicken Sie auf "Importieren".

Siehe auch

Projekttexte exportieren (Seite 86)

5.10 Anwendungsbeispiele für mehrsprachige Projekte

Gehen wir davon aus, Sie arbeiten in einem Team mit Kollegen, die jeweils Englisch, Französisch oder Deutsch sprechen. Mit Hilfe des TIA Portals haben Sie ein Projekt angelegt und bereits eine funktionierende Konfiguration erstellt.

Um das Projekt für die anderen Kollegen des Teams nachvollziehbar zu halten, möchten Sie alle Geräte in den Sprachen Englisch und Deutsch mit Kommentaren versehen. Zunächst möchten Sie die Kommentare in Deutsch eingeben. Anschließend möchten Sie aus Zeit- und Kostengründen ein externes Übersetzungsbüro beauftragen, um die Texte dort in einem Tabellenkalkulationsprogramm ins Englische übersetzen zu lassen.

Zusätzlich möchten Sie einen einzelnen Kommentar zu einem bestimmten Gerät auch in Französisch vorhalten, damit der französischsprachige Kollege dieses Gerät weiterbearbeiten kann.

Im Folgenden ist beispielhaft beschrieben, wie Sie dieses Ziel mit den Mitteln des TIA Portals erreichen können.

Das Projekt ins Englische übersetzen

Um die Kommentare zunächst auf Deutsch einzugeben und anschließend ins Englische übersetzen zu lassen, gehen Sie folgendermaßen vor:

1. Stellen Sie die Editiersprache auf "Deutsch" ein und befüllen Sie alle Kommentarfelder mit den jeweiligen Texten auf Deutsch.
   Bei dem ausgewählten Gerät für den französischsprachigen Kollegen geben Sie beispielsweise zunächst auf Deutsch "Unser neues Gerät" ein.
   Alle Kommentare sind nun in deutscher Sprache hinterlegt.
2. Exportieren Sie alle Anwendertexte in eine Office Open XML Datei mit der Endung ".xlsx".
3. Lassen Sie die in der Datei enthaltenen Anwendertexte in einem gängigen Tabellenkalkulationsprogramm, wie z. B. Microsoft Excel, ins Englische übersetzen.
4. Importieren Sie die Datei nach der Übersetzung in das TIA Portal.
   Alle Texte stehen nun in Deutsch und Englisch zur Verfügung.

Einzelnes Kommentarfeld auf Französisch übersetzen

Um den Inhalt eines einzelnen Kommentarfeldes auf Französisch zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Kommentarfeld zu dem Gerät, welches für die Weiterbearbeitung durch den französischsprachigen Kollegen vorgesehen ist.
2. Öffnen Sie die Palette "Sprachen & Ressourcen" in der Task Card "Aufgaben".

3. Stellen Sie in der Palette "Sprachen & Ressourcen" als Editiersprache "Französisch" ein. Als Referenzsprache stellen Sie beispielsweise "Englisch" ein.
   Da im Französischen noch keine Übersetzung gespeichert ist, ist das Kommentarfeld leer. In der Palette "Sprachen & Ressourcen" wird jedoch die englische Übersetzung "Our new device" als Referenz angezeigt.
4. Orientieren Sie sich am englischen Referenztext und geben Sie in das Kommentarfeld "Notre nouvel appareil" ein.
   Der Kommentar zu diesem Gerät liegt nun in den Sprachen Deutsch, Englisch und Französisch vor.

Siehe auch

Grundlagen zu Projekttexten (Seite 77)

Projekttexte exportieren (Seite 86)

Projekttexte übersetzen (Seite 84)

Mit Textlisten arbeiten 6

6.1 Textlisten

Einführung

Texte, die in Meldungen referenziert werden, lassen sich zentral verwalten. Alle Texte sind in so genannten Textlisten gespeichert. Jede Textliste hat einen eindeutigen Namen, mit dem ihr Inhalt aufgerufen wird. Jedem Text in einer Textliste ist ein Wertebereich zugeordnet. Tritt ein Wert aus einem Wertebereich auf, wird der zugehörige Text aufgerufen.

Alle Texte sind in sämtliche Projektsprachen übersetzbar. Dafür stehen Ihnen zwei Möglichkeiten zur Verfügung:

- Sie können die Übersetzung der Texte in eine Liste eingeben. Die Liste finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".
- Sie können alle Texte in eine Datei im Office Open XML Format exportieren und die Übersetzung in einem Tabellenkalkulationsprogramm eingeben. Anschließend können die Übersetzungen wieder importiert werden. Exportieren Sie die Daten nur in einen Bereich, der mit entsprechenden Zugriffsmechanismen geschützt ist. Importieren Sie nur Dateien, die aus vertrauenswürdigen Quellen stammen.

Jedes Gerät in einem Projekt kann über eigene Textlisten verfügen. Gerätespezifische Textlisten betreffen nur ein Gerät des Projekts und sind deshalb nur für dieses gültig. In der Projektnavigation sind sie daher unterhalb eines Gerätes angeordnet. Gerätespezifische Textlisten können anwenderdefiniert oder auch vom System vorgegeben sein.

Textlisten-Editor

Der Textlisten-Editor ist in einen unteren und einen oberen Bereich eingeteilt. Im oberen Bereich werden die einzelnen Textlisten angezeigt. Sobald Sie eine Textliste selektieren, werden im unteren Bereich die enthaltenen Texte und die zugehörigen Wertebereiche angezeigt. Die Tabellenspalten im Textlisten-Editor sind auf- und absteigend sortierbar, indem Sie auf den Tabellenkopf der jeweiligen Spalte klicken.

Anwenderdefinierte und systemdefinierte Textlisten

Es gibt zwei Arten von Textlisten:

- Anwenderdefinierte Textlisten
  Anwenderdefinierte Textlisten können Sie selbst anlegen und mit Texten befüllen, d. h. Sie können Wertebereiche und dazu gehörende Texte selbst festlegen. Bei anwenderdefinierten Textlisten beginnt der Name der Textliste standardmäßig mit "USER". Sie können diese Bezeichnung allerdings beliebig ändern.
- Systemdefinierte Textlisten
  Systemdefinierte Textlisten werden vom System vorgegeben. Es handelt sich immer um Texte, die Geräte betreffen. Sie werden automatisch angelegt, sobald Sie ein Gerät in das Projekt einfügen. Bei Systemmeldungen beginnt der Name der Textliste mit "SYSTEM". Der Name der Textliste sowie die darin enthaltenen Wertebereiche sind nicht veränderbar. Sie können nur Texte bearbeiten, die einzelnen Wertebereichen zugeordnet sind.

Anwenderdefinierte Textlisten	Systemdefinierte Textlisten
Sie können neue Textlisten anlegen und vorhandene löschen.	Sie können keine neuen Textlisten anlegen und keine Textlisten löschen.
Sie können in den Textlisten Wertebereiche hinzufügen und löschen.	Sie können in den Textlisten keine Wertebereiche hinzufügen oder löschen.
Sie können sowohl die Wertebereiche als auch die zugehörigen Texte festlegen.	Sie können nur den zu einem Wertebereich gehörigen Text bearbeiten.

Siehe auch

Projekttexte exportieren (Seite 86)

6.2 Anwenderdefinierte Textlisten anlegen

Sie können anwenderdefinierte Textlisten für einzelne Geräte neu anlegen.

Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät.

Vorgehen

Um anwenderdefinierte Textlisten anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät.
   Die unterhalb des Geräts angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "PLC-Meldetextlisten".
   Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Doppelklicken Sie in die erste freie Zeile der Tabelle.
   Eine neue anwenderdefinierte Textliste wird erstellt.
4. Tragen Sie in der Spalte "Name" einen Namen für Ihre neue Textliste ein.
5. Wählen Sie aus der Klappliste in der Spalte "Auswahl", ob Sie die Wertebereiche dezimal, binär oder in Bit angeben. Abhängig vom Gerät sind an dieser Stelle gegebenenfalls weitere Optionen möglich.
6. Geben Sie in der Spalte "Bemerkung" eine Bemerkung ein.
   Eine neue anwenderdefinierte Textliste wird angelegt und Sie können als Nächstes die Wertebereiche und Texte eingeben.

6.3 Anwenderdefinierte Textlisten bearbeiten

Sie können Wertebereiche und dazugehörigen Texte in anwenderdefinierte Textlisten eintragen. Anwenderdefinierte Textlisten befinden sich in der Projektnavigation immer unterhalb eines Geräts.

Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät

Vorgehen

Um anwenderdefinierte Textlisten mit Wertebereichen und Texten zu ergänzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät.
   Die unterhalb angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "PLC-Meldetextlisten".
   Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Selektieren Sie eine Textliste in der Tabelle.
   Im Arbeitsbereich werden die Inhalte der selektierten Textliste geöffnet. Dort können Sie Wertebereich eintragen und den einzelnen Wertebereichen Texte zuordnen.
4. Geben Sie die gewünschten Wertebereiche in die Spalten "Bereich von" und "Bereich bis" ein. Der Eintrag muss dem gewählten Zahlenformat für die Textliste entsprechen.
5. Geben Sie zu jedem Wertebereich einen Text in die Spalte "Eintrag" ein.

6.4 Systemdefinierte Textlisten bearbeiten

In systemdefinierten Textlisten können Sie die einzelnen, einem Wertebereich zugeordneten Texte ändern.

Systemdefinierte Textlisten befinden sich in der Projektnavigation unterhalb eines Geräts.

Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät.

Vorgehen

Um in systemdefinierten Textlisten Texte zu bearbeiten, die einem Wertebereich zugeordnet sind, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät.
   Die unterhalb angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "PLC-Meldetextlisten".
   Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Selektieren Sie eine Textliste in der Tabelle.
   Im Arbeitsbereich werden die Inhalte der selektierten Textlisten geöffnet. Dort können Sie einem Wertebereich zugeordnete Texte ergänzen oder bearbeiten.
4. Geben Sie zu jedem Wertebereich einen Text in die Spalte "Eintrag" ein.

6.5 Textlisten exportieren

Sie können Meldetexte zur Übersetzung exportieren und anschließend wieder importieren. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann in Microsoft Excel oder in vielen anderen Tabellenkalkulationsprogrammen bearbeitet werden.

Voraussetzungen

- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät.
- In dem Projekt sind Textlisten angelegt.
- Der Textlisten-Editor ist geöffnet.

Vorgehen

Um Textlisten zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät.
   Die unterhalb des Geräts angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "PLC-Meldetextlisten".
   Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Selektieren Sie eine Textliste in der Tabelle.
   Im Arbeitsbereich werden die Einträge der selektierten Textliste geöffnet.
4. Selektieren Sie im Arbeitsbereich die Textliste, die Sie exportieren wollen.
5. Klicken Sie in der Funktionsleiste des Textlisten-Editors auf das Symbol "Textlisten exportieren".
   Der Dialog "Textlisten exportieren" öffnet sich.
6. Geben Sie im Eingabefeld "Zu exportierende Datei auswählen" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
7. Wählen Sie die Projektsprachen aus, die Sie exportieren wollen.
8. Klicken Sie auf "Export".
   Eine Meldung über den Status des Exports öffnet sich.

Ergebnis

Die Textlisten werden in eine Office Open XML Datei mit der Endung ".xlsx" exportiert. Die Datei wird im angegebenen Ordner abgelegt.

6.6 Textlisten importieren

Nach der externen Übersetzung in einem Tabellenkalkulationsprogramm importieren Sie die Meldetexte in das TIA Portal.

Voraussetzungen

- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät.
- In dem Projekt sind Textlisten angelegt.
- Der Textlisten-Editor ist geöffnet.

Vorgehen

Um Textlisten zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät.
   Die unterhalb des Geräts angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "PLC-Meldetextlisten".
   Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Klicken Sie in der Funktionsleiste des Textlisten-Editors auf das Symbol "Textlisten importieren".
   Der Dialog "Textlisten importieren" öffnet sich.
4. Wählen Sie aus dem Feld "Datei importieren" den Pfad und den Dateinamen der Importdatei aus.
5. Klicken Sie auf "Import".
   Eine Meldung über den Status des Imports öffnet sich.

Ergebnis

Die Textlisten werden importiert. Die existierenden Daten in den Textlisten im Projekt werden dabei überschrieben und neue Daten werden hinzugefügt. Textlisten werden nur in den Sprachen importiert, die im Projekt aktiviert wurden.

Memory Cards verwenden 7

7.1 Grundlagen zu Memory Cards

Einführung

Memory Cards sind steckbare Speicherkarten, die für unterschiedliche Zwecke eingesetzt werden können und in verschiedenen Typen erhältlich sind. Je nach Gerätetyp bzw. Gerätefamilie können Memory Cards z. B. für folgende Zwecke eingesetzt werden:

- Als Ladespeicher einer CPU
- Speichermedium für Projekte, Firmware-Backups oder beliebige andere Dateien
- Speichermedium für die Durchführung einer Firmware-Aktualisierung
- Speichermedium für den PROFINET-Gerätenamen

Informationen zu den technischen Ausprägungen der jeweiligen Memory Cards und zur allgemeinen Hantierung finden Sie in den jeweiligen Dokumentationen zum Gerät. Informationen zur Hantierung von Memory Cards im TIA Portal finden Sie in der Online-Hilfe unter dem Stichwort "Memory Card".

ACHTUNG
Memory Card wird für SIMATIC-Geräte unbrauchbar Wenn Sie eine SIMATIC Memory Card für SIMATIC-fremde Zwecke nutzen oder nicht wie vorgesehen formatieren, überschreiben Sie die interne Struktur der SIMATIC Memory Card. Die Struktur ist nicht wiederherstellbar und die SIMATIC Memory Card wird für SIMATIC-Geräte unbrauchbar. Verwenden Sie Memory Cards nicht für SIMATIC-fremde Zwecke und formatieren Sie die SIMATIC Memory Card weder mit Fremdgeräten noch mit Windows-Mitteln.

Siehe auch

Benutzerdefinierten Kartenleser hinzufügen (Seite 102)

Auf Memory Cards zugreifen (Seite 103)

Eigenschaften von Memory Cards anzeigen (Seite 104)

7.2 Benutzerdefinierten Kartenleser hinzufügen

Einführung

Falls Ihr Kartenleser nicht automatisch erkannt wurde, können Sie ihn manuell hinzufügen.

Voraussetzung

Die Projektansicht ist geöffnet.

Vorgehen

Um einen Kartenleser hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Benutzer-definierten Card Reader hinzufügen".
   Der Dialog "Benutzer-definierten Card Reader hinzufügen" wird geöffnet.
3. Wählen Sie in der Klappliste den Pfad für den Kartenleser aus.
4. Bestätigen Sie die Eingabe mit "OK".

Siehe auch

Grundlagen zu Memory Cards (Seite 101)

Auf Memory Cards zugreifen (Seite 103)

Eigenschaften von Memory Cards anzeigen (Seite 104)

7.3 Auf Memory Cards zugreifen

Voraussetzung

- Eine Memory Card ist im Kartenleser vorhanden.
- Die Projektansicht ist geöffnet.

Hinweis

Sie können nicht gleichzeitig mit mehreren Memory Cards arbeiten. Stecken Sie daher immer nur eine Memory Card in den Kartenleser.

Vorgehen

Um auf Memory Cards zuzugreifen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Card Reader/USB-Speicher anzeigen".
   In der Projektnavigation wird der Ordner "Card Reader/USB-Speicher" angezeigt.
3. Öffnen Sie den Ordner "Card Reader/USB-Speicher".
   Sie können nun auf die Memory Card zugreifen.

   Hinweis

   Wenn auf der Memory Card Daten eines nicht installierten Produktes gespeichert sind, werden die Ordner, die diese Daten enthalten, grau angezeigt. Beim Zugriff auf einen solchen Ordner erhalten Sie eine Fehlermeldung. Installieren Sie bei Bedarf das entsprechende Produkt.

Siehe auch

Grundlagen zu Memory Cards (Seite 101)

Benutzerdefinierten Kartenleser hinzufügen (Seite 102)

Eigenschaften von Memory Cards anzeigen (Seite 104)

7.4 Eigenschaften von Memory Cards anzeigen

Sie können sich die Eigenschaften der verwendeten Memory Cards anzeigen lassen. Beachten Sie, dass abhängig vom Gerät unterschiedliche Memory Cards verwendet werden müssen, die unterschiedliche Eigenschaften besitzen.

Voraussetzung

- Eine Memory Card ist im Kartenleser vorhanden.
- Die Projektansicht ist geöffnet.

Vorgehen

Um die Eigenschaften einer Memory Card anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Memory Card, deren Eigenschaften Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Dialog "Memory Card <Name der Memory Card>" wird geöffnet. In diesem Dialog werden die Eigenschaften angezeigt.

Siehe auch

Grundlagen zu Memory Cards (Seite 101)

Benutzerdefinierten Kartenleser hinzufügen (Seite 102)

Auf Memory Cards zugreifen (Seite 103)

7.5 Daten auf der Memory Card eines Controllers anzeigen

Auf der Memory Card in einem Controller können zusätzlich zur Projektierung Daten wie z. B. Firmware Updates des Controllers, HMI-Images oder HMI-Backups gespeichert werden. Wenn eine Online-Verbindung zu dem Controller besteht, werden diese Daten in der Projektnavigation im Ordner "Online-Kartendaten" unterhalb des Controllers angezeigt.

Voraussetzungen

- Eine Memory Card ist im Kartenleser des Controllers vorhanden.
- Die Memory Card enthält darstellbare Daten.
- Die Projektansicht ist geöffnet.
- Es besteht eine Online-Verbindung zu dem Controller.

Vorgehen

Um Daten auf der Memory Card eines Controllers anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Geräteordner.
2. Öffnen Sie den Ordner "Online-Kartendaten".
   Die Daten werden in Unterordnern angezeigt.

Querverweise nutzen 8

8.1 Querverweise nutzen

Einführung zu Querverweisen

Die Querverweisliste bietet einen Überblick über die Verwendung von Objekten innerhalb des Projekts. Sie können ablesen, welche Objekte in gegenseitiger Abhängigkeit stehen und wo sich die einzelnen Objekte befinden. Querverweise sind somit Teil der Projektdokumentation.

Zusätzlich können Sie direkt zur Verwendungsstelle eines Objekts springen.

Welche Objekte in der Querverweisliste angezeigt und lokalisiert werden können, ist abhängig von den installierten Produkten.

Anwenderdefinierte Dokumentation bereitstellen 9

9.1 Anwenderdefinierte Dokumentation verwenden

Anwenderdefinierte Dokumentation zu Projekt- oder Bibliotheksinhalten

In einem Projekt oder einer Bibliothek erstellen Sie im Laufe der Zeit etliche eigene Inhalte. Eigene Inhalte sind z. B. Bausteine, Variablen oder Bibliothekstypen. Während die Funktionsweise des TIA Portals in der mitgelieferten Hilfe beschrieben ist, gibt es für Ihre selbst erstellten Inhalte keine Hilfe. Um anderen Mitarbeitern die Funktionsweise Ihres Projekts oder die Verwendung einzelner Bibliothekstypen zu erläutern, können Sie selbst anwenderdefinierte Dokumentation erstellen.

Sie können Ihre anwenderdefinierte Dokumentation in den verfügbaren Oberflächensprachen bereitstellen. Standardmäßig werden im TIA Portal die Oberflächensprachen Deutsch, Englisch, Französisch, Spanisch, Italienisch und Chinesisch angeboten. Damit zu einem Objekt die passende Hilfe aufgerufen wird, müssen Sie während der Erstellung der anwenderdefinierte Dokumentation einige Konventionen einhalten.

Entweder erstellen Sie die anwenderdefinierte Dokumentation in einem der unterstützten Office-Formate oder als kompilierte HTML-Hilfe im CHM-Format.

Mögliche Bereiche für anwenderdefinierte Dokumentation

In den folgenden Bereichen des TIA Portals können Sie z. B. anwenderdefinierte Dokumentation anbieten:

- Projektnavigation
- Task Card "Bibliotheken" und Bibliotheksansicht
- Einige Editoren, abhängig von den installierten Produkten, z. B.:
  - Programmiereditor
    Die Programmiersprachen KOP, FUP, AWL, SCL und GRAPH werden unterstützt. Auch Bausteinaufrufe unterstützen die Anbindung von anwenderdefinierter Dokumentation.
  - Editor "Bilder"
    Im Editor "Bilder" werden Verwendungen von Bildern und Instanzen von HMI Bildbausteinen unterstützt.

Verzeichnisse für anwenderdefinierte Dokumentation

Die anwenderdefinierte Dokumentation speichern Sie in einem der folgenden Verzeichnisse:

- Projektverzeichnis
  Wenn Sie anwenderdefinierte Dokumentation zu Objekten innerhalb eines Projekts anlegen, speichern Sie die Hilfe im Projektverzeichnis. Wenn Sie das Projekt weitergeben, ist die anwenderdefinierte Dokumentation ebenfalls enthalten.
- Verzeichnis einer globalen Bibliothek
  Wenn Sie anwenderdefinierte Dokumentation zu Objekten innerhalb einer globalen Bibliothek anlegen, speichern Sie die anwenderdefinierte Dokumentation im Verzeichnis der globalen Bibliothek. Wenn Sie die globale Bibliothek weitergeben, ist die anwenderdefinierte Dokumentation ebenfalls enthalten.
- Zentrales Verzeichnis auf der Festplatte oder einem Netzlaufwerk
  Die anwenderdefinierte Dokumentation können Sie in einem zentralen Verzeichnis auf der Festplatte oder auf einem Netzlaufwerk ablegen. So haben Sie in jedem Projekt Zugriff auf die anwenderdefinierte Dokumentation oder nutzen die Dokumentation auf einem Netzlaufwerk gemeinsam im Team. Das zentrale Ablageverzeichnis für die anwenderdefinierte Dokumentation legen Sie mithilfe einer XML-Datei oder in den Einstellungen des TIA Portals fest.

Einstiegsseite für die anwenderdefinierte Dokumentation

Sie können für jede Sprachversion der anwenderdefinierten Dokumentation eine eigene Einstiegsseite anlegen. Die Einstiegsseite für die anwenderdefinierte Dokumentation kann allgemeine Hilfe zu einem Projekt oder zu einer Bibliothek enthalten. Die Einstiegsseite muss im zentralen Ablageverzeichnis für anwenderdefinierte Dokumentation gespeichert sein.

Aufruf der anwenderdefinierten Dokumentation

Wenn anwenderdefinierte Dokumentation zu einem Objekt vorhanden ist, rufen Sie diese mit der Tastenkombination <Shift+F1> auf. Die anwenderdefinierte Dokumentation wird immer mit dem in Microsoft Windows festgelegten Standardprogramm für das jeweilige Dateiformat geöffnet.

Nachdem Sie <Shift+F1> gedrückt haben, werden bestimmte Verzeichnissen in einer festen Reihenfolge nach anwenderdefinierter Dokumentation durchsucht. Im Folgenden sehen Sie Suchreihenfolge:

1. Suche im zentralen Verzeichnis für anwenderdefinierte Dokumentation
   1.1 Suche nach einer CHM-Datei
   1.2 Suche nach Dokumentation in anderen Dateiformaten
2. Suche im Projekt- oder Bibliotheksverzeichnis
   2.1 Suche nach einer CHM-Datei
   2.2 Suche nach Dokumentation in anderen Dateiformaten

Die Suche wird zunächst im Sprachverzeichnis für die aktuell eingestellte Oberflächensprache des TIA Portals durchgeführt. Wenn in diesem Sprachverzeichnis keine Hilfe vorhanden ist, wird in derselben Reihenfolge im englischen Sprachverzeichnis nach anwenderdefinierter Dokumentation gesucht.

Sobald an einem Ort anwenderdefinierte Dokumentation gefunden wird, wird die anwenderdefinierte Dokumentation geöffnet und die weitere Suche abgebrochen. Wenn in keinem der Verzeichnisse anwenderdefinierte Dokumentation vorhanden ist, wird in der vorangehend dargestellten Reihenfolge nach einer Einstiegsseite für anwenderdefinierte Dokumentation gesucht. Zunächst wird wieder im Sprachverzeichnis für die aktuell eingestellte Oberflächensprache nach einer Einstiegsseite gesucht. Wenn dort keine Einstiegsseite gefunden wird, wird das englische Sprachverzeichnis durchsucht.

Aufrufprotokoll

Um Ihnen die Anbindung der anwenderdefinierten Dokumentation zu erleichtern, lässt sich ein Aufrufprotokoll zur anwenderdefinierten Dokumentation anzeigen. Die Meldungen innerhalb des Protokolls zeigen, in welchen Verzeichnissen nach Dokumentation gesucht wird und ob der Aufruf der anwenderdefinierten Dokumentation erfolgreich ist. Außerdem wird der Dateiname angezeigt, der für die Datei erwartet wird. Somit können Sie erkennen, wie Sie Ihre Dokumentation benennen müssen und in welchen Verzeichnissen Sie die anwenderdefinierte Dokumentation speichern müssen. Das Aufrufprotokoll entspricht derselben Reihenfolge, in der nach anwenderdefinierter Dokumentation oder einer Einstiegsseite gesucht wird.

Das Protokoll wird im Inspektorfenster im Register "Info" angezeigt. Bevor Sie das Aufrufprotokoll anzeigen können, müssen Sie das Aufrufprotokoll zunächst in den Einstellungen des TIA Portals aktivieren. Aktivieren Sie dazu das Optionskästchen "Aufrufprotokoll zu anwenderdefinierter Dokumentation anzeigen" unter "Einstellungen > Allgemein > Anwenderdokumentation". Alternativ aktivieren Sie die Anzeige des Aufrufprotokolls mithilfe einer XML-Datei.

Siehe auch

Konventionen für das Erstellen (Seite 117)

Einstellungen mit einer XML-Datei festlegen (Seite 113)

Einstiegsseite erstellen (Seite 116)

Anwenderdefinierte Dokumentation erstellen (Seite 122)

Anwenderdefinierte Dokumentation aufrufen (Seite 120)

9.2 Einstellungen im TIA Portal festlegen

In den Einstellungen des TIA Portals legen Sie folgende Einstellungen für die anwenderdefinierte Dokumentation fest:

- Aufrufprotokoll im Inspektorfenster anzeigen
  Im Inspektorfenster wird im Register 'Info > Allgemein" ein Protokoll vom Aufruf der anwenderdefinierten Dokumentation angezeigt. Das Protokoll hilft Ihnen, die Konventionen für den Aufruf der anwenderdefinierten Dokumentation einzuhalten.
- Suche nach anwenderdefinierter Dokumentation in einem zentralen Ablageverzeichnis
  Sie können die anwenderdefinierte Dokumentation in einem Verzeichnis außerhalb des aktuellen Projektverzeichnisses speichern, z. B. um projektübergreifende Dokumentation zur Verfügung zu stellen.
- Zentrales Verzeichnis für anwenderdefinierte Dokumentation
  Im zentralen Ablageverzeichnis für anwenderdefinierte Dokumentation speichern Sie projektübergreifende Dokumentation ab.

Hinweis

XML-Konfigurationsdatei gilt vorrangig zu den Einstellungen des TIA Portals

Wenn Sie eine XML-Konfigurationsdatei verwenden und dort Einstellungen für anwenderdefinierte Dokumentation festgelegt haben, dann gelten vorrangig die Einstellungen in der XML-Datei. Sobald Sie die XML-Konfigurationsdatei erneuern oder das TIA Portal neu starten, werden die Einstellungen aus der XML-Datei übernommen. Die Einstellungen, die Sie im TIA Portal vorgenommen haben, verlieren ihre Gültigkeit.

Vorgehen

Um einen zentralen Ablageort für Anwenderhilfe festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Öffnen Sie den Bereich "Allgemein > Allgemein".
3. Navigieren Sie zum Abschnitt "Anwenderdokumentation".
4. Aktivieren Sie das Optionskästchen "Aufrufprotokoll zu anwenderdefinierter Dokumentation anzeigen", um im Inspektorfenster ein Protokoll vom Aufruf der anwenderdefinierten Dokumentation anzuzeigen.
5. Aktivieren Sie das Optionskästchen "Suche nach anwenderdefinierter Dokumentation in einem zentralen Verzeichnis", um anwenderdefinierte Dokumentation in einem projektübergreifenden Verzeichnis abzulegen.
6. Geben Sie im Feld "Zentrales Verzeichnis für anwenderdefinierte Dokumentation" den Pfad an, an dem Sie projektübergreifende Dokumentation speichern.

Siehe auch

Einstellungen mit einer XML-Datei festlegen (Seite 113)

9.3 Einstellungen mit einer XML-Datei festlegen

Einstellungen für die anwenderdefinierte Dokumentation können Sie alternativ zu den Einstellungen im TIA Portal in einer XML-Datei vornehmen. Die XML-Datei ist dieselbe Datei, die Sie für die Einbindung von Unternehmensbibliotheken verwenden.

Wenn Sie eine XML-Konfigurationsdatei verwenden und dort Einstellungen für anwenderdefinierte Dokumentation festgelegt haben, dann gelten vorrangig die Einstellungen in der XML-Datei. Sobald Sie die XML-Konfigurationsdatei erneuern oder das TIA Portal neu starten, werden die Einstellungen aus der XML-Datei übernommen. Die Einstellungen, die Sie im TIA Portal vorgenommen haben, verlieren ihre Gültigkeit.

In der XML-Konfigurationsdatei können Sie folgende Optionen einstellen:

- Aufrufprotokoll im Inspektorfenster anzeigen
  Im Inspektorfenster wird im Register 'Info > Allgemein" ein Protokoll vom Aufruf der anwenderdefinierten Dokumentation angezeigt. Das Protokoll hilft Ihnen, die Konventionen für den Aufruf der anwenderdefinierten Dokumentation einzuhalten.
- Suche nach anwenderdefinierter Dokumentation in einem zentralen Ablageverzeichnis
  Sie können die anwenderdefinierte Dokumentation in einem Verzeichnis außerhalb des aktuellen Projektverzeichnisses speichern, z. B. um projektübergreifende Dokumentation zur Verfügung zu stellen.
- Zentrales Verzeichnis für anwenderdefinierte Dokumentation
  Im zentralen Ablageverzeichnis für anwenderdefinierte Dokumentation speichern Sie projektübergreifende Dokumentation ab.

Vorgehen

Um Einstellungen für die anwenderdefinierte Dokumentation festzulegen, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine XML-Datei mit dem Namen "CorporateSettings.xml", falls Sie noch keine XML-Konfigurationsdatei für die Einbindung von Unternehmensbibliotheken verwenden. Wenn Sie bereits eine Konfigurationsdatei einsetzen, fahren Sie mit Schritt 3 fort.
   Die Konfigurationsdatei muss mit der Kodierung "UTF-8" gespeichert sein.
2. Speichern Sie die Datei in folgendem Verzeichnis auf dem Computer:
   C:\ProgramData\Siemens\Automation\Portal V15_1\CorporateSettings\
3. Tragen Sie den unten genannten Inhalt in die XML-Konfigurationsdatei ein.
4. Passen Sie die Attribute für die Anzeige der anwenderdefinierten Dokumentation an. Die Bedeutung der einzelnen Elemente entnehmen Sie den Kommentaren in der XML-Konfigurationsdatei. Um eine Funktion zu aktivieren, verwenden Sie den Wert "true". Um eine Funktion zu deaktivieren, verwenden Sie den Wert "false".

Inhalt der XML-Konfigurationsdatei

Die XML-Konfigurationsdatei muss folgenden Inhalt besitzen:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<Document>
 <Settings.Settings ID="0">
  <ObjectList>
   <Settings.General ID="1" AggregationName="General">
    <!-- An dieser Stelle finden Sie gegebenenfalls die Einstellungen für globale
Unternehmensbibliotheken. -->
    <ObjectList>

<!-- Sektion Unternehmensbibliotheken -->
     <Settings.General ID="1" AggregationName="General">
     <AttributeList>
      <CorporateLibraryPaths>
      <!-- Beispiel für zwei Einträge -->
      <Item>F:\CorporateLibraries\Corporate_Library_1\Corporate_Library_1.al15_1</
Item>          <Item>F:\CorporateLibraries\Corporate_Library_2\CorporateLibraries.al15_1</
Item>
      <!-- Tragen Sie hier gegebenenfalls alle weiteren globalen Bibliotheken ein. --
>           </CorporateLibraryPaths>
    </AttributeList>
   </Settings.General>
<!-- Sektion Unternehmensbibliotheken Ende -->

<!-- Sektion anwenderdefinierte Dokumentation -->
     <Settings.UserDocumentation ID="2" AggregationeName="UserDocumentation">
     <!-- Im folgenden Abschnitt legen Sie die Werte für die Anzeige der anwenderdefinierte
Dokumentation fest. -->
      <AttributeList>
       <!-- Aktiviert oder deaktiviert die Anzeige des Zugriffsprotokolls. -->
       <DisplayLogInformation>
        <Value>true</Value>
        </DisplayLogInformation>
       <!-- Aktiviert oder deaktiviert die Suche nach anwenderdefinierter Dokumentation in
einem zentralen Verzeichnis. -->
       <EnableLookupFromCentralStorageLocation>
         <Value>true</Value>
       </EnableLookupFromCentralStorageLocation>
       <!-- Legt das zentrale Verzeichnis für anwenderdefinierte Dokumentation fest. -->
       <CentralStorageLocation>
        <Value>D:\CorporateDocumentation\UserDocumentation\</Value>
       </CentralStorageLocation>
      </AttributeList>
     </Settings.UserDocumentation>
<!-- Sektion anwenderdefinierte Dokumentation Ende -->

    </ObjectList>
   </Settings.General>
  </ObjectList>
 </Settings.Settings>
</Document>
```

Siehe auch

Anwenderdefinierte Dokumentation verwenden (Seite 109)
Einstellungen im TIA Portal festlegen (Seite 112)
Anwenderdefinierte Dokumentation aufrufen (Seite 120)
Einstiegsseite erstellen (Seite 116)

9.4 Einstiegsseite erstellen

Für die anwenderdefinierte Dokumentation können Sie eine Einstiegsseite gestalten. Die Einstiegsseite kann eine HTML-Seite sein, die Sie entweder innerhalb einer CHM-Datei oder im Verzeichnis der jeweiligen Sprache speichern. Auch andere für anwenderdefinierte Dokumentation zulässige Dateiformate sind möglich. Die Einstiegsseite der anwenderdefinierten Dokumentation gestalten Sie außerhalb des TIA Portals.

Vorgehen

Um eine Einstiegsseite zu erstellen, gehen Sie folgendermaßen vor:

1. Gestalten Sie eine Datei im HTML-Format oder in einem anderen für anwenderdefinierte Hilfe zulässigen Dateiformat.
2. Benennen Sie die Datei mit dem Namen "Home".
3. Kopieren Sie die Datei in das zentrale Verzeichnis für anwenderdefinierte Dokumentation auf der Festplatte oder einem Netzlaufwerk:
   <Zentrales Verzeichnis für anwenderdefinierte Dokumentation>\<Ordner für die jeweilige Sprache>
4. Falls der jeweilige Sprachordner noch nicht existiert, legen Sie den Ordner an.
   Alternativ: Wenn Sie die Einstiegsseite für eine CHM-Datei erstellen, dann platzieren Sie die Einstiegsseite im Hauptverzeichnis der CHM-Datei.

Beispielkonfiguration für die Einstiegsseite

Im Folgenden sehen Sie den korrekten Pfad für folgende Bedingungen:

- Die anwenderdefinierte Dokumentation ist auf Spanisch.
- Die Einstiegsseite ist eine HTML-Datei.

Der Pfad für diese Bedingungen lautet:

<Zentrales Verzeichnis für anwenderdefinierte Dokumentation>\es-ES\Home.html

Siehe auch

Einstellungen mit einer XML-Datei festlegen (Seite 113)

Anwenderdefinierte Dokumentation verwenden (Seite 109)

Konventionen für das Erstellen (Seite 117)

Anwenderdefinierte Dokumentation aufrufen (Seite 120)

Aufrufprotokoll anzeigen (Seite 121)

Anwenderdefinierte Dokumentation erstellen (Seite 122)

9.5 Konventionen für das Erstellen

Damit die anwenderdefinierte Dokumentation an der richtigen Stelle aufgerufen wird, müssen Sie einige Konventionen einhalten:

- Die anwenderdefinierte Dokumentation muss im richtigen Verzeichnis gespeichert sein.
- Der Dateiname muss exakt der Objektbezeichnung im TIA Portal entsprechen.

Um zu verhindern, dass Schadcode auf Ihrem Computer ausgeführt wird, sind nur als weitestgehend sicher eingestufte Dateiformate zulässig.

Unterstützte Dateiformate

Die anwenderdefinierte Dokumentation legen Sie in einem der folgenden Dateiformate an:

- Microsoft Word (.docx)
- Microsoft Excel (.xlsx)
- Microsoft Powerpoint (.pptx und .ppsx)
- HTML-Seiten (.htm oder .html)
- Microsoft XPS (.xps)
- Rich Text Format (.rtf)
- Textdokumente (.txt)
- Compiled HTML Help (.chm)
- PDF-Dokumente (.pdf)

Die Einstiegsseite der anwenderdefinierten Dokumentation speichern Sie im HTML-Format (.htm oder .html) oder Sie speichern die Einstiegsseite innerhalb einer CHM-Datei.

ACHTUNG
Infektion des Computers mit Schadcode Wenn die anwenderdefinierte Dokumentation Schadcode enthält, können Sie Ihren Computer damit infizieren. Insbesondere HTML-Seiten und CHM-Dateien können Schadcode enthalten. Stellen Sie sicher, dass die anwenderdefinierte Dokumentation von einer vertrauenswürdigen Quelle stammt. Treffen Sie zusätzlich die üblichen Sicherheitsvorkehrungen, wie z. B. die Verwendung einer Firewall und eines aktuellen Virenscanners.

Verzeichnisse für anwenderdefinierte Dokumentation

Die anwenderdefinierte Dokumentation legen Sie in einem der folgenden Verzeichnisse ab:

- Projektverzeichnis:
  UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>\<Objektkategorie>
- Verzeichnis einer globalen Bibliothek:
  UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>\<Objektkategorie>
- Zentrales Verzeichnis auf der Festplatte oder einem Netzlaufwerk:
  <Zentrales Verzeichnis für anwenderdefinierte Dokumentation>\<Ordner für die jeweilige Sprache>\<Objektkategorie>\

Die anwenderdefinierte Dokumentation muss im passenden Unterordner für die jeweilige Sprache platziert sein. Die folgende Tabelle zeigt die jeweiligen Sprachordner für die standardmäßig installierten Oberflächensprachen:

Sprache	Unterordner
Deutsch	\de-DE
Englisch	\en-US
Spanisch	\es-ES
Französisch	\fr-FR
Italienisch	\it-IT
Chinesisch	\zh-CN

Der Sprachordner muss für jede Objektkategorie einen eigenen Unterordner enthalten. Legen Sie die entsprechenden Unterordner für die Objekte an, für die Sie anwenderdefinierte Dokumentation bereitstellen. Verwenden Sie immer die englische Bezeichnung der Objektkategorie. Die folgende Tabelle zeigt die englischen Bezeichnungen der wichtigsten Objektkategorien im TIA Portal:

Objektkategorie	Englische Bezeichnung
HMI Bilder	Screens
Organisationsbausteine (OBs)	Organization Blocks
Funktionsbausteine (FBs)	Function Blocks
Funktionen (FCs)	Functions
Datenbausteine	Data Blocks
Typen in der Bibliothek	Library Types
Kopiervorlagen in der Bibliothek	Master Copies
Der Projektknoten in der Projektnavigation	Projects
Alle Arten von Ordnern in der Projektnavigation, in der Projektbibliothek oder in globalen Bibliotheken	Folders
Alle Arten von Links in der Projektnavigation, z. B. "Neuen Baustein hinzufügen", "Neues Gerät hinzufügen", usw.	ShortCut
Bibliotheken in der Task Card "Bibliotheken" oder der Bibliotheksansicht	Libraries

Wenn Sie sich nicht sicher sind, wie die englische Bezeichnung einer Objektkategorie lautet, schalten Sie die Oberflächensprache des TIA Portals auf Englisch um. Alternativ rufen Sie die anwenderdefinierte Dokumentation zu einem Objekt mit <Shift+F1> auf und sehen Sie im Aufrufprotokoll nach, welche Bezeichnung für die Objektkategorie erwartet wird.

Zulässige Dateinamen

Der Dateiname muss exakt dem Namen des Objekts im TIA Portal entsprechen.

Unter Microsoft Windows existieren jedoch Beschränkungen für Dateinamen. Ebensolche Beschränkungen gelten für das Dateisystem, mit dem die Festplatte formatiert ist. Der Dateiname darf nur bestimmte Zeichen enthalten und eine bestimmte Länge nicht überschreiten. Die Beschränkungen für Dateinamen unterscheiden sich abhängig von der Windows-Version und dem verwendeten Dateisystem der Festplatte.

Um sicherzugehen, dass der Hilfeaufruf funktioniert, lesen Sie mögliche Beschränkungen in der Dokumentation zu Microsoft Windows nach.

Besonderheiten bei CHM-Dateien

CHM-Dateien legen Sie direkt im jeweiligen Sprachordner ab. Die Ordner für die jeweilige Objektkategorie müssen in der CHM-Datei selbst enthalten sein. Innerhalb der kompilierten CHM-Datei müssen die Namen der einzelnen HTML-Dateien ebenfalls exakt dem Objektnamen im TIA Portal entsprechen.

Hinweis

Öffnen von CHM-Dateien auf Netzlaufwerken

Wenn CHM-Dateien auf einem Netzlaufwerk gespeichert sind, dann werden die CHM-Dateien in neueren Versionen von Microsoft Windows nicht korrekt angezeigt. Dieses Verhalten ist durch Sicherheitsrichtlinien des Betriebssystems bestimmt. Alle Versionen von Microsoft Windows ab Windows Server 2003 SP1 sind betroffen.

Die Sicherheitsrichtlinien können Sie durch eine Änderung der Registrierungsdatenbank in Microsoft Windows umgehen.

Um die Sicherheit Ihres Computers nicht zu gefährden, speichern Sie CHM-Dateien nur lokal auf Ihrem Computer ab und lassen Sie die Registrierungsdatenbank unverändert.

Siehe auch

Anwenderdefinierte Dokumentation verwenden (Seite 109)

Einstiegsseite erstellen (Seite 116)

9.6 Anwenderdefinierte Dokumentation aufrufen

Die anwenderdefinierte Dokumentation wird in der Sprache geöffnet, die aktuell als Oberflächensprache eingestellt ist. Wenn in der aktuell eingestellten Oberflächensprache keine anwenderdefinierte Dokumentation existiert, wird die englische anwenderdefinierte Dokumentation geöffnet. Wenn keine anwenderdefinierte Dokumentation existiert, wird nach einer Einstiegsseite gesucht.

Voraussetzung

Sie haben bereits anwenderdefinierte Dokumentation oder eine Einstiegsseite entsprechend den Konventionen gespeichert.

Vorgehen

Um die anwenderdefinierte Dokumentation zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Objekt, zu dem Sie die anwenderdefinierte Dokumentation anzeigen möchten.
2. Drücken Sie <Shift+F1>.
   Die passende anwenderdefinierte Dokumentation oder die Einstiegsseite wird geöffnet.

Siehe auch

Anwenderdefinierte Dokumentation verwenden (Seite 109)

Einstiegsseite erstellen (Seite 116)

Einstellungen mit einer XML-Datei festlegen (Seite 113)

Aufrufprotokoll anzeigen (Seite 121)

9.7 Aufrufprotokoll anzeigen

Ob Sie die anwenderdefinierte Dokumentation richtig angebunden haben, kontrollieren Sie mit dem Aufrufprotokoll. Das Aufrufprotokoll zeigt, in welchen Verzeichnissen nach anwenderdefinierter Dokumentation oder eine Einstiegsseite gesucht wird. Außerdem wird im Aufrufprotokoll angezeigt, welchen Namen die einzelnen Dateien besitzen müssen, um die anwenderdefinierte Dokumentation aufzurufen.

Voraussetzung

Das Aufrufprotokoll ist in den Einstellungen des TIA Portals oder mithilfe einer XML-Konfigurationsdatei aktiviert.

Vorgehen

Um das Aufrufprotokoll anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Register "Info" im Inspektorfenster.
2. Öffnen Sie das Register "Allgemein".
3. Selektieren Sie das Objekt, zu dem Sie Hilfe aufrufen möchten.
4. Drücken Sie die Tastenkombination <Shift+F1>.
   Wenn möglich, wird die passende anwenderdefinierte Dokumentation oder die Einstiegsseite der anwenderdefinierten Dokumentation aufgerufen. In jedem Fall werden Sie im Inspektorfenster darüber informiert, welche anwenderdefinierte Dokumentation aufgerufen wird. Gegebenenfalls wird angezeigt, in welchen Verzeichnissen keine anwenderdefinierte Dokumentation gefunden wurde.

Siehe auch

Anwenderdefinierte Dokumentation aufrufen (Seite 120)

Einstiegsseite erstellen (Seite 116)

9.8 Anwenderdefinierte Dokumentation erstellen

Anwenderdefinierte Dokumentation zu einzelnen Elementen innerhalb eines Projekts oder einer globalen Bibliothek erstellen Sie außerhalb des TIA Portals. Sie können anwenderdefinierte Dokumentation in allen verfügbaren Oberflächensprachen erstellen.

Wenn Sie die anwenderdefinierte Dokumentation als CHM-Datei erstellen, weicht das Vorgehen beim Erstellen der Hilfe geringfügig vom Erstellungsprozess bei anderen Dateiformaten ab.

Beachten Sie die Hinweise im Kapitel "Konventionen für das Erstellen (Seite 117)".

Anwenderdefinierte Dokumentation als einzelne Datei erstellen

Um anwenderdefinierte Dokumentation als einzelne Datei zu erstellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine Datei in einem zulässigen Dateiformat.
2. Benennen Sie die Datei identisch mit dem Objekt, zu dem Sie anwenderdefinierte Dokumentation aufrufen möchten.
   Wenn Sie z. B. Hilfe zu einem Bibliothekstypen anbieten, benennen Sie die Hilfedatei identisch mit dem Typen.
3. Kopieren Sie die Datei an einen der folgenden Ablageorte, abhängig davon, ob Sie die anwenderdefinierte Dokumentation für Projektinhalte oder Inhalte einer globalen Bibliothek erstellen:
   - Projektverzeichnis unter "UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>\<Objektkategorie>"
   - Verzeichnis einer globalen Bibliothek unter "UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>\<Objektkategorie>"
   - Zentrales Verzeichnis auf der Festplatte oder einem Netzlaufwerk:
     <Zentrales Verzeichnis für anwenderdefinierte Dokumentation>\<Ordner für die jeweilige Sprache>\<Objektkategorie>\

   Falls der jeweilige Sprachordner oder der Ordner für die Objektkategorie noch nicht existiert, legen Sie die notwendigen Ordner vor dem Kopieren der Datei an.

Beispielkonfiguration für anwenderdefinierte Dokumentation

Im Folgenden sehen Sie den korrekten Pfad für folgende Bedingungen:

- Die anwenderdefinierte Dokumentation ist für einen Typen in einer globalen Bibliothek bestimmt.
- Die anwenderdefinierte Dokumentation ist auf Französisch.
- Der Typ heißt "commande de moteur".
- Die anwenderdefinierte Dokumentation wird mit der globalen Bibliothek mitgeliefert.
- Die anwenderdefinierte Dokumentation ist im Microsoft Power Point-Format erstellt.

Der Pfad für diese Bedingungen lautet:

<Ordner der globalen Bibliothek>\UserFiles\UserDocumentation\fr-FR\Library Types \commande de moteur.pptx

Anwenderdefinierte Dokumentation als CHM-Datei erstellen

Um anwenderdefinierte Dokumentation im CHM-Format bereitzustellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie für jede Objektkategorie, für die Sie anwenderdefinierte Dokumentation erstellen möchten, einen Ordner im Windows Explorer. Verwenden Sie die englische Bezeichnung für die Objektkategorie.
2. Erstellen Sie für jedes Objekt, das Sie mit anwenderdefinierte Dokumentation versehen möchten, eine HTML-Datei. Benennen Sie die HTML-Datei identisch mit dem Objekt, zu dem Sie anwenderdefinierte Dokumentation aufrufen möchten. Wenn Sie z. B. anwenderdefinierte Dokumentation zu einem Bibliothekstypen anbieten möchten, benennen Sie die HTML-Datei identisch mit dem Typen.
3. Legen Sie die HTML-Dateien in den entsprechenden Ordnern der jeweiligen Objektkategorie ab.
4. Erzeugen Sie mithilfe des Microsoft HTML Help Workshop die CHM-Datei. Übernehmen Sie dabei die vorbereitete Ordnerstruktur mit den HTML-Dateien.
5. Kopieren Sie die CHM-Datei an einen der folgenden Ablageorte:
   - Projektverzeichnis unter "UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>"
   - Verzeichnis einer globalen Bibliothek unter "UserFiles\UserDocumentation\<Ordner für die jeweilige Sprache>"
   - Zentrales Verzeichnis auf der Festplatte oder einem Netzlaufwerk: <Zentrales Verzeichnis für anwenderdefinierte Dokumentation>\<Ordner für die jeweilige Sprache>

   Falls der jeweilige Sprachordner noch nicht existiert, legen Sie den Sprachordner vor dem Kopieren der CHM-Datei an.

Siehe auch

Einstiegsseite erstellen (Seite 116)

Konventionen für das Erstellen (Seite 117)

Index

A

Anwenderdefinierte Dokumentation, 109
    Aufrufprotokoll aktivieren, 112, 113
    Einstellungen festlegen, 112, 113
    Zentrales Ablageverzeichnis festlegen, 112, 113
Anwenderdefinierte Dokumentation aufrufen, 120
Anwenderdefinierte Dokumentation erstellen, 122
Anwenderdefinierte Textlisten
    anlegen, 95
    bearbeiten, 96
    Wertebereich und Texte bearbeiten, 96
Anwendertexte, 78
Aufrufprotokoll anzeigen, 121

B

Bausteinvergleich
    Aktionen ausführen, 39
    Vergleichsergebnisse aktualisieren, 36
Beispiel für anwenderdefinierte Dokumentation, 122
Beispiel für Einstiegsseite anwenderdefinierter Dokumentation, 116
Beschriftungsdaten als XML exportieren, 71
Beschriftungsstreifen drucken, 69
    Korrekturwert für Druckverschiebung ermitteln, 75
Beschriftungsstreifen erstellen, 67
Besonderheiten bei CHM-Dateien, 119

C

CHM-Datei erstellen, 123

D

Deckblatt
    Platzhalter für Dokument-Informationen, 61
Detailvergleich
    durchführen, 35
Dokumentationseditor
    Aufbau, 56
Dokumentationsfunktion
    Fertige Rahmen und Deckblätter verwenden, 54
Dokument-Information, 51
Druck abbrechen, 66
Druckbare Inhalte, 48
Drucken
    Aufbau des Ausdrucks, 47
    Deckblatt anlegen, 54
    Deckblatt bearbeiten, 55
    Deckblatt nutzen, 50
    Dokumentationsfunktion, 47
    Druckfläche festlegen, 59
    Druckinhalte, 51
    Druckvorlagen verwenden, 50
    Einstellungen ändern, 49
    Elemente in der Bibliothek, 53
    Elemente in der Projektnavigation, 52
    Layout festlegen, 50
    Nicht druckbare Elemente, 48
    Nicht druckbare Objekte, 49
    Rahmen anlegen, 53
    Rahmen bearbeiten, 55
    Rahmen nutzen, 52
    Rahmen und Deckblätter verwenden, 47
Druckumfang, 48
Druckvorschau erstellen, 61

E

Editiersprache, 77
Einstellungen für Anwenderhilfe, 113
Einstiegsseite der anwenderdefinierten Dokumentation erstellen, 116
Export von Beschriftungsstreifen, 67

H

Hilfe
    Eigene Hilfeinhalte bereitstellen, 109
    Einstellungen für anwenderdefinierte Dokumentation, 112, 113

K

Konventionen für anwenderdefinierte Dokumentation, 117

L

Laden
    Allgemeine Informationen, 9

auf eine Memory Card laden, 13
auf eine Memory Card schreiben, 13
Projektdaten in Gerät laden, 11
von einem Gerät, 15

M

Memory Card
Eigenschaften anzeigen, 104
Einführung, 101
Kartenleser hinzufügen, 102
zugreifen, 103

P

Passwortschutz
Schutzkonzept, 43
Zugriffsrechte zurücknehmen, 45
PLC-Variable
Vergleichsergebnisse aktualisieren, 36
Projekt
Daten vergleichen, 19
Detailvergleich, 35
Offline/Offline-Vergleich durchführen, 24
Offline/Online-Vergleich durchführen, 23
Projektdaten übersetzen, 8
Projekt drucken, 65
Projektdaten
auf eine Memory Card laden, 13
auf eine Memory Card schreiben, 13
Informationen zum Laden, 9
Informationen zum Übersetzen, 7
laden, 11
übersetzen, 8
von einem Gerät laden, 15
Projektsprache, 77
Projektsprachen
Editiersprache ändern, 82
Projektsprachen festlegen, 81
Verwendung, 77
Projekttexte
Alle Texte exportieren, 87
Anwendungsbeispiel, 90
Einzelne Texte exportieren, 86
importieren, 89
Projekttexte übersetzen, 83
Referenztext anzeigen, 85
Texte eines Geräts exportieren, 87
Texte von selektierten Objekten ändern, 84
Übersetzung in Projektsprachen, 77

R

Rahmen
Druckfläche festlegen, 59
Platzhalter für Dokument-Informationen, 61
Rahmen und Deckblätter aus der Bibliothek nutzen, 54
Referenzsprache, 77

S

Schutzkonzept
Einführung, 43
Zugriffsrechte zurücknehmen, 45
Schutzstufe
Zugriffsrechte zurücknehmen, 45
Spalten ein- und ausblenden, 33
Systemdefinierte Textlisten
bearbeiten, 97
Texte ändern, 97
Systemtexte, 78

T

Texte
Alle Texte exportieren, 87
Einzelne Texte exportieren, 86
importieren, 89
Texte eines Geräts exportieren, 87
Textlisten
anwenderdefiniert, 94
Einführung, 93
systemdefiniert, 94
Textlisten-Editor, 93

U

Übersetzen von Texten, 77
Unterstützte Dateiformate, 117

V

Vergleichseditor
Aktionen festlegen, 38
Ansicht ändern, 40
Ansicht filtern, 34
Spalten ein- und ausblenden, 33
Übersicht, 26

Verzeichnisse für anwenderdefinierte
Dokumentation, 118

Z

Zulässige Dateinamen, 119

SIEMENS

Benutzerverwaltung verwenden

Programmier- und Bedienhandbuch

Grundlagen zur Benutzerverwaltung im TIA Portal 1

Benutzerverwaltung aktivieren 2

Passwort-Richtlinien festlegen 3

Projektbenutzer verwalten 4

Globale Benutzer und Benutzergruppen verwalten 5

Rollen verwalten 6

Rollen zuweisen 7

An einem geschützten Projekt anmelden 8

Eigenes Passwort ändern 9

Vom Projekt abmelden 10

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Grundlagen zur Benutzerverwaltung im TIA Portal....5
2 Benutzerverwaltung aktivieren....9
3 Passwort-Richtlinien festlegen....11
4 Projektbenutzer verwalten....13
5 Globale Benutzer und Benutzergruppen verwalten....17
6 Rollen verwalten....23
7 Rollen zuweisen....27
8 An einem geschützten Projekt anmelden....29
9 Eigenes Passwort ändern....31
10 Vom Projekt abmelden....33

Grundlagen zur Benutzerverwaltung im TIA Portal 1

Einführung

Das TIA Portal bietet die Möglichkeit, für Projekte eine Benutzerverwaltung zu verwenden. Dadurch kann z. B. ein Projekt vor unbeabsichtigtem oder unberechtigtem Ändern geschützt werden. Um die Benutzerverwaltung zu aktivieren, richtet ein Benutzer den Projektschutz ein. Dieser Benutzer wird dadurch als Projektadministrator angelegt. Nach dem Aktivieren des Projektschutzes kann das Projekt nur noch von berechtigten Benutzern geöffnet und bearbeitet werden. Beachten Sie, dass der Projektschutz nicht wieder aufgehoben werden kann.

Als Benutzer sind lokale Projektbenutzer und globale Benutzer und Benutzergruppen zulässig, die durch einen Projektadministrator hinzugefügt werden:

- Lokale Projektbenutzer:
  Benutzer, die in einem TIA Portal-Projekt definiert und verwaltet werden. Diese Benutzerkonten sind nur für dieses eine Projekt gültig. Die Verwendung von Projektbenutzerkonten ist sinnvoll, wenn die gesamte Automatisierungslösung in einem Projekt engineered wird.
- Globale Benutzer und Benutzergruppen:
  Diese Benutzerkonten werden außerhalb des TIA Portals in UMC (User Management Component) definiert und verwaltet. Die globalen Benutzer und Benutzergruppen können Sie in die verschiedenen TIA Portal-Projekte importieren, mit denen diese Benutzer arbeiten sollen. Für das Hinzufügen von Benutzern und Benutzergruppen aus UMC sind die entsprechenden Rechte in UMC notwendig.

Den Benutzern oder Benutzergruppen können Sie bestimmte Rollen zuweisen, mit denen wiederum verschiedene Funktionsrechte verknüpft sein können. Es gibt die folgenden allgemeinen Engineering-Funktionsrechte, die einer Rolle zugewiesen werden können:

- Projekt schreibgeschützt öffnen
  Ein Benutzer, der ausschließlich über dieses Funktionsrecht verfügt, kann das Projekt zwar öffnen, aber nicht verändern. Die Editoren der Benutzerverwaltung kann er jedoch nicht öffnen.
- Projekt mit Schreibrechten öffnen
  Ein Benutzer mit diesem Funktionsrecht darf das Projekt verändern. Die Editoren der Benutzerverwaltung kann er damit jedoch nicht öffnen.
- Benutzer und Rollen verwalten
  Ein Benutzer mit diesem Funktionsrecht kann die Benutzer und Rollen des Projekts verwalten. Zusätzlich benötigt er dafür noch das Funktionsrecht "Projekt mit Schreibrechten öffnen", da mit dem Funktionsrecht "Benutzer und Rollen verwalten" alleine das Projekt nicht geöffnet oder bearbeitet werden kann.

Zusätzlich zu diesen allgemeinen Funktionsrechten können weitere Runtime-Rechte vergeben werden. Die Beschreibungen dieser spezifischen Funktionsrechte finden Sie in den entsprechenden Bereichen der Hilfe. Einer Rolle können mehrere Funktionsrechte zugewiesen werden.

Das System legt beim Aktivieren der Benutzerverwaltung die folgenden beiden Rollen an:

- "ES Administrator"
  Dem zuerst angelegten Projektbenutzer wird die Rolle "ES Administrator" zugewiesen. Diese Rolle verfügt standardmäßig über alle drei Engineering-Funktionsrechte. Jedes Projekt benötigt mindestens einen Administrator, der das Projekt und die Security-Einstellungen bearbeiten darf. Zusätzlich können weitere Benutzer das Recht erhalten, Benutzer und Rollen zu verwalten.
- "ES Standard"
  Benutzer, denen die Rolle "ES Standard" zugewiesen ist, verfügen über die Rechte "Projekt lesend öffnen" und "Projekt lesend/schreibend" öffnen.

Diese Systemrollen können Sie nicht ändern oder löschen. Sie können weitere Rollen anlegen und diese mit den benötigten Funktionsrechten versehen.

Hinweis

Zusätzliche lokale Benutzerverwaltungen

Neben der Benutzerverwaltung für Projekte gibt es in bestimmten Bereichen des TIA Portals zusätzliche Benutzerverwaltungen, z. B. für WinCC Panels.

Einstellungen für Benutzer

Für einen lokalen Projektbenutzer kann ein Projektadministrator die folgenden Einstellungen festlegen:

- Benutzername:
  Name des lokalen Projektbenutzers, der für die Anmeldung am Projekt verwendet werden muss.
- Passwort:
  Das vom Administrator vergebene Passwort des lokalen Projektbenutzers, mit dem sich der Projektbenutzer am Projekt anmelden kann. Der Projektbenutzer kann das Passwort nachträglich ändern.
- Authentifizierungsverfahren:
  Vom Administrator festgelegte Authentifizierungsverfahren. Es stehen die folgenden beiden Verfahren zur Auswahl:
  - Passwort: Die Anmeldung erfolgt über ein Passwort, das im TIA Portal oder in UMC festgelegt wurde.
  - Radius: Die Anmeldung erfolgt über einen RADIUS-Server, auf dem das Passwort hinterlegt ist. Diese Option kann nur für Geräte verwendet werden, die eine Anmeldung über einen RADIUS-Server unterstützen.
- Maximale Sitzungsdauer
  Maximale Dauer, für die der Benutzer an einem Gerät angemeldet sein kann. Anschließend wird der Benutzer vom Gerät abgemeldet. Diese Einstellung ist nur für Geräte verfügbar, die eine Sitzungsdauer unterstützen. Im TIA Portal selbst wird die Sitzungsdauer nicht ausgewertet.

UMC - User Management Component

Zusätzlich können Sie das Softwarepaket "User Management Component UMC" auf einem oder mehreren Rechnern installieren, das eine zentrale Benutzerverwaltung zur Verfügung stellt. Dadurch entsteht ein System von gegebenenfalls zusammenwirkenden UMC-Installationen (UMC Ring-Server, UMC-Server). Danach können Sie in diesem UMC-System Benutzer und Benutzergruppen festlegen oder aus einem Windows Active Directory übernehmen. Bei installiertem UMC können Sie vom TIA Portal aus auf den UMC-Server zugreifen, um die dort definierten Benutzer und Benutzergruppen zur Benutzerverwaltung des TIA Portals hinzuzufügen und ihnen über Rollen die benötigten Funktionsrechten an einem TIA Portal-Projekt zuzuweisen.

Sie können die Daten der aus UMC hinzugefügten Benutzer und Benutzergruppen jedoch nicht innerhalb des TIA Portals verändern. So können Sie z. B. auch als Administrator im Projekt keine Passwörter oder sonstige Daten von UMC-Benutzern oder UMC-Benutzergruppen ändern. Dies ist nur in UMC möglich. Sie haben aber die Möglichkeit, die Benutzerverwaltung im TIA Portal mit UMC zu synchronisieren oder den Synchronisationsstatus zu prüfen. Dadurch können Sie Inkonsistenzen zwischen den globalen Benutzern und Benutzergruppen in UMC und den bereits ins TIA Portal importierten UMC-Benutzern bzw. UMC-Benutzergruppen beheben.

Hinweis

UMC-Dokumentation

Die Installationsdatei für UMC und die englischsprachige Dokumentation zu UMC finden Sie auf dem TIA Portal-Installationsdatenträger ("..\support", "...\Documents\Readme\English").

Es wird dringend empfohlen, die UMC-Dokumentation vor Verwendung der Benutzerverwaltung vollständig zu lesen, vor allem die Abschnitte zum Thema "Secure Application Data Support (SADS)". SADS ist für die Verwendung der Benutzerverwaltung im TIA Portal zwingend erforderlich.

Benutzerverwaltung im TIA Portal und Multiuser Engineering

Beachten Sie bei der Verwendung der Benutzerverwaltung im TIA Portal in Verbindung mit Multiuser Engineering die folgenden Hinweise:

- Änderungen an der Benutzerverwaltung eines Multiuser-Projekts können Sie nur in einer Server-Session vornehmen, nicht in den lokalen Sessions.
- Wenn Sie ein Single-User-Projekt, das mithilfe der Benutzerverwaltung geschützt ist, auf einen Multiuser-Server hochladen, ist das daraus entstehende Multiuser-Projekt ebenfalls geschützt. Die im Projekt enthaltenen Benutzer und Benutzergruppen bleiben mit ihren Passwörtern bestehen und können sich damit in den lokalen Sessions des Multiuser-Projekts anmelden, wenn sie über die entsprechenden Funktionsrechte verfügen.
- Wenn Sie eine zuvor ungeschützte lokale Session mithilfe der Benutzerverwaltung schützen, wird dieser Schutz nach dem Einchecken und Aktualisieren der lokalen Session wieder entfernt, da das Multiuser-Projekt auf dem Server dadurch nicht geschützt wird. Dies gilt auch, falls mehrere Bearbeiter ihre lokalen Sessions schützen.
- Wenn Sie das Multiuser-Projekt auf dem Server mithilfe der Benutzerverwaltung schützen, werden auch die von diesem Multiuser-Projekt erstellten lokalen Sessions geschützt, sobald diese aktualisiert werden.

- Wenn Sie sich von Ihrer lokalen Session, die mithilfe der Benutzerverwaltung geschützt ist, abmelden, bleiben andere Benutzer an ihren lokalen Sessions des Multiuser-Projekts angemeldet. Sie melden nur sich selbst ab. Ihre lokale Session wird dadurch geschlossen.
- Falls während der Bearbeitung Ihrer ungeschützten lokalen Session das Multiuser-Projekt auf dem Server mithilfe der Benutzerverwaltung geschützt wird und Ihrem Benutzerkonto nicht das Funktionsrecht "Projekt lesend/schreibend öffnen" zugewiesen wurde, können Sie keine Änderungen an Ihrer lokalen Session einchecken oder Ihre lokale Session aktualisieren.

Weitere Informationen zum Multiuser Engineering finden Sie im Kapitel "Multiuser Engineering einsetzen".

Siehe auch

Benutzerverwaltung aktivieren (Seite 9)

Passwort-Richtlinien festlegen (Seite 11)

Projektbenutzer verwalten (Seite 13)

Globale Benutzer und Benutzergruppen verwalten (Seite 17)

Rollen verwalten (Seite 23)

Rollen zuweisen (Seite 27)

An einem geschützten Projekt anmelden (Seite 29)

Eigenes Passwort ändern (Seite 31)

Vom Projekt abmelden (Seite 33)

Benutzerverwaltung aktivieren 2

Sie können die Benutzerverwaltung in einem TIA Portal-Projekt verwenden, um Ihr Projekt vor unbefugten Zugriffen zu schützen. Mit dem Aktivieren der Benutzerverwaltung wird automatisch ein Projektadministrator angelegt. Dieser kann weitere Benutzer anlegen. Wenn die Benutzerverwaltung aktiviert ist, kann das Projekt nur noch mit einem Benutzerkonto mit ausreichenden Rechten geöffnet und geändert werden.

Hinweis

Deaktivierung der Benutzerverwaltung nicht möglich

Beachten Sie, dass die Benutzerverwaltung für das Projekt nicht deaktiviert werden kann, wenn sie aktiviert wurde. Das bedeutet auch, dass das Schützen des Projekts nicht rückgängig gemacht werden kann.

Vorgehen

Um die Benutzerverwaltung zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Einstellungen".
   Der Editor für die Benutzerverwaltung wird geöffnet und der Bereich für den Projektschutz wird angezeigt.
3. Klicken Sie auf "Dieses Projekt schützen".
   Der Dialog "Projektschutz" wird geöffnet.
4. Geben Sie einen Benutzernamen ein.
5. Geben Sie ein Passwort ein.

   Hinweis

   Passwort-Richtlinie

   Beim ersten Aktivieren der Benutzerverwaltung sind für das Passwort die folgenden Richtlinien definiert:

   - Minimale Passwortlänge: 8 Zeichen
   - Mindestanzahl an Ziffern: 1
   - Mindestanzahl an Sonderzeichen: 0
   - Mindestens ein Groß- und ein -Kleinbuchstabe: Aktiviert
   - Anzahl der zur Wiederverwendung gesperrten Passwörter: 5
   - Passwortalterung aktivieren: deaktiviert

   Als Projektadministrator können Sie diese Passwort-Richtlinien anschließend an Ihre Erfordernisse anpassen.

   ---

6. Geben Sie zur Bestätigung das Passwort erneut ein.

7. Geben Sie bei Bedarf einen Kommentar ein.
8. Bestätigen Sie Ihre Eingaben mit "OK".

Hinweis

Projektadministrator

Der erste angelegte Benutzer ist der Projektadministrator. Jedes Projekt benötigt mindestens einen Administrator.

Ergebnis

Die Benutzerverwaltung wurde aktiviert. Sie sind als Projektadministrator angemeldet und können nun weitere lokale Benutzer anlegen oder globale Benutzer oder Benutzergruppen aus UMC hinzufügen oder neue Rollen erstellen.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Passwort-Richtlinien festlegen 3

Als Projektadministrator können Sie den Aufbau und die Komplexität der Benutzerpasswörter definieren. Dazu können Sie die folgenden Richtlinien vorgeben:

- "Minimale Passwortlänge"
  Die Mindestanzahl an Zeichen, die das Benutzerpasswort haben muss.
- "Minimale Anzahl an Ziffern"
  Die Mindestanzahl an Ziffern, die das Benutzerpasswort enthalten muss.
- "Minimale Anzahl an Sonderzeichen"
  Die Mindestanzahl an Sonderzeichen, die das Benutzerpasswort enthalten muss.
- "Mindestens ein Groß- und ein Kleinbuchstabe"
  Legt fest, dass das Benutzerpasswort mindestens einen Groß- und einen Kleinbuchstaben enthalten muss.
- "Anzahl der zuletzt verwendeten gesperrten Passwörter"
  Legt die Anzahl der zuletzt verwendeten Passwörter fest, die nicht als neues Passwort verwendet werden dürfen.
- "Passwortalterung aktivieren"
  Legt fest, dass das Passwort nur eine bestimmte Gültigkeitsdauer besitzt und danach abläuft. Wenn diese Option aktiviert ist, muss das Passwort innerhalb der Passwortgültigkeit geändert werden.
- "Passwortgültigkeit (Tage)"
  Legt die Passwortgültigkeit in Tagen fest, in der das Passwort geändert werden muss, wenn die Passwortalterung aktiviert ist.
- "Vorwarnzeit (Tage)"
  Legt fest, wie viele Tage vor Ablauf des Passworts der Benutzer die Warnung erhält, dass das Passwort abläuft. Die Vorwarnzeit muss kleiner als die Passwortgültigkeit sein.

Hinweis

Erlaubte Zeichen für Benutzernamen und Passwörter

Sie können die folgenden Zahlen, Buchstaben und Sonderzeichen für Benutzernamen und Passwörter verwenden:

- 0123456789
- A...Z a...z
- !#$%&()*+,-./:;<=>?@ [\]_{|}~^

Voraussetzung

Sie verfügen über ein Benutzerkonto als Projektbenutzer oder als globaler Benutzer, das mit den beiden folgenden Funktionsrechten verknüpft ist:

- Projekt mit Schreibrechten öffnen

- Benutzer und Rollen verwalten

Vorgehen

Um die Passwort-Richtlinien festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das geschützte Projekt, für das Sie die Passwort-Richtlinien festlegen möchten.
2. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
3. Doppelklicken Sie auf den Befehl "Einstellungen".
   Der Editor für die Benutzerverwaltung wird geöffnet und der Bereich für den Projektschutz wird angezeigt.
4. Klicken Sie auf "Passwort-Richtlinie".
5. Nehmen Sie alle gewünschten Einstellungen vor.
6. Speichern Sie das Projekt.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Projektbenutzer verwalten 4

Um die lokalen Benutzer in Ihrem Projekt zu verwalten, können Sie die folgenden Aktionen durchführen:

- Neue Projektbenutzer anlegen
- Benutzernamen eines Projektbenutzers ändern
- Passwort eines Projektbenutzers ändern
- Authentifizierungsverfahren eines Projektbenutzers ändern
- Maximale Sitzungsdauer eines Projektbenutzers ändern
- Kommentar ändern
- Projektbenutzer löschen

Beachten Sie die folgenden Einschränkungen:

- Es können maximal 256 Projektbenutzer hinzugefügt werden.
- Der Benutzername darf maximal 255 Zeichen lang sein.
- Das Passwort darf maximal 120 Zeichen lang sein und muss den festgelegten Passwort-Richtlinien entsprechen.
- Der Kommentar darf maximal 1000 Zeichen lang sein.

Voraussetzung

- Ein geschütztes Projekt ist geöffnet.
- Sie sind mit einem Benutzerkonto angemeldet, das über die Funktionsrechte "Benutzer und Rollen verwalten" und "Projekt mit Schreibrechten öffnen" verfügt.

Neue Projektbenutzer anlegen

Um einen neuen Projektbenutzer anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie auf "Neuen Benutzer hinzufügen".
   Ein Untermenü wird geöffnet, in dem Sie den Benutzertyp auswählen können.
5. Klicken Sie auf "Neuen Projektbenutzer hinzufügen".
6. Geben Sie einen Benutzernamen ein.
7. Klicken Sie auf den Pfeil im Feld "Passwort".
8. Geben Sie ein Passwort ein.
9. Geben Sie zur Bestätigung das Passwort erneut ein.

10. Wählen Sie das Authentifizierungsverfahren und die maximale Sitzungsdauer für den neuen Projektbenutzer aus.
11. Geben Sie bei Bedarf einen Kommentar ein.
    Der neue Projektbenutzer wurde angelegt. Damit mit diesem Benutzerkonto im Projekt gearbeitet werden kann, sollten Sie ihm als Nächstes die notwendigen Rollen zuweisen.

Hinweis

Sie können einen neuen Projektbenutzer auch durch Kopieren eines vorhandenen Projektbenutzers anlegen. Dadurch werden die zugewiesenen Rollen auch dem kopierten Projektbenutzer zugewiesen. Sie müssen jedoch das Passwort für den kopierten Projektbenutzer neu vergeben.

Benutzernamen eines Projektbenutzers ändern

Um den Benutzernamen eines Projektbenutzers zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie in das Feld mit dem Benutzernamen, den Sie ändern möchten.
5. Geben Sie den neuen Benutzernamen ein.

Passwort eines Projektbenutzers ändern

Um das Passwort eines Projektbenutzers zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie auf den Pfeil neben dem Passwort, das Sie ändern möchten.
5. Geben Sie das neue Passwort ein.
6. Geben Sie zur Bestätigung das Passwort erneut ein.

Authentifizierungsverfahren eines Projektbenutzers ändern

Um das Authentifizierungsverfahren eines Projektbenutzers zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie auf den Pfeil neben dem Authentifizierungsverfahren, das Sie ändern möchten.
5. Wählen Sie das gewünschte Authentifizierungsverfahren aus.

Maximale Sitzungsdauer eines Projektbenutzers ändern

Um die maximale Sitzungsdauer eines Projektbenutzers zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Geben Sie die neue Zeitdauer für den Projektbenutzer ein, für den Sie die Sitzungsdauer ändern möchten. Alternativ können Sie die neue Zeitdauer über die Pfeiltasten auswählen.

Kommentar ändern

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie in das Feld mit dem Kommentar, den Sie ändern möchten.
5. Geben Sie den neuen Kommentar ein.

Projektbenutzer löschen

Um einen Projektbenutzer zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Selektieren Sie den Projektbenutzer, den Sie löschen möchten.
5. Wählen Sie im Kontextmenü den Befehl "Löschen" oder drücken Sie die Taste <Entf>.

Hinweis

Es muss mindestens ein Projektbenutzerkonto erhalten bleiben, dem die Rolle "ES Administrator" zugewiesen ist.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Globale Benutzer und Benutzergruppen verwalten 5

Sie können einem geschützten Projekt globale Benutzer und Benutzergruppen hinzufügen, die in UMC (User Management Component) angelegt wurden. Dadurch können diese Benutzer und Benutzergruppen mit dem Projekt arbeiten, wenn sie die entsprechenden Rechte zugewiesen bekommen.

Wenn ein Benutzer auf mehrere geschützte Projekte zugreifen muss, minimieren Sie Ihren Verwaltungsaufwand, wenn Sie diesen Benutzer entweder einzeln als globalen Benutzer oder zusammen mit einer Benutzergruppe in jedem Projekt hinzufügen. Globale Benutzer können in UMC verwaltet werden und die Änderungen wirken sich dann auf jedes Projekt aus, dem diese Benutzer oder Benutzergruppen angehören. Sie haben die Möglichkeit die Benutzerverwaltung im TIA Portal mit UMC zu synchronisieren, damit es nicht zu Inkonsistenzen kommt, falls Benutzer oder Benutzergruppen in UMC geändert wurden. Um zu ermitteln, ob eine Synchronisation notwendig ist, können Sie den Synchronisationsstatus prüfen. Informationen zur Synchronisation finden Sie im Banner in der Benutzerverwaltung.

Beachten Sie die folgenden Einschränkungen bei den globalen Benutzern und Benutzergruppen:

- Es können maximal 256 globale Benutzer hinzugefügt werden.
- Es können maximal 50 globale Benutzergruppen hinzugefügt werden.
- Globale Benutzer und Benutzergruppen können nicht kopiert werden.

Für globale Benutzer und Benutzergruppen können Sie sich die folgenden Informationen anzeigen lassen:

- Benutzergruppen: Die darin enthaltenen Benutzer und ob diese bereits in die Benutzerverwaltung importiert wurden
- Benutzer: Die Benutzergruppen, denen ein Benutzer angehört, und ob diese bereits in die Benutzerverwaltung importiert wurden

Voraussetzung

- Ein geschütztes Projekt ist geöffnet.
- Sie sind mit einem Benutzerkonto angemeldet, das über die Funktionsrechte "Benutzer und Rollen verwalten" und "Projekt mit Schreibrechten öffnen" verfügt.
- Die Verbindung zu UMC ist konfiguriert und Sie verfügen über ein Benutzerkonto mit den entsprechenden Rechten in UMC.
  Für weitere Informationen zur Installation und Konfiguration von UMC steht Ihnen die englischsprachige Dokumentation zu UMC auf dem Installationsdatenträger im Verzeichnis "/Documents/Readme/English" zur Verfügung.

Globale Benutzer hinzufügen

Um einen globalen Benutzer aus UMC hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Klicken Sie auf "Neuen Benutzer hinzufügen".
   Ein Untermenü wird geöffnet, in dem Sie den Benutzertyp auswählen können.
5. Klicken Sie auf "Neuen Benutzer aus UMC hinzufügen".
   Wenn Sie noch nicht in UMC angemeldet sind, wird der Dialog "Anmelden" geöffnet. Führen Sie dann zusätzlich die Schritte 6 und 7 aus.
6. Geben Sie Ihren UMC-Benutzernamen und das zugehörige Passwort ein.
7. Klicken Sie auf "OK".
   Sobald Sie in UMC angemeldet sind, wird der Dialog "Neuen Benutzer aus UMC hinzufügen" geöffnet. In diesem Dialog werden alle verfügbaren Benutzer aus UMC angezeigt. Bereits hinzugefügte Benutzer sind aktiviert und schreibgeschützt.
8. Aktivieren Sie die Benutzer, die Sie Ihrem TIA Portal-Projekt hinzufügen möchten.

   Hinweis

   Um die gewünschten Benutzer schneller zu finden, können Sie die Tabelle nach den Spalten "Namen" und "Langnamen" filtern. Geben Sie dazu in der ersten Zeile einen Teil des Namens oder Langnamens ein. Alle Benutzer, deren Namen oder Langnamen den eingegebenen Text enthalten, werden angezeigt. Um die Filterung wieder aufzuheben, klicken Sie auf die Filter-Schaltfläche oder wählen Sie "*" in der Klappliste.

9. Klicken Sie auf "OK", um die ausgewählten Benutzer hinzuzufügen.
   Die ausgewählten Benutzer werden als globale Benutzer hinzugefügt. Ihre Daten sind schreibgeschützt und können innerhalb des TIA Portal-Projekts nicht geändert werden. Das eigene Passwort kann ein globaler Benutzer jedoch ändern.

Globale Benutzer löschen

Um einen globalen Benutzer aus dem TIA Portal-Projekt zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Selektieren Sie den globalen Benutzer, den Sie löschen möchten. Sie können per Multiselektion auch mehrere Benutzer gleichzeitig auswählen.
5. Wählen Sie im Kontextmenü den Befehl "Löschen" oder drücken Sie die Taste <Entf>.

Globale Benutzergruppe hinzufügen

Um eine globale Benutzergruppe aus UMC hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".
4. Klicken Sie auf "Neue Benutzergruppe hinzufügen".
   Wenn Sie noch nicht in UMC angemeldet sind, wird der Dialog "Anmelden" geöffnet. Führen Sie dann zusätzlich die Schritte 5 und 6 aus.
5. Geben Sie Ihren UMC-Benutzernamen und das zugehörige Passwort ein.
6. Klicken Sie auf "OK".
   Sobald Sie in UMC angemeldet sind, wird der Dialog "Neue Benutzergruppe aus UMC hinzufügen" geöffnet. In diesem Dialog werden alle verfügbaren Benutzergruppen aus UMC angezeigt. Bereits hinzugefügte Benutzergruppen sind aktiviert und schreibgeschützt.
7. Aktivieren Sie die Benutzergruppen, die Sie Ihrem TIA Portal-Projekt hinzufügen möchten.
8. Klicken Sie auf "OK", um die ausgewählten Benutzergruppen hinzuzufügen.
   Die ausgewählten Benutzergruppen werden als globale Benutzergruppen hinzugefügt. Ihre Daten sind schreibgeschützt und können innerhalb des TIA Portal-Projekts nicht geändert werden.

Globale Benutzergruppe löschen

Um eine globale Benutzergruppe aus dem TIA Portal-Projekt zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".
4. Selektieren Sie die globale Benutzergruppe, die Sie löschen möchten. Sie können per Multiselektion auch mehrere Benutzergruppen gleichzeitig auswählen.
5. Wählen Sie im Kontextmenü den Befehl "Löschen" oder drücken Sie die Taste <Entf>.

Benutzer einer hinzugefügten globalen Benutzergruppe anzeigen

Um sich für eine hinzugefügte globale Benutzergruppe die enthaltenen Benutzer anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".

4. Selektieren Sie die Benutzergruppe, für die Sie die enthaltenen Benutzer anzeigen möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Öffnen Sie im unteren Bereich des Registers "Benutzergruppen" das Register "Benutzer". Die Benutzer der selektierten Gruppe werden angezeigt. Für Benutzer, die bereits in die Benutzerverwaltung importiert wurden, ist das Optionskästchen in der Spalte "Importiert" aktiviert.

Benutzergruppen eines globalen Benutzers anzeigen

Um sich für einen hinzugefügten globalen Benutzer die Benutzergruppen anzeigen zu lassen, denen er angehört, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Selektieren Sie den Benutzer, für den Sie die Benutzergruppen anzeigen möchten, denen er angehört. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Öffnen Sie im unteren Bereich des Registers "Benutzer" das Register "Benutzergruppen". Für den selektierten Benutzer werden die Benutzergruppen angezeigt, denen er angehört. Für Benutzergruppen, die bereits in die Benutzerverwaltung importiert wurden, ist das Optionskästchen in der Spalte "Importiert" aktiviert.

Synchronisationsstatus prüfen

Um den Synchronisationsstatus zu prüfen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Blenden Sie den Banner mit der Synchronisationsinformation ein und klicken Sie auf den Link "Status überprüfen".

Oder:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer" oder das Register "Benutzergruppen".
4. Klicken Sie in der Funktionsleiste auf "Status überprüfen".

Synchronisation durchführen

Um die Synchronisation durchzuführen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Blenden Sie den Banner mit der Synchronisationsinformation ein und klicken Sie auf den Link "Synchronisieren".

Oder:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer" oder das Register "Benutzergruppen".
4. Klicken Sie in der Funktionsleiste auf "Synchronisieren".

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Rollen verwalten

6

Die Zuordnung von Rechten zu Benutzern erfolgt über Rollen. Im TIA Portal sind die vom System definierten Rollen "ES Administrator" und "ES Standard" immer vorhanden. Diese Rollen können Sie nicht umbenennen oder entfernen. Auch die Zuordnung der Funktionsrechte zu diesen Rollen können Sie nicht verändern.

Sie können jedoch eigene Rollen definieren und ihnen bestimmte Funktionsrechte zuweisen. Zusätzlich können Sie für Ihre selbst definierten Rollen die folgenden Aktionen durchführen:

- Rollennamen ändern
- Maximale Sitzungsdauer ändern
- Kommentar ändern
- Zuordnung der Funktionsrechte ändern oder löschen
- Rolle löschen

Für jeden Projektbenutzer oder globalen Benutzer und für jede globale Benutzergruppe können Sie sich die zugewiesenen Funktionsrechte anzeigen lassen. So haben Sie jederzeit einen Überblick über die vergebenen Rechte.

Voraussetzung

- Ein geschütztes Projekt ist geöffnet.
- Sie sind mit einem Benutzerkonto angemeldet, das über die Funktionsrechte "Benutzer und Rollen verwalten" und "Projekt mit Schreibrechten öffnen" verfügt.

Neue Rolle definieren

Um eine neue Rolle zu definieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Klicken Sie auf "Neue Rolle hinzufügen".
5. Geben Sie einen Namen für die Rolle ein.
6. Ändern Sie ggf. die maximale Sitzungsdauer.
7. Geben Sie bei Bedarf einen Kommentar ein.
   Die neue Rolle wurde angelegt. Als Nächstes sollten Sie ihr die notwendigen Funktionsrechte zuweisen.

Hinweis

Sie können eine neue Rolle auch durch Kopieren einer vorhandenen Rolle definieren. Dadurch werden die zugewiesenen Funktionsrechte auch der kopierten Rolle zugewiesen.

Funktionsrechte zuweisen

Um einer Rolle Funktionsrechte zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Selektieren Sie eine Rolle. Beachten Sie, dass Sie beim Zuweisen keine Multiselektion verwenden können.
5. Klappen Sie im unteren Bereich "Kategorien von Funktionsrechten" die Kategorien auf, aus denen Sie Funktionsrechte zuweisen möchten.
6. Aktivieren Sie die Funktionsrechte, die Sie mit der Rolle verknüpfen möchten.

Rollennamen ändern

Um den Namen einer Rolle zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Klicken Sie in das Feld mit dem Rollennamen, den Sie ändern möchten.
5. Geben Sie den neuen Rollennamen ein.

Maximale Sitzungsdauer ändern

Um die maximale Sitzungsdauer für eine Rolle zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Geben Sie die neue Zeitdauer für die Rolle ein, für die Sie die Sitzungsdauer ändern möchten. Alternativ können Sie die neue Zeitdauer über die Pfeiltasten auswählen.

Kommentar ändern

Um den Kommentar zu einer Rolle zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".

3. Öffnen Sie das Register "Rollen".
4. Klicken Sie in das Feld mit dem Kommentar, den Sie ändern möchten.
5. Geben Sie den neuen Kommentar ein.

Zuordnung der Funktionsrechte ändern oder löschen

Um die Zuordnung von Funktionsrechten zu einer Rolle zu ändern oder zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Selektieren Sie die Rolle, für die Sie die Zuordnung der Funktionsrechte ändern möchten.
5. Klappen Sie im unteren Bereich "Kategorien von Funktionsrechten" die Kategorien auf, aus denen Sie Funktionsrechte zuweisen bzw. die Zuweisung aufheben möchten.
6. Aktivieren Sie die Funktionsrechte, die Sie der Rolle zuordnen möchten.
7. Deaktivieren Sie die Funktionsrechte, die der Rolle nicht mehr zugeordnet sein sollen.

Rolle löschen

Um eine Rolle zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Rollen".
4. Selektieren Sie die Rolle, die Sie löschen möchten. Beachten Sie, dass Sie die vom System definierten Rollen "ES Administrator" und "ES Standard" nicht löschen können.
5. Wählen Sie im Kontextmenü den Befehl "Löschen" oder drücken Sie die Taste <Entf>.

Zugewiesene Funktionsrechte eines Projektbenutzers oder eines globalen Benutzers anzeigen

Um sich die zugewiesenen Funktionsrechte eines Projektbenutzers oder eines globalen Benutzers anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Selektieren Sie den Projektbenutzer oder den globalen Benutzer, für den Sie die zugewiesenen Funktionsrechte anzeigen möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Öffnen Sie im unteren Bereich des Registers "Benutzer" das Register "Zugewiesene Rechte".

Zugewiesene Funktionsrechte einer globalen Benutzergruppe anzeigen

Um sich die zugewiesenen Funktionsrechte einer globalen Benutzergruppe anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".
4. Selektieren Sie die globale Benutzergruppe, für die Sie die zugewiesenen Funktionsrechte anzeigen möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Öffnen Sie im unteren Bereich des Registes "Benutzergruppen" das Register "Zugewiesene Rechte".

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Rollen zuweisen 7

Sie können den Projektbenutzern oder globalen Benutzern und Benutzergruppen Rollen zuweisen, die über verschiedene Funktionsrechte verfügen. Sie können die Zuweisungen jederzeit wieder aufheben.

Voraussetzung

- Ein geschütztes Projekt ist geöffnet.
- Sie sind mit einem Benutzerkonto angemeldet, das über die Funktionsrechte "Benutzer und Rollen verwalten" und "Projekt mit Schreibrechten öffnen" verfügt.

Projektbenutzern und globalen Benutzern Rollen zuweisen

Um einem Benutzer Rollen zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".
4. Selektieren Sie den Benutzer, dem Sie Rollen zuweisen möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Aktivieren Sie im unteren Bereich "Zugewiesene Rollen" die gewünschten Rollen.

Globalen Benutzergruppen Rollen zuweisen

Um einer globalen Benutzergruppe Rollen zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".
4. Selektieren Sie die Benutzergruppe, der Sie Rollen zuweisen möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Aktivieren Sie im unteren Bereich "Zugewiesene Rollen" die gewünschten Rollen.

Zuweisung von Rollen für Projektbenutzer und globale Benutzer aufheben

Um die Zuweisung einer Rolle zu einem Benutzer wieder aufzuheben, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzer".

4. Selektieren Sie den Benutzer, für den Sie die Zuweisung von Rollen aufheben möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Deaktivieren Sie im unteren Bereich "Zugewiesene Rollen" die Rollen, die dem Benutzer nicht mehr zugewiesen sein sollen.

Zuweisung von Rollen für globale Benutzergruppen aufheben

Um die Zuweisung einer Rolle zu einer globalen Benutzergruppe wieder aufzuheben, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Security-Einstellungen" in der Projektnavigation.
2. Doppelklicken Sie auf den Befehl "Benutzer und Rollen".
3. Öffnen Sie das Register "Benutzergruppen".
4. Selektieren Sie die Benutzergruppe, für die Sie die Zuweisung von Rollen aufheben möchten. Beachten Sie, dass Sie keine Multiselektion verwenden können.
5. Deaktivieren Sie im unteren Bereich "Zugewiesene Rollen" die Rollen, die der Benutzergruppe nicht mehr zugewiesen sein sollen.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

An einem geschützten Projekt anmelden 8

Voraussetzung

Sie verfügen über ein Benutzerkonto als Projektbenutzer oder als globaler Benutzer, mit dem mindestens eins der beiden folgenden Funktionsrechte verknüpft ist:

- Projekt schreibgeschützt öffnen
- Projekt mit Schreibrechten öffnen

Vorgehen

Um sich an einem geschützten Projekt anzumelden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".
   Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird angezeigt.
2. Wenn Sie beim Öffnen des ausgewählten Projekts überprüfen lassen möchten, ob das Projekt außerhalb des TIA Portals unzulässig verändert wurde, aktivieren Sie das Optionskästchen "Basis-Integritätsprüfung aktivieren". Die Basis-Integritätsprüfung kann einige Zeit in Anspruch nehmen.
   Siehe auch: Grundlagen zu Projekten
3. Selektieren Sie das geschützte Projekt aus der Liste, das Sie öffnen möchten. Falls das gewünschte Projekt nicht in der Liste vorhanden ist, führen Sie die folgenden Schritte aus:
   - Klicken Sie auf die Schaltfläche "Durchsuchen".
     Der Dialog "Bestehendes Projekt öffnen" wird geöffnet.
   - Navigieren Sie zum gewünschten Projektordner und selektieren Sie die Projektdatei.
4. Klicken Sie auf die Schaltfläche "Öffnen".
   Der Dialog "Anmelden" wird geöffnet.
5. Wählen Sie den Benutzertyp aus.
6. Geben Sie Ihren Benutzernamen ein.
7. Geben Sie Ihr Passwort ein.
8. Wenn Sie das Passwort für Ihr Benutzerkonto ändern möchten, klicken Sie auf die Schaltfläche "Passwort ändern".

   Hinweis

   Wenn Sie sich als globaler Benutzer anmelden, ist die Passwortänderung bei der ersten Anmeldung verpflichtend, falls dies in UMC so festgelegt wurde. Führen Sie dann die Schritte zur Passwortänderung durch.

   Siehe auch: Eigenes Passwort ändern (Seite 31)
9. Klicken Sie auf "OK".
   Das Projekt wird geöffnet, wenn Sie sich mit einem Benutzerkonto angemeldet haben, das über die entsprechenden Funktionsrechte verfügt.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Eigenes Passwort ändern

9

Sie können Ihr Passwort entweder beim Anmelden am Projekt ändern oder jederzeit beim Arbeiten in einem geschützten Projekt. Wenn Sie als Projektbenutzer Ihr Passwort ändern, wird nur das Passwort für dieses Projekt geändert. Wenn Sie als globaler Benutzer Ihr Passwort ändern, wird Ihr Passwort in UMC geändert. Dadurch wird Ihr Passwort für alle Projekte geändert, für die Sie als Benutzer eingetragen sind.

Voraussetzung

Sie sind mit einem Benutzerkonto angemeldet, das über das Funktionsrecht "Projekt mit Schreibrechten öffnen" verfügt.

Passwort beim Anmelden ändern

Um Ihr Passwort bei der Anmeldung am Projekt zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".
   Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird angezeigt.
2. Selektieren Sie ein geschütztes Projekt aus der Liste.
3. Klicken Sie auf die Schaltfläche "Öffnen".
   Der Dialog "Anmelden" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "Passwort ändern".
   Der Dialog "Passwort ändern" wird geöffnet.
5. Geben Sie Ihren Benutzernamen ein.
6. Geben Sie Ihr aktuelles Passwort ein.
7. Geben Sie Ihr neues Passwort ein.
8. Geben Sie Ihr neues Passwort zur Bestätigung erneut ein.
9. Klicken Sie auf die Schaltfläche "OK", um Ihr Passwort zu ändern.
   Wenn Sie alle Daten korrekt eingegeben haben, wird Ihr Passwort geändert. Verwenden Sie anschließend beim Anmelden an einem geschützten Projekt dieses neue Passwort.

Passwort beim Arbeiten in einem geschützten Projekt ändern

Um Ihr Passwort während des Arbeitens in einem geschützten Projekt zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektansicht.
2. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Benutzerverwaltung" auf den Pfeil-nach-unten.
   Eine Klappliste wird geöffnet, in der die Funktionen zur Benutzerverwaltung aufgelistet sind.

3. Klicken Sie auf "Passwort ändern".
   Der Dialog "Passwort ändern" wird geöffnet.
4. Geben Sie Ihren Benutzernamen ein.
5. Geben Sie Ihr aktuelles Passwort ein.
6. Geben Sie Ihr neues Passwort ein.
7. Geben Sie Ihr neues Passwort zur Bestätigung erneut ein.
8. Klicken Sie auf die Schaltfläche "OK", um Ihr Passwort zu ändern.
   Wenn Sie alle Daten korrekt eingegeben haben, wird Ihr Passwort geändert. Verwenden Sie anschließend beim Anmelden an einem geschützten Projekt dieses neue Passwort.

Siehe auch

An einem geschützten Projekt anmelden (Seite 29)

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

Vom Projekt abmelden

10

Sie können sich explizit von einem geschützten Projekt abmelden. Das Projekt wird dabei geschlossen.

Vorgehen

Um sich vom Projekt abzumelden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektansicht.
2. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Benutzerverwaltung" auf den Pfeil-nach-unten.
   Eine Klappliste wird geöffnet, in der die Funktionen zur Benutzerverwaltung aufgelistet sind.
3. Klicken Sie auf "Abmelden und Projekt schließen".
   Der Dialog "Abmelden und Projekt schließen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "Ja", um sich abzumelden und das Projekt zu schließen.
5. Klicken Sie auf die Schaltfläche "Nein", um am Projekt angemeldet zu bleiben.
6. Wenn Sie auf "Ja" geklickt haben und Sie seit dem letzten Speichern Änderungen am Projekt vorgenommen haben, wird eine Meldung eingeblendet. Geben Sie dann an, ob die Änderungen gespeichert werden sollen.

Siehe auch

Grundlagen zur Benutzerverwaltung im TIA Portal (Seite 5)

SIEMENS

Bibliotheken verwenden

Programmier- und Bedienhandbuch

Grundlagen zu Bibliotheken 1

Bibliotheken in verschiedenen TIA Produkten verwenden 2

Task Card "Bibliotheken" verwenden 3

Bibliotheksansicht verwenden 4

Bibliotheksverwaltung verwenden 5

Globale Bibliotheken verwenden 6

Kopiervorlagen einsetzen 7

Typen und ihre Versionen einsetzen 8

Multiuser Engineering in Bibliotheken 9

Typen einer Bibliothek aktualisieren und hochrüsten 10

Bibliothekstexte verwenden 11

Ordner in einer Bibliothek erstellen 12

Bibliothekselemente bearbeiten 13

Namen und Pfadstruktur harmonisieren 14

Bibliothekselemente vergleichen 15

Bibliothek aufräumen 16

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Grundlagen zu Bibliotheken....7
2 Bibliotheken in verschiedenen TIA Produkten verwenden....9
3 Task Card "Bibliotheken" verwenden....13
3.1 Übersicht über die Task Card "Bibliotheken"....13
3.2 Elementeansicht verwenden....16
4 Bibliotheksansicht verwenden....17
4.1 Übersicht über die Bibliotheksansicht....17
4.2 Bibliotheksansicht öffnen und schließen....20
5 Bibliotheksverwaltung verwenden....21
5.1 Übersicht über die Bibliotheksverwaltung....21
5.2 Bibliotheksverwaltung öffnen....25
5.3 Anzeige der Typen filtern....26
5.4 Instanzen im Projekt anzeigen....27
5.5 Querverweise einer Instanz anzeigen....28
5.6 Beziehungen zu anderen Bibliotheksobjekten anzeigen....29
6 Globale Bibliotheken verwenden....31
6.1 Globale Bibliothek erstellen....31
6.2 Schreibgeschützte globale Bibliothek erstellen....32
6.3 Kompatibilität globaler Bibliotheken....33
6.4 Globale Bibliothek öffnen....34
6.5 Bibliothekssprachen definieren....36
6.6 Eigenschaften von globalen Bibliotheken anzeigen....37
6.7 Protokolle zu globalen Bibliotheken anzeigen....38
6.8 Globale Bibliothek speichern....39
6.9 Globale Bibliothek schließen....41
6.10 Globale Bibliothek löschen....42
6.11 Globale Bibliotheken archivieren und dearchivieren....43
6.11.1 Arbeiten mit Archiven globaler Bibliotheken....43
6.11.2 Globale Bibliotheken archivieren....43
6.11.3 Globale Bibliotheken dearchivieren....44
6.12 Globale Unternehmensbibliotheken verwenden....46
6.12.1 Grundlagen zu Unternehmensbibliotheken....46
6.12.2 Konfigurationsdatei für Unternehmensbibliotheken erstellen....48

7 Kopiervorlagen einsetzen....51

7.1 Grundlagen zu Kopiervorlagen....51

7.2 Kopiervorlagen hinzufügen....52

7.3 Kopiervorlagen filtern....54

7.4 Kopiervorlagen verwenden....55

8 Typen und ihre Versionen einsetzen....57

8.1 Grundlagen zu Typen....57

8.2 Zustand von Typ-Versionen....60

8.3 Freigegebene Typ-Version anzeigen....62

8.4 Eigenschaften eines Typen oder einer Version anzeigen....63

8.5 Arbeiten mit Typen in der Projektbibliothek....64
8.5.1 Typen zur Projektbibliothek hinzufügen....64
8.5.2 Anforderungen an Bausteine....65
8.5.3 Referenzieren von Objekten innerhalb eines Bausteins....66
8.5.4 Typen duplizieren....68
8.5.5 Typen aus der Projektbibliothek verwenden....68
8.5.6 Typen zu einer Instanz anzeigen....70
8.5.7 Testversion eines Typen erstellen....71
8.5.8 Testversion eines Typs bearbeiten....73
8.5.9 Bearbeitungsversion eines Typen erstellen....73
8.5.10 Konsistenzprüfung für eine Version durchführen....74
8.5.11 Versionen verwerfen....75
8.5.12 Alle Versionen innerhalb eines Ordners verwerfen....75
8.5.13 Versionen freigeben....76
8.5.14 Alle Versionen innerhalb eines Ordners freigeben....78
8.5.15 Projekt auf die neuesten Versionen aktualisieren....79
8.5.16 Verbindung zwischen Instanz und Typ aufheben....80

8.6 Arbeiten mit Typen in globalen Bibliotheken....82
8.6.1 Typen zu einer globalen Bibliothek hinzufügen....82
8.6.2 Typen aus einer globalen Bibliothek verwenden....83
8.6.3 Projekt auf die neuesten Typ-Versionen aktualisieren....85

8.7 Version zuweisen....87

9 Multiuser Engineering in Bibliotheken....89

9.1 Multiuser Engineering mit Bibliothekselementen....89

9.2 Einchecken von Typen und Kopiervorlagen....91

10 Typen einer Bibliothek aktualisieren und hochrüsten....93

10.1 Bibliothek mit den Typen einer anderen Bibliothek aktualisieren....93

10.2 Typen in einem bestehenden Projekt hochrüsten....96

10.3 Beispiel für das Aktualisieren und Hochrüsten von Typen....99

11 Bibliothekstexte verwenden....105

11.1 Bibliothekstexte exportieren....105

11.2 Bibliothekstexte importieren....108
11.3 Projektsprachen der Projektbibliothek definieren....110
11.4 Aktualisierung von importierten Bibliothekstexten bei Typ-Instanzen....111
12 Ordner in einer Bibliothek erstellen....113
13 Bibliothekselemente bearbeiten....115
14 Namen und Pfadstruktur harmonisieren....119
15 Bibliothekselemente vergleichen....121
16 Bibliothek aufräumen....123
Index....125

Grundlagen zu Bibliotheken 1

Einführung

In den Bibliotheken legen Sie Bibliothekselemente ab, die Sie wiederverwenden möchten. Für jedes Projekt gibt es eine Projektbibliothek, die an das Projekt gebunden ist. Neben der Projektbibliothek können Sie beliebig viele globale Bibliotheken anlegen, die sich projektübergreifend verwenden lassen. Die Bibliotheken sind untereinander kompatibel, sodass Sie Bibliothekselemente zwischen den Bibliotheken kopieren und verschieben können. Mithilfe von Bibliotheken erstellen Sie z. B. Vorlagen für Bausteine, die Sie zunächst in die Projektbibliothek einfügen und dort weiterentwickeln. Schließlich kopieren Sie die Bausteine aus der Projektbibliothek in eine globale Bibliothek. Die globalen Bibliotheken stellen Sie anderen Mitarbeitern in Ihrem Projekt zur Verfügung. Die anderen Mitarbeiter verwenden die Bausteine weiter und passen sie gegebenenfalls an ihre persönlichen Erfordernisse an.

Sowohl die Projektbibliothek als auch globale Bibliotheken unterscheiden zwei unterschiedliche Arten von Bibliothekselementen:

- Typen
  Als Typen sind Objekte geeignet, die für den Ablauf von Anwenderprogrammen notwendig sind, z. B. Bausteine, PLC-Datentypen (UDT), HMI Anwenderdatentypen oder Bildbausteine. Typen sind versionierbar und unterstützen daher eine professionelle Weiterentwicklung. Projekte, in denen die Typen verwendet werden, lassen sich aktualisieren, sobald neue Versionen der Typen vorliegen.
- Kopiervorlagen
  Nahezu jedes Objekt lässt sich als Kopiervorlage speichern und später wieder in das Projekt einfügen. Sie können z. B. ganze Geräte mit ihren Inhalten oder Deckblätter für die Anlagendokumentation als Kopiervorlagen speichern.

Projektbibliothek

Jedes Projekt besitzt eine eigene Bibliothek, die Projektbibliothek. In der Projektbibliothek legen Sie Objekte ab, die Sie innerhalb des Projekts wiederverwenden möchten. Die Projektbibliothek wird zusammen mit dem aktuellen Projekt geöffnet, gespeichert und geschlossen.

Globale Bibliotheken

Zusätzlich zur Projektbibliothek verwenden Sie globale Bibliotheken, wenn Sie Bibliotheken projektübergreifend einsetzen möchten. Globale Bibliotheken existieren in drei Varianten:

- Systembibliotheken
  Zu eigenen Software-Produkten liefert Siemens globale Bibliotheken mit. Diese enthalten fertige Funktionen und Funktionsbausteine, die Sie innerhalb Ihres Projekts verwenden können. Die mitgelieferten Bibliotheken lassen sich nicht ändern. Die mitgelieferten Bibliotheken werden automatisch passend zum Projekt geladen.
- Unternehmensbibliotheken
  Unternehmensbibliotheken werden von Ihrer Organisation zentral zur Verfügung gestellt, z. B. in einem zentralen Ordner auf einem Netzlaufwerk. Das TIA Portal verwaltet die Unternehmensbibliotheken automatisch. Sobald eine neuere Version einer bereits vorhandenen Unternehmensbibliothek zur Verfügung steht, erhalten Sie die Aufforderung, die entsprechende Unternehmensbibliothek auf die neuere Version zu aktualisieren.
- Anwenderbibliotheken
  Globale Anwenderbibliotheken sind unabhängig von einem bestimmten Projekt und können daher an andere Anwender weitergegeben werden. Ein gemeinsamer Zugriff auf globale Anwenderbibliotheken, z. B. auf einem Netzlaufwerk, ist ebenfalls möglich, wenn alle Anwender die globale Anwenderbibliothek schreibgeschützt öffnen.
  Von Ihnen selbst erstellte globale Anwenderbibliotheken aus früheren Versionen des TIA Portals lassen sich weiterverwenden. Um globale Anwenderbibliotheken aus früheren Versionen des TIA Portals weiterzuverwenden, müssen Sie diese zunächst hochrüsten.

Vergleichen von Bibliothekselementen

Sie haben die Möglichkeit, Bausteine und PLC-Datentypen (UDT) mit den Objekten eines Geräts zu vergleichen. Sie können dadurch z. B. herausfinden, ob bestimmte Bausteine oder PLC-Datentypen (UDT) in einem Projekt verwendet wurden und ob diese verändert wurden.

Siehe auch

Übersicht über die Task Card "Bibliotheken" (Seite 13)

Übersicht über die Bibliotheksansicht (Seite 17)

Übersicht über die Bibliotheksverwaltung (Seite 21)

Grundlagen zu Kopiervorlagen (Seite 51)

Grundlagen zu Typen (Seite 57)

Bibliotheken in verschiedenen TIA Produkten verwenden

2

Beschreibung

Innerhalb einer TIA Portal Version (z. B. V15.1) können Sie verschiedene TIA Produkte verwenden (z. B. STEP 7 Professional, Safety, WinCC Advanced, etc.). Projekte mit Projektbibliotheken und globalen Bibliotheken, die Sie mit einem TIA Produkt erstellt haben, können Sie auch jederzeit mit einem anderen TIA Produkt öffnen. Allerdings können die Bibliothekselemente nur dann auch bearbeitet und im Projekt verwendet werden, wenn diese innerhalb des aktuellen Produkts auch unterstützt werden und alle notwendigen Lizenzen und Support Packages vorhanden sind. So können z. B. AWL-Programmbausteine nicht mit STEP 7 Basic bearbeitet werden.

Wenn eine Kopiervorlage oder ein Typ bzw. dessen Versionen nicht unterstützt werden, dann wird z. B. das Bausteinsymbol in der Bibliothek und in der Projektnavigation zusätzlich mit dem Symbol für nicht unterstützte Bibliothekselemente gekennzeichnet.

Wenn eine Kopiervorlage oder ein Typ als nicht unterstützt gekennzeichnet sind, dann stehen Ihnen die folgenden Funktionen zur Verfügung:

- umbenennen
- löschen
- innerhalb der Bibliothek verschieben
- in eine andere Bibliothek kopieren, solange diese Bibliothek das Bibliothekselement ebenfalls unterstützt

Wenn ein Typ als nicht unterstützt gekennzeichnet ist, dann stehen Ihnen darüber hinaus noch die folgenden Funktionen zur Verfügung:

- Die Funktion "Gehe zu Typ"
- Die Bibliotheksverwaltung
- Die Funktion "Bibliothek aufräumen"
- Bibliothek hochrüsten, solange keine nicht unterstützte Typversion enthalten ist
- Typversion zuweisen, solange keine nicht unterstützte Typversion enthalten ist

Beispiel

Das folgende Beispiel zeigt Ihnen wie es aussieht, wenn Sie eine Projektbibliothek aus einem umfangreicheren Produkt in einem weniger umfangreichen Produkt öffnen:

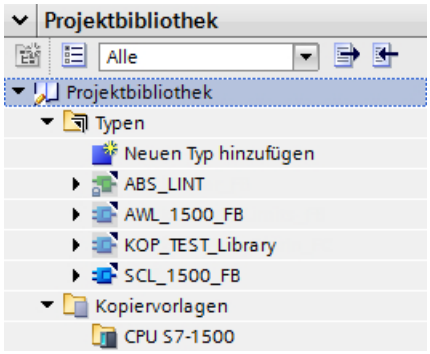

Die Projektbibliothek wurde mit STEP 7 Professional erstellt und beinhaltet Professional spezifische Typen.

Beim Öffnen des STEP 7 Professional-Projekts mit STEP 7 Basic erhalten Sie einen Hinweis, dass die enthaltenen Bibliotheken nicht ohne Einschränkungen bearbeitet werden können.

- Darstellung in der Projektbibliothek:

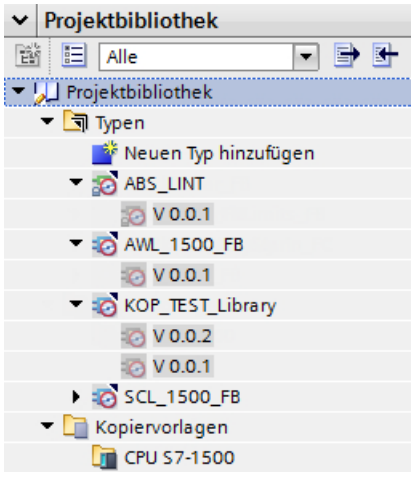

- Darstellung in der Projektnavigation:

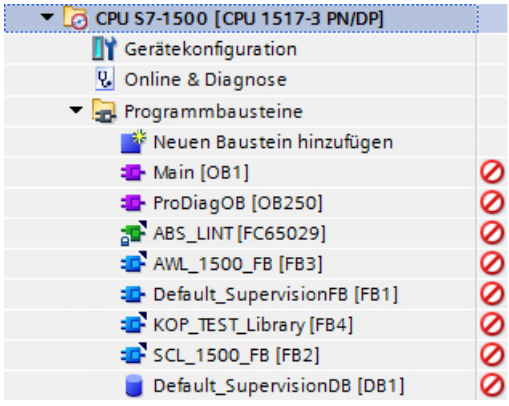

Alle STEP 7 Professional spezifischen Typen sind entsprechend als nicht unterstützt gekennzeichnet und können nur eingeschränkt bearbeitet werden.

Task Card "Bibliotheken" verwenden

3

3.1 Übersicht über die Task Card "Bibliotheken"

Funktion der Task Card "Bibliotheken"

Die Task Card "Bibliotheken" ermöglicht Ihnen ein effizientes Arbeiten mit der Projektbibliothek und den globalen Bibliotheken.

Aufbau der Task Card "Bibliotheken"

Die Task Card "Bibliotheken" besteht aus folgenden Komponenten:

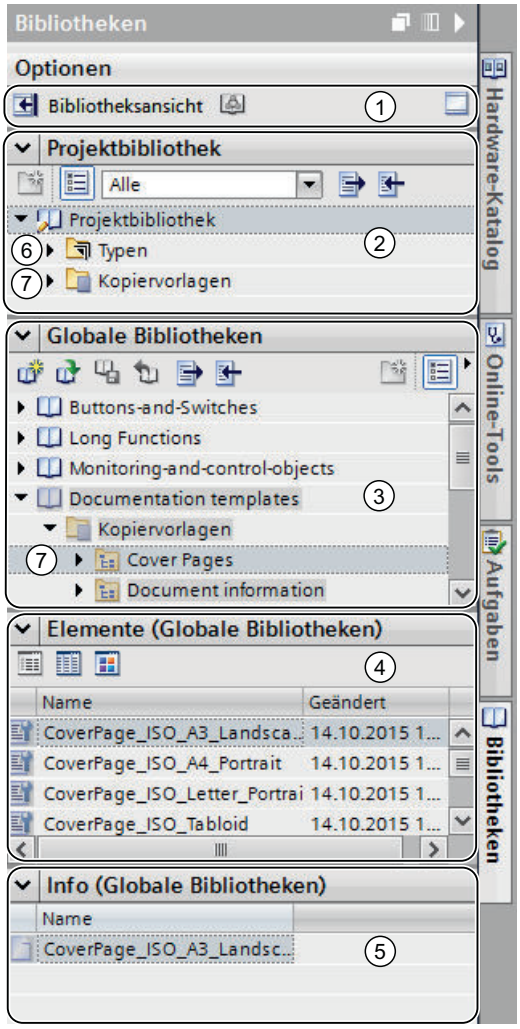

① Schaltflächen "Bibliotheksansicht" und "Bibliotheksverwaltung"
② Palette "Projektbibliothek"
③ Palette "Globale Bibliotheken"
④ Palette "Elemente"
⑤ Palette "Info"
⑥ Ordner "Typen"
⑦ Ordner "Kopiervorlagen"

Schaltfläche "Bibliotheksansicht"

Mit der Schaltfläche "Bibliotheksansicht" wechseln Sie zur Bibliotheksansicht. Die Task Card "Bibliotheken" und die Projektnavigation werden dabei ausgeblendet.

Siehe auch: Bibliotheksansicht verwenden (Seite 17)

Palette "Projektbibliothek"

In der Palette "Projektbibliothek" legen Sie Bibliothekselemente ab, die Sie innerhalb des Projekts mehrfach verwenden möchten.

Palette "Globale Bibliotheken"

In der Palette "Globale Bibliotheken" verwalten Sie globale Bibliotheken, deren Bibliothekselemente Sie projektübergreifend wiederverwenden können.

In der Palette "Globale Bibliotheken" werden auch Bibliotheken aufgeführt, die mit Ihren gekauften Produkten mitgeliefert werden. Diese Bibliotheken stellen Ihnen z. B. fertige Funktionen und Funktionsbausteine zur Verfügung. Die mitgelieferten globalen Bibliotheken sind nicht änderbar.

Palette "Elemente"

In dieser Palette lassen Sie sich die Inhalte von Ordnern in der Bibliothek anzeigen. Die Palette "Elemente" wird standardmäßig nicht angezeigt. Wenn Sie die Palette "Elemente" anzeigen möchten, muss diese zunächst aktiviert werden. In der Palette "Elemente" gibt es drei Darstellungsformen:

- Detailmodus
  Im Detailmodus werden die Eigenschaften von Ordnern, Kopiervorlagen und Typen in Tabellenform angezeigt.
- Listenmodus
  Im Listenmodus werden die Inhalte von Ordnern aufgelistet.
- Übersichtsmodus
  Im Übersichtsmodus werden die Inhalte von Ordnern mit großen Symbolen angezeigt.

Siehe auch: Elementeansicht verwenden (Seite 16)

Palette "Info"

In der Palette "Info" lassen Sie sich die Inhalte der Bibliothekselemente anzeigen. Außerdem werden die einzelnen Versionen von Typen sowie das letzte Änderungsdatum der Version angezeigt.

Ordner "Typen"

In den Verzeichnissen "Typen" verwalten Sie Typen und Typ-Versionen von Bibliothekselementen, die Sie als Instanzen im Projekt verwenden.

Siehe auch: Verwendung von Typen (Seite 57)

Ordner "Kopiervorlagen"

In den Verzeichnissen "Kopiervorlagen" verwalten Sie Kopiervorlagen von Bibliothekselementen, die Sie als Kopien im Projekt einfügen können.

Siehe auch: Verwendung von Kopiervorlagen (Seite 51)

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Bibliothekselemente vergleichen (Seite 121)

3.2 Elementeansicht verwenden

Einführung

Beim ersten Öffnen der Task Card "Bibliotheken" werden die Paletten "Projektbibliothek" und "Globale Bibliotheken" geöffnet und die Palette "Info" geschlossen angezeigt. Die Palette "Elemente" blenden Sie bei Bedarf ein.

Die Elementeansicht zeigt die Elemente der selektierten Bibliothek. In der Elementeansicht sind drei Darstellungsformen möglich:

- Details
  Im Detailmodus werden die Eigenschaften von Ordnern, Kopiervorlagen und Typen in Tabellenform angezeigt.
- Liste
  Im Listenmodus werden die Inhalte von Ordnern aufgelistet.
- Übersicht
  Im Übersichtsmodus werden die Inhalte von Ordnern mit großen Symbolen angezeigt.

Die Palette "Info" zeigt den Inhalt des selektierten Bibliothekselements. Wenn Sie beispielsweise einen Typen in der Elementeansicht selektieren, dann werden in der Palette "Info" die Typ-Versionen angezeigt.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet.

Vorgehen

Um die Elementeansicht zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Palette "Projektbibliothek" oder in der Palette "Globale Bibliotheken" auf "Elementeansicht öffnen oder schließen".
2. Um die Darstellungsform von der Detailansicht auf die Listenform oder Übersicht zu ändern, klicken Sie in der Funktionsleiste auf das entsprechende Symbol.

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Übersicht über die Task Card "Bibliotheken" (Seite 13)

Bibliothekselemente vergleichen (Seite 121)

Globale Bibliotheken verwenden (Seite 31)

Bibliotheksansicht verwenden 4

4.1 Übersicht über die Bibliotheksansicht

Funktion der Bibliotheksansicht

Die Bibliotheksansicht vereinigt die Funktionalitäten der Task Card "Bibliotheken" und dem Übersichtsfenster. In der Bibliotheksansicht zeigen Sie die Elemente einer Bibliothek in unterschiedlichen Ansichten an und sehen z. B. in der Detailansicht weitere Eigenschaften der einzelnen Elemente. Außerdem bearbeiten und versionieren Sie Typen in der Bibliotheksansicht.

Aufbau der Bibliotheksansicht

Das folgende Bild zeigt die Komponenten der Bibliotheksansicht:

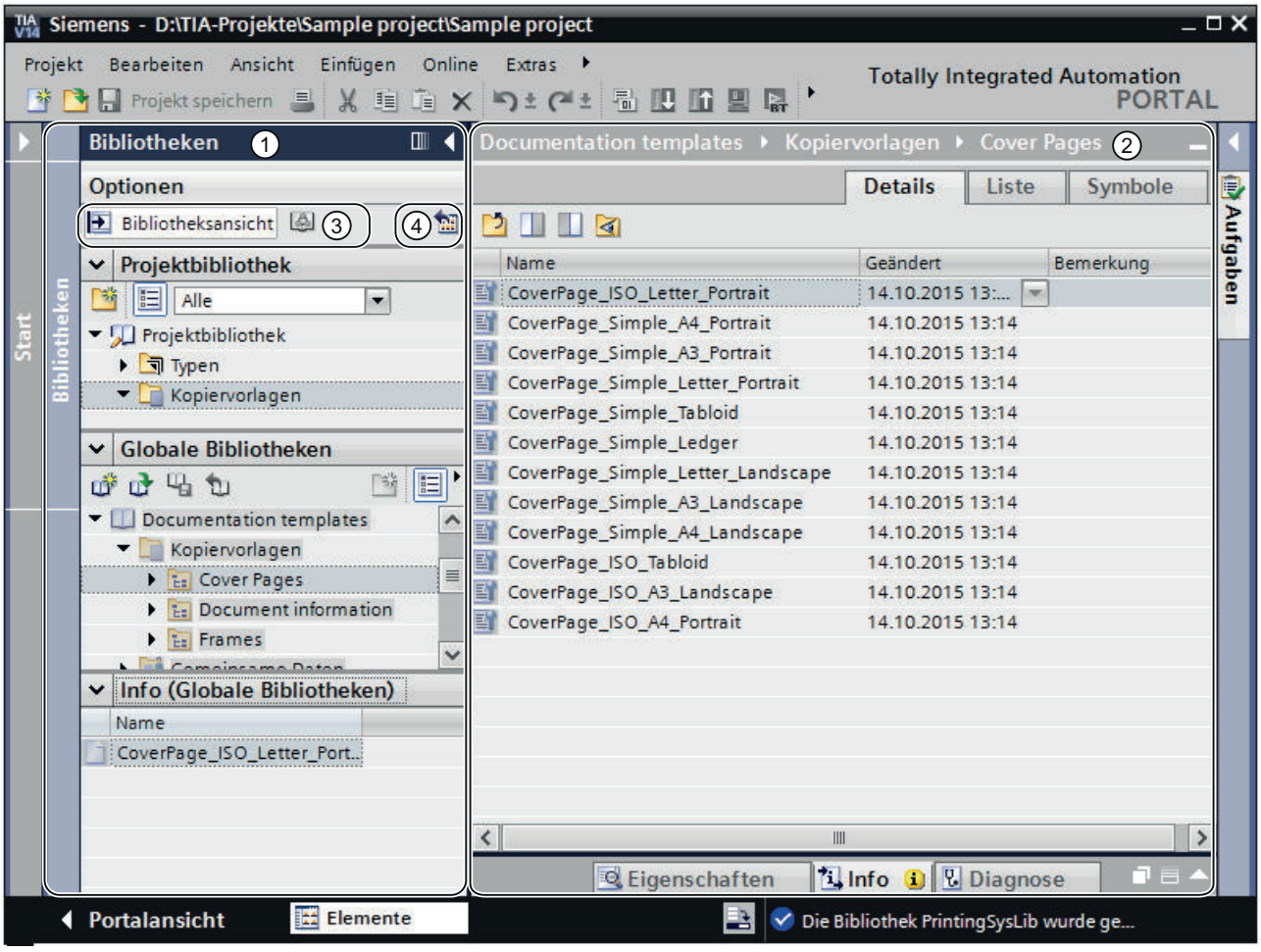

① Bibliotheksnavigation

② Bibliotheksübersicht

③ Schaltfläche "Bibliotheksansicht öffnen/schließen" und "Bibliotheksverwaltung öffnen"

④ Schaltfläche "Bibliotheksübersicht öffnen oder schließen"

Bibliotheksnavigation

Die Bibliotheksnavigation gleicht bis auf kleinere Abweichungen der Task Card "Bibliotheken". Im Gegensatz zur Task Card gibt es keine Palette "Elemente", da die Elemente in der Bibliotheksübersicht angezeigt werden. Zudem können Sie in der Bibliotheksnavigation die Bibliotheksansicht schließen oder die Bibliotheksübersicht öffnen oder schließen.

Siehe auch: Task Card "Bibliotheken" (Seite 13)

Bibliotheksübersicht

Die Bibliotheksübersicht entspricht dem Übersichtsfenster und zeigt die Elemente des aktuell in der Bibliotheksnavigation selektierten Objekts an. Sie können die Elemente in drei unterschiedlichen Ansichten anzeigen lassen:

- Detailansicht
  Die Objekte werden in einer Liste mit zusätzlichen Informationen, wie z. B. dem Änderungsdatum, angezeigt.
- Listenansicht
  Die Objekte werden in einer einfachen Liste angezeigt.
- Symbolansicht
  Die Objekte werden symbolisch nach Kategorien geordnet angezeigt.

Außerdem können Sie in der Bibliotheksübersicht folgende Aktionen durchführen:

- Elemente umbenennen
- Elemente löschen
- Elemente kopieren
- Elemente verschieben
- Instanzen von Typen bearbeiten
- Typen versionieren
- Nur WinCC: Bildbausteine und HMI Anwenderdatentypen bearbeiten

Siehe auch: Übersichtsfenster

Siehe auch

Grundlagen zu Kopiervorlagen (Seite 51)

Grundlagen zu Typen (Seite 57)

Bibliotheksansicht öffnen und schließen (Seite 20)

Grundlagen zu Bibliotheken (Seite 7)

Bibliothekselemente vergleichen (Seite 121)

4.2 Bibliotheksansicht öffnen und schließen

Die Bibliotheksansicht wird in einigen Fällen automatisch geöffnet, z. B. wenn Sie die Testinstanz eines Typs oder Bildbausteine und HMI Anwenderdatentypen bearbeiten. Sie können die Bibliotheksansicht jedoch auch manuell öffnen.

Bibliotheksansicht öffnen

Um die Bibliotheksansicht manuell zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Bibliotheken".
2. Klicken Sie in der Task Card "Bibliotheken" auf die Schaltfläche "Bibliotheksansicht öffnen/schließen".
   Die Bibliotheksnavigation wird geöffnet. Die Task Card "Bibliotheken" und die Projektnavigation werden geschlossen.
3. Wenn die Bibliotheksübersicht nicht angezeigt wird, klicken Sie in der Bibliotheksnavigation auf die Schaltfläche "Maximiert/minimiert die Elemente".
   Die Bibliotheksübersicht wird geöffnet.

Bibliotheksansicht verlassen

Um die Bibliotheksansicht zu verlassen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Bibliotheksnavigation auf die Schaltfläche "Bibliotheksansicht öffnen/schließen".
   Die Bibliotheksnavigation wird geschlossen. Die Task Card "Bibliotheken" und die Projektnavigation werden geöffnet.

Siehe auch

Übersicht über die Bibliotheksansicht (Seite 17)

Grundlagen zu Bibliotheken (Seite 7)

Bibliothekselemente vergleichen (Seite 121)

Task Card "Bibliotheken" verwenden (Seite 13)

Globale Bibliotheken verwenden (Seite 31)

Bibliotheksverwaltung verwenden 5

5.1 Übersicht über die Bibliotheksverwaltung

Funktion der Bibliotheksverwaltung

Kopiervorlagen und Typen mit Abhängigkeiten zu anderen Bibliothekselementen unterliegen einigen Funktionseinschränkungen. Sie lassen sich beispielsweise nicht löschen, solange noch Abhängigkeiten bestehen. Auf diese Weise wird verhindert, dass andere Bibliothekselemente unbrauchbar werden. Mit der Bibliotheksverwaltung identifizieren Sie Abhängigkeiten und verschaffen sich einen Überblick über den Arbeitsfortschritt.

Die Bibliotheksverwaltung bietet folgende Funktionen:

- Darstellung der Zusammenhänge von Typen und Kopiervorlagen
  Wird ein Typ in anderen Typen oder Kopiervorlagen referenziert, dann werden die Zusammenhänge in der Bibliotheksverwaltung angezeigt. Ebenso wird angezeigt, welche Bibliothekselemente ein Typ oder eine Kopiervorlage referenziert.
- Anzeige der Verwendungsstellen von Typen im Projekt
- Hochrüsten von Typen
- Eingrenzen der angezeigten Typen mithilfe von Filtern

Aufbau der Bibliotheksverwaltung

Das folgende Bild zeigt die Komponenten der Bibliotheksverwaltung:

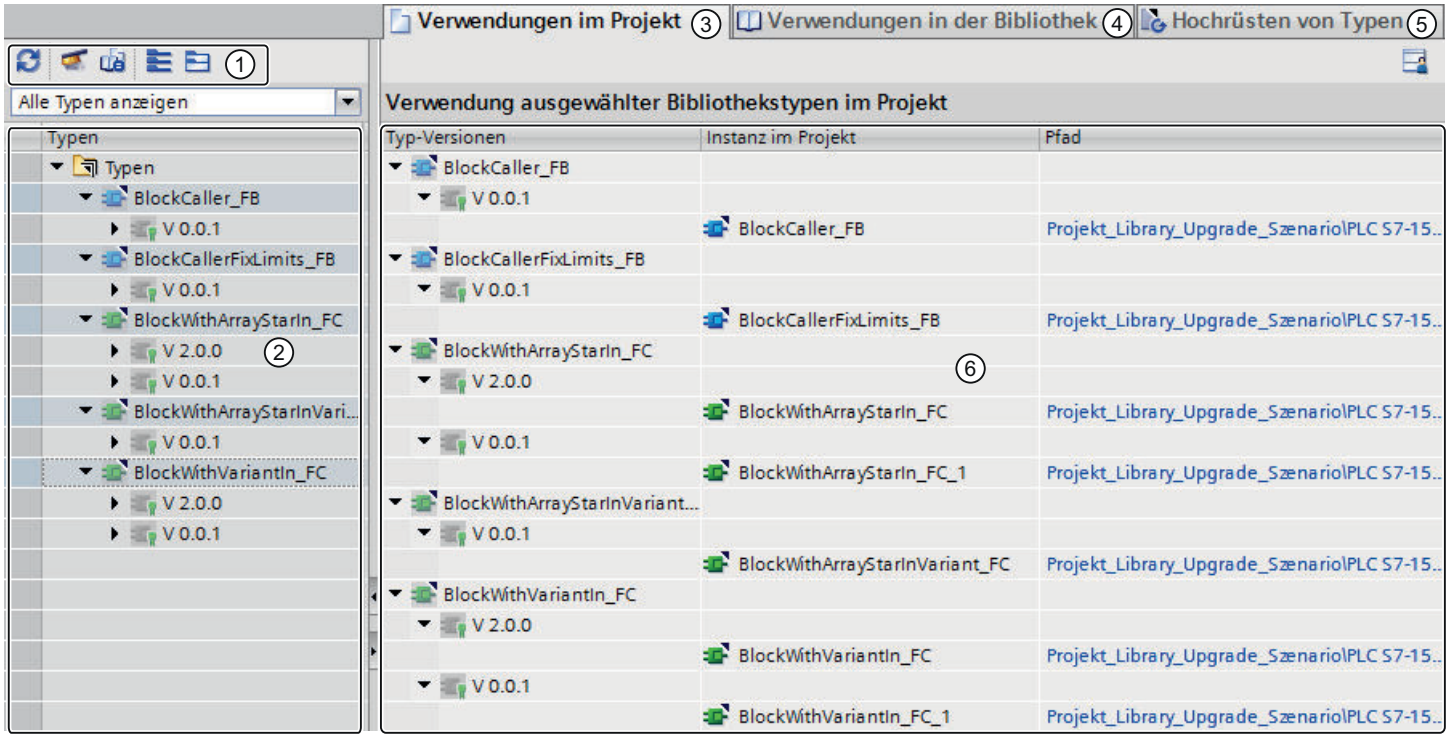

① Funktionsleiste der Bibliotheksverwaltung
② Bereich "Typen"
③ Register "Verwendungen im Projekt"
④ Register "Verwendungen in der Bibliothek"
⑤ Register "Hochrüsten von Typen"
⑥ Fenster "Verwendungen"

Funktionsleiste der Bibliotheksverwaltung

Über die Funktionsleiste der Bibliotheksverwaltung können Sie folgende Aktionen durchführen:

- Verwendungen aktualisieren
  Falls das Projekt geändert wurde, aktualisieren Sie die Ansicht der Bibliotheksverwaltung.
- Bibliothek aufräumen
  Durch das Aufräumen einer Bibliothek werden alle Typen und Typ-Versionen gelöscht, die mit keiner Instanz im Projekt verknüpft sind.
- Projekt harmonisieren
  Durch das Harmonisieren eines Projekts werden die Namen und die Pfadstrukuren von Typverwendungen im Projekt an die entsprechenden Namen und Pfadstrukturen der Typen innerhalb einer Bibliothek angepasst.
- Alle ausblenden
  Im Bereich "Typen" werden alle Untereinträge des obersten Knotens ausgeblendet. Die unterlagerten Elemente, wie z. B. Typen und einzelne Typ-Versionen, werden nicht mehr angezeigt.
- Alle einblenden
  Im Bereich "Typen" werden alle unterlagerten Elemente im Bereich "Typen" ausgeklappt. Die unterlagerten Elemente, wie z. B. Typen und Typ-Versionen, werden vollständig angezeigt.

Bereich "Typen"

Im Bereich "Typen" werden die Inhalte des Ordners angezeigt, den Sie in der Bibliotheksansicht selektiert haben. Zu jedem Typ werden die Typen angezeigt, die von ihm referenziert werden. Über die Schaltflächen in der Funktionsleiste des Bereichs "Typen" können Sie alle Typ-Knoten auf- oder zuklappen.

Außerdem können Sie mit der Klappliste "Filter" die Ansicht filtern. Weitere Informationen zu den einzelnen Filtereinstellungen finden Sie hier: Anzeige der Typen filtern (Seite 26)

Register "Verwendungen im Projekt"

Im Register "Verwendungen im Projekt" werden die Instanzen von Typ-Versionen, sowie ihre jeweilige Verwendungsstelle im Projekt, angezeigt. Wenn Sie eine Instanz selektieren, können Sie im Inspektorfenster die Querverweise der Instanz im Projekt anzeigen.

Weitere Informationen finden Sie hier:

Querverweise einer Instanz anzeigen (Seite 28)

Instanzen im Projekt anzeigen (Seite 27)

Register "Verwendungen in der Bibliothek"

Im Register "Verwendungen in der Bibliothek" werden alle Stellen innerhalb der Bibliothek angezeigt, an denen ein Typ oder eine Kopiervorlage verwendet wird.

Weitere Informationen finden Sie hier: Beziehungen zu anderen Bibliotheksobjekten anzeigen (Seite 29)

Register "Hochrüsten von Typen"

Im Register "Hochrüsten von Typen" können Sie alle Typen identifizieren, die geänderte Typen referenzieren, und hochrüsten.

Weitere Informationen zum Hochrüsten von Typen finden Sie hier:

Typen in einem bestehenden Projekt hochrüsten (Seite 96)

Beispiel für das Aktualisieren und Hochrüsten von Typen (Seite 99)

Siehe auch

Bibliotheksverwaltung öffnen (Seite 25)

Grundlagen zu Bibliotheken (Seite 7)

5.2 Bibliotheksverwaltung öffnen

Vorgehen

Um die Bibliotheksverwaltung zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksansicht.
2. Selektieren Sie einen Typen oder einen beliebigen Ordner, der Typen enthält.
3. Wählen Sie im Kontextmenü den Befehl "Bibliotheksverwaltung".

Oder:

1. Klicken Sie in der Projektbibliothek den Ordner "Typen" an.
2. Wählen Sie aus dem Kontextmenü den Eintrag "Bibliotheksverwaltung".

Ergebnis

Die Bibliotheksverwaltung wird geöffnet und die Typen werden mit ihren Versionen angezeigt.

Siehe auch

Übersicht über die Bibliotheksverwaltung (Seite 21)

5.3 Anzeige der Typen filtern

Um in umfangreichen Bibliotheken einen besseren Überblick über die Typen zu bekommen, können Sie die Filter nutzen. Mit einem Filter haben Sie die Möglichkeit, die angezeigten Typen einzugrenzen. Folgende Filter stehen zur Verfügung:

- Typen mit ausstehenden Änderungen
  Der Filter zeigt alle Typen, die eine Version im Zustand "in Test" oder "in Bearbeitung" besitzen.
- Freigegebene Typen
  Der Filter zeigt alle freigegebenen Typen an, die keine Versionen im Zustand "in Test" oder "in Bearbeitung" besitzen.
- Typen mit mehreren Versionen
  Der Filter zeigt alle Typen an, die mehr als eine Version besitzen.
- Typen ohne Verwendung im Projekt
  Der Filter zeigt alle Typen an, die keine Instanz im Projekt besitzen.
- Typen mit neuen Versionen für das Hochrüsten anderer Typen
  Der Filter zeigt alle Typen an, die für das Hochrüsten anderer Typen relevant sind. Die Typen müssen außerdem folgende Kriterien erfüllen:
  - Die Typen besitzen mehr als eine Version.
  - Alle Versionen des jeweiligen Typs sind freigegeben.

Voraussetzung

Die Bibliotheksverwaltung ist geöffnet.

Vorgehen

Um die angezeigten Typen zu filtern, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bereich "Typen" den Ordner, auf dessen Inhalt Sie den Filter anwenden möchten.
2. Wählen Sie aus der Klappliste den gewünschten Filter aus.
   Im Bereich "Typen" werden die Typen angezeigt, die den gewählten Filterkriterien entsprechen.

Siehe auch

Übersicht über die Bibliotheksverwaltung (Seite 21)

Zustand von Typ-Versionen (Seite 60)

Testversion eines Typen erstellen (Seite 71)

Testversion eines Typs bearbeiten (Seite 73)

Bearbeitungsversion eines Typen erstellen (Seite 73)

5.4 Instanzen im Projekt anzeigen

In der Bibliotheksverwaltung können Sie sich die Instanzen aller Versionen eines Typs oder einer einzelnen Typ-Version anzeigen lassen. Sie können direkt zu jeder Instanz im Projekt springen.

Voraussetzung

Die Bibliotheksverwaltung ist geöffnet.

Vorgehen

Um die Instanzen eines Typs oder seiner Versionen anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bereich "Typen" die gewünschten Ordner, Typen oder Versionen.
2. Öffnen Sie das Register "Verwendungen im Projekt".
   Zu jeder Typ-Version werden die Instanzen im Projekt angezeigt. In der Spalte "Pfad" wird der Pfad angezeigt, an dem die jeweilige Instanz im Projekt zu finden ist.
3. Um direkt zu der jeweiligen Instanz in der Projektnavigation zu springen, klicken Sie auf den Pfad.
   Die Bibliotheksverwaltung wird ausgeblendet und die Instanz in der Projektnavigation selektiert.

Siehe auch

Querverweise einer Instanz anzeigen (Seite 28)

Typen aus der Projektbibliothek verwenden (Seite 68)

Übersicht über die Bibliotheksverwaltung (Seite 21)

Beziehungen zu anderen Bibliotheksobjekten anzeigen (Seite 29)

5.5 Querverweise einer Instanz anzeigen

Sie können die Querverweise einer Instanz anzeigen, ohne die Bibliotheksverwaltung zu verlassen.

Voraussetzung

Die Bibliotheksverwaltung ist geöffnet.

Vorgehen

Um die Querverweise einer Instanz im Projekt anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bereich "Typen" die Typ-Version, deren Instanzen Sie anzeigen möchten.
2. Selektieren Sie im Register "Verwendungen im Projekt" die Instanz der gewünschten Typ-Version.
3. Öffnen Sie im Inspektorfenster das Register "Info > Querverweise".
   Die Querverweise der Instanz im Projekt werden angezeigt.

Siehe auch

Übersicht über die Bibliotheksverwaltung (Seite 21)

Instanzen im Projekt anzeigen (Seite 27)

5.6 Beziehungen zu anderen Bibliotheksobjekten anzeigen

In der Bibliotheksverwaltung lassen Sie sich die Beziehungen zwischen einzelnen Bibliotheksobjekten anzeigen. Die Referenzen einzelner Typ-Versionen zu anderen Bibliotheksobjekten werden automatisch im Bereich "Typen" angezeigt. Zusätzlich können Sie sich im Register "Verwendungen in der Bibliothek" anzeigen lassen, in welchen anderen Bibliotheksobjekten die jeweilige Typ-Version referenziert wird.

Voraussetzung

Die Bibliotheksverwaltung ist geöffnet.

Vorgehen

Um anzuzeigen, von welchen anderen Bibliotheksobjekten eine Typ-Version referenziert wird, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bereich "Typen" die gewünschten Ordner, Typen oder Versionen.
2. Öffnen Sie das Register "Verwendungen in der Bibliothek".
   Im Fenster "Verwendungen" wird angezeigt, welche anderen Bibliotheksobjekte die einzelnen Typ-Versionen referenzieren.
3. Um zu dem referenzierenden Bibliotheksobjekt zu springen, klicken Sie in der Spalte "Pfad" auf den jeweiligen Pfad.

Siehe auch

Instanzen im Projekt anzeigen (Seite 27)

Übersicht über die Bibliotheksverwaltung (Seite 21)

Referenzieren von Objekten innerhalb eines Bausteins (Seite 66)

Globale Bibliotheken verwenden 6

6.1 Globale Bibliothek erstellen

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um eine neue globale Bibliothek zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Palette "Globale Bibliotheken" auf das Symbol "Neue globale Bibliothek erstellen" oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Neue Bibliothek erstellen".
   Der Dialog "Neue globale Bibliothek erstellen" wird geöffnet.
2. Geben Sie den Namen und den Speicherort für die neue globale Bibliothek an.
3. Bestätigen Sie Ihre Eingaben mit "Erstellen".

Ergebnis

Die neue globale Bibliothek wird erzeugt und in die Palette "Globale Bibliotheken" eingefügt. Am Speicherort der globalen Bibliothek wird im Dateisystem ein Ordner mit dem Namen der globalen Bibliothek angelegt. Die eigentliche Bibliotheksdatei erhält die Dateinamenerweiterung ".al15_1".

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek öffnen (Seite 34)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 37)

Globale Bibliothek speichern (Seite 39)

Globale Bibliothek schließen (Seite 41)

Globale Bibliothek löschen (Seite 42)

6.2 Schreibgeschützte globale Bibliothek erstellen

Sie können globale Bibliotheken schützen, indem Sie diese als geschützte Bibliotheken exportieren. Diese Bibliotheken sind dann mit einem Schreibschutz versehen, sowohl die darin enthaltenen Typen als auch die globale Bibliothek sind somit vor Änderungen geschützt. Der Schreibschutz einmal geschützter globaler Bibliotheken kann nicht aufgehoben werden.

Voraussetzungen

- Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.
- Eine nicht schreibgeschützte globale Bibliothek liegt vor.

Vorgehen

Um eine schreibgeschützte globale Bibliothek zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die globale Bibliothek, aus der Sie eine schreibgeschützte globale Bibliothek erstellen wollen.
2. Rufen Sie mit der rechten Maustaste das Kontextmenü auf.
3. Wählen Sie im Kontextmenü der globalen Bibliothek den Befehl "Als geschützte Bibliothek exportieren" aus.
   Der Dialog "Als geschützte Bibliothek exportieren" wird geöffnet.
4. Geben Sie den Namen und den Speicherort für die schreibgeschützte globale Bibliothek an.
5. Bestätigen Sie Ihre Eingaben mit "Export".

Ergebnis

Die schreibgeschützte globale Bibliothek wird erzeugt und in die Palette "Globale Bibliotheken" eingefügt. Am Speicherort der globalen Bibliothek im Dateisystem wird ein Ordner mit dem von Ihnen vergebenen Namen der geschützten globalen Bibliothek angelegt. Die eigentliche Bibliotheksdatei erhält die Dateinamenerweiterung ".al15_1".

Die Typen und die Kopiervorlagen aus der ursprünglichen globalen Bibliothek werden als schreibgeschützte Typen und Kopiervorlagen in die neue globale Bibliothek kopiert.

Hinweis

Veränderung schreibgeschützter Typen verhindern

Schreibgeschützte Typen lassen sich unter Umständen verändern, wenn sie durch aktuellere nicht-schreibgeschützte Versionen des gleichen Typs überschrieben werden.

Wenn Sie schreibgeschützte Typen an Kunden ausliefern wollen, erstellen Sie daher möglichst schreibgeschützte Bibliotheken mit Typen in einer neuen höheren Version und liefern diese dann aus.

Somit ist ausgeschlossen, dass der Kunde bereits nicht-schreibgeschützte Versionen dieses Typs nutzt und Veränderungen vornehmen kann.

6.3 Kompatibilität globaler Bibliotheken

Bibliotheken aus dem TIA Portal V13 oder früher

Globale Bibliotheken, die mit dem TIA Portal V13 oder früher erstellt wurden, lassen sich in der aktuellen Version des TIA Portals nicht verwenden. Rüsten Sie die Bibliotheken zunächst mit der Version V13 SP1 des TIA Portals auf die aktuelle Bibliotheksversion hoch. Anschließend können Sie die Bibliothek öffnen und erneut hochrüsten.

Bibliotheken aus dem TIA Portal V13 SP1 oder später

Mit dem TIA Portal lassen sich globale Bibliotheken öffnen, die mit dem TIA Portal V13 SP1 oder später erstellt wurden. Beim Öffnen der Bibliothek werden Sie aufgefordert, die Bibliothek auf die aktuelle Version des TIA Portals hochzurüsten. Nach dem Hochrüsten der globalen Bibliothek lässt sich die globale Bibliothek nicht mehr mit früheren Versionen des TIA Portals öffnen.

Rückwärtskompatibilität der aktuellen Bibliotheksversion

Globale Bibliotheken, die im Bibliotheksformat der aktuellen Produktversion des TIA Portals gespeichert sind, sind aufgrund der erweiterten Funktionalität gegenüber früheren Versionen nicht rückwärts kompatibel. Globale Bibliotheken im aktuellen Bibliotheksformat können ausschließlich in Verbindung mit TIA Portal V15.1-Projekten verwendet werden.

Siehe auch

Globale Bibliothek öffnen (Seite 34)

6.4 Globale Bibliothek öffnen

Globale Bibliotheken lassen sich zentral weiter entwickeln und projektübergreifend verwenden. Mehrere Personen können eine globale Bibliothek gleichzeitig von einem zentralen Ablageort aus öffnen, sofern alle Anwender die globale Bibliothek schreibgeschützt öffnen.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um eine globale Bibliothek zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Palette "Globale Bibliotheken" auf das Symbol "Globale Bibliothek öffnen" oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek öffnen".
   Der Dialog "Globale Bibliothek öffnen" wird geöffnet.
2. Wählen Sie die globale Bibliothek, die Sie öffnen möchten. Sie erkennen die Bibliotheksdatei an der Dateinamenserweiterung ".al[Versionsnummer]". Globale Bibliotheken, die mit der aktuellen Produktversion des TIA Portals gespeichert wurden, besitzen demnach die Dateinamenserweiterung ".al15_1".
3. Für die Bibliothek ist der Schreibschutz aktiviert. Wenn Sie die globale Bibliothek ändern möchten, deaktivieren Sie die Option "Schreibgeschützt öffnen".
4. Klicken Sie auf "Öffnen".
   Wenn die Bibliothek in der Bibliotheksversion für das TIA Portal V15.1 gespeichert ist, dann wird die globale Bibliothek geöffnet und in die Palette "Globale Bibliotheken" eingefügt. Wenn Sie eine globale Bibliothek aus dem TIA Portal V13 SP1 oder später gewählt haben, wird zunächst der Dialog "Globale Bibliothek hochrüsten" geöffnet. Rüsten Sie die Bibliothek in diesem Fall auf die aktuelle Version des TIA Portals hoch.

Ergebnis

Die globale Bibliothek wird in der Palette "Globale Bibliotheken" angezeigt.

Wenn die globale Bibliothek schreibgeschützt geöffnet wurde, dann ist sie und alle ihre Unterordner grau hinterlegt.

Siehe auch

Globale Bibliotheken dearchivieren (Seite 44)

Kompatibilität globaler Bibliotheken (Seite 33)

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek erstellen (Seite 31)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 37)

Globale Bibliothek speichern (Seite 39)

Globale Bibliothek schließen (Seite 41)
Globale Bibliothek löschen (Seite 42)

6.5 Bibliothekssprachen definieren

Für globale Bibliotheken können Sie verschiedene Bibliothekssprachen definieren und daraus eine Editiersprache für alle Kommentare auswählen. Die Auswahl der Bibliothekssprachen ist für alle Bibliothekselemente (Typen, Typ-Versionen und Kopiervorlagen) innerhalb der globalen Bibliothek gültig.

Die Kommentare der Typen und deren Typ-Versionen können in mehrere Bibliothekssprachen übersetzt und in der globalen Bibliothek verwaltet werden. Die ausgewählte Editiersprache gibt vor, in welcher Sprache die Kommentare angezeigt werden. Der Kommentar eines Typs kann jederzeit bearbeitet werden. Sobald die Version eines Typs freigegeben ist, kann deren Kommentar nicht mehr bearbeitet werden und wird im Lesemodus angezeigt.

Wenn Sie die Editiersprache einer globalen Bibliothek ändern, dann werden die Kommentare aller Typen in der neu ausgewählten Editiersprache angezeigt, sofern die Übersetzungen vorhanden sind.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um die Bibliothekssprachen einer globalen Bibliothek zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Palette "Globale Bibliotheken" auf das Symbol "Globale Bibliothek öffnen" oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek öffnen".
   Der Dialog "Globale Bibliothek öffnen" wird geöffnet.
2. Wählen Sie die globale Bibliothek, die Sie öffnen möchten.
3. Deaktivieren Sie den Schreibschutz der globalen Bibliothek, indem Sie den Haken bei "Schreibgeschützt öffnen" entfernen.
   Wenn Sie die Option "Schreibgeschützt öffnen" nicht deaktivieren, dann können Sie in der globalen Bibliothek die Auswahl der Bibliothekssprachen sehen, aber nicht ändern.
4. Klicken Sie auf "Öffnen".
   Die globale Bibliothek wird angezeigt.
5. Öffnen Sie den Ordner "Sprachen & Ressourcen".
6. Klicken Sie den Eintrag "Bibliothekssprachen" doppelt an oder wählen Sie aus dem Kontextmenü "Öffnen".

Ergebnis

Die Bibliothekssprachen werden in der Bibliotheksansicht angezeigt.

6.6 Eigenschaften von globalen Bibliotheken anzeigen

Globale Bibliotheken besitzen Eigenschaften zur näheren Beschreibung der jeweiligen Bibliothek. Zu den Eigenschaften gehören die folgenden Punkte:

- Allgemeine Informationen zur Bibliothek
  Hier erhalten Sie beispielsweise Informationen zum Erstellzeitpunkt, zum Autor, den Speicherpfad, der Dateigröße, Hinweise zum Copyright, usw. Viele der Attribute sind änderbar.
- Bibliothekshistorie
  Die Bibliothekshistorie enthält eine Übersicht über die durchgeführten Migrationen. Hier können Sie auch die Protokolldatei zu den Migrationen aufrufen. Außerdem enthält die Bibliothekshistorie Informationen zu Updates der globalen Bibliothek.
- Support Packages in der Bibliothek
  Sie können eine Übersicht über zusätzliche Software anzeigen. Die zusätzliche Software ist notwendig, um alle Geräte des Projekts bearbeiten zu können.
- Softwareprodukte in der Bibliothek
  Sie können eine Übersicht aller installierten und für das Projekt notwendigen Softwareprodukte anzeigen.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um die Eigenschaften einer globalen Bibliothek anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, deren Eigenschaften Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Dialog mit den Eigenschaften der globalen Bibliothek wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die gewünschten Eigenschaften aus, die angezeigt werden sollen.

Siehe auch

Globale Bibliothek öffnen (Seite 34)

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek erstellen (Seite 31)

Globale Bibliothek speichern (Seite 39)

Globale Bibliothek schließen (Seite 41)

Globale Bibliothek löschen (Seite 42)

6.7 Protokolle zu globalen Bibliotheken anzeigen

Wenn Sie globale Bibliotheken aktualisieren, aufräumen oder mehreren Typen eine gemeinsame Version zuweisen, dann werden Protokolle angelegt. In den Protokollen sind alle durchgeführten Änderungen an der globalen Bibliothek aufgelistet. Die Protokolle werden in Zip-Dateien angelegt und zusammen mit der globalen Bibliothek gespeichert. Die Protokolle sind immer verfügbar, nachdem Sie die globale Bibliothek geöffnet haben.

Vorgehen

Um die Protokolle einer globalen Bibliothek aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
2. Öffnen Sie den unterlagerten Ordner "Gemeinsame Daten > Protokolle".
3. Doppelklicken Sie auf das gewünschte Protokoll.
   Das Protokoll wird im Arbeitsbereich geöffnet.

Siehe auch

Bibliothek mit den Typen einer anderen Bibliothek aktualisieren (Seite 93)

6.8 Globale Bibliothek speichern

Nachdem Sie eine globale Bibliothek geändert haben, speichern Sie diese ab. Über "Bibliothek speichern unter" können Sie eine globale Bibliothek unter einem anderen Namen speichern.

Hinweis

Rückwärtskompatibilität zu früheren Versionen des TIA Portals

Beachten Sie, dass sich globale Bibliotheken nicht mehr in früheren Versionen des TIA Portals öffnen lassen, nachdem Sie einmal in der aktuellen Version gespeichert wurden.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Änderungen speichern

Um eine globale Bibliothek zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie speichern möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek speichern".

Globale Bibliothek unter anderem Namen speichern

Um eine globale Bibliothek unter einem anderen Namen zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie unter einem anderen Namen speichern möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek speichern unter".
   Der Dialog "Globale Bibliothek speichern unter" wird geöffnet.
3. Wählen Sie den Speicherort und geben Sie den Dateinamen ein.
4. Bestätigen Sie Ihre Eingaben mit "Speichern".
   Die Bibliothek wird an dem angegebenen Ort unter dem neuen Namen gespeichert. Die ursprüngliche Bibliothek bleibt bestehen.

Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 43)

Globale Bibliotheken archivieren (Seite 43)

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek erstellen (Seite 31)

Globale Bibliothek öffnen (Seite 34)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 37)

Globale Bibliothek schließen (Seite 41)
Globale Bibliothek löschen (Seite 42)

6.9 Globale Bibliothek schließen

Globale Bibliotheken sind von Projekten unabhängig. Das bedeutet auch, dass globale Bibliotheken nicht zusammen mit dem Projekt geschlossen werden. Sie müssen globale Bibliotheken daher explizit schließen.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um eine geöffnete globale Bibliothek zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie schließen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek schließen".
   Wenn Sie Änderungen an der globalen Bibliothek vorgenommen haben, wählen Sie, ob Sie die Änderungen speichern möchten oder nicht.
   Die globale Bibliothek wird geschlossen.

Siehe auch

Globale Bibliothek erstellen (Seite 31)

Globale Bibliothek öffnen (Seite 34)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 37)

Globale Bibliothek speichern (Seite 39)

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek löschen (Seite 42)

6.10 Globale Bibliothek löschen

Wenn Sie eine globale Bibliothek nicht mehr benötigen, können Sie diese löschen. Von Siemens mitgelieferte Bibliotheken können nicht gelöscht werden.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um eine globale Bibliothek zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".

Ergebnis

Die globale Bibliothek wird aus der Palette "Globale Bibliotheken" entfernt. Das gesamte Verzeichnis für die globale Bibliothek wird aus dem Dateisystem gelöscht.

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Globale Bibliothek erstellen (Seite 31)

Globale Bibliothek öffnen (Seite 34)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 37)

Globale Bibliothek speichern (Seite 39)

Globale Bibliothek schließen (Seite 41)

6.11 Globale Bibliotheken archivieren und dearchivieren

6.11.1 Arbeiten mit Archiven globaler Bibliotheken

Wenn Sie globale Bibliotheken beispielsweise auf einer externen Festplatte sichern oder per E-Mail versenden möchten, reduzieren Sie den Speicherplatzbedarf der Bibliothek mit der Archivieren-Funktion.

Möglichkeiten zur Reduktion der Projektgröße

Um den Speicherplatzbedarf von globalen Bibliotheken zu reduzieren, gibt es zwei Möglichkeiten:

- Erstellen eines komprimierten Archivs von globalen Bibliotheken
  Archive globaler Bibliotheken sind komprimierte Dateien, die jeweils eine ganze globale Bibliothek inklusive der gesamten Ordnerstruktur der Bibliothek enthalten. Bevor das Verzeichnis mit der globalen Bibliothek zur Archivdatei gepackt wird, werden sämtliche Dateien auf ihre wesentlichen Bestandteile reduziert, um den Speicherplatzbedarf zusätzlich zu verringern. Komprimierte Archive globaler Bibliotheken eignen sich daher gut zum Versand per E-Mail.
  Komprimierte Archive globaler Bibliotheken der aktuellen Produktversion besitzen die Dateiendung ".zal15_1".
  Um ein komprimiertes Archiv einer globalen Bibliothek zu öffnen, dearchivieren Sie das Archiv. Die Archivdatei wird an einem von Ihnen gewählten Ort mit der gesamten Ordnerstruktur und sämtlichen Dateien entpackt.
- Minimieren globaler Bibliotheken
  Sie können auf die zusätzliche Komprimierung in einer Archivierdatei verzichten und stattdessen eine Kopie des Verzeichnisses der globalen Bibliothek erstellen. Die enthaltenen Dateien werden auf die wesentlichen Bestandteile reduziert. Somit wird der Speicherplatzbedarf minimiert. Der volle Funktionsumfang der globalen Bibliothek bleibt erhalten und die globale Bibliothek lässt sich gewohnt laden.
  Minimierte globale Bibliotheken sind besonders gut zur Archivierung geeignet, z. B. auf einem externen Datenträger.

Siehe auch

Globale Bibliotheken archivieren (Seite 43)

Globale Bibliotheken dearchivieren (Seite 44)

6.11.2 Globale Bibliotheken archivieren

Globale Bibliotheken lassen sich als komprimierte oder unkomprimierte Dateien archivieren. Die zu archivierenden globalen Bibliotheken müssen dafür nicht im TIA Portal geöffnet sein.

Der Speicherplatzbedarf von globalen Bibliotheken lässt sich reduzieren, indem Sie die globalen Bibliotheken in komprimierten Dateien archivieren.

Vorgehen

Um eine globale Bibliothek zu archivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Archivieren...".
   Alternativ wählen Sie im Kontextmenü der globalen Bibliothek, die Sie archivieren möchten, den Befehl "Bibliothek archivieren".
   Der Dialog "Archivieren" wird geöffnet.
2. Wählen Sie im Feld "Quellpfad" eine Bibliotheksdatei mit der Endung ".al15_1" aus.
3. Um eine komprimierte Archivdatei zu erzeugen, wählen Sie die Option "Als komprimierte Datei archivieren" aus.
4. Wenn Sie wiederherstellbare Daten nicht archivieren wollen, wählen Sie die Option "Wiederherstellbare Daten verwerfen" aus.
5. Um Datum und Zeit automatisch hinzuzufügen, wählen Sie die Option "Datum und Zeit zum Zielnamen hinzufügen" aus.
6. Wählen Sie im Feld "Zielpfad" den Speicherort, an dem Sie die Archivdatei oder das neue Verzeichnis der globalen Bibliothek speichern möchten.
   Das Verzeichnis darf nicht in einem Projektverzeichnis oder innerhalb des Verzeichnisses einer globalen Bibliothek liegen.
   Das Standardverzeichnis können Sie unter "Extras > Einstellungen > Allgemein > Archivierungseinstellungen > Speicherort für Bibliotheksarchive" einstellen.
7. Klicken Sie auf "Archivieren".

Ergebnis

Ein Archiv der globalen Bibliothek wird erzeugt. Komprimierte globale Bibliotheken erhalten die Dateiendung ".zal15_1". Die Datei enthält das komplette Verzeichnis der globalen Bibliothek. Die einzelnen Dateien der globalen Bibliothek wurden zusätzlich auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

Bei unkomprimierten globalen Bibliotheken wird lediglich eine Kopie des ursprünglichen Verzeichnisses der globalen Bibliothek am gewünschten Speicherort erstellt.

Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 43)

Globale Bibliotheken dearchivieren (Seite 44)

6.11.3 Globale Bibliotheken dearchivieren

Bevor Sie eine archivierte globale Bibliothek verwenden können, dearchivieren Sie diese. Dabei wird die globale Bibliothek entpackt und anschließend im TIA Portal geöffnet.

Vorgehen

Um das Archiv einer globalen Bibliothek zu entpacken, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek dearchivieren". Der Dialog "Archivierte globale Bibliothek dearchivieren" wird geöffnet.
2. Wählen Sie die Archivdatei aus.
3. Aktivieren Sie das Optionskästchen "Schreibgeschützt öffnen", wenn Sie die globale Bibliothek schreibgeschützt laden möchten.
4. Klicken Sie auf Öffnen.
5. Der Dialog "Ordner suchen" wird geöffnet.
6. Wählen Sie das Zielverzeichnis aus, an dem die archivierte globale Bibliothek entpackt wird.
7. Klicken Sie auf "OK".

Ergebnis

Die globale Bibliothek wird im gewählten Verzeichnis entpackt und sofort geöffnet.

Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 43)

Globale Bibliotheken archivieren (Seite 43)

Globale Bibliothek öffnen (Seite 34)

Kompatibilität globaler Bibliotheken (Seite 33)

6.12 Globale Unternehmensbibliotheken verwenden

6.12.1 Grundlagen zu Unternehmensbibliotheken

Einführung

Unternehmensbibliotheken sind globale Bibliotheken, die von einem Administrator zur Verfügung gestellt und dem TIA Portal zugewiesen werden. Der Administrator kann jederzeit neue Bibliotheken zuweisen oder Bibliotheken ändern. Neue Bibliotheken werden nach Ihrer Bestätigung automatisch im TIA Portal geladen. Sobald neuere Versionen von Unternehmensbibliotheken zur Verfügung stehen, werden die vorhandenen Unternehmensbibliotheken ebenfalls nach Ihrer Bestätigung auf die neuere Version aktualisiert.

Die Unternehmensbibliotheken befinden sich wie gewöhnliche globale Bibliotheken in der Palette "Globale Bibliotheken" der Task Card "Bibliotheken".

Bereitstellung von Unternehmensbibliotheken

Unternehmensbibliotheken können in einem beliebigen Verzeichnis auf der Festplatte des Computers oder auf einem Netzlaufwerk gespeichert sein. Um Unternehmensbibliotheken in das TIA Portal einzubinden, verwenden Sie eine XML-Datei. In der XML-Datei sind die Verzeichnisse und die Namen der zugewiesenen Unternehmensbibliotheken gespeichert. Die XML-Datei muss in folgendem Verzeichnis auf der Festplatte des Computers gespeichert sein:

`C:\ProgramData\Siemens\Automation\Portal V15_1\CorporateSettings\`

Die XML-Datei muss als "`CorporateSettings.xml`" benannt sein.

Die Konfigurationsdatei kopieren Sie entweder selbst in das entsprechende Verzeichnis oder die Konfigurationsdatei wird Ihnen über das Unternehmensnetz zugewiesen. Die gültige Konfiguration wird beim Start des TIA Portals automatisch angewendet. Während das TIA Portal gestartet ist, wird das Verzeichnis für Konfigurationsdateien kontinuierlich überwacht. Wenn sich die Konfigurationsdatei geändert hat, erhalten Sie eine Aufforderung, die neue Konfiguration anzuwenden. Die Aufforderung können Sie bis zu zweimal ablehnen. Die nächste Aufforderung erhalten Sie jeweils drei Stunden später. Bei der dritten Aufforderung müssen Sie die neue Konfiguration anwenden. Eine geänderte Konfigurationsdatei erhalten Sie von Ihrem Projektadministrator, zum Beispiel, wenn Unternehmensbibliotheken hinzukommen oder wegfallen.

Wenn Sie auf Unternehmensbibliotheken zugreifen, die auf einem Netzlaufwerk gespeichert sind, benötigen Sie nur Netzzugriff zu deren Aktualisierung. Die Unternehmensbibliotheken, die lokal gespeichert sind, werden nach einem Neustart automatisch geladen. Wenn Sie keinen Netzzugriff haben, können Sie nur lokal gespeicherte Kopien Ihrer Unternehmensbibliotheken nutzen.

Möglichkeiten als Projektadministrator

Die Konfigurationsdatei oder die Unternehmensbibliotheken können Sie den Computern der Teammitglieder automatisiert zuweisen oder Aktualisierungen an die Teammitglieder verbreiten. Diese Funktion ist nicht Bestandteil des TIA Portals und setzt eine entsprechende IT-Infrastruktur in Ihrem Unternehmen voraus. Wenn Sie die Konfigurationsdatei zentral verwalten möchten, klären Sie das Vorgehen mit den IT-Verantwortlichen Ihres Unternehmens.

Um Unternehmensbibliotheken an Mitarbeiter des Projekt-Teams weiterzugeben, eignet sich die Archiv-Funktion für globale Bibliotheken. Sie müssen die Unternehmensbibliotheken vor der Verwendung im TIA Portal jedoch wieder dearchivieren.

Das folgende Bild zeigt beispielhaft, wie Sie Unternehmensbibliotheken innerhalb Ihres Unternehmens an einzelne Mitarbeiter verteilen können:

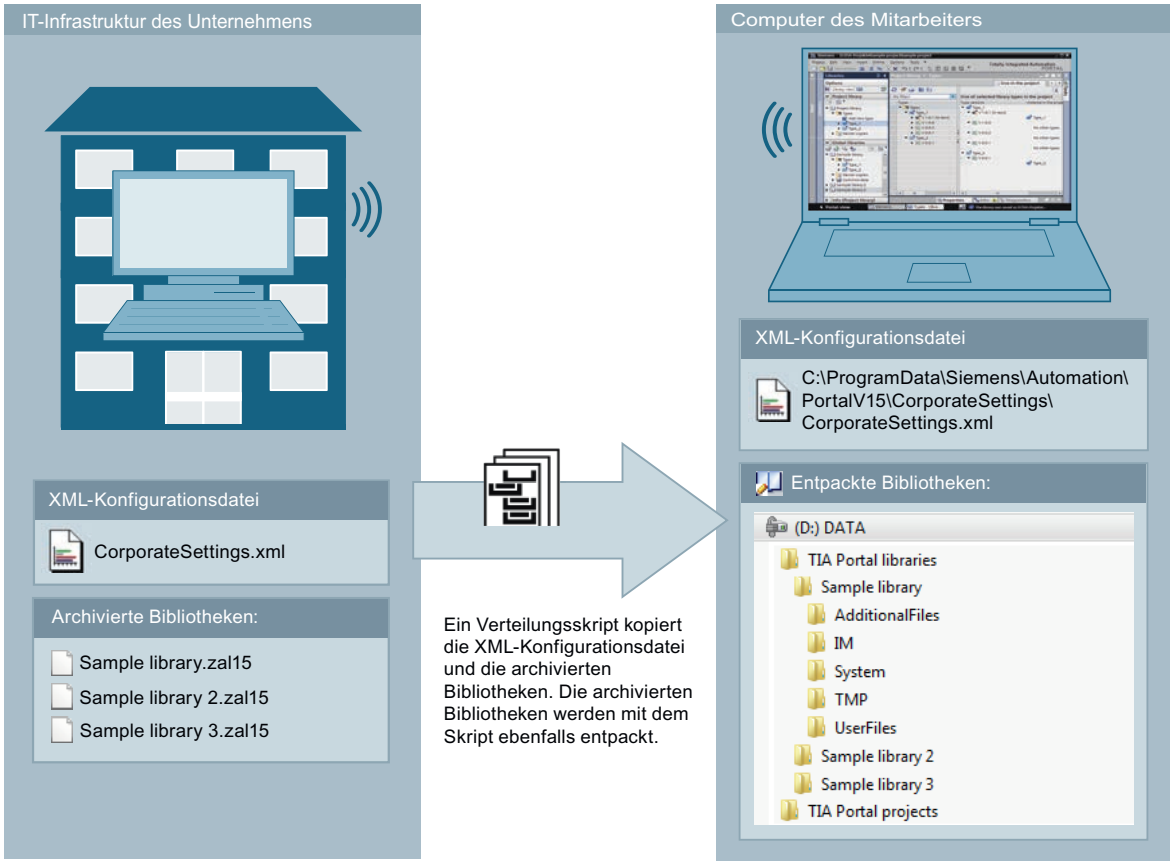

Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 43)

Globale Bibliotheken archivieren (Seite 43)

Globale Bibliotheken dearchivieren (Seite 44)

6.12.2 Konfigurationsdatei für Unternehmensbibliotheken erstellen

Um Unternehmensbibliotheken im TIA Portal verfügbar zu machen, verwenden Sie eine Konfigurationsdatei im XML-Format. Die Konfigurationsdatei enthält die Verzeichnisse und Dateinamen der zu ladenden Bibliotheken. Die XML-Datei ist dieselbe Datei, die Sie für die Einstellungen zur anwenderdefinierten Dokumentation verwenden.

Im Folgenden erfahren Sie, wie Sie die XML-Konfigurationsdatei erstellen und wo Sie diese speichern.

Vorgehen

Um eine Konfigurationsdatei für Unternehmensbibliotheken zur Verfügung zu stellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine XML-Konfigurationsdatei mit unten genanntem Inhalt. Verwenden Sie die Kodierung "UTF-8".
2. Ersetzen Sie die im unten stehenden Beispiel genannten Pfade im Abschnitt "<CorporateLibraryPaths>" mit den Pfaden, an denen Ihre Bibliotheken gespeichert sind.
3. Speichern Sie die XML-Datei unter dem Namen "CorporateSettings.xml".
4. Speichern Sie die Datei in folgendem Verzeichnis auf dem Computer: C:\ProgramData\Siemens\Automation\Portal V15_1\CorporateSettings\

Inhalt der XML-Konfigurationsdatei

Die XML-Konfigurationsdatei muss folgenden Inhalt besitzen:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<Document>
 <Settings.Settings ID="0">
  <ObjectList>
   <Settings.General ID="1" AggregationName="General">
    <!-- An dieser Stelle finden Sie gegebenenfalls die Einstellungen für globale
Unternehmensbibliotheken. -->
    <ObjectList>

<!-- Sektion für Unternehmensbibliotheken -->
     <Settings.General ID="1" AggregationName="General">
     <AttributeList>
      <CorporateLibraryPaths>
      <!-- Beispiel für zwei Einträge -->
      <Item>F:\CorporateLibraries\Corporate_Library_1\Corporate_Library_1.al15_1</
Item>          <Item>F:\CorporateLibraries\Corporate_Library_2\CorporateLibraries.al15_1</
Item>
      <!-- Tragen Sie hier gegebenenfalls alle weiteren globalen Bibliotheken ein. --
>           </CorporateLibraryPaths>
    </AttributeList>
   </Settings.General>
<!-- Sektion für Unternehmensbibliotheken Ende -->

<!-- Sektion für anwenderdefinierte Dokumentation -->
     <Settings.UserDocumentation ID="2" AggregationeName="UserDocumentation">
     <!-- Im folgenden Abschnitt legen Sie die Werte für die Anzeige der anwenderdefinierte
Dokumentation fest. -->
      <AttributeList>
       <!-- Aktiviert oder deaktiviert die Anzeige des Zugriffsprotokolls. -->
       <DisplayLogInformation>
        <Value>true</Value>
        </DisplayLogInformation>
       <!-- Aktiviert oder deaktiviert die Suche nach anwenderdefinierter Dokumentation in
einem zentralen Verzeichnis. -->
       <EnableLookupFromCentralStorageLocation>
         <Value>true</Value>
       </EnableLookupFromCentralStorageLocation>
       <!-- Legt das zentrale Verzeichnis für anwenderdefinierte Dokumentation fest. -->
       <CentralStorageLocation>
        <Value>D:\CorporateDocumentation\UserDocumentation\</Value>
       </CentralStorageLocation>
      </AttributeList>
     </Settings.UserDocumentation>
<!-- Sektion für anwenderdefinierte Dokumentation Ende -->

    </ObjectList>
   </Settings.General>
  </ObjectList>
 </Settings.Settings>
</Document>
```

Ergebnis

Sobald Sie die XML-Konfigurationsdatei im entsprechenden Verzeichnis gespeichert haben, erhalten Sie im TIA Portal die Aufforderung, die Unternehmensbibliotheken zu laden.

Kopiervorlagen einsetzen 7

7.1 Grundlagen zu Kopiervorlagen

Sie verwenden Kopiervorlagen, um häufig benötigte Elemente standardisiert zu erzeugen. Auf Basis einer Kopiervorlage können Sie bei Bedarf beliebig viele Elemente erstellen und in das Projekt einfügen. Die Elemente erben die Eigenschaften der Kopiervorlage.

Kopiervorlagen legen Sie entweder in der Projektbibliothek oder in einer globalen Bibliothek ab. Kopiervorlagen in der Projektbibliothek können Sie nur innerhalb des Projekts verwenden. Wenn Sie die Kopiervorlage in einer globalen Bibliothek erstellen, lässt sie sich in unterschiedlichen Projekten verwenden.

In der Bibliothek lassen sich z. B. die folgenden Elemente als Kopiervorlagen anlegen:

- Geräte mit ihrer Gerätekonfiguration
- Variablentabellen oder einzelne Variablen
- Anweisungsprofile
- Beobachtungstabellen
- Elemente aus den Dokumentationseinstellungen, z. B. Deckblätter und Rahmen
- Bausteine und Gruppen, die mehrere Bausteine enthalten
- PLC-Datentypen (UDTs) und Gruppen, die mehrere PLC-Datentypen enthalten
- Textlisten
- Meldeklassen
- Technologieobjekte

In vielen Fällen enthalten die Objekte, die Sie als Kopiervorlage hinzufügen, weitere Elemente. Eine CPU kann beispielsweise Bausteine enthalten. Wenn die enthaltenen Elemente Verwendungen einer Typ-Version sind, dann werden die verwendeten Versionen der Typen automatisch in der Bibliothek angelegt. Die enthaltenen Elemente sind anschließend als Instanz verwendet und mit dem Typ verbunden.

Siehe auch

Kopiervorlagen hinzufügen (Seite 52)

Kopiervorlagen verwenden (Seite 55)

Grundlagen zu Typen (Seite 57)

Kopiervorlagen filtern (Seite 54)

7.2 Kopiervorlagen hinzufügen

Objekte, die Sie mehrfach verwenden möchten, speichern Sie als Kopiervorlage in die Projektbibliothek oder in eine globale Bibliothek. Sie können beim Erstellen von Kopiervorlagen zwischen den folgenden Möglichkeiten wählen:

- Sie selektieren ein oder mehrere Elemente und erzeugen daraus einzelne Kopiervorlagen
- Sie selektieren mehrere Elemente und erzeugen daraus eine einzelne Kopiervorlage, die alle selektierten Elemente enthält.

Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Wenn Sie ein Gerät als Kopiervorlage hinzufügen, dann erfüllt das Gerät die folgenden Anforderungen:
  - Das Gerät ist übersetzt und in konsistentem Zustand.
  - Das Gerät enthält keine Testinstanz eines Typs.
- Wenn Sie die Kopiervorlage zu einer globalen Bibliothek hinzufügen, ist die globale Bibliothek mit Schreibrechten geöffnet.

Aus einem oder mehreren Elementen Kopiervorlagen erzeugen

Um aus einem oder mehreren Elementen eine Kopiervorlage zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliothek in der Task Card "Bibliotheken".
2. Selektieren Sie die gewünschten Elemente.
3. Ziehen Sie die Elemente per Drag&Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".

Alternative:

1. Selektieren Sie die gewünschten Elemente.
2. Kopieren Sie die Elemente in die Zwischenablage und fügen Sie die Elemente an der gewünschten Stelle ein.
   Die Elemente werden jeweils als eine Kopiervorlage in die Bibliothek eingefügt. Von eventuell enthaltenen Objekten (z. B. referenzierten Bausteinen) wird automatisch jeweils ein Typ erzeugt.

Aus mehreren Elementen eine einzelne Kopiervorlage erzeugen

Um aus mehreren Elementen eine einzelne Kopiervorlage für alle Elemente zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliothek in der Task Card "Bibliotheken".
2. Kopieren Sie die Elemente, die Sie als Kopiervorlagen anlegen möchten, in die Zwischenablage.

3. Klicken Sie in der Bibliothek mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner.
4. Wählen Sie im Kontextmenü den Befehl "Als einzelne Kopiervorlage einfügen".

Alternative:

1. Selektieren Sie die gewünschten Elemente.
2. Ziehen Sie die Elemente per Drag&Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen". Halten Sie währenddessen die Taste <Alt> gedrückt.
   Die Elemente werden als eine einzelne Kopiervorlage in die Bibliothek eingefügt. Die einzelne Kopiervorlage enthält alle gewählten Elemente. Von eventuell enthaltenen Objekten (z. B. referenzierten Bausteinen) wird automatisch jeweils ein Typ erzeugt.

Hinweis

Komplexe Strukturen von Kopiervorlagen vermeiden

Um Namenskonflikten und Konflikten hinsichtlich der Ordnerstruktur beim späteren Verwenden von Kopiervorlagen vorzubeugen, vermeiden Sie komplexe Kopiervorlagen. Komplexe Kopiervorlagen sind z. B. Kopiervorlagen, die aus mehreren Elementen und verschachtelten Ordnern bestehen.

Siehe auch

Grundlagen zu Kopiervorlagen (Seite 51)

Kopiervorlagen verwenden (Seite 55)

Grundlagen zu Bibliotheken (Seite 7)

Typen zur Projektbibliothek hinzufügen (Seite 64)

7.3 Kopiervorlagen filtern

Um bei einer großen Anzahl an Kopiervorlagen die Übersichtlichkeit zu erhöhen, können Sie die Anzeige nach der Art der Kopiervorlage filtern.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um die Ansicht zu filtern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Kopiervorlagen" in der Projektbibliothek oder einer globalen Bibliothek.
2. Wählen Sie in der Klappliste in der Funktionsleiste die Art der Objekte, die Sie unter "Kopiervorlagen" anzeigen möchten.

Ergebnis

Es wird nur die gewählte Art von Kopiervorlagen angezeigt. Um zu einer ungefilterten Ansicht zurückzukehren, setzen Sie den Filter auf "Alle".

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Ordner in einer Bibliothek erstellen (Seite 113)

Grundlagen zu Kopiervorlagen (Seite 51)

Kopiervorlagen verwenden (Seite 55)

Elementeansicht verwenden (Seite 16)

7.4 Kopiervorlagen verwenden

Kopiervorlagen befinden sich entweder in der Projektbibliothek oder in einer globalen Bibliothek. Sie können eine oder mehrere Kopiervorlagen auf einmal im Projekt einfügen. Wenn Sie mehrere Kopiervorlagen auf einmal einfügen, achten Sie darauf, dass alle Kopiervorlagen mit der gewünschten Verwendungsstelle kompatibel sind.

Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet.

Vorgehen

Um Kopiervorlagen in das Projekt einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in einer Bibliothek den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".
2. Ziehen Sie die gewünschten Kopiervorlagen oder ganze Ordner per Drag&Drop an die Verwendungsstelle.

Oder:

1. Öffnen Sie die Elementeansicht.
2. Ziehen Sie die gewünschten Kopiervorlagen oder ganze Ordner per Drag&Drop aus der Palette "Elemente" an die Verwendungsstelle.

Ergebnis

Von den Kopiervorlagen wird eine Kopie eingefügt. Wenn in einer Mehrfachauswahl nicht kompatible Kopiervorlagen enthalten waren, dann werden diese ausgelassen und es wird keine Kopie im Projekt erzeugt.

Siehe auch

Grundlagen zu Kopiervorlagen (Seite 51)

Kopiervorlagen hinzufügen (Seite 52)

Kopiervorlagen filtern (Seite 54)

Grundlagen zu Bibliotheken (Seite 7)

Elementeansicht verwenden (Seite 16)

Typen und ihre Versionen einsetzen 8

8.1 Grundlagen zu Typen

Verwendung von Typen

Typen sind Elemente, die für den Ablauf von Anwenderprogrammen notwendig sind. Typen lassen sich versionieren und zentral weiterentwickeln.

Die folgenden Elemente lassen sich als Typ in der Projektbibliothek oder der globalen Bibliothek ablegen:

- Funktionen (FCs)
- Funktionsbausteine (FBs)
- PLC-Datentypen (UDTs)
- Bildbausteine
- Bilder
- Stile
- Anwenderdefinierte Funktionen

Innerhalb einer CPU können Sie die gleiche Version eines Typen nur einmal verwenden. Wenn es verschiedene Versionen eines Typs gibt, dann können diese auch innerhalb einer CPU alle verwendet werden. Innerhalb eines Projekts lassen sich beliebig viele Instanzen, abhängig von der Anzahl der verwendeten CPUs, ableiten. Die Instanzen sind anschließend mit der Version des Typen in der Bibliothek verbunden. Wenn Sie Typen aus einer globalen Bibliothek verwenden, dann wird der Typ zusätzlich in der Projektbibliothek angelegt. Ist der Typ bereits in der Projektbibliothek vorhanden, werden bei Bedarf fehlende Typ-Versionen ergänzt. Die Instanz ist anschließend nur mit der jeweiligen Typ-Version in der Projektbibliothek verbunden.

Typen und ihre Instanzen werden mit einem schwarzen Dreieck gekennzeichnet. Das folgende Bild zeigt eine Instanz, gekennzeichnet mit schwarzem Dreieck, und einen gewöhnlichen Programmbaustein:

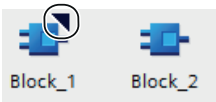

Grundlagen zur Versionierung von Typen

Mithilfe der Versionierung von Typen haben Sie die Möglichkeit, Typen zentral zu entwickeln und dann die neuste Version an die einzelnen Projekte als Aktualisierung auszurollen. Fehlerkorrekturen und Funktionserweiterungen lassen sich auf diese Weise einfach in bestehende Projekte integrieren. Wenn Sie eine neue Fassung einer globalen Bibliothek erstellt haben, aktualisieren Sie bestehende Projekte in einem automatischen Prozess. Dadurch werden Fehler minimiert und der Arbeitsaufwand zur Pflege großer Automatisierungslösungen mit vielen einzelnen Projekten sinkt.

Die Versionierung macht den Entwicklungsprozess einzelner Typen nachvollziehbar. Bevor Sie eine Version freigeben, können Sie in einer Testumgebung ausprobieren, ob sich die Änderungen an einem Typen reibungslos in ein bestehendes Projekt integrieren. Erst wenn Sie sichergestellt haben, dass alles fehlerfrei funktioniert, geben Sie eine Version für den Produktiveinsatz frei. Sie können jederzeit die Historie einzelner Instanzen im Projekt sehen und ermitteln, aus welcher Version die Instanz abgeleitet wird.

Das TIA Portal prüft außerdem automatisch, ob es zusammenhängende Objekte mit einzelnen Versionen eines Typen gibt. Zusammenhängende Objekte können beispielsweise in einem Baustein referenzierte PLC-Datentypen (UDTs) oder andere Bausteine sein. Schon beim Erstellen eines Typen oder dem Kopieren zwischen Bibliotheken werden alle zusammenhängenden Objekte berücksichtigt. Damit keine Inkonsistenzen im Projekt entstehen, werden Versionen von Typen vor der Freigabe auf ihre Konsistenz geprüft.

Versionen von Typen

Jedem Typen sind Versionen zugeordnet. Die Versionsnummer wird sowohl in der Task Card "Bibliotheken" als auch in der Bibliotheksansicht neben dem jeweiligen Typen angezeigt. Außerdem können Sie die Version eines Typen in der Projektnavigation neben den Instanzen anzeigen. Blenden Sie dazu die Spalte "Version" in der Projektnavigation ein. So können Sie jederzeit sehen, in welcher Version eine Instanz im Projekt verwendet wird.

Die Versionsnummer besteht aus drei durch Punkte getrennte Stellen. Die ersten beiden Stellen können Sie frei bestimmen. Zahlen von 1 bis 999 sind für die ersten beiden Stellen jeweils zulässig. Die dritte Stelle ist die Build-Nummer. Sie wird automatisch um eins erhöht, sobald Sie eine der Version zugehörige Instanz bearbeiten. Wenn Sie die Version freigeben, wird die Build-Nummer auf 1 zurückgesetzt.

Hinweis

Bibliothekseinstellung: "Bearbeitung der Typ-Versionsnummer deaktivieren"

Mithilfe der Bibliothekseinstellung "Bearbeitung der Typ-Versionsnummer deaktivieren" können Sie unterbinden, dass bei den Aktionen "Version freigeben" und "Version zuweisen" die Typ-Versionsnummer vom Anwender zugewiesen werden kann. Durch das Aktivieren der Bibliothekseinstellung kann die Typ-Versionsnummer vom Anwender nicht mehr verändert werden. Die Build-Nummer wird beim Freigeben oder Zuweisen einer Version weiterhin automatisch hochgezählt. Wenn die Obergrenze der Build-Nummer in einem Block (999) erreicht wurde, dann wird die Versionsnummer im nächsten Block hochgezählt.

Die Versionen von Typen können drei Zustände besitzen:

- In Bearbeitung (Bildbausteine und HMI Anwenderdatentypen)
- Im Test (Alle Arten von Typen außer Bildbausteine und HMI Anwenderdatentypen)
- Freigegeben

Das folgende Bild zeigt einen Typ mit zwei Versionen. Eine Version befindet sich im Zustand "im Test" und eine Version ist freigegeben:

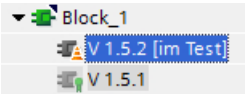

Falls Sie Funktionen (FCs) oder Funktionsbausteine (FBs) als Typen in der Projektbibliothek verwenden, beachten Sie Folgendes beim Übersetzen des Bausteins, wenn Sie unterschiedliche Anweisungsversionen nutzen:

- Sie können know-how-geschützte Bausteine ohne Passwort neu übersetzen, wenn die Schnittstellen der Bausteinversionen identisch sind.
- Sie können nicht know-how-geschützte Bausteine neu übersetzen, wenn für keinen Aktualparameter die Versorgung verloren geht und wenn durch die neue Version keine neuen Aktualparameter hinzukommen, die Sie versorgen müssten.
- Sollte sich der Typ mit einer anderen Anweisungsversion nicht übersetzen lassen, dann ist eine neue Version des Typen erforderlich, welche die notwendigen Anpassungen enthält.

Siehe auch

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Kopiervorlagen (Seite 51)

Typen zur Projektbibliothek hinzufügen (Seite 64)

Typen aus der Projektbibliothek verwenden (Seite 68)

Bibliothekselemente bearbeiten (Seite 115)

Typen duplizieren (Seite 68)

8.2 Zustand von Typ-Versionen

Die Versionen von Typen können sich in drei verschiedenen Zuständen befinden. Die Zustände sind an der Instanz oder in der Bibliothek zu erkennen.

Zustand "In Bearbeitung"

Nur Versionen von Bildbausteinen und HMI Anwenderdatentypen haben den Zustand "In Bearbeitung". Wenn sich eine Version in Bearbeitung befindet, steht "in Bearbeitung" neben der Version in der Bibliothek.

Wenn Sie einen neuen Typen oder eine neue Version eines freigebenen Typen erstellen, wird der Typ in den Zustand "in Bearbeitung" gesetzt.

Typen mit dem Zustand "in Bearbeitung" lassen sich in der Bibliotheksansicht bearbeiten. Ein Bezug zu einer Instanz im Projekt muss nicht vorhanden sein. Die Kompatibilität des Typen wird bei der Freigabe mit einer Konsistenzprüfung geprüft.

Zustand "Im Test"

Im Zustand "im Test" können sich alle Typen außer Bildbausteine und HMI Anwenderdatentypen befinden. Wenn sich eine Version im Test befindet, steht "im Test" neben der Instanz und in der Bibliothek. Eine Version, die sich im Test befindet, ist mit einer Testinstanz im Projekt verknüpft. Sie können so die Auswirkungen Ihrer Änderungen in einer Testumgebung inklusive aller Online-Funktionen ausprobieren, bevor Sie einen Typen zur Verwendung im laufenden Betrieb freigeben.

Zustand "Freigegeben"

Der Zustand "Freigegeben" ist für alle Typen verfügbar, unabhängig von der Verwendungsstelle. Wenn eine Version freigegeben ist, dann wird das Symbol der Version in der Bibliothek mit einem Siegel gekennzeichnet:

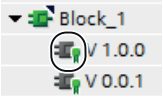

Freigegebene Versionen lassen sich in ihrer Instanz schreibgeschützt öffnen. Wenn Sie eine freigegebene Version bearbeiten möchten, müssen Sie zunächst eine neue Version "in Bearbeitung" oder "im Test" erzeugen.

Siehe auch

Grundlagen zu Typen (Seite 57)

Typen aus der Projektbibliothek verwenden (Seite 68)

Testversion eines Typen erstellen (Seite 71)

Testversion eines Typs bearbeiten (Seite 73)

Bearbeitungsversion eines Typen erstellen (Seite 73)

Konsistenzprüfung für eine Version durchführen (Seite 74)

Versionen verwerfen (Seite 75)
Versionen freigeben (Seite 76)
Version zuweisen (Seite 87)
Projekt auf die neuesten Versionen aktualisieren (Seite 79)
Verbindung zwischen Instanz und Typ aufheben (Seite 80)
Anzeige der Typen filtern (Seite 26)

8.3 Freigegebene Typ-Version anzeigen

Wenn Sie eine freigegebene Version anzeigen, aber nicht bearbeiten möchten, öffnen Sie die Instanz schreibgeschützt. Alle Typen außer Bildbausteine und HMI Anwenderdatentypen lassen sich direkt an der Instanz öffnen. Bildbausteine und HMI Anwenderdatentypen lassen sich ausschließlich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht öffnen.

Voraussetzung

Die freigegebene Version besitzt eine Instanz im Projekt, außer es handelt sich bei dem Typ um einen Bildbaustein oder einen HMI Anwenderdatentyp.

Typ-Version an einer Instanz öffnen

Um eine freigegebene Version eines Typen ausgehend von einer Instanz schreibgeschützt zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die freigegebene Version an der Instanz in der Projektnavigation.
2. Klicken Sie mit der rechten Maustaste auf die Instanz und wählen Sie im Kontextmenü den Befehl "Öffnen".
   Die Instanz wird schreibgeschützt geöffnet.

Typ-Version in der Task Card "Bibliotheken" oder Bibliotheksansicht öffnen

Um eine freigegebene Version eines Typen in der Task Card "Bibliotheken" oder der Bibliotheksansicht zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Version.
2. Klicken Sie mit der rechten Maustaste auf die Version und wählen Sie im Kontextmenü den Befehl "Öffnen".
   Wenn es sich um einen Bildbaustein oder HMI Anwenderdatentyp handelt, wird dieser direkt in der Bibliotheksansicht geöffnet. Lassen Sie in diesem Fall die weiteren Schritte aus. Wenn es sich um einen anderen Typen handelt, wird der Dialog "Typ öffnen" geöffnet.
3. Selektieren Sie in der Liste der Instanzen die Instanz mit der Version, die Sie anzeigen möchten.
4. Bestätigen Sie mit "OK".
   Die Instanz wird schreibgeschützt geöffnet.

8.4 Eigenschaften eines Typen oder einer Version anzeigen

Sie können sich die Eigenschaften eines Typs oder einer Version anzeigen lassen.

Vorgehen

Um die Eigenschaften eines Typs oder einer Version anzuzeigen und einen Kommentar einzugeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Typ oder die Version eines Typs in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
2. Klicken Sie mit der rechten Maustaste auf den Typen oder auf eine seiner Versionen und wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Dialog "Eigenschaften" wird geöffnet.
3. Geben Sie bei Bedarf einen Kommentar zu dem Typen in das Feld "Kommentar" ein oder ändern Sie einen bereits vorhandenen Kommentar.

Sichtbare und änderbare Eigenschaften

Die folgende Tabelle zeigt die Eigenschaften, die Sie bei einem Typen oder einer Version sehen und ändern können:

Eigenschaft	Beschreibung	Version	Typen
Name	Name des Typen	-	Sichtbar und änderbar
Version	Versionsnummer	Sichtbar	-
Letzte Änderung	Wenn Sie einen Typ erstellen, freigeben, duplizieren oder eine Version zuweisen, dann wird dies als Änderung am Typen erfasst. Das Datum und die Uhrzeit der Änderung werden in das Feld "Letzte Änderung" eingetragen.	Sichtbar	-
Autor	Als Autor wird der Ersteller der Version angezeigt.	Sichtbar	-
Ursprüngliche Bibliothek	Das Projekt und die Bibliothek werden anzeigt, aus der die aktuelle Version des Typen generiert wurde. Diese Information ist beispielsweise wichtig, um das Original des Typs wiederzufinden, nachdem er aus einer anderen Bibliothek kopiert wurde.	Sichtbar	-
Versions-GUID oder Typ-GUID	Mit der GUID ist der Typ oder die Version des Typen eindeutig identifizierbar, auch wenn es z. B. innerhalb der Projektbibliothek oder der globalen Bibliothek Typen oder Versionen mit einer identischen Benennung gibt. Die GUID ist nicht änderbar und wird automatisch vergeben.	Sichtbar	Sichtbar
Kommentar	Kommentar zum Typ oder zur Version	Sichtbar und bei Versionen in den Zuständen "im Test" oder "in Berbeitung" änderbar	Sichtbar und änderbar

8.5 Arbeiten mit Typen in der Projektbibliothek

8.5.1 Typen zur Projektbibliothek hinzufügen

In der Projektbibliothek lassen sich von etlichen Elementen Typen zur Wiederverwendung im Projekt anlegen. Folgende Elemente können Sie z. B. als Typ anlegen:

- Programmbausteine (FCs/FBs)
- PLC-Datentypen (UDTs)
- Bildbausteine
- HMI Anwenderdatentyp
- Bilder
- Stile

Wenn Sie ein Element als Typ zur Projektbibliothek hinzufügen, welches Abhängigkeiten zu anderen Elementen besitzt, dann werden die abhängigen Elemente ebenfalls automatisch als Typ angelegt.

Nach dem Hinzufügen eines Typs zur Projektbibliothek ist der Typ mit dem hinzugefügten Element aus dem Projekt verbunden. Das Element aus dem Projekt ist damit die Instanz des Typs.

Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Die Elemente, die Sie als Typ hinzufügen möchten, sind übersetzt.
- Die Elemente befinden sich in konsistentem Zustand.
- Bei Bausteinen sind alle weiteren Anforderungen, wie im Kapitel "Anforderungen an Bausteine (Seite 65) beschrieben, erfüllt

Vorgehen

Um der Projektbibliothek ein bestehendes Element als Typ hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektbibliothek in der Task Card "Bibliotheken".
2. Ziehen Sie ein oder mehrere Elemente per Drag & Drop in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".
   Alternativ: Kopieren Sie die Elemente in der Projektnavigation in die Zwischenablage und fügen Sie die Elemente in den gewünschten Ordner der Projektbibliothek ein.
   Der Dialog "Typ hinzufügen" wird geöffnet.

3. Geben Sie die Eigenschaften des neuen Typs ein:
   - Geben Sie einen Namen für den neuen Typ in das Feld "Typname" ein.
   - Geben Sie in das Feld "Version" eine Versionsnummer für den neuen Typen ein.
   - Geben Sie in das Feld "Autor" den Namen des für den Typen verantwortlichen Bearbeiters ein.
   - Geben Sie in das Feld "Kommentar" einen Kommentar zu dem Typen ein.
4. Bestätigen Sie mit "OK".
   Der neue Typ wird mit einer freigegebenen Version erzeugt. Die Version ist mit dem hinzugefügten Element (Instanz) verbunden.

Hinweis

Name für Autor einstellen

Sie können in den Einstellungen einen Benutzernamen eintragen. Wenn Sie in der Bibliothek Typen hinzufügen, Typversionen erstellen und freigeben oder Kopiervorlagen erstellen, wird der in den Einstellungen eingetragene Benutzer standardmäßig in den Eigenschaften als Autor hinterlegt.

Siehe auch

Grundlagen zu Typen (Seite 57)

Typen duplizieren (Seite 68)

Anforderungen an Bausteine (Seite 65)

Referenzieren von Objekten innerhalb eines Bausteins (Seite 66)

Grundlagen zu Bibliotheken (Seite 7)

Kopiervorlagen hinzufügen (Seite 52)

8.5.2 Anforderungen an Bausteine

Zulässige Bausteine für das Erstellen eines Typs

Von folgenden Bausteinen lassen sich Typen in der Projektbibliothek erstellen:

- Funktionsbausteine (FBs)
- Funktionen (FCs)
- PLC-Datentypen (UDTs)

Konsistenz und Übersetzung

Um aus einem Baustein einen Typ zu erzeugen, muss der Baustein konsistent und übersetzt sein. Wenn Sie einen Typ erstellen, dann wird automatisch die Konsistenz des Bausteins geprüft und der Baustein wird bei Bedarf übersetzt.

Anforderungen an den Baustein

Um aus einem Baustein einen Typ zu erzeugen, muss der Baustein folgende Anforderungen erfüllen:

- Der Baustein passt zum Typ der CPU.
  Einen Baustein, der nicht zu einer CPU passt, erkennen Sie in der Projektnavigation an einem Inkompatibilitäts-Symbol rechts neben dem Baustein. Der Fall tritt z. B. dann ein, wenn Sie einen Baustein von einer S7-1500-CPU in eine S7-300-CPU kopieren. Ein Baustein einer S7-1500 ist nicht mit einer S7-300-CPU kompatibel und lässt sich nicht übersetzen. Daher lässt sich aus dem Baustein kein Typ erzeugen.
- Der Baustein ist kein Systemdatenbaustein.
- Der Baustein enthält keinen Einzelinstanzaufruf eines Instanzdatenbausteins oder einen Aufruf eines Datenbausteins innerhalb eines AWL-Bausteins mit dem Befehl "OPN".

Besonderheiten bei Know-how-geschützten Bausteinen

Von know-how-geschützten Bausteinen lassen sich ebenfalls Typen erstellen. Beachten Sie jedoch die folgenden zusätzlichen Anforderungen an know-how-geschützte Bausteine:

- Freigabe des Bausteins zur Verwendung als Typ
  Know-How-geschützte Bausteine müssen zur Verwendung in einer Bibliothek freigegeben sein. Die entsprechende Einstellung nehmen Sie in den Bausteineigenschaften vor.
- Der Baustein enthält keine Zugriffe auf Datenbausteine, PLC-Variablen oder PLC-Konstanten.
  Da aus Datenbausteinen, PLC-Variablen und PLC-Konstanten keine Typen erzeugt werden können, lässt sich aus know-how-geschützten Bausteinen mit Zugriffen auf die genannten Objekte ebenfalls kein Typ erzeugen. Die Instanz eines know-how-geschützten Bausteins ist ohne die zugehörigen Datenbausteine, PLC-Variablen oder PLC-Konstanten nicht funktionstüchtig.
- Wenn Sie in der Task Card "Bibliotheken" oder in der Bibliotheksansicht die Testversion eines Typs bearbeiten, können Sie in den Bausteineigenschaften einen Know-how-Schutz einrichten.
  Der Know-how-Schutz lässt sich über die Bausteineigenschaften in der Projektnavigation wieder entfernen.

Siehe auch

Typen zur Projektbibliothek hinzufügen (Seite 64)

Referenzieren von Objekten innerhalb eines Bausteins (Seite 66)

8.5.3 Referenzieren von Objekten innerhalb eines Bausteins

In vielen Fällen referenzieren Sie innerhalb eines Typs andere Objekte. Zum Beispiel rufen Sie innerhalb eines Bausteins eine Variable oder einen Datenbaustein auf. Zwischen dem Baustein und den referenzierten Objekten besteht folglich eine Abhängigkeit. Der Baustein ist nur dann konsistent, wenn alle abhängigen Objekte ebenfalls verfügbar sind.

Referenzieren anderer Typen

Wenn Sie in einem Baustein andere Programmbausteine, Bildbausteine, PLC-Datentypen (UDTs) oder HMI Anwenderdatentypen aufrufen, dann werden diese abhängigen Objekte automatisch ebenfalls als Typ in der Projektbibliothek abgelegt. Beim späteren Wiederverwenden des referenzierenden Bausteins als Instanz im Projekt wird von den referenzierten Objekten ebenfalls eine Instanz angelegt. Das TIA Portal sorgt somit automatisch für einen stets konsistenten Bausteinzustand und ein ablauffähiges Anwenderprogramm.

Referenzieren weiterer Objekte

In Bausteinen ohne Know-how-Schutz sind Zugriffe auf Datenbausteine, PLC-Variablen oder PLC-Konstanten zulässig. Aus den Bausteinen lassen sich deshalb Typen erzeugen. Während jedoch z. B. in einem Baustein referenzierte Funktionsbausteine automatisch ebenfalls als Typ in der Projektbibliothek angelegt werden, ist dies bei Zugriffen auf Datenbausteine, PLC-Variablen und PLC-Konstanten nicht der Fall. Werden Datenbausteine, PLC-Variablen oder PLC-Konstanten in einem Baustein referenziert, dann wird lediglich der Baustein selbst als Typ angelegt. Die referenzierten Objekte werden nicht als Typ angelegt. Wenn Sie Objekte referenzieren, die nicht als Typ angelegt werden können, dann beachten Sie Folgendes:

- Verwenden des Bausteins an einer anderen Verwendungsstelle
  Wenn Sie aus dem Typ eine Instanz an einer anderen als der ursprünglichen Verwendungsstelle (z. B. eine andere CPU) erstellen, dann fehlen die referenzierten Objekte an der späteren Verwendungsstelle. Erstellen Sie daher referenzierte Objekte an der späteren Verwendungsstelle neu (z. B. Datenbausteine oder Variablentabellen). Achten Sie darauf, dass die referenzierten Objekte auch an der späteren Verwendungsstelle gleich benannt sind.
- Nachträgliche Namensänderungen an referenzierten Objekten
  Wenn Sie die Namen von referenzierten Objekten nachträglich ändern, dann ist der Baustein nicht mehr konsistent. Das referenzierte Objekt (z. B. eine Variable) wird nicht mehr gefunden. Bei bereits freigegebenen Versionen des Typs können Sie den Aufruf des Objekts nicht mehr korrigieren. Erstellen Sie in diesem Fall eine neue Version des Typs und korrigieren Sie den Namen des aufgerufenen Objekts. Aktualisieren Sie anschließend alle Instanzen des Typs im Projekt auf die neueste Version.

Siehe auch

Anforderungen an Bausteine (Seite 65)

Typen zur Projektbibliothek hinzufügen (Seite 64)

Konsistenzprüfung für eine Version durchführen (Seite 74)

Projekt auf die neuesten Versionen aktualisieren (Seite 79)

Beziehungen zu anderen Bibliotheksobjekten anzeigen (Seite 29)

8.5.4 Typen duplizieren

Typen in der Projektbibliothek lassen sich duplizieren. Wenn Sie einen Typ duplizieren, dann gilt Folgendes für das Duplikat:

- Das Duplikat wird im selben Ordner erstellt.
- Das Duplikat wird aus der höchsten Version des Typs erstellt.
- Das Duplikat besitzt keine Instanz im Projekt.

Voraussetzung

Der Typ ist freigegeben.

Vorgehen

Um einen Typ in der Projektbibliothek zu duplizieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen Typ.
2. Wählen Sie im Kontextmenü den Befehl "Typ duplizieren".
   Der Dialog "Typ duplizieren" wird geöffnet.
3. Geben Sie die Eigenschaften des neuen Typs ein:
   - Geben Sie einen Namen für den neuen Typ in das Feld "Name des Typen" ein.
   - Geben Sie in das Feld "Version" eine Versionsnummer für den neuen Typ ein.
   - Geben Sie in das Feld "Bearbeiter" den Namen des für den Typ verantwortlichen Bearbeiters ein.
   - Geben Sie in das Feld "Kommentar" einen Kommentar zu dem Typen ein.
4. Bestätigen Sie mit "OK".
   Der neue Typ wird mit einer freigegebenen Version erzeugt.

Siehe auch

Typen zur Projektbibliothek hinzufügen (Seite 64)

Bibliothekselemente bearbeiten (Seite 115)

8.5.5 Typen aus der Projektbibliothek verwenden

Typen aus der Projektbibliothek lassen sich beliebig oft innerhalb eines Projekts verwenden. Die Verwendung eines Typen ist immer mit einer Version des jeweiligen Typen in der Projektbibliothek verknüpft. Wenn der Typ abhängige Elemente enthält, dann werden die abhängigen Elemente ebenfalls als Verwendung an geeigneter Stelle im Projekt angelegt. Abhängige Elemente sind z. B. in einem Baustein referenzierte PLC-Datentypen.

Einem Gerät können Sie nur eine Version desselben Typen zuweisen. Bei Bedarf können Sie Verwendungen von mehreren Typen auf einmal erstellen.

Verwendungen eines Typen in der Projektnavigation heißen Instanz.

Mögliche Verwendungsstellen für Typ-Versionen

Um einen Typen zu verwenden, erzeugen Sie an einer geeigneten Stelle im Projekt eine Verwendung von einer bestimmten Version des Typen. Geeignete Verwendungsstellen sind:

- Ordner in der Projektnavigation
  In einem Ordner in der Projektnavigation lässt sich eine Instanz eines Typen erzeugen. Der Ordner muss für die Art des Typen geeignet sein. Wenn es sich bei dem Typen z. B. um einen Anwenderdatentyp handelt, dann können Sie die Instanz nur im Ordner "PLC-Datentypen" erzeugen.
- Editor
  In einem geeigneten Editor lässt sich von einem Typen eine Verwendung erzeugen. Sie können z. B. vom Typen eines Funktionsbausteins eine Verwendung in einem anderen Baustein erzeugen. Der Typ des Funktionsbausteins wird so aus einem anderen Baustein heraus aufgerufen.

Möglichkeiten eine Verwendung zu erzeugen

Sie haben zwei Möglichkeiten, Verwendungen von Typen zu erzeugen:

- Version eines Typen per Drag & Drop aus der Projektbibliothek an die Verwendungstelle ziehen
  Von den Typen und ihren abhängigen Elementen werden Verwendungen erzeugt und an der Verwendungsstelle eingefügt. Die Verwendungen sind an die jeweilige Version des Typen in der Projektbibliothek gebunden. Wenn Sie die Verwendungen in einem Editor erzeugen, werden von den Typen auch an den entsprechenden Stellen in der Projektnavigation Instanzen erzeugt. Standardmäßig wird die Ordnerstruktur aus der Bibliothek in der Projektnavigation übernommen. Sollten Sie sich nicht bewusst für einen anderen Ordner in der Projektnavigation entschieden haben, so finden Sie die Instanzen in den gleichen Ordnern wie in der Bibliothek wieder.
- Kopieren und Einfügen von Typ-Verwendungen
  Verwendungen von Typen können Sie in die Zwischenablage kopieren und an anderer Stelle wieder einfügen. So haben Sie eine weitere Verwendung der Typ-Version erzeugt. Die Verwendung ist weiterhin an dieselbe Typ-Version in der Projektbibliothek gebunden. Wenn Sie die Verwendung eines Typen in die Zwischenablage kopieren und in einem anderen Projekt wieder einfügen, dann werden alle benötigten Typ-Versionen in die Projektbibliothek des anderen Projekts übernommen.

Voraussetzung

- Die gewünschten Versionen sind freigegeben.
- Im Projekt befindet sich bereits ein Gerät, das die Art der gewünschten Typen unterstützt
- Dem Gerät ist noch keine weitere Verwendung derselben Typen zugeordnet.

Vorgehen

Um eine Verwendung eines Typen zu erzeugen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektbibliothek die Versionen, von denen Sie eine Instanz erzeugen möchten.
2. Ziehen Sie die Versionen per Drag & Drop aus der Projektbibliothek an die Verwendungsstelle in der Projektnavigation oder einem Editor.
   Alternativ: Um automatisch die neueste Version zu verwenden, ziehen Sie die Typen selbst per Drag & Drop aus der Projektbibliothek an die Verwendungsstelle.
   Ziehen Sie z. B. den Typen eines Funktionsbausteins per Drag & Drop in den Bausteinordner einer CPU in der Projektnavigation. Um den Typen z. B. direkt aus einem anderen Baustein heraus aufzurufen, ziehen Sie den Typen per Drag & Drop aus der Projektbibliothek an die Verwendungsstelle im Programmiereditor.

Alternative:

1. Kopieren Sie ein oder mehrere Verwendungen in die Zwischenablage.
2. Fügen Sie die die Verwendungen an einer geeigneten Stelle im selben Projekt oder in einem anderen Projekt wieder ein.

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Typen zu einer Instanz anzeigen (Seite 70)

Grundlagen zu Bibliotheken (Seite 7)

Kopiervorlagen verwenden (Seite 55)

8.5.6 Typen zu einer Instanz anzeigen

Sie können aus der Projektnavigation zu dem Typen in die Projektbibliothek springen, der zur Instanz in der Projektnavigation gehört.

Vorgehen

Um zu dem Typ in die Projektbibliothek zu springen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Instanz des Typs in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zum Typen".
   Die referenzierte Typ-Version des zugehörigen Typs wird in der Projektbibliothek angezeigt.

Siehe auch

Typen aus der Projektbibliothek verwenden (Seite 68)

8.5.7 Testversion eines Typen erstellen

Bevor Sie einen Typen für den Produktiveinsatz freigeben, testen Sie den Typen innerhalb eines Projekts und an der Automatisierungsanlage. Der Test erfolgt in einer bestimmten Testumgebung. Diese Testumgebung kann beispielsweise eine CPU sein.

Für den Test erstellen Sie eine Version im Zustand "im Test". Das Erstellen einer Version im Zustand "im Test" ist für alle Arten von Typen geeignet, außer für Bildbausteine und HMI Anwenderdatentypen. Von Bildbausteinen und HMI Anwenderdatentypen lassen sich hingegen Versionen im Zustand "in Bearbeitung" erstellen.

Es gibt zwei Möglichkeiten, eine Testversion eines Typen zu erstellen und die Testumgebung festzulegen:

- In der Task Card "Bibliotheken" oder in der Bibliotheksansicht
  Die neue Version im Zustand "im Test" erzeugen Sie in der Task Card "Bibliotheken" oder in der Bibliotheksansicht. Die neue Version können Sie entweder direkt aus dem Typen oder aus einer bestimmten Version des Typen erzeugen.
- An einer Instanz in der Projektnavigation
  Die Testversion können Sie auch direkt an der Instanz in der Projektnavigation erstellen. Da die Instanz immer in einer bestimmten Version im Projekt verwendet wird, wird aus der an der Instanz verwendeten Version eine neue Version des Typen erzeugt.

Sie können auch von mehreren Typen gleichzeitig Testversionen erstellen.

Für eine Version im Zustand "im Test" gelten die folgenden Regeln:

- Es ist immer nur möglich, eine einzige Version je Typ in den Zustand "im Test" zu versetzen.
- Eine Version im Test darf nur mit einer einzigen Instanz im Projekt verknüpft sein. Daher ist es nicht möglich eine Instanz in die Zwischenablage zu kopieren, sie zu duplizieren oder einen weiteren Typen aus der Instanz zu erstellen, solange diese sich im Zustand "im Test" befindet.

Voraussetzung

- Es existiert innerhalb des Projekts mindestens eine Instanz des Typen in einer beliebigen Version.
- Wenn Sie die neue Version aus einer bestimmten Version des Typen erstellen möchten, muss die Instanz im Projekt in dieser Version verwendet sein.

Vorgehen

Um eine neue Testversion eines Typen oder der Version eines Typen zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Typen, eine Version des Typen oder die Instanz.
   Wenn Sie die Testversion direkt an der Instanz erstellen, können Sie über Multiselektion mehrere Elemente oder Ordner selektieren. Da über die selektierte Instanz die Testumgebung bereits definiert ist, können Sie die Schritte 3 und 4 auslassen.
2. Klicken Sie mit der rechten Maustaste auf das selektierte Element und wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".
   Wenn Sie das Bearbeiten in der Task Card "Bibliotheken" oder in der Bibliotheksansicht gestartet haben, wird der Dialog "Typ bearbeiten" geöffnet. Wenn Sie das Bearbeiten an der Instanz in der Projektnavigation begonnen haben, wird die Testinstanz sofort zum Bearbeiten in der Bibliotheksansicht geöffnet.
3. Wählen Sie aus der Liste eine Instanz des Typen im Projekt aus.
   Wenn Sie das Bearbeiten am Typen selbst gestartet haben, dann gilt:
   - Die Stelle, an der die Instanz verwendet wird (z. B. die CPU), dient für das anschließende Bearbeiten des Typen als Testumgebung.
   - Mit der Auswahl der Testinstanz legen Sie auch die Version fest, die bearbeitet wird.

   Für das Bearbeiten einer bestimmten Version gilt:
   Wenn Ihr Ausgangspunkt eine bestimmte Version ist, können Sie in der Liste nur Instanzen wählen, die in derselben Version verwendet werden.
4. Bestätigen Sie mit "OK".

Ergebnis

Eine neue Version des Typen wird erstellt. Die neue Version befindet sich im Zustand "im Test" und wird an der Benutzeroberfläche entsprechend gekennzeichnet.

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

8.5.8 Testversion eines Typs bearbeiten

Wenn Sie eine Version im Zustand "im Test" weiterbearbeiten, wird keine neue Version erstellt. Sie können das Bearbeiten der Testversion an der Instanz in der Projektnavigation, in der Task Card "Bibliotheken" oder in der Bibliotheksansicht starten.

Hinweis

Löschen und Umbenennen von Schnittstellen-Parametern

Sie können neue Parameter hinzufügen. Wenn Sie jedoch bereits existierende Parameter umbenennen oder löschen, hat dies den Verlust der Parameterversorgung beim Bausteinaufruf zur Folge.

Vorgehen

Um die Testversion eines Typs zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Testversion oder die Instanz.
2. Wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".
   Die Testinstanz wird geöffnet und kann bearbeitet werden.

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

Konsistenzprüfung für eine Version durchführen (Seite 74)

Versionen verwerfen (Seite 75)

Versionen freigeben (Seite 76)

8.5.9 Bearbeitungsversion eines Typen erstellen

Wenn Sie einen Typen mit Bildbausteinen oder HMI Anwenderdatentypen bearbeiten möchten, erstellen Sie eine neue Version des Typen im Zustand "in Bearbeitung". Das Bearbeiten der neuen Version erfolgt in der Bibliotheksansicht. Um die Kompatibilität der Änderungen zu prüfen, wird für den Typen vor der Freigabe automatisch eine Konsistenzprüfung durchgeführt.

Voraussetzung

Die Projektbibliothek ist in der Task Card "Bibliotheken" oder der Bibliotheksansicht geöffnet.

Vorgehen

Um eine neue Version eines Typen in Bearbeitung zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Typen oder die Version des Typen.
2. Wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".
   Es wird eine neue Version "in Bearbeitung" erstellt und zum Bearbeiten in der Bibliotheksansicht geöffnet.

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

8.5.10 Konsistenzprüfung für eine Version durchführen

Bei der Bearbeitung einer Typ-Version kann diese versehentlich in einen inkonsistenten Zustand geraten. Um Fehler rechtzeitig im Entwicklungsprozess zu bemerken, können Sie regelmäßig eine Konsistenzprüfung durchführen. Die Konsistenzprüfung erfolgt jedoch immer automatisch, sobald Sie eine Version freigeben.

Im Folgenden erfahren Sie, wie Sie die Konsistenzprüfung für die Version eines Typen manuell starten.

Voraussetzung

- Die Projektbibliothek ist in der Task Card "Bibliotheken" oder in der Bibliotheksansicht geöffnet.
- Die Version befindet sich im Zustand "in Bearbeitung" oder "im Test".

Vorgehen

Um eine Konsistenzprüfung für die Version eines Typen durchzuführen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Version, die Sie auf ihre Konsistenz prüfen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Konsistenzprüfung durchführen".
   Die Konsistenzprüfung wird durchgeführt. Sie erhalten eine Meldung mit dem Ergebnis der Konsistenzprüfung.

Siehe auch

Versionen freigeben (Seite 76)

Versionen verwerfen (Seite 75)

Testversion eines Typs bearbeiten (Seite 73)

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

8.5.11 Versionen verwerfen

Versionen eines Typen im Zustand "im Test" oder "in Bearbeitung" verwerfen Sie, wenn Sie die Version nicht mehr benötigen. Sie können ebenso mehrere Typen oder Ordner auswählen und alle darin enthaltenen Test- oder Bearbeitungsversionen verwerfen. Alle Verwendungen der gelöschten Versionen werden auf den letzten freigegebenen Stand zurückgesetzt.

Voraussetzung

- Die Version, die Sie verwerfen möchten, befindet sich im Zustand "im Test" oder "in Bearbeitung".
- Sie befinden Sich in der Bibliotheksansicht oder die Task Card "Bibliotheken" ist geöffnet.

Vorgehen

Um eine Version zu verwerfen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Version, die Sie verwerfen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Änderungen verwerfen und Version löschen". Die Version wird gelöscht.

Alternative in der Bibliotheksansicht:

1. Klicken Sie, während Sie eine Version zur Bearbeitung geöffnet haben, in der Funktionsleiste auf die Schaltfläche "Änderungen verwerfen und Version löschen". Die Version wird gelöscht.

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

Konsistenzprüfung für eine Version durchführen (Seite 74)

8.5.12 Alle Versionen innerhalb eines Ordners verwerfen

Sie können alle Versionen im Zustand "im Test" oder "in Bearbeitung" innerhalb eines Ordners auf einmal verwerfen. Alle Verwendungen der gelöschten Version werden auf den letzten freigegebenen Stand zurückgesetzt.

Voraussetzung

Sie befinden Sich in der Bibliotheksansicht oder die Task Card "Bibliotheken" ist geöffnet.

Version eines einzelnen Typs verwerfen

Um alle Versionen innerhalb eines Ordners zu verwerfen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Ordner.
2. Wählen Sie im Kontextmenü den Befehl "Alle verwerfen".
   Alle Versionen "im Test" oder "in Bearbeitung" werden gelöscht.

8.5.13 Versionen freigeben

Wenn Sie mit dem Bearbeiten einer Typ-Version fertig sind, geben Sie die Version für den Produktiveinsatz frei. Bei der Freigabe weisen Sie eine Versionsnummer zu. Sie können durch Mehrfachauswahl auch mehrere Versionen auf einmal freigeben.

Voraussetzung

- Die Task Card "Bibliotheken" ist geöffnet oder Sie befinden sich in der Bibliotheksansicht.
- Die Versionen, die Sie freigeben möchten, befinden sich im Zustand "im Test" oder "in Bearbeitung".
- Die Versionen befinden sich in konsistentem Zustand.
  Eine Konsistenzprüfung wird durchgeführt, sobald Sie die Freigabe starten. Wenn bei der Konsistenzprüfung freigabeverhindernde Fehler auftreten, erhalten Sie einen Hinweis, wie Sie die Fehler beheben können.

Vorgehen

Um Typ-Versionen freizugeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Versionen, die Sie freigeben möchten.
2. Klicken Sie mit der rechten Maustaste auf die Auswahl.
3. Wählen Sie im Kontextmenü den Befehl "Version freigeben".
   Der Dialog "Typ-Version freigeben" wird geöffnet.

4. Ändern Sie bei Bedarf die Eigenschaften der Version:
   - Geben Sie in das Feld "Name" einen Namen für den Typen ein. Wenn Sie mehrere Versionen für die Freigabe gewählt haben, ist das Feld "Name" nicht änderbar.
   - Legen Sie im Feld "Version" eine Haupt- und eine Zwischenversionsnummer für die freizugebende Version fest. Wenn Sie mehrere Versionen für die Freigabe gewählt haben, ist das Feld "Version" nicht änderbar und die jeweils letzte Versionsnummer wird für die Freigabe verwendet.
   - Geben Sie in das Feld "Autor" den Bearbeiter der freizugebenden Version ein.
   - Geben Sie in das Feld "Kommentar" einen Kommentar zu der freizugebenden Version ein.
5. Optional: Aktivieren Sie das Optionskästchen "Nicht verwendete Typ-Versionen aus der Bibliothek löschen", um alle Versionen aus der Bibliothek zu löschen, die mit keiner Instanz im Projekt verbunden sind. Versionen, die Abhängigkeiten zu anderen Typen oder Kopiervorlagen besitzen, werden nicht gelöscht.
6. Bestätigen Sie mit "OK".

Alternative in der Bibliotheksansicht:

1. Klicken Sie, während Sie eine Version zur Bearbeitung geöffnet haben, in der Funktionsleiste auf das Symbol "Version freigeben".
2. Fahren Sie mit den Schritten 3 bis 5 der voranstehenden Beschreibung fort.

Ergebnis

Die gewählten Versionen werden freigegeben.

Die Eigenschaften werden für die Typen selbst, die freizugebenden Versionen und für alle zukünftig folgenden Versionen übernommen. Bereits freigegebene Versionen bleiben von den Änderungen unberührt.

Gegebenenfalls werden alle Instanzen mit derselben Ausgangsversion auf die neuste Version aktualisiert und die nicht verwendeten Versionen des Typen gelöscht.

Siehe auch

Alle Versionen innerhalb eines Ordners freigeben (Seite 78)

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

Konsistenzprüfung für eine Version durchführen (Seite 74)

Version zuweisen (Seite 87)

Typen zu einer globalen Bibliothek hinzufügen (Seite 82)

Projekt auf die neuesten Versionen aktualisieren (Seite 79)

8.5.14 Alle Versionen innerhalb eines Ordners freigeben

Wenn Sie mit dem Bearbeiten aller Typen innerhalb eines Ordners fertig sind, geben Sie alle Versionen auf einmal frei.

Voraussetzung

- Sie befinden sich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
- Der Ordner enthält Versionen im Zustand "im Test" oder "in Bearbeitung".
- Alle Versionen "im Test" oder "in Bearbeitung" befinden sich in konsistentem Zustand. Eine Konsistenzprüfung wird durchgeführt, sobald Sie die Freigabe starten. Wenn bei der Konsistenzprüfung freigabeverhindernde Fehler auftreten, erhalten Sie einen Hinweis, wie Sie die Fehler beheben.

Vorgehen

Um alle Typ-Versionen innerhalb eines Orderns freizugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den gewünschten Ordner.
2. Wählen Sie im Kontextmenü den Befehl "Alle freigeben".
   Der Dialog "Typ-Version freigeben" wird geöffnet.
3. Ändern Sie bei Bedarf die Eigenschaften der Version:
   - Geben Sie in das Feld "Autor" den Bearbeiter der freizugebenden Versionen ein.
   - Geben Sie in das Feld "Kommentar" einen Kommentar zu den freizugebenden Versionen ein.
4. Aktivieren Sie das Optionskästchen "Nicht verwendete Typ-Versionen aus der Bibliothek löschen", um alle Versionen aus der Bibliothek zu löschen, die mit keiner Instanz im Projekt verbunden sind. Versionen, die Abhängigkeiten zu anderen Typen oder Kopiervorlagen besitzen, werden nicht gelöscht.
5. Bestätigen Sie mit "OK".

Ergebnis

Alle Typ-Versionen "im Test" oder in "Bearbeitung" innerhalb des gewählten Ordners werden freigegeben.

Die Eigenschaften werden für die freizugebenden Versionen und für alle zukünftig folgenden Versionen übernommen. Bereits freigegebene Versionen bleiben von den Änderungen unberührt.

Gegebenenfalls werden die nicht im Projekt verwendeten Versionen der Typen gelöscht.

Siehe auch

Versionen freigeben (Seite 76)

8.5.15 Projekt auf die neuesten Versionen aktualisieren

Nachdem Sie mehrere Typen in der Projektbibliothek aktualisiert haben, aktualisieren Sie alle Instanzen im Projekt auf die neueste Version der Typen aus der Projektbibliothek. Wenn Sie die Änderungen nicht auf das gesamte Projekt anwenden möchten, beschränken Sie die Aktualisierung auf einzelne Geräte im Projekt.

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Die gesamte Projektbibliothek
- Einzelne Ordner innerhalb der Projektbibliothek
- Einzelne Typen
  Die Auswahl mehrerer Typen ist zulässig.

Hinweis

Löschen nicht verwendeter Typ-Instanzen aus dem Projekt

Standardmäßig werden beim Aktualisieren des Projekts auf die neuesten Versionen nicht genutzte Typ-Instanzen aus dem Projekt gelöscht.

Um das Löschen nicht genutzter Typ-Instanzen aus dem Projekt zu verhindern, aktivieren Sie in den Einstellungen unter "Allgemein > Bibliothekseinstellungen" das Optionskästchen "Löschen von nicht im Projekt verwendeten Typ-Instanzen deaktivieren".

Voraussetzung

Sie befinden sich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.

Vorgehen

Um Instanzen in einem Projekt mit den Inhalten aus der Projektbibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gesamte Projektbibliothek oder Elemente daraus.
2. Klicken Sie mit der rechten Maustauste auf die gewünschten Elemente und wählen Sie im Kontextmenü den Befehl "Typen aktualisieren > Projekt".
   Der Dialog "Typen im Projekt aktualisieren" wird geöffnet.
3. Wählen Sie entweder das gesamte Projekt oder einzelne Geräte für die Aktualisierung aus.
4. Wählen Sie Optionen für den Aktualisierungsvorgang:
   - Das Optionskästchen "Instanzen im Projekt aktualisieren" ist bei diesem Vorgang immer aktiv.
   - Aktivieren Sie das Optionskästchen "Nicht verwendete Typ-Versionen aus der Bibliothek löschen", um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen.
5. Bestätigen Sie mit "OK".
   Die Aktualisierung wird durchgeführt.

Ergebnis

Am Projekt werden folgende Änderungen durchgeführt:

- Gegebenenfalls werden alle älteren Versionen aus der Projektbibliothek gelöscht.
- Alle Instanzen innerhalb der gewählten Geräte werden auf die neueste Version des verknüpften Typs aktualisiert.
- Wenn eine CPU nicht im Anwenderprogramm aufgerufene Bausteininstanzen enthält, dann werden die Instanzen der Bausteine aus der CPU entfernt. Die Typen sind in der Bibliothek weiterhin vorhanden und können bei Bedarf wieder im Projekt verwendet werden.
- In der Projektnavigation unter "Gemeinsame Daten" finden Sie ein Protokoll des Aktualisierungsvorgangs.

Siehe auch

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 85)

Bibliothek mit den Typen einer anderen Bibliothek aktualisieren (Seite 93)

Grundlagen zu Typen (Seite 57)

Grundlagen zu Bibliotheken (Seite 7)

Zustand von Typ-Versionen (Seite 60)

8.5.16 Verbindung zwischen Instanz und Typ aufheben

Instanzen von Typen sind immer an die Version des korrespondierenden Typen gebunden. Sie lassen sich nicht wie ein gewöhnliches Objekt bearbeiten. Wenn Sie die Instanz bearbeiten, wird automatisch eine neue Version des Typs in der Projektbibliothek erstellt und die Änderungen haben somit Auswirkungen auf das gesamte Projekt.

Wenn Sie die Verbindung der Instanz zu ihrem Typ trennen, bearbeiten Sie anschließend das Objekt in der Projektnavigation wie ein gewöhnliches Objekt.

Hinweis

"Verbindung zum Typ aufheben" deaktivieren

Wenn Sie verhindern möchten, dass im Projekt Instanzen von ihren Typen getrennt werden können, dann können Sie die Option "Verbindung zum Typ aufheben" auch deaktivieren.

Aktivieren Sie dafür die Option "Kontextmenüeintrag 'Verbindung zum Typ aufheben' deaktivieren" unter "Extras > Einstellungen > Allgemein > Bibliothekseinstellungen".

Voraussetzung

Die Instanz darf sich nicht im Zustand "im Test" befinden.

Vorgehen

Um die Verbindung zwischen Instanzen und ihren Typ-Versionen aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Instanzen in der Projektnavigation.
2. Klicken Sie mit der rechten Maustaste auf die Selektion und wählen Sie im Kontextmenü den Befehl "Verbindung zum Typ aufheben".
3. Die Verbindung zu den korrespondierenden Typ-Versionen wird aufgehoben.

Siehe auch

Grundlagen zu Typen (Seite 57)

Grundlagen zu Bibliotheken (Seite 7)

Zustand von Typ-Versionen (Seite 60)

8.6 Arbeiten mit Typen in globalen Bibliotheken

8.6.1 Typen zu einer globalen Bibliothek hinzufügen

Globale Bibliotheken dienen als zentrale Ressource für die Arbeit an mehreren Projekten. Ein direktes Bearbeiten von Typen ist daher den Typen in der Projektbibliothek vorbehalten. Wenn Sie an Typen arbeiten möchten, verwenden Sie daher die Projektbibliothek. Wenn Sie mit dem Bearbeiten eines Typs in der Projektbibliothek fertig sind, können Sie den Typ in eine globale Bibliothek hinzufügen. Das Hinzufügen von Typen aus der Projektbibliothek entspricht einem gewöhnlichen Kopiervorgang aus der Projektbibliothek.

Voraussetzung

- Die Task Card "Bibliotheken" oder die Bibliotheksansicht ist geöffnet.
- Die globale Bibliothek, der Sie Typen hinzufügen möchten, ist mit Schreibrechten geöffnet.

Vorgehen

Um Typen zu einer globalen Bibliothek hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Task Card "Bibliotheken" oder in der Bibliotheksansicht den gewünschten Ordner in der globalen Bibliothek.
2. Ziehen Sie einen oder mehrere Typen aus der Projektbibliothek per Drag&Drop in den Ordner "Typen" oder einen beliebigen Unterordner der globalen Bibliothek.

Alternativ:

1. Kopieren Sie die gewünschten Typen aus der Projektbibliothek in die Zwischenablage.
2. Öffnen Sie in der Task Card "Bibliotheken" in der Palette "Globale Bibliothek" die gewünschte globale Bibliothek.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

Die Typen werden in die globale Bibliothek eingefügt. Abhängige Typen, wie z. B. Typen von HMI Anwenderdatentypen oder Variablen, werden ebenfalls in die globale Bibliothek kopiert, sofern sie dort noch nicht vorhanden sind. Somit wird sichergestellt, dass alle für das Erzeugen einer Instanz notwendigen Elemente in der globalen Bibliothek vorhanden sind.

Wenn ein Typ in der globalen Bibliothek bereits vorhanden ist, dann entspricht das beschriebene Vorgehen einer Aktualisierung der globalen Bibliothek. In diesem Fall werden die neuesten, freigegebenen Versionen des Typs in der globalen Bibliothek ergänzt.

Siehe auch

Grundlagen zu Typen (Seite 57)

Versionen freigeben (Seite 76)

Version zuweisen (Seite 87)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 85)

Grundlagen zu Bibliotheken (Seite 7)

8.6.2 Typen aus einer globalen Bibliothek verwenden

Um Typen aus der globalen Bibliothek zu verwenden, erzeugen Sie an einer geeigneten Stelle im Projekt eine Verwendung einer bestimmten Version des Typen. Bei Bedarf können Sie Verwendungen von mehreren Typen auf einmal erstellen. Verwendungen eines Typen in der Projektnavigation heißen Instanz.

Mögliche Verwendungsstellen für Typ-Versionen

Geeignete Verwendungsstellen für Typen aus globalen Bibliotheken sind:

- Ordner in der Projektnavigation
  In einem Ordner in der Projektnavigation lässt sich eine Instanz eines Typen erzeugen. Der Ordner muss für die Art des Typen geeignet sein. Wenn es sich bei dem Typen z. B. um einen Anwenderdatentyp handelt, dann können Sie die Instanz nur im Ordner "PLC-Datentypen" erzeugen.
- Editor
  In einem geeigneten Editor lässt sich von einem Typen eine Verwendung erzeugen. Sie können z. B. vom Typen eines Funktionsbausteins eine Verwendung in einem anderen Baustein erzeugen. Der Typ des Funktionsbausteins wird so aus einem anderen Baustein heraus aufgerufen.

Verknüpfung der Verwendung mit der Projektbibliothek

Verwendungen von Typen aus einer globalen Bibliothek sind im Projekt nicht mit dem Typen in der globalen Bibliothek verbunden. Stattdessen wird beim Erzeugen einer Verwendung eine Kopie des Typen und seiner abhängigen Elemente in der Projektbibliothek erzeugt. Abhängige Elemente können beispielsweise PLC-Datentypen sein, die in einem Baustein referenziert werden. Die Kopie des Typen und der abhängigen Elemente in der Projektbibliothek enthält jeweils die Version, die Sie mit den Verwendungen verknüpft haben. Wenn der Typ oder ein abhängiges Element bereits in der Projektbibliothek vorhanden ist, dann wird gegebenenfalls nur die fehlende Version in der Projektbibliothek ergänzt.

Die Verwendung ist schließlich mit der Kopie des Typen in der Projektbibliothek verbunden. Einem Gerät können Sie, unabhängig von der Version, einen Typen nur einmal zuweisen.

Voraussetzung

- Im Projekt befindet sich bereits ein Gerät, das die Art des Typen unterstützt.
- Dem Gerät ist noch keine weitere Instanz desselben Typen zugeordnet.

Vorgehen

Um die Version eines Typen im Projekt zu verwenden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der globalen Bibliothek die Versionen, von denen Sie eine Verwendung erzeugen möchten.
2. Ziehen Sie die gewünschten Versionen der Typen per Drag & Drop an die Verwendungsstelle.
   Alternativ: Um automatisch die neueste Version zu verwenden, ziehen Sie die Typen selbst per Drag & Drop aus der Bibliothek an die Verwendungsstelle.
   Ziehen Sie z. B. den Typen eines Funktionsbausteins per Drag & Drop in den Bausteinordner einer CPU in der Projektnavigation. Um den Typen z. B. direkt aus einem anderen Baustein heraus aufzurufen, ziehen Sie den Typen per Drag & Drop aus der Bibliothek an die Verwendungsstelle im Programmiereditor.

Ergebnis

Fehlende Typen oder einzelne Versionen werden in der Projektbibliothek ergänzt. Wenn ein Typ noch nicht in der Projektbibliothek vorhanden ist, dann wird er im gleichen Ordner wie zuvor in der globalen Bibliothek abgelegt. Von den Typen und ihren abhängigen Elementen wird eine Verwendung erzeugt und an der Verwendungsstelle eingefügt. Die Verwendungen sind an die jeweilige Version des Typen in der Projektbibliothek gebunden.

Wenn Sie die Verwendungen in einem Editor erzeugt haben, dann werden von den Typen auch an den entsprechenden Stellen in der Projektnavigation Instanzen erzeugt. Die Ordnerstruktur aus der Bibliothek wird in der Projektnavigation abgebildet. Sie finden daher die Verwendungen in den gleichen Ordnern wie in der globalen Bibliothek wieder.

Siehe auch

Grundlagen zu Typen (Seite 57)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 85)

Grundlagen zu Bibliotheken (Seite 7)

Elementeansicht verwenden (Seite 16)

8.6.3 Projekt auf die neuesten Typ-Versionen aktualisieren

Vor allem in großen Betrieben mit vielen Automatisierungsprojekten werden globale Bibliotheken häufig zentral bearbeitet. Nach der Fertigstellung einer neueren Version werden die aktualisierten globalen Bibliotheken den einzelnen Projekten zur Verfügung gestellt. Wenn Sie eine neuere Version einer globalen Bibliothek bekommen haben, ersetzen Sie veraltete Instanzen in Ihrem Projekt durch die neueste Version. Wenn Sie die Änderungen nicht auf das gesamte Projekt anwenden möchten, beschränken Sie die Aktualisierung auf einzelne Geräte im Projekt.

Während der Aktualisierung des Projekts oder einzelner Geräte wird auch die Projektbibliothek mit den neuen Versionen der Typen in der globalen Bibliothek aktualisiert.

Die folgenden Elemente sind als Quelle für die Aktualisierung wählbar:

- Der Wurzelknoten von globalen Bibliotheken
- Einzelne Ordner innerhalb einer globalen Bibliothek
- Einzelne Typen
  Die Auswahl mehrerer Typen ist zulässig.

Hinweis

Löschen nicht verwendeter Typ-Instanzen aus dem Projekt

Standardmäßig werden beim Aktualisieren des Projekts auf die neuesten Versionen nicht genutzte Typ-Instanzen aus dem Projekt gelöscht.

Um das Löschen nicht genutzter Typ-Instanzen aus dem Projekt zu verhindern, aktivieren Sie in den Einstellungen unter "Allgemein > Bibliothekseinstellungen" das Optionskästchen "Löschen von nicht im Projekt verwendeten Typ-Instanzen deaktivieren".

Voraussetzung

- Sie befinden sich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
- Die aktualisierte globale Bibliothek ist geöffnet.

Vorgehen

Um Instanzen in einem Projekt mit den Inhalten aus einer globalen Bibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die aktualisierte globale Bibliothek oder einzelne Elemente daraus.
2. Klicken Sie mit der rechten Maustauste auf die globale Bibliothek oder die gewünschten Elemente und wählen Sie im Kontextmenü den Befehl "Typen aktualisieren > Projekt".
   Der Dialog "Typen im Projekt aktualisieren" wird geöffnet.
3. Wählen Sie entweder das gesamte Projekt oder einzelne Geräte für die Aktualisierung aus.

4. Wählen Sie Optionen für den Aktualisierungsvorgang:
   – Das Optionskästchen "Instanzen im Projekt aktualisieren" ist bei diesem Vorgang immer aktiv.
   – Aktivieren Sie das Optionskästchen "Nicht verwendete Typ-Versionen aus der Bibliothek löschen", um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen.
5. Bestätigen Sie mit "OK".
   Die Aktualisierung wird durchgeführt.

Ergebnis

Am Projekt wurden folgende Änderungen durchgeführt:

- Die neueste Version der gewählten Typen befindet sich in der Projektbibliothek. Gegebenenfalls wurden alle älteren Versionen gelöscht.
- Alle Instanzen innerhalb der gewählten Geräte wurden auf die neueste Version des verknüpften Typen aktualisiert.
- In der Projektnavigation unter "Gemeinsame Daten" finden Sie ein Protokoll des Aktualisierungsvorgangs.

Siehe auch

Projekt auf die neuesten Versionen aktualisieren (Seite 79)

Bibliothek mit den Typen einer anderen Bibliothek aktualisieren (Seite 93)

Grundlagen zu Typen (Seite 57)

Grundlagen zu Bibliotheken (Seite 7)

Typen zu einer globalen Bibliothek hinzufügen (Seite 82)

8.7 Version zuweisen

Eine Bibliothek ist übersichtlicher, wenn inhaltlich zusammengehörige Typen dieselbe Versionsnummer besitzen. Die identische Versionsnummer spiegelt den Arbeitsfortschritt wider. Wenn Sie die Arbeit an mehreren zusammengehörigen Typen abgeschlossen haben, können Sie diesen Typen dieselbe Versionsnummer zuweisen.

Um Typen eine gemeinsame Version zuzuweisen, stehen Ihnen folgende Auswahlmöglichkeiten zur Verfügung:

- Einen oder mehrere Ordner in einer Bibliothek
- Einen oder mehrere Typen

Voraussetzung

- Die Task Card "Bibliotheken" oder die Bibliotheksansicht geöffnet.
- Innerhalb Ihrer Auswahl dürfen sich keine Typen im Zustand "Im Test" oder "In Bearbeitung" befinden.

Vorgehen

Um mehreren Typen dieselbe Version zuzuweisen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Typen, denen Sie eine gemeinsame Version zuweisen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Version zuweisen".
   Der Dialog "Version zuweisen" wird geöffnet.
3. Ändern Sie bei Bedarf die Eigenschaften der Version:
   - Legen Sie im Feld "Version" die neue Versionsnummer fest. Die Versionsnummer muss höher sein als die höchste Versionsnummer aller ausgewählten Typen.
   - Geben Sie in das Feld "Autor" den Verantwortlichen für die freizugebende Version ein.
   - Geben Sie in das Feld "Kommentar" einen Kommentar zu der freizugebenden Version ein.
4. Bestätigen Sie mit "OK".

Ergebnis

Die ausgewählten Typ-Versionen werden wie folgt geändert:

- Eine neue Version aller gewählten Typen wird mit der festgelegten Versionsnummer erstellt.
- Die Eigenschaften werden für alle gewählten Typen, die neue Version und für alle zukünftig folgenden Versionen übernommen. Niedrigere Versionen bleiben von den Änderungen unberührt. Wenn Sie keine Änderungen an den Eigenschaften durchführen, werden die Eigenschaften der letzten, freigegebenen Version jedes Typen übernommen.
- Die Build-Nummer von abhängigen Typen wird auf die nächste freie Build-Nummer erhöht, sofern die abhängigen Typen nicht in Ihrer Auswahl enthalten waren. Wenn Sie einen abhängigen Typen ebenfalls ausgewählt hatten, wird die von Ihnen festgelegte Versionsnummer zugewiesen.

Ein Protokoll der Änderungen wird angelegt. Wenn Sie Typen in der Projektbibliothek versioniert haben, finden Sie das Protokoll in der Projektnavigation unter "Gemeinsame Daten > Protokolle". Wenn Sie Typen in einer globalen Bibliothek versioniert haben, finden Sie das Protokoll der globalen Bibliothek unterlagert im Ordner "Gemeinsame Daten > Protokolle".

Siehe auch

Grundlagen zu Typen (Seite 57)

Zustand von Typ-Versionen (Seite 60)

Grundlagen zu Bibliotheken (Seite 7)

Versionen freigeben (Seite 76)

Typen zu einer globalen Bibliothek hinzufügen (Seite 82)

Protokolle zu globalen Bibliotheken anzeigen (Seite 38)

Multiuser Engineering in Bibliotheken 9

9.1 Multiuser Engineering mit Bibliothekselementen

Bibliothekselemente in der Projektbibliothek können Sie im Rahmen von Multiuser Engineering verwenden.

Weitere Informationen zum Arbeiten mit Multiuser Engineering finden Sie im Kapitel "AUTOHOTSPOT".

Markierungen für das Einchecken

Die einzelnen Bibliothekselemente innerhalb der Projektbibliothek werden auf dieselbe Weise wie in der Projektnavigation für das Multiuser Engineering markiert. Markierungen von Bibliothekselementen können Sie in folgenden Bereichen des TIA Portals vornehmen:

- Task Card "Bibliotheken"
- Bibliotheksansicht
- Bibliotheksverwaltung
- An der jeweiligen Verwendungsstelle, z. B. an Typ-Instanzen

Markieren Sie Bibliothekselemente immer manuell.

Das folgende Bild zeigt die Task Card "Bibliotheken" mit der Markierungsspalte und markierten Bibliothekselementen für die Multiuser-Funktion:

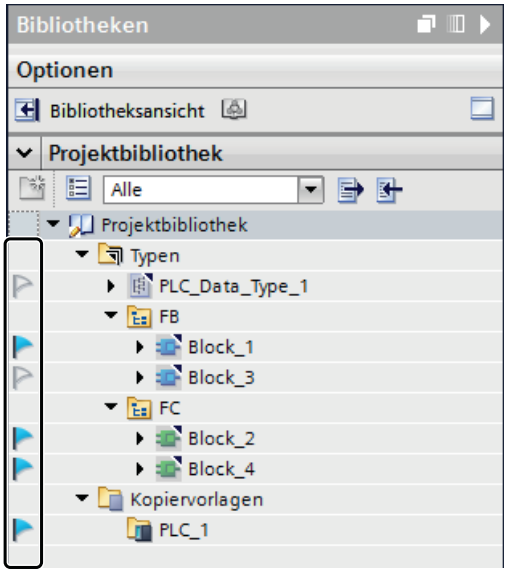

Markierbare Bibliothekselemente

Die folgende Tabelle zeigt, welche Typen in der Projektbibliothek und in der Instanz im Projekt für das Einchecken markierbar sind:

	Typ in der Bibliothek markierbar	Instanz markierbar
Funktionsbausteine (FB)	ja	ja
PLC Block Library Type (FC)	ja	ja
PLC-Datentypen (UDT)	ja	ja
HMI Bildbausteine	ja	Durch Markieren des Bildes, in dem die Instanz des Bildbausteins liegt
HMI Anwenderdatentyp (UDT)	ja	Durch Markieren des Objekts (z. B. Bild oder HMI Variable), in dem der UDT verwendet ist.
HMI Stil	ja	nein
HMI Formatvorlage	ja	nicht verfügbar
HMI Bild	ja	ja
HMI C-Skript	ja	ja
HMI VB-Skript	ja	ja

Löschen von Typen in der lokalen Session

Wenn Sie Typen in der lokalen Session löschen möchten, beachten Sie Folgendes:

- Geben Sie den Typ frei, falls sich noch eine Version im Test oder in Bearbeitung befindet.
- Markieren Sie den Typ in der lokalen Session vor dem Löschen, um den Typ im zugeordneten Serverprojekt zu löschen.

Einzelne Versionen können Sie in der lokalen Session zwar löschen, diese bleiben jedoch im Serverprojekt erhalten. Da es sich beim Löschen um eine administrative Tätigkeit handelt, löschen Sie die· Typen und ihre Versionen im Server-Projekt.

Projekt aktualisieren

Wenn Sie das Projekt aktualisieren möchten, beachten Sie Folgendes:

- Markieren Sie die Typen, bevor Sie das Projekt aktualisieren.
- Markieren Sie HMI-Anwenderdatentypen an der Verwendungsstelle, bevor Sie das Projekt aktualisieren.
- Änderungen an HMI Stilen werden in der lokalen Session nicht übernommen, wenn Sie das Projekt aktualisieren. Aktualisieren Sie HMI Stile in der Serverprojekt-Ansicht.

9.2 Einchecken von Typen und Kopiervorlagen

Änderungen an Typen und Kopiervorlagen checken Sie ebenso in das Multiuser-Serverprojekt ein wie alle anderen Objekte innerhalb der lokalen Session. Wenn Sie einen Typen einchecken, dann werden immer alle Versionen des Typen eingecheckt. Wenn Sie eine Instanz einchecken, dann wird auch der Typ automatisch eingecheckt.

Aktualisieren des Serverprojekts

Typen und Kopiervorlagen werden anhand der eindeutigen GUID im Serverprojekt identifiziert und abgeglichen. Beim Einchecken von Typen und Kopiervorlagen werden Ihre Änderungen aus der lokalen Session in das Serverprojekt übernommen:

- Die Eigenschaften des Typen oder der Kopiervorlage, z. B. der Name, werden im Serverprojekt übernommen.
- Fehlende Versionen von Typen werden ergänzt.
- Die Ordnerstruktur innerhalb der Bibliothek wird im Serverprojekt übernommen. Leere Ordner werden jedoch nicht gelöscht.

Durch das Aktualisieren des Serverprojekts können Konflikte entstehen. Eventuelle Konflikte werden folgendermaßen gelöst:

- Namenskonflikte
  Zwei Typen mit identischem Namen können eingecheckt werden. Dabei werden die Namen der betroffenen Typen automatisch mit einem Zusatz versehen (<_1> etc.).
  Zwei Typ-Instanzen mit identischem Namen sind nicht zulässig und führen zu einem Fehler beim Einchecken.
- Ein Typ ist im Zustand "im Test" oder "in Bearbeitung".
  Wenn ein Typ im Serverprojekt oder in der lokalen Session sich in einem der beiden Zustände befindet, dann können Sie das Serverprojekt nicht aktualisieren. Geben Sie alle markierten Typen zuerst frei, bevor Sie das Serverprojekt aktualisieren.
- Versionskonflikte
  Wenn es sowohl im Serverprojekt als auch in der lokalen Session Versionen mit identischer Versionsnummer gibt, die Versionen sich jedoch inhaltlich unterscheiden, werden Sie mit einer Meldung benachrichtigt. Ändern Sie die Versionsnummer, so dass diese nur einmal existiert.

Weitere Informationen zum Multiuser Engineering finden Sie im Kapitel "AUTOHOTSPOT".

Typen einer Bibliothek aktualisieren und hochrüsten 10

10.1 Bibliothek mit den Typen einer anderen Bibliothek aktualisieren

Eine bereits vorhandene Bibliothek lässt sich mit den Inhalten aus einer anderen Bibliothek aktualisieren. Folgende Möglichkeiten zur Aktualisierung von Bibliotheken bestehen:

- Aktualisieren einer globalen Bibliothek mit Typen aus einer anderen globalen Bibliothek oder der Projektbibliothek
- Aktualisieren der Projektbibliothek mit Typen aus einer globalen Bibliothek

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Wurzelknoten der Projektbibliothek
- Wurzelknoten von globalen Bibliotheken
- Ordner "Typen"
- Einzelne Ordner innerhalb des Ordners "Typen"
- Einzelne Typen
  Die Auswahl mehrerer Typen ist zulässig.

Bei der Aktualisierung werden bereits in der Zielbibliothek vorhandene Typen um die neuesten Versionen ergänzt. Typen, die noch nicht in der Zielbibliothek vorhanden sind, werden mit allen ihren Versionen in die Zielbibliothek kopiert.

Hinweis

Anwenderdefinierte Dokumentation beim Kopieren von Typen

Anwenderdefinierte Dokumentation wird nicht mit einem Typ zusammen in eine andere Bibliothek kopiert. Kopieren Sie die anwenderdefinierte Dokumentation zu Typen selbst in das passende Verzeichnis.

Weitere Hilfe zur Verwendung der anwenderdefinierten Dokumentation erhalten Sie im Kapitel "AUTOHOTSPOT".

Voraussetzung

Wenn Sie eine globale Bibliothek aktualisieren möchten, dann muss diese mit Schreibrechten geöffnet sein.

Vorgehen

Um eine Bibliothek mit Inhalten einer anderen Bibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Bibliothek den Ordner "Typen" oder einzelne Elemente daraus als Quelle für die Aktualisierung.
2. Klicken Sie mit der rechten Maustauste auf die Quelle und wählen Sie im Kontextmenü den Befehl "Typen aktualisieren > Bibliothek".
   Der Dialog "Typen in einer Bibliothek aktualisieren" wird geöffnet.
3. Wählen Sie aus, welche Art von Bibliothek Sie aktualisieren möchten:
   – Wählen Sie "Projektbibliothek aktualisieren", um die Projektbibliothek mit Typen aus einer globalen Bibliothek zu aktualisieren.
   – Wählen Sie "Globale Bibliothek aktualisieren", wenn Sie eine globale Bibliothek aktualisieren möchten.
4. Optional: Wählen Sie aus der Klappliste die globale Bibliothek, die Sie aktualisieren möchten, wenn Sie eine globale Bibliothek aktualisieren möchten.
5. Wählen Sie die Optionen für die Aktualisierung:
   – Die Option "Instanzen im Projekt aktualisieren" ist bei diesem Vorgang immer deaktiviert.
   – Aktivieren Sie das Optionskästchen "Nicht verwendete Typ-Versionen aus der Bibliothek löschen", wenn Sie alle Versionen von Typen aus der Bibliothek löschen möchten, die mit keiner Instanz im Projekt verbunden sind und die keine Abhängigkeiten zu anderen Typen besitzen. Die Option ist bei der Aktualisierung einer globalen Bibliothek nicht wählbar, da Typen einer globalen Bibliothek niemals eine Verwendungsstelle im Projekt besitzen.
6. Bestätigen Sie mit "OK".
   Die Aktualisierung wird durchgeführt.

Ergebnis

An der Zielbibliothek wurden folgende Änderungen durchgeführt:

- Typen, die in der Zielbibliothek noch nicht vorhanden waren, wurden mit allen ihren Typ-Versionen ergänzt. Typen, die bereits in der Zielbibliothek vorhanden waren, wurden um die neueren Typ-Versionen ergänzt. Wenn in der Zielbibliothek bereits eine neuere Version eines Typs vorhanden war, wurde die neueste Version aus der Quellbibliothek trotzdem kopiert und automatisch eine neuere Versionsnummer vergeben.
- Gegebenenfalls wurden alle Versionen von Typen aus der Projektbibliothek gelöscht, wenn diese in keiner Instanz im Projekt verwendet waren.
- Für den Aktualisierungsvorgang wurde ein Protokoll angelegt, das alle durchgeführten Änderungen an der Zielbibliothek enthält.
  Wenn Sie die Projektbibliothek aktualisiert haben, finden Sie das Protokoll in der Projektnavigation unter "Gemeinsame Daten > Protokolle".
  Wenn Sie eine globale Bibliothek aktualisiert haben, finden Sie das Protokoll, der globalen Bibliothek unterlagert, im Ordner "Gemeinsame Daten > Protokolle".

Siehe auch

Projekt auf die neuesten Versionen aktualisieren (Seite 79)
Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 85)
Protokolle zu globalen Bibliotheken anzeigen (Seite 38)
Grundlagen zu Bibliotheken (Seite 7)

10.2 Typen in einem bestehenden Projekt hochrüsten

Mithilfe der Funktion "Hochrüsten von Typen" können Sie die Abhängigkeiten von Typen auf die neuesten vorhandenen Typ-Versionen in einem Schritt im gesamten Projekt hochrüsten.

Wenn Sie innerhalb Ihrer Projektbibliothek einen Typ, der einen anderen Typ referenziert, bearbeiten und eine neue Typ-Version erstellen, dann wird der referenzierte Typ automatisch in den Zustand "in Test" oder "in Bearbeitung" gesetzt. Damit wird gewährleistet, dass alle Typen innerhalb des Projekts aktuell sind.

Wenn Sie einen Typ außerhalb des Projekts bearbeitet und eine neue Typ-Version erstellt haben, dann werden die referenzierten Typen nicht automatisch aktualisiert. Damit Sie diese Abhängigkeiten zu den einzelnen referenzierten Typen nicht manuell hochrüsten müssen, steht Ihnen die Funktion "Hochrüsten von Typen" in der Bibliotheksverwaltung zur Verfügung.

Mithilfe dieser Funktion identifizieren Sie alle referenzierten Typen, die von dem geänderten Typ abhängig sind und rüsten diese Typen automatisch hoch. Alle hochgerüsteten Typen werden in der Projektbibliothek mit dem Zusatz "im Test" oder "in Bearbeitung" versehen. Nach dem Hochrüsten sehen Sie dadurch auf einen Blick, welche Typen ebenfalls von den Änderungen betroffen sind und können diese gezielt prüfen und wieder freigeben.

Voraussetzung

Sie haben ein Projekt erstellt, das z. B. Programmbausteine enthält, die andere Programmbausteine oder Objekte, wie z. B. PLC-Datentypen (UDTs), aufrufen. Damit bestehen Abhängigkeiten zwischen den einzelnen Objekten.

Sie haben eine Projektbibliothek erstellt und die zu bearbeitenden Typen in eine globale Bibliothek gespeichert. Die Typen wurden außerhalb Ihres Projekts weiterentwickelt und sollen jetzt wieder mit den neuen Typ-Versionen in Ihre bestehende Projektbibliothek integriert werden. Aktualisieren Sie dafür die Projektbibliothek.

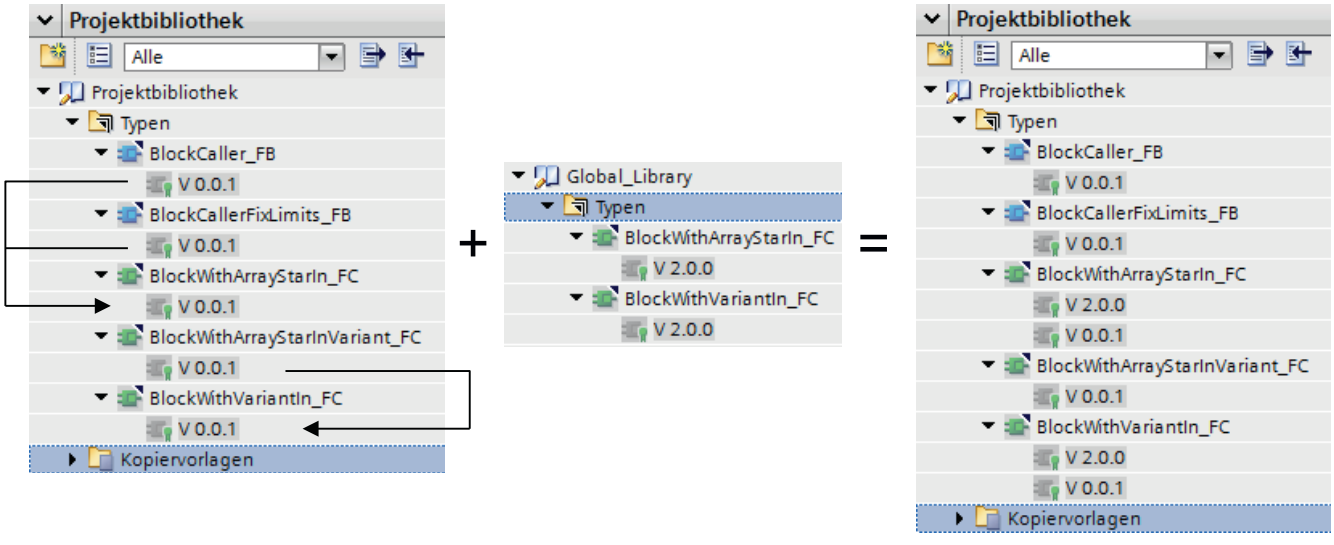

Um die neuen Typ-Versionen in die bestehende Projektbibliothek zu integrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek ohne Schreibschutz.
   Die bearbeiteten Typen habe eine höhere Typ-Version, als die Typen, die in der Projektbibliothek enthalten sind.
2. Klicken Sie in der globalen Bibliothek den Ordner "Typen" an und wählen Sie aus dem Kontextmenü "Typen aktualisieren > Projekt" aus.
   Die Projektbibliothek wird aktualisiert und die neuen Typ-Versionen sind enthalten.

Weitere Informationen zur Aktualisierung von Bibliotheken finden Sie hier: Bibliothek mit den Typen einer anderen Bibliothek aktualisieren (Seite 93)

Zu diesem Zeitpunkt sind die neuen Typ-Versionen in der Projektbibliothek enthalten, aber noch nicht im Projekt referenziert.

Vorgehen

Um die neuen Typ-Versionen automatisiert im Projekt zu referenzieren, müssen diese in der Bibliotheksverwaltung hochgerüstet werden.

Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektbibliothek den Ordner "Typen" an und öffnen Sie die Bibliotheksverwaltung.
2. Wählen Sie im Bereich "Typen" aus der Klappliste den Eintrag "Typen mit neuen Versionen für das Hochrüsten anderer Typen" aus.
   Es werden alle Typen angezeigt, für die eine neue Typ-Version in der Projektbibliothek vorliegt.
3. Klicken Sie den Ordner "Typen" mit einem rechten Mausklick an und wählen Sie aus dem Kontextmenü den Eintrag "Hochrüstbare Typen anzeigen" aus.
   Im Register "Hochrüsten von Typen" werden Ihnen alle referenzierten Typen angezeigt.
4. Wählen Sie unter "Hochzurüstende Typen" einen der beiden Einträge aus:
   - Nur mit Testumgebung
     Es werden alle Typen angezeigt, die eine Instanz in der Testumgebung referenzieren.
   - Kompatibel mit gewählter Testumgebung
     Es werden alle Typen angezeigt, die eine Instanz in der Testumgebung referenzieren können. Die Instanz wurde aber zu diesem Zeitpunkt noch nicht erstellt.
5. Wählen Sie eine Testumgebung aus.
   Mit der Auswahl einer Testumgebung legen Sie fest, in welcher Testumgebung die Typen hochgerüstet werden sollen.
6. Jetzt werden Ihnen alle Typen angezeigt, die hochgerüstet werden müssen.

Hinweis

Anzeige der Typen

Die Anzeige der Typen hängt davon ab, welchen Ordner Sie in der Bibliotheksverwaltung und welche Einstellungen Sie im Register "Hochrüsten von Typen" gewählt haben.

7. Klicken Sie auf "Starten".
   Nach erfolgreichem Hochrüsten wird der Dialog "Hochrüsten von Typen" geöffnet.

   Hinweis

   Status "Fehler"

   Wenn einer der hochzurüstenden Typen den Status "Fehler" hat, dann können Sie das Hochrüsten der Typen nicht starten. Sie haben zwei Möglichkeiten:

   1. Beheben Sie den Fehler und starten Sie das Hochrüsten erneut.
   2. Aktualisieren Sie die Ansicht der möglichen hochzurüstenden Typen, indem Sie unter "Hochzurüstende Typen" erneut den gewünschten Eintrag aus der Klappliste auswählen.

   Alle referenzierten Typen befinden sich jetzt in der Projektnavigation und in der Projektbibliothek im Zustand "im Test".

8. Prüfen Sie die Typen "im Test" und geben Sie sie anschließend frei. Dazu können Sie ebenfalls die Bibliotheksverwaltung nutzen.
9. Wählen Sie dafür aus der Klappliste den Eintrag "Typen mit ausstehenden Änderungen" aus und geben Sie alle angezeigten Typen frei.

Ergebnis

Alle Typen innerhalb Ihres Projekts referenzieren jetzt die aktuelle Typ-Version.

10.3 Beispiel für das Aktualisieren und Hochrüsten von Typen

Beschreibung

Sie haben eine Projektbibliothek, deren Typen unterschiedlich miteinander referenziert sind.

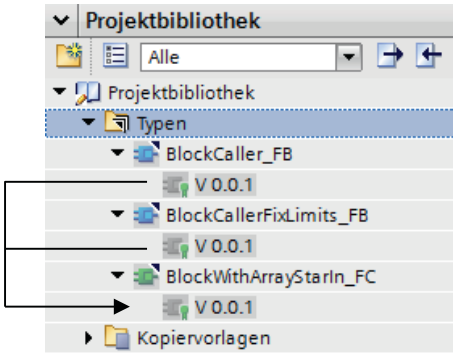

Die Funktion "BlockWithArrayStarIn_FC" wird jeweils von den beiden Funktionsbausteinen aufgerufen.

Die Funktion "BlockWithArrayStarIn_FC" soll unabhängig außerhalb der Projektbibliothek weiter entwickelt werden.

Voraussetzung

Sie haben die Projektbibliothek mit allen relevanten Typen erstellt.

Vorgehen

Um den Typ "BlockWithArrayStarIn_FC" unabhängig außerhalb des Projekts weiter zu entwickeln, gehen Sie folgendermaßen vor:

1. Kopieren Sie den Typ aus der Projektbibliothek in eine globale Bibliothek, um ihn übergeben zu können.

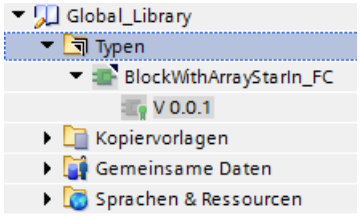

   Sie übergeben die Version V0.0.1 zur Weiterentwicklung.

2. Der Typ "BlockWithArrayStarIn_FC" wurde bearbeitet und freigegeben. Es wurde die neue Version V2.0.0 erstellt und die globale Bibliothek mit der neuen Typ-Version zurückgeliefert:

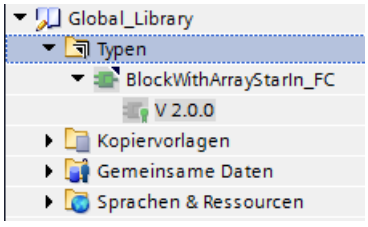

Um die neue Version der Funktion in Ihre bestehende Projektbibliothek zu integrieren, gehen Sie folgendermaßen vor:

1. Klicken Sie den Ordner "Typen" der globalen Bibliothek "Global_Library" an.
2. Wählen Sie aus dem Kontextmenü den Eintrag "Typen aktualisieren > Projekt".
   Der Dialog "Typen im Projekt aktualisieren" wird geöffnet.
3. Klicken Sie auf "OK".
   Die neue Typ-Version V2.0.0 wurden in die Projektbibliothek integriert.

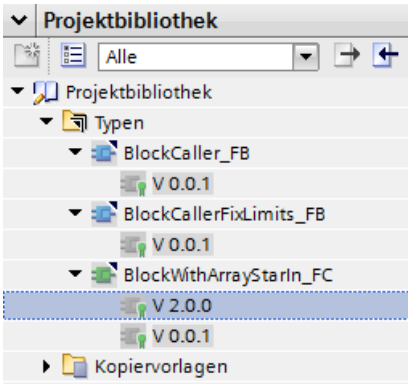

Hinweis

Aktuelle Typ-Version referenzieren

Zu diesem Zeitpunkt steht die neue Typ-Version nur in der Projektbibliothek zur Verfügung. Die Instanzen in der Projektnavigation referenzieren noch auf die veraltete Typ-Version V0.0.1.

Um die Instanzen in der Projektnavigation mit einem Schritt auf die neue Typ-Version hochzurüsten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektbibliothek auf den Ordner "Typen" und öffnen Sie die Bibliotheksverwaltung.
2. Wählen Sie im Bereich "Typen" aus der Klappliste den Eintrag "Typen mit neuen Versionen für das Hochrüsten anderer Typen" aus.
   Es wird die Funktion angezeigt, die eine neue Typ-Version hat, die in der Projektnavigation noch nicht verwendet wird.

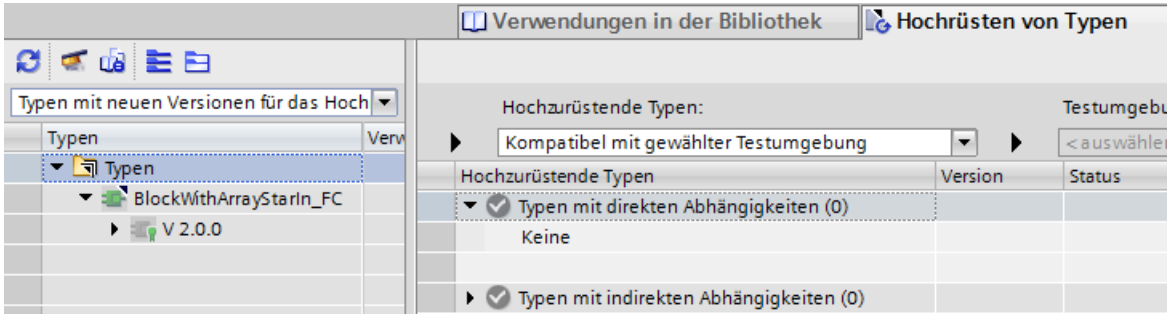

3. Klicken Sie den Ordner "Typen" an und wählen Sie aus dem Kontextmenü den Eintrag "Hochrüstbare Typen anzeigen".
   Das Register "Hochrüsten von Typen" wird geöffnet und es werden alle Typen angezeigt, die eine direkte Abhängigkeit zu dem geänderten Typ haben.

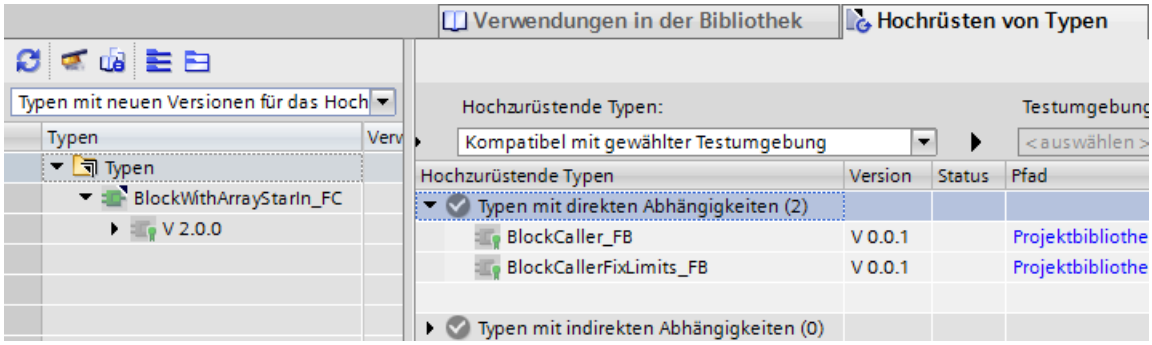

4. Um die hochzurüstenden Typen nach einer Testumgebung zu filtern, wählen Sie unter "Hochzurüstende Typen" den Eintrag "Nur mit Testumgebung" aus.

5. Wählen Sie die Testumgebung "PLC S7-1500" aus.

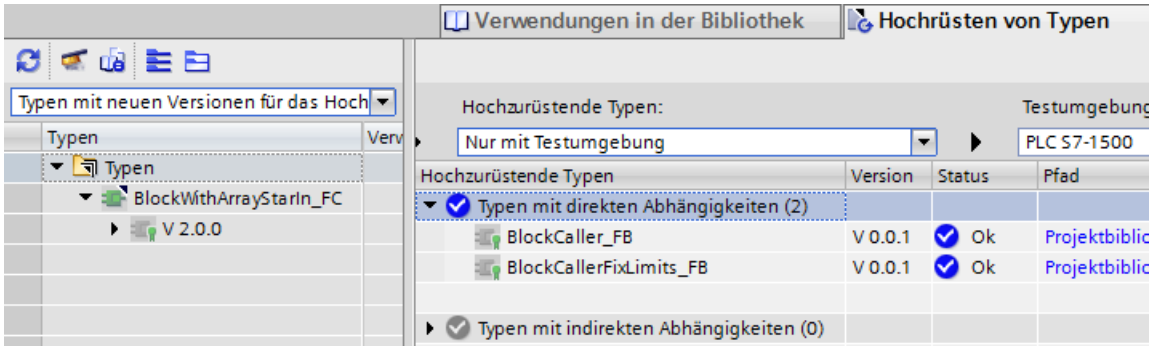

Alle Typen, die hochgerüstet werden können, haben den Status "Ok".
Wenn einer oder mehrere der Typen den Status "Fehler" haben, dann können Sie das Hochrüsten nicht starten. Beheben Sie zunächst den Fehler und wiederholen Sie dann die Schritte 1 - 5.

6. Klicken Sie auf "Starten".

Hinweis

Hochrüsten mit indirekt abhängigen und nicht instanzierten Typversionen

Wenn Sie einen Typen hochrüsten wollen, auf den eine oder mehrere Typversionen referenzieren, die selbst nicht im Projekt instanziert sind, beachten Sie die Auswirkungen der möglichen Optionen zum Hochrüsten in der Bibliotheksverwaltung.

Wenn Sie die Option "Nur mit Testumgebung" und eine Testumgebung eingestellt haben, ist das Hochrüsten des Typen möglich. Für die nicht instanzierten Typversionen wird eine Warnung angezeigt, diese Typversionen werden nicht hochgerüstet.

Wenn Sie die Option "Kompatibel mit gewählter Testumgebung" und eine Testumgebung einstellen, ist das Hochrüsten des Typen nicht möglich. Für die nicht instanzierten Typversionen wird eine Fehlermeldung angezeigt, diese Typversionen verhindern das Hochrüsten.

Ergebnis

Alle referenzierten Typen wurden in der Projektbibliothek und in der Projektnavigation auf eine neue Version hochgerüstet und mit dem Zusatz "im Test" versehen.

Prüfen Sie die neuen Versionen und geben Sie sie anschließend frei.

Bibliothekstexte verwenden 11

11.1 Bibliothekstexte exportieren

Beschreibung

Sie können die Texte der Bibliothekselemente zur Übersetzung exportieren und anschließend wieder importieren. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann in Microsoft Excel oder in vielen anderen Tabellenkalkulationsprogrammen bearbeitet werden.

Sie haben folgende Möglichkeiten für den Export:

- Texte ausgewählter Typen und Kopiervorlagen exportieren
  Sie können einzelne Typen oder Kopiervorlagen auswählen und mit Hilfe des Kontextmenüs ausschließlich die Texte des gewählten Objektes exportieren.
  Oder Sie können durch mehrfach Selektion mehrere oder alle Typen bzw. Kopiervorlagen auswählen und exportieren.
- Texte aller Bibliothekselemente der Projektbibliothek oder einer globalen Bibliothek exportieren
  Mit Hilfe der Schaltflächen in der Funktionsleiste der Paletten "Projektbibliothek" bzw. "Globale Bibliotheken" können Sie alle Texte einer Projektbibliothek bzw. einer globalen Bibliothek exportieren.

Wenn es von einem Typ mehrere Versionen gibt, dann werden die Texte der aktuellsten Version exportiert.

Hinweis

Kein Export von Bibliothekstexten

Wenn die aktuellste Version eines Typs noch nicht freigegeben ist und den Zusatz "im Test" oder "in Bearbeitung" hat, dann werden keine Bibliothekstexte exportiert.

Hinweis

Zeilenbeschränkung bei Microsoft Excel

Beachten Sie, dass Tabellenkalkulationsprogramme möglicherweise nur eine bestimmte Anzahl an Zeilen verarbeiten können. Microsoft Excel 2003 unterstützt beispielsweise maximal 65536 Zeilen. Neuere Versionen von Microsoft Excel unterstützen mehr Zeilen.

Quell- und Zielsprachen bei einer Projektbibliothek definieren

Bei der Projektbibliothek hängt die Auswahl der möglichen Quell- und Zielsprachen davon ab, welche Sprachen Sie als Projektsprachen definiert haben.

Weitere Informationen finden Sie hier: AUTOHOTSPOT

Quell- und Zielsprachen bei globalen Bibliotheken definieren

Bei einer globalen Bibliothek können Sie die Bibliothekssprachen selbst definieren, wenn die Bibliothek nicht schreibgeschützt ist.

Die Auswahl der möglichen Sprachen können Sie sich anzeigen lassen, indem Sie im Ordner "Sprachen & Ressourcen" den Unterordner "Bibliothekssprachen" doppelklicken. Bei einer schreibgeschützten Bibliothek werden die Bibliothekssprachen in der Bibliotheksansicht im Lesemodus angezeigt.

Weitere Informationen finden Sie hier: Bibliothekssprachen definieren (Seite 36)

Voraussetzung

Aktivieren Sie die gewünschten Zielsprachen:

- Für eine Projektbibliothek in den Projektsprachen
- Für eine globale Bibliothek in den Bibliothekssprachen

Texte von Bibliothekselementen exportieren

Um die Texte der ausgewählten Bibliothekselemente zu exportieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschten Bibliothekselemente (einzelnes Bibliothekselement, mehrere Bibliothekselemente, einen Ordner oder eine ganze Bibliothek).
2. Klicken Sie entweder mit der rechten Maustaste und wählen Sie aus dem Kontextmenü "Bibliothekstexte exportieren" oder klicken Sie in der Funktionsleiste auf die Schaltfläche "Bibliothekstexte exportieren".
   Der Dialog "Exportieren" wird geöffnet.

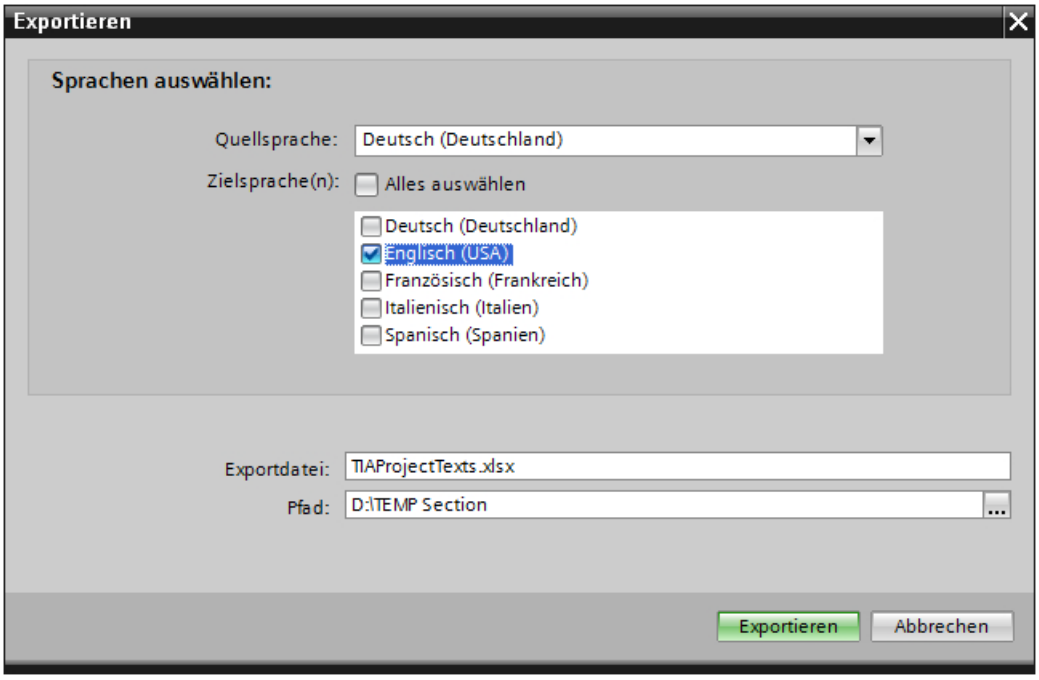

3. Wählen Sie die gewünschte Quellsprache aus.

4. Wählen Sie die gewünschten Zielsprachen aus.
   Wenn Sie alle Zielsprachen exportieren wollen, wählen Sie die Option "Alles auswählen".
5. Geben Sie der Exportdatei den gewünschten Namen.
6. Stellen Sie den gewünschten Pfad zum Ablageort der Exportdatei ein.
7. Klicken Sie auf "Exportieren".

Ergebnis

Der Export wurde erfolgreich abgeschlossen und die ".xlsx"-Datei am Ablageort erstellt.

A	B	C	D	E	F
Category	ViewPath	Internal ID	Substitutions	de-DE*	en-US
<BlockCommentCategoryData>	Programmbaust	a5exBpTAQ6Af2xKwOQN99		Start Motor	
<BlockCommentCategoryData>	Programmbaust	a5exBpTAQ6Af2xKwOQN99		Stop Motor	

Schreiben Sie die englischen Übersetzungen in die Spalte "en-US", um sie anschließend importieren zu können.

11.2 Bibliothekstexte importieren

Beschreibung

Nach der externen Übersetzung in einem Tabellenkalkulationsprogramm importieren Sie die Texte in das TIA Portal. Die Texte können Sie in der Projektbibliothek oder einer globalen Bibliothek an folgenden Stellen importieren:

- Mithilfe des Kontextmenüeintrags "Bibliothekstexte importieren"
- Mithilfe der Schaltfläche "Bibliothekstexte importieren"

Unabhängig davon, wie viele Bibliothekselemente Sie selektiert haben, werden alle Texte aus der Importdatei für die gesamte Bibliothek importiert. Die Bibliothekselemente aus der Importdatei müssen mit den Bibliothekselementen der Bibliothek übereinstimmen, in die die Texte importiert werden sollen. Die Zielsprachen, die Sie importieren wollen, müssen im Projekt bzw. in der globalen Bibliothek aktiviert sein.

Wenn Sie Texte für Kopiervorlagen importieren, dann werden die bereits vorhandenen Texte mit den neuen Texte überschrieben. Wenn Sie Texte für Typen importieren, dann werden nur die Texte der aktuellsten Version mit den neuen Texten überschrieben.

Hinweis

Kein Import von Bibliothekstexten möglich

Es ist kein Import von Bibliothekstexten möglich, wenn eine der folgenden Möglichkeiten zutrifft:

- Die Bibliothekstexte gehören zu einer Typ-Instanz, die in einer Kopiervorlage enthalten ist.
- Innerhalb der Projektbibliothek ist eine Version eines Typs noch nicht freigegeben und hat den Zusatz "im Test" oder "in Bearbeitung".
- Die globale Bibliothek ist schreibgeschützt.
- Es handelt sich um eine Systembibliothek.

Hinweis

Bibliothekstexte für ein Multiuser Projekt importieren

Damit auch die Bibliothekstexte von nicht markierten Objekten importiert werden, müssen in einem Multiuser Projekt die Bibliothekstexte in der Serverprojekt-Ansicht importiert werden.

Voraussetzung

Sie haben die Übersetzungen in der gewünschten Zielsprache in die Exportdatei eingetragen.

	A	B	C	D	E	F
1	Category	ViewPath	Internal ID	Substitutions	de-DE*	en-US
2	<BlockCommentCategoryData>	Programmbaust	a5exBpTAQ6Af2xKwOQN99	Start Motor		Start the motor
3	<BlockCommentCategoryData>	Programmbaust	a5exBpTAQ6Af2xKwOQN99	Stop Motor		Stop the motor

Wenn Sie Bibliothekstexte in eine Projektbibliothek importieren wollen, dann müssen die gewünschten Zielsprachen in den Projektsprachen aktiviert sein.

Wenn Sie Bibliothekstexte in eine globale Bibliothek importieren wollen, dann müssen die gewünschten Zielsprachen in der globalen Bibliothek aktiviert sein.

Alle Texte einer Projektbibliothek oder einer globalen Bibliothek importieren

Um alle übersetzten Bibliothekstexte zu importieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Projektbibliothek oder eine globale Bibliothek.
2. Klicken Sie entweder mit der rechten Maustaste und wählen Sie aus dem Kontextmenü "Bibliothekstexte importieren" oder klicken Sie in der Funktionsleiste auf die Schaltfläche "Bibliothekstexte importieren".
   Der Dialog "Importieren" wird geöffnet.
3. Wählen Sie die zu importierende Datei aus.
4. Sie haben die Möglichkeit, die Quellsprache ebenfalls zu importieren, indem Sie die Option "Quellsprache importieren" aktivieren.
5. Klicken Sie auf "Importieren".
   Nach dem Import wird der Dialog "Import abgeschlossen" geöffnet.

Ergebnis

- Import erfolgreich abgeschlossen:
  Nach dem erfolgreichen Import enthält die aktuellste Typ-Version die übersetzten Texte.
- Import mit Warnungen abgeschlossen:
  Der Grund für den Fehler wird entweder direkt im Dialog angezeigt, oder Sie können den Link "Hier klicken, um die Protokoll-Datei einzusehen" anklicken, um sich die Fehlerursachen anzuschauen.
- Import ist fehlgeschlagen:
  Sie finden das Protokoll mit den Fehlerursachen, indem Sie den Link "Hier klicken, um die Protokoll-Datei einzusehen" anklicken.

11.3 Projektsprachen der Projektbibliothek definieren

Alle definierten Projektsprachen stehen auch in der Projektbibliothek zur Verfügung.

Sie können die Projektsprachen zentral für die gesamte Projektbibliothek definieren und ändern. Alle Änderungen an den Projektsprachen wirken sich in diesem Fall auf alle bereits bestehenden und auf alle neu angelegten Typen und Kopiervorlagen der Projektbibliothek aus.

Sie können aber auch für eine einzelne Typ-Version die Projektsprachen ändern, ohne Auswirkung auf die restliche Projektbibliothek.

Vorgehen

Um die Projektsprachen für die gesamte Projektbibliothek zu definieren, gehen Sie folgendermaßen vor:

1. Öffnen oder bearbeiten Sie einen Typ oder eine Kopiervorlage.
2. Öffnen Sie die Task Card "Aufgaben".
3. Wählen Sie unter "Sprachen & Ressourcen" entweder eine andere Projektsprache aus den bereits vorhandenen Editiersprachen aus oder klicken Sie auf "Neue Editiersprache hinzufügen".
4. Wenn Sie "Neue Editiersprache hinzufügen" anklicken, dann wird die Liste aller verfügbaren Projektsprachen angezeigt.
5. Wählen Sie die gewünschte Editiersprache aus.

Vorgehen

Um die Projektsprachen für eine einzelne Typ-Version zu definieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die entsprechende Typ-Version in der Projektbibliothek aus.
2. Wählen Sie aus der Menüleiste den Befehl "Werkzeuge > Projektsprachen". Die Liste aller verfügbaren Projektsprachen wird geöffnet.
3. Wählen Sie die gewünschte Editiersprache aus.

Ergebnis

Die Projektbibliothek bzw. die Typ-Version haben die gewünschten Projekt- bzw. Editiersprachen.

11.4 Aktualisierung von importierten Bibliothekstexten bei Typ-Instanzen

Wenn Sie Texte von Typ-Versionen exportieren und anschließend wieder in die gleiche Typ-Version importieren, werden die vorgenommenen Änderungen nicht in den Typ-Instanzen angezeigt. Aufgrund der gleichen Versionen des Typs und der Instanz, z. B. V 0.0.1, wird beim Aktualisieren des Projekts oder der Bibliothek keine Änderung der Texte erkannt.

Importierte Texte aus der Projektbibliothek im Projekt anzeigen

Um die Änderungen von importierten Texten aus der Projektbibliothek im Projekt anzuzeigen, haben Sie folgende Optionen:

- Löschen Sie die Instanz im Projekt und erstellen Sie anschließend per Drag & Drop eine neue Instanz der gleichen Version im Projekt. Dieses Vorgehen ist notwendig, wenn Sie importierte Texte in einem anderen Gerät oder Objekt im Projekt anzeigen wollen.
- Importieren Sie die geänderten Texte direkt in die Instanzen im Projekt. Nutzen Sie dazu die Funktion "Projekttexte importieren" im Menü "Werkzeuge". Dieses Vorgehen ist nur möglich, wenn Sie importierte Texte im gleichen Gerät und Objekt im Projekt anzeigen wollen.

Importierte Texte aus globalen Bibliotheken anzeigen

Änderungen von importierten Texten aus globalen Bibliotheken können Sie in der Projektbibliothek oder in einer Instanz im Projekt anzeigen. Erstellen Sie dazu eine neue Version des importierten Objekts, z. B. V 0.0.2. Zur Anzeige der importierten Texte haben Sie folgende Optionen:

- Um die importierten Texte in der Projektbibliothek anzuzeigen, aktualisieren Sie die Projektbibliothek. Rufen Sie dazu das Kontextmenü des Objekts auf und wählen Sie den Menübefehl "Typen aktualisieren > Bibliothek" aus.
- Um die importierten Texte im Projekt anzuzeigen, aktualisieren Sie das Projekt. Rufen Sie dazu das Kontextmenü des Objekts auf und wählen Sie den Menübefehl "Typen aktualisieren > Projekt" aus.

Ordner in einer Bibliothek erstellen 12

Innerhalb der Bibliotheken werden die Bibliothekselemente, abhängig von ihrer Art, in den Ordnern "Typen" und "Kopiervorlagen" gespeichert. Um Kopiervorlagen und Typen weiter zu strukturieren, legen Sie weitere Ordner unterhalb von "Typen" und "Kopiervorlagen" an.

Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.
- Wenn Sie neue Ordner innerhalb einer globalen Bibliothek erstellen möchten, dann muss die globale Bibliothek mit Schreibrechten geöffnet sein.

Vorgehen

Um einen neuen Ordner zu erzeugen, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb der Bibliothek mit der rechten Maustaste auf einen beliebigen Ordner.
2. Wählen Sie im Kontextmenü den Befehl "Ordner hinzufügen".
   Ein neuer Ordner wird erstellt.
3. Geben Sie einen Namen für den neuen Ordner ein.

Siehe auch

Kopiervorlagen filtern (Seite 54)

Arbeiten mit Typen in der Projektbibliothek (Seite 64)

Bibliothekselemente bearbeiten 13

Innerhalb der Task Card "Bibliotheken" oder der Bibliotheksansicht lassen sich Typen, Kopiervorlagen und Ordner auf gewohnte Art und Weise ausschneiden, kopieren, einfügen, verschieben, umbenennen oder löschen. Globale Bibliotheken müssen für jeden der genannten Vorgänge mit Schreibrechten geöffnet sein.

Hinweis

Anwenderdefinierte Dokumentation zu Typen und Kopiervorlagen

Anwenderdefinierte Dokumentation ist von sämtlichen Operationen innerhalb der Bibliothek nicht betroffen. Wenn Sie eine Kopiervorlage oder einen Typen an einen anderen Ort bewegen, verschieben Sie die zugehörige anwenderdefinierte Dokumentation im Dateisystem manuell an den passenden Ort.

Weitere Hilfe zur Verwendung der anwenderdefinierten Dokumentation erhalten Sie im Kapitel "AUTOHOTSPOT".

Kopieren von Typen

Wenn Sie einen Typen in die Zwischenablage kopieren, dann gelten folgende Regeln:

- Typen werden immer mit allen zugehörigen Versionen in die Zwischenablage kopiert. Es werden jedoch nur Versionen kopiert, die bereits freigegeben sind.
- Typen werden immer mit allen abhängigen Elementen in die Zwischenablage kopiert.
- Kopiervorlagen werden immer zusammen mit den darin verwendeten Typ-Versionen in die Zwischenablage kopiert.

Kopieren und einfügen von Typ-Versionen

Wenn Sie Versionen von Typen kopieren und in eine andere Bibliothek einfügen, dann müssen die Typen bereits in der Zielbibliothek vorhanden sein.

Ausschneiden von Elementen

Ausgeschnittene Bibliothekselemente können Sie nur in die gleiche Bibliothek einfügen. Dabei können Sie Kopiervorlagen nur in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen" und Typen nur in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" einfügen.

Einfügen von Typen

Das Einfügen von Typen in eine andere Bibliothek entspricht einer Aktualisierung der Zielbibliothek.

Wenn Sie einen Typen in eine andere Bibliothek eingefügt haben, gelten folgende Regeln:

- Ein Typ wird immer mit all seinen Versionen eingefügt.
- Wenn der Typ bereits in der Zielbibliothek existiert, dann werden alle noch nicht vorhandenen Versionen beim dem Typen in der Zielbibliothek ergänzt.
- Wenn eine Version im freigegebenen Zustand bereits in der Zielbibliothek existiert, wird dieselbe Version nicht noch einmal eingefügt.
- Wenn dieselbe Version im Test oder in Bearbeitung bereits in der Zielbibliothek existiert, wird diese durch die freigegebene Version ersetzt.
- Wenn ein Typ andere Typen benötigt, dann werden diese an entsprechender Stelle ebenfalls eingefügt.

Einfügen von Kopiervorlagen

Beim Einfügen von Kopiervorlagen werden alle darin verwendeten Typ-Versionen ebenfalls eingefügt. Wenn die entsprechenden Typen bereits in der Bibliothek vorhanden sind, werden die fehlenden Versionen zu den einzelnen Typen ergänzt. Wenn ein verwendeter Typ noch nicht vorhanden ist, wird der Typ auf oberster Ebene in die Bibliothek eingefügt. Der Typ enthält die Typ-Version, die in der Kopiervorlage verwendet war.

Elemente verschieben

Wenn Sie ein Element aus einer Bibliothek in eine andere Bibliothek verschieben, wird das Element kopiert und nicht verschoben. Es gelten dann dieselben Regeln, wie sie unter "Einfügen von Typen" und "Einfügen von Kopiervorlagen" genannt sind.

Löschen von Typen und Typ-Versionen

Beim Löschen von Typen oder Typ-Versionen ist Folgendes zu beachten:

- Ein Typ oder eine Typ-Version lässt sich nur löschen, wenn keine Abhängigkeiten zu anderen Typen bestehen.
- Wenn Sie einen Typen löschen, dann werden alle Versionen des Typen gelöscht.
- Wenn Sie alle Versionen eines Typen löschen, dann wird der Typ ebenfalls gelöscht.
- Wenn Sie eine Version löschen, die Instanzen im Projekt besitzt, dann werden die Instanzen ebenfalls aus dem Projekt gelöscht.
- Wenn Sie einen Typen löschen, der gleichzeitig als Kopiervorlage gespeichert ist, dann wird die Kopiervorlage ebenfalls gelöscht.

Löschen von Instanzen

Wenn Sie eine Instanz löschen, die Abhängigkeiten zu anderen Instanzen besitzt, dann wird diese bei der nächsten Übersetzung wiederhergestellt. Die Instanz wird wieder mit der ursprünglichen Typ-Version verknüpft. Somit wird die Konsistenz des Projekts wieder hergestellt.

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)
Verbindung zwischen Instanz und Typ aufheben (Seite 80)
Bibliothek mit den Typen einer anderen Bibliothek aktualisieren (Seite 93)
Typen duplizieren (Seite 68)

Namen und Pfadstruktur harmonisieren 14

Sie haben die Möglichkeit, das Projekt mit einer Bibliothek zu harmonisieren. Dadurch können Sie folgende Punkte korrigieren lassen:

- Namen der Instanzen:
  Während der Entwicklungsphase einer Bibliothek können Instanzen entstehen, deren Namen durch eine automatische Korrektur um "_1", „_2“ etc. erweitert wurden. Die Erweiterung erfolgt, um doppelte Namen im Projekt zu vermeiden. Beim Harmonisieren erhalten die Instanzen wieder die Namen ihrer zugehörigen Typen.
- Pfadstruktur:
  Durch parallele Entwicklung oder Kopieren von abhängigen Instanzen kann die ursprüngliche Pfadstruktur verloren gehen. Dies beeinträchtigt die Übersichtlichkeit des Projekts. Beim Harmonisieren wird die Pfadstruktur innerhalb des Projekts an die Pfadstruktur der Bibliothek angepasst.

Vorgehen

Um die Namen und die Pfadstruktur zu harmonisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksverwaltung.
2. Klicken Sie in der Funktionsleiste auf "Projekt harmonisieren".
   Der Dialog "Projekt harmonisieren" wird geöffnet.
3. Wählen Sie das Gerät, mit dem Sie die Bibliothek harmonisieren möchten.
4. Aktivieren Sie das Optionskästchen "Pfade zwischen Projekt und Bibliothek harmonisieren", wenn Sie die Pfadstruktur wiederherstellen möchten.
5. Aktivieren Sie das Optionskästchen "Namen zwischen Projekt und Bibliothek harmonisieren", wenn Sie die Namen korrigieren lassen möchten.
6. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Abhängig von Ihren Einstellungen werden die Namen und die Pfadstruktur im Projekt mit der Bibliothek harmonisiert.

Die Änderungen am Projekt werden protokolliert. Das Protokoll finden Sie in der Projektnavigation unter "Gemeinsame Daten > Protokolle".

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Übersicht über die Bibliotheksansicht (Seite 17)

Übersicht über die Bibliotheksverwaltung (Seite 21)

Bibliothekselemente vergleichen 15

Einführung

Sie können Geräte aus Bibliotheken sowohl mit Geräten aus dem aktuellen Projekt als auch aus der gleichen oder aus anderen Bibliotheken oder aus Referenzprojekten vergleichen. Beachten Sie jedoch, dass Referenzprojekte schreibgeschützt sind. Sie haben auch die Möglichkeit, Instanzen in einem Gerät mit ihrer Typ-Version in einer Bibliothek zu vergleichen. Beim Vergleich mit Typen stehen Ihnen nicht alle Aktionen zur Verfügung. So können Sie z. B. eine Instanz einer neueren Version nicht mit einer älteren Typ-Version aus der Bibliothek überschreiben.

Sie können beim Vergleichen von Bibliothekselementen jederzeit zwischen automatischem und manuellem Vergleich wechseln.

Vorgehen

Um Bibliothekselemente mit den Gerätedaten eines Projekts zu vergleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, dessen Daten Sie mit einem Bibliothekselement vergleichen möchten und das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Öffnen Sie die Task Card "Bibliotheken".
4. Selektieren Sie das Bibliothekselement, das Sie mit den Gerätedaten vergleichen möchten.
5. Ziehen Sie das Bibliothekselement in die rechte Drop-Fläche des Vergleichseditors.
   Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.
   Sie können jederzeit aus dem aktuellen Projekt, einer Bibliothek oder aus einem Referenzprojekt andere Geräte in die Drop-Flächen ziehen und somit einen neuen Vergleich starten. Dabei ist es unerheblich, welches Gerät Sie in welche Drop-Fläche ziehen.

Siehe auch

Bibliotheksansicht verwenden (Seite 17)

Grundlagen zu Bibliotheken (Seite 7)

Übersicht über die Task Card "Bibliotheken" (Seite 13)

Übersicht über die Bibliotheksansicht (Seite 17)

Elementeansicht verwenden (Seite 16)

Globale Bibliotheken verwenden (Seite 31)

Bibliothek aufräumen 16

Sie können die Projektbibliothek und die globalen Bibliotheken aufräumen, um Typen oder Typ-Versionen zu entfernen, die mit keiner Instanz im Projekt verbunden sind. Dies erhöht die Übersichtlichkeit innerhalb der Bibliotheken und verringert die Größe der Bibliothek.

Projektbibliothek aufräumen

Um die Projektbibliothek aufzuräumen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksverwaltung.
2. Klicken Sie in der Funktionsleiste auf "Bibliothek aufräumen".
   Der Dialog "Projektbibliothek aufräumen" wird geöffnet.
3. Wählen Sie aus, in welchem Umfang Typen oder Typ-Versionen gelöscht werden:
   - Um die Version mit der höchsten Versionsnummer zu erhalten, auch wenn diese keine Instanz besitzt, wählen Sie die Option "Alte Typ-Versionen löschen und die neueste Typ-Version erhalten".
   - Um den kompletten Typ zu löschen, wenn keine Version mit einer Instanz verbunden ist, wählen Sie die Option "Komplette Typen löschen".
4. Bestätigen Sie Ihre Eingabe mit "OK".
   Abhängig von Ihrer Auswahl werden entweder nicht verwendete Typ-Versionen oder Typen aus der Projektbibliothek entfernt.
   Die Änderungen werden protokolliert. Das Protokoll finden Sie in der Projektnavigation unter "Gemeinsame Daten > Protokolle".

Globale Bibliothek aufräumen

Um eine globale Bibliothek aufzuräumen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek nicht schreibgeschützt.
2. Öffnen Sie die Bibliotheksverwaltung.
3. Klicken Sie in der Funktionsleiste auf "Bibliothek aufräumen".
   Der Dialog "Globale Bibliothek aufräumen" wird geöffnet.
4. Klicken Sie auf "OK".
   Nicht verwendete Typ-Versionen werden gelöscht. Die neueste Version eines Typs bleibt immer erhalten.
   Die Änderungen werden protokolliert. Das Protokoll finden Sie der globalen Bibliothek unterlagert im Ordner "Gemeinsame Daten > Protokolle".

Siehe auch

Grundlagen zu Bibliotheken (Seite 7)

Übersicht über die Bibliotheksansicht (Seite 17)

Übersicht über die Bibliotheksverwaltung (Seite 21)

Protokolle zu globalen Bibliotheken anzeigen (Seite 38)

Index

A

Anwenderbibliotheken, 8

B

Bibliothek
- aufräumen, 123
- Bibliothekselemente ausschneiden, 115
- Bibliothekselemente vergleichen, 121
- Bibliothekselemente verschieben, 116
- Filteransicht verwenden, 54
- Freigegebene Typ-Version, 60
- Freigegebene Typ-Version öffnen, 62
- Grundlagen, 7
- Instanzen löschen, 116
- Kopiervorlage hinzufügen, 52
- Kopiervorlagen, 7
- Kopiervorlagen aus der Zwischenablage einfügen, 116
- Kopiervorlagen ausschneiden, 115
- Kopiervorlagen einsetzen, 51
- Kopiervorlagen verwenden, 55
- Namen und Pfadstruktur harmonisieren, 119
- Ordner erstellen, 113
- Typen, 7
- Typen aus der Zwischenablage einfügen, 115
- Typen ausschneiden, 115
- Typen einsetzen, 57
- Typen in die Zwischenablage kopieren, 115
- Typen und Versionen löschen, 116
- Typ-Version im Test, 60
- Versionierung von Typen, 57

Bibliothek aktualisieren, 94

Bibliotheksansicht
- öffnen, 20
- Übersicht, 17
- verlassen, 20

Bibliotheksobjekte vergleichen, 8

Bibliotheksverwaltung
- Beziehungen zwischen Bibliotheksobjekten anzeigen, 29
- Filtern nach nicht freigegebenen Typen, 26
- Instanzen von Typ-Versionen anzeigen, 27
- öffnen, 25
- Querverweise einer Instanz anzeigen, 28
- Übersicht, 21

G

Globale Bibliothek
- archivieren, 44
- aufräumen, 123
- Eigenschaften anzeigen, 37
- erstellen, 31
- Filteransicht verwenden, 54
- löschen, 42
- öffnen, 34
- Ordner erstellen, 113
- Projekt aktualisieren, 85
- Protokolle anzeigen, 38
- schließen, 41
- speichern, 39
- Typen hinzufügen, 82
- Typen verwenden, 84

Globale Bibliothek mit Schreibschutz
- erstellen, 32

Globale Bibliotheken, 8
- Anwenderbibliotheken, 8
- archivieren, 43
- dearchivieren, 45
- Rückwärtskompatibilität zu früheren Produktversionen, 33
- Systembibliotheken, 8
- Unternehmensbibliotheken, 8

I

Instanz
- Instanz erzeugen, 69
- zugehörigen Typen anzeigen, 70

P

Projekt
- harmonisieren, 119

Projektbibliothek, 7
- Alle Versionen verwerfen, 76
- aufräumen, 123
- Filteransicht verwenden, 54
- Konsistenzprüfung für eine Typ-Version durchführen, 74
- Ordner erstellen, 113
- Projekt aktualisieren, 79
- Typen duplizieren, 68

Typen gemeinsame Version zuweisen, 87
Typen hinzufügen, 64
Typen verwenden, 69
Typ-Version verwerfen, 75
Projektibliothek
Alle Versionen freigeben, 78
Typ-Versionen freigeben, 76

S

Schreibgeschützte globale Bibliothek
erstellen, 32
Systembibliotheken, 8

T

Typ
Eigenschaften einer Version anzeigen, 63
Eigenschaften eines Typen anzeigen, 63
Instanz erstellen, 84
Typen
Instanz erstellen, 69
Typen verwenden, 69
Verwendungen einfügen, 69
Verwendungen kopieren, 69
Typen auf neueste Version aktualisieren, 94
Typen zu einer Instanz anzeigen, 70

U

Unternehmensbibliotheken, 8

V

Version eines Typen im Test bearbeiten, 73
Versionierung von Typen
Freigegebene Version, 60
Version im Test, 60

SIEMENS

Geräte und Netze konfigurieren 1

Geräte und Netze diagnostizieren 2

Geräte und Netze bearbeiten

Programmier- und Bedienhandbuch

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Geräte und Netze konfigurieren....9

1.1 Hardware- und Netzwerkeditor....9
1.1.1 Übersicht über den Hardware- und Netzwerkeditor....9
1.1.2 Übersicht der Einstellungen zur Hardware-Konfiguration....10
1.1.3 Netzsicht....11
1.1.4 Gerätesicht....14
1.1.5 Gerätesicht: Objekte in der Gerätesicht....17
1.1.6 Gerätesicht: Ablage nicht gesteckter Baugruppen....20
1.1.7 Topologiesicht....21
1.1.8 Inspektorfenster....24
1.1.9 Inspektorfenster: Darstellung von UDTs....27
1.1.10 Hardware-Katalog....29
1.1.11 Hardware-Katalog durchsuchen....32
1.1.12 Produktsupport aktivieren....34
1.1.13 Produktsupport zu Hardware-Komponenten anzeigen....34
1.1.14 Drucken von Hardware- und Netzkonfigurationen....35
1.1.15 Drucken: Einstellungen ändern....37
1.1.16 Drucken: Seitenumbruchvorschau aktivieren....38
1.1.17 Tastaturbedienung: Navigation im Editor....38
1.1.18 Tastaturbedienung: Bearbeiten von Objekten....39

1.2 Geräte konfigurieren....41
1.2.1 Einführung zum Konfigurieren der Hardware....41
1.2.2 Gerät zur Hardware-Konfiguration hinzufügen....42
1.2.3 Nicht spezifizierte CPU hinzufügen....43
1.2.4 CPU auswählen....45
1.2.5 Fehlersichere Baugruppen auswählen....47
1.2.6 Baugruppenträger: Grundlagen....48
1.2.7 Baugruppenträger: Allgemeine Steckplatzregeln....49
1.2.8 Baugruppenträger: Baugruppe stecken....51
1.2.9 Hardware-Komponente selektieren....53
1.2.10 Hardware-Komponente kopieren....57
1.2.11 Hardware-Komponente verschieben....58
1.2.12 Hardware-Komponente tauschen....60
1.2.13 Hardware-Komponente umbenennen....60
1.2.14 Hardware-Komponente löschen....62
1.2.15 Hardware-Komponente ändern oder aktualisieren....63
1.2.16 Eigenschaften und Parameter bearbeiten....64
1.2.17 Ein- und Ausgangsadressen in der Adressübersicht....66
1.2.18 Dezentrale Peripherie in der Projektnavigation....70
1.2.19 Geräte vergleichen....72
1.2.19.1 Grundlagen zum Gerätevergleich....72
1.2.19.2 Gerätevergleich durchführen....73

1.3 Netze konfigurieren....74
1.3.1 Geräte vernetzen....74
1.3.1.1 Kommunikation und Netze....74

1.3.1.2 Geräte in der Netzsicht vernetzen....75
1.3.1.3 Tabellarische Netzübersicht....79
1.3.1.4 Geräte in der Gerätesicht vernetzen....80
1.3.1.5 Netzparameter und Schnittstellenparameter prüfen oder ändern....81
1.3.1.6 Vernetzungen ändern....82
1.3.1.7 Subnetze kopieren, ausschneiden oder löschen....83
1.3.1.8 Netzkonfiguration MPI....84
1.3.1.9 Netzkonfiguration PROFIBUS....85
1.3.1.10 Netzkonfiguration Industrial Ethernet....92
1.3.1.11 Einstellungen für das Verschalten von Ethernet-Teilnehmern....94
1.3.2 Kommunikation über Verbindungen....98
1.3.2.1 Arbeiten mit Verbindungen....98
1.3.2.2 Verbindungen - Bedeutung und Auswahl....200
1.3.2.3 Verbindungen projektieren....221
1.3.2.4 Verbindungstypen....250
1.3.3 Topologie anzeigen und projektieren....304
1.3.3.1 Überblick über die Topologiesicht....304
1.3.3.2 Topologiesicht starten....307
1.3.3.3 Topologie anzeigen....307
1.3.3.4 Topologie projektieren....314
1.3.4 Secure Communication....323
1.3.4.1 Grundlagen zu Secure Communication....323
1.3.4.2 Globale Security-Einstellungen....343
1.3.4.3 Secure Open User Communication....351
1.3.5 Industrial Ethernet Security....367
1.3.5.1 Security projektieren....367
1.3.5.2 Security für SCALANCE S-600 projektieren....585

1.4 Konfigurationen erstellen....753
1.4.1 Automatisierungssysteme konfigurieren....753
1.4.1.1 Arbeitsweise von S7-300/400 CPUs....753
1.4.1.2 Arbeitsweise von S7-1500 CPUs....951
1.4.1.3 Arbeitsweise von S7-1500R/H CPUs....1095
1.4.1.4 Konfigurieren von zentralen Baugruppen....1138
1.4.1.5 Regeln für Baugruppen....1139
1.4.1.6 Regeln für Baugruppen....1140
1.4.1.7 Regeln für Baugruppen....1142
1.4.1.8 Zentrale Baugruppen auswählen....1151
1.4.1.9 Fehlersichere Baugruppen auswählen....1152
1.4.1.10 Eigenschaften zentraler Baugruppen ändern....1154
1.4.1.11 Kopplung von Baugruppenträgern....1155
1.4.1.12 Kopplung von Baugruppenträgern....1157
1.4.1.13 Baugruppen adressieren....1160
1.4.1.14 Flexible Automatisierungskonzepte....1177
1.4.1.15 OPC UA-Kommunikation nutzen....1191
1.4.1.16 Webserver konfigurieren....1317
1.4.2 Automatisierungssysteme konfigurieren....1364
1.4.2.1 Arbeitsweise von S7-1200 CPUs....1364
1.4.2.2 Baugruppen adressieren....1458
1.4.2.3 Signalboard....1462
1.4.2.4 Konfigurationen für Webserver....1463
1.4.2.5 Konfigurationssteuerung (Optionenhandling) für Geräte....1499
1.4.2.6 Weitere Konfigurationen....1510

1.4.3 PC-Systeme konfigurieren....1510
1.4.3.1 Übersicht....1510
1.4.3.2 Integration von IPCs ins TIA Portal....1542
1.4.3.3 Grundkonfiguration des Open Controllers....1548
1.4.3.4 Parametrierung....1550
1.4.3.5 Besonderheiten zum S7-modular Embedded Controller....1561
1.4.4 CPs und Applikationen für PC-Stationen konfigurieren....1564
1.4.4.1 CPs für PC-Station 1.0....1564
1.4.4.2 Applikationen für PC-Station 1.0....1586
1.4.5 Konfigurationen für Punkt-zu-Punkt-Kopplungen (S7-1500)....1614
1.4.5.1 Konfigurationen für Punkt-zu-Punkt-Kopplungen....1614
1.4.6 Konfigurationen für Punkt-zu-Punkt-Kopplungen (S7-300, S7-400)....1670
1.4.6.1 Konfigurationen für Punkt-zu-Punkt-Kopplungen....1670
1.4.7 TeleControl....1797
1.4.7.1 Telecontrol-Module konfigurieren....1797
1.4.8 SCALANCE X, W und M....2063
1.4.8.1 SCALANCE X / W / M projektieren....2063
1.4.9 PROFIBUS DP konfigurieren....2871
1.4.9.1 Konfigurieren von dezentraler Peripherie....2871
1.4.9.2 Grundlagen zum Konfigurieren eines DP-Mastersystems....2871
1.4.9.3 DP-Slaves im Hardware-Katalog....2873
1.4.9.4 DP/DP-Koppler im Hardware-Katalog....2873
1.4.9.5 Konfigurationen mit PROFIBUS-DP....2874
1.4.9.6 Dezentrale Peripheriesysteme konfigurieren....2878
1.4.9.7 Beispiel: ET 200S als DP-Slave konfigurieren....2889
1.4.9.8 Intelligente DP-Slaves konfigurieren....2901
1.4.9.9 Direkten Datenaustausch konfigurieren....2914
1.4.9.10 GSD-Dateien verwenden....2922
1.4.9.11 PROFIBUS DPV1 anwenden....2926
1.4.9.12 Äquidistanz und Taktsynchronität....2937
1.4.10 PROFIBUS DP konfigurieren....2956
1.4.10.1 Grundlagen zum Konfigurieren eines DP-Mastersystems....2956
1.4.10.2 DP-Slaves im Hardware-Katalog....2956
1.4.10.3 DP/DP-Koppler im Hardware-Katalog....2957
1.4.10.4 Konfigurationen mit PROFIBUS-DP....2957
1.4.10.5 Dezentrale Peripheriesysteme konfigurieren....2959
1.4.10.6 Intelligente DP-Slaves konfigurieren....2968
1.4.10.7 DP-Slaves der Dezentralen Peripherie konfigurieren....2970
1.4.10.8 GSD-Dateien verwenden....2979
1.4.11 PROFINET IO konfigurieren....2983
1.4.11.1 Wissenswertes zu PROFINET IO....2983
1.4.11.2 PROFINET IO konfigurieren....2989
1.4.11.3 GSD-Dateien verwenden....3035
1.4.11.4 Spezielle PROFINET-Konfigurationen....3040
1.4.12 Buskopplung mit PN/PN Coupler....3187
1.4.12.1 Anwendungsgebiet und Funktion....3187
1.4.12.2 Ethernet-Subnetze koppeln....3187
1.4.13 Externe Tools einbinden....3188
1.4.13.1 Einbindung von S7-externen Tools....3188
1.4.13.2 SIMATIC S7-PCT starten....3189
1.4.14 Konfiguration laden....3190
1.4.14.1 Einführung zum Laden einer Konfiguration....3190

1.4.14.2 Laden einer Konfiguration in ein Gerät....3191
1.4.14.3 Laden einer Konfiguration in PG/PC....3192

1.5 Meldungen anzeigen....3203
1.5.1 Übersicht über die Meldungsanzeige....3203
1.5.2 Ansicht "Aktuelle Meldungen"....3204
1.5.3 Aufbau der Meldungen in der Ansicht "Aktuelle Meldungen"....3204
1.5.4 Meldungsarchiv....3204
1.5.5 Aufbau der Meldungen im Meldungsarchiv....3205
1.5.6 Meldungsarchiv vollständig leeren....3205
1.5.7 Archiv exportieren....3206
1.5.8 Meldungen empfangen....3206
1.5.9 Meldungen einfrieren....3207
1.5.10 Meldungen quittieren....3207
1.5.11 Status der Meldungen....3208
1.5.12 Tabelle in der Meldungsanzeige sortieren....3209
1.5.13 Tabelle in der Meldungsanzeige filtern....3209
1.5.14 Tastaturbedienung in der Meldungsanzeige....3210

1.6 Zusatzinformationen zu Konfigurationen....3211
1.6.1 Kommunikationsmodule und Netzkomponenten....3211
1.6.1.1 Industrial Ethernet / PROFINET....3211
1.6.1.2 Industrial Ethernet....3246
1.6.1.3 CP PROFIBUS....3260
1.6.1.4 CM/CP PROFIBUS....3264
1.6.1.5 Protokollunabhängige / Netzunabhängige CP-Einstellungen....3265
1.6.1.6 Hintergrund zu Verbindungen....3267
1.6.1.7 IE/PB Link PN IO....3274
1.6.2 Identifikationssysteme....3277
1.6.2.1 RFID-Systeme....3277
1.6.2.2 Optische Lesesysteme....3355
1.6.3 Dezentrale Peripherie....3358
1.6.3.1 Dezentrale Peripheriesysteme....3358
1.6.3.2 Projektieren von HART-Variablen....3361
1.6.3.3 ET 200iSP....3362
1.6.3.4 ET 200eco PN....3376
1.6.3.5 ET 200SP....3382
1.6.3.6 ET 200AL....3424
1.6.3.7 ET 200MP....3440
1.6.3.8 ET 200M....3453
1.6.3.9 ET 200S....3463
1.6.3.10 ET 200pro....3464
1.6.4 IPv4-Routing beim CP 1543-1 / CM 1542-1....3472
1.6.5 IPv6-Konfiguration....3472
1.6.5.1 IPv6-Protokoll....3472
1.6.5.2 IPv6 beim CP 1543-1....3473

2 Geräte und Netze diagnostizieren....3475

2.1 Hardware diagnostizieren....3475
2.1.1 Überblick über die Hardware-Diagnose....3475
2.1.1.1 Prinzipielle Möglichkeiten, Hardware zu diagnostizieren....3475
2.1.1.2 Unter den online verbundenen Geräten diejenigen ermitteln, die gestört sind....3476
2.1.1.3 Diagnose- und Vergleichsstatus mittels Symbolen anzeigen....3477

2.1.1.4 Online- und Diagnosesicht starten....3485
2.1.1.5 Aktivieren der Task Card "Online-Tools"....3487
2.1.2 Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen....3487
2.1.2.1 Allgemeine Eigenschaften und anlagenrelevante Informationen einer Baugruppe anzeigen....3487
2.1.2.2 Parametrierte Zykluszeiten anzeigen....3489
2.1.2.3 Schnittstellen und Schnittstelleneigenschaften einer Baugruppe anzeigen....3489
2.1.2.4 IO-Controller anzeigen, die auf Module eines Shared Device zugreifen....3491
2.1.2.5 Sync-Domain-Eigenschaften eines PROFINET-Geräts anzeigen....3492
2.1.2.6 MRP-Domain-Eigenschaften eines PROFINET-Geräts anzeigen....3493
2.1.2.7 Aktuelle Firmware einer Baugruppe anzeigen....3494
2.1.3 Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen....3494
2.1.3.1 Gemessene Zykluszeiten anzeigen....3494
2.1.3.2 Aktuellen Zustand der LEDs einer CPU anzeigen....3495
2.1.3.3 Füllstand aller Speicherarten einer CPU anzeigen....3496
2.1.3.4 Füllstand aller Speicherarten einer S7-1500-CPU anzeigen....3497
2.1.4 Eine Baugruppe auf Störungen überprüfen....3498
2.1.4.1 Diagnosestatus einer Baugruppe ermitteln....3498
2.1.4.2 Kommunikationsdiagnose von PROFINET-Ports durchführen....3499
2.1.4.3 PROFINET IO-Komponenten diagnostizieren....3500
2.1.4.4 Den Diagnosepuffer einer CPU auslesen....3501
2.1.4.5 Servicedaten speichern....3505
2.1.5 Eigenschaften einer Baugruppe bzw. des PG/PC ändern....3506
2.1.5.1 Den Betriebszustand einer CPU umschalten....3506
2.1.5.2 Urlöschen durchführen....3507
2.1.5.3 Die Uhrzeit einer CPU ermitteln und einstellen....3508
2.1.5.4 Eine S7-1200-CPU auf Werkseinstellungen zurücksetzen....3509
2.1.5.5 Eine S7-1500-CPU auf Werkseinstellungen zurücksetzen....3510
2.1.5.6 Ein PROFINET IO-Gerät auf Werkseinstellungen zurücksetzen....3511
2.1.5.7 Eine S7-1500-Memory Card formatieren....3513
2.1.5.8 Die Firmware einer Baugruppe aktualisieren....3513
2.1.5.9 Einem PROFINET IO-Device eine IP-Adresse zuweisen....3516
2.1.5.10 PROFINET-Gerätename vergeben....3518
2.1.5.11 Ein S7-1500-Analogmodul kalibrieren....3522
2.1.5.12 I&M-Daten in PROFINET-IO-Geräte und ihre Module laden....3526
2.1.6 Diagnose im Betriebszustand STOP....3527
2.1.6.1 Grundlagen zum Diagnosepuffer....3527
2.1.6.2 Die STOP-Ursache einer CPU ermitteln....3529
2.1.7 Online-Zugänge in der Online- und Diagnosesicht....3529
2.1.7.1 Status der Online-Verbindung anzeigen....3529
2.1.7.2 PG/PC-Schnittstelle festlegen, Online verbinden....3530
2.1.7.3 Online-Verbindung trennen....3531
2.1.7.4 Den Blinktest bei einem online verbundenen Gerät durchführen....3532
2.1.8 PROFIBUS DP-Subnetze auf Störungen überprüfen....3533
2.1.8.1 Grundlagen zum Diagnose-Repeater....3533
2.1.8.2 Status der Segmentdiagnose mittels Symbolen anzeigen....3534
2.1.8.3 Status der Segmentdiagnose grafisch und textuell anzeigen....3534

2.2 Verbindungen diagnostizieren....3536
2.2.1 Überblick über die Verbindungsdiagnose....3536
2.2.2 Verbindungsstatus mittels Symbolen anzeigen....3537
2.2.3 Verbindungen im Detail diagnostizieren....3538

2.2.3.1 Verbindungen im Detail diagnostizieren - Überblick....3538
2.2.3.2 Online-Verbindungsressourcen bei S7-1200 ermitteln....3538
2.2.3.3 Online-Verbindungsressourcen bei S7-1500 ermitteln....3539
2.2.3.4 Verbindungsdetails ermitteln....3540
2.2.3.5 Adressdetails einer Verbindung ermitteln....3540

2.3 Systemdiagnose mit 'Report System Errors'....3542
2.3.1 Einführung zur Systemdiagnose mit "Report System Errors"....3542
2.3.2 Grundlagen zur Systemdiagnose....3542
2.3.3 Baugruppen mit eingeschränkten Diagnosemöglichkeiten....3543
2.3.4 Diagnosebausteine für das Melden von Systemfehlern....3544
2.3.5 Eigenschaften der Bausteine....3545
2.3.6 Unterstützte Fehler-OBs....3546
2.3.7 Diagnosestatus-DB....3547
2.3.8 Einstellungen der Systemdiagnose anzeigen....3550
2.3.9 Einstellungen....3551
2.3.9.1 Allgemeine Einstellungen....3551
2.3.9.2 Meldungseinstellungen....3551
2.3.9.3 Einstellungen für die Diagnoseunterstützung....3551
2.3.9.4 Einstellungen für Systemdiagnosebausteine....3552
2.3.10 Systemdiagnoseeinstellungen exportieren und importieren....3552

2.4 Systemdiagnose für S7-1500 PLCs....3553
2.4.1 Einführung....3553
2.4.2 Grundlagen zur Systemdiagnose....3553
2.4.3 Einstellungen der Systemdiagnose anzeigen....3553
2.4.4 Einstellungen....3554
2.4.4.1 Allgemeine Einstellungen....3554
2.4.4.2 Meldungseinstellungen....3554
2.4.5 Systemdiagnoseeinstellungen exportieren und importieren....3555
2.4.5.1 Grundlagen zum Exportieren und Importieren von Systemdiagnoseeinstellungen....3555
2.4.5.2 Systemdiagnoseeinstellungen exportieren....3555
2.4.5.3 Systemdiagnoseeinstellungen importieren....3555

2.5 Zusätzliche Diagnosemöglichkeiten nutzen....3557
2.5.1 Hardware diagnostizieren....3557
2.5.1.1 Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen....3557
2.5.1.2 Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen....3562
2.5.1.3 Eine Baugruppe auf Störungen überprüfen....3565
2.5.1.4 Eigenschaften einer Baugruppe bzw. des PG/PC ändern....3568
2.5.2 Verbindungen diagnostizieren....3576
2.5.2.1 Verbindungen im Detail diagnostizieren....3576

Index....3577

Geräte und Netze konfigurieren 1

1.1 Hardware- und Netzwerkeditor

1.1.1 Übersicht über den Hardware- und Netzwerkeditor

Funktion des Hardware- und Netzwerkeditors

Wenn Sie in der Projektnavigation auf den Eintrag "Geräte & Netze" doppelklicken, öffnet sich der Hardware- und Netzwerkeditor. Der Hardware- und Netzwerkeditor ist die integrierte Entwicklungsumgebung für das Konfigurieren, Parametrieren und Vernetzen von Geräten und Baugruppen. Er bietet maximale Unterstützung für die Durchführung des Automatisierungsprojekts.

Aufbau des Hardware- und Netzwerkeditors

Der Hardware- und Netzwerkeditor besteht aus den folgenden Komponenten:

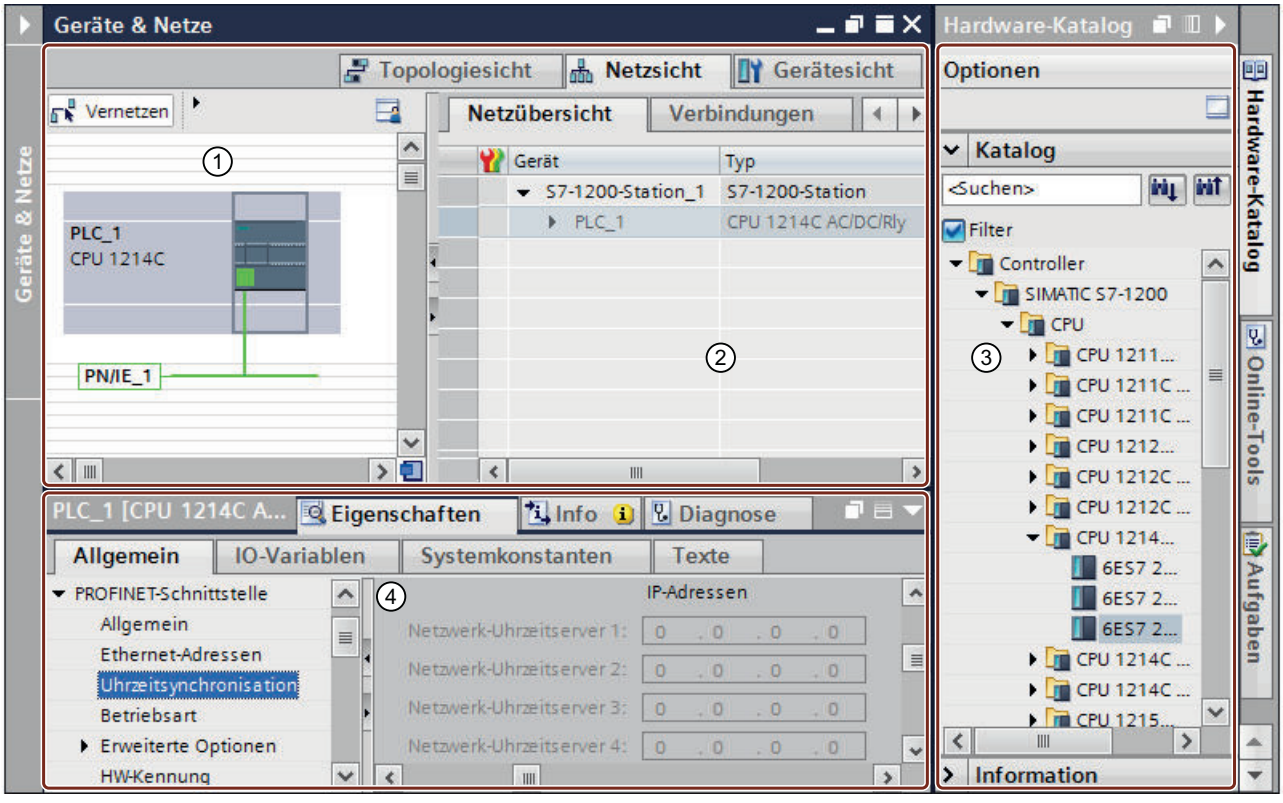

① Gerätesicht (Seite 14), Netzsicht (Seite 11), Topologiesicht (Seite 21): Grafischer Bereich

② Gerätesicht (Seite 14), Netzsicht (Seite 11), Topologiesicht (Seite 21): Tabellarische Bereich

③ Hardware-Katalog (Seite 29)

④ Inspektorfenster (Seite 24)

Der Hardware- und Netzwerkeditor bietet Ihnen drei unterschiedliche Sichten auf Ihr Projekt. Zwischen diesen drei Sichten können Sie jederzeit umschalten, je nachdem, ob Sie einzelne Geräte und Baugruppen, ganze Netzwerke und Gerätekonfigurationen oder den topologischen Aufbau ihres Projektes erstellen und bearbeiten möchten.

Aus dem Hardware-Katalog ziehen Sie sich die für Ihr Automatisierungssystem benötigten Geräte und Baugruppen in die Geräte-, Netz- oder Topologiesicht.

Im Inspektorfenster finden Sie Informationen über das aktuell markierte Objekt. Dort können Sie auch die Einstellungen für das markierte Objekt ändern.

1.1.2 Übersicht der Einstellungen zur Hardware-Konfiguration

Unter "Extras > Einstellungen > Hardware-Konfiguration" lassen sich verschiedene Einstellungen zur Hardware-Konfiguration vornehmen.

Überblick

Die folgende Tabelle zeigt eine Übersicht der Einstellungen zur Hardware-Konfiguration:

Gruppe	Einstellung	Beschreibung
Information zum Produktsupport	Deaktiviert	Verhindert den Zugriff auf den Siemens Industry Online Support
	Über Internet	Ermöglicht den Zugriff auf Produktinformationen zu einzelnen Geräten im Hardware-Katalog über das Internet.
Topologieübersicht	IP-Adresse temporär zuweisen	Vergibt zur Topologieerkennung eine temporäre IP-Adresse, falls ein Gerät keine gültige IP-Adresse besitzt. Ohne eine gültige IP-Adresse lassen sich keine Topologieinformationen (LLDP) von einem Gerät lesen.
	Warnhinweis anzeigen, wenn die Option aktiviert ist	Zeigt einen Warnhinweis an, wenn einem Gerät bei der Topologieerkennung eine temporäre IP-Adresse zugewiesen wird.
Übersetzen und herunterladen	Baugruppenkommentar herunterladen	Überträgt zusätzlich zur Hardware-Konfiguration eventuell vorhandenene Kommentare beim Laden der Hardware-Konfiguration in das Gerät. Der Kommentar steht nach dem Laden des Geräts in ein PG zur Verfügung.
	I&M-Daten herunterladen	Wenn Sie beim Laden der Hardware-Konfiguration oder der Software oder von beidem in ein Gerät zusätzlich I&M-Daten in die reale Hardware laden können wollen, muss diese Option aktiviert sein.

Siehe auch

Produktsupport aktivieren (Seite 34)

Produktsupport zu Hardware-Komponenten anzeigen (Seite 34)

Topologiesicht (Seite 21)

1.1.3 Netzsicht

Einführung

Die Netzsicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Geräte miteinander vernetzen
- Namen der Geräte bearbeiten

Aufbau

Das folgende Bild zeigt die Bestandteile der Netzsicht:

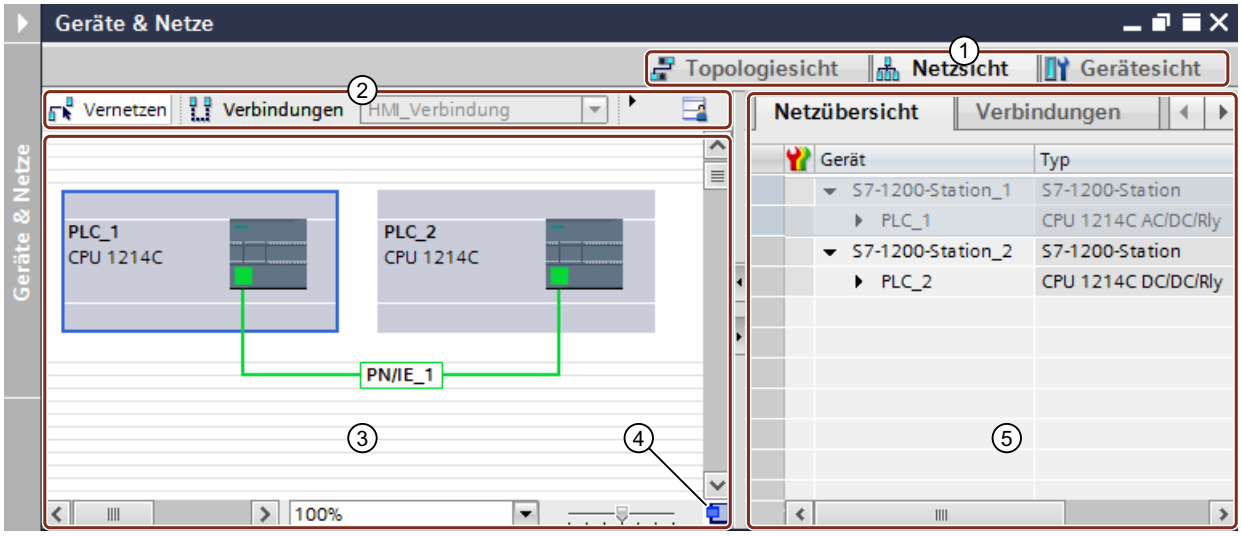

① Umschalter Netzsicht/Gerätesicht/Topologiesicht
② Funktionsleiste der Netzsicht
③ Grafischer Bereich der Netzsicht
④ Übersichtsnavigation
⑤ Tabellarischer Bereich der Netzsicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Netzsicht können Sie mit der Maus verändern. Die Teilung kann über eine Schaltfläche in der Funktionsleiste zwischen horizontaler und vertikaler Teilung umgeschaltet werden. Um die Position der Teilung zu verändern, klicken Sie mit der Maus zwischen den grafischen und tabellarischen Bereich und verändern Sie bei gedrückter Maustaste durch einer Verschieben des Teilers nach Links oder Rechts die Aufteilung. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

Symbol	Bedeutung
	Modus zum Vernetzen der Geräte.
	Modus zum Erstellen von Verbindungen. Über die nebenstehende Klappliste können Sie den Verbindungstyp einstellen.
	Modus zum Erstellen von Relationen.
	Öffnet den Dialog zur manuellen Namensvergabe für PROFINET-Geräte. Dazu muss das IO-Device gesteckt und online mit dem IO-System verbunden sein.

Symbol	Bedeutung
	Anzeige der Schnittstellenadressen. Sie können die Adressen für MPI-, PROFIBUS- und Ethernet-Schnittstellen direkt in der Netzsicht bearbeiten: Selektieren Sie dazu die gewünschte Adresse und klicken Sie dann auf das selektierte Adressfeld oder drücken Sie [F2].
	Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.
	Über das Zoomsymbol können Sie die Ansicht schrittweise vergrößern (+) oder verkleinern (-) oder einen Rahmen um einen zu vergrößernden Bereich ziehen.
	Schaltet die Teilung von grafischem und tabellarischem Bereich der Editorsicht zwischen horizontal und vertikal um.
	Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.

Grafischer Bereich

Im grafischen Bereich der Netzsicht werden Ihnen netzrelevante Geräte, Netze, Verbindungen und Relationen angezeigt. Hier fügen Sie Geräte aus dem Hardware-Katalog ein, verbinden diese über ihre Schnittstellen miteinander und bearbeiten die Kommunikationseinstellungen.

Am unteren Rand des grafischen Bereichs finden Sie Bedienelemente für die Ansichtssteuerung:

- Mit der Klappliste wählen Sie die Zoomstufe aus. Sie können auch direkt einen Wert in das Feld der Klappliste eintragen.
- Sie können die Zoomstufe auch mit dem Schieberegler einzustellen.
- Über das Icon in der rechten unteren Ecke können Sie das Fenster des grafischen Bereichs neu fokussieren.

Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

Tabellarischer Bereich

Der tabellarische Bereich der Netzsicht beinhaltet verschiedene Tabellen zu den vorhandenen Geräten, Verbindungen und Kommunikationseinstellungen:

- Netzübersicht
- Verbindungen
- Relationen
- E/A-Kommunikation
- VPN

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

Siehe auch

Gerät zur Hardware-Konfiguration hinzufügen (Seite 42)

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

Geräte in der Netzsicht vernetzen (Seite 75)

Tabellarische Netzübersicht (Seite 79)

1.1.4 Gerätesicht

Einführung

Die Gerätesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Baugruppen konfigurieren und parametrieren
- Namen der Geräte und Module bearbeiten

Aufbau

Das folgende Bild zeigt die Bestandteile der Gerätesicht:

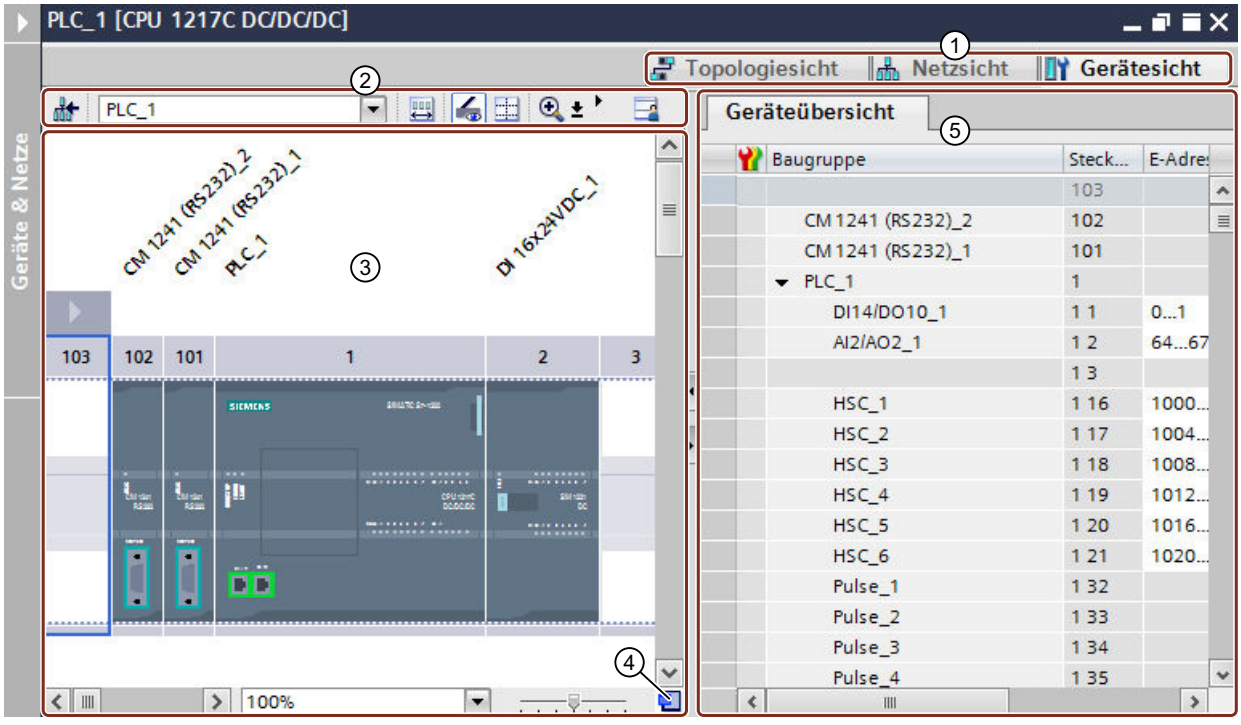

① Umschalter Netzsicht/Gerätesicht/Topologiesicht

② Funktionsleiste der Gerätesicht

③ Grafischer Bereich der Gerätesicht

④ Übersichtsnavigation

⑤ Tabellarischer Bereich der Gerätesicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Gerätesicht können Sie mit der Maus verändern. Die Teilung kann über eine Schaltfläche in der Funktionsleiste zwischen horizontaler und vertikaler Teilung umgeschaltet werden. Um die Position der Teilung zu verändern, klicken Sie mit der Maus zwischen den grafischen und tabellarischen Bereich und verändern Sie bei gedrückter Maustaste durch einer Verschieben des Teilers nach Links oder Rechts die Aufteilung. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

Funktionsleiste

Über die Klappliste in der Funktionsleiste können Sie in der Gerätesicht zwischen den Geräten Ihres Projektes umschalten. Neben der Klappliste zur Geräteauswahl stehen Ihnen außerdem die folgenden Funktionen zur Verfügung:

Symbol	Bedeutung
	Schaltet in die Netzsicht um. Über die nebenstehende Klappliste kann die Gerätesicht zwischen den vorhandenen Geräten umgeschaltet werden.
	Anzeige der Ablage nicht gesteckter Baugruppen.
	Öffnet den Dialog zur manuellen Namensvergabe für PROFINET-Geräte. Dazu muss das IO-Device gesteckt und online mit dem IO-System verbunden sein.
	Anzeige der Baugruppenbeschriftung. Sie können die Beschriftung direkt in der Gerätesicht bearbeiten: Selektieren Sie dazu die gewünschte Beschriftung und klicken Sie dann auf das selektierte Textfeld oder drücken Sie [F2].
	Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.
	Über das Zoomsymbol können Sie die Ansicht schrittweise vergrößern (+) oder verkleinern (-) oder einen Rahmen um einen zu vergrößernden Bereich ziehen. Ab einer Zoomstufe von 200 % können Sie bei Signalmodulen die Adressbezeichnungen der E/A-Kanäle erkennen.
	Schaltet die Teilung von grafischem und tabellarischem Bereich der Editorsicht zwischen horizontal und vertikal um.
	Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.

Grafischer Bereich

Im grafischen Bereich der Gerätesicht werden Ihnen Hardware-Komponenten und ggf. die zugehörigen Baugruppen angezeigt, die über einen oder mehrere Baugruppenträger einander zugeordnet sind. Bei Geräten mit Baugruppenträgern können Sie hier weitere Hardware-Objekte aus dem Hardware-Katalog in die Steckplätze der Baugruppenträger stecken.

Am unteren Rand des grafischen Bereichs finden Sie Bedienelemente für die Ansichtssteuerung:

- Mit der Klappliste wählen Sie die Zoomstufe aus. Sie können auch direkt einen Wert in das Feld der Klappliste eintragen.
- Sie können die Zoomstufe auch mit dem Schieberegler einzustellen.
- Über das Icon in der rechten unteren Ecke können Sie das Fenster des grafischen Bereichs neu fokussieren.

Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

Tabellarischer Bereich

Der tabellarische Bereich der Gerätesicht gibt Ihnen eine Übersicht über die verwendete Hardware mit den wichtigsten technischen und organisatorischen Daten.

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

Siehe auch

Baugruppenträger: Grundlagen (Seite 48)

Netzsicht (Seite 11)

Gerätesicht: Ablage nicht gesteckter Baugruppen (Seite 20)

Baugruppenträger: Baugruppe stecken (Seite 51)

Gerätesicht: Objekte in der Gerätesicht (Seite 17)

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

1.1.5 Gerätesicht: Objekte in der Gerätesicht

Im linken Bereich der Gerätesicht sehen Sie eine grafische Darstellung des Baugruppenträgers und der darauf gesteckten Geräte. Im rechten Bereich der Gerätesicht sehen Sie die Geräteübersicht. Die Geräteübersicht ist eine Tabelle, die Ihnen die wichtigsten Informationen der im Baugruppenträger eingefügten Baugruppen zeigt. Beide Bereiche werden in einem Fenster angezeigt. Die Größe beider Bereiche wird dabei durch einen Teiler eingestellt. Über diesen können Sie auch den einen oder den anderen Bereich vollständig anzeigen bzw. ausblenden.

Aufbau und Inhalt der Gerätesicht

In der grafischen Gerätesicht wird Ihnen die Offline-Konfiguration der Geräte auf dem Baugruppenträger angezeigt. Die Darstellung entspricht symbolisch dem Aufbau auf dem realen Baugruppenträger.

In der Geräteübersicht wird die Belegung des Baugruppenträgers in tabellarischer Form gezeigt. Eine Tabellenzeile beinhaltet jeweils die Informationen für die Belegung eines Steckplatzes.

Das folgende Bild zeigt die Gerätesicht mit der Konfiguration einer SIMATIC S7-1200 CPU.

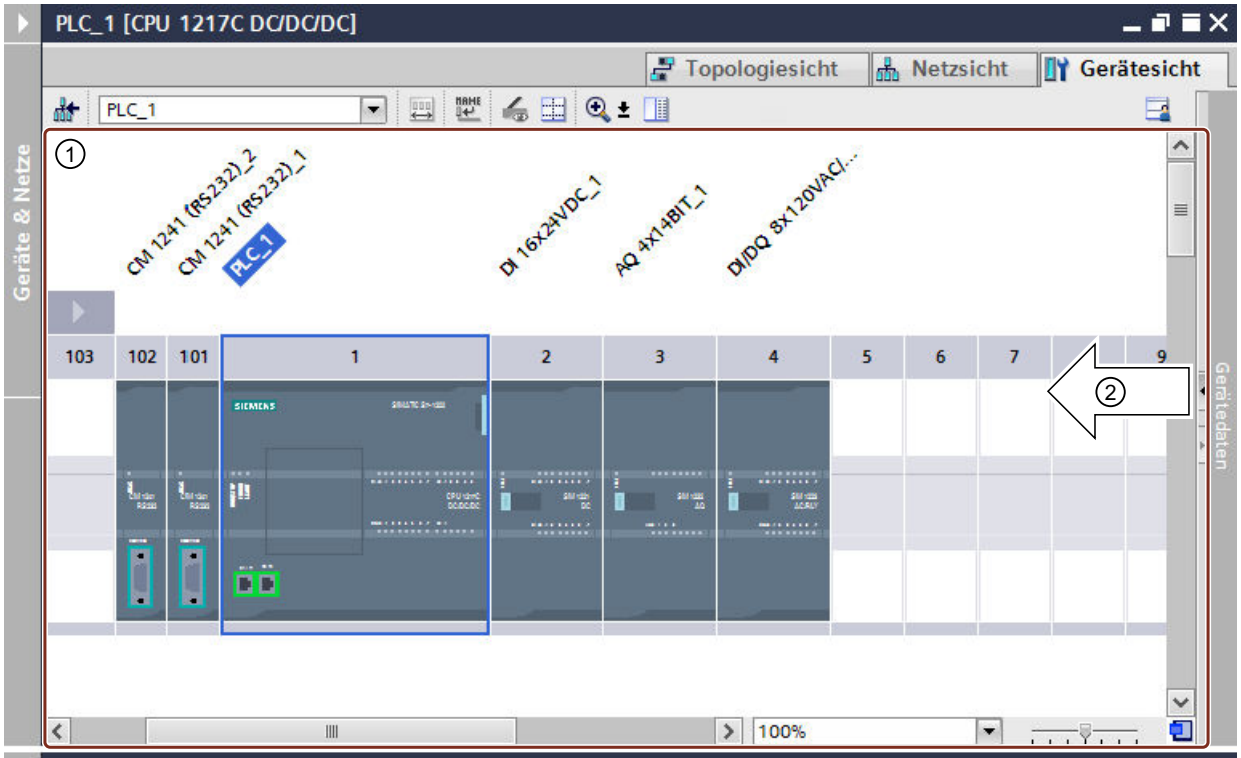

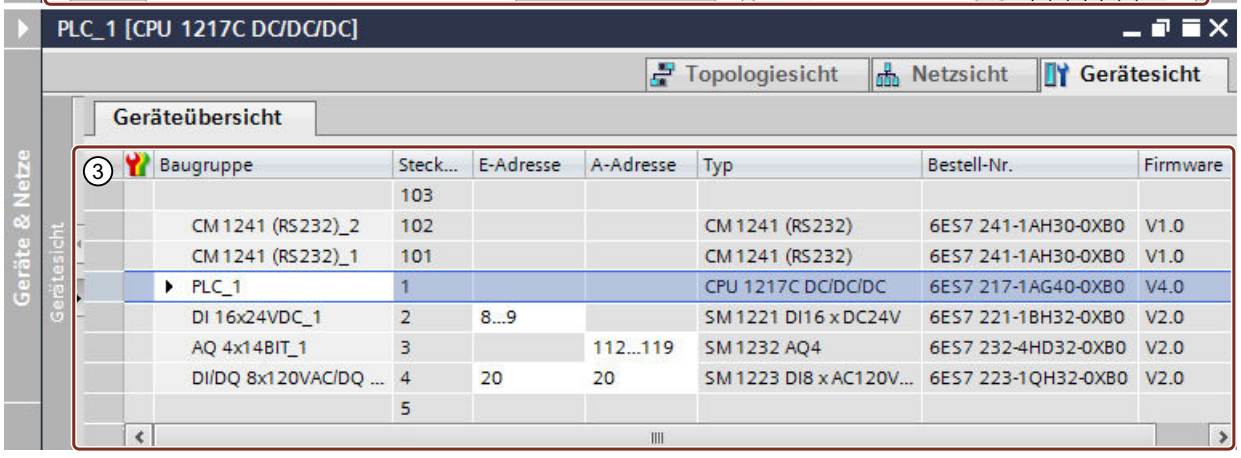

① Grafische Sicht mit der Belegung des Baugruppenträgers durch die CPU und verschiedene Module auf den Steckplätzen 1 bis 4, sowie 101 und 102.

② Über den Teiler können Sie die Teilung der Gerätesicht zwischen dem linken Bereich (grafische Sicht) und dem rechten Bereich (Geräteübersicht) einstellen. Wenn Sie auf die Pfeile klicken, können Sie die Teilung der Bereiche schnell umschalten.

③ Geräteübersicht mit der tabellarischen Darstellung der Steckplätze des Baugruppenträgers und der gesteckten Komponenten.Über das Kontextmenü der Spaltentitel können Sie weitere Spalten einblenden und angezeigte Spalten ausblenden.

Jede Zeile in der Geräteübersicht präsentiert einen Steckplatz. Für jeden Steckplatz werden in den verschiedenen Spalten die wichtigsten Informationen angezeigt:

Spalte	Bedeutung
Online-Status	Symbolische Darstellung des Online-Status
Fehlersicher	Symbolische Darstellung für fehlersichere Module
Baugruppe	Bezeichnung der Baugruppe, frei editierbar
Baugruppenträger	Nummer des Baugruppenträgers
Steckplatz	Nummer des Steckplatzes
E-Adresse	Eingangsadressbereich, editierbar
A-Adresse	Ausgangsadressbereich, editierbar
F-Quelladresse	F-Quelladresse bei Verwendung von F-Peripherie
F-Zieladresse	F-Zieladresse bei Verwendung von F-Peripherie
Typ	Katalogbezeichnung der Baugruppe
Artikel-Nr.	Artikelnummer der Baugruppe
Firmware	Firmware-Version der Baugruppe
Kommentar	Optionaler Kommentar

Anzeige der E/A-Kanäle

Wenn Sie die Zoomstufe in der Gerätesicht auf mindestens 200 % einstellen, sehen Sie bei Ein-/Ausgabemodulen die einzelnen E/A-Kanäle. Wenn Sie für die Kanäle PLC-Variablen definiert haben, werden die Namen der PLC-Variablen angezeigt.

Das folgende Bild zeigt die Eingabekanäle eines Digitaleingabemoduls mit den beiden PLC-Variablen "Test1" und "Test2" bei einer Zoomstufe von 400 %.

Sie können die einzelnen E/A-Kanäle selektieren und haben bei Kanälen mit PLC-Variablen folgende Möglichkeiten:

- Im Inspektorfenster unter "Eigenschaften" sehen Sie die allgemeinen Eigenschaften der selektierten PLC-Variable.
- Im Inspektorfenster unter "Info > Querverweise" finden Sie Querverweis-Informationen zu der selektierten PLC-Variable. Sie können die Querverweis-Informationen bei selektiertert PLC-Variable auch über das Kontextmenü öffnen.

Siehe auch

Gerätesicht (Seite 14)

1.1.6 Gerätesicht: Ablage nicht gesteckter Baugruppen

In manchen Fällen werden Baugruppen einer Hardware-Konfiguration temporär keinem Steckplatz zugewiesen. Solche nicht gesteckten Baugruppen werden in die Ablage nicht gesteckter Baugruppen geschoben, einem speziellen Bereich in der Gerätesicht.

Aufnahme von Baugruppen in die Ablage

In die Ablage nicht gesteckter Baugruppen werden automatisch die Baugruppen geschoben, die z. B. über eine Kopieraktion einem Gerät zugeordnet werden sollten, der entsprechende Baugruppenträger jedoch keinen freien kompatiblen Steckplatz aufweist.

Unter folgenden Bedingungen werden Baugruppen automatisch in die Ablage nicht gesteckter Baugruppen aufgenommen:

- Eine Baugruppe wird in der Netzsicht auf ein Gerät gezogen, auf dem Baugruppenträger ist aber kein kompatibler Steckplatz frei.
- Eine Baugruppe wird in der Gerätesicht aus dem Baugruppenträger, dem Hardware-Katalog oder der Projektnavigation direkt in die Ablage geschoben oder kopiert.

CPs und FMs, die eine Netzressource belegen, können in die Ablage nicht gesteckter Baugruppen geschoben werden, verlieren aber ihre bereits zugewiesenen Netzressourcen.

Sie können Baugruppen auch selber beispielsweise über Drag & Drop in die Ablage nicht gesteckter Baugruppen einfügen. Dazu muss die Ablage geöffnet sein.

Verwendung der Ablage nicht gesteckter Baugruppen

Sie öffnen die Ablage nicht gesteckter Baugruppen mit der entsprechenden Schaltfläche.

Die Ablage nicht gesteckter Baugruppen finden Sie in der Gerätesicht.

Sie öffnen die Ablage nicht gesteckter Baugruppen über das entsprechende Symbol in der Funktionsleiste der Gerätesicht (Seite 14).

Symbol	Bedeutung
	Ablage nicht gesteckter Baugruppen öffnen

Hinweis

Um Steckplätze frei zu machen, schieben Sie Baugruppen aus Ihrer Konfiguration in die Ablage und stecken die gewünschten Baugruppen aus der Ablage in die nun freien Steckplätze.

Bereits parametrierte Baugruppen nehmen Sie auf diese Weise temporär aus der Konfiguration, ohne diese zu löschen.

Behandlung von Baugruppen in der Ablage

Für Baugruppen in der Ablage gelten folgende Regeln:

- Die Baugruppen finden Sie in der Projektnavigation unter dem entsprechenden Gerät im Ordner "Lokale Baugruppen".
- Die Baugruppen behalten alle bisher vorgenommenen Einstellungen und Parameter.
- Die Baugruppen werden beim Laden in das Zielsystem nicht berücksichtigt, es findet also keine Konsistenzprüfung für Baugruppen in der Ablage nicht gesteckter Baugruppen statt.
- Die Baugruppen können über das Kontextmenü z. B. kopiert, ausgeschnitten oder gelöscht werden.

1.1.7 Topologiesicht

Einführung

Die Topologiesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Ethernet-Topologie anzeigen
- Ethernet-Topologie projektieren
- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren
- Namen der Geräte bearbeiten

Aufbau

Das folgende Bild gibt Ihnen einen Überblick über die Topologiesicht.

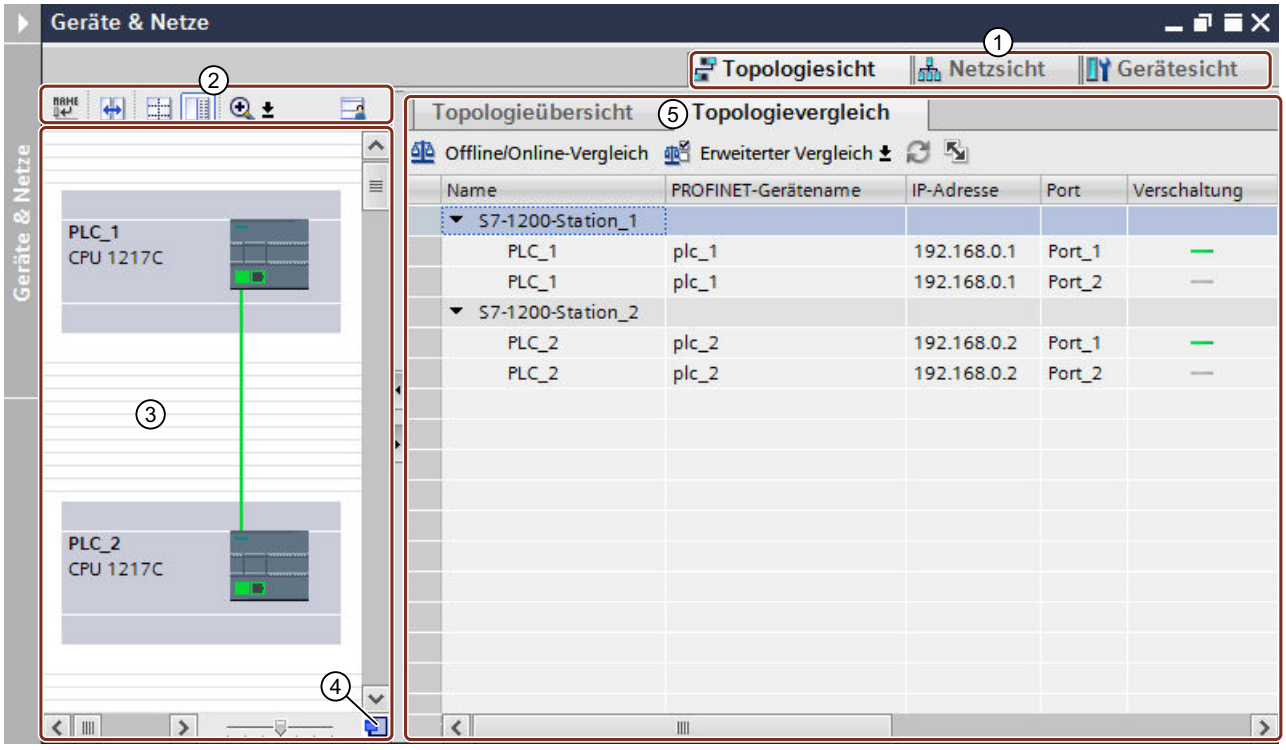

① Umschalter Gerätesicht/Netzsicht/Topologiesicht
② Funktionsleiste der Topologiesicht
③ Grafischer Bereich der Topologiesicht
④ Übersichtsnavigation
⑤ Tabellarischer Bereich der Topologiesicht (Topologieübersicht)

Die Teilung zwischen grafischem und tabellarischem Bereich der Topologiesicht können Sie mit der Maus verändern. Die Teilung kann über eine Schaltfläche in der Funktionsleiste zwischen horizontaler und vertikaler Teilung umgeschaltet werden. Um die Position der Teilung zu verändern, klicken Sie mit der Maus zwischen den grafischen und tabellarischen Bereich und verändern Sie bei gedrückter Maustaste durch einer Verschieben des Teilers nach Links oder Rechts die Aufteilung. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

Symbol	Bedeutung
	Öffnet den Dialog zur manuellen Namensvergabe für PROFINET-Geräte. Dazu muss das IO-Device gesteckt und online mit dem IO-System verbunden sein.
	Verändert die Position der Geräte in der Topologiesicht so, dass die neue Position der Position in der Netzsicht möglichst nahe kommt.

Symbol	Bedeutung
	Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.
	Über das Zoomsymbol können Sie die Ansicht schrittweise vergrößern (+) oder verkleinern (-) oder einen Rahmen um einen zu vergrößernden Bereich ziehen.
	Schaltet die Teilung von grafischem und tabellarischem Bereich der Editorsicht zwischen horizontal und vertikal um.
	Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.

Grafischer Bereich

Im grafischen Bereich der Topologiesicht werden Ihnen Ethernet-Baugruppen mit den zugehörigen Ports und Portverschaltungen angezeigt. Hier können Sie weitere Hardware-Objekte mit Ethernet-Schnittstellen hinzufügen. Siehe: Gerät zur Hardware-Konfiguration hinzufügen (Seite 42)

Am unteren Rand des grafischen Bereichs finden Sie Bedienelemente für die Ansichtssteuerung:

- Mit der Klappliste wählen Sie die Zoomstufe aus. Sie können auch direkt einen Wert in das Feld der Klappliste eintragen.
- Sie können die Zoomstufe auch mit dem Schieberegler einzustellen.
- Über das Icon in der rechten unteren Ecke können Sie das Fenster des grafischen Bereichs neu fokussieren.

Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

Tabellarischer Bereich

Der tabellarische Bereich besteht aus den folgenden beiden Registern:

- Register "Topologieübersicht"
- Register "Topologievergleich"

Im Register "Topologieübersicht werden Ihnen die Ethernet- bzw. PROFINET-Baugruppen, deren Ports und Portverschaltungen in tabellarischer Form angezeigt. Diese Tabelle entspricht der Netzübersichtstabelle in der Netzsicht.

Im Register "Topologievergleich" ermitteln Sie die Unterschiede zwischen Soll- und Ist-Topologie und minimieren sie.

Siehe auch

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

Übersicht der Einstellungen zur Hardware-Konfiguration (Seite 10)

1.1.8 Inspektorfenster

Die angezeigten Eigenschaften und Parameter des ausgewählten Objekts können innerhalb des Inspektorfensters bearbeitet werden.

Aufbau

Das Inspektorfenster besteht aus folgenden Komponenten:

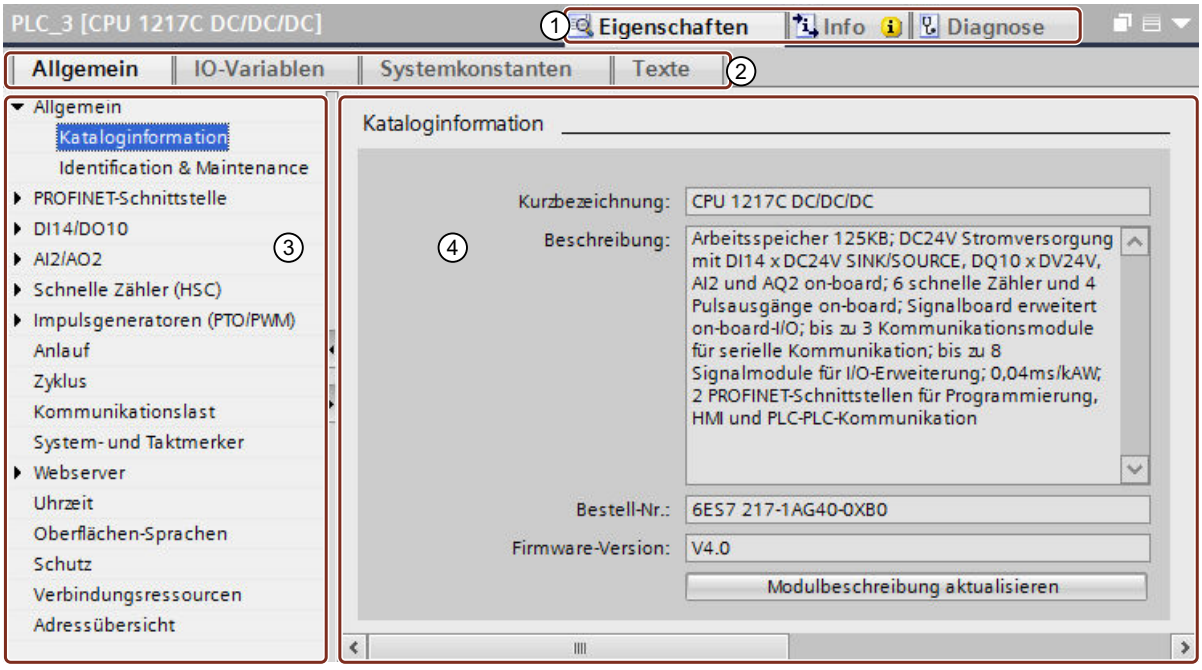

① Umschaltung zwischen verschiedenen Informations- und Arbeitsbereichen

② Umschaltung zwischen verschiedenen Registern der Informations- und Arbeitsbereiche

③ Navigation zwischen verschiedenen Informationen und Parametern

④ Anzeige der ausgewählten Informationen und Parameter

Beschreibung der Funktionen

Die Informationen und Parameter im Inspektorfenster sind in verschiedene Informationstypen mit verschiedenen Registern aufgeteilt:

- Eigenschaften
  In diesem Bereich finden Sie die Möglichkeit, das Gerät zu konfigurieren. Sie können die Einstellungen für die Hardware vornehmen, Variablen und Systemkonstanten bearbeiten und Projekttexte verwalten. Sie finden hier die folgenden Registerkarten:
  - Allgemein (siehe Eigenschaften und Parameter (Seite 64))
  - IO-Variablen (siehe AUTOHOTSPOT)
  - Systemkonstanten (siehe AUTOHOTSPOT)
  - Texte (siehe AUTOHOTSPOT)
- Info
  In diesem Bereich finden Sie allgemeine Informationen zu dem Gerät im Projekt sowie eine Querverweisliste und Meldungen und Hinweise zur Übersetzung des Geräts. Sie finden hier die folgenden Registerkarten:
  - Allgemein
  - Querverweisliste (siehe Querverweise im Inspektorfenster)
  - Übersetzen (siehe Informationen zum Übersetzen)
- Diagnose
  Wenn Sie mit dem Gerät online gehen, finden Sie hier die Diagnoseinformationen zum Gerät und seinen Verbindungen. Entsprechende Diagnosemeldungen werden Ihnen auch angezeigt. Sie finden hier die folgenden Registerkarten:
  - Geräte-Information (siehe Hardware-Diagnose (Seite 3475))
  - Verbindungsinformation (siehe Verbindungsdiagnose (Seite 3536))
  - Meldungsanzeige (siehe Meldungsanzeige)

Für die Anzeige der entsprechenden Informationen und Parameter klicken Sie auf den gewünschten Bereich.

Anzeige der Geräteeigenschaften

Für das Konfigurieren eines Automatisierungssystems ist der Bereich "Eigenschaften" der wichtigste Bereich, aus diesem Grund werden die Bearbeitungsmöglichkeiten hier gesondert beschrieben. Für Informationen zu den anderen Bereichen beachten Sie bitte die oben genannten Verlinkungen auf andere Stellen des Informationssystems.

Registerkarte "Allgemein"

Anzeige der Eigenschaften und Einstellungen der des Geräts oder des Moduls. Hier können Sie die Einstellungen und Parameter bearbeiten. Im linken Teil des Inspektorfensters befindet sich die Bereichsnavigation. Informationen und Parameter sind dort in Gruppen angeordnet. Wenn Sie auf das Pfeilsymbol links neben dem Gruppennamen klicken, können Sie die Gruppe bei vorhandenen Untergruppen aufklappen. Wenn Sie eine Gruppe oder eine Untergruppe auswählen, werden die entsprechenden Informationen und Parameter im rechten Teil des Inspektorfensters angezeigt und können dort auch bearbeitet werden.

Registerkarte "IO-Variablen"

Allgemein | IO-Variablen | Systemkonstanten | Texte

Name	Typ	Adresse	Variablentabelle	Kommentar
Start	Bool	%I0.0	Standard-Variablentabelle	
Stop	Bool	%I0.1	Standard-Variablentabelle	
Check	Bool	%I0.2	Standard-Variablentabelle	
	Bool	%I0.3		

en für die
erten
/ariablen

Sie können auf die Variablensymbole in der Tabelle klicken um Variablen per Drag & Drop in anderen Editoren zu verwenden. Sie können damit zum Beispiel eine Variable einer Anweisung im Programmiereditor zuweisen oder eine Variable in die Variablentabelle eines anderen Geräts kopieren.

Allgemein | IO-Variablen | Systemkonstanten | Texte

Name	Typ	Adresse	Variablentabelle
X23-08/15	Bool	%I8.0	Standard-Variablentabelle

Allgemein | IO-Variablen | Systemkonstanten | Texte

Name	Typ	Adresse	Variablentabelle
X23-08/15	Bool	%I8.0	Standard-Variablentabelle
X23-08/16	Bool	%I8.1	Standard-Variablentabelle
X23-08/17	Bool	%I8.2	Variablentabelle_1
	Bool	%I8.3	
	Bool	%I8.4	

öchten, können Sie auch den
en unteren Ecke über weitere
nen Feldern wird eine neue
Name der Ursprungsvariable
e bereits auf eine Zahl, wird

Allgemein | IO-Variablen | Systemkonstanten | Texte

Name	Typ	Adresse	Variablentabelle
X23-08/15	Bool	%I8.0	Standard-Variablentabelle
X23-08/16	Bool	%I8.1	Standard-Variablentabelle
X23-08/17	Bool	%I8.2	Variablentabelle_1
X23-08/18	Bool	%I8.3	Variablentabelle_1
X23-08/19	Bool	%I8.4	Variablentabelle_1
	Bool	%I8.5	

ir die Variablentabelle neue
riablentabelle zuweisen.
nen Sie den Rahmen wie oben
len in der entsprechenden

Sind von diesen Aktionen bereits vorhandene Variablen betroffen, werden Sie über ein Dialogfenster gefragt, ob die betroffenen Variablen überschrieben oder ob nur in leeren Feldern neue Variablen eingefügt werden sollen. Für den zweiten Fall bedeutet dies, dass beispielsweise bei einem Bereich von neun Feldern mit sechs vorhandenen Variablen nur drei Variablen hinzugefügt werden und die sechs vorhandenen Variablen werden nicht verändert.

Register "Systemkonstanten"

Allgemein | IO-Variablen | Systemkonstanten | Texte

Hardware-Systemkonstanten anz

Name	Typ	HW-Kennung	Verwendet von	Kommentar
Local~MC	Hw_SubModule	51	PLC_1	
Local~Common	Hw_SubModule	50	PLC_1	
Local~Device	Hw_Device	32	PLC_1	

Bei I-Slaves oder I-Devices werden auch die den I-Slave oder das I-Device betreffende Systemkonstanten des DP-Masters oder des IO-Controllers angezeigt.

Wenn Sie in der Spalte, in der die Verwendung der Konstante angezeigt wird, auf den Link klicken, wird die Gerätesicht des entsprechenden Gerätes geöffnet. Die Systemkonstanten werden auch in der PLC-Variablentabelle dargestellt.

Sie können auf die Konstantensymbole in der Tabelle klicken um Systemkonstanten per Drag & Drop in anderen Editoren zu verwenden. Sie können damit zum Beispiel eine

Systemkonstante einer Anweisung im Programmiereditor zuweisen oder eine Systemkonstante als Anwenderkonstante in einem anderen Gerät anlegen.

Registerkarte "Texte"

Anzeige der Referenzsprache und Angabe der Textquelle für die Projekttexte. Beachten Sie hierzu die weiteren Inhalte des Informationssystems.

Siehe auch

Übersicht über den Hardware- und Netzwerkeditor (Seite 9)

Adressierung von Baugruppen (Seite 1458)

Inspektorfenster: Darstellung von UDTs (Seite 27)

1.1.9 Inspektorfenster: Darstellung von UDTs

Im Inspektorfenster werden selbst definierte und aus mehreren Elementen zusammengesetzte Datenstrukturen unter "Eigenschaften > IO-Variablen" angezeigt. Die Darstellung dieser UDT (User Defined Structure) genannten PLC-Datentypen unterscheidet sich von der Darstellung einfacher IO-Variablen.

Darstellung von UDTs in der Tabelle der IO-Variablen

Strukturierte PLC-Variablen, die den Adressraum eines Moduls belegen, werden mit ihren enthaltenen Datenelementen über kombinierte Spalten angezeigt. Belegte Adressbereiche werden dabei durch senkrechte Linien und einzelne Adressen durch kleine Rauten dargestellt.

Abhängig von der Belegung der Kanaladressen der IO-Module durch den Adressbereich der UDTs werden folgende Symbole für die Darstellung verwendet:

Symbol	Bedeutung
	Beginn des über mehrere Kanäle gehenden Adressbereichs eines Datenelements. Der Adressbereich des ganzen Kanals wird verwendet.
	Teil des über mehrere Kanäle gehenden Adressbereichs eines Datenelements. Der Adressbereich des ganzen Kanals wird verwendet.
	Ende des über mehrere Kanäle gehenden Adressbereichs eines Datenelements. Der Adressbereich des ganzen Kanals wird verwendet.
	Kompletter Adressbereich eines Datenelements über den Adressbereich eines vollständig verwendeten Kanals. Beispiel: Nur das Eingangswort eines Kanals wird vollständig für ein Datenelement verwendet.
	Beginn des über mehrere Kanäle gehenden Adressbereichs eines Datenelements. Der Adressbereich des Kanals wird nur teilweise verwendet.
	Ende des über mehrere Kanäle gehenden Adressbereichs eines Datenelements. Der Adressbereich des Kanals wird nur teilweise verwendet.
	Kompletter Adressbereich eines Datenelements. Nur der erste Teil der Kanaladresse wird verwendet. Beispiel: Nur das erste Byte eines Worts ist belegt.
	Kompletter Adressbereich eines Datenelements. Nur der zweite Teil der Kanaladresse wird verwendet. Beispiel: Nur das zweite Byte eines Worts ist belegt.

Symbol	Bedeutung
◆	Einzelne Adresse eines Datenelements für einen Kanal, der nur eine Adresse hat. Beispiel: Einzelner Bitzugriff auf die Kanaladresse eines Digitaleingangs.
◇	Einzelne Adresse eines Datenelements für einen Kanal, der mehrere Adressen hat. Beispiel: Einzelner Bitzugriff auf die Kanaladresse eines Analogeingangs.

Anwendungsbeispiel

Sehen Sie im folgenden Bild eine Variablentabelle mit einem PLC-Datentyp UDT "Motor" mit den darin enthaltenen Datenelementen "Start", "Stop" und "Dummy":

	Name	Datentyp	Adresse
1	Start	Bool	%I2.0
2	Stop	Bool	%I2.1
3	ErrorCode	Byte	%IB3
4	Motor	"MotorUDT"	%I0.0
5	Dummy	Word	%IW0
6	Start	Bool	%I2.0
7	Stop	Bool	%I2.1
8	Status	Array[0..7] of B...	%I4.0
9	Status[0]	Bool	%I4.0
10	Status[1]	Bool	%I4.1

Bei einem Eingabemodul DI32 mit den Eingabeadressen I0.0 bis I3.7 sehen Sie im Inspektorfenster unter den IO-Variablen die kombinierte Spalte "Motor ("MotorUDT")". Diese kombinierte Spalte enthält eine Spalte für den Adressbereich der UDT "Motor" und eine Spalte für die in der UDT enthaltenen Datenelemente "Motor.Dummy", "Motor.Start" und "Motor.Stop". Das Datenelement "Status" beginnt erst ab Adresse I4.0 und wird bei den IO-Variablen für dieses Eingabemodul nicht angezeigt, da der Adressbereich der 32 Digitalkanäle nur von I0.0 bis I3.7 geht. Die Variable "ErrorCode (Byte)" ist kein Teil der UDT, daher wird sie nicht in der kombinierten Spalte unter UDT "Motor" angezeigt, sondern in einer separaten Spalte dargestellt:

Allgemein | IO-Variablen | Systemkonstanten | Texte

Name	Typ	Adresse	Variablentabelle	Motor ("MotorUDT")		ErrorCode (Byte)
	Bool	%I0.0		Motor	Motor.Dummy	
	Bool	%I0.1				
	Bool	%I0.2				
	⋮	⋮		⋮	⋮	
	Bool	%I1.7				
Start	Bool	%I2.0	Standard-Variablentabelle		Motor.Start	
Stop	Bool	%I2.1	Standard-Variablentabelle		Motor.Stop	
	Bool	%I2.2				
	⋮	⋮		⋮		
	Bool	%I2.7				
	Bool	%I3.0				
	Bool	%I3.1				
	Bool	%I3.2				
	⋮	⋮		⋮		⋮
	Bool	%I3.7				

Die UDT "Motor" belegt in diesem Modul den Adressbereich von I0.0 bis I3.7. Innerhalb dieses Adressbereiches belegt die Variable "Dummy" den Adressbereich I0.0 bis I1.7 und die beiden

Variablen "Start" und "Stop" belegen die einzelnen Adressen I2.0 und I2.1. Die nicht zur UDT gehörende Variable "ErrorCode (Byte)" belegt in einer eigenen Spalte die Adressen I3.0 bis I3.7.

Hinweis

Die Tabelle der IO-Variablen zeigt nur die Daten innerhalb des Adressbereichs des ausgewählten Geräts an, in diesem Fall den Adressbereich des Eingabemoduls DI32. Wenn ein weiteres Eingabemodul mit mindestens 16 Kanälen gesteckt wird, wird mit dem erweiterten Adressbereich die Anzeige der Adressbelegung durch die UDT fortgeführt: Der Adressbereich der UDT "Motor" geht bis Adresse I5.7 und das darin enthaltene Datenelement Status geht von Adresse I4.0 bis I5.7. Das Datenelement "Status" wiederum belegt noch von I4.0 bis I4.7 einzelne Adressen für einzelne Statusbits. Für die Anzeige der Statusbits wird die kombinierten Spalte der UDT um eine weitere Spalte ergänzt.

Siehe auch

Inspektorfenster (Seite 24)

1.1.10 Hardware-Katalog

Die Task Card "Hardware-Katalog" ermöglicht Ihnen einen einfachen Zugriff auf verschiedenste Hardware-Komponenten.

Aufbau

Die Task Card "Hardware-Katalog" besteht aus den folgenden Paletten:

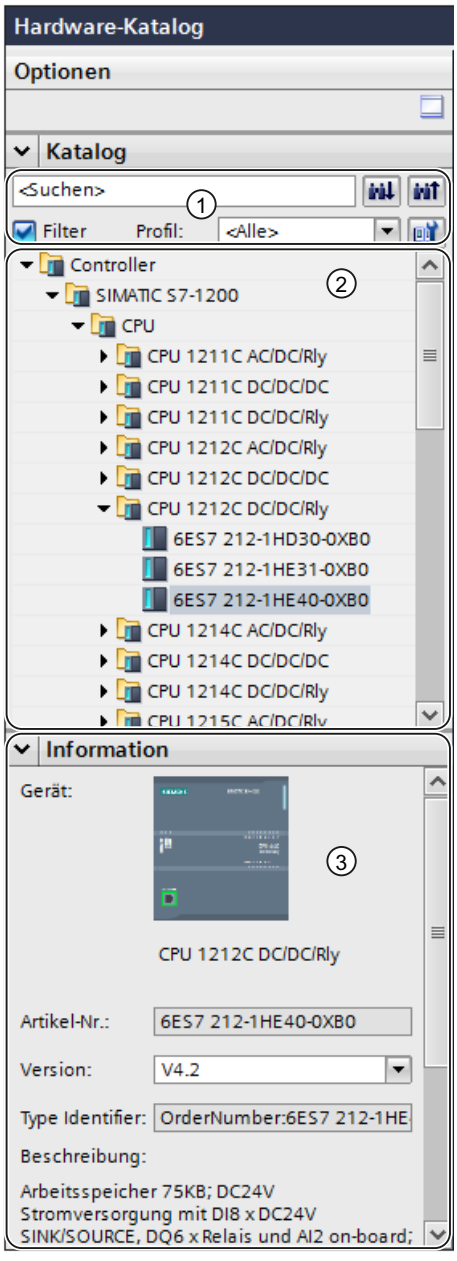

① Palette "Katalog", Such- und Filterfunktion

② Palette "Katalog", Komponentenauswahl

③ Palette "Information"

Such- und Filterfunktion

Die Such- und Filterfunktionen der Palette "Katalog" erlauben Ihnen eine komfortable Suche nach bestimmten Hardware-Komponenten. Nach eingebenem Suchtext können Sie von der aktuellen Katalogposition aus eine Suche mit der Suchrichtung nach oben oder unten starten.

Über die Filterfunktion können Sie die Anzeige der HW-Komponenten nach bestimmten Kriterien beschränken. Beispielsweise können Sie die Anzeige auf Objekte beschränken, die Sie auch im aktuellen Kontext platzieren können oder die bestimmte Funktionen beinhalten. Verwendbare Objekte im aktuellen Kontext sind z. B. vernetzbare Objekte in der Netzsicht oder nur zum Gerät kompatible Baugruppen in der Gerätesicht.

Sie können Hardware-Profile erstellen, mit denen die angezeigten Objekte im Hardware-Katalog eingeschränkt werden. Beispielsweise könnten Sie ein Hardware-Profil erstellen, in dem nur bestimmte Controller zu sehen sein sollen und ein weiteres Profil, in welchem nur bestimmte Geräte der Dezentralen Peripherie enthalten sein sollen. Sie können zwischen den angelegten Hardware-Profilen umschalten.

Komponentenauswahl

Die Komponentenauswahl der Palette "Katalog" enthält die installierten Hardware-Komponenten in einer Baumstruktur. Aus dem Katalog können Sie die gewünschten Geräte oder Baugruppen in den grafischen Arbeitsbereich der Geräte- oder Netzsicht ziehen.

Hardware-Komponenten, die zwar installiert sind, aber für die keine Lizenz vorhanden ist, werden gegraut angezeigt. Nicht-lizenzierte HW-Komponenten können Sie nicht verwenden.

Hardware-Komponenten, die thematisch zu verschiedenen Komponentengruppen gehören, sind zum Teil als verlinktes Objekt ausgeführt. Wenn Sie solch eine verlinkte Hardware-Komponente anklicken, öffnet sich die Katalogstruktur, in der Sie die entsprechende Hardware-Komponente finden.

Information

In der Palette "Information" sehen Sie Detail-Informationen des im Katalog selektierten Objekts:

- Schematische Darstellung
- Name
- Artikelnummer
- Versionsnummer
- Beschreibung
- Type Identifier (nur wenn aktiviert)

Hinweis

Type Identifier

Sie können im Menü "Extras > Einstellungen > Hardware-Konfiguration" die Anzeige des Attributes "Type Identifier" aktivieren. Ist der Type Identifier aktiviert, können Hardware-Objekte für die Verwendung in externen Entwicklungsumgebungen eindeutig identifiziert werden. Der Type Identifier wird unter anderem im Hardware-Katalog in der Palette "Informationen" angezeigt.

Siehe auch

Hardware-Katalog durchsuchen (Seite 32)

Übersicht über den Hardware- und Netzwerkeditor (Seite 9)

1.1.11 Hardware-Katalog durchsuchen

Einführung

Um für eine Hardware-Konfiguration die gewünschten Hardware-Komponenten auszuwählen, verwenden Sie die Task Card "Hardware-Katalog". Verwenden Sie den Hardware-Katalog für die Auswahl der vernetzbaren Hardware-Komponenten in der Netz- und Topologiesicht und für die Auswahl der gewünschten Baugruppen in der Gerätesicht.

Kontextfilter

Über die Option "Filter" des Hardware-Katalogs können Sie die Anzahl der angezeigten und durch die Suche auffindbaren HW-Komponenten einschränken.

Wenn Sie den Filter aktivieren, werden die im Hardware-Katalog verfügbaren Komponenten in Abhängigkeit von der gewählten Sicht und dem gewählten Gerät angezeigt. Wenn Sie zwischen den verschiedenen Sichten wechseln, wird die Ansicht der gefilterten Objekte an den aktuellen Kontext angepasst. Ist der Filter nicht aktiviert, wird der gesamte Hardware-Katalog in allen Sichten angezeigt.

Beispiel: Bei aktiviertem Filter in der Gerätesicht einer S7-1200 CPU werden im Hardware-Katalog nur S7-1200 CPUs und dazu passende Module angezeigt.

Suchmöglichkeiten

Über die Suchfunktion suchen Sie gezielt nach Einträgen im Hardware-Katalog. Beachten Sie bei der Eingabe von Suchbegriffen folgende Regeln:

- Zwischen Groß- und Kleinschreibung wird nicht unterschieden.
- Bindestriche und Leerzeichen werden bei der Suche ignoriert.
- Teile eines gesuchten Begriffs werden bei der Suche berücksichtigt.

- Mehrere Suchbegriffe müssen mit einem Leerzeichen getrennt werden.
- Das Sternchen "*" wird als Platzhalter verwendet.

Hinweis

Verwendung von Wildcards im HW-Katalog

In der Sucheingabe wird das Sternchen bzw. der Asterisk "*" als Platzhalter ("Wildcard") verwendet. Eine Suche nach "1A**0" findet zum Beispiel auch Einträge mit "1AD30" oder "1AE40". Umgekehrt gibt es im Hardware-Katalog auch Baugruppen, deren Artikelnummer Wildcards beinhalten. Die Wildcards der verschiedenen Artikelnummern können neben dem Sternchen auch die klein geschriebenen Buchstaben "x", "y" und "z" sein. Wenn Sie zum Beispiel in das Suchfeld die konkrete Artikelnummer "3RW4 422-1BC34" eingeben, würden Ihnen auch Artikel mit der Nummer "3RW4 422-*BC**" oder "3RW4 4xx-1By3z" angezeigt werden.

Sie starten die Suche von einem aktuell markierten Objekt im Hardware-Katalog mit der Suchrichtung nach oben oder nach unten.

Symbol	Bedeutung
	Suche abwärts
	Suche aufwärts

Hardware-Katalog durchsuchen

Wenn Sie den Hardware-Katalog durchsuchen möchten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der Maus in das Eingabefeld der Suchfunktion
2. Geben Sie einen Suchbegriff ein. Folgende Elemente werden durchsucht:
   - Name des Geräts oder der Baugruppe
   - Artikelnummer (MLFB)
   - Beschreibung in Palette "Information"
3. Klicken Sie auf eine der Schaltflächen "Suche abwärts" bzw. "Suche aufwärts".

   Hinweis

   Für die richtige Suchrichtung beachten Sie, welche Stelle Sie im Hardware-Katalog markiert haben. Um alles zu durchsuchen, klicken Sie auf das oberste Objekt des Hardware-Katalogs und starten Sie die Suche nach Eingabe des Suchbegriffs mit "Suche abwärts".

   Die erste gefundene Übereinstimmung mit dem Suchbegriff wird als Ergebnis angezeigt. Für weitere Suchergebnisse klicken Sie weitere Male auf die Schaltfläche "Suche abwärts" bzw. "Suche aufwärts".

   Beachten Sie den Kontextfilter des HW-Kataloges. Ist dieser eingeschaltet, beschränkt sich die Suche im HW-Katalog nur auf die angezeigten steckbaren HW-Komponenten.

Siehe auch

Hardware-Katalog (Seite 29)

1.1.12 Produktsupport aktivieren

Sie können sich zu jedem Gerät des Hardware-Katalogs weitere Informationen anzeigen lassen, die im Siemens Industry Online Support hinterlegt sind. Standardmäßig ist die Funktion deaktiviert. Im Folgenden erfahren Sie, wie Sie die Funktion aktivieren.

Voraussetzung

Das TIA Portal verfügt über Zugriff auf das Internet.

Vorgehen

Um den Zugriff auf den Siemens Industry Online Support zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Öffnen Sie in der Bereichsnavigation die Gruppe "Hardware-Konfiguration".
3. Aktivieren Sie das Optionskästchen "Über Internet".

Ergebnis

Innerhalb des Hardware-Katalogs erhalten Sie im Kontextmenü einer Baugruppe Zugriff auf den Produktsupport, die FAQs und die Handbücher.

Siehe auch

Produktsupport zu Hardware-Komponenten anzeigen (Seite 34)

Übersicht der Einstellungen zur Hardware-Konfiguration (Seite 10)

1.1.13 Produktsupport zu Hardware-Komponenten anzeigen

Im Hardware-Katalog haben Sie direkten Zugriff auf Informationen, die zu jedem Modul im Siemens Industry Online Support hinterlegt sind. Folgende Direkteinsprünge zu den jeweiligen Seiten im Siemens Industry Online Support stehen zur Verfügung:

- Informationen zum Produktsupport
- FAQs
- Handbücher

Voraussetzung

- Sie haben Zugriff auf das Internet.
- Der Zugriff auf den Produktsupport ist in den Einstellungen des TIA Portals aktiviert.
  Wie sie die Funktion aktivieren, lesen Sie im Kapitel "Produktsupport aktivieren (Seite 34)".

Vorgehen

Um die Informationen im Siemens Industry Online Support zu einem bestimmten Modul anzuzeigen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zum gewünschten Modul.
2. Klicken Sie mit der rechten Maustaste auf das Modul.
3. Wählen Sie im Kontextmenü einen der folgenden Einträge:
   - Informationen zum Produktsupport
   - FAQs
   - Handbücher

Ergebnis

Der im Betriebssystem eingestellte Standard-Browser wird geöffnet und die passende Seite im Siemens Industry Online Support wird geladen.

Siehe auch

Produktsupport aktivieren (Seite 34)

Übersicht der Einstellungen zur Hardware-Konfiguration (Seite 10)

1.1.14 Drucken von Hardware- und Netzkonfigurationen

Ausdruck von Hardware- und Netzwerkkonfigurationen

Im Rahmen der Projektdokumentation können Sie die folgenden Elemente der Hardware- und Netzsicht mit ausdrucken:

- Grafische Netzsicht
- Netzübersichtstabelle
- Grafische Gerätesicht
- Die Geräteübersichtstabelle
- Die Parameter des aktuell im Editor selektierten Objekts

Ausdruck der Inhalte eines Editors

Wenn Sie den Druck innerhalb eines geöffneten Editor starten und keine Baugruppe ist selektiert, werden immer die Inhalte des Editors gedruckt. Dazu gehört die grafische Darstellung des Editors wie auch die dem Editor zugehörige Tabelle. Sie können den Druckumfang jedoch anpassen. Sie können einstellen, ob nur die grafische Sicht, nur die Tabelle oder beides zusammen gedruckt werden sollen. Lesen Sie dazu das Kapitel "Drucken: Einstellungen ändern (Seite 37)".

Ist die Grafik größer als das gewählte Seitenlayout, wird der Ausdruck auf der nächsten Seite fortgesetzt. So gehen keine Inhalte verloren. Alternativ können Sie die Zoomstufe der grafischen Darstellung ändern, um den Ausdruck auf eine Seite anzupassen. Der Ausdruck erfolgt immer in der aktuell gewählten Zoomstufe.

Zur Kontrolle, ob alle Inhalte auf eine Seite passen, können Sie entweder die Druckvorschau nutzen oder Sie aktivieren die Seitenumbruchvorschau. Bei aktivierter Seitenumbruchvorschau werden innerhalb des grafischen Editors gestrichelte Linien an der Stelle angezeigt, wo später ein Seitenumbruch erfolgt.

Druck von sehr großen Tabellen

Ist eine Tabelle größer als die Druckfläche und kann somit nicht vollständig gedruckt werden, so werden die Inhalte der Tabelle nicht in tabellarischer Form gedruckt, sondern als Paare zwischen Wert und Schlüssel.

Beispiel:

Objektname	Eigenschaft 1	Eigenschaft 2
Objekt A	Wert A1	Wert A2
Objekt B	Wert B1	Wert B2

In diesem Fall sieht der Ausdruck wie folgt aus:

Objekt A

Eigenschaft 1: Wert A1

Eigenschaft 2: Wert A2

Objekt B

Eigenschaft 1: Wert B1

Eigenschaft 2: Wert B2

Sie können Sie diese Druckform auch voreinstellen, so dass Tabellen immer als Paar zwischen Schlüssel und Wert gedruckt werden. Lesen Sie dazu das Kapitel "AUTOHOTSPOT".

Ausdruck von Baugruppenparametern

Parameter von selektierten Baugruppen werden zusammen mit den aktuell eingestellten Werten in Textform ausgedruckt. Alle Parameter von zugehörigen Baugruppen werden ebenfalls mit ausgedruckt. Haben Sie beispielsweise eine CPU selektiert, werden die Parameter eines eventuell gesteckten Signal Boards mit gedruckt.

Sie können bestimmen, in welchem Umfang die Parameter von Baugruppen gedruckt werden. Im Dialog "Drucken" können Sie wählen, ob Sie alle Eigenschaften und Parameter einer

Baugruppe drucken möchten, oder ob Sie einen Kompakt-Ausdruck wünschen. Wählen Sie die kompakte Form, werden nur die Einträge gedruckt, die im Bereich "Allgemein" in den Eigenschaften der Baugruppe eingetragen sind. Davon ausgenommen sind Kommentare zur Baugruppe sowie der Autor und die Baugruppenbeschreibung. Im Kompakt-Modus werden also beispielsweise die folgenden Parameter einer Baugruppe gedruckt:

- Angaben zur Baugruppe
  Name, Baugruppensteckplatz, Kurzbeschreibung, Artikelnummer, Firmware-Version
- Name der PROFINET-Schnittstelle
- Angaben zum Subnetz
  Name des Subnetzes, ID des S7-Subnetzes

Siehe auch

Drucken: Einstellungen ändern (Seite 37)

Drucken: Seitenumbruchvorschau aktivieren (Seite 38)

1.1.15 Drucken: Einstellungen ändern

Umfang des Ausdrucks ändern

Beim Drucken aus einem Editor heraus, können Sie festlegen, ob sowohl Grafiken als auch Tabellen gedruckt werden sollen, oder ob Sie nur eines von beiden ausdrucken möchten. Standardmäßig wird beides ausgedruckt.

Vorgehen

Um den Umfang des Ausdrucks zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Öffnen Sie in der Bereichsnavigation die Parametergruppe "Druckeinstellungen" unter "Allgemein".
3. Scrollen Sie bis zur Gruppe "Hardware-Konfiguration".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "Aktive grafische Sicht", je nachdem ob Sie die Grafiken der Netz- und Gerätesicht mit ausdrucken möchten.
5. Aktivieren oder deaktivieren Sie das Optionskästchen "Aktive Tabelle", je nachdem ob Sie die einem Editor zugehörige Tabelle mit ausdrucken möchten.

Siehe auch

Drucken von Hardware- und Netzkonfigurationen (Seite 35)

1.1.16 Drucken: Seitenumbruchvorschau aktivieren

In den grafischen Editoren können Sie eine Seitenumbruchvorschau für den Druck aktivieren. Ist die Option aktiviert, werden an den Stellen innerhalb des grafischen Editors gestrichelte Linien angezeigt, an denen beim späteren Ausdruck der Seitenumbruch erfolgt.

Vorgehen

Um die Seitenumbruchvorschau zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie den grafischen Bereich der entsprechenden Ansicht aus.
2. Klicken Sie auf das Symbol "Zeige Seitenumbrüche" in der Funktionsleiste des grafischen Editors.
   Innerhalb des grafischen Editors werden gestrichelte Linien an den Stellen angezeigt, wo später ein Seitenumbruch erfolgt.
3. Um das Rahmenlayout zu ändern, wählen Sie im Menü "Projekt" den Befehl "Drucken".
4. Um die Seitenumbruchvorschau zu deaktivieren, klicken Sie erneut auf das Symbol "Zeige Seitenumbrüche" in der Funktionsleiste des grafischen Editors,.

1.1.17 Tastaturbedienung: Navigation im Editor

Sie können zwischen den Komponenten des Hardware- und Netzwerkeditor und dessen Objekten in der Netz- und Gerätesicht über Tastenkombinationen navigieren.

Zwischen Elementen und Funktionen navigieren

Funktion	Tastenkombination
Wechsel in die nächsttiefere Auswahlebene Mit <Return> können Sie zum Beispiel von einem selektierten Baugruppenträger in die tiefere Auswahleben der dort aufgeschnappten Geräte und Module wechseln. Ist ein Gerät selektiert, können Sie mit <Return> in die tiefere Auswahlebene der auf dem Gerät angezeigten Schnittstellen wechseln.	<Return>
Wechsel in die nächsthöhere Auswahlebene Mit <Esc> können Sie zum Beispiel von einer selektierten Schnittstelle in die höhere Auswahleben der Geräte und Module wechseln. Ist ein Gerät selektiert, können Sie mit <Esc> in die höhere Auswahlebene der Baugruppenträger wechseln.	<Esc>
Navigation zwischen Objekten in der aktuellen Auswahlebene Mit den Pfeiltasten können Sie in einer aktuellen Auswahlebene zwischen den Objekten navigieren. Zum Wechseln der Auswahlebene verwenden Sie die Tasten <Return> oder <Esc>.	<Pfeil-nach-oben> <Pfeil-nach-unten> <Pfeil-nach-rechts> <Pfeil-nach-links>
Schaltet auf die Gerätesicht um	<Strg+Shift+D>
Schaltet auf die Netzsicht um	<Strg+Shift+N>
Schaltet auf die Topologiesicht um	<Strg+Shift+T>

Funktion	Tastenkombination
Umschalten zwischen Editorelementen Mit der Taste <Tab> schalten Sie von einem Editorelement zu dem nächsten Element. Mit <Shift+Tab> schalten Sie zum vorigen Element um. Sie können so zum Beispiel zwischen grafischer Ansicht, Speedy Splitter, tabellarischer Ansicht oder unterlagerten Registerkarten umschalten.	<Tab> <Shift+Tab>
Umschalten zwischen den Registerkarten Mit den Tasten <Strg+Tab> schalten Sie von einer Registerkarte zu der nächsten rechten Registerkarte um. Mit <Strg+Shift+Tab> schalten Sie zur nächsten linken Registerkarte um. Sie können mit mit den Tasten zum Beispiel zwischen Gerätesicht, Netzsicht und Topologiesicht umschalten.	<Strg+Tab> <Strg+Shift+Tab>

Elemente und Funktionen öffnen

Funktion	Tastenkombination
Online- und Diagnosesicht öffnen Bei selektiertem Gerät öffnet sich für das selektierte Gerät mit <Strg+D> die Online- und Diagnosesicht.	<Strg+D>
Dialog zum Laden in Gerät öffnen Bei selektiertem Gerät öffnet sich für das selektierte Gerät mit <Strg+L> der Dialog für das erweiterte Laden.	<Strg+L>
Neues Gerät hinzufügen Mit <Strg+N> öffnet sich der Dialog für das Hinzufügen eines neuen Gerätes.	<Strg+N>
Öffnet die Task Card "Hardware-Katalog"	<Strg+Shift+C>
Öffnet die Task Card "Online-Tools"	<Strg+Shift+O>

1.1.18 Tastaturbedienung: Bearbeiten von Objekten

Sie können im Hardware- und Netzwerkeditor einige Funktionen der Netz- und Gerätesicht direkt über Kombinationen von Tastatur und Maus durchführen. Die Tastaturbedienung in Tabellen entspricht dem üblichen Verhalten. An dieser Stelle finden Sie die Tastaturbedienung für den grafischen Arbeitsbereich der Netz- und Gerätesicht.

Allgemeine Tastaturbedienung

Funktion	Tastenkombination
Ansicht auf gezogenen Rahmen vergrößern Ziehen Sie einen Rahmen in der grafischen Sicht, um die Größe der Ansicht entsprechend zu verändern.	<Strg+Leertaste> + gedrückte Maustaste
Ansicht verschieben Bewegen Sie den Mauszeiger, um die Ansicht zu verschieben.	<Leertaste> + gedrückte Maustaste
Aktuellen Vorgang abbrechen	<Esc>

Funktion	Tastenkombination
Verbinder lösen Mit <Esc> oder Doppelklick verlassen Sie den Verbindungsmodus beim Ziehen einer Verbindung.	<Esc> oder Doppelklick
Zoom in grafischer Sicht Die Vergrößerung oder Verkleinerung ist von der Drehrichtung abhängig.	<Strg> + Mausrad drehen

Ausgewählte Objekte

Funktion	Tastenkombination
Objekt selektieren	Mausklick
Objekt ausschneiden Das selektierte Objekt wird in die Zwischenablage kopiert und aus der grafischen Sicht gelöscht.	<Strg+X>
Objekt kopieren Das selektierte Objekt wird in die Zwischenablage kopiert.	<Strg+C>
Objekt einfügen Das Objekt aus der Zwischenablage wird in die Auswahl eingefügt.	<Strg+V>
Selektiertes Objekt löschen	<Entf>
Mehrere Objekte selektieren 1 Sie können mehrere Objekte einzeln durch Mausklick den selektierten Objekten hinzufügen. Alternativ können Sie mit <Shift > + gedrückter Maustaste einen Rahmen um die zu selektierenden Objekte ziehen.	<Shift > + Mausklick
Mehrere Objekte selektieren 2 Sie können mehrere Objekte einzeln durch Mausklick den selektierten Objekten hinzufügen. Alternativ können Sie mit <Shift > + gedrückter Maustaste einen Rahmen um die zu selektierenden Objekte ziehen. Bei gehaltener Taste <Strg> können Sie selektierte Objekte mit Mausklick deselektieren.	<Strg> + Mausklick
Auswahl verschieben Sie können mit gedrückter Maustaste Geräte oder Module auf erlaubte Steckplätze eines Baugruppenträgers ziehen.	gedrückte Maustaste
Auswahl kopieren Sie können mit <Strg> und gedrückter Maustaste Geräte oder Module auf erlaubte Steckplätze eines Baugruppenträgers ziehen. Die Geräte oder Module werden dabei kopiert.	<Strg> + gedrückte Maustaste

1.2 Geräte konfigurieren

1.2.1 Einführung zum Konfigurieren der Hardware

Um ein Automatisierungssystem aufzubauen, müssen Sie die einzelnen Komponenten der Hardware konfigurieren, parametrieren und miteinander verbinden. Die dafür notwendigen Arbeiten verrichten Sie in der Geräte- und Netzsicht.

Konfigurieren

Unter "Konfigurieren" versteht man das Anordnen, Einstellen und Vernetzen von Geräten und Baugruppen innerhalb der Geräte- oder Netzsicht. Baugruppenträger werden symbolisch repräsentiert. Wie "reale" Baugruppenträger lassen sie das Stecken einer festgelegten Anzahl von Baugruppen zu.

Jeder Baugruppe wird automatisch eine Adresse zugewiesen. Die Adressen können nachträglich geändert werden.

Beim Anlauf des Automatisierungssystems vergleicht die CPU die software-seitig erstellte Sollkonfiguration mit der tatsächlichen Istkonfiguration der Anlage. Eventuelle Fehler können somit sofort erkannt und gemeldet werden.

Parametrieren

Unter "Parametrieren" versteht man das Einstellen der Eigenschaften der verwendeten Komponenten. Dabei werden Hardware-Komponenten und Einstellungen für den Datenaustausch parametriert:

- Eigenschaften parametrierbarer Baugruppen
- Einstellungen für den Datenaustausch zwischen Komponenten

Die Parameter werden in die CPU geladen und im Anlauf von der CPU an die entsprechenden Baugruppen übertragen. Baugruppen lassen sich sehr einfach ersetzen, da die erstellten Parameter beim Anlauf automatisch in die neue Baugruppe geladen werden.

Anpassen der Hardware an Projekterfordernisse

Sie müssen Hardware konfigurieren, wenn Sie ein Automatisierungsprojekt aufbauen, erweitern oder verändern möchten. Dazu fügen Sie Ihrem Aufbau Hardware-Komponenten hinzu, verbinden diese mit bestehenden Komponenten und passen die Eigenschaften der Hardware den Aufgaben an.

Die Eigenschaften der Automatisierungssysteme und der Baugruppen sind so voreingestellt, dass Sie in vielen Fällen nichts mehr parametrieren müssen. In den folgenden Fällen ist das Parametrieren dennoch erforderlich:

- Sie möchten voreingestellte Parameter einer Baugruppe verändern.
- Sie möchten spezielle Funktionen verwenden.
- Sie möchten Kommunikationsverbindungen projektieren.

1.2.2 Gerät zur Hardware-Konfiguration hinzufügen

Einführung

Um ein vernetzbares Gerät der Hardware-Konfiguration hinzuzufügen, gibt es in der Netzsicht und der Topologiesicht folgende Möglichkeiten:

- Befehl "Neues Gerät hinzufügen" in der Projektnavigation
- Doppelklick auf Gerät im Hardware-Katalog
- Drag & Drop aus dem Hardware-Katalog in die Netzsicht bzw. in die Topologiesicht:
  - Texteintrag aus der Palette "Katalog"
  - Vorschaugrafik aus der Palette "Information"
- Befehl "Einfügen > Gerät" aus der Menüleiste der Netzsicht bzw. der Topologiesicht
- Kontextmenü eines Gerätes im Hardware-Katalog für "Kopieren" und "Einfügen"

Zusammen mit dem neuen Gerät wird ein passender Baugruppenträger angelegt. Das ausgewählte Gerät wird auf den ersten zulässigen Steckplatz des Baugruppenträgers gesteckt.

Unabhängig von dem gewählten Weg ist das hinzugefügte Gerät in der Projektnavigation und in der Netzsicht bzw. in der Topologiesicht des Hardware- und Netzwerkeditors sichtbar.

Gerät über die Projektnavigation hinzufügen

Um über die Projektnavigation ein Gerät der Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Befehl "Neues Gerät hinzufügen".
   Das Dialogfeld "Neues Gerät hinzufügen" wird geöffnet.
2. Lassen Sie sich das gewünschte Gerät in der Baumstruktur anzeigen:
   - Navigieren Sie in der Baumstruktur zu dem gewünschten Gerät.
   - Geben Sie im Eingabefeld einen Gerätenamen an.
3. Wählen Sie aus der Baumstruktur das gewünschte Gerät aus.
   Auf der rechten Seite des Dialogfelds werden Ihnen weitere Informationen zum aktuell selektierten Gerät angezeigt.
4. Stellen Sie ggf. den FW-Stand über die Klappliste des Dialogfeldes ein.
5. Markieren Sie das Optionskästchen "Gerätesicht öffnen", wenn Sie nach dem Hinzufügen des Gerätes sofort in die Gerätesicht wechseln wollen.
   Dort können Sie sofort mit der Gerätekonfiguration und Bestückung des Baugruppenträgers fortfahren.
6. Klicken Sie auf "OK", um das ausgewählte Gerät hinzuzufügen.
   Das Dialogfeld wird geschlossen.

Gerät aus dem Hardware-Katalog hinzufügen

Um über den Hardware-Katalog ein Gerät der Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Netzsicht bzw. die Topologiesicht.
2. Öffnen Sie den Hardware-Katalog.
3. Navigieren Sie im Hardware-Katalog zum gewünschten Gerät.
4. Selektieren Sie das ausgewählte Gerät mit einem Mausklick.
5. Stellen Sie ggf. den FW-Stand über die Klappliste des Hardware-Katalogs ein.
6. Ziehen Sie das Gerät mit Drag & Drop in die Netzsicht bzw. in die Topologiesicht.

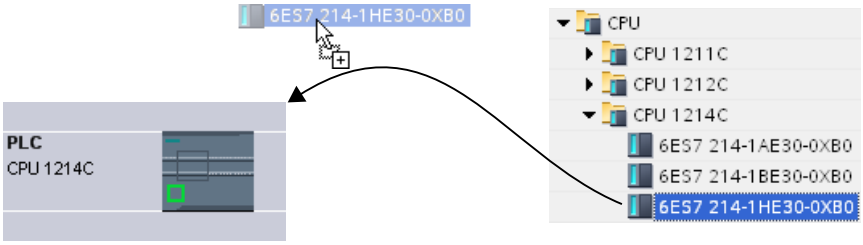

Sie haben nun das Gerät in der Netzsicht bzw. in der Topologiesicht platziert. Das angezeigte Rechteck (auch "Station") symbolisiert das gesteckte Gerät mit seinem Baugruppenträger und evtl. unterlagerten Baugruppen. Mit einem Doppelklick auf das Gerät oder die Station öffnen Sie die Gerätesicht und sehen dort den neuen Baugruppenträger mit dem gesteckten Gerät. In den nächsten Schritten können Sie das Gerät in der Gerätesicht konfigurieren und den Baugruppenträger mit Baugruppen bestücken.

Siehe auch

Netzsicht (Seite 11)

Nicht spezifizierte CPU hinzufügen (Seite 43)

Topologiesicht (Seite 21)

1.2.3 Nicht spezifizierte CPU hinzufügen

Einführung

Wenn Sie noch keine CPU ausgewählt haben aber bereits mit der Programmierung beginnen oder ein bestehendes Programm verwenden möchten, haben Sie die Möglichkeit, eine nicht spezifizierte CPU zu verwenden. Sie können auch bei unspezifizierten CPUs einige Einstellungen vornehmen. Die Einstellmöglichkeiten beschränken sich auf Parameter, die allen CPUs derselben CPU-Familie gemeinsam sind.

Hinzufügen einer nicht spezifizierten CPU in der Portalsicht

Um eine nicht spezifizierte CPU in der Portalsicht hinzuzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf eine der folgenden Möglichkeiten:
   – "Geräte & Netze > Neues Gerät hinzufügen"
   – "PLC-Programmierung" > Schaltfläche "Gerät"
2. Wählen Sie für eine Gerätefamilie eine nicht spezifizierte CPU aus der Baumstruktur des Dialogs "Neues Gerät hinzufügen".
3. Klicken Sie auf "Hinzufügen".

Eine nicht spezifizierte CPU wird hinzugefügt und die Gerätesicht für diese CPU wird geöffnet.

Weitere Möglichkeiten für das Hinzufügen von nicht spezifizierten CPUs

In der Projektansicht können Sie nicht spezifizierte CPUs wie spezifizierte CPUs hinzufügen:

- In der Projektnavigation über die Schaltfläche "Neues Gerät hinzufügen"
- In der Task-Card "Hardware-Katalog"

Sie können auf diesen Wegen auch mehrere nicht spezifizierte CPUs hinzufügen.

Nicht spezifizierte CPUs spezifizieren

Es gibt zwei Möglichkeiten, wie Sie nicht spezifizierte CPUs spezifizieren können:

- Weisen Sie einer nicht spezifizierten CPU über den Baugruppentausch (Seite 60) eine vorhandene CPU per Drag & Drop aus dem Hardware-Katalog zu.
- Wählen Sie bei markierter nicht spezifizierter CPU den Menübefehl "Online > Hardware-Erkennung" und weisen Sie eine online ermittelte CPU zu. Vergeben Sie zu diesem Zweck eine IP-Adresse über die Schaltfläche "Adresse für PG/PC hinzufügen".

Hinweis

Wenn Sie nach der Durchführung der Hardware-Erkennung online gehen möchten, müssen Sie zuerst die erkannte Konfiguration in ihr Projekt laden, sonst kann es zu einem Fehler wegen inkonsistenter Konfigurationen kommen. Bei der CPU im Projekt und der tatsächlich vorhandenen CPU sind nach der Hardware-Erkennung zwar die Artikelnummern identisch, nicht aber deren Parameter. Die Parameter der CPU im Projekt haben die voreingestellten Werte, die Parameter der tatsächlich vorhandenen CPU die von Ihnen eingestellten Werte.

F-CPUs verhalten sich wie nach einem Baugruppentausch mit einer CPU ohne Failsafe-Funktionalität. Im Gegensatz zu F-CPUs aus dem Hardware-Katalog ist nach einem Baugruppentausch mit einer nicht spezifizierten CPU oder einer Hardware-Erkennung die F-Fähigkeit ausgeschaltet und die Schutzstufe ist auf "Vollzugriff (kein Schutz)" eingestellt.

Siehe auch

CPU auswählen (Seite 45)

Gerät zur Hardware-Konfiguration hinzufügen (Seite 42)

IO-Device konfigurieren über die Hardware-Erkennung (Seite 3011)

1.2.4 CPU auswählen

Einführung

In der Netzsicht wählen Sie aus dem Hardware-Katalog eine CPU aus und legen diese zusammen mit einem Baugruppenträger an. Auf dieses Gerät ziehen Sie aus dem Hardware-Katalog die gewünschten Baugruppen, die automatisch auf dem Baugruppenträger angeordnet werden.

Auswahl der Komponente im Hardware-Katalog

Im Hardware-Katalog wird jede Hardware-Komponente als Ordner angezeigt. Wenn Sie diesen Ordner öffnen, sehen Sie die unterschiedlichen Versionen der ausgewählten Hardware-Komponente mit ihren jeweiligen Artikelnummern.

Es wird Ihnen hier beispielhaft gezeigt, wie Sie in der Netzsicht eine CPU mit einem Baugruppenträger anlegen.

Voraussetzung

- Der Hardware-Katalog ist geöffnet.
- Sie befinden sich in der Netzsicht.

Vorgehen

Um eine CPU aus dem Hardware-Katalog auszuwählen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog auf den Ordner mit den gewünschten CPUs.
2. Öffnen Sie den Ordner mit dem gesuchten CPU-Typ. Sie sehen alle Artikelnummern des angewählten CPU-Typs.

3. Klicken Sie auf eine CPU-Artikelnummer, um sich in der Palette "Information" über die gewählte CPU zu informieren.

4. Legen Sie die CPU und einen Baugruppenträger an. Sie haben dazu folgende Möglichkeiten:
   - Ziehen Sie mit Drag & Drop die CPU aus dem Hardware-Katalog in die Netzsicht.
   - Kopieren Sie die CPU mit Copy & Paste in die Netzsicht.
   - Doppelklicken Sie auf den CPU-Eintrag im Hardware-Katalog.

Siehe auch

Hardware-Katalog durchsuchen (Seite 32)

Gerät zur Hardware-Konfiguration hinzufügen (Seite 42)

Baugruppenträger: Baugruppe stecken (Seite 51)

Baugruppenträger: Grundlagen (Seite 48)

Nicht spezifizierte CPU hinzufügen (Seite 43)

1.2.5 Fehlersichere Baugruppen auswählen

Einführung

Fehlersichere Automatisierungssysteme (F-Systeme) dienen der Steuerung von Prozessen mit unmittelbar durch Abschaltung erreichbarem sicherem Zustand. F-Systeme steuern Prozesse, bei denen eine unmittelbare Abschaltung keine Gefahr für Mensch oder Umwelt nach sich zieht.

Auswahl der F-Komponente im Hardware-Katalog

Im Hardware-Katalog wird jede Hardware-Komponente als Ordner angezeigt. Fehlersichere Hardware-Komponenten werden im Gegensatz zu den blauen nicht-fehlersicheren Hardware-Komponenten im Hardware-Katalog mit gelben Symbolen dargestellt.

Im folgenden Bild sehen Sie einen Auszug aus dem Hardware-Katalog mit einer F-CPU (gelb).

Über die farbliche Kennzeichnung können Sie einfach und schnell die F-Komponenten im Hardware-Katalog identifizieren.

Anzeige der F-Komponenten im Hardware- und Netzwerkeditor

Wenn Sie fehlersichere Geräte und "normale" Geräte in den grafischen Sichten des Hardware- und Netzwerkeditors verwenden, können Sie diese optisch einfach über die gelbe Kennzeichung des Geräts erkennen.

Im folgenden Bild sehen Sie zwei CPUs. Die F-CPU besitzt eine gelbe Kennzeichung:

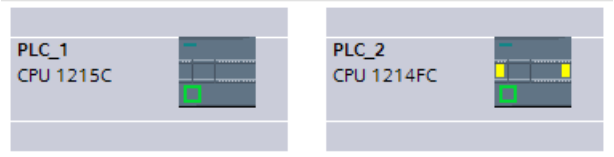

Fehlersichere Module auf Baugruppenträgern oder in Geräten können in der Gerätesicht auch an einer gelben Kennzeichnung erkannt werden. Da nicht-netzfähige Module in der Netz- oder Topologiesicht nicht angezeigt werden, werden Geräte mit nicht-netzfähigen F-Komponenten in der Netz- und Topologiesicht durch ein besonderes Symbol dargestellt.

Im folgenden Bild sehen Sie eine "normale" CPU und eine F-CPU in der Netzsicht, die aber beide F-Komponenten beinhalten:

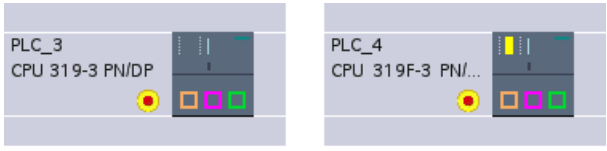

Über die farbliche Kennzeichnung der fehlersicheren Geräte und Module können Sie auch in den grafischen Sichten des Hardware- und Netzwerkeditors schnell die F-Komponenten identifizieren.

1.2.6 Baugruppenträger: Grundlagen

Einführung

Um Baugruppen einem Gerät zuzuweisen, benötigen Sie einen Baugruppenträger, z. B. eine Profilschiene. Auf dem Baugruppenträger befestigen Sie die Baugruppen und verbinden diese damit über den Rückwandbus mit der CPU, einer Stromversorgung oder anderen Baugruppen.

Anlegen eines Baugruppenträgers

Wenn Sie ein Gerät in die Netzsicht einfügen, wird automatisch eine Station und ein zum ausgewählten Gerät passender Baugruppenträger angelegt. In der Gerätesicht wird der Baugruppenträger mit den verfügbaren Steckplätzen angezeigt. Die Anzahl der verfügbaren Steckplätze richtet sich auch wieder nach dem Typ des verwendeten Geräts.

Aufbau eines Baugruppenträgers

Ein Baugruppenträger beinhaltet immer das Gerät, das in der Netzsicht eingefügt wurde. Das Gerät ist fest einem Steckplatz zugeordnet, wobei der Steckplatz von der Art des Geräts abhängt. Rechts und ggf. links von dem Gerät befinden sich weitere Steckplätze, oberhalb der Steckplätze befinden sich bei gesteckten Baugruppen die Steckplatznummern.

Über den gesteckten Geräten und Baugruppen wird eine entsprechende Kurzbezeichnung angezeigt. Diese Kurzbezeichnung können Sie über die Menüleiste unter "Ansicht" mit dem Befehl "Baugruppenbeschriftungen anzeigen" oder mit dem entsprechenden Symbol in der Funktionsleiste der Gerätesicht (Seite 14) ein- und ausschalten.

Symbol	Bedeutung
	Baugruppenbeschriftungen anzeigen

Wenn Baugruppen im Hardware-Katalog selektiert werden, werden alle für diese Baugruppe erlaubten Steckplätze markiert. Sie können dann sofort erkennen, an welchen Steckplatz die selektierte Baugruppe gesteckt werden kann.

In dem folgenden Bild wurde bei einem teilweise bestückten Baugruppenträger einer S7-1200 ein Signalmodul im Hardware-Katalog selektiert:

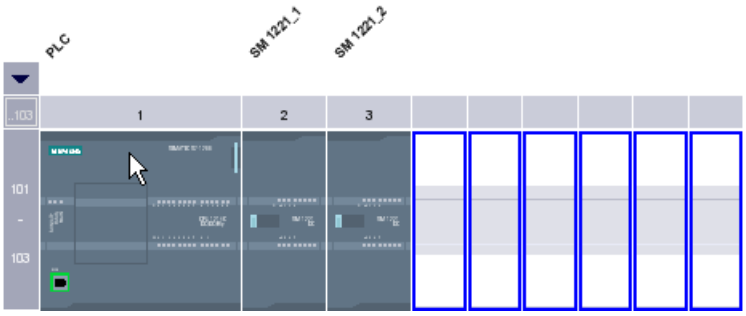

Da die Steckplätze 101-103 für Kommunikationsmodule reserviert sind, werden nur die anderen freien Steckplätze als mögliche Steckplätze angezeigt.

Die vordere Steckplatzgruppe können Sie über ein Pfeilsymbol oberhalb des erweiterbaren Steckplatzes auf- und zuklappen. Bei zugeklappter Steckplatzgruppe wird die erste und letzte Steckplatznummer der Gruppe angezeigt.

Das folgende Bild zeigt die aufgeklappte Steckplatzgruppe:

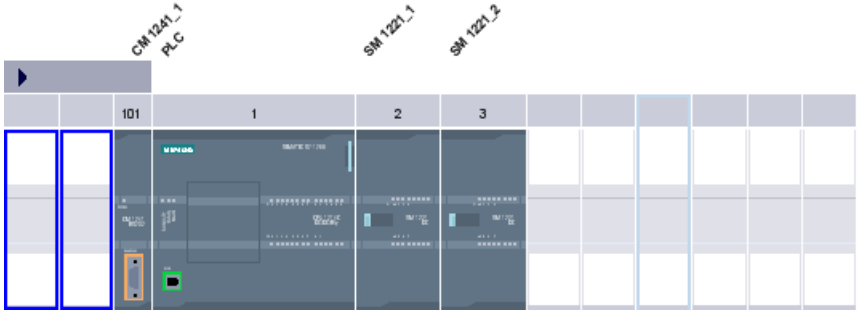

Steckplatzgruppen mit bereits gesteckten Baugruppen können nicht zugeklappt werden.

Mehrfachselektion von Baugruppen und Steckplätzen

Für die Selektion mehrerer Baugruppen oder Steckplätze stehen Ihnen mehrere Möglichkeiten zur Verfügung:

- Selektieren mehrerer Baugruppen oder Steckplätze bei gleichzeitigem Drücken von <Shift> oder <Strg>.
- Mausklick außerhalb des Baugruppenträgers und bei gedrückter Maustaste einen Rahmen um die zu selektierenden Baugruppen oder Steckplätze ziehen.

1.2.7 Baugruppenträger: Allgemeine Steckplatzregeln

Einführung

Für jedes Automatisierungssystem und jede Baugruppe gelten spezifische Steckplatzregeln.

Wenn Sie in der Gerätesicht eine Baugruppe im Hardware-Katalog selektieren, werden im Baugruppenträger alle möglichen Steckplätze für die selektierte Baugruppe markiert. Sie können Baugruppen nur auf markierte Steckplätze ziehen.

Wenn Sie eine Baugruppe einfügen, verschieben oder tauschen, werden ebenfalls die Steckplatzregeln angewendet.

Konsistenz

Einige Steckplatzregeln sind von der Konfiguration der Umgebung abhängig. Dies bedeutet, dass Sie manchmal Baugruppen in den Baugruppenträger stecken können, obwohl dies aktuell zu Inkonsistenzen führen würde. Wenn Sie die Konfiguration ändern, z. B. hinsichtlich der Auswahl anderer Baugruppen oder der Parametereinstellungen an den Baugruppen, können Sie die Konsistenz der Konfiguration wieder herstellen.

In Fällen, in denen das Stecken einer Baugruppe zu einer korrigierbaren Inkonsistenz führt, wird das Stecken zugelassen. Beim Übersetzen der Konfiguration wird eine Konsistenzprüfung durchgeführt. Inkonsistenzen werden als Meldung im Inspektorfenster unter "Info" angezeigt. Auf Grund des Ergebnisses der Konsistenzprüfung können Sie Ihre Konfiguration überarbeiten und und die Konsistenz wieder herstellen.

Regeln für die Anordnung von Baugruppen

Ganz allgemein gelten für Baugruppen in Baugruppenträgern folgende Regeln:

- In einen Baugruppenträger können Sie nur Baugruppen stecken.
- In eine Baugruppe können Sie nur Schnittstellenmodule stecken.
- Sie können nur Baugruppen der gleichen Produkt- oder Systemfamilie in einem Baugruppenträger verwenden.

Darüber hinaus gibt es für einige Baugruppen weitere spezielle Regeln:

- Steckbarkeit nur auf bestimmte Steckplätze
- Steckbarkeit in Abhängigkeit von anderen Baugruppen, CPUs oder Einstellungen
- Beschränkung der Anzahl der Verwendungen in einem Baugruppenträger

1.2.8 Baugruppenträger: Baugruppe stecken

Einführung

Nachdem Sie in der Netzsicht Geräte aus dem Hardware-Katalog Ihrer Konfiguration hinzugefügt haben, können Sie die Geräte mit Baugruppen bestücken. Um in der Gerätesicht eine Baugruppe in einen Baugruppenträger zu stecken, gibt es folgende Möglichkeiten:

- Bei freiem gültigen Steckplatz Doppelklick auf eine Baugruppe im Hardware-Katalog.
- Drag & Drop aus dem Hardware-Katalog auf einen freien gültigen Steckplatz in den grafischen oder tabellarischen Bereich:
  - Texteintrag aus der Palette "Katalog"
  - Vorschaugrafik aus der Palette "Information"
- "Kopieren" im Kontextmenü einer Baugruppe im Hardware-Katalog und Kontextmenü "Einfügen" auf einem freien gültigen Steckplatz im grafischen oder tabellarischen Bereich.

Um aus der Netzsicht in die Gerätesicht zu gelangen, doppelklicken Sie in der Netzsicht auf ein Gerät oder die Station oder verwenden Sie das Register Gerätesicht. In der Gerätesicht finden Sie eine Darstellung des ausgewählten Geräts innerhalb eines Baugruppenträgers. Die grafische Darstellung des Baugruppenträgers in der Software entspricht dabei dem realen Aufbau, d. h. es sind genauso viele Steckplätze sichtbar wie in dem realen Aufbau.

Hinweis

Sie können eine Baugruppe auch in der Netzsicht auf einen Baugruppenträger ziehen. Dazu muss die Filterfunktion des Hardware-Katalogs deaktiviert sein. Die Baugruppe wird automatisch auf einen freien und erlaubten Steckplatz gesteckt. Wenn kein Steckplatz vorhanden ist, wird die Baugruppe in die Ablage nicht gesteckter Baugruppen (Seite 20) geschoben.

Bestückung eines Baugruppenträgers

Ordnen Sie Baugruppen auf einem Baugruppenträger gemäß entsprechender Steckplatzregeln an.

Nach dem Stecken einer Baugruppe in einen Baugruppenträger mit einer bereits gesteckten CPU werden die Adressbereiche automatisch überprüft, damit es nicht zu einer Doppelbelegung von Adressen kommt. So besitzt jede Baugruppe nach dem Stecken einen gültigen Adressbereich. DP-Slaves und IO-Devices müssen hierfür mit einer CPU über das entsprechende DP-Master- oder IO-System vernetzt sein.

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Der Hardware-Katalog ist geöffnet.

Baugruppe aus dem Hardware-Katalog hinzufügen

Das Stecken einer Baugruppe aus dem Hardware-Katalog in einen Baugruppenträger wird Ihnen am Beispiel eines digitalen Signalmoduls gezeigt. Gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu der gewünschten Baugruppe.

   Hinweis

   Wenn Sie die Filterfunktion des Hardware-Katalogs aktivieren, werden Ihnen nur Baugruppen angezeigt, die zu dem ausgewählten Gerätetyp kompatibel sind.

2. Selektieren Sie die ausgewählte Baugruppe.
3. Stellen Sie ggf. den FW-Stand über die Klappliste des Hardware-Katalogs ein.
4. Ziehen Sie das Signalmodul mit Drag & Drop auf einen freien Steckplatz des Baugruppenträgers.

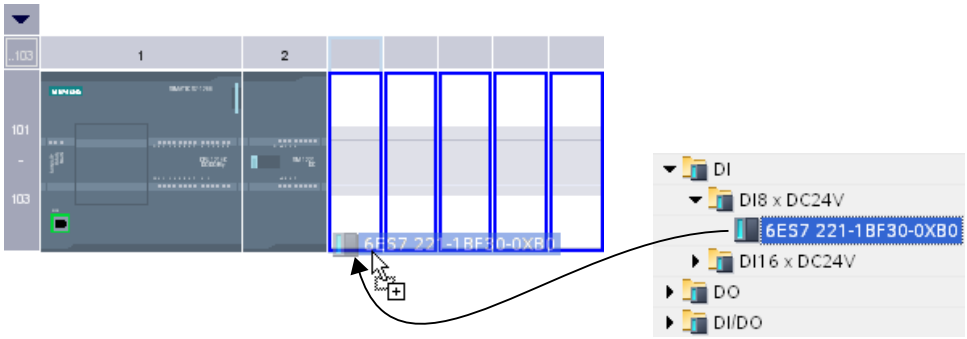

Sie haben nun das digitale Signalmodul in einen Steckplatz im Baugruppenträger gesteckt. Verfahren Sie mit weiteren Baugruppen genauso.

Über den eingefügten Baugruppen wird der Name der Baugruppe angezeigt. In der Menüleiste können Sie über "Ansicht > Baugruppenbeschriftungen anzeigen" die Beschriftung der Baugruppe ein- und ausschalten.

Baugruppe einfügen

Sie können Baugruppen mit Drag & Drop auch zwischen bereits gesteckte Baugruppen einfügen. Ziehen Sie dazu eine Baugruppe mit gedrückter Maustaste oberhalb zwischen zwei bereits gesteckte Baugruppen.

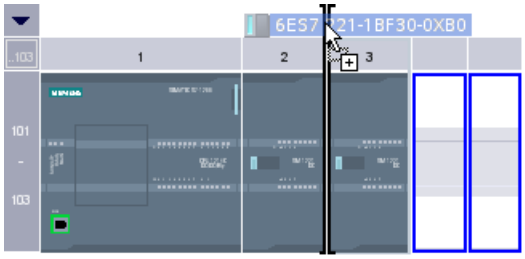

Es erscheint eine Einfügemarke. Wenn Sie den Mauszeiger loslassen, werden alle rechts der Einfügemarke gesteckten Baugruppen um einen Steckplatz nach rechts verschoben. Eine eventuell überzählige Baugruppe wird in die Ablage nicht gesteckter Baugruppen verschoben. Auf den entstehenden freien Steckplatz wird die neue Baugruppe gesteckt.

Siehe auch

Gerätesicht (Seite 14)

Gerätesicht: Ablage nicht gesteckter Baugruppen (Seite 20)

Baugruppenträger: Allgemeine Steckplatzregeln (Seite 49)

1.2.9 Hardware-Komponente selektieren

Sie können Hardware-Komponenten und weitere Objekte der grafischen Sichten über die Maus mit einem Linksklick selektieren. Über das Ziehen der Maus mit gedrückter linker Maustaste oder über die Kombination von Linksklick mit den Tasten <Shift> oder <Strg> ist außerdem eine Mehrfachauswahl der Objekte möglich.

Die folgenden Objekte können in den grafischen Sichten selektiert werden: Geräte (mit CPU, Baugruppenträger, Modulen), CPUs, CPs, Module, Subnetze, Verbindungen und Knoten.

Regeln

Sie können Objekte in den grafischen Sichten auf die folgenden Arten selektieren:

- Linksklick auf ein Objekt: Das Objekt wird selektiert.
- Linksklick und <Strg> auf ein Objekt: Das Objekt wird der aktuellen Auswahl hinzugefügt.
- Linksklick und <Shift> auf ein Objekt: Alle Objekte zwischen dem aktuell selektierten Objekt und den vorher selektierten Objekten werden der Auswahl hinzugefügt.
- Auswahlrahmen mit gedrückter linker Maustaste um mehrere Objekte ziehen: Alle vom Rahmen komplett umschlossenen Objekte werden der Auswahl hinzugefügt.

Eine Mehrfachauswahl ist nur bei Objekten der selben Kategorie möglich:

- Kategorie 1: Geräte, Subnetze, Verbindungen, Port-Verschaltungen
- Kategorie 2: CPUs, CPs, Schnittstellenmodule / Module
- Kategorie 3: Schnittstellen, Ports, Knoten

Verhalten bei Mehrfachauswahl mit Objekten unterschiedlicher Kategorie:

- Auswahlrahmen: Nur Objekte der jeweils höchsten Kategorie werden Bestandteil der Mehrfachauswahl
- Linksklick mit <Shift> oder <Strg>: Das zuletzt selektierte Objekt einer neuen Kategorie bleibt selektiert, die vorher selektierten Objekte fallen aus der Auswahl heraus.

Hinweis

Mehrfachauswahl unterschiedlicher Arten von Objekten

Bei einer Mehrfachauswahl können nur gleiche Kategorien von Objekten ausgewählt sein. Beispielsweise können mehrere CPUs ausgewählt sein oder auch mehrere Netzwerke. Es ist aber nicht möglich, in einer Mehrfachauswahl zuerst ein Gerät, von einem zweiten Gerät nur die CPU und von einem dritten Gerät nur die Schnittstelle der Mehrfachauswahl hinzuzufügen.

Mehrfachauswahl von Geräten mithilfe des Auswahlrahmens

Um mehrere Geräte in der Netzsicht durch das Ziehen eines Auswahlrahmens zu selektieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen um alle Geräte, die Sie in Ihrer Auswahl haben möchten:

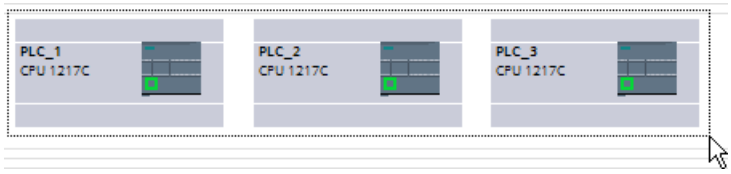

2. Lassen Sie die Maustaste los. Alle Geräte innerhalb des gezogenen Rahmes sind nun Teil der Mehrfachauswahl.

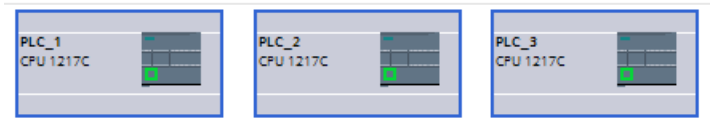

In dem obigen Beispiel wurde der Auswahlrahmen um ganze Geräte gezogen. Wenn Sie den Rahmen so ziehen, dass keine ganzen Geräte umschlossen werden, sondern nur die darin befindlichen CPUs, wird nicht das ganze Gerät in die Mehrfachauswahl aufgenommen, sondern lediglich die CPUs. Wenn Sie den Auswahlrahmen so ziehen, dass teilweise ganze Geräte und teilweise nur CPUs umschlossen sind, ist die höchste Kategorie der selektierten Objekte entscheidend und nur die ganzen Geräte werden der Mehrfachauswahl hinzugefügt.

Mehrfachauswahl von CPUs mithilfe der <Shift>-Taste

Statt der Mehrfachauswahl mit einem Auswahlrahmen können Sie die Mehrfachauswahl auch mit der <Shift>-Taste vornehmen. Sie selektieren dabei zwei Objekte, damit alle dazwischen liegenden Objekte ebenfalls in die Mehrfachauswahl aufgenommen werden.

Um mehrere CPUs in der Topologiesicht mithilfe der <Shift>-Taste zu selektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die erste CPU Ihrer gewünschten Mehrfachauswahl mit der linken Maustaste:

2. Selektieren Sie die letzte CPU Ihrer gewünschten Mehrfachauswahl mit der linken Maustaste und gedrückter <Shift>-Taste. Die beiden selektierten CPUs verhalten sich wie die Begrenzungen eines Auswahlrahmens, hier dargestellt durch die orangefarbene Linie:

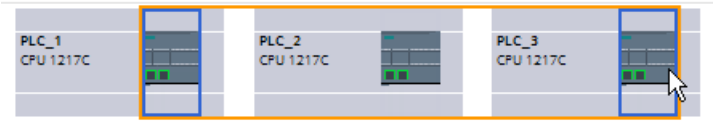

3. Alle CPUs zwischen der ersten und letzten selektierten CPU werden in die Mehrfachauswahl aufgenommen:

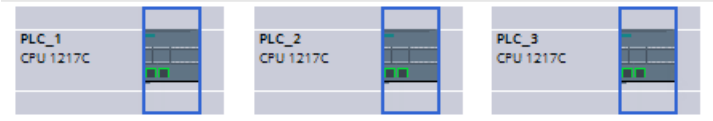

In dem obigen Beispiel wurden nur CPUs selektiert. Wenn Sie stattdessen ganze Geräte selektieren, werden die zwischen dem ersten und zweiten selektierten Gerät liegenden Geräte in die Mehrfachauswahl aufgenommen. Wenn Sie sowohl ein ganzes Gerät als auch eine CPU selektieren, ist die Kategorie des zuletzt selektierten Objektes entscheidend für die Kategorie der insgesamt selektierten Objekte.

Mehrfachauswahl von Geräten mithilfe der Tasten <Strg> und <Shift>

Sie können eine Mehrfachauswahl auch mit den beiden Tasten <Shift> und <Strg> vornehmen. Sie selektieren dabei mehrere Objekte mit linkem Mausklick und der <Strg>-Taste. Wenn Sie danach ein Objekt mit linkem Mausklick und der <Shift>-Taste selektieren, werden alle Objekte in die Mehrfachauswahl aufgenommen, die zwischen den vorher selektierten Objekten und dem zuletzt selektierten Objekt liegen.

Um mehrere Geräte in der Gerätesicht mithilfe der Tasten <Shift> und <Strg> zu selektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie mit der linken Maustaste und gedrückter <Strg>-Taste mehrere einzelne Geräte:

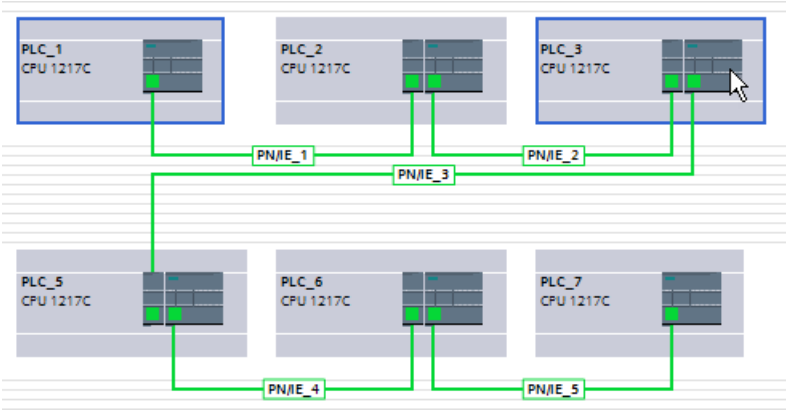

2. Selektieren Sie das letzte Gerät Ihrer gewünschten Mehrfachauswahl mit der linken Maustaste und gedrückter <Shift>-Taste. Die bereits selektierten Geräte verhalten sich wie die Begrenzungen eines Auswahlrahmens, hier dargestellt durch die orangefarbene Linie:

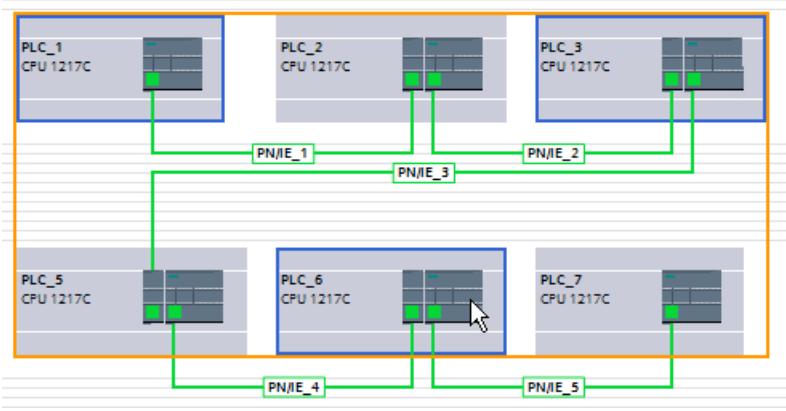

3. Nun werden alle Geräte des virtuellen Auswahlrahmens und die im Auswahlrahmen liegenden Subnetze in die Mehrfachauswahl aufgenommen.

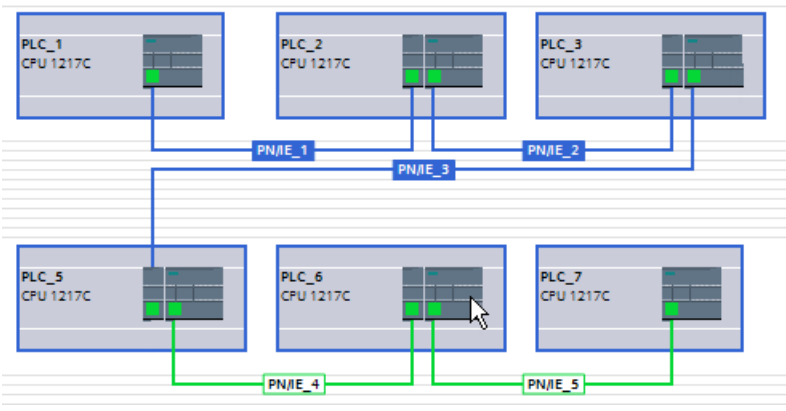

Es werden nur gleiche Kategorien von Objekten selektiert. In obigem Beispiel also neben den entsprechenden Geräten auch die Subnetze, da sie innerhalb des virtuellen Auswahlrahmens lagen und derselben Kategorie wie die Geräte angehören. Die Subnetze PN/IE_4 und PN/IE_5 lagen nicht innerhalb des virtuellen Auswahlrahmens und sind daher nicht Teil der Mehrfachauswahl.

1.2.10 Hardware-Komponente kopieren

Sie können Hardware-Komponenten in der Geräte- oder Netzsicht kopieren. Kopierte Hardware-Komponenten werden in einer Zwischenablage abgelegt und können an einer anderen Stelle aus dieser heraus eingefügt werden. Kopierte Stationen werden als neue Stationen in der Netzsicht eingefügt, kopierte Geräte und Baugruppen können in der Netz- und Gerätesicht in bestehende Baugruppenträger eingefügt werden.

Regeln

- Es können sowohl einzelne Objekte als auch mehrere Objekte gleichzeitig kopiert werden.
- Gesteckte Baugruppen aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können kopiert werden.
- Geräte und Baugruppen können unter Berücksichtigung der Steckplatzregeln nur auf freie und gültige Steckplätze kopiert werden.
- Baugruppenträger mit gesteckter CPU können nicht einzeln kopiert werden, sondern nur komplett mit allen gesteckten Hardware-Komponenten.

Vorgehen

Um eine Hardware-Komponente zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponente, die Sie kopieren möchten.
   - Gerätesicht: Selektieren Sie die Baugruppe in einem Baugruppenträger oder in der Ablage nicht gesteckter Baugruppen.
   - Netz- oder Topologiesicht: Selektieren Sie die Station oder die in der Netzsicht relevante Hardware-Komponente.
   - Projektnavigation: Selektieren Sie die Station oder Baugruppe.
2. Wählen Sie im Kontextmenü "Kopieren" oder drücken Sie <Strg+C>.
   Wenn der Menüpunkt "Kopieren" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht kopiert werden kann.

3. Selektieren Sie den Ort, in den der Inhalt aus der Zwischenablage eingefügt werden soll.
   - Selektieren Sie in der Gerätesicht einen freien Steckplatz des Baugruppenträgers oder die Ablage nicht gesteckter Baugruppen.
   - Selektieren Sie in der Netz- oder Topologiesicht eine Station, um Geräte oder Baugruppen einzufügen.
4. Wählen Sie im Kontextmenü "Einfügen" oder drücken Sie <Strg+V>.
   - Wenn Sie das kopierte Objekt ohne Selektion in der Netz- oder Topologiesicht mit <Strg +V> und einfügen, wird das kopierte Objekt rechts neben das Original-Objekt eingefügt.
   - Wenn Sie das kopierte Objekt ohne Selektion in der Netz- oder Topologiesicht über das Kontextmenü einfügen, wird es an der Stelle des Mauszeigers platziert.

   Wenn der Menüpunkt "Einfügen" gegraut dargestellt wird, ist die Zwischenablage leer oder enthält mindestens eine Komponente, die an dieser Stelle nicht eingefügt werden kann.

Das kopierte Objekt wird am gewählten Platz eingefügt.

Wenn Sie zum Einfügen einer Baugruppe eine Station in der Netz- oder Topologiesicht selektiert haben, wird die Baugruppe auf den ersten freien und gültigen Steckplatz gesteckt. Steht kein freier und gültiger Steckplatz zur Verfügung, wird das Objekt in die Ablage nicht gesteckter Baugruppen eingefügt.

Hinweis

Sie können eine Baugruppe auch von einem Gerät in ein anderes kopieren:

Kopieren Sie dazu eine Baugruppe im Hardware- und Netzwerkeditor, selektieren Sie dann in der Netz- oder Topologiesicht oder der Klappliste der Gerätesicht ein anderes Gerät und fügen Sie die Baugruppe ein.

In der Gerätesicht können Sie das kopierte Objekt direkt auf einen Steckplatz bzw. in die Ablage nicht gesteckter Baugruppen einfügen. Wenn Sie das kopierte Objekt in der Netzsicht einem Gerät oder einer Station hinzufügen, wird das kopierte Objekt in den ersten freien Steckplatz gesteckt.

Ist kein Steckplatz für das Objekt verfügbar, wird das Objekt automatisch in die Ablage nicht gesteckter Baugruppen (Seite 20) eingefügt.

Hinweis

Sie können eine selektierte Hardware-Komponente auch direkt mit <Strg> und Drag & Drop kopieren.

Siehe auch

Tastaturbedienung: Bearbeiten von Objekten (Seite 39)

1.2.11 Hardware-Komponente verschieben

Sie können Hardware-Komponenten in der Geräte- oder Netzsicht verschieben.

Regeln

- Geräte, Baugruppen und Module aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können unter Berücksichtigung der Steckplatzregeln verschoben werden.
- CPs können in der Netzsicht verschoben werden. Der CP wird im Zielgerät in einen freien und gültigen Steckplatz gesteckt. Ist kein freier Steckplatz vorhanden, wird der einzufügende CP in die Ablage nicht gesteckter Baugruppen verschoben.
- CPUs und Slave-Kopfbaugruppen können in der Netzsicht zwischen den Geräten verschoben werden, je nach CPU-Typ auch innerhalb des Baugruppenträgers.

Hinweis

Verschobene CPs werden von ihrem Netz getrennt, behalten aber die Netzparameter und Adresse. Wenn der CP wieder an das Netz angeschlossen wird, aber dessen Adresse bereits anderweitig vergeben wurde, können Sie dem CP über einen Dialog eine neue eindeutige Adresse zuweisen.

Vorgehen

Um eine Hardware-Komponente zu verschieben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponente, die Sie verschieben möchten.
   - Gerätesicht: Selektieren Sie die Baugruppe in einem Baugruppenträger oder in der Ablage nicht gesteckter Baugruppen.
   - Netzsicht: Selektieren Sie die netzsichtrelevante Hardware-Komponente.
2. Wählen Sie im Kontextmenü "Ausschneiden" oder drücken Sie <Strg+X>.
   Wenn der Menüpunkt "Ausschneiden" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht ausgeschnitten werden kann.
3. Selektieren Sie den Ort, an den das ausgeschnittene Objekt verschoben werden soll.
   - Gerätesicht: Selektieren Sie einen freien Steckplatz des Baugruppenträgers oder die Ablage nicht gesteckter Baugruppen.
   - Netzsicht: Selektieren Sie eine Station, um Geräte oder Baugruppen einzufügen.
4. Wählen Sie im Kontextmenü "Einfügen" oder drücken Sie <Strg+V>.
   Wenn der Menüpunkt "Einfügen" gegraut dargestellt wird, ist die Zwischenablage leer oder enthält mindestens eine Komponente, die an dieser Stelle nicht eingefügt werden kann.

Die selektierte Hardware-Komponente wird an den Zielort verschoben. Falls es sich bei der verschobenen Hardware-Komponente um ein vernetztes Objekt handelt, wird es vom Netz entkoppelt.

Hinweis

Sie können eine selektierte Hardware-Komponente auch direkt mit Drag & Drop verschieben.

Siehe auch

Tastaturbedienung: Bearbeiten von Objekten (Seite 39)

1.2.12 Hardware-Komponente tauschen

Sie können Hardware-Komponenten gegeneinander tauschen. Dadurch können Sie beispielsweise nicht spezifizierte CPUs (Seite 43) durch verfügbare CPUs aus dem Hardware-Katalog ersetzen.

Regeln

Es können nur Hardware-Komponenten gegeneinander getauscht werden, die den Baugruppentausch unterstützen und die miteinander kompatibel sind.

Vorgehen

Um Baugruppen gegeneinander zu tauschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Baugruppe, die Sie austauschen möchten.
2. Öffnen Sie das Kontextmenü:
   – Wenn der Eintrag "Gerät tauschen" aktiv ist, ist ein Baugruppentausch möglich.
   – Wenn der Eintrag "Gerät tauschen" deakiviert ist, ist ein Baugruppentausch nicht möglich.
3. Klicken Sie im Kontextmenü auf den Eintrag "Gerät tauschen". Das Dialogfeld "Gerät tauschen" wird geöffnet.
4. Selektieren Sie in der Baumstruktur unter "Neues Gerät" die Baugruppe, durch die Sie Ihre aktuelle Baugruppe ersetzen möchten.
5. Klicken Sie auf "OK".
6. Übersetzen Sie die Hardware und Software.

Die vorhandene Baugruppe wird durch die neue Baugruppe ersetzt.

Alternativ können Sie auch direkt eine Baugruppe aus dem Hardware-Katalog mit Drag & Drop auf die zu tauschende Baugruppe ziehen. Wenn die beiden Baugruppen gegeneinander ausgetauscht werden können, wird dies durch ein Symbol am Mauszeiger dargestellt.

1.2.13 Hardware-Komponente umbenennen

Sie können die Namen der Hardware-Komponenten in allen drei Sichten (Netzsicht, Gerätesicht, Topologiesicht) komfortabel direkt im grafischen Bereich, aber auch in den entsprechenden tabellarischen Bereichen oder im Inspektorfenster bearbeiten.

Regeln

Sie können Hardware-Komponenten auf die folgenden Arten umbennen:

- In der grafischen Sicht nach Selektion des angezeigten Namens.
- In der tabellarischen Sicht in der Tabellenzelle mit dem zu ändernden Namen.
- Im Inspektorfenster unter "Eigenschaften > Allgemein" für die selektierte Hardware-Komponente.

Vorgehen

Um eine Hardware-Komponente im grafischen Bereich einer Sicht umzubenennen, gehen Sie folgendermaßen vor:

Für die Gerätesicht gilt:

1. Aktivieren Sie die Anzeige der Baugruppennamen über die entsprechende Schaltfläche in der Funktionsleiste.
2. Selektieren Sie die Hardware-Komponente oder dessen Namen.
3. Sie haben jetzt zwei Möglichkeiten:
   - Drücken Sie <F2>.
   - Klicken Sie auf den vorher selektierten Namen.
4. Bearbeiten Sie den Namen

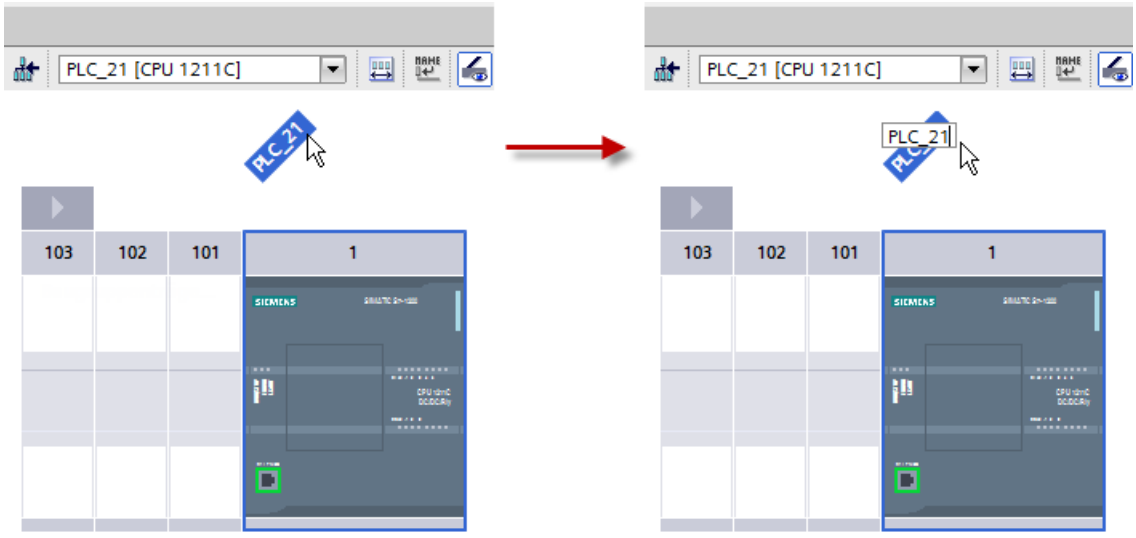

Für die Netz- und Topologiesicht gilt:

1. Selektieren Sie die Hardware-Komponente oder dessen Namen.
2. Sie haben jetzt drei Möglichkeiten:
   – Drücken Sie <F2>.
   – Klicken Sie auf den vorher selektierten Namen.
   – Wählen Sie im Kontextmenü den Befehl "Umbennen".

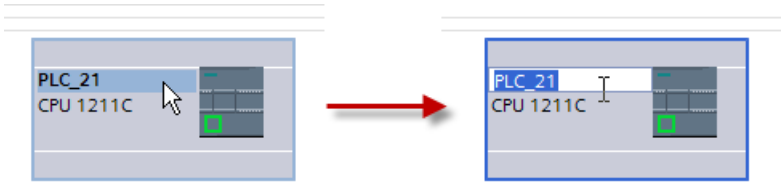

Für die Bearbeitung von Namen im tabellarischen Bereich selektieren Sie in der Tabelle die Zelle mit dem Namen, den Sie bearbeiten möchten. Klicken Sie danach erneut auf den Namen, damit Sie den Namen direkt in der Zelle bearbeiten können.

Für die Bearbeitung von Namen im Inspektorfenster selektieren Sie im grafischen Bereich die Hardware-Komponente, deren Namen sie bearbeiten möchten. Im Inspektorfenster werden Ihnen dann die relevanten Parameter angezeigt. Unter "Eigenschaften > Allgemein" können Sie nun die Namen bearbeiten.

Siehe auch

Netzsicht (Seite 11)

Gerätesicht (Seite 14)

Topologiesicht (Seite 21)

Inspektorfenster (Seite 24)

1.2.14 Hardware-Komponente löschen

Sie können Hardware-Komponenten über unterschiedliche Wege löschen. Gelöschte Hardware-Komponenten werden aus dem System entfernt und belegte Adressen stehen wieder zur Verfügung.

Regeln

- CPUs oder Baugruppen aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können gelöscht werden.
- Wird ein Baugruppenträger in der Gerätesicht gelöscht, werden die gesteckten Hardware-Komponenten in die Ablage nicht gesteckter Baugruppen verschoben.

Vorgehen

Um eine Hardware-Komponente zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponenten, die Sie löschen möchten.
   - Netzsicht: Selektieren Sie in der grafischen Sicht oder in der Netzübersicht Geräte oder netzrelevante Hardware-Komponenten.
   - Gerätesicht: Selektieren Sie in der grafischen Sicht oder in der Geräteübersicht Baugruppenträger oder Baugruppen in Baugruppenträgern oder in der Ablage nicht gesteckter Baugruppen.
   - Topologiesicht: Selektieren Sie in der grafischen Sicht oder in der Topologieübersicht Geräte oder Hardware-Komponenten mit Ethernet-Schnittstellen..
   - Projektnavigation: Selektieren Sie Geräte oder einzelne Hardware-Komponenten aus der Baumstruktur.
2. Wählen Sie im Kontextmenü "Löschen" oder drücken Sie <Entf>.
   Wenn der Menüpunkt "Löschen" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht gelöscht werden kann.

Die selektierten Hardware-Komponenten werden gelöscht.

Hinweis

Durch das Löschen von Hardware-Komponenten kann das Projekt Inkonsistenzen aufweisen, z. B. Verletzung von Steckplatzregeln. Auf die Inkonsistenzen wird bei der Konsistenzprüfung hingewiesen. Beseitigen Sie die Inkonsistenzen durch entsprechende Handlungen, z. B. stellen Sie die Einhaltung der Steckplatzregeln wieder her.

Siehe auch

Tastaturbedienung: Bearbeiten von Objekten (Seite 39)

1.2.15 Hardware-Komponente ändern oder aktualisieren

Begriffsklärung

Sie können die Firmware-Version einer Hardware-Komponente auf eine neue Version ändern (sofern verfügbar) oder mit einer Aktualisierung der Modulbeschreibung unter Umständen neue Funktionen der Hardware-Komponente nutzen.

Im Folgenden werden die Begriffe "Modulbeschreibung" und "Firmware-Version" genauer erklärt.

- Modulbeschreibung: diejenige Version der Projektierungs-Software, aus der die technische Beschreibung des Moduls stammt.
  Bsp.: V11.0.0.0
- Firmware-Version: der Ausgabestand der Firmware des offline parametrierten Moduls
  Bsp.: V2.0

Voraussetzung

- Sie haben eine Gerätekonfiguration erstellt.
- Sie haben zu einem späteren Zeitpunkt beispielsweise ein Update oder ein Optionspaket installiert. Durch diese Installation wurde im Hardware-Katalog bei mindestens einem Modultyp die Modulbeschreibung aktualisiert, wobei die neue Version der Modulbeschreibung inkompatibel zur bisherigen ist.
- Sie haben in Ihrer Gerätekonfiguration solche Module verwendet und wollen die geänderten bzw. hinzugekommenen Eigenschaften nutzen.

Vorgehen

Führen Sie für jeden betroffenen Modultyp die folgenden Schritte durch.

1. Selektieren Sie in der Gerätesicht ein betroffenes Modul.
2. Gehen Sie im Inspektorfenster zu Eigenschaften > Allgemein > Kataloginformation. Klicken Sie dort auf die Schaltfläche "Modulbeschreibung aktualisieren".
3. Legen Sie in der dann erscheinenden Abfrage fest, ob Sie im aktuellen Projekt nur für das selektierte Modul oder für alle Module dieses Typs die Modulbeschreibung aktualisieren wollen.

Ergebnis

Im aktuellen Projekt werden die ausgewählten Module durch die gleichen Module mit aktualisierter Modulbeschreibung ersetzt.

In welchen Fällen brauchen Sie die Aktualisierung der Modulbeschreibung nicht durchzuführen?

Die Aktualisierung der Modulbeschreibung ist in den folgenden Fällen überflüssig:

- Sie wollen die geänderten bzw. hinzugekommenen Eigenschaften der Module nicht nutzen.
- Sie öffnen ein bestehendes Projekt mit einem Stand der Projektierungs-Software, der aktueller ist als der Stand, mit dem Sie das Projekt erstellt haben, und das System führt eine automatische Projektkonvertierung z. B. von TIA Portal V12 nach V13 durch. In diesem Fall werden alle veralteten Modulbeschreibungen automatisch angepasst.

1.2.16 Eigenschaften und Parameter bearbeiten

Nachdem Sie Hardware-Komponenten auf Ihren Baugruppenträger gesteckt haben, können Sie in der Netz-, Geräte- oder Topologiesicht deren voreingestellte Eigenschaften bearbeiten, z. B. die Namen, Parameter oder Adressen.

Voraussetzung

Sie befinden sich in der Netz-, Geräte- oder Topologiesicht.

Hinweis

Sie können in den verschiedenen Sichten die relevanten Eigenschaften und Parameter sowohl in der grafischen Sicht, als auch in der tabellarischen Sicht bearbeiten. Welche Parameter Sie bearbeiten können, hängt davon ab, in welcher Sicht Sie sich befinden. In der grafischen Netzsicht haben Sie Zugriff auf die netzrelevanten Hardware-Komponenten und die Station. Auf Baugruppen und Hardware-Komponenten, die nicht in der grafischen Netzsicht angezeigt werden, können Sie über die tabellarische Netzsicht zugreifen. In der Topologiesicht und der Gerätesicht haben Sie wiederum auf weitere Parameter bezüglich der Topologie oder der Geräte Zugriff.

Vorgehen

Um die Eigenschaften und Parameter der Hardware-Komponenten in der Gerätesicht zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der grafischen Darstellung die CPU, die Baugruppe, den Baugruppenträger oder die Schnittstelle, die Sie bearbeiten möchten.
2. Bearbeiten Sie die Einstellungen für das gewählte Objekt:
   - Im grafischen Bereich können Sie z. B. bei aktivierter Baugruppenbeschriftung die Namen der Objekte ändern.
   - Im tabellarischen Bereich können Sie z. B. Adressen und Namen ändern.
   - Im Inspektorfenster stehen Ihnen unter "Eigenschaften" verschiedenste Einstellmöglichkeiten zur Verfügung.

Beachten Sie, dass Baugruppen nur dann vollständig parametriert werden können, wenn sie einer CPU zugeordnet sind. Daher müssen PROFIBUS- bzw. PROFINET-Interfacemodule zunächst mit der CPU bzw. einem zentral steckenden Kommunikationsmodul vernetzt werden, so dass sie ein Mastersystem bzw. IO-System bilden. Erst dann sind z. B. die Adressen der dezentral gesteckten Komponenten editierbar.

Beispiel für das Ändern von Einstellungen

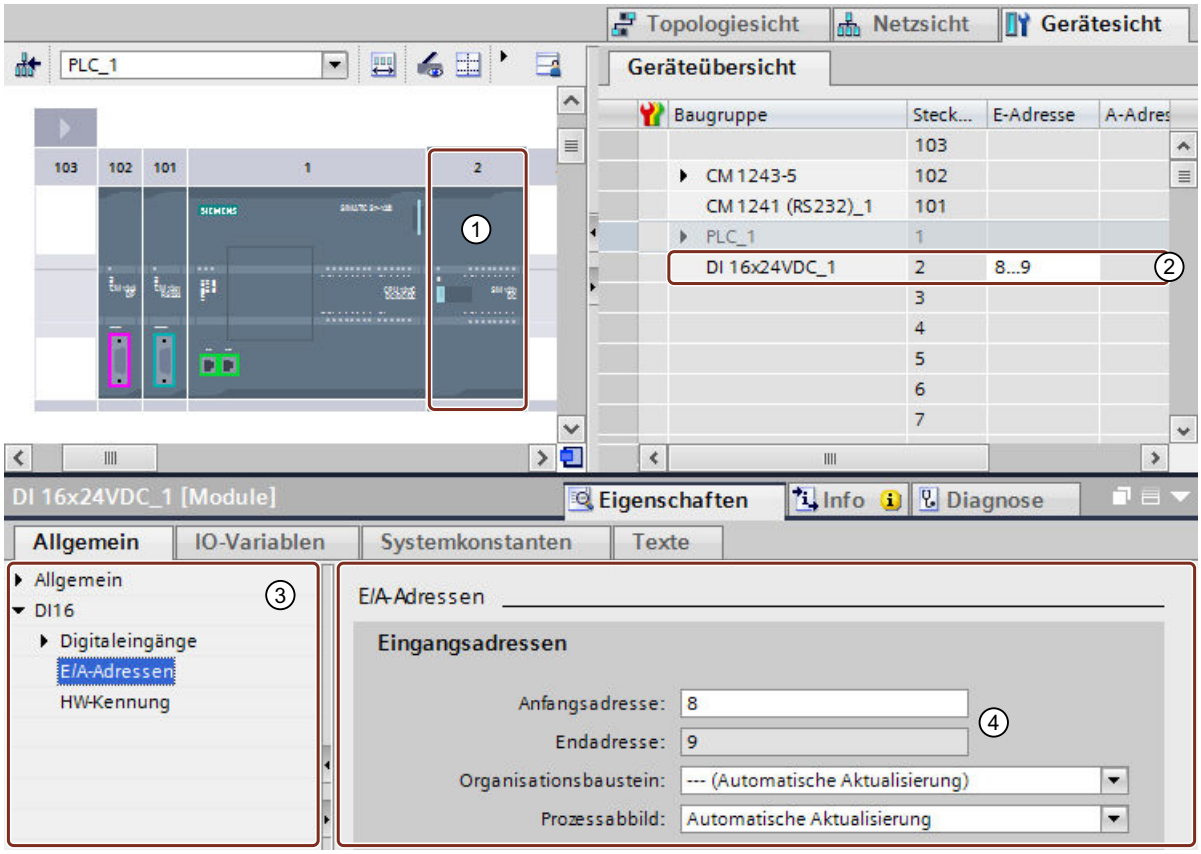

① Selektion einer Baugruppe
② Bearbeitungsmöglichkeit für Adressen in der Geräteübersicht
③ Auswahlmöglichkeiten im Inspektorfenster
④ Bearbeitungsmöglichkeit für Adressen im Inspektorfenster

Siehe auch

Inspektorfenster (Seite 24)

1.2.17 Ein- und Ausgangsadressen in der Adressübersicht

Einführung

Die aktuell verwendeten Ein- und Ausgangsadressen können in der Adressübersicht tabellarisch angezeigt werden. Die Adressübersicht finden Sie im Inspektorfenster unter "Eigenschaften" der CPU.

Aufbau der Adressübersicht

Über verschiedene Optionskästchen können Sie einstellen, welche Objekte in der Adressübersicht angezeigt werden sollen:

- Eingänge: Anzeige der Eingangsadressen
- Ausgänge: Anzeige der Ausgangsadressen
- Adresslücken: Anzeige freier Adressräume
- Steckplatz: Anzeige der Steckplatznummer

Typischerweise werden in der Adressübersicht die folgenden Informationen angezeigt:

Tabellenkopf	Bedeutung
Typ	Typ des Adressbereiches. Zeigt an, ob es sich um einen Eingangs- oder einen Ausgangsadressbereich handelt. • E = Eingangsadresse von Baugruppe mit Eingangs-Peripherieadressraum (mit Länge > 0) • E* = Eingangs- bzw. Diagnoseadresse von Baugruppen ohne Eingangs-Peripherieadressraum (mit Länge = 0). Über diese Adressen werden keine Daten (Signalzustände oder Messwerte) im Anwenderprogramm ausgetauscht. Verwendung zur Meldung von Diagnoseereignissen vom Betriebssystem (z. B. in der Startinformation von Fehler-OBs). • A = Ausgangsadresse
Adr. von	Anfangsadresse des Adressbereiches. Zeigt die erste belegte Adresse des Ein- oder Ausgangs an.
Adr. bis	Endadresse des Adressbereiches. Zeigt die letzte belegte Adresse des Ein- oder Ausgangs an.
Größe	Adresslänge des Adressbereiches, der von der Baugruppe für den Eingang oder Ausgang belegt wird. Beispiele: • Bei einer Anfangsadresse von 0 und einer Endadresse von 1 beträgt die Adresslänge 2 Bytes. • Bei einer Anfangsadresse von 0 und einer Endadresse von 3 beträgt die Adresslänge 4 Bytes. • Bei einer Anfangsadresse von 0.1 und einer Endadresse von 0.2 beträgt die Adresslänge 2 Bits. • Sind die Anfangs- und Endadresse identisch, wird 0 Bytes angezeigt.
Baugruppe	Baugruppe, von der der Adressbereich genutzt wird.
Baugruppenträger	Nummer des Baugruppenträgers, auf dem die Hardwarekomponente gesteckt ist.
Steckplatz	Nummer des Steckplatzes auf dem Baugruppenträger, in dem die Hardware-Komponente gesteckt ist.

Tabellenkopf	Bedeutung
Gerätename	Name und Typ des Gerätes, in dem die Baugruppe steckt. Beispiele: • Die Baugruppe steckt in einer PLC mit dem Namen "PLC_1" vom Typ CPU 1516-3 PN/DP, angezeigt wird "PLC_1 [CPU 1516-3]". • Die Baugruppe steckt in einem Kopfmodul mit dem Namen "IO Device_1" vom Typ IM 151-3 PN, angezeigt wird "IO Device_1 [IM 151-3 PN]"
Gerätenummer	PROFINET-Gerätenummer oder PROFIBUS-Adresse, wenn die Baugruppe über PROFINET oder PROFIBUS mit einer PLC verbunden ist. Steckt die Baugruppe direkt in einer PLC, wird "-" angezeigt. Beispiele: • Die Baugruppe steckt in einem Kopfmodul, das über eine PROFINET-Schnittstelle mit einer PLC verbunden ist, angezeigt wird die Gerätenummer der Ethernet-Adresse. • Die Baugruppe steckt in einem DP-Slave, der über eine PROFIBUS-Schnittstelle mit einer PLC verbunden ist, angezeigt wird der Adressparameter der PROFIBUS-Adresse.
Master/IO-System	Name des DP-Mastersystems oder IO-Systems, über das der DP-Slave oder das IO-Device mit der PLC verbunden ist. In eckigen Klammern wird die entsprechende Nummer des DP-Mastersystems oder IO-Systems angezeigt. Steckt die Baugruppe direkt in einer PLC, wird "-" angezeigt. Beispiele: • Die Baugruppe steckt in einem IO-Device und ist über ein PROFINET IO-System mit dem Namen "PROFINET IO-System_1" und der Nummer 100 mit einer PLC verbunden. Angezeigt wird "PROFINET IO-System_1 [100]". • Die Baugruppe steckt in einem DP-Slave und ist über ein PROFIBUS DP-Mastersystem mit dem Namen "DP-Mastersystem" und der Nummer 1 mit einer PLC verbunden. Angezeigt wird "DP-Mastersystem [1]".
OB	Organisationsbaustein, der dem Prozessabbild zugewiesen ist. Der Zyklus-OB "Main" bezieht sich dabei immer auf die automatische Aktualisierung des Prozessabbildes. Diese Spalte ist nicht bei jeder CPU verfügbar.
TPA	Teilprozessabbild. Zeigt an, ob die Adresse automatisch über das zyklische Prozessabbild aktualisiert wird oder sich in einem Teilprozessabbild befindet, welches explizit für eine Aktualisierung angestoßen werden muss. Siehe die weitere Beschreibung "Teilprozessabbild TPA" unten.

Teilprozessabbild TPA

Die Tabellenspalte "TPA" zeigt die Zuordnung der Adresse zum zyklischen Prozessabbild oder zu einem Teilprozessabbild (TPA).

Bei S7-300/400:

- "OB1-PA": Die Adresse ist dem zyklischen Prozessabbild zugeordnet. Das Betriebssystem aktualisiert diese Adresse automatisch in jedem Programmzyklus.
- "TPA x": Die Adresse ist dem Teilprozessabbild x zugeordnet (z. B. TPA 1, kein zyklisches Prozessabbild). Das Betriebssystem aktualisiert dieses TPA beim Ausführen des zugeordneten OBs. Wenn diesem TPA kein OB zugeordnet ist, dann aktualisiert das Betriebssystem dieses TPA nicht. Sie haben die Möglichkeit, im Anwenderprogramm mit den Anweisungen "UPDAT_PI" und "UPDAT_PO" das TPA selbst zu aktualisieren (bei S7-400 und einigen S7-300 CPU).

Bei S7-1200:

- "Automatische Aktualisierung": Die Adresse ist dem zyklischen Prozessabbild (TPA 0) zugeordnet. Das Betriebssystem aktualisiert diese Adresse automatisch in jedem Programmzyklus.
- "Keines": Die Adresse ist keinem Teilprozessabbild zugeordnet. Sie greifen im Anwenderprogramm direkt auf diese Adresse zu (direkter Peripheriezugriff, kein Prozessabbild).
- "TPA x": Das Betriebssystem aktualisiert dieses TPA beim Ausführen des zugeordneten OBs. Wenn diesem TPA kein OB zugeordnet ist, dann aktualisiert das Betriebssystem dieses TPA nicht. Sie haben die Möglichkeit, im Anwenderprogramm über direkte Peripheriezugriffe Eingänge zu lesen oder Ausgänge zu schreiben. Die Anweisungen "UPDAT_PI", "UPDAT_PO", "SYNC_PI" und "SYNC_PO" werden bei S7-1200 nicht unterstützt.
- "TPA OB Servo": Das Teilprozessabbild "TPA OB Servo" ist keinem Organisationsbaustein zugeordnet (feste Einstellung bei Organisationsbaustein: "---(Keiner)"). Das Betriebssystem aktualisiert dieses TPA und alle darin enthaltenen Adressen nicht: Sie greifen im Anwenderprogramm direkt auf die Adressen zu (direkter Peripheriezugriff).

Bei S7-1500:

- "Automatische Aktualisierung": Die Adresse ist dem zyklischen Prozessabbild (TPA 0) zugeordnet. Das Betriebssystem aktualisiert diese Adresse automatisch in jedem Programmzyklus.
- "None": Die Adresse ist keinem Teilprozessabbild zugeordnet. Sie greifen im Anwenderprogramm direkt auf diese Adresse zu (direkter Peripheriezugriff, kein Prozessabbild).
- "TPA x" (TPA 1 bis TPA 31): Das Betriebssystem aktualisiert TPA x beim Ausführen des zugeordneten OBs. Wenn diesem TPA kein OB zugeordnet ist, dann aktualisiert das Betriebssystem das TPA x nicht. Sie haben die Möglichkeit, im Anwenderprogramm mit den Anweisungen "UPDAT_PI" und "UPDAT_PO" das TPA x zu aktualisieren. Wenn TPA x einem Taktsynchronalarm-OB (OB 61 bis OB 64) zugeordnet ist, dann aktualisiert das Betriebssystem das TPA x nicht: Sie haben die Möglichkeit, in Ihrem Anwenderprogramm das TPA x mit den Anweisungen "SYNC_PI" und "SYNC_PO" zu aktualisieren.
- "TPA OB Servo": Das Teilprozessabbild "TPA OB Servo" ist fest dem OB "MC-Servo" zugeordnet. STEP 7 erzeugt diesen OB automatisch, wenn Sie ein Technologie-Objekt im Bereich Motion Control anlegen. Beim Ausführen des OBs wird das TPA OB Servo taktsynchron aktualisiert. Alle von Motion Control verwendeten Antriebe und Geber sind diesem Teilprozessabbild zugeordnet.

Siehe auch

Ein- und Ausgangsadressen festlegen (Seite 1459)

1.2.18 Dezentrale Peripherie in der Projektnavigation

Dezentrale Peripheriegeräte werden in der Projektnavigation in eigenen Ordnern angezeigt. Abhängig davon, ob ein Gerät einem PROFINET IO-System, einem DP-Mastersystem oder keinem System zugewiesen wurde, wird das Gerät als Knoten oder Verknüpfung zu dem Gerät in einem der nachfolgend beschriebenen Ordnern angezeigt.

Ordner "Nicht gruppierte Geräte" im Projekt

Im Ordner "Nicht gruppierte Geräte" werden alle dezentralen Peripheriegeräte des Projekts zusammengefasst.

Die Symbole der Ordner und Verknüpfungen zeigen den Status "zugeordnet" durch farbige Kennzeichnungen an: PROFINET IO-Devices sind grün gekennzeichnet, PROFIBUS DP-Slaves sind magenta gekennzeichnet, andere dezentrale Peripheriegeräte sind nicht gesondert gekennzeichnet.

Die Symbole der Ordner und Verknüpfungen zeigen den Status "nicht zugeordnet" durch ein Fragezeichen an.

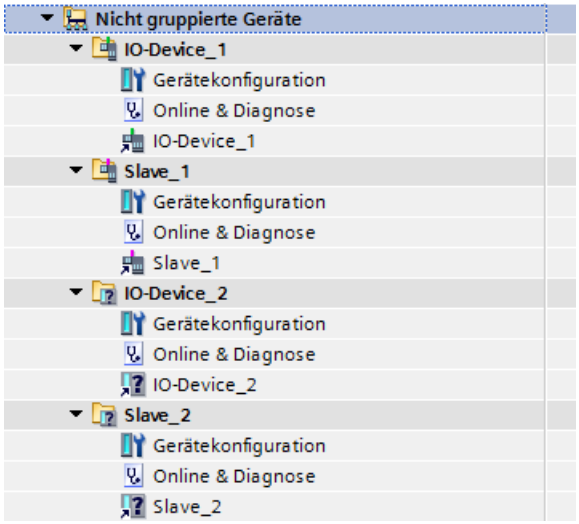

Wenn Sie über die Funktion "Online verbinden" eine Verbindung zu einem IO-Device bzw. einem DP-Slave hergestellt haben, werden die gleichen Diagnoseinformationen (z. B. grüner Haken) für die Verknüpfung eines Geräts wie für das Gerät selbst in der Projektnavigation angezeigt. Weitere Informationen finden Sie im Kapitel "Ansicht im Online-Modus".

Es ist möglich, dezentrale Peripheriegeräte aus dem Ordner "Nicht gruppierte Geräte" an eine andere Stelle im Projektbaum zu kopieren oder zu verschieben, nicht jedoch den Ordner selbst. Mögliche Ziele sind der Projektordner oder ein Geräteordner. Umgekehrt können Sie dezentrale Peripheriegeräte wieder in den Ordner "Nicht gruppierte Geräte" zurück kopieren oder verschieben.

Mit einem Doppelklick auf die Verknüpfung gelangen Sie in die Gerätesicht und können das dezentrale Peripheriegerät konfigurieren.

Ordner "Nicht zugeordnete Geräte" im Projekt

Geräte, die keinem dezentralen Peripheriesystem zugeordnet sind, werden im Ordner "Nicht zugeordnete Geräte" als Verknüpfung angezeigt.

Die Symbole für den Ordner sowie die Verknüpfung des jeweiligen Geräts im Projektbaum zeigen durch ein Fragezeichen, dass sie nicht an ein dezentrales Peripheriesystem angebunden sind. Die Verknüpfung ist durch einen kleinen Pfeil gekennzeichnet.

Die Anzeige im Projektbaum als Verknüpfung erlaubt es, IO-Devices bzw. DP-Slaves einzeln zu selektieren, z. B. um das Verknüpfungsziel zu selektieren.. Die Selektion von Gerätegruppen ist weiterhin möglich.

Mit einem Doppelklick auf die Verknüpfung gelangen Sie in die Gerätesicht und können das dezentrale Peripheriegerät konfigurieren.

Ordner "Dezentrale Peripherie" unter einem Gerät

Wenn Sie ein IO-Device einem PROFINET IO-System zuordnen, wird dieses im Projektbaum als Verknüpfung unterhalb des Knotens "Dezentrale Peripherie > PROFINET IO-System" angezeigt.

Entsprechend wird ein DP-Slave, der einem DP-Mastersystem zugeordnet ist, als Verknüpfung unterhalb des Knotens "Dezentrale Peripherie > DP-Mastersystem" angezeigt.

Die Verknüpfung verweist auf das zugehörige IO-Device bzw. den DP-Slave im Projektbaum.

Die Symbole für den Knoten sowie die Verknüpfung des jeweiligen Geräts im Projektbaum zeigen, ob sie per PROFINET (grün) oder per PROFIBUS (magenta) angebunden sind. Die Verknüpfung ist durch einen kleinen Pfeil gekennzeichnet.

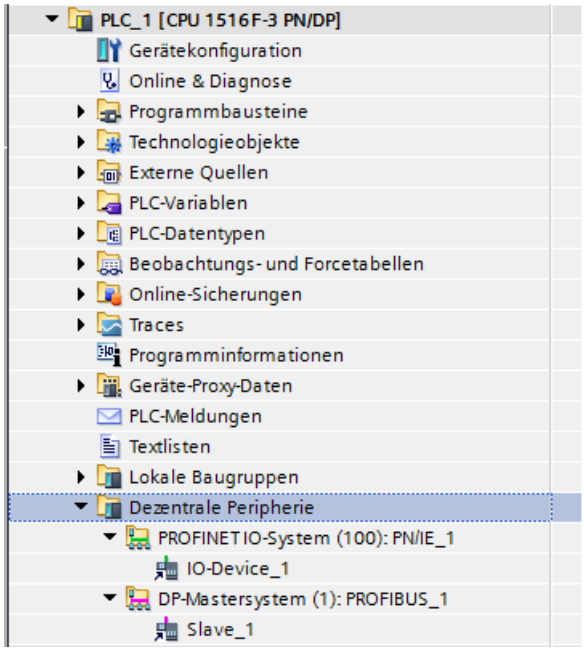

Gateways werden als Link unterhalb des DP-Master-/IO-Knotens angezeigt, dem sie zugeordnet wurden.

Mit einem Doppelklick auf die Verknüpfung gelangen Sie in die Gerätesicht und können das dezentrale Peripheriegerät konfigurieren.

1.2.19 Geräte vergleichen

1.2.19.1 Grundlagen zum Gerätevergleich

Funktion

Sie haben die Möglichkeit, die Hardware-Komponenten zweier Geräte miteinander zu vergleichen, um mögliche Unterschiede zu ermitteln. Dazu können Sie einen Offline/Offline-Vergleich durchführen. Die zu vergleichenden Geräte können dabei aus einem Projekt oder aus verschiedenen Projekten stammen.

Sie können sowohl zentrale als auch dezentrale Peripherie vergleichen. Die Zuordnung der zu vergleichenden Geräte kann entweder automatisch oder manuell erfolgen. Die automatische Zuordnung von zentraler Peripherie erfolgt anhand der Steckplatznummer. Bei dezentraler Peripherie kann die automatische Zuordnung nach den folgenden Kriterien erfolgen:

- Zuordnung über Adresse/HW-Kennung: Die Zuordnung erfolgt über die Adressen oder IDs der Geräte. Dieses Kriterium ist für den Vergleich von Geräten geeignet, die sich innerhalb eines Projekts befinden.
- Zuordnung über Namen: Die Zuordnung erfolgt über die Gerätenamen. Dieses Kriterium ist für den Vergleich von Geräten geeignet, die sich in unterschiedlichen Projekten befinden.

Sie können die Zuordnung entweder selbst festlegen oder das System entscheiden lassen. Im letzten Fall wählt das System die Zuordnung je nach Kontext selbst aus.

Um die Module von dezentraler Peripherie zu vergleichen, starten Sie einen Offline/Offline-Vergleich für die betreffenden dezentralen Stationen.

Siehe auch

Gerätevergleich durchführen (Seite 73)

1.2.19.2 Gerätevergleich durchführen

Vorgehen

Um einen Gerätevergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Öffnen Sie das Register "Hardware".
4. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich.
   Alle vorhandenen Objekte der ausgewählten Geräte werden abhängig von den Einstellungen im Register "Hardware" des Vergleichseditors angezeigt und es wird ein automatischer Vergleich durchgeführt. Den Status der Vergleichsobjekte können Sie an den Symbolen im Vergleichseditor ablesen. Für Submodule finden Sie ein detailliertes Ergebnis im Bereich "Eigenschaftsvergleich".
5. Wenn Sie das Zuordnungskriterium ändern möchten, klicken Sie in der Funktionsleiste auf den Pfeil der Schaltfläche "Verfügbare Zuordungskriterien anzeigen". Wählen Sie dann das Zuordnungskriterium, das Sie verwenden möchten.
6. Wenn Sie einen manuellen Vergleich durchführen möchten, klicken Sie oberhalb des Statusbereichs auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich. Selektieren Sie anschließend die Objekte, die Sie miteinander vergleichen möchten.
   Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen.

Siehe auch

Grundlagen zum Gerätevergleich (Seite 72)

1.3 Netze konfigurieren

1.3.1 Geräte vernetzen

1.3.1.1 Kommunikation und Netze

Kommunikation zwischen Geräten

Grundlage für jede Art von Kommunikation ist immer ein zuvor konfiguriertes Netz. Mit der Netzkonfiguration organisieren Sie die notwendige Erreichbarkeit der verschiedenen Netzteilnehmer für die Kommunikation. Für den Nachrichtenaustausch im S7-Netzwerk gelten folgende Voraussetzungen:

- Versorgung aller Teilnehmer des Netzes mit eindeutigen S7-topologischen Adressen
- Sicherstellen der Verschaltung der Anschaltungen über physikalische Leitungsabschnitte
- Kommunikation der Teilnehmer mit konsistenten Übertragungseigenschaften

Definition

Über das Netzwerk werden die verschiedenen Rechner und Endgeräte (PC, PG, AG, AS) mittels Anschaltungen, physikalischen Leitungen und entsprechender Software mit einander verbunden. Vernetzte Geräte können Daten untereinander austauschen. Die Vernetzung kann durch eine Verdrahtung oder über Funknetz, z. B. über WLAN, ausgeführt sein.

Die verschiedenen Formen der Verdrahtung führen aus der Sicht des Netzwerkes zu verschiedenen S7-Subnetzen. Ein S7-Subnetz ist Teil des gesamten S7-Netzwerkes. Die Parameter der Teilnehmer einer S7-Subnetzes (z. B. bei PROFIBUS) müssen bezüglich der physikalischen Verdrahtung einheitlich sein. Ein physikalisches S7-Subnetz umfasst die Buskomponenten und alle angeschlossenen Geräte. S7-Subnetze auf unterschiedlicher physikalischer Verdrahtung können mittels Gateways oder Routern im S7-Netzwerk gekoppelt werden.

Netzprojektierung

Um das S7-Netzwerk zu projektieren, sind folgende Schritte notwendig:

- Geräte mit einem S7-Subnetz verbinden
- Eigenschaften/Parameter für jedes S7-Subnetz festlegen
- Teilnehmereigenschaften für jede vernetzte Baugruppe festlegen (z.B. eindeutige S7-topologische Adresse)
- Konfigurationsdaten in die Geräte laden, um Schnittstellen mit den aus der S7-Netzprojektierung resultierenden Einstellungen zu versorgen
- S7-Netzkonfiguration dokumentieren

Für die Open User Communication wird das Anlegen und Konfigurieren eines S7-Subnetzes von der Verbindungsparametrierung unterstützt.

Zusammenhang zwischen Netzkonfiguration und Projekt

Innerhalb eines Projekts werden S7-Subnetze mit ihren Eigenschaften verwaltet. Eigenschaften ergeben sich im Wesentlichen aus einstellbaren S7-Netzparametern und aus der Anzahl und den Kommunikationseigenschaften der angeschlossenen Teilnehmer.

Die zu vernetzenden Teilnehmer müssen sich im selben Projekt befinden.

S7-Subnetz-Name und S7-Subnetz-ID

S7-Subnetze werden innerhalb des Projektes durch einen S7-Subnetz-Namen und eine S7-Subnetz-ID eindeutig gekennzeichnet. Die S7-Subnetz-ID ist in allen Komponenten mit vernetzbaren Schnittstellen hinterlegt. Damit können Komponenten auch nach einem Hochladen in ein Projekt eindeutig einem S7-Subnetz zugeordnet werden.

S7-Subnetze gibt es in zwei möglichen Ausprägungen:

- Physikalische Ethernet-Verdrahtung ("grüne Linie")
- Physikalische PROFIBUS-Verschaltung ("violette Linie")

Möglichkeiten der Vernetzung

Sie können im Projekt Geräte mit kommunikationsfähigen Komponenten anlegen und vernetzen. Für die Vernetzung der Geräte gibt es prinzipiell folgende Möglichkeiten:

- Sie verbinden die Schnittstellen der kommunikationsfähigen Komponenten untereinander. Dadurch wird ein zum Typ der Schnittstelle passendes neues S7-Subnetz angelegt.
- Sie verbinden die Schnittstelle der kommunikationsfähigen Geräte mit einem neuen oder einem bereits angelegten S7-Subnetz.
- Sie erstellen eine Verbindung der Open User Communication. Über die Verbindungsparametrierung der Open User Communication wird automatisch ein S7-Subnetz zwischen den Kommunikationspartnern angelegt.
- Sie projektieren Verbindungen über die grafische Verbindungsprojektierung; fehlende Vernetzungen werden hierbei erkannt und werden automatisch oder dialoggeführt angelegt.

Aufgrund unterschiedlicher Aufgaben der Geräte oder aufgrund der Ausdehnung der Anlage kann es erforderlich sein, mehrere S7-Subnetze zu betreiben. Diese S7-Subnetze werden in einem Projekt verwaltet.

1.3.1.2 Geräte in der Netzsicht vernetzen

Möglichkeiten

In der grafischen Netzsicht behalten Sie die Übersicht über die Subnetze der gesamten Anlage im Projekt. Unterstützend können Sie die tabellarische Netzübersicht verwenden.

Um die Schnittstelle einer kommunikationsfähigen Komponente zu vernetzen, haben Sie je nach Ausgangssituation unterschiedliche Möglichkeiten bei der Projektierung. Nachfolgend werden diese Vorgehensweisen beschrieben:

- Einzelnes Subnetz erstellen
- Mehrere Subnetze gleichzeitig erstellen
- Zwei Zielgeräte über ein neues Subnetz verbinden
- Geräte mit bestehendem Subnetz verbinden
- Vorhandenes Subnetz aus Liste auswählen
- Automatische Vernetzung während der Verbindungsprojektierung;
  Siehe auch: AUTOHOTSPOT

Mögliche Ausgangssituationen sind:

- Es ist noch kein geeignetes Subnetz vorhanden.
- Das Subnetz, mit dem Sie die Komponente verbinden wollen, ist bereits vorhanden.

Vorgehen - Einzelnes Subnetz erstellen

Um ein Subnetz zu erzeugen und es mit einer Schnittstelle zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle einer CPU / eines CP.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Subnetz erstellen".

Die selektierte Schnittstelle wird mit einem neuen Subnetz verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Im folgenden Bild sehen Sie eine Schnittstelle mit ausgehender Verbindungslinie zu einem Subnetz:

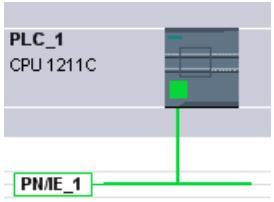

Vorgehen - Mehrere Subnetze gleichzeitig erstellen

Um mehrere Subnetze gleichzeitig zu erzeugen, gehen Sie folgendermaßen vor:

1. Selektieren Sie mehrere Schnittstellen durch Mausklick und gleichzeitiges Drücken der Taste <Strg>.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Subnetz erstellen".

Jede selektierte Schnittstelle wird mit einem neuen Subnetz verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch

oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Im folgenden Bild sehen Sie mehrere angelegte Subnetze durch eine Mehrfachselektion der Schnittstellen:

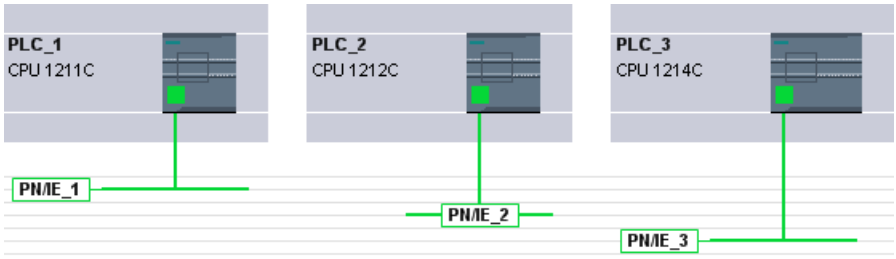

Vorgehen - Zwei Zielgeräte über ein neues Subnetz verbinden

Um eine Schnittstelle mit einem anderen Gerät über ein noch nicht vorhandenes Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf die zu vernetzende Schnittstelle einer kommunikationsfähigen Komponente.
2. Klicken Sie mit der linken Maustaste und halten Sie die Maustaste gedrückt.
3. Verschieben Sie den Mauscursor.
   Der Mauscursor zeigt jetzt mit dem Vernetzungssymbol den Arbeitsmodus "Vernetzen" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst über einer gültigen Zielposition verschwindet.

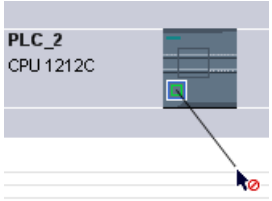

4. Ziehen Sie nun den Mauscursor im Vernetzungsmodus auf die Schnittstelle des Zielgeräts. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.
5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste.

Hinweis

Wenn Sie den Vernetzungsmodus vorher beenden möchten, drücken Sie <Esc>, klicken Sie die rechte Maustaste oder doppelklicken Sie in den Hintergrund der Netzsicht.

Ein neues Subnetz wird angelegt. Die Schnittstellen sind jetzt über das neue Subnetz verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Im folgenden Bild sehen Sie zwei vernetzte Geräte:

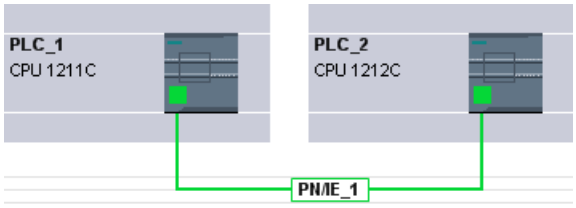

Vorgehen - Geräte mit bestehendem Subnetz verbinden

Um eine Schnittstelle mit einem bereits bestehendem Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf der zu vernetzenden Schnittstelle einer kommunikationsfähigen Komponente oder auf dem vorhandenen Subnetz.
2. Klicken Sie mit der linken Maustaste und halten Sie die Maustaste gedrückt.
3. Verschieben Sie den Mauscursor.
   Der Mauscursor zeigt jetzt mit dem Vernetzungssymbol den Arbeitsmodus "Vernetzen" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst bei Platzierung auf einer gültigen Zielposition verschwindet.
4. Ziehen Sie nun den Mauscursor auf das vorhandene Subnetz oder auf die zu vernetzende Schnittstelle. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.

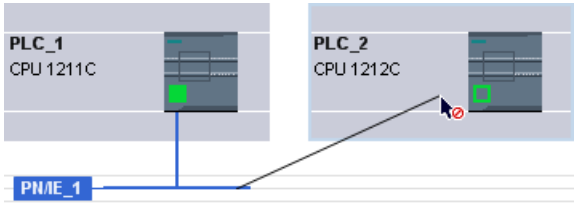

5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste. Wenn Sie den Vernetzungsmodus vorher beenden möchten, drücken Sie <Esc> oder klicken Sie die rechte Maustaste.

Ergebnis:

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Vorgehen - Vorhandenes Subnetz aus Liste auswählen

Um eine Schnittstelle mit einem bereits angelegten Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle einer CPU.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Neuem Subnetz zuweisen". Es erscheint eine Auswahlliste mit den vorhandenen Subnetzen.
3. Wählen Sie aus der Liste ein Subnetz aus.

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

1.3.1.3 Tabellarische Netzübersicht

Tabellarischer Bereich der Netzsicht

Der tabellarische Bereich der Netzsicht besteht aus mehreren Tabellen:

- Netzübersicht: Die Tabelle für die Netzübersicht ergänzt die grafische Netzsicht mit folgenden Funktionen:
  - Sie erhalten Detailinformationen über den Aufbau und die Parametrierung der Geräte.
  - Sie können über die Spalte "Subnetz" kommunikationsfähige Komponenten mit angelegten Subnetzen verbinden.
- Verbindungen: Kommunikationsverbindungen werden in dieser Tabelle angezeigt. Einige Aspekte der hier angezeigten Verbindungen können direkt in der Tabelle bearbeitet werden. Selektierte Verbindungen werden im Inspektorfenster angezeigt und können dort bearbeitet werden.
- Relationen: Logische Verbindungen der HMI-Client-Server-Relation werden hier angezeigt.
- E/A-Kommunikation: Gerätekommunikation über PROFIBUS DP und PROFINET IO wird in dieser Tabelle konfiguriert und angezeigt. In der E/A-Kommunikationstabelle werden nur diejenigen Schnittstellen eines Gerätes angezeigt, die für eine Kommunikation über PROFIBUS DP oder PROFINET IO verwendet werden können.
- VPN: Eigenschaften der VPN-Gruppen und ihrer zugeordneten Security-Module werden hier angezeigt.

Grundfunktionen für Tabellen

Die Netzübersicht unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden
  Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Tabelle sortieren
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen

1.3.1.4 Geräte in der Gerätesicht vernetzen

Vernetzung in der Gerätesicht

In der Gerätesicht können Sie alle Parameter der zu einem Gerät gehörigen Komponenten und der Schnittstellen detailliert prüfen und einstellen. Auch die Zuordnung der Schnittstellen auf die im Projekt angelegten Subnetze ist hier möglich.

Voraussetzung

- Das Subnetz, mit dem eine Schnittstelle verbunden werden soll, ist bereits angelegt.
- Falls das Subnetz noch nicht angelegt ist, wechseln Sie in die Netzsicht und führen dort die Vernetzung durch.

Vorgehen - Mit vorhandenem Subnetz verbinden

Um die Schnittstelle mit einem bereits angelegten Subnetz zu verbinden, gehen Sie in der Gerätesicht folgendermaßen vor:

1. Selektieren Sie die komplette kommunikationsfähigen Komponente oder die zu vernetzende Schnittstelle.
   Im Inspektorfenster werden die Eigenschaften der gewählten Schnittstelle oder Komponente angezeigt.
2. Wählen Sie im Inspektorfenster die zur gewählten Schnittstelle gehörende Parametergruppe aus, z. B. bei einer PROFINET-Schnittstelle die Parameter-Gruppe "Ethernet-Adressen".
3. Wählen Sie unter "Schnittstelle vernetzt mit" das zu verbindenden Subnetz aus der Klappliste "Subnetz" aus.

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Vorgehen - Neues Subnetz anlegen

Um ein Subnetz zu erzeugen und es mit der Schnittstelle zu verbinden, gehen Sie in der Gerätesicht folgendermaßen vor:

1. Selektieren Sie die komplette kommunikationsfähigen Komponente oder die zu vernetzende Schnittstelle.
   Im Inspektorfenster werden die Eigenschaften der gewählten Schnittstelle oder Komponente angezeigt.
2. Wählen Sie im Inspektorfenster die zur gewählten Schnittstelle gehörende Parametergruppe aus, z. B. bei einer PROFINET-Schnittstelle die Parameter-Gruppe "Ethernet-Adressen".
3. Klicken Sie unter "Schnittstelle vernetzt mit" auf die Schaltfläche "Neues Subnetz hinzufügen".

Die Schnittstelle wird mit einem neuen Subnetz des geeigneten Subnetztyps verbunden. Die sich aus der geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

1.3.1.5 Netzparameter und Schnittstellenparameter prüfen oder ändern

Einführung

Die Kommunikationsfähigkeit zwischen vernetzten Geräten erfordert die Konfiguration folgender Parameter:

- Netzparameter
  Netzparameter identifizieren das Netz innerhalb der Anlagen-Projektierung, z. B. über einen Namen.
- Schnittstellenparameter
  Schnittstellenparameter legen spezifische Eigenschaften einer kommunikationsfähigen Komponente fest. Eine mit den Netzparametern konsistente Einstellung der Adressen und der Übertragungseigenschaften wird automatisch hergestellt.

Hinweis

Netzparameter und Schnittstellenparameter sind bei der Vernetzung standardmäßig so eingestellt, dass die Kommunikation für viele Anwendungsfälle ohne Parameteränderung möglich ist. Die sich aus einer geänderten Konfiguration möglicherweise ergebenden Inkonsistenzen werden automatisch oder per Abfrage geklärt. Bei Konsistenzproblemen wird vorzugsweise die nächste freie Adresse verwendet.

Vorgehen - Netzparameter prüfen oder ändern

Um Netzparameter zu prüfen oder zu ändern, gehen Sie folgendermaßen vor:

1. Gehen Sie in die Netzsicht.
2. Selektieren Sie in der Netzsicht das Subnetz.
   Im Inspektorfenster sind die Netzparameter im Register "Eigenschaften" sichtbar.
3. Prüfen oder modifizieren Sie bei Bedarf die Netzparameter in der jeweiligen Parametergruppe.

Vorgehen - Schnittstellenparameter prüfen oder ändern

Sie können Schnittstellenparameter in der Netz- und Gerätesicht prüfen und ändern.

Um Schnittstellenparameter zu prüfen oder zu ändern, gehen Sie folgendermaßen vor:

1. Gehen Sie in die Netzsicht oder die Gerätesicht.
2. Selektieren Sie die Schnittstelle.
   Im Inspektorfenster sind die Schnittstellenparameter im Register "Eigenschaften" sichtbar.
3. Prüfen oder modifizieren Sie bei Bedarf die Schnittstellenparameter in der jeweiligen Parametergruppe.

Vorgehen - Schnittstellenadressen prüfen oder ändern

Sie können die MPI-, PROFIBUS und Ethernet-Adressen direkt in der grafischen Netzsicht bearbeiten.

Um MPI-, PROFIBUS und Ethernet-Adressen in der Netzsicht zu bearbeiten, gehen Sie folgendermaßen vor:

1. Aktivieren Sie in der Netzsicht die Anzeige der Schnittstellenadressen.
2. Selektieren Sie eine zu bearbeitende MPI-, PROFIBUS oder Ethernet-Adresse.
3. Klicken Sie auf das selektierte Adressfeld oder drücken Sie [F2], um die Adresse direkt in der Netzsicht zu bearbeiten.

1.3.1.6 Vernetzungen ändern

Einführung

Sie können den Netzanschluss einer Schnittstelle aufheben oder einem anderen Subnetz des selben Subnetztyps zuweisen.

Auswirkungen

Je nach Ausführung sind zu unterscheiden:

- Netzanschluss einer Schnittstelle aufheben
  Die konfigurierten Parameter der Schnittstelle bleiben unverändert.
- Netzanschluss einem anderen Subnetz zuweisen
  Wenn die Adressen im zugewiesenen Subnetz nicht eindeutig, d. h. bereits vorhanden sind, werden sie automatisch eindeutig gemacht.

Vorgehen - Netzanschluss aufheben

Um den Netzanschluss einer Schnittstelle aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die vernetzte Schnittstelle.

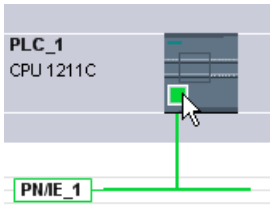

2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Vom Subnetz trennen"

Der Netzanschluss wird gelöscht, Adressen der Schnittstelle werden jedoch nicht verändert.

Projektierte Verbindungen bleiben erhalten, sind in der Verbindungstabelle jedoch wegen der fehlenden Vernetzung rot markiert. Spezifizierte Verbindungen bleiben spezifiziert.

Siehe auch

Geräte in der Netzsicht vernetzen (Seite 75)

1.3.1.7 Subnetze kopieren, ausschneiden oder löschen

Einführung

Subnetze können Sie als einzelne Objekte oder im Verbund mit vernetzten Geräten oder zusammen mit weiteren Netzen kopieren.

Damit können Sie z. B. komplexe Konfigurationen, die innerhalb des Projektes in mehrfacher Ausprägung angeordnet werden sollen, ohne zusätzlichen Aufwand anlegen.

Auswirkungen auf das kopierte Subnetz

Eigenschaften, die innerhalb eines Projektes eindeutig vergeben sein müssen, werden beim Kopieren den kopierten Objekten entsprechend neu zugewiesen.

Für Subnetze bedeutet dies: Die Subnetz-ID und der Subnetz-Name werden dem kopierten Subnetz neu zugewiesen.

Die konfigurierten Eigenschaften werden in das kopierte Subnetz übernommen.

Vorgehen - Subnetz kopieren

Um ein oder mehrere Subnetze zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Subnetze.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Die kopierten Subnetze werden im oberen Bereich der Netzsicht als "verwaiste" Subnetze angezeigt.

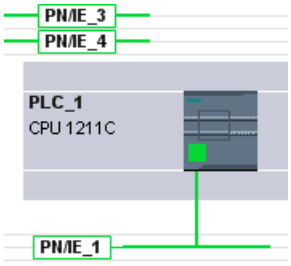

Vorgehen - Subnetz mit angeschlossenen Geräten kopieren

Um ein oder mehrere Subnetze mit vernetzten Geräten zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Subnetze mit den angeschlossenen Geräten - z. B. mit der Lasso-Funktion.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Es werden vollständige Kopien der Subnetze und der angeschlossenen Geräte angelegt.

Projektierte Verbindungen werden übernommen und bleiben innerhalb der kopierten Geräte erhalten. Verbindungen zu nicht mitkopierten Geräten werden aufgebrochen und sind damit unspezifiziert.

1.3.1.8 Netzkonfiguration MPI

MPI-Adressvergabe

Für Geräte mit einer MPI-Schnittstelle gilt, dass sämtliche Teilnehmer eines Subnetzes eine unterschiedliche Teilnehmeradresse besitzen müssen.

CPUs mit MPI-Adresse werden mit der Default-MPI-Adresse 2 ausgeliefert. Da Sie diese Adresse nur einmal im MPI-Subnetz benutzen dürfen, müssen Sie die voreingestellte Adresse in allen anderen CPUs ändern.

Für Geräte mit den Artikel-Nr. 6ES7 3xx-xxxxx-0AB0 gilt:

Bei der Planung der MPI-Adressen für mehrere CPUs müssen Sie "MPI-Adresslücken" für FMs und CPs mit eigenen MPI-Adressen vorsehen, um Doppelbelegung von Adressen zu vermeiden.

Erst wenn sämtliche Baugruppen eines Subnetzes unterschiedliche Adressen besitzen und Ihr tatsächlicher Aufbau mit der erstellten Netzkonfiguration übereinstimmt, sollten Sie die Einstellungen über das Netz laden.

Regeln für die MPI-Adressvergabe

- Vergeben Sie die MPI-Adressen in aufsteigender Reihenfolge.
- Reservieren Sie die MPI-Adresse 0 für ein PG.
- Bis zu 126 (adressierbare) Teilnehmer können Sie in einem MPI-Subnetz miteinander verbinden; bis zu 8 Teilnehmer bei einer Übertragungsgeschwindigkeit von 19,2 kbit/s.
- Alle MPI-Adressen eines MPI-Subnetzes müssen unterschiedlich sein.

Weitere Regeln zum Aufbau eines Netzes finden Sie in den Handbüchern zum Aufbauen von Automatisierungssystemen.

1.3.1.9 Netzkonfiguration PROFIBUS

PROFIBUS-Adressen

Regeln für die Netzkonfiguration

Sämtliche Teilnehmer eines Subnetzes müssen eine unterschiedliche PROFIBUS-Adresse besitzen.

Erst wenn sämtliche Baugruppen eines Subnetzes unterschiedliche Adressen besitzen und Ihr tatsächlicher Aufbau mit der erstellten Netzkonfiguration übereinstimmt, sollten Sie die Einstellungen über das Netz laden.

Am PROFIBUS-Subnetz können Teilnehmer angeschlossen sein, die über projektierte Verbindungen kommunizieren oder die zu einem PROFIBUS DP-Mastersystem gehören.

Zur Konfiguration eines DP-Mastersystems finden Sie weitere Informationen in den nachfolgenden Kapiteln.

Voraussetzung

Die CPU 121xC ist ab der Firmware-Version 2.0 PROFIBUS-fähig.

Regeln für die PROFIBUS-Adressvergabe

- Vergeben Sie die PROFIBUS-Adressen in aufsteigender Reihenfolge.
- Reservieren Sie die PROFIBUS-Adresse "0" für ein Programmiergerät.
- Vergeben Sie für jeden Teilnehmer am PROFIBUS-Netz bzw. für jeden DP-Master und jeden DP-Slave im PROFIBUS-Netz eine eindeutige PROFIBUS-Adresse im Bereich 0 bis 126.
- Es gibt Baugruppen, bei denen die kleinste einstellbare Adresse größer als 1 sein muss.
- Alle PROFIBUS-Adressen eines PROFIBUS-Subnetzes müssen unterschiedlich sein.

Weitere Regeln zum Aufbau eines Netzes finden Sie in den Handbüchern zum Aufbauen von Automatisierungssystemen, z. B. SIMATIC S7-1200.

Hinweis

PROFIBUS-Adresse "0"

Reservieren Sie die PROFIBUS-Adresse "0" für ein Programmiergerät, das Sie später für Servicezwecke kurzzeitig an das PROFIBUS-Netz anschließen.

Siehe auch

Wissenswertes zu PROFIBUS-Busparametern (Seite 86)

Wissenswertes zu PROFIBUS-Busparametern

Parameter aufeinander abgestimmt

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Sie sollten die voreingestellten Werte deshalb nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind.

Hinweis

Die Busparameter sind in Abhängigkeit vom Busprofil einstellbar oder nicht einstellbar. Wenn die Busparameter nicht einstellbar sind, dann sind sie gegraut dargestellt. Es werden immer die Offline-Werte der Busparameter angezeigt, auch wenn Sie online mit dem Zielsystem verbunden sind.

Die angezeigten Parameter gelten für das gesamte PROFIBUS-Subnetz und sind im Folgenden kurz erklärt.

Zyklisches Verteilen der Busparameter einschalten

Wenn bei ausgewähltem PROFIBUS-Subnetz im Inspektorfenster unter "Busparameter" das Optionskästchen "Zyklisches Verteilen der Busparameter einschalten" aktiviert ist, dann werden die Busparameter im Betrieb zyklisch von den Baugruppen gesendet, die diese Funktion unterstützen. Auf diese Weise kann z. B. ein PG ohne Probleme im laufenden Betrieb an den PROFIBUS angeschlossen werden.

Sie sollten diese Funktion deaktivieren:

- Bei heterogenem PROFIBUS-Subnetz (genauer: bei angeschlossenen Fremdgeräten, deren Protokoll den DSAP 63 für Multicast benutzt)
- Bei Äquidistanz-Betrieb (Buszyklus minimieren!)

Busparameter für das Busprofil von PROFIBUS-Subnetzen

Busparameter	Einstellbar?	Grenzwerte
Tslot_Init	Ja	Max. Tsdr + 15 <= Tslot_init <= 16.383 t_bit
Max. Tsdr	Ja	35 + 2*Tset + Tqui <= Max. Tsdr <= 1.023 t_bit
Min. Tsdr	Ja	11 t_bit <= Min. Tsdr <= MIN(255 t_bit, ... Max. Tsdr - 1, 34 + 2*Tset + Tqui)
Tset	Ja	1 t_bit <= Tset <= 494 t_bit
Tqui	Ja	0 t_bit <= Tqui <= MIN(31 t_bit, Min. Tsdr - 1)
GAP-Faktor	Ja	1 <= GAP-Faktor <= 100
Retry Limit	Ja	1 <= Retry Limit <= 15
Tslot	Nein	---
Tid2	Nein	Tid2 = Max. Tsdr
Trdy	Nein	Trdy = Min. Tsdr
Tid1	Nein	Tid1 = 35 + 2*Tset + Tqui

Busparameter	Einstellbar?	Grenzwerte
Ttr	Ja	256 t_bit <= Ttr <= 16.777.960 t_bit
Ttr typisch	Nein	Diese Zeit dient nur der Information und wird nicht an die Teilnehmer übertragen.
Ansprechüberwachung		10 ms <= Ansprechüberwachung (Watchdog) <= 650 s

Wenn Sie ein benutzerdefiniertes Busprofil erstellen wollen, dann empfehlen wir folgende Einstellungen:

- minimale Target-Rotation-Time (Ttr) = 5000mal HSA (höchste PROFIBUS-Adresse)
- minimale Ansprechüberwachung (Watchdog) = 6250mal HSA

Neu berechnen

Über die Schaltfläche "Neu berechnen" können Sie die Parameter neu berechnen lassen.

Siehe auch

PROFIBUS-Adressen (Seite 85)

Beschreibung der Busparameter (Seite 87)

Beschreibung der Busparameter

Detaillierte Beschreibung der PROFIBUS-Busparameter

Busparameter	Bedeutung
Tslot_Init	Die Warte-auf-Empfang-Zeit legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten. Wird in der Parametergruppe "Leitungskonfiguration" der Einfluss der Leitungs-Komponenten auf die Telegrammlaufzeiten konfiguriert, so müssen diese Anteile zusätzlich berücksichtigt werden. Der Anteil wird auf die vorgegebene Tslot_Init aufaddiert und die Summe als Tslot verwendet.
Max. Tsdr	Die maximale Protokoll-Bearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer spätestens geantwortet haben muss.
Min. Tsdr	Die minimale Protokoll-Bearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer frühestens antworten darf.
Tset	Die Auslösezeit ist die Zeit, die zwischen dem Empfang eines Datentelegramms und der Reaktion darauf im Teilnehmer verstreichen darf.
Tqui	Die Modulator-Ausklingzeit ist die Zeit, die ein sendender Teilnehmer nach Telegrammende für das Umstellen von Senden auf Empfangen benötigt.
GAP-Faktor	Der GAP-Aktualisierungsfaktor legt fest, nach wie vielen Token-Umläufen ein neu hinzugekommener, aktiver Teilnehmer in den logischen Tokenring aufgenommen werden kann.
Retry Limit	Mit dem Parameter wird festgelegt, wie viele Versuche (Telegrammwiederholungen) maximal unternommen werden, um einen Teilnehmer zu erreichen.

Busparameter	Bedeutung
Tslot	Die Warte-auf-Empfang-Zeit (slot-time) legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten. Wird im Register "Leitungen" der Einfluss der Busphysik-Komponenten auf die Telegrammlaufzeiten eingetragen, so müssen diese Anteile zusätzlich berücksichtigt werden. Der Anteil wird auf die vorgegebene Tslot_Init aufaddiert und die Summe als Tslot verwendet.
Tid2	Die Ruhezeit 2 legt fest, nach welcher Zeit ein sendender Teilnehmer nach der Versendung eines nicht quittierten Telegramms frühestens das nächste Telegramm versenden darf.
Trdy	Die Bereitschaftszeit gibt an, nach welcher Zeit ein sendender Teilnehmer frühestens ein Antworttelegramm empfangen kann.
Tid1	Die Ruhezeit 1 legt fest, nach welcher Zeit ein sendender Teilnehmer nach dem Empfang einer Antwort frühestens das nächste Telegramm versenden darf.
Ttr	Die Soll-Token-Umlaufzeit ist die maximal zur Verfügung gestellte Zeit für einen Token-Umlauf. In dieser Zeit erhalten alle aktiven Teilnehmer (DP-Master etc.) einmal das Senderecht (Token). Die Differenz zwischen der Soll-Token-Umlaufzeit und der tatsächlichen Token-Haltezeit eines Teilnehmers bestimmt, wie viel Zeit den anderen aktiven Teilnehmern (PG, weitere DP-Mastern etc.) für das Senden von Telegrammen übrig bleibt.
Ttr typisch	Die typische Datenzykluszeit ist die durchschnittliche Reaktionszeit am Bus, wenn alle projektierten Slaves mit dem DP-Master Daten austauschen. Keiner der Slaves meldet Diagnose und es erfolgt kein zusätzlicher Telegrammverkehr mit PGs oder anderen aktiven Teilnehmern etc. am Bus.
Ansprechüberwachung	Die Ansprechüberwachungszeit wird nur für PROFIBUS-DP-Bussysteme benötigt. Sie legt fest, nach welcher Zeit ein DP-Slave spätestens von seinem DP-Master mit einem neuen Datentelegramm angesprochen werden muss. Geschieht dies nicht, so geht der DP-Slave davon aus, dass der DP-Master ausgefallen ist und setzt seine Ausgänge in einen sicheren Zustand zurück.

Siehe auch

Wissenswertes zu PROFIBUS-Busparametern (Seite 86)

Busprofile bei PROFIBUS

Einführung

Je nach angeschlossenen Gerätetypen und verwendeten Protokollen am PROFIBUS stehen unterschiedliche Profile zur Verfügung. Die Profile unterscheiden sich hinsichtlich der Einstellmöglichkeiten und hinsichtlich der Berechnung der Busparameter. Im Folgenden werden die Profile erläutert.

Teilnehmer mit unterschiedlichen Profilen am selben PROFIBUS-Subnetz

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Busparameter aller Teilnehmer dieselben Werte haben. Wenn z. B. an einem Subnetz sowohl DP- als auch FMS-Dienste genutzt werden, dann müssen immer die "langsameren" Busparametersätze für alle Teilnehmer eingestellt werden, also auch für DP-Teilnehmer das Profil "Universell (DP/FMS)"!

Profile und Übertragungsgeschwindigkeiten

Profile	Unterstützte Übertragungsgeschwindigkeiten in kbit/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Standard	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universell (DP-FMS)	9,6 19,2 93,75 187,5 500 1500
Benutzerdefiniert	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000

Bedeutung der Profile

Profil	Bedeutung
DP	Wählen Sie das Busprofil "DP", wenn nur Geräte an das PROFIBUS-Subnetz angeschlossen sind, welche die Anforderungen der Norm EN 50170 Volume 2/3, Part 8-2 PROFIBUS erfüllen. Die Einstellung der Busparameter ist auf diese Geräte optimiert. Hierzu zählen Geräte mit DP-Master- und DP-Slave-Schnittstellen der SIMATIC S7 sowie Dezentrale Peripheriegeräte anderer Hersteller.
Standard	Gegenüber dem Profil "DP" bietet das Profil "Standard" zusätzlich die Möglichkeit, Teilnehmer eines anderen Projektes oder Teilnehmer, die hier nicht projektiert wurden, bei der Berechnung der Busparameter zu berücksichtigen. Die Busparameter werden dann nach einem einfachen, nicht optimierten Algorithmus berechnet.

Profil	Bedeutung
Universell (DP/FMS)	Wählen Sie das Busprofil "Universell (DP/FMS)", wenn einzelne Teilnehmer am PROFIBUS-Subnetz den Dienst PROFIBUS-FMS nutzen. Hierzu gehören z. B. folgende Geräte: • CP 343-5 (SIMATIC S7) • PROFIBUS-FMS-Geräte anderer Hersteller Auch hier besteht wie im Profil "Standard" die Möglichkeit, zusätzliche Teilnehmer bei der Berechnung der Busparameter zu berücksichtigen.
Benutzerdefiniert	Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Wählen Sie das Busprofil "Benutzerdefiniert", wenn zum Betrieb eines PROFIBUS-Gerätes keines der übrigen Profile "passt" und Sie die Busparameter Ihrem speziellen Aufbau anpassen müssen. Informationen hierzu sind der Dokumentation des PROFIBUS-Gerätes zu entnehmen. Sie sollten die voreingestellten Werte nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind. Auch mit diesem Busprofil sind nicht alle theoretisch einstellbaren Kombinationen eingebbar. Die PROFIBUS-Norm schreibt einige Parametergrenzwerte in Abhängigkeit von anderen Parametern fest vor. So ist es z. B. nicht zulässig, dass ein Responder bereits antworten darf (Min Tsdr), bevor der Initiator das Telegramm auch empfangen kann (Trdy). Diese Normvorgaben werden auch im Profil "Benutzerdefiniert" überprüft. Tipp: Als benutzerdefiniert werden automatisch immer die Busparameter eingestellt, die zuletzt am PROFIBUS-Subnetz gültig waren. Wenn beispielsweise das Busprofil "DP" für das Subnetz gültig war, dann werden die Busparameter für "DP" im Busprofil "Benutzerdefiniert" eingestellt. Ausgehend davon können Sie die Parameter modifizieren. Die Neuberechnung der Überwachungszeiten wird nicht automatisch durchgeführt, um die Gleichheit der eingestellten Werte z. B. zu Projektierungen anderer Projektierungswerkzeuge nicht ohne ihr Wissen zu gefährden. Die Überwachungszeiten Ttr und Ansprechüberwachung können Sie sich auf Basis der von Ihnen eingestellten Parameter berechnen lassen: Klicken Sie hierzu auf die Schaltfläche "Neu berechnen".

Hinweis

Bei allen PROFIBUS-Profilen ist sowohl Mono-Master-Betrieb als auch Multi-Master-Betrieb möglich.

Wissenswertes zur PROFIBUS-Leitungskonfiguration

Leitungskonfiguration und Busparameter

Angaben zur Leitungskonfiguration können bei der Berechnung der Busparameter berücksichtigt werden. Hierzu müssen Sie das Optionskästchen "Leitungskonfiguration berücksichtigen" in den Eigenschaften des PROFIBUS-Subnetzes aktivieren.

Die weiteren Angaben hängen dann vom Typ der verwendeten Leitung ab; einstellbar sind:

- Kupferleitung
- Lichtwellenleiter / optischer Ring

PROFIBUS-Leitungskonfiguration, optischer Ring

Die Berechnung ist abhängig von den eingesetzten OLM-Typen. Die Auswahl erfolgt durch Aktivieren der Optionskästchen (Mehrfach-Aktivierung ist möglich, mindestens ein OLM-Typ muss ausgewählt werden):

- OLM/P12
- OLM/G12
- OLM/G12-EEC
- OLM/G12-1300

Folgende Anpassungen der Busparameter werden durchgeführt:

- Projektierung eines nicht vorhandenen Teilnehmers

  Hinweis

  Die folgenden Einschränkungen gelten bei optischen Ringen auch für passive Teilnehmer (z. B. DP-Slaves):

  Es dürfen maximal HSA-1 Teilnehmer am PROFIBUS Netz angeschlossen sein. Bei einer HSA von 126 dürfen die Adressen 126 und 125 nicht benutzt werden. Damit sind maximal 125 Teilnehmer am Bus möglich (Nr. 0 bis 124).

  Wenn HSA kleiner bzw. gleich 125 ist, dann dürfen die Adressen HSA und größer nicht benutzt werden. Die Adresse HSA-1 darf dagegen benutzt werden.

- Erhöhung des Retry-Werts auf 3
- Einstellung der für den Ring-Betrieb erforderlichen Mindest-Slotzeit

  Hinweis

  Für OLM/P12 sind kurze, für OLM/G12 und OLM/G12-EEC mittlere sowie für OLM/G12-1300 hohe Slotzeit-Werte erforderlich. Hieraus ergeben sich hohe Performance für kleine Netzausdehnung bzw. mittlere bis niedrige Performance bei mittlerer bis großer Netzausdehnung.

PROFIBUS-Kommunikationslast

Kommunikationslast - zusätzliche Netzteilnehmer berücksichtigen

Die Busparameter sind vom Kommunikationsaufkommen der aktiven Netzteilnehmer abhängig. Hierbei gibt es Unterschiede zwischen der zyklischen Kommunikation (DP) und der verbindungsorientierten, azyklischen Kommunikation (S7-Kommunikation, Send/Receive (FDL), FMS). Im Gegensatz zu DP ist die Anzahl und Größe der Kommunikationsaufträge (Kommunikationslast) vom Anwenderprogramm abhängig. Daher ist die Kommunikationslast nicht immer automatisch ermittelbar.

Zur Berechnung der Buszeiten können Sie eine Netzkonfiguration in der Parametergruppe "zusätzliche Netzteilnehmer" festlegen, die von der projektierten Netzkonfiguration abweicht.

Profil berücksichtigen

Die Netzkonfiguration ist festlegbar für die Profile "Standard", "Universell (DP/FMS)" und "Benutzerdefiniert". Beim Profil "DP" sind die Parameter in der Parametergruppe "zusätzliche Netzteilnehmer" nicht eingebbar.

Quantifizierung der Kommunikationslast

Zur Berücksichtigung der Kommunikationslast sind folgende Einstellungen möglich:

- Angaben zur Anzahl nicht projektierter Netzteilnehmer;
- Angaben zu der aus den Anwenderprogrammen für FDL- oder S7-Kommunikation resultierenden Kommunikationslast. Hierzu können Sie aus folgenden Stufen wählen:
  - Niedrig
    Typisch für DP, keine größere Datenkommunikation außer DP.
  - Mittel
    Typisch für Mischbetrieb von DP und anderen Kommunikationsdiensten (z. B. S7-Kommunikation), wenn DP hohe Zeitanforderungen hat und bei mittlerem azyklischen Kommunikationsaufkommen.
  - Hoch
    Für Mischbetrieb von DP und anderen Kommunikationsdiensten (z. B. S7-Kommunikation), wenn DP geringe Zeitanforderungen hat und bei hohem azyklischen Kommunikationsaufkommen.

1.3.1.10 Netzkonfiguration Industrial Ethernet

Regeln für die Netzkonfiguration

Die Ethernet-Schnittstellen der Geräte besitzen als IP-Adresse eine Default-Adresse, die von Ihnen geändert werden kann.

IP-Adresse

Die IP-Parameter sind sichtbar, wenn die kommunikationsfähige Baugruppe das TCP/IP-Protokoll unterstützt. Dies ist in der Regel bei allen Ethernet-Baugruppen der Fall.

Die IP-Adresse besteht aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255. Die Dezimalzahlen sind durch einen Punkt voneinander getrennt.

Beispiel: 140.80.0.2

Die IP-Adresse setzt sich zusammen aus:

- Adresse des IP-Subnetzes
- Adresse des Teilnehmers (im allgemeinen auch Host oder Netzknoten genannt)

Subnetzmaske

Die Subnetzmaske trennt diese beiden Adressen. Sie bestimmt, welcher Teil der IP-Adresse das Netz adressiert und welcher Teil der IP-Adresse den Teilnehmer adressiert.

Die gesetzten Bits der Subnetzmaske bestimmen den Netzteil der IP-Adresse.

Beispiel:

Subnetzmaske: 255.255.0.0 = 11111111.11111111.00000000.00000000

Im Beispiel für die oben genannte IP-Adresse hat die hier gezeigte Subnetzmaske folgende Bedeutung:

Die ersten 2 Bytes der IP-Adresse bestimmen das Subnetz - also 140.80. Die letzten beiden Bytes adressieren den Teilnehmer - also 0.2.

Allgemein gilt:

- Die Netzadresse ergibt sich aus der UND-Verknüpfung von IP-Adresse und Subnetzmaske.
- Die Teilnehmeradresse ergibt sich aus der UND-NICHT-Verknüpfung von IP-Adresse und Subnetzmaske.

Zusammenhang IP-Adresse und Default-Subnetzmaske

Es gibt eine Vereinbarung hinsichtlich der Zuordnung von IP-Adressbereichen und so genannten "Default-Subnetzmasken". Die erste Dezimalzahl der IP-Adresse (von links) bestimmt den Aufbau der Default-Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt:

IP-Adresse (dez.)	IP-Adresse (bin.)	Adressklasse	Default-Subnetzmaske
0 bis 126	0xxxxxxx.xxxxxxxx....	A	255.0.0.0
128 bis 191	10xxxxxx.xxxxxxxx...	B	255.255.0.0
192 bis 223	110xxxxx.xxxxxxxx...	C	255.255.255.0

Hinweis

Wertebereich für erste Dezimalstelle

Für die erste Dezimalzahl der IP-Adresse ist auch ein Wert zwischen 224 und 255 möglich (Adressklasse D etc). Dies ist jedoch nicht empfehlenswert, da für diese Werte keine Adressprüfung erfolgt.

Weitere Subnetze maskieren

Über die Subnetzmaske können Sie ein Subnetz, das einer der Adressklassen A, B oder C zugeordnet ist, weiter strukturieren und "private" Subnetze bilden, indem Sie weitere niederwertige Stellen der Subnetzmaske auf "1" setzen. Pro jedem auf "1" gesetzten Bit verdoppelt sich die Anzahl der "privaten" Netze und halbiert sich die Anzahl der darin enthaltenen Teilnehmer. Nach außen wirkt das Netzwerk nach wie vor wie ein einzelnes Netzwerk.

Beispiel:

Sie ändern bei einem Subnetz der Adressklasse B (z. B. IP-Adresse 129.80.xxx.xxx) die Default-Subnetzmaske wie folgt:

Masken	Dezimal	Binär
Default-Subnetzmaske	255.255.0.0	11111111.11111111.00000000.00000000
Subnetzmaske	255.255.128.0	11111111.11111111.10000000.00000000

Ergebnis:

Alle Teilnehmer mit Adressen von 129.80.001.xxx bis 129.80.127.xxx befinden sich auf einem IP-Subnetz, alle Teilnehmer mit Adressen von 129.80.128.xxx bis 129.80.255.xxx auf einem anderen IP-Subnetz.

Netzübergang (Router)

Die Netzübergänge (Router) haben die Aufgabe, die IP-Subnetze zu verbinden. Wenn ein IP-Datagramm an ein anderes Netzwerk geschickt werden soll, dann muss es zunächst an einen Router vermittelt werden. Damit das möglich ist, müssen Sie für jeden Teilnehmer des IP-Subnetzes die Adresse des Routers eingeben.

Die IP-Adresse eines Teilnehmers im Subnetz und die IP-Adresse des Netzübergangs (Router) dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

Siehe auch

Einstellungen für das Verschalten von Ethernet-Teilnehmern (Seite 94)

1.3.1.11 Einstellungen für das Verschalten von Ethernet-Teilnehmern

Im Folgenden geht es um das Verhalten von STEP 7 beim Verschalten von PROFINET-Geräten und die Auswirkungen der Portverschaltung auf die Netzsicht.

Zusammenhang von Portverschaltung und Netzsicht

Topologiesicht

In der Topologiesicht legen Sie die physikalische Verschaltung von Ethernet-Ports fest. Sie bestimmen konkret, welcher Ethernet-Port eines Geräts mit einem bestimmten Ethernet-Port eines anderen Geräts über eine Ethernet-Leitung verbunden werden soll (Solltopologie).

Beispiel:

Sie legen fest, dass der Port 1 der PROFINET-Schnittstelle der CPU mit dem Port 2 der PROFINET-Schnittstelle von Device A über eine Ethernet-Leitung verbunden sein soll. Bei Geräten mit mehreren PROFINET-Schnittstellen bestimmen Sie darüber hinaus auch die Schnittstelle.

Beispiel:

Sie legen fest, dass der Port 1 der PROFINET-Schnittstelle X2 der CPU mit dem Port 2 der PROFINET-Schnittstelle von Device A über eine Ethernet-Leitung verbunden sein soll. Die Verschaltung der Ethernet-Ports ist tabellarisch oder grafisch möglich.

Netzsicht

In der Netzsicht legen Sie fest, welche Geräte über ein Ethernet-Subnetz miteinander verbunden sein sollen. Sie bestimmen nicht, über welche Ethernet-Ports die Geräte miteinander verschaltet sein sollen (das ist die Aufgabe der Portverschaltung).

Die Portverschaltung hat Auswirkungen auf die Netzsicht: Wenn Sie in der Topologiesicht Ethernet-Ports von Geräten miteinander verschalten, dann verbindet STEP 7 die verschalteten PROFINET-Schnittstellen der Geräte in der Netzsicht mit einem Ethernet-Subnetz (grüne Linie). Der Verlauf der grünen Linie zeigt aber nicht die tatsächliche Leitungsführung. Die konkrete Verkabelung legen Sie in der Topologiesicht fest.

Ein Ethernet-Subnetz besitzt einen Namen und eine S7-Subnetz-ID. Die beiden Werte lassen sich in den Eigenschaften des Subnetzes einstellen.

Mit welchem Ethernet-Subnetz werden verschaltete Geräte vernetzt?

STEP 7 unterscheidet die folgenden Fälle:

- Ethernet-Subnetz vorgegeben (Default-Subnetz).
- Kein Ethernet-Subnetz vorgegeben (kein Default-Subnetz).

Ethernet-Subnetz vorgegeben (Default-Subnetz)

In den Eigenschaften eines Ethernet-Subnetzes ist die Option "Nicht vernetzte Geräte bei Portverschaltung mit diesem Subnetz verbinden" aktiviert (Voreinstellung). Diese Option kann bei genau einem Ethernet-Subnetz aktiviert sein. Aktivieren Sie die Option, wenn erforderlich, bei dem Ethernet-Subnetz, das beim Verschalten nicht vernetzter Teilnehmer fortgeführt werden soll. Dieses Subnetz wird im Folgenden als "Default-Subnetz" bezeichnet.

Reaktion von STEP 7:

Wenn Sie die Ports zweier nicht vernetzter Geräte in der Topologiesicht verschalten, dann vernetzt STEP 7 diese Geräte mit dem Default-Subnetz.

Beispiel:

Schritt 1: Subnetz in der Netzsicht an PLC_4 anlegen, Option "Nicht vernetzte Geräte bei Portverschaltung mit diesem Subnetz verbinden" ist **aktiviert**.

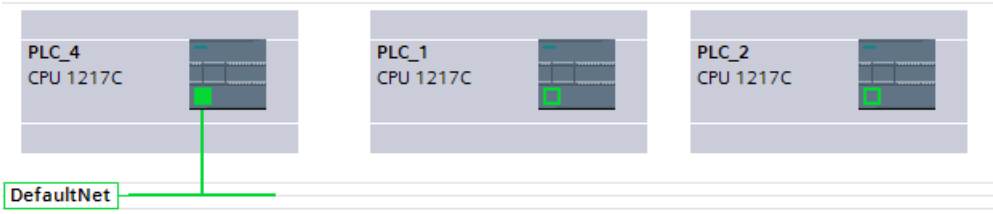

Schritt 2: PLC_1 mit PLC_2 verschalten (Topologiesicht).

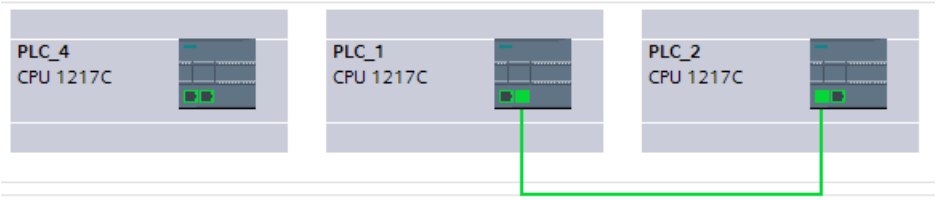

Ergebnis: Alle PLCs sind mit dem selben Default-Subnetz verbunden (Netzsicht).

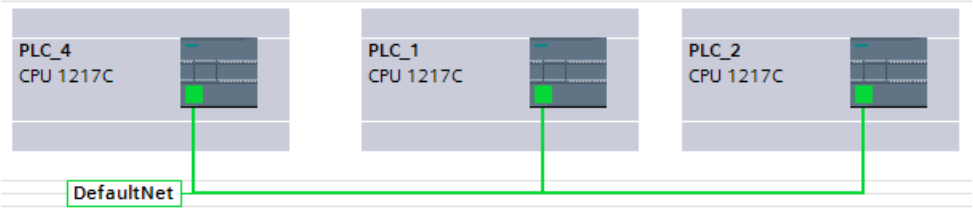

Kein Ethernet-Subnetz vorgegeben (kein Default-Subnetz).

Dieser Fall liegt vor, wenn die folgenden Bedingungen erfüllt sind:

- Einer PROFINET-Schnittstelle ist ein Subnetz hinzugefügt (Diese Schnittstelle wird im Folgendem als "Schnittstelle A" bezeichnet).
- In den Eigenschaften dieses Subnetzes ist die Option "Nicht vernetzte Geräte bei Portverschaltung mit diesem Subnetz verbinden“ deaktiviert (kein Default-Subnetz).
- Es gibt auch kein anderes Ethernet-Subnetz, bei dem die Option aktiviert ist.

Reaktion von STEP 7:

- Das Ethernet-Subnetz der Schnittstelle A wird nur dann fortgeführt, wenn Sie einen Port der Schnittstelle A mit einem Port eines anderen Geräts verschalten.
- Wenn Sie Ports anderer nicht vernetzter Teilnehmer miteinander verschalten, dann erzeugt STEP 7 ein neues Ethernet-Subnetz.

Beispiel:

Schritt 1: Subnetz in der Netzsicht an PLC_4 anlegen, Option "Nicht vernetzte Geräte bei Portverschaltung mit diesem Subnetz verbinden" ist **deaktiviert**.

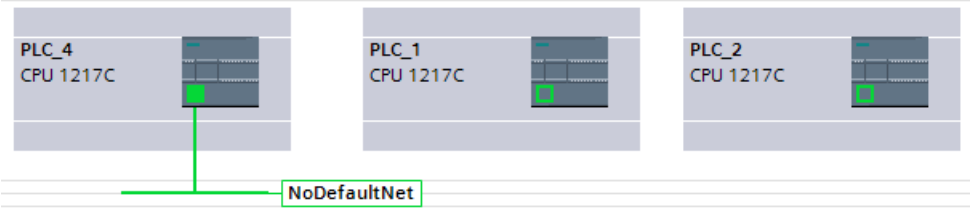

Schritt 2: PLC_1 mit PLC_2 verschalten (Topologiesicht).

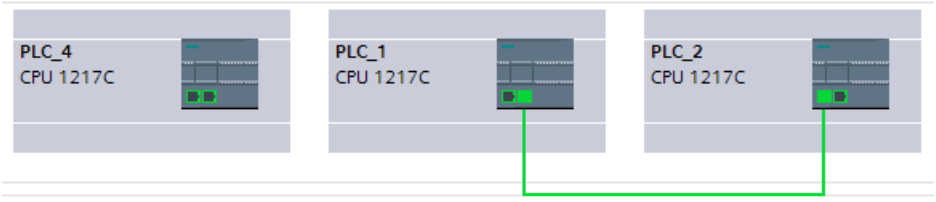

Ergebnis: PLC_1 und PLC_2 sind mit einem neuen Subnetz verbunden (Netzsicht).

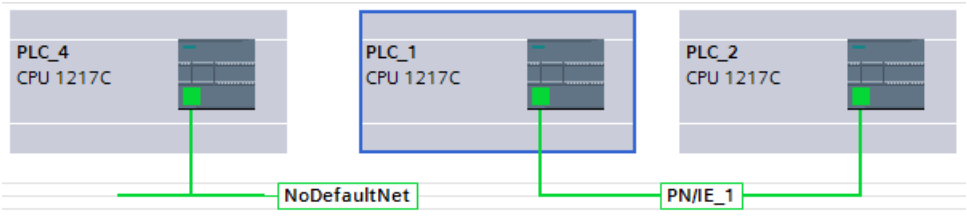

Manuelle Anpassung der IP-Adressen

Möglicherweise passt STEP 7 die IP-Adressen der miteinander verschalteten Geräte nicht so an, dass die resultierende Vernetzung übersetzt werden kann, z. B. bei Geräten mit mehreren PROFINET-Schnittstellen. In diesen Fällen ist es erforderlich, dass Sie die IP-Adressen der Geräte manuell anpassen.

Folgende Regeln gelten:

- Geräte, die ohne Router miteinander kommunizieren sollen, dürfen nicht unterschiedlichen IP-Subnetzen angehören.
- Bei Geräten mit mehreren PROFINET-Schnittstellen müssen die Schnittstellen in unterschiedlichen IP-Subnetzen liegen.

Um die IP-Adresse einer PROFINET-Schnittstelle zu ändern, führen Sie die folgenden Schritte aus:

1. Wechseln Sie in die Netzsicht (falls nicht schon ausgewählt).
2. Klicken Sie mit der linken Maustaste auf das Symbol der PROFINET-Schnittstelle, die nicht dem IP-Subnetz angehören soll.
3. In den Eigenschaften der PROFINET-Schnittstelle ändern Sie den Subnetz-Anteil der IP-Adresse (im Bereich "Ethernet-Adressen").

Beispiel: Manuelle Umstellung des IP-Subnetzanteils bei Geräten mit mehreren PROFINET-Schnittstellen an einem Subnetz

Die IP-Adresse lautet: "192.168.0.1".

Die Subnetzmaske ist "255.255.255.0".

Dann bilden die ersten drei Zahlen "192.168.0" den IP-Subnetz-Anteil der IP-Adresse "192.168.0.1".

Ändern Sie den IP-Subnetz-Anteil ab, z.B. in "192.168.1"

Siehe auch

Ports verschalten in der grafischen Ansicht (Seite 315)

Ports verschalten in der tabellarischen Ansicht (Seite 317)

Netzkonfiguration Industrial Ethernet (Seite 92)

1.3.2 Kommunikation über Verbindungen

1.3.2.1 Arbeiten mit Verbindungen

Open User Communication anwenden

Grundlagen der Open User Communication

Einführung

Open User Communication (OUC) ist eine Bezeichnung für ein programmgesteuertes Kommunikationsverfahren zur Kommunikation über die integrierte PN/IE-Schnittstelle der CPUs der S7-1200/1500 und der S7-300/400. Für dieses Kommunikationsverfahren stehen unterschiedliche Verbindungstypen zur Verfügung.

Die Open User Communication zeichnet sich durch eine hohe Flexibilität hinsichtlich der zu übertragenden Datenstrukturen aus und ermöglicht damit den offenen Datenaustausch mit beliebigen Kommunikationsteilnehmern, sofern diese die hier zur Verfügung gestellten Verbindungstypen unterstützen. Da die Kommunikation ausschließlich über Anweisungen im Anwenderprogramm gesteuert wird, ist ein ereignisgesteuertes Auf- und Abbauen von Verbindungen möglich. Verbindungen können außerdem zur Laufzeit durch das Anwenderprogramm modifiziert werden.

Für CPUs mit einer integrierten PN/IE-Schnittstelle stehen Ihnen für die Open User Communication die Verbindungstypen TCP, UDP und ISO-on-TCP zur Verfügung. Die Kommunikationspartner können zwei SIMATIC-PLCs oder auch eine SIMATIC-PLC und ein geeignetes Fremdgerät sein.

Anweisungen für die Open User Communication

Zum Erstellen der Verbindungen stehen Ihnen nach dem Öffnen im Programmiereditor in der Task Card "Anweisungen > Kommunikation > Open User Communication" je nach CPU-Typ verschiedene Anweisungen zur Verfügung:

- Kompakte Anweisungen zum Senden oder Empfangen von Daten über mit den integrierten Funktionen zum Auf- oder Abbau der Verbindung (nur S7-1200/1500):
  - TSEND_C (Verbindungsauf-/abbau, Senden)
  - TRCV_C (Verbindungsauf-/abbau, Empfangen)
- Einzelanweisungen zum Senden und Empfangen von Daten oder für den Auf- oder Abbau der Verbindungen:
  - TCON (Verbindungsaufbau)
  - TDISCON (Verbindungsabbau)
  - TSEND (TCP oder ISO-on-TCP: Senden)
  - TRCV (TCP oder ISO-on-TCP: Empfangen)
  - TUSEND (UDP: Senden)
  - TURCV (UDP: Empfangen)

Verbindungsaufbau

Für die Open User Communication müssen für beide Kommunikationspartner Anweisungen zum Auf- und Abbau der Verbindung vorhanden sein. Ein Kommunikationspartner sendet seine Daten über TSEND, TUSEND oder TSEND_C, während der andere Kommunikationspartner die Daten über TRCV, TURCV oder TRCV_C empfängt.

Einer der Kommunikationspartner beginnt als aktiver Partner mit dem Verbindungsaufbau. Der andere Kommunikationspartner reagiert darauf und startet als passiver Partner seinen Verbindungsaufbau. Wenn beide Kommunikationspartner ihren Verbindungsaufbau angestoßen haben, wird die Kommunikationsverbindung vollständig aufgebaut.

Verbindungsparametrierung

Sie können den Verbindungsaufbau über einen Verbindungsbeschreibungs-DB mit der Struktur TCON_Param, TCON_IP_v4 oder TCON_IP_RFC folgendermaßen parametrieren:

- Manuell erstellen, parametrieren und direkt an die Anweisung schreiben.
- Unterstützt über die Verbindungsparametrierung.

Die Verbindungsparametrierung unterstützt den Aufbau der Verbindung und ist daher gegenüber der anderen Methoden zu empfehlen.

In der Verbindungsparametrierung treffen Sie folgende Festlegungen:

- Verbindungspartner
- Verbindungstyp
- Verbindungs-ID

- Verbindungsbeschreibungs-DB
- Adressdetails in Abhängigkeit vom gewählten Verbindungstyp

Weiterhin legen Sie hier fest, welcher Kommunikationspartner den Verbindungsaufbau aktiviert und welcher auf eine Anforderung des Kommunikationspartners hin einen passiven Verbindungsaufbau durchführt.

Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 112)

Verbindungsparametrierung

Übersicht über die Verbindungsparametrierung

Einführung

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors, wenn Sie mit den Kommunikationsanweisungen TSEND_C, TRCV_C oder TCON eine Open User Communication programmieren möchten.

Die Verbindungsparametrierung unterstützt die flexible Funktionalität der Kommunikations-Programmierung: In einem automatisch erzeugten von der Struktur des Typs TCON_Param, TCON_IP_v4 oder TCON_IP_RFC abgeleiteten Global-DB werden die bei der Verbindungsparametrierung eingegebenen Parameter gespeichert. In diesem Verbindungsbeschreibungs-DB können Sie die Verbindungsparameter modifizieren.

Aufbau der Verbindungsparametrierung

Die Verbindungsparametrierung besteht aus folgenden Komponenten:

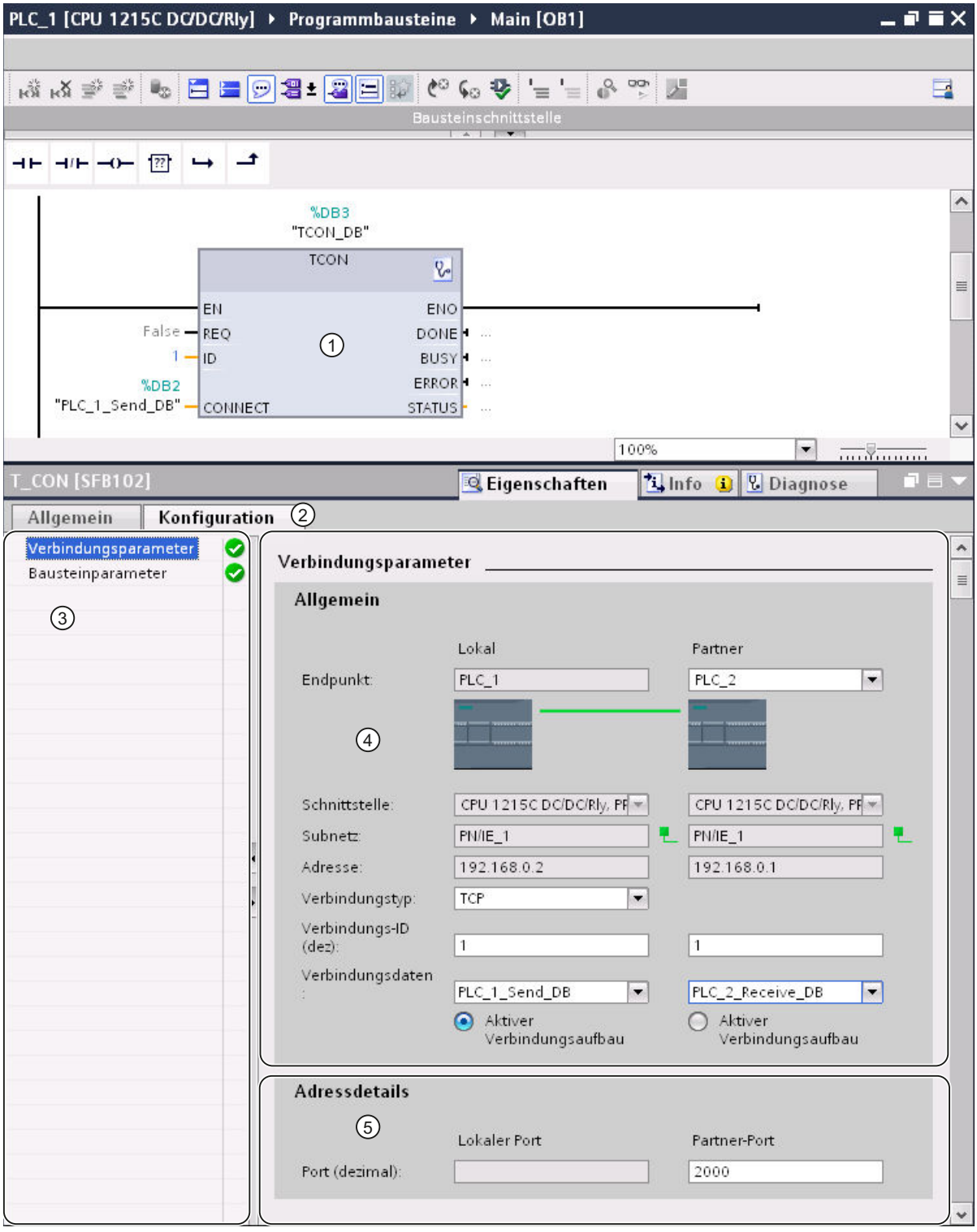

① Kommunikationsanweisung für TCON, TSEND_C oder TRCV_C

② Register "Konfiguration" im Register "Eigenschaften"
③ Bereichsnavigation des Registers "Konfiguration"
④ Allgemeine Eigenschaften der Verbindungsparameter
⑤ Adressdetails der Verbindungsparameter (bei ausgewählten Verbindungs-DBs)

Register "Konfiguration"

Tragen Sie die gewünschten Verbindungsparameter im Register "Konfiguration" ein. Sie finden in der Bereichsnavigation des Registers "Konfiguration" die Gruppe "Verbindungsparameter". Diese Gruppe enthält die Verbindungsparametrierung. Hier können Sie systemunterstützt die Parameter für die Verbindungen und die Adressdetails eintragen. Hier beschalten Sie auch die Bausteinparameter CONNECT (TCON, TSEND_C, TRCV_C) oder ID (TCON, TSEND, TRCV, TUSEND, TURCV) der gewählten Kommunikationsanweisungen.

Wenn alle erforderlichen Parameter vorhanden sind, wird in der Bereichsnavigation vor die Gruppe "Verbindungsparameter" ein Häkchen gesetzt.

Hinweis

Die Verbindungsparametrierung prüft nicht auf Eindeutigkeit von Verbindungs-IDs und Port-Nummern (TCP, UDP) bzw. TSAPs (ISO-on-TCP, ISO). Sorgen Sie daher bei der Konfiguration der Open User Communication für eine eindeutige Vergabe dieser Parameter innerhalb eines Gerätes.

Siehe auch

Verbindungsparameter nach TCON_Param (Seite 115)

Verbindungsparameter nach TCON_IP_v4 (Seite 117)

Verbindungsparameter nach TCON_IP_RFC (Seite 123)

Beschreibung der Verbindungsparameter

Überblick

Die folgende Tabelle zeigt die allgemeinen Verbindungsparameter:

Parameter	Beschreibung
Endpunkt	Die Namen des lokalen Endpunkts und des Partner-Endpunkts wird angezeigt. Der lokale Endpunkt ist die CPU, für die der TCON, TSEND_C oder TRCV_C programmiert wird. Der lokale Endpunkt ist daher immer bekannt. Der Partner-Endpunkt wird über die Klappliste ausgewählt. In der Klappliste werden alle verfügbaren möglichen Verbindungspartner angezeigt, darunter auch unspezifizierte Verbindungspartner für Geräte, deren Daten innerhalb des Projekts nicht bekannt sind. Bei S7-1500 kann als Partner-Endpunkt Broadcast ausgewählt werden (Nachricht an alle Teilnehmer des Subnetzes). Bei S7-1500 CPs/CMs kann als Partner-Endpunkt auch Multicast ausgewählt werden (Nachricht an eine Gruppe innerhalb des Subnetzes). Der Verbindungstyp wird dabei automatisch auf UDP umgestellt. Solange noch kein Verbindungspartner ausgewählt ist, sind bis auf den Parameter "Verbindungsdaten alle weiteren Parameter in der Maske deaktiviert.
Schnittstelle	Die Schnittstelle des lokalen Endpunkts wird angezeigt. Sind mehrere Schnittstellen verfügbar, zum Beispiel auch über CPs oder CMs, kann die Schnittstelle über die Klappliste ausgewählt werden. Die Anzeige oder Auswahl der Partner-Schnittstelle erfolgt erst nach Auswahl eines spezifizierten Partner-Endpunkts.
Subnetz	Das Subnetz des lokalen Endpunkts wird angezeigt, wenn dieses vorhanden ist. Die Anzeige des Partner-Subnetzes erfolgt erst nach Auswahl des Partner-Endpunkts. Wenn mindestens einer der beiden Verbindungspartner nicht mit einem Subnetz verbunden ist, werden die beiden Verbindungspartner miteinander vernetzt. Eine Verbindung von Verbindungspartnern an unterschiedlichen Subnetzen ist nur über IP-Routing möglich. Die IP-Routing-Einstellungen können über die Eigenschaften der betroffenen Ethernet-Schnittstellen bearbeitet werden.
Adresse	Die IP-Adresse oder die PROFIBUS-Adresse des lokalen Endpunkts wird angezeigt, abhängig vom verwendeten Subnetz. Die Anzeige der entsprechenden Adresse des Partners erfolgt erst nach Auswahl des Partner Endpunkts. Wenn Sie einen unspezifizierten Verbindungspartner ausgewählt haben, ist das Eingabefeld leer und rot hinterlegt. Sie müssen in dem Fall eine gültige IP-Adresse/PROFIBUS-Adresse angeben. Der Typ der Adresse (IP oder PROFIBUS) ist vom Typ des Subnetzes abhängig, das am lokalen Partner eingestellt ist. Broadcast (nur S7-1500): Ist als Partner-Endpunkt "Broadcast" eingestellt, wird für den Verbindungspartner automatisch eine nicht änderbare IP-Adresse mit der Host-Adresse 255 eingetragen. Der Netzwerkanteil entspricht dem des Senders. Beispiel: Lokale IP-Adresse 192.168.0.1, Partner-IP-Adresse 192.168.0.255. Multicast (nur S7-1500 CPs/CMs): Ist als Partner-Endpunkt "Multicast" eingestellt, wird für den Verbindungspartner automatisch die änderbare IP-Adresse 224.0.1.0 eingetragen.

Parameter	Beschreibung
Verbindungstyp	Wählen Sie über die Klappliste "Verbindungstyp" den zu verwendenden Verbindungstyp aus: • TCP • ISO-on-TCP • UDP Bei S7-1500 können Sie an TSEND_C und TRCV_C bzw. TCON bei der Konfigurationsart der konfigurierten Verbindungen auch den Verbindungstyp ISO auswählen. Die Verbindungstypen sind nur für Partner verwendbar, die das entsprechende Protokoll auch unterstützen.
Konfigurationsart (nur bei S7-1500)	Bei S7-1500 können an TSEND_C und TRCV_C zwei verschiedene Konfigurationsarten eingestellt werden: • Programmierte Verbindungen nutzen Programmbausteine für die Verbindungsbeschreibung. • Konfigurierte Verbindungen werden für die Konfiguration gespeichert und entstehen erst nach dem Laden in das Gerät zur Laufzeit. Über die konfigurierte Verbindung lässt sich auch der Verbindungstyp ISO auswählen. Die vorgegebene Konfigurationsart hängt von dem ausgewählten Verbindungstyp ab. Wenn beide Konfigurationsarten möglich sind, ist die programmierte Verbindung voreingestellt. Bei beiden Verbindungspartnern muss dieselbe Konfigurationsart eingestellt sein.
Verbindungs-ID	Tragen Sie im Eingabefeld die Verbindungs-ID ein. Sie können die Verbindungs-ID in den Eingabefeldern ändern oder direkt am TCON angeben. Sorgen Sie für eine eindeutige Vergabe der Verbindungs-ID innerhalb des Geräts!
Verbindungsdaten	Die Namen der für die Verbindungsbeschreibung nach TCON_IP_v4, TCON_IP_RFC oder TCON_Param strukturierten Verbindungsbeschreibungs-DB werden in den Klapplisten angezeigt. Sie können über die Klappliste einen neuen Datenbaustein generieren oder einen bereits vorhandenen Datenbaustein auswählen. Dieser wird automatisch mit den Werten aus der Verbindungsparametrierung befüllt. Der Name der gewählten Datenbausteine wird automatisch am Bausteinparameter CONNECT der markierten Anweisungen TSEND_C, TRCV_C oder TCON eingetragen. Sie können über die Klappliste auch einen anderen gültigen Datenbaustein referenzieren. Wenn über den Eingangsparameter CONNECT der erweiterten Anweisungen TSEND_C, TRCV_C oder TCON ein DB referenziert wird, der nicht der Struktur eines TCON_IP_v4, TCON_IP_RFC oder TCON_Param entspricht, wird die Klappliste bei leerem Inhalt rot hinterlegt.
Verbindungsname (nur bei S7-1500)	Wenn bei S7-1500 an TSEND_C und TRCV_C die Konfigurationsart der konfigurierten Verbindungen eingestellt ist, wird der Parameter "Verbindungsdaten durch den Parameter "Verbindungsname" ersetzt. Als Verbindungsdatum dient hier nun der Name der konfigurierten Verbindung. Nachdem ein Verbindungspartner ausgewählt wurde, ist die Klappliste noch leer. Sie können über die Klappliste eine neue Verbindung generieren oder eine bereits vorhandene Verbindung auswählen. Ein Datenbaustein wird ggf. erzeugt und automatisch mit den Werten aus der Verbindungsparametrierung befüllt. Der Name des Datenbausteins wird automatisch am Bausteinparameter CONNECT der Anweisung TSEND_C oder TRCV_C eingetragen. Sie können über die Klappliste auch eine vorhandene Verbindung referenzieren.
Aktiver Verbindungsaufbau	Bestimmen Sie über das Optionsfeld "Aktiver Verbindungsaufbau" den aktiven Partner der Open User Communication (nur bei TCP und ISO-on-TCP).

Parameter	Beschreibung
Port (nur bei TCP und UDP)	Adresskomponente für eine TCP- oder UDP-Verbindung. Der Vorgabewert nach dem Anlegen einer neuen TCP-Verbindung ist 2000. Die Port-Nummern können Sie ändern. Die Port-Nummern müssen im Gerät eindeutig sein!
TSAP (nur bei ISO-on-TCP)	Adresskomponente für eine ISO-on-TCP-Verbindung. Der Vorgabewert nach dem Anlegen einer neuen ISO-on-TCP-Verbindung ist E0.01.49.53.4F.6F.6E.54.43.50.2D.31 (S7-1200/1500) bzw. E0.02.49.53.4F.6F.6E.54.43.50.2D.31 (S7-300/400). Sie können die TSAP-ID mit Erweiterung oder als ASCII-TSAP eingeben. Die TSAPs müssen im Gerät eindeutig sein!

Hinweis

UDP-Verbindung für die Einstellung "Broadcast" (S7-300/400/1200)

Die Parameter der UDP-Verbindung für die Einstellung "Broadcast" am Partner-Endpunkt werden in einem Verbindungsbeschreibungs-DB TCON_IP_v4 abgelegt: Bezüglich der UDP-Kommunikation mit TCON und TUSEND/TURCV wird der TCON_IP_v4 mit keinen Partner-Parametern befüllt (Wert=0). Die Partneradresse und der Partnerport sind aber für das Senden der Daten notwendig und müssen vom Anwender in den TADDR_Param eingetragen werden. Der TADDR_Param für die UDP-Kommunikation wird von dem TUSEND-/TURCV-Bausteinparameter ADDR referenziert. Die Werte für beide Parameter können der Verbindungsparametrierung entnommen werden.

Auch für die anderen Empfänger der UDP-Kommunikation muss die Konfiguration angepasst werden: Um die Broadcast-Telegramme zu empfangen, muss an der Empfängerseite der Partnerport konfiguriert werden. Hierzu ist am ADDR-Baustein der Parameter RemotePort des TADDR_Param zu befüllen.

Hinweis

Kommunikation über TSEND_C und TRCV_C (S7-1500)

Bei der Verwendung von TSEND_C und TRCV_C wird für jede Kommunikation ein eigenes Bausteinpaar TSEND_C und TRCV_C mit einer konfigurierten Verbindung benötigt. Mehrere Bausteinpaare TSEND_C und TRCV_C können nicht gleichzeitig dieselbe konfigurierte Verbindung für ihre Kommunikation benutzen.

Weitere Verbindungen für eine Anweisung TSEND_C oder TRCV_C können im Inspektorfenster bei den Verbindungsparametern über die entsprechende Schaltfläche neben den Verbindungsdaten angelegt werden.

Die über TSEND_C und TRCV_C konfigurierten Verbindungen werden bei angewähltem Baustein TSEND_C oder TRCV_C im Inspektorfenster unter "Eigenschaften > Konfiguration > Übersicht über die konfigurierten Verbindungen" in einer Verbindungstabelle angezeigt.

Siehe auch

Vergabe von Port-Nummern (Seite 129)

TSAP-Struktur (Seite 132)

Beispiele zur TSAP-Vergabe (Seite 135)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 131)

Verbindungen anlegen und parametrieren (Seite 107)

Verbindungsparameter nach TCON_Param (Seite 115)

Verbindungsparameter nach TCON_IP_v4 (Seite 117)

Verbindungsparameter nach TCON_IP_RFC (Seite 123)

Verbindungsparametrierung starten

Die Verbindungsparametrierung zur Open User Communication wird aktiviert, sobald Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen.

Voraussetzung

- Ihr Projekt beinhaltet mindestens eine S7-CPU.
- Der Programmiereditor ist geöffnet.
- Ein Netzwerk ist vorhanden.

Vorgehen

Um die erweiterten Anweisungen für die Open User Communication einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card, Palette und Ordner "Anweisungen > Kommunikation > Open User Communication".
2. Ziehen Sie eine der folgenden Anweisungen via Drag & Drop in ein Netzwerk:
   - TSEND_C
   - TRCV_C
   - TCON

   Der Dialog "Aufrufoptionen" wird geöffnet.
3. Bearbeiten Sie im Dialog "Aufrufoptionen" die Eigenschaften des Instanz-DBs. Sie haben dabei folgende Möglichkeiten:
   - Ändern Sie den vorgegebenen Namen.
   - Wählen Sie das Optionsfeld "manuell", um eine eigene Nummer zu vergeben.
   - Für Funktionsbausteine können Sie den DB auch als Multiinstanz ausführen.
4. Klicken Sie auf "OK", um die Eingabe abzuschließen.

Ergebnis

Für die eingefügte Anweisung TSEND_C, TRCV_C oder TCON wird bei einer Einzelinstanz ein entsprechender Instanz-DB angelegt. Im Falle einer Multiinstanz wird der Instanz-DB des Funktionsbausteins verwendet.

Bei markiertem TSEND_C, TRCV_C oder TCON sehen Sie im Inspektorfenster unter "Eigenschaften" das Register "Konfiguration". Die Gruppe "Verbindungsparameter" in der Bereichsnavigation enthält die Verbindungsparametrierung, die Sie nun durchführen können.

Siehe auch

Verbindungen anlegen und parametrieren (Seite 107)

Verbindungen anlegen und parametrieren

In der Verbindungsparametrierung der Open User Communication können Sie Verbindungen vom Typ TCP, UDP und ISO-on-TCP anlegen und parametrieren.

Voraussetzung

Es existiert eine CPU mit einer Kommunikationsanweisung TCON, TSEND_C oder TRCV_C.

Vorgehen

Um eine Verbindung der Open User Communication anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Programmiereditor einen Baustein der Open User Communication TCON, TSEND_C oder TRCV_C.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, sind nur die Klapplisten für den Partner-Endpunkt und die lokalen Verbindungsdaten aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.
   Es werden die bereits bekannten Verbindungsparameter angezeigt:
   - Name des lokalen Endpunkts
   - Schnittstelle des lokalen Endpunkts
   - IP-Adresse (bei Ethernet-Subnetz) oder PROFIBUS-Adresse (bei PROFIBUS-Subnetz) des lokalen Endpunkts

4. Wenn Sie einen vorhandenen Verbindungsbeschreibungs-DBs verwenden möchten, gehen Sie weiter zu Schritt 6. Wenn Sie die Verbindung mit einem neuen entfernten Verbindungspartner konfigurieren möchten, fahren Sie hier fort:
   Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage. Bestimmte Verbindungsparameter werden danach als Vorgabe automatisch eingetragen.
   Die folgenden Parameter werden eingestellt:
   - Name des Partner-Endpunkts
   - Schnittstelle des Partner-Endpunkts
   - IP-Adresse (bei Ethernet-Subnetz) oder PROFIBUS-Adresse (bei PROFIBUS-Subnetz) des Partner-Endpunkts

   Wenn die Verbindungspartner vernetzt sind, wird der Name des Subnetzes angezeigt.
5. Bei S7-1500 wählen Sie in der Klappliste "Konfigurationsart" zwischen der Verwendung von Programmbausteinen oder konfigurierten Verbindungen.

6. Wählen Sie in der Klappliste "Verbindungsdaten" einen vorhandene Verbindungsbeschreibungs-DBs oder bei konfigurierten Verbindungen unter "Verbindungsname" eine vorhandene Verbindung. Sie können auch einen neue Verbindungsbeschreibungs-DBs oder eine neue konfigurierte Verbindung anlegen. Wenn Sie einen vorhandenen Verbindungsbeschreibungs-DBs auswählen, werden verfügbare Konfigurationsdaten übernommen und in die Verbindungsbeschreibung eingetragen. Eventuell vorher eingetragenen Daten eines entfernten Partners werden dabei überschrieben. Sie können später noch andere konfigurierte Verbindungen wählen oder die Namen der Verbindungsbeschreibungs-DBs ändern, um neue Datenbausteine zu erstellen:

   – Den ausgewählten Datenbaustein sehen Sie auch an der Beschaltung des Eingangsparameters CONNECT der ausgewählten Anweisung TCON, TSEND_C oder TRCV_C.

   – Wenn Sie für den Verbindungspartner bereits einen Verbindungsbeschreibungs-DB über den Parameter CONNECT der Anweisung TCON, TSEND_C oder TRCV_C angegeben haben, können Sie entweder diesen DB verwenden oder einen neuen DB anlegen.

   – Wenn Sie den Namen des angezeigten Datenbausteins in der Klappliste bearbeiten, wird automatisch ein neuer Datenbaustein mit dem geänderten Namen, aber derselben Struktur und demselben Inhalt generiert und für die Verbindung verwendet.

   – Geänderte Namen eines Datenbausteins müssen im Kontext des Kommunikationspartners eindeutig sein.

   – Ein Verbindungsbeschreibungs-DB muss je nach CPU-Typ und Verbindung die Struktur TCON_Param, TCON_IP_v4 oder TCON_IP_RFC haben.

   – Ein Datenbaustein kann nicht für einen unspezifizierten Partner ausgewählt werden.

   Nach Auswahl oder Anlegen des Verbindungsbeschreibungs-DBs oder der konfigurierten Verbindung werden weitere Werte ermittelt und eingetragen.
   Für spezifizierte Verbindungspartner gilt:

   – Verbindungstyp ISO-on-TCP

   – Verbindungs-ID mit dem Vorgabewert 1

   – Aktiver Verbindungaufbau vom lokalen Partner

   – TSAP-ID
     für S7-1200/1500: E0.01.49.53.4F.6F.6E.54.43.50.2D.31
     für S7-300/400: E0.02.49.53.4F.6F.6E.54.43.50.2D.31

   Für unspezifizierte Verbindungspartner gilt:

   – Verbindungstyp TCP

   – Partnerport 2000

   Bei konfigurierter Verbindung mit spezifiziertem Verbindungspartner gilt:

   – Verbindungstyp TCP

   – Verbindungs-ID mit dem Vorgabewert 257

   – Aktiver Verbindungaufbau vom lokalen Partner

   – Partnerport 2000

Bei konfigurierter Verbindung mit unspezifiziertem Verbindungspartner gilt:

- Verbindungstyp TCP
- Lokaler Port 2000

7. Geben Sie ggf. eine Verbindungs-ID für den Verbindungspartner an. Für einen unspezifizierten Partner kann keine Verbindungs-ID vergeben werden.

Hinweis

Sie müssen bei einem bekannten Verbindungspartner einen eindeutigen Wert für die Verbindungs-ID eingeben. Die Eindeutigkeit der Verbindungs-ID wird nicht durch die Verbindungsparametrierung geprüft und es wird bei Anlegen einer neuen Verbindung kein Vorgabewert für die Verbindungs-ID eingetragen!

8. Wählen Sie den gewünschten Verbindungstyp aus der entsprechenden Klappliste. Die Adressdetails werden abhängig vom Verbindungstyp mit Werten vorbelegt. Sie haben die Wahl zwischen:
   - TCP
   - ISO-on-TCP
   - UDP

   Für konfigurierte Verbindungen bei S7-1500 gilt zusätzlich ISO.
9. Sie können die Eingabefelder in den Adressdetails bearbeiten. Je nach eingestelltem Protokoll können Sie die Ports (für TCP und UDP) oder die TSAPs (für ISO-on-TCP und ISO) bearbeiten.
10. Stellen Sie bei TCP, ISO und ISO-on-TCP das Verhalten für den Verbindungsaufbau über die Optionsfelder "Aktiver Verbindungsaufbau" ein. Sie können auswählen, welcher Kommunikationspartner die Verbindung aktiv aufbauen soll.

Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft und in den Datenbaustein für die Verbindungsbeschreibung eingetragen.

Hinweis

Die Open User Communication zwischen zwei Kommunikationspartnern ist erst dann lauffähig, wenn auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurde. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

Siehe auch

Beschreibung der Verbindungsparameter (Seite 103)

Verbindungsparametrierung starten (Seite 106)

TSAP-Struktur (Seite 132)

Vergabe von Port-Nummern (Seite 129)

Verbindungsparameter nach TCON_Param (Seite 115)

Verbindungsparameter nach TCON_IP_v4 (Seite 117)

Verbindungsparameter nach TCON_IP_RFC (Seite 123)

Verbindungen löschen

Einführung

Die Daten einer angelegten Verbindung der Open User Communication werden in einem Verbindungsbeschreibungs-DB abgelegt. Sie können die Verbindung löschen, indem Sie den Datenbaustein mit der enthaltenen Verbindungsbeschreibung löschen.

Voraussetzung

Sie haben eine Verbindung der Open User Communication erstellt.

Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Kommunikationspartner der Open User Communication in der Projektnavigation.
2. Öffnen Sie den Ordner "Programmbausteine > Systembausteine > Programmressourcen" unterhalb des selektierten Kommunikationspartners.
3. Wählen Sie im Kontextmenü des Datenbausteines mit der Verbindungsparametrierung den Befehl "Löschen".

Hinweis

Wenn Sie nicht genau wissen, welchen Baustein Sie löschen müssen, öffnen Sie die erweiterte Anweisung TCON, TSEND_C oder TRCV_C. Sie finden den Namen des Datenbausteins als Eingangsparameter CONNECT oder in der Verbindungsparametrierung als Parameter "Verbindungsdaten".

Wenn Sie nur die Instanz-DBs der erweiterten Anweisungen TCON, TSEND_C oder TRCV_C löschen, werden die parametrierten Verbindungen nicht mitgelöscht.

Hinweis

Wenn der Verbindungs-DB noch von weiteren Bausteinen der erweiterten Anweisungen verwendet wird, müssen die entsprechenden Aufrufe, deren Instanz-DBs und wenn vorhanden, die Kombinationsbausteine TSEND_C und TRCV_C ebenfalls aus dem Bausteinordner gelöscht werden, sofern sie nicht anderweitig verwendet werden.

Durch diese Maßnahme wird verhindert, dass das Programm inkonsistent wird.

Ergebnis

Sie haben die Verbindung gelöscht.

Hinweis

Fügen Sie eine erweiterte Anweisung TCON, TSEND_C oder TRCV_C erneut ein, um eine existierende Verbindungsbeschreibungen mit der Struktur TCON_Param, TCON_IP_v4 oder TCON_IP_RFC über den Parameter "Verbindungsdaten" erneut referenzieren.

Verbindungsparameter

Arbeitsweise verbindungsorientierter Protokolle

Einführung

Verbindungsorientierte Protokolle bauen vor der Datenübertragung eine logische Verbindung zum Kommunikationspartner auf und bauen diese nach Abschluss der Datenübertragung ggf. wieder ab. Verbindungsorientierte Protokolle werden insbesondere dann eingesetzt, wenn es auf eine zuverlässige Datenübertragung ankommt. Über eine physikalische Leitung können mehrere logische Verbindungen bestehen.

Bei der Open User Communication werden die folgenden Verbindungstypen unterstützt:

- TCP
- ISO-on-TCP
- ISO (nur S7-1500)
- UDP

Für eine Verbindung müssen beide Kommunikationspartner denselben Verbindungstyp unterstützen. Wird beispielsweise eine Verbindung vom Typ ISO-on-TCP von einem Kommunikationspartner nicht unterstützt, verwenden Sie stattdessen vielleicht den Verbindungstyp TCP - sofern dieser auch unterstützt wird.

Für Kommunikationspartner, die nicht im TIA Portal konfiguriert werden können, beispielsweise bei Fremdgeräten oder PCs, tragen Sie in der Verbindungsparametrierung als Partner-Endpunkt "unspezifiziert" ein. Der benötigte Verbindungstyp für unspezifizierte Geräte ist den jeweiligen Dokumentationen zu entnehmen.

Hinweis

Verbindungen mit ISO

Bei S7-1500 CPUs können über die Anweisungen TSEND_C und TRCV_C konfigurierte Verbindungen vom Typ ISO erstellt werden. Beachten Sie für weitere Informationen zu diesen Verbindungstypen die allgemeinen Verbindungsbeschreibungen.

Merkmale von TCP

TCP ist ein Streaming-Protokoll, bei welchem die Länge des Datenstroms an den Empfänger übertragen wird, damit dieser den Datenstrom als einzelne TCP-Segmente empfangen kann. Bei der Datenübertragung über eine TCP-Verbindung werden also keine Informationen über Anfang und Ende einer Nachricht übertragen. Der Empfänger kann über die empfangenen Segmente des Datenstroms nicht erkennen, wo eine Nachricht im Datenstrom endet und wo die nächste beginnt. Es wird daher empfohlen, der Anzahl der zu empfangenden Bytes (Parameter LEN, Anweisung TRCV/TRCV_C) denselben Wert zuzuweisen wie der Anzahl der zu sendenden Bytes (Parameter LEN, Anweisung TSEND/TSEND_C).

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND_C): TRCV/TRCV_C kopiert die empfangenen Daten erst dann in den vorgegebenen Empfangsbereich (Parameter DATA), wenn die parametrierte Länge erreicht ist. Wenn die parametrierte Länge erreicht wird, werden bereits Daten eines nachfolgenden Auftrags empfangen. Im Empfangsbereich befinden sich dann Daten aus zwei unterschiedlichen Sendeaufträgen. Wenn Sie die exakte Länge der ersten Nachricht nicht wissen, haben Sie keine Möglichkeit, das Ende der ersten bzw. den Anfang der zweiten Nachricht zu erkennen.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND_C): TRCV/TRCV_C kopiert so viele Bytes in den Empfangsdatenbereich (Parameter DATA), wie Sie am Parameter LEN vorgegeben haben. Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD_LEN (Anzahl der tatsächlich empfangenden Daten) den Wert von LEN zu. Mit jedem weiteren Aufruf erhalten Sie einen weiteren Block der gesendeten Daten.

Über die Protokollvariante des Ad-hoc-Modus kann in den Anweisungen TRCV/TRCV_C ein Empfangsbereich mit fester Datenlänge festgelegt werden.

Merkmale von ISO-on-TCP

ISO-on-TCP ist ein Message-orientiertes Protokoll, welches auf Empfängerseite das Ende der Nachricht erkennt und die zur Nachricht gehörenden Daten an den Anwender indiziert. Dies ist von der spezifizierten Empfangslänge der Nachricht unabhängig. Bei der Datenübertragung über eine ISO-on-TCP-Verbindung werden also Informationen zur Länge und zum Ende einer Nachricht mitgeliefert.

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND_C): TRCV/TRCV_C kopiert die gesendeten Daten vollständig in den Empfangsdatenbereich (Parameter DATA). Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD_LEN (Anzahl der tatsächlich empfangenden Daten) die Länge der gesendeten Daten zu.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND_C): TRCV/TRCV_C kopiert keine Daten in den Empfangsdatenbereich (Parameter DATA), sondern liefert folgende Fehlerinformation: ERROR=1, STATUS=W#16#8088 (Zielpuffer zu klein).

Merkmale von UDP

UDP ist ein Message-orientiertes Protokoll, welches auf Empfängerseite das Ende der Nachricht erkennt und die zur Nachricht gehörenden Daten an den Anwender indiziert. Dies ist von der spezifizierten Empfangslänge der Nachricht unabhängig. Bei der Datenübertragung über eine UDP-Verbindung werden also Informationen zur Länge und zum Ende einer Nachricht mitgeliefert.

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TURCV/TRCV_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TUSEND/TSEND_C): TURCV/TRCV_C kopiert die gesendeten Daten vollständig in den Empfangsdatenbereich (Parameter DATA). Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD_LEN (Anzahl der tatsächlich empfangenden Daten) die Länge der gesendeten Daten zu.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TURCV/TRCV_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TUSEND/TSEND_C): TURCV/TRCV_C kopiert soviele Daten in den Empfangsdatenbereich (Parameter DATA), wie im Parameter LEN angefordert. Es wird keine weitere Fehlermeldung generiert. Der Anwender muss in diesem Fall nochmals einen T_URCV aufrufen um die restlichen Bytes zu empfangen.

Siehe auch

Grundlagen der Open User Communication (Seite 98)

Verbindungsparameter nach TCON_Param

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP, UDP und ISO-on-TCP zu parametrieren, wird für einige CPUs der S7-1200 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_Param verwendet. Die feste Datenstruktur des TCON_Param enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei Verwendung der Anweisungen TSEND_C, TRCV_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND_C, TRCV_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Aufbau der Verbindungsbeschreibung nach TCON_Param

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	block_length	UINT	64	Länge: 64 Bytes (fest)
2 … 3	id	CONN_OUC	1	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben.
4	connection_type	USINT	17	Verbindungstyp: • 17: TCP (17 dez = 0x11 hex) • 18: ISO-on-TCP (18 dez = 0x12 hex) • 19: UDP (19 dez = 0x13 hex)
5	active_est	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus. Bei UDP gilt immer FALSE, da über die lokale ID Daten gesendet oder empfangen werden können. Für TCP und ISO-on-TCP gilt: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau
6	local_device_id	USINT	1	Kennung für die lokale PN/IE-Schnittstelle.
7	local_tsap_id_len	USINT	0	Verwendete Länge des Parameters local_tsap_id in Byte; mögliche Werte: • 0 oder 2, falls Verbindungstyp = 17 (TCP) Für die aktive Seite ist nur der Wert 0 zulässig. • 2 bis 16, falls Verbindungstyp = 18 (ISO-on-TCP) • 2, falls Verbindungstyp = 19 (UDP)
8	rem_subnet_id_len	USINT	0	Dieser Parameter wird nicht verwendet.
9	rem_staddr_len	USINT	4	Länge der Adresse des Partner-Endpunkts in Byte: • 0: unspezifiziert, d. h. Parameter rem_staddr ist irrelevant. • 4: gültige IP-Adresse im Parameter rem_staddr (nur TCP und ISO-on-TCP)

Byte	Parameter	Datentyp	Startwert	Beschreibung
10	rem_tsap_id_len	USINT	2	Verwendete Länge des Parameters rem_tsap_id in Byte; mögliche Werte: • 0 oder 2, falls Verbindungstyp = 17 (TCP) Für die passive Seite ist nur der Wert 0 zulässig. • 2 bis 16, falls Verbindungstyp = 18 (ISO-on-TCP) • 0, falls Verbindungstyp = 19 (UDP)
11	next_staddr_len	USINT	0	Dieser Parameter wird nicht verwendet.

12 … 27	local_tsap_id	ARRAY [1..16] of BYTE	-	Lokale Adresskomponente der Verbindung:
				• TCP und UDP: lokale Port-Nr. (mögliche Werte: 1...49151; empfohlene Werte: 2000...5000);
				local_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung;
				local_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung;
				local_tsap_id[3-16] = irrelevant
				• ISO-on-TCP: lokale TSAP-ID:
				local_tsap_id[1] = B#16#E0;
				local_tsap_id[2] = Rack und Steckplatz der lokalen Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer);
				local_tsap_id[3-16] = TSAP-Erweiterung, optional
				Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id innerhalb der CPU eindeutig ist.

28 … 33	rem_subnet_id	ARRAY [1..6] of USINT	-	Dieser Parameter wird nicht verwendet.
34 … 39	rem_staddr	ARRAY [1..6] of USINT	-	Nur TCP und ISO-on-TCP: IP-Adresse des Partner-Endpunkts, z. B. für 192.168.002.003: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 002 • rem_staddr[4] = 003 • rem_staddr[5-6]= irrelevant

40 … 55	rem_tsap_id	ARRAY [1..16] of BYTE	-	Partner-Adresskomponente der Verbindung
				• TCP: Partnerport-Nr. (mögliche Werte: 1...49151; empfohlene Werte: 2000...5000);
				rem_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung;
				rem_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung;
				rem_tsap_id[3-16] = irrelevant
				• ISO-on-TCP: Partner-TSAP-ID:
				rem_tsap_id[1] = B#16#E0;
				rem_tsap_id[2] = Rack und Steckplatz des Partner-Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer);
				rem_tsap_id[3-16] = TSAP-Erweiterung, optional
				• UDP: Dieser Parameter wird nicht verwendet.

Byte	Parameter	Datentyp	Startwert	Beschreibung
56 … 61	next_staddr	ARRAY [1..6] of BYTE	-	Dieser Parameter wird nicht verwendet.
62 … 63	spare	WORD	W#16#0000	Reserviert.

Hinweis

TCON_Param für S7-1500 CPU

Der Verbindungsbeschreibungs-DB mit der Struktur nach TCON_Param wird aus Migrationsgründen auch von CPUs der S7-1500 unterstützt. Es wird jedoch empfohlen, die neuen Strukturen TCON_IP_v4 und TCON_IP_RFC zu verwenden.

Hinweis

TCON_Param für S7-1500-R/H-CPUs

Der Verbindungsbeschreibungs-DB mit der Struktur nach TCON_Param wird von S7-1500-R/H-CPUs nicht unterstützt.

Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 112)

Beschreibung der Verbindungsparameter (Seite 103)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 131)

Übersicht über die Verbindungsparametrierung (Seite 100)

TSAP-Struktur (Seite 132)

Vergabe von Port-Nummern (Seite 129)

Verbindungsparameter nach TCON_IP_v4

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP und UDP zu parametrieren, wird für CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_IP_v4 verwendet. Die feste Datenstruktur des TCON_IP_v4 enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei Verwendung der Anweisungen TSEND_C, TRCV_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND_C, TRCV_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Aufbau der Verbindungsbeschreibung nach TCON_IP_v4

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceId	HW_ANY	64	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	1	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben.
4	ConnectionType	BYTE	11	Verbindungstyp: • 11: TCP (11 dez = 0x0B hex) • 19: UDP (19 dez = 0x13 hex) Die CPU S7-1500 erlaubt aus Kompatibilitäts- und Migrationsgründen·auch die Werte des Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_Param. Für den Verbindungstyp TCP ist daher auch der Eintrag 17 zulässig (17 dez = 0x11 hex).
5	ActiveEstablished	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau
6 … 9	RemoteAddress	ARRAY [1..4] of BYTE	-	• IP-Adresse des Partner-Endpunkts, z. B. für 192.168.0.1: – addr[1] = 192 – addr[2] = 168 – addr[3] = 0 – addr[4] = 1 • Multicast-Adresse einer IPv4-Multicast-Gruppe (Für S7-1500-CPUs ab Firmware-Stand V2.0 bzw. V2.1 bei Benutzung von TSEND_C/TRCV_C)
10 … 11	RemotePort	UINT	2000	Port-Adresse des entfernten Verbindungspartners (Wertebereich: 1 bis 49151).
12 … 13	LocalPort	UINT	2000	Port-Adresse des lokalen Verbindungspartners (Wertebereich: 1 bis 49151).

Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 112)

Beschreibung der Verbindungsparameter (Seite 103)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 131)

Übersicht über die Verbindungsparametrierung (Seite 100)

Vergabe von Port-Nummern (Seite 129)

Verbindungsparameter nach TCON_IP_V4_SEC

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP über IPv4 mittels Secure Communication zu parametrieren, verwenden Sie folgende Module:

- S7-1500-CPU ab Firmware-Version V2.0
  mit Verbindungsbeschreibungs-DB mit der Struktur des SDT TCON_IP_V4_SEC
- Zusätzlich optional: CP 1543-1 ab Firmware-Version V2.0 bzw. CP1543SP-1 ab Firmware-Version V1.0

Hinweis

Non-secure TCP- oder UDP-Verbindung über IPv4

Sie können den SDT TCON_IP_V4_SEC auch für eine non-secure TCP- oder UDP-Verbindung über IPv4 benutzen.

Der Verbindungsparameter CONNECT der Instanz-DBs für TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Aufbau der Verbindungsbeschreibung nach TCON_IP_V4_SEC

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 15	ConnPara	TCON_IP_v4	-	SDT für die Verbindungsparameter Hinweis zu interface_id: • Wenn Sie interface_id auf dem voreingestellten Wert 0 belassen, wertet das Betriebssystem der CPU die remote IP-Adresse und die lokal vorhandenen IP-Routen aus und legt daraufhin eine Industrial-Ethernet-Schnittstelle der CPU für den Aufbau der Secure OUC-Verbindung fest. Die Diagnosedaten werden in diesem Fall stets der ersten Industrial-Ethernet-Schnittstelle der CPU zugeordnet. • Wenn Sie als interface_id die Hardware-Kennung einer Industrial-Ethernet-Schnittstelle der CPU oder eines CP angeben, wird die Secure OUC-Verbindung über die zugehörige Industrial-Ethernet-Schnittstelle aufgebaut.
16	ActivateSecure-Conn	BOOL	FALSE	Aktivierung von Secure Communication für diese Verbindung Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Security-Parameter irrelevant, das heißt die Verbindung ist ungesichert. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
17	TLSServerReqCli-entCert	BOOL	FALSE	Nur für die Server-Seite: Anforderung eines X.509-V3-Zertifikats vom TLS-Client

Byte	Parameter	Datentyp	Startwert	Beschreibung
18 ... 19	ExtTLSCapabilities	WORD	16#0	• Bit 0: Nur für die Client-Seite. Ein gesetztes Bit bedeutet, dass der Client den alternativen Namen des Zertifikatsinhabers (subjectAlternateName) im X.509-V3-Zertifikat des Servers validiert, um die Identität des Servers zu überprüfen. Die Zertifikate werden beim Verbindungsaufbau überprüft. • Bit 1 bis 15: für zukünftige Erweiterungen reserviert
20 ... 23	TLSServerCertRef	UDINT	0	• Server-Seite: ID des eigenen X.509-V3-Zertifikats • Client-Seite: ID des X.509-V3-Zertifikats (gewöhnlich ein CA-Zertifikat), das vom TLS-Client benutzt wird, um die Authentifizierung des TLS-Servers zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Client zur Validierung der Server-Authentifizierung alle (CA-) Zertifikate, die aktuell im Zertifikatspeicher des Client geladen sind.
24 … 27	TLSClientCertRef	UDINT	0	• Client-Seite: ID des eigenen X.509-V3-Zertifikats • Server-Seite: ID des X.509-V3-Zertifikats (oder einer Gruppe von X.509-V3-Zertifikaten), das vom TLS-Server benutzt wird, um die Authentifizierung des TLS-Client zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Server zur Validierung der Client-Authentifizierung alle (CA-) Zertifikate, die aktuell im Zertifikatspeicher des Servers geladen sind.

Siehe auch

Verbindungsparameter nach TCON_IP_v4 (Seite 117)

Secure OUC zwischen zwei S7-1500 CPUs (Seite 351)

Verbindungsparameter nach TCON_QDN

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP und UDP über den voll qualifizierten Domänen-Namen zu parametrieren, verwenden Sie für CPUs der S7-1500 ab Firmware-Version V2.0 einen Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_QDN.

Der Verbindungsparameter CONNECT der Instanz-DBs für TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Hinweis

Struktur TCON_DNS

TCON_QDN ersetzt TCON_DNS. Verwenden Sie daher nur Verbindungsbeschreibungs-DBs mit einer Struktur nach TCON_QDN.

Aufbau der Verbindungsbeschreibung nach TCON_QDN

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	interface_id	HW_ANY	0	irrelevant Hinweis: Ab der Firmware-Version V2.0 der S7-1500-CPUs gilt: Solange die Verbindung aufgebaut wird, ist sie der ersten PROFINET-Schnittstelle zugeordnet. Sobald die Verbindung erfolgreich aufgebaut ist, wird sie der verwendeten PROFINET-Schnittstelle zugeordnet.
2 … 3	id	CONN_OUC	1	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TCON unter ID angeben.
4	connection_type	BYTE	11	Verbindungstyp: • 11: TCP (11 dez = 0x0B hex) • 19: UDP (19 dez = 0x13 hex) Die CPU S7-1500 erlaubt aus Kompatibilitäts- und Migrationsgründen·auch die Werte des Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_Param. Für den Verbindungstyp TCP ist daher auch der Eintrag 17 zulässig (17 dez = 0x11 hex).
5	active_established	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau
6 … 261	remote_qdn	Array of STRING [1..254]	-	Fully qualified domain name des Partner-Endpunkts, der mit "." abgeschlossen sein muss. Beachten Sie, dass in einer SIMATIC-Umgebung der Name einschließlich des abschließenden Punkts maximal 254 Zeichen lang sein darf.
262 … 263	remote_port	UINT	2000	Port-Adresse des entfernten Verbindungspartners (Wertebereich: 1 bis 49151).
264 … 265	local_port	UINT	2000	Port-Adresse des lokalen Verbindungspartners (Wertebereich: 1 bis 49151).

Verbindungsparameter nach TCON_QDN_SEC

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP über den voll qualifizierten Domainnamen mittels secure communication zu parametrieren, verwenden Sie für CPUs der S7-1500 ab Firmware-Version V2.0 einen Verbindungsbeschreibungs-DB mit der Struktur des SDT TCON_QDN_SEC.

Hinweis

Non-secure TCP- oder UDP-Verbindung über den voll qualifizierten Domainnamen

Sie können den SDT TCON_QDN_SEC auch für eine non-secure TCP- oder UDP-Verbindung über den voll qualifizierten Domainnamen benutzen.

Der Verbindungsparameter CONNECT der Instanz-DBs für TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Aufbau der Verbindungsbeschreibung nach TCON_QDN_SEC

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 271	ConnPara	TCON_QDN	-	Parameter der Verbindung
272	ActivateSecure-Conn	BOOL	FALSE	Aktivierung von secure communication für diese Verbindung Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
273	TLSServerReqClientCert	BOOL	FALSE	Nur für die Server-Seite: Anforderung eines X.509-V3-Zertifikats vom TLS-Client
274 .. . 275	ExtTLSCapabilities	WORD	16#0	• Bit 0: Nur für die Client-Seite. Ein gesetztes Bit bedeutet, dass der Client den subjectAlternateName im X.509-V3-Zertifikat des Servers validiert, um die Identität des Servers zu überprüfen. Die Zertifikate werden beim Verbindungsaufbau überprüft. • Bit 1 bis 15: für zukünftige Erweiterungen reserviert

Byte	Parameter	Datentyp	Startwert	Beschreibung
276 .. . 279	TLSServerCertRef	UDINT	0	• Server-Seite: ID des eigenen X.509-V3-Zertifikats • Client-Seite: ID des X.509-V3-Zertifikats (gewöhnlich ein CA-Zertifikat), das vom TLS-Client benutzt wird, um die Authentifizierung des TLS-Servers zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Client zur Validierung der Server-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Clients geladen sind.
280 … 283	TLSClientCertRef	UDINT	0	• Client-Seite: ID des eigenen X.509-V3-Zertifikats • Server-Seite: ID des X.509-V3-Zertifikats (oder einer Gruppe von X.509-V3-Zertifikaten), das vom TLS-Server benutzt wird, um die Authentifizierung des TLS-Clients zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Server zur Validierung der Client-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Servers geladen sind.

Siehe auch

Verbindungsparameter nach TCON_QDN (Seite 120)

Secure OUC zwischen zwei S7-1500 CPUs (Seite 351)

Secure OUC von einer S7-1500 CPU als TLS-Client zu einem Fremd-PLC (TLS-Server) (Seite 355)

Secure OUC von einer S7-1500 CPU als TLS-Server zu einem Fremd-PLC (TLS-Client) (Seite 357)

Secure OUC über E-Mail (Seite 364)

Verbindungsparameter nach TCON_IP_RFC

Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei ISO-on-TCP zu parametrieren, wird für CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_IP_RFC verwendet. Die feste Datenstruktur des TCON_IP_RFC enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei Verwendung der Anweisungen TSEND_C, TRCV_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND_C, TRCV_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

Aufbau der Verbindungsbeschreibung nach TCON_IP_RFC

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	interface_id	HW_ANY	64	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	id	CONN_OUC	1	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben.
4	connection_type	BYTE	12	Verbindungstyp 12: ISO-on-TCP (12 dez = 0x0C hex) Die CPU S7-1500 erlaubt aus Kompatibilitäts- und Migrationsgründen·auch die Werte des Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_Param. Für den Verbindungstyp ISO-on-TCP ist daher auch der Eintrag 18 zulässig (18 dez = 0x12 hex).
5	active_established	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau
8 … 11	remote_address	ARRAY [1..4] of BYTE	-	IP-Adresse des Partner-Endpunkts, z. B. für 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1
12 … 45	remote_tselector	TSelector	-	TSelector des entfernten Verbindungspartners: • TSelLength = Wertebereich 0 bis 32 als UINT • TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes
46 … 79	local_tselector	TSelector	-	TSelector des lokalen Verbindungspartners: • TSelLength = Wertebereich 0 bis 32 als UINT • TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes

Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 112)
Beschreibung der Verbindungsparameter (Seite 103)
Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 131)
Übersicht über die Verbindungsparametrierung (Seite 100)
TSAP-Struktur (Seite 132)

Verbindungsparameter nach TCON_Phone

Systemdatentypen TCON_... für Telecontrol-Verbindungen

TCON_PHONE wird als Verbindungsbeschreibung für folgende Verbindungstypen eingesetzt:

- Bei Mobilfunk-CPs der S7-1200 für Verbindungen der Open User Communication zur Übertragung von SMS über TCON / TDISCON / TSEND oder TSEND_C
- Beim CP 1242-7 für Telecontrol-Verbindungen mit SMS-Clients über TC_CON

Um den Aufbau einer Verbindung mithilfe einer der zuvor genannten Anweisungen zu projektieren, wird der Parameter CONNECT der Anweisung für die Verbindungsbeschreibung verwendet.

Die Verbindungsbeschreibung ist durch die Struktur des Systemdatentyps (SDT) TCON_Phone vorgegeben. Die Struktur des jeweiligen SDT enthält die notwendigen Parameter, die zum Aufbau der Verbindung mit dem entfernten Kommunikationspartner erforderlich sind.

Anlegen eines DB vom Typ TCON_Phone

Den Datentyp des DB müssen Sie über die Tastatur eintippen. Er wird nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe des Datentyps keine Rolle.

Gehen Sie zum Anlegen des DB folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an. Deaktivieren Sie hierzu in den Bausteineigenschaften (Kontextmenü) im Register "Attribute" die Eigenschaft "Optimierter Zugriff".
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB einen SDT an.
3. Tippen Sie in der Deklarationstabelle des DB in die Zelle des Datentyps den folgenden Namen ein: TCON_Phone
   Der SDT mit seinen Parametern (siehe unten) wird angelegt.
4. Projektieren Sie die Parameter, die nachfolgend beschrieben sind.

Reservierte Bits werden nicht angezeigt.

Systemdatentyp TCON_Phone für SMS-Verbindungen

Hinweis

SMS-Text

- Auf programmierte SMS-Texte für zu sendende SMS wird über den Parameter DATA der Anweisung TC_SEND zugegriffen.
- Der Text einer empfangenen SMS wird dem Adressbereich der CPU über den Parameter DATA der Anweisung TC_RECV zugewiesen.

Tabelle 1-1 Parameter von TCON_Phone

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceId	HW_ANY		Referenz auf die lokale Schnittstelle (Telecontrol-Schnittstelle bzw. Ethernet-Schnittstelle des CP > "HW-Kennung")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale Mobilfunk-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hinweis zu Telecontrol-Verbindungen des CP 1242-7: Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	Byte	W#16#0E	Protokoll-Variante 14 (E_h): SMS-Verbindung
5	ActiveEstablished	Bool		Kennung für die Art des Verbindungsaufbaus (nicht relevant für den CP 1242-7): • 0: Passiver Verbindungsaufbau (hier nicht relevant) • 1: Aktiver Verbindungsaufbau
6...7	-	-	-	- reserviert -
8 ... 31	PhoneNumber	String[22]		Rufnummer des Verbindungspartners Erlaubte Werte: Plus-Zeichen (+) und Ziffern Achten Sie auf die genaue Zeichenfolge der vom Netzbetreiber zugewiesenen Ländervorwahl der betreffenden Rufnummer ("+"-Zeichen oder Nullen). Hinweis zu Telecontrol-Verbindungen des CP 1242-7: Ohne Eintrag des Parameters PhoneNumber wird kein Verbindungspartner spezifiziert und SMS können von allen autorisierten Verbindungspartnern empfangen werden. Ohne Eintrag liefert TC_RECV beim Anlauf zunächst die älteste empfangene SMS.

Verbindungsparameter nach TCON_FDL

Systemdatenbaustein (SDT) für FDL-Verbindungsbeschreibung

Um eine FDL-Verbindung für das CM 1542-5 zu parametrieren, wird ein Verbindungsbeschreibungs-DB mit einem SDT nach TCON_FDL verwendet. Die feste Datenstruktur des SDT enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-SDT muss in einem angelegten DB angelegt werden.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND_C bzw. TRCV_C enthält einen Verweis auf den verwendeten Datenbaustein.

Folgende Arten von FDL-Verbindungen können aufgebaut werden:

- **Projektierte FDL-Verbindungen**
  Diese Verbindungsarten werden im STEP 7-Programmeditor projektiert, siehe AUTOHOTSPOT. Als Bausteine werden TSEND_C und TRCV_C verwendet.
  - Spezifizierte Verbindung
    Vollprojektierte Verbindung zwischen zwei Mastern
  - Unspezifizierte Verbindung
    Projektierte Verbindung mit einem unspezifizierten Partner
- **Programmierte FDL-Verbindungen**
  Diese Verbindungsarten sind nicht im STEP 7-Programmeditor projektierbar. Statt dessen ist für diese Verbindungsarten der Aufruf der Anweisung TCON erforderlich zusammen mit TSEND/TRCV bzw. TUSEND/TURCV.
  - Unspezifizierte Layer-2-Verbindung
    Programmierte FDL-Verbindung mit einem unspezifizierten Partner mit freiem Layer-2-Zugang
  - Broadcast-Verbindung
    Verbindung mit allen angeschlossenen Partnern
  - Multicast-Verbindung
    Verbindung mit mehreren definierten Partnern

Wenn spezifische Einstellungen für die einzelnen Verbindungsarten erforderlich sind, wird dies bei den nachfolgenden Parametern des SDT erwähnt.

Aufbau der Verbindungsbeschreibung "TCON_FDL" für projektierte Verbindungen

Die nachfolgende Tabelle beschreibt die Verbindungsparameter für folgende projektierte FDL-Verbindungen:

- Spezifizierte Verbindung
- Unspezifizierte Verbindung

Byte	Parameter	Datentyp	Wertebereich / Startwert	Beschreibung
0 … 1	InterfaceId	HW_ANY	0 .. 65535	Hardware-Kennung der lokalen Schnittstelle
2 … 3	ID	CONN_OUC	0 .. 4095	Referenz auf diese Verbindung gemäß Verbindungsprojektierung
4	ConnectionType	Byte	15	Verbindungstyp: 0x15 (dez. 21) für FDL
5	ActiveEstablished	Bool	FALSE	Kennung für die Art des Verbindungsaufbaus • FALSE: Passiver Verbindungsaufbau • TRUE: Aktiver Verbindungsaufbau Der Parameterwert wird nicht weiter verarbeitet.

Byte	Parameter	Datentyp	Wertebereich / Startwert	Beschreibung
5	ServiceId	Byte	0 .. 2	Kennung des Datenübertragungsdienstes • 0: Automatische Voreinstellung mit Auswahl des Dienstes gemäß des in der Verbindungsbeschreibung projektierten Empfängers Für projektierte Verbindungen wird in dieser Einstellung nur SDA verwendet. • 1: Send Data Acknowledged (SDA) - für aktive Verbindungen zwischen zwei Mastern Verwenden Sie diese Einstellung für unspezifizierte Verbindungen. • 2: Send Data Non-Acknowledged (SDN) - für Multicast/ Broadcast
6 … 9	RemotePBAddress	Byte	0 .. 126	PROFIBUS-Adresse des Verbindungspartners • Wird für spezifizierte Verbindungen durch die Verbindungsprojektierung vergeben. • Ist für unspezifizierte Verbindungen explizit anzugeben.
6 … 9	LocalPBAddress	Byte	0 .. 126	Eigene (lokale) PROFIBUS-Adresse Wird automatisch von der Schnittstellenprojektierung vergeben.
10 … 11	RemoteLSAP	Byte	2 .. 32	LSAP (link layer service access point) des Verbindungspartners Ist für projektierte Verbindungen explizit anzugeben.
12 … 13	LocalLSAP	Byte	2 .. 32	Eigener (lokaler) LSAP Ist für projektierte Verbindungen explizit anzugeben.

Aufbau der Verbindungsbeschreibung "TCON_FDL" für programmierte Verbindungen

Die nachfolgende Tabelle beschreibt die Verbindungsparameter für folgende programmierte FDL-Verbindungen:

- Unspezifizierte Layer-2-Verbindung
- Broadcast-Verbindung
- Multicast-Verbindung

Byte	Parameter	Datentyp	Wertebereich / Startwert	Beschreibung
0 … 1	InterfaceId	HW_ANY	0 .. 65535	Hardware-Kennung der lokalen Schnittstelle
2 … 3	ID	CONN_OUC	0 .. 4095	Referenz auf diese Verbindung Referenzieren Sie auf den Wert der von TCON erzeugten ID. Geben Sie den Wert für TSEND/TRCV bzw. TUSEND/ TURCV am Parameter ID an.
4	ConnectionType	Byte	15	Verbindungstyp: 0x15 (dez. 21) für FDL

Byte	Parameter	Datentyp	Wertebereich / Startwert	Beschreibung
5	ActiveEstablished	Bool	FALSE	Kennung für die Art des Verbindungsaufbaus • FALSE: Passiver Verbindungsaufbau • TRUE: Aktiver Verbindungsaufbau Der Parameterwert wird nicht weiter verarbeitet.
5	ServiceId	Byte	0 .. 2	Kennung des Datenübertragungsdienstes • 0: Automatische Auswahl des Dienstes. Nicht zu verwenden für programmierte Verbindungen. • 1: Send Data Acknowledged (SDA) - für aktive Verbindungen zwischen zwei Mastern • 2: Send Data Non-Acknowledged (SDN) - für Multicast/ Broadcast Für programmierte Verbindungen muss "1" oder "2" programmiert werden.
6 … 9	RemotePBAddress	Byte	0 .. 127	PROFIBUS-Adresse des Verbindungspartners Geben Sie die Adresse bei Verwendung von TSEND bzw. TUSEND an. Für TRCV bzw. TURCV wird der Wert durch das Anwenderprogramm gesetzt. • Für Broadcast/Multicast: 127
6 … 9	LocalPBAddress	Byte	0 .. 126	Eigene (lokale) PROFIBUS-Adresse Wird automatisch von der Schnittstellenprojektierung vergeben.
10 … 11	RemoteLSAP	Byte	2 .. 32, 63	LSAP (link layer service access point) des Verbindungspartners Geben Sie die Adresse bei Verwendung von TSEND bzw. TUSEND an: • Für unspezifizierte Layer-2-Verbindungen: 2 .. 32. • Für Broadcast: 63 • Für Multicast: Verwenden Sie den gleichen Werte wie "LocalLSAP". Für TRCV bzw. TURCV wird der Wert durch das Anwenderprogramm gesetzt.
12 … 13	LocalLSAP	Byte	2 .. 32	Eigener (lokaler) LSAP Ist programmseitig zu vergeben.

Vergabe von Port-Nummern

Einführung

Beim Anlegen einer Open User Communication wird automatisch der Wert 2000 als Port-Nummer vergeben.

Für die Portnummern sind Werte von 1 bis 49151 erlaubt. Innerhalb dieses Bereichs können Sie Port-Nummern frei vergeben. Da einige Ports jedoch bereits systembedingt verwendet

werden können, wird die Verwendung von Port-Nummern im Bereich von 2000 bis 5000 empfohlen.

Hinweis

Port-Nummern müssen eindeutig sein. Bei doppelt vergebenen Port-Nummern wird die Verbindungsprojektierung bzw. ein entsprechender Bausteinaufruf mit einem Fehler abgelehnt.

Übersicht der Port-Nummern

Die folgende Tabelle zeigt eine Zusammenfassung der Systemreaktionen auf verschiedene Portnummern.

Port-Nr.	Beschreibung	Systemreaktion
2000 … 5000	empfohlener Bereich	Keine Warnung, keine Fehlermeldung bei Eingabe Port-Nummer wird zugelassen und übernommen
1 … 1999, 5001 … 49151	nutzbar aber ausserhalb des empfohlenen Bereiches	Warnmeldung bei Eingabe Port-Nummer wird zugelassen und übernommen
0, 20, 21, 25, 80, 102, 135, 161, 443, 465, 989, 990, 8080, 34962 … 34964	bedingt verwendbar*	
53, 80, 102, 135, 161, 162, 443, 520, 9001, 34962 … 34964	bedingt verwendbar**	

* Für bestimmte Funktionen festgelegte Ports ("well-known Ports"):
0: ANY - Port-Nummer wird automatisch von S7-1500 CPU als Wert vergeben (>49151)
20: FTP-Datenübertragung
21: FTP-Kontrolle
25: SMTP (Simple Mail Transfer Protocol) / TMAIL_C
80: HTTP (Hypertext Transfer Protocol) / Webserver
102: ISO-on-TCP (RFC1006)
135: DCE Endpoint Mapper für PROFINET
161: SNMP (Simple Network Management Protocol)
443: HTTPS (Hypertext Transfer Protocol Secure)
465: SMTPS (Simple Mail Transfer Protocol Secure)
989: FTPS-Datenübertragung
990: FTPS-Kontrolle
8080: HTTP alternativ / internal Web
34962 ... 34964: PROFINET

Hinweis

Bei UDP/TCP gibt der Anwender auf dem aktiven Verbindungsendpunkt üblicherweise den Wert 0 für den lokalen Port vor. In diesem Fall wählt das CPU-Betriebssystem den nächsten freien Port oberhalb 49151 aus. Beim passiven Verbindungsendpunkt wird umgekehrt der Partnerport meist mit 0 vorbelegt. In der Verbindungsparametrierung wird der entsprechende Parameter deaktiviert.

** Diese Ports sind abhängig vom Funktionsumfang der verwendeten CPU gesperrt. Die Belegung dieser Ports geht aus den Dokumentationen der jeweiligen CPUs hervor.

Siehe auch

Beschreibung der Verbindungsparameter (Seite 103)

Verbindungen anlegen und parametrieren (Seite 107)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung

Ändern von Parameterwerten in der Verbindungsbeschreibung

Im Verbindungsbeschreibungs-DB wird von der Verbindungsparametrierung die Verbindungsbeschreibung für genau eine Verbindung der Open User Communication eingetragen.

Sie können die Parameterwerte des Verbindungsbeschreibungs-DB außerhalb der Verbindungsparametrierung im Anwenderprogramm ändern. Verbindungsbeschreibungs-DBs mit nachträglich von Ihnen geänderten Werten können von der Verbindungsparametrierung rückgelesen werden. Im Inspektorfenster werden unter "Eigenschaften > Konfiguration > Verbindungsparameter" nur die im Verbindungsbeschreibungs-DB abgelegten Verbindungsparameter angezeigt.

Hinweis

Im laufenden Anwenderprogramm dürfen die Werte nur geändert werden, wenn die Anweisungen TCON, TSEND_C oder TRCV_C nicht bearbeitet werden bzw. die referenzierte Verbindung nicht aufgebaut ist.

Verschachtelt eingetragene Verbindungsbeschreibungen in nur per Offset-Referenzierung auffindbare DB-Typen (z. B. Global-DB) werden von der Verbindungsparametrierung nicht unterstützt.

Die Struktur der Verbindungsbeschreibung kann nicht geändert werden.

Rücklesbarkeit der einzelnen Verbindungsparameter

Für den Parameter "Adresse" des Kommunikationspartners in einer TCP- oder ISO-on-TCP-Verbindung wird dessen IP-Adresse aus dem Parameter "rem_staddr" der Verbindungsbeschreibung angezeigt.

Folgende Werte sind außerdem aus der Verbindungsbeschreibung rückladbar:

- Verbindungstyp
- Lokale Verbindungs-ID
- Verbindungsaufbau aktiv/passiv (nicht bei UDP)
- Lokaler TSAP (nur bei ISO-on-TCP)
- Partner-TSAP (nur bei ISO-on-TCP)

- Lokaler Port (nur bei TCP und UDP)
- Partnerport (nur bei TCP)

Die Werte der Parameter der Verbindungs-ID des Kommunikationspartners, der Verbindungsdaten sowie des Verbindungsaufbaus sind nicht Inhalt der Verbindungsbeschreibung im lokalen Verbindungsbeschreibungs-DB. Diese Parameter können somit beim erneuten Öffnen der Verbindungsparametrierung nicht angezeigt werden. Der Verbindungsaufbau des Partners ergibt sich aber aus dem lokalen Verbindungsaufbau und wird daher auch angezeigt.

In der Klappliste "Partner" kann jederzeit ein neuer Kommunikationspartner ausgewählt werden.

Bei Auswahl einer im Projekt erkannten CPU als spezifizierten Kommunikationspartner werden die Eingabemöglichkeiten für die Verbindungs-ID und die Verbindungsdaten wieder eingeblendet.

Siehe auch

Verbindungsparameter nach TCON_Param (Seite 115)

Beschreibung der Verbindungsparameter (Seite 103)

TSAP-Struktur

Einführung

Für eine Verbindung mit dem Verbindungstyp ISO-on-TCP müssen für beide Kommunikationspartner Transportation Service Access Points (TSAPs) vergeben werden. TSAP-IDs werden automatisch nach dem Anlegen einer ISO-on-TCP-Verbindung vergeben. Um die Eindeutigkeit von TSAP-IDs innerhalb eines Geräts zu gewährleisten, können Sie in der Verbindungsparametrierung die vorgegebenen TSAPs ändern.

Aufbau der TSAPs

Für die freie Vergabe von TSAPs müssen Sie bestimmte Regeln einhalten. Ein TSAP muss eine bestimmte Anzahl an Bytes beinhalten, die in der Verbindungsparametrierung als hexadezimale Werte (TSAP-ID) oder als ASCII-Zeichen (ASCII-TSAP) angezeigt und eingegeben werden können:

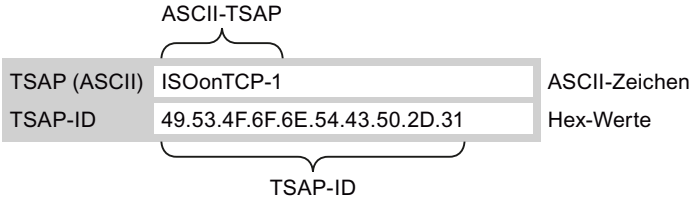

Einträge oder Änderungen der TSAP-ID oder der ASCII-TSAP in den entsprechenden Eingabefeldern wirken sich immer auch auf das jeweils andere Anzeigeformat aus.

Wenn ein TSAP keine gültigen ASCII-Zeichen enthält, wird der TSAP nur als TSAP-ID angezeigt, aber nicht als ASCII-TSAP. Dies ist nach dem Anlegen einer Verbindung der Fall.

Die ersten beiden Hex-Zeichen als TSAP-ID kennzeichnen die Kommunikationsart und den Baugruppenträger/Steckplatz. Da diese Zeichen für eine CPU keine gültigen ASCII-Zeichen sind, wird in diesem Fall der ASCII-TSAP nicht angezeigt:

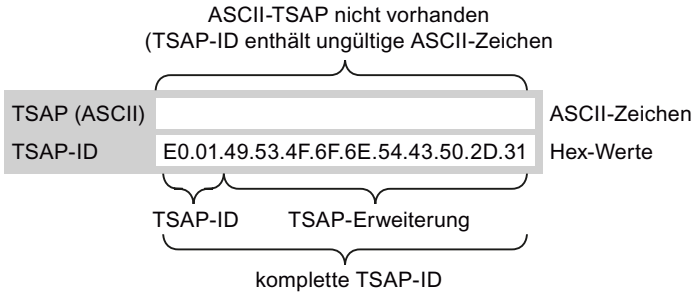

Neben den Regeln für Länge und Aufbau der TSAPs müssen Sie auf die Eindeutigkeit der TSAP-ID achten. TSAPs werden nicht automatisch eindeutig vergeben.

Länge und Inhalt der TSAPs

Aufbau der TSAP-ID mit TSAP-Erweiterung:

- Gültig für CPU S7-1200 FW V1
  Länge = 2 bis 16 Byte
  x_tsap_id[0] = 0xE0 (Open User Communication)
  x_tsap_id[1] (Bits 0 bis 4) = Steckplatz-Nummer der CPU
  x_tsap_id[1] (Bits 5 bis 7) = Baugruppenträger-Nummer der CPU
  x_tsap_id[2...15] = beliebige Zeichen (TSAP-Erweiterung, optional)
  (x = loc (lokal) oder x = rem (Partner))
- Gültig für CPU S7-1500 und S7-1200 FW ab V2
  Länge = 2 bis 16 Byte
  x_tsap_id[0] = 0xE0 (Open User Communication)
  x_tsap_id[1] = 0x00 bis 0xFF
  x_tsap_id[2...15] = beliebige Zeichen (TSAP-Erweiterung, optional)
  (x = loc (lokal) oder x = rem (Partner))

Aufbau der TSAP-ID als ASCII-TSAP:

- Gültig für CPU S7-1200 FW V1
  Länge = 3 bis 16 Byte
  x_tsap_id[0...2] = 3 ASCII-Zeichen (0x20 bis 0x7E)
  x_tsap_id[3...15] = beliebige Zeichen (optional)
  (x = loc (lokal) oder x = rem (Partner))
- Gültig für CPU S7-1200 FW ab V2
  Länge = 3 bis 16 Byte
  x_tsap_id[0...2] für aktive Verbindung = 3 ASCII-Zeichen (0x00 bis 0xFF) oder beliebige Bitfolge*
  x_tsap_id[0...2] für passive Verbindung = 3 ASCII-Zeichen (0x20 bis 0x7E) oder beliebige Bitfolge*
  x_tsap_id[3...15] = beliebige Zeichen (optional)
  (x = loc (lokal) oder x = rem (Partner))

- Gültig für CPU S7-1500
  Länge = 3 bis 16 Byte
  x_tsap_id[0...2] = 3 ASCII-Zeichen (0x00 bis 0xFF) oder beliebige Bitfolge*
  x_tsap_id[3...15] = beliebige Zeichen (optional)
  (x = loc (lokal) oder x = rem (Partner))

* ASCII-Zeichenfolgen dürfen nicht mit "SIMATIC-" beginnen

Die folgende Tabelle zeigt den schematischen Aufbau verschiedener TSAP-IDs:

TSAP-ID	tsap_id_len	tsap_id[0]	tsap_id[1]	tsap_id[2]	tsap_id[3..15]
...mit Erweiterung (CPU S7-1200 FW V1)	2...16 Byte	0xE0	0x01 oder 0x02 oder 0x00*	Erweiterung (optional)	Erweiterung (optional)
...mit Erweiterung (CPU S7-1500, S7-1200 FW ab V2)	2...16 Byte	0xE0	0x00...0xFF	Erweiterung (optional)	Erweiterung (optional)
...als ASCII-TSAP (CPU S7-1200 FW V1)	3...16 Byte	0x20...0x7E	0x20...0x7E	0x20...0x7	beliebig (optional)

* Eine CPU S7-1200 steckt für gewöhnlich auf Baugruppenträger 0 und Steckplatz 1, eine S7-300/400 CPU auf Baugruppenträger 0 und Steckplatz 2. Aus diesem Grund gilt für die zweite Stelle der TSAP-ID mit Erweiterung der Hex-Wert 01 oder 02. Sollte es sich bei dem Verbindungspartner um eine unspezifizierte CPU handeln, z. B. um ein Fremdgerät, ist für die Steckplatzadresse auch der Hex-Wert 00 zulässig.

Hinweis

Bei unspezifizierten Kommunikationspartnern dürfen die lokale TSAP-ID und die Partner-TSAP-ID eine Länge von 0 bis 16 Byte haben, wobei alle Hex-Werte von 00 bis FF erlaubt sind.

ASCII-Codetabelle für die Eingabe von ASCII-TSAPs

Bei der Eingabe von ASCII-TSAPs mit den Hexadezimal-Werten von 20 bis 7E sind lediglich folgende Zeichen zulässig:

Code	..0	..1	..2	..3	..4	..5	..6	..7	..8	..9	..A	..B	..C	..D	..E	..F
2..		!	"	#	$	%	&	’	(	)	*	+	,	-	.	/
3..	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4..	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5..	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6..	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7..	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

Siehe auch

Beispiele zur TSAP-Vergabe (Seite 135)

Beschreibung der Verbindungsparameter (Seite 103)

Verbindungen anlegen und parametrieren (Seite 107)

Beispiele zur TSAP-Vergabe

Die folgenden Beispiele zeigen die Bearbeitung der TSAPs für CPUs der S7-1200/1500 (CPU auf Steckplatz 1) unter verschiedenen Gesichtspunkten:

- Beispiel 1: Anlegen einer neuen Verbindung für eine PLC-PLC-Kommunikation
- Beispiel 2: Eingabe eines lokalen ASCII-TSAPs
- Beispiel 3: Eingabe einer TSAP-Erweiterung in der TSAP-ID
- Beispiel 4: Fehlerhaftes Editieren der TSAP-ID
- Beispiel 5: Eingabe eines ASCII-TSAPs über das Eingabefeld "TSAP-ID"

Beispiel 1: Anlegen einer neuen Verbindung für eine PLC-PLC-Kommunikation

Nachdem Sie für die Open User Communication eine neue Verbindung mit zwei PLCs angelegt haben, wird automatisch die TSAP-Erweiterung "ISOonTCP-1" vergeben.

Mit dieser TSAP-Erweiterung ergibt sich die TSAP-ID E0.01.49.53.4F.6F.6E.54.43.50.2D.31, die automatisch in den Verbindungsbeschreibungs-DB und in die Eingabefelder des lokalen und Partner-TSAPs eingetragen wird. Die Eingabefelder der ASCII-TSAPs bleiben leer:

	Lokaler TSAP	Partner-TSAP
TSAP (ASCII)		
TSAP-ID	E0.01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Sie können die Werte in den Eingabefeldern der TSAP-ID und des ASCII-TSAP jederzeit ändern.

Das Eingabefeld der TSAP-ID zeigt den vollständigen TSAP, abgelegt im Datenbaustein der Verbindungsbeschreibung. Die auf 16 Zeichen begrenzte TSAP-ID mit TSAP-Erweiterung wird nicht im Eingabefeld "TSAP (ASCII)" angezeigt, da das Zeichen E0 für den ASCII-TSAP kein gültiges Zeichen darstellt.

Wenn es sich bei der angezeigten TSAP-ID um eine gültige ASCII-TSAP handelt, wird sie im Eingabefeld "TSAP (ASCII)" angezeigt.

Änderungen in den Eingabefeldern für TSAP-ID und ASCII-TSAP beeinflussen sich gegenseitig.

Beispiel 2: Eingabe eines lokalen ASCII-TSAPs

Wenn Sie eine neue Verbindung angelegt haben und im Eingabefeld "TSAP (ASCII)" einen ASCII-Wert für den lokalen TSAP vergeben, z. B. "ISOonTCP-1", wird die resultierende TSAP-ID automatisch erzeugt.

Wenn Sie das Eingabfeld "TSAP (ASCII)" verlassen, wird automatisch die Einhaltung der ASCII-Zeichenbeschränkung von 3 bis 16 Zeichen geprüft und die resultierende TSAP-ID in das entsprechende Eingabefeld eingetragen:

	Lokaler TSAP	Partner-TSAP
TSAP (ASCII)	ISOonTCP-1	
TSAP-ID	49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Beispiel 3: Eingabe einer TSAP-Erweiterung in der TSAP-ID

Wenn Sie nach dem Anlegen einer Verbindung und der Eingabe eines ASCII-TSAPs (siehe Beispiele 1 und 2) in das Eingabefeld der lokalen TSAP-ID dem TSAP-Wert die Zeichen "E0.01." voranstellen, wird nach Verlassen des Eingabefeldes kein ASCII-TSAP mehr angezeigt:

	Lokaler TSAP	Partner-TSAP
TSAP (ASCII)		
TSAP-ID	E0.01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Nachdem Sie das Eingabefeld der TSAP-ID verlassen haben, wird automatisch geprüft, ob das erste Zeichen der TSAP-ID ein gültiges ASCII-Zeichen ist. Da das Zeichen "E0" in der TSAP-ID nun kein gültiges Zeichen für den ASCII-TSAP ist, wird im Eingabefeld "TSAP (ASCII)" kein ASCII-TSAP mehr angezeigt.

Bei der Verwendung gültiger ASCII-Zeichen erfolgt die Prüfung auf die Einhaltung der Länge von 2 bis 16 Zeichen.

Beispiel 4: Fehlerhaftes Editieren der TSAP-ID

Wenn Sie bei einer TSAP-ID beginnend mit "E0.01" den Hex-Wert "E0" entfernen, entspricht die nun mit "01" beginnende TSAP-ID nicht mehr den Regeln und ist ungültig:

	Lokaler TSAP	Partner-TSAP
TSAP (ASCII)		
TSAP-ID	01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Nach Verlassen des Eingabefeldes wird eine Meldung ausgegeben, da es sich bei der TSAP-ID weder um einen gültigen ASCII-TSAP handelt (dieser müsste als ersten Wert einen Hex-Wert im Bereich von 20 bis 7E haben), noch um eine gültige TSAP-ID (diese müsste als ersten Wert die Kennung "E0" besitzen).

Beispiel 5: Eingabe eines ASCII-TSAPs über das Eingabefeld "TSAP-ID"

Wenn Sie aus der fehlerhaften TSAP-ID des Beispiels 4 nach dem Wert "E0" auch noch den Wert "01" entfernen, beginnt die TSAP-ID mit dem Hex-Wert 49. Dieser Wert liegt im erlaubten Bereich für ASCII-TSAPs:

	Lokaler TSAP	Partner-TSAP
TSAP (ASCII)		
TSAP-ID	49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Wenn Sie das Eingabefeld verlassen, wird die TSAP-ID als gültiger ASCII-TSAP erkannt und der resultierende ASCII-TSAP "ISOonTCP-1" wird in das Eingabefeld "TSAP (ASCII)" geschrieben.

Siehe auch

TSAP-Struktur (Seite 132)

Beschreibung der Verbindungsparameter (Seite 103)

HMI-Verbindungen projektieren

Einführung zum Projektieren von Verbindungen

Definition

Eine Verbindung definiert eine logische Zuordnung zweier Kommunikationspartner zur Ausführung von Kommunikationsdiensten. Eine Verbindung legt Folgendes fest:

- Beteiligte Kommunikationspartner
- Typ der Verbindung (z. B. HMI-Verbindung)
- Spezielle Eigenschaften (z. B. ob eine Verbindung permanent aufgebaut bleibt oder ob sie im Anwenderprogramm dynamisch auf- und abgebaut wird und ob Betriebszustandsmeldungen gesendet werden sollen)
- Verbindungsweg

Wissenswertes zur Verbindungsprojektierung

Bei der Verbindungsprojektierung wird bei einer HMI-Verbindung ein lokaler Verbindungsname als eindeutige lokale Kennung vergeben.

In der Netzsicht wird neben dem Register "Netzübersicht" ein Register "Verbindungen" angezeigt. In diesem Register befindet sich die Verbindungstabelle. Eine Zeile in dieser Verbindungstabelle repräsentiert eine projektierte Verbindung, z. B. zwischen einem HMI-Gerät und einer PLC, mit ihren Eigenschaften.

Wissenswertes zum Verbrauch von Verbindungsressourcen

Einführung

Jede Verbindung benötigt auf den beteiligten Geräten Verbindungsressourcen für den Endpunkt bzw. für den Übergangspunkt. Die Anzahl der Verbindungsressourcen ist gerätespezifisch.

Sind alle Verbindungsressourcen eines Kommunikationspartners belegt, so kann keine neue Verbindung aufgebaut werden. Sie erkennen das daran, dass eine neu angelegte Verbindung in der Verbindungstabelle rot hinterlegt ist. Die Konfiguration ist dann inkonsistent und kann nicht übersetzt werden.

HMI-Verbindungen

Bei HMI-Verbindungen über die **integrierte** PN-Schnittstelle wird beim HMI-Gerät pro HMI-Verbindung eine Verbindungsressource für den Endpunkt belegt.

Für den Verbindungspartner (PLC) wird ebenfalls eine Verbindungsressource benötigt.

Ansichten mit Informationen zu den projektierten Verbindungen

Über die nachfolgend beschriebenen Ansichten haben Sie umfassenden Zugriff auf sämtliche Informationen und Funktionen zum Projektieren und Prüfen von Kommunikationsverbindungen.

- Verbindungsanzeige in der Netzsicht
- Verbindungstabelle
- Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

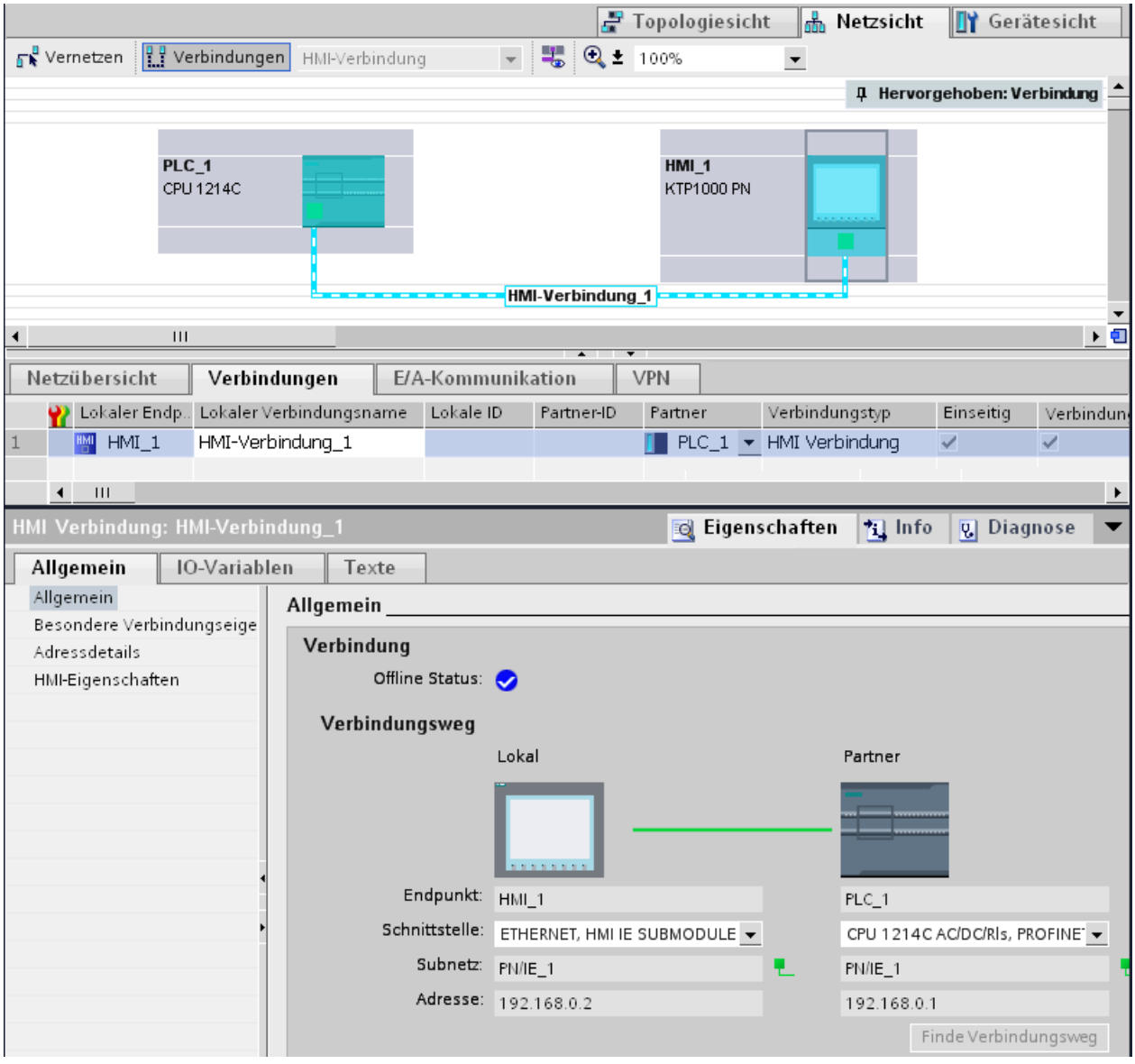

Vorteile

Die in diesen Ansichten dargestellten Informationen sind in Bezug auf augenblickliche Benutzeraktionen immer aktuell. Das bedeutet:

- Die Verbindungstabelle zeigt alle angelegten Verbindungen.
- Wenn Sie eine Verbindung in der Verbindungstabelle angewählt haben, gilt:
  - Sie sehen in der Netzsicht den Verbindungsweg grafisch.
  - Das Register "Eigenschaften" im Inspektorfenster zeigt die Parameter dieser Verbindung.

Die Verbindungstabelle

Die Verbindungstabelle bietet folgende Funktionen:

- Auflistung aller Verbindungen im Projekt
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht
- Ändern von Verbindungspartnern
- Anzeige von Zustandsinformationen

Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Der Eigenschaften-Dialog hat folgende Bedeutung:

- Anzeige für Verbindungsparameter
- Anzeige des Verbindungswegs
- Nachträgliches Spezifizieren von Verbindungen über die Schaltfläche "Finde Verbindungsweg"

Anlegen einer neuen Verbindung

Anlegen einer Verbindung - Alternativen

Sie haben folgende Möglichkeiten, eine Verbindung in der Netzsicht anzulegen:

- Grafische Verbindungsprojektierung
- Dialoggeführte Verbindungsprojektierung

Die einzelnen Schritte hierzu finden Sie in den Folgekapiteln.

Voraussetzung und Ergebnis

Sie haben in der Netzsicht die Geräte mit CPUs und HMI-Geräten angelegt, zwischen denen die Verbindungen projektiert werden sollen.

Verbindung spezifizieren

Wenn beide Partner für den gewählten Verbindungstyp am selben Netz vernetzt sind, legen Sie durch die grafische oder dialoggeführte Auswahl beider Verbindungspartner eine vollspezifizierte Verbindung an.

Diese wird automatisch in die Verbindungstabelle des HMI-Gerätes eingetragen. Für diese Verbindung wird ein lokaler Verbindungsname vergeben.

Die folgende Darstellung zeigt eine projektierte Verbindung bei vernetztem Gerät:

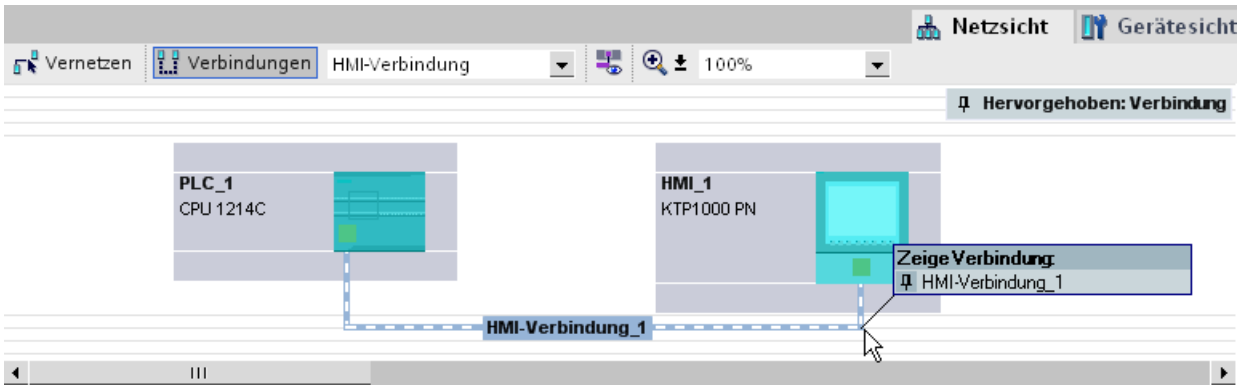

Neue Verbindung grafisch anlegen

Verbindungen grafisch projektieren

Bei der grafischen Verbindungsprojektierung werden Sie bei Bedarf vom System aufgefordert, den Verbindungsweg zu bestimmen. Wählen Sie in der aktuellen Konfiguration die zu verbindenden Geräte aus.

Verbindungsweg automatisch ermitteln

Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".

   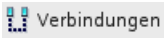

   Damit ist der Verbindungsmodus für den von Ihnen gewählten Verbindungstyp aktiviert. Sie erkennen dies an folgendem Verhalten:
   In der Netzsicht sind die für den gewählten Verbindungstyp in Ihrem Projekt verwendbaren Geräte farblich hervorgehoben.

2. Ziehen Sie den Mauscursor bei gedrückter Maustaste von dem Gerät, von dem eine Verbindung ausgehen soll, auf das Gerät, bei dem die Verbindung enden soll.

3. Lassen die auf dem Zielgerät die Maustaste los, um die Verbindung zwischen den beiden Geräten zu erstellen.

Ergebnis

- Eine spezifizierte Verbindung ist angelegt.
- Der Verbindungsweg ist hervorgehoben dargestellt.
- Die Verbindung ist in der Verbindungstabelle eingetragen.

Neue Verbindung dialoggeführt anlegen

Verbindungen dialoggeführt projektieren

Bestimmen Sie das lokale Gerät und seinen Verbindungspartner.

Vorgehen

Um eine Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü eines Verbindungspartners, für den eine Verbindung angelegt werden soll, den Befehl "Neue Verbindung erstellen".
   Der Dialog "Neue Verbindung erstellen" wird geöffnet.
2. Wählen Sie den Partner-Endpunkt.
   Im rechten Teil des Dialogs wird passend zum markierten Endpunkt ein möglicher Verbindungsweg angezeigt, soweit vorhanden. Unvollständige Wege wie z. B. bei einer Nicht spezifizierten CPU werden durch ein Ausrufezeichen auf rotem Grund gekennzeichnet.
3. Um die den Dialog zu beenden klicken Sie auf "OK".
   Um die projektierte Verbindung zu übernehmen und weitere Verbindungen zu anderen Endpunkten zu projektieren, klicken Sie auf "Übernehmen".

Arbeiten in der Netzsicht

Verbindungsweg und -partner in der Netzsicht hervorheben

Um die Verbindungspartner für alle oder für bestimmte Verbindungstypen in der Netzsicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".

   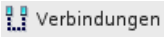

2. Wählen Sie im Kontextmenü des HMI-Gerätes, deren Verbindungspartner Sie in der Netzsicht anzeigen möchten, den Befehl "Verbindungspartner hervorheben".
3. Wählen Sie im Folgemenü "Alle Verbindungspartner".
   Das lokale Gerät und die CPUs der Zielgeräte sind markiert. Der lokale Verbindungspartner zeigt einen Pfeil nach rechts und die entfernten Verbindungspartner zeigen einen Pfeil nach links.
4. Öffnen Sie eine Liste mit Angabe der Zielgeräte, indem Sie den Pfeil des lokalen Gerätes anwählen. Diese Zusatzfunktion ist bei komplexen Netzkonfigurationen nützlich, bei denen nicht alle Geräte sichtbar sind.

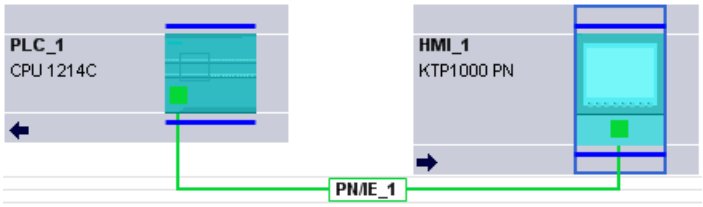

Hinweis

Sie können einen der Verbindungspartner, der im aktuellen Anzeigebereich der Netzsicht nicht sichtbar ist, anzeigen. Klicken Sie in der aufgeblendeten Liste auf den Kommunikationspartner. Ergebnis: Die Anzeige wird so verschoben, dass der Verbindungspartner sichtbar wird.

Siehe auch

Neue Verbindung grafisch anlegen (Seite 141)

Arbeiten mit der Verbindungstabelle

Grundfunktionen für Tabellen

Die Verbindungstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spaltenbreite verändern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen.

Spaltenbreite ändern

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Verbindungstabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form von zwei parallelen Linien annimmt (so, als wollten Sie die Breite der Spalte durch Ziehen mit dem Mauszeiger verändern).
2. Doppelklicken Sie auf diese Stelle.

oder

1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Tabelle.
2. Klicken Sie auf
   - "Spaltenbreite optimieren" oder
   - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

Spalten ein-/ausblenden

Über das Kontextmenü der Kopfzeilen der Verbindungstabelle können Sie die Anzeige der verschiedenen Tabellenspalten steuern. Über den Kontextmenüeintrag "Spalten ein-/ausblenden" sehen Sie eine Übersicht der verfügbaren Spalten. Mit den Optionskästchen steuern Sie die Sichtbarkeit der Spalten.

Mit Cursortasten in der Verbindungstabelle navigieren

Mit den Cursortasten NACH-OBEN und NACH-UNTEN können Sie eine Verbindung in der Verbindungstabelle auswählen; die ausgewählte Verbindung ist markiert und wird in der Netzsicht hervorgehoben dargestellt.

Eigenschaften der Verbindung ändern

Sie können die in der Verbindungstabelle angezeigten Parameter teilweise direkt bearbeiten. Um den Verbindungsnamen einer Verbindung zu ändern müssen Sie nicht zum Inspektorfenster navigieren.

Verbindungspartner ändern

Sie können den Verbindungspartner einer Verbindung wie folgt ändern:

1. Selektieren Sie die Verbindung.
2. Wählen Sie den neuen Verbindungspartner über die aktivierte Klappliste in der Spalte "Partner".

Verbindungen löschen

Sie können projektierte Verbindungen über die Netzsicht oder die Verbindungstabelle löschen.

In der Netzsicht können Sie jeweils eine hervorgehobene Verbindung löschen. In der Verbindungstabelle können Sie eine oder mehrere Verbindungen löschen.

Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie die zu löschende Verbindung aus:
   - In der Netzsicht: Markieren Sie die zu löschende Verbindung.
   - In der Verbindungstabelle: Markieren Sie die Zeilen der zu löschenden Verbindungen (Mehrfachauswahl möglich).
2. Öffnen Sie mit der rechten Maustaste das Kontextmenü.
3. Wählen Sie den Befehl "Löschen".

Ergebnis

Die markierte Verbindung wird vollständig gelöscht.

Kopieren von Verbindungen

Einführung

Verbindungen werden nicht einzeln kopiert, sondern immer im Kontext mit dem Projekt bzw. mit dem Gerät.

Sie können kopieren:

- gesamte Projekte
- eine oder mehrere Geräte innerhalb eines Projekts oder über Projektgrenzen hinweg

Projekt kopieren

Wenn Sie ein Projekt kopieren, dann werden sämtliche projektierten Verbindungen mitkopiert. Es sind keinerlei Einstellungen für die kopierten Verbindungen notwendig, da die Verbindungen konsistent bleiben.

Geräte kopieren

Wenn Sie Geräte kopieren, für die Verbindungen projektiert wurden (HMI-Geräte), dann werden die Verbindungen ebenfalls kopiert. Um den Verbindungsweg zu vervollständigen, müssen Sie noch die Vernetzung nachziehen.

Eine S7-1200 CPU mit einer V1.0 Firmware ist lediglich Server für HMI-Verbindungen und hat selber keine Verbindungsprojektierung. Daher werden beim Kopieren einer S7-1200 CPU mit einer V1.0 Firmware auch keine Verbindungen mitkopiert.

Inkonsistente Verbindungen - Verbindungen ohne Zuordnung

Bei einer inkonsistenten Verbindung ist die Struktur der Verbindungsdaten zerstört oder die Verbindung ist im Kontext mit dem Projekt nicht funktionsfähig.

Inkonsistente Verbindungen können nicht übersetzt und geladen werden - der Betrieb mit einer solchen Verbindung ist nicht möglich.

In der Verbindungstabelle sind inkonsistente Verbindungen durch die Farbe Rot zu erkennen.

Mögliche Ursachen für inkonsistente Verbindungen

- Löschen oder Änderungen der Hardware-Konfiguration
- Fehlende Vernetzungen von Schnittstellen im Projekt, die für eine Verbindung notwendig sind.
- Überschreitung von Verbindungsressourcen
- Fehler bei der Datensicherung durch unzureichenden Speicher
- Verbindungen zu einem unspezifizierten Verbindungspartner ohne Angabe der Partneradresse.

Detailinformationen über die Ursache der Inkonsistenz finden Sie im Register "Übersetzen" nach dem Übersetzen (Bearbeiten > Übersetzen).

Abhilfen

Wenn die Verbindung durch Öffnen der Verbindungseigenschaften, ändern bzw. rückgängig machen in der Projektierung nicht reparierbar ist, kann ein Löschen und Neuanlegen der Verbindung notwendig sein.

HMI-Verbindung Allgemeine Einstellungen

Allgemeine Verbindungsparameter

In der Parametergruppe "Allgemein" unter den Eigenschaften der Verbindung werden allgemeine Verbindungsparameter angezeigt, die den lokalen Verbindungsendpunkt identifizieren.

Sie haben hier außerdem die Möglichkeit, den Verbindungsweg zuzuordnen und die Verbindungspartner vollständig zu spezifizieren.

Besondere Verbindungseigenschaften

Anzeige der Verbindungseigenschaften (nicht änderbar):

- Aktiver Verbindungsaufbau
  Der Verbindungsaufbau geht immer vom HMI-Gerät aus. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.
- Einseitig
  Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.
- Betriebszustandsmeldungen senden
  Nicht relevant für HMI-Geräte.

Adressdetails

Anzeige der Adressdetails der HMI-Verbindung. Bei einem unspezifizierten Partner können die Werte für Baugruppenträger und Steckplatz geändert werden. Alle anderen Werte werden aus der aktuellen Konfiguration ermittelt und sind nicht änderbar.

Verschiedenes

Anzeige der Zugangspunkte für die Online-Verbindung zwischen HMI-Gerät und Verbindungspartner.

S7-Verbindungen projektieren

Einführung zum Projektieren von Verbindungen

Definition

Eine Verbindung definiert eine logische Zuordnung zweier Kommunikationspartner zur Ausführung von Kommunikationsdiensten. Eine Verbindung legt Folgendes fest:

- Beteiligte Kommunikationspartner
- Typ der Verbindung (z. B. S7-Verbindung)

- Spezielle Eigenschaften (z. B. ob eine Verbindung permanent aufgebaut bleibt oder ob sie im Anwenderprogramm dynamisch auf- und abgebaut wird und ob Betriebszustandsmeldungen gesendet werden sollen)
- Verbindungsweg

Wissenswertes zur Verbindungsprojektierung

Bei der Verbindungsprojektierung wird bei einer S7-Verbindung ein lokaler Verbindungsname als eindeutige lokale Kennung vergeben.

In der Netzsicht wird neben dem Register "Netzübersicht" ein Register "Verbindungen" angezeigt. In diesem Register befindet sich die Verbindungstabelle. Eine Zeile in dieser Verbindungstabelle repräsentiert eine projektierte Verbindung aus Sicht des lokalen Kommunikationspartners mit ihren Eigenschaften, z. B. zwischen zwei S7-1200 CPUs.

Wissenswertes zum Verbrauch von Verbindungsressourcen

Einführung

Jede Verbindung benötigt auf den beteiligten Geräten Verbindungsressourcen für den Endpunkt bzw. für den Übergangspunkt. Die Anzahl der Verbindungsressourcen ist gerätespezifisch.

Sind alle Verbindungsressourcen eines Kommunikationspartners belegt, so kann keine neue Verbindung aufgebaut werden. Sie erkennen das daran, dass eine neu angelegte Verbindung in der Verbindungstabelle rot hinterlegt ist. Die Konfiguration ist dann inkonsistent und kann nicht übersetzt werden.

S7-Verbindungen

Bei S7-Verbindungen über die PN-Schnittstelle wird bei der S7-1200 CPU pro S7-Verbindung eine Verbindungsressource für den Endpunkt belegt. Für den Verbindungspartner wird ebenfalls eine Verbindungsressource benötigt.

Sie finden eine Übersicht der verfügbaren und belegten Verbindungsressourcen bei selektierter S7-1200 CPU im Inspektorfenster unter "Eigenschaften > Verbindungsressourcen".

Ansichten mit Informationen zu den projektierten Verbindungen

Über die nachfolgend beschriebenen Ansichten haben Sie umfassenden Zugriff auf sämtliche Informationen und Funktionen zum Projektieren und Prüfen von Kommunikationsverbindungen.

- Verbindungsanzeige in der Netzsicht
- Verbindungstabelle
- Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

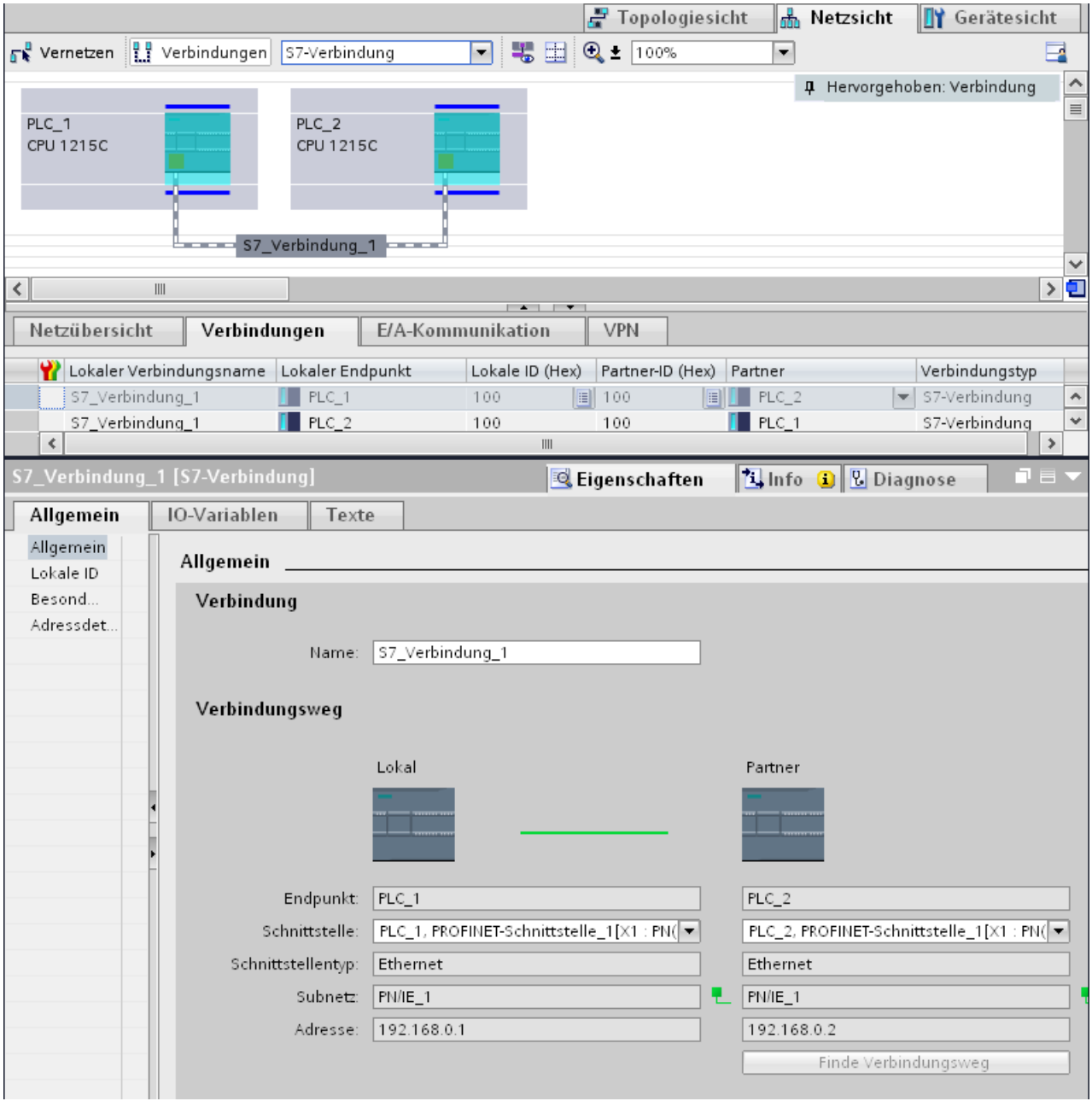

Vorteile

Die in diesen Ansichten dargestellten Informationen sind in Bezug auf augenblickliche Benutzeraktionen immer aktuell. Das bedeutet:

- Die Verbindungstabelle zeigt alle angelegten Verbindungen.
- Wenn Sie eine Verbindung in der Verbindungstabelle angewählt haben, gilt:
  - Bei aktivem Verbindungsmodus wird der Verbindungsweg in der Netzsicht grafisch hervorgehoben.
  - Das Register "Eigenschaften" im Inspektorfenster zeigt die Parameter dieser Verbindung.

Die Verbindungstabelle

Die Verbindungstabelle bietet folgende Funktionen:

- Auflistung aller Verbindungen im Projekt
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht (bei aktivem Verbindungsmodus)
- Ändern von Verbindungspartnern
- Anzeige von Zustandsinformationen

Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Der Eigenschaften-Dialog hat folgende Bedeutung:

- Anzeige für Verbindungsparameter
- Anzeige des Verbindungswegs
- Nachträgliches Spezifizieren von Verbindungen über die Schaltfläche "Finde Verbindungsweg"

Anlegen einer neuen Verbindung

Anlegen einer Verbindung - Alternativen

Sie haben folgende Möglichkeiten, eine Verbindung in der Netzsicht anzulegen:

- Grafische Verbindungsprojektierung
- Dialoggeführte Verbindungsprojektierung

Die einzelnen Schritte hierzu finden Sie in den Folgekapiteln.

Voraussetzung und Ergebnis

Sie haben in der Netzsicht die Geräte eingefügt, zwischen denen die Verbindungen projektiert werden sollen.

Verbindung spezifizieren

Wenn beide Partner für den gewählten Verbindungstyp am selben Netz vernetzt sind, legen Sie durch die grafische oder dialoggeführte Auswahl beider Verbindungspartner eine vollspezifizierte Verbindung an.

Diese wird automatisch in die Verbindungstabelle der S7-1200 CPU eingetragen. Für diese Verbindung wird ein lokaler Verbindungsname vergeben.

Die folgende Darstellung zeigt eine projektierte Verbindung bei vernetztem Gerät:

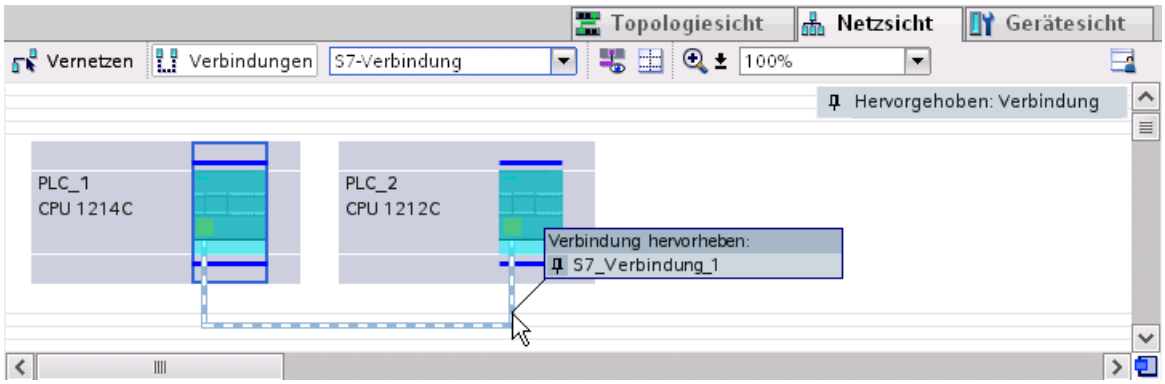

Neue Verbindung grafisch anlegen

Verbindungen grafisch projektieren

Bei der grafischen Verbindungsprojektierung wird der Verbindungsweg, sofern Schnittstellen und Ressourcen vorhanden sind, automatisch bestimmt. Wählen Sie in der aktuellen Konfiguration die zu verbindenden Geräte aus.

Verbindungsweg automatisch ermitteln

Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".

   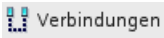

   Damit ist der Verbindungsmodus aktiviert. Sie können jetzt den gewünschten Verbindungstyp auswählen. Sie erkennen dies an folgendem Verhalten:
   In der Netzsicht sind die für den gewählten Verbindungstyp in Ihrem Projekt verwendbaren Geräte farblich hervorgehoben.

2. Ziehen Sie den Mauscursor bei gedrückter Maustaste von dem Gerät, von dem eine Verbindung ausgehen soll, auf das Gerät, bei dem die Verbindung enden soll.

   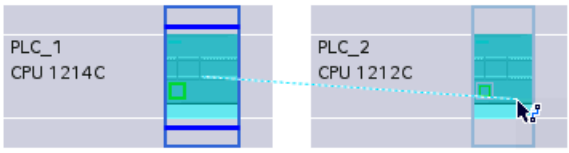

3. Lassen die auf dem Zielgerät die Maustaste los, um die Verbindung zwischen den beiden Geräten zu erstellen.

Ergebnis

- Eine spezifizierte Verbindung ist angelegt.
- Der Verbindungsweg ist hervorgehoben dargestellt.
- Die Verbindung ist in der Verbindungstabelle eingetragen.

Verbindung bei fehlender oder nicht eindeutiger Netzzuordnung projektieren

Fehlende Vernetzungen werden soweit möglich beim Anlegen einer Verbindung automatisch angelegt. Falls eine eindeutige Netzzuordnung nicht möglich ist, erfolgt eine Abfrage beim Abschluss der Verbindungsprojektierung. Hierbei werden die bestehenden Subnetze zur Auswahl angeboten.

Beispiel in folgendem Bild: Beim Anlegen einer Verbindung zwischen den noch nicht vernetzten Stationen PLC_1 und PLC_2 erfolgt eine Abfrage.

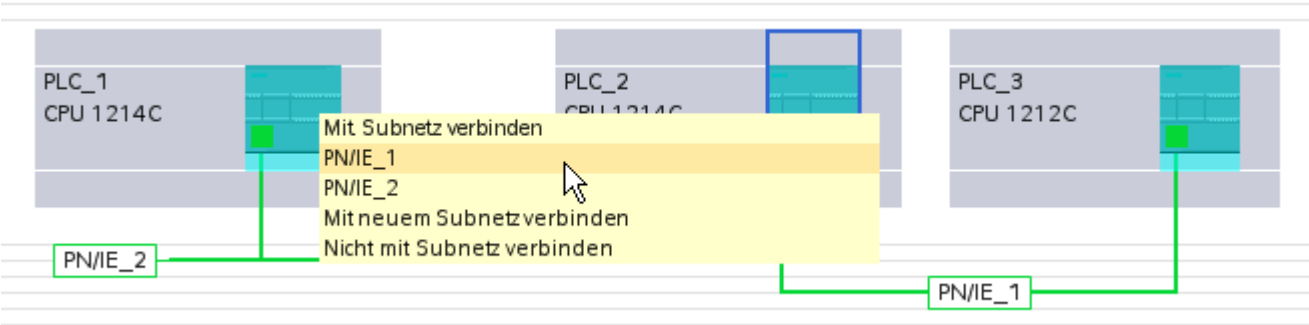

Neue Verbindung dialoggeführt anlegen

Verbindungen dialoggeführt projektieren

Bestimmen Sie das lokale Gerät und seinen Verbindungspartner.

Vorgehen

Um eine Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü eines Verbindungspartners, für den eine Verbindung angelegt werden soll, den Befehl "Neue Verbindung hinzufügen".
   Der Dialog "Neue Verbindung erstellen" wird geöffnet.
2. Wählen Sie den Partner-Endpunkt.
   Im rechten Teil des Dialogs wird passend zum markierten Endpunkt ein möglicher Verbindungsweg angezeigt, soweit vorhanden. Unvollständige Wege wie z. B. bei einer Nicht spezifizierten CPU werden durch ein Ausrufezeichen auf rotem Grund gekennzeichnet.
3. Um die projektierte Verbindung zu übernehmen und weitere Verbindungen zu anderen Endpunkten zu projektieren, klicken Sie auf "Hinzufügen".
   Um die den Dialog zu beenden klicken Sie auf "OK".

Arbeiten in der Netzsicht

Verbindungsweg und -partner in der Netzsicht hervorheben

Um die Verbindungspartner für alle oder für bestimmte Verbindungstypen in der Netzsicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".

   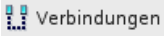

2. Wählen Sie im Kontextmenü der S7-CPU, deren Verbindungspartner Sie in der Netzsicht anzeigen möchten, den Befehl "Verbindungspartner hervorheben".
3. Wählen Sie im Folgemenü "Alle Verbindungspartner".
   Das lokale Gerät und die CPUs der Zielgeräte sind markiert. Der lokale Verbindungspartner zeigt einen Pfeil nach rechts und die entfernten Verbindungspartner zeigen einen Pfeil nach links.
4. Öffnen Sie eine Liste mit Angabe der Zielgeräte, indem Sie den Pfeil des lokalen Gerätes anwählen. Diese Zusatzfunktion ist bei komplexen Netzkonfigurationen nützlich, bei denen nicht alle Geräte sichtbar sind.

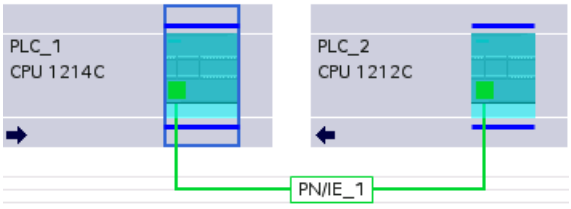

Hinweis

Sie können einen der Verbindungspartner, der im aktuellen Anzeigebereich der Netzsicht nicht sichtbar ist, anzeigen. Klicken Sie in der aufgeblendeten Liste auf den Kommunikationspartner. Ergebnis: Die Anzeige wird so verschoben, dass der Verbindungspartner sichtbar wird.

Arbeiten mit der Verbindungstabelle

Grundfunktionen für Tabellen

Die Verbindungstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spaltenbreite verändern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen.

Spaltenbreite ändern

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Verbindungstabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form von zwei parallelen Linien annimmt (so, als wollten Sie die Breite der Spalte durch Ziehen mit dem Mauszeiger verändern).
2. Doppelklicken Sie auf diese Stelle.

oder

1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Tabelle.
2. Klicken Sie auf
   - "Spaltenbreite optimieren" oder
   - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

Spalten ein-/ausblenden

Über das Kontextmenü der Kopfzeilen der Verbindungstabelle können Sie die Anzeige der verschiedenen Tabellenspalten steuern. Über den Kontextmenüeintrag "Spalten ein-/ausblenden" sehen Sie eine Übersicht der verfügbaren Spalten. Mit den Optionskästchen steuern Sie die Sichtbarkeit der Spalten.

Wenn Sie die Anordnung, Breite und Sichtbarkeit der Tabellenspalten speichern möchten, klicken Sie in der Netzsicht rechts oben auf die Funktion "Anordnung merken".

Symbol	Bedeutung
	Anordnung merken Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.

Mit Cursortasten in der Verbindungstabelle navigieren

Mit den Cursortasten NACH-OBEN und NACH-UNTEN können Sie eine Verbindung in der Verbindungstabelle auswählen; die ausgewählte Verbindung ist markiert und wird in der Netzsicht hervorgehoben dargestellt.

Eigenschaften der Verbindung ändern

Sie können einige der in der Verbindungstabelle angezeigten Parameter direkt bearbeiten. Der Name der Verbindung beispielsweise nur in der Verbindungstabelle geändert werden.

Verbindungspartner ändern

Sie können den Verbindungspartner einer Verbindung wie folgt ändern:

1. Selektieren Sie die Verbindung.
2. Wählen Sie den neuen Verbindungspartner über die aktivierte Klappliste in der Spalte "Partner".

Verbindungen löschen

Sie können projektierte Verbindungen über die Netzsicht oder die Verbindungstabelle löschen.

In der Netzsicht können Sie jeweils eine hervorgehobene Verbindung löschen. In der Verbindungstabelle können Sie eine oder mehrere Verbindungen löschen.

Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie die zu löschende Verbindung aus:
   - In der Netzsicht: Markieren Sie die zu löschende Verbindung.
   - In der Verbindungstabelle: Markieren Sie die Zeilen der zu löschenden Verbindungen (Mehrfachauswahl möglich).
2. Öffnen Sie mit der rechten Maustaste das Kontextmenü.
3. Wählen Sie den Befehl "Löschen".

Ergebnis

Die markierte Verbindung wird vollständig gelöscht.

Kopieren von Verbindungen

Einführung

Verbindungen werden nicht einzeln kopiert, sondern immer im Kontext mit dem Projekt bzw. mit dem Gerät.

Sie können kopieren:

- gesamte Projekte
- eine oder mehrere Geräte innerhalb eines Projekts oder über Projektgrenzen hinweg

Projekt kopieren

Wenn Sie ein Projekt kopieren, dann werden sämtliche projektierten Verbindungen mitkopiert. Es sind keinerlei Einstellungen für die kopierten Verbindungen notwendig, da die Verbindungen konsistent bleiben.

Geräte kopieren

Wenn Sie Geräte kopieren, für die Verbindungen projektiert wurden, dann werden die Verbindungen ebenfalls kopiert. Um den Verbindungsweg zu vervollständigen, müssen Sie noch die Vernetzung nachziehen.

Eine S7-1200 CPU mit einer V1.0 Firmware ist lediglich Server für Verbindungen und hat selber keine Verbindungsprojektierung. Daher werden beim Kopieren einer S7-1200 CPU mit einer V1.0 Firmware auch keine Verbindungen mitkopiert.

Inkonsistente Verbindungen - Verbindungen ohne Zuordnung

Bei einer inkonsistenten Verbindung ist die Struktur der Verbindungsdaten zerstört oder die Verbindung ist im Kontext mit dem Projekt nicht funktionsfähig.

Inkonsistente Verbindungen können nicht übersetzt und geladen werden - der Betrieb mit einer solchen Verbindung ist nicht möglich.

In der Verbindungstabelle sind inkonsistente Verbindungen durch die Farbe Rot zu erkennen.

Mögliche Ursachen für inkonsistente Verbindungen

- Löschen oder Änderungen der Hardware-Konfiguration
- Fehlende Vernetzungen von Schnittstellen im Projekt, die für eine Verbindung notwendig sind.
- Überschreitung von Verbindungsressourcen
- Verbindungen zu einem unspezifizierten Verbindungspartner ohne Angabe der Partneradresse.

Detailinformationen über die Ursache der Inkonsistenz finden Sie im Register "Übersetzen" nach dem Übersetzen (Bearbeiten > Übersetzen).

Abhilfen

Um einem bestehenden offenen Verbindungsweg einen geschlossenen Verbindungsweg zuzuweisen, ergänzen Sie die Gerätekonfiguration so, dass die für den Verbindungstyp benötigten Schnittstellen bei beiden Partnern vorhanden sind. Im Inspektorfenster unter "Eigenschaften > Allgemein > Schnittstelle" kann mit der Schaltfläche "Finde Verbindungsweg" eine Verbindung zu einem vorhandenem Partner erstellt werden.

S7-Verbindung Allgemeine Einstellungen

Allgemeine Verbindungsparameter

In der Parametergruppe "Allgemein" unter den Eigenschaften der Verbindung werden allgemeine Verbindungsparameter angezeigt, die den lokalen Verbindungsendpunkt identifizieren.

Sie haben hier die Möglichkeit, den Verbindungsweg zuzuordnen und die Verbindungspartner vollständig zu spezifizieren.

Lokale ID

Hier wird die lokale ID der Baugruppe angezeigt, von der aus die Verbindung betrachtet wird (lokaler Partner). Sie können die lokale ID ändern. Das ist dann notwendig, wenn Sie bereits Kommunikations-Funktionsbausteine programmiert haben und Sie die dort angegebene Lokale ID für die Verbindung verwenden wollen.

Besondere Verbindungseigenschaften

Anzeige der Verbindungseigenschaften (in Abhängigkeit der verwendeten Komponenten änderbar):

- Einseitig
  Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.
- Aktiver Verbindungsaufbau
  Bei einer einseitigen Verbindung, z. B. mit einer S7-1200 CPU des Firmewarestands V1.0, kann nur ein Verbindungspartner für den aktiven Verbindungsaufbau zur Verfügung stehen. Bei einer zweiseitigen Verbindung können Sie einstellen, welcher Verbindungspartner den aktiven Part übernehmen soll.
- Betriebszustandsmeldungen senden
  Zeigt an, ob der lokale Partner Betriebszustandsmeldungen an den Verbindungspartner sendet.

Adressdetails

Anzeige der Adressdetails der S7-Verbindung. Bei einem unspezifizierten Partner können die Werte für Baugruppenträger und Steckplatz geändert werden. Alle anderen Werte werden aus der aktuellen Konfiguration ermittelt und sind nicht änderbar.

S7-Verbindung Adressdetails

Bedeutung

In den Adressdetails sehen Sie die Endpunkte der Verbindung und können diese über die Angabe von Baugruppenträger und Steckplatz lokalisieren.

Wenn eine Verbindung aufgebaut wird, dann werden verbindungsspezifische Ressourcen einer Baugruppe dieser Verbindung fest zugeordnet. Diese Zuordnung setzt voraus, daß die Verbindungsressource adressiert werden kann. Der TSAP (Transport Service Access Point) ist quasi die Adresse der Ressource, die unter anderem mit Hilfe der Verbindungsressource oder bei S7-1200 CPUs ab Firmware V2.0 mit der SIMATIC-ACC (SIMATIC Application Controlled Communication) gebildet wird.

Aufbau des TSAPs für S7-1200

- Für S7-1200 CPU mit Firmware ab V2.0:
  "SIMATIC-ACC"<nnn><mm>
  nnn = lokale ID
  mm = freier Wert
- Für S7-1200 CPU mit Firmware V1.0:
  <xx>.<yz>
  xx = Nummer der Verbindungsressource
  y = Baugruppenträgernummer
  z = Steckplatznummer

TSAP-Aufbau in Abhängigkeit des Partners

Der Aufbau des TSAPs ist bei S7-1200 CPUs abhängig von der jeweiligen Firmware und von dem fernen Verbindungspartner. Wenn eine S7-1200 CPU mit einer S7-300/400 CPU verbunden wird, verwendet auch eine S7-1200 CPU den TSAP-Aufbau mit Einbeziehung der Verbindungsressource.

Siehe folgende Beispiele für TSAPs verschiedener Verbindungsprojektierungen

- Verbindung zwischen zwei S7-1200 CPUs (beide Firmware V2.0):
  - S7-1200 CPU "A" mit Firmware V2.0 und lokaler ID 100:
    TSAP: SIMATIC-ACC10001
  - S7-1200 CPU "B" mit Firmware V2.0 und lokaler ID 5AE:
    TSAP: SIMATIC-ACC5AE01
- Verbindung zwischen zwei S7-1200 CPUs (Firmware V2.0 und V1.0):
  - S7-1200 CPU mit Firmware V2.0 und lokaler ID 1FF:
    TSAP: SIMATIC-ACC1FF01
  - S7-1200 CPU mit Firmware V1.0 (Baugruppenträger 0, Steckplatz 1, Verbindungsressource 03):
    TSAP: 03.01
- Verbindung zwischen S7-1200 CPU (Firmware V2.0) und S7-300/400 CPU:
  - S7-1200 CPU mit Firmware V2.0 (Baugruppenträger 0, Steckplatz 1, Verbindungsressource 12):
    TSAP: 12.01
  - S7-300/400 CPU (Baugruppenträger 0, Steckplatz 2, Verbindungsressource 11):
    TSAP: 11.02

S7-Verbindungen über CM/CP

Einleitung

S7-1200 CPUs mit einem Firmwarestand ab V2.0 unterstützen einseitige und zweiseitige S7-Verbindungen über CM/CP-Schnittstellen. Dadurch wird die Anzahl der für S7-Verbindungen nutzbaren Ethernet-Ports und Netze erhöht. Obwohl die Verbindung dann über den CM/CP geführt wird, ist die zugehörige S7-1200 CPU ein Endpunkt der Verbindung. Der andere Endpunkt kann bei zweiseitigen Verbindungen ein beliebiges anderes Gerät sein. Dieses andere Gerät muss auch S7-Verbindungen unterstützen.

Datenvolumen und Mengengerüst

Die Anzahl Kommunikationsverbindungen, die über CM/CP unterstützt werden, entnehmen Sie bitte dem Gerätehandbuch, das jedem CM/CP beiliegt. Durch Hinzunahme weiterer CM/CPs kann die Anzahl der Verbindungen pro Gerät weiter erhöht werden.

Sind in einem Gerät mehrere CM/CPs eingebaut, so wird bei Überschreitung dieser Grenze automatisch auf den nächsten CP umgeschaltet. Bei Bedarf ordnen Sie Verbindungen über die Wegewahl gezielt zu.

Hinweis

Der Datentransfer > 240 Byte wird von den aktuellen CPs unterstützt.

CPs mit älterem Ausgabestand unterstützen den Datentransfer mit einer Datenlänge von bis zu 240 Byte.

Beachten Sie hierzu bitte die Angaben im Gerätehandbuch des Ethernet-CP.

Aufgaben des Ethernet-CM/CP im Online-Betrieb

Der Ethernet-CM/CP übernimmt beim Datentransfer über eine Verbindung folgende Aufgaben:

- Empfangen
  Empfangen von Daten vom Ethernet und Weitergabe an den Anwender-Datenbereich in der CPU.
- Senden
  Übernehmen von Daten aus dem Anwender-Datenbereich der CPU und Senden der Daten über Ethernet.

Der Verbindungsaufbau erfolgt automatisch, sobald der Partner erreichbar ist.

S7-Verbindungen über PUT- und GET-Anweisungen

Grundlagen zur Kommunikation über die PUT/GET-Anweisung

Grundlagen zu den PUT/GET-Anweisungen

Um Daten zwischen zwei CPUs über eine S7-Verbindung auszutauschen, verwenden Sie die Anweisungen PUT und GET.

Mit Hilfe der Anweisung GET können Sie Daten aus einer Partner-CPU auslesen. Mit der Anweisung PUT steuern Sie über das Anwenderprogramm das Schreiben von Variablen beim Kommunikationspartner. Neben den Anweisungen PUT und GET sind keine zusätzlichen Kommunikationsfunktionen für das Lesen und Schreiben von Variablen vorgesehen.

Um Ihnen die Verwendung der beiden Anweisungen zu erleichtern, legen Sie alle notwendigen Parameter für die Verbindung sowie alle Bausteinparameter im Inspektorfenster des Programmiereditors fest.

Voraussetzung

Um die Anweisungen PUT und GET verwenden zu können, müssen die folgenden Voraussetzungen erfüllt sein:

- Mindestens eine CPU der S7-1200/1500 oder S7-300/400 muss im Projekt angelegt sein. Auf einer S7-1200 CPU muss die Firmware 2.0 oder höher installiert sein. Wenn Sie noch keine zweite CPU im Projekt angelegt haben, können Sie die Verbindung zunächst zu einem unspezifizierten Partner aufbauen.
- Es muss eine S7-Verbindung zwischen beiden CPUs bestehen. Wenn Sie noch keine Verbindung zwischen zwei CPUs hergestellt haben, wird automatisch eine Verbindung während der Projektierung der Anweisungen erstellt.
- Für beide Anweisungen ist ein Instanz-Datenbaustein nötig, in dem alle Daten hinterlegt werden, die von der Anweisung genutzt werden. Der Instanz-Datenbaustein wird automatisch angelegt, sobald Sie die Anweisung PUT oder GET auf ein Netzwerk im Programmiereditor ziehen. Die Instanz-Datenbausteine dürfen für den korrekten Programmablauf nicht verändert werden und sind daher know-how-geschützt. Sie haben nur lesenden Zugriff auf die Instanz-Datenbausteine.

Siehe auch

Übersicht über die Verbindungsparametrierung (Seite 161)

Auftragsstart parametrieren (Seite 167)

PUT: Schreib- und Sendebereich parametrieren (Seite 167)

GET: Lese- und Speicherbereich parametrieren (Seite 168)

Verbindungsparametrierung

Übersicht über die Verbindungsparametrierung

Einführung

Die Verbindungsparametrierung für die Anweisungen PUT und GET erfolgt im Inspektorfenster des Programmiereditors. Alle Parameter werden im zugehörigen Instanz-Datenbaustein gespeichert.

Aufbau der Verbindungsparametrierung

Die Verbindungsparametrierung besteht aus folgenden Komponenten:

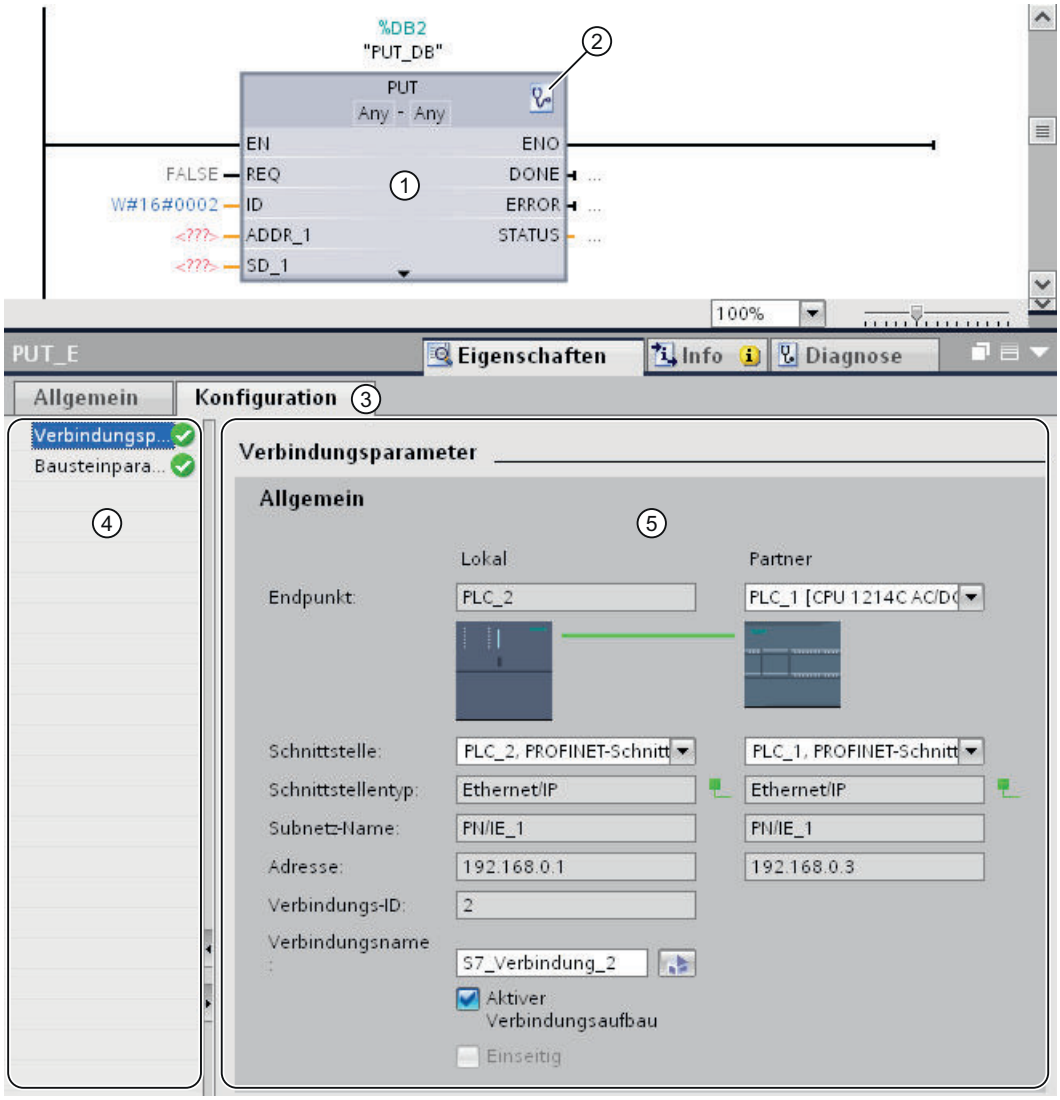

① Kommunikationsanweisung für PUT oder GET
② Aufruf von Online- und Diagnosefunktionen
③ Register "Konfiguration" im Register "Eigenschaften"
④ Bereichsnavigation des Registers "Konfiguration"
⑤ Allgemeine Eigenschaften der Verbindungsparameter

Anzeigen von Online- und Diagnosefunktionen

Wenn Sie auf das Symbol zum Start der Online- und Diagnosefunktionen klicken, wird die zugehörige CPU automatisch online verbunden. Die Verbindungstabelle in der Netzsicht wird geöffnet. Außerdem werden im Inspektorfenster das Register "Diagnose" und die Verbindungsinformationen angezeigt.

Eintragen der Verbindungsparameter

Tragen Sie die gewünschten Verbindungsparameter im Register "Konfiguration" ein. Sie finden in der Bereichsnavigation des Registers "Konfiguration" die Gruppe "Verbindungsparameter". Diese Gruppe enthält die Verbindungsparametrierung. Hier können Sie systemunterstützt die Parameter für die Verbindungen eintragen. Wenn alle erforderlichen Parameter vorhanden sind, wird in der Bereichsnavigation hinter die Gruppe "Verbindungsparameter" ein Häkchen gesetzt.

Siehe auch

Auftragsstart parametrieren (Seite 167)

PUT: Schreib- und Sendebereich parametrieren (Seite 167)

GET: Lese- und Speicherbereich parametrieren (Seite 168)

Beschreibung der Verbindungsparameter

Überblick

Die folgende Tabelle zeigt die allgemeinen Verbindungsparameter:

Parameter	Beschreibung
Endpunkt	Die Namen des lokalen Endpunkts und des Partner Endpunkts wird angezeigt. • Lokaler Endpunkt Der lokale Endpunkt ist die CPU, in der Sie die PUT/GET-Anweisung programmieren. • Partner-Endpunkt Der Partner-Endpunkt wird über die Klappliste ausgewählt. In der Klappliste werden alle verfügbaren möglichen Verbindungspartner angezeigt, darunter auch unspezifizierte Verbindungspartner für Geräte, deren Daten innerhalb des Projekts nicht bekannt sind. Solange noch kein Verbindungspartner eingestellt ist, bleiben alle weiteren Parameter in der Maske deaktiviert.
Schnittstelle	Die Schnittstelle der Partner-CPU wird angezeigt. Die Anzeige der Partner-Schnittstelle erfolgt erst nach Auswahl einer spezifizierten Partner-CPU.
Schnittstellentyp	Die Art der Schnittstelle, über die die Kommunikation abgewickelt wird, wird angezeigt.
Subnetz-Name	Das Subnetz des lokalen Endpunkts wird angezeigt, wenn dieses vorhanden ist. Die Anzeige des Partner-Subnetzes erfolgt erst nach Auswahl des Partner-Endpunkts. Wenn mindestens einer der beiden Verbindungspartner nicht mit einem Subnetz verbunden ist, werden die beiden Verbindungspartner automatisch miteinander vernetzt. Dabei wird der nicht vernetzte Partner an dasselbe Subnetz angeschlossen, über das der andere Partner bereits mit einem Netz verbunden ist. Eine Verbindung von Verbindungspartnern an unterschiedlichen Subnetzen ist nur über IP- oder S7-Routing möglich. Die IP-Routing-Einstellungen können über die Eigenschaften der betroffenen Ethernet-Schnittstellen bearbeitet werden.
Adresse	Die IP-Adresse des lokalen Endpunkts wird angezeigt. Die Anzeige der IP-Adresse des Partners erfolgt erst nach Auswahl des Partner Endpunkts. Wenn Sie einen unspezifizierten Verbindungspartner ausgewählt haben, ist das Eingabefeld leer und rot hinterlegt. Sie müssen in dem Fall eine gültige IP-Adresse für den Verbindungspartner angeben.

Parameter	Beschreibung
Verbindungs-ID	Die aktuell eingestellte Verbindungs-ID wird angezeigt. Sie können die Verbindungs-ID in der Verbindungstabelle in der Netzsicht ändern. Außerdem können Sie während der Verbindungsparametrierung direkt auf die Verbindungstabelle zugreifen. Klicken Sie dazu auf das Symbol "Neue Verbindung erstellen".
Verbindungsname	Der Name der Verbindung, welche beim Einfügen der PUT/GET-Anweisung automatisch angelegt wurde, wird angezeigt. Sie können den Namen der Verbindung ändern, indem Sie einen anderen Namen in das Feld eingeben. Sie können außerdem eine neue Verbindung erstellen oder bereits vorhandene Verbindungen bearbeiten, indem Sie auf das Symbol "Neue Verbindung erstellen" klicken.
Aktiver Verbindungsaufbau	Bestimmen Sie über das Optionsfeld "Aktiver Verbindungsaufbau", welcher Partner die Kommunikation startet. Beim Anlegen der Verbindung wird automatisch zunächst der lokale Partner für den Verbindungsaufbau bestimmt. Wenn ein Gerät den aktiven Verbindungsaufbau nicht unterstützt, müssen Sie beim anderen Partner den aktiven Verbindungsaufbau aktivieren.
Einseitig	Ist das Optionskästchen aktiviert, ist der Verbindungspartner Server für diese Verbindung. Er kann nicht aktiv senden oder empfangen. Dies entspricht dem Verhalten der PUT/GET-Anweisungen. Andere Anweisungen sind in diesem Fall nicht möglich. Wenn das Optionskästchen deaktiviert ist, dann ist die Verwendung anderer Anweisungen zur Kommunikation ebenfalls möglich.

Verbindungsparametrierung starten

Die Verbindungsparametrierung für PUT und GET können Sie im Inspektorfenster durchführen, sobald Sie eine PUT/GET-Anweisung in einen Programmbaustein eingefügt und selektiert haben.

Vorgehen

Um PUT/GET-Anweisungen einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen" den Ordner "Kommunikation > S7-Kommunikation".
2. Ziehen Sie eine PUT/GET-Anweisung per Drag & Drop in ein Netzwerk.
   Der Dialog "Aufrufoptionen" wird geöffnet.
3. Optional: Bearbeiten Sie im Dialog "Aufrufoptionen" die Eigenschaften des Instanz-DBs. Sie haben dabei folgende Möglichkeiten:
   – Ändern Sie den vorgegebenen Namen.
   – Wählen Sie das Optionsfeld "manuell", um eine eigene Nummer zu vergeben.
4. Klicken Sie auf "OK".

Ergebnis

Für die eingefügte Anweisung PUT oder GET wird ein entsprechender Instanz-Datenbaustein angelegt. Bei S7-300 CPUs wird zusätzlich ein Funktionsbaustein in den Programmressourcen angelegt.

Bei einer selektierten PUT/GET-Anweisung sehen Sie im Inspektorfenster unter "Eigenschaften" das Register "Konfiguration". Die Gruppe "Verbindungsparameter" in der Bereichsnavigation enthält die Verbindungsparametrierung, die Sie nun durchführen können.

Siehe auch

Verbindungen anlegen und parametrieren (Seite 165)

Verbindungen löschen (Seite 166)

Verbindungen anlegen und parametrieren

In der Verbindungsparametrierung der PUT/GET-Anweisungen können Sie S7-Verbindungen anlegen und parametrieren. Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft.

Voraussetzung

Es existiert eine CPU mit einer Kommunikationsanweisung PUT oder GET.

Vorgehen

Um eine S7-Verbindung über PUT/GET-Anweisungen zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Programmiereditor den Aufruf der Anweisung PUT oder GET.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, ist nur die leere Klappliste für den Partner-Endpunkt aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.
   Es werden die bereits bekannten Verbindungsparameter angezeigt:
   - Name des lokalen Endpunkts
   - Schnittstelle des lokalen Endpunkts
   - IP-Adresse des lokalen Endpunkts

4. Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage.
   Die folgenden Parameter werden automatisch eingetragen, sobald Sie den Verbindungspartner gewählt haben:
   - Name des Partner-Endpunkts
   - Schnittstelle des Partner-Endpunkts. Wenn mehrere Schnittstellen zur Verfügung stehen, dann können Sie die Schnittstelle bei Bedarf ändern.
   - Schnittstellentyp des Partner-Endpunkts
   - Subnetz-Name beider Endpunkte
   - IP-Adresse des Partner-Endpunkts
   - Name der Verbindung, die für die Kommunikation genutzt wird. Ist noch keine Verbindung vorhanden, wird automatisch eine erstellt.
5. Ändern Sie bei Bedarf den Verbindungsnamen im Eingabefeld "Verbindungsname" ab. Wenn Sie eine neue Verbindung erstellen, oder eine vorhandene Verbindung bearbeiten möchten, klicken Sie auf das Symbol "Neue Verbindung erstellen".

Hinweis

Die Anweisungen PUT und GET zwischen zwei Kommunikationspartnern sind erst dann lauffähig, wenn sowohl die Hardware-Konfiguration wie auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurden. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

Verbindungen löschen

Eine Verbindung, die beim Einfügen einer PUT/GET-Anweisung automatisch erstellt wurde, taucht wie jede gewöhnliche Verbindung in der Verbindungstabelle der Netzsicht auf. Sie lässt sich daher in der Verbindungstabelle löschen.

Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Verbindungstabelle in der Netzsicht.
2. Selektieren Sie die Verbindung, die Sie löschen möchten, in der Verbindungstabelle.
3. Klicken Sie mit der rechten Maustaste auf die Verbindung und wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die Verbindung ist gelöscht. Die PUT/GET-Anweisung sowie die zugehörigen Instanzdatenbausteine bleiben erhalten und müssen bei Bedarf manuell gelöscht werden.

Um die PUT/GET-Anweisung weiter zu verwenden, müssen Sie die Verbindungsprojektierung im Inspektorfenster des Programmiereditors erneut durchführen, da beim Löschen der

Verbindung alle Verbindungsparameter ebenfalls gelöscht werden. Legen Sie in diesem Fall einen neuen Kommunikationspartner und eine passende Verbindung fest.

Bausteinparametrierung

Auftragsstart parametrieren

Um eine Kommunikation über die PUT/GET-Anweisung zu beginnen, müssen Sie ein Ereignis festlegen, welches die Anweisung aktiviert. Dieses Ereignis wird als Steuerparameter (REQ) bezeichnet. Die Aktivierung des Kommunikationsauftrags erfolgt, sobald am Steuerparameter REQ eine positive Flanke anliegt.

Beachten Sie, dass der Steuerparameter REQ beim Erstaufruf mit FALSE belegt ist.

Voraussetzung

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine PUT/GET-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

Vorgehen

Um den Steuerparameter REQ festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PUT/GET-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen sie im Feld "REQ" eine Variable des Datentyps "BOOL" aus, um die Ausführung der Anweisung zu initialisieren. Sie können alternativ auch eine vorhergehende Anweisung im Programmiereditor verschalten.

Siehe auch

PUT: Schreib- und Sendebereich parametrieren (Seite 167)

GET: Lese- und Speicherbereich parametrieren (Seite 168)

PUT: Schreib- und Sendebereich parametrieren

Sie müssen bei der Kommunikation über die PUT-Anweisung festlegen, in welchen Speicherbereich der Partner-CPU die Daten geschrieben werden sollen. Außerdem müssen Sie den Speicherbereich der lokalen CPU festlegen, aus dem die Daten gelesen werden sollen.

Voraussetzung

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine PUT-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

Vorgehen

Um den Lese- und Speicherbereich für die Anweisung festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PUT-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen sie im Feld "Ein-Ausgänge > Schreibbereich (ADDR_1) > Start" einen Zeiger mit dem Datentyp "REMOTE" auf den zu beschreibenden Bereich der Partner-CPU.
   Es ist nur absolute Adressierung zulässig.
   Beispiel: P#DB10.DBX5.0 Byte 10
5. Geben Sie im Feld "Länge" die Länge des Schreibbereichs ein und wählen Sie aus der Klappliste den Datentyp des Speicherbereichs aus.
6. Wählen Sie im Feld "Ein-Ausgänge > Sendebereich (SD_1) > Start" einen Zeiger auf den Bereich in der lokalen CPU, der die zu sendenden Daten enthält.
7. Geben Sie die Länge des zu lesenden Speicherbereichs im Feld Länge ein und wählen Sie den Datentyp aus der Klappliste.
   Zulässig sind nur die Datentypen BOOL (bei einem Bitfeld muss als Adresse "0" und als Länge ein ganzzahlig Vielfaches von Byte verwendet werden), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.
   Wenn der VARIANT-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Siehe auch

GET: Lese- und Speicherbereich parametrieren (Seite 168)

GET: Lese- und Speicherbereich parametrieren

Sie müssen bei der Kommunikation über die GET-Anweisung festlegen, in welchen Speicherbereich der lokalen CPU die Daten geschrieben werden sollen. Außerdem müssen Sie den Lesebereich der Partner-CPU festlegen, aus dem die Daten gelesen werden sollen.

Voraussetzung

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine GET-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

Vorgehen

Um den Lese- und Speicherbereich für die Anweisung festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die GET-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen sie im Feld "Ein-Ausgänge > Lesebereich (ADDR_1) > Start" einen Zeiger mit dem Datentyp "REMOTE" auf den zu lesenden Bereich der Partner-CPU.
   Es ist nur absolute Adressierung zulässig.
   Beispiel: P#DB10.DBX5.0 Byte 10
5. Geben Sie im Feld "Länge" die Länge des Lesebereichs ein und wählen Sie aus der Klappliste den Datentyp des Speicherbereichs aus.
6. Wählen Sie im Feld "Ein-Ausgänge > Speicherbereich (RD_1) > Start" einen Zeiger auf den Bereich in der lokalen CPU, in dem die gelesenen Daten abgelegt werden.
7. Geben Sie die Länge des Speicherbereichs im Feld Länge ein und wählen Sie den Datentyp aus der Klappliste.
   Zulässig sind nur die Datentypen BOOL (bei einem Bitfeld muss als Adresse "0" und als Länge ein ganzzahlig Vielfaches von Byte verwendet werden), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.

Siehe auch

PUT: Schreib- und Sendebereich parametrieren (Seite 167)

Verbindungsressourcen

Mengengerüst in der Verbindungsprojektierung

Verbindungsressourcen

Einige Kommunikationsdienste benötigen Verbindungen. Kommunikationsverbindungen beanspruchen im Betrieb Speicher- und Programmressourcen der im Kommunikationsweg beteiligten CPUs, CPs und CMs (z. B. Speicherbereiche im Betriebssystem der CPU). Dies betrifft beispielsweise innerhalb einer Steuerung (PLC) die CPU als Verbindungsendpunkt sowie die Erweiterungen mit CPs als Schnittstellen zu den Subnetzen. In den meisten Fällen wird für eine Verbindung eine Ressource pro CPU/CP/CM belegt. In der HMI-Kommunikation werden pro HMI-Verbindung bis zu 3 Verbindungsressourcen benötigt.

Verfügbare Verbindungsressourcen

Das maximale Mengengerüst innerhalb des Automatisierungssystems wird immer durch die CPU bestimmt, entweder durch die von der CPU bestimmten Systemobergrenze (S7-1200/1500) oder durch die Summe der nutzbaren Ressourcen aus CPU und CP (S7-300/400). Die verfügbaren Verbindungsressourcen einer CPU/CP/CM werden in den meisten Fällen automatisch nach Bedarf auf ihre Schnittstellen verteilt.

Jede CPU bringt reservierte Verbindungsressourcen für PG-, HMI- und Webserver-Kommunikation in ihrer Projektierung mit. Daneben gibt es verfügbare Ressourcen, die für jede HMI-, S7-Kommunikation und offene Kommunikation (S7-1200/1500) genutzt werden können.

Welche Ressourcen Ihnen zur Verfügung stehen, können Sie über die Konfiguration der Geräte wie folgt bestimmen:

- Wahl des Gerätetyps beim Verbindungsendpunkt (CPU)
- Anzahl und Typ zusätzlicher Kommunikationsbaugruppen (CP, CM)

Die Anzahl der zur Verfügung stehenden Verbindungsressourcen ist bei den verschiedenen Automatisierungssystemen unterschiedlich:

- S7-300: Die eingesetzte CPU bestimmt die Anzahl der möglichen Ressourcen für Verbindungen. Die maximale Anzahl der Ressourcen für Verbindungen kann durch hinzugefügte CPs erweitert werden.
- S7-400: Die eingesetzte CPU bestimmt die Anzahl der möglichen Ressourcen für S7-Verbindungen. Die Ressourcen können auch für andere Verbindungstypen verwendet werden, wobei die maximale Anzahl der Ressourcen für andere Verbindungstypen durch hinzugefügte CPs erweitert werden kann.
- S7-1200: Die eingesetzte CPU bestimmt die Anzahl der möglichen Ressourcen für S7-Verbindungen. Diese Obergrenze kann nicht durch das Hinzufügen von CPs oder CMs erweitert werden.
- S7-1500: Die eingesetzte CPU bestimmt die für das ganze Automatisierungssystem geltende Obergrenze der Verbindungsressourcen. Die CPU selber steuert einen Teil dieser Verbindungsressourcen bei, weitere Ressourcen kommen über CPs und CMs hinzu. Unabhängig von der Anzahl der hinzugefügten CPs und CMs kann aber die Gesamtzahl der durch den CPU-Typ festgelegten Obergrenze an Verbindungsressourcen nicht überschritten werden.

Zeitpunkt der Belegung von Verbindungsressourcen

Welche Ressourcen tatsächlich im Betrieb beansprucht werden, hängt davon ab, welche Kommunikationsfunktionen zu einem bestimmten Zeitpunkt verwendet werden und welche Kommunikationsverbindungen zu einem bestimmten Zeitpunkt aufgebaut sind. Zum Zeitpunkt der Projektierung können daher nur begrenzte Überprüfungen auf Einhaltung bestimmter Mengengerüste und einer Verfügbarkeit von Kommunikations-Ressourcen durchgeführt werden. Die Verbindungsprojektierung bietet hierbei weitest mögliche Unterstützung.

Einflussfaktoren für das Mengengerüst

Mit der Konfiguration der Geräte bestimmen Sie die Kommunikationsressourcen:

- Anzahl der verfügbaren S7-Verbindungsressourcen in der CPU (S7-300/400)
- Anzahl der Verbindungsressourcen für S7 und offene Kommunikation in der Station (S7-1500)
- Anzahl der protokollspezifischen Verbindungsressourcen in den CPs/CMs
- Anzahl der verfügbaren CPs/CMs
- Automatisches, programmiertes oder projektiertes Einrichten der Verbindung

Verhalten der Verbindungsprojektierung

Die Verbindungsprojektierung prüft während der Eingabe folgende Sachverhalte und verhält sich wie folgt:

- Beim Anlegen von S7-Verbindungen
  Ist die maximale Anzahl verfügbarer S7-Verbindungsressourcen vergeben? Wenn ja, ist keine weitere S7-Verbindung über die gewählte Schnittstelle projektierbar.
  Bei S7-300 kann das Mengengerüst durch weitere CPs vergrößert werden.
- Beim Anlegen beliebiger Verbindungstypen
  Ist die maximale Anzahl der protokollspezifischen Verbindungsressourcen in der Schnittstelle (CP) vergeben? Wenn ja, ist keine weitere Verbindung des gewählten Typs über die gewählte Schnittstelle projektierbar.
  Bei S7-300/400 kann das Mengengerüst durch weitere CPs vergrößert werden.
- Beim Anlegen beliebiger Verbindungstypen bei OPC (PC-Stationen)
  Bei Verbindungen mit OPC als Verbindungsendpunkt können prinzipiell mehr Verbindungen projektiert werden, als der in der PC-Station verwendete CP zu einem Zeitpunkt betreiben kann. Dies hängt damit zusammen, dass bei den Verbindungseigenschaften zwischen "dauerhaft aufrecht erhalten" und "auf Anforderung aufbauen" unterschieden wird.
  Die Grenze für Softnet IE CPs liegt bei 64 gleichzeitigen Verbindungen und für Hardnet IE CPs bei 120 gleichzeitige Verbindungen.

Hinweis

Anzahl lokal projektierter Verbindungen größer als Anzahl zur Laufzeit möglicher Verbindungen

Der Anwender kann die zur Laufzeit maximal mögliche Anzahl S7- und SEND/RECEIVE-Verbindungen über OPC als "dauerhaft aufrecht erhalten" konfigurieren und dann keine weitere Verbindung als "auf Anforderung aufbauen" konfigurieren. Alternativ kann der Anwender weniger als die zur Laufzeit maximal mögliche Anzahl S7- und SEND/RECEIVE-Verbindungen über OPC als "dauerhaft aufrecht erhalten" konfigurieren und dafür weitere S7- und SEND/RECEIVE-Verbindungen als "auf Anforderung aufbauen" konfigurieren.

Wenn der Anwender mehr als 64 S7- und SEND/RECEIVE-Verbindungen konfiguriert, erfolgt eine Warnung bei der Übersetzung der Konfiguration. Die Warnung erfolgt unabhängig davon, ob die Verbindungen permanent ("dauerhaft aufrecht erhalten") oder temporär ("auf Anforderung aufbauen") konfiguriert sind, da selbst bei 65 temporären Verbindungen theoretisch zur Laufzeit eine Überschreitung der maximalen Anzahl gleichzeitiger Verbindungen eintreten kann.

Beim Übersetzen der Verbindungsprojektierung wird gegebenenfalls eine Warnung oder bei nicht zuordnebarer Schnittstelle ein Fehler ausgegeben.

Hinweis

S7-Verbindungsressourcen erschöpft

Wenn die maximale Anzahl an S7-Verbindungsressourcen reserviert ist, stehen im Betrieb ggf. keine S7-Verbindungsressourcen für Verbindungen zur Verfügung, die für eine Datenübertragung zwischen CPU und CP zusätzlich benötigt werden. Entsprechende Kommunikationsaufträge werden dann abgelehnt.

Sie sollten also in Abhängigkeit der insgesamt beanspruchten Kommunikationsfunktionen eine genügende Anzahl nicht belegter S7-Verbindungsressourcen sicherstellen.

Die Anzahl der reservierten S7-Verbindungsressourcen sehen Sie für die angewählte S7-300 oder S7-1200/1500 CPU in der Bereichsnavigation unter "Verbindungsressourcen". Für die S7-300 werden nur die Verbindungsressourcen der CPU angezeigt, bei S7-1200/1500 sehen Sie auch die Ressourcen der CPs und CMs.

Verbindungsressourcen und Kommunikationsart

Jede Verbindung benötigt auf den beteiligten Geräten Verbindungsressourcen für den Endpunkt bzw. für den Übergangspunkt (z. B. CP, CM). Die Anzahl der verfügbaren Verbindungsressourcen ist CPU/CP/CM-spezifisch.

Sind alle Verbindungsressourcen eines Kommunikationspartners belegt, so kann keine neue Verbindung aufgebaut werden. Im Folgenden wird jede Kommunikationsart einzeln betrachtet. Unter Berücksichtigung der verfügbaren Verbindungsressourcen sind jedoch beliebige Kombinationen möglich.

S7-Verbindungen über integrierte Schnittstelle

Bei S7-Verbindungen über die integrierte MPI-/PROFIBUS-DP/PN-Schnittstelle wird auf der CPU pro S7-Verbindung eine Verbindungsressource für den Endpunkt belegt. Dies gilt für alle S7-300/400/1200/1500-CPUs.

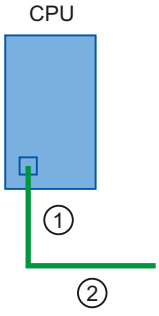

① Belegte S7-Verbindungsressource an der CPU

② Subnetz (z. B. MPI, PROFIBUS-DP oder Industrial Ethernet/PROFINET)

- S7-300/400 CPU: Es wird eine S7-Verbindungsressource auf der CPU belegt.
- S7-1200/1500 CPU: Es wird eine S7-Verbindungsressource auf der CPU belegt und damit auch eine S7-Verbindungsressource auf der Station.

S7-Verbindungen über externe Schnittstelle

Bei S7-Verbindungen über eine externe CP-Schnittstelle sind folgende Fälle zu unterscheiden:

- S7-300 CPU
  - Einseitige S7-Verbindung: Es werden auf der CPU (für den Endpunkt) und auf dem CP (Übergangspunkt) pro S7-Verbindung je eine S7-Verbindungsressource belegt.

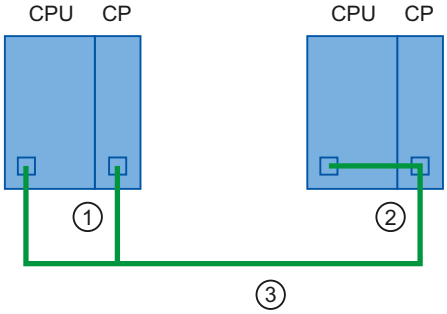

① Eine S7-Verbindungsressource in der CPU und eine im CP des lokalen Endpunkts
② Verbindungspartner
③ Subnetz, z.B. Industrial Ethernet

  - Zweiseitige S7-Verbindung: Es wird auf dem CP pro S7-Verbindung je eine S7-Verbindungsressource belegt.

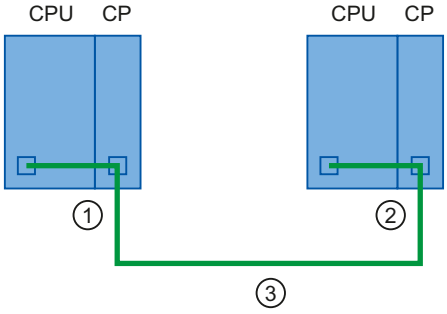

① Eine S7-Verbindungsressource im CP des lokalen Endpunkts
② Verbindungspartner
③ Subnetz, z.B. Industrial Ethernet

- S7-400 CPU
  Es werden auf der CPU (für den Endpunkt) und auf dem CP (Übergangspunkt) pro S7-Verbindung je eine S7-Verbindungsressource belegt.

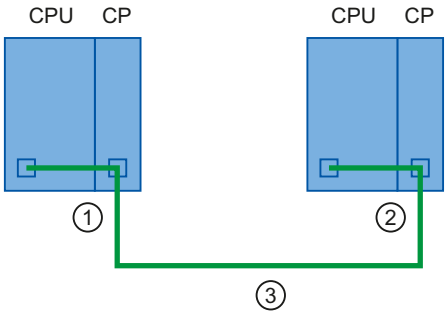

① Eine S7-Verbindungsressource in der CPU und eine im CP des lokalen Endpunkts
② Verbindungspartner
③ Subnetz, z.B. Industrial Ethernet

- S7-1200/1500 CPU
  - Einseitige S7-Verbindung: Es werden pro S7-Verbindung eine S7-Verbindungsressource auf der CPU belegt und damit auch eine S7-Verbindungsressource auf der Station.

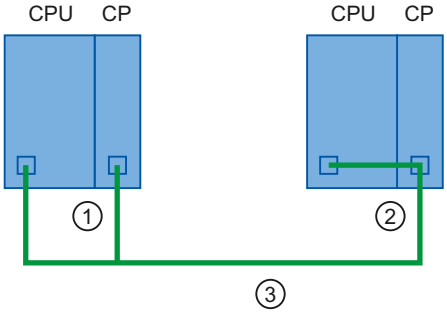

① Eine S7-Verbindungsressource in der CPU und eine im CP des lokalen Endpunkts
② Verbindungspartner
③ Subnetz, z.B. Industrial Ethernet

  - Zweiseitige S7-Verbindung: Es werden pro S7-Verbindung eine S7-Verbindungsressource auf der CPU belegt und damit auch eine S7-Verbindungsressource auf der Station.

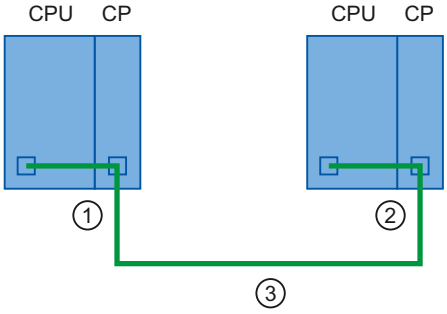

① Eine S7-Verbindungsressource im CP und somit auch eine in der Station
② Verbindungspartner
③ Subnetz, z.B. Industrial Ethernet

Hinweis

Die Anzahl der reservierten S7-Verbindungsressourcen sehen Sie bei bestimmten CPU-Typen der S7-300 für die angewählte CPU und bei S7-1200/1500 für die angewählte CPU und ihre angeschlossenen Kommunikationsmodule in der Bereichsnavigation unter "Verbindungsressourcen". S7-Verbindungsressourcen für OP-/PG- und S7-Basiskommunikation können hier reserviert werden.

Verbindungen für Offene Kommunikationsdienste (SEND/RECEIVE-Schnittstelle)

Die Kommunikation über die SEND/RECEIVE-Schnittstelle erfolgt ausschließlich über CPs. Hierbei wird pro Verbindung (d. h. FDL-, ISO-Transport- ISO-on-TCP-, UDP- und TCP-Verbindung) auf dem CP eine protokollspezifische Verbindungsressource für den Endpunkt belegt.

Auf der CPU werden hier keine Verbindungsressourcen für die Verbindung benötigt.

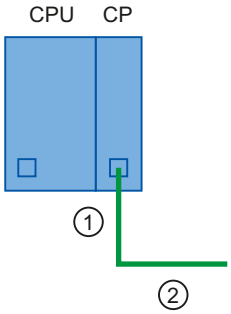

① 1 belegte Verbindungsressource im CP

② Subnetz, z.B. Industrial Ethernet

Ausnahme: Auf der S7-400-CPU benötigen Verbindungen mit der Eigenschaft "SPEED SEND/RECEIVE" eine zusätzliche S7-Verbindungsressource für jede "SPEED SEND/RECEIVE"-Verbindung auf der CPU.

Verbindungsressourcen und Verbindungstyp

Einrichten der Verbindung

Sie können sich oftmals alternativ für ein projektiertes oder programmiertes Einrichten entscheiden. Das programmierte Einrichten erleichtert die Freigabe von Verbindungsressourcen nach der Datenübertragung. Beim projektierten Einrichten werden Sie bei der Verwaltung von Verbindungsressourcen in STEP 7 unterstützt.

Tabelle 1-2 Einrichten des Kommunikationsdienstes

Kommunikationsdienst	Automatisches Einrichten	Programmiertes Einrichten	Projektiertes Einrichten
PG-Kommunikation	X	-	-
HMI-Kommunikation	X	-	X

Kommunikationsdienst	Automatisches Einrichten	Programmiertes Einrichten	Projektiertes Einrichten
S7-Kommunikation	-	X	X
Offene Kommunikation über TCP/IP	-	X	X
Offene Kommunikation über ISO-on-TCP	-	X	X
Offene Kommunikation über UDP	-	X	X
Offene Kommunikation über ISO	-	X	X
Offene Kommunikation über Email	-	X	-
Offene Kommunikation über FTP	-	X	-
Offene Kommunikation über FDL	-	X	X
S7-Routing	-	-	X

Belegung der Verbindungsressourcen in Abhängigkeit von der Verbindungseinrichtung

Je nach Zustandekommen der Verbindung erfolgt die Belegung der Verbindungsressourcen auf unterschiedliche Weise:

- Projektiertes Eintrichten
  Bei projektierten Verbindungen wird die Verbindungsressource belegt, sobald die Hardware-Konfiguration in die CPU geladen wurde. Für OPC-Server erfolgt die Belegung der Ressourcen nach Bedarf.
- Programmiertes Einrichten
  Mit dem Aufruf der entsprechenden Anweisungen zum Aufbau der Verbindung (über TSEND_C oder T_CON) wird eine Verbindungsressource belegt.
- Automatisches Einrichten
  Sobald Sie das PG bzw. HMI-Gerät mit einer CPU physikalisch und online verbunden haben, werden die Verbindungsressourcen belegt.

Verbindungsressourcen beim projektierten Einrichten

Die Verbindung richten Sie in der Netzsicht des Hardware- und Netzwerkeditors im Kontext einer CPU ein.

Nach der Datenübertragung wird die Verbindung nicht abgebaut, d. h. die Verbindungsressourcen bleiben dauerhaft belegt. Um die Verbindung abzubauen, müssen Sie die Verbindungsprojektierung löschen und die geänderte Projektierung in die CPU laden.

OPC-Verbindungen werden je nach Bedarf automatisch auf- und abgebaut.

Wenn Sie Verbindungen **projektiert** haben, wird die Einhaltung der maximal möglichen Anzahl der Verbindungsressourcen in dem Automatisierungssystem überwacht.

Verbindungsressourcen beim programmierten Einrichten

Die Verbindung richten Sie im Programmeditor im Kontext einer CPU durch die Parametrierung von Anweisungen für Kommunikation, z. B. TSEND_C ein.

Nach dem Verbindungsaufbau durch die Anweisungen TCON oder TSEND_C, wird die Verbindung durch entsprechende Parametrierung der Anweisung TSEND_C bzw. den Aufruf der Anweisung TDISCON nach der Datenübertragung abgebaut. Wenn die Verbindung abgebaut wird, stehen die Verbindungsressourcen in der CPU/CP/CM wieder zur Verfügung. Bleibt die Verbindung bestehen, ist die Bearbeitungszeit der Anweisungen bei einer erneuten Datenübertragung kürzer, da die Verbindung nicht erneut aufgebaut werden muss.

Wenn der Auf- und Abbau der Verbindungen **programmiert** im Anwenderprogramm erfolgt, müssen Sie selbst für die Einhaltung der Grenzen im Automatisierungssystem sorgen. Sie erhalten ggf. einen Kommunikationsfehler am Ausgangsparameter "STATUS" der Anweisung TSEND_C bzw. T_CON und können über die Verbindungsdiagnose nähere Informationen erhalten.

Verbindungsressourcen beim automatischen Einrichten

Die Verbindung wird automatisch eingerichtet (z. B. PG- oder HMI-Verbindung), sobald Sie die PG/PC-Schnittstelle physikalisch mit einer Schnittstelle der CPU verbunden haben und im Dialog "Online verbinden" die Zuordnung zur Schnittstelle vorgenommen haben.

Für **automatisch** eingerichtete Verbindungen wird die Einhaltung der dafür reservierten Verbindungsressourcen in dem Automatisierungssystem überwacht.

Verbindungsressourcen S7-1200

Anzeige der Verbindungsressourcen

Verbindungsressourcen im Inspektorfenster

Sie finden eine Übersicht über die reservierten und verfügbaren Verbindungsressourcen eines Automatisierungssystems S7-1200 in den Eigenschaften der CPU im Inspektorfenster des Hardware- und Netzwerkeditors.

Übersicht über die Verbindungsressourcen

Die CPU unterstützt maximal die folgende Anzahl von gleichzeitigen, asynchronen Kommunikationsverbindungen für PROFINET und PROFIBUS. Die maximale Anzahl der jeder Kategorie zugeordneten Verbindungsressourcen ist festgelegt. Sie können die Werte nicht ändern. Die 6 frei verfügbaren Verbindungen können jedoch genutzt werden, um die Anzahl Verbindungen in jeder Kategorie nach dem Bedarf Ihrer Anwendung zu erhöhen.

Bei der Anzeige der Verbindungsressourcen wird unterschieden, ob es sich um die Offline- oder Online-Sicht handelt. Im folgenden Bild sehen Sie die Offline-Anzeige der reservierten und verfügbaren Verbindungsressourcen einer CPU 1217C:

Verbindungsressourcen

	① Ressourcen der Station			② Ressourcen des Moduls
	Reserviert		Dynamisch	PLC_11 [CPU 1217C DC/DC/DC]
Maximale Anzahl der Ressourcen:	62		6	68
	Maximum	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-
HMI-Kommunikation:	12	0	0	0
S7-Kommunikation:	8	0	0	0
Open User Communication:	8	0	0	0
Web-Kommunikation:	30	-	-	-
Sonstige Kommunikation:	-	-	0	0
Insgesamt verwendete Ressourcen:		0	0	0
Verfügbare Ressourcen:		62	6	68

① Stationsspezifische Verbindungsressourcen

② Modulspezifische Verbindungsressourcen

Stationsspezifische Verbindungsressourcen

Die Spalten der stationsspezifischen Verbindungsressourcen informieren über die reservierten und dynamisch verwendbaren Verbindungsressourcen der Station. Im Beispiel stehen für das Automatisierungssystem maximal 68 stationsspezifische Verbindungsressourcen zur Verfügung:

- 62 Verbindungsressourcen sind für bestimmte Verbindungsarten reserviert
- 6 Verbindungsressourcen können frei für Verbindungsarten verwendet werden

Das Warndreieck in der Spalte der dynamischen Stationsressourcen wird deshalb angezeigt, weil die Summe der max. verfügbaren Verbindungsressourcen der CPU die Stationsgrenze von 68 erreicht.

Hinweis

Überschreiten der verfügbaren Verbindungsressourcen

Ein Erreichen der stationsspezifischen Verbindungsressourcen meldet STEP 7 mit einer Warnung. Wenn Sie mehr Verbindungsressourcen benötigen, müssen Sie entweder eine CPU mit einer größeren max. Anzahl verfügbarer stationsspezifischer Verbindungsressourcen einsetzen oder die Anzahl der Kommunikationsverbindungen reduzieren.

Modulspezifische Verbindungsressourcen

Die Spalten der modulspezifischen Verbindungsressourcen informieren über die Ressourcenbelegung in den CPUs, CPs und CMs eines Automatisierungssystems. Die Anzeige ist modul- und nicht schnittstellengranular. Im Beispiel stellt die CPU maximal 68 Verbindungsressourcen zur Verfügung.

Hinweis

Wenn Sie CM/CP-Module ergänzen, erhöht sich die Gesamtzahl der S7-1200 Kommunikationsressourcen nicht.

Online-Sicht

Wenn Sie online mit der CPU verbunden sind, dann können Sie sich im Inspektorfenster im Bereich "Diagnose" im Register "Verbindungsinformation" anzeigen lassen, wie viele Ressourcen jeweils aktuell verwendet werden.

Die Online-Sicht der Tabelle "Verbindungsressourcen" enthält zusätzlich zu den reservierten und dynamischen Verbindungsressourcen der projektierten Verbindungen auch Spalten mit den aktuell in der Station belegten Verbindungsressourcen der programmierten und automatisch eingerichteten Verbindungen. In der Online-Sicht werden also **alle** belegten Verbindungsressourcen im Automatisierungssystem angezeigt, unabhängig davon, wie die Verbindung eingerichtet wurde. In der Zeile "Sonstige Kommunikation" werden belegte Verbindungsressourcen für Kommunikation zu Fremdgeräten und Kommunikation über Datensatz-Routing angezeigt.

Die Tabelle wird automatisch aktualisiert. Falls Inkonsistenzen zwischen der projektierten Hardware-Konfiguration und der online vorhandenen bestehen, wird eine entsprechende Meldung ausgegeben.

Ressourcenbedarf der Verbindungen

Die Kommunikationsverbindungen von offener Benutzerkommunikation, S7-Verbindung, HMI, Programmiergerät und Webserver (HTTP) können basierend auf den verwendeten Funktionen mehrere Verbindungsressourcen nutzen.

Basierend auf den zugeordneten Verbindungsressourcen steht die folgende Anzahl von Verbindungen je Gerät zur Verfügung:

	Programmiergerät (PG)	HMI-Geräte	GET/PUT-Client/Server	Offene Benutzerkommunikation	Webbrowser
Maximale Anzahl von Verbindungsressourcen einer CPU 1217C	4 (garantierte Unterstützung einer PG-Verbindung)	12 (garantierte Unterstützung von vier HMI-Verbindungen)	8	8	30 (garantierte Unterstützung von drei HTTP-Verbindungen)

PG-Kommunikation

Beispielsweise hat stehen für eine PG-Kommunikation vier Verbindungsressourcen zur Verfügung. Je nach aktuell verwendeten PG-Funktionen belegt das PG für eine Verbindung bis zu drei Verbindungsressourcen. Bei der S7-1200 besteht jederzeit die Garantie für mindestens eine PG-Verbindung.

HMI-Kommunikation

Ein weiteres Beispiel ist die Anzahl der HMI-Geräte, wie in der nachstehenden Abbildung gezeigt. Für HMI-Geräte stehen 12 Verbindungsressourcen zur Verfügung. Je nach Typ oder Modell Ihres HMI-Geräts und den verwendeten HMI-Funktionen kann HMI-Gerät für eine HMI-Verbindung bis zu drei Verbindungsressourcen benötigen. Ein Basic Panel benötigt beispielsweise für eine HMI-Verbindung bis zu zwei, ein Comfort Panel bis zu drei Verbindungsressourcen in der CPU. Abhängig von der Anzahl der benötigten Verbindungsressourcen können mindestens vier HMI-Verbindungen gleichzeitig konfiguriert sein.

Ein HMI-Gerät kann seine verfügbaren Verbindungsressourcen für die folgenden Funktionen nutzen:

- Lesen
- Schreiben
- Alarme und Diagnosen

Dies ist nur ein Beispiel. Die tatsächliche Anzahl der Verbindungen kann je nach Typ und Version des HMI-Geräts unterschiedlich sein.

Beispiel	HMI 1	HMI 2	HMI 3	HMI 4	HMI 5	Gesamtzahl verfügbarer Verbin-dungsres-sourcen
Verwendete Verbin-dungsres-sourcen	2	2	2	3	3	12

Informationen über die Verfügbarkeit und Belegung von Verbindungsressourcen für HMI-Verbindungen finden Sie in der Offline-Sicht im Kontext des HMI-Geräts (im Inspektorfenster unter "Eigenschaften > Allgemein > Verbindungsressourcen"):

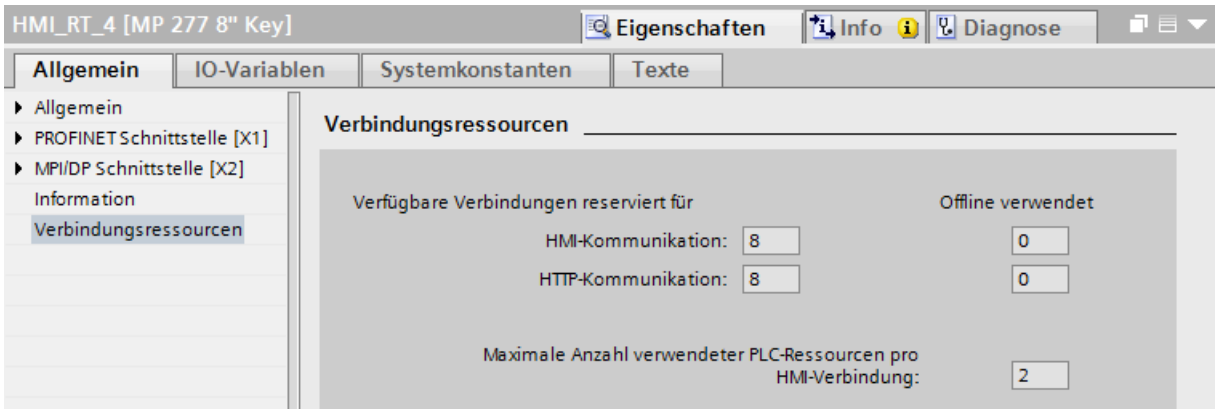

Im Bereich Verbindungsressourcen werden angezeigt:

- Anzahl der in HMI verfügbaren Verbindungen reserviert für HMI-Kommunikation und HTTP-Kommunikation
- Anzahl der offline in HMI verwendeten Verbindungsressourcen für HMI-Kommunikation und HTTP-Kommunikation
  Falls die Anzahl der maximal verfügbaren Verbindungsressourcen für ein HMI-Gerät überschritten ist, wird eine entsprechende Meldung von STEP 7 ausgegeben.
- Maximale Anzahl verwendeter PLC-Ressourcen pro HMI-Verbindung. Dieser Parameter ist ein Faktor, der mit der Anzahl der offline verwendeten HMI-Verbindungen zu multiplizieren ist. Das Produkt ergibt die Anzahl der in der CPU belegten HMI-Ressourcen.

Falls die Anzahl der maximal verfügbaren Verbindungsressourcen für ein HMI-Gerät überschritten ist, wird eine entsprechende Meldung ausgegeben.

Web-Kommunikation

Die CPU bietet Webserver (HTTP)-Verbindungen für mehrere Webbrowser. Die Anzahl von Browsern, die von der CPU gleichzeitig unterstützt werden können, richtet sich danach, wie viele Verbindungen ein Webbrowser anfordert/verwendet.

Verbindungsressourcen S7-1500

Verbindungsressourcen im Automatisierungssystem

Einleitung

Einige Kommunikationsdienste benötigen Verbindungen. Verbindungen belegen im Automatisierungssystem Ressourcen. Die Verbindungsressourcen werden von den CPUs, Kommunikationsprozessoren (CP) und Kommunikationsmodulen (CM) zur Verfügung gestellt.

Verbindungsressourcen einer Station

Die zur Verfügung stehenden Verbindungsressourcen sind abhängig von den eingesetzten CPUs, CPs und CMs und dürfen eine maximale Anzahl pro Station nicht überschreiten. Die maximale Anzahl der Ressourcen einer Station wird durch die CPU bestimmt.

Jede CPU bringt reservierte Verbindungsressourcen für PG-, HMI- und Webserver-Kommunikation mit. Damit ist z. B. sichergestellt, dass ein PG immer mindestens eine Online-Verbindung mit der CPU aufbauen kann, unabhängig davon, wie viele andere Kommunikationsdienste bereits Verbindungsressourcen belegen. Daneben gibt es dynamische Ressourcen. Die Differenz zwischen der maximalen Anzahl der Verbindungsressourcen und der Anzahl reservierter Verbindungsressourcen ist die maximale Anzahl dynamischer Verbindungsressourcen. Aus dem Pool der dynamischen Verbindungsressourcen bedienen sich die Kommunikationsdienste PG-Kommunikation, HMI-Kommunikation, S7-Kommunikation, Open User Communication, Web-Kommunikation und sonstige Kommunikation (z. B. OPC UA).

Das folgende Bild zeigt beispielhaft, wie einzelne Komponenten Verbindungsressourcen einer S7-1500-Station zur Verfügung stellen.

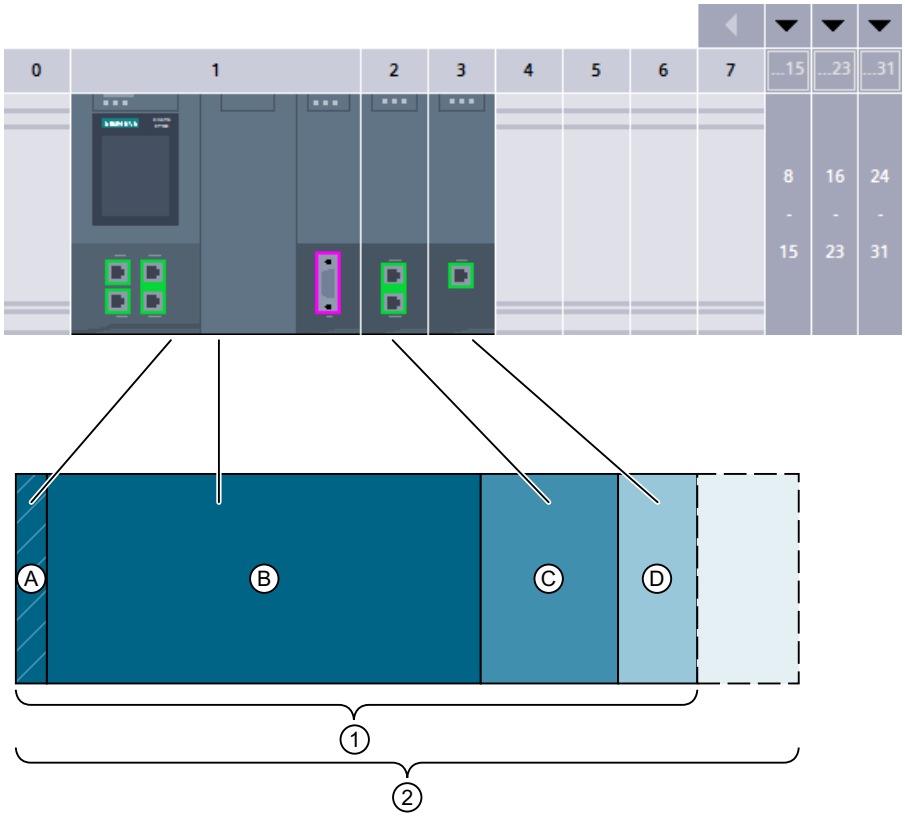

① Verfügbare Verbindungsressourcen der Station, davon

A	Reservierte Verbindungsressourcen der Station
A + B	Verbindungsressourcen der CPU 1518
C	Verbindungsressourcen des Kommunikationsmoduls CM 1542-1
D	Verbindungsressourcen des Kommunikationsprozessors CP 1543-1

② Maximale Verbindungsressourcen der Station am Beispiel einer Konfiguration aus CPU 1518, CM 1542-1 und CP 1543-1

Anzahl Verbindungsressourcen einer Station

In der folgenden Tabelle finden Sie die maximal verfügbaren Verbindungsressourcen für einige CPU-Typen:

Verbindungsressourcen einer Station	1511 1511C	1512C 1513	1515	1516	1517	1518
Maximal Verbindungsressourcen der Station	96	128	192	256	320	384
davon reserviert	10					
davon dynamisch	86	118	182	246	310	374
Verbindungsressourcen der CPU	64	88	108	128	160	192
Max. zusätzlich nutzbare Verbindungsressourcen durch Stecken von CMs/CPs	32	40	84	128	160	192

Verbindungsressourcen einer Station	1511 1511C	1512C 1513	1515	1516	1517	1518
Zusätzliche Verbindungsressourcen CM 1542-1	64					
Zusätzliche Verbindungsressourcen CP 1543-1	118					
Zusätzliche Verbindungsressourcen CM 1542-5	40					
Zusätzliche Verbindungsressourcen CP 1542-5	16					

Wieviele Verbindungsressourcen eine CPU bzw. ein Kommunikationsmodul unterstützt, finden Sie in den Gerätehandbüchern in den Technischen Daten beschrieben.

Beispiel

Sie haben eine CPU 1518-4PN/DP mit einem Kommunikationsmodul CM 1542-1 und einem Kommunikationsprozessor CP 1542-5 konfiguriert.

- Maximale Verbindungsressourcen der Station: **384**
- Verfügbare Verbindungsressourcen:
  - CPU 1518-4 PN/DP: 192
  - CM 1542-1: 64
  - CP 1542-5: 16
  - Gesamt: **272**

Der Aufbau stellt 272 Verbindungsressourcen zur Verfügung. Durch Hinzufügen weiterer Kommunikationsmodule kann die Station maximal 112 weitere Verbindungsressourcen unterstützen.

Reservierte Verbindungsressourcen

Für Stationen mit S7-1500 CPU, ET 200SP CPU und ET 200pro CPU auf Basis S7-1500 sind 10 Verbindungsressourcen reserviert:

- 4 für PG-Kommunikation, die von STEP 7 benötigt werden für z. B. Test- und Diagnosefunktionen oder das Laden in die CPU
- 4 für HMI-Kommunikation, die durch die ersten, in STEP 7 projektierten HMI-Verbindungen belegt werden
- 2 für Kommunikation zum Webserver

Belegung von Verbindungsressourcen

Überblick - Belegung von Verbindungsressourcen

Das folgende Bild zeigt, wie verschiedene Verbindungen die Ressourcen der S7-1500 belegen:

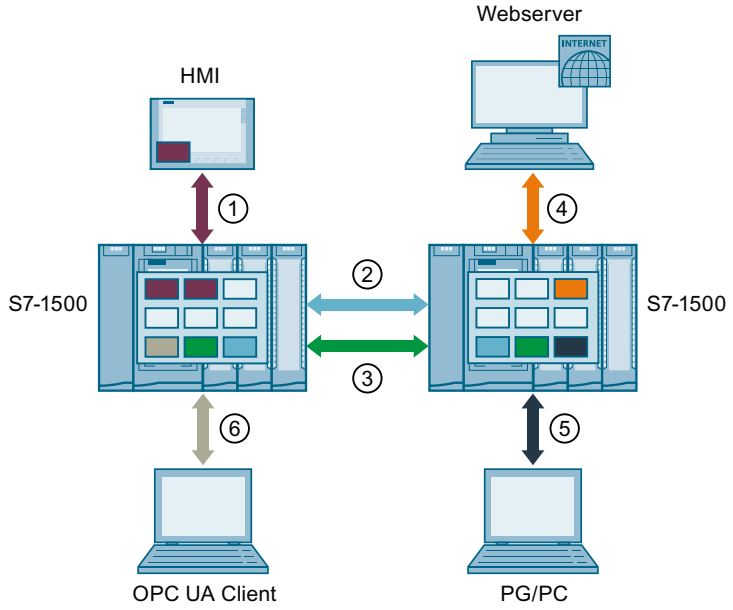

① HMI-Kommunikation: siehe unten

② Open User Communication: Verbindungen der Open User Communication belegen in jedem Endpunkt eine Verbindungsressource.

③ S7-Kommunikation: Verbindungen der S7-Kommunikation belegen in jedem Endpunkt eine Verbindungsressource.

④ Web-Kommunikation: Die Webserver-Verbindung belegt in der Station mindestens eine Verbindungsressource. Die Anzahl der belegten Verbindungen hängt vom Browser ab.

⑤ PG-Kommunikation: Die PG-Verbindung belegt in der Station eine Verbindungsressource.

⑥ OPC UA-Kommunikation: Jede Session, die der OPC UA-Server der CPU mit einem OPC UA-Client aufbaut, belegt in der Regel eine Verbindungsressource (Sonstige Kommunikation) in der Station.

- Verbindungsressource für HMI-Kommunikation
- Verbindungsressource für OpenUser-Kommunikation
- Verbindungsressource für S7-Kommunikation
- Verbindungsressource für Web-Kommunikation
- Verbindungsressource für PG-Kommunikation
- Verbindungsressource für Sonstige Kommunikation (z. B. OPC UA)

Verbindungsressourcen für HMI-Kommunikation

Bei HMI-Kommunikation ist die Belegung von Verbindungsressourcen in der Station abhängig vom eingesetzten HMI-Gerät. In der folgenden Tabelle finden Sie die maximal belegten Verbindungsressourcen für eine HMI-Verbindung in Abhängigkeit zu den verschiedenen HMI-Geräten:

HMI-Gerät	max. belegte Verbindungsressourcen der Station pro HMI-Verbindung
Basic Panel	1
Comfort Panel	2[1]
RT Advanced	2[1]
RT Professional	3

[1] Wenn Sie keine Systemdiagnose und keine Meldungsprojektierung nutzen, dann belegt die Station pro HMI-Verbindung nur eine Verbindungsressource.

Beispiel: Sie haben für eine CPU 1516-3 PN/DP folgende HMI-Verbindungen konfiguriert:

- Zwei HMI-Verbindungen zu einem HMI TP700 Comfort (jeweils 2 Verbindungsressourcen)
- Eine HMI-Verbindung zu einem HMI KTP1000 Basic (1 Verbindungsressource)

Insgesamt werden in der CPU 5 Verbindungsressourcen für HMI-Kommunikation belegt.

Verbindungsressourcen für Routing

S7-Verbindungen über Netzgrenzen hinweg (S7-Routing)

Für die Übertragung von Daten über S7-Subnetze hinweg ("S7-Routing") wird eine S7-Verbindung zwischen zwei CPUs aufgebaut. Die S7-Subnetze sind über Netzübergänge, sogenannte S7-Router miteinander verbunden. Der Router ist der Durchgangspunkt einer S7-Verbindung und kann S7-Verbindungen aufbauen. CPUs, CMs und CPs in S7-1500 sind S7-Router.

Für eine geroutete S7-Verbindung gilt folgendes:

- Eine geroutete Verbindung belegt in beiden Endpunkten jeweils eine Verbindungsressource, STEP 7 zeigt diese Verbindungsressourcen in der Tabelle "Verbindungsressourcen" an.
- Im S7-Router werden zwei spezielle Verbindungsressourcen für S7-Routing belegt. STEP 7 zeigt die speziellen Verbindungsressourcen für S7-Routing nicht in der Tabelle "Verbindungsressourcen" an. Die Anzahl der Ressourcen für S7-Routing ist CPU abhängig. Sie finden der Ressourcen für S7-Routing in den Technischen Daten der CPU unter "Anzahl S7-Routing Verbindungen".

Im folgenden Bild sehen Sie die Belegung von Verbindungsressourcen beim S7-Routing:

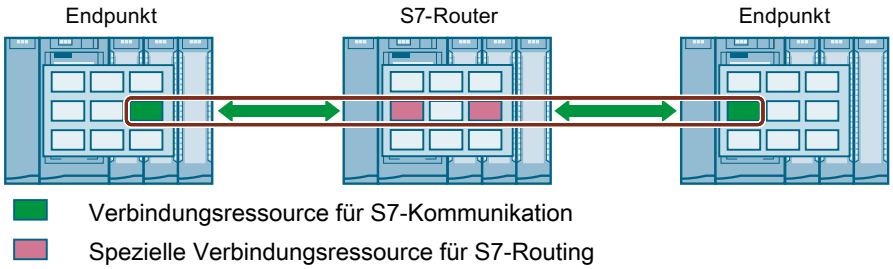

Datensatz-Routing

Datensatz-Routing ermöglicht ebenfalls die Übertragung von Daten über S7-Subnetze hinweg, von einer Engineering Station, die am PROFINET angeschlossen ist, über PROFIBUS an diverse Feldgeräte.

Wie beim S7-Routing werden auch beim Datensatz-Routing an den Netzübergängen in jedem Datensatz-Router zwei der speziellen Verbindungsressourcen für S7-Routing belegt. Im Endpunkt der Kommunikation (CPU) werden Verbindungsressourcen für Datensatz-Routing in der Online-Sicht der Tabelle "Verbindungsressourcen" in der Zeile "Sonstige Kommunikation" angezeigt.

Hinweis

Verbindungsressourcen beim Datensatz-Routing

Im Datensatz-Router werden beim Datensatz-Routing zwei spezielle Verbindungsressourcen für S7-Routing belegt. Weder die Datensatzverbindung, noch die belegten Verbindungsressourcen werden in der Tabelle Verbindungsressourcen angezeigt. Sie müssen die Einhaltung der Grenzen bei der Projektierung selber beachten.

Wann werden Verbindungsressourcen belegt?

Der Zeitpunkt für die Belegung der Verbindungsressourcen hängt davon ab, wie die Verbindung eingerichtet wird.

- Programmiertes Einrichten einer Verbindung
  Sobald im Anwenderprogramm der CPU eine Anweisung zum Aufbau einer Verbindung (TSEND_C/TRCV_C oder TCON) aufgerufen wird, wird eine Verbindungsressource belegt. Durch entsprechende Parametrierung des CONT Parameters der Anweisungen TSEND_C/TRCV_C bzw. den Aufruf der Anweisung TDISCON kann die Verbindung nach der Datenübertragung abgebaut und die Verbindungsressource freigegeben werden. Wenn die Verbindung abgebaut wird, stehen die Verbindungsressourcen in der CPU/CP/CM wieder zur Verfügung.
- Konfigurierte Verbindungen (z. B. HMI-Verbindung)
  Wenn Sie eine Verbindung in STEP 7 konfiguriert haben, dann wird die Verbindungsressource belegt, sobald die Hardware-Konfiguration in die CPU geladen wurde. Nach der Nutzung einer projektierten Verbindung zur Datenübertragung wird die Verbindung nicht abgebaut. Die Verbindungsressource bleibt dauerhaft belegt. Um die Verbindungsressource wieder frei zu geben, müssen Sie die projektierte Verbindung in STEP 7 löschen und die geänderte Projektierung in die CPU laden.

- PG-Verbindung
  Sobald Sie das PG mit einer CPU online in STEP 7 verbunden haben, werden Verbindungsressourcen belegt.
- Webserver
  Solange Sie den Webserver der CPU in einem Browser geöffnet haben, werden Verbindungsressourcen in der CPU belegt.
- OPC UA Server
  Solange eine Session zwischen dem OPC UA Server der CPU und einem OPC UA Client besteht, wird eine Verbindungsressource in der CPU belegt.

Überwachung der maximal möglichen Anzahl Verbindungsressourcen

Offline

STEP 7 überwacht beim Konfigurieren von Verbindungen die Belegung von Verbindungsressourcen. Ein Überschreiten der maximal möglichen Anzahl von Verbindungsressourcen meldet STEP 7 mit einer entsprechenden Warnung.

Online

Die CPU überwacht den Verbrauch von Verbindungsressourcen im Automatisierungssystem. Wenn Sie im Anwenderprogramm mehr Verbindungen aufbauen, als das Automatisierungssystem Verbindungsressourcen bereitstellt, dann quittiert die CPU die Anweisung zum Aufbau der Verbindung mit einem Fehler.

Anzeige der Verbindungsressourcen

Verbindungsressourcen im Inspektorfenster

Sie finden eine Übersicht über die reservierten, verfügbaren Verbindungsressourcen des Automatisierungssystems S7-1500 in den Eigenschaften der CPU im Inspektorfenster des Hardware- und Netzwerkeditors.

Anzeige der Verbindungsressourcen in STEP 7 (Offline-Sicht)

Bei der Anzeige der Verbindungsressourcen wird unterschieden, ob es sich um die Offline- oder Online-Sicht handelt. Im folgenden Bild sehen Sie die Offline-Anzeige der reservierten und verfügbaren Verbindungsressourcen einer S7-1500-CPU mit einem CP und einem CM:

Verbindungsressourcen

	① Ressourcen der Station			② Ressourcen ...	Ressourcen ...	Ressourcen ...
	Reserviert		Dynamis.. ⚠	PLC_1 [CPU ...	CP 1543-1_...	CM 1542-5_...
Maximale Anzahl der Ressourcen:	10		246	128	118	48
	Maximum	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	6	6	4	0
S7-Kommunikation:	0	-	23	2	0	21
Open User Communication:	0	-	52	39	13	0
Web-Kommunikation:	2	-	-	-	-	-
Sonstige Kommunikation:	-	-	0	0	0	0
Insgesamt verwendete Ressourcen:		4	81	47	17	21
Verfügbare Ressourcen:		6	165	81	101	27

① Stationsspezifische Verbindungsressourcen

② Modulspezifische Verbindungsressourcen

Stationsspezifische Verbindungsressourcen

Die Spalten der stationsspezifischen Verbindungsressourcen informieren über die reservierten und dynamisch verwendbaren Verbindungsressourcen der Station. Im Beispiel stehen für das Automatisierungssystem maximal 256 stationsspezifische Verbindungsressourcen zur Verfügung:

- 10 reservierte Verbindungsressourcen, davon sind 4 bereits verwendet und 6 noch verfügbar.
  Die belegten 4 Ressourcen werden für HMI-Kommunikation verwendet.
- 246 dynamische Verbindungsressourcen, davon sind 81 bereits verwendet und 165 noch verfügbar.
  Die konfigurierten Ressourcen teilen sich wie folgt auf:
  - 6 Ressourcen für HMI-Kommunikation
  - 23 Ressourcen für S7-Kommunikation
  - 52 Ressourcen für Open User Communication

Das Warndreieck in der Spalte der dynamischen Stationsressourcen wird deshalb angezeigt, weil die Summe der max. verfügbaren Verbindungsressourcen von CPU, CP und CM (= 294 Verbindungsressourcen) die Stationsgrenze von 256 überschreitet.

Hinweis

Überschreiten der verfügbaren Verbindungsressourcen

Ein Überschreiten der stationsspezifischen Verbindungsressourcen meldet STEP 7 mit einer Warnung. Wenn Sie die verfügbaren Verbindungsressourcen aus CPU, CP und CM voll ausnutzen wollen, dann müssen Sie entweder eine CPU mit einer größeren max. Anzahl verfügbarer stationsspezifischer Verbindungsressourcen einsetzen oder die Anzahl der Kommunikationsverbindungen reduzieren.

Modulspezifische Verbindungsressourcen

Die Spalten der modulspezifischen Verbindungsressourcen informieren über die Ressourcenbelegung in den CPUs, CPs und CMs eines Automatisierungssystems. Die Anzeige ist modul- und nicht schnittstellengranular. Im Beispiel stellt die CPU maximal 128 Verbindungsressourcen zur Verfügung, davon sind 47 bereits verwendet und 81 noch verfügbar:
Die konfigurierten Ressourcen teilen sich wie folgt auf:

- 6 Ressourcen für HMI-Kommunikation
- 2 Ressourcen für S7-Kommunikation
- 39 Ressourcen für Open User Communication

Der CP stellt 118 Ressourcen und das CM 48 Ressourcen zur Verfügung, von denen 17 bzw. 21 Ressourcen verwendet sind.

Anzeige der Verbindungsressourcen in STEP 7 (Online-Sicht)

Wenn Sie online mit der CPU verbunden sind, dann können Sie sich im Inspektorfenster im Bereich "Diagnose" im Register "Verbindungsinformation" anzeigen lassen, wie viele Ressourcen jeweils aktuell verwendet werden. Im folgenden Bild sehen Sie die Online-Anzeige der reservierten, verfügbaren und verwendeten Verbindungsressourcen einer S7-1500-CPU:

PLC_2 [CPU 1516-3 PN/DP] | Eigenschaften | Info | Diagnose

Geräte-Information | Verbindungsinformation | Meldungsanzeige

Verbindungsressourcen

	Ressourcen der Station					Ressourcen des Moduls	
	Reserviert			Dynamisch		PLC_2 [CPU 1516-3 PN/DP]	
Maximale Anzahl der Ressourcen:		10	10	194	194	86	86
	Maximum	Konfigurierte	Verwendet	Konfigurierte	Verwendet	Konfigurierte	Verwendet
PG-Kommunikation:	4	-	4	-	0	-	0
HMI-Kommunikation:	4	4	4	4	4	8	8
S7-Kommunikation:	0	0	0	72	68	34	34
Open User Communication:	0	0	0	118	118	45	45
Web-Kommunikation:	2	-	2	-	0	-	0
Sonstige Kommunikation:	-	-	0	-	0	-	0
Insgesamt verwendete Ressourc...		4	10	194	190	82	82
Verfügbare Ressourcen:		6	0	0	4	4	4

Die Online-Sicht der Tabelle "Verbindungsressourcen" enthält zusätzlich zu den reservierten und dynamischen Verbindungsressourcen der projektierten Verbindungen auch Spalten mit den aktuell in der Station belegten Verbindungsressourcen der programmierten und automatisch eingerichteten Verbindungen. In der Online-Sicht werden also **alle** belegten Verbindungsressourcen im Automatisierungssystem angezeigt, unabhängig davon, wie die Verbindung eingerichtet wurde. In der Zeile "Sonstige Kommunikation" werden belegte Verbindungsressourcen für Kommunikation zu Fremdgeräten und Kommunikation über Datensatz-Routing angezeigt.

Die Tabelle wird automatisch aktualisiert. Falls Inkonsistenzen zwischen der projektierten Hardware-Konfiguration und der online vorhandenen bestehen, wird eine entsprechende Meldung ausgegeben.

Hinweis

Wenn eine geroutete S7-Verbindung über eine CPU geht, dann werden die dafür benötigten Verbindungsressourcen der CPU nicht in der Tabelle der Verbindungsressourcen angezeigt!

Anzeige der Verbindungsressourcen für HMI

Bei der HMI-Kommunikation ist die Belegung von Verbindungsressourcen in der CPU abhängig vom eingesetzten HMI-Gerät und von den Applikationen, in denen das HMI-Gerät eingesetzt wird. Ein Basic Panel benötigt beispielsweise für eine HMI-Verbindung bis zu zwei, ein Comfort Panel bis zu drei Verbindungsressourcen in der CPU.

Informationen über die Verfügbarkeit und Belegung von Verbindungsressourcen für HMI-Verbindungen finden Sie in der Offline-Sicht im Kontext des HMI-Geräts (im Inspektorfenster unter "Eigenschaften > Allgemein > Verbindungsressourcen").

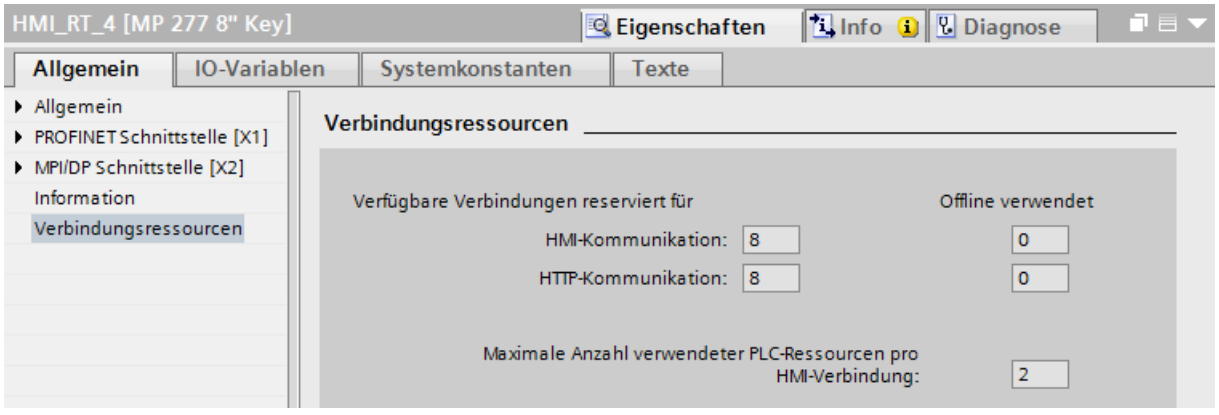

Im Bereich Verbindungsressourcen werden angezeigt:

- Anzahl der im HMI verfügbaren Verbindungen reserviert für HMI-Kommunikation und HTTP-Kommunikation
- Anzahl der offline im HMI verwendeten Verbindungsressourcen für HMI-Kommunikation und HTTP-Kommunikation
  Falls die Anzahl der maximal verfügbaren Verbindungsressourcen für ein HMI-Gerät überschritten ist, wird eine entsprechende Meldung von STEP 7 ausgegeben.
- "Maximale Anzahl verwendeter PLC-Ressourcen pro HMI-Verbindung": Dieser Parameter ist ein Faktor, der mit der Anzahl der offline verwendeten HMI-Verbindungen zu multiplizieren ist. Das Produkt ergibt die Anzahl der in der CPU belegten HMI-Ressourcen.

Falls die Anzahl der maximal verfügbaren Verbindungsressourcen für ein HMI-Gerät überschritten ist, wird eine entsprechende Meldung ausgegeben.

Anzeige der Verbindungsressourcen im Webserver

Sie können sich die Verbindungsressourcen nicht nur in STEP 7 anzeigen lassen, sondern auch mit einem Browser, der die entsprechende Seite des Webservers anzeigt. Informationen zur Anzeige der Verbindungsressourcen im Webserver finden Sie im Funktionshandbuch zum Webserver.

Beispiel für die Berechnung der Verbindungsressourcen

Einführung

In diesem Beispiel zur Tabelle mit der Anzeige der Verbindungsressourcen wird Ihnen gezeigt, wie sich die einzelnen Tabelleneinträge zusammensetzen.

Voraussetzung

- 2 x CPU 1516-3 PN/DP mit je einem CP 1543-1 und einem CM 1542-5 zur Erweiterung des Mengengerüsts als Verbindungspartner
- 1 x PC-System WinCC RT Professional mit CP 1616 onboard als Verbindungspartner für HMI-Verbindungen
- Die Geräte sind untereinander vernetzt.

Anlegen der Verbindungen

Nachdem die Verbindungspartner (CPUs, CPs, CMs, WinCC RT Professional) in der Netzsicht des Hardware- und Netzwerkeditors angelegt und vernetzt wurden, gehen Sie folgendermaßen vor:

1. Wechseln Sie in der Netzsicht in den Verbindungsmodus.
2. Legen Sie folgende Verbindungen an:
   - S7-Verbindungen: 2 x über CPU, 21 x über CM
   - TCP-Verbindungen: 2 x über CPU, 4 x über CM
   - ISO-Verbindung: 6 x über CP
   - ISO-on-TCP: 3 x über CPU, 4 x über CM
   - UDP-Verbindungen: 4 x über CPU, 106 x über CP
   - HMI-Verbindungen: 1 x über CPU, 2 x über CP

Ergebnis

In der Anzeige der Verbindungsressourcen sehen Sie außer den verfügbaren Ressourcen die benötigten Ressourcen für die angelegten Verbindungen.

In dem folgenden Bild sind die zur Verfügung stehenden Ressourcen der Geräte und Module hervorgehoben:

	Ressourcen der Station			Ressourcen...	Ressourcen...	Ressourcen...
	Reserviert		Dynami... ③	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Maximale Anzahl der Ressourcen:	① 10		246	② 128	118	118
	Maximum	Konfigu...	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	5	3	6	0
S7-Kommunikation:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Web-Kommunikation:	2	-	-	-	-	-
Sonstige Kommunikation:	-	-	-	-	-	-
Insgesamt verwendete Ressourc...		4	157	14	118	29
Verfügbare Ressourcen:		6	89	114	0	89

① Abhängig von der CPU, stehen dieser Station höchstens 256 Verbindungsressourcen zur Verfügung. 10 Ressourcen sind für bestimmte Verbindungstypen reserviert und 246 Ressourcen sind dynamisch verfügbar.

② Die CPU stellt 128 Verbindungsressourcen zur Verfügung, der CP und das CM stellen jeweils nochmal 118 Ressourcen zur Verfügung.

③ CPU, CP und CM stellen gemeinsam 128 + 118 + 118 = 364 Ressourcen zur Verfügung. Da über die Geräte und Module mehr Verbindungsressourcen angeboten werden, als die Station mit maximal 256 Verbindungsressourcen bedienen kann, wird ein Warnsymbol angezeigt.

In dem folgenden Bild sind die durch die angelegten Verbindungen verbrauchten Ressourcen hervorgehoben:

	Ressourcen der Station			Ressourcen...	Ressourcen...	Ressourcen...
	Reserviert		Dynami... ⚠	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Maximale Anzahl der Ressourcen:		10	246	128	118	118
	Maximum	Konfigu...	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	5	3	6	0
S7-Kommunikation:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Web-Kommunikation:	2	-	-	-	-	-
Sonstige Kommunikation:	-	-	-	-	-	-
Insgesamt verwendete Ressourc...		4	157	14	118	29
Verfügbare Ressourcen:		6	89	114	0	89

① Dies sind die über die HMI-Verbindungen belegten Ressourcen. Eine HMI-Verbindung verbraucht in diesem Beispiel 3 Verbindungsressourcen. Für die ganze Station werden 3 + 6 = 9 Verbindungsressourcen für die HMI-Kommunikation benötigt.

② Für die HMI-Kommunikation sind 4 Ressourcen reserviert.

③ Die benötigten 9 Ressourcen der HMI-Kommunikation teilen sich auf: Für die ersten 4 Ressourcen werden die bereits für HMI-Kommunikation reservierten Ressourcen verwendet. Die 5 weiteren benötigten Ressourcen werden aus den dynamischen Ressourcen genommen.

④ Dies sind die über die S7-Verbindungen belegten Ressourcen. Eine S7-Verbindung verbraucht 1 Verbindungsressource. Für die ganze Station werden 2 + 21 = 23 Verbindungsressourcen benötigt.

⑤ Für die S7-Kommunikation gibt es keine reservierten Ressourcen. Alle 23 benötigten Ressourcen kommen aus den dynamisch verfügbaren Ressourcen.

⑥ Dies sind die Ressourcen der Open User Communication (für TCP, FDL, ISO, ISO-on-TCP und UDP). Für die ganze Station werden 9 + 112 + 8 = 129 Verbindungsressourcen benötigt.

⑦ Für die Open User Communication gibt es keine reservierten Ressourcen. Alle 129 benötigten Ressourcen kommen aus den dynamisch verfügbaren Ressourcen.

Berechnung der Verbindungsressourcen

Über die verfügbaren und verbrauchten Ressourcen einzelner Geräte oder Module werden die noch freien und in Summe verwendeten Ressourcen berechnet:

	Ressourcen der Station			Ressourcen...	Ressourcen...	Ressourcen...
	Reserviert		Dynami...	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Maximale Anzahl der Ressourcen:		10	246	128	③ 118	118
	Maximum	Konfigu...	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	5	3	6	0
S7-Kommunikation:	0	-	23	① 2	0	21
Open User Communication:	0	-	129	9	112	8
Web-Kommunikation:	2	-	-	-	-	-
Sonstige Kommunikation:	-	-	-	-	-	-
Insgesamt verwendete Ressourc...		4	157	② 14	④ 118	29
Verfügbare Ressourcen:		6	89	114	⑤ 0	89

① Hier werden alle benötigten Ressourcen für die jeweiligen Kommunikationsarten der CPU angezeigt. 3 Ressourcen werden für die HMI-Kommunikation benötigt, 2 für die S7-Kommunikation und 9 für die Verbindungstypen der Open User Communication.

② Unter der Liste der benötigten Ressourcen für die Kommunikationsarten findet man die Anzahl der insgesamt benötigten Ressourcen. Für die CPU also 3 + 2 + 9 = 14 benötigte Ressourcen.

③ Der CP 1543-1 hat 118 verfügbare Verbindungsressourcen. Sind diese verbraucht, können keine weiteren Verbindungen für den CP angelegt werden.

④ Für den CP 1543-1 werden bereits 118 Verbindungsressourcen durch die 2 HMI-Verbindungen (je 3 Ressourcen = 6 Ressourcen) und 112 Ressourcen durch Verbindungen der offenen Kommunikation belegt.

⑤ Der CP 1543-1 hat keine freien Ressourcen mehr. Es können keine Verbindungen mehr für den CP 1543-1 angelegt werden.

Die Darstellung der noch freien und in Summe verwendeten Ressourcen findet sich auch für die gesamte Station:

	Ressourcen der Station			Ressourcen...	Ressourcen...	Ressourcen...
	Reserviert		Dynami... ⚠	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Maximale Anzahl der Ressourcen:		10	246	128	118	118
	Maximum	Konfigu...	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	5	3	6	0
S7-Kommunikation:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Web-Kommunikation:	2	-	-	-	-	-
Sonstige Kommunikation:	-	-	-	-	-	-
Insgesamt verwendete Ressourc...		4	157	14	118	29
Verfügbare Ressourcen:		6	89	114	0	89

① Für die einzelnen Kommunikationsarten in der ganzen Station verwendete (konfigurierte) reservierte und dynamische Verbindungsressourcen. Die Werte ergeben sich aus der Summe der verbrauchten Ressourcen bei den Geräten und Modulen (CPU, CP, CM).

② Für alle Kommunikationsarten in der ganzen Station verwendete (konfigurierte) reservierte und dynamische Verbindungsressourcen.

③ Insgesamt noch verfügbare reservierte und dynamische Verbindungsressourcen. Die Werte ergeben sich durch die maximal konfigurierbaren Ressourcen abzüglich der verwendeten Ressourcen.

Datenkonsistenz

Definition

Für die Übertragung von Daten ist die Datenkonsistenz eine wichtige Größe, die Sie bei der Projektierung einer Kommunikationsaufgabe berücksichtigen müssen. Geschieht das nicht, kann es zu Fehlfunktionen kommen.

Ein Datenbereich, der nicht durch konkurrierende Prozesse verändert werden kann, wird als konsistenter Datenbereich bezeichnet. Das heißt, ein in sich zusammengehöriger Datenbereich, der größer ist als die Maximalgröße des konsistenten Datenbereichs, kann zu einem Zeitpunkt teilweise aus neuen und aus alten Daten bestehen.

Eine Inkonsistenz kann entstehen, wenn eine Anweisung für Kommunikation z. B. durch einen Prozessalarm-OB mit höherer Priorität unterbrochen wird. Dadurch wird auch die Übertragung des Datenbereichs unterbrochen. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die noch nicht von der Kommunikationsanweisung verarbeitet wurden, stammen die übertragenen Daten aus unterschiedlichen Zeitpunkten.

Das folgende Bild zeigt einen Datenbereich, der kleiner ist als die Maximalgröße des konsistenten Datenbereichs. In diesem Fall wird bei der Übertragung des Datenbereichs sichergestellt, dass während des Datenzugriffs keine Unterbrechung durch das Anwenderprogramm erfolgt und damit die Daten nicht geändert werden.

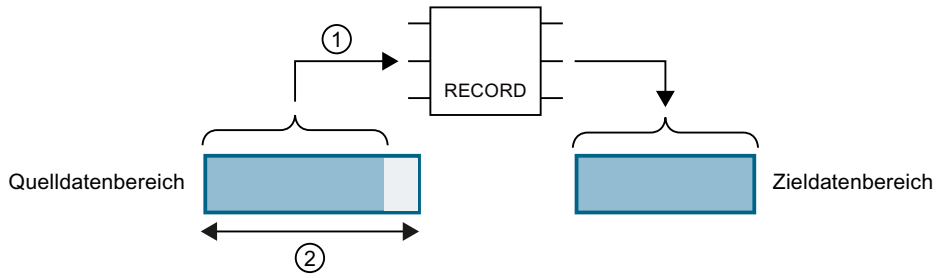

① Der Quelldatenbereich ist kleiner als die Maximalgröße des konsistenten Datenbereichs (②). Die Anweisung überträgt die Daten zusammenhängend in den Zieldatenbereich.

② Maximalgröße konsistenter Datenbereich

Bild 1-1 Konsistente Übertragung von Daten

Das folgende Bild zeigt einen Datenbereich, der größer ist als die Maximalgröße des konsistenten Datenbereichs. In diesem Fall können die Daten während einer Unterbrechung der Datenübertragung verändert werden. Eine Unterbrechung entsteht z.B. auch, wenn der Datenbereich in mehreren Teilen übertragen werden muss. Werden die Daten während der Unterbrechung verändert, dann stammen die übertragenen Daten aus unterschiedlichen Zeitpunkten.

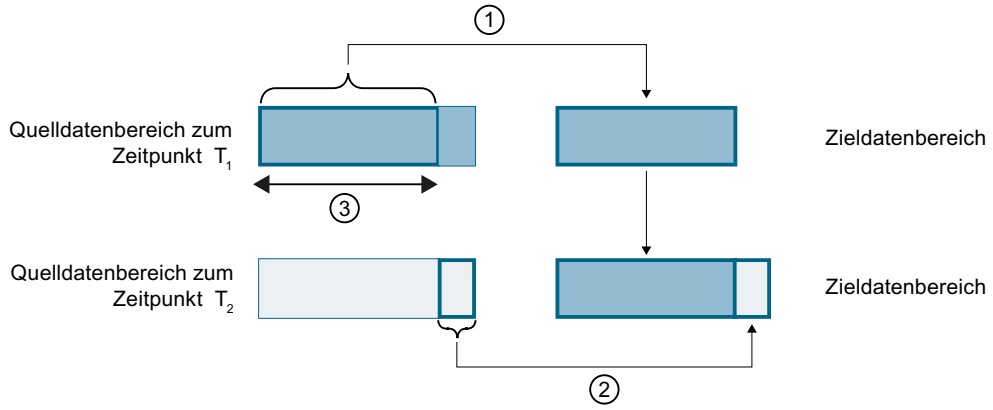

① Der Quelldatenbereich ist größer als die Maximalgröße des konsistenten Datenbereichs (③). Zum Zeitpunkt T1 überträgt die Anweisung nur so viel Daten vom Quelldatenbereich in den Zieldatenbereich, wie in den konsistenten Datenbereich reinpassen.

② Zum Zeitpunkt T2 überträgt die Anweisung den Rest des Quelldatenbereichs in den Zieldatenbereich. Nach der Übertragung liegen im Zieldatenbereich Daten aus unterschiedlichen Zeitpunkten. Wenn sich die Daten im Quelldatenbereich in der Zwischenzeit geändert haben, kann eine Inkonsistenz entstehen.

③ Maximalgröße konsistenter Datenbereich

Bild 1-2 Übertragung von Daten größer dem maximalen Konsistenzbereich

Beispiel für eine Inkonsistenz

Das folgende Bild zeigt ein Beispiel für Veränderung von Daten während der Übertragung. Im Zieldatenbereich liegen Daten aus unterschiedlichen Zeitpunkten.

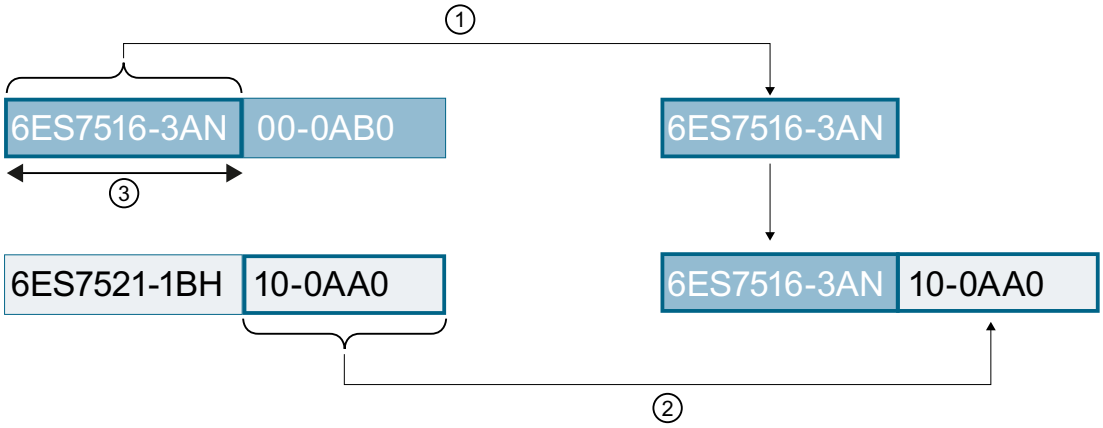

Bild 1-3 Beispiel: Veränderung von Daten während der Übertragung

Systemspezifische maximale Datenkonsistenz für S7-1500:

Eine Inkonsistenz tritt nicht auf, wenn die systemspezifische Maximalgröße der konsistenten Daten eingehalten wird. Bei S7-1500 werden die Kommunikationsdaten in Blöcken bis maximal 512 Byte während des Programmzyklus konsistent in/aus dem Anwenderspeicher kopiert. Für alle größeren Datenbereiche wird keine Datenkonsistenz garantiert. Ist eine definierte Datenkonsistenz gefordert, so dürfen die Kommunikationsdaten im Anwenderprogramm der CPU nicht größer als 512 Byte sein. Auf diese Datenbereiche können Sie dann z. B. von einem HMI-Gerät mit Lesen/Schreiben von Variablen konsistent zugreifen.

Wenn mehr Daten als die systemspezifische Maximalgröße konsistent übertragen werden sollen, dann müssen Sie selbst durch entsprechende Maßnahmen im Anwenderprogramm die Datenkonsistenz sicherstellen.

Datenkonsistenz sicherstellen

Einsatz von Anweisungen für Zugriff auf gemeinsame Daten:

Existieren im Anwenderprogramm Kommunikationsanweisungen, welche auf gemeinsame Daten zugreifen, z. B. TSEND/TRCV, können Sie den Zugriff auf diesen Datenbereich z. B. über den Parameter "DONE" selbst koordinieren. Die Datenkonsistenz der Datenbereiche, die

mit einer Anweisung für Kommunikation übertragen werden, kann deshalb im Anwenderprogramm sichergestellt werden.

Hinweis

Maßnahmen im Anwenderprogramm

Um Datenkonsistenz zu erreichen, können Sie die zu übertragenden Daten auf einen separaten Datenbereich (z. B. globaler Datenbaustein) umkopieren. Während das Anwenderprogramm weiterhin mit den Originaldaten arbeitet, können Sie die im separaten Datenbereich gespeicherten Daten konsistent mit der Kommunikationsanweisung übertragen.

Verwenden Sie für das Umkopieren nicht unterbrechbare Anweisungen, wie UMOVE_BLK oder UFILL_BLK. Diese Anweisungen gewährleisten eine Datenkonsistenz bis 16 KByte.

Einsatz von Anweisungen PUT/GET bzw. Schreiben/Lesen über HMI-Kommunikation:

Bei S7-Kommunikation mit den Anweisungen PUT/GET bzw. Schreiben/Lesen über HMI-Kommunikation müssen Sie bereits bei der Programmierung bzw. Projektierung die Größe der konsistenten Datenbereiche berücksichtigen. Im Anwenderprogramm einer S7-1500 als Server ist keine Anweisung vorhanden, die die Datenübertragung im Anwenderprogramm koordinieren kann. Die über PUT/GET-Anweisungen ausgetauschten Daten aktualisiert die S7-1500 während der Laufzeit des Anwenderprogramms. Es gibt keinen Zeitpunkt innerhalb der Bearbeitung des zyklischen Anwenderprogramms, an dem die Daten konsistent ausgetauscht werden. Die Länge des zu übertragenden Datenbereichs sollte kleiner sein als 512 Bytes.

Weitere Informationen

- Die max. Anzahl konsistenter Daten finden Sie auch in den Gerätehandbüchern von CPU, CM bzw. CP in den Technischen Daten.
- Weitere Informationen zur Datenkonsistenz finden Sie in der Beschreibung der Anweisungen in der Online-Hilfe STEP 7.

1.3.2.2 Verbindungen - Bedeutung und Auswahl

Einführung zum Projektieren von Verbindungen

Wozu Verbindungen einrichten?

Eine Kommunikationsverbindung oder kurz Verbindung definiert eine logische Zuordnung zweier Kommunikationspartner zur Ausführung von Kommunikationsdiensten.

Kommunikationsverbindungen können Sie projektieren oder unter bestimmten Voraussetzungen auch programmgesteuert einrichten.

In diesem Kapitel lesen Sie, wie Sie Verbindungen definieren, welche Besonderheiten Sie beachten müssen und welche Kommunikationsanweisungen Sie im Anwenderprogramm einsetzen können.

Definition

Eine Verbindung legt Folgendes fest:

- Beteiligte Kommunikationspartner
- Typ der Verbindung (z. B. S7-, ISO-on-TCP-, ISO-Transport, FDL-, UDP-, E-Mail- oder TCP-Verbindung)
- Spezielle Eigenschaften wie z. B.
  - Das Aufbauverhalten
    Eine Verbindung kann permanent aufgebaut bleiben oder bei Bedarf auf- und abgebaut werden.
  - Das Senden von Betriebszustandsmeldungen
- Kommunikationsweg
  - innerhalb der Geräte (CPU - CP)
  - im Netz (Subnetze / Router)

Wissenswertes zur Verbindungsprojektierung

Bei der Verbindungsprojektierung wird pro Verbindung eine eindeutige lokale Kennung vergeben, die "Lokale ID". Nur diese lokale ID wird bei der Parametrierung der Kommunikationsbausteine benötigt. Die lokale ID ist den Adressen des lokalen und des fernen Verbindungspartners eindeutig zugeordnet.

Wenn es sich bei einem Verbindungsendpunkt um die Applikation einer PC-Station handelt, dann wird zur Verbindungsidentifikation als "Lokale ID" der Verbindungsname verwendet.

Nach dem Öffnen des Projektes wird die projektweite Verbindungstabelle angezeigt. Wenn Sie im Kontextmenü der Verbindungstabelle das Optionskästchen "Alle Verbindungen anzeigen" deaktivieren, werden nur noch objektspezifische Verbindungen ausgewählter Objekte angezeigt. Für folgende Objekte können die angelegten Verbindungen separat angezeigt werden:

- Programmierbare Baugruppen, die Endpunkt einer Verbindung sein können
- Kommunikationsbaugruppen (CPs), die auf dem Verbindungsweg genutzt werden
- Subnetze

Die Adressen des lokalen und des fernen Verbindungspartners werden durch die Vernetzung der Geräte vorgegeben und den Verbindungen bei der Projektierung zugeordnet. Eine Ausnahme hiervon bilden folgende Verbindungen:

- Freie UDP-Verbindung
  Bei der freien UDP-Verbindung wird die Adresse an der Kommunikationsschnittstelle im Anwenderprogramm angegeben.
- Programmierte Kommunikationsverbindung

Hinweis

Der Begriff "Verbindung" wird unter der SIMATIC-Kommunikation auch bei UDP verwendet. Der Grund dafür ist, dass bei der Projektierung die Kommunikationspartner einander zugeordnet und somit logisch "verbunden" werden. Tatsächlich erfolgt bei UDP kein expliziter Verbindungsaufbau zwischen den Kommunikationspartnern.

Vernetzung und Wegewahl bestimmen den Projektierungsstatus

Abhängig davon, ob die Verbindungspartner im Projekt vorhanden und vernetzt sind, können die Adressparameter der Verbindungsendpunkte vollständig oder nur teilweise spezifiziert sein.

- voll spezifiziert
  - Die Adressen und die Netzparameter des lokalen und des fernen Kommunikationspartners sind festgelegt.
  - Die Kommunikationsverbindung ist nach dem Laden der Verbindungsparameter in das Gerät betriebsfähig.
- unspezifiziert (teilweise spezifiziert)
  Nur der lokale Kommunikationspartner ist festgelegt. Der Partner für den gewählten Verbindungstyp ist entweder nicht vernetzt, liegt außerhalb des Projekts oder soll wegen einer speziellen Handhabung nicht festgelegt werden.
  Anwendungsfall:
  - Die Hardware-Konfiguration ist noch unvollständig, so dass die vollständige Vernetzung der Geräte noch nicht möglich ist.
  - Eine Empfangsbereitschaft für beliebige (unspezifizierte) Kommunikationspartner soll hergestellt werden.

  Die Kommunikationsverbindung ist nach dem Laden der Verbindungsparameter in das Gerät bedingt betriebsfähig.

Programmierte Kommunikationsverbindungen

In bestimmten Anwendungsbereichen ist es vorteilhaft, die Kommunikationsverbindungen nicht über die Projektierung statisch einzurichten. Alternativ besteht daher die Möglichkeit, bestimmte Verbindungstypen über eine spezifische Applikation programmgesteuert und damit bei Bedarf dynamisch einzurichten.

Merkmale der Verbindungsprojektierung

Mit der Netzsicht finden Sie eine übersichtliche Plattform zur Projektierung von Kommunikationsverbindungen zwischen projektierten Geräten.

Wesentliche Funktionen

- Grafische Verbindungsprojektierung in der Netzsicht
  Mit wenigen Mausklicks legen Sie eine Verbindung grafisch an.
- Hervorheben möglicher Verbindungspartner
  Sie wählen einen Verbindungstyp - die hierfür infrage kommenden Kommunikationspartner werden automatisch hervorgehoben.
- Automatische Suche von Verbindungswegen
  Sie veranlassen die automatische Zuweisung des bestmöglichen Verbindungswegs oder die Auswahl von alternativen Verbindungswegen.
- Automatische Übernahme von Konfigurationen
  Wenn Sie Baugruppen nachträglich konfigurieren und vernetzen, werden bereits projektierte, von der Baugruppe unterstützte Verbindungen automatisch zugewiesen.
- Komfortable Auswahl der Verbindungsanzeige
  Wenn Sie ein Objekt in der Netzsicht selektieren, z. B. ein Subnetz oder die Schnittstelle einer kommunikationsfähigen Baugruppe, wird Ihnen in der Verbindungstabelle die zugehörige projektierte Kommunikationsverbindung angezeigt.
- Berücksichtigung der verfügbaren Ressourcen
  Soweit Verbindungs-Ressourcen für einen Verbindungstyp benötigt werden, wird eine gegebene Begrenzung beim Anlegen von Verbindungen berücksichtigt. Die maximale Anzahl an Verbindungsressourcen variiert sowohl bei den CPU-Gerätetypen als auch bei den CP-Gerätetypen.

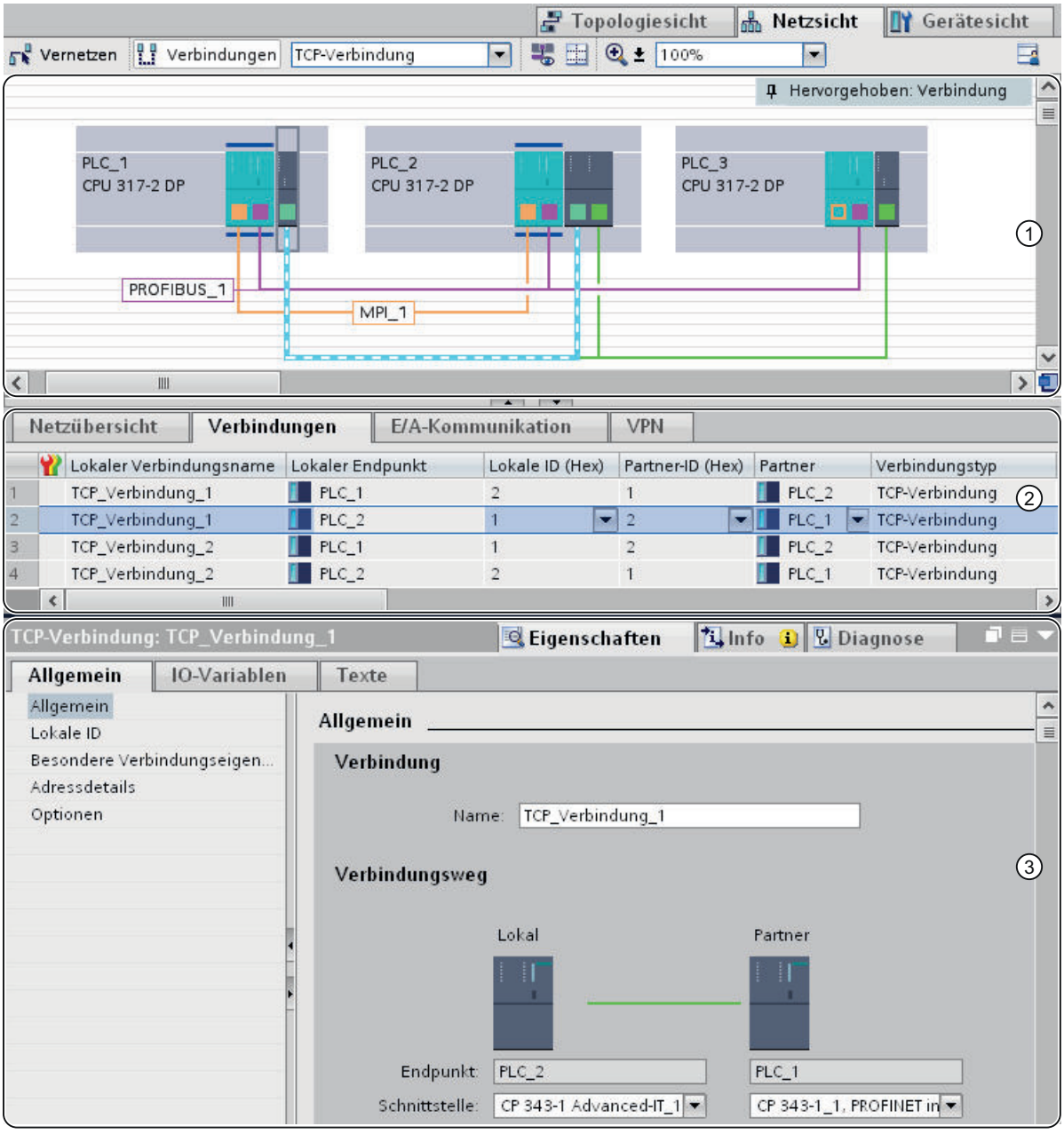

① Verbindungsanzeige in der Netzsicht

② Verbindungstabelle

③ Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Siehe auch

Überblick über die Verbindungsdiagnose (Seite 3536)

Einsatzbereiche der Verbindungstypen

Einleitung

Die folgenden Abschnitte geben einen kurzen Überblick der Verbindungstypen, die Sie für die Kommunikation je nach Gerätetyp und Einsatzzweck nutzen können.

Weitere Informationen

Beachten Sie für weitere Informationen über die Kommunikationsmöglichkeiten in SIMATIC und den Verwendungszweck der Verbindungstypen das Handbuch über die Kommunikation mit SIMATIC.

S7-Verbindungen

S7-Verbindungen bieten u. a. folgende Merkmale:

- Verbindungstyp in allen S7-Geräten nutzbar;
- Auf folgenden Subnetz-Typen einsetzbar: MPI-, PROFIBUS, Industrial Ethernet;
- Bei Verwendung der SFBs BSEND/BRCV: Sicheres Übertragen von Daten zwischen SIMATIC S7-400/1500-Stationen; z. B. Austausch von Datenbaustein-Inhalten (bis 64 kByte);
- Mit CPU 317-2 PN/DP sowie mit CPU 31x und einem CP ist bei Verwendung von FBs BSEND/BRCV ebenfalls sicheres Übertragen von Daten zu S7-300, S7-400 und S7-1500 möglich;
- Bei Verwendung der SFBs USEND/URCV: Schnelles, ungesichertes Übertragen von Daten unabhängig von der zeitlichen Bearbeitung des Kommunikationspartners; z. B. für Betriebs- und Wartungsmeldungen;
- Quittierung der Datenübertragung vom Kommunikationspartner auf Schicht 7 des ISO-Referenzmodells.

FDL-Verbindung

PROFIBUS-FDL (Fieldbus Data Link) ist gekennzeichnet durch folgende Merkmale:

- Übertragung von Daten zu einem Kommunikationspartner (z. B. SIMATIC S5 oder PC), der das Senden bzw. Empfangen entsprechend der SDA-Funktion (Send Data with Acknowledge) unterstützt;
- Empfang der Daten wird vom FDL-Dienst des Kommunikationspartners durch eine Quittung bestätigt;
- Nutzbarer Subnetztyp: PROFIBUS;
- Entspricht der Norm EN 50170 Vol.2 PROFIBUS;
- Lässt sich der Schicht 2 des ISO-Referenzmodells zuordnen;
- Schnittstelle im S7-Anwenderprogramm: SEND/RECEIVE;
- FDL-Dienste als C-Funktionen auf dem PC verfügbar.

ISO-Transportverbindung

Die ISO-Transportverbindung ist gekennzeichnet durch folgende Merkmale:

- "Daten-Blockung" geeignet für große Datenmengen;
- Kommunikation zu einem Partner (z. B. SIMATIC S5 oder PC), der das Senden bzw. Empfangen von Daten gemäß ISO-Transport unterstützt;
- Nutzbarer Subnetztyp: Industrial Ethernet;
- Empfang der Daten wird vom ISO-Transportdienst des Kommunikationspartners durch eine Quittung bestätigt;
- Schnittstelle im S7-Anwenderprogramm: SEND/RECEIVE und FETCH/WRITE;
- ISO-Transportdienst (ISO 8073 class 4) entspricht der Schicht 4 des ISO-Referenzmodells;
- ISO-Transportdienste als C-Funktionen auf dem PC verfügbar.

ISO-on-TCP-Verbindung

Die ISO-on-TCP-Verbindung ist gekennzeichnet durch folgende Merkmale:

- Entspricht dem Standard TCP/IP (Transmission Control Protocol/Internet Protocol) mit der Erweiterung RFC 1006 gemäß der Schicht 4 des ISO-Referenzmodells. RFC 1006 beschreibt, wie die Dienste von ISO Schicht 4 auf TCP abgebildet werden können;
- Kommunikation zu einem Partner (z. B. PC oder Fremdsystem), der das Senden bzw. Empfangen von Daten gemäß ISO-on-TCP unterstützt;
- Nutzbarer Subnetztyp: Industrial Ethernet;
- Empfang der Daten wird durch eine Quittung bestätigt;
- Schnittstelle im STEP 7-Anwenderprogramm: SEND/RECEIVE und FETCH/WRITE;
- ISO-on-TCP-Dienste als C-Funktionen auf dem PC verfügbar.

TCP-Verbindung

Die TCP-Verbindung ist gekennzeichnet durch folgende Merkmale:

- Entspricht dem Standard TCP/IP (Transmission Control Protocol/Internet Protocol);
- Kommunikation zu einem Partner (z. B. PC oder Fremdsystem), der das Senden bzw. Empfangen von Daten gemäß TCP/IP unterstützt;
- Nutzbarer Subnetztyp: Industrial Ethernet;
- Schnittstelle im STEP 7-Anwenderprogramm: SEND/RECEIVE und FETCH/WRITE;
- TCP/IP-Implementierung des PC-Betriebssystem in der Regel nutzbar.

UDP-Verbindung

Die UDP-Verbindung (User Datagramm Protocol) ist gekennzeichnet durch folgende Merkmale:

- Nutzbarer Subnetztyp: Industrial Ethernet (TCP/IP-Protokoll);
- Ungesicherte Übertragung zusammenhängender Datenblöcke zwischen zwei Teilnehmern;
- Schnittstelle im S7-Anwenderprogramm: SEND/RECEIVE.

E-Mail Verbindung

Die E-Mail-Verbindung ist gekennzeichnet durch folgende Merkmale:

- Nutzbarer Subnetztyp: Industrial Ethernet (TCP/IP-Protokoll);
- Ermöglicht z. B. das Versenden von z. B. Prozessdaten aus Datenbausteinen via E-Mail über einen CP mit IT-Funktionalität (IT-CP);
- Mit der E-Mail-Verbindung ist der Mail Server festgelegt, über den sämtliche von einem IT-CP gesendeten E-Mails zugestellt werden.

FTP-Verbindung (speziell eingerichtete TCP-Verbindung)

Die FTP-Verbindung verwenden Sie für die Abwicklung von FTP-Auftragssequenzen.

Mit den File-Transfer-Funktionen (FTP) bieten Ihnen die CPs mit IT-Funktionalität die Möglichkeit, Dateien zu und von den S7-Geräten zu übertragen.

HMI Verbindung

AUTOHOTSPOT

Wahl der Verbindungstypen

Wahl des Verbindungstyps

Welchen Verbindungstyp Sie wählen, ist neben den geforderten Übertragungseigenschaften abhängig vom Subnetz und dem Übertragungsprotokoll, über das die Verbindung aufgebaut wird. Zusätzlich grenzt der Gerätetyp am jeweiligen Verbindungsendpunkt die Verbindungstypen ein.

Welche Anweisungen oder Dienste Sie an der Schnittstelle (Endpunkte der Verbindung) einsetzen, hängt vom Verbindungstyp und vom Gerätetyp ab.

Die folgende Tabelle stellt diese Abhängigkeiten dar:

Verbindungstyp	Subnetztyp / Protokoll	Gerätetyp: Verbindung zwischen SIMATIC ...	Schnittstelle
S7-Verbindung	MPI, PROFIBUS, Industrial Ethernet	S7 - S7, S7 - PG/PC, S7 - PG/PC mit WinCC, PG/PC - PG/PC S7 - Partner in anderem Projekt (S7, PG/PC mit WinCC)	**Anweisungen für S7:** USEND, URCV, BSEND, BRCV, GET, PUT, START, STOP, RESUME, STATUS, USTATUS **PG/PC:** OPC-Dienste oder Programmierschnittstellen
FDL-Verbindung	PROFIBUS (FDL-Protokoll)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - Fremdgerät, PG/PC - PG/PC S7 - Partner in anderem Projekt (S7, S5, PG/PC, Fremdgerät)	**Anweisungen für S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV **PG/PC:** OPC-Dienste
ISO-Transportverbindung	Industrial Ethernet (ISO-Transportprotokoll)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - Fremdgerät, PG/PC - PG/PC S7 - unspezifiziert S7 - Partner in anderem Projekt (S7, S5, PG/PC, Fremdgerät, unspezifiziert)	**Anweisungen für S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK **PG/PC:** OPC-Dienste
ISO-on-TCP-Verbindung	Industrial Ethernet (TCP/IP-Protokoll)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - Fremdgerät, PG/PC - PG/PC, S7 - unspezifiziert S7 - Partner in anderem Projekt (S7, S5, PG/PC, Fremdgerät, unspezifiziert)	**Anweisungen für S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK **PG/PC:** OPC-Dienste
TCP-Verbindung	Industrial Ethernet (TCP/IP-Protokoll)	S7 - S7, S7 - S5, S7 - PC/PG, PG/PC - PG/PC S7 - Fremdgerät, S7 - unspezifiziert S7 - Partner in anderem Projekt (S7, S5, PG/PC, Fremdgerät, unspezifiziert)	**Anweisungen für S7:** AG_SEND*, AG_RECV*, AG_LSEND**, AG_LRECV** AG_LOCK AG_UNLOCK **PG/PC:** OPC-Dienste
UDP-Verbindung	Industrial Ethernet (TCP/IP-Protokoll)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - Fremdgerät, S7 - unspezifiziert S7 - Partner in anderem Projekt (S7, S5, PG/PC, Fremdgerät, unspezifiziert)	**Anweisungen für S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV

Verbindungstyp	Subnetztyp / Protokoll	Gerätetyp: Verbindung zwischen SIMATIC ...	Schnittstelle
E-Mail-Verbindung	Industrial Ethernet (TCP/IP-Protokoll)	S7 - unspezifiziert (S7 - Mail-Server), S7 - PC/PG (S7 - Mail-Server)	**Anweisungen für S7:** AG_SEND, AG_LSEND
FTP-Verbindung (speziell eingerichtete TCP-Verbindung)	Industrial Ethernet (TCP/IP-Protokoll)	S7 - unspezifiziert	**Anweisungen für S7:** FTP_CONNECT, FTP_STORE, FTP_RETRIEVE, FTP_DELETE, FTP_QUIT

Verbindung zu Broadcast und Multicast-Teilnehmern

Hinweis

Für spezielle Verbindungstypen gibt es die Möglichkeit, nicht nur einen Verbindungspartner auszuwählen, sondern mehrere (Broadcast- und Multicast-Teilnehmer). Die Verbindungspartner "Zu allen Broadcast-..." bzw. "... Multicast-Teilnehmern" können Sie im Dialogfeld zum Eingeben einer neuen Verbindung auswählen.

- Sie können eine Verbindung zu "allen Broadcast-Teilnehmern" (gleichzeitiges Senden zu allen Broadcast-Empfängern) einrichten für die Verbindungstypen FDL- und UDP-Verbindung.
- Sie können eine Verbindung zu "allen Multicast-Teilnehmern" (gleichzeitiges Senden zu mehreren Teilnehmern) einrichten für die Verbindungstypen FDL- und UDP-Verbindung.

Wahl der Verbindungstypen

Wahl des Verbindungstyps

Sie müssen sich in Abhängigkeit von Ihrer Automatisierungsaufgabe für einen Kommunikationsdienst entscheiden. Die Wahl des Kommunikationsdienstes beeinflusst Folgendes:

- Die Funktionalität, die zur Verfügung steht
- Den Zeitpunkt des Verbindungsaufbaus
- Die Tätigkeiten, die Sie ausführen müssen (z. B. Projektieren von Verbindungen, Programmieren von Anweisungen)

Die folgende Tabelle gibt Ihnen eine Übersicht über die zur Verfügung stehenden Kommunikationsdienste.

Kommunikationsdienst	Funktionalität	Über Schnittstelle:		
		PN	DP	serielle
PG-Kommunikation	Zur Inbetriebnahme, Test, Diagnose	X	X	-
HMI-Kommunikation	Zum Bedienen und Beobachten	X	X	-

Kommunikationsdienst	Funktionalität	Über Schnittstelle:		
		PN	DP	serielle
E/A-Kommunikation	Datenaustausch zwischen z. B.: • IO-Controllern und IO-Devices • DP-Master und DP-Slave	X	X	-
S7-Kommunikation	Datenaustausch Client/Server bzw. Client/Client: Verbindungsprojektierung erforderlich Über Anweisungen PUT/GET, BSEND/BRCV oder USEND/URCV	X	X	-
Offene Kommunikation über TCP/IP	Datenaustausch über PROFINET mit TCP/IP-Protokoll Mit Verbindungsprojektierung: Anweisungen TSEND_C/TRCV_C oder TSEND/TRCV Nur über Anwenderprogramm: Anweisungen TCON, TSEND, T_RCV, T_DISCON oder TSEND_C und TRCV_C	X	-	-
Offene Kommunikation über ISO-on-TCP	Datenaustausch über PROFINET mit ISO-on-TCP-Protokoll Mit Verbindungsprojektierung: Anweisungen TSEND_C/TRCV_C oder TSEND/TRCV Nur über Anwenderprogramm: Anweisungen TCON, TSEND, T_RCV, T_DISCON oder TSEND_C und TRCV_C	X	-	-
Offene Kommunikation über UDP	Datenaustausch über PROFINET mit UDP-Protokoll Mit Verbindungsprojektierung: Anweisungen TSEND_C/TRCV_C oder TSEND/TRCV Nur über Anwenderprogramm: Anweisungen TCON, TSEND, T_RCV, T_DISCON oder TSEND_C und TRCV_C	X	-	-
Offene Kommunikation über ISO (nur CPs mit PN-Schnittstelle)	Datenaustausch über PROFINET mit ISO-Protokoll Mit Verbindungsprojektierung: Anweisungen TSEND_C/TRCV_C oder TSEND/TRCV Nur über Anwenderprogramm: Anweisungen TCON, TSEND, T_RCV, T_DISCON oder TSEND_C und TRCV_C	X	-	-
Offene Kommunikation über Email	Datenaustausch über Email Nur über Anwenderprogramm: Anweisung TMAIL_C	X	-	-
Offene Kommunikation über FTP (nur CPs mit PN-Schnittstelle)	Datenaustausch über FTP (File Transfer Protocol) Nur über Anwenderprogramm: Anweisung FTP_CMD	X	-	-
Offene Kommunikation über FDL (nur CMs mit DP-Schnittstelle)	Datenaustausch über PROFIBUS mit FDL-Protokoll Mit Verbindungsprojektierung: Anweisungen TSEND_C/TRCV_C oder TSEND/TRCV Nur über Anwenderprogramm: Anweisungen TCON, TSEND, T_RCV, T_DISCON oder TSEND_C und TRCV_C	-	X	-

Kommunikationsdienst	Funktionalität	Über Schnittstelle:		
		PN	DP	serielle
Punkt-zu-Punkt-Kopplung	Datenaustausch über Punkt-zu-Punkt mit Freeport-, 3964 (R)-, USS- oder Modbus-Protokoll Über Anweisungen SEND_P2P/RCV_P2P bzw. MB_COMM_LOAD, MB_MASTER , MB_SLAVE	-	-	X
S7-Routing	Datenaustausch über Netzwerkgrenzen, z. B. zu Parametrier-, Test- und Diagnosezwecken	X	X	-
Datensatz-Routing	Erweiterung des S7-Routing, z. B. zur Parametrierung und Diagnose von Feldgeräten am PROFIBUS durch ein Engineering-System an einer PROFINET-Schnittstelle	X	X	-
Webserver	Datenaustausch über das Internet , z. B. zur Diagnose	X	-	-
SNMP (Simple Network Management Protokoll)	Datenaustausch über Standardprotokoll SNMP, zur Netzwerkdiagnose und -parametrierung	X	-	-
Uhrzeitsynchronisation	über NTP (Network Time Protocol); CPU ist Client	X	-	-
	CPU ist Uhrzeitmaster oder Uhrzeitslave	-	X	-

Verbindung zu Broadcast und Multicast-Teilnehmern

Hinweis

Für spezielle Verbindungstypen gibt es die Möglichkeit, nicht nur einen Verbindungspartner auszuwählen, sondern mehrere (Broadcast- und Multicast-Teilnehmer). Die Verbindungspartner "Zu allen Broadcast-..." bzw. "... Multicast-Teilnehmern" können Sie im Dialogfeld zum Eingeben einer neuen Verbindung auswählen.

- Sie können eine Verbindung zu "allen Broadcast-Teilnehmern" (gleichzeitiges Senden zu allen Broadcast-Empfängern) einrichten für die Verbindungstypen FDL- und UDP-Verbindung.
- Sie können eine Verbindung zu "allen Multicast-Teilnehmern" (gleichzeitiges Senden zu mehreren Teilnehmern) einrichten für die Verbindungstypen FDL- und UDP-Verbindung.

Anweisungen für Kommunikationsaufgaben

Schnittstelle zum Anwenderprogramm

Für Ihre Kommunikations-Schnittstelle im Anwenderprogramm von SIMATIC S7-300/400/1500 stehen spezielle Anweisungen zur Verfügung. Sie erhalten hier eine Übersicht über die Anweisungen für die einzelnen Verbindungstypen.

Für die S7-1500 werden die Anweisungen PUT/GET verwendet. Siehe die folgenden Tabellen für die Anweisungen der S7-300/400.

Anweisungen für S7-Verbindungen

Die den Anweisungen zugeordneten Systemfunktionsbausteine sind in den CPUs der S7-400 integriert.

Für S7-300 gibt es bei neueren CPUs und CPs die Möglichkeit, über die Schnittstelle des CP die S7-Kommunikation aktiv (d. h. als Client) zu betreiben. Die Anweisungen haben dieselbe Bezeichnung wie die SFBs der S7-400, müssen aber im Anwenderprogramm der S7-300 CPU zyklisch aufgerufen werden.

Der CP muss die Client-Funktion für S7-Kommunikation unterstützen.

Eine S7-300 CPU mit integrierter PROFINET-Schnittstelle ist ebenfalls als Client für die S7-Kommunikation projektierbar. Es kommen dieselben Bausteine zur Anwendung wie im oben erwähnten Fall für die S7-300 mit CP. Die Client-Funktionalität steht nur an der PROFINET-Schnittstelle zur Verfügung.

Bezeichnung	Kurzbeschreibung
USEND URCV	Unkoordinierter Datenaustausch über einen Sende- und Empfangs-SFB. Max. Länge SFB 8/9: 440 Bytes aufgeteilt in 4x100 Bytes. Max. Länge FB 8/9: 160 Bytes.
BSEND BRCV	Austausch von Datenblöcken variabler Länge zwischen einer Sende-SFB und einem Empfangs-SFB. Max. Länge SFB 12/13: 64 kByte Max. Länge FB 12/13: 32 kByte
GET	Aus einem entfernten (remoten) Gerät Daten lesen Max. Länge SFB 14: 400 Bytes aufgeteilt in 4x100 Bytes Max. Länge FB 14: 160
PUT	In ein remotes Gerät Daten schreiben Max. Länge SFB 15: 400 Bytes aufgeteilt in 4x100 Bytes Max. Länge FB 15: 160
START	In einem remoten Gerät Neustart (Warmstart) durchführen
STOP	Ein remotes Gerät in den Zustand "STOP" versetzen
RESUME	In einem remoten Gerät Wiederanlauf durchführen
STATUS	Gezielte Abfrage des Status eines remoten Geräts
USTATUS	Empfang von Statusmeldungen remoter Geräte
CONTROL	Den Zustand der Verbindung, die zu einer SFB-Instanz gehört, abfragen
C_CNTRL	Den Zustand einer Verbindung abfragen (für S7-300-CPUs)

Anweisungen für S7-Kommunikation

Anweisungen für Punkt-zu-Punkt-Verbindungen

Für den Verbindungstyp Punkt-zu-Punkt-Verbindung können Sie die SFBs BSEND, BRCV, GET, PUT und STATUS einsetzen (siehe obige Tabelle).

Außerdem ist der SFB PRINT einsetzbar:

Bezeichnung	Kurzbeschreibung
PRINT	Daten an einen Drucker senden

Anweisungen für FDL-, ISO-on-TCP-, UDP und TCP-Verbindung sowie E-Mail-Verbindung

Bezeichnung	Kurzbeschreibung
AG_SEND	Sendet Daten über eine projektierte Verbindung zum Kommunikationspartner (bei älteren CPs und bei CPs für S7-400 gilt für diesen FC die Beschränkung: <= 240 Byte).
AG_RECV	Empfängt Daten über eine projektierte Verbindung vom Kommunikationspartner (bei älteren CPs und bei CPs für S7-400 gilt für diesen FC die Beschränkung:<= 240 Byte, nicht E-Mail).
AG_LSEND	Sendet Daten über eine projektierte Verbindung zum Kommunikationspartner.
AG_LRECV	Empfängt Daten über eine projektierte Verbindung vom Kommunikationspartner (nicht E-Mail).
AG_LOCK	Sperren des externen Datenzugriffs mittels FETCH/WRITE (nicht bei UPD, E-Mail).
AG_UNLOCK	Freigeben des externen Datenzugriffs mittels FETCH/WRITE (nicht bei UPD, E-Mail).
AG_CNTRL	Verbindungen über das Anwenderprogramm diagnostizieren.

AUTOHOTSPOT

AUTOHOTSPOT

Anweisungen für FTP-Dienste über TCP-Verbindungen (FTP-Verbindung)

Bezeichnung	Kurzbeschreibung
FTP_CONNECT	FTP-Verbindung aufbauen
FTP_STORE	Datenbaustein (File-DB) vom FTP-Client zum FTP-Server übertragen.
FTP_RETRIEVE	Datei vom FTP-Server zum FTP-Client übertragen.
FTP_DELETE	Datei auf dem FTP-Server löschen.
FTP_QUIT	Die über die ID benannte FTP-Verbindung abbauen.

AUTOHOTSPOT

Anweisungen für programmierte offene Kommunikationsdienste mit ISO-on-TCP-, UDP und TCP-Verbindung sowie E-Mail-Verbindung über CPs

Bezeichnung	Kurzbeschreibung
IP_CONFIG	Programmgesteuert Kommunikationsverbindungen und IP-Konfiguration des CP einrichten.

AUTOHOTSPOT

Anweisungen für programmierte offene Kommunikationsdienste (Open User Communication) mit ISO-on-TCP-, UDP und TCP-Verbindung über CPU oder CP*)

*) Anmerkung: Kommunikation über CP ist mit diesen Anweisungen nur über ISO-on-TCP möglich.

Bezeichnung	Kurzbeschreibung
TCON	Verbindungsaufbau
TDISCON	Verbindungsabbau
TSEND	Senden von Daten
TRCV	Empfangen von Daten

Bezeichnung	Kurzbeschreibung
TUSEND	Senden von Daten; verbindungsloses Protokoll UDP gemäß RFC 768
TURCV	Empfangen von Daten; verbindungsloses Protokoll UDP gemäß RFC 768

AUTOHOTSPOT

Kommunikationsverbindungen über CPs

Datenvolumen und Mengengerüst

Die Anzahl Kommunikationsverbindungen, die durch den jeweilige CP unterstützt werden, entnehmen Sie bitte dem Gerätehandbuch, das jedem CP beiliegt . Durch Hinzunahme weiterer CPs kann die Anzahl der Verbindungen pro Gerät erhöht werden.

Sind in einem Gerät mehrere CPs eingebaut, so wird bei Überschreitung dieser Grenze automatisch auf den nächsten CP umgeschaltet. Bei Bedarf ordnen Sie Verbindungen über die Wegewahl gezielt zu.

Ethernet-CP

Der Ethernet-CP kann über eine Verbindung pro Auftrag folgende Datenmengen übertragen:

	ISO-Transport	ISO-on-TCP	TCP	UDP
Senden	8192 Byte	8192 Byte	8192 Byte	2048 Byte
Empfangen	8192 Byte	8192 Byte	8192 Byte	2048 Byte

Hinweis

Der Datentransfer > 240 Byte wird von den aktuellen CPs unterstützt.

CPs mit älterem Ausgabestand unterstützen den Datentransfer mit einer Datenlänge von bis zu 240 Byte.

Beachten Sie hierzu bitte die Angaben im Gerätehandbuch des Ethernet-CP.

Aufgaben des Ethernet-CP im Online-Betrieb

Der Ethernet-CP übernimmt beim Datentransfer über eine Verbindung folgende Aufgaben:

- Empfangen
  Empfangen von Daten vom Ethernet und Weitergabe an den Anwender-Datenbereich in der CPU.
- Senden
  Übernehmen von Daten aus dem Anwender-Datenbereich der CPU und Senden der Daten über Ethernet.

Der Verbindungsaufbau erfolgt automatisch, sobald der Partner erreichbar ist.

Bei einer freien UDP/FDL-Verbindung kommt hinzu:

- beim Empfangen
  Eintragen des Absenders, von dem die Nachricht gesendet wurde, in den Auftrags-Header.
- beim Senden
  Auswerten des Auftrags-Headers und Adressieren des Partners.

PROFIBUS-CP

Der PROFIBUS-CP kann über eine Verbindung pro Auftrag folgende Datenmengen übertragen:

FDL-Verbindungs-typ	Spezifiziert	Unspezifiziert - Freier Layer 2 Zu-gang	Broadcast	Multicast
Senden	240 Byte	236 Byte [1)]	236 Byte [1)]	236 Byte [1)]
Empfangen	240 Byte	236 Byte [1)]	236 Byte [1)]	236 Byte [1)]

1) Der Auftrags-Header belegt neben den genannten Nutzdaten zusätzlich 4 Byte

Aufgaben des PROFIBUS-CP im Online-Betrieb

Der PROFIBUS-CP übernimmt für die Abwicklung des Datentransfers über eine FDL-Verbindung folgende Aufgaben:

- bei spezifizierten Verbindungen
  - Empfangen
    Empfangen von Daten vom PROFIBUS und Weitergabe an den Anwender-Datenbereich in der CPU.
  - Senden
    Übernehmen von Daten aus dem Anwender-Datenbereich der CPU und Senden der Daten über PROFIBUS.
- bei unspezifizierten Verbindungen zusätzlich
  - beim Empfangen
    Eintragen des Absenders und des FDL-Dienstes in den Auftrags-Header.
  - beim Senden
    Auswerten des Auftragsheaders und Adressieren des Partners; Ausführen des gewählten FDL-Dienstes.

Kommunikationsverbindungen zu PC-Stationen

Verbindungsendpunkte einer PC-Station

Bei PC-Stationen projektieren Sie Verbindungen mit einer der folgenden Komponenten als Verbindungsendpunkt:

- OPC-Server als zentraler Kommunikationsdienstleister
  Je nach verfügbarem CP können Sie folgende Verbindungstypen nutzen:
  - S7-Verbindung
  - ISO-Verbindung
  - ISO-on-TCP Verbindung
  - TCP-Verbindung
  - FDL-Verbindung
- Standard-Applikation
  Mit dem Verbindungstyp S7-Verbindung ist je nach verfügbarem CP die Kommunikation über Ind. Ethernet oder PROFIBUS möglich.

Hinweis

CPU 1515SP PC als Router für S7-Verbindungen

Wenn Sie die Schnittstelle "PROFINET onboard [X2]" der CPU 1515SP PC der SIMATIC PC-Station zuweisen, kann die CPU 1515SP PC als Router für S7-Verbindungen verwendet werden. Mit der Einstellung "Keine oder eine andere Windows-Einstellung" an der CP-Schnittstelle "PROFINET onboard [X2]" ist keine Verwendung der CPU 1515SP PC als Router für geroutete S7-Verbindungen möglich.
Eine bestehende durch die CPU 1515SP PC geroutete S7-Verbindung wird ungültig, wenn die Zuweisung der Schnittstelle der CPU 1515SP PC von "SIMATIC PC-Station" zu "Keine oder eine andere Windows-Einstellung" geändert wird. Da die PLC nun keine Routingfunktion mehr für diese Verbindung übernimmt, wird beim Übersetzen der CPU 1515SP PC kein Hinweis auf die ungültige Verbindung angezeigt. Die ungültige geroutete S7-Verbindung wird Ihnen erst beim Übersetzen der Endpunkte der Verbindung angezeigt.
Die für geroutete S7-Verbindungen benötigten Schnittstellen müssen in der CPU 1515SP PC explizit zugewiesen bleiben. Sie können die Zuweisung der Schnittstelle der CPU 1515SP PC in den Eigenschaften unter "PROFINET onboard [X2] > Schnittstellenzuweisung" bearbeiten.

Besonderheiten bei der Verbindungsprojektierung mit OPC-Server

Bei Verbindungen mit OPC als Verbindungsendpunkt können Sie prinzipiell mehr Verbindungen projektieren, als der in der PC-Station verwendete CP zu einem Zeitpunkt betreiben kann.

Dies hängt damit zusammen, dass bei den Verbindungseigenschaften zwischen "dauerhaft aufrechterhalten" und "auf Anforderung aufbauen" unterschieden wird.

Dementsprechend darf die Anzahl Verbindungen mit der Eigenschaft "dauerhaft aufrechterhalten" nicht die maximale Anzahl der CP-Verbindungsressourcen überschreiten. Falls es möglich sein soll, weitere Verbindungen auf Anforderung aufzubauen, müssen Sie

entsprechende Verbindungsressourcen des CP frei halten. Projektieren Sie daher eine geringere Anzahl Verbindungen mit der Eigenschaft "dauerhaft aufrechterhalten", als die maximale Anzahl der CP-Verbindungsressourcen zulässt!

Beim Übersetzen der Verbindungsprojektierung wird gegebenenfalls eine Warnung ausgegeben.

Weitere Informationen

Industrielle Kommunikation mit dem PG/PC Band 1 unter der Beitrags-ID: 42783968 (http://support.automation.siemens.com/WW/view/de/42783968)

Industrielle Kommunikation mit dem PG/PC Band 2 unter der Beitrags-ID: 42783660 (http://support.automation.siemens.com/WW/view/de/42783660)

PC-Stationen in Betrieb nehmen - Handbuch unter der Beitrags-ID: 13542666 (http://support.automation.siemens.com/WW/view/de/13542666)

Unspezifizierte Verbindungen

Unspezifizerte Verbindungen bei Industrial Ethernet

Verbindungen zu einem noch nicht bekannten Gerät (z.B. Diagnosegerät) werden als "unspezifizierte" Verbindungen projektiert. Die unspezifizierte Verbindung kann auf verschiedene Arten genutzt werden, wie nachfolgend am Beispiel ISO-on-TCP-Verbindung erläutert wird; auf ISO-Transport-, TCP-Verbindung ist die Vorgehensweise sinngemäß anzuwenden.

- **Kommunikationsbereitschaft erklären - passiver Verbindungsaufbau**
  Der Verbindungsaufbau ist "passiv" einzustellen.
  Für die Adresseinstellung bei ISO-on-TCP-Verbindung gilt dann:
  die ferne IP-Adresse und der ferne TSAP sind leer, d.h. sie sind für den CP nicht relevant.
  Beim Verbindungsaufbau wird dann jeder Partner akzeptiert (Partner = Verbindungsname, der den CP mit korrekter IP-Adresse und TSAP adressiert).
  Es besteht auch die Möglichkeit der Teilspezifizierung, d.h. die Kommunikation wird mit einem beliebigen Partner zugelassen, der in dem spezifizierten TSAP übereinstimmt.
- **Verbindung zu einer bestimmten Station in einem beliebigen Projekt**
  Für die Adresseinstellung bei ISO-on-TCP-Verbindung gilt dann:
  Sie können die ferne IP-Adresse und den Port für eine beliebige Zielstation angeben. Die Zielstation kann innerhalb oder außerhalb des aktuellen STEP 7-Projekts liegen.
  Verwenden Sie diese Variante, wenn Sie im aktuellen Projekt die Partnerstation nicht angelegt haben.

- **Verbindung ohne Port-Spezifizierung**
  TCP-Verbindungen sind in folgenden Fällen unspezifiziert:
  – Der lokale Port wird nicht angegeben (aktiver Verbindungsaufbau).
  – Der ferne Port wird nicht angegeben (kein aktiver Verbindungsaufbau).
- **IP-Adressierung über DHCP**
  Wenn Sie die Option IP-Adressierung über DHCP wählen, kann zunächst keine voll spezifizierte Verbindung im Projekt angelegt werden, da die lokale IP-Adresse nicht bekannt ist. Sie müssen daher als Verbindungstyp "unspezifiziert" ohne aktiven Verbindungsaufbau wählen.

Die folgende Tabelle fasst die Möglichkeiten zur Einstellung der "fernen" Adressparameter zusammen:

Bedeutung für Verbindungaufbau	IP-Adresse / MAC-Adresse (fern)	TSAP / Port (fern)	aktiver Verbindungsaufbau
von beliebigem Partner	leer	leer	nein
von beliebigem Partner über bestimmten TSAP	leer	spezifiziert	nein
zu oder von einem bestimmten Partner	spezifiziert	spezifiziert	ja Der lokale Port kann unspezifiziert bleiben (muss aber nicht).
		unspezifiziert	nein

Eine weitere Variante stellt die Freie UDP-Verbindung dar. Bei dieser Verbindungsart bleibt die Adresse des Verbindungspartners bei der Projektierung offen. Die Kommunikationsteilnehmer sind durch Adressangaben im Kommunikationsauftrag des Anwenderprogrammes bestimmt.

Unspezifizierte Verbindungen bei PROFIBUS

Eine unspezifizierte FDL-Verbindung mit offenem Layer 2 Zugang ermöglicht die programmgesteuerte Adressierung des Kommunikationspartners und die Kommunikation zwischen zwei Teilnehmern am PROFIBUS mit folgenden Eigenschaften:

Der lokale Teilnehmer ist per Projektierung festgelegt. Der ferne Teilnehmer wird vom Anwenderprogramm beim Aufruf der Anweisung AG_SEND im Auftragsheader des Auftragspuffers eingetragen. Damit kann jeder Teilnehmer am PROFIBUS (PROFIBUS-Adressen von 0 bis 126) erreicht werden.

Weitere Informationen

Weitere Informationen hierzu finden Sie nachfolgend in den verbindungspezifischen Einzelkapiteln.

Freie UDP-Verbindung (Seite 279)

FDL Unspezifizierte FDL-Verbindung (Freier Layer 2 Zugang) (Seite 288)

Programmierte Kommunikationsverbindungen über CP

Anwendungsbereich

Bei bestimmten Anwendungsbereichen ist es vorteilhaft, die Kommunikationsverbindungen oder die IP-Konfiguration des CP nicht über die Projektierung, sondern über eine spezifische Applikation programmgesteuert einzurichten.

Sie haben die Möglichkeit, Kommunikationsverbindungen über Industrial Ethernet-CPs programmgesteuert frei zu konfigurieren. Beachten Sie hierzu die Leistungsmerkmale des von Ihnen verwendeten CP-Typs.

Typische Anwendungsfälle finden sich z. B. bei Herstellern von Serienmaschinen. Um ihren Kunden möglichst komfortable Bedienoberflächen anzubieten, passen diese Hersteller die Kommunikationsdienste an die jeweiligen Bedieneingaben an. Der Endanwender benötigt keine Kenntnisse zur Projektierung von Verbindungen.

Für diese Anwendungen steht die Anweisung IP_CONFIG zur Verfügung, welche die flexible Übergabe von Datenbausteinen mit Projektierdaten an einen Ethernet-CP ermöglicht.

Zusammenspiel von Programmierung und Projektierung

Verbindungen werden entweder projektiert oder per Anwenderprogramm zur Laufzeit des S7-Gerätes konfiguriert. Eine Mischform dieser Varianten ist innerhalb eines CP nicht möglich!

Prinzip

Über die Anweisung IP_CONFIG im Anwenderprogramm werden Konfigurationsdaten für Kommunikationsverbindungen an den CP übertragen.

Der Konfigurations-DB kann auf diesem Weg jederzeit in den CP geladen werden. Die vorher aktuellen Verbindungen und Konfigurationsdaten (IP-Adresse, Subnetz-Maske, Default-Router, NTP-Uhrzeit-Server und weitere Parameter) werden hierbei überschrieben.

Der Ethernet-CP erkennt anhand der Projektierungsdaten, dass die Kommunikationsverbindungen über das Anwenderprogramm eingerichtet werden sollen.

Mengengerüst

In der Anweisung IP_CONFIG können maximal 64 Verbindungen angegeben werden. Maßgebend ist jedoch die maximale Anzahl Verbindungen, die der von Ihnen genutzte CP-Typ unterstützt.

Besonderheiten/Einschränkungen

- Konsistenzprüfung nur bei projektierter Verbindung
  Die Verbindungsprojektierung ist mit Konsistenzprüfungen verbunden, die bei der programmierten Konfiguration nicht oder nur bedingt möglich sind!
- Verbindungsprojektierung beim Partner erforderlich
  Bei der Projektierung spezifizierter Verbindungen wird bei der Projektierung implizit die Verbindung für den Partner angelegt. Dies ist bei einer programmierten Konfiguration nicht möglich! Hier müssen für die Partner entsprechende konsistente Verbindungen projektiert bzw. konfiguriert werden.
- Projektierung des IP-Zugriffschutzes
  Über den IP-Zugriffschutz besteht die Möglichkeit, die Kommunikation über den CP des lokalen S7-Gerätes auf Partner mit ganz bestimmten IP-Adressen einzuschränken. Diese Parametrierung gilt auch für programmierte Kommunikationsverbindungen.
- DHCP/DNS wird unterstützt
  Die IP-Adressierung ist bei der programmierten Konfiguration auch über DHCP/DNS möglich.

Verbindungen mit Open User Communication über CPU und CP

Open User Communication über Industrial Ethernet - programmierte Kommunikationsverbindungen

Die Open User Communication stellt Dienste für Industrial Ethernet zur Verfügung, die Sie über die Anweisungsschnittstelle im Anwenderprogramm nutzen.
Kommunikationsverbindungen werden über die Anweisungsschnittstelle im Anwenderprogramm eingerichtet, eine Projektierung von Verbindungen ist nicht erforderlich.

Dienste über die Schnittstelle der CPU

Diese Dienste der Open User Communication stehen bei S7-1500 für den Datenaustausch über die integrierte Ethernet-Schnittstelle einer CPU zur Verfügung:

- TCP gemäß RFC 793
- ISO-on-TCP gemäß RFC 1006
- UDP gemäß RFC 768 (Broadcast oder Singlecast, keine Unterstützung von Multicast)

Dienste über die Schnittstelle des CP

Diese Dienste der Open User Communication stehen bei S7-300/400/1500 für den Datenaustausch über die Ethernet-Schnittstelle eines CP zur Verfügung:

- TCP gemäß RFC 793
- ISO gemäß RFC 905
- ISO-on-TCP gemäß RFC 1006
- UDP gemäß RFC 768 (Broadcast oder Singlecast, keine Unterstützung von Multicast)

1.3.2.3 Verbindungen projektieren

Anlegen einer neuen Verbindung - Übersicht

Anlegen einer Verbindung - Alternativen

Sie haben folgende Möglichkeiten, eine Verbindung in der Netzsicht anzulegen:

- Grafische Verbindungsprojektierung
  Sie ziehen eine Verbindungslinie zum Verbindungspartner auf oder weisen den zuvor selektierten Verbindungspartnern eine neue Verbindung zu. Dies ist eine komfortable Möglichkeit zum Anlegen einzelner Verbindungen in der Netzsicht.
  Fehlende Vernetzungen werden automatisch oder dialoggeführt ergänzt.
- Dialoggeführte Verbindungsprojektierung
  Sie wählen den Verbindungspartner in einer Liste. Wenn größere Mengen an Verbindungen zu projektieren sind, dann ist diese Vorgehensweise gegenüber der grafischen Vorgehensweise im Vorteil.

Die einzelnen Schritte hierzu finden Sie in den Folgekapiteln.

Verbindungsweg zuweisen

Je nach Einsatzfall sind folgende Vorgehensweisen zu unterscheiden:

- Auswahl des Verbindungsendpunkts in Quell- und Zielgerät
  Die Software ermittelt automatisch den geeigneten Verbindungsweg zum Zielgerät. Diese Verbindung wird dann als "Standard Verbindungsweg" bezeichnet. Die Software berücksichtigt bei der Wegewahl die verfügbaren Verbindungs-Ressourcen.
  Verbindungsendpunkt kann eine CPU oder eine PC-Applikation sein.
  Anmerkung: Diese Variante ist nur bei der grafischen Verbindungsprojektierung möglich.
- Auswahl von Verbindungswegen in Quell- und Zielgerät
  Sofern Sie zum Zwecke der Lastteilung mehrere Komponenten (CPs) und evtl. auch unterschiedliche Subnetze vorsehen, können Sie den Weg einer Verbindung auch gezielt projektieren. Sie legen einen Verbindungsweg über den Verbindungsendpunkt und die Wahl der Schnittstelle zum Subnetz fest.

Voraussetzung und Ergebnis

Sie haben in der Netzsicht die Geräte mit CPUs und PC-Applikationen sowie gegebenenfalls kommunikationsfähigen Baugruppen (CP) angelegt.

Das Projektierungsergebnis hängt dann von der aktuellen Konfiguration der Geräte und von der Vernetzung im Projekt ab. Mit den in den Folgekapiteln beschriebenen Vorgehensweisen erzielen Sie daher folgende Ergebnisse:

- Verbindung spezifiziert
  Wenn beide ausgewählte Partner am selben Subnetz vernetzt sind, legen Sie grafisch oder dialoggeführt eine voll-spezifizierte Verbindung an.
  Diese voll-spezifizierte Verbindung wird automatisch sowohl in die Verbindungstabelle des lokalen Teilnehmers als auch des Partners eingetragen. Für diese Verbindung wird die Lokale ID und die Partner-ID vergeben. Diese IDs benötigen Sie bei der Programmierung der Kommunikationsanweisungen (Wert für den Bausteinparameter "ID") oder der PC-Applikationen.
  Ausnahmen:
  - Einseitige S7-Verbindungen
    Bei einer einseitigen S7-Verbindung hat der Verbindungspartner eine Server-Funktion. Daher erfolgt beim Partner kein Eintrag in der Verbindungstabelle. Nur für den lokalen Partner wird eine Verbindungs-ID vergeben.

  Die folgende Darstellung zeigt eine spezifizierte Verbindung:

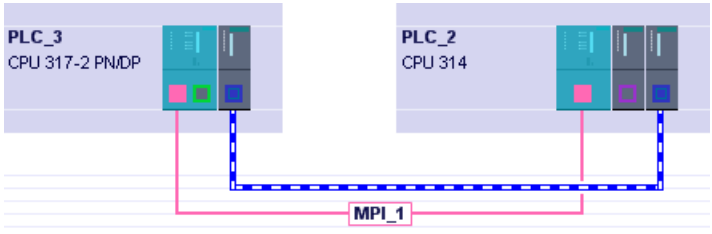

- Verbindung unspezifiziert
  In folgenden Fällen legen Sie unspezifizierte Verbindungen gezielt an, indem Sie nur den lokalen Verbindungspartner grafisch oder dialoggeführt anwählen:
  - Wenn bestimmte Verbindungseigenschaften eine unspezifizierte Verbindung erfordern. Beispiele: E-Mail Verbindung, FTP-Verbindung, FDL-Verbindung mit freiem Layer 2 Zugang.
  - Wenn einer der beiden Partner für den gewählten Verbindungstyp (noch) nicht verfügbar ist, Sie aber eine Verbindungsressource reservieren möchten.

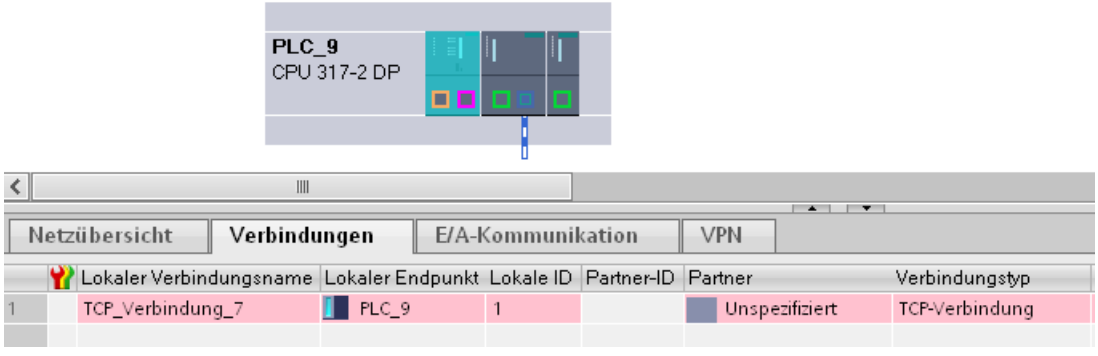

Einfluss der Vernetzung - Verhalten bei fehlender Vernetzung

Die Vernetzung der Verbindungspartner ist eine Voraussetzung für die Eindeutigkeit und Konsistenz der zugeordneten Verbindungsparameter. Wenn einer bzw. beide Partner nicht am selben Subnetz vernetzt sind, ist folgendes Verhalten bei der grafischen Verbindungsprojektierung zu unterscheiden:

Hinweis

Dialoggeführte Verbindungsprojektierung - keine automatische Vernetzung

Bei der dialoggeführten Verbindungsprojektierung werden keine automatischen Vernetzungen ausgeführt.

- Schnittstellen der zu verbindenden Partner sind nicht vernetzt oder zumindest nicht über das selbe Subnetz vernetzt
  Der aufgeblendete Dialog "Mit Subnetz verbinden" bietet folgende Auswahlmöglichkeiten:
  - Vorhandenes Subnetz zuweisen
    Sie wählen ein Subnetz aus der Liste der vorhandenen Subnetze aus. Die Verbindung kann konsistent und spezifiziert angelegt werden.
  - "Mit neuem Subnetz verbinden"
    Die Schnittstellen der Verbindungspartner werden über ein neues Subnetz verbunden. Die Verbindung kann konsistent und spezifiziert angelegt werden.
  - "Nicht mit Subnetz verbinden"
    Die Verbindung wird spezifiziert angelegt; die Konsistenz ist aufgrund der fehlenden Vernetzung nicht gesichert, weshalb der Eintrag in der Verbindungstabelle rot erscheint. Verbindungen sind in der Netzsicht als gestrichelte Linie dargestellt;
- Mehrere Schnittstellen / CPs beim Partner - mindestens eine der verfügbaren Schnittstellen beim Partner ist nicht vernetzt
  Je nachdem, wie Sie die Verbindung in diesem Fall anlegen, gilt:
  - Wenn Sie die Verbindung nur über die Auswahl der CPU / OPC-Server als Verbindungsendpunkt festlegen, wird der Verbindungsweg unabhängig von der Vernetzung festgelegt. Dies kann zu dem unter Fall b beschriebenen Verhalten führen, so dass der Auswahldialog aufgeblendet wird.
  - Wenn Sie als Verbindungsendpunkt gezielt die noch freie, nicht vernetzte Schnittstelle beim Partner wählen, wird das unter Fall a beschriebene Verhalten wirksam.
    Die Schnittstellen der Verbindungspartner werden über ein neues Subnetz verbunden. Die Verbindung kann konsistent und spezifiziert angelegt werden.

Ansichten mit Informationen zu den projektierten Verbindungen

Über die nachfolgend beschriebenen Ansichten haben Sie umfassenden Zugriff auf sämtliche Informationen und Funktionen zum Projektieren und Prüfen von Kommunikationsverbindungen.

Sie erreichen die Ansicht über die Projektnavigation unter "Geräte & Netze".

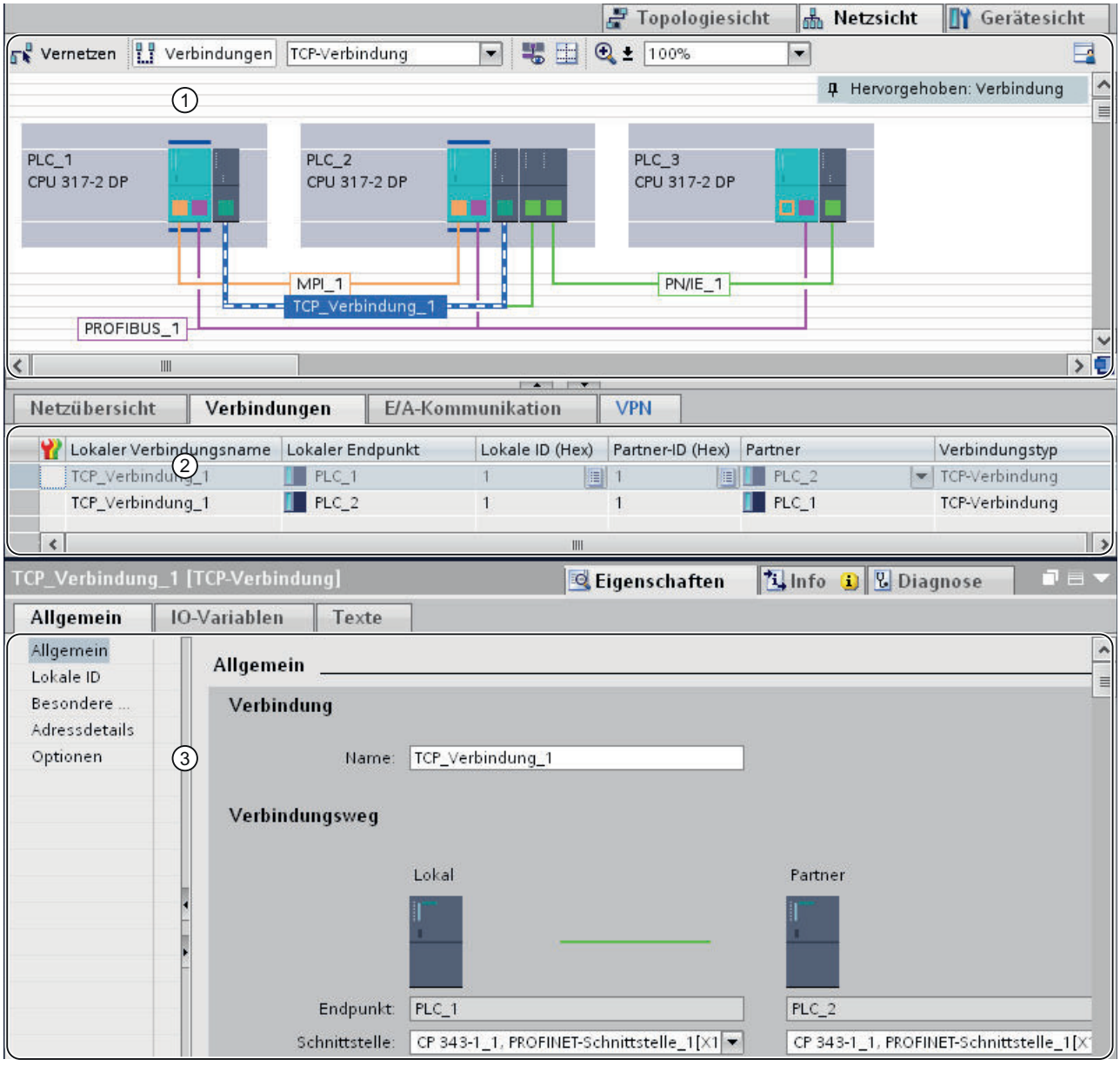

① Verbindungsanzeige in der Netzsicht

② Verbindungstabelle

③ Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Vorteile

Die in diesen Ansichten dargestellten Informationen sind in Bezug auf augenblickliche Benutzeraktionen immer aktuell. Das bedeutet:

- Wenn Sie z. B. in der Netzsicht eine CPU angewählt haben, zeigt die Verbindungstabelle alle zu dieser CPU angelegten Verbindungen.
  Anmerkung: Über das Kontextmenü der Verbindungstabelle können Sie zwischen der Anzeige sämtlicher Verbindungen im Projekt und der Anzeige der Verbindungen einer selektierten Komponente wählen.
- Wenn Sie eine Verbindung in der Verbindungstabelle angewählt haben, gilt:
  - Bei eingeschaltetem Vebindungsmodus sehen Sie in der Netzsicht den Verbindungsweg grafisch.
  - Die Eigenschaften-Ansicht im Inspektorfenster zeigt die Parameter dieser Verbindung.

Verbindungsanzeige in der Netzsicht

Die Netzsicht zeigt Geräte mit ihren Komponenten sowie Netze und Kommunikationsverbindungen.

In Bezug auf Verbindungen erkennen Sie in der Netzsicht:

- Zwischen welchen Geräten Verbindungen angelegt sind.
- Von welchen Geräten unspezifizierte Verbindungen ausgehen.

Die Netzsicht bietet folgende Funktionen:

- Verbindungen grafisch anlegen;
- Verbindungswege beim grafischen Anlegen wählen.

Die Verbindungstabelle

Die Verbindungstabelle bietet folgende Funktionen:

- Auflistung der Verbindungen zu einem in der Netzsicht angewählten Objekt (Netz, CPU, CP);
- Auflistung aller Verbindungen im Projekt;
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht sowie Anzeige der Eigenschaften einer selektierten Verbindung im Inspektorfenster;
- Ändern von Verbindungspartnern;
- Eingabemöglichkeit von zentralen Parametern;
- Anzeige von Zustandsinformationen;
- Einstieg zum dialoggeführten Anlegen neuer Verbindungen (möglich bei abgewählter Option "Alle Verbindungen anzeigen").

Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Der Eigenschaften-Dialog hat folgende Bedeutung:

- Anzeige und Einstellmöglichkeit aller Verbindungsparameter;
- Einstellung des Verbindungswegs; automatisch oder wählbar
- Nachträgliches Spezifizieren von unspezifizierten Verbindungen.

Neue Verbindung grafisch anlegen

Verbindungen grafisch projektieren

- Im einfachsten Fall wählen Sie lediglich die zu verbindenden Geräte aus. Ein möglicher Verbindungsweg in der aktuellen Konfiguration wird ausgewählt und als "Standard Verbindungsweg" gekennzeichnet. Der Verbindungsweg wird dann so gewählt, dass die verfügbaren Ressourcen bestmöglich genutzt werden. Folgendes Verhalten liegt der Festlegung zu Grunde:
  - Nicht vernetzte Komponenten werden möglichst automatisch vernetzt, um voll spezifizierte Verbindungen direkt anlegen zu können.
  - Die Ressourcen einer Komponente werden erst vollständig aufgebraucht, bis auf eine andere Komponente umgeschaltet wird.
  - Netztypen unterliegen bei netzunabhängigen Verbindungstypen (S7-Verbindungen) folgender Vorzugsregel: Industrial Ethernet vor PROFIBUS.
- Alternativ legen Sie den Verbindungsweg - Verbindungsendpunkt und Schnittstelle zum Subnetz - gezielt fest.

Hinweis

S7-Verbindungen an der CPU 1515SP PC grafisch anlegen

Wenn Sie eine S7-Verbindung an einer CPU 1515SP PC anlegen möchten, verwenden Sie beim grafischen Anlegen der Verbindung die rechte Schnittstelle der CPU 1515SP PC "PROFINET onboard [X1]". Diese Schnittstelle ist direkt mit der CPU 150xS verbunden.

Die linke Schnittstelle der CPU 1515SP PC "PROFINET onboard [X2]" ist mit der SIMATIC PC-Station oder einer Windows-Anwendung verbunden und kann beim grafischen Anlegen einer S7-Verbindung nicht verwendet werden.

Verbindungsweg automatisch ermitteln

Um eine Verbindung über einen Default Verbindungsweg grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

2. Wählen Sie den benötigten Verbindungstyp in der danebenstehenden Klappliste.
   Sie erkennen anschließend folgendes Verhalten:
   – In der Netzsicht sind die für den gewählten Verbindungstyp in Ihrem Projekt als Verbindungsendpunkte verwendbaren Komponenten farblich hervorgehoben. Bei den Komponenten kann es sich um CPUs oder PC-Applikationen handeln.

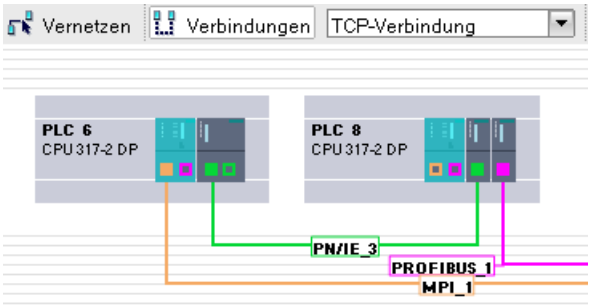

3. Selektieren Sie die Komponente, von der eine Verbindung ausgehen soll.
4. Zum Anlegen der Verbindung haben Sie 2 Möglichkeiten:
   – Klicken Sie auf die Komponente, von der eine Verbindung ausgehen soll und ziehen Sie den Mauscursor im Verbindungsmodus auf die Ziel-Komponente. Bestätigen Sie den Verbindungsendpunkt durch erneuten Mausklick.

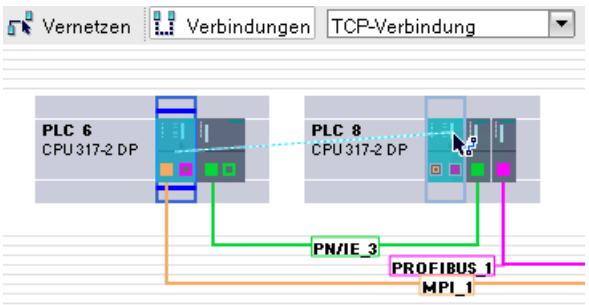

   – Selektieren Sie bei gedrückter <Shift>-Taste zusätzlich die Zielkomponente und wählen Sie mit der rechten Maustaste den Befehl "Neue Verbindung hinzufügen".

Hinweis

Wenn Sie den Verbindungsmodus vorher beenden möchten, drücken Sie <Esc>, klicken Sie die rechte Maustaste oder doppelklicken Sie in den Hintergrund der Netzsicht.

Ergebnis

- Eine spezifizierte Verbindung ist angelegt.
- Der Verbindungsweg ist hervorgehoben dargestellt. Der Verbindungsweg wurde automatisch entsprechend den verfügbaren Verbindungs-Ressourcen und Schnittstellen eingestellt.
- Die Verbindung ist in den Verbindungstabellen der Komponenten (CPU / PC-Applikation) eingetragen.

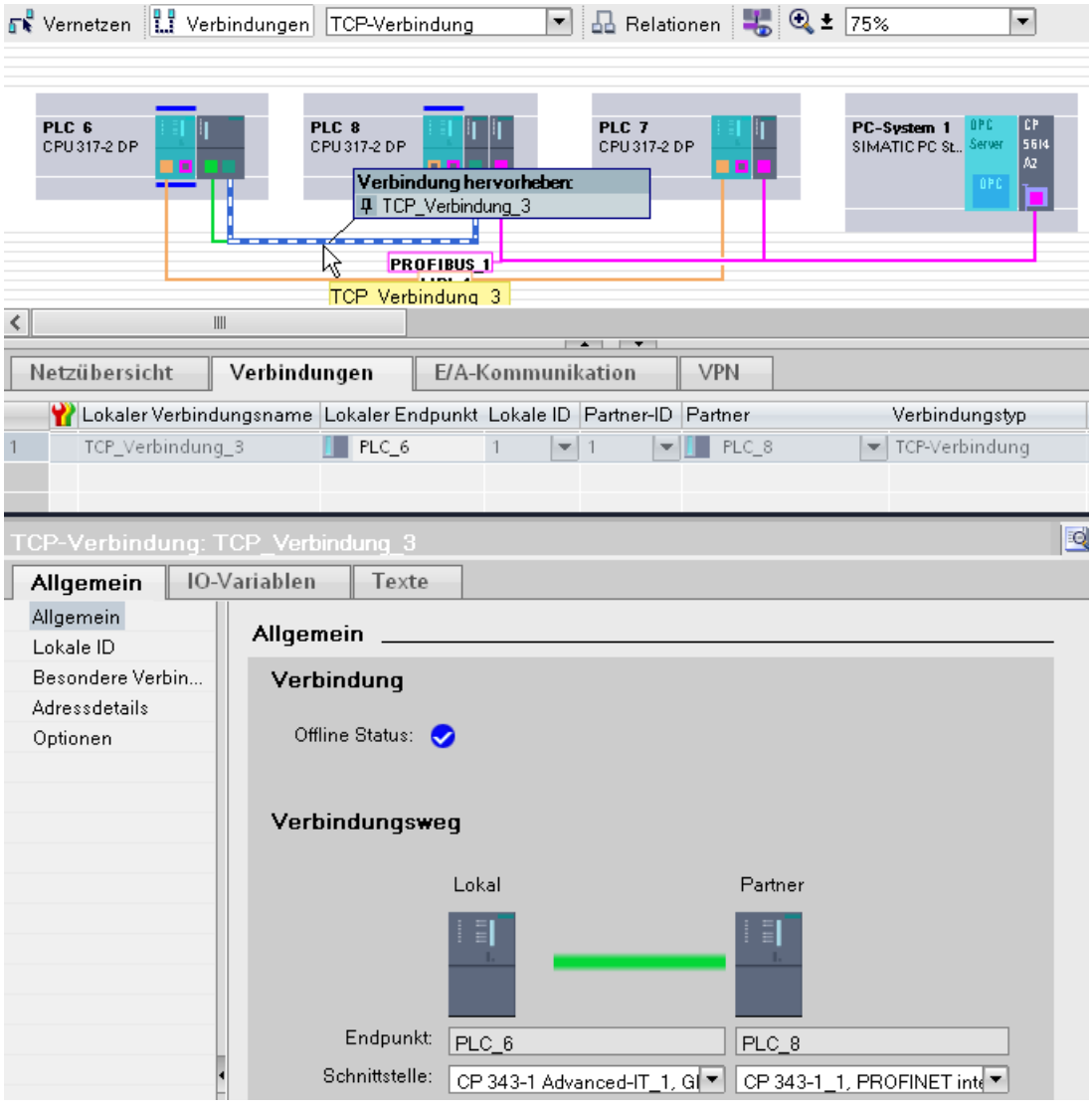

Verbindungsweg gezielt wählen

Um eine Verbindung mit explizit gewähltem Verbindungsweg anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

2. Wählen Sie den benötigten Verbindungstyp in der Klappliste.
3. Klicken Sie auf die Subnetz-Schnittstelle in dem Gerät, von dem eine Verbindung ausgehen soll.
4. Ziehen Sie den Mauscursor im Verbindungsmodus bei gedrückter Maustaste auf die gewünschte Schnittstelle im Zielgerät und lassen Sie die Maustaste los.

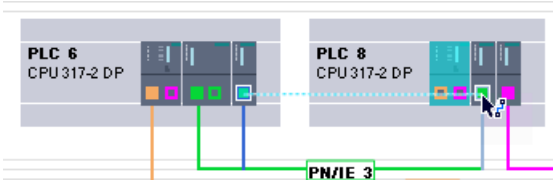

Hinweis

Wenn Sie den Verbindungsmodus vorher beenden möchten, drücken Sie <Esc>, klicken Sie die rechte Maustaste oder doppelklicken Sie in den Hintergrund der Netzsicht.

Ergebnis:

- Eine spezifizierte Verbindung ist angelegt.
- Die zwischen den Endpunkten angelegte Verbindung verwendet die gewählten Schnittstellen für den Verbindungsweg.
- Der Verbindungsweg ist hervorgehoben dargestellt.
- Die Verbindung ist in den Verbindungstabellen der Geräte (CPU / PC-Applikation) eingetragen.

Verbindung bei fehlender oder nicht eindeutiger Netzzuordnung projektieren

Fehlende Vernetzungen werden soweit möglich beim Anlegen einer Verbindung automatisch angelegt. Falls eine eindeutige Netzzuordnung nicht möglich ist, erfolgt eine Abfrage beim Abschluss der Verbindungsprojektierung. Hierbei werden die bestehenden Subnetze zur Auswahl angeboten.

Beispiel in folgendem Bild: Beim Anlegen einer Verbindung zwischen den noch nicht vernetzten Stationen PLC2 und PLC3 erfolgt eine Abfrage.

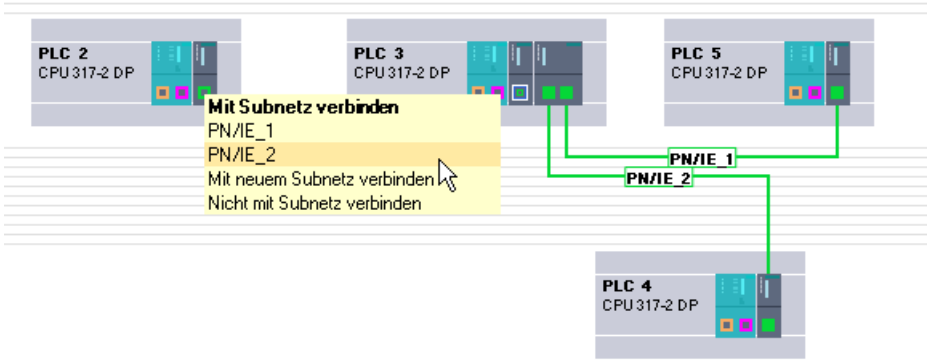

Neue Verbindung dialoggeführt anlegen

Verbindungen dialoggeführt projektieren

Die dialoggeführte Verbindungsprojektierung setzen Sie vorteilhaft ein, um innerhalb einer Station mehrere Verbindungen in einem Arbeitsgang anzulegen.

Sie haben dabei einerseits die vollständige Kontrolle über die Wahl des Verbindungswegs, da das System alle Schnittstellenvarianten anzeigt. Andererseits haben sie die Möglichkeit, dem System die Berücksichtigung der verfügbaren System-Ressourcen und damit die Wahl des Verbindungswegs zu überlassen.

Hinweis

Dialoggeführte Verbindungsprojektierung - keine automatische Vernetzung

Bei der dialoggeführten Verbindungsprojektierung werden keine automatischen Vernetzungen ausgeführt.

Der Dialog "Neue Verbindung erstellen"

Nachfolgend der Aufbau des Dialogs "Neue Verbindung erstellen". Dargestellt ist eine Vernetzung zwischen mehreren Stationen. Der Verbindungsmodus ist aktiviert und die oben angeordnete Station ist selektiert.

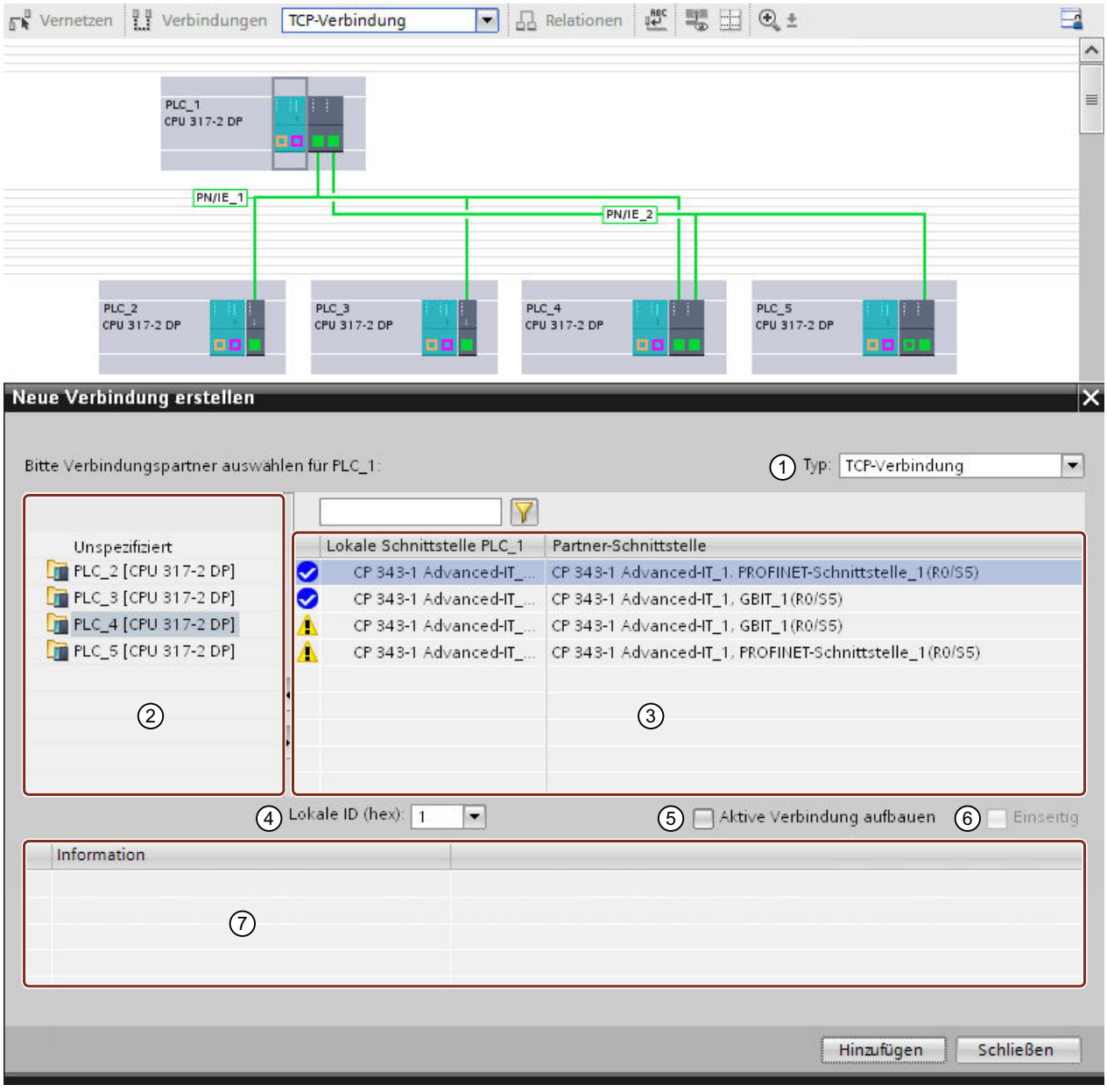

① Klappliste zur Auswahl des Verbindungstyps.

② Anzeige der Verbindungspartner, die für den angewählten Verbindungstyp infrage kommen.

Wählen Sie aus folgenden Alternativen:

- Je nach Verbindungstyp eine spezielle Verbindungskonfiguration:
  - "unspezifiziert"
  - "Broadcast"
  - "Multicast"
- Den Verbindungspartner

③ Inhaltsbereich zur Auswahl der Schnittstellen.

Sie sehen die der Konfiguration entsprechend möglichen Schnittstellen-Kombinationen. Angezeigt werden nur die Schnittstellen, die für den angewählten Verbindungstyp infrage kommen. Die Statusanzeige gibt zusätzlich Auskunft darüber, ob die Verbindung in Abhängigkeit von der Vernetzung und der verfügbaren Ressourcen eingerichtet werden kann. Wenn eine fehlende Vernetzung der Schnittstellen vorliegt, können Sie eine Verbindung dennoch anlegen.

Siehe auch AUTOHOTSPOT

Gezeigtes Beispiel:

Zwischen PLC1 und PLC4 sind jeweils die Gbit-Ports einerseits sowie die PROFINET-Schnittstellen andererseits vernetzt. Entsprechend werden nur diese beiden Kombinationen als vollständig zuordenbar angezeigt

④ Klappliste zur Auswahl oder Eintragung der lokalen ID.

⑤ Optionskästchen zur Anzeige des aktiven Verbindungsaufbaus.

Beim Anlegen einer Verbindung wird grundsätzlich immer für die lokale Seite das Optionskästchen für den aktiven Verbindungsaufbau aktiviert. Nicht jeder Teilnehmer besitzt die Fähigkeit den Verbindungsaufbau zu übernehmen. In diesem Fall muss der Partner den Aufbau übernehmen. Das Optionskästchen zeigt dabei immer die Fähigkeit der selektierten Verbindung an.

⑥ Optionskästchen zur Anzeige einer einseitigen Verbindung.

Bei einer einseitigen Verbindung liegt eine Client-Server-Beziehung vor. Nur dem Client ist der Server bekannt, während der Server keinerlei Informationen über den Client hat. Bei einer zweiseitigen Verbindung sind in beiden Endpunkten die Informationen über den Partner bekannt. Das Optionskästchen zeigt dabei immer die Information zur selektierten Verbindung an.

⑦ Zeigt die Bestätigung für das Anlegen einer neuen Verbindung an.

Der Typ der Verbindung und die beiden Verbindungspartner werden genannt. Gelbe Icons zeigen an, dass die Verbindung erstellt, aber nicht konsistent ist. Prüfen Sie in dem Fall die Verbindungsparameter.

Statusanzeige

Symbol	Bedeutung
	Schnittstellen sind vernetzt und bei den gewählten Partnern stehen Verbindungs-Ressourcen zur Verfügung.
	Schnittstellen sind nicht vernetzt oder nicht am selben Subnetz angeschlossen.

Vorgehen

Um eine Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

2. Wählen Sie im Kontextmenü der Komponente, die Verbindungsendpunkt sein soll (CPU oder PC-Applikation), den Befehl "Verbindung hinzufügen".
   oder
   Wählen Sie bei abgewählter Option "Alle Verbindungen anzeigen" und ausgewählter CPU oder Applikation im Kontextmenü der Verbindungstabelle den Befehl "Verbindung hinzufügen".
   Der Dialog "Neue Verbindung erstellen" wird geöffnet.
3. Wählen Sie in der Klappliste "Typ" den gewünschten Verbindungstyp, z. B. "FDL-Verbindung".
4. Prüfen Sie in der Fußzeile die Voreinstellung zusätzlich angezeigter Verbindungsparameter und korrigieren Sie diese gemäß Ihren Anforderungen.

   Hinweis

   Sie können diese Parameter auch nachträglich in der Verbindungstabelle oder in den Eigenschaften der Verbindung ändern.

5. Wählen Sie im linken Bereich des Dialogs den Verbindungspartner (Gerät oder CPU). Im Inhaltsbereich werden die der Konfiguration entsprechend möglichen Schnittstellen-Kombinationen mit Statusinformationen angezeigt. Die erste Schnittstellenkombination, der eine Verbindung vom ausgewählten Typ zugewiesen werden kann, ist selektiert.

   Hinweis

   Je nach gewähltem Verbindungstyp werden neben den möglichen Verbindungspartnern spezielle Verbindungstypen angeboten, wie beispielsweise "unspezifiziert", "Broadcast" oder "Multicast".

6. Belassen Sie es bei der Auswahl oder wählen Sie den gewünschten Verbindungsweg aus.
7. Bestätigen Sie die Auswahl über die Schaltfläche "Hinzufügen".
   Eine Verbindung wird angelegt und ist in der Verbindungstabelle sichtbar.
8. Wenn Sie den Dialog nicht geschlossen haben, gehen Sie alternativ weiter so vor:
   - Eine weitere Verbindung desselben Typs zum selben Zielgerät legen Sie an, indem Sie erneut die Schaltfläche "Hinzufügen" Klicken.
   - Wiederholen Sie den oben beschriebenen Vorgang mit der Auswahl eines anderen Verbindungstyps aus der Klappliste oder eines anderen Verbindungspartners.

Verbindung bei fehlender Vernetzung projektieren

Fehlende Vernetzungen werden soweit möglich beim Anlegen einer Verbindung automatisch angelegt. Falls eine eindeutige Netzzuordnung möglich ist, wird daher nach Abschluss des Dialoges die entsprechende Schnittstelle in der Netzsicht vernetzt angezeigt.

Falls eine eindeutige Netzzuordnung nicht möglich ist, wird bei der dialoggeführten Verbindungsprojektierung keine Vernetzung eingefügt. So angelegte Verbindungen sind daher noch nicht vollständig spezifiziert und in der Verbindungstabelle rot markiert.

Arbeiten in der Netzsicht

Bedeutung der Netzsicht im Verbindungsmodus

Die Netzsicht zeigt Geräte mit ihren Komponenten und Schnittstellen zu den Subnetzen. Im Verbindungsmodus können Sie die Kommunikationswege projektierter Verbindungen sichtbar machen.

In Bezug auf Verbindungen erkennen Sie:

- Zwischen welchen Geräten Verbindungen angelegt sind
- Welche Verbindungstypen ausgehend von einem Gerät angelegt sind
- Von welchen Geräten unspezifizierte Verbindungen ausgehen

Die Netzsicht bietet im Verbindungsmodus folgende Funktionen:

- Verbindungen grafisch anlegen; siehe auch Neue Verbindung grafisch anlegen (Seite 226)
- Verbindungen eines Subnetzes selektiv hervorheben;
- Verbindungspartner hervorheben.

Verbindungsweg und -partner in der Netzsicht hervorheben

Um die Verbindungspartner für alle oder für bestimmte Verbindungstypen in der Netzsicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü des Verbindungsendpunktes (CPU oder PC-Applikation), dessen Verbindungspartner Sie in der Netzsicht anzeigen möchten, den Befehl "Verbindungspartner hervorheben".
2. Wählen Sie im Folgemenü entweder "Alle Verbindungspartner" oder "Verbindungstyp auswählen..."
   Der lokale Endpunkt und der für alle oder den gewählten Verbindungstyp zutreffenden Endpunkte der Zielgeräte sind markiert. Zusätzlich zeigen die Geräte einen Pfeil.
3. Öffnen Sie eine Liste mit Angabe der Zielgeräte, indem Sie den Pfeil in der Fußzeile des lokalen Gerätes mit der rechten Maustaste anwählen. Diese Zusatzfunktion ist bei komplexen Netzkonfigurationen nützlich, bei denen nicht alle Geräte sichtbar sind.

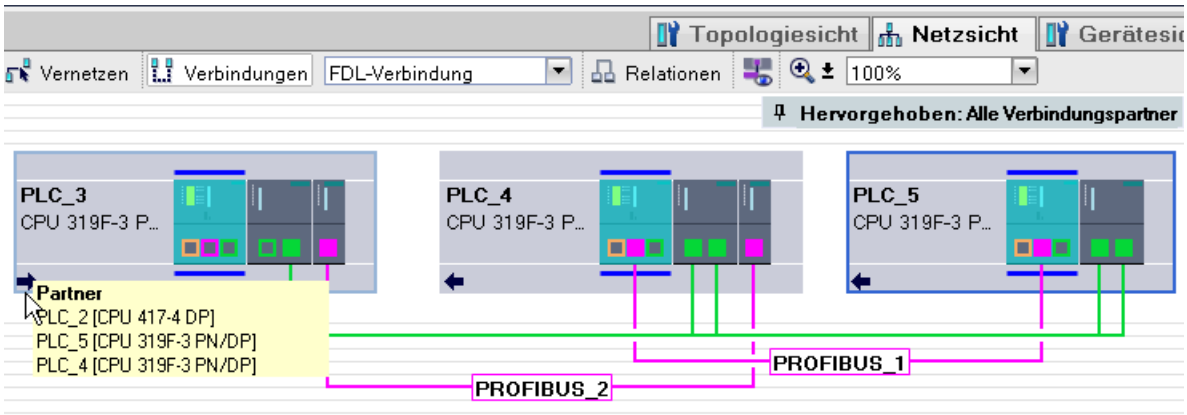

Hinweis

Sie können einen der Verbindungspartner, der im aktuellen Anzeigebereich der Netzsicht nicht sichtbar ist, anzeigen. Klicken Sie in der aufgeblendeten Liste auf den Kommunikationspartner. Ergebnis: Die Anzeige wird so verschoben, dass der Verbindungspartner sichtbar wird.

Verbindung hervorheben / auswählen

Sie können einzelne Verbindungen auswählen und hervorheben. Sie erreichen damit Folgendes:

- Die Verbindung ist in der Verbindungstabelle markiert; bestimmte Eigenschaften können Sie dort einstellen.
- Die Verbindung wird im Inspektorfenster mit sämtlichen Parametriermöglichkeiten angezeigt.

Gehen Sie so vor:

1. Zeigen Sie mit dem Mauscursor auf das Subnetz.
2. Wählen und markieren Sie im aufgeblendeten Dialog "Verbindung hervorheben" die gewünschte Verbindung.
   Der Verbindungsweg wird als hellblau gestrichelte Linie angezeigt.

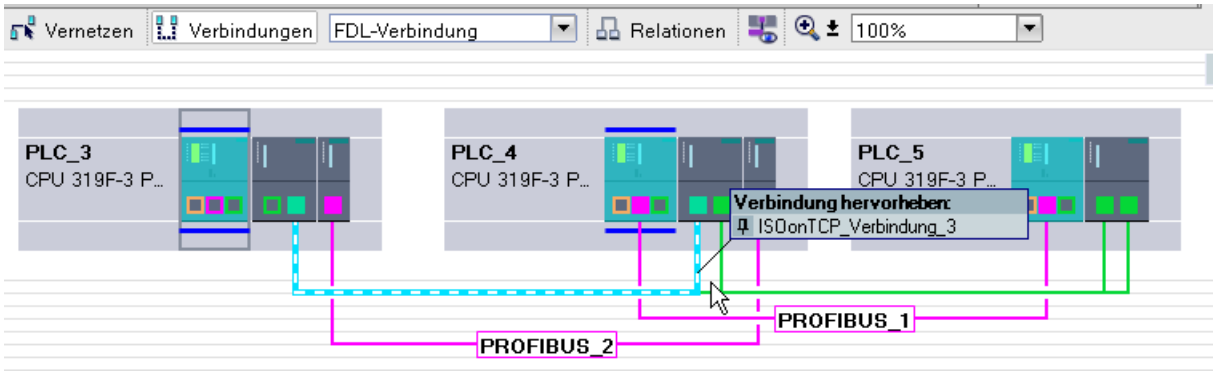

3. Klicken Sie auf die gestrichelte Linie.
   Die Linienfarbe schlägt in dunkelblau um; die Verbindung wird in der Verbindungstabelle sowie im Inspektorfenster angezeigt.

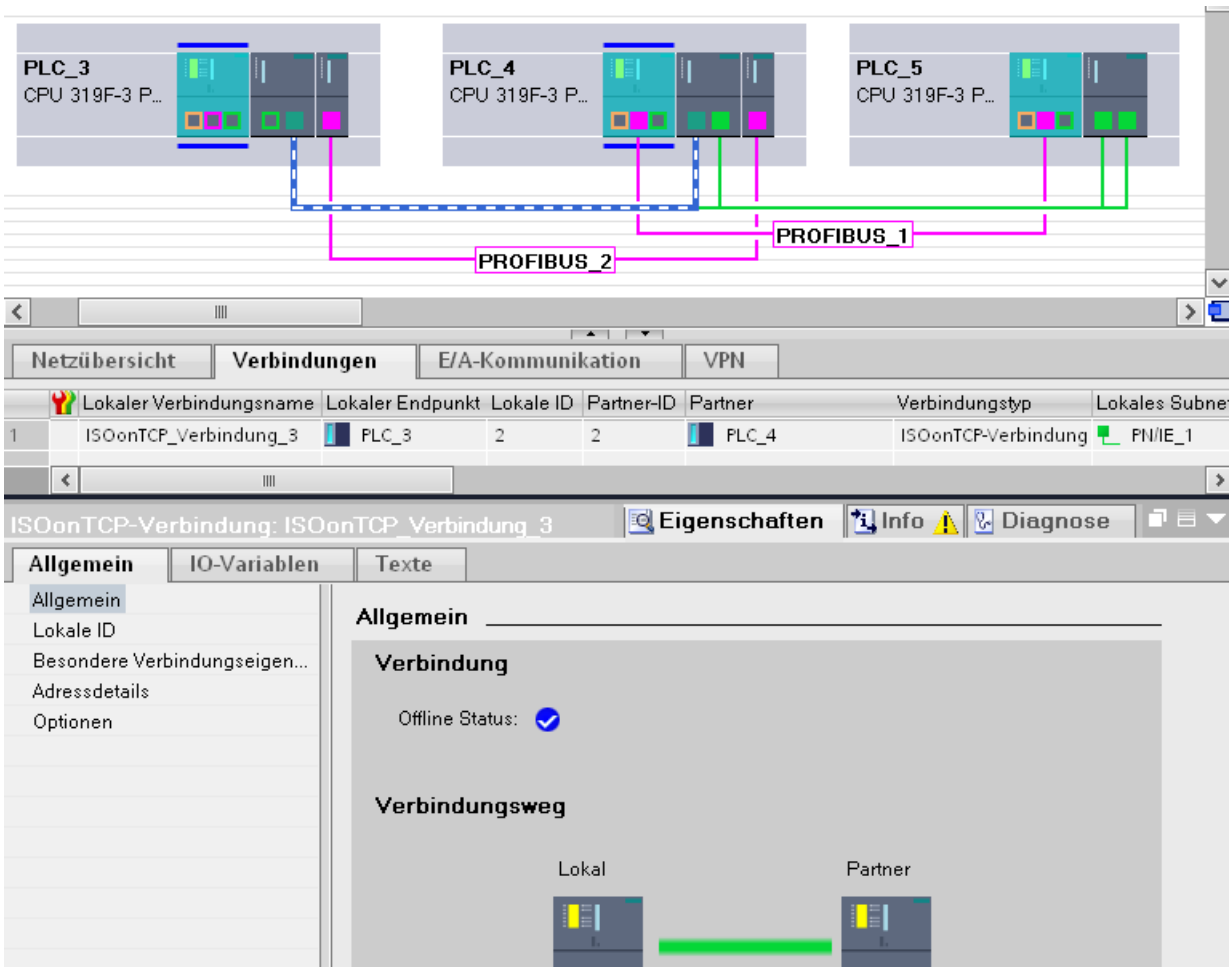

Arbeiten mit der Verbindungstabelle

Funktionen

Neben der Möglichkeit, Verbindungen dialoggeführt neu anzulegen, bietet die Verbindungstabelle weitere Funktionen:

- Auflistung der Verbindungen zu einem in der Netzsicht angewählten Objekt (Netz, CPU, CP)
- Auflistung aller Verbindungen im Projekt
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht
- Ändern von Verbindungspartnern
- Eingabemöglichkeit von zentralen Parametern
- Anzeige von Zustandsinformationen

Verbindungen dialoggeführt neu anlegen

Neue Verbindung dialoggeführt anlegen (Seite 230)

Inhalt der Verbindungstabelle

Über die Verbindungstabelle können Sie sich alle im Projekt angelegten Kommunikationsverbindungen im Detail anzeigen lassen:

- Alle Verbindungen anzeigen (Standardeinstellung - wählbar über Kontextmenü)
  Die Verbindungstabelle zeigt alle zum Projekt angelegten Verbindungen.
- Auswahlbezogene Verbindungen anzeigen (empfohlene Einstellung bei komplexen Projekten)
  Deaktivieren Sie hierzu über das Kontextmenü die Option "Alle Verbindungen anzeigen".
  Die Verbindungstabelle zeigt die zu einem gewählten Subnetz oder einer gewählten CPU bzw. CP zugehörenden Verbindungen. Sie ist also leer, wenn in der Netzsicht keines dieser genannten Objekte angewählt ist.
  Die Selektion mehrerer jeweils gleichartiger Objekte (mehrere Subnetze oder mehrere CPUs/CPs oder mehrere Schnittstellen) gleichzeitig ist ebenfalls möglich. Die Verbindungstabelle zeigt dann alle zugehörenden Verbindungen an.

Grundfunktionen für Tabellen

Die Verbindungstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle selektiv ein- und ausblenden
  Wählen Sie hierzu in der Kopfzeile der Tabelle im Kontextmenü den Befehl "Spalten ein-/ausblenden".
- Spaltenbreite manuell verändern
  Verschieben Sie hierzu in der Kopfzeile die rechts von der Tabellenspalte liegende Trennlinie.
- Spaltenbreite automatisch optimieren
  Wählen Sie hierzu in der Kopfzeile der Tabelle im Kontextmenü den Befehl "Spaltenbreite optimieren".
- Bedeutung einer Spalte über Tooltips anzeigen.

Wenn Sie die Anordnung, Breite und Sichtbarkeit der Tabellenspalten speichern möchten, klicken Sie in der Netzsicht rechts oben auf die Funktion "Anordnung merken".

Symbol	Bedeutung
	Anordnung merken Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.

Mit Cursortasten in der Verbindungstabelle navigieren

Mit den Cursortasten NACH-OBEN und NACH-UNTEN können Sie eine Verbindung in der Verbindungstabelle auswählen; die ausgewählte Verbindung ist markiert und wird in der Netzsicht bei eingeschaltetem Verbindungsmodus hervorgehoben dargestellt.

Eigenschaften der Verbindung ändern

Sie können einige der in der Verbindungstabelle angezeigten Parameter direkt bearbeiten. Der Name der Verbindung beispielsweise nur in der Verbindungstabelle geändert werden.

Verbindungspartner ändern

Sie können den Verbindungspartner einer Verbindung wie folgt ändern:

1. Selektieren Sie die Verbindung.
2. Wählen Sie den neuen Verbindungspartner über die aktivierte Klappliste in der Spalte "Partner".

Siehe auch

Überblick über die Verbindungsdiagnose (Seite 3536)

Verbindungsstatus mittels Symbolen anzeigen (Seite 3537)

Verbindungen im Detail diagnostizieren (Seite 3538)

Verbindungspartner ändern

Für eine bereits projektierte Verbindung ändern Sie bei Bedarf den Verbindungspartner gemäß nachfolgender Beschreibung. Die lokale ID und der Verbindungstyp werden beibehalten.

Änderungen der Verbindungsendpunkte sind nur in der Verbindungstabelle oder dem Inspektorfenster möglich.

Verbindungen dialoggeführt ändern

Um einen anderen Verbindungspartner auf einem Default-Verbindungsweg zu erreichen, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht den lokalen Endpunkt, für den Sie einen neuen Verbindungspartner zuweisen.
   In der Verbindungstabelle werden die zugehörenden Verbindungen angezeigt.
2. Markieren Sie in der Verbindungstabelle die gewünschte Verbindung.
3. Wählen Sie in der Tabellenspalte "Partner" über die Klappliste das neue Zielgerät für die Verbindung aus.
   Alternativ wählen Sie das neue Zielgerät im Inspektorfenster unter "Verbindungsweg".

Hinweis

Auswirkung auf bereits projektierte Eigenschaften

Beachten Sie, dass bei Änderung des Verbindungspartners die parametrierten Eigenschaften der Verbindung auf die Voreinstellung zurückgesetzt werden.

Benutzen Sie die Eingabemöglichkeiten in der Verbindungstabelle oder im Inspektorfenster, wenn Sie die "Eigenschaften" der Verbindung ändern wollen.

Hinweis

Auswirkung auf die Vernetzung

Bei den Vorgängern zum Ändern des Verbindungspartners werden bei fehlender Vernetzung zum neuen Verbindungsendpunkt Subnetze nicht automatisch angelegt.

Verbindungsweg einstellen oder ändern

Je nachdem, welche Komponenten und Schnittstellen innerhalb eines Gerätes zur Verfügung stehen, können Verbindungen über unterschiedliche Wege innerhalb des Gerätes geführt werden.

Je nach Status der Gerätekonfiguration befinden sich Verbindungen in der Projektierung in folgendem Projektierungszustand und werden entsprechend dargestellt:

- Verbindungsweg geschlossen
  Beide Endpunkte (CPU oder PC-Applikation) sowie für die Kommunikation benötigte Schnittstellen (lokal in der CPU oder zusätzlicher CP) sind vorhanden.

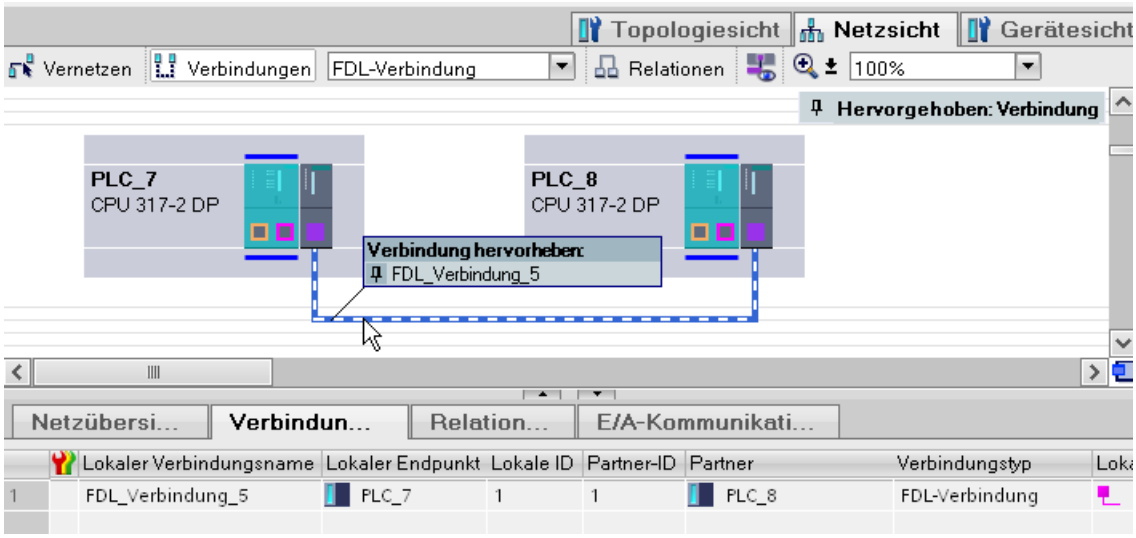

- Verbindungsweg offen
  - Der Verbindungsweg ist aufgrund fehlender Schnittstellen nicht vollständig zuordenbar.
  - Der Verbindungsweg ist aufgrund einer unspezifizierten Verbindung nur in einem Endpunkt festgelegt.

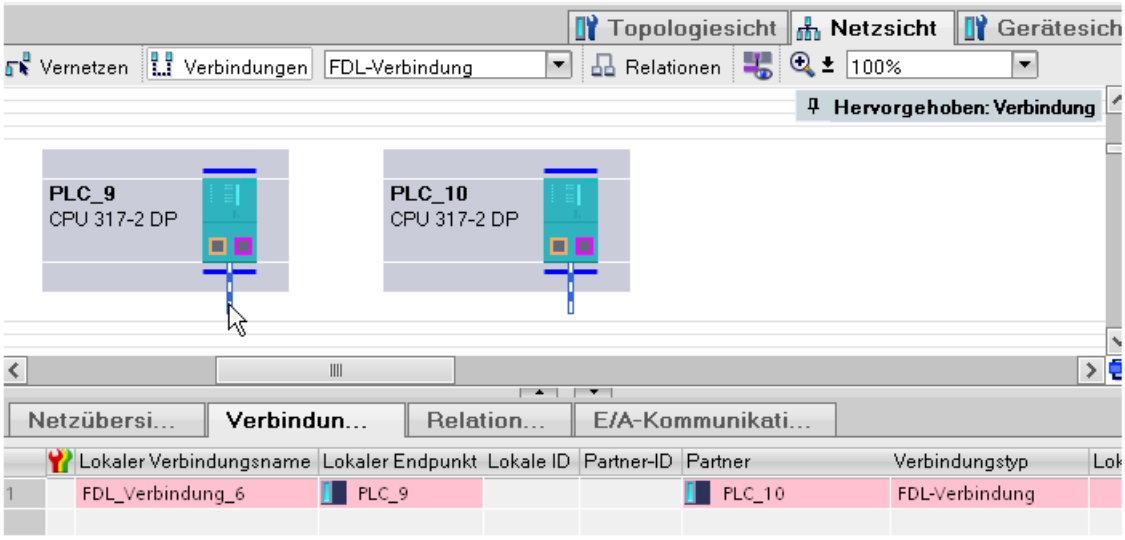

Nachfolgend wird das Vorgehen beschrieben, um einen projektierten Verbindungsweg in einem lokalen oder einem Partner-Gerät zu ändern oder neu zuzuweisen.

Sie können den Verbindungsweg in folgenden Ansichten einstellen oder ändern:

- Inspektorfenster unter "Eigenschaften > Allgemein"

Verbindungsweg im Inspektorfenster ändern

Um einer bestehenden Verbindung einen anderen Verbindungsweg zuzuweisen, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht den Endpunkt (CPU oder PC-Applikation), für den eine Verbindung geändert werden soll.
2. Markieren Sie in der Verbindungstabelle die Verbindung, deren Verbindungsweg Sie ändern wollen.
   Tipp: In der Netzsicht wird der aktuelle Verbindungsweg hervorgehoben dargestellt, wenn Sie die Ansicht "Verbindung hervorheben" entsprechend eingestellt haben.
3. Wählen Sie die neue Schnittstelle im Inspektorfenster unter "Eigenschaften > Allgemein > Schnittstelle" aus. Angeboten werden beim Partner die Schnittstellen, die im Kontext der lokalen Schnittstelle infrage kommen (Anmerkung: S7-Verbindungen können prinzipiell über unterschiedliche Subnetztypen geführt werden).
   In der Netzsicht wird der neue Verbindungsweg hervorgehoben dargestellt.

Verbindungsweg zuweisen

Um einem bestehenden offenen Verbindungsweg einen geschlossenen Verbindungsweg zuzuweisen, gehen Sie folgendermaßen vor:

1. Ergänzen Sie die Gerätekonfiguration so, dass die für den Verbindungstyp benötigten Schnittstellen bei beiden Partnern vorhanden sind.
2. Markieren Sie in der Verbindungstabelle die Verbindung, deren Verbindungsweg Sie zuweisen wollen.
   Tipp: In der Netzsicht wird der aktuelle Verbindungsweg hervorgehoben dargestellt, wenn Sie die Ansicht "Verbindung hervorheben" entsprechend engestellt haben.
3. Betätigen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Schnittstelle" die Schaltfläche "Finde Verbindungsweg".
   In der Netzsicht wird der neue Verbindungsweg hervorgehoben dargestellt.

Unspezifizierte Verbindung anlegen

In folgenden Fällen legen Sie unspezifizierte Verbindungen gezielt an, indem Sie nur den lokalen Verbindungspartner grafisch oder dialoggeführt anwählen:

- Wenn bestimmte Dienst- oder Verbindungseigenschaften eine unspezifizierte Verbindung erfordern (Beispiele: E-Mail-Verbindung, FTP-Verbindung, FDL-Verbindung mit der Eigenschaft "Freier Layer 2 Zugang").
- Wenn eine Verbindung eingerichtet werden soll, die für die Kommunikation mit einem beliebigen Verbindungspartner bereit sein soll.
- Wenn einer der beiden Partner für den gewählten Verbindungstyp (noch) nicht verfügbar ist, Sie aber eine Verbindungsressource reservieren möchten.

- Wenn sich der Verbindungspartner außerhalb des Projektes befindet (z. B. Fremdstationen).
- UDP-Verbindung mit Broadcast / Multicast:
  - siehe UDP mit Broadcast und Multicast (Seite 275)
  - siehe Freie UDP-Verbindung (Seite 279)

Voraussetzung

Sie befinden sich in der Netzsicht.

Unspezifizierte Verbindung grafisch projektieren

Um eine unspezifizierte Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Verbindungsmodus den benötigten Verbindungstyp in der Klappliste, beispielsweise "TCP-Verbindung".

   Die für den gewählten Verbindungstyp infrage kommenden CPUs sind selektiert und hervorgehoben dargestellt.
2. Klicken Sie auf die CPU, von der eine Verbindung ausgehen soll.
3. Bewegen Sie den Mauscursor innerhalb der CPU oder auf die Schnittstelle des CP und bestätigen Sie den Verbindungsendpunkt durch einen Doppelklick mit der Maus.

Ergebnis

- Eine unspezifizierte Verbindung ist angelegt. Je nach gewähltem Endpunkt - CPU oder CP - gilt für die Wegewahl:
  - Endpunkt CPU: Es wird lokal automatisch der bestmögliche Verbindungsweg gewählt. Siehe auch Neue Verbindung grafisch anlegen (Seite 226)
  - Endpunkt CP-Schnittstelle: Der Verbindungsweg ist durch die Eingabe bestimmt.
- Die unspezifizierte Verbindung ist durch eine kurze gestrichelte Linie hervorgehoben dargestellt.

- Die unspezifizierte Verbindung ist in der Verbindungstabelle des Gerätes eingetragen; in der Spalte "Partner" ist "unbekannt" eingetragen.

Unspezifizierte Verbindung dialoggeführt projektieren

Um eine unspezifizierte Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü des Gerätes (CPU), für das eine Verbindung angelegt werden soll, den Befehl "Verbindung hinzufügen".
   oder
   Wählen Sie bei abgewählter Option "Alle Verbindungen anzeigen" im Kontextmenü der Verbindungstabelle den Befehl "Verbindung hinzufügen".
   Ergebnis: Das Dialogfeld "Neue Verbindung anlegen" wird geöffnet.
2. Wählen Sie im Dialogfeld "Neue Verbindung anlegen" den gewünschten Verbindungstyp, z. B. "FDL-Verbindung".
3. Wählen Sie im oberen Bereich des Dialogs den Eintrag "unspezifiziert".
4. Falls mehrere lokale Schnittstellen verfügbar sind, können Sie aus einer Liste die gewünschte Schnittstelle wählen. In der ersten Listenzeile wird die vom System als bestmögliche erkannte Schnittstelle angezeigt und als selektiert dargestellt.
5. Bestätigen Sie die Auswahl über die Schaltfläche "Hinzufügen".
   Eine unspezifizierte Verbindung wird angelegt und ist in der Verbindungstabelle sichtbar.
6. Wiederholen Sie den Vorgang oder verlassen Sie den Dialog über "OK".

Ergebnis

- Eine oder mehrere unspezifizierte Verbindungen sind angelegt.
- Die unspezifizierten Verbindungen sind in der Verbindungstabelle eingetragen.

Hinweis

Beachten Sie, dass Sie für jede Verbindung noch spezielle Eigenschaften parametrieren können.

UDP- / FDL-Verbindungen mit Multicast / Broadcast anlegen

Bedeutung

Multicast und Broadcast sind spezielle Verbindungsoptionen, welche von Industrial Ethernet-CPs bei UDP-Verbindungen und von PROFIBUS bei FDL-Verbindungen unterstützt werden und projektierbar sind.

Verbindung dialoggeführt anlegen - Vorgehen

Um eine Multicast-/Brodcast-Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü der Komponente, die Verbindungsendpunkt sein soll den Befehl "Verbindung hinzufügen".
   oder
   Wählen Sie bei abgewählter Option "Alle Verbindungen anzeigen" im Kontextmenü der Verbindungstabelle den Befehl "Verbindung hinzufügen".
   Der Dialog "Neue Verbindung erstellen" wird geöffnet.
2. Wählen Sie in der Klappliste "Typ" den Verbindungstyp UDP-Verbindung / FDL-Verbindung.
3. Wählen Sie im linken Bereich des Dialogs den Verbindungstyp Broadcast oder Multicast. Im Inhaltsbereich werden die der Konfiguration entsprechend möglichen lokalen Schnittstellen mit Statusinformationen angezeigt. Die erste Schnittstelle, der eine Verbindung vom ausgewählten Typ zugewiesen werden kann, ist selektiert.
4. Belassen Sie es bei der Auswahl oder wählen Sie den gewünschten Verbindungsweg aus.
5. Bestätigen Sie die Auswahl über die Schaltfläche "Hinzufügen".
   Eine Verbindung wird angelegt und ist in der Verbindungstabelle sichtbar.
6. Falls Sie den Dialog nicht abgeschlossen haben, gehen Sie alternativ weiter so vor:
   - Eine weitere Verbindung des selben Typs zum selben Zielgerät legen Sie an, indem Sie erneut die Schaltfläche "Hinzufügen" Klicken.
   - Wiederholen Sie den oben beschriebenen Vorgang mit der Auswahl eines anderen Verbindungstyps aus der Klappliste oder eines anderen Verbindungspartners.

Verbindung grafisch anlegen - Vorgehen

1. Legen Sie zunächst mit der grafischen Verbindungsprojektierung eine unspezifizierte UDP-Verbindung / FDL-Verbindung an; Neue Verbindung grafisch anlegen (Seite 226)
2. Weisen Sie nun die Adresseinstellungen für den Verbindungspartner entsprechend den Anforderungen für Multicast bzw. Broadcast zu. Alternativ gehen Sie so vor:
   - Wählen Sie in der Verbindungstabelle in der Spalte "Partner" den gewünschten Verbindungstyp in der Klappliste aus; Adressparameter werden automatisch zugewiesen.
     oder
   - Tragen Sie bei UDP die IP-Adresse bzw. den Port unter "Allgemein > Adressdetails" direkt ein. Bei FDL stellen Sie die PROFIBUS-Teilnehmeradresse 63 ein.

Siehe auch

UDP mit Broadcast und Multicast (Seite 275)

FDL-Verbindung mit Broadcast (Seite 289)

FDL-Verbindung mit Multicast (Seite 290)

Unspezifizierte Verbindung spezifizieren

Merkmale der unspezifizierten Verbindung

Bei einer unspezifizierten Verbindung fehlen je nach Verbindungstyp ganz oder teilweise Parameter, die für den lokalen Zugang zum Netz und für die Adressierung des Partners benötigt werden.

Eine unspezifizierte Verbindung ist wie folgt dargestellt:

- Netzsicht:
  Nach dem Neuanlegen der Verbindung oder wenn die Verbindung in der Verbindungstabelle markiert ist, erscheint die unspezifizierte Verbindung in der Netzsicht als unterbrochene gestrichelte Linie.
- Verbindungstabelle:
  - Anzeige in der Spalte "Partner": "unbekannt"

Eine unspezifizierte Verbindung entsteht wie folgt: siehe auch Unspezifizierte Verbindung anlegen (Seite 241)

Beim nachträglichen Spezifizieren der Verbindung ist prinzipiell zu unterscheiden:

- Der Verbindungspartner befindet sich im selben Projekt
  Sie spezifizieren die Verbindung durch nachträgliche Selektion von im Projekt vorhandener Komponenten.
- Der Verbindungspartner befindet sich außerhalb des Projektes
  Sie spezifizieren die Verbindung durch Eingabe von Adressparametern der außerhalb des Projektes vorhandener Kommunikationspartner.

Automatische Ergänzung fehlender Verbindungsparameter

Wenn das Fehlen von Ressourcen durch Hinzufügen von Komponenten beseitigt wird, werden Verbindungen wenn möglich automatisch zugewiesen und damit ggf. zu einer vollspezifizierten Verbindung.

Voraussetzung

- Die Komponenten beider Geräte haben freie Ressourcen für den Verbindungstyp.

Vorgehen in der Verbindungstabelle

Um eine unspezifizierte Verbindung mit den Adressparametern eines im Projekt vorhandenen Zielgerätes zu versorgen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU, deren Verbindung spezifiziert werden soll.
   In der Verbindungstabelle sind alle für die CPU projektierten unspezifizierten Verbindungen mit dem Vermerk "unbekannt" in der Spalte "Partner" kenntlich gemacht.
2. Wählen Sie in der Verbindungstabelle in der Spalte "Partner" das gewünschte Zielgerät aus.

Ergebnis

- Die Verbindung ist mit den entsprechenden Adressdetails für den Verbindungspartner spezifiziert.
- Der neue Verbindungsweg wird in der Netzsicht als gestrichelte Linie hervorgehoben dargestellt.

Die folgende Darstellung zeigt eine spezifizierte Verbindung:

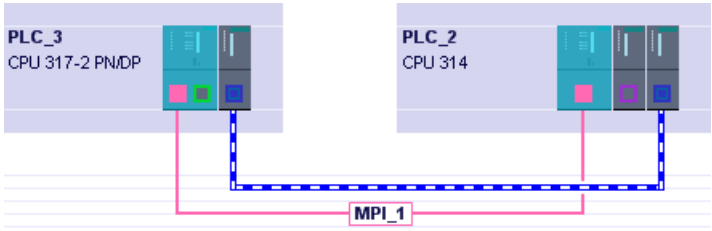

Verbindung löschen

Sie können projektierte Verbindungen über die Netzsicht oder die Verbindungstabelle löschen.

In der Netzsicht können Sie jeweils eine hervorgehobene Verbindung löschen. In der Verbindungstabelle können Sie eine oder mehrere Verbindungen löschen.

Vorgehen in der Netzsicht

Sie können in der Netzsicht einzelne Verbindungen über das Kontextmenü löschen. Gehen Sie folgendermaßen vor:

1. Schalten Sie in den Verbindungsmodus.
2. Zeigen Sie mit dem Mauscursor auf das Subnetz.
   Der Dialog "Verbindung hervorheben" zeigt die aktuell projektierten Verbindungen.
3. Selektieren Sie die zu löschende Verbindung.
4. Zeigen Sie erneut mit dem Mauscursor auf das Subnetz.
5. Öffnen Sie mit der rechten Maustaste das Kontextmenü.
6. Wählen Sie den Befehl "Löschen".
   Die markierte Verbindung wird vollständig gelöscht. Befindet sich bei einer spezifizierten Verbindung der Partner innerhalb des Projektes, wird die Verbindung auch aus der Verbindungstabelle des Partners gelöscht.

Hinweis

Verbindungsdaten in Baugruppe laden

Laden Sie die Verbindungstabelle mit der gelöschten Verbindung in die durch den Löschvorgang betroffenen programmierbaren Baugruppen.

Vorgehen in der Verbindungstabelle / Netzsicht

Sie können in der Verbindungstabelle eine oder mehrere gewählte Verbindungen über das Kontextmenü löschen. Gehen Sie folgendermaßen vor:

1. Markieren Sie in der Verbindungstabelle die Verbindungen, die Sie löschen wollen.
2. Wählen Sie mit der rechten Maustaste das Kontextmenü.
3. Wählen Sie den Befehl "Löschen".
   Die jeweils markierte Verbindung wird vollständig gelöscht. Befindet sich bei einer spezifizierten Verbindung der Partner innerhalb des Projektes, wird die Verbindung auch aus der Verbindungstabelle des Partners gelöscht.
4. Laden Sie die Verbindungstabelle mit den gelöschten Verbindungen in die betreffende programmierbare Baugruppe.

Hinweis

Verbindungsdaten in Baugruppe laden

Laden Sie die Verbindungstabelle mit der gelöschten Verbindung in die durch den Löschvorgang betroffenen Baugruppen.

Hinweis

Um alle Verbindungen der Baugruppe zu löschen, müssen Sie eine leere Verbindungstabelle laden.

Verbindung kopieren

Einführung

Verbindungen werden nicht einzeln kopiert, sondern immer im Kontext mit dem Projekt bzw. mit dem Gerät.

Sie können kopieren:

- gesamte Projekte
- eine oder mehrere Geräte innerhalb eines Projekts oder über Projektgrenzen hinweg

Projekt kopieren

Wenn Sie ein Projekt kopieren, dann werden sämtliche projektierten Verbindungen mitkopiert. Neue Einstellungen für die kopierten Verbindungen sind nicht notwendig, da die Verbindungen konsistent bleiben.

Geräte kopieren

- Gerät innerhalb eines Projekts kopieren
  Wenn Sie einzelne Geräte kopieren, dann entstehen unspezifizierte Verbindungen und Sie müssen dem lokalen Teilnehmer die Verbindungspartner neu zuordnen.
  Wenn Sie eine Gruppe von Geräten kopieren, werden die Verbindungen innerhalb der Gerätegruppe mitkopiert. Hierbei entstehen jedoch durch das Duplizieren Duplikate der Verbindungsnamen. Ändern Sie diese Verbindungsnamen nach dem Kopiervorgang.
  Die kopierten Geräte müssen Sie erneut vernetzen.
- Geräte in ein anderes Projekt kopieren
  Sie können ein oder mehrere Geräte in ein anderes Projekt kopieren. Zum parallelen Bearbeiten des weiteren Projektes starten Sie hierzu das TIA Portal erneut.
  Zwischen den kopierten Geräten projektierte Verbindungen bleiben im Zielprojekt erhalten. Im Zielprojekt weisen Sie lediglich die Vernetzung erneut zu.
  Wenn Sie die Geräte nacheinander ins Zielprojekt kopieren, entstehen hierbei temporär unspezifizierte Verbindungen. Beim Einfügen eines weiteren Geräts werden jedoch die Kommunikationspartner, sofern möglich, wieder zugewiesen.

Weitere Informationen

Siehe auch Unspezifizierte Verbindung spezifizieren (Seite 245)

Inkonsistente Verbindungen - Verbindungen ohne Zuordnung

Bei einer inkonsistenten Verbindung sind die Verbindungsdaten unvollständig oder unschlüssig; die Verbindung ist im Kontext mit dem Projekt nicht funktionsfähig.

Inkonsistente Verbindungen können nicht geladen werden - der Betrieb mit einer solchen Verbindung ist nicht möglich.

In der Verbindungstabelle sind inkonsistente Verbindungen durch die Farbe Rot zu erkennen.

Hinweis

Unspezifizierte Verbindungen

Unspezifizierte Verbindungen sind nicht generell inkonsistent obwohl diese in der Verbindungstabelle grundsätzlich rot markiert sind.

Unspezifizierte Verbindung anlegen (Seite 241)

Mögliche Ursachen für inkonsistente Verbindungen

- Löschen oder Änderungen der Hardware-Konfiguration
- Fehlende Vernetzungen von Schnittstellen im Projekt, die für eine Verbindung notwendig sind.
- Überschreitung von Verbindungsressourcen
- Verbindungen zu einem unspezifizierten Verbindungspartner ohne Angabe der Partneradresse.

Detailinformationen über die Ursache der Inkonsistenz finden Sie im Inspektorfenster in der Eigenschaften-Seite einer Verbindung; inkonsistente Parameter sind dort rot markiert. Zu diesen Parametern erhalten Sie eine Zusatzinformation, indem Sie den Mauscursor auf dem Parameter platzieren.

Beispiele und Abhilfe für Inkonsistenzen bei typischen Aktionen in der Projektierung

Nachfolgend werden die Aktionen erläutert, die dazu führen können, dass projektierte Verbindungen ihre Zuordnung verlieren oder gelöscht werden.

Hinweis

Beachten Sie, dass im Gegensatz zu den S7-Verbindungen den Verbindungen der SEND/RECEIVE-Schnittstelle eine CP-abhängige ID zugewiesen wird. Bei den nachfolgend beschriebenen Aktionen kann es daher zu Anpassungen der ID kommen, so dass im Anwenderprogramm die Schnittstelle ebenfalls angepasst werden muss.

Wird ein CP durch einen anderen ersetzt, so muss dieser mindestens die gleichen Dienste bereitstellen und mindestens den gleichen Versionsstand haben. Nur so ist gewährleistet, dass die über den CP projektierten Verbindungen konsistent erhalten bleiben und genutzt werden können.

Tabelle 1-3 Aktionen bei einer CP-Schnittstelle, die zu Änderungen an projektierten Verbindungen führen

Aktion	Folge für die Verbindungen	Was Sie tun müssen, um die Verbindung wieder herzustellen
Den CP (Baugruppe) löschen.	Die Verbindungen bleiben ohne Zuordnung zu einem CP in der Verbindungstabelle erhalten. Die Zuordnung zur CPU / PC-Applikation bleibt erhalten.	Nachdem Sie einen CP in der Hardware-Konfiguration platziert und vernetzt haben: 1. Den CP der Verbindung zuordnen; 2. Die Baugruppen-Anfangsadresse LADDR prüfen und ggf. im Anwenderprogramm anpassen. 3. Verbindungs-IDs im Anwenderprogramm anpassen. 4. Verbindungsprojektierung erneut in den CP laden.
Gerät löschen.	Sämtliche Verbindungen zu dieserm Gerät werden innerhalb des Projekts aufgelöst: • Verbindungen bleiben beim Partner als unspezifizierte Verbindungen erhalten. • Verbindungen ohne Verbindungspartner werden komplett gelöscht.	Gerät und Verbindungen neu projektieren. Unspezifizierte Verbindungen erneut zuordnen oder löschen.
Die Subnetzzuordnung des CP ändern.	Die Verbindungen, die über den CP zugeordnet waren, bleiben erhalten, können inkonsistent werden.	Verbindungen ggf. neu zuweisen.

1.3.2.4 Verbindungstypen

S7-Verbindung

S7-Verbindung Allgemeine Einstellungen

Allgemeine Verbindungsparameter

In der Parametergruppe "Allgemein" unter den Eigenschaften der Verbindung werden allgemeine Verbindungsparameter angezeigt, die den lokalen Verbindungsendpunkt identifizieren.

Sie haben hier die Möglichkeit, den Verbindungsweg zuzuordnen und die Verbindungspartner vollständig zu spezifizieren.

Lokale ID

Hier wird die lokale ID der Baugruppe angezeigt, von der aus die Verbindung betrachtet wird (lokaler Partner). Sie können die lokale ID ändern. Das ist dann notwendig, wenn Sie bereits Kommunikations-Funktionsbausteine programmiert haben und Sie die dort angegebene Lokale ID für die Verbindung verwenden wollen.

Besondere Verbindungseigenschaften

Anzeige der Verbindungseigenschaften (in Abhängigkeit der verwendeten Komponenten änderbar):

- Einseitig
  Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.
  Zweiseitige Kommunikationsanweisungen ("Bausteinpärchen") sind bei einseitigen Verbindungen nicht einsetzbar; wie USEND und URCV, BSEND und BRCV oder USTATUS.
- Aktiver Verbindungsaufbau
  Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von diesem Gerät aus erfolgen soll.
  Wenn Sie beim Anlegen der Verbindung als Verbindungspartner "unspezifiziert" gewählt haben, ist die Option standardmäßig abgewählt. Wenn Sie die Option aktivieren, müssen Sie die Adresse des Partners im Register "Adressen" spezifizieren.
- Betriebszustandsmeldungen senden
  Zeigt an, ob der lokale Partner Betriebszustandsmeldungen an den Verbindungspartner sendet.

Adressdetails

Anzeige der Adressdetails der S7-Verbindung. Bei einem unspezifizierten Partner können die Werte für Baugruppenträger und Steckplatz geändert werden. Alle anderen Werte werden aus der aktuellen Konfiguration ermittelt und sind nicht änderbar.

S7-Verbindung Adressdetails

Bedeutung

Eine Verbindungsressource, wie sie unter "Eigenschaften > Adressdetails" angezeigt wird, ist Teil des TSAP (Transport Service Access Point) der lokalen Baugruppe bzw. des Partners.

Wenn eine Verbindung aufgebaut wird, dann werden verbindungsspezifische Ressourcen einer Baugruppe dieser Verbindung fest zugeordnet. Diese Zuordnung setzt voraus, daß die Verbindungsressource adressiert werden kann. Der TSAP (Transport Service Access Point) ist quasi die Adresse der Ressource, die unter anderem mit Hilfe der Verbindungsressource oder bei S7-1500 CPUs mit der SIMATIC-ACC (SIMATIC Application Controlled Communication) gebildet wird.

Es bestehen Einschränkungen und Regeln hinsichtlich der Verwendbarkeit der Verbindungsressourcen. Nicht jede Verbindungsressource ist z. B. für jeden Verbindungstyp verwendbar.

Aufbau des TSAPs

- Für S7-1500 CPU:
  "SIMATIC-ACC"<nnn><mm>
  nnn = lokale ID
  mm = freier Wert
- Für S7-300/400 CPU:
  <xx>.<yz>
  xx = Nummer der Verbindungsressource
  y = Baugruppenträgernummer
  z = Steckplatznummer

TSAP-Aufbau in Abhängigkeit des Partners

Der Aufbau des TSAPs ist bei S7-1500 CPUs abhängig von dem fernen Verbindungspartner. Wenn eine S7-1500 CPU mit einer S7-300/400 CPU verbunden wird, verwendet auch eine S7-1500 CPU den TSAP-Aufbau mit Einbeziehung der Verbindungsressource.

Siehe folgende Beispiele für TSAPs verschiedener Verbindungsprojektierungen

- Verbindung zwischen zwei S7-1500 CPUs:
  - S7-1500 CPU "A" mit lokaler ID 200:
    TSAP: SIMATIC-ACC20001
  - S7-1500 CPU "B" mit lokaler ID 3B2:
    TSAP: SIMATIC-ACC3B201
- Verbindung zwischen S7-1500 CPU und S7-300/400 CPU:
  - S7-1500 CPU (Baugruppenträger 0, Steckplatz 1, Verbindungsressource 10):
    TSAP: 10.01
  - S7-300/400 CPU (Baugruppenträger 0, Steckplatz 2, Verbindungsressource 11):
    TSAP: 11.02

Wertebereiche der Verbindungsressource

Die folgende Tabelle gibt Auskunft über die Bedeutung der Werte und den Typ der Verbindung. Je nach Verbindungspartner und Verbindungstyp wird der Wertebereich automatisch auf gültige Werte eingeschränkt oder der Wert der Verbindungsressource wird fest vergeben.

Verbindungsressource (1.Byte des TSAP)	Bedeutung	Typ	Bedeutung
0x01 ("PG")	PG-Verbindung	Freie Verbindung (nicht projektiert)	Eine Ressource ist für PG-Verbindungen imindestens pro CPU reserviert. Es können aber bei bestimmten S7-300-CPUs mehrere Ressourcen in den CPU-Eigenschaften reserviert werden.
0x02 ("OP")	OP-Verbindung	Freie Verbindung (nicht projektiert)	Eine Ressource ist für OP-Verbindungen imindestens pro CPU reserviert. Es können aber bei bestimmten S7-300-CPUs mehrere Ressourcen in den CPU-Eigenschaften reserviert werden.
0x03	sonstige	Freie Verbindung (Projektierte unspezifizierte Verbindung)	Diese Verbindungsressource kann mehrere Verbindungen bedienen. Verwendung: einseitige Verbindung bei unspezifiziertem Verbindungspartner! Der Verbindungspartner braucht nicht projektiert zu werden, wenn die Verbindungsressource 0x03 adressiert wird.
0x10..0xDF	Verbindungen mit statischem oder dynamischem Verbindungsaufbau	Projektierte unspezifizierte Verbindung	Eine dieser Verbindungsressourcen kann genau eine Verbindung bedienen. Verwendung: zweiseitige Verbindung bei unspezifiziertem Verbindungspartner!

Freie S7-Verbindungen

Freie S7-Verbindungen werden zur Laufzeit dynamisch aufgebaut; es gibt folgende freien Verbindungen:

- PG-Verbindungen (0x01)
  S7-Verbindungen, die typischerweise von einem PG oder von einem PC (mit ES-Funktionalität) aufgebaut werden. Über eine solche Verbindung wird die adressierte Station/Baugruppe projektiert und programmiert, getestet und in Betrieb genommen; danach wird die Verbindung i.d.R. wieder abgebaut. Über diese Verbindungsressource kann sowohl gelesen als auch geschrieben werden (z. B. Beobachten und Laden).
- OP-Verbindungen (0x02)
  S7-Verbindungen, die typischerweise von einem OP oder von einem PC (mit OS-Funktionalität) aus aufgebaut werden. Über eine solche Verbindung wird die adressierte Station/Baugruppe bzgl. des zu steuernden Prozesses beobachtet.
- Sonstige (0x03)
  Nutzung ist nicht festgelegt. Bei der Projektierung einer einseitigen S7-Verbindung von einer S7-400 zu einer S7-300 wird z. B. diese Ressource automatisch verwendet.
- S7-Basis-Kommunikation (0xFD)
  Verbindungen, die typischerweise von einer CPU zu einer weiteren Baugruppe (CPU, FM, ...) innerhalb eines Subnetzes aufgebaut werden. Der Verbindungsaufbau wird durch das Anwendungsprogramm initiiert, wobei keine Verbindungsprojektierung vorliegt. Über eine solche Verbindung können Prozessdaten zwischen den Baugruppen ausgetauscht werden. Bei bestimmten S7-300-CPUs können Ressourcen für S7-Basis-Kommunikation reserviert werden.

Empfehlung für die Wahl der Verbindungsressource

Für den lokalen Partner wird von STEP 7 immer eine freie Verbindungsressource vorgeschlagen. Wir empfehlen Ihnen, diese zu übernehmen.

ISO-Transportverbindung

ISO-Transport - Besondere Verbindungseigenschaften

Bezug

In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften" können Sie zusätzliche Eigenschaften des lokalen Verbindungsendpunkts festlegen.

Aktive Verbindung aufbauen

Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von diesem Gerät aus erfolgen soll. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.

- EIN: Verbindung wird aktiv aufgebaut.
- AUS: Verbindung wird vom Partner aufgebaut.

Wenn Sie beim Anlegen der Verbindung als Verbindungspartner "unspezifiziert" gewählt haben, ist die Option standardmäßig abgewählt. Wenn Sie die Option aktivieren, müssen Sie die Adresse des Partners im Register "Adressen" spezifizieren.

Hinweis

Auswirkung auf die Betriebsart

Beachten Sie die Auswirkung auf die Betriebsart.

- S7-300 / S7-400
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" deaktivieren.
- OPC-Server
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" aktivieren.

Name

Beim Anlegen der Verbindung wird hier ein Vorschlagsname für den Verbindungsendpunkt angezeigt, der als Suffix eine Verbindungsnummer enthält.

Nutzen Sie dieses Feld bei unspezifizierten Verbindungen zur Kennzeichnung des Partners.

ISO OPC-Aufbaudetails

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > Aufbaudetails".

Optionen des Verbindungsaufbaus

Eingaben sind in diesem Bereich nur möglich, wenn Sie unter "Besondere Verbindungseigenschaften" den aktiven Verbindungsaufbau gewählt haben.

Wählen Sie aus den folgenden beiden Alternativen:

- Die Verbindung dauerhaft aufrecht erhalten
  Die Verbindung wird mit dem Hochlauf der PC-Station sowie nach einer Verbindungsunterbrechung aufgebaut und aufrecht erhalten.
- Die Verbindung auf Anforderung aufbauen (Zugriff auf Variable)
  Sie können hier zusätzlich angeben, wie lange die Verbindung nach einem Zugriff gehalten werden soll.

Überwachung des Verbindungsaufbaus

- Zeitüberschreitung: Fehlerwartezeit während des Verbindungsaufbaus
  Kann innerhalb dieser Wartezeit die Verbindung nicht aufgebaut werden, so wird die Verbindung zurückgesetzt und ein anstehender Auftrag mit einem Fehler quitttiert.
  Wertebereich: 0...99999 ms
  Default: 15000 ms

Hinweis

Die hier einstellbare Verbindungsaufbau-Überwachung steht wie folgt im Zusammenhang mit der in der Baugruppenprojektierung einstellbaren Zeitüberwachung: gültig ist der kleinere Wert.

In der Baugruppenprojektierung ist die Zeitüberwachung unter "Eigenschaften Ethernet Schnittstelle" projektierbar.

ISO-Transport Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Adressparameter

Eine ISO-Transport Verbindung wird durch den lokalen und fernen Verbindungsendpunkt spezifiziert .

- Lokale Adressen:
  Lokale MAC-Adresse und lokaler TSAP (Transport Service Access Point)
- Ferne Adressen:
  Ferne MAC-Adresse und ferner TSAP

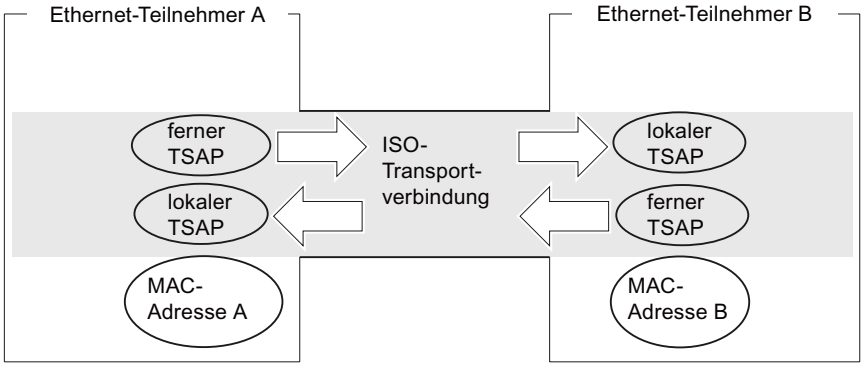

Bild 1-4 Senden und Empfangen über jeweils eine ISO-Transportverbindung

TSAP-Format

ISO-Transportverbindungen haben eine TSAP-Länge von 1 bis 16 Byte. Bei der Eingabe wird die aktuelle Länge automatisch angezeigt (sichtbare Anzeige: 16 ASCII-Zeichen). Lokale und ferne TSAPs können als Hexadezimal-Wert oder als ASCII-String eingegeben werden. Bei ASCII-Eingabe werden die eingegebenen Zeichen auch hexadezimal angezeigt. Bei Hexadezimal-Eingabe werden druckbare Zeichen als ASCII-Wert (sichtbar sind 8 Hexadezimalzeichen) dargestellt. Werden nicht druckbare Zeichen eingegeben, so wird die ASCII-Anzeige gegraut (keine ASCII-Eingabe mehr möglich) und die nicht druckbaren Zeichen werden als Punkt dargestellt.

Lokale und ferne TSAPs

Ferne und lokale TSAPs können identisch sein, da die Verbindung durch die unterschiedlichen MAC-Adressen eindeutig ist. Wenn zwischen zwei Stationen mehr als eine Verbindung eingerichtet werden soll, müssen sich die TSAPs unterscheiden.

- Spezifizierte Verbindung
  Die TSAPs einer ISO-Transportverbindung müssen wie folgt übereinstimmen:
  - Ferner TSAP (im Ethernet-CP) = lokaler TSAP (in Ziel-Station)
  - Lokaler TSAP (im Ethernet-CP) = ferner TSAP (in Ziel-Station)
- Unspezifizierte Verbindung

Adressparameter werden teilweise nicht festgelegt: Unspezifizierte Verbindungen (Seite 217)

Default TSAPs

Bei der Projektierung der lokalen und fernen TSAPs werden Standardwerte (änderbar) vorgeschlagen (z. B. ISO-1 für die erste Verbindung zwischen zwei Partnern) . Werden zwischen den selben Partnern neue Verbindungen projektiert, werden die Standardwerte automatisch hochgezählt (z. B. ISO-2 usw.). Bei einer neuen Verbindung zu einem neuen Partner wird erneut ISO-1 begonnen.

ISO-Transport Optionen

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Optionen".

Betriebsarten

- **SEND/RECV:**
  Dies ist der Standardbetrieb mit den FCs AG_SEND/AG_LSEND und AG_RECV/AG_LRECV.
- **FETCH/WRITE:**
  Wenn Sie eine der folgenden Betriebsarten für die ISO-Transportverbindung wählen, ist der Zugriff auf die Systemspeicherbereiche in der SIMATIC S7 von folgenden Geräten aus möglich:
  - SIMATIC S5
  - SIMATIC PC-Stationen
  - Fremdgeräten

  Die Verbindung kann dann ausschließlich für diese Betriebsart genutzt werden. Senden oder Empfangen über die Anweisungen AG_SEND/AG_LSEND bzw. AG_RECV/AG_LRECV ist dann nicht möglich.
  - FETCH
    Indem Sie die Betriebsart Fetch wählen, ermöglichen Sie den direkten, lesenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
  - WRITE
    Indem Sie die Betriebsart Write wählen, ermöglichen Sie den direkten, schreibenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
- **SPEED SEND/RECV:**
  Indem Sie die Betriebsart SPEED SEND/RECV wählen, ermöglichen Sie die Verwendung der Anweisungen AG_SSEND und AG_SRECV.
  AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) bieten eine optimierte (da schnellere) Übertragung innerhalb einer S7-400-Station. Informieren Sie sich anhand der Gerätedokumentation des eingesetzten CP, ob diese Betriebsart durch den CP und zusammen mit der gewählten CPU unterstützt wird.

Voraussetzung für FETCH/WRITE: Art des Verbindungsaufbaus

Wählen Sie bitte abhängig vom Stationstyp für den Verbindungsaufbau folgende Option In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften":

- SIMATIC S7-Station: PASSIVER Verbindungsaufbau
- PC-Station: Aktiver Verbindungsaufbau

Option "S7-Adressierungsmodus" (nur bei FETCH/WRITE)

Bei der Projektierung für die Betriebsart FETCH / WRITE können Sie hier den Adressierungsmodus wählen.

- S7-Adressierungsmodus: Byte-Adresse
- S5-Adressierungsmodus: Wort-Adresse

Es ist dadurch möglich, dass Applikationen ohne Anpassung der Adressen auf S5- oder S7-Stationen zugreifen. Dies ist beispielsweise besonders für bestehende S5-Applikationen interessant, die jetzt unverändert für den Zugriff auf S7-Stationen verwendet werden sollen.

Standardmäßig eingestellt ist der Adressierungsmodus für den Zugriff auf SIMATIC S7 (Option angewählt).

Siehe auch

Verbindungsprojektierung für FETCH/WRITE (Seite 283)

Anwendung von FETCH/WRITE (Seite 281)

ISO-Transport - Dynamik

Bezugnahme

Verbindungsprojektierung In der Parametergruppe "Eigenschaften > Allgemein > Dynamik".

Angezeigt werden die relevanten Timer und Counter der Verbindung. Sie können diese Defaultwerte übernehmen.

Bei Bedarf (z.B. bei Kopplungen zu Fremdsystemen) können Sie die Timer und Counter und damit das dynamische Verhalten der Verbindung auch individuell einstellen.

ISO OPC-Eigenschaften

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > OPC".

Stellen Sie hier verbindungsspezifische Eigenschaften für eine vom OPC-Server genutzte ISO-Transportverbindung ein.

Folgende Parameter sind hier einstellbar:

Zeitüberschreitung für Auftrag

Kann innerhalb der hier eingestellten Wartezeit ein ausstehender Auftrag nicht bearbeitet werden, so wird die Verbindung zurückgesetzt und der Auftrag mit einem Fehler quittiert.

Wertebereich: 0..99999 ms

- Default-Einstellung: 15000 ms

Größe Sendepuffer

Voreingestellte Größe des Default-Sendepuffers. Zur Laufzeit können weitere zusätzliche Sendepuffer beliebiger Länge (maximal 65535 Byte) verwendet werden.

Wertebereich:

- Default-Einstellung: 256 Byte
- Wertebereich: 65535 Byte maximal

Option "Optimierung Direktzugriff Variablendienste"

Indem Sie diese Option nutzen, versucht der OPC-Server direkte Zugriffe auf mehrere Variablen zusammenzufassen, so dass möglichst wenige Aufträge übertragen werden müssen.

Standardmäßig ist die Option eingeschaltet.

Option "Sofortige Rückmeldung bei erkannter Verbindungsunterbrechung"

Indem sie diese Option wählen legen Sie fest, dass die Applikation bei nicht verfügbarer Verbindung sofort eine Anzeige über einen nicht ausführbaren Kommunikationsauftrag (FETCH oder WRITE) erhält. Die Anzeige erfolgt unabhängig davon, ob ein gerade laufender Versuch, die Verbindung erneut aufzubauen, gelingt oder nicht.

Wenn sie diese Option nicht wählen, erhält die Applikation im Falle einer nicht verfügbaren Verbindung erst dann eine Anzeige, wenn der Verbindungs-Timeout abgelaufen ist.

Die hier beschriebene Option ist nur dann relevant und verfügbar, wenn die beiden nachfolgend genannten Betriebsarten verwendet werden:

- Die Verbindung ist als dauerhafte Verbindung projektiert.
  Verbindungen können als dauerhaft aufgebaute Verbindung projektiert sein (unter "Details zum Verbindungsaufbau"). Dadurch können Aufträge mit kürzest möglicher Reaktionszeit zum Partner weitergeleitet werden, ohne dass zunächst ein Verbindungsaufbau erforderlich ist.
  Ist die Verbindung gestört, wird automatisch versucht, die Verbindung neu aufzubauen. In diesem Zustand kann es zu Verzögerungen kommen, bis eine entsprechende Meldung an die Applikation weitergeleitet wird.
- Sie verwenden die Auftragsarten FETCH oder WRITE (Auswahlmöglichkeit unter "Optionen".

Nur bei diesen Auftragsarten ist in bestimmten Anwendungsfällen eine unmittelbare Reaktion bei fehlender Verbindung erforderlich.

ISO-on-TCP-Verbindung

ISO-on-TCP - Besondere Verbindungseigenschaften

Bezug

In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften" können Sie zusätzliche Eigenschaften des lokalen Verbindungsendpunkts festlegen.

Aktiver Verbindungsaufbau

Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von diesem Gerät aus erfolgen soll. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.

- EIN: Verbindung wird aktiv aufgebaut.
- AUS: Verbindung wird vom Partner aufgebaut.

Wenn Sie beim Anlegen der Verbindung als Verbindungspartner "unspezifiziert" gewählt haben, ist die Option standardmäßig abgewählt. Wenn Sie die Option aktivieren, müssen Sie die Adresse des Partners im Register "Adressen" spezifizieren.

Hinweis

Auswirkung auf die Betriebsart

Beachten Sie die Auswirkung auf die Betriebsart.

- S7-300 / S7-400
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" deaktivieren.
- OPC-Server
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" aktivieren.

Name

Beim Anlegen der Verbindung wird hier ein Vorschlagsname für den Verbindungsendpunkt angezeigt, der als Suffix eine Verbindungsnummer enthält.

Nutzen Sie dieses Feld bei unspezifizierten Verbindungen zur Kennzeichnung des Partners.

ISO-on-TCP OPC-Aufbaudetails

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > Aufbaudetails".

Optionen des Verbindungsaufbaus

Eingaben sind in diesem Bereich nur möglich, wenn Sie unter "Besondere Verbindungseigenschaften" den aktiven Verbindungsaufbau gewählt haben.

Wählen Sie aus den folgenden beiden Alternativen:

- Die Verbindung wird permanent aufrecht erhalten
  Die Verbindung wird mit dem Hochlauf der PC-Station sowie nach einer Verbindungsunterbrechung aufgebaut und aufrecht erhalten.
- Die Verbindung wird bei Bedarf aufgebaut (Zugriff auf Variable)
  Sie können hier zusätzlich angeben, wie lange die Verbindung nach einem Zugriff gehalten werden soll.

Verbindungsaufbau-Überwachung

Zeitüberschreitung: Fehlerwartezeit während des Verbindungsaufbaus

Kann innerhalb dieser Wartezeit die Verbindung nicht aufgebaut werden, so wird die Verbindung zurückgesetzt und ein anstehender Auftrag mit einem Fehler quitttiert.

Wertebereich: 0...99999 ms

Default: 15000 ms

Hinweis

Die hier einstellbare Verbindungsaufbau-Überwachung steht wie folgt im Zusammenhang mit der in der Baugruppenprojektierung einstellbaren Zeitüberwachung: gültig ist der kleinere Wert.

In der Baugruppenprojektierung ist die Zeitüberwachung unter "Eigenschaften Ethernet Schnittstelle" projektierbar.

ISO-on-TCP Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Adressparameter

Eine Verbindung wird durch den lokalen und fernen Verbindungsendpunkt spezifiziert.

- Lokale Adressen:
  Lokale IP-Adresse und lokaler TSAP (Transport Service Access Point)
- Ferne Adressen:
  Ferne IP-Adresse und ferner TSAP

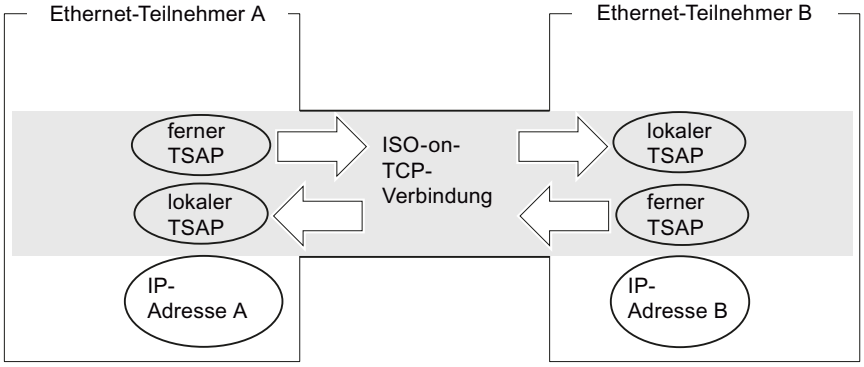

Bild 1-5 Senden und Empfangen über jeweils eine ISO-on-TCP-Verbindung

Register Adressen

Im Register Adressen werden die relevanten lokalen und fernen Adressinformationen als Vorschlagswerte angezeigt. Sie haben die Möglichkeit die TSAPs individuell einzustellen.

TSAP-Format

ISO-on-TCP-Verbindungen haben eine TSAP-Länge von 1 bis 16 Byte. Bei der Eingabe wird die aktuelle Länge automatisch angezeigt (sichtbare Anzeige: 16 ASCII-Zeichen). Lokale und ferne TSAPs können als Hexadezimal-Wert oder als ASCII-String eingegeben werden. Bei ASCII-Eingabe werden die eingegebenen Zeichen auch hexadezimal angezeigt. Bei Hexadezimal-Eingabe werden druckbare Zeichen als ASCII-Wert (sichtbar sind 8 Hexadezimalzeichen) dargestellt. Werden nicht druckbare Zeichen eingegeben, so wird die ASCII-Anzeige gegraut (keine ASCII-Eingabe mehr möglich) und die nicht druckbaren Zeichen werden als Punkt dargestellt.

Lokale und ferne TSAPs

Ferne und lokale TSAPs können identisch sein, da die Verbindung durch die unterschiedlichen IP-Adressen eindeutig ist. Wenn zwischen zwei Stationen mehr als eine Verbindung eingerichtet werden soll, müssen sich auch die TSAPs unterscheiden.

- Spezifizierte Verbindung
  Bei der Projektierung des Ethernet-CP und in der Ethernet-Zielstation müssen die TSAPs einer ISO-on-TCP-Verbindung kreuzweise übereinstimmen:
  - Ferner TSAP (im Ethernet-CP) = lokaler TSAP (in Ziel-Station)
  - Lokaler TSAP (im Ethernet-CP) = ferner TSAP (in Ziel-Station)
- Unspezifizierte Verbindung
  Adressparameter werden teilweise nicht festgelegt: Unspezifizierte Verbindungen (Seite 217)

Default TSAPs

Für die Projektierung der lokalen und fernen TSAPs gibt es einen Standardwert "TCP-1" für die erste Verbindung zwischen beiden Partnern (änderbar). Bei einer neuen Verbindung zwischen den selben Partnern wird der Standardwert "TCP-2" vorgeschlagen. Bei einer neuen Verbindung zu einem neuen Partner wird erneut TCP-1 verwendet.

ISO-on-TCP Optionen

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Optionen".

Betriebsarten

- **SEND/RECV:**
  Dies ist der Standardbetrieb mit den FCs AG_SEND/AG_LSEND und AG_RECV/AG_LRECV.
- **FETCH/WRITE:**
  Wenn Sie eine der folgenden Betriebsarten für die ISO-Transportverbindung wählen, ist der Zugriff auf die Systemspeicherbereiche in der SIMATIC S7 von folgenden Geräten aus möglich:
  - SIMATIC S5
  - SIMATIC PC-Stationen
  - Fremdgeräten

  Die Verbindung kann dann ausschließlich für diese Betriebsart genutzt werden. Senden oder Empfangen über die Anweisungen AG_SEND/AG_LSEND bzw. AG_RECV/AG_LRECV ist dann nicht möglich.
  - FETCH
    Indem Sie die Betriebsart Fetch wählen, ermöglichen Sie den direkten, lesenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
  - WRITE
    Indem Sie die Betriebsart Write wählen, ermöglichen Sie den direkten, schreibenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
- **SPEED SEND/RECV:**
  Indem Sie die Betriebsart SPEED SEND/RECV wählen, ermöglichen Sie die Verwendung der Anweisungen AG_SSEND und AG_SRECV.
  AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) bieten eine optimierte (da schnellere) Übertragung innerhalb einer S7-400-Station. Informieren Sie sich anhand der Gerätedokumentation des eingesetzten CP, ob diese Betriebsart unterstützt wird.

Voraussetzung für FETCH/WRITE: Art des Verbindungsaufbaus

Wählen Sie bitte abhängig vom Stationstyp für den Verbindungsaufbau folgende Option In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften":

- SIMATIC S7-Station: PASSIVER Verbindungsaufbau
- PC-Station: Aktiver Verbindungsaufbau

Option "S7-Adressierungsmodus" (nur bei FETCH/WRITE)

Bei der Projektierung für die Betriebsart FETCH / WRITE können Sie hier den Adressierungsmodus wählen.

- S7-Adressierungsmodus: Byte-Adresse
- S5-Adressierungsmodus: Wort-Adresse

Es ist dadurch möglich, dass Applikationen ohne Anpassung der Adressen auf S5- oder S7-Stationen zugreifen. Dies ist beispielsweise besonders für bestehende S5-Applikationen interessant, die jetzt unverändert für den Zugriff auf S7-Stationen verwendet werden sollen.

Standardmäßig eingestellt ist der Adressierungsmodus für den Zugriff auf SIMATIC S7 (Option angewählt).

Siehe auch

Verbindungsprojektierung für FETCH/WRITE (Seite 283)

Anwendung von FETCH/WRITE (Seite 281)

ISO-on-TCP OPC-Eigenschaften

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > OPC".

Stellen Sie hier verbindungsspezifische Eigenschaften für eine vom OPC-Server genutzte ISO-on-TCP-Verbindung ein.

Folgende Parameter sind hier einstellbar:

Zeitüberschreitung für Auftrag

Kann innerhalb der hier eingestellten Wartezeit ein ausstehender Auftrag nicht bearbeitet werden, so wird die Verbindung zurückgesetzt und der Auftrag mit einem Fehler quittiert.

Wertebereich: 0..99999 ms

- Default-Einstellung: 15000 ms

Größe Sendepuffer

Voreingestellte Größe des Default-Sendepuffers. Zur Laufzeit können weitere zusätzliche Sendepuffer beliebiger Länge (maximal 65535 Byte) verwendet werden.

Wertebereich:

- Default-Einstellung: 256 Byte

- Wertebereich: 65535 Byte maximal

Option "Optimierung Direktzugriff Variablendienste"

Indem Sie diese Option nutzen, versucht der OPC-Server direkte Zugriffe auf mehrere Variablen zusammenzufassen, so dass möglichst wenige Aufträge übertragen werden müssen.

Standardmäßig ist die Option eingeschaltet.

Option "Sofortige Rückmeldung bei erkannter Verbindungsunterbrechung"

Indem sie diese Option wählen legen Sie fest, dass die Applikation bei nicht verfügbarer Verbindung sofort eine Anzeige über einen nicht ausführbaren Kommunikationsauftrag (FETCH oder WRITE) erhält. Die Anzeige erfolgt unabhängig davon, ob ein gerade laufender Versuch, die Verbindung erneut aufzubauen, gelingt oder nicht.

Wenn sie diese Option nicht wählen, erhält die Applikation im Falle einer nicht verfügbaren Verbindung erst dann eine Anzeige, wenn der Verbindungs-Timeout abgelaufen ist.

Die hier beschriebene Option ist nur dann relevant und verfügbar, wenn die beiden nachfolgend genannten Betriebsarten verwendet werden:

- Die Verbindung ist als dauerhafte Verbindung projektiert.
  Verbindungen können als dauerhaft aufgebaute Verbindung projektiert sein (unter "Details zum Verbindungsaufbau"). Dadurch können Aufträge mit kürzest möglicher Reaktionszeit zum Partner weitergeleitet werden, ohne dass zunächst ein Verbindungsaufbau erforderlich ist.
  Ist die Verbindung gestört, wird automatisch versucht, die Verbindung neu aufzubauen. In diesem Zustand kann es zu Verzögerungen kommen, bis eine entsprechende Meldung an die Applikation weitergeleitet wird.
- Sie verwenden die Auftragsarten FETCH oder WRITE (Auswahlmöglichkeit unter "Optionen".
  Nur bei diesen Auftragsarten ist in bestimmten Anwendungsfällen eine unmittelbare Reaktion bei fehlender Verbindung erforderlich.

TCP-Verbindung

TCP - Besondere Verbindungseigenschaften

Bezug

Verbindungsprojektierung In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften".

Verbindungsparameter legen zusätzliche Eigenschaften des lokalen Verbindungsendpunkts fest.

Aktive Verbindung aufbauen

Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von diesem Gerät aus erfolgen soll. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.

- EIN: Verbindung wird aktiv aufgebaut.
- AUS: Verbindung wird vom Partner aufgebaut.

Wenn Sie beim Anlegen der Verbindung als Verbindungspartner "unspezifiziert" gewählt haben, ist die Option standardmäßig abgewählt. Wenn Sie die Option aktivieren, müssen Sie die Adresse des Partners im Register "Adressen" spezifizieren.

Hinweis

Auswirkung auf die Betriebsart

Beachten Sie die Auswirkung auf die Betriebsart.

- S7-300 / S7-400
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" deaktivieren.
- OPC-Server
  Wenn die Betriebsart FETCH oder WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" aktivieren.

FTP-Modus (bei S7-300 / S7-400 - Verbindung unspezifiziert)

Wenn Sie die Option wählen, hat dies folgende Auswirkungen:

- Die TCP-Verbindung wird jetzt als FTP-Verbindung genutzt.
- Parametergruppe "Adressdetails": Die Adressen sind automatisch spezifiziert (Port = 21; Anmerkung: diese Einstellung ist unter "Adressdetails" nicht sichtbar).
- Parametergruppe "Optionen": Die Betriebsart ist fest auf FTP eingestellt.

Die Option "Aktiver Verbindungsaufbau" ist ohne Bedeutung und daher nicht bedienbar.

Voraussetzung: Die Option ist nur bei einer unspezifizierten TCP-Verbindung für bestimmte CPs bei S7-300 / S7-400 wählbar.

TCP OPC-Aufbaudetails

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > Aufbaudetails".

Optionen des Verbindungsaufbaus

Eingaben sind in diesem Bereich nur möglich, wenn Sie unter "Besondere Verbindungseigenschaften" den aktiven Verbindungsaufbau gewählt haben.

Wählen Sie aus den folgenden beiden Alternativen:

- Die Verbindung wird permanent aufrecht erhalten
  Die Verbindung wird mit dem Hochlauf der PC-Station sowie nach einer Verbindungsunterbrechung aufgebaut und aufrecht erhalten.
- Die Verbindung wird bei Bedarf aufgebaut (Zugriff auf Variable)
  Sie können hier zusätzlich angeben, wie lange die Verbindung nach einem Zugriff gehalten werden soll.

Verbindungsaufbau-Überwachung

- Timeout: Fehlerwartezeit während des Verbindungsaufbaus
  Kann innerhalb dieser Wartezeit die Verbindung nicht aufgebaut werden, so wird die Verbindung zurückgesetzt und ein anstehender Auftrag mit einem Fehler quitttiert.
  Wertebereich: 0...99999 ms
  Default: 15000 ms

Hinweis

Die hier einstellbare Verbindungsaufbau-Überwachung steht wie folgt im Zusammenhang mit der in der Baugruppenprojektierung einstellbaren Zeitüberwachung: gültig ist der kleinere Wert.

In der Baugruppenprojektierung ist die Zeitüberwachung unter "Eigenschaften Ethernet Schnittstelle" projektierbar.

TCP Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Adressparameter und Verbindungstypen

Bei TCP werden die Kommunikationspartner durch den lokalen und fernen Endpunkt wie folgt adressiert.

- Lokale Adressen: Lokale IP-Adresse und lokaler Port
- Ferne Adressen: Ferne IP-Adresse und ferner Port

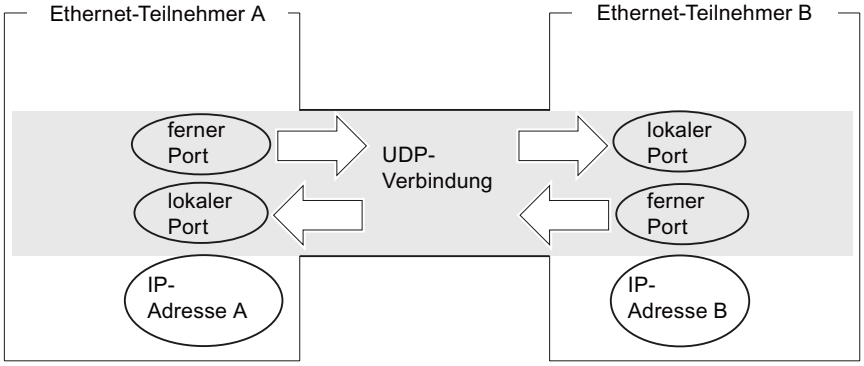

Ports

Die Ports bzw. die Port-Adressen definieren den Zugangspunkt zum Anwenderprogramm innerhalb der Station/CPU. Sie müssen innerhalb der Station/CPU eindeutig sein!

- Spezifizierte Verbindung
  Bei der Projektierung des Ethernet-CP und in der Ethernet-Zielstation müssen die Ports einer TCP-Verbindung kreuzweise übereinstimmen:
  - Ferner Port (im Ethernet-CP) = lokaler Port (in Ziel-Station)
  - Lokaler Port (im Ethernet-CP) = ferner Port (in Ziel-Station)
- Unspezifizierte Verbindung
  Adressparameter werden teilweise nicht festgelegt: Unspezifizierte Verbindungen (Seite 217)

Port-Wertebereich - siehe auch

Port-Einstellungen (Seite 271)

TCP Optionen

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Optionen".

Betriebsart

- SEND/RECV:
  Dies ist der Standardbetrieb mit den FCs AG_SEND/AG_LSEND und AG_RECV/AG_LRECV.
- FETCH/WRITE:
  Wenn Sie eine der folgenden Betriebsarten für die TCP-Verbindung wählen, ist der Zugriff auf die Systemspeicherbereiche in der SIMATIC S7 von folgenden Geräten aus möglich:
  - SIMATIC S5
  - SIMATIC PC-Stationen
  - Fremdgeräten

  Die Verbindung kann dann ausschließlich für diese Betriebsart genutzt werden. Senden oder Empfangen über die FCs AG_SEND/AG_LSEND bzw. AG_RECV/AG_LRECV ist dann nicht möglich.
  - FETCH
    Indem Sie die Betriebsart wählen, ermöglichen Sie den direkten, lesenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
  - WRITE
    Indem Sie die Betriebsart wählen, ermöglichen Sie den direkten, schreibenden Zugriff auf Datenbereiche in der SIMATIC S7. Dieser Zugriff kann beispielsweise von einer SIMATIC S5-Station oder einer Fremdstation aus erfolgen.
- FTP
  Wenn Sie unter "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften" die Option "Für FTP-Protokoll nutzen" gewählt haben, wird hier "FTP" angezeigt; die TCP-Verbindung wird dann als "FTP-Verbindung" genutzt und es besteht keine weitere Auswahlmöglichkeit.
  Voraussetzung: Die Option ist nur bei einer unspezifizierten TCP-Verbindung wählbar.
- SPEED SEND/RECV
  Indem Sie die Betriebsart SPEED SEND/RECV wählen, ermöglichen Sie die Verwendung der Anweisungen AG_SSEND und AG_SRECV.
  Die Anweisungen AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) bieten eine optimierte (da schnellere) Übertragung innerhalb einer S7-400-Station. Informieren Sie sich anhand der Gerätedokumentation des eingesetzten CP, ob diese Betriebsart unterstützt wird.

Voraussetzung für FETCH/WRITE: Art des Verbindungsaufbaus

Wählen Sie abhängig vom Stationstyp für den Verbindungsaufbau folgende Option im Register "Allgemein":

- SIMATIC S7-Station: PASSIVER Verbindungsaufbau
- PC-Station: Aktiver Verbindungsaufbau

Option "S7-Adressierungsmodus" (nur bei FETCH/WRITE)

Bei der Projektierung einer für die Betriebsart FETCH / WRITE können Sie hier den Adressierungsmodus wählen:

- S7-Adressierungsmodus: Byte-Adresse
- S5-Adressierungsmodus: Wort-Adresse

Es ist dadurch möglich, dass Applikationen ohne Anpassung der Adressen auf S5- oder S7-Stationen zugreifen. Dies ist beispielsweise besonders für bestehende S5-Applikationen interessant, die jetzt unverändert für den Zugriff auf S7-Stationen verwendet werden sollen.

Standardmäßig eingestellt ist der Adressierungsmodus für den Zugriff auf SIMATIC S7 (Option angewählt).

Siehe auch

Verbindungsprojektierung für FETCH/WRITE (Seite 283)

Anwendung von FETCH/WRITE (Seite 281)

TCP OPC-Eigenschaften

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > OPC".

Stellen Sie hier verbindungsspezifische Eigenschaften für eine vom OPC-Server genutzte TCP-Verbindung ein.

Folgende Parameter sind hier einstellbar:

Auftragstimeout

Kann innerhalb der hier eingestellten Wartezeit ein ausstehender Auftrag nicht bearbeitet werden, so wird die Verbindung zurückgesetzt und dert Auftrag mit einem Fehler quittiert.

Wertebereich:

- Default-Einstellung: 15000 ms

Größe Sendepuffer

Voreingestellte Größe des Default-Sendepuffers. Zur Laufzeit können weitere zusätzliche Sendepuffer beliebiger Länge (maximal 65535 Byte) verwendet werden.

Wertebereich:

- Default-Einstellung: 256 Byte
- Wertebereich: 65535 Byte maximal

Optimierung Direktzugriff Variablendienste

Indem Sie diese Option nutzen, versucht der OPC-Server direkte Zugriffe auf mehrere Variablen zusammenzufassen, so dass möglichst wenige Aufträge übertragen werden müssen.

Standardmäßig ist die Option eingeschaltet.

Miniprotokoll verwenden

Legen Sie mit dieser Option fest, ob die Daten bei TCP als einzelne Blöcke mit jeweils einer Längenangabe für die Blockgrösse (2 Byte im Auftragspuffer) übertragen werden sollen. Der Empfänger liest dann die Daten entsprechend der Längenangabe aus. Sie müssen diese Eigenschaft bei beiden Kommunikationspartnern identisch projektieren.

Standardmäßig ist die Option ausgeschaltet; der Empfänger liest dann Daten gemäß der eingestellten Empfangspuffergröße aus.

- Größe Empfangspuffer
  Wenn kein Miniprotokoll verwendet wird, können Sie hier die Größe des Empfangspuffers einstellen.
  Wertebereich: 0..65535 Byte
  - Default-Einstellung: 256 Byte

Port-Einstellungen

Port-Einstellungen

Die Ports bzw. die Port-Adressen definieren den Zugangspunkt zum Anwenderprogramm innerhalb der Station/CPU. Sie müssen innerhalb der Station/CPU eindeutig sein! Zu Besonderheiten bei unspezifischen Verbindungen zu redundanten Partnern siehe Abschnitt unten.

Die folgende Tabelle gibt Aufschluss über den Wertebereich:

Port-Adressen		Anwendung/Hinweis
0		fest vergeben; darf nicht verwendet werden!
1..1023		standardmäßig vergeben; sollte nicht verwendet werden (well known ports)
1024..49151	Ports für anwendungsspezifische Protokolle	
	ab 2000...5000	Bereich, in dem vom Projektierwerkzeug jeweils eine freie Port-Adresse gesucht und zugeordnet wird. Sie können die Port-Adresse in diesem Bereich individuell einstellen.
	5001..49151	Die Port-Adressen ab 5000 werden vom System genutzt! Anmerkung: Wenn Sie diese Port-Adressen nutzen möchten, wenden Sie sich zur Prüfung der Verfügbarkeit ggf. an Ihren Systemadministrator.
49152..65535		Dynamisch zugewiesene Ports Es wird empfohlen, diese Ports nicht zu verwenden.

Die folgenden lokalen Port-Nummern sind reserviert, Sie sollten diese nicht andersweitig bei der Verbindungsprojektierung verwenden.

Port-Nummer	Protokoll	Dienst
20, 21	TCP	FTP
25	TCP	SMTP
80	TCP	HTTP
102	TCP	RFC1006
123	UDP (TCP)	NTP *
135	TCP	RPC-DCOM
161	UDP	SNMP_REQUEST
34964	UDP	PN IO
65532	UDP	NTP
65533	UDP	NTP
65534	UDP	NTP
65535	UDP	NTP

* Port 123 ist beim CP 443-1 und CP 443-1 Advanced ab Firmware-Version V3 für NTP reserviert..

Portnummmern bei unspezifische Verbindungen zu redundanten Partnern

Wenn Sie unspezifische Verbindungen zu einem redundanten Partner anlegen, bspw. von einer S7-400 mit CP zu einem redundanten IEC-Master, dann können Sie im CP 443-1 (Advanced) für die zwei unspezifischen TCP-Verbindungen dieselbe lokale Portnummer verwenden.

UDP-Verbindung

UDP Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Adressparameter und Verbindungstypen

Bei UDP werden die Kommunikationspartner durch den lokalen und fernen Endpunkt wie folgt adressiert.

- Lokale Adressen:
  Lokale IP-Adresse und lokaler Port
- Ferne Adressen:
  Ferne IP-Adresse und ferner Port

Hinweis

Der Begriff "Verbindung" wird hier auch bei UDP verwendet, weil bei der Projektierung - wie z. B. auch bei TCP - die Kommunikationspartner einander zugeordnet und somit logisch "verbunden" werden. Tatsächlich erfolgt bei UDP im Betrieb der Stationen kein expliziter Verbindungsaufbau zwischen den Kommunikationspartnern.

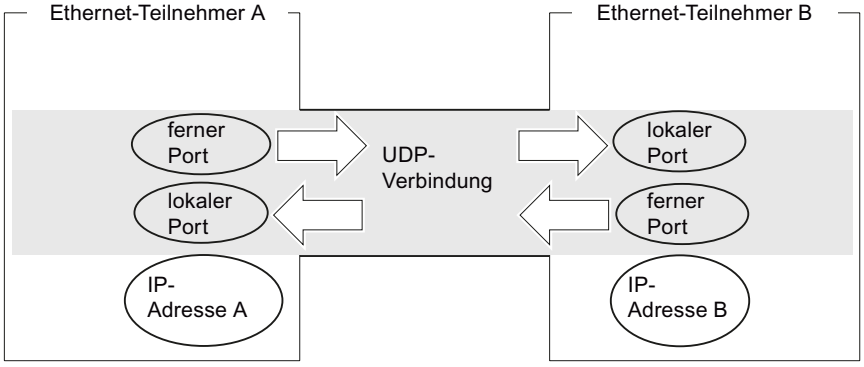

Bild 1-6 Senden und Empfangen über eine UDP-Verbindung

Abhängig vom gewünschten Verbindungstyp werden die fernen Adressparameter bei der Projektierung spezifiziert oder offen gelassen.

- spezifizierte UDP-Verbindung
  Sie haben beim Neuanlegen der Verbindung eine Zielstation angegeben.
- Als Broadcast spezifizierte UDP-Verbindung
  Sie haben beim Neuanlegen der Verbindung als Partner "Broadcast" angegeben (Verbindung muss dialoggeführt oder grafisch zunächst als unspezifizierte Verbindung angelegt werden).
- Als Multicast spezifizierte UDP-Verbindung
  Sie haben beim Neuanlegen der Verbindung als Partner "Multicast" angegeben (Verbindung muss dialoggeführt oder grafisch zunächst als unspezifizierte Verbindung angelegt werden).
- unspezifizierte UDP-Verbindung
  Sie haben beim Neuanlegen der Verbindung den Verbindungstyp "unspezifiziert" gewählt.

Unspezifizierte UDP-Verbindung

Die unspezifizierte UDP-Verbindung kann auf zwei Arten genutzt werden:

- Freie UDP-Verbindung
  Um eine Freie UDP-Verbindung zu projektieren, wählen Sie die Option "Adressvergabe am Baustein". Die Eingabefelder für die ferne IP-Adresse und den fernen Port können dann nicht mehr eingegeben werden, da die Zieladressen jetzt durch das Anwenderprogramm festgelegt werden.
- Verbindung zu einer "Fremdstation" in einem anderen Projekt
  Sie können die ferne IP-Adresse und den Port für eine beliebige Zielstation angeben. Die Zielstation kann innerhalb oder außerhalb des aktuellen Projekts liegen.

Hinweis

Partner-Adressen müssen spezifiziert sein

Da bei UDP kein Verbindungsaufbau stattfindet (Datagrammdienst), ist die Kommunikation über die projektierte UDP-Verbindung nur dann möglich, wenn auch die Partner-Adressen (IP-Adresse und Port) spezifiziert sind.

Siehe auch

Unspezifizierte Verbindungen (Seite 217)

Verbindung zu allen Broadcast-Teilnehmern

Indem Sie als Verbindungspartner "alle Broadcast-Teilnehmer" wählen, legen Sie fest, dass UDP-Telegramme an alle erreichbaren Broadcast-Teilnehmer gesendet werden.

In der Parametergruppe "Adressdetails" wird unter der IP-Adresse (IP) für den Partner eine im Netz gültige Broadcast-Adresse vorgeschlagen.

Unter PORT müssen Sie eine für alle zu erreichenden Partner zutreffende PORT-Adresse eintragen.

Porteinstellungen - siehe auch

Port-Einstellungen (Seite 280)

UDP mit Broadcast und Multicast

Anwendung

Bei der Auswahl des Verbindungspartners haben Sie bei UDP-Verbindungen zusätzlich die beiden Optionen:

- Verbindung zu allen Broadcast-Teilnehmern
  Indem Sie als Partner "alle Broadcast-Teilnehmer" wählen, legen Sie fest, dass UDP-Telegramme an alle erreichbaren Broadcast-Teilnehmer gesendet werden.

  Hinweis

  Bei Broadcast ist über S7-CPs nur Senden möglich; ein S7-CP ermöglicht auf Broadcast-Verbindungen keinen Empfang von Nutzdaten (siehe unten)!

- Verbindung zu allen Multicast-Teilnehmern
  Indem Sie als Partner "alle Multicast-Teilnehmer" wählen, legen Sie fest, dass UDP-Telegramme an alle Teilnehmer einer Multicast-Gruppe gesendet werden und Multicast-Telegramme empfangen werden können.
  Multicast und Broadcast sind spezielle Verbindungsoptionen, welche von Industrial Ethernet-CPs bei UDP-Verbindungen unterstützt werden und projektierbar sind.

Die Telegramme werden ohne Quittung versendet, da das Protokoll UDP keine Quittierungen vorsieht. Dies ist so vorgesehen, um "Datenfluten" durch Quittungen zu verhindern. Werden beispielsweise Telegramme an 100 Partner versendet, würden 100 Quittungen (pro Partner 1) gleichzeitig ankommen. Solche Datenfluten könnten von der Senderbaugruppe nicht ausgewertet werden.

Siehe auch

UDP- / FDL-Verbindungen mit Multicast / Broadcast anlegen (Seite 243)

Wann Multicast anstelle von Broadcast einsetzen?

Um ein gleichzeitiges Versenden eines Telegramms an eine Vielzahl von Partnern zu ermöglichen, wurde die Verbindungsoption "Multicast" für UDP-Verbindungen eingeführt.

Im Gegensatz zur Verbindungsoption Broadcast ist auf diesem Verbindungstyp auch der Empfang von Telegrammen möglich, die an mehrere Teilnehmer im Multicast-Kreis gesendet werden.

Durch die Ausrichtung auf eine bestimmte Empfängergruppe - den Multicast-Kreis - kann eine Belastung von nicht angesprochenen Empfängern verhindert werden. Daher stellt Multicast in jedem Fall gegenüber Broadcast die bessere Lösung dar, wenn Telegramme an Gruppen von Partnerstationen verschickt werden sollen.

Warum ermöglicht ein S7-CP auf Broadcast-Verbindungen keinen Empfang von Nutzdaten?

S7-CPs empfangen keine Broadcast-Telegramme zum Nutzdatentransfer. Lediglich verwertbare Telegramme wie z. B. ARP-Request werden über den LAN-Controller hinaus weitergereicht und ausgewertet. Es ist jedoch möglich mit S7-CPs Broadcast-Telegramme in das Netz zu versenden.

Begründung:

Oft besteht der Wunsch, von einem Gerät aus Telegramme an eine Vielzahl von Partnergeräten zu verschicken. Wichtig ist hierbei, dass die Telegramme gleichzeitig verschickt werden und auch nahezu gleichzeitig bei den Partnergeräten ankommen. Es wird daher immer ein Versenden und Empfangen von Broadcast-Telegrammen gefordert. Bei einer Broadcast-Nachricht wird das Telegramm auch wirklich von allen Teilnehmern am Netz aufgenommen.

Ein typischer Anwendungsfall ist, dass Broadcast-Telegramme zum Suchen einer MAC-Adresse zu einer IP-Adresse benötigt (ARP-Request) werden.

Daher muss eine Kommunikationsbaugruppe Broadcast-Telegramme generell aufnehmen und softwaremäßig auswerten. Ein gravierender Nachteil besteht hierbei darin, dass bei zu vielen Broadcast-Telegramme auf dem Netz die Performance deutlich sinkt. Dies liegt daran, dass die einzelnen Baugruppen alle Broadcast-Telegramme bearbeiten müssen, um festzustellen, ob diese für sie bestimmt waren.

Um die genannten Nachteile zu vermeiden, verhalten sich S7-CPs gegenüber Broadcast wie folgt:

- Nach dem Empfang werden bei allen Ethernet-CPs die Broadcast-Telegramme hochprior herausgefiltert. Das heißt, dass alle Telegramme, bei denen es sich nicht um verwertbare Telegramme handelt, direkt verworfen werden. Lediglich verwertbare Telegramme wie z. B. ARP-Request werden über den LAN-Controller hinaus weitergereicht und ausgewertet. Dadurch wird ein möglicher negativer Einfluss von Broadcast-Telegrammen auf die anderen Verbindungen verhindert.

Parametergruppe "Adressdetails" - Verbindung zu allen Multicast-Teilnehmern

Indem Sie als Verbindungspartner "alle Multicast-Teilnehmer" wählen, legen Sie folgendes fest:

- Gesendete UDP-Telegramme werden allen erreichbaren Multicast-Teilnehmern der Multicast-Gruppe zugestellt;
- Das lokale Gerät für Multicast-Telegramme ist in der angegebenen Multicast-Gruppe empfangsbereit.

Der Multicast-Kreis wird über die IP-Adresse und die Port-Adressen festgelegt.

Unter IP-Adresse (IP) für den Partner wird eine im Netz gültige IP-Adresse für Multicast-Kreise vorgeschlagen. Partner ist bei Multicast jeweils eine Gruppe von Empfängern (Multicast-Gruppe).

Für alle Partner, die Sie erreichen wollen, tragen Sie unter PORT eine PORT-Adresse ein. Sie muss für alle diese Partner zutreffen.

Es ist prinzipiell möglich, unter einer IP-Adresse mehrere Multicast-Kreise anzusprechen. Hierzu können Sie mehrere UDP-Verbindungen mit derselben IP-Adresse aber unterschiedlichen PORT-Adressen anlegen.

Hinweis

Sie sollten innerhalb eines Multicast-Kreises für den lokalen Port und für den Partnerport identische Port-Adressen vergeben. Nur so können innerhalb des einen Multicast-Kreises Telegramme vom CP gesendet und auch empfangen werden!

Beachten Sie das folgende Beispiel für drei Teilnehmer im Multicast-Kreis:

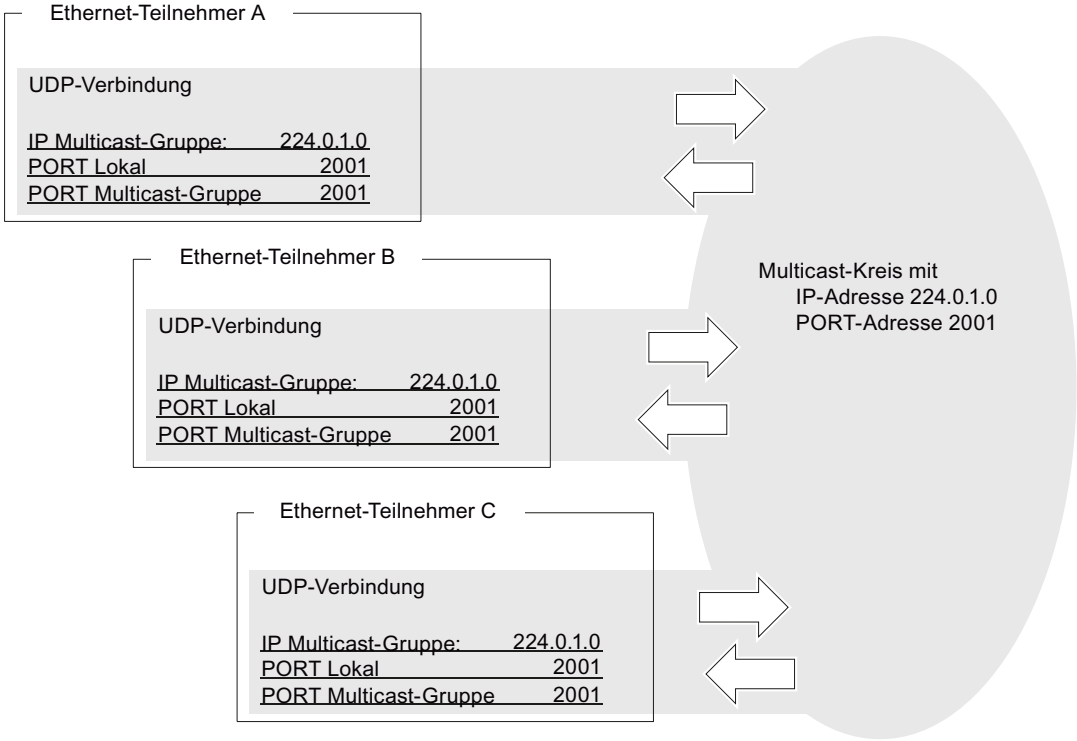

Bild 1-7 Senden und Empfangen in einem Multicast-Kreis über identische Port-Adressen

IP-Adressen für IP-Multicast

- Wertebereich
  Für IP-Multicast können die IP-Adressen von 224.0.0.0 bis 239.255.255.255 verwendet werden.
  Da die IP-Adressen bis 224.0.0.255 für besondere Zwecke reserviert sind, wird empfohlen, für IP-Multicast die IP-Adressen ab 224.0.1.0 (Default-Einstellung) zu verwenden.
- Identifikation des Multicast-Kreises
  Die Identifikation eines Multicast-Kreises erfolgt nicht über die gesamte IP-Adresse; es werden vielmehr das erste Adressbyte und das höchstwertige Bit des zweiten Adressbyte ignoriert. Dies ist eine wichtige Besonderheit, denn dadurch können scheinbar unterschiedliche IP-Adressen den selben Multicast-Kreis adressieren.
  Beispiel:
  Die folgenden IP-Adressen adressieren jeweils den selben Multicast-Kreis.

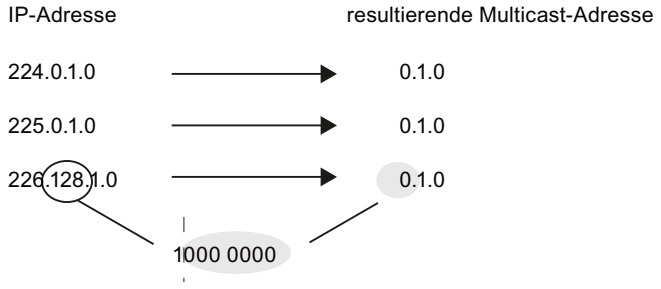

Freie UDP-Verbindung

Programmgesteuerte Adressierung

Eine Freie UDP-Verbindung ermöglicht die programmgesteuerte Adressierung des Kommunikationspartners. Die Kommunikation zwischen zwei Teilnehmern an Industrial Ethernet hat folgende Eigenschaften:

- Der Datentransfer ist bidirektional, d. h. auf der UDP-Verbindung kann gleichzeitig gesendet und empfangen werden.
- Der lokale Teilnehmer ist per Projektierung festgelegt. Der ferne Teilnehmer wird vom Anwenderprogramm beim AG_SEND-Aufruf im Auftragsheader des Auftragspuffers eingetragen. Damit kann jeder beliebige Teilnehmer am Ethernet/LAN/WAN erreicht werden.
- Aus dem Auftragsheader des AG_RECV kann die IP-Adresse und der Port des Senders gelesen werden.

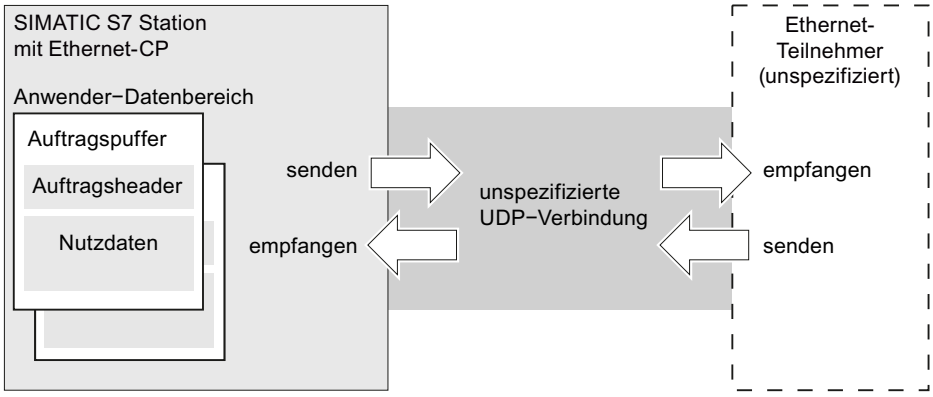

Bild 1-8 Senden und Empfangen über eine unspezifizierte UDP-Verbindung - Adressierung per Programm

Datenvolumen und Mengengerüst

Wieviele UDP-Verbindungen der jeweilige Ethernet-CP unterstützt, entnehmen Sie bitte dem Ethernet-CP beiliegenden Gerätehandbuch. Durch Hinzunahme weiterer CPs kann die Anzahl der Verbindungen pro Station erhöht werden.

Pro Auftragspuffer können bis zu 2042 Byte Nutzdaten übertragen werden. Der Auftragsheader belegt zusätzlich 6 Byte.

UDP Optionen

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Optionen".

Betriebsart

- **SEND/RECV:**
  Dies ist der Standardbetrieb mit den FCs AG_SEND/AG_LSEND und AG_RECV/AG_LRECV.
- **SPEED SEND/RECV**
  Indem Sie die Betriebsart SPEED SEND/RECV wählen, ermöglichen Sie die Verwendung der Anweisungen AG_SSEND und AG_SRECV.
  Die Anweisungen AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) bieten eine optimierte (da schnellere) Übertragung innerhalb einer S7-400-Station. Informieren Sie sich anhand der Gerätedokumentation des eingesetzten CP, ob diese Betriebsart unterstützt wird.

Port-Einstellungen

Port-Einstellungen

Die Ports bzw. die Port-Adressen definieren den Zugangspunkt zum Anwenderprogramm innerhalb der Station/CPU. Sie müssen innerhalb der Station/CPU eindeutig sein! Zu Besonderheiten bei unspezifischen Verbindungen zu redundanten Partnern siehe Abschnitt unten.

Die folgende Tabelle gibt Aufschluss über den Wertebereich:

Port-Adressen		Anwendung/Hinweis
0		fest vergeben; darf nicht verwendet werden!
1..1023		standardmäßig vergeben; sollte nicht verwendet werden (well known ports)
1024..49151	Ports für anwendungsspezifische Protokolle	
	ab 2000...5000	Bereich, in dem vom Projektierwerkzeug jeweils eine freie Port-Adresse gesucht und zugeordnet wird. Sie können die Port-Adresse in diesem Bereich individuell einstellen.
	5001..49151	Die Port-Adressen ab 5000 werden vom System genutzt! Anmerkung: Wenn Sie diese Port-Adressen nutzen möchten, wenden Sie sich zur Prüfung der Verfügbarkeit ggf. an Ihren Systemadministrator.
49152..65535		Dynamisch zugewiesene Ports Es wird empfohlen, diese Ports nicht zu verwenden.

Die folgenden lokalen Port-Nummern sind reserviert, Sie sollten diese nicht andersweitig bei der Verbindungsprojektierung verwenden.

Port-Nummer	Protokoll	Dienst
20, 21	TCP	FTP
25	TCP	SMTP
80	TCP	HTTP
102	TCP	RFC1006
123	UDP (TCP)	NTP *

Port-Nummer	Protokoll	Dienst
135	TCP	RPC-DCOM
161	UDP	SNMP_REQUEST
34964	UDP	PN IO
65532	UDP	NTP
65533	UDP	NTP
65534	UDP	NTP
65535	UDP	NTP

* Port 123 ist beim CP 443-1 und CP 443-1 Advanced ab Firmware-Version V3 für NTP reserviert..

Portnummmern bei unspezifische Verbindungen zu redundanten Partnern

Wenn Sie unspezifische Verbindungen zu einem redundanten Partner anlegen, bspw. von einer S7-400 mit CP zu einem redundanten IEC-Master, dann können Sie im CP 443-1 (Advanced) für die zwei unspezifischen TCP-Verbindungen dieselbe lokale Portnummer verwenden.

Betriebsart FETCH/WRITE

Anwendung von FETCH/WRITE

FETCH/WRITE

Mit den FETCH/WRITE-Diensten greifen Sie auf Systemspeicherbereiche der S7-CPU zu.

- FETCH: Daten direkt lesen
- WRITE: Daten direkt schreiben

Der Zugriff ist von folgenden Geräten aus möglich:

- SIMATIC S5
- SIMATIC PC-Stationen
- Fremdgeräte

Verbindungstypen

Die FETCH/WRITE-Dienste können bei SIMATIC S7 auf folgenden Verbindungstypen projektiert und genutzt werden:

- ISO-Transportverbindungen
- ISO-on-TCP-Verbindungen
- TCP-Verbindungen

Systemspeicher

Auf folgende Operandenbereiche im Systemspeicher der SIMATIC S7 können Sie über FETCH oder WRITE zugreifen:

- Datenbausteine (DB)
  (beachten Sie für den DB-Zugriff folgende Beschränkung: die höchste DB-Nr ist 255)
- Merker (M)
- Prozessabbild der Eingänge (PAE)
- Prozessabbild der Ausgänge (PAA)
- Peripheriebereich Eingänge (PEW, PED, PEB)
- Peripheriebereich Ausgänge (PAW, PAD, PAB)
- Zähler (Z)
- Zeiten (T)

Kopplung zu Fremdsystemen

FETCH und WRITE können prinzipiell von beliebigen Fremdgeräten aus für den Zugriff auf S7-Systemspeicherbereiche genutzt werden.

Damit Sie diesen Zugriff z. B. auch für PC-Anwendungen implementieren können, müssen Sie den PDU-Aufbau für die Aufträge kennen. Beachten Sie hierzu die Gerätedokumentation.

Meldungen im Diagnosepuffer

Aufgrund von FETCH/WRITE-Zugriffen kann es zu negativen Quittierungen seitens der S7-CPU kommen. Dies führt dann zu entsprechenden verbindungsorientierten Einträgen im Diagnosepuffer.

Tabelle 1-4 Meldungscodierung im Diagnosepuffer bei FETCH/WRITE

Codierung	Bedeutung
01_H	Hardware-Fehler
03_H	Objektzugriff ist nicht erlaubt.
05_H	Ungültige Adresse (Syntax-ID, Bereich, Typ, Bit-Nummer)
06_H	Datentyp wird nicht unterstützt.
07_H	Datentyp ist nicht konsistent.
$0A_H$	Das Objekt existiert nicht oder das Bereichsende ist überschritten.
FF_H	Interner Protokollfehler

Weitere Informationen

Der PDU-Aufbau für den Zugriff von Fremdsystemen aus ist in der Gerätedokumentation für S7-CPs beschrieben.

Verbindungsprojektierung für FETCH/WRITE

Betriebsart abhängig vom Gerätetyp projektieren

Je nach Stationstyp sind folgende Betriebsarten für den Verbindungsendpunkt projektierbar:

- SIMATIC S7-Station: FETCH PASSIV / WRITE PASSIV
  Indem Sie eine der Betriebsarten FETCH PASSIV bzw. WRITE PASSIV für die Verbindung wählen, können direkte Zugriffe auf die Systemspeicherbereiche in der SIMATIC S7 von einer SIMATIC S5-Station oder einer Fremdstation (unspezifizierte Verbindung) aus erfolgen.
  Die Verbindung kann dann ausschließlich für diese Betriebsart genutzt werden. Senden oder Empfangen über die FCs AG_SEND/AG_LSEND bzw. AG_RECV/AG_LRECV ist dann nicht möglich.
  Der Verbindungsaufbau erfolgt passiv, d. h. nur die Partnerstation (SIMATIC S5-Station, eine PC-Station oder eine Fremdstation) kann die Verbindung aufbauen. Die entsprechende Einstellung nehmen Sie unter "Eigenschaften > Optionen" vor. Nachdem Sie die Betriebsart gewählt haben, ist der Typ des Verbindungsaufbaus nicht mehr änderbar.
- SIMATIC PC-Station: FETCH AKTIV / WRITE AKTIV
  Indem Sie eine der Betriebsarten FETCH AKTIV bzw. WRITE AKTIV wählen, können direkte Zugriffe auf die Systemspeicherbereiche in der SIMATIC S7 oder SIMATIC S5-Station von der PC-Station aus erfolgen.
  Der Verbindungsaufbau erfolgt aktiv, d. h. die Partnerstation muss den Verbindungsaufbau erwarten (passiver Verbindungsaufbau beim Partner). Die entsprechende Einstellung nehmen Sie unter "Eigenschaften > Optionen" vor. Nachdem Sie die Betriebsart gewählt haben, ist der Typ des Verbindungsaufbaus nicht mehr änderbar.

Option "S7-Adressierungsmodus"

Bei der Projektierung für die Betriebsart FETCH AKTIV/WRITE AKTIV können Sie den Adressierungsmodus wählen. Damit wird festgelegt, wie die Adressangabe im FETCH/WRITE-Aufruf in der SIMATIC S7-Station beim Zugriff auf DBs interpretiert werden soll:

- S7-Adressierungsmodus: Byte-Adresse
- S5-Adressierungsmodus: Wort-Adresse

Es ist dadurch möglich, dass Applikationen ohne Anpassung der Adressen auf S5- oder S7-Stationen zugreifen. Dies ist beispielsweise besonders für bestehende S5-Applikationen interessant, die jetzt unverändert für den Zugriff auf S7-Stationen verwendet werden sollen.

Standardmäßig eingestellt ist der Adressierungsmodus für den Zugriff auf SIMATIC S7 (Option angewählt).

FDL-Verbindung

FDL-Verbindungen - Übersicht

Anwendung

Die Datenübertragung über eine projektierte FDL-Verbindung ist geeignet für die Übertragung zusammenhängender Datenblöcke zwischen zwei oder mehreren PROFIBUS-Teilnehmern.

Zu unterscheiden sind

- spezifizierte FDL-Verbindung
  Die Kommunikationsteilnehmer sind durch die Verbindungsprojektierung eindeutig festgelegt.
  Der Verbindungspartner kann innerhalb oder außerhalb des STEP7-Projektes liegen.
- unspezifizierte FDL-Verbindung (freier Layer 2 Zugang)
  Die Adresse des Verbindungspartners bleibt bei der Projektierung offen. Die Kommunikationsteilnehmer sind durch Adressangaben im Kommunikationsauftrag des Anwenderprogrammes bestimmt. Dadurch können bis zu 126 Teilnehmer über eine projektierte unspezifizierte FDL-Verbindung erreicht werden, soweit diese Teilnehmer FDL-Verbindungen unterstützen.
  Der Verbindungspartner kann innerhalb oder außerhalb des Projektes liegen.
- FDL-Verbindung mit Broadcast
  Es werden alle für Broadcast empfangsbereiten Teilnehmer am PROFIBUS erreicht.
- FDL-Verbindung mit Multicast
  Es werden alle zum Multicast-Kreis gehörenden Teilnehmer am PROFIBUS erreicht.

Hinweis

Voraussetzung für die Projektierung

Wenn FDL-Verbindungen genutzt werden sollen, darf die CP-Betriebsart des PROFIBUS-CP **nicht** auf **DP-Slave passiv** eingestellt werden!

Hinweis

Priorität der Telegramme

Beachten Sie, dass die PROFIBUS-CPs für SIMATIC S7 die Telegramme in der Priorität "LOW" senden.

Partnerstationen (SIMATIC S5, S7 oder Fremdstationen) müssen ebenfalls die Priorität "LOW" verwenden, da sonst kein Verbindungsaufbau zustande kommt.

FDL Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Adressparameter bei FDL-Verbindungen

Eine FDL-Verbindung wird spezifiziert durch den lokalen und fernen Verbindungsendpunkt. Hierzu gehören:

- PROFIBUS-Adresse des Teilnehmers, der erreicht werden soll.
- Lokaler LSAP (Link ServiceAccessPoint):
  Der lokale LSAP steuert die Empfangsbereitschaft des PROFIBUS-CP. Für den LSAP werden im PROFIBUS-CP die Empfangsressourcen für den Datenempfang auf der FDL-Verbindung bereitgestellt.
- Ferner LSAP (Link ServiceAccessPoint):
  Der ferne LSAP steuert den Sendebetrieb im PROFIBUS-CP. Über den LSAP sendet der PROFIBUS-CP zum Teilnehmer auf der FDL-Verbindung. Der Zielteilnehmer muss für diesen SAP empfangsbereit sein.

spezifizierte FDL-Verbindung

Bei einer spezifizierten FDL-Verbindung werden die relevanten lokalen und fernen Adressinformationen als Vorschlagswerte angezeigt. Bei Bedarf haben Sie die Möglichkeit die LSAP-Adressen individuell einzustellen.

unspezifizierte FDL-Verbindung

Die unspezifizierte FDL-Verbindung kann auf zwei Arten genutzt werden:

- Verbindung zu einer "Fremdstation" in einem anderen Projekt
  Sie können die ferne PROFIBUS-Adresse und den LSAP für eine beliebige Zielstation angeben. Die Zielstation kann innerhalb oder außerhalb des aktuellen STEP 7-Projekts liegen.
  Solange die ferne Adresse unspezifiziert ist, ist keine Kommunikation über die FDL-Verbindung möglich.
- Freier Layer-2-Zugang
  Um einen freien Layer-2-Zugang zu projektieren, selektieren Sie das entsprechende Optionskästchen. Die Eingabefelder für die ferne PROFIBUS-Adresse und den fernen LSAP können dann nicht mehr eingegeben werden, da die Zieladressen jetzt durch das Anwenderprogramm festgelegt werden.

FDL-Verbindung mit Broadcast

Bei der FDL-Verbindung mit Broadcast sind die fernen Adressparameter fest eingestellt. Alle Broadcast-Teilnehmer sind über die PROFIBUS-Adresse 127 zu erreichen. Empfangsdaten werden über den LSAP 63 von allen Broadcast-Teilnehmern entgegengenommen.

Die lokalen Adressparameter werden beim Senden in den Auftragsheader der Nachricht eingetragen und dem Empfänger zugestellt. Das Anwenderprogramm beim fernen Partner kann damit den Sender des Broadcast-Telegrammes ermitteln.

Hinweis

CP empfängt nur auf einer Broadcast Verbindung

Wenn Sie eine FDL-Verbindung mit Broadcast nutzen, können Sie auf dem betreffenden CP keine weitere Broadcast-Verbindung nutzen.

VORSICHT

Funktion "Zyklisches Verteilen der Busparameter einschalten"

Wenn Sie für das Subnetz unter "Allgemein > Busparameter" diese Option aktiviert haben, werden die Busparameter im Betrieb zyklisch als Broadcast-Telegramme gesendet. Um Konflikte bei der Telegrammhantierung im Anwenderprogramm, das Broadcast-Telegramme empfängt, zu vermeiden, müssen Sie wie folgt vorgehen:

- Sie ignorieren alle Telegramme, die mit einem LSAP >56 gesendet werden;
  oder
- Sie deaktivieren die genannte Option unter "Allgemein > Busparameter".

FDL-Verbindung mit Multicast

Bei der FDL-Verbindung mit Multicast sind die fernen Adressparameter fest eingestellt. Alle Multicast-Teilnehmer sind über die PROFIBUS-Adresse 127 zu erreichen. Daten senden und empfangen wird von allen Teilnehmern im Multicast-Kreis über den selben LSAP (Bereich von 1 bis 56) abgewickelt. Der Wert des LSAP ist daher nur lokal wählbar und wird in den fernen LSAP automatisch übernommen.

Die lokalen Adressparameter werden beim Senden in den Auftragsheader der Nachricht eingetragen und dem Empfänger zugestellt. Das Anwenderprogramm beim fernen Partner kann damit den Sender des Multicast-Telegrammes ermitteln.

FDL Optionen

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Optionen".

Betriebsart

- SEND/RECV:
  Dies ist der Standardbetrieb mit den FCs AG_SEND/AG_LSEND und AG_RECV/AG_LRECV. Bei FDL-Verbindungen ist diese Betriebsart voreingestellt und nicht änderbar.

FDL OPC-Eigenschaften

Bezug

Verbindungsprojektierung bei OPC in der Parametergruppe "Eigenschaften > Allgemein > OPC".

Stellen Sie hier verbindungsspezifische Eigenschaften für eine vom OPC-Server genutzte FDL-Verbindung ein.

Folgende Parameter sind hier einstellbar:

Fehlerwartezeit

Kann innerhalb dieser Wartezeit ein ausstehender Auftrag nicht bearbeitet werden, so wird die Verbindung zurückgesetzt und der Auftrag mit einem Fehler quittiert.

Max. Anzahl Sendewiederholungen

Geben Sie hier an, wieviel Sendewiederholungen für einen Auftrag erfolgen sollen, wenn der Verbindungspartner beispielsweise wegen temporärem Ressourcenmangel nicht erreichbar ist.

Größe Sendepuffer

Hier wird die Größe des Default-Sendepuffers voreingestellt.

Die getroffene Einstellung gilt sowohl für quittiertes als auch für nicht quittiertes Senden.

Zur Laufzeit können zusätzliche Sendepuffer beliebiger Länge (maximal jeweils 244 beziehungsweise 246 Byte) verwendet werden.

Default-Einstellung:

244 Byte beziehungsweise 246 Byte (bei Default SAPs, wenn für den lokalen und fernen SAP jeweils der Default-SAP 255 eingestellt wurde)

Max. Anzahl paralleler Sendeaufträge

Maximale Anzahl paralleler Sendeaufträge auf der hier projektierten Verbindung.

Anzahl Ressourcen für empfangene Aufträge

Geben Sie an, für wie viele Empfangsaufträge der OPC-Server auf der hier projektierten FDL-Verbindung gleichzeitig empfangsbereit sein soll.

Wählen Sie einen Wert, der dem zu erwartenden Kommunikationsaufkommen entspricht; das heißt

- Bei insgesamt geringer Anzahl Verbindungen: den Wert eher höher wählen, um eine schnelle Reaktionszeit zu sichern.
- Bei insgesamt hoher Anzahl Verbindungen: den Wert eher niedriger wählen, um Ressourcenkonflikte zu vermeiden.
  - Default-Einstellung: 3

Mit hoher Priorität senden

Geben Sie an, ob die FDL-Telegramme auf der hier projektierten Verbindung über das LAN bevorzugt (Option eingeschaltet) gesendet werden sollen.

FDL Unspezifizierte FDL-Verbindung (Freier Layer 2 Zugang)

Eigenschaften

Eine unspezifizierte FDL-Verbindung mit offenem Layer 2 Zugang ermöglicht die programmgesteuerte Adressierung des Kommunikationspartners und die Kommunikation zwischen zwei Teilnehmern am PROFIBUS mit folgenden Eigenschaften:

- Der Datentransfer ist bidirektional, d. h. auf der FDL-Verbindung kann gleichzeitig gesendet und empfangen werden.
- Der lokale Teilnehmer ist per Projektierung festgelegt. Der ferne Teilnehmer wird vom Anwenderprogramm beim Aufruf der Anweisung AG_SEND im Auftragsheader des Auftragspuffers eingetragen. Damit kann jeder Teilnehmer am PROFIBUS (PROFIBUS-Adressen von 0 bis 126) erreicht werden.
- Aus dem Auftragsheader des AG_RECV kann die PB-Adresse, der LSAP und der Dienst des Senders gelesen werden.

Die folgende Darstellung verdeutlicht die programmgesteuerte Adressierung im Auftragspuffer:

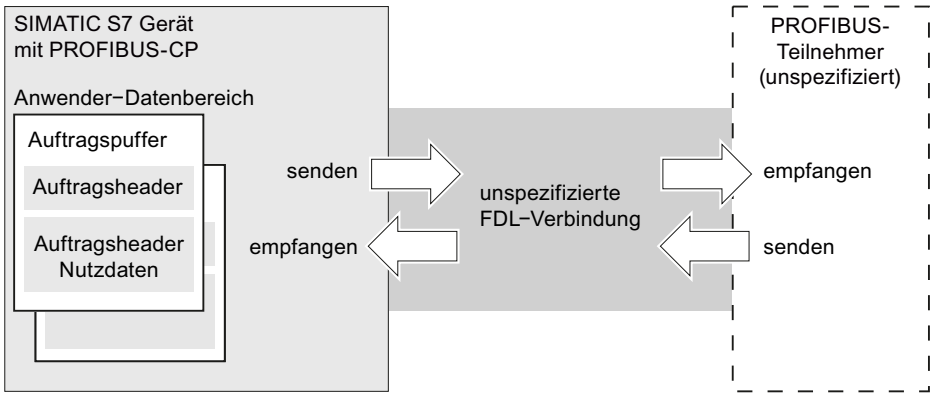

Datenvolumen und Mengengerüst

Pro Auftragspuffer können bis zu 236 Byte Nutzdaten übertragen werden. Der Auftragsheader belegt zusätzlich 4 Byte.

FDL-Verbindung mit Broadcast

Eigenschaften

Eine Broadcast-Verbindung ermöglicht das Senden einer Nachricht an mehrere Empfänger mit **einem** Auftrag. Entsprechend können Nachrichten auf der selben Broadcast-Verbindung entgegengenommen werden, die gleichzeitig von anderen Teilnehmern am PROFIBUS empfangen werden.

Die Eigenschaften lassen sich wie folgt zusammenfassen:

- Der Datentransfer ist bidirektional, d. h. auf der Broadcast-Verbindung kann gleichzeitig gesendet und empfangen werden.
- Das Senden und Empfangen erfolgt über den FDL-Dienst SDN (Send Data with No Acknowledge).
- Der AG_SEND-Aufruf braucht beim Senden einen Auftragspuffer. Der Auftragsheader dieses Puffers muss reserviert werden.
- Aus dem Auftragsheader des AG_RECV kann die PB-Adresse, der LSAP und der Dienst des Broadcast-Senders gelesen werden.
- Für das Senden wird der LSAP-Bereich von 1 bis 56 verwendet. Für das Empfangen wird der LSAP 63 für alle Broadcast-Teilnehmer reserviert.

Die folgende Darstellung verdeutlicht das Senden und Empfangen über eine FDL-Verbindung mit Broadcast- Adressierung per Programm:

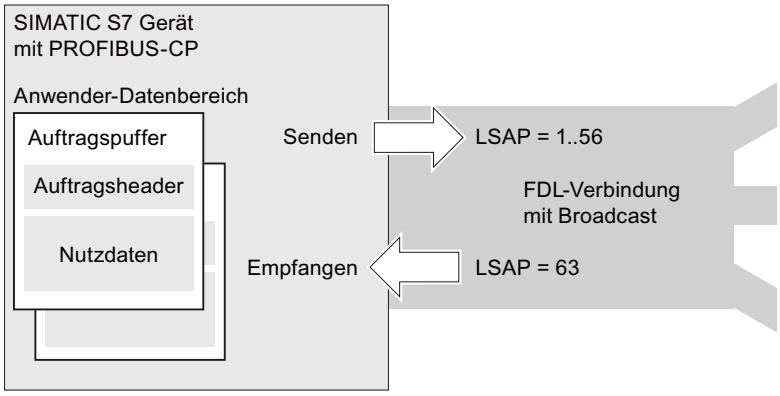

FDL-Verbindung mit Broadcast projektieren

Wählen Sie beim Anlegen der FDL-Verbindung als Verbindungspartner/Station "Alle Broadcast-Teilnehmer".

Siehe auch

UDP- / FDL-Verbindungen mit Multicast / Broadcast anlegen (Seite 243)

Datenvolumen und Mengengerüst

Der PROFIBUS-CP unterstützt jeweils **eine** Broadcast-Verbindung.

Pro Auftragspuffer können bis zu 236 Byte Nutzdaten übertragen werden. Der Auftragsheader belegt zusätzlich 4 Byte.

Hinweis

CP empfängt nur auf einer Broadcast Verbindung

Wenn Sie eine FDL-Verbindung mit Broadcast nutzen, können Sie auf dem betreffenden CP auf keiner weiteren Broadcast-Verbindung Nachrichten empfangen, auch auf keiner FMS-Verbindung mit Broadcast.
Begründung:
Der Empfangs-LSAP für Broadcast (63) ist mit einer Broadcast-Verbindung belegt.

FDL-Verbindung mit Multicast

Eigenschaften

Eine FDL-Verbindung mit Multicast ermöglicht das Senden einer Nachricht an mehrere Empfänger eines Multicast-Kreises mit einem Auftrag.

Die Eigenschaften lassen sich wie folgt zusammenfassen:

- Der Datentransfer ist bidirektional, d. h. auf der FDL-Verbindung mit Multicast kann gleichzeitig gesendet und empfangen werden.
- Das Senden und Empfangen erfolgt über den FDL-Dienst SDN (Send Data with No Acknowledge).
- Gesendet wird über einen für den Multicast-Kreis einheitlichen LSAP (Bereich von 1 bis 56).
- Der AG_SEND-Aufruf braucht beim Senden einen Auftragspuffer. Der Auftragsheader dieses Puffers muss reserviert werden.
- Aus dem Auftragsheader des AG_RECV kann die PB-Adresse, der LSAP und der Dienst des Multicast-Senders gelesen werden.

Die folgende Darstellung verdeutlicht das Senden und Empfangen über eine FDL-Verbindung mit Multicast- Adressierung per Programm:

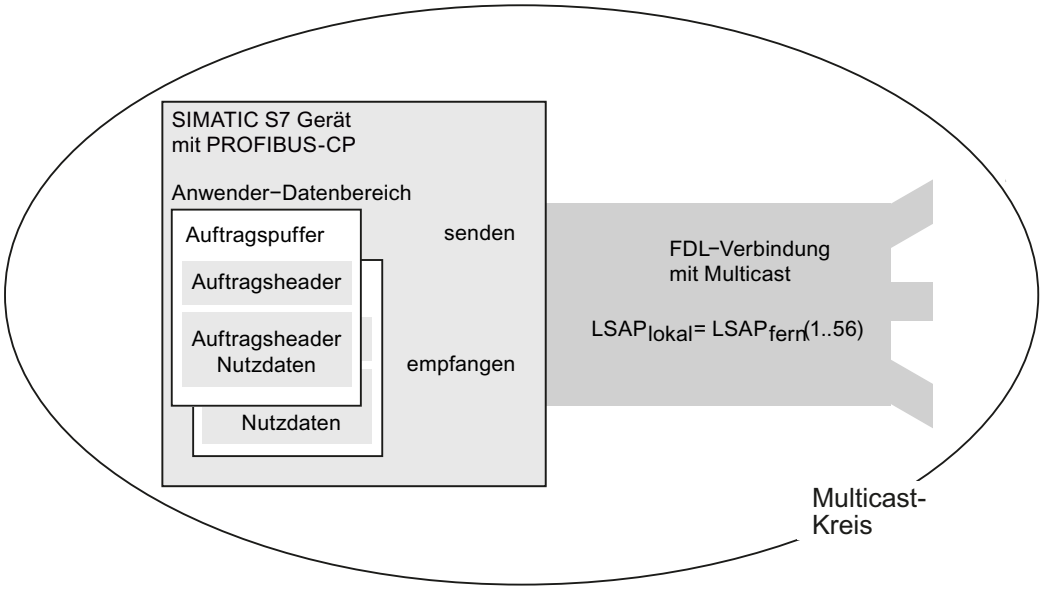

FDL-Verbindung mit Multicast projektieren

Wählen Sie beim Anlegen der FDL-Verbindung als Verbindungspartner/Station "Alle Multicast-Teilnehmer".

Siehe auch

UDP- / FDL-Verbindungen mit Multicast / Broadcast anlegen (Seite 243)

Datenvolumen und Mengengerüst

Die Anzahl der FDL-Verbindungen, die der jeweilige PROFIBUS-CP unterstützt, entnehmen Sie bitte der Gerätedokumentation zum PROFIBUS-CP. Durch Hinzunahme weiterer CPs kann die Anzahl der Verbindungen pro Station erhöht werden.

Pro Auftragspuffer können bis zu 236 Byte Nutzdaten übertragen werden. Der Auftragsheader belegt zusätzlich 4 Byte.

Kommunikation über FDL einrichten

Voraussetzung

- Projektierungs-Software: STEP 7 Professional V14
- Endpunkt der Verbindung: CPU S7-1500 ab Firmware-Version V2.0 mit dem Kommunikationsmodul CM 1542-5 mit Firmware-Version V2.0

Einrichten einer konfigurierten FDL-Verbindung

Um in STEP 7 eine konfigurierte FDL-Verbindung einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie im Programmeditor eine Anweisung TSEND_C an.
2. Selektieren Sie die Anweisung TSEND_C und navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Verbindungsparameter".
3. Wählen Sie unter Endpunkt den Partner-Endpunkt aus. Nutzen Sie einen der beiden folgenden Partner-Endpunkten:
   – CPU S7-1500 mit CM 1542-5
   – Unspezifiziert
4. Wählen Sie unter Konfigurationsart "Konfigurierte Verbindung verwenden aus".
5. Wählen Sie unter Verbindungstyp "FDL" aus.
6. Wählen Sie unter Schnittstelle die folgenden Schnittstellen aus:
   – Lokal: PROFIBUS-Schnittstelle des CM 1542-5
   – Spezifizierter Partner: PROFIBUS-Schnittstelle des CM 1542-5
7. Wählen Sie bei Verbindungsdaten die Einstellung <neu> aus.

Das folgende Bild zeigt eine vollständig konfigurierte FDL-Verbindung in STEP 7.

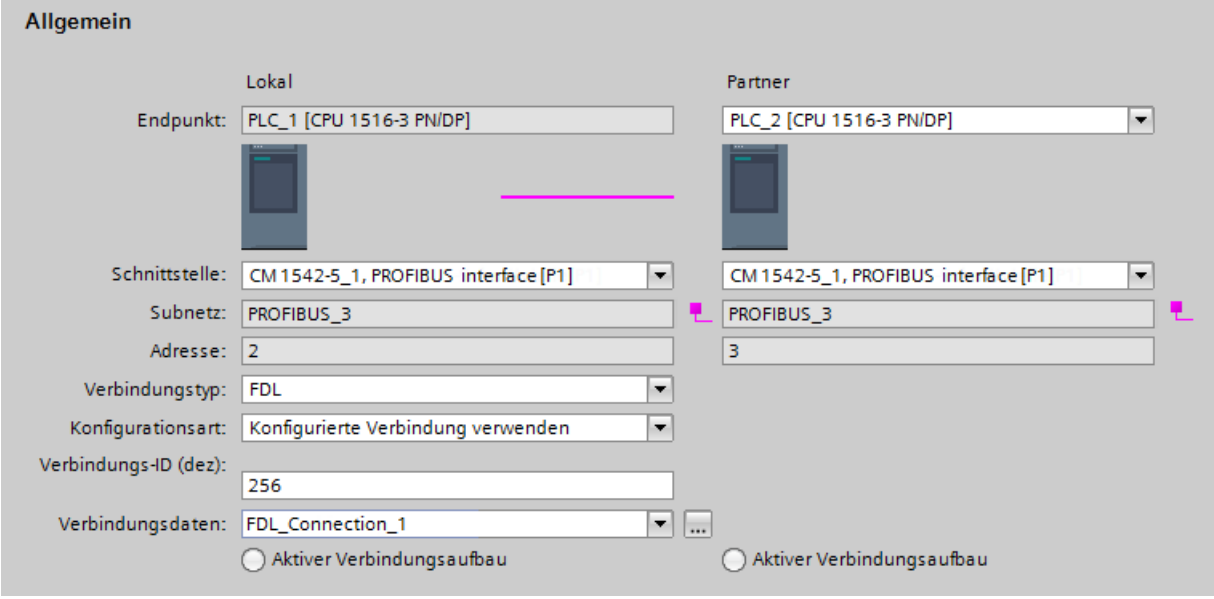

Bild 1-9 FDL-Verbindung konfigurieren

Einrichten einer FDL-Verbindung im Anwenderprogramm

Für die Kommunikation über FDL müssen Sie jeweils den Datenbaustein des Systemdatentyps TCON_FDL selbst erstellen, parametrieren und direkt an der Anweisung aufrufen. Gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_FDL.
   Das folgende Beispiel zeigt den globalen Datenbaustein "FDL_connection", in dem die Variable "FDL_connection" vom Datentyp TCON_FDL definiert ist.

FDL_connection

	Name	Datentyp	Startwert	R...	E...	S..	S..	Kommentar
1	Static							
2	FDL_connection	TCON_FDL			☑		☑	
3	InterfaceId	HW_ANY	0		☑		☑	HW identifier of PB interface submodule
4	ID	CONN_OUC	16#0		☑		☑	connection reference / identifier
5	ConnectionType	Byte	16#15		☑		☑	type of connection: 21= FDL connection
6	ActiveEstablished	Bool	false		☑		☑	active/passive connection establishment
7	ServiceId	Byte	16#0		☑		☑	service id: 0 – default, 1 – SDA, 2 – SDN
8	RemotePBAddress	Byte	16#0		☑		☑	remote ProfiBus partner address
9	LocalPBAddress	Byte	16#0		☑		☑	local ProfiBus partner address
10	RemoteLSAP	Byte	16#0		☑		☑	remote PB link-layer service access point
11	LocalLSAP	Byte	16#0		☑		☑	local PB link-layer service access point

Bild 1-10 FDL-Verbindung programmieren

3. Programmieren Sie die Parameter der FDL-Verbindung (z. B. die PROFIBUS-Adressen) in der Variablen vom Datentyp TCON_FDL.
4. Legen Sie im Programmeditor eine Anweisung TCON an.
5. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variable vom Datentyp TCON_FDL.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "FDL_Connection" (Datentyp TCON_FDL) verschaltet.

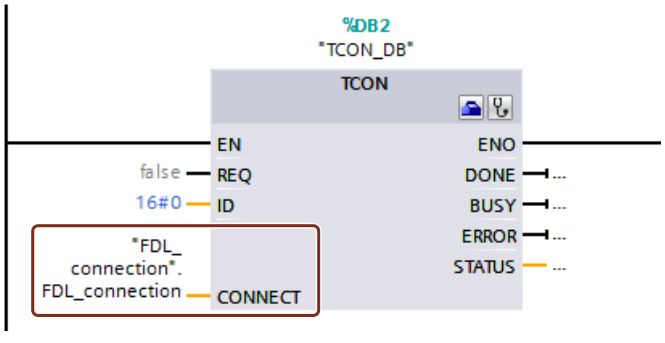

Bild 1-11 Beispiel: Anweisung TCON für FDL-Verbindung

E-Mail-Verbindung

E-Mail-Verbindung einrichten

E-Mails können vom Automatisierungssystem gesendet, nicht jedoch empfangen werden.

Für das Versenden der E-Mail im Anwenderprogramm der S7-CPU verwenden Sie den Sendeaufruf der SEND/RECEIVE-Schnittstelle (FC AG_SEND / AG_LSEND).

Übersicht zur Projektierung

Richten Sie für das Senden von E-Mails genau eine E-Mail Verbindung pro CP ein. Mit der E-Mail-Verbindung ist der Mail-Server festgelegt, über den sämtliche vom CP gesendeten E-Mails zugestellt werden.

Eine E-Mail-Verbindung kann wie folgt eingerichtet werden:

- Verbindungsprojektierung (Standard-Anwendung)
  Dieser Anwendungsfall wird nachfolgend beschrieben.
- Anwenderprogramm mit FB CP_CONFIG und Konfigurations-Datenbaustein
  Es gibt Anwendungsbereiche, bei denen es vorteilhaft ist, die Kommunikationsverbindungen nicht über die Projektierschnittstelle, sondern über spezifische Applikationen programmgesteuert einzurichten.

Vorgehen - Beispiel für grafische Projektierung

Im folgenden ist die Projektierung einer E-Mail-Verbindung beschrieben. Prinzipiell gehen Sie dabei so vor wie bei einer unspezifizierten Verbindung: Unspezifizierte Verbindung anlegen (Seite 241).

Wählen Sie den Verbindungstyp E-Mail-Verbindung in der Klappliste.

Damit ist der Verbindungs-Modus aktiviert: die für den gewählten Verbindungstyp infrage kommenden CPUs sind selektiert und hervorgehoben dargestellt.

1. Klicken Sie auf die CPU, von der eine Verbindung ausgehen soll.
2. Bewegen Sie den Mauszeiger innerhalb der CPU und bestätigen Sie den Verbindungsendpunkt durch erneuten Mausklick.
   Eine unspezifizierte Verbindung ist angelegt.
3. Geben Sie im Inspektorfenster unter "Eigenschaften > Adressdetails" die Adressparameter entsprechend den Angaben zu den Adressdetails ein:
   E-Mail Adressdetails (Seite 295)

Nach dem Laden der Projektierungsdaten kann das Anwenderprogramm E-Mails über diese Verbindung E-Mails senden.

OPC-Server als SMTP-Server nutzen

Statt einer unspezifizierten Verbindung können Sie als Verbindungspartner einen OPC-Server als SMTP-Server wählen. Dadurch wird bei den Partner-Adressdetails automatisch die IP-Adresse der SMTP-Schnittstelle der PC-Station eingetragen.

Hinweis

PC-Schnittstelle muss SMTP-Protokoll unterstützen

Die E-Mail-Verbindung über den OPC-Server wird nur dann konsistent angelegt, wenn in der PC-Station eine Schnittstelle mit SMTP-Protokollunterstützung aktiviert ist. Beachten Sie die Dokumentation zu der von Ihnen verwendeten Baugruppe.

Siehe auch

DNS-Konfiguration (Seite 3226)

E-Mail Adressdetails

Bezug

Verbindungsprojektierung in der Parametergruppe "Eigenschaften > Allgemein > Adressdetails".

Parameter

Geben Sie die Parameter ein, die für die Zustellung der E-Mail benötigt werden. Folgende Parameter sind eingebbar:

Parameter	Beschreibung	Beispiel
E-Mail Server (SMTP)	Adresse des Mail-Servers, über den die E-Mails gesendet werden. Die IP-Adresse kann absolut oder symbolisch angegeben werden. Für die symbolische Eingabe gilt: Ein gültiger Name hat 1 bis 64 Zeichen; von diesen Zeichen muß mindestens 1 Zeichen ein Buchstabe sein. Werden in der Eingabe ein oder mehrere Zeichen "." verwendet, gilt zusätzlich: Ein gültiger Name hat hinter dem letzten "." 1 bis 3 Zeichen; von diesen Zeichen muß mindestens 1 Zeichen ein Buchstabe sein. Die symbolische Angabe setzt voraus, daß dem Internet-CP die Adresse des Domain Name System (DNS) bekannt ist. Ein entsprechender Eintrag ist bei der Projektierung des Internet-CP in der DNS-Konfiguration vorzunehmen. Maximal eingebbar sind 64 Zeichen	absolut:140.80.0.4 symbolisch:t-online.de
Default Absendername	Angabe einer Adresse, die immer dann in die E-Mail als Absenderadresse eingefügt wird, wenn im Header der E-Mail die Absenderangabe (Parameter FROM) leer ist. Die Angabe ist in der Regel ohne Bedeutung für die Mail-Zustellung. Sie dient zur Information für den Mail-Empfänger. Da einige Mail-Server Aufträge nicht weiterleiten, wenn die Absenderangabe leer ist, wird empfohlen, einen Eintrag vorzunehmen! Maximal eingebbar sind 126 Zeichen. **Hinweis:** Beachten Sie, dass ein Default Absendername angegeben sein muss, wenn von der Spezialdiagnose aus eine Test-Mail initiiert werden soll.	Station2.CPU412@xy.werk2.de

Siehe auch

DNS-Konfiguration (Seite 3226)

E-Mail senden

Vorgehen

Für das Senden einer E-Mail führen Sie folgende Schritte aus:

1. Sie stellen die E-Mail-Daten in einem Datenbaustein bereit.
2. Sie verwenden die Anweisung AG_SEND bzw. AG_LSEND im Anwenderprogramm.

Voraussetzung

Sie können E-Mails versenden, wenn die E-Mail-Verbindung über die Verbindungsprojektierung eingerichtet wurde. Sie verwenden die bei der Verbindungsprojektierung vorgegebene ID beim Aufruf der FC AG_SEND/AG_LSEND.

Datenbaustein

Die gesamte E-Mail, also die Adressangaben und die Nachricht selbst, wird in einem beliebigen Datenbaustein aufgebaut. Nachfolgend finden Sie anhand eines Beispiels in AWL-Notation entsprechende Angaben zur erforderlichen DB-Struktur.

Verwenden Sie den Programmiereditor zum Anlegen und Eingeben der DB-Daten.

Die folgende Tabelle enthält Informationen über E-Mail-Datenbausteine in AWL-Notation:

Adresse	Name	Typ	Anfangswert	Kommentar	Eintrag
0.0		STRUCT			
+0.0	TO[1)]	STRING[40]	'TO:name.name@t-online.de;'	Empfänger	zwingend
+42.0	CC[1)]	STRING[40]	'CC:name.name@t-online.de;'	CC Empfänger	optional
+84.0	FROM	STRING[40]	'FROM:anlage.werk2@xyz-on-line.de;'	Absender	optional
+126.0	SUB	STRING[40]	'SUB:Status Station 7;'	Thema	optional
+168.0	Text	STRING[100]	'TXT:Störung in Anlagenabschnitt 2;'	Mail-Text	zwingend
+270.0	Anlage	STRING[4]	'BNY:'	Hier wird die Anlage eingeleitet[3)]	optional
+276.0	Wert1	BYTE	**B#16#27**[2)]	Anlage/Binärwert[3)]	optional
+277.0	Wert2	BYTE	**B#16#03**[2)]	Anlage/Binärwert[3)]	optional
=278.0		END_STRUCT			

1) Es können mehrere Empfänger angegeben werden. Die Angaben sind dann durch Komma zu trennen.

2) Die fett markierten Angaben werden dem Empfänger als Anlage zugestellt

3) Daten können auch dynamisch versorgt werden.

Hinweise zur Tabelle

- Struktur und Syntax der Daten im E-Mail DB
  Die hier vorgeschlagene Struktur mit mehreren STRINGs ist eine von mehreren Varianten. Entscheidend sind die Einträge in der Spalte "Anfangswert" mit den darin enthaltenen Kennungen (TO:, SUB:, CC:, FROM:, TXT:, BNY:) die in exakt dieser Schreibweise im DB zur Kennzeichnung der Mail-Inhalte verwendet werden müssen! Sämtliche Einträge müssen hierbei mit Semikolon abgeschlossen werden; lediglich beim letzten Eintrag darf kein Semikolon verwendet werden.
  Die String-Länge ist in der Tabelle nur beispielhaft angegeben; sie kann der tatsächlichen Zeichenzahl angepasst werden (Ausnahme: die String-Länge für die Kennzeichnung der Anlage muss mit [4] angegeben werden).
  Eine weitere Variante wäre zum Beispiel, nur insgesamt einen STRING zu verwenden und diesem den gesamten Text mit den Kennungen zuzuweisen.
- Falls es Probleme bereitet, das @-Zeichen einzugeben, verwenden Sie die Eingabe ALT +64.

- Anlagen
  Die im E-Mail DB eingetragenen Nutzdaten können dem Empfänger ganz oder teilweise auch als Anlage zugestellt werden. Die Daten müssen hierzu vom Absender mit der Kennung 'BNY:' versehen werden.
  Die nach dieser Kennung angegebenen Daten werden dann dem Empfänger als Anlage zugestellt.
  In der Tabelle umfasst die Anlage 2 Byte; dies ist nur ein Beispiel! Es können beliebig komplexe Anlagen eingetragen werden.
- Datenlänge
  Die im Aufruf AG_SEND/AG_LSEND angegebene Datenlänge muss mindestens die Länge der Daten im DB umfassen; beachten Sie die Angaben in der Spalte Adresse im Programmiereditor (Hinweis: die Angabe entspricht der Anzahl Byte).

E-Mail senden mit AG_SEND/AG_LSEND [1)]

Verwenden Sie die Anweisung AG_SEND bzw. bei Datenlängen >240 Byte AG_LSEND, um eine E-Mail zu versenden. Eine detaillierte Beschreibung zu den Aufrufparametern finden Sie in der Online-Hilfe zu den Anweisungen.

Siehe auch AUTOHOTSPOT

Beispiel:

AWL	Erläuterung
call fc 50	//AG_LSEND Bausteinaufruf
ACT := M 10.0	//Bit für den Auftragsanstoß
ID := MW 12	//Verbindungs-ID (Verbindungsprojektierung)
LADDR := W#16#0100	//BG-Adresse 256Dez. in Hardware-Konfigura-tion
SEND := P#db99.dbx10.0 byte 278,	//Adresse des Datenbausteins; DB-Länge
LEN := MW 14	//Länge des zu sendenden Datenbereiches
DONE := M 10.6	//Adresse für Rückgabeparameter DONE
ERROR := M 10.7	//Adresse für Rückgabeparameter ERROR
STATUS := MW 16	//Adresse für Rückgabeparameter STATUS

PtP-Verbindung

PtP-Kommunikation

Kommunikationsarten der PtP-Kommunikation

Bei der Kommunikation über eine PtP-Verbindung unterscheidet man zwischen zwei verschiedenen Kommunikationsarten:

- Homogene Kommunikation
- Inhomogene Kommunikation

Durch die Unterstützung der PtP-Kommunikation können externe Geräte wie Drucker angesprochen werden.

Homogene PtP-Kommunikation

Der PtP-CP leitet Daten zwischen Baugruppen weiter, die das S7-Protokoll beherrschen. Dem Anwender steht der volle Vorrat an Kommunikationsfunktionen zur Verfügung, welche die über PtP miteinander verbundenen Endgeräte (z. B. CPU mit CPU oder PG mit CPU) beherrschen.

Die PtP-CPs sind bei dieser Verbindung nicht Endpunkt der Verbindung, sondern übernehmen die Funktion eines Routers.

Bei homogener Kopplung können alle in S7 definierten Kommunikationsfunktionen genutzt werden, welche von den beiden Verbindungspartnern beherrscht werden. Eine Protokollabwicklung für spezifische Aufrufe von Kommunikationsanweisungen ist daher auf dem CP nicht notwendig.

Inhomogene PtP-Kommunikation

Der PtP-CP arbeitet wie ein Gateway über andere Protokolle, wie RK512, 3964, Offener Treiber und Sonderprotokolle. Ausgewählte Bausteine dienen als programmsynchrone Anwenderschnittstelle im Anwenderprogramm. Empfangsseitig kann das Anwenderprogramm als Server auch über programmasynchrone Kommunikationsfunktionen über die inhomogene PtP-Kopplung angesprochen werden.

Der PtP-CP ist bei der inhomogenen Kommunikation aus kommunikationstechnischer Sicht Endpunkt der S7-Verbindung. Die Kommunikation zwischen CPU und CP erfolgt dabei auf S7- Protokollelementen. Der CP übernimmt die Umsetzung dieser Protokollelemente auf das PtP- Protokoll und umgekehrt.

Für die inhomogene Kommunikation können folgende Kommunikationsanweisungen auf der CPU verwendet werden:

- BSEND
- BRCV
- PUT
- GET

Auf dem PtP-CP müssen zu den Anweisungen BSEND, BRCV, PUT und GET als Client (CP ist aktiv) und PUT und GET als Server (CP ist passiv) die entsprechenden S7-Protokollelemente implementiert werden.

Anweisungen	Client	Server
BSEND	ja	nein
BRCV	ja	nein
PUT	ja	ja
GET	ja	ja

Kommunikationsfunktionen

Die inhomogene Kommunikation kann abhängig von der Anwendung in zwei unterschiedlichen Qualitäten erfolgen:

- Gesteuerte Kommunikationsfunktionen:
  Bei gesteuerten Kommunikationsfunktionen ist es möglich, die Übertragung gesteuert durchzuführen. Dies bedeutet, dass die Übertragung nicht nur auf der aktiven Seite gestartet werden kann, sondern auch auf der passiven Seite gesteuert freigegeben und das Ende der Übertragung erkannt werden.
  Diese Funktionalität wird bei den Kommunikationsanweisungen in S7 mit den Parametern "EN_R" (Empfangsbereitschaft) und "NDR" (Datenübernahme fertig) realisiert.
- Ungesteuerte Kommunikationsfunktionen:
  Bei ungesteuerten Kommunikationsfunktionen wird die Übertragung vom aktiven Partner ohne Eingriffsmöglichkeiten des passiven Partners durchgeführt.

PtP-Verbindung parametrieren

Hardware für PtP konfigurieren

Um eine PtP-Verbindung aufzubauen, benötigen Sie einen CP 441. Kommunikationsmodule vom Typ CP 441 besitzen Schächte für die Verwendung von ein (CP 441-1) oder zwei (CP 441-2) PtP-Schnittstellenmodulen.

PtP-Schnittstellenmodule vom Typ IF963 finden Sie im Hardware-Katalog der SIMATIC S7-400 unterhalb der Kommunikationsmodule für PtP. Die IF963-Module gibt es für folgende Schnittstellentypen:

- RS232C
- TTY
- RS422/485

Die IF963-Module unterstützten je nach Schnittstellentyp maximal folgende Kommunikationsprotokolle:

- ASCII
- 3964(R)
- RK512
- Drucker
- Modbus Master
- Modbus Slave

Wenn Sie eine PtP-Schnittstelle in einem CP 441 verwenden, wird automatisch ein PtP-Subnetz angezeigt und eine unspezifizierte PtP-Verbindung hinzugefügt.

Nachdem Sie den CP 441 oder direkt die gewünschte Schnittstelle ausgewählt haben, können Sie im Inspektorfenster das Protokoll auswählen und konfigurieren.

PtP-Verbindung zu einem S7-Gerät als Verbindungspartner

Für den lokalen Partner können Sie maximal 8 Verbindungen in die Verbindungstabelle eintragen. Für den Verbindungspartner müssen Sie nur **eine** Verbindung in die Verbindungstabelle eintragen. Der Verbindungspartner kann Daten von den 8 Verbindungen über diese eine Verbindung empfangen.

Beispiel: Festlegungen für den lokalen Partner:

- Protokoll 3964(R)
- Kommunikationsrichtung 1: Lokal --> Partner
- max. 8 Verbindungen, lokale IDs: 1000; 1001, ..., 1007

Festlegungen für den Verbindungspartner:

- Protokoll 3964(R)
- Kommunikationsrichtung 2: Partner --> Lokal
- max. 1 Verbindung, lokale ID: 1000

Anzahl Verbindungen

Die Anzahl der maximal projektierbaren Verbindungen pro Gerät (CPU/FM) ist abhängig von den Ressourcen der eingesetzten Baugruppe.

PtP - Kommunikationsrichtung

Kommunikationsrichtung

Für PtP-Verbindungen können Sie verschiedene Kommunikationsrichtungen festlegen. Aktivieren Sie dazu das entsprechende Optionskästchen:

- Lokal → Partner:
  Verbindungen, über die Sie Telegramme senden wollen.
- Partner → Lokal:
  Verbindungen, über die Sie Telegramme empfangen wollen.
- Lokal ↔ Partner:
  Verbindungen, über die Sie Telegramme senden und empfangen wollen.

Hinweis

Genauere Informationen zur Festlegung der Kommunikationsrichtung finden Sie in der Dokumentation zu Ihrer Baugruppe.

PtP - Verbindungsweg

Einstellungen zum Verbindungsweg

Unter "Allgemein > Verbindungsweg" und können Sie unter den Einstellungen zur Kommunikationsrichtung den Weg einstellen, den die Verbindung vom lokalen Endpunkt zum Verbindungspartner nimmt. Sie können den Weg in Abhängigkeit von den konfigurierten PtP-CPs wählen. Beispiel: Der lokale Partner enthält zwei PtP-CPs. In diesem Fall können Sie wählen, über welchen CP die Verbindung "laufen" soll.

Besonderheit: Im Unterschied zu den S7-Verbindungen wird für die Projektierung von PtP-Verbindungen zu einem unspezifizierten Partner keine Vernetzung des lokalen Partners vorausgesetzt. Die Kommunikationspartner müssen Sie erst dann in Ihrer realen Anlage vernetzen, bevor Sie die Verbindung in Betrieb nehmen.

Auswahlmöglichkeiten für die Verbindungspartner

- Über CP:
  Für den lokalen Partner können Sie den PtP-CP auswählen, über den die Verbindung "laufen" soll (Name des CPs mit Angabe von Rack und Steckplatz).
- Schnittstelle:
  CPs aus SIMATIC S7 besitzen mehrere Kanäle (Schnittstellen), über die die PtP-Verbindungen aufgebaut werden können. Es werden der Kanal und das für den Kanal benutzte Protokoll angezeigt (Kanal/Protokoll). Das Protokoll haben Sie mit der speziellen Projektiersoftware für den CP projektiert.
- Typ:
  Hier wird der Schnittstellentyp "PtP" angezeigt.

"Verbindung ausgewählt über RK512 CPU Nr.:" (lokal)

Multicomputingfähige Automatisierungssysteme (wie z. B. SIMATIC S7-400) können mehrere CPUs enthalten. Aus diesem Grund müssen Sie eine CPU-Nummer angeben, über die der Partner die Verbindung ansprechen kann.

Wenn Sie als Kommunikationsrichtung **Partner -> Lokal** oder **Lokal <-> Partner** gewählt haben und Telegramme über einen SFB vom Partner empfangen wollen, dann geben Sie hier die Nummer der CPU (1 bis 4) des **lokalen Partners** an.

Die hier angegebene CPU-Nummer muss mit der im Befehlstelegrammkopf des CPs eingetragenen CPU-Nummer übereinstimmen (Aufbau Befehlstelegramm siehe Dokumentation zum eingesetzten CP). Bei Übereinstimmung spricht der Partner über diese CPU-Nummer die Verbindung an.

Hinweis

Wenn Sie Telegramme über SFBs senden und empfangen wollen, dann müssen Sie beide Felder, unter "Lokal" und "Partner", ausfüllen. Die maximale Anzahl der Verbindungen, über die Sie Telegramme senden/empfangen können, finden Sie in der Dokumentation zum eingesetzten CP.

"RK512 CPU Nr.:" (Partner)

Multicomputingfähige Automatisierungssysteme (wie z. B. SIMATIC S7-400) können mehrere CPUs enthalten. Aus diesem Grunde müssen Sie eine CPU-Nummer angeben, zu der die Verbindung gehen soll.

Wenn Sie als Kommunikationsrichtung **Lokal -> Partner** oder **Lokal <-> Partner** gewählt haben und Telegramme über einen SFB an den Partner senden wollen, dann geben Sie hier die Nummer der CPU (1 bis 4) **des Verbindungspartners** an.

Der CP trägt die hier eingegebene CPU-Nr. in den Befehlstelegrammkopf des zu sendenden Telegramms ein. Über diese CPU-Nummer wird eine der 4 CPUs des Verbindungspartners angesprochen.

Bei der Übertragung zu Fremdgeräten ist es auch möglich, als CPU-Nr. "0" anzugeben. Damit wird Broadcast-Zugriff bei der Partnerstation möglich. Beim CP441 als Partner wird in diesem Fall immer die CPU 1 angesprochen.

Hinweis

Wenn Sie Telegramme über SFBs senden und empfangen wollen, dann müssen Sie beide Felder, unter "Lokal" und "Partner", ausfüllen. Die maximale Anzahl der Verbindungen, über die Sie Telegramme senden/empfangen können, finden Sie in der Dokumentation zum eingesetzten CP.

PtP - Besondere Verbindungseigenschaften

Bezug

In der Parametergruppe "Eigenschaften > Allgemein > Besondere Verbindungseigenschaften" können Sie zusätzliche Eigenschaften des lokalen Verbindungsendpunkts festlegen.

Einseitig

Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.

Aktive Verbindung aufbauen

Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von diesem Gerät aus erfolgen soll. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.

- EIN: Verbindung wird aktiv aufgebaut.
- AUS: Verbindung wird vom Partner aufgebaut.

Wenn Sie beim Anlegen der Verbindung als Verbindungspartner "unspezifiziert" gewählt haben, ist die Option standardmäßig abgewählt. Wenn Sie die Option aktivieren, müssen Sie die Adresse des Partners im Register "Adressen" spezifizieren.

Hinweis

Nicht jeder Teilnehmer besitzt die Fähigkeit, den Verbindungsaufbau zu übernehmen. In diesem Fall muss der Partner den Aufbau übernehmen; die Optionskästchen sind gegraut und können nicht verändert werden.

Betriebszustandsmeldungen senden

Über eine bestehende Verbindung können die Verbindungspartner auch Ihre Betriebszustandsmeldungen austauschen.

Aktivieren Sie das Optionskästchen, wenn der lokale Teilnehmer seine Betriebszustandsmeldungen an den Partner senden soll. Die Zustandsmeldungen können dann im Partner über die Anweisung USTATUS (SFB 23) empfangen werden.

1.3.3 Topologie anzeigen und projektieren

1.3.3.1 Überblick über die Topologiesicht

Funktionen der Topologiesicht

Die Topologiesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Ethernet-Topologie anzeigen
  - Sämtliche PROFINET-Geräte und passiven Ethernet-Komponenten des Projekts samt Ports anzeigen
  - Verschaltungen zwischen den Ports anzeigen
  - Zugehörige logische Netze anzeigen
  - Diagnoseinformationen aller Ports anzeigen
- Ethernet-Topologie projektieren
  - Verschaltungen der Ports anlegen, ändern und löschen
  - Stationen, Geräte, Schnittstellen, Ports umbenennnen
  - PROFINET-Geräte und passive Ethernet-Komponenten aus dem Hardware-Katalog zum Projekt hinzufügen
- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren
  - Offline-/Online-Vergleich von Ethernet-Baugruppen, -Ports und -Portverschaltungen durchführen
  - Online vorliegende Topologieinformationen in das Offline-Projekt übernehmen

Hinweis

Geräte ohne gültige IP-Adresse

Ohne eine gültige IP-Adresse können von einem Gerät keine Topologieinformationen (LLDP) gelesen werden.

Um zu vermeiden, dass ein Gerät keine gültige IP-Adresse hat, können Sie in den Einstellungen des TIA Portals zur Hardware-Konfiguration vorgeben, dass Geräten ohne gültige IP-Adresse temporär eine IP-Adresse zugewiesen wird.

Hinweis

Subnetz-Zugehörigkeit

Vor der Topologie-Ermittlung müssen Sie für die lokale Schnittstelle Ihres PG/PC diejenige Netzadresse einstellen, die auch die zu ermittelnden Ethernet-Komponenten in der realen Anlage haben. Wenn Sie diese Voraussetzung nicht einhalten, können keine erweiterten Topologieinformationen (Daten über Ports und Nachbarschaftsbeziehungen) ermittelt werden.

Unterschiede zwischen Netzsicht und Topologiesicht

- Die Netzsicht zeigt alle logischen Subnetze des Projekts an.
  Die Topologiesicht zeigt alle Ethernet-Komponenten des Projekts an. Dazu gehören auch passive Komponenten wie Switches und Medienkonverter und Leitungen.

  Hinweis

  Darüber hinaus werden auch Stationen mit Nicht-Ethernet-Komponenten angezeigt, wenn sich in der Station mindestens eine Ethernetkomponente befindet.

- Die Position eines Geräts in der Netzsicht und seine Position in der Topologiesicht sind zunächst unabhängig voneinander, d. h. in der Regel befindet sich ein und dasselbe Gerät in den beiden Sichten jeweils an einer anderen Stelle.
  Wenn Sie die Geräte in der Topologiesicht wie in der Netzsicht ausrichten wollen, dann klicken Sie auf das folgende Symbol in der Funktionsleiste der Topologiesicht:

Symbol	Bedeutung
	Verändert die Position der Geräte in der Topologiesicht so, dass die neue Position der Position in der Netzsicht möglichst nahe kommt.

- Wennn Sie den Hardware-Katalog aus der Topologiesicht heraus öffnen, werden Ihnen ausschließlich Geräte mit Ethernet-Schnittstelle angezeigt.

Aufbau der Topologiesicht

Die Topologiesicht (Seite 21) besteht im Wesentlichen aus einem grafischen Bereich (im Folgenden als grafische Ansicht bezeichnet) und einem tabellarischen Bereich (im Folgenden als tabellarische Ansicht bezeichnet).

Welche Funktionen gibt es in der grafischen, welche in der tabellarischen Ansicht?

- Ethernet-Topologie anzeigen

Funktion	Grafische Ansicht	Tabellarische Ansicht
Sämtliche PROFINET-Geräte und passiven Ethernet-Komponenten des Projekts samt Ports anzeigen	ja	ja
Verschaltungen zwischen den Ports anzeigen (einschließlich Medientyp)	ja	ja
Zugehörige logische Netze anzeigen	nein	ja
Eigenschaften der Leitungen zwischen den Ports anzeigen	nein	ja
Diagnoseinformationen aller Ports anzeigen	ja	ja

- Ethernet-Topologie projektieren

Funktion	Grafische Ansicht	Tabellarische Ansicht
Verschaltungen von Ports anlegen, ändern und löschen	• Anlegen: ja • Ändern: ja • Löschen: ja	• Anlegen: ja • Ändern: ja • Löschen: ja
Stationen, Geräte, Schnittstellen, Ports umbenennnen	nein	ja
PROFINET-Geräte und passive Ethernet-Komponenten aus dem Hardware-Katalog zum Projekt hinzufügen	ja	nein

- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren

Funktion	Grafische Ansicht	Tabellarische Ansicht
Offline-/Online-Vergleich von Ethernet-Baugruppen, -Ports und -Portverschaltungen durchführen	nein	ja
Online vorliegende Topologieinformationen in das Offline-Projekt übernehmen	nein	ja

Aufbau der tabellarischen Ansicht

Die tabellarische Ansicht besteht aus den folgenden beiden Registern:

- Register "Topologieübersicht"
- Register "Topologievergleich"

Das Register "Topologieübersicht" bietet die folgenden Funktionen:

- Anzeige der projektierten Topologie mit hierarchischen Strukturen (von der Station bis zum Port)
- Projektierung der Ethernet-Topologie
- Anzeige des Diagnosestatus von Hardware-Komponenten

Das Register "Topologievergleich" bietet die folgende Funktion:

- Ermittlung und Minimierung der Unterschiede zwischen Soll- und Ist-Topologie

1.3.3.2 Topologiesicht starten

Voraussetzung

Die Geräte- oder die Netzsicht innerhalb des Hardware- und Netzwerkeditors ist geöffnet.

Vorgehen

Um die Topologiesicht Ihres Projekts zustarten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Register "Topologiesicht".

Oder:

1. Öffnen Sie die Netzsicht des Hardware-Editors.
2. Selektieren Sie ein PROFINET-Gerät oder eine PROFINET-Baugruppe.
3. Wählen Sie im Kontextmenü den Befehl "Gehe zur Topologiesicht".

Ergebnis

Die Topologiesicht wird gestartet. Falls Sie die Topologiesicht mittels Kontextmenü aufgerufen haben, bleibt die selektierte Komponente nach der Umschaltung selektiert.

1.3.3.3 Topologie anzeigen

Grafische Ansicht der projektierten Topologie anzeigen

Was wird dargestellt?

Die grafische Ansicht der projektierten Topologie zeigt das Folgende an:

- Projektierte PROFINET-Geräte und passive Ethernet-Komponenten samt Ports
- Projektierte Stationen mit Nicht-Ethernet-Komponenten, wenn sich in der Station mindestens eine Ethernet-Komponente befindet
- Projektierte Verschaltungen zwischen den Ports

Art der Darstellung

Die grafische Ansicht der Topologiesicht und der Netzsicht sind sich hinsichtlich der Art der Darstellung sehr ähnlich:

- Komponenten werden gegenüber der Gerätesicht vereinfacht dargestellt.
- Die Verschaltungen zwischen Ports werden als waagrechte und senkrechte Linien dargestellt. Diese sind gestrichelt, wenn es sich um die Verschaltung zwischen einem Werkzeugwechslerport und seinen möglichen Partnerports handelt.

Tabellarische Ansicht der projektierten Topologie anzeigen

Voraussetzung

- Die tabellarische Ansicht der Topologie ist geöffnet.
- Sie befinden sich im Register "Topologieübersicht".

Was wird dargestellt?

Die tabellarische Ansicht der projektierten Topologie zeigt mit Ausnahme der projektierten logischen PROFINET-Subnetze das Gleiche an wie die grafische Ansicht:

- Sämtliche projektierten PROFINET-Geräte und passiven Ethernet-Komponenten samt Ports
- Sämtliche projektierten Stationen mit Nicht-Ethernet-Komponenten, wenn sich in der Station mindestens eine Ethernet-Komponente befindet
- Projektierte Verschaltungen zwischen den Ports
  Zu einem Port mit der Eigenschaft "wechselnder Partnerport" gehören so viele ausgefüllte Zeilen, wie er potenzielle Partnerports hat, plus eine Leerzeile.

Art der Darstellung

Wie der Name schon sagt, besteht die tabellarische Ansicht der Topologiesicht aus einer Tabelle, der Topologieübersichtstabelle. Sie zeigt die projektierte Topologie mit ihren hierarchischen Strukturen an (von der Station bis zum Port). Sie ist wie die Netzübersichtstabelle aufgebaut. Sie besteht aus den folgenden Spalten:

- Gerät/Port
  Das ist die wichtigste Spalte der Tabelle. Die Einträge dieser Spalte sind hierarchisch aufgebaut, wobei das letzte Element in der Hierarchie die PROFINET-Ports sind. Sie können die hierarchischen Einträge aufklappen und zuklappen. Für eine CPU beispielsweise besteht ein Eintrag aus den folgenden Elementen:
  - Stationsname
  - Gerätename
  - Name der PROFINET-Schnittstelle
  - Name der Ports

  Hinweis: Alle weiteren Spalten enthalten nur in den Zeilen mit den Port-Namen Einträge.
- Steckplatz
  Zeigt die Nummer des Steckplatzes auf einem Baugruppenträger an, auf dem die jeweilige Baugruppe gesteckt ist. Bei Schnittstellen wird zusätzlich die Bezeichnung der Schnittstelle angezeigt.
- Index (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Diese Spalte ist nur für PC-Stationen relevant. Der Index ist die virtuelle Steckplatzadresse der Komponente.

- Gerätetyp (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Zeigt an, auf welche Art von Station, Gerät oder Schnittstelle sich die Tabellenzeile bezieht oder ob sie zu einem Port gehört.
- Artikel-Nr. (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Artikel-Nr. des Geräts
- Subnetz (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Projektiertes Subnetz, zu dem die Schnittstelle gehört
- Master-/IO-System (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Zeigt an, ob die Schnittstelle zu einem PROFIBUS DP-Mastersystem oder einem PROFINET IO-System gehört.
- IP-Adresse (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
  Projektierte Adresse der Schnittstelle im Subnetz
- Partner-Station
  Name der Station, die den Partnerport enthält
- Partner-Gerät
  Name des Geräts, das den Partnerport enthält
- Partner-Schnittstelle
  Schnittstelle, zu der der Partnerport gehört
- Partnerport
- Leitungsdaten
  Enthält die Leitungslänge und die Signallaufzeit der Leitung, die die Ports verbindet

Grundfunktionen für Tabellen

Die Topologieübersichtstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden
  Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Aktuelle Darstellung speichern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen

Diagnosestatus von Ports in der grafischen Ansicht anzeigen

Voraussetzung

Die grafische Ansicht der Topologiesicht ist geöffnet.

Vorgehen

Um den Diagnosestatus der Ports zu ermitteln, gehen Sie folgendermaßen vor:

1. Gehen Sie mit der gewünschten Komponente bzw. den gewünschten Komponenten online.

Ergebnis

Die folgenden Symbole werden angezeigt:

- Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
- Falls in mindestens einer unterlagerten Komponente ein Fehler vorliegt, wird zusätzlich das Diagnosesymbol "Fehler in unterlagerter Komponente" in die linke untere Ecke des Diagnosesymbols eingeblendet.
- Für jeden Port wird das zugehörige Diagnosesymbol angezeigt.
- Jede Leitung zwischen zwei Ports, die online sind, erhält die ihrem Diagnosezustand entsprechende Farbe.

Die möglichen Diagnosesymbole für Ports und die farbliche Kennzeichnung von Ethernet-Leitungen finden Sie bei der Beschreibung der Hardware-Diagnose. Siehe: Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

Diagnosestatus von Hardware-Komponenten in der tabellarischen Ansicht anzeigen

Voraussetzung

- Die tabellarische Ansicht der Topologiesicht ist geöffnet.
- Sie befinden sich im Register "Topologieübersicht".

Vorgehen

Um den Diagnosestatus von Hardware-Komponenten der Topologieübersichtstabelle zu ermitteln, gehen Sie folgendermaßen vor:

1. Gehen Sie mit den gewünschten Komponenten online.

Ergebnis

Die folgenden Symbole werden am linken Rand der Topologieübersichtstabelle angezeigt, und zwar in jeder Zeile, die zu der betroffenen Komponente gehört:

- Das zur Hardware-Komponente gehörige Diagnosesymbol wird angezeigt.
- Bei Hardware-Komponenten mit unterlagerten Komponenten wird - falls in mindestens einer unterlagerten Komponente ein Fehler vorliegt - zusätzlich das Diagnosesymbol "Fehler in unterlagerter Komponente" in die linke untere Ecke des Diagnosesymbols der Hardware-Komponente eingeblendet.

Die möglichen Diagnosesymbole für Hardware-Komponenten finden Sie bei der Beschreibung der Hardware-Diagnose. Siehe: Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

Hinweis

Die Anzeige des Diagnosestatus von Hardware-Komponenten in der Topologieübersichtstabelle und der Netzübersichtstabelle ist gleich.

Offline-/Online-Vergleich durchführen und das Ergebnis anzeigen

Voraussetzung

- Die tabellarische Ansicht der Topologiesicht ist geöffnet.
- Sie befinden sich im Register "Topologievergleich".
- Eine Online-Verbindung zu einem oder mehreren Geräten kann bestehen, muss aber nicht.

Vorgehen

Um die Unterschiede zwischen projektierter und tatsächlich vorhandener Topologie zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Offline-/Online-Vergleich".

Ergebnis

Die tatsächlich vorhandene Topologie wird ermittelt und mit der projektierten Topologie verglichen.

Online ermittelte Geräte werden projektierten Geräten automatisch zugeordnet, sofern dies möglich ist. Siehe Automatische Zuordnung von Geräten durch den Offline-/Online-Vergleich (Seite 320)

Die Spalten auf der rechten Seite der Tabelle werden befüllt bzw. aktualisiert:

- Spalten für das Ergebnis des Offline-/Online-Vergleichs: "Status" und "Handlung"
- Spalten für die online ermittelte Topologie

Hinweis

Abbruch eines laufenden Offline-/Online-Vergleichs

Das Klicken auf die Schaltfläche "Offline-/Online-Vergleich" während einer laufenden Topologieermittlung bewirkt den Abbruch der laufenden Topologieermittlung.

Die folgenden Schaltflächen in der Funktionsleiste der Tabelle werden aktiv geschaltet:

Schaltfläche	Name	Bedeutung
	Aktualisieren	Die Ermittlung der online vorhandenen Topologie wird erneut gestartet.
	Synchronisieren	• Online ermittelte Portverschaltungen in das Projekt übernehmen (Seite 321) • Online ermittelte Geräte in das Projekt übernehmen (Seite 322)

Hinweis

Falls Sie das eigene PG bzw. den eigenen PC offline nicht projektiert haben, wird als Diagnosesymbol der grau ausgefüllte Kreis angezeigt.

Sie können auch ein zweites und drittes PG bzw. einen zweiten und dritten PC offline projektieren.

Spalten für das Ergebnis des Offline-/Online-Vergleichs

Folgende Spalten werden angezeigt:

- "Status"
  Hier wird in Form von Diagnosesymbolen das Ergebnis des Offline-/Online-Vergleichs angezeigt. Folgende Symbole sind möglich:

Diagnosesymbol	Bedeutung
	Unterschiedliche Topologieinformation in mindestens einer unterlagerten Komponente
	identische Topologieinformation
	Topologieinformation nur offline vorhanden oder Gerät ist deaktiviert
	Topologieinformation nur online vorhanden
	Unterschiedliche Topologieinformation
	Keine Port-Verschaltung konfiguriert, online keine Portverschaltung ermittelt
	Wenn ein Gerät keine Topologiefunktionen unterstützt, bleibt die Spalte "Status" leer.

- "Handlung"
  Hier werden die möglichen Handlungen angezeigt. Folgende Handlungen sind möglich:
  - keine Handlung
  - Übernehmen

Hinweis

Trennspalten

Um die Offline- und Online-Spalten der Topologievergleichstabelle optisch voneinander zu trennen, befindet sich links von der Spalte "Status" und rechts von der Spalte "Handlung" je eine Trennspalte. Die Trennspalten können nicht ausgeblendet werden, und ihe Breite kann nicht verändert werden.

Spalten für die online ermittelte Topologie

Folgende Spalten werden angezeigt:

- "PROFINET-Gerätename"
- "Artikel-Nr." (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "Gerätefamilie" (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "IP-Adresse" (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "Port"
- "Verschaltung"
  - Grüner Strich, falls der Port verschaltet ist
  - Grauer Strich, falls der Port nicht verschaltet ist
- "Partnerport"
- "Partner-IP-Adresse"
- "Partner PROFINET-Gerätename"
- "Partner-Artikel-Nr." (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "Partner-Gerätefamilie" (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)

Grundfunktionen für Tabellen

Die Topologievergleichstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden
  Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Aktuelle Darstellung speichern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen

Erweiterten Offline-/Online-Vergleich durchführen und das Ergebnis anzeigen

Überblick

Beim Standard-Offline-/Online-Vergleich wird ein Suchverfahren verwendet, welches das DCP-Protokoll benutzt.

Beim Erweiterten Offline-/Online-Vergleich wird zusätzlich ICMP benutzt, um auch Geräte zu erkennen, die kein DCP unterstützen. Zu solchen Geräten gehören z. B. PG/PC-Stationen.

Voraussetzung

- Die tabellarische Ansicht der Topologiesicht ist geöffnet.
- Sie befinden sich im Register "Topologievergleich".
- Die mittels ICMP zu erkennenden Geräte liegen im selben Subnetz wie die momentan genutzte Netzwerkkarte Ihres PG/PC.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil in der Schaltfläche "Erweiterter Vergleich".
2. Mittels der Eingabefelder "Anfangsadresse" und "Endadresse" wählen Sie einen IP-Adressbereich, in welchem mit ICMP nach erreichbaren Ethernet-Teilnehmern gesucht werden soll. Bestätigen Sie diesen z. B. durch Klicken auf die Schaltfläche mit dem grünen Häkchen.
3. Klicken Sie auf das Symbol im linken Teil der Schaltfläche "Erweiterter Vergleich".

Ergebnis

Die mittels DCP und ICMP erkennbaren Ethernet-Teilnehmer werden ermittelt.

Die Spalten auf der rechten Seite der Tabelle werden befüllt bzw. aktualisiert.

1.3.3.4 Topologie projektieren

Ports verschalten

Übersicht

Ports verschalten in der Topologiesicht

Sie haben in der Topologiesicht die folgenden Möglichkeiten, Ports zu verschalten:

- in der grafischen Ansicht (Seite 315)
- in der grafischen Ansicht eines Werkzeugwechslers (Seite 317)

- in der tabellarischen Ansicht (Seite 317)
- in der tabellarischen Ansicht eines Werkzeugwechslers (Seite 318)
- durch die Übernahme von online ermittelten Portverschaltungen (Seite 321)

Hinweis

Verschaltung eines elektrischen mit einem optischen Port

Wenn Sie einen elektrischen und einen optischen Port miteinander verschalten wolllen, müsen Sie zwischen RT- und IRT-Kommunikation unterscheiden:

- Bei RT-Kommunikation ist die Projektierung eines Medienkonverters nicht erforderlich.
- Bei IRT-Kommunikation müssen Sie die Verschaltung über einen Medienkonverter vornehmen.

Welche Auswirkungen auf die Netzsicht hat das Verschalten von Ports?

Hinweis

Sie können in der Netzsicht bei den Eigenschaften eines Subnetzes vorgeben, dass beim Anlegen einer Portverschaltung zwischen zwei nicht vernetzten Geräten dieses Subnetz verwendet wird.

Beim Anlegen einer Verschaltung zwischen zwei Ports sind folgende Auswirkungen in der Netzsicht möglich:

- Wenn die zugehörigen Schnittstellen nicht vernetzt sind: Falls Sie ein Subnetz voreingestellt haben, wird dies verwendet. Andernfalls wird ein neues Subnetz angelegt, um die beiden Schnittstellen zu verbinden.
- Wenn genau eine der beiden zugehörigen Schnittstellen vernetzt ist: Die unvernetzte Schnittstelle wird mit demselben Subnetz verbunden wie die bereits vernetzte Schnittstelle.
- In allen anderen Fällen: Die zugehörigen Schnittstellen werden nicht mit einem logischen Subnetz verbunden.

Siehe auch

Ports verschalten (Seite 3014)

Einstellungen für das Verschalten von Ethernet-Teilnehmern (Seite 94)

Ports verschalten in der grafischen Ansicht

Voraussetzung

Sie befinden sich in der grafischen Ansicht der Topologiesicht.

Vorgehen – Neue Verschaltung zwischen zwei Ports anlegen

Um einen Port eines Geräts mit einem Port eines anderen Geräts zu verschalten, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf den zu verschaltenden Port.
2. Klicken Sie mit der linken Maustaste und halten Sie diese gedrückt.
3. Verschieben Sie den Mauscursor.
   Der Mauscursor zeigt jetzt mit dem Verschaltungssymbol den Arbeitsmodus "Verschalten" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst über einer gültigen Zielposition verschwindet.
4. Ziehen Sie nun den Mauscursor auf den Ziel-Port. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.
5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste.

Ergebnis: Eine neue Portverschaltung wird angelegt.

Hinweis

Erzeugen eines Rings bei S7-300-, S7-400- und S7-1500-CPUs

Wenn Sie bei S7-300-, S7-400- oder S7-1500-CPUs mittels Portverschaltungen einen Ring erzeugen, dann wird automatisch eine MRP-Domain erzeugt.

Vorgehen – Eine bestehende Portverschaltung ändern, ohne sie vorher zu löschen

Gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf denjenigen Port einer bestehenden Verschaltung, der einen neuen Partnerport erhalten soll.
2. Ziehen Sie ihn auf den neuen Partnerport.

Ergebnis: Die bestehende Portverschaltung wird gelöscht. Die neue Portverschaltung wird angelegt.

Alternative Vorgehensweise:

1. Platzieren Sie den Mauscursor auf einen bisher unverschalteten Port, der mit einem bereits verschalteten Port verbunden werden soll.
2. Ziehen Sie ihn auf den bereits verschalteten Port.

Ergebnis: Die bestehende Portverschaltung wird gelöscht. Die neue Portverschaltung wird angelegt.

Vorgehen – Zwei bereits verschaltete Ports miteinander verschalten, ohne die beiden bestehenden Portverschaltungen vorher zu löschen

Gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf einen bereits verschalteten Port, der einen neuen Partnerport erhalten soll.
2. Ziehen Sie ihn auf den neuen Partnerport, der ebenfalls bereits verschaltet ist.

Ergebnis: Die beiden bestehenden Portverschaltungen werden gelöscht. Die neue Portverschaltung wird angelegt.

Ports verschalten in der tabellarischen Ansicht

Welche Aktionen mit Portverschaltungen sind in der tabellarischen Ansicht möglich?

In der tabellarischen Ansicht sind folgende Aktionen mit Portverschaltungen möglich:

- Neue Portverschaltung anlegen
- Bestehende Portverschaltung ändern
- Bestehende Portverschaltung löschen

Voraussetzung

Im Register "Topologieübersicht" ist die Zeile mit demjenigen Port sichtbar, dessen Verschaltung Sie neu anlegen, ändern oder löschen wollen.

Vorgehen

Um die Verschaltung eines Port erstmalig anzulegen, zu ändern oder zu löschen, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger in der Zeile des Quell-Ports in die Spalte "Partnerport".
2. Klicken Sie auf die dort befindliche Klappliste.
3. Wählen Sie den gewünschten Partnerport (bei der Neuanlage oder der Änderung einer Portverschaltung) bzw. den Eintrag "beliebiger Partner" (beim Löschen einer Portverschaltung).

Ergebnis

Die gewünschte Aktion wird durchgeführt. In der Spalte "Partnerport" wird der neue Partnerport (nach der Neuanlage oder Änderung einer Portverschaltung) bzw. der Schriftzug "Wähle Port" (nach dem Löschen einer Portverschaltung) angezeigt.

Einen Port mit mehreren Partnerports verschalten in der grafischen Ansicht

Voraussetzung

- Sie haben einen Port eines PROFINET-Geräts mit der Eigenschaft "alternative Partner" parametriert und seine möglichen Partnerports vorgegeben.
- Die grafische Ansicht der Topologiesicht ist geöffnet.

Vorgehen

1. Verschalten Sie diesen Port (im Folgenden als Quell-Port bezeichnet) mit einem der von Ihnen vorgegebenen Partnerports (im Folgenden als Ziel-Ports bezeichnet).
2. Verschalten Sie den Quell-Port mit einem weiteren Ziel-Port.
   Dies ist auf mehrere Arten möglich:
   – Ziehen Sie den Mauszeiger von einem bereits verschalteten Partnerport zu einem Ziel-Port.
   – Ziehen Sie den Mauszeiger von einer bereits angelegten Verschaltung auf einen Ziel-Port.
   – Ziehen Sie den Mauszeiger von einem Ziel-Port auf einen bereits verschalteten Partnerport.
   – Ziehen sie den Mauszeiger von einem Ziel-Port auf eine bereits angelegte Verschaltung.
3. Wiederholen Sie ggf. den vorhergehenden Schritt einmal oder mehrmals.

Ergebnis

Es wird eine Verschaltung zwischen dem Quell-Port und den wechselnden Partnerports angelegt. Diese wird in Form einer gestrichelten Linie angezeigt.

Einen Port mit mehreren Partnerports verschalten in der tabellarischen Ansicht

Welche Aktionen mit Portverschaltungen zu mehreren Partnerports sind in der tabellarischen Ansicht möglich?

Bei einem Werkzeugwechsler sind in der tabellarischen Ansicht folgende Aktionen mit Portverschaltungen zu mehreren Partnerports möglich:

- Neue Portverschaltung anlegen
- Bestehende Portverschaltung ändern
- Bestehende Portverschaltung löschen

Voraussetzung

- Sie haben einen Port eines PROFINET-Geräts mit der Eigenschaft "alternative Partner" parametriert und seine möglichen Partnerports vorgegeben.
- Im Register "Topologieübersicht" ist die Zeile mit demjenigen Port sichtbar, dessen Verschaltung Sie neu anlegen, ändern oder löschen wollen.

Vorgehen

Um die Verschaltung eines Port zu einem von mehreren Partnerports erstmalig anzulegen, zu ändern oder zu löschen, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger in der Zeile des Quell-Ports in die Spalte "Partnerport".
2. Klicken Sie auf die dort befindliche Klappliste.
3. Wählen Sie den gewünschten Partnerport (bei der Neuanlage oder der Änderung einer Portverschaltung) bzw. den Eintrag "beliebiger Partner" (beim Löschen einer Portverschaltung).

Ergebnis

Die gewünschte Aktion wird durchgeführt:

- Bei einer Neuanlage wird eine neue Zeile in die Topologieübersichtstabelle eingefügt. Dort wird in der Spalte "Partnerport" der neue Partnerport angezeigt.
- Bei einer Änderung wird in der Spalte "Partnerport" der neue Partnerport angezeigt.
- Beim Löschen wird die Zeile mit der bisherigen Portverschaltung gelöscht.

Hinweis

Bei einem Werkzeugwechsler gehören zu einem Port mit Portverschaltungen zu mehreren Partnerports in der Regel mehrere Zeilen. Dabei ist die letzte Zeile stets eine Leerzeile. Die erste Zeile ist editierbar, alle weiteren Zeilen sind nur lesbar.

Stationen, Geräte, Schnittstellen und Ports umbenennen

Eine Station, ein Gerät, eine Schnittstelle oder einen Port umbenennen

Voraussetzung

- Die tabellarische Ansicht der Topologiesicht ist geöffnet.
- Sie befinden sich im Register "Topologieübersicht".

Vorgehen

Um eine Station, ein Gerät, eine Schnittstelle oder einen Port umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie zwei Mal in das zugehörige Feld der Topologieübersichtstabelle (Mit dem zweiten Klick gelangen Sie in den Editiermodus.).
2. Geben Sie den neuen Namen ein und drücken Sie die danach ENTER-Taste (beendet den Editiermodus).

Ergebnis

Das Objekt wird umbenannt.

Offline-/Online-Vergleich

Automatische Zuordnung von Geräten durch den Offline-/Online-Vergleich

Übersicht

Beim Offline-/Online-Vergleich wird die projektierte Topologie mit der tatsächlich vorhandenen Topologie verglichen. Dabei werden online ermittelte Geräte projektierten Geräten automatisch zugeordnet, soweit dies möglich ist.

Start der Verfügbarkeitsermittlung

Sie starten die Verfügbarkeitsermittlung das erste Mal, indem Sie in der Funktionsleiste des Registers "Topologievergleich" auf die Schaltfläche "Offline-/Online-Vergleich" klicken.

Sie starten die Verfügbarkeitsermittlung erneut, indem Sie auf die Schaltfläche "Aktualisieren" klicken.

Hinweis

Die Verfügbarkeitsermittlung kann einige Sekunden dauern. In dieser Zeit sind keine Bedieneingaben möglich.

Automatische Zuordnung eines PNIO-Geräts

Ein online ermitteltes PNIO-Gerät wird einem projektierten Gerät automatisch zugeordnet, wenn die beiden Geräte in folgenden Eigenschaften übereinstimmen:

- Artikel-Nr.
- Gerätetyp bzw. Gerätefamilie
- PROFINET-Gerätename

Keine automatische Zuordnung

In den folgenden Fällen ist keine automatische Zuordnung möglich:

- Zu einem projektierten Gerät gibt es kein zugehöriges online ermitteltes Gerät (Dann sind die entsprechenden Spalten im Bereich "Online-Topologie" der Topologievergleichstabelle leer.).
  In diesem Fall sollten Sie Ihre Anlage um das bereits projektierte Gerät ergänzen oder das projektierte Gerät aus der Projektierung löschen.
- Ein online ermitteltes Gerät kann einem projektierten Gerät zugeordnet werden, jedoch gibt es Unterschiede in den Portverschaltungen.
  In diesem Fall können Sie die online ermittelten Portverschaltungen manuell in das Projekt übernehmen (Seite 321).
- Ein online ermitteltes Gerät kann keinem projektierten Gerät zugeordnet werden (Dann sind die entsprechenden Spalten im Bereich "Offline-Topologie" der Topologievergleichstabelle leer.).
  In diesem Fall können Sie das online ermittelte Gerät manuell in das Projekt übernehmen (Seite 322).

Online ermittelte Portverschaltungen manuell in das Projekt übernehmen

Voraussetzung

Sie haben in der Topologiesicht einen Offline-/Online-Vergleich durchgeführt. Dieser hatte als Ergebnis, dass mindestens ein online ermitteltes Gerät einem projektierten Gerät automatisch zugeordnet wurde und es dabei jedoch Unterschiede bei der Verschaltung gibt.

Vorgehen

Um eine oder mehrere online ermittelte Portverschaltungen manuell in das Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Markieren Sie die zur Portverschaltung gehörende Zeile.
2. Markieren Sie ggf. weitere Zeilen per Multiselektion.
3. Wählen Sie im Kontextmenü "Übernehmen" > "Ausgewählte anwenden".
   Daraufhin ändert sich der Inhalt der zugehörigen Tabellenfelder in der Spalte "Handlung" auf "Übernehmen".
4. Falls Sie fälschlicherweise zu viele Portverschaltungen für die Übernahme in das Projekt vorbereitet haben:
   Markieren Sie die Zeilen, die zu den fälschlicherweise zur Übernahme in das Projekt vorbereiteten Portveschaltungen gehören, per Multiselektion.
   Wählen Sie im Kontextmenü "Rücksetzen" > "Ausgewählte rücksetzen".
   Daraufhin ändern sich die zugehörigen Tabellenfelder in der Spalte "Handlung" auf "keine Handlung".
5. Klicken Sie auf die Schaltfläche "Synchronisieren".

Ergebnis

Für die zugehörigen Geräte werden die online ermittelten Portverschaltungen in das Projekt übernommen. Eine erfolgreiche Übernahme wird durch das Diagnosesymbol "identische Topologieinformation" für jeden Port angezeigt.

Hinweis

Wenn für ein online ermitteltes Gerät andere Portverschaltungen erkannt werden als die im Projekt vorliegenden, führt deren Übernahme in das Projekt dazu, dass die bisher im Projekt vorhandenen Portverschaltungen durch die online erkannten ersetzt werden. Wenn nun für ein online ermitteltes Gerät gar keine Portverschaltungen erkannt werden, führt die Übernahme in das Projekt dazu, dass sämtliche Portverschaltungen dieses Geräts im Projekt gelöscht werden.

Online ermittelte Geräte manuell in das Projekt übernehmen

Voraussetzung

Sie haben in der Topologiesicht einen Offline-/Online-Vergleich durchgeführt. Dieser hatte als Ergebnis, dass mindestens ein online ermitteltes Gerät keinem projektierten Gerät zugeordnet werden konnte.

Vorgehen

Um eine oder mehrere online ermittelte Geräte manuell in das Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Bewegen Sie bei einem projektierten Gerät ohne Online-Partner den Mauszeiger in die Spalte "Gerät/Port" der Online-Topologie.
2. Wählen Sie in der Klappliste dieses Felds das Gerät, das Sie dem projektierten Gerät zuordnen wollen.
3. Wiederholen Sie die bisherigen Schritte ggf. für weitere projektierte Geräte ohne Online-Partner.

Ergebnis

Das ausgewählte online ermittelte Gerät wird vom Tabellenende nach oben verschoben. Danach befindet es sich in der Zeile desjenigen projektierten Geräts, dem Sie es soeben zugeordnet haben.

Automatische Zuordnung von Geräten durch den Erweiterten Offline-/Online-Vergleich

Überblick

Beim Erweiterten Offline-/Online-Vergleich wird neben DCP zusätzlich ICMP benutzt, um auch Geräte zu erkennen, die kein DCP unterstützen.

Automatische Zuordnung von Geräten, die mittels ICMP ermittelt wurden

Bei Geräten, die mittels ICMP ermittelt wurden, ist kein Typ verfügbar.

Bei passiven Geräten ist keine Artikel-Nr. verfügbar. Daher können passive Geräte nur dann automatisch zugeordnet werden, wenn Sie in den projektierten Daten keine Artikel-Nr. vergeben haben und die Offline- und die Online-IP-Adresse übereinstimmen.

Bei Switches ist eine automatische Zuordnung möglich, wenn Offline- und Online-Artikel-Nr., -IP-Adresse und -PROFINET-Gerätename übereinstimmen.

1.3.4 Secure Communication

1.3.4.1 Grundlagen zu Secure Communication

Grundlagen zu Secure Communication

Für STEP 7 (TIA Portal) ab V14 und für S7-1500 CPUs ab Firmware V2.0 sind die Möglichkeiten zur sicheren Kommunikation, im Folgenden als "Secure Communication" bezeichnet, erheblich erweitert worden.

Mit "S7-1500 CPUs" sind auch die CPU-Varianten S7-1500F, S7-1500T, S7-1500C sowie S7-1500pro CPUs und ET200SP CPUs gemeint.

Einleitung

Das Attribut "secure" wird für die Kennzeichnung von Kommunikationsmechanismen verwendet, die auf einer Public Key Infrastructure (PKI) aufbauen (z. B. RFC 5280 für Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List Profile). Mit Public Key Infrastructure (PKI) ist ein System gemeint, das digitale Zertifikate ausstellen, verteilen und prüfen kann. Die ausgestellten digitalen Zertifikate werden innerhalb der PKI zur Absicherung von rechnergestützter Kommunikation verwendet. Wenn eine PKI ein asymmetrisches Schlüsselverfahren nutzt, dann können die Nachrichten in einem Netzwerk digital signiert und verschlüsselt werden.

Komponenten, die Sie in STEP 7 (TIA Portal) für secure Communication projektiert haben, verwenden ein asymmetrisches Schlüsselverfahren mit öffentlichem Schlüssel (Public Key) und privatem Schlüssel (Private Key). Als Verschlüsselungsprotokoll wird TLS (Transport Layer Security) eingesetzt. TLS ist Nachfolger für das Protokoll SSL (Secure Sockets Layer).

Ziele für Secure Communication

Secure Communication wird angewendet, um folgende Ziele zu erreichen:

- Vertraulichkeit
  d. h. die Daten sind für nicht autorisierten Lauscher geheim bzw. nicht lesbar.
- Integrität
  d. h. die Nachricht, die beim Empfänger eintrifft ist dieselbe, unveränderte Nachricht, die der Sender geschickt hat. Die Nachricht wurde auf dem Transportweg nicht verändert.
- Endpunkt Authentifizierung
  d. h. der Kommunikationspartner als Endpunkt ist genau derjenige, der er vorgibt zu sein und der erreicht werden soll. Die Identität des Partners ist geprüft.

Waren diese Ziele in der Vergangenheit hauptsächlich für die IT-Welt und für die vernetzten Computer von Belang, so sind heutzutage auch im industriellen Umfeld Maschinen und Steuerungen mit schützenswerten Daten durch ihre Vernetzung genauso gefährdet und stellen hohe Anforderungen an einen sicheren Datenaustausch.

Bisher und weiterhin gängig ist der Schutz der Automatisierungszelle mit Hilfe des Zellenschutzkonzepts per Firewall, oder per Verbindung über VPN, z. B. mit dem Security Modul.

Zunehmend besteht aber der Kommunikationsbedarf, auch Daten an externe Rechner in verschlüsselter Form über Intranet oder öffentliche Netze zu übertragen

Gemeinsame Prinzipien von Secure Communication

Unabhängig vom Kontext basiert Secure Communication auf dem Konzept der Public Key Infrastructure (PKI) und beinhaltet folgende Komponenten:

- Ein asymmetrisches Verschlüsselungsverfahren. Dieses Verfahren ermöglicht Folgendes:
  - Verschlüsselung oder Entschlüsselung der Nachrichten mit Hilfe von öffentlichen oder privaten Schlüsseln.
  - Überprüfung von Signaturen an Nachrichten und Zertifikaten.
    Die Nachrichten/Zertifikate werden vom Sender/Zertifikatsinhaber mit ihrem privaten Schlüssel signiert. Der Empfänger/Prüfer überprüft die Signatur mit dem öffentlichen Schlüssel des Senders/Zertifikatsinhabers.
- Transport und Speicherung der öffentlichen Schlüssel mit Hilfe von X.509-Zertifikaten:
  - X.509-Zertifikate sind digital signierte Daten, mit der die Echtheit von öffentlichen Schlüsseln in Bezug auf die gebundene Identität überprüft werden kann.
  - X.509-Zertifikate können Informationen enthalten, die die Benutzung der öffentlichen Schlüssel genauer charakterisieren bzw. einschränken. Zum Beispiel ab wann ein öffentlicher Schlüssel in einem Zertifikat gültig ist und ab wann er ungültig wird.
  - X.509-Zertifikate enthalten abgesichert die Informationen über den Herausgeber des Zertifikats.

Die folgenden Ausführungen geben einen Überblick über diese Grundkonzepte, die z.B. für den Umgang mit Zertifikaten in STEP 7 (TIA Portal) oder für die Programmierung der Kommunikationsanweisungen für secure Open User Communication (sOUC) erforderlich sind.

Secure Communication bei STEP 7

STEP 7 ab V14 stellt die jeweils notwendige PKI zur Verfügung, die für die Projektierung und den Betrieb einer Secure Communication erforderlich ist.

Beispiele:

- Das Hypertext Transfer Protokoll (HTTP) wird mit Hilfe des Protokolls TLS (Transport Layer Security) zum Hypertext Transfer Protokoll Secure (HTTPS). Weil HTTPS eine Kombination aus HTTP und TLS ist, wird es im entsprechenden RFC "HTTP over TLS" genannt. Die Verwendung von HTTPS erkennt man im Browser daran, dass in der Aufrufzeile des Browsers das URL-Schema "https://" statt "http://" verwendet wird. Die meisten Browser heben eine solchermaßen abgesicherte Verbindung zusätzlich optisch hervor.
- Open User Communication wird zur secure Open User Communication. Das zugrunde liegende Protokoll ist ebenfalls TLS.
- E-Mail Provider bieten ebenfalls Zugang über das Protokoll "Secure SMTP over TLS" an, um die Sicherheit des E-Mail-Verkehrs zu erhöhen.

Das folgende Bild zeigt das Protokoll TLS im Kontext der Kommunikationsschichten.

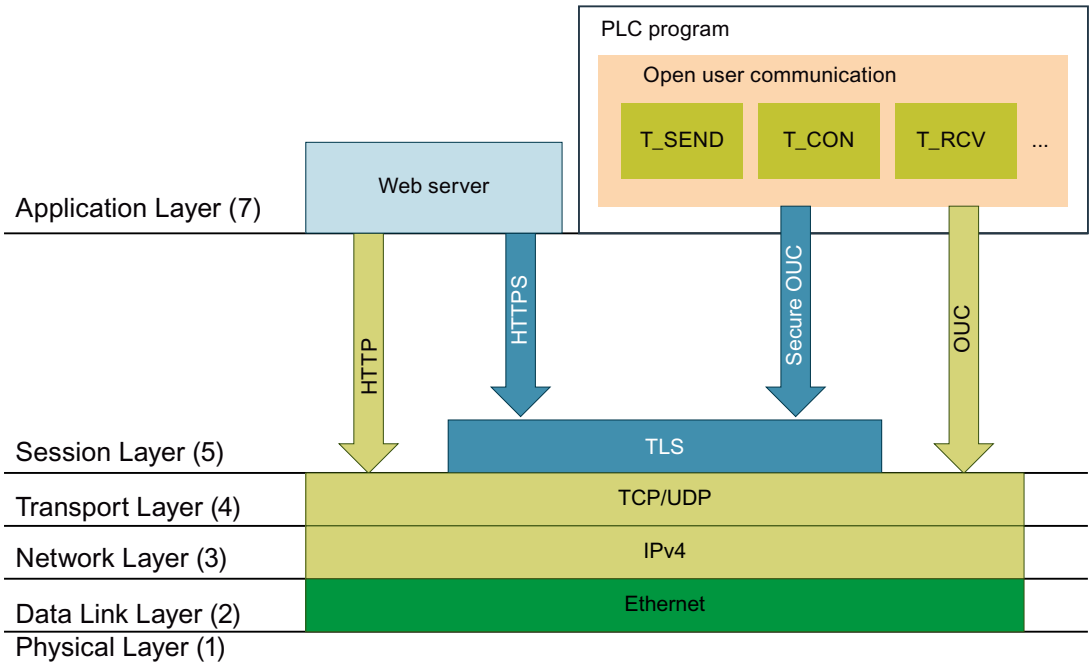

Secure Communication bei OPC UA

In den S7-1500 CPUs ist ab Firmware V2.0 ein OPC UA-Server implementiert. OPC UA Security umfasst ebenfalls Authentifizierung, Verschlüsselung und Datenintegrität über digitale X.509-Zertifikate und nutzt ebenfalls eine Public Key Infrastructure (PKI). Entsprechend den Anforderungen der Anwendung können Sie verschiedene Security-Stufen für die Endpunkt-Security wählen. Der Beschreibung der OPC UA Server-Funktionalität ist ein eigenes Kapitel gewidmet.

Siehe auch

Einstellmöglichkeiten für Security-Funktionen (Seite 1038)

Vertraulichkeit durch Verschlüsselung

Ein wichtiger Beitrag zur Datensicherheit ist die Verschlüsselung der Nachrichten. Wenn verschlüsselte Nachrichten auf dem Transportweg von einem Dritten abgefangen werden, kann dieser potentielle Lauscher nichts mit diesen Nachrichten anfangen.

Es gibt eine Vielzahl mathematischer Verfahren (Algorithmen) zur Verschlüsselung von Nachrichten.

Allen Algorithmen gemeinsam ist, dass sie einen Parameter "Schlüssel" verarbeiten, um Nachrichten zu verschlüsseln bzw. zu entschlüsseln.

- Algorithmus + Schlüssel + Nachricht => verschlüsselte Nachricht
- Verschlüsselte Nachricht + Schlüssel + Algorithmus => (entschlüsselte) Nachricht

Symmetrische Verschlüsselung

Wesentlich beim symmetrischen Verschlüsselungsverfahren ist, dass beide Kommunikationspartner zur Verschlüsselung und Entschlüsselung von Nachrichten denselben Schlüssel anwenden, wie in folgendem Bild dargestellt ist: Bob verwendet denselben Schlüssel zum Verschlüsseln wie Alice zum Entschlüsseln. Allgemein sagt man auch, dass beide Seiten als Geheimnis den geheimen Schlüssel teilen, mit dem sie eine Nachricht verschlüsseln oder entschlüsseln können.

① Bob verschlüsselt seine Nachricht mit dem symmetrischen Schlüssel

② Alice entschlüsselt die verschlüsselte Nachricht mit dem symmetrischen Schlüssel

Anschaulich ist das Verfahren mit einem Aktenkoffer vergleichbar, für den Absender und Empfänger jeweils den gleichen Schlüssel haben, um ihn verschließen bzw. öffnen zu können.

- Vorteil: Symmetrische Verschlüsselungsalgorithmen (z. B. AES, Advanced Encryption Algorithm) arbeiten schnell.
- Nachteile: Wie kommt der Schlüssel zu einem Empfänger, ohne dass er in falsche Hände gerät? Dies ist ein Schlüssel-Verteilungsproblem. Außerdem lässt sich bei einer genügend großer Anzahl abgefangener Nachrichten der Schlüssel erraten, daher muss der Schlüssel ausreichend oft neu vereinbart werden.

Bei einer Vielzahl von Kommunikationspartnern ist außerdem auch eine Vielzahl von Schlüsseln zu verteilen.

Asymmetrische Verschlüsselung

Das asymmetrische Schlüsselverfahren arbeitet mit einem Schlüsselpaar, das aus einem öffentlichen Schlüssel und einem privaten Schlüssel besteht. Im Zusammenhang mit einer PKI wird es auch Public-Key-Verfahren oder nur PKI-Verfahren genannt: Ein Kommunikationspartner, im Bild unten Alice, besitzt einen privaten Schlüssel und einen öffentlichen Schlüssel. Der öffentliche Schlüssel wird der Öffentlichkeit, also jedem potenziellen Kommunikationspartner, zur Verfügung gestellt. Jeder, der den öffentlichen Schlüssel hat, kann Nachrichten für Alice verschlüsseln. Im Bild unten ist das Bob.

Der private Schlüssel von Alice, der von ihr geheim gehalten werden muss, wird von Alice dazu benutzt, um eine an sie adressierte verschlüsselte Nachricht zu entschlüsseln.

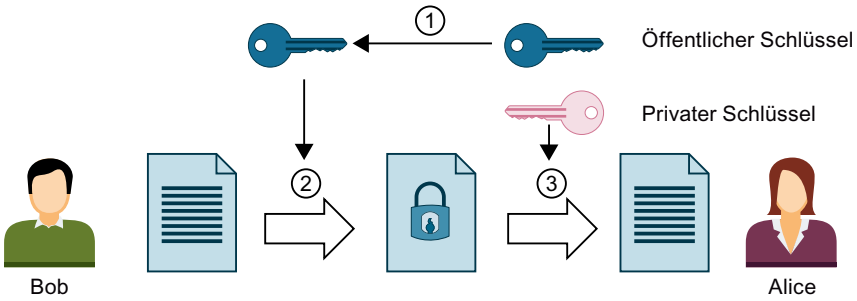

① Alice stellt Bob ihren öffentlichen Schlüssel zur Verfügung. Dazu sind keine Vorsichtsmaßnahmen erforderlich: Jeder darf den öffentlichen Schlüssel für Nachrichten an Alice nutzen, wenn er sich sicher ist, dass es tatsächlich der öffentliche Schlüssel von Alice ist.

② Bob verschlüsselt seine Nachricht mit dem öffentlichen Schlüssel von Alice.

③ Alice entschlüsselt die verschlüsselte Nachricht von Bob mit ihrem privaten Schlüssel. Da nur Alice den privaten Schlüssel besitzt und ihn niemals aus der Hand gibt, kann auch nur sie diese Nachricht entschlüsseln. Mit dem privaten Schlüssel kann sie jede Nachricht entschlüsseln, die mit ihrem öffentlichen Schlüssel verschlüsselt wurde - nicht nur die von Bob!

Anschaulich ist das Verfahren mit einem Briefkasten vergleichbar, in den jeder eine Nachricht hineinwerfen kann, aber nur derjenige, der den Schlüssel für den Briefkasten besitzt, kann die Nachricht wieder herausholen.

- Vorteile: Eine mit öffentlichem Schlüssel verschlüsselte Nachricht kann nur vom Inhaber des privaten Schlüssels entschlüsselt werden. Da ein anderer (privater) Schlüssel zum Entschlüsseln benutzt werden muss, ist es auch wesentlich schwerer, den Schlüssel zur Entschlüsselung aus der Menge der verschlüsselten Nachrichten zu erraten. Dadurch müssen die öffentlichen Schlüssel nicht streng geheim aufbewahrt werden, wie es bei symmetrischen Schlüsseln nötig ist.
  Ein weiterer Vorteil besteht in der einfacheren Verteilung von öffentlichen Schlüsseln. Beim asymmetrischen Verfahren ist kein besonders gesicherter Kanal erforderlich zur Übertragung der öffentlichen Schlüssel vom Empfänger zum Sender, der die Nachrichten verschlüsselt. Bei der Verwaltung der Schlüssel gibt es also weniger Aufwand als im symmetrischen Verschlüsselungsverfahren.
- Nachteile: Rechenintensiver Algorithmus (z. B. RSA, benannt nach den drei Mathematikern Rivest, Shamir und Adleman), daher geringere Performance im Vergleich zur symmetrischen Verschlüsselung.

Verschlüsselungsverfahren in der Praxis

In der Praxis wie z. B. beim Webserver der CPU und bei der Secure Open User Communication wird das TLS-Protokoll unterhalb der jeweiligen Anwendungsschicht genutzt. Anwendungsschichten sind z. B. HTTP oder SMTP wie im vorhergehenden Abschnitt gezeigt.

TLS (Transport Layer Security) nutzt eine Kombination aus asymmetrischer Verschlüsselung und symmetrischer Verschlüsselung (hybrides Verschlüsselungsverfahren) zur sicheren Datenübertragung z. B. im Internet und nutzt folgende Unterprotokolle:

- TLS Handshake Protocol, zuständig für Authentifizierung der Kommunikationspartner sowie Aushandeln der später für die Datenübertragung zu nutzenden Algorithmen und Schlüssel auf Basis asymmetrischer Verschlüsselungsverfahren.
- TLS Record Protocol, zuständig für Verschlüsselung der Nutzdaten mittels symmetrischer Verschlüsselungsverfahren und Datenaustausch.

Sowohl die asymmetrische als auch die symmetrische Verschlüsselung gelten als sichere Verschlüsselungsverfahren - es gibt bezüglich der Verfahren keinen prinzipiellen Unterschied hinsichtlich der Sicherheit. Der Grad der Sicherheit hängt ab von den Parametern wie z. B. von der gewählten Schlüssellänge.

Missbrauch von Verschlüsselung

Man sieht einem öffentlichen Schlüssel als eine Bitfolge nicht an, welcher Identität dieser öffentliche Schlüssel zugeordnet ist. Ein Betrüger könnte seinen öffentlichen Schlüssel zur Verfügung stellen und behaupten, er sei eine ganz andere Person. Wenn ein Dritter diesen Schlüssel im Glauben nutzt, er hätte den gewünschten Kommunikationspartner adressiert, landen möglicherweise vertrauliche Informationen bei dem Betrüger. Der Betrüger entschlüsselt dann mit seinem privaten Schlüssel die gar nicht für ihn bestimmte Nachricht und vertrauliche Informationen geraten dann in die falschen Hände.

Um solchen Missbrauch zu verhindern, muss bei den Kommunikationspartnern das Vertrauen geschaffen werden, dass sie es mit dem gewünschten Kommunikationspartner zu tun haben. Um dieses Vertrauen herzustellen, nutzt man in einer PKI digitale Zertifikate.

Authentizität und Integrität durch Signaturen

Angriffe von Programmen, welche die Kommunikation zwischen Server und Client abfangen und agieren, als wären sie selbst Client oder Server, nennt man "Man-In-The-Middle Attacks". Wenn die falsche Identität dieser Programme nicht erkannt wird, können sie z. B. wichtige Informationen über das S7-Programm erhalten oder Werte in der CPU setzen und damit eine Maschine oder Anlagen angreifen. Zur Vermeidung solcher Angriffe werden digitale Zertifikate verwendet.

Secure Communication verwendet digitale Zertifikate, die dem Standard X.509 der International Telecommunication Union (ITU) entsprechen. Damit lässt sich die Identität eines Programms, eines Rechners oder einer Organisation prüfen (authentifizieren).

Wie Zertifikate Vertrauen schaffen

Die wesentliche Aufgabe von X.509-Zertifikaten ist es, eine Identität mit den Daten eines Zertifikatsinhabers (z. B. E-Mail-Adresse, Rechnername) an den öffentlichen Schlüssel der Identität zu binden. Identitäten können Personen, Rechner oder Maschinen sein.

Zertifikate werden von Zertifizierungsstellen (Certificate Authority, CA) oder dem Inhaber des Zertifikates selber ausgegeben. PKI-Systeme legen fest, wie die Nutzer den Zertifizierungsstellen und den von ihnen herausgegebenen Zertifikaten trauen können.

Der Weg zum Zertifikat:

1. Wer ein Zertifikat haben möchte, reicht über eine bei der Zertifizierungsstelle angeschlossene Registrierungsstelle einen Zertifikatsantrag ein.
2. Die Zertifizierungsstelle bewertet Antrag und Antragsteller anhand festgelegter Kriterien.
3. Wenn sich die Identität des Antragstellers eindeutig feststellen lässt, beglaubigt die Zertifizierungsstelle diese Identität durch Ausstellen eines signierten Zertifikats. Aus dem Antragsteller ist nun der Zertifikatsinhaber geworden.

Im folgenden Bild ist der Sachverhalt vereinfacht dargestellt. Nicht gezeigt ist die Möglichkeit, wie Alice die digitale Signatur prüfen kann.

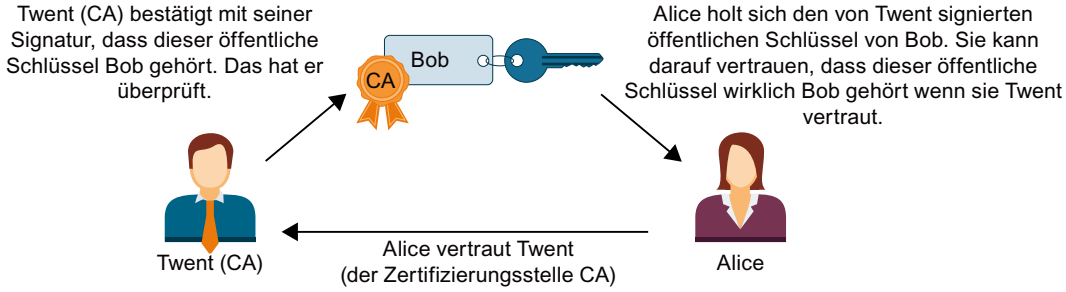

Selbstsignierte Zertifikate

Selbstsignierte Zertifikate sind Zertifikate, dessen Signatur vom Zertifikateinhaber stammt und nicht von einer unabhängigen Zertifizierungsstelle.

Beispiele:

- Sie können ein Zertifikat erstellen und selbst signieren, um zum Beispiel Nachrichten zu einem Kommunikationspartner zu verschlüsseln. Im Beispiel oben könnte Bob selber (statt Twent) sein Zertifikat mit seinem privaten Schlüssel signieren. Alice kann mit Hilfe von Bobs öffentlichem Schlüssel prüfen, dass Signatur und öffentlicher Schlüssel von Bob zusammenpassen. Für eine einfache Anlagen-interne Kommunikation, die verschlüsselt ablaufen soll, ist das ausreichend.
- Bei einem Stammzertifikat handelt es sich z. B. um ein von der Zertifizierungsstelle (Aussteller) selbstsigniertes Zertifikat, welches den öffentlichen Schlüssel der Zertifizierungsstelle enthält.

Besonderheiten von selbstsignierten Zertifikaten

Die Attribute "CN" (Common Name of Subject) für den Zertifikatsinhaber und "Issuer" (Aussteller) von selbstsignierten Zertifikaten sind identisch: Sie haben Ihr Zertifikat ja selbst signiert. Das Feld "CA" (Certificate Autority) für die Zertifizierungsstelle muss auf "False" stehen; das selbstsignierte Zertifikat soll ja nicht dazu benutzt werden, andere Zertifikate zu signieren.

Selbstsignierte Zertifikate sind nicht in eine PKI-Hierarchie eingebettet.

Inhalte von Zertifikaten

Ein Zertifikat nach dem Standard X.509 V3, der Standard, der auch von STEP 7 bzw. den S7-1500 CPUs genutzt wird, besteht im Wesentlichen aus folgenden Teilen:

- Öffentlicher Schlüssel
- Angaben über den Zertifikatsinhaber (d. h. den Schlüsselinhaber); das ist z. B. der Common Name (CN) of Subject
- Attribute wie Seriennummer und Gültigkeitsdauer
- Digitale Signatur (Beglaubigung) der Zertifizierungsstelle (CA), dass die Angaben stimmen.

Daneben gibt es Erweiterungen, z. B.

- Angabe, für welchen Verwendungszweck der öffentliche Schlüssel verwendet werden darf (Key Usage), z. B. zum Signieren oder zur Schlüssel-Verschlüsselung.
  Wenn Sie mit STEP 7 ein neues Zertifikat erstellen, z. B. im Kontext Secure Open User Communication, wählen Sie aus der Liste der möglichen Verwendungszwecke den treffenden Eintrag aus der Liste aus, z. B. "TLS".
- Angabe eines "Alternativen Namens des Zertifikatsinhabers" ("SAN", Subject Alternative Name), der z. B. bei der sicheren Kommunikation mit Webservern (HTTP over TLS) dazu genutzt wird, um sicherzustellen, dass das Zertifikat auch dem Webserver gehört, der im URL der Adresszeile des Web-Browsers angegeben wurde.

Wie Signaturen erzeugt und verifiziert werden

Die technische Voraussetzung, dass Zertifikate geprüft werden können, liefert die asymmetrische Schlüsselverwendung: Am Beispiel des Zertifikats "MyCert" werden die Prozesse "Signieren" und "Signatur prüfen" gezeigt.

Signatur erzeugen:

1. Der Aussteller des Zertifikates "MyCert" erzeugt aus den Daten des Zertifikats mit einer bestimmten Hash-Funktion (z.B. SHA-1, Secure Hash Algorithm) einen Hash-Wert.
   Der Hash-Wert ist eine Bitfolge mit konstanter Länge. Die stets gleiche Länge des Hash-Wertes bietet den Vorteil, dass das Signieren des Hash-Wertes immer die gleiche Zeit beansprucht.
2. Aus dem so erzeugten Hash-Wert erzeugt der Aussteller des Zertifikats mit Hilfe des private Schlüssels eine digitale Signatur. Häufig wird dazu das RSA-Signaturverfahren benutzt.
3. Die digitale Signatur wird im Zertifikat gespeichert. Dadurch ist das Zertifikat signiert.

Signatur verifizieren:

1. Der Prüfer des Zertifikats "MyCert" besorgt sich das Zertifikat des Ausstellers und damit den öffentlichen Schlüssel.
2. Mit demselben Hash-Algorithmus, der bei der Signierung verwendet wurde (z.B. SHA-1), wird aus den Daten des Zertifikats erneut ein Hash-Wert gebildet.

3. Dieser Hash-Wert wird dann verglichen mit dem Hash-Wert, der mit Hilfe des öffentlichen Schlüssel des Zertifikate-Ausstellers und dem Signaturalgorithmus zur Prüfung der Signatur ermittelt wird.
4. Wenn die Prüfung der Signatur ein positives Ergebnis liefert, ist sowohl die Identität des Zertifikatsinhabers als auch die Integrität, d. h. die Echtheit und Unverfälschtheit des Zertifikate-Inhalts nachgewiesen. Jeder, der den öffentlichen Schlüssel, d. h. das Zertifikat der Zertifizierungsstelle hat, kann die Signatur prüfen und so erkennen, dass das Zertifikat tatsächlich von der Zertifizierungsstelle signiert wurde.

Im folgenden Bild ist gezeigt, wie Alice mit Hilfe des öffentlichen Schlüssels des Zertifikats von Twent (verkörpert die Zertifizierungsstelle CA) die Signatur an Bobs öffentlichem Schlüssel verifiziert. Voraussetzung für die Prüfung ist also die Verfügbarkeit des Zertifikats der Zertifizierungsstelle zum Prüfungszeitpunkt. Die Validierung selbst läuft automatisch in der TLS-Session ab.

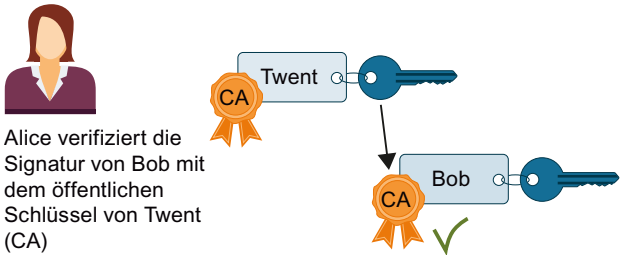

Signatur von Nachrichten

Die oben beschriebene Methode zum Signieren und Verifizieren nutzt die TLS-Session auch zum Signieren und Verifizieren von Nachrichten:

Wenn von einer Nachricht ein Hash-Wert erzeugt wird und dieser Hash-Wert mit dem privaten Schlüssel des Senders signiert und an die originale Nachricht angehängt wird, ist der Empfänger der Nachricht in der Lage, die Integrität (Unversehrtheit) der Nachricht zu erkennen. Der Empfänger entschlüsselt den Hash-Wert mit dem öffentlichen Schlüssel des Senders, bildet aus der empfangenen Nachricht selbst den Hash-Wert und vergleicht beide Werte. Wenn die Werte unterschiedlich sind, wurde die Nachricht auf dem Transportweg verfälscht.

Kette von Zertifikaten bis zum Stammzertifikat

Die Zertifikate einer PKI sind häufig hierarchisch organisiert: An der Spitze der Hierarchie stehen Stammzertifikate, auch Wurzelzertifikate oder Root-Zertifikate genannt. Das sind Zertifikate, die nicht durch eine übergeordnete Zertifizierungsstelle beglaubigt werden. Zertifikatsinhaber und Zertifikatsaussteller von Stammzertifikaten sind identisch. Stammzertifikate genießen absolutes Vertrauen, sie sind der "Anker" des Vertrauens und müssen deshalb beim Empfänger als vertrauenswürdige Zertifikate bekannt sein. Sie werden in einem für vertrauenswürdige Zertifikate vorgesehenen Bereich gespeichert.

Die Funktion von Stammzertifikaten kann je nach PKI z. B. darin bestehen, Zertifikate von untergeordneten Zertifizierungsstellen, sogenannte Zwischenzertifikate, zu signieren. Damit überträgt sich das Vertrauen vom Stammzertifikat auf das Zwischenzertifikat. Ein

Zwischenzertifikat kann genauso gut ein Zertifikat signieren wie ein Stammzertifikat, daher werden beide auch "CA-Zertifikate" genannt.

Diese Hierarchie lässt sich über mehrere Zwischenzertifikate weiterführen bis zum End-Entity Zertifikat. Das End-Entity Zertifikat ist das Zertifikat des Benutzers, der identifiziert werden soll.

Bei der Validierung wird die Hierarchie in umgekehrter Richtung durchlaufen: Wie oben beschrieben wird der Zertifikatsaussteller ermittelt, mit seinem öffentlichen Schlüssel die Signatur geprüft, dann das Zertifikat des übergeordneten Zertifikatsausstellers ermittelt bis die Vertrauenskette bis zum Stammzertifikat durchlaufen ist.

Fazit: Die Kette von Zwischenzertifikaten bis zum Stammzertifikat, der Zertifikate-Pfad, muss in jedem Gerät vorhanden sein, das ein End-Entity-Zertifikat vom Kommunikationspartner validieren soll, unabhängig davon, welche Art von Secure Communication Sie projektieren.

Verwalten von Zertifikaten mit STEP 7

STEP 7 ab Version V14 zusammen mit den S7-1500-CPUs ab FW Version 2.0 unterstützen die Internet-PKI (RFC 5280) soweit, dass eine S7-1500-CPU in der Lage ist, mit Geräten zu kommunizieren, die ebenfalls die Internet PKI unterstützen.

Die Nutzung von X.509-Zertifikaten z. B. zur Prüfung von Zertifikaten wie in den vorangegangenen Abschnitten beschrieben ist eine Konsequenz hieraus.

STEP 7 ab V14 nutzt eine PKI ähnlich der Internet PKI. Nicht unterstützt werden z. B. Certificate Revocation Lists (CRLs).

Zertifikate erstellen oder zuweisen

Für Geräte mit Security-Eigenschaften wie z. B. eine S7-1500-CPU ab Firmware V2.0 erstellen Sie in STEP 7-Zertifikate für verschiedene Anwendungen.

Folgende Bereiche im Inspektorfenster der CPU erlauben das Erstellen neuer oder das Auswählen vorhandener Zertifikate:

- "Schutz & Security > Zertifikatsmanager" - für die Erzeugung bzw. Zuweisung aller Arten von Zertifikaten; voreingestellt für das Erstellen von Zertifikaten sind TLS-Zertfikate für Secure Open User Communication.
- "Webserver > Security" - für die Erzeugung bzw. Zuweisung von Webserver-Zertifikaten.
- "OPC UA > Server > Security" - für die Erzeugung bzw. Zuweisung von OPC UA-Zertifikaten

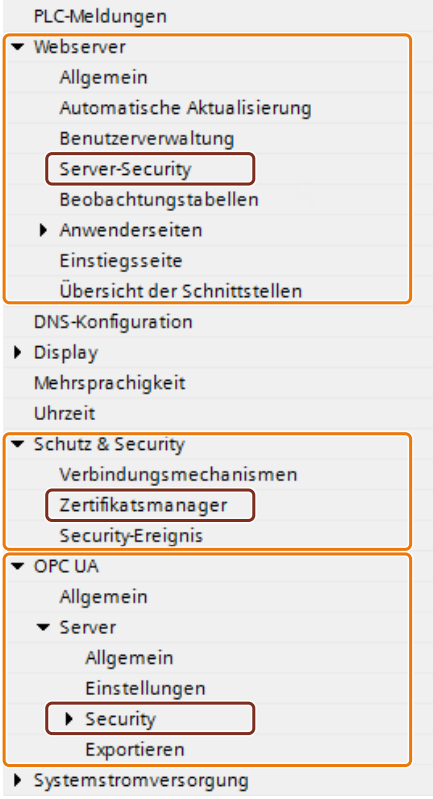

Besonderheit des Bereichs "Schutz & Security > Zertifikatsmanager"

Nur in diesem Bereich des Inspektorfensters schalten Sie zwischen dem globalen, d. h. projektweitem und lokalen, d. h. gerätespezifischem Zertifikatsmanager um (Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden"). Die Option entscheidet darüber, ob Sie Zugriff auf alle Zertifikate im Projekt haben oder nicht.

- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen **nicht** verwenden, haben Sie nur Zugriff auf den lokalen Zertifikatsspeicher der CPU. Sie haben keinen Zugriff z. B. auf importierte Zertifikate oder Stammzeritifikate. Ohne diese Zertifikate ist nur eine eingeschränkte Funktionalität verfügbar; Sie können z. B. nur selbstsignierte Zertifikate erzeugen.
- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen nicht verwenden und Sie z. B. als Administrator angemeldet sind, haben Sie Zugriff auf den globalen, projektweiten Zertifikatsspeicher. Sie können z. B. der CPU importierte Zertifikate zuweisen oder Zertifikate erstellen, die von der Projekt-CA (Zertifizierungsstelle des Projekts) ausgestellt und signiert sind.

Das folgende Bild zeigt, wie nach dem Aktivieren der Option "Globale Security-Einstellungen für den Zertifikatemanager verwenden" im Inspektorfenster der CPU die "Globalen Security-Einstellungen" in der Projektnavigation erscheinen.

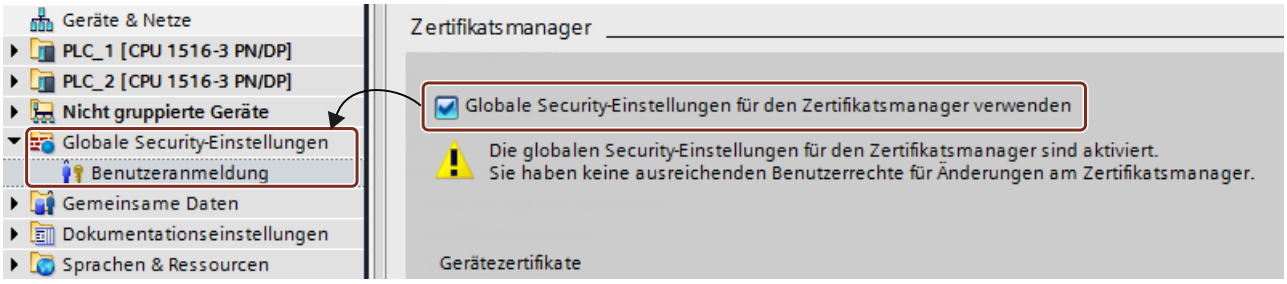

Wenn Sie in der Projektnavigation auf "Benutzeranmeldung" unterhalb der Globalen Security-Einstellungen doppelklicken und sich anmelden, erscheint dort auch unter anderem eine Zeile "Zertifikatsmanager".

Mit einem Doppelklick auf die Zeile "Zertifikatsmanager" erhalten Sie Zugang zu allen Zertifikaten im Projekt, aufgeteilt in die Register "CA" (Zertifizierungsstellen), "Gerätezertifikate" und "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen".

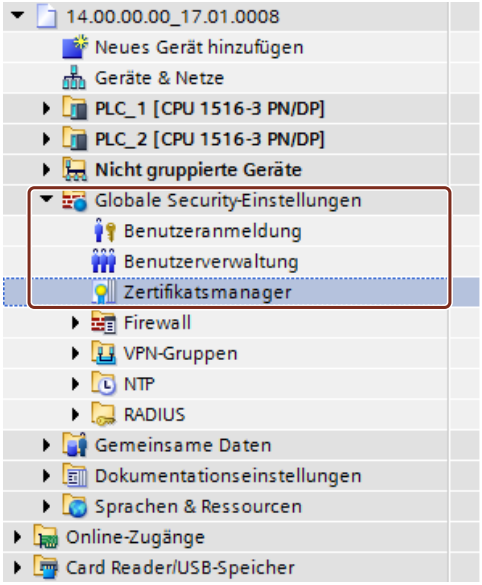

Private Schlüssel

STEP 7 erzeugt private Schlüssel beim Erzeugen von Gerätezertifikaten bzw. Server-Zertifikaten (End-Entity-Zertifikate). Wo der private Schlüssel verschlüsselt gespeichert wird, hängt von der Verwendung der globalen Security-Einstellungen für den Zertifikatsmanager ab:

- Wenn Sie die globalen Security-Einstellungen verwenden, dann wird der private Schlüssel im globalen (projektweiten) Zertifikatsspeicher verschlüsselt gespeichert.
- Wenn Sie die globalen Security-Einstellungen nicht verwenden, dann wird der private Schlüssel im lokalen (CPU-spezifischen) Zertifikatsspeicher verschlüsselt gespeichert.

Angezeigt wird das Vorhandensein des privaten Schlüssels, der z. B. für die Entschlüsselung von Daten notwendig ist, in der Spalte "Privater Schlüssel" im Register "Gerätezertifikate" des Zertifikatsmanagers in den globalen Security-Einstellungen.

Beim Laden der Hardware-Konfiguration wird das Gerätezertifikat, der öffentliche Schlüssel sowie der private Schlüssel in die CPU geladen.

ACHTUNG
Die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" beeinflusst die bisher verwendeten privaten Schlüssel: Wenn Sie bereits Zertifikate erstellt haben ohne Verwendung des Zertifikatsmanagers in den globalen Security-Einstellungen und dann die Option zur Verwendung des Zertifikatsmanagers umstellen, dann gehen die privaten Schlüssel verloren und die Zertifikats-ID kann sich ändern! Eine Warnung macht Sie auf diesen Sachverhalt aufmerksam. Legen Sie daher zu Beginn der Projektierung fest, welche Option zum Zertifikatsmanagement erforderlich ist.

Beispiele zum Verwalten von Zertifikaten

Wie in den vorangegangenen Abschnitten erläutert, sind Zertifikate für jede Art von Secure Communication erforderlich. Im Folgenden wird beispielhaft gezeigt, wie Sie mit STEP 7 die Zertifikate handhaben, damit die Voraussetzungen für Secure Open User Communication gegeben sind.

Dabei wird im Folgenden unterschieden, um welche Geräte es sich bei den beteiligten Kommunikationspartnern handelt. Die jeweiligen Schritte zum Versorgen der Kommunikationsteilnehmer mit den benötigten Zertifikaten sind jeweils beschrieben. Vorausgesetzt wird immer eine S7-1500 CPU bzw. ein S7-1500 Software Controller ab Firmware Version 2.0.

Allgemein gilt:

Während des Aufbaus einer sicheren Verbindung ("Handshake") übermitteln die Kommunikationspartner in der Regel nur ihre End-Entity-Zertifikate (Gerätezertifikate).

Daher müssen sich die zur Prüfung des übermittelten Gerätezertifikats notwendigen CA-Zertifikate im Zertifikatsspeicher des jeweiligen Kommunikationspartners befinden.

Hinweis

In der CPU muss das aktuelle Datum / die aktuelle Uhrzeit eingestellt sein.

Wenn Sie Secure Communication nutzen (z. B. HTTPS, Secure OUC, OPC UA), dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.

Secure Open User Communication zwischen zwei S7-1500 CPUs

Zwei S7-1500-CPUs PLC_1 und PLC_2 sollen miteinander Daten austauschen über Secure Open User Communication.

Die erforderlichen Gerätezertifikate erzeugen Sie mit STEP 7 und weisen Sie den CPUs zu wie im Folgenden beschrieben.

Es werden STEP 7-Projekt-Zertifizierungsstellen (CA des Projekts) verwendet, um die Gerätezertifikate zu signieren.

Die Zertifikate sind im Anwenderprogramm (Kommunikationsanweisung TCON in Verbindung mit dem zugehörigen Sytemdatentyp, z. B. TCON_IPV4_SEC) über ihre Zertifikats-ID zu referenzieren. Die Zertifikats-ID vergibt STEP 7 automatisch beim Erzeugen oder beim Anlegen von Zertifikaten.

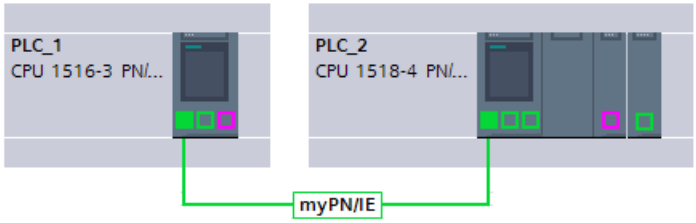

Vorgehen

STEP 7 lädt automatisch die erforderlichen CA-Zertifikate zusammen mit der Hardware-Konfiguration in die beteiligten CPUs, so dass die Voraussetzungen für die Zertifikatsprüfung für beide CPUs gegeben sind. Sie müssen also nur die Gerätezertifikate für die jeweilige CPU erzeugen - alles Übrige erledigt STEP 7 für Sie.

1. Markieren Sie PLC_1 und aktivieren Sie im Bereich "Schutz & Security" die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden".
2. Melden Sie sich in der Projektnavigation im Bereich "Globale Security-Einstellungen" als Benutzer an. Bei einem neuen Projekt ist bei der erstmaligen Anmeldung die Rolle "Administrator" vorgesehen.
3. Kehren Sie zurück zur PLC-1 in den Bereich "Schutz & Security". Klicken Sie in der Tabelle "Gerätezertifikate" in eine leere Zeile der Spalte "Zertifikatsinhaber", um ein neues Zertifikat hinzuzufügen.
4. Klicken Sie in der Klappliste zur Auswahl eines Zertifikats auf die Schaltfläche "Hinzufügen". Der Dialog "Neues Zertifikat erzeugen" wird geöffnet.
5. Belassen Sie die Voreinstellungen in diesem Dialog; sie sind auf die Verwendung für Secure Open User Communication zugeschnitten (Verwendung: TLS).
   Tipp: Ergänzen Sie den voreingestellten Namen des Zertifikatsinhabers, in diesem Fall den CPU-Namen. Belassen Sie zur besseren Unterscheidung den voreingestellten CPU-Namen für den Fall, dass Sie viele Gerätezertifikate verwalten müssen.
   Beispiel: PLC_1/TLS wird zu PLC_1-SecOUC-Chassis17FactoryState.
6. Übersetzen Sie die Konfiguration.
   Das Gerätezertifikat und das CA-Zertifikat sind Bestandteil der Konfiguration.
7. Wiederholen Sie die beschriebenenen Schritte für PLC_2.

Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Selbstsignierte Zertifikate statt CA-Zertifikate verwenden

Beim Anlegen von Gerätezertifikaten können Sie die Option "Selbstsigniert" wählen. Selbstsignierte Zertifikate können Sie erstellen, ohne für die globalen Security-Einstellungen angemeldet zu sein. Diese Vorgehensweise wird nicht empfohlen, da die so erstellten Zertifikate nicht im globalen Zertifikatsspeicher vorhanden sind und daher nicht direkt einer Partner-CPU zugewiesen werden können.

Wie oben beschrieben sollten Sie sorgfältig den Namen des Zertifikateinhabers wählen, um das richtige Zertifikat zweifelsfrei einem Gerät zuordnen zu können.

Für selbstsignierte Zertifikate ist keine Prüfung mit den CA-Zertifikaten des STEP 7-Projekts möglich. Um selbstsignierte Zertifikate prüfen zu können, müssen Sie für jede CPU das selbstsignierte Zertifikat des Kommunikationspartners in die Liste der vertrauenswürdigen Partnergeräte aufnehmen. Dazu müssen Sie die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" aktiviert haben und in den globalen Security-Einstellungen als Benutzer angemeldet sein.

Um das selbstsignierte Zertifikat vom Kommunikationspartner der CPU hinzuzufügen, gehen sie folgendermaßen vor:

1. Markieren Sie PLC_1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Schutz & Security".
2. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die Klappliste zum Hinzufügen oder Auswählen von Zertifikaten zu öffnen.
3. Wählen Sie aus der Klappliste das selbstsignierte Zertifikat des Kommunikationspartners und bestätigen Sie die Auswahl.

Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Secure Open User Communiction zwischen S7-1500 CPU als TLS-Client und Fremdgerät als TLS-Server

Zwei Geräte sollen miteinander Daten austauschen über eine TLS-Verbindung bzw. TLS-Sitzung, z. .B zum Austausch von Rezepturen, Produktionsdaten oder Qualitätsdaten:

- Eine S7-1500-CPUs (PLC_1) als TLS-Client; die CPU nutzt Secure Open User Communication
- Ein Fremdgerät (z. B. ein Manufacturing Execution System (MES) als TLS-Server

Die S7-1500 CPU baut als TLS-Client die TLS-Verbindung/Sitzung zum MES-System auf.

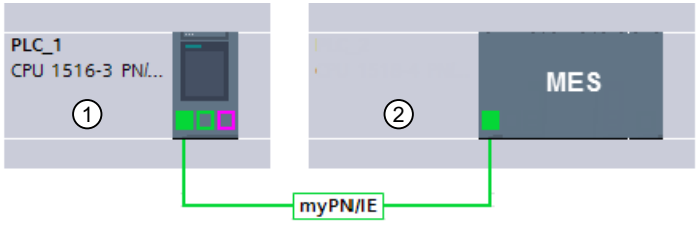

① TLS-Client
② TLS-Server

Zur Authentifizierung des TLS-Servers benötigt die S7-1500 CPU die CA-Zertifikate des MES-Systems: Das Stammzertifikat und ggf. die Zwischenzertifikate zur Prüfung des Zertifikatspfads.

Diese Zertifikate müssen Sie in den globalen Zertifikatespeicher der S7-1500 CPU importieren.

Um Zertifikate des Kommunikationspartners zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu importierende Zertifikat die passende Tabelle (Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie in der Tabelle mit Rechtsklick das Kontextmenü. Klicken Sie auf "Importieren" und importieren Sie das benötigte Zertifikat bzw. die benötigten CA-Zertifikate.
   Das Zertifikat erhält durch den Import eine Zertifikats-ID und kann im nächsten Schritt einer Baugruppe zugewiesen werden.
4. Markieren Sie PLC_1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Schutz & Security".
5. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die importierten Zertifikate hinzuzufügen.
6. Wählen Sie aus der Klappliste die benötigten CA-Zertifikate des Kommunikationspartners und bestätigen Sie die Auswahl.

Optional kann das MES-System zur Authentifizierung der CPU (d. h. des TLS-Clients) ebenfalls ein Gerätezertifikat der CPU anfordern. Dem MES-System müssen in diesem Fall die CA-Zertifikate der CPU zur Verfügung gestellt werden.Voraussetzung für den Import der Zertifikate ins MES-System ist ein vorhergehender Export der CA-Zertifikate aus dem STEP 7-Projekt der CPU. Gehen sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu exportierende Zertifikat die passende Tabelle (CA-Zertifikate).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Exportieren".
5. Wählen Sie das Exportformat des Zertifikats.

Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Secure Open User Communiction zwischen S7-1500 CPU als TLS-Server und Fremdgerät als TLS-Client

Wenn die S7-1500 CPU als TLS-Server agiert und das Fremdgerät, z. B. ein ERP-System (Enterprise Resource Planning System) die TLS-Verbindung/Sitzung aufbaut, benötigen Sie folgende Zertifikate:

- Für die S7-1500 CPU erzeugen Sie ein Gerätezertifikat (Server-Zertifikat) mit privatem Schlüssel und mit der Hardware-Konfiguration laden. Sie verwenden beim Erzeugen des Server-Zertifikats die Option "Von Zertifizierungsstelle signiert".
  Der private Schlüssel wird für den Schlüsselaustausch benötigt, wie im Bild für das Beispiel "HTTP over TLS" erläutert.
- Für das ERP-System müssen Sie das CA-Zertifikat des STEP 7-Projekts exportieren und in das ERP-System importieren/laden. Mit dem CA-Zertifikat prüft das ERP-System das Serverzertifikat der S7-1500, das von der CPU an das ERP-System während des Aufbaus der TLS-Verbindung/Sitzung übermittelt wird.

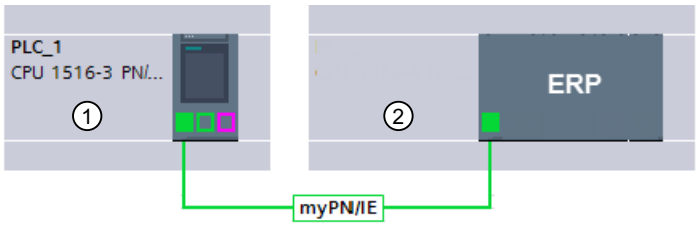

① TLS-Server
② TLS-Client

Die Beschreibung der erforderlichen Handlungsschritte entnehmen Sie dem vorangegangenen Abschnitten.

Secure Open User Communication über S7-1543-1 CP als E-Mail-Client (SMTP over TLS)

Eine S7-1500 CPU kann über den CP 1543-1 mit der Kommunikationsanweisung TMAIL-C eine sichere Verbindung zu einem E-Mail-Server aufbauen.

Die Systemdatentypen TMail_V4_SEC, TMail_V6_SEC und TMail_QDN_SEC ermöglichen Ihnen, den Partnerport des E-Mail-Servers zu bestimmen und so den E-Mail-Server über "SMTP over TLS" zu erreichen.

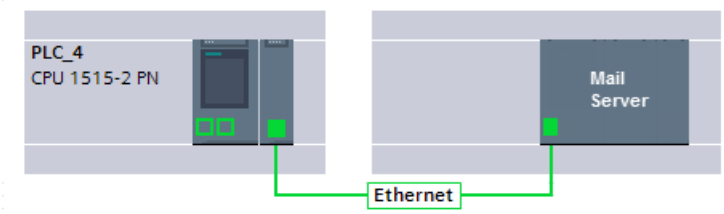

Notwendige Voraussetzung für eine sichere E-Mail-Verbindung ist ein Import des Stammzertifikats und der Zwischenzertifikate vom E-Mail-Server (Provider) in den globalen Zertifikatsspeicher des STEP 7-Projekts. Wenn diese Zertifikate dem CP zugewiesen wurden,

kann der CP mithilfe dieser Zertifikate das Serverzertifikat vom Mail-Server prüfen, das beim Aufbau der TLS-Verbindung/Sitzung vom Mail-Server gesendet wird.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu importierende Zertifikat die passende Tabelle (Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie in der Tabelle mit Rechtsklick das Kontextmenü. Klicken Sie auf "Importieren" und importieren Sie das benötigte Zertifikat bzw. die benötigten CA-Zertifikate.
   Das Zertifikat erhält durch den Import eine Zertifikats-ID und kann im nächsten Schritt dem CP zugewiesen werden.
4. Markieren Sie den CP 1543-1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Security".
5. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die importierten Zertifikate hinzuzufügen.
6. Wählen Sie aus der Klappliste die benötigten CA-Zertifikate des E-Mail-Servers und bestätigen Sie die Auswahl.

Im nächsten Schritt müssen Sie die Anwenderprogramme für die E-Mail-Client-Funktion der CPU erstellen und die Konfigurationen zusammen mit dem Programm laden.

Siehe auch

Secure OUC zwischen zwei S7-1500 CPUs (Seite 351)

Secure OUC von einer S7-1500 CPU als TLS-Client zu einem Fremd-PLC (TLS-Server) (Seite 355)

Secure OUC von einer S7-1500 CPU als TLS-Server zu einem Fremd-PLC (TLS-Client) (Seite 357)

Secure OUC über E-Mail (Seite 364)

Secure OUC über CP-Schnittstelle (Seite 360)

Beispiel: HTTP over TLS

Im Folgenden wird gezeigt, wie die beschriebenen Mechanismen genutzt werden, um eine Secure Communication zwischen einem Webbrowser und dem Webserver einer S7-1500 CPU aufzubauen.

Zunächst sind die Änderungen für die Option "Zugriff nur über HTTPS" in STEP 7 beschrieben. Ab STEP 7 V14 haben Sie die Möglichkeit, Einfluss auf das Server-Zertifikat des Webservers einer S7-1500-CPU ab Firmware V2.0 zu nehmen: Das Server-Zertifikat wird ab diesen Versionen mit STEP 7 erzeugt.

Außerdem wird gezeigt, welche Prozesse beim Aufruf einer Webseite des Webservers der CPU mit einem Webbrowser eines PCs über eine verschlüsselte HTTPS-Verbindung ablaufen.

Umgang mit Webserver-Zertifikaten für S7-1500 CPUs ab FW V2.0

Für S7-1500 CPUs mit einem Firmware-Stand kleiner V2.0 haben Sie beim Einstellen der Webserver-Eigenschaften ohne Voraussetzungen die Option "Zugriff nur über HTTPS zulassen" wählen können.

Mit der Hantierung von Zertifikaten sind Sie bei diesen CPUs nicht in Berührung gekommen; die erforderlichen Zertifikate für den Webserver erzeugt die CPU automatisch.

Bei S7-1500 CPUs ab Firmware V2.0 erzeugt STEP 7 das Server-Zertifikat (End-Entitiy-Zertifikat) für die CPU. In den Eigenschaften der CPU (Webserver > Server Security) weisen Sie dem Webserver ein Server-Zertifikat zu.

Da immer ein Server-Zertifikatsname voreingestellt ist, ändert sich an der einfachen Projektierung vom Webserver nichts: Sie aktivieren den Webserver und aktivieren die Option "Zugriff nur über HTTPS zulassen" - STEP 7 generiert beim Übersetzen ein Server-Zertifikat mit dem voreingestellten Namen.

Unabhängig davon, ob Sie den Zertifikatsmanager in den globalen Security-Einstellungen verwenden oder nicht: STEP 7 hat alle Informationen, um das Server-Zertifikat erzeugen zu können.

Zusätzlich haben Sie die Möglichkeit, die Eigenschaften des Server-Zertifikats zu bestimmen, z. B. den Namen oder die Gültigkeitsdauer.

Hinweis

In der CPU muss das aktuelle Datum / die aktuelle Uhrzeit eingestellt sein.

Wenn Sie Secure Communication nutzen (z. B. HTTPS, Secure OUC, OPC UA), dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.

Laden des Webserver-Zertifikats

Mit dem Laden der Hardware-Konfiguration in die CPU wird das von STEP 7 erzeugte Server-Zertifikat automatisch mitgeladen.

- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen verwenden, signiert die Zertifizierungsstelle des Projekts (CA-Zertifikat) das Server-Zertifikat des Webservers. Beim Laden wird das CA-Zertifikat des Projekts automatisch mitgeladen.
- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen nicht verwenden, erzeugt STEP 7 das Server-Zertifikat als selbstsigniertes Zertifikat.

Wenn Sie den Webserver der CPU über die IP-Adresse der CPU adressieren, dann ist mit jeder Änderung der IP-Adresse einer Ethernet-Schnittstelle der CPU ein neues Serverzertifikat (End-Entitiy-Zertifikat) zu erzeugen und zu laden. Der Grund ist, dass sich mit der IP-Adresse die Identität der CPU ändert - und die muss nach den Regeln der PKI beglaubigt (signiert) werden.

Sie können dieses Problem vermeiden, indem Sie die CPU über einen Domainnamen adressieren statt mit ihrer IP-Adresse, z. B. "myconveyer-cpu.room13.myfactory.com". Dazu müssen Sie die Domainnamen der CPUs über einen DNS-Server verwalten.

Webbrowser mit CA-Zertifikat des Webservers versorgen

Im Webbrowser sollte der Anwender, der per HTTPS auf die Webseiten der CPU zugreift, das CA-Zertifikat der CPU installieren. Wenn kein Zertifikat installiert ist, wird nämlich eine Warnung angezeigt, mit der Empfehlung, die Seite nicht zu benutzen. Um die Seite zu sehen, muss der Anwender dann explizit eine "Ausnahme hinzufügen".

Das gültiges Wurzelzertifikat (Certification Authority) erhält der Anwender als Download auf der Webseite "Intro" des CPU-Webservers unter "Zertifikat herunterladen".

Eine andere Möglichkeit bietet STEP 7: Exportieren Sie das CA-Zertifikat des Projekts mit dem Zertifikatsmanager in den Globalen Security-Einstellungen in STEP 7. Anschließend importieren Sie das CA-Zertifikat in den Browser.

Ablauf der Secure Communication

Das folgende Bild zeigt vereinfacht den prinzipiellen Ablauf des Aufbaus der Kommunikation ("Handshake") mit dem Schwerpunkt auf dem Aushandeln der Schlüssel, die zum Datenaustausch (hier über HTTP over TLS) verwendet werden.

Der Ablauf ist aber prinzipiell übertragbar auf alle Kommunikationsmöglichkeiten, die auf der Nutzung von TLS basieren, also auch für Secure Open User Communication (siehe Grundlagen zur Secure Communication).

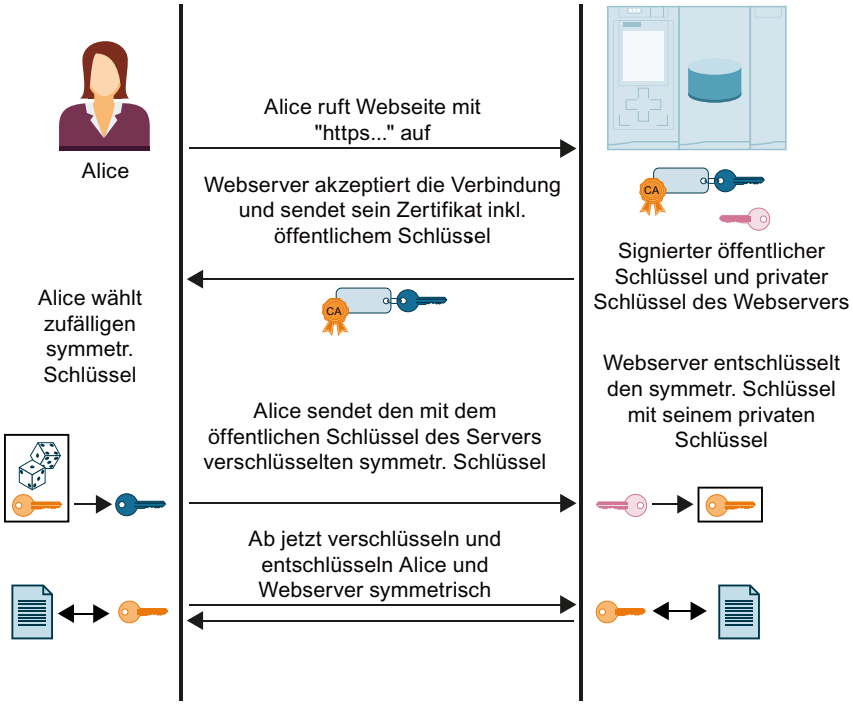

Nicht dargestellt im Bild sind die Aktionen, die auf Seiten von Alice (Browser) ablaufen, um das vom Webserver gesendete Zertifikat zu prüfen. Vom positiven Ausgang der Prüfung hängt es ab, ob Alice dem übermittelten Webserver-Zertifikat und damit der Identität des Webservers trauen kann und sich auf den Datenaustausch einlassen darf.

Die Schritte zum Prüfen der Authentizität des Webservers im Einzelnen:

1. Alice muss die öffentlichen Schlüssel aller beteiligten Zertifizierungsstellen kennen, d. h. Alice benötigt die gesamte Zertifikate-Kette zum Prüfen des Webserver-Zertifikats (d. h. des End-Entity-Zertifikats des Webservers):
   Üblicherweise hat Alice in ihrem Zertifikatespeicher das benötigte Stammzertifikat. Mit der Installation eines Webbrowsers werden z. B. eine Reihe von vertrauenswürdigen Stammzertifikaten mitinstalliert. Wenn sie das Stammzertifikat nicht hat, muss sie es von der Zertifizierungsstelle herunterladen und im Zertifikatespeicher des Browsers installieren. Die Zertifizierungsstelle kann auch das Gerät sein, auf dem sich der Webserver befindet. Um die Zwischenzertifikate zu erhalten, gibt es folgende Möglichkeiten:
   - Der Server selber sendet zusammen mit seinem End-Entity-Zertifikat die erforderlichen Zwischenzertifikate an Alice, und zwar als signierte Nachricht, damit Alice die Integrität der Zertifikate-Kette prüfen kann.
   - In den Zertifikaten befinden sich oft die URLs vom jeweiligen Zertifikate-Aussteller. Über diese URLs kann Alice die erforderlichen Zwischenzertifikate laden.

   Wenn Sie Zertifikate in STEP 7 hantieren, wird immer davon ausgegangen, dass Sie die erforderlichen Zwischenzertifikate und das Stammzertifikat in das Projekt importiert und der Baugruppe zugeordnet haben.
2. Mit den öffentlichen Schlüsseln der Zertifikate validiert Alice die Signaturen der Zertifikate-Kette.
3. Der symmetrische Schlüssel muss erzeugt und an den Webserver übermittelt werden.
4. Wenn der Webserver mit seinem Domainnamen adressiert wird, verifiziert Alice gemäß den in RFC 2818 festgelegten PKI-Regeln die Identität des Webservers: Das kann sie, weil die URL des Webservers, in diesem Fall der "Fully Qualified Domain Name" (FQDN), im End-Entity-Zertifikat des Webservers hinterlegt ist. Wenn der Zertifikatseintrag im Feld "Subject Alternative Name" mit dem Eintrag in der Adresszeile des Browsers übereinstimmt, ist alles in Ordnung.

Weiter geht es mit dem Datenaustausch mit dem symmetrischen Schlüssel, wie im Bild oben gezeigt.

Siehe auch

Zertifikate für Webserver (Seite 1045)

1.3.4.2 Globale Security-Einstellungen

Globaler Security-Zertifikatsmanager

Einleitung

Im globalen Security-Zertifikatsmanager erhalten Sie eine Übersicht aller im Projekt verwendeten Zertifikate, z. B. CA-Zertifikate mit Angaben zu Zertifikatsinhaber, Aussteller, Gültigkeit, Verwendung und Vorhandensein eines privaten Schlüssels. Mit der entsprechenden Berechtigung können Sie dort auch Zertifikate für das Projekt verwalten.

Aktivierung der globalen Security-Einstellungen

Die globalen Security-Einstellungen sind erst sichtbar, wenn diese im Projekt schon für mindestens ein Gerät aktiviert wurden. Zu diesem Zweck steht Ihnen im lokalen CPU-spezifischen Zertifikatsmanager das Kontrollkästchen "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" zur Verfügung:

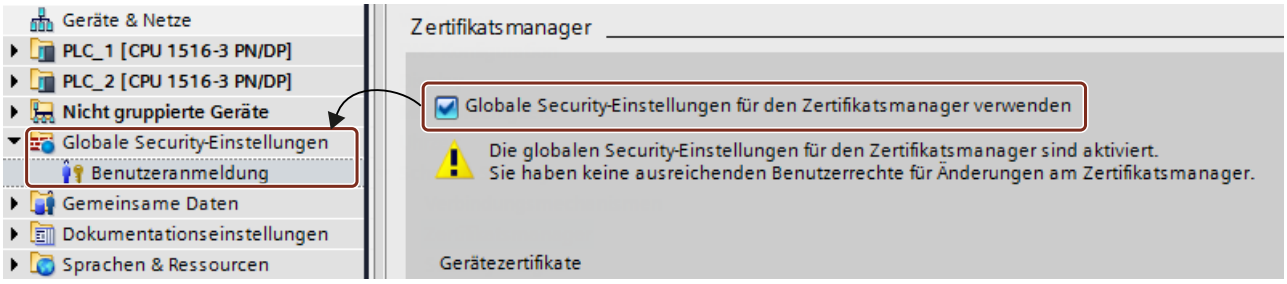

Das Kontrollkästchen finden Sie im Inspektorfenster in den allgemeinen Eigenschaften eines Gerätes mit Security-Funktionen unter "Schutz & Security > Zertifikatsmanager".

Prinzip

Das CA-Zertifikat ist ein durch eine Zertifizierungsstelle, die so genannte "Certificate Authority", ausgestelltes Zertifikat, von dem die Gerätezertifikate abgeleitet werden.

Zertifizierungsstellen können sein:

- STEP 7: Sind Zertifikatsinhaber und Aussteller gleich, dann handelt es sich um ein selbstsigniertes, also durch STEP 7 ausgestelltes Zertifikat.
- Übergeordnete Zertifizierungsstelle: Diese projektexternen Fremdzertifikate werden importiert und im Zertifikatsspeicher von STEP 7 abgelegt.

Zertifikate, die von einer der beiden Zertifizierungsstellen angelegt werden, haben immer einen privaten Schlüssel, damit die Gerätezertifikate abgeleitet werden können.

Funktionen des globalen Security-Zertifikatsmanagers

Im globalen Security-Zertifikatsmanager stehen Ihnen die folgenden Funktionen zur Auswahl:

- Importieren von neuen Zertifikaten und Zertifizierungsstellen.
- Importieren von SSL-Zertifikaten (nur CP x43-1 Adv.), z. B. für FTP-Kommunikation.
- Exportieren der im Projekt verwendeten Zertifikate und Zertifizierungsstellen.
- Erneuern von abgelaufenen Zertifikaten und Zertifizierungsstellen.
- Ersetzen bestehender Zertifizierungsstellen.
- Hinzufügen von vertrauenswürdigen Zertifikaten und Zertifizierungsstellen.

Löschen von manuell importierten Zertifikaten.

Hinweis

Laden der Projektierung

Nach dem Ersetzen oder Erneuern von Zertifikaten muss die Projektierung auf die entsprechenden Geräte geladen werden. Nach dem Ersetzen oder Erneuern von CA-Zertifikaten muss die Projektierung auf alle Geräte geladen werden.

Hinweis

Aktuelles Datum und aktuelle Uhrzeit auf den Security-Modulen

Achten Sie bei der Verwendung von gesicherter Kommunikation (z. B. HTTPS) darauf, dass die betroffenen Geräte über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die verwendeten Zertifikate werden sonst als nicht gültig ausgewertet und die gesicherte Kommunikation funktioniert nicht

Siehe auch

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Benutzer und Rollen verwalten

Einleitung

Nachdem Sie die globalen Security-Einstellungen für den Zertifikatsmanager aktiviert haben, müssen Sie sich bei den globalen Security-Einstellungen anmelden oder Sie benötigen Administrator-Rechte, sonst haben Sie keine ausreichenden Benutzerrechte für die Bearbeitung der globalen Zertifikate. Ohne Anmeldung oder wenn Sie nicht die erforderlichen Rechte haben, können Sie nicht auf den globalen Zertifikatsmanager zugreifen.

Rollen und Rechte

Die Security-Einstellungen können Sie konfigurieren, wenn Ihnen die vordefinierten Rollen "Administrator" oder "Standard" zugewiesen sind oder Ihnen eine andere definierte Rolle mit der entsprechenden Berechtigung zugewiesen ist. Sie können die Rollen in der Projektnavigation unter "Globale Security-Einstellungen > Benutzerverwaltung > Rollen" ansehen bzw. bei entsprechender Berechtigung bearbeiten.

Zugewiesene Rolle	Diagnostizieren	Konfigurieren	Benutzer und Rollen verwalten
Administrator	Ja	Ja	Ja
Standard	Ja	Ja	Nein
Diagnose	Ja	Nein	Nein

Sie können in der Benutzerverwaltung als Administrator neue Rollen und Benutzer anlegen und Benutzer und Rollen einander zuweisen.

Voraussetzung

Sie müssen bei den globalen Security-Einstellungen über die Benutzeranmeldung als Administrator angemeldet sein.

Vorgehen

Um die Benutzer und Rollen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Gehen Sie in der Projektnavigation in den globalen Security-Einstellungen in die Benutzerverwaltung.
2. Wählen Sie die zu ändernde Rolle aus oder legen Sie eine neue Rolle an.
3. Bearbeiten Sie in der Rechteliste die Rechte der ausgewählten Rolle.

Zertifikate importieren und exportieren

Einleitung

Sie können Zertifikate und die dazu passenden privaten Schlüssel importieren und exportieren. Zertifikate können nur über den globalen Zertifikatsmanager importiert werden. Der Export von Zertifikaten kann auch über den lokalen CPU-spezifischen Zertifikatsmanager erfolgen.

Sie können die Zertifikate in die folgenden Dateitypen importieren oder exportieren:

- Zertifikate ohne privaten Schlüssel nach CER, DER, CRT oder PEM
- Zertifikate mit ihrem privaten Schlüssel nach P12 (PKCS12).

Voraussetzung

Beim Import von Zertifikaten mit privatem Schlüssel benötigen Sie ggf. ein Passwort für den Zugriff.

Zertifikate importieren

Um ein Zertifikat zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu importierende Zertifikat die passende Tabelle (CA-Zertifikate, Gerätezertifikate, Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie in der Tabelle mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Importieren".
5. Wählen Sie das Importformat des Zertifikats:
   - CER, DER, CRT oder PEM für Zertifikate ohne privaten Schlüssel.
   - P12 (PKCS12-Archiv) für Zertifikate mit privatem Schlüssel.
6. Klicken Sie auf "Öffnen" um das Zertifikat zu importieren.

Das importierte Zertifikat muss dem gewünschten Gerät oder Modul manuell zugewiesen werden.

Zertifikate exportieren

Um ein Zertifikat zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu exportierende Zertifikat die passende Tabelle (CA-Zertifikate, Gerätezertifikate, Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Exportieren".
5. Wählen Sie das Exportformat des Zertifikats:
   - CER, DER, CRT oder PEM für Zertifikate ohne privaten Schlüssel.
   - CRL für Zertifikatsperrliste nach DER kodiert
   - P12 (PKCS12-Archiv) für Zertifikate mit privatem Schlüssel.
6. Klicken Sie auf "Speichern" um das Zertifikat zu exportieren.

Sie können Zertifikate auch direkt aus dem lokalen CPU-spezifischen Zertifikatsmanager exportieren.

Auswählbare Dateiformate

X.509-Zertifikate nach dem Zertifikats-Standard X.509 V3 aus der RFC 5280 können in verschiedenen Formaten mit unterschiedlichen Dateinamenserweiterungen importiert oder exportiert werden:

Dateinamener-weiterung	PEM
Dateityp	Privacy Enhanced Mail Security Certificate
Verwendung	Base64-kodiertes Zertifikat
MIME-Type	application/x-pem-file
Bedeutung	Zertifikat im PEM/X509-Format (PEM = Privacy-Enhanced Mail) mit einer Kodierung in ASCII (Base64). Häufig benutztes Format der Zertifizierungsstellen für die Zertifikatsübertragung per E-Mail an den Antragssteller des Zertifikats. Anstelle der Dateinamenerweiterung PEM kann ein Base64-kodiertes Zertifikat unter anderem auch die Erweiterungen CER oder CRT haben.
Verwendung	Im TIA Portal wird der Import und Export mit PEM-Kodierung unterstützt. Ein Export mit PEM-Kodierung kann im Dateiformat PEM oder CRT erfolgen.

Dateinamener-weiterung	DER
Dateityp	DER Encoded X509 Certificate
Verwendung	binär kodiertes Zertifikat
MIME-Type	application/x-x509-ca-cert, application/pkix-cert

Dateinamenerweiterung	DER
Bedeutung	Zertifikat im DER/X.509-Format (DER = Distinguished Encoding Rules) mit einer binären Kodierung. Das binäre DER-Format ist die Grundlage für das ASCII-kodierte PEM-Format. Das DER-Format wird vor allem im Microsoft Windows-Umfeld und auf Java-Plattformen verwendet. Anstelle der Dateinamenerweiterung DER kann ein binär-kodiertes Zertifikat unter anderem auch die Erweiterungen CER oder CRT haben.
Verwendung	Im TIA Portal wird der Import und Export mit DER-Kodierung unterstützt. Ein Export mit DER-Kodierung kann im Dateiformat DER oder CER erfolgen.

Dateinamenerweiterung	CER
Dateityp	Internet Security Certificate File
Verwendung	nach PEM oder DER kodiertes Zertifikat
MIME-Type	application/x-x509-ca-cert, application/pkix-cert, application/keychain_access
Bedeutung	Alternative Dateinamenerweiterung CER (Canonical Encoding Rules) für ein binär kodiertes Zertifikat im DER-Format oder für ein Base64/ASCII-kodiertes Zertifikat im PEM-Format.
Verwendung	Im TIA Portal wird der Import und Export im Dateiformat CER unterstützt. Beim Import eines Zertifikats im Dateiformat CER kann es sich um eine Kodierung nach PEM oder DER handeln. Für den Export im Dateiformat CER wird die DER-Kodierung verwendet.

Dateinamenerweiterung	CRT
Dateityp	Internet Security Certificate File
Verwendung	nach PEM oder DER kodiertes Zertifikat
MIME-Type	application/x-x509-ca-cert, application/pkix-cert, application/keychain_access
Bedeutung	Alternative Dateinamenerweiterung CRT für ein binär kodiertes Zertifikat im DER-Format oder für ein Base64/ASCII-kodiertes Zertifikat im PEM-Format.
Verwendung	Im TIA Portal wird der Import und Export im Dateiformat CRT unterstützt. Beim Import eines Zertifikats im Dateiformat CRT kann es sich um eine Kodierung nach PEM oder DER handeln. Für den Export im Dateiformat CRT wird die PEM-Kodierung verwendet.

Dateinamenerweiterung	CRL
Dateityp	Certificate Revocation List
Verwendung	nach DER kodierte Zertifikatsperrliste
MIME-Type	application/pkix-crl, application/x-pkcs7-crl

Dateinamenerweiterung	CRL
Bedeutung	Die CRL (Certificate Revocation List) ist eine X.509 V3-Erweiterung und enthält die zu einem bestimmten Stammzertifikat zugehörige Liste gesperrter oder widerrufener Zertifikate. Zertifizierungsstellen verwenden die Möglichkeit, um die Autorisierung bereits ausgegebener Zertifikate zurückzunehmen. Dies ist der Fall, wenn Zertifikate kompromittiert sind oder sich deren Inhalt als fehlerhaft herausstellt. Gesperrte Zertifikate können über die CRL wieder entsperrt werden, zum Beispiel wenn sich das Zertifikat als doch nicht kompromittiert herausstellt oder der Inhalt doch korrekt war. Widerrufene Zertifikate bleiben dauerhaft ungültig können nicht erneut autorisiert werden. Die Zertifizierungsstelle aktualisiert ihre CRLs, sobald deren Gültigkeit abgelaufen ist oder sich der Inhalt ändert. Die Bezugsquelle einer CRL ist im jeweiligen Zertifikat eingetragen.
Verwendung	Das TIA Portal unterstützt keinen Zertifikatsrückruf über CRL. Da einige externe Clients zwingend Zertifikatssperrlisten für die Validierung der Zertifikate benötigen, wird jedoch ein Export der CRL unterstützt. Die Liste ist jedoch leer und erfüllt lediglich den Zweck, die Validierung eines ausgegebenen Zertifikats durch bestimmte externe Clients zu ermöglichen.

Dateinamenerweiterung	P12
Dateityp	Personal Information Exchange File
Verwendung	als PKCS12-Archiv kodierte Zertifikate, optional mit privatem Schlüssel
MIME-Type	application/keychain_access, application/x-pkcs12
Bedeutung	Dateitypen im Format P12 können neben dem Server-Zertifikat und Zwischenzertifikaten auch noch den privaten Schlüssel enthalten. Beim Speichern kann ein Passwort vergeben werden.
Verwendung	Im TIA Portal wird der Import und Export im Dateiformat P12 unterstützt. Die Option zum Export in ein PKCS12-Archiv besteht nur, wenn das Zertifikat über einen privaten Schlüssel verfügt. Beim Export kann ein Passwort für das PKCS12-Archiv vergeben werden. Analog zur Passwortvergabe beim Export kann beim Import der Zugriff auf das Zertifikat die Eingabe eines Passworts notwendig machen.

Zertifikate erneuern

Einleitung

Sie können Zertifikate erneuern, zum Beispiel um kompromittierte Zertifikate durch ein importiertes oder neues Zertifikat zu aktualisieren.

Vorgehen

Um ein Zertifikat zu erneuern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu erneuernde Zertifikat die passende Tabelle (CA-Zertifikate, Gerätezertifikate).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Erneuern".

5. Im Dialog für neue Zertifikate können Sie die Daten des ausgewählten Zertifikates bearbeiten.
6. Wenn Sie auf "OK" klicken, wird das Zertifikat mit den geänderten Werten erneuert.

Wenn Sie ein Zertifikat erneuern, wird das alte Zertifikat durch das Zertifikat mit den neuen Werten ersetzt. Durch die Erneuerung von Zertifikaten wird keine Signaturanfrage an eine Zertifizierungsstelle generiert.

Zertifikate ersetzen

Einleitung

Sie können ein bestehendes Zertifikat durch ein anderes Zertifikat ersetzen.

Vorgehen

Um ein Zertifikat zu ersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu ersetzende Zertifikat die passende Tabelle (CA-Zertifikate, Gerätezertifikate, Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Ersetzen".
5. Im Dialog für die Änderung der Zertifizierungsstelle ersetzen Sie das bestehende CA-Zertifikat des Projekts oder das CA-Gruppenzertifikat durch ein neues.

Alle in der Tabelle "Betroffene Zertifikate" aufgelisteten Zertifikate werden neu abgeleitet.

Zertifikate anzeigen

Einleitung

Sie können sich eine Übersicht aller Zertifikatsdaten über den Zertifikatsdialog von Windows anzeigen lassen.

Vorgehen

Um Zertifikatsdaten anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das anzuzeigende Zertifikat die passende Tabelle (CA-Zertifikate, Gerätezertifikate, Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Anzeigen".

Ergebnis

In einem Fenster werden Ihnen verschiedene Informationen zu dem ausgewählten Zertifikat angezeigt:

- Allgemeine Informationen zu dem Zertifikat
- Detailinformationen zu dem Zertifikat
- Informationen zu dem Zertifizierungspfad

Zertifikate löschen

Einleitung

Sie können nicht mehr benötigte oder kompromitierte Zertifikate löschen.

Vorgehen

Um ein Zertifikat zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu löschende Zertifikat die passende Tabelle (Gerätezertifikate, Vertrauenswürdige Zertifikate von Stammzertifizierungsstellen).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Löschen".

Hinweis

Dem Projekt vorbehaltene CA-Zertifikate und Zertifikate, die automatisch einem CP zugewiesen sind, können nicht gelöscht werden.

Sie können Zertifikate auch direkt im lokalen CPU-spezifischen Zertifikatsmanager löschen. Das Zertifikat bleibt dann für das Projekt im globalen Security-Zertifikatsmanager erhalten, wird aber nicht mehr für das Gerät verwendet.

1.3.4.3 Secure Open User Communication

Secure OUC zwischen zwei S7-1500 CPUs

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP zwischen zwei S7-1500 CPUs einrichten. Dabei agiert eine S7-1500 CPU als TLS-Client (aktiver Verbindungsaufbau) und die andere S7-1500 CPU als TLS-Server (passiver Verbindungsaufbau).

Gesicherte TCP-Verbindung zwischen zwei S7-1500 CPUs einrichten

Für die gesicherte TCP-Kommunikation zwischen zwei S7-1500 CPUs müssen Sie in jeder CPU einen Datenbaustein mit dem Systemdatentyp TCON_IPv4_SEC erstellen, parametrieren und direkt an der Anweisung aufrufen. Die Anweisung TCON unterstützt die Systemdatentypen TCON_QDN_SEC und TCON_IP_V4_SEC. Ab Firmware-Version V2.5 unterstützen auch die Anweisungen TSEND_C und TRCV_C die Systemdatentypen TCON_QDN_SEC und TCON_IP_V4_SEC.

Voraussetzungen:

- Beide S7-1500 CPU haben mindestens Firmwarestand V2.0
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

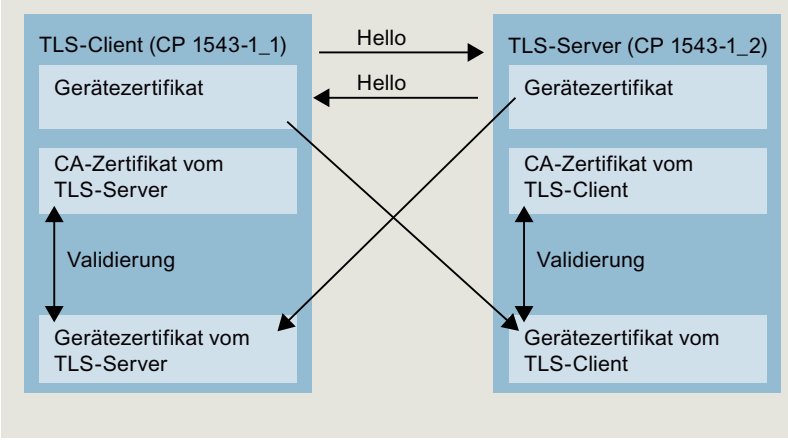

Einstellungen am TLS-Client

Um eine gesicherte TCP-Verbindung im TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC . Geben Sie dazu im Feld "Datentyp" die Zeichenfolge "TCON_IP_V4_SEC" ein.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Client" vom Datentyp TCON_IP_V4_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Client	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	72	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	100	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against th
19	TLSServerCertRef	UDInt	1	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	5	for Client side: add id of own X.509 V3 client certificate; for Serv

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAdress" die IPv4-Addresse des TLS-Servers ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine ungesicherte TCP- oder UDP-Verbindung einrichten.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.
   - "TLSClientCertRef": Tragen Sie z. B. den Wert 1 ein. Damit referenzieren Sie das CA-Zertifikat des TIA Portal-Projekts (SHA1).
5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Einstellungen am TLS-Server

Um eine gesicherte TCP-Verbindung im TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Server" vom Datentyp TCON_IP_V4_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Server	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	120	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	10	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	true	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against the
19	TLSServerCertRef	UDInt	6	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	1	for Client side: add id of own X.509 V3 client certificate; for Serv

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAdress" die IPv4-Addresse des TLS-Clients ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von secure communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine ungesicherte TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert ": Anforderung eines X.509-V3-Zertifikats vom TLS-Client. Tragen Sie den Wert "true" ein.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.
   - "TLSClientCertRef": Tragen Sie z. B. den Wert 1 ein. Damit referenzieren Sie das CA-Zertifikat des TIA Portal-Projekts (SHA1).
5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Siehe auch

Verbindungsparameter nach TCON_IP_V4_SEC (Seite 119)

Verbindungsparameter nach TCON_QDN_SEC (Seite 122)

Beispiele zum Verwalten von Zertifikaten (Seite 335)

Secure OUC von einer S7-1500 CPU als TLS-Client zu einem Fremd-PLC (TLS-Server)

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP von einer S7-1500 CPU als TLS-Client zu einem TLS-Server einrichten.

Gesicherte TCP-Verbindung von einer S7-1500 CPU als TLS-Client zu einem TLS-Server einrichten

S7-1500 CPUs ab Firmwarestand V2.0 unterstützen Secure Communication mit Adressierung über ein Domain Name System (DNS).

Für die gesicherte TCP-Kommunikation über den Domainnamen müssen Sie selbst einen Datenbaustein mit dem Systemdatentyp TCON_QDN_SEC erstellen, parametrieren und direkt an der Anweisung aufrufen. Die Anweisung TCON unterstützt den Systemdatentyp TCON_QDN_SEC.

Voraussetzungen:

- In Ihrem Netz befindet sich mindestens ein DNS-Server.
- Sie haben für die S7-1500 CPU mindestens einen DNS-Server konfiguriert.
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

Um eine gesicherte TCP-Verbindung zu einem TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_QDN_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "DNS ConnectionSEC" vom Datentyp TCON_QDN_SEC definiert ist.

Data_block_1

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	DNS Connection SEC	TCON_QDN_SEC		
3	ConnPara	TCON_QDN		parameter of the TCP connection
4	InterfaceId	HW_ANY	0	not relevant
5	ID	CONN_OUC	5	connection reference / identifier
6	ConnectionType	Byte	11	type of connection: 16#0B=11=TCP/IP, 16#13
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteQDN	String[254]	'plc_1.factory127.'	fully or partially qualified domain name of rem
9	RemotePort	UInt	4000	remote UDP / TCP port number
10	LocalPort	UInt	0	local UDP / TCP port number
11	ActivateSecureConn	Bool	true	activate the security functionality of that conn
12	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a
13	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 a
14	TLSServerCertRef	UDInt	7	for Server side: Reference to own X.509 V3 se
15	TLSClientCertRef	UDInt	0	for Client side: add id of own X.509 V3 client c

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteQDN" den vollqualifizierten Domainnamen (FQDN) des TLS-Servers ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von secure communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "ExtTLSCapabilities": Wenn Sie den Wert 1 eintragen, dann validiert der Client den subjectAlternateName im X.509-V3-Zertifikat des Servers, um die Identität des Servers zu überprüfen. Die Zertifikate werden beim Verbindungsaufbau überprüft.
   - "TLSServerCertRef": ID des X.509-V3-Zertifikats (gewöhnlich ein CA-Zertifikat), das vom TLS-Client benutzt wird, um die Authentifizierung des TLS-Servers zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Client zur Validierung der Server-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Clients geladen sind.die Zertifikat-ID vom CA-Zertifikat des TLS-Servers ein.

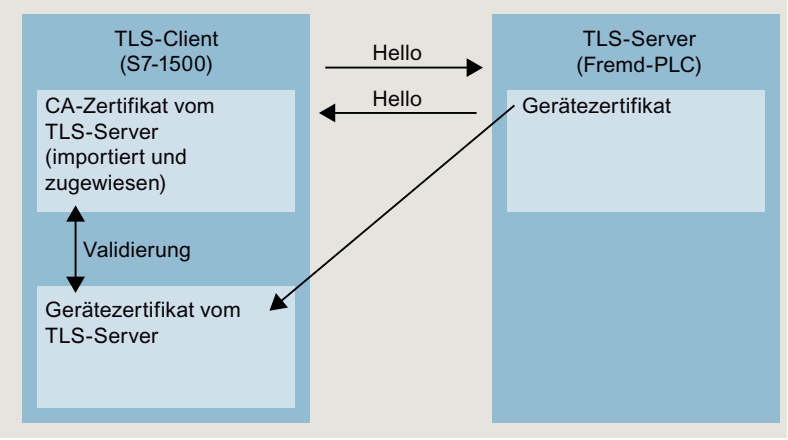

   - "TLSClientCertRef": ID des eigenen X.509-V3-Zertifikats.

5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_QDN_SEC.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "DNS connectionSEC" (Datentyp TCON_QDN_SEC) verschaltet.

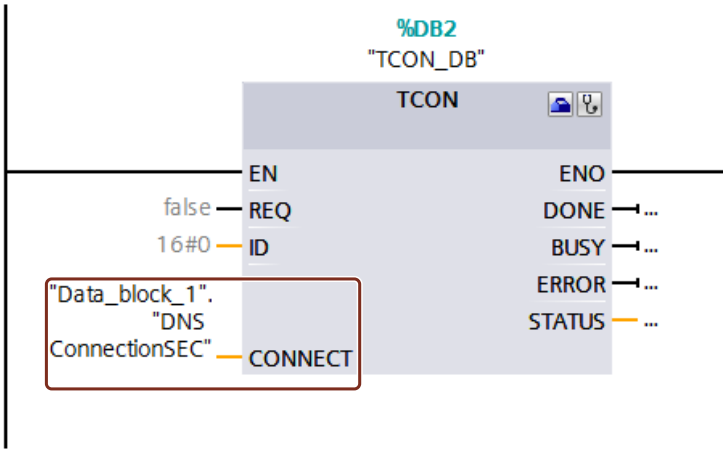

Siehe auch

Verbindungsparameter nach TCON_QDN_SEC (Seite 122)

Beispiele zum Verwalten von Zertifikaten (Seite 335)

Secure OUC von einer S7-1500 CPU als TLS-Server zu einem Fremd-PLC (TLS-Client)

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP von einer S7-1500 CPU als TLS-Server zu einem TLS-Client einrichten.

Gesicherte TCP-Verbindung über den Domainnamen des Kommunikationspartners einrichten

S7-1500 CPUs ab Firmwarestand V2.0 unterstützen Secure Communication mit Adressierung über ein Domain Name System (DNS).

Für die gesicherte TCP-Kommunikation über den Domainnamen müssen Sie selbst einen Datenbaustein mit dem Systemdatentyp TCON_QDN_SEC erstellen, parametrieren und direkt an der Anweisung aufrufen. Die Anweisung TCON unterstützt den Systemdatentyp TCON_QDN_SEC.

Voraussetzungen:

- In Ihrem Netz befindet sich mindestens ein DNS-Server.
- Sie haben für die S7-1500 CPU mindestens einen DNS-Server konfiguriert.
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

Um eine gesicherte TCP-Verbindung zu einem TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_QDN_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "DNS ConnectionSEC" vom Datentyp TCON_QDN_SEC definiert ist.

Data_block_1

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	DNS Connection SEC2	TCON_QDN_SEC		
3	ConnPara	TCON_QDN		parameter of the TCP connection
4	InterfaceId	HW_ANY	0	not relevant
5	ID	CONN_OUC	8	connection reference / identifier
6	ConnectionType	Byte	11	type of connection: 16#0B=11=TCP/IP, 16#13
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteQDN	String[254]	''	fully or partially qualified domain name of rem
9	RemotePort	UInt	0	remote UDP / TCP port number
10	LocalPort	UInt	2010	local UDP / TCP port number
11	ActivateSecureConn	Bool	true	activate the security functionality of that conn
12	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a
13	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 a
14	TLSServerCertRef	UDInt	5	for Server side: Reference to own X.509 V3 se
15	TLSClientCertRef	UDInt	0	for Client side: add id of own X.509 V3 client c

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "ID" die lokale ID der TCP-Verbindung ein.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von secure communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert": Anforderung eines X.509-V3-Zertifikats vom TLS-Client.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.

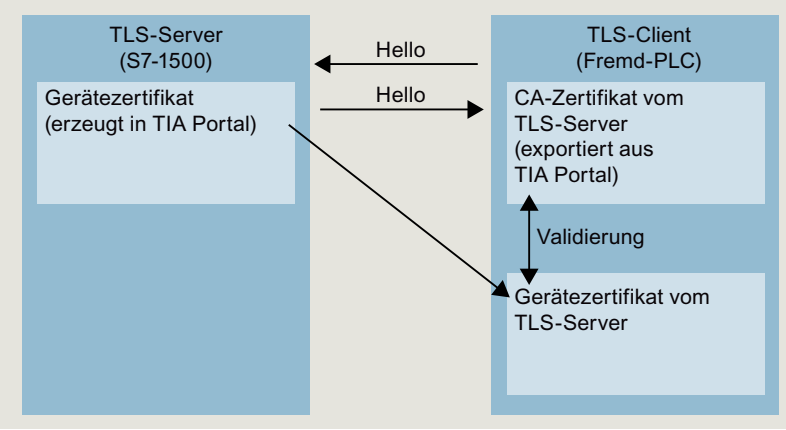

   - "TLSClientCertRef": ID des X.509-V3-Zertifikats (oder einer Gruppe von X.509-V3-Zertifikaten), das vom TLS-Server benutzt wird, um die Authentifizierung des TLS-Clients zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Server zur Validierung der Client-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Servers geladen sind.
5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_QDN_SEC.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "DNS connectionSEC" (Datentyp TCON_QDN_SEC) verschaltet.

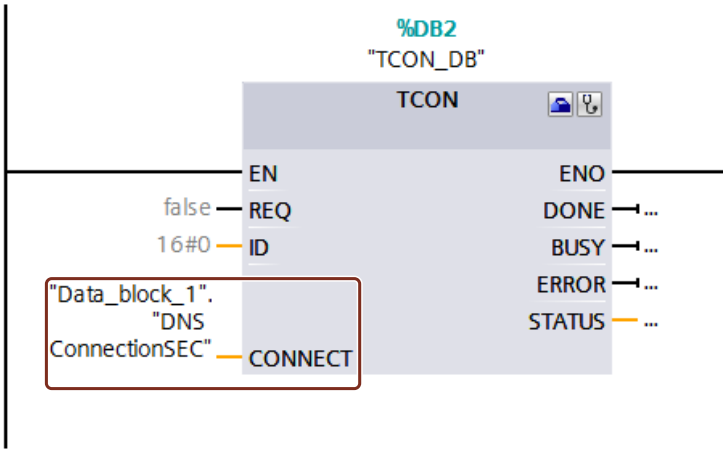

Siehe auch

Verbindungsparameter nach TCON_QDN_SEC (Seite 122)

Beispiele zum Verwalten von Zertifikaten (Seite 335)

Secure OUC über CP-Schnittstelle

Im Folgenden sind die Besonderheiten beschrieben, die bei Secure Open User Communication über eine CP-Schnittstellen zu berücksichtigen sind. Mindestens eine Station ist eine S7-1500 Station mit folgenden Baugruppen:

- S7-1500 CPU ab Firmwarestand V2.0 (ausgenommen S7-1500 Software Controller)
- CP 1543-1 ab Firmwarestand V2.0 bzw. CP 1543SP-1 ab Firmwarestand V1.0

Der CP agiert in einer S7-1500 Station als TLS-Client (aktiver Verbindungsaufbau) oder als TLS-Server (passiver Verbindungsaufbau).

Die grundsätzliche Vorgehensweise und das Konzept für die Nutzung von Secure Communication über eine CP-Schnittstelle ist ähnlich wie Secure Communication über die Schnittstellen der S7-1500 CPUs. Im Wesentlichen müssen Sie dem CP in der Rolle als TLS-Server oder TLS-Client die Zertifikate zuordnen und nicht der CPU. Daher gelten andere Regeln und Vorgehensweisen, die im Folgenden beschrieben sind.

Hantierung von Zertifikaten für CPs

Generell gilt: Sie müssen beim Zertifikatsmanager in den Globalen Security-Einstellungen angemeldet sein. Auch das Erstellen von selbst signierten Zertifikaten ist nicht ohne Anmeldung für die Globalen Security-Einstellungen möglich! Sie müssen als Benutzer mit ausreichenden Rechten ausgestattet sein (Administrator oder Benutzer mit der Rolle "Standard" mit dem Recht "Security konfigurieren").

Ausgangspunkt für die Erstellung oder Zuweisung von Zertifikaten beim CP ist der Bereich "Security > Security-Eigenschaften". In diesem Bereich melden Sie sich für die Globalen Security-Einstellungen an.

Vorgehensweise:

1. Markieren Sie in der Netzsicht von STEP 7 den CP und wählen im Inspektorfenster den Bereich "Security > Security-Eigenschaften".
2. Klicken Sie auf die Schaltfläche "Benutzeranmeldung".
3. Melden Sie sich mit Benutzernamen und Passwort an.
4. Aktivieren Sie die Option "Aktiviere Security-Funktionen".
   Die Security-Eigenschaften werden initialisiert.
5. Klicken Sie in die erste Zeile der Tabelle "Gerätezertifikate", um ein neues Gerätezertifikat zu erzeugen oder ein bestehendes Gerätezertifikat auszuwählen.
6. Falls der Kommunikationspartner ebenfalls eine S7-1500 Station ist, müssen Sie dem Kommunikationspartner ebenfalls mit STEP 7 ein Gerätezertifikat zuweisen wie hier bzw. wie bei der S7-1500 CPU beschrieben.

Beispiel: Gesicherte TCP-Verbindung zwischen zwei S7-1500 CPUs über CP-Schnittstellen einrichten

Für die gesicherte TCP-Kommunikation zwischen zwei S7-1500 CPs müssen Sie in jeder CPU einen Datenbaustein mit dem Systemdatentyp TCON_IPv4_SEC erstellen, parametrieren und direkt an der Anweisung aufrufen.

Voraussetzungen:

- Beide S7-1500 CPUs haben mindestens Firmwarestand V2.0; wenn Sie den CP 1543SP-1 verwenden: Firmwarestand ab V1.0.
- Beide CPs (z. B. CP 1543-1) haben mindestens Firmwarestand V2.0
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.
  - Ein Gerätezertifikat (End-Entity-Zertifikat) für den CP muss erzeugt sein und sich im Zertifikatsspeicher des CP befinden. Wenn ein Kommunikationspartner ein Fremdgerät ist (z. B. ein MES oder ERP-System), muss für dieses Gerät ebenfalls ein Gerätezertifikat vorhanden sein.
  - Das Stammzertifikat (CA-Zertifikat), mit dem das Gerätezertifikat des Kommunikationspartners signiert ist, muss sich im Zertifikatsspeicher des CP bzw. im Zertifikatsspeicher des Fremdgeräts befinden. Falls Sie Zwischenzertifikate nutzen, müssen Sie sicherstellen, dass der gesamte Zertifikatepfad im validierenden Gerät vorhanden ist. Diese Zertifikate nutzt ein Gerät zur Validierung des Gerätezertifikats des Kommunikationspartners.
- Den Kommunikationspartner müssen Sie grundsätzlich über seine IPv4-Adresse adressieren, nicht über seinen Domänennamen.

Das folgende Bild zeigt die verschiedenen Zertifikate in den Geräten für den Fall, dass beide Kommunikationspartner über einen CP 1543-1 kommunizieren. Außerdem zeigt das Bild die Übertragung der Gerätezertifikate beim Verbindungsaufbau ("Hello").

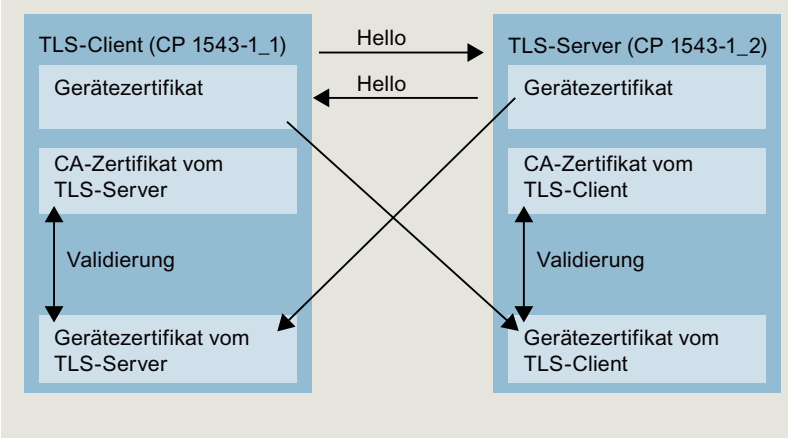

Einstellungen am TLS-Client

Um eine gesicherte TCP-Verbindung im TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Geben Sie dazu im Feld "Datentyp" die Zeichenfolge "TCON_IP_V4_SEC" ein.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Client" vom Datentyp TCON_IP_V4_SEC definiert ist. Die InterfaceId hat den Wert der HW-Kennung der IE-Schnittstelle des lokalen CP (TLS-Client).

Data_block_1

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	SEC connection 1 TLS-Client	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	258	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	100	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against th
19	TLSServerCertRef	UDInt	1	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	5	for Client side: add id of own X.509 V3 client certificate; for Sen

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Servers ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   – "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine ungesicherte TCP- oder UDP-Verbindung einrichten.
   – "TLSServerCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)).
   – "TLSClientCertRef": ID des eigenen X.509-V3-Zertifikats.
5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Einstellungen am TLS-Server

Um eine gesicherte TCP-Verbindung im TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Server" vom Datentyp TCON_IP_V4_SEC definiert ist. Die InterfaceId hat den Wert der HW-Kennung der IE-Schnittstelle des lokalen CP (TLS-Server).

Data_block_1

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	SEC connection 1 TLS-Server	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	260	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	10	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	true	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against the
19	TLSServerCertRef	UDInt	6	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	1	for Client side: add id of own X.509 V3 client certificate; for Serv

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Clients ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine unsichere TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert ": Anforderung eines X.509-V3-Zertifikats vom TLS-Client. Tragen Sie den Wert "true" ein.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.
   - "TLSClientCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)).

5. Legen Sie im Programmeditor eine Anweisung TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Laden des Geräts als neue Station

Wenn Sie eine Konfiguration mit Zertifikaten und projektierter Secure Open User Communication als neue Station in Ihr STEP 7-Projekt hochladen, dann werden die Zertifikate des CP im Gegensatz zu den Zertifikaten der CPU nicht mit hochgeladen. Nach dem Laden des Geräts als neue Station sind keine Zertifikate in den entspechenden Tabellen der CPs für die Gerätezertifikate mehr enthalten!

Sie müssen die Projektierung der Zertifikate nach dem Hochladen erneut durchführen. Andernfalls führt ein erneutes Laden der Konfiguration dazu, dass die ursprünglich im CP vorhandenen Zertifikate gelöscht werden und die Secure Communication nicht funktioniert.

Secure OUC-Verbindungen über CPU- und CP-Schnittstellen - Gemeinsamkeiten

- Verbindungsressourcen:
  Keine Unterschiede zwischen OUC und Secure OUC. Eine programmierte Secure OUC-Verbindung verbraucht ebenso eine Verbindungsressource wie eine OUC-Verbindung, unabhängig davon, über welche IE-/PROFINET-Schnittstelle die Station kommuniziert.
- Verbindungsdiagnose:
  Keine Unterschiede zwischen OUC und Secure OUC-Verbindungsdiagnose.
- Laden von Projekten mit Secure OUC-Verbindungen in die CPU:
  Nur im STOP der CPU möglich, falls Zertifikate mitgeladen werden.
  Empfehlung: Laden in Gerät > Hardware und Software. Grund: Sicherstellen der Konsistenz zwischen Programm mit Secure OUC, Hardware-Konfiguration und Zertifikaten. Zertifikate werden mit der Hardware-Konfiguration geladen - daher erfordert das Laden ein Stoppen der CPU. Das Nachladen von Bausteinen, die weitere Secure OUC-Verbindungen nutzen, ist nur dann im RUN möglich, wenn sich die dafür erforderlichen Zertifikate bereits auf der Baugruppe befinden.

Siehe auch

Verbindungsparameter nach TCON_IP_V4_SEC (Seite 119)

Zertifikate erstellen (Seite 1041)

Beispiele zum Verwalten von Zertifikaten (Seite 335)

Secure OUC über E-Mail

Im Folgenden ist beschrieben, wie Sie mit der Kommunikationsanweisung TMAIL_C eine gesicherte Verbindung (SNMP over TLS) zu einem Mailserver einrichten.

Gesicherte Verbindung zu einem Mailserver einrichten

Für die gesicherte Verbindung zu einem Mailserver müssen Sie selbst einen Datenbaustein mit einem der Systemdatentypen TMAIL_V4_SEC, TMAIL_V6_SEC oder TMAIL_QDN_SEC erstellen, parametrieren und direkt an der Anweisung TMAIL_C aufrufen.

Voraussetzungen:

- S7-1500 CPU ab Firmwarestand V2.0 mit Kommunikationsmodul CP 1543-1 ab Firmwarestand V2.0
- ET 200SP CPU ab Firmwarestand V2.0 mit Kommunikationsmodul CP 1542SP-1 (IRC) ab Firmwarestand V1.0
- Sie haben dem CP (TLS-Client) alle CA-Zertifikate des Mailservers (TLS-Server) zugewiesen und die Konfiguration in die CPU geladen.

Um eine gesicherte Verbindung Mailserver über die IPv4-Adresse des Mailservers einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TMAIL_V4_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "MailConnDB", in dem die Variable "MailConnectionSEC" vom Datentyp TMAIL_V4_SEC definiert ist.

MailConnDB

Name	Data type	Start value	Comment
Static			
MailConnectionSEC	TMail_V4_SEC		
InterfaceId	HW_ANY	"Local~CP_1543-1_1~Ethernet-Schnittstelle_1"	Use HW-identifier of the IE-interface to specify th
ID	CONN_OUC	100	connection reference / identifier
ConnectionType	Byte	16#20	type of connection 16#20=32=TMail_V4 or TMai
ActiveEstablished	Bool	true	active / passive connection establishment
WatchDogTime	Time	T#5000ms	watchdog time to monitor SMTP server associat
MailServerAddress	IP_V4		IPv4 address of mail server
ADDR	Array[1..4] of Byte		IPv4 address
ADDR[1]	Byte	144	IPv4 address
ADDR[2]	Byte	145	IPv4 address
ADDR[3]	Byte	2	IPv4 address
ADDR[4]	Byte	20	IPv4 address
UserName	String[254]	'myName'	user name which is necessary to login into the
PassWord	String[254]	'myPW'	user password which is necessary to login into
From	EMAIL_ADDR		source mail address
LocalPartPlusAt...	String[64]	'Mustermann@'	local part of e-mail address plus "@" sign
FullQualifiedD...	String[254]	'siemens.com'	full qualified domain name part of e-mail addre
RemotePort	UInt	587	remote TCP port number
ActivateSecureConn	Bool	TRUE	activate the security functionality of that conne
ExtTLSCapabilities	Byte	16#0	for further capability extensions of the TLS hand
TLSServerCertRef	UDInt	7	Reference to the X.509 V3 (CA-) certificate of the

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "MailServerAdress" die IPv4-Adresse des Mailservers ein.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein. Tragen Sie z. B. bei "TLSServerCertRef" die Zertifikat-ID vom CA-Zertifikat des Kommunikationspartners ein.
   - "ActivateSecureConn": Aktivierung von secure communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerCertRef": Referenz auf das X.509 V3 (CA)-Zertifikat des Mail Servers, welches vom TLS Client benutzt wird, um die Authentifizierung des Mailservers zu validieren.

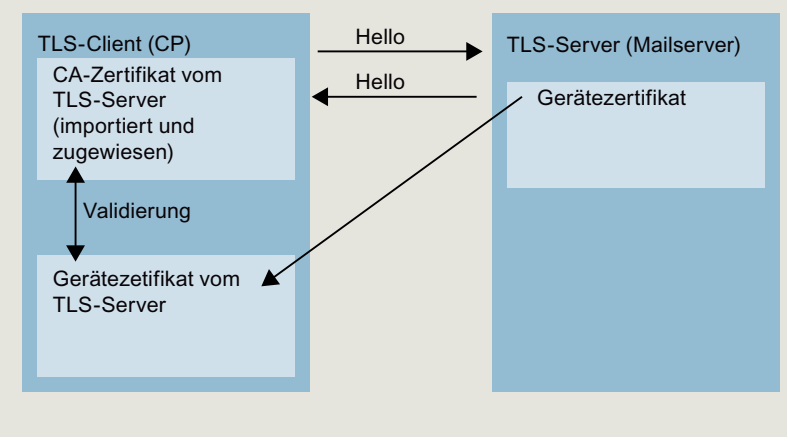

5. Legen Sie im Programmeditor eine Anweisung TMAIL_C an.
6. Verschalten Sie den Parameter MAIL_ADDR_PARAM der Anweisung TMAIL_C mit der Variablen vom Datentyp TMAIl_V4_SEC.
   Im folgenden Beispiel ist der Parameter Mail_ADDR_PARAM der Anweisung TMAIL_C mit der Variablen "MailConnectionSEC" (Datentyp TMAIL_V4_SEC) verschaltet.

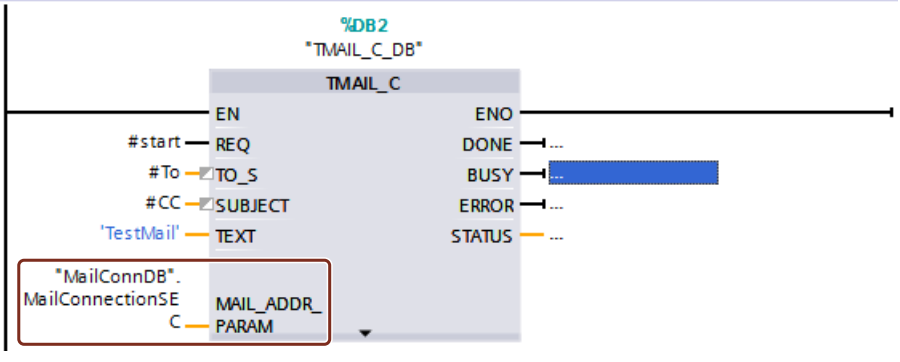

Siehe auch

Verbindungsparameter nach TCON_QDN_SEC (Seite 122)

Beispiele zum Verwalten von Zertifikaten (Seite 335)

1.3.5 Industrial Ethernet Security

1.3.5.1 Security projektieren

Allgemein

Unterstützte Geräte

Unterstützte Geräte

Die in diesem Hilfeabschnitt beschriebenen Security-Funktionen werden von folgenden Produkten unterstützt:

- SCALANCE S-600 V3:
  - S602 V2*/V3/V4
  - S612 V2*/V3/V4
  - S613 V2*
  - S623 V3/V4
  - S627-2M V4
- SOFTNET Security Client:
  - SOFTNET Security Client V4
- S7-CPs / TIM: CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 443-1 OPC UA, CP 1543-1, CP 1542SP-1 IRC, CP 1543SP-1, CP 1243-1 BX30, CP 1243-1 PCC, CP 1242-7 KX31, CP 1243-7 LTE, CP 1243-8 IRC, TIM 4R-IE, TIM 4R-IE DNP3, TIM 1531 IRC
- PC-CP: CP 1628
- Mobilfunk-Router: SCALANCE M875

* SCALANCE S V2 Geräte können nicht in ein STEP 7 Projekt eingefügt werden. SCALANCE S V2 Geräte aus Projekten älterer STEP 7 Versionen werden lediglich mit ihren Eigenschaften angezeigt. Die Eigenschaften dieser Geräte können jedoch nicht geändert werden. Der Download auf sowie das Übersetzen und die Diagnose von SCALANCE S V2 Geräten ist ebenfalls nicht möglich. Sämtliche für SCALANCE S Geräte beschriebenen Funktionen beziehen sich auf die Firmwareversionen ab V3.

Beim Öffnen eines STEP 7 Projekts mit bereits vorhandenen SCALANCE S V2 Geräten können aus den für das Gerät vorhandenen lokalen Firewall-Regeln globale Firewall-Regeln erstellt werden.

Allgemeine Benennung "Security-Modul"

Im vorliegenden Abschnitt des Informationssystems wird die Benennung "Security-Modul" stellvertretend für die oben genannten Produktnamen verwendet. Die Funktionsunterschiede zwischen den Produkten finden Sie in der Siemens Industry Mall.

Verwendung der Benennungen "Schnittstelle" und "Port"

In der vorliegenden Dokumentation werden die Ports von SCALANCE S Modulen wie folgt benannt:

- "Externe Schnittstelle": Der externe Port beim SCALANCE S602 / S612 / S613 / S623 bzw. ein externer Port beim SCALANCE S627-2M (rote Markierung)
- "Interne Schnittstelle": Der interne Port beim SCALANCE S602 / S612 / S613 / S623 bzw. ein interner Port beim SCALANCE S627-2M (grüne Markierung)
- "DMZ-Schnittstelle": Der DMZ-Port beim SCALANCE S623 / S627-2M (gelbe Markierung)

Die Benennung "Port" selbst wird dann verwendet, wenn ein spezieller Port einer Schnittstelle im Vordergrund steht.

Struktur dieses Hilfeabschnitts

Themen, die für mehrere Modultypen relevant sind, finden Sie im Kapitel "Allgemein". Informationen, die nur für einen bestimmten Modultyp relevant sind, finden Sie in den jeweiligen modultypspezifischen Abschnitten.

Übersicht - Leistungsumfang und Arbeitsweise

Allgemeine Benennung "STEP 7"

Die Projektierung von Security-Funktionen wird ab STEP 7 V12 unterstützt. Deshalb wird im vorliegenden Abschnitt des Informationssystems die Benennung "STEP 7" stellvertretend für alle Versionsstände von STEP 7 ab V12 verwendet.

Leistungsumfang

Folgende Security-Funktionen können Sie in STEP 7 nutzen:

- Projektierung der Security-Module
- Erstellen von VPN-Konfigurationsdaten für SOFTNET Security Client V4
- Erstellen von VPN-Konfigurationsdaten für SCALANCE M875
- Erstellen von VPN-Konfigurationsdateien für VPN-Geräte von Fremdherstellern
- Test- und Diagnosefunktionen, Statusanzeigen

Offline-Projektierungssicht und Online-/Diagnosesicht

Die Security-Funktionen werden in zwei Sichtweisen konfiguriert:

- Offline-Projektierungssicht
  In der Offline-Projektierungssicht erfolgt die Projektierung der Konfigurationsdaten. Vor dem Ladevorgang muss hierbei keine Verbindung zu den Security-Modulen bestehen.
- Online-/Diagnosesicht
  Die Online-/Diagnosesicht dient der Diagnose der Security-Module und bietet unter anderem die Möglichkeit, Firmware-Updates durchzuführen.

Arbeitsweise - Sicherheit und Konsistenz

- Zugriff nur für autorisierte Benutzer
  Die Security-Funktionen jedes Projekts sind nur bei aktivem Projektschutz verfügbar. Dadurch können die Security-Funktionen vor unbeabsichtigtem oder unberechtigtem Ändern geschützt werden.
- Konsistente Projektdaten
  Schon während der Eingabe in den einzelnen Dialogen erfolgen Konsistenzprüfungen. Zusätzlich werden dialogübergreifende, projektweite Konsistenzprüfungen durchgeführt. Auf die Security-Module können nur konsistente Projektdaten geladen werden.
- Schutz der Projektdaten durch Verschlüsselung
  Die Security-relevanten Projekt- und Konfigurationsdaten sind jeweils durch Verschlüsselung geschützt. Die Datenablage kann je nach Security-Modul im Projekt und/oder auf dem C-PLUG erfolgen.

Bedienoberfläche für Security-Funktionen

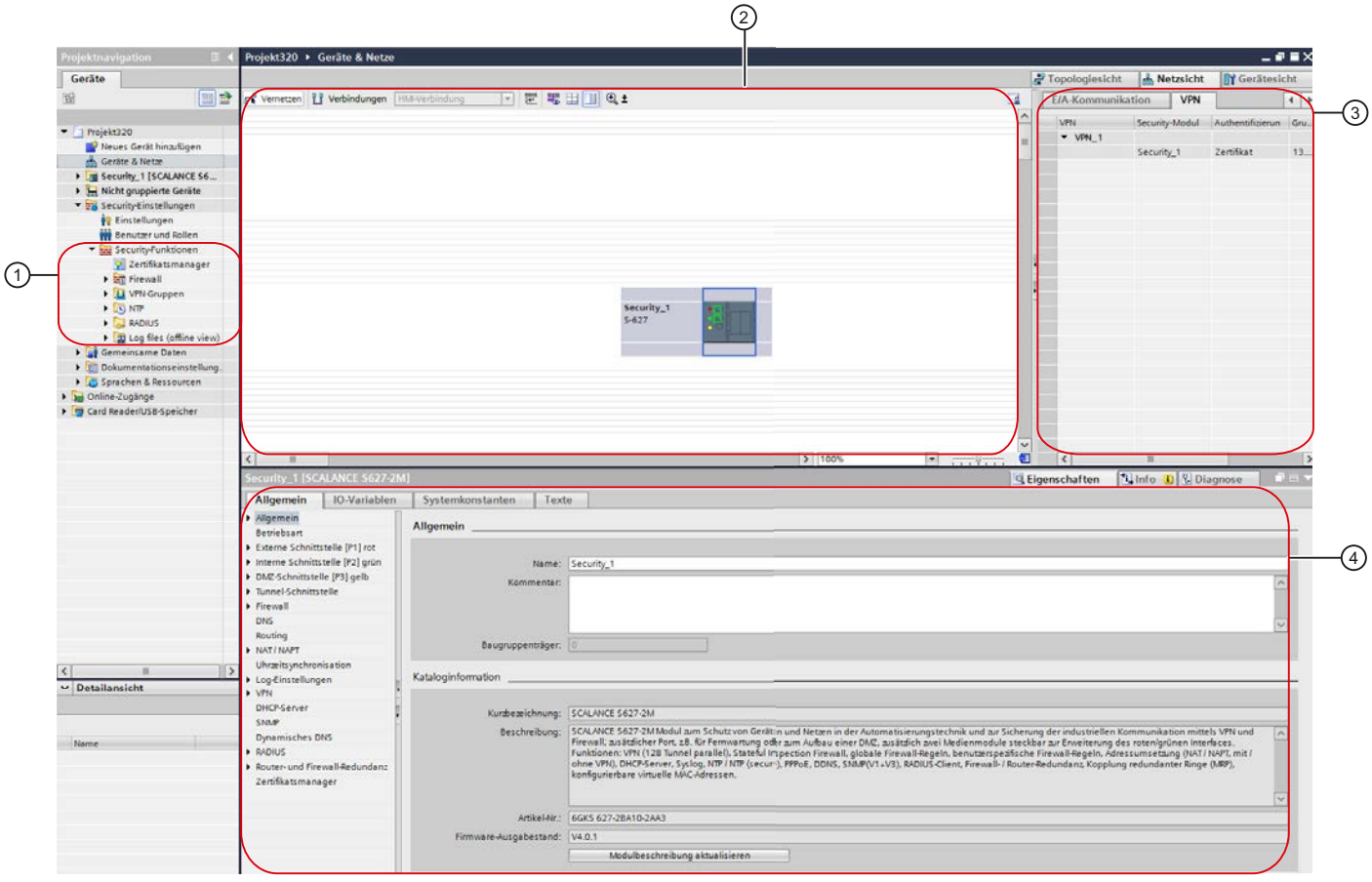

① Security-Funktionen

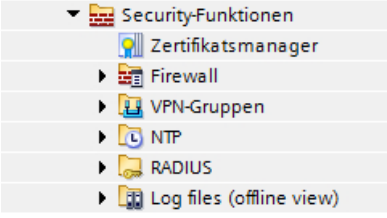

In der Projektnavigation befinden sich die globalen Security-Funktionen. Die globalen Security-Funktionen können modulunabhängig projektiert und im Anschluss ggf. einzelnen Security-Modulen zugewiesen werden. Änderungen in den globalen Security-Funktionen müssen auf alle betroffenen Security-Module geladen werden. Dies gilt auch für die Einstellungen von Redundanzbeziehungen.

Für die vollständige Anzeige der globalen Security-Funktionen in der Projektnavigation muss der Projektschutz aktiviert und mindestens ein Security-Modul im Projekt vorhanden sein. Handelt es sich bei dem ersten vorhandenen Security-Modul um einen CP, müssen zusätzlich die lokalen Security-Einstellungen des CPs aktiviert werden.

Folgende Hauptordner und Einträge sind in den globalen Security-Funktionen verfügbar:

- Zertifikatsmanager
  Im Zertifikatsmanager erhalten Sie eine Übersicht über alle im Projekt verwendeten Zertifikate. Sie können z. B. neue Zertifikate importieren sowie bestehende Zertifikate exportieren, erneuern oder ersetzen.

- Firewall
  Unter dem Eintrag "Firewall" können Sie globale IP- und MAC-Firewall-Regelsätze sowie benutzerspezifische IP-Regelsätze (nur für SCALANCE S Module) definieren und Security-Modulen zuweisen. Globale Firewall-Regelsätze können zudem aus STEP 7 exportiert und in STEP 7 importiert werden. Mit Hilfe von IP- und MAC-Dienst-Definitionen lassen sich die IP- und MAC-Firewall-Regeln kompakt und übersichtlich definieren.
- VPN-Gruppen
  In diesem Ordner sind alle erstellten VPN-Gruppen enthalten. Sie können hier neue VPN-Gruppen erstellen und Security-Module diesen VPN-Gruppen zuordnen. Zudem können die VPN-Gruppeneigenschaften von bereits angelegten VPN-Gruppen angepasst werden.
- NTP
  Hier können Sie gesicherte NTP-Server anlegen und einem oder mehreren Security-Modulen zuweisen. Damit erreichen Sie, dass die Zeitsynchronisation über den zugewiesenen NTP-Server erfolgt. Nicht gesicherte NTP-Server sind in den lokalen Security-Einstellungen konfigurierbar.
- RADIUS
  Hier können Sie RADIUS-Server anlegen und einem oder mehreren Security-Modulen zuweisen. Damit erreichen Sie, dass Authentifizierungsanfragen von Benutzern, die sich auf dem ausgewählten Security-Modul zur Aktivierung benutzerspezifischer IP-Regelsätze anmelden, an den zugeordneten RADIUS-Server weitergeleitet werden.
- Log-Dateien (Offline-Ansicht)
  Öffnen von geloggten System-, Audit- und Paketfilterereignissen, die Sie in der Online-Diagnose als Datei gespeichert haben.

② Arbeitsbereich

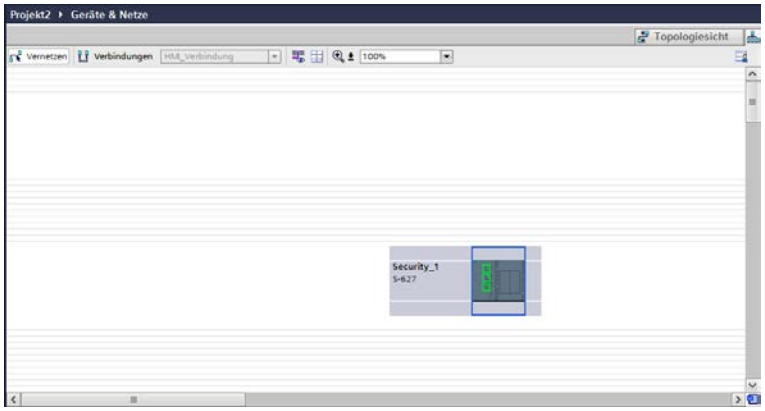

Nachdem Sie im Arbeitsbereich ein Security-Modul, eine SCALANCE M Modul oder ein Security-Gerät selektiert haben, können Sie unter "Eigenschaften" > "Allgemein" dessen lokale Security-Einstellungen projektieren. Wenn sich das selektierte Objekt in einer VPN-Gruppe befindet, werden zugehörige Informationen im VPN-Register angezeigt.

③ VPN-Register

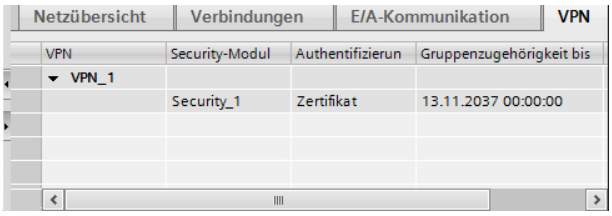

In diesem Register werden Informationen zu allen VPN-Gruppen angezeigt, zu denen das Security-Modul gehört, das im Arbeitsbereich selektiert wurde. Informationen zu den jeweiligen Teilnehmern einer VPN-Gruppe können ein- und ausgeblendet werden.

④ Lokale Security-Einstellungen

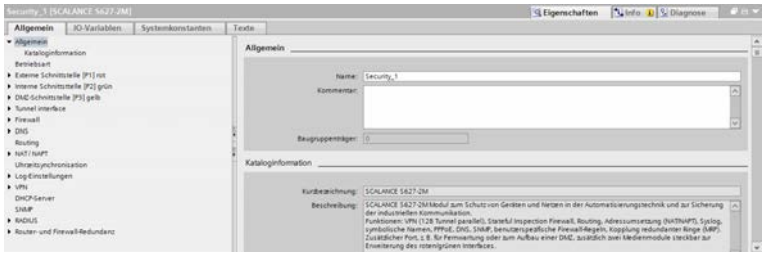

Lokale Security-Einstellungen werden für ein bestimmtes Security-Modul / SCALANCE M Modul / Security-Gerät projektiert. Nachdem eines dieser Objekte im Arbeitsbereich selektiert wurde, sind dessen lokale Security-Einstellungen im Register "Eigenschaften" > "Allgemein" des Inspektorfensters verfügbar. Für die Anzeige der lokalen Security-Einstellungen muss der Projektschutz aktiviert sein.

Für CPs muss zusätzlich im Inspektorfenster im Register "Eigenschaften" > "Allgemein", Eintrag "Security" das Kontrollkästchen "Aktiviere Security-Funktionen" aktiviert sein. Die lokalen Security-Einstellungen werden danach unterhalb des Eintrags "Security" angezeigt. Durch das Aktivieren des Kontrollkästchens werden einige der CP-Einstellungen, sofern sie aktiviert waren, automatisch in die lokalen Security-Einstellungen migriert.

Außerdem stehen Ihnen abhängig vom jeweiligen Security-Modul zusätzliche Security-Funktionen wie z. B. NTP (secure), SNMPv3, FTPS zur Verfügung.

Zusätzlich werden für projektierte Verbindungen automatisch Firewall-Regeln angelegt, die den Verbindungsaufbau freigeben. Zum Aufzeichnen geblockter Pakete stehen Ihnen entsprechende Log-Einstellungen zur Verfügung.

Gesicherte und ungesicherte Projektierungsbereiche

Die Bedienoberfläche lässt sich in gesicherte und ungesicherte Projektierungsbereiche unterteilen.

Gesichert sind diejenigen Bereiche, in welchen die Projektierung erst nach Aktivierung des Projektschutzes erfolgen kann. Diese Bereiche sind verschlüsselt und somit nur für in der Benutzerverwaltung autorisierte Personen verfügbar, auch wenn das Projekt für einen erweiterten Personenkreis zugänglich ist.

Funktionen aus ungesicherten Bereichen können dagegen auch ohne Aktivierung des Projektschutzes projektiert werden. Diese Einstellungen müssen vor dem Laden des Projekts auf die Anlagenkomponenten auf ihre Korrektheit überprüft werden, sofern ein erweiterter Personenkreis Änderungen an dem Projekt vornehmen kann.

In der folgenden Auflistung ist dargestellt, welche Projektierungsbereiche der Bedienoberfläche gesichert und welche ungesichert sind. Dies hängt zum Teil vom Security-Modul ab, für welches die Projektierung erfolgt.

- Alle Einstellungen aus den globalen Security-Funktionen sind gesichert.
- Gesicherte und ungesicherte Projektierungsbereiche für SCALANCE S Module:
  - Die Einstellungen unterhalb des Eintrags „Allgemein“ in den lokalen Security-Einstellungen sind ungesichert.
  - Übergeordnete Einstellungen (z.B. MRP-Einstellungen wie MRP-Manager etc.), welche nicht am Security-Modul selbst projektiert werden, sich aber ggf. auf das Security-Modul auswirken, sind ungesichert. Dies betrifft nicht die globalen Security-Funktionen.
  - Die übrigen Einstellungen sind gesichert.
- Gesicherte und ungesicherte Projektierungsbereiche für CPs:
  - Alle Einstellungen außerhalb des Eintrags „Security“ sind ungesichert.
  - Übergeordnete Einstellungen (z.B. MRP-Einstellungen wie MRP-Manager, PROFINET-Einstellungen, Verbindungen etc.), welche nicht am Security-Modul selbst projektiert werden, sich aber ggf. auf das Security-Modul auswirken, sind ungesichert. Dies betrifft nicht die globalen Security-Funktionen.
  - Alle Einstellungen für die Schnittstellen und Ports, insbesondere IP-Adressen, sind ungesichert.
  - Die Einstellungen unterhalb des Eintrags „Security" sind gesichert.

Einschränkungen bei der Verwendung von Bibliotheken

Wenn für ein Security-Modul eine Kopiervorlage in der globalen Bibliothek erstellt wird, geht die Projektierung dieses Security-Moduls in der Kopiervorlage verloren. Bei Verwendung der Kopiervorlage ist eine erneute Anmeldung erforderlich.

Wenn für ein Security-Modul eine Kopiervorlage in der Projektbibliothek erstellt wird, bleibt die Projektierung dieses Security-Moduls mit Ausnahme von modulübergreifenden Informationen für VPN- und Redundanzbeziehungen erhalten.

IP-Adressen für projektierbare Server

Bei der Projektierung von Servern dürfen nicht die (Alias-) IP-Adressen des Security-Moduls verwendet werden. Für alle projektierbaren Server findet bei IP-Adressangabe eine Prüfung auf Konsistenz statt.

Konsistenzprüfung durchführen

Übersicht

Folgende Konsistenzprüfungen gibt es:

- Lokale Konsistenzprüfungen
- Projektweite Konsistenzprüfungen

In den einzelnen Dialogbeschreibungen dieser Hilfe sind unter dem Stichwort "Konsistenzprüfung" die Regeln aufgeführt, auf die Sie jeweils bei der Eingabe achten müssen.

Lokale Konsistenzprüfungen

Eine Konsistenzprüfung heißt lokal, wenn sie direkt innerhalb eines Dialogs durchgeführt werden kann. Bei folgenden Aktionen werden lokale Konsistenzprüfungen durchgeführt:

- nach dem Verlassen eines Feldes
- nach dem Verlassen einer Zeile in einer Tabelle
- beim Bestätigen eines Dialogs mit OK.

Projektweite Konsistenzprüfungen

Projektweite Konsistenzprüfungen geben Aufschluss über die Korrektheit der konfigurierten Projektdaten. Bei folgenden Aktionen wird eine projektweite Konsistenzprüfung durchgeführt:

- beim Übersetzen einer Konfiguration
- beim Laden einer Konfiguration

Hinweis

Projektierte Daten können Sie nur auf ein Security-Modul laden, wenn die projektweite Konsistenzprüfung für das Security-Modul bestanden wurde.

Security-Modul austauschen

Modulspezifische Funktion

Es können SCALANCE S Module ab V3 gegen SCALANCE S Module ab V3 getauscht werden, siehe Abschnitt:
Security-Modul austauschen (Seite 501).

Zusätzlich können CP 1543-1 und TIM 1531 IRC jeweils gegen Module höherer Firmware-Versionen ausgetauscht werden. Hierzu müssen die lokalen Security-Einstellungen der Module aktiviert sein.

Voraussetzung

Um Security-Module austauschen zu können, müssen deren Modulbeschreibungen aktuell sein. Um die Modulbeschreibung von Security-Modulen zu aktualisieren, gehen Sie wie folgt vor:

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Klicken Sie in den lokalen Security-Einstellungen auf den Eintrag "Allgemein" > "Kataloginformation".
3. Klicken Sie auf die Schaltfläche "Modulbeschreibung aktualisieren".

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Security-Modul in der Topologie- oder Netzsicht.
2. Klicken Sie mit der rechten Maustaste auf das Security-Modul und wählen Sie im Kontextmenü den Menübefehl "Gerät tauschen...".

Zertifikate verwalten

Übersicht zu Zertifikaten

Verwaltung von Zertifikaten

Der Zertifikatsmanager in den globalen Security-Funktionen enthält eine Übersicht aller im Projekt verwendeten Zertifikate, z. B. CA-Zertifikate mit Angaben zu Zertifikatsinhaber, Aussteller, Gültigkeit, Verwendung und Vorhandensein eines privaten Schlüssels.

Ein CA-Zertifikat ist ein durch eine Zertifizierungsstelle, die so genannte "Certificate Authority", ausgestelltes Zertifikat, von dem die Gerätezertifikate abgeleitet werden. Zu den Gerätezertifikaten gehören SSL-Zertifikate. SSL-Zertifikate werden zur Authentifizierung bei der gesicherten Kommunikation zwischen einem Netzwerkteilnehmer und einem Security-Modul benötigt. Weitere Gerätezertifikate sind die VPN-Gruppen-Zertifikate von Security-Modulen, die sich in VPN-Gruppen befinden sowie OPC UA Client-/Serverzertifikate von CP 443-1 OPC UA Modulen.
Zertifizierungsstellen können sein:

- Das STEP 7 Projekt selbst. Sind "Zertifikatsinhaber" und "Aussteller" gleich, dann handelt es sich um ein selbst-signiertes, also durch das STEP 7 Projekt ausgestelltes Zertifikat.
- Eine übergeordnete Zertifizierungsstelle. Diese projektexternen Fremdzertifikate werden importiert und im Zertifikatsspeicher von STEP 7 abgelegt.

Zertifikate, die von einer der beiden Zertifizierungsstellen angelegt werden, haben immer einen privaten Schlüssel, damit die Gerätezertifikate abgeleitet werden können.

Jedem Zertifikat ist eine ID zugeordnet, über die das Zertifikat in Programmbausteinen referenziert werden kann. Die IDs werden bei der Erstellung von Zertifikaten fortlaufend erzeugt. Systembedingt werden einige IDs nicht verwendet.

Hinweis

Laden der Projektierung

Nach dem Ersetzen oder Erneuern von Zertifikaten muss die Projektierung auf die entsprechenden Security-Module geladen werden.

Nach dem Ersetzen oder Erneuern von CA-Zertifikaten muss die Projektierung auf alle Security-Module geladen werden.

Hinweis

Aktuelles Datum und aktuelle Uhrzeit auf den Security-Modulen

Achten Sie bei der Verwendung von gesicherter Kommunikation (z. B. HTTPS, VPN...) darauf, dass die betroffenen Security-Module über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die verwendeten Zertifikate werden sonst als nicht gültig ausgewertet und die gesicherte Kommunikation funktioniert nicht.

Hinweis

Zeichensatz von Zertifikaten

Zertifikate werden grundsätzlich mit dem ASCII-Zeichensatz abgespeichert. Sollte ein Zertifikat Zeichen enthalten, die nicht im ASCII-Zeichensatz vorhanden sind, wird das Zertifikat mit dem Zeichensatz UTF-8 abgespeichert. Dies ist für SCALANCE S V2 Module nicht möglich.

Globaler Zertifikatsmanager

Im Zertifikatsmanager in den globalen Security-Funktionen stehen die folgenden Funktionen zur Auswahl:

- Importieren von neuen Zertifikaten und Zertifizierungsstellen sowie Zertifikatsketten.
- Importieren von SSL-Zertifikaten (nur CP 343-1 Adv. / CP 443-1 Adv.), z. B. für FTP-Kommunikation.
- Exportieren der im Projekt verwendeten Zertifikate und Zertifizierungsstellen.
- Zuweisen eines Zertifikats zu einem oder mehreren Modulen, siehe Kapitel Zertifikate zuweisen (Seite 381).
- Erneuern von Zertifikaten, beispielsweise bei abgelaufener Gültigkeit oder für die Aktualisierung von Signaturen, siehe Kapitel Zertifikate erneuern (Seite 380).
- Ersetzen von Zertifikaten, beispielsweise für die Aktualisierung von Signaturen, siehe Kapitel Zertifikate ersetzen (Seite 381).
- Hinzufügen von vertrauenswürdigen Zertifikaten und Zertifizierungsstellen.
- Löschen von manuell importierten Zertifikaten.

Doppelklicken Sie in den globalen Security-Funktionen auf den Eintrag "Zertifikatsmanager".

In den einzelnen Registern stehen Ihnen die folgenden Befehle über das Kontextmenü zur Verfügung:

Befehl	Bedeutung
Importieren	Import von Zertifikaten. Folgende Formate sind erlaubt: *.der: DER-kodiertes Zertifikat (ohne privaten Schlüssel) *.cer: DER-kodiertes Zertifikat (ohne privaten Schlüssel) *.crt: PEM-kodiertes Zertifikat (ohne privaten Schlüssel) *.pem: PEM-kodiertes Zertifikat (ohne privaten Schlüssel) *.p12: Das Format kann neben dem Geräte- oder CA-Zertifikat auch alle Zertifikate des Zertifizierungspfads und zudem den dazugehörigen privaten Schlüssel enthalten.
Exportieren	Export von Zertifikaten. Folgende Formate sind erlaubt: *.der: DER-kodiertes Zertifikat (ohne privaten Schlüssel) *.cer: DER-kodiertes Zertifikat (ohne privaten Schlüssel) *.crt: PEM-kodiertes Zertifikat (ohne privaten Schlüssel) *.pem: PEM-kodiertes Zertifikat (ohne privaten Schlüssel) Für CA- und Gerätezertifikat können Sie zusätzlich folgende Formate auswählen: *.crl: Zertifikatssperrliste, DER-kodiert *.p12 PKSC12-Archiv: Zertifikat mit dazugehörigem privaten Schlüssel *.p12 PKSC12-Zertifikatskette: Zertifikat mit dazugehörigem privaten Schlüssel und alle Zertifikate des Zertifizierungspfads
Anzeigen	Öffnet den Zertifikatsdialog von Windows, in dem Sie eine Übersicht aller Zertifikatsdaten angezeigt bekommen. Damit das Zertifikat als gültig angezeigt wird, installieren Sie das Zertifikat über die Schaltfläche "Zertifikat installieren..." des Registers "Allgemein" im Zertifikatspeicher von Windows. Für die Installation des Zertifikats sind entsprechende Windows-Benutzerrechte erforderlich.
Erneuern (nur in den Registern "Zertifizierungsstelle (CA)" und "Gerätezertifikate")	Öffnet den Dialog "Zertifikat erneuern", in dem Sie bei Bedarf, z. B. bei kompromittierten Zertifikaten, ein neues Zertifikat von STEP 7 erzeugen lassen können.
Ersetzen	Öffnet den Dialog "Zertifikat ersetzen", in dem Sie ein bestehendes Zertifikat durch ein neues ersetzen können.
Zuweisen	Öffnet den Dialog "Zertifikat zuweisen", in dem Sie ein bestehendes Zertifikat einem oder mehreren Geräten zuweisen können.
Löschen	Löscht ein Zertifikat im Register "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen".

Importieren und Erneuern von *.p12-Zertifikaten

Wenn Sie eine Datei vom Typ *.p12 importieren oder erneuern erkennt das System automatisch, welche Zertifikate Teil der Datei sind.

Beispielsweise kann es sich um eine Zertifikatskette handeln, die aus folgenden Inhalten besteht:

- Gerätezertifikat A, das von Zertifizierungsstelle 1 ausgestellt wurde, mit privatem Schlüssel
- CA-Zertifikat B der Zertifizierungsstelle 1, ohne privaten Schlüssel
- CA-Zertifikat C der Zertifizierungsstelle 2, die Zertifikat B ausgestellt hat, ohne privaten Schlüssel

Beim Import der Datei werden nun die folgende Zertifikate abgelegt:

- Das Gerätezertifikat A im Register "Gerätezertifikate"; Privater Schlüssel: Ja
- Das CA-Zertifikat B der Zertifizierungsstelle im Register "Zertifizierungsstelle (CA)"; Privater Schlüssel: Nein
- Das CA-Zertifikat C der Zertifizierungsstelle 2 im Register "Zertifizierungsstelle (CA)"; Privater Schlüssel: Nein

Um beim Erneuern als Zertifizierungsstelle 1 neue Gerätezertifikate erzeugen zu können, gehen Sie folgendermaßen vor:

- Importieren Sie ein Zertifikat von Zertifizierungsstelle 1 mit privatem Schlüssel. Ergebnis:
  - Das Zertifikat B der Zertifizierungsstelle im Register "Zertifizierungsstelle (CA)" wird nicht überschrieben, da es breits importiert wurde.
  - Der Staus des der privaten Schlüssels wird von "Nein" auf "Ja" gesetzt .
  - Beim Erneuern können nun neue Gerätezertifikate signiert werden.

Lokaler Zertifikatsmanager

Zertifikate, die über den Zertifikatsmanager in den globalen Security-Funktionen importiert wurden, werden nicht automatisch den zugehörigen Modulen zugeordnet. Importierte Zertifikate müssen manuell über den Eintrag "Zertifikatsmanager" in den lokalen Security-Einstellungen in die Liste der vertrauenswürdigen Partnerzertifikate aufgenommen werden. Von STEP 7 erzeugte Zertifikate wie SSL-Zertifikate oder VPN-Gruppen-Zertifikate werden automatisch den zugehörigen Modulen zugeordnet und müssen diesen nicht über die lokalen Security-Einstellungen zugeordnet werden.

Bevor Zertifikate in Programmbausteinen für "Secure Communication" referenziert werden können, müssen diese Zertifikate dem Security-Modul über den lokalen Zertifikatsmanager als Gerätezertifikate zugeordnet werden. Nach Auswahl einer Tabellenzelle können Sie entweder ein neues Zertifikat über die Schaltfläche "Hinzufügen" erstellen oder ein bestehendes Zertifikat des Projekts über das Haken-Symbol verwenden. Für den Zugriff auf bestehende Zertifikate des Projekts sowie für die Signierung eines neuen Zertifikats durch eine Zertifizierungsstelle muss das Kontrollkästchen "Globale Security-Funktionen für den Zertifikatsmanager verwenden" aktiviert sein. Um das zugeordnete Zertifikat dem Partnermodul bekanntzumachen, muss dieses Zertifikat nach dem Import in die Liste der vertrauenswürdigen Partnerzertifikate aufgenommen werden.

Zertifizierungsstellen

Register "CA"

Folgende Zertifizierungsstellen werden angezeigt:

- CA-Zertifikate eines Projekts: Beim Erstellen eines neuen Projekts werden für das Projekt CA-Zertifikate erzeugt, die verschiedene Hash-Algorithmen für die Zertifikatsignatur verwenden. CA-Zertifikate für Projekte werden stets auf die zugehörigen Security-Module geladen. Von CA-Zertifikaten werden die SSL-Zertifikate sowie OPC UA Client-/Serverzertifikate für die einzelnen Security-Module abgeleitet.
- CA-Gruppenzertifikate: Beim Erstellen einer neuen VPN-Gruppe wird für die VPN-Gruppe ein CA-Zertifikat erzeugt. Von diesem Zertifikat werden die VPN-Gruppen-Zertifikate von VPN-Gruppenteilnehmern abgeleitet.

Gerätezertifikate

Register "Gerätezertifikate"

Anzeige der gerätespezifischen Zertifikate, die von den zugehörigen Zertifizierungsstellen für Module erzeugt werden. Dazu gehören:

- SSL-Zertifikat eines Moduls: Für jedes angelegte Modul wird ein SSL-Zertifikat erzeugt, das aus dem CA-Zertifikat des Projekts abgeleitet ist. SSL-Zertifikate werden zur Authentifizierung bei der gesicherten Kommunikation zwischen PG/PC und Modul sowie beim Laden der Konfiguration herangezogen.
- OPC UA Client-/Serverzertifikat der Baugruppe: Je nach projektierter OPC UA-Client-/Serverfunktion des CP 443-1 OPC UA wird ein OPC UA Client-/Serverzertifikat erzeugt, das zur Authentifizierung beim jeweiligen Kommunikationspartner verwendet wird.
- VPN-Gruppen-Zertifikat eines Moduls: Zusätzlich wird für jedes Modul pro VPN-Gruppe, in der es sich befindet, ein VPN-Gruppen-Zertifikat erzeugt.

Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen

Register "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen"

Anzeige der in STEP 7 importierten Fremdzertifikate. Importiert werden können z. B. Server-Zertifikate von externen FTPS-Servern, Projektzertifikate von anderen STEP 7 Projekten sowie Zertifikate, die für externe Dienste wie dyn. DNS notwendig sind.

Zertifikate müssen nach dem Import den zugehörigen Modulen über die lokalen Security-Einstellungen zugeordnet werden. Weitere Informationen hierzu finden Sie in Kapitel Übersicht zu Zertifikaten (Seite 375).

Zertifikate erneuern

Bedeutung

In diesem Dialog erneuern Sie CA-Zertifikate und Gerätezertifikate.

So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf einen Listeneintrag im Zertifikatsmanager.
2. Wählen Sie im Kontextmenü den Befehl "Erneuern".
3. Wählen Sie aus, ob das neue Zertifikat selbst signiert oder durch eine Zertifizierungsstelle signiert werden soll.
4. Wenn das Zertifikat durch eine Zertifizierungsstelle signiert werden soll, wählen Sie über die Schaltfläche "Auswählen" die zu verwendende Zertifizierungsstelle aus. Zur Auswahl stehen dabei nur Zertifizierungsstellen, die im Zertifikatsspeicher des aktuellen Projekts abgelegt sind. Dazu könne auch Zertifikatsketten gehören. Siehe Kapitel Übersicht zu Zertifikaten (Seite 375).
5. Geben Sie je nach Zertifikat die folgenden Parameter in die dafür vorgesehenen Eingabefelder ein:

Parameter	Bedeutung
Zertifikatsinhaber	Inhaber des Zertifikats. Der Wert für den Zertifikatsinhaber darf maximal 35 Zeichen enthalten. Die angezeigte ID zählt nicht zu diesem Wert.
Signatur	Algorithmus, der für das Signieren des Zertifikats verwendet wird.
Gültig von	Gültigkeitszeitraum des Zertifikats
Gültig bis	
Verwendungszweck	Wert, der in den Erweiterungen "KeyUsage" und "ExtendedKeyUsage" des Standards X.509 V3 für digitale Zertifikate eingetragen wird.
Alternativer Name des Zertifikatsinhabers (SAN)	Je nach zu erneuerndem Zertifikat wird der alternative Name des Zertifikatsinhabers mit einem anpassbaren Namenstyp und Wert angezeigt.

Beispiel: Signatur von Gerätezertifikat aktualisieren

Das Erneuern von Zertifikaten ist beispielsweise bei Gerätezertifikaten sinnvoll, deren Signatur von SHA-1 auf SHA-256 aktualisiert werden soll. Gehen Sie hierzu wie folgt vor:

1. Selektieren Sie das zu erneuernde Gerätezertifikat.
2. Wählen Sie den Kontextmenübefehl "Erneuern".
3. Klicken Sie im Dialog "Zertifikat erneuern" auf die Schaltfläche "Auswählen" und wählen Sie eine Zertifizierungsstelle (CA) mit einer SHA-256 Signatur aus.
   In den Zertifikatsparametern des Dialogs "Zertifikat erneuern" wird jetzt SHA-256 als Signatur des Gerätezertifikats angezeigt.

Zertifikate ersetzen

Bedeutung

In dem Dialog "Zertifikat ersetzen" ersetzen Sie ein bestehendes Zertifikat durch ein neues.

Beachten Sie, dass ein CA-Zertifikat nur durch ein CA-Zertifikat und ein Gerätezertifikat nur durch ein Gerätezertifikat ersetzt werden darf.

So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf den Listeneintrag eines Zertifikats.
2. Wählen Sie im Kontextmenü den Befehl "Ersetzen".
3. Der Dialog "Zertifikat ersetzen" wird geöffnet.

Alle in der Tabelle "Betroffene Zertifikate" aufgelisteten Zertifikate werden neu abgeleitet. Somit kann das CA-Gruppenzertifikat einer bereits projektierten VPN-Gruppe innerhalb des Projekts durch das CA-Gruppenzertifikat eines anderen Projekts ersetzt werden. Die VPN-Gruppen-Zertifikate für die VPN-Gruppenteilnehmer werden also in beiden Projekten von dem gleichen CA-Gruppenzertifikat abgeleitet.

Laden Sie nach Änderungen im Zertifikatsmanager die Projektierung auf alle betroffenen Security-Module.

Zertifikate zuweisen

Bedeutung

In dem Dialog "Zertifikat Modul zuweisen" weisen Sie ein bestehendes Zertifikat einem oderer mehreren Modulen zu.

So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf den Listeneintrag eines Zertifikats.
2. Wählen Sie im Kontextmenü den Befehl "Zuweisen".
3. Der Dialog "Zertifikat Modul zuweisen" wird geöffnet.
4. Wählen Sie ein oderer mehrere Module aus, denen das Zertifikat als vertrauenswürdiges Zertifikat zugewiesen werden soll.

Register "Gerätezertifikate"

Im Register "Gerätezertifikate" wählen Sie zusätzlich aus, ob das Zertifikat als Gerätezertifikat oder als vertrauenswürdiges Zertifikat zugewiesen werden soll. Wurde das Zertifikat bereits zugewiesen, ist die Spalte farblich hinterlegt.

Ein Zertifikat kann Modulen als Gerätezertifikat und anderen als vertrauenswürdiges Zertifikat zugewiesen werden.

Besonderheiten der Benutzerverwaltung für Security-Funktionen

Zugriff auf Security-Funktionen

Auf Security-Funktionen können Sie nur nach Aktivierung des Projektschutzes und nach erfolgreicher Anmeldung eines Benutzers zugreifen. Der angemeldete Benutzer muss zudem die erforderlichen Engineering- und Runtime-Rechte besitzen.

Grundlegende Informationen zur Benutzerverwaltung in STEP 7 finden Sie im Abschnitt "Benutzerverwaltung verwenden" des Informationssystems.

Engineering-Rechte für Security-Module

Diese Engineering-Rechte berechtigen zur Diagnose und Projektierung von Security-Modulen:

- Security: Security-Module mit Leserechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Module anzeigen.
- Security: Security-Module mit Schreibrechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Module projektieren und die Diagnose der Security-Module durchführen.

Rollen mit Runtime-Rechten für Security-Module

Den systemdefinierten Rollen mit dem Präfix "NET" sind standardmäßig Runtime-Rechte für Security-Module zugeordnet. Runtime-Rechte berechtigen zur Ausführung bestimmter Aufgaben auf Security-Modulen, beispielsweise zur Diagnose oder Firmware-Aktualisierung. Um diese Aufgaben ausführen zu können, müssen Sie Ihrem Benutzer eine geeignete systemdefinierte Rolle mit dem Präfix "NET" oder eine benutzerdefinierte Rolle mit den entsprechenden Runtime-Rechten zuordnen. Runtime-Rechte können Sie ebenso wie Engineering-Rechte nicht für systemdefinierte Rollen ändern. Für benutzerdefinierte Rollen können Sie die Runtime-Rechte beliebig ändern. Die Runtime-Rechte einer benutzerdefinierten Rolle legen Sie pro Security-Modul fest.

Es muss mindestens ein Benutzer vorhanden sein, dem eine Rolle mit vollständigen Runtime-Rechten für ein Security-Modul zugeordnet ist, z.B. die systemdefinierte Rolle "NET Administrator".

Den systemdefinierten Rollen mit dem Präfix "NET" sind folgende Engineering-Rechte zugeordnet:

Recht	NET Administrator	NET Standard	NET Diagnose
Projekt schreibgeschützt öffnen	x	x	x
Projekt mit Schreibrechten öffnen	x	x	-
Security-Einstellungen bearbeiten: "Einstellungen" und "Benutzer und Rollen"	x	-	-
Security: Security-Module mit Leserechten	x	x	x
Security: Security-Module mit Schreibrechten	x	x	-

Runtime-Rechte für Security-Module

Je nach Security-Modul stehen Ihnen die folgenden Runtime-Rechte zur Auswahl:

Tabelle 1-5 Runtime-Rechte CP 343-1 Advanced / 443-1 Advanced

Recht	NET Administrator	NET Standard	NET Diagnose
FTP: Dateien vom CP-Dateisystem lesen	x	x	x
FTP: Dateien auf das CP-Dateisystem schreiben	x	x	-
Webserver: CP-Dateisystem formatieren *	x	-	-
Applets: Werte über PLC-Variablen lesen *	x	x	x
Applets: Werte über Adressen lesen *	x	x	x
Applets: Werte über Adressen schreiben *	x	x	-
FTP: Dateien (DBs) von der S7-CPU lesen **	x	x	x
FTP: Dateien (DBs) in die S7-CPU schreiben ***	x	x	-
Applets: Status der Baugruppen im Rack lesen *	x	x	x
Applets: Artikelnummern der Baugruppen im Rack lesen *	x	x	x
Applets: Werte über PLC-Variablen schreiben *	x	x	-
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
SNMP: LLDP-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
Webserver: IP-Zugriffsschutzliste erweitern *	x	-	-
Webserver: Auf Web-Diagnose und CP-Dateisystem zugreifen	x	x	x
Webserver: Test-E-Mail versenden *	x	x	x
Webserver: Firmware aktualisieren *	x	x	-
SNMP: MRP-MIB lesen	x	x	x
SNMP: MRP-MIB schreiben	x	x	-
SNMP: SNMPv2-MIB lesen	x	x	x
Webserver: Nachladen von Diagnosetexten *	x	x	-

Tabelle 1-6 Runtime-Rechte CP 443-1 OPC UA

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
SNMP: LLDP-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
Webserver: Firmware aktualisieren	x	x	-
SNMP: SNMPv2-MIB lesen	x	x	x
Webserver: Nachladen von Diagnosetexten	x	x	-
OPC UA: Variablen lesen	x	x	x
OPC UA: Variablen schreiben	x	x	-
Webserver: Diagnose	x	x	x

Tabelle 1-7 Runtime-Rechte CP 1628

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x

Tabelle 1-8 Runtime-Rechte SCALANCE S-600 V3

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
SNMP: LLDP-MIB lesen	x	x	x
Laden der Konfigurationsdateien	x	x	-
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
Webserver: Firmware aktualisieren	x	x	-
SNMP: MRP-MIB lesen (nur für SCALANCE S627-2M)	x	x	x
SNMP: MRP-MIB schreiben (nur für SCALANCE S627-2M)	x	x	-
SNMP: SNMPv2-MIB lesen	x	x	x

Tabelle 1-9 Runtime-Rechte CP 1543-1

Recht	NET Administrator	NET Standard	NET Diagnose
FTP: Dateien vom CP-Dateisystem lesen	x	x	x
FTP: Dateien auf das CP-Dateisystem schreiben	x	x	-
FTP: Dateien (DBs) von der S7-CPU lesen **	x	x	x
FTP: Dateien (DBs) in die S7-CPU schreiben ***	x	x	-
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
SNMP: LLDP-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x
SNMP: IPv6-MIB lesen	x	x	x

Tabelle 1-10 Runtime-Rechte CP 1542SP-1 IRC

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x
SNMP: IPv6-MIB lesen	x	x	x
TeleService nutzen	x	-	-

Tabelle 1-11 Runtime-Rechte CP 1543SP-1

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
SNMP: LLDP-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x
SNMP: IPv6-MIB lesen	x	x	x

Tabelle 1-12 Runtime-Rechte CP 1243-1

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x
SNMP: IPv6-MIB lesen	x	x	x
TeleService nutzen	x	-	-

Tabelle 1-13 Runtime-Rechte CP 1242-7

Recht	NET Administrator	NET Standard	NET Diagnose
TeleService nutzen	x	-	-

Tabelle 1-14 Runtime-Rechte CP 1243-7 LTE

Recht	NET Administrator	NET Standard	NET Diagnose
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
TeleService nutzen	x	-	-

Tabelle 1-15 Runtime-Rechte CP 1243-8 IRC

Recht	NET Administrator	NET Standard	NET Diagnose
SNMP: MIB-II lesen	x	x	x
SNMP: MIB-II schreiben	x	x	-
SNMP: Automation-MIB lesen	x	x	x
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x
SNMP: SNMPv2-MIB lesen	x	x	x
SNMP: IPv6-MIB lesen	x	x	x

* Um die Funktion anzuwenden, muss das Runtime-Recht "Webserver: Auf Web-Diagnose und CP-Dateisystem zugreifen" ebenfalls aktiviert sein.

** Um die Funktion anzuwenden, muss das Runtime-Recht "FTP: Dateien vom CP-Dateisystem lesen" ebenfalls aktiviert sein.

*** Um die Funktion anzuwenden, muss das Runtime-Recht "FTP: Dateien auf das CP-Dateisystem schreiben" ebenfalls aktiviert sein.

Mengengerüst für Benutzer und Rollen auf Security-Modulen

- Maximale Benutzeranzahl auf einem Security-Modul:
  Die maximale Anzahl von Benutzern, denen eine Rolle mit Runtime-Rechten für ein Security-Modul zugeordnet ist, beträgt 33 (32 + 1 Administrator beim Anlegen des Projekts).
- Maximale Rollenanzahl auf einem Security-Modul:
  Die maximale Anzahl von Rollen, denen Runtime-Rechte für ein Security-Modul zugeordnet sind und die Benutzern zugeordnet sind, beträgt 37.

Parameter für benutzerspezifische IP-Regelsätze

Beachten Sie bei der Verwendung benutzerspezifischer IP-Regelsätze folgende Parameter der Benutzerverwaltung:

- Authentifizierungsmethode "RADIUS" (nur für SCALANCE S ab V4):
  Die Authentifizierung des Benutzers wird bei der Aktivierung benutzerspezifischer IP-Regelsätze durch einen RADIUS-Server durchgeführt. Das Passwort des Benutzers wird bei dieser Authentifizierungsmethode nicht in STEP 7 projektiert, sondern Sie müssen dieses auf dem RADIUS-Server hinterlegen. Verwenden Sie diese Authentifizierungsmethode ausschließlich für Benutzer, die sich lediglich auf der Webseite eines Security-Moduls anmelden sollen. Ein Benutzer mit der Authentifizierungsmethode "RADIUS" kann sich nicht an STEP 7 Projekten anmelden.
- Maximale Sitzungsdauer (nur für SCALANCE S ab V3):
  Eingabe der Zeitdauer, nach welcher ein Benutzer, der auf der Webseite für benutzerspezifische IP-Regelsätze von SCALANCE S Modulen angemeldet ist, automatisch abgemeldet wird. Die hier angegebene Zeitdauer beginnt nach der Anmeldung sowie nach dem Erneuern der Sitzung auf der Webseite des Security-Moduls.
  - Standardeinstellung: 30 Minuten
  - Mindestwert: 5 Minuten
  - Maximalwert: 480 Minuten

Authentifizierung durch RADIUS-Server

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V4 verfügbar, siehe Abschnitt:

AUTOHOTSPOT.

Konfigurationsdaten für SCALANCE M Module generieren

Bedeutung

Sie können die VPN-Informationen zur Parametrierung eines SCALANCE M mit STEP 7 generieren. Dazu muss sich das Modul in mindestens einer VPN-Gruppe mit einem Security-Modul oder einem SOFTNET Security Client befinden. Mit den generierten Dateien können Sie dann den SCALANCE M über das Web Based Management des Geräts konfigurieren.

Generierte Dateien

Folgende Dateitypen werden generiert:

- Exportdatei mit den Konfigurationsdaten
  - Dateityp: *.txt-Datei im ASCII-Format
  - Enthält die exportierten Konfigurationsinformationen für das SCALANCE M einschließlich einer Information über die zusätzlich erzeugten Zertifikate.
- VPN-Gruppen-Zertifikate des Moduls
  - Dateityp des privaten Schlüssels: *.p12-Datei
  - Die Datei enthält das Modulzertifikat und das Schlüsselmaterial.
  - Der Zugriff ist passwortgeschützt.
- CA-Zertifikate von VPN-Gruppen
  - Dateityp: *.cer-Datei

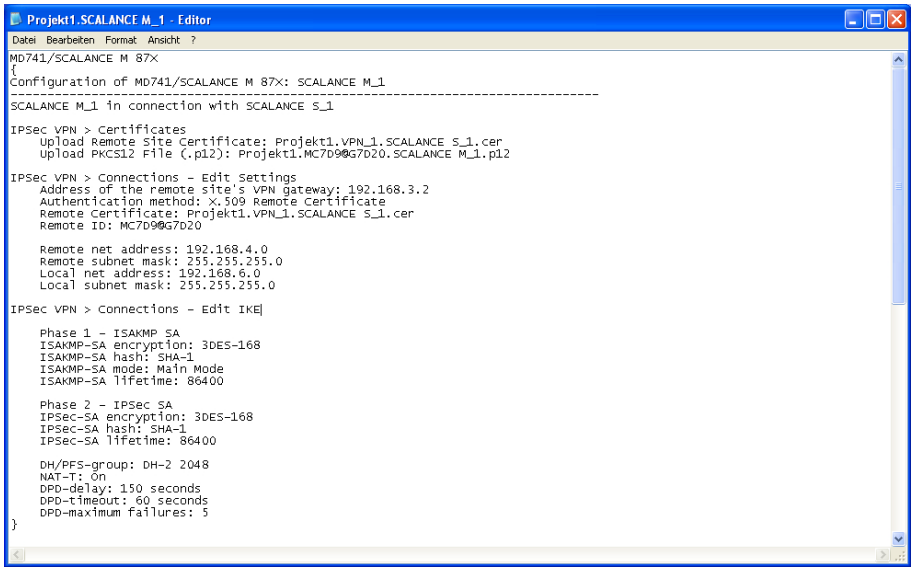

Bild 1-12 SCALANCE M Konfigurationsdatei

Hinweis

Keine Übertragung an das SCALANCE M Modul

Konfigurationsdateien werden nicht an das SCALANCE M Modul übertragen. Es wird eine ASCII-Datei generiert, mit der Sie die VPN-relevanten Eigenschaften des SCALANCE M konfigurieren können. Dazu muss sich der SCALANCE M in mindestens einer VPN-Gruppe mit einem anderen Security-Modul oder einem SOFTNET Security Client befinden.

Hinweis

Exportierte Konfigurationsdateien vor unberechtigtem Zugriff schützen

Aus STEP 7 exportierte Konfigurationsdateien für SCALANCE M können sicherheitsrelevante Informationen enthalten. Stellen Sie deshalb sicher, dass diese Dateien vor unberechtigtem Zugriff geschützt sind. Dies ist insbesondere bei der Weitergabe der Dateien zu beachten.

Gehen Sie so vor

1. Markieren Sie das Modul vom Typ "SCALANCE M".
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "SCALANCE M Konfiguration".
3. Aktivieren Sie das Kontrollkästchen "SCALANCE M Dateien generieren" und wählen Sie einen Speicherort für die Konfigurationsdateien.
4. Legen Sie ein Passwort für die Verschlüsselung der VPN-Gruppen-Zertifikate fest, indem Sie den Projektnamen als Passwort verwenden oder ein eigenes Passwort vergeben.
5. Übersetzen Sie die Konfiguration des SCALANCE M Moduls.

Ergebnis: Die Dateien (.txt-Datei und Zertifikate) werden in dem von Ihnen angegebenen Verzeichnis abgespeichert.

Konfigurationsdaten für Security-Geräte generieren

Bedeutung

Ein Security-Gerät steht stellvertretend für ein Gerät von einem anderen Hersteller.

V1.0: Das Gerät kann nur als VPN-Gerät betrieben werden.

Ab V1.1: Sie können wählen, ob das Gerät als VPN-Gerät oder Internet-Gateway betrieben werden soll.

Security-Gerät als VPN-Gerät

Sie können die VPN-Informationen zur Parametrierung eines VPN-Geräts mit STEP 7 generieren. Dazu muss sich das VPN-Gerät in mindestens einer VPN-Gruppe mit einem Security-Modul befinden. Mit den generierten Dateien können Sie dann das VPN-Gerät konfigurieren.

Hinweis

Exportierte Konfigurationsdateien vor unberechtigtem Zugriff schützen

Aus STEP 7 exportierte Konfigurationsdateien für VPN-Geräte können sicherheitsrelevante Informationen enthalten. Stellen Sie deshalb sicher, dass diese Dateien vor unberechtigtem Zugriff geschützt sind. Dies ist insbesondere bei der Weitergabe der Dateien zu beachten.

Um das Security-Gerät als VPN-Gerät zu verwenden, gehen Sie folgenermaßen vor:

1. Markieren Sie das zu bearbeitende Security-Gerät.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Einstellungen".
3. V1.1: Wählen Sie aus der Klappliste die Betriebsart "VPN-Gerät". Für V1.0 ist "VPN-Gerät" vorbelegt.
4. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN-Gerät".
5. Aktivieren Sie das Kontrollkästchen "Dateien des VPN-Geräts generieren".
6. Wählen Sie aus der Klappliste "VPN-Betriebsart" die Betriebsart des VPN-Geräts aus. Diese Auswahl ist nur möglich, wenn sich das VPN-Gerät in keiner VPN-Gruppe befindet.
   - VPN-Client: Das VPN-Gerät kann die folgende Rolle annehmen:
     Initiator: Das VPN-Gerät versucht aktiv, zu einer Gegenstelle eine VPN-Verbindung aufzubauen.
   - VPN-Server: Das VPN-Gerät kann die folgenden Rollen annehmen:
     Initiator/Responder: Das VPN-Gerät versucht aktiv, zu einer Gegenstelle eine VPN-Verbindung aufzubauen. Die Annahme von Anfragen zum VPN-Verbindungsaufbau ist ebenfalls möglich. Diese Rolle ist standardmäßig in den VPN-Verbindungseigenschaften ausgewählt.
     Responder: Die Annahme von Anfragen zum VPN-Verbindungsaufbau ist möglich.
   - NCP VPN-Client für Android (ab Version 2.1): Das VPN-Gerät mit installierter Software NCP Secure VPN Client for Android versucht aktiv, zu einer Gegenstelle eine VPN-Verbindung aufzubauen.
   - SOFNET Security Client V5: Das VPN-Gerät mit installierter Software SSC versucht aktiv, zu einer Gegenstelle eine VPN-Verbindung aufzubauen.
   - SINEMA RC-Client: Das VPN-Gerät mit installierter Software SINEMA RC-Client versucht aktiv, zu einer Gegenstelle eine VPN-Verbindung aufzubauen.
7. Wählen Sie einen Speicherort für die VPN-Konfigurationsdateien.
8. Legen Sie ein Passwort für die Verschlüsselung der VPN-Gruppen-Zertifikate fest, indem Sie den Projektnamen als Passwort verwenden oder ein eigenes Passwort vergeben.

Wenn Sie das VPN-Gerät als "VPN-Client" konfiguriert haben:

1. Legen Sie im Eintrag "Exportierte Zertifikate und Schlüssel" die zugehörigen Dateitypen fest.
2. Geben Sie im Eintrag "Durch Tunnel erreichbare Subnetze" die Subnetze und/oder IP-Adressen an, für die die Tunnelkommunikation freigegeben werden soll.

Wenn Sie das VPN-Gerät als "VPN-Server" konfiguriert haben:

1. Geben Sie im Feld "VPN-Server-Adresse (WAN-IP-Adresse)" die WAN-IP-Adresse an, unter welcher der VPN-Server vom Initiator der VPN-Verbindung erreichbar ist.
2. Legen Sie im Eintrag "Exportierte Zertifikate und Schlüssel" die zugehörigen Dateitypen fest.
3. Geben Sie im Eintrag "Durch Tunnel erreichbare Subnetze" die Subnetze und/oder IP-Adressen an, für die die Tunnelkommunikation freigegeben werden soll.

Wenn Sie das VPN-Gerät als "NCP VPN-Client für Android (ab Version 2.1)" konfiguriert haben:

1. Geben Sie im Eintrag "NCP VPN-Client für Android Einstellungen" den Speicherpfad auf dem NCP VPN-Client an.

Hinweis

Freigabe von Subnetzen und Teilnehmern bei VPN-Verbindung zwischen NCP Secure VPN-Client und SCALANCE S

Das lokale interne Subnetz von SCALANCE S Baugruppen ist standardmäßig für die Tunnelkommunikation freigegeben. Dieses wird deshalb stets als erstes freigegebenes Subnetz in der für den NCP VPN-Client generierten Datei eingetragen. Da NCP VPN-Client maximal 5 Einträge für freigegebene Subnetze / Teilnehmer unterstützt, wird standardmäßig das lokale interne Netz der SCALANCE S Baugruppe und die ersten 4 Tabelleneinträge aus dem Eintrag "Knoten" im NCP VPN-Client freigegeben. Um andere Subnetze und Teilnehmer für die Tunnelkommunikation freizugeben, muss die aus STEP 7 generierte Datei entsprechend angepasst werden.

Wenn Sie das VPN-Gerät als SOFTNET Security Client V5 konfiguriert haben:

- Es sind keine zusätzlichen Einstellungen notwendig.

Wenn Sie das VPN-Gerät als SINEMA RC-Client konfiguriert haben:

1. Wählen Sie über die Schaltfläche "Suchen" die Datei zum Starten des SINEMA RC-Client.
2. Über die Schaltfläche "SINEMA RC-Client starten" öffen Sie den SINEMA RC-Client. Für den SIENEMA RC-Client wird keine VPN-Konfigurationsdatei erstellt.

Security-Gerät als Internet-Gateway

Um das Security-Gerät als Internet-Gateway zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie das zu bearbeitende Security-Gerät.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Einstellungen".
3. Wählen Sie aus der Klappliste die Betriebsart "Internet-Gateway".
4. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Internet-Gateway".
5. Geben Sie die WAN-IP-Adresse oder den FQDN ein.
6. Unter dem Eintrag "Zugewiesene Module" können Sie dem Internet-Gateway Module zuweisen, die sich in einer VPN-Gruppe befinden.

Schnittstellen für SCALANCE S Module projektieren

Übersicht

Informationen zur Projektierung der Schnittstellen von SCALANCE S Modulen finden Sie im Abschnitt AUTOHOTSPOT. Die hier enthaltenen Informationen behandeln die folgenden Projektierungsmöglichkeiten:

- Betriebsart (Bridge-Modus / Routing-Modus / Ghost-Modus (nur für SCALANCE S602 ab V3.1)): Betriebsart einstellen (Seite 504)
- IP-Adressparameter: IP-Adressparameter projektieren (Seite 505)
- Porteinstellungen (nur für SCALANCE S ab V3): Port-Modus projektieren (Seite 506)
- Einstellungen zum Internet Service Provider (ISP), wenn eine der Schnittstellen über PPPoE betrieben wird (nur für SCALANCE S ab V3): Internetverbindung projektieren (Seite 507)
- DNS (nur für SCALANCE S ab V3): DNS projektieren (Seite 508)
- Dynamisches DNS (nur für SCALANCE S ab V3): Dynamisches DNS projektieren (Seite 509)
- LLDP (nur für SCALANCE S ab V4 im Routing-Modus): LLDP projektieren (Seite 512)
- Medienredundanz in Ringtopologien (z.B. MRP-Client), nur für SCALANCE S627-2M ab V4 im Routing-Modus): AUTOHOTSPOT

Informationen zu den Besonderheiten des Ghost-Modus finden Sie in folgendem Kapitel: Besonderheiten des Ghost-Modus (Seite 516)

Die Projektierung der Schnittstellen von CPs wird in den Kapiteln beschrieben, die sich mit CPs befassen.

Firewall einrichten

Übersicht zur Firewall

Bedeutung

Die Firewall-Funktionalität der Security-Module hat die Aufgabe, Netze und Stationen vor Fremdbeeinflussungen und Störungen zu schützen. Das bedeutet, dass nur bestimmte, vorher festgelegte Kommunikationsbeziehungen erlaubt werden. Nicht zugelassene Telegramme werden von der Firewall verworfen, ohne dass eine Antwort gesendet wird.

Zur Filterung des Datenverkehrs können u. a. IP-Adressen, IP-Subnetze, Dienste oder MAC-Adressen verwendet werden. Außerdem können Sie eine Begrenzung der Übertragungsgeschwindigkeit einstellen.

Die Firewall-Funktionalität kann für folgende Protokollebenen konfiguriert werden:

- IP-Firewall mit Stateful Packet Inspection (Layer 3 und 4)
- Firewall auch für Ethernet-"Non-IP"-Telegramme gemäß IEEE 802.3 (Layer 2)

Bei einem VPN-fähigen Security-Modul kann die Firewall auch für den verschlüsselten Datenverkehr (IPsec-Tunnel) eingesetzt werden. Beim Security-Modul SCALANCE S602 kann die Firewall nur für den unverschlüsselten Datenverkehr eingesetzt werden.

Firewall-Regeln

Firewall-Regeln beschreiben, welche Pakete in welche Richtung erlaubt bzw. verboten werden. IP-Regeln wirken auf alle IP-Pakete ab Ebene 3. MAC-Regeln wirken nur auf Frames unterhalb Ebene 3.

Arten von Firewall-Regeln

- Globale Firewall-Regelsätze: Globale Firewall-Regelsätze können mehreren Security-Modulen gleichzeitig zugewiesen werden. Globale Firewall-Regelsätze werden in den globalen Security-Funktionen projektiert.
- Lokale Firewall-Regeln: Lokale Firewall-Regeln werden in den lokalen Security-Einstellungen eines Security-Moduls projektiert.
- Benutzerspezifische IP-Regelsätze (nur für SCALANCE S ab V3): Benutzerspezifische IP-Regelsätze können einzelnen oder mehreren Security-Modulen gleichzeitig zugewiesen werden. Benutzerspezifische IP-Regelsätze werden in den globalen Security-Funktionen projektiert und dort einem oder mehreren Benutzern zugewiesen.
  SCALANCE S ab V4 (RADIUS): Benutzerspezifischen IP-Regelsätzen können neben einzelnen oder mehreren Benutzern auch einzelne oder mehrere Rollen zugewiesen sein.

Dienst-Definitionen

Zusätzlich haben Sie die Möglichkeit, mit Hilfe von Dienst-Definitionen Firewall-Regeln kompakt und übersichtlich zu definieren. Dienst-Definitionen werden in den globalen Security-Funktionen projektiert und können in den globalen, lokalen und benutzerspezifischen Firewall-Regeln verwendet werden.

Standardregeln für IP-Dienste anpassen

Für SCALANCE S Module ab V3 können Sie dienstspezifische Firewall-Regeln, die für die Schnittstellen der Security-Module standardmäßig eingestellt sind, anpassen. Informationen zur Projektierung dieser Firewall-Regeln finden Sie im Abschnitt Standardregeln für IP-Dienste anpassen (Seite 532).

Automatisch erzeugte Firewall-Regeln für CP-Verbindungen

Für Verbindungen, die über CPs projektiert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

Welche Firewall-Regeln automatisch erzeugt werden, erfahren Sie in folgenden Abschnitten.

- Für S7-300-/S7-400-/PC-CPs: Verbindungsbezogene automatische Firewall-Regeln (Seite 566) im Kapitel "Security für S7-300-/S7-400-/PC-CPs.
- Für S7-1200-/S7-1500-CPs: Verbindungsbezogene automatische Firewall-Regeln (Seite 580) im Kapitel "Security für S7-1200-/S7-1500-CPs".

Firewall aktivieren

In den lokalen Security-Einstellungen wird die Firewall-Funktionalität für ein bestimmtes Security-Modul über das Kontrollkästchen "Firewall aktivieren" gesteuert. Ist das Kontrollkästchen aktiviert, kann die Firewall konfiguriert werden und ist nach dem Ladevorgang wirksam. Befindet sich das Security-Modul in einer VPN-Gruppe, ist das Kontrollkästchen "Firewall aktivieren" standardmäßig aktiviert und kann nicht deaktiviert werden. Nach dem Umschalten in den erweiterten Firewall-Modus ist es nicht möglich, in den Standard-Modus zurückzuschalten. Für nähere Informationen zum Standard-Modus und erweiterten Modus der Firewall finden Sie im Kapitel:

Übersicht Lokale Firewall-Regeln (Seite 416).

Globale Firewall-Regelsätze

Anwendung

Globale Firewall-Regelsätze werden modulunabhängig in den globalen Security-Funktionen projektiert. Ein Firewall-Regelsatz besteht aus einer oder mehreren Firewall-Regeln und wird einzelnen Security-Modulen zugewiesen.

Unterschieden wird bei den globalen Firewall-Regelsätzen zwischen:

- IP-Regelsätzen
- MAC-Regelsätzen

Die folgende Darstellung verdeutlicht den Zusammenhang zwischen global definierten Regelsätzen und lokal verwendeten Regelsätzen.

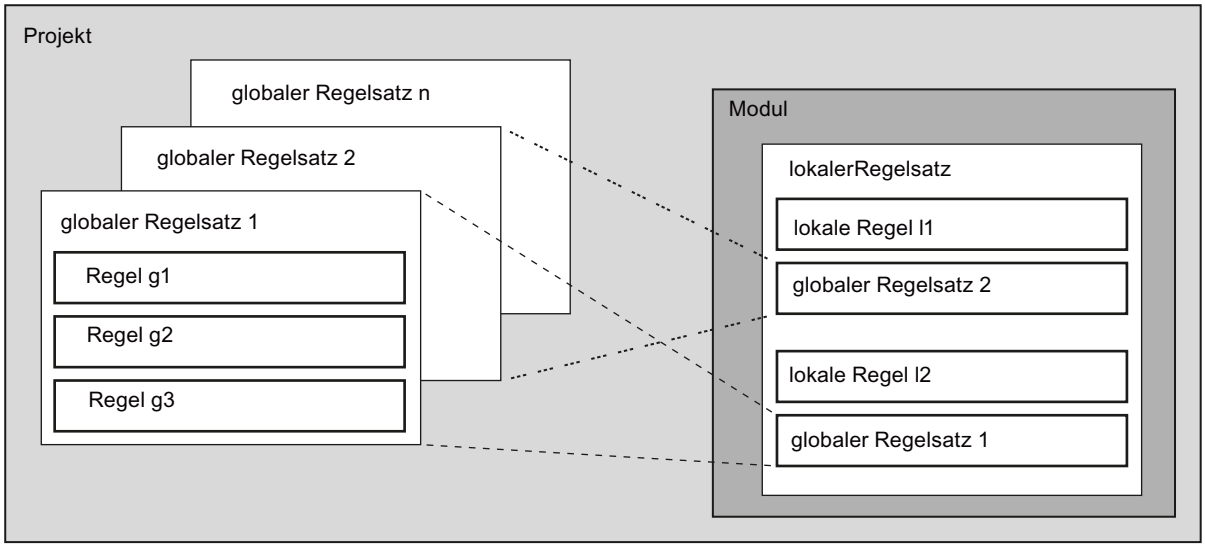

Projektierung

Bei der Projektierung von globalen Firewall-Regeln können Sie detaillierte Firewall-Einstellungen vornehmen. Sie können für einen einzelnen Teilnehmer einzelne Dienste freischalten oder für den Teilnehmer alle Dienste für den Zugriff auf die Station bzw. das Netz freigeben.

Wann sind globale IP- und MAC-Firewall-Regeln sinnvoll?

Globale Firewall-Regeln sind dann sinnvoll, wenn Sie für mehrere Security-Module identische Filterkriterien für die Kommunikation definieren wollen.

Hinweis

Zuweisung von Firewall-Regelsätzen mit inkompatiblen Firewall-Regeln

Es werden für ein Security-Modul nur die Firewall-Regeln aus Firewall-Regelsätzen korrekt übernommen, die das Security-Modul auch unterstützt. Eine in einem globalen Firewall-Regelsatz enthaltene Firewall-Regel mit der Richtung "Von: Extern" bzw. "Nach: Any" beinhaltet, wird beispielsweise einem CP 1628 nicht zugewiesen. Die restlichen Firewall-Regeln des globalen Firewall-Regelsatzes werden übernommen, sofern diese von dem CP 1628 unterstützt werden.

Globale Firewall-Regelsätze - Vereinbarungen

Globale Firewall-Regelsätze werden lokal genutzt

Folgende Vereinbarungen gelten bei der Erstellung eines globalen Firewall-Regelsatzes sowie bei der Zuweisung zu einem Modul:

- Projektierungsansicht
  Globale Firewall-Regelsätze werden in den globalen Security-Funktionen projektiert.
- Priorität
  Lokal definierte Regeln haben standardmäßig eine höhere Priorität als globale IP- und MAC-Firewall-Regelsätze. Neu zugewiesene globale IP- und MAC-Firewall-Regelsätze werden daher in der lokalen Regelliste zunächst unten eingefügt.
  Die Priorität kann durch Verändern der Platzierung in der Regelliste verändert werden.
- Regelsätze eingeben, ändern oder löschen
  Globale Firewall-Regelsätze sind in der lokalen Regelliste der Firewall-Regeln bei den Moduleigenschaften nicht editierbar. Sie können dort nur angezeigt und gemäß der gewünschten Priorität platziert werden.
  Eine einzelne Regel kann in den lokalen Security-Einstellungen nicht aus einem zugeordneten Regelsatz gelöscht werden. Es kann nur der komplette Regelsatz aus der lokalen Regelliste entfernt werden. Die Firewall-Regelsätze in den globalen Security-Funktionen bleiben davon unberührt.

Globale Firewall-Regelsätze anlegen

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Globale Firewall-Regelsätze" > "IP-Regelsätze" bzw. "MAC-Regelsätze".
   Ergebnis: Unter dem ausgewählten Eintrag werden die bisher erstellten IP-Regelsätze bzw. MAC-Regelsätze angezeigt.
2. Doppelklicken Sie auf den Eintrag "Neuen IP-Regelsatz hinzufügen" bzw. "Neuen MAC-Regelsatz hinzufügen".
   Ergebnis: Der erstellte Firewall-Regelsatz wird mit einer automatisch vergebenen Nummer unterhalb des Eintrags "IP-Regelsätze" bzw. "MAC-Regelsätze" angezeigt.
3. Doppelklicken Sie auf den erstellten Firewall-Regelsatz.
   Ergebnis: Im Arbeitsbereich wird der Zuweisungsdialog für Firewall-Regelsätze angezeigt. In der Klappliste "Regelsatz" ist der selektierte Firewall-Regelsatz ausgewählt. Diesem können die gewünschten Security-Module zugewiesen werden, siehe Kapitel Globale Firewall-Regelsätze zuweisen (Seite 397). In den lokalen Security-Einstellungen werden die projektierbaren Eigenschaften des Firewall-Regelsatzes angezeigt.

4. Geben Sie in den lokalen Security-Einstellungen im Eintrag "Allgemein" die folgenden Daten ein:
   – Name: Projektweit eindeutige Bezeichnung des Regelsatzes. Der Name erscheint nach der Zuweisung des Regelsatzes in der lokalen Regelliste des Security-Moduls.
   – Beschreibung (optional): Geben Sie eine Beschreibung für den globalen Regelsatz ein.
5. Tragen Sie in den lokalen Security-Einstellungen im Eintrag "Firewall-Regelsatz" die Firewall-Regeln in die Liste ein.
   Beachten Sie die Parameterbeschreibung in folgenden Kapiteln:
   Für IP-Regelsätze: IP-Paketfilter-Regeln definieren (Seite 417)
   Für MAC-Regelsätze: MAC-Paketfilter-Regeln definieren (Seite 422)
   Um zu den lokalen Security-Einstellungen eines anderen Firewall-Regelsatzes umzuschalten, wählen Sie diesen aus der Klappliste "Regelsatz" des Zuweisungsdialogs aus oder doppelklicken Sie auf den zugehörigen Eintrag in den globalen Security-Funktionen.

Ergebnis

Sie haben den globalen Firewall-Regelsatz angelegt und können diesen nun den gewünschten Security-Modulen zuweisen.
Beachten Sie die Beschreibungen in folgendem Kapitel:

Globale Firewall-Regelsätze zuweisen (Seite 397)

Globale Firewall-Regelsätze zuweisen

Voraussetzung

Sie haben für die Security-Module, die Sie einem Firewall-Regelsatz zuweisen möchten, den erweiterten Firewall-Modus aktiviert.

Vorgehensweise

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Globale Firewall-Regelsätze" > "Modul einem Firewall-Regelsatz zuweisen".
2. Wählen Sie aus der Klappliste "Regelsatz" den Regelsatz aus, dem Sie das Security-Modul zuweisen möchten.
   In der rechten Tabelle werden die Security-Module angezeigt, die Sie dem ausgewählten Firewall-Regelsatz zuweisen können. In der linken Tabelle werden die Security-Module angezeigt, die dem ausgewählten Firewall-Regelsatz bereits zugeordnet sind.
3. Wählen Sie im Bereich "Verfügbare Module" die Security-Module aus, die Sie dem ausgewählten Regelsatz zuweisen möchten.
4. Klicken Sie auf die Schaltfläche "<<", um die ausgewählten Module dem ausgewählten Regelsatz zuzuweisen.

Ergebnis

Der globale Regelsatz wird von den zugewiesenen Security-Modulen als lokaler Regelsatz verwendet und erscheint in den lokalen Security-Einstellungen automatisch am Ende der Liste der Firewall-Regeln.

Globale Firewall-Regelsätze exportieren und importieren

Bedeutung und Funktion

Globale IP-Regelsätze und globale MAC-Regelsätze können aus STEP 7 in das XLSX-Format exportiert und in STEP 7 importiert werden. Pro Firewall-Regelsatz wird beim Export eine XLSX-Datei erstellt.

Der Export und Import bietet die Möglichkeit, Firewall-Regelsätze zwischen unterschiedlichen STEP 7 Projekten auszutauschen. Exportierte Firewall-Regelsätze können zudem in Microsoft Excel® bearbeitet und im Anschluss wieder in STEP 7 importiert werden. Dies erleichtert die Durchführung von Massenänderungen.

Aus STEP 7 exportierte Firewall-Regelsätze sind mit dem Security Configuration Tool (SCT) weitgehend kompatibel. Aus dem Security Configuration Tool exportierte Firewall-Regelsätze sind mit STEP 7 vollständig kompatibel.

Hinweis

Einschränkungen beim Import von Firewall-Regelsätzen aus STEP 7 in das Security Configuration Tool

In SCT werden IPv6-Firewall-Regeln und ICMPv6-Dienste nicht unterstützt. Beim Import werden Firewall-Regeln, die als IPv6-Firewall-Regeln gekennzeichnet sind, sowie ICMPv6-Dienste ignoriert.

Für die Begrenzung der Übertragungsgeschwindigkeit sind in SCT maximal 100 Mbit/s zulässig. Beim Import werden Firewall-Regeln mit mehr als 100 Mbit/s ignoriert.

Firewall-Regelsätze aus STEP 7 exportieren

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Globale Firewall-Regelsätze".
2. Doppelklicken Sie auf den Eintrag "Übersicht".
   Ergebnis: Im Arbeitsbereich werden die in STEP 7 vorhandenen IP-Regelsätze und MAC-Regelsätze angezeigt.
3. Wenn Sie nicht alle Firewall-Regelsätze exportieren wollen, aktivieren Sie die Kontrollkästchen der zu exportierenden Firewall-Regelsätze.
4. Klicken Sie auf die Schaltfläche

5. Wählen Sie im Dialog "Firewall-Regelsätze exportieren" aus, ob alle Firewall-Regelsätze, nur die ausgewählten Firewall-Regelsätze oder alle von SCT unterstützten Firewall-Regelsätze exportiert werden sollen. Mit der dritten Option werden Firewall-Regelsätze, die Firewall-Regeln mit von SCT nicht unterstützten Kommunikationsrichtungen enthalten, vom Export ausgeschlossen.
6. Geben Sie einen Pfad an, unter welchem die Firewall-Regelsätze abgespeichert werden sollen, und klicken Sie auf die Schaltfläche "Exportieren".
   Ergebnis: Pro Firewall-Regelsatz wurde eine XLSX-Datei erstellt. Die XLSX-Dateien wurden nach den Namen der zugehörigen Firewall-Regelsätze sowie nach dem Datum und der Uhrzeit des Exports benannt.

Firewall-Regelsätze in STEP 7 importieren

Exportierte Firewall-Regelsätze können vor dem Import bearbeitet werden. Beachten Sie hierzu die Angaben im Abschnitt "Anforderungen an zu importierende Firewall-Regelsätze".

Gehen Sie wie folgt vor, um Firewall-Regelsätze in STEP 7 zu importieren:

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Globale Firewall-Regelsätze".
2. Doppelklicken Sie auf den Eintrag "Übersicht".
   Ergebnis: Im Arbeitsbereich werden die in STEP 7 vorhandenen IP-Regelsätze und MAC-Regelsätze angezeigt.
3. Klicken Sie auf die Schaltfläche
4. Wenn gleichnamige Firewall-Regelsätze beim Import überschrieben werden sollen, aktivieren Sie das Kontrollkästchen "Vorhandene Firewall-Regelsätze überschreiben".
   Anmerkung: Für den Abgleich zwischen den in STEP 7 vorhandenen Firewall-Regelsätzen und den XLSX-Dateien werden ausschließlich die Namen der Firewall-Regelsätze verwendet. Die Angaben zu Datum und Uhrzeit im Dateinamen der XLSX-Dateien werden ignoriert.
5. Wählen Sie die zu importierenden XLSX-Dateien aus. Eine Mehrfachauswahl ist möglich.
6. Klicken Sie auf die Schaltfläche "Importieren".
   Ergebnis: Die Firewall-Regelsätze der ausgewählten XLSX-Dateien wurden in STEP 7 importiert.

Aufbau exportierter IP-Regelsätze

Für jeden aus STEP 7 exportierten IP-Regelsatz wird eine XLSX-Datei mit den folgenden Tabellenblättern erstellt:

- IP Ruleset: Enthält die IP-Paketfilterregeln des IP-Regelsatzes
- IP Services: Enthält die IP-Dienste, die in IP-Paketfilterregeln des IP-Regelsatzes verwendet sind. IP-Dienste in verwendeten IP-Dienstgruppen werden hier ebenfalls angezeigt.

- ICMP Services: Enthält die ICMP-Dienste, die in IP-Paketfilterregeln des IP-Regelsatzes verwendet sind. ICMP-Dienste in verwendeten IP-Dienstgruppen werden hier ebenfalls angezeigt.
- IP Service Groups: Enthält die IP-Dienstgruppen, die in IP-Paketfilterregeln des IP-Regelsatzes verwendet sind. Für jede IP-Dienstgruppe werden die zugehörigen IP- und ICMP-Dienste angezeigt.

Aufbau exportierter MAC-Regelsätze

Für jeden aus STEP 7 exportierten MAC-Regelsatz wird eine XLSX-Datei mit den folgenden Tabellenblättern erstellt:

- MAC Ruleset: Enthält die MAC-Paketfilterregeln des MAC-Regelsatzes
- MAC Services: Enthält die MAC-Dienste, die in MAC-Paketfilterregeln des MAC-Regelsatzes verwendet sind. MAC-Dienste in verwendeten MAC-Dienstgruppen werden hier ebenfalls angezeigt.
- MAC Service Groups: Enthält die MAC-Dienstgruppen, die in MAC-Paketfilterregeln des MAC-Regelsatzes verwendet sind. Für jede MAC-Dienstgruppe werden die zugehörigen MAC-Dienste angezeigt.

Vereinbarungen für zu importierende Regelsätze

Folgende Vereinbarungen gelten für in STEP 7 zu importierende Firewall-Regelsätze:

- Die Dateien haben die Dateiendung *.XLSX und das zugehörige Office Open XML Format.
- Wenn der Dateiname der XLSX-Datei länger als 128 Zeichen ist, wird der Regelsatz in STEP 7 mit einem Standardnamen angelegt.
- Die Namen der Spalten innerhalb der Tabellenblätter dürfen nicht verändert werden, die Reihenfolge der Spalten darf jedoch verändert werden.
- Die Namen der Tabellenblätter dürfen nicht verändert werden, die Reihenfolge der Tabellenblätter darf jedoch verändert werden.

Erlaubte Parameterwerte für zu importierende IP-Regelsätze

Folgende Werte dürfen in die Tabellenspalten von XLSX-Dateien für IP-Regelsätze eingetragen werden. Groß- und Kleinschreibung wird bei der Prüfung auf korrekte Schreibweise nicht berücksichtigt.

Tabelle 1-16 Erlaubte Parameterwerte für Tabellenblatt "IP Ruleset"

Action	From	To	IPv6*	Source IP address	Destination IP Address	Service	Bandwidth	Logging	Stateful	Comment
• Accept • Drop • Reject	External	• Internal • DMZ • Tunnel • Any • Station • Backplane	• True • False	[Freitext mit max. 255 Zeichen]	[Freitext mit max. 255 Zeichen]	[Freitext mit max. 128 Zeichen]	[Mögliche Werte: 0.001…1000]	• True • False	• True • False	[Freitext mit max. 255 Zeichen]
	Internal	• External • Internal • DMZ • Tunnel • Any • Station								
	DMZ	• External • Internal • Tunnel • Any								
	Tunnel	• External • Internal • DMZ • Tunnel • Any • Station • Backplane								
	Any	• External								

Action	From	To	IPv6*	Source IP address	Destination IP Address	Service	Bandwidth	Logging	Stateful	Comment
		• Internal • DMZ • Backplane								
	Station	• External • Internal • Tunnel • Backplane								
	Backplane	• Any • External • Station • Tunnel								

* Diese Spalte ist optional. Wenn die Spalte beim Import in STEP 7 nicht vorhanden ist, wird in STEP 7 der Standardwert "False" verwendet.

Tabelle 1-17 Erlaubte Parameterwerte für Tabellenblatt "IP Services"

Name	Protocol	Source Port	Destination Port
[Freitext mit max. 128 Zeichen]	• UDP • TCP • All	[Freitext mit max. 32 Zeichen]	[Freitext mit max. 32 Zeichen]

Tabelle 1-18 Erlaubte Parameterwerte für Tabellenblatt "ICMP Services"

Name	ICMPv6*	Type	Code
[Freitext mit max. 128 Zeichen]	True	• PacketToBig • EchoRequest • EchoReply • MulticastListenerQuery • MulticastListenerReport • V2MulticastListenerReport • MulticastListenerDone • RouterSolicitation • RouterAdvertisement • NeighbourSolicitation • NeighbourAdvertisement • RedirectMessage • InverseDiscoverySolicitation • InverseDiscoveryAdvertisement • HomeDiscoveryRequest • HomeDiscoveryReply • MobilePrefixSolicitation • MobilePrefixAdvertisement • CertificationPathSolicitation • CertificatePathAdvertisement • IcmpMessagesUtilizedBySeamoby • MulticastRouterAdvertisement • MulticastRouterSolicitation • MulticastRouterTermination • FmIpV6 • RplControl	NoCode
		RouterRenumbering	• RouterRenumberingCommand • RouterRenumberingResult • SequenceNumberReset
		IcmpNodeInformationQuery	• ContainsIpV6

Name	ICMPv6*	Type	Code
			• ContainsName • ContainsIpV4
		IcmpInformationResponse	• Successful • RefuseToSupplyAnswer • UnknownQtypeOfQuery
		TimeExceeded	• HopLimitExceededInTransit • FragmentReassemblyTimeExceeded
		ParameterProblem	• ErroneousHeaderFieldEncountered • UnrecognizedNextHeaderTypeEncountered • UnrecognizedIpV6OptionEncountered
	False	• EchoReply • SourceQuench • AlternateHostAddress • EchoRequest • RouterSolicitation • TimestampReply • TimestampRequest • InformationRequest • InformationReply • AddressMaskReply • AddressMaskRequest • MobileHostRedirect • IpV6WhereAreYou • IpV6IAmHere • MobileRegistrationReply • MobileRegistrationRequest • Skip • Star	NoCode
		DestinationUnreachable	• Star • NetUnreachable • HostUnreachable • ProtocolUnreachable • PortUnreachable • FragmentationNeededAndDontFragmentWasSet • SourceRouteFailed

Name	ICMPv6*	Type	Code
			• DestinationNetworkUnknown • DestinationHostUnknown • SourceHostIsolated • CommunicationWithDestinationNetworkIsAdministrativelyProhibited • CommunicationWithDestinationHostIsAdministrativelyProhibited • DestinationNetworkUnreachableForTypeOfService • DestinationHostUnreachableForTypeOfService • CommunicationAdministrativelyProhibited • HostPrecedenceViolation • PrecedenceCutOffInEffect
		Redirect	• Star • DatagramForNetwork • DatagramForTheHost • DatagramForTheTypeOfServiceAndNetwork • DatagramForTheTypeOfServiceandHost
		RouterAdvertisement	• Star • NormalRouterAdvertisement • DoesNotRouteCommonTraffic
		TimeExceeded	• Star • TimeToLiveExceededInTransit • FragmentReassemblyTimeExceeded
		ParameterProblem	• Star • TheIpHeaderIsInvalid • ArequiredOptionIsMissing
		Traceroute	• Star • OutboundPacketSuccessfullyForwarded

Name	ICMPv6*	Type	Code
			• NoRouteForOutboundPacketThePacketWasDiscarded
		ConversionError	• Star • UnknownOrUnspecifiedError • DontConvertOptionPresent • UnknownMandatoryOptionPresent • KnownUnsupportedOptionPresent • UnsupportedTransportProtocol • OverallLengthExceeded • IpHeaderLengthExceeded • TransportProtocolGreaterThen255 • PortConversionOutOfRange • TransportHeaderLengthExceeded • BitRolloverMissingAndAckSet • UnknownMandatoryTransportOptionPresent
		Photuris	• Star • BadSpi • AuthenticationFailed • DecompressionFailed • DecryptionFailed • NeedAuthentication • NeedAuthorization

* Diese Spalte ist optional. Wenn die Spalte beim Import in STEP 7 nicht vorhanden ist, wird in STEP 7 der Standardwert "False" verwendet.

Tabelle 1-19 Erlaubte Parameterwerte für Tabellenblatt "IP Service Groups"

Name	Description	IP Services
[Freitext mit max. 128 Zeichen]	[Freitext mit max. 255 Zeichen]	[Freitext, Dienste werden mit Komma voneinander getrennt]

Erlaubte Parameterwerte für zu importierende MAC-Regelsätze

Folgende Werte dürfen in die Tabellenspalten von XLSX-Dateien für MAC-Regelsätze eingetragen werden. Groß- und Kleinschreibung wird bei der Prüfung auf korrekte Schreibweise nicht berücksichtigt.

Tabelle 1-20 Erlaubte Parameterwerte für Tabellenblatt "MAC Ruleset"

Action	From	To	Source MAC Address	Destination MAC Address	Service	Bandwidth	Logging	Comment
• Accept • Drop • Reject	External	• Internal • Tunnel • Any • Station	[MAC-Adresse im korrekten Format: xx-xx-xx-xx-xx-xx]	[MAC-Adresse im korrekten Format: xx-xx-xx-xx-xx-xx]	[Freitext mit max. 128 Zeichen]	[Mögliche Werte: 0.001… 1000]	• True • False	[Freitext mit max. 255 Zeichen]
	Internal	• External • Tunnel • Any						
	Tunnel	• External • Internal • Station						
	Any	• External • Internal						
	Station	• External • Tunnel						

Tabelle 1-21 Erlaubte Parameterwerte für Tabellenblatt "MAC Services"

Name	Protocol	DSAP	SSAP	CTRL	OUI	OUI Type
[Freitext mit max. 128 Zeichen]	• SNAP • PROFINET IO • ISO • [Hexadezimalwerte beginnend mit 0x]	[Hexadezimalwerte mit max. 2 Zeichen]	[Hexadezimalwerte mit max. 2 Zeichen]	[Hexadezimalwerte mit max. 2 Zeichen]	[Hexadezimalwerte mit max. 6 Zeichen]	[Hexadezimalwerte mit max. 4 Zeichen]

Tabelle 1-22 Erlaubte Parameterwerte für Tabellenblatt "MAC Service Groups"

Name	Description	MAC Services
[Freitext mit max. 128 Zeichen]	[Freitext mit max. 255 Zeichen]	[Freitext, Dienste werden mit Komma voneinander getrennt]

IP-Dienste

IP-Dienste definieren

So erreichen Sie diese Funktion

Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".

Vorgehensweise

Mit Hilfe der Definition von IP-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu.

Zusätzlich können Sie so definierte Dienste wiederum unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der Paketfilter-Regeln verwenden Sie dann diese Namen.

Parameter für IP-Dienste

Die Definition der IP-Dienste erfolgt über folgende Parameter:

Tabelle 1-23 IP-Dienste: Parameter

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Name	Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppenzusammenfassung verwendet wird. Die Namen vordefinierter Dienste können nicht geändert werden.	• Name muss mit einem Buchstaben beginnen. • Name darf keine Sonderzeichen enthalten. • Name darf nicht doppelt vorkommen.
Protokoll	Auswahl des Protokolltyps	• TCP • UDP • TCP+UDP • Alle

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Quell-Port	Es erfolgt eine Filterung anhand der hier angegebenen Portnummer; diese definiert den Dienstzugang beim Telegrammabsender.	Bei der Protokollauswahl "TCP +UDP" ist keine Portangabe möglich. Beispiele: *: Port wird nicht geprüft 20 bzw. 21: FTP-Service
Ziel-Port	Es erfolgt eine Filterung anhand der hier angegebenen Portnummer; diese definiert den Dienstzugang beim Telegrammempfänger.	Bei der Protokollauswahl "TCP +UDP" ist keine Portangabe möglich. Beispiele: *: Port wird nicht geprüft TCP 80: Web-HTTP-Service TCP 102: S7-Protokoll

ICMP-Dienste definieren

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "ICMP".

Vorgehensweise

Mit Hilfe der Definition von ICMP-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu.

Zusätzlich können Sie so definierte Dienste unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der Paketfilter-Regeln verwenden Sie dann diese Namen.

Parameter für ICMP-Dienste

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Name	Frei definierbarer Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppenzusammenfassung verwendet wird. Die Namen vordefinierter ICMPv6-Dienste sind nicht änderbar.	• Name darf nicht doppelt vorkommen
ICMPv6	Wenn Sie dieses Kontrollkästchen aktivieren, wird der ICMP-Dienst als ICMPv6-Dienst deklariert und Sie können einen ICMPv6-spezifischen Typ und Code für den Dienst auswählen. Ein ICMPv6-Dienst kann nur in der Firewall-Regel eines Security-Moduls verwendet werden, welches IPv6 unterstützt.	• aktiviert • deaktiviert (Standard)
Typ	Typ der ICMPv4- bzw. ICMPv6-Message.	Wenn das Kontrollkästchen "ICMPv6" deaktiviert ist, sind ICMPv4-spezifische Typen auswählbar. Wenn das Kontrollkästchen aktiviert ist, sind ICMPv6-spezifische Typen auswählbar.
Code	Code des ICMP-Typs.	Werte sind abhängig vom gewählten Typ.

Dienstgruppen anlegen

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "Dienstgruppen".

Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können. IPv4- und IPv6-Dienste können in einer Dienstgruppe zusammengefasst werden.

Im geöffneten Register "Dienstgruppen" legen Sie Gruppen an. Anschließend ordnen Sie im Register "Gruppen-Management" die Dienste einer Gruppe zu.

Gehen Sie so vor

1. Legen Sie in diesem Register zunächst Gruppen mit einem für Ihre Anforderungen zweckmäßigen Namen und einem optionalen Beschreibungstext an.
2. Wählen Sie dann das Register "Gruppen-Management". Dort können Sie den hier definierten Gruppen die zuvor festgelegten IP-Dienste zuordnen.

Dienstgruppen verwalten

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "Gruppen-Management".

Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können. IPv4- und IPv6-Dienste können in einer Dienstgruppe zusammengefasst werden.

In dem Register "Gruppen-Management " ordnen Sie einer ausgewählten Dienstgruppe Dienste zu, die Sie zuvor im Register "Dienstgruppen" angelegt haben.

Gehen Sie so vor

1. Wählen Sie in diesem Register zunächst über die Klappliste "Dienstgruppen" eine zuvor im Register "Dienstgruppen" angelegte Gruppe aus.
2. Ordnen Sie dann aus dem rechten Auswahlfeld "Verfügbare Dienste" die gewünschten Dienste der Gruppe zu.

MAC-Dienste

MAC-Dienste definieren

So erreichen Sie diese Funktion

Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".

Bedeutung

Mit Hilfe der Definition von MAC-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben einen Namen und ordnen diesem die Dienstparameter zu.

Zusätzlich können Sie so definierte Dienste unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der globalen oder lokalen Paketfilter-Regeln verwenden Sie dann diese Namen.

Parameter für MAC-Dienste

Eine MAC-Dienst-Definition wird durch protokollspezifische MAC-Parameter gebildet:

Tabelle 1-24 MAC-Dienste Parameter

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Name	Frei definierbarer Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppenzusammenfassung verwendet wird.	• Name muss mit einem Buchstaben beginnen. • Name darf keine Sonderzeichen enthalten. • Name darf nicht doppelt vorkommen.
Protokoll	Name des Protokolltyps: • ISO ISO bezeichnet Telegramme mit folgenden Eigenschaften: Lengthfield <= 05DC (hex), DSAP= userdefined SSAP= userdefined CTRL= userdefined • SNAP SNAP bezeichnet Telegramme mit folgenden Eigenschaften: Lengthfield <= 05DC (hex), DSAP=AA (hex), SSAP=AA (hex), CTRL=03 (hex), OUI=userdefined, OUI-Type=userdefined • PROFINET IO • Alternativ kann eine Protokollnummer eingegeben werden. Die Protokolleingaben 0800 (hex) und 0806 (hex) werden nicht akzeptiert, da diese Werte für IP- bzw. ARP-Telegramme gelten.	• ISO • SNAP • PROFINET IO • 0x (Eingabe der Protokollnummer)
DSAP	Destination Service Access Point: LLC-Empfänger-Adresse	
SSAP	Source Service Access Point: LLC-Sender-Adresse	
CTRL	LLC Control Field	
OUI	Organizationally Unique Identifier (die ersten 3 Bytes der MAC-Adresse = Hersteller Identifizierung)	
OUI-Typ	Protokoll-Typ/-Identifizierung	

Hinweis

Verarbeitung für S7-CPs

Es werden nur Einstellungen zu ISO-Frames mit DSAP=SSAP=FE (hex) verarbeitet. Andere Frame-Typen sind für S7-CPs nicht relevant und werden daher schon vor der Bearbeitung durch die Firewall verworfen.

Spezielle Einstellungen für SIMATIC NET Dienste

Verwenden Sie für die Filterung spezieller SIMATIC NET Dienste bitte die folgenden Protokoll-Einstellungen:

- DCP:
  PROFINET IO
- SiClock:
  OUI= 08 00 06 (hex), OUI-Type= 01 00 (hex)

Dienstgruppen anlegen

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".
2. Selektieren Sie das Register "Dienstgruppen".

Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können.

Im geöffneten Register "Dienstgruppen" legen Sie Gruppen an. Anschließend ordnen Sie im Register "Gruppen-Management" die Dienste einer Gruppe zu.

Gehen Sie so vor

1. Legen Sie in diesem Register zunächst Gruppen mit einem für Ihre Anforderungen zweckmäßigen Namen und einem optionalen Beschreibungstext an.
2. Wählen Sie dann das Register "Gruppen-Management". Dort können Sie den hier definierten Gruppen die zuvor festgelegten MAC-Dienste zuordnen.

Dienstgruppen verwalten

So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".
2. Selektieren Sie das Register "Gruppen-Management".

Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können.

In dem Register "Gruppen-Management " ordnen Sie einer ausgewählten Dienstgruppe Dienste zu, die Sie zuvor im Register "Dienstgruppen" angelegt haben.

Gehen Sie so vor

1. Wählen Sie in diesem Register zunächst über die Klappliste "Dienstgruppen" eine zuvor im Register "Dienstgruppen" angelegte Gruppe aus.
2. Ordnen Sie dann aus dem rechten Auswahlfeld "Verfügbare Dienste" die gewünschten Dienste der Gruppe zu.

Übersicht Lokale Firewall-Regeln

Bedeutung

Lokale Firewall-Regeln werden in den lokalen Security-Einstellungen eines Security-Moduls projektiert und gelten nur für dieses Security-Modul. Nach dem Aktivieren der Firewall-Funktionalität können Sie entweder vordefinierte Firewall-Regeln verwenden oder Firewall-Regeln im erweiterten Firewall-Modus definieren.

Vordefinierte Firewall-Regeln verwenden

Hier greifen Sie auf einfache, vordefinierte Regeln zurück. Sie können nur dienstspezifische Regeln freischalten. Die freigeschalteten Dienste sind für alle Teilnehmer in der angegebenen Richtung zulässig. Detaillierte Informationen zur Definition von Firewall-Regeln in diesem Dialogfenster entnehmen Sie den folgenden, modulspezifischen Kapiteln:

- Für SCALANCE S: AUTOHOTSPOT
- Für S7-300-/S7-400-/PC-CPs: AUTOHOTSPOT
- Für S7-1200-/S7-1500-CPs: AUTOHOTSPOT

Die für CPs projektierbare Kommunikationsrichtung "Station" enthält den Zugriff auf den CP sowie den Zugriff auf die CPU über den CP.

Firewall-Regeln im erweiterten Firewall-Modus definieren

Im erweiterten Firewall-Modus können Sie detaillierte Firewall-Einstellungen vornehmen. Sie können für einen einzelnen Teilnehmer einzelne Dienste freischalten oder für den Teilnehmer alle Dienste für den Zugriff auf die Station bzw. das Netz freigeben. Den erweiterten Firewall-Modus aktivieren Sie über das Kontrollkästchen "Firewall im erweiterten Modus aktivieren". Die Editoren zur Projektierung von IP-Regeln und MAC-Regeln im erweiterten Firewall-Modus öffnen Sie über die zugehörigen Schaltflächen unter "Firewall > IP-Regeln" bzw. "Firewall > MAC-Regeln". Welche Projektierungsmöglichkeiten in den Editoren zur Verfügung stehen, siehe folgende Kapitel:

Für IP-Paketfilter-Regeln: IP-Paketfilter-Regeln definieren (Seite 417)

Für MAC-Paketfilter-Regeln: MAC-Paketfilter-Regeln definieren (Seite 422)

Hinweis

Kein Deaktivieren des erweiterten Firewall-Modus möglich

Wenn Sie den erweiterten Firewall-Modus einmal aktiviert haben, können Sie diesen nicht mehr deaktivieren.

Übernahme vordefinierter Firewall-Regeln in den erweiterten Firewall-Modus

Beim Umschalten in den erweiterten Firewall-Modus werden aktivierte vordefinierte Firewall-Regeln sowie Logging-Einstellungen übernommen. Ist für ein SCALANCE S Modul beispielsweise das Kontrollkästchen "Erlaube S7-Protokoll" in der Richtung "Von Extern nach Intern" aktiviert, dann wird im erweiterten Firewall-Modus automatisch eine Accept-Regel mit dieser Kommunikationsrichtung und dem Dienst "S7" angelegt.

Wenn für S7-1200-/S7-1500-CPs vordefinierte Logging-Einstellungen vorhanden sind, werden diese im erweiterten Firewall-Modus nicht als Firewall-Regeln dargestellt, sondern unter dem Eintrag "Firewall > Log-Einstellungen" editierbar angezeigt.

Mengengerüst

Anzahl Firewall-Regeln (Erweiterter Firewall-Modus)	
SCALANCE S ab V3	256 maximal
S7-1200-/S7-1500-CPs	256 maximal
CP 343-1 Adv. / 443-1 Adv.	226 maximal
CP 1628	226 maximal

IP-Paketfilter-Regeln definieren

Bedeutung

Mit IP-Paketfilter-Regeln filtern Sie nach IP-Telegrammen wie z. B. TCP, UDP oder ICMP.

Innerhalb einer Paketfilter-Regel können Sie auf Definitionen der IP-Dienste zurückgreifen.

Zugewiesene globale Firewall-Regeln sowie für NAT-/NAPT-Regeln automatisch erzeugte Firewall-Regeln werden als Regelsätze dargestellt.

Bedienelemente der Menüleiste

Die Bedienelemente der Menüleiste haben die folgenden Funktionen:

Bedienelement	Funktion
	Firewall aktivieren Aktiviert den Editor für IP-Regeln. Bei aktiviertem Editor sind die zugehörigen Regeln projektierbar und werden beim Laden auf das Modul übertragen.
	Regel oberhalb einfügen Fügt eine Regel oberhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen oberhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regel unterhalb einfügen Fügt eine Regel unterhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen unterhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach oben verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach oben. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach unten verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach unten. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regelsätze ausklappen Stellt vorhandene Regelsätze ausgeklappt dar.
	Regelsätze einklappen Stellt vorhandene Regelsätze eingeklappt dar.
	Separieren Wandelt die ausgewählte Zeile in separate Regel(n) um. Wenn die erste Zeile eines Regelsatzes ausgewählt wird, werden alle Regeln des Regelsatzes in separate Regeln umgewandelt. Wenn eine im Regelsatz enthaltene Regel ausgewählt wird, wird diese in eine separate Regel umgewandelt. Zusätzlich werden die nachfolgenden Regeln des Regelsatzes in einen separaten Regelsatz umgewandelt.

IP-Paketfilter-Regeln eintragen

Nach dem Aktivieren des Editors sind die IP-Regeln über folgende Parameter projektierbar:

Tabelle 1-25 IP-Regeln: Parameter

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Aktion	Zulassungsfestlegung (Freigabe/Sperre)	• Accept Telegramme gemäß Definition zulassen. • Drop Telegramme gemäß Definition sperren. • Reject Die Telegramme werden abgewiesen und der Absender erhält eine entsprechende Meldung. Für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt und danach manuell angepasst wurden: • Accept* • Drop* Wenn Sie automatisch angelegte Verbindungsregeln verändern, werden diese bei der Wahl der Option "*" nicht erneut von STEP 7 erzeugt und überschrieben.
Von / Nach	Auswahl der Kommunikationsrichtungen, für die die Regel gelten soll.	Werden in separaten Abschnitten beschrieben. • Für SCALANCE S Module: IP-Paketfilter-Richtungen SCALANCE S (Seite 531) • Für S7-300-/S7-400-/PC-CPs: IP-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs (Seite 563) • Für S7-1200-/S7-1500-CPs: IP-Paketfilter-Richtungen S7-1200-/S7-1500-CPs (Seite 579) Die Kommunikationsrichtung "Station" enthält den Zugriff auf den CP sowie den Zugriff auf die CPU über den CP. Die Kommunikationsrichtung "Any" bezieht alle zum Ladezeitpunkt vom Security-Modul unterstützten Firewall-Richtungen ein. Die Kommunikationsrichtungen "Station", "Rückwandbus" (nur für CP 1543-1 ab V2.1 mit aktivierter Option "IP-Routing zwischen Kommunikationsmodulen") und "Tunnel" sind standardmäßig als vertrauenswürdig eingestuft. Für die Kommunikation zwischen vertrauenswürdigen Richtungen müssen keine Firewall-Regeln projektiert werden.

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Quell-IP-Adresse	Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Sender die hier angegebene IP-Adresse hat. Wenn Sie keine IP-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Von" ausgewählt haben.	Weitere Informationen zu IP-Adressen finden Sie im Abschnitt IP-Adressen in IP-Paketfilter-Regeln (Seite 426). **Projektierungsmöglichkeiten im Ghost-Modus (nur für SCALANCE S602 ab V3.1):** Bei aktiviertem Ghost-Modus wird die IP-Adresse des internen Teilnehmers zur Laufzeit dynamisch vom Security-Modul ermittelt. Je nach ausgewählter Richtung können Sie eine der folgenden Optionen in der Spalte "Quell-IP-Adresse" (bei Richtung "Von Intern nach Extern") bzw. in der Spalte "Ziel-IP-Adresse" (bei Richtung "Von Extern nach Intern") auswählen: • IP-Adresse des internen Teilnehmers: Die IP-Adresse des internen Teilnehmers wird durch den SCALANCE S in die Firewall-Regel eingefügt • Limited broadcast: Die Broadcast-IP-Adresse 255.255.255.255 wird durch den SCALANCE S in die Firewall-Regel eingefügt. • Directed broadcast: Die Broadcast-IP-Adresse des SCALANCE S Netzes wird durch den SCALANCE S in die Firewall-Regel eingefügt. Ein Directed broadcast kann auch über Router ins Zielnetz weitergeleitet werden. • Multicast: Das Multicast-Adressband 224.0.0.0 /24 wird durch den SCALANCE S in die Firewall-Regel eingefügt. Nach der Auswahl dieser Option kann alternativ eine bestimmte Multicast-IP-Adresse aus dem Multicast-Adressband angegeben werden.
Ziel-IP-Adresse	Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Empfänger die hier angegebene IP-Adresse hat. Wenn Sie keine IP-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Nach" ausgewählt haben.	
Dienst	Auswahl des zu verwendenden IP-/ICMP-Dienstes oder der Dienstgruppe. Hinweis: Stellen Sie vor der Verwendung eines vordefinierten Diensts sicher, dass dessen vordefinierte Parameter Ihre Anforderungen erfüllen. Achten Sie hierbei insbesondere auf die vordefinierten Quell- und Ziel-Ports.	Die Klappliste bietet die vordefinierten und projektierten IP-Dienste und Dienstgruppen zur Auswahl an. Diese werden in den globalen Security-Funktionen unter "Firewall > Dienste > Dienste für IP-Regeln definieren" angezeigt.
Übertragungsgeschwindigkeit (Mbit/s)	Begrenzung der Übertragungsgeschwindigkeit Kann nur eingegeben werden, wenn bei Aktion "Accept" ausgewählt ist. Ein Paket passiert die Firewall, wenn die Accept-Regel zutrifft und die zulässige Übertragungsgeschwindigkeit für diese Regel noch nicht überschritten worden ist.	CP 343-1 Adv. / CP 443-1 Adv., S7-1200-/S7-1500-CPs, CP 1628: 0.001 ... 100 Mbit/s SCALANCE S: 0.001 ... 1000 Mbit/s Für globale und benutzerspezifische Regeln: 0.001 ... 1000 Mbit/s Für IP-Regeln mit Richtung "Rückwandbus": 0.001 ... 1 Mbit/s Hinweis: Wenn in einer Firewall-Regel von S7-1200-/S7-1500-CPs die Richtung "Von Tunnel nach Station" projektiert wird, kann keine Übertragungsgeschwindigkeit angegeben werden. Für CP 1543-1 V2.1 kann nur eine Übertragungsgeschwindigkeit angegeben werden, wenn für eine der beiden Richtungen "Any" projektiert ist. Dies gilt nicht für die Kombination aus den Richtungen "Any" und "Station".

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Logging	Ein- und Ausschalten des Loggings für diese Regel. Bei aktiviertem Logging gelten die Einstellungen für das Paketfilter-Logging, die in den lokalen Security-Einstellungen projektiert werden.	• aktiviert • deaktiviert (Standard)
Nummer	Automatisch vergebene Nummer der Regel. Die Nummern werden beim Verschieben von Regeln neu ermittelt.	
Stateful	Wenn dieses Kontrollkästchen für eine IP-Regel mit der Aktion "Accept" deaktiviert ist, werden durch Pakete, auf die die Accept-Regel zutrifft, keine Firewall-States erzeugt. Durch Firewall-States werden die Antworten auf zugelassene Pakete automatisch zugelassen. Kann nur angepasst werden, wenn bei Aktion "Accept" ausgewählt ist. Die Projektierung von IP-Regeln ohne Firewall-States ist nur für SCALANCE S Module ab Firmware V3 möglich. Sollen die Antworten auf Pakete, die die Firewall gemäß solcher IP-Regeln passiert haben, ebenfalls zugelassen werden, so müssen für diese Antworten zusätzliche IP-Regeln projektiert werden.	Für Accept-Regeln von SCALANCE S ≥ V3: • aktiviert (Standard) • deaktiviert Für Accept-Regeln von SCALANCE S < V3 und CPs: • aktiviert (Standard) Anmerkung: Für CPs ist das Kontrollkästchen nicht vorhanden, implizit wird für Accept-Regeln von CPs jedoch stets die aktivierte Option verwendet. Für Drop-Regeln: • deaktiviert (Standard)
Kommentar	Platz für eigene Erläuterung der Regel	Ist ein Kommentar mit "AUTO" gekennzeichnet, wurde er für eine automatische Verbindungsregel angelegt. Für von Ihnen angelegte Regeln ist die Kommentareingabe optional.

Tabelle 1-26 Bedeutung der Einträge des Kontextmenüs

Eintrag des Kontextmenüs	Bedeutung
Regel oberhalb einfügen	Fügen Sie hiermit eine Regel oberhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein.
Regel unterhalb einfügen	Fügen Sie hiermit eine Regel unterhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein.

Eintrag des Kontextmenüs	Bedeutung
Löschen	Löschen Sie hiermit die markierte Regel oder den markierten Regelsatz. Hinweise zum Löschen eines global definierten, lokal zugewiesenen Regelsatzes: Indem Sie den Regelsatz hier löschen, wird nur die Zuordnung zum Security-Modul aufgehoben.
Als globalen Regelsatz speichern (nur für lokale Firewall-Regeln)	Kopiert die markierte Firewall-Regel(n) und fügt sie als globalen Firewall-Regelsatz in die globalen Security-Funktionen ein. Die aktuell für das Security-Modul projektierte Firewall-Konfiguration bleibt von diesem Vorgang unberührt.
Nach oben verschieben	Bewegen Sie hiermit die markierte Regel oder den markierten Regelsatz um eine Position in der Liste nach oben. Alternativ können Sie die markierte Regel oder den markierten Regelsatz per Drag & Drop verschieben. Eine Multiselektion ist hierbei möglich. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer höheren Priorität behandelt.
Nach unten verschieben	Bewegen Sie hiermit die markierte Regel oder den markierten Regelsatz um eine Position in der Liste nach unten. Alternativ können Sie die markierte Regel oder den markierten Regelsatz per Drag & Drop verschieben. Eine Multiselektion ist hierbei möglich. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer niedrigeren Priorität behandelt.
Dienst für IP-Regeln definieren	Öffnen Sie hiermit den Dialog, in dem die IP-Dienste und Dienstgruppen verwaltet werden.

Erweiterter IP-Regel-Editor

Nach Auswahl einer IP-Regel erscheint im Inspektorfenster unter "Eigenschaften > Allgemein" ein erweiterter IP-Regel-Editor zur vereinfachten Projektierung von IP-Regeln. Änderungen in diesem Editor werden in die ausgewählte IP-Regel übernommen.

Projektierung von IP-Regeln

Adressbereiche können in den Eingabefeldern der Spalten "Von" und "Bis" projektiert werden. Wenn mehrere Adressbereiche für eine IP-Regel projektiert werden, muss die Anzahl der Adressen jeweils übereinstimmen. Nach der Projektierung von Adressbereichen wird aus der ausgewählten IP-Regel von STEP 7 ein IP-Regelsatz erstellt, der für jede der Adressen eine entsprechende IP-Regel enthält.

Die anderen Einstellungen, die in der Tabelle "IP-Regeln: Parameter" beschrieben werden, nehmen Sie unter "Zusätzliche Einstellungen" vor.

IP-Regeln und IP-Regelsätze kopieren

Zusätzlich können Sie selektierte IP-Regeln und IP-Regelsätze kopieren. Für das Kopieren eines IP-Regelsatzes muss dessen erste Zeile ausgewählt werden.

MAC-Paketfilter-Regeln definieren

Bedeutung

Mit MAC-Paketfilter-Regeln filtern Sie nach MAC-Telegrammen.

Innerhalb einer Paketfilter-Regel können Sie auf Definitionen der MAC-Dienste zurückgreifen.

Zugewiesene globale Firewall-Regeln werden als Regelsätze dargestellt.

Bedienelemente der Menüleiste

Die Bedienelemente der Menüleiste haben die folgenden Funktionen:

Bedienelement	Funktion
	Firewall aktivieren Aktiviert den Editor für MAC-Regeln. Bei aktiviertem Editor sind die zugehörigen Regeln projektierbar und werden beim Laden auf das Modul übertragen.
	Regel oberhalb einfügen Fügt eine Regel oberhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen oberhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regel unterhalb einfügen Fügt eine Regel unterhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen unterhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach oben verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach oben. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach unten verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach unten. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regelsätze ausklappen Stellt vorhandene Regelsätze ausgeklappt dar.
	Regelsätze einklappen Stellt vorhandene Regelsätze eingeklappt dar.

MAC-Paketfilter-Regeln eintragen

Nach dem Aktivieren des Editors sind die MAC-Regeln über folgende Parameter projektierbar:

Tabelle 1-27 MAC-Regeln: Parameter

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Aktion	Zulassungsfestlegung (Freigabe/ Sperre)	• Accept Telegramme gemäß Definition zulassen. • Drop Telegramme gemäß Definition sperren. • Reject Die Telegramme werden abgewiesen und der Absender erhält eine entsprechende Meldung. Für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt und danach manuell angepasst wurden: • Accept* • Drop* Wenn Sie automatisch angelegte Verbindungsregeln verändern, werden diese bei der Wahl der Option "*" nicht erneut von STEP 7 erzeugt und überschrieben.
Von / Nach	Auswahl der Kommunikationsrichtungen, für die die Regel gelten soll.	Werden in separaten Abschnitten beschrieben. • Für SCALANCE S Module: MAC-Paketfilter-Richtungen SCALANCE S (Seite 531) • Für S7-300-/S7-400-/PC-CPs: MAC-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs (Seite 564) • Für S7-1200-/S7-1500-CPs: MAC-Paketfilter-Richtungen S7-1200-/S7-1500-CPs (Seite 579) Die Kommunikationsrichtung "Station" enthält den Zugriff auf den CP sowie den Zugriff auf die CPU über den CP. Die Kommunikationsrichtungen "Station" und "Tunnel" sind standardmäßig als vertrauenswürdig eingestuft. Für die Kommunikation zwischen vertrauenswürdigen Richtungen müssen keine Firewall-Regeln projektiert werden.
Quell-MAC-Adresse	Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Sender die hier angegebene MAC-Adresse hat. Wenn Sie keine MAC-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Von" ausgewählt haben.	MAC-Adresse im korrekten Format
Ziel-MAC-Adresse	Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Empfänger die hier angegebene MAC-Adresse hat. Wenn Sie keine MAC-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Nach" ausgewählt haben.	

Parameter	Bedeutung/Kommentar	Auswahlmöglichkeiten / Wertebereiche
Dienst	Auswahl des zu verwendenden MAC-Dienstes oder der Dienstgruppe. Hinweis: Stellen Sie vor der Verwendung eines vordefinierten Diensts sicher, dass dessen vordefinierte Parameter Ihre Anforderungen erfüllen.	Die Klappliste bietet die vordefinierten und projektierten Dienste und Dienstgruppen zur Auswahl an. Diese werden in den globalen Security-Funktionen unter "Firewall > Dienste > Dienste für MAC-Regeln definieren" angezeigt.
Übertragungsgeschwindigkeit (Mbit/s)	Begrenzung der Übertragungsgeschwindigkeit Kann nur eingegeben werden, wenn bei Aktion "Accept" ausgewählt ist. Ein Paket passiert die Firewall, wenn die Accept-Regel zutrifft und die zulässige Übertragungsgeschwindigkeit für diese Regel noch nicht überschritten worden ist.	CP 343-1 Adv. / CP 443-1 Adv., S7-1200-/S7-1500-CPs: 0.001 ... 100 Mbit/s CP 1628 und SCALANCE S: 0.001 ... 1000 Mbit/s Für globale Regeln: 0.001 ... 1000 Mbit/s Hinweis: Wenn in einer Firewall-Regel für S7-1200-/S7-1500-CPs die Richtung "Von Tunnel nach Station" projektiert wird, kann keine Übertragungsgeschwindigkeit angegeben werden.
Logging	Ein- und Ausschalten des Logging für diese Regel. Bei aktiviertem Logging gelten die Einstellungen für das Paketfilter-Logging, die in den lokalen Security-Einstellungen projektiert werden.	• aktiviert • deaktiviert (Standard)
Nummer	Automatisch vergebene Nummer der Regel. Die Nummern werden beim Verschieben von Regeln neu ermittelt.	
Kommentar	Platz für eigene Erläuterung der Regel	Ist ein Kommentar mit "AUTO" gekennzeichnet, wurde er für eine automatische Verbindungsregel angelegt. Für von Ihnen angelegte Regeln ist die Kommentareingabe optional.

Tabelle 1-28 Bedeutung der Menübefehle

Schaltfläche	Bedeutung
Löschen	Löschen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz. Hinweise zum Löschen eines global definierten, lokal zugewiesenen Regelsatzes: Indem Sie den Regelsatz hier löschen, wird nur die Zuordnung zum Security-Modul aufgehoben.
Als globalen Regelsatz speichern (nur für lokale Firewall-Regeln)	Kopiert die markierte Firewall-Regel(n) und fügt sie als globalen Firewall-Regelsatz in die globalen Security-Funktionen ein. Die aktuell für das Security-Modul projektierte Firewall-Konfiguration bleibt von diesem Vorgang unberührt.
Nach oben verschieben	Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach oben. Alternativ können Sie die markierte Regel oder den markierten Regelsatz per Drag & Drop verschieben. Eine Multiselektion ist hierbei möglich. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer höheren Priorität behandelt.

Schaltfläche	Bedeutung
Nach unten verschieben	Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach unten. Alternativ können Sie die markierte Regel oder den markierten Regelsatz per Drag & Drop verschieben. Eine Multiselektion ist hierbei möglich. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer niedrigeren Priorität behandelt.
Dienst für MAC-Regeln definieren	Öffnen Sie hiermit den Dialog, in dem die MAC-Dienste und Dienstgruppen verwaltet werden.

IP-Adressen in IP-Paketfilter-Regeln

Eingabe von IP-Adressen in IP-Paketfilter-Regeln

In IP-Paketfilter-Regeln haben Sie folgende Möglichkeiten, IP-Adressen einzugeben:

- keine Angabe
  Die Regel gilt für alle IP-Adressen.
- eine IP-Adresse
  Die Regel gilt genau für die angegebene Adresse.
- Adressband
  Die Regel gilt für alle im Adressband erfassten IP-Adressen.
  Ein Adressband wird definiert, indem die Anzahl der gültigen Bit-Stellen in der IP-Adresse angegeben wird und zwar in der Form: [IP-Adresse]/[Anzahl der zu berücksichtigenden Bits]
  - [IP-Adresse]/24 bedeutet demnach, dass nur die höchstwertigen 24 Bit der IP-Adresse in der Filterregel berücksichtigt werden. Das sind die ersten drei Stellen der IP-Adresse.
  - [IP-Adresse]/25 bedeutet, dass nur die ersten drei Stellen und das höchstwertige Bit der vierten Stelle der IP-Adresse in der Filterregel berücksichtigt werden.
- Adressbereich
  Für die Quell-IP-Adresse kann ein Adressbereich in folgendem Format eingegeben werden: [Start-IP-Adresse]-[End-IP-Adresse]

IPv4-Adressen

Eine IPv4-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind.

Tabelle 1-29 Beispiele für Adressbänder bei IPv4-Adressen

Quell-IP-Adresse bzw. Ziel-IP-Adresse	Adressband		Anzahl Adressen *)
	von	bis	
192.168.0.0/**16**	192.168.0.0	192.168.255.255	65.536
192.168.10.0/**24**	192.168.10.0	192.168.10.255	256
192.168.10.0/**25**	192.168.10.0	192.168.10.127	128
192.168.10.0/**26**	192.168.10.0	192.168.10.63	64

Quell-IP-Adresse bzw. Ziel-IP-Adresse	Adressband		Anzahl Adressen *)
	von	bis	
192.168.10.0/**27**	192.168.10.0	192.168.10.31	32
192.168.10.0/**28**	192.168.10.0	192.168.10.15	16
192.168.10.0/**29**	192.168.10.0	192.168.10.7	8
192.168.10.0/**30**	192.168.10.0	192.168.10.3	4
*) Hinweis: Beachten Sie, dass die Netzwerkadresse und die Broadcastadresse in einem Adressband nicht als IP-Adressen von Netzwerkteilnehmern zur Verfügung stehen.			

IPv6-Adressen

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vier Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt.

Beispiel: fd00:ffff:ffff:ffff:ffff:ffff:2f33:8f21

Regeln / Vereinfachungen:

- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.
  Beispiel: Statt 2001:0db8:2426:08d3:1457:8a2e:0070:7344 kann auch 2001:db8:2426:8d3:1457:8a2e:70:7344 notiert werden.
- Wenn ein oder mehrere Blöcke den Wert 0 (bzw. 0000) haben, ist eine verkürzte Schreibweise möglich.
  Beispiel: Statt 2001:0db8:0:0:0:0:1428:57ab kann auch 2001:db8::1428:57ab notiert werden.
  Damit die Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden.
- Dezimalnotation mit Punkt-Schreibweise
  Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.
  Beispiel: Die IPv6-Adresse fd00::ffff.125.1.0.1 ist äquivalent zu fd00::ffff:7d01:1.
- Adressbandnotation in IP-Paketfilterregeln: Analog zu IPv4-Adressen können IPv6-Adressen ebenfalls in Form von Adressbändern notiert werden.
  Beispiel: Der Eintrag "2001:0db8:85a3:08d3:1319:8a2e:0:0/96" umfasst alle IPv6-Adressen von 2001:0db8:85a3:08d3:1319:8a2e:0:0 bis 2001:0db8:85a3:08d3:1319:8a2e:ffff:ffff.

Modulspezifische Log-Einstellungen vornehmen

Log-Einstellungen Übersicht

Modulspezifische Funktion

Einige Logging-Funktionen sind nur für bestimmte Security-Module verfügbar.

Log-Einstellungen in der Konfiguration

Die hier vorgenommenen Log-Einstellungen werden mit der Konfiguration in das Modul geladen und werden mit dem Start des Security-Moduls wirksam.

Sie können die projektierten Paketfilter-Log-Einstellungen in den Online-Funktionen bei Bedarf einschränken. Haben Sie beispielsweise IP- und MAC-Logging projektiert, können Sie in den Online-Funktionen einstellen, dass lediglich IP-Logging angezeigt wird.

Aufzeichnungsverfahren und Ereignisklassen

Sie können hier festlegen, welche Daten aufgezeichnet werden sollen. Dadurch aktivieren Sie die Aufzeichnung bereits mit dem Laden der Konfiguration in das Security-Modul.

Außerdem wählen Sie in der Konfiguration eine oder beide der möglichen Aufzeichnungsverfahren:

- Lokales Logging
- Netzwerk Syslog

Das Security-Modul kennt für beide Aufzeichnungsverfahren jeweils die drei folgenden Arten von Ereignissen:

- Paketfilter-Ereignisse
- Audit-Ereignisse
- System-Ereignisse

Lokales Logging konfigurieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "Lokaler Log-Speicher"

Lokales Logging konfigurieren

Tabelle 1-30 Lokales Logging - Einstellungen für Log-Ereignisse

Log-Ereignis	Bedeutung	Bemerkungen
Paketfilter-Log (Firewall)	Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur solche Datenpakete geloggt, auf die eine projektierte Paket-Filter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paketfilter-Regel aktiviert ist. Aus der Klappliste "Aufgezeichnete Pakete" können Sie die Menge aufgezeichneter Datenpakete festlegen: • "Alle Pakete": Es werden die Datenpakete aufgezeichnet, auf die eine projektierte Firewall-Regel zutrifft. Zusätzlich werden die Antwortpakete auf solche Pakete aufgezeichnet, welche die Firewall gemäß einer projektierten Accept-Regel passiert haben. • "Zustandserzeugende Pakete" (nur für SCALANCE S ab V3): Es werden ausschließlich die Datenpakete aufgezeichnet, auf die eine projektierte Firewall-Regel zutrifft und die initial einen Zustand in der Firewall erzeugen. Datenpakete, welche die Firewall passieren, indem sie diesen Firewall-Zustand nutzen, werden nicht aufgezeichnet.	Paketfilter-Logdaten sind nicht remanent Die Daten werden in einem flüchtigen Speicher des Security-Moduls abgelegt und stehen deshalb nach einem Ausschalten der Spannungsversorgung nicht mehr zur Verfügung. Zur remanenten Speicherung können Sie die im "Online & Diagnose"-Dialog angezeigten Log-Daten zusätzlich in einer Datei abspeichern.
Audit-Log	Das Logging für Audit-Ereignisse ist immer aktiviert. Die Speicherung erfolgt immer im Umlaufspeicher. Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf, beispielsweise Benutzeraktionen wie das Ein- oder Ausschalten des Paket-Logging oder das Laden von Konfigurationen auf das Security-Modul.	Audit-Logdaten sind remanent Die Daten werden in einem remanenten Speicher des Security-Moduls abgelegt und stehen deshalb auch nach einem Ausschalten der Spannungsversorgung noch zur Verfügung. **Hinweis für CPs:** Die Audit-Logdaten sind bei CPs nicht remanent. Zur Sicherung der Daten sollte deshalb ein Syslog-Server verwendet werden.

Log-Ereignis	Bedeutung	Bemerkungen
System-Log	Der System-Log zeichnet automatisch fortlaufend System-Ereignisse, wie z. B. den Start eines Prozesses oder den gescheiterten Login-Versuch eines Benutzers, auf. Zur Konfiguration des Ereignisfilters und der Leitungsdiagnose wählen Sie den Eintrag "Log-Einstellungen" > "System-Ereignisse".	System-Logdaten sind nicht remanent Die Daten werden in einem flüchtigen Speicher des Security-Moduls abgelegt und stehen deshalb nach einem Ausschalten der Spannungsversorgung nicht mehr zur Verfügung. Zur remanenten Speicherung können Sie die im "Online & Diagnose"-Dialog angezeigten Log-Daten zusätzlich in einer Datei abspeichern.

Tabelle 1-31 Lokales Logging - Speicherverfahren für die Daten-Aufzeichnung

Speicherverfahren	Bedeutung
Umlaufspeicher	Bei Erreichen des Puffer-Endes wird die Aufzeichnung am Puffer-Anfang mit dem Überschreiben der ältesten Einträge fortgesetzt.
Linearer Speicher	Die Aufzeichnung stoppt, wenn der Puffer voll ist.

System-Ereignisse konfigurieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "System-Ereignisse"

Filterung der System-Ereignisse

Sie stellen in diesem Dialog für die System-Ereignisse eine Filterebene ein. Standardmäßig sind die folgenden Werte eingestellt:

- SCALANCE S: Ebene 3 (Error)
- CP: Ebene 3 (Error)

Es werden diejenigen System-Ereignisse aufgezeichnet, die die gleiche Priorität wie die ausgewählte Filterebene haben und die eine höhere Priorität als die ausgewählte Filterebene haben. Je kleiner der Wert der Filterebene ist, desto höher ist ihre Priorität und desto weniger Ereignisse werden aufgezeichnet.

Die ausgewählte Filterebene muss die gleiche oder eine niedrigere Priorität haben als die für die Leitungsdiagnose eingestellte Severity, siehe Tabelle "Parametrierung der Leitungsdiagnose" (nicht für CPs).

Empfehlung: Wählen Sie die Filterebene "Error" oder eine Filterebene mit einer höheren Priorität, um die Aufzeichnung von allgemeinen, nicht kritischen Ereignissen zu unterbinden.

Hinweis für CP

Wählen Sie für CPs nur Ebene 3 oder Ebene 6 aus, da für CPs nur Ereignisse dieser Ebenen erzeugt werden.

- Bei Auswahl von Ebene 3 werden die Fehlermeldungen der Ebenen 0 bis 3 ausgegeben.
- Bei Auswahl von Ebene 6 werden die Fehlermeldungen der Ebenen 0 bis 6 ausgegeben.

Eigenschaften der System-Ereignisse – Leitungsdiagnose (nur für SCALANCE S)

Die Leitungsdiagnose erzeugt ein spezielles System-Ereignis. Dabei wird bei einem einstellbaren Prozentsatz fehlerhafter Telegramme ein System-Ereignis erzeugt. Diesem System-Ereignis wird die in diesem Dialog eingestellte Severity und Facility zugewiesen.

Tabelle 1-32 Parametrierung der Leitungsdiagnose

Funktion / Option / Parameter	Bedeutung
Aktivieren	Aufzeichnung ein- und ausschalten.
Limit	Einstellbarer Prozentsatz fehlerhafter Telegramme, bei dessen Erreichen ein System-Ereignis ausgelöst wird.
Facility	Sie wählen hier aus der Klappliste eine Facility, mit der das zur Aufzeichnung erfasste System-Ereignis gekennzeichnet wird.
Severity	Über die Severity gewichten Sie die System-Ereignisse der Leitungsdiagnose im Verhältnis zur Severity der übrigen System-Ereignisse.

Hinweis

Severity der System-Ereignisse der Leitungsdiagnose

Den System-Ereignissen der Leitungsdiagnose darf keine geringere Severity zugewiesen werden, als Sie für den Filter eingestellt haben. Bei einer geringeren Severity können diese Ereignisse den Filter nicht passieren und werden nicht aufgezeichnet.

Netzwerk-Syslog konfigurieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "Netzwerk-Syslog".

Netzwerk-Syslog konfigurieren

Tabelle 1-33 Netzwerk-Syslog - Grundeinstellungen

Option / Parameter	
Netzwerk-Syslog aktivieren	Die Übermittlung von Logging-Ereignissen zum Syslog-Server ein- und ausschalten.
Syslog-Server	Geben Sie hier die IP-Adresse des Syslog-Servers ein. Für SCALANCE S Module ab V4 kann alternativ ein FQDN eingegeben werden. Der Syslog-Server muss vom Security-Modul aus über die angegebene Adresse erreichbar sein, ggf. über die Router-Projektierung in den lokalen Security-Einstellungen unter dem Eintrag "Routing". Wenn der Syslog-Server nicht erreichbar ist, werden keine Syslog-Nachrichten versendet. Sie können diesen Betriebszustand anhand entsprechender System-Meldungen erkennen. Um das Versenden der Syslog-Informationen erneut zu aktivieren, müssen Sie ggf. die Routing-Informationen aktualisieren und einen Neustart des Security-Moduls veranlassen.
Ereignisklassen aktivieren	Aktivieren Sie die Klassen der Ereignisse, die an den Syslog-Server übermittelt werden sollen. Paketfilter- sowie Audit-Ereignisse können Sie über die Severity nach Schweregrad und über die Facility nach Herkunft klassifizieren.
Modulname	Der Modulname wird an dieser Stelle angezeigt und kann nicht geändert werden.

Hinweis

Ungesicherte Übertragung von Logging-Ereignissen

Logging-Ereignisse werden an Syslog-Server im Klartext übertragen. Dies sollte beim Einsatz von Syslog-Servern berücksichtigt werden.

Tabelle 1-34 Netzwerk-Syslog - Einstellungen für Log-Ereignisse

Log-Ereignis	Projektierung	Bemerkungen
Paketfilter-Ereignisse (Firewall)	Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur solche Datenpakete geloggt, auf die eine projektierte Paket-Filter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paket-Filter-Regel aktiviert ist. Durch die Einstellung von Facility und Severity lassen sich Syslog-Meldungen entsprechend ihrer Herkunft und ihres Schweregrades klassifizieren. Die Zuordnung erfolgt über Klapplisten. Jedem Ereignis wird die Severity und Facility zugewiesen, die Sie hier einstellen.	Welchen Wert Sie hier wählen, hängt von der Auswertung im Syslog-Server ab. Lassen Sie den Standardwert "default" eingestellt, wird durch das Security-Modul festgelegt, mit welcher Kombination aus Facility und Severity das Ereignis angezeigt wird.
Audit-Ereignisse	Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf, beispielsweise Benutzeraktionen wie das Ein- oder Ausschalten des Paket-Logging oder das Laden von Konfigurationen auf das Security-Modul. Die Zuordnung der Severity und Facility erfolgt über Klapplisten. Jedem Ereignis wird die Severity und Facility zugewiesen, die Sie hier einstellen.	Welchen Wert Sie hier für Severity und Facility wählen, hängt von der Auswertung im Syslog-Server ab. Lassen Sie den Standardwert "default" eingestellt, wird durch das Security-Modul festgelegt, mit welcher Kombination aus Facility und Severity das Ereignis angezeigt wird.
System-Ereignisse	Der System-Log zeichnet automatisch fortlaufend System-Ereignisse, wie z. B. den Start eines Prozesses oder den gescheiterten Login-Versuch eines Benutzers, auf.	Zur Konfiguration des Ereignisfilters und der Leitungsdiagnose wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "System-Ereignisse".

Security-Modul als Router

Übersicht zu Routing-Einstellungen

Bedeutung

Indem Sie das Security-Modul im Routing-Modus betreiben, werden die Netze an der internen und externen Schnittstelle zu separaten Subnetzen. Die DMZ-Schnittstelle (nur SCALANCE S623/S627-2M) wird unabhängig von der Betriebsart im Routing-Modus angebunden. Im Routing-Modus werden die Telegramme weitergeleitet, die an eine im jeweiligen Subnetz vorhandene IP-Adresse gerichtet sind. Darüber hinaus gelten die für die jeweilige Übertragungsrichtung getroffenen Firewall-Regeln.

Zusätzlich haben Sie folgende Möglichkeiten:

- Spezifische Routen einstellen - einstellbar in den lokalen Security-Einstellungen unter "Routing" (nur für SCALANCE S), siehe Routen festlegen (Seite 533).
- Standard-Router verwenden - einstellbar in den lokalen Security-Einstellungen unter "Externe Schnittstelle [P1] rot", "Interne Schnittstelle [P2] grün" bzw. "DMZ-Schnittstelle [P3] gelb" (nur SCALANCE S623/S627-2M), siehe IP-Adressparameter projektieren (Seite 505).
  Pro Security-Modul kann maximal ein Standard-Router verwendet werden.
- NAT/NAPT-Routing - einstellbar in den lokalen Security-Einstellungen unter "NAT / NAPT" (nur für SCALANCE S und CP 343-1 Adv. / CP 443-1 Adv.).

Routing-Modus aktivieren (nur für SCALANCE S Module nötig)

Für diese Betriebsart müssen Sie in den lokalen Security-Einstellungen eine interne IP-Adresse und eine interne Subnetzmaske für die Adressierung des Routers am internen Subnetz projektieren. Alle nicht zu einem Subnetz gehörenden Netzwerkanfragen werden durch das Security-Modul in ein anderes Subnetz weitergeleitet.

Hinweis: Im Gegensatz zum Bridge-Modus des Security-Moduls gehen VLAN-Tags im Routing-Modus verloren.

1. Wählen Sie in den lokalen Security-Einstellungen unter "Betriebsart" die Option "Routing-Modus".
2. Tragen Sie in den lokalen Security-Einstellungen unter "Interne Schnittstelle [P2] grün" > "Ethernet-Adressen" eine interne IP-Adresse und eine interne Subnetzmaske für die Adressierung des Routers am internen Subnetz in die Eingabefelder ein.

Übersicht zu NAT/NAPT

Voraussetzungen

- Das Security-Modul befindet sich im Routing-Modus oder die DMZ-Schnittstelle (nur SCALANCE S623 / S627-2M) ist aktiviert.
- Da für NAT-/NAPT-Regeln automatisch Firewall-Regeln erzeugt werden, die die Kommunikation in der projektierten Adressumsetzungsrichtung freigeben, muss für das Security-Modul der erweiterte Firewall-Modus sowie die Firewall im Editor für IP-Regeln aktiviert sein. Weitere Informationen finden Sie in Kapitel Zusammenhang zwischen NAT-/NAPT-Router und Firewall (Seite 449)

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "NAT / NAPT".
3. Rufen Sie den entsprechenden Editor über die Schaltfläche "Editor öffnen" auf.
4. Projektieren Sie je nach Anforderung eine Adressumsetzung gemäß NAT (Network Address Translation) oder NAPT (Network Address Port Translation).

Adressumsetzung mit NAT (Network Address Translation)

NAT ist ein Verfahren zur Adressumsetzung zwischen zwei Adressräumen. Hauptaufgabe ist die Umsetzung von privaten IP-Adressen in öffentliche, d. h. in IP-Adressen, die im Internet verwendet und auch geroutet werden. Dadurch wird erreicht, dass die IP-Adressen des internen Netzes nach außen im externen Netz nicht bekannt werden. Die internen Teilnehmer sind im externen Netz nur über die in der Adressumsetzungsliste (NAT-Tabelle) festgelegten externen IP-Adressen sichtbar. Handelt es sich bei der externen IP-Adresse nicht um die Adresse des Security-Moduls und ist die interne IP-Adresse eindeutig, wird dies als 1:1 NAT bezeichnet. Beim 1:1 NAT wird die interne Adresse ohne Portumsetzung auf diese externe Adresse umgesetzt. Ansonsten handelt es sich um n:1 NAT.

Adressumsetzung mit NAPT (Network Address Port Translation)

Die Adressumsetzung bei NAPT verändert die Zieladresse und den Zielport in einer Kommunikationsbeziehung (Portweiterleitung).

Umgesetzt werden Telegramme, die vom externen Netz oder vom DMZ-Netz kommen und an die IP-Adresse des Security-Moduls gerichtet sind. Ist der Ziel-Port des Telegramms identisch mit einem der Werte, der in der Spalte "Quell-Port" angegeben ist, so ersetzt das Security-Modul die Ziel-IP-Adresse und den Ziel-Port, wie in der entsprechenden Zeile der NAPT-Tabelle angegeben. Bei der Rückantwort setzt das Security-Modul als Quell-IP-Adresse und als Quell-Port die Werte ein, die beim Initialtelegramm als Ziel-IP-Adresse bzw. Ziel-Port stehen.

Der Unterschied zu NAT liegt darin, dass bei diesem Protokoll auch Ports umgesetzt werden können. Es gibt keine 1:1 Umsetzung der IP-Adresse. Vielmehr existiert nur noch eine öffentliche IP-Adresse, die durch den Zusatz von Portnummern an eine Reihe von privaten IP-Adressen umgesetzt wird.

Adressumsetzung in VPN-Tunneln

Adressumsetzungen mit NAT/NAPT können auch für Kommunikationsbeziehungen durchgeführt werden, welche über VPN-Tunnel aufgebaut werden. Dies wird für die Verbindungspartner vom Typ SCALANCE S612 / S623 / S627-2M ab V4 unterstützt.

Weitere Informationen zu Adressumsetzungen in VPN-Tunneln finden Sie in folgenden Kapiteln:

- NAT-/NAPT-Routing (Seite 436)
- Adressumsetzung mit NAT/NAPT in VPN-Tunneln (Seite 449)

Konsistenzprüfung - diese Regeln müssen Sie beachten

Beachten Sie unter anderem folgende Regeln, um konsistente Einträge zu erhalten:

- Die IP-Adresse der internen Schnittstelle darf nicht in der NAT-/NAPT-Tabelle verwendet werden.
- Eine IP-Adresse, die in der NAT/NAPT-Adressumsetzungsliste verwendet wird, darf keine Multicast-Adresse und keine Broadcast-Adresse sein.

- Die für die NAPT-Umsetzung vergebenen externen Ports liegen im Bereich > 0 und ≤ 65535. Port 123 (NTP), Port 443 (HTTPS), Port 514 (Syslog), Port 161 (SNMP), Ports 67+68 (DHCP) und Ports 500+4500 (IPsec) sind davon ausgeschlossen, sofern die jeweiligen Dienste auf dem Security-Modul aktiviert sind.
- Die externe IP-Adresse des Security-Moduls bzw. die IP-Adresse der DMZ-Schnittstelle darf in der NAT-Tabelle nur für die Aktion "Source-NAT" verwendet werden.
- Duplikatsprüfung in der NAT-Tabelle
  Eine externe IP-Adresse bzw. eine IP-Adresse im DMZ-Netz, die mit Richtung "Destination-NAT", "Source-NAT + Destination-NAT" oder "Double-NAT" verwendet wird, darf in jeder angegebenen Richtung nur einmal verwendet werden.
- Duplikatsprüfung in der NAPT-Tabelle: Eine Quell-Portnummer darf für jede Schnittstelle nur einmal eingetragen sein.
- Interne NAPT-Ports können im Bereich > 0 und ≤ 65535 liegen.

Siehe auch

Übersicht zu Routing-Einstellungen (Seite 433)

NAT-/NAPT-Routing

Adressumsetzung mit NAT

Folgende Bedienelemente stehen in der Menüleiste zur Verfügung:

Bedienelement	Funktion
	NAT aktivieren Aktiviert den Editor für NAT-Regeln. Bei aktiviertem Editor sind die zugehörigen Regeln projektierbar und werden beim Laden auf das Modul übertragen. Da für NAT-Regeln automatisch IP-Regeln erzeugt werden, die die Kommunikation in der projektierten Adressumsetzungsrichtung freigeben, kann der Editor für NAT-Regeln nur bei aktivem erweiterten Firewall-Modus sowie bei aktiver IP-Firewall aktiviert werden.
	Regel oberhalb einfügen Fügt eine Regel oberhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen oberhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regel unterhalb einfügen Fügt eine Regel unterhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen unterhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach oben verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach oben. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.

Bedienelement	Funktion
	Nach unten verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach unten. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regelsätze ausklappen Stellt vorhandene Regelsätze ausgeklappt dar.
	Regelsätze einklappen Stellt vorhandene Regelsätze eingeklappt dar.
	Separieren Wandelt die ausgewählte Zeile in separate Regel(n) um. Wenn die erste Zeile eines Regelsatzes ausgewählt wird, werden alle Regeln des Regelsatzes in separate Regeln umgewandelt. Wenn eine im Regelsatz enthaltene Regel ausgewählt wird, wird diese in eine separate Regel umgewandelt. Zusätzlich werden die nachfolgenden Regeln des Regelsatzes in einen separaten Regelsatz umgewandelt.

Nach dem Anlegen von NAT-Regeln werden die zugehörigen Firewall-Regeln erzeugt und im Erweiterten Firewall Modus angezeigt, siehe Kapitel:

Zusammenhang zwischen NAT-/NAPT-Router und Firewall (Seite 449)

Wenn für die externe Schnittstelle oder die DMZ-Schnittstelle PPPoE aktiviert ist, kann bei der Aktion "Destination-NAT" die externe Schnittstelle bzw. die DMZ-Schnittstelle nicht als Nach-Richtung ausgewählt werden. Bei der Projektierung der Aktion "Source-NAT" kann die IP-Adresse im Eingabefeld "Quell-Umsetzung" nicht eingegeben werden, da diese dynamisch zur Laufzeit ermittelt wird.

Mögliche Adressumsetzungsaktionen für NAT

In den folgenden Tabellen sind die Eingabemöglichkeiten zur Adressumsetzung mit NAT dargestellt.

Aktion "Destination-NAT" - "Redirect"

Die Aktion "Destination-NAT" kann in folgender Richtung durchgeführt werden:

- Extern nach Intern

Wenn die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) aktiviert ist, kann die Aktion "Destination-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Extern nach DMZ
- DMZ nach Intern
- DMZ nach Extern

Wenn sich das SCALANCE S Modul (nur SCALANCE S612/S623/S627-2M ab V4) in einer VPN-Gruppe befindet und die Tunnel-Schnittstelle aktiviert ist, kann die Aktion "Destination-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Tunnel nach Intern
- Tunnel nach Extern
- Tunnel nach DMZ (nur bei aktivierter DMZ-Schnittstelle)

Für die Richtung "Extern nach Intern" gilt beispielsweise: Die Ziel-IP-Adresse eines vom externen Netz kommenden Telegramms wird auf Übereinstimmung mit der IP-Adresse geprüft, die im Eingabefeld "Ziel-IP-Adresse" angegeben ist. Bei Übereinstimmung wird das Telegramm in das interne Netz weitergeleitet, indem die Ziel-IP-Adresse des Telegramms durch die IP-Adresse ersetzt wird, die im Eingabefeld "Ziel-Umsetzung" angegeben ist. Der Zugriff von Extern nach Intern über die externe IP-Adresse ist möglich.

Die folgende Tabelle zeigt das Eingabeschema für die Aktion "Destination-NAT".

Feld	Mögliche Eingaben	Bedeutung
Quell-IP-Adresse	Für diese Aktion nicht relevant.	-
Quell-Umsetzung	Für diese Aktion nicht relevant.	-
Ziel-IP-Adresse	IP-Adresse im Quell-Netz	Ziel-IP-Adresse im Quell-Netz, über die auf eine IP-Adresse im Ziel-Netz zugegriffen werden soll. Die Ziel-IP-Adresse darf nicht mit der IP-Adresse des Security-Moduls im Quell-Netz übereinstimmen. Stimmt in einem Telegramm die Ziel-IP-Adresse mit der eingegebenen Adresse überein, wird die Adresse durch die entsprechende IP-Adresse im Ziel-Netz ausgetauscht. Die angegebene Ziel-IP-Adresse wird zur Alias-Adresse. Dies bedeutet, dass die angegebene IP-Adresse zusätzlich als IP-Adresse an der ausgewählten Schnittstelle registriert wird. Stellen Sie sicher, dass mit der Alias-Adresse kein IP-Adresskonflikt im Netzwerk besteht. Die Alias-IP-Adressen eines Security-Moduls werden unter dem Eintrag "Alias-IP-Adressen" der jeweiligen Schnittstelle angezeigt.
Ziel-Umsetzung	IP-Adresse im Ziel-Netz	Die Ziel-IP-Adresse wird durch die hier angegebene IP-Adresse ersetzt.
Nr.	-	Von STEP 7 vergebene, fortlaufende Nummer, die zur Bezugnahme auf die Firewall-Regel verwendet wird, welche von STEP 7 für die NAT-Regel erzeugt wird.

Aktion "Source-NAT" - "Masquerading"

Die Aktion "Source-NAT" kann in folgender Richtung durchgeführt werden:

- Intern nach Extern

Wenn die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) aktiviert ist, kann die Aktion "Source-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Intern nach DMZ
- Extern nach DMZ
- DMZ nach Extern

Wenn sich das SCALANCE S Modul (nur SCALANCE S612/S623/S627-2M ab V4) in einer VPN-Gruppe befindet und die Tunnel-Schnittstelle aktiviert ist, kann die Aktion "Source-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Intern nach Tunnel
- Extern nach Tunnel
- DMZ nach Tunnel (nur bei aktivierter DMZ-Schnittstelle)

Für die Richtung "Intern nach Extern" gilt beispielsweise: Die Quell-IP-Adresse eines vom internen Netz kommenden Telegramms wird auf Übereinstimmung mit der IP-Adresse geprüft, die im Eingabefeld "Quell-IP-Adresse" angegeben ist. Bei Übereinstimmung wird das Telegramm mit der im Eingabefeld "Quell-Umsetzung" angegebenen externen IP-Adresse als neue Quell-IP-Adresse in das externe Netz weitergeleitet. Am externen Netz wirkt die externe IP-Adresse.

Die folgende Tabelle zeigt das Eingabeschema für die Aktion "Source-NAT".

Feld	Mögliche Eingaben	Bedeutung
Quell-IP-Adresse	IP-Adresse im Quell-Netz	Die Quell-IP-Adresse des angegebenen Teilnehmers wird durch die im Eingabefeld "Quell-Umsetzung" angegebene IP-Adresse ersetzt.
	IP-Adressbereich / IP-Adressband im Quell-Netz	Die IP-Adressen des IP-Adressbereichs / des IP-Adressbands werden durch die im Eingabefeld "Quell-Umsetzung" angegebene IP-Adresse ersetzt.
	*	Die IP-Adressen aller Teilnehmer im Quell-Netz werden durch die im Eingabefeld "Quell-Umsetzung" angegebene IP-Adresse ersetzt.
Quell-Umsetzung	IP-Adresse im Ziel-Netz	Eingabe der IP-Adresse, die als neue Quell-IP-Adresse verwendet werden soll. Handelt es bei der hier eingegebenen IP-Adresse nicht um die IP-Adresse des Security-Moduls, wird diese zur Alias-Adresse. Dies bedeutet, dass die angegebene Adresse zusätzlich als IP-Adresse an der ausgewählten Schnittstelle registriert wird. Stellen Sie sicher, dass mit der Alias-Adresse kein IP-Adresskonflikt im Netzwerk besteht. Die Alias-IP-Adressen eines Security-Moduls werden unter dem Eintrag "Alias-IP-Adressen" der jeweiligen Schnittstelle angezeigt.
Ziel-IP-Adresse	Für diese Aktion nicht relevant.	-
Ziel-Umsetzung	Für diese Aktion nicht relevant.	-
Nr.	-	Von STEP 7 vergebene, fortlaufende Nummer, die zur Bezugnahme auf die Firewall-Regel verwendet wird, welche von STEP 7 für die NAT-Regel erzeugt wird.

Hinweis

Sie können für alle von einem Quell-Netz in ein Ziel-Netz gehenden Telegramme eine Adressumsetzung auf die Modul-IP-Adresse im Ziel-Netz projektieren. Zusätzlich wird vom Security-Modul an jedes Telegramm eine Port-Nummer vergeben. Es handelt sich hierbei um eine n:1-NAT-Adressumsetzung, bei welcher mehrere IP-Adressen des Quell-Netzes auf eine IP-Adresse des Ziel-Netzes umgesetzt werden.

Geben Sie beispielsweise für die Richtung "Intern nach Extern" die folgenden Parameter ein:

- Aktion: "Source-NAT"
- Von: "Intern"
- Nach "Extern"
- Quell-IP-Adresse: "*"
- Quell-Umsetzung: Externe IP-Adresse des Security-Moduls

Aktion "Source-NAT + Destination-NAT" - "1:1-NAT"

Die Aktion "Source-NAT + Destination-NAT" kann in folgender Richtung durchgeführt werden:

- Intern nach Extern

Wenn die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) aktiviert ist, kann die Aktion "Source-NAT + Destination-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Intern nach DMZ
- Extern nach DMZ
- DMZ nach Extern

Wenn sich das SCALANCE S Modul (nur SCALANCE S612/S623/S627-2M ab V4) in einer VPN-Gruppe befindet und die Tunnel-Schnittstelle aktiviert ist, kann die Aktion "Source-NAT + Destination-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Extern nach Tunnel
- Intern nach Tunnel
- DMZ nach Tunnel (nur bei aktivierter DMZ-Schnittstelle)

Für die Richtung "Intern nach Extern" gilt beispielsweise: Beim Zugriff von Intern nach Extern wird die Aktion "Source-NAT" durchgeführt. Beim Zugriff von Extern nach Intern wird die Aktion "Destination-NAT" durchgeführt.

Die folgende Tabelle zeigt das Eingabeschema für die Aktion "Source-NAT + Destination-NAT":

Feld	Mögliche Eingaben	Bedeutung
Quell-IP-Adresse	IP-Adresse im Quell-Netz	Die Projektierung wird immer in Source-NAT-Richtung angegeben. Die IP-Adressen der Destination-NAT-Richtung werden daraufhin automatisch von STEP 7 eingefügt.
Quell-Umsetzung	IP-Adresse im Ziel-Netz	
Ziel-IP-Adresse	Für diese Aktion nicht relevant.	
Ziel-Umsetzung	Für diese Aktion nicht relevant.	
Nr.	-	Von STEP 7 vergebene, fortlaufende Nummer, die zur Bezugnahme auf die Firewall-Regeln verwendet wird, welche von STEP 7 für die NAT-Regel erzeugt werden.

Aktion "Double-NAT"

Die Aktion "Double-NAT" kann für SCALANCE S Module in folgenden Richtungen durchgeführt werden:

- Intern nach Extern
- Extern nach Intern

Wenn die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) aktiviert ist, kann die Aktion "Double-NAT" zusätzlich in folgenden Richtungen durchgeführt werden:

- Intern nach DMZ
- Extern nach DMZ
- DMZ nach Intern
- DMZ nach Extern

In jeder Richtung findet immer Source- und Destination-NAT zugleich statt.
Für die Richtung "Extern nach Intern" gilt beispielsweise: Beim Zugriff von Extern nach Intern wird die Quell-IP-Adresse des externen Teilnehmers ausgetauscht (Source-NAT). Zudem erfolgt der Zugriff auf das interne Netz über die im Eingabefeld "Ziel-IP-Adresse" angegebene externe IP-Adresse (Destination-NAT).

Sie können diese Aktion beispielsweise dann verwenden, wenn für ein Gerät, auf das mit Hilfe von Destination-NAT zugegriffen wird, ein anderer Standard-Router als das Security-Modul eingetragen ist. Antworttelegramme von diesem Gerät werden dann nicht an den eingetragenen Standard-Router, sondern an die zugehörige Schnittstelle des Security-Moduls gesendet.

Die folgende Tabelle zeigt das Eingabeschema für die Aktion "Double-NAT":

Feld	Mögliche Eingaben	Bedeutung
Quell-IP-Adresse	IP-Adresse im Quell-Netz	IP-Adresse des Teilnehmers im Quell-Netz
Quell-Umsetzung	-	Die Source-NAT-Adressumsetzung erfolgt immer auf die IP-Adresse des Security-Moduls im Ziel-Netz. Das Eingabefeld "Quell-Umsetzung" ist deshalb nicht projektierbar.

Feld	Mögliche Eingaben	Bedeutung
Ziel-IP-Adresse	IP-Adresse im Quell-Netz	Ziel-IP-Adresse im Quell-Netz, über die auf eine IP-Adresse im Ziel-Netz zugegriffen werden soll. Stimmt in einem Telegramm die Ziel-IP-Adresse mit der eingegebenen IP-Adresse überein, wird die IP-Adresse durch die im Eingabefeld "Ziel-Umsetzung" angegebene IP-Adresse ausgetauscht. Handelt es sich bei der hier eingegebenen IP-Adresse nicht um die IP-Adresse des Security-Moduls, wird diese zur Alias-Adresse. Dies bedeutet, dass die angegebene Adresse zusätzlich als IP-Adresse an der ausgewählten Schnittstelle registriert wird. Stellen Sie sicher, dass mit der Alias-Adresse kein IP-Adresskonflikt im Netzwerk besteht. Die Alias-IP-Adressen eines Security-Moduls werden unter dem Eintrag "Alias-IP-Adressen" der jeweiligen Schnittstelle angezeigt.
Ziel-Umsetzung	IP-Adresse im Ziel-Netz	Die Ziel-IP-Adresse wird durch die hier angegebene IP-Adresse ersetzt.
Nr.	-	Von STEP 7 vergebene, fortlaufende Nummer, die zur Bezugnahme auf die Firewall-Regel verwendet wird, welche von STEP 7 für die NAT-Regel erzeugt wird.

Erweiterter NAT-Editor

Nach Auswahl einer NAT-Regel erscheint im Inspektorfenster unter "Eigenschaften > Allgemein" ein erweiterter NAT-Editor zur vereinfachten Projektierung von Adressbereichen für Ports und IP-Adressen. Änderungen in diesem Editor werden in die ausgewählte NAT-Regel übernommen. Zusätzlich können selektierte NAT-Regeln und NAT-Regelsätze im erweiterten NAT-Editor kopiert werden. Für das Kopieren eines NAT-Regelsatzes muss dessen erste Zeile ausgewählt werden. Welche der kopierten NAT-Regeln jeweils angewendet wird, kann durch Anpassung der für die NAT-Regeln erzeugten IP-Regeln festgelegt werden. Der folgende Abschnitt veranschaulicht anhand von Beispielen die Projektierung von Adressbereichen im erweiterten NAT-Editor.

Projektierung von Adressbereichen im erweiterten NAT-Editor

Adressbereiche können in den Eingabefeldern der Spalten "Von" und "Bis" projektiert werden. Wenn mehrere Adressbereiche für eine NAT-Regel projektiert werden, muss die Anzahl der Adressen jeweils übereinstimmen. Nach der Projektierung von Adressbereichen wird aus der ausgewählten NAT-Regel von STEP 7 ein NAT-Regelsatz erstellt, der für jede der Adressen eine entsprechende NAT-Regel enthält.

In den Eingabefeldern für Ports in der Spalte "Bis" kann alternativ das Platzhalterzeichen "*" verwendet werden. In diesem Fall werden die betroffenen Portbereiche und NAT-Regeln gemäß der übrigen NAT-Parameter gebildet.

Adressumsetzung mit NAPT

Folgende Bedienelemente stehen in der Menüleiste zur Verfügung:

Bedienelement	Funktion
	NAPT aktivieren Aktiviert den Editor für NAPT-Regeln. Bei aktiviertem Editor sind die zugehörigen Regeln projektierbar und werden beim Laden auf das Modul übertragen. Nach Auswahl einer NAPT-Regel ist der Editor für erweiterte NAPT-Einstellungen im Register "Eigenschaften > Allgemein" des Inspektorfensters verfügbar.
	Regel oberhalb einfügen Fügt eine Regel oberhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen oberhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regel unterhalb einfügen Fügt eine Regel unterhalb der ausgewählten Regel / des ausgewählten Regelsatzes ein. Für das Einfügen unterhalb eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach oben verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach oben. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Nach unten verschieben Verschiebt die ausgewählte Regel / den ausgewählten Regelsatz um eine Zeile nach unten. Für das Verschieben eines Regelsatzes muss dessen erste Zeile ausgewählt werden.
	Regelsätze ausklappen Stellt vorhandene Regelsätze ausgeklappt dar.
	Regelsätze einklappen Stellt vorhandene Regelsätze eingeklappt dar.
	Separieren Wandelt die ausgewählte Zeile in separate Regel(n) um. Wenn die erste Zeile eines Regelsatzes ausgewählt wird, werden alle Regeln des Regelsatzes in separate Regeln umgewandelt. Wenn eine im Regelsatz enthaltene Regel ausgewählt wird, wird diese in eine separate Regel umgewandelt. Zusätzlich werden die nachfolgenden Regeln des Regelsatzes in einen separaten Regelsatz umgewandelt.

Nach dem Anlegen von NAPT-Regeln werden die zugehörigen Firewall-Regeln erzeugt und im Erweiterten Firewall Modus angezeigt, siehe Kapitel:
Zusammenhang zwischen NAT-/NAPT-Router und Firewall (Seite 449)

Die IP-Adressumsetzung mit NAPT kann in folgender Richtung durchgeführt werden:

- Extern nach Intern

Wenn die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) aktiviert ist, kann die IP-Adressumsetzung mit NAPT zusätzlich in folgenden Richtungen durchgeführt werden:

- Extern nach DMZ
- DMZ nach Intern
- DMZ nach Extern

Wenn sich das SCALANCE S Modul (nur SCALANCE S612/S623/S627-2M ab V4) in einer VPN-Gruppe befindet und die Tunnel-Schnittstelle aktiviert ist, kann die IP-Adressumsetzung mit NAPT zusätzlich in folgenden Richtungen durchgeführt werden:

- Tunnel nach Intern
- Tunnel nach Extern
- Tunnel nach DMZ (nur bei aktivierter DMZ-Schnittstelle)

Für die Richtung "Extern nach Intern" gilt beispielsweise: Telegramme, die an die externe IP-Adresse des Security-Moduls und an den Port gerichtet sind, der in der Spalte "Virtueller Port" eingetragen ist, werden an die angegebene Ziel-IP-Adresse im internen Netz sowie an den angegebenen Ziel-Port weitergeleitet.

Die IP-Adresse, von der aus Telegramme in das Ziel-Netz weitergeleitet werden, wird in der Spalte "Virtuelle IP-Adresse" angezeigt. Dies ist bei einer Richtung von Extern die externe IP-Adresse des Security-Moduls und bei einer Richtung von DMZ die IP-Adresse der DMZ-Schnittstelle des Security-Moduls.

Die folgende Tabelle zeigt das Eingabeschema für die Adressumsetzung mit NAPT im Register "NAPT":

Feld	Mögliche Eingaben	Bedeutung
Virtueller Port	TCP-/UDP-Port oder -Portbereich Beispiel für die Eingabe eines Portbereiches: 78:99	Ein Teilnehmer im Quell-Netz kann einem Teilnehmer im Ziel-Netz ein Telegramm senden, indem er diese Portnummer verwendet.
Ziel-IP-Adresse	IP-Adresse im Ziel-Netz	Telegramme, die an die IP-Adresse des Security-Moduls im Quell-Netz sowie an den im Feld "Virtueller Port" angegebenen TCP-/UDP-Port gerichtet sind, werden an die hier angegebene IP-Adresse weitergeleitet.
Ziel-Port	TCP-/UDP-Port	Portnummer, an die die vom Quell-Netz kommenden Telegramme weitergeleitet werden.
Protokoll	• TCP+UDP • TCP • UDP	Auswahl der Protokollfamilie für die angegebenen Portnummern
Nr.	-	Von STEP 7 vergebene, fortlaufende Nummer, die zur Bezugnahme auf die Firewall-Regel verwendet wird, welche von STEP 7 für die NAPT-Regel erzeugt wird.

Erweiterter NAPT-Editor

Nach Auswahl einer NAPT-Regel erscheint im Inspektorfenster unter "Eigenschaften > Allgemein" ein erweiterter NAPT-Editor zur vereinfachten Projektierung von Adressbereichen für Ports und IP-Adressen. Änderungen in diesem Editor werden in die ausgewählte NAPT-Regel übernommen. Zusätzlich können selektierte NAPT-Regeln und NAPT-Regelsätze im erweiterten NAPT-Editor kopiert werden. Für das Kopieren eines NAPT-Regelsatzes muss dessen erste Zeile ausgewählt werden. Welche der kopierten NAPT-Regeln jeweils angewendet wird, kann durch Anpassung der für die NAPT-Regeln erzeugten IP-Regeln festgelegt werden. Der folgende Abschnitt veranschaulicht anhand von Beispielen die Projektierung von Adressbereichen im erweiterten NAPT-Editor.

Projektierung von Adressbereichen im erweiterten NAPT-Editor

Adressbereiche können in den Eingabefeldern der Spalten "Von" und "Bis" projektiert werden. Wenn mehrere Adressbereiche für eine NAPT-Regel projektiert werden, muss die Anzahl der Adressen jeweils übereinstimmen. Nach der Projektierung von Adressbereichen wird aus der ausgewählten NAPT-Regel von STEP 7 ein NAPT-Regelsatz erstellt, der für jede der Adressen eine entsprechende NAPT-Regel enthält.

Das folgende Beispiel zeigt die Projektierung von Adressbereichen für den virtuellen Port, für die Ziel-IP-Adresse sowie für den Ziel-Port:

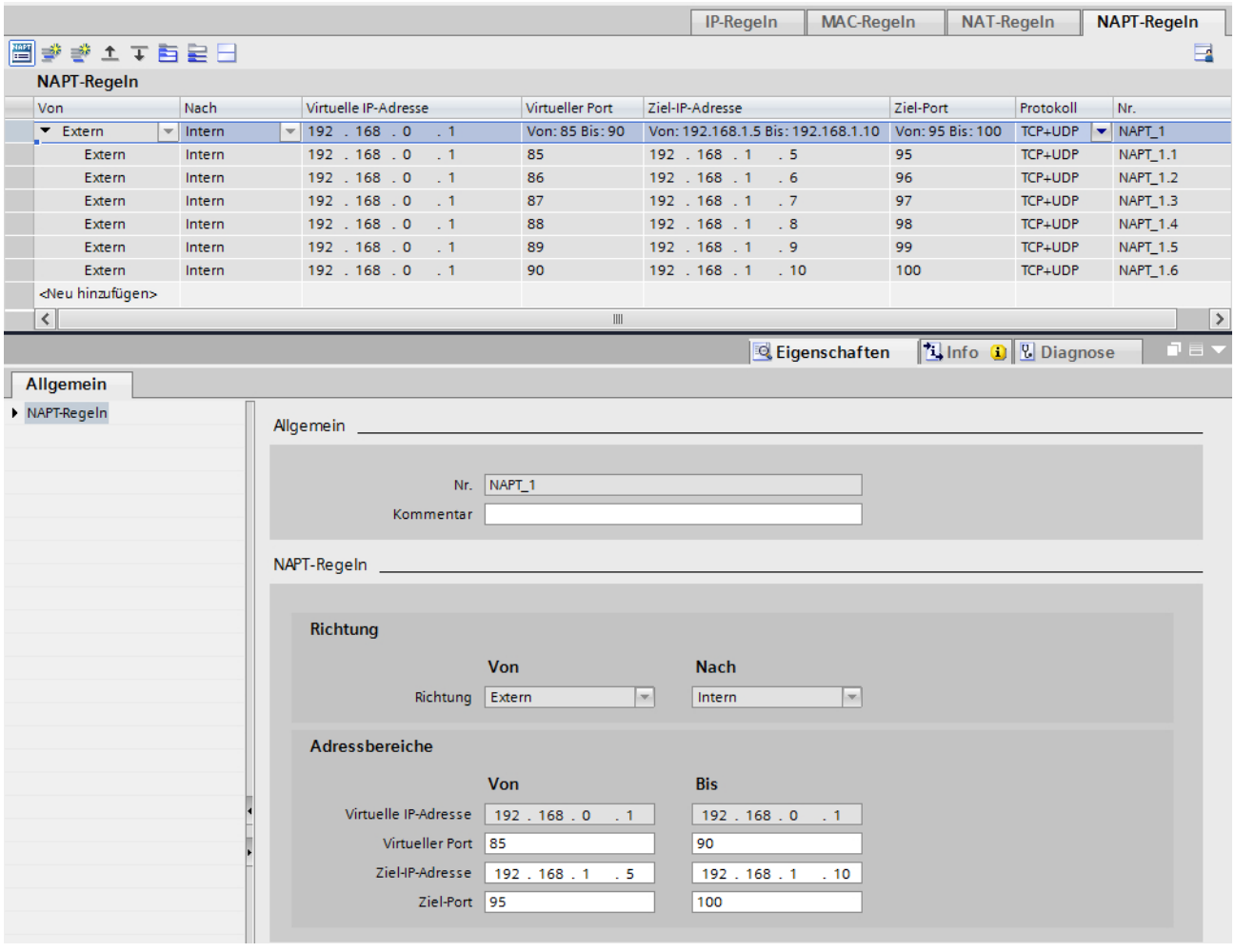

Bild 1-13 Erweiterter NAPT-Editor: Beispiel 1

In den Eingabefeldern für Ports in der Spalte "Bis" kann alternativ das Platzhalterzeichen "*" verwendet werden. In diesem Fall werden die betroffenen Portbereiche und NAPT-Regeln gemäß der übrigen NAPT-Parameter gebildet.

Im folgenden Beispiel werden die Portbereiche für den virtuellen Port und den Ziel-Port durch STEP 7 ermittelt:

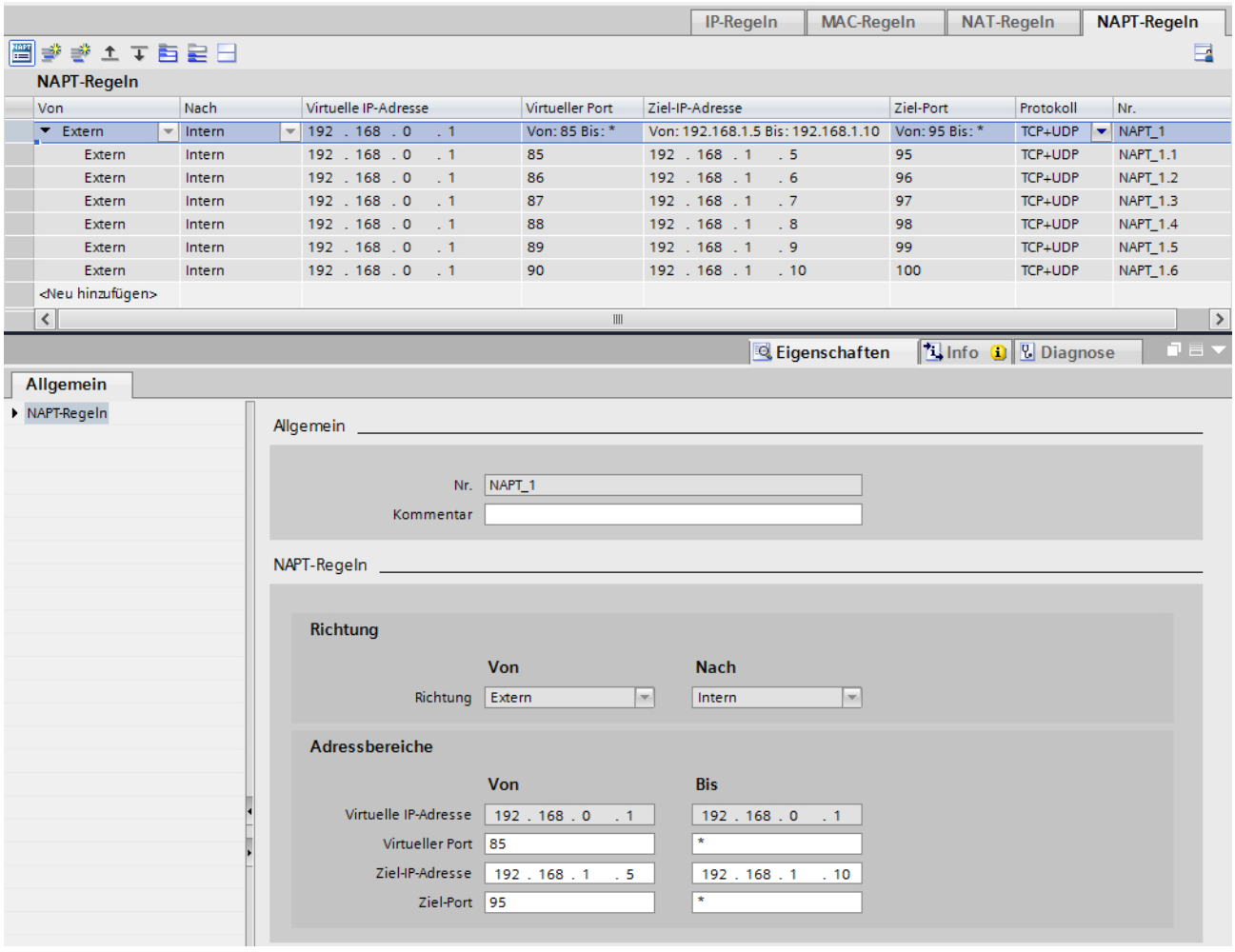

Bild 1-14 Erweiterter NAPT-Editor: Beispiel 2

Um denselben Portbereich für jede NAPT-Regel eines NAPT-Regelsatzes zu projektieren, können Sie den Start- und Endport des Portbereichs für den virtuellen Port in der Spalte "Von" durch einen Doppelpunkt getrennt angeben. Für den Ziel-Port kann ein solcher Portbereich nicht angegeben werden. Ein Platzhalterzeichen "*" nach dem Doppelpunkt wird als der Maximalwert 65535 interpretiert.

Im folgenden Beispiel wird ein Portbereich mit der erläuterten Notation für jede NAPT-Regel des NAPT-Regelsatzes verwendet:

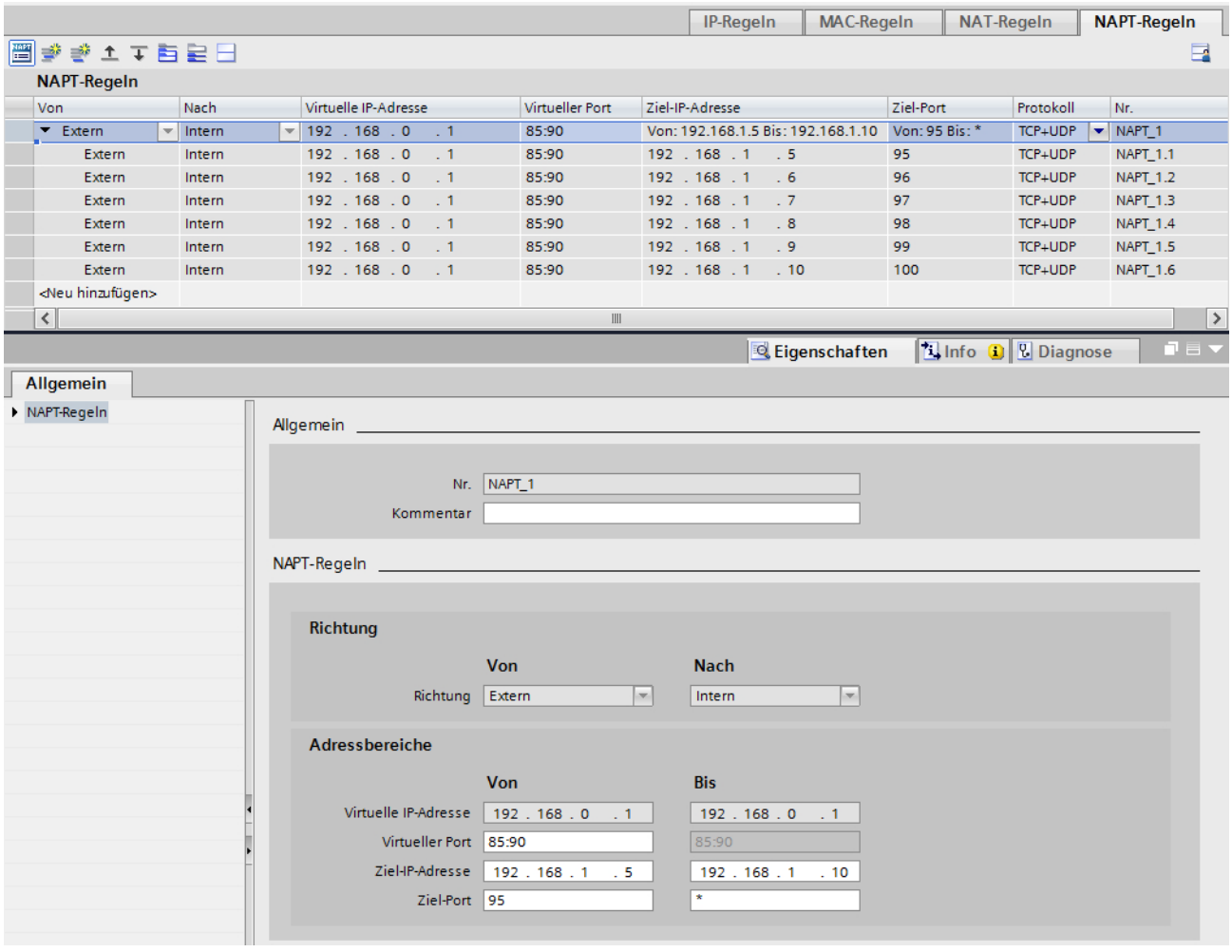

Bild 1-15 Erweiterter NAPT-Editor: Beispiel 3 (NAPT-Regeln)

In Beispiel 3 sind die virtuelle IP-Adresse und der virtuelle Port für jede NAPT-Regel identisch. Welche NAPT-Regel angewendet wird, entscheidet in diesem Fall die Priorität der Firewall-Regeln, die von STEP 7 für die NAPT-Regeln erstellt wurden. Die höchste Priorität hat die Firewall-Regel, die für die NAPT-Regel mit der Nummer NAPT_1.1 erstellt wurde.

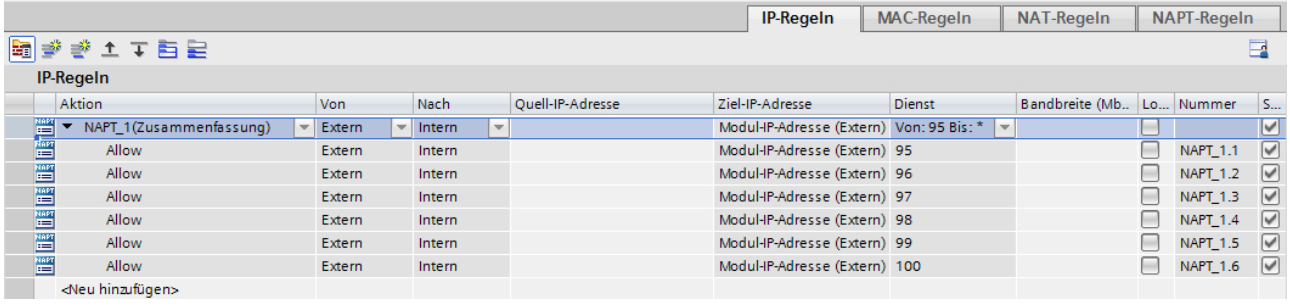

Bild 1-16 Erweiterter NAPT-Editor: Beispiel 3 (Firewall-Regeln)

Deshalb werden alle Telegramme, die an die IP-Adresse 192.168.0.1 und an einen Port im Bereich 85...90 gerichtet sind, an die Ziel-IP-Adresse 192.168.1.5 und an den Ziel-Port 95 weitergeleitet.

Um NAPT-Regeln bei gleicher virtueller IP-Adresse und gleichem virtuellen Port / Portbereich eindeutig zu definieren, können in den zugehörigen IP-Regeln Quell-IP-Adressen projektiert werden. Die Quell-IP-Adressen entscheiden dann jeweils darüber, welche NAPT-Regel angewendet wird:

IP-Regeln | MAC-Regeln | NAT-Regeln | NAPT-Regeln

IP-Regeln

Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse	Dienst	Bandbreite (Mb...	Lo...	Nummer	S...
NAPT_1(Zusammenfassung)	Extern	Intern		Modul-IP-Adresse (Extern)	Von: 95 Bis: *				
Allow	Extern	Intern	192 . 168 . 0 . 20	Modul-IP-Adresse (Extern)	95			NAPT_1.1	
Allow	Extern	Intern	192 . 168 . 0 . 21	Modul-IP-Adresse (Extern)	96			NAPT_1.2	
Allow	Extern	Intern	192 . 168 . 0 . 22	Modul-IP-Adresse (Extern)	97			NAPT_1.3	
Allow	Extern	Intern	192 . 168 . 0 . 23	Modul-IP-Adresse (Extern)	98			NAPT_1.4	
Allow	Extern	Intern	192 . 168 . 0 . 24	Modul-IP-Adresse (Extern)	99			NAPT_1.5	
Allow	Extern	Intern	192 . 168 . 0 . 25	Modul-IP-Adresse (Extern)	100			NAPT_1.6	
<Neu hinzufügen>									

Bild 1-17 Erweiterter NAPT-Editor: Erweiterung von Beispiel 3

Adressumsetzung mit NAT/NAPT in VPN-Tunneln

Modulspezifische Funktion

Die Adressumsetzung mit NAT/NAPT in VPN-Tunneln ist nur für SCALANCE S612/S623/S627-2M Module ab V4 verfügbar, siehe Abschnitt:
Adressumsetzung mit NAT/NAPT in VPN-Tunneln (Seite 534)

Zusammenhang zwischen NAT-/NAPT-Router und Firewall

Bedeutung

Von STEP 7 werden nach dem Anlegen von NAT-/NAPT-Regeln automatisch Firewall-Regeln erzeugt, die die Kommunikation in der projektierten Adressumsetzungsrichtung freigeben. Die erzeugten Firewall-Regeln sind im erweiterten Firewall-Modus als Firewall-Regelsätze sichtbar und können ggf. erweitert werden (zusätzliche IP-Adresse / IP-Adressbereich / IP-Adressband, Dienste, Übertragungsgeschwindigkeit). Zudem sollten die automatisch erzeugten Firewall-Regeln im Hinblick auf ihre Priorität bezüglich ihrer Position überprüft werden. Befinden sich zusätzlich manuell projektierte Firewall-Regeln in der Regelliste, die höher priorisiert sind als die automatisch erzeugten Firewall-Regeln, wird unter Umständen kein NAT/NAPT durchgeführt.

Sind mehrere identische NAT-/Firewall-Regelpaare vorhanden, entscheidet die Priorität in der Firewall-Regelliste darüber, welche Regel zur Anwendung kommt.

Von STEP 7 erzeugte Firewall-Parameter können nicht angepasst werden. Nach dem Deaktivieren von NAT/NAPT werden die von STEP 7 erzeugten Firewall-Regeln deaktiviert.

Zur einfacheren Bezugnahme zwischen NAT-/NAPT-Regeln und den zugehörigen Firewall-Regeln sind diese jeweils mit korrespondierenden, fortlaufenden Nummern gekennzeichnet.

In Firewall-Regeln, die für NAT-/NAPT-Regeln automatisch erzeugt wurden, kann das Kontrollkästchen "Stateful" nicht deaktiviert werden.

In der folgenden Tabelle werden die Schemata der Firewall-Regeln dargestellt, die für SCALANCE S Module für NAT-Regeln erzeugt werden.

Tabelle 1-35 NAT-Adressumsetzung und zugehörige Firewall-Regeln für SCALANCE S Module

NAT-Aktion	Angelegte Firewall-Regel				
	Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse
Destination-NAT	Accept	Quell-Netz	Ziel-Netz	-	IP-Adresse, die im Eingabefeld "Ziel-IP-Adresse" angegeben wurde
Source-NAT	Accept	Quell-Netz	Ziel-Netz	IP-Adresse des Teilnehmers, die im Eingabefeld "Quell-IP-Adresse" angegeben wurde	-
Source-NAT + Destination-NAT	Accept	Quell-Netz	Ziel-Netz	IP-Adresse des Teilnehmers, die im Eingabefeld "Quell-IP-Adresse" angegeben wurde	-
	Accept	Ziel-Netz	Quell-Netz	-	IP-Adresse, die von STEP 7 in das Eingabefeld "Ziel-IP-Adresse" eingefügt wurde
Double-NAT	Accept	Quell-Netz	Ziel-Netz	IP-Adresse des Teilnehmers, die im Eingabefeld "Quell-IP-Adresse" angegeben wurde	IP-Adresse, die im Eingabefeld "Ziel-IP-Adresse" angegeben wurde
	Accept	Quell-Netz	Ziel-Netz	IP-Adresse des Teilnehmers, die im Eingabefeld "Quell-IP-Adresse" angegeben wurde	IP-Adresse des Teilnehmers, die im Eingabefeld "Ziel-Umsetzung" angegeben wurde

In der folgenden Tabelle werden die Schemata der Firewall-Regeln dargestellt, die für CP 343-1 Adv. / CP 443-1 Adv. für NAT-Regeln erzeugt werden.

Tabelle 1-36 NAT-Adressumsetzung und zugehörige Firewall-Regeln für CP 343-1 Adv. / CP 443-1 Adv.

NAT-Aktion	Angelegte Firewall-Regel				
	Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse
Destination-NAT	Drop	Extern	Station	-	IP-Adresse des Security-Moduls im externen Netz
	Accept	Extern	Any	-	IP-Adresse des Teilnehmers, die im Eingabefeld "Ziel-Umsetzung" angegeben wurde

NAT-Aktion	Angelegte Firewall-Regel				
	Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse
Source-NAT	Accept	Any	Extern	IP-Adresse, die im Eingabefeld "Quell-Umsetzung" angegeben wurde	-
Source-NAT + Destination-NAT	Accept	Any	Extern	IP-Adresse, die im Eingabefeld "Quell-Umsetzung" angegeben wurde	-
	Drop	Extern	Station	-	IP-Adresse des Security-Moduls im externen Netz
	Accept	Extern	Any	-	IP-Adresse des Teilnehmers, die von STEP 7 in das Eingabefeld "Ziel-Umsetzung" eingefügt wurde

In der folgenden Tabelle wird das Schema für Firewall-Regeln dargestellt, die für SCALANCE S Module für NAPT-Regeln erzeugt werden.

Tabelle 1-37 NAPT-Umsetzungen und angelegte Firewall-Regeln für SCALANCE S Module

Angelegte Firewall-Regel					
Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse	Dienst
Accept	Quell-Netz	Ziel-Netz	-	IP-Adresse des Security-Moduls im Quell-Netz	[Dienst_NAPT-Regel]

In der folgenden Tabelle wird das Schema für Firewall-Regeln dargestellt, die für CP 343-1 Adv. / CP 443-1 Adv. für NAPT-Regeln erzeugt werden.

Tabelle 1-38 NAPT-Umsetzungen und angelegte Firewall-Regeln für CP 343-1 Adv. / CP 443-1 Adv.

Angelegte Firewall-Regeln					
Aktion	Von	Nach	Quell-IP-Adresse	Ziel-IP-Adresse	Dienst
Drop	Extern	Station	-	IP-Adresse des Security-Moduls im externen Netz	[Dienst_NAPT-Regel]
Accept	Extern	Any	-	IP-Adresse des Security-Moduls im externen Netz	[Dienst_NAPT-Regel]

Stateful Packet Inspection

Firewall und NAT-/NAPT-Router unterstützen den Mechanismus "Stateful Packet Inspection". Daher können Antworttelegramme den NAT-/NAPT-Router und die Firewall passieren, ohne dass deren Adressen in der Firewall-Regel und der NAT/NAPT-Adressumsetzung zusätzlich aufgenommen werden müssen.

Zusammenhang zwischen NAT-/NAPT-Router und benutzerspezifischer Firewall

Modulspezifische Funktion

Die Projektierung von NAT-/NAPT-Regeln in der benutzerspezifischen Firewall ist nur für SCALANCE S Module ab V3 verfügbar, siehe Abschnitt:

Zusammenhang zwischen NAT-/NAPT-Router und benutzerspezifischer Firewall (Seite 536)

Uhrzeitsynchronisation konfigurieren

Verfahren der Uhrzeitsynchronisation

Bedeutung

Zur zeitlichen Synchronisation der Übertragung der Produktivdaten, zur Überprüfung der zeitlichen Gültigkeit eines Zertifikats und für die Zeitstempel von Log-Einträgen werden auf dem Security-Modul Datum und Uhrzeit geführt. Für die Security-Funktionen muss die Uhrzeit der Security-Module eines Projekts synchronisiert werden.

Hinweis

Firewall für Kommunikation mit NTP-Server projektieren

Wenn der NTP-Server vom Security-Modul nicht erreichbar ist, müssen Sie die Telegramme des NTP-Servers in der Firewall explizit freigeben (UDP, Port 123).

Hinweis

Die Uhrzeitsynchronisation bezieht sich lediglich auf das Security-Modul und kann nicht zur Synchronisation von Geräten im internen Netz des Security-Moduls verwendet werden.

Einige CPs können die Uhrzeit an andere Module der Station weiterleiten.

Der CP 443-1 OPC UA kann die Uhrzeit im SIMATIC-Verfahren nur von der Station empfangen, aber nicht weiterleiten.

Hinweis

Bevor die Security-Funktionen eines CP (Uhrzeit-Slave) zur Anwendung kommen, muss der CP ein gültiges Uhrzeittelegramm vom Uhrzeit-Master erhalten.

Uhrzeitsynchronisation für ein Security-Modul projektieren

So erreichen Sie diese Funktion

1. Selektieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Uhrzeitsynchronisation".
3. Aktivieren Sie das Kontrollkästchen "Uhrzeitsynchronisation aktivieren".

Verfahren der Uhrzeitsynchronisation

Abhängig vom verwendeten Security-Modul sind folgende Verfahren projektierbar:

Verfahren	Bedeutung
Uhrzeit beim Laden setzen	Automatisches Stellen der Modul-Uhrzeit mit der PC-Uhrzeit beim Laden einer Konfiguration.
SIMATIC	Uhrzeitsynchronisation durch empfangene MMS-Uhrzeitnachrichten (MMS = Manufacturing Message Specification) Wählen Sie für den CP aus, ob der CP die Uhrzeit übernehmen oder auch weiterleiten soll. Beim CP 1628 wird die Uhrzeit stets weitergeleitet. Der CP 443-1 OPC UA verwendet das SIMATIC-Verfahren, wenn keine Uhrzeitsynchronisation über NTP projektiert ist. Er kann die Uhrzeit nur von der Station empfangen, aber nicht weiterleiten. Verfügbare Richtungen: • Automatisch Der CP nimmt die Uhrzeit aus der Station bzw. vom LAN entgegen und leitet diese an die Station bzw. zum LAN weiter. Wenn mehrere CPs in der Station betrieben werden, kann die Voreinstellung zu Kollisionen führen. Um dies zu vermeiden, sollten Sie die Richtung der Weiterleitung gezielt festlegen. • Von Station Der CP leitet die Uhrzeit von der Station an das LAN weiter. • Von LAN Der CP leitet die Uhrzeit vom LAN an die Station weiter. Bei aktivierter Weiterleitung der Uhrzeit können Sie über das Kontrollkästchen "Korrigierte Uhrzeit verwenden" festlegen, ob ein im Uhrzeittelegramm ggf. enthaltener Korrekturfaktor verwendet werden soll. Für den CP 1628 ist diese Option voreingestellt und kann nicht deaktiviert werden.

Verfahren	Bedeutung
Uhrzeit vom Partner	Die Uhrzeit wird vom jeweiligen Kommunikationspartner bezogen. Synchronisationszyklus: Legt den Zyklus der Uhrzeitsynchronisation fest. Für den Synchronisationszyklus des CP kann ein individuelles Stunden- oder Minutenraster festgelegt werden.
NTP / NTP (secure)	Uhrzeitsynchronisation durch einen NTP-Server / NTP-Server vom Typ "NTP (secure)" • Zeitzone: Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time). Durch die Projektierung der lokalen Zeitzone kann der Zeitversatz gegenüber UTC eingestellt werden. • Aktualisierungsintervall in Sekunden: Legt den Zeitabstand der Uhrzeitanfragen in Sekunden fest. Für SCALANCE S ab V3 wird das Zeitintervall für die Abfrage des NTP-Servers automatisch festgelegt. Wenn das Kontrollkästchen "Aktiviere Security-Funktionen" in den lokalen Security-Einstellungen von CPs aktiviert wird, wird die Einstellung des Aktualisierungsintervalls aus den lokalen Einstellungen des CPs in die lokalen Security-Einstellungen des CPs übernommen. • Uhrzeitsynchronisation zur vollen Minute: Mit dieser Option legen Sie fest, dass die Uhrzeit genau zur vollen Minute auf den K-Bus weitergeleitet wird. Diese Option ist nur für einige Spezialanwendungen erforderlich. • Uhrzeit von nicht synchronisierten NTP-Servern annehmen: Hier ist einstellbar, ob das Security-Modul die Uhrzeit auch von nicht synchronisierten NTP-Servern annehmen soll. • Uhrzeit an Station weiterleiten: Deaktivieren Sie diese Option, wenn die CPU die Uhrzeit eigenständig von einem NTP-Server anfordert. Sie vermeiden damit, dass die in der CPU vom NTP-Server direkt erfasste Uhrzeit von der im CP erfassten Uhrzeit wieder überschrieben wird. Durch die Weiterleitung über den CP könnte sich eine geringere Genauigkeit ergeben. • NTP-Server: Das Anlegen von NTP-Servern in den lokalen Security-Einstellungen wird in Kapitel NTP-Server projektieren (Seite 455) beschrieben.

NTP-Server projektieren

NTP-Server vom Typ "NTP (secure)" in globalen Security-Funktionen anlegen

In den globalen Security-Funktionen können nur NTP-Server vom Typ "NTP (secure)" angelegt und CPs bzw. SCALANCE S Modulen ab V4 zugewiesen werden.

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "NTP".
   Ergebnis: Unter dem ausgewählten Eintrag werden die bisher angelegten NTP-Server (secure) angezeigt.
2. Doppelklicken Sie auf den Eintrag "Neuen NTP-Server hinzufügen".
   Ergebnis: Der erstellte NTP-Server (secure) wird mit einer automatisch vergebenen Nummer unterhalb des Eintrags "NTP" angezeigt.
3. Doppelklicken Sie auf den erstellten NTP-Server (secure).
   Ergebnis: Im Arbeitsbereich wird der Zuweisungsdialog für NTP-Server (secure) angezeigt. In der Klappliste "NTP-Server" ist der selektierte NTP-Server (secure) ausgewählt. Diesem können die gewünschten Security-Module zugewiesen werden, siehe Kapitel Security-Modul einem NTP-Server (secure) zuweisen (Seite 457). In den lokalen Security-Einstellungen werden die projektierbaren Eigenschaften des NTP-Servers (secure) angezeigt.

4. Geben Sie in den lokalen Security-Einstellungen im Eintrag "Allgemein" die folgenden Daten ein:
   – Name: Es können maximal 25 Zeichen für den Namen des NTP-Servers (secure) verwendet werden.
   – IP-Adresse / FQDN: Adresse des NTP-Servers (secure).
5. Tragen Sie in den lokalen Security-Einstellungen im Eintrag "NTP-Schlüssel" die folgenden Verschlüsselungsparameter ein:

Eigenschaft	Bedeutung
Schlüssel-ID	Numerischer Wert zwischen 1 ... 65534.
Hash-Algorithmus	Wählen Sie den Hash-Algorithmus aus. Beachten Sie je nach Auswahl die Länge für den NTP-Schlüssel.
Hex/ASCII	Wählen Sie das Format für den NTP-Schlüssel aus.
Schlüssel	Geben Sie den NTP-Schlüssel ein. SCALANCE S-600 V3 / S7-CPs Auswahl MD5 oder SHA-1: • Hex: 10 ... 40 Zeichen • ASCII: 5 ... 20 Zeichen TIM, CP 1628 Auswahl MD5 oder SHA-1: • Hex: 22 ... 40 Zeichen • ASCII: 11 ... 20 Zeichen SCALANCE SC-600 Auswahl MD5: • ASCII: 16 ... 128 Zeichen Auswahl SHA1: • ASCII: 20 ... 128 Zeichen Auswahl DES: • ASCII: 8 Zeichen Die aufgeführten Zeichenanzahlen decken alle von den Security-Modulen unterstützten Schlüssellängen ab.

Um zu den lokalen Security-Einstellungen eines anderen NTP-Servers (secure) umzuschalten, wählen Sie diesen aus der Klappliste "NTP-Server" des Zuweisungsdialogs aus oder doppelklicken Sie auf den zugehörigen Eintrag in den globalen Security-Funktionen.

Ergebnis

Sie haben den NTP-Server (secure) angelegt und können diesen nun den gewünschten Security-Modulen zuweisen.
Beachten Sie die Beschreibungen in folgendem Kapitel:
Security-Modul einem NTP-Server (secure) zuweisen (Seite 457)

Ungesicherte NTP-Server in den lokalen Security-Einstellungen anlegen

Ungesicherte NTP-Server für SCALANCE S-Module unter V4 und CPs müssen Sie in den lokalen Security-Einstellungen anlegen.

1. Selektieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Uhrzeitsynchronisation".
3. Wählen Sie das Synchronisierungsverfahren aus.
4. Geben Sie einen Namen und die IP-Adresse des NTP-Servers ein. Für den Namen können maximal 25 Zeichen verwendet werden. Falls Sie den Synchronisierungsmodus "NTP (secure)" ausgewählt haben, können Sie in der Spalte "Name" einen NTP-Server (secure) auswählen, den Sie in den globalen Security-Funktionen angelegt haben.

Mengengerüst für NTP-Server

Sie können einem Security-Modul max. 4 NTP-Server zuweisen.

Vereinbarungen für Namen von NTP-Servern

Die Zeichen 0-9, a-z, A-Z, -._ können für die Namen von NTP-Servern verwendet werden. Ein Name muss mit einem Buchstaben beginnen und mit einem Buchstaben oder einer Zahl enden. Vor dem Zeichen '.' sind nur Buchstaben und Zahlen, nach dem Zeichen '.' nur Buchstaben erlaubt. Vor und nach dem Zeichen '_' oder '-' sind nur Buchstaben und Zahlen erlaubt.

Import / Export von NTP-Servern (secure)

Über die Menübefehle "Importieren" bzw. "Exportieren" des Kontextmenüs können Sie die Schlüsselliste des aktuell selektierten NTP-Servers (secure) in den globalen Security-Funktionen exportieren und die Datei in einen NTP-Server (secure) importieren bzw. umgekehrt.

Security-Modul einem NTP-Server (secure) zuweisen

Voraussetzung

- Sie haben in den globalen Security-Funktionen einen NTP-Server (secure) angelegt.
- In den lokalen Security-Einstellungen des Security-Moduls, das Sie einem NTP-Server (secure) zuweisen möchten, ist als Synchronisierungsmodus "NTP" oder "NTP (secure)" ausgewählt.

Vorgehensweise

1. Doppelklicken Sie in den globalen Security-Funktionen auf den Eintrag "NTP".
2. Doppelklicken Sie auf den Eintrag "Modul einem NTP-Server zuweisen".
3. Wählen Sie aus der Klappliste "NTP Server" den NTP-Server (secure) aus, dem Sie ein Security-Modul zuweisen möchten.

4. Wählen Sie im Bereich "Verfügbare Module" das Security-Modul aus, das Sie dem ausgewählten NTP-Server (secure) zuweisen möchten.
5. Klicken Sie auf die Schaltfläche "<<", um das ausgewählte Security-Modul dem ausgewählten NTP-Server (secure) zuzuweisen.

Ergebnis

Sie haben das Security-Modul dem NTP-Server (secure) zugewiesen. Der NTP-Server (secure) erscheint in den lokalen Security-Einstellungen automatisch in der Liste der NTP-Server.

Security-Modul als DHCP-Server

Modulspezifische Funktion

Die Verwendung des Security-Moduls als DHCP-Server ist nur bei SCALANCE S Modulen möglich, siehe Abschnitt:
AUTOHOTSPOT.

SNMP konfigurieren

Übersicht SNMP

Was ist SNMP?

Das Security-Modul unterstützt die Übertragung von Managementinformationen über das Simple Network Management Protocol (SNMP). Dafür ist auf dem Security-Modul ein SNMP-Agent installiert, der die SNMP-Anfragen entgegennimmt und beantwortet. Informationen über die Eigenschaften von SNMP-fähigen Geräten sind in sogenannten MIB-Dateien (Management Information Base) hinterlegt, für die der Benutzer die notwendigen Rechte haben muss.

Beim SNMPv1 wird der "Community String" mitgesendet. Der Community String ist wie ein Passwort, welches zusammen mit der SNMP-Anfrage verschickt wird. Ist der Community String korrekt, antwortet das Security-Modul mit der angeforderten Information. Ist der String falsch, verwirft das Security-Modul die Anfrage und antwortet nicht. Der Community String wird bei SNMPv1 unverschlüsselt übertragen.

Mit SNMPv3 können die Daten verschlüsselt übertragen werden.

SNMP konfigurieren - Eintrag "SNMP"

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "SNMP".

3. Aktivieren Sie das Kontrollkästchen "SNMP aktivieren".
4. Wählen Sie eine der folgenden SNMP-Protokollversionen aus:

   Hinweis

   Verschlüsselte Datenübertragung bei SNMPv3

   Um die Sicherheit zu erhöhen, sollten Sie SNMPv3 verwenden, da die Daten hierbei verschlüsselt übertragen werden.

   – SNMPv1
     Das Security-Modul verwendet zur Steuerung der Zugriffsrechte im SNMP-Agent folgende Standardwerte für die Community Strings. Diese Standardwerte sollten zur Erhöhung der Sicherheit angepasst werden.
     Für Lesezugriff: public
     Für Lese- und Schreibzugriff: private
     Um den Schreibzugriff über SNMP zu aktivieren, wählen Sie die Option "Erlaube schreibenden Zugriff".
   – SNMPv3
     Wählen Sie entweder nur einen Authentifizierungs-Algorithmus oder einen Authentifizierungs-Algorithmus und einen Verschlüsselungs-Algorithmus aus.
     Authentifizierungs-Algorithmus: keine, MD5, SHA-1
     Verschlüsselungs-Algorithmus: keine, AES-128, DES

     Hinweis

     Verwendung von DES vermeiden

     Bei DES handelt es sich um einen unsicheren Verschlüsselungsalgorithmus. Er sollte deshalb nur aus Gründen der Abwärtskompatibilität verwendet werden.

     Hinweis

     Bei der Verwendung von SNMPv3 ist keine RADIUS-Authentifizierung möglich.

5. Wenn SNMPv3 verwendet werden soll, weisen Sie einem Benutzer eine Rolle zu, bei der die entsprechenden SNMP-Rechte aktiviert sind, damit er das Modul über SNMP erreichen kann. Eine Übersicht der SNMP-Rechte finden Sie in Kapitel:
   Besonderheiten der Benutzerverwaltung für Security-Funktionen (Seite 382)
6. Konfigurieren Sie im Bereich "Erweiterte Einstellungen" für SCALANCE S Module modulspezifische Angaben zu Autor, Ort und E-Mail-Adresse, welche die Angaben aus den Projekteigenschaften überschreiben.
   Für Werte, die durch ein SNMP-Werkzeug über einen SNMP-SET-Befehl auf das Security-Modul geschrieben wurden, gilt:
   Wenn Sie das Kontrollkästchen "Durch SNMP-Set geschriebene Werte beibehalten" aktivieren, werden die Werte durch das erneute Laden einer STEP 7 Konfiguration auf das Security-Modul nicht überschrieben.

Proxy-ARP konfigurieren

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V3 verfügbar, siehe Abschnitt:
Proxy-ARP konfigurieren (Seite 541)

Webserver auf Security-Modul aktivieren

Modulspezifische Funktion

Diese Funktion ist nur für CP 343-1 Advanced / CP 443-1 Advanced und CP 443-1 OPC UA verfügbar, siehe Abschnitt:
Webserver auf Security-Modul aktivieren (Seite 568) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

IPsec-VPN-Tunnel: VPN-Gruppen anlegen und zuordnen

So legen Sie mittels VPN-Gruppen IPsec-Tunnel an

Voraussetzung

Hinweis

Aktuelles Datum und aktuelle Uhrzeit auf den Security-Modulen

Achten Sie bei der Verwendung von gesicherter Kommunikation (z. B. HTTPS, VPN...) darauf, dass die betroffenen Security-Module über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die verwendeten Zertifikate werden sonst als nicht gültig ausgewertet und die gesicherte Kommunikation funktioniert nicht.

VPN-Gruppe anlegen

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "VPN-Gruppen".
   Ergebnis: Unter dem ausgewählten Eintrag werden die bisher angelegten VPN-Gruppen angezeigt.
2. Doppelklicken Sie auf den Eintrag "Neue VPN-Gruppe hinzufügen".
   Ergebnis: Die erstellte VPN-Gruppe wird mit einer automatisch vergebenen Nummer unterhalb des Eintrags "VPN-Gruppen" angezeigt.
   Alternativ können Sie in der Netzsicht mit der rechten Maustaste auf eine VPN-fähige Schnittstelle eines Moduls klicken und über den Kontextmenübefehl "Neue VPN-Gruppe hinzufügen" eine VPN-Gruppe anlegen (nicht für SOFTNET Security Client möglich). Der erstellten VPN-Gruppe wird das ausgewählte Security-Modul automatisch zugeordnet.
3. Doppelklicken Sie auf die erstellte VPN-Gruppe.
   Ergebnis: Im Arbeitsbereich wird der Zuweisungsdialog für VPN-Gruppen angezeigt. In der Klappliste "VPN" ist die selektierte VPN-Gruppe ausgewählt. Dieser können Security-Module über den zugehörigen Zuweisungsdialog zugewiesen werden, siehe Kapitel Security-Modul einer VPN-Gruppe zuweisen (Seite 464). In den lokalen Security-Einstellungen können die VPN-Gruppeneigenschaften der ausgewählten VPN-Gruppe angepasst werden, siehe Kapitel VPN-Gruppeneigenschaften projektieren (Seite 466).

Vereinbarungen für Namen von VPN-Gruppen

Die Zeichen 0-9, a-z, A-Z, -._ können für die Namen von VPN-Gruppen verwendet werden. Ein Name muss mit einem Buchstaben beginnen und mit einem Buchstaben oder einer Zahl enden. Vor dem Zeichen '.' sind nur Buchstaben und Zahlen, nach dem Zeichen '.' nur Buchstaben erlaubt. Vor und nach dem Zeichen '_' oder '-' sind nur Buchstaben und Zahlen erlaubt.

Anzeige der VPN-Gruppen mit ihren Eigenschaften

Wenn Sie ein Security-Modul auswählen, das sich in einer oder mehreren VPN-Gruppe(n) befindet, werden im Bereich "Netzwerkdaten" die Eigenschaften der VPN-Gruppe(n) angezeigt, in der/denen sich das Security-Modul befindet.

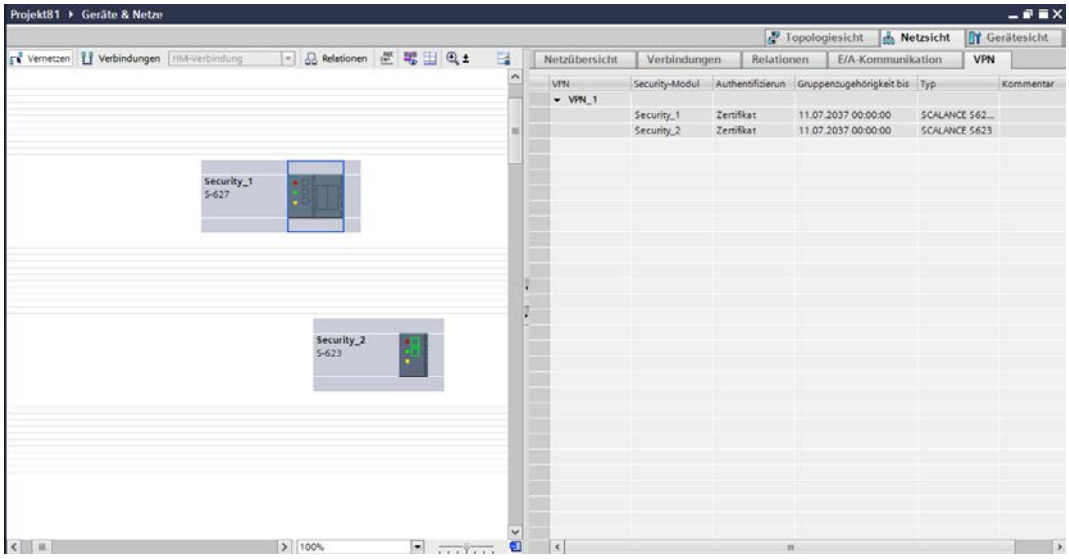

Folgende Eigenschaften der VPN-Gruppen werden im Register "VPN" des Bereichs "Netzwerkdaten" spaltenweise angezeigt:

Eigenschaft/Spalte	Bedeutung
VPN	Namen der VPN-Gruppen, in denen sich das ausgewählte Security-Modul befindet
Security-Modul	Namen der zugeordneten Security-Module
Authentifizierung	Authentifizierungstyp: Pre-shared key oder Zertifikat
Gruppenzugehörigkeit bis	Datum und Uhrzeit, bis zu denen das VPN-Gruppenzertifikat des Security-Moduls gültig ist
Typ	Modellnummern der zugeordneten Security-Module
Kommentar	Kommentar

Lebensdauer von Zertifikaten einstellen

Öffnen Sie den Dialog, in dem Sie das Ablaufdatum des Zertifikats eingeben können, wie folgt:

1. Markieren Sie im Register "VPN" die zu bearbeitende VPN-Gruppe.
2. Wählen Sie im Register "Eigenschaften" > "Allgemein" des Inspektorfensters den Eintrag "Authentifizierung".

Hinweis

Ablauf eines Zertifikats

Die Kommunikation durch den VPN-Tunnel läuft nach Ablauf des Zertifikats weiter, bis der Tunnel abgebaut wird oder die SA-Lebensdauer abläuft. Weitere Informationen zu Zertifikaten finden Sie im Kapitel:
AUTOHOTSPOT.

Mengengerüste

Anzahl der VPN-Tunnel	
SCALANCE S612 V3/V4	128 maximal
SCALANCE S623 V3/V4	128 maximal
SCALANCE S627-2M V4	128 maximal
CP 343-1 Advanced / CP 443-1 Advanced	32 maximal
CP 1628	64 maximal
S7-1200-CPs/ET 200SP-CPs	8 maximal
S7-1500-CPs	16 maximal

Für die Anzahl der Kommunikationsbeziehungen zwischen Subnetzen des Security-Moduls und Subnetzen anderer über VPN-Tunnel erreichbarer Security-Module bestehende folgende Mengengerüste:

Anzahl der Subnetzbeziehungen	
SCALANCE S612 V3/V4	128 maximal
SCALANCE S623 V3/V4	128 maximal
SCALANCE S627-2M V4	128 maximal
CP 343-1 Advanced / CP 443-1 Advanced	32 maximal
CP 1628	64 maximal
S7-1200-/S7-1500-CPs	8 maximal

Security-Modul einer VPN-Gruppe zuweisen

Voraussetzungen

- Sie haben in den globalen Security-Funktionen eine VPN-Gruppe angelegt. Beachten Sie vor der Zuweisung von Security-Modulen zu VPN-Gruppen die zugehörigen Regeln im Kapitel Betriebsarten von VPN-Gruppen (Seite 478).
- Vor der Zuweisung zu VPN-Gruppen muss für folgende Module zunächst VPN aktiviert und der VPN-Verbindungstyp "Internet Key Exchange (IKE)" ausgewählt werden:
  - CP 1243-1 ab V3.1
  - CP 1243-8 IRC ab V3.1
  - CP 1243-7 LTE ab V3.1
  - CP 1543SP-1 ab V2

Vorgehensweise

1. Doppelklicken Sie in den globalen Security-Funktionen auf den Eintrag "VPN-Gruppen".
2. Doppelklicken Sie auf den Eintrag "Modul einer VPN-Gruppe zuweisen".
3. Wählen Sie aus der Klappliste "VPN" die VPN-Gruppe aus, der Sie ein Security-Modul zuweisen möchten.
4. Wählen Sie im Bereich "Verfügbare Module" das Security-Modul aus, das Sie der ausgewählten VPN-Gruppe zuweisen möchten.
5. Klicken Sie auf die Schaltfläche "<<", um das ausgewählte Security-Modul der ausgewählten VPN-Gruppe zuzuweisen.
   Wenn das Security-Modul eine oder mehrere Eigenschaften der VPN-Gruppe nicht unterstützt, wird das Modulsymbol im Bereich der zugewiesenen Module mit einem Fehlersymbol dargestellt.
   Alternativ zur oben beschriebenen Vorgehensweise können Sie ein Security-Modul einer bestehenden VPN-Gruppe zuweisen, indem Sie in der Netzsicht mit der rechten Maustaste auf eine VPN-fähige Schnittstelle eines Moduls klicken und dieses über den Kontextmenübefehl "Modul einer VPN-Gruppe zuweisen" einer VPN-Gruppe zuordnen (nicht für SOFTNET Security Client möglich).

Ergebnis

Sie haben das Security-Modul der VPN-Gruppe zugewiesen.

Authentifizierungsmethoden

Folgende Methoden stehen Ihnen zur Verfügung

Die Authentifizierungsmethode wird pro VPN-Gruppe festgelegt und bestimmt die Art der verwendeten Authentifizierung.

Es werden schlüsselbasierte oder zertifikatsbasierte Authentifizierungsmethoden unterstützt:

- Pre-shared key
  Die Authentifizierung erfolgt über eine zuvor festgelegte Zeichenfolge, die an alle in der Gruppe befindlichen Module verteilt wird.
  Geben Sie dafür in den VPN-Gruppeneigenschaften unter "Authentifizierung" > "Allgemein" im Feld "Schlüssel" einen Pre-shared key ein. Wenn Sie auf die Schaltfläche "Pre-shared key erneuern" klicken, erneuert STEP 7 den zu verwendenden Pre-shared key.
- Zertifikat
  Die zertifikatbasierte Authentifizierung "Zertifikat" ist die Standardeinstellung. Das Verhalten ist wie folgt:
  - Beim Anlegen einer Gruppe wird automatisch ein CA-Zertifikat als Stammzertifikat erzeugt.
  - Jedes Security-Modul, das Teilnehmer der VPN-Gruppe ist, erhält zusätzlich ein Gruppenzertifikat, das mit dem Schlüssel des CA-Zertifikats signiert ist.

  Wenn Sie auf die Schaltfläche "Zertifikat erneuern" klicken, erneuert STEP 7 das CA-Zertifikat der VPN-Gruppe.
  Sämtliche Zertifikate basieren auf dem ITU-Standard X.509v3 (ITU, International Telecommunications Union).
  Die Zertifikate werden von einer in STEP 7 enthaltenen Zertifizierungsstelle erzeugt.

Hinweis

Einschränkung bei VLAN-Betrieb

Bei der Übertragung von IP-Telegrammen durch den VPN-Tunnel des Security-Moduls wird kein VLAN-Tagging übertragen. Die in den IP-Telegrammen enthaltenen VLAN-Tags gehen beim Passieren der Security-Module verloren, da für die Übertragung der IP-Telegramme IPsec verwendet wird.

Standardmäßig können mit IPsec keine IP-Broadcast- bzw. IP-Multicast-Telegramme durch einen Layer 3 VPN-Tunnel übertragen werden. Durch einen Layer 2 VPN-Tunnel des Security-Moduls werden IP-Broadcast- bzw. IP-Multicast-Telegramme genau wie MAC-Pakete inklusive Ethernet-Header in UDP "verpackt" und übertragen. Daher bleibt bei diesen Paketen das VLAN-Tagging erhalten.

VPN-Gruppeneigenschaften

VPN-Verbindungsaufbau durch Known Peers und Unknown Peers

Die Eigenschaften einer VPN-Gruppe setzen sich aus dem gewählten Authentifizierungsverfahren sowie aus den projektierten Parametern für IKE-Phase 1 und 2 zusammen und werden beim Laden auf alle VPN-Gruppenteilnehmer übertragen. Beim Responder werden zusätzlich die IP-Adressen der Security-Baugruppen hinterlegt, die den VPN-Verbindungsaufbau mit den projektierten Eigenschaften der VPN-Gruppe initiieren dürfen. Wenn eine Security-Baugruppe versucht, eine VPN-Verbindung zum Responder aufzubauen, prüft dieser zunächst, ob ihm die IP-Adresse des Initiators bekannt ist. Ist dies der Fall, handelt es sich bei dem Initiator um einen Known Peer. Wenn die von diesem Known Peer vorgeschlagenen VPN-Gruppeneigenschaften mit denen übereinstimmen, die beim Responder für den Known Peer hinterlegt sind, lässt der Responder den VPN-Verbindungsaufbau zu. Zusätzlich zu den Eigenschaften einer VPN-Gruppe können einer VPN-Gruppe Fallback-VPN-Profile zugewiesen werden. VPN-Profile enthalten vordefinierte Eigenschaften für das Authentifizierungsverfahren, IKE Phase 1 und 2. Fallback-VPN-Profile können für SCALANCE S-600 V3, CP 343-1 Adv. / CP 443-1 Adv. und CP 1628 als alternative Konfigurationen zum VPN-Verbindungsaufbau verwendet werden. Diese treten in Kraft, wenn VPN-Eigenschaften des Initiators nicht zu VPN-Eigenschaften des Responders passen. Fallback-VPN-Profile stellen in solchen Fällen den VPN-Verbindungsaufbau sicher. Die projektierten Fallback-VPN-Profile werden ebenso wie die Eigenschaften der VPN-Gruppe auf alle VPN-Gruppenteilnehmer übertragen. Die Priorität für die Verwendung der Fallback-VPN-Profile wird über deren Reihenfolge in der Projektierung festgelegt. Die Fallback-VPN-Profile und die Eigenschaften der VPN-Gruppe dürfen sich nicht im Authentifizierungsverfahren unterscheiden.

Ist dem Responder die IP-Adresse des Initiators nicht bekannt, weil der Initiator im Produktivbetrieb beispielsweise über eine NAT-Umsetzung angebunden ist, handelt es sich bei dem Initiator um einen Unknown Peer (Road Warrior). Standardmäßig lässt der Responder den Verbindungsaufbau durch Unknown Peers zu. Ob und mit welchen VPN-Gruppeneigenschaften der Verbindungsaufbau durch Unknown Peers vom Responder zugelassen ist, können Sie in den baugruppenspezifischen VPN-Eigenschaften des Responders festlegen, sofern es sich um eine Baugruppe vom Typ SCALANCE S-600 V3, CP 343-1 Adv. / CP 443-1 Adv. oder CP 1628 handelt, siehe Kapitel Modul- und verbindungsspezifische VPN-Einstellungen projektieren (Seite 482).

VPN-Gruppeneigenschaften projektieren

VPN-Gruppeneigenschaften

Hinweis

IPsec-Kenntnisse erforderlich

Um diese Parameter einstellen zu können, benötigen Sie IPsec-Kenntnisse. Wenn Sie keine Einstellungen vornehmen bzw. verändern, gelten die Standardeinstellungen.

In den Eigenschaften einer VPN-Gruppe sind folgende Einstellungen projektierbar:

- Kompatible VPN-Konfiguration automatisch einstellen (Eintrag: "Einstellungen")
- Authentifizierungsverfahren (Eintrag: "Allgemein")
- IKE-Einstellungen (Eintrag: "Erweiterte Einstellungen Phase 1")
- IPsec-Einstellungen (Eintrag: "Erweiterte Einstellungen Phase 2")
- Fallback-VPN-Profile

Wenn die von Ihnen projektierten VPN-Gruppeneigenschaften weniger sicher sind als die empfohlenen Eigenschaften, weist STEP 7 Sie darauf mit einem entsprechenden Meldungstext hin.

So erreichen Sie diese Funktion

1. Selektieren Sie in den globalen Security-Funktionen unter dem Eintrag "VPN-Gruppen" die VPN-Gruppe, deren Eigenschaften Sie projektieren wollen.
2. Wählen Sie im Kontextmenü dieses Eintrags den Menübefehl "Öffnen".
   Ergebnis: Die VPN-Gruppeneigenschaften erscheinen unter "Authentifizierung" im Bereich der lokalen Security-Einstellungen.
3. Wählen Sie unter dem Eintrag "Einstellungen" aus, ob die kompatible VPN-Konfiguration automatisch eingestellt werden soll. Wenn die Funktion aktivert ist, werden die Einstellungen der einzelnen Gruppenmitglieder gemäß den unterstützten Gerätefunktionen automatisch so geändert, dass sie den optimalen Sicherheitsanforderungen entsprechen.
4. Wählen Sie unter dem Eintrag "Allgemein" aus, ob für die Authentifizierung ein Pre-shared key oder Zertifikat verwendet werden soll. Weitere Informationen hierzu finden Sie im Kapitel:
   Authentifizierungsmethoden (Seite 464).
5. Wählen Sie die IKE-Version aus. Je nach Auswahl stehen in den "Erweiterten Einstellungen" verschiedene Auswahlen zur Verfügung.

Erweiterte Einstellungen Phase 1

Phase 1: IKE-Aushandlung der Security Association (SA) für Phase 2:

Stellen Sie hier die Parameter für die Aushandlung der Sicherheitsparameter ein, die in Phase 2 verwendet werden:

Folgende VPN-Gruppeneigenschaften werden von den einzelnen Modulen unterstützt:

Modul	IKE-Version	Phase 1				
		IKE-Modus	DH-Gruppe	SA-Lebensdauer	Verschlüsselung	Authentifizierung
CP 1543-1 (ab V2.1) CP 1543SP-1 (V2.0) CP 1243-1 BX30 (ab V3.0) CP 1243-7 LTE BX30 (ab V3.0) CP 1243-8 IRC BX30 (ab V3.0)	IKEv1	Main	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 Gruppe 15	Time: 1440 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
CP 1543-1 (bis 2.0) CP 1543SP-1 (V1.0) CP 1243-1 BX30 (bis V2.1) CP 1243-7 LTE BX30 (V2.1) CP 1243-8 IRC BX30 (V2.1)	IKEv1	Main	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14	Time: 1440 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
CP 343-1 GX31 Advanced CP 443-1 GX30 Advanced CP 443-1 OPC UA CP 1628 SCALANCE S-600 ab V3	IKEv1	Main	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14	Time: 1440 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
SOFTNET Security Client V4, V5 (WinXP)	IKEv1	Main	Gruppe 2 Gruppe 14	Time: 1440 ... 2879 Minuten	3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
Security-Gerät projektiert als SOFTNET Security Client V5 (Win7)	IKEv1	Main	Gruppe 2 Gruppe 14	Time: 1440 ... 2879 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
MD741-1 V1.1, M875 V2.0	IKEv1	Main	Gruppe 2	Time: 1440 ... 2500000 Minuten	3DES-168**: Dreifach-DES***	SHA-1***: Secure Hash Algorithm 1

Modul	IKE-Version	Phase 1				
		IKE-Modus	DH-Gruppe	SA-Lebensdauer	Verschlüsselung	Authentifizierung
Security-Gerät projektiert als VPN-Client VPN-Server	IKEv1	Main Aggressive	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 Gruppe 15	Time: 10 ... 1666666 Minuten	AES-256, 192, 128: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
	IKEv2	Main	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 - Gruppe 18	Time: 10 ... 1666666 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC, CTR, CCM 16 oder GCM 16) 3DES-168**: Dreifach-DES***	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
Security-Gerät projektiert als NCP VPN-Client	IKEv1	Main Aggressive	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 Gruppe 15	Time: 10 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5

Modul	IKE-Version	Phase 1				
		IKE-Modus	DH-Gruppe	SA-Lebensdauer	Verschlüsselung	Authentifizierung
SCALANCE SC-600 SCALANCE S615	IKEv1	Main Aggressive	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 Gruppe 15	Time: 10 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5
	IKEv2	Main	Gruppe 1 Gruppe 2 Gruppe 5 Gruppe 14 - Gruppe 18	Time: 10 ... 2500000 Minuten	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC, CTR, CCM 16 oder GCM 16) 3DES-168**: Dreifach-DES***	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5

* 256 Bit, 192 Bit oder 128 Bit Schlüssellänge

** 168 Bit Schlüssellänge

*** Die Einstellung ist als weniger sicher eingestuft. Es wird empfohlen, nach Möglichkeit eine sicherere Einstellung zu verwenden.

VPN-Gruppeneigenschaften für Phase 2

Phase 2: IKE-Aushandlung der Security Association (SA) für IPsec-Datenaustausch:

Stellen Sie hier die Parameter für die Aushandlung der Sicherheitsparameter ein, die für den IPsec-Datenaustausch mit ESP (Encapsulating Security Payload) und AH (Authentication Header) verwendet werden. Die Kommunikation in Phase 2 erfolgt bereits verschlüsselt.

Folgende VPN-Gruppeneigenschaften werden von den einzelnen Modulen unterstützt:

Modul	IKE-Version	Phase 2			
		SA-Lebensdauer	Verschlüsselung	Authentifizierung	Perfect Forward Secrecy
CP 1543-1 (ab V2.1) CP 1543SP-1 (V2.0) CP 1243-1 BX30 (ab V3.0) CP 1243-7 LTE BX30 (ab V3.0) CP 1243-8 IRC BX30 (ab V3.0)	IKEv1	Time: 60 ... 16666666 Minuten Limit: 2000 ... 500000 MByte	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
CP 1543-1 (bis 2.0) CP 1543SP-1 (V1.0) CP 1243-1 BX30 (bis V2.1) CP 1243-7 LTE BX30 (V2.1) CP 1243-8 IRC BX30 (V2.1)	IKEv1	Time: 60 ... 16666666 Minuten Limit: 2000 ... 500000 MByte	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
CP 343-1 GX31 Advanced CP 443-1 GX30 Advanced CP 443-1 OPC UA CP 1628 SCALANCE S-600 ab V3	IKEv1	Time: 60 ... 16666666 Minuten Limit: 2000 ... 500000 MByte	AES-192, 128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.

Modul	IKE-Version	Phase 2			
		SA-Lebensdauer	Verschlüsselung	Authentifizierung	Perfect Forward Secrecy
SOFTNET Security Client V4, V5 (WinXP)	IKEv1	Time: 60 ... 2879 Minuten	3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
Security-Gerät projektiert als SOFTNET Security Client V5 (Win7)	IKEv1	Time: 60 ... 2879 Minuten	AES-128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
MD741-1 V1.1, M875 V2.0	IKEv1	Time: 60 ... 16666666 Minuten	3DES-168**: Dreifach-DES***	SHA-1***: Secure Hash Algorithm 1	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.

Modul	IKE-Version	Phase 2			
		SA-Lebensdauer	Verschlüsselung	Authentifizierung	Perfect Forward Secrecy
Security-Gerät projektiert als VPN-Client VPN-Server	IKEv1	Time: 10 ... 16666666 Minuten Limit: 10 ... 4096 MByte	AES-128*: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
	IKEv2	Time: 10 ... 16666666 Minuten Limit: 10 ... 4096 MByte	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC, CTR, CCM 16 oder GCM 16) 3DES-168**: Dreifach-DES***	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
Security-Gerät projektiert als NCP VPN-Client	IKEv1	Time: 10 ... 16666666 Minuten Limit: 10 ... 4096 MByte	AES-128: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES*** DES	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.

Modul	IKE-Version	Phase 2			
		SA-Lebensdauer	Verschlüsselung	Authentifizierung	Perfect Forward Secrecy
SCALANCE SC-600 SCALANCE S615	IKEv1	Time: 10 ... 16666666 Minuten Limit: 10 ... 4096 MByte	AES-128: Advanced Encryption Standard (Modus CBC) 3DES-168**: Dreifach-DES***	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.
	IKEv2	Time: 10 ... 16666666 Minuten Limit: 10 ... 4096 MByte	AES-256, 192, 128*: Advanced Encryption Standard (Modus CBC, CTR, CCM 16 oder GCM 16) 3DES-168**: Dreifach-DES (Modus CBC)***	SHA-512: Secure Hash Algorithm 512 SHA-384: Secure Hash Algorithm 384 SHA-256: Secure Hash Algorithm 256 SHA-1***: Secure Hash Algorithm 1 MD5***: Message Digest Algorithm 5	EIN: Für die Neuberechnung der Schlüssel werden neue Diffie Hellman Public Key Werte ausgetauscht. AUS***: Für die Neuberechnung der Schlüssel werden die bereits in Phase 1 ausgetauschten Werte verwendet.

* 256 Bit, 192 Bit oder 128 Bit Schlüssellänge

** 168 Bit Schlüssellänge

*** Die Einstellung ist als weniger sicher eingestuft. Es wird empfohlen, nach Möglichkeit eine sicherere Einstellung zu verwenden.

Fallback-VPN-Profile projektieren

Zusätzlich zu den Eigenschaften einer VPN-Gruppe können einer VPN-Gruppe Fallback-VPN-Profile zugewiesen werden. VPN-Profile enthalten vordefinierte Eigenschaften für das Authentifizierungsverfahren, IKE Phase 1 und 2. Fallback-VPN-Profile können als alternative Konfigurationen zum VPN-Verbindungsaufbau verwendet werden. Diese treten in Kraft, wenn VPN-Eigenschaften des Initiators nicht zu VPN-Eigenschaften des Responders passen. Fallback-VPN-Profile stellen in solchen Fällen den VPN-Verbindungsaufbau sicher. Die projektierten Fallback-VPN-Profile werden ebenso wie die Eigenschaften der VPN-Gruppe auf alle VPN-Gruppenteilnehmer übertragen. Die Priorität für die Verwendung der Fallback-VPN-Profile wird über deren Reihenfolge in der Projektierung festgelegt. Die Fallback-VPN-Profile und die Eigenschaften der VPN-Gruppe dürfen sich nicht im Authentifizierungsverfahren unterscheiden.

Folgende Fallback-VPN-Profile sind verfügbar:

Tabelle 1-39 VPN-Profil 1

Parameter	Einstellung
Authentifizierungsmethode	Zertifikat
IKE-Modus	Main
Phase 1 DH-Gruppe	Gruppe 14
Phase 1 Verschlüsselung	AES-256
Phase 1 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 480 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 480 … 2880 Minuten
Phase 1 Authentifizierung	SHA-1
Phase 2 SA-Lebensdauertyp	Time
Phase 2 Verschlüsselung	AES-128
Phase 2 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 240 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 60 ... 2880 Minuten
Phase 2 Authentifizierung	SHA-1
Perfect Forward Secrecy	AUS

Tabelle 1-40 VPN-Profil 2

Parameter	Einstellung
Authentifizierungsmethode	Zertifikat
IKE-Modus	Main
Phase 1 DH-Gruppe	Gruppe 2
Phase 1 Verschlüsselung	AES-256
Phase 1 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 480 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 480 … 2880 Minuten
Phase 1 Authentifizierung	SHA-1
Phase 2 SA-Lebensdauertyp	Time
Phase 2 Verschlüsselung	3DES-168
Phase 2 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 2880 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 60 ... 2880 Minuten
Phase 2 Authentifizierung	SHA-1
Perfect Forward Secrecy	AUS

Tabelle 1-41 VPN-Profil 3

Parameter	Einstellung
Authentifizierungsmethode	Zertifikat
IKE-Modus	Main
Phase 1 DH-Gruppe	Gruppe 2
Phase 1 Verschlüsselung	3DES-168
Phase 1 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 480 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 480 … 2880 Minuten
Phase 1 Authentifizierung	SHA-1
Phase 2 SA-Lebensdauertyp	Time
Phase 2 Verschlüsselung	3DES-168
Phase 2 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 2880 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 60 ... 2880 Minuten
Phase 2 Authentifizierung	SHA-1
Perfect Forward Secrecy	AUS

Tabelle 1-42 VPN-Profil 4

Parameter	Einstellung
Authentifizierungsmethode	Zertifikat
IKE-Modus	Main
Phase 1 DH-Gruppe	Gruppe 2
Phase 1 Verschlüsselung	DES
Phase 1 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 480 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 480 … 2880 Minuten
Phase 1 Authentifizierung	MD5
Phase 2 SA-Lebensdauertyp	Time
Phase 2 Verschlüsselung	3DES-168
Phase 2 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 2880 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 60 ... 2880 Minuten
Phase 2 Authentifizierung	SHA-1
Perfect Forward Secrecy	AUS

Tabelle 1-43 VPN-Profil 5

Parameter	Einstellung
Authentifizierungsmethode	Pre-shared key
IKE-Modus	Main
Phase 1 DH-Gruppe	Gruppe 2
Phase 1 Verschlüsselung	3DES-168
Phase 1 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 480 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 480 … 2880 Minuten
Phase 1 Authentifizierung	SHA-1
Phase 2 SA-Lebensdauertyp	Time
Phase 2 Verschlüsselung	3DES-168
Phase 2 SA-Lebensdauer	Vom Initiator vorgeschlagene SA-Lebensdauer: 2880 Minuten Vom Responder zugelassener Bereich für die SA-Lebensdauer: 60 ... 2880 Minuten
Phase 2 Authentifizierung	SHA-1
Perfect Forward Secrecy	AUS

Nach dem Öffnen eines STEP 7 Projekts, das zuletzt mit einer älteren Version als V14 SP1 gespeichert wurde, sind allen vorhandenen VPN-Gruppen mit dem Authentifizierungsverfahren "Zertifikat" standardmäßig die vordefinierten Fallback-VPN-Profile 2 - 5 zugeordnet.

Besonderheiten für den SOFTNET Security Client und SCALANCE M

Projektierte Fallback-VPN-Profile fließen nicht in die Konfigurationsdateien für die Roadwarrior-Baugruppen SOFTNET Security Client und SCALANCE M ein. Für SOFTNET Security Client und SCALANCE M Baugruppen gilt:

- Das zu verwendende VPN-Profil für VPN-Verbindungen von SOFTNET Security Client / SCALANCE M Baugruppen zu SCALANCE S / CPs wählen Sie in den Responder-Einstellungen aus, siehe Kapitel Modul- und verbindungsspezifische VPN-Einstellungen projektieren (Seite 482).
- Für VPN-Verbindungen von SCALANCE M Baugruppen zu SCALANCE M Baugruppen werden die projektierten VPN-Gruppeneinstellungen verwendet.
- Für VPN-Verbindungen von SOFTNET Security Client zu SCALANCE M Baugruppen werden stets die Einstellungen des VPN-Profils 3 verwendet.

Betriebsarten von VPN-Gruppen

VPN-Betriebsarten

Je nachdem, in welcher Betriebsart sich die Security-Module befinden, die einer VPN-Gruppe hinzugefügt wurden, wird zwischen unterschiedlichen Betriebsarten von VPN-Gruppen unterschieden. Die Betriebsart einer VPN-Gruppe gibt Aufschluss darüber, welche Security-Module in welchen Betriebsarten der VPN-Gruppe hinzugefügt werden können.

Regeln für die Gruppenbildung

Beachten Sie die folgenden Regeln, wenn Sie VPN-Gruppen bilden wollen:

- Für SCALANCE S612 / S613 / S623 / S627-2M / SCALANCE M / VPN-Gerät
  Das erste in einer VPN-Gruppe zugeordnete Modul bestimmt, welche zusätzlichen Module hinzugefügt werden können.
  Ist das erste hinzugefügte SCALANCE S Modul im Routing-Modus oder handelt es sich bei dem ersten Modul um ein SCALANCE M Modul bzw. um ein VPN-Gerät, so können zusätzlich nur SCALANCE S Module mit aktiviertem Routing oder SCALANCE M Module bzw. VPN-Geräte hinzugefügt werden, da SCALANCE M Module sowie VPN-Geräte immer im Routing-Modus betrieben werden. Ist das erste hinzugefügte SCALANCE S Modul im Bridge-Modus, so können zusätzlich nur SCALANCE S Module im Bridge-Modus hinzugefügt werden. Soll der Modus einer VPN-Gruppe geändert werden, so müssen alle in der Gruppe enthaltenen Module entfernt und neu hinzugefügt werden. Ein CP sowie ein SSC kann in eine Gruppe mit einem SCALANCE S im Bridge- oder Routing-Modus hinzugefügt werden.
- Für CP / SSC
  Befindet sich ein CP / SSC als erstes in einer VPN-Gruppe, dann können Security-Module in beliebigen Modi hinzugefügt werden. Das nächste die Betriebsart festlegende Modul bestimmt dann die Betriebsart der VPN-Gruppe. Ein CP / SSC kann gleichzeitig mehreren VPN-Gruppen mit unterschiedlicher Betriebsart zugeordnet werden. Der CP / SSC wird dann im Mischbetrieb betrieben.
- Es ist nicht möglich, ein SCALANCE M Modul einer VPN-Gruppe hinzuzufügen, die ein Modul im Bridge-Modus enthält.

Entnehmen Sie der folgenden Tabelle, welche Module in einer VPN-Gruppe zusammengefasst werden können:

Tabelle 1-44 Security-Module und VPN-Betriebsarten

Modul	Kann aufgenommen werden in VPN-Gruppe im...	
	Bridge-Modus	Routing-Modus
SCALANCE S612 / S613 / S623 / S627-2M im Bridge-Modus	x	-*
SCALANCE S612 / S613 / S623 / S627-2M im Routing-Modus	-	x
CP 343-1 Adv. / CP 443-1 Adv.	x	x
S7-1200-/S7-1500-CPs	x	x
CP 1628	x	x

Modul	Kann aufgenommen werden in VPN-Gruppe im...	
	Bridge-Modus	Routing-Modus
SOFTNET Security Client V4.0	x	x
SCALANCE M875 / VPN-Gerät	-	x

* SCALANCE S623/S627-2M Module im Bridge-Modus können in eine VPN-Gruppe im Routing-Modus eingefügt werden, wenn deren DMZ-Schnittstelle aktiviert ist (nicht gleichzeitig).

Security-Modul in konfigurierte VPN-Gruppe aufnehmen

Die projektierten VPN-Gruppeneigenschaften werden für Security-Module, die in eine bestehende VPN-Gruppe aufgenommen werden, übernommen.

Vorgehen nach Aufnahme eines Security-Moduls in eine konfigurierte VPN-Gruppe

Je nachdem, ob Sie die VPN-Gruppeneigenschaften seit dem letzten Ladevorgang geändert haben oder nicht, müssen Sie beim Vorgehen unterscheiden:

- **Fall a:** Wenn Sie die VPN-Gruppeneigenschaften nicht geändert haben und das hinzuzufügende Modul die Verbindung zu Modulen vom Typ SCALANCE S, CP 343-1 Adv. / CP 443-1 Adv. bzw. CP 1628 aktiv aufbaut:

1. Fügen Sie das neue Security-Modul der VPN-Gruppe hinzu.
2. Laden Sie die Konfiguration in das neue Modul.

- **Fall b:** Wenn Sie die VPN-Gruppeneigenschaften geändert haben oder das hinzuzufügende Modul die Verbindung zu den bereits konfigurierten Modulen nicht aktiv aufbaut:

1. Fügen Sie das neue Security-Modul der VPN-Gruppe hinzu.
2. Laden Sie die Konfiguration in alle Module, die zur VPN-Gruppe gehören.

Bereits vorhandene, in Betrieb genommene Security-Module müssen im Fall a nicht neu projektiert und geladen werden. Die laufende Kommunikation wird nicht beeinflusst oder unterbrochen.

Vorgehen nach Entnahme eines aktiven Teilnehmers aus einer VPN-Gruppe

Wenn Sie einen aktiven Teilnehmer aus einer bestehenden VPN-Gruppe entfernen, kann dieser immer noch eine Verbindung zu den Gruppenteilnehmern aufbauen, auch wenn Sie das Projekt erneut auf alle Teilnehmer der VPN-Gruppe geladen haben.

Soll der entfernte aktive Teilnehmer keine Verbindung mehr aufbauen können, erneuern Sie das CA-Gruppenzertifikat und laden Sie das Projekt erneut auf die Teilnehmer der VPN-Gruppe. Erneuert werden kann das Zertifikat in den Gruppeneigenschaften der VPN-Gruppe oder im Zertifikatsmanager, Register "CA".

Durch Tunnel erreichbare Netzknoten konfigurieren

Übersicht

Durch Tunnel erreichbare Netzknoten konfigurieren

Jedem Security-Modul müssen die über VPN-Tunnel erreichbaren Netzknoten bekannt sein, um die Authentizität eines Telegramms feststellen zu können.

Das Security-Modul muss sowohl seine eigenen Knoten kennen als auch die Knoten der Security-Module, mit denen es sich zusammen in einer VPN-Gruppe befindet. Diese Information wird auf einem Security-Modul dazu verwendet, um zu bestimmen, welches Datenpaket in welchem Tunnel übertragen werden soll.

Durch das Hinzufügen eines Security-Moduls zu einer VPN-Gruppe werden die lokalen, internen Netzknoten/Subnetze des Security-Moduls dem Security-Modul automatisch bekannt gemacht. Um die Kommunikation über den VPN-Tunnel mit weiteren Subnetzen oder Teilnehmern eines weiteren Subnetzes (geroutetes internes Netz; DMZ-Netz, wenn VPN-Tunnel an externer Schnittstelle und umgekehrt) zu ermöglichen, müssen diese Subnetze oder Teilnehmer für die VPN-Tunnelkommunikation über die Konfiguration freigegeben werden.

SCALANCE S Module bieten die Möglichkeit, die Netzknoten automatisch zu erlernen oder statisch zu konfigurieren. Über die Möglichkeiten, die zum Lernen interner Netzknoten zur Verfügung stehen, entscheidet zudem die Betriebsart des Security-Moduls.

SCALANCE S im Bridge-Modus

Im Bridge-Modus können Sie die IP-/MAC-Knoten sowie die Subnetze konfigurieren oder alternativ das automatische Lernen von Knoten durch den SCALANCE S zulassen.

SCALANCE S im Routing-Modus

Im Routing-Modus steht kein automatisches Lernen zur Verfügung. Stattdessen tragen Sie hier komplette Subnetze ein, die für die Tunnelkommunikation freigegeben werden sollen.

CP 343-1 Advanced / CP 443-1 Advanced und CP 1628

- CP 343-1 Advanced / CP 443-1 Advanced
  Wählen Sie aus, ob die Tunnelkommunikation zum CP (GBit-Schnittstelle) und/oder zum internen Subnetz (PROFINET-Subnetz) für VPN-Verbindungspartner im Routing-Modus (SCALANCE S / M) erlaubt ist.
- CP 1628
  Tragen Sie die NDIS-Knoten ein, die durch den Tunnel von VPN-Verbindungspartnern im Routing-Modus (SCALANCE S / M) erreichbar sein sollen.

CP 1543-1 ab V2.1 mit aktiviertem IP-Routing zwischen Kommunikationsmodulen

Tragen Sie die Subnetze bzw. IP-Adressen ein, die über den Rückwandbus der Station erreichbar sind und die für die Tunnelkommunikation freigegeben werden sollen.

Automatisches Lernen interner Netzknoten

Modulspezifische Funktion

SCALANCE S Module im Bridge-Modus bieten einen Lernmodus, mit dem interne Netzknoten im Betrieb automatisch gelernt werden können. Weitere Informationen hierzu finden Sie im Abschnitt:
Lernmodus zum Lernen interner Knoten verwenden (Seite 541).

IP-Netzknoten für SCALANCE S manuell konfigurieren

Modulspezifische Funktion

Wie Sie IP-Netzknoten für SCALANCE S Module manuell konfigurieren, siehe Abschnitt:
IP-Netzknoten manuell konfigurieren (Seite 543).

MAC-Netzknoten für SCALANCE S manuell konfigurieren

Modulspezifische Funktion

Wie Sie MAC-Netzknoten für SCALANCE S Module manuell konfigurieren, siehe Abschnitt:
MAC-Netzknoten manuell konfigurieren (Seite 543).

Interne Subnetze für SCALANCE S manuell konfigurieren

Modulspezifische Funktion

Wie Sie interne Subnetze für SCALANCE S Module konfigurieren, siehe Abschnitt:
Interne Subnetze manuell konfigurieren (Seite 544).

Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben

Modulspezifische Funktion

Wie Sie den Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben, siehe Abschnitt:
Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben (Seite 569) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

Durch Tunnel erreichbare NDIS Knoten für PC-CPs konfigurieren

Modulspezifische Funktion

Wie Sie durch Tunnel erreichbare NDIS Knoten bei PC-CPs konfigurieren, siehe Abschnitt: Durch Tunnel erreichbare NDIS-Knoten bei PC-CPs manuell konfigurieren (Seite 569) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

Über Rückwandbus erreichbare Subnetze für CP 1543-1 konfigurieren

Modulspezifische Funktion

Wie Sie über den Rückwandbus der Station erreichbare Subnetze und Netzknoten für CP 1543-1 ab V2.1 konfigurieren, siehe Abschnitt:
Über Rückwandbus erreichbare Subnetze für CP 1543-1 konfigurieren (Seite 582).

Modul- und verbindungsspezifische VPN-Einstellungen projektieren

Voraussetzung

Das Modul ist Teilnehmer einer VPN-Gruppe.

Modul- und verbindungsspezifische Einstellungen

Mit modul- und verbindungsspezifischen Einstellungen lassen sich spezifische VPN-Einstellungen projektieren. Modulspezifische Einstellungen werden speziell für ein Security-Modul projektiert, während verbindungsspezifische Einstellungen speziell für ein Security-Modul in einer bestimmten VPN-Gruppe projektiert werden.

Folgende **modulspezifische** Eigenschaften können Sie in den lokalen Security-Einstellungen unter dem Eintrag "VPN" konfigurieren:

- Dead-Peer-Detection
- Erlaubnis zur Initiierung des Verbindungsaufbaus
- Öffentliche Adresse (IP-Adresse / FQDN) zur Kommunikation über Internet-Gateways
- Responder-Einstellungen für Road Warrior Verbindungen
- Knoten, die für die Tunnelkommunikation freigegeben werden sollen

Wenn Sie in der Auflistung der VPN-Gruppen im Bereich "Netzwerkdaten" ein Security-Modul auswählen, sind folgende **verbindungsspezifische** VPN-Einstellungen einsehbar bzw. projektierbar:

- Erlaubnis zur Initiierung des Verbindungsaufbaus
- Partner-Baugruppen, zu denen Tunnelverbindungen bestehen
- Art der übertragenen Pakete

- Auswahl der lokalen Schnittstelle des ausgewählten Security-Moduls, die als Tunnelendpunkt fungieren soll
- Auswahl der Partner-Schnittstelle, die als Tunnelendpunkt fungieren soll

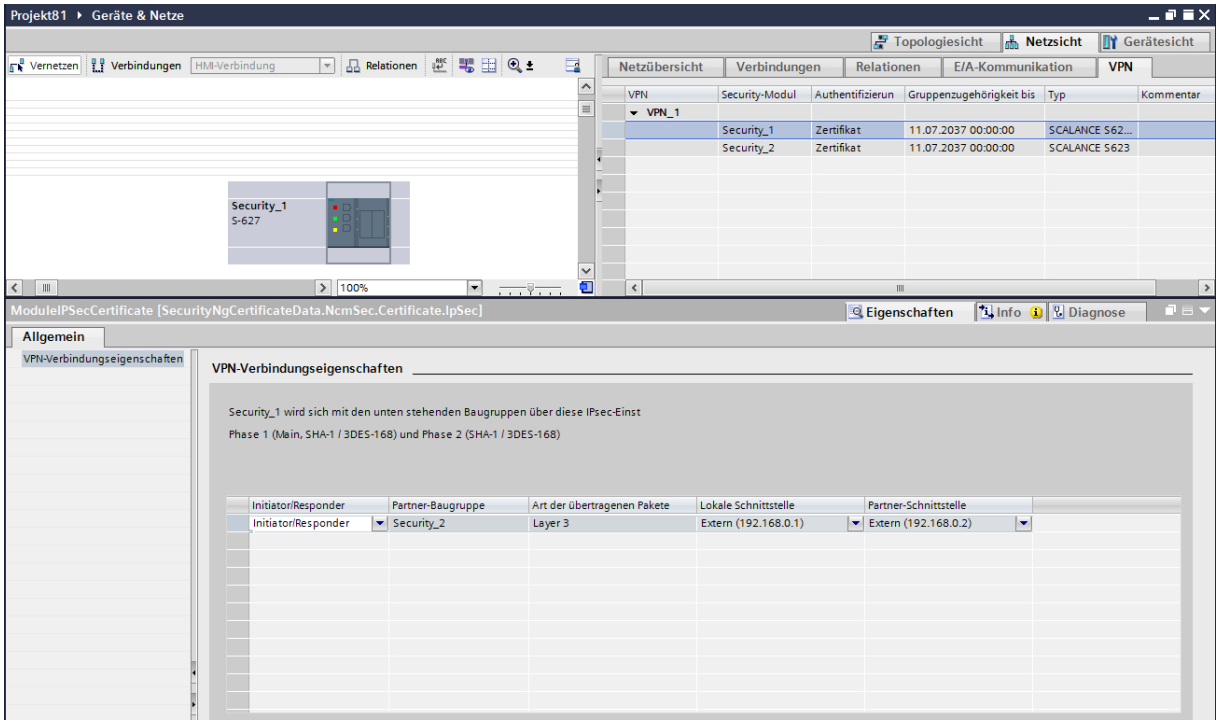

Dead-Peer-Detection (DPD)

Standardmäßig ist DPD aktiviert.

Bei aktivierter DPD tauschen die Module in einstellbaren Zeitintervallen zusätzliche Nachrichten aus, sofern zum jeweiligen Zeitpunkt keine Kommunikation stattfindet. Hierdurch kann erkannt werden, ob die IPsec-Verbindung noch gültig ist oder eventuell neu aufgebaut werden muss. Besteht keine Verbindung mehr, werden die "Security Associations" (SA) von Phase 2 vorzeitig beendet. Bei deaktivierter DPD wird die SA erst nach Ablauf der SA-Lebensdauer beendet. Zur Einstellung der SA-Lebensdauer siehe Abschnitt: VPN-Gruppeneigenschaften projektieren (Seite 466).

Erlaubnis zur Initiierung des Verbindungsaufbaus

Sie können die Erlaubnis zur Initiierung des VPN-Verbindungsaufbaus auf bestimmte Module im VPN beschränken.

Maßgebend für die Einstellung des beschriebenen Parameters ist die Vergabe der Adresse für das Gateway des zu projektierenden Moduls. Bei einer statisch vergebenen IP-Adresse kann das Modul von der Gegenstelle gefunden werden. Bei dynamisch vergebener, und sich

daher ständig ändernder IP-Adresse, kann die Gegenstelle nicht ohne Weiteres eine Verbindung aufbauen.

Modus	Bedeutung
Starte Verbindung zur Gegenstelle (Initiator/Responder) (Standard)	Bei dieser Option ist das Modul "aktiv", d. h. es versucht, eine Verbindung zur Gegenstelle mit fester IP-Adresse herzustellen. Die Annahme von Anfragen zum VPN-Verbindungsaufbau ist ebenfalls möglich. Diese Option wird empfohlen, wenn Sie von dem Provider für das Gateway des hier zu projektierenden Security-Moduls eine dynamische IP-Adresse zugewiesen bekommen haben. Die Adressierung der Gegenstelle erfolgt über deren projektierte WAN-IP-Adresse, deren projektierte externe Baugruppen-IP-Adresse oder den projektierten FQDN.
Warte auf Gegenstelle (Responder)	Bei dieser Option ist das Modul "passiv", d. h. es wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird. Diese Option wird empfohlen, wenn Sie von dem Provider für das Gateway des hier zu projektierenden Security-Moduls eine statische IP-Adresse zugewiesen bekommen haben. Sie erreichen dadurch, dass Verbindungsaufbauversuche nur durch die Gegenstelle gestartet werden. Diese Gegenstelle kann beispielsweise eine dynamische WAN-IP-Adresse besitzen.

Hinweis

Stellen Sie nicht alle Module einer VPN-Gruppe auf "Warte auf Gegenstelle", da sonst keine Verbindung aufgebaut wird.

WAN-IP-Adresse / FQDN - Adressen der Module und Gateways bei einem VPN über Internet

Beim Betrieb eines VPN mit IPsec-Tunnel über das Internet sind in der Regel zusätzliche IP-Adressen für die Internet-Gateways, z. B. DSL-Router, erforderlich. Den einzelnen Security- oder SCALANCE M-Modulen müssen die externen IP-Adressen der Partner-Module im VPN bekannt sein.

Hinweis

Für die Nutzung eines WAN als externes, öffentliches Netzwerk geben Sie als externe IP-Adresse Ihre vom Provider erhaltene IP-Adresse an, über welche das Security-Modul dann im WAN (Internet) erreichbar ist. Damit das Security-Modul Pakete über das WAN versenden kann, müssen Sie als "Standard-Router" Ihren DSL-Router eintragen.

Wenn Sie einen DSL-Router als Internet-Gateway nutzen, müssen an diesem mindestens die folgenden Ports gemäß der Angaben in der zugehörigen Dokumentation freigeschaltet werden:

- Port 500 (ISAKMP)
- Port 4500 (NAT-T)

Hierzu besteht die Möglichkeit, in den modulspezifischen VPN-Einstellungen eine IP-Adresse als "WAN-IP-Adresse" zuzuordnen. Beim Laden der Modulkonfiguration werden den

Gruppenteilnehmern dann die WAN-IP-Adressen der Partner-Module mitgeteilt. Alternativ zu einer WAN-IP-Adresse können Sie auch einen FQDN eintragen. Je nachdem, welche Adressen vorhanden sind, werden VPN-Endpunkte standardmäßig gemäß der folgenden Prioritäten verwendet:

1. WAN-Adresse
2. FQDN des primären dyn. DNS-Diensts
3. FQDN des sekundären dyn. DNS-Diensts
4. Externe IP-Adresse / DMZ-IP-Adresse des Security-Moduls

Anmerkung: Nach dem Entfernen einer vorhandenen WAN-Adresse wird stets die externe IP-Adresse / DMZ-IP-Adresse verwendet.

In der Schnittstellenauswahl der verbindungsspezifischen VPN-Einstellungen können Sie selbst bestimmen, welche Adresse dem Partner mitgeteilt werden soll. Hier können Sie auch einstellen, über welche Schnittstelle die Teilnehmer einer VPN-Gruppe kommunizieren sollen und welches Security-Modul die Erlaubnis zum Verbindungsaufbau hat.

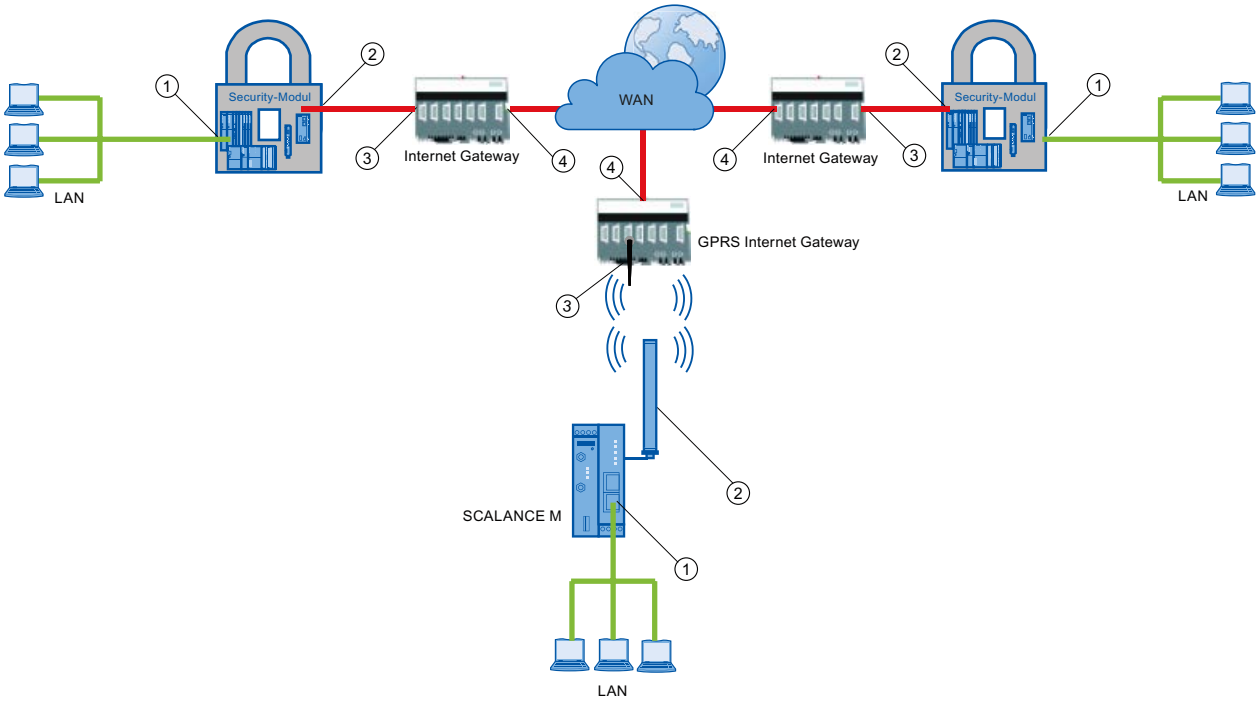

① IP-Adresse intern - eines Moduls
② IP-Adresse extern - eines Moduls
③ IP-Adresse eines Internet Gateways (z. B. GPRS-Gateway)
④ IP-Adresse (WAN-IP-Adresse) eines Internet Gateways (z. B. DSL-Router)

Responder-Einstellungen für Road Warrior Verbindungen

Der VPN-Verbindungsaufbau durch Unknown Peers (Road Warriors) wird vom Responder standardmäßig mit den sicheren VPN-Eigenschaften des VPN-Profils 1 zugelassen. Den VPN-Verbindungsaufbau durch Unknown Peers sowie die dabei verwendeten Eigenschaften können Sie über die Auswahl eines VPN-Gruppen-Profils und über die Auswahl von VPN-Profilen zulassen oder sperren.

Bedienelement	Bedeutung
VPN-Verbindungsaufbau durch Road Warriors zulassen	Der Responder lässt den VPN-Verbindungsaufbau durch Unknown Peers grundsätzlich zu.
VPN-Profile 1 - 5 im Auswahldialog für VPN-Profile	Der Responder lässt den VPN-Verbindungsaufbau durch Unknown Peers mit den Eigenschaften der vordefinierten VPN-Profile 1 - 5 zu. Die Eigenschaften dieser VPN-Profile entsprechen den Eigenschaften der Fallback-VPN-Profile, die Sie für VPN-Gruppen auswählen können, siehe Kapitel VPN-Gruppeneigenschaften projektieren (Seite 466).
VPN-Gruppen-Profile im Auswahldialog für VPN-Profile	Der Responder lässt den VPN-Verbindungsaufbau durch Unknown Peers mit den Eigenschaften der VPN-Gruppe zu, die aus dem Auswahldialog für VPN-Profile über das Haken-Symbol ausgewählt wird. Es stehen diejenigen VPN-Gruppen zur Auswahl, in denen sich der Responder befindet. Es kann maximal eine VPN-Gruppe ausgewählt und als VPN-Gruppenprofil verwendet werden. Über die Schaltfläche "Hinzufügen" können neue VPN-Gruppen angelegt werden, denen der Responder automatisch zugeordnet wird. Diese VPN-Gruppen stehen danach für die Bildung eines VPN-Gruppenprofils zur Auswahl. Wenn der VPN-Verbindungsaufbau durch Unknown Peers mit den Eigenschaften der VPN-Gruppe zugelassen werden soll, in der sich der Responder und die Unknown Peers befinden, muss diese VPN-Gruppe aus dem Auswahldialog ausgewählt werden.

Nach dem Öffnen eines STEP 7 Projekts, das zuletzt mit einer älteren Version als V14 SP1 gespeichert wurde, ist standardmäßig das Kontrollkästchen "VPN-Verbindungsaufbau durch Road Warriors zulassen" aktiviert. Zudem sind dem Responder die vordefinierten VPN-Profile 2 - 5 zugeordnet.

Für VPN-Verbindungen von SOFTNET Security Client / SCALANCE M-800 Baugruppen zu SCALANCE S / CPs wird stets das ausgewählte VPN-Profil mit der höchsten Priorität verwendet und ausgeleitet. Das Authentifizierungsverfahren des ausgewählten VPN-Profils muss mit dem Authentifizierungsverfahren der zugehörigen VPN-Gruppe übereinstimmen. Die Priorität der VPN-Profile lautet:

1. Ausgewähltes VPN-Gruppen-Profil
2. VPN-Profil 1
3. VPN-Profil 2
4. VPN-Profil 3
5. VPN-Profil 4
6. VPN-Profil 5

Für VPN-Verbindungen von SCALANCE M875 Baugruppen zu SCALANCE S / CPs muss das VPN-Profil 3 ausgewählt werden, wenn das Authentifizierungsverfahren der zugehörigen VPN-Gruppe "Zertifikat" ist. Wenn das Authentifizierungsverfahren der zugehörigen VPN-Gruppe "Pre-shared key" ist, muss das VPN-Profil 5 ausgewählt werden.

Interne Netzknoten konfigurieren

Die Konfiguration interner Netzknoten wird in folgendem Kapitel beschrieben:
AUTOHOTSPOT

Siehe auch

So legen Sie mittels VPN-Gruppen IPsec-Tunnel an (Seite 460)

OpenVPN-Tunnel zu SINEMA Remote Connect Server herstellen

Modulspezifische Funktion

Diese Funktion ist nur für die folgenden Baugruppen verfügbar:

- CP 1243-1 ab V3.1
- CP 1243-8 IRC ab V3.1
- CP 1243-7 LTE ab V3.1
- CP 1542SP-1 IRC ab V2
- CP 1543SP-1 ab V2

Detaillierte Informationen finden Sie in Abschnitt Automatische OpenVPN-Konfiguration und Open-VPN-Tunnelaufbau (Seite 583) des Kapitels "Security für S7-1200-/S7-1500-CPs".

Router- und Firewallredundanz projektieren

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S623/S627-2M ab V4 verfügbar, siehe Abschnitt:
AUTOHOTSPOT

Online-Funktionen - Diagnose und Logging

Diagnose und Logging in der Übersicht

Zu Test- und Überwachungszwecken verfügt das Security-Modul über Diagnose- und Logging-Funktionen.

- Diagnosefunktionen
  Hierunter sind verschiedene System- und Statusfunktionen zu verstehen, die Sie bei einer bestehenden Netzwerkverbindung zum Security-Modul anwenden können.
- Logging-Funktionen
  Hierbei geht es um die Aufzeichnung von System- und Sicherheitsereignissen sowie von Datenpaketen.

Ereignisse mit Logging-Funktionen aufzeichnen

Welche Ereignisse aufgezeichnet werden sollen, legen Sie mit den Log-Einstellungen zum jeweiligen Security-Modul fest.

Dabei können Sie für die Aufzeichnung folgende Varianten konfigurieren:

- Lokales Logging
  Bei dieser Variante zeichnen Sie die Ereignisse in lokalen Puffern des Security-Moduls auf. Im "Online & Diagnose"-Dialog können Sie dann auf diese Aufzeichnungen zugreifen, diese sichtbar machen und in der Service-Station archivieren. Die Auswertung der Pufferbereiche des Security-Moduls setzt eine Netzwerkverbindung zum ausgewählten Security-Modul voraus.
- Netzwerk-Syslog
  Beim Netzwerk-Syslog nutzen Sie einen im Netz vorhandenen Syslog-Server. Dieser zeichnet die Ereignisse entsprechend der Konfiguration in den Log-Einstellungen des jeweiligen Security-Moduls auf.

Log-Daten archivieren und aus Datei einlesen

Sie können die aufgezeichneten Ereignisse zur Archivierung in einer Log-Datei speichern und diese später auch ohne eine bestehende Netzwerkverbindung zum Security-Modul öffnen. Die Funktion zum Öffnen der Log-Dateien finden Sie in den globalen Security-Funktionen.

Diagnose im Ghost-Modus (nur für SCALANCE S602 ab V3.1)

Wenn Sie das Security-Modul im Ghost-Modus betreiben, nimmt die externe Schnittstelle des Security-Moduls zur Laufzeit die IP-Adresse des internen Teilnehmers an. Bevor Sie die Diagnose eines Security-Moduls im Ghost-Modus durchführen können, müssen Sie die Verbindung mit dem Security-Modul über die IP-Adresse herstellen, die das Security-Modul zur Laufzeit vom internen Teilnehmer bezogen hat.

Um herauszufinden, welche IP-Adresse das Security-Modul aktuell hat, können Sie in STEP 7 über den Menübefehl "Online" > "Erreichbare Teilnehmer" nach erreichbaren Teilnehmern suchen.

Exportierte Log-Dateien vor unberechtigtem Zugriff schützen

Aus STEP 7 exportierte Log-Dateien können sicherheitsrelevante Informationen enthalten. Stellen Sie deshalb sicher, dass diese Dateien vor unberechtigtem Zugriff geschützt sind. Dies ist insbesondere bei der Weitergabe der Dateien zu beachten.

Funktionsübersicht Online-Dialog

Funktionen des "Online & Diagnose"-Dialogs

Das Security-Modul bietet folgende Funktionen im "Online & Diagnose"-Dialog von STEP 7:

Eintrag im Online-Dialog	Funktion
Zustand	Anzeige von Statusinformationen zum ausgewählten Security-Modul wie z.B. die aktuellen IP-Adressen der Schnittstellen sowie die laufende Uhrzeit und das Datum.
Schnittstellen-Einstellungen	Übersicht zu den Einstellungen der einzelnen Schnittstellen.
Dynamisches DNS	Übersicht zu den Einstellungen für dynamisches DNS.
System-Log	Anzeige von geloggten System-Ereignissen, Starten und Stoppen des Loggings (nur bei einer bestehenden Online-Verbindung zu SCALANCE S Modulen) sowie Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls.
Audit-Log	Anzeige von geloggten Sicherheits-Ereignissen, Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls.
Paketfilter-Log	Anzeige von geloggten Daten-Paketen, Starten und Stoppen des Loggings (nur bei einer bestehenden Online-Verbindung zu SCALANCE S Modulen) sowie Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls.
ARP-Tabelle	Anzeige der ARP-Tabelle des Security-Moduls.
Angemeldete Benutzer	Anzeige der Benutzer, die an der Internetseite für benutzerspezifische IP-Regelsätze angemeldet sind.
Kommunikationszustand	Anzeige von Statusinformationen zu VPN-Tunnelverbindungen sowie Teilnehmern von VPN-Gruppen, zu denen das ausgewählte Security-Modul gehört.
SINEMA RC - Automatische VPN-Konfiguration	Anzeige des Status der OpenVPN-Konfiguration sowie von OpenVPN-Verbindungen zum SINEMA RC-Server.
Interne Knoten	Anzeige der gelernten oder konfigurierten internen Netzknoten des Security-Moduls.
Dynamisch aktualisierte Firewall-Regeln	Anzeige der IP-Adressen, die über HTTP oder HTTPS dynamisch freigeschaltet oder von einem Benutzer nachgeladen wurden.
Firewall-Blacklist	Anzeige der IP-Adressen, die in die Blacklist der Firewall eingetragen wurden.
Ghost-Modus	Dialog für den Ghost-Modus des SCALANCE S602 mit Informationen zu den Adressparametern des internen Teilnehmers (identisch zur externen IP-Adresse des Security-Moduls) und zu IP-Adresswechseln des internen Teilnehmers.
Datum und Uhrzeit	Einstellung von Datum und Uhrzeit.
Firmware-Update	Aktualisierung der Firmware.

Zugriffsvoraussetzungen

Damit Sie die Online-Funktionen eines Security-Moduls nutzen können, müssen folgende Voraussetzungen erfüllt sein:

- eine Netzwerkverbindung zum ausgewählten Modul besteht
- das Projekt, mit dem das Modul konfiguriert wurde, ist geöffnet

- ein Nutzer mit den nötigen Rechten muss im Projekt angemeldet sein
- Für CPs muss der Diagnosezugang in der Firewall freigegeben sein (CP 343-1 Adv. / 443-1 Adv. und CP 1628: TCP 443; S7-1200-/S7-1500-CPs: TCP 8448)

Hinweis

Voraussetzung für die Online-Diagnose im Ghost-Modus (nur für SCALANCE S602 ab V3.1)

Die Online-Diagnose ist im Ghost-Modus erst dann verfügbar, wenn das Security-Modul die IP-Adresse des internen Teilnehmers gelernt und für ihre externe Schnittstelle übernommen hat. Danach ist das Security-Modul über die IP-Adresse der externen Schnittstelle erreichbar.

So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf das zu bearbeitende Modul.
2. Wählen Sie im Kontextmenü den Menübefehl "Online & Diagnose".
3. Falls noch keine Online-Verbindung zum Security-Modul hergestellt ist, klicken Sie im Eintrag "Diagnose" auf die Schaltfläche "Online verbinden".

Online-Einstellungen werden nicht in der Konfiguration gespeichert

Einstellungen, die Sie in der Online-Betriebsart vornehmen (z. B. Einstellungen zum Log-Speicher), werden nicht in der Konfiguration auf dem Security-Modul gespeichert. Nach einem Modul-Neuanlauf sind deshalb immer die Einstellungen aus der Projektierung wirksam.

Funktionen der Online-Diagnose

Statusinformationen des Security-Moduls - Eintrag "Zustand"

Bedeutung

Anzeige des Status des im Projekt angewählten Security-Moduls.

Tabelle 1-45 Online & Diagnose: Eintrag "Zustand"

System- und Statusfunktionen	Bedeutung
Übersicht	
Hardware-Typ	Der Typ des Security-Moduls.
Externe IP-Adresse	Die externe IP-Adresse des Security-Moduls. Für S7-1200-/S7-1500-CPs, CP 1628: Die IP-Adresse der Industrial Ethernet-Schnittstelle. Für CP 343-1 Adv. / 443-1 Adv.: Die IP-Adresse der GBit-Schnittstelle.

System- und Statusfunktionen	Bedeutung
Interne IP-Adresse	Die interne IP-Adresse des Security-Moduls. Für CP 1628: Die IP-Adresse der NDIS-Schnittstelle. Bei mehreren NDIS-Adressen wird nur eine angezeigt. Für CP 343-1 Adv. / 443-1 Adv.: Die IP-Adresse der PROFINET-Schnittstelle.
DMZ-IP-Adresse (nur SCALANCE S623 / S627-2M)	Die DMZ-IP-Adresse des Security-Moduls.
Tunnel-IP-Adresse (nur SCALANCE S612/S623/ S627-2M ab V4)	Die erste Alias-Tunnel-IP-Adresse des Security-Moduls im VPN-Tunnel.
Seriennummer	Die Seriennummer des Security-Moduls.
Artikelnummer	Die Artikelnummer des Security-Moduls, die bei einer Bestellung verwendet wird.
Firmware-Version	Der Firmware-Ausgabestand des Security-Moduls.
Betriebsart	Betriebsart des Security-Moduls (Bridge-Modus / Routing-Modus)
Externe MAC-Adresse	Die externe MAC-Adresse des Security-Moduls. Für S7-1200-/S7-1500-CPs, CP 1628: Die MAC-Adresse der Industrial Ethernet-Schnittstelle. Für CP 343-1 Adv. / 443-1 Adv.: Die MAC-Adresse der GBit-Schnittstelle.
Interne MAC-Adresse	Die interne MAC-Adresse des Security-Moduls. Für CP 1628: Die MAC-Adresse der NDIS-Schnittstelle. Für CP 343-1 Adv. / 443-1 Adv.: Die MAC-Adresse der PROFINET-Schnittstelle.
DMZ-MAC-Adresse (nur SCALANCE S623 / S627-2M)	Die DMZ-MAC-Adresse des Security-Moduls
HW-Release	Der Hardware-Erzeugnisstand des Security-Moduls.
C-PLUG	Anzeige, ob ein C-PLUG gesteckt ist oder nicht.
Alias-IP-Adressen (nur für SCALANCE S ab V4)	
IP-Adresse	Alias-IP-Adresse, die durch eine NAT-Regel an einer Schnittstelle des Security-Moduls registriert wurde.
Zugehörige Schnittstelle	Schnittstelle des Security-Moduls, an der die Alias-IP-Adresse registriert wurde.
Lokale Zeit	

System- und Statusfunktionen	Bedeutung
Aktuelle Zeit	Datum und Uhrzeit, die auf dem Security-Modul angezeigt werden. Format bei Oberflächensprache "Deutsch": dd.mm.yyyy (Datum) hh:mm:ss (Uhrzeit) Format bei Oberflächensprache "Englisch": mm/dd/yyyy (Datum) hh:mm:ss AM/PM (Uhrzeit) Format bei Oberflächensprache "Französisch", "Italienisch" und "Spanisch": dd/mm/yyyy (Datum) hh:mm:ss (Uhrzeit) Format bei Oberflächensprache "Chinesisch": yyyy/mm/dd (Datum) hh:mm:ss **Hinweis (nicht für CPs)** Die lokale Zeit auf dem SCALANCE S Modul stellen Sie unter "Funktionen" > "Datum und Uhrzeit" ein.
Betriebsdauer	Zeit, die seit dem letzten Neustart des Security-Moduls vergangen ist. Format bei Oberflächensprache "Deutsch", "Englisch", "Französisch", "Italienisch", "Spanisch" und "Chinesisch": dddd.hh:mm:ss
Uhrzeit Quelle	Die Quelle, von der Datum und Uhrzeit bezogen werden.
Konfiguration	
Erzeugt	Datum und Uhrzeit, zu denen das Projekt zum ersten Mal erstellt wurde. Format bei Oberflächensprache "Deutsch", "Englisch", "Französisch", "Italienisch", "Spanisch" und "Chinesisch": dd.mm.yyyy (Datum) hh:mm:ss (Uhrzeit)
Name	Dateiname des Projekts, das zuletzt geladen wurde.
Autor	Name des Benutzers, der das Projekt erstellt hat. Wird aus den Projekteigenschaften übernommen.
Übersetzt	Datum und Uhrzeit, zu denen das Projekt zum letzten Mal übersetzt wurde. Format bei Oberflächensprache "Deutsch", "Englisch", "Französisch", "Italienisch", "Spanisch" und "Chinesisch": dd.mm.yyyy (Datum) hh:mm:ss (Uhrzeit)
Speicherort	Angabe des Ortes (z.B. Stadt), der in den Eigenschaften eines SCT-Projekts eingetragen wurde.
Dateisystem (nicht für CPs)	
RAM	Anzeige, wie viel RAM und Flash im Dateisystem belegt sind.
Flash	

Übersicht zu den einzelnen Schnittstellen - Eintrag "Schnittstellen-Einstellungen"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V3 verfügbar, siehe Abschnitt:
Übersicht zu den einzelnen Schnittstellen - Eintrag "Schnittstellen-Einstellungen" (Seite 549)

Übersicht zu Dyn. DNS-Einstellungen - Eintrag "Dynamisches DNS"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V3 verfügbar, siehe Abschnitt:
Übersicht zu Dyn. DNS-Einstellungen - Eintrag "Dynamisches DNS" (Seite 550)

Anzeige der ARP-Tabelle - Eintrag "ARP-Tabelle"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V3 verfügbar, siehe Abschnitt:
Anzeige der ARP-Tabelle - Eintrag "ARP-Tabelle" (Seite 552)

An Webseite angemeldete Benutzer - Eintrag "Angemeldete Benutzer"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V3 verfügbar, siehe Abschnitt:
An Webseite angemeldete Benutzer - Eintrag "Angemeldete Benutzer" (Seite 552)

VPN-Verbindungen des Security-Moduls - Eintrag "Kommunikationszustand"

Bedeutung

Anzeige des Kommunikationsstatus der folgenden Netzkomponenten:

- Andere Security-Module der VPN-Gruppe, zu der das ausgewählte Security-Modul gehört
- Interne Netzknoten dieser Security-Module

Tabelle 1-46 Online & Diagnose: Eintrag "Kommunikationszustand"

System- und Statusfunktionen	Bedeutung
Bekannte Security-Geräte oder -Module	Anzeige der Teilnehmer, mit denen sich das ausgewählte Security-Modul in einer VPN-Gruppe befindet. Außerdem wird angezeigt, ob der Tunnelzustand aktiv ist oder nicht. Damit Sie zusätzliche Informationen zu einem der Teilnehmer erhalten, markieren Sie diesen in der Liste. **Hinweis:** Projektierte, aber nicht aktive Tunnel werden ausschließlich für CPs angezeigt.
Endpunkte	Anzeige von Informationen zu den internen Netzknoten des Security-Moduls, das Sie in der Tabelle "Bekannte Security-Geräte oder -Module" ausgewählt haben. Für jeden internen Netzknoten wird angezeigt, ob er gelernt oder konfiguriert wurde. Zudem wird angezeigt, in welchem Subnetz sich der interne Netzknoten befindet. Bei SCALANCE S Modulen wird das Subnetz von Netzknoten nur im Bridge-Modus angezeigt.
Tunneleigenschaften	Anzeige von Eigenschaften des VPN-Tunnels, der zu dem Security-Modul aufgebaut wurde, das Sie in der Tabelle "Bekannte Security-Geräte oder -Module" ausgewählt haben.

Automatische OpenVPN-Konfiguration über SINEMA RC-Server

Modulspezifische Funktion

Diese Funktion ist nur für die folgenden Module verfügbar:

- CP 1243-1 ab V3.1
- CP 1243-8 IRC ab V3.1
- CP 1243-7 LTE ab V3.1
- CP 1542SP-1 IRC ab V2
- CP 1543SP-1 ab V2

Detaillierte Informationen finden Sie in Abschnitt Diagnose der automatischen OpenVPN-Konfiguration und OpenVPN-Verbindungen (Seite 584).

Gefundene interne Netzknoten - Eintrag "Interne Knoten"

Bedeutung

Anzeige aller gelernten und konfigurierten Netzknoten. Zudem wird angezeigt, ob der Lernmodus des Security-Moduls aktiviert oder deaktiviert ist.

Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln"

Modulspezifische Funktion

Diese Funktion ist nur für CP 343-1 Adv. / 443-1 Adv. verfügbar, siehe Abschnitt: Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln" (Seite 570) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

Anzeige der Firewall-Blacklist - Eintrag "Firewall-Blacklist"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V4 verfügbar, siehe Abschnitt: Anzeige der Firewall-Blacklist - Eintrag "Firewall-Blacklist" (Seite 553)

Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S verfügbar, siehe Abschnitt: Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit" (Seite 553).

Diagnose im Ghost-Modus - Eintrag "Ghost-Modus"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S602 ab V3.1 verfügbar, siehe Abschnitt: Diagnose im Ghost-Modus - Eintrag "Ghost-Modus" (Seite 553)

Logging-Funktionen

System-Ereignisse aufzeichnen - Eintrag "System-Log"

Bedeutung

Anzeige von geloggten System-Ereignissen sowie Starten und Stoppen des Lesens von System-Ereignissen aus dem lokalen Speicher des Security-Moduls.

Das System-Log zeichnet automatisch fortlaufend System-Ereignisse wie z. B. den Start eines Prozesses auf. Anhand von Ereignisklassen ist die Aufzeichnung skalierbar.

System- und Statusfunktionen	Bedeutung
Starte/Stoppe Logging (nicht für CPs)	Startet/Stoppt das Aufzeichnen von System-Ereignissen. Das Verfahren sowie die Ereignisklassen, die aufgezeichnet werden, werden in den lokalen Security-Einstellungen projektiert.
Starte/Stoppe Lesen	Startet/Stoppt das Auslesen von System-Ereignissen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an. **Hinweis** Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden.
Anzeige löschen	Löscht die in der Tabelle angezeigten Log-Daten.

Log-Dateien mit gespeicherten System-Ereignissen können Sie in den globalen Security-Funktionen unter "Log-Dateien (Offline-Ansicht)" öffnen.

Sicherheits-Ereignisse aufzeichnen - Eintrag "Audit-Log"

Bedeutung

Anzeige von geloggten Sicherheits-Ereignissen sowie Starten und Stoppen des Lesens von Sicherheits-Ereignissen aus dem lokalen Speicher des Security-Moduls.

Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf. Dazu gehören z. B. Benutzeraktionen wie das Ein- und Ausschalten des Paket-Logging.

System- und Statusfunktionen	Bedeutung
Starte/Stoppe Lesen	Startet/Stoppt das Auslesen von Sicherheits-Ereignissen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an. **Hinweis** Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden.
Anzeige löschen	Löscht die in der Tabelle angezeigten Log-Daten.

Log-Dateien mit gespeicherten Audit-Ereignissen können Sie in den globalen Security-Funktionen unter "Log-Dateien (Offline-Ansicht)" öffnen.

Datenpakete aufzeichnen - Eintrag "Paketfilter-Log"

Bedeutung

Anzeige von geloggten Datenpaketen sowie Starten und Stoppen des Lesens von Paketfilter-Ereignissen.

Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur Datenpakete geloggt, auf die eine projektierte Paketfilter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paketfilter-Regel aktiviert ist.

Informationen zum Aktivieren des Logging finden Sie in Kapitel:
AUTOHOTSPOT.

Die geloggten Daten können neben dem Lesen aus dem Puffer und dem Übertragen in die Anzeige zusätzlich zur Archivierung in einer Datei gespeichert werden.

System- und Statusfunktionen	Bedeutung
Starte/Stoppe Logging (nicht für CPs)	Startet/Stoppt das Aufzeichnen von Datenpaketen. Das Verfahren, nach dem die Daten aufgezeichnet werden, wird in den lokalen Security-Einstellungen projektiert.
Starte/Stoppe Lesen	Startet/Stoppt das Auslesen von aufgezeichneten Datenpaketen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an. **Hinweis** Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden.
Anzeige löschen	Löscht die in der Tabelle angezeigten Log-Daten.
Log-Kategorie	Wählen Sie aus, von welchen Datenpaketen das Logging angezeigt werden soll. Die Auswahl ist abhängig von den Einstellungen, die Sie offline in den lokalen Security-Einstellungen projektiert haben. Es werden nur die Datenpakete geloggt, für die das Logging aktiviert wurde. Wählen Sie eine Kategorie aus, für die das Logging nicht aktiviert wurde, werden für diese Kategorie auch keine Daten aufgezeichnet.

Log-Dateien mit gespeicherten Paketfilter-Ereignissen können Sie in den globalen Security-Funktionen unter "Log-Dateien (Offline-Ansicht)" öffnen.

Ladefunktionen

Besonderheiten beim Laden von Security-Konfigurationen

Security-Konfigurationen können die Erreichbarkeit des Security-Moduls durch den Projektierungs-PC beeinflussen. Dies ist beispielsweise der Fall, wenn in einer Konfiguration für ein Security-Modul eine Tunnelverbindung zu einem anderen Security-Modul projektiert und diese Konfiguration vom Projektierungs-PC auf das Security-Modul geladen wird. Das Security-Modul ist nach dem Laden durch den Projektierungs-PC nicht mehr erreichbar und der Erreichbarkeitstest, der durch STEP 7 standardmäßig nach dem Laden einer Konfiguration durchgeführt wird, schlägt fehl. Die daraufhin von STEP 7 ausgegebene Fehlermeldung bezieht sich ausschließlich auf den Erreichbarkeitstest; das eigentliche Laden der Konfiguration ist bei konsistenten Projektdaten sowie korrekter IP-Adressbeziehung zwischen Security-Modul und Projektierungs-PC gewährleistet.

Besonderheiten beim Laden von Konfigurationen und Firmware in SCALANCE S Module sind in folgendem Abschnitt beschrieben:
AUTOHOTSPOT

Laden von Konfigurationen auf die Engineering-Station

Das Laden einer Konfiguration aus einem SCALANCE S Modul oder aus dem CP 1628 auf eine Engineering-Station ist nicht möglich.

Das Laden einer Konfiguration aus einem S7-CP, der Security unterstützt, auf eine Engineering-Station ist auch möglich, wenn in der Konfiguration Security-Funktionen projektiert wurden. Projektierte Security-Funktionen werden jedoch nicht auf die Engineering-Station übertragen. In der übertragenen Konfiguration auf der Engineering-Station ist zudem das Kontrollkästchen "Aktiviere Security-Funktionen" deaktiviert.

SOFTNET Security Client

Einsatz des SOFTNET Security Client

Einsatzbereich - Zugriff über VPN

Mit der PC-Software SOFTNET Security Client (SSC) sind gesicherte Fernzugriffe vom PG/PC auf Automatisierungsgeräte, die durch ein Security-Modul geschützt sind, über öffentliche Netze hinweg, möglich. Für S7-300-/S7-400-CPs sowie für den PC-CP CP 1628 wird der SOFTNET Security Client V4.0 HF1 benötigt. Mit SOFTNET Security Client ≤ V4.0 sind diese CPs nicht freigegeben. Für weitere CPs ist der SOFTNET Security Client nicht freigegeben.

Mittels des SOFTNET Security Clients wird ein PG/PC automatisch so konfiguriert, dass er eine gesicherte IPsec-Tunnelkommunikation im VPN (Virtual Private Network) zu einem oder mehreren Security-Modulen aufbauen kann.

Durch diese IPsec-Tunnelkommunikation ist es möglich, mit PG/PC-Applikationen wie NCM Diagnose auf gesichertem Weg auf Geräte oder Netzwerke zuzugreifen, die sich in einem durch das Security-Modul geschützten internen Netz befinden.

Wie funktioniert der SOFTNET Security Client?

Der SOFTNET Security Client liest die von STEP 7 erstellte Konfiguration ein und ermittelt ggf. die aus der Datei zu importierenden Zertifikate.

Das CA-Zertifikat und die privaten Schlüssel werden ggf. importiert und im lokalen PG/PC abgelegt.

Anschließend werden mit den Daten aus der Konfiguration Security-Einstellungen vorgenommen, damit Applikationen auf IP-Adressen hinter den Security-Modulen zugreifen können.

Ist der Lernmodus für die internen Teilnehmer bzw. Automatisierungsgeräte aktiviert, wird zunächst eine Sicherheitsrichtlinie für den gesicherten Zugriff auf Security-Module eingestellt. Danach spricht der SOFTNET Security Client die Security-Module an, um die IP-Adressen der jeweils internen Teilnehmer zu ermitteln.

Der SOFTNET Security Client trägt diese IP-Adressen in spezielle Filterlisten dieser Sicherheitsrichtlinie ein. Anschließend können Applikationen mit den Automatisierungsgeräten über VPN kommunizieren.

Hinweis

Zusätzliche Security-Maßnahmen bei der Verwendung des SOFTNET Security Clients

Der SOFTNET Security Client bietet eine Lösung für die sichere Kommunikation mit Automatisierungszellen über VPN. Für den Eigenschutz des PCs/PGs sowie der damit verbundenen Automatisierungszelle wird empfohlen, zusätzliche Maßnahmen wie z.B. Virenscanner sowie die Windows-Firewall einzusetzen.

Unter Windows 7 muss die Firewall des Betriebssystems aktiviert sein, damit der VPN-Tunnelaufbau funktioniert.

Konfigurationsdatei in STEP 7 erstellen

SOFTNET Security Client-Modul im Projekt konfigurieren

Der SOFTNET Security Client wird im Projekt als Modul angelegt. Im Gegensatz zu den anderen Security-Modulen müssen Sie keine weiteren Eigenschaften projektieren.

Weisen Sie das SSC-Modul der oder den VPN-Gruppen zu, in denen IPsec-Tunnel zum PG/PC eingerichtet werden sollen.

Hinweis

Beachten Sie die Angaben zu den Parametern im Kapitel:
Security-Modul in konfigurierte VPN-Gruppe aufnehmen (Seite 479).

Hinweis

Wenn Sie mehrere SOFTNET Security Clients innerhalb einer Gruppe anlegen, werden keine Tunnel zwischen diesen Clients aufgebaut, sondern nur vom jeweiligen Client zu den Security-Modulen.

Konfigurationsdateien für den SOFTNET Security Client

Die Schnittstelle zwischen STEP 7 als Projektierwerkzeug und dem SOFTNET Security Client wird über Konfigurationsdateien bedient.

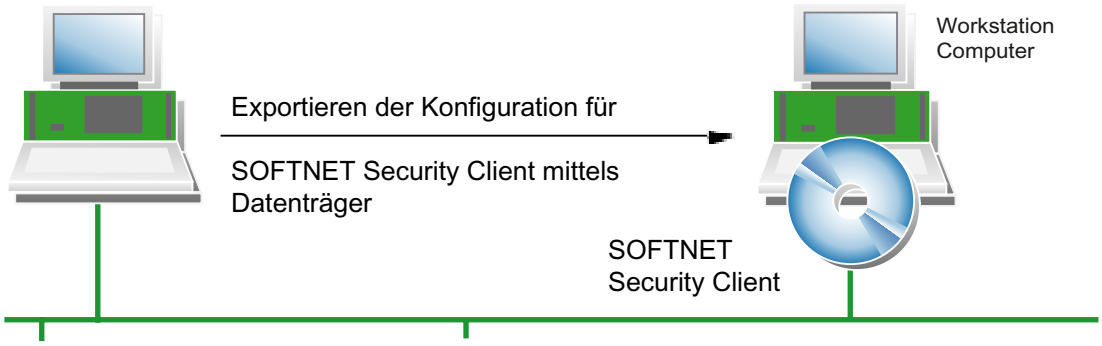

Die Konfiguration wird in den Dateitypen "*.dat", "*.p12" und "*.cer" hinterlegt.

Vorgehensweise

Um die Konfigurationsdateien zu erzeugen, führen Sie in STEP 7 folgende Schritte aus:

1. Selektieren Sie in der Ansicht "Geräte & Netze" das Register "Topologiesicht" oder das Register "Netzsicht".
2. Fügen Sie aus dem Hardware-Katalog über die Auswahl "Netzkomponenten > Industrial Security" ein "Security-Gerät" in das ausgewählte Register ein.
3. Wählen Sie in den lokalen Security-Einstellungen unter "VPN-Gerät > VPN-Betriebsart" den SOFTNET Security Client V5 aus.
4. Ordnen Sie den SOFTNET Security Client den VPN-Gruppen zu, in denen der PG/PC über IPsec-Tunnel kommunizieren soll.
5. Stellen Sie sicher, dass in den lokalen Security-Einstellungen des SOFTNET Security Clients unter dem Eintrag "Konfiguration des SOFTNET Security Clients" das Kontrollkästchen "SSC-Dateien generieren" aktiviert ist.
6. Wählen Sie den Speicherort für die Konfigurationsdateien.
7. Übersetzen Sie die Konfiguration des SOFTNET Security Clients, um die Konfigurationsdatei zu exportieren.
8. Wenn Sie als Authentifizierungsmethode "Zertifikat" gewählt haben, geben Sie ein Passwort für das Zertifikat der VPN-Konfiguration an. Vergeben Sie kein Passwort, wird der Projektname (nicht das Projektpasswort) als Passwort übernommen.
   Ergebnis: Der Export der Konfigurationsdateien ist abgeschlossen.
9. Übernehmen Sie die Dateien vom Typ *.dat, *.p12, *.cer auf den PG/PC, auf dem Sie den SOFTNET Security Client betreiben möchten.

Hinweis

Exportierte Konfigurationsdateien vor unberechtigtem Zugriff schützen

Aus STEP 7 exportierte Konfigurationsdateien für SOFTNET Security Client können sicherheitsrelevante Informationen enthalten. Stellen Sie deshalb sicher, dass diese Dateien vor unberechtigtem Zugriff geschützt sind. Dies ist insbesondere bei der Weitergabe der Dateien zu beachten.

SCALANCE S-600 V3

Security-Modul austauschen

Voraussetzung

Um Security-Module austauschen zu können, müssen deren Modulbeschreibungen aktuell sein. Um die Modulbeschreibung von Security-Modulen zu aktualisieren, gehen Sie wie folgt vor:

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Klicken Sie in den lokalen Security-Einstellungen auf den Eintrag "Allgemein" > "Kataloginformation".
3. Klicken Sie auf die Schaltfläche "Modulbeschreibung aktualisieren".

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Security-Modul in der Topologie- oder Netzsicht.
2. Klicken Sie mit der rechten Maustaste auf das Security-Modul und wählen Sie im Kontextmenü den Menübefehl "Gerät tauschen...".
   Entnehmen Sie der folgenden Tabelle, welche Security-Module Sie ohne und mit eventuellen Datenverlusten austauschen können.

Aus-gangsmo-dul	Möglicher Modultausch									
	S602 V3	S602 V3.1	S602 V4	S612 V3	S612 V4	S623 V3	S623 V4	S623 V4.0.1	S627-2 M V4	S627-2 M V4.0.1
S602 V3	-	x	x	x	x	x	x	x	x	x
S602 V3.1	!	-	x	!	!	!	!	!	!	!
S602 V4	!	!	-	!	!	!	!	!	!	!
S612 V3	!	!	!	-	x	x	x	x	x	x
S612 V4	!	!	!	!	-	!	x	x	x	x
S623 V3	!	!	!	!	!	-	x	x	x	x
S623 V4	!	!	!	!	!	!	-	x	x	x
S623 V4.0.1	!	!	!	!	!	!	!	-	!	x

Aus-gangsmo-dul	Möglicher Modultausch									
	S602 V3	S602 V3.1	S602 V4	S612 V3	S612 V4	S623 V3	S623 V4	S623 V4.0.1	S627-2 M V4	S627-2 M V4.0.1
S627-2M V4	!	!	!	!	!	!	!	!	-	x
S627-2M V4.0.1	!	!	!	!	!	!	!	!	!	-

x Ohne Verluste

! Mit eventuellen Verlusten

- Der Modultyp und die Firmwareversion werden nicht geändert.

Schnittstellen für SCALANCE S Module projektieren

Übersicht

Projektierung der Betriebsart

Über die Betriebsart legen Sie den Modus für das Schnittstellenrouting (Extern/Intern) fest. Die DMZ-Schnittstelle des Security-Moduls (nur SCALANCE S623/S627-2M) wird immer im Routing-Modus angebunden. Nähere Informationen finden Sie im Abschnitt: IP-Adressparameter projektieren (Seite 505)

Projektierung der Schnittstellen

Wenn die externe Schnittstelle, die DMZ-Schnittstelle (nur SCALANCE S623/S627-2M) oder die Tunnel-Schnittstelle (nur SCALANCE S612/S623/S627-2M ab V4 in VPN-Gruppe(n)) eines Security-Moduls projektiert werden soll, müssen diese über das Kontrollkästchen "Schnittstelle aktivieren" aktiviert sein. Legen Sie die IP-Adressangaben pro Schnittstelle sowie Einstellungen einzelner Ports fest. Für die Zuweisung einer IP-Adresse stehen für die externe Schnittstelle und für die DMZ-Schnittstelle (nur SCALANCE S623/S627-2M) im Eintrag "Allgemein" folgende Zuweisungsmodi zur Verfügung:

- Statische IP-Adresse mit Subnetzmaske. Nähere Informationen finden Sie im Abschnitt: IP-Adressparameter projektieren (Seite 505)
- Adresszuweisung über PPPoE. Nähere Informationen finden Sie im Abschnitt: Internetverbindung projektieren (Seite 507)
  Die interne Schnittstelle sowie die Tunnel-Schnittstelle kann nur über eine statische IP-Adresse projektiert werden.

Wenn durch die Projektierung von NAT-Regeln Alias-IP-Adressen an den Schnittstellen des Security-Moduls registriert wurden, werden diese unter dem Eintrag "Alias-IP-Adressen" angezeigt.

Hinweis

Externe Schnittstelle und DMZ-Schnittstelle als Internetzugang

Der gleichzeitige Betrieb von PPPoE an der externen Schnittstelle und an der DMZ-Schnittstelle (Dual-ISP) ist nicht möglich.

Point to Point Protocol over Ethernet (PPPoE)

Um einen Internet-/WAN-Anschluss direkt über ein DSL-Modem zu ermöglichen, erfolgt die Zuweisung der IP-Adresse an der externen Schnittstelle bzw. an der DMZ-Schnittstelle über PPPoE. Bei PPPoE handelt es sich um ein Einwahlprotokoll zum Bezug von IP-Adressen von einem Internet Service Provider (ISP). SCALANCE S wird dabei im Routing-Modus betrieben.

Zur Verwendung dieser IP-Adresszuweisungsmethode geben Sie Angaben zum ISP im Eintrag "PPPoE" an. Die IP-Adresse, die Subnetzmaske, der Standard-Router sowie der DNS-Server der Schnittstelle werden dann vom ISP vorgegeben.

Hinweis

Ein projektierter Standard-Router wird bei Verwendung von PPPoE nicht berücksichtigt. Dieser wird der Baugruppe dynamisch vom ISP vorgegeben.

Hinweis

Keine Netzwerkkomponenten zwischen SCALANCE S und DSL-Modem

Wenn die Schnittstelle eines SCALANCE S Moduls über PPPoE betrieben wird, dürfen sich zwischen dieser Schnittstelle und dem angeschlossenen DSL-Modem keine weiteren Netzwerkkomponenten befinden, da die Einwahldaten des Internet Service Providers auf dieser Strecke ggf. unverschlüsselt übertragen werden. Bei Verwendung des Authentifizierungsprotokolls "CHAP" werden die Daten verschlüsselt übertragen.

Konfiguration von Medienmodulen

Zusätzlich zu den Funktionen des SCALANCE S623 besitzt der S627-2M zwei Medienmodulslots, in die jeweils ein elektrisches oder optisches Medienmodul mit zwei Ports eingesetzt werden kann. Dadurch wird die externe und die interne Schnittstelle um jeweils bis zu zwei Ports erweitert. Im Routing-Modus können die zusätzlichen Ports des Security-Moduls für die Anbindung der externen und internen Schnittstelle an Ringtopologien verwendet werden.

Um Medienmodule in den SCALANCE S627-2M zu integrieren, selektieren Sie das Security-Modul und wechseln Sie in die Geräteansicht. Wählen Sie dann die gewünschten Medienmodule aus dem Hardware-Katalog aus.

Für Ports mit dem Port-Typ "Kupfer" können Sie die Übertragungsgeschwindigkeit sowie das Duplex-Verfahren über den Port-Modus manuell festlegen. Für Ports mit dem Port-Typ

"Optisch" ist der Port-Modus durch das verwendete Medienmodul bzw. den verwendeten SFP fest vorgegeben und kann nicht angepasst werden.

Informationen zur Anbindung der Medienmodulports an MRP-Ringe finden Sie in folgendem Kapitel:
AUTOHOTSPOT

Betriebsart einstellen

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Betriebsart".

Betriebsart - Auswahlmöglichkeiten

Wenn sich das Security-Modul in keiner VPN-Gruppe befindet, kann die Betriebsart unter diesem Eintrag verändert werden. Wenn sich das Security-Modul in einer VPN-Gruppe befindet, kann die Betriebsart nicht verändert werden.

Die Auswahl gilt für das Schnittstellenrouting zwischen externer und interner Schnittstelle. Die DMZ-Schnittstelle (nur SCALANCE S623 und S627-2M) wird immer im Routing-Modus angebunden.

Bridge-Modus	Für den Betrieb in flachen Netzen. Externe und interne Schnittstelle befinden sich im selben IP-Subnetz. Für S623 / S627-2M: Externe und interne Schnittstelle befinden sich im selben IP-Subnetz, DMZ-Schnittstelle befindet sich in einem anderen IP-Subnetz oder ist deaktiviert.
Routing-Modus	Alle Schnittstellen befinden sich in verschiedenen IP-Subnetzen. Wenn Sie den Routing-Modus aktiviert haben, müssen Sie eine interne IP-Adresse und eine interne Subnetzmaske für die interne Schnittstelle des Security-Moduls projektieren. **Hinweis** Wenn Sie für das SCALANCE S Modul den Routing-Modus aktiviert haben, können keine MAC-Firewall-Regeln definiert werden.
Ghost-Modus (nur für SCALANCE S602 ab V3.1)	Im Betrieb übernimmt das Security-Modul für die externe Schnittstelle die IP-Adresse des Teilnehmers, der an der internen Schnittstelle des Security-Moduls angeschlossen ist. Die für die externe Schnittstelle anzugebenden IP-Adressdaten dienen lediglich dem Projektierungsladen vor dem Betrieb im Ghost-Modus.

IP-Adressparameter projektieren

Bedeutung

Netzparameter wie IP-Adresse und Subnetzmaske für die Schnittstelle(n) des Security-Moduls festlegen.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Externe Schnittstelle [P1] rot" bzw. "DMZ-Schnittstelle [P3] gelb". Die virtuelle Tunnel-Schnittstelle kann nur projektiert werden, wenn sich das Security-Modul (nur SCALANCE S612/S623/S627-2M ab V4) in mindestens einer VPN-Gruppe befindet, siehe Abschnitt "Bedeutung der Tunnel-IP-Adresse".

   Hinweis

   Projektierung der internen Schnittstelle bei Routing-Betrieb

   Wenn Sie für das Security-Modul die Betriebsart "Routing" ausgewählt haben, müssen Sie zusätzlich eine interne IP-Adresse und Subnetzmaske für die interne Schnittstelle des Security-Moduls projektieren. Sie erreichen diese Funktion in den lokalen Security-Einstellungen unter "Interne Schnittstelle [P2] grün" > "Ethernet-Adressen".

3. Aktivieren Sie ggf. die Schnittstelle über das Kontrollkästchen "Schnittstelle aktivieren".
4. Wählen Sie den Eintrag "Ethernet-Adressen".
5. Nehmen Sie die Einstellungen vor, die in der folgenden Tabelle beschrieben sind.

Parameter	Bedeutung
IP-Adresse	IP-Adresse für die externe Schnittstelle. Die IP-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind, beispielsweise 141.80.0.16
Subnetzmaske	Die Subnetzmaske besteht aus 4 Dezimalzahlen, die durch einen Punkt voneinander getrennt sind, beispielsweise 255.255.0.0
Router verwenden (nicht für Tunnel-Schnittstelle möglich)	Aktivieren Sie dieses Kontrollkästchen, wenn Sie einen Standard-Router verwenden wollen und geben Sie seine IP-Adresse in das Eingabefeld "Router-Adresse" ein.

Hinweis

Vernetzung der physischen Schnittstellen

Vernetzen Sie die physischen Schnittstellen des Security-Moduls mit entsprechenden Subnetzen, um IP-Adresskonflikte zu vermeiden.

Bedeutung der Tunnel-IP-Adresse

Wenn Sie für ein SCALANCE S612/S623/S627-2M Modul ab V4 die Funktion "NAT/NAPT im VPN-Tunnel" benutzen, müssen Sie eine Tunnel-IP-Adresse für das Security-Modul vergeben. Damit wird die Erreichbarkeit des Security-Moduls über den VPN-Tunnel sichergestellt und eine Konfigurations- und Diagnosemöglichkeit gewährleistet. Die projektierte Tunnel-IP-Adresse kann mit Hilfe entsprechender NAT-/NAPT-Regeln um Alias-Tunnel-IP-Adressen ergänzt werden. Die Subnetzmaske ist mit 32 Bit fest für die Tunnel-IP-Adresse vorgegeben und kann durch die Projektierung nicht beeinflusst werden. Die Tunnel-IP-Adresse kann nur projektiert werden, wenn sich das Security-Modul in mindestens einer VPN-Gruppe befindet.

Weitere Informationen zur Adressumsetzung mit NAT/NAPT in VPN-Tunneln finden Sie in folgendem Kapitel:
Adressumsetzung mit NAT/NAPT in VPN-Tunneln (Seite 534)

Besonderheiten beim Standard-Router

- Ist die IP-Zuweisung über "PPPoE" konfiguriert, wird ein projektierter Standard-Router ignoriert, da die Standard-Route automatisch immer über die PPPoE-Schnittstelle führt.
- Ist die IP-Zuweisung über "Statische Adresse" konfiguriert und ist das Security-Modul über einen DSL-(NAPT-) Router am Internet angeschlossen, muss der DSL-Router als Standard-Router eingetragen werden.
- Für Security-Module im Ghost-Modus (nur SCALANCE S602 ab V3.1) sind keine Standard-Router projektierbar, da diese zur Laufzeit ermittelt werden. Spezifische Routen sind für Security-Module im Ghost-Modus nicht projektierbar.

Port-Modus projektieren

Bedeutung

Über den Port-Modus wird die Übertragungsgeschwindigkeit sowie das Duplex-Verfahren festgelegt. Für an der Kommunikation beteiligte Ports sollten jeweils die gleichen Parameter eingestellt werden.

Projektierbare Port-Modi

Für SCALANCE S ab V3 sind die fest eingebauten Ports mit folgenden Port-Modi projektierbar:

Port-Modus	Bedeutung
Auto-Negotiation	Die Übertragungsgeschwindigkeit und das Duplex-Verfahren werden automatisch ausgewählt. **Hinweis** Nur wenn Autonegotiation ausgewählt ist, wird eine Übertragungsgeschwindigkeit von 1000 MBit/s sowie die Autocrossing-Funktion unterstützt.
10 Mbps, half- und full-duplex	Übertragungsgeschwindigkeit von 10 MBit/s
100 Mbps, half- und full-duplex	Übertragungsgeschwindigkeit von 100 MBit/s

Das Deaktivieren eines Ports ist nur für externe Ports sowie für den DMZ-Port bei SCALANCE S623/S627-2M möglich. Die Port-Modi für Medienmodulports werden in der Gerätesicht projektiert und richten sich nach dem Funktionsumfang des zugehörigen Medienmoduls.

Internetverbindung projektieren

Voraussetzung

Der Eintrag "PPPoE" wird in den lokalen Security-Einstellungen nur angezeigt, wenn für eine der Schnittstellen die IP-Zuweisungsmethode "PPPoE" projektiert ist.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Security-Modul
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "PPPoE".

Bedeutung

Wenn für eine der Schnittstellen des Security-Moduls eine Verbindung über PPPoE eingestellt ist, nehmen Sie in diesem Eintrag Einstellungen zum Internet Service Provider (ISP) vor.

Tabelle 1-47 Einstellungen zum ISP-Account

Funktion	Beschreibung
Authentifizierungsprotokoll	Wählen Sie keines oder eines der folgenden Authentifizierungsprotokolle: • PAP (Password Authentication Protocol) • CHAP (Challenge Handshake Authentication Protocol) **Hinweis** Beide Kommunikationspartner müssen dasselbe Authentifizierungsverfahren verwenden, ansonsten kann keine Verbindung aufgebaut werden.
Benutzername	Geben Sie den Namen zur Anmeldung beim ISP-Account ein.
Passwort	Geben Sie das Passwort zur Anmeldung beim ISP-Account ein.
Passwort wiederholen	Geben Sie das Passwort zur Anmeldung beim ISP-Account erneut ein.

Tabelle 1-48 Regeln für Benutzernamen und Passwörter

Erlaubte Zeichen	Erlaubt sind folgende Zeichen des Zeichensatzes ANSI X 3.4-1986: 0123456789 A...Z a...z !#$%&()"*'+`,-./:;<=>?@ [\]_{|}~^
Länge des Benutzernamens	1 ... 120 Zeichen
Länge des Passworts	1 ... 31 Zeichen

Tabelle 1-49 Einstellungen zur Verbindung

Funktion	Beschreibung
Dauerhafte Verbindung	Ständige Internetverbindung. Nach einer Trennung durch den Provider wird die Verbindung automatisch wieder hergestellt, auch wenn aktuell keine Pakete gesendet werden sollen.
On-Demand-Verbindung	Die Internetverbindung wird automatisch aufgebaut, wenn Pakete ins Internet gesendet werden sollen. In dieser Einstellung sind Verzögerungen beim Versenden der Pakete möglich.
Maximale Leerlaufzeit (nur bei Einstellung "On-Demand-Verbindung")	Werden innerhalb einer bestimmten Zeit keine Pakete gesendet, wird die Internetverbindung automatisch getrennt. Geben Sie im Feld "Maximale Leerlaufzeit" die Zeit in Sekunden ein, nach der die Verbindung getrennt werden soll. • Standardeinstellung: 300 • Mindestwert: 10 • Maximalwert: 3600
Zwangstrennung (nur bei Einstellung "Dauerhafte Verbindung")	Aktivieren Sie das Kontrollkästchen, um die Uhrzeit der Zwangstrennung durch das Security-Modul anpassen zu können.
Uhrzeit Zwangstrennung (nur bei Einstellung "Dauerhafte Verbindung")	Der Provider trennt nach einem bestimmten Zeitraum die Internetverbindung automatisch. Tragen Sie in diesem Feld eine Uhrzeit ein, trennt das Security-Modul zu diesem Zeitpunkt von sich aus die Internetverbindung. Damit kann eine Trennung der Internetverbindung von Seiten des Providers unter Umständen verschoben werden. Eine selbst initiierte Zwangstrennung ist nur bei einer bestehenden dauerhaften Verbindung möglich. • Standardeinstellung: 00.00 • Erlaubte Eingaben: 00:00 ... 23:59

DNS projektieren

Modulspezifische Funktion

DNS-Server für die Auflösung von FQDNs, die im Eintrag "Dynamisches DNS" angegeben sind, können von SCALANCE S Modulen ab V3 verwendet werden. DNS-Server für alle weiteren Funktionen können von SCALANCE S Modulen ab V4 verwendet werden.

DNS

Für die Adressierung von Netzwerkteilnehmern über deren FQDNs muss dem Security-Modul die IP-Adresse eines DNS-Servers bekannt sein, der DNS-Anfragen in die zugehörigen IP-Adressen der Netzwerkteilnehmer umwandelt.

DNS einrichten - Gehen Sie so vor:

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "DNS".

Option	Bedeutung
DNS-Serveradresse automatisch beziehen	Die Adresse des DNS-Servers kann über PPPoE automatisch bezogen werden, wenn das Security-Modul über ein DSL-Modem mit dem Internet verbunden ist. Kann nur für die externe Schnittstelle und die DMZ-Schnittstelle eingestellt werden. Wenn PPPoE für eine dieser Schnittstellen eingestellt wird und kein DNS-Server projektiert ist, wird die Option "DNS-Serveradresse automatisch beziehen" von STEP 7 eingestellt.
Folgende DNS-Serveradresse verwenden:	Geben Sie die Adresse des bevorzugten und des alternativen DNS-Servers manuell ein.

Dynamisches DNS projektieren

Dynamisches DNS

Mit dynamischem DNS können Sie mit einem fest definierten Namen (FQDN) auf eine sich ständig ändernde IP-Adresse zugreifen. Dies ist notwendig, wenn Sie z. B. auf einen Server zugreifen möchten, der über eine öffentliche, sich ändernde IP-Adresse erreichbar ist.

Das Security-Modul meldet einem Anbieter für dynamisches DNS (z. B. DynDNS.org, no-ip.com) die aktuelle WAN-IP-Adresse, über die das Security-Modul erreichbar ist. Der Provider sorgt dafür, dass DNS-Anfragen auf den FQDN des Security-Moduls mit der aktuellen WAN-IP-Adresse des Security-Moduls beantwortet werden.

Dynamisches DNS ist auf folgenden Schnittstellen zugelassen:

- Externe Schnittstelle
- DMZ-Schnittstelle

Voraussetzung

Bei einem Anbieter für dynamisches DNS ist ein Account angelegt und ein FQDN registriert.

Dynamisches DNS einrichten - Gehen Sie so vor:

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Dynamisches DNS".

3. Aktivieren Sie das Kontrollkästchen "Service aktivieren" im Bereich "Primärer dyn. DNS-Dienst" und nehmen Sie die folgenden Einstellungen vor:

Einstellung	Bedeutung
Anbieter	Wählen Sie aus, bei welchem Anbieter Sie einen Account für dynamisches DNS eingerichtet haben. Bei den vordefinierten Anbietern (DynDNS.org und No-IP.com) wird die Anbieter-Aktualisierungs-URL sowie die Prüf-IP-Service-URL vorausgefüllt. Bei der Auswahl der Option "Benutzerdefiniert" können Sie die beiden URLs manuell angeben.
Servicetyp	Wählen Sie aus, ob für die Anbieter-Aktualisierungs-URL der vordefinierten Anbieter das Protokoll HTTP oder HTTPS verwendet werden soll.
FQDN	Geben Sie den Hostnamen (z. B. mysecuritymodule) und den Domainnamen (z. B. dyndns.org), der beim Anbieter registriert ist, durch einen Punkt getrennt ein. Der FQDN kann als VPN-Endpunkt fungieren und von dem FQDN im Eintrag "VPN" abweichen. Welcher VPN-Endpunkt dem VPN-Partner mitgeteilt wird, können Sie in den verbindungsspezifischen VPN-Einstellungen projektieren. Die Syntax eines FQDN muss DNS-konform sein. Maximale Zeichenanzahl: 99
Fehler bei Überprüfung des Server-Zertifikats ignorieren	Zum Schutz der Authentifizierungsdaten wird das Zertifikat des Update-Servers überprüft. Schlägt die Prüfung des Zertifikates fehl, so wird die HTTPS-Verbindung standardmäßig beendet und die Account-Daten werden nicht übertragen. Wenn Sie das Kontrollkästchen aktivieren, wird die HTTPS-Verbindung bei fehlgeschlagener Prüfung (z.B. bei abgelaufenem Zertifikat) nicht beendet und die Account-Daten werden übertragen. Es wird empfohlen, die Überprüfung nicht zu ignorieren und das Kontrollkästchen nicht zu aktivieren. Das Kontrollkästchen kann nur aktiviert werden, wenn der Servicetyp "DDNS-HTTPS" ausgewählt wurde.
Benutzer	Geben Sie den Benutzernamen ein, den Sie beim Anlegen des Accounts festgelegt haben. Die Syntax des Benutzernamens muss DNS-konform sein. Maximale Zeichenanzahl: 24
Passwort	Geben Sie das Passwort ein, das Sie beim Anlegen des Accounts festgelegt haben. Die Syntax des Passworts muss DNS-konform sein. Maximale Zeichenanzahl: 24

Einstellung	Bedeutung
IP-Adresswechsel auf DSL-Router überwachen	Ist das Security-Modul über einen DSL-Router mit dem Internet verbunden, wird durch Aktivieren der Funktion der Prüf-IP-Dienst aktiviert. Das Security-Modul sendet periodisch Anfragen zur Bestimmung der aktuellen IP-Adresse des DSL-Routers sowie zur Detektion eines IP-Adresswechsels auf dem DSL-Router. Die so bestimmte IP-Adresse wird bei jeder Änderungserkennung an den Anbieter gesendet.
Periode	Geben Sie an, in welchem Zyklus der Prüf-IP-Dienst aufgerufen wird. • Standardeinstellung: 20 Minuten • Mindestwert: 10 Minuten • Maximalwert: 1440 Minuten
Anbieter-Aktualisierungs-URL	Geben Sie die URL ein, mit der das Security-Modul dem Anbieter die aktuelle öffentliche IP-Adresse meldet. Die Platzhaltertexte <FQDN> und <CurrentWanIP> müssen hierbei an den zugehörigen Stellen der URL platziert werden. Die Syntax der URL muss DNS-konform sein. Maximale Zeichenanzahl: 127
Prüf-IP-Service-URL	Geben Sie die URL ein, die das Security-Modul zur Bestimmung seiner öffentlichen IP-Adresse verwenden soll. Die Syntax der URL muss DNS-konform sein. Maximale Zeichenanzahl: 127

4. Legen Sie für den Fall, dass der primäre Anbieter ausfällt, einen weiteren Anbieter im Bereich "Sekundärer dyn. DNS-Dienst" fest (optionale Einstellung).

LLDP projektieren

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V4 verfügbar.

Voraussetzung

Das Security-Modul befindet sich im Routing-Modus.

Bedeutung

LLDP (Link Layer Discovery Protocol) ist ein Protokoll, das zur Erkennung von Netzwerktopologien verwendet wird. Ein LLDP-fähiges Gerät ist dazu in der Lage, in regelmäßigen Intervallen Informationen über sich selbst an Nachbargeräte zu senden und gleichzeitig Informationen von Nachbargeräten zu empfangen. Die empfangenen Informationen werden auf jedem LLDP-fähigen Gerät in einer LLDP-MIB-Datei gespeichert. Netzwerkmanagementsysteme können auf diese LLDP-MIB-Dateien mit Hilfe von SNMP zugreifen und damit die vorliegende Netzwerktopologie nachbilden.

Projektierbare Parameter

Das Ausmaß der Aktivität des Security-Moduls in Bezug auf LLDP kann unter dem Eintrag "LLDP-Modus" der jeweiligen Schnittstelle projektiert werden.

Parameter	Beschreibung
Name	Name des Ports, für den die Einstellung projektiert wird.
LLDP-Modus	Projektierter LLDP-Modus: • RxTx: LLDP-Telegramme senden und empfangen • Off: Keine LLDP-Telegramme senden oder empfangen

Medienredundanz in Ringtopologien

Medienredundanz mit MRP

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S627-2M verfügbar.

Bedeutung

Unter dem Begriff "Medienredundanz" werden verschiedene Verfahren zur Erhöhung der Verfügbarkeit von Industrial Ethernet-Netzen zusammengefasst, bei denen Geräte über mehrere Wege erreichbar sind. Dies kann über die Vermaschung von Netzen, Parallelschaltung von Übertragungswegen oder das Schließen einer Linientopologie zu einem Ring erfolgen.

Medienredundanzverfahren MRP

Medienredundanz innerhalb einer Ringtopologie gibt es bei SIMATIC NET-Produkten unter anderem in dem Verfahren MRP (Media Redundancy Protocol).

Bei dem Verfahren wird einer der Teilnehmer als Redundanz-Manager konfiguriert. Die anderen Teilnehmer sind Redundanz-Clients. SCALANCE S627-2M Module können ausschließlich die Rolle eines MRP-Clients einnehmen. Mit Testtelegrammen überprüft der Redundanz-Manager den Ring auf Unterbrechungsfreiheit. Die Redundanz-Clients leiten die Testtelegramme weiter. Wenn die Testtelegramme des Redundanz-Managers bei einer Unterbrechung des Rings nicht mehr am anderen Ringport des Redundanz-Managers ankommen, schaltet der Redundanz-Manager seine beiden Ringports durch und informiert die Redundanz-Clients umgehend über den Wechsel.

Die Zeitdauer, die SCALANCE X Switche als Redundanz-Manager zum Durchschalten ihrer Ringports benötigen, beträgt 200 ms.

Hinweis zum Einsatz von MRP

- MRP wird in Ringtopologien mit bis zu 100 Geräten unterstützt. Eine Überschreitung der Geräteanzahl kann zum Ausfall des Datenverkehrs führen.
- Es wird empfohlen, die beteiligten Ringports auf Full-Duplex und 100 Mbit/s einzustellen. Andernfalls kann es zum Ausfall des Datenverkehrs kommen.

Einsatzmöglichkeiten von MRP auf Medienmodulports

MRP wird ausschließlich auf den Medienmodulports des SCALANCE S627-2M unterstützt. Die folgende Tabelle zeigt die Einsatzmöglichkeiten von MRP auf den Medienmodulports eines SCALANCE S627-2M:

Ringports	Medienmodul 1		Medienmodul 2	
	P4	P5	P6	P7
MRP-Client	-	-	-	-
	Ring 1	Ring 1	-	-
	-	-	Ring 2	Ring 2
	Ring 1	Ring 1	Ring 2	Ring 2

Bei zwei unterlagerten Ringen pro SCALANCE S Modul ist Ebene 3 Kommunikation zwischen den Ringen möglich.

MRP für das Security-Modul projektieren

Voraussetzung

- Das Security-Modul befindet sich im Routing-Modus.
- Für die Schnittstellen, die an MRP-Ringe angebunden werden sollen, sind Medienmodule projektiert.
- Die Schnittstellen des Security-Moduls, die an Ringe angebunden werden sollen, sind mit den entsprechenden Ring-Managern vernetzt.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Wählen Sie in den Einstellungen der gewünschten Schnittstelle den Eintrag "Media Redundancy".

Konfigurierbare Parameter

Parameter	Bedeutung	Auswahlmöglichkeiten
MRP-Domain (nur bei Auswahl der Medienredundanzrolle "MRP-Client")	Mit Hilfe von MRP-Domains werden die Teilnehmer eines MRP-Rings festgelegt. Für die Schnittstellen aller Baugruppen, die an demselben MRP-Ring angebunden sein sollen, muss dieselbe MRP-Domain ausgewählt sein.	Anzeige der für die Schnittstelle verwendeten MRP-Domain.
Medienredundanzrolle	Auswahl des Medienredundanzprotokolls bzw. Deaktivieren von Medienredundanz für die Schnittstelle.	• Nicht Teilnehmer des Rings • MRP-Client
Ringport 1 (nur bei Auswahl der Medienredundanzrolle "MRP-Client")	Bezeichnung des ersten Ringports der ausgewählten Schnittstelle, wenn für diese die Medienredundanzrolle "MRP-Client" ausgewählt wurde.	-
Ringport 2 (nur bei Auswahl der Medienredundanzrolle "MRP-Client")	Bezeichnung des zweiten Ringports der ausgewählten Schnittstelle, wenn für diese die Medienredundanzrolle "MRP-Client" ausgewählt wurde.	-
Domain-Einstellungen	Über die Domain-Einstellungen können Sie neue MRP-Domains hinzufügen, die Namen bestehender MRP-Domains bearbeiten und bestehende MRP-Domains löschen.	-
Alternatives Medienredundanzprotokoll	Aktivieren Sie dieses Kontrollkästchen, um die Schnittstelle des Security-Moduls für andere Medienredundanzprotokolle freizuschalten.	• Schnittstelle für andere Medienredundanzprotokolle freischalten • Schnittstelle für andere Medienredundanzprotokolle deaktivieren (Standardeinstellung)
Passive Listening	Aktivieren Sie dieses Kontrollkästchen, wenn die ausgewählte Schnittstelle an Fremdnetze gekoppelt werden soll, in denen STP/RSTP (Spanning-Tree-Protocol/Rapid-Spanning-Tree-Protocol) eingesetzt wird.	• Passive Listening aktivieren (Standardeinstellung) • Passive Listening deaktivieren

Besonderheiten des Ghost-Modus

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S602 ab V3.1 verfügbar.

Bedeutung

Im Ghost-Modus hat das Security-Modul weder an der internen, noch an der externen Schnittstelle eine eigene IP-Adresse. Stattdessen bezieht das Security-Modul zur Laufzeit die IP-Adresse für ihre externe Schnittstelle von einem Teilnehmer, der an der internen Schnittstelle des Security-Moduls angeschlossen ist und dessen IP-Adressparameter zum Projektierungszeitpunkt unbekannt sein können. Eine IP-Adressänderung des internen Teilnehmers und eine damit verbundene IP-Adressänderung an der externen Schnittstelle ist möglich. Da der interne Teilnehmer anhand seiner MAC-Adresse identifiziert wird, werden IP-Adressänderungen nur für die gelernte MAC-Adresse vorgenommen. An der internen Schnittstelle des Security-Moduls wird keine IP-Adresse projektiert bzw. bezogen.

Hinsichtlich der MAC-Adressen tauscht das Security-Modul in allen an der externen Schnittstelle ausgehenden Datenpaketen (Antworten des internen Teilnehmers) die MAC-Adresse des internen Teilnehmers gegen die MAC-Adresse des Security-Moduls aus.

Ghost-Modus aktivieren

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Betriebsart".
3. Wählen Sie die Option "Ghost-Modus".

Projektierbare Moduleigenschaften

Im Ghost-Modus sind die folgenden Moduleigenschaften in den lokalen Security-Einstellungen projektierbar:

- Externe Schnittstelle [P1] rot
- Firewall
- Uhrzeitsynchronisation
- Log-Einstellungen
- SNMP

Da im Ghost-Modus keine DNS-Server projektiert werden können, ist keine FQDN-Auflösung möglich.

Voraussetzung zur Erkennung eines internen Teilnehmers

Das Security-Modul kann nur dann die IP-Adresse des internen Teilnehmers ermitteln, wenn der interne Teilnehmer von sich aus eine Datenkommunikation mit einem Kommunikationspartner des externen Netzes initiiert.
Das Security-Modul bietet zudem während der Ermittlung der IP-Adresse keine Serverdienste an. Erst nachdem vom internen Teilnehmer Datenpakete an das Security-Modul versendet wurden, kann das Security-Modul Anfragen von extern beantworten.

Portbelegung für ein- und ausgehende Datenverbindungen

Da die externe Schnittstelle des Security-Moduls und der interne Teilnehmer die gleiche IP-Adresse besitzen, muss eine gezielte Adressierung der Netzkomponenten über die TCP-/UDP-Ports erfolgen. Die Ports sind deshalb entweder dem Security-Modul oder dem internen Teilnehmer zugeordnet. In den folgenden Tabellen sind die Zuordnungen der Ports zu den jeweiligen Geräten für eingehende und ausgehende Datenverbindungen dargestellt:

Tabelle 1-50 Portbelegung für eingehende Verbindungen (von extern auf Security-Modul)

Dienst	Port	Protokoll	Kommentar
Webdienste, Projektierungs- und Diagnosezugang	443	TCP	Der HTTPS-Port ist für den Projektierungs- und Diagnosezugang über STEP 7 immer aktiviert und nicht veränderbar.
SNMP	161	TCP UDP	Nach der Aktivierung von SNMP in STEP 7 werden eingehende SNMP-Anfragen über UDP-Port 161 übertragen. Eine Übertragung über TCP-Port 161 ist ebenfalls möglich, um so beispielsweise den internen Teilnehmer erreichen zu können. **Hinweis** Nach dem Aktivieren von SNMP ist der SNMP-Port fest dem Security-Modul zugeordnet. Ist SNMP nicht aktiviert, kann mithilfe einer Firewall-Regel über SNMP auf den internen Teilnehmer zugegriffen werden.

Tabelle 1-51 Portbelegung für ausgehende Verbindungen (von Security-Modul nach extern)

Dienst	Port	Protokoll	Kommentar
Syslog	514	UDP	Wenn der Syslogdienst in STEP 7 aktiviert ist, werden Syslog-Meldungen vom Security-Modul über UDP-Port 514 übertragen. Diese Portbelegung ist nicht änderbar.
NTP	123	UDP	Wenn NTP-Server zur Uhrzeitsynchronisation genutzt werden, werden NTP-Anfragen über UDP-Port 123 übertragen. Diese Portbelegung ist nicht änderbar.

Erkennbare IP-Adressen und Subnetzmasken

Das Security-Modul erkennt ausschließlich interne Teilnehmer, die IP-Adressen im Bereich der Netzklassen A, B oder C aufweisen. Die Subnetzmaske wird entsprechend der zugehörigen Netzklasse vom Security-Modul ermittelt (siehe Tabelle "Netzklassen und zugehörige Subnetzmasken"). Damit die Subnetzmaske korrekt ermittelt werden kann, muss für den internen Teilnehmer ein Standard-Router eingetragen sein.

Teilnehmer mit IP-Adressen der Netzklassen D und E werden vom Security-Modul abgelehnt.

Netzklasse	IP-Adressen		Subnetzmaske
	Untergrenze	Obergrenze	
A	0.0.0.0	127.255.255.255	255.0.0.0
B	128.0.0.0	191.255.255.255	255.255.0.0
C	192.0.0.0	223.255.255.255	255.255.255.0
D	224.0.0.0	239.255.255.255	Wird vom Security-Modul abgelehnt
E	240.0.0.0	255.255.255.255	Wird vom Security-Modul abgelehnt

Mengengerüst

Es wird maximal ein interner Teilnehmer vom Security-Modul erkannt. Das Security-Modul verhält sich bei mehreren internen Teilnehmern wie folgt:

- Das erste durch das Security-Modul erkannte Gerät im internen Netz erhält Zugriff auf das externe Netzsegment, sofern die Firewall entsprechend konfiguriert ist.
- Der Datenverkehr von eventuell zusätzlich vorhandenen Teilnehmern im internen Netzbereich wird ausgehend auf Ebene 2 (MAC-Schicht) anhand der Absenderadresse geblockt.

Laden von Konfigurationen und Diagnose nach Inbetriebnahme

Nach dem Bezug einer IP-Adresse vom internen Teilnehmer besitzt das Security-Modul an der externen Schnittstelle eine IP-Adresse, die von derjenigen IP-Adresse abweichen kann, mit welcher das Security-Modul initial projektiert wurde. Um eine Konfiguration zu laden oder eine Diagnose durchzuführen, müssen Sie in STEP 7 für die Verbindung zur externen Schnittstelle die IP-Adresse angeben, die das Security-Modul zur Laufzeit vom internen Teilnehmer bezogen hat. Dies ist in den lokalen Security-Einstellungen oder direkt in den Dialogen "Erweitertes Laden" bzw. "Online verbinden" möglich. Weitere Informationen zum Herstellen von Online-Verbindungen finden Sie in Kapitel: Laden einer Konfiguration (Seite 555)

Routinginformationen für hierarchische Netze an der externen Schnittstelle

Wenn sich an der externen Schnittstelle des Security-Moduls hierarchische Netze mit Subnetzübergängen befinden, muss das Security-Modul die zugehörigen Routinginformationen vom internen Teilnehmer beziehen. Hierfür muss der interne Teilnehmer auf an ihn gerichtete ICMP-Anfragen antworten. Antworten auf ICMP-Broadcasts sind nicht notwendig.

Authentifizierung durch RADIUS-Server

Übersicht

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S ab V4 verfügbar.

Bedeutung

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Authentifizierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können. Durch den Einsatz von RADIUS-Servern kann der Schutz von Benutzernamen, zugeordneten Rollen und Passwörtern erhöht werden.

Einsatzszenario von RADIUS-Servern

Die Authentifizierung durch RADIUS-Server kann im Rahmen der Aktivierung benutzerspezifischer IP-Regelsätze durchgeführt werden.

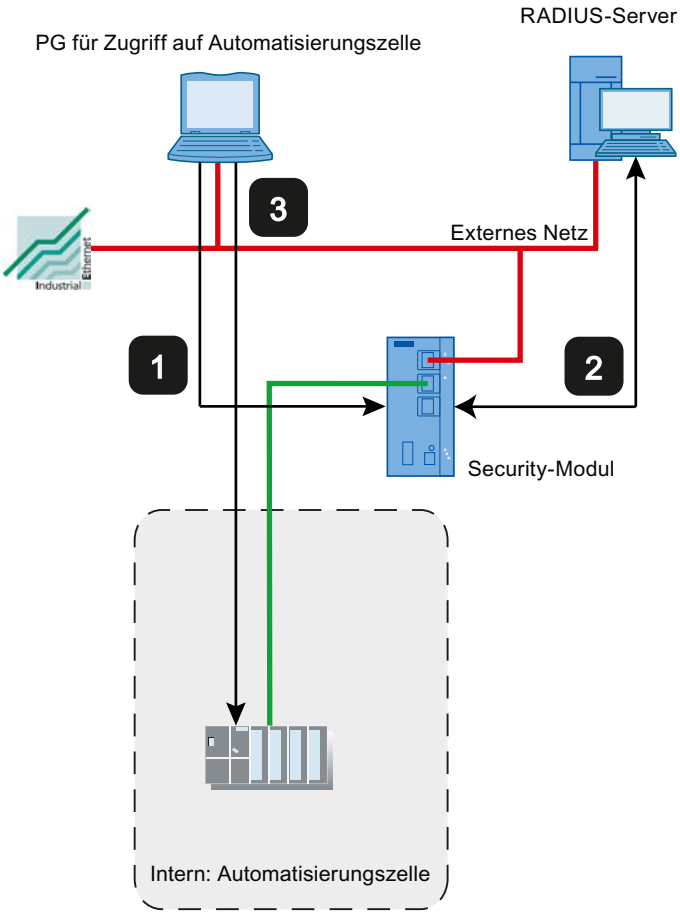

1 Eingabe der Benutzerdaten auf der Webseite des Security-Moduls

2 Authentifizierung durch RADIUS-Server und Aktivierung des benutzerspezifischen IP-Regelsatzes

3 Zugriff auf Automatisierungszelle

Der oben dargestellte Netzaufbau ist exemplarisch. Der RADIUS-Server kann sich auch im internen Netz oder im DMZ-Netz des Security-Moduls befinden.

Für die im Folgenden beschriebenen Projektierungsmöglichkeiten wird jeweils vorausgesetzt, dass ein RADIUS-Server in STEP 7 projektiert und dem jeweiligen Security-Modul zugeordnet wurde. Zudem muss ein Benutzer bzw. eine Rolle mit der Authentifizierungsmethode "RADIUS" projektiert sein. Informationen hierzu finden Sie in folgenden Kapiteln:

- RADIUS-Server definieren (Seite 522)
- RADIUS-Server einem Security-Modul zuweisen (Seite 523)

Generelle Informationen zu benutzerspezifischen IP-Regelsätzen finden Sie in folgendem Kapitel:

- AUTOHOTSPOT

Projektierungsmöglichkeiten

Für die Authentifizierung des Benutzers durch einen RADIUS-Server stehen zwei Projektierungsmöglichkeiten zur Verfügung:

- Der Benutzer ist mit seiner Rolle auf dem Security-Modul bekannt, lediglich die Passwortverwaltung für den Benutzer erfolgt über den RADIUS-Server. Auf dem RADIUS-Server ist der Benutzer mit zugehörigem Passwort konfiguriert.
  - Es wird ein Benutzer mit der Authentifizierungsmethode "RADIUS" projektiert.
  - Dem benutzerspezifischen IP-Regelsatz wird der Benutzer zugewiesen.

  Ergebnis:
  - Bei der Anmeldung eines Benutzers an der Webseite des Security-Moduls wird die Authentifizierungsanfrage an den RADIUS-Server weitergeleitet.
  - Der RADIUS-Server führt eine Passwortprüfung durch und meldet das Ergebnis an das Security-Modul zurück.
  - Ist die Passwortprüfung bestanden, wird der benutzerspezifische IP-Regelsatz aktiviert.
- Die Rolle ist auf dem Security-Modul bekannt, die Benutzerverwaltung erfolgt über den RADIUS-Server. Auf dem RADIUS-Server ist der Benutzer mit zugehörigem Passwort konfiguriert.
  - Dem benutzerspezifischen IP-Regelsatz wird eine benutzerdefinierte Rolle oder eine systemdefinierte Rolle zugewiesen.
  - Unter dem Eintrag "RADIUS" > "RADIUS-Einstellungen" in den lokalen Security-Einstellungen des Security-Moduls wird das Kontrollkästchen "RADIUS-Authentifizierung nicht projektierter Benutzer erlauben" sowie das Kontrollkästchen "Filter-ID wird für die Authentifizierung benötigt" aktiviert.

  Ergebnis:
  - Bei der Anmeldung eines Benutzers an der Webseite des Security-Moduls wird die Authentifizierungs- und Autorisierungsanfrage an den RADIUS-Server weitergeleitet.
  - Der RADIUS-Server führt eine Passwortprüfung durch und meldet das Ergebnis an das Security-Modul zurück.
  - Fall a: Wenn zusätzlich der Rollenname auf dem RADIUS-Server projektiert ist:
    Der RADIUS-Server gibt den dem Benutzer zugeordneten Rollennamen an das Security-Modul zurück.
  - Fall b: Wenn der Rollenname auf dem RADIUS-Server nicht projektiert ist:
    Das Security-Modul vergibt dem Benutzer die systemdefinierte Rolle "radius".
  - Ist die Passwortprüfung bestanden, wird der benutzerspezifische IP-Regelsatz aktiviert.

Vereinbarungen für RADIUS-Server

- Die RADIUS-Server können sich in jedem mit dem Security-Modul verbundenen Netzwerk befinden.
- Es können maximal zwei RADIUS-Server pro Security-Modul projektiert werden. Im Betrieb ist dann jeweils nur einer der RADIUS-Server aktiv.

- Bei der Definition eines RADIUS-Servers kann anstelle einer IP-Adresse auch ein FQDN verwendet werden.
- Für den Namen eines RADIUS-Servers können maximal 25 Zeichen verwendet werden.

RADIUS-Server definieren

Bedeutung

Bevor die Authentifizierung durch einen RADIUS-Server erfolgen kann, müssen Sie diesen zunächst in STEP 7 hinterlegen. Im Anschluss müssen Sie den definierten RADIUS-Server dem Security-Modul zuweisen, für welches der RADIUS-Server die Benutzerauthentifizierung übernehmen soll.

Vorgehensweise

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "RADIUS".
   Ergebnis: Unter dem ausgewählten Eintrag werden die bisher angelegten RADIUS-Server angezeigt.
2. Doppelklicken Sie auf den Eintrag "Neuen RADIUS-Server hinzufügen".
   Ergebnis: Der erstellte RADIUS-Server wird mit einer automatisch vergebenen Nummer unterhalb des Eintrags "RADIUS" angezeigt.
3. Doppelklicken Sie auf den erstellten RADIUS-Server.
   Ergebnis: Im Arbeitsbereich wird der Zuweisungsdialog für RADIUS-Server angezeigt. In der Klappliste "RADIUS-Server" ist der selektierte RADIUS-Server ausgewählt. Diesem können die gewünschten Security-Module zugewiesen werden, siehe Kapitel RADIUS-Server einem Security-Modul zuweisen (Seite 523). In den lokalen Security-Einstellungen werden die projektierbaren Eigenschaften des RADIUS-Servers angezeigt.
4. Geben Sie in den lokalen Security-Einstellungen im Eintrag "Allgemein" die folgenden Daten ein:

Parameter	Bedeutung
Name	Name für den RADIUS-Server. Der Name kann aus maximal 25 Zeichen bestehen. Dabei können die Zeichen 0-9, a-z, A-Z, -._ verwendet werden. Ein Name muss mit einem Buchstaben beginnen und mit einem Buchstaben oder einer Zahl enden. Vor dem Zeichen '.' sind nur Buchstaben und Zahlen, nach dem Zeichen '.' nur Buchstaben erlaubt. Vor und nach dem Zeichen '_' oder '-' sind nur Buchstaben und Zahlen erlaubt.
IP-Adresse / FQDN	IP-Adresse oder FQDN des RADIUS-Servers.
Port	UDP-Port, unter welchem der RADIUS-Server erreichbar ist. Standardmäßig werden Authentifizierungsdaten an Port 1812 empfangen.

Parameter	Bedeutung
Shared-Secret	Eingabe des Passworts, das bei der Übertragung der Anmeldedaten zwischen RADIUS-Server und Security-Modulen zur Verschlüsselung verwendet wird. Erlaubt sind folgende Zeichen des Zeichensatzes ANSI X 3.4-1986: 0123456789 A...Z a...z !#$%&()"*'+`,-./:;<=>?@ [\]_{|}~^ Länge des Shared-Secrets: 1 ... 32 Zeichen
Shared-Secret wiederholen	Bestätigung des Passworts.
Authentifizierungsmethode	Anzeige des Verfahrens, das zur Überprüfung der Benutzerdaten verwendet wird. Es wird ausschließlich das Verfahren "PAP" (Password Authentication Protocol) unterstützt.
Kommentar	Frei wählbare, optionale Kommentareingabe.

Ergebnis

Sie haben einen RADIUS-Server definiert und können diesen nun den gewünschten Security-Modulen zuweisen. Beachten Sie die Beschreibungen in folgendem Kapitel:
RADIUS-Server einem Security-Modul zuweisen (Seite 523)

RADIUS-Server einem Security-Modul zuweisen

Voraussetzung

Sie haben einen RADIUS-Server definiert.

Vorgehensweise

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "RADIUS".
3. Aktivieren Sie das Kontrollkästchen "RADIUS-Authentifizierung aktivieren".

   Hinweis
   Änderung der Methode zur Authentifizierung mit Webserver auf Security-Modul

   Wenn die RADIUS-Authentifizierung auf dem Security-Modul aktiviert wird, wird die Methode zur Authentifizierung mit dem Webserver von "Digest Access Authentication" auf "Basic Access Authentication" umgestellt.

4. Geben Sie in das Eingabefeld "RADIUS-Timeout" die Zeit in Sekunden ein, die das Security-Modul maximal auf eine Antwort des RADIUS-Servers warten soll.
5. Geben Sie in das Eingabefeld "RADIUS-Wiederholungen" die Anzahl der Verbindungsversuche mit dem RADIUS-Server ein.

6. Aktivieren Sie das Kontrollkästchen "RADIUS-Authentifizierung nicht projektierter Benutzer erlauben", wenn dem zu aktivierenden, benutzerspezifischen IP-Regelsatz anstelle eines Benutzers eine Rolle zugewiesen wurde.
7. Aktivieren Sie das Kontrollkästchen "Filter-ID wird für die Authentifizierung benötigt", wenn es sich bei der zugewiesenen Rolle um eine benutzerdefinierte Rolle handelt.
8. Wählen Sie unter dem Eintrag "RADIUS-Server" aus der Klappliste "Name" den RADIUS-Server, den Sie dem Security-Modul zuweisen möchten.
   Alternativ können Sie Security-Module, für die die RADIUS-Authentifizierung aktiviert ist, in den globalen Security-Funktionen RADIUS-Servern zuweisen. Generelle Informationen zur Authentifizierung durch RADIUS-Server finden Sie in folgendem Kapitel:
   AUTOHOTSPOT

Firewall einrichten

Lokale Firewall-Regeln für SCALANCE S Module

Firewall mit vordefinierten Firewall-Regeln projektieren

Firewall mit vordefinierten IP-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall".

Firewall standardmäßig aktiviert

Das Kontrollkästchen "Firewall aktivieren" ist standardmäßig aktiviert. Die Firewall ist also automatisch aktiviert und alle Zugriffe von extern auf das Security-Modul sind gesperrt. Schalten Sie durch Anklicken der entsprechenden Kontrollkästchen die Firewall-Regeln für die jeweiligen Richtungen frei.

Hinweis

Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Optionskästchen "Firewall im erweiterten Modus aktivieren". Für nähere Informationen zum erweiterten Firewall-Modus, siehe Kapitel:
Übersicht Lokale Firewall-Regeln (Seite 416)

Firewall-Projektierung mit VPN

Befindet sich das Security-Modul in einer VPN-Gruppe, ist standardmäßig das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Dies bedeutet, dass über die externe Schnittstelle oder über die DMZ-Schnittstelle nur verschlüsselter IPsec-Datentransfer zugelassen ist. Lediglich der HTTPS-Zugriff auf das Modul (TCP-Port 443) ist weiterhin ungetunnelt zugelassen.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

Tabelle 1-52 Verfügbare Firewall-Regeln und Richtungen

Dienst	Von Intern nach Extern	Von Extern nach Intern	Von Intern nach DMZ	Von DMZ nach Intern	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	x	x	x	-	Die IP-Kommunikation für die ausgewählten Kommunikationsrichtungen wird zugelassen.
Erlaube S7-Protokoll	x	x	x	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube FTP/FTPS (expliziter Modus)	x	x	x	x	TCP-Port 20 TCP-Port 21	Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.
Erlaube HTTP	x	x	x	x	TCP-Port 80	Zur Kommunikation mit einem Webserver.
Erlaube HTTPS	x	x	x	x	TCP-Port 443	Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.
Erlaube DNS	x	x	x	x	TCP-Port 53 UDP-Port 53	Kommunikationsverbindung zu einem DNS-Server wird zugelassen.
Erlaube SNMP	x	x	x	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.
Erlaube SMTP	x	x	x	x	TCP-Port 25	Zum Versenden von E-Mails über einen SMTP-Server.
Erlaube NTP	x	x	x	x	UDP-Port 123	Zur Synchronisation der Uhrzeit.
Erlaube DHCP	x	x	x	x	UDP-Port 67 UDP-Port 68	Nur bei Betriebsart Bridge-Modus: Kommunikation mit einem DHCP-Server wird zugelassen.

Tabelle 1-53 Logging

Option	Aktion bei Aktivierung
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden IP-Pakete, die verworfen werden, werden geloggt.

Die geloggten Pakete können Sie im Eintrag "Paketfilter-Log" des "Online & Diagnose"-Dialogs einsehen. Nähere Informationen hierzu finden Sie im Abschnitt Datenpakete aufzeichnen - Eintrag "Paketfilter-Log" (Seite 497).

Firewall mit vordefinierten MAC-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall".

Firewall standardmäßig aktiviert

Das Kontrollkästchen "Firewall aktivieren" ist standardmäßig aktiviert. Die Firewall ist also automatisch aktiviert und alle Zugriffe von extern auf das Security-Modul sind gesperrt. Schalten Sie durch Anklicken der entsprechenden Kontrollkästchen die Firewall-Regeln für die jeweiligen Richtungen frei.

Hinweis

Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Optionskästchen "Firewall im erweiterten Modus aktivieren". Für nähere Informationen zum erweiterten Firewall-Modus, siehe Kapitel Übersicht Lokale Firewall-Regeln (Seite 416).

Firewall-Projektierung mit VPN

Befindet sich das Security-Modul in einer VPN-Gruppe, ist standardmäßig das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

Verfügbare MAC-Regeln und Richtungen

Dienst	Von Intern nach Extern	Von Extern nach Intern	Bedeutung
Erlaube MAC-Kommunikation	x	x	Der MAC-Verkehr für die ausgewählten Kommunikationsrichtungen ist zugelassen.
Erlaube ISO-Protokoll	x	x	Der ISO-Verkehr für die ausgewählten Kommunikationsrichtungen ist zugelassen.
Erlaube SiClock	x	x	SiClock-Uhrzeittelegramme für die ausgewählten Kommunikationsrichtungen sind zugelassen.
Erlaube DCP	x	x	DCP-Verkehr zur Vergabe von IP-Adressen ist für die ausgewählten Kommunikationsrichtungen zugelassen.

Tabelle 1-54 Logging

Option	Aktion bei Aktivierung
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.

Benutzerspezifische IP-Regelsätze

Übersicht

Bedeutung

Benutzerspezifischen IP-Regelsätzen werden zunächst einzelne oder mehrere Benutzer zugewiesen. Anschließend werden die benutzerspezifischen IP-Regelsätze einzelnen oder mehreren Security-Modulen zugeordnet. Dadurch ist es möglich, benutzerspezifische Zugriffe zu erlauben. Wenn z. B. standardmäßig alle Zugriffe auf die Netze hinter einem Security-Modul gesperrt sind, können bestimmte Teilnehmer über ihre IP-Adressen für einen Benutzer freigeschaltet werden. Somit ist für diesen Benutzer der Zugriff erlaubt, für andere Benutzer bleiben die Zugriffe weiterhin gesperrt.

Anmelden des Benutzers über das Internet

Der Benutzer kann sich über die Webseite des Security-Moduls an der externen Schnittstelle oder an der DMZ-Schnittstelle anmelden. Wenn die Authentifizierung erfolgreich ist, wird der für den Benutzer definierte IP-Regelsatz für die IP-Adresse des Geräts, von dem die Anmeldung erfolgt ist, aktiviert.

Die Verbindung zur Webseite des Security-Moduls erfolgt über HTTPS unter Verwendung der IP-Adresse des angebundenen Ports bei Beachtung der gültigen Routing-Regeln:

Beispiel:

Externe Schnittstelle: 192.168.10.1

Aufruf der Login-Seite über: https://192.168.10.1/

Anmelden können sich Benutzer mit jeder Rolle, sofern der Benutzer oder die Rolle einem benutzerspezifischen IP-Regelsatz zugeordnet ist.

Möglichkeiten zur Authentifizierung des Benutzers

Je nachdem, welche Authentifizierungsmethode beim Anlegen des Benutzers gewählt wurde, der sich am Security-Modul anmeldet, wird die Authentifizierung von verschiedenen Instanzen durchgeführt:

- Authentifizierungsmethode "Passwort": Die Authentifizierung wird vom Security-Modul übernommen.
- Authentifizierungsmethode "RADIUS": Die Authentifizierung wird von einem RADIUS-Server übernommen (nur SCALANCE S ab V4)

Zuweisung von Rollen zu benutzerspezifischen IP-Regelsätzen

An SCALANCE S Module ab V4 ist auch die Zuweisung von benutzerspezifischen IP-Regelsätzen möglich, welchen Rollen zugeordnet sind. Damit ist es möglich, eine Gruppe von Benutzern für den Zugriff auf bestimmte IP-Adressen freizuschalten.

Wenn ein RADIUS-Server zur Benutzerauthentifizierung verwendet wird und dem benutzerspezifischen IP-Regelsatz eine Rolle zugewiesen wird, können auch solche Benutzer vom RADIUS-Server authentifiziert werden, die nicht auf dem Security-Modul projektiert sind. Diese Benutzer müssen auf dem RADIUS-Server oder einer separaten Datenbank hinterlegt und dort derjenigen Rolle zugeordnet sein, welche in STEP 7 dem benutzerspezifischen IP-Regelsatz zugeordnet ist. Dieses Vorgehen hat den Vorteil, dass alle Benutzerdaten ausschließlich auf dem RADIUS-Server abgelegt werden.

Weitere Informationen zur Authentifizierung durch RADIUS-Server finden Sie in folgendem Kapitel:

AUTOHOTSPOT

Benutzerspezifische IP-Regelsätze werden lokal genutzt - Vereinbarungen

Es gelten dieselben Vereinbarungen wie in folgendem Kapitel beschrieben:
Globale Firewall-Regelsätze - Vereinbarungen (Seite 396)

Benutzerspezifische IP-Regelsätze anlegen und zuweisen

Benutzerspezifische IP-Regelsätze anlegen

1. Wählen Sie in den globalen Security-Funktionen den Eintrag "Firewall" > "Benutzerspezifische IP-Regelsätze" > "IP-Regelsätze".
2. Doppelklicken Sie auf den Eintrag "Neuen IP-Regelsatz hinzufügen", um einen benutzerspezifischen IP-Regelsatz anzulegen.
   Ergebnis: Unter dem Eintrag wird der angelegte benutzerspezifische IP-Regelsatz angezeigt.
3. Doppelklicken Sie auf den angelegten benutzerspezifischen IP-Regelsatz.
   Ergebnis: Im Register "Eigenschaften" > "Allgemein" des Inspektorfensters werden die projektierbaren Eigenschaften des benutzerspezifischen IP-Regelsatzes angezeigt.
4. Klicken Sie im Inspektorfenster auf den Eintrag "Allgemein" und geben Sie die folgenden Daten ein:
   - Name: Projektweit eindeutige Bezeichnung des Regelsatzes. Der Name erscheint nach der Zuweisung des Regelsatzes in der lokalen Regelliste des Security-Moduls.
   - Beschreibung (optional): Geben Sie eine Beschreibung für den benutzerspezifischen IP-Regelsatz ein.
5. Klicken Sie auf den Eintrag "IP-Regeln" und tragen Sie der Reihe nach die Firewall-Regeln in die Liste ein.
   Im Feld "Quell-IP-Adresse" kann keine IP-Adresse eingetragen werden. Diese wird bei der Anmeldung des Teilnehmers am Security-Modul automatisch eingetragen.
   Das Kontrollkästchen "Stateful" kann für IP-Regeln benutzerspezifischer IP-Regelsätze nicht deaktiviert werden.
   Beachten Sie die Parameterbeschreibung in folgenden Kapiteln:
   IP-Paketfilter-Regeln definieren (Seite 417)
   Beachten Sie die Besonderheiten bei Firewall-Regeln, die für NAT-/NAPT-Regeln automatisch durch STEP 7 erzeugt wurden:
   Zusammenhang zwischen NAT-/NAPT-Router und benutzerspezifischer Firewall (Seite 536)

Benutzerspezifische IP-Regelsätze zuweisen

1. Klicken Sie im Inspektorfenster auf den Eintrag "Benutzer und Rollen".
2. Wählen Sie im Bereich "Verfügbare Benutzer und Rollen" die Benutzer bzw. Rollen aus, die Sie dem benutzerspezifischen IP-Regelsatz zuweisen möchten.

3. Klicken Sie auf die Schaltfläche "<<", um die ausgewählten Benutzer bzw. Rollen dem benutzerspezifischen IP-Regelsatz zuzuweisen. Die Zuweisung von Rollen an benutzerspezifische IP-Regelsätze ist nur für SCALANCE S Module ab V4 möglich.
4. Weisen Sie den angelegten benutzerspezifischen IP-Regelsatz über den Eintrag "Benutzerspezifischen IP-Regelsatz zuweisen" in den globalen Security-Funktionen den gewünschten Security-Modulen zu. Hierzu muss für die Security-Module der erweiterte Firewall-Modus aktiviert sein.

Hinweis

Zuordnung von benutzerspezifischen IP-Regelsätzen

- Einem Security-Modul kann nur ein benutzerspezifischer IP-Regelsatz pro Benutzer zugewiesen werden.
- Durch die Zuordnung wird für alle Benutzer bzw. Rollen, welche dem IP-Regelsatz zugeordnet sind, implizit das Recht zur Anmeldung am Security-Modul aktiviert.

Ergebnis

- Der benutzerspezifische IP-Regelsatz wird von den Security-Modulen als lokaler Regelsatz verwendet und erscheint automatisch in der lokalen Liste der Firewall-Regeln.
- Der Benutzer kann sich am Security-Modul anmelden. Die Authentifizierung des Benutzers wird je nach eingestellter Authentifizierungsmethode vom Security-Modul oder von einem RADIUS-Server durchgeführt.

Wertebereich für maximale Sitzungsdauer

Die Zeitdauer, nach deren Ablauf der Benutzer automatisch abgemeldet wird, kann beim Anlegen oder Bearbeiten eines Benutzers festgelegt werden und beträgt standardmäßig 30 Minuten. Die Sitzungsdauer kann auf der Webseite des Security-Moduls auf den dem Benutzer zugewiesenen Wert verlängert werden.

IP-Paketfilter-Richtungen SCALANCE S

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Security-Modul		
Von	Nach	S602	S61x	S623 / S627-2M
Intern	Extern	x	x	x
	Tunnel	-	x	x
	Any	-	x	x
	DMZ	-	-	x
	Intern	x	x	x
Extern	Intern	x	x	x
	Any	-	-	x
	Tunnel	-	-	x
	DMZ	-	-	x
Tunnel	Intern	-	x	x
	Extern	-	x	x
	DMZ	-	-	x
Any	Intern	-	x	x
	Extern	-	-	x
	DMZ	-	-	x
DMZ	Intern	-	-	x
	Extern	-	-	x
	Any	-	-	x
	Tunnel	-	-	x

x = Kommunikationsrichtung ist projektierbar

- = Kommunikationsrichtung ist nicht projektierbar

MAC-Paketfilter-Richtungen SCALANCE S

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Security-Modul		
Von	Nach	S602	S61x	S623 / S627-2M
Intern	Extern	x	x	x
	Tunnel	-	x	x
	Any	-	x	x
Extern	Intern	x	x	x
	Any	-	-	x
	Tunnel	-	-	x
Tunnel	Intern	-	x	x
	Extern	-	x	x
Any	Intern	-	x	x
	Extern	-	-	x

x = Kommunikationsrichtung ist projektierbar

- = Kommunikationsrichtung ist nicht projektierbar

Standardregeln für IP-Dienste anpassen

Voraussetzung

Diese Funktion ist nur im erweiterten Firewall-Modus verfügbar.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Standardregeln für IP-Dienste".

Bedeutung der erweiterten Einstellungen

Parameter	Bedeutung bei Aktivierung
Erweiterte Zustandsoptionen verwenden	Die zulässige Anzahl von Verbindungen und Firewall-Zuständen pro Zeit wird begrenzt. Überschreitet ein Netzwerkteilnehmer diese Begrenzung, wird dessen IP-Adresse in die IP-Blacklist des Security-Moduls aufgenommen. Der Teilnehmer kann dann nicht mehr über das Security-Modul kommunizieren. Die IP-Blacklist des Security-Moduls kann im Online-Modus eingesehen werden.
Alle aktivierten Regeln loggen	Pakete, die gemäß den Standardregeln für IP-Dienste zugelassen werden, werden geloggt.
ICMP-Test für Schnittstellen freischalten	Ping-Anfragen, die an einer Schnittstelle des Security-Moduls eingehen, können an andere Schnittstellen weitergeleitet werden. Aus dem externen Netz können somit beispielsweise Ping-Anfragen auf die interne Schnittstelle des Security-Moduls durchgeführt werden.

Standardmäßige Firewall-Regeln für SCALANCE S

In der folgenden Tabelle sind die standardmäßigen Firewall-Regeln für SCALANCE S Module aufgeführt. Die Firewall-Regeln sind zum Teil nur dann aktiv, wenn der betreffende Dienst vom Security-Modul verwendet wird (z.B. SNMP).

Dienst	Richtung	Schnittstelle X1 (rot)	Schnittstelle X2 (grün)	Schnittstelle X3 (gelb) (nur für S623, S627-2M)	Tunnelschnittstelle* (nicht für S602)
Schnittstellenrerouting	ausgehend	-	x	-	-
HTTPS		x	x*	x	x
ICMP	eingehend	-	x	-	x
ICMP Pathfinder (nur für SCALANCE S602 ab V3.1 im Ghost-Modus)	ausgehend	-	x	-	-
SNMP	eingehend	x	x	x	x
Syslog	ausgehend	x	x	x	x
NTP	ausgehend	x	x	x	x
DNS	ausgehend	x	x	x	x
HTTP	ausgehend	x	-	x	-
VPN (IKE)		x	-	x	-
VPN (NAT Traversal)		x	-	x	-
BootP Server	eingehend	-	x	x	-
BootP Client	ausgehend	-	x	x	-
RADIUS	ausgehend	x	x	x	x
CARP (nur für SCALANCE S62x ab V4)	ausgehend	x*	x*	-	-
Pfsync (nur für SCALANCE S62x ab V4)	ausgehend	-	-	x*	-

x standardmäßig aktiviert

- standardmäßig deaktiviert

* nicht anpassbar

SCALANCE S Modul als Router

Routen festlegen

Bedeutung

Angabe von Routen für die Adressierung von Subnetzen, die nicht direkt über das Security-Modul erreicht werden können.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Routing".
3. Doppelklicken Sie auf den Eintrag "Neu hinzufügen" in der Tabelle, um eine Route hinzuzufügen.
4. Tragen Sie die folgenden Werte ein:

Parameter	Funktion	Beispiel-Wert
Netz-ID	Anfragen an Teilnehmer des Subnetzes mit der hier angegebenen Netz-ID und der angegebenen Subnetzmaske werden über die angegebene Router-IP-Adresse in das Subnetz geleitet. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse im oder außerhalb des Subnetzes liegt. Die angegebene Netz-ID darf nicht im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.	192.168.11.0
Subnetzmaske	Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.	255.255.255.0
Router-IP-Adresse	IP-Adresse des Routers, über den das Subnetz erreicht wird. Für SCALANCE S Module ab V4 kann alternativ ein FQDN eingegeben werden. Die angegebene Router-IP-Adresse muss im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.	192.168.10.2
Rerouting aktivieren (nur für SCALANCE S ab V3)	Aktivieren Sie dieses Kontrollkästchen, wenn die Telegramme der eingegebenen Route an derselben Schnittstelle des Security-Moduls ein- und ausgehen sollen (Rerouting). Rerouting wird nur an der internen Schnittstelle des Security-Moduls unterstützt.	

Adressumsetzung mit NAT/NAPT in VPN-Tunneln

Modulspezifische Funktion

Die Adressumsetzung mit NAT/NAPT in VPN-Tunneln ist nur für SCALANCE S612/S623/S627-2M Module ab V4 verfügbar.

Bedeutung

Adressumsetzungen mit NAT/NAPT können auch für Kommunikationsbeziehungen durchgeführt werden, die über VPN-Tunnel aufgebaut sind.

Voraussetzungen

Für ein SCALANCE S Modul, das eine Adressumsetzung mit NAT/NAPT in einem VPN-Tunnel durchführen soll, gelten allgemein die folgenden Voraussetzungen:

- Das SCALANCE S Modul befindet sich in einer VPN-Gruppe.
- Das SCALANCE S Modul befindet sich im Routing-Modus und/oder die DMZ-Schnittstelle des SCALANCE S Moduls ist aktiviert.
- Die Tunnel-Schnittstelle ist aktiviert.
- Der erweiterte Firewall-Modus ist aktiviert.
- Die Firewall im Editor für IP-Regeln ist aktiviert.

Unterstützte Adressumsetzungsrichtungen

Es werden die Adressumsetzungsrichtungen unterstützt, die in folgendem Kapitel beschrieben sind:
NAT-/NAPT-Routing (Seite 436)

Unterstützte Adressumsetzungsaktionen

Bei getunnelten Kommunikationsbeziehungen werden die folgenden Adressumsetzungsaktionen unterstützt:

- Destination-NAT ("Redirect")
- Source-NAT ("Masquerading")
- Source- und Destination-NAT ("1:1-NAT")
- NAPT ("Portforwarding")

Grundlegende Informationen zu diesen Adressumsetzungsaktionen finden Sie in folgendem Kapitel:
NAT-/NAPT-Routing (Seite 436)

Unterstützte VPN-Kopplungen

Im Zusammenspiel mit NAT/NAPT werden die folgenden VPN-Kopplungen unterstützt:

VPN-Kopplung		VPN-Verbindung wird initiiert von	Adressumsetzung wird durchgeführt von
SCALANCE S (a)	SCALANCE S (b)	SCALANCE S (a) oder SCALANCE S (b)	SCALANCE S (a) und/oder SCALANCE S (b)
SCALANCE S	CP 343-1 Adv. / 443-1 Adv. / PC-CP	SCALANCE S oder CP 343-1 Adv. / 443-1 Adv. / PC-CP	SCALANCE S
SCALANCE S	SCALANCE M	SCALANCE M	SCALANCE S und/oder SCALANCE M*
SOFTNET Security Client	SCALANCE S	SOFTNET Security Client	SCALANCE S

* Es wird ausschließlich 1:1-NAT unterstützt.

SCALANCE S Module vom Typ SCALANCE S623 V4 und SCALANCE S627-2M V4, die einen VPN-Endpunkt an der externen Schnittstelle und an der DMZ-Schnittstelle besitzen, können an beiden Schnittstellen gleichzeitig Adressumsetzungen durchführen.

Adressumsetzungsverhalten bei Teilnahme an mehreren VPN-Gruppen

Ist ein SCALANCE S Modul Teilnehmer mehrerer VPN-Gruppen, dann gelten die Adressumsetzungsregeln, die für die Tunnel-Schnittstelle des SCALANCE S Moduls projektiert wurden, für alle VPN-Verbindungen dieses SCALANCE S Moduls.

Beachten Sie: Sobald Sie eine NAT-Adressumsetzung in oder aus Richtung Tunnel konfiguriert haben, sind lediglich noch die beteiligten IP-Adressen der NAT-Adressumsetzungsregeln über den VPN-Tunnel erreichbar.

Zusammenhang zwischen NAT-/NAPT-Router und benutzerspezifischer Firewall

Modulspezifische Funktion

Die Projektierung von NAT-/NAPT-Regeln in der benutzerspezifischen Firewall ist nur für SCALANCE S Module ab V3 verfügbar.

Bedeutung

Von STEP 7 wird nach dem Anlegen von NAT-/NAPT-Regeln in der benutzerspezifischen Firewall automatisch ein benutzerspezifischer IP-Regelsatz erzeugt, der die Kommunikation in der projektierten Adressumsetzungsrichtung freigibt. Diesen benutzerspezifischen IP-Regelsatz können Sie dann einzelnen oder mehreren Benutzern und/oder einzelnen oder mehreren Rollen (nur für SCALANCE S Module ab V4) zuweisen.

Die erzeugten Firewall-Regeln können ggf. verschoben und erweitert werden (zusätzliche IP-Adresse, Dienste, Übertragungsgeschwindigkeit). Von STEP 7 erzeugte Firewall-Parameter können nicht angepasst werden. Wird der benutzerspezifische IP-Regelsatz einem Security-Modul mit deaktiviertem NAT/NAPT zugewiesen, dann kommen die NAT-/NAPT-Regeln aus der benutzerspezifischen Firewall ebenso auf diesem Security-Modul nicht zur Anwendung.

Hinweis

Die Adressumsetzungsaktionen "Source-NAT + Destination-NAT" und "Double-NAT" werden in Zusammenhang mit der benutzerspezifischen Firewall nicht unterstützt.

So erreichen Sie diese Funktion

Eintrag "NAT" bzw. "NAPT" im Editor für benutzerspezifische IP-Regelsätze, siehe folgendes Kapitel:
Benutzerspezifische IP-Regelsätze anlegen und zuweisen (Seite 529)

Unterstützte Adressumsetzungsrichtungen für die Aktion "Source-NAT"

Die Aktion "Source-NAT" kann in folgenden Richtungen durchgeführt werden:

- Extern nach DMZ
- DMZ nach Extern

Im Feld "Quell-IP-Adresse" kann keine IP-Adresse eingetragen werden. Diese wird bei der Anmeldung des Teilnehmers am Security-Modul automatisch eingetragen.

Unterstützte Adressumsetzungsrichtungen für die Aktion "Destination-NAT"

Die Aktion "Destination-NAT" kann in folgenden Richtungen durchgeführt werden:

- Extern nach Intern
- Extern nach DMZ
- DMZ nach Intern
- DMZ nach Extern
- Tunnel nach Intern (nur SCALANCE S612/S623/S627-2M ab V4)
- Tunnel nach Extern (nur SCALANCE S612/S623/S627-2M ab V4)
- Tunnel nach DMZ (nur SCALANCE S612/S623/S627-2M ab V4)

Unterstützte Adressumsetzungsrichtungen für NAPT

Die Adressumsetzung mit NAPT kann in folgenden Richtungen durchgeführt werden:

- Extern nach Intern
- Extern nach DMZ
- DMZ nach Intern
- DMZ nach Extern
- Tunnel nach Intern (nur SCALANCE S612/S623/S627-2M ab V4)
- Tunnel nach Extern (nur SCALANCE S612/S623/S627-2M ab V4)
- Tunnel nach DMZ (nur SCALANCE S612/S623/S627-2M ab V4)

NAT-/NAPT-Adressumsetzung und zugehörige benutzerspezifische IP-Regelsätze

In den Firewall-Regeln für benutzerspezifische IP-Regelsätze, die auf Basis von NAT-/NAPT-Regeln erzeugt werden, kann im Feld "Quell-IP-Adresse" keine IP-Adresse eingetragen werden. Diese wird bei der Anmeldung des Teilnehmers am Security-Modul automatisch eingetragen. Die restlichen Eigenschaften sind zu den Firewall-Regeln, die lokal für einzelne Security-Module erzeugt werden, identisch. Siehe Kapitel:
Zusammenhang zwischen NAT-/NAPT-Router und Firewall (Seite 449)

Security-Modul als DHCP-Server

Übersicht DHCP-Server

Übersicht

Sie können das SCALANCE S Modul am internen Netz und am DMZ-Netz als DHCP-Server (DHCP = Dynamic Host Configuration Protocol) betreiben. Damit ist es möglich, den am internen Netz bzw. den am DMZ-Netz angeschlossenen Geräten automatisch IP-Adressen zuzuweisen.

Der gleichzeitige DHCP-Server-Betrieb an beiden Schnittstellen ist möglich (nur SCALANCE S623/S627-2M).

Die IP-Adressen werden hierbei entweder dynamisch aus einem von Ihnen vergebenen Adressband verteilt oder es wird gemäß Ihrer Vorgabe eine bestimmte IP-Adresse einem bestimmten Gerät zugewiesen. Sollen Geräte an der internen Schnittstelle bzw. an der DMZ-Schnittstelle für die Firewall-Konfiguration immer die gleiche IP-Adresse bekommen, darf die Adresszuordnung nur statisch anhand der MAC-Adresse oder anhand der Client-ID erfolgen.

Voraussetzungen

Sie müssen die Geräte am internen Netz bzw. am DMZ-Netz so konfigurieren, dass diese die IP-Adresse von einem DHCP-Server beziehen.

Je nach Betriebsart übermittelt das Security-Modul den Teilnehmern im jeweiligen Subnetz eine IP-Adresse des Standard-Routers oder Sie müssen den Teilnehmern im Subnetz eine Router-IP-Adresse bekannt machen.

- Router-IP-Adresse wird übermittelt
  In folgenden Fällen wird durch das DHCP-Protokoll vom Security-Modul eine Router-IP-Adresse an den Teilnehmer übermittelt:
  - Der Teilnehmer befindet sich an der DMZ-Schnittstelle (nur SCALANCE S623/S627-2M)
    Das Security-Modul übermittelt in diesem Fall die eigene IP-Adresse als Router-IP-Adresse.
  - Der Teilnehmer befindet sich an der internen Schnittstelle und das Security-Modul ist für den Router-Betrieb konfiguriert
    Das Security-Modul übermittelt in diesem Fall die eigene IP-Adresse als Router-IP-Adresse.
  - Der Teilnehmer befindet sich an der internen Schnittstelle und das Security-Modul ist nicht für den Router-Betrieb konfiguriert, es ist aber in der Konfiguration des Security-Moduls ein Standard-Router angegeben.
    Das Security-Modul übermittelt in diesem Fall die IP-Adresse des Standard-Routers als Router-IP-Adresse.
- Router-IP-Adresse wird nicht übermittelt
  Tragen Sie in folgenden Fällen die Router-IP-Adresse beim Teilnehmer manuell ein:
  - Der Teilnehmer befindet sich an der internen Schnittstelle und das Security-Modul ist nicht für den Router-Betrieb konfiguriert. Zusätzlich ist in der Konfiguration des Security-Moduls kein Standard-Router angegeben.

Siehe auch

DHCP-Server konfigurieren (Seite 539)

DHCP-Server konfigurieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "DHCP-Server".
3. Wählen Sie aus, für welche Schnittstelle die DHCP-Einstellungen vorgenommen werden sollen.
4. Nehmen Sie die Adresszuordnung vor. Sie haben die beiden folgenden Möglichkeiten zur Konfiguration:
   - Statische Adresszuordnung
     Geräten mit einer bestimmten MAC-Adresse oder Client-ID werden jeweils vorgegebene IP-Adressen zugeordnet. Tragen Sie hierzu diese Geräte in die Adressliste im Eingabebereich "Statische Adresszuordnung" ein.
   - Dynamische Adresszuordnung
     Geräte, deren MAC-Adresse oder deren Client-ID nicht explizit angegeben wurde, erhalten eine beliebige IP-Adresse aus dem vorgegebenen Adressband. Aktivieren Sie hierzu das Kontrollkästchen "Dynamischen IP-Adressbereich" aktivieren". Stellen Sie im Anschluss das Adressband im Eingabebereich "Dynamischen IP-Adressbereich aktivieren" ein.

Hinweis

Dynamische Adressvergabe - Verhalten nach Unterbrechung der Spannungsversorgung

Beachten Sie, dass die dynamisch vergebenen IP-Adressen nicht gespeichert werden, wenn die Spannungsversorgung unterbrochen wird. Nach Wiederkehr der Spannungsversorgung müssen Sie daher dafür sorgen, dass die Teilnehmer erneut eine IP-Adresse anfordern.

Sie sollten daher die dynamische Adressvergabe nur für folgende Teilnehmer vorsehen:

- Teilnehmer, die im Subnetz temporär genutzt werden (wie beispielsweise Service-Geräte).
- Teilnehmer, die eine einmal zugewiesene IP-Adresse bei einer erneuten Anforderung an den DHCP-Server als "Vorzugsadresse" übermitteln (wie beispielsweise PC-Stationen).

Für dauerhafte Teilnehmer ist die statische Adresszuweisung über die Angabe einer Client-ID oder der MAC-Adresse vorzuziehen.

Konsistenzprüfung - diese Regeln müssen Sie beachten

Berücksichtigen Sie bei Ihrer Eingabe die nachfolgend aufgeführten Regeln.

- Die in der Adressliste im Eingabebereich "Statische Adresszuordnung" zugewiesenen IP-Adressen dürfen nicht im Bereich der dynamischen IP-Adressen liegen.
- IP-Adressen, MAC-Adressen und Client-IDs dürfen in der Tabelle "Statische Adresszuordnung" nur einmal vorkommen (bezogen auf das Security-Modul).
- Sie müssen bei den statisch zugewiesenen IP-Adressen entweder die MAC-Adresse oder die Client-ID (Rechnername) angeben.
- Die Client-ID ist eine Zeichenfolge mit maximal 63 Zeichen. Es dürfen nur die folgenden Zeichen verwendet werden: a-z, A-Z, 0-9 und - (Bindestrich).
  Hinweis
  Bei SIMATIC S7 kann den Geräten an der Ethernet-Schnittstelle für den Bezug einer IP-Adresse über DHCP eine Client-ID zugewiesen werden.
  Bei PCs ist die Vorgehensweise abhängig vom verwendeten Betriebssystem; empfohlen wird, hier die MAC-Adresse für die Zuordnung zu verwenden.
- Sie müssen bei den statisch zugewiesenen IP-Adressen die IP-Adresse angeben.
- Folgende IP-Adressen dürfen nicht im Bereich des freien IP-Adressbereichs (dynamische IP-Adressen) liegen:
  - alle Router-IP-Adressen im Eintrag "Routing"
  - Syslog-Server
  - Standard-Router
  - Security-Modul-Adresse(n)
- DHCP wird vom Security-Modul an der Schnittstelle zum internen Subnetz und an der Schnittstelle zum DMZ-Netz unterstützt. Aus diesem Betriebsverhalten des Security-Moduls ergeben sich weiterhin für IP-Adressen im Bereich der dynamischen Adresszuordnungen folgende Anforderungen:
  - Bridge-Modus
    Der Bereich muss in dem durch das Security-Modul definierten Netz liegen.
  - Routing-Modus
    Der Bereich muss in dem durch das Security-Modul definierten internen Subnetz liegen.
    Hinweis
    Das DMZ-Netz stellt stets ein separates Subnetz dar. Bei Verwendung von DHCP auf der DMZ-Schnittstelle muss beachtet werden, dass sich der freie IP-Adressbereich (dynamische IP-Adressen) innerhalb des DMZ-Subnetzes befindet.
- Der freie IP-Adressbereich muss durch die Angabe der Startadresse und der Endadresse vollständig angegeben sein. Die Endadresse muss größer als die Startadresse sein.
- Die IP-Adressen, die Sie in die Adressliste im Eingabebereich "Statische Adresszuordnung" eingeben, müssen im Adressbereich des internen Subnetzes bzw. des DMZ-Subnetzes des Security-Moduls liegen.

Siehe auch

Konsistenzprüfung durchführen (Seite 373)

Proxy-ARP konfigurieren

Übersicht

Proxy-ARP ermöglicht Routern, ARP-Anfragen für Hosts zu beantworten. Die Hosts befinden sich dabei in durch Router getrennten Netzen, verwenden jedoch den gleichen IP-Adressenbereich.

Sendet PC1 eine ARP-Anforderung an PC2, bekommt er von dem dazwischen liegenden Security-Modul und nicht von PC2 eine ARP-Antwort und die Hardwareadresse der Schnittstelle (MAC-Adresse des Ports des Security-Moduls), auf der die Anfrage empfangen wurde. Der anfragende PC1 sendet dann seine Daten an das Security-Modul, das sie dann an PC2 weiterleitet.

So erreichen Sie diese Funktion

Diese Funktion ist nur für die interne Schnittstelle eines Security-Moduls verfügbar, das Teilnehmer einer VPN-Gruppe ist und sich im Bridge-Modus befindet.

1. Markieren Sie das zu bearbeitende Security-Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Proxy-ARP".
3. Wenn das Security-Modul eine ARP-Anfrage aus dem eigenen LAN stellvertretend für den spezifischen Verbindungspartner beantworten soll, tragen Sie die entsprechende IP-Adresse ein.

IPSec-Tunnel: Gruppen anlegen und zuordnen

Interne Netzknoten konfigurieren

Lernmodus zum Lernen interner Knoten verwenden

Teilnehmer für die Tunnelkommunikation automatisch finden

Ein großer Vorteil für die Konfiguration und den Betrieb der Tunnelkommunikation ist, dass SCALANCE S Module im Bridge-Modus die Teilnehmer an der internen Schnittstelle selbsttätig auffinden können. Neue Teilnehmer werden vom Security-Modul im laufenden Betrieb erkannt. Die erkannten Teilnehmer werden an die Security-Module gemeldet, die zur selben VPN-Gruppe gehören. Dadurch ist der Datenaustausch innerhalb der Tunnel einer Gruppe jederzeit in beide Richtungen gewährleistet.

Erkennbare Teilnehmer

Erkannt werden folgende Teilnehmer:

- IP-fähige Netzknoten
  IP-fähige Netzknoten werden gefunden, wenn sie eine ICMP-Antwort auf den ICMP-Subnetz-Broadcast senden.
  IP-Knoten hinter Routern sind auffindbar, wenn die Router ICMP-Broadcasts weiterleiten.
- ISO-Netzknoten
  Netzknoten, die nicht IP-fähig, jedoch über ISO-Protokolle ansprechbar sind, können ebenfalls gelernt werden.
  Voraussetzung ist, dass sie auf XID- bzw. TEST-Telegramme antworten. TEST und XID (Exchange Identification) sind Hilfsprotokolle zum Informationsaustausch auf der Layer-2-Ebene. Durch das Versenden dieser Telegramme mit einer Broadcast-Adresse können diese Netzknoten ausfindig gemacht werden.
- PROFINET-Knoten
  Mit Hilfe von DCP (Discovery and basic Configuration Protocol) werden PROFINET-Knoten gefunden.

Netzknoten, die diese Bedingungen nicht erfüllen, müssen Sie manuell konfigurieren.

Subnetze, die sich hinter internen Routern befinden, müssen Sie ebenfalls manuell konfigurieren.

So erreichen Sie die Funktion

1. Markieren Sie das Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN" > "Knoten".

Lernmodus ein-/ausschalten

Die Lernfunktion ist in der Konfiguration standardmäßig für jedes Security-Modul eingeschaltet.

Das Lernen kann für SCALANCE S auch völlig abgeschaltet werden. Dann müssen Sie alle internen Netzknoten, die an der Tunnelkommunikation teilnehmen sollen, manuell konfigurieren.

Wann ist es sinnvoll, den automatischen Lernmodus auszuschalten?

Die Standardeinstellungen für das Security-Modul gehen davon aus, dass interne Netze stets sicher sind; das heißt auch, dass im Normalfall keine Netzknoten in das interne Netz zugeschaltet werden, die nicht vertrauenswürdig sind.

Das Ausschalten des Lernmodus kann sinnvoll sein, wenn das interne Netz statisch ist, d. h., wenn sich die Anzahl der internen Knoten und deren Adressen nicht ändern.

Mit Ausschalten des Lernmodus entfällt im internen Netz die Belastung des Mediums und der Knoten durch die Lerntelegramme. Auch das Security-Modul wird etwas leistungsfähiger, da es nicht durch die Bearbeitung der Lerntelegramme belastet wird.

Anmerkung: Im Lernmodus werden alle Knoten im internen Netz erfasst. Die Angaben zum VPN-Mengengerüst beziehen sich nur auf die Knoten, die im internen Netz über VPN kommunizieren.

Hinweis

Werden im internen Netz mehr als 128 interne Knoten betrieben, wird damit das zulässige Mengengerüst überschritten und ein nicht erlaubter Betriebszustand erzeugt. Aufgrund der Dynamik im Netzwerkverkehr kommt es dann dazu, dass interne Knoten, die bereits gelernt wurden, wieder durch neue, bis jetzt noch nicht bekannte interne Knoten ersetzt werden.

Siehe auch

Interne Subnetze manuell konfigurieren (Seite 544)

IP-Netzknoten manuell konfigurieren

Bedeutung

Alternativ zum Lernmodus, den Sie über das Kontrollkästchen "Lernen interner Knoten zulassen" aktivieren und über den die internen Netzknoten dynamisch vom Security-Modul gelernt werden, können Sie die zu lernenden Netzknoten im Eintrag "Interne IP-Knoten" manuell eintragen und damit für die VPN-Tunnelkommunikation freigeben. Die MAC-Adresse eines Netzknotens ist hierbei eine optionale Angabe.

Voraussetzung

- Das Security-Modul befindet sich im Bridge-Modus.
- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN" > "Knoten" > "Interne IP-Knoten".

MAC-Netzknoten manuell konfigurieren

Bedeutung

Alternativ zum Lernmodus, den Sie über das Kontrollkästchen "Lernen interner Knoten zulassen" aktivieren und über den die internen Netzknoten dynamisch vom Security-Modul gelernt werden, können Sie die zu lernenden Netzknoten im Eintrag "Interne MAC-Knoten" manuell eintragen und damit für die VPN-Tunnelkommunikation freigeben.

Voraussetzung

- Das Security-Modul befindet sich im Bridge-Modus.
- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN" > "Knoten" > "Interne MAC-Knoten".

Interne Subnetze manuell konfigurieren

Voraussetzung

- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN" > "Knoten" > "Interne Subnetze".

Security-Modul im Bridge-Modus - Eintrag "Interne Subnetze"

Um interne Subnetze manuell für die VPN-Tunnelkommunikation freizugeben, müssen Sie folgende Adressparameter eingeben:

Parameter	Funktion	Beispiel-Wert
Netz-ID	Netz-ID des Subnetzes, das für die VPN-Tunnelkommunikation freigegeben werden soll. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt. Darf nicht im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.	192.168.11.0
Subnetzmaske	Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.	255.255.255.0
Router-IP-Adresse	IP-Adresse des Routers, über den das freizugebende Subnetz erreicht wird. Muss im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.	192.168.10.2

Security-Modul im Routing-Modus - Eintrag "Durch Tunnel erreichbare Subnetze"

Im Routing-Modus werden stets komplette Subnetze getunnelt. Um durch Router erreichbare interne Subnetze, das externe Subnetz bzw. das DMZ-Subnetz manuell für die VPN-Tunnelkommunikation freizugeben, müssen Sie folgende Adressparameter eingeben:

Parameter	Funktion	Beispiel-Wert
Netz-ID	Netz-ID des Subnetzes, das für die VPN-Tunnelkommunikation freigegeben werden soll. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.	192.168.11.0
Subnetzmaske	Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.	255.255.255.0
Kommentar	Zusätzliche, optionale Kommentareingabe.	

Router- und Firewallredundanz

Übersicht

Bedeutung

Durch Router- und Firewall-Redundanz können Ausfälle der Security-Module SCALANCE S623 ab V4 und SCALANCE S627-2M ab V4 während des Betriebs automatisiert kompensiert werden. Hierzu fassen Sie zwei Security-Module des Typs SCALANCE S623 oder SCALANCE S627-2M in einer Redundanzbeziehung zusammen, indem Sie für beide Security-Module Router- und Firewallredundanz aktivieren. Im Anschluss bestimmen Sie, welches das im Normalbetrieb passive Security-Modul der Redundanzbeziehung sein soll (Sekundärmodul). Diese Einstellung nehmen Sie am im Normalbetrieb aktiven Security-Modul der Redundanzbeziehung vor (Primärmodul). Fällt das Primärmodul im Betrieb aus, übernimmt das Sekundärmodul automatisch die Funktion als Firewall und (NAT-/NAPT-) Router. Um eine identische Konfiguration beider Security-Module zu gewährleisten, werden diese über deren DMZ-Schnittstellen miteinander verbunden und während des Betriebs in ihrer Konfiguration synchronisiert. Die DMZ-Schnittstellen der beteiligten Security-Module sind in diesem Fall nicht für andere Zwecke nutzbar.

Adressredundanz

Zusätzlich zu ihren jeweiligen Modul-IP-Adressen teilen sich die beiden Security-Module an der externen und an der internen Schnittstelle jeweils eine gemeinsame IP-Adresse, damit bei Ausfall eines Security-Moduls keine Änderungen an den IP-Adressen notwendig sind. Deshalb müssen Sie für die externe und für die interne Schnittstelle der Redundanzbeziehung eine IP-Adresse projektieren.

Auswirkungen von Redundanzbeziehungen auf Security-Module

Beim Erstellen von Redundanzbeziehungen zwischen Security-Modulen werden einige Eigenschaften dieser Security-Module automatisch angepasst, um die Kompatibilität mit der Redundanzbeziehung herzustellen. Folgende Eigenschaften sind von dieser Anpassung betroffen:

Moduleigenschaft	Auswirkung auf die Moduleigenschaft
Betriebsart	Die Betriebsart wird ggf. auf die Option "Routing-Modus" eingestellt.
VPN-Gruppenzugehörigkeiten	Das Security-Modul wird ggf. aus zugehörigen VPN-Gruppen entfernt.
Schnittstellenkonfiguration	Die externe Schnittstelle und die DMZ-Schnittstelle des Security-Moduls werden ggf. aktiviert. Die IP-Zuweisungsmethode "Statische Adresse" wird ggf. für alle Schnittstellen projektiert.

Projektierung von Security-Modulen in Redundanzbeziehungen

Nachdem Router- und Firewallredundanz für die Security-Module aktiviert und das Primärmodul der Redundanzbeziehung bestimmt wurde, wird ein Teil der Moduleigenschaften ausschließlich über das Primärmodul projektiert. Die für das Primärmodul projektierten Eigenschaften gelten dann für die Redundanzbeziehung und sind nicht für das Sekundärmodul projektierbar. Folgende Eigenschaften können für die Redundanzbeziehung projektiert werden:

- Grundeinstellungen der Redundanzbeziehung (Sekundärmodul, Netzparameter)
- Firewall (Standardregeln für IP-Dienste werden für die einzelnen Security-Module separat projektiert)
- Routing
- NAT-/NAPT-Routing (kein 1:1-NAT)

Die Werte der oben aufgeführten Eigenschaften werden initial vom Primärmodul für die Redundanzbeziehung übernommen.

Die im Folgenden aufgeführten Einstellungen werden auch nach dem Einbinden in eine Redundanzbeziehung für die einzelnen Security-Module separat projektiert. Die Projektierung dieser Eigenschaften für das Primärmodul hat deshalb keine Auswirkungen auf das Sekundärmodul.

- Schnittstellenkonfiguration (Die Deaktivierung von Schnittstellen sowie die Veränderung der IP-Zuweisungsmethode "Statische Adresse" ist nicht möglich).
- Standardregeln für IP-Dienste (Firewall)
- DDNS
- Uhrzeitsynchronisation
- Log-Einstellungen
- SNMP
- RADIUS

Hinweis

Laden einer Konfiguration auf Security-Module einer Redundanzbeziehung (nur SCALANCE S623/S627-2M ab V4)

Die für das Primärmodul projektierten Eigenschaften einer Redundanzbeziehung müssen sowohl auf das Primärmodul als auch auf das Sekundärmodul geladen werden. Für das Laden der Konfiguration muss jeweils die physische IP-Adresse verwendet werden, über die Ihre Engineering-Station das Security-Modul erreichen kann. Die virtuellen IP-Adressen der Redundanzbeziehung können nicht für das Laden verwendet werden.

Hinweis

Projektierung von Routing bei Verwendung von Router- und Firewallredundanz

In einer Redundanzbeziehung werden ausschließlich diejenigen Routing-Informationen zwischen Primär- und Sekundärmodul synchronisiert, die in den lokalen Security-Einstellungen des Primärmoduls im Eintrag „Routing“ statisch projektiert worden sind. Routing-Einträge, die dynamisch durch die Nutzung von Standard-Routern entstehen, werden nicht synchronisiert. Bei der Verwendung von Router- und Firewallredundanz wird daher empfohlen, möglichst alle bekannten Routen statisch zu projektieren.

Redundanzbeziehungen zwischen Security-Modulen erstellen

Voraussetzung

Die Security-Module SCALANCE S623/S627-2M ab V4 sind keiner anderen Redundanzbeziehung zugeordnet.

Vorgehensweise

1. Selektieren Sie das Security-Modul, das im Normalbetrieb das aktive Security-Modul (Primärmodul) sein soll.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Router- und Firewallredundanz".
3. Aktivieren Sie das Kontrollkästchen "Router- und Firewallredundanz".
4. Wählen Sie aus der Klappliste "Sekundärmodul" das Security-Modul aus, das im Normalbetrieb das passive Security-Modul sein soll.

Ergebnis: Sie haben eine Redundanzbeziehung zwischen den Security-Modulen erstellt.

Redundanzbeziehungen konfigurieren

So erreichen Sie diese Funktion

1. Selektieren Sie das Primärmodul der Redundanzbeziehung.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Router- und Firewallredundanz".

Netzparameter der Redundanzbeziehung projektieren

Konfigurierbarer Parameter	Bedeutung
IP-Adresse	IP-Adresse der virtuellen externen bzw. internen Schnittstelle der Redundanzbeziehung. Die IP-Adresse muss im externen bzw. internen Subnetz des Primärmoduls liegen.
Subnetzmaske	Subnetzmaske der virtuellen externen bzw. internen Schnittstelle der Redundanzbeziehung
MAC-Adresse (anpassbar nur für SCALANCE S623/ S627-2M ab V4.0.1)	MAC-Adresse der virtuellen externen bzw. internen Schnittstelle der Redundanzbeziehung

Für generelle Informationen zur Projektierung von Netzparametern, siehe folgendes Kapitel:
IP-Adressparameter projektieren (Seite 505)

Firewall projektieren

Die Projektierung von IP-Paketfilterregeln für Redundanzbeziehungen erfolgt über das Primärmodul. Zur Verfügung stehen die Kommunikationsrichtungen "Von Extern nach Intern" und " Von Intern nach Extern".

Für generelle Informationen zur Projektierung von IP-Paketfilterregeln im erweiterten Firewall-Modus, siehe folgende Kapitel:
IP-Paketfilter-Regeln definieren (Seite 417)

Adressumsetzung mit NAT/NAPT projektieren

Die Projektierung von Adressumsetzung mit NAT/NAPT für die Redundanzbeziehung erfolgt über das Primärmodul. Für Redundanzbeziehungen kann ausschließlich Source-NAT und NAPT projektiert werden. Bei Source-NAT können Quell-IP-Adressen im internen Subnetz nur gegen die virtuelle externe IP-Adresse der Redundanzbeziehung ausgetauscht werden. Es können keine Alias-IP-Adressen an der externen Schnittstelle der Redundanzbeziehung registriert werden. Bei NAPT ist lediglich die Adressumsetzungsrichtung "Extern nach Intern" projektierbar.

Für generelle Informationen zur Projektierung von Adressumsetzungen mit NAT/NAPT, siehe folgendes Kapitel:
Übersicht zu NAT/NAPT (Seite 434)

Routing projektieren

Die Projektierung von Routen für die Redundanzbeziehung erfolgt über das Primärmodul. Standard-Router müssen für beide Module unter dem Eintrag "Externe Schnittstelle [P1] rot" bzw. "Interne Schnittstelle [P2] grün" eingetragen werden und pro Schnittstelle identisch sein.

Für generelle Informationen zur Projektierung von Routing, siehe folgendes Kapitel: Routen festlegen (Seite 533)

Online-Funktionen - Diagnose und Logging

Übersicht zu den einzelnen Schnittstellen - Eintrag "Schnittstellen-Einstellungen"

Bedeutung

Tabelle 1-55 Online Diagnose: Eintrag "Schnittstellen-Einstellungen"

System- und Statusfunktionen	Bedeutung
Schnittstellen	Tabelle oben: Allgemeine Übersicht zu den Schnittstellen des Security-Moduls. Tabelle unten: Angaben zur Schnittstelle, die über PPPoE betrieben wird. • Zustand: Zeigt an, ob eine Verbindung zum Internet Service Provider (ISP) hergestellt wurde oder nicht. • Aktuelle IP-Adresse: Aktuelle IP-Adresse der Schnittstelle • Gateway: IP-Adresse des Gateways • Primärer dyn. DNS-Dienst: IP-Adresse des primären dyn. DNS-Diensts • Sekundärer dyn. DNS-Dienst: IP-Adresse des sekundären dyn. DNS-Diensts • Fehlercode (numerisch): Fehlerinformationen, falls keine Verbindung zum ISP hergestellt werden konnte.
CARP (nur SCALANCE S623/S627-2M ab V4)	• CARP-Schnittstelle: Anzeige der virtuellen CARP-Schnittstelle • Physikalische Schnittstelle: Physikalische Schnittstelle, an welcher die virtuelle CARP-Schnittstelle betrieben wird (Extern / Intern). • Status: Anzeige, welches der Module aus der Redundanzbeziehung aktiv ist. • MAC-Adresse: MAC-Adresse der virtuellen CARP-Schnittstelle • Bevorzugt: Anzeige, welches der Module aus der Redundanzbeziehung als Primärmodul projektiert ist.

System- und Statusfunktionen	Bedeutung
Medienredundanz (nur SCALANCE S627-2M)	• Schnittstelle: Schnittstelle, die an den MRP-Ring angebunden ist. • Protokoll: Verwendetes Protokoll (MRP) • Ringport 1: Name des ersten Medienmodulports der Schnittstelle, der an den MRP-Ring angebunden ist. • Ringport 2: Name des zweiten Medienmodulports der Schnittstelle, der an den MRP-Ring angebunden ist. • Domain-Name: Name der MRP-Domain. • Diskrepanz: Anzeige, ob sich die Domain des Clients von der des Redundanzmanagers unterscheidet. • Domain-UUID: UUID der MRP-Domain.
Medienmodule (nur SCALANCE S627-2M)	• Port: Port-ID(s) der/des Medienmodulports • Name: Name des Medienmoduls • MLFB: MLFB des Medienmoduls • Revision: Ausgabestand des Medienmoduls • Diskrepanz: Anzeige, ob Unterschiede zwischen den Projektierungsdaten in STEP 7 und den Medienmodulen vorliegen.

Übersicht zu Dyn. DNS-Einstellungen - Eintrag "Dynamisches DNS"

Bedeutung

Tabelle 1-56 Online-Diagnose: "Dynamisches DNS"

System- und Statusfunktionen	Bedeutung
Client-Status	Zeigt an, ob eine Verbindung zum dyn. DNS-Server hergestellt wurde oder nicht.
Aktuelle IP-Adresse	WAN-IP-Adresse, über die das Security-Modul aktuell erreichbar ist.
Definierte IP-Adresse	IP-Adresse, die dem FQDN aktuell zugewiesen ist.
Aktuelle Zeit	Die aktuelle Uhrzeit.
Update erzwingen	Das Security-Modul ermittelt die aktuelle IP-Adresse seines Internetzugangs und sendet eine Aktualisierungsanfrage zu dem oder den projektierten Update-Server(n). Dadurch wird die aktuelle IP-Adresse im Internet publiziert. Der Status wird in den Feldern Primärer bzw. Sekundärer dyn. DNS-Dienst angezeigt. Somit kann z. B. geprüft werden, ob projektierte Daten wie Benutzername und Passwort des dyn. DNS-Accounts korrekt sind.
Suspension aufheben	Hebt die Suspension (Sperre auf Security-Modul für IP-Adressaktualisierungen beim dyn. DNS-Anbieter) auf, z. B. nachdem das dyn. DNS-Anbieterpasswort korrigiert wurde oder der aufgetretene Fehler behoben wurde.
Primärer und sekundärer dyn. DNS-Dienst	
FQDN	Fully Qualified Domain Name, der beim Anbieter registriert ist.
Server-IP-Adresse	IP-Adresse des genutzten Update-Servers.
Erfolgreiche Aktualisierung	Erfolgreiche Aktualisierung beim dyn. DNS-Update-Service.
Letzter Aktualisierungsversuch	Letzter Aktualisierungsversuch beim dyn. DNS-Update-Service.

System- und Statusfunktionen	Bedeutung
Letzte fehlgeschlagene Aktualisierung	Letzter Aktualisierungsfehler beim dyn. DNS-Update-Service.
Fehler-Code	Fehlerstatus des letzten nicht erfolgreichen Aktualisierungsversuchs beim dyn. DNS-Update.

Welche Bedeutung haben die Meldungen

Die Meldungen des letzten DDNS-Aktualisierungsversuches haben folgende Bedeutungen:

Meldung	Bedeutung
Erfolg	
DDNS_OK	Die Aktualisierungsanfrage war erfolgreich.
Verbindungsbezogene Statusmeldungen	
DDNS_E_CON_UDP_SRV_RESOLV_ERR	DNS-Name des Aktualisierungs-Servers unbekannt, FQDN kann nicht über die bekannten DNS-Server aufgelöst werden.
DDNS_E_CON_UDP_SRV_UNREACHABLE	Update-Server nicht erreichbar ("Timeout").
Security-bezogene Statusmeldungen (Fehler)	
DDNS_E_CERT_SUBJECT_INVALID	Der Common-Name des Subjects im Zertifikat stimmt nicht mit dem Domain-Namen des Update-Servers bzw. seiner IP-Adresse überein.
DDNS_E_CERT_UNABLE_TO_GET_ISSUER_CERT	Aussteller-Zertifikat nicht gefunden. Die Zertifikatskette konnte nicht bis zur Root-CA verfolgt werden, weil ein Aussteller-Zertifikat nicht gefunden wurde. Die Trust-Kette ist unvollständig.
DDNS_E_CERT_SIGNATURE_INVALID	Die Signatur eines Zertifikats konnte nicht gelesen werden oder ist ungültig.
DDNS_E_CERT_NO_TRUST	Ein Zertifikat in der Trust-Kette ist ungültig, d. h.: • Noch nicht gültig oder schon abgelaufen • V3-Extensions ungültig • kritische V3-Extension wird nicht unterstützt
DDNS_E_CERT_DEPTH_ZERO_SELFSIGNED_CERT	Der Update-Server hat ein selbstsigniertes Zertifikat geliefert und dieses Zertifikat ist nicht im Certificate-Store für vertrauenswürdige Root-CA-Zertifikate.
DDNS_E_CERT_SELF_SIGNED_CERT_IN_CHAIN	Die Zertifikatskette konnte unter Verwendung nicht vertrauenswürdiger Zertifikate aufgebaut werden, aber es wurde kein passendes Root-CA-Zertifikat im Certificate Store für vertrauenswürdige Zertifikate gefunden.
DDNS_E_CERT_CHAIN_TOO_LONG	Die Zertifikatskette überschreitet die maximal unterstützte Verifikationstiefe.
DDNS_E_CERT_INVALID_CA	Ein CA-Zertifikat ist ungültig, d.h. abgelaufen, noch nicht gültig oder die V3-Extensions sind nicht passend zum vorgesehenen Verwendungszweck (z.B. CA nicht auf TRUE für CA-Zertifikate).
DDNS_E_CERT_KEYUSAGE_UNSUITED	Die in einem Zertifikat der Trust-Chain gesetzten V3-Extensions Key-Usage oder Extended Key-Usage passen nicht zur Verwendung des Zertifikats.

Meldung	Bedeutung
DDNS_E_CERT_EXTENSION_UNSUPPORTED	Ein Zertifikat in der Trustkette verwendete eine als kritisch markierte Extension, die nicht unterstützt wird.
Agent-bezogene Statusmeldungen (Fehler)	
DDNS_E_AGT_BAD_AGENT	• Der Update-Request entspricht nicht dem von cRSP geforderten Aufbau, z.B. fehlen URL-Parameter. • Der Update-Request wurde an einen nicht erlaubten URL auf dem Update-Server gesendet. • Der projektierte Update-String enthält Fehler.

Anzeige der ARP-Tabelle - Eintrag "ARP-Tabelle"

Bedeutung

Anzeige der ARP-Tabelle des Security-Moduls.

Tabelle 1-57 Online Diagnose: Register "ARP-Tabelle"

System- und Statusfunktionen	Bedeutung
ARP-Tabelle	Anzeige der statischen (Proxy-ARP) und dynamischen Einträge der ARP-Tabelle auf dem Security-Modul. Im Register "Veröffentlichungstyp" wird angegeben, ob es sich um einen statisch projektierten Eintrag oder um einen gelernten Eintrag handelt.

An Webseite angemeldete Benutzer - Eintrag "Angemeldete Benutzer"

Bedeutung

Anzeige der Benutzer, die an der Internetseite für benutzerspezifische IP-Regelsätze angemeldet sind.

System- und Statusfunktionen	Bedeutung
Benutzername	Name des angemeldeten Benutzers.
Quell-IP-Adresse	IP-Adresse, mit der sich der Benutzer angemeldet hat.
Restlaufzeit	Restzeit, bis der Benutzer automatisch abgemeldet wird.
Maximale Sitzungsdauer	Projektierte Gesamtzeit der Sitzung.
Abmelden	Der ausgewählte Benutzer wird abgemeldet.

Anzeige der Firewall-Blacklist - Eintrag "Firewall-Blacklist"

Bedeutung

Anzeige der IP-Adressen von Teilnehmern, die die zulässige Anzahl von Verbindungen und Firewall-Zuständen pro Zeit überschritten haben. Diese Teilnehmer werden in die IP-Blacklist der Firewall eingetragen.

Die Anzahl von Verbindungen und Firewall-Zuständen pro Zeit ist nur dann begrenzt, wenn unter dem Eintrag "Standardregeln für IP-Dienste" in den lokalen Security-Einstellungen das Kontrollkästchen "Erweiterte Zustandsoptionen verwenden" aktiviert ist.

Wenn Sie auf die Schaltfläche "Alle löschen" klicken, werden die angezeigten IP-Adressen aus der Firewall-Blacklist des Security-Moduls entfernt. Zudem werden die IP-Adressen nicht mehr angezeigt.

Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit"

So erreichen Sie diese Funktion

1. Selektieren Sie das Security-Modul, dessen Uhrzeit und Datum Sie prüfen bzw. einstellen möchten.
2. Wählen Sie im Kontextmenü den Menübefehl "Online & Diagnose".
3. Wählen Sie in der Online-Diagnose den Eintrag "Funktionen" > "Datum und Uhrzeit".

Lokale Uhrzeit auf dem Security-Modul

In diesem Bereich können Sie die Uhrzeit und das Datum aus dem Security-Modul auslesen und einstellen. Wenn Sie auf die Schaltfläche "Übernehmen" klicken, werden dem Security-Modul die Uhrzeit und das Datum zugewiesen, die aktuell in den Eingabefeldern "Datum" und "Uhrzeit" eingetragen sind.

Lokale Uhrzeit auf dem PC

In diesem Bereich wird die aktuelle Uhrzeit und das aktuelle Datum des PCs angezeigt, auf dem STEP 7 installiert ist. Wenn Sie auf die Schaltfläche "Für Baugruppe übernehmen" klicken, wird dem Security-Modul die aktuelle Uhrzeit und das aktuelle Datum des PCs zugewiesen.

Diagnose im Ghost-Modus - Eintrag "Ghost-Modus"

Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S602 ab V3.1 verfügbar.

Bedeutung

Anzeige von Adressangaben sowie Informationen zu IP-Adressänderungen des internen Teilnehmers.

System- und Statusfunktionen	Bedeutung
Status des SCALANCE S602	Anzeige, in welchem Zustand sich das Security-Modul in Bezug auf den Betrieb im Ghost-Modus befindet.
IP-Adresse	IP-Adresse des internen Teilnehmers (identisch zur externen IP-Adresse des Security-Moduls).
Subnetzmaske	Subnetzmaske des Security-Moduls.
MAC-Adresse	MAC-Adresse des internen Teilnehmers.
Teilnehmer gefunden am	Anzeige, wann der interne Teilnehmer vom Security-Modul erkannt wurde bzw. wann eine IP-Adressänderung am internen Teilnehmer durchgeführt wurde.
Anzahl der IP-Adressänderungen	Anzahl der vom Security-Modul erkannten IP-Adressänderungen.
IP-Adresse	IP-Adresse im externen Netz, für welche das Security-Modul eine Routeninformation benötigt.
Standard-Router	Standard-Router für die IP-Adresse im externen Netz.

Ladefunktionen

Laden einer Konfiguration

Konfiguration laden / Online-Verbindung herstellen

1. Wählen Sie aus der Klappliste "PG/PC-Schnittstelle" des Dialogs "Erweitertes Laden" bzw. "Online verbinden" den Netzwerkadapter aus, über den Sie das Modul erreichen können.
2. Wenn sich das Modul im Zustand der Werkseinstellungen befindet, gehen Sie wie folgt vor:
   - Wählen Sie aus der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle des Moduls aus, mit der Ihre Engineering-Station verbunden ist und für die in den lokalen Security-Einstellungen die zuzuweisende IP-Adresse projektiert ist.
   - Wählen Sie aus der Klappliste die Option "Alle kompatiblen Teilnehmer anzeigen".
   - Klicken Sie auf die Schaltfläche "Suche starten".
   - Ergebnis: Das Modul wird in der Tabelle "Zielgerät auswählen" mit seiner erkannten MAC-Adresse angezeigt.
   - Selektieren Sie den Modul-Eintrag in der Tabelle und klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".
   - Ergebnis: Dem Modul wird die IP-Adresse zugewiesen, die in den lokalen Security-Einstellungen für die ausgewählte Schnittstelle projektiert ist.
3. Wenn sich das Modul nicht im Zustand der Werkseinstellungen befindet, gehen Sie wie folgt vor:
   - Wählen Sie aus der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle / die FQDN-Adresse / die WAN-Adresse des Moduls aus, über welche Ihre Engineering-Station das Modul erreichen kann. STEP 7 verwendet daraufhin für den Zugriff auf das Modul die Adresse, die in den lokalen Security-Einstellungen für die ausgewählte Komponente projektiert ist.
   - Wenn Sie aus der Klappliste die Option "Geräte mit gleichen Adressen anzeigen" auswählen, werden die Teilnehmer angezeigt, deren Adresse mit der der ausgewählten Komponente übereinstimmt. Wenn Sie den Eintrag "Alle kompatiblen Teilnehmer anzeigen" auswählen, werden die Teilnehmer angezeigt, deren Modultyp mit dem des ausgewählten Moduls übereinstimmt.
   - Klicken Sie auf die Schaltfläche "Suche starten".
   - Ergebnis: In der Spalte "Adresse" der Tabelle "Kompatible Teilnehmer im Zielsubnetz" wird die erkannte IP-Adresse / FQDN-Adresse des Moduls angezeigt.
   - Selektieren Sie den Adresseintrag in der Tabelle und klicken Sie auf die Schaltfläche "Laden" bzw. "Verbinden".

Konfigurieren Sie die Module einer Gruppe vorzugsweise über das gemeinsame externe Netz dieser Module (Schnittstelle X1). Falls sich die Engineering-Station in einem internen Netz

befindet, müssen Sie in der Firewall dieses SCALANCE S die IP-Adressen der anderen Module der Gruppe explizit freigeben und dieses Modul als erstes konfigurieren.

Hinweis

Laden einer Konfiguration bei Betrieb im Ghost-Modus (nur SCALANCE S602 ab V3.1)

Wenn Sie das Security-Modul im Ghost-Modus betreiben, nimmt die externe Schnittstelle des Security-Moduls zur Laufzeit die IP-Adresse des internen Teilnehmers an. Bevor Sie eine neue Konfiguration über die externe Schnittstelle auf das Security-Modul laden können, müssen Sie für das Laden einer Konfiguration die IP-Adresse angeben, die das Security-Modul zur Laufzeit vom internen Teilnehmer bezogen hat.
Um herauszufinden, welche IP-Adresse das Security-Modul aktuell hat, können Sie in STEP 7 über den Menübefehl "Online" > "Erreichbare Teilnehmer" nach erreichbaren Teilnehmern suchen.

Hinweis

Laden einer Konfiguration auf Security-Module einer Redundanzbeziehung (nur SCALANCE S623/S627-2M ab V4)

Die für das Primärmodul projektierten Eigenschaften einer Redundanzbeziehung müssen sowohl auf das Primärmodul als auch auf das Sekundärmodul geladen werden. Für das Laden der Konfiguration muss jeweils die physische IP-Adresse verwendet werden, über die Ihre Engineering-Station das Security-Modul erreichen kann. Die virtuellen IP-Adressen der Redundanzbeziehung können nicht für das Laden verwendet werden.

Abweichende Adresse festlegen

Im Dialogbereich "Zielgerät auswählen" haben Sie die Möglichkeit, eine IP-Adresse / eine FQDN-Adresse anzugeben, die von der IP-Adresse / der FQDN-Adresse in den lokalen Security-Einstellungen abweicht. Geben Sie hierzu in die editierbare Zelle der Tabellenspalte "Adresse" die IP-Adresse / die FQDN-Adresse des Moduls ein. Nach dem Ladevorgang ist das Security-Modul über die IP-Adresse / die FQDN-Adresse aus den lokalen Security-Einstellungen erreichbar.

Firmwareversion

Die Projektierung eines SCALANCE S Moduls kann auch auf ein SCALANCE S Modul geladen werden, dessen Firmwareversion höher als die Firmwareversion des SCALANCE S Moduls in STEP 7 ist.

Betriebszustand

Konfigurationen können im laufenden Betrieb der SCALANCE S Module geladen werden. Damit die Änderungen aus einer Konfiguration wirksam werden, muss das SCALANCE S Modul neu gestartet werden.

Hinweis

Besonderheiten

- Solange ein Modul noch keine IP-Parameter eingestellt hat (d. h. vor der ersten Konfiguration), darf sich zwischen Modul und Konfigurationsrechner kein Router befinden.
- Wenn Sie einen PC von der internen auf die externe Schnittstelle des SCALANCE S umstecken, dann werden Zugriffe dieses PCs auf den SCALANCE S für ca. 20 min blockiert.

Konfigurationszustand

Vor jedem Ladevorgang wird die auf dem Security-Modul vorhandene Konfiguration überprüft und mit der zu ladenden Konfiguration aus dem STEP 7 Projekt verglichen. Wenn die Konfiguration des Moduls aus dem aktuell zu ladenden STEP 7 Projekt stammt und Unterschiede zwischen diesen Konfigurationen bestehen, besteht die Möglichkeit, nur die Dateien auf das Security-Modul zu laden, bei welchen Abweichungen zwischen Modul- und Projektkonfiguration bestehen. Dies kann den Ladevorgang unter Umständen beschleunigen.

Übertragen einer Firmware

Das müssen Sie vor dem Übertragen einer neuen Firmware beachten

Zum Übertragen einer neuen Firmware auf ein Security-Modul müssen folgende Voraussetzungen erfüllt sein:

- Sie besitzen die notwendigen Rechte zur Übertragung einer Firmware, siehe Kapitel: Besonderheiten der Benutzerverwaltung für Security-Funktionen (Seite 382).
- Das Security-Modul ist mit einer IP-Adresse projektiert.

Die Übertragung ist sicher

Das Übertragen der Firmware erfolgt über eine gesicherte Verbindung und kann deshalb auch aus dem ungeschützten Netzwerk vorgenommen werden.

Die Firmware selbst ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur authentische Firmware auf das SCALANCE S Modul geladen werden kann.

Neustart nach Übertragung erforderlich

Eine neu geladene Firmware wird erst nach einem Neustart des SCALANCE S Moduls aktiv. Wurde die Übertragung gestört und abgebrochen, so startet die Baugruppe wieder mit dem alten Firmwarestand.

Security für S7-300-/S7-400-/PC-CPs

Firewall einrichten

Lokale Firewall-Regeln für S7-300-/S7-400-/PC-CPs

Übersicht S7-300-/S7-400-/PC-CPs

Paketfilter-Regeln aktivieren

Wenn Sie in den lokalen Security-Einstellungen für die CPs die Security-Funktion aktivieren, sind zunächst alle Zugriffe auf und über den CP zugelassen. Um einzelne Paketfilter-Regeln zu aktivieren, aktivieren Sie das Kontrollkästchen "Firewall aktivieren". Schalten Sie anschließend die gewünschten Dienste frei. Firewall-Regeln, die aufgrund einer Verbindungsprojektierung automatisch angelegt werden, haben Vorrang vor den manuell eingestellten Regeln.

Hinweis

Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Kontrollkästchen "Firewall im erweiterten Modus aktivieren".

Firewall-Projektierung mit VPN

Wird das Security-Modul einer VPN-Gruppe hinzugefügt, wird standardmäßig die Firewall aktiviert. Zudem wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Dies bedeutet, dass über die externe Schnittstelle nur verschlüsselter IPsec-Datentransfer zugelassen ist. Datenverkehr nach extern wird geblockt.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

Verbindungsregeln aktualisieren

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen. Die geänderten Firewall-Regeln werden dann im erweiterten Firewall-Modus angezeigt.

Firewall mit vordefinierten Firewall-Regeln projektieren - CP 343-1 Adv. / 443-1 Adv.

Firewall mit vordefinierten IP-Regeln projektieren - CP 343-1 Adv. / 443-1 Adv.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 1-58 Verfügbare Dienste und Richtungen

Dienst	Von Station/ intern nach extern	Von extern nach intern	Von extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	x	x	alle	Der IP-Verkehr für die ausgewählten Kommunikationsrichtungen wird zugelassen.
Erlaube S7-Protokoll	x	x	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube FTP/ FTPS (expliziter Modus)	x	x	x	TCP-Port 20 TCP-Port 21	Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.
Erlaube HTTP	x	x	x	TCP-Port 80	Zur Kommunikation mit einem Webserver.
Erlaube HTTPS	x	x	x	TCP-Port 443	Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.
Erlaube DNS	x	x	-	TCP-Port 53 UDP-Port 53	Kommunikationsverbindung zu einem DNS-Server wird zugelassen.
Erlaube SNMP	x	x	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.
Erlaube SMTP	x	x	-	TCP-Port 25	Zum Versenden von E-Mails über einen SMTP-Server.
Erlaube NTP	x	x	-	UDP-Port 123	Zur Synchronisation der Uhrzeit.

Tabelle 1-59 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten MAC-Regeln projektieren - CP 343-1 Adv. / 443-1 Adv.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Firewall" > "Vordefinierte MAC-Regeln".

Tabelle 1-60 Verfügbare Dienste und Richtungen

Dienst	Von Station nach extern	Von extern nach Station	Bedeutung
Erlaube MAC-Kommunikation	x	x	Der MAC-Verkehr von Station nach extern und umgekehrt ist zugelassen.
Erlaube ISO-Protokoll	x	x	Der ISO-Verkehr von Station nach extern und umgekehrt ist zugelassen.

Tabelle 1-61 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
MAC-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Station	Extern

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten Firewall-Regeln projektieren - CP1628

Firewall mit vordefinierten IP-Regeln projektieren - CP1628

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Security" > "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 1-62 Verfügbare Dienste und Richtungen

Dienst	Von Extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	alle	Der IP-Verkehr von Extern nach Station wird zugelassen.
Erlaube S7-Protokoll	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube FTP/FTPS (expliziter Modus)	x	TCP-Port 20 TCP-Port 21	Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.
Erlaube HTTP	x	TCP-Port 80	Zur Kommunikation mit einem Webserver.
Erlaube HTTPS	x	TCP-Port 443	Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.
Erlaube DNS	x	TCP-Port 53 UDP-Port 53	Kommunikationsverbindung zu einem DNS-Server wird zugelassen.
Erlaube SNMP	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.
Erlaube SMTP	x	TCP-Port 25	Zum Versenden von E-Mails über einen SMTP-Server.
Erlaube NTP	x	UDP-Port 123	Zur Synchronisation der Uhrzeit.

Tabelle 1-63 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten MAC-Regeln projektieren - CP1628

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Security" > "Firewall" > "MAC-Regeln".

Tabelle 1-64 Verfügbare Dienste und Richtungen

Dienst	Von Station nach Extern	Von Extern nach Station	Bedeutung
Erlaube MAC-Ebene-Kommunikation	x	x	Der MAC-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube ISO-Kommunikation	x	x	Der ISO-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube SiClock	x	x	SiClock-Uhrzeittelegramme von extern zur Station und umgekehrt sind zugelassen.

Tabelle 1-65 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
MAC-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Station	Extern

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

IP-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Security-Modul		Bedeutung
Von	Nach	CP 343-1 Adv. / 443-1 Adv.	CP 1628	
Intern	Station	x	-	Zugriff vom internen Netz auf die Station.
	Any	x	-	Zugriff vom internen auf das externe Netz, VPN-Tunnelpartner und die Station.
Extern	Station	x	x	Zugriff vom externen Netz auf die Station.
	Any	x	-	Zugriff vom externen auf das interne Netz und die Station.

Auswahlmöglichkeiten / Wertebereiche		Security-Modul		Bedeutung
Station	Intern	x	-	Zugriff von der Station auf das interne Netz.
	Extern	x	x	Zugriff von der Station auf das externe Netz.
	Tunnel	x	x	Zugriff von der Station auf VPN-Tunnelpartner.
Tunnel	Station	x	x	Zugriff über VPN-Tunnelpartner auf die Station.
	Any	x	-	Zugriff von VPN-Tunnelpartnern auf das interne Netz und die Station.
Any	Extern	x	-	Zugriff vom internen Netz und der Station auf das externe Netz.

MAC-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Security-Modul		Bedeutung
Von	Nach	CP 343-1 Adv. / 443-1 Adv.	CP 1628	
Extern	Station	x	x	Zugriff vom externen Netz auf die Station.
Station	Extern	x	x	Zugriff von der Station auf das externe Netz.
	Tunnel	x	x	Zugriff von der Station auf VPN-Tunnelpartner.
Tunnel	Station	x	x	Zugriff über VPN-Tunnelpartner auf die Station.

Zugriffsliste projektieren

Modulspezifische Funktion

Diese Funktion ist für CP 1628 nicht verfügbar.

Bedeutung

Mit Hilfe von IP-Zugriffslisten stellen Sie den Zugriffsschutz für bestimmte IP-Adressen ein. Bereits in den lokalen Einstellungen des CPs angelegte Listeneinträge mit den entsprechenden Rechten werden unter dem Eintrag "Firewall" > "IP-Regeln" (erweiterter Firewall-Modus) angezeigt.

Hinweis

Geändertes Verhalten nach Aktivierung von Security

- Nach der Aktivierung von Security für einen CP wirkt der Zugriffsschutz nur noch an der externen Schnittstelle. Damit der Zugriffsschutz auch an der internen Schnittstelle wirkt, konfigurieren Sie im erweiterten Firewall-Modus die entsprechenden Firewall-Regeln.
- Der CP antwortet auch auf ARP-Anfragen von nicht freigegebenen IP-Adressen (Layer 2).
- Wenn die IP-Zugriffsliste eines CPs keine Einträge enthält und Security für den CP aktiviert wird, wird die Firewall aktiviert und auf den CP kann von extern nicht mehr zugegriffen werden. Damit der CP erreichbar ist, konfigurieren Sie im erweiterten Firewall-Modus die entsprechenden Firewall-Regeln.

Auswirkung von Einträgen der IP-Zugriffsliste beim Aktivieren von Security

Wenn Security in den lokalen Einstellungen eines CPs aktiviert wird, werden im erweiterten Firewall-Modus die entsprechenden Regeln angelegt. Wenn in der Zugriffliste eine IP-Adresse angegeben wurde, wird für diese IP-Adresse eine Firewall-Regel "Accept" > "Extern" > "Station" angelegt. Als Quell-IP-Adresse wird hierbei die IP-Adresse aus der IP-Zugriffsliste übernommen. IP-Adressen aus einem definierten IP-Adressbereich werden ebenfalls in entsprechende Firewall-Regeln umgesetzt.

Voraussetzungen zur Bearbeitung

Bevor die angelegten Firewall-Regeln bearbeitet werden können, muss folgende Voraussetzung erfüllt sein:

- für die Bearbeitung über STEP 7: Projektierungsrecht "Security konfigurieren".
- für die Bearbeitung über einen Webserver: Baugruppenrecht "Web: IP Access Control-Liste erweitern".

Die Voraussetzungen zur Bearbeitung der IP-Zugriffslisten außerhalb der lokalen Security-Einstellungen werden in den Kapiteln der jeweiligen CPs beschrieben.

Verbindungsbezogene automatische Firewall-Regeln

Bedeutung

Für Verbindungen, die über CPs projektiert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

Hinweis

UDP-Multicast-Verbindungen manuell freigeben

Für UDP-Multicast-Verbindungen werden keine automatischen Firewall-Regeln angelegt. Um die Verbindungen freizugeben, fügen Sie die entsprechenden Firewall-Regeln im erweiterten Firewall-Modus manuell hinzu.

Je nachdem, wie der Verbindungsaufbau projektiert ist, werden die folgenden Ebene 3-Firewall-Regeln angelegt. Befindet sich das Security-Modul in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel". Dies gilt nur für CPs, die VPN unterstützen.

In die Spalte "Quell-IP-Adresse" bzw. "Ziel-IP-Adresse" dieser Firewall-Regeln wird jeweils die IP-Adresse des Verbindungspartners eingetragen.

CP->extern	Aktion	Von	Nach
aktiv	Accept	Station	Extern
	Drop	Extern	Station
passiv	Drop	Station	Extern
	Accept	Extern	Station
aktiv und passiv	Accept	Extern	Station
	Accept	Station	Extern

CP->intern	Aktion	Von	Nach
aktiv	Accept	Station	Intern
	Drop	Intern	Station
passiv	Drop	Station	Intern
	Accept	Intern	Station
aktiv und passiv	Accept	Intern	Station
	Accept	Station	Intern

Für Ebene-2-Verbindungen werden "Accept"-Regeln für beide Richtungen angelegt. Befindet sich das Security-Modul in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel".

In die Spalte "Quell-MAC-Adresse" bzw. "Ziel-MAC-Adresse" dieser Firewall-Regeln wird jeweils die MAC-Adresse des Verbindungspartners eingetragen.

CP->extern	Aktion	Von	Nach
aktiv, passiv, aktiv und passiv	Accept	Station	Extern
	Accept	Extern	Station

Hinweis

Verändern der Verbindungsprojektierung

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen.

Vereinbarungen für automatisch angelegte Firewall-Regeln

- Priorität
  Die Regeln haben die höchste Priorität und werden daher in der lokalen Regelliste oben eingefügt.
- Regeln löschen
  Die Regeln können nicht gelöscht werden. Das Logging kann aktiviert und Dienste können zugewiesen werden. Außerdem kann eine Übertragungsgeschwindigkeit und ein Kommentar eingefügt werden.
- Aktion umstellen
  Stellen Sie die Aktion von "Accept" auf "Drop" oder umgekehrt um, werden diese beim erneuten Systemabgleich wieder überschrieben. Sollen die vorgenommenen Änderungen bestehen bleiben, wählen Sie als Aktion "Accept*" oder "Drop*". In diesem Fall wird nur die IP-Adresse abgeglichen und Aktion und Richtung bleiben wie eingestellt bestehen. Einstellungen zu Logging, Dienst, Übertragungsgeschwindigkeit und Kommentar bleiben bei einem erneuten Systemabgleich auch ohne das Umstellen der Aktion auf "Accept*" bzw. "Drop*" bestehen. Wird die projektierte Verbindung gelöscht, werden die entsprechenden Regeln aus der Liste entfernt.

Security-Modul in VPN-Gruppe

Standardmäßig wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Deaktivieren Sie das Kontrollkästchen, kann zusätzlich zur Tunnelkommunikation zwischen Tunnelpartnern Kommunikation mit weiteren Netzwerkteilnehmern aufgebaut werden, zu welchen keine Tunnel bestehen.

- Die Kommunikation läuft ungetunnelt, wenn die Partneradresse zu einer in STEP 7 bekannten Station gehört, zu der kein VPN-Tunnel projektiert ist.
- Die Kommunikation läuft durch den VPN-Tunnel, wenn die Partneradresse ein VPN-Endpunkt ist.
- Kann nicht eindeutig zugeordnet werden, ob eine Verbindung innerhalb oder außerhalb des VPN-Tunnels laufen soll, wird die Verbindung dem VPN-Tunnel zugeordnet und ein entsprechender Hinweis wird angezeigt. Die Zuordnung kann im erweiterten Firewall-Modus, z. B. durch Ändern der "Von"-Richtung "Tunnel" auf "Extern", angepasst werden. Damit diese Anpassung bei einem erneuten Systemabgleich nicht wieder überschrieben wird, muss die Aktion "Accept*" bzw. "Drop*" ausgewählt werden.

Hinweis

Soll sichergestellt werden, dass ausschließlich eine Kommunikation durch den Tunnel möglich ist, müssen Sie im erweiterten Firewall-Modus entsprechende Firewall-Regeln, z. B. für interne Teilnehmer oder NDIS-Adressen, anlegen.

Um für einen CP ausschließlich getunnelte Kommunikation zuzulassen, fügen Sie eine Regel mit den folgenden Einstellungen ein:

- "Aktion": "Drop"
- "Von": "Any"
- "Nach": "Extern"

Für CP 1628 fügen Sie eine Regel mit den folgenden Einstellungen ein:

- "Aktion": "Drop"
- "Von": "Station"
- "Nach": "Extern"

Zusätzlich müssen Sie bereits bestehende Firewall-Regeln, die ungetunnelte Kommunikation erlauben, entfernen.

Webserver auf Security-Modul aktivieren

Modulspezifische Funktion

Diese Funktion ist nur für CP 343-1 Adv. / 443-1 Adv. und CP 443-1 OPC UA verfügbar.

Bedeutung

Nach der Aktivierung des Webservers erhalten Sie Zugriff auf die Webseiten der Baugruppe. In den lokalen Security-Einstellungen können Sie den Zugriff auf diese Webseiten für CP 343-1 Adv. / 443-1 Adv. auf das HTTPS-Protokoll beschränken. Dieser Zugriff wird über das Kontrollkästchen "Zugriff nur über HTTPS zulassen" gesteuert. Zusätzlich müssen Sie die Firewall passend konfigurieren. Für CP 443-1 OPC UA ist der Zugriff auf den Webserver nur über HTTPS zulässig.

IPSec-Tunnel: Gruppen anlegen und zuordnen

Interne Netzknoten konfigurieren - Eintrag "Knoten"

Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben

Auswahlmöglichkeiten

Wählen Sie aus, ob die VPN-Verbindungspartner im Routing-Modus (SCALANCE S / M) auf den CP und/oder auf das interne Subnetz des CPs Zugriff haben dürfen.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten".
3. Konfigurieren Sie den Zugriff für die VPN-Verbindungspartner des CPs im Routing-Modus (SCALANCE S / M):
   - Verbindung erlauben zum CP (GBit-Schnittstelle)
   - Verbindung erlauben zum internen Subnetz (PROFINET-Subnetz)

Durch Tunnel erreichbare NDIS-Knoten bei PC-CPs manuell konfigurieren

Durch Tunnel erreichbare NDIS-Knoten konfigurieren

Die internen Knoten werden gelernt und dynamisch den Routen zugeordnet. Dies betrifft die NDIS-IP-Adressen des Windows-PCs.

Gehen Sie so vor

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten" > "Durch Tunnel erreichbare NDIS-Knoten".
3. Geben Sie die NDIS-IP-Adressen ein.

Online-Funktionen - Test / Diagnose und Logging

Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln"

Modulspezifische Funktion

Diese Funktion ist nur für CP 343-1 Adv. / 443-1 Adv. verfügbar.

Bedeutung

Anzeige der IP-Adressen bzw. der IP-Adressbereiche, die über HTTP oder HTTPS dynamisch freigeschaltet oder von einem Benutzer nachgeladen wurden. Für die freigeschalteten IP-Adressen werden die ihnen zugewiesenen Zugriffsrechte auf den S7-CP angezeigt. Eine Aktualisierung der IP-Adressen in diesem Register kann durch folgende Ereignisse erfolgen:

- Erweiterung/Veränderung der IP Access Control-Liste
- Aktualisierung der Firewall-Regeln
- Dynamische Erweiterungen, die vom CP zur Laufzeit eingetragen werden, z.B. PROFINET IO-Devices

Da hier nur die dynamisch aktualisierten Firewall-Regeln angezeigt werden, müssen in eine vollständige Betrachtung des aktuellen Firewall-Zustands der Baugruppe auch diejenigen Firewall-Regeln einbezogen werden, die offline projektiert und auf die Station geladen wurden.

Security für S7-1200-/S7-1500-CPs

Firewall einrichten

Lokale Firewall-Regeln für S7-1200-/S7-1500-CPs

Übersicht Lokale Firewall-Regeln S7-1200-/S7-1500-CPs

Paketfilter-Regeln aktivieren

Wenn Sie in den lokalen Security-Einstellungen für die CPs die Security-Funktion aktivieren, sind zunächst alle Zugriffe auf und über den CP zugelassen. Um einzelne Paketfilter-Regeln zu aktivieren, aktivieren Sie das Kontrollkästchen "Firewall aktivieren". Schalten Sie anschließend die gewünschten Dienste frei. Firewall-Regeln, die aufgrund einer Verbindungsprojektierung automatisch angelegt werden, haben Vorrang vor den manuell eingestellten Regeln.

Hinweis

Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Kontrollkästchen "Firewall im erweiterten Modus aktivieren".

Verbindungsregeln aktualisieren

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen. Die geänderten Firewall-Regeln werden dann im erweiterten Firewall-Modus angezeigt.

Firewall mit vordefinierten Firewall-Regeln projektieren - CP 1543-1 / CP 1543SP-1

Firewall mit vordefinierten IP-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 1-66 Verfügbare Dienste und Richtungen

Dienst	Von Extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	alle	Der IP-Verkehr von Extern nach Station wird zugelassen.
Erlaube S7-Protokoll	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube FTP/ FTPS* (expliziter Modus)	x	TCP-Port 20 TCP-Port 21	Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.
Erlaube HTTP	x	TCP-Port 80	Zur Kommunikation mit einem Webserver.
Erlaube HTTPS	x	TCP-Port 443	Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.
Erlaube SNMP	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.
Erlaube Security-Diagnose	x	TCP-Port 8448	Security-Diagnose erlauben.

* Für CP 1543SP-1 nicht verfügbar

Tabelle 1-67 IP-Routing über Rückwandbus (nur für CP 1543-1 ab V2.1)

Dienst	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation über Rückwandbus	alle	Der IPv4-Verkehr über den Rückwandbus wird zugelassen. Wenn die Option "IP-Routing zwischen Kommunikationsmodulen" aktiviert ist, ist dieses Kontrolllkästchen aktiviert und kann nicht deaktiviert werden. Bei deaktivierter Option ist dieses Kontrollkästchen deaktiviert und kann nicht aktiviert werden.

Tabelle 1-68 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten IPv6-Regeln projektieren

Bedeutung

Die vordefinierten IPv6-Regeln bieten die Möglichkeit, die Firewall bezüglich Diensten zu konfigurieren, in denen IPv6 verwendet wird. Durch das Freischalten einer vordefinierten IPv6-Regel in den lokalen Security-Einstellungen des CP 1543-1 V1.1 werden gleichzeitig die systemdefinierten ICMPv6-Dienste in der Firewall freigeschalten, die in den globalen Security-Funktionen unter "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren", Register "ICMP" ersichtlich sind. Die Firewall des CP 1543-1 V1.0 lässt ICMPv6-Pakete auch ohne das Freischalten einer vordefinierten IPv6-Regel passieren.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IPv6-Regeln".

Tabelle 1-69 Verfügbare Dienste und Richtungen

Dienst	Von Extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	alle	Der IP-Verkehr von Extern nach Station ist zugelassen.
Erlaube S7-Protokoll	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube FTP/ FTPS* (expliziter Modus)	x	TCP-Port 20 TCP-Port 21	Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.
Erlaube SNMP	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.

* Für CP 1543SP-1 nicht verfügbar

Tabelle 1-70 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IPv6-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten MAC-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Firewall" > "Vordefinierte MAC-Regeln".

Tabelle 1-71 Verfügbare Dienste und Richtungen

Dienst	Von Station nach extern	Von extern nach Station	Freigegebene Ports	Bedeutung
Erlaube MAC-Kommunikation	x	x	-	Der MAC-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube ISO-Protokoll*	x	x	-	Der ISO-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube DCP	x	x	-	Der DCP-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube LLDP*	x	x	-	Der LLDP-Verkehr von extern zur Station und umgekehrt ist zugelassen.

* Für CP 1543SP-1 nicht verfügbar

Tabelle 1-72 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
MAC-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Station	Extern

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten Firewall-Regeln projektieren - S7-1200-CPs

Firewall mit vordefinierten IP-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 1-73 Verfügbare Dienste und Richtungen

Dienst	Von Extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	alle	Der IP-Verkehr von Extern nach Station ist zugelassen.
Erlaube S7-Protokoll	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube HTTP	x	TCP-Port 80	Zur Kommunikation mit einem Webserver.
Erlaube HTTPS	x	TCP-Port 443	Zur gesicherten Kommunikation mit einem Webserver, z.B. zur Web-Diagnose.
Erlaube SNMP	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.
Erlaube Security-Diagnose	x	TCP-Port 8448	Security-Diagnose erlauben.

Tabelle 1-74 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten IPv6-Regeln projektieren

Bedeutung

Die vordefinierten IPv6-Regeln bieten die Möglichkeit, die Firewall bezüglich Diensten zu konfigurieren, in denen IPv6 verwendet wird. Durch das Freischalten einer vordefinierten IPv6-Regel in den lokalen Security-Einstellungen eines IPv6-fähigen S7-1200-CPs werden gleichzeitig die systemdefinierten ICMPv6-Dienste in der Firewall freigeschalten, die in den globalen Security-Funktionen unter "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren", Register "ICMP" ersichtlich sind.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IPv6-Regeln".

Tabelle 1-75 Verfügbare Dienste und Richtungen

Dienst	Von Extern nach Station	Freigegebene Ports	Bedeutung
Erlaube IP-Kommunikation	x	alle	Der IP-Verkehr von Extern nach Station ist zugelassen.
Erlaube S7-Protokoll	x	TCP-Port 102	Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.
Erlaube SNMP	x	TCP-Port 161/162 UDP-Port 161/162	Zur Überwachung von SNMP-fähigen Netzteilnehmern.

Tabelle 1-76 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
IP-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IPv6-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

Firewall mit vordefinierten MAC-Regeln projektieren

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Firewall" > "Vordefinierte MAC-Regeln".

Tabelle 1-77 Verfügbare Dienste und Richtungen

Dienst	Von Station nach extern	Von extern nach Station	Freigegebene Ports	Bedeutung
Erlaube MAC-Kommunikation	x	x	-	Der MAC-Verkehr von extern zur Station und umgekehrt ist zugelassen.
Erlaube DCP	x	x	-	Der DCP-Verkehr von extern zur Station und umgekehrt ist zugelassen.

Tabelle 1-78 Logging

Option	Aktion bei Aktivierung	Zutreffende Firewall-Regel		
MAC-Log-Einstellungen		Aktion	Von	Nach
Aufzeichnen getunnelter Pakete	Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt.	Accept	Station	Tunnel
		Accept	Tunnel	Station
Aufzeichnen blockierter eingehender Pakete	Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Extern	Station
Aufzeichnen blockierter ausgehender Pakete	Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.	Drop	Station	Extern

Hinweis

Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

IP-Paketfilter-Richtungen S7-1200-/S7-1500-CPs

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Bedeutung
Von	Nach	
Extern	Station	Zugriff vom externen Netz auf die Station.
	Rückwandbus*	Zugriff vom externen Netz auf den Rückwandbus der Station
	Any**	Zugriff vom externen Netz auf alle unterstützten Richtungen.
Station	Extern	Zugriff von der Station auf das externe Netz.
	Tunnel	Zugriff von der Station auf VPN-Tunnelpartner.
	Rückwandbus*	Zugriff von der Station auf deren Rückwandbus.
	Any**	Zugriff von der Station auf alle unterstützten Richtungen.
Tunnel	Station	Zugriff von VPN-Tunnelpartnern auf die Station.
	Rückwandbus*	Zugriff von VPN-Tunnelpartnern auf den Rückwandbus der Station.
	Any**	Zugriff von VPN-Tunnelpartnern auf alle unterstützten Richtungen.
Rückwandbus*	Extern	Zugriff vom Rückwandbus der Station auf das externe Netz.
	Tunnel	Zugriff vom Rückwandbus der Station auf VPN-Tunnelpartner.
	Station	Zugriff vom Rückwandbus auf die Station.
	Any**	Zugriff vom Rückwandbus auf alle unterstützten Richtungen.
Any**	Extern	Zugriff von allen unterstützten Richtungen auf das externe Netz.
	Tunnel	Zugriff von allen unterstützten Richtungen auf VPN-Tunnelpartner.
	Station	Zugriff von allen unterstützten Richtungen auf die Station.
	Rückwandbus*	Zugriff von allen unterstützten Richtungen auf den Rückwandbus der Station.

* Nur für CP 1543-1 ab V2.1 mit IPv4-Kommunikation. Bei aktivierter Option "IP-Routing zwischen Kommunikationsmodulen" kann über diese Richtung IP-Routing über den Rückwandbus zugelassen / gesperrt werden.

** Nur für CP 1543-1 ab V2.1

MAC-Paketfilter-Richtungen S7-1200-/S7-1500-CPs

Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

Folgende Richtungen sind verfügbar

Auswahlmöglichkeiten / Wertebereiche		Bedeutung
Von	Nach	
Extern	Station	Zugriff vom externen Netz auf die Station.
Station	Extern	Zugriff von der Station auf das externe Netz.
	Tunnel	Zugriff von der Station auf VPN-Tunnelpartner.
Tunnel	Station	Zugriff von VPN-Tunnelpartnern auf die Station.

Verbindungsbezogene automatische Firewall-Regeln

Bedeutung

Für Verbindungen, die über CPs projektiert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

Hinweis

UDP-Multicast- und UDP-Broadcast-Verbindungen manuell freigeben

Für UDP-Multicast- und UDP-Broadcast-Verbindungen werden keine automatischen Firewall-Regeln angelegt. Um die Verbindungen freizugeben, fügen Sie die entsprechenden Firewall-Regeln im erweiterten Firewall-Modus manuell hinzu.

Je nachdem, wie der Verbindungsaufbau projektiert ist, werden die folgenden Ebene 3-Firewall-Regeln angelegt. Befindet sich der CP in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel".

In die Spalte "Quell-IP-Adresse" bzw. "Ziel-IP-Adresse" dieser Firewall-Regeln wird jeweils die IP-Adresse des Verbindungspartners eingetragen.

CP->extern	Aktion	Von	Nach
aktiv	Drop	Extern	Station
	Accept	Station	Extern
passiv	Drop	Station	Extern
	Accept	Extern	Station
aktiv und passiv	Accept	Extern	Station
	Accept	Station	Extern

Für Ebene-2-Verbindungen werden "Accept"-Regeln für beide Richtungen angelegt. Befindet sich der CP in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel".

In die Spalte "Quell-MAC-Adresse" bzw. "Ziel-MAC-Adresse" dieser Firewall-Regeln wird jeweils die MAC-Adresse des Verbindungspartners eingetragen.

CP->extern	Aktion	Von	Nach
aktiv, passiv, aktiv und passiv	Accept	Station	Extern
	Accept	Extern	Station

Hinweis

Verändern der Verbindungsprojektierung

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen.

Vereinbarungen für automatisch angelegte Firewall-Regeln

- Priorität
  Die Regeln haben die höchste Priorität und werden daher in der lokalen Regelliste oben eingefügt.
- Regeln löschen
  Die Regeln können nicht gelöscht werden. Das Logging kann aktiviert und Dienste können zugewiesen werden. Außerdem kann eine Übertragungsgeschwindigkeit und ein Kommentar eingefügt werden.
- Aktion umstellen
  Stellen Sie die Aktion von "Accept" auf "Drop" oder umgekehrt um, werden diese beim erneuten Systemabgleich wieder überschrieben. Sollen die vorgenommenen Änderungen bestehen bleiben, wählen Sie als Aktion "Accept*" oder "Drop*". In diesem Fall wird nur die IP-Adresse abgeglichen und Aktion und Richtung bleiben wie eingestellt bestehen. Einstellungen zu Logging, Dienst, Übertragungsgeschwindigkeit und Kommentar bleiben bei einem erneuten Systemabgleich auch ohne das Umstellen der Aktion auf "Accept*" bzw. "Drop*" bestehen. Wird die projektierte Verbindung gelöscht, werden die entsprechenden Regeln aus der Liste entfernt.

Security-Modul in VPN-Gruppe

Standardmäßig wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Deaktivieren Sie das Kontrollkästchen, kann zusätzlich zur Tunnelkommunikation zwischen Tunnelpartnern Kommunikation mit weiteren Geräten aufgebaut werden, zu welchen keine Tunnel bestehen.

- Die Kommunikation läuft ungetunnelt, wenn die Partneradresse zu einer in STEP 7 bekannten Station gehört, zu der kein VPN-Tunnel projektiert ist.
- Die Kommunikation läuft durch den VPN-Tunnel, wenn die Partneradresse ein VPN-Endpunkt ist.

Hinweis

Soll sichergestellt werden, dass ausschließlich eine Kommunikation durch den Tunnel möglich ist, müssen Sie im erweiterten Firewall-Modus entsprechende Firewall-Regeln anlegen.

Um für einen CP ausschließlich getunnelte Kommunikation zuzulassen, fügen Sie eine Regel mit den folgenden Einstellungen ein:

- "Aktion": "Drop"
- "Von": "Station"
- "Nach": "Extern"

Zusätzlich müssen Sie bereits bestehende Firewall-Regeln, die ungetunnelte Kommunikation erlauben, entfernen.

Über Rückwandbus erreichbare Subnetze für CP 1543-1 konfigurieren

Voraussetzung

- Das Security-Modul CP 1543-1 ab V2.1 ist Teilnehmer einer VPN-Gruppe und die Option "IP-Routing zwischen Kommunikationsmodulen" ist aktiviert.

So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN" > "Knoten" > "Durch Tunnel erreichbare Subnetze".

Projektierung von über Rückwandbus erreichbaren Subnetzen

Die VPN-Kommunikation mit Netzwerkknoten und Subnetzen, die über den Rückwandbus einer Station erreichbar sind, ist nur möglich, wenn diese für die VPN-Kommunikation freigegeben werden. Im Eintrag "Durch Tunnel erreichbare Subnetze" können hierfür maximal 16 IP-Adressen bzw. Subnetze angegeben werden. IP-Adressen müssen mit einer 32-Bit-Subnetzmaske angegeben werden, Subnetze mit einer zulässigen Subnetzmaske im Bereich 1 ... 31 Bit. Broadcast-Adressen dürfen nicht angegeben werden.

Automatische OpenVPN-Konfiguration über SINEMA Remote Connect Server

Automatische OpenVPN-Konfiguration und Open-VPN-Tunnelaufbau

Unterstützte Module

Die Funktion steht für einige Security-Module und Telecontrol-Kommunikationsmodule zur Verfügung.

Funktion

Das Modul kann eine OpenVPN-Verbindung zur Applikation "SINEMA Remote Connect Server" aufbauen. Die dafür zu verwendenden OpenVPN-Verbindungsparameter werden vom SINEMA Remote Connect-Server festgelegt. Das Modul kann diese Verbindungsparameter zyklisch abrufen und erhält dadurch automatisch die zu verwendende OpenVPN-Konfiguration.

Vorgehensweise

1. Selektieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "VPN".
3. Aktivieren Sie das Kontrollkästchen "VPN aktivieren" und wählen Sie den VPN-Verbindungstyp "Automatische OpenVPN-Konfiguration über SINEMA Remote Connect Server" aus.
4. Projektieren Sie die folgenden Parameter für die OpenVPN-Verbindung:

Parameter	Bedeutung
SINEMA RC Adresse	IPv4- / IPv6-Adresse, FQDN oder PROFINET-Gerätename des SINEMA Remote Connect-Servers, von dem das Modul die Verbindungsparameter für die OpenVPN-Verbindung abruft.
SINEMA RC Port	Port für die Verbindung zum SINEMA Remote Connect-Server, die zum Abruf der Verbindungsparameter für die OpenVPN-Verbindung verwendet wird.
CA-Zertifikat	Das Modul authentifiziert sich beim SINEMA Remote Connect-Server mit einem CA-Zertifikat, das vom Server heruntergeladen und für das Modul ausgewählt wurde. Es können nur Zertifikate ausgewählt werden, die dem Modul über dessen lokalen Zertifikatsmanager zugeordnet wurden. Für die Authentifizierung des SINEMA Remote Connect-Servers prüft das Modul das CA-Zertifikat des Servers. Hierbei wird auch geprüft, ob sich die aktuelle Uhrzeit im Modul im Gültigkeitszeitraum des Zertifikats befindet.
Fingerabdruck	Das Modul authentifiziert sich beim SINEMA Remote Connect-Server durch die Eigenschaft "Fingerabdruck" des CA-Zertifikats vom SINEMA Remote Connect-Server.
Geräte-ID	Die Geräte-ID wird von SINEMA Remote Connect Server für jedes Modul vergeben, das auf dem SINEMA Remote Connect-Server projektiert wird. Die Geräte-ID dient der Identifizierung eines Moduls.
Gerätepasswort	Passwort, das für die Geräte-ID auf dem SINEMA Remote Connect-Server projektiert ist.

Parameter	Bedeutung
Aktualisierungsintervall	Intervall in Minuten, in dem das Modul die zu verwendenden OpenVPN-Verbindungsparameter vom SINEMA Remote Connect-Server abruft. Ob und mit welchen Verbindungsparametern OpenVPN-Verbindungen aufgebaut werden können, wird durch den SINEMA Remote Connect-Server festgelegt. Mit dem Wert "0" wird die Aktualisierung deaktiviert. Mindestwert für die Aktualisierung: 10 Minuten Maximalwert für die Aktualisierung: 10080 Minuten
Verbindungsart	Die Verbindungsart legt fest, wann das Modul eine OpenVPN-Verbindung zum SINEMA Remote Connect-Server aufbaut: • Permanent: Das Modul baut nach dem Lesen der OpenVPN-Verbindungsparameter eine OpenVPN-Verbindung auf. Die OpenVPN-Verbindung bleibt bis zum Ändern der Verbindungsparameter durch den SINEMA Remote Connect-Server bestehen. • PLC-Trigger: Das Modul baut eine OpenVPN-Verbindung auf, sobald die ausgewählte PLC-Variable den Wert "TRUE" hat.
PLC-Variable für Verbindungsaufbau	Auswahl einer Variable der Station vom Typ "BOOL". Wenn die Variable den Wert "TRUE" annimmt, baut das Modul eine OpenVPN-Verbindung zum SINEMA Remote Connect-Server auf. Wenn die Variable den Wert "FALSE" annimmt, wird die OpenVPN-Verbindung abgebaut.

Diagnose der automatischen OpenVPN-Konfiguration und OpenVPN-Verbindungen

Bedeutung

Diese Diagnoseseite zeigt den Status der automatischen OpenVPN-Konfiguration und der OpenVPN-Verbindungen an.

Der Status der automatischen OpenVPN-Konfiguration verdeutlicht, ob das Modul die Konfigurationsdatei mit den zu verwendenden OpenVPN-Verbindungsparametern vom SINEMA Remote Connect Server abrufen konnte.

Für OpenVPN-Verbindungen zum SINEMA Remote Connect Server werden die Verbindungszustände sowie deren Ursachen angezeigt.

1.3.5.2 Security für SCALANCE S-600 projektieren

Unterstützte Geräte

Unterstützte Geräte

Die in diesem Hilfeabschnitt beschriebenen Security-Funktionen werden von folgenden Produkten unterstützt:

- SCALANCE S615 V5
- SCALANCE SC-600:
  - SCALANCE SC632-2C V1
  - SCALANCE SC636-2C V1
  - SCALANCE SC642-2C V1
  - SCALANCE SC646-2C V1

Allgemeine Benennung "Gerät"

Im vorliegenden Abschnitt des Informationssystems wird die Benennung "Gerät" stellvertretend für die oben genannten Produktnamen verwendet. Die Funktionsunterschiede finden Sie in Kapitel Hardware-Ausstattung und Systemfunktionen (Seite 585). Abhängig vom ausgewählten Gerät sind auf den einzelnen Konfigurationsseiten unterschiedliche Parameter verfügbar.

Hardware-Ausstattung und Systemfunktionen

Verfügbarkeit der Hardware-Ausstattung

Die nachfolgende Tabelle zeigt die Hardware-Ausstattung der Geräte

Technische Änderungen sind vorbehalten.

	SCALANCE S615	SCALANCE SC632-2C SCALANCE SC642-2C	SCALANCE SC636-2C SCALANCE SC646-2C
Insgesamt nutzbare Ports	5 RJ45 Ports	2 RJ45 Ports	6 RJ45 Ports
Stecktransceiver-Steckplätze (SFP)	-	2	2
Elektrische Anschlüsse	5	2	6
Combo Ports	-	2	2
PLUG-Steckplatz	1	1	1

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Geräten.

Technische Änderungen sind vorbehalten.

		SCALANCE S615	SCALANCE SC-600
Informationen	Versionen	✓	✓
	I&M	✓	-
	ARP-Tabelle	✓	✓
	Log-Tabellen	✓	✓
	Fehler	✓	✓
	SNMP	✓	✓
	LLDP	✓	✓
	Routing-Tabelle	✓	✓
	IPsec VPN	✓	✓ (nur SCALANCE SC64x-2C)
	SINEMA RC	✓ (nur mit KEY-PLUG)	✓
	OpenVPN-Client	✓	-
	Redundanz	✓	-
	Security	✓	✓

		SCALANCE S615	SCALANCE SC-600
System	Konfiguration	✓	✓ (kein Telnet-Server, kein SMTP-Client)
	Allgemein	✓	✓
	DNS	✓	✓
	TFTP	✓	✓
	SFTP	✓	-
	Ereignisse	✓	✓
	SMTP-Client	✓	-
	PLUG	✓	-
	DHCP-Client	✓	✓ (nur IPv4)
	DHCP-Server	✓	✓
	SNMP	✓	✓
	Systemzeit	✓	✓
	Auto-Logout	✓	✓
	Taster	✓ (keine Meldemaske)	✓
	Syslog-Client	✓	✓
	Ports	✓ (unter "Schnittstellen")	✓
	Fehlerkontrolle	✓ (nur Link Change)	✓
	Port-Diagnose (nur Online verfügbar)	-	✓
	cRSP / SRS	✓	✓
	Proxy-Server	✓	✓
	SINEMA RC	✓ (nur mit KEY-PLUG)	✓
Schnittstellen	Ethernet	✓	-
	PPP	✓	-
Layer 2	Port-basiertes VLAN	✓	✓
	Dynamic MAC Aging	✓	✓
	LLDP	✓	✓
	Spanning Tree	✓	-
Layer 3	Statische Routen	✓	✓
	Subnetze	✓	✓
	NAT	✓	✓

		SCALANCE S615	SCALANCE SC-600
Security	Benutzer	✓ (nur im WBM)	✓
	AAA (Authentication, Authorization, Accounting)	✓	-
	Zertifikate	✓	✓
	Firewall	✓	✓
	IPsec VPN	✓	✓ (nur SCALANCE SC64x-2C)
	OpenVPN	✓	-

Rechtlicher Hinweis

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung qualifiziertem Personal gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. **Qualifiziertes Personal** ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Wissenswertes

IPv4-Adresse, Subnetzmaske und Adresse des Netzübergangs

Wertebereich für IPv4-Adresse

Die IPv4-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 141.80.0.16

Adressformat IPv4 - Notation

Eine IPv4-Adresse besteht aus 4 Byte. Jedes Byte wird dezimal dargestellt und ist durch einen Punkt vom vorherigen getrennt.

XXX.XXX.XXX.XXX

XXX steht für eine Zahl zwischen 0 und 255

Die IPv4-Adresse besteht aus zwei Teilen:

- Der Adresse des (Sub-)Netzes
- Der Adresse des Teilnehmers (im allgemeinen auch Endteilnehmer, Host oder Netzknoten genannt)

Wertebereich für Subnetzmaske

Die Subnetzmaske besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 255.255.0.0

Die 4 Dezimalzahlen der Subnetzmaske müssen in ihrer binären Darstellung von links eine Folge von lückenlosen Werten "1" und von rechts eine Folge von lückenlosen Werten "0" enthalten.

Die Werte "1" bestimmen die Netznummer innerhalb der IPv4-Adresse. Die Werte "0" die Host-Adresse innerhalb der IPv4-Adresse.

Beispiel:

richtige Werte:

255.255.0.0 D = 1111 1111.1111 1111.0000 0000.0000 0000 B

255.255.128.0 D = 1111 1111.1111 1111.1000 0000.0000 0000 B

255.254.0.0 D = 1111 1111.1111 1110.0000 0000.0000.0000 B

falscher Wert:

255.255.1.0 D = 1111 1111.1111 1111.0000 0001.0000 0000 B

Zusammenhang IPv4-Adresse und Subnetzmaske

Die erste Dezimalzahl der IPv4-Adresse (von links) bestimmt den Aufbau der Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt (für "x" steht die Host-Adresse):

Erste Dezimalzahl der IPv4-Adresse	Subnetzmaske
0 bis 127	255.x.x.x
128 bis 191	255.255.x.x
192 bis 223	255.255.255.x

Classless Inter-Domain Routing (CIDR)

CIDR ist ein Verfahren das mehrerer IPv4-Adressen zu einem Adressbereich zusammenfasst, indem eine IPv4-Adresse mit ihrer Subnetzmaske kombiniert dargestellt wird. Dazu wird an die IPv4-Adresse ein Suffix angehängt, das die Anzahl der auf 1 gesetzten Bits der Netzmaske angibt. Durch die CIDR-Notation lassen sich Routing-Tabellen reduzieren und die verfügbaren Adressbereiche besser ausnutzen.

Beispiel:

IPv4-Adresse 192.168.0.0 mit Subnetzmaske 255.255.255.0

Der Netzanteil der Adresse umfasst in der binären Darstellung 3 x 8 Bits, also 24 Bits.

Daraus ergibt sich die CIDR-Notation 192.168.0.0/24.
Der Host-Anteil umfasst in der binären Darstellung 1 x 8 Bits. Daraus ergibt sich der Adressbereich von 28, also 256 mögliche Adressen.

Wertebereich für Adresse des Netzübergangs

Die Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 141.80.0.1

Zusammenhang IPv4-Adresse und Adresse des Netzübergangs

Die IPv4-Adresse und die Adresse des Netzübergangs dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

Beispiel:

Sie haben eingegeben: für Subnetzmaske 255.255.255.0; für IPv4-Adresse 141.30.0.5 und für die Adresse des Netzübergangs 141.30.128.0. Die IPv4-Adresse und die Adresse des Netzübergangs dürfen nur in der 4. Dezimalzahl einen unterschiedlichen Wert haben. Im Beispiel ist aber die 3. Stelle schon unterschiedlich.

Im Beispiel müssen Sie also alternativ ändern:

die Subnetzmaske auf: 255.255.0.0 oder

die IPv4-Adresse auf: 141.30.128.1 oder

die Adresse des Netzübergangs auf: 141.30.0.1

ICMP

Die Abkürzung ICMP steht für Internet Control Message Protocol (RFC792) und dient zum Austausch von Fehler- und Informationsmeldungen.

- Fehlermeldung
  Informiert den Absender des IP-Telegramms, dass beim Weiterleiten des Telegramms ein Fehler oder ein Parameterproblem aufgetreten ist.
- Informationsmeldung
  Kann Informationen zur Zeitmessung, zur Adressmaske, zur Erreichbarkeit des Ziels oder zum Auffinden des Routers enthalten.

Aufbau des ICMP-Datenpakets

0	4	8	12	16	20	24	28	31
ICMP-Pakettyp Art der Meldung		Code Weitere Details der Meldung		Prüfsumme				
Daten (optional)								

- **ICMP-Pakettyp**
  Die wichtigsten ICMP-Pakettypen sind:
  - Redirect
    Der Router teilt dem Host in einem seiner Subnetze mit, dass es eine bessere Route zum Ziel gibt. Auf diesen ICMP-Pakettyp wird in der folgenden Beschreibung näher eingegangen.
  - Destination Unreachable
    IP-Telegramm ist nicht zustellbar.
  - Time Exceeded
    Zeitlimit überschritten
  - Echo-Request
    Echo-Frage, besser bekannt als Ping.
- **Code**
  Der Code beschreibt den ICMP-Pakettyp genauer. Die Auswahl ist abhängig vom gewählten ICMP-Pakettyp. Bei "Destination Unreachable" ist z. B. "Code 1" Host ist nicht erreichbar.

Eine vollständige Liste der ICMP-Pakettypen und Codes finden Sie auf der Webseite von IANA (http://www.iana.org/assignments/icmp-parameters).

ICMP-Pakettyp 5 - Redirect

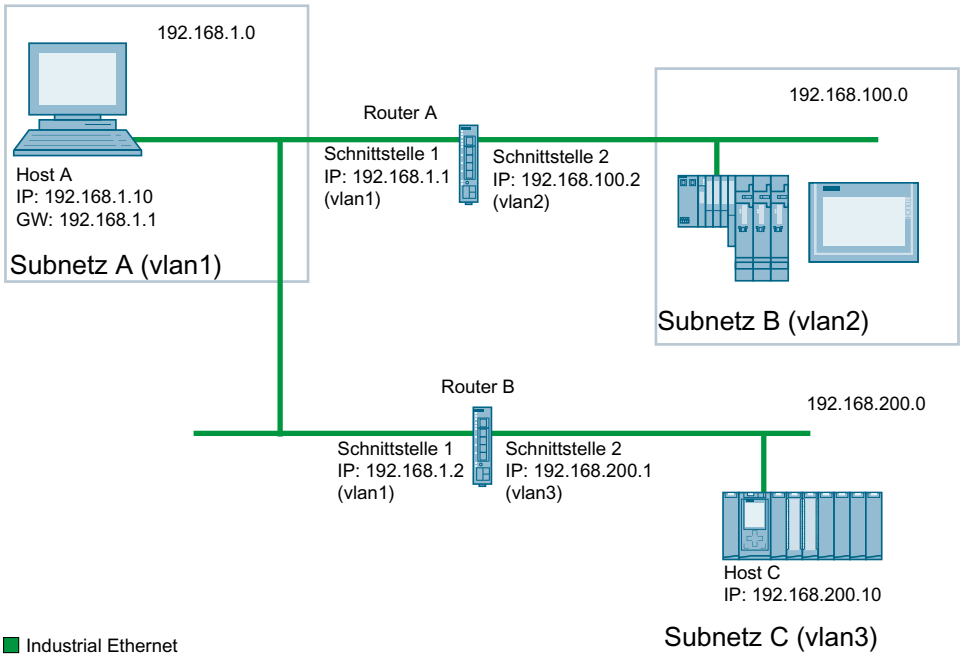

Der Host A will an den Host C ein IP-Telegramm senden. Der Host C liegt nicht im gleichen Subnetz wie der Host A. Deshalb sendet der Host A das IP-Telegramm an seinen Default-Gateway. Das Default-Gateway von Host A ist die Schnittstelle 1 des Routers A. Der Router A kann das IP-Telegramm nicht weiterleiten, da er das Zielnetzwerk nicht kennt. Über seine Routingtabelle weiß Router A jedoch, dass das Subnetz C über den Router B erreichbar ist. Der Router B verbindet das Subnetz A mit dem Subnetz C. Der Router A schickt an den Host

A eine Redirect-Nachricht. Darin weist der Router A den Host A an, die IP-Telegramme an Host C zukünftig über Router B zu senden, dessen IP-Adresse in der Redirect-Nachricht enthalten ist. Das initiale IP-Telegramm wird vom Router A direkt an Router B gesendet, der es an Host C weitergibt.

Bedingungen für den Versand von Redirect-Nachrichten

- Das IP-Telegramm wird über die gleiche Schnittstelle des Routers A empfangen und gesendet.
- Die Quell-IP-Adresse (Host A) ist aus dem gleichen Subnetz wie die Next-Hop-Adresse (Router B) in der Routingtabelle.
- Das IP-Telegramm ist nicht von einer Source NAT-Regel (Masquerading, Source-NAT oder NETMAP) betroffen.
- Damit der Router A das initiale IP-Telegramm an Router B weiterleitet, wird eine Firewall Regel vlanX → vlanX benötigt.

VLAN

VLAN

Netzwerkdefinition unabhängig von der räumlichen Lage der Teilnehmer

VLAN (Virtuelles Local Area Network) teilt ein physikalisches Netzwerk in mehrere logische Netzwerke, die voneinander abgeschirmt sind. Hierbei werden Geräte zu logischen Gruppen zusammengefasst. Nur Teilnehmer des gleichen VLANs können sich untereinander adressieren. Da auch Multicast- und Broadcast-Telegramme nur innerhalb des jeweiligen VLANs weitergeleitet werden, wird von Broadcast-Domänen gesprochen.

Daraus ergibt sich als besonderer Vorteil von VLANs eine geringere Netzlast für die Teilnehmer bzw. Netzsegmente anderer VLANs.

Für die Kennung, welches Paket welchem VLAN zugeordnet ist, wird das Telegramm um 4 Byte erweitert, siehe VLAN-Tagging (Seite 594). Diese Erweiterung enthält neben der VLAN-ID auch Prioritätsinformationen.

Möglichkeiten der VLAN-Zuordnung

Es gibt verschiedene Möglichkeiten der Zuordnung zu VLANs:

- Port-basiertes VLAN
  Jedem Port eines Geräts wird eine VLAN-ID zugewiesen. Port-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN (Seite 701)".

VLAN-Zuordnung am Gerät

VLAN-Zuordnung am Gerät

Werkseitig sind folgende Zuordnungen eingestellt:

Tabelle 1-79 SCALANCE S615

P1 bis P4	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
P5	vlan2 Für den Zugriff vom externen Netz (WAN) zum Gerät

Tabelle 1-80 SCALANCE SC6x2-2C

P0.1	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
P0.2	vlan2 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät

Tabelle 1-81 SCALANCE SC6x6-2C

P0.1 - P0.4	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
P0.5 - P0.6	vlan2 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät

Die Zuordnung können Sie unter "Layer 2 > VLAN > Allgemein (Seite 698)" ändern.

Die VLANs sind in verschiedenen IP-Subnetzen. Damit diese miteinander kommunizieren können, muss im Gerät die entsprechende Route und die Firewall-Regel konfiguriert sein.

VLAN-Tagging

Erweiterung der Ethernet-Telegramme um vier Byte

Für CoS (Class of Service, Telegrammpriorisierung) und für VLAN (Virtuelles Netzwerk) wurde in der Norm IEEE 802.1Q die Erweiterung der Ethernet-Telegramme um das VLAN-Tag festgelegt.

Hinweis

Durch das VLAN-Tag erhöht sich die zulässige Gesamtlänge des Telegramms von 1518 auf 1522 Byte.
Es muss geprüft werden, ob die Endteilnehmer im Netz diese Länge / diesen Telegrammtyp verarbeiten können. Ist dies nicht der Fall, dürfen an diese Teilnehmer nur Telegramme mit der Standardlänge gesendet werden.

Die zusätzlichen 4 Bytes befinden sich im Header des Ethernet-Telegramms zwischen der Quelladresse und dem Ethernet-Typ-/Längenfeld:

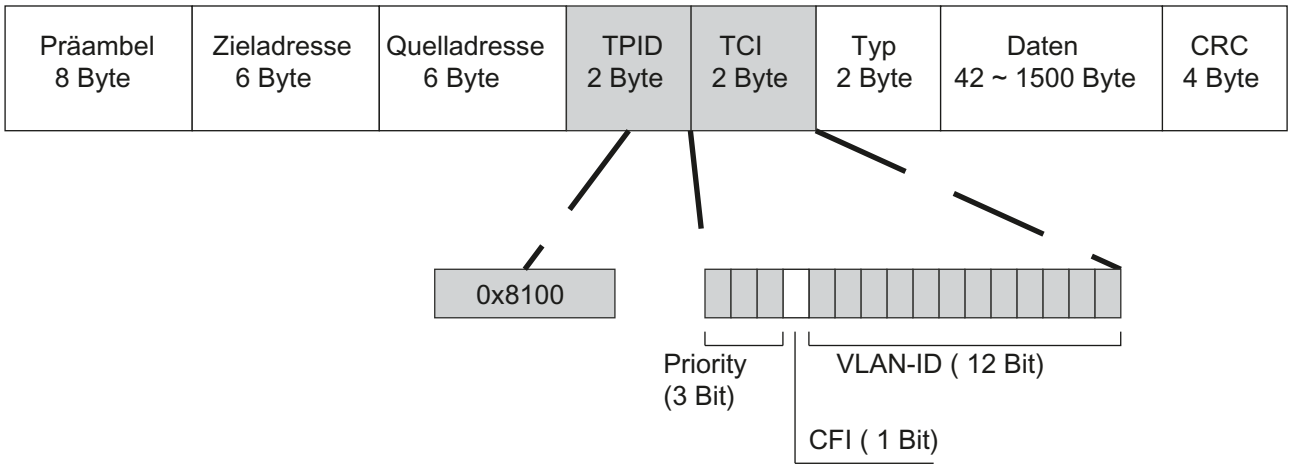

Bild 1-18 Aufbau des erweiterten Ethernet-Telegramms

Die zusätzlichen Bytes beinhalten den Tag Protocol Identifier (TPID) und die Tag Control Information (TCI).

Tag Protocol Identifier (TPID)

Die ersten 2 Bytes bilden den Tag Protocol Identifier (TPID) und sind fest mit 0x8100 belegt. Dieser Wert gibt an, dass das Datenpaket VLAN-Informationen oder Prioritätsangaben beinhaltet.

Tag Control Information (TCI)

Die 2 Bytes der Tag Control Information (TCI) beinhalten folgende Informationen:

CoS-Priorisierung

In dem getaggten Telegramm gibt es 3 Bits für die Priorität, die auch als Class of Service (CoS) bezeichnet werden, siehe auch IEEE 802.1Q.

CoS-Bits	Priorität	Art des Datenverkehrs
000	0 (niedrigste)	Background (Hintergrund)
001	1	Best Effort
010	2	Excellent Effort
011	3	Critical Applications (Kritische Anwendungen)
100	4	Video, < 100 ms Verzögerung (Latenz und Jitter)
101	5	Voice (Sprache), < 10 ms Verzögerung (Latenz und Jitter)
110	6	Internetwork Control
111	7 (höchste)	Network Control

Die Priorisierung der Datenpakete setzt eine Warteschlange in den Komponenten voraus, in der sie die Datenpakete mit der niedrigeren Priorität puffern können.

Das Gerät besitzt mehrere parallele Warteschlangen, in denen die verschieden priorisierten Telegramme abgearbeitet werden. Standardmäßig werden zuerst die Telegramme mit der höchsten Priorität abgearbeitet. Dieses Verfahren gewährleistet auch bei einem hohen Datenaufkommen, dass die Telegramme mit der höchsten Priorität auf jeden Fall gesendet werden.

Canonical Format Identifier (CFI)

Der CFI wird für die Kompatibilität zwischen Ethernet und Token Ring benötigt.
Die Werte haben folgende Bedeutung:

Wert	Bedeutung
0	Das Format der MAC-Adresse ist kanonisch. Bei kanonischer Darstellung der MAC-Adresse wird das niederwertigste Bit zuerst übertragen. Standardeinstellung für Ethernet-Switches.
1	Das Format der MAC-Adresse ist nicht kanonisch.

VLAN-ID

Im 12 Bit-Datenfeld können bis zu 4096 VLAN-IDs gebildet werden. Dabei gelten folgende Festlegungen:

VLAN-ID	Bedeutung
0	Das Telegramm beinhaltet nur Prioritätsinformation (Priority Tagged Frames) und keine gültige VLAN-Kennung.
1 - 4094	Gültige VLAN-Kennung, das Telegramm ist einem VLAN zugeordnet, es kann zusätzlich auch Prioritätsinformationen beinhalten.
4095	Reserviert

SNMP

SNMP

Einleitung

Mit Hilfe des Simple Network Management Protocol (SNMP) überwachen und steuern Sie Netzwerkkomponenten, z. B. Router oder Switches, von einer zentralen Station aus. SNMP regelt dabei die Kommunikation zwischen den überwachten Geräten und der Überwachungsstation.

Aufgaben von SNMP:

- Überwachung von Netzwerkkomponenten
- Fernsteuerung und Fernparametrierung von Netzwerkkomponenten
- Fehlererkennung und Fehlerbenachrichtigung

In den Versionen v1 und v2c verfügt SNMP über keine Sicherheitsmechanismen. Jeder Nutzer im Netzwerk kann mit geeigneter Software auf die Daten zugreifen und auch Parametrierungen verändern.

Für die einfache Steuerung von Zugriffsrechten ohne Sicherheitsaspekte werden Community-Strings verwendet.

Der Community-String wird zusammen mit der Anfrage übertragen. Wenn der Community-String korrekt ist, antwortet der SNMP-Agent und sendet die geforderten Daten. Wenn der Community-String nicht korrekt ist, verwirft der SNMP-Agent die Anfrage. Für Lese- und Schreibrechte definieren Sie verschiedene Community-Strings. Die Community-Strings werden in Klartext übertragen.

Standardwerte der Community-Strings:

- public
  besitzt nur Leserechte
- private
  besitzt Lese- und Schreibrechte

Hinweis

Da es sich bei den SNMP-Community Strings um einen Zugriffsschutz handelt, verwenden Sie nicht die Standardwerte "public" oder "private". Ändern Sie diese Werte nach der Erst-Inbetriebnahme.

Weitere einfache Schutzmechanismen auf Geräteebene:

- Allowed Host
  Dem überwachten System sind die IP-Adressen der überwachenden Systeme bekannt.
- Read Only
  Wenn Sie einem überwachten Gerät "Read Only" zuweisen, können Überwachungsstationen nur Daten auslesen, aber nicht ändern.

SNMP-Datenpakete sind nicht verschlüsselt und können einfach mitgelesen werden.

Die zentrale Station wird auch als Management-Station bezeichnet. Auf den zu überwachenden Geräten ist ein SNMP-Agent installiert, mit dem die Management-Station Daten austauscht.

Die Management-Station sendet Datenpakete folgenden Typs:

- GET
  Anfordern eines Datensatzes vom SNMP-Agent
- GETNEXT
  Ruft den nächsten Datensatz auf.
- GETBULK (verfügbar ab SNMPv2c)
  Fordert mehrere Datensätze auf einmal an, z. B. mehrere Zeilen einer Tabelle.
- SET
  Beinhaltet Parametrierungsdaten für das entsprechende Gerät.

Der SNMP-Agent sendet Datenpakete folgenden Typs:

- RESPONSE
  Der SNMP-Agent sendet die vom Manager angeforderten Daten zurück.
- TRAP
  Wenn ein bestimmtes Ereignis eintritt, sendet der SNMP-Agent eigenständig Traps.

SNMPv1/v2c/v3 verwenden UDP (User Datagram Protocol) und nutzen die UDP-Ports 161 und 162. Die Beschreibung der Daten erfolgt in einer Management Information Base (MIB).

SNMPv3

SNMPv3 führt gegenüber den Vorgängerversionen SNMPv1 und SNMPv2c ein umfangreicheres Sicherheitskonzept ein.

SNMPv3 unterstützt:

- Vollständig verschlüsselte Benutzerauthentifizierung
- Verschlüsselung des gesamten Datenverkehrs
- Zugriffskontrolle der MIB-Objekte auf Benutzer-/Gruppenebene

Mit der Einführung von SNMPv3 können Sie Benutzerkonfigurationen nicht mehr ohne Weiteres auf andere Geräte übertragen, z. B. indem Sie eine Konfigurationsdatei laden oder den C-PLUG austauschen.

Das SNMPv3-Protokoll verwendet gemäß des Standards eine eindeutige SNMP-Engine-ID als internen Bezeichner für einen SNMP-Agenten. Diese ID muss im Netzwerk eindeutig sein. Sie wird verwendet, um die Zugangsdaten von SNMPv3-Benutzern zu authentifizieren und zu verschlüsseln.

Abhängig davon, ob Sie die Funktion "SNMPv3 Benutzermigration" aktiviert oder deaktiviert haben, wird die SNMP-Engine-ID unterschiedlich generiert.

Einschränkung bei der Verwendung der Funktion

Verwenden Sie die Funktion "SNMPv3 Benutzermigration" nur, um im Ersatzteilfall Ihre konfigurierten SNMPv3-Benutzer auf ein Ersatzgerät zu übertragen.
Verwenden Sie die Funktion nicht, um konfigurierte SNMPv3-Benutzer auf mehrere Geräte zu übertragen. Wenn Sie eine Konfiguration mit angelegten SNMPv3-Benutzern in mehrere Geräte laden, verwenden diese Geräte hierdurch die gleiche SNMP-Engine-ID. Wenn Sie

diese Geräte im gleichen Netzwerk verwenden, widerspricht Ihre Konfiguration dem SNMP-Standard.

Kompatibilität mit Vorgängerprodukten

Sie können SNMPv3-Benutzer nur auf ein anderes Gerät übertragen, wenn Sie die Benutzer als migrierbare Benutzer erstellt haben. Um einen migrierbaren Benutzer zu erstellen, muss die Funktion "SNMPv3 Benutzermigration" aktiviert sein, wenn Sie den Benutzer erstellen.

Security-Funktionen

Benutzerverwaltung (WBM, CLI)

Übersicht zur Benutzerverwaltung

Der Zugriff auf das Gerät wird durch konfigurierbare Benutzereinstellungen verwaltet. Richten Sie Benutzer mit jeweils einem Passwort zur Authentifizierung ein. Weisen Sie den Benutzern eine Rolle mit entsprechenden Rechten zu.

Die Authentifizierung von Benutzern kann entweder von dem Gerät lokal oder von einem externen RADIUS-Server durchgeführt werden. Wie die Authentifizierung erfolgen soll, konfigurieren Sie auf der Seite "Security > AAA > Allgemein".

Lokale Anmeldung

Die lokale Anmeldung von Benutzern durch das Gerät läuft wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät prüft, ob ein Eintrag für den Benutzer vorhanden ist:
   → Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet.
   → Wenn kein entsprechender Eintrag existiert, wird dem Benutzer der Zugriff verweigert.

Anmeldung über einen externen RADIUS-Server

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Authentifizierung und Autorisierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können.

Anhängig davon, welchen RADIUS-Autorisierungsmodus Sie auf der Seite "Security > AAA > RADIUS-Client" eingestellt haben, wertet das Gerät unterschiedliche Informationen des RADIUS-Servers aus.

RADIUS-Autorisierungsmodus "Standard"

Wenn Sie den RADIUS-Autorisierungsmodus "Standard" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück:
     → Der Benutzer wird mit Administratorrechten angemeldet.
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt einen anderen oder gar keinen Wert für das Attribut "Service Type" an das Gerät zurück:
     → Der Benutzer wird mit Leserechten angemeldet.
   - Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
     → Dem Benutzer wird der Zugriff verweigert.

RADIUS-Autorisierungsmodus "Herstellerspezifisch"

Voraussetzung

Für den RADIUS-Autorisierungsmodus "Herstellerspezifisch" ist am RADIUS-Server Folgendes einzustellen:

- Herstellercode: 4196
- Attributnummer: 1
- Attributformat: Zeichenfolge (Gruppenname)

Vorgehen

Wenn Sie den RADIUS-Autorisierungsmodus "Herstellerspezifisch" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   Fall A: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt die dem Benutzer zugeordnete Gruppe an das Gerät zurück.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird mit den Rechten der zugeordneten Gruppe angemeldet.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird der Rolle mit den größeren Rechten zugeordnet und mit diesen Rechten angemeldet.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle angemeldet, die mit seinem Benutzeraccount verknüpft ist.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall B: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung, gibt jedoch keine Gruppe an das Gerät zurück:
   - Der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der verknüpften Rolle angemeldet.
   - Der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall C: Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
   - Dem Benutzer wird der Zugriff verweigert.

Firewall

Firewall

Zu den Security-Funktionen des Geräts gehört eine Stateful Inspection Firewall. Dabei handelt es sich um eine Methode der Paketfilterung bzw. Paketüberprüfung.

Die IP-Pakete werden anhand von Firewall-Regeln geprüft, in denen Folgendes festgelegt wird:

- Die erlaubten Protokolle
- IP-Adressen und Ports der erlaubten Quellen
- IP-Adressen und Ports der erlaubten Ziele

Wenn ein IP-Paket den festgelegten Parametern entspricht, dann darf es die Firewall passieren. Zusätzlich wird festgelegt, wie mit IP-Paketen verfahren wird, welche die Firewall nicht passieren dürfen.

Einfache Paketfiltertechniken benötigen pro Verbindung zwei Firewall-Regeln.

- Eine Regel für Anfragerichtung von der Quelle zum Ziel.
- Eine zweite Regel für die Antwortrichtung vom Ziel zur Quelle

Stateful Inspection Firewall

Sie müssen nur eine Firewall-Regel für die Anfragerichtung von der Quelle zum Ziel festlegen. Die zweite Regel wird implizit hinzugefügt. Der Paketfilter merkt sich, wenn z. B. Rechner "A" mit Rechner "B" kommuniziert und erlaubt nur dann Antworten darauf. Eine Anfrage von Rechner "B" ist somit ohne vorherige Anforderung durch Rechner "A" nicht möglich.

Die Firewall konfigurieren Sie unter "Security > Firewall".

Hinweis

IP-Pakete über Layer 2 (innerhalb des gleichen VLANs)

Wenn die IP-Pakete von dem Gerät über einen Switch-Port (Layer 2) gesendet werden, werden diese IP-Pakete nicht anhand von Firewall-Regeln geprüft. Die Firewall hat keine Wirkung auf Pakete, die auf der Layer 2-Ebene weitergeleitet werden.

Kommunikationsrichtungen

von	nach	Bedeutung
vlan x	vlan x	Zugriff vom IP-Subnetz vlan x auf das IP-Subnetz vlan x. Beispiel: vlan1 (INT) → vlan2 (EXT) Zugriff vom lokalen IP-Subnetz auf das externe IP-Subnetz.
	ppp2	Zugriff vom IP-Subnetz auf die WAN-Schnittstelle des Geräts.
	Gerät	Zugriff vom IP-Subnetz auf das Gerät.
	SINEMA RC	Zugriff vom IP-Subnetz auf die SINEMA RC-Verbindung.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom IP-Subnetz auf die VPN-Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.

von	nach	Bedeutung
Gerät	vlan x	Zugriff vom Gerät auf das IP-Subnetz.
	ppp2	Zugriff vom Gerät auf die WAN-Schnittstelle des Geräts.
	SINEMA RC	Zugriff vom Gerät auf die SINEMA RC-Verbindung.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom Gerät auf die Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.
SINEMA RC	vlan x	Zugriff von SINEMA RC-Verbindungen auf das IP-Subnetz.
	ppp2	Zugriff vom IP-Subnetz auf die WAN-Schnittstelle des Geräts.
	Gerät	Zugriff von SINEMA RC-Verbindungen auf das Gerät.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom SINEMA RC-Server auf die VPN-Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.
IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	vlan x	Zugriff über VPN-Tunnelpartner auf das IP-Subnetz.
	ppp2	Zugriff vom IP-Subnetz auf die WAN-Schnittstelle des Geräts.
	Gerät	Zugriff über VPN-Tunnelpartner auf das Gerät.
	SINEMA RC	Zugriff über VPN-Tunnelpartner auf die SINEMA RC-Verbindung.
ppp0/usb	vlan x	Zugriff von der Mobilfunk-Schnittstelle auf das IP-Subnetz.
	Device	Zugriff von der Mobilfunk-Schnittstelle auf das Gerät.
	SINEMA RC	Zugriff von der Mobilfunk-Schnittstelle auf die SINEMA RC-Verbindung.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff von der Mobilfunk-Schnittstelle auf die VPN-Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.

Voreinstellung der Firewall

Werkseinstellung der Firewall

Dienst	Zugriff	
	von intern (vlan1) auf das Gerät	von extern (vlan2) auf das Gerät
HTTP	ja	nein
HTTPS	ja	nein
TFTP	ja	nein
DNS	ja	nein

Dienst	Zugriff	
	von intern (vlan1) auf das Gerät	von extern (vlan2) auf das Gerät
SNMP	ja	nein
Telnet	ja	nein
IPsec VPN	nein	ja
SSH	ja	nein
DHCP	ja	ja (für die Funktion DHCP-Client)
Ping	ja	nein

NAT

NAT (Network Address Translation) ist eine Methode IP-Adressen in Datenpaketen umzuschreiben. Damit können zwei verschiedene Netze (intern und extern) miteinander verbunden werden.

Man unterscheidet zwischen Source-NAT, bei dem die Quell-IP-Adresse umgeschrieben wird und Destination-NAT, bei dem die Ziel-IP-Adresse umgeschrieben wird.

Weitere Informationen zu NAT-Szenarien, die mit dem Gerät realisiert werden können, finden Sie unter folgender Adresse:

IP-Masquerading

IP-Masquerading ist ein vereinfachtes Source-NAT. Dabei wird bei jedem ausgehenden Datenpaket, das über diese Schnittstelle gesendet wird, die Quell-IP-Adresse durch die IP-Adresse der Schnittstelle ersetzt. Das angepasste Datenpaket wird an die Ziel-IP-Adresse gesendet. Für den Ziel-Host sieht es so aus, als kämen die Anfragen immer von dem gleichen Absender. Die internen Teilnehmer sind aus dem externen Netz nicht direkt erreichbar. Mithilfe von NAPT lassen sich die Dienste der internen Teilnehmer über die externe IP-Adresse des Geräts erreichbar machen.

IP-Masquerading kann benutzt werden, wenn die internen IP-Adressen extern nicht weitergeleitet werden können oder sollen, z. B. weil die interne Netzstruktur verborgen werden soll.

Masquerading konfigurieren Sie unter "Layer 3" > "NAT" > "IP-Masquerading (Seite 713)".

NAPT

NAPT (Network Address and Port Translation) ist eine Form des Destination-NAT und wird oft auch als Portweiterleitung (Port Forwarding) bezeichnet. Damit lassen sich Dienste der internen Teilnehmer von außen erreichbar machen, die durch IP-Masquerading oder Source-NAT versteckt sind.

Umgesetzt werden eingehende Datenpakete, die vom externen Netz kommen und an eine externe IP-Adresse des Geräts (Ziel-IP-Adresse) gerichtet sind. Die Ziel-IP-Adresse wird mit der IP-Adresse des internen Teilnehmers ersetzt. Zusätzlich zur Adressumsetzung ist auch eine Port-Umsetzung möglich.

Es gibt folgende Möglichkeiten der Port-Umsetzung:

von	zu	Verhalten
einem einzigen Port	dem gleichen Port	Wenn die Ports gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.
einem einzigen Port	einem einzigen Port	Die Telegramme werden auf den Port umgesetzt.
einem Port-Bereich	einem einzigen Port	Die Telegramme aus dem Port-Bereich werden auf den gleichen Port umgesetzt (n:1).
einem Port-Bereich	dem gleichen Port-Bereich	Wenn die Port-Bereiche gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.
einem Port-Bereich	einem anderen Port-Bereich	Die Telegramme werden auf einen beliebigen freien Port aus dem Zielbereich umgesetzt. Bei einzelnen Verbindungen wird meist auf den ersten Port im Zielbereich umgesetzt. Bei gleichzeitigen Verbindungen wird mittels einer Reih-um-Methode (round robin) auf einen freien Port im Zielbereich umgesetzt.
einem einzigen Port	einem Port-Bereich	Die Telegramme werden auf einen beliebigen freien Port aus dem Zielbereich umgesetzt. Bei einzelnen Verbindungen wird meist auf den ersten Port im Zielbereich umgesetzt. Bei gleichzeitigen Verbindungen wird mittels einer Reih-um-Methode (round robin) auf einen freien Port im Zielbereich umgesetzt.

Port Forwarding kann benutzt werden, um externen Teilnehmern den Zugriff auf bestimmte Dienste des internen Netzes zu ermöglichen, z. B. FTP, HTTP.

NAPT konfigurieren Sie unter "Layer 3" > "NAT" > "NAPT (Seite 714)".

Source-NAT

Wie beim Masquerading wird beim Source-NAT die Quelladresse umgeschrieben. Zusätzlich können die ausgehenden Datenpakete beschränkt werden. Dazu gehören Beschränkungen auf bestimme IP-Adressen oder IP-Adressbereiche und Beschränkungen auf bestimmte Schnittstellen.

Source-NAT kann benutzt werden, wenn die internen IP-Adressen extern nicht weitergeleitet werden können oder sollen, z. B. weil ein privater IP-Adressbereich wie 192.168.x.x benutzt wird.

Source-NAT konfigurieren Sie unter "Layer 3" > "NAT" > "Source NAT (Seite 716)".

NETMAP

Mit NETMAP ist es möglich, komplette Subnetze auf ein anderes Subnetz umzusetzen. Bei dieser Umsetzung wird der Subnetzanteil der IP-Adresse geändert und der Hostanteil bleibt bestehen. Für die Umsetzung wird bei NETMAP nur eine Regel benötigt. NETMAP kann sowohl die Quell-IP-Adresse als auch die Ziel-IP-Adresse umsetzen. Um die Umsetzung mit Destination-NAT und Source-NAT durchzuführen, wären viele Regeln notwendig. NETMAP kann auch auf VPN-Verbindungen angewendet werden.

NETMAP konfigurieren Sie unter "Layer 3" > "NAT" > "NETMAP (Seite 717)".

NAT und Firewall

Firewall und NAT-Router unterstützen den Mechanismus "Stateful Inspection". Wenn der IP-Datenverkehr von intern nach extern freigegeben ist, können interne Teilnehmer eine Kommunikationsverbindung in das externe Netz initiieren.

Die Antworttelegramme aus dem externen Netz können den NAT-Router und die Firewall passieren, ohne dass deren Adressen in der Firewall-Regel und der NAT-Adressumsetzung zusätzlich aufgenommen werden müssen. Telegramme, die keine Antwort auf eine Anfrage aus dem internen Netz sind, werden ohne zutreffende Firewallregel verworfen.

NAT-Umsetzung und Firewall-Regeln

Ein Beispiel für NAT-Umsetzungen finden Sie auf den auf den Internetseiten des Siemens Industry Online Support:

Link: (https://support.industry.siemens.com/cs/ww/de/view/109744660)

Zertifikate

Zertifikatstypen

Das Gerät verwendet verschiedene Zertifikate, um die verschiedenen Teilnehmer zu authentifizieren.

Zertifikat		Wird verwendet in ...
CA-Zertifikat	Das CA-Zertifikat ist ein durch eine Zertifizierungsstelle, die so genannte "Certificate Authority", ausgestelltes Zertifikat, von denen die Server-, Geräte- und Gegenstellenzertifikate abgeleitet werden. Damit ein Zertifikat abgeleitet werden kann, besitzt das CA-Zertifikat einen privaten Schlüssel, der durch die Zertifizierungsstelle signiert wurde. Der Schlüsselaustausch zwischen dem Gerät und dem VPN-Gateway der Gegenstelle erfolgt automatisch beim Aufbau der Verbindung. Es ist kein manueller Austausch von Schlüsseldateien notwendig.	IPsec VPN
Serverzertifikat	Serverzertifikate werden zum Aufbau einer gesicherten Kommunikation (z. B. HTTPS, VPN ...) zwischen Gerät und einem weiteren Netzwerkteilnehmer benötigt. Bei dem Serverzertifikat handelt es sich um ein verschlüsseltes SSL-Zertifikat. Das Serverzertifikat wird von der ältesten gültigen CA abgeleitet, auch wenn dieses "außer Dienst" ist. Entscheidend ist das Gültigkeitsdatum der CA.	SINEMA RC
Gerätezertifikat	Zertifikate mit dem privaten Schlüssel (Key file), mit denen sich das Gerät ausweist.	IPsec VPN
Gegenstellenzertifikat	Zertifikate, mit denen sich das VPN-Gateway der Gegenstelle bei dem Gerät authentifiziert.	IPsec VPN

Dateitypen

Dateityp	Beschreibung
*.crt	Datei, die das Zertifikat enthält.
*.p12	Bei der PKCS12-Zertifikatsdatei wird der private Schlüssel mit dem zugehörigen Zertifikat passwortgeschützt gespeichert. Die CA erstellt für beide Gegenstellen einer VPN-Verbindung je eine Zertifikatsdatei (PKCS12) mit der Dateiendung ".p12". Diese Zertifikatsdatei enthält den öffentlichen und privaten Schlüssel der eigenen Station, das signierte Zertifikat der CA und den öffentlichen Schlüssel der CA.
*.pem	Zertifikat und Schlüssel als Base64-kodierten ASCII-Text.

VPN

Übersicht

Das Gerät unterstützt folgende VPN-Systeme

- IPsec VPN (nur für SCALANCE S615 und SCALANCE SC64x-2C)
- OpenVPN (nur für SCALANCE S615)

IPsec VPN

Die IPsec-Verbindungen konfigurieren Sie unter "Security" > "IPsec VPN"

Bei IPsec VPN werden die Telegramme im Tunnel-Modus übertragen. Damit das Gerät einen VPN-Tunnel aufbauen kann, muss das entfernte Netz über ein VPN-Gateway als Gegenstation verfügen.

Für die VPN-Verbindungen unterscheidet das Gerät zwei Modi:

- Roadwarrior-Modus
  In diesem Modus wird entweder die Adresse der Gegenstelle fest vorgegeben oder ein IP-Bereich eingetragen, aus dem Verbindungen entgegengenommen werden. Die erreichbaren Remote-Subnetze lernt das Gerät von der Gegenstelle.
- Standard-Modus
  In diesem Modus ist die Adresse der Gegenstelle oder das Remote-Subnetz fest eingetragen. Das Gerät kann entweder als VPN-Client die VPN-Verbindung aktiv aufbauen, oder passiv auf den Verbindungsaufbau durch die Gegenstelle warten.

Das IPsec-Verfahren

Das Gerät verwendet für den VPN-Tunnel das IPsec-Verfahren im Tunnelmodus. Dabei werden die zu übertragenden Telegramme vollkommen verschlüsselt und mit einem neuen Header versehen, bevor sie zum VPN-Gateway der Gegenstelle gesendet werden. Von der Gegenstelle werden die empfangenen Telegramme entschlüsselt und an den Empfänger weitergeleitet.

Zum Absichern verwendet das IPsec-Verfahren verschiedene Protokolle:

- Der IP-Authentication-Header (**AH**) wickelt die Authentifizierung und Identifizierung der Quelle ab.
- Die Encapsulation Security Payload (**ESP**) verschlüsselt die Daten.
- Die Security Association (**SA**) enthält die Festlegungen, die zwischen den Partner ausgehandelt wurden, z. B. über die Lebensdauer des Schlüssels, den Verschlüsselungsalgorithmus, den Zeitraum für eine neue Authentifizierung etc.
- Das Internet Key Exchange (**IKE**) ist ein Schlüsselaustauschverfahren. Der Schlüsselaustausch erfolgt in zwei Phasen:
  - Phase 1
    In dieser Phase sind noch keine Sicherheitsdienste wie Verschlüsselung, Authentifizierung und Integritätsprüfung verfügbar, da die notwendigen Schlüssel und die IPsec-SA noch nicht erstellt wurden. Phase 1 dient zum Aufbau eines sicheren VPN-Tunnels für Phase 2. Dafür verhandeln die Kommunikationspartner eine ISAKMP Security Association (ISAKMP-SA), welche die notwendigen Sicherheitsdienste (verwendete Algorithmen, Authentifizierungsmethoden) definiert. Damit werden die weiteren Nachrichten und Phase 2 abgesichert.
  - Phase 2
    Phase 2 dient zur Aushandlung der benötigten IPsec-SA. Ähnlich wie bei Phase 1 wird durch das wechselseitige Anbieten eine Einigung über die Authentifizierungsmethoden, die Algorithmen und die Verschlüsselungsverfahren getroffen, um die IP-Pakete mit IPsec-AH und IPsec-ESP zu schützen.
    Geschützt wird der Nachrichtenaustausch über die ISAKMP-SA, die in Phase 1 vereinbart wurde. Durch die in Phase 1 ausgehandelte ISAKMP-SA ist die Identität der Teilnehmer sowie das Verfahren zur Integritätsprüfung bereits gegeben.

Authentifizierungsverfahren

- CA-Zertifikat, Geräte- und Gegenstellenzertifikat (digitale Signaturen)
  Die Verwendung von Zertifikaten ist ein asymmetrisches Kryptosystem, wobei jeder Teilnehmer (Gerät) über ein Schlüsselpaar verfügt. Jeder Teilnehmer besitzt einen geheimen, privaten Schlüssel und einen öffentlichen Schlüssel der Gegenstelle. Der private Schlüssel ermöglicht es, sich zu authentisieren und digitale Signaturen zu erzeugen.
- Preshared Key
  Die Verwendung eines Preshared Key ist ein symmetrisches Kryptosystem. Jeder Teilnehmer besitzt nur einen geheimen Schlüssel für die Ent- und Verschlüsselung von Datenpaketen. Die Authentifizierung erfolgt über ein gemeinsames Passwort.

Lokale-ID und Remote-ID

Die Lokale-ID und die Remote-ID werden vom IPsec genutzt, um beim Aufbau der VPN-Verbindung die Gegenstellen (VPN-Endpunkt) eindeutig zu identifizieren.

Verschlüsselungsverfahren

Folgende Verschlüsselungsverfahren werden unterstützt. Die Auswahl ist abhängig von der Phase und vom Schlüsselaustauschverfahren (IKE):

	Phase 1		Phase 2	
	IKEv1	IKEv2	IKEv1	IKEv2
3DES	x	x	x	x
AES128 CBC	x	x	x	x
AES192 CBC	x	x	x	x
AES256 CBC	x	x	x	x
AES128 CTR	-	x	x	x
AES192 CTR	-	x	x	x
AES256 CTR	-	x	x	x
AES128 CCM 16	-	x	x	x
AES192 CCM 16	-	x	x	x
AES256 CCM 16	-	x	x	x
AES128 GCM 16	-	x	x	x
AES192 GCM 16	-	x	x	x
AES256 GCM 16	-	x	x	x

x: wird unterstützt

-: wird nicht unterstützt

Default-Chiffre

Beim Verbindungsaufbau kann eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt werden. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Die Kombinationen sind abhängig von der Phase und vom Schlüsselaustauschverfahren (IKE).

Kombination			Phase 1		Phase 2	
Verschlüsselung	Authentifizierung	Schlüsselableitung	IKEv1	IKEv2	IKEv1	IKEv2
AES128	SHA1	DH Group 14	x	x	x	x
AES256	SHA512	DH Group 16	x	x	x	x
AES128 CCM 16	SHA256	DH Group 14	-	x	x	x
AES256 CCM 16	SHA512	DH Group 16	-	x	x	x
AES128	SHA1	none	-	-	x	x
AES256	SHA512	none	-	-	x	x
AES128 CCM 16	SHA256	none	-	-	x	x
AES256 CCM 16	SHA512	none	-	-	x	x

x: Kombination ist Teil der Default-Chiffre

-: Kombination ist nicht Teil der Default-Chiffre

none: Für die Phase 2 werden keine separaten Schlüssel ausgetauscht. Damit ist Perfect Forward Secrecy (PFS) deaktiviert.

Anforderungen an die VPN-Gegenstelle

Die VPN-Gegenstelle muss IPsec mit folgender Konfiguration unterstützen, um erfolgreich eine IPsec-Verbindung aufzubauen:

- Authentifizierung über Gegenstellenzertifikate, CA-Zertifikate oder Pre-Shared Key
- IKEv1 oder IKEv2
- Unterstützung von mindestens einer der folgenden DH-Gruppen: Diffie-Hellman Gruppe 1, 2, 5 und 14 - 18
- 3DES- oder AES-Verschlüsselung
- MD5, SHA1, SHA256, SHA384 oder SHA512
- Tunnel-Modus

Wenn sich die VPN-Gegenstelle hinter einem NAT-Router befindet, dann muss die Gegenstelle NAT-T unterstützen. Oder aber der NAT-Router muss das IPsec-Protokoll kennen (IPsec/VPN Passthrough).

NAT-Traversal (NAT-T)

Eventuell befindet sich zwischen dem Gerät und dem VPN-Gateway des entfernten Netzes ein NAT-Router. Nicht alle NAT-Router lassen IPsec-Telegramme passieren. Daher kann es erforderlich sein, die IPsec-Telegramme in UDP-Pakete einzukapseln, um den NAT-Router passieren zu können.

Dead Peer Detection

Voraussetzung ist, dass die VPN-Gegenstelle DPD unterstützt. DPD prüft, ob die Verbindung noch störungsfrei arbeitet oder ob es eine Unterbrechung auf der Strecke gab. Ohne DPD muss je nach Konfiguration bis zum Ablauf der SA-Lebensdauer gewartet oder die Verbindung manuell neu initiiert werden. Um zu prüfen, ob die IPsec-Verbindung noch störungsfrei arbeitet, sendet das Gerät selber DPD-Anfragen zur VPN-Gegenstelle. Wenn die VPN-Gegenstelle nicht antwortet, wird nach Ablauf einer bestimmten Zeitspanne die Verbindung zur VPN-Gegenstelle für ungültig erklärt. Die Einstellungen für DPD konfigurieren Sie bei der Phase 1.

OpenVPN

Mit OpenVPN lassen sich virtuelle private Netzwerke (VPN) aufbauen. Das Gerät kann als OpenVPN-Client eine VPN-Verbindung zu einem entfernten Netzwerk aufbauen.

Den OpenVPN-Client konfigurieren Sie unter "Security" > "OpenVPN-Client".

Die VPN-Verbindung wird über virtuelle Gerätetreiber hergestellt, dem TAP- und TUN-Device. Dabei werden virtuelle Netzwerkschnittstellen angelegt, die wie eine physische Schnittstelle des Geräts wirken und den Endpunkt des VPN-Tunnels darstellen.

Das Gerät unterstützt Folgendes:

- TUN-Device: Routing-Modus
  Die LAN-Schnittstelle und die virtuelle Netzwerkschnittstelle befinden sich in verschiedenen IP-Subnetzen. Der virtuellen Tunnelschnittstelle wird vom OpenVPN-Server eine virtuelle IP-Adresse aus einem erdachten Subnetz zugewiesen. Die IP-Pakete (Layer 3) werden zwischen der virtuellen Tunnelschnittstelle und der LAN-Schnittstelle geroutet.

Authentifizierungsverfahren

- Zertifikate: CA-Zertifikat und Gerätezertifikat
  Die Verwendung von Zertifikaten ist ein asymmetrisches Kryptosystem. Jeder Teilnehmer (Gerät) besitzt einen geheimen, privaten Schlüssel und einen öffentlichen Schlüssel der Gegenstelle. Der private Schlüssel ermöglicht es, sich zu authentisieren und digitale Signaturen zu erzeugen.
- Benutzername/Passwort
  Der Zugang wird über einen Benutzernamen und ein Passwort beschränkt.

Verschlüsselungsverfahren

Das Gerät unterstützt dabei die folgenden Verfahren:

- BF CBC
- AES128 CBC
- AES192 CBC
- AES256 CBC
- DES EDE3

VPN-Verbindungsaufbau

Das Gerät unterstützt folgende Möglichkeiten, um eine VPN-Verbindung aufzubauen.

- OpenVPN: Security > OpenVPN > Verbindungen
- IPsec VPN: Security > IPsec VPN > Verbindungen (Seite 742)
- SINEMA RC: System > SINEMA RC (Seite 691)

Möglichkeiten	Verwendung			Beschreibung
	OpenVPN	IPsec VPN	SINEMA RC [1]	
Starten	x	x	-	Das Gerät ist "aktiv", d. h. es wird versucht, zu einer Gegenstelle eine Verbindung herzustellen. Die Adressierung der Gegenstelle erfolgt über deren projektierte WAN-IP-Adresse oder den projektierten FQDN.
Warten	-	x	-	Das Gerät ist "passiv", d. h. es wird gewartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
Auf Anforderung	-	x	-	Das Gerät versucht bei Bedarf eine Verbindung zu einer Gegenstelle herzustellen. Die Annahme von Anfragen zum VPN-Verbindungsaufbau ist ebenfalls möglich. Für die projektierten lokalen und entfernten Subnetze wird in der Routing-Tabelle ein Eintrag angelegt. Wenn ein Teilnehmer aus einem der entsprechenden Netze versucht Datenpakete über den VPN-Tunnel zu senden, wird die VPN-Verbindung aufgebaut. Der einstellbare Timeout bewirkt, dass nach dieser Zeit ohne weitere Datenpakete der VPN-Tunnel wieder abgebaut wird.

Möglichkeiten	Verwendung			Beschreibung
	OpenVPN	IPsec VPN	SINEMA RC [1]	
Bei DI starten	x	x	-	Der Verbindungsaufbau wird über den digitalen Eingang (DI) gesteuert.
Auf DI warten	-	x	-	
Digitaler Eingang	-	-	x	
Auto	-	-	x	Das Gerät übernimmt die Einstellungen des SINEMA RC Server. Die Einstellungen auf dem SINEMA RC Server konfigurieren Sie unter "Fernverbindungen > Geräte". Weiterführende Informationen hierzu finden Sie in der Betriebsanleitung "SINEMA RC Server".
Permanent	-	-	x	Das Gerät baut eine VPN-Verbindung zum SINEMA RC-Server. Der VPN-Tunnel wird permanent aufrecht erhalten.

[1] Für SCALANCE S615: KEY-PLUG SINEMA REMOTE CONNECT notwendig

Digitaler Eingang (DI)

Der Aufbau des VPN-Tunnels kann auch über den digitalen Eingang gesteuert werden, z. B. über einen Taster. Wenn der Taster schließt, liegt am digitalen Eingang Spannung an und die LED vom digitalen Eingang leuchtet auf. Die aufleuchtende LED signalisiert, dass das Signal 1 (TRUE / HIGH) anliegt. Das Signal 1 löst auf dem Gerät ein Ereignis aus, mit dem der Aufbau des VPN-Tunnels gesteuert wird. Informationen zum Anschließen und zur maximalen Strombelastbarkeit finden Sie in der Betriebsanleitung der Geräte.

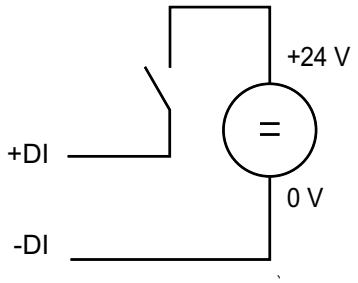

Voraussetzung

- Unter "System > Ereignisse > Konfiguration" ist beim Ereignis "Digitaler Eingang" "VPN-Tunnel" aktiviert.
  Wenn diese Einstellung nicht aktiviert ist, wird das Ereignis nicht an die VPN-Verbindung weitergegeben.

Möglichkeiten

Das Gerät unterstützt folgende Möglichkeiten zur Steuerung des VPN-Tunnels über den digitalen Eingang:

- Bei DI starten
  Wenn das Ereignis "Digitaler Eingang" eintritt, wird das Gerät "aktiv". Das Gerät versucht, zu einer Gegenstelle eine VPN-Verbindung (OpenVPN, IPsec) herzustellen.
- Auf DI warten
  Wenn das Ereignis "Digitaler Eingang" eintritt, ist das Gerät "passiv". Das Gerät wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
- Digitaler Eingang
  Die Einstellungen des SINEMA RC-Servers werden ignoriert. Wenn das Ereignis "Digital In" eintritt, wird das Gerät "aktiv". Das Gerät versucht, zum SINEMA RC-Server eine VPN-Verbindung herzustellen.

Benachrichtigungsmöglichkeiten

Wenn sich der Status des digitalen Eingangs oder eines VPN-Tunnels (IPsec, OpenVPN, SINEMA RC) ändert, bietet das Gerät auf der Seite "Ereignisse" mehrere Möglichkeiten der Benachrichtigung.

Art der Benachrichtigung	Digital In	VPN Tunnel	Verhalten bei Statusänderung
E-Mail	x	x	Das Gerät sendet eine E-Mail. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Voraussetzung: • Ein SMTP-Server ist eingerichtet. • Unter "System > SMTP-Client" ist die Funktion aktiviert, ein Empfänger und die IP-Adresse des SMTP-Servers konfiguriert.
Trap	x	x	Das Gerät löst einen SNMP-Trap aus. Voraussetzung: • Unter "System > Konfiguration" ist "SNMPv1 Traps" aktiviert. • Unter "System > Konfiguration > Traps" ist ein Empfänger konfiguriert, an den das Gerät die SNMP-Traps sendet.
Log-Tabelle	x	x	Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle. Der Inhalt der Ereignisprotokoll-Tabelle wird unter "Information > Log Table" angezeigt.
Syslog	x	x	Das Gerät schreibt einen Eintrag auf den Syslog-Server. Voraussetzung: • Ein Syslog-Server ist eingerichtet. • Unter "System > Syslog-Client" ist die Funktion aktiviert und die IP-Adresse des Syslog-Servers konfiguriert.
Fehler-LED	x	-	Die Fehler-LED am Gerät leuchtet auf.

Art der Benachrichtigung	Digital In	VPN Tunnel	Verhalten bei Statusänderung
Digitaler Eingang	x	x	Steuert den digitalen Ausgang an oder signalisiert die Zustandsänderung mit der LED "DO". An dem digitalen Ausgang kann ein Verbraucher angeschlossen werden. Informationen zum Anschließen finden Sie in der Betriebsanleitung der Geräte. Der Verbraucher signalisiert eine Statusänderung. 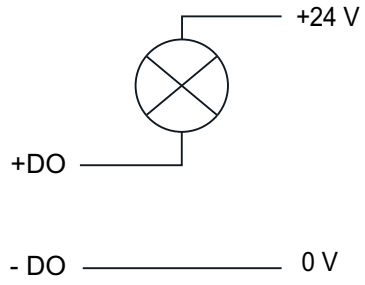 **Hinweis** Über CLI und über SNMP können Sie den digitalen Ausgang direkt ansteuern. Im WBM und CLI können Sie bei den "Events" die Verwendung des digitalen Ausgangs projektieren. Steuern Sie den digitalen Ausgang nicht direkt an, wenn Sie diesen im WBM und CLI verwenden.
Status aus der MIB-Variable auslesen	x	-	Über die private MIB-Variable snMspsDigitalInputLevel können Sie den Status des digitalen Eingangs auslesen: • OID der privaten MIB-Variable snMspsDigitalInputLevel: `iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).siemens(4329).industrialComProducts(20).iComPlatforms(1).simaticNet(1).snMsps(1).snMspsCommon(1).snMspsDigitalIO(39).snMspsDigitalIOObjects(1).snMspsDigitalInputTable(2).snMspsDigitalInputEntry(1).snMspsDigitalInputLevel(6)` • Werte der MIB-Variable – 1: Signal 0 am digitalen Eingang (DI) – 2: Signal 1 am digitalen Eingang (DI)

Besonderheiten der Benutzerverwaltung für Security-Funktionen

Zugriff auf Security-Funktionen

Auf Security-Funktionen können Sie nur nach Aktivierung des Projektschutzes und nach erfolgreicher Anmeldung eines Benutzers zugreifen. Der angemeldete Benutzer muss zudem die erforderlichen Engineering- und Runtime-Rechte besitzen.

Grundlegende Informationen zur Benutzerverwaltung in STEP 7 finden Sie im Abschnitt "Benutzerverwaltung verwenden" des Informationssystems.

Engineering-Rechte für Security-Geräte

Diese Engineering-Rechte berechtigen zur Diagnose und Projektierung von Security-Geräten:

- Security: Security-Module mit Leserechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Geräte anzeigen.
- Security: Security-Module mit Schreibrechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Geräte projektieren und die Diagnose der Security-Geräte durchführen.

Rollen mit Runtime-Rechten für SCALANCE S-600

Einigen systemdefinierten Rollen mit dem Präfix "NET" sind standardmäßig Runtime-Rechte für SCALANCE S-600 zugeordnet. Runtime-Rechte berechtigen zur Ausführung bestimmter Aufgaben auf den Geräten. Runtime-Rechte können Sie ebenso wie Engineering-Rechte nicht für systemdefinierte Rollen ändern. Für benutzerdefinierte Rollen können Sie die Runtime-Rechte beliebig ändern. Die Runtime-Rechte einer benutzerdefinierten Rolle legen Sie pro Gerät fest.

Es muss mindestens ein Benutzer vorhanden sein, dem eine Rolle mit vollständigen Runtime-Rechten für ein Security-Modul zugeordnet ist, z.B. die systemdefinierte Rolle "NET Administrator".

Den systemdefinierten Rollen mit dem Präfix "NET" sind folgende Engineering-Rechte zugeordnet:

Recht	NET Administrator	NET Standard	NET Diagnose	NET Remote Access	NET Administrator Radius	NET Radius
Projekt schreibgeschützt öffnen	x	x	x	-	-	-
Projekt mit Schreibrechten öffnen	x	x	-	-	-	-
Security-Einstellungen bearbeiten: "Einstellungen" und "Benutzer und Rollen"	x	-	-	-	-	-
Security: Security-Module mit Leserechten	x	x	x	-	x	x
Security: Security-Module mit Schreibrechten	x	x	-	-	x	-

Runtime-Rechte für SCALANCE S-600

Folgende Runtime-Rechte stehen zur Auswahl:

Recht	NET Administrator	NET Standard	NET Diagnose	NET Remote Access	NET Administrator Radius	NET Radius
Laden der Konfigurationsdateien	x	x	-	-	x	-
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x	-	x	x

Zertifikate für Security-Funktionen

Bei der Projektierung können nur Zertifikate verwendet werden, die vom TIA Portal erzeugt wurden, beispielsweise VPN-Gruppen-Zertifikate. Nicht vom TIA Portal erzeugte Zertifikate, beispielsweise CA-Zertifikate für SINEMA Remote Connect, können nicht verwendet werden. Solche Zertifikate müssen Sie nachträglich über das WBM hinzufügen und auf das Gerät laden. Weiterführende Informationen finden Sie im zugehörigen WBM-Projektierungshandbuch.

Eigenschaften und Parameter bearbeiten

Möglichkeiten zum Bearbeiten

Zum Bearbeiten von Eigenschaften und Parameter gibt es folgende Möglichkeiten:

- Hardware- und Netzwerkeditor
  Nachdem Sie die Netzwerkkomponente eingefügt haben, können Sie die Eigenschaften und Parameter bearbeiten, z. B. den Gerätenamen. Weitere Informationen dazu finden Sie unter "Hardware- und Netzwerkeditor".
- Web Based Management (WBM)
  Die Parameter und Eigenschaften sind über mitgelieferten HTML-Seiten (WBM-Seiten) erreichbar. Jede WBM-Seite hat eine eigene Hilfeseite, die die Eigenschaften und Parameter beschreiben. Beachten Sie für weitere Informationen die Projektierungshandbücher der Geräte. Sie finden diese auf den Seiten des Siemens Industry Online Support.
- Command Line Interface
  Mit dem CLI können alle Konfigurationseinstellungen für das Gerät festgelegt werden. Das CLI bietet die gleichen Möglichkeiten wie das Web Based Management (WBM). Beachten Sie für weitere Informationen die Projektierungshandbücher der Geräte. Sie finden diese auf den Seiten des Siemens Industry Online Support.

Betriebsmodi

Folgende Betriebsmodi gibt es:

- Offline-Projektierungsmodus
  In diesem Modus sind nur die Einstellungen verfügbar, die keine Verbindung zum Gerät benötigen.
- Online-Diagnosemodus
  Wenn eine Online-Verbindung zum Gerät besteht, sind unter "Online & Diagnose" zusätzliche Seiten verfügbar. In der Dokumentation dieser Seiten ist der Hinweis "Diese Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht" enthalten. Bei einzelnen Einstellungen ist der Zusatz "Nur Online verfügbar" enthalten.

Eintrag anlegen und löschen

Als Beispiel wird ein Eintrag beim Syslog-Client angelegt und gelöscht. Das Vorgehen ist bei allen Seiten gültig.

Neuen Eintrag anlegen

1. Selektieren Sie in der Netz- oder Gerätesicht das Gerät.
2. Öffnen Sie im Inspektorfenster die Eigenschaften des Geräts.
3. Gehen Sie im Inspektorfenster zu System > Syslog-Client.
4. Tragen Sie bei IP-Adresse des Syslog-Servers die IP-Adresse ein.
5. Klicken Sie in die Tabelle.
6. Wählen Sie im Kontextmenü den Eintrag "Neuer Eintrag".
   In der Tabelle wird ein neuer Eintrag erstellt.

Eintrag löschen

1. Markieren Sie in der Tabelle den gewünschten Eintrag.
2. Wählen Sie im Kontextmenü den Eintrag "Löschen".

Häufig verwendete Schaltflächen

- **Aktualisieren der Anzeige mit "Aktualisieren"**
  Seiten, die aktuelle Parameter anzeigen, haben am unteren Rand die Schaltfläche "Aktualisieren". Klicken Sie auf diese Schaltfläche, wenn Sie für die angezeigte Seite aktuelle Daten vom Gerät anfordern wollen.
- **Speichern von Einstellungen für alle Ports mit "In Tabelle übernehmen"**
  Seiten, auf denen Sie mehrere Ports konfigurieren können, verfügen über 2 Tabellen. In der ersten Tabelle können Sie Einstellungen für alle Ports vornehmen und in die zweite Tabelle übernehmen. In der letzten Spalte der ersten Tabelle gibt es die Schaltfläche "In Tabelle übernehmen". Klicken Sie auf die Schaltfläche, um eingegebene Einstellungen für alle Ports zu speichern.

Informationen anzeigen

Versionen

Die Seite zeigt die Ausgabestände der Hardware und der Software für das Gerät an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Hardware**
  - Basic Device
    Zeigt das Grundgerät an
  - PX.X
    X.X = Port, bei dem das SFP-Modul gesteckt ist.
  - SlotX
    "X" = Steckplatznummer: In diesem Steckplatz eingestecktes Modul.
- **Name**
  Zeigt den Namen des Geräts oder des Moduls an.
- **Ausgabestand**
  Zeigt den Hardware-Ausgabestand des Geräts an.
- **Artikelnummer**
  Zeigt die Artikelnummer des Geräts oder des beschriebenen Moduls an.
- **Software**
  - Firmware
    Zeigt die aktuelle Firmware-Version an. Wenn eine neue Firmware-Datei geladen wurde und das Gerät noch nicht neu gestartet ist, wird hier die Firmware-Version der geladenen Firmware-Datei angezeigt. Nach dem nächsten Neustart wird die geladene Firmware aktiviert und verwendet.
  - Bootloader
    Zeigt die Version der Boot-Software an, die im Gerät gespeichert ist.
  - Firmware_Running
    Zeigt die Firmware-Version an, die aktuell vom Gerät verwendet wird.
- **Beschreibung**
  Zeigt die Kurzbeschreibung der Software an.
- **Version**
  Zeigt die Versionsnummer des Software-Ausgabestands an.
- **Datum**
  Zeigt das Erstellungsdatum des Software-Ausgabestands an.

I&M

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite beinhaltet Informationen zu gerätespezifischen Hersteller- und Wartungsdaten wie Artikelnummer, Seriennummer, Versionsnummern etc. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Zeilen:

- **Hersteller-ID**
  Zeigt die Herstellerkennung an.
- **Artikelnummer**
  Zeigt die Artikelnummer an.
- **Seriennummer**
  Zeigt die Seriennummer an.
- **Hardware-Ausgabestand**
  Zeigt den Hardware-Ausgabestand an.
- **Software-Ausgabestand**
  Zeigt den Software-Ausgabestand an.
- **Versionszähler**
  Zeigt den Versionsänderungs-Zähler an: Zähler für Versionsänderungen seit der Erst-Inbetriebnahme
- **Aktualisierungsdatum**
  Datum und Uhrzeit der letzten Versionsänderung
- **Funktionskennzeichen**
  Zeigt das Funktionskennzeichen (Anlagenkennzeichen) des Geräts an. Das Anlagenkennzeichen (AKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Ortskennzeichen**
  Zeigt das Ortskennzeichen des Geräts an. Das Ortskennzeichen (OKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Datum**
  Zeigt das Datum, das bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.
- **Deskriptor**
  Zeigt die Beschreibung, die bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.

ARP-Tabelle

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zuordnung von MAC-Adresse und IP-Adresse

Über das Address Resolution Protocol (ARP) erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv4-Adresse. Dieser Zuordnung wird von jedem Netzteilnehmer in seiner eigenen ARP-Tabelle gepflegt. Die Seite zeigt die ARP-Tabelle des Geräts.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IP-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

Log-Tabellen

Ereignis-Log

Protokollierung von Ereignissen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt in tabellarischer Form die aufgetretenen Systemereignisse an. Einige der Systemereignisse sind unter "System > Ereignisse" konfigurierbar, z. B. wann sich der Verbindungsstatus eines Ports geändert hat.

Der Inhalt der Tabelle bleibt auch nach dem Ausschalten des Gerätes erhalten. Die Ereignisprotokolldatei können Sie über HTTP, TFTP oder SFTP herunterladen.

Beschreibung der angezeigten Werte

- **Severity-Filter**
  Die Einträge der Tabelle können Sie im WBM nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.
  - Info
    Informativ
  - Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Info" angezeigt.
  - Warning
    Warnend
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Warning" angezeigt.
  - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Critical" angezeigt.

  Die Tabelle gliedert sich in folgende Spalten:
  - **Neustart**
    Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis eingetreten ist.
  - **Systembetriebszeit**
    Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.
  - **Systemzeit**
    Wenn die Systemzeit gesetzt ist, werden Datum und Uhrzeit angezeigt, bei der das Ereignis eingetreten ist.
    Wenn keine Systemzeit eingestellt ist, enthält das Feld die Angabe "Date/time not set"
  - **Severity**
    Einordnung des Eintrags in obige Kategorien.
  - **Log-Meldung**
    Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an.

Hinweis

Die Anzahl der Einträge in dieser Tabelle ist auf 1200 beschränkt. Die Tabelle kann für jede Severity 400 Einträge enthalten. Wenn diese Zahl erreicht ist, werden die ältesten Einträge der jeweiligen Severity verworfen. Die Tabelle verbleibt permanent im Speicher.

Security-Log

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt in tabellarischer Form die Ereignisse an, die bei der Kommunikation über einen gesicherten VPN-Tunnel aufgetreten sind.

Beschreibung der angezeigten Werte

- **Severity-Filter**
  Die Einträge der Tabelle können Sie im WBM nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.
  - Info
    Informativ
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Info" angezeigt.
  - Warning
    Warnend
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Warning" angezeigt.
  - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis eingetreten ist.
- **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.
- **Systemzeit**
  Wenn die Systemzeit gesetzt ist, werden Datum und Uhrzeit angezeigt, bei der das Ereignis eingetreten ist. Wenn keine Systemzeit eingestellt ist, enthält das Feld die Angabe "Date/time not set"
- **Severity**
  Einordnung des Eintrags in obige Kategorien.
- **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an.

Schaltflächen

Schaltfläche "Leeren"

Klicken Sie auf diese Schaltfläche, um den Inhalt der Ereignisprotokolldatei zu löschen. Es werden alle Einträge gelöscht, unabhängig davon, was Sie unter "Severity-Filter" ausgewählt haben.

Die Anzeige wird dabei ebenfalls geleert. Erst wenn nach einem Wiederherstellen der Werkseinstellungen das Gerät neu gestartet ist, wird der Neustart-Zähler zurückgesetzt.

Hinweis

Die Anzahl der Einträge in dieser Tabelle ist auf 1200 beschränkt. Die Tabelle kann für jede Severity 400 Einträge enthalten. Wenn diese Zahl erreicht ist, werden die ältesten Einträge der jeweiligen Severity verworfen. Die Tabelle verbleibt permanent im Speicher.

Firewall-Log

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Das Firewall-Logbuch protokolliert die Ereignisse, die an der Firewall eingetreten sind. Beim Anlegen von Firewall-Regeln können Sie festlegen, mit welcher Ereignisschwere diese protokolliert werden.

Beschreibung der angezeigten Werte

- **Severity-Filter**
  Die Einträge der Tabelle können Sie im WBM nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.
  - Info
    Informativ
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Info" angezeigt.
  - Warning
    Warnend
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Warning" angezeigt.
  - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis eingetreten ist.
- **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.

- **Systemzeit**
  Wenn die Systemzeit gesetzt ist, werden Datum und Uhrzeit angezeigt, bei der das Ereignis eingetreten ist. Wenn keine Systemzeit eingestellt ist, enthält das Feld die Angabe "Date/time not set"
- **Severity**
  Einordnung des Eintrags in obige Kategorien.
- **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an.

Beschreibung der Schaltflächen und Eingabefelder

Schaltfläche "Leeren"

Klicken Sie auf diese Schaltfläche, um den Inhalt der Ereignisprotokolldatei zu löschen. Es werden alle Einträge gelöscht, unabhängig davon, was Sie unter "Severity-Filter" ausgewählt haben.

Die Anzeige wird dabei ebenfalls geleert. Erst wenn nach einem Wiederherstellen der Werkseinstellungen das Gerät neu gestartet ist, wird der Neustart-Zähler zurückgesetzt.

Hinweis

Die Anzahl der Einträge in dieser Tabelle ist auf 1200 beschränkt. Die Tabelle kann für jede Severity 400 Einträge enthalten. Wenn diese Zahl erreicht ist, werden die ältesten Einträge der jeweiligen Severity verworfen. Die Tabelle verbleibt permanent im Speicher.

Fehler

Fehlerstatus

Wenn ein Fehler auftritt, wird er auf dieser Seite angezeigt. Am Gerät werden Fehler durch das Leuchten der Fehler-LED signalisiert.

Gemeldet werden interne Fehler des Geräts sowie Fehler, die Sie auf folgenden Seiten konfigurieren:

- "System > Ereignisse"
- "System > Fehlerkontrolle"

Fehler des Ereignisses "Kalt-/Warmstart" können Sie durch eine Bestätigung löschen.

Die Berechnung des Fehlerzeitpunkts beginnt jeweils nach dem letzten Systemstart.

Wenn keine Fehler vorliegen, schaltet sich die Fehler-LED ab.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

- **Anzahl der gemeldeten Fehler**
  Anzahl der seit dem letzten Hochlauf angezeigten Fehler.
- **Schaltfläche "Zähler zurücksetzen"**
  Klicken Sie auf "Zähler zurücksetzen", um den Zähler zurückzusetzen. Der Zähler wird durch einen Neustart zurückgesetzt.

Die Tabelle enthält die folgenden Spalten:

- **Fehlerzeitpunkt**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der der beschriebene Fehler aufgetreten ist.
- **Fehlerbeschreibung**
  Zeigt eine Kurzbeschreibung des aufgetretenen Fehlers an.
- **Fehlerstatus löschen**
  Wenn die Schaltfläche "Fehlerstatus löschen" aktiv ist, können Sie den Fehler löschen.

SNMP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die angelegten SNMPv3-Gruppen. Die SNMPv3-Gruppen konfigurieren Sie unter "System > SNMP".

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt den Gruppennamen an.
- **Benutzername**
  Zeigt den Benutzer an, welcher der Gruppe zugeordnet ist.

LLDP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Status der Nachbarschaftstabelle

Diese Seite zeigt den aktuellen Inhalt der Nachbarschaftstabelle. In dieser Tabelle sind die Informationen gespeichert, die der LLDP-Agent von angeschlossenen Geräten empfangen hat. Über welche Schnittstellen der LLDP-Agent Informationen empfängt bzw. versendet, legen Sie in folgendem Kapitel fest: "Layer 2 > LLDP".

Beschreibung der angezeigten Werte

Die Tabelle enthält die folgenden Spalten:

- **Systemname**
  Systemname des angeschlossenen Geräts.
- **Geräte-ID**
  Gerätekennung des angeschlossenen Geräts. Die Geräte-ID entspricht dem Gerätenamen, der über PST (STEP 7) vergeben wird. Wenn kein Gerätename vergeben ist, wird die MAC-Adresse des Geräts angezeigt.
- **Lokale Schnittstelle**
  Der Port, an dem das Gerät die Informationen empfangen hat.
- **Speicherzeit**
  Ein Eintrag bleibt für die hier angegebene Zeit im Gerät gespeichert. Wenn das Gerät in dieser Zeit keine neuen Informationen von dem angeschlossenen Gerät erhält, wird der Eintrag gelöscht.
- **Eigenschaft**
  Zeigt die Eigenschaften des angeschlossenen Geräts an:
  - Router
  - Bridge
  - Telephone
  - DOCSIS Cable Device
  - WLAN Access Point
  - Repeater
  - Station
  - Other
- **Port-ID**
  Port des Geräts, der mit dem das Gerät verbunden ist.

Routing

Einleitung

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Routen an, die aktuell verwendet werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Zieladresse dieser Route an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske dieser Route an.
- **Gateway**
  Zeigt das Gateway für diese Route an.
- **Schnittstelle**
  Zeigt die Schnittstelle für diese Route an.
- **Metrik**
  Zeigt die Metrik der Route an. Je größer der Wert, desto länger benötigen Pakete zu Ihrem Ziel.
- **Routing-Protokoll**
  Zeigt an, aus welchem Routing-Protokoll der Eintrag der Routingtabelle stammt. Folgende Einträge sind möglich:
  - Connected: Verbundene Routen
  - Static: Statische Routen

IPsec VPN

Die Seite zeigt den Status der aktivierten VPN-Verbindungen an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an.
- **Lokaler Host**
  Zeigt die IP-Adresse des Geräts an.
- **Lokale DN**
  Zeigt den Distinguished Name (DN) des Geräts an, der während des Verbindungsaufbaus an die Gegenstelle gemeldet wurde. Der Eintrag wird aus dem Feld "Local ID", dem Gerätezertifikat, oder der IP Adresse des Geräts übernommen.
- **Lokales Subnetz**
  Zeigt das lokale Netz an.
- **Remote-Host**
  Zeigt die IP-Adresse oder den Hostnamen der Gegenstelle an.

- **Remote-DN**
  Zeigt den Distinguished Name (DN) an, den die Gegenstelle beim Verbindungsaufbau gemeldet hat.
- **Remote Subnetz**
  Zeigt das entfernte Netz an.
- **Schlüssel-Lebensdauer**
  Zeigt an, wann die Gültigkeit des Schlüssels abläuft.
- **Status**
  Zeigt den Status der VPN-Verbindung an.

SINEMA RC

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zeigt Informationen zum SINEMA RC-Server an.

Hinweis

Für SCALANCE S615 ist diese Funktion nur mit KEY PLUG nutzbar.

Beschreibung der angezeigten Werte

- **Status**
  Zeigt den Status der Verbindung zum SINEMA RC-Server an.
- **Name des Geräts**
  Wenn projektiert, wird der Name des Geräts angezeigt.
- **Gerätestandort**
  Wenn projektiert, wird der Standort des Geräts angezeigt
- **GSM-Nummer**
  Wenn projektiert, wird die Rufnummer des Geräts angezeigt
- **Hersteller**
  Wenn projektiert, wird der Eintrag angezeigt
- **Kommentar**
  Wenn projektiert, wird der Kommentar angezeigt
- **Verbindungsart (Server)**
  Zeigt an, welche Verbindungsart auf dem SINEMA RC-Server eingestellt ist.
- **Verbindungsart (Gerät)**
  Zeigt an, welche Verbindungsart auf dem Gerät eingestellt ist
- **Fingerabdruck**
  Zeigt den Fingerabdruck des Serverzertifikats an. Wird nur angezeigt, wenn zum Verifizieren der Fingerabdruck verwendet wird.

- Remote-Adresse
  Zeigt die IP-Adresse des SINEMA RC-Servers an.
- Verbundene lokale Subnetze
  Zeigt die IP-Adressen der lokalen Subnetze an. Wird nur angezeigt, wenn auf dem SINEMA RC-Server die Option "Verbundene lokale Subnetze" aktiviert ist. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- Verbundene lokale Hosts
  Zeigt die Ziel-IP-Adresse der Hosts an, die erreichbar sind.
- Adresse Tunnel-Schnittstelle
  Zeigt die IP-Adresse der virtuellen Tunnelschnittstelle an.
- Verbundene Remote-Subnetze
  Zeigt die Subnetze des SINEMA RC-Servers an, die für das Gerät erreichbar sind. Welche Subnetze für das Gerät erreichbar sind, ist von den Kommunikationsbeziehungen auf dem SINEMA RC-Server abhängig. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.

OpenVPN-Client

Die Seite zeigt den Status der aktivierten OpenVPN-Verbindungen an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- Name
  Zeigt den Namen der OpenVPN-Verbindung an.
- Remote-Server
  Zeigt die IP-Adresse oder den Hostnamen des OpenVPN-Servers an.
- Tunnel-Schnittstelle IP
  Zeigt die IP-Adresse der virtuellen Tunnelschnittstelle an.
- Exportierte Subnetze
  Zeigt die IP-Adresse der lokalen Subnetze an.
- Geroutete Subnetze
  Zeigt die Subnetze des Open VPN-Servers an.
- Status
  Zeigt den Status der OpenVPN-Verbindung an.

Redundanz

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein.
- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Port**
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten**
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge
- **Topologieänderungen / Letzte Topologieänderung**
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit des letzten Hochlaufs. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz "Sek." hinter der Zahlenangabe
  - Minuten: Zusatz "Min." hinter der Zahlenangabe
  - Stunden: Zusatz "Std." hinter der Zahlenangabe
- **Bridge Hello Time[s] / Root Hello Time[s]**
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei solchen Telegrammen ist die Hello-Time.
- **Bridge Forward Delay / Root Forward Delay**
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort, sondern erst nach dem im Parameter Weiterleitungsverzögerung festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben. Der
- **Bridge Max Age / Root Max Age**
  Nach Ablauf des Max Age-Timers wird die empfangene BPDU verworfen, um vom Switch als gültig akzeptiert zu werden. Der Defaultwert liegt bei 20s.

- **Bridge Max Hop Count**
  Dieser Parameter gibt an, wie viele STP-Teilnehmer eine BPDU passieren darf. Wird eine MSTP-BPDU empfangen, deren Hop Count den hier konfigurierten Wert übersteigt, wird sie verworfen.
- **Root Hop Count**
  Die Anzahl der Teilnehmer, die bis zur Root-Bridge durchlaufen werden müssen.

Spanning Tree

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Einleitung

Die Seite zeigt die aktuellen Informationen zu Spanning Tree und die Einstellungen der Root Bridge an.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Spanning Tree-Modus**
  Zeigt den eingestellten Modus an. Den Modus legen Sie bei "Layer 2 > Konfiguration" und bei "Layer 2 > Spanning Tree > Allgemein" fest.
  Folgende Werte sind möglich:
  - '-'
  - RSTP
- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 32768.
- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Switch an.
- Root-Kosten
  Zeigt die Pfadkosten von dem Gerät bis zur Root Bridge.
- Bridge-Status
  Zeigt den Status der Bridge an, z. B. ob das Gerät die Root-Bridge ist.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die Schnittstellen an, über den das Gerät kommuniziert.
- **Rolle**
  Zeigt den Status des Ports an. Folgende Werte sind möglich:
  - Disabled
    Der Port wurde manuell aus dem Spanning Tree entfernt und wird vom Spanning Tree nicht mehr berücksichtigt.
  - Designated
    Die Ports, die von der Root-Bridge wegführen.
  - Alternate
    Der Port mit einem alternativen Weg zu einem Netzwerksegment
  - Backup
    Wenn ein Switch mehrere Ports zu dem gleichen Netzwerksegment hat, wird der "schlechtere" Port zum Backup-Port.
  - Root
    Der Port, der den besten Weg zur Root Bridge bietet.
  - Master
    Dieser Port zeigt zu einer Root-Bridge, die außerhalb der MST-Region liegt.
- **Status**
  Zeigt den aktuellen Status an, in dem sich die Schnittstelle befindet. Die Werte werden nur angezeigt. Der Parameter ist abhängig vom projektierten Protokoll.
  - Discarding
    Der Port empfängt BPDU-Telegramme. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    Der Port empfängt und sendet BPDU-Telegramme. Der Port ist in den Spanning Tree-Algorithmus einbezogen. Andere aus- und eingehende Telegramme werden verworfen.
  - Learning
    Der Port lernt aktiv die Topologie, d. h. die Teilnehmeradressen. Andere aus- und eingehende Telegramme werden verworfen.
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz. Der Port empfängt und sendet Datentelegramme.
- **Oper. Version**
  Zeigt den Kompatibilitätsmodus von Spanning Tree, der vom Port verwendet wird.
- **Priorität**
  Kann der vom Spanning-Tree ermittelte Weg alternativ über mehrere Ports eines Gerätes führen, so wird der Port mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) ausgewählt. Für die Priorität kann ein Wert von 0 bis 240 in 16er Schritte eingegeben werden. Wenn Sie einen Wert eingeben, der nicht durch 16 teilbar ist, wird der Wert automatisch angepasst. Der Standardwert ist 128.

- **Pfadkosten**
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Es wird die Strecke mit dem geringsten Wert als Weg ausgewählt. Haben mehrere Ports eines Gerätes den gleichen Wert, wird der Port mit der niedrigsten Portnummer ausgewählt.
  Ist der Wert im Feld "Cost Calc" "0", so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "Cost Calc" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000
- **Edge-Typ**
  Zeigt den Typ der Verbindung an. Folgende Werte sind möglich:
  - Edge Port
    An diesem Port befindet sich ein Endgerät.
  - No Edge Port
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.
- **P.t.P. Type**
  zeigt die Art der Punkt-zu-Punkt-Verbindung an. Folgende Werte sind möglich:
  - P.t.P.
    Bei Halbduplex wird von einer Punkt-zu-Punkt-Verbindung ausgegangen.
  - Shared Media
    Bei einer Vollduplexverbindung wird nicht von einer Punkt-zu-Punkt-Verbindung ausgegangen.

VRRPv3-Statistiken

Einleitung

Diese Seite zeigt die Statistiken des VRRPv3-Protokolls und aller konfigurierten virtuellen Router an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **VRID-Fehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine nicht unterstütze VRID enthalten.
- **Versionsfehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Versionsnummer enthalten.
- **Prüfsummenfehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Prüfsumme enthalten.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstellen**
  Schnittstelle, auf die sich die Einstellungen beziehen.
- **VRID**
  Zeigt die ID des virtuellen Routers an. Gültige Werte sind 1 ... 255.
- **Typ**
  Zeigt die Version des IP-Protokolls an
- **Übergang in Master-Zustand**
  Zeigt an, wie häufig dieser virtuelle Router in den Zustand "Master" ging.
- **Empf. Advertisements**
  Zeigt an, wie viele VRRPv3-Paket empfangen wurden.
- **Advertisement-Intervall-Fehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Intervall nicht mit dem lokal gesetzten Wert übereinstimmt.
- **IP TTL Fehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren TTL (Time to live) Wert im IP Header nicht stimmt.
- **Empf. Prio-0-Tel**
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 empfangen wurden. VRRPv3-Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- **Ges. Prio-0-Tel**
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 versendet wurden. Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- **Ungültiger Typ**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Wert im Feld "Typ" des IP-Headers ungültig ist.

- **Adresslistenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Adressliste nicht mit der lokal konfigurierten Liste übereinstimmt.
- **Telegrammlängenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Länge nicht korrekt ist.

Security

Übersicht

Die Seite zeigt die Sicherheitseinstellungen und die lokalen und externen Benutzerkonten an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Dienste

Die Liste "Dienste" zeigt die Sicherheitseinstellungen an.

- **Telnet-Server**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration"
  - Aktiviert: Unverschlüsselter Zugriff auf das CLI
  - Deaktiviert: Kein unverschlüsselter Zugriff auf das CLI
- **SSH-Server**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration".
  - Aktiviert: Verschlüsselter Zugriff auf das CLI
  - Deaktiviert: Kein verschlüsselter Zugriff auf das CLI
- **Webserver**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration"
  - HTTP/HTTPS: Der Zugriff auf das WBM ist über HTTP und HTTPS möglich.
  - HTTPS: Der Zugriff auf das WBM ist nur noch über HTTPS möglich.

- **SNMP**
  Die Einstellung konfigurieren Sie unter "System > SNMP > Allgemein".
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich.
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit der SNMP Version 3 möglich.
- **Login-Authentifizierung**
  Die Einstellung konfigurieren Sie unter "Security > AAA > Allgemein".
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
  - Lokal und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen.
    Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS mit Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
    Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.
- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie aktuell verwendet wird.

Lokale und externe Benutzerkonten

Lokale Benutzerkonten und Rollen konfigurieren Sie unter "Security > Benutzer".

Wenn Sie ein lokales Benutzerkonto anlegen, wird automatisch auch ein externes Benutzerkonto erzeugt. Bei lokalen Benutzerkonten handelt es sich um Benutzer mit jeweils einem Passwort zur Anmeldung auf dem Gerät.

In der Tabelle "Externe Benutzerkonten" wird ein Benutzer mit einer Rolle verknüpft, z. B. der Benutzer "Observer" mit der Rolle "user". In diesem Fall ist der Benutzer auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert, die zugehörige Gruppe jedoch unbekannt oder nicht vorhanden ist, prüft das Gerät, ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzerkonten" gibt. Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet. Wenn die zugehörige Gruppe auf dem Gerät bekannt ist, werden beide Tabellen ausgewertet. Dem Benutzer wird die Rolle mit den größeren Rechten zugewiesen.

Hinweis

Die Tabelle "Externe Benutzerkonten" wird nur ausgewertet, wenn Sie im RADIUS Authorisierungsmodus "Herstellerspezifisch" eingestellt haben.

Über CLI können Sie auf die externen Benutzerkonten zugreifen.

Die Tabelle "Lokale Benutzerkonten" gliedert sich in folgende Spalten:

- **Benutzerkonto**
  Zeigt den Namen des lokalen Benutzers an.
- **Rolle**
  Zeigt die Rolle des Benutzers an. Weitere Informationen zu den Funktionsrechten der Rolle erhalten Sie unter "Information > Security > Rollen".

Unterstützte Funktionsrechte

Die Seite zeigt die Funktionsrechte an, die lokal auf dem Gerät verfügbar sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

- **Funktionsrecht**
  Zeigt die Nummer des Funktionsrechts an. Den Nummern sind unterschiedliche Rechte in Bezug auf die Geräteparameter zugeordnet.
- **Beschreibung**
  Zeigt die Beschreibung des Funktionsrechts an.

Rollen

Die Seite zeigt die Rollen an, die lokal auf dem Gerät gültig sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Rolle**
  Zeigt die Rolle an.
- **Funktionsrecht**
  Zeigt die Funktionsrechte der Rolle an:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen, aber nicht verändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.
  - 0
    Hierbei handelt es sich um eine Rolle, die das Gerät intern vergibt, wenn der Benutzer nicht authentifiziert werden konnte.
    Dem Benutzer wird der Zugriff auf das Gerät verweigert.
- **Beschreibung**
  Zeigt eine Beschreibung der Rolle an.

Gruppen

Diese Seite zeigt an, welche Gruppe mit welcher Rolle verknüpft ist. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an. Der Name entspricht der Gruppe auf dem RADIUS-Server.
- **Rolle**
  Zeigt die Rolle an. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät.
- **Beschreibung**
  Zeigt die Beschreibung für die Verknüpfung an.

System-Funktionen projektieren

Konfiguration

Systemkonfiguration

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes.

Legen Sie fest, über welche Dienste auf das Gerät zugegriffen wird. Zu einigen Diensten gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind.

Beschreibung

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes

- **Telnet-Server**
  Aktivieren oder deaktivieren Sie den Dienst Telnet-Sever für den unverschlüsselten Zugriff auf das CLI.
- **SSH-Server**
  Aktivieren oder deaktivieren Sie den Dienst SSH-Server für den verschlüsselten Zugriff auf das CLI.
- **HTTP-Dienste**
  Legen Sie fest, wie auf das WBM zugegriffen wird:
  - HTTPS
    Zugriff auf das WBM nur über HTTPS möglich.
  - HTTP/HTTPS
    Zugriff auf das WBM nur über HTTP und HTTPS möglich.
  - HTTP nach HTTPS umleiten
    Beim Zugriff über HTTP wird automatisch nach HTTPS umgeleitet.
- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client. Weitere Einstellungen konfigurieren Sie unter "System > SMTP-Client".
- **Syslog-Client**
  Aktivieren oder deaktivieren Sie den Syslog-Client. Weitere Einstellungen konfigurieren Sie unter "System > Syslog-Client".
- **DCP-Server**
  Legen Sie fest, ob auf das Gerät mit DCP (Discovery and Configuration Protocol) zugegriffen werden kann:
  - "-" (Deaktiviert)
    DCP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben
    Mit DCP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt
    Mit DCP können Geräteparameter zwar gelesen aber nicht verändert werden.

- **Zeiteinstellung**
  Wählen Sie die gewünschte Einstellung aus. Folgende Einstellungen sind möglich:
  - Manuell
    Die Systemzeit wird manuell eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > Manuelle Einstellung".
  - SIMATIC Time
    Die Systemzeit wird über einen SIMATIC Zeitgeber eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SIMATIC Time Client".
  - SNTP-Client
    Die Systemzeit wird über einen SNTP-Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SNTP-Client".
  - NTP-Client
    Die Systemzeit wird über einen NTP-Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > NTP-Client".
- **SNMP**
  Wählen Sie das gewünschte Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit SNMP Version 3 möglich. Weitere Einstellungen konfigurieren Sie unter " System > SNMP > Allgemein".
- **SNMPv1/v2 schreibgeschützt**
  Aktivieren oder deaktivieren Sie den schreibenden Zugriff auf SNMP-Variablen bei SNMPv1/v2c.
- **DHCP-Client**
  Aktivieren oder deaktivieren Sie den DHCP-Client. Weitere Einstellungen konfigurieren Sie unter "System > DHCP"

- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Versenden von SNMPv1-Traps (Alarmtelegramme). Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Traps".
- **Konfigurationsmodus**
  Wählen Sie die gewünschte Betriebsart. Folgende Betriebsarten sind möglich:
  – Automatisches Speichern
    Automatischer Sicherungsbetrieb. Ca. 1 Minute nach der letzten Parameteränderung oder vor dem Neustart des Geräts wird die Konfiguration automatisch abgespeichert. Zusätzlich erscheint im Anzeigebereich des WBM folgende Meldung "Die Änderungen werden automatisch in x Sekunden gespeichert. Um die Änderungen sofort zu speichern, klicken Sie auf 'Schreiben der Startkonfiguration'."

    Hinweis

    Unterbrechung des Speichervorgangs

    Der Speichervorgang startet erst, nachdem der Timer in der Meldung abgelaufen ist. Die Dauer des Speichervorgangs ist vom Gerät abhängig.
    - Schalten Sie das Gerät nicht sofort aus, nachdem der Timer abgelaufen ist.

  – Trial
    Trial-Modus. Im Trial-Modus werden Änderungen zwar übernommen aber nicht in der Konfigurationsdatei (Startup Configuration) gespeichert.
    Um Änderungen in der Konfigurationsdatei abzuspeichern, verwenden Sie im WBM die Schaltfläche "Schreiben der Startkonfiguration". Zusätzlich wird im Anzeigebereich die Meldung "Der Konfigurationsmodus Trial ist aktiv - Klicken Sie auf die Schaltfläche "Schreiben der Startkonfiguration" um Ihre Einstellungen zu speichern" angezeigt, sobald es ungespeicherte Änderungen gibt. Diese Meldung ist auf jeder WBM-Seite sichtbar, bis die vorgenommenen Änderungen entweder gespeichert werden oder das Gerät neu gestartet wird.

Allgemein

Gerät

Diese Seite enthält die allgemeinen Geräteinformationen.

Beschreibung

Die Seite enthält folgende Felder:

- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemuhrzeit an. Die Systemuhrzeit wird entweder vom Anwender eingestellt oder per Uhrzeittelegramm synchronisiert: entweder SINEC H1 Uhrzeittelegramm, NTP oder SNTP. (Nur lesbar)
- **Systembetriebszeit** (nur Online verfügbar)
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an. (Nur lesbar)
- **Gerätetyp** (nur Online verfügbar)
  Zeigt die Typenbezeichnung des Geräts an. (Nur lesbar)

- **Systemname**
  Sie können den Namen des Gerätes eintragen. Der eingetragene Name wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich. Der Systemname wird auch in der CLI-Eingabeaufforderung (Prompt) angezeigt. In der CLI-Eingabeaufforderung ist die Anzahl der Zeichen begrenzt. Der Systemname wird nach 16 Zeichen abgeschnitten.
- **Kontaktperson**
  Sie können den Namen einer Kontaktperson eintragen, die für die Verwaltung des Gerätes zuständig ist. Es sind maximal 255 Zeichen möglich.
- **Gerätestandort**
  Sie können den Montageort des Gerätes eintragen. Der eingetragene Montageort wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich.

Hinweis

Erlaubte Zeichen

Folgende darstellbare ASCII-Zeichen (0x20 bis 0x7e) in den Eingabfeldern Felder **"Systemname"**, **"Kontaktperson"** und **"Gerätestandort"** sind erlaubt:

- 0123456789
- A...Z a...z
- !"#$%&'()*+,-./:;<=>?@ [\]_{|}~^`

Koordinaten

Auf dieser Seite können Informationen über die geografischen Koordinaten (Breitengrad, Längengrad und die Höhe über dem Ellipsoid gemäß WGS84) eingetragen werden.

Ermittlung der Koordinaten

Nutzen Sie zur Ermittlung der geografischen Koordinaten des Gerätes entsprechendes Kartenmaterial.

Die geografischen Koordinaten können auch durch einen GPS-Empfänger ermittelt werden. Meist werden die geografischen Koordinaten von diesen Geräten direkt angezeigt und müssen nur noch in die Eingabefelder dieser Seite übertragen werden.

Beschreibung

Die Seite enthält folgende Eingabefelder mit einer maximalen Länge von 32 Zeichen:

- **Eingabefeld "Geographische Breite"**
  Geben Sie die nördliche oder die südliche Breite für den Standort des Gerätes ein.
  Z. B. +49° 1´ 31.67" bedeutet, dass sich das Gerät auf 49 Grad, 1 Bogenminute und 31.67 Bogensekunden nördlicher Breite befindet.
  Die südliche Breite wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben N (nördliche Breite) oder S (südliche Breite) an die Zahlenangabe anhängen (49° 1´ 31.67" N).
- **Eingabefeld "Geographische Länge"**
  Geben Sie die östliche oder die westliche Länge für den Standort des Gerätes ein.
  Z. B. +8° 20´ 58.73" bedeutet, dass sich das Gerät auf 8 Grad, 20 Bogenminuten und 58.73 Bogensekunden östliche Länge befindet.
  Die westliche Länge wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben O bzw. E (östliche Länge) oder W (westliche Länge) an die Zahlenangabe anhängen (8° 20´ 58.73" E).
- **Eingabefeld "Geographische Höhe"**
  Geben Sie die geografische Höhe über oder unter normal Null (Meereshöhe) in Metern ein.
  Z.B. 158 m bedeutet, dass sich das Gerät in einer Höhe von 158 m über normal Null befindet.
  Höhenangaben unterhalb von normal Null (z. B. am toten Meer) werden mit einem führenden Minuszeichen dargestellt.

Laden & Speichern

TFTP

Laden und Speichern von Daten über einen TFTP-Server

Auf der Seite können Sie den TFTP-Server und die Dateinamen konfigurieren. Weiter können Sie Gerätedaten in einer externen Datei auf einem TFTP-Server speichern bzw. solche Daten aus einer externen Datei vom TFTP-Server in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von einem TFTP-Server laden.

Zudem lassen sich auf dieser Seite die Zertifikate laden, die für den Aufbau einer gesicherten VPN-Verbindung notwendig sind.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die Seite "System > PLUG" gelöscht oder neu beschrieben werden.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

Beschreibung

- **Adresse des TFTP-Servers**
  Tragen Sie die IP-Adresse oder den FQDN des TFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des TFTP-Servers**
  Tragen Sie hier den Port des TFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 69 entsprechend Ihren spezifischen Anforderungen ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.

- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis

  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktion** (nur Online verfügbar)
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp, z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - Datei speichern
    Mit dieser Auswahl speichern Sie eine Datei auf den TFTP-Server.
  - Datei hochladen
    Mit dieser Auswahl laden Sie eine Datei vom TFTP-Server.

SFTP

Laden und Speichern von Daten über einen SFTP-Server

SFTP (SSH File Transfer Protocol) überträgt die Dateien verschlüsselt. Auf dieser Seite konfigurieren Sie die Zugangsdaten für den SFTP-Server.

Weiter können Sie Gerätedaten in einer externen Datei auf Ihrem Client-PC speichern bzw. solche Daten aus einer externen Datei vom PC in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Admin-PC laden.

Zudem lassen sich auf dieser Seite die Zertifikate laden, die für den Aufbau einer gesicherten VPN-Verbindung notwendig sind.

Firmware

Die Firmware ist signiert und verschlüsselt. Es ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

Beschreibung

Die Seite enthält folgende Felder:

- **Adresse des SFTP-Servers**
  Geben Sie die IP-Adresse oder den FQDN des SFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des SFTP-Servers**
  Geben Sie den Port des SFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 22 entsprechend Ihren spezifischen Anforderungen ändern.
- **SFTP Benutzer**
  Geben Sie den Benutzer für den Zugriff auf den SFTP-Server ein. Vorausgesetzt, auf dem SFTP-Server ist ein Benutzer mit den entsprechenden Rechten angelegt.
- **SFTP Passwort**
  Geben Sie das Passwort für den Benutzer ein
- **SFTP Passwort bestätigen**
  Bestätigen Sie das Passwort.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis

  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktionen**
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp. z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - **Datei speichern**
    Mit dieser Auswahl speichern Sie eine Datei auf dem SFTP-Server.
  - **Datei hochladen**
    Mit dieser Auswahl laden Sie eine Datei vom SFTP-Server.

Passwörter

Passwort für Dateien

Es gibt Dateien, deren Zugriff passwortgeschützt ist. Um die Datei erfolgreich ins Gerät zuladen, geben Sie auf der Seite das für die Datei festgelegte Passwort ein.

Beschreibung

- **Typ**
  Zeigt den Dateityp an
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Aktiviert**
  Wenn aktiviert, wird die Datei verwendet. Nur aktivierbar, wenn das Passwort konfiguriert ist
- **Passwort**
  Geben Sie das Passwort für die Datei ein.
- **Passwort bestätigen**
  Bestätigen Sie das Passwort.
- **Status** (nur Online verfügbar)
  Zeigt an, ob das Passwort zur der Datei auf dem Gerät passen.
  - gültig
    Das Optionskästchen "Aktiviert" ist aktiviert und das Passwort passt zu der Datei.
  - ungültig
    Das Optionskästchen "Aktiviert" ist aktiviert, aber das Passwort passt nicht zu der Datei oder es ist noch keine Datei geladen.
  - '-'
    Das Passwort kann nicht ausgewertet werden oder wird noch nicht verwendet. Das Optionskästchen "Aktiviert" ist nicht aktiviert.

Ereignisse

Konfiguration

Systemereignisse auswählen

Auf der Seite legen Sie fest, welche Systemereignisse wie protokolliert werden.

Folgende Meldungen werden immer in die Ereignisprotokoll-Tabelle eingetragen und sind nicht abwählbar:

- Ändern des Admin-Kennworts
- Starten des Geräts

- Betriebstatus des Geräts, z. B. ob ein PLUG vorhanden ist oder nicht
- Status unerledigter Fehler

Für SCALANCE S615 können Sie diese Meldungen zusätzlich an einen Syslog-Server senden, aktivieren Sie hierzu beim Ereignis "System Allgemein Logs" das Optionskästchen "Syslog".

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Alle Ereignisse**
  Zeigt an, dass die Einstellungen für alle Ereignisse der Tabelle 2 gültig sind.
- **E-Mail / Trap / Log-Tabelle / Syslog / Fehler / Digitaler Ausgang / VPN-Tunnel**
  Aktivieren oder deaktivieren Sie die gewünschte Art der Benachrichtigung für alle Ereignisse. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ereignisse der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Ereignis**
  Die Spalte "Ereignis" enthält Folgendes:
  - Kalt-/Warmstart
    Das Gerät wurde eingeschaltet oder vom Anwender neu gestartet. Im Fehlerspeicher des Geräts wird ein neuer Eintrag mit der Art des durchgeführten Neustarts erzeugt.
  - Link Change
    Dieses Ereignis tritt nur auf, wenn der Port-Status überwacht wird und sich entsprechend geändert hat, siehe "System > Fehlerkontrolle > Link Change".
  - Authentifizierungsfehler
    Dieses Ereignis tritt beim Versuch eines Zugriffs mit fehlerhaftem Passwort auf.
  - Umschalten der Spannungsversorgung
    Dieses Ereignis tritt nur auf, wenn die Spannungsversorgungsleitungen 1 und 2 überwacht werden. Es zeigt an, dass ein Wechsel auf Leitung 1 bzw. auf Leitung 2 stattgefunden hat. Siehe "System > Fehlerkontrolle > Spannungsversorgung".
  - Änderung des Fehlerstatus
    Der Fehlerstatus hat sich geändert. Der Fehlerstatus kann sich auf die aktivierte Portüberwachung, auf das Ansprechen des Meldekontakts oder die Spannungsüberwachung beziehen.
  - Security-Logs
    Im Sicherheitslogbuch wird eingetragen, wenn das IPsec-Verfahren für VPN angewendet wird oder eine SINEMA RC-Verbindung aktiviert ist.
  - Firewall-Logs
    Im Firewall-Logbuch wird eingetragen, wann einzelne Firewall-Regeln angewendet wurden. Dazu muss zu den verschiedenen Firewall-Funktionen die LOG-Funktion aktiviert werden.
  - DDNS Client-Logs
    Das Ereignis tritt auf, wenn der DDNS-Client die zugewiesene IP-Adresse mit dem im DDNS-Provider registrierten Hostnamen synchronisiert.
  - Digitaler Eingang
    Das Ereignis tritt auf, wenn sich der Zustand des digitalen Eingangs geändert hat.
  - System-Verbindungsstatus
    Der Verbindungsstatus hat sich geändert.
  - System Allgemein Logs
    Verbindungsaufbau, Änderung der Konfiguration.
  - Digitaler Eingang
    Das Ereignis tritt auf, wenn sich der Zustand des digitalen Eingangs geändert hat.
  - VPN-Tunnel
    Das Ereignis tritt auf, wenn sich der Zustand von VPN (IPsec, OpenVPN, SINEMA RC) geändert hat.

  - Secure NTP
    Das Ereignis tritt auf, wenn sich der Zustand vom Secure NTP-Server geändert hat.

- **E-Mail**
  Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- **Trap**
  Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.
- **Log-Tabelle**
  Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle, siehe "Information > Log-Tabelle".
- **Syslog**
  Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- **Fehler**
  Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf
- **Digitaler Ausgang**
  Steuert den digitalen Ausgang an oder signalisiert die Zustandsänderung mit der LED "DO".
- **VPN-Tunnel**
  Steuert die Weitergabe eines Ereignisses an eine VPN-Verbindung (IPsec, OpenVPN, SINEMA RC). Solange das Ereignis anliegt, wird die VPN-Verbindung aktiv geschaltet.

Severity-Filter

Auf dieser Seite konfigurieren Sie die Fehlerschwere für das Versenden von Systemereignisbenachrichtigungen.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Client-Typ**
  Wählen Sie den Client-Typ, für den Sie die Einstellungen vornehmen:
  - **E-Mail**
    Versand von Systemereignismeldungen per E-Mail.
  - **Log-Tabelle**
    Eintragen von Systemereignissen in die Log-Tabelle.
  - **Syslog**
    Eintragen von Systemereignissen in die Syslog-Datei.
- **Severity**
  Wählen Sie die gewünschte Stufe aus. Folgende Einstellungen sind möglich:
  - **Info**
    Die Meldungen aller Stufen werden versendet bzw. protokolliert.
  - **Warning**
    Die Meldung dieser Stufe und der Stufe "critical" werden versendet bzw. protokolliert.
  - **Critical**
    Nur die Meldungen dieser Stufe werden versendet bzw. protokolliert.

SMTP Client

Das Gerät bietet die Möglichkeit, beim Auftreten eines Alarmereignisses automatisch eine E-Mail (z.B. an den Netzwerkadministrator) zu senden. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Damit kann für Netze mit wenigen Teilnehmern eine einfache zentrale Netzüberwachung auf Basis eines E-Mail-Systems aufgebaut werden. Bei eintreffenden E-Mail-Störmeldungen kann über die Identifikation des Absenders per Browser das WBM gestartet werden, um weitere Diagnoseinformationen auszulesen.

Auf dieser Seite können Sie bis zu drei SMTP-Server und die dazugehörigen E-Mail-Adressen konfigurieren.

Beschreibung

Die Seite enthält folgende Felder:

- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client.
- **E-Mail-Adresse des Absenders**
  Geben Sie den Absendernamen ein, der in der E-Mail angegeben werden soll, z. B. den Gerätenamen.
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **Test-E-Mail senden**
  Verschicken Sie eine Test-E-Mail, um Ihre Konfiguration zu prüfen.

- **SMTP-Port**
  Geben Sie den Port ein, über den Ihr SMTP-Server erreichbar ist.
  Werkseinstellung: 25
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **SMTP-Server-Adresse**
  Geben Sie die IP-Adresse, den FQDN (Fully Qualified Domain Name) oder den Hostnamen des SMTP-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in einer zu löschenden Zeile das Optionskästchen.
- **SMTP-Server-Adresse**
  Zeigt die IP-Adresse, den FQDN (Fully Qualified Domain Name) oder den Hostnamen des SMTP-Servers.
- **E-Mail-Adresse des Empfängers**
  Geben Sie die E-Mail-Adresse ein, an die das Gerät im Fehlerfall eine E-Mail sendet.

SNMP

Allgemein

Auf dieser Seite treffen Sie grundlegende Einstellungen für SNMP. Aktivieren Sie die Optionen abhängig von der Funktion, die Sie nutzen wollen.

Beschreibung

- **SNMP**
  Wählen Sie das SNMP-Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (Deaktiviert)
    SNMP deaktiviert.
  - SNMPv1/v2c/v3
    SNMPv1/v2c/v3 wird unterstützt.

    Hinweis

    Beachten Sie, dass SNMP in den Versionen 1 und 2c über keine Sicherheitsmechanismen verfügt.
  - SNMPv3
    Nur SNMPv3 wird unterstützt.
- **SNMPv1/v2 schreibgeschützt**
  Wenn Sie diese Option aktivieren, kann SNMPv1/v2c nur lesend auf die SNMP-Variablen zugreifen.

  Hinweis

  Community String

  Verwenden Sie aus Sicherheitsgründen nicht die Standardwerte "public" oder "private". Ändern Sie die Community Strings nach der Erst-Installation.

  Die empfohlene Mindestlänge für Community Strings sind 6 Zeichen.
- **SNMPv1/v2c Read Community String - Lesen**
  Tragen Sie den Community String für den Zugriff des SNMP-Protokolls ein.
- **SNMPv1/v2c Read/Write Community String - Lesen/Schreiben**
  Tragen Sie den Community String für den lesenden und schreibenden Zugriff des SNMP-Protokolls ein.
- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Senden von SNMPv1-Traps (Alarmtelegramme). Im Eintrag "Trap" legen Sie die IP-Adressen der Geräte fest, an die SNMPv1-Traps gesendet werden.
- **SNMPv1/v2c Trap Community String - Trap**
  Tragen Sie den Community String für das Senden von SNMPv1/v2c-Meldungen ein.

- **SNMPv3 Benutzermigration**
  - **Aktiviert**
    Wenn die Funktion aktiviert ist, wird eine SNMP-Engine-ID generiert, die migriert werden kann. Sie können konfigurierte SNMPv3-Benutzer auf ein anderes Gerät übertragen. Wenn Sie diese Funktion aktivieren und die Konfiguration des Geräts auf ein anderes Gerät laden, bleiben konfigurierte SNMPv3-Benutzer erhalten.
  - **Deaktiviert**
    Wenn die Funktion deaktiviert ist, wird eine gerätespezifische SNMP-Engine-ID generiert. Um die ID zu generieren, wird die Agent-MAC-Adresse des Geräts verwendet. Sie können diese SNMP-Benutzerkonfiguration nicht auf andere Geräte übertragen. Wenn Sie die Konfiguration des Geräts auf ein anderes Gerät laden, werden alle konfigurierten SNMPv3-Benutzer gelöscht.
- **SNMP-Engine-ID** (nur Online verfügbar)
  Zeigt die SNMP-Engine-ID an.

Traps

Beim Eintreten eines Alarmereignisses kann ein Gerät SNMP-Traps (Alarmtelegramme) an bis zu zehn verschiedene Management-Stationen gleichzeitig senden. Es werden nur bei solchen Ereignissen Traps gesendet, die unter "Ereignisse" festgelegt sind.

Hinweis

SNMP-Traps werden nur dann versendet, wenn Sie bei "SNMP >Allgemein" oder unter "System > Konfiguration" die Einstellung "SNMPv1 Traps" aktiviert haben.

Beschreibung

- **Trap-Empfängeradresse**
  Tragen Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) der Station ein, an die das Gerät SNMP-Traps sendet. Sie können bis zu zehn verschiedene Empfänger angeben.

Die Tabelle gliedert sich in folgende Spalten:

- **Trap-Empfängeradresse**
  Ändern Sie bei Bedarf die IP-Adressen oder den FQDN (Fully Qualified Domain Name) der Stationen.
- **Trap**
  Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps. Stationen, die eingetragen, aber nicht selektiert sind, erhalten keine SNMP-Traps.

v3-Gruppen

Sicherheitseinstellungen und Rechtevergabe

SNMP Version 3 bietet eine Rechtevergabe, Authentifizierung und Verschlüsselung auf Protokollebene. Das Security-Level und die Lese-/Schreibrechte werden gruppenspezifisch definiert. Für jedes Mitglied einer Gruppe gelten automatisch die entsprechenden Einstellungen.

Beschreibung

- Gruppenname
  Geben Sie den Namen der Gruppe ein. Die maximale Länge beträgt 32 Zeichen.
- Security-Level
  Wählen Sie die Sicherheitsstufe (Authentifizierung, Verschlüsselung) aus, die für die gewählte Gruppe gültig ist. Es gibt folgende Möglichkeiten:
  - Keine Auth./keine Priv.
    Keine Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./keine Priv.
    Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./Priv.
    Authentifizierung aktiviert / Verschlüsselung aktiviert.

Die Tabelle gliedert sich in folgende Spalten:

- Gruppenname
  Zeigt die definierten Gruppennamen an.

  Hinweis

  Der einmal vergebene Gruppenname und die Sicherheitsstufe können nach dem Anlegen nicht mehr geändert werden. Wenn Sie den Gruppennamen oder die Sicherheitsstufe ändern wollen, müssen Sie die Gruppe löschen und mit dem neuen Namen neu anlegen und neu konfigurieren.

- Security-Level
  Zeigt die konfigurierte Sicherheitsstufe an.
- Lesen
  Aktivieren oder deaktivieren Sie den Lesezugriff für die gewünschte Gruppe.

- **Schreiben**
  Aktivieren oder deaktivieren Sie den Schreibzugriff für die gewünschte Gruppe.

  Hinweis

  Damit der Schreibzugriff funktioniert, müssen Sie ebenfalls den Lesezugriff aktivieren.

- **Persistenz**
  Zeigt an, ob die Gruppe einem SNMPv3-Benutzer zugeordnet ist. Wenn die Gruppe keinem SNMPv3-Benutzer zugeordnet ist, wird kein automatisches Speichern ausgelöst und die konfigurierte Gruppe ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Die Gruppe ist einem SNMPv3-Benutzer zugeordnet.
  - Nein
    Die Gruppe ist keinem SNMPv3-Benutzer zugeordnet.

v3-Benutzer

Benutzerspezifische Sicherheitseinstellungen

Auf der Seite können Sie SNMPv3-Benutzer neu anlegen, ändern oder löschen. Das benutzerbasierte Sicherheitsmodell arbeitet mit dem Konzept des Benutzernamens, d. h. jedes Telegramm wird mit einer Benutzerkennung versehen. Diesen Benutzernamen und die betreffenden Sicherheitseinstellungen überprüfen sowohl der Absender wie auch der Empfänger.

Beschreibung

- **Benutzername**
  Geben Sie einen frei wählbaren Benutzernamen ein. Nach der Datenübernahme können Sie den Namen nicht mehr ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Benutzername**
  Zeigt die angelegten Benutzer an.
- **Gruppenname**
  Wählen Sie die Gruppe aus, die dem Benutzer zugeordnet wird.
- **Authentifizierungsprotokoll**
  Legen Sie das Authentifizierungsprotokoll fest, für das ein Passwort hinterlegt werden soll. Folgende Einstellungen gibt es:
  - Keine
  - MD5
  - SHA
- **Verschlüsselungsprotokoll**
  Legen Sie fest, ob ein Passwort zur Verschlüsselung mit dem DES-Algorithmus hinterlegt werden soll. Nur aktivierbar, wenn auch ein Authentifizierungsprotokoll ausgewählt wurde.

- **Authentifizierungspasswort**
  Geben Sie das Authentifizierungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Authentifizierungspasswort bestätigen**
  Bestätigen Sie das Passwort durch die Wiederholung der Eingabe.
- **Verschlüsselungspasswort**
  Geben Sie Ihr Verschlüsselungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Verschlüsselungspasswort bestätigen**
  Bestätigen Sie das Verschlüsselungspasswort durch die Wiederholung der Eingabe.
- **Persistenz**
  Zeigt an, ob der Benutzer einer SNMPv3-Gruppe zugeordnet ist. Wenn der Benutzer keiner SNMPv3-Gruppe zugeordnet ist, wird kein automatisches Speichern ausgelöst und der konfigurierte Benutzer ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Der Benutzer ist einer SNMPv3-Gruppe zugeordnet.
  - Nein
    Der Benutzer ist keiner SNMPv3-Gruppe zugeordnet.

Systemzeit

Übersicht

Um die Systemzeit des Geräts einzustellen, gibt es unterschiedliche Methoden. Es kann immer nur eine Methode aktiv sein.

Wenn eine Methode aktiviert wird, dann wird automatisch die bisher aktivierte Methode deaktiviert.

Manuelle Einstellung

Manuelle Einstellung der Systemzeit

Auf dieser Seite stellen Sie das Datum und die Uhrzeit des Systems ein. Damit diese Einstellung verwendet wird, müssen Sie "Manuelle Zeiteinstellung" aktivieren.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Manuelle Zeiteinstellung**
  Aktivieren oder deaktivieren Sie die manuelle Zeiteinstellung. Wenn Sie die Option aktivieren, wird das Eingabefeld "Systemzeit" editierbar.
- **Systemzeit**
  Geben Sie Datum und Uhrzeit im Format "MM/DD/YYYY HH:MM:SS" ein.
- **PC-Zeit verwenden**
  Klicken Sie auf die Schaltfläche, um die Zeiteinstellung des PCs zu übernehmen.
- **Letzter Synchronisationszeitpunkt**
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
  Wenn keine Uhrzeitsynchronisation möglich war, enthält das Feld die Angabe "Datum/Zeit nicht eingestellt".

- **Letzter Synchronisationsmechanismus**
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde.
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm.
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. Die aktuelle Systemzeit sehen Sie oben rechts im Auswahlbereich des WBM. In dem Feld "Systemzeit" wird weiterhin die eingestellte Zeit angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.

SNTP-Client

Um die Systemzeit des Geräts einzustellen, gibt es unterschiedliche Methoden. Es kann immer nur eine Methode aktiv sein. Wenn eine Methode aktiviert wird, dann wird automatisch die bisher aktivierte Methode deaktiviert.

Das SNTP (Simple Network Time Protocol) dient zur Zeitsynchronisation im Netzwerk. Die entsprechenden Telegramme werden von einem SNTP-Server im Netz versendet.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SNTP-Client**
  Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über SNTP.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom Gerät empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.

- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **SNTP-Modus**
  Wählen Sie die Synchronisationsart aus. Folgende Synchronisierungsarten sind möglich:
  - Poll
    Wenn Sie diese Protokollart wählen, werden die Eingabefelder "SNTP-Server-Adresse", "Port des SNTP Servers" und "Poll-Intervall[s]" zur weiteren Konfiguration eingeblendet. Bei dieser Synchronisationsart ist das Gerät aktiv und sendet eine Zeitabfrage an den SNTP-Server.
  - Listen
    Bei dieser Synchronisationsart ist das Gerät passiv und empfängt SNTP-Telegramme, die die Uhrzeit liefern. Legen Sie für diesen Modus manuell die folgende Firewall-Regel an:
    Aktion: Accept
    Von: vlanX
    Nach: Gerät
    Quell-Port: *
    Ziel-Port: 123 (UDP)
- **Poll-Intervall[s]**
  Geben Sie den Zeitabstand zwischen zwei Zeitanfragen ein. In diesem Feld geben Sie das Abfrageintervall in Sekunden an. Mögliche Werte sind 16 bis 16284 Sekunden.
- **SNTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SNTP-Servers an.

- **Port des SNTP-Servers**
  Geben Sie den Port des SNTP-Servers ein.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564
- **Primär**
  Bei dem SNTP-Server, den Sie zuerst anlegen, wird das Häckchen gesetzt. Wenn mehrere SNTP-Server angelegt sind, wird der primäre Server zuerst angefragt.

NTP-Client

Wenn die Uhrzeitsynchronisation über NTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **NTP-Client**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem NTP-Server.
- **Nur NTP-Client (gesichert)**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem gesicherten NTP-Server. Die Einstellung gilt für alle Severeinträge.
  Um den gesicherten NTP-Client zu aktivieren, sind die Parameter für die Authentifizierung (Schlüssel-ID, Hash-Algorithmus, Schlüssel) zu konfigurieren. Auf dem Secure NTP-Server müssen diese Einträge vorhanden sein.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom Gerät empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.

- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **NTP-Serverindex**
  Wählen Sie den Index des NTP-Servers aus. Der Server mit dem kleinsten Index wird zuerst angefragt.
- **NTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des NTP-Servers an.
- **Port des NTP-Servers**
  Geben Sie den Port des NTP-Servers an.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564
- **Poll-Intervall**
  Legen Sie den Zeitabstand zwischen zwei Uhrzeitanfragen fest. Je größer der Zeitabstand, desto ungenauer ist die Uhrzeit des Geräts. Mögliche Werte sind 64 bis 2592000 Sekunden (30 Tage).
- **Schlüssel-ID**
  Geben Sie die ID des Authentifizierungsschlüssels ein.
- **Hash-Algorithmus**
  Legen Sie das Format für den Authentifizierungsschlüssel fest.
- **Schlüssel**
  Geben Sie den Authentifizierungsschlüssel ein. Die Länge ist abhängig vom Hash-Algorithmus.
  - DES: ASCII 8 Zeichen
  - MD5: ASCII 16 – 128 Zeichen
  - SHA1: ASCII 20 – 128 Zeichen

SIMATIC Time Client

Zeiteinstellung über SIMATIC Time Client

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den SIMATIC Time Client.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SIMATIC Time Client**
  Markieren Sie dieses Optionskästchen, um das Gerät als SIMATIC Time Client zu aktivieren.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemzeit an.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP

NTP-Server

Auf dieser Seite konfigurieren Sie das Gerät als NTP-Server. Die anderen Geräte können über diesen NTP-Server die aktuelle Zeit abrufen. Damit sind die versorgten Geräte nicht auf eine Verbindung zu einem externen Zeitserver angewiesen.

Voraussetzung

Damit die NTP-Telegramme empfangen werden, müssen Sie die folgende Firewall-Regel anlegen:

Aktion: Accept

Von: vlanX

Nach: Gerät

Quell-Port: *

Ziel-Port: 123 (UDP)

Beschreibung

Die Seite enthält folgende Felder:

- **NTP-Server**
  Aktivieren oder deaktivieren Sie den Dienst NTP-Server.

Die Tabelle 1 gliedert sich in folgende Spalten

- **An allen Schnittstellen**
  Zeigt an, dass die Einstellungen für alle Schnittstellen der Tabelle 2 gültig sind.
- **Mithören**
  Wählen Sie in der Klappliste die Einstellung für alle Schnittstellen aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Schnittstellen der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Schnittstelle**
  Über diese Schnittstelle wird die Zeit mittels NTP übermittelt.
- **Mithören**
  Wenn aktiviert, können die anderen Geräte über diese Schnittstelle die Uhrzeit abrufen.

Auto-Logout

Stellen Sie in dieser Seite die Zeiten ein, nach denen bei Inaktivität des Benutzers automatisch eine Abmeldung vom WBM oder dem CLI erfolgt.

Wenn Sie automatisch abgemeldet wurden, dann müssen Sie sich wieder neu anmelden.

Hinweis

Keine automatische Abmeldung vom CLI

Wenn die Verbindung nach der eingestellten Zeit nicht beendet wird, prüfen Sie am Telnet Client die Einstellung des "Keep alive"- Mechanismus.

Ist die Intervallzeit kleiner als die projektierte Zeit, wird die die Verbindung aufrecht erhalten, obwohl keine Nutzdaten übertragen werden. Z. B. Sie haben bei der automatischen Abmeldung 300 Sekunden eingestellt und bei der "Keep alive"- Funktion steht 120 Sekunden. In diesem Fall wird alle 120 Sekunden ein Paket gesendet, dass die Verbindung aufrecht erhält.

- Schalten Sie den Kepp-Alive-Mechanismus aus (Intervallzeit=0)
  oder
- Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung bei Inaktivität beendet wird.

Beschreibung

- **Web Based Management [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom WBM ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.
- **CLI (TELNET, SSH) [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom CLI ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.

Taster

Funktionalität SCALANCE S615

Der SET-Taster dient zum:

- Neustart
- Laden einer neuen Firmware,
- Zurücksetzen auf Werkseinstellungen.

Eine detaillierte Beschreibung der Funktionen, finden Sie in der Betriebsanleitung des Geräts.

Auf dieser Seite kann die Funktionalität des Tasters eingeschränkt werden.

Beschreibung SCALANCE S615

Folgende Funktionalität ist möglich:

- **Neustart / Auf Werkseinstellungen zurücksetzen**
  Wenn deaktiviert, kann der SET-Taster nicht zum Neustart oder zum Zurücksetzen auf Werkseinstellungen verwendet werden.

  ⚠ VORSICHT

  Tasterfunktion "Neustart / Auf Werkseinstellungen zurücksetzen" beim Hochlauf aktiv

  Wenn Sie diese Funktion in ihrer Projektierung deaktiviert haben, ist die Deaktivierung nur im laufenden Betrieb gültig. Bei einem Hochlauf, z.B. nach "Stromaus", ist die Funktion bis zum Laden der Projektierung aktiv und das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät dann neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferungszustand versetzt.

Funktionalität SCALANCE SC63x-2C/SC64x-2C

Der SET-Taster dient zum:

- Zurücksetzen auf Werkseinstellungen,
- Definieren der Meldemaske und der LED-Anzeige.

Eine detaillierte Beschreibung der Funktionen, finden Sie in der Betriebsanleitung des Geräts.

Auf dieser Seite kann die Funktionalität des Tasters eingeschränkt werden.

Beschreibung SCALANCE SC63x-2C/SC64x-2C

Folgende Funktionalität ist möglich:

- **Auf Werkseinstellungen zurücksetzen**
  Wenn deaktiviert, kann der SET-Taster nicht zum Zurücksetzen auf Werkseinstellungen verwendet werden.

  ⚠ VORSICHT

  Tasterfunktion "Auf Werkseinstellungen zurücksetzen" beim Hochlauf aktiv

  Wenn Sie diese Funktion in ihrer Projektierung deaktiviert haben, ist die Deaktivierung nur im laufenden Betrieb gültig. Bei einem Hochlauf, z.B. nach "Stromaus", ist die Funktion bis zum Laden der Projektierung aktiv und das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät dann neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferungszustand versetzt.

- **Meldemaske aktivieren**
  Aktivieren oder deaktivieren Sie die Funktion "Definieren der Meldemaske über die LED-Anzeige" am SELECT/SET-Taster.

Syslog-Client

Systemereignis-Agent

Syslog nach RFC 3164 wird für die Übermittlung von kurzen, unverschlüsselten Textmeldungen per UDP im IP-Netz verwendet. Dazu wird ein Syslog-Server benötigt.

Voraussetzungen für das Versenden von Log-Einträgen

- Die Syslog-Funktion ist im Gerät aktiviert.
- Die Syslog-Funktion für das jeweilige Ereignis ist aktiviert.
- In Ihrem Netz befindet sich ein Syslog-Server, der die Log-Einträge entgegen nimmt. Da es sich um eine UDP-Verbindung handelt, gibt es keine Rückmeldung an den Absender.
- Die IP-Adresse oder der FQDN (Fully Qualified Domain Name) des Syslog-Servers ist im Gerät eingetragen.

Beschreibung

Die Seite enthält folgende Felder:

- Syslog-Client
  Aktivieren oder deaktivieren Sie die Syslog-Funktion.
- Adresse des Syslog-Servers
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers ein.

Die Tabelle enthält folgende Spalten:

- Adresse des Syslog-Servers
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers an.
- Server-Port
  Geben Sie den verwendeten Port des Syslog-Servers ein.

Ports

Übersicht (SC-600)

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die konfigurierbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Port-Name**
  Zeigt den Namen des Ports.
- **Port-Typ** (Nur bei Routing)
  Zeigt den Typ des Ports an. Folgende Typen sind möglich:
  - Switch-Port VLAN Hybrid
  - Switch-Port VLAN Trunk
- **Combo Port Medientyp** (nur Online verfügbar)
  Diese Spalte enthält nur bei Combo Ports einen Wert.
  Zeigt den Modus des Combo Ports an:
  - auto
  - rj45
  - sfp
- **Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Betriebszustand** (nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig. Es gibt folgende Möglichkeiten:
  - Up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - Down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Akt. Übertragungsmodus** (nur Online verfügbar)
  Zeigt die Übertragungsparameter des Ports an.
- **MTU (Maximum Transmission Unit)**
  Zeigt die Paketgröße an.
- **Negotiation**
  Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **Flow Ctrl.-Typ**
  Gibt an, ob für den Port die Flusskontrolle aktiviert oder deaktiviert ist.
- **Flow Ctrl.**
  Gibt an, ob bei diesem Port die Flusskontrolle arbeitet.

- **MAC-Adresse** (nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Geblockt durch** (nur Online verfügbar)
  Zeigt an, warum sich der Port im Zustand "blocked" befindet:
  - -
    Der Port ist nicht geblockt.
  - Admin down
    Für den Port ist der Status "disabled" konfiguriert, siehe "System > Ports > Konfiguration".
  - Link down
    Für den Port ist der Status "enabled" konfiguriert, aber es besteht keine Verbindung, siehe "System > Ports > Konfiguration".
  - Power down
    Für den Port ist der Status "Link down" konfiguriert, siehe "System > Ports > Konfiguration".

Abweichende Anzeige der Übertragungsparameter bei Combo Ports

Im Verbindungsstatus „down“ entsprechen die angezeigten Übertragungsparameter nicht den eigentlichen Werten des Combo Ports. Im Verbindungsstatus "up" werden die korrekten Werte angezeigt.

Ausgangslage

Es ist ein Stecktransceiver am Combo Port gesteckt, mit folgenden Einstellungen:

- Combo Port Medientyp: auto
- Status: enabled
- Link: down

Anzeige der Übertragungsparameter

Bei 100 MBit/s-Stecktransceivern

- Ist-Verhalten: Mode: 100M HD
- Soll-Verhalten: Mode: 100M FD

Bei 1 GBit/s-Stecktransceivern

- Ist-Verhalten: Mode: 1G HD
- Soll-Verhalten: Mode: 1G FD

Konfiguration (SC-600)

Ports konfigurieren

Mit dieser Seite können Sie alle Ports des Geräts konfigurieren.

Beschreibung

- **Port**
  Wählen Sie in der Klappliste den zu konfigurierenden Port aus.
- **Status**
  Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled
    Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled
    Der Port ist ausgeschaltet, aber die Verbindung besteht noch.

    Hinweis

    Schalten Sie nicht genutzte Ports aus.

  - Link down
    Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**
  Tragen Sie hier einen Namen für den Port ein.
- **MAC-Adresse** (nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Übertragungsmodus**
  Wählen Sie aus dieser Klappliste die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports aus. Wenn Sie die Betriebsart auf "Auto negotiation" stellen, werden diese Parameter automatisch mit dem angeschlossenen oder der Netzkomponente ausgehandelt. Dieses muss sich hierzu ebenfalls in der Betriebsart "Auto negotiation" befinden.

  Hinweis

  Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

  Hinweis
  "Übertragungsmodus" bei Combo Ports

  Um den "Übertragungsmodus" eines Combo Ports einstellen zu können, ändern Sie den "Combo Port Medientyp" in "rj45". Wenn für den "Combo Port Medientyp" "auto" eingestellt ist und der RJ45-Port verwendet wird, können Sie den "Übertragungsmodus" nicht einstellen.

- **Akt. Übertragungsmodus** (nur Online verfügbar)
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Folgende Einstellungen sind möglich:
  - 10 Mbit/s Vollduplex (FD) oder Halbduplex (HD)
  - 100 Mbit/s Vollduplex (FD) oder Halbduplex (HD)
  - 1000 Mbit/s Vollduplex (FD)

- **Negotiation**
  Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.

  Hinweis

  Ein-/Ausschalten der Flusskontrolle bei Auto Negotiation

  Die Flusskontrolle kann nur bei ausgeschalteter Funktion "Auto Negotiation" aktiviert oder deaktiviert werden. Die Funktion kann danach wieder aktiviert werden.

- **Flow Ctrl.-Typ**
  Aktivieren oder deaktivieren Sie die Flusskontrolle für den Port.
- **MTU**
  Tragen Sie die Paketgröße ein.
- **Flow Ctrl.**
  Zeigt an, ob bei diesem Port die Flusskontrolle arbeitet.
- **Port-Typ**
  Wählen Sie aus der Klappliste die Art des Ports aus.
  - Switch-Port VLAN Hybrid
    Der Port sendet getaggte und ungetaggte Telegramme. Er ist nicht automatisch Mitglied eines VLANs.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.

- **Combo Port Medientyp**
  Legen Sie den Modus des Combo Ports fest:
  - auto
    Wenn Sie diesen Modus wählen, hat der Stecktransceiver-Port Priorität.
    Sobald ein Stecktransceiver gesteckt wird, wird eine bestehende Verbindung am festen RJ45-Port getrennt. Wenn kein Stecktransceiver gesteckt ist, kann eine Verbindung über den festen RJ45-Port hergestellt werden.
  - rj45
    Wenn Sie diesen Modus wählen, wird der feste RJ45-Port verwendet, unabhängig vom Stecktransceiver-Port.
    Wenn ein Stecktransceiver gesteckt ist, wird er deaktiviert und stromlos geschaltet.
  - sfp
    Wenn Sie diesen Modus wählen, wird der Stecktransceiver-Port verwendet, unabhängig vom festen RJ45-Port.
    Wenn eine RJ45-Verbindung besteht, wird diese getrennt, da der RJ45-Port stromlos geschaltet wird.

  Die Werkseinstellung für die Combo Ports ist der Modus auto.

  Hinweis

  Automatische Anpassung durch PROFINET-Konfiguration

  Beim Aufbau einer PROFINET-Verbindung wird auch der die Einstellung des Combo Port Medientyps automatisch angepasst:
  - Wenn eine Stecktransceiver konfiguriert ist, wird der Combo Port Medientyp auf "sfp" gesetzt.
  - Wenn der feste RJ45-Port konfiguriert ist, wird der Combo Port Medientyp auf "rj45" gesetzt.

  Damit die automatische Anpassung durchgeführt werden kann, muss der Combo Port Medientyp auf "auto" gesetzt sein.

  Konfigurieren Sie den Combo Port Medientyp entsprechend über das WBM oder CLI.

- **Betriebszustand** (nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig. Es gibt folgende Möglichkeiten:
  - Up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - Down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
  - not present
    Bei modularen Geräten wird dieser Status angezeigt, wenn z. B. kein Medienmodul gesteckt ist.

- **Geblockt durch** (nur Online verfügbar)
- Zeigt an, warum sich der Port im Zustand "blocked" befindet:
  - -
    Der Port ist nicht geblockt.
  - Admin down
    Für den Port ist der Status "disabled" konfiguriert, siehe "System > Ports > Konfiguration".
  - Link down
    Für den Port ist der Status "enabled" konfiguriert, aber es besteht keine Verbindung, siehe "System > Ports > Konfiguration".
  - Power down
    Für den Port ist der Status "Link down" konfiguriert, siehe "System > Ports > Konfiguration".

Veränderung der Port-Konfiguration

Klicken Sie in das entsprechende Feld, um die Konfiguration zu ändern.

Hinweis

Optische Ports arbeiten immer mit dem Übertragungsverfahren Vollduplex und mit maximaler Übertragungsgeschwindigkeit. Deshalb können Sie bei optischen Ports folgende Einstellungen nicht vornehmen:

- Automatische Konfiguration
- Übertragungsgeschwindigkeit
- Übertragungsverfahren

Hinweis

Das Gerät verhindert oder reduziert bei Überlastung eines Ports durch verschiedene Automatismen die Rückwirkung auf andere Ports und Prioritätsklassen (Class of Service). Dies kann auch bei aktivierter Flusskontrolle dazu führen, dass Telegramme verworfen werden.

Port-Überlastungen treten auf, wenn das Gerät mehr Telegramme empfängt, als es senden kann, z. B. infolge unterschiedlicher Übertragungsgeschwindigkeiten.

Fehlerkontrolle

Spannungsversorgung

Einstellungen zur Überwachung der Spannungsversorgung

Konfigurieren Sie, ob die Spannungsversorgung durch das Meldesystem überwacht werden soll. Bei redundanter Spannungsversorgung konfigurieren Sie die Überwachung für jede einzelne Zuleitung getrennt.

Es wird dann ein Fehler durch das Meldesystem signalisiert, wenn an einem überwachten Anschluss (Versorgung 1 oder Versorgung 2) keine oder eine zu geringe Spannung anliegt.

Hinweis

Die zulässigen Betriebsspannungsgrenzen entnehmen Sie der Kompaktbetriebsanleitung des Geräts.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät und kann abhängig von der Konfiguration einen Trap oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Aktivieren Sie das Kontrollkästchen vor dem entsprechenden Anschlussnamen, den Sie überwachen wollen, um die Überwachungsfunktion ein- oder auszuschalten.

Link Change

Konfiguration der Fehlerüberwachung von Zustandsänderungen bei Verbindungen

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

Bei aktivierter Verbindungsüberwachung wird ein Fehler signalisiert,

- wenn an einem Port ein Link vorhanden sein soll und dieser fehlt.
- oder wenn an dem Port kein Link vorhanden sein soll und ein Link erkannt wird.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Möglichkeiten:
  - "-" (Deaktiviert)
  - Up
  - Down
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Wählen Sie die Einstellung aus. Sie haben folgende Möglichkeiten:
  - Up
    Die Fehlerbehandlung wird beim Übergang in den aktiven Zustand des Ports ausgelöst. (Von "Link down" nach "Link up")
  - Down
    Die Fehlerbehandlung wird beim Übergang in den inaktiven Zustand des Ports ausgelöst.
    (Von "Link up" nach "Link down")
  - "-" (Deaktiviert)
    Die Fehlerbehandlung wird nicht ausgelöst.

PLUG

Konfiguration

Hinweis

Wenn keine Online-Verbindung zum Gerät besteht, ist nur das Kontrollkästchen "Firmware auf PLUG" (siehe unten) verfügbar.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.

Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt.

Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurückgesetzt.

Informationen über die Konfiguration des C-PLUG

Diese Seite liefert Detailinformationen über die Konfiguration, die im C-PLUG abgelegt ist. Darüber hinaus gibt es die Möglichkeit, den PLUG auf Werkseinstellungen zurückzusetzen oder mit einem neuen Inhalt zu versehen.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da die sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Beschreibung

Die Tabelle gliedert sich in folgende Zeilen:

- **Status**
  Zeigt den Status des PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Es ist ein PLUG mit einer gültigen und passenden Konfiguration im Gerät vorhanden.
  - NOT ACCEPTED
    Ungültige bzw. inkompatible Konfiguration auf dem gesteckten PLUG.
  - NOT PRESENT
    Im Gerät ist kein C-PLUG oder KEY-PLUG gesteckt.
  - FACTORY
    PLUG ist gesteckt und enthält keine Konfiguration. Dieser Status wird auch angezeigt, wenn der PLUG im Betrieb formatiert wurde.
  - MISSING
    Es ist kein PLUG gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
- **Gerätegruppe**
  Zeigt an, von welcher SIMATIC NET-Produktlinie der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Gerätetyp**
  Zeigt den Gerätetyp innerhalb der Produktlinie an, von dem der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.

- **Version der Konfiguration**
  Die Version der Konfigurationsstruktur. Diese Angabe betrifft die vom Gerät unterstützten Konfigurationsmöglichkeiten und hat nichts mit der konkreten Hardware-Konfiguration zu tun. Diese Revisionsangabe ändert sich also nicht, wenn Sie Zusatzkomponenten (z.B. Module bzw. Extender) hinzufügen oder entfernen, sie kann sich aber ändern, wenn Sie ein Firmware-Update durchführen.
- **Dateisystem**
  Zeigt den Typ des Dateisystems an, das auf dem PLUG vorhanden ist.
- **Verfügbarer Speicherplatz [Byte]**
  Zeigt die maximale Speicherkapazität des Dateisystems an, das auf dem PLUG vorhanden ist.
- **Belegter Speicherplatz [Byte]**
  Zeigt den belegten Speicherplatz im Dateisystem des PLUG an.
- **Firmware auf PLUG**
  Wenn aktiviert, wird die Firmware auf dem PLUG abgespeichert. Damit können mit dem PLUG automatische Firmware-Updates/Downgrades durchgeführt werden.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.
  Wenn ein PLUG als PRESET-PLUG konfiguriert wurde, wird dies hier als Zusatzinformation in der ersten Zeile angezeigt.
- **PLUG ändern**
  Wählen Sie die gewünschte Einstellung. Sie haben folgende Möglichkeiten, um die Konfiguration auf dem C-PLUG bzw. KEY-PLUG zu ändern:
  - Aktuelle Konfiguration auf den PLUG schreiben
    Diese Option ist nur verfügbar, wenn der Status des PLUG "NOT ACCEPTED" oder "FACTORY" ist.
    Die im internen Flash-Speicher des Gerätes vorhandene Konfiguration wird auf den PLUG kopiert.
  - PLUG auf Werkseinstellungen zurücksetzen
    Löscht alle Daten vom PLUG und führt eine Low-Level-Formatierung durch.

Lizenz

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

ACHTUNG
C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken! Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden. Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt. Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurück.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Informationen über die Lizenz des KEY-PLUG

Ein C-PLUG kann nur die Konfiguration eines Geräts speichern. Ein KEY-PLUG enthält zusätzlich zur Konfiguration eine Lizenz, die bestimmte Funktionen Ihres SIMATIC NET-Geräts freischaltet.

Diese Seite liefert Detailinformationen über die Lizenz auf dem KEY-PLUG.

Beschreibung

- **Status**
  Zeigt den Status des KEY-PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Der im Gerät vorhandene KEY-PLUG enthält eine passende und gültige Lizenz.
  - NOT ACCEPTED
    Die Lizenz des gesteckten KEY-PLUG ist nicht gültig.
  - NOT PRESENT
    Im Gerät ist kein KEY-PLUG gesteckt.
  - MISSING
    Es ist kein KEY-PLUG oder ein C-PLUG mit dem Status "FACTORY" gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
  - WRONG
    Der gesteckte KEY-PLUG passt nicht zum Gerät.
  - UNKNOWN
    Unbekannter Inhalt des KEY-PLUG.
  - DEFECTIVE
    Der Inhalt des KEY-PLUG ist fehlerhaft.
- **Artikelnummer**
  Zeigt die Artikelnummer des KEY-PLUG an. Es gibt den KEY-PLUG für unterschiedliche Funktionserweiterungen und für verschiedene Zielsysteme.
- **Seriennummer**
  Zeigt die Seriennummer des KEY-PLUG.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den KEY-PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.

Hinweis

Beim Speichern der Konfiguration wird die Information mitgespeichert, ob zu diesem Zeitpunkt ein KEY-PLUG im Gerät gesteckt war. Diese Konfiguration ist dann auch nur lauffähig, wenn ein KEY-PLUG mit der gleichen Artikelnummer / Lizenz gesteckt ist..

Port-Diagnose

SFP-Diagnose

Mit dieser Seite führen Sie für jeden einzelnen SFP-Port eine unabhängige Fehlerdiagnose durch. Dieser Test wird durchgeführt, ohne dass ein Kabel ausgesteckt, ein Kabeltester angeschlossen und am anderen Ende ein Loopback-Modul installiert werden muss.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

Die Seite enthält folgende Felder:

- **Port**
  Wählen Sie aus der Klappliste den gewünschten Port aus.

Die Werte werden in den folgenden Feldern angezeigt:

- **Name**
  Zeigt den Namen der Schnittstelle an.
- **Modell**
  Zeigt die Art der Schnittstelle an.
- **Ausgabestand**
  Zeigt den Hardware Ausgabestand des SFP an.
- **Seriell**
  Zeigt die Seriennummer des SFP an.
- **Nominale Bit-Rate [MBit/s]**
  Zeigt die Nenn-Bitrate der Schnittstelle an.
- **Max. Link (50.0/125um) [m]**
  Zeigt die maximale Distanz in Metern an, die mit diesem Medium möglich sind.
- **Max. Link (62.5/125um) [m]**
  Zeigt die maximale Distanz in Metern an, die mit diesem Medium möglich sind.

Die folgenden Felder zeigen die Werte des in diesem Port eingesetzten SFP-Stecktransceivers an:

- **Temperatur [°C]**
  Zeigt die Temperatur der Schnittstelle an.
- **Spannung [V]**
  Zeigt die an der Schnittstelle anliegende Spannung an [V].
- **Strom [mA]**
  Zeigt die Stromaufnahme der Schnittstelle an [mA].
- **Rx-Leistung [µW]/Rx-Leistung[dBm]**
  Zeigt die Empfangsleistung der Schnittstelle an [µW]/[dBm].

- **Tx-Leistung [µW]/Tx-Leistung[dBm]**
  Zeigt die Sendeleistung der Schnittstelle an [µW]/[dBm].
- **Bereich Aktuell**
  Zeigt den aktuellen Wert an.
- **Bereich Niedrig**
  Zeigt den niedrigsten Wert an.
- **Bereich Hoch**
  Zeigt den höchsten Wert an.

DNS

DNS-Client

Der DNS-Server (Domain Name System) ordnet einer IP-Adresse einen Domainnamen zu, sodass ein Gerät eindeutig identifiziert werden kann.

Wenn diese Einstellung aktiviert ist, kann das Gerät als DNS-Client mit einem DNS-Server kommunizieren.

Beschreibung

- **DNS-Client**
  Aktivieren oder deaktivieren Sie, dass das Gerät als DNS-Client fungiert.
- **Verwendete DNS-Server**
  - learned only
    Das Gerät verwendet nur die durch DHCP zugewiesenen DNS-Server.
  - manual only
    Das Gerät verwendet nur die manuell projektierten DNS-Server. Die DNS-Server müssen mit dem Internet verbunden sein.
  - all
    Das Gerät verwendet alle verfügbaren DNS-Server.
- **Adresse des DNS-Servers**
  Tragen Sie die IP-Adresse des DNS-Servers ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Adresse des DNS-Servers**
  Zeigt die IP-Adresse des DNS-Servers an.
- **Erstellung**
  Hier wird angezeigt, ob der DNS-Server manuell konfiguriert wurde oder durch DHCP zugewiesen wurde.

DNS-Proxy

Das Gerät stellt dem lokalen Netz einen DNS-Server zur Verfügung. Wenn Sie in der lokalen Anwendung die IP-Adresse des Gerätes als DNS-Server eintragen, beantwortet das Gerät DNS-Anfragen aus seinem Cache.

Wenn das Gerät die IP-Adresse zu einer Domain-Adresse nicht kennt, leitet es die Anfrage an einen externen DNS-Server weiter. Wie lange das Gerät eine Domain-Adresse im Cache behält, ist abhängig vom adressierten Host. Die DNS-Anfrage an einen externen DNS-Server liefert außer der IP-Adresse auch die Lebensdauer dieser Information zurück.

Beschreibung

Die Seite enthält folgende Felder:

- DNS-Proxy aktivieren
  Aktivieren oder deaktivieren Sie den Proxy des DNS-Servers.
- Cache Name Errors (NXDOMAIN)
  Aktivieren oder deaktivieren Sie das Zwischenspeichern von NXDOMAIN-Antworten. Wenn Sie die Option aktivieren, verbleiben auch die Domain-Namen im Cache, die dem DNS-Server unbekannt waren.

DDNS-Client

Der DDNS (Dynamic Domain Name System) ist ein Internetdienst, der es ermöglicht, einen festen Hostnamen als Pseudonym für eine sich dynamisch ändernde IP-Adresse einzurichten.

Der DDNS-Client synchronisiert die zugewiesene IP-Adresse mit dem im DDNS-Provider registrierten Hostnamen. Damit ist das Gerät immer unter demselben Hostnamen erreichbar.

Voraussetzung:

- Benutzernamen und Passwort, dass Sie zur Nutzung des DDNS-Dienstes berechtigt.
- Registrierter Hostname z. B. example.no-ip.com
- Der UDP-Port 53 für DNS ist freigeschaltet und wird nicht bei NAT verwendet.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- Dienst
  Zeigt an, welche Anbieter unterstützt werden.
- Aktiviert
  Wenn aktiviert, meldet sich das Gerät an dem DDNS-Server an.
- Host
  Tragen Sie den Hostnamen ein, den Sie für das Gerät mit Ihrem DDNS-Anbieter vereinbart haben, z. B. example.no-ip-com.
- Benutzername
  Tragen Sie den Benutzernamen ein, mit dem sich das Gerät am DDNS-Server anmeldet.
- Passwort
  Tragen Sie das dem Benutzer zugeordnete Passwort ein.
- Passwort bestätigen
  Bestätigen Sie das Passwort.

DHCP

DHCP-Client

Wenn das Gerät als DHCP-Client konfiguriert ist, startet es eine DHCP-Anfrage. Das Gerät erhält vom DHCP-Server als Antwort eine IPv4-Adresse zugewiesen. Der Server verwaltet einen Adressbereich, aus welchem er IPv4-Adressen vergibt. Es ist auch möglich, den Server so zu konfigurieren, dass der Client auf seine Anfrage immer dieselbe IPv4-Adresse zugewiesen bekommt.

Beschreibung

Die Seite enthält folgende Felder:

- **DHCP-Client Konfigurationsanfrage (Opt. 66, 67)**
  Wenn aktiviert, verwendet der DHCP-Client die Optionen dazu, die Konfigurationsdatei (Option 67) vom TFTP-Server (Option 66) herunterzuladen. Nach dem Neustart verwendet das Gerät die Daten aus der Konfigurationsdatei.

  Hinweis

  Konfigurationsdatei und Firmware-Version

  Die Konfigurationsdatei dient zum Abspeichern und Einlesen von Konfigurationsdaten innerhalb einer Firmware-Version z. B. 4.3. Konfigurationsdateien, die mit einer Firmware-Version <4.2 erstellt wurden, können nicht auf einem Gerät mit einer Firmware-Version 4.3 eingelesen werden.

  Hinweis

  Wird eine Konfigurationsdatei heruntergeladen, so kann dies einen Neustart des Systems auslösen. Wenn sich die aktuell laufende Konfiguration und die Konfiguration in der heruntergeladenen Konfigurationsdatei unterscheiden, startet das System neu.

  Achten Sie darauf, dass in dieser Konfigurationsdatei die Option "DHCP-Client Konfigurationsanfrage (Opt. 66, 67)" nicht mehr gesetzt ist.

- **DHCP-Modus**
  Legen Sie fest, mit welcher Art von Kennung sich der DHCP-Client bei seinem DHCP-Server anmeldet:
  - über MAC-Adresse
    Die Identifikation läuft über die MAC-Adresse ab.
  - über DHCP-Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab.
  - über Systemname
    Die Identifikation läuft über den Systemnamen ab. Ist der Systemname 255 Zeichen lang, dann wird das letzte Zeichen nicht zur Identifikation benutzt.
  - über IAID und DUID
    Damit kann sich der DHCP-Client an DHCP-Servern anmelden, die den Parallelbetrieb von IPv4-und IPv6 unterstützten.
    Die Identifikation läuft über die IAID und die DUID ab und bezeichnet genau eine IP-Schnittstelle des Geräts.
    IAID (Interface Association Identifier): Für jede IP-Schnittstelle wird mindestens eine IAID generiert. Die IAID bleibt bei Neustart des DHCP-Clients unverändert.
    DUID (DHCP Unique Identifier): Identifiziert Server und Clients eindeutig und gilt für alle IP-Schnittstellen des Geräts. Die DUID bleibt bei Neustart unverändert. Es sei denn, der Benutzer ändert diese.

- **DUID-Typ**
  Legen Sie fest, welcher DUID-Typ verwendet wird. Die DUID-Typen sind in der RFC 3315 definiert.
  - DUID-LLT
    DUID basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel
  - DUID-EN
    DUID wird vom Hersteller (EN = Enterprise number) vergeben
  - DUID-LL
    DUID basiert auf der Link-layer-Adresse der Schnittstelle
- **Link-layer Adresse plus Zeit (LLT)**
  Der Wert basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Unternehmensnummer des Herstellers (EN)**
  Der Wert basiert auf der Unternehmensnummer, die spezifisch für den Hersteller ist. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Link-layer Adresse (LL)**
  Die Link-layer-Adresse basiert auf der MAC-Adresse. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.
- **DHCP**
  Aktivieren oder deaktivieren Sie den DHCP-Client für die entsprechende Schnittstelle.
- **IAID-Wert**
  Wert mit dem sich die Schnittstelle (DHCP-Client) am DHCP-Server identifiziert.

DHCP-Server

Das Gerät können Sie als DHCP-Server betreiben. Damit ist es möglich, den angeschlossenen Geräten automatisch IP-Adressen zuzuweisen. Die IP-Adressen werden entweder dynamisch aus einem von Ihnen vergebenen Adressband (Pool) verteilt oder es wird eine bestimmte IP-Adresse einem bestimmten Gerät zugewiesen.

Auf dieser Seite legen Sie das Adressband fest, aus dem das Gerät eine beliebige IP-Adresse erhält. Die statische Zuordnung der IP-Adressen konfigurieren Sie unter "Statische Zuordnungen".

Hinweis

DHCP-Server-Zuordnungen löschen

Wenn Sie ein IPv4-Adressband deaktivieren bzw. löschen oder den DHCP-Server aus- und wieder einschalten, werden die DHCP-Server-Zuordnungen gelöscht, siehe "Information > DHCP-Server".

Der Aufbau dieser Seite ist davon abhängig, über wie viele VLAN-IP-Schnittstellen das Gerät verfügt.

Voraussetzung

Die angeschlossenen Geräte sind so konfiguriert, dass diese die IP-Adresse von einem DHCP-Server beziehen.

Geräte mit einer VLAN-IP-Schnittstelle

Auf dieser Seite legen Sie IPv4-Adressen fest, die über bestimmte Ports vergeben werden.

Beschreibung

Die Seite enthält folgende Felder:

- **DHCP-Server**
  Aktivieren oder deaktivieren Sie den DHCP-Server auf dem Gerät.

  Hinweis

  Damit keine Konflikte mit IPv4-Adressen entstehen, darf im Netzwek nur ein Gerät als DHCP-Server konfiguriert sein.

- **Adresse vor dem Anbieten mit ICMP-Echo prüfen**
  Wenn aktiviert, prüft der DHCP-Server, ob die IP-Adresse schon vergeben ist. Dazu sendet der DHCP-Server ICMP-Echomeldungen (Ping) an die IPv4-Adresse. Wenn keine Antwort zurückkommt, wird die IPv4-Adresse vergeben.

  Hinweis

  Bei statischen Zuordnungen wird diese Prüfung nicht durchgeführt.

  Hinweis

  Wenn es in Ihrem Netzwerk Geräte gibt, bei denen der Echo-Dienst standardmäßig deaktiviert ist, kann es zu Konflikten bei den IPv4-Adressen kommen. Um dies zu vermeiden, vergeben Sie diesen Geräten eine IPv4-Adresse, die außerhalb des IPv4-Adressbands liegt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an. Wenn Sie auf die Schaltfläche "Erstellen" klicken, wird eine neue Zeile mit einer eindeutigen Nummer (Pool-ID) angelegt.
- **Schnittstelle**
  Wählen Sie eine VLAN-IP-Schnittstelle aus. Über diese Schnittstelle werden die IPv4-Adressen dynamisch vergeben. Voraussetzung für die Vergabe ist, dass die IPv4-Adresse der Schnittstelle im Subnetz des IPv4-Adressbands liegt. Wenn das nicht der Fall ist, vergibt die Schnittstelle keine IPv4-Adressen.

- **Aktivieren**
  Legen Sie fest, ob dieses IPv4-Adressband verwendet wird.

  Hinweis

  Wenn Sie das IPv4-Adressband aktivieren, werden dessen Einstellungen in diesem sowie in den weiteren DHCP-Registern ausgegraut und sind nicht mehr editierbar.

- **Subnetz**
  Tragen Sie den Netzadressbereich ein, der den Geräten zugewiesen wird. Verwenden Sie die CIDR-Schreibweise.
- **Untere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die den Anfang des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Obere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die das Ende des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Gültigkeitsdauer [Sek]**
  Legen Sie fest, für wie viele Sekunden die vergebene IPv4-Adresse gültig bleibt. Nachdem die Gültigkeitsdauer zur Hälfte abgelaufen ist, kann der DHCP-Client die vergebene IPv4-Adresse verlängern. Nach Ablauf der gesamten Zeitdauer muss der DHCP-Client eine neue IPv4-Adresse anfordern.

Geräte mit mehreren VLAN-IP-Schnittstellen

Auf dieser Seite legen Sie das Adressband fest, aus dem das angeschlossene Gerät eine beliebige IP-Adresse erhält. Die statische Zuordnung der IP-Adressen konfigurieren Sie unter "Statische Zuordnung".

DHCP-Optionen

Auf dieser Seite legen Sie fest, welche DHCP-Optionen der DHCP-Server unterstützt. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte Adressband aus.
- **Optionswert**
  Geben Sie die Nummer der gewünschten DHCP-Option ein.

  Hinweis

  Unterstützte DHCP-Optionen

  SCALANCE S615: 1, 2, 3. 4. 5, 6, 42, 66, 67

  SCALANCE SC-600: 1, 3, 6, 12, 66, 67

  Die DHCP-Optionen 1, 3, 6, 66 und 67 werden automatisch beim Erstellen des IPv4-Adressbands angelegt. Mit Ausnahme der Option 1 sind die Optionen löschbar.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen
- **Pool-ID**
  Zeigt die Nummer des Adressbands an.
- **Optionswert**
  Zeigt Nummer der DHCP-Option an.
- **Beschreibung**
  Beschreibender Text zum Optionswert.

- **Schnittstellen-IP verwenden**
  Legen Sie fest, ob die interne IP-Adresse des Geräts verwendet wird oder nicht.
- **Wert**
  Geben Sie den DHCP-Parameter ein, der dem DHCP-Client übergeben wird. Der Inhalt ist abhängig von der DHCP-Option.

Wert	Optionsname		
1	Subnetzmaske	Die Subnetzmaske wird automatisch eingetragen.	Option nicht löschbar.
2	Offset-Zeit	Offset-Zeit zu der koordinierten Weltzeit UTC.	Geben Sie die Offset-Zeit in Sekunden im Hexadezimal-Format an.
3	Router	Die IPv4-Adresse für Router im Subnetz des DHCP-Clients. Wenn das Gerät selbst der Router ist, wird die IPv4-Adresse der Schnittstelle verwendet.	Sie können mehrere IPv4-Adressen durch Komma getrennt angeben.
4	Zeitserver	Die IPv4-Adresse des Zeitservers, die dem DHCP-Client zur Verfügung steht .	
5	Namenserver	Die IPv4-Adresse des Namesservers, die DHCP-Client zur Verfügung steht.	
6	DNS-Server	Die IPv4-Adresse des DNS-Servers, die dem DHCP-Client zur Verfügung steht. Wenn das Gerät selbst der DNS-Server ist, wird die IPv4-Adresse der Schnittstelle verwendet.	
12	Host-Name	Geben Sie den Hostnamen im String-Format an.	
42	NTP-Server	Die IPv4-Adresse des NTP-Servers, die dem DHCP-Client zur Verfügung stehen .	
66	TFTP-Server	Die IPv4-Adresse oder der Hostnamen des TFTP-Servers, die dem DHCP-Client zur Verfügung steht.	Geben Sie die Adresse des TFTP-Servers an.
67	Namen der Boodatei	Der Namen der Boodatei, die der Client vom TFTP-Server herunterlädt.	Geben Sie den Namen der Bootdatei im String-Format an.

Statische Zuordnung

Auf dieser Seite legen Sie fest, dass bestimmten Geräten eine bestimmte IP-Adresse zugewiesen wird. Die Adresszuordnung erfolgt anhand der MAC-Adresse, anhand der Client-ID oder anhand der DUID.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte Adressband aus.
- **Identifikationsmethode des Clients**
  Wählen Sie die Methode, nach der ein Client identifiziert wird.
  - Ethernet MAC
    Die Identifikation läuft über die MAC-Adresse ab. Tragen Sie bei "Wert" die MAC-Adresse ein. Die MAC-Adresse besteht aus sechs Bytes, die, durch Bindestriche getrennt, hexadezimal notiert werden, z. B. 00-ab-1d-df-b4-1d.
  - Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab. Tragen Sie bei "Wert" die gewünschte Bezeichnung ein.
  - DUID
    Die Identifikation läuft über DUID und IAID ab. Tragen Sie bei "Wert" die gewünschte Bezeichnung ein, z. B. 00-00-01-C2-00-01-00-01-00-00-00-72-00-1B-1B-B6-32-9D.
- **Wert**
  Tragen Sie den gewünschten Wert ein. Die Eingabe ist abhängig von der gewählten Identifikationsmethode des Clients.

Hinweis

Maximal 20 Einträge sind möglich.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des Adressbands an.
- **Identifikationsmethode**
  Zeigt an, mit welcher Methode sich der Client am DHCP-Server identifiziert.
- **Wert**
  Zeigt die MAC-Adresse, die Client-ID oder DUID des Clients an.
- **IP-Adresse**
  Legen Sie die IPv4-Adresse fest, die dem Client zugewiesen wird. Die IPv4-Adresse muss innerhalb des Adressbands liegen.

cRSP/SRS

Hinweis

Common Remote Service Platform (cRSP) / Siemens Remote Service (SRS) ist eine Fernwartungsplattform über die der Fernwartungszugriff durchgeführt wird.

Zur Nutzung der Plattform sind zusätzliche Serviceverträge notwendig und Randbedingungen zu beachten. Bei Interesse an cRSP / SRS wenden Sie sich an Ihren Siemens-Ansprechpartner vor Ort und besuchen Sie folgende Webseite (http://www.industry.siemens.com/topics/global/de/service/remote-service/seiten/home.aspx).

Auf dieser Seite konfigurieren Sie die Zugangsdaten für SRS / cRSP nach der URI-Syntax. Der Uniform Resource Identifier (URI) ist in der RFC 3986 definiert.

Beschreibung

Die Seite enthält folgende Felder:

- **DDNS für cRSP / SRS aktivieren**
  Aktivieren oder deaktivieren Sie die Nutzung von cRSP / SRS.
- **Update-Intervall**
  Geben Sie die Zeitspanne ein.
- **Serverzertifikat überprüfen**
  Wenn aktiviert, überprüft das Gerät das empfangene Serverzertifikat auf Gültigkeit.

Die Tabelle gliedert sich in folgende Spalten:

- **Index**
  Die Nummer des Eintrags.
- **Schema**
  Identifiziert die Zugriffsmethode und den Ressourcentyp.
  https: Gesicherter Zugriff auf eine Webseite.
- **Authority**
  Enthält die Adresse des Zielservers
- **Pfad**
  Enthält den Zielpfad zur Ressource. Der Zielpfad kann einem Verzeichnisnamen oder Dateinamen entsprechen.
- **Abfrage**
  Eine Anfrage kann Parameterwerte für eine Anwendung enthalten.
- **Frag**.
  Adressiert lokale Teile der Ressource, z. B. das Anker-Attribut einer Webseite.
- **Status** (nur Online verfügbar)
  Zeigt den Status des letzten cRSP / SRS-Zugriffs des Eintrages an.
- **Aktiviert**
  Wenn aktiviert, wird dieser Eintrag verwendet.

Proxy-Server

Auf dieser Seite konfigurieren Sie den Proxy-Server, der von verschiedenen Komponenten verwendet wird, z. B. SINEMA RC.

Beschreibung

- **Proxy-Name**
  Tragen Sie einen Namen für den Proxy-Server ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen des Proxy-Servers an.
- **Adresse**
  Tragen Sie die IPv4-Adresse des Proxy-Servers ein.
- **Typ**
  Legen Sie die Art des Proxy-Servers fest.
  - HTTP: Proxy-Server nur für Zugriffe über HTTP.
  - SOCKS: Universeller Proxy-Server
- **Port**
  Geben Sie den Port ein, auf dem der Proxydienst läuft.
- **Auth.-Methode**
  Legen Sie die Authentifizierungsmethode fest.
  - None
    Ohne Authentifizierung
  - Basic
    Standardauthentifizierung. Benutzernamen und Passwort werden unverschlüsselt gesendet.
  - NTML (NT LAN Manager)
    Authentifizierung nach NTML Standard (Windows-Benutzeranmeldung)
- **Benutzername**
  Geben Sie den Benutzernamen für den Zugang zum Proxy-Server ein.
- **Passwort**
  Geben Sie das Passwort für den Zugang zum Proxy-Server ein.
- **Passwort bestätigen**
  Geben Sie nochmals das Passwort ein, um es zu bestätigen.

SINEMA RC

Auf dieser Seite konfigurieren Sie den Zugriff zum SINEMA RC-Server.

Hinweis

Für SCALANCE S615 ist diese Funktion nur mit KEY PLUG nutzbar.

Beschreibung

Die Seite enthält Folgendes:

- SINEMA RC aktivieren
  - Aktiviert:
    Eine Verbindung zum konfigurierten SINEMA RC-Server wird aufgebaut. Die Felder sind nicht editierbar.
  - Deaktiviert:
    Die Felder lassen sich editieren. Eine eventuell bestehende Verbindung wird abgebaut.

Bereich "Server-Einstellungen"

- SINEMA RC-Adresse
  Geben Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des SINEMA RC-Servers ein.
- SINEMA RC-Port
  Geben Sie den Port ein, über den der SINEMA RC-Server erreichbar ist.

Bereich "Serverüberprüfung"

- Prüfungsart
  - Fingerabdruck: Die Identität des Servers wird über den Fingerabdruck verifiziert.
  - CA-Zertifikat: Die Identität des Servers wird über das CA-Zertifikat verifiziert
- Fingerabdruck
  Nur bei der Einstellung "Fingerabdruck" notwendig. Geben Sie den Fingerabdruck des Geräts ein. Der Fingerabdruck wird bei der Inbetriebnahme des SINEMA RC-Servers vergeben. Anhand des Fingerabdrucks überprüft das Gerät, ob es sich um den korrekten SINEMA RC-Server handelt. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- CA-Zertifikat
  Nur bei der Einstellung "CA-Zertifikat" notwendig. Wählen Sie das CA-Zertifikat des Servers aus, das zur Signierung des Serverzertifikats verwendet wird. Nur geladene CA-Zertifikate sind auswählbar.

Bereich "Geräteanmeldedaten"

- Geräte-ID
  Geben Sie die Geräte-ID ein. Die Geräte-ID wird beim Konfigurieren des Geräts am SINEMA RC-Server vergeben. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- Geräte-Passwort
  Geben Sie das Passwort ein, mit dem sich das Gerät am SINEMA RC-Server anmeldet. Das Passwort wird beim Konfigurieren des Geräts am SINEMA RC-Server vergeben. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- Gerät Passwort Bestätigung
  Wiederholen Sie das Passwort ein, um es zu bestätigen.

Bereich "Optionale Einstellungen"

- Auto Firewall/NAT-Regeln
  - Aktiviert
    Für die VPN-Verbindung werden automatisch die Firewall und NAT-Regeln angelegt. Dabei werden die Verbindungen, die zwischen den projektierten exportierten Subnetzen und den Subnetzen, die über den SINEMA RC-Server erreichbar sind, zugelassen. Die NAT-Einstellungen werden wie im SINEMA RC-Server projektiert umgesetzt.
  - Deaktiviert
    Sie müssen selbst die Firewall und NAT-Regeln anlegen.
- Verbindungsart
  Legen Sie die Art der VPN-Verbindung fest. Weitere Informationen dazu finden Sie im Kapitel "VPN-Verbindungsaufbau".
  - Auto
    Das Gerät übernimmt die Einstellungen des SINEMA RC Server. Die Einstellungen auf dem SINEMA RC Server konfigurieren Sie unter "Fernverbindungen > Geräte". Weiterführende Informationen hierzu finden Sie in der Betriebsanleitung "SINEMA RC Server".
  - Permanent
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Das Gerät baut eine VPN-Verbindung zum SINEMA RC-Server. Der VPN-Tunnel wird permanent aufrechterhalten.
  - Digitaler Eingang
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Beim Eintreten des Ereignisses "Digital In" versucht das Gerät zum SINEMA RC-Server eine VPN-Verbindung aufzubauen. Vorausgesetzt ist, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
- Proxy verwenden
  Legen Sie fest, ob die Verbindung zu dem definierten SINEMA RC-Server über einen Proxy-Server aufgebaut wird. Es sind nur die Proxy-Server auswählbar, die Sie unter "System > Proxy Server" konfiguriert haben.
- Autokonfigurations-Intervall [min]
  Geben Sie die Zeitspanne in Minuten an, nach der Anfragen an den SINEMA RC Server gesendet werden. Mit dieser Anfrage prüft das Gerät, ob auf dem SINEMA RC Server eine neuere Firmware-Datei vorhanden ist bzw. ob sich die Konfigurationsdaten geändert haben.
  Wenn Sie den Wert 0 eintragen, ist diese Funktion deaktiviert.

Schnittstellen projektieren

Ethernet

Übersicht (S615)

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an.

Beschreibung

- **Port**
  Zeigt die konfigurierbaren Ports an.
- **Port-Name**
  Zeigt den Namen des Ports.
- **Port-Typ** (Nur bei Routing)
  Zeigt den Typ des Ports an. Folgende Typen sind möglich:
  - Router-Port
  - Switch-Port VLAN Hybrid
  - Switch-Port VLAN Trunk
- **Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Betriebszustand** (Nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.
- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsparameter des Ports an
- **Negotiation**
  Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.

Konfiguration (S615)

Auf dieser Seite konfigurieren Sie die Ethernet-Ports des Geräts.

Beschreibung

- **Port**
  Wählen Sie die zu konfigurierenden Port aus. .
- **Status**
  Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled
    Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled
    Der Port ist ausgeschaltet, aber die Verbindung besteht noch.
  - Link down
    Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**
  Tragen Sie hier einen Namen für den Port ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Übertragungsmodus**
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Wenn Sie die Betriebsart auf "Auto negotiation" stellen, werden diese Parameter automatisch mit dem angeschlossenen Endgerät oder der Netzkomponente ausgehandelt. Dieses muss sich hierzu ebenfalls in der Betriebsart "Auto negotiation" befinden.

  Hinweis

  Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Die Anzeige ist abhängig von dem eingestellten "Mode Type".
- **Negotiation**
  Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.
- **Port-Typ** (Nur bei Routing)
  Wählen Sie die Art des Ports aus:
  - Router-Port
    Der Port ist eine IP-Schnittstelle. Er unterstützt keine Layer 2-Funktionen.
  - Switch-Port VLAN Hybrid
    Der Port sendet getaggte und ungetaggte Telegramme. Er ist nicht automatisch Mitglied eines VLANs.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.

- **Betriebszustand** (Nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - Up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - Down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.

PPP

Übersicht

Die Seite zeigt den akutellen Status der PPP-Verbindung an.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die PPP-Schnittstelle. Der Eintrag ist ein Link. Wenn Sie auf den Link klicken, wird die entsprechende Konfigurationsseite geöffnet.
- **Name**
  Zeigt den Namen der PPP-Schnittstelle an.
- **Typ**
  Zeigt das Protokoll der PPP-Verbindung an.

- **Betrieb**
  Zeigt an, ob die PPP-Verbindung aktiviert oder deaktiviert ist.
- **Status**
  Zeigt den Status der PPP-Verbindung an.
  - Bereit
    Die PPP-Verbindung kann konfiguriert und aktiviert werden.
  - Stellt Verbindung her
    Die PPP-Verbindung ist konfiguriert, aktiviert und der Verbindungsaufbau läuft.
  - Verbunden
    Die PPP-Verbindung ist aufgebaut.
  - Fehler
    Fehlerzustand, in dem ein Benutzereingriff nötig ist, z. B. falsches Passwort.

Konfiguration

Auf dieser Seite konfigurieren Sie die PPP-Verbindung. Das Punkt-zu-Punkt-Protokoll (PPP) erlaubt den Anschluss eines externen ADSL-Modems an eine Ethernet-Schnittstelle und darüber dann die Verbindung ins Internet. Die Schnittstelle wird auch als PPP-Schnittstelle bezeichnet.

Das Gerät agiert als Router und meldet sich mit Benutzernamen und Passwort an. Alle angeschlossenen Geräte können die PPP-Verbindung nutzen.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle**
  Wählen Sie die zu konfigurierende PPP-Schnittstelle aus.
- **Name**
  Zeigt den Namen der PPP-Schnittstelle an. Den Namen können Sie unter "Layer 3 > Subnets" ändern.
- **Typ**
  Legen Sie das Protokoll für die PPP-Verbindung fest.
  - PPPoE (Point-to-Point over Ethernet)
    Die PPP-Daten wird in einen Ethernet-Frame eingekapselt.
- **Betrieb**
  Legen Sie fest, ob die PPP-Verbindung aktiviert oder deaktiviert ist.
- **L2-Schnittstelle**
  Legen Sie fest, über welche Schnittstelle die PPP-Verbindung aufgebaut wird. Nur VLANs mit konfigurierten Subnetz auswählbar.
- **Benutzername**
  Geben Sie den Benutzernamen ein. Den Benutzernamen erhalten Sie vom DSL-Anbieter.
- **Passwort**
  Geben Sie das Passwort ein. Das Password erhalten Sie vom DSL-Anbieter.

- **Passwort bestätigen**
  Wiederholen Sie das Passwort.
- **Zwangstrennung**
  Der DSL-Anbieter trennt nach einem bestimmten Zeitraum die Verbindung. Aktivieren Sie diese Option, wenn Sie die Zwangstrennung durch Ihren Provider auf eine bestimmte Uhrzeit verschieben wollen, z. B. Nachts außerhalb der üblichen Bürozeiten.
- **Zeit für Zwangstrennung**
  Legen Sie die Uhrzeit fest zu der Sie die Zwangstrennung des DSL-Anbieters verschieben wollen. Vorausgesetzt, im Gerät ist die korrekte Systemzeit eingestellt.
  Eingabeformat: HH:MM

Layer 2-Funktionen projektieren

Konfiguration

Layer 2 konfigurieren

Auf dieser Seite konfigurieren Sie Passive Listening.

Beschreibung

- **Passive Listening**
  Wenn aktiviert, sorgt die Funktion dafür, dass die BPDUs aus dem RSTP-Netzwerk transparent weitergeleitet werden und auch wieder zurückgelangen. Wäre dies nicht der Fall, würde es zur Schleifenbildung an der Verbindungsstelle zwischen RSTP und Ring kommen.

VLAN

Allgemein

Auf dieser Seite legen Sie fest, ob das Gerät Telegramme mit VLAN-Tags transparent weiterleitet (IEEE 802.1D/VLAN-unaware-Modus) oder VLAN-Informationen berücksichtigt (IEEE 802.1Q/VLAN-aware-Modus). Wenn sich das Gerät im Modus "802.1Q VLAN Bridge" befindet, können Sie VLANs definieren und die Verwendung der Ports festlegen.

Hinweis

Ändern der Agent VLAN-ID

Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN-ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

Beschreibung

Die Seite enthält folgende Felder:

- **Base Bridge-Modus**
  Wählen Sie den gewünschten Modus aus. Folgende Modi sind möglich:

  Hinweis

  Base Bridge-Modus wechseln

  Beachten Sie den Abschnitt "Base Bridge-Modus wechseln". In diesem Abschnitt ist beschrieben, wie sich ein Wechsel auf die bestehende Konfiguration auswirkt.

  – 802.1 D Transparent Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLAN-Tags nicht verändert, sondern transparent weitergeleitet. Die VLAN-Priorität wird für CoS ausgewertet. Sie können in diesem Modus keine VLANs anlegen. Es ist nur ein Management-VLAN verfügbar: VLAN 1.

  – 802.1 Q VLAN Bridge
    Stellt bei dem Gerät den Modus "VLAN-aware" ein. In diesem Modus werden VLAN-Informationen berücksichtigt.

- **VLAN-ID**
  Tragen Sie die VLAN-ID (eine Zahl zwischen 1 und 4094) ein.

Die Tabelle gliedert sich in folgende Spalten:

- **VLAN-ID**
  Zeigt die VLAN ID an. Die VLAN ID (eine Zahl zwischen 1 und 4094) kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.

- **Name**
  Tragen Sie einen Namen für das VLAN ein. Der Name hat nur informativen Charakter und keine Auswirkungen auf die Konfiguration.

- **Status**
  Zeigt die Statusart des Eintrags in der internen Portfiltertabelle internen Portfiltertabellean. Dabei bedeutet "Static", dass das VLAN vom Anwender statisch eingetragen wurde. Die Angabe GVRP bedeutet, dass die Konfiguration über ein GVRP-Telegramm registriert wurde. Voraussetzung dafür ist allerdings, dass GVRP für das Gerät aktiviert wurde.
- **Liste der Ports**
  Legen Sie die Verwendung des Ports fest. Folgende Möglichkeiten gibt es:
  - "-"
  - Der Port ist kein Mitglied des angegebenen VLANs.
    Bei der Neudefinition sind alle Ports mit der Kennung "-" belegt.
  - M
    Der Port ist Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden mit dem entsprechenden VLAN-Tag weitergeleitet.
  - R
    Der Port ist Mitglied des VLANs. Die Registrierung erfolgt über ein GVRP-Telegramm.
  - U (Großbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet. Von diesem Port werden Telegramme ohne VLAN-Tag gesendet.
  - u (Kleinbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs, jedoch ist das VLAN nicht als Port-VLAN konfiguriert. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet.
  - F
    Der Port ist kein Mitglied des angegebenen VLANs und es ist nicht möglich, dass das VLAN dynamisch über GVRP an diesem Port registriert werden kann. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".
  - T
    Diese Option wird nur angezeigt und kann nicht ausgewählt werden.
    Dieser Port ist Trunk-Port und wurde dadurch Mitglied in allen VLANs.

Base Bridge-Modus wechseln

VLAN-unaware (802.1D Transparent Bridge) → VLAN-aware (802.1Q VLAN Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-unaware in VLAN-aware ädern, hat dies folgende Auswirkungen:

- Alle statischen und dynamischen Unicast-Einträe werden gelöcht.

VLAN-aware (802.1Q VLAN Bridge) → VLAN-unaware (802.1D Transparent Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-aware in VLAN-unaware ädern, hat dies folgende Auswirkungen:

- Alle VLAN-Konfigurationen werden gelöcht.
- Es wird ein Management-VLAN angelegt: VLAN 1.
- Alle statischen und dynamischen Unicast-Einträe werden gelöcht.

802.1Q VLAN Bridge: Wichtige Regeln für VLANs

Berücksichtigen Sie bei der Konfiguration und beim Betrieb Ihrer VLANs folgende Regeln:

- Telegramme mit der VLAN ID "0" werden wie ungetaggte Telegramme behandelt, behalten jedoch ihren Prioritäswert.
- Alle Ports am Gerä senden standardmäßg Telegramme ohne VLAN-Tag, um sicher zu gehen, dass der Endteilnehmer diese Telegramme empfangen kann.
- Bei SCALANCE W-Geräen ist an allen Ports die VLAN ID "1" voreingestellt.
- Wenn an einem Port ein Endteilnehmer angebunden ist, dann sollen ausgehende Telegramme ohne Tag versendet werden (statischer Zugriffs-Port). Wenn sich an dem Port ein weiterer Switch befindet, so ist das Telegramm mit einem Tag zu versehen (Trunk Port).

Port-basiertes VLAN

Verarbeitung empfangener Telegramme

Auf dieser Seite legen Sie die Konfiguration der Port-Eigenschaften für den Telegrammempfang fest.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Alle Ports**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Priorität / Port-VID / Erlaubte Telegrammtypen / Ingress Filterung**
  Wählen Sie in der Klappliste die Einstellung für alle Ports aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorität**
  Wählen Sie die Priorität aus, mit der ungetaggte Telegramme versehen werden.
  Die CoS-Priorität (Class of Service), die im VLAN-Tag verwendet wird. Wird ein Telegramm ohne Tag empfangen, wird ihm diese Priorität zugeordnet. Diese Priorität legt fest, wie dieses Telegramm im Vergleich zu anderen Telegrammen weiterhin bearbeitet wird. Es gibt insgesamt acht Prioritäten, mit den Werten 0 bis 7, wobei 7 der höchsten Priorität entspricht (IEEE 802.1p Port Priority).

- **Port-VID**
  Wählen Sie die VLAN-ID aus. Nur die VLAN-IDs sind wählbar, die Sie auf der Seite "VLAN > Allgemein" definiert haben.
  Wenn ein empfangenes Telegramm kein VLAN-Tag hat, so wird es um ein Tag mit der hier angegebenen VLAN-ID ergänzt und entsprechend den Regeln am Port gesendet.
- **Erlaubte Telegrammtypen**
  Legen Sie fest, welche Arten von Telegrammen akzeptiert werden. Es gibt folgende Alternativen:
  - Nur getraggte Frames
    Das Gerät verwirft alle ungetaggten Telegramme. Telegramme, die mit "0" getagged sind, werden wie ungetaggte Telegramme behandelt. Das Gerät leitet alle getaggten Telegramme weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration.
  - Alle
    Das Gerät leitet alle Telegramme weiter
- **Ingress Filterung**
  Legen Sie fest, ob die VID von empfangenen Telegrammen ausgewertet wird. Folgende Möglichkeiten gibt es:
  - Aktiviert
    Die VLAN-ID empfangener Telegramme bestimmt die Weiterleitung: Für die Weiterleitung eines VLAN-getaggten Telegramms muss der Empfangsport Mitglied im gleichen VLAN sein. Am Empfangsport werden Telegramme aus unbekannten VLANs verworfen.
  - Deaktiviert
    Alle Telegramme werden weitergeleitet.

Dynamic MAC Aging

Das Gerät lernt automatisch die Quelladressen der angeschlossenen Teilnehmer.

Diese Information wird dazu benutzt, um Datentelegramme gezielt an die betroffenen Teilnehmer weiterzuleiten. Die Netzlast für die anderen Teilnehmer wird reduziert.
Erhält ein Gerät innerhalb einer bestimmten Zeitspanne kein Telegramm, dessen Quelladresse mit einer gelernten Adresse übereinstimmt, dann wird die gelernte Adresse gelöscht. Dieser Mechanismus wird als "Aging" bezeichnet. Durch Aging wird verhindert, dass Telegramme fehlgeleitet werden, wenn z.B. ein Endgerät (beispielsweise ein Programmiergerät) an einen anderen Port angeschlossen wird. Wenn die Option nicht aktiviert ist, löscht ein Gerät gelernte Adressen nicht automatisch.

Beschreibung

- **Dynamic MAC Aging**
  Aktivieren oder deaktivieren Sie die Funktion zum automatischen Aging von gelernten MAC-Adressen:
- **Aging Time[s]**
  Tragen Sie die Zeitspanne in Sekunden in 15er-Schritten ein. Nach dieser Zeitspanne wird eine gelernte Adresse gelöscht, wenn das Gerät keine weiteren Telegramme von dieser Absenderadresse mehr empfängt.

Hinweis

Rundungen der Werte, Abweichung vom Sollwert

Beachten Sie bei der Eingabe der Aging Time, dass auf korrekte Werte gerundet wirde. Wenn Sie einen Wert eingeben, der nicht durch 15 teilbar ist, wird der Wert automatisch abgerundet.

Spanning Tree

Allgemein

Dies ist die Basisseite zu Spanning Tree. Standardmäßig ist Rapid Spanning Tree aktiviert.

Beschreibung

Die Seite enthält folgende Felder:

- **Spanning Tree**
  Aktivieren oder deaktivieren Sie Spanning Tree.
- **Protokollkompatibilität**
  Folgende Einstellung gibt es:
  - RSTP

ST Allgemein

Die Seite besteht aus folgenden Teilen:

- Der linke Teil der Seite zeigt die Konfiguration des Geräts.

Beschreibung

Die Seite enthält folgende Felder:

- **Bridge-Priorität / Root-Priorität** (Nur Online verfügbar)
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 32768
- **Bridge-Adresse** (Nur Online verfügbar)**/ Root-Adresse** (Nur Online verfügbar)
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Port** (Nur Online verfügbar)
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten** (Nur Online verfügbar)
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge
- **Topologieänderungen** (Nur Online verfügbar) **/ Letzte Topologieänderung** (Nur Online verfügbar)
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit des letzten Hochlaufs. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz "Sek." hinter der Zahlenangabe
  - Minuten: Zusatz "Min." hinter der Zahlenangabe
  - Stunden: Zusatz "Std." hinter der Zahlenangabe
- **Bridge Hello Time[s] / Root Hello Time[s]** (Nur Online verfügbar)
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei Konfigurationstelegrammen ist die "Hello Time".
- **Bridge Forward Delay[s] / Root Forward Delay[s]** (Nur Online verfügbar)
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort angewendet, sondern erst nach dem im Parameter "Forward Delay" festgelegten Zeitraum. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.
- **Bridge Max Age[s] / Root Max Age[s]** (Nur Online verfügbar)
  Wenn die BPDU älter ist als das angegebene "Max Age", wird sie verworfen.
- **Zähler zurücksetzen** (Nur Online verfügbar)
  Klicken Sie auf diese Schaltfläche, um die Zähler auf dieser Seite zurückzusetzen.

ST-Port

In der Tabelle wird beim Aufruf der Seite der aktuelle Stand der Konfiguration der Port-Parameter angezeigt.

Klicken Sie zur Konfiguration in die entsprechenden Felder der Port-Tabelle.

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Alle Ports**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Spanning Tree-Status**
  Wählen Sie in der Klappliste die Einstellung für alle Ports aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an.
- **Spanning Tree-Status**
  Legen Sie fest, ob der Port im Spanning-Tree integriert ist oder nicht.

  Hinweis

  Wenn Sie die Option "Spanning Tree Status" für einen Port deaktivieren, kann es zur Schleifenbildung kommen. Die Topologie muss beachtet werden.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur ausgewertet, wenn die Pfadkosten gleich sind.
  Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240.
  Der Standardwert ist 128.
- **Kalk. Kosten**
  Tragen Sie die Wegekostenberechnung ein. Wenn Sie den Wert "0" eintragen, wird im Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.

- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn im Feld "Kalk. Kosten" der Wert "0" ist, wird der automatisch ermittelte Wert angezeigt. Wenn ein Wert ungleich "0" eingetragen wird, wird der Wert des Feldes "Kalk. Kosten" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht parametriert werden. Der Parameter "Status" ist abhängig von dem projektierten Protokoll. Folgende Werte sind möglich:
  - Disabled
    Der Port empfängt nur und nimmt nicht am STP, MSTP und RSTP teil.
  - Discarding
    In der Betriebsart "Discarding" werden BPDU-Telegramme empfangen. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    In diesem Status werden sowohl BPDU-Telegramme empfangen als auch gesendet. Der Port ist in den Spanning Tree-Algorithmus einbezogen.
  - Learning
    Vorstufe zum Status "Forwarding", der Port lernt aktiv die Topologie (d. h. die Teilnehmeradressen).
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz, er empfängt und sendet Datentelegramme.
- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Wechsel vom Status "Discarding" zum Status "Forwarding" an.

- **Edge Type**
  Legen Sie die Art des "Edge Port" fest. Sie haben folgende Möglichkeiten:
  - "-"
    Edge Port ist deaktiviert. Der Port wird wie ein "no Edge Port" behandelt.
  - Admin
    Wählen Sie diese Option, wenn sich an diesem Port immer ein Endgerät befindet. Sonst wird bei jeder Verbindungsänderung eine Rekonfiguration des Netzwerks ausgelöst.
  - Auto
    Wählen Sie diese Option, wenn an diesem Port automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist. Beim ersten Verbindungsaufbau wird der Port wie ein "no Edge Port" behandelt.
  - Admin/Auto
    Wählen Sie diese Optionen, wenn Sie an diesem Port eine Kombination aus beidem betreiben. Beim ersten Verbindungsaufbau wird der Port als "Edge Port" behandelt.
- **Edge** (Nur Online verfügbar)
  Zeigt an, in welchem Status der Port ist.
  - Aktiviert
    An diesem Port befindet sich ein Endgerät.
  - Deaktiviert
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.

  Bei einem Endgerät kann ein Switch ohne Rücksicht auf Spanning Tree-Telegramme schneller den Port umschalten. Wird entgegen dieser Einstellung ein Spanning Tree-Telegramm empfangen, wechselt der Port automatisch auf die Einstellung "Deaktiviert".
- **P.t.P.-Typ**
  Wählen Sie in der Klappliste die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.
  - "-"
    Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
  - P.t.P.
    Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
  - Shared Media
    Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

    Hinweis

    Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.
- **P.t.P.** (Nur Online verfügbar)
  - Aktiviert
    Zeigt an, dass eine Punkt zu Punkt Verbindung besteht.
  - Deaktiviert
    Zeigt an, dass keine Punkt zu Punkt Verbindung besteht

LLDP

Bestimmung der Netzwerktopologie

LLDP (Link Layer Discovery Protocol) ist im Standard IEEE 802.AB definiert.

LLDP ist ein Verfahren zur Bestimmung der Netzwerktopologie. Netzwerkkomponenten tauschen über LLDP Informationen mit ihren Nachbargeräten aus.

Netzwerkkomponenten, die LLDP unterstützen, verfügen über einen LLDP-Agenten. Der LLDP-Agent versendet in periodischen Abständen Informationen über sich selbst und empfängt Informationen von angeschlossenen Geräten. Die empfangenen Informationen werden in der MIB gespeichert.

Anwendungen

PROFINET benutzt das LLDP-Protokoll für die Topologie-Diagnose. In der Werkseinstellung ist LLDP für alle Ports aktiviert, d. h. es werden LLDP-Telegramme auf allen Ports gesendet und empfangen.

Die gesendeten Informationen werden auf jedem LLDP-fähigen Gerät in einer LLDP-MIB-Datei gespeichert. Netzwerkmanagementsysteme können auf diese LLDP-MIB-Dateien mit Hilfe von SNMP zugreifen und damit die vorliegende Netzwerktopologie nachbilden. Ein Administrator kann auf die Weise z. B. feststellen, welche Netzwerkkomponenten miteinander verbunden sind und auftretende Störungen lokalisieren.

Auf dieser Seite haben Sie die Möglichkeit das Aussenden und/oder Empfangen pro Port ein- oder auszuschalten.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- 1. Spalte
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- Einstellung
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Legen Sie fest, ob der Port LLDP-Telegramme senden oder empfangen soll. Folgende Möglichkeiten gibt es:
  - Rx
    Dieser Port kann LLDP-Telegramme nur empfangen.
  - Tx
    Dieser Port kann LLDP-Telegramme nur senden.
  - Rx & Tx
    Dieser Port kann LLDP-Telegramme empfangen und senden.
  - "-" (Deaktiviert)
    Dieser Port kann LLDP-Telegramme weder empfangen noch senden.

Layer 3-Funktionen für IPv4 projektieren

Statische Routen

Statische Route

Auf dieser Seite legen Sie fest, über welche Routen ein Datenaustausch zwischen den verschiedenen Subnetzen stattfinden kann. Dynamische Routingprotokolle werden nicht unterstützt, z. B. RIP, OSPF.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Zielnetzwerk**
  Tragen Sie die Netzwerkadresse des Ziels ein, das über diese Route erreichbar ist.
- **Subnetzmaske**
  Tragen Sie die dazugehörende Subnetzmaske ein.
- **Gateway**
  Tragen Sie die IPv4-Adresse des Gateways ein, über den diese Netzwerkadresse erreichbar ist.

- **Schnittstelle**
  Wählen Sie die Schnittstelle der Route aus.
- **Administrative Distanz**
  Tragen Sie die administrative Distanz für die Route ein. Die administrative Distanz entspricht der Güte einer Verbindung, z. B. Geschwindigkeit, Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt.
  Wenn Sie nicht eintragen, wird automatisch "nicht verwendet" eingetragen. Die administrative Distanz ist nachträglich änderbar.
  Wertebereich: 1 - 254 oder -1 für "not used". Dabei ist 1 der Wert für die bestmögliche Route.

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Netzwerkadresse des Ziels an.
- **Subnetzmaske**
  Zeigt die dazugehörende Subnetzmaske an.
- **Gateway**
  Zeigt die IPv4-Adresse des nächsten Gateways an.
- **Schnittstelle**
  Zeigt die Schnittstelle der Route an.
- **Administrative Distanz**
  Tragen Sie die administrative Distanz für die Route ein.
- **Status** (Nur Online verfügbar)
  Zeigt an, ob die Route aktiv ist oder nicht.

Subnetze

Übersicht

Die Seite zeigt die Subnetze für die ausgewählte Schnittstelle. Ein Subnetz bezieht sich immer auf eine Schnittstelle und wird im Eintrag "Konfiguration" angelegt.

Beschreibung

Die Seite enthält folgendes Feld:

- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus, an die Sie ein weiteres Subnetz projektieren.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an.
- **TIA-Schnittstelle**
  Zeigt die ausgewählte TIA-Schnittstelle an.
- **Schnittstellenname**
  Zeigt den Namen der Schnittstelle.

- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse an.
- **IP-Adresse**
  Zeigt die IPv4-Adresse des Subnetzes an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske.
- **Adresstyp**
  Zeigt den Adressentyp an. Folgende Werte sind möglich:
  - Primär
    Die erste IPv4-Adresse, die auf einer IPv4-Schnittstelle konfiguriert wurde.
  - Sekundär
    Alle weiteren IPv4-Adressen, die auf einer IPv4-Schnittstelle konfiguriert wurden.
- **Methode der IP-Adresszuweisung**
  Zeigt an, wie die IPv4-Adresse zugeordnet wird. Folgende Werte sind möglich:
  - Statisch
    Die IPv4-Adresse ist statisch. Tragen Sie die Einstellungen bei "IP-Adresse" und "Subnetzmaske" ein.
  - Dynamisch (DHCP)
    Das Gerät bezieht eine dynamische IPv4-Adresse von einem DHCPv4-Server.

- **Status der Erkennung von Adresskollisionen** (Nur Online verfügbar)
  Wenn neue IPv4-Adressen im Netz aktiv werden, prüft die Funktion "Erkennung von Adresskollisionen ", ob es zu Adresskollisionen kommen kann. Dadurch werden IPv4-Adressen erkannt, die doppelt vergeben werden sollen.

  Hinweis

  Die Funktion führt keine zyklische Prüfung durch.

  Diese Spalte zeigt an, in welchem Status sich die Funktion befindet. Folgende Werte sind möglich:

  – Idle
    Die Schnittstelle ist nicht aktiv und besitzt keine IPv4-Adresse.
  – Starting
    Dieser Status bezeichnet die Anlaufphase. In dieser Phase sendet das Gerät zunächst eine Anfrage, ob es die geplante IPv4-Adresse bereits gibt. Wenn die Adresse noch nicht vergeben ist, sendet das Gerät die Mitteilung, dass es ab jetzt diese IP-Adresse verwendet.
  – Conflict
    Die Schnittstelle ist nicht aktiv. Die Schnittstelle versucht eine IPv4-Adresse zu verwenden, die bereits vergeben ist.
  – Defending
    Die Schnittstelle verwendet eine eindeutige IPv4-Adresse. Eine andere Schnittstelle versucht die gleiche IPv4-Adresse zu verwenden.
  – Active
    Die Schnittstelle verwendet eine eindeutige IPv4-Adresse. Es gibt keine Kollisionen.
  – Not supported
    Die Funktion zur Erkennung von Adresskollisionen wird nicht unterstützt.
  – Disabled
    Die Funktion zur Erkennung von Adresskollisionen ist deaktiviert.

- **MTU**
  Zeigt die Paketgröße an.

Konfiguration

Auf dieser Seite konfigurieren Sie das Subnetz für die Schnittstelle.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle (Name)**
  Wählen Sie aus der Klappliste die Schnittstelle aus.
- **Schnittstellenname**
  Tragen Sie den Namen für die Schnittstelle ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse der ausgewählten Schnittstelle an.

- **DHCP / Dynamisch**
  Legen Sie fest, ob die Schnittstelle die IP-Adresse dynamisch bezieht.
  - DHCP: Die VLAN-Schnittstelle bezieht die IPv4-Adresse von einem DHCP-Server
  - Dynamisch: Die PPP-Schnittstelle bezieht die IP-Adresse vom Provider.

  Hinweis

  Wenn Sie das Gerät als Router mit mehreren Schnittstellen betreiben wollen, deaktivieren Sie DHCP auf allen Schnittstellen.

- **IP-Adresse**
  Tragen Sie die IPv4-Adresse der Schnittstelle ein. Die IPv4-Adressen dürfen nicht mehrfach verwendet werden.
- **Subnetzmaske**
  Tragen Sie die Subnetzmaske des zu erstellenden Subnetzes ein. Subnetze an unterschiedlichen Schnittstellen dürfen sich nicht überlappen.
- **Broadcast-IP-Adresse**
  Wenn eine bestimmte IP-Adresse als Broadcast-IP-Adresse des Subnetzes verwendet werden soll, dann tragen Sie diese ein. Ansonsten wird die letzte IP-Adresse des Subnetzes verwendet.
- **Adresstyp**
  Zeigt den Adresstyp an. Folgende Werte sind möglich:
  - Primär
    Das erste Subnetz der Schnittstelle.
  - Sekundär
    Alle weiteren Subnetze der Schnittstelle.
- **TIA-Schnittstelle**
  Wählen Sie aus, ob diese Schnittstelle zur TIA-Schnittstelle werden soll.
  Für die TIA-Schnittstelle gelten folgende Bedingungen:
  - Nur Schnittstellen mit dem Adresstyp "Primär" können als TIA-Schnittstelle aktiviert werden.
  - Es muss immer eine TIA-Schnittstelle geben.
  - Es kann nur eine TIA-Schnittstelle geben.
  - Eine TIA-Schnittstelle ist immer eine VLAN-Schnittstelle.
- **MTU**
  Mit MTU (Maximum Transmission Unit) wird die maximale Größe des Pakets festgelegt. Wenn die Pakete größer sind, als die eingestellte MTU, werden Sie fragmentiert.

NAT

Masquerading

Auf dieser Seite aktivieren Sie die Regeln für IP-Masquerading.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht. Nur Schnittstellen mit konfiguriertem Subnetz sind verfügbar.
- **Masquerading aktivieren**
  Wenn aktiviert, wird bei jedem ausgehenden Datenpaket, das über diese Schnittstelle gesendet wird, die Quell-IP-Adresse durch die IP-Adresse der Schnittstelle ersetzt.

NAPT

Auf dieser Seite konfigurieren Sie die Portweiterleitung.

Beschreibung

Die Seite enthält folgende Felder:

- **Quell-Schnittstelle**
  Wählen Sie die Schnittstelle aus, auf der die Anfragen eintreffen sollen.
- **Traffic-Typ**
  Legen Sie fest, für welches Protokoll die Adresszuordnung gültig ist.
- **Schnittstellen-IP der Quell-Schnittstelle verwenden**
  Wenn aktiviert, wird bei "Ziel-IP-Adresse" die IP-Adresse der ausgewählten Schnittstelle verwendet.
- **Ziel-IP-Adresse**
  Geben Sie die Ziel-IP-Adresse ein. An dieser IP-Adresse werden die Telegramme empfangen. Nur editierbar, wenn "Schnittstellen-IP der Quell-Schnittstelle verwenden" deaktiviert ist.
- **Ziel-Port**
  Geben Sie den Ziel-Port ein. Eingehende Telegramme mit diesem Port als Ziel-Port werden weitergeleitet. Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.

- **Ziel-IP-Umsetzung**
  Geben Sie die IP-Adresse des Teilnehmers an, an den dieses Telegramm weitergeleitet wird.
- **Ziel-Port Umsetzung**
  Geben Sie die Nummer des Ports ein. Das ist der neue Ziel-Port, an den das eingehende Telegramm weitergeleitet wird. Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  Wenn der "Ziel-Port" und der "Ziel-Port Umsetzung" gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.

Hinweis

Wenn der Port bereits durch einen lokalen Dienst z. B. Telnet belegt ist, wird eine Warnmeldung ausgegeben.
Vermeiden Sie auf jeden Fall die Nutzung von folgenden Ports:

- Telnet / SSH: TCP-Port 23
- HTTP: TCP-Port 80
- HTTPS: TCP-Port 443
- SNMP: UDP-Port 161
- ISAKMP: UDP-Port 500
- IPsec Nat-T: UDP-Port 4500
- NTP: UDP-Port 123

Die Tabelle gliedert sich in folgende Spalten:

- **Quell-Schnittstelle**
  Zeigt die Schnittstelle an, von dem die Pakete kommen müssen. Nur diese Pakete kommen für eine Portweiterleitung in Frage.
- **Traffic-Typ**
  Zeigt an, für welches Protokoll die Adresszuordnung gilt.
- **Schnittstellen-IP**
  Zeigt an, ob die IP-Adresse der Schnittstelle verwendet wird.
- **Ziel-IP**
  Zeigt die Ziel-IP-Adresse an. An dieser IP-Adresse werden die Telegramme empfangen.
- **Ziel-Port**
  Zeigt den Ziel-Port an. Eingehende Telegramme mit diesem Port als Ziel-Port werden weitergeleitet.
- **Ziel-IP-Umsetzung**
  Zeigt die IP-Adresse Teilnehmers an, an dem die Pakete weitergeleitet werden.
- **Ziel-Port Umsetzung**
  Zeigt an, auf welchen Ziel-Port übersetzt wird.

Source-NAT

Auf dieser Seite konfigurieren Sie die Regeln für Source-NAT.

Hinweis

Firewallregeln bei Source-NAT

Wenn Sie für eine Source-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Quell-IP-Subnetz". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz".

Beschreibung

- **Quell-Schnittstelle / Ziel-Schnittstelle**
  Legen Sie die Richtung des Verbindungsaufbaus fest. Es werden nur Verbindungen berücksichtigt, die in dieser festgelegten Richtung aufgebaut werden.
  Zur Auswahl stehen auch die virtuellen Schnittstellen von VPN-Verbindungen:
  - VLANx: VLANs mit konfigurierten Subnetz
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung

  Hinweis

  Wenn Sie eine NAT-Adressumsetzung in oder aus Richtung VPN-Tunnel konfigurieren, sind nur noch die beteiligten IP-Adressen der NAT-Adressumsetzungsregeln über VPN-Tunnel erreichbar.

- **Quell-IP-Adressen**
  Legen Sie fest, für welche Quell-IP-Adressen diese Source-NAT-Regel gültig ist. Nur die Pakete werden berücksichtigt, die der eingegebenen Adressen entsprechen.
  Folgende Eingaben sind möglich:
  - IP-Adresse: Gilt genau für die angegebene IP-Adresse.
  - IP-Adressbereich: Gilt für einen bestimmten IP-Adressbereich: Start-IP-Adresse "-" End-IP-Adresse an, z. B. 192.168.100.10 - 192.168.100.20
  - IP-Subnetz: Gilt für mehrere IPv4-Adressen, die zu einem IP-Adressbereich zusammengefasst werden : IP-Adresse/Anzahl Bits des Netzanteils (CIDR-Notation)
- **Schnittstellen-IP der Ziel-Schnittstelle verwenden**
  Wenn aktiviert, wird bei "Quell-IP-Umsetzung" die IP-Adresse der ausgewählten Ziel-Schnittstelle verwendet.

- **Quell-IP-Umsetzung**
  Geben Sie die IP-Adresse ein, mit der die IP-Adresse des Absenders ersetzt wird. Nur editierbar, wenn "Schnittstellen-IP der Ziel-Schnittstelle verwenden" deaktiviert ist.
- **Ziel-IP-Adressen**
  Legen Sie fest, für welche Ziel-IP-Adressen diese Source-NAT-Regel gültig ist. Nur die Pakete werden berücksichtigt, deren Ziel-IP-Adresse im Bereich der eingegebenen Adressen liegt.
  - IP-Adresse: Gilt genau für die angegebene IP-Adresse.
  - IP-Adressbereich: Gilt für einen bestimmten IP-Adressbereich: Start-IP-Adresse "-" End-IP-Adresse an, z. B. 192.168.100.10 - 192.168.100.20
  - IP-Subnetz: Gilt für mehrere IPv4-Adressen, die zu einem IP-Adressbereich zusammengefasst werden: IP-Adresse/Anzahl Bits des Netzanteils (CIDR-Notation)

Die Tabelle gliedert sich in folgende Spalten:

- Quell-Schnittstelle
  Zeigt die Quell-Schnittstelle an.
- Ziel-Schnittstelle
  Zeigt die Ziel-Schnittstelle an.
- Quell-IP-Adressen
  Zeigt die IP-Adressen der Absender an, für die eine Adressumsetzung gewünscht ist.
- Quell-IP-Umsetzung
  Zeigt die IP-Adresse an, mit der die IP-Adresse der Absender ersetzt wird.
- Ziel-IP-Adressen
  Zeigt die IP-Adressen der Empfänger an, für die eine Adressumsetzung gewünscht ist.

NETMAP

Auf dieser Seite legen Sie die Regeln für NETMAP fest. NETMAP ist ein statisches 1:1-Mapping von Netzwerkadressen, wobei der Hostanteil erhalten bleibt. Weitere Informationen dazu finden Sie im Kapitel "NAT und Firewall".

Hinweis

Firewallregeln bei Source-NAT

Wenn Sie für eine Source-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Quell-IP-Subnetz". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz".

Firewallregeln bei Destination-NAT

Wenn Sie für eine Destination-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Quell-IP Subnetz Umsetzung". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz Umsetzung".

Beschreibung

- **Typ**
  Legen Sie die Art der Adressumsetzung fest.
  - Quelle: Ersetzen der Quell-IP-Adresse
  - Ziel: Ersetzen der Ziel-IP-Adresse.
- **Quell-Schnittstelle**
  Legen Sie die Quell-Schnittstelle fest.
  - VLANx: VLANs mit konfigurierten Subnetz
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung
  - ppp2: WAN-Schnittstelle
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- **Ziel-Schnittstelle**
  Legen Sie die Ziel-Schnittstelle fest.
  - VLANx: VLANs mit konfigurierten Subnetz
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung
  - ppp2: WAN-Schnittstelle
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- **Quell-IP-Subnetz**
  Tragen Sie das Subnetz des Absenders ein.
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- **Quell-IP-Subnetz Umsetzung**
  Tragen Sie das Subnetz ein, mit der das Subnetz des Absenders ersetzt wird. Nur editierbar, bei den Einstellungen "SourceNAT".
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- **Ziel-IP-Subnetz**
  Tragen Sie das Subnetz des Empfängers ein.
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- **Ziel-IP-Subnetz Umsetzung**
  Tragen Sie Subnetz ein, mit der das Subnetz des Empfängers ersetzt wird. Nur editierbar, bei den Einstellungen "DestinationNAT".
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.

Die Tabelle gliedert sich in folgende Spalten:

- Typ
  Zeigt die Richtung der Adressumsetzung an.
- Quell-Schnittstelle
  Zeigt die Quell-Schnittstelle an.
- Ziel-Schnittstelle
  Zeigt die Ziel-Schnittstelle an.
- Quell-IP-Subnetz
  Zeigt das Subnetz des Absenders an.
- Quell-IP-Subnetz Umsetzung
  Zeigt das Subnetz des Absenders an, mit der das Subnetz des Absenders ersetzt wird.
- Ziel-IP-Subnetz
  Zeigt das Subnetz des Empfängers an.
- Ziel-IP-Subnetz Umsetzung
  Zeigt das Subnetz des Empfängers an, mit der das Subnetz des Empfängers ersetzt wird.

VRRPv3

Router

Einleitung

Auf dieser Seite legen Sie neue virtuelle Router an. Weitere Parameter konfigurieren Sie unter "Layer 3 (IPv4) > VRRPv3 > Konfiguration".

Hinweis

- Sie können VRRPv3 an VLAN-Schnittstellen nutzen. Router-Ports werden nicht unterstützt.

Voraussetzung

Damit die eingehenden VRRP-Pakete an das Gerät weitergeleitet werden, ist folgende Firewallregel zu konfigurieren:

Security > Firewall > IP-Protokoll:

- Protokollname: "VRRP“
- Protokollnummer: 112

Security > Firewall > IP-Regeln:

- Protokoll: IPv4
- Aktion: Accept
- Von: <Schnittstelle>
- Nach: Gerät

- Quelle (Bereich): 0.0.0.0/0
- Ziel (Bereich): 224.0.0.18/32
- Dienste: VRRP

Beschreibung

- **VRRPv3**
  Aktivieren oder deaktivieren Sie das Routing über VRRPv3.
- **Antworte an virtuellen Schnittstellen auf Ping-Anfragen**
  Wenn aktiviert, antworten auch die virtuellen IPv4-Adressen auf den Ping.
- **VRID-Tracking**
  Aktivieren oder deaktivieren Sie das VRID-Tracking. Wenn aktiviert, werden alle VRRP-Instanzen überwacht.
  Wenn sich der Status einer VRRP-Instanz nach "Initialize" ändert, wird die Priorität aller VRRP-Instanzen auf den Wert "1" verringert.
  Wenn sich der Status der VRRP-Instanz ändert, wird die ursprüngliche Priorität aller VRRP-Instanzen wiederhergestellt.
- **Schnittstelle**
  Wählen Sie die gewünschte VLAN-Schnittstelle aus, die als virtueller Router fungiert.
- **VRID**
  Geben Sie die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die IPv4-Schnittstelle an, die als virtueller Router fungiert.
- **VRID**
  Zeigt die ID des virtuellen Routers an.
- **Virtuelle MAC-Adresse** (Nur Online verfügbar)
  Zeigt die virtuelle MAC-Adresse des virtuellen Routers an.
- **Primäre IP-Adresse**
  Zeigt die kleinste numerische IPv4-Adresse auf diesem VLAN an. Der Eintrag 0.0.0.0 bedeutet, dass die "Primäre" Adresse auf diesem VLAN verwendet wird. Ansonsten sind alle IPv4-Adressen, die auf diesem VLAN in dem Menü "Layer 3 > Subnetze" konfiguriert wurden, gültige Werte.

- **Router-Status** (Nur Online verfügbar)
  Zeigt den derzeitigen Zustand des virtuellen Routers an. Mögliche Werte sind:
  - Master
    Der Router ist der Master-Router und übernimmt die Routingfunktionalität für alle zugeordneten IPv4-Adressen.
  - Backup
    Der Router ist der Backup-Router. Wenn der Master-Router ausfällt, übernimmt der Backup-Router die Aufgaben des Master-Routers.
  - Initialize
    Der virtuelle Router wurde soeben eingeschaltet. In kurzer Zeit wird er in den Zustand "Master" oder "Backup" wechseln.
- **Master IP-Adresse** (Nur Online verfügbar)
  Zeigt die IPv4-Adresse des Master-Routers an.
- **Priorität** (Nur Online verfügbar)
  Zeigt die Priorität des virtuellen Routers an.
  Wenn eine IPv4-Adresse dem VRRP-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- **Advert.-Intervall**
  Zeigt das Intervall an, in dem der Master-Router VRRPv3-Pakete verschickt.
- **Unterbrechen**
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - ja
    Dieser Router ist beim Rollenwechsel bevorzugt.
  - nein
    Dieser Router ist beim Rollenwechsel nicht bevorzugt.

VRRP und DHCP-Server

Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, muss der DHCP-Server auf dem Master-Router konfiguriert sein. Backup-Router reagieren nicht auf DHCP-Anfragen. Stellen Sie sicher, dass der Master-Router statisch konfiguriert ist und nach einem Ausfall wieder zum Master der VRRP-Gruppe wird.

Konfiguration

Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle / VRID**
  Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Primäre Adresse**
  Wählen Sie die vorrangige IPv4-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IPv4-Adresse.

  Hinweis

  Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist dann 0.0.0.0.
  Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IPv4-Adresse als Quelladresse für VRRP-Pakete genutzt wird, wählen Sie die entsprechende IPv4-Adresse aus. Ansonsten wird die numerisch kleinste IPv4-Adresse verwendet.

- **Master**
  Wenn aktiviert, wird die numerisch kleinste IPv4-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die numerisch kleinste IPv4-Adresse des VRRPv3-Routers als virtuelle IP-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IPv4-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**
  Gegen Sie die Priorität dieses virtuellen Routers ein. Gültige Werte sind 1-254.
  Wenn eine IPv4-Adresse dem VRRPv3-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRPv3-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRPv3-Router zum "Master".
- **Advertisement-Intervall**
  Geben Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRPv3-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**
  Erlauben sie die Bevorzugung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.
- **VRRP-Kompatibilitätsmodus**
  Wenn aktiviert, sendet und empfängt der VRRPv3-Router für konfigurierte IPv4-Adressen zusätzlich zu den VRRPv3-Telegrammen auch VRRPv2-Telegramme. Nur notwendig, wenn nicht alle VRRP-Router VRRPv3 unterstützen
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Priorität verringern**
  Geben Sie ein, um welchen Wert die Priorität der VRRP-Schnittstelle verringert wird.
- **Aktuelle Priorität**
  Zeigt die Priorität der VRRP-Schnittstelle an, nachdem die überwachte Schnittstelle in den Zustand "down" gewechselt hat.

Adressübersicht

Übersicht

Diese Seite zeigt an, welche IPv4-Adressen der virtuelle Router überwacht. Jeder virtueller Router kann maximal 10 IPv4-Adressen überwachen.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstelle
  Zeigt die Schnittstelle an, der als virtueller Router fungiert.
- VRID
  Zeigt die ID dieses virtuellen Routers an.
- Anzahl der Adressen
  Zeigt die Anzahl der IPv4-Adressen an.
- Zugeordnete IP-Adresse (1) ...Zugeordnete IP-Adresse (10)
  Zeigt die Router IPv4-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IPv4-Adressen durch diesen Router übernommen.

Adresskonfiguration

Anlegen oder ändern der überwachten IP-Adressen

Auf dieser Seite können Sie die zu überwachenden IPv4-Adressen anlegen, ändern oder löschen.

Einstellungen

Die Seite enthält Folgendes:

- Schnittstelle / VRID
  Wählen Sie die ID des virtuellen Routers aus.
- Zugeordnete IP-Adresse
  Geben Sie die IPv4-Adresse ein, die der virtuelle Router überwachen soll.

Die Tabelle gliedert sich in folgende Spalten:

- Zugeordnete IP-Adresse
  Zeigt die IPv4-Adressen an, die der virtuelle Router überwacht.

Schnittstellenüberwachung

Einleitung

Auf dieser Seite konfigurieren Sie die Überwachung von Schnittstellen.

Wenn sich der Link einer überwachten Schnittstelle von "up" nach "down" ändert, wird die Priorität der zugeordneten VRRP-Schnittstelle verringert. Um welchen Wert die Priorität verringt wird, konfigurieren Sie auf der Seite "Layer 3 > VRRP/VRRPv3 > Konfiguration".

Wenn sich der Link der Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstelle wiederhergestellt.

Beschreibung

Die Seite enthält folgende Felder:

- Schnittstelle
  Wählen Sie die gewünschte Schnittstelle aus, die überwacht werden soll.
- Track-ID
  Geben Sie eine Track-ID ein.
- Track-ID
  Wählen Sie eine Track-ID aus.
- Track-Schnittstellenzähler
  Geben Sie ein, wie viele der überwachten Schnittstellen in den Zustand "down" wechseln müssen, bevor die Priorität geändert wird.

Die Tabelle gliedert sich in folgende Spalten:

- Track-ID
  Zeigt die Track-ID an.
- Schnittstelle
  Zeigt die Schnittstelle an, die überwacht wird.

Security-Funktionen projektieren

Benutzer

Lokale Benutzer

Hinweis

Die Seite ist bei einigen Geräten nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Lokale Benutzer

Auf dieser Seite erstellen Sie lokale Benutzeraccounts für WBM und CLI mit den entsprechenden Rechten.

Wenn Sie einen lokalen Benutzer anlegen oder löschen, wird diese Änderung automatisch auch in der Tabelle "Externe Benutzerkonten" durchgeführt. Wenn Sie eine Änderung explizit für die interne oder externe Benutzertabelle durchführen wollen, nutzen Sie die CLI-Befehle.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung

Die Seite enthält Folgendes:

- **Benutzerkonto**
  Geben Sie den Namen für den Benutzer ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 250 Zeichen lang sein.
  - Er darf folgende Zeichen nicht enthalten: § ? " ; :
  - Folgende Benutzernamen sind nicht erlaubt: admin, service, debug

  Hinweis

  Benutzername nicht änderbar

  Nach dem Anlegen eines Benutzers kann der Benutzername nicht mehr geändert werden.

  Wenn ein Benutzername geändert werden soll, dann muss der Benutzer gelöscht und ein neuer Benutzer angelegt werden.

  Hinweis

  Benutzernamen: admin, service, debug

  Werkseitig sind bei Auslieferung folgende Benutzernamen vordefiniert: admin, service, debug.
  - admin: Mit diesem Benutzernamen können Sie das Gerät konfigurieren. Wenn Sie sich das erste Mal anmelden oder nach einem "Auf Werkseinstellungen zurücksetzen und Neustart" anmelden, werden Sie aufgefordert das vordefinierte Passwort "admin" zu ändern.
  - service, debug: Diese Benutzernamen sind für Servicezwecke vorbehalten.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie verwendet wird:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

- **Passwort**
  Geben Sie das Passwort an. Die Stärke des Passworts ist abhängig von der eingestellten Passwortrichtlinie:
- **Passwort bestätigen**
  Geben Sie das Passwort erneut ein, um es zu bestätigen.
- **Rolle**
  Wählen Sie eine Rolle aus.
  Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".

Die Tabelle enthält folgende Spalten:

Hinweis

Die voreingestellten Benutzer sowie angemeldete Benutzer können nicht gelöscht oder geändert werden.

- **Benutzerkonto**
  Zeigt den Benutzernamen an.
- **Rolle**
  Zeigt die Rolle des Benutzers an.
- **Beschreibung**
  Zeigt eine Beschreibung des Benutzerkontos an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Rollen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Rollen

Auf dieser Seite erstellen Sie Rollen, die lokal auf dem Gerät gültig sind.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Rollenname**
  Geben Sie den Namen für die Rolle ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

  Hinweis
  Rollenname nicht änderbar

  Nach dem Anlegen einer Rolle kann der Rollenname nicht mehr geändert werden.

  Wenn ein Rollenname geändert werden soll, dann muss die Rolle gelöscht und eine neue Rolle angelegt werden.

Die Tabelle enthält folgende Spalten:

Hinweis
Rollen löschen

Die voreingestellten Rollen sowie zugewiesene Rollen können nicht gelöscht oder geändert werden

- **Rolle**
  Zeigt den Namen der Rolle an.
- **Funktionsrecht**
  Wählen Sie die Funktionsrechte der Rolle aus:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen aber nicht verändern.
    Benutzer mit dieser Rolle können ihr eigenes Passwort ändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.

  Hinweis
  Funktionsrecht nicht änderbar

  Wenn Sie eine Rolle zugewiesen haben, können Sie das Funktionsrecht der Rolle nicht mehr ändern.

  Wenn Sie das Funktionsrecht einer Rolle ändern wollen, gehen Sie wie folgt vor:
  1. Löschen Sie alle zugewiesenen Benutzer.
  2. Ändern Sie das Funktionsrecht der Rolle.
  3. Weisen Sie die Rolle erneut zu.

- **Beschreibung**
  Geben Sie eine Beschreibung für die Rolle ein. Bei vordefinierten Rollen wird eine Beschreibung angezeigt. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Gruppen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Benutzergruppen

Auf dieser Seite verknüpfen Sie eine Gruppe mit einer Rolle.

In diesem Beispiel wird die Gruppe "Administrators" mit der Rolle "admin" verknüpft. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert und der Gruppe "Administrators" zuordnet, erhält dieser Benutzer auf dem Gerät die Rechte der Rolle "admin".

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Der Name muss der Gruppe auf dem RADIUS-Server entsprechen. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

Die Tabelle enthält folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an.
- **Rolle**
  Wählen Sie eine Rolle aus. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät. Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".
- **Beschreibung**
  Geben Sie eine Beschreibung für die die Verknüpfung der Gruppe mit einer Rolle an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

AAA

Allgemein

Anmeldung von Netzteilnehmern

Die im Menü verwendete Bezeichnung "AAA" steht dabei für "Authentication, Authorization, Accounting". Dieses Feature dient dazu, Netzteilnehmer zu identifizieren und zuzulassen und ihnen die entsprechenden Dienste bereitzustellen.

Auf dieser Seite konfigurieren Sie die Anmeldung.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

Hinweis

Um den Login-Authentifizierung "RADIUS", "Lokal und RADIUS" oder "RADIUS und Fallback Lokal" nutzen zu können, muss ein RADIUS-Server hinterlegt und für die Benutzerauthentifizierung konfiguriert sein.

- **Login-Authentifizierung**
  Legen Sie fest, wie die Anmeldung erfolgt:
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen
  - Local und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen. Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS und Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen. Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.

RADIUS-Client

Authentifizierung über einen externen Server

Das Konzept von RADIUS basiert auf einem externen Authentifizierungsserver.

Jede Zeile der Tabelle enthält die Zugangsdaten für je einen Server. In der Suchreihenfolge wird der primäre Server zuerst angefragt. Ist der primäre Server nicht erreichbar, werden in der eingetragenen Reihenfolge sekundäre Server angefragt. Wenn keiner der Server antwortet, findet keine Authentifizierung statt.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **RADIUS-Autorisierungsmodus**
  Der RADIUS-Autorisierungsmodus legt bei der Login-Authentifizierung fest, wie bei einer erfolgreichen Authentifizierung die Rechtevergabe für die Benutzer erfolgt.
  - Standard
    In diesem Modus wird der Benutzer mit Administratorrechten angemeldet, wenn der Server für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück gibt. In allen anderen Fällen wird der Benutzer mit Leserechten angemeldet.
  - Herstellerspezifisch
    In diesem Modus ist die Rechtevergabe davon abhängig, ob und welche Gruppe der Server für den Benutzer zurück gibt und ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzeraccounts" gibt.

Die Tabelle gliedert sich in folgende Spalten:

- **Auth.-Servertyp**
  Wählen Sie aus, für welche Authentifizierungsverfahren der Server verwendet werden soll.
  - Login
    Der Server wird nur für die Login-Authentifizierung verwendet.
  - 802.1X
    Der Server wird nur für die 802.1X-Authentifizierung verwendet.
  - Login & 802.1X
    Der Server wird für beide Authentifizierungsverfahren verwendet.
- **Adresse des RADIUS-Servers**
  Tragen Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des RADIUS Servers ein.
- **Server-Port**
  Tragen Sie den Eingangs-Port auf dem RADIUS-Server ein. Standardmäßig ist der Eingangs-Port 1812 eingestellt. Der Wertebereich ist 1 ... 65535.
- **Shared Secret**
  Tragen Sie Ihre Zugangskennung ein. Der Wertebereich ist 1 ... 128 Zeichen.
- **Shared Secret bestätigen**
  Tragen Sie die Zugangskennung zur Bestätigung erneut ein.
- **Max. Retrans**
  Geben Sie hier die maximale Anzahl der Wiederholungen eines Anfrageversuchs ein. Der initiale Verbindungsversuch wird um den hier angegebenen Wert wiederholt, bevor ein anderer konfigurierter RADIUS-Server angefragt wird oder die Anmeldung für gescheitert erklärt wird. Standardmäßig sind 3 Wiederholungen eingestellt, das bedeutet 4 Verbindungsversuche. Der Wertebereich ist 1 ... 5.
- **Primärer Server**
  Legen Sie fest, ob dieser Server der primäre Server ist. Sie können aus den Optionen "ja" oder "nein" auswählen.

- **Test**
  Mit dieser Schaltfläche können Sie testen, ob der angegebene RADIUS-Server verfügbar ist oder nicht. Der Test wird einmalig durchgeführt und nicht zyklisch wiederholt.
- **Testergebnis**
  Zeigt an, ob der RADIUS-Server verfügbar ist oder nicht:
  - Fehlgeschlagen, keine Testpakete gesendet
    Die IP-Adresse ist nicht erreichbar.
    Die IP-Adresse ist erreichbar, der RADIUS-Server läuft jedoch nicht.
  - Erreichbar, das Shared Secret wurde nicht akzeptiert
    Die IP-Adresse ist erreichbar, der RADIUS-Server akzeptiert jedoch das angegebene Shared Secret nicht.
  - Erreichbar, das Shared Secret wurde akzeptiert
    Die IP-Adresse ist erreichbar und der RADIUS-Server akzeptiert das angegebene Shared Secret.

  Das Testergebnis wird nicht automatisch aktualisiert. Um das Testergebnis zu löschen, klicken Sie auf die Schaltfläche "Aktualisieren".

Zertifikate

Übersicht

Auf dieser Seite werden die geladenen Dateien (Zertifikate und Schlüssel) angezeigt. Folgende Möglichkeiten gibt es, um die Dateien ins Gerät zu laden:

- System > Laden & Speichern > HTTP
- System > Laden & Speichern > TFTP
- System > Laden & Speichern > SFTP

Beschreibung

- **Typ**
  Zeigt die Art der geladenen Datei an.
  - CA-Zert
    Das CA-Zertifikat ist von einer zertifizierenden Stelle (CA = Certification Authority) signiert.
  - Gerätezertifikat
  - Schlüsseldatei
  - Remote-Zert
    Gegenstellenzertifikat
- **Dateiname**
  Zeigt den Dateinamen an.
- **Status**
  Zeigt an, ob das Zertifikat gültig oder bereits abgelaufen ist.

- Antragsteller DN
  Zeigt den Namen des Antragstellers an.
- Aussteller DN
  Zeigt den Namen des Zertifikatsausstellers an.
- Ausstellungsdatum
  Zeigt den Beginn des Gültigkeitszeitraums des Zertifikats an
- Ablaufdatum
  Zeigt das Ende des Gültigkeitszeitraums des Zertifikats an.
- Verwendet
  Zeigt an, welche Funktion das Zertifikat nutzt.

Zertifikate

Das Format des Zertifikats basiert auf X.509, einem Standard der ITU-T zum Erstellen digitaler Zertifikate. In diesem Standard ist der schematische Aufbau von X.509-Zertifikaten beschrieben. Weitere Informationen dazu finden Sie im Internet unter "http://www.itu.int".

Auf dieser Seite kann der Inhalt folgender Strukturelemente angezeigt werden. Wenn in dem ausgewählten Zertifikat das Strukturelement nicht vorhanden oder befüllt ist, wird im Feld nichts angezeigt. Bestimmte Einträge sind nur editierbar, wenn Sie unterstützt werden.

Beschreibung

- Dateiname
  Wählen Sie das gewünschte Zertifikat aus.
- Typ
  Zeigt die Art der geladenen Datei an.
  - CA-Zert
    Das CA-Zertifikat ist von einer zertifizierenden Stelle (CA = Certification Authority) signiert.
  - Gerätezertifikat
  - Schlüsseldatei
  - Remote-Zert
    Gegenstellenzertifikat
- Antragsteller DN
  Zeigt den Namen des Antragsstellers an.
- Aussteller DN
  Zeigt den Namen des Zertifikatsausstellers an.
- Alternativer Name des Antragsstellers
  Wenn vorhanden, wird ein alternativer Name des Antragsstellers angezeigt.
- Ausstellungsdatum
  Zeigt den Beginn des Gültigkeitszeitraums des Zertifikats an
- Ablaufdatum
  Zeigt das Ende des Gültigkeitszeitraums des Zertifikats an.

- **Seriennummer**
  Zeigt die Seriennummer des Zertifikats an.
- **Verwendet**
  Zeigt an, welche Funktion das Zertifikat nutzt.
- **Verschlüsselungs-Algorithmus**
  Zeigt an, welches kryptografisches Verfahren verwendet wird.
- **Schlüsselverwendung**
  Zeigt an, für welchen Zweck der zum Zertifikat gehörende Schlüssel verwendet wird, z. B. zum Verifizieren digitaler Signaturen.
- **Erweiterte Schlüsselverwendung**
  Zeigt an, ob der Verwendungszweck noch zusätzlich beschränkt ist, z. B. nur zum Verifizieren von Signaturen des CA-Zertifikats.
- **Schlüsseldatei**
  Zeigt die Schlüsseldatei an.
- **Zertifikatssperrliste 1. URL**
  Tragen Sie die URL ein, über die die Sperrliste abgerufen werden kann. Nur editierbar, wenn vom Zertifikat unterstützt.
- **Zertifikatssperrliste 2. URL**
  Tragen Sie eine Alternativ-URL ein. Wenn die Sperrliste über die 1. URL nicht abrufbar ist, wird die Alternativ-URL verwendet. Nur editierbar, wenn vom Zertifikat unterstützt.
- **Zertifikat**
  Zeigt den Namen des Zertifikats an.
- **Passphrase**
  Tragen Sie das Passwort für das Zertifikat ein. Nur editierbar, wenn die verschlüsselte Datei Passwort-geschützt ist.
- **Passphrase bestätigen**
  Tragen Sie das Passwort nochmals ein. Nur editierbar, wenn die verschlüsselte Datei Passwort-geschützt ist.

Firewall

Allgemein

Auf dieser Seite aktivieren Sie die Firewall.

Hinweis

Bitte beachten Sie, wenn Sie die Firewall deaktivieren, dann ist ihr internes Netz ungeschützt.

Beschreibung

Die Seite enthält Folgendes:

- **Firewall aktivieren**
  Wenn aktiviert, ist die Firewall aktiv.
- **TCP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die TCP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 86400 Sekunden
- **UDP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die UDP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 300 Sekunden
- **ICMP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die ICMP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 300 Sekunden

Vordefinierte IPv4-Regeln

Die Seite enthält vordefinierte IP-Paketfilter-Regeln. Wenn Sie eigene IP-Paketfilter-Regeln anlegen, haben diese eine höhere Priorität als die vordefinierten IP-Paketfilter-Regeln.

Hier kann man einstellen, welche Dienste des Gerätes von welcher Schnittstelle/Subnetz aus erreichbar sein sollen.

Beschreibung

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht. Die Liste der Schnittstellen/Subnetze ist dynamisch und richtet sich nach den Einstellungen aus "Layer 3 >Subnetz".
  - VLANx: Erlaubt den Zugriff vom IP-Subnetz auf das Gerät.
- Der Zugriff auf folgende IPv4-Dienste wird erlaubt:
  - Alle
    Alle IPv4-Dienste
  - HTTP
    Zum Zugriff auf das Web Based Management.
  - HTTPS
    Zum gesicherten Zugriff auf das Web Based Management.

    Hinweis

    HTTP und HTTPS deaktiviert

    Wenn Sie HTTP und HTTPS deaktivieren, ist das WBM des Geräts nicht mehr erreichbar.

    HTTPS deaktiviert

    Wenn Sie HTTPS deaktivieren, können Sie nur noch über HTTP auf das WBM zugreifen. Vorausgesetzt, unter "System > Konfiguration > HTTP-Dienste" ist "HTTP & HTTPS" eingestellt. Ist z. B. "HTTP nach HTTPS umleiten" eingestellt, kann der Zugriff über HTTP nicht nach HTTPS umgeleitet werden. Somit ist das WBM des Geräts nicht mehr erreichbar.

  - DNS
    DNS-Anfragen an das Gerät. Nur notwendig, wenn am Gerät die Funktion "DNS-Relay" aktiv ist.
  - SNMP
    Eingehende SNMP-Verbindungen. Notwendig, um z. B. mit einem MIB-Browser auf die SNMP-Informationen des Geräts zuzugreifen.
  - Telnet
    Zum unverschlüsselten Zugriff auf das CLI.
  - IPsec VPN
    Erlaubt den IKE (Internet Key Exchange) Datenverkehr vom externen Netz zum Gerät. Notwendig, wenn eine IPsec VPN-Gegenstelle eine Verbindung zu diesem Gerät herstellen soll.
  - SSH
    Zum verschlüsselten Zugriff auf das CLI.
  - DHCP
    Zugriff auf den DHCP-Server oder den DHCP-Client
  - Ping
    Zugriff auf die Ping-Funktion.
  - Systemzeit
    Zugriff auf NTP und SNTP

IP-Dienste

Auf dieser Seite definieren Sie IP-Dienste. Mithilfe der IP-Dienst-Definitionen können Sie Firewall-Regeln definieren, die auf bestimmte Dienste angewendet werden. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Namen.

Beschreibung

Die Seite enthält Folgendes:

- **Name des Diensts**
  Tragen Sie den Namen für den IP-Dienst ein. Der Name muss eindeutig sein.

Die Tabelle enthält folgende Spalten:

- **Name des Diensts**
  Zeigt den Namen des IP-Diensts an.
- **Transport**
  Legen Sie den Protokolltyp fest.
  - UDP
    Die Regel gilt nur für UDP-Telegramme.
  - TCP
    Die Regel gilt nur für TCP-Telegramme.
- **Quell-Port (Bereich)**
  Tragen Sie den Quell-Port ein. Die Regel gilt genau für den angegebenen Port.
  - Wenn die Regel für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Regel für alle Ports gelten soll, geben Sie "*" ein.
- **Ziel-Port (Bereich)**
  Tragen Sie den Ziel-Port ein. Die Regel gilt genau für den angegebenen Port.
  - Wenn die Regel für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Regel für alle Ports gelten soll, geben Sie "*" ein.

ICMP-Dienste

Auf dieser Seite definieren Sie ICMP-Dienste. Mithilfe der ICMP-Dienst-Definitionen können Sie Firewall-Regeln definieren, die auf bestimmte Dienste angewendet werden. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Namen.

Beschreibung

Die Seite enthält Folgendes:

- **Name des Diensts**
  Tragen Sie einen Namen für den ICMP-Dienst ein. Der Name muss eindeutig sein.

Die Tabelle enthält folgende Spalten:

- **Name des Diensts**
  Zeigt den Namen des ICMP-Diensts an.
- **Protokoll**
  Zeigt die Version des ICMP-Protokolls an.
- **Typ**
  Legen Sie den ICMP-Pakettyp fest. Einige Beispiele sind:
  - Destination Unreachable
    IP-Telegramm kann nicht zugestellt werden.
  - Time Exeeded
    Zeitlimit überschritten
  - Echo-Request
    Echo-Frage, besser bekannt als Ping.
- **Code**
  Der Code beschreibt den ICMP-Pakettyp genauer. Die Auswahl ist abhängig vom gewählten ICMP-Pakettyp.
  Bei "Destination Unreachable" ist z. B. "Code 1" Host ist nicht erreichbar.

IP-Protokolle

Auf dieser Seite können Sie benutzerdefinierte Protokolle konfigurieren, z. B. IGMP für Multicast-Gruppen. Sie vergeben hierbei einen Protokollnamen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Protokollnamen.

Beschreibung

Die Seite enthält Folgendes:

- **Protokollname**
  Tragen Sie einen Namen für das Protokoll ein.

Die Seite enthält folgende Optionskästchen:

- **Protokollname**
  Zeigt den Protokollnamen an.
- **Protokollnummer**
  Tragen Sie die Protokollnummer ein, z. B. 2. Eine Liste der Protokollnummern finden Sie auf den Internetseiten von iana.org

IP-Regeln

Auf dieser Seite legen Sie eigene IP-Paketfilter-Regeln für die Firewall fest.

Die hier erstellten IP-Paketfilter-Regeln haben Vorrang:

- vor den vordefinierten IP-Paketfilter-Regeln (Predefined IPv4) und
- vor den IP-Paketfilter-Regeln, die aufgrund einer Verbindungsprojektierung (SINEMA RC) automatisch angelegt werden.

Beschreibung der angezeigten Felder

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Protokoll**
  Zeigt die Version des IP-Protokolls an.
- **Aktion**
  Wählen Sie aus, wie mit eintreffenden IP-Paketen zu verfahren ist:
  - "Accept" – Die Datenpakete dürfen passieren,
  - "Reject" – Die Datenpakete werden abgewiesen, der Absender erhält eine entsprechende Meldung,
  - "Drop" – Die Datenpakete werden ohne Rückmeldung an den Absender verworfen.
- **Von / Nach**
  Legen Sie die Kommunikationsrichtung der IP-Regel fest.
  - VLANx: VLANs mit konfiguriertem Subnetz
  - Gerät
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsecVPN-Verbindung
  - ppp2: WAN-Schnittstelle
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- **Quelle (Bereich)**
  Tragen Sie die IP-Adresse oder einen IP-Bereich ein, die IP-Pakete senden darf.
  - Wenn die Regel für einen IP-Bereich gelten soll, geben Sie den Bereich mit Startadresse "-" Endadresse an, z. B. 192.168.100.10 - 192.168.100.20.
  - Wenn die Regel für alle IP-Adressen gelten soll, geben Sie " 0.0.0.0/0" ein.
- **Ziel (Bereich)**
  Tragen Sie die IP-Adresse oder einen IP-Bereich ein, die IP-Pakete empfangen darf.
  - Wenn die Regel für einen IP-Bereich gelten soll, geben Sie den Bereich mit Startadresse "-" Endadresse an, z. B. 192.168.100.10 - 192.168.100.20.
  - Wenn die Regel für alle IP-Adressen gelten soll, geben Sie " 0.0.0.0/0" ein.
- **Dienst**
  Wählen Sie den Dienst oder den Protokollnamen aus, für den diese Regel gültig ist.

- **Log**
  Legen Sie fest, ob das Zutreffen der Regel protokolliert wird und welche Ereignisschwere der Eintrag hat.
  Folgende Einstellungen gibt es:
  - none
    Das Zutreffen wird nicht protokolliert.
  - info / warning / critical
    Das Zutreffen wird mit der gewählten Ereignisschwere protokolliert. Die Logdatei wird unter "Information" > "Log-Tabellen" > "Firewall-Log" angezeigt.
- **Reihenfolge**
  Legen Sie die Reihenfolge der Regel fest.

Hinweis

WBM über VPN-Tunnel öffnen

Um das WBM über einen VPN-Tunnel zu öffnen, legen Sie die folgende IP-Regel an:

Aktion: Accept

Von: IPsec

Nach: Gerät

Dienst: HTTPS

IPsec VPN

Übersicht

Unter "Security > IPsec VPN" können Sie die Parameter zum Aufbau von IPsec-VPN-Verbindungen manuell projektieren.

Alternative Projektierung über VPN-Gruppen

Alternativ können Sie die IPsec-Verbindungen über VPN-Gruppen projektieren, siehe Kapitel AUTOHOTSPOT. Bei der Projektierung über VPN-Gruppen werden die Remote-Adressen und Remote-Subnetze von STEP 7 anhand der Schnittstellennamen ermittelt:

- Schnittstellenname "INT": STEP 7 behandelt die Schnittstelle als interne Schnittstelle.
- Schnittstellenname "EXT": STEP 7 behandelt die Schnittstelle als externe Schnittstelle.

STEP 7 legt beim Hinzufügen von Geräten in VPN-Gruppen ausschließlich VPN-Verbindungen zwischen den externen Schnittstellen an. Nur die lokalen Subnetze an den internen Schnittstellen werden dadurch über VPN miteinander verbunden. Zusätzliche VPN-Verbindungen müssen Sie manuell projektieren.

Besonderheiten bei der Projektierung über VPN-Gruppen

Gegenüber SCALANCE S-600 V3 Geräten gelten folgende Besonderheiten:

- Die Erlaubnis zur Initiierung des Verbindungsaufbaus legen Sie bei der Zuweisung zu VPN-Gruppen fest.
- In den verbindungsspezifischen VPN-Einstellungen wird das Gerät als Verbindungspartner nicht editierbar angezeigt. Bei der Auswahl des Geräts kann die Schnittselle anderer Partnergeräte angepasst werden.
- Das Gerät verhält sich bezüglich der VPN-Betriebsart stets als Routing-Gerät.
- Die Eigenschaften einer VPN-Gruppe werden in die lokalen Eigenschaften von VPN-Gruppenteilnehmern übernommen. Die Eigenschaften einer VPN-Gruppe können in Bezug auf folgende Parameter lokal überschrieben werden:
  - Phase 1: Keying-Versuche, DPD, DPD-Zeitraum, DPD-Timeout
  - Phase 2: Lifebytes, Protokoll, Port (Bereich), Auto-Firewallregeln

Allgemein

Auf der Seite konfigurieren Sie die Grundeinstellungen für VPN.

Beschreibung

Die Seite enthält Folgendes:

- **IPsec VPN aktivieren**
  Aktivieren oder deaktivieren Sie das IPsec-Verfahren für VPN.
- **CRL-Richtlinie strikt durchsetzen**
  Wenn aktiviert, wird die Gültigkeit der Zertifikate anhand der Zertifikatssperrliste (CRL-Certificate Revocation List) überprüft. In der Zertifikatssperrliste sind die von der Zertifizierungsstelle ausgestellten Zertifikate aufgeführt, die vor ihrem gesetzten Ablaufdatum ihre Gültigkeit verloren haben. Welche Zertifikatssperrliste verwendet wird, konfigurieren Sie auf der Seite "Zertifikate (Seite 732)".
- **NAT Keep Alive-Zeitintervall**
  Legen Sie fest, in welchen Zeitabständen Lebenszeichentelegramme (Keep Alive) gesendet werden. Befindet sich ein NAT-Gerät zwischen zwei VPN-Endpunkten, dann wird bei Inaktivität die Verbindung aus dessen dynamischer NAT-Tabelle gelöscht. Um dies zu verhindern, werden die Lebenszeichentelegramme gesendet.

Remote-Endpunkt

Auf dieser Seite konfigurieren Sie die Gegenstelle (VPN-Endpunkt).

Beschreibung

Die Seite enthält Folgendes:

- **Name Remote-Endpunkt**
  Tragen Sie einen Namen für die Gegenstelle ein und klicken Sie auf "Neuer Eintrag", um eine neue Gegenstelle zu erstellen.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der Gegenstelle an.
- **Remote-Modus**
  Legen Sie fest, welche Rolle die Gegenstellen einnimmt.
  - Roadwarrior
    Die erreichbaren Gegenstellen (Remote-Adresse) werden eingetragen. Die erreichbaren Remote-Subnetze werden von der Gegenstelle gelernt.
  - Standard
    Die erreichbare Gegenstelle (Remote-Adresse) und die erreichbaren Remote-Subnetze werden fest eingetragen.
- **Remote-Typ**
  Legen Sie die Art der Gegenstellen-Adresse fest.
  - Manuell
    Die Adresse der Gegenstelle ist bekannt. Das Gerät kann entweder als VPN-Client die VPN-Verbindung aktiv aufbauen, oder passiv auf den Verbindungsaufbau durch die Gegenstelle warten.
  - Beliebig
    Nimmt die Verbindung von Gegenstellen mit beliebiger IP-Adresse an. Das Gerät kann nur auf VPN-Verbindungen warten, aber nicht als aktiver Partner einen VPN-Tunnel aufbauen.
- **Remote-Adresse**
  Nur beim Remote-Typ "Manuell" editierbar.
  - Im Standard-Modus tragen Sie die WAN-IP-Adresse oder den DDNS-Hostnamen der Gegenstelle ein. Die Netzmaske ist immer /32
  - Im Roadwarrior-Modus können Sie entweder die Adresse der Gegenstelle vorgeben oder einen IP-Bereich eintragen, aus dem Verbindungen entgegengenommen werden.
- **Remote-Subnetz**
  - Im Standard-Modus geben Sie das erreichbare Subnetz der Gegenstelle ein. Verwenden Sie die CIDR-Schreibweise.
  - Im Roadwarrior-Modus teilt die Gegenstelle dem Gerät seinen erreichbaren Subnetze mit und das Gerät lernt diese.
- **Virtueller IP-Modus**
  Legen Sie fest, ob der Gegenstelle eine virtuelle IP-Adresse angeboten wird.
  Folgende Möglichkeiten gibt es:
  - Benutzerdefinierte IPv4
    Die virtuelle IP-Adresse ist aus dem bei "Virtuelle IP" festgelegten Band.
  - Keine
    Keine virtuelle IP-Adresse. Der VPN-Tunnel wird dynamisch zur internen IP-Adresse der Gegenstelle aufgebaut.
- **Virtuelle IP**
  Legen Sie das Subnetz fest (CIDR), aus dem die Gegenstelle eine virtuelle IP-Adresse angeboten bekommt.
  Nur editierbar, wenn bei "Virtueller IP-Modus" "Benutzerdefinierte IPv4" ausgewählt ist.

Verbindungen

Auf dieser Seite konfigurieren Sie die Grundeinstellungen für die VPN-Verbindung. Mit diesen Einstellungen kann das Gerät (lokaler Endpunkt) einen gesicherten VPN-Tunnel zur Gegenseite aufbauen. Die Sicherheitseinstellungen legen Sie auf der WBM-Seite "Authentifizierung" fest.

Hinweis

Wenn Sie "NETMAP" verwenden,

- werden nur Auto-Firewall-Regeln unterstützt.
- ist bei "Betrieb" die Einstellung "Auf Anforderung" nicht auswählbar.

Hinweis

Mehrere IPsec VPN-Verbindungen über denselben VPN-Endpunkt

Wenn Sie über denselben VPN-Endpunkt IPsec VPN-Verbindungen zu verschiedenen Remote-Subnetzen angelegt haben, ist die zuerst projektierte VPN-Verbindung (niedrigster Index) die Hauptverbindung (Parent).

Über die Hauptverbindung werden alle weiteren IPsec VPN-Verbindungen (Child) angelegt und aufgebaut. Wenn nun alle VPN-Tunnel aufgebaut sind und die Hauptverbindung beendet wird, so werden allen Child-Verbindungen unterbrochen. Nach Ablauf des DPD-Timeouts werden alle IPsec VPN-Verbindungen wieder neu über Hauptverbindung aufgebaut.

Wenn nur eine Child-Verbindung beendet wird, bleiben die Hauptverbindung und die weiteren Child-Verbindungen bestehen.

Beschreibung

Die Seite enthält folgende Felder:

- **Verbindungsname**
  Geben Sie einen Namen für die VPN-Verbindung ein und klicken Sie auf "Erstellen", um eine neue Verbindung zu erstellen.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Name**
  Zeigt den Namen der VPN-Verbindung an.

- **Betrieb**
  Legen Sie fest, von wem die VPN-Verbindung aufgebaut wird.
  - Deaktiviert
    Die VPN-Verbindung ist deaktiviert.
  - Starten
    Das Gerät versucht, zu der Gegenstelle eine VPN-Verbindung aufzubauen.
  - Warten
    Das Gerät wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
  - Auf Anforderung
    Die VPN-Verbindung wird bei Bedarf aufgebaut.
  - Bei DI starten
    Beim Eintreten des Ereignisses "Digitaler Eingang" versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Auf DI warten
    Beim Eintreten des Ereignisses "Digitaler Eingang" wartet das Gerät, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
- **Keying-Protokoll**
  Legen Sie fest, ob IKEv2 oder IKEv1 verwendet wird.
- **Remote-Endpunkt**
  Wählen Sie die gewünschte Gegenstelle aus. Nur die Gegenstellen sind projektierbar, die Sie auf der WBM-Seite "Remote-Endpunkt" konfiguriert haben.
- **Lokales Subnetz**
  Tragen Sie das lokale Subnetz ein. Verwenden Sie die CIDR-Schreibweise. Das lokale Netz kann auch nur ein einzelner PC oder eine andere Untermenge des lokalen Netzes sein.
- **Virtuelle IP anfordern**
  Wenn aktiviert, wird beim Verbindungsaufbau eine virtuelle IP-Adresse von der Gegenstelle angefordert.
- **Timeout [Sek]**
  Nur bei der Einstellung "Auf Anforderung" notwendig. Tragen Sie die Zeitspanne ein, nach der die VPN-Verbindung getrennt wird. Wenn innerhalb dieser Zeit keine Pakete gesendet werden, wird die VPN-Verbindung automatisch getrennt.

Authentifizierung

Auf dieser Seite legen Sie fest, wie sich die VPN-Verbindungspartner gegenseitig authentisieren.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Authentifizierung**
  Wählen Sie das Authentifizierungsverfahren aus. Voraussetzung für die VPN-Verbindung ist, dass die Gegenstelle das gleiche Authentisierungsverfahren verwendet.
  - Deaktiviert
    Es ist kein Authentifizierungsverfahren gewählt. Ein Verbindungsaufbau ist nicht möglich.
  - Remote-Zert
    Für die Authentifizierung wird das Gegenstellenzertifikat verwendet. Das Zertifikat legen Sie bei "Remote-Zertifikat" fest
  - CA-Zert
    Für die Authentifizierung wird das Zertifikat der Zertifizierungsstelle verwendet. Das Zertifikat legen Sie bei "CA-Zertifikat" fest. Wenn Sie eine IPsec-Verbindung zu SINEMA RC aufbauen wollen, müssen Sie das Zertifikat nachträglich über das WBM hinzufügen und auf das Gerät laden.
  - PSK
    Für die Authentifizierung wird ein Schlüssel verwendet. Den Schlüssel konfigurieren Sie bei "PSK".
- **CA-Zertifikat**
  Wählen Sie das Zertifikat aus. Nur geladene Zertifikate sind auswählbar.
- **Lokales Zertifikat**
  Wählen Sie das Gerätezertifikat aus. Nur die Gerätezertifikat sind auswählbar, die vom TIA Portal erzeugt wurden.
- **Lokale-ID**
  Geben Sie die Lokale-ID aus dem Gegenstellenzertifikat ein. Nur wenn Sie das Gegenstellenzertifikat verwenden, können Sie das Feld leer lassen. Das Feld wird automatisch mit dem Wert aus dem Gegenstellenzertifikat befüllt.
- **Remote-Zertifikat**
  Wählen Sie das Gegenstellenzertifikat aus. Nur die Gegenstellenzertifikate sind auswählbar, die vom TIA Portal erzeugt wurden.
- **Remote-ID**
  Geben Sie den "Distinguished Name" oder "Alternate Name" aus dem Gegenstellenzertifikat ein. Nur wenn Sie das Gegenstellenzertifikat verwenden, können Sie das Feld leer lassen. Das Feld wird automatisch mit dem Wert aus dem Gegenstellenzertifikat befüllt.
- **PSK**
  Geben Sie den Schlüssel ein.
- **PSK bestätigen**
  Wiederholen Sie den Schlüssel.

Phase 1

Phase 1: Verschlüsselungsvereinbarung und Authentisierung (IKE = Internet Key Exchange)

Auf dieser Seite stellen Sie die Parameter für das Protokoll des IPsec-Schlüsselmanagements ein. Der Schlüsselaustausch erfolgt über das standardisierte Verfahren IKE, für das Sie folgende Protokollparameter einstellen können.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Default-Chiffre**
  Wenn aktiviert, wird beim Verbindungsaufbau eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Die Auswahl ist abhängig vom Schlüsselaustauschverfahren. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 606)".
- **Verschlüsselung**
  Wählen Sie für die Phase 1 den gewünschten Verschlüsselungsalgorithmus aus. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Die Auswahl ist abhängig vom Schlüsselaustauschverfahren. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 606)".

  Hinweis

  Die AES-Modi CCM und GCM beinhalten separate Mechanismen für die Authentisierung von Daten. Wenn Sie bei "Verschlüsselung" einen Modus AES x CCM verwenden, dann wird dieser auch zur Authentisierung verwendet. Von dem Parameter "Authentifizierung" wird dann nur noch die Pseudo-Random-Funktion abgeleitet. Damit eine VPN-Verbindung aufgebaut wird, müssen alle Geräte die gleichen Einstellungen verwenden.

- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende Verfahren werden unterstützt:
  - MD5
  - SHA1
  - SHA512
  - SHA256
  - SHA384

- **Schlüsselableitung**
  Wählen Sie die gewünschte Diffie-Hellmann-Gruppe (DH), aus der ein Schlüssel erzeugt wird. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende DH-Gruppen werden unterstützt:
  - DH Group 1
  - DH Group 2
  - DH Group 5
  - DH Group 14
  - DH Group 15
  - DH Group 16
  - DH Group 17
  - DH Group 18
- **Keying-Versuche**
  Tragen Sie die Anzahl der Wiederholungen für einen fehlgeschlagenen Verbindungsaufbau ein. Wenn Sie den Wert 0 eintragen, wird der Verbindungsaufbau unendlich oft zu versucht.
- **Lebensdauer [min]**
  Tragen Sie einen Zeitraum in Minuten ein, der die Lebensdauer der Authentisierung festlegt. Nach Ablauf der Zeit müssen sich die beteiligten VPN-Endpunkte erneut gegenseitig authentisieren und einen neuen Schlüssel erzeugen.
- **DPD**
  Wenn aktiviert, wird DPD verwendet. Mit DPD lässt sich feststellen, ob die VPN-Verbindung noch besteht oder ob sie abgebrochen ist.

  Hinweis

  Durch das Versenden der DPD-Anfragen steigt die Anzahl der gesendeten und empfangenen Daten. Dies kann zu erhöhten Kosten führen.

- **DPD-Zeitraum [Sek]**
  Tragen Sie eine Zeitspanne ein, nach der DPD-Anfragen gesendet werden. Diese Anfragen testen, ob die Gegenstelle noch verfügbar ist
- **DPD-Timeout [Sek]**
  Tragen Sie eine Zeitspanne ein. Wenn auf die DPD-Anfragen keine Antwort erfolgt, dann wird nach Ablauf dieser Zeit die Verbindung zur Gegenstelle für ungültig erklärt.
- **Aggressive Mode**
  - Deaktiviert:
    Main Mode wird verwendet.
  - Aktiviert
    Aggressive Mode wird verwendet

  Der Unterschied zwischen Main- und Aggressive Mode ist die "Identity-Protection", die im Main Mode verwendet wird. Die Identität wird im Main Mode verschlüsselt übertragen, im Aggressive Mode nicht.

Phase 2

Phase 2: Datenaustausch (ESP = Encapsulating Security Payload)

Auf dieser Seite stellen Sie die Parameter für das Protokoll des IPsec-Datenaustauschs ein. Die gesamte Kommunikation in dieser Phase erfolgt verschlüsselt über das standardisierte Sicherheitsprotokoll ESP, für das Sie folgende Protokollparameter einstellen können.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Default-Chiffre**
  Wenn aktiviert, wird beim Verbindungsaufbau eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 606)".
- **Verschlüsselung**
  Wählen Sie für die Phase 2 den gewünschten Verschlüsselungsalgorithmus aus. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 606)".

  Hinweis

  Die AES-Modi CCM und GCM beinhalten separate Mechanismen für die Authentisierung von Daten. Wenn Sie bei "Verschlüsselung" einen Modus AES x CCM oder AES x GCM verwenden, dann wird dieser auch zur Authentisierung verwendet. Von dem Parameter "Authentifizierung" wird dann nur noch die Pseudo-Random-Funktion abgeleitet.

- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende Verfahren werden unterstützt:
  - MD5
  - SHA1
  - SHA512
  - SHA256
  - SHA384

- **Schlüsselableitung**
  Wählen Sie die gewünschte Diffie-Hellmann-Gruppe (DH), aus der ein Schlüssel erzeugt wird. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende DH-Gruppen werden unterstützt:
  - None: Für die Phase 2 werden keine separaten Schlüssel ausgetauscht. Damit ist Perfect Forward Secrecy (PFS) deaktivert.
  - DH Group 1
  - DH Group 2
  - DH Group 5
  - DH Group 14
  - DH Group 15
  - DH Group 16
  - DH Group 17
  - DH Group 18

  Hinweis

  Damit eine VPN-Verbindung aufgebaut wird, müssen alle Geräte die gleichen Einstellungen verwenden oder kompatible Verschlüsselungsverfahren anbieten.

- **Lebensdauer [min]**
  Tragen Sie einen Zeitraum in Minuten ein, der die Lebensdauer der vereinbarten Schlüssel festlegt. Nach Ablauf der Zeit wird der Schlüssel neu ausgehandelt.
- **Lifebytes**
  Tragen Sie das Datenlimit in Bytes ein, das die Lebensdauer der vereinbarten Schlüssel festlegt. Nach Ablauf des Datenlimits wird der Schlüssel neu ausgehandelt.
- **Protokoll**
  Legen Sie fest, für welches Protokoll die VPN-Verbindung gültig ist, z. B. UDP, TCP, ICMP. Wenn die Einstellung für alle Protokolle gelten soll, geben Sie "*" ein.
- **Port (Bereich)**
  Legen Sie den Port fest, durch den der VPN-Tunnel kommunizieren kann. Die Einstellung gilt genau für den angegebenen Port
  - Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Einstellung für alle Ports gelten soll, geben Sie "*" ein.

  Die Einstellung hat nur bei portbasierten Protokollen Auswirkungen.
- **Auto-Firewallregeln**
  - akiviert
    Für die VPN-Verbindung werden automatisch die Firewall-Regeln angelegt.
  - deaktiviert
    Sie müssen selbst die Firewall-Regeln anlegen.

OpenVPN-Client

Allgemein

Auf dieser Seite aktivieren Sie den OpenVPN-Client.

Beschreibung

Die Seite enthält Folgendes:

- **OpenVPN-Client aktivieren**
  Aktivieren oder deaktivieren Sie den OpenVPN-Client.

Verbindungen

Auf dieser Seite konfigurieren Sie die Grundeinstellungen für die OpenVPN-Verbindung. Die Sicherheitseinstellungen legen Sie auf der WBM-Seite "Authentifizierung" fest.

Beschreibung

- **Verbindungsname**
  Geben Sie einen eindeutigen Namen für die OpenVPN-Verbindung ein und klicken Sie auf "Erstellen", um eine neue Verbindung zu erstellen.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Name**
  Zeigt den Namen der OpenVPN-Verbindung an.
- **Betrieb**
  Legen Sie fest, wie die Verbindung aufgebaut wird. Weitere Informationen dazu finden Sie unter "VPN-Verbindungsaufbau".
  - Starten
    Das Gerät versucht, zu der Gegenstelle eine VPN-Verbindung aufzubauen.
  - Bei DI starten
    Beim Eintreten des Ereignisses "Digitaler Eingang" versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Deaktiviert
    Die VPN-Verbindung ist deaktiviert.

- **Verschlüsselung**
  Wählen Sie den gewünschten Verschlüsselungsalgorithmus aus.
  - AES-128-CBC (Default)
  - AES-192-CBC
  - AES-256-CBC
  - DES-EDE3
  - BF-CBC
- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest.
  - SHA256 (Default)
  - SHA384
  - SHA512
  - SHA224
  - SHA1
  - MD5
- **LZO Komp.**
  - Deaktiviert (-)
    Die Komprimierung ist deaktiviert. Der Server kann Komprimierung nicht wieder einschalten.
  - Nein
    Die Komprimierung ist standardmäßig deaktiviert. Der Server kann Komprimierung einschalten
  - Ja
    Die Komprimierung ist standardmäßig aktiviert. Der Server kann die Komprimierung ausschalten
  - Adaptiv
    Komprimierung wird standardmäßig adaptiv aktiviert. Nur wenn gut komprimierbare Daten vorliegen wird komprimiert, ansonsten wird die Kompression für eine gewisse Zeit deaktiviert.
- **Auto-Firewallregeln**
  - Aktiviert
    Für die VPN-Verbindung werden automatisch die Firewall-Regeln angelegt.
  - Deaktiviert
    Sie müssen selbst die geeigneten Firewall-Regeln anlegen.
- **NAT aktivieren**
  Mit dieser Einstellung aktivieren Sie automatisches IP-Masquerading für diese Schnittstelle. Die lokalen Geräte sind von außen nicht direkt erreichbar, sondern nur über die IP-Adresse der Schnittstelle. Die lokalen Geräte können sich aber mit den Geräten hinter dem OpenVPN-Server verbinden. Weitere Informationen zu NAT finden Sie unter "NAT".

Remote

Auf dieser Seite konfigurieren Sie die Gegenstelle (OpenVPN-Endpunkt). Pro Verbindung können Sie mehrere OpenVPN.Gegenstellen festlegen. Das Gerät probiert alle projektierten OpenVPN-Gegenstellen der Reihe nach aus bis ein Verbindungsaufbau erfolgreich ist.

Beschreibung

Die Seite enthält Folgendes:

- **Remote-Name**
  Geben Sie einen Namen für die OpenVPN-Gegenstelle ein.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der OpenVPN-Gegenstelle an.
- **Verbindung**
  Wählen Sie die zugehörige Verbindung aus. Nur die Verbindungen sind projektierbar, die Sie auf der Seite "Verbindungen" konfiguriert haben.
- **Remote-Adresse**
  Geben Sie die WAN-IP-Adresse oder den DNS-Hostnamen der OpenVPN-Gegenstelle ein.
- **Port**
  Legen Sie den Port fest, durch den der OpenVPN-Tunnel kommunizieren kann. Die Einstellung gilt genau für den angegebenen Port.
- **Protokoll**
  Legen Sie fest, für welches Protokoll die OpenVPN-Verbindung genutzt werden soll.

Authentifizierung

Auf dieser Seite legen Sie fest, wie sich die VPN-Verbindungspartner gegenseitig authentisieren.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Methode**
  Wählen Sie das Authentifizierungsverfahren aus. Voraussetzung für die VPN-Verbindung ist, dass die Gegenstelle das gleiche Authentisierungsverfahren verwendet.
  - Deaktiviert
    Es ist kein Authentifizierungsverfahren gewählt. Ein Verbindungsaufbau ist nicht möglich.
  - Zertifikate
    Für die Authentifizierung werden Zertifikate verwendet. Die Zertifikate müssen nachträglich über das WBM hinzugefügt und auf das Gerät geladen werden.
  - Benutzename/Passwort
    Für die Authentifizierung werden Benutzername/Passwort verwendet. Das Zertifikat muss nachträglich über das WBM hinzugefügt und auf das Gerät geladen werden.
- **CA-Zertifikat**
  Wählen Sie das CA-Zertifikat über das WBM aus und laden Sie es auf das Gerät.
- **Gerätezertifikat**
  Wählen Sie das Gerätezertifikat über das WBM aus und laden Sie es auf das Gerät.
- **Benutzername**
  Geben Sie den Benutzernamen an.
- **Passwort**
  Geben Sie das Passwort an.
- **Passwort bestätigen**
  Bestätigen Sie das Passwort.

1.4 Konfigurationen erstellen

1.4.1 Automatisierungssysteme konfigurieren

1.4.1.1 Arbeitsweise von S7-300/400 CPUs

Betriebszustände

Grundlagen zu den Betriebszuständen von S7-CPUs

Betriebszustände beschreiben das Verhalten der CPU. Sie können die Betriebszustände bei der Programmierung des Anlaufs, beim Test der Steuerung und bei der Fehlerdiagnose zu jedem beliebigen Zeitpunkt abfragen und auswerten.

Für die S7-CPUs der verschiedenen CPU-Familien wie z. B. S7-300, S7-400, ET 200S, ET 200Pro oder auch WinLC (PC-basierte Controller) oder Embedded Controller gibt es folgende Betriebszustände:

- STOP
- ANLAUF
- RUN
- HALT

In diesen Betriebszuständen ist die CPU über ihre Schnittstellen (z. B. über eine MPI-Schnittstelle) kommunikationsfähig.

Betriebszustand "STOP"

Im Betriebszustand "STOP" prüft die CPU, ob alle konfigurierten bzw. über Standard-Adressierung eingesetzten Baugruppen vorhanden sind. Davon abhängig setzt die CPU die Peripherie in einen vordefinierten Grundzustand. Das Anwenderprogramm wird im Betriebszustand "STOP" nicht bearbeitet.

Siehe auch: Betriebszustand "STOP" (Seite 755)

Betriebszustand "ANLAUF"

Im Betriebszustand "ANLAUF" werden zwischen den Anlaufarten "Neustart (Warmstart)", "Wiederanlauf" und "Kaltstart" unterschieden.

Siehe auch: Betriebszustand "ANLAUF" (Seite 756)

Betriebszustand "RUN"

Im Betriebszustand "RUN" bearbeitet die CPU das Anwenderprogramm, aktualisiert die Ein- und Ausgänge und bearbeitet Alarme und Fehlermeldungen.

Siehe auch: Betriebszustand "RUN" (Seite 765)

Betriebszustand "HALT"

Im Betriebszustand "HALT" wird die Bearbeitung des Anwenderprogramms angehalten. Der Betriebszustand "HALT" wird nur beim Testen mit dem Programmiergerät erreicht.

Siehe auch: Betriebszustand "HALT" (Seite 765)

Weitere Betriebszustände

Wenn die CPU nicht betriebsbereit ist, befindet sie sich in einem der beiden Betriebszustände:

- Spannungslos, d. h. die Netzspannung ist ausgeschaltet.
- Defekt, d. h. es ist ein nicht behebbarer Fehler aufgetreten.
  Überprüfen Sie, ob die CPU tatsächlich defekt ist. Setzen Sie dazu die CPU auf "STOP", und schalten Sie den Netzschalter aus und wieder ein. Um den Fehler zu analysieren, lesen Sie den Diagnosepuffer aus, wenn die CPU anläuft. Wenn die CPU nicht anläuft, tauschen Sie sie aus.

Betriebszustandsübergänge

Übersicht

Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge von S7-300-CPUs:

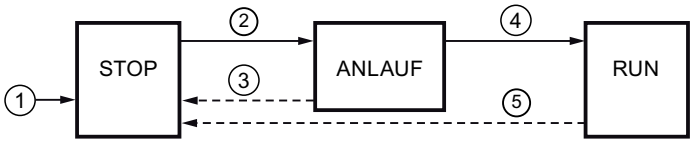

Die folgende Tabelle zeigt die Bedingungen, unter welchen die Betriebszustände wechseln:

Nr.	Betriebszustandsübergang	Bedingungen
①	STOP	Nach dem Einschalten der Versorgungsspannung befindet sich die CPU im Betriebszustand "STOP", bestimmt die angeforderte Anlaufart und nimmt dann den nächsten Betriebszustand ein.
②	STOP → ANLAUF	Die CPU geht in folgenden Fällen in den Betriebszustand "ANLAUF": • Die CPU wird über den Betriebsartenschalter oder vom Programmiergerät aus auf "RUN" gesetzt. • Nach automatischer Auslösung einer Anlaufart durch "NETZ-EIN". • Die Kommunikationsfunktion "RESUME" oder "START" wird ausgeführt. Der Betriebsartenschalter muss in den letzten beiden Fällen auf "RUN" stehen.

Nr.	Betriebszustandsübergang	Bedingungen
③	ANLAUF → STOP	Die CPU geht in folgenden Fällen wieder in den STOP-Zustand: • Während des Anlaufs wird ein Fehler erkannt. • Die CPU wird über den Betriebsartenschalter oder vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anlauf-OB bearbeitet. • Die Kommunikationsfunktion "STOP" wird ausgeführt.
④	ANLAUF → RUN	Wenn der Anlauf erfolgreich ist, wechselt die CPU in "RUN".
⑤	RUN → STOP	Die CPU geht in folgenden Fällen wieder in den STOP-Zustand: • Im RUN-Zustand wird ein Fehler erkannt und der zugehörige OB ist nicht geladen. • Die CPU wird über den Betriebsartenschalter oder vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anwenderprogramm bearbeitet. • Die Kommunikationsfunktion "STOP" wird ausgeführt.

Priorität der Betriebszustände

Funktion

Wenn mehrere Betriebszustandswechsel gleichzeitig angefordert werden, wird in den Betriebszustand mit der höchsten Priorität gewechselt. Steht z. B. der Betriebsartenschalter auf "RUN", und vom Programmiergerät aus wird versucht, die CPU in "STOP" zu schalten, geht die CPU in "STOP", weil dieser Betriebszustand die höchste Priorität hat.

Priorität	Betriebszustand
Höchste	STOP (Seite 755)
	HALT (Seite 765)
	ANLAUF (Seite 756)
Niedrigste	RUN (Seite 765)

Betriebszustand "STOP"

Funktion

Im Betriebszustand "STOP" wird das Anwenderprogramm nicht bearbeitet. Alle Ausgänge werden auf Ersatzwerte gesetzt und damit der gesteuerte Prozess in einen sicheren Betriebszustand gebracht. Die CPU prüft folgende Punkte:

- Hardware, z. B. ob alle Baugruppen verfügbar sind
- Ob für die CPU die Standardeinstellungen gelten oder Parametersätze vorliegen
- Ob die Randbedingungen für das programmierte Anlaufverhalten stimmen

Im STOP-Zustand können Sie Globaldaten empfangen. Außerdem ist eine passive einseitige Kommunikation über Kommunikations-Systemfunktionsbausteinen für projektierte

Verbindungen und über Kommunikations-Systemfunktionen für nicht projektierte Verbindungen möglich.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 753)

Betriebszustand "ANLAUF"

Grundlagen zum Betriebszustand "ANLAUF"

Funktion

Bevor die CPU nach dem Einschalten mit der Bearbeitung des Anwenderprogramms beginnt, wird ein Anlaufprogramm bearbeitet. Im Anlaufprogramm können Sie durch entsprechende Programmierung von Anlauf-OBs bestimmte Voreinstellungen für Ihr zyklisches Programm festlegen.

Es gibt folgende Anlaufarten:

- Neustart/Warmstart
- Wiederanlauf
- Kaltstart

Besonderheiten

Beachten Sie für den Betriebszustand "ANLAUF" folgende Punkte:

- Das Programm im Anlauf-OB wird abgearbeitet. Dabei steht der OB 100 für "Neustart (Warmstart)", OB 101 für Wiederanlauf und der OB 102 für "Kaltstart".
- Keine zeit- und alarmgesteuerte Programmbearbeitung möglich.
- Die Zeiten werden aktualisiert.
- Der Betriebsstundenzähler läuft.
- Die Ausgänge auf Baugruppen sind gesperrt.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 753)

Wiederanlauf (Seite 758)

Kaltstart (Seite 757)

Anlauf-OBs (OB 100, OB 101 und OB 102) (Seite 919)

Gesamtübersicht des Anlaufs (Seite 763)

Kaltstart

Kaltstart

Beim Kaltstart werden alle Daten (Prozessabbild, Merker, Zeiten, Zähler und Datenbausteine) auf die im Programm (Ladespeicher) hinterlegten Startwerte zurückgesetzt - unabhängig davon, ob sie als remanent oder nicht remanent parametriert wurden.

Die Programmbearbeitung wird wieder am Anfang (Anlauf-OB oder OB 1) begonnen.

Neustart (Warmstart)

Zulässiger Neustart (Warmstart)

In folgenden Fällen können Sie einen "Neustart (Warmstart)" durchführen:

- Nach dem Urlöschen
- Nach dem Laden des Anwenderprogramms im STOP-Zustand der CPU

Sie können keinen Neustart (Warmstart) durchführen, wenn vom System Urlöschen angefordert wurde.

Manueller Neustart (Warmstart)

Ein manueller "Neustart (Warmstart)" wird ausgelöst über:

- Den Betriebsartenschalter
- Einen Menübefehl vom PG aus bzw. über Kommunikationsfunktionen, wenn der Betriebsartenschalter auf "RUN" steht.

Automatischer Neustart (Warmstart)

Ein automatischer "Neustart (Warmstart)" kann in folgenden Fällen bei "NETZ-EIN" ausgelöst werden:

- Die CPU war bei "NETZ-AUS" nicht in "STOP".
- Der Betriebsartenschalter steht auf "RUN".
- Die CPU wurde im "Neustart (Warmstart)" durch Netzausfall unterbrochen. Dies ist unabhängig von der Parametrierung der Anlaufart.

Ungepufferter automatischer Neustart (Warmstart)

Wird Ihre CPU ohne Pufferbatterie betrieben, z. B. wenn ein wartungsfreier Betrieb notwendig ist, wird nach dem Einschalten oder bei Spannungswiederkehr nach "NETZ-AUS" die CPU automatisch urgelöscht und anschließend ein "Neustart (Warmstart)" durchgeführt. Das Anwenderprogramm muss auf Flash-EPROM (Memory Card) vorhanden sein, damit das Anwenderprogramm wieder in den Arbeitsspeicher kopiert werden kann.

Wiederanlauf

Wiederanlauf

Diese Anlaufart ist nur möglich bei S7-400. Der Wiederanlauf ist durch folgende Eigenschaften gekennzeichnet:

- Beim Wiederanlauf behalten alle Daten inklusive des Prozessabbildes ihren zuletzt gültigen Wert.
- Die Programmbearbeitung wird genau mit dem Befehl fortgesetzt, bei dem die Unterbrechung eingetreten ist.
- Bis zum Ende des aktuellen Zyklus werden die Ausgänge nicht verändert.
- Bei Unterbrechung der Stromversorgung steht der Wiederanlauf nur bei gepuffertem Betrieb zur Verfügung.

Nach Spannungsausfall im RUN und anschließender Spannungswiederkehr durchlaufen S7-400-CPUs eine Initialisierungsroute und führen dann automatisch einen Wiederanluaf durch. Bei einem Wiederanlauf wird das Anwenderprogramm an der Stelle fortgesetzt, an der die Bearbeitung unterbrochen wurde. Der Teil des Anwenderprogramms, der vor Netzausfall nicht mehr bearbeitet wurde, wird als Restzyklus bezeichnet. Der Restzyklus kann auch zeit- und alarmgesteuerte Programmteile enthalten.

Ein Wiederanlauf ist prinzipiell nur zulässig, wenn das Anwenderprogramm im STOP-Zustand nicht verändert wurde (z. B. durch Nachladen eines geänderten Bausteins) bzw. wenn nicht aus anderen Gründen Neustart (Warmstart) erforderlich ist. Man unterscheidet zwischen manuellem und automatischem Wiederanlauf.

Manueller Wiederanlauf

Ein manueller Wiederanlauf ist nur möglich bei entsprechender Parametrierung der CPU und nach den folgenden STOP-Ursachen:

- der Betriebsartenschalter wurde von RUN auf STOP gesetzt
- vom Anwender programmierte STOPs, STOPs nach Aufruf nicht geladener OBs
- STOP-Zustand wurde vom PG oder durch Kommunikationsfunktionen herbeigeführt

Ein manueller Wiederanlauf kann ausgelöst werden über Menübefehl vom PG aus bzw. über Kommunikationsfunktionen (wenn der Betriebsartenschalter auf RUN oder RUN-P steht). Vorraussetzung dafür ist, dass der Wiederanlauf durch Bedienung nicht gesperrt ist (Anlaufparameter).

- wenn der Parametersatz der CPU manueller Wiederanlauf parametriert wurde

Automatischer Wiederanlauf

Ein automatischer Wiederanlauf wird ausgelöst bei NETZ-EIN, wenn die CPU bei NETZ-AUS nicht im STOP oder HALT war. Vorraussetzung dafür ist, dass der Wiederanlauf durch Bedienung nicht gesperrt ist (Anlaufparameter).

Remanenzverhalten nach Netzausfall

Remanente Datenbereiche nach Netzausfall

Um Datenverlust bei Spannungsausfall zu vermeiden, können Sie die Remanenz von Merkern, Zeiten, Zählern und Bereichen in Datenbausteinen festlegen. Vorausgesetzt, die Anlaufart ist "Neustart (Warmstart)" - das ist die Voreinstellung - erfolgt bei Spannungswiederkehr ein "automatischer Neustart mit Gedächtnis".

S7-300- und S7-400-CPUs reagieren unterschiedlich bei Spannungswiederkehr nach Netzausfall.

S7-300-CPUs kennen nur die Anlaufart "Neustart". Mit STEP 7 können Sie die Remanenz vom Merkern, Zeiten, Zählern und Bereichen in Datenbausteinen festlegen, um Datenverlust bei Spannungsausfall zu vermeiden. Bei Spannungswiederkehr erfolgt ein "automatischer Neustart mit Gedächtnis".

S7-400-CPUs reagieren auf Spannungswiederkehr abhängig von der Parametrierung entweder mit Neustart (Warmstart) (nach gepuffertem oder ungepuffertem NETZ-EIN) oder mit Wiederanlauf (nur nach gepuffertem NETZ-EIN möglich).

Remanenzverhalten bei CPUs mit Pufferung

Die folgende Tabelle zeigt das Remanenzverhalten im Arbeitsspeicher der S7-300- und S7-400-CPUs mit Pufferung bei "Neustart (Warmstart)" und "Kaltstart":

Daten	Neustart (Warmstart)	Kaltstart	Wiederanlauf
Bausteine im Ladespeicher	X	X	X
DB im Arbeitsspeicher	X	0	X
Merker, Zeiten, Zähler remanent parametriert	X	0	X
Merker, Zeiten, Zähler nicht remanent parametriert	0	0	X
X: Daten bleiben erhalten 0: Daten werden zurückgesetzt bzw. gelöscht (Inhalt von DBs)			

Remanenzverhalten bei CPUs ohne Pufferung

Bei S7-400-CPUs ohne Pufferung ist kein Wiederanlauf erlaubt!

Die folgende Tabelle zeigt das Remanenzverhalten im Arbeitsspeicher der S7-300-CPUs ohne Pufferung bei "Neustart (Warmstart)" und "Kaltstart":

Daten	Neustart (Warmstart)		Kaltstart	
	S7-300	S7-400	S7-300	S7-400
Bausteine im Ladespeicher	VC	VC	VC	VC
DB im Arbeitsspeicher remanent parametriert	VX	nicht möglich	V	nicht möglich
DB im Arbeitsspeicher nicht remanent parametriert	V	V	V	V

Daten	Neustart (Warmstart)		Kaltstart	
	S7-300	S7-400	S7-300	S7-400
Merker, Zeiten, Zähler remanent parametriert	X	0	0	0
Merker, Zeiten, Zähler nicht remanent parametriert	0	0	0	0
X: Daten bleiben erhalten 0: Daten werden zurückgesetzt bzw. gelöscht (Inhalt von DBs) V: Daten werden auf den Vorbesetzungswert aus dem EPROM-Speicher gesetzt VC: Codebaustein im EPROM bleibt erhalten, eventuell überladener Codebaustein geht verloren VX: Datenbaustein bleibt nur erhalten, wenn im EPROM vorhanden, remanente Daten werden aus NV-RAM übernommen (geladene oder erzeugte Datenbausteine im RAM gehen verloren)				

Anlauftätigkeiten

Übersicht

Die folgende Tabelle zeigt, welche Tätigkeiten die CPU beim Anlauf durchführt:

Tätigkeiten in der Bearbeitungsreihenfolge	bei Neustart (Warmstart)	bei Kaltstart	bei Wiederanlauf
Unterbrechungs- und Baustein-Stack löschen	X	X	0
Nicht remanente Merker, Zeiten, Zähler löschen	X	0	0
Alle Merker, Zeiten, Zähler löschen	0	X	0
Prozessabbild der Ausgänge löschen	X	X	parametrierbar
Ausgänge der Signalbaugruppen zurücksetzen	X	X	parametrierbar
Prozessalarme verwerfen	X	X	0
Verzögerungsalarme verwerfen	X	X	0
Diagnosealarme verwerfen	X	X	X
Systemzustandsliste (SZL) aktualisieren	X	X	X
Baugruppenparameter auswerten und an Baugruppen übergeben oder Standardwerte übergeben	X	X	X
Bearbeitung des jeweiligen Anlauf-OB	X	X	X
Restzyklus (Teil des Anwenderprogramms, das wegen eines "NETZ-AUS" nicht weiterbearbeitet werden konnte) bearbeiten	0	0	X

Tätigkeiten in der Bearbeitungsreihenfolge	bei Neustart (Warmstart)	bei Kaltstart	bei Wiederanaluf
Prozessabbild der Eingänge aktualisieren	X	X	X
Digitale Ausgänge freigeben (Signal OD (Output Disabled) aufheben) nach Betriebszustandsübergang in "RUN"	X	X	X
X: Tätigkeit wird durchgeführt 0: Tätigkeit wird nicht durchgeführt			

Siehe auch

Gesamtübersicht des Anlaufs (Seite 763)

Abbrechen eines Anlaufs

Gründe für den Abbruch

Wenn während des Anlaufs Fehler auftreten, wird der Anlauf abgebrochen, und die CPU geht oder bleibt in "STOP". Das hat folgende Auswirkungen auf den Betriebszustand Neustart (Warmstart):

- Ein abgebrochener Neustart (Warmstart) muss wiederholt werden.
- Bei einem angebrochenen Wiederanlauf ist sowohl ein Neustart (Warmstart) als auch ein Wiederanlauf möglich.

- Ein Neustart (Warmstart) oder Wiederanlauf wird unter folgenden Bedingungen nicht durchgeführt oder abgebrochen:
  - Wenn der Betriebsartenschalter der CPU auf "STOP" steht.
  - Wenn Urlöschen angefordert wird.
  - Wenn eine Memory Card gesteckt ist, deren Anwendungskennung nicht zulässig ist.
  - Wenn im Einzelprozessorbetrieb mehr als eine CPU gesteckt ist.
  - Wenn das Anwenderprogramm einen Organisationsbaustein enthält, den die CPU nicht kennt oder der gesperrt wurde.
  - Wenn nach dem Einschalten der Netzspannung die CPU feststellt, dass nicht alle Baugruppen, die in der mit STEP 7 erstellten Konfiguration aufgeführt sind, tatsächlich gesteckt sind (Parametrierung Soll-Ist-Unterschied unzulässig).
  - Wenn bei der Parametrierung der Baugruppen Fehler auftreten.
- Ein Wiederanlauf wird unter folgenden Bedingungen nicht durchgeführt oder abgebrochen:
  - Wenn die CPU zuvor urgelöscht wurde (nach dem Urlöschen ist nur Neustart (Warmstart) zulässig).
  - Wenn die Unterbrechungsgrenze überschritten ist (Die Unterbrechungszeit ist die Zeit, die nach dem Verlassen des RUN vergeht bis der Anlauf-OB inklusive Restzykluszeit bearbeitet wurde.).
  - Wenn die Baugruppenkonfiguration geändert wurde (z. B. Baugruppentausch).
  - Wenn die Parametrierung nur Neustart (Warmstart) zugelassen ist.
  - Wenn im STOP Bausteine geladen, gelöscht oder geändert wurden.

Gesamtübersicht des Anlaufs

Ablauf

Das folgende Bild zeigt die Tätigkeiten der CPU in den Betriebszuständen "STOP", "ANLAUF" und "RUN".

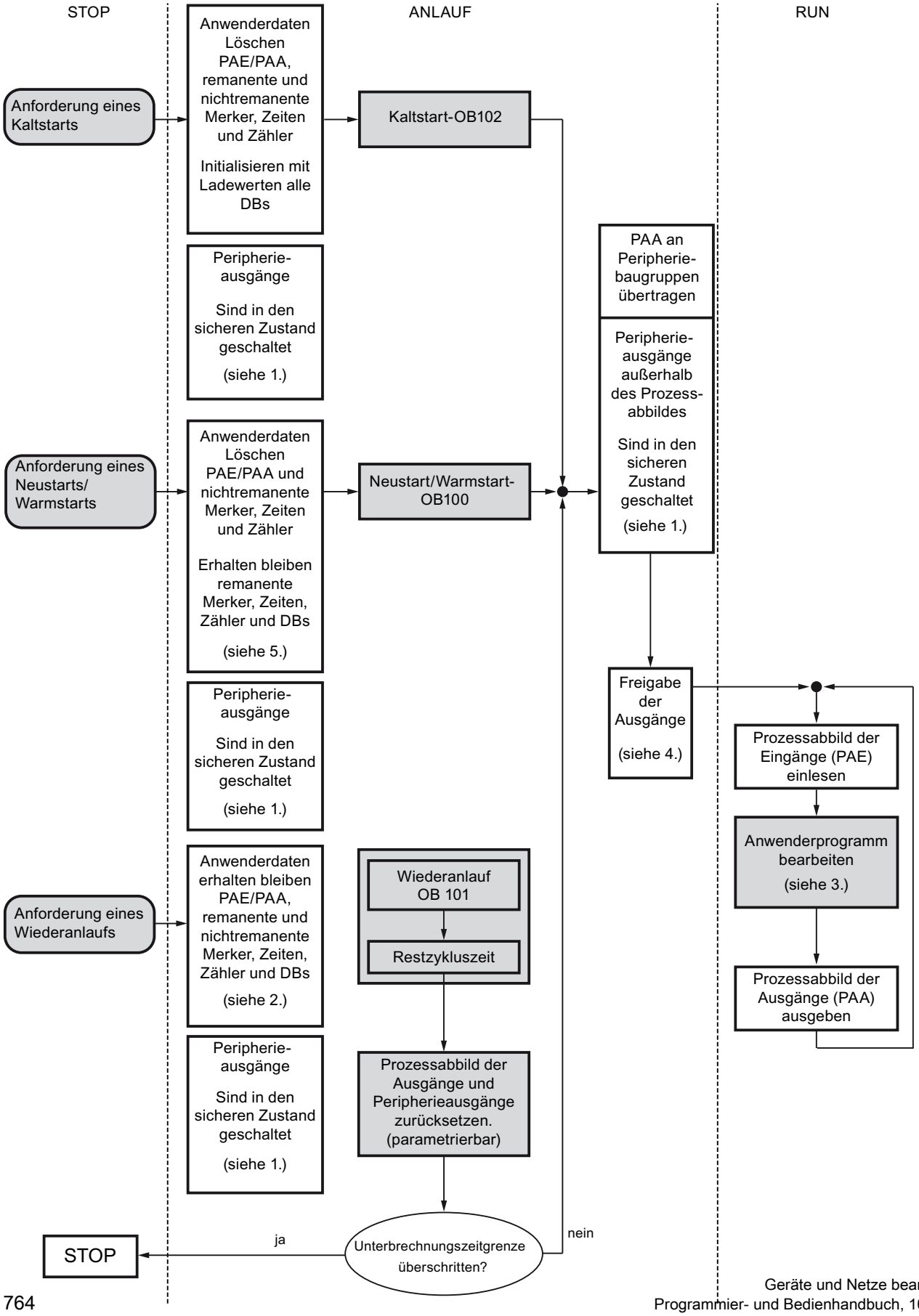

= Verhalten ist durch Anwenderfunktionen beeinflussbar

Legende zum Bild:

1. Alle Peripherieausgänge sind durch die Peripheriebaugruppen hardwareseitig in den sicheren Zustand geschaltet (Defaultwert = "0"). Dies ist unabhängig davon, ob sie im Anwenderprogramm innerhalb oder außerhalb des Prozessabbildbereiches verwendet werden.
   Beim Einsatz von ersatzwertfähigen Signalbaugruppen kann das Verhalten der Ausgänge parametriert werden, z. B. letzter Wert halten.
2. Ist notwendig für die Bearbeitung des Restzyklus.
3. Auch den Alarm-OBs steht beim erstmaligen Aufruf ein aktuelles Prozessabbild der Eingänge zur Verfügung.
4. Den Zustand der zentralen und dezentralen Peripherieausgänge im ersten Zyklus des Anwenderprogramms können Sie durch folgende Maßnahmen festlegen:
   – Parametrierbare Ausgabebaugruppen verwenden, um Ersatzwerte ausgeben zu können oder um den letzen Wert zu halten.
   – Bei Wiederanlauf: CPU-Anlauf-Parameter "Ausgänge zurücksetzen bei Wiederanlauf" aktivieren, um eine "0" auszugeben (entspricht der Voreinstellung).
   – Ausgänge im Anlauf-OB (OB 100, OB 101, OB 102) vorbesetzen.
5. In S7-300-Systemen bleiben nur die als remanent projektierten DB-Bereiche erhalten.

Betriebszustand "RUN"

Funktion

Im Betriebszustand "RUN" erfolgt die zyklische, zeit- und alarmgesteuerte Programmbearbeitung:

- Das Prozessabbild der Eingänge wird eingelesen.
- Das Anwenderprogramm wird abgearbeitet.
- Das Prozessabbild der Ausgänge wird ausgegeben.

Der aktive Austausch von Daten zwischen CPUs über die Globaldaten-Kommunikation (Globaldatentabelle) und über Kommunikationsanweisungen für projektierte Verbindungen und über Kommunikationsanweisungen für nicht projektierte Verbindungen ist nur im RUN-Zustand möglich.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 753)

Betriebszustand "HALT"

Funktion

Der Betriebszustand "HALT" nimmt eine Sonderstellung ein, da er nur für Tests in "ANLAUF" oder "RUN" verwendet wird.

Betriebszustand "HALT":

- Alle Zeiten werden eingefroren. Zeiten und Betriebsstundenzähler werden nicht bearbeitet, Überwachungszeiten und die Grundtakte der zeitgesteuerten Ebenen werden angehalten.
- Die Echtzeituhr läuft.
- Ausgänge werden nicht freigeschaltet, Sie können sie aber zu Testzwecken freigeben.
- Ein- und Ausgänge können gesteuert werden.
- Gepufferte CPUs gehen bei Netzausfall und -wiederkehr von "HALT" in "STOP" und führen keinen "Neustart (Warmstart)" aus. Ungepufferte CPUs führen bei Netzwiederkehr einen ungepufferten automatischen "Neustart (Warmstart)" aus.
- Sie können Globaldaten empfangen. Außerdem ist eine passive einseitige Kommunikation über Kommunikations-SFBs für projektierte Verbindungen und über Kommunikations-SFCs für nicht projektierte Verbindungen möglich.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 753)

Grundlagen zum Urlöschen

Funktion

Im STOP-Zustand kann die CPU urgelöscht werden. Urlöschen kann manuell über den Betriebsartenschalter (MRES) oder vom Programmiergerät aus durchführen, z. B. vor dem Laden eines Anwenderprogramms.

Durch Urlöschen wird die CPU in den "Urzustand" versetzt. Das bedeutet:

- Das gesamte Anwenderprogramm im Arbeitsspeicher und im RAM-Ladespeicher sowie alle Operandenbereiche werden gelöscht.
- Die Systemparameter sowie die CPU- und Baugruppenparameter werden auf die Standardeinstellung zurückgesetzt. Die vor dem Urlöschen eingestellten MPI-Parameter bleiben erhalten.
- Wenn eine Memory Card (Flash-EPROM) bzw. Micro Memory Card gesteckt ist, kopiert die CPU das Anwenderprogramm aus der Memory Card in den Arbeitsspeicher. Wenn sich die entsprechenden Konfigurationsdaten ebenfalls auf der Memory Card befinden, werden auch die CPU- und Baugruppenparameter kopiert.
- Der Diagnosepuffer, die MPI-Parameter, die Uhrzeit und der Betriebsstundenzähler werden nicht zurückgesetzt.

Speicherbereiche

Wissenwertes zu Micro Memory Cards für S7-300

Funktion der Micro Memory Card

Die SIMATIC Micro Memory Card ist ein Speichermodul für eine S7-300-CPU. Sie könne die Micro Memory Card prinzipiell als Ladespeicher oder als transportablen Datenträger einsetzen.

Mit STEP 7 V11 wird der Einsatz als Ladespeicher unterstützt.

Für den Betrieb müssen Sie die SIMATIC Micro Memory Card in die CPU gesteckt haben.

ACHTUNG
Formatieren der Micro Memory Card Wenn Sie die Micro Memory Card mit Windows-Mitteln formatieren, z. B. mit einem handelsüblichen Kartenleser, machen Sie die Micro Memory Card als Speichermedium für eine S7-CPU unbrauchbar.

Was auf einer SIMATIC Micro Memory Card gespeichert wird

Auf einer Micro Memory Card speichern Sie das Anwenderprogramm (Bausteine und Systemdaten).

Alternativ können Sie die Micro Memory Card auch für die Firmware-Aktualisierung bzw. für die Sicherung der CPU-Firmware nutzen.

Objekte aus dem Projekt auf Micro Memory Card übertragen

Wenn die Micro Memory Card im PG bzw. im externen Prommer steckt, können Sie folgende Objekte aus der Projektnavigation auf die Micro Memory Card übertragen:

- Einzelne Bausteine (Mehrfachselektion möglich)
  In diesem Fall wird ein konsistentes Übertragen angeboten, d. h. die Abhängigkeiten der Bausteine untereinander durch Bausteinaufrufe wird berücksichtigt.
- PLC
  In diesem Fall werden alle ablaufrelevanten Objekte, u. a. Bausteine und die Hardware-Konfiguration auf die Micro Memory Card übertragen - wie beim Laden.

Um die Übertragung durchzuführen, können Sie die Objekte per Drag&Drop übertragen oder Sie verwenden den Befehl "Auf Speicherkarte schreiben" im Menü "Projekt".

Objekte von der Micro Memory Card in das Projekt übertragen

Einzelne Bausteine (Mehrfachselektion möglich) übertragen Sie per Drag&Drop in das Projekt. Eine Hardware-Konfiguration ist nicht von der Memory Card ins Projekt übertragbar.

Firmware sichern auf der Micro Memory Card

Wir empfehlen Ihnen, eine Sicherungskopie der Firmware für Notfälle zu erstellen.

Dazu muss eine leere Micro Memory Card in der CPU stecken und eine bestimmte Bedienreihenfolge eingehalten werden: Die Beschreibung finden Sie in der Dokumentation zur CPU.

Dokumentation zu S7-300 (http://support.automation.siemens.com/WW/view/de/13008499)

Firmware aktualisieren mit einer Micro Memory Card

Die neueste Firmware erhalten Sie als *.UPD-Datei aus dem Internet über die Seiten des Service & Support:

http://support.automation.siemens.com (http://support.automation.siemens.com)

Speichern Sie die Firmware-Dateien auf der Festplatte.

Um die Datei auf die Micro Memory Card zu speichern, wählen Sie den Befehl "Firmware-Update Speicherkarte erstellen" im Menü "Projekt".

Folgen Sie anschließend den Anweisungen im Service & Support-Portal zur Durchführung des Firmware-Updates mit ihrer CPU.

Siehe auch

Speicherbereiche der S7-300-CPUs (Seite 771)

Wissenwertes zu Memory Cards für S7-400

Funktion der Memory Card

Bei allen CPUs der S7-400 können Sie eine Memory Card stecken. Sie stellt eine Erweiterung des Ladespeichers dar. Je nach Art, der verwendeten Memory Card, bleibt das Anwenderprogramm auch im spannungslosen Zustand auf der Memory Card erhalten.

Arten von Memory Cards

- RAM-Card: ausschließlich zur Erweiterung des integrierten Ladespeichers.
- FLASH-Card: zur Speicherung des Anwenderprogramms auch im spannungslosen Zustand ohne Pufferung oder außerhalb der CPU.

Objekte aus dem Projekt auf Memory Card übertragen

Wenn die Memory Card im PG bzw. im externen Prommer steckt, können Sie folgende Objekte aus der Projektnavigation auf die Memory Card übertragen:

- Einzelne Bausteine (Mehrfachselektion möglich)
  In diesem Fall wird ein konsistentes Übertragen angeboten, d. h. die Abhängigkeiten der Bausteine untereinander durch Bausteinaufrufe wird berücksichtigt.
- PLC
  In diesem Fall werden alle ablaufrelevanten Objekte, u. a. Bausteine und die Hardware-Konfiguration auf die Memory Card übertragen - wie beim Laden.

Um die Übertragung durchzuführen, können Sie die Objekte per Drag&Drop übertragen oder Sie verwenden den Befehl "Auf Speicherkarte schreiben" im Menü "Projekt" und folgen Sie ggf. den Anweisungen im Folgedialog.

Objekte von der Memory Card in das Projekt übertragen

Einzelne Bausteine (Mehrfachselektion möglich) übertragen Sie per Drag&Drop in das Projekt. Eine Hardware-Konfiguration ist nicht von der Memory Card ins Projekt übertragbar.

Anwenderprogramm auf Memory Card laden

Mit dem Befehl "Anwenderprogramm auf Memory Card laden" im Menü "Online" speichern Sie das gesamte Anwenderprogramm auf die FLASH-Card. Die FLASH-Card befindet sich in der CPU. Vor dem Laden wird alles gelöscht, was sich vorher auf der FLASH-Card befunden hat.

- Die CPU muss sich im Betriebszustand STOP befinden
- Sie können nur FLASH-Cards mit diesem Menübefehl laden

Firmware aktualisieren mit einer Memory Card

Die neueste Firmware erhalten Sie als *.UPD-Datei aus dem Internet über die Seiten des Service & Support:

http://support.automation.siemens.com (http://support.automation.siemens.com)

Speichern Sie die Firmware-Dateien auf der Festplatte.

Um die Datei auf die Memory Card zu speichern, wählen Sie den Befehl "Firmware-Update Speicherkarte erstellen" im Menü "Projekt".

Folgen Sie anschließend den Anweisungen im Service & Support-Portal zur Durchführung des Firmware-Updates mit ihrer CPU.

Siehe auch

Speicherbereiche der S7-400-CPUs (Seite 773)

Aufteilung der Speicherbereiche

Aufteilung

Der Speicher der S7-CPUs lässt sich in drei Bereiche aufteilen:

- Ladespeicher (je nach CPU- bzw. Memory-Card-Typ als RAM-, EPROM- oder Flash-Speicher ausgeführt)
- Arbeitsspeicher (integriertes RAM)
- Systemspeicher (integriertes RAM)

Speicherbereiche S7-300

Speicherbereiche der S7-300-CPUs

Die drei Speicherbereiche Ihrer CPU

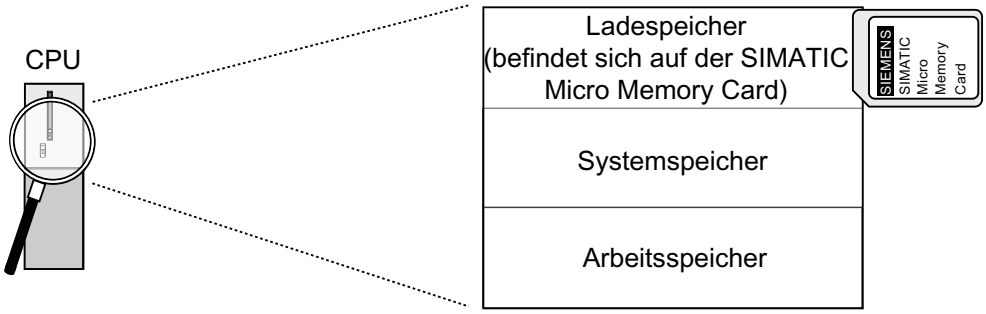

Ladespeicher

Der Ladespeicher befindet sich auf der SIMATIC Micro Memory Card und entspricht genau der Größe der SIMATIC Micro Memory Card. Er dient zur Aufnahme von Code- und Datenbausteinen sowie von Systemdaten (Konfiguration, Verbindungen, Baugruppenparameter, usw.). Bausteine, die als nicht ablaufrelevant gekennzeichnet sind werden ausschließlich in den Ladespeicher aufgenommen. Zusätzlich können die kompletten Projektierungsdaten eines Projekts auf der SIMATIC Micro Memory Card abgelegt werden.

Hinweis

Das Laden von Anwenderprogrammen und damit der Betrieb der CPU ist nur möglich, wenn Sie eine SIMATIC Micro Memory Card in die CPU gesteckt haben.

Systemspeicher

Der Systemspeicher ist in der CPU integriert und nicht erweiterbar.

Er enthält

- die Operandenbereiche Merker, Zeiten und Zähler
- die Prozessabbilder der Ein- und Ausgänge
- die Lokaldaten

Arbeitsspeicher

Der Arbeitsspeicher ist in der CPU integriert und nicht erweiterbar. Er dient zur Abarbeitung des Codes sowie zur Bearbeitung der Daten des Anwenderprogramms. Die Programmbearbeitung erfolgt ausschließlich im Bereich von Arbeitsspeicher und Systemspeicher.

Siehe auch

Wissenwertes zu Micro Memory Cards für S7-300 (Seite 767)

Remanenz des Lade-, System- und Arbeitsspeichers

Ihre CPU besitzt einen wartungsfreien remanenten Speicher, d. h. Sie benötigen keine Pufferbatterie für den Betrieb. Durch die Remanenz bleibt der Inhalt von remanentem Speicher auch über NETZ-AUS und Neustart (Warmstart) hinweg erhalten.

Remanente Daten im Ladespeicher

Ihr Programm im Ladespeicher ist immer remanent: Es wird bereits beim Laden netzausfallsicher und urlöschfest auf der SIMATIC Micro Memory Card hinterlegt.

Remanente Daten im Systemspeicher

Für Merker, Zeiten und Zähler bestimmen Sie durch Projektierung (Eigenschaften der CPU), welche Teile remanent sein sollen und welche bei Neustart (Warmstart) mit "0" initialisiert werden sollen.

Diagnosepuffer, MPI-Adresse und Übertragungsgeschwindigkeit (Baudrate) sowie Betriebsstundenzähler sind generell im remanenten Speicherbereich auf der CPU abgelegt. Mit der Remanenz der MPI-Adresse und Übertragungsgeschwindigkeit wird sichergestellt, dass Ihre CPU nach einem Stromausfall, nach Urlöschen oder nach Verlust der Kommunikationsparametrierung (durch Ziehen der SIMATIC Micro Memory Card oder Löschen der Kommunikationsparameter) noch kommunikationsfähig ist.

Remanente Daten im Arbeitsspeicher

Inhalte von remanenten DBs sind bei Neustart und NETZ-AUS-EIN grundsätzlich remanent. Remanente Datenbausteine können bis zur maximalen Remanenzgrenze des Arbeitsspeichers in den Arbeitsspeicher geladen werden.

Nicht remanente DBs werden bei Neustart und NETZ-AUS-EIN mit ihren Anfangswerten aus dem Ladespeicher initialisiert. Nicht remanente Datenbausteine und Code-Bausteine können bis zur maximalen Grenze des Arbeitsspeichers geladen werden.

Die Größe des remanenten Arbeitsspeichers (für remanente Datenbausteine) der CPU finden Sie in den Technischen Daten der jeweiligen CPUs.

Speicherbereiche S7-400

Speicherbereiche der S7-400-CPUs

Aufteilung der Speicherbereiche

Der Speicher der S7-400-CPUs lässt sich in folgende Bereiche aufteilen:

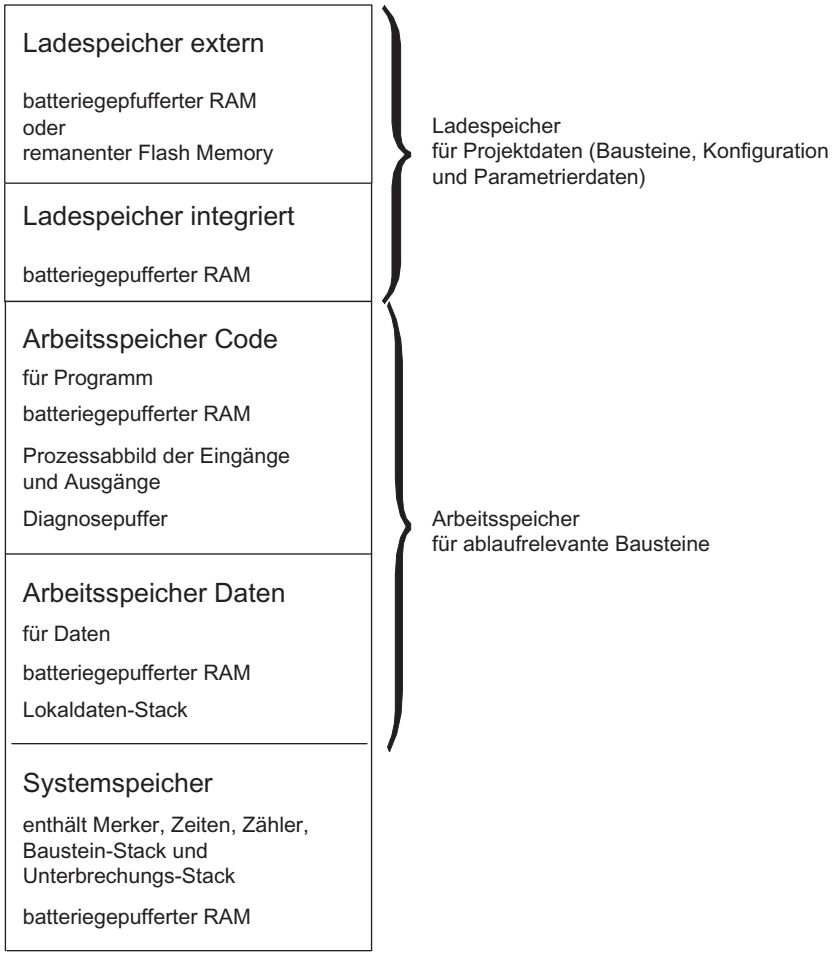

Speichertypen bei S7-400-CPUs

- Ladespeicher für die Projektdaten, z. B. Bausteine, Konfiguration und Parametrierdaten.
- Arbeitsspeicher für die ablaufrelevanten Bausteine (Codebausteine und Datenbausteine).

- Systemspeicher (RAM) enthält die Speicherelemente, die jede CPU dem Anwenderprogramm zur Verfügung stellt, wie z. B.: Merker, Zeiten und Zähler. Außerdem enthält der Systemspeicher den Baustein-Stack und den Unterbrechungs-Stack.
- Systemspeicher der CPU stellt außerdem temporären Speicher (Lokaldaten-Stack, Diagnosepuffer und Kommunikationsressourcen) zur Verfügung, der dem Programm beim Aufrufen eines Bausteins für dessen temporäre Daten zugeordnet wird. Diese Daten sind nur so lange gültig, solange der Baustein aktiv ist.
  Durch Veränderung der Defaultwerte für Prozessabbild, Lokaldaten, Diagnosepuffer und Kommunikationsressourcen (parametrierbare Eigenschaften der CPU) beeinflussen Sie den für ablaufrelevante Bausteine verfügbaren Arbeitsspeicher.

Hinweis

Beachten Sie Folgendes, wenn Sie das Prozessabbild einer CPU vergrößern. Projektieren Sie Baugruppen, deren Adressen über der höchsten Adresse des Prozessabbilds liegen müssen, so um, dass die neuen Adressen weiterhin über der größten Adresse des vergrößerten Prozessabbildes liegen.

Wichtiger Hinweis für CPUs nach geänderter Parametrierung der Aufteilung des Arbeitsspeichers

Wenn Sie die Aufteilung des Arbeitsspeichers per Parametrierung ändern, dann wird beim Laden der Systemdaten in die CPU der Arbeitsspeicher reorganisiert. Das hat zur Folge, dass Datenbausteine, die per SFC erzeugt wurden, gelöscht werden und die übrigen Datenbausteine mit Initialwerten aus dem Ladespeicher vorbesetzt werden.

Die nutzbare Größe des Arbeitsspeichers für Code- bzw. Datenbausteine wird beim Laden der Systemdaten geändert, wenn Sie folgende Parameter ändern:

- Größe des Prozessabbildes (byteweise; im Register "Zyklus/Taktmerker")
- Kommunikationsressourcen (nur S7-400; im Register "Speicher")
- Größe des Diagnosepuffers (im Register "Diagnose/Uhr")
- Anzahl Lokaldaten für alle Prioritätsklassen (Register "Speicher")

Berechnungsgrundlage für die Abschätzung des benötigten Arbeitsspeichers

Damit Sie die zur Verfügung stehende Größe des Arbeitsspeichers der CPU nicht überschreiten, müssen Sie bei der Parametrierung folgenden Speicherbedarf berücksichtigen:

Tabelle 1-82 Speicherbedarf

Parameter	Benötigter Arbeitsspeicher	In Code-/ Datenspeicher
Größe des Prozessabbildes (Eingänge)	12 Byte je Byte im Prozessabbild der Eingänge Ab V6.0: 20 Byte je Byte im Prozessabbild der Eingänge	Codespeicher
Größe des Prozessabbildes (Ausgänge)	12 Byte je Byte im Prozessabbild der Ausgänge Ab V6.0: 20 Byte je Byte im Prozessabbild der Eingänge	Codespeicher
Kommunikationsressourcen (Kommunikationsaufträge)	72 Byte je Kommunikationsauftrag	Codespeicher

Parameter	Benötigter Arbeitsspeicher	In Code-/ Datenspeicher
Größe des Diagnosepuffers	32 Byte je Eintrag im Diagnosepuffer	Codespeicher
Anzahl Lokaldaten	1 Byte je Byte Lokaldaten	Datenspeicher

Flexible Speichergröße

- Arbeitsspeicher:
  Die Größe des Arbeitsspeichers ist durch die Auswahl der passenden CPU aus dem abgestuften CPU-Spektrum bestimmt.
- Ladespeicher:
  Für kleine und mittlere Programme reicht der integrierte Ladespeicher aus.
  Für größere Programme kann der Ladespeicher durch Zustecken einer RAM-Memory-Card vergrößert werden.
  Zusätzlich stehen Flash Memory Cards zur Verfügung, um Programme auch ohne Pufferbatterie bei Spannungsausfall zu erhalten. Außerdem eignen sich solche Flash Memory Cards (ab 8 MB) zum Versenden und Durchführen von Betriebssystem-Updates.

Pufferung

- Die Pufferbatterie puffert den integrierten und externen Teil des Ladespeichers, den Datenteil des Arbeitsspeichers und den Codeteil.

Siehe auch

Wissenwertes zu Memory Cards für S7-400 (Seite 769)

Inhalte von Arbeits- und Systemspeicher

Operandenbereiche der CPUs

Übersicht

Speicherelemente, die jede CPU dem Anwenderprogramm zur Verfügung stellt, sind z. B. das Prozessabbild der Ein- und Ausgänge, Merker, Zeiten und Zähler. Der Arbeits- bzw. Systemspeicher der S7-CPUs ist in Operandenbereiche aufgeteilt. Durch Verwendung der entsprechenden Operationen adressieren Sie in Ihrem Programm die Daten direkt in den jeweiligen Operandenbereich.

Die folgende Tabelle zeigt die Operandenbereiche:

Operandenbereich	Zugriff über Einheiten der folgenden Größe:	S7- Notation	Beschreibung
Prozessabbild der Eingänge	Eingang (Bit)	E	Zu Beginn jedes Zyklus liest die CPU die Eingänge aus den Eingabebaugruppen und speichert die Werte in das Prozessabbild der Eingänge.
	Eingangsbyte	EB	
	Eingangswort	EW	
	Eingangsdoppelwort	ED	
Prozessabbild der Ausgänge	Ausgang (Bit)	A	Während des Zyklus berechnet das Programm die Werte für die Ausgänge und legt sie im Prozessabbild der Ausgänge ab. Am Ende des Zyklus schreibt die CPU die errechneten Ausgangswerte in die Ausgabebaugruppen.
	Ausgangsbyte	AB	
	Ausgangswort	AW	
	Ausgangsdoppelwort	AD	
Merker	Merker (Bit)	M	Dieser Bereich stellt Speicherplatz für im Programm errechnete Zwischenergebnisse zur Verfügung.
	Merkerbyte	MB	
	Merkerwort	MW	
	Merkerdoppelwort	MD	
Zeiten	Zeit (T)	T	In diesem Bereich stehen Zeiten zur Verfügung.
Zähler	Zähler (Z)	Z	In diesem Bereich stehen Zähler zur Verfügung.

Operandenbereich	Zugriff über Einheiten der folgenden Größe:	S7- Notation	Beschreibung
Datenbaustein	Datenbaustein, geöffnet mit "AUF DB":	DB	Datenbausteine speichern Informationen für das Programm. Sie können entweder so definiert sein, dass alle Codebausteine auf sie zugreifen können (globale DBs), oder sie sind einem bestimmten FB oder SFB zugeordnet (Instanz-DB).
	Datenbit	DBX	
	Datenbyte	DBB	
	Datenwort	DBW	
	Datendoppelwort	DBD	
	Datenbaustein, geöffnet mit "AUF DI":	DI	
	Datenbit	DIX	
	Datenbyte	DIB	
	Datenwort	DIW	
	Datendoppelwort	DID	
Lokaldaten	Lokaldatenbit	L	Dieser Speicherbereich nimmt die temporären Daten eines Bausteins für die Dauer der Bearbeitung dieses Bausteins auf. Der L-Stack stellt auch Speicher zum Übertragen von Bausteinparametern und zum Speichern von Zwischenergebnissen aus KOP-Netzwerken zur Verfügung.
	Lokaldatenbyte	LB	
	Lokaldatenwort	LW	
	Lokaldatendoppelwort	LD	
Peripheriebereich: Eingänge	Peripherieeingangsbyte	PEB	Die Peripheriebereiche der Ein- und Ausgänge erlauben direkten Zugriff auf zentrale und dezentrale Ein- und Ausgabebaugruppen.
	Peripherieeingangswort	PEW	
	Peripherieeingangsdoppelwort	PED	
Peripheriebereich: Ausgänge	Peripherieausgangsbyte	PAB	
	Peripherieausgangswort	PAW	
	Peripherieausgangsdoppelwort	PAD	

Prozessabbild der Ein- und Ausgänge

Grundlagen zu den Prozessabbildern

Funktion

Werden im Anwenderprogramm die Operandenbereiche Eingänge (E) und Ausgänge (A) angesprochen, dann werden nicht die Signalzustände auf den digitalen Signalbaugruppen abgefragt oder geändert, sondern ein Speicherbereich im Systemspeicher der CPU. Dieser Speicherbereich wird als Prozessabbild bezeichnet.

Vorteile des Prozessabbilds

Der Zugriff auf das Prozessabbild hat gegenüber dem direkten Zugriff auf die Ein- und Ausgabebaugruppen den Vorteil, dass der CPU für die Dauer der zyklischen Programmbearbeitung ein konsistentes Abbild der Prozesssignale zur Verfügung steht. Wenn sich während der Programmbearbeitung ein Signalzustand auf einer Eingabebaugruppe ändert, bleibt der Signalzustand im Prozessabbild bis zur Prozessabbildaktualisierung im nächsten Zyklus erhalten. Dadurch können Sie ein Eingangssignal in Ihrem Programm mehrmals abfragen und erhalten immer eine konsistente Eingangsinformation.

Außerdem benötigt der Zugriff auf das Prozessabbild wesentlich weniger Zeit als der direkte Zugriff auf die Signalbaugruppen, weil sich das Prozessabbild im internen Speicher der CPU befindet.

Aktualisieren der Prozessabbilder

Ablauf

Die Aktualisierung der Prozessabbilder der Ein- und Ausgänge (OB 1-Prozessabbild) erfolgt in dieser Reihenfolge:

1. Die internen Aufgaben des Betriebssystems werden ausgeführt.
2. Das Prozessabbild der Ausgänge (PAA) wird auf die Ausgänge der Baugruppen geschrieben.
3. Der Zustand der Eingänge wird ins Prozessabbild der Eingänge (PAE) gelesen.
4. Das Anwenderprogramm mit allen darin aufgerufenen Bausteinen wird abgearbeitet.

Das Betriebssystem steuert selbstständig das Schreiben des Prozessabbilds der Ausgänge auf die Ausgänge der Baugruppen und das Einlesen des Prozessabbilds der Eingänge.

Peripheriezugriffsfehler (PZF) bei Prozessabbild-Aktualisierung

S7-300-CPUs reagieren bei einem Fehler während der Prozessabbild-Aktualisierung folgendermaßen:

- Kein Diagnosepuffer-Eintrag, kein OB-Aufruf. Die entsprechenden Eingangsbytes werden auf "0" gesetzt und bleiben auf "0", bis der Fehler behoben ist.

S7-400-CPUs reagieren bei einem Fehler während der Prozessabbild-Aktualisierung folgendermaßen:

- Diagnosepuffer-Eintrag und Start des OB 85 bei jedem Zugriff auf die Peripherie bei jeder Aktualisierung des entsprechenden Prozessabbilds. Die fehlerhaften Eingangsbytes werden bei jedem Zugriff auf das Prozessabbild auf "0" gesetzt.

Sie können die Reaktion bei Peripheriezugriffsfehlern auf eine der folgenden Alternativen umparametrieren:

- Die CPU erzeugt einen Diagnosepuffer-Eintrag und startet den OB 85 bei jedem einzelnen Zugriff.
- Die CPU erzeugt einen Diagnosepuffer-Eintrag und startet den OB 85 nur bei kommendem und gehendem Peripheriezugriffsfehler. Vor Aufruf des OB 85 werden die fehlerhaften Eingangsbytes auf "0" gesetzt und danach bis zum gehenden Peripheriezugriffsfehler nicht mehr vom Betriebssystem überschrieben.
- Die CPU zeigt das voreingestellte Verhalten der S7-300. Es erfolgt kein OB-85-Aufruf, die entsprechenden Eingangsbytes werden auf "0" gesetzt und vom Betriebssystem nicht mehr überschrieben, bis der Fehler behoben ist.

Häufigkeit des OB 85-Starts

Neben der parametrierten Reaktion auf Peripheriezugriffsfehler beeinflusst auch der Adressraum einer Baugruppe die Häufigkeit des OB-85-Starts:

- Bei einer Baugruppe mit einem Adressraum bis zu einem Doppelwort startet der OB 85 einmal, z. B. bei einer Digitalbaugruppe mit bis zu 32 Eingängen oder Ausgängen, oder einer Analogbaugruppe mit 2 Kanälen.
- Bei Baugruppen mit größerem Adressraum startet der OB 85 so oft, wie mit Doppelwortbefehlen darauf zugegriffen wird, z. B. bei einer 4-kanaligen Analogbaugruppe zweimal.

Lokaldaten-Stack

Funktion

Beim Erstellen von Organisationsbausteinen können Sie temporäre Variablen (TEMP) deklarieren, die nur während der Bearbeitung des Bausteins zur Verfügung stehen und dann wieder überschrieben werden. Außerdem benötigt jeder Organisationsbaustein für seine Startinformation 20 Lokaldaten-Byte.

Die CPU besitzt einen begrenzten Speicher für die temporären Variablen der Lokaldaten gerade bearbeiteter Bausteine. Die Größe dieses Speicherbereichs, des so genannten Lokaldaten-Stacks oder L-Stacks, ist CPU-abhängig. Der Lokaldaten-Stack wird als Voreinstellung zu gleichen Teilen unter den Prioritätsklassen aufgeteilt. Das bedeutet, jede Prioritätsklasse verfügt über einen eigenen Bereich im Lokaldaten-Stack. Damit ist gewährleistet, dass auch hochpriore Prioritätsklassen und ihre zugeordneten OBs Platz für ihre Lokaldaten zur Verfügung haben.

Vor dem ersten Zugriff müssen die Lokaldaten initialisiert werden.

Der Lokaldaten-Stack speichert folgende Daten:

- Temporäre Variablen der Lokaldaten von Bausteinen
- Startinformationen der Organisationsbausteine
- Informationen zum Übergeben von Parametern
- Zwischenergebnisse der Logik in Kontaktplan-Programmen

Das folgende Bild zeigt die Zuordnung von Lokaldaten zu den Prioritätsklassen:

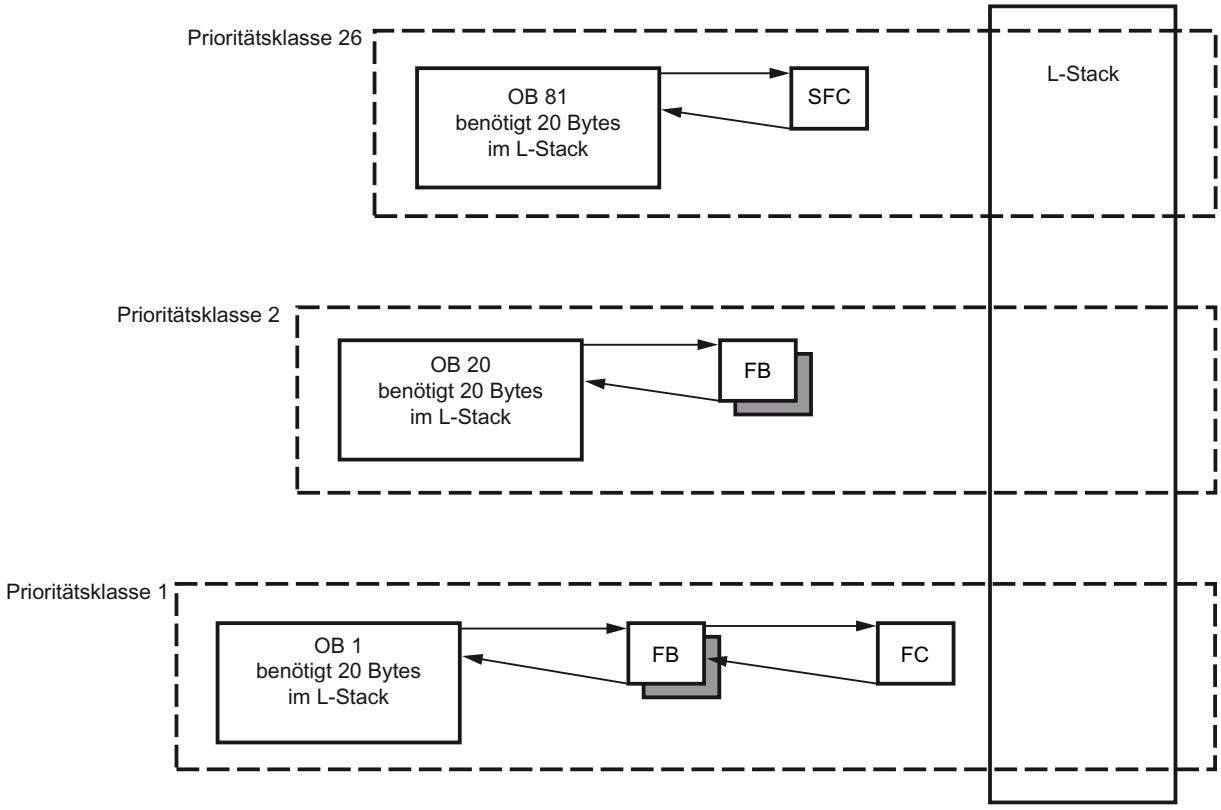

Im Lokaldaten-Stack wird der OB 1 zuerst vom OB 20 unterbrochen, der dann wiederum vom OB 81 unterbrochen wird.

⚠ VORSICHT

Alle temporären Variablen eines OBs und seiner unterlagerten Bausteine werden im Lokaldaten-Stack gespeichert. Wenn Sie viele Schachtelungsebenen in Ihrer Bausteinbearbeitung verwenden, besteht daher die Gefahr, dass der Lokaldaten-Stack überläuft.

Wenn Sie die zulässige Größe des Lokaldaten-Stacks für ein Programm überschreiten, wechseln S7-CPUs in den Betriebszustand "STOP".

Eine Übersicht über die benötigten Lokaldaten erhalten Sie mit der Funktion "Programminformation anzeigen".

Berücksichtigen Sie auch den Lokaldatenbedarf von Synchronfehler-OBs.

Zuordnen von Lokaldaten zu Prioritätsklassen

Bei S7-300: Die maximale Anzahl der Lokaldaten pro Priotitätsklasse und pro Baustein ist bei S7-300 beschränkt (siehe Technische Daten der CPU).

Bei S7-400: Die maximale Anzahl der Lokaldaten insgesamt ist beschränkt (siehe Technische Daten der CPU). Die verfügbaren Lokaldaten verteilen Sie im Rahmen der CPU-Parametrierung frei auf die verschiedenen Prioritätsklassen. Die maximale Anzahl Lokaldaten darf dabei nicht überschritten werden.

Unterbrechungs-Stack

Funktion

Wenn die Programmbearbeitung durch einen Organisationsbaustein mit höherer Priorität unterbrochen wird, speichert das Betriebssystem folgende Daten im Unterbrechungs-Stack (U-Stack):

- Die aktuellen Inhalte der Akkumulatoren und Adressregister
- Die Nummer und die Größe der geöffneten Datenbausteine

Nach der Bearbeitung des neuen Organisationsbausteins lädt das Betriebssystem die Informationen aus dem Unterbrechungs-Stack. Die Bearbeitung des unterbrochenen Bausteins wird anschließend an der Stelle wieder aufgenommen, an der die Unterbrechung eintrat.

Baustein-Stack

Funktion

Wenn die Bearbeitung eines Bausteins durch den Aufruf eines anderen Bausteins oder durch eine höhere Prioritätsklasse unterbrochen wird, z. B. Alarm oder Fehlerbehandlung, speichert das Betriebssystem folgende Daten im Baustein-Stack (B-Stack):

- Nummer, Art (OB, FB, FC, SFB, SFC) und Absprungadresse des Bausteins, der unterbrochen wurde.
- Nummer der Datenbausteine aus DB- und DI-Register, die zum Zeitpunkt der Unterbrechung geöffnet waren.

Mit den gespeicherten Daten kann das Anwenderprogramm nach der Unterbrechung wieder fortgesetzt werden.

Im Betriebszustand "STOP" können Sie am PG den Baustein-Stack bei geöffnetem Programmiereditor mit Hilfe der Testfunktion "Aufrufhierarchie" auslesen. Der Baustein-Stack führt alle Bausteine auf, deren Bearbeitung zu dem Zeitpunkt nicht beendet war, als die CPU in den Betriebszustand "STOP" versetzt wurde.

Das folgende Bild zeigt die Aufrufreihenfolge der Bausteine:

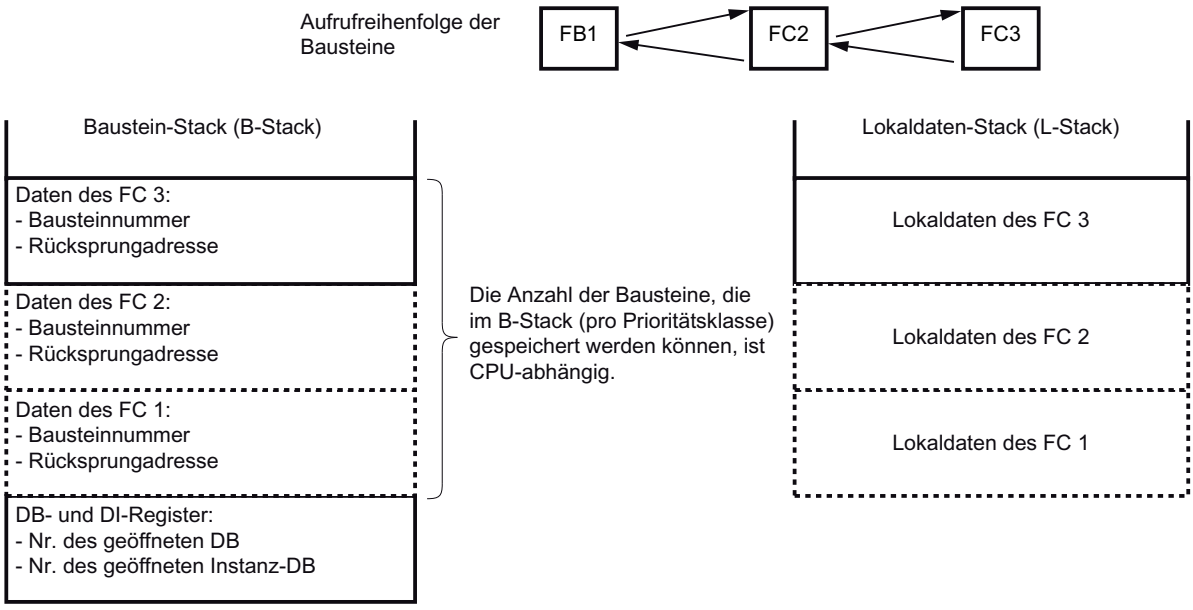

Datenbausteinregister

Es existieren zwei Datenbausteinregister. Sie enthalten die Nummern der aufgeschlagenen Datenbausteine:

- Im DB-Register steht die Nummer des aufgeschlagenen globalen Datenbausteins.
- Im DI-Register steht die Nummer des aufgeschlagenen Instanz-Datenbausteins.

Diagnosepuffer

Funktion

Im Diagnosepuffer werden die System- und anwenderdefinierten Diagnoseereignisse in der Reihenfolge ihres Auftretens eingetragen. Der erste Eintrag enthält das neueste Ereignis. Die Anzahl der Einträge im Diagnosepuffer ist abhängig von der jeweiligen Baugruppe und von ihrem aktuellen Betriebszustand.

Mögliche Diagnoseereignisse:

- Fehler auf einer Baugruppe
- Fehler in der Prozessverdrahtung
- Systemfehler in der CPU
- Betriebszustandsübergänge der CPU
- Fehler im Anwenderprogramm
- Über die Systemfunktion SFC 52 vom Anwender definierte Diagnoseereignisse

Die Information, die beim Auftreten eines System-Diagnoseereignisses in den Diagnosepuffer eingetragen wird, ist identisch mit der Startinformation, die an den entsprechenden Organisationsbaustein übergeben wird.

Die Einträge im Diagnosepuffer können nicht gelöscht werden, der Inhalt des Diagnosepuffers bleibt auch nach dem Urlöschen erhalten.

Vorteile des Diagnosepuffers

Der Diagnosepuffer bietet folgende Vorteile:

- Bei einem Anlagenstopp können Sie die letzten Ereignisse vor dem "STOP" auswerten und so die STOP-Ursache finden.
- Sie können Fehlerursachen schneller erkennen und dadurch die Verfügbarkeit der Anlage erhöhen.
- Sie können das dynamische Anlagenverhalten auswerten und anschließend optimieren.

Organisation des Diagnosepuffers

Der Diagnosepuffer ist als Ringpuffer ausgelegt. Die maximale Anzahl der Einträge ist von der Baugruppe abhängig, für die S7-300 CPU 314 z. B. 100 Einträge. Wenn diese maximale Anzahl von Einträgen erreicht wird, wird bei einem neuen Diagnosepufferereignis der älteste Eintrag gelöscht. Alle Einträge rücken entsprechend weiter. Dadurch steht der jüngste Diagnoseeintrag immer an erster Stelle.

Die Anzahl der angezeigten Einträge im Diagnosepuffer ist abhängig von der Baugruppe und ihrem aktuellen Betriebszustand. Bei bestimmten CPUs ist die Länge des Diagnosepuffers parametrierbar.

Auswertung des Diagnosepuffers

Sie haben folgende Möglichkeiten, auf Inhalte des Diagnosepuffers zuzugreifen:

- Online über die Diagnosesicht
- Im Anwenderprogramm über SFC 51 "RDSYSST".

Die CPU ist so voreingestellt, dass sie den letzten Diagnosepuffereintrag vor dem Übergang von "RUN" nach "STOP" automatisch an ein angemeldetes Beobachtungsgerät (PG, OP oder TD) sendet. Dadurch kann die Ursache für den Betriebszustandswechsel schnell gefunden und behoben werden. Sie können diese Voreinstellung ändern.

Hinweis

Um die Zeitangaben der Diagnosepuffer-Einträge sinnvoll nutzen zu können, stellen Sie die Uhrzeit und das Datum auf der Baugruppe ein und überprüfen Sie die Einstellungen gelegentlich.

Zeiten und Zähler

Definition

Zeiten und Zähler sind Speicherbereiche des Systemspeichers. Die Funktion einer Zeit oder Zählers wird durch das Anwenderprogramm festgelegt, z. B. die Einschaltverzögerung oder der Zähleranfangswert. Die Anzahl der verfügbaren Zeiten und Zähler ist CPU-abhängig.

Hinweis

Beachten Sie folgende Hinweise zu den Zeiten:

- Wenn Sie in Ihrem Anwenderprogramm mehr Zeiten und Zähler einsetzen, als die CPU zulässt, wird ein synchroner Fehler gemeldet und der Organisationsbaustein "OB 121" gestartet.

Zugriff auf den Peripheriedatenbereich

Peripheriedatenbereich

Aufbau des Peripheriedatenbereichs

Peripheriedatenbereich

Der Peripheriedatenbereich lässt sich unterteilen in:

- Nutzdaten
- Diagnose- und Parameterdaten

Beide Bereiche haben einen Eingangsbereich mit nur lesendem Zugriff und einen Ausgangsbereich mit nur schreibendem Zugriff.

Nutzdaten

Definition

Nutzdaten können sein:

- Digitale und analoge Ein- und Ausgangssignale von Signalbaugruppen
- Steuer- und Statusinformationen von Funktionsbaugruppen
- Informationen für Punkt-zu-Punkt und Buskopplungen von Kommunikationsbaugruppen

Zugriff auf die Nutzdaten

Adressiert werden die Nutzdaten bei digitalen Signalbaugruppen über die Byteadresse des Ein- bzw. Ausgangsbereichs.

Bei analogen Signalbaugruppen erfolgt die Adressierung über die Wortadresse des Ein- bzw. Ausgangsbereichs.

Auf Nutzdaten können Sie über Lade- und Transferbefehle oder Kommunikationsfunktionen (Bedienen und Beobachten, BuB) bzw. über den Prozessabbildtransfer zugreifen.

Bei Lade- und Transferbefehlen kann eine Datenkonsistenz von 1, 2 und 4 Bytes erreicht werden. Daten aus dem Prozessabbild sind immer konsistent.

Wenn Sie eine Datenkonsistenz von 3 oder mehr als 4 Bytes erreichen möchten, müssen Sie die Anweisung "DPRD_DAT" (SFC 14) bzw. "DPWR_DAT" (SFC 15) einsetzen.

Um 4 Byte zusammenhängend und unverändert zu übertragen, verwenden Sie die Anweisung "Transferiere Doppelwort". Wenn Sie vier einzelne Anweisungen "Transferiere Eingangsbyte" verwenden, kann die Konsistenz der Bytes nicht erhalten werden und es kann zu unerwünschten Effekten kommen.

Diagnose- und Parameterdaten

Datensätze der Diagnose- und Parameterdaten

Die Diagnose- und Parameterdaten einer Baugruppe können nicht einzeln adressiert werden, sondern nur zusammengefasst als ganze Datensätze. Diagnose- und Parameterdaten werden grundsätzlich konsistent übertragen.

Adressiert werden die Diagnose- und Parameterdaten über die Anfangsadresse der betreffenden Baugruppe und die Datensatznummer. Datensätze werden unterteilt in Eingangs- und Ausgangsdatensätze. Eingangsdatensätze können nur gelesen, Ausgangsdatensätze können nur geschrieben werden. Auf Datensätze können Sie mithilfe von Systemfunktionen oder Kommunikationsfunktionen zugreifen.

Die folgende Tabelle zeigt die Zuordnung der Datensätze zu Diagnose- und Parameterdaten:

Daten	Beschreibung
Diagnosedaten	Bei diagnosefähigen Baugruppen erhalten Sie beim Lesen der Datensätze 0 und 1 die Diagnosedaten dieser Baugruppe.
Parameterdaten	Bei parametrierbaren Baugruppen übertragen Sie beim Schreiben der Datensätze die Parameter dieser Baugruppe.

Zugreifen auf die Datensätze einer Baugruppe

Anweisungen für den Zugriff auf die Datensätze

Um parametrierbare Baugruppen nachzuparametrieren und Diagnoseinformationen diagnosefähiger Baugruppen auszulesen, können Sie die Informationen in den Datensätzen einer Baugruppe nutzen.

Die folgende Tabelle zeigt, mit welchen Anweisungen Sie auf Datensätze zugreifen können:

Anweisung	Anwendung
Baugruppen parametrieren	
WR_PARM	Übertragen der änderbaren Parameter (Datensatz 1) zur adressierten Signalbaugruppe
WR_DPARM	Übertragen einzelner Parameterdatensätze aus den SDBs 100 bis 129 zur adressierten Signalbaugruppe
PARM_MOD	Übertragen aller Parameter aus den SDBs 100 bis 129 zur adressierten Signalbaugruppe
WR_REC	Übertragen eines beliebigen Datensatzes zur adressierten Signalbaugruppe
Diagnoseinformation auslesen	
RD_REC	Auslesen der Diagnosedaten

Hinweis

Wenn ein DPV1-Slave über eine GSD-Datei (GSD ab Rev. 3) projektiert ist, und die DP-Schnittstelle des DP-Masters (S7-400-CPU) auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit der Anweisung "WR_REC/RD_REC" bzw. den Anweisungen "WRREC/RDREC" von den E/A-Baugruppen gelesen bzw. beschrieben werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz, d. h. den projektierten Steckplatz +3. Um das Problem zu umgehen, stellen Sie die Schnittstelle des DP-Masters auf "DPV1" um.

Baugruppen-Anfangsadresse

Die Baugruppen-Anfangsadresse ist die niedrigste Byteadresse einer Baugruppe. Sie stellt die Anfangsadresse des Nutzdatenbereichs der Baugruppe dar und wird in vielen Fällen stellvertretend für die ganze Baugruppe verwendet.

Die Baugruppen-Anfangsadresse wird z. B. bei Prozessalarmen, Diagnosealarmen, Ziehen/Stecken-Alarmen und Stromversorgungsfehlern in die Startinformation des zugehörigen Organisationsbausteins eingetragen und identifiziert damit die alarmgebende Baugruppe.

Baugruppen-Anfangsadressen konfigurieren

Die Zuordnung zwischen Adressen, die im Anwenderprogramm verwendet werden, und den Baugruppen erfolgt durch die Konfiguration der Baugruppen.

Die beim Platzieren der Baugruppen im Baugruppenträger automatisch vergebenen Anfangsadressen können Sie in den Eigenschaften der Baugruppe ändern (Gruppe "Adressen").

Nutzdatenbereich

Zugriff auf die Peripherieadressen

Nutzdaten im Operandenbereich Peripherie

Die Nutzdaten stellen den Operandenbereich Peripherie dar. Es gibt folgende Bereiche:

- Peripherie-Eingänge PE
- Peripherie-Ausgänge PA

Wenn die Nutzdaten im Prozessabbild liegen, dann übernimmt die CPU automatisch den Datenaustausch zwischen Baugruppe und Prozessabbild-Bereich beim Aktualisieren der Prozessabbilder.

Wenn die Nutzdaten nicht im Prozessabbild liegen, dann müssen Sie im Anwenderprogramm über direkten Peripheriezugriff auf diese Daten zugreifen (z. B. L %IB500:P).

Die Adressen von Analogein- und Analogausgabebaugruppen liegen normalerweise nicht im Prozessabbild-Bereich. Sie haben aber die Möglichkeit, die Adressen von Analogbaugruppen in diesen Bereich zu legen.

Siehe auch

Prozessabbilder (Speicherbereich der Ein- und Ausgänge) (Seite 863)

Direkter Zugriff auf die Peripherie (Speicherbereich PE und PA) (Seite 864)

Adressbereiche

Peripherie-Adressbereiche

Für Ein- und Ausgänge existiert jeweils ein eigener Adressbereich. Deshalb muss die Adresse eines Peripheriebereichs außer der Byte- oder Wortangabe zusätzlich die Kennung "E" für Eingänge und die Kennung "A" für Ausgänge beinhalten.

Die folgende Tabelle zeigt die zur Verfügung stehenden Peripherie-Adressbereiche:

Operandenbereich	Zugriff über Einheiten der folgenden Größe:	S7-Notation
Peripheriebereich: Eingänge	Peripherieeingangsbyte Peripherieeingangswort Peripherieeingangsdoppelwort	PEB PEW PED
Peripheriebereich: Ausgänge	Peripherieausgangsbyte Peripherieausgangswort Peripherieausgangsdoppelwort	PAB PAW PAD

Siehe auch

Direkter Zugriff auf die Peripherie (Speicherbereich PE und PA) (Seite 864)

Einstellen des Betriebsverhaltens

Einführung in das Einstellen des Betriebsverhaltens

Eigenschaften von Automatisierungssystemen beeinflussen

Eigenschaften von Automatisierungssystemen, die nicht vorgegeben sind, können Sie folgendermaßen beeinflussen:

- Durch das Einstellen von Systemparametern
- Durch den Einsatz von Systemfunktionen (SFCs) in erweiterten Anweisungen

Ändern von Baugruppenparametern

Standardeinstellungen

Alle parametrierbaren Baugruppen sind bei Lieferung auf Werte eingestellt, die für Standardanwendungen geeignet sind. Mit diesen Standardwerten können Sie die Baugruppen ohne weitere Einstellungen einsetzen.

Sie können jedoch das Verhalten und die Eigenschaften der Baugruppen parametrieren und damit auf Ihre Erfordernisse und die Gegebenheiten Ihrer Anlage einstellen. Parametrierbare Baugruppen sind CPUs, FMs, CPs sowie einige Analogein- und Analogausgabebaugruppen und Digitalein- und Digitalausgabebaugruppen.

Baugruppenparameter

Die Baugruppenparameter werden zur besseren Übersicht gruppiert, für CPUs z. B. in folgende Gruppen:

- Anlauf
- Zyklus
- PROFINET-Schnittstelle bzw andere Schnittstellen
- Diagnose
- Remanenz
- Taktmerker
- Alarme
- Schutz
- Uhr
- Verbindungsressourcen

Einstellen und Laden der Parameter

Sie können die Baugruppenparameter einstellen. Beim Speichern einer Gerätekonfiguration mit ihren Parametern werden Systemdaten erzeugt, die mit dem Anwenderprogramm in die CPU geladen und von dort beim Anlauf in die zugehörigen Baugruppen übertragen wird.

Parametrieren mit Anweisungen

Neben der Parametrierung besteht die Möglichkeit, Baugruppenparameter mithilfe von erweiterten Anweisungen vom Programm aus zu ändern.

Die folgende Tabelle zeigt, welche Anweisungen welche Baugruppenparameter ändern können:

Systemfunktion	Anwendung
WR_PARM	Übertragen der änderbaren Parameter (Datensatz 1) zur adressierten Signalbaugruppe.
WR_DPARM	Übertragen einzelner Parameterdatensätze aus den zugehörigen SDBs zur adressierten Signalbaugruppe.
PARM_MOD	Übertragen aller Parameter aus den zugehörigen SDBs zur adressierten Signalbaugruppe.
WR_REC	Übertragen eines beliebigen Datensatzes zur adressierten Signalbaugruppe

Allgemeine CPU-Parameter

Allgemeine Eigenschaften von Baugruppen

Die Parametergruppe "Allgemein" informiert über die Art und den Ort der Baugruppe sowie bei Baugruppen mit Schnittstellen über netzrelevante Informationen, z. B. die Teilnehmeradresse.

Kurzbezeichnung, Artikel-Nr./Firmware, Name, Anlagenkennzeichen, Ortskennzeichen

Die Kurzbezeichnung, die Info darunter und Artikel-Nr./Firmware sind identisch zu den Angaben im Hardware-Katalog.

Im Feld "Name" steht der Name der Baugruppe, den Sie nach Ihren Vorgaben ändern können. Wenn Sie den Namen ändern, erscheint der geänderte Name auch in der Gerätesicht bzw. in der Netzsicht.

CPUs bieten die Möglichkeit, ein spezifisches Anlagenkennzeichen festzulegen. Mit dem Anlagenkennzeichen werden Teile der Anlage eindeutig nach funktionalen Gesichtspunkten gekennzeichnet. Es ist (gemäß IEC 1346-1) hierarchisch aufgebaut. Ebenfalls möglich ist die Eingabe eines bis zu 32 Zeichen langen Ortskennzeichens. Es ist Teil des Betriebsmittelkennzeichens, das die genaue Lage z. B. einer Messstelle bzw. Messstellengruppe innerhalb einer prozesstechnischen Anlage kennzeichnet (ähnlich einer Straße in einem Stadtplan).

Das im Feld "Anlagenkennzeichen" bzw. "Ortskennzeichen" eingegebene Kennzeichen kann im Anwenderprogramm der CPU ausgewertet werden. Nutzen Sie die Eingabemöglichkeit,

wenn Sie das Anwenderprogramm abhängig vom Anlagenkennzeichen bzw. Ortskennzeichen abarbeiten wollen.

Kommentar

Jede Baugruppe verfügt über ein Kommentarfeld. In diesem Feld können Sie z. B. den Einsatzzweck der Baugruppe eingeben.

Anlauf

Anlaufparameter

Das Anlaufverhalten der CPU

Im Folgenden sind die Parameter von S7-CPUs beschrieben, die das Anlaufverhalten bestimmen.

Anlauf bei Sollausbau ungleich Istausbau

Der Sollausbau wird repräsentiert durch die in die CPU geladene projektierte Konfiguration (Systemdatenbausteine). Der Istausbau ist der tatsächliche Ausbau des Automatisierungssystems.

Wenn das Optionskästchen "Anlauf bei Sollausbau ungleich Istausbau" deaktiviert ist, geht die CPU in folgenden Fällen in "STOP":

- Wenn mindestens eine Baugruppe nicht auf dem projektierten Steckplatz steckt.
- Wenn auf dem projektieren Steckplatz eine Baugruppe von einem anderen Typ steckt, z. B. bei der Inbetriebnahme.
- Wenn dezentrale Peripherie fehlt.

Wenn über den projektierten Ausbau hinaus noch weitere Baugruppen (zentral oder dezentral) vorhanden sind, werden sie nicht für den Soll-Ist-Vergleich herangezogen.

Wenn das Optionskästchen "Anlauf bei Sollausbau ungleich Istausbau" aktiviert ist, dann läuft die CPU auch dann in den oben genannten Fällen an.

Der Peripherieausbau (zentrale und dezentrale Baugruppen, zentrale Baugruppenträger und DP-Slaves) wird von der CPU nicht überprüft. Eine Ausnahme bilden PROFIBUS-DP-Anschaltungen. Projektierte Baugruppen dieses Typs müssen wegen ihrer engen Anbindung an die CPU in jedem Fall gesteckt sein, damit die CPU anlaufen kann.

In der Voreinstellung der CPU ist "Anlauf bei Sollausbau ungleich Istausbau" aktiviert, so dass der Anlauf trotz Unterschied zwischen Soll- und Istausbau erfolgt.

Überwachungszeit für Fertigmeldung der Baugruppen

Maximale Dauer für die Fertigmeldung aller konfigurierten Baugruppen nach NETZ-EIN. Wenn nach Ablauf dieser Überwachungszeit die Baugruppen keine Fertigmeldung zur CPU senden, ist der Istausbau ungleich dem Sollausbau. Die Reaktion der CPU wird daher bestimmt vom Parameter "Anlauf bei Sollausbau ungleich Istausbau".

Innerhalb dieser Überwachungszeit können z. B. bei mehrzeiligem Aufbau oder bei Geräten, die über die DP-Master-Schnittstelle an die CPU angeschlossen sind, alle Stromversorgungen eingeschaltet werden, ohne dass eine Einschalt-Reihenfolge beachtet werden muss.

Überwachungszeit für Parameterübertragung zu Baugruppen

Maximale Dauer für die "Verteilung" der Parameter an die parametrierbaren Baugruppen (Zeit beginnt nach "Fertigmeldung der Baugruppen". Bei CPUs mit DP-Master-Schnittstelle stellen Sie mit diesem Parameter auch die Hochlaufzeitüberwachung der DP-Slaves ein. Das bedeutet, dass in der eingestellten Zeit die DP-Slaves hochlaufen und von der CPU (als DP-Master) parametriert sein müssen.

Wenn nach Ablauf dieser Überwachungszeit nicht alle Baugruppen/DP-Slaves parametriert sind, ist der Istausbau ungleich dem Sollausbau. Die Reaktion der CPU wird daher bestimmt vom Parameter "Anlauf bei Sollausbau ungleich Istausbau".

Anlaufparameter nur für S7-400-CPUs

Ausgänge zurücksetzen bei Wiederanlauf

Ausgänge zurücksetzen bei Wiederanlauf

Die CPUs der S7-400 können bei einem Wiederanlauf die Ausgänge (und damit das Prozessabbild der Ausgänge PAA) zurücksetzen.

Siehe auch

Wiederanlauf (Seite 758)

Wiederanlauf sperren bei Anlauf durch Bedienung

Wiederanlauf sperren bei Anlauf durch Bedienung (z. B. PG) oder Kommunikationsauftrag (z. B. von MPI-Teilnehmern)

Bei S7-400 haben Sie die Möglichkeit, die Anlaufart "Wiederanlauf" zu sperren, wenn sie durch PG-Bedienung oder per Kommunikationsauftrag ausgelöst werden sollte. Damit ist der Wiederanlauf nur "vor Ort" durchführbar.

- Aktivieren Sie dieses Optionskästchen, wenn die durch Bedienung oder Kommunikationsauftrag ausgelöste Anlaufart eingeschränkt werden soll auf den Neustart (=manueller Neustart) oder Kaltstart.
- Wenn das Optionskästchen nicht aktiviert ist, dann sind alle Anlaufarten möglich.

Ein Wiederanlauf bei Bedienung oder Kommunikationsauftrag (=manueller Wiederanlauf) wird ausgelöst, wenn folgende Voraussetzungen erfüllt sind:

- Optionskästchen "Wiederanlauf sperren ..." wurde nicht aktiviert und diese Parametrierung in die CPU übertragen
- Es liegen keine Hindernisse für einen Wiederanlauf vor (Batterie o.k., keine Änderung des Anwenderprogramms im STOP-Zustand, ...)
- Am PG wird der Betriebszustand Wiederanlauf gewählt oder ein anderer Teilnehmer verursacht über Kommunikationsauftrag einen Wiederanlauf.

Siehe auch

Wiederanlauf (Seite 758)

Zyklusverhalten

Einstellungen zum Zyklusverhalten

OB1-Prozessabbild zyklisch aktualisieren

Wenn die Option aktiviert ist, wird das OB1-Prozessabbild zyklisch aktualisiert (Voreinstellung). Bei S7-400-CPUs kann die OB1-Aktualisierung mit Hilfe dieses Parameters abgeschaltet werden.

Zyklusüberwachungszeit

Die Zyklusüberwachungszeit wird in ms eingestellt.

Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, dann geht die CPU in den Betriebszustand "STOP". Sie haben die Möglichkeit den "STOP" zu verhindern, in dem Sie den Organisationsbaustein OB 80 programmieren.

Mögliche Ursachen für eine Überschreitung sind:

- Kommunikationsprozesse
- Häufung von Alarmereignissen
- Fehler im CPU-Programm.

Steuerung von zyklusbelastenden Kommunikationsprozessen

Mit dem Parameter "Zyklusbelastung durch Kommunikation" können Sie die Dauer von Kommunikationsprozessen, die immer auch die Zykluszeit verlängern, in einem gewissen Rahmen steuern. Kommunikationsprozesse können z. B. sein:

- Datenübertragung zu einer anderen CPU
- Laden von Bausteinen (über PG angestoßen)

Testfunktionen mit dem PG werden durch diesen Parameter kaum beeinflusst, sie können die Zykluszeit erheblich verlängern. Die für Testfunktionen zur Verfügung gestellte Zeit kann im Prozessbetrieb begrenzt werden.

Wirkungsweise des Parameters

Das Betriebssystem der CPU stellt laufend der Kommunikation den projektierten Prozentsatz der gesamten CPU-Verarbeitungsleistung zur Verfügung (Zeitscheiben-Technik). Wird diese Verarbeitungsleistung für die Kommunikation nicht benötigt, steht sie der übrigen Verarbeitung zur Verfügung.

Auswirkung auf die tatsächliche Zykluszeit

Ohne zusätzliche asynchrone Ereignisse verlängert sich die OB 1-Zykluszeit um einen Faktor, der sich nach folgender Formel berechnen lässt:

$$\frac{100}{100 - \text{“Zyklusbelastung durch Kommunikation“}}$$

Einstellen der Größe des Prozessabbildes

Bei CPUs mit variabler Größe des Prozessabbildes ist die Größe des Prozessabbildes (in Bytes) einstellbar. Der Bereich des Prozessabbildes beginnt immer bei Eingangs- bzw. Ausgangs-Byte 0.

Bei S7-300-CPUs, die diese Funktion unterstützen, sind Besonderheiten für den Zugriff auf den Prozessabbild-Bereich bei der Programmerstellung zu berücksichtigen. Diese Besonderheiten sind der entsprechenden Dokumentation zur CPU zu entnehmen.

Beispiel: Die variable Einstellung des Prozessabbildes wirkt derzeit nur auf die Aktualisierung des Prozessabbildes am Zykluskontrollpunkt. Das PAE wird nur bis zur parametrierten PAE-Größe aktualisiert und nur der parametrierte PAA-Bereich an die Ausgabebaugruppen ausgegeben. Zugriffe des Anwenderprogramms auf den Bereich zwischen eingestellter Prozessabbildgröße und maximaler Prozessabbildgröße führen nicht zu einem synchronen Zugriffsfehler.

Siehe auch

Beispiele für die Zyklusbelastung durch Kommunikation (Seite 793)

Beispiele für die Zyklusbelastung durch Kommunikation

Im Folgenden finden Sie zwei Beispiele, wie sich die Kommunikation auf die Zyklusbelastung auswirken kann.

Beispiel für die Zyklusbelastung durch Kommunikation ohne zusätzliche asynchrone Ereignissen

Bei Einstellung der Zyklusbelastung durch Kommunikation auf 50 % kann sich eine Verdopplung der OB 1-Zykluszeit ergeben.

Gleichzeitig wird die OB 1-Zykluszeit auch noch durch asynchrone Ereignisse (z. B. Prozessalarme oder Weckalarme) beeinflusst. Durch die Verlängerung der Zykluszeit durch

den Kommunikationsanteil treten statistisch gesehen auch mehr asynchrone Ereignisse innerhalb eines OB 1-Zyklus auf. Dies verlängert der OB 1-Zyklus zusätzlich. Diese Verlängerung ist abhängig davon, wie viele Ereignisse pro OB 1-Zyklus auftreten und der Dauer der Ereignisbearbeitung.

Beispiel für die Zyklusbelastung durch Kommunikation mit zusätzlichen asynchronen Ereignissen

Bei einer reinen OB 1-Ablaufzeit von 500 ms kann sich durch eine Kommunikationslast von 50 % eine tatsächliche Zykluszeit von bis zu 1000 ms ergeben (unter der Voraussetzung, dass die CPU immer genügend Kommunikationsaufträge zum Bearbeiten hat). Wenn nun parallel dazu alle 100 ms ein Weckalarm mit 20 ms Bearbeitungszeit abläuft, dann würde sich dieser ohne Kommunikationslast mit insgesamt 5*20 ms = 100 ms auf den Zyklus verlängernd auswirken, d. h. die tatsächliche Zykluszeit wäre 600 ms. Da ein Weckalarm auch die Kommunikation unterbricht, wirkt er sich bei 50% Kommunikationslast mit 10 * 20 ms auf die Zykluszeit aus, d.h. in diesem Fall beträgt die tatsächliche Zykluszeit nicht 1000 ms sondern 1200 ms.

Hinweis

Überprüfen Sie die Auswirkungen einer Wertänderung des Parameters "Zyklusbelastung durch Kommunikation" im Anlagenbetrieb.

Die Kommunikationslast muss beim Einstellen der minimalen Zykluszeit berücksichtigt werden, da es sonst zu Zeitfehlern kommt.

OB 85-Aufruf bei Peripheriezugriffsfehler (PZF)

Einstellungen zum OB 85-Aufruf

Sie können die voreingestellte Reaktion der CPU bei Peripheriezugriffsfehlern (PZF) während der systemseitigen Aktualisierung des Prozessabbildes ändern.

Empfehlung: Wenn Sie den OB 85-Aufruf bei Peripheriezugriffsfehlern aktivieren wollen, dann wählen Sie die Option "nur bei kommenden und gehenden Fehlern", damit die Zykluszeit der CPU nicht durch wiederholtes Aufrufen des OB 85 ansteigt.

Die S7-300-CPUs sind so voreingestellt, dass sie bei Peripheriezugriffsfehlern keinen OB 85 aufrufen und auch keinen Eintrag in den Diagnosepuffer erzeugen.

Remanenzverhalten

Remanenzeinstellungen für Merker, Timer und Zähler

Bei der Parametrierung der Remanenz legen Sie für die CPU fest, welche Speicherbereiche nach einem Netzausfall oder nach einem Übergang von "STOP" nach "RUN" erhalten bleiben.

Anzahl Merkerbytes ab MB 0

Tragen Sie die Anzahl der remanenten Merkerbytes ab Merkerbyte 0 ein.

Beispiel

Wenn Sie "2" eingeben, dann sind die Merkerbytes MB 0 und MB 1 remanent.

Anzahl S7-Timer ab T0

Tragen Sie die Anzahl der remanenten S7-Timer (Zeiten (T)) ab T 0 ein.

Jeder S7-Timer belegt 2 Byte.

Beispiel

Wenn Sie "2" eingeben, dann sind die S7-Timer T 0 und T 1 remanent.

Anzahl S7-Zähler ab Z0

Tragen Sie die Anzahl der remanenten S7-Zähler (Zähler (Z)) ab Z 0 ein.

Beispiel

Wenn Sie "2" eingeben, dann sind die S7-Zähler Z 0 und Z 1 remanent.

Remanenzeinstellungen für Datenbausteine

Funktion

In den Remanenzeigenschaften können Sie festlegen, welche Speicherbereiche nach einem Netzausfall und nach einem Übergang von "STOP" nach "RUN" auch ohne Batteriepufferung im Backup-Speicher der CPU erhalten bleiben. Ob bzw. wie viele remanente Bereiche projektiert werden können, hängt von der CPU ab.

Wenn eine Pufferbatterie eingelegt ist, dann sind Datenbausteine immer remanent.

Remanenzbereiche (nur für S7-300 ohne Batteriepufferung bzw. ohne Micro Memory Card relevant)

Remanenzbereiche sind Datenbereiche von Datenbausteinen.

Wenn Datenbereiche im Arbeitsspeicher der CPU remanent sein sollen, geben Sie folgendes ein:

- DB-Nr.
  Nummern der Datenbausteine ein, die remanent sein sollen.
- Byteadresse (Anfangsadresse mit der Länge der Datenbytes)
  Byteadressen der Datenbausteine ein, die remanent sein sollen.
- Anzahl Bytes
  Anzahl der remanenten Datenbytes ab der Byteadresse. Wenn Sie z. B. "1" eingeben, dann ist das Datenbyte "0" remanent.

Einstellmöglichkeiten für Alarme

Einstellungen bei Prozessalarmen

Prozessalarme

Je höher die Priorität gewählt wird, desto höher prior wird der zugehörige Prozessalarm bearbeitet. Mit der Priorität "0" wählen Sie den Alarm-OB ab. Die voreingestellte Priorität ist bei S7-300-CPUs nicht änderbar.

Einem Prozessalarm können bei CPUs, die Teilprozessabbilder unterstützen, Teilprozessabbilder zugeordnet werden. Beim Start des Prozessalarm-OBs wird dann das zugeordnete Teilprozessabbild aktualisiert, so dass der OB mit aktuellen Prozesswerten arbeitet. Bei der Einstellung "---" ist kein Prozessabbild zugeordnet.

Siehe auch

Prozessalarm-OBs (OB 40 bis OB 47) (Seite 878)

Einstellungen bei Verzögerungsalarmen

Verzögerungsalarme

Mit Verzögerungsalarm-OBs können Teile des Anwenderprogramms zeitverzögert bearbeitet werden.

Je höher die Priorität gewählt wird, desto höherprior wird der zugehörige Verzögerungsalarm bearbeitet. Mit der Priorität "0" wählen Sie den Alarm-OB ab. Die voreingestellte Priorität ist bei S7-300-CPUs nicht änderbar.

Einem Verzögerungsalarm können bei CPUs, die Teilprozessabbilder unterstützen, Teilprozessabbilder zugeordnet werden. Beim Start des Prozessalarm-OBs wird dann das zugeordnete Teilprozessabbild aktualisiert, so dass der OB mit aktuellen Prozesswerten arbeitet. Bei der Einstellung "---" ist kein Prozessabbild zugeordnet.

Die Verzögerungsalarme werden mit folgenden Anweisungen hantiert:

- SRT_DINT Verzögerungsalarm starten
- CAN_DINT Verzögerungsalarm stornieren
- QRY_DINT Zustand eines Verzögerungsalarms abfragen

Siehe auch

Verzögerungsalarm-OBs (OB 20 bis OB 23) (Seite 874)

Einstellungen bei Asynchronfehleralarmen

Asynchronfehleralarme

Die voreingestellte Priorität ist bei S7-300-CPUs nicht änderbar.

Die Änderung der Priorität von Asynchronfehler-OBs bietet sich an, damit die CPU auch unter folgenden Umständen im Betriebszustand "RUN" bleibt:

- Es treten gehäuft Asynchronfehler auf, die denselben Alarm-OB auslösen (z. B. Diagnosealarm, Aufruf OB 82).
- Es tritt zusätzlich ein anderes Asynchronfehler-Ereignis auf, das einen anderen Alarm-OB auslöst (z. B. Programmablauffehler, Aufruf OB 85).
- Es soll nur eine Meldung abgesetzt werden (z. B. bei Stationsausfall, welche Station ausgefallen ist). Eine Prozessreaktion ist nicht erforderlich. In diesem Fall kann dem OB 86 eine niedrige Priorität zugewiesen werden, um die Reaktionszeit auf andere Alarme nicht zu erhöhen.

Beispiel für die Notwendigkeit der Prioritätsänderung

Bei Durchtrennung eines Stranges von Signalleitungen könnte jede betroffene Signalbaugruppe aufgrund des Drahtbruchs einen Diagnosealarm (ruft OB 82 auf) auslösen - der Puffer für die Startinformationen des OB 82 ist in diesem Fall sofort gefüllt.

Damit bei einem Ereignis, das einen anderen Asynchronfehler-OB aufruft (z. B. Programmablauffehler, OB 85), die CPU nicht in den Betriebszustand "STOP" geht, weisen Sie z. B. dem OB 85 eine höhere Priorität zu als dem OB 82.

Siehe auch

Stromversorgungsfehler-OB (OB 81) (Seite 891)

Diagnosealarm-OB (OB 82) (Seite 894)

Ziehen/Stecken-OB (OB 83) (Seite 897)

CPU-Hardwarefehler-OB (OB 84) (Seite 902)

Programmablauffehler-OB (OB 85) (Seite 904)

Baugruppenträgerausfall-OB (OB 86) (Seite 908)

Kommunikationsfehler-OB (OB 87) (Seite 915)

Einstellungen bei Uhrzeitalarmen

Priorität

Die Priorität von Uhrzeitalarmen legt das Unterbrechungsverhalten der Alarm-OBs fest. Bei S7-300 ist die Priorität gegenwärtig nicht änderbar.

Aktiv

Aktivieren Sie das Optionskästchen des Uhrzeitalarm-OBs, wenn dieser bei einem Neustart automatisch gestartet werden soll. Oder aktivieren Sie den Uhrzeitalarm in Ihrem Anwenderprogramm mit dem Parameter OB_NR an der Anweisung ACT_TINT.

Ausführung

Wählen Sie aus, wie oft die Alarme ausgeführt werden sollen.

Es stehen Intervalle von **minütlich** bis **jährlich** zur Verfügung.

Die Intervalle beziehen sich auf die Einstellungen unter Startdatum und Uhrzeit.

Hinweis

Bei **Ausführung** = **Monatsletzter** ist der Tag des Startdatums nicht relevant (immer Monatsletzter!).

Startzeit

Geben Sie ein, an welchem Datum und zu welcher Uhrzeit der Alarm zum ersten Mal ausgeführt werden soll.

Beispiel

Startdatum	Uhrzeit
30.03.2006	10:15

Abhängig vom Parameter "Ausführung" erzeugt die CPU weitere Uhrzeitalarme in bestimmten Zeitintervallen. Die Startzeit bezieht sich immer auf die Basiszeit der CPU.

Hinweis

Bei der Ausführung "Monatlich" haben Sie nicht die Möglichkeit, den 29., 30. oder 31. des Monats als Startdatum zu wählen, da der Uhrzeitalarm im Monat Februar nicht kommen würde.

Wählen Sie stattdessen die Ausführung "Monatsletzter"! In diesem Fall ist der Tag des Startdatums nicht relevant (immer Monatsletzter!).

Teilprozessabbild

Einem Uhrzeitalarm können bei CPUs, die Teilprozessabbilder unterstützen, Teilprozessabbilder zugeordnet werden. Beim Start des Alarm-OBs wird dann das zugeordnete Teilprozessabbild aktualisiert, so dass der OB mit aktuellen Prozesswerten arbeitet. Bei der Einstellung "---" ist kein Teilprozessabbild zugeordnet.

Sonderfall: projektierter einmaliger Startzeitpunkt liegt in der Vergangenheit

Folgende Projektierung liegt vor:

- Ausführung: einmalig
- Aktiv: ja
- Startdatum/Uhrzeit: liegt in der Vergangenheit (bezogen auf Echtzeituhr der CPU)

Verhalten der CPU: Nach Kaltstart oder nach Neustart (Warmstart) wird der entsprechende Uhrzeitalam-OB vom Betriebssystem **einmalig aufgerufen!**

Siehe auch

Uhrzeitalarm-OBs (OB 10 bis OB 17) (Seite 871)

Einstellungen bei Weckalarmen

Ein Weckalarm ist ein periodisches Signal, das zum automatischen Aufruf eines Weckalarm-OBs führt. Im Folgenden sind die möglichen Einstellungen beschrieben.

Priorität

Die Priorität legt das Unterbrechungsverhalten der Alarm-OBs fest. Sie ist für S7-300-CPUs nicht änderbar.

Ausführung

Geben Sie die Zeitabstände in angezeigten Einheiten an (z. B. Millisekunden), in denen die Weckalarm-OBs bearbeitet werden.

Startzeitpunkt des Zeittaktes ist der Betriebszustandwechsel von "STOP" nach "RUN".

Phasenverschiebung

Falls die Phasenverschiebung einstellbar ist: Geben Sie die Zeit in Millisekunden (ms) an, um die der tatsächliche Ausführungszeitpunkt des Weckalarms verzögert werden soll.

Hinweis

Eine Phasenverschiebung ist dann sinnvoll, wenn mehrere Weckalarme aktiv sind. Ohne Phasenverschiebung würden Weckalarme gleichzeitig aufgerufen und entsprechend ihrer Priorität hintereinander bearbeitet.

Durch die Phasenverschiebung kann die Bearbeitungszeit der Weckalarme über den Zyklus hinweg verteilt werden.

Einheit

Bezieht sich auf den Parameter "Ausführung" und "Phasenverschiebung". Kann nur bei speziellen CPUs umgestellt werden, die eine Mikrosekunden-granulare Einstellung von Weckalarmen unterstützen.

Teilprozessabbild

Einem Weckalarm können bei CPUs, die Teilprozessabbilder unterstützen, Teilprozessabbilder zugeordnet werden. Beim Start des Weckalarm-OBs wird dann das zugeordnete Teilprozessabbild aktualisiert, so dass der OB mit aktuellen Prozesswerten arbeitet. Bei der Einstellung "---" ist kein Prozessabbild zugeordnet.

Siehe auch

Weckalarm-OBs (OB 30 bis OB 38) (Seite 876)

Einstellungen bei DPV1-Alarmen

Alarme für DPV1

Alarme können von einem DPV1-Slave gestellt werden, die eine Behandlung des alarmauslösenden Ereignisses in der Master-CPU sicherstellen. Auch im Betriebszustand "STOP" werden die Alarmdaten in der CPU ausgewertet (Aktualisierung des Diagnosepuffers und des Baugruppenzustands). Eine OB-Bearbeitung findet aber im "STOP" nicht statt.

Neben den aus der SIMATIC bekannten Alarmen (z. B. Diagnosealarm bei ET 200M) werden auch Alarme wie Statusalarm, Update-Alarm und Herstellerspezifischer Alarm unterstützt.

Siehe auch

Statusalarm-OB (OB 55) (Seite 880)

Update-Alarm-OB (OB 56) (Seite 882)

OB für herstellerspezifische Alarme (OB 57) (Seite 885)

Einstellungen bei Taktsynchronalarmen

Eigenschaften von Taktsynchronalarmen

Taktsynchronalarm-OBs werden synchron zum Äquidistanztakt bearbeitet.

DP-Mastersystem-Nr.

Wählen Sie das DP-Mastersystem aus, dessen DP-Slaves dem Alarm-OB zugeordnet werden sollen. Die Ein- und Ausgänge dieser DP-Slaves können dann mit dem Äquidistanztakt und dem Anwenderprogramm im Alarm-OB synchronisiert werden.

Teilprozessabbild(er)

Geben Sie hier die Nummern der Teilprozessabbilder ein, auf die der Alarm-OB zugreifen soll. Über die Teilprozessabbilder wird die zugeordnete dezentrale Peripherie synchron zum Äquidistanztakt aktualisiert.

Die zugeordneten Teilprozessabbilder müssen im Taktsynchronalarm-OB mit der Anweisung "SYNC_PI" und "SYNC_PO" aktualisiert werden. Die Anweisung "SYNC_PI" aktualisiert das Teilprozessabbild der Eingänge, "SYNC_PO" das Teilprozessabbild der Ausgänge.

Wenn die dezentralen Peripherieadressen verschiedenen Teilprozessabbildern zugeordnet wurden, so müssen Sie die Nummern der verwendeten Teilprozessabbilder durch Kommata getrennt eingeben (z. B. 1,2,3).

Verzögerungszeit

Die Verzögerungszeit ist die Zeit zwischen dem Global-Control-Telegramm und dem Start des OB 61 (bzw. OB 62..OB 64). In dieser Zeit wickelt der DP-Master den zyklischen Datenaustausch mit den DP-Slaves ab.

Wenn die Verzögerungszeit = 0 ist, dann startet der OB 6x mit dem Global-Control-Telegramm. In diesem Fall ist es sehr wahrscheinlich, dass ein Zugriff auf die taktsynchrone Peripherie über "SYNC_PI" zu Beginn des OBs fehlschlägt, da die Peripheriedaten noch nicht zur Verfügung stehen.

Wenn die Verzögerungszeit größer als der Vorgabe-Wert ist, dann wird OB-Laufzeit "verschenkt", da der Zugriff auf die taktsynchrone Peripherie bereits früher möglich wäre.

Siehe auch

Taktsynchronalarm-OBs (OB 61 bis OB 64) (Seite 887)

Äquidistanz und Taktsynchronität (Seite 2937)

Uhrzeitfunktionen

Grundlagen zu den Uhrzeitfunktionen

Alle S7-300/400-CPUs sind mit einer Uhr ausgestattet, entweder einer Echtzeituhr oder Software-Uhr. Die Uhr kann im Automatisierungssystem sowohl als Uhrzeitmaster als auch als Slave mit externer Synchronisation fungieren. Sie ermöglicht die Verwendung von Uhrzeitalarmen und Betriebsstundenzählern.

Uhrzeitformat

Die Uhr zeigt immer die Uhrzeit mit einer Mindestauflösung von 1 Sekunde und das Datum mit Wochentag an. Bei einigen CPUs ist auch die Anzeige von Millisekunden möglich.

Stellen und Lesen der Uhrzeit

Stellen der Uhrzeit

Sie können Uhrzeit und Datum der CPU-Uhr über folgende Anweisungen stellen und starten:

- "SET_CLK"
- "SET_CLKS", falls sie in der CPU vorhanden ist

Hinweis

Um Anzeigedifferenzen in HMI-Systemen in der Uhrzeit zu vermeiden, stellen Sie auf der CPU Winterzeit ein oder verwenden Sie eine zentrale Uhrzeitsynchronisation.

Lesen der Uhrzeit

Mit der Anweisung "READ_CLK" oder über einen Menübefehl vom Programmiergerät aus lesen Sie das aktuelle Datum und die Uhrzeit der CPU aus.

Uhrzeitsynchronisation SIMATIC-Verfahren

Bezug

Parametrierung der Schnittstellen-Eigenschaften in den folgenden Parametergruppen:

- Für die CPU: "Eigenschaften > Allgemein > Uhrzeit"
- Für die DP-Schnittstelle: "Eigenschaften > Allgemein > DP-Schnittstelle > Uhrzeitsynchronisation".

SIMATIC-Verfahren

Beim SIMATIC-Verfahren wird die Uhrzeit nach MMS-Uhrzeitnachrichten (MMS - Manufacturing Message Specifaction) gestellt. Die MMS-Uhrzeitnachrichten stammen von einem SIMATIC-Uhrzeitsender oder einer CPU, welche als Uhrzeit-Master projektiert ist.

Bei einem CP können die Uhrzeitnachrichten entweder von der lokalen CPU oder über LAN empfangen werden. Es ist einstellbar, ob der CP die empfangene Uhrzeit weiterleiten soll.

Gegenüber eines eventuell verfügbaren NTP-Verfahrens bietet das SIMATIC-Verfahren eine höhere Genauigkeit.

Parameter der Uhr

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Synchronisationsart
  Die Synchronisationseigenschaften können für die CPU intern (PLC) und extern (MPI) eingestellt werden.
  Wenn die Synchronisationsart "Als Master" eingestellt ist, dann ist das Zeitintervall für die automatische Synchronisation wählbar.
  Wenn in einem Netz mehr als eine Baugruppe mit Uhr vorhanden ist, müssen Sie einstellen, welche CPU bei der Synchronisation der Uhrzeit als Master und welche als Slave fungieren soll. Es darf nur einen Uhrzeit-Master geben.
- Zeitintervall
  Das Zeitintervall definiert den Zeitabstand der Uhrzeitanfragen.
- Korrekturfaktor
  Mit dem Korrekturfaktor wird für die CPU eine Abweichung der Uhr innerhalb von 24 Stunden ausgeglichen.
  Sie können positive oder negative Werte in ms eingeben.
  Beispiel: Wenn die Uhr nach 24 Stunden um 4 Sekunden nachgeht, müssen Sie einen Korrekturfaktor von "+4000 ms" eingeben.

Anweisungen für die Synchronisation der Uhrzeit

Um sicherzustellen, dass die Uhrzeit aller Baugruppen im Netz übereinstimmt, werden die Uhrzeit-Slaves vom Systemprogramm in regelmäßigen, parametrierbaren Abständen synchronisiert. Mit der Anweisung "SNC_RTCB" können Sie Datum und Uhrzeit unabhängig vom parametrierten Zeitintervall vom Uhrzeit-Master an den Uhrzeit-Slave übertragen.

Uhrzeitsynchronisation NTP-Verfahren

Bezug

Parametrierung der Schnittstellen-Eigenschaften in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Uhrzeitsynchronisation". Die Option "Uhrzeitsynchronisation über NTP-Server aktivieren" ist gewählt.

NTP-Verfahren (NTP: Network Time Protocol)

Beim NTP-Verfahren sendet das Gerät in regelmäßigen Zeitabständen Uhrzeitanfragen (im Client-Modus) an NTP-Server im Subnetz (LAN). Anhand der Anworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und die Uhrzeit der Station synchronisiert.

Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren. Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen (in Sekunden). Der Wertebereich des Intervalls liegt zwischen 10 Sekunden und einem Tag.

Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen; dies entspricht GMT (Greenwich Mean Time).

Parameter der Uhrzeitsynchronisation im NTP-Verfahren

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Uhrzeitsynchronisation über NTP-Server aktivieren
  Aktivieren Sie das Optionskästchen, wenn die interne Uhr mit dem NTP-Synchronisationsverfahren synchronisiert werden soll.
- Server 1-4
  Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren.
- Aktualisierungsintervall
  Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen.

Verwenden eines Betriebsstundenzählers

Definition

Ein Betriebsstundenzähler zählt die Einschaltzeiten eines angeschlossenen Betriebsmittels oder die Betriebsdauer der CPU als Summe der Betriebsstunden. Eine CPU kann bis zu 8 Betriebsstundenzähler haben. Die Nummerierung beginnt bei 0.

Im Betriebszustand "STOP" wird der Betriebsstundenzähler angehalten. Sein Wert bleibt auch nach Urlöschen erhalten. Bei "Neustart (Warmstart)" muss der Betriebsstundenzähler vom Anwenderprogramm wieder gestartet werden.

Betriebsstundenzähler programmieren

Sie können die Betriebsstundenzähler mit folgenden Anweisungen programmieren:

- Mit der Anweisung "SET_RTM" können Sie den Betriebsstundenzähler auf einen Anfangswert setzen.
- Mit der Anweisung "CTRL_RTM" können Sie den Betriebsstundenzähler starten oder stoppen.
- Mit der Anweisung "READ_RTM" können Sie die aktuelle Anzahl der Betriebsstunden und den Zustand des Zählers, entweder "gestoppt" oder "zählt", auslesen.

Verwenden von Taktmerkern

Taktmerker

Ein Taktmerker ist ein Merker, der seinen Binärzustand periodisch im Puls-Pausen-Verhältnis 1:1 ändert.

Welches Merkerbyte der CPU zum Taktmerkerbyte wird, bestimmen Sie bei der Parametrierung des Taktmerkers.

Nutzen

Taktmerker können Sie im Anwenderprogramm verwenden, um z. B. Leuchtmelder mit Blinklicht anzusteuern oder periodisch wiederkehrende Vorgänge anzustoßen, etwa das Erfassen eines Istwerts.

Mögliche Frequenzen

Jedem Bit des Taktmerkerbytes ist eine Frequenz zugeordnet. Die folgende Tabelle zeigt die Zuordnung:

Bit des Taktmerkerbytes	7	6	5	4	3	2	1	0
Periodendauer (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frequenz (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Hinweis

Taktmerker laufen asynchron zum CPU-Zyklus, d. h. in langen Zyklen kann sich der Zustand des Taktmerkers mehrfach ändern.

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

Web-Einstellungen

In dieser Parametergruppe wählen Sie:

- Start des Webservers der CPU
- Sprachen, in denen die Texte geladen werden sollen (Sprachabhängig sind z. B. die Texte im Diagnosepuffer oder Meldungen)

Webserver auf dieser Baugruppe aktivieren

Wenn das Optionskästchen aktiviert ist, wird nach Laden der Projektierungsdaten der Webserver der CPU gestartet und es können Informationen über einen Web-Browser aus der CPU gelesen werden.

Weitere Einstellungen betreffen die Aktualisierung von angezeigten Webseiten. Die Aktualisierungszeit können Sie bei Bedarf einstellen.

Zugriff nur über HTTPS zulassen

Aktivieren Sie das Optionskästchen, um die Web-Seiten des Webservers verschlüsselt an den Webbrowser zu übertragen.

Hinweis

Der Zugriff über HTTPS funktioniert nur, wenn in der CPU die Uhrzeit gestellt wurde!

Auswählen der Sprachen

Wählen Sie die Landessprachen für sprachabhängige Texte, die in die CPU geladen werden sollen. Die Anzahl wählbarer Sprachen ist CPU-abhängig.

Anzeige und Auswahl von CPU-spezifischen Sprachen

Angezeigt werden die Sprachen, die bei Web-Zugriff auf die CPU eingestellt werden können. Sie können diesen "Webserver-Sprachen" CPU-spezifisch eine Anzahl von Projektsprachen zuordnen. Für die ausgewählten Sprachen werden z. B. Texte von Diagnosepuffer-Einträgen bzw. Ereignistexten in die CPU geladen. Für diese Sprachen werden im Betrieb die Informationen auch in der entsprechenden Sprache im Web-Browser angezeigt.

Wenn Sie im Betrieb im Webbrowser eine Sprache wählen, die zuvor nicht geladen wurde, dann werden die Diagnosepuffer-Einträgen bzw. Ereignisse hexadezimal angezeigt.

Berücksichtigen Sie, dass jede Sprache Speicherplatz auf der CPU benötigt.

Benutzerverwaltung

Für den Zugriff auf die CPU über den Webbrowser können Sie Benutzer definieren, sie mit Rechten versehen und Passwörter vergeben.

Für den Standard-Benutzer, der sich nicht mit Passwort im Webbrowser anmeldet ("everybody"), können ebenfalls Rechte definiert werden.

Beobachtungstabellen

Sie haben die Möglichkeit, Ihre definierten Beobachtungstabellen über den Webbrowser für Anwender zugänglich zu machen. Wählen Sie die bestehenden über die Klappliste aus und bestimmen Sie über die Klappliste der zweiten Spalte, ob lesender bzw. schreibender Zugriff möglich sein soll.

Anzeigeklasse der Meldungen

Die ausgewählten Anzeigeklassen der Meldungen werden später auf den Webseiten der CPU angezeigt.

Sie können den Speicherbedarf der Web-SDBs verringern, indem Sie nur die Anzeigeklassen auswählen, den benötigt werden.

Die Anzeigeklasse stellen Sie bei den CPU-Eigenschaften "Meldeattribute" unter "Systemdiagnose" im Inspektorfenster ein.

Anwenderdefinierte Webseiten

Über den Abschnitt "Anwenderdefinierte Webseiten" haben Sie die Möglichkeit, eigene Webseiten in die CPU zu laden und damit über den Webbrowser eigene Webapplikationen zur Verfügung zu stellen.

Siehe auch

Wissenswertes zum Webserver (Seite 1317)

Anwenderseiten erstellen und laden (Seite 1330)

Reservieren von Kommunikationsressourcen

Funktionsweise der Vergabe von Verbindungsressourcen

Jede Kommunikationsverbindung benötigt auf der S7-CPU eine Verbindungsressource als Verwaltungselement für die Dauer des Bestehens der Kommunikationsverbindung. Entsprechend den technischen Daten stehen jeder S7-CPU eine bestimmte Anzahl von Verbindungsressourcen zur Verfügung, welche von verschiedenen Kommunikationsdiensten (PG/OP-Kommunikation, S7-Kommunikation oder S7-Basis-Kommunikation) belegt werden.

Vergabe der Verbindungsressourcen nach zeitlicher Reihenfolge

Wenn sich Kommunikationsdienste anmelden, werden die Verbindungsressourcen nach der zeitlichen Reihenfolge der Anmeldung belegt.

Manuelle Vergabe der Verbindungsressourcen

Um die Belegung der Verbindungsressourcen nicht nur von der zeitlichen Reihenfolge der Anmeldung verschiedener Kommunikationsdiensten abhängen zu lassen, besteht für folgende Dienste die Möglichkeit, Verbindungsressourcen zu reservieren:

- PG-Kommunikation und OP-Kommunikation
- S7-Basis-Kommunikation

Für die PG/OP-Kommunikation wird jeweils mindesten eine Verbindungsressource als Vorbelegung reserviert. Kleinere Werte sind nicht möglich.

Andere Kommunikationsdienste wie z. B. S7-Kommunikation mit den erweiterten Anweisungen "PUT" und "GET", können diese Verbindungsressource nicht belegen, auch wenn sie zeitlich früher ihre Verbindung aufbauen. Stattdessen werden hierfür die noch zur Verfügung stehenden Verbindungsressourcen belegt, welche nicht speziell für einen Dienst reserviert wurden.

Information zur maximalen Anzahl und bereits belegten Verbindungs-Ressourcen

In den CPU-Eigenschaften zur Reservierung von Verbindungsressourcen werden zusätzlich folgende Informationen angezeigt (nur lesbar):

- Anzahl bereits belegter Ressourcen für S7-Kommunikation. Es handelt sich hierbei um Ressourcen z. B. für einseitige S7-Verbindungen von einer anderen CPU zu dieser CPU oder um zweiseitige S7-Verbindungen. Diese Verbindungen wurden in der Netzansicht bereits projektiert.
- Maximale Anzahl Verbindungsressourcen für diese CPU.

Test, Inbetriebnahme und Routing

Test, Inbetriebnahme, Routing

Aktivieren Sie das Optionskästchen, wenn Sie z. B. PG-Funktionen, die bei der Inbetriebnahme und beim Testen benötigt werden, häufig über diese Schnittstelle durchführen wollen.

Das Aktivieren der Option "Test, Inbetriebnahme, Routing" bewirkt, dass die Schnittstelle zum aktiven Teilnehmer am PROFIBUS wird (d. h. die Schnittstelle ist am Token-Umlauf des PROFIBUS beteiligt). Folgende Funktionen sind dann möglich:

- Programmieren (z. B. Laden)
- Testen (Status/Steuern)
- S7-Routing (I-Slave als Netzübergang)

Die Busumlaufzeit kann sich verlängern. Bei zeitkritischen Anwendungen, und wenn S7-Routing sowie die Client-Funktionalität für die Kommunikation nicht erforderlich sind, sollte diese Option deshalb nicht aktiviert werden.

Wenn das Optionskästchen "Test, Inbetriebnahme, Routing" nicht aktiviert ist, arbeitet die Schnittstelle nur als Server für Kommunikationsdienste.

Server-Funktionalität bedeutet, dass nur der Partner (z. B. PG, OP oder Automatisierungssystem) die Kommunikation initiieren kann. Folgende Besonderheiten sind bei nicht aktiviertem "Test, Inbetriebnahme, Routing" zu beachten:

- S7-Routing ist nicht möglich
- Programmieren und Testen ist möglich, aber langsam (keine Verlängerung der Busumlaufzeit)

Einstellungen zur Systemdiagnose

Mit dem Diagnosesystem ist die Erfassung, Auswertung und Meldung eines Fehlers des Automatisierungssystems gemeint.

Beispiele für Fehler

- Fehler im CPU-Programm
- Ausfall von Baugruppen
- Drahtbruch zu Gebern und Aktoren

STOP-Ursache melden

Aktivieren Sie das Optionskästchen, wenn die CPU bei Übergang nach "STOP" die STOP-Ursache an die angemeldeten Anzeigegeräte (z. B. PG oder OP) melden soll.

Die Meldung wird auch in den Diagnosepuffer eingetragen.

Quittierungsgetriggertes Melden der erweiterten Anweisungen ALARM

Dieser Parameter ist relevant für CPUs, die für Prozessleitsysteme eingesetzt werden. Wenn diese Funktion aktiviert ist, melden die erweiterten Anweisungen "ALARM", "ALARM_8" und "ALARM_8P" (SFBs 33 bis 35) nur dann erneut einen Signalwechsel, wenn der vorherige Signalwechsel (d. h. die vorherige Kommend-Meldung) quittiert wurde. Auf diese Weise wird verhindert, dass Meldungen gehäuft auftreten und nicht verarbeitet werden können.

Einstellmöglichkeiten für die Schutzstufe

Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Schutzstufen der S7-CPUs nutzen und welchen Einfluss die Art der S7-CPU auf die Parametrierung und Nutzung der jeweiligen Schutzstufe hat.

Voreingestelltes Verhalten

Im voreingestellten Verhalten der Schutzstufe gibt es keine Passwortvergabe. Diese Schutzstufe heißt "Kein Schutz".

Eine Passworteingabe ist hier nur möglich, wenn Sie die Option "Durch Passwort aufhebbar" aktiviert haben. Durch diese Option können Sie auch dann schreibend auf die CPU im RUN zugreifen, wenn durch die Anweisung "PROTECT" im Anwenderprogramm die CPU mit einem Schreibschutz belegt wurde.

Hinweis

Falls bei Ihrer CPU die voreingestellte Schutzstufe aktiv ist und Ihre CPU die Anweisung "PROTECT" zur Verfügung stellt, können Sie mit dieser Anweisung zwischen der voreingestellten Schutzstufe "Kein Schutz" und der Schutzstufe "Schreibschutz" wechseln.

Weitere passwortgeschützte Schutzstufen

Für die folgenden Schutzstufen ist die Vergabe eines Passwortes erforderlich:

- Schreibschutz: Mit dieser Schutzstufe ist ohne Angabe des richtigen Passwortes nur lesender Zugriff möglich, unabhängig von der Stellung des Schlüsselschalters. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes ist erforderlich.
- Schreib-/Leseschutz: Mit dieser Schutzstufe ist ohne Angabe des richtigen Passwortes weder lesender noch schreibender Zugriff möglich, unabhängig von der Stellung des Schlüsselschalters. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes erforderlich.

Für "Passwort-Kenner" sind schreibende und lesende Zugriffe möglich, unabängig von der parametrierten Schutzstufe.

Verhalten einer Passwort-geschützten Baugruppe im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft und im Falle eines Passwortschutzes zur Passworteingabe aufgefordert.

Beispiel: Die Baugruppe wurde mit Schreibschutz parametriert und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht mit Passwort anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird.

Hinweis

Funktionen zur Prozessführung, -beobachtung und -kommunikation können Sie nicht einschränken. So kann z. B. die Funktion "Uhrzeit/Datum stellen" nicht durch ein Passwort verriegelt werden.

Testbetrieb und Prozessbetrieb

Einführung

Bei einigen CPU-Typen können Sie die Betriebsart für den Programmtest einstellen.

Im Prozessbetrieb werden die Testfunktionen wie Programmstatus oder Variable beobachten/steuern so eingeschränkt, dass die eingestellte zulässige Zykluszeiterhöhung nicht überschritten wird. Das Testen mit Haltepunkten und schrittweiser Programmausführung kann nicht ausgeführt werden.

Im Testbetrieb sind alle Testfunktionen über PG/PC ohne Einschränkung nutzbar, die auch größere Verlängerungen der Zykluszeit bewirken können.

Einstellmöglichkeiten für das Betriebsverhalten

Sie können beim Testen mit dem Programmstatus im Inspektorfenster unter "Eigenschaften > Betriebsart" folgende Betriebsarten einstellen:

- Testbetrieb
  Alle Testfunktionen sind ohne Einschränkung nutzbar. Größere Verlängerungen der CPU-Zykluszeit können auftreten, da z. B. der Status von Anweisungen in programmierten Schleifen bei jedem Durchlauf ermittelt wird.
- Prozessbetrieb
  Die Testfunktion Programmstatus wird eingeschränkt, um eine möglichst geringe Belastung der Zykluszeit zu gewährleisten:
  - Es sind z. B. keine Aufrufbedingungen erlaubt.
  - Die Statusanzeige einer programmierten Schleife wird an der Rücksprungstelle abgebrochen.
  - Die Testfunktionen "HALT" und schrittweise Programmausführung können nicht ausgeführt werden.

Hinweis

Wenn sich die CPU im Testbetrieb befindet, müssen Sie dafür sorgen, dass die CPU bzw. der Prozess große Zykluszeitverlängerungen "vertragen" kann.

Mit der Option "Prozessbetrieb" verhindern Sie, dass im Programmiereditor die Aufrufumgebung eingestellt werden kann.

Parameter der Kompakt-CPUs

Funktion Zählen

Eigenschaften Zählen

Beschreibung

Wenn Sie die CPU mit integrierter Zähl- und Messfunktion in der Netzsicht bzw. Gerätesicht markiert haben, können Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Zählen" für jeden der aufgeführten Kanäle die Zählfunktion parametrieren.

Wenn Sie Kanal 0 oder Kanal 1 als Zählkanal verwenden, können Sie die Technologie "Positionieren" nicht mehr verwenden.

Hinweis

Die Anzahl der nutzbaren Zählkanäle hängt von der eingesetzten CPU ab.

Im Auswahlfeld "Betriebsart" können Sie für jeden Kanal festlegen, wie Sie mit der CPU zählen wollen. Dazu haben Sie verschiedene Optionen. Als Voreinstellung ist "nicht parametriert" ausgewählt.

Siehe auch

Dokumentation zu Technologischen Funktionen der CPU (http://support.automation.siemens.com/WW/view/de/12429336/0/de)

Alarmauswahl

Für die Funktion "Zählen" können Sie verschiedene Alarme auswählen.

Voraussetzung

- Die CPU ist selektiert.
- Die Alarmauswahl ist unter "Eigenschaften > Allgemein > Zählen > Alarmauswahl" zu finden.

Alarmauswahl (Grundparameter)

Wählen Sie in der Klappliste "Alarmauswahl" aus, welche Alarme die CPU auslösen soll. Sie haben folgende Auswahlmöglichkeiten:

- Keine
- Diagnose
- Prozess
- Diagnose und Prozess

Als Voreinstellung ist "Keine" ausgewählt.

Die Ereignisse, die einen Prozessalarm auslösen sollen, wählen Sie, abhängig von der eingestellten Betriebsart.

Zählen

Ereignisse, die einen Prozessalarm auslösen können:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)
- Schließen des HW-Tors (bei geöffnetem SW-Tor)
- Bei Erreichen (Ansprechen) des Vergleichers
- Bei Zählimpuls
- Überlauf (bei Überschreiten der oberen Zählgrenze)
- Unterlauf (bei Unterschreiten der unteren Zählgrenze)

Frequenzmessen

Ereignisse, die einen Prozessalarm auslösen können:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)
- Schließen des HW-Tors (bei geöffnetem SW-Tor)
- Messende
- Überschreiten der Obergrenze
- Unterschreiten der Untergrenze

Pulsweitenmodulation

Ereignisse, die einen Prozessalarm auslösen können:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)

Kanalspezifische Parameter

Allgemein

Unter "Eigenschaften > Allgemein > Zählen > Kanal 0 (... Kanal n)" werden die Kanalspezifischen Parameter eingestellt. Für die Betriebsart stehen sechs Optionen zur Verfügung "Nicht konfiguriert", "Endloses Zählen", "Einmaliges Zählen", "Periodisches Zählen", "Frequenzmessen" und "Pulsweitenmodulation".

Als Vorbelegung der Betriebsart ist "Nicht konfiguriert" ausgewählt.

Zählen

Betriebsparameter Zählen

Hauptzählrichtung

In den Betriebsarten "Einmaliges Zählen" und "Periodisches Zählen" können Sie in der Klappliste "Hauptzählrichtung" auswählen, ob Sie den Zähler ohne Hauptzählrichtung oder mit der Hauptzählrichtung vorwärts bzw. rückwärts betreiben wollen. Als Vorbelegung ist "keine" ausgewählt.

In der Betriebsart "Endloses Zählen" ist die Klappliste "Hauptzählrichtung" nicht vorhanden.

Torfunktion

In der Klappliste "Torfunktion" können Sie auswählen, ob der Zählvorgang abgebrochen oder unterbrochen wird, wenn das Tor geschlossen wird.

Wählen Sie hier "Zählvorgang abbrechen" so beginnt der Zählvorgang nach Schließen des Tors und erneutem Torstart wieder ab dem Ladewert. Wählen Sie hier "Zählvorgang

unterbrechen" so wird der Zählvorgang nach Schließen des Tors und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Als Vorbelegung ist "Zählvorgang abbrechen" ausgewählt.

Anfangswert/Endwert

Je nachdem, welche Hauptzählrichtung Sie in der entsprechenden Klappliste ausgewählt haben, können Sie im Eingabefeld "Anfangswert/Endwert" einen Wert für die entsprechende Zählgrenze eintragen.

Zwischen der Hauptzählrichtung und diesem Eintrag besteht folgender Zusammenhang:

- Bei Hauptzählrichtung vorwärts tragen Sie hier einen Endwert ein.
- Bei Hauptzählrichtung rückwärts tragen Sie hier einen Anfangswert ein.

Wenn Sie keine Hauptzählrichtung ausgewählt haben, ist dieses Eingabefeld nicht vorhanden.

In der Voreinstellung (nur sichtbar, wenn Sie eine Hauptzählrichtung ausgewählt haben) ist jeweils "2147483647" eingetragen.

Vergleichswert

Im Eingabefeld "Vergleichswert" können Sie einen Wert eintragen, der dem Digitalausgang, dem Statusbit Vergleicher STS_CMP und einem Prozessalarm zugeordnet ist. Der Zählwert wird mit dem Vergleichswert verglichen. Abhängig vom Ergebnis dieses Vergleiches kann der Ausgang und das Statusbit Vergleicher STS_CMP gesetzt werden und es kann der Prozessalarm "Erreichen des Vergleichers" generiert werden.

Wie der Ausgang abhängig vom Zählerstand konkret gesetzt wird, stellen Sie in der Gruppe Ausgang ein.

Als Vorbelegung ist für den Vergleichswert "0" eingetragen.

Hysterese

Im Eingabefeld "Hysterese" tragen Sie einen Hysteresebereich zwischen 0 und 255 (0 und 1 bedeuten: Hysterese abgeschaltet) ein.

Ein Geber kann an einer bestimmten Position stehenbleiben und dann um diese Position "pendeln". Dieser Zustand führt dazu, dass der Zählerstand um einen bestimmten Wert herum schwankt. Liegt nun in diesem Schwankungsbereich zum Beispiel ein Vergleichswert, würde der zugehörige Ausgang im Rhythmus dieser Schwankungen ein- und ausgeschaltet werden. Um dieses Schalten bei kleinen Schwankungen zu verhindern, ist die CPU mit einer parametrierbaren Hysterese ausgestattet.

Sie können die Hysterese über das Anwenderprogramm in Betrieb ändern.

Die Hysterese wirkt auf den Ausgang und auf die Statusbits Vergleicher STS_CMP, Überlauf STS_OFLW, Unterlauf STS_UFLW und Nulldurchgang STS_ZP.

Als Vorbelegung ist "0" eingetragen.

Zeitbasis

Sie können wählen, ob bei einer max. Zählfrequenz von 1 kHz die Periodendauer in 125 ns oder 1 µs Einheiten gemessen wird. Bei einer max. Zählfrequenz größer als 1 kHz erfolgt keine Periodendauermessung. Sie können bis zu einer max. Zählfrequenz von 1 kHz die Periodendauer des Zählsignals messen.

Bei einer max. Zählfrequenz von 1 kHz wird immer die Zeit zwischen zwei aufeinanderfolgenden Zählflanken gemessen.

Bei Zeitbasis 1 µs können Periodendauern von maximal 4294 s = 71 min, 34 s und bei einer Zeitbasis von 125 ns von maximal 178 s = 2 min, 58 s gemessen werden.

Liegen die Zählflanken zeitlich weiter auseinander, ist die gemessene Periodendauer falsch, da ein Überlauf nicht berücksichtigt wird.

Eingang 0 (Zählen)

Signalauswertung

In der Klappliste "Signalauswertung" wählen Sie aus, welche Signale der angeschlossene Geber liefert. Sie können aus folgenden Signalen auswählen:

- Impuls und Richtung
- Drehgeber einfach
- Drehgeber zweifach
- Drehgeber vierfach

Als Voreinstellung ist "Impuls und Richtung" eingetragen.

HW-Tor

Mit dem Optionskästchen "HW-Tor" wählen Sie aus, ob Sie ein Hardware-Tor verwenden wollen oder nicht.

Als Voreinstellung ist dieses Optionskästchen nicht angewählt.

Zählrichtung invertiert

Mit dem Optionskästchen "Zählrichtung invertiert" wählen Sie aus, ob Sie die Zählrichtung invertieren wollen oder nicht.

Als Voreinstellung ist dieses Optionskästchen nicht angewählt.

Ausgabe (Zählen)

Verhalten des Ausgangs

In der Klappliste "Verhalten des Ausgangs" wählen Sie eine der folgenden Alternativen aus:

- Kein Vergleich
  Der Ausgang wird wie ein normaler Ausgang geschaltet.
  Die SFB-Eingangsparameter CTRL_DO und SET_DO sind unwirksam.
  Die Statusbits STS_DO und STS_CMP (Status Vergleicher im IDB) bleiben rückgesetzt.
- Setzen bei Zählerstand ≥ Vergleichswert
  Der Ausgang wird gesetzt, sobald der Zählerstand den Vergleichswert erreicht und bleibt gesetzt, solange der Zählerstand größer oder gleich dem Vergleichswert ist.
- Setzen bei Zählerstand ≤ Vergleichswert
  Der Ausgang wird gesetzt, sobald der Zählerstand den Vergleichswert erreicht und bleibt gesetzt, solange der Zählerstand kleiner oder gleich dem Vergleichswert ist.
- Impuls bei Erreichen des Vergleichswert
  Zur Anpassung an die verwendeten Aktoren können Sie eine Impulsdauer vorgeben. Der Ausgang wird für die eingestellte Impulsdauer gesetzt, sobald der Zählerstand den Vergleichswert erreicht hat. Wenn Sie eine Hauptzählrichtung eingestellt haben, wird der Ausgang nur bei Erreichen des Vergleichswertes aus der Hauptzählrichtung gesetzt.

Als Voreinstellung ist "Kein Vergleich" ausgewählt.

Impulsdauer

Wenn Sie in der Klappliste "Verhalten des Ausgangs" die Alternative "Impuls bei Erreichen des Vergleichswert" gewählt haben, müssen Sie im Eingabefeld "Impulsdauer" die Impulsdauer für das Ausgangssignal in ms eintragen. Die maximale Dauer beträgt 510 ms. Sie können nur geradzahlige Werte eintragen.

Die Impulsdauer beginnt mit dem Setzen des jeweiligen Digitalausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 2 ms.

Wenn Sie als Impulsdauer 0 ms eintragen, wird der Ausgang so lange gesetzt, bis die Vergleichsbedingung nicht mehr erfüllt ist.

Als Voreinstellung sind "0 ms" eingetragen.

Hinweis

Wenn der Ausgang bei Erreichen des Vergleichswertes gesetzt wird, wird auch das Bit STS_CMP, "Status Vergleicher" gesetzt.

Max. Frequenz

Zählsignale/HW-Tor

In dieser Klappliste können Sie die maximale Frequenz der Signale Spur A/Impuls, Spur B/ Richtung und Hardwaretor in festen Stufen einstellen. Der Maximalwert ist CPU-spezifisch:

CPU	Wertebereich	Vorbelegung
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Latch

In dieser Klappliste können Sie die maximale Frequenz des Latch-Signals in festen Stufen einstellen. Der Maximalwert ist CPU-spezifisch:

CPU	Wertebereich	Vorbelegung
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	10 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	10 kHz

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Prozessalarm bei (Zählen)

Beschreibung

Wenn Sie das oder die entsprechenden Optionskästchen anklicken, löst das zugehörige Ereignis einen Prozessalarm aus. Die folgenden Ereignisse können einen Prozessalarm auslösen:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)
- Schließen des HW-Tors (bei geöffnetem SW-Tor)
- Bei Erreichen (Ansprechen) des Vergleichers
- Bei Zählimpuls
- Überlauf (bei Überschreiten der oberen Zählgrenze)
- Unterlauf (bei Unterschreiten der unteren Zählgrenze)

Damit ein Prozessalarm auch tatsächlich von der CPU generiert wird, müssen Sie dies auch im Register Grundparameter freigegeben haben.

Als Voreinstellung ist kein Optionskästchen angewählt.

Das Auslösen eines Prozessalarms bei jeder Zählflanke führt bei höheren Zählfrequenzen zu einer starken CPU Auslastung. Treten die Prozessalarme in Submodul "Zählen" schneller auf, als sie im Prozessalarm-OB (OB 40) verarbeitet werden können, dann führt dies bei freigegebenem Diagnosealarm zu einer Diagnose "Prozessalarm ging verloren".

Zuweisung der Eingangsdaten

Zuweisung der Eingangsdaten

Sie können wählen, ob bei einer max. Zählfrequenz von 1 kHz in den Eingangsdaten (E-Daten) des Submoduls "Zählen" der Zählwert oder die Periodendauer lesbar ist. Bei einer max. Zählfrequenz größer als 1 kHz ist nur "Zählwert" möglich.

Sie können bis zu einer max. Zählfrequenz von 1 kHz die Periode des Zählsignals messen.

Bei einer max. Zählfrequenz von 1 kHz wird immer die Zeit zwischen zwei aufeinanderfolgenden Zählflanken gemessen. Bei einer max. Zählfrequenz von größer 1 kHz erfolgt keine Periodendauermessung, der Wert ist 0.

Frequenzmessen

Frequenzmessen

Beschreibung

In dieser Betriebsart zählt die CPU die Impulse, die in einem vorgegebenen Zeitintervall (Integrationszeit) eintreffen und berechnet aus der Anzahl der Impulse und der Dauer der Integrationszeit die Frequenz.

Der Wert der ermittelten Frequenz wird in der Einheit "mHz" zur Verfügung gestellt.

Betriebsparameter Frequenzmessen

Beschreibung

In dieser Betriebsart zählt die CPU die Impulse, die in einer parametrierten Integrationszeit eintreffen.

Die Messung wird während der Integrationszeit durchgeführt. Wenn die Integrationszeit abgelaufen ist, wird der Messwert aktualisiert.

Lag in der Integrationszeit nicht mindestens eine steigende Flanke, so wird abhängig von der Parametrierung als Messwert 0 oder der gemittelte Wert zurückgemeldet.

Integrationszeit

Im Eingabefeld "Integrationszeit" können Sie die Integrationszeit zwischen 10 ms und 10000 ms eintragen.

Als Voreinstellung sind "100 ms" eingetragen.

Unterer Grenzwert und Oberer Grenzwert

In den Eingabefeldern "Unterer Grenzwert" und "Oberer Grenzwert" geben Sie Werte ein, mit denen die ermittelte Frequenz nach abgelaufener Integrationszeit jeweils verglichen wird. Für diese Grenzwertüberwachung sind folgende CPU-abhängigen Wertebereiche zulässig:

- Untergrenze
  - CPU 312C: 0 bis 9 999 999 mHz
  - CPU 313C: 0 bis 29 999 999 mHz
  - CPU 314C: 0 bis 59 999 999 mHz
- Obergrenze
  - CPU 312C: 1mHz bis 10 000 000 mHz
  - CPU 313C: 1mHz bis 30 000 000 mHz
  - CPU 314C: 1mHz bis 60 000 000 mHz

  Hinweis

  Bei der Eingabe wird überprüft, ob Obergrenze > Untergrenze ist.

Als Voreinstellung sind für die Untergrenzen jeweils 0 mHz eingetragen. Als Voreinstellung sind für die Obergrenzen die folgenden CPU-abhängigen Werte eingetragen:

- CPU 312C: 10 000 000 mHz
- CPU 313C: 30 000 000 mHz
- CPU 314C: 60 000 000 mHz

Max. Zählfrequenz/HW-Tor

In dieser Klappliste können Sie die maximale Frequenz der Signale Spur A/Impuls, Spur B/ Richtung und Hardwaretor in festen Stufen einstellen. Der Maximalwert ist CPU-spezifisch:

CPU	Wertebereich	Vorbelegung
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

Messwert ausgeben

Sie können auswählen, ob der Messwert direkt oder gemittelt ausgegeben wird. Hierzu können Sie bei "Messwert ausgeben" in der Klappliste die entsprechende Option bei "Direkt" bzw. "Gemittelt" anklicken.

Am Ende der Integrationszeit, in der mindestens eine positive Flanke fällt, wird die gemessene Frequenz angezeigt.

Lag in der parametrierten Integrationszeit nicht mindestens eine positive Flanke, wird

- bei Frequenz direkt am Ende der Integrationszeit der Wert "0" ausgegeben.
- bei Frequenz gemittelt der letzte Wert über die folgenden Messintervalle ohne positive Flanke verteilt (f ≥ 1 mHz). Dies entspricht einer Verlängerung der Integrationszeit. Dazu wird der letzte gemessene Wert durch die Anzahl der Messintervalle ohne positive Flanke geteilt.
  Beispiel: War der letzte Messwert 12 000 mHz, so wird nach drei Messintervallen der Wert 4000 mHz ausgegeben.

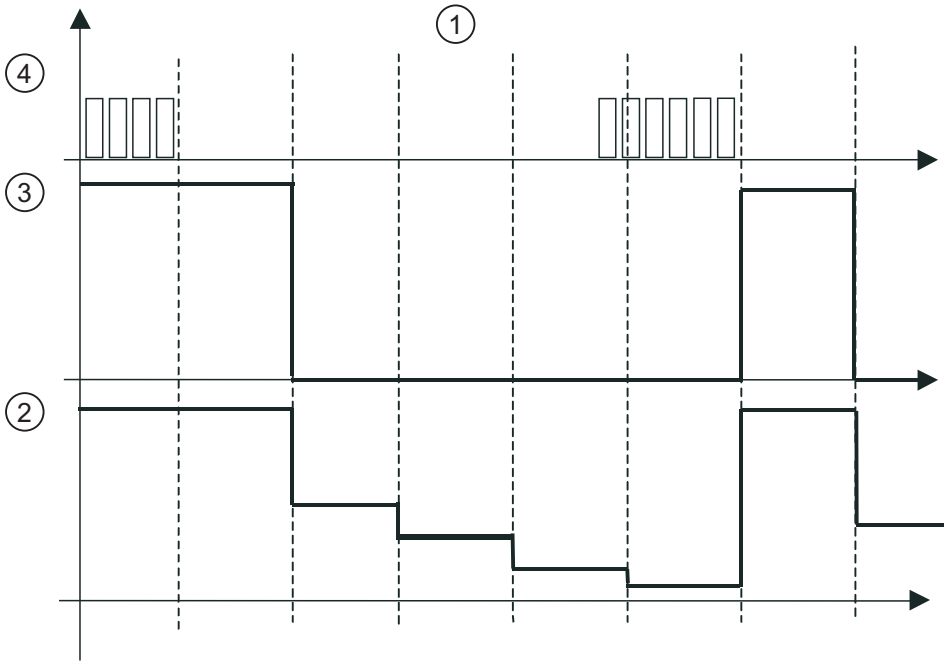

① Zeitintervall
② Frequenz gemittelt
③ Frequenz direkt
④ Impulse

Als Voreinstellung ist aus der Klappliste die Option "Direkt" ausgewählt.

Eingang 0 (Frequenzmessen)

Signalauswertung

In der Klappliste "Zählrichtung invertiert" wählen Sie aus, welche Signale der angeschlossene Geber liefert. Sie können aus folgenden Signalen auswählen:

- Impuls und Richtung
- Drehgeber einfach

Als Voreinstellung ist "Impuls und Richtung" eingetragen.

HW-Tor

Mit dem Optionskästchen "HW-Tor" wählen Sie aus, ob Sie ein Hardware-Tor verwenden wollen oder nicht.

Als Voreinstellung ist dieses Optionskästchen nicht angewählt.

Zählrichtung invertiert

Mit dem Optionskästchen "Zählrichtung invertiert" wählen Sie aus, ob Sie die Zählrichtung invertieren wollen oder nicht.

Als Voreinstellung ist dieses Optionskästchen nicht angewählt.

Ausgabe (Frequenzmessen)

Verhalten des Ausgangs

In der Klappliste "Verhalten des Ausgangs" wählen Sie eine der folgenden Alternativen aus:

- Kein Vergleich
  Der Ausgang wird wie ein normaler Ausgang geschaltet.
  Die SFB-Eingangsparameter MAN_DO und SET_DO sind unwirksam.
  Das Statusbit STS_DO bleibt rückgesetzt.
- Setzen wenn außerhalb der Grenzen
  Der Ausgang wird gesetzt, sobald die gemessene Frequenz größer als die Obergrenze oder kleiner als die Untergrenze ist.
- Setzen wenn unterhalb der Untergrenze
  Der Ausgang wird gesetzt, sobald die gemessene Frequenz kleiner als die Untergrenze ist.
- Setzen wenn oberhalb der Obergrenze
  Der Ausgang wird gesetzt, sobald die gemessene Frequenz größer als die Obergrenze ist.

Als Voreinstellung ist "Kein Vergleich" ausgewählt.

Prozessalarm bei (Frequenzmessen)

Beschreibung

Wenn Sie das oder die entsprechenden Optionskästchen anklicken, löst das zugehörige Ereignis einen Prozessalarm aus. Die folgenden Ereignisse können einen Prozessalarm auslösen:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)
- Schließen des HW-Tors (bei geöffnetem SW-Tor)
- Messende
- Oberer Grenzwert überschritten
- Unterer Grenzwert unterschritten

Damit ein Prozessalarm auch tatsächlich von der CPU generiert wird, müssen Sie dies auch im Register Grundparameter freigegeben haben.

Als Voreinstellung ist kein Optionskästchen angewählt.

Pulsweitenmodulation

Pulsweitenmodulation

Beschreibung

Der von Ihnen vorgegebene Ausgabewert wird von der CPU in eine Impulsfolge mit entsprechendem Impuls-/Pauseverhältnis (Pulsweitenmodulation) umgewandelt.

Die Impulsfolge wird nach Ablauf der parametrierten Einschaltverzögerung am Digitalausgang DO ausgegeben (Ausgabesequenz).

Betriebsparameter Pulsweitenmodulation

Ausgabeformat

In der Klappliste "Ausgabeformat" wählen Sie den Wertebereich des Ausgabewertes indem Sie die Option "Promille" oder "S7-Analogwert" anklicken.

Liegt Ihr Ausgabewert zwischen 0 und 1000 klicken Sie die Option "Promille" an.

Ist Ihr Ausgabewert ein SIMATIC S7 Analogwert (zwischen 0 und 27648), klicken Sie die Option "S7-Analogwert" an.

Als Voreinstellung ist das Optionsfeld "Promille" angewählt.

Zeitbasis

In der Klappliste "Zeitbasis" wählen Sie die Zeitbasis, die für die Auflösung und für den Wertebereich der Periodendauer, der Mindestimpulsdauer und der Einschaltverzögerung gelten soll.

Wenn Sie die Option "1 ms" anklicken, dann können Sie Zeiten in einer Auflösung von 1 ms einstellen.

Wenn Sie die Option "0,1ms" anklicken, dann können Sie Zeiten in einer Auflösung von 0,1 ms einstellen.

Als Voreinstellung ist die Option "0,1 ms" angewählt.

Einschaltverzögerung

Im Eingabefeld "Einschaltverzögerung" tragen Sie einen Wert für die Zeit ein, die ab dem Start der Ausgabesequenz bis zur Ausgabe des Impulses abläuft.

Sie tragen den Wert in der oben eingestellten Auflösung ein.

Wenn Sie in der Klappliste "Zeitbasis" die Option "1 ms" angeklickt haben, dann können Sie Zeiten von 0 ms bis 65535 ms mit einer Auflösung von 1 ms einstellen.

Wenn Sie in der Klappliste "Zeitbasis" die Option "0,1 ms" angeklickt haben, dann können Sie Zeiten von 0 ms bis 6553,5 ms mit einer Auflösung von 0,1 ms einstellen.

Als Voreinstellung ist "0" eingetragen.

Die Impulsfolge wird nach Ablauf der Einschaltverzögerung am Digitalausgang DO ausgegeben (Ausgabesequenz).

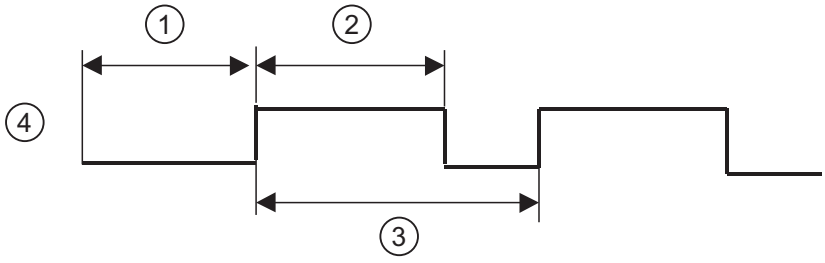

① Einschaltverzögerung
② Impulsdauer
③ Periodendauer
④ Ausgang DO

Als Voreinstellung ist der Wert "0" eingetragen.

Periodendauer

Die Periodendauer definiert die Länge der Ausgabesequenz, bestehend aus Impulsdauer und Impulspause.

Im Eingabefeld "Periodendauer" tragen Sie einen Wert in der oben eingestellten Auflösung ein.

Wenn Sie in der Klappliste "Zeitbasis" die Option "1 ms" angeklickt haben, dann können Sie Zeiten von 1 ms bis 65535 ms mit einer Auflösung von 1 ms einstellen.

Wenn Sie in der Klappliste "Zeitbasis" die Option "0,1 ms" angeklickt haben, dann können Sie Zeiten von 0,4 ms bis 6553,5 ms mit einer Auflösung von 0,1 ms einstellen.

Als Voreinstellung ist der Wert "20000" eingetragen.

Mindestimpulsdauer

Mit der Mindestimpulsdauer können Sie kurze Ausgangsimpulse und kurze Impulspausen unterdrücken. Alle Impulse bzw. Pausen, die kleiner als die Mindestimpulsdauer sind, werden unterdrückt.

Im Eingabefeld "Mindestimpulsdauer" tragen Sie einen Wert in der oben eingestellten Auflösung ein.

Wenn Sie in der Klappliste "Zeitbasis" die Option "1 ms" angeklickt haben, dann können Sie Werte von 0 bis Periodendauer/2 eintragen.

Wenn Sie in der Klappliste "Zeitbasis" die Option "0,1 ms" angeklickt haben, dann können Sie Werte von 2 bis Periodendauer/2 eintragen.

Wenn Sie für die Zeitbasis 1 ms ausgewählt haben und als Mindestimpulsdauer 0 eintragen so wird die Mindestimpulsdauer intern auf 0,2 ms eingestellt.

Als Voreinstellung ist der Wert "2" eingetragen.

Eingang 0 (Pulsweitenmodulation)

HW-Tor

Mit dem Optionskästchen "HW-Tor" wählen Sie aus, ob Sie ein Hardware-Tor verwenden wollen oder nicht.

Als Voreinstellung ist dieses Optionskästchen nicht angewählt.

Ein Start der Pulsweitenmodulation mit Hardware-Tor ist nur möglich, wenn Sie zuerst das Software-Tor öffnen und anschließend am Digitaleingang "Hardware-Tor" eine positive Flanke erzeugen.

Ein Stoppen der Pulsweitenmodulation ist nur durch eine negative Flanke am Software-Tor möglich. Der Zustand der Hardware-Tores ist beliebig.

Filterfrequenz

HW-Tor

Diese Klappliste ist nur aktiv, wenn unter "Eingang" die Option "HW-Tor" angewählt ist. Hier können Sie die maximale Frequenz des Signals Hardwaretor in festen Stufen einstellen. Der Maximalwert ist CPU-spezifisch:

CPU	Wertebereich	Vorbelegung
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

Prozessalarm bei (Pulsweitenmodulation)

Beschreibung

Wenn Sie das entsprechende Optionskästchen anklicken, löst das zugehörige Ereignis einen Prozessalarm aus. Das folgende Ereignis kann einen Prozessalarm auslösen:

- Öffnen des HW-Tors (bei geöffnetem SW-Tor)

Damit ein Prozessalarm auch tatsächlich von der CPU generiert wird, müssen Sie dies auch im Register Grundparameter freigegeben haben.

Als Voreinstellung ist kein Optionskästchen angewählt.

Endloses Zählen

Beschreibung

Die Betriebsart "Endloses Zählen" ist folgendermaßen definiert:

Hat der Zähler die obere Zählgrenze erreicht und es kommt ein weiterer Zählimpuls, so springt der Zähler auf die untere Zählgrenze und zählt von dort aus weiter.

Hat der Zähler die untere Zählgrenze erreicht und kommt ein weiterer Zählimpuls, so springt er auf die obere Zählgrenze und zählt von dort aus weiter.

Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.

Das Starten und Stoppen des Zählvorgangs geschieht in dieser Betriebsart über die Torfunktion.

Einmaliges Zählen

Beschreibung

Hat der Zähler beim Zählen in Hauptzählrichtung vorwärts den Endwert - 1 erreicht und kommt ein weiterer Zählimpuls, so springt der Zähler auf den Zählbereichsanfang (Ladewert) und bleibt stehen, auch wenn noch weitere Zählimpulse folgen.

Hat der Zähler beim Zählen in Hauptzählrichtung rückwärts den Zählwert = 1 erreicht und kommt ein weiterer Zählimpuls, so springt er auf den Startwert (Ladewert) und bleibt stehen, auch wenn noch weitere Zählimpulse folgen.

Ist keine Hauptzählrichtung eingestellt, so sind die Zählgrenzen auf den maximalen Zählbereich eingestellt.

Das Starten und Stoppen des Zählvorgangs geschieht in dieser Betriebsart über die Torfunktion.

Wird das Tor geöffnet, können Zählimpulse zu dem Zähler gelangen, der Zählvorgang wird gestartet. Wird das Tor geschlossen können keine Zählimpulse mehr zum Zähler gelangen, der Zählvorgang ist gestoppt.

Periodisches Zählen

Beschreibung

Hat der Zähler beim Zählen in Hauptzählrichtung vorwärts den Endwert - 1 erreicht und kommt ein weiterer Zählimpuls, so springt der Zähler auf den Ladewert und zählt von dort aus weiter.

Hat der Zähler beim Zählen in Hauptzählrichtung rückwärts den Zählwert = 1 erreicht und kommt ein weiterer Zählimpuls, so springt er auf den Startwert (Ladewert) und zählt von dort aus weiter.

Ist keine Hauptzählrichtung eingestellt, so sind die Zählgrenzen auf den maximalen Zählbereich eingestellt.

Das Starten und Stoppen des Zählvorgangs geschieht in dieser Betriebsart über die Torfunktion.

Wird das Tor geöffnet, können Zählimpulse zu dem Zähler gelangen, der Zählvorgang wird gestartet. Wird das Tor geschlossen können keine Zählimpulse mehr zum Zähler gelangen, der Zählvorgang ist gestoppt.

Hauptzählrichtung

Beschreibung

Sie können als Hauptzählrichtung "vorwärts" oder "rückwärts" parametrieren. Damit legen Sie für die Betriebsarten Einmalig Zählen und Periodisch Zählen fest, welche Zählgrenze als Startwert bzw. Endwert gilt.

Bei Hauptzählrichtung vorwärts zählt der Zähler von 0 bzw. vom Ladewert bis zum parametrierten Endwert.

Bei Hauptzählrichtung rückwärts beginnt der Zähler beim parametrierten Startwert und zählt bis 0.

Auch wenn Sie als Hauptzählrichtung "rückwärts" parametriert haben, müssen Sie zum Rückwärtszählen entweder ein entsprechendes Richtungssignal anlegen oder bei der Parametrierung "Zählrichtung invertiert" einstellen.

Torfunktion

Software-Tor und Hardware-Tor

Die CPU hat zwei logisch UND-verknüpfte Tore:

Ein **Software-Tor** (SW-Tor), das über das Anwenderprogramm gesteuert wird.

Das Software-Tor kann ausschließlich durch einen Flankenwechsel 0-1 des Eingangsparameters SW_GATE am SFB geöffnet werden. Es wird geschlossen durch Rücksetzen dieses Parameters.

Ein **Hardware-Tor** (HW-Tor), das über den Digitaleingang "Hardware-Tor" gesteuert wird. Sie stellen im Register "Zählen" ein, ob Sie das Hardware-Tor verwenden wollen. Es wird bei einem Flankenwechsel 0-1 am Digitaleingang geöffnet und bei einem Flankenwechsel 1-0 geschlossen.

Funktion Positionieren

Alarmauswahl

Für die Funktion "Positionieren" haben Sie die Möglichkeit Alarme auszuwählen.

Voraussetzung

- Die CPU ist selektiert.
- Die Alarmauswahl ist unter "Eigenschaften > Allgemein > Positionieren > Alarmauswahl" zu finden.

Alarmauswahl

Wählen Sie in der Klappliste "Alarmauswahl" aus, ob die CPU Diagnosealarme auslösen soll. Sie haben folgende Auswahlmöglichkeiten:

- Keine
- Diagnose

Als Voreinstellung ist "keine" ausgewählt.

Damit durch die entsprechenden Ereignisse auch Diagnosealarme ausgelöst werden, müssen Sie noch weitere Einstellungen vornehmen.

Aktivieren Sie die gewünschte Überwachung unter "Eigenschaften > Allgemein > Positionieren > Kanal 0" bei gewählter Betriebsart "Positionieren mit Analogausgang":

- Unter "Antrieb"
  Istwert, Zieleinlauf und Zielbereich können überwacht werden.
- Unter "Achse"
  Für eine Linearachse können Sie hier festlegen, ob der Arbeitsbereich überwacht wird. Der Verfahrbereich wird immer überwacht.
- Unter "Geber"
  Bei Gebern mit Nullmarke wird überprüft, ob der Geber zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Impulse liefert.

Welche Ereignisse einen Diagnosealarm auslösen sollen, wählen Sie unter "Kanal 0 - Diagnose" aus.

Für einen Diagnosealarm müssen Sie folgende Einstellungen parametrieren:

- Die Alarmauswahl unter "Eigenschaften > Allgemein > Positionieren > Kanal 0"
- Die entsprechende Überwachung im entsprechenden Abschnitt "Antrieb", "Achse" oder "Geber"
- Die entsprechende Freigabe im Abschnitt "Diagnose"

Betriebsart

Betriebsart

Mit dem Parameter "Betriebsart" legen Sie fest, wie Sie positionieren wollen. Sie haben folgende Optionen:

- Analogausgänge: Positionieren mit Analogausgang
- Digitalausgänge: Positionieren mit Digitalausgängen

Als Vorbelegung ist "nicht parametriert" ausgewählt.

Die Wahl der Betriebsart finden Sie unter "Eigenschaften > Allgemein > Positionieren > Kanal 0".

Hinweis

Die Technologie "Positionieren" können Sie nicht parametrieren, wenn Sie bei der Technologie Zählen Kanal 0 oder Kanal 1 parametriert haben.

Positionieren mit Analogausgang können Sie nur parametrieren, wenn Sie im Abschnitt AI5/AO2 den Ausgang 0 deaktiviert haben.

Die durch das Positionieren belegten Ein- und Ausgangskanäle können aus dem Anwenderprogramm heraus nicht mehr angesprochen werden.

Siehe auch

Dokumentation zu Technologischen Funktionen der CPU (http://support.automation.siemens.com/WW/view/de/12429336/0/de)

Positionieren mit Analogausgang

Antrieb

Ansteuerung des Antriebs und Ansteuerart

Ansteuerung des Antriebs und Ansteuerart

Unter "Eigenschaften > Positionieren > Kanal 0" im Abschnitt "Kanal 0 - Antrieb" legen Sie die Parameter für den Antrieb der Achse bei einer Positionierung mit Analogausgang fest.

Beschreibung des Antriebs

Die Ansteuerung des Antriebs erfolgt über zwei fest zugeordnete Analogausgänge entweder mit einer Spannung von ±10 V (Pin 16) oder mit einem Strom von ±20 mA (Pin 17).

Hinweis

Positionieren mit Analogausgang können Sie nur parametrieren, wenn Sie im Abschnitt AI5/AO2 der Bereichsnavigation den Ausgang 0 deaktiviert haben.

Beschreibung der Ansteuerart

Sie können in der Klappliste zwischen 2 Ansteuerarten wählen. Die Ansteuerart beschreibt, wie Ihr angeschlossenes Leistungsteil angesteuert wird.

- Spannung ±10 V bzw. Strom ±20 mA:

Bei Fahrten in Richtung Plus (vorwärts) wird eine positive Spannung bzw. Strom ausgegeben. Bei Fahrten in Richtung Minus (rückwärts) wird eine negative Spannung bzw. Strom ausgegeben.

- Spannung 0 bis 10 V bzw. Strom von 0 bis 20 mA und Richtungssignal:

Bei Fahrten in Richtung Plus (vorwärts) wird eine positive Spannung bzw. Strom ausgegeben und der Digitalausgang CONV_DIR ist ausgeschaltet.

Bei Fahrten in Richtung Minus (rückwärts) wird eine positive Spannung bzw. Strom ausgegeben und der Digitalausgang CONV_DIR ist eingeschaltet.

Als Voreinstellung ist "±10" gewählt.

Max. Frequenz

Wegerfassung

In dieser Klappliste können Sie die maximale Zählfrequenz der Wegerfassungssignale (Gebersignale A, B, N) in festen Stufen einstellen.

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Als Voreinstellung ist "60 kHz" eingetragen.

Begleitsignale

In dieser Klappliste können Sie die maximale Frequenz der Signale Längenmessung und Referenzpunktschalter in festen Stufen einstellen.

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Als Voreinstellung ist "10 kHz" eingetragen.

Zieleinlauf (Analog)

Zielbereich

Im Eingabefeld "Zielbereich" tragen Sie einen geradzahligen Wert zwischen 0 bis 2 x 10 Impulse ein. Ein ungerader Wert wird intern gerundet. Der Zielbereich liegt symmetrisch um das Ziel. Tragen Sie hier den Wert 0 ein, muss das Ziel impulsgenau erreicht werden. Die Überwachung des Zielbereichs ist dann abgeschaltet, das entsprechende Optionskästchen im Bereich "Überwachung" ist gegraut.

Als Voreinstellung ist "50 Impulse" eingetragen.

Überwachungszeit

Im Eingabefeld "Überwachungszeit" tragen Sie eine Zeit zwischen 0 und 100000 ms ein. Dieser Wert wird von der CPU gerundet auf 4 ms Bearbeitungszyklus.

Mit der Überwachungszeit überwacht die CPU den Istwert und den Zieleinlauf.

Tragen Sie hier den Wert 0 ein, werden die Überwachungen von Istwert und Zieleinlauf abgeschaltet, die entsprechenden Optionskästchen im Bereich "Überwachung" sind dann gegraut.

Als Voreinstellung ist "2000 ms" eingetragen.

Maximale Geschwindigkeit

Im Eingabefeld "Maximale Geschwindigkeit" tragen Sie einen Wert zwischen 10 und 1 000 000 Impulse/s ein, der dann dem Pegel von 10 V bzw. 20 mA am Analogausgang entspricht.

Als Voreinstellung ist "1000 Impulse/s" eingetragen.

Schleich-/Referenziergeschwindigkeit

Im Eingabefeld "Schleich-/Referenziergeschwindigkeit" tragen Sie einen Wert zwischen 10 Impulse/s und der parametrierten Maximalgeschwindigkeit ein.

Nach Erreichen des Bremseinsatzpunktes wird auf die Schleichgeschwindigkeit verzögert.

Bei der Referenzpunktfahrt wird nach Erreichen des Referenzpunktschalters auf die Referenziergeschwindigkeit verzögert.

Als Voreinstellung ist "100 Impulse/s" eingetragen.

Freigabe Leistungsteil

Abschaltverzögerung

Im Eingabefeld "Abschaltverzögerung" tragen Sie eine Zeit in Millisekunden ein. Dieser Wert wird von der CPU gerundet auf 4 ms Bearbeitungszyklus. Um diese Zeit verzögert wird bei Abbruch einer Fahrt (Wegnahme der Freigabe für das Leistungsteil) der Ausgang CONV_EN rückgesetzt.

Falls Sie über den Digitalausgang eine Bremse ansteuern, können Sie mit der Verzögerungszeit gewährleisten, dass die Achse zum Zeitpunkt des Rücksetzens des Ausgangs so langsam ist, dass die Bremse die Bewegungsenergie aufnehmen kann.

Als Voreinstellung sind "1000 ms" eingetragen.

Überwachung (Analog)

Beschreibung

Hier können Sie die Überwachung von Istwert, Zieleinlauf und Zielbereich ein- und ausschalten.

Zielbereich

Haben Sie das Optionskästchen "Zielbereich" angeklickt wird nach dem Erreichen des Zielbereichs überwacht, ob der Antrieb auf einer angefahrenen Zielposition stehenbleibt oder von ihr wegdriftet.

Haben Sie unter "Zielbereich" eine Überwachungszeit von 0 ms eingestellt, können Sie keine Überwachung des Zielbereichs einschalten, die entsprechenden Felder sind gegraut.

Nach Ansprechen der Überwachung wird ein Externfehler generiert. Den Externfehler müssen Sie mit ERR_A quittieren. Die Überwachung wird abgeschaltet. Erst mit dem Start einer neuen Fahrt wird die Überwachung wieder eingeschaltet.

Als Vorseinstellung ist diese Überwachung ausgeschaltet.

Zieleinlauf

Haben Sie das Optionskästchen "Zieleinlauf" angeklickt so wird überwacht, ob die Achse nach Erreichen des Abschaltpunktes innerhalb der Überwachungszeit den Zielbereich erreicht.

Haben Sie unter "Zieleinlauf" eine Überwachungszeit von 0 ms eingestellt, können Sie keine Überwachung des Zieleinlaufs einschalten, die entsprechenden Felder sind gegraut.

Als Vorseinstellung ist diese Überwachung ausgeschaltet.

Istwert

Haben Sie das Optionskästchen "Istwert" angeklickt so wird die Bewegung der Achse vom Start der Fahrt bis zum Erreichen des Abschaltpunktes überwacht. Wenn sich die Achse innerhalb der Überwachungszeit nicht um mindestens einen Impuls in vorgegebener Richtung bewegt, wird die Fahrt abgebrochen.

Wenn Sie unter "Zieleinlauf" eine Überwachungszeit von 0 ms eingestellt haben, können Sie keine Überwachung des Istwerts einschalten, die entsprechenden Felder sind gegraut.

Der Ausfall eines Digitaleinganges wird von der CPU nicht erkannt. Durch die Istwertüberwachung können Sie den Ausfall eines Gebersignales erkennen. Dieser Ausfall kann folgende Ursachen haben:

- Ausfall des Digitaleingangs
- Drahtbruch
- Defekt des Gebers
- Fehler am Leistungsteil

Als Voreinstellung ist diese Überwachung eingeschaltet.

Achse

Achsart (Analog)

Beschreibung

Unter "Eigenschaften > Positionieren > Kanal 0" im Abschnitt "Kanal 0 - Achse" legen Sie die Parameter für die Achse fest.

Achsart

Bei "Achsart" wählen Sie aus, ob Sie eine Linearachse oder eine Rundachse ansteuern.

Als Voreinstellung ist "Linearachse" ausgewählt.

Anfang und Ende des Software-Grenzschalters

Der Arbeitsbereich einer Linearachse liegt zwischen Anfang des Software-Grenzschalters und Ende des Software-Grenzschalters. Die Software-Grenzschalter gehören zum Arbeitsbereich. Geben Sie in die entsprechenden Eingabefelder Werte zwischen - 5 x 10^8 und + 5 x 10^8 Impulse ein. Der Anfang des Software-Grenzschalters muss immer kleiner sein als das Ende des Software-Grenzschalters.

Die Software-Grenzschalter werden überwacht, wenn die Achse synchronisiert ist und die Überwachung des Arbeitsbereichs eingeschaltet ist.

Als Voreinstellung sind "-100000000 Impulse" für den Anfang des Software-Grenzschalters und "100000000 Impulse" für das Ende des Software-Grenzschalters eingetragen.

Rundachsen-Ende

Geben Sie im Eingabefeld "Rundachsen-Ende" einen Wert zwischen 1 und 10^9 Impulse ein.

Der Wert "Rundachsen-Ende" ist der theoretisch größte Wert, den der Istwert erreichen kann. Er hat physikalisch die gleiche Position wie der Anfang der Rundachse (0).

Als Voreinstellung ist "100000 Impulse" eingetragen.

Längenmessung (Analog)

Beschreibung

Mit der Funktion "Längenmessung" können Sie während einer Fahrt Längen messen. Der Beginn und das Ende der Längenmessung erfolgen über Flanken am Digitaleingang "Längenmessung". In der Klappliste "Erkennung" wählen Sie die Art der Flanken aus.

Sie haben folgende Optionen:

- Aus
- Beginn/Ende mit steigender Flanke
  Eine steigende Flanke am Digitaleingang startet die Längenmessung, die nächste steigende Flanke beendet die Messung.
- Beginn/Ende mit fallender Flanke
  Eine fallende Flanke am Digitaleingang startet die Längenmessung, die nächste fallende Flanke beendet die Messung.
- Beginn mit steigender Flanke, Ende mit fallender Flanke
  Eine steigende Flanke am Digitaleingang startet die Längenmessung, die nächste fallende Flanke beendet die Messung.
- Beginn mit fallender Flanke, Ende mit steigender Flanke
  Eine fallende Flanke am Digitaleingang startet die Längenmessung, die nächste steigende Flanke beendet die Messung.

Als Voreinstellung ist "aus" eingetragen.

Referenzpunkt (Analog)

Koordinate

Nach einem STOP-RUN-Übergang der CPU wird bei noch nicht synchronisierter Achse der Istwert auf den Wert der Referenzpunktkoordinate gesetzt.

Nach einer Referenzpunktfahrt wird dem Referenzpunkt der Wert der Referenzpunktkoordinate zugeordnet.

Bei einer Linearachse muss der Wert der Referenzpunktkoordinate innerhalb des Arbeitsbereichs (einschließlich Anfang des Software-Grenzschalters und Endes des Software-Grenzschalters) liegen.

Bei einer Rundachse muss der Wert der Referenzpunktkoordinate im Bereich 0 bis "Rundachsenende - 1" liegen.

Als Voreinstellung ist "0 Impulse" eingetragen.

Referenzpunktlage zu Referenzpunktschalter

In der Klappliste "Referenzpunktlage zu Referenzpunktschalter" wählen Sie aus, ob der Referenzpunkt in Richtung Plus (Istwerte werden größer) oder in Richtung Minus (Istwerte werden kleiner) vom Referenzpunktschalter liegt.

Als Voreinstellung ist "in Richtung Plus" ausgewählt.

Überwachung Achse (Analog)

Arbeitsbereich

Für eine Linearachse können Sie hier festlegen, ob der Arbeitsbereich überwacht wird. Haben Sie das Optionskästchen "Arbeitsbereich" angeklickt, so wird überprüft, ob sich der Lageistwert außerhalb der Software-Grenzschalter befindet. Die Überwachung wirkt nur bei synchronisierter Achse.

Die Koordinaten der Software-Grenzschalter selbst gehören mit zum Arbeitsbereich.

Bei Ansprechen der Überwachung wird die Fahrt abgebrochen.

Als Voreinstellung ist diese Überwachung eingeschaltet.

Verfahrbereich

Mit der Verfahrbereichsüberwachung überprüfen Sie, ob der zulässige Verfahrbereich von $- 5 \times 10^8$ bis $+ 5 \times 10^8$ verlassen wird. Der Verfahrbereich wird immer überwacht. Die Überwachung ist nicht abschaltbar (Im Parameter "Überwachungen" immer eingeschaltet).

Bei Ansprechen der Überwachung wird die Fahrt abgebrochen.

Linearachse

Beschreibung

Die **Linearachse** ist eine Achse, die einen beschränkten physikalischen Verfahrbereich hat.

Der Arbeitsbereich auf der Linearachse wird durch die Software-Grenzschalter begrenzt.

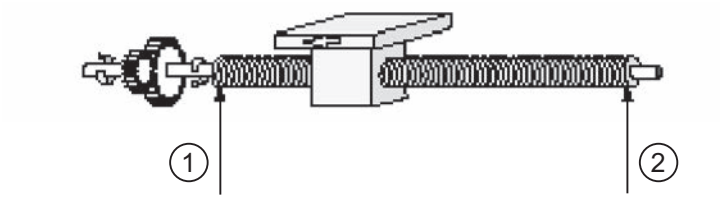

① Physikalischer Anfang
② Physikalisches Ende

Rundachse

Beschreibung

Die **Rundachse** ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

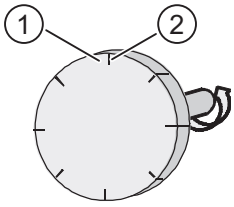

① Größter anzeigbarer Wert = Rundachsenende - 1
② Anfang der Rundachse (Koordinate 0) = Rundachsenende

Eine Umdrehung der Rundachse beginnt bei der Koordinate "Null" und endet bei der Koordinate "Ende der Rundachse". Physikalisch ist die Koordinate "Null" mit der Koordinate "Ende der Rundachse" identisch (= 0). An diesem Punkt springt die Anzeige des Lageistwertes um. Die Anzeige des Lageistwertes ist immer positiv.

Geber

Inkremente pro Geberumdrehung

Im Eingabefeld "Inkremente pro Geberumdrehung" tragen Sie ein, wie viele Inkremente Ihr Geber bei einer Vollumdrehung ausgibt. Diesen Wert können Sie der Beschreibung Ihres Gebers entnehmen.

Die CPU wertet die Inkremente 4-fach aus, d. h. ein Inkrement entspricht vier Impulsen.

Als Voreinstellung ist "1000 Inkremente" eingetragen.

Zählrichtung

Bei "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an.

Klicken Sie in der Klappliste "Zählrichtung" die Option "Normal" aus, wenn aufsteigende Zählimpulse aufsteigenden Lageistwerten entsprechen.

Klicken Sie in der Klappliste "Zählrichtung" die Option "Invertiert" aua, wenn aufsteigende Zählimpulse absteigenden Lageistwerten entsprechen.

Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

Als Voreinstellung ist die Option "Normal" angewählt.

Überwachung

Bei "Überwachung" können Sie festlegen, ob der Geber auf Fehlimpulse überwacht werden soll. Dies ist nur bei Gebern möglich, deren Wert für "Inkremente pro Geberumdrehung" durch 10 oder 16 teilbar ist.

Wenn Sie das Optionskästchen "Fehlimpuls" anklicken so wird überprüft, ob der Geber zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Impulse liefert.

Wenn Sie einen Geber ohne Nullmarke verwenden, müssen Sie die Überwachung der Fehlimpulse abschalten.

Als Voreinstellung ist diese Überwachung ausgeschaltet.

Diagnose

Beschreibung

Unter "Eigenschaften > Positionieren > Kanal 0" im Abschnitt "Kanal 0 - Diagnose" wählen Sie aus, bei welchen Ereignissen ein Diagnosealarm generiert werden soll. Als Voreinstellung ist kein Optionskästchen ausgewählt.

Fehlimpuls

Haben Sie das Optionskästchen "Fehlimpuls" angeklickt, so wird ein Diagnosealarm generiert, sobald der Geber zwischen zwei aufeinanderfolgenden Nullmarken nicht dieselbe Anzahl Impulse liefert. Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Geber" einschalten.

Verfahrbereich

Haben Sie das Optionskästchen "Verfahrbereich" angeklickt, so wird ein Diagnosealarm generiert, sobald die Achse den Verfahrbereich verlässt. Die entsprechende Überwachung ist immer eingeschaltet.

Arbeitsbereich

Haben Sie das Optionskästchen "Arbeitsbereich" angeklickt, so wird ein Diagnosealarm generiert, wenn sich der Lageistwert außerhalb der Softwareendschalter befindet. Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Achse" einschalten.

Bei Positionierung auf einer Rundachse können Sie diesen Diagnosealarm nicht verwenden. Das entsprechende Optionskästchen ist gegraut.

Istwert

Haben Sie das Optionskästchen "Istwert" angeklickt, so wird ein Diagnosealarm generiert, wenn sich während einer Verfahrbewegung die Achse innerhalb der Überwachungszeit nicht um mindestens einen Impuls in vorgegebener Richtung bewegt. Die Fahrt wird abgebrochen.

Der Istwert wird vom Start der Fahrt bis zum Erreichen des Abschaltpunktes überwacht.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Der Ausfall eines Digitaleinganges wird von der CPU nicht erkannt. Durch Einschalten der Istwertüberwachung können Sie den Ausfall eines Gebersignales erkennen.

Als Voreinstellung ist dieses Optionskästchen angewählt.

Zieleinlauf

Haben Sie das Optionskästchen "Zieleinlauf" angeklickt, so wird ein Diagnosealarm generiert, wenn die Achse nach Erreichen des Abschaltpunktes innerhalb der Überwachungszeit den Zielbereich nicht erreicht.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Zielbereich

Haben Sie das Optionskästchen "Zielbereich" angeklickt, so wird ein Diagnosealarm generiert, wenn der Antrieb nach dem Erreichen des Zielbereichs nicht auf einer angefahrenen Zielposition stehen bleibt oder von der angefahrenen Zielposition wegdriftet.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Positionieren mit Digitalausgängen

Antrieb

Ansteuerung des Antriebs

Unter "Eigenschaften > Positionieren > Kanal 0" im Abschnitt "Kanal 0 - Antrieb" legen Sie die Parameter für den Antrieb der Achse bei einer Positionierung mit Digitalausgängen fest.

Beschreibung

Die Ansteuerung des Antriebs erfolgt über vier fest zugeordnete 24-V-Digitalausgänge

Sie können in der Klappliste zwischen 4 Ansteuerarten wählen. Die Ansteuerart beschreibt, wie die 4 Digitalausgänge (DO + 1.0 bis DO + 1.3) das Leistungsteil ansteuern.

Nachfolgend ist die Funktion der einzelnen Ausgänge bei den verschiedenen Ansteuerarten aufgelistet

Ausgang	Ansteuerart 1	Ansteuerart 2	Ansteuerart 3	Ansteuerart 4
DO + 1.0	Eilgang	Eil-/Schleichgang	Eilgang	Eilgang plus
DO + 1.1	Schleichgang	Position erreicht	Schleichgang	Schleichgang plus
DO + 1.2	Fahren plus	Fahren plus	Fahren plus	Eilgang minus
DO + 1.3	Fahren minus	Fahren minus	Fahren minus	Schleichgang minus

Max. Frequenz

Wegerfassung

In dieser Klappliste können Sie die maximale Zählfrequenz der Wegerfassungssignale (Gebersignale A, B, N) in festen Stufen einstellen.

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Als Voreinstellung ist "60 kHz" eingetragen.

Begleitsignale

In dieser Klappliste können Sie die maximale Frequenz der Signale Längenmessung und Referenzpunktschalter in festen Stufen einstellen.

Falls Ihre CPU diese Funktion nicht unterstützt, ist die Klappliste gegraut.

Als Voreinstellung ist "10 kHz" eingetragen.

Zieleinlauf (Digital)

Zielbereich

Im Eingabefeld "Zielbereich" tragen Sie einen geradzahligen Wert zwischen 0 bis 2×10^8 Impulse ein. Ein ungerader Wert wird intern gerundet. Der Zielbereich liegt symmetrisch um das Ziel. Tragen Sie hier den Wert 0 ein, muss das Ziel impulsgenau erreicht werden. Die Überwachung des Zielbereichs ist dann abgeschaltet, das entsprechende Optionskästchen im Bereich "Überwachung" ist dann gegraut.

Als Voreinstellung sind "50 Impulse" eingetragen.

Überwachungszeit

Im Eingabefeld "Überwachungszeit" tragen Sie eine Zeit zwischen 0 und 100000 ms ein. Dieser Wert wird von der CPU gerundet auf 4 ms Bearbeitungszyklus.

Mit der Überwachungszeit überwacht die CPU den Istwert und den Zieleinlauf.

Tragen Sie hier den Wert 0 ein, werden die Überwachungen von Istwert und Zieleinlauf abgeschaltet, die entsprechenden Optionskästchen im Bereich "Überwachung" sind dann gegraut.

Als Voreinstellung sind "2000 ms" eingetragen.

Überwachung (Digital)

Beschreibung

Hier können Sie die Überwachung von Istwert, Zieleinlauf und Zielbereich ein- und ausschalten.

Zieleinlauf

Haben Sie das Optionskästchen "Zieleinlauf" angeklickt wird überwacht, ob die Achse nach Erreichen des Abschaltpunktes innerhalb der Überwachungszeit den Zielbereich erreicht.

Haben Sie unter "Zieleinlauf" eine Überwachungszeit von 0 ms eingestellt, können Sie keine Überwachung von Istwert und Zieleinlauf einschalten, die entsprechenden Felder sind gegraut.

Als Voreinstellung ist diese Überwachung ausgeschaltet.

Zielbereich

Haben Sie das Optionskästchen "Zielbereich" angeklickt wird nach dem Erreichen des Zielbereichs überwacht, ob der Antrieb auf einer angefahrenen Zielposition stehenbleibt oder von ihr wegdriftet.

Nach Ansprechen der Überwachung wird ein Externfehler generiert. Den Externfehler müssen Sie mit ERR_A quittieren. Die Überwachung wird abgeschaltet. Erst mit dem Start einer neuen Fahrt wird die Überwachung wieder eingeschaltet.

Als Voreinstellung ist diese Überwachung ausgeschaltet.

Istwert

Haben Sie das Optionskästchen "Istwert" angeklickt so wird die Bewegung der Achse vom Start der Fahrt bis zum Erreichen des Abschaltpunktes überwacht. Wenn sich die Achse innerhalb der Überwachungszeit nicht um mindestens einen Impuls in vorgegebener Richtung bewegt, wird die Fahrt abgebrochen.

Wenn Sie unter "Zieleinlauf" eine Überwachungszeit von 0 ms eingestellt haben, können Sie keine Überwachung von Istwert und Zieleinlauf einschalten, die entsprechenden Felder sind gegraut.

Der Ausfall eines Digitaleinganges wird von der CPU nicht erkannt. Durch die Istwertüberwachung können Sie den Ausfall eines Gebersignales erkennen. Dieser Ausfall kann folgende Ursachen haben:

- Ausfall des Digitaleingangs
- Drahtbruch

- Defekt des Gebers
- Fehler am Leistungsteil

Als Voreinstellung ist diese Überwachung eingeschaltet.

Achse

Achsart (Digital)

Beschreibung

Bei "Achsart" wählen Sie aus, ob Sie eine Linearachse oder eine Rundachse ansteuern.

Als Voreinstellung ist "Linearachse" ausgewählt.

Anfang und Ende des Software-Grenzschalters

Der Arbeitsbereich einer Linearachse liegt zwischen Anfang des Software-Grenzschalters und Ende des Software-Grenzschalters. Die Software-Grenzschalter gehören zum Arbeitsbereich. Geben Sie in die entsprechenden Eingabefelder Werte zwischen -5×10^8 und $+5 \times 10^8$ Impulse ein. Der Anfang des Software-Grenzschalters muss immer kleiner sein als das Ende des Software-Grenzschalters.

Die Software-Grenzschalter werden überwacht, wenn die Achse synchronisiert ist und die Überwachung des Arbeitsbereichs eingeschaltet ist.

Als Voreinstellung sind "-100000000 Impulse" für den Anfang des Software-Grenzschalters und "100000000 Impulse" für das Ende des Software-Grenzschalters eingetragen.

Rundachsen-Ende

Geben Sie im Eingabefeld "Rundachsen-Ende" einen Wert zwischen 1 und 10^9 Impulse ein.

Der Wert "Rundachsen-Ende" ist der theoretisch größte Wert, den der Istwert erreichen kann. Er hat physikalisch die gleiche Position wie der Anfang der Rundachse (0).

Als Voreinstellung ist "100000 Impulse" eingetragen.

Längenmessung (Digital)

Beschreibung

Mit der Funktion "Längenmessung" können Sie während einer Fahrt Längen messen. Der Beginn und das Ende der Längenmessung erfolgen über Flanken am Digitaleingang "Längenmessung". In der Klappliste "Erkennung" wählen Sie die Art der Flanken aus.

Sie haben folgende Optionen:

- Aus
- Beginn/Ende mit steigender Flanke
  Eine steigende Flanke am Digitaleingang startet die Längenmessung, die nächste steigende Flanke beendet die Messung.
- Beginn/Ende mit fallender Flanke
  Eine fallende Flanke am Digitaleingang startet die Längenmessung, die nächste fallende Flanke beendet die Messung.
- Beginn mit steigender Flanke, Ende mit fallender Flanke
  Eine steigende Flanke am Digitaleingang startet die Längenmessung, die nächste fallende Flanke beendet die Messung.
- Beginn mit fallender Flanke, Ende mit steigender Flanke
  Eine fallende Flanke am Digitaleingang startet die Längenmessung, die nächste steigende Flanke beendet die Messung.

Als Voreinstellung ist "aus" eingetragen.

Referenzpunkt (Digital)

Koordinate

Nach einem STOP-RUN-Übergang der CPU wird bei noch nicht synchronisierter Achse der Istwert auf den Wert der Referenzpunktkoordinate gesetzt.

Nach einer Referenzpunktfahrt wird dem Referenzpunkt der Wert der Referenzpunktkoordinate zugeordnet.

Bei einer Linearachse muss der Wert der Referenzpunktkoordinate innerhalb des Arbeitsbereichs (einschließlich Anfang des Software-Grenzschalters und Ende des Software-Grenzschalters) liegen.

Bei einer Rundachse muss der Wert der Referenzpunktkoordinate im Bereich 0 bis "Rundachsenende - 1" liegen.

Als Voreinstellung ist "0 Impulse" eingetragen.

Referenzpunktlage zu Referenzpunktschalter

In der Klappliste "Referenzpunktlage zu Referenzpunktschalter" wählen Sie aus, ob der Referenzpunkt in Richtung Plus (Istwerte werden größer) oder in Richtung Minus (Istwerte werden kleiner) vom Referenzpunktschalter liegt.

Als Voreinstellung ist "in Richtung Plus" ausgewählt.

Überwachung Achse (Digital)

Arbeitsbereich

Für eine Linearachse können Sie hier festlegen, ob der Arbeitsbereich überwacht wird. Haben Sie das Optionskästchen "Arbeitsbereich" angeklickt, so wird überprüft, ob sich der Lageistwert außerhalb der Software-Grenzschalter befindet. Die Überwachung wirkt nur bei synchronisierter Achse.

Die Koordinaten der Software-Grenzschalter selbst gehören mit zum Arbeitsbereich.

Bei Ansprechen der Überwachung wird die Fahrt abgebrochen.

Als Voreinstellung ist diese Überwachung eingeschaltet.

Verfahrbereich

Mit der Verfahrbereichsüberwachung überprüfen Sie, ob der zulässige Verfahrbereich von $- 5 \times 10^8$ bis $+ 5 \times 10^8$ verlassen wird. Der Verfahrbereich wird immer überwacht. Die Überwachung ist nicht abschaltbar (Im Parameter "Überwachungen" immer eingeschaltet).

Bei Ansprechen der Überwachung wird die Fahrt abgebrochen.

Linearachse

Beschreibung

Die **Linearachse** ist eine Achse, die einen beschränkten physikalischen Verfahrbereich hat.

Der Arbeitsbereich auf der Linearachse wird durch die Software-Grenzschalter begrenzt.

① Physikalischer Anfang
② Physikalisches Ende

Rundachse

Beschreibung

Die **Rundachse** ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

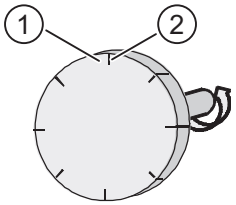

① Größter anzeigbarer Wert = Rundachsenende - 1

② Anfang der Rundachse (Koordinate 0) = Rundachsenende

Eine Umdrehung der Rundachse beginnt bei der Koordinate "Null" und endet bei der Koordinate "Ende der Rundachse". Physikalisch ist die Koordinate "Null" mit der Koordinate "Ende der Rundachse" identisch (= 0). An diesem Punkt springt die Anzeige des Lageistwertes um. Die Anzeige des Lageistwertes ist immer positiv.

Geber

Inkremente pro Geberumdrehung

Im Eingabefeld "Inkremente pro Geberumdrehung" tragen Sie ein, wie viele Inkremente Ihr Geber bei einer Vollumdrehung ausgibt. Diesen Wert können Sie der Beschreibung Ihres Gebers entnehmen.

Die CPU wertet die Inkremente 4-fach aus, d. h. ein Inkrement entspricht vier Impulsen.

Als Voreinstellung ist "1000 Inkremente" eingetragen.

Zählrichtung

Bei "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an.

Klicken Sie in der Klappliste "Zählrichtung" die Option "Normal" aus, wenn aufsteigende Zählimpulse aufsteigenden Lageistwerten entsprechen.

Klicken Sie in der Klappliste "Zählrichtung" die Option "Invertiert" aua, wenn aufsteigende Zählimpulse absteigenden Lageistwerten entsprechen.

Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

Als Voreinstellung ist die Option "Normal" angewählt.

Überwachung

Bei "Überwachung" können Sie festlegen, ob der Geber auf Fehlimpulse überwacht werden soll. Dies ist nur bei Gebern möglich, deren Wert für "Inkremente pro Geberumdrehung" durch 10 oder 16 teilbar ist.

Wenn Sie das Optionskästchen "Fehlimpuls" anklicken so wird überprüft, ob der Geber zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Impulse liefert.

Wenn Sie einen Geber ohne Nullmarke verwenden, müssen Sie die Überwachung der Fehlimpulse abschalten.

Als Voreinstellung ist diese Überwachung ausgeschaltet.

Diagnose

Beschreibung

Unter "Eigenschaften > Positionieren > Kanal 0" im Abschnitt "Kanal 0 - Diagnose" wählen Sie aus, bei welchen Ereignissen ein Diagnosealarm generiert werden soll. Als Voreinstellung ist kein Optionskästchen ausgewählt.

Fehlimpuls

Haben Sie das Optionskästchen "Fehlimpuls" angeklickt, so wird ein Diagnosealarm generiert, sobald der Geber zwischen zwei aufeinanderfolgenden Nullmarken nicht dieselbe Anzahl Impulse liefert. Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Geber" einschalten.

Verfahrbereich

Haben Sie das Optionskästchen "Verfahrbereich" angeklickt, so wird ein Diagnosealarm generiert, sobald die Achse den Verfahrbereich verlässt. Die entsprechende Überwachung ist immer eingeschaltet.

Arbeitsbereich

Haben Sie das Optionskästchen "Arbeitsbereich" angeklickt, so wird ein Diagnosealarm generiert, wenn sich der Lageistwert außerhalb der Softwareendschalter befindet. Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Achse" einschalten.

Bei Positionierung auf einer Rundachse können Sie diesen Diagnosealarm nicht verwenden. Das entsprechende Optionskästchen ist gegraut.

Istwert

Haben Sie das Optionskästchen "Istwert" angeklickt, so wird ein Diagnosealarm generiert, wenn sich während einer Verfahrbewegung die Achse innerhalb der Überwachungszeit nicht um mindestens einen Impuls in vorgegebener Richtung bewegt. Die Fahrt wird abgebrochen.

Der Istwert wird vom Start der Fahrt bis zum Erreichen des Abschaltpunktes überwacht.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Der Ausfall eines Digitaleinganges wird von der CPU nicht erkannt. Durch Einschalten der Istwertüberwachung können Sie den Ausfall eines Gebersignales erkennen.

Als Voreinstellung ist dieses Optionskästchen angewählt.

Zieleinlauf

Haben Sie das Optionskästchen "Zieleinlauf" angeklickt, so wird ein Diagnosealarm generiert, wenn die Achse nach Erreichen des Abschaltpunktes innerhalb der Überwachungszeit den Zielbereich nicht erreicht.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Zielbereich

Haben Sie das Optionskästchen "Zielbereich" angeklickt, so wird ein Diagnosealarm generiert, wenn der Antrieb nach dem Erreichen des Zielbereichs nicht auf einer angefahrenen Zielposition stehen bleibt oder von der angefahrenen Zielposition wegdriftet.

Sie müssen hierzu die entsprechende Überwachung im Abschnitt "Kanal 0 - Antrieb" einschalten.

Funktion Punkt-zu-Punkt-Kopplung

ASCII-Protokoll

PtP - Übertragung (ASCII)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "ASCII" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "ASCII" gibt es 4 Parametergruppen Übertragung, Endezeichen, Datenempfang und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Übertragung und ist in 3 Abschnitte unterteilt.

Übertragungsgeschwindigkeit

In der Klappliste "Übertragungsgeschwindigkeit" wählen Sie die Geschwindigkeit der Datenübertragung in bit/s (Baud) aus. Sie können einen Wert zwischen 300 und 38400 Baud auswählen. Bei Vollduplexbetrieb beträgt die maximale Übertragungsgeschwindigkeit 19200 Baud.

Als Voreinstellung ist "9600 Baud" eingestellt.

Zeichenrahmen

Die Daten werden über die serielle Schnittstelle in einem Zeichenrahmen übertragen. Für jeden Zeichenrahmen stehen zwei Datenformate zur Verfügung.

- Datenbits
  In der Klappliste "Datenbits" wählen Sie die Anzahl der Bits aus, auf die ein Zeichen abgebildet wird.
  Als Voreinstellung ist "8 bit" ausgewählt.
- Stoppbits
  In der Klappliste "Stoppbits" wählen Sie die Anzahl der Bits aus, die bei der Übertragung jedem zu übertragenden Zeichen nachgesetzt werden. Die Stoppbits kennzeichnen das Ende eines Zeichens.
  Als Voreinstellung ist "1 bit" ausgewählt.
- Parität
  In der Klappliste "Parität" wählen Sie aus, ob die Sicherheit der Datenübertragung durch die Verwendung eines Paritätsbits erhöht werden soll. Parität "keine" bedeutet, dass kein Paritätsbit gesendet wird. **Bei Einstellung von 7 Datenbits ist die Einstellung "keine" nicht möglich.**
  Eine Folge von Informationsbits kann um ein weiteres Bit, das Paritätsbit, erweitert werden, das durch seinen addierten Wert ("0" oder "1") den Wert aller Bits auf einen vereinbarten Zustand - gerade oder ungerade - ergänzt.
  Als Voreinstellung ist "gerade" ausgewählt.

Übertragung Datenflusskontrolle

- Datenflusskontrolle
  Hier können Sie nur dann Parameter festlegen, wenn Sie als Datenflusskontrolle "XON/XOFF" gewählt haben.
  Als Voreinstellung ist "keine" ausgewählt.
- XON-Zeichen
  Im Eingabefeld "XON-Zeichen" geben Sie einen Code für das Zeichen XON ein. Geben Sie den Code Hexadezimal codiert ein.
  Als Voreinstellung ist "11" eingetragen.

- XOFF-Zeichen
  Im Eingabefeld "XOFF-Zeichen" geben Sie einen Code für das Zeichen XOFF ein. Geben Sie den Code Hexadezimal codiert ein.
  Welche Werte Sie eingeben können hängt davon ab, ob Sie Ihre Daten mit 7 oder mit 8 Datenbits übertragen:
  - bei 7 Datenbits: 0 bis 7FH
  - bei 8 Datenbits: 0 bis FFH
    Als Voreinstellung ist "13" eingetragen.
- Warten auf XON nach XOFF
  Im Eingabefeld "Warten auf XON nach XOFF" geben Sie die Zeit an, die die Baugruppe beim Senden auf das Zeichen XON des Kommunikationspartners warten soll.
  Sie können eine Zeit zwischen 20 und 65530 ms - in 10 ms Schritten - eingeben.
  Als Voreinstellung ist "20000 ms" eingetragen.

PtP - Endezeichen (ASCII)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "ASCII" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "ASCII" gibt es 4 PtP-Bereiche Übertragung, Endezeichen, Datenempfang und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit dem PtP-Endezeichen und ist in 2 Abschnitte unterteilt.

Erkennung eines Empfangstelegramms

Wählen Sie in der Klappliste "Endeerkennung eines Empfangstelegramms" die gewünschte Option aus. Es stehen drei Optionen zur Verfügung "Nach Ablauf der Zeichenverzugszeit", "Durch Endezeichen" und "Nach Empfang einer festen Zeichenzahl".

- Nach Ablauf der Zeichenverzugszeit
  Durch die Wahl "Nach Ablauf der Zeichenverzugszeit" aus der Klappliste legen Sie fest, dass nach Ablauf einer parametrierten Zeitspanne das Ende eines Telegramms erkannt wird. Die Zeichenverzugszeit definiert dann den maximal zulässigen Abstand zwischen 2 hintereinander empfangenen Zeichen.
  Als Voreinstellung ist diese Option aktiviert.
- Zeichenverzugszeit
  Im Eingabefeld "Zeichenverzugszeit" tragen Sie diese Zeitspanne in Millisekunden zwischen 1 und 65535 ein.
  Die kleinste Zeichenverzugszeit, die Sie wählen können, ist abhängig von der Übertragungsgeschwindigkeit.

Übertragungsgeschwindigkeit	Kleinste Zeichenverzugszeit
300 bit/s	130 ms
600 bit/s	65 ms
1200 bit/s	32 ms
2400 bit/s	16 ms
4800 bit/s	8 ms
9600 bit/s	4 ms
19200 bit/s	2 ms
38400 bit/s	1 ms

Als Voreinstellung sind "4 ms" eingestellt.

- Durch Endezeichen
  Durch die Wahl "Durch Endezeichen" aus der Klappliste legen Sie fest, dass durch das Endezeichen das Ende eines Telegramms erkannt wird.
- Überwachungszeit bei fehlender Endekennung
  Im Eingabefeld "Überwachungszeit bei fehlender Endekennung" finden Sie eine ggf. parametrierte Zeichenverzugszeit die als Überwachungszeit auf fehlende Endekennung verwendet wird. Dies gilt bei folgenden Einstellungen für die Endekennung:
  - nach Empfang einer festen Zeichenanzahl
  - nach Empfang der/des Endezeichen(s)
    Sie können die Überwachungszeit hier auch ändern. Es gelten dieselben Randbedingungen, die auch für die Zeichenverzugszeit gelten.
    Die Berechnung beim Sender und die Auswertung des Blockprüfzeichens beim Empfänger müssen Sie im Anwenderprogramm selbst vornehmen.
    Als Voreinstellung ist "4 ms" eingetragen.
- Endezeichen
  Mit der Einstellung "Durch Endezeichen" des Parameters "Endeerkennung eines Empfangstelegramms" legen Sie fest, dass das Ende eines Telegramms mit einem oder zwei definierten Endezeichen gekennzeichnet ist. Sie können mit einem oder mit zwei Endezeichen arbeiten. Wahlweise werden zusätzlich ein oder zwei Zeichen nach dem Endekennzeichen empfangen. Diese Zeichen können Sie z. B. verwenden, um ein Blockprüfzeichen (BCC) mit zu senden.
  Wählen Sie aus der zugehörigen Klappliste eine der folgenden Möglichkeiten aus:
  - 1. Zeichen
  - 1. Zeichen mit 1 BCC
  - 1. Zeichen mit 2 BCC
  - 1. UND 2. Zeichen
  - 1. UND 2. Zeichen mit 1 BCC
  - 1. UND 2. Zeichen mit 2 BCC
    Wenn Sie diese Option gewählt haben, ist als Voreinstellung "1. Zeichen" angewählt.

- Endezeichen Empfänger
  Geben Sie im Eingabefeld "1. Endezeichen Empfänger" und ggf. im Eingabefeld "2. Endezeichen Empfänger" jeweils einen Code für die entsprechende Endekennung ein. Welche Werte Sie eingeben können hängt davon ab, ob Sie Ihre Daten mit 7 oder mit 8 Datenbits übertragen:
  - bei 7 Datenbits: 0 bis 7FH
  - bei 8 Datenbits: 0 bis FFH
    Als Voreinstellung ist für das erste Endezeichen "3" und für das zweite Endezeichen "0" eingetragen.
- Nach Empfang einer festen Zeichenanzahl
  Durch die Wahl "Nach Empfang einer festen Zeichenanzahl" aus der Klappliste legen Sie fest, dass nach Empfang einer festen Zeichenanzahl das Ende eines Telegramms erkannt wird. Die Länge der Empfangstelegramme ist dann immer gleich.
  Als Voreinstellung ist diese Option nicht aktiviert.
- Sendepause zwischen den Telegrammen in Höhe der Überwachungszeit
  Haben Sie das Optionskästchen "Sendepause zwischen den Telegrammen in Höhe der Überwachungszeit" angeklickt, dann wird beim Senden zwischen zwei Telegrammen eine Pause in der Länge der Überwachungszeit (auf fehlende Endekennung) eingehalten, damit der Partner den Empfang des Telegramms erkennt und sich so synchronisieren kann.
  Als Voreinstellung ist dieses Optionskästchen nicht aktiviert.
- Telegrammlänge
  Im Eingabefeld "Telegrammlänge" stellen Sie die Anzahl der Bytes, aus denen ein Telegramm bestehen soll, ein. Sie können einen Wert zwischen 1 und 1024 Byte einstellen.
  Als Voreinstellung ist "200 Byte" eingestellt.

Senden mit Endezeichen

Nachdem Sie ein Endezeichen als Endekennung eines Empfangstelegramms festgelegt haben können Sie mit Endezeichen senden.

Als Voreinstellung ist "Senden bis einschließlich Endezeichen" eingestellt.

- Senden bis einschließlich Endezeichen
  Mit der Klappliste "Endekriterium > Senden bis einschließlich Endezeichen" legen Sie fest, dass das Endekennzeichen in den zu sendenden Daten enthalten sein muss. Es werden die Daten nur bis zum Endekennzeichen einschließlich gesendet, auch wenn am SFB eine größere Datenlänge angegeben ist.
- Senden bis zur am Baustein parametrierten Länge
  Mit der Klappliste "Endekriterium > Senden bis zur am Baustein parametrierten Länge" legen Sie fest, dass die Daten bis zu der am SFB parametrierten Länge gesendet werden. Das letzte Zeichen muss das Endezeichen sein.

- Senden mit Automatischem Anhängen der Endezeichen
  Mit der Klappliste "Endekriterium > Automatisches Anhängen der Endezeichen" legen Sie fest, dass die Daten bis zu der am SFB parametrierten Länge gesendet werden. Zusätzlich werden das/die Endezeichen automatisch angehängt; d. h. die Endekennzeichen dürfen nicht in den zu sendenden Daten enthalten sein. Je nach Anzahl der Endekennzeichen werden 1 oder 2 Zeichen mehr zum Partner geschickt, als vom SFB angegeben (maximal 1024 Byte).
- Endezeichen Sender
  Wenn Sie ein Endezeichen verwenden, können Sie jetzt seinen HEX-Code festlegen. Ebenso können Sie auswählen, ob Sie auch ein zweites Endezeichen verwenden wollen und Sie können dessen HEX-Code festlegen. Geben Sie im Eingabefeld "1. Endezeichen Sender" und ggf. im Eingabefeld "2. Endezeichen Sender" jeweils einen Code für die entsprechende Endekennung ein.
  Als Voreinstellung ist für das erste Endezeichen "3" und für das zweite Endezeichen "0" eingetragen.

PtP - Datenempfang (ASCII)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "ASCII" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "ASCII" gibt es 4 PtP-Bereiche Übertragung, Endezeichen, Datenempfang und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit dem PtP-Datenempfang.

Empfangspufferung auf CPU

- CPU-Empfangspuffer im Anlauf löschen
  Klicken Sie auf das Optionskästchen "CPU-Empfangspuffer im Anlauf löschen" dann wird bei Netz-Ein bzw. beim Übergang der CPU von STOP nach RUN der Empfangspuffer der technologischen Funktion gelöscht.
  Als Voreinstellung ist dieses Optionskästchen nicht aktiviert.
- Überschreiben verhindern
  Klicken Sie auf das Optionskästchen "Überschreiben verhindern" damit bei einem vollen Empfangspuffer Daten im Empfangspuffer nicht überschrieben werden.
  Als Voreinstellung ist dieses Optionskästchen aktiviert.

- Gesamten Puffer ausnutzen
  Klicken Sie auf das Optionskästchen "Gesamten Puffer ausnutzen" damit Sie den gesamten Empfangspuffer von 2048 Byte nutzen können. Wenn Sie den gesamten Puffer nutzen, ist die Anzahl der gepufferten Empfangstelegramme nur von der Länge der Telegramme abhängig.
  Als Voreinstellung ist dieses Optionskästchen aktiviert.
- Anzahl maximal gepufferter Empfangstelegramme
  Im Eingabefeld "Anzahl maximal gepufferter Empfangstelegramme" können Sie die Anzahl der Empfangstelegramme angeben, die im Empfangspuffer der technologischen Funktion gepuffert werden sollen. Dazu müssen Sie das Optionskästchen "Gesamten Puffer ausnutzen" deaktivieren. Sie können einen Wert zwischen 1 und 10 angeben.
  Als Voreinstellung ist der Wert "10" eingetragen.

Hinweis

Wenn Sie hier "1" parametrieren und den Parameter "Überschreiben verhindern" deaktivieren **und** die Empfangsdaten im Anwenderprogramm zyklisch auslesen, wird immer ein aktuelles Telegramm zur CPU übertragen.

PtP - Signalbelegung (ASCII)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "ASCII" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "ASCII" gibt es 4 PtP-Bereiche Übertragung, Endezeichen, Datenempfang und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Signalbelegung und ist in 2 Abschnitte unterteilt.

Betriebsart

Wählen Sie eine der drei Betriebsarten aus.

Als Voreinstellung ist "Vollduplex (RS 422) Vierdraht-Betrieb Punkt-zu-Punkt" ausgewählt.

Vollduplex (RS 422) Vierdraht-Betrieb Punkt-zu-Punkt

Mit dem runden Optionsfeld "Vollduplex (RS 422) Vierdraht-Betrieb Punkt-zu-Punkt" wählen Sie die geeignete Betriebsart für Punkt-zu-Punkt-Kopplungen im Vierdraht-Betrieb aus. Als Vorbelegung der Empfangsleitung sind dann folgende Varianten möglich:

- Keine
- Signal R(A) 5 Volt, Signal R(B) 0 Volt
- Signal R(A) 0 Volt, Signal R(B) 5 Volt

Vollduplex (RS 422) Vierdraht-Betrieb Mehrpunkt Master

Mit dem runden Optionsfeld "Vollduplex (RS 422) Vierdraht-Betrieb Mehrpunkt Master" wählen Sie die geeignete Betriebsart für mehrpunktfähige (Multipoint) Kopplung im Vierdraht-Betrieb, wenn die CPU Master ist. Als Vorbelegung der Empfangsleitung ist dann folgende Variante möglich:

- Signal R(A) 0 Volt, Signal R(B) 5 Volt

Halbduplex (RS 485) Zweidraht-Betrieb

Mit dem runden Optionsfeld "Halbduplex (RS 485) Zweidraht-Betrieb" wählen Sie die geeignete Betriebsart für Punkt-zu-Punkt oder mehrpunktfähige (Multipoint) Kopplung im Zweidraht-Betrieb. Die CPU kann Master oder Slave sein. Als Vorbelegung der Empfangsleitung sind dann folgende Varianten möglich:

- Keine
- Signal R(A) 0 Volt, Signal R(B) 5 Volt

Vorbelegung der Empfangsleitung

- Keine
  Mit dem runden Optionsfeld "Keine" wählen Sie eine Einstellung, die nur sinnvoll ist für busfähige Sondertreiber.
- Signal R(A) 5 Volt, Signal R(B) 0 Volt
  Mit dem runden Optionsfeld "Signal R(A) 5 Volt, Signal R(B) 0 Volt" wählen Sie eine Einstellung bei der Breakerkennung möglich ist. (Nicht einstellbar bei Vollduplex (RS422) Vierdraht-Betrieb Mehrpunkt Master und Halbduplex (RS485) Zweidraht-Betrieb)
- Signal R(A) 0 Volt, Signal R(B) 5 Volt
- Mit dem runden Optionsfeld "Signal R(A) 0 Volt, Signal R(B) 5 Volt" wählen Sie eine Einstellung, die dem Ruhezustand entspricht (kein Sender aktiv). Bei dieser Einstellung ist keine Breakerkennung möglich.
- Als Voreinstellung ist Signal R(A) 5 Volt, Signal R(B) 0 Volt ausgewählt.

3964(R)-Protokoll

PtP - Übertragung (3964(R))

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "3964(R)" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "3964(R)" gibt es 3 PtP-Bereiche Übertragung, Datenempfang und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Übertragung und ist in 3 Abschnitte unterteilt.

Übertragungsgeschwindigkeit

In der Klappliste "Übertragungsgeschwindigkeit" wählen Sie die Geschwindigkeit der Datenübertragung in Bit/s (Baud) aus. Sie können einen Wert zwischen 300 und 38400 Baud auswählen.

Als Voreinstellung ist "9600 Baud" ausgewählt.

Zeichenrahmen

Die Daten werden über die serielle Schnittstelle in einen Zeichenrahmen übertragen. Für jeden Zeichenrahmen stehen zwei Datenformate zur Verfügung.

- Datenbits
  In der Klappliste "Datenbits" wählen Sie die Anzahl der Bits aus, auf die ein Zeichen abgebildet wird.
  Als Voreinstellung ist "8 bit" ausgewählt.
- Stoppbits
  In der Klappliste "Stoppbits" wählen Sie die Anzahl der Bits aus, die bei der Übertragung jedem zu übertragenden Zeichen nachgesetzt werden. Die Stoppbits kennzeichnen das Ende eines Zeichens.
  Als Voreinstellung ist "1 bit" ausgewählt.
- Parität
  In der Klappliste "Parität" wählen Sie aus, ob die Sicherheit der Datenübertragung durch die Verwendung eines Paritätsbits erhöht werden soll. Parität "keine" bedeutet, dass kein Paritätsbit gesendet wird. **Bei Einstellung von 7 Datenbits ist die Einstellung "keine" nicht möglich.**
  Eine Folge von Informationsbits kann um ein weiteres Bit, das Paritätsbit, erweitert werden, das durch seinen addierten Wert ("0" oder "1") den Wert aller Bits auf einen vereinbarten Zustand - gerade oder ungerade - ergänzt.
  Als Voreinstellung ist "gerade" ausgewählt.
- Priorität
  In der Klappliste "Priorität" wählen Sie aus, ob die CPU hohe oder niedrige Priorität haben soll.
  Ein Partner hat hohe Priorität, wenn sein Sendewunsch Vorrang vor dem Sendewunsch des anderen Partners hat. Ein Partner hat niedrige Priorität, wenn sein Sendewunsch hinter dem Sendewunsch des anderen Partners zurückstehen muss. Bei der Prozedur 3964(R) müssen Sie beide Kommunikationspartner mit verschiedener Priorität parametrieren, d. h. ein Partner bekommt hohe Priorität, der andere Partner bekommt niedrige Priorität.
  Als Voreinstellung ist "hoch" ausgewählt.

Telegramm-Parameter

- Mit Blockcheck
  Klicken Sie auf das Optionskästchen "Mit Blockcheck", um festzulegen, ob Telegramme mit oder ohne Blockcheck gesendet werden sollen.
  - Ohne Blockcheck
    Die Protokollparameter sind frei parametrierbar.
    Erkennt die CPU die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei (oder NAK für einen fehlerhaft) empfangenen Block an den Kommunikationspartner.
  - Mit Blockcheck
    Die Protokollparameter sind mit Defaultwerten belegt.
    Erkennt die CPU die Zeichenfolge DLE ETX BCC, beendet sie den Empfang. Sie vergleicht das empfangene Blockprüfzeichen BCC mit der intern gebildeten Längsparität. Ist das Blockprüfzeichen korrekt und kein anderer Empfangsfehler aufgetreten, sendet sie das Zeichen DLE (bei einem Fehler wird das Zeichen NAK an den Kommunikationspartner gesendet).

  Als Voreinstellung ist das Optionskästchen "Mit Blockcheck" aktiviert.
- Standardwerte verwenden
  Klicken Sie auf das Optionskästchen "Standardwerte verwenden", wenn Sie die Protokollparameter mit Defaultwerten belegen wollen.
  Erkennt die Baugruppe die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei (oder NAK für einen fehlerhaft) empfangenen Block an den Kommunikationspartner.
  Wenn Sie keine Standardwerte verwenden wollen, können Sie die nachfolgenden Parameter einzeln festlegen.
  Als Voreinstellung ist das Optionskästchen "Standardwerte verwenden" aktiviert.
- Zeichenverzugszeit
  Im Eingabefeld "Zeichenverzugszeit" legen Sie den maximal zulässigen zeitlichen Abstand zwischen zwei empfangenen Zeichen innerhalb eines Telegramms fest. Sie können einen Wert zwischen "20 ms" bis "65530 ms" in 10 ms Schritten eintragen.
  Die kleinste Zeichenverzugszeit, die Sie wählen können, ist abhängig von der Übertragungsgeschwindigkeit.

Übertragungsgeschwindigkeit	Kleinste Zeichenverzugszeit
300 bit/s	60 ms
600 bit/s	40 ms
1200 bit/s	30 ms
2400 bit/s	20 ms
4800 bit/s	20 ms
9600 bit/s	20 ms
19200 bit/s	20 ms
38400 bit/s	20 ms

Als Voreinstellung sind "220 ms" eingetragen.

- Quittungsverzugszeit
  Im Eingabefeld "Quittungsverzugszeit" legen Sie die maximal zulässige Zeitspanne bis zur Quittung des Partners bei Verbindungsaufbau (Zeit zwischen STX und Quittung DLE des Partners) bzw. Verbindungsabbau (Zeit zwischen DLE ETX und Quittung DLE des Partners) fest. Sie können einen Wert zwischen "20 ms" bis "65530 ms" in 10 ms Schritten eintragen.
  Die kleinste Quittungsverzugszeit, die Sie wählen können, ist abhängig von der Übertragungsgeschwindigkeit.

Übertragungsgeschwindigkeit	Kleinste Quittungsverzugszeit
300 bit/s	60 ms
600 bit/s	40 ms
1200 bit/s	30 ms
2400 bit/s	20 ms
4800 bit/s	20 ms
9600 bit/s	20 ms
19200 bit/s	20 ms
38400 bit/s	20 ms

Als Voreinstellung sind "2000 ms" eingetragen.

- Aufbauversuche
  In der Liste "Aufbauversuche" legen Sie die maximale Anzahl der Versuche der Baugruppe, eine Verbindung aufzubauen, fest. Sie können einen Wert zwischen "1" und "255" wählen. Als Voreinstellung ist "6" ausgewählt.
- Übertragungsversuche
  In der Liste "Übertragungsversuche" legen Sie die maximale Anzahl der Versuche der Baugruppe, ein Telegramm zu übertragen (einschließlich des ersten Telegramms), bei Fehlern fest. Sie können einen Wert zwischen "1" und "255" wählen.
  Als Voreinstellung ist "6" ausgewählt.

PtP - Datenempfang (3964(R))

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "3964(R)" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "3964(R)" gibt es 3 PtP-Bereiche Übertragung, Datenempfang und Signalbelegung.

Dieses Topic beschäftigt sich mit dem PtP-Datenempfang.

Empfangspuffer auf CPU

- CPU-Empfangspuffer im Anlauf löschen
  Klicken Sie auf das Optionskästchen "CPU-Empfangspuffer im Anlauf löschen" dann wird bei Netz-Ein bzw. beim Übergang der CPU von STOP nach RUN der CPU-Empfangspuffer gelöscht.
  Als Voreinstellung ist dieses Optionskästchen nicht aktiviert.
- Überschreiben verhindern
  Klicken Sie auf das Optionskästchen "Überschreiben verhindern" damit bei einem vollen Empfangspuffer Daten im Empfangspuffer nicht überschrieben werden.
  Als Voreinstellung ist dieses Optionskästchen aktiviert.
- Gesamten Puffer ausnutzen
  Klicken Sie auf das Optionskästchen "Gesamten Puffer ausnutzen", wenn Sie den gesamten CPU-Empfangspuffer von 2048 Byte nutzen wollen. Wenn Sie den gesamten Puffer nutzen, ist die Anzahl der gepufferten Empfangstelegramme nur von der Länge der Telegramme abhängig.
  Als Voreinstellung ist dieses Optionskästchen aktiviert.
- Anzahl maximal gepufferter Empfangstelegramme
  Im Eingabefeld "Anzahl maximal gepufferter Empfangstelegramme" geben Sie die Anzahl der Empfangstelegramme an, die im CPU-Empfangspuffer gepuffert werden. Geben Sie einen Wert zwischen 1 und 10 an.
  Als Voreinstellung ist der Wert "10" eingetragen.

Hinweis

Wenn Sie hier "1" parametrieren und den Parameter "Überschreiben verhindern" deaktivieren **und** die Empfangsdaten im Anwenderprogramm zyklisch auslesen, wird immer ein aktuelles Telegramm zur CPU übertragen.

PtP - Signalbelegung (3964(R))

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "3964(R)" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "3964(R)" gibt es 3 PtP-Bereiche Übertragung, Datenempfang, und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Signalbelegung.

Vorbelegung der Empfangsleitung

Für das Protokoll 3964(R) wird die X27 (RS 422/485)-Schnittstelle Vollduplex (RS 422) im Vierdrahtbetrieb betrieben. Dies ist die geeignete Betriebsart für Punkt-zu-Punkt-Kopplungen im Vierdraht-Betrieb.

Als Vorbelegung der Empfangsleitung sind folgende Varianten möglich:

- Keine
  Mit dem Optionsfeld "Keine" wählen Sie eine Einstellung die nur sinnvoll ist für busfähige Sondertreiber.
- Signal R(A) 5 V, Signal R(B) 0 V (Break-Erkennung)
  Mit diesem Optionsfeld wählen Sie eine Einstellung, bei der Break-Erkennung möglich ist.
- Signal R(A) 0 V, Signal R(B) 5 V
  Mit diesem Optionsfeld wählen Sie eine Einstellung die dem Ruhezustand entspricht (kein Sender aktiv). Bei dieser Einstellung ist keine Break-Erkennung möglich.

Als Voreinstellung ist "Signal R(A) 5 V, Signal R(B) 0 V (Break-Erkennung)" ausgewählt.

RK512-Protokoll

PtP - Übertragung (RK512)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "RK512" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "RK512" gibt es 2 PtP-Bereiche Übertragung und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Übertragung und ist in 3 Abschnitte unterteilt.

Übertragungsgeschwindigkeit

In der Klappliste "Übertragungsgeschwindigkeit" wählen Sie die Geschwindigkeit der Datenübertragung in bit/s (Baud) aus. Sie können einen Wert zwischen 300 und 38400 Baud auswählen.

Als Voreinstellung ist "9600 Baud" ausgewählt.

Zeichenrahmen

Die Daten werden über die serielle Schnittstelle in einem Zeichenrahmen übertragen. Für jeden Zeichenrahmen stehen zwei Datenformate zur Verfügung.

- Datenbits
  Die Klappliste "Datenbits" ist gegraut, da bei RK 512 ein Zeichen immer auf 8 bit abgebildet wird.
- Stoppbits
  In der Klappliste "Stoppbits" wählen Sie die Anzahl der Bits aus, die bei der Übertragung jedem zu übertragenden Zeichen nachgesetzt werden. Die Stoppbits kennzeichnen das Ende eines Zeichens.
  Als Voreinstellung ist "1 bit" ausgewählt.

- Parität
  In der Klappliste "Parität" wählen Sie aus, ob die Sicherheit der Datenübertragung durch die Verwendung eines Paritätsbits erhöht werden soll. Parität "keine" bedeutet, dass kein Paritätsbit gesendet wird. Eine Folge von Informationsbits kann um ein weiteres Bit, das Paritätsbit, erweitert werden, das durch seinen addierten Wert ("0" oder "1") den Wert aller Bits auf einen vereinbarten Zustand - gerade oder ungerade - ergänzt.
  Als Voreinstellung ist "gerade" ausgewählt.
- Priorität
  In der Klappliste "Priorität" wählen Sie aus, ob die CPU hohe oder niedrige Priorität haben soll.
  Ein Partner hat hohe Priorität, wenn sein Sendewunsch Vorrang vor dem Sendewunsch des anderen Partners hat. Ein Partner hat niedrige Priorität, wenn sein Sendewunsch hinter dem Sendewunsch des anderen Partners zurückstehen muss. Bei RK 512 müssen Sie beide Kommunikationspartner mit verschiedener Priorität parametrieren, d. h. ein Partner bekommt hohe Priorität, der andere Partner bekommt niedrige Priorität.
  Als Voreinstellung ist "hoch" ausgewählt.

Telegramm-Parameter

- Mit Blockcheck
  Klicken Sie auf das Optionskästchen "Mit Blockcheck", um festzulegen, ob Telegramme mit oder ohne Blockcheck gesendet werden sollen.
  - Ohne Blockcheck
    Die Protokollparameter sind frei parametrierbar.
    Erkennt die CPU die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei (oder NAK für einen fehlerhaft) empfangenen Block an den Kommunikationspartner.
  - Mit Blockcheck
    Die Protokollparameter sind mit Defaultwerten belegt.
    Erkennt die CPU die Zeichenfolge DLE ETX BCC, beendet sie den Empfang. Sie vergleicht das empfangene Blockprüfzeichen BCC mit der intern gebildeten Längsparität. Ist das Blockprüfzeichen korrekt und kein anderer Empfangsfehler aufgetreten, sendet sie das Zeichen DLE (bei einem Fehler wird das Zeichen NAK an den Kommunikationspartner gesendet).

  Als Voreinstellung ist das Optionskästchen "Mit Blockcheck" aktiviert.
- Standardwerte verwenden
  Klicken Sie auf das Optionskästchen "Standardwerte verwenden", wenn Sie die Protokollparameter mit Defaultwerten belegen wollen.
  Erkennt die Baugruppe die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei (oder NAK für einen fehlerhaft) empfangenen Block an den Kommunikationspartner.
  Wenn Sie keine Standardwerte verwenden wollen, können Sie die nachfolgenden Parameter einzeln festlegen.
  Als Voreinstellung ist das Optionskästchen "Standardwerte verwenden" aktiviert.
- Zeichenverzugszeit
  Im Eingabefeld "Zeichenverzugszeit" legen Sie den maximal zulässigen zeitlichen Abstand zwischen zwei empfangenen Zeichen innerhalb eines Telegramms fest. Sie können einen Wert zwischen "20 ms" bis "65530 ms" in 10 ms Schritten eintragen.
  Die kleinste Zeichenverzugszeit, die Sie wählen können, ist abhängig von der Übertragungsgeschwindigkeit.

Übertragungsgeschwindigkeit	Kleinste Zeichenverzugszeit
300 bit/s	60 ms
600 bit/s	40 ms
1200 bit/s	30 ms
2400 bit/s	20 ms
4800 bit/s	20 ms
9600 bit/s	20 ms
19200 bit/s	20 ms
38400 bit/s	20 ms

Als Voreinstellung sind "220 ms" eingetragen.

- Quittungsverzugszeit
  Im Eingabefeld "Quittungsverzugszeit" legen Sie die maximal zulässige Zeitspanne bis zur Quittung des Partners bei Verbindungsaufbau (Zeit zwischen STX und Quittung DLE des Partners) bzw. Verbindungsabbau (Zeit zwischen DLE ETX und Quittung DLE des Partners) fest. Sie können einen Wert zwischen "20 ms" bis "65530 ms" in 10 ms Schritten eintragen.
  Die kleinste Quittungsverzugszeit, die Sie wählen können, ist abhängig von der Übertragungsgeschwindigkeit.

Übertragungsgeschwindigkeit	Kleinste Quittungsverzugszeit
300 bit/s	60 ms
600 bit/s	40 ms
1200 bit/s	30 ms
2400 bit/s	20 ms
4800 bit/s	20 ms
9600 bit/s	20 ms
19200 bit/s	20 ms
38400 bit/s	20 ms

Als Voreinstellung ist "2000 ms" eingetragen.

- Aufbauversuche
  In der Liste "Aufbauversuche" legen Sie die maximale Anzahl der Versuche der Baugruppe, eine Verbindung aufzubauen, fest. Sie können einen Wert zwischen 1 und 255 wählen.
  Als Voreinstellung ist "6" ausgewählt.
- Übertragungsversuche
  In der Liste "Übertragungsversuche" legen Sie die maximale Anzahl der Versuche der Baugruppe, ein Telegramm zu übertragen (einschließlich des ersten Telegramms) fest. Sie können einen Wert zwischen "1" und "255" wählen.
  Als Voreinstellung ist "6" ausgewählt.

PtP - Signalbelegung (RK512)

Beschreibung

Unter "Eigenschaften > Allgemein > PtP > Alarmauswahl" werden die Alarme, Verhalten bei CPU-STOP und das Protokoll eingestellt. Wählen Sie für Protokoll "RK512" aus.

Je nach dem, welche Protokollart gewählt wurde, ändert sich die weitere Unterteilung in der Bereichsnavigation.

Bei der Wahl von "RK 512" gibt es 2 PtP-Bereiche Übertragung und Signalbelegung.

Dieser Abschnitt beschäftigt sich mit der PtP-Übertragung und ist in 3 Abschnitte unterteilt.

Vorbelegung der Empfangsleitung

Die X27 (RS 422/485)-Schnittstelle wird Vollduplex (RS 422) im Vierdrahtbetrieb betrieben. Dies ist die geeignete Betriebsart für Punkt-zu-Punkt-Kopplungen im Vierdraht-Betrieb.

Als Vorbelegung der Empfangsleitung sind folgende Varianten möglich:

- Keine
  Mit dem Optionsfeld "Keine" wählen Sie eine Einstellung die nur sinnvoll ist für busfähige Sondertreiber.
- Signal R(A) 5 V, Signal R(B) 0 V (Break-Erkennung)
  Mit diesem Optionsfeld wählen Sie eine Einstellung, bei der Break-Erkennung möglich ist.
- Signal R(A) 0 V, Signal R(B) 5 V
  Mit diesem Optionsfeld wählen Sie eine Einstellung die dem Ruhezustand entspricht (kein Sender aktiv). Bei dieser Einstellung ist keine Break-Erkennung möglich.
- Als Voreinstellung ist "Signal R(A) 5 V, Signal R(B) 0 V (Break-Erkennung)" ausgewählt.

CPU Informationen

Register der CPU

Übersicht

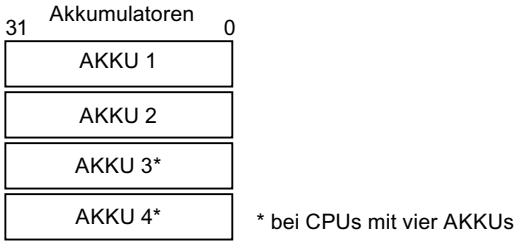

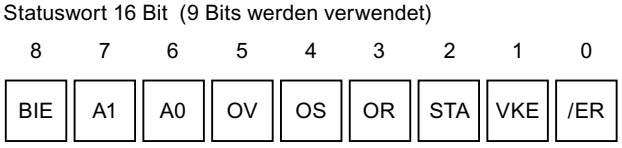

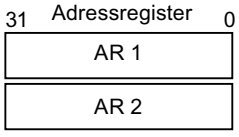

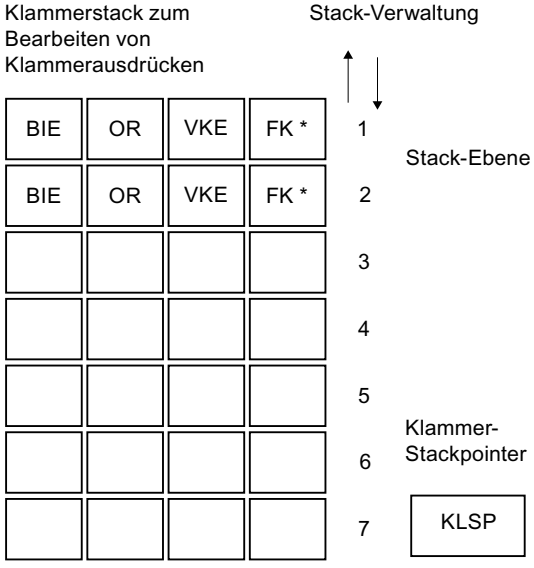

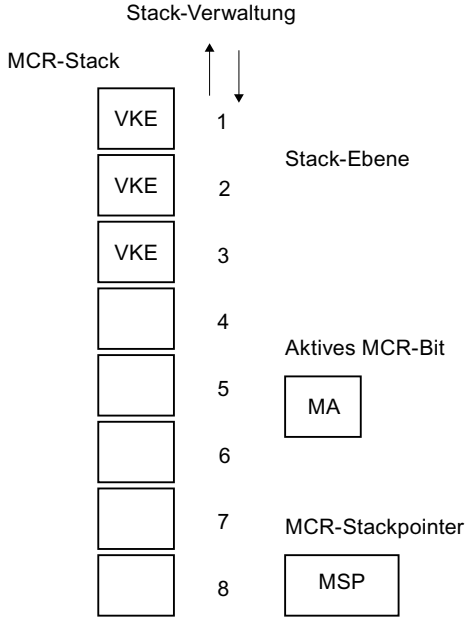

Akkumulatoren

Beschreibung

Die 32-Bit-Akkumulatoren sind Universalregister zur Verarbeitung von Bytes, Wörtern und Doppelwörtern. Dazu werden die Operanden in die AKKUs geladen und dort verknüpft. Das Ergebnis der Operation steht immer im AKKU 1 und kann von dort in eine Speicherzelle transferiert werden.

Der Stack-Mechanismus für die Akkumulatorverwaltung funktioniert folgendermaßen:

- Eine Ladeoperation wirkt nur auf AKKU 1 und speichert den alten Inhalt in AKKU 2.
- Eine Transferoperation (Kopieroperation) ändert die Inhalte der Akkumulatoren nicht.
- Die Operation **TAK** tauscht die Inhalte von AKKU 1 und AKKU 2.
- Das Ergebnis von Verknüpfungen zwischen AKKU 1 und AKKU 2 (arithmetische Operationen, Vergleichsoperationen, UND, ODER, ...) wird immer in AKKU 1 gespeichert.

Definition von Bytes und Wörtern in einem 32-Bit-Register (am Beispiel von AKKU 1)

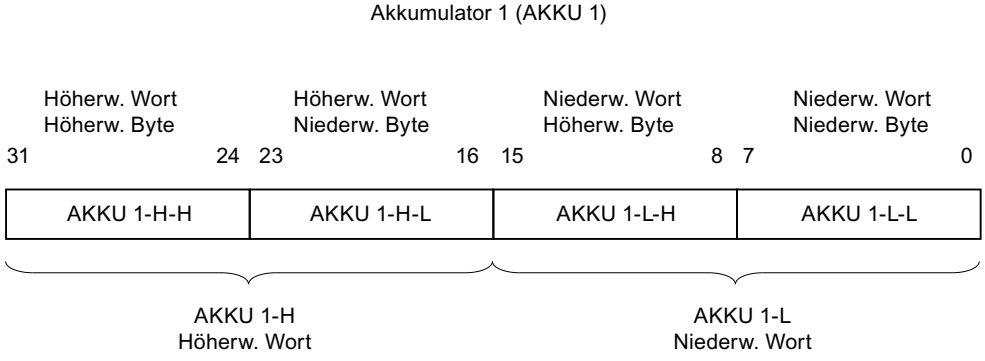

Adressierung

Prozessabbilder (Speicherbereich der Ein- und Ausgänge)

Speicherbereich der Prozessabbilder

Das OB-1-Prozessabbild (OB1-PA) wird vom Betriebssystem zyklisch aktualisiert.

Dieses Prozessabbild wird von der Hardware der Peripherie während jedes Zyklus der Anweisungsliste gespeichert. Deshalb ändern sich die Bits des Speicherbereichs E im Prozessabbild während der Zugriffe auf die Anweisungen nicht. Obwohl die Bits des Speicherbereichs A im Prozessabbild während des Programmzyklus verschiedene Zustände vorweisen können, wird nur der letzte Signalzustand an die Ausgabebaugruppe geschickt.

Siehe auch: Nutzdaten (Seite 784) und Aktualisieren der Prozessabbilder (Seite 778)

Ausnahmen

In folgenden Fällen wird nicht über das Prozessabbild auf die Peripherie zugegriffen:

- Direkte Adressierung der Ein- und Ausgänge im Speicherbereich der Peripherie (P)
- Verwenden von Funktionen, die Daten der Peripherie sofort verarbeiten.

Beispiel für einen normalen Zugriff auf die Peripherie über das Prozessabbild

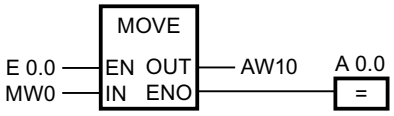

Aktualisiere AW10 im Peripheriebereich Ausgänge mit dem Wert von MW0.

Siehe auch

Zugriff auf die Peripherieadressen (Seite 787)

Direkter Zugriff auf die Peripherie (Speicherbereich PE und PA)

Direkter Zugriff auf die Peripherie

Ist es erforderlich, dass Ihr Projekt die Daten der Peripherie schneller aktualisiert als das Prozessabbild diese zur Verfügung stellt, können Sie auf die Signalbaugruppen direkt im Peripheriebereich Ein- und Ausgänge (PE, PA) zugreifen. Der aktuelle Status der Eingabebaugruppen kann aus dem Speicherbereich PE ausgelesen werden. Ausgabebaugruppen können gesetzt werden, indem Sie in den Speicherbereich PA schreiben.

⚠ VORSICHT

Einzelne Bits (Byte. Bit) können im Peripheriebereich nicht adressiert werden. Zum Setzen einzelner Bits müssen Sie Bytes, Wörter oder Doppelwörter in Verknüpfungsoperationen adressieren. Wenn Sie ein aktualisiertes Bit (innerhalb eines Bytes, Worts oder Doppelworts) in den Speicherbereich PA schreiben, werden alle Bits in der Adresse dieses Peripheriebereichs aktualisiert!

Informieren Sie sich unter dem Eintrag **Prozessabbild** über die Verbindung zwischen dem Speicherbereich Peripherie (Speicherbereich PE, PA) und dem Prozessabbild (Speicherbereiche E und A). Lesen Sie dort auch, wie das Prozessabbild das speichert (stabilisiert), was aus den Signalbaugruppen geschrieben bzw. an sie geschickt wird.

Beispiel für den direkten Zugriff auf die Peripherie über den Speicherbereich Peripherie

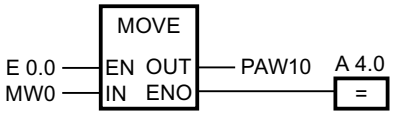

Aktualisiere PAW10 im Peripheriebereich Ausgänge mit dem Wert von MW0.

Siehe auch

Zugriff auf die Peripherieadressen (Seite 787)
Adressbereiche (Seite 787)

Ausgeschriebene Syntax für die Eingabe von Konstanten

Übersicht

Kurzfassung	Langfassung
B#	BYTE#
W#	WORD#
DW#	DWORD#
D#	DATE#
T#	TIME#
TOD#	TIME_OF_DAY#
S5T#	S5TIME#
DT#	DATE_AND_TIME#
P#	POINTER#

Zusammenfassung der Bit-, Byte-, Wort- und Doppelwortkennungen

Übersicht

Speicherbereich der CPU	Operandenkennzeichen			
	Bit	Byte	Wort	Doppelwort
Prozessabbild der Eingänge	E	EB	EW	ED
Prozessabbild der Ausgänge	A	AB	AW	AD
Merker	M	MB	MW	MD
Lokale Daten	L	LB	LW	LD
Direkter Peripheriebereich Eingänge		PEB	PEW	PED
Direkter Peripheriebereich Ausgänge		PAB	PAW	PAD
Datenbaustein [OPN] DB [OPN] DI	 DBX DIX	 DBB DIB	 DBW DIW	 DBD DID
Timer (T)	T		T	
Zähler (Z)	Z		Z	

Hinweis

Die beiden Operanden **EB0** und **EB1** (Speicherbereich E, Format: Byte, Adressenbytes 0, 1) adressieren die gleichen zwei Bytes wie der Operand **EW0** (Speicherbereich E, Format: Wort, 2 Bytes, Inkrementierung von Byte 0 zu Adressenbytes 0, 1).

Kennungen von Elementen und Bausteinen

Übersicht

T	Zeit
DB	globaler Datenbaustein
DI	Instanz-Datenbaustein
FC	Funktion
FB	Funktionsbaustein
OB	Organisationsbaustein
Z/C	Zähler (abhängig von der Mnemonik)
SDB	Systemdatenbaustein
SFC	Systemfunktion
SFB	Systemfunktionsbaustein

Organisationsbausteine

Englische Bezeichnungen der OB-Typen

Englische Bezeichnung der OB-Typen

Beim Anlegen eines OB im Dialog "Neuen Baustein hinzufügen" können die OB-Typen aus technischen Gründen nur in Englisch angezeigt werden – unabhängig von der eingestellten Oberflächensprache. Um Ihnen die Handhabung zu erleichtern, werden in der folgenden Tabelle den englischen Begriffen die Bezeichnungen in der eingestellten Oberflächensprache gegenübergestellt. Die Tabelle gilt für die Obermenge von S7-300- und S7-400-CPUs.

Englische Bezeichnung des OB-Typs im Dialog "Neuen Baustein hinzufügen"			Bezeichnung im Informationssystem
Time interrupts			Zeitalarme
	Time of day		Uhrzeitalarme
		TOD_INT0 [OB10] bis TOD_INT7 [OB17]	Uhrzeitalarm-OBs

Englische Bezeichnung des OB-Typs im Dialog "Neuen Baustein hinzufügen"			Bezeichnung im Informationssystem
	Time delay		Verzögerungsalarme
		DEL_INT0 [OB20] bis DEL_INT3 [OB23]	Verzögerungsalarm-OBs
	Cyclic		Weckalarme
		CYC_INT0 [OB30] bis CYC_INT8 [OB38]	Weckalarm-OBs
Hardware Interrupts			Prozessalarme
		HW_INT0 [OB40] bis HW_INT7 [OB47]	Prozessalarm-OB
Startup			Anlauf
		BACKGROUND [OB90]	Hintergrund-OB
		COMPLETE RESTART [OB100]	Neustart (Warmstart)
		RESTART [OB101]	Wiederanlauf
		COLD RESTART [OB102]	Kaltstart
Alarming			
		DP: STATUS ALARM [OB55]	Statusalarm-OB
		DP: UPDATE ALARM [OB56]	Update-Alarm-OB
		DP: MANUFACTURE ALARM [OB57]	OB für herstellerspezifische Alarme
		MULTI_INT [OB 60]	Multicomputingalarm-OB
		SYNC_1 [OB61] bis SYNC_4 [OB64]	Taktsynchronalarm-OBs
Fault Interrupts			Fehler
		CYCL_FLT [OB 80]	Zeitfehler-OB
		PS_FLT [OB81]	Stromversorgungsfehler-OB
		I/O_FLT1 [OB82]	Diagnosealarm-OB
		I/O_FLT2 [OB83]	Ziehen/Stecken-OB
		CPU_FLT [OB84]	CPU-Hardwarefehler-OB
		OBNL_FLT [OB85]	Programmablauffehler-OB
		RACK_FLT [OB86]	Baugruppenträgerausfall-OB
		COMM_FLT [OB87]	Kommunikationsfehler-OB
		BREAKUP ERROR [OB88]	Bearbeitungsabbruch-OB
		PROG_ERR [OB121]	Programmierfehler-OB
		MOD_ERR [OB122]	Peripheriezugriffsfehler-OB

Übersicht über die OBs

Organisationsbausteine

Organisationsbausteine bilden die Schnittstelle zwischen dem Betriebssystem der CPU und dem Anwenderprogramm. Mit Hilfe von OBs können Programmteile gezielt zur Ausführung gebracht werden:

- beim Anlauf der CPU
- in zyklischer oder auch zeitlich getakteter Ausführung

- beim Auftreten von Fehlern
- beim Auftreten von Prozessalarmen.

Organisationsbausteine werden entsprechend der ihnen zugeordneten Priorität bearbeitet.

Verfügbare OBs

Nicht alle CPUs können alle in S7 verfügbaren OBs bearbeiten. Sie entnehmen der Operationsliste Ihrer CPU, welche OBs Ihnen zur Verfügung stehen.

Startereignisse und voreingestellte Prioritätsklasse

Die folgende Tabelle enthält zu jedem OB das zugehörige Startereignis und die voreingestellte Prioritätsklasse.

OB	Startereignis	Voreingestellte Prioritätsklasse	Erläuterung
OB 1	Anlaufende oder OB 1-Ende	1	Freier Zyklus
OB 10	Uhrzeitalarm 0	2	Keine Defaultzeitvorgaben
OB 11	Uhrzeitalarm 1	2	
OB 12	Uhrzeitalarm 2	2	
OB 13	Uhrzeitalarm 3	2	
OB 14	Uhrzeitalarm 4	2	
OB 15	Uhrzeitalarm 5	2	
OB 16	Uhrzeitalarm 6	2	
OB 17	Uhrzeitalarm 7	2	
OB 20	Verzögerungsalarm 0	3	Keine Defaultzeitvorgaben
OB 21	Verzögerungsalarm 1	4	
OB 22	Verzögerungsalarm 2	5	
OB 23	Verzögerungsalarm 3	6	
OB 30	Weckalarm 0 (Default: 5 s-Takt)	7	Weckalarme
OB 31	Weckalarm 1 (Default: 2 s-Takt)	8	
OB 32	Weckalarm 2 (Default: 1 s-Takt)	9	
OB 33	Weckalarm 3 (Default: 500 ms-Takt)	10	
OB 34	Weckalarm 4 (Default: 200 ms-Takt)	11	
OB 35	Weckalarm 5 (Default: 100 ms-Takt)	12	
OB 36	Weckalarm 6 (Default: 50 ms-Takt)	13	
OB 37	Weckalarm 7 (Default: 20 ms-Takt)	14	
OB 38	Weckalarm 8 (Default: 10 ms-Takt)	15	

OB	Startereignis	Voreingestellte Prioritätsklasse	Erläuterung
OB 40	Prozessalarm 0	16	Prozessalarme
OB 41	Prozessalarm 1	17	
OB 42	Prozessalarm 2	18	
OB 43	Prozessalarm 3	19	
OB 44	Prozessalarm 4	20	
OB 45	Prozessalarm 5	21	
OB 46	Prozessalarm 6	22	
OB 47	Prozessalarm 7	23	
OB 55	Statusalarm	2	DPV1-Alarme
OB 56	Update-Alarm	2	
OB 57	Herstellerspezifischer Alarm	2	
OB 61	Taktsynchronalarm 1	25	Taktsynchron-Alarme
OB 62	Taktsynchronalarm 2	25	
OB 63	Taktsynchronalarm 3	25	
OB 64	Taktsynchronalarm 4	25	
OB 80	Zeitfehler	26, 28 [1]	Asynchrone Fehleralarme
OB 81	Stromversorgungsfehler	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 82	Diagnosealarm	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 83	Ziehen/Stecken-Alarm	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 84	CPU-Hardwarefehler	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 85	Programmablauffehler	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 86	Ausfall eines Erweiterungsgeräts, eines DP-Mastersystems oder einer Station bei Dezentraler Peripherie	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 87	Kommunikationsfehler	26, 28 [1] bei S7-300, 25, 28 [1] bei S7-400	
OB 88	Bearbeitungsabbruch	28	
OB 90	Neustart (Warmstart) oder Kaltstart oder Löschen eines im OB 90 in Bearbeitung befindlichen Bausteins oder Laden eines OB 90 in die CPU oder OB 90-Ende	29 [2)]	Hintergrundzyklus
OB 100	Neustart (Warmstart)	27 [1]	Anlauf
OB 101	Wiederanlauf	27 [1]	
OB 102	Kaltstart	27 [1]	

OB	Startereignis	Voreingestellte Prioritätsklasse	Erläuterung
OB 121	Programmierfehler	Priorität des Fehler verursachenden OB	Synchrone Fehleralarme
OB 122	Peripheriezugriffsfehler	Priorität des Fehler verursachenden OB	
[1] Die Prioritätsklassen 27 und 28 sind gültig im Prioritätsklassenmodell des Anlaufs. [2] Der Prioritätsklasse 29 entspricht die Priorität 0.29. Der Hintergrundzyklus hat also eine niedrigere Priorität als der Freie Zyklus.			

Zyklisches Programm (OB 1)

Beschreibung

Das Betriebssystem der S7 CPU bearbeitet den OB 1 zyklisch: Ist die Bearbeitung des OB 1 beendet, beginnt das Betriebssystem erneut, den OB 1 zu bearbeiten. Die zyklische Bearbeitung des OB 1 wird nach Beendigung des Anlaufs begonnen. Sie können im OB 1 Anweisungen aufrufen.

Funktionsweise von OB 1

Der OB 1 hat die niedrigste Priorität aller laufzeitüberwachten OBs. Mit Ausnahme des OB 90 können alle anderen OBs die Bearbeitung des OB 1 unterbrechen. Die folgenden Ereignisse bewirken, dass das Betriebssystem den OB 1 aufruft:

- Ende der Bearbeitung des Anlaufs
- Ende der Bearbeitung des OB 1 (des vorherigen Zyklus).

Wenn die Bearbeitung des OB 1 beendet ist, sendet das Betriebssystem Globaldaten. Bevor der OB 1 neu gestartet wird, schreibt das Betriebssystem das Prozessabbild der Ausgänge in die Ausgabebaugruppen, aktualisiert das Prozessabbild der Eingänge und empfängt Globaldaten für die CPU.

S7 bietet eine Überwachung der maximalen Zykluszeit, wodurch die maximale Reaktionszeit garantiert wird. Der Wert für die maximale Zykluszeit ist auf 150 ms voreingestellt; Sie können diesen Wert umparametrieren oder aber mit der Anweisung "RE_TRIGR" an jeder Stelle Ihres Programms die Zeitüberwachung neu starten. Überschreitet Ihr Programm die maximale Zykluszeit für den OB 1, dann ruft das Betriebssystem den OB 80 (Zeitfehler-OB) auf. Ist der OB 80 nicht programmiert, dann geht die CPU in den Betriebszustand STOP.

Neben der Überwachung der maximalen Zykluszeit kann die Einhaltung einer minimalen Zykluszeit garantiert werden. Das Betriebssystem verzögert den Beginn eines neuen Zyklus (Schreiben des Prozessabbilds der Ausgänge in die Ausgabebaugruppen) solange, bis die minimale Zykluszeit erreicht ist.

Eine Änderung der Parameter maximale und minimale Zykluszeit erfolgt per Projektierung.

Lokaldaten des OB 1

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des OB 1. Als Variablennamen wurden die Defaultnamen des OB 1 gewählt.

Variable	Datentyp	Beschreibung
OB1_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: OB 1 ist aktiv
OB1_SCAN_1	BYTE	• B#16#01: Abschluss des Neustarts (Warmstarts) • B#16#02: Abschluss des Wiederanlaufs • B#16#03: Abschluss des freien Zyklus • B#16#04: Abschluss des Kaltstarts • B#16#05: Erster OB 1-Zyklus der neuen Master-CPU nach Master-Reserve-Umschaltung und STOP des bisherigen Masters
OB1_PRIORITY	BYTE	Prioritätsklasse: 1
OB1_OB_NUMBR	BYTE	OB-Nr. (01)
OB1_RESERVED_1	BYTE	Reserviert
OB1_RESERVED_2	BYTE	Reserviert
OB1_PREV_CYCLE	INT	Laufzeit des vorherigen Zyklus (ms)
OB1_MIN_CYCLE	INT	Minimale Zykluszeit (ms) seit dem letzten Anlauf
OB1_MAX_CYCLE	INT	Maximale Zykluszeit (ms) seit dem letzten Anlauf
OB1_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Uhrzeitalarm-OBs (OB 10 bis OB 17)

Beschreibung

S7 stellt bis zu acht OBs (OB 10 bis OB 17) zur Verfügung, die einmalig oder periodisch gestartet werden. Sie können Ihre CPU so parametrieren, dass diese OBs in den folgenden Intervallen bearbeitet werden:

- Einmalig
- Minütlich
- Stündlich
- Täglich
- Wöchentlich
- Monatlich
- Jährlich
- Am Monatsende

Hinweis

Bei monatlicher Bearbeitung eines Uhrzeitalarm-OB sind beim Startdatum nur die Tage 1, 2, ... 28 möglich.

Funktionsweise der Uhrzeitalarm-OBs

Um einen Uhrzeitalarm zu starten, müssen Sie den Alarm erst einstellen und dann aktivieren. Es gibt folgende drei Startmöglichkeiten:

- Automatischer Start des Uhrzeitalarms. Dieser erfolgt, wenn Sie sowohl das Stellen als auch das Aktivieren des Uhrzeitalarms per Projektierung vorgenommen haben. Die folgende Tabelle zeigt die prinzipiellen Möglichkeiten beim Aktivieren eines Uhrzeitalarms per Projektierung.
- Sie stellen den Uhrzeitalarm per Projektierung und aktivieren ihn durch Aufrufen der Anweisung "ACT_TINT" aus Ihrem Programm heraus.
- Sie stellen den Uhrzeitalarm durch Aufruf der Anweisung "SET_TINT" und aktivieren ihn durch Aufruf der Anweisung "ACT_TINT".

Intervall	Beschreibung
Nicht aktiviert	Der Uhrzeitalarm-OB wird nicht bearbeitet, auch nicht, wenn er in die CPU geladen ist. Sie können den Uhrzeitalarm aktivieren durch Aufruf der Anweisung "ACT_TINT".
Einmalig aktiviert	Der Uhrzeitalarm-OB wird automatisch storniert, nachdem er einmal wie angegeben bearbeitet wurde. Ihr Programm kann mit der Anweisung "SET_TINT" den Uhrzeitalarm erneut stellen und ihn mit der Anweisung "ACT_TINT" wieder aktivieren.
Periodisch aktiviert	Die CPU ermittelt zum Zeitpunkt des Uhrzeitalarms aus der aktuellen Uhrzeit und der Periode den nächsten Startzeitpunkt des Uhrzeitalarms.

Hinweis

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB einmal bearbeitet werden soll, dürfen das Datum und die Uhrzeit nicht in der Vergangenheit liegen (bezogen auf die Echtzeituhr der CPU).

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB periodisch bearbeitet werden soll, das Datum und die Uhrzeit aber in der Vergangenheit liegen, dann wird der Uhrzeitalarm-OB bei dem nächsten fälligen Zeitpunkt nach der aktuellen Uhrzeit bearbeitet. Das folgende Bild zeigt die erste Bearbeitung eines Uhrzeitalarm-OBs, falls der Startzeitpunkt in der Vergangenheit liegt und periodische Aktivierung eingestellt wurde.

Sie können den Aufruf der Uhrzeitalarm-OBs mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT " wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

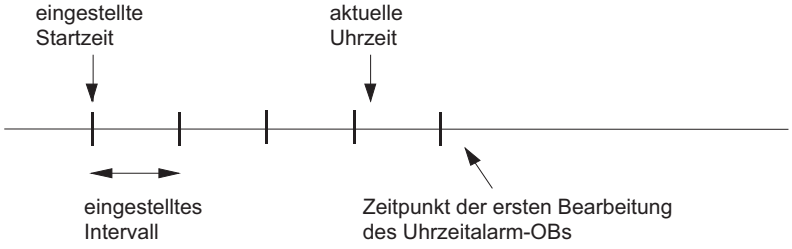

Bedingungen, die Uhrzeitalarm-OBs beeinträchtigen

Da ein Uhrzeitalarm nur in bestimmten Intervallen auftritt, können gewisse Bedingungen die Funktion der zugehörigen OBs während der Bearbeitung Ihres Programms beeinträchtigen. Die folgende Tabelle zeigt einige dieser Bedingungen und beschreibt die Wirkung auf die Bearbeitung des Uhrzeitalarm-OBs.

Bedingung	Ergebnis
Ihr Programm ruft die Anweisung "CAN_TINT" auf und storniert einen Uhrzeitalarm.	Das Betriebssystem löscht das Startereignis (Datum und Uhrzeit) des Uhrzeitalarms. Sie müssen das Startereignis neu einstellen und aktivieren, wenn der OB wieder aufgerufen werden soll.
Ihr Programm hat versucht, einen Uhrzeitalarm-OB zu aktivieren, der zum Zeitpunkt der Aktivierung nicht in die CPU geladen war.	Das Betriebssystem ruft den OB 85 auf. Wurde der OB 85 nicht programmiert (in die CPU geladen), dann geht die CPU in den Betriebszustand STOP.
Beim Synchronisieren oder Korrigieren der Systemuhr der CPU haben Sie die Uhrzeit vorgestellt und das Startereignis, das Datum oder die Uhrzeit für einen Uhrzeitalarm-OB übersprungen.	Das Betriebssystem ruft den OB 80 auf und codiert die Nummer des Uhrzeitalarm-OB und die Startereignis-Informationen in den OB 80. Das Betriebssystem bearbeitet dann den Uhrzeitalarm-OB einmal, unabhängig davon, wie oft dieser OB eigentlich hätte bearbeitet werden sollen. Die Startereignis-Informationen des OB 80 zeigen das Datum und die Uhrzeit, zu denen der Uhrzeitalarm-OB zum ersten Mal übersprungen wurde.
Beim Synchronisieren oder Korrigieren der Systemuhr der CPU haben Sie die Uhrzeit zurückgestellt und das Startereignis, das Datum oder die Uhrzeit für einen Uhrzeitalarm-OB wiederholt.	S7-400-CPUs: Wurde der Uhrzeitalarm-OB bereits aktiviert, bevor die Uhr zurückgestellt wurde, dann wird er für die bereits durchlaufenen Uhrzeiten nicht erneut aufgerufen. S7-300-CPUs: Der Uhrzeitalarm-OB wird ausgeführt.
Die CPU führt einen Neustart (Warmstart) oder Kaltstart aus.	Jeder Uhrzeitalarm-OB, der über eine Anweisung konfiguriert wurde, nimmt wieder die Konfiguration an, die Sie projektiert haben. Falls Sie einen Uhrzeitalarm auf einmaligen Start des zugehörigen OB konfiguriert und ihn per Projektierung gestellt und aktiviert haben, dann wird der OB nach einem Neustart (Warmstart) oder Kaltstart vom Betriebssystem einmalig aufgerufen, falls der projektierte Startzeitpunkt in der Vergangenheit liegt (bezogen auf die Echtzeituhr der CPU).
Ein Uhrzeitalarm-OB wird noch bearbeitet, während bereits das Startereignis für das nächste Intervall auftritt.	Das Betriebssystem ruft den OB 80 auf. Wurde der OB 80 nicht programmiert, dann wechselt die CPU in den Betriebszustand STOP. Andernfalls wird nach der Bearbeitung des OB 80 und des Uhrzeitalarm-OB die angeforderte OB-Bearbeitung nachgeholt.

Lokaldaten der Uhrzeitalarm-OBs

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen eines Uhrzeitalarm-OB. Als Variablennamen wurden die Defaultnamen des OB 10 gewählt.

Variable	Datentyp	Beschreibung
OB10_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: Alarm ist aktiv
OB10_STRT_INFO	BYTE	B#16#11: Startanforderung für OB 10 (B#16#12: Startanforderung für OB 11) : : (B#16#18: Startanforderung für OB 17)
OB10_PRIORITY	BYTE	Parametrierte Prioritätsklasse; Defaultwert: 2
OB10_OB_NUMBR	BYTE	OB-Nr. (10 bis 17)
OB10_RESERVED_1	BYTE	Reserviert
OB10_RESERVED_2	BYTE	Reserviert
OB10_PERIOD_EXE	WORD	Der OB wird zu dem angegebenen Intervall bearbeitet: W#16#0000: Einmalig W#16#0201: Minütlich W#16#0401: Stündlich W#16#1001: Täglich W#16#1201: Wöchentlich W#16#1401: Monatlich W#16#1801: Jährlich W#16#2001: Am Monatsende
OB10_RESERVED_3	INT	Reserviert
OB10_RESERVED_4	INT	Reserviert
OB10_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Verzögerungsalarm-OBs (OB 20 bis OB 23)

Beschreibung

S7 stellt bis zu vier OBs (OB 20 bis OB 23) zur Verfügung, die jeweils nach einer parametrierbaren Verzögerungszeit bearbeitet werden. Jeder Verzögerungsalarm-OB wird durch einen Aufruf der Anweisung "SRT_DINT" gestartet. Die Verzögerungszeit ist ein Eingangsparameter der Anweisung.

Wenn Ihr Programm die Anweisung "SRT_DINT" aufruft, dann übergeben Sie ihr die OB-Nummer, die Verzögerungszeit und ein anwenderspezifisches Kennzeichen. Nach der angegebenen Verzögerungszeit startet der zugehörige OB. Sie können die Bearbeitung eines Verzögerungsalarms, der noch nicht gestartet wurde, auch stornieren.

Funktionsweise der Verzögerungsalarm-OBs

Nach Ablauf der Zeit, deren Wert in ms Sie zusammen mit einer OB-Nummer an die Anweisung "SRT_DINT" übergeben haben, startet das Betriebssystem den zugehörigen OB.

Wenn Sie Verzögerungsalarm-OBs verwenden möchten, dann müssen Sie die folgenden Aufgaben ausführen:

- Sie müssen die Anweisung "SRT_DINT" aufrufen.
- Sie müssen den Verzögerungsalarm-OB als Teil Ihres Programms in die CPU laden.

Verzögerungsalarm-OBs werden nur bearbeitet, wenn die CPU im Betriebszustand RUN ist. Ein Neustart (Warmstart) und ein Kaltstart löschen jedes Startereignis eines Verzögerungsalarm-OB. Einen Verzögerungsalarm, der noch nicht aktiviert ist, können Sie mit der Anweisung "CAN_DINT" stornieren.

Die Verzögerungszeit wird mit einer Genauigkeit von 1 ms gemessen. Eine Verzögerungszeit kann sofort nach ihrem Ablauf wieder gestartet werden. Mit Hilfe der Anweisung "QRY_DINT" können Sie den Zustand eines Verzögerungsalarms ermitteln.

Das Betriebssystem ruft einen OB für asynchrone Fehler auf, wenn eines der folgenden Ereignisse auftritt:

- Wenn das Betriebssystem einen OB zu starten versucht, der nicht geladen ist und dessen Nummer Sie beim Aufruf der Anweisung "SRT_DINT" angegeben haben.
- Wenn das nächste Startereignis für einen Verzögerungsalarm auftritt, bevor die Bearbeitung des zugehörigen Verzögerungsalarm-OBs beendet ist.

Sie können den Aufruf der Verzögerungsalarm-OBs mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten der Verzögerungsalarm-OBs

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen eines Verzögerungsalarm-OBs. Als Variablennamen wurden die Defaultnamen des OB 20 gewählt.

Variable	Datentyp	Deklaration	Beschreibung
OB20_EV_CLASS	BYTE	TEMP	Ereignisklasse und Kennungen: B#16#11: Alarm ist aktiv
OB20_STRT_INF	BYTE	TEMP	• B#16#21: Startanforderung für OB 20 • (B#16#22: Startanforderung für OB 21) • (B#16#23: Startanforderung für OB 22) • (B#16#24: Startanforderung für OB 23)
OB20_PRIORITY	BYTE	TEMP	Parametrierte Prioritätsklasse: 3 (OB 20) bis 6 (OB 23) (Defaultwerte) Defaultwert bei S7-1500-CPUs: 3
OB20_OB_NUMBR	BYTE	TEMP	OB-Nr. (20 bis 23)

Variable	Datentyp	Deklaration	Beschreibung
OB20_RESERVED_1	BYTE	TEMP	Reserviert
OB20_RESERVED_2	BYTE	TEMP	Reserviert
OB20_SIGN	WORD	TEMP	Anwenderkennung: Eingangsparameter SIGN aus dem Aufruf der Anweisung "SRT_DINT"
OB20_DTIME	TIME	TEMP	Parametrierte Verzögerungszeit in ms
OB20_DATE_TIME	DATE_AND_TIME	TEMP	Datum und Uhrzeit, zu denen der OB angefordert wurde

Weckalarm-OBs (OB 30 bis OB 38)

Beschreibung

S7 stellt bis zu neun Weckalarm-OBs (OB 30 bis OB 38) zur Verfügung. Mit ihrer Hilfe können Sie Programme nach äquidistanten Zeitabschnitten starten. Die folgende Tabelle zeigt die voreingestellten Werte für die Zeitraster und die Prioritätsklassen der Weckalarm-OBs.

Weckalarm-OB	Defaultwert für Zeitraster	Defaultwert für die Prioritätsklasse
OB 30	5 s	7
OB 31	2 s	8
OB 32	1 s	9
OB 33	500 ms	10
OB 34	200 ms	11
OB 35	100 ms	12
OB 36	50 ms	13
OB 37	20 ms	14
OB 38	10 ms	15

Funktionsweise der Weckalarm-OBs

Die äquidistanten Startzeitpunkte der Weckalarm-OBs ergeben sich aus dem jeweiligen Zeittakt und der jeweiligen Phasenverschiebung.

Hinweis

Sie müssen dafür sorgen, dass die Laufzeit jedes Weckalarm-OB deutlich kleiner ist als sein Zeittakt. Falls ein Weckalarm-OB noch nicht beendet ist, aber wegen des abgelaufenen Zeittakts erneut zur Bearbeitung ansteht, wird der Zeitfehler-OB (OB 80) gestartet. Anschließend wird der Fehler verursachende Weckalarm nachgeholt.

Sie können den Aufruf der Weckalarm-OBs mit Hilfe der Anweisung "DIS_IRT sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Die Wertebereiche für die Parameter Zeittakt, Prioritätsklasse und Phasenverschiebung sind den Technischen Daten Ihrer CPU zu entnehmen. Eine Änderung der Parameter erfolgt durch Projektierung.

Lokaldaten der Weckalarm-OBs

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen eines Weckalarm-OB.

Variable	Datentyp	Beschreibung
OB3x_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: Alarm ist aktiv
OB3x_STRT_INF	BYTE	• B#16#30: Sonderstartanforderung für einen Weckalarm-OB im H-System (parametrierte Sonderbehandlung beim Wechsel in den Systemzustand "Redundant") • B#16#31: Startanforderung für OB 30 … • B#16#36: Startanforderung für OB 35 … • B#16#39: Startanforderung für OB 38 • B#16#3A: Startanforderung für Weckalarm-OBs (OB 30 bis OB 38) mit Weckalarmtakt kleiner eine Millisekunde
OB3x_PRIORITY	BYTE	Parametrierte Prioritätsklasse; Defaultwerte: 7 (OB 30) bis 15 (OB 38) Defaultwerte bei S7-1500-CPUs: 8 bis 14; 16 bis 17
OB3x_OB_NUMBR	BYTE	OB-Nummer (30 bis 38)
OB3x_RESERVED_1	BYTE	Reserviert
OB3x_RESERVED_2	BYTE	Reserviert
OB3x_PHS_OFFSET	INT	für OBs 30 bis 34 und 36 bis 38: • falls OB3x_STRT_INF=B#16#3A: Phasenverschiebung in µs • in allen anderen Fällen: Phasenverschiebung in ms
OB35_PHASE_OFFSET	WORD	für OB 35: • falls OB35_STRT_INF=B#16#3A: Phasenverschiebung in µs • in allen anderen Fällen: Phasenverschiebung in ms
OB3x_RESERVED_3	INT	Reserviert
OB3x_EXC_FREQ	INT	• falls OB3x_STRT_INF=B#16#3A: Zeittakt in µs • in allen anderen Fällen: Zeittakt in ms
OB3x_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde.

Prozessalarm-OBs (OB 40 bis OB 47)

Beschreibung

S7 stellt bis zu acht voneinander unabhängige Prozessalarme mit je einem OB zur Verfügung.

Per Projektierung legen Sie für jede Signalbaugruppe, die Prozessalarme auslösen soll, fest,

- welche Kanäle bei welcher Randbedingung einen Prozessalarm auslösen sollen.
- welcher Prozessalarm-OB den einzelnen Kanalgruppen zugeordnet wird (defaultmäßig werden alle Prozessalarme durch OB 40 bearbeitet).

Bei CPs und FMs müssen Sie hierzu die entsprechende Software für die Baugruppe verwenden.

Die Prioritätsklassen für die einzelnen Prozessalarm-OBs legen Sie per Projektierung fest.

Funktionsweise der Prozessalarm-OBs

Nach Auslösen eines Prozessalarms durch die Baugruppe identifiziert das Betriebssystem den Steckplatz und ermittelt den zugehörigen Prozessalarm-OB. Hat dieser eine höhere Priorität als die momentan aktive Prioritätsklasse, dann wird er gestartet. Die kanalspezifische Quittierung erfolgt nach Beendigung dieses Prozessalarm-OB.

Tritt in der Zeit zwischen der Identifikation und der Quittierung eines Prozessalarms auf derselben Baugruppe erneut ein Ereignis auf, das einen Prozessalarm auslösen soll, gilt:

- Tritt das Ereignis bei dem Kanal auf, der vorher den Prozessalarm ausgelöst hat, so geht der zugehörige Alarm verloren. Das folgende Bild zeigt den Zusammenhang zwischen einem Prozesssignal und der Bearbeitung des zugehörigen Prozessalarm-OB am Beispiel eines Kanals einer Digitaleingabebaugruppe. Auslösendes Ereignis sei die steigende Flanke. Zugehöriger Prozessalarm-OB sei der OB 40.

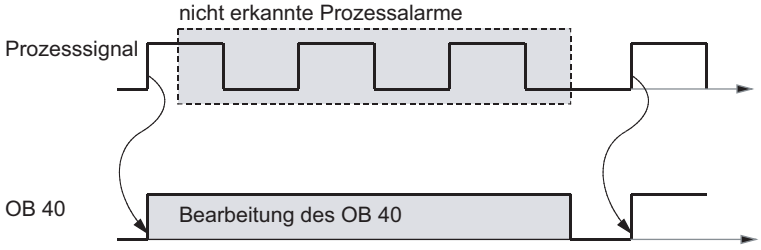

- Tritt das Ereignis bei einem anderen Kanal derselben Baugruppe auf, so kann momentan kein Prozessalarm ausgelöst werden. Er geht jedoch nicht verloren, sondern wird nach der Quittierung des gerade aktiven Prozessalarms ausgelöst.

Wird ein Prozessalarm ausgelöst, dessen zugehöriger OB momentan gerade aufgrund eines Prozessalarms einer anderen Baugruppe aktiv ist, so wird die erneute Anforderung registriert und der OB zu gegebener Zeit abgearbeitet.

Sie können den Aufruf der Prozessalarm-OBs mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Sie können die Parametrierung der Prozessalarme einer Baugruppe mit den Anweisungen "WR_PARM", " WR_DPARM" und "PARM_MOD" vornehmen.

Lokaldaten der Prozessalarm-OBs

Die folgende Tabelle beschreibt die temporären (TEMP) Variablen eines Prozessalarm-OB. Als Variablennamen wurden die Defaultnamen des OB 40 gewählt.

Variable	Datentyp	Beschreibung
OB40_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: Alarm ist aktiv
OB40_STRT_INF	BYTE	• B#16#41: Alarm über Interruptleitung 1 • B#16#42: Alarm über Interruptleitung 2 (nur bei S7-400) • B#16#43: Alarm über Interruptleitung 3 (nur bei S7-400) • B#16#44: Alarm über Interruptleitung 4 (nur bei S7-400) • B#16#45: WinAC: Alarm vom PC ausgelöst Hinweis: Die Interruptleitungen 1, ... 4 sind im Multicomputing-Betrieb den CPUs 1, ... 4 zugeordnet.
OB40_PRIORITY	BYTE	Parametrierte Prioritätsklasse: Defaultwerte: 16 (OB 40) bis 23 (OB 47) Defaultwert bei S7-1500-CPUs: 16
OB40_OB_NUMBR	BYTE	OB-Nummer (40 bis 47)
OB40_RESERVED_1	BYTE	Reserviert
OB40_IO_FLAG	BYTE	• B#16#54: Eingabebaugruppe • B#16#55: Ausgabebaugruppe • B#16#00: nicht verfügbar
OB40_MDL_ADDR	WORD	Logische Basisadresse der Baugruppe, die den Alarm auslöst
OB40_POINT_ADDR	DWORD	• Bei Digitalbaugruppen: Bitfeld mit denjenigen Eingängen auf der Baugruppe, die den Prozessalarm ausgelöst haben Die Zuordnung, welches Bit von OB40_POINT_ADDR zu welchem Kanal der Baugruppe gehört, ist der Beschreibung der jeweiligen Baugruppe zu entnehmen. • Bei Analogbaugruppen:Bitfeld mit der Information, welcher Kanal welche Grenze überschritten hat (Den genauen Aufbau entnehmen Sie bitte der Baugruppenbeschreibung). • Bei CPs oder IMs: Alarmzustand der Baugruppe (nicht anwenderrelevant)
OB40_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Hinweis

Falls Sie eine DPV1-fähige CPU einsetzen, können Sie mit Hilfe der Anweisung "RALRM" weitere Informationen über den Alarm erhalten, die über die Startinformation des OB hinausgehen. Dies gilt auch dann, wenn der DP-Master im S7-kompatiblen Mode betrieben wird.

Statusalarm-OB (OB 55)

Beschreibung

Das Betriebssystem der CPU ruft den OB 55 auf, wenn von einem Steckplatz eines DPV1-Slaves oder eines IO-Devices ein Statusalarm ausgelöst wurde. Dies kann der Fall sein, wenn eine Komponente (Baugruppe bzw. Modul) eines DPV1-Slaves oder eines IO-Devices ihren Betriebszustand wechselt, z. B. von RUN nach STOP. Die genauen Ereignisse, die einen Statusalarm auslösen, entnehmen Sie der Dokumentation des DPV1-Slave- oder IO-Device-Herstellers.

Hinweis

Wenn S7-400-CPUs bzw. S7-400-PROFIBUS-CPs auf den DP-Mode "S7-kompatibel" eingestellt sind, kann kein Statusalarm-OB (OB 55) genutzt werden.

Lokaldaten des Statusalarm-OB

Die folgende Tabelle enthält die temporären (TEMP) Variablen des Statusalarm-OB. Als Variablennamen wurden die Defaultnamen des OB 55 gewählt.

Variable	Datentyp	Beschreibung
OB55_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11 (kommendes Ereignis)
OB55_STRT_INF	BYTE	• B#16#55: Statusalarm bei DP • B#16#58: Statusalarm bei PROFINET IO
OB55_PRIORITY	BYTE	Parametrierte Prioritätsklasse, Defaultwert: 2 bzw. 4 (bei S7-1500-CPUs)
OB55_OB_NUMBR	BYTE	OB-Nummer (55)
OB55_RESERVED_1	BYTE	Reserviert
OB55_IO_FLAG	BYTE	Eingabebaugruppe/-modul: B#16#54 Ausgabebaugruppe/-modul: B#16#55
OB55_MDL_ADDR	WORD	Logische Basisadresse der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
OB55_LEN	BYTE	Länge des Datenblocks, den der Alarm liefert
OB55_TYPE	BYTE	Kennung für den Alarmtyp "Statusalarm"

Variable	Datentyp	Beschreibung
OB55_SLOT	BYTE	Steckplatz-Nr. der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
OB55_SPEC	BYTE	Specifier: • Bit 0 bis 1: Alarm-Specifier • Bit 2: Add_Ack • Bit 3 bis 7: Seq-Nr.
OB55_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Hinweis

Die in der obigen Tabelle angegebene Bedeutung von OB55_LEN, OB55_TYPE, OB55_SLOT, OB55_SPEC gilt nur für einen Statusalarm bei DP. Falls ein Statusalarm bei PROFINET IO vorliegt, müssen Sie die lokalen Variablen so organisieren wie in der nächsten Tabelle angegeben.

Hinweis

Die vollständige Alarmzusatzinformation aus dem Telegramm erhalten Sie durch Aufruf der Anweisung "RALRM" innerhalb des OB 55.

Falls Sie den OB 55 in Abhängigkeit von den Startereignissen programmieren wollen, wird empfohlen, die lokalen Variablen wie folgt zu organisieren:

Variable	Datentyp
OB55_EV_CLASS	BYTE
OB55_STRT_INF	BYTE
OB55_PRIORITY	BYTE
OB55_OB_NUMBR	BYTE
OB55_RESERVED_1	BYTE
OB55_IO_FLAG	BYTE
OB55_MDL_ADR	WORD
OB55_Z2	WORD
OB55_Z3	WORD
OB55_DATE_TIME	DATE_AND_TIME

Je nach Startereignis enthalten die Variablen OB55_Z2 und OB55_Z3 unterschiedliche Informationen. Dies wird im Folgenden näher erläutert.

Bedeutung von OB55_Z2

OB55_STRT_INF	Bedeutung von OB55_Z2
B#16#55	• low byte: Kennung für den Alarmtyp "Statusalarm" • high byte: Länge des Datenblocks, den der Alarm liefert
B#16#58	Kennung für den Alarmtyp: • W#16#0005: Status-Alarm

Bedeutung von OB55_Z3

OB55_STRT_INF	Bedeutung von OB55_Z3
B#16#55	• low byte: Specifier – Bits 0 bis 1: Alarm Specifier – Bit 2: Add_Ack – Bits 3 bis 7: Sequenznummer • high byte: Steckplatz der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
B#16#58	Alarm Specifier • Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) • Bits 11 bis 15: 0

Update-Alarm-OB (OB 56)

Beschreibung

Das Betriebssystem der CPU ruft den OB 56 auf, wenn von einem Steckplatz eines DPV1-Slaves oder eines IO-Devices ein Update-Alarm ausgelöst wurde. Dies kann der Fall sein, wenn Sie an einem Steckplatz eines DPV1-Slaves oder eines IO-Devices eine Parameteränderung durchgeführt haben (durch lokalen oder entfernten Zugriff). Die genauen Ereignisse, die einen Update-Alarm auslösen, entnehmen Sie der Dokumentation des DPV1-Slave- oder IO-Device-Herstellers.

Hinweis

Wenn S7-400-CPUs bzw. S7-400-PROFIBUS-CPs auf den DP-Mode "S7-kompatibel" eingestellt sind, kann kein Update-Alarm-OB (OB 56) genutzt werden.

Lokaldaten des Update-Alarm-OB

Die folgende Tabelle enthält die temporären (TEMP) Variablen des Update-Alarm-OB. Als Variablennamen wurden die Defaultnamen des OB 56 gewählt.

Variable	Datentyp	Beschreibung
OB56_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11 (kommendes Ereignis)
OB56_STRT_INF	BYTE	• B#16#56: Update-Alarm bei DP • B#16#59: Update-Alarm bei PROFINET IO
OB56_PRIORITY	BYTE	Parametrierte Prioritätsklasse, Defaultwert: 2 bzw. 4 (bei S7-1500-CPUs)
OB56_OB_NUMBR	BYTE	OB-Nummer (56)
OB56_RESERVED_1	BYTE	Reserviert
OB56_IO_FLAG	BYTE	Eingabebaugruppe/-modul: B#16#54 Ausgabebaugruppe/-modul: B#16#55
OB56_MDL_ADDR	WORD	Logische Basisadresse der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
OB56_LEN	BYTE	Länge des Datenblocks, den der Alarm liefert
OB56_TYPE	BYTE	Kennung für den Alarmtyp "Update-Alarm"
OB56_SLOT	BYTE	Steckplatz-Nr. der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
OB56_SPEC	BYTE	Specifier: • Bit 0 bis 1: Alarm-Specifier • Bit 2: Add_Ack • Bit 3 bis 7: Seq-Nr.
OB56_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Hinweis

Die in der obigen Tabelle angegebene Bedeutung von OB56_LEN, OB56_TYPE, OB56_SLOT, OB56_SPEC gilt nur für einen Update-Alarm bei DP. Falls ein Update-Alarm bei PROFINET IO vorliegt, müssen Sie die lokalen Variablen so organisieren wie in der nächsten Tabelle angegeben.

Hinweis

Die vollständige Alarmzusatzinformation aus dem Telegramm erhalten Sie durch Aufruf der Anweisung "RALRM" innerhalb des OB 56.

Falls Sie den OB 56 in Abhängigkeit von den Startereignissen programmieren wollen, wird empfohlen, die lokalen Variablen wie folgt zu organisieren:

Variable	Datentyp
OB56_EV_CLASS	BYTE
OB56_STRT_INF	BYTE
OB56_PRIORITY	BYTE
OB56_OB_NUMBR	BYTE
OB56_RESERVED_1	BYTE
OB56_IO_FLAG	BYTE
OB56_MDL_ADR	WORD
OB56_Z2	WORD
OB56_Z3	WORD
OB56_DATE_TIME	DATE_AND_TIME

Je nach Startereignis enthalten die Variablen OB56_Z2 und OB56_Z3 unterschiedliche Informationen. Dies wird im Folgenden näher erläutert.

Bedeutung von OB56_Z2

OB56_STRT_INF	Bedeutung von OB56_Z2
B#16#56	• low byte: Kennung für den Alarmtyp "Update-Alarm" • high byte: Länge des Datenblocks, den der Alarm liefert
B#16#59	Kennung für den Alarmtyp • W#16#0006: Update-Alarm • W#16#001E: Upload-and-Retrieval-Alarm Im Siemens Industry Online Support wird Ihnen in Form eines Anwendungsbeispiels ein Funktionsbaustein IPARSERV zur Verfügung gestellt, mit dem sich zusätzliche Parametrierdaten auf Anforderung eines PROFINET-Geräts oder eines darin verwendeten Moduls in einem DB abspeichern und auf Grund von Wartung oder Reparatur wiederherstellen lassen. Ein PROFINET-Gerät sendet einen Upload-and-Retrieval-Alarm an den IO-Controller und nutzt dazu den Update-Alarm-OB. https://support.industry.siemens.com/cs/ww/de/view/45841087 (https://support.industry.siemens.com/cs/ww/de/view/45841087)

Bedeutung von OB56_Z3

OB56_STRT_INF	Bedeutung von OB56_Z3
B#16#56	• low byte: Specifier – Bits 0 bis 1: Alarm Specifier – Bit 2: Add_Ack – Bits 3 bis 7: Sequenznummer • high byte: Steckplatz der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
B#16#59	Alarm Specifier: • Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) • Bits 11 bis 15: 0

OB für herstellerspezifische Alarme (OB 57)

Beschreibung

Das Betriebssystem der CPU ruft den OB 57 auf, wenn von einem Steckplatz eines DPV1-Slaves oder eines IO-Devices ein herstellerspezifischer Alarm ausgelöst wurde.

Hinweis

Wenn S7-400-CPUs bzw. S7-400-PROFIBUS-CPs auf den DP-Mode "S7-kompatibel" eingestellt sind, kann kein herstellerspezifischer Alarm-OB (OB 57) genutzt werden.

Lokaldaten des OB für herstellerspezifische Alarme

Die folgende Tabelle enthält die temporären (TEMP) Variablen des OB für herstellerspezifische Alarme. Als Variablennamen wurden die Defaultnamen des OB 57 gewählt.

Variable	Datentyp	Beschreibung
OB57_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11 (kommendes Ereignis)
OB57_STRT_INF	BYTE	• B#16#57: Manufacture Alarm bei DP • B#16#5A: Manufacture Alarm bei PROFINET IO • B#16#5B: IO: Profile Specific Alarm
OB57_PRIORITY	BYTE	Parametrierte Prioritätsklasse, Defaultwert: 2 bzw. 4 (bei S7-1500-CPUs)
OB57_OB_NUMBR	BYTE	OB-Nummer (57)
OB57_RESERVED_1	BYTE	Reserviert
OB57_IO_FLAG	BYTE	Eingabebaugruppe/-modul: B#16#54 Ausgabebaugruppe/-modul: B#16#55
OB57_MDL_ADDR	WORD	Logische Basisadresse der Alarm auslösenden Komponente (Baugruppe bzw. Modul)

Variable	Datentyp	Beschreibung
OB57_LEN	BYTE	Länge des Datenblocks, den der Alarm liefert
OB57_TYPE	BYTE	Kennung für den Alarmtyp "herstellerspezifischer Alarm"
OB57_SLOT	BYTE	Steckplatz-Nr. der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
OB57_SPEC	BYTE	Specifier: • Bit 0 bis 1: Alarm-Specifier • Bit 2: Add_Ack • Bit 3 bis 7: Seq-Nr.
OB57_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Hinweis

Die in der obigen Tabelle angegebene Bedeutung von OB57_LEN, OB57_TYPE, OB57_SLOT, OB57_SPEC gilt nur für einen herstellerspezifischen Alarm bei DP. Falls ein herstellerspezifischer Alarm bei PROFINET IO vorliegt, müssen Sie die lokalen Variablen so organisieren wie in der nächsten Tabelle angegeben.

Hinweis

Die vollständige Alarmzusatzinformation aus dem Telegramm erhalten Sie durch Aufruf der Anweisung "RALRM" innerhalb des OB 57.

Falls Sie den OB 57 in Abhängigkeit von den Startereignissen programmieren wollen, wird empfohlen, die lokalen Variablen wie folgt zu organisieren:

Variable	Datentyp
OB57_EV_CLASS	BYTE
OB57_STRT_INF	BYTE
OB57_PRIORITY	BYTE
OB57_OB_NUMBR	BYTE
OB57_RESERVED_1	BYTE
OB57_IO_FLAG	BYTE
OB57_MDL_ADR	WORD
OB57_Z2	WORD
OB57_Z3	WORD
OB57_DATE_TIME	DATE_AND_TIME

Je nach Startereignis enthalten die Variablen OB57_Z2 und OB57_Z3 unterschiedliche Informationen. Dies wird im Folgenden näher erläutert.

Bedeutung von OB57_Z2

OB57_STRT_INF	Bedeutung von OB57_Z2
B#16#57	• low byte: Kennung für den Alarmtyp "Herstellerspezifischer Alarm" • high byte: Länge des Datenblocks, den der Alarm liefert
B#16#5A	Kennung für den Alarmtyp: • W#16#0020 bis 007F: Herstellerspezifischer Alarm

Bedeutung von OB57_Z3

OB57_STRT_INF	Bedeutung von OB57_Z3
B#16#57	• low byte: Specifier – Bits 0 bis 1: Alarm Specifier – Bit 2: Add_Ack – Bits 3 bis 7: Sequenznummer • high byte: Steckplatz der Alarm auslösenden Komponente (Baugruppe bzw. Modul)
B#16#5A	Alarm Specifier • Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) • Bits 11 bis 15: 0

Taktsynchronalarm-OBs (OB 61 bis OB 64)

Beschreibung

Taktsynchronalarme bieten die Möglichkeit, Programme taktsynchron zum DP-Takt bzw. PN-Sendetakt zu starten. Zum Taktsynchronalarm TSAL y gehört der OB 6y , 1 <= y <= 4, als Schnittstellen-OB. Sie können die Priorität der OBs 61 bis 64 zwischen 0 (OB abgewählt) und 2 bis 26 einstellen.

⚠ VORSICHT

Direktzugriffe

Vermeiden Sie bei Direktzugriffen sowohl mit L- bzw. T-Befehlen (z. B. L PEB, T PAB) als auch bei Verwendung der Anweisung "DPRD_DAT" und "DPWR_DAT" Zugriffe auf Peripheriebereiche, denen Teilprozessabbilder mit OB6x-Anbindung (Taktsynchronalarme) zugeordnet sind.

Lokaldaten des Taktsynchronalarm-OB

Folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Taktsynchronalarm-OB. Als Variablennamen wurden die Defaultnamen des OB 61 gewählt.

Variable	Datentyp	Beschreibung
OB61_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: Alarm ist aktiv
OB61_STRT_INF	BYTE	B#16#64: Startanforderung für OB 61 : B#16#67: Startanforderung für OB 64
OB61_PRIORITY	BYTE	Parametrierte Prioritätsklasse; Defaultwert: 25 bzw. 21 (bei S7-1500-CPUs)
OB61_OB_NUMBR	BYTE	OB-Nummer: 61 ... 64
OB61_RESERVED_1	BYTE	Reserviert
OB61_RESERVED_2	BYTE	Reserviert
OB61_GC_VIOL	BOOL	GC-Verletzung bei PROFIBUS DP
OB61_FIRST	BOOL	Erste Ausführung nach Anlauf bzw. Haltzustand
OB61_MISSED_EXEC	BYTE	Anzahl der ausgefallenen OB 61-Starts seit der letzten OB 61-Ausführung
OB61_DP_ID	BYTE	DP-Mastersystem-ID des taktsynchronen DP-Mastersystems (1 bis 32) bzw. PROFINET IO System-ID des taktsynchronen PNIO-Systems (100 bis 115)
OB61_RESERVED_3	BYTE	Reserviert
OB61_RESERVED_4	WORD	Reserviert
OB61_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Zeitfehler-OB (OB 80)

Beschreibung

Das Betriebssystem der CPU ruft den OB 80 auf, wenn bei der Bearbeitung eines OB einer der folgenden Fehler auftritt:

- Überschreiten der Zykluszeit
- Quittierungsfehler bei der Bearbeitung eines OB
- Vorstellen der Uhrzeit (Uhrzeitsprung) zum Starten eines OB

Tritt beispielsweise ein Startereignis für einen Weckalarm-OB auf, bevor die vorherige Bearbeitung desselben OB beendet ist, dann ruft das Betriebssystem den OB 80 auf.

Wurde der OB 80 nicht programmiert, dann wechselt die CPU in den Betriebszustand STOP.

Sie können den Aufruf des Zeitfehler-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Hinweis

Wenn der OB 80 in demselben Zyklus zweimal aufgrund der Zykluszeitüberschreitung aufgerufen wird, geht die CPU in STOP. Sie können dies durch Aufruf der Anweisung "RE_TRIGR" an geeigneter Stelle verhindern.

Hinweis zur Verwendung von "Report System Errors"

"Report System Errors" verzögert die Bearbeitung von Alarmen, die während seiner Laufzeit auftreten.

Wenn die Bedingung **Laufzeit von "Report System Errors" > 50 % der Zykluszeit** erfüllt ist, wird dieser Effekt verstärkt (Dabei wird unter der Laufzeit von "Report System Errors" die Summe aller Bearbeitungszeiten von "Report System Errors" innerhalb der Zykluszeit verstanden.). Dies kann zum Aufruf des Zeitfehler-OB und evtl. zum STOP der CPU führen (wenn der Zeitfehler-OB nicht geladen ist).

Um dies zu verhindern, gibt es die folgenden Maßnahmen:

- Wenn Sie Taktsynchronalarm-OBs (OB61 bis OB64) verwenden, müssen Sie dafür sorgen, dass diese pro Zeiteinheit weniger häufig aufgerufen werden als bisher. Dies erreichen Sie, indem Sie für jeden Taktsynchronalarm-OB den zugehörigen Parameter "Applikationszyklusfaktor" erhöhen.
- Vergrößern Sie das Verhältnis (Zykluszeit - Laufzeit von "Report System Errors")/(Laufzeit von "Report System Errors"). Dazu gibt es die folgenden beiden Möglichkeiten:
  - Wenn Sie "Report System Errors" in Weckalarm-OBs (OB30 bis OB38) aufrufen, dann erhöhen Sie für die zugehörigen OBs die Zeittakte.
  - Erhöhen Sie im OB1 die Laufzeit des von "Report System Errors" unabhängigen Programmabschnitts. Dies erreichen Sie z. B. durch Einsatz der Anweisung "WAIT".

Lokaldaten des Zeitfehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Zeitfehler-OB. Als Variablennamen wurden die Defaultnamen des OB 80 gewählt.

Variable	Datentyp	Beschreibung
OB80_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#35
OB80_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#01, B#16#02, B#16#05, B#16#06, B#16#07, B#16#08, B#16#09, B#16#0A, B#16#0B)
OB80_PRIORITY	BYTE	Prioritätsklasse: Der OB 80 läuft im RUN in Prioritätsklasse 26 und bei Überlauf des OB-Anforderungspuffers in Prioritätsklasse 28 Bei S7-1500-CPUs: Priorität 22
OB80_OB_NUMBR	BYTE	OB-Nummer (80)

Variable	Datentyp	Beschreibung
OB80_RESERVED_1	BYTE	Reserviert
OB80_RESERVED_2	BYTE	Reserviert
OB80_ERROR_INFO	WORD	Fehlerinformation: abhängig vom Fehlercode
OB80_ERR_EV_CLASS	BYTE	Klasse des Ereignisses, das den Fehler ausgelöst hat
OB80_ERR_EV_NUM	BYTE	Nummer des Ereignisses, das den Fehler ausgelöst hat
OB80_OB_PRIORITY	BYTE	Fehlerinformation: abhängig vom Fehlercode
OB80_OB_NUM	BYTE	Fehlerinformation: abhängig vom Fehlercode
OB80_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die vom Fehlercode abhängigen Variablen haben folgende Bedeutung:

Fehlercode	Variable	Bit	Bedeutung
B#16#01			Zykluszeit überschritten
	OB80_ERROR_INFO		Laufzeit des letzten Zyklus (ms)
	OB80_ERR_EV_CLASS		Klasse des Ereignisses, das den Alarm ausgelöst hat
	OB80_ERR_EV_NUM		Nummer des Ereignisses, das den Alarm ausgelöst hat
	OB80_OB_PRIORITY		Prioritätsklasse des OB, der bearbeitet wurde, als der Fehler auftrat
	OB80_OB_NUM		Nummer des OB, der bearbeitet wurde, als der Fehler auftrat
B#16#02			Der angeforderte OB ist noch in Bearbeitung.
	OB80_ERROR_INFO		Die zugehörige temporäre Variable des angeforderten OB. Dieser ist bestimmt durch OB80_ERR_EV_CLASS und OB80_ERR_EV_NUM.
	OB80_ERR_EV_CLASS		Klasse des Ereignisses, das den Alarm ausgelöst hat
	OB80_ERR_EV_NUM		Nummer des Ereignisses, das den Alarm ausgelöst hat
	OB80_OB_PRIORITY		Prioritätsklasse des Fehler verursachenden OB (z. B.: "7" für OB 30/Prioritätsklasse 7, der gestartet werden sollte, aber nicht gestartet werden konnte)
	OB80_OB_NUM		Nummer des Fehler verursachenden OB (z. B.: "30" für OB 30, der gestartet werden sollte, aber nicht gestartet werden konnte)
B#16#05			Abgelaufener Uhrzeitalarm durch Uhrzeitsprung
	OB80_ERROR_INFO	Bit 0 gesetzt	Für den Uhrzeitalarm 0 liegt der Startzeitpunkt in der Vergangenheit
		...	
		Bit 7 gesetzt	Für den Uhrzeitalarm 7 liegt der Startzeitpunkt in der Vergangenheit
		Bit 8 bis 15	Nicht verwendet
	OB80_ERR_EV_CLASS		Nicht verwendet
	OB80_ERR_EV_NUM		Nicht verwendet
	OB80_OB_PRIORITY		Nicht verwendet
	OB80_OB_NUM		Nicht verwendet

Fehlercode	Variable	Bit	Bedeutung
B#16#06	wie beim Fehlercode B#16#05		Abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN nach HALT
B#16#07	wie beim Fehlercode B#16#02		Überlauf des OB-Anforderungspuffers für die aktuelle Prioritätsklasse (Jede OB-Startanforderung für eine Prioritätsklasse wird in den zugehörigen OB-Anforderungspuffer eingetragen; nach Beendigung des OB wird der Eintrag wieder gelöscht. Falls für eine Prioritätsklasse mehr OB-Startanforderungen vorliegen als die maximal mögliche Anzahl der Einträge im zugehörigen OB-Anforderungspuffer, wird der OB 80 mit dem Fehlercode B#16#07 aufgerufen.)
B#16#08	wie beim Fehlercode B#16#02		Taktsynchronalarm-Zeitfehler
B#16#09	wie beim Fehlercode B#16#02		Alarmverlust durch zu hohe Alarmlast
B#16#0A			Wiedereintritt in RUN nach CiR
	OB80_ERROR_INFO		CiR-Synchronisationszeit in ms
B#16#0B			Technologiesynchronalarm-Zeitfehler
	OB80_ERR_EV_NUM		Nummer des Ereignisses, das den Alarm ausgelöst hat: W#16#116A
	OB80_OB_PRIORITY		Prioritätsklasse des OB, der bearbeitet wurde, als der Fehler auftrat
	OB80_OB_NUM		Nummer des OB, der bearbeitet wurde, als der Fehler auftrat: 65
B#16#0C			(nur bei S7-1500-R/H-CPUs) Zykluszeit zum zweiten Mal überschritten
	OB80_ERROR_INFO		Laufzeit des letzten Zyklus (ms)
	OB80_ERR_EV_CLASS		Klasse des Ereignisses, das den Alarm ausgelöst hat
	OB80_ERR_EV_NUM		Nummer des Ereignisses, das den Alarm ausgelöst hat
	OB80_OB_PRIORITY		Prioritätsklasse des OB, der bearbeitet wurde, als der Fehler auftrat
	OB80_OB_NUM		Nummer des OB, der bearbeitet wurde, als der Fehler auftrat

Stromversorgungsfehler-OB (OB 81)

Beschreibung

Das Betriebssystem der CPU ruft den OB 81 auf, wenn ein Ereignis auftritt, das durch einen Fehler in der Stromversorgung (nur bei S7-400) oder der Pufferung ausgelöst wird (sowohl bei kommendem als auch bei gehendem Ereignis).

Bei S7-400 wird der OB 81 bei Batteriefehler nur dann aufgerufen, wenn die Batterieprüfung mit dem Schalter BATT.INDIC aktiviert ist.

Die CPU geht nicht in den Betriebszustand STOP, wenn der OB 81 nicht programmiert ist.

Sie können den Aufruf des Stromversorgungsfehler-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten des Stromversorgungsfehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Stromversorgungsfehler-OB. Als Variablennamen wurden die Defaultnamen des OB 81 gewählt.

Variable	Datentyp	Beschreibung
OB81_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#38: gehendes Ereignis • B#16#39: kommendes Ereignis
OB81_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#21, B#16#22, B#16#23, B#16#25, B#16#26, B#16#27, B#16#31, B#16#32, B#16#33)
OB81_PRIORITY	BYTE	Prioritätsklasse: einstellbar per Projektierung Z.B. sind für den Betriebszustand RUN die Werte 2 bis 26 einstellbar.
OB81_OB_NUMBR	BYTE	OB-Nummer (81)
OB81_RESERVED_1	BYTE	Reserviert
OB81_RESERVED_2	BYTE	Reserviert
OB81_RACK_CPU	WORD	• Bits 0 bis 7: B#16#00 • Bits 8 bis 15: – bei einer Standard-CPU: B#16#00 – bei einer H-CPU: Bits 8 bis 10: Baugruppenträger-Nr., Bit 11: 0=Reserve-CPU, 1=Master-CPU, Bits 12 bis 15: 1111
OB81_RESERVED_3	BYTE	Nur relevant für die Fehlercodes B#16#31, B#16#32, B#16#33
OB81_RESERVED_4	BYTE	Nur relevant für die Fehlercodes B#16#31, B#16#32, B#16#33
OB81_RESERVED_5	BYTE	Nur relevant für die Fehlercodes B#16#31, B#16#32, B#16#33
OB81_RESERVED_6	BYTE	Nur relevant für die Fehlercodes B#16#31, B#16#32, B#16#33
OB81_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die Variablen OB81_RESERVED_i, 3 ≤ i ≤ 6, enthalten diejenigen Erweiterungsgeräte, bei denen die Batteriepufferung (bei Fehlercode B#16#31), die Pufferspannung (bei Fehlercode B#16#32) oder die 24V-Versorgung (bei Fehlercode B#16#33) ausgefallen bzw. wiedergekommen ist. Die folgende Tabelle zeigt, welches Bit in den Variablen OB81_RESERVED_i, 3 ≤ i ≤ 6, zu welchem Erweiterungsgerät gehört.

	OB81_RESER-VED_6	OB81_RESER-VED_5	OB81_RESER-VED_4	OB81_RESERVED_3
Bit 0	Reserviert	8. Erw.-Gerät	16. Erw.-Gerät	Reserviert
Bit 1	1. Erw.-Gerät	9. Erw.-Gerät	17. Erw.-Gerät	Reserviert.
Bit 2	2. Erw.-Gerät	10. Erw.-Gerät	18. Erw.-Gerät	Reserviert
Bit 3	3. Erw.-Gerät	11. Erw.-Gerät	19. Erw.-Gerät	Reserviert

	OB81_RESER-VED_6	OB81_RESER-VED_5	OB81_RESER-VED_4	OB81_RESERVED_3
Bit 4	4. Erw.-Gerät	12. Erw.-Gerät	20. Erw.-Gerät	Reserviert
Bit 5	5. Erw.-Gerät	13. Erw.-Gerät	21. Erw.-Gerät	Reserviert
Bit 6	6. Erw.-Gerät	14. Erw.-Gerät	Reserviert	Reserviert
Bit 7	7. Erw.-Gerät	15. Erw.-Gerät	Reserviert	Reserviert

Die Bits in den Variablen OB81_RESERVED_i haben folgende Bedeutung (für das betroffene Erweiterungsgerät):

Beim kommenden Ereignis werden die Erweiterungsgeräte markiert (die zugehörigen Bits sind gesetzt), bei denen mindestens eine Batterie bzw. Pufferspannung bzw. die 24V-Versorgung ausgefallen ist. Erweiterungsgeräte, bei denen bereits früher mindestens eine Batterie bzw. eine Pufferspannung bzw. die 24V-Versorgung ausgefallen ist, werden nicht mehr angezeigt. Beim gehenden Ereignis wird eine wiedergekommene Pufferung in mindestens einem Erweiterungsgerät gemeldet (die zugehörigen Bits sind gesetzt).

Die folgende Tabelle zeigt, welches Ereignis den Start des OB 81 verursacht hat.

OB81_EV_CLASS	OB81_FLT_ID	Bedeutung
B#16#39/B#16#38	B#16#21	Mindestens eine Pufferbatterie des Zentralgeräts leer/beseitigt (BATTF). **Hinweis:** Das kommende Ereignis tritt nur beim Ausfall einer der beiden Batterien (bei redundanten Pufferbatterien) auf. Fällt anschließend auch noch die andere Batterie aus, tritt das Ereignis nicht nochmals auf.
B#16#39/B#16#38	B#16#22	Pufferspannung im Zentralgerät fehlt/beseitigt (BAF).
B#16#39/B#16#38	B#16#23	Ausfall der 24V-Versorgung im Zentralgerät /beseitigt.
B#16#39/B#16#38	B#16#25	Mindestens eine Pufferbatterie in mindestens einem redundanten Zentralgerät leer/beseitigt (BATTF).
B#16#39/B#16#38	B#16#26	Pufferspannung in mindestens einem redundanten Zentralgerät fehlt/beseitigt (BAF).
B#16#39/B#16#38	B#16#27	Ausfall der 24V-Versorgung in mindestens einem redundanten Zentralgerät/beseitigt.
B#16#39/B#16#38	B#16#31	Mindestens eine Pufferbatterie in mindestens einem Erweiterungsgerät leer/beseitigt (BATTF).
B#16#39/B#16#38	B#16#32	Pufferspannung in mindestens einem Erweiterungsgerät fehlt/beseitigt (BAF).
B#16#39/B#16#38	B#16#33	Ausfall der 24V-Versorgung in mindestens einem Erweiterungsgerät /beseitigt.

Beispielprogramm für den OB 81

Anhand eines AWL-Beispielprogramms wird gezeigt, wie Sie den Fehlercode im OB 81 lesen können.

Das Programm ist wie folgt aufgebaut:

Der Fehlercode in OB 81 (OB81_FLT_ID) wird gelesen und verglichen mit der Kennung für die Ereignisse "Mindestens eine Pufferbatterie des Zentralgeräts leer" (B#16#21) und "Pufferspannung im Zentralgerät fehlt" (B#16#22).

Wenn der Fehlercode einem der genannten Ereignisse entspricht, dann springt das Programm zur Marke BF. Andernfalls wird der Baustein beendet.

Ab der Marke BF setzt das Programm die Variable *Batteriefehler*, wenn ein kommendes Ereignis vorliegt. Wenn ein gehendes Ereignis vorliegt, dann setzt das Programm diese Variable zurück.

```
L    B#16#21                                  //Kennung für das Ereignis "Mindestens eine Puffer-
                                              batterie des Zentralgeräts leer"
L    #OB81_FLT_ID                             //Fehlercode OB 81
==I                                           //Wenn gleich,
SPB    BF                                     //dann springe zu BF
L    B#16#22                                  //Kennung für das Ereignis "Pufferspannung im Zent-
                                              ralgerät fehlt"
==I                                           //Wenn gleich mit dem Fehlercode OB 81,
SPB    BF                                     //dann springe zu BF
BEA                                           //keine Meldung über Batteriefehler

BF:    L    B#16#39                           //Kennung für kommendes Ereignis
L    #OB81_EV_CLASS                           //Ereignisklasse und Kennungen beim Aufruf des OB
                                              81
==I                                           //Wenn gleich,
S    Batteriefehler                           //dann setze Batteriefehler (Variable aus der Va-
                                              riablentabelle)
L    B#16#38                                  //Kennung für gehendes Ereignis
==I                                           //Wenn gleich mit Ereignisklasse und Kennungen
                                              beim Aufruf des OB 81,
R    Batteriefehler                           //dann rücksetze Batteriefehler (Variable aus der
                                              Variablentabelle)
```

Diagnosealarm-OB (OB 82)

Beschreibung

Wenn eine diagnosefähige Baugruppe, bei der Sie den Diagnosealarm freigegeben haben, eine Änderung ihres Diagnosezustands erkennt, stellt sie eine Diagnosealarmanforderung an die CPU:

- Es liegt eine Störung vor oder eine Komponente muss gewartet werden oder beides (kommendes Ereignis).
- Es liegt keine Störung mehr vor, und keine Komponente muss mehr gewartet werden (gehendes Ereignis).

Daraufhin ruft das Betriebssystem den OB 82 auf.

Der OB 82 enthält in seinen lokalen Variablen die logische Basisadresse sowie eine vier Byte lange Diagnoseinformation der fehlerhaften Baugruppe (siehe folgende Tabelle).

Haben Sie den OB 82 nicht programmiert, geht die CPU in den Betriebszustand STOP.

Sie können den Aufruf des Diagnosealarm-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Hinweis

Bei PROFINET IO-Controllern können Sie per Projektierung vorgeben, ob Kommunikationsalarme zum Aufruf des Diagnosealarm-OB führen sollen oder nicht. Bei diesen Alarmen handelt es sich um Diagnoseereignisse der PROFINET-Schnittstelle.

Voreinstellung ist, dass diese Ereignisse keinen OB 82-Aufruf zur Folge haben.

Lokaldaten des Diagnosealarm-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Diagnosealarm-OB. Als Variablennamen wurden die Defaultnamen des OB 82 gewählt.

Variable	Datentyp	Beschreibung
OB82_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#38: gehendes Ereignis: Baugruppe fehlerfrei (Alle Fehler sind beseitigt.) • B#16#39: kommendes Ereignis: Baugruppe wird fehlerhaft oder bleibt fehlerhaft (mit Wechsel der Anzahl ihrer anstehenden Fehler)
OB82_FLT_ID	BYTE	Fehlercode (B#16#42)
OB82_PRIORITY	BYTE	• Prioritätsklasse; einstellbar per Projektierung • Bei S7-1500-CPUs: Priorität, Defaultwert: 5
OB82_OB_NUMBR	BYTE	OB-Nummer (82)
OB82_RESERVED_1	BYTE	Reserviert
OB82_IO_FLAG	BYTE	• bei S7-300/S7-400: – Eingabebaugruppe: B#16#54 – Ausgabebaugruppe: B#16#55 • bei S7-1500: Null
OB82_MDL_ADDR	WORD	• bei S7-300/S7-400: Logische Basisadresse der Baugruppe, in der der Fehler aufgetreten ist • bei S7-1500: Hardware-Kennung des Hardware-Objekts, das den Diagnosealarm ausgelöst hat
OB82_MDL_DEFECT	BOOL	Baugruppenstörung
OB82_INT_FAULT	BOOL	Interner Fehler
OB82_EXT_FAULT	BOOL	Externer Fehler
OB82_PNT_INFO	BOOL	Kanalfehler vorhanden
OB82_EXT_VOLTAGE	BOOL	Externe Hilfsspannung fehlt
OB82_FLD_CONNCTR	BOOL	Frontstecker fehlt

Variable	Datentyp	Beschreibung
OB82_NO_CONFIG	BOOL	Baugruppe nicht parametriert
OB82_CONFIG_ERR	BOOL	Falsche Parameter in Baugruppe
OB82_MDL_TYPE	BYTE	• Bit 0 bis 3: Baugruppenklasse • Bit 4: Kanalinformation vorhanden • Bit 5: Anwenderinformation vorhanden • Bit 6: Diagnosealarm von Stellvertreter • Bit 7: Wartungsbedarf
OB82_SUB_MDL_ERR	BOOL	Anwendermodul falsch / fehlt
OB82_COMM_FAULT	BOOL	Kommunikationsstörung
OB82_MDL_STOP	BOOL	Betriebszustand (0: RUN, 1: STOP)
OB82_WTCH_DOG_FLT	BOOL	Zeitüberwachung hat angesprochen
OB82_INT_PS_FLT	BOOL	Baugruppeninterne Versorgungsspannung ausgefallen
OB82_PRIM_BATT_FLT	BOOL	Batterie leer
OB82_BCKUP_BATT_FLT	BOOL	Gesamte Pufferung ausgefallen
OB82_RESERVED_2	BOOL	Wartungsanforderung
OB82_RACK_FLT	BOOL	Erweiterungsgeräteausfall
OB82_PROC_FLT	BOOL	Prozessorausfall
OB82_EPROM_FLT	BOOL	EPROM-Fehler
OB82_RAM_FLT	BOOL	RAM-Fehler
OB82_ADU_FLT	BOOL	ADU/DAU-Fehler
OB82_FUSE_FLT	BOOL	Sicherungsausfall
OB82_HW_INTR_FLT	BOOL	Prozessalarm verloren
OB82_RESERVED_3	BOOL	Reserviert
OB82_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Hinweis

Falls Sie eine DPV1-fähige CPU einsetzen, können Sie mit Hilfe der Anweisung "RALRM" weitere Informationen über den Alarm erhalten, die über die Startinformation des OB hinausgehen. Dies gilt auch dann, wenn der DP-Master im S7-kompatiblen Mode betrieben wird.

Programmieren des OB 82

Den OB 82 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 82 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 82 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 82 aus.
- Sie nehmen eine genaue Diagnose des aufgetretenen Fehlers vor.

Wenn ein Diagnosealarm ausgelöst wird, trägt die gestörte Baugruppe automatisch 4 Byte Diagnosedaten und ihre Anfangsadresse in die Startinformation des Diagnosealarm-OB und in den Diagnosepuffer ein. Damit erhalten Sie die Information, auf welcher Baugruppe und wann ein Fehler aufgetreten ist.

Weitere Diagnosedaten der gestörten Baugruppe (auf welchem Kanal der Fehler aufgetreten ist, um welchen Fehler es sich handelt) können Sie mit einem entsprechenden Programm im OB 82 auswerten. Mit der Anweisung "RDSYSST" können Sie Baugruppendiagnosedaten auslesen und mit der Anweisung "WR_USMSG" diese Information in den Diagnosepuffer eintragen. Außerdem können Sie zusätzlich die selbst definierte Diagnosemeldung an ein angemeldetes Beobachtungsgerät senden.

Ziehen/Stecken-OB (OB 83)

Beschreibung

Das Betriebssystem der CPU ruft den OB 83 auf, nachdem eine projektierte Baugruppe gezogen oder gesteckt wurde.

Wenn Sie den OB 83 nicht programmiert haben, geht die CPU in den Betriebszustand STOP.

Sie können den Ziehen/Stecken-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Ziehen und Stecken von Baugruppen

Jedes Ziehen und Stecken einer projektierten Baugruppe (nicht erlaubt: Stromversorgungsbaugruppen, CPUs, Adaptionskapseln und IMs) in den Betriebszuständen RUN, STOP und ANLAUF führt zu einem Ziehen/Stecken-Alarm. Dieser Alarm führt bei der zugehörigen CPU zu je einem Eintrag in den Diagnosepuffer und die Systemzustandsliste. Darüber hinaus wird im Betriebszustand RUN der Ziehen/Stecken-OB gestartet. Haben Sie diesen OB nicht programmiert, geht die CPU in den Betriebszustand STOP.

Das Ziehen und Stecken von S7-400-Baugruppen wird innerhalb des Systems sekündlich überwacht. Damit das Ziehen und Stecken von der CPU erkannt wird, muss zwischen dem Ziehen und Stecken einer S7-400-Baugruppe eine Mindestzeit von zwei Sekunden liegen. Bei anderen Baugruppen ist diese Mindestzeit etwas größer.

Beim Ziehen einer projektierten Baugruppe im Betriebszustand RUN wird der OB 83 gestartet. Zuvor kann ein Zugriffsfehler beim Direktzugriff oder der Prozessabbildaktualisierung erkannt werden.

Beim Stecken einer Baugruppe in einen projektierten Steckplatz im Zustand RUN überprüft das Betriebssystem, ob der Baugruppentyp der gesteckten Baugruppe mit der Projektierung übereinstimmt. Anschließend wird der OB 83 gestartet, und bei Übereinstimmung der Baugruppentypen erfolgt die Parametrierung.

Besonderheiten bei S7-300, ET 200S und ET 200Pro

- Das Ziehen und Stecken von zentraler Peripherie ist bei S7-300 nicht zulässig.
- Bei S7-300-CPUs gibt es den Ziehen/Stecken-Alarm nur für die CPUs 31x PN/DP, und zwar nur für die PROFINET IO-Komponenten.
- ET 200Pro mit z. B. IM 154-4-8 CPU gibt es den Ziehen/Stecken-Alarm nur für die zentrale Peripherie.
- ET 200S mit z. B. IM151-8 PN/DP CPU gibt es den Ziehen/Stecken-Alarm nur für die zentrale Peripherie.

Lokaldaten des Ziehen/Stecken-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Ziehen/Stecken-OB. Als Variablennamen wurden die Defaultnamen des OB 83 gewählt.

Variable	Datentyp	Beschreibung
OB83_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#32: Ende Umparametrieren der Baugruppe • B#16#33: Beginn Umparametrieren der Baugruppe • B#16#38: Baugruppe gesteckt • B#16#39: Baugruppe gezogen bzw. nicht ansprechbar bzw. Ende Umparametrieren
OB83_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58, B#16#61, B#16#63, B#16#64, B#16#65, B#16#66, B#16#67, B#16#68)
OB83_PRIORITY	BYTE	Prioritätsklasse; einstellbar per Projektierung Bei S71500-CPUs: Priorität, Defaultwert: 6
OB83_OB_NUMBR	BYTE	OB-Nummer (83)
OB83_RESERVED_1	BYTE	Kennung für Baugruppe bzw. Submodul/Schnittstellen-modul
OB83_MDL_ID	BYTE	Bereich: • B#16#54: Peripheriebereich der Eingänge (PE) • B#16#55: Peripheriebereich der Ausgänge (PA)
OB83_MDL_ADDR	WORD	• Zentral oder dezentral PROFIBUS DP: Logische Basisadresse der betroffenen Baugruppe, bei einer Mischbaugruppe die kleinste verwendete logische Adresse der Baugruppe. Sind die logischen E- und A-Adressen der Mischbaugruppe gleich, erhält die logische Basisadresse die E-Kennung. • Dezentral PROFINET IO: Logische Basisadresse des Moduls/Submoduls

Variable	Datentyp	Beschreibung
OB83_RACK_NUM	INT	• Falls OB83_RESERVED_1 =B#16#A0: Nr. des Submoduls/Schnittstellenmoduls (low byte) • Falls OB83_RESERVED_1 = B#16#C4: – zentral: Nr. des Baugruppenträgers – dezentral PROFIBUS DP: Nr. der DP-Station (low byte) und DP-Mastersystem-ID (high byte) – dezentral PROFINET IO: physikalische Adresse: Kennungsbit (Bit 15, 1 = PROFINET IO), IO-System-ID (Bit 11 bis 14) und Stationsnummer (Bit 0 bis 10)

Variable	Datentyp	Beschreibung
OB83_MDL_TYPE	WORD	Bei S7-300 / S7-400: • Zentral oder dezentral PROFIBUS DP: Baugruppentyp der betroffenen Baugruppe (X: nicht anwenderrelevant). Hier nicht aufgeführte Baugruppentypen sind im Handbuch zur jeweiligen Baugruppe dokumentiert. – W#16#X5XX: Analogbaugruppe – W#16#X8XX: Funktionsbaugruppe – W#16#XCXX: CP – W#16#XFXX: Digitalbaugruppe – W#16#8340: Ersatztypkennung für Eingabebaugruppe – W#16#9340: Ersatztypkennung für Ausgabebaugruppe – W#16#A340: Ersatztypkennung für Mischbaugruppe (E/A) – W#16#F340: Ersatztypkennung für Leerbaugruppe oder nicht eindeutig identifizierbare Baugruppe (z. B. bei gepackten Adressen) – Hier nicht aufgeführte Baugruppentypen sind im Handbuch zur jeweiligen Baugruppe dokumentiert. • Dezentral PROFINET IO: – W#16#8101: Baugruppentyp der gesteckten Baugruppe ist gleich dem Baugruppentyp der gezogenen Baugruppe – W#16#8102: Baugruppentyp der gesteckten Baugruppe ist ungleich dem Baugruppentyp der gezogenen Baugruppe
		Bei S7-1500: • W#16#8101: bei allen Ziehen-Ereignissen; bei allen Stecken-Ereignissen mit Ausnahme des Steckens einer Baugruppe mit falschem Baugruppentyp • W#16#8102: Baugruppentyp der gesteckten Baugruppe ist ungleich dem Baugruppentyp der gezogenen Baugruppe
OB83_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die folgende Tabelle zeigt, welches Ereignis den Start des OB 83 verursacht hat.

OB83_EV_CLASS	OB83_FLT_ID	Bedeutung
B#16#39	B#16#51	• PROFINET IO-Modul gezogen • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Modul gezogen
B#16#39	B#16#54	• PROFINET IO-Submodul gezogen • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul gezogen
B#16#38	B#16#54	• PROFINET IO-Submodul gesteckt und entspricht parametriertem Submodul • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul gesteckt und entspricht parametriertem Submodul
B#16#38	B#16#55	• PROFINET IO-Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul
B#16#38	B#16#56	• PROFINET IO-Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung
B#16#38	B#16#57	• PROFINET IO-Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung
B#16#38	B#16#58	• PROFINET IO-Submodul Zugriffsfehler beseitigt • bei den CPUs 1510SP-1 PN und 1512SP-1 PN: IO-Submodul Zugriffsfehler beseitigt
B#16#39	B#16#61	Baugruppe gezogen bzw. nicht ansprechbar OB83_MDL_TYPE: Istbaugruppentyp
B#16#38	B#16#61	Baugruppe gesteckt, Baugruppentyp ok OB83_MDL_TYPE: Istbaugruppentyp
B#16#38	B#16#63	Baugruppe gesteckt, jedoch falscher Baugruppentyp OB83_MDL_TYPE: Istbaugruppentyp
B#16#38	B#16#64	Baugruppe gesteckt, jedoch gestört (Baugruppenkennung nicht lesbar) OB83_MDL_TYPE: Sollbaugruppentyp
B#16#38	B#16#65	Baugruppe gesteckt, jedoch Fehler bei Baugruppenparametrierung OB83_MDL_TYPE: Istbaugruppentyp
B#16#39	B#16#66	Baugruppe nicht ansprechbar, Lastspannungsfehler
B#16#38	B#16#66	Baugruppe wieder ansprechbar, Lastspannungsfehler beseitigt

OB83_EV_CLASS	OB83_FLT_ID	Bedeutung
B#16#33	B#16#67	Beginn Umparametrieren einer Baugruppe
B#16#32	B#16#67	Ende Umparametrieren einer Baugruppe
B#16#39	B#16#68	Umparametrieren einer Baugruppe mit Fehler beendet

Hinweis

Falls Sie eine DPV1- oder PROFINET-fähige CPU einsetzen, können Sie mit Hilfe der Anweisung "RALRM" weitere Informationen über den Alarm erhalten, die über die Startinformation des OB hinausgehen. Dies gilt auch dann, wenn der DP-Master im S7-kompatiblen Mode betrieben wird.

Programmieren des OB 83

Den OB 83 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 83 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 83 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 83 aus.
- Sie parametrieren die neu gesteckte Baugruppe mit Hilfe der Anweisungen "WR_PARM", "WR_DPARM", "PARM_MOD", "WR_REC" und "RD_REC" nach.

CPU-Hardwarefehler-OB (OB 84)

Beschreibung

Das Betriebssystem der CPU ruft den OB 84 in folgenden Fällen auf:

- nach der Erkennung und Beseitigung von Speicherfehlern
- bei WinAC RTX: Fehler im PC-Betriebssystem, z. B. blue screen

Wenn Sie den OB 84 nicht programmiert haben, geht die CPU nicht in den Betriebszustand STOP.

Sie können den CPU-Hardwarefehler-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten des CPU-Hardwarefehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des CPU-Hardwarefehler-OB. Als Variablennamen wurden die Defaultnamen des OB 84 gewählt.

Variable	Datentyp	Beschreibung
OB84_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#38: gehendes Ereignis • B#16#35, B#16#39: kommendes Ereignis
OB84_FLT_ID	BYTE	Fehlercode (B#16#82, B#16#83, B#16#85, B#16#86, B#16#87)
OB84_PRIORITY	BYTE	Prioritätsklasse; einstellbar per Projektierung (Hardware-Konfiguration)
OB84_OB_NUMBR	BYTE	OB-Nummer (84)
OB84_RESERVED_1	BYTE	Reserviert
OB84_RESERVED_2	BYTE	Reserviert
OB84_RESERVED_3	WORD	Reserviert
OB84_RESERVED_4	DWORD	Reserviert
OB84_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die folgende Tabelle zeigt, welches Ereignis den Start des OB 84 verursacht hat.

OB84_EV_CLASS	OB84_FLT_ID	Startereignis des OB 84
B#16#35	B#16#82	Speicherfehler vom Betriebssystem erkannt und beseitigt
B#16#35	B#16#83	Häufung von erkannten und korrigierten Speicherfehlern
B#16#35	B#16#85	Fehler im PC-Betriebssystem
B#16#39	B#16#86	Leistung einer H-Sync-Kopplung beeinträchtigt
B#16#35	B#16#87	Mehrbitspeicherfehler erkannt und korrigiert

Programmieren des OB 84

Den OB 84 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 84 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 84 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 84 aus.
- Sie senden mit Hilfe der Anweisung "WR_USMSG" eine Meldung zum Diagnosepuffer.

Programmablauffehler-OB (OB 85)

Beschreibung

Das Betriebssystem der CPU ruft den OB 85 auf, wenn eines der folgenden Ereignisse auftritt:

- Startereignis für einen nicht geladenen OB (außer OB 80, OB 81, OB 82, OB 83 und OB 86)
- Fehler beim Zugriff des Betriebssystems auf einen Baustein
- Peripheriezugriffsfehler bei der systemseitigen Aktualisierung des Prozessabbilds, falls der OB 85-Aufruf nicht per Projektierung unterdrückt wurde.

Hinweis

Wenn der OB 85 nicht programmiert ist, dann wechselt die CPU in den Betriebszustand STOP, wenn eines der genannten Ereignisse auftritt.

Sie können den Programmablauffehler-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten des Programmablauffehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Programmablauffehler-OB. Als Variablennamen wurden die Defaultnamen des OB 85 gewählt.

Variable	Datentyp	Beschreibung
OB85_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#35, B#16#38 (nur bei Fehlercode B#16#B3 und B#16#B4), B#16#39 (nur bei Fehlercode B#16#B1, B#16#B2, B#16#B3 und B#16#B4)
OB85_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#A1, B#16#A2, B#16#A3, B#16#A4, B#16#B1, B#16#B2, B#16#B3, B#16#B4)
OB85_PRIORITY	BYTE	Prioritätsklasse; einstellbar per Projektierung
OB85_OB_NUMBR	BYTE	OB-Nummer (85)
OB85_RESERVED_1	BYTE	Reserviert
OB85_RESERVED_2	BYTE	Reserviert
OB85_RESERVED_3	INT	Reserviert
OB85_ERR_EV_CLASS	BYTE	Klasse des Ereignisses, das den Fehler ausgelöst hat
OB85_ERR_EV_NUM	BYTE	Nummer des Ereignisses, das den Fehler ausgelöst hat
OB85_OB_PRIOR	BYTE	Prioritätsklasse des OB, der bearbeitet wurde, als der Fehler auftrat (nur bei manchen Fehlercodes, Genaueres siehe unten)
OB85_OB_NUM	BYTE	Nummer des OB, der bearbeitet wurde, als der Fehler auftrat (nur bei manchen Fehlercodes, Genaueres siehe unten)
OB85_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Falls Sie den OB 85 in Abhängigkeit von den möglichen Fehlercodes programmieren wollen, wird empfohlen, die lokalen Variablen wie folgt zu organisieren:

Variable	Datentyp
OB85_EV_CLASS	BYTE
OB85_FLT_ID	BYTE
OB85_PRIORITY	BYTE
OB85_OB_NUMBR	BYTE
OB85_DKZ23	BYTE
OB85_RESERVED_2	BYTE
OB85_Z1	WORD
OB85_Z23	DWORD
OB85_DATE_TIME	DATE_AND_TIME

Die folgende Tabelle zeigt, welches Ereignis den Start des OB 85 verursacht hat und wie die vom Fehlercode abhängigen Variablen belegt sind.

OB85_EV_CLASS	OB85_FLT_ID	Bedeutung
B#16#35	B#16#A1	Ihr Programm oder das Betriebssystem (aufgrund Ihrer Projektierung) erzeugt ein Startereignis für einen OB, der nicht in die CPU geladen ist. • OB85_Z1: Die zugehörige temporäre Variable des angeforderten OB. Dieser ist bestimmt durch OB85_Z23. • OB85_Z23: – high word: Klasse und Nummer des verursachenden Ereignisses – low word, high byte: Zum Fehlerzeitpunkt aktive Programmebene low word, low byte: Aktiver OB
B#16#35	B#16#A2	Ihr Programm oder das Betriebssystem (aufgrund Ihrer Projektierung) erzeugt ein Startereignis für einen OB, der nicht in die CPU geladen ist. OB85_Z1 und OB85_Z23 wie bei OB85_FLT_ID=B#16#A1
B#16#35	B#16#A3	Fehler beim Zugriff des Betriebssystems auf einen Baustein • OB85_Z1: Fehlerkennung des Betriebssystems – high byte: 1=Integrierte Funktion, 2=IEC-Timer – low byte: 0=keine Fehlerauflösung, 1=Baustein nicht geladen, 2=Bereichslängenfehler, 3=Schreibschutzfehler • OB85_Z23: – high word: Bausteinnummer – low word: Relativadresse des Fehler verursachenden MC7-Befehls. Der Bausteintyp ist OB 85_DKZ23 zu entnehmen (B#16#88: OB, B#16#8C: FC, B#16#8E: FB, B#16#8A: DB).

OB85_EV_CLASS	OB85_FLT_ID	Bedeutung
B#16#35	B#16#A4	PROFINET Interface-DB nicht ansprechbar
B#16#34	B#16#A4	PROFINET Interface-DB wieder ansprechbar
B#16#39	B#16#B1	Peripheriezugriffsfehler beim Aktualisieren des Prozessabbilds der Eingänge • OB85_DKZ23: Kennung für die Art des Prozessabbildtransfers, bei dem der Peripheriezugriffsfehler auftrat – B#16#10: Bytezugriff – B#16#20: Wortzugriff – B#16#30: Doppelwortzugriff – B#16#56 oder B#16#57: Übertragung eines projektierten Konsistenzbereichs • OB85_Z1: reserviert für interne Verwendung der CPU: logische Basisadresse der Baugruppe Falls OB85_RESERVED_2 den Wert B#16#76 hat, enthält OB85_Z1 den Rückgabewert der betroffenen Anweisung ("DPRD_DAT", "DPWR_DAT", "UPDAT_PI" oder "UPDAT_PO"). • OB85_Z23: – Byte 0: Falls OB85_DKZ23=B#16#57: Teilprozessabbild-Nr. – Byte 1: irrelevant, falls OB85_DKZ23=B#16#10, 20 oder 30; Länge des Konsistenzbereichs in Bytes, falls OB85_DKZ23=B#16#57 – Byte 0 und Byte 1, falls OB85_DKZ23=B#16#56: Länge des Konsistenzbereichs in Bytes – Bytes 2 und 3: die PZF verursachende Peripherieadresse, falls OB85_DKZ23=B#16#10, 20 oder 30; logische Anfangsadresse des Konsistenzbereichs, falls OB85_DKZ23=B#16#57
B#16#39	B#16#B2	Peripheriezugriffsfehler bei Übertragung des Prozessabbilds der Ausgänge zu den Ausgabebaugruppen OB85_DKZ23, OB85_Z1 und OB85_Z23 wie bei OB85_FLT_ID=B#16#B1
Sie erhalten die Fehlercodes B#16#B1 und B#16#B2, falls Sie für die systemseitige Prozessabbildaktualisierung den wiederholten OB 85-Aufruf bei Peripheriezugriffsfehlern projektiert haben.		

OB85_EV_CLASS	OB85_FLT_ID	Bedeutung
B#16#39/B#16#38	B#16#B3	Peripheriezugriffsfehler beim Aktualisieren des Prozessabbilds der Eingänge kommend/gehend • OB85_DKZ23: Kennung für die Art des Prozessabbildtransfers, bei dem der Peripheriezugriffsfehler auftrat – B#16#10: Bytezugriff – B#16#20: Wortzugriff – B#16#30: Doppelwortzugriff – B#16#56 oder B#16#57: Übertragung eines projektierten Konsistenzbereichs • OB85_Z1: reserviert für interne Verwendung der CPU: logische Basisadresse der Baugruppe Falls OB85_RESERVED_2 den Wert B#16#76 hat, enthält OB85_Z1 den Rückgabewert der betroffenen Anweisung ("DPRD_DAT", "DPWR_DAT ", "UPDAT_PI", "UPDAT_PO"). • OB85_Z23: – Byte 0: Falls OB85_DKZ23=B#16#57: Teilprozessabbild-Nr. – Byte 1: irrelevant, falls OB85_DKZ23=B#16#10, 20 oder 30; Länge des Konsistenzbereichs in Bytes, falls OB85_DKZ23=B#16#57 – Byte 0 und Byte 1, falls OB85_DKZ23=B#16#56: Länge des Konsistenzbereichs in Bytes – Bytes 2 und 3: die PZF verursachende Peripherieadresse, falls OB85_DKZ23=B#16#10, 20 oder 30; logische Anfangsadresse des Konsistenzbereichs, falls OB85_DKZ23=B#16#57
B#16#39/B#16#38	B#16#B4	Peripheriezugriffsfehler beim Aktualisieren des Prozessabbilds der Ausgänge kommend/gehend OB85_DKZ23, OB85_Z1, OB85_Z23 wie bei OB85_FLT_ID=B#16#B3
Sie erhalten die Fehlercodes B#16#B3 und B#16#B4, falls Sie für die systemseitige Prozessabbildaktualisierung den OB 85-Aufruf nur bei kommenden und gehenden Peripheriezugriffsfehlern projektiert haben. Nach Kaltstart oder Neustart (Warmstart) werden bei der nächsten Prozessabbildaktualisierung alle Zugriffe auf nicht vorhandene Ein- und Ausgänge als kommende Peripheriezugriffsfehler gemeldet.		

Programmieren des OB 85

Den OB 85 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 85 bearbeitet werden soll, in den erzeugten Baustein und laden Sie diesen als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 85 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 85 aus und stellen fest, welche Baugruppe defekt ist oder fehlt (Angabe der Baugruppen-Anfangsadresse).
- Sie ermitteln mit Hilfe der Anweisung "LGC_GADR" den Steckplatz der betreffenden Baugruppe.

Baugruppenträgerausfall-OB (OB 86)

Beschreibung

Das Betriebssystem der CPU ruft den OB 86 in folgenden Fällen auf:

- Der Ausfall eines zentralen Erweiterungsgeräts (nicht bei S7-300) wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines DP-Mastersystems wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall einer Station bei Dezentraler Peripherie (PROFIBUS DP oder PROFINET IO) wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Sie haben eine Station bei Dezentraler Peripherie (PROFIBUS DP oder PROFINET IO) mit der Anweisung "D_ACT_DP" mittels MODE=4 deaktiviert.
- Sie haben eine Station bei Dezentraler Peripherie (PROFIBUS DP oder PROFINET IO) mit der Anweisung "D_ACT_DP" mittels MODE=3 aktiviert.
- Der Ausfall eines PROFINET IO-Systems oder einer PROFINET IO-Station oder eines Teils der Submodule eines PROFINET I-Device wird erkannt.

Haben Sie den OB 86 nicht programmiert und ein solcher Fehler tritt auf, geht die CPU in den Betriebszustand STOP.

Sie können den Baugruppenträgerausfall-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten des Baugruppenträgerausfall-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Baugruppenträgerausfall-OB. Als Variablennamen wurden die Defaultnamen des OB 86 gewählt.

Variable	Datentyp	Beschreibung
OB86_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#32: Aktivierung einer Station mit der Anweisung "D_ACT_DP" mittels MODE=3 • B#16#33: Deaktivierung einer Station mit der Anweisung "D_ACT_DP" mittels MODE=4 • B#16#38: gehendes Ereignis • B#16#39: kommendes Ereignis
OB86_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#C1, B#16#C2, B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)
OB86_PRIORITY	BYTE	Prioritätsklasse; einstellbar per Projektierung Bei S7-1500-CPUs: Priorität, Defaultwert: 6
OB86_OB_NUMBR	BYTE	OB-Nummer (86)
OB86_RESERVED_1	BYTE	Reserviert
OB86_RESERVED_2	BYTE	Reserviert
OB86_MDL_ADDR	WORD	Abhängig vom Fehlercode
OB86_RACKS_FLTD	ARRAY [0 ..31] OF BOOL	Abhängig vom Fehlercode
OB86_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Falls Sie den OB 86 in Abhängigkeit von den möglichen Fehlercodes programmieren wollen, wird empfohlen, die lokalen Variablen wie folgt zu organisieren:

Variable	Datentyp
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

Die folgende Tabelle zeigt, welches Ereignis den Start des OB 86 verursacht hat.

OB86_EV_CLASS	OB86_FLT_ID	Bedeutung
B#16#39	B#16#C1	Erweiterungsgeräteausfall • OB86_MDL_ADDR: logische Basisadresse der IM • OB86_Z23: enthält für jedes mögliche Erweiterungsgerät ein Bit: Diejenigen Erweiterungsgeräte werden als ausgefallen gemeldet (die zugehörigen Bits sind gesetzt), die den Aufruf des OB 86 verursacht haben. Bereits früher ausgefallene Erweiterungsgeräte werden nicht mehr angezeigt. – Bit 0: stets 0 – Bit 1: 1. Erweiterungsgerät – : – Bit 21: 21. Erweiterungsgerät – Bit 22 bis 29: stets 0 – Bit 30: Ausfall mindestens eines Erweiterungsgeräts im SIMATIC S5-Bereich – Bit 31: stets 0
B#16#38	B#16#C1	Erweiterungsgerätewiederkehr OB86_MDL_ADDR wie bei OB86_FLT_ID=B#16#C1. In OB86_Z23 werden die wiedergekommenen Erweiterungsgeräte gemeldet (die zugehörigen Bits sind gesetzt).
B#16#38	B#16#C2	Erweiterungsgerätewiederkehr (Erweiterungsgeräteausfall gehend mit Abweichung Soll-/ Istausbau) • OB86_MDL_ADDR: logische Basisadresse der IM • OB86_Z23: enthält für jedes mögliche Erweiterungsgerät ein Bit, siehe OB86_FLT_ID B#16#C1. Bedeutung eines gesetzten Bits: Im betroffenen Erweiterungsgerät: – sind Baugruppen mit falscher Baugruppenkennung vorhanden. – fehlen projektierte Baugruppen. – ist mindestens eine Baugruppe defekt.
B#16#39	B#16#C3	Dezentrale Peripherie: Ausfall eines DP-Mastersystems Nur ein kommendes Ereignis führt zum OB 86-Start mit dem Fehlercode B#16#C3. Ein gehendes Ereignis führt zum OB 86-Start mit Fehlercode B#16#C4 und Ereignisklasse B#16#38: Die Wiederkehr jeder unterlagerten DP-Station hat einen OB 86-Start zur Folge.) • OB86_MDL_ADDR: Logische Basisadresse des DP-Masters • OB86_Z23: DP-Mastersystem-ID: – Bit 0 bis 7: reserviert – Bit 8 bis 15: DP-Mastersystem-ID – Bit 16 bis 31: reserviert

OB86_EV_CLASS	OB86_FLT_ID	Bedeutung
B#16#39/B#16#38	B#16#C4	Ausfall einer DP-Station • OB86_MDL_ADDR: Logische Basisadresse des DP-Masters • OB86_Z23: Adresse des betroffenen DP-Slaves: – Bit 0 bis 7: Nr. der DP-Station – Bit 8 bis 15: DP-Mastersystem-ID – Bit 16 bis 30: logische Basisadresse bei einem S7-Slave bzw. Diagnoseadresse bei einem DP-Normslave – Bit 31: I/O-Kennung
B#16#39/B#16#38	B#16#C5	Wiederkehr einer DP-Station, Station jedoch gestört OB86_MDL_ADDR und OB86_Z23 wie bei FLT_ID=B#16#C4
B#16#38	B#16#C6	Erweiterungsgerätewiederkehr, jedoch Fehler bei Baugruppenparametrierung • OB86_MDL_ADDR: logische Basisadresse der IM • OB86_Z23: enthält für jedes mögliche Erweiterungsgerät ein Bit: – Bit 0: stets 0 – Bit 1: 1. Erweiterungsgerät: – … – Bit 21: 21. Erweiterungsgerät – Bit 22 bis 30: reserviert – Bit 31: stets 0 • Bedeutung eines gesetzten Bits: Im betroffenen Erweiterungsgerät – sind Baugruppen mit falscher Baugruppenkennung vorhanden. – sind Baugruppen mit fehlenden oder falschen Parametern vorhanden.
B#16#38	B#16#C7	Wiederkehr einer DP-Station, jedoch Fehler bei Baugruppenparametrierung • OB86_MDL_ADDR: logische Basisadresse des DP-Masters • Adresse des betroffenen DP-Slaves: – Bit 0 bis 7: Nr. der DP-Station – Bit 8 bis 15: DP-Mastersystem-ID – Bit 16 bis 30: logische Basisadresse des DP-Slave – Bit 31: I/O-Kennung

OB86_EV_CLASS	OB86_FLT_ID	Bedeutung
B#16#38	B#16#C8	Wiederkehr einer DP-Station, jedoch Abweichung Soll-/Ist-ausbau • OB86_MDL_ADDR: logische Basisadresse des DP-Masters • OB86_Z23: Adresse des betroffenen DP-Slaves: – Bit 0 bis 7: Nr. der DP-Station – Bit 8 bis 15: DP-Mastersystem-ID – Bit 16 bis 30: logische Basisadresse des DP-Slave – Bit 31: I/O-Kennung
B#16#32/B#16#33	B#16#C9	Aktivierung/Deaktivierung eines DP-Slaves mit der Anweisung "D_ACT_DP" mittels MODE=3/MODE=4 • OB86_MDL_ADDR: Logische Basisadresse des DP-Masters • OB86_Z23: Adresse des betroffenen DP-Slaves – Bit 0 bis 7: Nr. der DP-Station – Bit 8 bis 15: DP-Mastersystem-ID – Bit 16 bis 30: Logische Basisadresse bei einem S7-Slave bzw. Diagnoseadresse bei einem DP-Normslave – Bit 31: I/O-Kennung
B#16#39	B#16#CA	PROFINET IO-Systemausfall • OB86_MDL_ADDR: logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: 0 (Stationsnummer) – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 31: 0
B#16#39/38	B#16#CB	PROFINET IO-Stationsausfall/Stationswiederkehr • OB86_RESERVED_1: B#16#C4 • OB86_MDL_ADDR: logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: logische Basisadresse der Station – Bit 31: I/O-Kennung Hinweis: Bevor Sie in Ihrem Programm überprüfen, ob ein PROFINET IO-Stationsausfall die Ursache für den OB86-Start ist, sollten Sie überprüfen, ob ein PROFINET IO-Systemausfall vorliegt.

OB86_EV_CLASS	OB86_FLT_ID	Bedeutung
B#16#/38	B#16#CC	PROFINET IO-Stationswiederkehr mit Störung oder Wartung • OB86_RESERVED_1: B#16#C4 • OB86_MDL_ADDR: logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: logische Basisadresse der Station – Bit 31: I/O-Kennung
B#16#38	B#16#CD	PROFINET IO-Stationswiederkehr, Sollausbau weicht von Istausbau ab • OB86_MDL_ADDR: logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: logische Basisadresse der Station – Bit 31: I/O-Kennung
B#16#38	B#16#CE	PROFINET IO-Stationswiederkehr, Fehler bei der Baugruppenparametrierung • OB86_MDL_ADDR: logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: logische Basisadresse der Station – Bit 31: I/O-Kennung
B#16#32/B#16#33	B#16#CF	Aktivierung/Deaktivierung eines PROFINET IO-Devices mit der Anweisung "D_ACT_DP" mittels MODE=3/MODE=4 • OB86_MDL_ADDR: Logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: Logische Basisadresse der Station – Bit 31: I/O-Kennung

OB86_EV_CLASS	OB86_FLT_ID	Bedeutung
B#16#39/B#16#38	B#16#F8	Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Device • OB86_MDL_ADDR: Logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: Logische Basisadresse der Station – Bit 31: I/O-Kennung
B#16#38	B#16#F9	Wiederkehr eines Teils der Submodule eines PROFINET I-Device mit Device-Konfigurationsunterschied • OB86_MDL_ADDR: Logische Basisadresse des IO-Controllers • OB86_Z23: – Bit 0 bis 10: Stationsnummer – Bit 11 bis 14: IO-System-ID – Bit 15: 1 – Bit 16 bis 30: Logische Basisadresse der Station – Bit 31: I/O-Kennung

Hinweis

Falls Sie eine DPV1- oder PROFINET-fähige CPU einsetzen, können Sie mit Hilfe der Anweisung "RALRM" weitere Informationen über den Alarm erhalten, die über die Startinformation des OB hinausgehen. Dies gilt auch dann, wenn der DP-Master im S7-kompatiblen Mode betrieben wird.

Programmieren des OB 86

Den OB 86 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 86 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 86 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 86 aus und stellen fest, welcher Baugruppenträger defekt ist oder fehlt.
- Sie tragen mit Hilfe der Anweisung "WR_USMSG" eine Meldung in den Diagnosepuffer ein und senden die Meldung zu einem Beobachtungsgerät.

Kommunikationsfehler-OB (OB 87)

Beschreibung

Das Betriebssystem der CPU ruft den OB 87 auf, wenn ein Ereignis auftritt, das durch einen Kommunikationsfehler ausgelöst wurde.

Wenn Sie den OB 87 nicht programmiert haben und ein Startereignis für den OB 87 auftritt, verhält sich Ihre CPU wie folgt:

- Eine S7-300-CPU geht in den Betriebszustand STOP.
- Eine S7-400-CPU geht nicht in den Betriebszustand STOP.

Sie können den Kommunikationsfehler-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT und "EN_AIRT" verzögern.

Lokaldaten des Kommunikationsfehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Kommunikationsfehler-OB. Als Variablennamen wurden die Defaultnamen des OB 87 gewählt.

Variable	Datentyp	Beschreibung
OB87_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#39
OB87_FLT_ID	BYTE	Fehlercode (mögliche Werte: B#16#D2, B#16#D3, B#16#D4, B#16#D5, B#16#E1, B#16#E2, B#16#E3, B#16#E4, B#16#E5, B#16#E6)
OB87_PRIORITY	BYTE	Prioritätsklasse; einstellbar per Projektierung
OB87_OB_NUMBR	BYTE	OB-Nummer (87)
OB87_RESERVED_1	BYTE	Reserviert
OB87_RESERVED_2	BYTE	Reserviert
OB87_RESERVED_3	WORD	Abhängig vom Fehlercode
OB87_RESERVED_4	DWORD	Abhängig vom Fehlercode
OB87_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die vom Fehlercode abhängigen Variablen haben folgende Bedeutung:

Fehlercode	Variable	Bedeutung
B#16#D2		Senden der Diagnoseeinträge ist derzeit nicht möglich.
B#16#D3		Synchronisationstelegramme können nicht gesendet werden (Master).
B#16#D4		Unzulässiger Uhrzeitsprung durch Uhrzeitsynchronisation
B#16#D5		Fehler bei Übernahme der Synchronisationszeit (Slave)
	OB87_RESERVED_3	Enthält keine weitere Information
	OB87_RESERVED_4	Enthält keine weitere Information
B#16#E1		Falsche Telegrammkennung bei Globaldatenkommunikation
B#16#E2		GD-Paketstatus ist nicht in einen DB eintragbar

Fehlercode	Variable	Bedeutung
	OB87_RESERVED_3	DB-Nummer
	OB87_RESERVED_4	• High word: Enthält keine weitere Information
		• Low word: GD-Kreisnummer (high byte), GD-Paketnummer (low byte)
B#16#E3		Telegrammlängenfehler bei Globaldatenkommunikation
B#16#E4		Unzulässige GD-Paketnummer empfangen
	OB87_RESERVED_3	Schnittstellenkennung (0: K-Bus, 1: MPI)
	OB87_RESERVED_4	• High byte: GD-Kreisnummer • Low byte: Enthält keine weitere Information
B#16#E5		Fehler bei Zugriff auf DB beim Datenaustausch über Kommunikationsfunktionsbausteine
	OB87_RESERVED_1	Bausteintyp: B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB
	OB87_RESERVED_3	Reserviert für interne Verwendung der CPU
	OB87_RESERVED_4	• High word: Nummer des Bausteins mit dem Fehler verursachenden MC7-Befehl • Low word: Relativadresse des Fehler verursachenden MC7-Befehls
B#16#E6		GD-Gesamtstatus ist nicht in DB eintragbar
	OB87_RESERVED_3	DB-Nummer
	OB87_RESERVED_4	Enthält keine weitere Information

Programmieren des OB 87

Den OB 87 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 87 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 87 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 87 aus.
- Sie legen einen Datenbaustein an, wenn der Datenbaustein für die Statusinformation der Globaldatenkommunikation fehlt.

Bearbeitungsabbruch-OB (OB 88)

Beschreibung

Das Betriebssystem der CPU ruft den OB 88 auf, wenn die Bearbeitung eines Programmbausteins abgebrochen wird. Mögliche Abbruchursachen sind:

- zu große Schachtelungstiefe bei Synchronfehlern
- zu große Schachtelungstiefe von Bausteinaufrufen (U-Stack)
- Fehler beim Allokieren von Lokaldaten

Wenn Sie den OB 88 nicht programmiert haben und ein Bearbeitungsabbruch tritt auf, geht die CPU in den Betriebszustand STOP (Ereignis W#16#4570).

Wenn der Bearbeitungsabbruch in der Prioritätsklasse 28 auftritt, geht die CPU in STOP.

Sie können den Bearbeitungsabbruch-OB mit Hilfe der Anweisung "DIS_IRT" sperren, mit der Anweisung "EN_IRT" wieder freigeben und mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" verzögern.

Lokaldaten des Bearbeitungsabbruch-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Bearbeitungsabbruch-OB. Als Variablennamen wurden die Defaultnamen des OB 88 gewählt.

Variable	Datentyp	Beschreibung
OB88_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#35
OB88_SW_FLT	BYTE	Fehlercode Mögliche Werte: • B#16#71: Zu große Schachtelungstiefe von Klammerebenen • B#16#72: Zu große Schachtelungstiefe von Master Control Relais • B#16#73: Zu große Schachtelungstiefe bei Synchronfehlern • B#16#74: Zu große Schachtelungstiefe von Bausteinaufrufen (U-Stack) • B#16#75: Zu große Schachtelung von Bausteinaufrufen (B-Stack) • B#16#76: Fehler beim Allokieren von Lokaldaten • B#16#78: Unbekannte Anweisung • B#16#7A: Sprunganweisung mit Ziel außerhalb des Bausteins Bitte lesen Sie in der Operationsliste Ihrer CPU nach, welche Fehlercodes für Ihre CPU gültig sind.
OB88_PRIORITY	BYTE	Prioritätsklasse: 28
OB88_OB_NUMBR	BYTE	OB-Nummer (88)
OB88_BLK_TYPE	BYTE	Art des Bausteins, in dem der Fehler aufgetreten ist: • B#16#88: OB • B#16#8C: FC • B#16#8E: FB • B#16##00: Unterbrechungsstelle nicht ermittelbar
OB88_RESERVED_1	BYTE	Reserviert
OB88_FLT_PRIORITY	BYTE	Prioritätsklasse des Fehler verursachenden OB
OB88_FLT_OB_NUMBR	BYTE	Nummer des Fehler verursachenden OB

Variable	Datentyp	Beschreibung
OB88_BLK_NUM	WORD	Nummer des Bausteins mit dem Fehler verursachenden MC7-Befehl
OB88_PRG_ADDR	WORD	Relativadresse des Fehler verursachenden MC7-Befehls
OB88_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Programmieren des OB 88

Den OB 88 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 88 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 88 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 88 aus.
- Sie tragen die Fehlerursache in einen Melde-Datenbaustein ein.

Hintergrund-OB (OB 90)

Beschreibung

S7 bietet die Überwachung einer maximalen Zykluszeit und garantiert die Einhaltung einer minimalen Zykluszeit. Falls die Bearbeitungszeit des OB 1 einschließlich aller eingeschachtelten Alarmbearbeitungen und Systemtätigkeiten kleiner ist als die von Ihnen vorgegebene Mindestzykluszeit,

- ruft das Betriebssystem den Hintergrund-OB auf (falls er auf der CPU vorhanden ist).
- verzögert das Betriebssystem den nächsten Start des OB 1 (falls der OB 90 nicht auf der CPU vorhanden ist).

Funktionsweise des Hintergrund-OB

Der OB 90 hat die niedrigste Priorität aller OBs. Er wird durch jede Systemtätigkeit und jede Alarmbearbeitung (auch vom OB 1 nach Ablauf der Mindestzykluszeit) unterbrochen. Eine Ausnahme bildet die Bearbeitung von Anweisungen, die im OB 90 gestartet werden. Diese werden mit der Priorität des OB 1 bearbeitet und deshalb nicht durch die OB 1-Bearbeitung unterbrochen. Eine Zeitüberwachung des OB 90 findet nicht statt.

Das Anwenderprogramm im OB 90 wird in folgenden Fällen beginnend vom ersten Befehl bearbeitet:

- nach Neustart (Warmstart) oder Kaltstart oder Wiederanlauf
- nach dem Laden oder Löschen eines Bausteins

- nach dem Laden des OB 90 in die CPU im Betriebszustand RUN
- nach Beendigung des Hintergrundzyklus

Hinweis

Bei Projektierungen, bei denen die Mindestzykluszeit und die Zyklusüberwachungszeit nahe beieinander liegen, kann es beim Aufruf von Anweisungen im Hintergrund-OB zu unerwarteten Zykluszeitüberschreitungen kommen.

Lokaldaten des Hintergrund-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des OB 90. Als Variablennamen wurden die Defaultnamen des OB 90 gewählt.

Variable	Datentyp	Beschreibung
OB90_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#11: aktiv
OB90_STRT_INF	BYTE	• B#16#91: Neustart (Warmstart)/Kaltstart/ Wiederanlauf • B#16#92: Löschen eines Bausteins • B#16#93: Laden des OB 90 in die CPU im Betriebszustand RUN • B#16#95: Abschluss des Hintergrundzyklus
OB90_PRIORITY	BYTE	Prioritätsklasse: 29 (entspricht der Priorität 0.29)
OB90_OB_NUMBR	BYTE	OB-Nummer (90)
OB90_RESERVED_1	BYTE	Reserviert
OB90_RESERVED_2	BYTE	Reserviert
OB90_RESERVED_3	INT	Reserviert
OB90_RESERVED_4	INT	Reserviert
OB90_RESERVED_5	INT	Reserviert
OB90_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Anlauf-OBs (OB 100, OB 101 und OB 102)

Anlaufarten

Man unterscheidet die Anlaufarten

- Wiederanlauf (S7-400)
- Neustart (Warmstart)
- Kaltstart

Der folgenden Tabelle können Sie entnehmen, welchen OB das Betriebssystem im Anlauf jeweils aufruft.

Anlaufart	Zugehöriger OB
Wiederanlauf	OB 101
Neustart (Warmstart)	OB 100
Kaltstart	OB 102

Startereignisse für den Anlauf

Die CPU führt einen Anlauf durch

- nach NETZ EIN
- nach Anforderung durch eine Kommunikationsfunktion (über Menübefehl vom PG aus oder durch Aufruf der Anweisungen "START" oder "RESUME" auf einer anderen CPU)

Abhängig vom Startereignis, von der vorliegenden CPU und deren eingestellten Parametern wird der zugehörige Anlauf-OB (OB 100, OB 101 bzw. OB 102) aufgerufen. Darin können Sie durch entsprechende Programmierung bestimmte Voreinstellungen für Ihr zyklisches Programm vornehmen.

Lokaldaten der Anlauf-OBs

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen eines Anlauf-OB. Als Variablennamen wurden die Defaultnamen gewählt.

Variable	Datentyp	Beschreibung
OB10x_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#13: aktiv
OB10x_STRTUP	BYTE	Anlaufanforderung: • B#16#81: Manuelle Neustart- (Warmstart-) -Anforderung • B#16#82: Automatische Neustart- (Warmstart-) -Anforderung • B#16#83: Manuelle Wiederanlaufanforderung • B#16#84: Automatische Wiederanlaufanforderung • B#16#85: Manuelle Kaltstartanforderung • B#16#86: Automatische Kaltstartanforderung • B#16#87: Master: Manuelle Kaltstartanforderung • B#16#88: Master: Automatische Kaltstartanforderung • B#16#8A: Master: Manuelle Neustart (Warmstart)-Anforderung • B#16#8B: Master: Automatische Neustart (Warmstart)-Anforderung
OB10x_PRIORITY	BYTE	Prioritätsklasse: 27 bzw. 1 (bei S7-1500-CPUs)

Variable	Datentyp	Beschreibung
OB10x_OB_NUMBR	BYTE	OB-Nummer (100, 101 oder 102)
OB10x_RESERVED_1	BYTE	Reserviert
OB10x_RESERVED_2	BYTE	Reserviert
OB10x_STOP	WORD	Nummer des Ereignisses, das die CPU in STOP versetzt hat
OB10x_STRT_INFO	DWORD	Ergänzende Informationen zum aktuellen Anlauf
OB10x_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die folgende Tabelle zeigt die Belegung der Variablen OB10x_STRT_INFO.

Bit-Nr.	Bedeutung	mögliche Werte binär	Erläuterung
31-24	Anlaufinformation	0000 xxxx	Baugruppenträger-Nr. 0 (nur bei H-CPUs)
		0100 xxxx	Baugruppenträger-Nr. 1 (nur bei H-CPUs)
		1000 xxxx	Baugruppenträger-Nr. 2 (nur bei H-CPUs)
		0001 xxxx	Multicomputing (nur S7-400)
		0010 xxxx	Betrieb mehrerer Zentralbaugruppen im segmentierten Baugruppenträger (nur S7-400)
		xxxx xxx0	Keine Soll-Istausbau-Differenz vorhanden (nur S7-300)
		xxxx xxx1	Soll-Istausbau-Differenz vorhanden (nur S7-300)
		xxxx xx0x	Keine Soll-Istausbau-Differenz vorhanden
		xxxx xx1x	Soll-Istausbau-Differenz vorhanden
		xxxx x0xx	Keine H-CPU
		xxxx x1xx	H-CPU
		xxxx 0xxx	Uhr für Zeitstempel bei letztem NETZ EIN nicht gepuffert
		xxxx 1xxx	Uhr für Zeitstempel bei letztem NETZ EIN gepuffert

Bit-Nr.	Bedeutung	mögliche Werte binär	Erläuterung
23-16	Soeben durchgeführte Anlaufart Hinweis: Diese Bits haben bei S7-1500-CPUs keine Bedeutung.	0000 0001	Neustart (Warmstart) bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 0011	Neustart (Warmstart) über Betriebsartenschalter
		0000 0100	Neustart (Warmstart) über PG-Bedienung
		0000 0101	Kaltstart bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 0111	Kaltstart über Betriebsartenschalter
		0000 1000	Kaltstart über PG-Bedienung
		0000 1010	Wiederanlauf bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 1011	Wiederanlauf über Betriebsartenschalter (nur S7-400)
		0000 1100	Wiederanlauf über PG-Bedienung (nur S7-400)
		0001 0000	Automatischer Neustart (Warmstart) nach gepuffertem NETZ EIN
		0001 0001	Kaltstart nach gepuffertem NETZ EIN, entsprechend der Parametrierung
		0001 0011	Neustart (Warmstart) über Betriebsartenschalter; letzter NETZ EIN gepuffert
		0001 0100	Neustart (Warmstart) über PG-Bedienung; letzter NETZ EIN gepuffert
		0010 0000	Automatischer Neustart (Warmstart) nach ungepuffertem NETZ EIN (mit systemseitigem Urlöschen)
		0010 0001	Kaltstart nach ungepuffertem NETZ EIN (mit systemseitigem Urlöschen)
		0010 0011	Neustart (Warmstart) über Betriebsartenschalter; letzter NETZ EIN ungepuffert
		0010 0100	Neustart (Warmstart) über PG-Bedienung; letzter NETZEIN ungepuffert
		1010 0000	Automatischer Wiederanlauf nach gepuffertem NETZ EIN entsprechend der Parametrierung (nur S7-400)
15-12	Zulässigkeit automatischer Anlaufarten Hinweis: Diese Bits haben bei S7-1500-CPUs keine Bedeutung.	0000	Automatischer Anlauf unzulässig, Urlöschen angefordert
		0001	Automatischer Anlauf unzulässig, Parameteränderung etc. erforderlich
		0111	Automatischer Neustart (Warmstart) zulässig
		1111	Automatischer Neustart (Warmstart)/Wiederanlauf zulässig (nur S7-400)

Bit-Nr.	Bedeutung	mögliche Werte binär	Erläuterung
11-8	Zulässigkeit manueller Anlaufarten Hinweis: Diese Bits haben bei S7-1500-CPUs keine Bedeutung.	0000	Anlauf unzulässig, Urlöschen angefordert
		0001	Anlauf unzulässig, Parameteränderung etc. erforderlich
		0111	Neustart (Warmstart) zulässig
		1111	Neustart (Warmstart) und Wiederanlauf zulässig (nur S7-400)
7-0	Letzte gültige Bedienung oder Einstellung der automatischen Anlaufart bei NETZ EIN	0000 0000	Keine Anlaufart
		0000 0001	Neustart (Warmstart) bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 0011	Neustart (Warmstart) über Betriebsartenschalter
		0000 0100	Neustart (Warmstart) über PG-Bedienung
		0000 0101	Kaltstart bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 0111	Kaltstart über Betriebsartenschalter
		0000 1000	Kaltstart über PG-Bedienung
		0000 1010	Wiederanlauf bei Multicomputing ohne Bedienung auf der CPU entsprechend der Parametrierung (nur S7-400)
		0000 1011	Wiederanlauf über Betriebsartenschalter (nur S7-400)
		0000 1100	Wiederanlauf über PG-Bedienung (nur S7-400)
		0001 0000	Automatischer Neustart (Warmstart) nach gepuffertem NETZ EIN
		0001 0001	Kaltstart nach gepuffertem NETZ EIN, entsprechend der Parametrierung
		0001 0011	Neustart (Warmstart) über Betriebsartenschalter; letzter NETZ EIN gepuffert
		0001 0100	Neustart (Warmstart) über PG-Bedienung; letzter NETZEIN gepuffert
		0010 0000	Automatischer Neustart (Warmstart) nach ungepuffertem NETZ EIN (mit systemseitigem Urlöschen)
		0010 0001	Kaltstart nach ungepuffertem NETZ EIN (mit systemseitigem Urlöschen)
		0010 0011	Neustart (Warmstart) über Betriebsartenschalter; letzter NETZ EIN ungepuffert
		0010 0100	Neustart (Warmstart) über PG-Bedienung; letzter NETZ EIN ungepuffert
		1010 0000	Automatischer Wiederanlauf nach gepuffertem NETZ EIN entsprechend der Parametrierung (nur S7-400)

Siehe auch

Grundlagen zum Betriebszustand "ANLAUF" (Seite 756)

Programmierfehler-OB (OB 121)

Beschreibung

Das Betriebssystem der CPU ruft den OB 121 auf, wenn ein Ereignis auftritt, das durch einen Fehler während der Bearbeitung Ihres Programms ausgelöst wurde. Wenn Sie beispielsweise in Ihrem Programm einen Baustein aufrufen, der nicht in die CPU geladen wurde, dann wird der OB 121 aufgerufen.

Funktionsweise des Programmierfehler-OB

Der OB 121 läuft in derselben Prioritätsklasse wie der unterbrochene Baustein.

Ist der OB 121 nicht programmiert, dann wechselt die CPU den Betriebszustand von RUN nach STOP.

S7 verfügt über die folgenden Anweisungen, mit denen Sie Startereignisse des OB 121 maskieren und demaskieren können, während Ihr Programm bearbeitet wird:

- Die Anweisung "MSK_FLT" maskiert bestimmte Fehlercodes.
- Die Anweisung "DMSK_FLT" demaskiert die Fehlercodes, die von der Anweisung "MSK_FLT" maskiert wurden.
- Die Anweisung "READ_ERR" liest das Ereignisstatusregister.

Lokaldaten des Programmierfehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Programmierfehler-OB. Als Variablennamen wurden die Defaultnamen des OB 121 gewählt.

Variable	Datentyp	Beschreibung
OB121_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#25
OB121_SW_FLT	BYTE	Fehlercode (mögliche Werte: B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#3A, B#16#3C, B#16#3D, B#16#3E, B#16#3F)
OB121_PRIORITY	BYTE	Prioritätsklasse des OB, in dem der Fehler aufgetreten ist Bei S7-1500-CPUs: Priorität, Defaultwert: 7
OB121_OB_NUMBR	BYTE	OB-Nummer (121)
OB121_BLK_TYPE	BYTE	Art des Bausteins, in dem der Fehler aufgetreten ist (bei S7-300 wird hier kein gültiger Wert eingetragen): B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB
OB121_RESERVED_1	BYTE	Reserviert

Variable	Datentyp	Beschreibung
OB121_FLT_REG	WORD	Fehlerquelle (abhängig vom Fehlercode), z. B.: • Register, in dem der Konvertierungsfehler aufgetreten ist • Fehlerhafte Adresse (Lese-/Schreibfehler) • Fehlerhafte Nummer eines Timers, eines Zählers oder eines Bausteins • Fehlerhafter Speicherbereich
OB121_BLK_NUM	WORD	Nummer des Bausteins mit dem Fehler verursachenden MC7-Befehl (bei S7-300 wird hier kein gültiger Wert eingetragen)
OB121_PRG_ADDR	WORD	Relativadresse des Fehler verursachenden MC7-Befehls (bei S7-300 wird hier kein gültiger Wert eingetragen)
OB121_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Die vom Fehlercode abhängigen Variablen haben folgende Bedeutung:

Fehlercode	Variable	Bedeutung
B#16#21		BCD-Konvertierungsfehler
	OB121_FLT_REG	Kennung für das betroffene Register (W#16#0000: Akku 1)
B#16#22		Bereichslängenfehler beim Lesen
	OB121_RESERVED_1	• Bit 7 bis 4 Zugriffsart: – 0: Bitzugriff, – 1: Bytezugriff, – 2: Wortzugriff, – 3: Doppelwortzugriff • Bit 3 bis 0 Speicherbereich: – 0: Peripheriebereich – 1: Prozessabbild der Eingänge – 2: Prozessabbild der Ausgänge – 3: Merker – 4: Global-DB – 5: Instanz-DB – 6: eigene Lokaldaten – 7: Lokaldaten des Aufrufers
B#16#23		Bereichslängenfehler beim Schreiben
	OB121_RESERVED_1	wie beim Fehlercode B#16#22
B#16#28		Lesender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich 0 ist
	OB121_RESERVED_1	wie beim Fehlercode B#16#22

Fehlercode	Variable	Bedeutung
B#16#29		Schreibender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich 0 ist Fehlerhafte Byteadresse. Der Datenbereich und die Zugriffsart sind OB121_RESERVED_1 zu entnehmen
	OB121_RESERVED_1	wie beim Fehlercode B#16#22
B#16#24		Bereichsfehler beim Lesen
	OB121_FLT_REG	Enthält im Low Byte die Kennung des unzulässigen Bereichs (B#16#86 eigener Lokaldatenbereich)
B#16#25		Bereichsfehler beim Schreiben
	OB121_FLT_REG	Enthält im Low Byte die Kennung des unzulässigen Bereichs (B#16#86 eigener Lokaldatenbereich)
B#16#26		Fehler bei Timernummer
	OB121_FLT_REG	Unzulässige Nummer
B#16#27		Fehler bei Zählernummer
	OB121_FLT_REG	Unzulässige Nummer
B#16#30		Schreibender Zugriff auf einen schreibgeschützten Global-DB
	OB121_FLT_REG	Unzulässige DB-Nummer
B#16#31		Schreibender Zugriff auf einen schreibgeschützten Instanz-DB
	OB121_FLT_REG	Unzulässige DB-Nummer
B#16#32		DB-Nummernfehler beim Zugriff auf einen Global-DB
	OB121_FLT_REG	Unzulässige DB-Nummer
B#16#33		DB-Nummernfehler beim Zugriff auf einen Instanz-DB
	OB121_FLT_REG	Unzulässige DB-Nummer
B#16#34		Nummernfehler beim FC-Aufruf
	OB121_FLT_REG	FC-Nummer
B#16#35		FB-Nummernfehler beim FB-Aufruf
	OB121_FLT_REG	FB-Nummer
B#16#3A		Zugriff auf einen nicht geladenen DB; die DB-Nummer liegt im zulässigen Bereich
	OB121_FLT_REG	DB-Nummer
B#16#3C		Zugriff auf eine nicht geladene FC; die FC-Nummer liegt im zulässigen Bereich
	OB121_FLT_REG	FC-Nummer
B#16#3D		Zugriff auf eine nicht vorhandene Anweisung; die SFC-Nummer liegt im zulässigen Bereich
	OB121_FLT_REG	SFC-Nummer
B#16#3E		Zugriff auf einen nicht geladenen FB; die FB-Nummer liegt im zulässigen Bereich
	OB121_FLT_REG	FB-Nummer
B#16#3F		Zugriff auf einen nicht vorhandenen SFB; die SFB-Nummer liegt im zulässigen Bereich
	OB121_FLT_REG	SFB-Nummer

Programmieren des OB 121

Den OB 121 müssen Sie zu den Bausteinen Ihrer CPU hinzufügen. Schreiben Sie das Programm, das im OB 121 bearbeitet werden soll, in den erzeugten Baustein und laden Sie ihn als Teil Ihres Anwenderprogramms in die CPU.

Sie können den OB 121 beispielsweise wie folgt nutzen:

- Sie werten die Startinformation des OB 121 aus.
- Sie tragen die Fehlerursache in einen Melde-Datenbaustein ein.

Peripheriezugriffsfehler-OB (OB 122)

Beschreibung

Das Betriebssystem der CPU ruft den OB 122 auf, wenn beim Zugreifen auf Daten einer Baugruppe ein Fehler auftritt. Wenn die CPU beispielsweise einen Lesefehler beim Zugriff auf Daten einer Signalbaugruppe erkennt, dann ruft das Betriebssystem den OB 122 auf.

Funktionsweise des Peripheriezugriffsfehler-OB

Der OB 122 läuft in derselben Prioritätsklasse wie der unterbrochene Baustein. Ist der OB 122 nicht programmiert, dann wechselt die CPU den Betriebszustand von RUN nach STOP.

S7 verfügt über die folgenden Anweisungen, mit denen Sie Startereignisse des OB 122 maskieren und demaskieren können, während Ihr Programm bearbeitet wird:

- Die Anweisung "MSK_FLT" maskiert bestimmte Fehlercodes.
- Die Anweisung "DMSK_FLT" demaskiert die Fehlercodes, die von der Anweisung "MSK_FLT" maskiert wurden.
- Die Anweisung "READ_ERR" liest das Ereignisstatusregister.

Lokaldaten des Peripheriezugriffsfehler-OB

Die folgende Tabelle beinhaltet die temporären (TEMP) Variablen des Peripheriezugriffsfehler-OB. Als Variablennamen wurden die Defaultnamen des OB 122 gewählt.

Variable	Datentyp	Beschreibung
OB122_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#29
OB122_SW_FLT	BYTE	Fehlercode: • B#16#42: Peripheriezugriffsfehler, lesend. • B#16#43: Peripheriezugriffsfehler, schreibend.
OB122_PRIORITY	BYTE	Prioritätsklasse des OB, in dem der Fehler aufgetreten ist Bei S7-1500-CPUs: Priorität, Defaultwert: 7
OB122_OB_NUMBR	BYTE	OB-Nummer (122)

Variable	Datentyp	Beschreibung
OB122_BLK_TYPE	BYTE	Bausteintyp, in dem der Fehler aufgetreten ist (B#16#88: OB, B#16#8C: FC, B#16#8E: FB) (bei S7-300 wird hier kein gültiger Wert eingetragen)
OB122_MEM_AREA	BYTE	Speicherbereich und Zugriffsart: • Bit 7 bis 4: Zugriffsart – 0: Bitzugriff – 1: Bytezugriff – 2: Wortzugriff – 3: Doppelwortzugriff • Bit 3 bis 0:Speicherbereich – 0: Peripheriebereich – 1: Prozessabbild der Eingänge – 2: Prozessabbild der Ausgänge
OB122_MEM_ADDR	WORD	Adresse im Speicher, an der der Fehler aufgetreten ist
OB122_BLK_NUM	WORD	Nummer des Bausteins mit dem Fehler verursachenden MC7-Befehl (bei S7-300 wird hier kein gültiger Wert eingetragen)
OB122_PRG_ADDR	WORD	Relativadresse des Fehler verursachenden MC7-Befehls (bei S7-300 wird hier kein gültiger Wert eingetragen)
OB122_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Beispielprogramm für den OB 122

Im folgenden Beispielprogramm wird ein Ersatzwert in der Anweisung REPL_VAL bereitgestellt. Wenn eine Eingabebaugruppe ausfällt, erzeugt die Bearbeitung der Anweisung L PEB0 einen synchronen Fehler und startet den OB 122. Standardmäßig liest der Ladebefehl den Wert 0 ein. Sie können jedoch mit der Anweisung "REPL_VAL" einen beliebigen zum Prozess passenden Ersatzwert vorgeben. Diese Anweisung ersetzt den AKKU-Inhalt durch den vorgegebenen Ersatzwert.

```
AWL                                  Beschreibung
L    B#16#42                         //Kennung für das Ereignis "Peripheriezugriffsfehler lesend"
L    #OB122_SW_FLT                   //Fehlercode von OB 122
==I                                  //Wenn gleich,
SPB    Qfeh                          //dann springe zu Qfeh
L    B#16#43                         //Kennung für das Ereignis "Peripheriezugriffsfehler schrei-
                                     bend"
<>I                                  //Wenn ungleich mit dem Fehlercode von OB 122,
SPB    STOP                          //dann springe zu STOP
Qfeh:
CALL    "REPL_VAL"                   //Aufruf der Anweisung "REPL_VAL"
    VAL:= DW#16#2912                 //Ersatzwert, der in AKKU 1 geladen wird
    RET_VAL:= #Fehler                //Speicherung des Rückgabewerts in #Fehler
```

```
AWL                                     Beschreibung
L    #Fehler                            //Rückgabewert der Anweisung "REPL_VAL"
L    0
==I                                     //Wenn gleich 0 (Es ist kein Fehler aufgetreten),
BEB                                     //dann beende den Baustein
STOP:
CALL    "STP"                           //Aufruf der Anweisung "STP": Die CPU wird in den Betriebszu-
                                        stand STOP versetzt.
```

Ereignisse

Ereignisse und Ereignis-ID

Ereignis

Alle Ereignisse sind innerhalb der SIMATIC S7 durchnummeriert. Damit können Sie einem Ereignis einen Meldetext zuordnen.

Ereignis-ID

Jedem Ereignis ist eine Ereignis-ID zugeordnet. Die Ereignis-ID ist wie folgt aufgebaut:

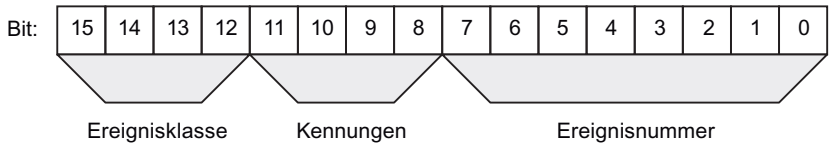

Bild 1-19 Aufbau der Ereignis-ID

Ereignisklasse

Die Ereignisklassen sind wie folgt aufgeteilt:

Nummer	Ereignisklasse
1	Standard-OB-Ereignisse
2	Synchrone Fehlerereignisse
3	Asynchrone Fehlerereignisse
4	Betriebszustandsübergänge
5	Betriebszustandsablaufereignis
6	Kommunikationsereignisse
7	Ereignisse für fehlersichere und hochverfügbare Systeme
8	Normierte Diagnosedaten auf Baugruppen
9	Vordefinierte Anwenderereignisse
A, B	Frei definierbare Ereignisse

Nummer	Ereignisklasse
C, D, E	reserviert
F	Ereignisse für Nicht-Zentralbaugruppen (z. B. CP, FM)

Kennung

Die Kennung dient dazu, um Ereignisse in Ihrer Art unterscheiden zu können. Die vier Kennbits haben die folgende Bedeutung:

Bit-Nr. in der Ereignis-ID	Bedeutung
8	= 0: gehendes Ereignis = 1: kommendes Ereignis
9	= 1: Eintrag in Diagnosepuffer
10	= 1: interner Fehler
11	= 1: externer Fehler

Ereignisklasse 1 - Standard-OB-Ereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#1381	Manuelle Neustart- (Warmstart-) -Anforderung
W#16#1382	Automatische Neustart- (Warmstart-) -Anforderung
W#16#1383	Manuelle Wiederanlaufanforderung
W#16#1384	Automatische Wiederanlaufanforderung
W#16#1385	Manuelle Kaltstartanforderung
W#16#1386	Automatische Kaltstartanforderung
W#16#1387	Master-CPU: Manuelle Kaltstartanforderung
W#16#1388	Master-CPU: Automatische Kaltstartanforderung
W#16#138A	Master-CPU: Manuelle Neustart- (Warmstart-) -Anforderung
W#16#138B	Master-CPU: Automatische Neustart- (Warmstart-) -Anforderung
W#16#138C	Reserve-CPU: Manuelle Anlaufanforderung
W#16#138D	Reserve-CPU: Automatische Anlaufanforderung

Ereignisklasse 2 - Synchrone Fehlerereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis	OB
W#16#2521	BCD - Wandlungsfehler	OB 121
W#16#2522	Bereichslängenfehler beim Lesen	OB 121
W#16#2523	Bereichslängenfehler beim Schreiben	OB 121
W#16#2524	Bereichsfehler beim Lesen	OB 121

Ereignis-ID	Ereignis	OB
W#16#2525	Bereichsfehler beim Schreiben	OB 121
W#16#2526	Timer-Nummernfehler	OB 121
W#16#2527	Zähler-Nummernfehler	OB 121
W#16#2528	Ausrichtungsfehler beim Lesen	OB 121
W#16#2529	Ausrichtungsfehler beim Schreiben	OB 121
W#16#2530	Schreibfehler beim Zugriff auf den DB	OB 121
W#16#2531	Schreibfehler beim Zugriff auf den DI	OB 121
W#16#2532	Bausteinnummernfehler beim Aufschlagen eines DB	OB 121
W#16#2533	Bausteinnummernfehler beim Aufschlagen eines DI	OB 121
W#16#2534	Bausteinnummernfehler beim FC-Aufruf	OB 121
W#16#2535	Bausteinnummernfehler beim FB-Aufruf	OB 121
W#16#253A	DB nicht geladen	OB 121
W#16#253C	FC nicht geladen	OB 121
W#16#253D	SFC nicht geladen	OB 121
W#16#253E	FB nicht geladen	OB 121
W#16#253F	SFB nicht geladen	OB 121
W#16#2942	Peripherie-Zugriffsfehler, lesend	OB 122
W#16#2943	Peripherie-Zugriffsfehler, schreibend	OB 122

Ereignisklasse 3 - Asynchrone Fehlerereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis	OB
W#16#3501	Zykluszeitüberschreitung	OB 80
W#16#3502	Anwenderschnittstelle (OB bzw. FRB) -Anforderungs-fehler	OB 80
W#16#3505	Uhrzeitalarm(e) abgelaufen durch Uhrzeitsprung	OB 80
W#16#3506	Uhrzeitalarm(e) abgelaufen bei Wiedereintritt in RUN nach HALT	OB 80
W#16#3507	Mehrfacher OB-Anforderungsfehler verursachte einen internen Puffer-Überlauf	OB 80
W#16#3508	Taktsynchronalarm-Zeitfehler	OB 80
W#16#3509	Alarmverlust durch zu hohe Alarmlast	OB 80
W#16#350A	Wiedereintritt in RUN nach CiR	OB 80
W#16#350B	Technologiesynchronalarm-Zeitfehler	OB 80
W#16#3921/3821	BATTF: Ausfall mindestens einer Pufferbatterie im Zentralgerät/ beseitigt Hinweis: Das kommende Ereignis tritt nur beim Ausfall einer der Batterien (bei redundanten Pufferbatterien) auf. Fällt anschließend auch noch die andere Batterie aus, tritt das Ereignis nicht nochmals auf.	OB 81
W#16#3922/3822	BAF: Ausfall der Pufferspannung im Zentralgerät/ beseitigt	OB 81

Ereignis-ID	Ereignis	OB
W#16#3923/3823	Ausfall der 24V-Versorgung im Zentralgerät/ beseitigt	OB 81
W#16#3925/3825	BATTF: Ausfall mindestens einer Pufferbatterie in einem redundanten Zentralgerät/ beseitigt	OB 81
W#16#3926/3826	BAF: Ausfall der Pufferspannung in einem redundanten Zentralgerät/ beseitigt	OB 81
W#16#3927/3827	Ausfall der 24V-Versorgung in einem redundanten Zentralgerät/ beseitigt	OB 81
W#16#3931/3831	BATTF: Ausfall mindestens einer Pufferbatterie in mindestens einem Erweiterungsgerät/ beseitigt	OB 81
W#16#3932/3832	BAF: Ausfall der Pufferspannung in mindestens einem Erweiterungsgerät/ beseitigt	OB 81
W#16#3933/3833	Ausfall der 24V-Versorgung in mindestens einem Erweiterungsgerät/ beseitigt	OB 81
W#16#3942	Diagnosealarm (Modul/Submodul gestört oder Wartung erforderlich oder beides)	OB 82
W#16#3842	Diagnosealarm (Modul/Submodul ok)	OB 82
W#16#3951	PROFINET IO-Modul gezogen	OB 83
W#16#3954	PROFINET IO-Submodul/Modul gezogen	OB 83
W#16#3854	PROFINET IO-Submodul/Modul gesteckt und entspricht parametriertem Submodul/Modul	OB 83
W#16#3855	PROFINET IO-Submodul/Modul gesteckt, entspricht aber nicht dem parametrierten Submodul/Modul	OB 83
W#16#3856	PROFINET IO-Submodul/Modul gesteckt, jedoch Fehler bei Baugruppenparametrierung	OB 83
W#16#3857	PROFINET IO-Submodul/Modul gesteckt, jedoch mit Störung oder Wartung	OB 83
W#16#3858	PROFINET IO-Submodul Zugriffsfehler beseitigt	OB 83
W#16#3861	Baugruppe / Schnittstellenmodul gesteckt, Baugruppentyp o. k.	OB 83
W#16#3961	Baugruppe / Schnittstellenmodul gezogen bzw. nicht ansprechbar	OB 83
W#16#3863	Baugruppe / Schnittstellenmodul gesteckt, jedoch falscher Baugruppentyp	OB 83
W#16#3864	Baugruppe / Schnittstellenmodul gesteckt, jedoch gestört (Baugruppenkennung nicht lesbar)	OB 83
W#16#3865	Baugruppe gesteckt, jedoch Fehler bei Baugruppenparametrierung	OB 83
W#16#3866	Baugruppe wieder ansprechbar, Lastspannungsfehler beseitigt	OB 83
W#16#3966	Baugruppe nicht ansprechbar, Lastspannungsfehler	OB 83
W#16#3367	Beginn Umparametrieren einer Baugruppe	OB 83
W#16#3267	Ende Umparametrieren einer Baugruppe	OB 83
W#16#3968	Umparametrieren einer Baugruppe mit Fehler beendet	OB 83
W#16#3571	Zu große Schachtelungstiefe von Klammerebenen	OB 88
W#16#3572	Zu große Schachtelungstiefe von Master Control Relais	OB 88

Ereignis-ID	Ereignis	OB
W#16#3573	Zu große Schachtelungstiefe bei Synchronfehlern	OB 88
W#16#3574	Zu große Schachtelungstiefe von Bausteinaufrufen (U-Stack)	OB 88
W#16#3575	Zu große Schachtelungstiefe von Bausteinaufrufen (B-Stack)	OB 88
W#16#3576	Fehler beim Allokieren von Lokaldaten	OB 88
W#16#3578	Unbekannte Anweisung	OB 88
W#16#357A	Sprunganweisung mit Ziel außerhalb des Bausteins	OB 88
W#16#3981	Schnittstellenfehler, kommend	-
W#16#3881	Schnittstellenfehler, gehend	-
W#16#3582	Speicherfehler vom Betriebssystem erkannt und beseitigt	OB 84
W#16#3583	Häufung von erkannten und korrigierten Speicherfehlern	OB 84
W#16#3585	Fehler im PC-Betriebssystem (nur bei Win AC-Controllern)	OB 84
W#16#3986	Leistung einer H-Sync-Kopplung beeinträchtigt	OB 84
W#16#3587	Mehrbitspeicherfehler erkannt und korrigiert	OB 84
W#16#35A1	Anwenderschnittstelle (OB bzw. FRB) nicht vorhanden	OB 85
W#16#35A2	OB nicht geladen (gestartet durch SFC oder durch Besy aufgrund Projektierung)	OB 85
W#16#35A3	Fehler beim Zugriff durch Besy auf einen Baustein	OB 85
W#16#35A4	PROFInet Interface-DB nicht ansprechbar	OB 85
W#16#34A4	PROFInet Interface-DB wieder ansprechbar	OB 85
W#16#39B1	Peripheriezugriffsfehler bei Prozessabbildaktualisierung der Eingänge	OB 85
W#16#39B2	Peripheriezugriffsfehler bei der Übertragung des Prozessabbilds zu den Ausgabebaugruppen	OB 85
W#16#39B3/38B3	Peripheriezugriffsfehler bei Prozessabbildaktualisierung der Eingänge	OB 85
W#16#39B4/38B4	Peripheriezugriffsfehler bei der Prozessabbildaktualisierung der Ausgänge	OB 85
W#16#38C1	Wiederkehr Erweiterungsgerät (1 bis 21), gehend	OB 86
W#16#39C1	Ausfall Erweiterungsgerät (1 bis 21), kommend	OB 86
W#16#38C2	Erweiterungsgerätwiederkehr mit Abweichung Soll-/Istausbau	OB 86
W#16#39C3	Dezentrale Peripherie: Mastersystemausfall kommend	OB 86
W#16#39C4	Dezentrale Peripherie: Station ausgefallen, kommend	OB 86
W#16#38C4	Dezentrale Peripherie: Station ausgefallen, gehend	OB 86
W#16#38C5	Dezentrale Peripherie: Stationswiederkehr mit Störung	OB 86
W#16#38C6	Erweiterungsgerätewiederkehr, jedoch Fehler bei Baugruppenparametrierung	OB 86
W#16#38C7	DP: Stationswiederkehr, jedoch Fehler bei Baugruppenparametrierung	OB 86
W#16#38C8	DP: Stationswiederkehr mit Abweichung Soll-/Istausbau	OB 86
W#16#39CA	PROFINET IO-Systemausfall	OB 86

Ereignis-ID	Ereignis	OB
W#16#39CB	PROFINET IO-Stationsausfall	OB 86
W#16#38CB	PROFINET IO-Stationswiederkehr	OB 86
W#16#38CC	PROFINET IO-Stationswiederkehr mit Störung oder Wartung	OB 86
W#16#39CD	PROFINET IO-Stationswiederkehr, Sollausbau weicht von Istausbau ab.	OB 86
W#16#39CE	PROFINET IO-Stationswiederkehr, Fehler bei der Baugruppenparametrierung	OB 86
W#16#33CF/32CF	Ausfall/Wiederkehr eines PROFINET IO-Geräts infolge Deaktivierung/Aktivierung durch die Anweisung D_ACT_DP	OB 86
W#16#39F8/38F8	Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Device	OB 86
W#16#38F9	Wiederkehr eines Teils der Submodule eines PROFINET I-Device mit Device-Konfigurationsunterschied	OB 86
W#16#35D2	Senden der Diagnoseeinträge derzeit nicht möglich	OB 87
W#16#35D3	Synchronisationstelegramme können nicht gesendet werden	OB 87
W#16#35D4	Unzulässiger Uhrzeitsprung durch Uhrzeitsynchronisation	OB 87
W#16#35D5	Fehler bei Übernahme der Synchronisationszeit	OB 87
W#16#35E1	Falsche Telegrammkennung bei GD	OB 87
W#16#35E2	GD-Paketstatus nicht in DB eintragbar	OB 87
W#16#35E3	Telegrammlängenfehler bei GD	OB 87
W#16#35E4	Unzulässige GD-Paketnummer empfangen	OB 87
W#16#35E5	Fehler beim Zugriff auf DB bei Kommunikations-Anweisungen für projektierte S7-Verbindungen	OB 87
W#16#35E6	GD-Gesamtstatus nicht in DB eintragbar	OB 87

Ereignisklasse 4 - Stopereignisse und andere Betriebszustandsübergänge

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#4300	NETZ-EIN gepuffert
W#16#4301	Betriebszustandsübergang von STOP nach ANLAUF
W#16#4302	Betriebszustandsübergang von ANLAUF nach RUN
W#16#4303	STOP durch Stopschalter-Bedienung
W#16#4304	STOP durch PG-Stop-Bedienung oder wegen der Anweisung STOP
W#16#4305	HALT: Haltepunkt erreicht
W#16#4306	HALT: Haltepunkt verlassen
W#16#4307	Start Urlöschen durch PG-Bedienung
W#16#4308	Start Urlöschen durch Schalterbedienung
W#16#4309	Start Urlöschen automatisch (ungepuffertes NETZ-EIN)

Ereignis-ID	Ereignis
W#16#430A	HALT verlassen, Übergang in den STOP
W#16#430D	STOP durch andere CPU bei Multicomputing
W#16#430E	Urlöschen durchgeführt
W#16#430F	STOP der Baugruppe durch STOP einer CPU
W#16#4510	STOP wegen Verletzung des Datumsbereichs der CPU
W#16#4318	Beginn des CiR-Vorgangs
W#16#4319	CiR-Vorgang beendet
W#16#4520	DEFEKT: STOP nicht erreichbar
W#16#4521	DEFEKT: Ausfall des Befehlsbearbeitungsprozessors
W#16#4522	DEFEKT: Ausfall des Uhrenbausteins
W#16#4523	DEFEKT: Ausfall des Zeittaktgebers
W#16#4524	DEFEKT: Ausfall der Zeitzellenaktualisierung
W#16#4525	DEFEKT: Ausfall der Synchronisation bei Multicomputing
W#16#4926	DEFEKT: Ausfall der Zeitüberwachung bei Peripheriezugriffen
W#16#4527	DEFEKT: Ausfall der Peripheriezugriffsüberwachung
W#16#4528	DEFEKT: Ausfall der Zykluszeitüberwachung
W#16#4530	DEFEKT: Speichertestfehler im internen Speicher
W#16#4931	STOP bzw. DEFEKT: Speichertestfehler im Modulspeicher
W#16#4532	DEFEKT: Ausfall von Kernressourcen
W#16#4933	Quersummenfehler
W#16#4934	DEFEKT: Speicher nicht vorhanden
W#16#4935	DEFEKT: Abbruch durch Watchdog/processor exceptions
W#16#4536	DEFEKT: Betriebsartenschalter defekt
W#16#4540	STOP: Speichererweiterung des internen Arbeitsspeichers nicht lückenlos. Erste Speichererweiterung ist zu klein oder fehlt.
W#16#4541	STOP durch das Prioritätsklassen-Ablaufsystem
W#16#4542	STOP durch Objektverwaltungssystem
W#16#4543	STOP durch Test und Inbetriebsetzung
W#16#4544	STOP durch Diagnosesystem
W#16#4545	STOP durch Kommunikationssystem
W#16#4546	STOP durch CPU-Speicherverwaltung
W#16#4547	STOP durch Prozessabbildverwaltung
W#16#4548	STOP durch Peripherieverwaltung
W#16#4949	STOP wegen Dauer-Prozessalarm
W#16#454A	STOP durch Projektierung: Ein per Projektierung abgewählter OB war beim Anlauf in der CPU geladen.
W#16#494D	STOP durch Peripheriefehler
W#16#494E	STOP durch Netzausfall
W#16#494F	STOP durch Konfigurationsfehler
W#16#4550	DEFEKT: interner Systemfehler
W#16#4555	Wiederanlauf nicht möglich, da Überwachungszeitgrenze abgelaufen
W#16#4556	STOP: Urlöschanforderung durch Kommunikation / Dateninkonsistenz
W#16#4357	Baugruppenüberwachungszeit gestartet

Ereignis-ID	Ereignis
W#16#4358	Alle Baugruppen sind betriebsbereit
W#16#4959	Nicht alle Baugruppen sind betriebsbereit
W#16#4562	STOP durch Programmierfehler (OB nicht geladen oder nicht möglich)
W#16#4563	STOP durch Peripheriezugriffsfehler (OB nicht geladen oder nicht möglich)
W#16#4567	STOP durch H-Ereignis
W#16#4568	STOP durch Zeitfehler (OB nicht geladen oder nicht möglich)
W#16#456A	STOP durch Diagnosealarm (OB nicht geladen oder nicht möglich)
W#16#456B	STOP durch Ziehen/Stecken (OB nicht geladen oder nicht möglich)
W#16#456C	STOP durch CPU-Hardwarefehler (OB nicht geladen oder nicht möglich)
W#16#456D	STOP durch Programmablauffehler (OB nicht geladen oder nicht möglich)
W#16#456E	STOP durch Kommunikationsfehler (OB nicht geladen oder nicht möglich)
W#16#456F	STOP durch Baugruppenträgerausfall (OB nicht geladen oder nicht möglich)
W#16#4570	STOP durch Bearbeitungsabbruch (OB nicht geladen oder nicht möglich)
W#16#4571	STOP durch Klammerstackfehler
W#16#4572	STOP durch Master-Control-Relais-Stackfehler
W#16#4573	STOP durch Überschreiten der Schachtelungstiefe bei Synchronfehlern
W#16#4574	STOP durch zu große U-Stack-Verschachtelung im Prioritätsklassen-Stack
W#16#4575	STOP durch zu große B-Stack-Verschachtelung im Prioritätsklassen-Stack
W#16#4576	STOP durch Fehler beim Allokieren von Lokaldaten
W#16#4578	STOP durch unbekannten Opcode
W#16#457A	STOP durch Codelängenfehler
W#16#457B	STOP durch nicht geladenen DB bei Onboard-Peripherie
W#16#497C	STOP durch integrierte Technologie
W#16#457D	Urlöschanforderung, weil die Version der internen Schnittstelle zur integrierten Technologie geändert wurde
W#16#457F	STOP durch STOP-Befehl
W#16#4580	STOP: Backup-Pufferinhalt inkonsistent (kein RUN-Übergang)
W#16#4590	STOP wegen Überlast der Internen Funktionen
W#16#49A0	STOP wegen Parametrierfehler oder unzulässige Differenz zwischen Soll- und Istausbau: Anlauf gesperrt
W#16#49A1	STOP wegen Parametrierfehler: Urlöschanforderung
W#16#49A2	STOP wegen Fehler beim Nachparametrieren: Anlauf gesperrt
W#16#49A3	STOP wegen Fehler beim Nachparametrieren: Urlöschanforderung
W#16#49A4	STOP: Inkonsistenz der Projektierungsdaten
W#16#49A5	STOP: Dezentrale Peripherie: Unstimmigkeiten der geladenen Projektierinformation
W#16#49A6	STOP: Dezentrale Peripherie: ungültige Projektierinformation
W#16#49A7	STOP: Dezentrale Peripherie: Projektierinformation nicht vorhanden
W#16#49A8	STOP: Fehleranzeige der Anschaltung für Dezentrale Peripherie
W#16#43B0	Firmwareupdate / -sicherung erfolgreich durchgeführt
W#16#49B1	Fehlerhafte Firmwareupdate-Daten
W#16#49B2	Firmwareupdate: Hardwarestand passt nicht zur Firmware
W#16#49B3	Firmwareupdate: Baugruppentyp passt nicht zur Firmware

Ereignis-ID	Ereignis
W#16#43B4	Fehler bei der Firmware-Sicherung
W#16#43B6	Abbruch des Firmware-Updates von redundanten Baugruppen
W#16#43D0	Abweisung ANKOPPELN wegen Verletzung von Koordinierungsregeln
W#16#43D1	Abbruch der Sequenz ANKOPPELN/AUFDATEN
W#16#49D2	STOP der Reserve-CPU wegen STOP der Master-CPU während der Ankopplung
W#16#43D3	STOP einer Reserve-CPU
W#16#49D4	STOP eines Masters, da Partner-CPU auch Master ist (Kopplungsfehler)
W#16#43D5	Abweisung ANKOPPELN wegen ungleichem Speicherausbau des Teil-AS
W#16#43D6	Abweisung ANKOPPELN wegen ungleichem Systemprogramm des Teil-AS
W#16#43D7	Abweisung ANKOPPELN wegen Konfigurationsänderung
W#16#49D8	STOP/Fehlersuchbetrieb/DEFEKT: Hardwarefehler durch anderen Fehler erkannt
W#16#49D9	STOP wegen Synchronisationsmodul-Fehler
W#16#49DA	STOP wegen Synchronisationsfehler zwischen H-CPUs
W#16#43DC	Abbruch beim Ankoppeln mit Umschalten
W#16#43DD	Abweisung ANKOPPELN wegen laufender Test- oder anderer Online-Funktionen
W#16#43DE	Abbruch des Aufdatvorgangs wegen Überschreitung einer Überwachungszeit beim n-ten Versuch, erneuter Aufdatversuch initiiert
W#16#43DF	Endgültiger Abbruch des Aufdatvorgangs wegen Überschreitung einer Überwachungszeit nach der maximalen Anzahl von Versuchen, erneute Bedienung erforderlich
W#16#43E0	Wechsel von Solobetrieb nach Ankoppeln
W#16#43E1	Wechsel von Ankoppeln nach Aufdaten
W#16#43E2	Wechsel vom Systemzustand Aufdaten in Redundant
W#16#43E3	Master-CPU: Wechsel vom Systemzustand Redundant nach Solobetrieb
W#16#43E4	Reserve-CPU: Wechsel vom Systemzustand Redundant nach FEHLERSUCHE
W#16#43E5	Reserve-CPU: Wechsel von FEHLERSUCHE nach Ankoppeln oder STOP
W#16#43E6	Abbruch Ankoppeln der Reserve-CPU
W#16#43E7	Abbruch Aufdaten der Reserve-CPU
W#16#43E8	Reserve-CPU: Wechsel von Ankoppeln nach Anlauf
W#16#43E9	Reserve-CPU: Wechsel von Anlauf nach Aufdaten
W#16#43F1	Reserve-Master-Umschaltung
W#16#43F2	Kopplung inkompatibler H-CPUs durch Systemprogramm blockiert
W#16#42F3	Prüfsummenfehler vom Betriebssystem erkannt und korrigiert
W#16#43F4	Reserve-CPU: Sperre des Ankoppelns/Aufdatens mittels der Anweisung H_CTRL in der Master-CPU

Ereignisklasse 5 - Betriebszustands-Ablaufereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#530D	Neue Anlaufinformation im Betriebszustand STOP
W#16#510F	Bei WinLC ist ein Problem aufgetreten, das zum STOP oder Defekt der CPU führte.
W#16#5311	Anlauf trotz fehlender Fertigmeldung der Baugruppe(n)
W#16#5545	Beginn des Umparametrierens im Rahmen einer Anlagenänderung im laufenden Betrieb
W#16#5445	Ende des Umparametrierens im Rahmen einer Anlagenänderung im laufenden Betrieb
W#16#5961	Parametrierfehler
W#16#5962	Parametrierfehler mit Anlaufhindernis
W#16#5963	Parametrierfehler mit Urlöschanforderung
W#16#5966	Parametrierfehler beim Umschalten
W#16#5967	Parametrierfehler: Unbekannte Projektierungsdaten
W#16#5968/5868	Notfalladresse einer Ethernetschnittstelle aktiviert/deaktiviert
W#16#5969	Parametrierfehler mit Anlaufhindernis
W#16#596A	PROFINET IO: IP-Adresse eines IO-Device bereits vorhanden
W#16#596B	IP-Adresse einer Ethernet-Schnittstelle bereits vorhanden
W#16#596C	Name einer Ethernet-Schnittstelle bereits vorhanden
W#16#596D	Die vorhandene Netzkonfiguration passt nicht zu den Systemanforderungen oder der Projektierung.
W#16#5371	Dezentrale Peripherie: Ende der Synchronisation mit einem DP-Master
W#16#5979/5879	Diagnosemeldung von DP-Anschaltung: EXTF-LED an/aus
W#16#597C	DP-Kommando Global Control ausgefallen oder verschoben
W#16#5380	Diagnosepuffereinträge von Alarm- und asynchronen Fehlerereignissen gesperrt
W#16#5581	Eine oder mehrere Lizenzen für Runtime-Software fehlen.
W#16#5481	Alle Lizenzen für Runtime-Software sind wieder vollständig.
W#16#558A	Unterschied zwischen der MLFB der projektierten und der gesteckten CPU
W#16#558B	Unterschied zwischen der Firmware-Version der projektierten und der gesteckten CPU
W#16#5966	Parametrierfehler beim Umschalten
W#16#597C	DP-Kommando Global Control ausgefallen oder verschoben
W#16#5395	Dezentrale Peripherie: Rücksetzen eines DP-Masters
W#16#5598	Beginn potentieller Inkonsistenz mit DP-Mastersystemen durch CiR
W#16#5498	Ende potentieller Inkonsistenz mit DP-Mastersystemen durch CiR
W#16#59A0	Alarm in der CPU nicht zuordenbar
W#16#59A1	Konfigurationsfehler der integrierten Technologie
W#16#53A2	Laden der Technologie-Firmware erfolgreich durchgeführt
W#16#59A3	Fehler beim Laden der integrierten Technologie
W#16#53A4	Laden des Technologie-DB nicht erfolgreich

Ereignis-ID	Ereignis
W#16#55A5	Versionskonflikt der internen Schnittstelle zur integrierten Technologie
W#16#55A6	Die Maximalanzahl der Technologieobjekte wurde überschritten.
W#16#55A7	Es ist bereits ein Technologie-DB dieses Typs vorhanden.
W#16#53FF	Rücksetzen in den Auslieferungszustand

Ereignisklasse 6 – Kommunikationsereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#6500	Verbindungsreferenz (ID) auf Baugruppe doppelt vorhanden
W#16#6501	Verbindungsressourcen nicht ausreichend
W#16#6502	Fehler in der Verbindungsbeschreibung
W#16#6905/6805	Ressourcenproblem bei fest projektierten Verbindungen/ beseitigt
W#16#6510	CFB-Strukturfehler im Instanz-DB bei Auswertung EPROM erkannt
W#16#6514	GD-Paketnummer auf der Baugruppe doppelt vorhanden
W#16#6515	Inkonsistente Längenangaben in GD-Projektierungsinformation
W#16#6316	Schnittstellenfehler beim Hochlauf des AS
W#16#6521	Weder Modul noch interner Speicher vorhanden
W#16#6522	Unzulässiges Modul: Modultausch und Urlöschen erforderlich
W#16#6523	Urlöschanforderung durch Fehler bei Zugriff auf Modul
W#16#6524	Urlöschanforderung durch Fehler im Bausteinkopf
W#16#6526	Urlöschanforderung wegen Speichertausch
W#16#6527	Speichertausch, deshalb kein Wiederanlauf möglich
W#16#6528	Objekthandlingsfunktion im STOP/HALT, kein Wiederanlauf möglich
W#16#6529	Kein Anlauf möglich während der Funktion "Anwenderprogramm laden"
W#16#652A	Kein Anlauf, da Baustein im Anwenderspeicher doppelt vorhanden
W#16#652B	Kein Anlauf, da Bausteinlänge zu groß für Modul: Modultausch erforderlich
W#16#652C	Kein Anlauf wegen unzulässigem OB auf dem Modul
W#16#6532	Kein Anlauf wegen unzulässiger Projektierinformation auf Modul
W#16#6533	Urlöschanforderung durch ungültigen Modulinhalt
W#16#6534	Kein Anlauf: Baustein auf Modul mehrfach vorhanden
W#16#6535	Kein Anlauf: Nicht genügend Speicher, um Baustein aus Modul aufzunehmen
W#16#6536	Kein Anlauf: Modul enthält eine unzulässige Bausteinnummer
W#16#6537	Kein Anlauf: Modul enthält einen Baustein unzulässiger Länge
W#16#6538	Lokaldaten oder Schreibschutzkennung (bei DB) eines Bausteins für CPU unzulässig
W#16#6539	Unzulässiger Befehl im Baustein (vom Compiler erkannt)
W#16#653A	Urlöschanforderung, da OB-Lokaldaten auf Modul zu kurz sind
W#16#6543	Kein Anlauf: Bausteintyp unzulässig
W#16#6544	Kein Anlauf: Attribut "ablaufrelevant" unzulässig
W#16#6545	Erstellungssprache unzulässig

Ereignis-ID	Ereignis
W#16#6546	Maximale Anzahl der Bausteine eines Bausteintyps erreicht
W#16#6547	Parametrierfehler beim Parametrieren von Baugruppen (nicht über P-Bus, sondern Abbruch Download)
W#16#6548	Plausibilitätsfehler bei Bausteinprüfung
W#16#6549	Strukturfehler im Baustein
W#16#6550	Ein Baustein hat im Prüfwert (CRC) einen Fehler
W#16#6551	Ein Baustein hat keinen Prüfwert (CRC)
W#16#6353	Firmware-Update: Beginn des Firmwaredownload über das Netz
W#16#6253	Firmware-Update: Ende des Firmwaredownload über das Netz
W#16#6560	SCAN-Overflow
W#16#6981	Schnittstellenfehler kommend
W#16#6881	Schnittstellenfehler gehend
W#16#6390	Formatieren einer Micro Memory Card durchgeführt

Ereignisklasse 8 - Diagnoseereignisse für Baugruppen

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis	Baugruppentyp
W#16#8x00	Baugruppe gestört/ok	Alle
W#16#8x01	Fehler intern	
W#16#8x02	Fehler extern	
W#16#8x03	Kanalfehler vorhanden	
W#16#8x04	Externe Hilfsspannung fehlt	
W#16#8x05	Frontstecker fehlt	
W#16#8x06	Parametrierung fehlt	
W#16#8x07	Falsche Parameter in Baugruppe	
W#16#8x30	Anwendermodul falsch/fehlt	
W#16#8x31	Kommunikationsstörung	
W#16#8x32	Betriebszustand RUN/STOP (STOP: kommend, RUN: gehend)	
W#16#8x33	Zeitüberwachung angesprochen (watch dog)	
W#16#8x34	Baugruppeninterne Versorgungsspannung ausgefallen	
W#16#8x35	BATTF Batterie leer	
W#16#8x36	Gesamte Pufferung ausgefallen	
W#16#8x40	Erweiterungsgeräteausfall	
W#16#8x41	Prozessorausfall	
W#16#8x42	EPROM-Fehler	
W#16#8x43	RAM-Fehler	
W#16#8x44	ADU/DAU-Fehler	
W#16#8x45	Sicherungsfall	

Ereignis-ID	Ereignis	Baugruppentyp
W#16#8x46	Prozessalarm verloren	
W#16#8x50	Projektierungs-/Parametrierfehler	Analogeingabe
W#16#8x51	Gleichtaktfehler (Common-Mode-Fehler)	
W#16#8x52	P-Kurzschluss	
W#16#8x53	M-Kurzschluss	
W#16#8x54	Drahtbruch	
W#16#8x55	Referenzkanal-Fehler	
W#16#8x56	Messbereichsunterschreitung	
W#16#8x57	Messbereichsüberschreitung	
W#16#8x60	Projektierungs-/Parametrierfehler	Analogausgabe
W#16#8x61	Gleichtaktfehler (Common-Mode-Fehler)	
W#16#8x62	P-Kurzschluss	
W#16#8x63	M-Kurzschluss	
W#16#8x64	Drahtbruch	
W#16#8x66	Lastspannung fehlt	
W#16#8x70	Projektierungs-/Parametrierfehler	Digitaleingabe
W#16#8x71	Massefehler	
W#16#8x72	P-Kurzschluss (Geber)	
W#16#8x73	M-Kurzschluss	
W#16#8x74	Drahtbruch	
W#16#8x75	Geberversorgung fehlt	
W#16#8x80	Projektierungs-/Parametrierfehler	Digitalausgabe
W#16#8x81	Massefehler	
W#16#8x82	P-Kurzschluss	
W#16#8x83	M-Kurzschluss	
W#16#8x84	Drahtbruch	
W#16#8x85	Sicherungsfall	
W#16#8x86	Lastspannung fehlt	
W#16#8x87	Übertemperatur	
W#16#8xB0	FM-Zähl: Signal A fehlerhaft	FM
W#16#8xB1	FM-Zähl: Signal B fehlerhaft	
W#16#8xB2	FM-Zähl: Signal N fehlerhaft	
W#16#8xB3	FM-Zähl: Fehlerhaften Wert zwischen den Kanälen übergeben	
W#16#8xB4	FM-Zähl: Geberversorgung 5,2 V fehlerhaft	
W#16#8xB5	FM-Zähl: Geberversorgung 24 V fehlerhaft	

Ereignisklasse 9 - Standard-Anwenderereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#9001	Betriebsart Automatik
W#16#9101	Betriebsart Hand
W#16#9x02	AUF/ZU, EIN/AUS
W#16#9x03	Handbefehlsfreigabe
W#16#9x04	Aggregatschutzbefehl (AUF/ZU)
W#16#9x05	Prozessfreigabe
W#16#9x06	Systemschutzbefehl
W#16#9x07	Istwertüberwachung angesprochen
W#16#9x08	Stellgrößenüberwachung angesprochen
W#16#9x09	Regelabweichung größer als zulässig
W#16#9x0A	Endlagenfehler
W#16#9x0B	Laufzeitfehler
W#16#9x0C	Befehlsausführungsfehler (Ablaufsteuerung)
W#16#9x0D	Betriebszustand läuft > AUF
W#16#9x0E	Betriebszustand läuft > ZU
W#16#9x0F	Befehlsblockierung
W#16#9x11	Prozesszustand AUF/EIN
W#16#9x12	Prozesszustand ZU/AUS
W#16#9x13	Prozesszustand Zwischenstellung
W#16#9x14	Prozesszustand Ein über AUTO
W#16#9x15	Prozesszustand Ein über Hand
W#16#9x16	Prozesszustand Ein über Schutzbefehl
W#16#9x17	Prozesszustand AUS über AUTO
W#16#9x18	Prozesszustand AUS über Hand
W#16#9x19	Prozesszustand AUS über Schutzbefehl
W#16#9x21	Funktionsfehler beim Anfahren
W#16#9x22	Funktionsfehler beim Abfahren
W#16#9x31	Wirkglied (DE/WE) Endlage AUF
W#16#9x32	Wirkglied (DE/WE) Endlage Nicht AUF
W#16#9x33	Wirkglied (DE/WE) Endlage ZU
W#16#9x34	Wirkglied (DE/WE) Endlage Nicht ZU
W#16#9x41	Nicht erlaubter Zustand, Toleranzzeit abgelaufen
W#16#9x42	Nicht erlaubter Zustand, Toleranzzeit nicht abgelaufen
W#16#9x43	Verriegelungsfehler, Toleranzzeit = 0
W#16#9x44	Verriegelungsfehler, Toleranzzeit > 0
W#16#9x45	Reaktion nicht erfolgt
W#16#9x46	Endzustand unzulässig verlassen, Toleranzzeit = 0
W#16#9x47	Endzustand unzulässig verlassen, Toleranzzeit > 0

Ereignis-ID	Ereignis
W#16#9x50	Obergrenze Signalbereich OSF
W#16#9x51	Obergrenze Messbereich OMF
W#16#9x52	Untergrenze Signalbereich USF
W#16#9x53	Untergrenze Messbereich UMF
W#16#9x54	Obere Alarmgrenze OOG
W#16#9x55	Obere Warngrenze OG
W#16#9x56	Obere Toleranzgrenze OT
W#16#9x57	Untere Toleranzgrenze UT
W#16#9x58	Untere Warngrenze UG
W#16#9x59	Untere Alarmgrenze UUG
W#16#9x60	GRAPH 7 Schritt kommt/geht
W#16#9x61	GRAPH 7 Verriegelungsfehler
W#16#9x62	GRAPH 7 Ablauffehler
W#16#9x63	GRAPH 7 Fehler zur Kenntnis genommen
W#16#9x64	GRAPH 7 Fehler quittiert
W#16#9x70	Trend positiv überschritten
W#16#9x71	Trend negativ überschritten
W#16#9x72	Reaktion nicht erfolgt
W#16#9x73	Endzustand unzulässig verlassen
W#16#9x80	Grenzwert überschritten, Toleranzzeit = 0
W#16#9x81	Grenzwert überschritten, Toleranzzeit > 0
W#16#9x82	Grenzwert unterschritten, Toleranzzeit = 0
W#16#9x83	Grenzwert unterschritten, Toleranzzeit > 0
W#16#9x84	Gradient überschritten, Toleranzzeit = 0
W#16#9x85	Gradient überschritten, Toleranzzeit > 0
W#16#9x86	Gradient unterschritten, Toleranzzeit = 0
W#16#9x87	Gradient unterschritten, Toleranzzeit > 0
W#16#9190/9090	Anwenderparametrierungsfehler kommend/gehend
W#16#91F0	Überlauf
W#16#91F1	Unterlauf
W#16#91F2	Division durch 0
W#16#91F3	Unzulässige Rechenoperation

Ereignisklasse A und B - Freie Anwenderereignisse

Beschreibung der Ereignis-IDs

Ereignis-ID	Ereignis
W#16#Axyz	frei verfügbare Ereignisse
W#16#Bxyz	frei verfügbare Ereignisse

Reservierte Ereignisklassen

Reserviert

Folgende Ereignisklassen sind für spätere Erweiterungen reserviert:

- C
- D
- E
- F reserviert für Nicht-Zentralbaugruppen (z. B. CP, FM)

Symbolische und numerische Bezeichnung von Anweisungen

Beschreibung

Die Anweisungen aus der Task Card setzen sich aus Funktionen (FCs), Funktionsbausteinen (FBs), Systemfunktionen (SFCs) und Systemfunktionsbausteinen (SFBs) zusammen, die intern durch Nummern bezeichnet werden.

Die folgenden Tabellen zeigen die Zuordnungen der numerischen und symbolischen Bezeichnung.

Funktionsbausteine (FBs)

Numerische Bezeichnung	Symbolische Bezeichnung
FB 2	S_RCV
FB 3	S_SEND
FB 4	S_VSTAT
FB 5	S_VSET
FB 6	S_XON
FB 7	S_RTS
FB 8	S_V24
FB 8 (1)	USEND_S
FB 9 (1)	URCV_S
FB 14 (1)	GET_S
FB 15 (1)	PUT_S
FB 40 (1)	FTP_CMD
FB 46	PG_DIAL
FB 47	AS_DIAL
FB 48	SMS_SEND
FB 49	AS_MAIL
FB 52 (1)	PNIO_RW_REC
FB 53	PE_DS3_Write_ET200S
FB 54 (1)	PNIO_ALARM
FB 55	IP_CONFIG

Numerische Bezeichnung	Symbolische Bezeichnung
FB 56 (1)	LOGICAL_TRIGGER
FB 60	SET_SW
FB 61	SET_SW_S
FB 62	TIMESTMP
FB 80	LEAD_LAG
FB 81	DCAT
FB 81	S_MODB
FB 82	MCAT
FB 83	IMC
FB 84	SMC
FB 85	DRUM
FB 86	PACK
FB 210	FW_TCP
FB 220	FW_IOT
FB 450 (2)	RED_IN_MGP
FB 451 (2)	RED_OUT_MGP
FB 452 (2)	RED_DIAG_MGP
FB 453 (2)	RED_STATUS_MGP
FB 815	PE_Start_End_Pause
FB 816	PE_CMD
FB 817 (1)	PE_I_DEV
(1) bei S7-300 (2) bei S7-400 (3) nicht bei WinAC (4) bei S7-400, nicht bei WinAC (5) bei WinAC	

Funktionen (FCs)

Numerische Bezeichnung	Symbolische Bezeichnung
FC 0 (1)	PE_Error_RSP
FC 1 (1)	PE_Start_RSP
FC 1 (1)	DP_SEND
FC 2	CONCAT
FC 2 (1)	PE_End_RSP
FC 2 (1)	DP_RECV
FC 3 (1)	PE_List_Modes_RSP
FC 3 (1)	DP_DIAG
FC 4	DELETE
FC 4 (1)	PE_Get_Mode_RSP
FC 4 (1)	DP_CTRL
FC 5 (1)	PE_PEM_Status_RSP

Numerische Bezeichnung	Symbolische Bezeichnung
FC 5 (1)	AG_SEND
FC 6 (1)	PE_Identify_RSP
FC 6 (1)	AG_RECV
FC 7	Asi_3422
FC 7 (1)	PE_Measurement_List_RSP
FC 7 (1)	AG_LOCK
FC 8 (1)	PE_Measurement_Value_RSP
FC 8 (1)	AG_UNLOCK
FC 10 (1)	AG_CNTRL
FC 11	FIND
FC 11 (1)	PNIO_SEND
FC 12 (1)	PNIO_RECV
FC 17	INSERT
FC 17	S_USST
FC 18	S_USSR
FC 19	S_USSI
FC 20	LEFT
FC 21	LEN
FC 22	LIMIT
FC 25	MAX
FC 26	MID
FC 27	MIN
FC 31	REPLACE
FC 32	RIGHT
FC 36	ENCO
FC 36	SEL
FC 37	DECO
FC 60	LOC_TIME
FC 61	BT_LT
FC 62	LT_BT
FC 62 (1)	C_CNTRL
FC 63	S_LTINT
FC 74 (3)	I_ABORT
FC 80	TONR
FC 82	RSET
FC 83	SET
FC 84	ATT
FC 85	FIFO
FC 86	TBL_FIND
FC 87	LIFO
FC 88	TBL
FC 89	TBL_WRD
FC 90	WSR

Numerische Bezeichnung	Symbolische Bezeichnung
FC 91	WRD_TBL
FC 92	SHRB
FC 93	SEG
FC 98	BCDPL
FC 99	BITSUM
FC 100	RSETI
FC 101	SETI
FC 102	DEV
FC 103	CDT
FC 104	TBL_TBL
FC 105	SCALE
FC 106	UNSCALE
FC 450 (2)	RED_INIT_MGP
FC 451 (2)	RED_DEPA_MGP
(1) bei S7-300 (2) bei S7-400 (3) nicht bei WinAC (4) bei S7-400, nicht bei WinAC (5) bei WinAC	

Systemfunktionsbausteine (SFBs)

Numerische Bezeichnung	Symbolische Bezeichnung
SFB 0	CTU
SFB 1	CTD
SFB 2	CTUD
SFB 3	TP
SFB 4	TON
SFB 5	TOF
SFB 16 (4)	PRINT
SFB 19 (2)	START
SFB 20 (2)	STOP
SFB 21 (4)	RESUME
SFB 22 (2)	STATUS
SFB 23 (2)	USTATUS
SFB 29 (1)	HS_COUNT_300C
SFB 30 (1)	FREQ_MES_300C
SFB 31 (2)	NOTIFY_8P
SFB 32	DRUM_X
SFB 33 (2)	ALARM
SFB 34 (2)	ALARM_8
SFB 35 (2)	ALARM_8P

Numerische Bezeichnung	Symbolische Bezeichnung
SFB 36 (2)	NOTIFY_8P
SFB 37 (2)	AR_SEND
SFB 38 (1)	HSC_A_B_300C
SFB 39 (1)	POS_300C
SFB 44 (1)	ANALOG_300C
SFB 46 (1)	DIGITAL_300C
SFB 47 (1)	COUNT_300C
SFB 48 (1)	FREQUENC_300C
SFB 49 (1)	PULSE_300C
SFB 52	RDREC
SFB 53	WRREC
SFB 54	RALRM
SFB 60 (1)	SEND_PTP_300C
SFB 61 (1)	RCV_PTP_300C
SFB 62 (1)	RES_RCVB_300C
SFB 63 (1)	SEND_RK_300C
SFB 64 (1)	FETCH_RK_300C
SFB 65 (1)	SERVE_RK_300C
SFB 73	RCVREC
SFB 74	PRVREC
SFB 75	SALRM
SFB 81	RD_DPAR
SFB 104	IP_CONF
SFB 65001 (5)	CREA_COM
SFB 65002 (5)	EXEC_COM
SFB 65002 (5)	ASYN_COM
(1) bei S7-300 (2) bei S7-400 (3) nicht bei WinAC (4) bei S7-400, nicht bei WinAC (5) bei WinAC	

Systemfunktionen (SFCs)

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 2	SET_RTM
SFC 3	CTRL_RTM
SFC 4	READ_RTM
SFC 5	GADR_LGC
SFC 7 (3)	DP_PRAL
SFC 9 (2)	EN_MSG
SFC 10 (2)	DIS_MSG

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 11	DPSYC_FR
SFC 12	D_ACT_DP
SFC 13	DPNRM_DG
SFC 14	DPRD_DAT
SFC 17	ALARM_SQ
SFC 18	ALARM_S
SFC 19	ALARM_SC
SFC 20	BLKMOV
SFC 21	FILL
SFC 22	CREAT_DB
SFC 23	DEL_DB
SFC 24	TEST_DB
SFC 26 (2)	UPDAT_PI
SFC 27 (2)	UPDAT_PO
SFC 28	SET_TINT
SFC 29	CAN_TINT
SFC 30	ACT_TINT
SFC 31	QRY_TINT
SFC 32	SRT_DINT
SFC 33	CAN_DINT
SFC 34	QRY_DINT
SFC 35 (4)	MP_ALM
SFC 36	MSK_FLT
SFC 37	DMSK_FLT
SFC 38	READ_ERR
SFC 39	DIS_IRT
SFC 40	EN_IRT
SFC 41	DIS_AIRT
SFC 42	EN_AIRT
SFC 43	RE_TRIGR
SFC 44	REPL_VAL
SFC 46	STP
SFC 47	WAIT
SFC 48 (4)	SNC_RTCB
SFC 49	LGC_GADR
SFC 50	RD_LGADR
SFC 51	RDSYSST
SFC 52	WR_USMSG
SFC 54 (2)	RD_DPARM
SFC 55	WR_PARM
SFC 56	WR_DPARM
SFC 57	PARM_MOD
SFC 58	WR_REC

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 59	RD_REC
SFC 60 (4)	GD_SND
SFC 61 (4)	GD_RCV
SFC 62 (2)	CONTROL
SFC 64	TIME_TCK
SFC 64	RD_SINFO
SFC 65 (3)	X_SEND
SFC 66 (3)	X_RCV
SFC 67 (3)	X_GET
SFC 68 (3)	X_PUT
SFC 69 (3)	X_ABORT
SFC 70	GEO_LOG
SFC 71	LOG_GEO
SFC 72 (3)	I_GET
SFC 73 (3)	I_PUT
SFC 75	SET_ADDR
SFC 78 (2)	OB_RT
SFC 79	SETP
SFC 80	RESETP
SFC 81	UBLKMOV
SFC 82 (1) (6)	CREA_DBL
SFC 83 (1) (6)	READ_DBL
SFC 84 (1) (6)	WRIT_DBL
SFC 85 (6)	CREA_DB
SFC 87 (2)	C_DIAG
SFC 90 (4)	H_CTRL
SFC 99	WWW
SFC 100 (2)	SET_CLKS
SFC 101	RTM
SFC 102 (1)	RD_DPARA
SFC 103 (3)	DP_TOPOL
SFC 104 (2)	CIR
SFC 105	READ_SI
SFC 106	DEL_SI
SFC 107	ALARM_DQ
SFC 108	ALARM_D
SFC 109	PROTECT
SFC 112	PN_IN
SFC 113	PN_OUT
SFC 114	PN_DP
SFC 126	SYNC_PI
SFC 127	SYNC_PO

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 154	DPWR_DAT
(1) bei S7-300 (2) bei S7-400 (3) nicht bei WinAC (4) bei S7-400, nicht bei WinAC (5) bei WinAC	

1.4.1.2 Arbeitsweise von S7-1500 CPUs

Betriebszustände

Grundlagen zu den Betriebszuständen von S7-CPUs

Einführung

Betriebszustände beschreiben den Zustand der CPU. Folgende Betriebszustände sind möglich:

- ANLAUF
- RUN
- STOP
- HALT

In diesen Betriebszuständen ist die CPU kommunikationsfähig, z. B. über die PN/IE-Schnittstelle.

Weitere Betriebszustände

Wenn die CPU nicht betriebsbereit ist, befindet sie sich in einem der beiden folgenden Betriebszustände:

- Spannungslos, d. h. die Netzspannung ist ausgeschaltet.
- Defekt, d. h. es ist ein interner Fehler aufgetreten.
  Wenn der Zustand "Defekt" durch einen Firmware-Fehler hervorgerufen wurde, wird dieser Zustand durch die Status-LEDs der CPU signalisiert (siehe Beschreibung zur CPU in den Gerätehandbüchern). Gehen Sie folgendermaßen vor, um die Ursache zu ermitteln:
  - Schalten Sie den Netzschalter aus und wieder ein.
  - Lesen Sie den Diagnosepuffer aus, wenn die CPU anläuft und senden die Daten zur Analyse an den Customer Support.

  Wenn die CPU nicht anläuft, tauschen Sie sie aus.

Betriebszustandsübergänge

Übersicht

Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge von S7-1500-CPUs:

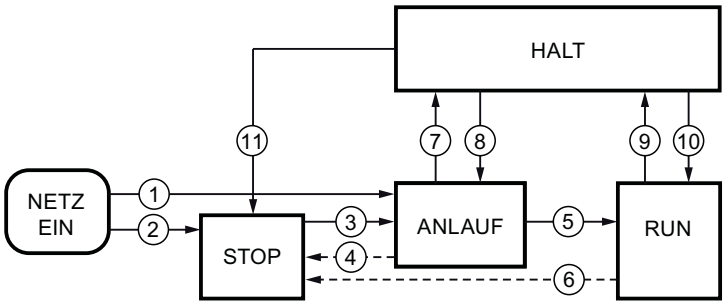

Die folgende Tabelle zeigt die Bedingungen, unter welchen die Betriebszustände wechseln:

Nr.	Betriebszustandsübergang	Bedingungen
①	NETZ EIN → ANLAUF	Die CPU geht nach dem Einschalten in den Betriebszustand "ANLAUF", wenn: • die Anlaufart "Warmstart" eingestellt ist und • die Hardware-Konfiguration und die Programmbausteine konsistent sind. Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten.
②	NETZ EIN → STOP	Bei Einstellung der Anlaufart "Kein Anlauf" geht die CPU nach dem Einschalten der Versorgungsspannung in den Betriebszustand "STOP". Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten.
③	STOP → ANLAUF	Die CPU geht in den Betriebszustand "ANLAUF", wenn: • die CPU vom Programmiergerät aus auf "RUN" gesetzt wird und • die Hardware-Konfiguration und die Programmbausteine konsistent sind.
④	ANLAUF → STOP	Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": • Während des Anlaufs wird ein Fehler erkannt. • Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anlauf-OB bearbeitet.
⑤	ANLAUF → RUN	Wenn der Anlauf erfolgreich ist, wechselt die CPU in "RUN".
⑥	RUN → STOP	Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": • Ein Fehler wird erkannt, der die Weiterarbeit verhindert. • Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anwenderprogramm bearbeitet.
⑦	ANLAUF → HALT	Haltepunkt im Anlaufprogramm erreicht.
⑧	HALT → ANLAUF	Haltepunkt im Anlaufprogramm verlassen, durch vom Programmiergerät aus auf "RUN" gesetzt.
⑨	RUN → HALT	Haltepunkt erreicht.

Nr.	Betriebszustandsübergang	Bedingungen
⑩	HALT → RUN	Haltepunkt verlassen, durch vom Programmiergerät aus auf "RUN" gesetzt.
⑪	HALT → STOP	Bedienung Schalter/Display oder vom Programmiergerät auf "STOP" gesetzt.

Siehe auch

Betriebszustand "RUN" (Seite 957)

Betriebszustand "STOP" (Seite 957)

Betriebszustand "HALT" (Seite 958)

Betriebszustand "ANLAUF"

Grundlagen zum Betriebszustand "ANLAUF"

Funktion

Bevor die CPU nach dem Einschalten mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, wird ein Anlaufprogramm bearbeitet.

Im Anlaufprogramm können Sie durch entsprechende Programmierung von Anlauf-OBs bestimmte Initialisierungs-Variablen für Ihr zyklisches Programm festlegen. Die Anzahl von Anlauf-OBs ist nicht vorgeschrieben. Sie haben die Möglichkeit, keinen Anlauf-OB, einen Anlauf-OB oder mehrere Anlauf-OBs zu programmieren.

Parametrieren des Anlaufverhaltens

Sie können festlegen, ob die CPU im Betriebszustand "STOP" verbleiben soll, oder ob ein Warmstart durchgeführt wird. Darüber hinaus können Sie das Verhalten im Anlauf ("RUN" oder vorheriger Betriebszustand) in der Gruppe "Anlauf" der CPU-Eigenschaften parametrieren.

Besonderheiten

Beachten Sie für den Betriebszustand "ANLAUF" folgende Punkte:

- Die Ausgänge auf den Modulen sind gesperrt.
- Das Prozessabbild ist initialisiert.
- Das Prozessabbild wird nicht aktualisiert.
  - Um den aktuellen Zustand von Eingängen im "ANLAUF" zu lesen, können Sie über direkten Peripheriezugriff auf Eingänge zugreifen.
  - Um Ausgänge im ANLAUF zu initialisieren, können Werte über das Prozessabbild oder über direkten Peripheriezugriff geschrieben werden. Die Werte werden beim Übergang in den Betriebszustand "RUN" an die Ausgänge ausgegeben.
- Die nicht remanenten Merker, Zeiten und Zähler sind initialisiert.
- Die nicht remanenten Variablen in Datenbausteinen sind initialisiert.

- Im Anlauf läuft noch keine Zykluszeitüberwachung.
- Die Anlauf-OBs werden abgearbeitet. Unabhängig von der gewählten Anlaufart werden alle programmierten Anlauf-OBs bearbeitet.
  Folgende OBs können im Anlauf gestartet werden, falls ein entsprechendes Ereignis auftritt:
  - OB 82: Diagnosealarm
  - OB 83: Ziehen/Stecken von Modulen
  - OB 86: Baugruppenträgerfehler
  - OB 121: Programmierfehler (nur bei globaler Fehlerbehandlung)
  - OB 122: Peripheriezugriffsfehler (nur bei globaler Fehlerbehandlung)

  Alle anderen OBs können erst beim Übergang in den Betriebszustand "RUN" gestartet werden.

Neustart (Warmstart)

Funktion

Beim Neustart (Warmstart) werden alle nicht remanenten Merker gelöscht und nicht remanente DB-Inhalte auf Startwerte aus dem Ladespeicher zurückgesetzt. Remanente Merker und remanente DB-Inhalte bleiben erhalten.

Die Programmbearbeitung beginnt mit dem ersten Anlauf-OB.

Auslösen von Neustart (Warmstart)

In folgenden Fällen können Sie über einen Menübefehl vom PG aus einen "Neustart (Warmstart)" durchführen:

- Die CPU ist im Betriebszustand "STOP".
- Nach dem Urlöschen
- Nach dem Laden eines konsistenten Programms und einer konsistenten Hardware-Konfiguration im Betriebszustand "STOP" der CPU.

Bei "NETZ-EIN" wird ein "Neustart (Warmstart)" ausgelöst, wenn Sie das Anlaufverhalten folgendermaßen parametriert haben:

- Anlaufart "Warmstart - RUN" (unabhängig vom Betriebszustand der CPU vor NETZ-AUS).
- "Warmstart - Betriebsart vor NETZ-AUS" (abhängig vom Betriebszustand der CPU vor NETZ-AUS. Die CPU muss vorher im RUN gewesen sein.)

Anlauftätigkeiten

Übersicht

Die folgende Tabelle zeigt, welche Tätigkeiten die CPU beim Anlauf durchführt:

Tätigkeiten in der Bearbeitungsreihenfolge	bei Neustart (Warmstart)
Nicht remanente Merker löschen	ja
Alle Merker löschen	nein
Prozessabbild der Ausgänge löschen	ja
Bearbeitung der Anlauf-OBs	ja
Prozessabbild der Eingänge aktualisieren	ja
Ausgänge freigeben nach Betriebszustandsübergang in "RUN"	ja

Ablauf

Das folgende Bild zeigt die Tätigkeiten der CPU in den Betriebszuständen "STOP", "ANLAUF" und "RUN".

Den Zustand der Peripherieausgänge im ersten Zyklus des Anwenderprogramms können Sie durch folgende Maßnahmen festlegen:

- Parametrierbare Ausgabebaugruppen verwenden, um Ersatzwerte ausgeben zu können oder um den letzten Wert zu halten.
- Ausgänge in Anlauf-OBs vorbesetzen.

Während des Anlaufs werden alle Alarm-Ereignisse in eine Warteschlange gestellt, um sie dann im Betriebszustand RUN abzuarbeiten. Im Betriebszustand RUN können zu jeder Zeit Prozess-Alarme verarbeitet werden.

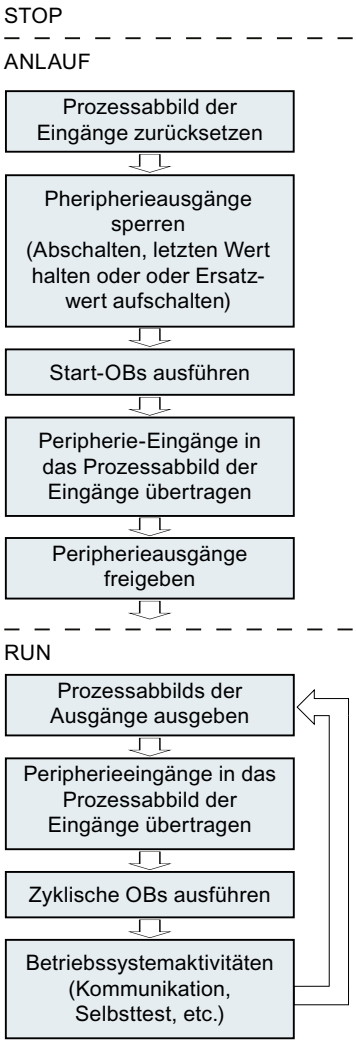

Besonderheiten beim Anlauf

Verhalten bei Sollausbau ungleich Istausbau

Der Sollausbau wird repräsentiert durch die in die CPU geladene projektierte Konfiguration. Der Istausbau ist der tatsächliche Ausbau des Automatisierungssystems.

Wenn Sollausbau und Istausbau voneinander abweichen, wird das Verhalten der CPU durch den Parameter "Unterstützte Hardwarekompatibilität" bestimmt.

Abbrechen eines Anlaufs

Wenn während des Anlaufs Fehler auftreten, wird der Anlauf abgebrochen und die CPU fällt zurück in "STOP".

Ein Anlauf wird unter folgenden Bedingungen nicht durchgeführt oder abgebrochen:

- Wenn keine oder eine ungültige SIMATIC Memory Card gesteckt ist.
- Wenn keine Hardware-Konfiguration geladen wurde.

Betriebszustand "RUN"

Funktion

Im Betriebszustand "RUN" erfolgt die zyklische, zeit- und alarmgesteuerte Programmbearbeitung. Adressen, die im Prozessabbild "Automatische Aktualisierung" liegen, werden in jedem Programmzyklus automatisch aktualisiert:

- Das Prozessabbild der Ausgänge wird ausgegeben.
- Das Prozessabbild der Eingänge wird gelesen.
- Das Anwenderprogramm wird abgearbeitet.

Der aktive Austausch von Daten zwischen S7-1200 CPUs über Open User Communication ist nur im Betriebszustand "RUN" möglich.

Abarbeiten des Anwenderprogramms

Nachdem die CPU die Eingänge gelesen hat, wird das zyklische Programm beginnend mit der ersten Anweisung bis zur letzten Anweisung abgearbeitet.

Wenn Sie eine Mindestzykluszeit parametriert haben, dann beendet die CPU den Zyklus erst nach Ablauf der Mindestzykluszeit, auch wenn das Anwenderprogramm in kürzerer Zeit durchlaufen wurde.

Um sicherzustellen, dass das zyklische Programm in einer festgelegten Zeit durchlaufen wird, ist eine Zyklusüberwachungszeit eingestellt, die Sie nach Ihren Anforderungen anpassen können. Wenn das zyklische Programm nicht innerhalb dieser Zeit durchlaufen ist, reagiert das System mit einem Zeitfehler.

Weitere Ereignisse wie z. B. Prozessalarme oder Diagnosealarme können den zyklischen Programmfluss unterbrechen und die Zykluszeit verlängern.

Siehe auch

Betriebszustandsübergänge (Seite 952)

Betriebszustand "STOP"

Funktion

Im Betriebszustand "STOP" wird das Anwenderprogramm nicht bearbeitet. Alle Ausgänge werden deaktiviert bzw. reagieren wie parametriert: sie liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.

Die CPU prüft folgende Punkte:

- Ob für die CPU die Standardeinstellungen gelten oder Parametersätze vorliegen
- Ob die Randbedingungen für das programmierte Anlaufverhalten stimmen

Siehe auch

Betriebszustandsübergänge (Seite 952)

Betriebszustand "HALT"

Funktion

Der Betriebszustand "HALT" wird erreicht durch einen Haltepunkt (Breakpoint) in einem Baustein, den Sie im Einzelschrittmodus testen. Die Testfunktion Einzelschrittmodus steht Ihnen in den Programmiersprachen AWL und SCL zur Verfügung.

Im Betriebszustand "HALT" werden keine Ereignisse ausgelöst, das Anwenderprogramm wird nicht ausgeführt.

Alle Ausgänge sind deaktiviert bzw. reagieren wie parametriert: Ausgänge liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.

Zeiten bzw. Überwachungszeiten werden angehalten. Die Zeiten für zeitgesteuerte Ereignisse werden gestoppt.

Prozessalarm-OBs werden im HALT nicht durch externe Alarmauslöser (z. B. Prozessalarm-Eingang) gestartet.

Die Systemzeit läuft weiter.

Bei NETZ-AUS > NETZ-EIN Übergang während des HALT-Zustands geht die CPU in den Betriebszustand STOP und bleibt in diesem Zustand. Die CPU führt in diesem Fall keinen automatischen Anlauf aus.

Wenn das Anwenderprogramm einen Haltepunkt erreicht, wird die Zyklusüberwachungszeit gestoppt. Wenn der Haltepunkt wieder verlassen wird, dann startet die Überwachungszeit erneut.

Fehler, die im Betriebszustand RUN zu einem STOP der CPU führen, führen auch im Betriebszustand HALT zum STOP.

Siehe auch

Betriebszustandsübergänge (Seite 952)

Grundlagen zum Urlöschen

Funktion

Das Urlöschen der CPU ist nur im Betriebszustand "STOP" möglich.

Beim Urlöschen wird die CPU in einen so genannten "Anfangszustand" versetzt. Das bedeutet:

- Eine bestehende Online-Verbindung zwischen PG/PC und der CPU wird abgebaut.
- Der Inhalt des Arbeitsspeichers sowie remanente und nichtremanente Daten werden gelöscht..
- Der Diagnosepuffer, die Uhrzeit und die IP-Adresse bleiben erhalten.
- Anschließend wird die CPU mit den geladenen Projektdaten initialisiert (Hardware-Konfiguration, Code- und Datenbausteine, Forceaufträge). Diese Daten werden vom Ladespeicher in den Arbeitsspeicher kopiert.
  Ergebnis:
  - Wenn in der Hardware-Konfiguration eine IP-Adresse eingestellt wurde (Option "IP-Adresse im Projekt einstellen"), dann ist diese IP-Adresse nach dem Urlöschen gültig.
  - Datenbausteine haben keine Aktualwerte mehr, sondern ihre projektierten Startwerte.
  - Forceaufträge bleiben aktiv

Speicherbereiche

Wissenswertes zu SIMATIC Memory Cards

Funktion der SIMATIC Memory Card

Die SIMATIC Memory Card für eine S7-1500 ist eine mit dem Windows Betriebssystem kompatible und von Siemens vorformatierte SD-Speicherkarte für das Anwenderprogramm der CPU.

ACHTUNG
Formatieren der SIMATIC Memory Card Sie dürfen nur Dateien und Ordner löschen. Wenn Sie die SIMATIC Memory Card mit Windows-Mitteln formatieren, z. B. mit einem handelsüblichen Kartenleser, machen Sie die SIMATIC Memory Card als Speichermedium für eine S7-CPU unbrauchbar.

Einstellen des Kartentyps

Sie können die SIMATIC Memory Card als Programmkarte oder als Firmware-Update-Karte verwenden.

Zur Einstellung des Kartentyps stecken Sie die SIMATIC Memory Card in den Kartenleser des Programmiergeräts und wählen den Ordner "Card Reader/USB-Speicher" in der Projektnavigation. In den Eigenschaften der markierten Memory Card bestimmen Sie den Kartentyp:

- Programm
  Eine Programmkarte nutzen Sie als externen Ladespeicher für die CPU. Sie enthält das vollständige Anwenderprogramm für die CPU. Das Anwenderprogramm wird in den Arbeitsspeicher übertragen und läuft dort ab. Wird die SIMATIC Memory Card mit dem Anwenderprogramm entfernt, ist kein Programm mehr vorhanden.
  Verzeichnis: SIMATIC.S7S
- Firmware Card
  Auf einer SIMATIC Memory Card kann Firmware für S7-1500 Baugruppen gespeichert werden. Daher ist es möglich mit Hilfe einer speziell vorbereiteten SIMATIC Memory Card ein Firmware Update durchzuführen.
  Verzeichnis: FWUPDATE.S7S

Ordner und Dateien auf der SIMATIC Memory Card

Auf der SIMATIC Memory Card können sich folgende Ordner und Dateien befinden:

Ordner	Beschreibung
FWUPDATE.S7S	Firmware-Update-Dateien für CPU und Peripheriemodule
SIMATIC.S7S	Anwenderprogramm, d. h. alle Bausteine (OBs, FCs, FBs, DBs) und Systembausteine, Projektdaten der CPU
SIMATIC.HMI	HMI relevante Daten
DataLogs	DataLog-Dateien
Recipes	Rezeptdateien
UserFiles	Nutzerdaten
Backups	Dateien für sichern und wiederherstellen über das Display

Der Inhalt der Ordner "DataLogs", "Recipes" und "UserFiles" wird beim Laden des Geräts als neue Station ebenfalls in das STEP 7-Projekt (ab TIA Portal V15.1) geladen. Sie finden die Dateien im Projektpfad unter "AdditionalFiles/MemoryCards/<Name of the PLC>". Nähere Informationen zu DataLogs und Rezeptdateien finden Sie im Kapitel "Rezepturen und Data Logging".

Folgende Dateistruktur finden Sie auf der SIMATIC Memory Card vor:

Dateityp	Beschreibung
S7_JOB.S7S	Auftragsdatei
SIMATIC.HMI\Backup*.psb	Panel-Backup-Dateien
SIMATICHMI_Back-ups_DMS.bin	Geschützte Datei (notwendig für die Nutzung von Panel-Backup-Dateien in STEP 7)
__LOG__	Geschützte Systemdatei (notwendig für die Nutzung der Karte)
crdinfo.bin	Geschützte Systemdatei (notwendig für die Nutzung der Karte)

Dateityp	Beschreibung
DUMP.S7S	Servicedaten-Datei
*.pdf, *.txt, *.csv, ...	Weitere Datei mit unterschiedlichen Formaten, die Sie auch in Ordnern der SIMATIC Memory Card speichern können. Nur wenn Sie diese Dateien in einem Ordner namens "UserFiles" gespeichert haben, werden sie beim Laden des Geräts als neue Station mit im STEP 7 Projekt abgelegt.

Objekte aus dem Projekt auf SIMATIC Memory Card übertragen

Wenn die SIMATIC Memory Card im PG bzw. im externen Kartenleser steckt, können Sie folgende Objekte aus der Projektnavigation auf die SIMATIC Memory Card übertragen:

- Einzelne Bausteine (Mehrfachselektion möglich)
  In diesem Fall wird ein konsistentes Übertragen angeboten, d. h. die Abhängigkeiten der Bausteine untereinander durch Bausteinaufrufe wird berücksichtigt.
- PLC
  In diesem Fall werden alle ablaufrelevanten Objekte, u. a. Bausteine und die Hardware-Konfiguration auf die SIMATIC Memory Card übertragen - wie beim Laden.

Um die Übertragung durchzuführen, können Sie die Objekte per Drag&Drop übertragen oder Sie verwenden den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben" im Menü "Projekt".

Nutzerdaten auf SIMATIC Memory Card speichern

Nutzerdaten, z. B. PDF-Dateien zur Projektdokumentation, übertragen Sie auf die SIMATIC Memory Card, indem Sie manuell einen Ordner mit dem Namen "UserFiles" auf der Memory Card anlegen und Ihre Dateien in diesem Ordner speichern. Nur Dateien in diesem Ordner (*.pdf, *.txt, *.csv, ...) werden beim Laden des Geräts als neue Station ebenfalls in das STEP 7-Projekt geladen. Der Ordner "UserFiles" kann z. B. zur Wiederherstellung bei einem Defekt der Memory Card oder der CPU genutzt werden.

Nach dem Laden von der Memory Card wird der Ordner "UserFiles" mit den darin enthaltenen Dateien im Projektbaum im Verzeichnis "AdditionalFiles > MemoryCards > [Name der CPU]" gespeichert.

Wenn der Ordner "[Name der CPU]" bereits existiert, z. B. bei erneutem Laden von der Memory Card, wird der Ordner im Projektbaum mit den Daten von der Memory Card überschrieben.

Objekte von der SIMATIC Memory Card in das Projekt übertragen

Objekte übertragen Sie, indem Sie das SIMATIC Memory Card-Symbol in der Projektnavigation markieren und per Drag&Drop auf eine CPU im Projekt ziehen. Der Inhalt der SIMATIC Memory Card muss dem CPU-Typ in der Projektnavigation entsprechen. Wenn bereits Bausteine vorhanden waren, werden die vor dem Übertragen gelöscht.

Firmware aktualisieren mit einer SIMATIC Memory Card

Die neuesten Firmware-Dateien erhalten Sie aus dem Internet über die Seiten des Service & Support:

http://support.automation.siemens.com

Speichern Sie die Firmware-Dateien auf der Festplatte und stecken Sie die SIMATIC Memory Card in den Card Reader Ihres Programmiergeräts.

Um die Datei auf der SIMATIC Memory Card zu speichern, markieren Sie in der Projektnavigation die Memory Card im Ordner "Card Reader/USB-Speicher". Wählen Sie das Kontextmenü "Card Reader/USB-Speicher > Firmware-Update Memory Card erstellen".

Folgen Sie anschließend den Anweisungen im Service & Support-Portal bzw. im Systemhandbuch S7-1500 zur Durchführung des Firmware-Updates.

Durch die Aktualisierung der Firmware erhält das Modul, z. B. CPU oder Peripheriemodul, einen neuen Firmwarestand. Wenn Sie das Modul im Projekt verwendet haben, müssen Sie durch einen Modultausch das bereits projektierte Modul durch das Modul mit dem neuen Firmwarestand offline nachziehen und das Programm bzw. die Konfiguration anpassen und anschließend laden.

Manipulation von SIMATIC Memory Cards diagnostizieren

Wenn eine CPU zeitweise nicht in Betrieb gewesen ist, kann möglicherweise eine Manipulation an der SIMATIC Memory Card nicht ausgeschlossen werden. Das Programm funktioniert möglicherweise nicht wie vor der Außerbetriebnahme.

S7-1500 CPUs ab FW 2.0 sind in der Lage, folgende Manipulationen zu diagnostizieren und tragen diese Ereignisse im Anlauf als Security-Ereignis in den Diagnosepuffer ein:

- Das Projekt auf der SIMATIC Memory Card hat sich geändert (die SIMATIC Memory Card ist dieselbe geblieben)
- Die SIMATIC Memory Card wurde gegen eine andere Memory Card ausgetauscht

Lebensdauer von SIMATIC Memory Cards

Die Lebensdauer von SIMATIC Memory Cards ist begrenzt und u. A. von folgenden Faktoren abhängig:

- Kapazität der SIMATIC Memory Card
- Schreibhäufigkeit
- Datenmenge, die auf die SIMATIC Memory Card geschrieben wird.

Schreib- bzw. Löschvorgänge, insbesondere zyklische Schreib-/Löschvorgänge über das Anwenderprogramm auf die SIMATIC Memory Card verkürzen deren Lebensdauer.

Die zyklische Ausführung der Anweisungen "CREATE_DB" (mit ATTRIB "DB im Ladespeicher erstellen"), "DataLogWrite", "RecipeExport", "RecipeImport" und das Setzen der Zeitzone "SET_TIMEZONE" reduzieren abhängig von der Anzahl der Schreibvorgänge und Daten die Lebensdauer der Speicherkarte.

Neben den zyklischen Schreib-/Löschvorgängen wirkt sich auch das Schreiben bzw. Löschen sehr großer Datenmengen negativ auf die Lebensdauer der SIMATIC Memory Card aus.

Eine Beschreibung, wie Sie die Lebensdauer von SIMATIC Memory Cards abschätzen, finden Sie für die S7-1500 CPUs, ET 200SP CPUs und ET 200Pro CPU unter folgendem Beitrag:

SIMATIC S7-1500 Struktur und Verwendung des CPU-Speichers (https://support.industry.siemens.com/cs/ww/de/view/59193101)

Für die CPUs der S7-1200 finden Sie die Information unter folgendem Beitrag:

SIMATIC S7 S7-1200 Automatisierungssystem (https://support.industry.siemens.com/cs/ww/de/view/109478121)

Sie haben die Möglichkeit, in den CPU-Eigenschaften einen Lebensdauer-Anteil zu parametrieren, so dass die CPU nach Ablauf des parametrierten Lebensdauer-Anteils einen Diagnosealarm und einen Diagnosepuffereintrag (Maintenance-Ereignis) erzeugt. So bleibt ausreichend Zeit, die SIMATIC Memory Card auszutauschen, bevor sie infolge der erreichten maximalen Anzahl von Schreibzugriffen unbrauchbar wird.

Mit der Anweisung "GetSMCinfo" lesen Sie Informationen über die gesteckte SIMATIC Memory Card aus. Sie wählen die auszulesende Information über den Parameter „Mode“ aus, z. B.

- Mode = 2: Bisher verbrauchter Anteil der Lebensdauer (in %)
- Mode = 3: Parametrierter Anteil der Lebensdauer (in %), nach dessen Ablauf die CPU ein Maintenance-Ereignis in den Diagnosepuffer einträgt und die Maintenance-LED einschaltet.

Forceaufträge auf der SIMATIC Memory Card

Beachten Sie, dass ein auf der SIMATIC Memory Card gespeicherter Forceauftrag auf jeder CPU ausgeführt wird, in die diese SIMATIC Memory Card gesteckt wird! Falls dies nicht gewünscht wird, beenden Sie den Forceauftrag bevor Sie die SIMATIC Memory Card in eine andere CPU stecken.

Siehe auch

Wissenswertes zu Sammelmeldungen bei Security-Ereignissen (Seite 1035)

Speicherbereiche der CPU

Speicherbereiche der S7-1500-CPU

Das folgende Bild zeigt die Speicherbereiche der CPU und den Ladespeicher auf der SIMATIC Memory Card. Neben dem Ladespeicher können sich weitere Daten auf der SIMATIC Memory Card befinden, z. B. Rezepturen, Data Logs, HMI-Backups.

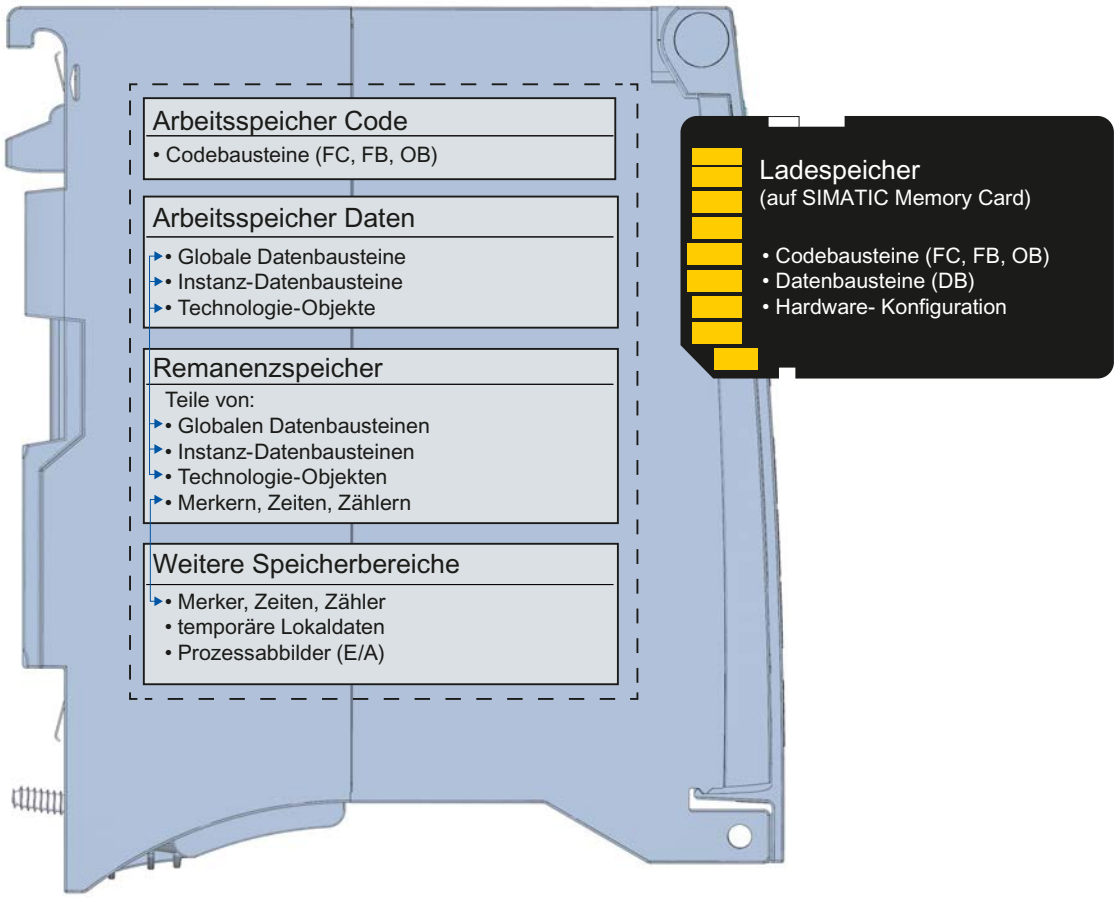

Ladespeicher

Der Ladespeicher ist ein nicht flüchtiger Speicher für Programmcode, Datenbausteine und Hardware-Konfiguration. Beim Laden dieser Objekte in die CPU werden sie zunächst im Ladespeicher abgelegt. Dieser Speicher befindet sich auf der SIMATIC Memory Card. Für den Betrieb der CPU ist eine gesteckte SIMATIC Memory Card zwingend erforderlich.

Arbeitsspeicher

Der Arbeitsspeicher ist ein flüchtiger Speicher, der die Code- und Datenbausteine enthält. Der Arbeitsspeicher ist in der CPU integriert und nicht erweiterbar.

Der Arbeitsspeicher ist bei der S7-1500 in zwei Bereiche aufgeteilt:

- Code-Arbeitsspeicher für ablaufrelevante Teile des Programmcodes.
- Daten-Arbeitsspeicher für ablaufrelevante Teile von Datenbausteinen und Technologieobjekten. Bei den Betriebszustandsübergängen NETZ-EIN → Anlauf und bei STOP → Anlauf werden Variablen von Global-Datenbausteinen, Instanz-Datenbausteinen und Technologie-Objekten mit ihren Startwerten initialisiert. Remanente Variablen erhalten ihre im Remanenzspeicher gesicherten Aktualwerte.

Remanenzspeicher

Der Remanenzspeicher ist ein nicht flüchtiger Speicher zur Sicherung einer begrenzten Menge an Daten bei Spannungsausfall. Im Remanenzspeicher werden die als remanent definierten Variablen und Operandenbereiche gesichert.

Diese Daten bleiben über eine Abschaltung oder einen Spannungsausfall hinweg erhalten. Alle anderen Variablen gehen verloren und werden bei den Betriebszustandsübergängen Anlauf nach NETZ-EIN und bei Anlauf nach STOP auf ihre Startwerte gesetzt.

Der Inhalt des Remanenzspeichers wird durch folgende Aktionen gelöscht:

- Urlöschen
- Rücksetzen auf Werkseinstellungen

Im Remanenzspeicher werden auch bestimmte Variablen von Technologieobjekten gespeichert. Diese werden beim Urlöschen nicht gelöscht.

Weitere Speicherbereiche

Neben den beschriebenen Speicherbereichen für das Anwenderprogramm und Daten verfügt die CPU über weitere Speicherbereiche.

Zu diesen Bereichen gehören die folgenden ablaufrelevanten Bereiche:

- Merker, Zeiten, Zähler
- Temporäre Lokaldaten
- Prozessabbilder

Die jeweiligen CPU-spezifischen Größen finden Sie in den Technischen Daten der CPU.

Informationen zu Speicherbereichen

Sie können sich offline und online Informationen über die Speicherbereiche Ihrer S7-1500 CPU ansehen.

Offline:

Sie können während der Erstellung oder der Änderung eines Programms feststellen, ob es für eine bestimmte CPU zu groß wird. Die Informationen finden Sie z. B. unter "Programminformationen" in der Projektnavigation. Im Register "Speicherauslastung" finden Sie Informationen über die Gesamtgröße der Speicherbereiche und über bereits belegten Speicherplatz. An dieser Stelle finden Sie auch Informationen über die belegten Ein- und Ausgänge.

Bei der S7-1500 können Sie die Gesamtgröße des Ladespeichers festlegen (Programminformationen; das Feld Ladespeicher - Gesamt ist eine Klappliste zur Auswahl der Ladespeichergröße).

Online:

Starten Sie "Online & Diagnose" in der Projektnavigation. Die Funktion "Speicher" gibt Auskunft über den Füllstand aller Speicher der CPU.

Weitere Informationen zur Speicherauslastung Ihrer CPU finden Sie in den Registern "Belegungsplan" und "Aufrufstruktur" der Programminformation sowie in der PLC-Variablentabelle.

Siehe auch

Füllstand aller Speicherarten einer CPU anzeigen (Seite 3496)

Remanente Speicherbereiche

Einleitung

Die S7-1500-CPUs verfügen über einen Remanenzspeicher zur Ablage von remanenten Daten bei NETZ-AUS. Die Größe des Remanenzspeichers ist in den technischen Daten der CPU dokumentiert.

Die Auslastung des Remanenzspeichers der konfigurierten CPU finden Sie offline in der Projektnavigation unter "Programminformationen > Speicherauslastung" bzw. online in der Projektnavigation unter "Online & Diagnose > Diagnose > Speicher".

Wenn Sie Daten als remanent definieren, bleibt deren Inhalt nach Spannungsausfall, beim Anlauf der CPU und beim Laden eines geänderten Programms erhalten.

Folgende Daten bzw. Objekte können Sie als remanent definieren:

- Merker, Zeiten, Zähler
- Variablen von globalen Datenbausteinen
- Variablen von Instanz-Datenbausteinen eines Funktionsbausteins

Bestimmte Variablen von Technologieobjekten sind immer remanent, z. B. Justagewerte von Absolutwertgebern.

Merker, Zeiten, Zähler

Die Anzahl remanenter Merker, Zeiten und Zähler können Sie in der PLC-Variablentabelle über die Schaltfläche "Remanenz" definieren.

Variablen eines globalen Datenbausteins

In einem globalen Datenbaustein können Sie abhängig von der Einstellung des Zugriffs entweder einzelne oder alle Variablen des Bausteins gemeinsam als remanent definieren:

- Optimierter Bausteinzugriff aktiviert: Remanenz ist für jede einzelne Variable einstellbar.
- Optimierter Bausteinzugriff deaktiviert: Remanenzeinstellung gilt für alle Variablen des Datenbausteins; entweder sind alle Variablen remanent oder keine Variable ist remanent.

Variablen eines Instanz-Datenbausteins (Funktionsbaustein)

Die Variablen des Instanz-Datenbausteins eines Funktionsbausteins können Sie als remanent definieren. Sie können abhängig vom Zugriff auf den Instanz-Datenbaustein für einzelne Variablen oder für alle Variablen des Bausteins gemeinsam definieren:

- Optimierter Bausteinzugriff aktiviert: Sie können sowohl in der Schnittstelle des Funktionsbausteins einzelne Variablen als remanent definieren als auch im zugeordneten Instanz-Datenbaustein.
- Optimierter Bausteinzugriff deaktiviert: Sie können im Instanz-Datenbaustein die Remanenz für alle Variablen gemeinsam definieren.

Erzeugen eines Datenbausteins im Anwenderprogramm

Mit der Anweisung "Create_DB" erzeugen Sie einen Datenbaustein. Je nach Wahl des Parameters ATTRIB hat der erzeugte Datenbaustein entweder die Eigenschaft "remanent" oder "nicht remanent". Die Einstellung der Remanenz für einzelne Variablen ist dabei nicht möglich.

Variablen von Technologieobjekten

Bestimmte Variablen von Technologieobjekten sind ebenfalls remanent, z. B. Justagewerte von Absolutwertgebern. STEP 7 verwaltet die Remanenz von Technologieobjekten automatisch, so dass Sie keine Remanenz projektieren müssen. Diese Variablen sind urlöschfest und können nur durch Rücksetzen auf Werkseinstellungen gelöscht werden.

Tipp: Remanenzverlust erkennen im Anlauf-OB und Zyklus-OBs

In folgendem Fall verlieren alle Variablen (Merker, Zeiten, Zähler, DB-Inhalte) eines Anwenderprogramms ihre Remanenz:

1. Sie stecken die SIMATIC Memory Card mit Projekt (Anwenderprogramm) in den Card Reader Ihres Programmiergeräts.
2. Sie ändern im zugehörigen Projekt die Größe von Datenbausteinen oder fügen Datenbausteine mit remanenten Variablen hinzu.
3. Sie übertragen das geänderte Projekt wieder auf die SIMATIC Memory Card im Card Reader.
4. Sie lassen die CPU mit dem geänderten Projekt anlaufen.

Durch die Offline-Änderung verlieren alle Variablen (Merker, Zeiten, Zähler, DB-Inhalte) ihre Remanenz. Die Variablen sind nach dem STOP-RUN-Übergang mit ihren Initialwerten vorbelegt.

Um im Anwenderprogramm auf einen Remanenzverlust zu reagieren, werten Sie folgende lokalen Variablen der Startinformation aus:

- Anlauf-OB: LostRetentive (Wahr bei Remanenzverlust)
- Zyklus-OBs: Remanence (Wahr, wenn remanente Daten verfügbar sind)

Zusammenfassung Remanenzverhalten

Remanenzverhalten der Speicherobjekte

Neben den bisher beschriebenen remanenten Speicherbereichern gibt es noch weitere Objekte mit Remanenzverhalten, z. B. der Diagnosepuffer. Diese Objekte belegen keinen Remanenzspeicher.

Die folgende Tabelle zeigt das Remanenzverhalten der Speicherobjekte

- bei den Betriebszustandsübergängen STOP → Anlauf und NETZ-AUS → Anlauf
- bei den speicherbeeinflussenden Funktionen Urlöschen und Rücksetzen auf Werkseinstellungen

Tabelle 1-83 Remanenzverhalten der Speicherobjekte ("x" = Inhalt bleibt erhalten; "-" = Inhalt wird initialisiert)

Speicherobjekt	Betriebszustandsübergänge		Urlöschen	Rücksetzen auf Werkseinstellungen
	STOP → ANLAUF	NETZ-AUS → ANLAUF		
Aktualwerte der Datenbausteine, Instanz-Datenbausteine	In den Eigenschaften des DB einstellbar.		-	-
Merker, Zeiten und Zähler - als remanent projektiert	x	x	-	-
Merker, Zeiten und Zähler - als nicht remanent projektiert	-	-	-	-
Bestimmte remanente Variablen von Technologieobjekten (z. B. Justagewerte von Absolutwertgebern)	x		x	-
Diagnosepuffer-Einträge (remanenter Bereich)	x	x	x	-
Diagnosepuffer-Einträge (nicht-remanenter Bereich)	x	-	-	-
Betriebsstundenzähler	x	x	x	-
Uhrzeit	x	x	x	-

Diagnosepuffer

Bei den S7-1500 CPUs ist ein Teil des Diagnosepuffers remanent. Die aktuellsten Einträge im Diagnosepuffer bleiben nach Spannungsausfall erhalten und sind urlöschfest. Der remanente Teil des Diagnosepuffers kann nur durch Rücksetzen auf Werkseinstellungen gelöscht werden. Die Anzahl der remanenten Diagnosepuffer-Einträge ist von der CPU abhängig. Die Einträge im Diagnosepuffer benötigen keinen Remanenzspeicher.

Betriebsstundenzähler

Der Betriebsstundenzähler der S7-1500 CPUs ist remanent und urlöschfest. Durch Rücksetzen auf Werkseinstellungen wird er auf Null gesetzt.

Uhrzeit

Die Uhrzeit der S7-1500 CPUs ist remanent und urlöschfest. Durch Rücksetzen auf Werkseinstellungen wird die Uhrzeit auf den 01.01.2012 00:00:00 gesetzt.

Speicherverhalten beim Laden von Softwareänderungen

Einleitung

Die S7-1500 CPU bietet Ihnen die Möglichkeit, im STOP und im RUN der CPU Änderungen der Software zu laden, ohne dass beim Laden die Aktualwerte von schon geladenen Variablen beeinflusst werden.

In STEP 7 laden Sie Änderungen der Software unter "Laden in Gerät > Software (nur Änderungen)".

Auswirkungen von Softwareänderungen bei PLC-Variablen

Sie können folgende Softwareänderungen laden, **ohne** dass die Aktualwerte von bereits geladenen Variablen betroffen sind:

- Änderung von Namen
- Änderung von Kommentaren
- Erweitern um neue Variablen
- Löschen von Variablen
- Änderung der Remanenzeinstellungen für Merker, Zeiten und Zähler.

Beim Laden von folgenden Softwareänderungen sind die Aktualwerte betroffen:

- Änderung von Datentypen
- Änderung von Adressen

Auswirkungen von Softwareänderungen bei Datenbausteinen mit Speicherreserve

Wenn Sie die Speicherreserve für Datenbausteine einsetzen (Attribut "Optimierter Bausteinzugriff" und Schaltfläche "Laden ohne Reinitialisierung" aktiviert), können Sie folgende Softwareänderungen laden, **ohne** dass die Aktualwerte von bereits geladenen Variablen reinitialisiert werden:

- Änderung von Startwerten
- Änderung von Kommentaren
- Erweitern um neue Variablen

Zum Laden von folgenden Softwareänderungen müssen Sie die Schaltfläche "Laden ohne Reinitialisierung" deaktivieren. Beim nächsten Laden werden alle Aktualwerte des Datenbausteins reinitialisiert:

- Änderung von Namen
- Änderung von Datentypen
- Änderung von Remanenzeinstellungen
- Löschen von Variablen
- Änderungen an den Einstellungen der Speicherreserve

Auswirkungen von Softwareänderungen bei Datenbausteinen ohne Speicherreserve

Wenn Sie keine Speicherreserve einsetzen, können Sie folgende Softwareänderungen laden, ohne dass die Aktualwerte von schon geladenen Variablen reinitialisiert werden:

- Änderung von Startwerten
- Änderung von Kommentaren

Beim Laden von folgenden Softwareänderungen werden alle Aktualwerte des Datenbausteins reinitialisiert:

- Änderung von Name
- Änderung von Datentypen
- Änderung von Remanenzeinstellungen
- Erweitern um neue Variablen
- Löschen von Variablen

Speicherreserve von Datenbausteinen

Jeder Funktions- oder Datenbaustein mit aktiviertem Attribut "Optimierter Bausteinzugriff" enthält per Voreinstellung eine Speicherreserve, die Sie für nachträgliche Schnittstellenänderungen verwenden können. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass Sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die neuen Variablen auf Ihre Startwerte initialisiert, die bisher schon geladenen Variablen werden nicht reinitialisiert.

Die Einstellung der Speicherreserve finden Sie in den Datenbaustein-Eigenschaften unter "Laden ohne Reinitialisierung".

Speicherbedarf beim Laden von Softwareänderungen im RUN

Zur konsistenten und atomaren Abwicklung des gesamten Ladevorgangs benötigt die CPU ausreichend freien Speicherplatz auf der SIMATIC Memory Card. Die vom Laden der Softwareänderungen in die CPU betroffenen Dateien werden erst gelöscht, nachdem die neuen Dateien angelegt wurden. Die SIMATIC Memory Card benötigt daher freien Speicherplatz in der Größenordnung des Speicherbedarfs aller Programmobjekte auf der Speicherkarte.

Informationen und Tipps zum Hantieren von Softwareänderungen finden Sie im Handbuch "Struktur und Verwendung des CPU-Speichers".

Peripheriedatenbereiche

Zugriff auf die Peripherieadressen

Peripherieadressen

Wenn Sie eine Baugruppe in der Gerätesicht stecken, liegen deren Nutzdaten im Prozessabbild der S7-1500-CPU (Voreinstellung). Die CPU übernimmt automatisch den Datenaustausch zwischen Baugruppe und Prozessabbild-Bereich beim Aktualisieren der Prozessabbilder.

Wenn das Programm direkt auf die Baugruppe und nicht über das Prozessabbild zugreifen soll, dann müssen Sie die Peripherieadresse durch ":P" ergänzen.

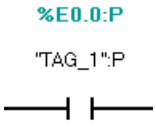

Für S7-1500-CPUs ergeben sich allerdings durch den direkten Peripheriezugriff keine Performance-Verbesserungen; Sie sollten die Peripherie daher i. d. R. über Prozessabbilder bzw. Teilprozessabbilder ansprechen.

Anfangsadresse einer Baugruppe

Definition

Die Anfangsadresse ist die niedrigste Byteadresse einer Baugruppe. Sie stellt die Anfangsadresse des Nutzdatenbereichs der Baugruppe dar.

Baugruppen-Anfangsadressen konfigurieren

Die Zuordnung zwischen Adressen, die im Anwenderprogramm verwendet werden, und den Baugruppen erfolgt durch die Konfiguration der Baugruppen.

Die beim Platzieren der Baugruppen automatisch vergebenen Anfangsadressen können Sie in den Eigenschaften der Baugruppe ändern (Bereich "E/A-Adressen").

Sie haben außerdem Möglichkeit einzustellen, ob die Adressen im Prozessabbild liegen bzw. in welchem Teilprozessabbild sie liegen.

Neben einem Teilprozessabbild können Sie auch den Organisationsbaustein wählen, dem das Teilprozessabbild zugeordnet ist. Mit Ausnahme des Taktsynchronalarm-OBs wird beim Aufruf des zugeordneten OBs das zugehörige Teilprozessabbild aktualisiert. Unmittelbar vor Beginn des OB-Starts wird das Teilprozessabbild der Eingänge aktualisiert, unmittelbar nach Ende des OBs wird das Teilprozessabbild der Ausgänge an die Baugruppen übertragen.

Wählen Sie das Prozessabbild "Automatische Aktualisierung", wenn das Prozessabbild, in dem die Adressen der Baugruppe liegen, am Zykluskontrollpunkt automatisch aktualisiert werden soll.

Wertstatus auswerten

Einführung

Beim Wertstatus (QI, Quality Information) handelt es sich um Diagnoseinformationen von Peripheriekanälen, die über das Prozessabbild der Eingänge (PAE) dem Anwenderprogramm zur Verfügung gestellt werden. Der Wertstatus wird synchron mit den Nutzdaten übertragen.

Jedes Bit des Wertstatus ist einem Kanal zugeordnet und gibt Auskunft über die Gültigkeit des Werts (0 = Wert ist fehlerhaft).

Liegt z. B. ein Drahtbruch an einem digitalen Eingang vor, ist das Nutzdatensignal logisch "0". Das Modul setzt aufgrund der Drahtbruch-Diagnose das zugehörige Bit im Wertstatus ebenfalls auf 0, d. h. Sie haben die Möglichkeit, durch Abfrage des Wertstatus festzustellen, dass der Wert nicht gültig ist.

Die Belegung der Wertstatus-Bytes im Prozessabbild der Eingänge hängt ab vom verwendeten Modul. Detaillierte Information finden Sie im Gerätehandbuch zum jeweiligen Peripheriemodul.

Zusammenfassung: Mögliche Ursache für Wertstatus = 0:

- die Versorgungsspannung L+ an den Klemmen fehlt oder ist nicht ausreichend
- ein Kanal wurde deaktiviert
- Ausgänge sind inaktiv (z.B. CPU in STOP)
- ein Ausgabekanal nutzt Technologiefunktionen (HSC, PWM oder PTO), siehe unten

Wertstatus aktivieren

Aktivieren Sie das Optionskästchen "Wertstatus" in den Eigenschaften des Peripheriemoduls, wenn Sie den Wertstatus des Kanals auswerten wollen. Durch die Aktivierung des Wertstatus werden zusätzliche Adressen im Prozessabbild der Eingänge (PAE) für den Wertstatus belegt. Bei Eingabemodulen belegt STEP 7 Eingangsadressen direkt im Anschluss an die Nutzdaten, bei Ausgabemodulen werden die nächsten freien Eingangsadressen belegt.

Besonderheiten bei Modulen mit MSI/MSO

Vorraussetzung:

- GSD-Datei für Peripheriemodule mit MSI/MSO
- IO-Device unterstützt MSI/MSO (z. B. IM 155-5 PN ST ab Firmware-Version V2.0.0)

Bei diagnosefähigen Modulen, für die Sie modulinternes Shared Input (MSI) bzw. modulinternes Shared Output (MSO) konfiguriert haben, hat der Wertstatus folgende Bedeutung:

MSI:

Beim 1. Submodul (=Basis-Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist.

Beim 2. bis 4. Submodul (=MSI-Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist oder dass das Basis-Submodul noch nicht parametriert ist (nicht betriebsbereit).

MSO:

Beim 1. Submodul (=Basis-Submodul) zeigt der Wertstatus 1 an, dass der vom Anwenderprogramm vorgegebene Ausgangswert auch tatsächlich an der Klemme des Moduls ausgegeben wird.

Mögliche Ursachen für Wertstatus = 0:

- Wert ist fehlerhaft, weil z. B. die Versorgungsspannung fehlt.
- IO-Controller vom Basis-Submodul befindet sich im STOP-Zustand.

Beim 2. bis 4. Submodul (=MSO-Submodul) zeigt der Wertstatus 1 an, dass der vom Anwenderprogramm vorgegebene Ausgangswert auch tatsächlich an der Klemme des Moduls ausgegeben wird.

Mögliche Ursachen für Wertstatus = 0:

- Wert ist fehlerhaft, weil z. B. die Versorgungsspannung fehlt.
- IO-Controller vom Basis-Submodul befindet sich im STOP-Zustand.
- Das Basis-Submodul ist noch nicht konfiguriert.

Verhalten des Wertstatus an Ausgabekanälen für Technologiefunktionen bei Kompakt-CPUs wie z. B. CPU 1511C-1 PN

Die CPU 1511C-1 PN unterstützt den Wertstatus (QI) für die Kanäle der analogen und digitalen Onboard-Peripherie.

Nicht unterstützt wird der Wertstatus von den Ausgabekanälen, welche Sie für Technologiefunktionen (HSC, PWM oder PTO) verwenden.

Aus diesem Grund liefern die QI-Bits an Ausgabekanälen für Technologiefunktionen stets den Wertstatus 0 ("Bad"). Dabei ist es unerheblich, ob der Ausgabewert fehlerhaft ist oder nicht.

Siehe auch

Zyklus-OBs (Seite 1057)

Beispiel für die Auswertung des Wertstatus (Seite 974)

Systemdiagnose (http://support.automation.siemens.com/WW/view/de/68052815)

Modulinternes Shared Input/Shared Output (MSI/MSO) (http://support.automation.siemens.com/WW/view/de/68052815)

Beispiel für die Auswertung des Wertstatus

Das Modul DI 32x24VDC HF der S7-1500 läßt sich in STEP 7 (Gerätesicht) für verschieden Einsatzbereiche konfigurieren. Je nach Konfiguration werden zusätzliche/unterschiedliche Adressen im Prozessabbild der Eingänge belegt.

In diesem Beispiel werden verschiedene Konfigurationen jeweils mit parametriertem Wertstatus gezeigt und die Auswirkung auf die Adressbelegung beschrieben.

Konfigurationsmöglichkeiten des DI 32x24VDC HF

Folgende Konfigurationen sind möglich:

- 1 x 32-kanalig ohne Wertstatus
- 1 x 32-kanalig mit Wertstatus
- 4 x 8-kanalig ohne Wertstatus
- 4 x 8-kanalig mit Wertstatus
- 1 x 32-kanalig mit Wertstatus für Modulinternes Shared Input (MSI) mit bis zu 4 Submodulen

Bei den Konfigurationen mit Wertstatus ist jedem Kanal ein zusätzliches Bit für den Wertstatus zugeordnet. Das Bit für den Wertstatus gibt an, ob der eingelesene Digitalwert gültig ist. (0 = Wert ist fehlerhaft).

Adressraum bei Konfiguration als 32-kanaliges DI 32x24VDC HF mit Wertstatus

Zunächst wird die voreingestellte DI-Konfiguration beibehalten (32 Kanäle; weder Aufteilung in Submodule noch modulinterne Shared-Input-Konfiguration (MSI)), es soll aber der Wertstatus aktiviert werden.

- Aktivieren Sie in den Eigenschaften des Moduls (Inspektorfenster) im Bereich "DI-Konfiguration" das Optionskästchen "Wertstatus freigeben".

Das folgende Bild zeigt die Belegung des Adressraums bei der Konfiguration als 32-kanaliges Modul mit Wertstatus. Für das Modul können Sie die Anfangsadresse frei vergeben. Die Adressen der Kanäle ergeben sich aus der Anfangsadresse.

Beispiel: Anfangsadresse 2 (Eingangsbyte 2), Endadresse 9 (Eingangsbyte 9).

STEP 7 reserviert automatisch die Adressen für die Eingangswerte (4 Bytes) und die darauffolgenden Adressen für den Wertstatus (ebenfalls 4 Bytes), so dass sich folgende Adressbelegung ergibt:

Belegung im Prozessabbild der Eingänge (PAE)

		Eingangswert:
	7 6 5 4 3 2 1 0	
EB 2		Kanal 0 bis 7 (Eingang CH0 bis CH7)
	15 8	
EB 3		Kanal 8 bis 15 (Eingang CH8 bis CH15)
	23 16	
EB 4		Kanal 16 bis 23 (Eingang CH16 bis CH23)
	31 24	
EB 5		Kanal 24 bis 31 (Eingang CH24 bis CH31)

		(QI) Wertstatus
	7 6 5 4 3 2 1 0	
EB 6		Kanal 0 bis 7 (Wertstatus QI0 bis QI7)
	15 8	
EB 7		Kanal 8 bis 15 (Wertstatus QI8 bis QI15)
	23 16	
EB 8		Kanal 16 bis 23 (Wertstatus QI16 bis QI23)
	31 24	
EB 9		Kanal 24 bis 31 (Wertstatus QI24 bis QI31)

QI=0: eingelesener Wert am Kanal ist fehlerhaft

Adressraum bei Konfiguration als 4 x 8-kanaliges DI 32x24VDC HF mit Wertstatus

Die DI-Konfiguration soll angepasst werden, so dass sich die 32 Kanäle des Moduls DI 32x24VDC HF der S7-1500 auf vier Submodule zu je 8 Kanälen aufteilen. Diese Submodule lassen sich beim Einsatz des Moduls in einem Shared Device unterschiedlichen IO-Controllern zugewiesen. Der Wertstatus soll auch in diesem Fall aktiviert sein.

- Wählen Sie in den Eigenschaften des Moduls (Inspektorfenster) im Bereich "DI-Konfiguration" die Submodul-Konfiguration "4 Submodule mit 8 digitalen Eingängen".
- Aktivieren Sie das Optionskästchen "Wertstatus freigeben".

In der Bereichsnavigation des Inspektorfensters erscheinen die 4 Submodule mit eigenem Namen (z. B. DI 32x24VDC BA_1)

Im Unterschied zur Konfiguration 1 x 32-kanaliges Modul hat jedes der vier Submodule eine frei vergebbare Anfangsadresse.

Beispiel: Die Anfangsadressen werden folgendermaßen vergeben:

- 1. Submodul: 10
- 2. Submodul: 20
- 3. Submodul 30
- 4. Submodul 40

STEP 7 reserviert automatisch für jedes Submodul ein Byte für den Wertstatus. Das Byte für den Wertstatus hat die Adresse "Anfangsadresse+1", so dass sich folgende Adressbelegung ergibt:

Belegung im Prozessabbild der Eingänge (PAE)

Eingangswert:

	7 6 5 4 3 2 1 0		
EB 10	15 … 8	Kanal 0 bis 7 (Eingang CH0 bis CH7)	1. Submodul
EB 20	23 … 16	Kanal 8 bis 15 (Eingang CH8 bis CH15)	2. Submodul
EB 30	31 … 24	Kanal 16 bis 23 (Eingang CH16 bis CH23)	3. Submodul
EB 40		Kanal 24 bis 31 (Eingang CH24 bis CH31)	4. Submodul

(QI) Wertstatus

	7 6 5 4 3 2 1 0		
EB 11	15 … 8	Kanal 0 bis 7 (Wertstatus QI0 bis QI7)	1. Submodul
EB 21	23 … 16	Kanal 8 bis 16 (Wertstatus QI8 bis QI15)	2. Submodul
EB 31	31 … 24	Kanal 16 bis 23 (Wertstatus QI16 bis QI23)	3. Submodul
EB 41		Kanal 24 bis 31 (Wertstatus QI24 bis QI31)	4. Submodul

QI = 0: eingelesener Wert am Kanal ist fehlerhaft

Adressraum bei Konfiguration als 1 x 32-kanaliges DI 32x24VDC HF mit MSI und Wertstatus

Die hier gezeigte Konfiguration nutzt die Funktion "Modulinternes Shared Input (MSI)". Diese Funktion erlaubt ein gleichzeitiges Lesen der Eingangswerte durch verschiedene IO-Controller (CPUs) beim Einsatz des Moduls in einem Shared Device (ET 200MP).

Bei der Konfiguration 1 x 32-kanaliges Modul mit modulinternem Shared Input (MSI) werden die Kanäle 0 bis 31 des Moduls in bis zu 4 Submodule kopiert. Die Kanäle 0 bis 31 sind dann mit identischen Eingangswerten in verschiedenen Submodulen vorhanden. Diese Submodule können beim Einsatz des Moduls in einem Shared Device bis zu vier IO-Controllern zugewiesen werden. Jeder IO-Controller kann auf dieselben Kanäle lesend zugreifen ("mithören").

Die Anzahl nutzbarer Submodule (= Kopien der Eingänge) ist abhängig vom eingesetzten Interfacemodul. Beachten Sie die Angaben hierzu im jeweiligen Gerätehandbuch des Interfacemoduls.

- Wählen Sie in den Eigenschaften des Moduls (Inspektorfenster) im Bereich "DI-Konfiguration" folgende Optionen:
  - Bei "Aufteilung des Moduls": Keine
  - Bei "Kopien des Moduls": Eine zusätzliche Kopie der Eingänge
    Diese Einstellung bewirkt , dass ein zusätzlicher IO-Controller die Eingänge des Moduls lesen kann.

Der Wertstatus ist bei MSI-Konfigurationen automatisch freigeben (nicht änderbar). Der Grund für diese Einstellung ist, dass ein "mithörender" IO-Controller des Shared Device über den Wertstatus feststellen kann, ob das Modul bereits betriebsbereit ist und gültige Werte anbietet.

Im Folgenden werden die Submodule, die durch die oben beschriebenen Einstellungen entstehen, folgendermaßen benannt:

- Basis-Submodul: Das ist das Submodul, das dem parametrierenden IO-Controller zugeordnet wird. Nur der parametrierende IO-Controller hat Zugriff auf Diagnosen des Moduls und empfängt Prozessalarme.
- MSI-Submodul: Das ist das Submodul, das einem "mithörenden" IO-Controller zugeordnet wird. Der mithörende IO-Controller hat keinen Zugriff auf Diagnosen des Moduls und erhät nur durch den Wertstatus Information über die Gültigkeit der Eingangswerte.

Wertstatus (Quality Information, QI)

Die Bedeutung des Wertstatus hängt davon ab, um welches Submodul es sich handelt.

Beim 1. Submodul (=Basis-Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist.

Beim 2. bis 4. Submodul (=MSI-Submodul bzw. Kopie der Eingänge) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist oder dass das Basis-Submodul noch nicht parametriert ist (nicht betriebsbereit).

Beispiel: Das Modul DI 32x24VDC HF ist wie oben beschrieben konfiguriert (Eine zusätzliche Kopie der Eingänge). Es entsteht ein Basis-Submodul und ein MSI-Submodul.

Das Basis-Submodul erhält in diesem Beispiel die Anfangsadresse 10, das MSI-Submodul erhält die Anfangsadresse 100. Die Adresse des MSI-Submoduls ist für den IO-Controller nicht relevant, weil im weiteren Verlauf der Shared -Device -Projektierung dieses Submoul einem anderen IO-Controller zugewiesen wird und damit aus dem Adressraum des parametrierenden IO-Controllers verschwindet.

Das folgende Bild zeigt die Belegung des Adressraums mit Submodul 1 und 2 und dem Wertstatus. Alle Submodule sind zunächst dem IO-Controller zugeordnet.

Belegung im Prozessabbild der Eingänge (PAE) für 1. und 2. Submodul

Eingangswert 1. Submodul (Basis-Submodul):

	7 6 5 4 3 2 1 0	
EB 10		Kanal 0 bis 7 (Eingang CH0 bis CH7)
EB 11	15 … 8	Kanal 8 bis 15 (Eingang CH8 bis CH15)
EB 12	23 … 16	Kanal 16 bis 23 (Eingang CH16 bis CH23)
EB 13	31 … 24	Kanal 24 bis 31 (Eingang CH24 bis CH31)
EB 14	7 6 5 4 3 2 1 0	Kanal 0 bis 7 (Wertstatus QI0 bis QI7)
EB 15	15 … 8	Kanal 8 bis 15 (Wertstatus QI8 bis QI15)
EB 16	23 … 16	Kanal 16 bis 23 (Wertstatus QI16 bis QI23)
EB 17	31 … 24	Kanal 24 bis 31 (Wertstatus QI24 bis QI31)

Eingangswert 2. Submodul (MSI-Submodul):

	7 6 5 4 3 2 1 0	
EB 100		Kanal 0 bis 7 (Eingang CH0 bis CH7)
EB 101	15 … 8	Kanal 8 bis 15 (Eingang CH8 bis CH15)
EB 102	23 … 16	Kanal 16 bis 23 (Eingang CH16 bis CH23)
EB 103	31 … 24	Kanal 24 bis 31 (Eingang CH24 bis CH31)
EB 104	7 6 5 4 3 2 1 0	Kanal 0 bis 7 (Wertstatus QI0 bis QI7)
EB 105	15 … 8	Kanal 8 bis 15 (Wertstatus QI8 bis QI15)
EB 106	23 … 16	Kanal 16 bis 23 (Wertstatus QI16 bis QI23)
EB 107	31 … 24	Kanal 24 bis 31 (Wertstatus QI24 bis QI31)

QI = 0: eingelesener Wert am Kanal ist fehlerhaft (Basis-Submodul) bwz. Modul ist noch nicht betriebsbereit (MSI-Submodul)

Weitere Infomationen

Weitere Informationen und ein Programmierbeispiel zum Auswerten des Wertstatus finden Sie hier (http://support.automation.siemens.com/WW/view/de/59192926).

Siehe auch

Wertstatus auswerten (Seite 972)

Grundlagen zur Programmbearbeitung

Englische Bezeichnungen der OB-Typen

Englische Bezeichnung der OB-Typen

Beim Anlegen eines OB im Dialog "Neuen Baustein hinzufügen" können die OB-Typen aus technischen Gründen nur in Englisch angezeigt werden – unabhängig von der eingestellten Oberflächensprache. Um Ihnen die Handhabung zu erleichtern, werden in der folgenden Tabelle den englischen Begriffen die Bezeichnungen in der eingestellten Oberflächensprache gegenübergestellt.

Englische Bezeichnung des OB-Typs im Dialog "Neuen Baustein hinzufügen"	Bezeichnung im Informationssystem
Program cycle	Zyklus-OB, OB für zyklische Programmbearbeitung, Zyklisches Programm
Startup	Anlauf-OB
Time delay interrupt	Verzögerungsalarm-OB
Cyclic interrupt	Weckalarm-OB
Hardware interrupt	Prozessalarm-OB
Time error interrupt	Zeitfehler-OB
Diagnostic error interrupt	Diagnosealarm-OB
Pull or plug of modules	Ziehen/Stecken-OB
Rack or station failure	Baugruppenträgerausfall-OB
Programming error	Programmierfehler-OB
IO access error	Peripheriezugriffsfehler-OB
Time of day	Uhrzeitalarm-OB
MC-Interpolator	MC-Interpolator-OB
MC-Servo	MC-Servo-OB
MC-PreServo	MC-PreServo-OB
MC-PostServo	MC-PostServo-OB
MC-Transformation	MC-Transformation-OB
Synchronous cycle	Taktsynchronalarm-OB
Status	Statusalarm-OB
Update	Update-Alarm-OB
Profile	OB für hersteller- bzw. profilspezifischen Alarm

Ereignisse und OBs

OB-Startereignisse

Ein OB-Startereignis hat nach seinem Auftreten diese Reaktion zur Folge:

- Falls das Ereignis aus einer Ereignisquelle stammt, der Sie einen OB zugeordnet haben, stößt es die Ausführung des zugeordneten OB an. Das bedeutet, dass das Ereignis in eine seiner Priorität entsprechende Warteschlange eingereiht wird.
- Falls das Ereignis aus einer Ereignisquelle stammt, der Sie keinen OB zugeordnet haben, wird die voreingestellte Systemreaktion durchgeführt.

Hinweis

Einige Ereignisquellen sind auch ohne Ihre Konfiguration vorhanden, z. B. Anlauf, Ziehen/Stecken.

Die folgende Tabelle gibt einen Überblick über die OB-Startereignisse samt den möglichen Werten für die OB-Priorität, möglichen OB-Nummern, voreingestellter Systemreaktion und OB-Anzahl. Sie ist nach der voreingestellten OB-Priorität sortiert. Dabei entspricht 1 der niedrigsten Priorität.

Typen von Ereignisquellen	Mögliche Prioritäten (Voreingestellte Priorität)	Mögliche OB-Nummern	Voreingestellte Systemreaktion	Anzahl unterstützter OBs
Anlauf	1	100, >= 123	Ignorieren	100
Zyklisches Programm	1	1, >= 123	Ignorieren	100
Uhrzeitalarm	2 bis 24 (2)	10 bis 17, >= 123	nicht zutreffend	20
Statusalarm	2 bis 24 (4)	55	Ignorieren	1
Update-Alarm	2 bis 24 (4)	56	Ignorieren	1
Hersteller- bzw. profilspezifischer Alarm	2 bis 24 (4)	57	Ignorieren	1
Verzögerungsalarm	2 bis 24 (3)	20 bis 23, >= 123	nicht zutreffend	20
Weckalarm	2 bis 24 (8 bis 17, abhängig von der Zykluszeit)	30 bis 38, >= 123	nicht zutreffend	20
Prozessalarm	2 bis 26 (16)	40 bis 47, >= 123	Ignorieren	50
Taktsynchronalarm	16 bis 26 (21)	61 bis 64, >=123	Ignorieren	20 (einer je taktsynchroner Schnittstelle)
CPU-Redundanzfehler (nur bei R/H-CPUs)	2 bis 26 (26)	72	Ignorieren	1
MC-Interpolator	16 bis 25 (24)	92	nicht zutreffend	1
MC-Servo	17 bis 26 (26)	91	nicht zutreffend	1
MC-PreServo	Entspricht MC-Servo	67	nicht zutreffend	1
MC-PostServo	Entspricht MC-Servo	95	nicht zutreffend	1
MC-Transformation	17 bis 25 (25)	98	nicht zutreffend	1
Zeitfehler [1)]	22	80	Ignorieren	1
Zyklusüberwachungszeit ein mal überschritten			STOP	
Diagnosealarm	2 bis 26 (5)	82	Ignorieren	1

Typen von Ereignisquellen	Mögliche Prioritäten (Voreingestellte Priorität)	Mögliche OB-Nummern	Voreingestellte Systemreaktion	Anzahl unterstützter OBs
Ziehen/Stecken von Modulen	2 bis 26 (6)	83	Ignorieren	1
Baugruppenträgerfehler	2 bis 26 (6)	86	Ignorieren	1
Programmierfehler (nur bei globaler Fehlerbehandlung)	2 bis 26 (7)	121	STOP	1
Peripheriezugriffsfehler (nur bei globaler Fehlerbehandlung)	2 bis 26 (7)	122	Ignorieren	1

1) Wie verhält sich ein R/H-System bei Zykluszeitüberschreitungen? Siehe dazu "Redundantes System S7-1500R/H Systemhandbuch".

Zuordnung zwischen Ereignisquelle und OBs

Wo Sie die Zuordnung zwischen OB und Ereignisquelle vornehmen, hängt vom OB-Typ ab:

- Bei Prozessalarmen und Taktsynchronalarmen erfolgt die Zuordnung bei der Konfigurierung der Hardware oder beim Anlegen des OB.
- Beim MC-Servo-Alarm und MC-Interpolator-Alarm ordnet STEP 7 automatisch die OBs 91/92 zu, sobald ein Technologieobjekt "S7-1500 Motion Control" hinzugefügt wird.
- Bei allen anderen OB-Typen erfolgt die Zuordnung beim Anlegen des OB, ggf. nachdem Sie die Ereignisquelle konfiguriert haben.

Eine einmal getroffene Zuordnung können Sie bei den Prozessalarmen zur Laufzeit mit den Anweisungen ATTACH und DETACH wieder ändern. Dabei wird nicht die konfigurierte, sondern lediglich die tatsächlich wirksame Zuordnung geändert. Die konfigurierte Zuordnung wird nach dem Laden und bei jedem Anlauf wirksam.

Prozessalarme, denen kein OB zugeordnet ist (weil so konfiguriert oder nach Aufruf der Anweisung DETACH), werden ignoriert. Die Prüfung, ob einem Ereignis ein OB zugeordnet ist, findet nicht beim Eintreffen des zugehörigen Ereignisses statt, sondern erst, wenn der Prozessalarm tatsächlich bearbeitet werden soll.

OB-Priorität und Ablaufverhalten

Die S7-1500-CPUs unterstützen die Prioritäten 1 (niedrigste Priorität) bis 26 (höchste Priorität). Ein OB erhält die Priorität seines Startereignisses.

Die OBs werden rein prioritätsgesteuert bearbeitet. Das heißt, dass bei gleichzeitigem Vorliegen mehrerer OB-Anforderungen der OB mit der höchsten Priorität zuerst bearbeitet wird. Wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt, wird dieser OB unterbrochen. Ereignisse gleicher Priorität werden in der Reihenfolge ihres Auftretens bearbeitet.

Überlastverhalten der CPU

Prinzip des Überlastverhaltens der CPU

Der Ablauf des Anwenderprogramms bei S7-1500-CPUs basiert auf Ereignissen unterschiedlicher Priorität. Ein Ereignis stößt die Ausführung eines OB an. Die OBs werden rein prioritätsgesteuert bearbeitet. Das heißt, dass bei gleichzeitigem Vorliegen mehrerer OB-Anforderungen der OB mit der höchsten Priorität zuerst bearbeitet wird. Wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt, wird dieser OB unterbrochen. Ereignisse gleicher Priorität werden in der Reihenfolge ihres Auftretens bearbeitet.

Für die im Folgenden betrachteten Ereignis-Szenarien wird vorausgesetzt, dass Sie jeder Ereignisquelle einen OB zugeordnet haben und dass diese OBs dieselbe Priorität haben. Insbesondere die zweite Voraussetzung dient lediglich einer vereinfachten Darstellung.

Wenn ein Ereignis auftritt, wird die Ausführung des zugehörigen OB angestoßen. Das bedeutet, dass es in eine seiner Priorität entsprechende Warteschlange eingereiht wird. Abhängig von der OB-Priorität und der aktuellen Prozessorlast kommt es meist zu einer verzögerten Ausführung des OB. Dasselbe Ereignis kann daher einmal oder mehrmals erneut auftreten, bevor der zum vorhergehenden Ereignis gehörende OB bearbeitet wurde. Eine solche Situation wird von der CPU wie folgt behandelt: Das Betriebssystem reiht die Ereignisse in der Reihenfolge ihres Auftretens in die ihrer Priorität zugehörige Warteschlange ein.

Die Zahl anstehender Ereignisse, die aus ein und derselben Quelle eingekettet werden, können Sie begrenzen, um temporäre Überlastsituationen zu beherrschen. Sobald die maximale Zahl anstehender Startereignisse z. B. eines bestimmten Weckalarm-OB erreicht ist, wird das nächste Ereignis verworfen.

Eine Überlast entsteht, wenn Ereignisse, die aus ein und derselben Quelle stammen, schneller auftreten, als sie verarbeitet werden können.

Die Details werden in den folgenden Abschnitten näher erläutert.

Das Nachholen und das Verwerfen von gleichartigen Ereignissen

Im Folgenden werden mit dem Begriff "gleichartige Ereignisse" die Ereignisse aus einer Quelle verstanden, also z. B. die Starterereignisse für einen bestimmten Weckalarm-OB.

Mit dem OB-Parameter "Anzahl einreihbarer Ereignisse" legen Sie fest, wie viele gleichartige Ereignisse das Betriebssystem in die zugehörige Warteschlange einreiht und folglich nachbearbeitet. Hat dieser Parameter z. B. den Wert 1, wird genau ein Ereignis zwischengespeichert.

Hinweis

Das Nachbearbeiten von zyklischen Ereignissen ist oft unerwünscht, da es zu einer Überlast bei gleich- oder niedrigerprioren OBs führen kann. Daher ist es meist vorteilhaft, entsprechende Ereignisse zu verwerfen und bei der nächsten regulären OB-Bearbeitung auf die Überlastsituation zu reagieren. Ein kleiner Wert des Parameters "Anzahl einreihbarer Ereignisse" sorgt dafür, dass eine Überlastsituation nicht verstärkt, sondern abgemildert wird.

Wenn z. B. für einen Weckalarm-OB die maximale Anzahl von Startereignissen in der Warteschlange erreicht ist, wird jedes weitere Startereignis lediglich gezählt und anschließend verworfen. Bei der nächsten regulären OB-Bearbeitung wird Ihnen die Anzahl verworfener

Startereignisse im Input-Parameter "event_count" zur Verfügung gestellt (in der optimierten Startinformation). Dann können Sie auf die Überlastsituation geeignet reagieren. Anschließend setzt die CPU den Zähler für verlorene Ereignisse auf Null.

Wenn die CPU z. B. ein Startereignis eines Weckalarm-OB erstmals verwirft, hängt ihr weiteres Verhalten vom OB-Parameter "Diagnosepuffereintrag bei Ereignisüberlauf" ab: Bei gesetztem Optionskästchen trägt die CPU für die Überlastsituation an dieser Ereignisquelle einmalig das Ereignis DW#16#0002:3507 in den Diagnosepuffer ein. Weitere Diagnosepuffereinträge des Ereignisses DW#16#0002:3507, die sich auf diese Ereignisquelle beziehen, werden unterdrückt, bis alle Ereignisse aus dieser Quelle nachbearbeitet sind.

Schwellwert-Mechanismus zur Anforderung des Zeitfehler-OB

Mit dem OB-Parameter "Zeitfehler freigeben" ("Enable time error") legen Sie für gleichartige Ereignisse fest, ob bei einer bestimmten Überlast der Zeitfehler-OB aufgerufen werden soll.

Falls ja (gesetztes Optionskästchen), legen Sie mit dem OB-Parameter "Ereignisschwelle für Zeitfehler" fest, bei welcher Anzahl gleichartiger Ereignisse in der Warteschlange der Zeitfehler-OB aufgerufen werden soll. Hat dieser Parameter z. B. den Wert 1, wird beim Auftreten des zweiten Ereignisses das Ereignis DW#16#0002:3502 einmalig in den Diagnosepuffer eingetragen und der Zeitfehler-OB angefordert. Weitere Diagnosepuffereinträge des Ereignisses DW#16#0002:3502, die sich auf diese Ereignisquelle beziehen, werden unterdrückt, bis alle Ereignisse aus dieser Quelle nachbearbeitet sind.

Damit haben Sie die Möglichkeit, bei Überlast bereits deutlich vor Erreichen der Grenze für gleichartige Ereignisse und damit vor dem Verwerfen von Ereignissen eine Reaktion zu programmieren.

Für den Parameter "Ereignisschwelle für Zeitfehler" gilt der folgende Wertebereich: 1 <= "Ereignisschwelle für Zeitfehler" <= "Anzahl einreihbarer Ereignisse"

Beispiel für ein Prozessalarm-Ereignis

Am Beispiel eines Prozessalarm auslösenden Moduls wird die Funktionsweise der ereignisorientierten Bearbeitung des Programms in der S7-1500-CPU beschrieben.

Prozess-Ereignisse sind Ereignisse, die von der Peripherie (z. B. einem digitalen Eingang) ausgelöst werden und in der Folge in der S7-1500-CPU den Aufruf eines zugeordneten OB bewirken. OBs, die einem Prozess-Ereignis zugeordnet sind, werden daher Prozessalarm-OBs genannt.

Vorgehen

Um einen Prozessalarm zu projektieren gehen Sie folgendermaßen vor:

1. Konfigurieren Sie eine S7-1500 mit CPU und einem prozessalarmfähigen Modul, z. B. DI 16x24VDC HF.
2. Markieren Sie das prozessalarmfähige Modul und navigieren Sie in der Bereichsnavigation auf den Bereich "Eingänge > Kanal 0".
3. Aktivieren Sie die Option "Erkennung der steigenden Flanke aktivieren".
4. Geben Sie dem Ereingnis einen aussagefähigen Namen.

5. Ändern Sie, falls erforderlich, die Priorität dieses Ereignisses.
   In den PLC-Variablen (Register "Systemkonstanten") wird für das Ereignis eine Systemkonstante vom Datentyp Event_HwInt angelegt.

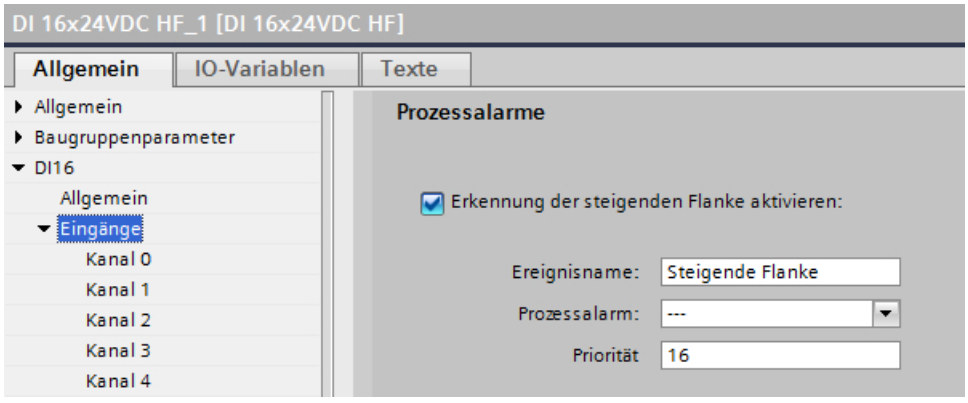

6. Klicken Sie auf die Klappliste "Prozessalarm", um einen neuen Prozessalarm-OB anzulegen. Falls bereits ein Prozessalarm-OB vorhanden ist, kann er ausgewählt werden.

7. Klicken Sie auf die Schaltfläche "Objekt hinzufügen", um einen neuen Prozessalarm-OB anzulegen.

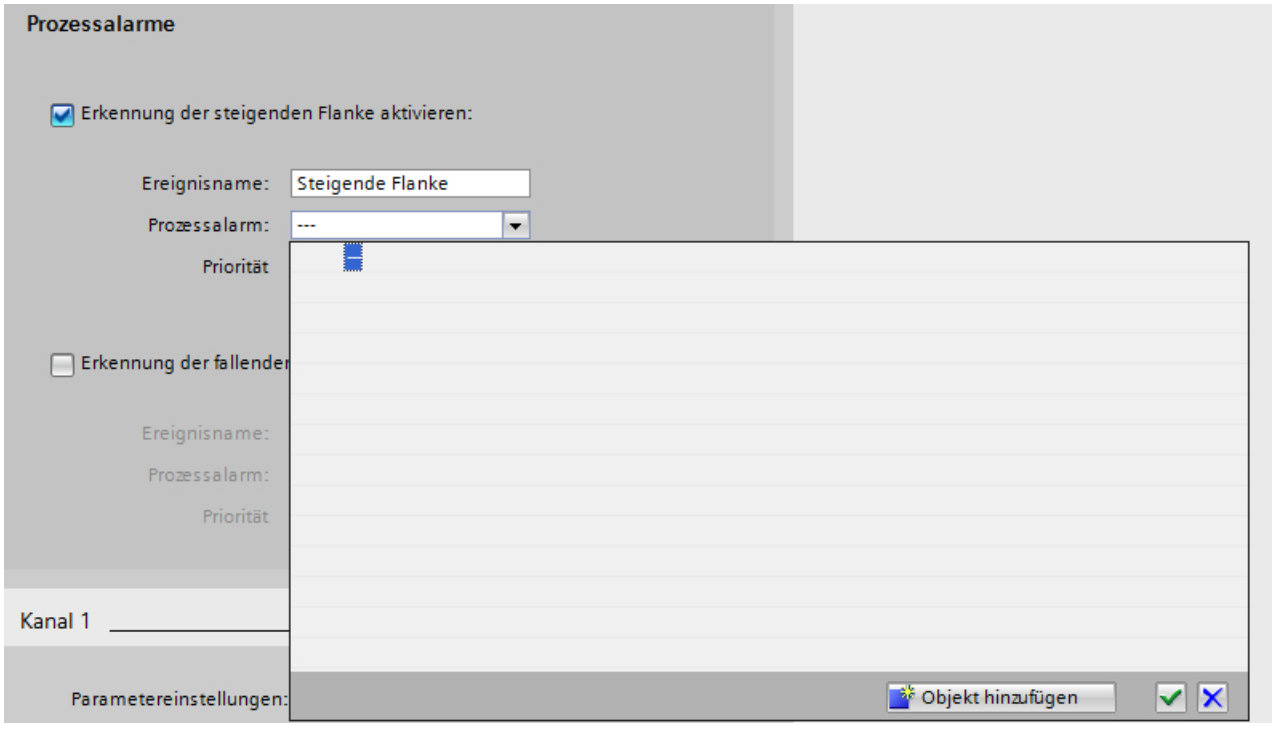

8. Im Auswahldialog können Sie entscheiden, ob sofort der Programmiereditor mit geöffnetem Prozessalarm-OB starten soll (Option "Neu hinzufügen und öffnen").

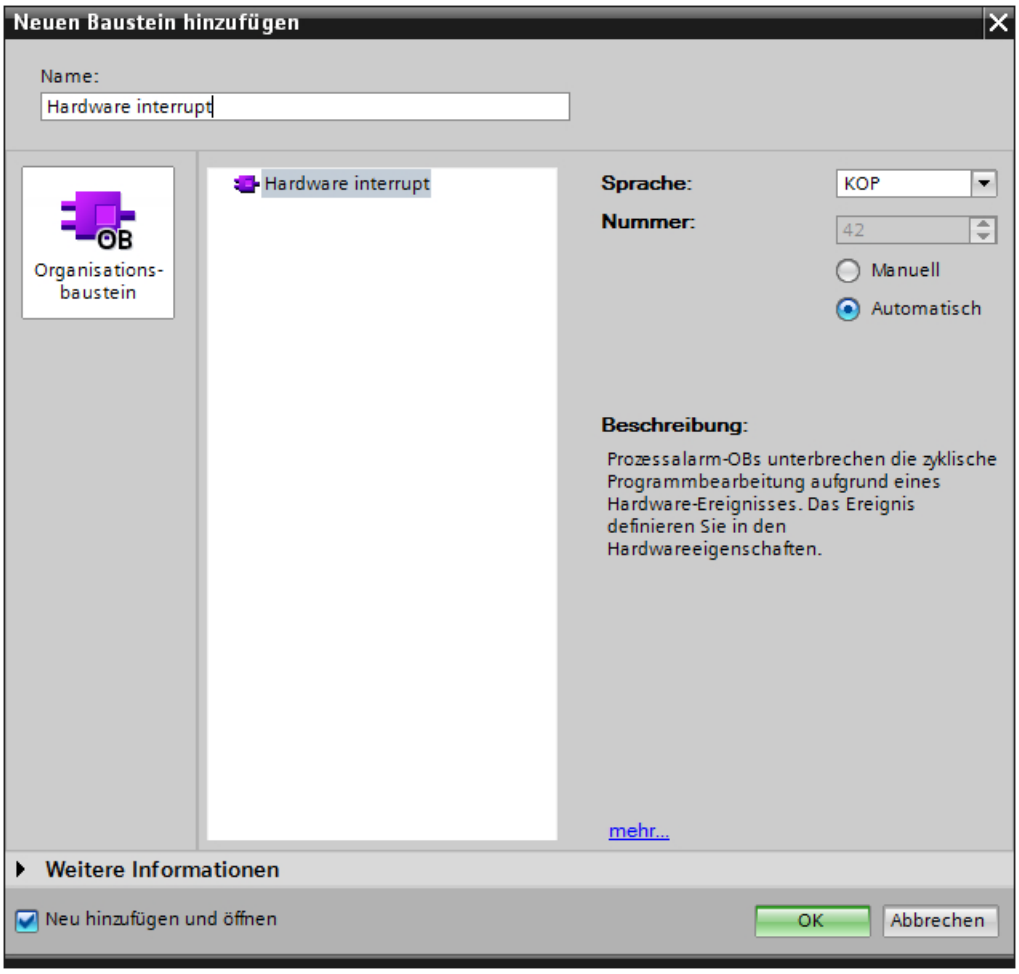

Prozessalarm-OB-Eigenschaften

In den Eigenschaften des Prozessalarm-OB finden Sie für jedes alarmauslösendes Ereignis die entsprechenden Informationen zum Startereignis:

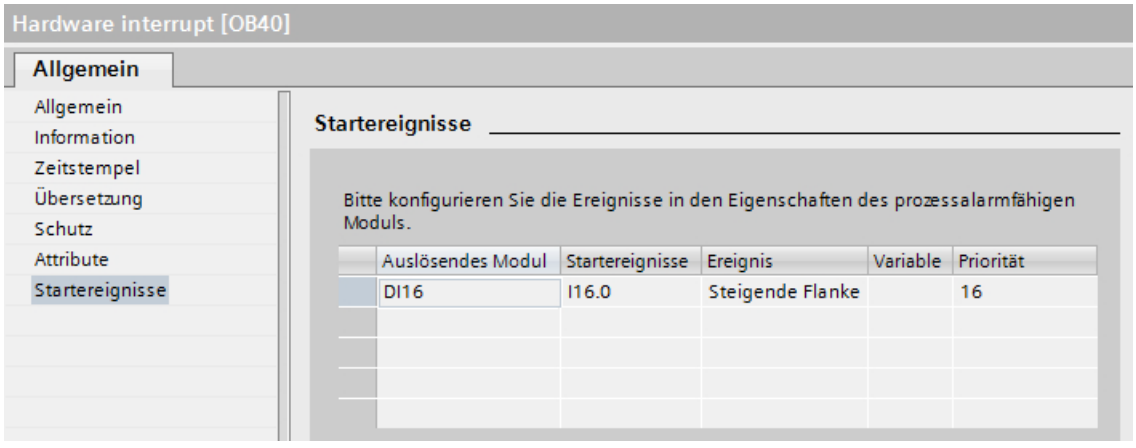

Prozessalarm-OB-Programm

Im Anwenderprogramm können Sie die Startinformationen des Prozessalarm-OB dazu verwenden, das Prozessalarm-Ereignis zu identifizieren und auszuwerten.

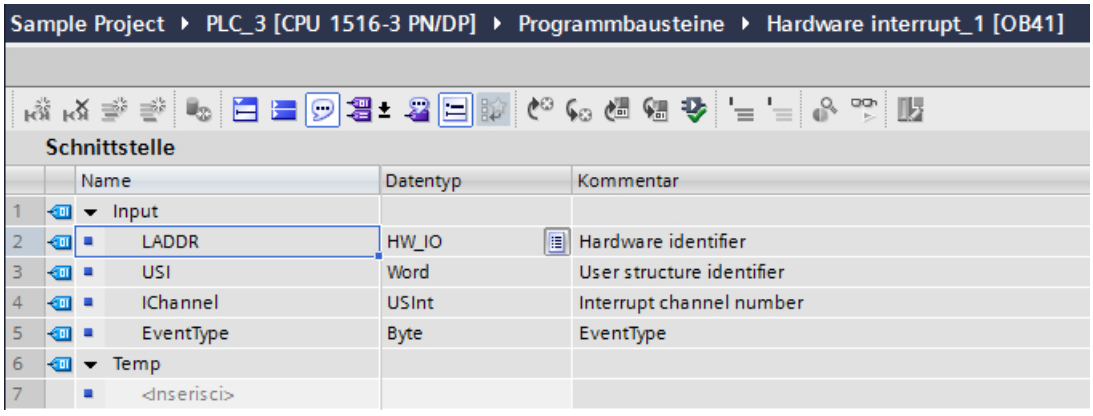

Siehe auch

Prozessalarm-OBs parametrieren (Seite 1091)

Prozessalarm-OBs (Seite 1068)

Einstellen des Betriebsverhaltens

CPU-Eigenschaften

Übersicht über die CPU-Eigenschaften

Übersicht

Die folgende Tabelle zeigt die Eigenschaften der CPU im Überblick:

Gruppe	Eigenschaften	Beschreibung
Allgemein	Projektinformationen	Allgemeine Angabe zur Beschreibung der gesteckten CPU. Bis auf die Steckplatznummer können Sie die Angaben ändern.
	Kataloginformationen	Nur lesbare Informationen aus dem Hardware-Katalog für diese CPU.
	Identification & Maintenance	Zur Hinterlegung einsatzspezifischer Informationen wie dem Namen der Anlage und dem Einbauort.
	Prüfsummen	Zur Prüfung der Identität bzw. Integrität von PLC-Programmen. Bausteine im Bausteinordner und Textlisten werden beim Übersetzen automatisch mit eindeutigen Prüfsummen gekennzeichnet. Sie können leicht feststellen, ob das Programm, das aktuell auf der CPU läuft, dasselbe Programm ist, das Sie vor längerer Zeit geladen haben, oder ob das Programm zwischenzeitlich geändert wurde. Um die Prüfsumme zur Laufzeit im Programm auszulesen, steht die Anweisung "GetChecksum" zur Verfügung. Siehe: Vergleich von PLC-Programmen anhand von Prüfsummen

Gruppe	Eigenschaften	Beschreibung
PROFINET-Schnitt-stelle	Allgemein	Name und Kommentar für diese PROFINET-Schnittstelle. Der Name ist auf 110 Zeichen begrenzt.
	Ethernet-Adressen	Auswahl, ob die PROFINET-Schnittstelle vernetzt ist. Wenn bereits Subnetze im Projekt angelegt wurden, stehen Sie zur Auswahl in der Klappliste. Ansonsten kann mit der Schaltfläche "Neues Subnetz hinzufügen" ein neues Subnetz angelegt werden. Unter IP-Protokoll stehen Angaben zur IP-Adresse, Subnetzmaske und IP-Router-Verwendung im Subnetz. Wenn ein IP-Router verwendet wird, ist die Angabe der IP-Adresse des IP-Routers notwendig. Siehe: Adressen- und Namensvergabe für PROFINET-Geräte (Seite 2989)
	Uhrzeitsynchronisation	Einstellungen zur Uhrzeitsynchronisation im NTP-Uhrzeitformat. Das NTP (network time protocol) ist ein allgemeines Verfahren zur Synchronisation von Systemuhren in lokalen und globalen Netzwerken. Beim NTP-Verfahren sendet die Schnittstelle der CPU in regelmäßigen Zeitabständen Uhrzeitanfragen (im Client-Modus) an NTP-Server im Subnetz (LAN), deren Adressen hier parametriert werden müssen. Anhand der Antworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und synchronisiert. Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg. Die Genauigkeit ist abhängig von der Qualität des verwendeten NTP-Servers.
	Betriebsart	Auswahl des Betriebes der CPU als IO-Controller bzw. IO-Device inklusive der jeweils erforderlichen Einstellungen. Als IO-Device (I-Device) müssen Sie einen IO-Controlle zugeweisen und einstellen, ob die PROFINET-Schnittstelle durch den übergeordneten IO-Controller parametriert wird oder der lokale IO-Controller die PROFINET-Schnittstelle parametriert. Außerdem ist für die Rolle IO-Device der priorisierte Hochlauf aktivierbar, sofern der übergeordnete IO-Controller die Funktion unterstützt. Wenn Sie die CPU als IO-Device parametriert haben, erscheint die Tabelle zum Projektieren der Transferbereiche. Siehe: I-Device projektieren
	Erweiterte Optionen	Name, Kommentar und weitere Einstellmöglichkeiten der Ethernet-Schnittstelle und den Ports. Darunter fallen z B. folgende Einstellungen: • Schnittstellen-Optionen wie die Reaktion des Anwenderprogramms bei Kommunikationsfehlern, ob Gerätetausch ohne Wechselmedium bzw. die automatische Inbetriebnahme möglich ist, Einstellungen hinsichtlich des IEEE-konformen LLDP-Modus (PROFINET V2.3), zeitlicher Abstand für das Senden von Keep Alives für Verbindungen. • Medienredundanz-Einstellungen wie z. B. die Medienredundanzrolle (Client, Manager (Auto) oder nicht Teilnehmer des Medienredundanz-Rings), Anzeige der Ringports und ob bei Redundanzfehlern Diagnosealarme erzeugt werden. Eine Schaltfläche "Domain-Einstellungen"

Gruppe	Eigenschaften	Beschreibung
		verlinkt zu den Domain-Einstellungen des angeschlossenen Subnetzes (MRP-Domain und Sync-Domain). • Echtzeit-Einstellungen wie Sendetakt und Synchronisationseigenschaften mit jeweils einer Schaltfläche "Domain-Einstellungen" verlinkt zu den Domain-Einstellungen des angeschlossenen Subnetzes (MRP-Domain und Sync-Domain). In der Sync-Domain stellen Sie den Sendetakt für taktsynchrone IO-Systeme ein und bestimmen die RT-Klassen (RT und IRT) sowie bei IRT die Synchronsiationsrollen. • Ports: Für die Ports stellen Sie die Verschaltung mit Nachbargeräten ein (alternativ: über Topologiesicht), wenn erforderlich. Erforderlich sind Port-Verschaltungen für taktsynchrone Anwendungen (IRT) sowie für die automatische Inbetriebnahme (Option "Gerätetausch ohne Wechselmedium" ist aktiviert). Außerdem stellen Sie Port-Optionen ein, wie z. B., ob der Port verwendet wird (nicht genutzte Ports können Sie sperren). Ebenfalls zu den Port-Optionen gehören die Verbindungseinstellungen (Übertragungsrate/Duplex) mit den Optionen für Überwachung und Autonegotiation. Ebenfalls an den Ports stellen Sie Boundaries ein für das Ende der Erfassung erreichbarer Teilnehmer, für das Ende der Topologieerkennung und das Ende der Sync-Domain. Siehe: Sendetakt einstellen (Seite 3016) Ports verschalten (Seite 3014) Festlegen der Port-Optionen (Seite 3021) Wissenswertes zur Medienrdundanz Einführung: Isochronous Realtime Ethernet Gerätetausch ohne Wechselmedium aktivieren (Seite 3026)
	Webserver-Zugriff	Ermöglicht den Zugriff auf die Webserver-Funktion.

Gruppe	Eigenschaften	Beschreibung
PROFIBUS-Schnittstelle	Allgemein	Name und Kommentar für diese PROFIBUS-Schnittstelle. Der Name ist auf 110 Zeichen begrenzt.
	PROFIBUS-Adresse	Auswahl, ob die PROFIBUS-Schnittstelle vernetzt ist. Wenn bereits Subnetze im Projekt angelegt wurden, stehen Sie zur Auswahl in der Klappliste. Ansonsten kann mit der Schaltfläche "Neues Subnetz hinzufügen" ein neues Subnetz angelegt werden. Die PROFIBUS-Adresse wird vergeben und die höchste Adresse und die Übertragungsgeschwindigkeit werden angezeigt.
	Betriebsart	Auswahl der Betriebsart der CPU als DP-Master oder DP-Slave. Einem DP-Slave kann ein DP-Master zugewiesen werden und die I-Slave-Kommunikation kann hier parametriert werden.
	Uhrzeitsynchronisation	Die Uhrzeit kann synchronisiert werden. Eine Vorgabe der Uhrzeitquelle und der Häufigkeit der Synchronisation kann vorgenommen werden.
	SYNC/FREEZE	Der DP-Master kann an eine Gruppe von DP-Slaves die Steuerkommandos SYNC und/oder FREEZE zur Synchronisation der DP-Slaves senden. Die DP-Slaves werden dazu SYNC/FREEZE-Gruppen zugeordnet.

Gruppe	Eigenschaften	Beschreibung
Anlauf	Anlauf nach NETZ-EIN	Einstellung des Anlaufverhaltens nach einem NETZ-AUS/NETZ-EIN-Übergang. Siehe: Grundlagen zum Betriebszustand "ANLAUF" (Seite 953)
	Vergleich Sollausbau zu Istausbau	Legt das Anlaufverhalten der CPU fest für den Fall, dass der tatsächliche Aufbau der S7-1500-Station nicht dem konfigurierten Aufbau entspricht: • Anlauf der CPU nur bei Kompatibilität • Anlauf der CPU auch bei Unterschieden In der Einstellung "Anlauf der CPU nur bei Kompatibilität" muss ein Modul auf einem konfigurierten Steckplatz kompatibel zum konfigurierten Modul sein. Kompatibel heißt, dass das vorhandene Modul hinsichtlich seiner Anzahl Ein-/Ausgänge passt und hinsichtlich seiner elektrischen und funktionalen Eigenschaften übereinstimmen muss. Ein kompatibles Modul muss ein konfiguriertes Modul vollständig ersetzen können; es darf mehr können, aber nicht weniger. Beispiel für die Einstellung "Anlauf der CPU nur bei Kompatibilität": Eine CPU kann ein kompatibler Ersatz für eine CPU des selben Typs mit einem niedrigeren Firmwarestand sein. Ein Eingangsmodul mit 32 Digitaleingängen kann ein kompatibler Ersatz für ein Signalmodul mit 16 Digitaleingängen sein. Die CPU läuft an, wenn das konfigurierte Modul oder ein kompatibles Modul steckt. Wenn ein nicht-kompatibles Modul steckt, läuft die CPU nicht an. Beispiel für die Einstellung "Anlauf der CPU auch bei Unterschieden": Anstelle eines konfigurierten Digitaleingabemoduls wird ein Analogausgabemodul gesteckt oder es ist kein Modul auf diesem Steckplatz vorhanden. Obwohl die konfigurierten Eingänge nicht erreichbar sind, läuft die CPU an. Beachten Sie, dass in diesen Fall das Anwenderprogramm nicht ordungsgemäß funktionieren kann und treffen Sie entsprechende Maßnahmen! **Hinweis**: Für S7-1500-Module und ET 200SP-Module können Sie für jeden Steckplatz den Vergleich Sollbaugruppe zu Istbaugruppe einzeln einstellen. Die Voreinstellung ist "Von CPU" (d. h. für den Steckplatz gilt die Einstellung der CPU). Für Peripheriemodule aus anderen Peripheriefamilien gilt generell die Einstellung der CPU (nicht änderbar).
	Parametrierungszeit	Legt einen maximalen Zeitraum (Voreinstellung: 60000 ms) fest, in dem die zentrale und dezentrale Peripherie betriebsbereit sein muss. Die CMs und CPs werden während des Anlaufs von der CPU mit Spannung und Kommunikationsparametern versorgt. Die Parametrierungszeit begrenzt den Zeitraum, während dessen die an CM oder CP angeschlossenen E/A-Module betriebsbereit sein müssen. Wenn die zentrale und dezentrale Peripherie nicht innerhalb der Parametrierungszeit betriebsbereit ist, hängt das Anlaufverhalten der CPU von der Einstellung des Parameters "Unterstützte Hardwarekompatibilität" ab.

Gruppe	Eigenschaften	Beschreibung
Uhr	-	Festlegung zur Synchronisation der Uhr über den Rückwandbus. Die Einstellungsmöglichkeiten sind CPU-abhängig. • Als Slave: Die Uhr wird von einer anderen Uhr synchronisiert. • Als Master: Die Uhr synchronisiert andere Uhren. • Keine: Es findet keine Synchronisation statt. Die Zeitintervalle für die Synchronisation können ausgewählt werden. Siehe: AUTOHOTSPOT
Zyklus	Maximale Zykluszeit (Zyklusüberwachungszeit) und Mindestzykluszeit	Festlegen einer maximalen Zykluszeit (Zyklusüberwachungszeit) oder einer festen Mindestzykluszeit. Wenn die Zykluszeit die maximale Zykluszeit überschreitet, dann ruft die CPU den Organisationsbaustein OB 80 "Zeitfehler" auf. Ist der OB 80 nicht vorhanden, dann geht die CPU in den Betriebszustand STOP. Wenn in einem Zyklus die maximale Zykluszeit zwei Mal überschritten wird, dann geht die CPU in den Betriebszustand STOP, auch wenn der OB 80 vorhanden ist. Ursachen für eine Überschreitung: • Kommunikationsprozesse • Häufung von Alarmereignissen • Fehler im CPU-Programm. Ist die Zykluszeit kürzer als die eingegebene Mindestzykluszeit, dann wartet die CPU so lange, bis die Mindestzykluszeit erreicht ist. In dieser zusätzlichen Programmbearbeitungszeit bearbeitet die CPU den OB 90 (Hintergrund-OB), falls er geladen ist. Siehe: Zykluszeit und maximale Zykluszeit (Zyklusüberwachungszeit) (Seite 999)
Kommunikationslast	Zyklusbelastung durch Kommunikation	Steuert in einem gewissen Rahmen die Dauer von Kommunikationsprozessen, die immer auch die Zykluszeit verlängern. Kommunikationsprozesse können z. B. sein: Datenübertragung zu einer anderen CPU oder Laden von Bausteinen (über PC angestoßen). Siehe: Zyklusbelastung durch Kommunikation (Seite 1000)
System- und Taktmerker	Systemmerkerbits und Taktmerkerbits	Setzen eines Bytes für die Systemmerkerfunktionen und Setzen eines Bytes für die Taktmerkerfunktionen (dabei schaltet jedes Bit in einer vorgegebenen Frequenz ein und aus). Systemmerker verwenden Sie für folgende Abfragen: • Ist der aktuelle Zyklus der erste Zyklus nach NETZ-AUS/NETZ-EIN? • Haben sich Diagnosezustandsänderungen gegenüber dem vorangegangenen Zyklus ergeben? • Abfrage auf "1" (high) • Abfrage auf "0" (low) Taktmerker wechseln periodisch in festgelegen Zeitabständen ihren Wert. Siehe: Verwenden von Systemmerkern (Seite 1025) Siehe: Verwenden von Taktmerkern (Seite 1026)

Gruppe	Eigenschaften	Beschreibung
SIMATIC Memory Card	-	Aktivieren der Diagnose für die Lebensdauer der SIMATIC Memory Card.
Systemdiagnose	Allgemein	Systemdiagnose ist die Erfassung, Auswertung und Meldung eines Fehlers innerhalb des Automatisierungssystems. Beispiele für Fehler: • Fehler im CPU-Programm • Ausfall von Baugruppen • Drahtbruch zu Gebern und Aktoren Für die Ermittlung der Systemdiagnose stehen Ihnen vordefinierte Meldetexte zur Verfügung. Bei Bedarf können bestehende Meldetexte geändert und neue hinzugefügt werden. Die Einstellungen werden mit dem Projekt gespeichert und wirken sich erst nach dem Übersetzen und Laden der Hardware-Konfiguration in die beteiligten Komponenten aus. Für S7-1500 ist die Systemdiagnose automatisch aktiviert (nicht deaktivierbar).
PLC-Meldungen	Allgemein Zentrale Meldungsverwaltung in der PLC	Aktiviert die zentrale Meldungsverwaltung in der CPU. Eine Meldung wird in komplett in der CPU zusammengesetzt und dann an das HMI-Gerät gesendet. Vorteil: Ein Laden der Meldungstexte in die HMI-Geräte entfällt. Siehe: Zentrale Meldungsverwaltung in der CPU (Seite 1008)

Gruppe	Eigenschaften	Beschreibung
Webserver	Allgemein	Aktiviert und konfiguriert die Webserverfunktion. Siehe: AUTOHOTSPOT
	Automatische Aktualisierung	Schickt die angeforderte Webseite mit aktuellen CPU-Daten periodisch zum Webbrowser. Die Periodendauer geben Sie unter "Aktualisierungsintervall" ein. Die automatische Aktualisierung ist nur aktivierbar, wenn der Webserver aktiviert ist. Siehe: AUTOHOTSPOT
	Benutzerverwaltung	Hier werden Benutzernamen mit ihren Zugriffsrechten eintragen. • Jeder: Vordefinierter Benutzer, dessen Zugriffsrechte Sie über eine Klappliste vergeben. Der Benutzer "Jeder" ist ein Benutzer, der sich beim Web-Zugriff nicht mit Passwort anmeldet. • User#: Editierbarer Name für einen Benutzer, für den Sie Zugriffsrechte und jeweils ein Passwort vergeben. Sobald Sie in eine Zeile unterhalb von "Jeder" klicken, wird ein voreingestellter Name angezeigt, den Sie ändern können. Jedem Benutzer können Sie verschiedene Zugriffsrechte über eine Klappliste zuweisen. Es gibt Zugriffsrechte, die miteinander gekoppelt sind. Beispiel: Wenn Sie das Zugriffsrecht "Firmware-Update durchführen" aktivieren, dann werden automatisch die Zugriffsrechte "Betriebszustand ändern" und "... die Diagnose abfragen" aktiviert. Diese beiden Rechte sind notwendig, um eine Firmware-Update vom Webserver aus durchführen zu können. Siehe: AUTOHOTSPOT
	Security	Ermöglicht die Verwendung eines Zertifikates für die Identifizierung des Servers.
	Beobachtungstabellen	Ermöglicht das Anlegen von Beobachtungstabellen und legt den Zugriff darauf fest.
	Anwenderdefinierte Webseiten	Ermöglicht den Zugriff auf frei gestaltete Webseiten der CPU mit einem Webbrowser. Siehe: Anwenderseiten erstellen und laden (Seite 1330)
	Einstiegsseite	Ermöglicht die Auswahl der Einstiegsseite.
	Übersicht der Schnittstellen	Tabellarische Darstellung aller Baugruppen mit ihren Ethernet-Schnittstellen zur Webserver-Funktionalität für dieses Gerät. Hier können Sie für jede Ethernet-Schnittstelle des Geräts (CPU, CP, CM) den Zugriff auf den Webserver über die jeweilige Schnittstelle erlauben bzw. nicht erlauben.
DNS-Konfiguration	-	Projektierung der DNS-Server-Adresse. Ein DNS-Server kann erforderlich sein, wenn die CPU oder ein Kommunikationspartner über den Host-Namen (FQDN) erreichbar sein soll. Siehe: AUTOHOTSPOT

Gruppe	Eigenschaften	Beschreibung
Display	Allgemein	Im Standby-Betrieb bleibt das Display dunkel und wird wieder aktiviert, sobald eine Taste des Displays gedrückt wird. Den Standby-Betrieb können Sie auch im Display-Menü des Displays ändern. Im Energiesparmodus zeigt das Display die Information mit verminderter Helligkeit an. Sobald eine Taste des Displays gedrückt wird, wird der Energiesparbetrieb beendet. Den Energiesparbetrieb können Sie auch im Display-Menü des Displays ändern. Sprache Display: Unmittelbar nach dem Laden der Hardware-Konfiguration mit der eingestellten Standardsprache wird die Sprache umgestellt. Die Sprache können Sie auch im Display-Menü des Displays ändern.
	Automatische Aktualisierung	Eingabe des Zeitintervalls, in dem sich die Display-Anzeige aktualisiert.
	Passwort	Einstellung, ob Schreibzugriff über das Display der CPU ermöglicht oder verhindert wird. Wenn Sie den Schreibzugriff aktiviert haben, können Sie zusätzlich einen Displayschutz aktivieren. Das Display ist dann durch ein parametrierbares Passwort geschützt. Die Zeitdauer bis zum automatischen Abmelden nach Eingabe des Passworts am Display können Sie einstellen.
	Beobachtungstabellen	Projektierte Beobachtungstabellen werden zur Auswahl angeboten. Im Betrieb können Sie die ausgewählten Beobachtungstabellen am Diplay nutzen.
	Anwenderdefiniertes Logo	Sie können ein anwenderdefiniertes Logo auswählen und mit der Hardware-Konfiguration in die CPU laden.
Mehrsprachigkeit	-	Wenn Sie Übersetzungen von Projekttexten für den Webserver/für das CPU-Display nutzen wollen, müssen Sie hier den Sprachen für den Webserver/für das CPU-Display jeweils die gewünschte Projektsprache zuweisen (Anzahl ist CPU-abhängig). Sie legen damit die Projektsprachen fest, die beim Zugriff auf den Webserver bzw. im CPU-Display angezeigt werden. Beispiel: Sie weisen einer Sprache "Deutsch" des Webservers/des CPU-Displays die Projektsprache "Deutsch (Luxemburg)" durch Auswahl aus der Klappliste zu. Damit die Projektsprache in der Klappliste auswählbar ist, muss sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" aktiviert sein. Siehe: Mehrsprachigkeit (Seite 1002)
Uhrzeit	Ortszeit und Sommerzeit	Auswählen der Zeitzone, in der die CPU betrieben wird und Einstellen der Sommer-/Winterzeit. Siehe auch: Uhrzeit einer CPU ermitteln und einstellen (Seite 3508) Stellen und Lesen der Uhrzeit (Seite 1023)

Gruppe	Eigenschaften	Beschreibung
Schutz & Security	Zugriffsstufe	Einstellen des Lese-/Schreibschutzes und des Passworts für den Zugriff auf die CPU. Siehe: Einstellmöglichkeiten für den Schutz (Seite 1026) Siehe: Zugriffsstufen parametrieren (Seite 1031)
	Verbindungsmechanismen	Aktivieren des Zugriffes über eine PUT/GET-Kommunikation.
	Zertifikatsmanager	Aktivieren der globalen Security-Einstellungen für den Zertifikatsmanager und Verwalten der Zertifikate für das Gerät.
	Security-Ereignis	Aktivieren von Sammelmeldungen bei Security-Ereignissen und Festlegen eines Überwachungszeitraums (Intervall). Siehe: Security-Ereignisse zusammenfassen (Seite 1035)
OPC UA	Allgemein	Zeigt den Namen der OPC UA-Applikation an. Dieser wird für den OPC UA-Server auf der CPU verwendet.
	Server	Aktivieren und konfigurieren des OPC UA-Servers. Neben den allgemeinen Einstellungen für die Vergabe von Portnummern, Sendeintervall und Abtastintervall können Sie hier die Security-Einstellungen konfigurieren und den Export der im OPC UA-Adressraum verfügbaren PLC-/DB-Variablen in eine XML-Datei durchführen. Unter den Security-Einstellungen haben Sie die folgenden Möglichkeiten: • Secure-Channel: Auswahl zwischen der Nutzung der globalen Security-Einstellungen oder der Verwendung des lokalen CPU-spezifischen Zertifikatsmanager mit limitierter Funktionalität. Erzeugen oder wählen Sie hier das Server-Zertifikat, aktivieren Sie die geeignete Security Policy und verwalten Sie die Liste der vertrauenswürdigen Clients. • Benutzer-Authentifizierung: Anlegen oder Auswählen neuer Benutzer und Festlegen der Authentifizierungs-Optionen.
	Client	Aktivieren des OPC UA-Clients für diese CPU.
Systemstromversorgung	Allgemein	Gibt an, dass die CPU mit dem Stromnetz verbunden ist. Somit fließt diese Leistung der CPU positiv in die Leistungsbilanz ein.
	Leistungsbilanzierung	In der Leistungsbilanzierung wird die zur Verfügung gestellte Leistung durch die CPU und die Stromversorgungsmodule mit dem Leistungsbedarf der Signalmodule verglichen. Bei einer negativen Leistungsbilanz, also bei einem höheren Leistungsbedarf, als durch die CPU und die Stromversorgungsmodule zu Verfügung gestellt werden, wird dies in der Tabelle kenntlich gemacht.
Konfigurationssteuerung	-	Aktivieren der Konfigurationssteuerung. Damit wird in einem gewissen Rahmen eine Konfigurationsänderung im Anwenderprogramm ermöglicht. Siehe: Konfigurationssteuerung
Verbindungsressourcen	-	Bietet einen Überblick über die zugewiesenen reservierten und dynamischen Ressourcen für die Verbindungen der CPU. In der Online-Sicht werden auch die aktuell verwendeten Ressourcen angezeigt. Siehe: Verbindungsressourcen und Kommunikationsart (Seite 172)

Gruppe	Eigenschaften	Beschreibung
Adressübersicht	-	Tabellarische Darstellung aller von der CPU verwendeten Adressen für die integrierten Ein/Ausgänge sowie für die gesteckten Module. Adressen, die von keinem Modul verwendet werden, sind als Lücke dargestellt. Die Ansicht lässt sich filtern nach • Eingabeadressen • Ausgabeadressen • Adresslücken
Runtime-Lizenzen	CPU-Applikationen, die eine RT-Lizenz benötigen, z. B. OPC UA, ProDiag, Energy Suite	Je nach CPU-Applikation wählen Sie hier die Anzahl der verwendeten Lizenzen. Der Kauf dieser Lizenzen ist Voraussetzung für Ihre Verwendung. Ausnahme: Wenn Sie diese CPU-Applikationen im Kontext von Simulationen mit PLCSim Advanced nutzen, dann ist der Kauf der Lizenzen nicht erforderlich. Sie wählen in diesem Fall einfach die benötigte Anzahl Lizenzen aus. Beispiel für Lizenzauswahl: ProDiag-Lizenzen einstellen

Siehe auch

Gerätename und IP-Adresse zuweisen (Seite 2990)

CPU-Einstellungen am Display (Seite 998)

Regeln für CPUs mit mehreren PROFINET-Schnittstellen (Seite 1010)

IEEE-konformes LLDP (Seite 1014)

CPU-Einstellungen am Display

Die S7-1500-CPUs haben ein Display mit Bedientastern. Das Display kann zum Auslesen von Statusinformationen zu den konfigurierten Modulen, zur Diagnose, aber auch zum Anpassen einiger CPU-Einstellungen verwendet werden.

Beispiel: Einstellen der IP-Adresse und des PROFINET-Gerätenamens am Display

Wenn Sie die IP-Adressparameter (IP-Adresse, Subnetzmaske sowie Router-Einstellungen) am Display vergeben wollen, dann müssen Sie im entsprechenden Projekt für die PROFINET-Schnittstelle der CPU folgende Option wählen:

- IP-Adresse am Gerät einstellen

Dasselbe gilt sinngemäß für den PROFINET-Gerätenamen: Wenn Sie den PROFINET-Gerätenamen am Display vergeben wollen, dann müssen Sie im entsprechenden Projekt für die PROFINET-Schnittstelle der CPU folgende Option wählen:

- PROFINET-Gerätenamen am Gerät einstellen

Zykluszeit und maximale Zykluszeit (Zyklusüberwachungszeit)

Funktion

Die Zykluszeit ist die Zeit, die das Betriebssystem für die zyklische Programmbearbeitung sowie für alle diesen Zyklus unterbrechende Programmteile benötigt. Die Programmbearbeitung kann unterbrochen werden durch:

- Zeitfehler und 2xMaxCycleTime-Fehler
- Systemtätigkeiten, z. B. Prozessabbildaktualisierung

Die Zykluszeit (TZ) ist daher nicht für jeden Zyklus gleich lang.

Das folgende Bild zeigt ein Beispiel für unterschiedliche Zykluszeiten (TZ1 ≠ TZ2) für S7-1500-CPUs:

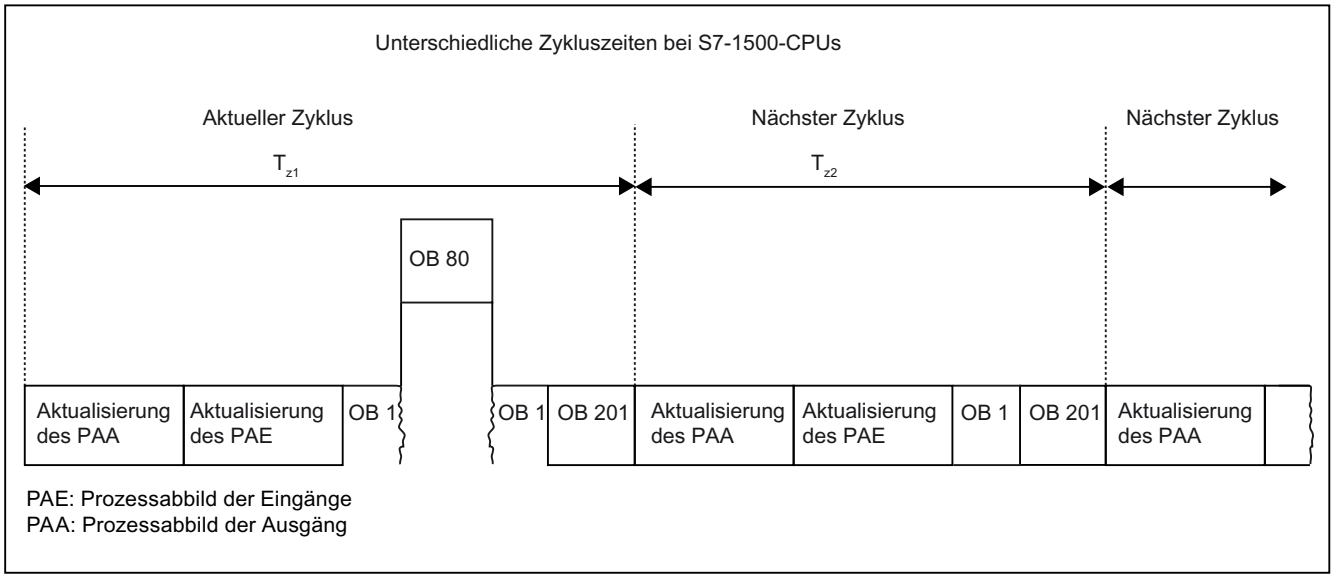

Im aktuellen Zyklus wird der hier verwendete zyklische OB (z. B. OB 1) durch einen Zeitfehler (z. B. OB 80) unterbrochen. Im Anschluss des zyklischen OBs wird der nächste Zyklus-OB 201 abgearbeitet.

Maximale Zykluszeit (Zyklusüberwachungszeit)

Das Betriebssystem überwacht die Laufzeit des zyklischen Programms auf eine projektierbare Obergrenze, die maximale Zykluszeit. Sie können durch Aufruf der Anweisung RE_TRIGR an jeder Stelle Ihres Programms diese Zeitüberwachung neu starten.

Wenn das zyklische Programm die maximale Zykluszeit überschreitet, versucht das Betriebssystem den Zeitfehler-OB (OB 80) zu starten. Ist der OB nicht vorhanden, geht die CPU in den Betriebszustand STOP.

Neben der Überwachung der Laufzeit auf Überschreitung der Zyklusüberwachungszeit wird die Einhaltung einer minimalen Zykluszeit garantiert. Das Betriebssystem verzögert dabei den Beginn eines neuen Zyklus so lange, bis die minimale Zykluszeit erreicht ist. Während dieser Wartezeit werden neue Ereignisse und Betriebssystemdienste bearbeitet.

Wenn die Zyklusüberwachungszeit ein zweites Mal überschritten wird, z. B. während der Zeitfehler-OB bearbeitet wird (2xMaxCycleTime-Fehler), geht die CPU in den Betriebszustand STOP.

Zyklusbelastung durch Kommunikation

Funktion

Die Zykluszeit der CPU kann sich durch Kommunikationsprozesse verlängern. Zu diesen Kommunikationsprozessen gehören z. B.:

- Datenübertragung zu einer anderen CPU
- Über ein Programmiergerät (PG) angestoßenes Laden von Bausteinen
- Bestimmte Motion Control-Funktionen

Die Dauer dieser Kommunikationsprozesse können Sie mit dem CPU-Parameter "Zyklusbelastung durch Kommunikation" in einem gewissen Rahmen steuern.

Neben den Kommunikationsprozessen verlängern auch Testfunktionen die Zykluszeit. Dabei kann über den Parameter "Zyklusbelastung durch Kommunikation" Einfluss auf die Dauer genommen werden.

Die Motion Control-Funktionen Inbetriebnahme von Technologieobjekten, Restart von Technologieobjekten und Interpolation von Kurvenscheiben werden ebenfalls als Kommunikationsprozesse bearbeitet. Solange diese Funktionen bearbeitet werden, verlängern sie als Kommunikationslast die Zykluszeit.

Wirkungsweise des Parameters

Über den Parameter "Zyklusbelastung durch Kommunikation" geben Sie den Prozentsatz der gesamten CPU-Verarbeitungsleistung an, der den Kommunikationsprozessen zur Verfügung stehen soll.

Wenn die Kommunikation diese Verarbeitungsleistung nicht benötigt, steht sie der Programmbearbeitung zur Verfügung.

Hinweis

Welche Auswirkungen kann die Vorgabe einer minimalen Zykluszeit haben?

Wenn Sie eine minimale Zykluszeit vorgegeben haben und das Anwenderprogramm diese nicht vollständig benötigt, kann die restliche zur Verfügung stehende Zeit erneut für Kommunikationsprozesse benutzt werden. Dies kann dazu führen, dass Kommunikationsprozesse einen größeren Teil der CPU-Verarbeitungsleistung beanspruchen, als Sie durch den CPU-Parameter "Zyklusbelastung durch Kommunikation" vorgegeben haben.

Für die Prioritätsklassen **kleiner oder gleich 15** wird die Verarbeitungsleistung der Kommunikation ständig von der CPU zugewiesen.

Prioritätsklassen **größer als 15** werden nicht durch Kommunikation unterbrochen.

Sie beeinflussen also durch die Zuweisung der Priorität eines Ereignisses bzw. eines OBs die Unterbrechbarkeit des OBs und der Bausteine, die von dem OB aufgerufen werden. Wenn auf diese Weise größerer Programmabschnitte nicht unterbrechbar gemacht werden, um die Zykluszeiten zu minimieren, verzögern sich Online-Funktionen von STEP 7!

Auswirkung auf die tatsächliche Zykluszeit

Durch die Verwendung des Parameters "Zyklusbelastung durch Kommunikation" verlängert sich die Zykluszeit des zyklischen Organisationsbausteins (z. B. OB 1) um einen Faktor, der sich nach der folgenden Formel berechnen lässt:

$$\frac{100}{100 - \text{“Zyklusbelastung durch Kommunikation“}}$$

Die Formel berücksichtigt dabei den Einfluss von asynchronen Ereignissen wie Prozess- oder Weckalarmen auf die Zykluszeit nicht.

Wenn sich die Zykluszeit durch Kommunikationsprozesse verlängert, können auch mehr asynchrone Ereignisse innerhalb der Zykluszeit des zyklischen Organisationsbausteins auftreten. Dies verlängert den Zyklus zusätzlich. Die Verlängerung ist davon abhängig, wie viele Ereignisse auftreten und wie lange ihre Bearbeitung dauert.

Beispiel 1 – keine zusätzlichen asynchronen Ereignisse:

Bei Einstellung des Parameters "Zyklusbelastung durch Kommunikation" auf 50 % kann die Zykluszeit des zyklischen Organisationsbausteins auf das Doppelte ansteigen.

Beispiel 2 – zusätzliche asynchrone Ereignisse:

Bei einer reinen Zykluszeit von 500 ms kann sich durch eine Kommunikationslast von 50 % eine tatsächliche Zykluszeit von bis zu 1000 ms ergeben, unter der Voraussetzung, dass die CPU immer genügend Kommunikationsaufträge zum Bearbeiten hat.

Wenn nun parallel dazu alle 100 ms ein Weckalarm (Prioritätsklasse kleiner oder gleich 15) mit 20 ms Bearbeitungszeit abläuft, dann würde sich dieser ohne Kommunikationslast mit insgesamt 5*20 ms = 100 ms auf den Zyklus verlängernd auswirken, d. h. die tatsächliche Zykluszeit wäre 600 ms.

Da die Kommunikation bei 50 % Kommunikationslast sowohl auf das zyklische Programm als auch auf den Weckalarm verlängernd auswirkt, beträgt die tatsächliche Zykluszeit nicht 1000 ms, sondern 1200 ms.

Wenn der Weckalarm eine Priorität größer als 15 hat, verlängert sich zwar nicht die Bearbeitungszeit des Weckalarm-OBs durch Kommunikation, aber die tatsächliche Zykluszeit bleibt gleich (1200 ms).

Hinweis

Beachten Sie folgende Hinweise:

- Überprüfen Sie die Auswirkungen einer Wertänderung des Parameters "Zyklusbelastung durch Kommunikation" im Anlagenbetrieb.
- Die Kommunikationslast muss beim Einstellen der minimalen Zykluszeit berücksichtigt werden, da es sonst zu Zeitfehlern kommen kann.

Empfehlungen

- Vergrößern Sie den Wert nur dann, wenn die CPU hauptsächlich zu Kommunikationszwecken eingesetzt wird und das Anwenderprogramm zeitunkritisch ist.
- In allen anderen Fällen sollten Sie den Wert nur verringern.
- Wenn Alarm-OBs nicht durch Kommunikation unterbrochen werden sollen, dann setzen Sie deren Priorität hoch auf einen Wert größer 15.

Mehrsprachigkeit

Wenn Sie eine CPU projektieren, entstehen Texte unterschiedlicher Kategorien, z. B.

- Objektnamen (Namen von Bausteinen, Modulen, Variablen, ...)
- Kommentare (für Bausteine, Netzwerke, Beobachtungstabellen, ...)
- Meldungstexte und Diagnosetexte

Texte werden vom System zur Verfügung gestellt (z. B. Diagnosepuffer-Texte) oder während der Projektierung erstellt (z. B. Meldungen).

Je nach konkretem Element liegen Texte einsprachig vor oder sie lassen sich in andere Spachen übersetzen und können dann mehrsprachig im Projekt vorliegen. Sie können Projekttexte in allen Sprachen pflegen, die Ihnen zur Auswahl in der Projektnavigation (Sprachen & Ressourcen > Projekttexte) zur Verfügung stehen.

Siehe: Grundlagen zu Projekttexten.

Die während der Projektierung entstehenden Texte lassen sich auf die CPU laden. Aufgrund begrenzter Ressourcen (z. B. Ladespeicher) und ist nur eine begrenzte Auswahl an Sprachen ladbar.

Die folgende Beschreibung zeigt, welchen Nutzen mehrsprachige Texte auf der CPU haben, welche Texte geladen werden und wie Sie bestimmen, in welchen Sprachen die Texte auf die CPU geladen werden.

Außerdem erfahren Sie, welchen Einfluss die Sprachzuordnung hat, die Sie in den CPU-Eigenschaften (Mehrsprachigkeit) parametrieren und wie sich diese Einstellung im Betrieb auswirkt.

Nutzen mehrsprachiger Projekttexte

Als Projekt-Ersteller fassen Sie Texte wie z. B. Kommentare und Meldungen in Ihrer Muttersprache ab. Für einen Erstausrüster (OEM-Kunde) übersetzen Sie die Texte in die Sprache des OEM-Kunden, z. B. Englisch sowie zusätzlich in der Sprache des Endkunden. Sie laden die maximal drei Sprachen in die CPU.

Die Kunden haben den Vorteil, dass die jeweiligen Texte, z. B. Kommentare und Meldungen, in ihrer jeweiligen Muttersprache vorliegen, auch wenn das STEP 7-Projekt nicht auf dem PG/PC verfügbar ist! Ein Laden des Geräts als neue Station lädt die Sprachen ebenfalls mit und das Projekt kann vor Ort analysiert und angepasst werden.

Ein weiterer Vorteil ist, dass bestimmte mehrsprachigen Texte auch ohne STEP 7 einem Bediener zur Verfügung stehen:

Über die Sprachauswahl des Webservers oder über die Sprachauswahl des CPU-Displays sind z. B. Meldungen, Diagnosepuffer-Einträge und Statusmeldungen für den Baugruppenzustand in der eingestellten Sprache sichtbar.

Eine Anzeige mehrsprachiger Texte an HMI-Geräten ist ebenfalls möglich,. Im Graph- bzw. Codeviewer können Sie z. B. Variablenkommentare und Schrittkommentare in den geladenen Sprachen anzeigen lassen.

Voraussetzungen

- Die zuzuweisenden Projektsprachen müssen aktiviert sein
- Die entsprechenden Texte (Übersetzungen) müssen im Projekt vorliegen.
  Die Projektsprachen-Auswahl finden Sie in der Projektnavigation unter "Sprachen & Ressourcen".

Verfügbare Sprachen für CPU-Display und Webserver einstellen

Die Einstellungen zur Mehrsprachigkeit finden Sie unter den CPU-Parametern: Hier stellen Sie folgendes ein:

- Welche Projektsprachen sind den Sprachen des CPU-Displays/des Webservers zugeordnet
- Welche Projektsprachen werden überhaupt in die CPU geladen

Gehen Sie folgendermaßen vor:

1. Navigieren Sie bei markierter CPU im Inspektorfenster zum Bereich "Mehrsprachigkeit" ("Eigenschaften > Allgemein > Mehrsprachigkeit").
   In diesem Bereich befindet sich eine Tabelle, in der Sie Sprachen zuweisen: In der linken Spalte sind die Projektsprachen auswählbar. In der rechten Spalte sind alle Sprachen aufgeführt, die am CPU-Display bzw. für den Webserver auswählbar sind.
2. Weisen Sie den Sprachen für CPU-Display/Webserver jeweils eine Projektsprache zu. Durch diese Zuweisungen werden für Sprachen, die nicht in der CPU vorhanden sind, vorhandene Projektsprachen (als Ersatzsprachen) angezeigt.
   Beachten Sie, dass sich diese Spracheinstellungen nur auf die Texte bezieht, die aus der Projektierung geladen werden. Sowohl der Webserver als auch das CPU-Display verfügen über weit mehr Oberflächensprachen, so dass Sie die statischen Texte im Webbrowser bzw. auf dem CPU-Display in sehr viel mehr Sprachen anzeigen lassen können.

Projektsprachen	Sprachen des CPU-Displays/des Webservers
Deutsch (Deutschland)	Deutsch
Deutsch (Deutschland)	Englisch
Deutsch (Deutschland)	Französisch
Deutsch (Deutschland)	Spanisch
Deutsch (Deutschland)	Italienisch
Deutsch (Deutschland)	Japanisch
Deutsch (Deutschland)	Chinesisch (vereinfacht)
Deutsch (Deutschland)	Koreanisch
Deutsch (Deutschland)	Russisch
Deutsch (Deutschland)	Türkisch
Deutsch (Deutschland)	Portugiesisch (Brasilien)

Hinweis

Die Anzahl ladbarer Sprachen ist CPU-spezifisch begrenzt. Wenn z. B. zwei Sprachen möglich sind, weisen Sie in der Spalte "Sprachen des CPU-Displays/des Webservers" Ihrer Muttersprache die entsprechende Projektsprache zu. In den übrigen Zeilen weisen Sie den Sprachen des CPU-Displays/des Webservers jeweils dieselbe Ersatzsprache zu.

Beispiel

Vorhandene Projektsprachen in STEP 7 (PG): Englisch, Französisch, Deutsch, Italienisch.

Aufgrund der Zuweisung im Bereich "Mehrsprachigkeit" der CPU-Parameter werden folgende Projektsprachen beim nächsten Ladevorgang geladen:

Deutsch, Englisch.

Die Projektsprache Deutsch ist automatisch eingestellt bei Display-Sprache Deutsch. Die Projektsprache Englisch wird bei allen anderen europäischen Display-Sprachen angezeigt.

Den übrigen Display-Sprachen ist keine Sprache zugeordnet.

Am Display kann man nur die Sprache für das Display wählen; aufgrund der Sprachzuordnung aus der Projektierung ist mit der Sprache für das Display automatisch die zugewiesene Projektsprache eingestellt:

- Wählt man englisch als Sprache für das Display, ist die Projektsprache englisch eingestellt.
- Wählt man französisch als Sprache für das Display, ist ebenfalls die Projektsprache englisch eingestellt.
- Wählt man chinesisch als Sprache für das Display, werden keine Projekttexte angezeigt.

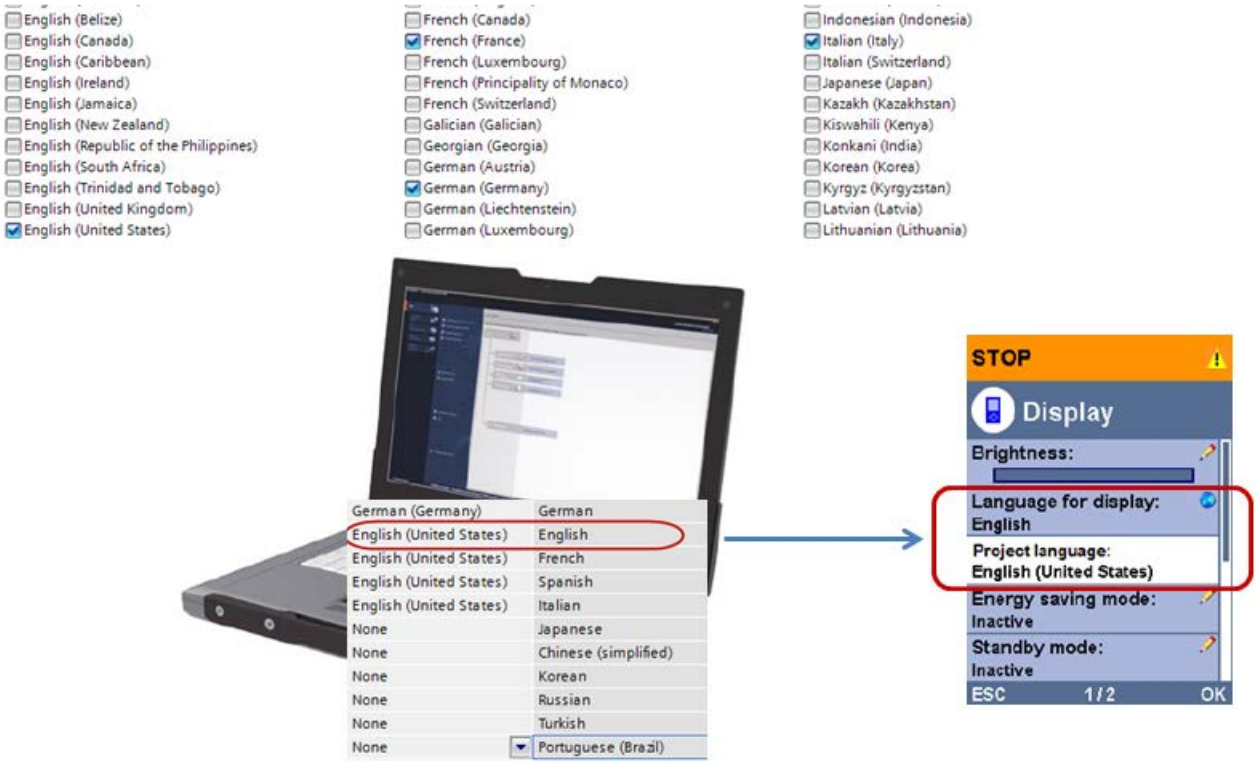

Welche Projekttexte werden mit den gewählten Sprachen geladen?

Folgende Texte werden in den gewählten Sprachen mit den Projektdaten in die CPU geladen und auch vom Webserver/CPU-Display genutzt:

- Diagnosepuffer-Texte (nicht änderbar)
- Statustexte für den Baugruppenzustand (nicht änderbar)
- Meldetexte mit zugehörigen Textlisten
- Variablenkommentare und Schrittkommentare für Graph und PLC Code Viewer
- Kommentare in Beobachtungstabellen

Folgende Texte werden ebenfalls in den gewählten Sprachen mit den Projektsprachen in die CPU geladen, werden aber nicht vom Webserver/CPU-Display genutzt:

- Kommentare in Variablentabellen (für Variablen und Konstanten)
- Kommentare in globalen Datenbausteinen

- Kommentare von Elementen in Bausteinschnittstellen von FBs, FCs, DBs und UDTs
- Netzwerktitel in Bausteinen, die in KOP, FUP oder AWL geschrieben sind
- Bausteinkommentare
- Netzwerkkommentare
- Kommentare von KOP- und FUP-Elementen

Das folgende Bild zeigt noch einmal den Zusammenhang zwischen mehrsprachigen Projekttexten und benötigtem Speicherplatz auf einer SIMATIC Memory Card.

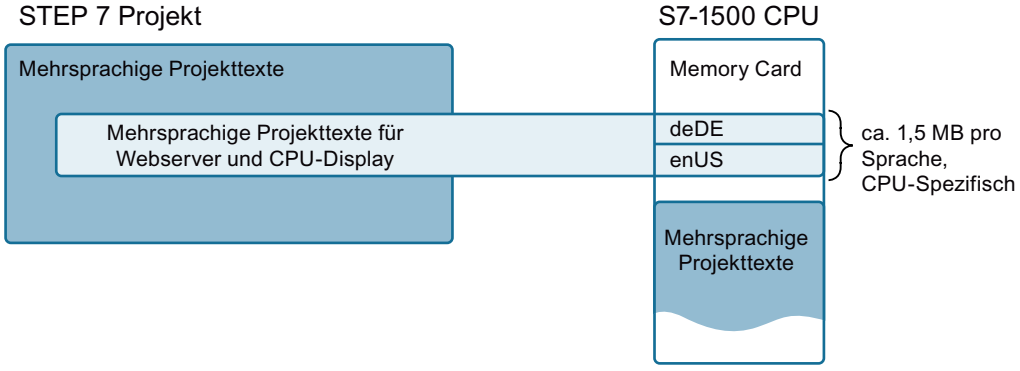

Besonderheiten: Modulkommentare

Modulkommentare werden nur in einer Sprache geladen.

Für das Laden von Baugruppen- bzw. Modulkommentaren gilt:

- Voraussetzung: In den Projekteinstellungen (Extras > Einstellungen, Bereich "Hardware-Konfiguration") muss die Option "Baugruppenkommentar herunterladen" aktiviert sein (Voreinstellung).
- Modulkommentare werden in der Sprache geladen, die als Editiersprache eingestellt ist. Die Editiersprachen ändern Sie unter "Sprachen & Ressourcen > Projektsprachen".

Besonderheiten: Diagnosepuffer-Texte und Statustexte für Baugruppenzustand

Die vordefinierten Texte für Diagnosepuffer-Einträge stehen nicht in allen Sprachen zur Verfügung, die Sie in STEP 7 bei den Sprachen & Ressourcen als Projektsprachen auswählen können. Bei der Zuordnung zwischen Projektsprache und Sprache für Webserver/CPU-Display wird deshalb die "nächstliegende" Sprache in die CPU geladen.

Beispiel: Die oben genannten Texte liegen u. a. in der Spache Deutsch vor; es gibt nur eine deutsche Sprachvariante. Wenn als Projektsprache Deutsch (Luxemburg) oder Deutsch (Österreich) ausgewählt ist, wird immer dieselbe Sprache Deutsch geladen.

Hinzufügen oder Wegnehmen von Sprachen - Folgen für das Laden in die CPU

Wenn sich die Anzahl der Projektsprachen für das CPU-Display/den Webserver geändert hat, ist ein Laden der Betriebszustand STOP erforderlich!

STEP 7 führt in diesem Fall ein komplettes Laden durch (HW und SW).

Dies gilt unabhängig davon, welche Objekte Sie für das Laden gewählt haben: Auch ein Baustein-Laden, normalerweise im RUN durchführbar, erfordert immer dann einen CPU-STOP, wenn sich die Anzahl der Sprachen für das CPU-Display/den Webserver geändert hat seit dem letzten Laden.

Auch wenn Sie eine hinzugefügte Projektsprache wieder abwählen, ist ein CPU-STOP für das Laden erforderlich. Nur wenn Sie über den Befehl "Rückgängig" im Menü "Bearbeiten" die Auswahl einer zusätzlichen Projektsprache rückgängig machen und sonst nichts geändert haben, was einen STOP erforderlich macht, können Sie die CPU im RUN laden.

Grundsätzlich gilt:

STEP 7 prüft vor jedem Laden, ob sich die Anzahl der Sprachen für das CPU-Display/den Webserver geändert hat. Wenn sich die Anzahl geändert hat (weniger oder mehr Sprachen gewählt seit dem letzten Laden), dann ist ein CPU-STOP erforderlich.

Besonderheiten beim Laden des Geräts als neue Station (Hardware und Software)

Wenn Sie eine Online-Verbindung zu einem Gerät hergestellt haben und dieses Gerät (CPU mit allen zugeordneten Modulen) in Ihr Projekt laden, dann gelten folgende Regeln:

- Die Projektsprachen im Projekt werden aktualisiert, d. h. wenn im Gerät zusätzliche Projektsprachen vorhanden sind, sind sie nach dem Laden auch im Projekt vorhanden (unter "Sprachen & Ressourcen" in der Projektnavigation aktiviert). Die "Offline- und Online-Texte" sind nach dem Laden abgeglichen, d. h. der Stand der Texte im Projekt entspricht dem Stand im Gerät.
- Die im Projekt bereits vorhandenen Projektsprachen bleiben vom Ladevorgang unberührt; es wird keine Projektsprache gelöscht.
- Die Zuordnungen "Projektsprache - Sprachen des CPU-Displays/des Webservers" (Mehrsprachigkeit) im Projekt entspricht nach dem Laden den Zuordnungen, die in die CPU geladen wurden. Ein erneutes Laden ins Gerät lädt dieselben Sprachen, die in der CPU vorhanden sind.

Besonderheiten beim Laden vom Gerät (Software)

Wenn Sie Software aus einer CPU in Ihr Projekt laden, z. B. einzelne Bausteine oder das komplette Programm, gelten folgende Regeln:

- Die Projektsprachen im Projekt werden aktualisiert, d. h. wenn im Gerät zusätzliche Projektsprachen vorhanden sind, sind sie nach dem Laden auch im Projekt vorhanden (unter "Sprachen & Ressourcen" in der Projektnavigation aktiviert). Die "Offline- und Online-Texte" der geladenen Komponenten sind nach dem Laden abgeglichen, d. h. der Stand der Texte im Projekt entspricht dem Stand im Gerät.
- Die Zuordnungen "Projektsprache - Sprachen des CPU-Displays/des Webservers" (Mehrsprachigkeit) im Projekt **bleiben erhalten**, d. h. sie ändern sich nicht, auch wenn z. B. im geladenen Baustein eine neue Projektsprache verwendet wurde!
  Beachten Sie, dass Sie in diesem Fall (neue Sprache nur in der online erreichbaren CPU vorhanden) die Einstellungen unter "Mehrsprachigkeit" in den Eigenschaften der CPU anpassen müssen, falls die neue Sprache auch wieder in die CPU geladen werden soll!

- Es besteht die Gefahr, dass bereits vorhandene Übersetzungen im Projekt beim Laden vom Gerät **gelöscht** werden, und zwar bei folgender Situation:
  - Eine Sprache ist offline (z. B. in einem Baustein im Projekt) vorhanden
  - Diese Sprache ist online **nicht** vorhanden
    In diesem Fall verlieren Sie die beim Laden vom Gerät die online vorhandene Sprache!
- Team Engineering: Die Einstellungen für Mehrsprachigkeit sollten Sie nur im Master-Projekt ändern.

Besonderheiten beim Team Engineering (gemeinsame Inbetriebnahme)

Im Fall einer gemeinsamen Inbetriebnahme haben Sie die Möglichkeit, fehlende Übersetzungen durch den Synchronisationsdialog zu erkennen.

Fehlende Sprachen beim Online/Offline-Vergleich erkennen

Im Online/Offline-Vergleich (Vergleichseditor) können Sie Unterschiede bei den vorhandenen Projektsprachen für folgende Objekte erkennen:

- Bausteine
- Variablentabellen

Unterschiede bei den vorhandenen Projektsprachen für Meldungen und Textlisten werden nicht vom Vergleichseditor berücksichtigt.

Falls beim Vergleich auch Know-how-geschützte Bausteine dabei sind: Die Eingabe von Passwörtern für diese Bausteine ist für den Vergleich nicht erforderlich.

Für den Vergleich werden herangezogen:

- Online: die Sprachen, die tatsächlich in der CPU vorhanden sind
- Offline: die Sprachen, die in der Tabelle "zugewiesene Projektsprachen" im Bereich "Mehrsprachigkeit" gewählt sind und damit auch beim Laden in die CPU berücksichtigt werden. Es sind nicht die Sprachen, in denen z. B. die Kommentare der Bausteine tatsächlich vorliegen!

Der Offline/Offline-Vergleich vergleicht die jeweils aktivierten Projektsprachen.

Siehe auch

Übersicht über die CPU-Eigenschaften (Seite 988)

Zentrale Meldungsverwaltung in CPU

Die S7-1500-CPUs ab Firmware Version V2.0 unterstützen die zentrale Meldungsverwaltung.

Sie finden den CPU-Parameter "Zentrale Meldungsverwaltung in der PLC" unter "PLC-Meldungen > Allgemein".

Voreinstellung: Die zentrale Meldungsverwaltung ist aktiviert.

Wirkung des CPU-Parameters "Zentrale Meldungsverwaltung"

Wenn die Zentrale Meldungsverwaltung aktiviert ist, überträgt die CPU den gesamten Meldetext, der aus mehreren Textlisten-Einträgen bestehen kann, zum HMI-Gerät. Die CPU übernimmt dabei die Zusammenstellung des kompletten Meldungstexts. Sobald das triggernde Ereignis für eine Meldung ansteht, sendet die CPU den gesamten Meldetext mit eventuell vorhandenen Begleitwerten über die projektierte HMI-Verbindung zum HMI-Gerät.

Die aktivierte zentrale Meldungsverwaltung erspart das Laden der Texte der angeschlossenen HMI-Geräte.

Wenn die Zentrale Meldungsverwaltung **nicht** aktiviert ist, müssen Sie die Texte mit dem Engineering System (WinCC) auf das HMI-Gerät laden, damit die Anzeige der Meldungstexte im Betrieb funktioniert. Wenn das triggernde Ereignis für eine Meldung ansteht, sendet die CPU nur rudimentäre Informationen zum HMI-Gerät (z. B. Meldungs-ID, Version der Meldung und Begleitwerte). Die Meldungsverwaltung übernimmt das HMI-Gerät: Es stellt den kompletten Meldungstext aus den geladenen und aus den übermittelten Daten zusammen und bereitet ihn auf. Verarbeitungsleistung der CPU wird für diese Aufgabe nicht benötigt.

Jede Änderung an den Meldungstexten erfordert ein erneutes Laden in die angeschlossenen HMI-Geräte.

Hinweise und Regeln für die zentrale Meldungsverwaltung

- Vorausgesetzte Version der HMI-Runtime: ab V14
- Vorausgesetzte CPU: S7-1500 ab Firmware Version 2.0 (auch S7-1500SP CPU, S7-1500Pro CPU und S7-1500 Software Controller).
- Meldungen, für die zentrale Meldungsverwaltung möglich ist: Controller Meldungen (Systemdiagnose, Prozessdiagnose, Programmeldungen).
- In den Runtime-Einstellungen des HMI-Geräts (Abschnitt "Steuerungsmeldungen" im Bereich "Meldungen") muss die "Automatische Aktualisierung" aktiviert sein.
- Geänderte Texte, die Sie in die CPU laden, haben keine Auswirkungen auf Meldungstexte, die bereits an das HMI-Gerät übertragen wurden. Der aktualisierte Meldungstext wird erst bei einem neuen triggernden Ereignis angezeigt.
- Für Meldungen, die beim HMI-Gerät in der Meldungssicht oder Meldungspuffer angezeigt werden, gilt Folgendes:
  - Die Sprachen, in der die Meldungen am HMI-Gerät angezeigt werden sollen, müssen Sie in die CPU laden (max. 3 Sprachen, siehe Mehrsprachigkeit (Seite 1002)).
  - Die Sprachen am HMI-Gerät müssen die Sprachen sein, die auch in die CPU geladen wurden. Sonst zeigt das HMI-Gerät im Betrieb nur an, dass Meldungstexte fehlen. Bei mehr als 3 Sprachen müssen Sie das HMI-Gerät mit den Texten laden (keine zentrale Meldungsverwaltung möglich).
- Für Meldungen, die im Alarmarchiv angezeigt werden, gilt Folgendes:
  - Das Alarmarchiv enthält Meldungen nur in einer Sprache. Diese Sprache wird bei der Projektierung des HMI-Geräts eingestellt unter "Archive > Allgemein > Archivsprache". Auch die Archivsprache muss dann eine der Sprachen sein, die in die CPU geladen wurde.

Hinweis

Anzeige von Fehlern

Bei Fehlern wird ein Text wie folgt angezeigt : #TextlistID, TextID (z. B. #123, 456)

Beispiel: Sie haben bei der verwendeten Textliste einen Eintrag (Range) vergessen. Dann ist erste Zahl die Textlisten-ID (Spalte ist in der Voreinstellung ausgegraut, kann aber sichtbar gemacht werden). Die zweite Zahl zeigt den Wert, für den der Eintrag fehlt.Textlisten-ID = 123, Wert = 456.

Wird Range ohne Text definiert, dann erzeugt das keinen Fehler.

Siehe auch

Übersicht über die CPU-Eigenschaften (Seite 988)

CPUs mit mehreren PROFINET-Schnittstellen

Regeln für CPUs mit mehreren PROFINET-Schnittstellen

Innerhalb der CPU-Familie der S7-1500-CPUs gibt es CPUs mit zwei bzw. drei PROFINET-Schnittstellen (Interfaces), z. B. CPU 1516-3 PN/DP.

Eine PROFINET-Schnittstelle kann für PROFINET IO-Kommunikation verwendet werden, während die anderen z. B. für die Kommunikation mit einem übergeordneten Netz (Backbone, Router, Internet) oder für die Kommunikation mit einer anderen Maschine bzw. mit einer anderen Automatisierungszelle verwendet werden.

Beachten Sie, dass einige Einstellungen nur an der Schnittstelle X1 geändert werden können und an den anderen Schnittstellen (z. B. X2) nur lesbar sind, siehe unten.

CPUs ab Firmware V2.0 mit mehreren PN IO-Schnittstellen

Bei Zentralbaugruppen ab CPU 1515 mit Firmware-Version ab V2.0 bieten beide PROFINET-Schnittstellen IO-Funktionalität:

- Sie haben die Möglichkeit, zwei voneinander unabhängige PROFINET IO-Systeme zu betreiben, da jede der PN IO-Schnittstellen die Funktion IO-Controller unterstützt.
- Die PROFINET-Schnittstelle X1 unterstützt die Echtzeitklassen IRT und RT, die PROFINET-Schnittstelle X2 die Echtzeitklasse RT.

- Beide PROFINET-Schnittstellen unterstützen die Funktion I-Device.
- Die 2. bzw. 3. PROFINET-Schnittstelle hat gegenüber der 1. PROFINET-Schnittstelle (X1) Einschränkungen:
  - Aufbau redundanter Ringstrukturen nicht möglich
  - IRT nicht möglich (RT an der PROFINET-Schnittstelle X2 möglich, aber Auswahl der Sendetakte und Anzahl anschließbarer IO-Devices eingeschränkt)

  Details zur Leistungsfähigkeit der einzelnen Schnittstellen sind in den Gerätehandbüchern zu den jeweiligen CPUs beschrieben.

Im Folgenden ist beispielhaft dargestellt, wie zwei CPUs ihr jeweiliges Maschinennetz an der PROFINET-Schnittstelle X1 betreiben. Der Betrieb der Maschinennetze funktioniert auch mit Isochronem Realtime (IRT).

Die PROFINET-Schnittstellen X2 sind jeweils zu Koordinierungszwecken an einem Anlagennetz angeschlossen, das vom lokalen Maschinennetz unabhängig ist: Mehrere "Sub_Machines" sind über die I-Device-Funktionalität an eine "Main_Machine" gekoppelt.

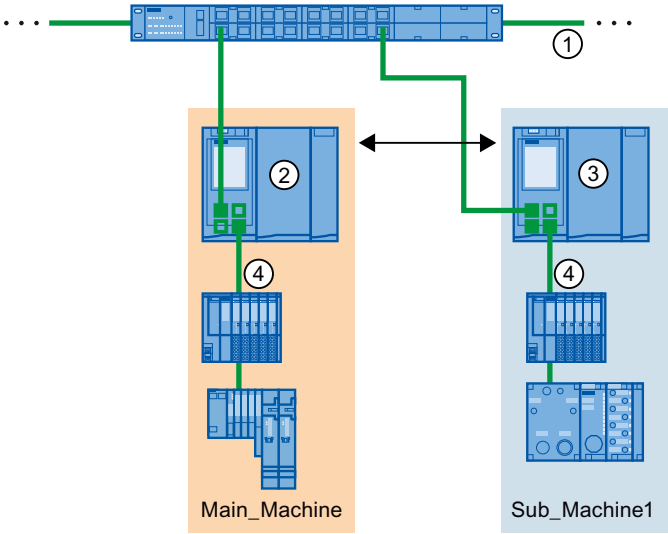

① Anlagennetz

② Die CPU ist IO-Controller am Anlagennetz (Schnittstelle X2) und gleichzeitig IO-Controller am Maschinennetz (Schnittstelle X1). Das Maschinennetz können Sie mit IRT betreiben.

③ Die CPU ist I-Device am Anlagennetz (Schnittstelle X2) und gleichzeitig IO-Controller am Maschinennetz (Schnittstelle X1).

④ Maschinennetz

Regeln

Für eine S7-1500-CPU mit mehreren PROFINET-Schnittstellen gilt:

- Bei CPUs mit mehreren PROFINET-Schnittstellen **ab Firmware-Version V2.0** unterstützt die zweite PROFINET-Schnittstelle (X2) ebenfalls IO-Funktionalität mit der Echtzeitklasse RT, also die Rollen IO-Controller und IO-Device.
  Außerdem unterstützt die zweite bzw. alle weiteren PROFINET-Schnittstellen (z. B. X2) HMI-Kommunikation, die Kommunikation mit dem Projektierungssystem, die Kommunikation mit einem übergeordneten Netz (Backbone, Router, Internet) und die Kommunikation mit einer anderen Maschine bzw. Automatisierungszelle. Außerdem wird unterstützt: Topologie senden über LLDP-Protokoll, LLDP-MIBs verwalten, SNMP-Anfragen beantworten und verwalten sowie die Funktion "Erreichbare Teilnehmer" (DCP-Protokoll).
- Bei CPUs mit einer **Firmware-Version < V2.0** gilt: Die zweite bzw. alle weiteren PROFINET-Schnittstellen (z. B. X2) unterstützen keine IO-Funktionalität, also keine IO-Controller/IO-Device-Rolle. Unterstützt wird die HMI-Kommunikation, die Kommunikation mit dem Projektierungssystem, die Kommunikation mit einem übergeordneten Netz (Backbone, Router, Internet) und die Kommunikation mit einer anderen Maschine bzw. Automatisierungszelle. Außerdem wird unterstützt: Topologie senden über LLDP-Protokoll, LLDP-MIBs verwalten, SNMP-Anfragen beantworten und verwalten sowie die Funktion "Erreichbare Teilnehmer" (DCP-Protokoll).
- Die IP-Subnetze aller Schnittstellen müssen unterschiedlich sein. D. h. die IP-Adresse aller Schnittstellen müssen sich im Subnetz-Teil unterscheiden.
  Nur wenn die Schnittstellen in unterschiedlichen IP-Subnetzen liegen, ergeben sich Vorteile hinsichtlich der Kommunikation über mehrere PROFINET-Schnittstellen.
  Die Subnetzmaske, die Sie bei der IP-Adressvergabe ebenfalls projektieren, entscheidet mit darüber, ob die IP-Subnetze der beiden Schnittstellen unterschiedlich sind oder nicht. Wenn die CPU einen Ethernet-Teilnehmer adressiert, entscheidet sie anhand der Regeln, die in RFC 4632 beschrieben sind, über welche Schnittstelle die entsprechenden Datenpakete gesendet bzw. empfangen werden.
  Beispiel: Beide IP-Adressen liegen in unterschiedlichen Subnetzen
  - Subnetz-Maske zweier Schnittstellen: 255.255.255.0
  - IP-Adresse der ersten Schnittstelle: 192.168.**0**.10
  - IP-Adresse der zweiten Schnittstelle: 192.168.**1**.10

  Negativ-Beispiel: Beide IP-Adressen liegen im selben Subnetz
  - Subnetz-Maske zweier Schnittstellen: 255.255.0.0
  - IP-Adresse der ersten Schnittstelle: 192.168.0.10
  - IP-Adresse der zweiten Schnittstelle: 192.168.1.10
- Die Schnittstellen können am selben physikalischen Netz angeschlossen sein (ein physikalisches Netzwerk mit zwei logischen Teilnetzen, d. h. mit zwei IPv4-Netzen).
- Maximal ein Router darf wirksam sein. Die IP-Adresse des Routers muss im IP-Subnetz der PROFINET-Schnittstelle liegen, an der der Router projektiert ist.
- Bei der automatischen Vergabe des Gerätenamens wird neben dem Namen des Geräts (CPU) der Name des Interfaces verwendet. Beispiel: plc_1.profinet interface_1

- Uhrzeitsynchronisation: Die Aktivierung der Synchronisation und die Einstellungen für die NTP-Server sind nur an der PROFINET-Schnittstelle X1 möglich. Die Einstellungen werden für die Schnittstelle X2 und weitere Schnittstellen übernommen (dort nur lesbar).
- Keep-Alive-Option: Bei der Aktivierung der Erweiterten Option "Sende Keep Alives für Verbindungen ..." gilt dasselbe wie für die Uhrzeitsynchronisation: Nur an Schnittstelle X1 aktivierbar, an Schnittstelle X2 und weiteren Schnittstellen nur lesbar.

Online verbinden

Wenn Sie mit einem PG/PC mit STEP 7 online auf eine CPU mit zwei oder drei PROFINET-Schnittstellen zugreifen wollen, dann laufen im Hintergrund folgende Aktivitäten:

- STEP 7 prüft beim Online verbinden, ob sich eine der mit dem S7-Subnetz verbundenen PROFINET-Schnittstellen im gleichen IP-Subnetz wie die eigene IP-Adresse (= IP-Adresse des Online-Zugangs des PG/PC) befindet und geht dann über diese IP-Adresse / PROFINET-Schnittstelle online.
- Falls keine der PROFINET-Schnittstellen im gleichen IP-Subnetz liegt, geht STEP 7 über die PROFINET-Schnittstellen online, für die ein Router konfiguriert ist.
- Wenn die Voraussetzungen nicht erfüllt sind, bietet STEP 7 über eine Meldung an, die IP-Subnetzadresse der PG/PC-Schnittstelle zu ändern, so dass das PG im selben IP-Subnetz liegt.

Begrenzung der Dateneinspeisung ins Netz

Dateneinspeisung ins Netz an PROFINET-Schnittstellen begrenzen

Die Funktion "Dateneinspeisung ins Netz begrenzen" begrenzt die Netzlast von Standard-Ethernet-Kommunikation, welche von der Schnittstelle in das Netzwerk eingespeist wird, auf einen maximalen Wert. Zyklische Echtzeitkommunikation (RT/IRT) ist davon nicht betroffen.

Bei einer Linientopologie können schnell kritische Netzlasten durch Standard-Ethernet-Kommunikation entstehen. Geräte in einer Linientopologie sollten die "Begrenzung der Dateneinspeisung ins Netz" unterstützen.

Geräte, die viel "Standard-Ethernet-Kommunikation" in das Netz einspeisen, sollten Sie nicht in einer Linientopologie, sondern direkt hinter einem Switch platzieren.

Abhängig von der Schnittstelle können Sie die Funktion "Dateneinspeisung ins Netz begrenzen" aktivieren bzw. deaktivieren.Wenn Sie die Schnittstelle X1 einer S7-1500 CPU als IO-Controller oder I-Device verwenden, dann ist die Funktion "Dateneinspeisung ins Netz begrenzen" immer aktiv. Wenn Sie die Schnittstelle X1 einer S7-1500 CPU nicht als IO-Controller oder I-Device nutzen, dann können Sie die Funktion aktivieren bzw. deaktivieren. Für andere Schnittstellen können Sie die Funktion "Dateneinspeisung ins Netz begrenzen" immer aktivieren bzw. deaktivieren.

Nutzen von der Begrenzung der Dateneinspeisung ins Netz

- Faires Aufteilen der verbleibenden Bandbreite:
  In PROFINET-Netzen teilen sich zyklische und nicht zyklische Kommunikation das selbe Netz. Das führt dazu, dass für nichtzyklische Kommunikation nur eine begrenzte Bandbreite übrig bleibt. Die Begrenzung der Dateneinspeisung stellt sicher, dass die verbleibende Bandbreite für nichtzyklische Kommunikation fair zwischen den Geräten im Netz aufgeteilt wird. Die Begrenzung der Dateneinspeisung löst keine globalen Überlastprobleme im Netz, aber sie vermeidet, dass ein einzelnes Gerät die gesamte Bandbreite besetzt.
- Spitzen in der Dateneinspeisung glätten:
  Die Begrenzung der Dateneinspeisung glättet Spitzenlasten von nicht-zyklischen Daten (z. B. von Open User Communication, Zugriffe vom Webserver).
- Probleme in der Dateneinspeisung eingrenzen:
  Geräten ist es verboten, zu viel Netzlast in ein PROFINET-Netzwerk einzuspeisen. Wenn Anwendungen in einem Gerät zuviele Daten erzeugen, dann werden diese Daten nicht ins PROFINET-Netz weitergeleitet. Negative Auswirkungen (z. B. Datenverlust, Kommunikationsabruch) bleiben auf das Gerät und dessen Kommunikationspartner beschränkt. Weitere Teilnehmer werden nicht gestört.

Begrenzung der Dateneinspeisung ins Netz für eine CPU einstellen

Um die Begrenzung der Dateneinspeisung ins Netz einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 die Schnittstelle der CPU.
2. Navigieren Sie im Inspektirfenster zu
   "Eigenschaften" > "Allgemein" > "Erweiterte Optionen" > "Schnittstellen-Optionen".
3. Aktivieren bzw. Deaktivieren Sie das Optionskästchen "Dateneinspeisung ins Netz begrenzen".

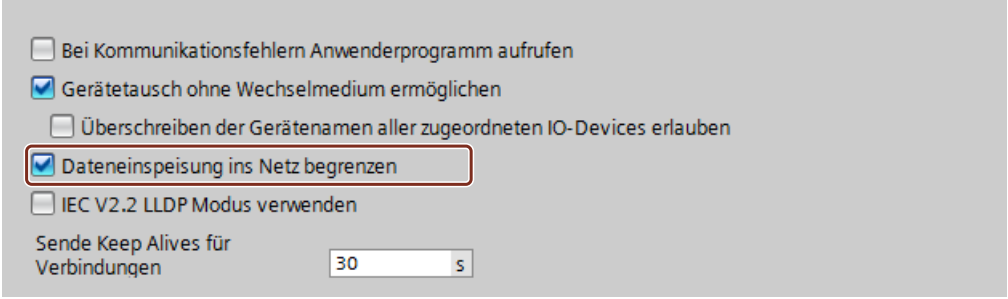

IEEE-konformes LLDP

LLDP steht für das Protokoll "Link Layer Discovery Protocol"; ein herstellerunabhängiges Protokoll, das in der Norm IEEE-802.1AB definiert ist.

Mit LLDP senden Ethernet-Teilnehmer in regelmäßigen Abständen Informationen über sich an ihre Nachbargeräte. Die Nachbargeräte speichern die Informationen.

SNMP-Tools lesen diese Daten und erstellen daraus die Topologie des Netzes (SNMP steht für "Simple Network Management Protocol").

LLDP und IEC 61158

Die Feldbusnorm IEC 61158 V2.2, kurz "IEC V2.2", setzt LLDP für Geräte mit einer PROFINET-Schnittstelle um. Damit erkennen SNMP-Tools die Topologie von Ethernet-Netzen, an die Geräte mit einer PROFINET-Schnittstelle angeschlossen sind.

Geräte mit mehreren PROFINET-Schnittstellen erfordern eine neue Version dieser Norm, die IEC V2.3. Damit erkennen Siemens-Tools und Fremdtools alle Schnittstellen eines Geräts mit mehreren PROFINET-Schnittstellen.

Alle vernetzten PROFINET-Schnittstellen müssen auf denselben Modus eingestellt sein (entweder IEC V2.3 oder IEC V2.2).

Wenn Sie alle Geräte am PROFINET-Subnetz innerhalb eines Projektes konfigurieren, stellt STEP 7 automatisch den richtigen Modus ein und Sie brauchen sich um die Einstellungen nicht zu kümmern. Im Folgenden erfahren Sie, nach welchen Regeln STEP 7 die Einstellungen automatisch vornimmt und in welchen Fällen Sie den Modus manuell einstellen müssen.

Woran erkennt man, welchen Modus eine PROFINET-Schnittstelle unterstützt?

Welchen Modus eine PROFINET-Schnittstelle unterstützt, sehen Sie in den Eigenschaften der PROFINET-Schnittstelle bei markiertem Gerät bzw. bei markierter Schnittstelle in der Gerätesicht oder Netzsicht. Im Bereich "PROFINET-Schnittstelle > Erweiterte Optionen" finden Sie die Option "IEC V2.2 LLDP Modus verwenden".

- Wenn die Option "IEC V2.2 LLDP Modus verwenden" aktiviert und nicht veränderbar ist, dann unterstützt die PROFINET-Schnittstelle nur den Modus V2.2.
- Wenn die Option "IEC V2.2 LLDP Modus verwenden" deaktiviert und veränderbar ist, dann unterstützt die PROFINET-Schnittstelle sowohl den Modus V2.2.als auch den Modus V2.3.

Automatische Einstellung der PROFINET-Schnittstellen

STEP 7 setzt den Modus der PROFINET-Schnittstellen nach den folgenden Regeln:

- Wenn die Geräte den Modus IEC V2.3 unterstützen, dann stellt STEP 7 alle PROFINET-Schnittstellen am Subnetz einheitlich auf den Modus IEC V2.3 ein.
- Wenn ein Gerät nur den Modus IEC V2.2 unterstützt, dann stellt STEP 7 alle PROFINET-Schnittstellen am Subnetz auf den Modus IEC V2.2.

Beispiel 1: Automatische Einstellung

In diesem Beispiel wurde ein I-Device im Projekt 1 angelegt und anschließend in das Projekt 2 als GSD-Datei eingefügt (zwei Projekte).

Im Projekt 1 unterstützt ein Gerät (3) nur den Modus V2.2. Deshalb setzte STEP 7 alle PROFINET-Schnittstellen in diesem Projekt auf den Modus V2.2 und vermerkte diese Einstellung in der GSD-Datei.

Anschließend wurde das I-Device-Projekt in einer GSD-Datei exportiert (2) und in das Projekt 2 eingefügt.

STEP 7 setzte wegen des Eintrags in der GSD-Datei automatisch alle PROFINET-Schnittstellen im Projekt 2 ebenfalls auf den Modus V2.2.

In diesem Beispiel parametriert der übergeordnete IO-Controller das untergeordnete IO-System (1).

SNMP-Tools erkennen in diesem Beispiel nur eine Schnittstelle pro Gerät. Tools von Siemens erkennen alle PROFINET-Schnittstellen.

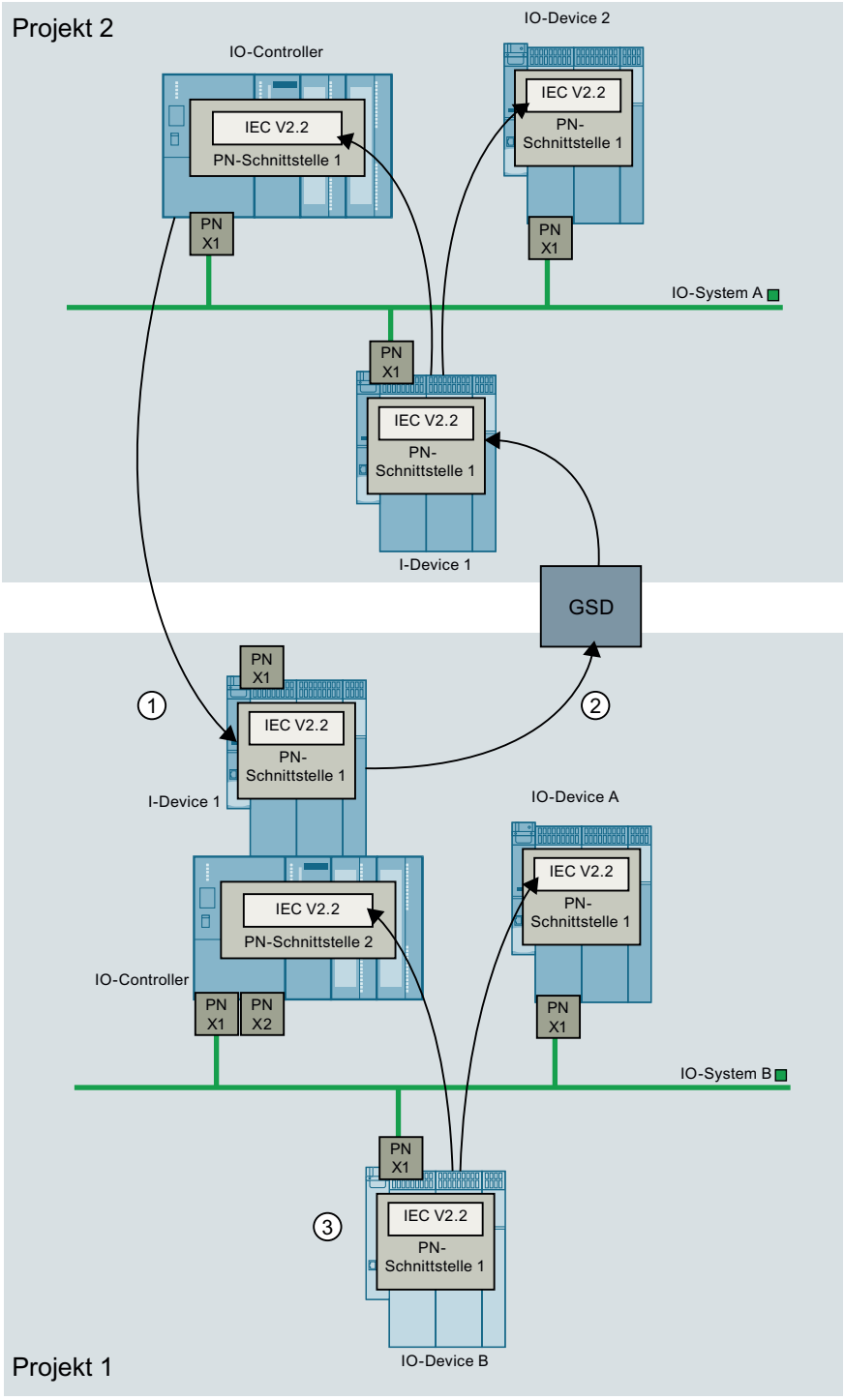

① Übergeordneter IO-Controller parametriert die PROFINET-Schnittstelle das I-Device am IO-System A.

② PROFINET GSD-Datei enthält Eintrag, dass nur Modus IEC V2.2 unterstützt wird.

③ Geräte mit Modus IEC V2.2 erzwingen bei allen PROFINET-Schnittstellen im Projekt 1 den Modus IEC V2.2.

Manuelle Einstellung der PROFINET-Schnittstellen

Die manuelle Einstellung einer PROFINET-Schnittstelle ist in den folgenden Fällen erforderlich:

- Sie projektieren eine Anlage in zwei oder mehreren Projekten. In einem der Projekte (Projekt 1) wird ein Gerät verwendet, das nur den Modus V2.2 unterstützt. In einem anderen Projekt (Projekt 2) werden ausschließlich Geräte im Modus V2.3 genutzt. Dann müssen Sie eines der Geräte im Projekt 2 manuell auf den Modus V2.2 setzen.
- Sie verwenden zwei Projekte. Das erste Projekt enthält ein I-Device. Das zweite Projekt enthält ein übergeordnetes IO-System, in das das I-Device eingebunden wird. Dieser Fall wird im Folgenden näher dargestellt:

IEC-Modus manuell einstellen

Wenn die folgenden Bedingungen vorliegen, dann müssen Sie eine PROFINET-Schnittstelle manuell auf den Modus IEC V2.2 einstellen:

- Sie projektieren ein I-Device und exportieren das Projekt als GSD-Datei.
- Sie verwenden die GSD-Datei in einem anderen Projekt
- Sie fügen das I-Device in ein übergeordnetes IO-System ein.
- Das Projekt mit dem I-Device verwendet ausschließlich Geräte, die den Modus V2.3 unterstützen. Dieser Modus wird automatisch eingestellt.
- Im überlagerten IO-System befindet sich ein Gerät, das nur die Norm IEC V2.2 unterstützt.

Sie zwingen eine PROFINET-Schnittstelle eines Geräts am unterlagerten IO-System in den Modus V2.2, indem Sie die folgenden Schritte ausführen.

1. Öffnen Sie das I-Device-Projekt in STEP 7.
2. Markieren Sie ein beliebigen Gerät.
3. Aktivieren Sie die Option "IEC V2.2 LLDP Modus verwenden" im Bereich "Schnittstellen-Optionen".

Generieren Sie anschließend die GSD-Datei neu. Verwenden Sie die neue GSD-Datei im übergeordneten IO-System.

Laden Sie die neue I-Device-Projektierung in die I-Device-CPU.

Beispiel 2: IEC-Modus manuell einstellen

Projekt 1: Im Projekt ist ein I-Device angelegt.

Alle PROFINET-Schnittstellen unterstützen den Modus V2.3.

Deshalb änderte STEP 7 den Modus der PROFINET-Schnittstellen nicht und vermerkte den Modus V2.3 in der GSD-Datei (2).

Projekt 2: I-Device aus Projekt 1 wird eingefügt (GSD-Datei).

Ein Device unterstützt nur den Modus V2.2.

STEP 7 setzt deshalb alle PROFINET-Schnittstellen in diesem Projekt auf den Modus V2.2.

Die PROFINET-Schnittstellen im Projekt 1 und 2 sind unterschiedlich eingestellt.

Deshalb müssen Sie das I-Device-Projekt (Projekt 1) wieder in STEP 7 öffnen und eine PROFINET-Schnittstelle manuell auf den Modus V2.2 einstellen.

Sie stellen eine PROFINET-Schnittstelle eines Geräts am unterlagerten IO-System auf den Modus V2.2 ein, indem Sie die folgenden Schritte ausführen.

1. Öffnen Sie das I-Device-Projekt in STEP 7.
2. Markieren Sie ein beliebiges Gerät.
3. Aktivieren Sie die Option "IEC V2.2 LLDP Modus verwenden" im Bereich "Schnittstellen-Optionen".

Anschließend erzeugen Sie eine neue GSD-Datei. Fügen Sie die neue GSD-Datei in das übergeordnete IO-System (Projekt 2) ein.

Laden das Projekt neu in die I-Device-CPU.

SNMP-Tools erkennen in diesem Beispiel nur eine Schnittstelle pro Gerät. Tools von Siemens erkennen alle PROFINET-Schnittstellen.

Wenn Sie die Einstellung der PROFINET-Schnittstelle nicht ändern, dann sendet jede Schnittstelle zur Laufzeit (während des Betriebs) eine Diagnosemeldung.

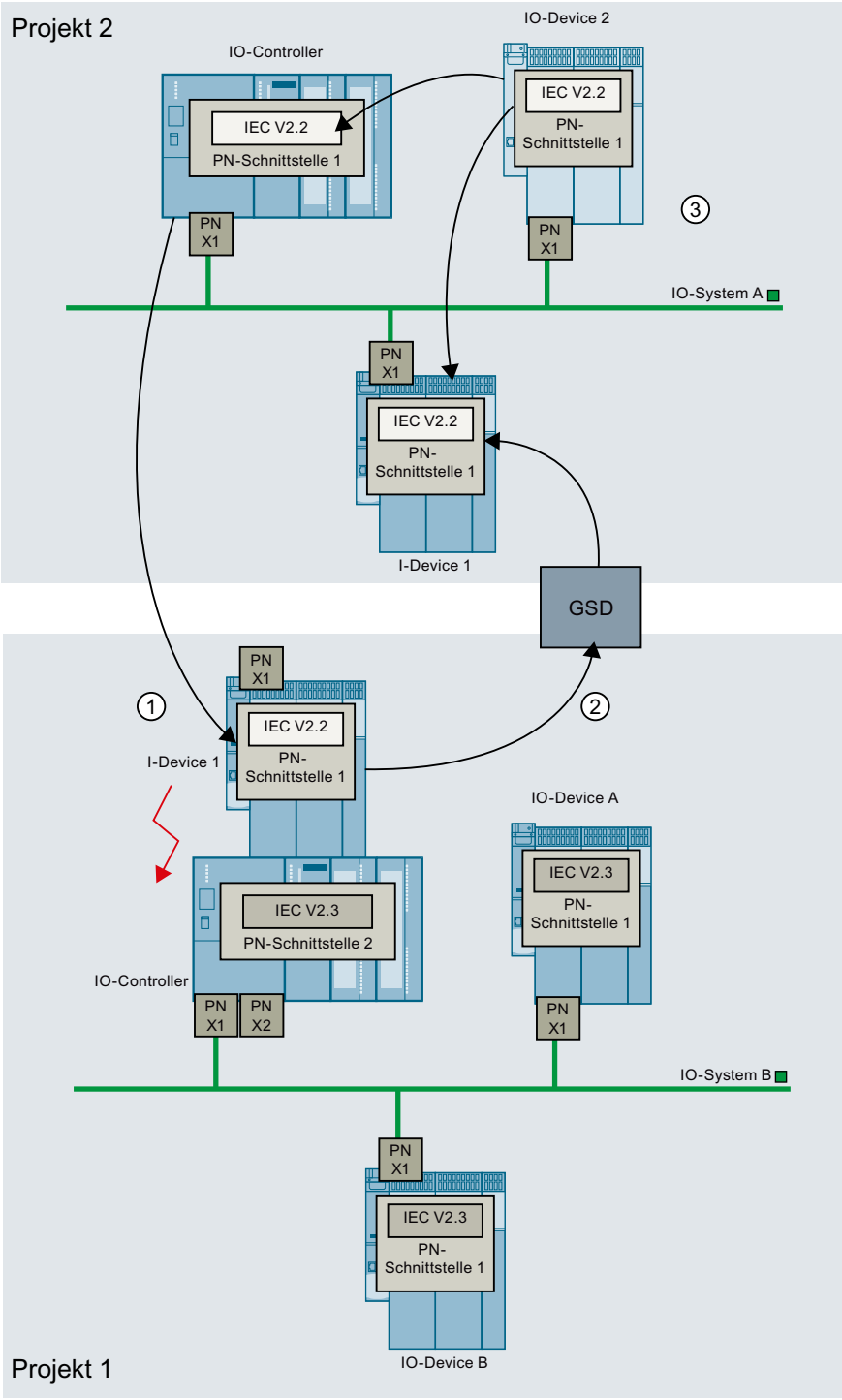

① Die PROFINET-Schnittstellen des I-Devices sind unterschiedlich eingestellt (die PROFINET-Schnittstelle am IO-System A auf Modus V2.2, die PROFINET-Schnittstelle am IO-System B auf Modus V2.3).

② GSD-Datei enthält Eintrag, dass das I-Device den Modus IEC V2.3 unterstützt.

③ IO-Device 2 unterstüzt nur den Modus V2.2. Deshalb setzt STEP 7 alle anderen PROFINET-Schnittstellen im Projekt 2 auf diesen Modus (auch die PROFINET-Schnittstelle des I-Devices am IO-System A).

Beispiel 3: IEC-Modus manuell einstellen

Dieses Beispiel entspricht weitgehend dem Beispiel 2.

Einziger Unterschied: Im I-Device sind die Einstellungen der PROFINET-Schnittstellen gespeichert (keine Parametrierung durch den übergeordneten IO-Controller wie im Beispiel 2).

Am IO-System A sind die PROFINET-Schnittstellen unterschiedlich eingestellt (I-Device auf V2.3, IO-Controller auf V2.2, IO-Device 2 unterstützt nur V2.2).

Deshalb müssen Sie eine PROFINET-Schnittstelle am IO-System B auf den Modus V2.2 einstellen.

Sie stellen eine PROFINET-Schnittstelle eines Geräts am unterlagerten IO-System auf den Modus V2.2 ein, indem Sie die folgenden Schritte ausführen.

1. Öffnen Sie das I-Device-Projekt in STEP 7.
2. Markieren Sie ein beliebigen Gerät.
3. Aktivieren Sie die Option "IEC V2.2 LLDP Modus verwenden" im Bereich "Schnittstellen-Optionen".

Generieren Sie anschließend die GSD-Datei neu und verwenden Sie die neue GSD-Datei.

Laden Sie die neue I-Device-Projektierung in die I-Device-CPU.

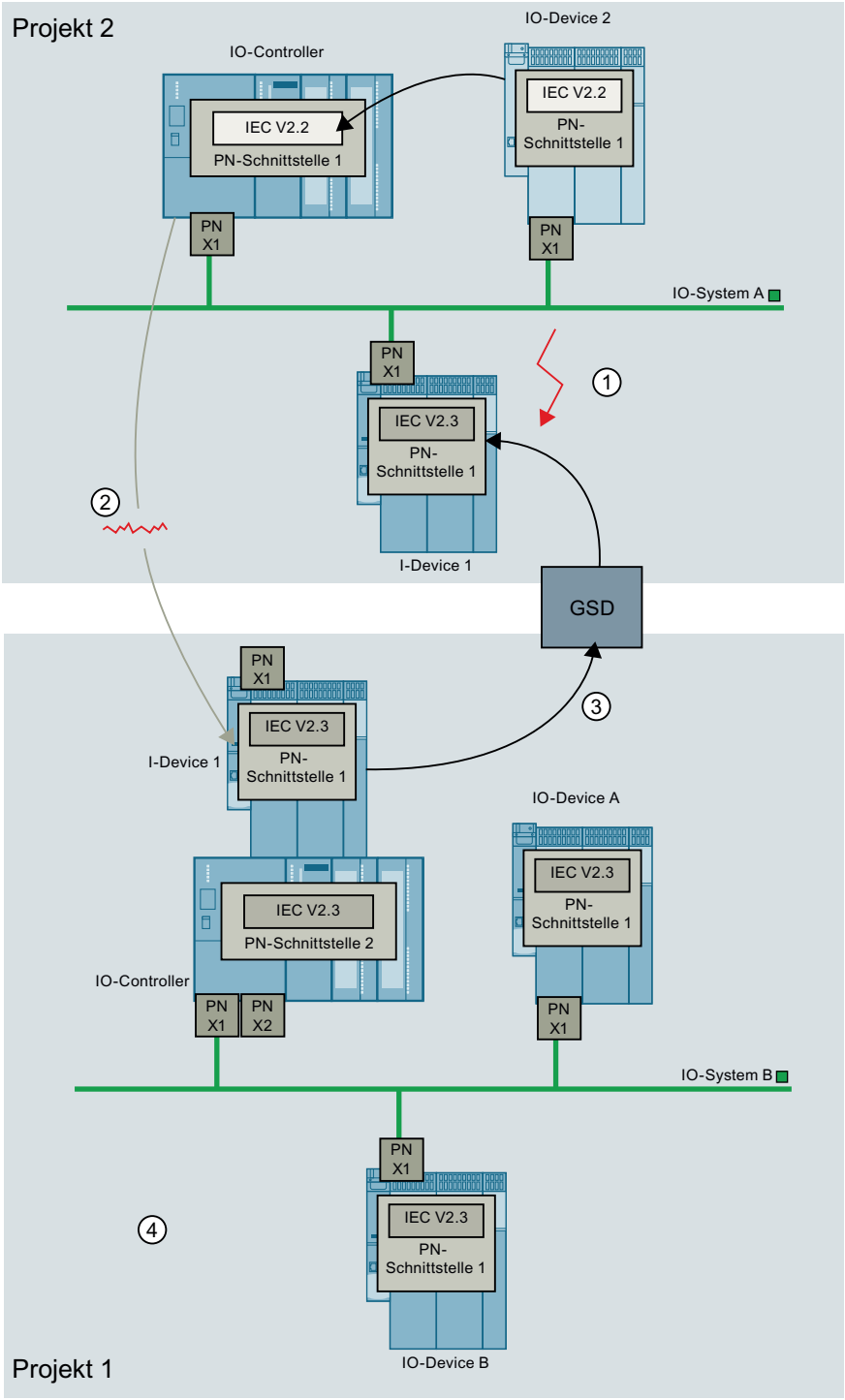

① Am IO-System A sind die PROFINET-Schnittstellen unterschiedlich eingestellt (I-Device auf V2.3, IO-Controller auf V2.2, IO-Device 2 unterstützt nur V2.2).

② Im I-Device sind die Einstellungen der PROFINET-Schnittstellen gespeichert (keine Parametrierung durch den übergeordneten IO-Controller).

③ GSD-Datei enthält Eintrag, dass Mode IEC V2.3 eingestellt ist.

④ Alle Geräte sind automatisch auf den Modus V2.3 eingestellt: Deshalb müssen Sie eine PROFINET-Schnittstelle im Projekt 1 manuell auf den Modus V2.2 einstellen.

Uhrzeitfunktionen

Grundlagen zu den Uhrzeitfunktionen

Alle S7-1500 CPUs sind mit einer internen Uhr ausgestattet. Die Pufferung unterstützt die Anzeige der korrekten Uhrzeit für eine Unterbrechung der Spannungsversorgung für bis zu 10 Stunden.

Uhrzeitformat

Die Uhr zeigt immer die Uhrzeit mit einer Auflösung von 1 Millisekunde und das Datum mit Wochentag an. Die durch die Sommerzeit bedingte Zeitumstellung wird berücksichtigt.

Siehe auch

Stellen und Lesen der Uhrzeit (Seite 1023)

Stellen und Lesen der Uhrzeit

Uhrzeit stellen und lesen mit Anweisungen

Sie können Uhrzeit und Datum der CPU-Uhr über folgende Anweisungen im Anwenderprogramm stellen, starten und lesen:

- Uhrzeit stellen: "WR_SYS_T"
- Uhrzeit lesen "RD_SYS_T"
- Lokalzeit lesen "RD_LOC_T"
- Lokalzeit schreiben "WR_LOC_T"

Manuelle Einstellung

Das Lesen und Einstellen der Uhrzeit können Sie auch manuell in der Online- und Diagnosesicht unter "Funktionen > Uhrzeit einstellen" durchführen.

Uhrzeitsynchronisation SIMATIC-Verfahren

Bezug

Parametrierung der Schnittstellen-Eigenschaften in der Parametergruppe "Eigenschaften > Allgemein > DP-Schnittstelle > Uhrzeitsynchronisation".

SIMATIC-Verfahren

Beim SIMATIC-Verfahren wird die Uhrzeit nach MMS-Uhrzeitnachrichten (MMS - Manufacturing Message Specifaction) gestellt. Die MMS-Uhrzeitnachrichten stammen von einem SIMATIC-Uhrzeitsender oder einer CPU, welche als Uhrzeit-Master projektiert ist.

Bei einem CP können die Uhrzeitnachrichten entweder von der lokalen CPU oder über LAN empfangen werden. Es ist einstellbar, ob der CP die empfangene Uhrzeit weiterleiten soll.

Gegenüber eines eventuell verfügbaren NTP-Verfahrens bietet das SIMATIC-Verfahren eine höhere Genauigkeit.

Parameter der Uhr

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Synchronisationsart
  Wenn die Synchronisationsart "Als Master" eingestellt ist, dann ist das Zeitintervall für die automatische Synchronisation wählbar.
  Wenn in einem Netz mehr als eine Baugruppe mit Uhr vorhanden ist, müssen Sie einstellen, welche CPU bei der Synchronisation der Uhrzeit als Master und welche als Slave fungieren soll. Es darf nur einen Uhrzeit-Master geben.
- Zeitintervall
  Das Zeitintervall definiert den Zeitabstand der Uhrzeitanfragen.

Anweisungen für die Synchronisation der Uhrzeit

Um sicherzustellen, dass die Uhrzeit aller Baugruppen im Netz übereinstimmt, werden die Uhrzeit-Slaves vom Systemprogramm in regelmäßigen, parametrierbaren Abständen synchronisiert. Mit der Anweisung "SNC_RTCB" können Sie Datum und Uhrzeit unabhängig vom parametrierten Zeitintervall vom Uhrzeit-Master an den Uhrzeit-Slave übertragen.

Uhrzeitsynchronisation NTP-Verfahren

Bezug

Parametrierung der Schnittstellen-Eigenschaften in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Uhrzeitsynchronisation". Die Option "Uhrzeitsynchronisation über NTP-Server aktivieren" ist gewählt.

NTP-Verfahren (NTP: Network Time Protocol)

Beim NTP-Verfahren sendet das Gerät in regelmäßigen Zeitabständen Uhrzeitanfragen (im Client-Modus) an NTP-Server im Subnetz (LAN). Anhand der Anworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und die Uhrzeit der Station synchronisiert.

Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren. Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen (in Sekunden). Der Wertebereich des Intervalls liegt zwischen 10 Sekunden und einem Tag.

Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen; dies entspricht GMT (Greenwich Mean Time).

Parameter der Uhrzeitsynchronisation im NTP-Verfahren

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Uhrzeitsynchronisation über NTP-Server aktivieren
  Aktivieren Sie das Optionskästchen, wenn die interne Uhr mit dem NTP-Synchronisationsverfahren synchronisiert werden soll.
- Server 1-4
  Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren.
- Aktualisierungsintervall
  Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen.

Verwenden von Systemmerkern

Systemmerker

Ein Systemmerker ist ein Merker mit definierten Werten.

Welches Merkerbyte der CPU zum Systemmerkerbyte wird, bestimmen Sie bei der Parametrierung des Systemmerkers.

Nutzen

Systemmerker können Sie im Anwenderprogramm verwenden, um z. B. Programmteile nur im ersten Zyklus nach dem Anlauf auszuführen oder die Diagnose bei Änderung des Diagnosezustands auszuwerten. Zwei Systemmerker sind konstant 1 bzw. konstant 0.

Bits des Systemmerkerbytes

Die folgende Tabelle zeigt die Bedeutung der Systemmerker:

Bit des Systemmerkerbytes	7	6	5	4	3	2	1	0
Bedeutung	Reserviert (=0)	Reserviert (=0)	Reserviert (=0)	Reserviert (=0)	=0	=1	=1 bei Änderung des Diagnosestatus	=1 während des Anlaufs und im ersten Zyklus nach dem Anlauf, sonst 0

Hinweis

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

Verwenden von Taktmerkern

Taktmerker

Ein Taktmerker ist ein Merker, der seinen Binärzustand periodisch im Puls-Pausen-Verhältnis 1:1 ändert.

Welches Merkerbyte der CPU zum Taktmerkerbyte wird, bestimmen Sie bei der Parametrierung des Taktmerkers.

Nutzen

Taktmerker können Sie im Anwenderprogramm verwenden, um z. B. Leuchtmelder mit Blinklicht anzusteuern oder periodisch wiederkehrende Vorgänge anzustoßen, etwa das Erfassen eines Istwerts.

Mögliche Frequenzen

Jedem Bit des Taktmerkerbytes ist eine Frequenz zugeordnet. Die folgende Tabelle zeigt die Zuordnung:

Bit des Taktmerkerbytes	7	6	5	4	3	2	1	0
Periodendauer (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frequenz (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Hinweis

Taktmerker laufen asynchron zum CPU-Zyklus, d. h. in langen Zyklen kann sich der Zustand des Taktmerkers mehrfach ändern.

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

Schutz

Einstellmöglichkeiten für den Schutz

Zugriffsstufen

Im Folgenden erfahren Sie, wie Sie die einzelnen Zugriffsstufen der S7-1500 CPUs nutzen.

S7-1500 CPUs bieten verschiedene Zugriffsstufen, um den Zugang zu bestimmten Funktionen einzuschränken.

Die einzelnen Zugriffsstufen sowie die dazugehörigen Passwörter legen Sie in den Objekteigenschaften der CPU fest. Sie parametrieren die Zugriffsstufe in einer Tabelle.

Zugriffsstufe	Zugriff			Zugriffserlaubnis
	HMI	Lesen	Schreiben	Passwort
Vollzugriff (kein Schutz)	✓	✓	✓	
Lesezugriff	✓	✓		
HMI-Zugriff	✓			
Kein Zugriff (kompletter Schutz)				

Die grünen Haken in den Spalten rechts der jeweiligen Zugriffsstufe geben an, welche Operationen maximal möglich sind, ohne das Passwort dieser Zugriffsstufe zu kennen.

Wenn Sie die Funktionen nicht markierter Felder in der Spalte "Zugriff" nutzen, ist die Eingabe eines Passwortes notwendig:

Beispiel:

Sie stellen die Zugriffsstufe "Lesezugriff" ein. Aus der Tabelle ist ersichtlich, dass im laufenden Betrieb ohne Passworteingabe kein schreibender Zugriff möglich ist.

Aus der Tabelle geht ebenfalls hervor, dass für die Funktion "Schreiben" Vollzugriff notwendig ist.

Wer im laufenden Betrieb eine Funktion nutzt, die schreibenden Zugriff benötigt, muss also das Passwort für den Vollzugriff eingeben!

ACHTUNG
Projektierung einer Zugriffsstufe ersetzt nicht den Know-how-Schutz Die Parametrierung von Zugriffsstufen bietet hochwertigen Schutz vor unrechtmäßigen Änderungen an der CPU, indem die Rechte zum Laden in die CPU eingeschränkt werden. Bausteine auf der Memory Card sind jedoch nicht schreib- oder lesegeschützt. Um den Code von Bausteinen auf der Memory Card zu schützen, verwenden Sie den Know-how-Schutz.

Voreingestelltes Verhalten

Voreingestellt ist die Zugriffsstufe "Vollzugriff (kein Schutz)". Jeder Nutzer kann die Hardware-Konfiguration und die Bausteine lesen und verändern. Ein Passwort ist nicht parametriert und wird auch für den Online-Zugriff nicht benötigt.

Die Zugriffsstufen im Einzelnen

Im Folgenden finden Sie die Erläuterung zu den existierenden Zugriffsstufen und welche Funktionen in der jeweiligen Zugriffsstufe möglich sind.

- Vollzugriff (kein Schutz):
  Hardware-Konfiguration und Bausteine können von jedem gelesen und verändert werden.
- Lesezugriff für F-Bausteine (nur bei F-CPUs):
  F-Bausteine des Sicherheitsprogramms können nicht ohne Legitimation durch das Passwort dieser oder einer höheren Zugriffsstufe verändert werden.
  Weitere Informationen dazu erhalten Sie im Programmier- und Bedienhandbuch *SIMATIC Safety - Projektieren und Programmieren.*
- Lesezugriff:
  Nur lesender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich, d. h. Sie können die Hardware-Konfiguration und Bausteine ins Programmiergerät laden. Möglich ist außerdem der HMI-Zugang und Zugriff auf Diagnosedaten, Anzeige von Online/Offline Vergleichsergebnissen, Uhrzeit stellen sowie Wechsel des Betriebszustands (RUN/STOP). Sie können keine Bausteine und keine Hardware-Konfiguration in die CPU laden. Außerdem sind keine Testfunktionen und Firmware-Updates möglich.
- HMI-Zugriff:
  Nur HMI-Zugang und Zugriff auf Diagnosedaten möglich.
  Beispiel: Sie können mit Zugriffsstufe HMI-Zugriff Online gehen und sich die Diagnosesymbole für die Zustände von Objekten anzeigen lassen.
  Über ein HMI-Gerät können Variablen gelesen und geschrieben werden.
  Sie können in dieser Zugriffsstufe weder Bausteine und Hardware-Konfiguration in die CPU laden noch von der CPU Bausteine und Hardware-Konfiguration ins Programmiergerät laden.
  Außerdem ist Folgendes **nicht** möglich: Testfunktionen, Wechsel des Betriebszustands (RUN/STOP), Firmware-Update und Anzeige von Online/Offline Vergleichsergebnissen.
- Kein Zugriff (kompletter Schutz):
  Nur Identifiktionsdaten lesbar z. B. über "Erreichbare Teilnehmer".
  Weder lesender noch schreibender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich. Auch der HMI-Zugriff ist nicht möglich. Die Server-Funktion für PUT/GET-Kommunikation ist in dieser Zugriffsstufe deaktiviert (nicht änderbar).
  Durch die Legitimation mit einem parametrierten Passwort erhalten Sie Zugriff entsprechend der zugehörigen Schutzstufe.

Verhalten von Funktionen bei unterschiedlichen Zugriffsstufen

In der folgenden Tabelle ist beschrieben, welche Online-Funktionen in den verschiedenen Schutzstufen möglich sind.

Funktion	Vollzugriff	Lesezugriff	HMI-Zugriff	Kein Zugriff
Identifikation des Geräts z. B. über Erreichbare Teilnehmer	Ja	Ja	Ja	Ja
Zuweisen einer Notfalladresse (Emergency IP)	Ja	Ja	Ja	Ja
HMI-Diagnoseanzeige	Ja	Ja	Ja	Nein

Variable beobachten (M, E, A, DB-Inhalte) über HMI-Gerät	Ja	Ja	Ja	Nein
Variable steuern (M, E, A, DB-Inhalte) über HMI-Gerät	Ja	Ja	Ja	Nein
Diagnoseanzeige (z. B. Geräteinformation, Verbindungsanzeige, Meldungsanzeige, Diagnosepuffer)	Ja	Ja	Ja	Nein
Lesen von Zykluszeitstatistiken (Online & Diagnose)	Ja	Ja	Ja	Nein
Informationen aus der Hardware-Konfiguration lesen (Online & Diagnose)	Ja	Ja	Ja	Nein
Uhrzeit lesen	Ja	Ja	Ja	Nein
Onlinefunktionen innerhalb der Hardware-Konfiguration ausführen (Online & Diagnose)	Ja	Ja	Ja	Nein
Meldungen quittieren	Ja	Ja	Ja	Nein
Meldungen empfangen	Ja	Ja	Ja	Nein
Meldungen freigeben/sperren	Ja	Ja	Nein	Nein
Variable lesen über Testfunktion (STEP 7, Variablentabelle oder Beobachtungstabelle)	Ja	Ja	Nein	Nein
Betriebszustandswechsel online anfordern (RUN/STOP/Warmstart)	Ja	Ja	Nein	Nein
Laden von Datenbausteinen, Codebausteinen, Hardware-Konfiguration ins PG/PC	Ja	Ja	Nein	Nein
Uhrzeit stellen	Ja	Ja	Nein	Nein
Online/Offline Vergleichsergebnisse anzeigen	Ja	Ja	Nein*	Nein
Löschen von Datenbausteinen, Codebausteinen, Hardware-Konfiguration in der CPU	Ja	Nein	Nein	Nein
Laden von einzelnen Datenbausteinen, Codebausteinen, Hardware-Konfiguration in die CPU	Ja	Nein	Nein	Nein
PLC-Programm ins Gerät laden und zurücksetzen	Ja	Nein	Nein	Nein
Firmware-Update von CPUs oder Peripheriemodulen	Ja	Nein	Nein	Nein
Variablen steuern über Testfunktion (STEP 7, Beobachtungstabelle)	Ja	Nein	Nein	Nein
Variablen im Programmstatus lesen	Ja	Nein	Nein	Nein

Online-Editieren von Bausteinen	Ja	Nein	Nein	Nein
Ausgänge steuern im Betriebszustand STOP	Ja	Nein	Nein	Nein

* ab STEP 7 V14 reicht die Zugriffsstufe "HMI-Zugriff" für die Anzeige von Online/Offline Vergleichsergebnisse nicht aus, wie sie z. B. beim Online gehen mit Projekt für Bausteine, PLC-Datentypen oder PLC-Variablen angezeigt werden. Um Online/Offline Vergleichsergebnisse angezeigt zu bekommen, müssen Sie Lesezugriff haben. Grund: Die Gleichheit bzw. Ungleichheit von Online/Offline Daten stellt STEP 7 aufgrund eines Vergleichs der Prüfsummen für Software und Textlisten fest. Das Lesen dieser Prüfsummen erfordern die Zugriffsstufe "Lesezugriff".

Verhalten einer Passwort-geschützten Baugruppe im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft und im Falle eines Passwortschutzes zur Passworteingabe aufgefordert.

Online gehen mit einer Passwort-geschützten CPU

Online gehen mit einer Passwort-geschützen CPU setzt ab STEP 7 V14 Lesezugriff voraus. Sie müssen daher das Passwort für den Lesezugriff beim Online gehen eingeben oder, wenn hierfür kein Passwort parametriert wurde, das Passwort für den Vollzugriff.

Wenn Sie bei einer komplett geschützten CPU nur ein Passwort für den HMI-Zugriff zur Hand haben, brechen Sie die Passwortabfrage nach dem Passwort zum Lesezugriff ab. Sie werden anschließend aufgefordert, das Passwort für den HMI-Zugriff einzugeben. Die Berechtigung für den HMI-Zugriff reicht allerdings für den Online/Offline Vergleich nicht aus - hierfür benötigen Sie die Berechtigung für den Lesezugriff!

Beispiele:

- Eine Baugruppe wurde mit Lesezugriff parametriert und Sie wollen die Testfunktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff einer Testfunktion handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden (Passwort für den Vollzugriff).
- Eine Baugruppe wurde mit HMI-Zugriff parametriert und Sie wollen mit STEP 7 zu dieser Baugruppe online gehen. Da die Daten, aus denen STEP 7 die Gleichheit zwischen Online- und Offline-Objekten ermittelt (Prüfsumme), Lesezugriff voraussetzen, werden Sie beim online gehen zur Eingabe des Passworts für den Lesezugriff aufgefordert. Wenn Sie den Dialog zur Eingabe des Passworts abbrechen, wird trotzdem eine Online-Verbindung zur Baugruppe aufgebaut. Allerdings werden bei einem Online/Offline-Vergleich nur Fragezeichen in der Spalte für die Vergleichssymbole eingeblendet.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung. Wenn die Online-Verbindung nach einer Unterbrechung wiederhergestellt wird, müssen Sie die Zugangsdaten nicht erneut eingeben. Um die Zugangsberechtigung manuell aufzuheben, klicken Sie auf "Online > Zugriffsrechte löschen“.

Jede Zugriffsstufe lässt auch ohne Eingabe eines Passwortes den uneingeschränkten Zugriff auf bestimmte Funktionen zu, z. B. Identifikation über die Funktion "Erreichbare Teilnehmer".

Der Zugriff auf eine passwortgeschützte S7-1500-CPU kann lokal am Display eingeschränkt werden. Die Einschränkung wirkt nur, wenn der Betriebsartenschalter in Stellung RUN steht.

Siehe auch

Online-Zugriff auf CPU über Display sperren (Seite 1037)

Einschränken von Kommunikationsdiensten (Seite 1034)

Zugriffsstufen parametrieren (Seite 1031)

Zugriffsstufen parametrieren

Im Folgenden erfahren Sie, wie Sie für eine S7-1500 CPU eine Zugriffsstufe parametrieren und Passwörter eingeben.

Für eine S7-1500 CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Zugriffsstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Zugriffsstufe".

Beispiel

Sie wählen die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede der in der Tabelle darüber liegenden Zugriffsstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Vollzugriff (kein Schutz)) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Lesezugriff) wirkt, als wäre die CPU schreibgeschützt. Trotz Passwort-Kenntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (HMI-Zugriff) wirkt, als wäre die CPU schreib- und lesegeschüzt. Für Nutzer, die dieses Passwort kennen, sind nur HMI-Zugriffe möglich.

Vorgehen

Um die Zugriffsstufen für eine S7-1500 CPU zu parametrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften der Baugruppe im Inspektorfenster.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Schutz".
   Eine Tabelle mit den möglichen Zugriffsstufen wird im Inspektorfenster angezeigt.

Zugriffsstufe	Zugriff			Zugriffserlaubnis
	HMI	Lesen	Schreiben	Passwort
Vollzugriff (kein Schutz)	✓	✓	✓	
Lesezugriff	✓	✓		
HMI-Zugriff	✓			
Kein Zugriff (kompletter Schutz)				

3. Wählen Sie die gewünschte Zugriffsstufe in der ersten Spalte der Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Schutzstufe zeigen Ihnen, welche Operationen noch möglich sind, ohne das Passwort einzugeben.
4. Wenn Sie eine andere Zugriffsstufe als "Vollzugriff gewählt haben:
   Vergeben Sie in der Spalte "Passwort" in der ersten Zeile (Vollzugriff) ein Passwort für den Vollzugriff.
   Wiederholen Sie zum Schutz vor Fehleingaben das gewählte Passwort in der Spalte "Bestätigung".
   Achten Sie darauf, dass das Passwort ausreichend sicher ist, d. h. dass es kein durch eine Maschine erkennbares Muster besitzt!
   Die Eingabe eines Passwortes in der ersten Zeile "Vollzugriff (kein Schutz)" ist obligatorisch und ermöglicht dem Passwort-Kenner uneingeschränkten Zugriff auf die CPU, unabhängig von der gewählten Schutzstufe.
5. Weisen Sie weiteren Zugriffsstufen nach Bedarf weitere Passwörter zu, falls die gewählte Zugriffsstufe das erlaubt.
6. Laden Sie die Hardware-Konfiguration, damit die Zugriffsstufe wirksam wird.

Ergebnis

Die Hardware-Konfiguration und die Bausteine sind entsprechend der eingestellten Zugriffsstufe vor unberechtigtem Online-Zugriff geschützt. Wenn eine Operation aufgrund der parametrierten Zugriffsstufe nicht ohne Passwort ausgeführt werden kann, wird ein Dialog zur Eingabe eines Passwortes aufgeblendet.

Siehe auch

Einstellmöglichkeiten für den Schutz (Seite 1026)

Passwort-Übernahme beim Baugruppentausch

Wenn Sie eine Schutzstufe vergeben, die eine Passwort-Eingabe erfordert, dann wandelt die CPU diese Eingabe mit einem speziellen Algorithmus und speichert, vereinfacht ausgedrückt, eine spezielle Prüfsumme von diesem Passwort.

STEP 7 speichert niemals das Passwort selbst und in der CPU wird ebenfalls kein Passwort in rekonstruierbarer Form hinterlegt, so dass Sie im Fall eines Passwort-Verlustes immer eine neue CPU-Projektierung laden müssen, siehe folgenden Abschnitt.

Der Algorithmus zur Bildung bzw. Verschlüsselung der Prüfsumme aus dem Passwort wird ständig weiterentwickelt, um größtmöglichen Schutz vor unbefugten Zugriffen zu bieten. Das hat auf die Vorgehensweise bei der Projektierung keinen Einfluss mit Ausnahme des Baugruppentauschs: Je nach Firmware-Version der betroffenen CPUs (zu tauschende CPU und Austausch-CPU) werden Sie mit Informationen konfrontiert, die Ihnen ein Update auf einen aktuellen Algorithmus anbieten bzw. Sie dazu auffordern, neue Passwörter zu vergeben, da die Austausch-CPU die vorhandene Passwort-Projekierung nicht übernehmen kann.

Wenn sich die zu tauschende CPU und die Austausch-CPU hinsichtilich des verwendeten Algorithmus identisch verhalten, ist keine Aktion erforderlich; die Passwort-Projektierung wird wie die übrigen Parametereinstellungen übernommen.

Vergleichbares gilt für die projektierten Benutzer des CPU-Webservers: Wenn Sie Benutzer mit spezifischen Rechten anlegen, müssen Sie für jeden Benutzer ein Passwort projektieren. Auch hier werden Sie bem Baugruppentausch informiert, wenn die Algorithmen nicht kompatibel sind und müssen ggf. neue Benutzer mit ihren jeweiligen Passwörtern neu anlegen.

Passwort vergessen - neue CPU-Projektierung laden

Wenn Sie das Passwort für den Schutz der CPU nicht mehr besitzen, dann ist auch keine Online-Verbindung zu Laden der CPU mit einer neuen Konfiguration (mit neuem Passwort) möglich.

Um in diesem Fall eine neue Projektierung (mit neuer Passwortvergabe) zu laden, gehen Sie folgendermaßen vor:

1. Stecken Sie die SIMATIC Memory Card in den Card Reader Ihres Programmiergeräts.
2. Ziehen Sie in der Projektnavigation den CPU-Ordner mit der neuen Projektierung per Drag & Drop auf das Symbol für die SIMATIC Memory Card im Card-Reader.
3. Über den folgenden Lade-Dialog bestätigen Sie das Überschreiben von Konfiguration und Programm der bestehenen passwort-geschützten CPU-Projektierung.

Verhalten einer passwort-geschützten CPU beim Baugruppentausch

- Sie tauschen eine vorhandene S7-1500 CPU (**Firmware < 2.0**) mit einer S7-1500 (**Firmware ≥ 2.0**):
  Sie werden informiert, dass ein aktuellerer Algorithmus für die Passwort-Verschlüsselung zur Verfügung steht. Sie haben die Möglichkeit, den neuen Algorithmus anzuwenden, indem Sie auf die angebotene Schaltfläche klicken, oder - aus Kompatibilitätsgründen - den bisherigen Algorithmus beizubehalten.
- Sie tauschen eine vorhandene S7-1500 CPU (**Firmware ≥ 2.0**) mit einer S7-1500 (**Firmware < 2.0**):
  Sie werden informiert, dass die bisherigen Passwörter verloren gehen und neue Passwörter projektiert werden müssen.

Verhalten einer CPU mit projektierten Webserver-Benutzern beim Baugruppentausch

Um verschiedene Benutzer mit ihren jeweiligen Rechten anzulegen, müssen Sie für jeden Benutzer ein Passwort projektieren. Sinngemäß gelten ähnliche Regeln für den Baugruppentausch wie bei einer passwort-geschützten CPU:

- Sie tauschen eine vorhandene S7-1500 CPU (**Firmware < V2.0**) mit einer S7-1500 (**Firmware ≥ V2.0**):
  Sie werden informiert, dass ein aktuellerer Algorithmus für die Passwort-Verschlüsselung zur Verfügung steht. Sie haben die Möglichkeit, den neuen Algorithmus anzuwenden, indem Sie auf die angebotene Schaltfläche klicken, oder - aus Kompatibilitätsgründen - den bisherigen Algorithmus beizubehalten.
- Sie tauschen eine vorhandene S7-1500 CPU (**V1.7 ≤ Firmware ≤ V1.8**) mit einer S7-1500 (**Firmware ≤ 1.6**) oder
- Sie tauschen eine vorhandene S7-1500 CPU (**Firmware ≥ 2.0**) mit einer S7-1500 (**Firmware < 2.0**):
  Sie werden informiert, dass alle projektierten Benutzer gelöscht werden und der Benutzer "Jeder" auf die Standard-Benutzerrechte zurückgesetzt wird.
  In diesem Fall müssen Sie die Benutzer mir ihren Rechten und Passwörtern neu anlegen.

Einschränken von Kommunikationsdiensten

Einführung

Die CPU kann Server für eine Reihe von Kommunikationsdiensten sein, d. h. ohne dass Sie für die CPU Verbindungen projektiert und programmiert haben, können andere Kommunikationsteilnehmer auf CPU-Daten zugreifen.

Damit entfällt für die lokale CPU als Server die Möglichkeit, die Kommunikation zu den Clients kontrollieren.

Ob diese Art der Kommunikation im Betrieb für die lokale CPU zulässig ist oder nicht, wird durch den Parameter "Verbindungsmechanismen" im Bereich "Schutz" der CPU-Parameter bestimmt.

Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben

In der Voreinstellung ist die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner (...) erlauben" deaktiviert. In diesem Fall ist lesender und schreibender Zugriff auf CPU-Daten nur möglich bei Kommunikationsverbindungen, die eine Projektierung bzw. Programmierung sowohl für die lokale CPU als auch für den Kommunikationspartner voraussetzen. Zugriffe über BSEND/BRCV-Anweisungen sind z. B. möglich.

Verbindungen, für die die lokale CPU nur Server ist (d. h. für die lokale CPU ist keine Projektierung/Programmierung der Kommunikation zum Kommunikationspartner vorhanden), sind damit im Betrieb der CPU nicht möglich; z. B.

- bei PUT/GET-, FETCH/WRITE- oder FTP-Zugriffen über Kommunikationsmodule
- bei PUT/GET-Zugriffen von anderen S7-CPUs
- bei HMI-Zugriffen, die über PUT/GET-Kommunikation realisiert sind

Wenn Sie den Client-seitigen Zugriff auf CPU-Daten erlauben wollen, d. h. wenn Sie die Kommunikationsdienste der CPU nicht einschränken wollen, dann aktivieren Sie die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben".

Sammelmeldungen bei Security-Ereignissen

Wissenswertes zu Sammelmeldungen bei Security-Ereignissen

Security-Ereignisse im Diagnosepuffer

Die CPU legt Informationen zu wichtigen Vorgängen und Ereignissen im Diagnosepuffer ab.

Die folgenden Security-Ereignisse (Ereignistypen) führen jeweils zu einem Eintrag im Diagnosepuffer der S7-1500:

- Online-Gehen, mit richtigem oder falschem Passwort
- Manipulierte Kommunikationsdaten erkannt
- Manipulierte Daten auf Memory Card erkannt
- Manipulierte Firmware-Update-Datei erkannt
- Geänderte Schutzstufe (Zugriffsschutz) in CPU geladen
- Passwort-Legitimierung eingeschränkt oder freigegeben (per Anweisung oder CPU-Display)
- Online-Zugriff verweigert wegen Überschreitung der Anzahl gleichzeitig möglicher Zugriffe
- Zeitüberschreitung bei Inaktivität einer bestehenden Online-Verbindung
- Anmelden am Webserver mit richtigem oder falschem Passwort
- Sicherung der CPU erstellen (Backup)
- Wiederherstellen der CPU-Projektierung (Restore)
- Im Anlauf:
  - Projekt auf der SIMATIC Memory Card hat sich geändert (die SIMATIC Memory Card ist dieselbe)
  - SIMATIC Memory Card wurde ausgetauscht

Sammelmeldung für Security-Ereignisse

Um zu verhindern, dass der Diagnosepuffer von sehr vielen identischen Security-Ereignissen "überschwemmt" wird, können Sie ab STEP 7 V13 Servicepack 1 parametrieren, dass diese Ereignisse als Sammelmeldung in den Diagnosepuffer eingetragen werden. Pro Interval (Überwachungszeitraum) erzeugt die CPU dann nur noch eine Sammelmeldung je Ereignistyp.

Beispiel: Brute-Force-Attacken

Bei Brute-Force-Attacken versuchen Angreifer, Zugang zur CPU zu erlangen, indem sie systematisch sehr viele der möglichen Passwortkombinationen ausprobieren. Dadurch erhält die CPU innerhalb von Sekunden sehr viele Login-Anfragen und der große Ansturm

identischer Einzel-Einträge im Diagnosepuffer führt zum Verlust wichtiger Diagnoseinformationen. Sammelmeldungen sorgen hier für Abhilfe.

Funktionsweise bei Sammelmeldungen

Die CPU trägt die ersten drei Ereignisse eines Ereignistyps sofort in den Diagnosepuffer ein. Alle danach folgenden Security-Ereignisse dieses Typs ignoriert sie.

Am Ende des Überwachungszeitraums (Intervalls) erzeugt die CPU eine Sammelmeldung, in der sie den Ereignistyp und die Häufigkeit dieses Ereignisses während des abgelaufenen Intervalls einträgt.

Treten diese Security-Ereignisse auch in den folgenden Intervallen auf, so generiert die CPU nur noch jeweils eine Sammelmeldung pro Folgeintervall.

Im Diagnosepuffer hinterlässt eine Attacke folgendes Muster: Drei Einzelmeldungen gefolgt von einer Reihe von Sammelmeldungen. Diese Reihe kann aus zwei, drei oder mehr Sammelmeldungen bestehen, das ist abhängig von der bewählten Überwachungszeit und der Dauer der Attacke.

Da die Meldungen im Diagnosepuffer einen Zeitstempel enthalten, lässt sich die Dauer der Attacke bestimmen.

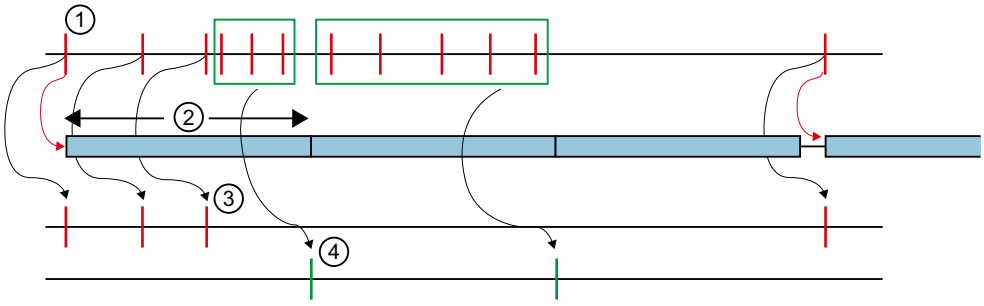

① Security-Ereignisse eines Typs, zum Beispiel Login-Versuche mit ungültigem Passwort.

② Intervall (Überwachungszeitraum): Wenn ein Security-Ereignis eines Typs zum ersten Mal auftritt (oder erneut auftritt), wird die Überwachungszeit gestartet (oder erneut gestartet).

③ Einzelmeldungen: Die drei ersten Security-Ereignisse eines Typs werden sofort in den Diagnosepuffer eingetragen. Ab dem vierten Security-Ereignis dieses Typs erzeugt die CPU nur mehr Sammelmeldungen.

Tritt nach einer Pause von mindestens einem Intervall wieder ein Security-Ereignis dieses Typs auf, trägt die CPU eine Einzelmeldung in den Diagnosepuffer ein und startet den Überwachungszeitraum erneut.

④ Sammelmeldungen: Die CPU erzeugt nach drei Security-Ereignissen nur noch eine Sammelmeldung als Zusammenfassung aller weiteren Security-Ereignisse in diesem Intervall. Treten diese Security-Ereignisse auch in den folgenden Intervallen auf, so generiert die CPU jeweils nur eine Sammelmeldung pro Folgeintervall.

Siehe auch

Sammelmeldung bei Security-Ereignissen projektieren (Seite 1037)

Wissenswertes zu SIMATIC Memory Cards (Seite 959)

Sammelmeldung bei Security-Ereignissen projektieren

Voraussetzungen

- STEP 7 ab V13 Service Pack 1
- S7-1500 CPU mit Firmware ab Version 1.7

Vorgehen

Um Sammelmeldungen bei Security-Ereignissen zu projektieren, gehen Sie in folgendermaßen vor:

1. Klicken Sie in der Netzsicht auf das Symbol der CPU.
   Die Eigenschaften der CPU werden im Inspektorfenster angezeigt.
2. Navigieren Sie zum Bereich "Schutz" > "Security-Ereignis".
3. Klicken Sie auf "Security-Ereignis".
4. Aktivieren Sie die Option "Security-Ereignisse bei hohem Nachrichtenaufkommen zusammenfassen", um Sammelmeldungen bei Security-Ereignissen einzuschalten.
5. Stellen Sie die Dauer eines Intervalls (Überwachungszeitraum) ein, voreingestellt sind 20 Sekunden.

Siehe auch

Wissenswertes zu Sammelmeldungen bei Security-Ereignissen (Seite 1035)

Online-Zugriff auf CPU über Display sperren

Am Display einer S7-1500-CPU können Sie den Zugriff auf eine passwortgeschützte CPU sperren (Vor-Ort-Sperre).

Die Zugriffssperre wirkt nur, wenn der Betriebsartenschalter auf RUN steht.

Die Zugriffssperre wirkt unabhängig vom Passwortschutz, d. h. wenn jemand über ein angeschlossenes Programmiergerät auf die CPU zugreift und das korrekte Passwort eingegeben hat, bleibt der Zugriff auf die CPU verwehrt.

Die Zugriffssperre ist für jede Schutzstufe getrennt am Display einstellbar, d. h. dass z. B. der lesende Zugriff lokal erlaubt ist, der schreibende Zugriff lokal aber nicht erlaubt ist.

Zusätzlich können Sie noch ein Passwort für das Display in den Eigenschaften der CPU parametrieren, so dass der lokale Zugriffsschutz über ein lokales Passwort geschützt ist.

Vorgehen

Um den lokalen Zugriffsschutz für eine S7-1500 CPU am Display einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Display das Menü Einstellungen > Schutz.
2. Bestätigen Sie die Wahl mit "OK" und stellen Sie für jede Schutzstufe ein, ob der Zugriff im RUN erlaubt ist oder nicht:
   Erlauben: Zugriff auf die CPU ist möglich mit Passworteingabe.
   Sperren im RUN: Wenn der Betriebsartenschalter auf RUN steht, kann sich kein Benutzer mehr mit den Rechten dieser Schutzstufe auf der CPU anmelden, obwohl ihm das Passwort dafür bekannt ist. Im STOP ist der Zugriff möglich mit Passworteingabe.

Siehe auch

Einstellmöglichkeiten für den Schutz (Seite 1026)

Security

Einstellmöglichkeiten für Security-Funktionen

Einleitung

Für eine gesicherte Datenübertragung zwischen zwei Geräten spielen digitale Zertifikate eine wichtige Rolle. Bei der Verwendung digitaler Zertifikate kann ein Client nur dann eine gesicherte Verbindung zu einem Server aufbauen, wenn der Server das digitale Zertifikat des Clients akzeptiert und es als vertrauenswürdig einstuft und ihm vertraut. Ebenso muss der Client das Zertifikat des Servers prüfen und ihm vertrauen.

CPUs der S7-1500 ab Firmware V2.0 unterstützen die Verwendung von Zertifikaten.

Gerätespezifischer und projektweiter Zertifikatsmanager

Mit dem lokalen CPU-spezifischen Zertifikatsmanager können für das jeweilige Gerät Zertifikate erzeugt und verwaltet werden, die dann von folgenden Objekten verwendet werden können:

- OPC UA-Server des Geräts
- Webserver des Geräts
- Jedes weitere Systemmerkmal, welches Zertifikate benötigt und von einer Zertifikats-ID referenziert wird (im TIA Portal V14 sind das sOUC und TMAIL).

Hinweis

Jedes Zertifikat im Zertifikatsmanager erhält eine ID für eine eindeutige Referenzierung in bestimmten SDTs. Die Zertifikats-ID wird vom Zertifikatsmanager vergeben, wenn ein Zertifikat erstellt oder importiert wird. Die Zertifikats-ID ist innerhalb des Projekts eindeutig und kann nicht geändert werden.

Zertifikate des lokalen CPU-spezifischen Zertifikatsmanagers beziehen sich alleine auf das Gerät. Wenn Sie Zertifikate projektweit verwenden möchten, müssen Sie im Gerät den globalen Security-Zertifikatsmanager aktivieren.

Im globalen Security-Zertifikatsmanager können Sie weitere Funktionen für die Verwaltung ihrer Zertifikate nutzen:

- Importieren von neuen Zertifikaten und Zertifizierungsstellen.
- Exportieren der im Projekt verwendeten Zertifikate und Zertifizierungsstellen.
- Erneuern von abgelaufenen Zertifikaten und Zertifizierungsstellen.
- Ersetzen bestehender Zertifizierungsstellen.
- Hinzufügen von vertrauenswürdigen Zertifikaten und Zertifizierungsstellen.
- Löschen von manuell importierten Zertifikaten.

Die globalen Security-Einstellungen sind erst sichtbar, wenn diese im Projekt schon für mindestens ein Gerät aktiviert wurden. Zu diesem Zweck steht Ihnen im lokalen CPU-spezifischen Zertifikatsmanager das Kontrollkästchen "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" zur Verfügung. Dieses finden Sie im Inspektorfenster in den allgemeinen Eigenschaften eines Gerätes mit Security-Funktionen unter "Schutz & Security > Zertifikatsmanager". Abhängig von dieser Einstellung wird entweder nur der lokale CPU-spezifische Zertifikatsmanager mit limitierter Funktionalität verwendet oder der globale Security-Zertifikatsmanager.

ACHTUNG
Wenn Sie den globalen Security-Zertifikatsmanager für das Projekt aktivieren, geht der Inhalt des lokalen CPU-spezifischen Zertifikatsmanagers verloren: • Die privaten Schlüssel können nicht wiederhergestellt werden. • Exportieren Sie Ihre Zertifikate, wenn Sie sie im globalen Security Zertifikatsmanager weiterverwenden möchten. • Aktualisieren Sie Ihre Konfiguration oder Ihr Programm, da sich die IDs der Zertifikate ändern können.

Der lokale CPU-spezifische Zertifikatsmanager wirkt nach Aktivierung des globalen Security-Zertifikatsmanager wie ein Zwischenspeicher der CPU-spezifischen globalen Zertifikate. Wenn Sie im lokalen CPU-spezifischen Zertifikatsmanager ein neues Zertifikat erstellen, wird es in den globalen Security-Zertifikatsmanager eingetragen. Zertifikate im globalen Security-Zertifikatsmanager können Sie auch im lokalen CPU-spezifischen Zertifikatsmanager als Zertifikat für die CPU auswählen.

Siehe auch

Zertifikate für Webserver (Seite 1045)

Zertifikate für OPC UA-Server (Seite 1047)

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Grundlagen zu Secure Communication (Seite 323)

Lokaler CPU-spezifischer Zertifikatsmanager

Einleitung

Mit dem lokalen CPU-spezifischen Zertifikatsmanager können Sie ohne Zugriff auf den globalen Security-Zertifikatsmanager Zertifikate für den OPC UA-Server und den Webserver nutzen.

Funktionen des lokalen Zertifikatsmanagers

Der lokale CPU-spezifische Zertifikatsmanager hat gegenüber dem globalen Security-Zertifikatsmanager nur eine limitierte Funktionalität. Über den lokalen Zertifikatsmanager können Sie Zertifikate lediglich erstellen, zuweisen, exportieren oder löschen.

Die Einstellungen für den lokalen CPU-spezifischen Zertifikatsmanager finden Sie im Inspektorfenster in den allgemeinen Eigenschaften des Gerätes unter "Schutz & Security > Zertifikatsmanager":

- Option zur Aktivierung des globalen Zertifikatsmanagers
- Tabelle für Gerätezertifikate: Anzeige und Verwaltung der Zertifikate, die Sie beispielsweise für secure OUC (TLS-Server / TLS-Client), den Webserver und den OPC UA-Server des Gerätes verwenden.
- Tabelle für Zertifikate der Partner-Geräte: Anzeige der Zertifikate, die Sie als vertrauenswürdige Benutzer oder Clients zugewiesen haben.

Gerätezertifikate

Gerätezertifikate gelten für das Gerät und seine dazugehörigen Komponenten. Die hier angezeigten Zertifikate wurden beispielsweise auch für den Webserver oder für OPC UA erstellt und werden hier angezeigt. Ebenso können hier Zertifikate mit den Einstellungen für den Webserver oder für OPC UA erstellt werden, die aber an der entsprechenden Stelle verwendet werden müssen.

Wenn Sie ein neues Zertifikat erstellen, werden im entsprechenden Dialog für das Gerät die folgenden Werte als Voreinstellung eingetragen:

- Zertifikatsinhaber: <Name der CPU>/TLS. Der Zeichensatz ist eingeschränkt, um die Vorgaben nach ASN.1 einzuhalten.
- Art der Signatur: sha256RSA oder sha1RSA, abhängig von der verwendeten Security Policy.
- Gültigkeitsdauer: In den beiden Feldern für die Gültigkeitsdauer werden in Abhängigkeit der Systemzeit Beginn und Ende der Gültigkeitsdauer eingetragen.
- Verwendungszweck: TLS. Die Erweiterungen "KeyUsage" und "ExtendedKeyUsage" für Zertifikate des Standards X.509 V3 werden passend für OPC UA befüllt.
- Alternativer Name des Zertifikatsinhabers (SAN): IP-Adresse der verwendeten Schnittstellen.

Sie können die Voreinstellungen durch eigene Werte überschreiben.

Zertifikate der Partner-Geräte

Bei den Zertifikaten der Partner-Geräte können Sie Gerätezertifikate des lokalen Zertifikatsmanagers auswählen. Gerätezertifikate werden ohne ihren Schlüssel verwendet. Für eine sichere Kommunikation innerhalb des Projekts können Sie auch die Zertifikate anderer CPUs oder CPs auswählen. Die Zertifikate anderer CPUs oder CPs werden ohne privaten Schlüssel auf die CPU geladen. Sofern die Kette von Zertifikaten über den globalen Zertifikatsmanager verfügbar ist, wird die zu einer CPU oder einem CP gehörende Kette von Zertifikaten dennoch nicht unterbrochen und die entsprechende Zertifizierungsstelle und Zwischenzertifikate bleiben der CPU oder dem CP zugeordnet.

Wenn Sie in der Tabelle der Zertifikate der Partner-Geräte ein Zertifikat löschen, wird nur die Verknüpfung des Zertifikats zum globalen Zertifikatsmanager gelöst. Das Zertifikat steht also dem Gerät nicht mehr zur Verfügung, befindet sich aber weiterhin im globalen Zertifikatsmanager. Im Gerät kann also bei Bedarf auch wieder auf das Zertifikat zugegriffen werden, wenn es in der Tabelle als neues Zertifikat aus dem globalen Zertifikatsmanager ausgewählt wird. Soll das Zertifikat für das ganze Projekt gelöscht werden, müssen Sie dies über das Kontextmenü des globalen Zertifikatsmanagers tun.

Siehe auch

Einstellmöglichkeiten für Security-Funktionen (Seite 1038)

Zertifikate erstellen (Seite 1041)

Zertifikate zuweisen (Seite 1043)

Zertifikate exportieren (Seite 1044)

Zertifikate löschen (Seite 1045)

Globaler Security-Zertifikatsmanager (Seite 343)

Zertifikate erstellen

Einleitung

Sie können ein neues Zertifikat für ein Gerät erstellen. Wenn Sie ein neues Zertifikat bei deaktiviertem globalen Security-Zertifikatsmanager erstellen, gilt das Zertifikat nur für dieses Gerät. Bei der ausschließlichen Verwendung des lokalen CPU-spezifischen Zertifikatsmanagers können nur selbstsignierte Zertifikate erstellt werden.

Sie können neue Zertifikate außerdem direkt am Webserver oder bei OPC UA erstellen.

Voraussetzung

Das Gerät muss über Security-Funktionen mit Zertifikatsmanager verfügen. Dies gilt für S7-1500 CPUs ab Firmware V2.0.

Vorgehen

Um ein neues Zertifikat zu erstellen, gehen Sie folgendermaßen vor:

1. Gehen Sie im Inspektorfenster in den Eigenschaften des Gerätes auf "Schutz & Security > Zertifikatsmanager".
2. Klicken Sie in der Tabelle der Zertifikate auf "Hinzufügen". Eine neue Zeile wird in die Tabelle eingetragen.
3. Klicken Sie in die neue Zeile. Die Auswahl für neue Zertifikate öffnet sich.
4. Klicken Sie auf die Schaltfläche "Hinzufügen". Es öffnet sich das entsprechende Dialogfenster, in dem Sie die Daten für das neue Zertifikat eintragen können.
5. Wählen Sie aus, wie das neue Zertifikat signiert werden soll:
   - Selbstsigniert: Wenn Sie die Signatur selber vergeben möchten.
   - Durch eine Zertifizierungsstelle (CA) signiert: Wählen Sie eine Zertifizierungsstelle aus der Klappliste aus.
6. Tragen Sie die Parameter für das Zertifikat ein:
   - Zertifikatsinhaber: Der Zertifikatsinhaber ist das Gerät, für das das Zertifikat gilt.
   - Gültig von: Ab dem genannten Datum ist das Zertifikat gültig.
   - Gültig bis: Nach Ablauf des genannten Datums ist das Zertifikat nicht mehr gültig.
   - Verwendungszweck: Dieser Wert wird bei digitalen Zertifikaten des Standards X.509 V3 in den Erweiterungen "KeyUsage" und "ExtendedKeyUsage" eingetragen.
   - Alternativer Name des Zertifikatsinhabers (SAN).
7. Wenn Sie auf "OK" klicken, wird das neue Zertifikat erstellt und in die Tabelle eingetragen.

Hinweis

Selbstsignierte Zertifikate

Mit einem selbstsignierten Zertifikat hat man bereits eine Verschlüsselung für den Transportweg geschaffen, aber damit hat der Empfänger einer Nachricht keine Bestätigung, ob es sich bei dem genannten Absender wirklich um den echten Absender handelt. Eine Nachricht, die durch eine Zertifizierungsstelle signiert wurde, weist dem Empfänger gegenüber den Absender als "echt" aus. Sie können ein selbstsigniertes Zertifikat aber für das Gerät erstellen, wenn Sie noch auf das signierte Zertifikat von einer Zertifizierungsstelle warten müssen. Über den globalen Security-Zertifikatsmanager in der Projektnavigation können Sie dann nach Erhalt des "offiziellen" Zertifikats durch die Zertifizierungsstelle (CA) das vorübergehend verwendete selbstsignierte Zertifikat entweder ersetzen oder durch die Angabe der Zertifizierungsstelle erneuern. Sie können ein selbstsigniertes Zertifikat aber selbstverständlich auch im Projekt behalten und auf die Verwendung durch ein "offizielles" CA-Zertifikat verzichten.

Ergebnis

Es wird ein neues Zertifikat für das Gerät erstellt. Bei einer S7-1500 CPU wird als Verwendungszweck automatisch "TLS" eingetragen. Dies ist der Fall wenn ein neues selbstsigniertes Zertifikat ausgestellt wird oder wenn über den aktivierten globalen Security-Zertifikatsmanager ein EE-Zertifikat von der Stammzertifizierungsstelle des TIA Portals ausgestellt wird.

Zertifikats-ID

Jedes Zertifikat im Zertifikatsmanager erhält eine ID. Diese ID wird in folgenden Fällen zur Identifikation der Zertifikate verwendet:

- OPC UA-Server-Zertifikat
- Liste der vertrauenswürdigen OPC UA-Clients
- Liste der vertrauenswürdigen OPC UA-Benutzer
- Webserver-Zertifikat
- Secure OUC-Anweisungen

Die Zertifikats-ID wird vom Zertifikatsmanager vergeben, wenn ein Zertifikat erstellt oder importiert wird. Die Zertifikats-IDs sind notwendig, um die Zertifikate eindeutig bestimmten Systemdatentypen zuordnen zu können, beispielsweise dem SDT TCON_IP_v4_Sec. Die Zertifikats-ID ist innerhalb des Projekts eindeutig und kann nicht geändert werden.

Siehe auch

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Zertifikate zuweisen

Einleitung

Sie können den CPUs und CPs im Inspektorfenster verfügbare Zertifikate zuweisen.

Voraussetzung

Das Gerät muss über Security-Funktionen mit Zertifikatsmanager verfügen. Dies gilt für S7-1500 CPUs ab Firmware V2.0.

Vorgehen

Um einem Gerät ein verfügbares Zertifikat zuzuweisen, gehen Sie folgendermaßen vor:

1. Gehen Sie im Inspektorfenster in den Eigenschaften des Gerätes auf "Schutz & Security > Zertifikatsmanager".
2. Klicken Sie in der Tabelle der Zertifikate auf "Hinzufügen". Eine neue Zeile wird in die Tabelle eingetragen.
3. Klicken Sie in die neue Zeile. Die Auswahl für neue Zertifikate öffnet sich.

4. Wählen Sie ein verfügbares Zertifikat aus. Wenn nur der lokale CPU-spezifische Zertifikatsmanager aktiv ist, sehen Sie nicht die global verfügbaren Zertifikate. Global verfügbare Zertifikate können Sie nur sehen und auswählen, wenn der globale Zertifikatsmanager aktiviert ist.
5. Klicken Sie auf den grünen Haken, um die Auswahl in die Tabelle der Zertifikate einzutragen.

Ergebnis

Dem Gerät wurde das ausgewählte Zertifikat zugewiesen.

Siehe auch

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Zertifikate exportieren

Einleitung

Über den Zertifikatsmanager des Gerätes können Sie verfügbare Zertifikate exportieren.

Voraussetzung

Das Gerät muss über Security-Funktionen mit Zertifikatsmanager verfügen. Dies gilt für S7-1500 CPUs ab Firmware V2.0.

Vorgehen

Um ein Zertifikat zu exportieren, gehen Sie folgendermaßen vor:

1. Gehen Sie im Inspektorfenster in den Eigenschaften des Gerätes auf "Schutz & Security > Zertifikatsmanager".
2. Öffnen Sie in der Zeile mit dem zu exportierenden Zertifikat mit Rechtsklick das Kontextmenü.
3. Klicken Sie auf "Zertifikat exportieren".
4. Exportieren Sie das Zertifikat in eines der auswählbaren Dateiformate: CER, DER.

Private Schlüssel können nicht exportiert werden, da diese asymmetrisch verschlüsselt sind und nicht wieder hergestellt werden können.

Ergebnis

Das ausgewählte Zertifikat wurde exportiert und kann nun gesichert oder in anderen Projekten verwendet werden.

Siehe auch

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Zertifikate löschen

Einleitung

Über den Zertifikatsmanager des Gerätes können Sie verfügbare Zertifikate löschen.

Voraussetzung

Das Gerät muss über Security-Funktionen mit Zertifikatsmanager verfügen. Dies gilt für S7-1500 CPUs ab Firmware V2.0.

Vorgehen

Um ein Zertifikat zu löschen, gehen Sie folgendermaßen vor:

1. Gehen Sie im Inspektorfenster in den Eigenschaften des Gerätes auf "Schutz & Security > Zertifikatsmanager".
2. Öffnen Sie in der Zeile mit dem zu löschenden Zertifikat mit Rechtsklick das Kontextmenü.
3. Klicken Sie auf "Zertifikat löschen".

Ergebnis

Das ausgewählte Zertifikat wurde im Gerät gelöscht. Bei aktiviertem globalen Security-Zertifikatsmanager, bleibt das Zertifikat für das Projekt erhalten, wird aber nicht mehr für das Gerät verwendet.

Siehe auch

Lokaler CPU-spezifischer Zertifikatsmanager (Seite 1040)

Zertifikate für Webserver

Einleitung

Sie können für den Webserver im Inspektorfenster unter "Webserver > Server-Security" ein Server-Zertifikat erstellen oder zuweisen.

Lokale Webserver-Zertifikate

Wenn Sie Webserver-Zertifikate lokal erstellen, ohne die globalen Security-Einstellungen für den Zertifikatsmanager zu verwenden, gelten die folgenden Einschränkungen:

- Das Zertifikat gilt nur für die projektierte CPU und steht nicht projektweit zur Verfügung.
- Es werden ausschließlich selbstsignierte Zertifikate erstellt. Selbstsignierte Zertifikate sind nicht in eine "Public Key Infrastructure" (PKI) eingebettet und können keine anderen Zertifikate signieren.
- Private Schlüssel des Zertifikats können nicht exportiert werden.
- Importieren und Erneuern von Zertifikaten sind nur in den globalen Security-Einstellungen für den Zertifikatsmanager möglich.

Voreinstellungen für Webserver-Zertifikate

Wenn Sie ein neues Zertifikat erstellen, werden im entsprechenden Dialog für einen Webserver die folgenden Werte als Voreinstellung eingetragen:

- Zertifikatsinhaber: <Name der CPU>/Webserver. Der Zeichensatz ist eingeschränkt, um die Vorgaben nach ASN.1 einzuhalten.
- Art der Signatur: sha256RSA oder sha1RSA, abhängig von der verwendeten Security Policy.
- Gültigkeitsdauer: In den beiden Feldern für die Gültigkeitsdauer werden in Abhängigkeit der Systemzeit Beginn und Ende der Gültigkeitsdauer eingetragen.
- Verwendungszweck: Webserver. Die Erweiterungen "KeyUsage" und "ExtendedKeyUsage" für Zertifikate des Standards X.509 V3 werden passend für den Webserver befüllt.
- Alternativer Name des Zertifikatsinhabers (SAN): IP-Adresse der verwendeten Schnittstellen von CPU und CP.

Sie können die Voreinstellungen durch eigene Werte überschreiben.

ACHTUNG
SAN mit reservierten IP-Adressen Wenn der alternative Name des Zertifikatsinhabers ein oder mehr reservierte IP-Adressen beinhaltet, stimmt das Zertifikat nicht mit der Richtlinie "Baseline Requirements Certificate Policy for the Issuance and Management of Publicly-Trusted Certificates" überein. Dies kann zu Inkompatibilitäten mit einigen Webbrowsern führen. Um eine Übereinstimmung mit der Richtlinie zu erzielen, sollten Sie die IP-Adressen des alternativen Namens des Zertifikatsinhabers durch Domainnamen ersetzen. Dies erfordert einen DNS-Server im Netzwerk, der die IP-Adressen der Geräte auflöst.

Siehe auch

Einstellmöglichkeiten für Security-Funktionen (Seite 1038)

Beispiel: HTTP over TLS (Seite 340)

Zertifikate für OPC UA-Server

Einleitung

Sie können für OPC UA im Inspektorfenster unter "OPC UA > Server > Security" ein Server-Zertifikat erstellen oder zuweisen.

Lokale Server-Zertifikate für OPC UA

Wenn Sie Server-Zertifikate für OPC UA lokal erstellen, ohne die globalen Security-Einstellungen für den Zertifikatsmanager zu verwenden, gelten die folgenden Einschränkungen:

- Das Zertifikat gilt nur für die projektierte CPU und steht nicht projektweit zur Verfügung.
- Es werden ausschließlich selbstsignierte Zertifikate erstellt. Selbstsignierte Zertifikate sind nicht in eine "Public Key Infrastructure" (PKI) eingebettet und können keine anderen Zertifikate signieren.
- Private Schlüssel des Zertifikats können nicht exportiert werden.
- Importieren und Erneuern von Zertifikaten sind nur in den globalen Security-Einstellungen für den Zertifikatsmanager möglich.

Voreinstellungen für Server-Zertifikate

Wenn Sie ein neues Zertifikat erstellen, werden im entsprechenden Dialog für OPC UA die folgenden Werte als Voreinstellung eingetragen:

- Zertifikatsinhaber: <Name der CPU>/OPCUA. Der Zeichensatz ist eingeschränkt, um die Vorgaben nach ASN.1 einzuhalten.
- Art der Signatur: sha256RSA oder sha1RSA, abhängig von der verwendeten Security Policy.
- Gültigkeitsdauer: In den beiden Feldern für die Gültigkeitsdauer werden in Abhängigkeit der Systemzeit Beginn und Ende der Gültigkeitsdauer eingetragen.
- Verwendungszweck: OPC UA. Die Erweiterungen "KeyUsage" und "ExtendedKeyUsage" für Zertifikate des Standards X.509 V3 werden passend für OPC UA befüllt.
- Alternativer Name des Zertifikatsinhabers (SAN): IP-Adresse der verwendeten Schnittstellen und die URI des OPC UA-Servers.

Sie können die Voreinstellungen durch eigene Werte überschreiben.

Siehe auch

Einstellmöglichkeiten für Security-Funktionen (Seite 1038)

Test, Inbetriebnahme und Routing

Test, Inbetriebnahme, Routing

Aktivieren Sie das Optionskästchen, wenn Sie z. B. PG-Funktionen, die bei der Inbetriebnahme und beim Testen benötigt werden, häufig über diese Schnittstelle durchführen wollen.

Das Aktivieren der Option "Test, Inbetriebnahme, Routing" bewirkt, dass die Schnittstelle zum aktiven Teilnehmer am PROFIBUS wird (d. h. die Schnittstelle ist am Token-Umlauf des PROFIBUS beteiligt). Folgende Funktionen sind dann möglich:

- Programmieren (z. B. Laden)
- Testen (Status/Steuern)
- S7-Routing (I-Slave als Netzübergang)

Die Busumlaufzeit kann sich verlängern. Bei zeitkritischen Anwendungen, und wenn S7-Routing sowie die Client-Funktionalität für die Kommunikation nicht erforderlich sind, sollte diese Option deshalb nicht aktiviert werden.

Wenn das Optionskästchen "Test, Inbetriebnahme, Routing" nicht aktiviert ist, arbeitet die Schnittstelle nur als Server für Kommunikationsdienste.

Server-Funktionalität bedeutet, dass nur der Partner (z. B. PG, OP oder Automatisierungssystem) die Kommunikation initiieren kann. Folgende Besonderheiten sind bei nicht aktiviertem "Test, Inbetriebnahme, Routing" zu beachten:

- S7-Routing ist nicht möglich
- Programmieren und Testen ist möglich, aber langsam (keine Verlängerung der Busumlaufzeit)

SNMP deaktivieren

Das Netzwerk-Management-Protokoll SNMP (Simple Network Management Protocol) ist ein Protokoll, das verschiedene Dienste und Tools zur Erkennung und Diagnose der Netzwerktopologie nutzen. SNMP nutzt das Transportprotokoll UDP. Das SNMP kennt zwei Netzkomponenten, den SNMP-Manager und den SNMP-Client. Der SNMP-Manager überwacht die Netzwerkknoten. Die SNMP-Clients sammeln in den einzelnen Netzwerkknoten verschiedene netzwerkspezifische Informationen und legen Sie in strukturierter Form in der MIB (Management Information Base) ab. Mit Hilfe dieser Daten können verschiedene Dienste und Tools eine ausführliche Netzwerkdiagnose durchführen.

Unter bestimmten Voraussetzungen kann es sinnvoll sein, SNMP zu deaktivieren. Beispiele:

- Die Sicherheitsrichtlinien in Ihrem Netzwerk lassen den Einsatz von SNMP nicht zu.
- Sie verwenden eine eigene SNMP-Lösung, z. B. über eigene Kommunikationanweisungen.

Wenn Sie SNMP für ein Gerät deaktivieren, dann stehen Ihnen verschiedene Möglichen zur Diagnose der Netzwerktopologie (z.B. über das PRONETA-Tool oder über den Webserver der CPU) nicht mehr zur Verfügung.

Voraussetzung

- CPU S7-1200 ab V4.1
- CPU S7-1500 ab V1.8

SNMP deaktivieren

Um SNMP zu deaktivieren, gehen Sie folgendermaßen vor:

1. Legen Sie in STEP 7 einem Datenbaustein an, der die Struktur des Datensatzes $F003_H$ enthält. Die folgende Tabelle zeigt die Struktur des Datensatzes $F003_H$.

Byte	Element	Kodierung	Erläuterung
0-1	BlockID	$F003_H$	Header Die Datensatzlänge wird ab dem Byte 4 "Version" gezählt.
2-3	BlockLenght	8	
4	Version	01_H	
5	Subversion	00_H	
6-7	Reserviert	-	-
8-11	SNMP-Steuerung	Deaktivieren/Aktivieren von SNMP	Wenn Sie SNMP deaktivieren wollen, dann tragen den Wert 0 ein. Wenn Sie SNMP aktivieren wollen, dann tragen Sie den Wert 1 ein.

2. Übertragen Sie den Datensatz $F003_H$ im Anlauf-OB (OB100) mit der Anweisung WRREC (Datensatz Schreiben) an die CPU. Nutzen Sie als Hardware-ID eine integrierte Schnittstelle der CPU.

Beispiel: SNMP für eine CPU 1516-3 PN/DP deaktivieren

Einleitung

Weil die Sicherheitsrichtlinien in Ihrem Netzwerk kein SNMP zulassen, wollen Sie für eine CPU 1516-3 PN/DP das SNMP deaktivieren.

Voraussetzung

- CPU 1516-3 PN/DP mit Firmwarestand V2.0
- STEP 7 V14

SNMP deaktivieren

Um SNMP für die CPU 1516-3 PN/DP zu deaktivieren, gehen Sie folgendermaßen vor:

1. Legen Sie einen Datenbaustein an, der die Struktur des Datensatzes $B071_H$ enthält. Das folgende Bild zeigt den Datenbaustein "Deactivate SNMP". Der Datenbaustein "Deactivate SNMP" enthält neben dem Datensatz $B071_H$ weitere Variablen, die Sie zum Übertragen des Datensatzes verwenden. Die Variable "snmp_deactivate" dient zum Anstoßen des Auftrags für WRREC. Legen Sie diese Variable in den remanenten Speicherbereich, damit der Wert im Anlauf-OB (OB100) auch verfügbar ist.

Deactivate SNMP

	Name	Data type	Start value	R...	A...	W...	V...	Comment
1	Static							
2	snmp_deactivate	Bool	TRUE	☑	☑	☐	☑	Tag for deactivation
3	snmp_record	Struct		☑	☑	☐	☑	Data record 16#B071
4	BlockID	UInt	16#F003	☑	☑	☐	☑	
5	BlockLenght	UInt	8	☑	☑	☐	☑	
6	Version	USInt	1	☑	☑	☐	☑	
7	Subversion	USInt	0	☑	☑	☐	☑	
8	Reserved	UInt	0	☑	☑	☐	☑	
9	SNMPControl	UDInt	0	☑	☑	☐	☑	
10	snmp_done	Bool	false	☐	☑	☐	☑	
11	snmp_error	Bool	false	☐	☑	☐	☑	
12	snmp_status	DWord	16#0	☐	☑	☐	☑	

2. Übertragen Sie den Datensatz $B071_H$ im Anlauf-OB (OB100) mit der Anweisung WRREC (Datensatz Schreiben) an die CPU 1516-3 PN/DP. Im folgenden Programmcode wird der Datensatz $B071_H$ mit der Anweisung WRREC in einer REPEAT UNTIL Schleife übertragen.

```
ORGANIZATION_BLOCK "Startup"
TITLE = "Complete Restart"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
BEGIN
 REPEAT
 "WRREC_DB_1"
  (REQ := "Deactivate
SNMP".snmp_deactivate,
   //Transfer data record
 INDEX:=16#b071,
   //Data record number for SNMP deactivation
 ID:="Local~PROFINET_interface_1",
   //any integrated PROFINET Interface
 DONE => "Deactivate SNMP".snmp_done,
 ERROR => "Deactivate SNMP".snmp_error,
 STATUS => "Deactivate SNMP".snmp_status,
 RECORD := "Deactivate
SNMP".snmp_record)
   //Data record
 UNTIL "Deactivate SNMP".snmp_done OR "Deactivate SNMP".snmp_error
 END_REPEAT;
END_ORGANIZATION_BLOCK
```

SNMP wieder aktivieren

Mit kleinen Änderungen können Sie den oben verwendeten Programmcode zum Aktivieren von SNMP verwenden. Weisen Sie im Anwenderprogramm der Variablen "Deactivate SNMP".snmp_record.SNMPControl den Wert "1" zu:

```
"Deactivate SNMP".snmp_record.SNMPControl := 1;
```

Im nächsten Anlauf der CPU wird SNMP wieder aktiviert.

SNMP deaktivieren - vollständiges Beispiel

Einleitung

Das Beispiel gilt für S7-1500 und S7-1200.

Um den Programmcode in Ihr Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Kopieren Sie den gesamten Programmcode in die Zwischenablage.
2. Öffnen Sie einen Texteditor (z. B. "Editor").
3. Fügen Sie den Inhalt der Zwischenablage in den Texteditor ein mit Strg+V.Speichern Sie das Dokument als scl-Datei ab, z. B. SNMP_DEACT.scl.
4. Öffnen Sie Ihr Projekt in STEP 7.
5. Importieren Sie scl-Datei als externe Quelle.
6. Erzeugen Sie den Anlauf-OB und die Datenbausteine.
7. Rechtsklicken Sie auf die scl-Datei und wählen Sie im Kontextmenü "Baustein aus Quelle generieren".

Programmbeispiel mit WRREC-Aufruf im OB 100

```
DATA_BLOCK "Deactivate_SNMP"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
NON_RETAIN
 VAR
   snmp_deactivate : Bool;
   snmp_record : Struct
     BlockID : UInt;
     BlockLength : UInt;
     "Version" : USInt;
     Subversion : USInt;
     Reserved : UInt;
     SNMPControl : UDInt;
   END_STRUCT;
   snmp_done : Bool;
   snmp_error : Bool;
   snmp_Status : DWord;
 END_VAR

BEGIN
  snmp_deactivate := true;
```

```
  snmp_record.BlockID := 16#f003;
  snmp_record.BlockLength := 8;
  snmp_record."Version" := 1;
END_DATA_BLOCK

ORGANIZATION_BLOCK "Startup"
TITLE = "Complete Restart"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
BEGIN
  REPEAT
    "WRREC_DB_1"(REQ := "Deactivate_SNMP".snmp_deactivate,
                 ID := "Local~PROFINET-Schnittstelle_1",
                 INDEX := 16#b071,
                 DONE => "Deactivate_SNMP".snmp_done,
                 ERROR => "Deactivate_SNMP".snmp_error,
                 STATUS => "Deactivate_SNMP".snmp_Status,
                 RECORD := "Deactivate_SNMP".snmp_record);
   UNTIL "Deactivate_SNMP".snmp_done OR "Deactivate_SNMP".snmp_error
   END_REPEAT;
END_ORGANIZATION_BLOCK

DATA_BLOCK "WRREC_DB_1"
{OriginalPartName := 'WRREC';
 VersionGUID := 'bc169451-58cd-44a3-855b-3f78cc0623c8';
 S7_Optimized_Access := 'TRUE' }
AUTHOR : SIMATIC
FAMILY : DP
NAME : WRREC
VERSION : 1.0
NON_RETAIN
WRREC
BEGIN
END_DATA_BLOCK
```

Taktsynchronisation (zentral)

Taktsynchronität für zentrale Peripherie in S7-1500 projektieren

Bei der Projektierung der Taktsynchronität eines zentral in einer S7-1500-Station gesteckten Moduls, z. B. eines Analogeingabemoduls, gehen Sie prinzipiell genau so vor wie bei der Projektierung eines dezentral gesteckten Moduls:

- In den Eigenschaften der Peripheriebaugruppen müssen Sie den taktsynchronen Betrieb aktivieren, die Adressen des Moduls einem Teilprozessabbild zuordnen sowie dieses Teilprozessabbild dem Taktsynchronalarm-OB zuordnen.
- In des Eigenschaften des Taktsynchronalarm-OBs passen Sie Appliktionszylus und Verzögerungszeit an (Empfehlung: Automatische Einstellung belassen).
- In den Eigenschaften der CPU aktivieren Sie ebenfalls den taktsynchronen Betrieb und passen die Einstellungen für Sendetakt und Ti/To-Werte an.
  Neu ist, dass Sie eine Synchronisationsart wählen können: Hier stellen Sie ein, ob die CPU einen lokalen Sendetakt verwenden soll oder ob der Sendetakt für die zentral gesteckten Module auf den Sendetakt der PROFINET-Schnittstelle X1 synchronisiert werden soll. Im letzteren Fall ist immer der Sendetakt der PROFINET-Schnittstelle führend.

Voraussetzung

- S7-1500 CPU ab Firmware V2.6 (ohne Kompakt-CPUs und ohne S7-1500R/H CPUs)
- STEP 7 (TIA Portal) ab V15.1
- Taktsynchrone Module zentral gesteckt
- Keine PROFINET IO- und PROFIBUS DP-Systeme an zentral gesteckten Kommunikationsprozessoren (CPs) oder Kommunikationsmodulen (CMs)
- Keine Kommunikationsprozessoren (CPs) oder Kommunikationsmodulen (CMs) als I-Devices oder I-Slaves im zentralen Aufbau

Prinzipielles Vorgehen

Peripherie und Taktsynchronalarm-OB projektieren

Um eine taktsynchrone Anbindung von Peripherie und Anwenderprogramm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die S7-1500 CPU in der Netzsicht von STEP 7. Wechseln Sie in die Gerätesicht.
2. Stecken Sie ein Peripheriemodul, das taktsynchron betrieben werden kann (z. B. AI 8 x U/I HS).
3. Navigieren Sie im Inspektorfenster des markierten Peripheriemoduls zum Bereich "E/A-Adressen".

4. Nehmen Sie im Bereich E/A-Adressen folgende Einstellungen vor:
   - Aktivieren Sie die Option "Taktsynchroner Betrieb".
   - Wählen Sie ein Teilprozessabbild aus, z. B. Teilprozessabbild 1.
   - Klicken Sie auf die Klappliste "Organisationsbaustein" und klicken Sie auf die Schaltfläche "Objekt hinzufügen" oder wählen Sie einen bereits vorhandenen OB aus. Ein Dialog zur Auswahl von Organisationsbausteinen wird geöffnet.
   - Wählen Sie den OB "Synchronous Cycle". Bestätigen Sie die Auswahl mit "OK". Bei automatischer Nummernvergabe wird der OB 61 erzeugt und geöffnet. Im Inspektorfenster ist der Bereich "Taktsynchronität" markiert und Sie können unmittelbar mit der Einstellung von Applikationszyklus und Verzögerungszeit fortfahren sowie im Anweisungsteil mit der Programmierung des OB beginnen.
5. Wenn die taktsynchrone Anlage aufgrund der verwendeten Module nur mit einem bestimmten Sendetakt arbeitet (z. B. 1 ms), die Prozesswerte jedoch schneller abgetastet werden müssen, nutzen Sie die Funktion Oversampling. Bei Modulen, die diese Funktion unterstützen, tasten Sie die Prozesswerte für die Kanäle z. B. eines Analogeingabemoduls in kürzeren Intervallen ab.
6. Fügen Sie weitere Peripheriemodule nach Bedarf in die Gerätesicht ein. Passen Sie die Konfiguration und die taktsynchronen Einstellungen an.

CPU projektieren

1. Markieren Sie die CPU in der Gerätesicht von STEP 7, um die Einstellungen für die Taktsynchronisation zu kontrollieren
2. Navigieren Sie im Inspektorfenster der markierten CPU zum Bereich "Taktsynchronisation".
3. Im Bereich "Taktsynchronisation für lokale Module" kontrollieren Sie:
   - ob taktsynchroner Betrieb angewählt ist
   - ob als Synchronisationsart "Lokaler Sendetakt" gewählt ist, falls die Taktsynchronisation nur für die zentral gesteckten Module greifen soll
     bzw. ob als Synchronisationsart "Sendetakt der PROFINET-Schnittstelle verwenden" gewählt ist, falls der Sendetakt für die zentral gesteckten Module sich auf Sendetakt der PROFINET-Schnittstelle X1 synchronisieren soll.

   Bei Bedarf können Sie den Sendetakt und die Zeiten T_i/T_o zum taktsynchronen Einlesen/Ausgeben der Daten anpassen.
4. Im Bereich "Detailübersicht" sehen Sie alle Module der Konfiguration, die Sie taktsynchron betreiben können. Wählen Sie entsprechend Taktsynchronität für die gewünschten Module an oder ab.

Zeitlicher Ablauf der Synchronisation im zentralen Aufbau

Vom Einlesen der Eingangsdaten bis zum Ausgeben der Ausgangsdaten

Im Folgenden ist der prinzipielle zeitliche Ablauf aller an der Synchronisation beteiligten Komponenten erläutert:

① Messwert-Erfassung im Prozess

② Taktsynchrones Einlesen der Messwerte (Eingangsdaten) von der zentralen Peripherie

③ Transport der Eingangsdaten zur CPU

④ Weiterverarbeitung der Eingangsdaten in der taktsynchronen Applikation der CPU (OB 61)

⑤ Transport der Ausgangsdaten zur zentralen Peripherie

⑥ Taktsynchrones Ausgeben der Ausgangsdaten

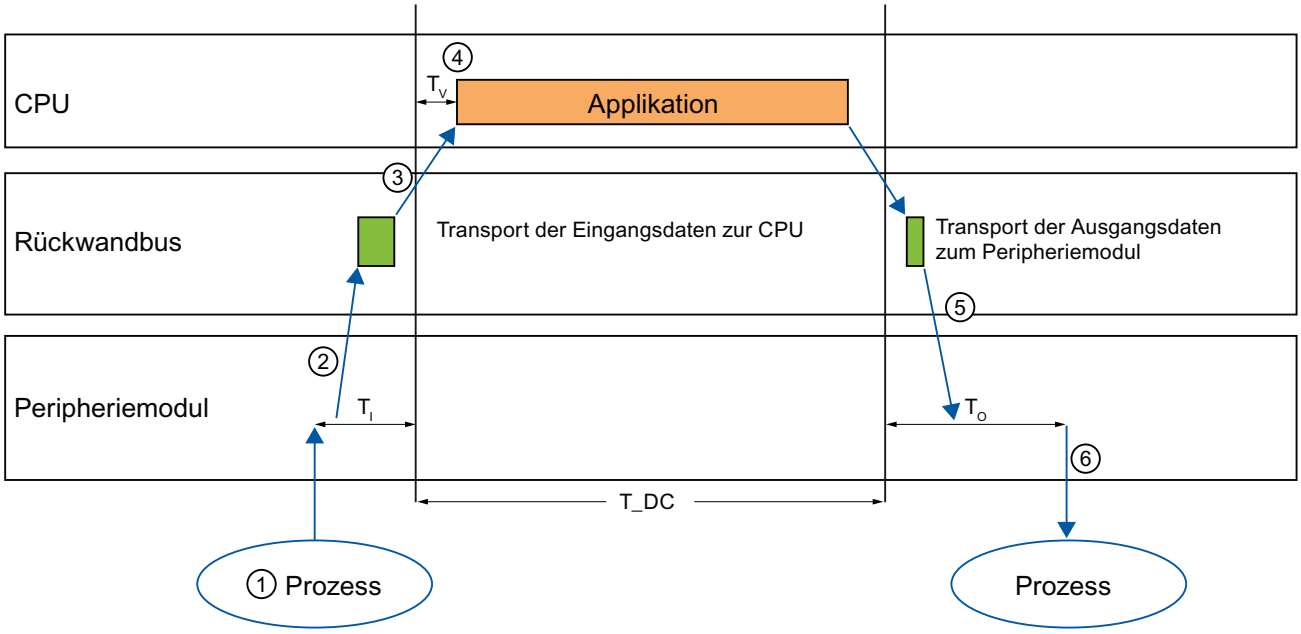

Damit zum jeweils nächsten Beginn des Zyklus alle Eingangsdaten zum Transport zur CPU bereitstehen, wird der Peripherie-Einlesezyklus um eine Vorlaufzeit T_I früher gestartet. Das T_I ist das "Blitzlicht" für die Eingänge. Danach werden alle synchronisierten Eingänge eingelesen. T_I ist notwendig, um Analog-Digital-Umwandlung und Ähnliches zu kompensieren. Die Vorlaufzeit T_I kann von STEP 7 oder von Ihnen projektiert werden. Lassen Sie die Vorlaufzeit T_I von STEP 7 automatisch vergeben. STEP 7 stellt mit der Standardeinstellung sicher, dass eine gemeinsame, kleinstmögliche T_I eingestellt wird.

Der Rückwandbus transportiert die Eingangsdaten zur CPU. Die Applikation wird synchronisiert zum Zyklus gestartet. D. h., der Taktsynchronalarm-OB wird nach einer parametrierbaren Verzögerungszeit T_V aufgerufen. Das Anwenderprogramm im Taktsynchronalarm-OB bestimmt die Prozessreaktion und stellt die Ausgangsdaten rechtzeitig bis zum Beginn des nächsten Datenzyklus bereit. Die Länge des Datenzyklus (Sendetakt) wird immer von Ihnen projektiert.

Die Daten werden innerhalb der Zeit T_O:

- über den Rückwandbus zum Peripheriemodul transportiert
- im Peripheriemodul verarbeitet, z. B. in einen Analogwert gewandelt

Die Daten werden nach Ablauf von T_O an den Prozess ausgegeben.

Die Zeit T_O kann von STEP 7 oder von Ihnen projektiert werden. Lassen Sie die Vorlaufzeit T_O von STEP 7 automatisch vergeben. STEP 7 berechnet automatisch eine gemeinsame, kleinstmögliche T_O.

Organisationsbausteine

OB-Startinformation

Struktur der OB-Startinformation

Bei der Parametrierung der Eigenschaften eines OB für S7-1500-CPUs legen Sie auch die Struktur der OB-Startinformation fest:

- wie bei S7-300- und S7-400-CPUs
- optimierte Startinformation

OB-Startinformation wie bei S7-300- und S7-400-CPUs

Die OB-Startinformation ist Bestandteil der Lokaldaten und hat dieselbe Struktur wie bei S7-300- und S7-400-CPUs.

Optimierte Startinformation

Die optimierte Startinformation ist ein Satz von Eingangsvariablen.

Sie stellt nur diejenige Information zur Verfügung, die zur Identifikation des OB-Startereignisses notwendig ist.

EIne semantisch gleiche Information wird bei verschiedenen OBs unter demselben Variablennamen innerhalb der optimierten Startinformation angezeigt.

Beschreibung aller Typen von Organisationsbausteinen

Anlauf-OBs

Beschreibung

Das Betriebssystem ruft jeden Anlauf-OB beim Übergang vom Betriebszustand "STOP" nach "RUN" einmalig auf. Wenn mehrere Anlauf-OBs vorhanden sind, werden sie in der Reihenfolge ihrer OB-Nummern aufgerufen, beginnend mit dem OB mit der kleinsten OB-Nummer.

Die Gesamtheit aller Anlauf-OBs wird als Anlaufprogramm bezeichnet.

Im Anlaufprogramm können Sie Voreinstellungen für Ihr zyklisches Programm vornehmen.

Hier haben alle Werte des Prozessabbilds der Eingänge den Wert Null. Für die Ausführung des Anlaufprogramms besteht keine zeitliche Begrenzung. Die Verwendung von zeit- oder alarmgesteuerten Organisationsbausteinen ist nicht möglich.

Nach der vollständigen Bearbeitung des Anlaufprogramms liest das Betriebssystem das Prozessabbild der Eingänge ein und startet das zyklische Programm.

Struktur der Startinformation

Ein Anlauf-OB hat die folgende Startinformation:

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LostRetentive	BOOL	=TRUE, falls die Inhalte remanenter Datenbereiche verloren gegangen sind
LostRTC	BOOL	=TRUE, falls die Uhrzeit der Echtzeituhr verloren gegangen ist

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Zyklus-OBs

Beschreibung

Die wesentliche Grundlage des Anwenderprogramms ist das zyklische Programm. Seine zugehörigen Programmteile werden zyklisch bearbeitet: Ist ihre Bearbeitung beendet, beginnt das Betriebssystem erneut mit ihrer Bearbeitung.

Zum zyklischen Programm gehören ein oder mehrere Zyklus-OBs. Diese haben die Priorität 1. Das ist die niedrigste Priorität aller OBs. Das zyklische Programm kann also von Ereignissen jeder anderen Ereignisklasse unterbrochen werden.

Wenn Sie mehrere Zyklus-OBs angelegt haben, werden diese nacheinander aufgerufen, und zwar in der Reihenfolge ihrer OB-Nummern. Der Zyklus-OB mit der kleinsten OB-Nummer wird zuerst aufgerufen.

Die folgenden Ereignisse starten das zyklische Programm:

- Ende der Bearbeitung des Anlaufs
- Ende der Bearbeitung des vorherigen Durchlaufs des zyklischen Programms

Nach Beendigung des zyklischen Programms aktualisiert das Betriebssystem die Prozessabbilder wie folgt:

1. Es schreibt die Werte aus dem Prozessabbild der Ausgänge in die Ausgabebaugruppen.
2. Es liest die Eingänge an den Eingabebaugruppen und überträgt sie in das Prozessabbild der Eingänge.

Zykluszeitüberwachung

Unter der Zykluszeit versteht man die Laufzeit des zyklischen Programms einschließlich der Laufzeit aller eingeschachtelten Programmteile höherpriorer OBs. Wenn Sie mehrere Zyklus-OBs angelegt haben, trägt also jeder Zyklus-OB zur Zykluszeit bei.

Das Betriebssystem überwacht, ob die Zykluszeit kleiner bleibt als die parametrierte Maximale Zykluszeit. Falls die Zykluszeit die Maximale Zykluszeit. überschreitet, wird der OB 80 aufgerufen (Startereignis W#16#0002:3501). Sie können dies verhindern, indem Sie an geeigneter Stelle Ihres Programms die Zeitüberwachung neu starten, und zwar durch Aufruf der Anweisung RE_TRIGR. Wenn die maximale Zykluszeit innerhalb eines Zyklus ein zweites Mal überschritten wird (ohne Neustart der Zeitüberwachung), geht die CPU in STOP (Ereignis W#16#0002:3500).

Minimale Zykluszeit

Neben der Überwachung der Maximalen Zykluszeit kann die Einhaltung einer Minimalen Zykluszeit garantiert werden. Das Betriebssystem verzögert den Beginn eines neuen Zyklus so lange, bis die Minimale Zykluszeit erreicht ist.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - SCAN_1=B#16#01 beim Erstaufruf und B#16#03 bei allen weiteren Aufrufen
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB bei: • Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
Remanence	BOOL	=TRUE, falls remanente Daten verfügbar sind

Siehe auch

Ereignisse und OBs (Seite 980)

Wertstatus auswerten (Seite 972)

OB-Startinformation (Seite 1056)

Uhrzeitalarm-OBs

Beschreibung

Die S7-1500-CPUs stellen Uhrzeitalarm-OBs zur Verfügung, die wie folgt gestartet werden können:

- Einmalig zu einem bestimmten Zeitpunkt (Datum mit Uhrzeit)
- Periodisch (Dabei sind die folgenden Zeitintervalle möglich: minütlich, stündlich, täglich, wöchentlich, monatlich, am Monatsende, jährlich)

Start eines Uhrzeitalarm-OB

Damit die CPU einen Uhrzeitalarm-OB starten kann, müssen die folgenden Voraussetzungen erfüllt sein:

- Sie müssen den zugehörigen Uhrzeitalarm vorher gestellt und aktiviert haben. (Unter dem Stellen eines Uhrzeitalarms versteht man die Vorgabe seines Startzeitpunkts und seiner Periode.)
- Sie müssen den Uhrzeitalarm-OB in Ihre CPU geladen haben.

Um einen Uhrzeitalarm zu stellen und zu aktivieren, haben Sie die folgenden drei Möglichkeiten:

- Sie stellen und aktivieren den Uhrzeitalarm mit STEP 7.
- Sie stellen den Uhrzeitalarm mit STEP 7 und aktivieren ihn durch Aufruf der Anweisung ACT_TINT aus Ihrem Programm heraus.
- Sie stellen den Uhrzeitalarm durch Aufruf der Anweisung SET_TINTL und aktivieren ihn durch Aufruf der Anweisung ACT_TINT jeweils aus Ihrem Programm heraus.

Regeln für Uhrzeitalarme

- Wenn Sie einen Uhrzeitalarm so stellen, dass der zugehörige OB einmalig bearbeitet werden soll, darf der Startzeitpunkt nicht in der Vergangenheit liegen (bezogen auf die Echtzeituhr der CPU).
- Wenn Sie einen Uhrzeitalarm so stellen, dass der zugehörige OB periodisch bearbeitet werden soll, der Startzeitpunkt aber in der Vergangenheit liegt, dann wird der Uhrzeitalarm-OB bei dem nächsten fälligen Zeitpunkt nach der aktuellen Uhrzeit bearbeitet.
- Das Datum periodischer Uhrzeitalarme muss einem realen Datum entsprechen. Die monatliche Wiederholung eines Uhrzeitalarm-OB mit Startdatum 31.1. beispielsweise ist also nicht möglich. In diesem Fall würde der OB nur in den Monaten gestartet, die 31 Tage haben.
- Ein Uhrzeitalarm, der während des Anlaufs aktiviert wird, wird erst nach Beendigung des Anlaufs bearbeitet.

- Ein Anlauf löscht alle Uhrzeitalarme, die mittels Anweisung im Anwenderprogramm gestellt und aktiviert wurden.
- Bei einem Übergang von HALT nach RUN prüft das Betriebssystem, ob abgelaufene Uhrzeitalarme vorhanden sind. Falls ja, ruft es den Zeitfehler-OB auf. Wenn Sie für ihn die Startinformation wie bei S7-300- und S7-400-CPUs gewählt haben, trägt es für die OBs 10 bis 17 (und nur für diese) dort ein, ob Uhrzeitalarme verpasst wurden. Nach Abarbeitung des OB 80 wird der jeweils erste ausgefallene OB nachgeholt.

Verhalten der Uhrzeitalarme bei Vor- bzw. Zurückstellen der Uhrzeit

Das Verhalten der Uhrzeitalarme hängt davon ab, um welche Zeitdifferenz die Uhrzeit vor- bzw. zurückgestellt wird. Das wird in der folgenden Tabelle näher erläutert.

Uhrzeitänderung	Auswirkung auf Uhrzeitalarm
< 20 s	• Beim Vorstellen der Uhr: Jeder übersprungene Uhrzeitalarm wird nachgeholt. Die Startinfo-Variable "CaughtUp" wird gesetzt. OB 80 wird nicht aufgerufen. • Beim Zurückstellen der Uhr: Ein Uhrzeitalarm, der im Bereich der zurückgestellten Uhrzeit bereits ausgeführt wurde oder der gerade aktiv ist, wird nicht wiederholt.
> = 20 s	• Beim Vorstellen der Uhr: Wenn ein oder mehrere Uhrzeitalarme übersprungen wurden, wird OB 80 aufgerufen. Die Startinfo liefert für jede Priorität die Information, ob mindestens ein Uhrzeitalarm übersprungen wurde. Wenn Sie den Uhrzeitalarm im OB 80 nicht löschen, wird der erste übersprungene Uhrzeitalarm nachgeholt, und die Startinfo-Variable "CaughtUp" wird gesetzt. • Beim Zurückstellen der Uhr: Die Uhrzeitalarme im Bereich der zurückgestellten Uhrzeit werden wiederholt. Wenn die Zeitkorrektur größer als die Periode eines Uhrzeitalarms ist, berechnet das Betriebssystem die Uhrzeit des ersten zu wiederholenden Uhrzeitalarms. Bei einem wiederholten Uhrzeitalarm ist die Startinfo-Variable "SecondTime" gesetzt.

Uhrzeitänderung	Auswirkung auf Uhrzeitalarm
Wechsel von Winter- nach Sommerzeit und umgekehrt	Der Wechsel wirkt sich nur dann aus, wenn Sie die Lokalzeit verwenden. • Beim Wechsel von Winter- nach Sommerzeit wird ein übersprungener Uhrzeitalarm nachgeholt, und die Startinfo-Variable "CaughtUp" wird gesetzt. Der OB 80 wird nicht aufgerufen. • Beim Wechsel von Sommer- auf Winterzeit: werden die Uhrzeitalarme im Bereich der zurückgestellten Uhrzeit wiederholt. Bei einem wiederholten Uhrzeitalarm ist die Startinfo-Variable "SecondTime" gesetzt.
Änderung der Zeitzone (hat Änderung der Lokalzeit zur Folge)	Die Startuhrzeit der aktiven Uhrzeitalarme wird neu berechnet. • Beim Vorstellen der Lokalzeit wird ein übersprungener Uhrzeitalarm nachgeholt, und die Startinfo-Variable "CaughtUp" wird gesetzt. Der OB 80 wird nicht aufgerufen. • Beim Zurückstellen der Lokalzeit werden bereits ausgeführte Uhrzeitalarme ggf. wiederholt. Bei einem wiederholten Uhrzeitalarm ist die Startinfo-Variable "SecondTime" gesetzt.

Zeitüberwachung

Aufgrund der Zeitüberwachung durch das Betriebssystem können beim Arbeiten mit Uhrzeitalarmen die folgenden Ereignisse auftreten:

- W#16#0002:3504
  Die Uhrzeit wurde durch Winter-/Sommerzeit-Umstellung vorgestellt. Dabei wurde der Startzeitpunkt von Uhrzeitalarmen übersprungen. Der OB 80 wird aufgerufen.
- W#16#0002:3505
  Die Uhrzeit wurde vorgestellt. Dabei wurde der Startzeitpunkt von Uhrzeitalarmen übersprungen. Der OB 80 wird aufgerufen.
- W#16#0002:3507
  Pufferüberlauf für OB-Startereignisse. Es erfolgt ein Diagnosepuffereintrag.
- W#16#0002:3502
  Anzahl anstehender OB-Startereignisse überschreitet eine vorgegebene Grenze Der OB 80 wird aufgerufen.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#1F, wennn die OB-Nummer > 122 ist
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
CaughtUp	BOOL	=TRUE, wenn ein OB-Aufruf wegen vorgestellter Uhrzeit nachgeholt wurde
SecondTime	BOOL	=TRUE, wenn ein OB wegen zurückgestellter Uhrzeit ein zweites Mal aufgerufen wurde. Hinweis: SecondTime wird bei zurückgestellter Uhrzeit nur ein einziges Mal gesetzt.

Siehe auch

Uhrzeitalarm-OBs parametrieren (Seite 1088)

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Statusalarm-OB

Beschreibung

Das Betriebssystem der S7-1500-CPU ruft den Statusalarm-OB auf, wenn es einen Statusalarm empfangen hat. Dies kann der Fall sein, wenn ein Modul eines Slaves seinen Betriebszustand wechselt, z. B. von "RUN" nach "STOP". Genauere Informationen zu den Ereignissen, die einen Statusalarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm: • PROFIBUS: – Bit 0 bis 1: Alarm-Specifier – Bit 2: Add_Ack – Bit 3 bis 7: Seq-Nr. – Bit 8 bis 15: reserviert • Zentral und PNIO: – Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) – Bits 11 bis 15: 0

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Update-Alarm-OB

Beschreibung

Das Betriebssystem der S7-1500-CPU ruft den Update-Alarm-OB auf, wenn es einen Update-Alarm empfangen hat. Dies kann der Fall sein, wenn Sie an einem Steckplatz eines Slaves oder Devices eine Parameteränderung durchgeführt haben. Genauere Informationen zu den Ereignissen, die einen Update-Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm: • PROFIBUS: – Bit 0 bis 1: Alarm-Specifier – Bit 2: Add_Ack – Bit 3 bis 7: Seq-Nr. – Bit 8 bis 15: reserviert • Zentral und PNIO: – Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) – Bits 11 bis 15: 0

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

OB für hersteller- bzw. profilspezifischen Alarm

Beschreibung

Das Betriebssystem der S7-1500-CPU ruft den OB57 auf, wenn es einen hersteller- oder profilspezifischen Alarm empfangen hat. Genauere Informationen zu den Ereignissen, die einen solchen Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#67
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm: • PROFIBUS: – Bit 0 bis 1: Alarm-Specifier – Bit 2: Add_Ack – Bit 3 bis 7: Seq-Nr. – Bit 8 bis 15: reserviert • Zentral und PNIO: – Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) – Bits 11 bis 15: 0

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Verzögerungsalarm-OBs

Beschreibung

Ein Verzögerungsalarm-OB wird nach Ablauf einer parametrierbaren Verzögerungszeit vom Betriebssystem aufgerufen. Das Ablaufen der Verzögerungszeit beginnt mit dem Aufruf der Anweisung "SRT_DINT" (Das ist der Start des zugehörigen Verzögerungsalarms).

Damit ein Verzögerungsalarm-OB vom Betriebssystem aufgerufen werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Verzögerungsalarm muss gestartet sein durch Aufruf der Anweisung "SRT_DINT".
- Der Verzögerungsalarm-OB darf nicht per Projektierung abgewählt sein.

Der Verzögerungsalarm-OB muss in der CPU vorhanden sein.

Verzögerungsalarme mit Anweisungen beeinflussen

Sie können einen bereits gestarteten Verzögerungsalarm, dessen zugehöriger Verzögerungsalarm-OB vom Betriebssystem noch nicht aufgerufen wurde, mit der Anweisung "CAN_DINT" wieder stornieren bzw. mit der Anweisung "SRT_DINT" erneut starten (und ihm so eine andere Verzögerungszeit zuweisen). Außerdem können Sie mit der Anweisung "QRY_DINT" den derzeitigen Zustand eines Verzögerungsalarms abfragen.

Sie können Verzögerungsalarme mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT" sperren und wieder freigeben.

Hinweis

Wenn Sie nach der Ausführung von "SRT_DINT" diesen Alarm mit "DIS_AIRT" sperren, wird er erst nach Freigabe mit "EN_AIRT" weiter bearbeitet, d. h. die Verzögerungszeit verlängert sich entsprechend.

Aufruf von Fehler-OBs im Zusammenhang mit Verzögerungsalarm-OBs

Wenn die Verzögerungszeit eines Verzögerungsalarms abgelaufen ist, bevor die Bearbeitung des zugehörigen Verzögerungsalarm-OB beendet ist, erzeugt das Betriebssystem einen Zeitfehler (Aufruf des OB80).

Bearbeitung von Verzögerungsaalrmen je nach Betriebszustand

Der Aufruf von "SRT_DINT" und damit das Ablaufen der Verzögerungszeit kann sowohl im Anlauf als auch im RUN erfolgen. Verzögerungsalarm-OBs hingegen werden nur im Betriebszustand RUN bearbeitet. Wenn also der Start eines Verzögerungsalarms in einem Anlauf-OB erfolgt und die Verzögerungszeit abgelaufen ist, bevor die CPU im RUN ist, verzögert sich der Aufruf des zugehörigen Verzögerungsalarm-OB so lange, bis die CPU im RUN ist.

Ein Kaltstart und ein Warmstart löschen jedes Startereignis eines Verzögerungsalarm-OB.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#2F, wennn die OB-Nummer > 122 ist
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
Sign	WORD	Anwenderkennung: Eingangsparameter SIGN aus dem Aufruf der Anweisung "SRT_DINT"

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Weckalarm-OBs

Beschreibung

Mit Hilfe von Weckalarm-OB können Sie Programmteile nach äquidistanten Zeitabschnitten starten.

Die Startzeitpunkte eines Weckalarm-OB ergeben sich wie folgt aus seinem Zeitintervall und seiner Phasenverschiebung:

Startzeitpunkte = n * Zeitintervall + Phasenverschiebung (n = 0, 1, 2 ...)

- Das Zeitintervall ist die Zeitspanne zwischen zwei Aufrufen. Es ist ein ganzzahliges Vielfaches des Grundtakts von 1 µs.
- Die Phasenverschiebung ist diejenige Zeitspanne, um die die Startzeitpunkte gegenüber den Vielfachen des Zeitintervalls verschoben sind.
  Eine Phasenverschiebung kann bei der Benutzung mehrerer Weckalarm-OBs sinnvoll sein. Wenn deren Zeitintervalle nämlich gemeinsame Vielfache haben, können Sie durch Verwendung einer Phasenverschiebung gleichzeitige Startzeitpunkte verhindern.

Beispiel für die Verwendung einer Phasenverschiebung

Sie benutzen in Ihrem Programm zwei Weckalarm-OBs: einen mit einem Zeitintervall von 20 ms und einen mit einem Zeitintervall von 100 ms. Um zu gewährleisten, dass diese bei ganzzahligen Vielfachen von 100 ms nicht gleichzeitig aufgerufen werden, verwenden Sie eine Phasenverschiebung.

Aufruf von Fehler-OBs im Zusammenhang mit Weckalarm-OBs

Die Laufzeit eines Weckalarm-OB muss deutlich kleiner sein als sein Zeitintervall. Andernfalls kann es passieren, dass das Zeitintervall eines Weckalarms abgelaufen ist, der zugehörige Weckalarm-OB aber noch bearbeitet wird. In diesem Fall erzeugt das Betriebssystem einen Zeitfehler (Aufruf des OB80). Anschließend wird der Fehler verursachende Weckalarm nachgeholt oder verworfen. Für jede Priorität kann allerdings nur ein Weckalarm-OB nachgeholt werden.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#3F, wennn die OB-Nummer > 122 ist und die Phasenverschiebung und der Zeittakt in ms angegeben werden
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB • Beim Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
Event_count	INT	Anzahl der verworfenen Startereignisse seit dem letzten Start dieses OB

Siehe auch

Weckalarm-OBs parametrieren (Seite 1090)

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Prozessalarm-OBs

Beschreibung

Mit Hilfe von Prozessalarm-OBs reagieren Sie auf diejenigen Ereignisse von prozessalarmfähigen Modulen, die einen Prozessalarm auslösen. Welche Ereignisse das sind, legen Sie mittels Projektierung fest.

Hinweis

Verwendung von Prozessalarmen

Verwenden Sie Prozessalarme nach Möglichkeit nur, um auf sporadisch auftretende Ereignisse von prozessalarmfähigen Modulen zu reagieren.

Die Verwendung von Prozessalarmen zur Reaktion auf häufig oder sehr häufig auftretende wiederkehrende Ereignisse ist nicht sinnvoll und kann unter ungünstigen Umständen zu Zeitfehlern auf Ihrer CPU führen.

Sie können jedem Ereignis, das einen Prozessalarm auslöst, nur genau einen Prozessalarm-OB zuordnen. Einem Prozessalarm-OB hingegen können mehrere Ereignisse zugeordnet sein (im Extremfall alle Prozessalarm auslösenden Ereignisse eines Moduls).

Bei S7-1500-Modulen können einzelne Kanäle Prozessalarme auslösen (z. B. ein Binäreingang bei steigender Flanke).

Funktionsweise eines Prozessalarm-OB

Nach Auslösen eines Prozessalarms durch ein Modul oder Submodul ermittelt das Betriebssystem das zugehörige Ereignis und den zugehörigen Prozessalarm-OB. Wenn der momentan aktive OB eine kleinere Priorität hat als dieser Prozessalarm-OB, wird der Prozessalarm-OB gestartet. Andernfalls wird der Prozessalarm-OB in die seiner Priorität zugehörige Warteschlange eingereiht. Die Quittierung des Prozessalarms erfolgt nach Beendigung des zugehörigen Prozessalarm-OB.

Wenn in der Zeit zwischen der Identifikation und der Quittierung eines Prozessalarms auf demselben Modul oder Submodul erneut ein Prozessereignis auftritt, dann gilt:

- Wenn das Ereignis bei dem Kanal auftritt, der den aktuellen Prozessalarm ausgelöst hat, geht der zugehörige Prozessalarm verloren. Ein erneuter Prozessalarm kann von diesem Kanal erst dann ausgelöst werden, wenn der aktuelle Prozessalarm quittiert ist.
- Wenn das Ereignis bei einem anderen Kanal auftritt, wird ein Prozessalarm ausgelöst.

Für Dezentrale Peripherie kann es darüber hinaus gehende Einschränkungen geben, z. B. dass ein Modul oder Submodul nur genau einen Alarm eines Alarmtyps auslösen kann, bis der aktuelle Alarm quittiert ist.

Prozessalarm-OBs werden nur im Betriebszustand RUN der CPU aufgerufen.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs: Die Variablen USI, IChannel und EventType (Erläuterung erfolgt bei der Optimierten Startinformation) werden nach "OB40_POINT_ADDR" kopiert.
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#41
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	HW-Kennung des Moduls, das den Prozessalarm auslöst
USI	WORD	Kennung für zukünftige Erweiterungen (nicht anwenderrelevant)
IChannel	USINT	Nummer des Kanals, der den Prozessalarm auslöst
EventType	BYTE	Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke) Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen.
PointAddr (ab Version V14.0 des TIA Portals)	DWORD	• Bei Digitalbaugruppen: Bitfeld mit denjenigen Eingängen auf der Baugruppe, die den Prozessalarm ausgelöst haben. Die Zuordnung, welches Bit von Point_Addr zu welchem Kanal der Baugruppe gehört, ist der Beschreibung der jeweiligen Baugruppe zu entnehmen. • Bei Analogbaugruppen: Bitfeld mit der Information, welcher Kanal welche Grenze überschritten hat. Den genauen Aufbau entnehmen Sie bitte der Baugruppenbeschreibung. • Bei CPs oder IMs: Alarmzustand der Baugruppe (nicht anwenderrelevant)

Hinweis

USI hat den Wert 0xFFFF

Falls Sie das betroffene Modul in HWKonfig mit GSDML-Mitteln projektiert haben, hat USI den Wert 0xFFFF. In diesem Fall enthalten IChannel und EventType keine gültigen Werte. Um das Alarm auslösende Ereignis zu ermitteln, gehen Sie wie folgt vor:

Entweder Sie verwenden die Struktur der Startinformation wie bei S7-300- und S7-400-CPUs.

Oder Sie verwenden die optimierte Startinformation und rufen im Prozessalarm-OB die Anweisung RALRM auf.

Öffnen, Übersetzen und Laden von Prozessalarm-OBs

Da die optimierte Startinformation der Prozessalarm-OBs ab Version 14.0 des TIA Portals eine weitere Variable enthält, kann es zu Konstellationen kommen, in denen das Öffnen, Übersetzen und Laden der Prozessalarm-OBs in die CPU nicht mehr möglich ist. Diese Konstellationen sind im Folgenden aufgelistet:

- Sie kopieren im Engineering System einen Prozessalarm-OB mit neuer Schnittstelle in eine S7-1500-CPU mit Firmware-Version < V2.0 oder in eine S7-1200-CPU. Der OB kann danach nicht mehr geöffnet, übersetzt und geladen werden.
- Sie ersetzen im Engineering System eine S7-1500-CPU mit Firmware-Version >= V2.0, die einen Prozessalarm-OB mit neuer Schnittstelle enthält, durch eine S7-1500-CPU mit Firmware-Version < V2.0. Der OB kann danach nicht mehr geöffnet, übersetzt und geladen werden.

Keine Schwierigkeiten beim Öffnen, Übersetzen und Laden gibt es hingegen in den folgenden Konstellationen:

- Sie kopieren im Engineering System einen Prozessalarm-OB mit alter Schnittstelle in eine S7-1500-CPU mit Firmware-Version >= V2.0. Der Prozessalarm-OB behält die alte Schnittstelle.
- Sie ersetzen im Engineering System eine S7-1500-CPU mit Firmware-Version < V2.0, die einen Prozessalarm-OB mit alter Schnittstelle enthält, durch eine S7-1500-CPU mit Firmware-Version >= V2.0. Der Prozessalarm-OB behält die alte Schnittstelle.

Siehe auch

Prozessalarm-OBs parametrieren (Seite 1091)

Ereignisse und OBs (Seite 980)

Beispiel für ein Prozessalarm-Ereignis (Seite 983)

OB-Startinformation (Seite 1056)

Taktsynchronalarm-OBs

Beschreibung

Mit Hilfe von Taktsynchronalarm-OBs können Sie Programmteile taktsynchron zum DP-Takt bzw. PN-Sendetakt starten.

Sie aktualisieren das zugehörige Teilprozessabbild der Eingänge und das zugehörige Teilprozessabbild der Ausgänge durch Aufruf der Anweisungen SYNC_PI und SYNC_PO im Taktsynchronalarm-OB.

⚠ VORSICHT
Direktzugriffe Vermeiden Sie bei Direktzugriffen Zugriffe auf Peripheriebereiche, denen Teilprozessabbilder mit Taktsynchronalarm-OB-Anbindung zugeordnet sind.

Hinweis

Wirksamkeit des OB-Parameters "Diagnosepuffereintrag bei Ereignisüberlauf"

Das Setzen des Optionskästchens "Diagnosepuffereintrag bei Ereignisüberlauf" bei den OB-Parametern wird erst dann wirksam, wenn Sie die OB-Eigenschaften in die CPU geladen haben und die CPU danach einen STOP-RUN-Übergang durchgeführt hat.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
  Abweichend von der Belegung der Startinformation bei S7-300- und S7-400-CPUs gilt:
  - STRT_INF=B#16#6F, wennn die OB-Nummer > 122 ist
  - OB_NUMBR=B#16#FF, wenn die OB-Nummer > 254 ist
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB beim Übergang von STOP nach RUN
PIP_Input	BOOL	TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell Hinweis: Bei S7-1500-CPUs hat diese Variable stets den Wert FALSE, da die Aktualisierung des zugehörigen Teilprozessabbilds der Eingänge durch Aufruf der Anweisung SYNC_PI erfolgen muss.
PIP_Output	BOOL	TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen Hinweis: Bei S7-1500-CPUs hat diese Variable stets den Wert FALSE, da die Aktualisierung des zugehörigen Teilprozessabbilds der Ausgänge durch Aufruf der Anweisung SYNC_PO erfolgen muss.
IO_System	USINT	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
Event_Count	INT	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).
SyncCycleTime	LTIME	Berechnete Zykluszeit

Siehe auch

Taktsynchronalarm-OBs parametrieren (Seite 1092)

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Zeitfehler-OB

Beschreibung

Das Betriebssystem der S7 1500-CPU ruft den Zeitfehler-OB auf, wenn einer der folgenden Fehler auftritt:

- Erste Überschreitung der Zykluszeit innerhalb eines Zyklus

  Hinweis

  Wenn die Zykluszeit in ein und demselben Zyklus zweimal überschritten wird, geht die CPU in STOP. Sie können dies durch Aufruf der Anweisung "RE_TRIGR" an geeigneter Stelle verhindern.

- Die Anzahl angeforderter, aber nicht fertig bearbeiteter Anforderungen eines OB hat die projektierte Warngrenze erreicht. (Bei migrierten Programmen ist diese Warngrenze auf 1 eingestellt.)
  Wenn Sie die Warngrenze auf 1 eingestellt haben, bedeutet dies also das Folgende: Wenn ein Startereignis z. B. für einen Weckalarm-OB auftritt, bevor die vorherige Bearbeitung desselben OB beendet ist, ruft das Betriebssystem den OB80 auf.
  Dies gilt für alle Ereignisse, deren zugehöriger OB einen Zeitfehler verursachen kann.
- Abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN, nachdem die CPU z. B. in HALT war
- Übersprungener Uhrzeitalarm wegen Vorstellen der Uhrzeit um mehr als 20 s (auch bei der Umstellung von Winter- auf Sommerzeit)

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation

Name	Datentyp	Bedeutung
Fault_ID	BYTE	Fehlercode. Mögliche Werte: • B#16#01: Zykluszeit überschritten • B#16#02: Der angeforderte OB ist noch in Bearbeitung. • B#16#05: abgelaufener Uhrzeitalarm durch Uhrzeitsprung • B#16#06: abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN nach HALT • B#16#07: Überlauf des OB-Anforderungspuffers für die aktuelle Prioritätsklasse • B#16#08: Taktsynchronalarm-Zeitfehler • B#16#09: Alarmverlust durch zu hohe Alarmlast • B#16#0B: Technologiesynchronalarm-Zeitfehler • B#16#0C: (nur bei S7-1500-R/H-CPUs) Zykluszeit zum zweiten Mal überschritten
Csg_OBnr	OB_ANY	Nummer des OB, der bearbeitet wurde, als der Zeitfehler auftrat
Csg_Prio	UINT	Priorität des OB, der bearbeitet wurde, als der Zeitfehler auftrat

Hinweis

Csg_OBnr=0 und Csg_Prio=0

Wenn Csg_OBnr und Csg_Prio (bzw. OB80_ERR_EV_CLASS und OB80_ERR_EV_NUM, falls Sie als Struktur der Startinformation die der S7-300- und S7-400-CPUs gewählt haben) beide den Wert Null haben , führte die CPU Hintergund-Aktivitäten aus, als der Zeitfehler auftrat.

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Diagnosealarm-OB

Beschreibung

Das Betriebssystem der S7 1500-CPU ruft den Diagnosealarm-OB in den folgenden Fällen auf:

- Eine diagnosefähige S7-1500-Baugruppe erkennt eine Änderung ihres Diagnosezustands und stellt eine Diagnosealarm-Anforderung an die CPU.
- Eine diagnosefähige S7-300- oder S7-400-Baugruppe, bei der Sie den Diagnosealarm freigegeben haben, erkennt eine Änderung ihres Diagnosezustands und stellt eine Diagnosealarm-Anforderung an die CPU.

- Es tritt ein Ereignis auf, das durch einen Fehler in der Stromversorgung oder der Batteriepufferung ausgelöst wird.

  Hinweis

  Ein solches Ereignis führt bei S7-300- und S7-400-CPUs zum Aufruf des Stromversorgungsfehler-OB.

- Das Betriebssystem hat Speicherfehler erkannt.

  Hinweis

  Ein solches Ereignis führt bei S7-300- und S7-400-CPUs zum Aufruf des CPU-Hardwarefehler-OB.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
IO_state	WORD	Zustand des Hardware-Objekts: • Bit 0: gut • Bit 1: deaktiviert • Bit 2: Wartungsbedarf • Bit 3: Wartungsanforderung • Bit 4: Fehler • Bit 5: nicht erreichbar • Bit 6: qualifiziert • Bit 7: nicht verfügbar • Bits 8 bis 14: reserviert (immer "0") • Bit 15: Netzwerk- / Hardware-Fehler Wenn Bit 4 = 1 oder Bit 5 = 1 ist: – Bit 15 = 0: Netzwerk-Fehler – Bit 15 = 1: Hardware-Fehler
LADDR	HW_ANY	Hardware-Kennung des Hardware-Objekts, das den Diagnosealarm ausgelöst hat
Channel	UINT	Kanalnummer
MultiError	BOOL	=TRUE, falls mehr als ein Fehler vorliegt

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Ziehen/Stecken-OB

Beschreibung

Das Betriebssystem der S7 1500-CPU ruft den Ziehen/Stecken-OB auf, wenn ein projektiertes und nicht deaktiviertes Modul oder Submodul der dezentralen Peripherie gezogen oder gesteckt wird.

Hinweis

Das Ziehen bzw. Stecken einer zentralen Baugruppe führt mit Ausnahme der CPUs 1510SP-1 PN und 1512SP-1 PN zum STOP der CPU.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung des betroffenen Moduls bzw. Submoduls
Event_Class	BTYE	• B#16#38: (Sub-) Modul gesteckt • B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar
Fault_ID	BYTE	Fehlercode (mögliche Werte: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58, B#16#61, B#16#63, B#16#64, B#16#65, B#16#66)

Die folgende Tabelle zeigt, welches Ereignis den Start des Ziehen/Stecken-OB verursacht hat.

ev_class B#16# ...	fault_id B#16# ...	Bedeutung
39	51	IO-Modul gezogen
39	54	IO-Submodul gezogen
38	54	IO-Submodul gesteckt und entspricht parametriertem Submodul
38	55	IO-Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul
38	56	IO-Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung
38	57	IO-Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung
38	58	IO-Submodul Zugriffsfehler beseitigt
39	61	Baugruppe gezogen bzw. nicht ansprechbar
38	61	Baugruppe gesteckt

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Baugruppenträgerfehler-OB

Beschreibung

Das Betriebssystem der S7 1500-CPU ruft den OB86 in den folgenden Fällen auf:

- Der Ausfall eines DP-Mastersystems bzw. eines PROFINET IO-Systems wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines DP-Slaves bzw. eines IO-Devices wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines Teils der Submodule eines PROFINET I-Device wird erkannt.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_Device	Hardware-Kennung des fehlerhaften Hardware-Objekts
Event_Class	BYTE	• B#16#38: gehendes Ereignis • B#16#39: kommendes Ereignis
Fault_ID	BYTE	Fehlercode (mögliche Werte: B#16#C3, B#16#C4, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)

Die folgende Tabelle zeigt, welches Ereignis den Start des OB86 verursacht hat.

ev_class B#16# ...	fault_id B#16# ...	Bedeutung
39	C3	Ausfall eines DP-Mastersystems
39/38	C4	Ausfall/Wiederkehr eines DP-Slaves
39	CA	Ausfall eines PROFINET IO-Systems
39/38	CB	Ausfall/Wiederkehr eines PROFINET IO-Devices
38	CC	Wiederkehr eines PROFINET IO-Devices mit Störung oder Wartung
38	CD	Wiederkehr eines PROFINET IO-Devices, Sollausbau weicht von Istausbau ab
38	CE	Wiederkehr eines PROFINET IO-Devices, Fehler bei der Baugruppenparametrierung
39/38	F8	Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Devices
38	F9	Wiederkehr eines Teils der Submodule eines PROFINET I-Devices mit Device-Konfigurationsunterschied

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Programmierfehler-OB

Voraussetzung für den Aufruf des Programmierfehler-OB

Eine notwendige Voraussetzung dafür, dass die S7-1500-CPU den Programmierfehler-OB aufrufen kann, ist, dass Sie die globale Fehlerbehandlung eingestellt haben.

Beschreibung

Das Betriebssystem der S7-1500-CPU ruft den Programmierfehler-OB auf, wenn während der Bearbeitung einer Anweisung des Anwenderprogramms ein Programmierfehler auftritt. Der Programmierfehler-OB wird mit der für ihn eingestellten Priorität bearbeitet.

Hinweis

Wenn Sie die lokale Fehlerbehandlung eingestellt haben, wird beim Auftreten eines Programmierfehlers die im zugehörigen Baustein programmierte Fehlerbehandlung wirksam.

Struktur der Startinformation

- wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
BlockNr	UINT	Nummer des Bausteins, in dem der Programmierfehler auftrat
Reaction	USINT	• 0: Fehler ignorieren • 1: fehlerhaften Wert ersetzen • 2: Befehl überspringen • 3: Programmierte Fehlerbehandlung, ausgelöst z. B. durch einen Feldzugriff mit unzulässigem Index oder durch einen Fehler bei der Parameterversorgung eines FC oder FB
Fault_ID	BYTE	Fehlercode (mögliche Werte: B#16#00, B#16#03, B#16#04, B#16#05, B#16#20, B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#2C, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#38, B#16#39, B#16#3A, B#16#3B, B#16#3C, B#16#3D, B#16#3E, B#16#3F, B#16#50, B#16#51, B#16#75, B#16#76, B#16#A1, B#16#A2)

Name	Datentyp	Bedeutung
BlockType	USINT	Art des Bausteins, in dem der Fehler aufgetreten ist: • OB: 1 • FC: 2 • FB: 3 • SFC: 4 • SFB: 5 • DB: 6
Area	USINT	Bereich, bei dem der fehlerhafte Zugriff auftrat: • Lokaldaten: B#16#40 bis 4E, 86, 87, 8E, 8F, C0 bis CE • Prozessabbild der Eingänge: B#16#01 • Prozessabbild der Ausgänge: B#16#02 • Technologie-DB: B#16#04 • E: B#16#81 • A: B#16#82 • M: B#16#83 • DB: B#16#84, 85, 8A, 8B
DBNr	Block_DB	DB-Nr, falls AREA = DB oder DI
Csg_OBNr	OB_ANY	OB-Nummer (121)
Csg_Prio	USINT	OB-Priorität
Width	USINT	Art des Zugriffs, bei dem der Fehler auftrat: • Bit: – B#16#00 bei Zugriff auf den Standard-Speicherbereich – B#16#01 bei Zugriff auf den optimierten Speicherbereich • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04

Die folgende Tabelle zeigt die Bedeutung der Fehlercodes:

fault_ID B#16# ...	Bedeutung
00	maximale Verschachtelungstiefe von Bausteinaufrufen überschritten
03	Sie haben zur Adressierung eines Operanden einen NULL-Pointer verwendet.
04	unbekannte Anweisung
05	Stop-Anweisung
20	Der adressierte String hat eine falsche Längenangabe.
21	BCD-Konvertierungsfehler
22	Bereichslängenfehler beim Lesen
23	Bereichslängenfehler beim Schreiben
24	Bereichsfehler beim Lesen
25	Bereichsfehler beim Schreiben
26	Fehler bei Timer-Nr.

fault_ID B#16# ...	Bedeutung
27	Fehler bei Zähler-Nr.
28	lesender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich Nulll ist
29	schreibender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich Nulll ist
2C	Sie haben zur Adressierung eines Operanden einen NULL-Pointer verwendet.
30	Schreibender Zugriff auf einen schreibgeschützten Global-DB
31	Schreibender Zugriff auf einen schreibgeschützten Instanz-DB
32	DB-Nummernfehler beim Zugriff auf einen Global-DB
33	DB-Nummernfehler beim Zugriff auf einen Instanz-DB
34	Nummernfehler bei FC-Aufruf
35	Nummernfehler bei FB-Aufruf
38	Der typsichere Datenbaustein ist nicht geladen.
39	Der Datenbaustein ist nicht typsicher definiert.
3A	Zugriff auf einen nicht geladenen DB; die DB-Nummer liegt im zulässigen Bereich
3B	DB nicht vorhanden
3C	Zugriff auf eine nicht geladene FC; die FC-Nummer liegt im zulässigen Bereich
3D	Zugriff auf eine nicht geladene SFC; die SFC-Nummer liegt im zulässigen Bereich
3E	Zugriff auf einen nicht geladenen FB; die FB-Nummer liegt im zulässigen Bereich
3F	Zugriff auf einen nicht vorhandenen SFB; die SFB-Nummer liegt im zulässigen Bereich
50	Der typsichere Datenbaustein ist nicht geladen.
51	Der Datenbaustein ist nicht typsicher definiert.
75	Die maximale Verschachtelungstiefe für Bausteinaufrufe wurde überschritten.
76	Das Maximum der verfügbaren Lokaldaten wurde überschritten.
A1	Schreibender Zugriff auf eine schreibgeschützte Variable
A2	Zugriff auf eine Variable, die einen ungültigen Zahlenwert enthält

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

Peripheriezugriffsfehler-OB

Voraussetzung für den Aufruf des Peripheriezugriffsfehler-OB

Eine notwendige Voraussetzung dafür, dass die S7-1500-CPU den Peripheriezugriffsfehler-OB aufrufen kann, ist, dass Sie die globale Fehlerbehandlung eingestellt haben.

Beschreibung

Das Betriebssystem der S7-1500-CPU ruft den Peripheriezugriffsfehler-OB auf, wenn während der Bearbeitung einer Anweisung des Anwenderprogramms ein Fehler beim direkten Zugreifen auf Peripheriedaten auftritt. Das ist z. B. bei einem Lesefehler beim direkten Zugriff auf Daten einer Eingabebaugruppe der Fall. Der Peripheriezugriffsfehler-OB wird mit der für ihn eingestellten Priorität bearbeitet.

Hinweis

Wenn Sie die lokale Fehlerbehandlung eingestellt haben, wird beim Auftreten eines Peripheriezugriffsfehlers die im zugehörigen Baustein programmierte Fehlerbehandlung wirksam.

Struktur der Startinformation

- Wie bei S7-300- und S7-400-CPUs
  Siehe AUTOHOTSPOT
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
BlockNr	UINT	Nummer des Bausteins, in dem der Peripheriezugriffsfehler auftrat
Reaction	USINT	0: Fehler ignorieren, 1: fehlerhaften Wert ersetzen, 2: Befehl überspringen
Fault_ID	BYTE	Fehlercode: • B#16#42: Peripheriezugriffsfehler, lesend • B#16#43: Peripheriezugriffsfehler, schreibend
BlockType	USINT	Art des Bausteins, in dem der Fehler aufgetreten ist : • OB: 1 • FC: 2 • FB: 3 • SFC: 4 • SFB: 5 • DB: 6
Area	USINT	Kennung für den Bereich, in dem der fehlerhafte Zugriff auftrat: • B#16#01: Direktzugriff auf Eingang • B#16#02: Direktzugriff auf Ausgang • B#16#81: Zugriff auf Prozessabbild der Eingänge • B#16#82: Zugriff auf Prozessabbild der Ausgänge
DBNr	Block_DB	nicht anwenderrelevant
Csg_OBNr	OB_ANY	Nummer des OB, der den Peripheriezugriffsfehler verursacht

Name	Datentyp	Bedeutung
Csg_Prio	USINT	Priorität des OB, der den Peripheriezugriffsfehler verursacht
Width	USINT	Art des Zugriffs, bei dem der Fehler auftrat: • Bit: B#16#00 • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04

Siehe auch

Ereignisse und OBs (Seite 980)

OB-Startinformation (Seite 1056)

ProDiag-OB

Beschreibung

Bei ProDiag ordnen Sie jeder projektierten Überwachung einen ProDiag-Funktionsbaustein zu. Die Überwachungen eines ProDiag-FB werden allerdings erst mit seinem Aufruf aktiv. Dieser Aufruf kann entweder im ProDiag-Organisationsbaustein erfolgen oder an einer beliebigen anderen Stelle Ihres Programms:

- Falls Sie in Ihrem Programm einen ProDiag-FB explizit aufrufen, wird er im ProDiag-OB nicht mehr aufgerufen.
  Wenn Sie in Ihrem Programm alle ProDiag-FBs explizit aufrufen, wird kein ProDiag-OB angelegt.
- Wenn Sie sich um den Aufruf eines oder mehrerer ProDiag-FBs nicht kümmern wollen und sie deshalb an keiner Stelle Ihres Programms aufrufen, wird das beim Übersetzen Ihres Projekts erkannt. In diesem Fall wird automatisch ein ProDiag-OB angelegt, und die besagten ProDiag-FBs werden darin aufgerufen.
  Wenn Sie keinen ProDiag-FB Ihres Projekts explizit in Ihrem Programm aufrufen, werden alle ProDiag-FBs im ProDiag-OB aufgerufen.

Hinweis

Anlegen eines ProDiag-OB

Der ProDiag-OB ist der einzige Organisationsbaustein, der automatisch angelegt wird.

Die Information, welche ProDiag-FBs im ProDiag-OB aufgerufen werden, können Sie mit Hilfe der Querverweise auslesen.

Motion Control OBs

MC-Servo-OB

Beschreibung

Wenn Sie ein Technologieobjekt für S7-1500 Motion Control anlegen, wird automatisch der Organisationsbaustein "MC-Servo [OB91]" für die Bearbeitung der Technologieobjekte angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Applikationszyklus im Ablaufsystem der SIMATIC S7-1500 aufgerufen.

Der MC-Servo-OB ist schreibgeschützt. Der Inhalt kann nicht verändert werden.

Innerhalb des MC-Servo-OB werden die Algorithmen der Lageregelung aller auf der CPU projektierten Technologieobjekte für Motion Control berechnet.

Sie können den Applikationszyklus und die Priorität des Organisationsbausteins entsprechend Ihren Anforderungen an die Regelgüte und die Systembelastung einstellen.

Applikationszyklus

Den Applikationszyklus, in dem der MC-Servo-OB aufgerufen wird, können Sie in den Eigenschaften des Organisationsbausteins einstellen:

- Synchron zum BUS
  Der MC-Servo-OB wird synchron zu einem Bussystem aufgerufen. Den Sendetakt stellen Sie in den Eigenschaften des ausgewählten Bussystems ein. Sie können folgende Bussysteme auswählen:
  - Taktsynchroner PROFIBUS DP
  - Taktsynchrones PROFINET IO

  Den MC-Servo-OB können Sie nicht synchron zu einem Bussystem aufrufen, das über einen Kommunikationsprozessor / Kommunikationsmodul (CP / CM) an die CPU angebunden ist.
- Zyklisch
  Der MC-Servo-OB wird zyklisch mit dem angegebenen Applikationszyklus aufgerufen.

Den Applikationszyklus müssen Sie groß genug wählen, um alle Technologieobjekte für Motion Control in einem Zyklus zu bearbeiten. Wenn der Applikationszyklus nicht eingehalten wird, treten Überläufe auf. Bei einem Überlauf des MC-Servo-OB geht die CPU in den Betriebszustand STOP.

Taktuntersetzung (ab CPU V1.5)

Den Applikationszyklus des MC-Servo-OB können Sie zum Sendetakt eines taktsynchronen PROFINET IO-Systems untersetzen. Als Faktor können Sie ein ganzzahliges Vielfaches des Sendetakts einstellen. Für den Applikationszyklus sind Werte bis zum 14-fachen des Sendetakts (maximal 32 ms) möglich.

Wenn Sie einen Taktsynchronalarm-OB und den MC-Servo-OB synchron zum selben PROFINET IO-System aufrufen, müssen Sie bei beiden Organisationsbausteinen den gleichen Applikationszyklus einstellen.

Siehe auch

MC-Servo-OB parametrieren (Seite 1083)

MC-PreServo-OB (Seite 1085)

MC-PostServo-OB (Seite 1086)

MC-Interpolator-OB (Seite 1084)

MC-Transformation-OB (Seite 1086)

MC-Servo-OB parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines MC-Servo-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten MC-Servo-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Zykluszeit".

Überblick über die einstellbaren Parameter

In der Gruppe "Zykluszeit" wählen Sie aus, wie der MC-Servo-OB im Ablaufsystem der SIMATIC S7-1500 aufgerufen wird:

- Zyklisch
- Synchron zum Bus

Entsprechend der gewählten Option können Sie folgende Parameter einstellen:

- Applikationszyklus
- Dezentrale Peripherie
- Faktor

Option "Zyklisch"

Der MC-Servo-OB wird zyklisch mit dem angegebenen Applikationszyklus aufgerufen.

Im Feld "Applikationszyklus" stellen Sie den Zyklus ein, in dem der MC-Servo-OB aufgerufen wird.

Option "Synchron zum Bus"

Der MC-Servo-OB wird synchron zu einem Bussystem aufgerufen. Den Sendetakt stellen Sie in den Eigenschaften des ausgewählten Bussystems ein.

Folgende Parameter sind für die Option "Synchron zum Bus" verfügbar:

- **Dezentrale Peripherie**
  In der Klappliste wählen Sie das Bussystem, zu dem der MC-Servo-OB taktsynchron aufgerufen wird. Sie können folgende Bussysteme auswählen:
  - Taktsynchrones PROFINET IO
  - Taktsynchroner PROFIBUS DP

  Den MC-Servo-OB können Sie nicht synchron zu einem Bussystem aufrufen, das über einen Kommunikationsprozessor / Kommunikationsmodul (CP / CM) an die CPU angebunden ist.
- **Sendetakt (ms)**
  In diesem Feld wird der Sendetakt bzw. der DP-Zyklus angezeigt, der im ausgewählten PROFINET IO-System bzw. DP-Mastersystem eingestellt ist.
- **Faktor**
  Den Applikationszyklus des MC-Servo-OB können Sie zum Sendetakt eines taktsynchronen PROFINET IO-Systems untersetzen. Im Feld "Faktor" stellen Sie ein ganzzahliges Vielfaches des Sendetakts ein. Für den Applikationszyklus sind Werte bis zum 14-fachen des Sendetakts (maximal 32 ms) möglich.
- **Applikationszyklus (ms)**
  In diesem Feld wird Applikationszyklus angezeigt, der sich aus dem Sendetakt und dem eingestellten Faktor ergibt.

Siehe auch

MC-Servo-OB (Seite 1082)

MC-Interpolator-OB

Beschreibung

Wenn Sie ein Technologieobjekt für S7-1500 Motion Control anlegen, wird automatisch der Organisationsbaustein "MC-Interpolator [OB91]" für die Bearbeitung der Technologieobjekte angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Applikationszyklus des MC-Servo-OB (Seite 1082) im Ablaufsystem der SIMATIC S7-1500 aufgerufen.

Der MC-Interpolator-OB ist schreibgeschützt. Der Inhalt kann nicht verändert werden.

Innerhalb des MC-Interpolator-OB erfolgt die Auswertung der Motion Control-Anweisungen sowie die Überwachung und die Sollwertgenerierung für alle auf der CPU projektierten Technologieobjekte (Motion Control).

Der MC-Interpolator-OB wird am Ende der Bearbeitung des MC-Servo-OB aufgerufen. Das Taktverhältnis zwischen MC-Interpolator-OB und MC-Servo-OB ist immer 1:1.

Ablaufverhalten

Den Applikationszyklus des MC-Servo-OB müssen Sie groß genug wählen, um alle Technologieobjekte für Motion Control in einem Applikationszyklus zu bearbeiten. Wenn der Applikationszyklus nicht eingehalten wird, treten Überläufe auf. Beachten Sie bei Überläufen folgendes Verhalten:

- Die CPU toleriert maximal drei aufeinander folgende Überläufe des MC-Interpolator-OB.
- Die Bearbeitung eines MC-Interpolator-OB darf maximal durch einen Aufruf des MC-Servo-OB unterbrochen werden.

Wenn mehr Überläufe bzw. Unterbrechungen auftreten, geht die CPU in den Betriebszustand STOP.

Siehe auch

MC-Servo-OB parametrieren (Seite 1083)

MC-Transformation-OB (Seite 1086)

MC-PreServo-OB

Beschreibung

Der Organisationsbaustein MC-PreServo [OB67] ist programmierbar und wird in dem Applikationszyklus aufgerufen, der am MC-Servo-OB (Seite 1082) konfiguriert ist. MC-PreServo [OB67] wird direkt vor dem MC-Servo [OB91] aufgerufen.

Über den Organisationsbaustein lässt sich der konfigurierte Applikationszyklus (Angabe in µs) auslesen.

Struktur der Startinformation

Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB beim Übergang von STOP nach RUN
PIP_Input	BOOL	TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell
PIP_Output	BOOL	TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen
IO_System	USINT	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
Event_Count	INT	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).
Synchronous	BOOL	TRUE: Der MC-Servo-OB wird synchron zu einem Bussystem aufgerufen.
CycleTime	UDINT	Anzeige des am MC-Servo-OB konfigurierten Applikationszyklus in ns

MC-PostServo-OB

Beschreibung

Der Organisationsbaustein MC-PostServo [OB95] ist programmierbar und wird in dem Applikationszyklus aufgerufen, der am MC-Servo-OB (Seite 1082) konfiguriert ist. MC-PostServo [OB95] wird direkt nach dem MC-Servo [OB91] aufgerufen.

Über den Organisationsbaustein lässt sich der konfigurierte Applikationszyklus (Angabe in µs) auslesen.

Struktur der Startinformation

Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB beim Übergang von STOP nach RUN
PIP_Input	BOOL	TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell
PIP_Output	BOOL	TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen
IO_System	USINT	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
Event_Count	INT	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).
Synchronous	BOOL	TRUE: Der MC-Servo-OB wird synchron zu einem Bussystem aufgerufen.
CycleTime	UDINT	Anzeige des am MC-Servo-OB konfigurierten Applikationszyklus in ns

MC-Transformation-OB

Beschreibung

Im Organisationsbaustein MC-Transformation [OB98] programmieren Sie die Transformation der kartesischen Koordinaten und der achsspezifischen Sollwerte für anwenderdefinierte Kinematiken. Dabei umfasst die Programmierung die Transformation der Positionen und der Dynamikwerte (Geschwindigkeit, Beschleunigung, Ruck).

Wenn Sie den MC-Transformation [OB98] im TIA Portal hinzufügen, wird automatisch der Datenbaustein "TransformationParameter" unter "Programmbausteine > Systembausteine > Programmressourcen" angelegt. In den Eigenschaften des Organisationsbausteins zeigt der MC-Transformation [OB98] die Nummer des Datenbausteins "TransformationParameter" unter "Allgemein > Transformation" an.

Im Anwenderprogramm versorgen Sie die Eingangsparameter des MC-Transformation [OB98] mit den nötigen Startinformationen. Im Datenbaustein "TransformationParameter" schreiben Sie die achsspezifischen Parameter bzw. die kartesischen Parameter. Der

MC-Transformation [OB98] wird mit diesen Startinformationen im Ablaufsystem der CPU aufgerufen.

Hinweis

Nur ein MC-Transformation [OB98] je CPU

Alle benötigten Anwendertransformationen programmieren Sie in einem Organisationsbaustein MC-Transformation [OB98]. Wenn Sie mehrere Transformationen oder mehrere Instanzen einer Tranformation benötigen, müssen Sie die Transformationen bzw. die Instanzen in Ihrem Anwenderprogramm identifizieren.

Bausteinaufruf und Priorität

Der MC-Transformation [OB98] wird entsprechend der konfigurierten Priorität im Ablaufsystem der CPU aufgerufen.

Die Priorität des MC-Transformation [OB98] konfigurieren Sie in den Eigenschaften des Organisationsbausteins unter "Allgemein > Attribute > Priorität". Für die Priorität können Sie Werte von 17 bis 25 einstellen (Voreinstellung 25):

- Die Priorität des MC-Transformation [OB98] muss mindestens um eins niedriger sein als die Priorität des MC-Servo [OB91].
- Die Priorität des MC-Transformation [OB98] muss mindestens um eins höher sein als die Priorität des MC-Interpolator [OB92].

Parameter

Die folgende Tabelle zeigt die Parameter des MC-Transformation [OB98]:

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
KinematicsObject	INPUT	DB_ANY	-	Technologieobjekt Kinematik, für das der MC-Transformation [OB98] bei Aufruf die Transformation berechnet.	
ExecutionContext	INPUT	DINT	-	Bearbeitungskontext des OB MC-Transformation	
				0	MOTION_EXECUTION Berechnung der Achssollwerte in der Bewegungsausführung im Interpolator. Die berechneten Werte sind für die aktuelle Bewegungsführung notwendig. Drehmomentenvorsteuerungswerte können berechnet und vorgegeben werden.
				1	NON_MOTION_EXECUTION Berechnung der Achssollwerte aus IPO-Task und anderen OBs möglich. Die Transformation ist für die Bewegungsplanung notwendig.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
TransformationType	INPUT	DINT	-	Angeforderte Berechnung	
				0	Vorwärtstransformation Berechnung der kartesischen Parameter aus den Achsstellungen
				1	Rückwärtstransformation Berechnung der achsspezifischen Parameter aus den kartesischen Parametern
TransformationParameters	InOut	VARIANT	-	Zeiger auf den Datenbaustein "TransformationParameter"	
FunctionResult	OUTPUT	DINT	-	Rückgabewert des MC-Transformation [OB98] an das Technologieobjekt Kinematik	
				0	Berechnung erfolgt und Parameter ausgegeben
				< 0	Fehler bei der Berechnung Wenn bei der Berechnung Fehler auftreten, stoppt das Technologieobjekt Kinematik die Bewegung. Das Technologieobjekt Kinematik gibt einen Technologie-Alarm mit der Fehlerkennung als Begleitwert aus und löscht die Auftragskette.
				Wenn bei der Berechnung Fehler auftreten, stoppt das Technologieobjekt Kinematik die Bewegung. Das Technologieobjekt Kinematik gibt einen Technologie-Alarm mit der Fehlerkennung als Begleitwert aus und löscht die Auftragskette.	

Siehe auch

MC-Servo-OB (Seite 1082)

MC-Interpolator-OB (Seite 1084)

Parameter von Organisationsbausteinen einstellen

Uhrzeitalarm-OBs parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines Uhrzeitalarm-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Uhrzeitalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Uhrzeitalarm".

Überblick über die einstellbaren Parameter

Sie können folgende Parameter einstellen:

- Ausführung
- Startdatum und Uhrzeit
- Optionsfelder "Lokalzeit" und "Systemzeit"

Parameter "Ausführung"

Über die Klappliste "Ausführung" geben Sie vor, mit welcher Periode der Uhrzeitalarm -OB ausgeführt werden soll. Die Zeitintervalle beziehen sich auf die Einstellungen bei "Startdatum" und "Uhrzeit".

Folgende Werte von "Ausführung" sind möglich:

- nie
- einmalig
- minütlich
- stündlich
- täglich
- wöchentlich
- monatlich
- jährlich
- am Monatsende

Hinweis

Beim Wert "am Monatsende" ist der unter "Startdatum" angegebene Wert irrelevant.

Parameter "Startdatum" und "Uhrzeit"

Hier stellen Sie ein, zu welchem Zeitpunkt der Uhrzeitalarm-OB zum ersten Mal ausgeführt werden soll.

Beispiel: Startdatum = 05.01.2013, Uhrzeit =11:16

Abhängig vom Wert des Parameters "Ausführung" erzeugt die CPU periodisch weitere Uhrzeitalarme. Die Startzeit bezieht sich je nach Einstellung entweder auf die Lokalzeit oder auf die koordinierte Weltzeit UTC.

Hinweis

Wenn Sie für den Parameter "Ausführung" "monatlich" einstellen, sind der 29., der 30. und der 31. als Startdatum nicht möglich. Wenn der Uhrzeitalarm-OB am Monatsende gestartet werden soll, wählen Sie stattdessen "am Monatsende" beim Parameter "Ausführung".

"Lokalzeit" oder "Systemzeit"

Hier stellen Sie ein, auf welche Zeit sich die Startzeit des Uhrzeitalarm-OB bezieht:

- "Lokalzeit": Die Startzeit bezieht sich auf die eingestellte Zeitzone der CPU.
- "Systemzeit": Die Startzeit bezieht sich auf die koordinierte Weltzeit UTC (Universal Time Coordinated).

Siehe auch

Uhrzeitalarm-OBs (Seite 1059)

Weckalarm-OBs parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines Weckalarmalarm-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Weckalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Weckalarm".

Überblick über die in der Gruppe "Weckalarm" einstellbaren Parameter

In der Gruppe "Weckalarm" können Sie folgende Parameter einstellen:

- Zykluszeit
- Phasenverschiebung

Parameter "Zykluszeit"

Über den Parameter "Zykluszeit" stellen Sie die Zeitspanne zwischen zwei Aufrufen des Weckalarm-OB ein. Sie ist ein ganzzahliges Vielfaches von 1 µs.

Parameter "Phasenverschiebung"

Hier stellen Sie diejenige Zeitspanne ein, um die die Startzeitpunkte gegenüber den Vielfachen der Zykluszeit verschoben sind.

Eine Phasenverschiebung kann bei der Benutzung mehrerer Weckalarm-OBs sinnvoll sein. Wenn deren Zykluszeiten nämlich gemeinsame Vielfache haben, können Sie durch Verwendung einer Phasenverschiebung gleichzeitige Startzeitpunkte verhindern.

Priorität des Weckalarm-OB

Beim Hinzufügen eines neuen Weckalarm-OB zu Ihrem Projekt geben Sie dessen Zykluszeit vor. Dabei wird gemäß folgender Tabelle die Priorität des Weckalarm-OB voreingestellt.

Zykluszeit	Voreingestellte Priorität
0,25 ms <= Zykluszeit <= 2 ms	17
2 ms < Zykluszeit <= 5 ms	16
5 ms < Zykluszeit <= 10 ms	14
10 ms < Zykluszeit <= 100 ms	13
100 ms < Zykluszeit <= 200 ms	12
200 ms < Zykluszeit <= 500 ms	11
500 ms < Zykluszeit <= 1000 ms	10
1000 ms < Zykluszeit <= 2000 ms	9
Zykluszeit > 2000 ms	8

Die voreingestellte Priorität können Sie folgendermaßen ändern:

1. Öffnen Sie den zum betrachteten Weckalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Attribute".
3. Geben Sie im Bereich "Priorität" in das Eingabefeld "Priorität" den gewünschten Wert ein.

Siehe auch

Weckalarm-OBs (Seite 1066)

Prozessalarm-OBs parametrieren

Vorgehen zur Konfigurierung der Startereignisse eines Prozessalarm-OB

Jedes Startereignis eines Prozessalarm-OB konfigurieren Sie in den Eigenschaften des zugehörigen Moduls. Dazu gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum zugehörigen Modul gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf denjenigen Kanal, der einen Prozessalarm auslösen können soll.
3. Aktivieren Sie diejenigen Optionskästchen, deren zugehörigen Ereignisse einen Prozessalarm auslösen sollen. Geben Sie ggf. zugehörige Grenzwerte vor.
4. Geben Sie einen Ereignisnamen ein, z. B. "Steigende Flanke 0".
5. Wählen Sie aus der Klappliste "Prozessalarm" einen bereits bestehenden Prozessalarm-OB aus oder legen sie einen neuen Prozessalarm-OB an.
6. Geben Sie ggf. die Priorität des Prozessalarm-OB vor.

Für das Ereignis, das durch seinen eindeutigen Ereignisnamen gekennzeichnet ist, wird automatisch eine Systemkonstante vom Datentyp "Event_HwInt" angelegt. Diese wird in der Standard-Variablentabelle im Register "Systemkonstanten" angezeigt.

Überblick über die Startereignisse eines Prozessalarm-OB

Um die aktuell konfigurierten Startereignisse eines Prozessalarm-OB anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Prozessalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Startereignisse".

Siehe auch

Prozessalarm-OBs (Seite 1068)

Beispiel für ein Prozessalarm-Ereignis (Seite 983)

Taktsynchronalarm-OBs parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines Taktsynchronalarm-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Taktsynchronalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Taktsynchronität".

Überblick über die einstellbaren Parameter

Folgende Parameter können Sie einstellen:

- Applikationszyklus
- Automatische Einstellung
- Verzögerungszeit

Parameter "Applikationszyklus"

- Bei PROFINET: ganzzahliges Vielfaches (mögliche Werte: 1, ... 14) des Sendetakts
- Bei PROFIBUS DP: DP-Zyklus

Optionskästchen "Automatische Einstellung"

- Bei deaktiviertem Optionskästchen können Sie die Verzögerungszeit manuell vorgeben.
- Bei aktiviertem Optionskästchen wird die Verzögerungszeit automatisch berechnet. Das zugehörige Eingabefeld ist dann gegraut.

Parameter "Verzögerungszeit"

Hinweis

Den Parameter "Verzögerungszeit" können Sie nur bei deaktiviertem Optionskästchen "Automatische Einstellung" vorgeben.

Die Verzögerungszeit ist die Zeitspanne zwischen dem Sendetakt bzw. dem Global-Control-Telegramm und dem Start des Taktsynchronalarm-OB. In dieser Zeit wickelt der PROFINET IO-Controller bzw. der DP-Master den zyklischen Datenaustausch mit den IO-Devices bzw. den DP-Slaves ab.

Wenn die Verzögerungszeit Null ist, startet der Taktsynchronalarm-OB mit dem Sendetakt bzw. dem Global-Control-Telegramm. In diesem Fall ist es sehr wahrscheinlich, dass ein Zugriff auf die taktsynchrone Peripherie über die erweiterte Anweisung "SYNC_PI" zu Beginn des OB fehlschlägt, da die Peripheriedaten noch nicht zur Verfügung stehen.

Wenn die Verzögerungszeit größer als der Vorgabewert ist, wird OB-Laufzeit "verschenkt", da der Zugriff auf die taktsynchrone Peripherie bereits früher möglich wäre.

Überblick über die angezeigten Parameter

Folgende Parameter werden angezeigt:

- Dezentrale Peripherie
- Teilprozessabbild

"Dezentrale Peripherie"

Hier wird dasjenige PROFINET IO-System bzw. DP-Mastersystem angezeigt, dessen IO-Devices bzw. DP-Slaves dem Taktsynchronalarm-OB zugeordnet sind.

Die Definition eines PROFINET IO-Systems bzw. DP-Mastersystems erfolgt bei der Zuordnung zwischen Teilprozessabbild und Taktsynchronalarm-OB.

"Teilprozessabbild"

Hier wird dasjenige Teilprozessabbild angezeigt, das dem Taktsynchronalarm-OB zugeordnet ist. Diese Zuordnung haben Sie bei der Hardware-Konfigurierung vorgenommen.

Siehe auch

Taktsynchronalarm-OBs (Seite 1070)

Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen

Im Folgenden erhalten Sie Informationen zur Kompatibilität von CPU-Firmware-Versionen und zu deren Abhängigkeiten von verschiedenen STEP 7-Versionen.

Es ist möglich, eine CPU mit neuer Firmware (z. B. Firmware V1.7, die ab STEP 7 V13 SP1 konfigurierbar ist) mit einer nicht aktuellen STEP 7-Version (z. B. STEP 7 V12, die nur die

Firmware V1.0 kennt) zu programmieren. In diesem Fall ist die Funktionalität der CPU mit Firmware-Stand V1.7 eingeschränkt auf die Funktionalität des Firmware-Stands V1.0.

Die S7-1500-CPUs sind rückwärts kompatibel zu ihren direkten Vorgängern.

Die S7-1500 CPUs sind kompatibel zu vergleichbaren S7-1500T CPUs. Beispiel: Eine CPU 1511 ist kompatibel zu einer CPU 1511T.

Umgekehrt ist eine Kompatibilität nur gegeben, wenn Sie keine erweiterten Motion Control Funktionen nutzen.

Beispiele für Rückwärtskompatibilität

- CPU 1516 V2.0 ist kompatibel zu allen Firmware-Versionen der CPU 1516.
  Die CPU 1516 V2.0 wird mit einer neuen Bestellnummer ausgeliefert, ist aber uneingeschränkt mit den Vorgänger-CPUs mit der bisherigen Bestellnummer kompatibel. Sie brauchen nichts zu beachten.
- CPU 1516 V1.7 ist kompatibel zu den CPUs 1516 V1.6, V1.5, V1.1 und V1.0, aber nicht kompatibel zu allen anderen S7-1500-CPUs, z. B. zu CPU 1518-4PN/DP.
- CPU 1513 V1.5 ist kompatibel zu den CPUs 1513 V1.1 und V1.0, aber nicht kompatibel zu allen anderen S7-1500-CPUs.
- CPU 1511 V1.0 ist nicht kompatibel zu allen anderen S7-1500-CPUs.
- CPU 1518F V1.7 ist kompatibel zu CPU 1518F V1.6.

Hinweis

Kompatibilität von F-CPUs

F-CPU-Versionen sind nicht kompatibel mit Nicht-F-CPUs. Z. B. ist CPU 1516F V1.6 nicht kompatibel zu CPU 1516 V1.5 und andersherum.

1.4.1.3 Arbeitsweise von S7-1500R/H CPUs

Funktionsweise des redundanten Systems S7-1500R/H

Besonderheiten des redundanten Systems S7-1500R/H

Redundante Automatisierungssysteme werden eingesetzt, um eine höhere Verfügbarkeit zu erreichen. Bei hochverfügbaren Systemen wird durch den Parallelbetrieb zweier Systeme die Wahrscheinlichkeit von Produktionsausfällen reduziert. Das redundante System S7-1500R/H weist gegenüber dem Automatisierungssystem S7-1500 beispielhaft folgende Besonderheiten auf:

- Beim redundanten System S7-1500R/H sind die CPUs doppelt, d. h. redundant vorhanden. Wenn eine der beiden CPUs ausfällt, übernimmt die andere CPU an der Unterbrechungsstelle die Prozessführung. Der Ausfall einer CPU hat keinen Einfluss auf den gesteuerten Prozess. Um eine Übernahme der Prozessführung zu gewährleisten, synchronisiert sich die prozessführende CPU (Primary-CPU) über eine redundant vorhandene Synchronisationsverbindung (Redundanzverbindung) mit der nicht-prozessführenden CPU (Backup-CPU).
  Informationen zur Projektierung des S7-1500R/H-Systems finden Sie im Kapitel "Projektierung des redundanten Systems S7-1500R/H (Seite 1097)".
- Die Peripherie ist in einem S7-1500R/H-System über einen PROFINET-Ring an die CPUs angeschlossen und als Medienredundanzring konfiguriert.
  Informationen zur Konfiguration eines Medienredundanzrings finden Sie im Kapitel "Medienredundanz konfigurieren (Seite 3098)".
- In S7-1500R/H-Systemen gibt es neue Betriebs- und Systemzustände.
  Weitere Informationen finden Sie im Kapitel "Betriebs- und Systemzustände (Seite 1104)".
- Für S7-1500R/H CPUs gibt es besondere Anweisungen und Organisationsbausteine (OBs).
  Weitere Informationen finden Sie im Kapitel "Grundlagen der Programmbearbeitung (Seite 1111)".
- Projektdaten können in die Primary-CPU oder in die Backup-CPU geladen werden.
  Weitere Informationen finden Sie im Kapitel "Laden einer S7-1500R/H CPU (Seite 1115)".
- Neben den Geräte IP-Adressen unterstützen redundante Systeme eine System IP-Adresse.
  Weitere Informationen finden Sie im Kapitel "Kommunikation mit dem R/H-System (Seite 1119)".
- Für das redundante System S7-1500R/H gibt es neue Diagnosemöglichkeiten.
  Weitere Informationen finden Sie im Kapitel "Diagnose des redundanten Systems S7-1500R/H (Seite 1130)".

Die vollständige Dokumentation des redundanten Systems S7-1500R/H finden Sie im Systemhandbuch Redundantes System S7-1500R/H und in den verlinkten Dokumenten.

Einschränkungen und Besonderheiten (Hardware)

Beim Einsatz des redundanten Systems S7-1500R/H gibt es Einschränkungen bei der Hardware gegenüber dem Automatisierungssystem S7-1500. Nachfolgend finden Sie beispielhaft einige Einschränkungen aufgelistet:

- Keine Unterstützung von zentraler Peripherie und zentralen Kommunikationsmodulen im Hardwareaufbau des redundanten Systems S7-1500R/H
  Ausnahmen: Laststromversorgung (PM), Systemstromversorgung (PS)
- Keine Unterstützung von fehlersicheren Modulen im redundanten System S7-1500R/H
- Längere Zyklus- und Reaktionszeiten

Einen vollständigen Überblick über die Einschränkungen des redundanten Systems S7-1500R/H gegenüber dem Automatisierungssystem S7-1500 finden Sie im Kapitel "Einsatzplanung > Einschränkungen" im Systemhandbuch Redundantes System S7-1500R/H.

Einschränkungen und Besonderheiten (Software)

Beim Einsatz des redundanten Systems S7-1500R/H gibt es Einschränkungen bei der Software gegenüber dem Automatisierungssystem S7-1500. Nachfolgend finden Sie beispielhaft einige Einschränkungen aufgelistet:

- Keine Unterstützung bzw. Einschränkungen von bestimmten Anweisungen
- Temporäre Lokaldaten bei Funktionen (FCs): Die systemseitige Initialisierung bei Standardzugriff ist bei S7-1500R/H-CPUs identisch wie beim optimierten Zugriff, zur Beschreibung der systemseitigen Initialisierung bei optimiertem Zugriff siehe AUTOHOTSPOT.
- Keine Unterstützung von IRT, MRPD
- Keine Unterstützung von Taktsynchronität

Einen vollständigen Überblick über die Einschränkungen des redundanten Systems S7-1500R/H gegenüber dem Automatisierungssystem S7-1500 finden Sie im Kapitel "Einsatzplanung > Einschränkungen" im Systemhandbuch Redundantes System S7-1500R/H.

Einschränkungen für die Kommunikation

- PG-Kommunikation:
  - Es ist nicht möglich, online auf beide CPUs gleichzeitig zuzugreifen. Sie können entweder auf die Primary-CPU oder auf die Backup-CPU zugreifen.
  - Kein Laden von Objekten (z. B. Bausteine) im redundanten Betrieb (Zustand RUN-Redundant) möglich; weder ein "Laden in Gerät" noch ein "Laden von Gerät" ist möglich. Außerdem ist das Arbeiten mit Momentaufnahmen für Variablen eines Datenbausteins in diesem Systemzustand nicht möglich.
  - Keine Unterstützung der Funktion "Laden des Geräts als neue Station"
- Open User Communication:
  - Keine projektierten Verbindungen
  - Secure Open User Communication:
    Keine Unterstützung, da bei den S7-1500R/H-CPUs kein Zertifikatemanagement möglich ist: Wenn Sie Secure OUC aktiviert haben, dann können Sie das Anwenderprogramm zwar übersetzen und laden aber den S7-1500R/H-CPUs keine Zertifikate hinzufügen.
    Wegen fehlendem Zertifikatsmanagement sind E-Mail-Verbindungen auch nur non-secure (d. h. die Anweisung TMAIL_C kann nur bis einschließlich V4.1 genutzt werden)
  - Keine FDL-Verbindungen
  - Verbindungsaufbau (Anweisung TCON) nicht mit Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_Param möglich
- Kein OPC UA
- Keine S7-Kommunikation als Client
- Kein Webserver
- Kein S7-Routing zwischen der PROFINET-Schnittstelle X1 und der PROFINET-Schnittstelle X2 der CPUs

Siehe auch

System IP-Adressen (Seite 1119)

Projektierung des redundanten Systems S7-1500R/H

R/H-CPUs anlegen

Das redundante System S7-1500R/H können Sie in unterschiedlichen Varianten aufbauen. Bei allen Aufbauvarianten ist ein PROFINET-Ring zwingend erforderlich.

Bei den Aufbauvarianten des S7-1500R/H-Systems besteht Redundanz bei folgenden Komponenten:

- R/H-CPUs
- Schnittstellen der Synchronisation
- Netzwerk im PROFINET-Ring

Dieses Kapitel beschreibt eine mögliche Aufbauvariante. Eine Beschreibung weiterer Aufbauvarianten finden Sie im Kapitel "Aufbauvarianten" im Systemhandbuch Redundantes System S7-1500R/H.

R/H-CPUs anlegen

In der Netzsicht wählen Sie aus dem Hardware-Katalog eine R/H-CPU aus und legen diese zusammen mit einem Baugruppenträger an. Auf dieses Gerät ziehen Sie aus dem Hardware-Katalog die gewünschten Baugruppen, die automatisch auf dem Baugruppenträger angeordnet werden.

Um eine R/H-CPU aus dem Hardware-Katalog auszuwählen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog auf den Ordner mit den gewünschten R/H-CPUs.
2. Öffnen Sie den Ordner mit dem gesuchten R/H-CPU-Typ. Sie sehen alle Artikelnummern des angewählten R/H-CPU-Typs.
3. Klicken Sie auf eine R/H-CPU-Artikelnummer, um sich in der Palette "Information" über die gewählte R/H-CPU zu informieren.
4. Stellen Sie in der Klappliste "Version" die passende Firmware-Version Ihrer R/H-CPU ein.
5. Legen Sie die R/H-CPU und einen Baugruppenträger an. Sie haben dazu folgende Möglichkeiten:
   - Ziehen Sie per Drag & Drop die R/H-CPU aus dem Hardware-Katalog in die Netzsicht.
   - Doppelklicken Sie auf den R/H-CPU-Eintrag im Hardware-Katalog.

Darstellung in der Netzsicht

STEP 7 legt automatisch beide R/H-CPUs des redundanten Systems an. STEP 7 stellt beide R/H-CPUs grafisch in der Netzsicht dar:

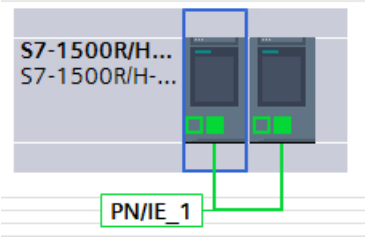

Darstellung in der Projektnavigation

STEP 7 stellt in der Projektnavigation beide R/H-CPUs des redundanten Systems durch einen eigenen Baum dar. Die Baumstruktur beinhaltet alle Bestandteile und Editoren des Projekts.

S7-1500R/H-System_1 [S7-1500R/H-Station] Gerätekonfiguration Online & Diagnose	Für jedes redundante System im Projekt gibt es einen eigenen Ordner mit dessen projektinternem Namen. Unterhalb dieses Ordners finden Sie die Gerätekonfiguration und Diagnosemöglichkeiten, die das Gesamtsystem betreffen.
PLC_1 [CPU 1515R-2 PN] Gerätekonfiguration Online & Diagnose Programmbausteine Technologieobjekte Externe Quellen PLC-Variablen PLC-Datentypen Beobachtungs- und Forcetabellen Online-Sicherungen Traces Geräte-Proxy-Daten Programminformationen PLC-Überwachungen & -Meldungen PLC-Meldetextlisten Lokale Module PLC_1 [CPU 1515R-2 PN] Dezentrale Peripherie PROFINET IO-System (100): PN/IE_1 IO-Device_1 [IM 155-6 PN HF]	Für beide CPUs des redundanten Systems im Projekt gibt es je einen eigenen Ordner mit dessen projektinternem Namen. Die im Projektbaum oben dargestellte CPU besitzt die Redundanz-ID "1". Unter der CPU sind ihre Eigenschaften dargestellt. Außerdem finden Sie hier weitere Eigenschaften des redundanten Systems, das Anwenderprogramm und weitere systemrelevante Projektteile. Im Ordner "Dezentrale Peripherie" finden Sie Verknüpfungen zu den der CPU zugeordneten dezentralen Peripheriegeräten.
PLC_2 [CPU 1515R-2 PN] Gerätekonfiguration Online & Diagnose Lokale Module PLC_2 [CPU 1515R-2 PN]	Die im Projektbaum unten dargestellte CPU besitzt die Redundanz-ID "2". Unter der CPU sind ihre Eigenschaften dargestellt.
Nicht gruppierte Geräte IO-Device_1 [IM 155-6 PN HF] Gerätekonfiguration Online & Diagnose IO-Device_1 [IM 155-6 PN HF] DI 8x24VDC HF_1 DQ 8x24VDC/0.5A HF_1	Im Ordner "Nicht gruppierte Geräte" werden alle dezentralen Peripheriegeräte des Projekts zusammengefasst.

Weitere Informationen finden Sie im Kapitel "Funktion und Aufbau der Projektnavigation" des Informationssystems.

Redundanz-IDs und IP-Adressen

Jede CPU des redundanten Systems hat eine Redundanz-ID. Mit der Redundanz-ID wird der realen CPU im Aufbau ein Projektbaum in STEP 7 zugewiesen. Die obere der beiden CPUs im Baum (im Beispiel hier: PLC_1) ist immer die CPU mit der Redundanz-ID "1". Die untere CPU (im Beispiel hier: PLC_2) besitzt die Redundanz-ID "2".

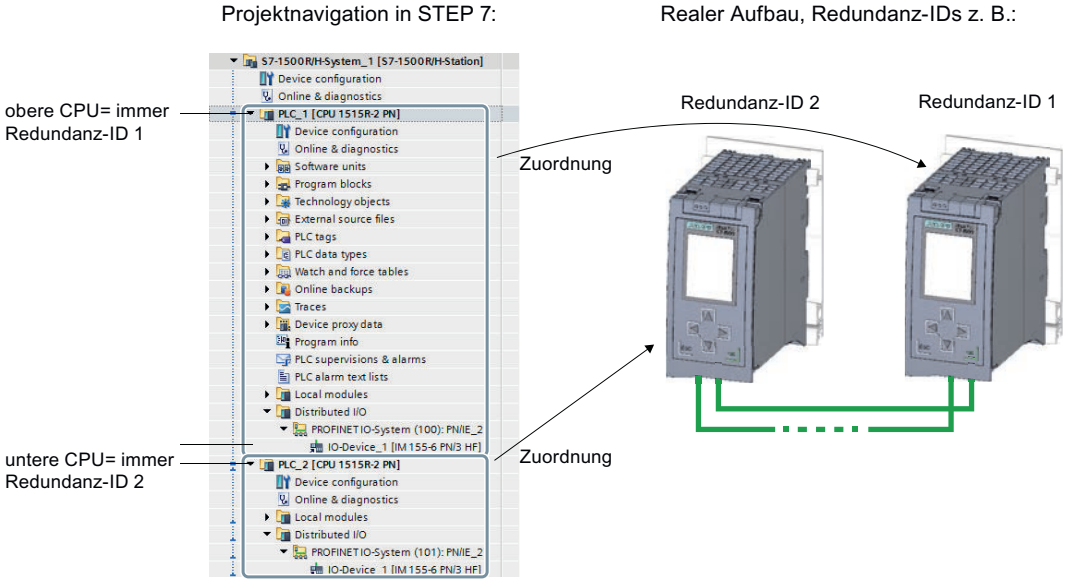

Wenn eine CPU über eine gültige Hardware-Konfiguration verfügt und Sie die Redundanz-ID dieser CPU ändern, ändern sich auch der Name und die IP-Adressen dieser CPU. Weitere Informationen finden Sie im Kapitel "Redundanz-IDs" des Systemhandbuchs "Redundantes System S7-1500R/H".

R/H-CPUs projektieren

IP-Adresse vergeben

STEP 7 weist jeder PROFINET-Schnittstelle einer CPU automatisch eine IP-Adresse zu. Sie können die IP-Adressen auch manuell vergeben, z. B. im Konfliktfall.

Für die PROFINET-Schnittstelle X1 der CPUs müssen sich die IP-Adressen im gleichen Subnetz befinden.

Die IP-Adresse wird angezeigt in den CPU-Eigenschaften, im Bereich "PROFINET-Schnittstelle [X1]" des Abschnitts "IP-Protokoll".

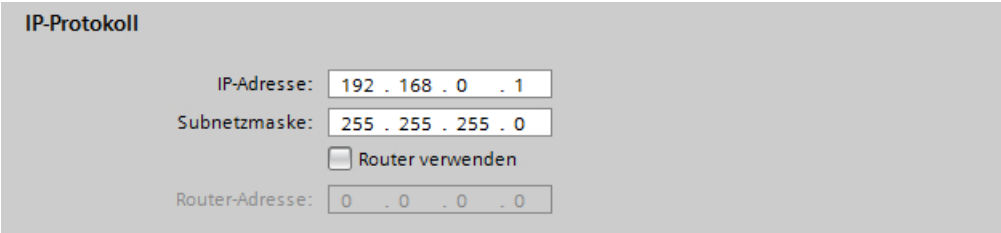

System IP-Adresse vergeben

Zusätzlich zu den Geräte IP-Adressen der CPUs können Sie für das redundante System S7-1500R/H System IP-Adressen vergeben; für jede PROFINET-Schnittstelle können Sie neben der "normalen" (Geräte-) IP-Adresse eine System IP-Adresse für geschaltete Kommunikation vergeben.

Die System IP-Adresse verwenden Sie für die Kommunikation mit anderen Geräten (z. B. HMI-Geräte, CPUs, PG/PC). Die Geräte kommunizieren über die System IP-Adresse immer mit der Primary-CPU des redundanten Systems. Dadurch wird z. B. sichergestellt, dass der Kommunikationspartner nach einem Ausfall der Primary-CPU im redundanten Betrieb mit der neuen Primary-CPU (vorher Backup-CPU) im Systemzustand RUN-Solo kommunizieren kann.

Um z. B. die System IP-Adresse für die PROFINET-Schnittstellen X1 der beiden CPUs zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Netzsicht eine CPU aus. Wählen Sie im Inspektorfenster das Register "Eigenschaften".
2. Wählen Sie den Bereich "PROFINET-Schnittstelle [X1]" und in der Bereichsnavigation den Abschnitt "System-IP-Adresse für geschaltete Kommunikation".
3. Vergewissern Sie sich, dass das Optionskästchen "Aktivieren Sie die System-IP-Adresse für geschaltete Kommunikation" für die Schnittstelle X1 gesetzt ist. Übernehmen oder vergeben Sie die System IP-Adresse im Feld "IP-Adresse".
   Die Subnetzmaske ist unveränderlich und entspricht der Subnetzmaske des IP-Protokolls.
4. Übernehmen oder weisen Sie anschließend der System IP-Adresse eine virtuelle MAC-Adresse zu.
   Die virtuelle MAC-Adresse ist sechs Bytes lang. Die Belegung der Bytes ist hexadezimal.

   Hinweis

   Virtuelle MAC-Adresse

   Stellen Sie sicher, dass sämtliche in der Ethernet-Broadcast-Domäne hinterlegten MAC-Adressen eindeutig sind. Dies gilt vor allem für Anlagen mit Fremdgeräten bestehend aus VRRP und redundanten Systemen, die durch mehrere STEP 7-Projekte konfiguriert werden.

5. Die andere CPU übernimmt die Einstellungen automatisch.

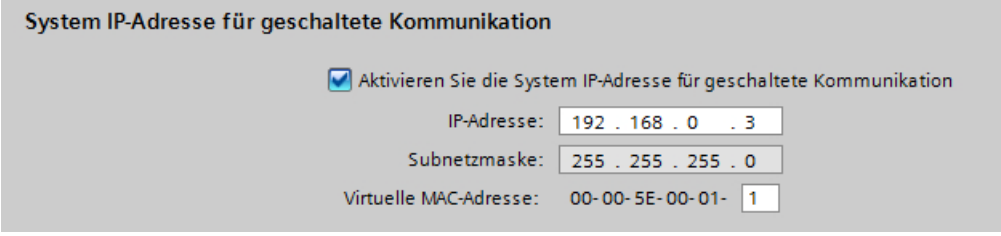

Weitere Informationen zur System IP-Adresse finden Sie im "Funktionshandbuch Kommunikation".

Zyklusüberwachungszeit einstellen

STEP 7 vergibt für die Mindestzykluszeit und die maximale Zykluszeit Standardwerte.

Die voreingestellten Werte werden Ihnen in den CPU-Eigenschaften, im Bereich "Zyklus" angezeigt.

Hinweis

Zykluszeit möglichst groß einstellen

Wählen Sie die Mindestzykluszeit und die maximale Zykluszeit möglichst groß, so wie es Ihr Prozess erlaubt.

- Die Zeit für die laufende Synchronisation der beiden CPUs im redundanten Betrieb geht in die Zykluszeit ein.
- Beim Systemzustandsübergang SYNCUP → RUN-Redundant kann es zu einer temporären Zykluszeiterhöhung kommen.

Wenn nur eine CPU den Prozess steuert (Systemzustand RUN-Solo), dann ist die Zykluszeit wesentlich kürzer als im redundanten Betrieb.

IO-Devices anlegen

Fügen Sie den R/H-CPUs IO-Devices mit Systemredundanz S2 hinzu. Gehen Sie dazu folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog auf den Ordner mit den gewünschten IO-Devices.
2. Öffnen Sie den Ordner mit dem gesuchten IO-Device-Typ. Sie sehen alle Artikelnummern des angewählten IO-Device-Typs.
3. Klicken Sie auf eine IO-Device-Artikelnummer, um sich in der Palette "Information" über das gewählte IO-Device zu informieren.
4. Stellen Sie in der Klappliste "Version" die passende Firmware-Version Ihres IO-Devices ein.
5. Legen Sie das IO-Device und einen Baugruppenträger an. Sie haben dazu folgende Möglichkeiten:
   - Ziehen Sie per Drag & Drop das IO-Device aus dem Hardware-Katalog in die Netzsicht.
   - Doppelklicken Sie auf den IO-Device-Eintrag im Hardware-Katalog.
6. Ziehen Sie die gewünschten Module auf die jeweiligen Steckplätze des IO-Device.
7. Fügen Sie gegebenenfalls weitere IO-Devices hinzu.

IO-Devices dem redundanten System zuordnen

Um IO-Devices dem redundanten System S7-1500R/H zuzuordnen, verbinden Sie jedes IO-Device mit jeder CPU.

Gehen Sie dazu folgendermaßen vor:

1. Ziehen Sie per Drag & Drop eine Linie zwischen der PROFINET-Schnittstelle des IO-Devices und der PROFINET-Schnittstelle X1 der linken CPU.
2. Ziehen Sie per Drag & Drop eine Linie zwischen der PROFINET-Schnittstelle des IO-Devices und der PROFINET-Schnittstelle X1 der rechten CPU.
3. Ordnen Sie genauso weitere IO-Devices den beiden CPUs zu.
   Die IO-Devices werden systemredundant an das S7-1500R/H-System angebunden.

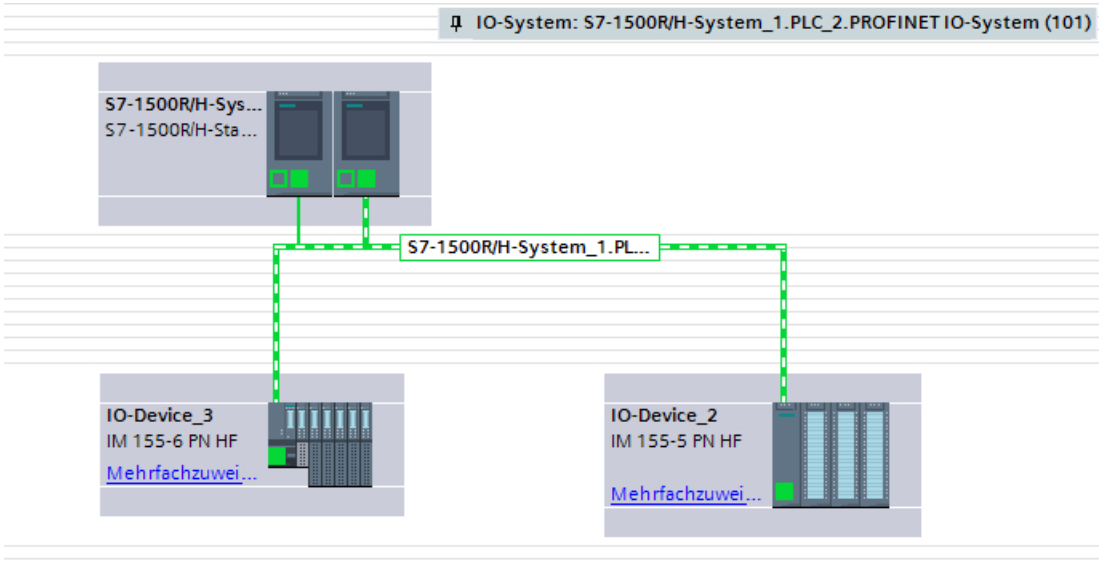

MRP-Ringkonfiguration erstellen

MRP-Rolle der R/H-CPUs einstellen

Sobald Sie in STEP 7 ein redundantes System S7-1500R/H anlegen, vergibt STEP 7 automatisch für die PROFINET-Schnittstellen X1 der beiden CPUs die MRP-Rolle "Manager (auto)".

MRP-Rolle für weitere Teilnehmer des Rings festlegen

Um die Medienredundanz für weitere Teilnehmer im Ring festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 die PROFINET-Schnittstelle X1 einer der beiden CPUs des redundanten Systems S7-1500R/H.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Erweiterte Optionen" > "Medienredundanz".

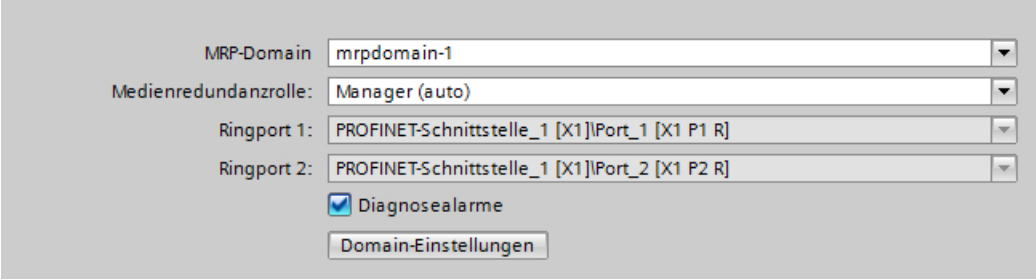

3. Klicken Sie auf die Schaltfläche "Domain-Einstellungen".
   STEP 7 zeigt im Inspektorfenster die Eigenschaften der MRP-Domain an, in der sich die PROFINET-Schnittstelle X1 der CPU befindet.
4. Vergeben Sie in der Tabelle "Geräte" in der Spalte "MRP-Rolle" für alle weiteren Teilnehmer des Rings die MRP-Rolle "Client".

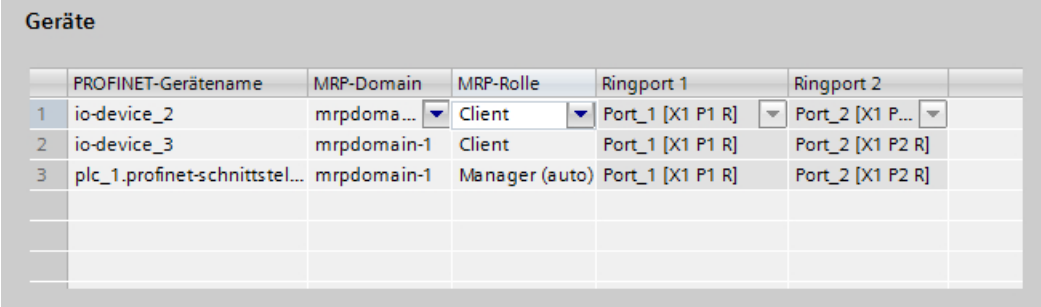

Teilnehmer außerhalb des STEP 7-Projekts parametrieren

Stellen Sie für Teilnehmer des Rings, die sich nicht in STEP 7 befinden, die MRP-Rolle "Client" ein.
Zum Beispiel stellen Sie für einen Switch die MRP-Rolle "Client" über das Webinterface des Switchs ein.

Betriebs- und Systemzustände

Übersicht

Die S7-1500R/H CPUs verfügen über die gleichen Betriebszustände wie S7-1500 Standard-CPUs und zusätzliche Betriebszustände.

Systemzustände bilden einen Zustand ab, der aus den Betriebszuständen beider CPUs im redundanten System resultiert.

Weitere Informationen finden Sie im Kapitel "Betriebs- und Systemzustände" im Systemhandbuch Redundantes System S7-1500R/H.

Betriebszustände

Betriebszustände beschreiben das Verhalten einer einzelnen CPU zu einem bestimmten Zeitpunkt. Das Wissen über die Betriebszustände der CPUs ist nützlich für die Programmierung des Anlaufs, den Test und die Fehlerdiagnose. Die Status-LEDs auf der Vorderseite der CPU und das CPU-Display geben den aktuellen Betriebszustand an.

Die S7-1500R/H CPUs verfügen wie S7-1500 Standard-CPUs über die Betriebszustände STOP, ANLAUF und RUN. Für den Betrieb als redundantes System kann eine der beiden CPUs einen weiteren Betriebszustand einnehmen, SYNCUP zur Synchronisation der beiden Teilsysteme. Der Betriebszustand RUN gliedert sich für redundante Systeme in RUN, RUN-Syncup und RUN-Redundant.

Systemzustände

Die Systemzustände ermöglichen die direkte Beurteilung des Verhaltens eines redundanten Systems. Sie resultieren aus der Kombination der Betriebszustände der einzelnen CPUs.

Betriebs- und Systemzustände im Überblick

Das folgende Bild zeigt die möglichen Betriebszustände der CPUs und die daraus resultierenden Systemzustände.

Generell sind beide CPUs gleichberechtigt, sodass jede CPU entweder Primary- oder Backup-CPU sein kann. Die CPU, die Sie zuerst von STOP nach RUN geschaltet haben, wird zur Primary-CPU.

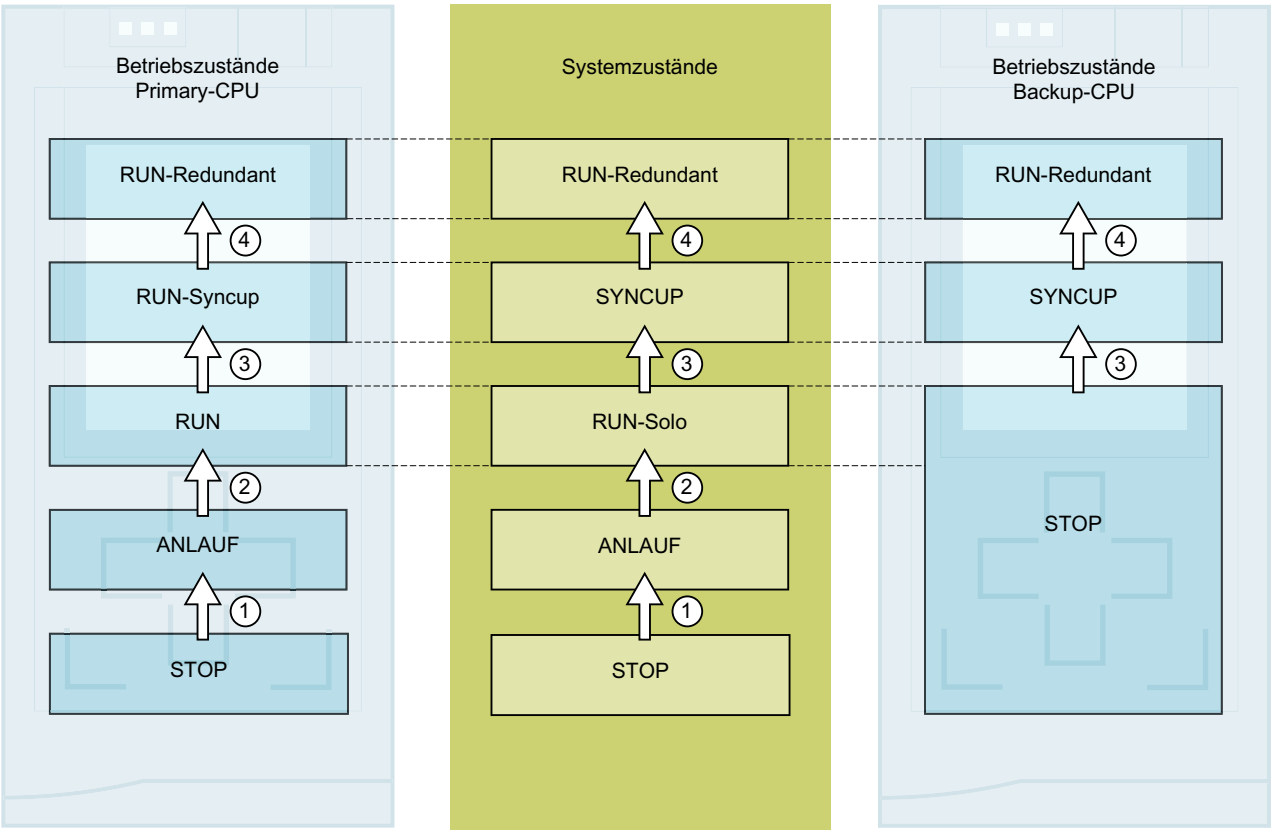

In der folgenden Tabelle finden Sie einen Überblick, wie das redundante System hochläuft und dabei die verschiedenen Betriebs- und Systemzustände durchläuft. Die folgende Ausgangssituation und die Handlungsschritte sind ein Beispiel.

Die Betriebs- und Systemzustände sind in den nachfolgenden Kapiteln detailliert beschrieben.

Tabelle 1-84 Hochlauf des redundanten Systems

Nr. im Bild	Primary-CPU	Systemzustand	Backup-CPU
Ausgangssituation: Beide CPUs befinden sich im Betriebszustand STOP. Die Betriebsartenschalter befinden sich ebenfalls in Stellung STOP.			
1. Schritt: Schalten Sie den Betriebsartenschalter an der CPU, die **Primary-CPU** sein soll, von STOP nach RUN.			
①	Die CPU geht in den ANLAUF und bearbeitet den Anlauf-OB 100 und ggf. weitere Anlauf-OBs.	STOP → ANLAUF	Die CPU bleibt im STOP.
②	Nach erfolgreichem ANLAUF wechselt die CPU in RUN. Die CPU läuft im RUN, wie eine Standard-CPU und bearbeitet ihr Anwenderprogramm.	ANLAUF → RUN-Solo	Die CPU bleibt im STOP.
2. Schritt: Schalten Sie den Betriebsartenschalter an der **Backup-CPU** von STOP nach RUN.			

Nr. im Bild	Primary-CPU	Systemzustand	Backup-CPU
③	RUN → RUN-Syncup	RUN-Solo → SYNCUP	STOP → SYNCUP
	Die beiden Anwenderprogramme werden für den redundanten Betrieb synchronisiert. Die Primary-CPU kopiert die Lade- und Arbeitsspeicherinhalte in die Backup-CPU. Die Backup-CPU holt die Anwenderprogramm-Bearbeitung der Primary-CPU auf. Nach erfolgreicher Synchronisation besitzen beide CPUs identische Speicherinhalte.		
④	RUN-Syncup → RUN-Redundant	SYNCUP → RUN-Redundant	SYNCUP → RUN-Redundant
	Nach dem SYNCUP gehen beide CPUs in den Betriebszustand RUN-Redundant. Beide CPUs bearbeiten synchron das Anwenderprogramm.		

Betriebszustand ANLAUF

Anlaufbearbeitung (nur in der Primary-CPU)

Der ANLAUF wird nur von der Primary-CPU durchgeführt.

Die CPU verhält sich im ANLAUF genauso wie eine S7-1500-Standard-CPU.

Verhalten

Bevor die CPU mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, wird ein Anlaufprogramm bearbeitet.

Im ANLAUF vergleicht die CPU den vorhandenen Peripherieausbau mit der Hardware-Konfiguration, die Sie mit STEP 7 erstellt haben.

Im Anlaufprogramm können Sie durch entsprechende Programmierung von Anlauf-OBs Initialisierungs-Variablen für Ihr zyklisches Programm festlegen. Sie haben die Möglichkeit, keinen Anlauf-OB, einen Anlauf-OB oder mehrere Anlauf-OBs zu programmieren.

Besonderheiten

- Alle Ausgänge sind deaktiviert bzw. reagieren, wie für das jeweilige Modul parametriert: Sie liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.
- Das Prozessabbild ist initialisiert.
- Das Prozessabbild wird nicht aktualisiert.
  Um den aktuellen Zustand von Eingängen im ANLAUF zu lesen, können Sie über direkten Peripheriezugriff auf Eingänge zugreifen.
  Um Ausgänge im ANLAUF zu initialisieren, können Sie Werte über das Prozessabbild oder über direkten Peripheriezugriff schreiben. Die Werte werden beim Übergang in den Betriebszustand RUN an die Ausgänge ausgegeben.
- Die CPU läuft immer im Warmstart an.
  - Wenn Sie Daten als remanent definieren, bleibt deren Inhalt bei einem Programmanlauf nach STOP oder Spannungsausfall erhalten.
  - Die nicht remanenten Merker, Zeiten und Zähler sind initialisiert.
  - Die nicht remanenten Variablen in Datenbausteinen sind initialisiert.

- Im ANLAUF läuft noch keine Zykluszeitüberwachung.
- Die CPU arbeitet die Anlauf-OBs in der Reihenfolge der Anlauf-OB-Nummern ab. Unabhängig von der gewählten Anlaufart bearbeitet die CPU alle programmierten Anlauf-OBs (siehe Bild unten, "Einstellen des Anlaufverhaltens").
- Falls ein entsprechendes Ereignis auftritt, kann die CPU folgende OBs im Anlauf starten:
  - OB 82: Diagnosealarm
  - OB 83: Ziehen/Stecken von Modulen
  - OB 86: Baugruppenträgerfehler
  - OB 121: Programmierfehler (nur bei globaler Fehlerbehandlung)
  - OB 122: Peripheriezugriffsfehler (nur bei globaler Fehlerbehandlung)

  Alle anderen OBs startet die CPU erst beim Übergang in den Betriebszustand RUN.

Siehe auch

Übersicht (Seite 1104)

Betriebszustand STOP

Verhalten

Die CPUs des redundanten Systems verhalten sich im Betriebszustand STOP genauso wie S7-1500-Standard-CPUs.

Im Betriebszustand STOP führt die CPU das Anwenderprogramm nicht aus.

Wenn beide CPUs im Betriebszustand STOP sind, sind alle Ausgänge deaktiviert bzw. reagieren, wie für das jeweilige Modul parametriert: Sie liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und halten damit den gesteuerten Prozess in einem sicheren Betriebszustand.

Besonderheiten

Die Backup-CPU baut im Betriebszustand STOP keine Verbindungen zu den IO-Devices auf.

Die Primary-CPU baut im Betriebszustand STOP Verbindungen zu den IO-Devices auf. Die Primary-CPU aktiviert auch im Betriebszustand STOP die System IP-Adresse, sofern die System IP-Adresse projektiert wurde.

Wenn sich beide CPUs im STOP befinden und Sie eine Konfiguration in eine CPU laden, dann müssen Sie auf Folgendes achten:

- Die CPU, in die Sie die Konfiguration geladen haben, müssen Sie zuerst starten, damit sie Primary-CPU wird/bleibt.
- Beim Laden in die Primary-CPU werden auch im Betriebszustand STOP die angeschlossenen IO-Devices entsprechend der geladenen Hardware-Konfiguration parametriert.

Siehe auch

Übersicht (Seite 1104)

Betriebszustand SYNCUP

Verhalten

Das Betriebssystem synchronisiert im Betriebszustand SYNCUP die Backup-CPU mit der Primary-CPU. Die Primary-CPU ist im Betriebszustand RUN-Syncup und steuert den Prozess.

Im Gegensatz zur Primary-CPU durchläuft die Backup-CPU den Betriebszustand ANLAUF nicht.

Weitere Informationen finden Sie im Kapitel Systemzustand SYNCUP (Seite 1111).

Siehe auch

Übersicht (Seite 1104)

Betriebszustände RUN

Betriebszustände RUN

Die Primary-CPU durchläuft mehrere Betriebszustände auf dem Weg zum Systemzustand RUN-Redundant:

- RUN
- RUN-Syncup
- RUN-Redundant

Die Backup-CPU kennt von diesen Betriebszuständen nur den Betriebszustand RUN-Redundant.

Betriebszustand RUN

Die Primary-CPU verhält sich im Betriebszustand RUN genauso wie eine S7-1500-Standard-CPU. Sie führt die zyklische, zeit- und alarmgesteuerte Programmbearbeitung allein durch.

Adressen, die im Prozessabbild "Automatische Aktualisierung" liegen, werden in jedem Programmzyklus automatisch aktualisiert. Weitere Informationen finden Sie im Kapitel "Prozess- und Teilprozessabbilder" des Systemhandbuchs Redundantes System S7-1500R/H.

Nachdem die CPU die Ausgänge geschrieben und die Eingänge gelesen hat, arbeitet sie das zyklische Programm beginnend mit der ersten Anweisung bis zur letzten Anweisung ab. Ereignisse mit höherer Priorität z. B. Prozessalarme, Diagnosealarme und Kommunikation können den zyklischen Programmfluss unterbrechen und die Zykluszeit verlängern.

Wenn Sie eine Mindestzykluszeit parametriert haben, beendet die CPU den Zyklus erst nach dem Ablauf der Mindestzykluszeit, auch wenn das Anwenderprogramm in kürzerer Zeit durchlaufen wurde.

Das Betriebssystem überwacht die Laufzeit des zyklischen Programms auf eine projektierbare Obergrenze, die maximale Zykluszeit. Sie können durch Aufruf der Anweisung RE_TRIGR an jeder Stelle Ihres Programms diese Zeitüberwachung neu starten.

Wenn das zyklische Programm die Zyklusüberwachungszeit überschreitet, wird gegebenenfalls der Zeitfehler-OB (OB 80) gestartet. Weitere Informationen finden Sie im Kapitel "Startereignisse" des Systemhandbuchs Redundantes System S7-1500R/H.

Besonderheiten im Betriebszustand RUN

Im nicht redundanten Betrieb sind die CPUs unabhängig voneinander. Sie können unterschiedliche Projekte besitzen.

Betriebszustand RUN-Syncup

Im Betriebszustand RUN-Syncup synchronisiert sich die Backup-CPU mit der Primary-CPU. In der Backup-CPU läuft gleichzeitig der SYNCUP, der vorübergehend Einfluss auf die Primary-CPU hat (z. B. Verzögerung von asynchronen Diensten, Zykluszeitverlängerung durch Zusammenstellung der Ladespeicherinhalte). Weitere Informationen finden Sie im Kapitel "Systemzustand SYNCUP" des Systemhandbuchs Redundantes System S7-1500R/H.

Das Betriebssystem synchronisiert den Arbeitsspeicherinhalt der Primary-CPU konsistent an einem Zykluskontrollpunkt. Während der Synchronisation verlängert sich der Programmzyklus in der Primary-CPU entsprechend der zu synchronisierenden Datenmenge, zusätzlich zur sonstigen Zykluszeit der Applikation.

Betriebszustand RUN-Redundant

Im Betriebszustand RUN-Redundant bearbeiten beide CPUs synchron das Anwenderprogramm.

Im Betriebszustand RUN-Redundant ist kein Laden von Objekten (z. B. Bausteine) im redundanten Betrieb möglich; weder ein "Laden in Gerät" noch ein "Laden von Gerät" ist möglich. Außerdem ist das Arbeiten mit Momentaufnahmen für Variablen eines Datenbausteins in diesem Systemzustand nicht möglich.

Siehe auch

Übersicht (Seite 1104)

Systemzustand SYNCUP

Systemzustand SYNCUP

Im Systemzustand SYNCUP synchronisiert das Betriebssystem die Anwenderprogramme in beiden CPUs für den redundanten Betrieb. Das Betriebssystem synchronisiert im Systemzustand SYNCUP die Backup-CPU mit der Primary-CPU. Die Backup-CPU ist im Betriebszustand SYNCUP; die Primary-CPU ist im Betriebszustand RUN-Syncup und steuert den Prozess. Mit dem Systemzustand SYNCUP geht das redundante System vom RUN-Solo in den Systemzustand RUN-Redundant über. Beide CPUs bearbeiten danach synchron das gleiche Anwenderprogramm. Bei einem Ausfall der Primary-CPU im redundanten Betrieb übernimmt die Backup-CPU an der Unterbrechungsstelle als neue Primary-CPU die Kontrolle des Prozesses.

Eigenschaften

- Eine zu Beginn des Systemzustands SYNCUP zwischen der Primary-CPU und dem PG/PC bestehende Online-Verbindung wird während des SYNCUP abgebrochen. Sie wird nach Beendigung des SYNCUP nicht automatisch wieder aufgebaut.
- Wenn Sie die Anweisungen RDREC und WRREC während des Systemzustands SYNCUP aufrufen, werden stets die Fehlercodes 0x7001 und 0x7002 zurückgeliefert. Auch im Fehlerfall, wenn z. B. die Datensatzquelle oder das Datensatzziel nicht vorhanden ist, wird kein anderer Fehlercode geliefert. In diesem Fall wird der korrekte Fehlercode erst nach Beenden des SYNCUP geliefert.
  Wenn Sie die Anweisungen RDREC und WRREC während des Systemzustands SYNCUP aufrufen, wird kein Datensatzauftrag angestoßen. Der Anstoß des Datensatzauftrags efolgt erst, wenn Sie die Anweisung nach Beendigung des SYNCUP erneut aufrufen. Es wird daher empfohlen, RDREC und WRREC stets zyklisch aufzurufen.

Weitere Eigenschaften bzw. Auswirkungen des SYNCUP auf das Anwenderprogramms finden Sie im Kapitel "Systemzustand SYNCUP" des Systemhandbuchs "SIMATIC S7-1500R/H".

Siehe auch

Übersicht (Seite 1104)

Grundlagen zur Programmbearbeitung

Programmierung von S7-1500R/H CPUs

Anwenderprogramm für das redundante System S7-1500R/H

Für den Entwurf und die Programmierung des Anwenderprogramms gelten für das redundante System S7-1500R/H die gleichen Regeln wie für das Automatisierungssystem S7-1500.

Im redundanten Betrieb ist das Anwenderprogramm in beiden CPUs identisch hinterlegt. Die beiden CPUs bearbeiten das Anwenderprogramm ereignissynchron.

Aus Sicht der Anwenderprogrammbearbeitung verhält sich das redundante System S7-1500R/H wie das Automatisierungssystem S7-1500. Die Synchronisation ist im Betriebssystem integriert und läuft zwischen Primary- und Backup-CPU automatisch und verdeckt ab.

Spezifische Anweisungen und Bausteine für das redundante System S7-1500R/H

Für das redundante System S7-1500R/H stehen Ihnen spezifische Anweisungen und OBs zur Verfügung.

Mit der Anweisung "RH_CTRL" beeinflussen Sie Abläufe bei R/H-Systemen. Mit dem Firmware-Stand V2.6 sperren Sie den Systemzustand SYNCUP bzw. geben den Systemzustand SYNCUP wieder frei. Das Ziel ist, den Systemzustand SYNCUP bei Bedarf nur in weniger kritischen Prozessphasen zu erlauben (siehe Kapitel "Systemzustand SYNCUP sperren/freigeben mit der Anweisung RH_CTRL" im Systemhandbuch Redundantes System S7-1500R/H).

Mit der Anweisung "RH_GetPrimaryID" lesen Sie im Anwenderprogramm aus, welche CPU gerade die Primary-CPU ist. Weitere Informationen finden Sie im Kapitel "Primary-CPU ermitteln mit der Anweisung 'RH_GetPrimaryID'" im Systemhandbuch Redundantes System S7-1500R/H.

Zusätzlich zu den OBs der S7-1500 CPU steht Ihnen der OB 72 zur Verfügung. Mit dem OB 72 behandeln Sie z. B. Redundanzfehler.

Besonderheiten bei der Programmbearbeitung

- Sie erstellen das Anwenderprogramm für das redundante System S7-1500R/H in der oberen CPU (z. B. PLC_1) in der Projektnavigation von STEP 7.
- Das redundante System S7-1500R/H unterstützt einige Anweisungen der S7-1500 CPUs nicht. Anweisungen, die das redundante System S7-1500R/H nicht unterstützt, sehen Sie in STEP 7 in der Task Card "Anweisungen" gegraut.
  Die nicht unterstützten Anweisungen im Programmcode zeigt STEP 7 rot an. Wenn Sie Programmcode mit nicht unterstützten Anweisungen übersetzen, dann liefert STEP 7 eine entsprechende Fehlermeldung.
  Die nicht unterstützten Anweisungen finden Sie im Kapitel "Grundlagen zur Programmbearbeitung > Einschränkungen" im Systemhandbuch Redundantes System S7-1500R/H.
- Bei Anweisungen mit dem Bausteinparameter "LADDR" bestimmen Sie über diesen Parameter, welche der beiden CPUs das Ziel dieser Anweisung ist.
  Beispiel: Um die I&M-Daten der CPU mit der Redundanz-ID 1 auszulesen, geben Sie am Bausteinparameter "LADDR" der Anweisung Get_IM_Data die HW-Kennung 65149 (bzw. Systemkonstante "Local1") an.
- Verhalten asynchroner Anweisungen
  - Normalerweise verzögern asynchrone Anweisungen den SYNCUP-Prozess, bis ihre Ausführung beendet ist.
  - Wenn eine asynchrone Anweisung gestartet wird, während der SYNCUP-Prozess läuft, wird sie in einen Wartezustand versetzt. Sie bleibt in diesem Wartezustand, bis der SYNCUP-Prozess beendet ist. Wenn der SYNCUP-Prozess beendet ist, geht die asynchrone Anweisung in den aktiven Zustand.

- Verhalten des redundanten Systems S7-1500R/H bei Überschreitungen der maximalen Zykluszeit, falls Sie keinen Zeitfehler-OB angelegt haben
  Das erste Überschreiten der maximalen Zykluszeit in den Systemzuständen RUN-Redundant und SYNCUP führt zum Redundanzverlust: Die Primary-CPU nimmt den Betriebszustand RUN und die Backup-CPU den Betriebszustand STOP ein.
  Die zweite Überschreitung innerhalb desselben Programmzyklus bewirkt, dass die Primary-CPU den Betriebszustand STOP einnimmt.
  Weitere Details finden Sie im Kapitel "Grundlagen zur Programmbearbeitung" des Systemhandbuchs S7-1500R/H.
- Verhalten des redundanten Systems S7-1500R/H bei Überschreitungen der maximalen Zykluszeit, falls Sie einen Zeitfehler-OB angelegt haben
  Beim ersten Überschreiten der maximalen Zykluszeit in den Systemzuständen RUN-Redundant und SYNCUP verbleibt das R/H-System in seinem Systemzustand; der OB 80 wird aufgerufen und die Zykluszeit wird zurückgesetzt.
  Die zweite Überschreitung innerhalb desselben Programmzyklus führt zum Redundanzverlust: Die Primary-CPU nimmt den Betriebszustand RUN und die Backup-CPU den Betriebszustand STOP ein. Der OB 80 wird aufgerufen und die Zykluszeit wird zurückgesetzt.
  Die dritte Überschreitung innerhalb desselben Programmzyklus bewirkt, dass die Primary-CPU den Betriebszustand STOP einnimmt.
  Weitere Details finden Sie im Kapitel "Grundlagen zur Programmbearbeitung" des Systemhandbuchs S7-1500R/H.

Siehe auch

Organisationsbausteine (Seite 1113)

Organisationsbausteine

CPU-Redundanzfehler-OB

Beschreibung

Das Betriebssystem jeder CPU eines R/H-Systems ruft den CPU-Redundanzfehler-OB (OB72) auf, wenn eines der folgenden Ereignisse auftritt:

- Redundanzverlust durch Ausfall einer CPU
- Redundanzverlust durch Übergang einer CPU in den Betriebszustand STOP (vom Anwender oder vom System ausgelöst)
- R/H-System hat den Systemzustand RUN-Redundant eingenommen

Struktur der Startinformation

Ein CPU-Redundanzfehler-OB hat die folgende Startinformation:

- Wie bei S7-400-CPUs:

Variable	Datentyp	Beschreibung
OB72_EV_CLASS	BYTE	Ereignisklasse und Kennungen: • B#16#73: kommendes Ereignis
OB72_FLT_ID	BYTE	Fehlercode Mögliche Werte: • B#16#01: Redundanzverlust durch Ausfall einer CPU • B#16#02: Redundanzverlust durch Übergang einer CPU in den Betriebszustand STOP • B#16#03: R/H-System hat den Systemzustand RUN-Redundant eingenommen
OB72_PRIORITY	BYTE	Prioritätsklasse
OB72_OB_NUMBER	BYTE	OB-Nummer (72)
OB72_RESERVED_1	WORD	• High byte: B#16#00 • Low byte: reserviert
OB72_INFO_1	WORD	W#16#0000
OB72_INFO_2	WORD	W#16#0000
OB72_INFO_3	WORD	W#16#0000
OB72_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

- Optimierte Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_ANY	HW-Kennung des fehlerhaften Hardware-Objekts
Event_Class	BYTE	Identisch zur Variable OB72_EV_CLASS in der obigen Tabelle
Fault_ID	BYTE	Identisch zur Variable OB72_FLT_ID in der obigen Tabelle

Startereignisse des OB72

OB72_EV_CLASS bzw. Event_Class	OB72_FLT_ID bzw. Fault_ID	Startereignis
B#16#73	B#16#01	Redundanzverlust durch Ausfall einer CPU
B#16#73	B#16#02	Redundanzverlust durch Übergang einer CPU in den Betriebszustand STOP
B#16#73	B#16#03	R/H-System hat den Systemzustand RUN-Redundant eingenommen

Laden einer S7-1500R/H CPU

Projektdaten in die CPUs laden

Projektdaten lassen sich über eine Online-Verbindung vom PG/PC oder HMI-Gerät in die CPUs des S7-1500R/H-Systems laden. Die Projektdaten können entweder in die Primary-CPU oder in die Backup-CPU geladen werden, ein gleichzeitiger Zugriff auf beide CPUs ist nicht möglich.

Standardmäßig werden die Projektdaten in die Primary-CPU geladen. Im Systemzustand SYNCUP werden dann die Projektdaten von der Primary-CPU in die Backup-CPU kopiert.

Projektdaten lassen sich auch in die Backup-CPU laden. Das ist dann sinnvoll, wenn die Backup-CPU bei einem Neustart Primary-CPU werden soll.

Die kompletten Projektdaten (alle Konfigurationsdaten und das komplette Anwenderprogramm) lassen sich nur im Betriebszustand STOP in eine CPU laden.

Das Anwenderprogramm lässt sich im Systemzustand RUN-Solo auch in die im RUN befindliche Primary-CPU laden. Während des Ladens hält die Primary-CPU die Kontrolle über den Prozess weiterhin aufrecht, es gibt keinen Stillstand der Anlage.

Siehe auch

Projektdaten in die Primary-CPU laden (Seite 1115)

Primary-CPU nach dem Laden starten (Seite 1116)

Projektdaten in die Backup-CPU laden (Seite 1117)

Anwenderprogramm in die Primary-CPU laden (Seite 1118)

Projektdaten in die Primary-CPU laden

Standardmäßig laden Sie die Projektdaten in die Primary-CPU. Um die kompletten Projektdaten (alle Konfigurationsdaten und das komplette Anwenderprogramm) zu laden, muss sich die CPU im Betriebszustand STOP befinden. Im Systemzustand SYNCUP werden dann die Projektdaten von der Primary-CPU in die Backup-CPU kopiert.

Voraussetzungen

- Die IP-Adressen wurden vergeben.
- Die Primary-CPU befindet sich im Betriebszustand STOP.

Vorgehen

Um Projektdaten in die Primary-CPU zu laden, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Projektnavigation das S7-1500R/H-System.
2. Rufen Sie mit der rechten Maustaste das Kontextmenü auf.

3. Wählen Sie aus dem Kontextmenü den Befehl "Laden in Gerät > Hardware und Software (nur Änderungen)".
   Der Dialog "Erweitertes Laden" öffnet sich.
4. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" das Subnetz aus.
5. Wählen Sie aus der Klappliste "PG/PC-Schnittstellte" den Adapter aus.
6. Wählen Sie aus der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle aus, zu der eine Verbindung vom PG/PC aus besteht.
   Alternativ wählen Sie den Eintrag "Versuche alle Schnittstellen".
7. Klicken Sie auf die Schaltfläche "Suche starten".
   Die Tabelle "Zielgerät auswählen" zeigt die CPUs des S7-1500R/H-Systems mit ihren Rollen an. Die Primary-CPU ist standardmäßig ausgewählt.
8. Klicken Sie auf die Schaltfläche "Laden".

Ergebnis

Falls es erforderlich ist, werden vor dem Laden die Projektdaten übersetzt.

Der Dialog "Ergebnisse des Ladevorgangs" zeigt die Ergebnisse des Ladevorgangs an.

Primary-CPU nach dem Laden starten

Nach dem Laden der Projektdaten starten Sie die Primary-CPU.

Voraussetzungen

- Es wurden fehlerfrei übersetzte, konsistente Projektdaten geladen.
- Der Betriebsartenschalter der Primary-CPU steht auf RUN.

Vorgehen

⚠ WARNUNG

Starten der Primary-CPU mit fehlerhaftem Anwenderprogramm

Stellen Sie vor dem Starten der Primary-CPU sicher, dass ein eventuell fehlerhaftes Anwenderprogramm keine Sach- und Personenschäden und keine gefährlichen Zustände verursachen kann.

Um die Primary-CPU nach dem Laden zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Dialog "Ergebnisse des Ladevorgangs" in der Spalte "Aktion" den Eintrag "Baugruppe starten".
2. Klicken Sie auf die Schaltfläche "Fertig stellen".
   Die Primary-CPU geht in den Betriebszustand RUN.
3. Schalten Sie die Backup-CPU in den Betriebszustand RUN.

Hinweis

Rollenwechsel beim Laden

Achten Sie auf einen möglichen Rollenwechsel zwischen Primary- und Backup-CPU kurz vor, während oder nach dem Laden.

Beim Laden kann es zu einem Rollenwechsel kommen, wenn die Primary-CPU während des Ladens ausfällt (Netzausfall, Hardware-Defekt) oder sich im Betriebszustand STOP befindet und eine der folgenden Bedingungen eintritt:

- Sie schalten die Backup-CPU in dieser Zeit in den Betriebszustand RUN.
- Sie schalten die Backup-CPU in dieser Zeit auf NETZ-EIN und das Anlaufverhalten wurde in STEP 7 so parametriert, dass die CPU nach dem Schalten auf NETZ-EIN automatisch in den Betriebszustand RUN geht.

Im Fall eines Rollenwechsels startet die neue Primary-CPU mit den alten Projektdaten. Die neuen Projektdaten in der jetzigen Backup-CPU werden während der Synchronisation der beiden CPUs für den redundanten Betrieb mit den alten Projektdaten überschrieben.

Ergebnis

Nach erfolgreichem SYNCUP geht das S7-1500R/H-System in den redundanten Betrieb.

Projektdaten in die Backup-CPU laden

Standardmäßig laden Sie die Projektdaten in die Primary-CPU. Sie können die Projektdaten auch in die Backup-CPU laden. Das ist dann sinnvoll, wenn die Backup-CPU bei einem Neustart Primary-CPU mit ihren Projektdaten werden soll.

Während des Ladens der Projektdaten steuert die Primary-CPU weiterhin den Prozess.

Nach dem Laden müssen Sie die Primary-CPU in den Betriebszustand STOP und die Backup-CPU in den Betriebszustand RUN schalten. Damit wird die bisherige Backup-CPU zur neuen Primary-CPU.

Hinweis

Laden in die Backup-CPU

Wenn das Projekt remanente Daten verwendet, dann läuft die Backup-CPU gegebenenfalls mit veralteten Daten an.

Voraussetzung

Die Backup-CPU befindet sich im Betriebszustand STOP.

Projektdaten in die Backup-CPU laden

Um Projektdaten in die Backup-CPU zu laden, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Projektnavigation mit der rechten Maustaste das S7-1500R/H-System.
2. Wählen Sie unter "Laden in Backup-CPU" den Menüpunkt "Hardware und Software (nur Änderungen)".
   Im Dialog "Erweitertes Laden" ist nun anstelle der Primary-CPU die Backup-CPU selektiert.
3. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" das Subnetz aus.
4. Wählen Sie aus der Klappliste "PG/PC-Schnittstellte" den Adapter aus.
5. Wählen Sie aus der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle aus, zu der eine Verbindung vom PG/PC aus besteht.
6. Alternativ wählen Sie den Eintrag "Versuche alle Schnittstellen".
7. Klicken Sie auf die Schaltfläche "Suche starten".
   Die Tabelle "Zielgerät auswählen" zeigt die CPUs des S7-1500R/H-Systems mit ihren Rollen an. Die Backup-CPU ist standardmäßig ausgewählt.
8. Klicken Sie auf die Schaltfläche "Laden".
   Der Dialog "Ergebnisse des Ladevorgangs" zeigt die Ergebnisse des Ladevorgangs an.

Primary- und Backup-CPU umschalten

Nach dem Laden der Projektdaten in die Backup-CPU schalten Sie die CPUs um, damit die Backup-CPU zur neuen Primary-CPU wird. Gehen Sie dazu folgendermaßen vor:

1. Schalten Sie die Primary-CPU in den Betriebszustand STOP.
2. Schalten Sie die Backup-CPU in den Betriebszustand RUN.
   Die bisherige Backup-CPU wird die neue Primary-CPU und steuert im Systemzustand RUN-Solo allein den Prozess.

Ergebnis

Das S7-1500R/H-System geht nach dem SYNCUP mit dem geänderten Anwenderprogramm wieder in den Systemzustand RUN-Redundant.

Anwenderprogramm in die Primary-CPU laden

Das Anwenderprogramm lässt sich in eine CPU im Betriebszustand RUN laden. Während des Ladens hält die Primary-CPU die Kontrolle über den Prozess weiterhin aufrecht, es gibt keinen Stillstand der Anlage.

Voraussetzung

Die Primary-CPU befindet sich im Betriebszustand RUN.

Vorgehen

Um ein geändertes Anwenderprogramm in die Primary-CPU zu laden, gehen Sie folgendermaßen vor:

1. Schalten Sie die Backup-CPU in den Betriebszustand STOP.
   Das S7-1500R/H-System wechselt vom Systemzustand RUN-Redundant in den Systemzustand RUN-Solo.
2. Laden Sie das geänderte Anwenderprogramm über "Laden in Gerät > Software (nur Änderungen)" in die Primary-CPU.
   Die Primary-CPU steuert weiterhin den Prozess.
3. Schalten Sie die Backup-CPU in den Betriebszustand RUN.

Ergebnis

Die Primary-CPU bleibt im Betriebszustand RUN und synchronisiert im SYNCUP das geänderte Anwenderprogramm in die Backup-CPU.

Das S7-1500R/H-System geht mit dem geänderten Anwenderprogramm wieder in den Systemzustand RUN-Redundant.

Kommunikation mit dem R/H-System

System IP-Adressen

Informationen zu den Einschränkungen beim R/H-System hinsichtlich der Kommunikationsfunktionen siehe Funktionsweise des redundanten Systems S7-1500R/H (Seite 1095).

Die folgenden Erläuterungen beschreiben die Möglichkeiten, die Kommunikation zum R/H-System aufrechtzuerhalten, auch wenn z. B. die prozessführende CPU ausfällt.

Die System IP-Adressen des redundanten Systems S7-1500R/H

Neben den "normalen" IP-Adressen der PROFINET-Schnittstellen unterstützt das redundante System S7-1500R/H auch **System IP-Adressen**.

System IP-Adressen verwenden Sie für die Kommunikation mit anderen Geräten (z. B. HMI-Geräte, CPUs, PG/PC). Diese Geräte kommunizieren über die System IP-Adresse immer mit der Primary-CPU des redundanten Systems. Dadurch wird z. B. sichergestellt, dass der Kommunikationspartner nach einem Ausfall der Primary-CPU im redundanten Betrieb mit der neuen Primary-CPU (vorher Backup-CPU) im Systemzustand RUN-Solo kommunizieren kann.

Zur Unterscheidung zwischen "normaler" und "System" IP-Adresse wird die normale IP-Adresse "Geräte IP-Adresse" genannt - nur im Zusammenhang mit R/H-Systemen.

Folgende System IP-Adressen existieren:

- System IP-Adresse für die PROFINET-Schnittstellen X1 der beiden CPUs (System IP-Adresse X1)
- System IP-Adresse für die PROFINET-Schnittstellen X2 der beiden CPUs (System IP-Adresse X2)

Zu jeder System IP-Adresse gehört eine virtuelle MAC-Adresse.

Vorteile der System IP-Adressen gegenüber den Geräte IP-Adressen

- Der Kommunikationspartner kommuniziert gezielt mit der Primary-CPU.
- Die Kommunikation des redundanten Systems S7-1500R/H über eine System IP-Adresse funktioniert auch bei Ausfall der Primary-CPU (Primary-Backup-Umschaltung) weiterhin.

Anwendungsfälle

Die System IP-Adressen nutzen Sie für folgende Anwendungen:

- HMI-Kommunikation mit dem redundanten System S7-1500R/H: Mit einem HMI führen bzw. beobachten Sie den Prozess auf dem redundanten System S7-1500R/H.
- Open User Communication mit dem redundanten System S7-1500R/H:
  - Eine andere CPU oder eine Applikation auf einem PC greift auf Daten des redundanten Systems S7-1500R/H zu.
  - Das redundante System S7-1500R/H greift auf ein anderes Gerät zu.

  Möglich sind TCP, UDP und ISO-on-TCP-Verbindungen.

Voraussetzungen

- Der Kommunikationspartner und die PROFINET-Schnittstellen der beiden CPUs befinden sich im selben Subnetz.
- Der Kommunikationspartner ist mit beiden CPUs verbunden, jeweils über die gleiche Schnittstelle (z. B. X2).
- Die System IP-Adresse ist aktiviert.

Kommunikation über die System IP-Adresse X2

Wenn die CPUs des redundanten Systems S7-1500R/H zwei PROFINET-Schnittstellen haben, verwenden Sie für Kommunikation mit anderen Geräten bevorzugt die PROFINET-Schnittstelle X2.

Das folgende Bild zeigt eine Konfiguration, bei der die Kommunikationspartner über die jeweiligen PROFINET-Schnittstellen X2 mit den CPUs des redundanten Systems S7-1500R/H verbunden sind.

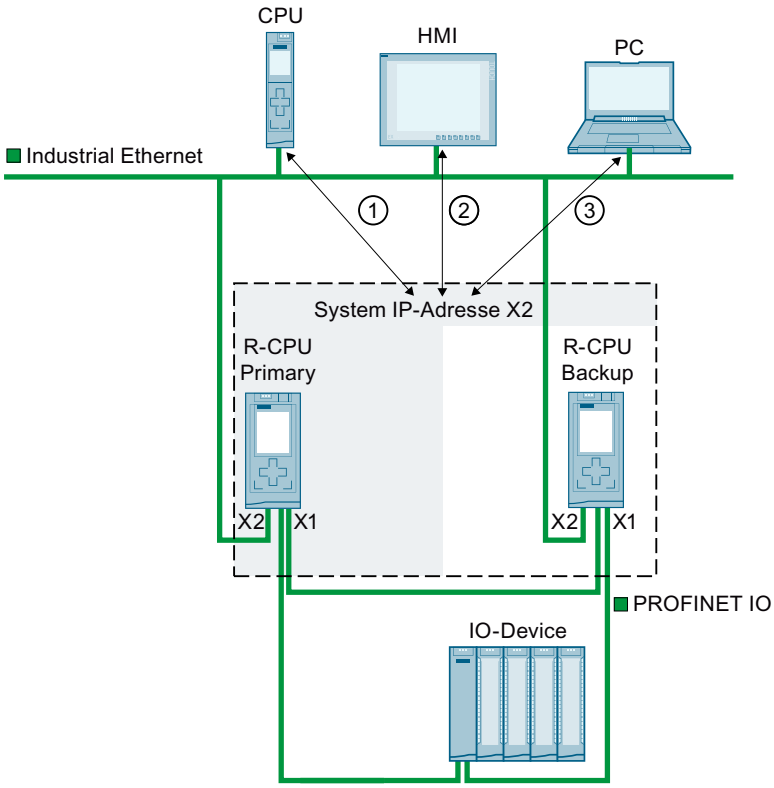

① Open User Communication zwischen einer anderen CPU und dem redundanten System S7-1500R/H

② HMI-Kommunikation mit dem redundanten System S7-1500R/H

③ Open User Communication zwischen dem redundanten System S7-1500R/H und einem PC

Kommunikation über die System IP-Adresse X1

Das folgende Bild zeigt eine Konfiguration, bei der die Kommunikationspartner mit einem Switch an den PROFINET-Ring des redundanten Systems S7-1500R/H angeschlossen sind. Der PROFINET-Ring verbindet die Kommunikationspartner mit den jeweiligen PROFINET-Schnittstellen X1 der beiden CPUs.
Da die CPU 1513R nur eine PROFINET-Schnittstelle hat, ist der Anschluss über den PROFINET-Ring die einzige Möglichkeit, um über die System IP-Adresse X1 zu kommunizieren.

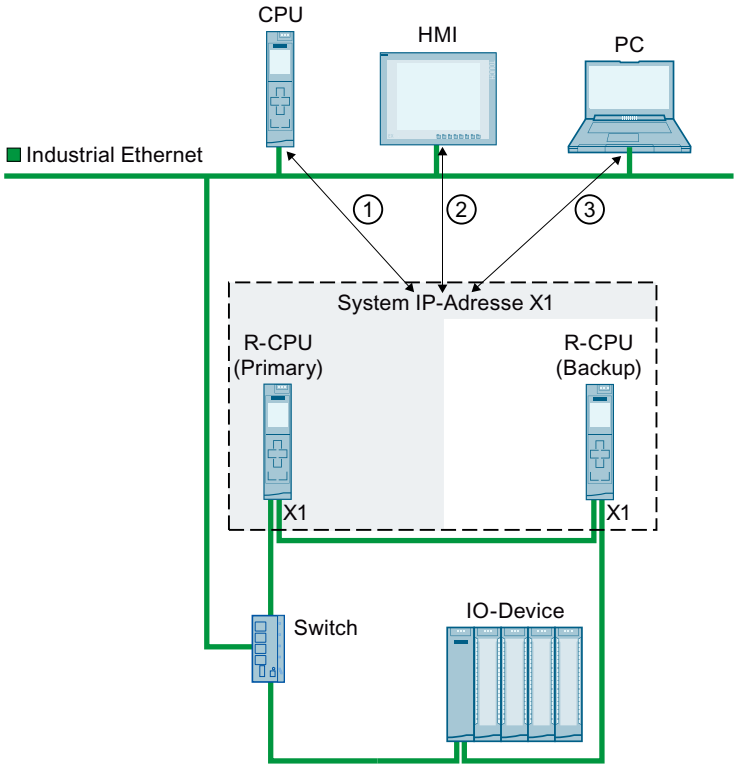

① Open User Communication zwischen dem redundanten System S7-1500R/H und einer anderen CPU

② HMI-Kommunikation mit dem redundanten System S7-1500R/H

③ Open User Communication zwischen dem redundanten System S7-1500R/H und einem PC

Kommunikation über die System IP-Adressen X1 und X2

Wenn die CPUs des redundanten Systems S7-1500R/H zwei PROFINET-Schnittstellen (X1 und X2) haben, dann können Sie für beide PROFINET-Schnittstellen die jeweiligen System IP-Adressen verwenden. PROFINET-Geräte, die mit den Schnittstellen X1 der CPUs verbunden sind, kommunizieren über die System IP-Adresse X1. PROFINET-Geräte, die mit den Schnittstellen X2 der CPUs verbunden sind, kommunizieren über die System IP-Adresse X2.

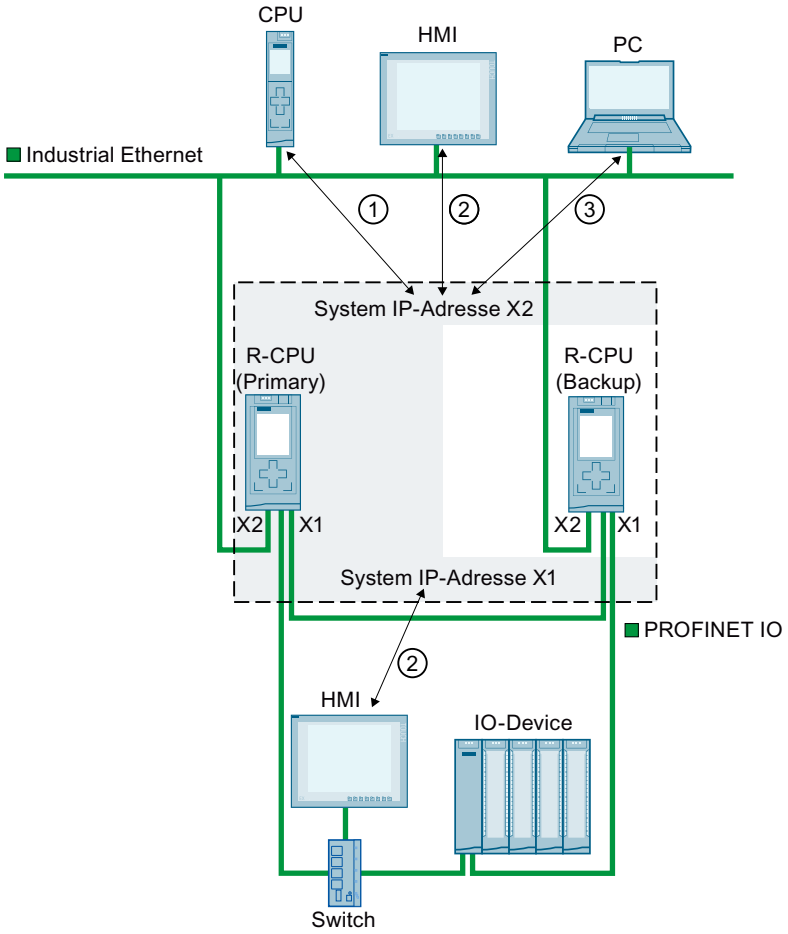

① Open User Communication zwischen dem redundanten System S7-1500R/H und einer anderen CPU.

② HMI-Kommunikation mit dem redundanten System S7-1500R/H

③ Open User Communication zwischen dem redundanten System S7-1500R/H und einem PC

System IP-Adressen aktivieren

Wenn die CPUs des redundanten Systems S7-1500R/H zwei PROFINET-Schnittstellen (X1 und X2) besitzen, dann können Sie für beide PROFINET-Schnittstellen eine System-IP-Adresse aktivieren. Im Folgenden ist beschrieben, wie Sie die System-IP-Adresse für die Schnittstelle X1 aktivieren.

Um die System-IP-Adresse für ihr redundantes System S7-1500R/H zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 die Schnittstelle X1 einer der beiden CPUs.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Ethernet-Adressen" in den Bereich "System-IP-Adresse für geschaltete Kommunikation".
3. Aktivieren Sie das Optionskästchen "Aktivieren Sie die System-IP-Adresse für geschaltete Kommunikation".
   STEP 7 legt automatisch eine System-IP-Adresse an.
4. Passen Sie die System-IP-Adresse bei Bedarf an.
5. Passen Sie bei Bedarf die virtuelle MAC-Adresse an. Vergeben Sie dazu bei "Virtuelle MAC-Adresse" für das letzte Byte einen projektweit eindeutigen Wert (Wertebereich 01_H bis FF_H).

Hinweis

Eindeutigkeit der virtuellen MAC-Addresse

Das redundante System S7-1500R/H verwendet für die System IP-Adresse und die dazugehörige virtuelle MAC-Adresse das Virtual Router Redundancy Protocol VRRP. Wenn Sie weitere Geräte mit VRRP einsetzen, z. B. Switche, achten Sie auf die Eindeutigkeit der MAC-Adressen innerhalb einer Ethernet-Broadcast-Domain.

Siehe auch

Verhalten der Verbindungen bei Primary-Backup-Umschaltung (Seite 1130)

HMI-Verbindung über die System IP-Adresse einrichten

Voraussetzungen

- Redundantes System S7-1500R/H, z. B. CPU 1513R-1PN ist in der Netzsicht platziert
- System IP-Adresse ist aktiviert
- HMI-Gerät mit PROFINET-Schnittstelle ist in der Netzsicht platziert

Vorgehen

Um eine HMI-Verbindung zu einem redundanten System S7-1500R/H einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 eine PROFINET-Schnittstelle des HMI-Geräts.
2. Ziehen Sie per Drag&Drop eine Linie zwischen der PROFINET-Schnittstelle des HMI-Geräts und einer PROFINET-Schnittstelle des redundanten Systems S7-1500R/H. Das HMI-Gerät und das redundante System S7-1500R/H sind miteinander vernetzt.
3. Klicken Sie in der Funktionsliste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

4. Ziehen Sie per Drag&Drop eine Linie zwischen HMI-Gerät und einer CPU des redundanten Systems S7-1500R/H.
   Die Liste "Verbindungspartner" öffnet sich.
5. Wählen Sie in der Liste "Verbindungspartner" das redundante System S7-1500R/H aus.

Ergebnis: Sie haben eine HMI-Verbindung zwischen dem HMI-Gerät und dem redundanten System S7-1500R/H eingerichtet. Die HMI-Verbindung nutzt die System-IP-Adresse.

HMI-Verbindung auf die Geräte IP-Adresse umstellen

Um die HMI-Verbindung fest auf die ausgewählte CPU umzustellen, deaktivieren Sie das Kontrollkästchen "Verwenden Sie die System-IP-Adresse für geschaltete Kommunikation" in den Eigenschaften der HMI-Verbindung. Die HMI-Verbindung nutzt dann die Geräte IP-Adresse der PROFINET-Schnittstelle. Wenn diese CPU ausfällt, dann fällt die HMI-Verbindung zu dieser CPU dauerhaft aus.

Hinweis

Automatische Einrichtung HMI-Verbindung

Wenn Sie vom redundanten System S7-1500R/H eine Variable per Drag & Drop in ein HMI-Bild oder in die HMI-Variablentabelle hineinziehen, richtet STEP 7 automatisch eine HMI-Verbindung ein. Diese HMI-Verbindung besteht standardmäßig zwischen der PROFINET-Schnittstelle des HMI-Geräts und der PROFINET-Schnittstelle X1 der CPU mit Redundanz-ID 1. Die Verbindung verwendet die Geräte IP-Adresse der PROFINET-Schnittstelle X1.

In den Eigenschaften der HMI-Verbindung können Sie die HMI-Verbindung auf eine System IP-Adresse umstellen.

Open User Communication über die System IP-Adresse einrichten

Das redundante System S7-1500R/H kann über Open User Communication mit anderen Geräten kommunizieren.

- Die Verbindungen richten Sie im Anwenderprogramm ein, z. B. über die Anweisung "TSEND_C".

  Hinweis

  Projektierte Verbindungen unterstützt das redundante System S7-1500R/H nicht.

- Die Verbindungen können Sie entweder über die Geräte IP-Adressen oder über die System IP-Adressen der PROFINET-Schnittstellen einrichten.
  - Wenn Sie die Verbindung über eine System IP-Adresse einrichten, dann läuft die Kommunikation immer über die Primary-CPU.
    Empfehlung: Verwenden Sie für Open User Communication immer eine System IP-Adresse.
  - Im redundanten Betrieb kann das redundante System über jede Geräte IP-Adresse Verbindungen auf- bzw. abbauen und Daten senden bzw. empfangen.
    Wenn Sie die Verbindung über eine Geräte IP-Adresse einrichten, dann läuft die Kommunikation über die zugehörige CPU. Wenn die CPU ausfällt, dann fällt die gesamte Kommunikation über die Geräte IP-Adressen dieser CPU aus.

Nutzbare Protokolle / Verbindungstypen

Neben dem unten beispielhaft erläuterten Protokoll TCP (über die Verbindungsparametrierung für die Anweisungen TSEND_C und TRCV_C eingerichtet) können Sie auch die Protokolle ISO-on-TCP, UDP und Modbus TCP mit den entsprechenden Systemdatentypen und Anweisungen einsetzen, siehe Tabelle.

Der Systemdatentyp TCON_Param wird nicht unterstützt.

Protokoll	Systemdatentyp	Anweisungen
TCP	• TCON_IP_v4 • TCON_QDN • TCON_IP_v4_SEC* • TCON_QDN_SEC*	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TCON, TSEND/TRCV oder • TCON, TUSEND/TURCV (Abbau der Verbindung über TDISCON möglich)
ISO-on-TCP	• TCON_IP_RFC	

Protokoll	Systemdatentyp	Anweisungen
UDP	• TCON_IP_v4 • TCON_IP_QDN • TADDR_Param • TADDR_SEND_QDN • TADDR_RCV_IP • TCON_QDN_SEC* • TCON_IP_v4_SEC*	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C • TUSEND/TURCV/TRCV (Abbau der Verbindung über TDISCON möglich)
Modbus TCP	• TCON_IP_v4 • TCON_QDN • TCON_IP_v4_SEC* • TCON_QDN_SEC*	• MB_CLIENT • MB_SERVER

* Das redundante System S7-1500R/H unterstützt noch keine Secure Open User Communication. Wenn Sie die Systemdatentypen TCON_IP_v4_SEC oder TCON_QDN_SEC im redundanten System S7-1500R/H verwenden, muss deren Parameter ActivateSecureConn den Wert FALSE haben. Damit sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.

Beispiel: Verbindung über die System IP-Adresse einrichten

Im Folgenden ist beschrieben, wie Sie eine Verbindung vom redundanten System S7-1500R/H zu einer anderen CPU über die System IP-Adresse einrichten.

- Im Anwenderprogramm des redundanten Systems S7-1500R/H richten Sie die Verbindung mit einer Anweisung TSEND_C ein.
- Im Anwenderprogramm der Partner CPU legen Sie eine entsprechende Anweisung TRCV_C an.

Die Vorgehensweise ist am Beispiel einer TCP-Verbindung zwischen dem redundanten System S7-1500R/H und einer CPU 1516-3 PN/DP beschrieben.

Voraussetzungen

- Lokaler Verbindungspartner: Redundantes System S7-1500R mit zwei CPUs 1513-1 PN System IP-Adresse der PROFINET-Schnittstelle X1 ist aktiviert.
- Entfernter Verbindungspartner: CPU 1516-3 PN/DP
- Die PROFINET-Schnittstellen X1 der CPUs 1513R und die PROFINET-Schnittstelle X2 der CPU 1516-3 PN/DP befinden sich im selben Subnetz.

TCP-Verbindung einrichten im redundanten System S7-1500R/H (Senden)

Um eine TCP-Verbindung zu einer anderen CPU einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie im Anwenderprogramm eine Anweisung "TSEND_C" an.
2. Selektieren Sie die Anweisung "TSEND_C".

3. Navigieren Sie ins Inspektorfenster zu "Eigenschaften" > "Konfiguration" > "Verbindungsparameter".
   Auf der linken Seite sehen Sie das redundante System S7-1500R/H als lokalen Endpunkt der Verbindung:
   - "Schnittstelle:": Die Schnittstelle X1 ist voreingestellt.
   - "Subnetz:": Wenn die Schnittstelle X1 einem S7-Subnetz zugeordnet ist, dann zeigt STEP 7 den Namen des S7-Subnetzes an.
   - **Die Option "System IP-Adresse verwenden" ist aktiviert**.
4. Wählen Sie bei "Partner" unter "Endpunkt:" die CPU 1516-3 PN/DP als Kommunikationspartner aus.
5. Wählen Sie bei "Partner" unter "Schnittstelle:" die PROFINET-Schnittstelle X2 der CPU 1516-3 PN/DP aus.
6. Wählen Sie bei "Lokal" unter "Verbindungsdaten" die Einstellung "<neu>" aus.
   STEP 7 legt im Anwenderprogramm des redundanten Systems S7-1500R/H einen Datenbaustein für die Verbindungsdaten an.
7. Wählen Sie bei "Partner" unter "Verbindungstyp:" die Einstellung "TCP" aus.
   STEP 7 legt im Anwenderprogramm der anderen CPU einen Datenbaustein für die Verbindungsdaten an.

TCP-Verbindung einrichten im Verbindungspartner CPU 1516 (Empfangen)

Legen Sie im Anwenderprogramm der CPU 1516-3PN/DP eine Anweisung TRCV_C an und parametrieren Sie diese passend ("komplementär") zum Verbindungspartner.

Verbindung über Geräte IP-Adresse einrichten

Um eine OUC-Verbindung über eine Geräte IP-Adresse einer der beiden CPUs einzurichten:

1. Wählen Sie eine passende PROFINET-Schnittstelle des redundanten Systems S7-1500R/H aus.
2. Deaktivieren Sie die Option "Adresse des H-Systems verwenden".

Verbindungsressourcen des redundanten Systems S7-1500R/H

Maximale Anzahl Verbindungsressourcen des redundanten Systems S7-1500R/H

Das redundante System S7-1500R/H unterstützt eine maximale Anzahl an Verbindungsressourcen.

Die eingesetzte CPU bestimmt die maximale Anzahl der Ressourcen für eine S7-1500R/H Station:

- CPU 1513R: max. 88 Verbindungsressourcen
- CPU 1515R: max. 108 Verbindungsressourcen
- CPU 1517H: max. 160 Verbindungsressourcen

Belegung der Verbindungsressourcen

Kommunikationsverbindungen belegen Verbindungsressourcen im redundanten System S7-1500R/H.

Jede Kommunikationsverbindung zum redundanten System S7-1500R/H belegt Verbindungsressourcen in der S7-1500R/H Station.

Abhängig von der verwendeten IP-Adresse belegt eine Kommunikationsverbindung auch Ressourcen in einer oder in beiden CPUs des redundanten Systems S7-1500R/H.

Die folgende Tabelle zeigt, in welcher CPU eine Kommunikationsverbindung in Abhängigkeit von der verwendeten IP-Adresse Verbindungsressourcen belegt.

Verbindung über...	Verbindungsressourcen der Station	Verbindungsressourcen CPU mit Redundanz-ID 1	Verbindungsressourcen CPU mit Redundanz-ID 2
eine System IP-Adresse	X	X	X
eine Geräte IP-Adresse der CPU mit Redundanz-ID 1	X	X	-
eine Geräte IP-Adresse der CPU mit Redundanz-ID 2	X	-	X

Anzeige der belegten Verbindungsressourcen in STEP 7

Voraussetzung: Online-Verbindung zum redundanten System S7-1500R/H

Sie finden die Verbindungsressourcen im Inspektorfenster in den Eigenschaften der CPUs. STEP 7 zeigt immer die Verbindungsressourcen der selektierten CPU und der S7-1500R/H-Station an.

Verbindungsressourcen

	Ressourcen der Station					Ressourcen des Moduls	
	Reserviert			Dynamisch		CPU 1515R-2 PN (R0/S1)	
Maximale Anzahl der Ressourcen:		10	10	98	98	108	108
	Maximum	Konfigurierte	Verwendet	Konfigurierte	Verwendet	Konfigurier...	Verwendet
PG-Kommunikation:	4	-	2	-	0	-	2
HMI-Kommunikation:	4	0	0	0	0	0	0
S7-Kommunikation:	0	-	0	0	0	0	0
Open User Communication:	0	-	0	0	1	0	1
Web-Kommunikation:	2	-	0	-	0	-	0
Sonstige Kommunikation:	-	-	0	0	0	0	0
Insgesamt verwendete Ressourcen:		0	2	0	1	0	3
Verfügbare Ressourcen:		10	8	98	97	108	105

Verhalten der Verbindungen beim Syncup

Verhalten von Kommunikationsverbindungen über die System IP-Adresse im Systemzustand SYNCUP

- HMI-, PG-, und S7-Verbindungen werden vorübergehend geschlossen. Für eine kurze Zeitdauer während des SYNCUP ist es nicht möglich, Verbindungen zum redundanten System S7-1500R/H aufzubauen.
- Alle vorhanden Verbindungen der Open User Communication werden unterbrochen:
  - Verbindungen, die die CPUs des redundanten Systems als aktiver Verbindungspartner eingerichtet haben, werden nach dem SYNCUP neu eingerichtet.
  - Das redundante System S7-1500R/H richtet nach dem SYNCUP Verbindungsendpunkte für den passiven Verbindungsaufbau neu ein.
- Die Bearbeitung von laufenden Instanzen der Anweisungen TSEND und TRCV wird gestoppt. Der Bausteinparameter STATUS liefert $80C4_H$ (Temporärer Kommunikationsfehler).

Verhalten der Verbindungen bei Primary-Backup-Umschaltung

Verhalten von Kommunikationsverbindungen über die System IP-Adresse während einer Primary-Backup-Umschaltung

- Laufende Instanzen der Anweisungen TSEND und TRCV werden gestoppt und liefern den Status $80C4_H$ (Temporärer Kommunikationsfehler).
- Verbindungen, die das redundante System S7-1500R/H erfolgreich aktiv aufgebaut hatte, baut die neue Primary-CPU erneut auf.
- Die neue Primary-CPU richtet Verbindungsendpunkte für den passiven Verbindungsaufbau neu ein.

Siehe auch

System IP-Adressen (Seite 1119)

Diagnose des redundanten Systems S7-1500R/H

Überblick über den aktuellen Zustand des R/H-Systems gewinnen

Einführung

In der Projektnavigation verschaffen Sie sich einen Überblick über den aktuellen Zustand des R/H-Systems.

In den folgenden Abschnitten wird erklärt, wie Sie aus den dort angezeigten Symbolen diese Informationen ermitteln:

- den aktuellen Systemzustand des R/H-Systems
- die aktuellen Betriebszustände der CPUs des R/H-Systems
- die aktuellen Rollen der CPUs des R/H-Systems
- zu welcher der beiden CPUs aktuell eine Online-Verbindung besteht

Informationen zu den Systemzuständen des R/H-Systems sowie zu den Betriebszuständen und Rollen der CPUs eines R/H-Systems finden Sie im Systemhandbuch "S7-1500 Redundantes System S7-1500R/H"

Den aktuellen Systemzustand des R/H-Systems ermitteln

In welchem Systemzustand sich das R/H-System aktuell befindet, lesen Sie aus den beiden quadratischen Overlay-Icons ab, die sich auf der rechten Seite des R/H-System-Ordnersymbols befinden.

Die folgende Tabelle enthält die möglichen Ordnersymbole und ihre Bedeutung.

Ordnersymbol des R/H-Systems	Systemzustand des R/H-Systems
	STOP
	RUN-Solo
	RUN-Redundant

Hinweis

Ordnersymbol und CPU-Rollen

Aus dem Ordnersymbol für die Systemzustände RUN-Solo und RUN-Redundant können Sie die Rollen der einzelnen CPUs nicht ermitteln.

Die aktuellen Betriebszustände der CPUs des R/H-Systems ermitteln

Welchen Betriebszustand eine CPU des R/H-Systems aktuell einnimmt, lesen Sie aus dem Overlay-Icon ab, das sich rechts oben im Ordnersymbol dieser CPU befindet.

Die folgende Tabelle enthält am Beispiel der Primary-CPU die möglichen Ordnersymbole und ihre Bedeutung.

Ordnersymbol der Primary-CPU	Betriebszustand der Primary-CPU
	ANLAUF
	STOP
	RUN

Hinweis

Abhängigkeit der Betriebszustandsanzeige einer CPU von der bestehenden Online-Verbindung

In den Systemzuständen STOP und RUN-Solo können Sie den aktuellen Betriebszustand nur für diejenige CPU ermitteln, zu der aktuell eine Online-Verbindung besteht.

Die CPU, zu der aktuell keine Online-Verbindung besteht, hat in ihrem CPU-Ordnersymbol rechts oben kein Overlay-Icon.

Dass eine CPU aktuell keine Online-Verbindung zum PG/PC hat, wird durch das folgende Symbol in der vorletzten Spalte der CPU-Zeile dargestellt:

	"Online über Partner": Die andere CPU des R/H-Systems ist online mit dem PG/PC verbunden.

Die aktuellen Rollen der CPUs des R/H-Systems ermitteln

Welche Rolle im R/H-System eine CPU aktuell einnimmt, lesen Sie aus dem Overlay-Icon ab, das sich rechts unten im Ordnersymbol der CPU befindet. Es besteht entweder aus dem Buchstaben "P" oder aus dem Buchstaben "B":

- "P" bedeutet, dass die CPU im R/H-System Primary-CPU ist.
- "B" bedeutet, dass die CPU im R/H-System Backup-CPU ist.

Ermitteln, zu welcher CPU aktuell eine Online-Verbindung besteht

- In den Systemzuständen STOP und RUN-Solo wird bei derjenigen CPU, zu der aktuell keine Online-Verbindung besteht, in der vorletzten Spalte der CPU-Zeile das folgende Symbol dargestellt:

	"Online über Partner": Die andere CPU des R/H-Systems ist online mit dem PG/PC verbunden.

  Damit wissen Sie, dass zu der anderen CPU eine Online-Verbindung besteht.
- Im Systemzustand RUN-Redundant wird das Symbol "Online über Partner" nicht angezeigt. Es ist überflüssig, weil in RUN-Redundant jede CPU auch alle Diagnose-Informationen der anderen CPU kennt.

Erweiterungen der Online- und Diagnosesicht für das R/H-System

Überblick

Es gibt drei Online- und Diagnosesichten:

- eine für das S7-1500R/H-System
- je eine für die beiden CPUs des S7-1500R/H-Systems

Online- und Diagnosesicht des S7-1500R/H-Systems

Diese Online- und Diagnosesicht wurde für das R/H-System neu eingeführt.

Sie besteht aus den folgenden Elementen:

- Gruppe "Online-Zugänge" mit dem Bereich "Online-Zugänge"
- Gruppe "Diagnose" mit dem Bereich "Diagnose"

In "Online-Zugänge" gibt es die folgenden Erweiterungen gegenüber der Online- und Diagnosesicht einer Standard-CPU:

- Unter "Status" wird im Falle einer bestehenden Online-Verbindung zwischen dem PG/PC und einer der beiden CPUs zusätzlich angezeigt, welche Rolle die online verbundene CPU im R/H-System hat:
  - Anzeige "Online (über Primary-CPU", falls die online verbundene CPU Primary-CPU ist
  - Anzeige "Online (über Backup-CPU)", falls die online verbundene CPU Backup-CPU ist
- Unter "Information" erhalten Sie allgemeine Informationen über das R/H-System und die Information, zu welcher CPU beim Online-Gehen eine Online-Verbindung aufgebaut wird.

In "Diagnose" erhalten Sie einen Überblick über den aktuellen Zustand des R/H-Systems. Angezeigt werden:

- der Systemzustand des R/H-Systems
- der Pairing-Zustand der beiden CPUs
- die Betriebszustände der Primary- und der Backup-CPU
- die Namen der Primary- und der Backup-CPU (voreingestellt oder von Ihnen vergeben). Damit wissen Sie, welche CPU die Primary- und welche die Backup-CPU ist.

Online- und Diagnosesichten der beiden CPUs des S7-1500R/H-Systems

Zunächst wird auf die Abhängigkeit der Anzeige vom Systemzustand eingegangen:

- Im Systemzustand RUN-Redundant werden sowohl in der Online- und Diagnosesicht der Primary-CPU als auch in der Online- und Diagnosesicht der Backup-CPU Diagnosedaten angezeigt. Dies gilt unabhängig davon, zu welcher CPU die Online-Verbindung besteht.
- In allen anderen Systemzuständen werden nur in der Online- und Diagnosesicht der online verbundenen CPU Diagnosedaten angezeigt.

Die Online- und Diagnosesicht einer CPU des R/H-Systems entspricht fast vollständig der Darstellung für Standard-CPUs. Der einzige Unterschied befindet sich in der Gruppe "Online-Zugänge": Dort wird unter "Status" im Falle einer bestehenden Online-Verbindung zusätzlich angezeigt, welche Rolle die CPU im R/H-System hat: Primary-CPU oder Backup-CPU.

Abhängigkeiten der Diagnoseanzeigen in der Projektnavigation und in der Online- und Diagnosesicht

Einleitung

Welche Diagnosen in der Projektnavigation und in der Online- und Diagnosesicht angezeigt werden, hängt von den folgenden Faktoren ab:

- Welche Rolle hat die CPU, mit der das PG/PC online verbunden ist: Primary-CPU (in der folgenden Tabelle mit "P" abgekürzt) oder Backup-CPU (in der folgenden Tabelle mit "B" abgekürzt)?
- Welchen Systemzustand hat das R/H-System?

Detaillierte Beschreibung der Abhängigkeiten

Wenn eine Online-Verbindung des PG/PC mit der Primary-CPU besteht, gilt:

Systemzustand des R/H-Systems (und Betriebszustände der CPUs)	Mögliche Diagnosen/Diagnoseanzeigen
STOP (P=STOP, B=STOP)	• Diagnose der Primary-CPU ist möglich. • Diagnose der Backup-CPU ist nicht möglich. • Diagnose der Dezentralen Peripherie ist nicht möglich.
RUN-Solo (P=RUN, B=STOP)	• Diagnose der Primary-CPU ist möglich. • Diagnose der Backup-CPU ist nicht möglich. • Diagnose der Dezentralen Peripherie ist möglich.
RUN-Redundant (P=RUN-Redundant, B=RUN-Redundant)	• Diagnose der Primary-CPU ist möglich. • Diagnose der Backup-CPU ist möglich. • Diagnose der Dezentralen Peripherie ist möglich.

Wenn eine Online-Verbindung des PG/PC mit der Backup-CPU besteht, gilt:

Systemzustand des R/H-Systems (und Betriebszustände der CPUs)	Mögliche Diagnosen/Diagnoseanzeigen
STOP (P=STOP, B=STOP)	• Diagnose der Primary-CPU ist nicht möglich. • Diagnose der Backup-CPU ist möglich. • Diagnose der Dezentralen Peripherie ist nicht möglich.
RUN-Solo (P=RUN, B=STOP)	• Diagnose der Primary-CPU ist nicht möglich. • Diagnose der Backup-CPU ist möglich. • Diagnose der Dezentralen Peripherie ist nicht möglich.
RUN-Redundant (P=RUN-Redundant, B=RUN-Redundant)	• Diagnose der Primary-CPU ist möglich. • Diagnose der Backup-CPU ist möglich. • Diagnose der Dezentralen Peripherie ist möglich.

Zusammenfassung

Die oben detailliert beschriebenen Abhängigkeiten lassen sich in den folgenden Regeln zusammenfassen:

- In den Systemzuständen STOP und RUN-Solo kann nur die CPU diagnostiziert werden, zu der eine Online-Verbindung besteht.
- Im Systemzustand RUN-Redundant können beide CPUs diagnostiziert werden, unabhängig davon, zu welcher CPU eine Online-Verbindung besteht.
- Im Systemzustand STOP kann die Dezentrale Peripherie nicht diagnostiziert werden.
- Im Systemzustand RUN-Solo kann die Dezentrale Peripherie nur dann diagnostiziert werden, wenn eine Online-Verbindung zur Primary-CPU besteht.
- Im Systemzustand RUN-Redundant kann die Dezentrale Peripherie stets diagnostiziert werden, unabhängig davon, zu welcher CPU eine Online-Verbindung besteht.

Erweiterungen der Task Card "Online Tools" für das R/H-System

Überblick

Die Task Card "Online-Tools" zeigt Informationen des Systems S7-1500R/H an, wenn zu einer der beiden CPUs eine Online-Verbindung besteht.

Sie besteht aus den folgenden Paletten:

- R/H-System-Bedienpanel
- je ein CPU-Bedienpanel für die beiden CPUs
- Zykluszeit
- Speicher

R/H-System-Bedienpanel

Die Palette "R/H-System-Bedienpanel" gibt Ihnen einen Überblick über den aktuellen Zustand des R/H-Systems. Die folgenden Informationen werden angezeigt:

- der Name des R/H-Systems
- die aktuellen Zustände der LEDs der Primary-CPU
- der Systemzustand des R/H-Systems
- die Namen und die Betriebszustände der Primary- und der Backup-CPU

Wenn Sie auf die Schaltfläche "STOP R/H-System" klicken, ist die Wirkung vom aktuellen Systemzustand abhängig:

- In den Systemzuständen RUN-Redundant und RUN-Solo hat das Klicken zur Folge, dass der Systemzustand STOP eingenommen wird.
- In den anderen Systemzuständen ist diese Schaltfläche gegraut.

Die Schaltfläche "RUN R/H-System" ist derzeit gegraut und daher ohne Funktion.

CPU-Bedienpanel für die beiden CPUs

Für jede der beiden CPUs des R/H-Systems gibt es ein eigenes CPU-Bedienpanel.

Zuerst wird das Bedienpanel der CPU mit Redundanz-ID 1 angezeigt (entspricht der oberen CPU in der Projektnavigation), dann das der CPU mit Redundanz-ID 2 (entspricht der unteren CPU in der Projektnavigation).

Im Bedienpanel jeder CPU wird das Folgende angezeigt:

- der Name der CPU und ihre Rolle im R/H-System (Primary- oder Backup-CPU)
- die aktuellen Zustände der LEDs der CPU
- die Stellung des Betriebsartenschalters der CPU

Über die Schaltflächen "RUN" und "STOP" wechseln Sie den Betriebszustand der CPU.

Hinweis

Umschaltung des Betriebszustands

Eine Umschaltung des Betriebszustands der CPU über die Schaltflächen "RUN" und "STOP" hat einen Wechsel des Systemzustands zur Folge.

Über die Schaltfläche "MRES" führen Sie ein Urlöschen der CPU aus.

Gegenüber dem Bedienpanel einer Standard-CPU gibt es die folgende Erweiterung: Hinter dem CPU-Namen wird zusätzlich die Rolle der CPU im R/H-System angezeigt (Primary- oder Backup-CPU).

Zykluszeit

Die Palette "Zykluszeit" entspricht der gleichnamigen Palette bei Standard-CPUs. Sie zeigt die Werte der CPU an, zu der aktuell eine Online-Verbindung besteht.

Die Erweiterung gegenüber einer Standard-CPU besteht lediglich darin, dass ganz oben in der Palette hinter "Zykluszeit" die Rolle der online verbundenen CPU in eckigen Klammern angezeigt wird: "Primary" oder "Backup"

Speicher

Die Palette "Speicher" entspricht der gleichnamigen Palette bei Standard-CPUs. Sie zeigt die Werte der CPU an, zu der aktuell eine Online-Verbindung besteht.

Die Erweiterung gegenüber einer Standard-CPU besteht lediglich darin, dass ganz oben in der Palette hinter "Speicher" die Rolle der online verbundenen CPU in eckigen Klammern angezeigt wird: "Primary" oder "Backup"

Verbindungsdiagnose im R/H-System

Anzeige der Verbindungsressourcen einer Verbindung, welche die System-IP-Adresse des R/H-Systems benutzt

Verbindungen zu einem R/H-System, welche dessen System-IP-Adresse benutzen, verhalten sich hinsichtlich der belegten Verbindungsressourcen anders als Standardverbindungen zu genau einer CPU des R/H-Systems: Verbindungen über die System-IP-Adresse belegen in jeder der beiden CPUs und in der R/H-Station Ressourcen.

Entsprechend verhält sich die Anzeige der Verbindungsressourcen ("Diagnose" > "Verbindungsinformation" des Inspektorfensters, Gruppe "Verbindungsressourcen").

Weitere Informationen finden Sie im Funktionshandbuch "Kommunikation".

Vergleichsfunktionen für ein R/H-System

Für ein R/H-System können Sie einen Offline/Offline- sowie einen Offline/Online-Vergleich durchführen.

Im Folgenden sind die Besonderheiten beim Vergleichen im Zusammenhang mit R/H-Systemen beschrieben.

Offline/Offline-Vergleich (Software)

Bei einem R/H-System können Sie für den Vergleich (Software) mit einer anderen CPU nur eine R/H-CPU nutzen.

Die Durchführung des Offline/Offline-Vergleichs unterscheidet sich grundsätzlich nicht vom Offline/Offline-Vergleich mit Standard-CPUs.

Da SW-Objekte nur für die R/H-CPU mit der Redundanz-ID "1" angezeigt werden (obere CPU in der Projektnavigation), können Sie auch nur diese CPU für einen Vergleich heranziehen.

Wenn Sie das Symbol für das R/H-System (beide CPUs) aus der Projektnavigation für den SW-Vergleich in den Vergleichseditor ziehen, dann wählt STEP 7 automatisch die CPU mit der Redundanz-ID "1" aus.

Ein Offline/Offline-Vergleich (Software) der beiden R/H-CPUs (Vergleich Primary-CPU mit Backup-CPU) ist **nicht** möglich.

Offline/Offline-Vergleich (Hardware)

Sie können zwischen zwei S7-CPUs einen Hardware-Vergleich durchführen.

Beispiele:

- Vergleich zweier kompletter R/H-Systeme:
  Platzieren Sie per Drag & Drop die jeweiligen R/H-Stationen im Vergleichseditor. Die Hardware-Struktur beider Stationen inklusive der lokalen Module und Dezentraler Peripherie wird verglichen.
- Vergleich der beiden CPUs desselben R/H-Systems

- Vergleich von jeweils erster oder zweiter CPU zweier R/H-Systeme
- Eine beliebige CPU eines R/H-Systems mit einer beliebigen anderen S7-1500 CPU

Nicht vergleichen können Sie eine R/H-CPU z. B. mit PC-Systemen (andere Aufbautechnik).

Offline/Online-Vergleich

Die Durchführung des Offline/Online-Vergleichs unterscheidet sich grundsätzlich nicht vom Offline/Online-Vergleich mit Standard-CPUs:

Sie haben bei als Ausgangspunkt für diesen Vergleich die Möglichkeit, entweder die R/H-Station oder eine bzw. beide CPUs der R/H-Station in der Projektnavigation zu markieren und wählen anschließend wie gewohnt den Kontext-Menübefehl "Vergleichen > Offline/Online". STEP 7 vergleicht mit der Online-CPU, die die Rolle "Primary" hat.

Für den Offline/Online-Vergleich mit der Backup-CPU existiert kein eigener Menübefehl. Wenn Sie eine Offline-CPU/das R/H-System mit der Backup-CPU als Zielgerät vergleichen wollen, dann müssen Sie zunächst mit dem Kontext-Menübefehl "Online-Verbindung zur Backup-CPU" eine Online-Verbindung zur Backup-CPU aufbauen und dann den Offline/Online-Vergleich starten.

R/H-System ist offline

Wenn beide R/H-CPUs des Systems offline sind, startet der Vergleich des R/H-Systems mit der Primary-CPU.

STEP 7 baut daher eine Verbindung zur Primary-CPU auf. Wenn noch kein Online-Pfad eingestellt wurde, öffnet sich der Online-verbinden-Dialog und schlägt als Zielgerät die Primary-CPU vor. Nach Quittieren des Dialogs (Schaltfläche "Verbinden") startet der Vergleichseditor und zeigt die Unterschiede zwischen Offline- und Online-Bausteinen der Primary-CPU.

R/H-System ist online

Wenn eine der beiden R/H-CPUs online ist, zieht STEP 7 das entsprechende Zielgerät, zu dem die Online-Verbindung besteht, für den Offline/Online-Vergleich heran.

Wo finden Sie weitere Informationen?

Weitere Informationen zur Diagnose des redundanten Systems S7-1500R/H finden Sie im Funktionshandbuch "Diagnose".

1.4.1.4 Konfigurieren von zentralen Baugruppen

In den folgenden Kapiteln erfahren Sie, wie Sie zentrale Baugruppen konfigurieren. Dazu erhalten Sie auch Informationen über das Adressieren von Baugruppen.

Übersicht zum Konfigurieren zentraler Baugruppen

Schritt	Beschreibung
1	Wählen Sie eine zentrale Baugruppe aus (Seite 1151).
2	Legen Sie die Eigenschaften der Baugruppen fest (Seite 1154).
3	Legen Sie die Ein- und Ausgangsadressen fest (Seite 1161).

Siehe auch

Adressierung von Baugruppen (Seite 1161)

Adressierung von Baugruppen (Seite 1160)

Regeln für Baugruppen (Seite 1139)

Regeln für Baugruppen (Seite 1140)

1.4.1.5 Regeln für Baugruppen

Einführung

Für S7-300 Baugruppen gelten spezifische Steckplatzregeln.

Wenn Sie in der Gerätesicht eine Baugruppe im Hardware-Katalog selektieren, werden im Baugruppenträger alle möglichen Steckplätze für die selektierte Baugruppe markiert. Sie können Baugruppen nur auf markierte Steckplätze ziehen.

Wenn Sie eine Baugruppe einfügen, verschieben oder tauschen, werden ebenfalls die Steckplatzregeln angewendet.

Steckplatzregeln für S7-300

Für S7-300 gelten zusätzlich zu den allgemeinen Steckplatzregeln die folgenden Regeln:

- Sie können bis zu 4 Baugruppenträger verwenden.
  Baugruppenträger 0:
  - Steckplatz 1: nur Stromversorgung (z. B. 6ES7 307-...) oder leer
  - Steckplatz 2: nur Zentralbaugruppe (z. B. 6ES7 314-...)
  - Steckplatz 3: Anschaltungsbaugruppe (z. B. 6ES7 360-.../365-...) oder leer
  - Steckplatz 4 bis 11: Signal- oder Funktionsbaugruppen, Kommunikationsprozessoren oder leer

  Baugruppenträger 1 bis 3:
  - Steckplatz 1: nur Stromversorgungsbaugruppe (z. B. 6ES7 307-...) oder leer
  - Steckplatz 2: leer
  - Steckplatz 3: Anschaltungsbaugruppe (z. B. 6ES7 361-.../365-...)
  - Steckplatz 4 bis 11: Signal- oder Funktionsbaugruppen, Kommunikationsprozessoren (abhängig von der verwendeten Anschaltungsbaugruppe) oder leer
- In Baugruppenträger 3 können Sie bei Verwendung einer CPU mit integrierter Peripherie den Steckplatz 11 nicht verwenden.
- Baugruppen müssen Sie lückenlos aneinanderreihen.

Hinweis

Bei S7-300 ist der Steckplatz 3 für die Anschaltungsbaugruppe reserviert. Bei Verwendung eines einzelnen Baugruppenträgers wird dieser Steckplatz nicht belegt, da keine Anschaltungsbaugruppe gesteckt werden muss. Im realen Aufbau haben Sie keine Lücke, da der Rückwandbus sonst unterbrochen wäre!

Eine ET 200M mit aktivem Rückwandbus darf Lücken in der Belegung aufweisen.

Besondere Regeln zur Platzhalterbaugruppe DM 370 Dummy

Die Platzhalterbaugruppe DM 370 Dummy (6ES7 370-0AA01-0AA0) können Sie anstelle einer später einzusetzenden Baugruppe stecken.

Die DM 370 Dummy reserviert einen Adressraum von 1 Byte für eine Baugruppe im modularen DP-Slave (z. B. für eine digitale E/A-Baugruppe).

1.4.1.6 Regeln für Baugruppen

Einführung

Für S7-400 Baugruppen gelten spezifische Steckplatzregeln. Die Regeln für die Anordnung von Baugruppen auf einen Baugruppenträger der S7-400 hängen ab von der Art des verwendeten Baugruppenträgers.

Wenn Sie in der Gerätesicht eine Baugruppe im Hardware-Katalog selektieren, werden im Baugruppenträger alle möglichen Steckplätze für die selektierte Baugruppe markiert. Sie können Baugruppen nur auf markierte Steckplätze ziehen.

Wenn Sie eine Baugruppe einfügen, verschieben oder tauschen, werden ebenfalls die Steckplatzregeln angewendet.

Steckplatzregeln der S7-400 für zentralen Baugruppenträger

Für S7-400 gelten zusätzlich zu den allgemeinen Steckplatzregeln die folgenden Regeln für Baugruppen auf einem zentralen Baugruppenträger:

- Stromversorgungsbaugruppen nur auf Steckplatz 1 stecken (Ausnahme: redundierbare Stromversorgungsbaugruppen)
- max. 6 Anschaltungsbaugruppen (Sende-IMs) stecken; davon max. 2 mit Stromübertragung
- max. 21 Erweiterungsbaugruppenträger über Anschaltungsbaugruppen an den zentralen Baugruppenträger koppeln
- max. 1 Erweiterungsbaugruppenträger **mit Stromübertragung** an eine Schnittstelle der Sende-IM koppeln (IM 460-1 mit IM 461-1)
- max. 4 Erweiterungsbaugruppenträger **ohne Stromübertragung** (IM 460-0 mit IM 461-0 bzw. IM 460-3 mit 461-3)

Steckplatzregeln der S7-400 für Erweiterungsbaugruppenträger

Für S7-400 gelten zusätzlich zu den allgemeinen Steckplatzregeln die folgenden Regeln für Baugruppen auf Erweiterungsbaugruppenträgern:

- Stromversorgungsbaugruppen nur auf Steckplatz 1 stecken (Ausnahme: redundierbare Stromversorgungsbaugruppen)
- die Anschaltungsbaugruppe (Empfangs-IM) nur auf den äußeren rechten Steckplatz stecken (Steckplatz 9 bzw. Steckplatz 18)
- K-Bus-Baugruppen nur in Erweiterungsbaugruppenträger stecken, deren Nr. nicht größer als 6 ist (sonst können sie nicht angesprochen werden)

Besondere Regeln für redundierbare Stromversorgungsbaugruppen

Redundierbare Stromversorgungsbaugruppen können in einem Baugruppenträger zweimal gesteckt werden. Ob es sich bei einer Stromversorgungsbaugruppe um eine redundierbare Stromversorgungsbaugruppe handelt, können Sie in der Palette "Information" der Task Card "Hardware-Katalog" lesen.

Es sind folgende Regeln zu beachten:

- Das Stecken redundierbarer Stromversorgungsbaugruppen ist nur in den dafür vorgesehenen Baugruppenträgern möglich (erkennbar am Infotext in der Task Card "Hardware Katalog").
- Die redundierbaren Stromversorgungsbaugruppen müssen auf Steckplatz 1 und dem sich anschließenden Steckplatz gesteckt werden (keine Lücken zulässig!).
- Redundierbare und nicht redundierbare Stromversorgungsbaugruppen sind nicht im selben Baugruppenträger steckbar (d. h. kein "Mischbetrieb" möglich).

1.4.1.7 Regeln für Baugruppen

Einsatz von S7-1500 Baugruppen

Einleitung

Die CPUs des Automatisierungssystems S7-1500 sind modulare Controller für Systemlösungen im mittleren und oberen Leistungsbereich für die Automatisierung von diskreten und kontinuierlichen Prozessen.

Einsatzgebiet

Das Automatisierungssystem S7-1500 ist konzipiert für Systemlösungen in der Prozess- und Fertigungsautomatisierung. Das Automatisierungssystem S7-1500 stellt als universelles Automatisierungssystem die optimale Lösung für Anwendungen in zentralen und dezentralen Aufbauten dar.

Durch den skalierbaren Aufbau können Sie den Aufbau Ihrer Steuerung exakt auf den jeweiligen Bedarf vor Ort ausrichten.

Das Automatisierungssystem S7-1500 ist für die Schutzart IP20 zugelassen und für den Einbau in einen Schaltschrank vorgesehen.

Aufbau

Das Automatisierungssystem S7-1500 wird auf eine Profilschiene montiert und kann aus bis zu 32 Modulen bestehen. Die Module werden über Busverbinder miteinander verbunden. Die Stromversorgung erfolgt über den Rückwandbus, wobei die CPU selbst auch einige Module mit Strom versorgen kann. Wenn die Module mehr Strom verbrauchen als die CPU liefern kann, können Sie bis zu drei zusätzlich Stromversorgungsmodule stecken.

Kompakte Bauform

Das Automatisierungssystem S7-1500 ist zudem in einer kompakten Bauform verfügbar. Dabei ist die CPU bereits mit integrierten Ein- und Ausgängen ausgestattet.

Die Kompakt-CPUs lassen sich mit allen Modulen des Automatisierungssystems S7-1500 erweitern. Der Gesamtausbau beträgt insgesamt 32 Module, mit der CPU und den in der CPU integrierten Modulen (siehe Gerätehandbücher CPU 1511C-1 PN (http://

support.automation.siemens.com/WW/view/de/109478675)und CPU 1512C-1 PN (http://support.automation.siemens.com/WW/view/de/109478676)).

Siehe auch

Steckplatzregeln für S7-1500 Baugruppen (Seite 1143)

Ausbau einer S7-1500 Station (Seite 1144)

Leistungsbilanzierung (Seite 1146)

Leistungsbilanz prüfen (Seite 1147)

Steckplatzregeln für S7-1500 Baugruppen

Für S7-1500 Baugruppen gelten spezifische Steckplatzregeln. Wenn Sie in der Gerätesicht eine Baugruppe im Hardware-Katalog selektieren, werden im Baugruppenträger alle möglichen Steckplätze für die selektierte Baugruppe markiert. Sie können Baugruppen nur auf markierte Steckplätze ziehen. Die Steckplatzregeln gelten immer, wenn Sie eine Baugruppe einfügen, verschieben oder tauschen.

Steckplatzregeln für S7-1500

Für S7-1500 gelten die folgenden Steckplatzregeln:

- Sie können zusätzlich zur CPU bis zu 31 Baugruppen auf einen Baugruppenträger stecken.
- Sie können insgesamt maximal drei Stromversorgungsmodule stecken. Diese versorgen zusammen mit der CPU sämtliche nachfolgenden Geräte auf dem Baugruppenträger mit Strom. Wie viele Stromversorgungsmodule Sie mindestens einsetzen müssen, hängt von der Leistungsaufnahme der einzelnen Module ab.
  - Links der CPU (Steckplatz 0) lässt sich ein Laststromversorgungsmodul PM oder ein Systemstromversorgungmodul PS stecken.
  - Rechts der CPU (Steckplätze 2 bis 31) lassen sich bis zu zwei weitere Standard-Systemstromversorgungen (PS) stecken.

Hinweis

Powersegmente

Ein Stromversorgungsmodul und die von ihr versorgten Module bilden ein Powersegment. Ein Stromversorgungsmodul versorgt immer die Module mit Strom, die rechts von ihr gesteckt sind. Bei dem nächsten Stromversorgungsmodul endet das Powersegment und es beginnt ein neues Powersegment für das nächste Stromversorgungsmodul und die rechts von ihr gesteckten Module. Die Versorgung der von einem Stromversorgungsmodul links liegenden Module übernimmt entweder das vorhergehende Stromversorgungsmodul oder die CPU.

Siehe auch

Leistungsbilanzierung (Seite 1146)

Ausbau einer S7-1500 Station (Seite 1144)

Erweiterten remanenten Datenbereich nutzen mit PS 60W 24/48/60VDC HF (Seite 1148)

Ausbau einer S7-1500 Station

Zur Erweiterung der einzelnen S7-1500 CPUs stehen Ihnen verschiedene Peripheriemodule zur Verfügung. Neben analogen und digitalen Ein- und Ausgabemodulen können Sie Technologiemodule, Kommunikationsmodule und Spezialmodule verwenden. Trotz der hohen Leistungsfähigkeit der S7-1500 CPUs sind jedoch in Abhängigkeit von der verwendeten CPU Grenzen bei der Anzahl der Peripheriemodule und den projektierten IO-Systemen gesetzt, wenn der Hardware-Aufbau sehr groß ist. Bei der Projektierung sollten Sie daher die im technischen Datenblatt der jeweiligen CPU beschriebenen Grenzen beachten.

Maximalausbau einer Station

Beim Übersetzen der Hardware-Konfiguration wird geprüft, ob Maximalgrenzen beim Ausbau einer Station überschritten wurden. Es ist jedoch empfehlenswert schon beim Konfigurieren einer S7-1500 Station auf eventuelle Mengenbeschränkungen zu achten. Prüfen Sie insbesondere die folgenden Kriterien anhand der technischen Daten Ihrer Geräte beim Erstellen der Hardware-Konfiguration:

- Maximale Anzahl Kommunikationsmodule für PROFINET, PROFIBUS und Industrial ETHERNET sowie Kommunikationsprozessoren
  Bei einer S7-1513 (6ES7 513-1AL00-0AB0) sind dies beispielsweise sechs Stück.
- Maximale Anzahl dezentraler PROFINET IO-Systeme und dezentraler PROFIBUS DP Systeme.
  IO-Systeme, die über ein I-Device verbunden sind, müssen bei der Maximalanzahl nicht mit berücksichtig werden. In diesem Fall gelten die Regeln für den Maximalausbau des I-Device.

- Maximale Anzahl dezentraler Peripheriegeräte (IO-Devices und DP-Slaves)
  Gezählt werden alle dezentralen Peripheriegeräte, die der CPU zugeordnet sind, unabhängig davon, an welcher Schnittstelle sie angeschlossen sind (integrierte Schnittstelle oder Schnittstelle der CPs/CMs). Die CPU wird nur dann zu den Peripheriegeräten dazu gezählt, wenn sie als I-Device projektiert ist.
  Wenn am PROFINET IO-System I-Devices angeschlossen sind, dann werden die IO-Devices am unterlagerten PROFINET IO-System eines I-Devices nicht gezählt.
- Maximale Anzahl von Modulen/Submodulen
  Die maximale Anzahl steckbarer Module ist sehr groß und wird bei üblichen Hardware-Aufbauten eines S7-1500-Automatisierungssystems nur selten errreicht.
  Wenn in sehr großen Anlagen viele Module mit Submodulen verwendet werden, können die verfügbaren Ressourcen für Module/Submodule der CPU überschritten werden.
  Einfache Peripheriemodule, die keine Submodule haben, benötigen eine Ressource.
  Komplexe Module können mehrere Submodule haben und benötigen für jedes Submodul eine Ressource. Außerdem benötigt ein Modul mit Submodulen noch eine Ressource zusätzlich (Stellvertretung für das gesamt Modul).
  Beispiel:
  Ein Interfacemodul IM 155-5PNHF benötigt fünf Ressourcen nach folgender Zählweise: Zwei Ressourcen für die beiden Ports der PROFINET-Schnittstelle, eine Ressource für das Interface der PROFINET-Schnittstelle, eine Ressource für das Submodul für die Parameter des Interfacemoduls. Wie oben erläutert, kommt noch eine Ressource hinzu, da es sich um ein Modul mit Submodulen handelt.

Beispiel

Die folgenden Bilder zeigen beispielhaft eine Hardware-Konfiguration und verdeutlichen die oben beschriebene Zählweise:

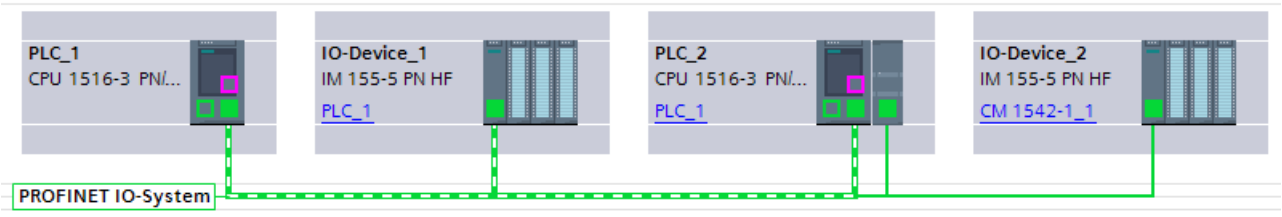

Bild 1-20 In diesem Bild ist PLC_1 als IO-Controller konfiguriert. IO-Device_1 und PLC_2 sind Devices am IO-Systems von PLC_1. IO-Device_2 ist zwar mit demselben Netzwerk verbunden, ist aber nicht Teil des IO-Systems. Somit zählt das IO-System von PLC_1 zwei Devices.

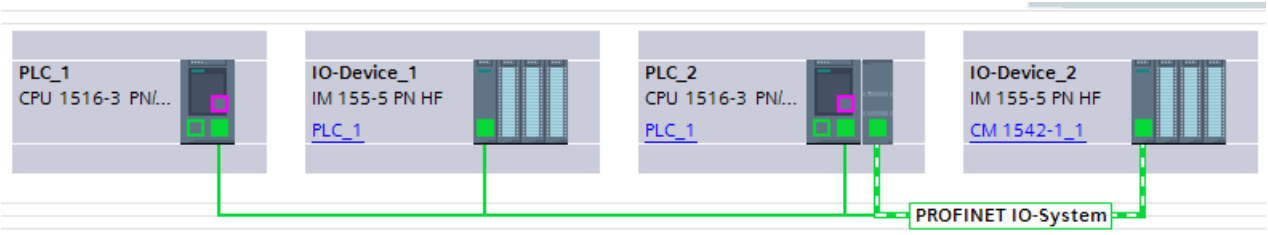

Bild 1-21 Das zweite Bild zeigt, dass PLC_2 zwar ein I-Device im IO-System von PLC_1 ist, aber sie ist gleichzeitig auch IO-Controller. Über das Kommunikationsmodul wird ein unterlagertes IO-System aufgespannt, mit dem IO-Device_2 verbunden ist. Insgesamt werden für PLC_2 zwei Devices gezählt.

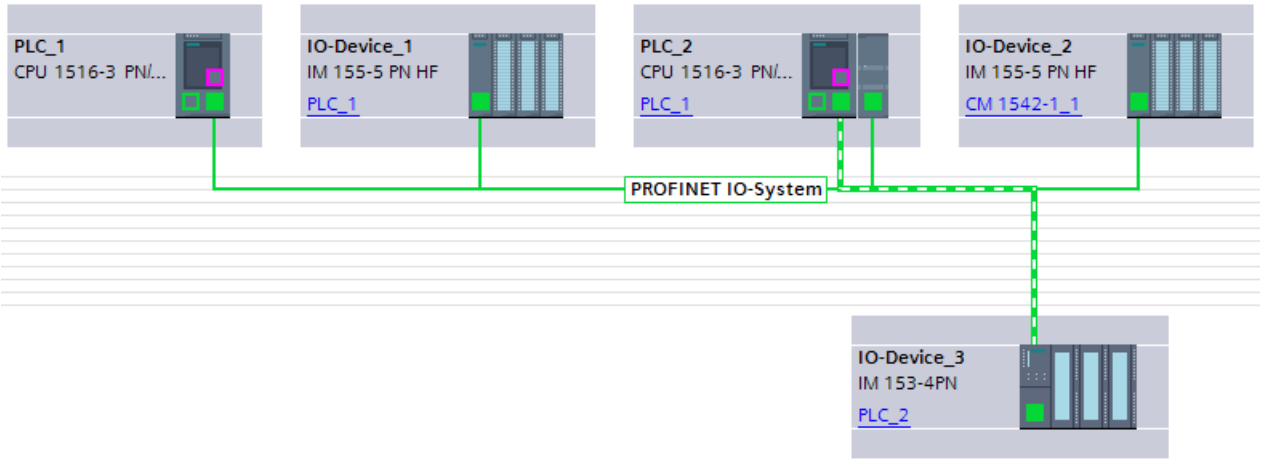

Bild 1-22 Im dritten Bild ist ein weiteres IO-System PLC_2 zugeordnet. PLC_2 verfügt nun insgesamt über zwei IO-Systeme, eines an der internen Schnittstelle und eines am Kommunikationsmodul. Am neuen IO-System ist ein weiteres Device hinzugekommen. Insgesamt werden bei diesem Hardware-Aufbau für PLC_2 drei Devices gezählt - ein I-Device (PLC_2 ist PLC_1 zurgeordnet) und zwei IO-Devices (PLC_2 zugeordnet).

Siehe auch

Steckplatzregeln für S7-1500 Baugruppen (Seite 1143)

Leistungsbilanzierung (Seite 1146)

Leistungsbilanzierung

Einleitung

Die zur Ansteuerung der Elektronik benötigte Leistung entnehmen alle Peripheriemodule aus dem Rückwandbus. Diese Leistung wird von der Zentralbaugruppe und bei Bedarf auch durch zusätzliche Systemstromversorgungen in den Rückwandbus eingespeist.

Prüfung der Leistungsbilanz bei der Projektierung

Bei der Prüfung der Leistungsbilanz werden Sie von STEP 7 unterstützt. In einer Tabelle werden alle gesteckten Peripheriemodule und ihr maximaler Stromverbrauch aufgeführt. Der Stromverbrauch wird mit den gesteckten Systemstromversorgungsmodulen gegengerechnet. So behalten Sie den Überblick, ob genügend Leistung für alle Module zur Verfügung steht.

Die Tabelle finden Sie im Inspektorfenster, Bereich "Systemstromversorgung", wenn Sie die entsprechende CPU selektiert und das Register "Eigenschaften" gewählt haben.

Zusätzlich wird bereits bei der Projektierung überprüft, ob genügend Leistung zur Verfügung steht. Entnehmen die Peripheriemodule mehr Leistung aus dem Rückwandbus als eingespeist wird, werden die entsprechenden Tabellenzeilen rot eingefärbt und Sie erhalten Sie eine entsprechende Warnung nach dem Übersetzen der Hardware-Konfiguration.

Prüfung der Leistungsbilanz durch die CPU

Die Einhaltung der Leistungsbilanz wird im Betrieb von der CPU überwacht:

- Bei jedem NETZ-EIN
- Bei jeder Änderung des Hardware-Aufbaus

Stellt die CPU eine Überlastung des Rückwandbusses fest, schaltet die CPU sämtliche Peripheriemodule ab, die von der CPU oder der Systemstromversorgung versorgt werden und trägt eine Meldung in den Diagnosepuffer ein. Die CPU läuft nicht an, solange die Leistungsbilanz negativ ist.

Ursachen für Überlast

Ursachen für Überlast können sein, dass der Hardware-Aufbau nicht der Projektierung entspricht. Das heißt, dass z. B. die zum Betrieb notwendigen Systemstromversorgungen nicht gesteckt oder eingeschaltet sind oder mehr Module gesteckt sind als projektiert wurden.

Siehe auch

Leistungsbilanz prüfen (Seite 1147)

Steckplatzregeln für S7-1500 Baugruppen (Seite 1143)

Ausbau einer S7-1500 Station (Seite 1144)

Leistungsbilanz prüfen

Sie können bereits bei der Projektierung einer Station ersehen, ob genügend Leistung für alle Peripheriemodule zur Verfügung steht. Dafür müssen Sie zunächst angeben, ob die CPU mit Spannung versorgt ist.

Vorgehen

Um die Stromversorgung der CPU zu aktivieren und die Leistungsbilanz auszuwerten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die S7-1500 CPU, z. B. in der Netzsicht.
2. Öffnen Sie das Register "Eigenschaften" im Inspektorfenster.
   Die Eigenschaften der CPU werden angezeigt.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Systemstromversorgung".
4. Aktivieren Sie die Option "Anschluss an Versorgungsspannung L+", wenn die CPU mit Spannung versorgt ist. Standardmäßig ist die Option aktiviert. Somit wird die Leistungsaufnahme der CPU in der Leistungsbilanz berücksichtigt.
5. Prüfen Sie in der Tabelle "Leistungsbilanzierung", ob die Leistungsbilanz positiv ist. Ist die Leistungsbilanz negativ, werden die unterversorgten Baugruppen rot markiert.

Siehe auch

Leistungsbilanzierung (Seite 1146)

Erweiterten remanenten Datenbereich nutzen mit PS 60W 24/48/60VDC HF

Wenn Sie die Systemstromversorgung PS 60W 24/48/60VDC HF einsetzen, dann können Sie den erweiterten remanenten Datenbereich für eine S7-1500 CPU nutzen.

Verwenden Sie die Systemstromversorgung PS 60W 24/48/60VDC HF für Prozesse, die für ein nahtloses Aufsetzen nach einem Stromausfall eine sehr große Anzahl an remanenten Daten benötigen.

Die Werte für die erweiterte Remanenz sind abhängig von der verwendeten CPU. Die Werte finden Sie in den technischen Daten der CPU-Gerätehandbücher bzw. in der entsprechenden Produktinformation.

Link zur Produktinformation:

https://support.industry.siemens.com/cs/ww/de/view/68052815 (https://support.industry.siemens.com/cs/ww/de/view/68052815)

Voraussetzung

Voraussetzungen für den Einsatz der Systemstromversorgung PS 60W 24/48/60VDC HF zur Nutzung des erweiterten remanenten Datenbereichs:

- CPU S7-1500 ab FW-Stand V2.1
- Kompatibler HW-Funktionsstand der CPU S7-1500
  Bei folgenden CPUs können Sie erst **ab dem HW-Funktionsstand FS03** den erweiterten remanenten Datenbereich nutzen:
  - CPU 1517-3 PN/DP, CPU 1517F-3 PN/DP
  - CPU 1518-4 PN/DP, CPU 1518F-4 PN/DP

  Bei folgender CPU können Sie erst **ab dem HW-Funktionsstand FS02** den erweiterten remanenten Datenbereich nutzen:
  - CPU 1518F-4 PN/DP ODK

  Die genannten CPUs lassen kein Laden des Anwenderprogramms (Software) zu, wenn Sie den erweiterten remanente Datenbereich nutzen.
  Abhilfe: Verwenden Sie eine CPU mit passendem HW-Funktionsstand.
- STEP 7 ab V14 Servicepack 1

Regeln für den Aufbau einer S7-1500 mit PS 60W 24/48/60VDC HF

- Die Systemstromversorgung PS 60W 24/48/60VDC HF darf nur links neben der CPU (Steckplatz 0) gesteckt sein.
  Bei dezentralem Aufbau einer ET 200MP darf diese Systemstromversorgung nur links neben dem Interfacemodul gesteckt sein. Bei dezentraler Verwendung erweitert die Stromversorgung nicht den remanenten Datenbereich der CPU.
  Die zulässigen Steckplätze zeigt STEP 7 beim Platzieren von Modulen blau umrandet an, sodass eine Falschkonfiguration ausgeschlossen ist.
  Das Bild zeigt links neben Steckplatz 0 den Steckplatz 100 für eine Laststromversorgung (PM). Dieser Steckplatz ist nur dann sichtbar, wenn ein CPU ab Firmware-Version 2.1 steckt.

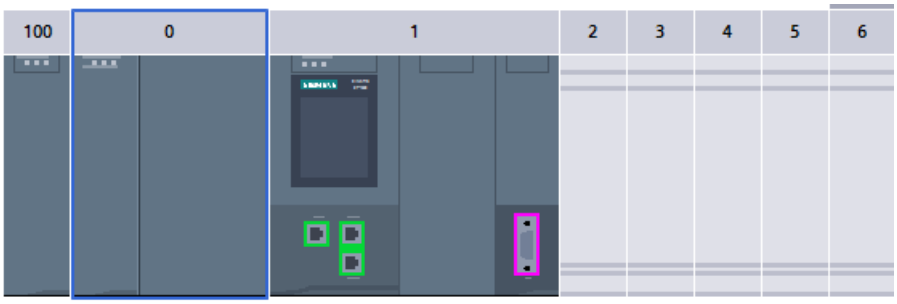

- Rechts neben der CPU bzw. rechts neben einem Interfacemodul (ET 200MP) ist keine Systemstromversorgung PS 60W 24/48/60VDC HF steckbar. Wenn Sie weitere Systemstromversorgungen zur Leistungserhöhung für den Rückwandbus benötigen, dann verwenden Sie eine Standard-Systemstromversorgung ab Steckplatz 2.

- Wenn Sie die PS 60W 24/48/60VDC HF einsetzen, dann wird die 24V-Versorgung der CPU nicht in der Leistungsbilanzierung berücksichtigt. Sie sollten daher die Versorgungsspannung DC 24V nicht an die CPU anzuschließen. Der CPU-Parameter "Systemstromversorgung" muss daher auf die Option "Kein Anschluss an Versorgungsspannung L+" eingestellt sein.

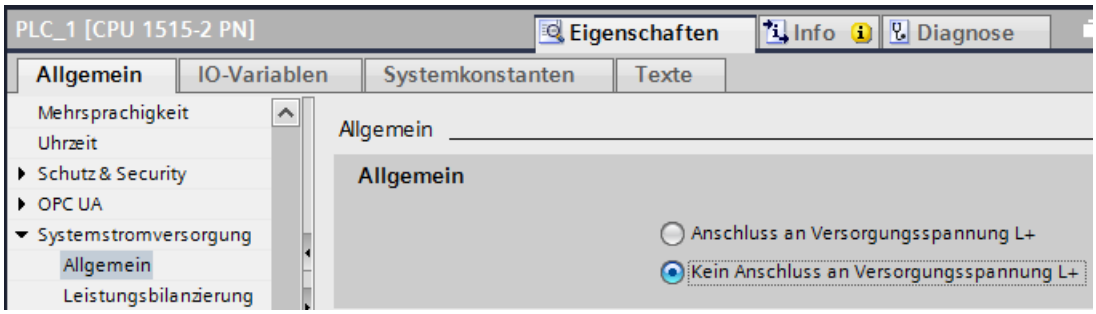

- Der Parameter "Vergleich Sollbaugruppe zu Istbaugruppe" muss auf den Wert "Anlauf der CPU nur bei Kompatibilität" stehen.
  Grund: Die Remanenz des kompletten Arbeitsspeichers (Daten) der CPU ist nur gewährleistet, wenn eine PS 60W 24/48/60VDC HF gesteckt ist.
  Beim Stecken der PS 60W 24/48/60VDC HF stellt STEP 7 automatisch diesen Wert für den Steckplatz ein.

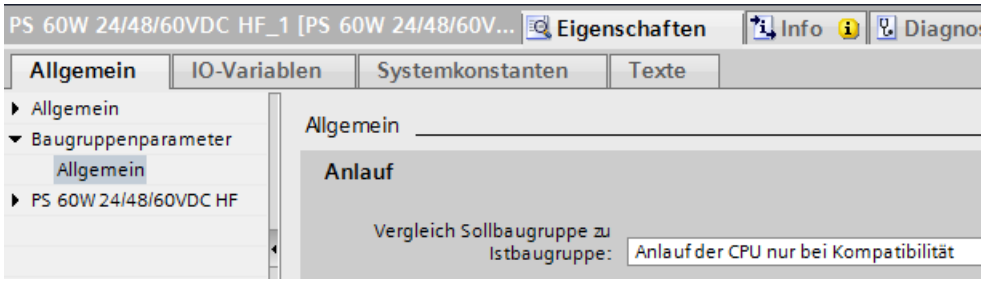

Die oben genannten Regeln prüft STEP 7 beim Übersetzen bzw. beim Laden des Anwenderprogramms in die CPU.

Ersatzteil-Fall

Die Systemstromversorgung PS 60W 24/48/60VDC HF können Sie als Ersatz für eine projektierte PS 60W 24/48/60VDC (Standard-PS) auf Steckplatz 0 einsetzen. In diesem Fall profitieren Sie nicht vom erweiterten remanenten Datenbereich.

Umgekehrt können Sie die Standard-PS **nicht** als Ersatz für eine projektierte PS 60W 24/48/60VDC HF einsetzen.

Fehlende Diagnosen der Systemstromversorgung PS 60W 24/48/60VDC HF

Beim NETZ-AUS steht das Speichern der erweiterten remanenten Daten im Vordergrund. Die CPU ab FW V2.1 gibt in diesem Fall die folgenden Diagnosen der PS 60W 24/48/60VDC HF **nicht** mehr aus:

- Versorgungsspannungsfehler
- Schalterstellung aus

Siehe auch

Steckplatzregeln für S7-1500 Baugruppen (Seite 1143)

1.4.1.8 Zentrale Baugruppen auswählen

Einführung

In der Netzsicht wählen Sie aus dem Hardware-Katalog eine CPU aus und legen diese zusammen mit einem Baugruppenträger an. Auf dieses Gerät ziehen Sie aus dem Hardware-Katalog die gewünschten Baugruppen, die automatisch auf dem Baugruppenträger angeordnet werden.

Auswahl der Komponente im Hardware-Katalog

Im Hardware-Katalog wird jede Hardware-Komponente als Ordner angezeigt. Wenn Sie diesen Ordner öffnen, sehen Sie die unterschiedlichen Versionen der ausgewählten Hardware-Komponente mit ihren jeweiligen Artikelnummern.

Es wird Ihnen hier beispielhaft gezeigt, wie Sie in der Netzsicht eine CPU mit einem Baugruppenträger anlegen.

Voraussetzung

- Der Hardware-Katalog ist geöffnet.
- Sie befinden sich in der Netzsicht.

Vorgehen

Um eine CPU aus dem Hardware-Katalog auszuwählen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog auf den Ordner mit den gewünschten CPUs.
2. Öffnen Sie den Ordner mit dem gesuchten CPU-Typ. Sie sehen alle Artikelnummern des angewählten CPU-Typs.

3. Klicken Sie auf eine CPU-Artikelnummer, um sich in der Palette "Information" über die gewählte CPU zu informieren.

4. Stellen Sie in der Klappliste "Version" die passende Firmware-Version Ihrer CPU ein.
5. Legen Sie die CPU und einen Baugruppenträger an. Sie haben dazu folgende Möglichkeiten:
   – Ziehen Sie mit Drag & Drop die CPU aus dem Hardware-Katalog in die Netzsicht.
   – Doppelklicken Sie auf den CPU-Eintrag im Hardware-Katalog.

1.4.1.9 Fehlersichere Baugruppen auswählen

Einführung

Fehlersichere Automatisierungssysteme (F-Systeme) dienen der Steuerung von Prozessen mit unmittelbar durch Abschaltung erreichbarem sicherem Zustand. F-Systeme steuern Prozesse, bei denen eine unmittelbare Abschaltung keine Gefahr für Mensch oder Umwelt nach sich zieht.

Auswahl der F-Komponente im Hardware-Katalog

Im Hardware-Katalog wird jede Hardware-Komponente als Ordner angezeigt. Fehlersichere Hardware-Komponenten werden im Gegensatz zu den blauen nicht-fehlersicheren Hardware-Komponenten im Hardware-Katalog mit gelben Symbolen dargestellt.

Im folgenden Bild sehen Sie Auszüge aus dem Hardware-Katalog mit zwei CPUen und zwei digitalen Ausgabebaugruppen, jeweils in normaler Ausführung (blau) und F-Ausführung (gelb):

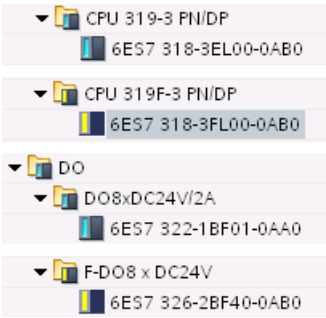

Über die farbliche Kennzeichnung können Sie einfach und schnell die F-Komponenten im Hardware-Katalog identifizieren.

Anzeige der F-Komponenten im Hardware- und Netzwerkeditor

Wenn Sie fehlersichere Geräte und "normale" Geräte in den grafischen Sichten des Hardware- und Netzwerkeditors verwenden, können Sie diese optisch einfach über die gelbe Kennzeichnung des Geräts erkennen.

Im folgenden Bild sehen Sie zwei CPUen. Die F-CPU besitzt eine gelbe Kennzeichnung:

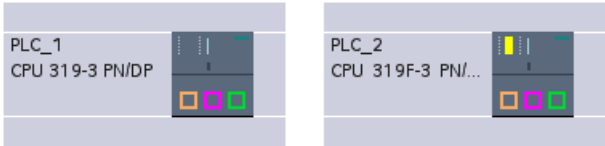

Fehlersichere Module auf Baugruppenträgern oder in Geräten können in der Gerätesicht auch an einer gelben Kennzeichnung erkannt werden. Da nicht-netzfähige Module in der Netz- oder Topologiesicht nicht angezeigt werden, werden Geräte mit nicht-netzfähigen F-Komponenten in der Netz- und Topologiesicht durch ein besonderes Symbol dargestellt.

Im folgenden Bild sehen Sie eine "normale" CPU und eine F-CPU in der Netzsicht, die aber beide F-Komponenten beinhalten:

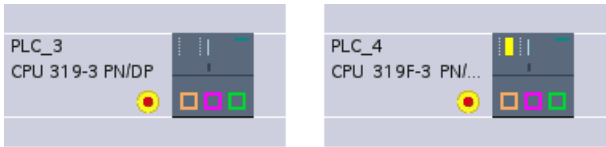

Über die farbliche Kennzeichnung der fehlersicheren Geräte und Module können Sie auch in den grafischen Sichten des Hardware- und Netzwerkeditors schnell die F-Komponenten identifizieren.

1.4.1.10 Eigenschaften zentraler Baugruppen ändern

Jede Hardware-Komponente (Baugruppe, Schnittstelle oder Schnittstellenmodul) hat voreingestellte Eigenschaften, z. B. bei Analogeingabebaugruppen voreingestellte Messarten und Messbereiche.

Eigenschaften zentraler Baugruppen

Von besonderer Bedeutung für das Systemverhalten sind die Eigenschaften der CPUs. Bei einer CPU können Sie z. B. einstellen:

- Anlaufverhalten
- Lokaldatenbereiche
- Prioritäten für Alarme
- Speicherbereiche
- Remanenzverhalten
- Taktmerker
- Schutzstufe
- Passwort

Was einstellbar ist und in welchen Wertebereichen, wird durch die Eingabemöglichkeiten vorgegeben. Nicht editierbare Felder sind ausgegraut, bzw. erscheinen nicht im Eigenschaftsfenster.

In den Eigenschaften einer Schnittstelle können Sie z. B. eine integrierte MPI- oder PROFIBUS-DP-Schnittstelle parametrieren und mit einem bestehenden Subnetz vernetzen.

Voraussetzung

Sie haben die Hardware-Komponenten, deren Eigenschaften Sie ändern wollen, bereits auf einem Baugruppenträger angeordnet.

Vorgehen

Um die Eigenschaften und Parameter der Hardware-Komponenten zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die Hardware-Komponente, die Sie bearbeiten möchten.
   Sie können auch einzelne Teile einer Hardware-Komponente selektieren, z. B.:
   - Schnittstelle
   - Eingang
   - Ausgang
2. Bearbeiten Sie die Einstellungen für das gewählte Objekt:
   - In der Geräteübersicht können Sie z. B. Adressen und Namen bearbeiten.
   - Im Inspektorfenster stehen Ihnen zusätzliche Einstellmöglichkeiten zur Verfügung.

Sie müssen Ihre Eingabe nicht bestätigen, die geänderten Werte werden sofort übernommen.

Weitere Möglichkeiten zur Parametrierung

Für Automatisierungssysteme der S7-300/400 haben Sie z. B. bei Analogbaugruppen die Möglichkeit, Parameter im Anwenderprogramm einzustellen. Zu diesem Zweck rufen Sie im Anwenderprogramm die Systemfunktionen WR_PARM (SFC 55), DR_DPARM (SFC 56) und PARM_MOD (SFC 57) auf. Diese Einstellungen gehen aber bei einem Anlauf bzw. Neustart (Warmstart) verloren.

Weitere Informationen

Beachten Sie für weitere Informationen über die System- und Standardfunktionen für S7-300/400 das Referenzhandbuch zur Systemsoftware für S7-300/400.

1.4.1.11 Kopplung von Baugruppenträgern

Einführung

In Abhängigkeit des verwendeten Geräts gibt es bei einigen S7-300 CPUs die Möglichkeit, Baugruppenträger über Anschaltungsbaugruppen zu koppeln.

Kopplung von Baugruppenträgern

Um Baugruppenträger über Anschaltungsbaugruppen zu koppeln, muss ein weiterer Baugruppenträger aus dem Hardware-Katalog in die Gerätesicht gezogen werden. Anschließend wird in den zentralen Baugruppenträger eine Senderbaugruppe und in den neuen Baugruppenträger eine Empfängerbaugruppe gesteckt. Es wird automatisch eine Kopplung zwischen Sender- und Empfängerbaugruppe gezogen.

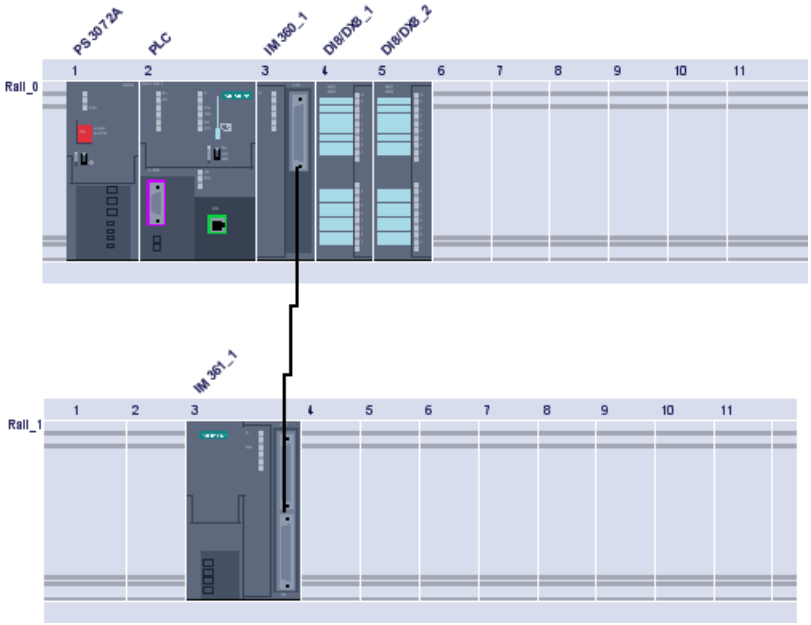

Bei S7-300 können Sie z. B. bis zu vier Baugruppenträger koppeln. Die Anschaltungsbaugruppen werden immer auf Steckplatz 3 gesteckt.

Hinweis

Sollte Ihre Konfiguration keine Erweiterung beinhalten, wird Steckplatz 3 immer als leer gekennzeichnet, auch wenn in der Realität die Baugruppe von Steckplatz 4 direkt an der CPU angeschlossen ist und somit keine Lücke besteht.

Verwenden Sie folgende Anschaltungsbaugruppen:

- Für die Erweiterung um genau einen Baugruppenträger:
  Baugruppenträger 0 und 1: IM 365
- Für die Erweiterung um bis zu 3 Baugruppenträgern:
  Baugruppenträger 0: IM 360
  Baugruppenträger 1 bis 3: IM 361

Wenn Sie eine manuelle Kopplung bevorzugen, löschen Sie zuerst die Verbindungslinie, die die Kopplung darstellt. Anschließend ziehen Sie mit gedrückter Maustaste eine neue Verbindungslinie zwischen den Schnittstellen der Sender- und Empfängerbaugruppe.

Erweiterbare Baugruppenträger

In Abhängigkeit des verwendeten Geräts gibt es bei einigen CPUs (z. B. IM 151-1) erweiterbare Baugruppenträger, deren Steckplatzgruppen Sie über die Pfeilsymbole oberhalb eines erweiterbaren Steckplatzes auf- und zuklappen können. Bei zugeklappten Steckplatzgruppen wird die erste und letzte Steckplatznummer der Gruppe angezeigt.

Das folgende Bild zeigt einen Ausschnitt eines erweiterten Baugruppenträgers:

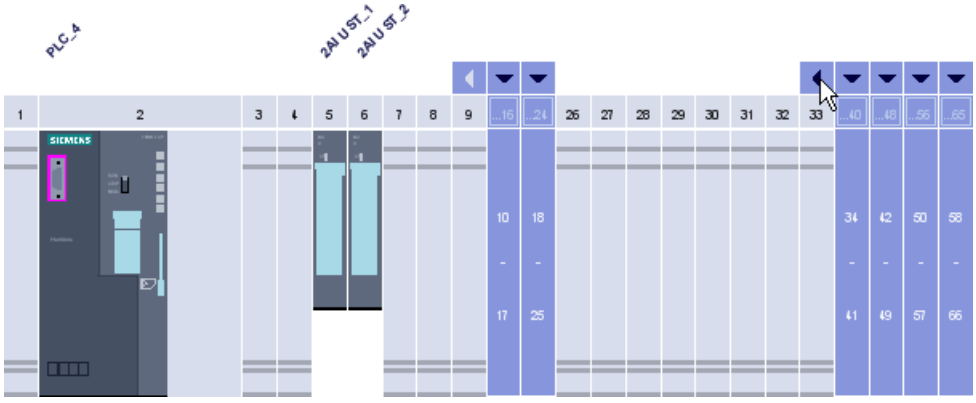

Steckplatzgruppen mit bereits gesteckten Baugruppen können nicht zugeklappt werden. Aus diesem Grund kann auch die erste Steckplatzgruppe mit dem Gerät nicht zugeklappt werden. Wenn eine Steckplatzgruppe nicht zugeklappt werden kann, erkennen Sie dies an dem gegrauten Pfeilsymbol.

1.4.1.12 Kopplung von Baugruppenträgern

Ergänzen eines zentralen Baugruppenträgers um Erweiterungsbaugruppenträger

Konfigurieren von Erweiterungsbaugruppenträgern bei SIMATIC 400

Bei SIMATIC 400 haben Sie durch die aufgrund der unterschiedlichen Baugruppenträger und Anschaltungsbaugruppen komplexe Erweiterungsmöglichkeiten.

Alle Erweiterungsbaugruppenträger, die an eine Schnittstelle der Sende-IM des zentralen Baugruppenträgers angeschlossen sind, bilden einen Strang.

Im folgenden Bild sind je drei Erweiterungsbaugruppenträger an eine Schnittstelle der Sende-IM angeschlossen.

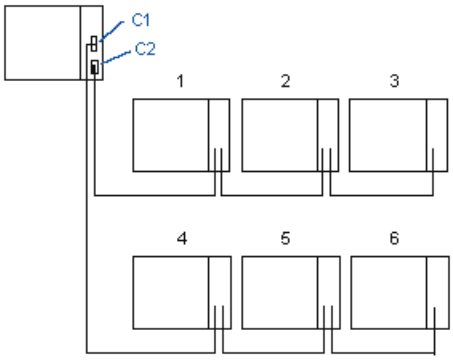

Regeln für die Kopplung von Erweiterungsbaugruppenträgern

Erweiterungsbaugruppenträger koppeln

Wenn Sie Erweiterungsbaugruppenträger (SIMATIC 400) an eine Schnittstelle der Anschaltungsbaugruppe (Sende-IM) des zentralen Baugruppenträgers koppeln, dann müssen jeweils folgende Eigenschaften bei Sende-IM und Empangs-IM übereinstimmen:

- Stromübertragung (mit/ohne)
- Art der Kopplung (zentral/dezentral)
- K-Bus-Übertragung (mit/ohne Alarmübertragung)

Steckplatzregeln der S7-400 für Erweiterungsbaugruppenträger

Für S7-400 gelten zusätzlich zu den allgemeinen Steckplatzregeln die folgenden Regeln für Baugruppen auf Erweiterungsbaugruppenträgern:

- Stromversorgungsbaugruppen auf Steckplatz 1 stecken
- Anschaltungsbaugruppe (Empfangs-IM) auf den äußeren rechten Steckplatz stecken (Steckplatz 9 bzw. Steckplatz 18)
- K-Bus-Baugruppen nur in Erweiterungsbaugruppenträger stecken, deren Nr. nicht größer als 6 ist (sonst können sie nicht angesprochen werden)

Anordnen des Erweiterungsbaugruppenträgers

Prinzipielles Vorgehen

Um Erweiterungsbaugruppenträger anzuordnen und Verbindungen zwischen den Anschaltungsbaugruppen in den Baugruppenträgern zu ziehen, gehen Sie folgendermaßen vor.

1. Wählen Sie eine CPU Baugruppenträger (z. B. CPU 417) aus dem "Hardware-Katalog".
2. Ziehen Sie die Steuerung per Drag&Drop in die Gerätesicht.
3. Ordnen Sie dem Baugruppenträger der Steuerung Baugruppen zu (z. B. Stromversorgung und IM 460).

4. Wählen Sie geeignete Erweiterungsbaugruppenträger (z. B. ER1) aus dem "Hardware-Katalog".
5. Ziehen Sie die Erweiterungsbaugruppenträger per Drag&Drop nacheinander in die Gerätesicht.
6. Ordnen Sie jedem Baugruppenträger Baugruppen zu (z. B. Stromversorgung oder IM 461). Wichtig: Die Anschaltungsbaugruppen müssen in allen Baugruppenträgern gesteckt sein, damit Sie die Baugruppenträger miteinander koppeln können!
7. Verbinden Sie die Schnittstelle der Sende-IM mit der der Empfangs-IM des gewünschten Baugruppenträgers.
   Verbindungslinien zeigen anschließend die Zuordnung der Baugruppenträger untereinander.

Im folgenden Bild sind je zwei Erweiterungsbaugruppenträger über eine Empfänger-IM an eine Schnittstelle der Sende-IM angeschlossen.

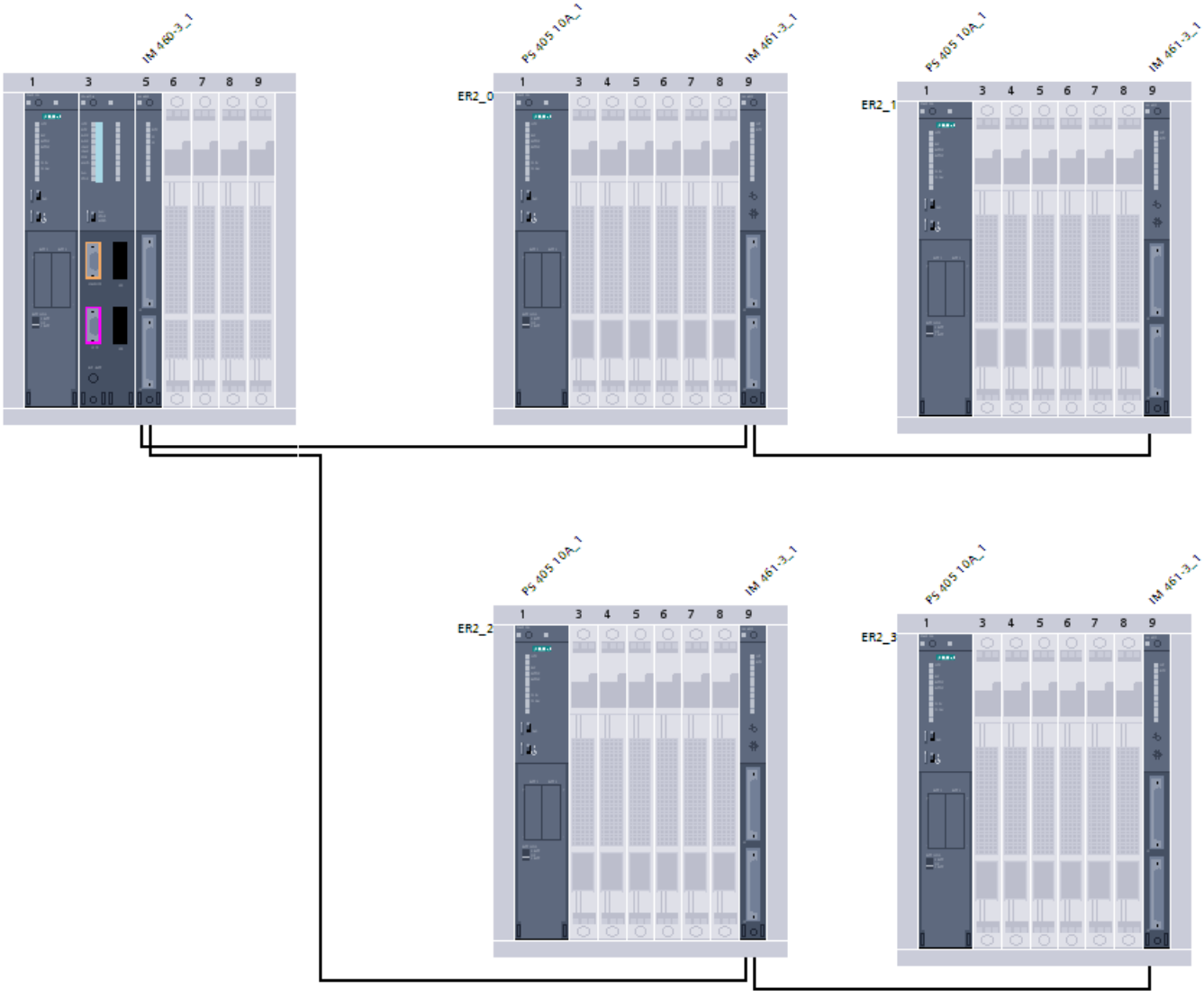

1.4.1.13 Baugruppen adressieren

Adressierung von Baugruppen

Einführung

In der Geräteübersicht sehen Sie in den Spalten E-Adresse und A-Adresse die Adressen oder Adressbereiche der Baugruppen.

Ein- und Ausgangsadressen werden automatisch beim Stecken von Baugruppen auf den Baugruppenträger vergeben. Die Adresse des ersten Kanals bildet die Anfangsadresse einer Baugruppe. Die Adressen der übrigen Kanäle ergeben sich aus dieser Anfangsadresse. Die Endadresse ergibt sich aus der baugruppenspezifischen Adresslänge.

Besonderheit: Eingangsadressen mit Sternchen (*)

In der Geräteübersicht werden bestimmte Adressen aus dem Eingangsadressbereich mit einem Sternchen (*) gekennzeichnet. In diesem Fall handelt es sich um Diagnoseadressen, z. B. von Anschaltungsbaugruppen, die keine Nutzdaten adressieren, sondern für die Systemdiagnose benötigt werden.

Unterscheidung zwischen Teilnehmeradressen und Peripherieadressen

Bei der Zuweisung von Adressen wird zwischen Teilnehmeradressen und Peripherieadressen (Ein-/Ausgangsadressen) unterschieden:

- Teilnehmeradressen sind Adressen von programmierbaren Baugruppen (MPI-, PROFIBUS-, Industrial Ethernet-Adressen). Sie werden benötigt, um die verschiedenen Teilnehmer eines Subnetzes zu adressieren, z. B. um ein Anwenderprogramm in eine CPU zu laden.
- Peripherieadressen werden dazu benötigt, um im Anwenderprogramm Eingänge zu lesen bzw. Ausgänge zu setzen.

Besonderheit: MPI-Adressen von FMs und CPs (S7-300)

Bei CPs und FMs mit eigener MPI-Adresse wird die MPI-Adresse von STEP 7 vorbelegt.

Die MPI-Adressen können anschließend geändert werden (im Inspektorfenster über "Eigenschaften > MPI-Schnittstelle").

Bei CPs und FMs ohne eigene MPI-Adresse adressiert die CPU diese Baugruppen über den K-Bus über die Nummer des Baugruppenträgers und Steckplatznummer.

Siehe auch

Ein- und Ausgangsadressen festlegen (Seite 1161)

Adressen einer Verwendungsstelle im Programm zuweisen (Seite 1175)

Adressierung von Baugruppen

Einführung

In der Geräteübersicht sehen Sie in den Spalten E-Adresse und A-Adresse die Adressen oder Adressbereiche der Baugruppen. Daneben gibt es noch weitere Adressen, die im folgenden erläutert werden.

E/A-Adresse (Peripherieadresse)

E/A-Adressen (Ein/Ausgabeadressen) werden dazu benötigt, um im Anwenderprogramm Eingänge zu lesen bzw. Ausgänge zu setzen.

Ein- und Ausgangsadressen werden automatisch beim Stecken von Baugruppen auf den Baugruppenträger vergeben. Die Adresse des ersten Kanals bildet die Anfangsadresse einer Baugruppe. Die Adressen der übrigen Kanäle ergeben sich aus dieser Anfangsadresse. Das Adressende ergibt sich aus der baugruppenspezifischen Adresslänge.

Teilnehmeradresse (z. B. Ethernet-Adresse)

Teilnehmeradressen sind Adressen von programmierbaren Baugruppen (Industrial Ethernet-Adressen). Sie werden benötigt, um die verschiedenen Teilnehmer eines Subnetzes zu adressieren, z. B. um ein Anwenderprogramm in eine CPU zu laden.

HW-Kennung zur Identifikation von Baugruppen bzw. Submodulen

Zusätzlich zu den E-Adressen und A-Adressen wird automatisch eine HW-Kennung (HW-ID) vergeben, die zur Adressierung und Identifizierung der Baugruppe verwendet wird. Auch Submodule (Einheiten einer Baugruppe), z. B. ein integrierter Zähler erhalten eine HW-Kennung. Die HW-Kennung besteht aus einer Ganzzahl und wird vom System bei Diagnosemeldungen mit ausgegeben, um die fehlerhafte Baugruppe bzw. das fehlerhafte Submodul zu lokalisieren. Außerdem verwenden Sie die HW-Kennung für eine Reihe von Anweisungen, um die entsprechende Baugruppe zu adressieren. Die HW-Kennung ist nicht änderbar.

Siehe auch

Wissenswertes zu HW-Kennungen (Seite 1163)

Ein- und Ausgangsadressen festlegen

Die Ein- und Ausgangsadressen werden zwar automatisch voreingestellt, Sie können aber die Adressierung nachträglich verändern.

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Die Baugruppe ist ein kompakter DP-Slave oder steckt in einem der folgenden Objekte:
  - Zentraler Baugruppenträger
  - Erweiterungsbaugruppenträger
  - DP-Slave
- Die CPU lässt freie Adresszuweisung zu.

Vorgehen

Um den voreingestellten Adressbereich zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Gerätesicht auf die Baugruppe, deren Anfangsadresse Sie einstellen wollen.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften" auf "E/A-Adressen".
3. Geben Sie unter "Anfangsadresse" die neue gewünschte Anfangsadresse ein.
4. Drücken Sie <Return> oder klicken auf ein beliebiges anderes Objekt um den geänderten Wert zu übernehmen.

Wenn Sie eine ungültige Adresse eingegeben haben, wird eine Meldung eingeblendet, in der Sie eine vorgeschlagene Adresse akzeptieren oder eine andere Adresse eingeben können.

Hinweis

Sie können die Adressen auch direkt in der Geräteübersicht ändern. Wenn durch die CPU keine Adressvorgabe ermöglicht wird, werden die Adressen gegraut dargestellt.

Bei Baugruppen innerhalb eines Lokalbussegments, gebildet durch eine FM (S7-300) oder bei speziellen FMs (S7-400), vergeben Sie zusätzlich eine weitere Anfangsadresse. Neben der Anfangsadresse für die CPU hat die Baugruppe dann auch eine Anfangsadresse für die FM. In der Geräteübersicht wird in diesem Fall immer die Anfangsadresse aus Sicht der FM angezeigt!

Wissenswertes zu Diagnoseadressen

Einführung

Diagnoseadressen dienen dazu, Komponenten zu adressieren, die selbst keine Nutzdaten haben, zum Beispiel Interfacemodule oder Ports.

Eingangsadressen als Diagnoseadressen

Um Komponenten zu adressieren, die selbst keine Nutzdaten haben, nützen Sie Diagnoseadressen. STEP 7 weist den Komponenten automatisch beim Konfigurieren eine E-Adresse „von oben“ zu, d.h. von der höchsten E-Adresse an abwärts. Diese E-Adresse wird Diagnoseadresse genannt.

Diese Eingangsadresse dient nicht dazu, Eingangswerte (Prozesswerte) einzulesen. Die Diagnoseadresse dient dazu, zum Beispiel in den Anweisungen "DPNRM_DG" oder "RDREC" Komponenten zu adressieren.

Beispiele

Um den Diagnosedatensatz eines DP-Slaves zu lesen, verwenden Sie bei der Anweisung "DPNRM_DG" (Parameter "LADDR") die Diagnoseadresse des Interfacemoduls dieses DP-Slaves.

Um den Datensatz zu lesen, in dem die Nachbarschaftsbeziehungen eines Ports enthalten sind, verwenden Sie bei der Anweisung "RDREC" die Diagnoseadresse dieses Ports (Parameter "ID").

Adressierung bei Eingabe- und Ausgabebaugruppen

Um eine Eingabebaugruppe zu adressieren, verwenden Sie bei den Anweisungen "RDREC" oder "WRREC" die Anfangsadresse. Sind einer Eingabebaugruppe die Adressen 5 und 6 zugeordnet, dann ist die 5 die Anfangsadresse. Setzen Sie den Parameter "ID" auf den Wert "DW#16#0005".

Um eine Ausgabebaugruppe zu adressieren, verwenden Sie bei den Anweisungen "RDREC" oder "WRREC" ebenfalls die Anfangsadresse. Setzen Sie für die Anfangsadresse 5 den Parameter "ID" auf den Wert "DW#16#8005". Bei einer Ausgabebaugruppe muss Bit 15 gesetzt (Wert "1") werden. Deshalb ergibt sich bei der Anfangsadresse 5 der Wert von DW#16#8005 für den Parameter "ID".

Zum Adressieren einer Mischbaugruppe (Eingabe- und Ausgabebaugruppe) tragen Sie im Parameter "ID" die kleinere der beiden Anfangsadressen ein. Haben der Eingangs- und Ausgangsbereich die gleiche Anfangsadresse, dann verwenden Sie die Kennung für eine Eingangsadresse (Bit 15 der Adresse nicht gesetzt, "0").

Siehe auch

Topologiedaten eines PROFINET-Systems auslesen (http://support.automation.siemens.com/WW/view/de/38566021)

Wissenswertes zu HW-Kennungen

Einführung

Jedes Hardware-Objekt, z. B. ein Modul, eine Schnittstelle oder ein Port eines Moduls oder auch Transferbereiche von I-Devices, hat eine HW-Kennung, die automatisch beim Konfigurieren einer Hardware-Komponente von STEP 7 vergeben wird.

Diese HW-Kennungen haben bei S7-1500 und auch bei S7-1200 dieselbe Funktion wie Diagnoseadressen von HW-Komponenten ohne Nutzdaten bzw. Anfangsadressen von HW-Objekten mit Nutzdaten bei S7-300/400. Auch bei Änderungen der E/A-Adressen der Baugruppen bleiben deren HW-Kennungen gleich. Bei Änderungen von E/A-Adressen müssen die Anweisungen, die über eine HW-Kennung adressieren, nicht angepasst werden.

HW-Kennungen im Register "Systemkonstanten" im Inspektorfenster

Wenn Sie ein HW-Objekt in der Gerätesicht markieren und im Inspektorfenster das Register "Systemkonstanten" wählen, werden Ihnen alle HW-Kennungen des HW-Objekts als Systemkonstanten mit Namen und Typ angezeigt.

Die Namen der HW-Kennungen sind hierarchisch aufgebaut. Sie bestehen aus maximal vier Hierarchieebenen, die jeweils durch eine Tilde "~" getrennt sind. Somit können Sie anhand des Namens den "Pfad" zu dem betreffenden Hardware-Modul erkennen.

Die Anzeige der HW-Kennungen ist abhängig vom Objekt, welches Sie in der grafischen Sicht selektiert haben. Wenn Sie ein ganzes Gerät in der Netzsicht oder Gerätesicht selektieren, werden in den Systemkonstanten alle HW-Kennungen von allen Objekten des entsprechenden Gerätes angezeigt (Module, Schnittstellen, Ports usw.). Wenn Sie nur ein einzelnes Modul oder einen einzelnen Port selektieren, wird in den Systemkonstanten nur die HW-Kennung des einzelnen selektierten Objekts angezeigt.

Beispiel

Die Systemkonstante mit dem Namen "Local~DI_16x24VDC_HF_1" bezeichnet die HW-Kennung für das Digitaleingabemodul mit dem Namen "DI_16x24VDC_HF_1" in der lokalen ("Local") CPU. Der Name des Digitalmoduls entspricht seinem editierbaren Namen in den Eigenschaften der Komponente (Inspektorfenster). Wenn Sie nur dieses eine Modul selektieren, wird auch nur die HW-Kennung dieses einen Moduls angezeigt.

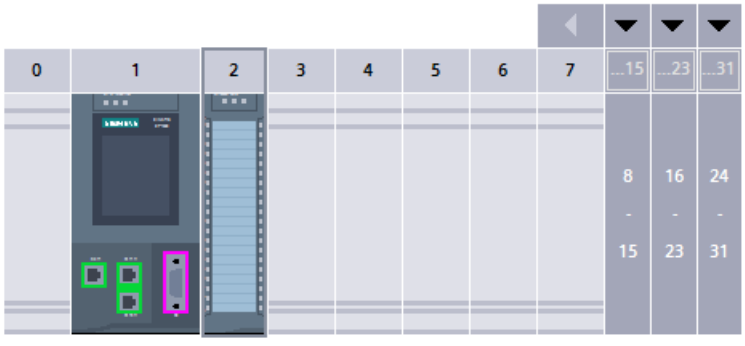

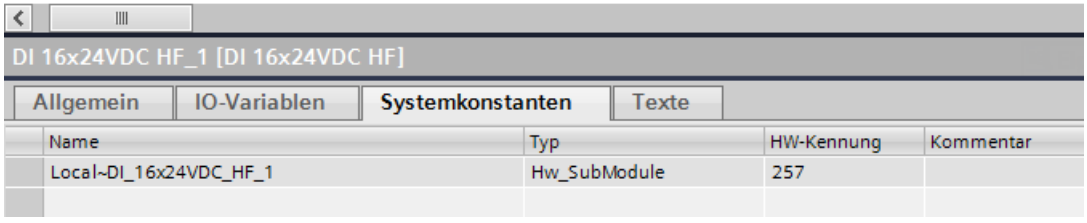

Hinweis

Um die Identifizierung von HW-Objekten zu erleichtern, vergeben Sie möglichst "sprechende" Namen für die HW-Objekte beim Konfigurieren der Hardware. Die Namen sollten die Zugehörigkeit zu einem Anlagenteil bzw. zu einer Station berücksichtigen.

HW-Kennungen in den Systemkonstanten der PLC-Variablentabelle

Für jede projektierte Hardware-Komponente werden von STEP 7 Einträge in den PLC-Variablen (Register "Systemkonstanten") abgefüllt. Diese Einträge sind nur lesbar. Neben dem ganzzahligen Wert werden auch der Name und der HW-Datentyp angegeben.

Beispiel:

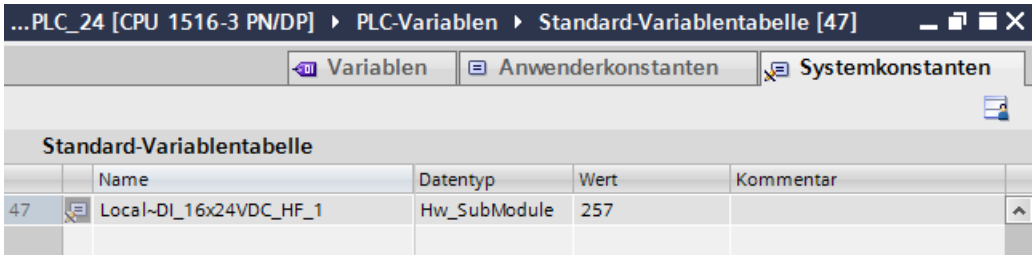

HW-Kennungen beim Programmieren

Wenn Sie in einer Anweisung ein HW-Komponente adressieren, z. B. mit dem Eingangsparameter "ID" bzw. "LADDR", wird Ihnen beim Doppelklick auf diesen Eingangsparameter eine Liste aller in Frage kommenden HW-Komponenten zur Auswahl angezeigt.

In den Startinformationen von z. B. Prozessalarm-OBs können Sie über die HW-Kennung die alarmauslösende Baugruppe identifizieren.

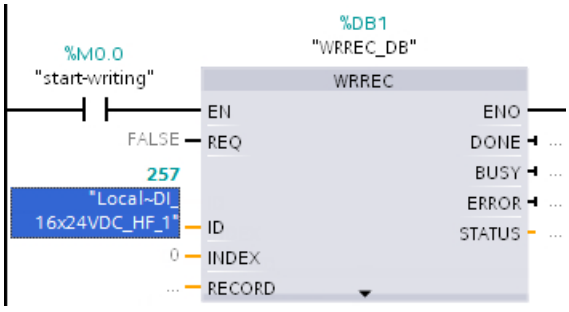

Weitere Informationen

Informationen zur Umwandlung von Adressinformationen (EA-Adressen, Steckplatz, HW-Kennung) finden Sie hier.

Siehe auch

Beispiele für die Verwendung von HW-Kennungen (Seite 1166)

Beispiele für die Verwendung von HW-Kennungen

HW-Kennungen verwenden Sie zum Adressieren von HW-Objekten wie z. B. Module, Submodule oder Geräte. Je nach Anweisung haben die HW-Kennungen unterschiedliche HW-Datentypen. Der zulässige HW-Datentyp ist für alle konfigurierten Komponenten aufgeführt in der Standard-Variablentabelle, Register "Systemkonstanten".

Im Folgenden sind für einige ausgewählte Anweisungen die zu verwendenden HW-Kennungen erläutert.

Hinweis

Informationen zur jeweiligen HW-Kennung

Für jede erweiterte Anweisung, die HW-Objekte adressiert, ist das zu verwendende HW-Objekt in der Hilfe dokumentiert. Die zur selektierten Anweisung gehörende Hilfe erreichen Sie über die Taste <F1>.

Empfehlung

Sie können den Namen des HW-Objekts oder die numerischen HW-Kennung des HW-Objekts im Programmiereditor eingeben.

Empfehlung: Vergeben Sie "sprechende" Namen für die HW-Objekte, z. B. für die Module. Lesen Sie Name und HW-Kennung aus dem Register "Systemkonstanten" in der Gerätesicht ab und tragen sie im Programmiereditor bei der entsprechenden Anweisung ein. Wenn Sie einen sprechenden Namen vergeben haben, finden Sie diesen Namen leichter in der Klappliste zur Auswahl der HW-Kennung am Bausteinparameter "ID".

Weitere Möglichkeit zur Eingabe der HW-Kennung

Sie haben die Möglichkeit, per Drag & Drop die HW-Kennung aus der Variablentabelle auf den Bausteinparameter "ID" zu ziehen.

Vorteil: Der HW-Datentyp wird vom Programmiereditor geprüft und gegebenenfalls abgewiesen, falls ein nicht zulässiges HW-Objekt gewählt wurde.

Bei einer numerischen Eingabe ist keine Prüfung möglich.

Wenn z. B. das zulässige HW-Objekt ein IO-Device ist, dann markieren Sie in der Netzsicht das gesamte IO-Device und lesen unter "Eigenschaften" (Register "Allgemein") im Inspektorfenster die HW-Kennung ab. Wenn Sie in diesem Fall nicht das IO-Device, sondern das Interfacemodul im IO-Device markieren, wäre die dann im Inspektorfenster sichtbare HW-Kennung für diese Anweisung unzulässig! Diesen Fehler bemerken Sie nicht bei Programmierung mit numerischer Eingabe der HW-Kennung.

Beispiel für die Eingabe einer HW-Kennung

In der Standard-Variablentabelle, Register "Systemkonstanten", steht z. B. folgender Name eines Moduls:
`myModule[AI]`

1. Öffnen Sie den Programmiereditor
2. Öffnen Sie die Standard-Variablentabelle (Ordner "PLC-Variablen")

3. Legen Sie die Editor-Fenster des Programmiereditors und der Standard-Variablentabelle nebeneinander.
   Klicken Sie dazu auf die Schaltfläche "Editorbereich vertikal teilen" in der Funktionsleiste.

4. Ziehen Sie "myModule[AI]" aus dem Register "Systemkonstanten" per Drag & Drop in das Feld "ID" bzw. "LADDR" der betreffenden Anweisung wie im Bild unten dargestellt.

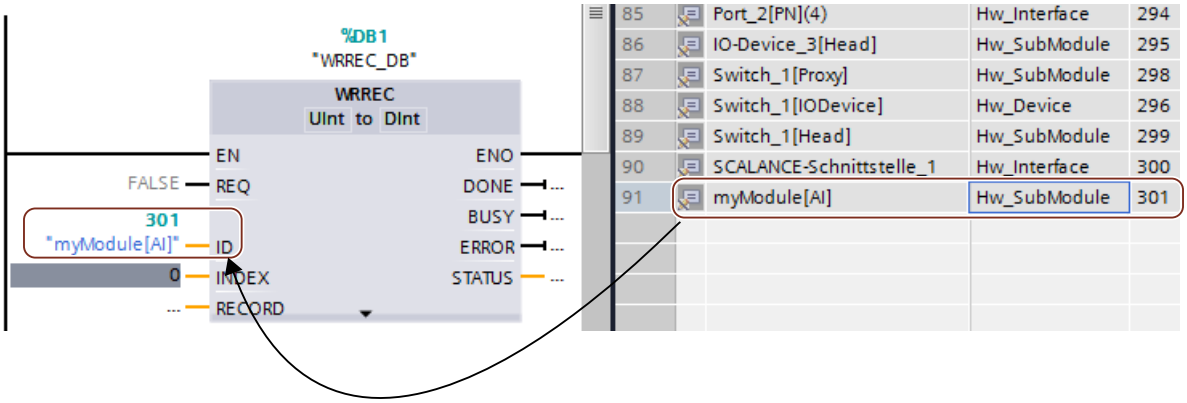

Beispiele für die Adressierung mit HW-Kennung

Erweiterte Anweisung D_ACT_DP: DP-Slave oder PROFINET IO-Device deaktivieren/aktivieren.

Der HW-Datentyp ist Hw_DpSlave oder Hw_Device. Zulässige HW-Objekte für den Eingangsparameter "LADDR" sind DP-Slaves bzw. IO-Devices.

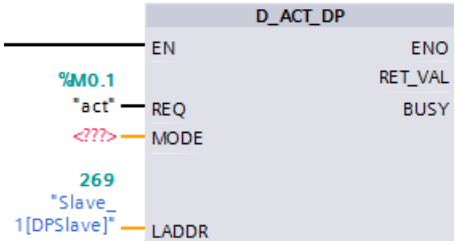

Erweiterte Anweisung DPSYC_FR: DP-Slaves synchronisieren/einfrieren.

Der HW-Datentyp ist Hw_Interface. Zulässiges HW-Objekt für den Eingangsparameter "LADDR" ist die DP-Master-Schnittstelle.

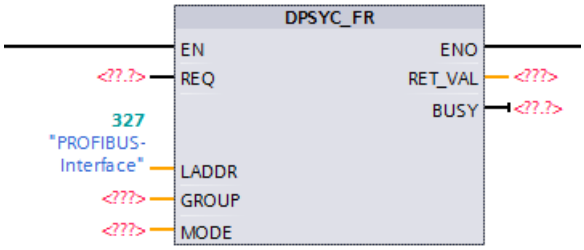

Erweiterte Anweisung RDREC: Datensatz lesen.

Der HW-Datentyp ist Hw_SubModule. Zulässige HW-Objekte für den Eingangsparameter "ID" sind Module bzw. Submodule, von denen der Datensatz gelesen werden soll.

Diesen HW-Datentyp verwenden Sie auch bei der erweiterten Anweisung DPRD_DAT.

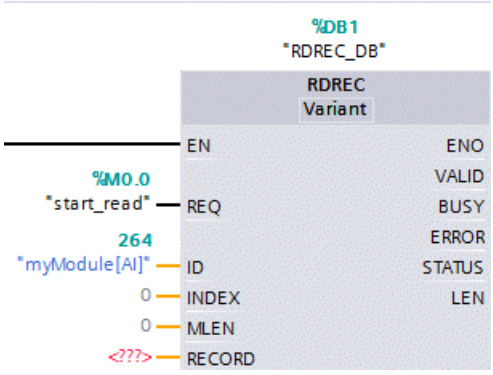

Erweiterte Anweisung WRREC: Datensatz schreiben. Die Anweisung wird z. B. dazu genutzt, um Datensätze auf ein Modul oder Submodul zu schreiben.

Der HW-Datentyp ist ebenfalls Hw_SubModule. Zulässige HW-Objekte für den Eingangsparameter "ID" sind Module bzw. Submodule, zu denen der Datensatz übertragen werden soll.

Diesen HW-Datentyp verwenden Sie auch bei der erweiterten Anweisung DPWR_DAT.

Hinweise:

- Bei Aufteilung eines Peripheriemoduls in mehrere Submodule (z. B. 1 x 8 AI => 8 x 1AI) ist die HW-Kennung eines Submoduls zu verwenden. Die HW-Kennung des Moduls ist in diesem Fall nicht zu verwenden.
- Bei Interfacemodulen eines dezentralen Peripheriegerätes, zu denen ein Datensatz für die Konfigurationssteuerung (Optionenhandling) übertragen werden soll, ist die HW-Kennung des Kopfes [Head] zu verwenden, siehe Bild.

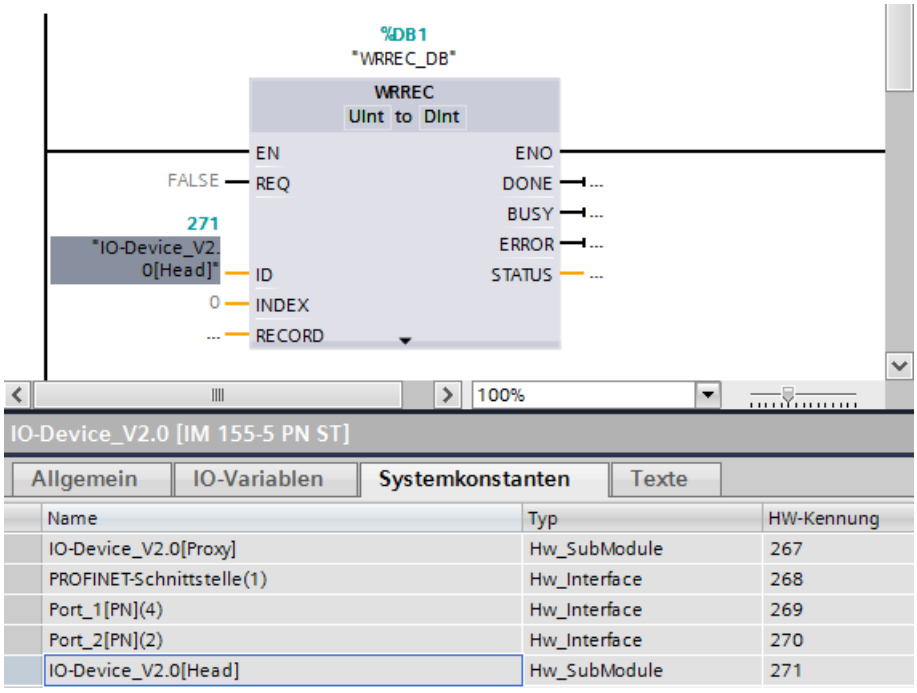

Erweiterte Anweisung Station_Info: Informationen eines IO-Devices auslesen.

Der HW-Datentyp ist Hw_Device. Zulässiges HW-Objekt für den Eingangsparameter "ID" ist ein IO-Device.

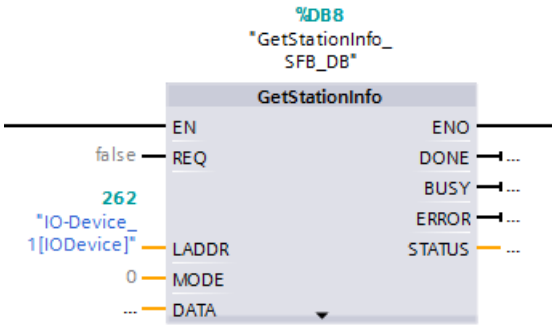

Weitere Informationen

Weitere InformationenInformationen zur Umwandlung von Adressinformationen (EA-Adressen, Steckplatz, HW-Kennung) finden Sie hier.

Siehe auch

Wissenswertes zu HW-Kennungen (Seite 1163)

Ein- und Ausgangsadressen in der Adressübersicht

Einführung

Die aktuell verwendeten Ein- und Ausgangsadressen können in der Adressübersicht tabellarisch angezeigt werden. Die Adressübersicht finden Sie im Inspektorfenster unter "Eigenschaften" der CPU.

Aufbau der Adressübersicht

Über verschiedene Optionskästchen können Sie einstellen, welche Objekte in der Adressübersicht angezeigt werden sollen:

- Eingänge: Anzeige der Eingangsadressen
- Ausgänge: Anzeige der Ausgangsadressen
- Adresslücken: Anzeige freier Adressräume
- Steckplatz: Anzeige der Steckplatznummer

Typischerweise werden in der Adressübersicht die folgenden Informationen angezeigt:

Tabellenkopf	Bedeutung
Typ	Zeigt an, ob es sich um einen Eingangs- oder einen Ausgangsadressbereich handelt.
Adr. von	Anfangsadresse des Adressbereiches.
Adr. bis	Endadresse des Adressbereiches.
Baugruppe	Baugruppe, von der der Adressbereich genutzt wird.
TPA	Nr. des Teilprozessabbildes. Zeigt an, ob der zyklische OB im Teilprozessabbild liegt.
OB	Zugeordneter Organisationsbaustein
DP	Nummer des Mastersystems. So lässt sich ermitteln, welche Slaves einem Master zugeordnet sind. Der Wert in Klammern gibt die PROFIBUS-Adresse der Hardwarekomponente an.
PN	Nummer des IO-Systems. Der Wert in Klammern steht für die Gerätenummer der Hardwarekomponente.
Baugruppenträger	Nummer des Baugruppenträgers, auf dem die Hardwarekomponente gesteckt ist.
Steckplatz	Nummer des Steckplatzes auf dem Baugruppenträger, in dem die Hardwarekomponente gesteckt ist.

Prozessabbild/Teilprozessabbild zuweisen

Einführung

Wenn sich Ein- und Ausgangsadressen in einem Prozessabbild befinden, können sie bei S7-300/400 im OB1-Prozessabbild (OB1-PA) oder in einem Teilprozessabbild (TPA) liegen.

Ein Prozessabbild bietet den Vorteil, dass Sie während der zyklischen Programmbearbeitung auf ein konsistentes Abbild der Prozesssignale zugreifen können. Sie haben bei S7-300/400 die Auswahl zwischen "OB1-PA" und "TPA 1" bis max. "TPA 15".

Jede Ein-/Ausgangsadresse, die Sie mit STEP 7 einem Teilprozessabbild zugewiesen haben, gehört nicht mehr zum OB 1-Prozessabbild der Ein-/Ausgänge! Ein-/Ausgangsadressen sind über das OB 1-Prozessabbild und alle Teilprozessabbilder nur einmalig vorgebbar.

Das Teilprozessabbild definieren Sie bei der Adresszuweisung (welche Ein-/Ausgangsadressen der Baugruppen in welchem Teilprozessabbild geführt werden). Die Aktualisierung des Teilprozessabbilds erfolgt entweder anwenderseitig mit Hilfe von Anweisungen oder systemseitig automatisch durch Kopplung an einen OB. Ausnahme: Teilprozessabbilder der Taktsynchronalarm-OBs werden nicht systemseitig aktualisiert, obwohl sie an einen OB (OB 61 bis OB 64) gekoppelt sind.

Jedes Teilprozessabbild können Sie im Anwenderprogramm mit speziellen Anweisungen aktualisieren. Verwenden Sie "UPDAT_PI" (SFC 26) für das Teilprozessabbild von Eingängen und "UPDAT_PO" (SFC 27) für das Teilprozessabbild von Ausgängen.

Die Kopplung eines Teilprozessabbilds an einen OB parametrieren Sie in den Eigenschaften der CPU (Bereich "Alarme").

Hinweis

Die Option ist nur bei Baugruppen mit Eingängen bzw. Ausgängen aktivierbar. Zudem muss die CPU Teilprozessabbilder unterstützen. Die Anzahl der zur Verfügung stehenden Teilprozessabbilder hängt ebenfalls von der CPU ab.

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Die Baugruppe ist ein kompakter DP-Slave oder steckt in einem der folgenden Objekte:
  - zentraler Baugruppenträger
  - Erweiterungsbaugruppenträger
  - DP-Slave
- Die CPU lässt freie Adresszuweisung zu.

Vorgehen

Um den voreingestellten Prozessabbild-Bereich zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht die Baugruppe, deren Anfangsadresse Sie einstellen wollen.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften" auf "E/A-Adressen".
3. Wählen Sie unter "Prozessabbild " das gewünschte Prozessabbild.

Hinweis

Wenn im Feld "Prozessabbild" der Eintrag "- - -" sichtbar ist (nicht änderbar), so bedeutet dies, dass für den angegebenen Adressbereich kein Prozessabbild zur Verfügung steht.

- Wählen Sie bei digitalen Eingabe- bzw. Ausgabebaugruppen eine niedrigere Anfangsadresse.
- Soll eine Analogbaugruppe über ein Prozessabbild angesprochen werden, wählen Sie ebenfalls eine niedrigere Anfangsadresse. Eine Verschiebung der Anfangsadresse in den Bereich des Adressbereiches für Prozessabbilder muss von der CPU unterstützt werden.
- Wenn die CPU eine parametrierbare Größe des Prozessabbildes hat, dann können Sie das Prozessabbild vergrößern (Zyklus-Eigenschaften der CPU), damit die eingegebene Adresse wieder im Prozessabbild liegt. Der Parameter heißt "Größe des Prozessabbildes ...".

Prozessabbild/Teilprozessabbild zuweisen

Einführung

Ein Prozessabbild bietet den Vorteil, dass Sie während der zyklischen Programmbearbeitung auf ein konsistentes Abbild der Prozesssignale zugreifen können. Sie haben bei S7-1500 die Auswahl zwischen "automatisch aktualisieren" und "TPA 1" bis "TPA 31".

Adressen dem automatisch aktualisierenden Prozessabbildbereich zuweisen

Das Prozessabbild definieren Sie bei der Adresszuweisung (Inspektorfenster, Bereich "E/'A-Adressen"). Wenn die Adressen eines Moduls im Prozessabbildbereich liegen sollen, dass systemseitig automatisch in jedem Programmzyklus aktualisiert wird, ist folgende Einstellung zu wählen:

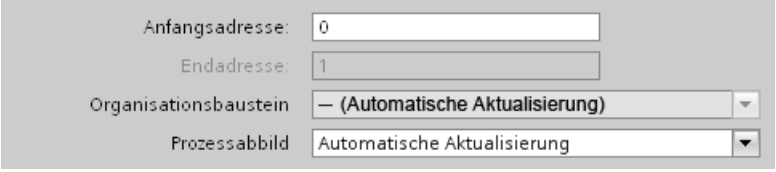

Zum Vergleich: Bei S7-300/400 Baugruppen wird diese Einstellung "OB1-Prozessabbild" genannt.

Wenn die Adressen in keinem Prozessabbildbereich liegen sollen, dann wählen Sie als Prozessabbild "Keines".

Adresse einem Teilprozessabbild zuweisen

Das Teilprozessabbild definieren Sie bei der Adresszuweisung (Inspektorfenster, Bereich "E/'A-Adressen"). Die Aktualisierung eines Teilprozessabbilds erfolgt entweder anwenderseitig mit Hilfe von Anweisungen oder systemseitig automatisch durch Kopplung an einen OB.

Teilprozessabbild an einen OB koppeln

Ebenfalls im Bereich E/A-Adressen eines Moduls koppeln Sie ein Teilprozessabbild an einen OB (Parameter "Organisationsbaustein").

OBs z. B. vom Typ "Cyclic Interrupt" (Weckalarm) wird automatisch das nächste freie Teilprozessabbild zugewiesen, das Sie ändern können. Nach dem Start des OBs wird systemseitig das zugeordnete Teilprozessabbild für die Eingänge aktualisiert. Am Ende des OBs werden systemseitig die Ausgänge des zugeordneten Teilprozessabbilds auf die Peripherieausgänge geschrieben. Die Teilprozessabbilder sind von der automatischen Aktualisierung ausgenommen.

Beispiel für Kopplung des Teilprozessabbilds 2 (TPA 2) an einen Weckalarm:

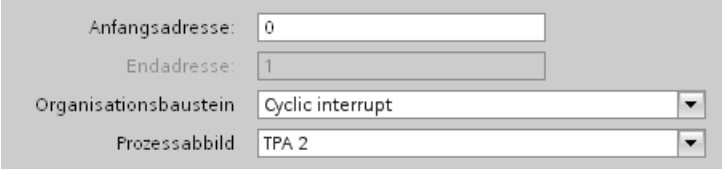

Ausnahme: Teilprozessabbilder der Taktsynchronalarm-OBs werden nicht systemseitig aktualisiert, obwohl sie an einen OB (OB 61 bis OB 64) gekoppelt sind.

Das zugeordnete Prozessabbild bzw. Teilprozessabbild wird in den Eigenschaften des zugeordneten OBs angezeigt (nicht änderbar).

Aktualisieren eines Teilprozessabbilds im Anwenderprogramm

Jedes Teilprozessabbild können Sie im Anwenderprogramm mit speziellen Anweisungen aktualisieren. Verwenden Sie "UPDAT_PI" für das Teilprozessabbild von Eingängen und "UPDAT_PO" für das Teilprozessabbild von Ausgängen.

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Das Modul bzw. Submodul hat E/A-Adressen.

Vorgehen

Um den voreingestellten Prozessabbild-Bereich zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das Modul, deren Anfangsadresse Sie einstellen wollen.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften" auf den Bereich "E/A-Adressen".
3. Wählen Sie unter "Prozessabbild" das gewünschte Prozessabbild.

Teilprozessabbilder in Anweisungen verwenden

Einführung

Anweisungen wie "UPDAT_PI" für das Aktualisieren von Teilprozessabbild-Eingängen und "UPDAT_PO" für das Ausgeben von Teilprozessabbild-Ausgängen an Peripherieausgänge verwenden den Eingangsparameter "PART" für die Eingabe des Teilprozessabbilds.

Im Folgenden wird gezeigt, wie Sie eine Anwenderkonstante für ein Teilprozessabbild definieren und für die Eingabe im Anwenderprogramm nutzen. Wenn Sie für Ihren Peripherieausbau verschiedene Teilprozessabbilder verwenden, erleichtert die Symbolik die Programmierung.

Voraussetzung

Sie haben den Peripherieausbau projektiert, Adressen und Teilprozessabbilder zugewiesen sowie erforderliche Bausteine angelegt.

Die Programmiersprache ist für dieses Beispiel auf KOP eingestellt.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Öffnen Sie die Variablentabelle und tragen Sie in das Register "Anwenderkonstanten" Konstanten vom Datentyp "Pip" (Process image part) ein. Für jedes verwendete Teilprozessabbild vergeben Sie einen Namen und wenn erforderlich einen Kommentar.

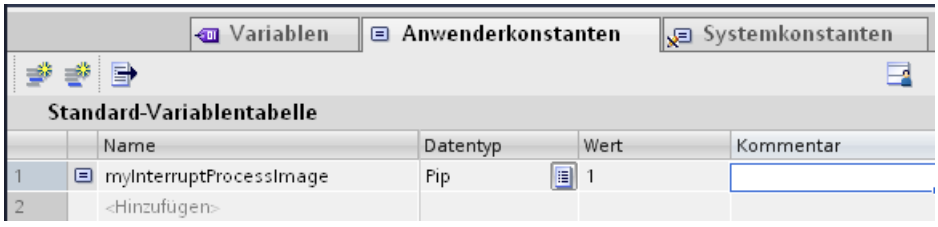

2. Öffnen Sie den Programmeditor (z. B. durch Doppelklick auf einen OB in der Projektnavigation).
3. Ziehen Sie per Drag & Drop die Anweisung UPDAT_PI in das Netzwerk ein.
4. Doppelklicken Sie auf den Eingangsparameter PART, so dass die Variablenauswahl öffnet. In der Variablenauswahl werden nur Teilprozessabbild-Konstanten zur Auswahl angeboten.
5. Wählen Sie das gewünschte Teilprozessabbild.

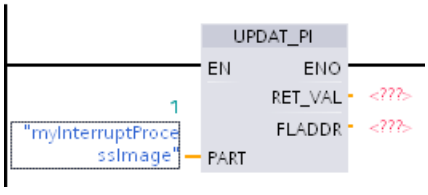

Adressen einer Verwendungsstelle im Programm zuweisen

Sie können Adressen von den E/A-Kanälen der Baugruppen direkt den Verwendungsstellen im Programm oder einer Variablentabelle zuordnen.

Voraussetzung

- Die Gerätesicht des Hardware- und Netzwerkeditors ist geöffnet.
- Die Zoomstufe in der Gerätesicht muss auf mindestens 200 % eingestellt sein, damit Sie die einzelnen E/A-Kanäle sehen können.
- Das Anweisungsfenster des Programmiereditors oder eine Variablentabelle ist geöffnet.

Vorgehen

Um E/A-Kanäle von Baugruppen den Verwendungsstellen im Programm oder einer Variablentabelle zuzuordnen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Gerätesicht zu der Baugruppe mit dem gewünschten E/A-Kanal.
2. Ziehen Sie mit gedrückter Maustaste die gewünschte E/A-Adresse auf die entsprechende Verwendungsstelle des Bausteins oder in die Variablentabelle.

Die Adresse der Baugruppe wird der Verwendungsstelle im Programm zugewiesen, bzw. als Variable in die Variablentabelle eingetragen.

Hinweis

Die Variable für einen Ein- oder Ausgang eines Bausteins kann auch auf den Ein- oder Ausgang einer Baugruppe gezogen werden, um die Variable mit dem E/A-Kanal der Baugruppe zu verknüpfen.

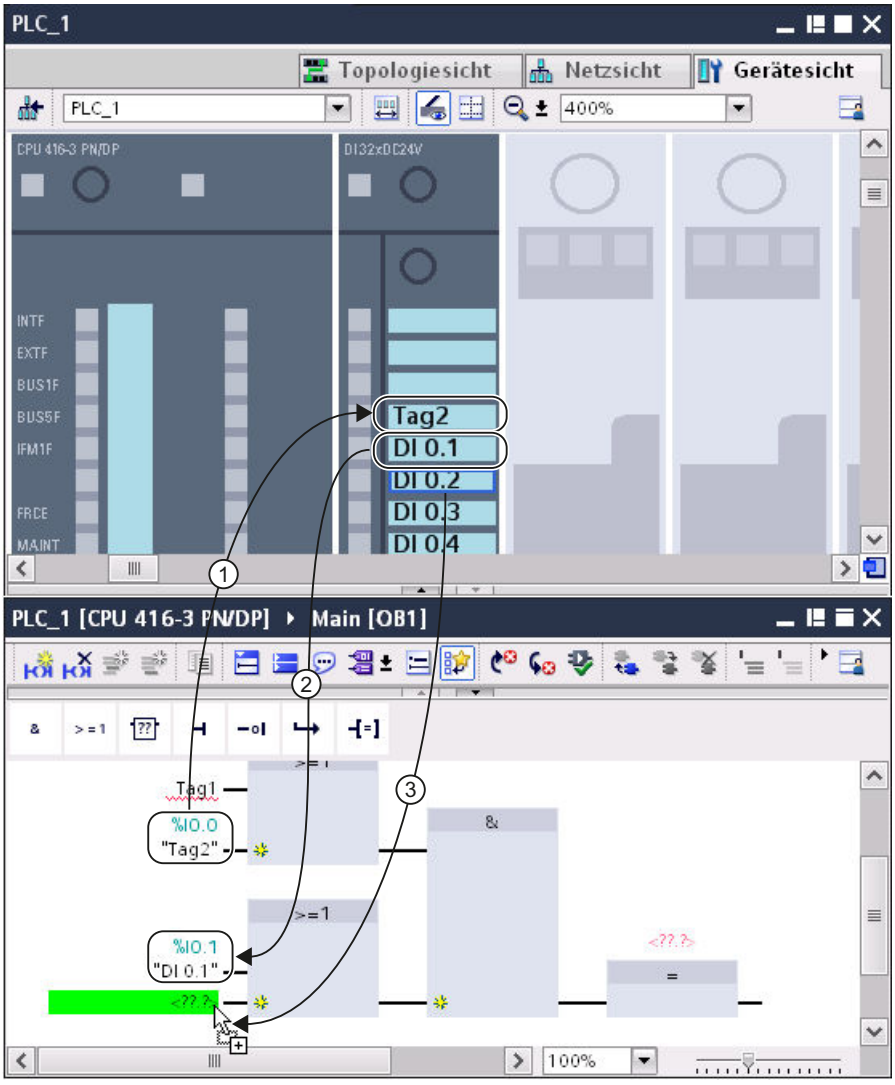

① Die Verwendungsstelle eines Bausteins wurde auf einen E/A-Kanal gezogen.

② Der Name eines E/A-Kanals wurde auf eine Verwendungsstelle im Programm gezogen.

③ Ziehen mit gedrückter Maustaste vom Namen eines E/A-Kanals auf den Eingang eines Bausteins.

1.4.1.14 Flexible Automatisierungskonzepte

Konfigurationssteuerung (Optionenhandling) für Geräte

Konfigurationssteuerung (Optionenhandling) zentral S7-1500

Wissenswertes zur Konfigurationssteuerung (Optionenhandling)

Funktionsprinzip

Durch die Konfigurationssteuerung ist es möglich, den Aufbau einer Steuerung (hier: zentraler Aufbau einer S7-1500) zu konfigurieren und von dieser Konfiguration abweichende Varianten (Optionen) zu betreiben.

- In einem Masterprojekt mit einer Maximalkonfiguration (Stationsmaster) sind alle Module konfiguriert, die in einer Reihe von ähnlichen Anlagenteilen oder Maschinen benötigt werden.
- Im Anwenderprogramm des Masterprojekts sind verschiedene Stationsoptionen für verschiedene Anlagenteile oder Maschinen sowie die Auswahl einer Stationsoption vorgesehen. Eine Stationsoption nutzt z. B. nur einen Teil der konfigurierten Module und diese Module sind in abweichender Reihenfolge gesteckt.
- Vor Ort wählt ein Bediener eine Stationsoption für eine konkrete Anlage aus. Er muss dazu das Projekt nicht ändern und damit auch keine geänderte Konfiguration laden.

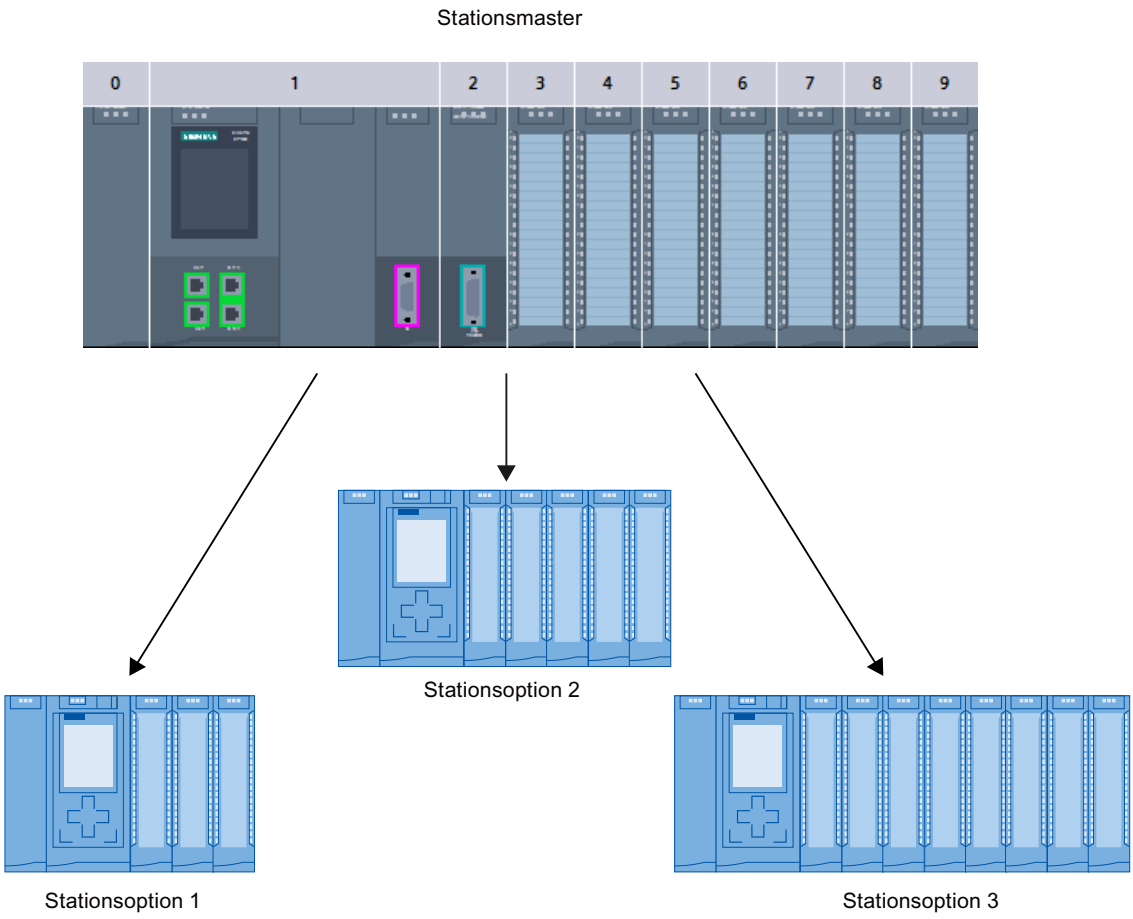

Der CPU wird durch einen von Ihnen programmierten Steuerdatensatz im Anlaufprogramm mitgeteilt, welche Module abweichend von der Projektierung im realen Aufbau fehlen oder welche Module sich abweichend von der Projektierung auf einem anderen Steckplatz befinden. Auf die Parametrierung der Module hat die Konfigurationssteuerung keinen Einfluss.

Durch die Konfigurationssteuerung können Sie flexibel den zentralen Aufbau variieren, solange sich die reale Konfiguration aus der projektierten Konfiguration eines Maximalausbaus ableiten lässt.

Im Folgenden ist beschrieben, wie Sie die Konfigurationssteuerung aktivieren (CPU-Parametrierung) und wie der erforderliche Datensatz aufgebaut sein muss.

Voraussetzung

- STEP 7 Professional ab Version V13
- CPU S7-15XX ab Firmware-Version 1.5. Module, welche die Funktion "Konfigurationssteuerung" unterstützen, haben auch in der Beschreibung (Infotext) des Hardware-Katalogs den Eintrag "Konfigurationssteuerung".

- Empfehlung: Bevor Sie ein neues Programm mit geändertem Steuerdatensatz laden, führen Sie Rücksetzen auf Werkseinstellungen durch. Durch diese Maßnahme werden inkonsistente Zustände vermieden, die durch einen eventuell vorhandenen, inkompatiblen Steuerdatensatz entstehen.
- Der Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf "Anlauf der CPU auch bei Unterschieden" (Voreinstellung).

Erforderliche Schritte

1. Aktivieren Sie bei der Projektierung der CPU den Parameter "Umkonfigurieren des Geräts über Anwenderprogramm ermöglichen" (Bereich "Konfigurationssteuerung").

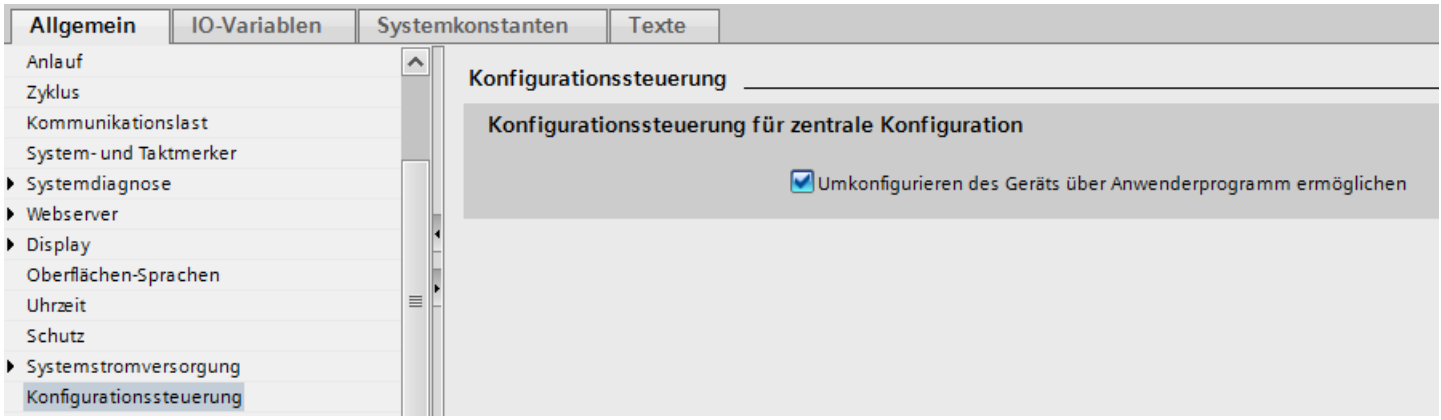

2. Erstellen Sie einen Steuerdatensatz (z. B. in einem Datenbaustein) gemäß der aktuellen Konfiguration nach dem unten beschriebenen Muster für den Steuerdatensatz. Der Steuerdatensatz hat die Nummer 196. Beachten Sie, dass Sie zunächste einen PLC-Datentyp erstellen müssen, der die Struktur des Steuerdatensatzes enthält und der Datenbaustein auf diesem PLC-Datentyp basiert.

ConfDB

		Name	Datentyp	Startwert	...	...	...	...	Kommentar
1		Static							
2		ConfigControl	Struct			✓	✓		
3		Block_length	USInt	9		✓	✓		Length of control data record, including header
4		Block_ID	USInt	196		✓	✓		Data record number
5		Version	USInt	4		✓	✓		
6		Subversion	USInt	0		✓	✓		
7		slot_0	USInt	16#FF		✓	✓		configured slot_0 / assigned "real" slot
8		slot_1	USInt	16#FF		✓	✓		configured slot_1 / assigned "real" slot
9		slot_2	USInt	16#FF		✓	✓		configured slot_2 / assigned "real" slot
10		slot_3	USInt	16#FF		✓	✓		configured slot_3 / assigned "real" slot
11		slot_4	USInt	16#FF		✓	✓		configured slot_4 / assigned "real" slot

3. Übertragen Sie den Steuerdatensatz im Anlaufprogramm zur CPU.
   Die Konfigurationssteuerung für die zentral gesteckten Module wirkt nur beim Betriebszustandswechsel der CPU von STOP nach RUN. Rufen Sie daher im Anlauf-OB die erweiterte Anweisung WRREC (Datensatz schreiben) auf und übertragen Sie den erstellten Steuerdatensatz an die CPU, siehe nächsten Abschnitt.
   Wenn im Anlauf-OB kein gültiger Steuerdatensatz übertragen wird, dann ist die Steuerung nicht betriebsbereit. Die CPU kehrt in diesem Fall vom Anlauf zurück in den STOP-Zustand.

Steuerdatensatz im Anlaufprogramm übertragen

Die CPU bearbeitet die Anweisung WRREC zur Übertragung des Steuerdatensatzes asynchron. Sie müssen daher WRREC in einer Schleife im Anlauf-OB wiederholt aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz übertragen ist.

Tipp: Verwenden Sie zur Programmierung der Schleife die Programmiersprache SCL mit der Anweisung REPEAT ... UNTIL.

```
REPEAT
    "WRREC_DB"(REQ := "start_config_control",
               ID := 33,
               INDEX := 196,
               LEN := "conf_LEN",
               DONE => "conf_DONE",
               BUSY => "conf_BUSY",
               ERROR => "conf_ERROR",
               RECORD := "ConfDB".ConfigControl,
               STATUS => "conf_STATUS");
UNTIL NOT "conf_BUSY"
END_REPEAT;
```

Im Folgenden finden Sie Erläuterungen zu einzelnen Bausteinparametern, die Sie im Kontext Konfigurationssteuerung mit bestimmten Werten versorgen müssen. Zu den übrigen Bausteinparametern siehe auch WRREC:

Parameter	Erläuterung
ID	HW-Kennung; ist immer 33 (dezimal) bei Konfigurationssteuerung für zentral angeordnete Module.
INDEX	Datensatznummer; ist immer 196 (dezimal) bei Konfigurationssteuerung für zentral angeordnete Module.
RECORD	Zu übertragender Steuerdatensatz. Zum Aufbau des Steuerdatensatzes siehe Abschnitt "Steuerdatensatz". Tipp: Der Bausteinparameter "RECORD" der Anweisung WRREC (ab V1.1) ist vom Datentyp "VARIANT" und verlangt daher eine Variable mit Datentyp. Wenn Sie den Steuerdatensatz in einem Datenbaustein ablegen, muss dieser Datenbaustein folglich auch einen Datentyp haben. Der erzeugte Datenbaustein darf nicht vom Typ "Global-DB" sein, sondern muss von einem Anwenderdatentyp abgeleitet sein. Gehen Sie wie folgt vor: 1. Erzeugen Sie zunächst einen neuen PLC-Datentyp (Anwenderdatentyp) mit der Struktur des Steuerdatensatzes und benennen ihn z. B. "ConfDB". 2. Erzeugen Sie einen neuen Datenbaustein. Wählen Sie als Typ für diesen Datenbaustein den neu erstellten Anwenderdatentyp, z. B. "ConfDB".

In den grafischen Programmiersprachen setzen Sie die Schleife mithilfe von Anweisungen zur Programmsteuerung um.

Beispiel in FUP: Mit der Anweisung LABEL (Sprungmarke) und mit der Anweisung JMP (Springe bei VKE=1) programmieren Sie eine Schleife.

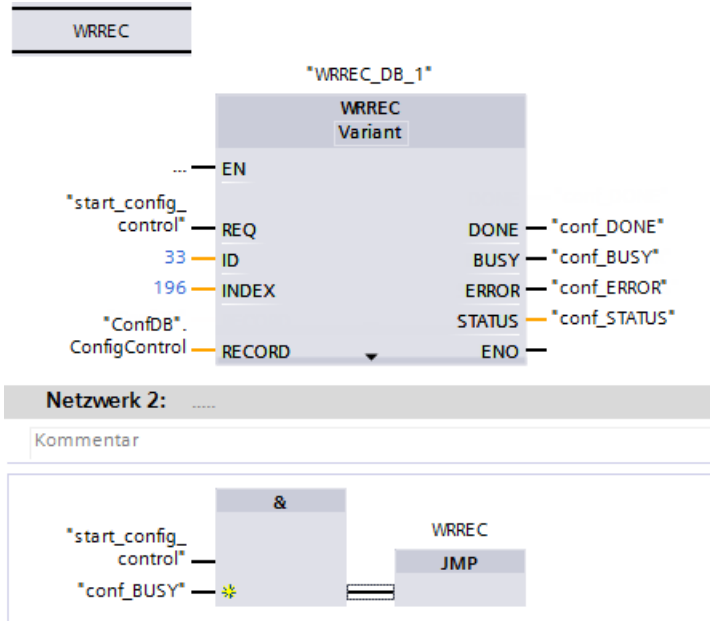

Anordnung der Module

Die nachfolgende Tabelle zeigt die Zuordnung der Steckplatznummern:

Steckplatz	Module	Bemerkung
0	Stromversorgungsmodul (optional)	Steckplatz links von der CPU
1	CPU	Steckplatz 1 ist immer die CPU
2 - 31	Peripheriemodule / Systemstromversorgungsmodule, je nach Aufbauvariante	Steckplätze rechts von der CPU

Steuerdatensatz

Für die Konfigurationssteuerung wird ein Steuerdatensatz 196 definiert, der eine Steckplatzzuordnung enthält.

Byte	Element	Kodierung	Erläuterung
0	Blocklänge	4 + Anzahl der Steckplätze	Header
1	Block-ID	196	
2	Version	4 (für zentrale Peripherie)	
3	Subversion	0	

Byte	Element	Kodierung	Erläuterung
4	Zuordnung projektierter Steckplatz 0	Realer Steckplatz 0	Steuerelement Beschreibt in jedem Element, welcher reale Steckplatz im Gerät dem projektierten Steckplatz zugeordnet ist. Der Aufbau eines Steuerelements ist im folgenden Abschnitt beschrieben.
5	Zuordnung projektierter Steckplatz 1	Realer Steckplatz 1 (immer 1, da die CPU immer auf Steckplatz 1 steckt)	
6	Zuordnung projektierter Steckplatz 2	Realer Steckplatz oder 16#FF	
7	Zuordnung projektierter Steckplatz 3	Realer Steckplatz oder 16#FF	
...	...	...	
4 + (max. Steckplatznr.)	Zuordnung projektierter maximaler Steckplatz	Realer Steckplatz oder 16#FF	

Aufbau eines Steuerelements

Ein Steuerelement enthält die Information, welches Modul auf welchem Steckplatz steckt.

Die Byte-Nummern repräsentieren die projektierten Steckplätze in aufsteigender Reihenfolge (siehe oben):

- Byte 4 steht für den projektierten Steckplatz 0
- Byte 5 steht für den projektierten Steckplatz 1
- Byte 6 steht für den projektierten Steckplatz 2
- usw.

Welchen Wert Sie im jeweiligen Byte eintragen müssen, ergibt sich aus folgender Regel:

- Wenn das Modul im realen Aufbau vorhanden ist, tragen Sie die reale Steckplatznummer des Moduls ein.
  - Beispiel 1: Das Modul vom projektierten Steckplatz 2 befindet sich auf Steckplatz 2. Tragen Sie im Byte 6 (= projektierter Steckplatz 2) den Wert 2 (= tatsächlicher Steckplatz) ein.
  - Beispiel 2: Das Modul vom projektierten Steckplatz 3 befindet sich auf Steckplatz 2. Tragen Sie im Byte 7 (= projektierter Steckplatz 3) den Wert 2 (= tatsächlicher Steckplatz) ein.
- Wenn das Modul im realen Aufbau fehlt, dann tragen Sie 16#FF (255) im Byte für den projektierten Steckplatz ein.

Regeln

Beachten Sie folgende Regeln:

- Wenn Sie die Konfigurationssteuerung aktiviert haben, ist die CPU ohne Steuerdatensatz nicht betriebsbereit. Wenn im Anlauf-OB kein gültiger Steuerdatensatz übertragen wird, kehrt die CPU vom Anlauf zurück in den STOP-Zustand. Die zentrale Peripherie wird in diesem Fall nicht initialisiert. Im Diagnosepuffer wird die Ursache für den Betriebszustand STOP eingetragen.
- Verwenden Sie zur Adressierung der Anweisung WRREC die HW-Kennung 33 (dezimal, für den Bausteinparameter ID), um den Steuerdatensatz zu schreiben.
- Der Steuerdatensatz wird remanent in der CPU gespeichert, so dass bei unveränderter Konfiguration kein erneutes Schreiben des Steuerdatensatzes 196 beim Neuanlauf erforderlich ist. Vor der Inbetriebnahme empfehlen wir, die CPU urzulöschen, um einen eventuell vorhandenen Steuerdatensatz zu löschen.
- Steckplatzeinträge im Steuerdatensatz außerhalb der projektierten Sollkonfiguration ignoriert die CPU.
- Den Steuerdatensatz können Sie verkürzen. Die Einträge bis zum letzten Steckplatz der aktuellen Sollkonfiguration müssen enthalten sein.
- Jeder reale Steckplatz darf nur einmal im Steuerdatensatz vorhanden sein.
- Ein realer Steckplatz darf nur einem projektierten Steckplatz zugeordnet werden.
- Verwenden von CPs/CMs:
  - Punkt-zu-Punkt-CPs/CMs verhalten sich hinsichtlich der Konfigurationssteuerung wie Digitalmodule und Analogmodule (keine Einschränkungen).
  - CPUs mit Firmwareversion ab V1.7:
    Wenn in der zentralen Konfiguration CPs/CMs stecken, also z. B. ein CM 1542-5 (DP-Master oder DP-Slave), dann sind diese CMs/CPs nicht durch die Konfigurationssteuerung beeinflussbar. Sie müssen daher diese Module auf ihren konfigurierten Steckplätzen belassen und mit ihrer projektierten Steckplatznummer im Steuerdatensatz eintragen ("realer Steckplatz = konfigurierter Steckplatz"). Die größtmögliche Flexibilität erhalten Sie, wenn Sie CMs/CPs direkt rechts neben die CPU stecken.
  - CPUs mit Firmwareversion bis V1.6:
    CMs und CPs sind generell nicht einsetzbar für die Konfigurationssteuerung.
- Auch Systemstromversorgungsmodule (PS) können der Konfigurationssteuerung unterliegen. Beachten Sie hierzu die Angaben im Systemhandbuch S7-1500 (http://support.automation.siemens.com/WW/view/de/59191792). Insbesondere für ein Systemstromversorgungsmodul (PS) auf Steckplatz 0 empfehlen wir, auf eine Umkonfigurierung zu verzichten.

Hinweis

Geänderte Konfiguration

Das Schreiben eines Steuerdatensatzes mit geänderter Konfiguration führt automatisch zum erneuten Anlauf mit dieser geänderten Konfiguration.

Durch diese Reaktion wird der remanent gespeicherte ursprüngliche Datensatz 196 gelöscht und der neue Datensatz 196 remanent gespeichert.

Verhalten im Betrieb

Auswirkung der Diskrepanz zwischen projektierter Konfiguration und realer Konfiguration:

- Für die Online-Anzeige und für die Anzeige im Diagnosepuffer (Modul o.k. oder Modul fehlerhaft) wird immer die Hardware-Konfiguration herangezogen; nicht die davon abweichende reale Konfiguration.
  Beispiel: Ein Modul liefert eine Diagnose. Konfiguriert ist dieses Modul auf Steckplatz 4, real steckt es auf Steckplatz 3 (fehlendes Modul; siehe Beispiel im nächsten Kapitel). In der Online-Sicht wird ein projektierter Steckplatz 4 als fehlerhaft angezeigt, im realen Aufbau zeigt das Modul auf Steckplatz 3 über LED-Anzeige einen Fehler an.

Wenn im Steuerdatensatz Module als fehlend eingetragen sind, verhält sich das Automatisierungssystem wie folgt:

- Im Steuerdatensatz als nicht vorhanden gekennzeichnete Module liefern keine Diagnose, ihr Zustand ist immer o.k. Der Wertstatus ist o.k.
- Schreibender Direktzugriff auf die Ausgänge oder schreibender Zugriff auf das Prozessabbild der nicht vorhandenen Ausgänge: Bleibt wirkungslos; es wird kein Zugriffsfehler gemeldet.
- Lesender Direktzugriff auf die Eingänge oder lesender Zugriff auf das Prozessabbild der nicht vorhandenen Eingänge: Wert "0" wird geliefert; es wird kein Zugriffsfehler gemeldet.
- Datensatz auf nicht vorhandenes Modul schreiben: Bleibt wirkungslos; es wird kein Fehler gemeldet.
- Datensatz von nicht vorhandenem Modul lesen: Es wird ein Fehler gemeldet, da kein gültiger Datensatz zurückgeliefert werden kann.

Fehlermeldungen

Beim Schreiben des Steuerdatensatzes werden Ihnen im Fehlerfall folgende Fehlermeldungen zurückgegeben:

Tabelle 1-85 Fehlermeldungen

Fehlercode	Bedeutung
16#80B1	Unzulässige Länge; Die Längenangabe im Datensatz 196 ist nicht korrekt.
16#80B5	Konfigurationssteuerung nicht parametriert.
16#80E2	Datensatz wurde im falschen OB-Kontext übertragen. Der Datensatz muss im Anlaufprogramm übertragen werden.
16#80B8	Parameterfehler; Modul meldet ungültige Parameter.

Weiterführende Informationen und Beispiele

Eine Bibliothek der Datensätze und Anwendungsbeispiele finden Sie hier: Vorlagen für die Datensätze (http://support.automation.siemens.com/WW/view/de/29430270)

Siehe auch

Systemhandbuch S7-1500 (http://support.automation.siemens.com/WW/view/de/59191792)

Beispiel für eine Konfigurationssteuerung

Im folgenden wird eine Konfiguration projektiert, bestehend aus Systemstromversorgung, CPU und 3 Peripheriemodulen.

Das Modul auf Steckplatz 3 ist in der ersten Ausbaustufe nicht vorhanden und wird von der Konfigurationssteuerung "ausgeblendet".

In der zweiten Ausbaustufe wird das zunächst ausgeblendete Modul auf dem letzten Steckplatz angeordnet. Der hinzugekommene Steckplatz wird über einen geänderten Steuerdatensatz der CPU bekannt gemacht.

Istkonfiguration mit fehlendem Modul

Die projektierte Konfiguration enthält alle Module, die in einem Endausbau vorhanden sein dürfen.

Das Modul, das in der projektierten Konfiguration auf Steckplatz 3 steckt, fehlt im realen Ausbau. Daher muss der Steckplatz 3 im Steuerdatensatz entsprechend mit "FF_H" (= nicht vorhanden) kodiert werden.

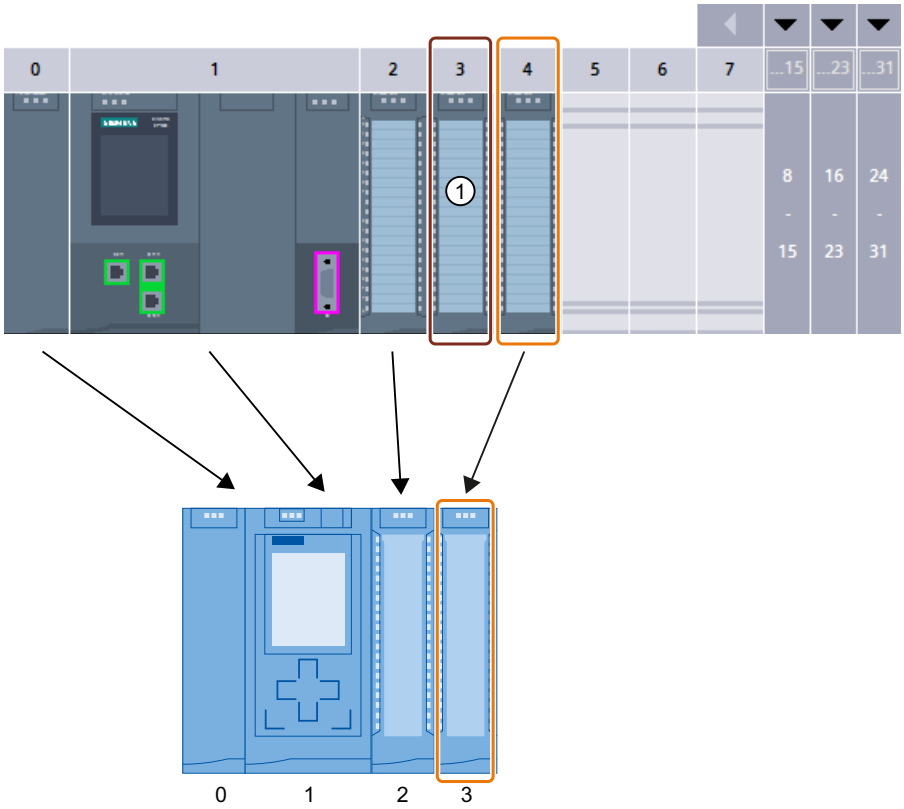

ConfDB

		Name	Datentyp	Startwert	...	...	...	...	Kommentar
1		Static							
2		ConfigControl	Struct			✓	✓		
3		Block_length	USInt	9		✓	✓		4+number of slots
4		Block_ID	USInt	196		✓	✓		
5		Version	USInt	4		✓	✓		
6		Subversion	USInt	0		✓	✓		
7		slot_0	USInt	0		✓	✓		configured slot_0 / assigned "real" slot 0
8		slot_1	USInt	1		✓	✓		configured slot_1 / assigned "real" slot 1
9		slot_2	USInt	2		✓	✓		configured slot_2 / assigned "real" slot 2
10		slot_3	USInt	16#FF		✓	✓		configured slot_3 / not used ①
11		slot_4	USInt	3		✓	✓		configured slot_4 / assigned "real" slot 3

① Modul fehlt im realen Aufbau

Istkonfiguration mit nachträglich hinzugefügtem Modul

Das in der projektierten Konfiguration vorhandene Modul auf Steckplatz 3 wird im realen Aufbau "hinten dran gehängt"; als letztes Modul auf Steckplatz 4 gesteckt.

Der Steuerdatensatz wird entsprechend angepasst.

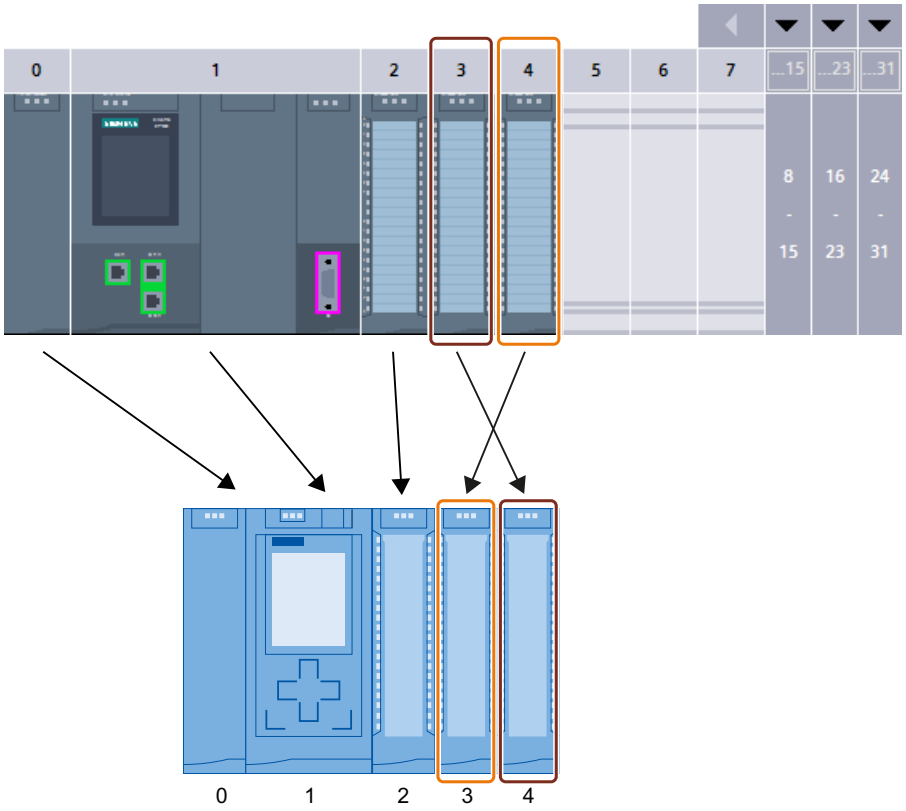

ConfDB

		Name	Datentyp	Startwert	...	...	...	...	Kommentar
1		Static							
2		ConfigControl	Struct			✓	✓		
3		Block_length	USInt	9		✓	✓		4+number of slots
4		Block_ID	USInt	196		✓	✓		
5		Version	USInt	4		✓	✓		
6		Subversion	USInt	0		✓	✓		
7		slot_0	USInt	0		✓	✓		configured slot_0 / assigned "real" slot 0
8		slot_1	USInt	1		✓	✓		configured slot_1 / assigned "real" slot 1
9		slot_2	USInt	2		✓	✓		configured slot_2 / assigned "real" slot 2
10		slot_3	USInt	4		✓	✓		configured slot_3 / assigned "real" slot 4
11		slot_4	USInt	3		✓	✓		configured slot_4 / assigned "real" slot 3

Steuerdatensatz für eine ET 200SP CPU

Steckplatzzuordnung

Die nachfolgende Tabelle zeigt die für die verschiedenen Module möglichen Steckplätze für eine ET 200SP CPU:

Tabelle 1-86 Steckplatzzuordnung

Module	Mögliche Steckplätze	Bemerkung
CPU	1	Steckplatz 1 ist immer die CPU
Peripheriemodule	2 - 65	Nach der CPU
Servermodul	2 - 66	Das Servermodul schließt den Aufbau der ET 200SP-Station nach der CPU/dem letzten Peripheriemodul ab.

Steuerdatensatz

Für die Konfigurationssteuerung bei einer ET 200SP CPU definieren Sie einen Steuerdatensatz 196 V2.0, der eine Steckplatzzuordnung enthält. Der maximale Steckplatz entspricht dabei dem Steckplatz des Servermoduls.

Die folgende Tabelle zeigt den Aufbau des Steuerdatensatzes mit Erläuterungen zu den einzelnen Elementen.

Tabelle 1-87 Konfigurationssteuerung: Struktur des Steuerdatensatzes 196

Byte	Element	Kodierung	Erläuterung
0	Blocklänge	4 + (Anzahl der Steckplätze × 2)	Header
1	Block-ID	196	
2	Version	2	
3	Version	0	

Byte	Element	Kodierung	Erläuterung
4	Steckplatz 1 des Stations-masters	Zuordnung Steckplatz 1 in der Stationsoption (immer 1, da die CPU immer auf Steckplatz 1 steckt)	**Steuerelement** Enthält die Information, welches Modul auf welchem Steckplatz steckt. Welchen Wert Sie im jeweiligen Byte eintragen müssen, ergibt sich aus folgender Regel: • Wenn das Modul in der Stationsoption vorhanden ist, tragen Sie die Steckplatznummer des Moduls ein. • Wenn das Modul als Leerplatz (mit BU-Cover) vorhanden ist, dann tragen Sie die Steckplatznummer des Moduls + 128 ein. (Beipiel: Modul als Leerplatz auf Steckplatz 3: Tragen Sie im Steuerelement 131 ein) • Wenn das Modul in der Stationsoption nicht vorhanden ist, dann tragen Sie 0 ein. **Zusatzfunktion** Enthält die Information, ob eine neue Potenzialgruppe in der Stationsoption geöffnet wird - durch Austausch einer dunklen BaseUnit gegen eine helle BaseUnit. • Wenn Sie eine dunkle BaseUnit durch eine helle BaseUnit austauschen, dann tragen Sie 1 als Zusatzfunktion ein. • Wenn Sie die BaseUnit aus dem Stationsmaster übernehmen, dann tragen Sie 0 als Zusatzfunktion ein.
5	Zusatzfunktion für den Steckplatz 1		
6	Steckplatz 2 des Stations-masters	Zuordnung Steckplatz in der Stationsoption	
7	Zusatzfunktion für den Steckplatz 2		
8	Steckplatz 3 des Stations-masters	Zuordnung Steckplatz in der Stationsoption	
9	Zusatzfunktion für den Steckplatz 3		
:	:	:	
4 + ((max. Steckplatz - 1) × 2)	Steckplatz Servermodul	Zuordnung Steckplatz Servermodul in der Stationsoption*	
4 + ((max. Steckplatz - 1)× 2) + 1	Zusatzfunktion für den Steckplatz Servermodul		

* Das Servermodul muss in der Stationsoption vorhanden sein und darf nicht als Leerplatz (BU-Cover) gekennzeichnet sein.

Bausteinbibliothek "OH_S71x00_Library"

Im Internet finden Sie die Bausteinbibliothek OH_S71x00_Library (https://support.industry.siemens.com/cs/#document/29430270?lc=de-WW) zum Download. Die Bausteinbibliothek enthält Datentypen mit der Struktur der Steuerdatensätze für das Dezentrale Peripheriesystem ET 200SP. Mit Hilfe dieser Datentypen können Sie aufwandsarm die Konfigurationssteuerung für Ihre flexible Automatisierungslösung realisieren.

Siehe auch

Systemhandbuch ET 200SP (https://support.industry.siemens.com/cs/ww/de/view/58649293)

Konfigurationssteuerung (Optionenhandling) für IO-Systeme

Verwendung von Serienmaschinen-Projekten

Die Verwendung von Serienmaschinen-Projekten bietet sich in den folgenden Fällen an:

- Die Automatisierungslösung (Maschine) wird bei Kunden mehrfach eingesetzt.
- Die Maschine wird in verschiedenen Anlagen bei verschiedenen Kunden eingesetzt.

Variabilität im Aufbau durch Konfigurationssteuerung von IO-Devices

Zusätzliche Flexibilität für eine Anlage oder Maschine erhalten Sie durch die Projektierung von optionalen IO-Devices und durch die Projektierung von Ports, die für eine Verschaltung durch das Anwenderprogramm vorbereitet werden.

Per Anwenderprogramm lassen sich optionale IO-Devices zur Konfiguration hinzunehmen, ohne das Projekt neu laden zu müssen. Auch die Verschaltung der Ports von IO-Devices und damit die Reihenfolge der IO-Devices lässt sich per Anwenderprogramm anpassen und damit die jeweils benötigte Konfiguration steuern - ohne Engineering-System!

Vorteile von Serienmaschinen-Projekten

- Ein zentral gewartetes Projekt (Konfiguration und Programm) kann ohne Änderungen auf verschiedene Maschinen gleichen Typs geladen werden.
- Vor Ort sind nur noch wenige aufwandsarme Anpassungen notwendig, um die Maschine in eine bestehende Netzwerk-Infrastruktur einzubinden.
- Modular aufgebaute Maschinen lassen sich einfach für verschiedene Anwendungen anpassen, indem Sie dem PROFINET IO-System über die Konfigurationssteuerung mitteilen, wie der aktuelle reale Aufbau des IO-Systems beschaffen ist.

Weitere Informationen

Detaillierte Informationen über Serienmaschinen-Projekte finden Sie hier (Seite 3138) und im Handbuch PROFINET mit STEP 7 (http://support.automation.siemens.com/WW/view/de/49948856).

1.4.1.15 OPC UA-Kommunikation nutzen

Wissenswertes zu OPC UA

OPC UA und Industrie 4.0

Einheitlicher Standard für den Informations- und Datenaustausch

Industrie 4.0 steht für die intensive Nutzung, Auswertung und Analyse der zahlreichen Daten aus der Produktion in IT-Systemen der Unternehmensebene. Mit Industrie 4.0 nimmt der Datenaustausch zwischen Produktions- und Unternehmensebene sehr stark zu. Eine Voraussetzung für das Gelingen ist ein einheitlicher Standard zum Informations- und Datenaustausch.

Der Standard OPC UA (OPC Unified Architecture) ist wegen seiner Unabhängigkeit von bestimmten Betriebssystemen, seinem sicheren Übertragungsverfahren und der semantischen Beschreibung der Daten für den Ebenen übergreifenden Datenaustausch besonders geeignet. OPC UA stellt nicht nur Daten bereit, sondern auch Informationen zu den Daten (z. B. Datentypen, Objekttypen), und damit wird ein maschineninterpretierbarer Zugriff auf die Daten möglich.

OPC UA bei S7-1500 CPUs

Bei OPC UA arbeitet ein System als Server und stellt anderen Systemen (Clients) die vorhandenen Informationen zur Verfügung.

OPC UA-Clients greifen z. B. lesend und schreibend auf Daten eines OPC UA-Servers zu. OPC UA-Clients rufen Methoden im OPC UA-Server auf.

Sie können mit einem Client online auf diese Daten zugreifen, auch z. B. auf Informationen zur Leistungsfähigkeit und Diagnose. Im Sprachgebrauch von OPC UA heißt diese Funktion "Browsen". Die Funktion "Subscription" erspart das regelmäßige Lesen einer Variablen - ein Client lässt sich vom Server einfach Werte nur bei Änderung übermitteln.

Ein System kann sowohl Client als auch Server sein.

OPC UA-Server der S7-1500 CPU

Ab Firmware V2.0 ist eine S7-1500 CPU mit einem OPC UA-Server ausgestattet.

Die folgenden Kapitel beschreiben, wie Sie den OPC UA-Server der S7-1500 CPU konfigurieren und damit Daten und Methoden für OPC UA-Clients zur Verfügung stellen, sodass Clients lesend oder schreibend auf PLC-Variablen der CPU zugreifen und Server-Methoden aufrufen können.

Außerdem zeigen die folgenden Kapitel, wie Sie Companion Spezifikationen in den Adressraum des OPC UA-Servers einbinden.

OPC UA-Client der S7-1500 CPU

Ab Firmware V2.6 ist eine S7-1500 CPU zusätzlich auch mit einem OPC UA-Client ausgestattet.

Die folgenden Kapitel zeigen, wie Sie mit standardisierten Anweisungen (PLCopen-Funktionsbausteine) ein Anwenderprogramm erstellen, das als OPC UA-Client Daten von einem OPC UA-Server liest oder Daten zu einem OPC UA-Server schreibt oder Methoden von einem OPC UA-Server aufruft.

STEP 7 (TIA Portal) hilft Ihnen beim Erstellen von Anwenderprogrammen mit einem Editor für Client-Schnittstellen und einer Parametrierung für OPC UA -Verbindungen.

Die OPC UA-Anweisungen für eine S7-1500 CPU als Client sind ausführlich in der Hilfe zu den Anweisungen beschrieben (Anweisungen > Kommunikation > OPC UA).

OPC UA-Client für Testzwecke

Um den Umgang mit OPC UA-Clients zu verdeutlichen, verwendet die folgende Beschreibung unterschiedliche OPC UA-Clients:

- "UaExpert" von Unified Automation. Ein umfangreicher Client, der kostenlos verwendet werden kann:
  Link zum Download von UaExpert (https://www.unified-automation.com/downloads/opc-ua-clients.html)
- "UA Sample Client" der OPC Foundation. Dieser Client ist kostenlos verfügbar für Anwender, die bei der OPC Foundation registriert sind:
  Link zum Download des Beispiel-Clients der OPC Foundation (https://opcfoundation.org)

Anwendungsbeispiel im Industry Online Support

Der Siemens Industry Online Support stellt ein kostenloses Anwendungsbeispiel mit einer Client-API zur Verfügung. Mit den Funktionen dieser Schnittstelle können .NET-Entwickler auf den OPC UA-Server einer S7-1500 zugreifen. Die Client-API setzt auf dem .NET OPC UA-Stack der OPC Foundation auf.

Das Anwendungsbeispiel zeigt z. B. den Aufbau von Verbindungen zwischen Server und Client. Das Beispiel zeigt ebenso das Lesen und Schreiben von PLC-Variablen.

Link zum Download: OPC UA .NET Client für den SIMATIC S7-1500 OPC UA Server (http://support.automation.siemens.com/WW/view/de/109737901)

Allgemeine Eigenschaften von OPC UA

Die wichtigsten Merkmale von OPC UA

- OPC UA ist unabhängig von einer bestimmten Betriebssystemplattform
  OPC UA kann zum Beispiel unter Windows, Linux, Mac OS X, einem Echtzeitbetriebssystem oder einem mobilen Betriebssystem (etwa Android) eingesetzt werden.
- OPC UA ist in verschiedenen Programmiersprachen umgesetzt.
  Die OPC Foundation hat den Standard OPC UA in mehreren Programmiersprachen implementiert: Stacks für ANSI C, .NET und Java sind verfügbar.
- Die OPC Foundation bietet den Java- und .NET-Stack sowie Beispielprogramme als Open Source Software an. Siehe Github (https://github.com/opcfoundation).
- Mehrere Unternehmen bieten Software Development Kits (SDK) an, die die Stacks der OPC Foundation und weitere Funktionen enthalten, welche die Entwicklung von Lösungen erleichtern.
  Vorteil der Nutzung von SDKs:
  - Support durch den Zulieferer
  - Getestete Software
  - Ausführliche Dokumentation
  - Klare Lizenzbedingungen (wichtig für Weiterverkauf von Lösungen)
- Skalierbarkeit
  OPC UA kann in Sensoren genauso verwendet werden wie in eingebetteten Systemen, in Steuerungen, in PC-Systemen und Smartphones wie auch in Servern, auf denen MES- oder ERP-Anwendungen laufen.
  Auch kann OPC UA und PROFINET gemeinsam genutzt werden. Beide Protokolle nutzen dieselbe Netzwerk-Infrastruktur.
- Einfaches Client-Server-Prinzip
  Ein OPC UA-Server stellt innerhalb eines Netzwerks Informationen bereit und ein OPC UA-Client ruft diese Informationen ab.
- Integrierte Sicherheitsmechanismen
  OPC UA verwendet Sicherheitsmechanismen auf unterschiedlichen Ebenen:
  - Der Aufbau sicherer Verbindungen zwischen einem OPC UA-Server und einem OPC UA-Client ist nur möglich, wenn sich Client und Server mit Hilfe von X.509-v3 Zertifikaten anmelden können und gegenseitig die Zertifikate anerkennen (Sicherheit auf Anwendungsebene). Verschiedene Security Policies sind möglich, auch eine ungesicherte Verbindung zwischen Server und Client (Security Policy: "Keine Security").
  - Ein Server kann für den Zugriff folgende Informationen vom Anwender fordern:
    - ein Zertifikat (nicht in STEP 7 projektierbar)
    - Benutzernamen und Passwort
    - keine Legitimation des Anwenders

  Die Sicherheitsmechanismen sind optional und konfigurierbar.

- Unabhängigkeit von einer bestimmten Transportschicht
  Die folgenden Transportmechanismen unterstützt OPC UA aktuell:
  - Übertragen von Nachrichten als Binärstrom direkt über TCP/IP
  - Übertragen von Nachrichten mit XML über TCP/IP und HTTP. Dieser Transportmechanismus lässt nur eine langsame Übertragung zu und wird deshalb kaum verwendet. Die S7-1500 CPUs unterstützen diesen Transportmechanismus nicht.

  Den binären Datenaustausch unterstützt jede OPC UA-Anwendung (durch die OPC UA-Spezifikation vorgeschrieben).
- Abbildung der PLC-Variablen
  Die Informationen des OPC UA-Servers (z. B. PLC-Variablen) sind als Knoten (Nodes) modelliert, die miteinander über Referenzen verbunden sind. Dadurch ist es möglich, mit einem OPC UA-Client von Knoten zu Knoten zu browsen und zu erfahren, welche Inhalte gelesen, beobachtet oder geschrieben werden können.
- Informationen
  OPC UA-Server stellen sehr viele Informationen bereit, z. B. zur CPU, zum OPC UA-Server selbst, zu den Daten und zu den Datentypen.
- Instanz-Konzept
  OPC UA basiert auf einem Typ-Instanz-Konzept. Sowohl Instanzen als auch deren Typdefinitionen sind zur Laufzeit verfügbar.
- Differenzierung der Funktionalität über Profile: Die S7-1500 unterstützt z. B. das Embedded UA-Server Profil

Siehe auch

Unified Automation (https://www.unified-automation.com)

OPC Foundation (https://opcfoundation.org)

Von der klassischen OPC-Schnittstelle zu OPC UA

Einheitliche Schnittstelle

Das klassische OPC läuft nur unter Windows-Betriebssystemen.

Um diese Einschränkung zu umgehen, entwickelte die OPC Foundation den Standard OPC UA.

Der Standard ist plattformunabhängig und verwendet ein optimiertes, TCP-basiertes Binärprotokoll für High-Performance Anwendungen.

Damit können unterschiedliche Systeme miteinander Daten austauschen, z. B.:

- Steuerungen mit MES- und ERP-Systemen
- Steuerungen von Siemens mit Steuerungen anderer Hersteller
- Smartphones mit Steuerungen
- Eingebettete Systeme mit Steuerungen
- Intelligente Sensoren mit Steuerungen

Adressierung von Knoten

Knoten im OPC UA-Adressraum werden durch eine NodeId (Node ID oder Node Identifier) eindeutig bestimmt.

Die NodeId besteht aus dem Identifier, Identifier Type und einem Namensraumindex. Namensräume werden verwendet, um Namenskonflikte zu vermeiden. Die OPC Foundation hat eine Reihe von Knoten definiert, die Auskunft über den jeweiligen OPC UA-Server geben. Diese Knoten sind im Namensraum der OPC Foundation zu finden und besitzen den Index 0.

Weiterhin hat die OPC Foundation Daten- und Variablentypen definiert.

Namespace

Alle Variablen bzw. Methoden einer S7-1500 befinden sich im Namensraum (Namespace) der Standard-Server-Schnittstelle "http://www.siemens.com/simatic-s7-opcua". Standardmäßig besitzt dieser Namensraum den Index 3. Wenn weitere Namensräume in den Server eingefügt oder vorhandene gelöscht werden sollten, kann sich der Index später ändern. Deshalb ist es erforderlich, den aktuellen Index des Namensraums beim Server zu erfragen, bevor Werte gelesen oder geschrieben werden.

Das folgende Bild zeigt ein beispielhaftes Ergebnis einer solchen Anfrage.

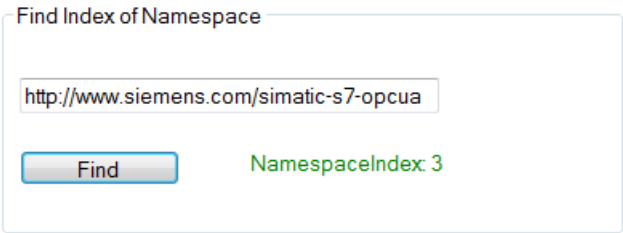

Identifier

Der Identifier entspricht dem Namen der PLC-Variablen in Anführungszeichen. Das Anführungszeichen ist das einzige Zeichen, das in STEP 7 nicht als Namensbestandteil erlaubt ist. Durch die Anführungszeichen werden Namenskonflikte vermieden.

Das folgende Beispiel liest den Wert der Variablen "StartTimer":

Der Identifier kann aus mehreren Komponenten bestehen. Die einzelnen Komponenten sind dann durch einen Punkt getrennt. Das folgende Beispiel liest den Array-Datenbaustein "MyDB" komplett ein. In diesem Datenbaustein befindet sich ein Array mit zehn Integer-Werten. Alle zehn Werte sollen auf einmal gelesen werden. Deshalb ist bei Array Range "0:9" eingetragen:

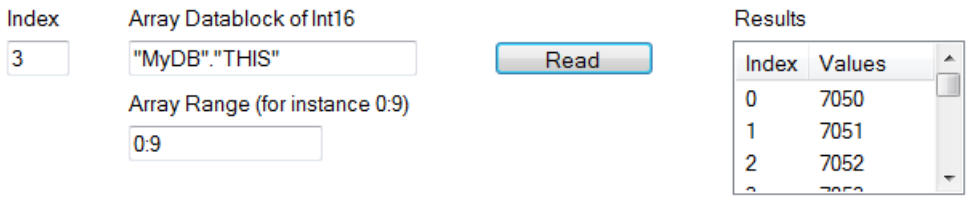

PLC-Variablen im Adressraum des OPC UA-Servers

Das folgende Bild zeigt, wo sich die PLC-Variablen des Beispiels im Adressraum des OPC UA-Servers befinden (Ausschnitt aus UA Client):

Der Datenbaustein "MyDB" ist ein globaler Datenbaustein. Deshalb befindet sich der Datenbaustein unterhalb des Knotens "DataBlocksGlobal". "StartTimer" ist eine Merker-Variable und wird deshalb unterhalb des Knotens "Memory" dargestellt.

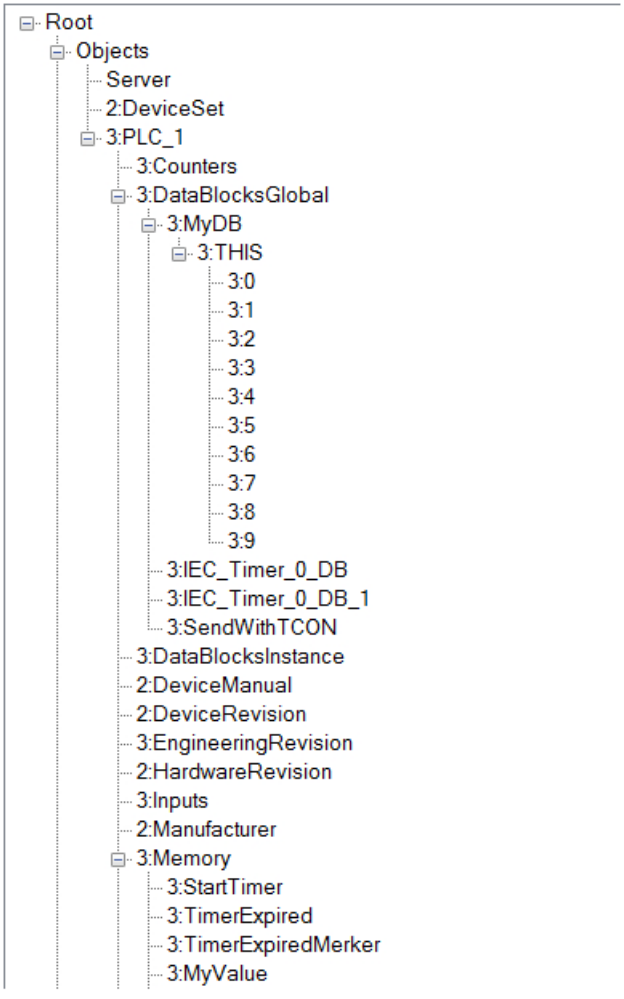

Methoden im Adressraum des OPC UA-Servers

Wenn Sie über Ihr Anwenderprogramm eine Methode implementieren, dann sieht das im Adressraum des OPC UA-Servers folgendermaßen aus (siehe AUTOHOTSPOT):

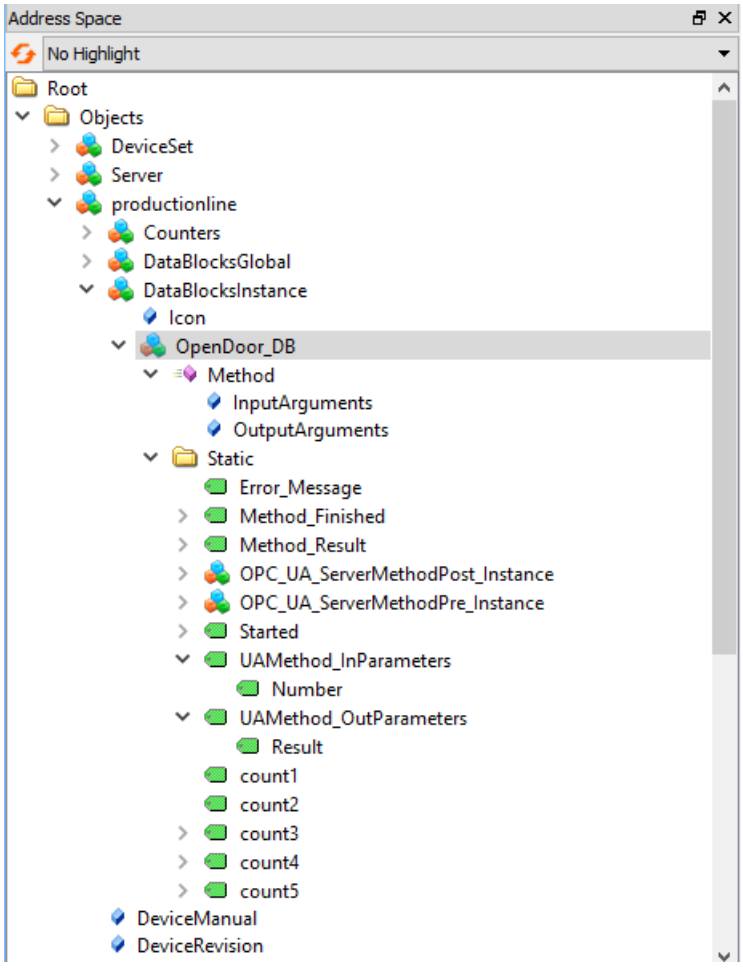

Siehe auch

Funktionshandbuch S7-1500 Kommunikation (https://support.industry.siemens.com/cs/ww/de/view/59192925)

Mapping von Datentypen

SIMATIC- und OPC UA-Datentypen

SIMATIC-Datentypen stimmen nicht immer mit OPC UA-Datentypen überein.

S7-1500 CPUs stellen SIMATIC-Variablen (mit SIMATIC-Datentypen) dem eigenen OPC UA-Server als OPC UA-Datentypen bereit, sodass OPC UA-Clients über die Server-Schnittstelle auf diese Variablen mit OPC UA-Datentypen zugreifen können.

Ein Client kann von einer solchen Variablen das Attribut "DataType" lesen und darüber den Original-Datentyp in SIMATIC rekonstruieren.

Beispiel

Eine Variable hat den SIMATIC-Datentyp "COUNTER". Sie lesen in der Tabelle COUNTER → UInt16. Sie wissen nun, dass Sie nicht umkodieren müssen, der COUNTER-Wert geht als UInt16 Datentyp über die Leitung.

Der Client erkennt am Attribut "DataType", dass die Variable eigentlich vom SIMATIC-Datentyp "COUNTER" ist. Mit diesem Wissen rekonstruiert der Client den Datentyp.

Tabelle 1-88 SIMATIC- und OPC UA-Datentypen

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
BYTE	BYTE → Byte
WORD	WORD → UInt16
DWORD	DWORD → UInt32
LWORD	LWORD → UInt64
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32
ULINT	UInt64
REAL	Float
LREAL	Double
S5TIME	S5TIME → UInt16
TIME	TIME → Int32
LTIME	LTIME → Int64
DATE	DATE → UInt16
TIME_OF_DAY (TOD)	TOD → UInt32
LTIME_OF_DAY (LTOD)	LTOD → UInt64

SIMATIC-Datentyp	OPC UA-Datentyp
DATE_AND_TIME (DT)	DT → Byte[8]
LDT	DateTime
DTL Besonderheit: Die Struktur können Sie mit einem OPC UA-Client nur komplett beschreiben. Lesend können Sie auf einzelne Elemente dieser Struktur zugreifen (z. B. auf "YEAR")	als Struktur gemappt
CHAR	CHAR → Byte
WCHAR	WCHAR → UInt16
STRING (Codepage 1252 bzw. Windows-1252)	STRING → String
WSTRING (UCS-2; Universal Coded Character Set)	String
TIMER	TIMER → UInt16
COUNTER	COUNTER → UInt16

Arrays

Ein Lese- bzw. Schreibauftrag bei OPC UA ist immer ein Array-Zugriff, d. h. grundsätzlich mit Index und Länge versehen. Eine Einzelvariable ist ein Sonderfall eines Arrays (Index 0 und Länge 1). Auf der Leitung wird der Datentyp einfach mehrfach hintereinander gesendet. Bei der Variablen zeigt das Attribut "DataType" auf den Basisdatentyp. Aus den Attributen "ValueRank" und "ArrayDimensions" ergibt sich, ob es sich um ein Array handelt und wie groß das Array ist.

Auf Arrays basierende Datentypen

Es gibt SIMATIC-Datentypen, bei denen ein OPC UA-Wert auf ein Array von Bytes abgebildet wird. Ein Array dieser Datentypen wird dann auf ein zweidimensionales Array abgebildet.

Beispiel: Der SIMATIC-Datentyp DATE_AND_TIME (DT) wird auf ein 8-Byte-Array (Byte[8]) gemappt, siehe Tabelle oben. Wenn Sie ein Array des SIMATIC-Datentyps DATE_AND_TIME (DT) definieren, dann zählt es als zweidimensionales Array.

Auswirkungen hat diese Tatsache z. B. auf die Verwendung von Systemdatenypen wie OPC_UA_NodeAdditionalInfo und OPC_UA_NodeAdditionalInfoExt:

Für die oben beschriebenen Datentypen müssen Sie den Systemdatentyp OPC_UA_NodeAdditionalInfoExt für mehrdimensionale Arrays verwenden statt OPC_UA_NodeAdditionalInfo.

Strukturen

Strukturen werden als ExtensionObject übertragen. Der Server der S7-1500 nutzt die binäre Darstellung für die Übertragung des ExtensionObjects über die Leitung, wobei die einzelnen Strukturelemente direkt hintereinanderliegen. Vorne befindet sich die NodeId des Datentyps, mit deren Hilfe ein Client den Aufbau der Struktur herausfindet.

Bei der OPC UA Specification <= V1.03 muss ein Client dazu das komplette DataTypeDictionary lesen, decodieren und interpretieren (sofern er es nicht bereits vorher Offline durch einen XML-Import gelernt hat).

Weitere Information

Nähere Informationen zur Abbildung der Basisdatentypen, aber auch von Arrays und Strukturen, finden Sie in der OPC UA Spezifikation Part 6, "Mappings" (siehe dort "OPC UA BINARY").

Wissenswertes zu OPC UA-Clients

Grundlagen zu OPC UA-Clients

OPC UA-Clients sind Programme, die Folgendes leisten:

- Zugriff auf Informationen von einem OPC UA-Server (z. B. einer S7-1500 CPU): lesend, schreibend, Subscriptions
- Methoden durch den OPC UA-Server ausführen lassen

OPC UA-Clients können jedoch nur auf Daten zugreifen, die dafür freigegeben sind (siehe "Schreib- und Leserechte verwalten (Seite 1222)").

Um eine Verbindung zu einem OPC UA-Server aufzubauen, benötigen Sie den Endpunkt des Servers (siehe "Endpunkte der OPC UA-Server (Seite 1217)").

Informationen aus dem OPC UA-Server auslesen

Wenn eine Verbindung zu einem Endpunkt des Servers besteht, dann können Sie die Navigationsfunktion des Clients nutzen: Sie navigieren von einem definierten Ausgangspunkt ausgehend (vom Wurzelknoten "Root" aus) durch den Adressraum des Servers.

Dadurch erhalten Sie unter anderem die folgenden Informationen:

- Freigegebene PLC-Variablen, Datenbausteine und Datenbaustein-Komponenten
- Namensraumindex und Identifier dieser PLC-Variablen, Datenbausteine und DB-Komponenten
- Datentypen der PLC-Variablen und DB-Komponenten
- Anzahl von Komponenten in Arrays (fürs Lesen und Schreiben von Arrays erforderlich)

Darüber hinaus erhalten Sie Information über den OPC UA-Server selbst, sowie Informationen über die S7-1500 basierend auf dem Standard "OPC UA for Devices" der OPC Foundation, zum Beispiel Seriennummer, Firmware-Version.

Daten vom Server lesen und zum Server schreiben

Sie kennen nun den Namensraumindex, Identifier und Datentyp von PLC-Variablen. Damit können Sie gezielt einzelne PLC-Variablen und DB-Komponenten wie auch ganze Arrays und Strukturen lesen. Beispiele für das Lesen von Bool´schen Variablen und Array-Datenbausteinen finden Sie unter Knoten adressieren (Seite 1195).

Regeln für den Zugriff auf Strukturen finden Sie hier (Seite 1301).

Mit den Informationen, die Sie beim Navigieren durch den Adressraum des Servers erhalten (Index, Identifier und Datentyp), können Sie mit dem OPC UA-Client auch Werte in die S7-1500 übertragen. Das folgende Beispiel überschreibt im Array-Datenbaustein "MyDB" die ersten drei Werte.

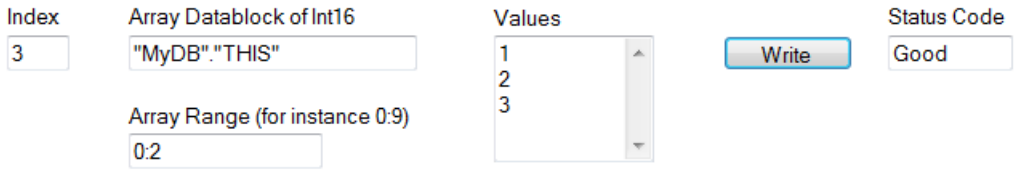

Bei "Array Range" geben Sie an, welche Komponenten des Arrays Sie überschreiben wollen. Am Status Code "Good" ist zu sehen, dass die Werte erfolgreich übertragen werden konnten. Sie können jedoch nur die Werte zur S7-1500 schreiben, nicht die Zeitstempel dieser Werte. Die Zeitstempel können nur gelesen werden.

Schnellerer Zugriff durch Registrierung

Die bisherigen Beispiele verwenden als Identifier-Zeichenketten, zum Beispiel "MyBD2"."THIS". Wenn aber als Identifier eine numerische NodeID anstelle einer String NodeID dient, dann sind Zugriffe wesentlich schneller. Deshalb sollten Sie bei regelmäßigen Zugriffen auf bestimmte Variablen die Funktionen "RegisteredRead" und "RegisteredWrite" nutzen.

Hierbei meldet Ihr Client zunächst die PLC-Variable beim Server an. Der Server sendet dann einen Identifier zurück, den der Client für die eigentlichen Zugriffe nutzt. Dieser Identifier gilt ausschließlich für die aktuelle Session und muss im Falle des Abbaus/Verlustes der Session neu erfragt werden.

Beim folgenden Beispiel wurde zunächst die Variable "StartTimer" beim Server registriert. Anschließend wird für das Setzen des Wertes die schnelle Funktion "RegisteredWrite" genutzt.

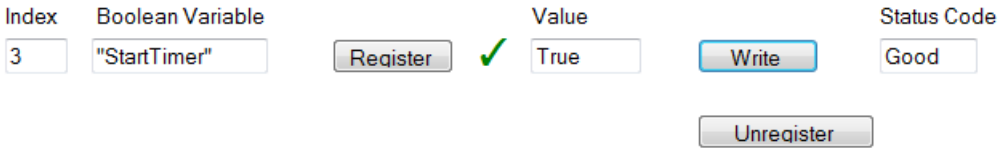

Nach dem gleichen Schema lässt sich auch die Funktion "RegisteredRead" nutzen, was insbesondere beim wiederkehrendem Auslesen von Daten sinnvoll ist. Beachten Sie jedoch, dass es je nach Anwendung sinnvoller sein kann, eine Subscription zu verwenden.

Empfehlung: Registrierungen platzieren Sie am besten im Hochlauf-Programm des OPC UA-Clients, da die Registrierung Zeit in Anspruch nimmt.

Beachten Sie, dass Sie in den Eigenschaften der S7-1500 CPU die maximale Anzahl registrierter Knoten einstellen können und die Clients diese Anzahl berücksichtigen müssen, siehe Allgemeine Einstellungen des OPC UA-Servers (Seite 1231).

Subscription

Mit "Subscription" wird eine Funktion bezeichnet, bei der nur Variablen übertragen werden, für die sich ein OPC UA-Client beim OPC UA-Server angemeldet hat. Der OPC UA-Server sendet für diese angemeldeten Variablen (Subscriptions) nur dann eine Nachricht an den OPC UA-Client, wenn sich ein Wert geändert hat. Durch die Überwachung dieser Variablen entfällt ein ständiges Abfragen durch den OPC UA-Client (Polling); die Netzlast wird reduziert.

Um diese Funktion zu nutzen, müssen Sie eine Subscription anlegen. Dazu geben Sie beim UA-Client das Sendeintervall ("Publishing Interval") vor und klicken auf die Schaltfläche "Create". Das Sendeintervall ist das Zeitintervall, in dem der Server neue Werte in einer Benachrichtigung (data change notification) an den Client sendet.

Beim folgenden Beispiel wurde eine Subscription angelegt: Alle 50 Millisekunden erhält hier der Client eine Nachricht mit den neuen Werten (Sendeintervall 50 ms).

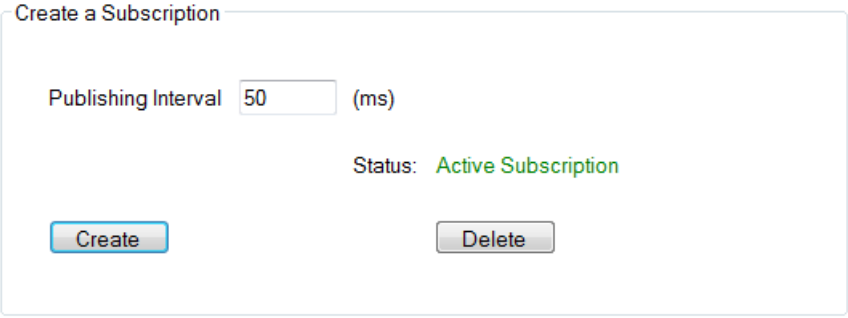

Server vor Überlast schützen

Den OPC UA-Server der S7-1500 CPU können Sie mithilfe des Parameters "Kleinstes Sendeintervall" so einstellen, dass er nicht extrem kurze, vom Client gewünschte Sendeintervalle bedient, siehe Einstellungen des Servers für Subscriptions (Seite 1232).

Beispiel: Ein Client möchte wie oben beschrieben in einem Sendeintervall von 50 ms bedient werden. Ein so kurzes Sendeintervall würde aber eine hohe Netzlast und eine hohe Belastung des Servers hervorrufen. Daher stellen Sie als "Kleinstes Sendeintervall" beim Server 1000 ms ein. Clients, die kürzere Sendeintervalle in ihrer Subscription fordern, werden so auf 1000 ms "heruntergebremst", der Server vor Überlastung geschützt.

Abtastung und Senden (Sampling & Publishing) im Rahmen einer Subscription sind Kommunikationsprozesse, die ebenso wie andere Kommunikationsprozesse (TCP/UDP/ Webserver-Kommunikation ...) mit der Priorität 15 von der CPU abgearbeitet werden. OBs mit höherer Priorität unterbrechen die Kommunikation. Wenn Sie die Abtast- und Sendeintervalle zu kurz einstellen, verursacht diese Einstellung eine hohe Kommunikationslast. Wählen Sie daher möglichst große Intervalle, die für die Anwendung noch ausreichend sind.

Informationen zur Konsistenz von Variablen finden Sie in Konsistenz von CPU-Variablen (Seite 1278).

Überwachung von PLC-Variablen

Wenn die Subscription angelegt ist, teilen Sie dem Server mit, welche Variablen er damit überwachen soll. Im folgenden Beispiel wurde die Variable "Voltage" der Subscription hinzugefügt.

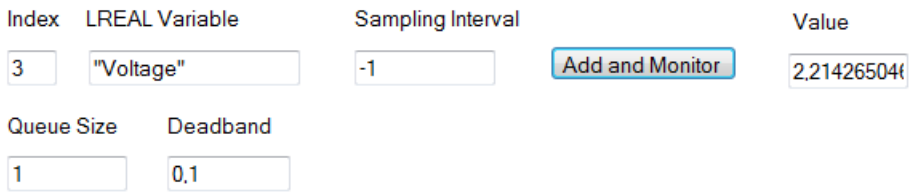

Die Variable "Voltage" enthält z. B. den Wert einer Spannungsgröße, die von einer S7-1500 CPU erfasst wird.

Das Abtastintervall ("Sampling Interval") enthält einen negativen Wert (-1). Dadurch wird festgelegt, dass die Default-Einstellung des OPC UA-Servers für das Abtastintervall verwendet wird. Die Default-Einstellung ist durch das Sendeintervall ("Publishing Interval") der Subscription festgelegt. Wenn Sie das kleinstmögliche Abtastintervall einstellen wollen, dann wählen Sie den Wert "0".

Die Länge der Warteschlange ist hier im Beispiel auf "1" festgelegt: Es wird immer nur ein Wert im Intervall von 50 Millisekunden aus der CPU gelesen und anschließend an den OPC UA-Client gesendet, wenn sich der Wert verändert hat.

Der Parameter "Deadband" ist im Beispiel "0,1": Wertänderungen müssen mindestens 0,1 Volt betragen, nur dann sendet der Server den neuen Wert an den Client. Kleinere Wertänderungen sendet der Server nicht. Mit diesem Parameter können Sie zum Beispiel Rauschen ausblenden: geringfügige Änderungen einer Prozessgröße, denen keine reale Bedeutung zukommt.

Security bei OPC UA

Security-Einstellungen

Gefahren begegnen

OPC UA erlaubt den Datenaustausch zwischen unterschiedlichen Systemen, sowohl innerhalb der Prozess- und Produktionsebene, als auch zu Systemen der Leit- und Unternehmensebene.

Diese Möglichkeit birgt auch Security-Risiken. Deshalb verwendet OPC UA eine Reihe von Sicherheitsmechanismen:

- Prüfung der Identität von OPC UA-Server und -Clients.
- Prüfung der Identität der Anwender.
- Signierter/verschlüsselter Datenaustausch zwischen OPC UA-Server und -Clients.

Die Sicherheitseinstellungen sollten nur in begründeten Fällen umgangen werden:

- Während der Inbetriebnahme
- Bei Inselprojekten ohne Ethernet-Verbindung nach außen

Wenn Sie z. B. beim "UA Sample Client" der OPC Foundation den Endpunkt "None" auswählen, dann gibt das Programm eine deutliche Warnung aus:

Warning: Selected Endpoint has no security.

Ebenso prüft STEP 7 beim Übersetzen Ihres Projekts, ob Sie die Einstellungsmöglichkeiten für den Schutz berücksichtigt haben und warnt bei möglichen Risiken. Dazu gehört auch eine OPC UA Security Policy mit der Einstellung "keine Security", was dem Endpunkt "None" entspricht.

Hinweis

Nicht erwünschte Security Policies deaktivieren

Wenn Sie bei den Secure-Channel-Einstellungen des S7-1500 OPC UA-Servers alle Security Policies aktiviert haben (Voreinstellung) - also auch den Endpunkt "None" (Keine Security) - dann ist der Datenverkehr zwischen Server und Client auch ungesichert möglich (weder signiert noch verschlüsselt). Der OPC UA-Server der S7-1500 CPU sendet auch bei "None" (Keine Security) sein öffentliches Zertifikat an den Client. Und manche Clients prüfen dieses Zertifikat. Doch der Client ist nicht gezwungen, ein Zertifikat an den Server zu senden. Die Identität des Clients bleibt möglicherweise unbekannt. Jeder OPC UA-Client kann sich dann mit dem Server verbinden, unabhängig von sämtlichen noch folgenden Security-Einstellungen.

Achten Sie bei der Projektierung des OPC UA-Servers darauf, dass nur Security Policies aktiviert sind, die mit dem Schutzkonzept für Ihre Maschine oder Anlage vereinbar sind. Alle anderen Security Policies sind zu deaktivieren.

Empfehlung: Verwenden Sie die Einstellung "Basic256Sha256 - Signieren & Verschlüsseln", bei der der Server nur Sha256-Zertifikate akzeptiert. Die Security Policy "Basic128Rsa15" ist in der Voreinstellung deaktiviert und sollte nicht als Endpunkt verwendet werden. Wählen Sie Endpunkte mit einer höheren Security-Policy.

Weitere Security-Regeln

- Nutzen Sie nur im Ausnahmefall den Endpunkt "None".
- Verwenden Sie nur im Ausnahmefall die "Gast-Authentifizierung" des Benutzers.
- Erlauben Sie nur dann den Zugriff auf PLC-Variablen und DB-Komponenten über OPC UA, wenn es tatsächlich erforderlich ist.
- Nutzen Sie die Listen vertrauenswürdiger Clients in den Einstellungen des S7-1500 OPC UA-Clients, um nur bestimmten Clients Zugriff zu erlauben.

Zertifikate gemäß X.509 der ITU

Bei OPC UA sind Sicherheitsmechanismen in mehreren Schichten integriert. Dabei spielen digitale Zertifikate eine wichtige Rolle. Ein OPC UA-Client kann nur eine gesicherte Verbindung zu einem OPC UA-Server aufbauen, wenn der Server das digitale Zertifikat des Clients akzeptiert und als vertrauenswürdig einstuft.

Siehe Kapitel "OPC UA-Server der S7-1500 konfigurieren".

Außerdem muss auch der Client das Zertifikat des Servers prüfen und ihm vertrauen. Server und Client müssen sich ausweisen und beweisen, dass sie tatsächlich der sind, der sie zu sein behaupten: Sie müssen ihre Identität nachweisen. Die gegenseitige Authentifizierung von Client und Server verhindert zum Beispiel Angriffe durch einen "Man in the Middle".

Angriffe durch "Man in the Middle"

Zwischen Server und Client könnte sich ein "Man in the Middle" befinden, ein Programm, das die Kommunikation zwischen Server und Client abfängt und behauptet, selbst Client oder Server zu sein, und so wichtige Informationen über das S7-Progamm erhält oder Werte in der CPU setzt und damit eine Maschine oder Anlage angreifen kann.

Bei OPC UA werden digitale Zertifikate verwendet, die dem Standard X.509 der International Telecommunication Union (ITU) entsprechen.

Damit lässt sich die Identität eines Programms, eines Rechners oder einer Organisation nachweisen (authentifizieren).

X.509-Zertifikate

Ein X.509-Zertifikat enthält unter anderem die folgenden Informationen:

- Versionsnummer des Zertifikats
- Seriennummer des Zertifikats
- Informationen über den Algorithmus, den die Zertifizierungsstelle zum Signieren des Zertifikats verwendete.
- Name der Zertifizierungsstelle
- Beginn und Ende der Gültigkeit des Zertifikats
- Name des Programms, der Person oder Organisation, für die das Zertifikat von der Zertifizierungsstelle signiert wurde.
- Der öffentliche Schlüssel des Programms, der Person oder Organisation.

Somit verknüpft ein X509-Zertifikat eine Identität (Name eines Programms, einer Person oder einer Organisation) mit dem öffentlichen Schlüssel des Programms, der Person oder Organisation.

Prüfung beim Verbindungsaufbau

Beim Verbindungsaufbau zwischen Client und Server prüfen die Teilnehmer alle Informationen aus dem Zertifikat, die zur Feststellung der Integrität notwendig sind, z. B. Signatur, Gültigkeitsdauer, Applikationsname (URN) und ab Firmware Version V2.5 auch die IP-Adressen des Clients im Client-Zertifikat.

Außerdem wird der Gültigkeitszeitraum geprüft, der im Zertifikat hinterlegt ist. Die CPU-Uhr muss daher gestellt sein und Datum/Uhrzeit müssen sich im Gültigkeitszeitraum befinden, sonst findet keine Kommunikation statt.

Signieren und Verschlüsseln

Damit überprüft werden kann, ob ein Zertifikat manipuliert wurde, werden Zertifikate signiert.

Hier gibt es verschiedene Vorgehensweisen:

- Innerhalb des TIA Portals haben Sie die Möglichkeit, Zertifikate zu erzeugen und zu signieren. Wenn Sie Ihr Projekt geschützt haben und als Benutzer angemeldet sind mit dem Funktionsrecht, Security-Einstellung vornehmen zu dürfen, dann sind auch die globalen Security-Einstellungen nutzbar. Die globalen Security-Einstellungen erlauben Zugriff auf den Zertifikatsmanager und damit auch auf die Zertifizierungsstelle (CA) des TIA Portals.
- Ihnen stehen weitere Möglichkeiten für die Erzeugung und Signierung von Zertifikaten zur Verfügung. Im TIA Portal können Sie Zertifikate in den globalen Zertifikatsmanager importieren.
  - Sie wenden sich an eine Zertifizierungsstelle (CA) und lassen Ihr Zertifikat signieren. In diesem Fall überprüft die Zertifizierungsstelle Ihre Identität und signiert Ihr Zertifikat mit dem privaten Schlüssel der Zertifizierungsstelle. Senden Sie dazu einen CSR (Certificate Signing Request) an die Zertifizierungsstelle. Wie Sie einen CSR mit dem Tool OpenSSL selbst erzeugen, ist hier beschrieben. (Seite 1208)
  - Sie erstellen selbst ein Zertifikat und signieren es.
    Dazu nutzen Sie zum Beispiel das Programm "Opc.Ua.CertificateGenerator" der OPC Foundation. Wie Sie vorgehen, ist hier beschrieben. Oder Sie verwenden OpenSSL: Eine Anleitung finden Sie unter PKI-Schlüsselpaare und Zertifikate selbst erzeugen (Seite 1209).

Exkurs: Zertifikatstypen

- Selbst signiertes Zertifikat:
  Jeder Teilnehmer erzeugt sein eigenes Zertifikat und signiert es. Beispielanwendungen: Statische Konfiguration mit begrenzter Anzahl von Kommunikationsteilnehmern.
  Aus einem selbst signierten Zertifikat können keine neuen Zertifikate abgeleitet werden. Allerdings müssen Sie alle selbst signierten Zertifikate der Partnergeräte in die CPU laden (STOP erforderlich).
- CA-Zertifikat:
  Alle Zertifikate werden von einer Zertifizierungsstelle erstellt und signiert.
  Beispielanwendungen: Dynamisch wachsende Anlagen.
  Sie müssen nur das Zertifikat der Zertifizierungsstelle in die CPU laden. Die Zertifizierungsstelle kann neue Zertifikate erzeugen (Hinzufügen von Partnergeräten ohne STOP der CPU möglich).

Signieren

Durch die Signatur lässt sich die Integrität und Herkunft einer Nachricht nachweisen, wie im Folgenden beschrieben ist.

Beim Signieren bildet der Sender zunächst aus dem Klartext (Klarnachricht) einen Hashwert. Dann verschlüsselt der Sender den Hashwert mit seinem privaten Schlüssel und überträgt schließlich den Klartext zusammen mit dem verschlüsselten Hashwert zum Empfänger. Der Empfänger benötigt zur Überprüfung der Signatur den öffentlichen Schlüssel des Senders (ist im X509-Zertifikat des Senders enthalten). Mit dem öffentlichen Schlüssel des Senders entschlüsselt der Empfänger den erhaltenen Hashwert. Dann bildet der Empfänger selbst aus dem empfangenen Klartext den Hashwert (das Hashverfahren ist im Zertifikat des Senders enthalten). Anschließend vergleicht der Empfänger die beiden Hashwerte:

- Wenn die beiden Hashwerte gleich sind, dann ist die Klarnachricht unverändert beim Empfänger angekommen und wurde nicht manipuliert.
- Wenn die beiden Hashwerte nicht gleich sind, dann ist die Klarnachricht nicht identisch beim Empfänger angekommen. Die Klarnachricht wurde manipuliert oder bei der Übertragung verfälscht.

Verschlüsseln

Durch Verschlüsseln von Daten verhindern Sie, dass Unbefugte Kenntnis vom Inhalt erhalten. X509-Zertifikate werden nicht verschlüsselt; sie sind öffentlich und jedermann kann sie einsehen.

Beim Verschlüsseln verschlüsselt der Sender die Klarnachricht mit dem öffentlichen Schlüssel des Empfängers. Dazu benötigt der Sender das X509-Zertifikat des Empfängers, weil darin der öffentliche Schlüssel des Empfängers enthalten ist. Der Empfänger entschlüsselt die Nachricht mit seinem privaten Schlüssel. Nur der Empfänger kann die Nachricht entschlüsseln. Er allein besitzt den privaten Schlüssel. Deshalb darf der private Schlüssel nie weitergegeben werden.

Secure Channel

OPC UA verwendet die privaten und öffentlichen Schlüssel von Client und Server beim Aufbau einer gesicherten Verbindung, des Secure Channels. Wenn die gesicherte Verbindung aufgebaut ist, dann erzeugen Client und Server einen internen, nur ihnen bekannten Schlüssel, den sie zum Signieren und Verschlüsseln von Nachrichten verwenden. Dieses symmetrische Verfahren (ein gemeinsamer Schlüssel) ist sehr viel schneller als unsymmetrische Verfahren (private und öffentliche Schlüssel).

Zertifikate bei OPC UA

Verwendung von X.509-Zertifikaten bei OPC UA

OPC UA verwendet beim Aufbau einer Verbindung von Client zu Server drei Arten von X.509-Zertifikaten:

- OPC UA Applikations-Zertifikate
  Solche X.509-Zertifikate identifizieren die Software-Instanz, die jeweilige Installation einer Client- oder Server-Software. Beim Attribut "Organisation Name" tragen Sie den Namen des Unternehmens ein, das die Software einsetzt.

  Hinweis

  Der OPC UA-Server der S7-1500 verwendet Applikations-Zertifikate auch bei der Security-Einstellung "None" (Keine Security). Dadurch wird die Kompatibilität zu OPC UA V1.1 und früher gewahrt.

- OPC UA Software-Zertifikate
  Dieses X.509-Zertifikat identifiziert eine konkrete Version der Client- oder Server-Software. Solche Zertifikate enthalten Attribute, die beschreiben, welche Tests diese Version der Software bei der Zertifizierung durch die OPC Foundation (bzw. anerkannte Testlabors) bestanden hat. Beim Attribut "Organisation Name" tragen Sie den Namen des Unternehmens ein, das die Software entwickelt hat oder vertreibt.

  Hinweis

  Bei STEP 7 werden keine Software-Zertifikate unterstützt.

- OPC UA Anwender-Zertifikate
  Dieses X.509-Zertifikat identifiziert den konkreten Anwender, der zum Beispiel vom OPC UA-Server einer S7-1500 CPU Prozessdaten abruft. Dieses Zertifikat ist nicht erforderlich, wenn der Anwender seine Berechtigung mit seinem Passwort nachweisen kann oder ein anonymer Zugang konfiguriert ist.

  Hinweis

  Bei STEP 7 werden keine Anwender-Zertifikate unterstützt.

Diese Zertifikate sind End-Entity-Zertifikate: Sie identifizieren zum Beispiel eine Person, eine Organisation, ein Unternehmen, eine Instanz (Installation) einer Software.

Selbst-signierte Zertifikate erzeugen

Zertifikatsgenerator des Clients verwenden

Viele OPC UA-Client-Applikationen bzw. SDKs sind in eine Beispielanwendung eingebunden, die es Ihnen erlaubt, aus dieser Applikation heraus Zertifikate für den Client zu erzeugen.

Die Beschreibung zur Zertifikatserzeugung finden Sie im Allgemeinen im Kontext zur Beschreibung der OPC UA-Client-Applikation.

Beispiel-Client aus dem Online-Support

Der OPC UA .NET Client für den SIMATIC S7-1500 OPC UA-Server (https://support.industry.siemens.com/cs/ww/de/view/109737901) erstellt während des ersten Programmstarts ein selbst-signiertes Softwarezertifikat der Client-Applikation im Windows Certificate Store. Die Dokumentation zu diesem Beispiel beschreibt die Vorgehensweise zur Handhabung dieser Zertifikate.

Zertifikatsgenerator des TIA Portals verwenden

Wenn Sie einen OPC UA-Client verwenden, der kein Client-Zertifikat erzeugt, können selbst-signierte Zertifikate mit STEP 7 erstellen.

Dazu gehen Sie folgendermaßen vor:

1. In den Eigenschaften der CPU, unter "Schutz & Security > Zertifikatsmanager > Gerätezertifikate" doppelklicken Sie auf "<Neu hinzufügen>"
2. Klicken Sie auf "Hinzufügen".
3. Im Dialog "Neues Zertifikat erzeugen" wählen Sie bei "Verwendungszweck" die Option "OPC UA-Client".
4. Klicken Sie auf "OK".

Im Feld "Alternativer Antragstellername" (Subject Alternative Name) trägt STEP 7 automatisch die URI für das erstellte Zertifikat ein. Bei der programmtechnischen Zertifikatserstellung über den .NET-Stack der OPC Foundation heißt das Feld z. B. "ApplicationUri", bei anderen Tools zur Zertifikatserstellung kann es anders heißen.

Siehe auch

Handling der Client-Zertifikate der S7-1500 CPU (Seite 1306)

PKI-Schlüsselpaare und Zertifikate selbst erzeugen

Dieses Kapitel ist für Sie nur dann relevant, wenn Sie einen OPC UA-Client verwenden wollen, der nicht selbst ein PKI-Schlüsselpaar und ein Client-Zertifikat erzeugen kann. Sie erzeugen in diesem Fall mit OpenSSL einen privaten und öffentlichen Schlüssel, generieren ein X.509-Zertifikat und signieren das Zertifikat selbst.

OpenSSL verwenden

OpenSSL ist ein Tool für Transport Layer Security, das Sie zum Erzeugen von Zertifikaten nutzen können. Sie können auch andere Tools verwenden, z. B. XCA, eine Schlüsselverwaltungssoftware mit grafischer Oberfläche für eine bessere Übersicht von ausgestellten Zertifikaten.

Um mit OpenSSL unter Windows zu arbeiten, gehen Sie folgendermaßen vor:

1. Installieren Sie OpenSSL unter Windows. Wenn Sie eine 64-Bit-Version des Betriebssystems verwenden, dann installieren Sie OpenSSL zum Beispiel im Verzeichnis "C:\OpenSSL-Win64". Sie erhalten OpenSSL-Win64 als Download bei verschiedenen Anbietern für Open Source Software.
2. Legen Sie ein Verzeichnis an, zum Beispiel "C:\demo".
3. Öffnen Sie die Kommandozeile (cmd.exe). Dazu klicken Sie auf "Start" und tragen im Suchfeld "cmd" ein. Klicken Sie in der Ergebnisliste mit der rechten Maustaste auf "cmd.exe" und führen Sie das Programm als Administrator aus. Windows öffnet die Kommandozeile (DOS-Prompt).
4. Wechseln Sie zum Verzeichnis "C:\demo". Dazu geben Sie den folgenden Befehl ein: "cd C:\demo".
5. Setzen Sie die folgenden Umgebungsvariablen:
   – set RANDFILE=c:\demo\.rnd
   – set OPENSSL_CONF=C:\OpenSSL-Win64\bin\openssl.cfg

   Das folgende Bild zeigt die Kommandozeile mit den Befehlen:

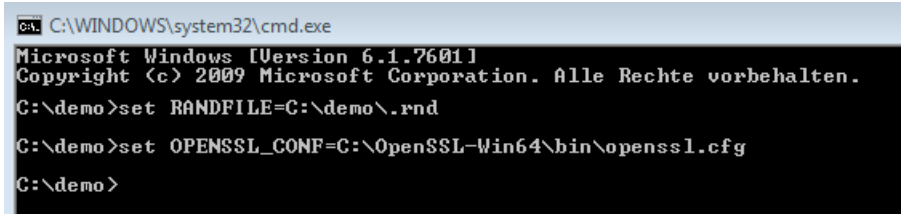

6. Starten Sie nun OpenSSL. Wenn OpenSSL im Verzeichnis C:\OpenSSL-Win64 installiert wurde, dann geben Sie ein: C:\OpenSSL-Win64\bin\openssl.exe. Das folgende Bild zeigt die Kommandozeile mit dem Befehl:

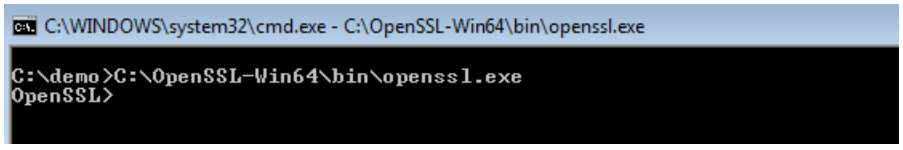

7. Generieren Sie einen privaten Schlüssel. Speichern Sie den Schlüssel in die Datei "myKey.key". Der Schlüssel ist in diesem Beispiel 1024 Bit lang; für eine erhöhte Sicherheit von RSA verwenden Sie in der Praxis 2048 Bit! Geben Sie den folgenden Befehl ein: "genrsa -out myKey.key 2048" ("genrsa -out myKey.key 1024" im Beispiel). Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:

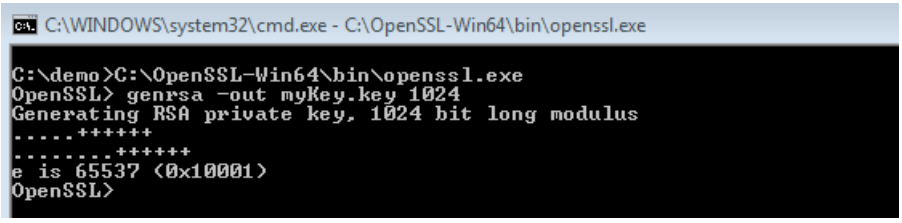

8. Erzeugen Sie einen CSR (Certificate Signing Request), eine Aufforderung, ein Zertifikat zu signieren. Dazu geben Sie den folgenden Befehl ein: "req -new -key myKey.key -out myRequest.csr". Während der Ausführung dieses Befehls fragt OpenSSL Sie nach Angaben zu Ihrem Zertifikat:
   - Country Name: z. B. "DE", für Deutschland, "FR" für Frankreich
   - State or Province Name: z. B. "Bayern".
   - Location Name: z. B. "Augsburg".
   - Organisation Name: Tragen Sie den Namen Ihres Unternehmens ein.
   - Organisational Unit Name: z. B. "IT"
   - Common Name: z. B. "OPC UA Client der Maschine A"
   - Email Address:

Wichtig: Im Feld "Alternativer Antragstellername" (Subject Alternative Name) des erstellten Zertifikats muss die IP-Adresse des Client-Programms hinterlegt sein, sonst akzeptiert die CPU das Zertifikat nicht.

Ihre Angaben werden in das Zertifikat eingefügt. Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:

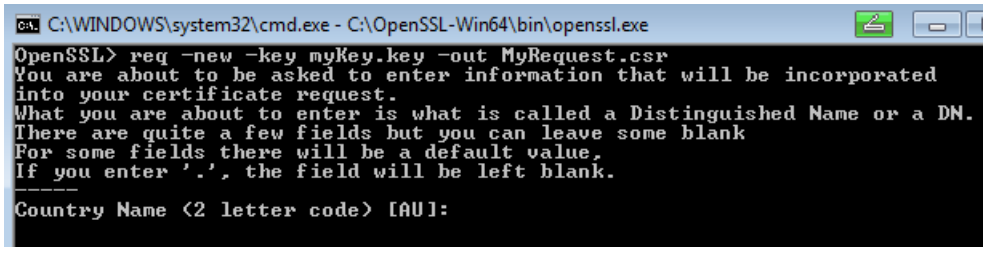

Der Befehl legt im Verzeichnis C:\demo eine Datei an, die den Certificate Signing Request (CSR) enthält, im Beispiel "myRequest.csr".

Verwendung des CSR

Sie können einen CSR auf zwei Arten verwenden:

- Sie senden den CSR an eine Zertifizierungsstelle (CA): Beachten Sie dabei die Hinweise der jeweiligen Zertifizierungsstelle. Die Zertifizierungsstelle (CA) überprüft Ihre Angaben und Identität (Authentifizierung) und signiert das Zertifikat mit dem privaten Schlüssel der Zertifizierungsstelle. Sie erhalten das signierte X.509-Zertifikat und verwenden dieses Zertifikat zum Beispiel für OPC UA, HTTPS oder Secure OUC (secure open user communication). Ihre Kommunikationspartner überprüfen durch den öffentlichen Schlüssel der Zertifizierungsstelle, ob Ihr Zertifikat wirklich von dieser Stelle herausgegeben und signiert wurde (d. h. die Zertifizierungsstelle Ihre Angaben im Zertifikat bestätigt hat).
- Sie signieren den CSR selbst: Dazu verwenden Sie Ihren privaten Schlüssel. Diese Möglichkeit zeigt der nächste Schritt.

Zertifikat selber signieren

Damit Sie Ihr Zertifikat (Self-Signed Certificate) selber erzeugen und signieren können, geben Sie den folgenden Befehl ein: "x509 -req -days 365 -in myRequest.csr -signkey myKey.key -out myCertificate.crt".

Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:

Der Befehl erzeugt ein X.509-Zertifikat mit den Attributen, die Sie mit dem CSR (im Beispiel "myRequest.csr") übergeben, z. B. mit einer Gültigkeit von einem Jahr (-days 365). Außerdem signiert der Befehl das Zertifikat mit Ihrem privaten Schlüssel (im Beispiel "myKey.key"). Ihre Kommunikationspartner können durch Ihren öffentlichen Schlüssel (in Ihrem Zertifikat enthalten) überprüfen, dass Sie im Besitz des zu diesem öffentlichen Schlüssel gehörenden privaten Schlüssels sind. Damit ist auch ausgeschlossen, dass Ihr öffentlicher Schlüssel von einem Angreifer missbräuchlich verwendet wurde.

Bei Self-Signed Zertifikaten bestätigen Sie selbst, dass Ihre Angaben in Ihrem Zertifikat richtig sind. Es gibt keine unabhängige Stelle, die Ihre Angaben überprüft.

Siehe auch

Handling der Client-Zertifikate der S7-1500 CPU (Seite 1306)

Nachrichten gesichert übertragen

Aufbau sicherer Verbindungen bei OPC UA

OPC UA verwendet sichere Verbindungen zwischen Client und Server. Dabei überprüft OPC UA die Identität der Kommunikationspartner. Für die Authentifizierung von Client und Server nutzt OPC UA Zertifikate gemäß X.509-V3 der ITU (International Telecommunication Union). Ausnahme: Bei der Security Policy "Keine Security" wird keine sichere Verbindung aufgebaut.

Message Security Modus

OPC UA verwendet die folgenden Security Policies zum Schutz von Nachrichten:

- Keine Security
  Alle Nachrichten sind ungesichert. Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem None-Endpunkt eines Servers auf.
- Signieren
  Alle Nachrichten werden signiert. Dadurch lässt sich die Integrität der empfangenen Nachrichten überprüfen. Manipulationen werden erkannt. Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem Sign-Endpunkt eines Servers auf.
- Signieren & Verschlüsseln
  Alle Nachrichten werden signiert und verschlüsselt. Dadurch lässt sich die Integrität der empfangenen Nachrichten überprüfen. Manipulationen werden erkannt. Außerdem kann kein Angreifer den Inhalt der Nachricht lesen (Schutz der Vertraulichkeit). Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem "Signieren & Verschlüsseln"-Endpunkt eines Servers auf.

Die Security Policies sind zusätzlich nach den verwendeten Algorithmen benannt. Beispiel: "Basic256Sha256 - Signieren & Verschlüsseln" bedeutet: Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.

Erforderliche Schichten

Das folgende Bild zeigt die drei Schichten Transport-Schicht, Secure Channel und Session, die für den Aufbau einer Verbindung stets erforderlich sind.

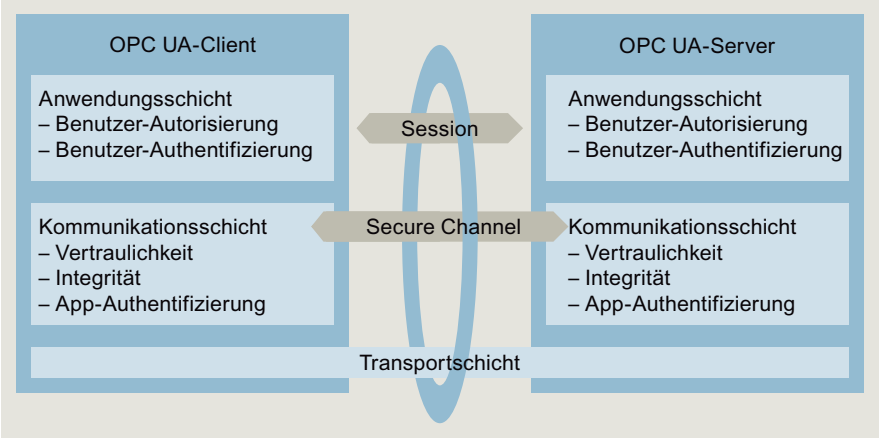

Bild 1-23 Erforderliche Schichten Transport-Schicht, Secure Channel und Session

- Transportschicht:
  Diese Schicht sendet und empfängt Nachrichten. Dazu verwendet OPC UA ein optimiertes TCP-basiertes Binärprotokoll. Die Transportschicht ist die Grundlage für den nachfolgenden Secure Channel.
- Secure Channel
  Der Secure Channel erhält von der Transportschicht die empfangenen Daten und leitet sie an die Session weiter. Daten von der Session, die gesendet werden sollen, leitet der Secure Channel an die Transportschicht weiter.
  Beim Security-Modus "Signieren" signiert der Secure Channel Daten (Nachrichten), die gesendet werden. Bei ankommenden Nachrichten überprüft der Secure Channel die Signatur, um Manipulationen erkennen zu können
  Bei einer "Signieren & Verschlüsseln" Security Policy signiert und verschlüsselt der Secure Channel die Sendedaten. Empfangene Daten entschlüsselt der Secure Channel. Anschließend überprüft der Secure Channel die Signatur.
  Bei der Security Policy "Keine Security" passieren die Nachrichtenpakete den Secure Channel unverändert (die Nachrichten werden im Klartext empfangen und gesendet).
- Session
  Die Session gibt die Nachrichten vom Secure Channel an die Anwendung weiter, bzw. erhält von der Anwendung Nachrichten, die gesendet werden sollen. Die Anwendung verwendet die Prozesswerte, bzw. stellt die Werte bereit.

Aufbau des Secure Channels

Der Aufbau des Secure Channels läuft folgendermaßen ab:

1. Der Server beginnt mit dem Aufbau des Secure Channels, wenn er eine Aufforderung dazu vom Client erhält. Dieser Request ist entweder signiert, signiert und verschlüsselt, oder die Nachricht wird im Klartext gesendet (Security-Modus des gewählten Server-Endpunkts). Bei "Signieren" und "Signieren & Verschlüsseln" sendet der Client ein "Geheimnis" (eine Zufallszahl) mit dem Request.
2. Der Server validiert das Client-Zertifikat (unverschlüsselt im Request enthalten) und überprüft die Identität des Clients. Vertraut der Server dem Client-Zertifikat, dann
   - entschlüsselt der Server die Nachricht und überprüft die Signatur ("Signieren & Verschlüsseln")
   - oder überprüft nur die Signatur ("Signieren")
   - oder lässt die Nachricht unverändert ("Keine Security").
3. Danach sendet der Server eine Antwort zum Client (gleichermaßen gesichert wie der Request). Im Response ist das Server-Geheimnis enthalten. Aus dem Client- und Server-Geheimnis errechnen Client und Server einen symmetrischen Schlüssel. Damit ist der Secure Channel aufgebaut.

Der symmetrische Schlüssel wird nun für das Signieren und Verschlüsseln von Nachrichten verwendet (anstelle der privaten und öffentlichen Schlüssel von Client und Server).

Aufbau der Session

Der Aufbau der Session läuft folgendermaßen ab:

1. Der Client startet den Session-Aufbau, indem er einen CreateSessionRequest an den Server schickt. Diese Nachricht enthält ein Nonce, eine nur einmal verwendete Zufallszahl. Der Server muss diese Zufallszahl (Nonce) signieren, um zu beweisen, dass er Inhaber des privaten Schlüssels ist. Der private Schlüssel gehört zu dem Zertifikat, den der Server beim Aufbau des Secure Channels verwendet. Diese Nachricht (und alle nachfolgenden) ist entsprechend den Sicherheitseinstellungen des gewählten Server-Endpunkts (ausgewählte Security Policies) gesichert.
2. Der Server antwortet mit der CreateSession Response. Diese Nachricht enthält den öffentlichen Schlüssel des Servers sowie das signierte Nonce. Der Client überprüft das signierte Nonce.
3. Wenn der Server den Test bestanden hat, dann sendet der Client einen SessionActivateRequest an den Server. Diese Nachricht enthält die Angaben, die für die Legitimierung des Anwenders erforderlich sind:
   - entweder Username und Passwort
   - oder das X.509-Zertifikat des Anwenders (in STEP 7 V15 nicht unterstützt)
   - oder keine Daten (wenn ein anonymer Zugang konfiguriert ist).
4. Wenn der Anwender über die notwendigen Rechte verfügt, sendet der Server eine Nachricht an den Client zurück (ActivateSessionResponse). Damit ist die Session aktiviert.

Die sichere Verbindung zwischen OPC UA-Client und -Server ist aufgebaut.

Aufbau einer Verbindung mit PLCopen Funktionsbausteinen

Die PLCopen Spezifikation hat eine Reihe von IEC 61131 Funktionsbausteinen für OPC UA-Clients definiert. Die Anweisung UA_Connect initiiert hierbei sowohl einen Secure Channel als auch eine Session nach oben beschriebenem Muster.

S7-1500 CPU als OPC UA-Server nutzen

Wissenswertes zum OPC UA-Server der S7-1500 CPU

Der OPC UA-Server der S7-1500 CPUs

Die S7-1500 CPUs ab Firmware V2.0 sind mit einem OPC UA-Server ausgestattet. Dies betrifft neben den Standard-S7-1500 CPUs auch die Varianten S7-1500F, S7-1500T, S7-1500C, S7-1500pro CPUs, ET 200SP CPUs, SIMATIC S7-1500 SW Controller und PLCSIM Advanced.

Konvention: Mit "S7-1500 CPUs" sind auch die oben genannten CPU-Varianten gemeint.

Grundlagen zum OPC UA-Server der S7-1500 CPU

Der Zugriff auf den OPC UA-Server der CPU ist über alle integrierten Ethernet-Schnittstellen der S7-1500 CPU möglich.

Über CP oder CM ist kein direkter Zugriff über den Rückwandbus des Automatisierungssystems auf den OPC UA-Server der CPU möglich.

Für den Zugriff durch Clients speichert der Server die freigegebenen PLC-Variablen und andere Informationen in Form von Knoten ab (siehe AUTOHOTSPOT). Diese Knoten sind miteinander verbunden und bilden ein Netzwerk. OPC UA definiert Einstiegspunkte in dieses Netzwerk (Well-known Nodes), die das Navigieren zu unterlagerten Knoten ermöglichen.

Mit einem OPC UA-Client können Sie Werte von Variablen des SPS-Programms lesen, beobachten oder schreiben sowie Methoden aufrufen, die der Server zur Verfügung stellt. Ab Firmware Version 2.5 können Sie Methoden implementieren, siehe Wissensertes zu Server-Methoden (Seite 1251).

Knotenklassen

OPC UA-Server stellen Informationen in Form von Knoten (Nodes) zur Verfügung. Ein Knoten kann zum Beispiel ein Objekt, eine Variable, eine Methode oder eine Property sein.

Das folgende Beispiel zeigt den Adressraum des OPC UA-Servers einer S7-1500 CPU (Ausschnitt aus dem OPC UA-Client "UaExpert" von Unified Automation).

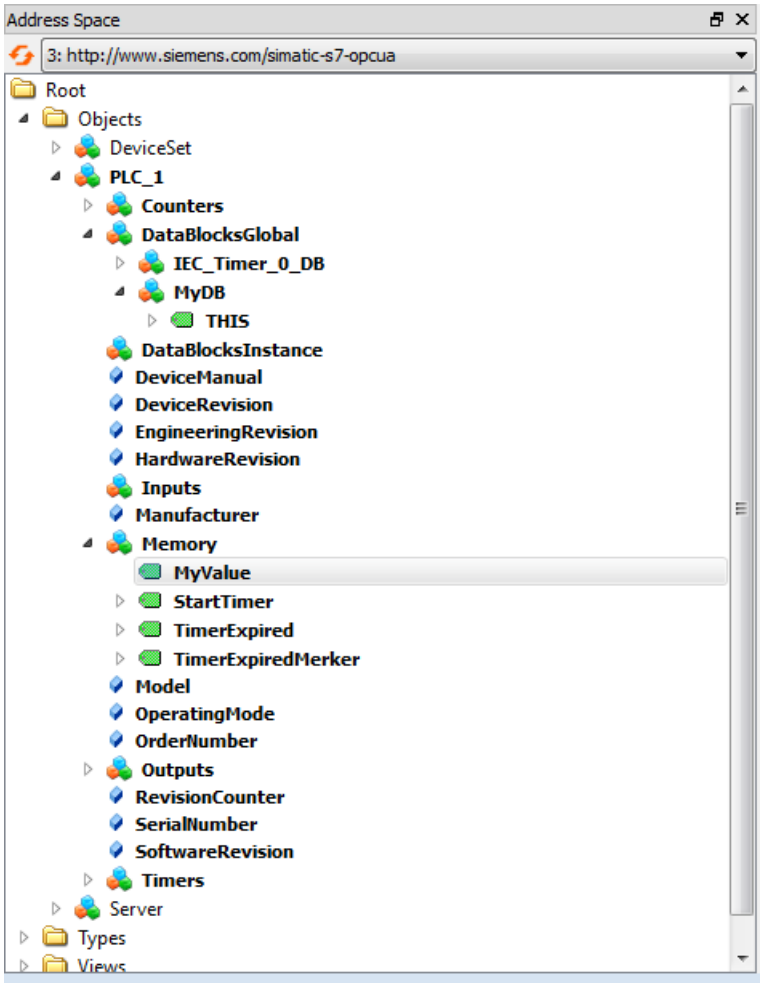

Bild 1-24 Beispiel für den Adressraum des OPC UA-Servers einer S7-1500 CPU

Im Bild oben ist die Variable "MyValue" markiert (grau unterlegt).

Diese Variable befindet sich unterhalb des Knotens "Memory", der die Knotenklasse "Object" besitzt.

"Memory" wiederum befindet sich unterhalb des Knotens "PLC_1" (ebenfalls ein Object).

Adressraum

Die Knoten sind untereinander über Referenzen verbunden, zum Beispiel über die Referenz "HasComponent", die eine hierarchische Beziehung zwischen einem Knoten und seinen unterlagerten Knoten wiedergibt. Über ihre Referenzen bilden die Knoten ein Netzwerk, das zum Beispiel die Form eines Baums besitzen kann.

Ein Netzwerk aus Knoten wird auch als Adressraum bezeichnet. Von der Wurzel ausgehend sind alle Knoten im Adressraum erreichbar.

Endpunkte der OPC UA-Server

Die Endpunkte der OPC UA-Server definieren die Sicherheitsstufe für eine Verbindung. Je nach Einsatzzweck oder gewünschter Sicherheitsstufe müssen Sie am Endpunkt die entsprechenden Einstellungen für die Verbindung vornehmen.

Verschiedene Security-Einstellungen

Vor dem Aufbau einer gesicherten Verbindung erfragen OPC UA-Clients beim Server, mit welchen Security-Einstellungen Verbindungen möglich sind. Der Server sendet eine Liste zurück mit allen Security-Einstellungen (Endpunkten), die der Server anbietet.

Aufbau von Endpunkten

Endpunkte bestehen aus den folgenden Komponenten:

- Kennung für OPC: "opc.tcp"
- IP-Adresse: 192.168.178.151 (in dem Beispiel)
- Port-Nummer für OPC UA: 4840 (Standard-Port)
  Die Port-Nummer ist konfigurierbar.
- Security-Einstellung für Nachrichten (Message Security-Modus): None, Sign, SignAndEncrypt.
- Verschlüsselungs- und Hash-Verfahren (Security Policy): None, Basic128Rsa15, Basic256, Basic256Sha256 (in dem Beispiel).

Das folgende Bild zeigt das Programm "UA Sample Client" der OPC Foundation.

Der Client hat eine gesicherte Verbindung zum OPC UA-Server einer S7-1500 CPU aufgebaut, zum Endpunkt "opc.tcp://192.168.178.151:4840 - [SignAndEncrypt: Basic128Rsa15:Binary]". Die Security-Einstellungen "SignAndEncrypt:Basic128Rsa15" sind im Endpunkt enthalten.

Hinweis

Endpunkt mit möglichst hoher Security Policy auswählen

Wählen Sie für die Endpunkte eine für die Anwendung angemessene hohe Security Policy aus und deaktivieren Sie am OPC UA-Server die Security Policy mit geringerer Security Policy.

Für die sichersten Endpunkte (Basic256Sha256) des OPC UA-Servers der S7-1500 CPU ist ein Sha256-Zertifikat notwendig.

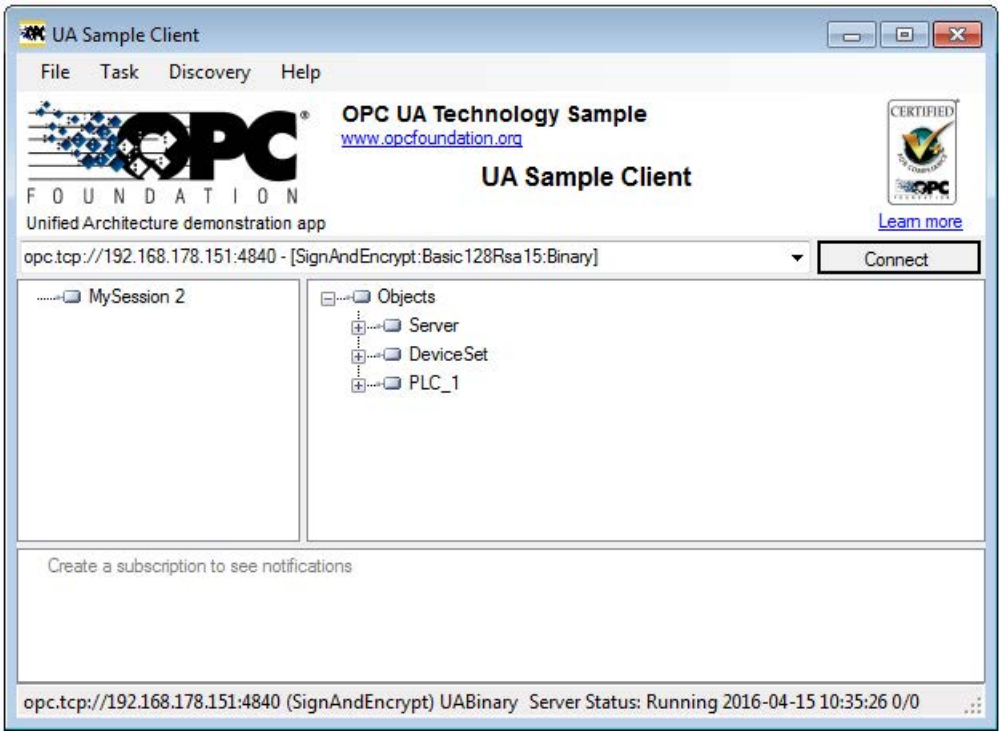

Bild 1-25 Programm "UA Sample Client" der OPC Foundation

Der Aufbau einer Verbindung zu einem Endpunkt des Servers kommt nur zu Stande, wenn der OPC UA-Client die geforderten Sicherheitseinstellungen dieses Endpunkts erfüllt.

Durch den OPC UA-Server bereitgestellte Informationen

OPC UA-Server stellen zahlreiche Informationen bereit:

- Die Werte von PLC-Variablen und DB-Komponenten, auf die Clients zugreifen dürfen.
- Die Datentypen dieser PLC-Variablen und DB-Komponenten.
- Angaben zum OPC UA-Server selbst und zur CPU.

Dadurch können sich Clients einen Überblick verschaffen und gezielt Informationen auslesen. Ein vorhergehendes Wissen über das SPS-Programm und den Datenhaushalt der CPU ist nicht erforderlich. Es ist nicht nötig, beim Entwickler des SPS-Programms nachzufragen, wenn PLC-Variablen gelesen werden sollen. Im Server selbst sind alle erforderlichen Angaben gespeichert (zum Beispiel die Datentypen der PLC-Variablen).

Anzeige der Informationen des OPC UA-Servers

Sie haben folgende Möglichkeiten:

- Online: Sie lassen sich zur Laufzeit des OPC UA-Servers alle verfügbaren Informationen anzeigen. Navigieren (Browsen) Sie dazu im Adressraum des Servers.
- Offline: Sie exportieren eine XML-Datei, die auf den XML-Schemata der OPC Foundation basiert.
  Selbst erstellte Server-Methoden (FB-Instanz, die von einem OPC UA-Client aufgerufen werden kann) werden ab STEP 7 V15.1 mit exportiert, siehe AUTOHOTSPOT.
- Offline mit dem Openness-API: Sie verwenden in Ihrem Programm das API (Application Programming Interface) des TIA Portals, um die Funktion zum Export aller von OPC UA lesbaren PLC-Variablen aufzurufen. Dafür ist .NET Framework 4.0 erforderlich, siehe TIA Portal Openness, SIMATIC Projekte über Skripte automatisieren (https://support.industry.siemens.com/cs/ww/de/view/109477163).
- Wenn Sie bereits die Syntax und das SPS-Programm kennen, können Sie ohne vorhergehende Recherche auf den OPC UA-Server zugreifen.

Verhalten des OPC UA-Servers im Betrieb

Der OPC UA-Server im Betrieb

Der OPC UA-Server der S7-1500 CPU startet, wenn Sie den Server aktivieren und das Projekt in die CPU laden.

Wie Sie den OPC UA-Server aktivieren, ist hier beschrieben.

Verhalten bei STOP der CPU

Ein aktivierter OPC UA-Server bleibt im Betrieb, auch wenn die CPU in den Betriebszustand "STOP" wechselt. Der OPC UA-Server antwortet dann nach wie vor auf Anfragen von OPC UA-Clients.

Das Verhalten des Servers im Einzelnen:

- Wenn Sie die Werte von PLC-Variablen abfragen, dann erhalten Sie die Werte, die aktuell waren, bevor die CPU in den Betriebszustand "STOP" wechselte oder gesetzt wurde.
- Wenn Sie Werte zum OPC UA-Server schreiben, dann übernimmt der OPC UA-Server diese Werte.
  Aber die CPU verarbeitet die Werte nicht, weil das Anwenderprogramm im Betriebszustand "STOP" nicht ausgeführt wird.
  Jedoch kann ein OPC UA-Client die bei STOP geschriebenen Werte aus dem OPC UA-Server der CPU lesen.
  Bei Wiederanlauf überschreibt die CPU die bei STOP geschriebenen Werte mit den Startwerten der PLC-Variablen.
- Wenn Sie eine Server-Methode aufrufen, dann erhalten Sie die Fehlermeldung 16#00AF_0000 (BadInvalidState), da die Server-Methode (Anwenderprogramm) nicht ausgeführt wird.

Neustart des Servers

Der OPC UA-Server wird bei jedem Laden in die CPU gestoppt (z. B. nach dem Laden einer Konfiguration oder eines Bausteins) und startet anschließend neu. Die Dauer des Neustarts ist abhängig vom Umfang der Datenstruktur.

Betriebszustand der CPU über OPC UA-Server auslesen

Der OPC UA-Server erlaubt Ihnen, den Betriebszustand der CPU auszulesen, siehe folgendes Bild:

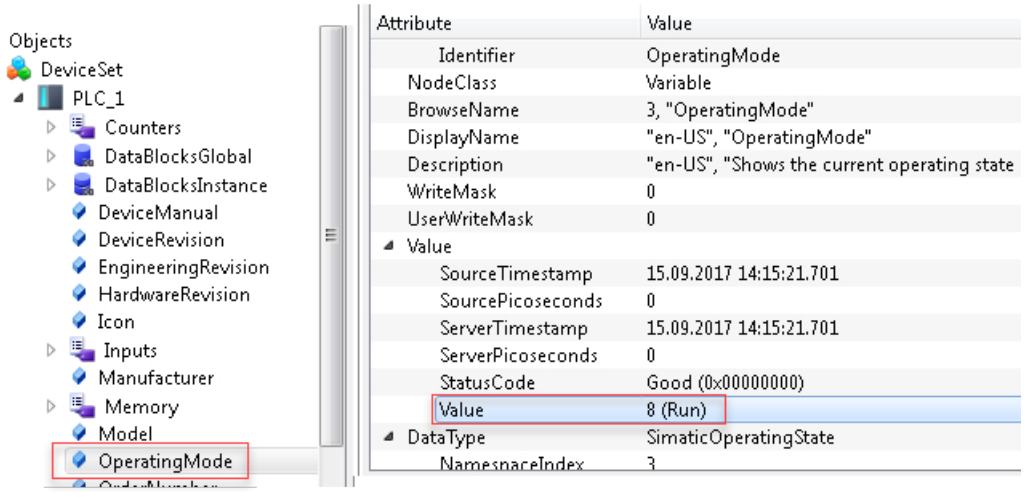

Bild 1-26 Betriebszustand der CPU über OPC UA-Server auslesen

Neben dem Betriebszustand der CPU können Sie z. B. den Diagnosestatus der CPU (hier Good) auslesen.

Diagnose des OPC UA-Servers

Online-Diagnose des OPC UA-Servers

Den OPC UA-Server der S7-1500 CPU können Sie mit gängigen OPC UA-Clients, z. B. UaExpert, online diagnostizieren.

Die Diagnoseinformationen sind in folgende Bereiche aufgeteilt:

- Server Diagnostics
- Sessions Diagnostics
- Subscriptions Diagnostics

Im Adressraum des Servers sind z. B. folgende Knoten mit Diagnoseinformationen vorhanden:

- **ServerDiagnosticsSummary**: Server Diagnose-Zusammenfassung
  - CurrentSessionCount: Anzahl der aktiven Sessions
- **SessionsDiagnosticsSummary**: Session Diagnose-Zusammenfassung
  - SessionTimeout: Eingestellte Zeit, die eine Session z. B. bei Verbindungsabbruch überdauert
  - SecurityRejectedSessionCount: Anzahl der Sessions, die wegen nicht zusammenpassender Endpunkt-Security-Einstellungen zwischen Client und Server zurückgewiesen wurden
- **SubscriptionsDiagnosticsArray**: Array mit je einem Element pro Subscription für die jeweilige Session

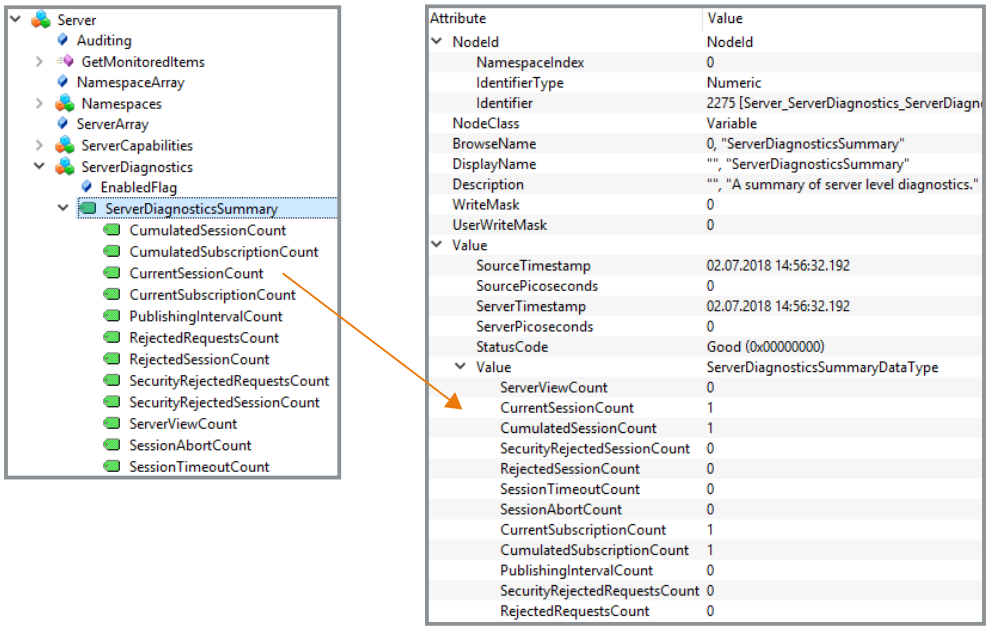

Der Knoten SessionsDiagnosticsSummary zeigt auch die Eingeschaften der Client-Applikation, die innerhalb der Session auf den Server zugreift.

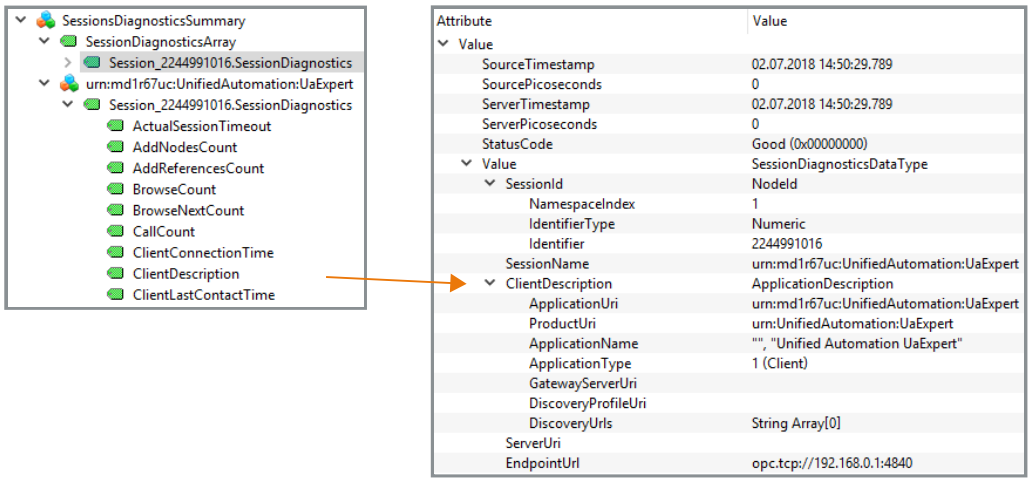

Diagnose der Verbindung zwischen Client und Server

Um den Status der Verbindung während der Laufzeit des Programms im Client zu diagnostizieren, verwenden Sie die folgende Anweisung:

OPC_UA_ConnectionGetStatus: Verbindungsstatus lesen.

Zugriff auf PLC-Variablen projektieren

Schreib- und Leserechte verwalten

PLC-Variablen und DB-Variablen für OPC UA frei geben

OPC UA-Clients können auf PLC-Variablen und DB-Variablen lesend und schreibend zugreifen, wenn die Variablen für OPC UA freigegeben sind (Voreinstellung). Bei freigegebenen Variablen ist das Optionskästchen bei "Erreichbar aus HMI/OPC UA" aktiviert.

Das folgende Beispiel zeigt einen Array-Datenbaustein:

MyDB

Name	Datentyp	Erreichbar aus HMI/OPC UA	Schreibbar aus HMI/OPC UA	Sichtbar in HMI Engineering
MyDB	Arr...	☑	☑	☑
MyDB[0]	Int	☑	☑	☑
MyDB[1]	Int	☑	☑	☑
MyDB[2]	Int	☑	☑	☑
MyDB[3]	Int	☑	☑	☑
MyDB[4]	Int	☑	☑	☑
MyDB[5]	Int	☑	☑	☑
MyDB[6]	Int	☑	☑	☑
MyDB[7]	Int	☑	☑	☑
MyDB[8]	Int	☑	☑	☑
MyDB[9]	Int	☑	☑	☑

Bild 1-27 PLC-Variablen und DB-Variablen für OPC UA frei geben

Das Array kann von OPC UA-Clients komplett in einem Zug gelesen werden (siehe Adressierung von Knoten (Seite 1195)). Bei allen Komponenten des Arrays sind die Optionskästchen "Erreichbar aus HMI/OPC UA" und "Schreibbar aus HMI/OPC UA" aktiviert.

Folge: OPC UA-Clients dürfen diese Komponenten sowohl lesen als auch schreiben.

Schreibrechte entziehen

Wenn Sie eine Variable gegen Schreibzugriffe schützen wollen, dann deaktivieren Sie bei dieser Variablen die Option "Schreibbar aus HMI/OPC UA". Dadurch entziehen Sie OPC UA-Clients und HMI-Geräten das Schreibrecht.

Folge: Ausschließlich lesende Zugriffe durch OPC UA-Clients und durch HMI-Geräte sind möglich. OPC UA-Clients können dieser Variable keine Werte zuweisen und dadurch keinen Einfluss auf den Ablauf des S7-Programms nehmen.

Schreib- und Leserechte entziehen

Um eine Variable gegen Schreib- und Lesezugriffe zu schützen, deaktivieren Sie bei dieser Variablen die Option "Erreichbar aus HMI/OPC UA" (Häkchen nicht gesetzt). Dadurch entfernt der OPC UA-Server diese Variable aus seinem Adressraum. OPC UA-Clients sehen diese CPU-Variable nicht mehr.

Folge: OPC UA-Clients und HMI-Geräte können diese Variable weder lesen noch schreiben.

Schreib- und Leserechte von Strukturen

Wenn Sie für eine Komponente einer Struktur das Schreib- oder Leserecht entziehen, dann kann die Struktur oder der Datenbaustein nicht mehr als Ganzes beschrieben oder gelesen werden.

Wenn Sie Schreib- und Leserechte einzelnen Komponenten eines PLC-Datentyps (UDT) entziehen, dann sind die Rechte auch in einem auf dem UDT basierenden Datenbaustein entzogen!

Sichtbar in HMI Engineering

Die Option "Sichtbar in HMI Engineering" bezieht sich auf Engineering-Tools von Siemens. Wenn Sie die Option "Sichtbar in HMI Engineering" deaktivieren (Häkchen nicht gesetzt), dann können Sie die Variable nicht mehr in WinCC (TIA Portal) projektieren.

Die Option hat keine Auswirkungen auf OPC UA.

Regeln

- Erlauben Sie in STEP 7 nur dann lesende Zugriffe auf PLC-Variablen und Variablen von Datenbausteinen, wenn es für die Kommunikation zu anderen Systemen (Steuerungen, eingebetteten Systemen, MES) erforderlich ist.
  Andere PLC-Variablen sollten Sie nicht frei geben.
- Gewähren Sie nur dann schreibende Zugriffe über OPC UA, wenn Schreibrechte tatsächlich bei bestimmten PLC-Variablen und Variablen von Datenbausteinen erforderlich sind.
- Wenn Sie für alle Elemente eines Datenbausteins die Option "Erreichbar aus HMI/OPC UA" zurückgesetzt haben, dann ist der Datenbaustein für einen OPC UA-Client im Adressraum des OPC UA-Servers der S7-1500 CPU nicht mehr sichtbar.
- Sie können auch zentral den Zugriff auf einen gesamten Datenbaustein verhindern (siehe Schreib- und Leserechte für kompletten DB verwalten (Seite 1224)). Diese Einstellung "überstimmt" die Einstellungen an den Komponenten im DB-Editor.

Schreib- und Leserechte für kompletten DB verwalten

DBs oder DB-Inhalte für OPC UA-Clients verbergen

Ab STEP 7 V15 haben Sie die Möglichkeit, den Zugriff auf einen kompletten Datenbaustein durch einen OPC UA-Client auf einfache Weise zu verhindern.

Auf diese Weise bleiben die Daten des entsprechenden DBs, auch Instanz-DBs von Funktionsbausteinen, für OPC UA-Clients verborgen.

Voreingestellt ist, dass Datenbausteine von OPC UA-Clients lesbar und schreibbar sind.

Vorgehen

Um einen Datenbaustein für OPC UA-Clients komplett zu verbergen bzw. um einen Datenbaustein vor Schreibzugriffen von OPC UA-Clients zu schützen, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Projektnavigation den zu schützenden Datenbaustein.
2. Wählen Sie das Kontextmenü "Eigenschaften".

3. Wählen Sie den Bereich "Attribute".
4. Aktivieren/Deaktivieren Sie die Optionskästchen "DB erreichbar aus OPC UA" nach Ihren Erfordernissen.

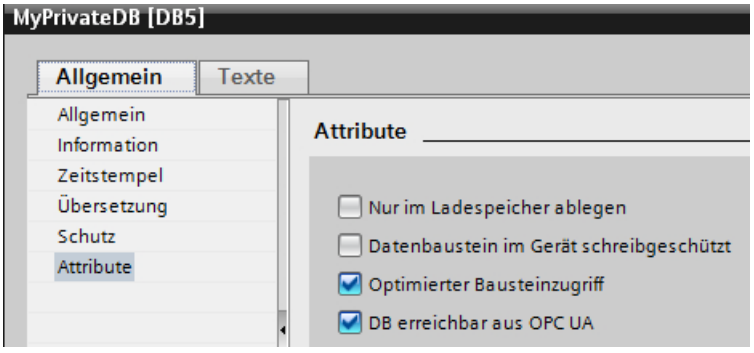

Bild 1-28 DBs oder DB-Inhalte für OPC UA-Clients verbergen

Hinweis

Beeinflussung der Einstellungen im DB-Editor

Wenn Sie über das hier beschriebene DB-Attribut einen DB verbergen, dann sind die Einstellungen an den Komponenten im DB-Editor nicht mehr relevant; einzelne Komponenten können nicht mehr erreicht oder beschrieben werden.

Zugriffsmöglichkeiten auf Daten des OPC UA-Servers

Hohe Performance abhängig vom Anwendungsfall

OPC UA ist für die Übertragung vieler Daten in kurzer Zeit ausgelegt. Sie können die Leistung deutlich steigern, wenn Sie nicht auf einzelne PLC-Variablen zugreifen, sondern Arrays und Strukturen als Ganzes lesen und schreiben.

Am schnellsten ist der Zugriff auf Arrays. Deshalb sollten Sie die Daten für OPC UA-Clients in Arrays zusammenfassen.

Empfehlungen für den Zugriff auf den OPC UA-Server durch den OPC UA-Client

- Nutzen Sie für den einmaligen oder seltenen Datenzugriff den normalen Read/Write-Zugriff.
- Nutzen Sie für den zyklischen Zugriff auf wenige Daten (bis ca. alle 5 Sekunden) Subscriptions.
  Optimieren Sie am OPC UA-Server die Einstellungen für das kleinste Sendeintervall und das kleinste Abtastintervall.
- Nutzen Sie für einen regelmäßigen (wiederkehrenden) Zugriff auf bestimmte Variablen die Funktionen "RegisteredRead" und "RegisteredWrite".

Gewähren Sie der CPU mehr Kommunikationslast, indem Sie den Wert für "Zyklusbelastung durch Kommunikation" erhöhen. Vergewissern Sie sich, dass Ihre Anwendung mit den geänderten Einstellungen noch ordnungsgemäß funktioniert.

Vorgehensweise zum Anlegen eines Arrays-DBs

Arrays können sie z. B. in globalen Datenbausteinen, im Instanzdatenbaustein eines Funktionsbausteins oder als Array-DB anlegen. Im Folgenden ist beschrieben, wie Sie einen Array-DB anlegen.

Um einen Datenbaustein mit einem Array (Array-Datenbaustein) anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation die CPU mit dem OPC UA-Server.
2. Doppelklicken Sie auf "Programmbausteine".
3. Doppelklicken Sie auf "Neuen Baustein hinzufügen".
4. Klicken Sie auf die Schaltfläche "Datenbaustein".
5. Wählen Sie einen eindeutigen Namen für den Datenbaustein oder übernehmen Sie den bereits eingetragenen Namen.
6. Wählen Sie in der Klappliste bei "Typ" den Eintrag "Array-DB".
7. Wählen Sie in der Klappliste bei "Array-Datentyp" den Datentyp für die einzelnen Komponenten des Arrays.
8. Tragen Sie die obere Grenze des Arrays bei "Array-Grenze" ein.
9. Klicken Sie auf die Schaltfläche "OK".

OPC UA-XML-Datei exportieren

OPC UA-Exportdatei erzeugen

OPC UA spezifiziert ein Informationsmodell-XML-Schema mit einer Standard-Syntax, mit der Hersteller ihr Informationsmodell von z. B. einem Anlagenteil definieren und in maschinenlesbaren Form zur Verfügung stellen können. Eine Datei, die dieses Schema nutzt, wird auch Nodeset-Datei genannt.

Mit STEP 7 (TIA Portal) können Sie auf einfache Weise das Standard-SIMATIC-Informationsmodell der S7-1500 CPU als Server in eine OPC UA-XML-Datei (Nodeset-Datei) exportieren; inklusive aller PLC-Variablen und Methoden, die Sie für OPC UA freigegeben haben.

Die OPC UA-XML-Datei nutzen Sie für die Offline-Projektierung eines OPC UA-Clients; sie ist gemäß der OPC UA-Spezifikation aufgebaut und fungiert als Standard-SIMATIC-Server-Schnittstelle.

Um die OPC UA-XML-Datei zu erzeugen und zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "Allgemein > OPC UA > Server > Exportieren".
3. Klicken Sie auf die Schaltfläche "OPC UA XML-Datei exportieren":
4. Wählen Sie das Verzeichnis, in dem Sie die Exportdatei speichern wollen.

5. Wählen Sie einen neuen Namen für die Datei. Oder behalten Sie den Namen bei, der bereits eingetragen ist.
6. Klicken Sie auf "Speichern".

Hinweis

Server-Methoden sind ab STEP 7 (TIA Portal) V15.1 in der OPC UA-Exportdatei (Nodeset) mit ihren Ein- und Ausgangsparametern enthalten.

Alle Arrayelemente separat exportieren

Wenn in den CPU-Eigenschaften "OPC UA > Server > Exportieren" die Option "Alle Arrayelemente als separate Knoten exportieren" aktiviert ist, dann enthält die OPC UA XML-Datei alle Elemente von Arrays jeweils als einzelne XML-Elemente. In der XML-Datei sind außerdem die Arrays selbst jeweils in einem XML-Element beschrieben.

Wenn viele Array-Elemente in einem Array enthalten sind, kann die XML-Datei sehr umfangreich werden.

Tipp

Im folgenden FAQ finden Sie einen Konverter, mit dem Sie die Exportdatei in das CSV-Format wandeln können. Sie erhalten damit eine Liste der für OPC UA erreichbaren Variablen der CPU.

Den FAQ finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/109742903).

OPC UA-Server der S7-1500 CPU konfigurieren

OPC UA-Server aktivieren

Voraussetzung

- Sie haben eine Runtime-Lizenz für den Betrieb des OPC UA-Servers erworben, siehe Lizenzen für den OPC UA-Server (Seite 1251).
- Wenn Sie Zertifikate zur gesicherten Kommunikation nutzen z. B. HTTPS, Secure OUC, OPC UA, dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.

OPC UA-Server in Betrieb nehmen

In der Grundeinstellung ist der OPC UA-Server der CPU aus Sicherheitsgründen nicht freigegeben: OPC UA-Clients können weder schreibend noch lesend auf die S7-1500 CPU zugreifen.

Um den OPC UA-Server der CPU zu aktivieren, gehen Sie folgendermaßen vor.

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "OPC UA > Server".
3. Aktivieren Sie den OPC UA-Server der CPU.
4. Bestätigen Sie die Sicherheitshinweise.
5. Wählen Sie bei den CPU-Eigenschaften den Bereich "Runtime-Lizenzen" und stellen die erworbene Runtime-Lizenz für den OPC UA-Server ein.
6. Kompilieren Sie das Projekt.
7. Laden Sie das Projekt in die CPU.
   Der OPC UA-Server der CPU startet nun.

Einstellungen bleiben gespeichert

Falls Sie den Server bereits aktiviert und Einstellungen vorgenommen hatten, dann gehen diese Einstellungen nicht verloren, wenn Sie den Server deaktivieren. Die Einstellungen sind nach wie vor gespeichert und stehen wieder zur Verfügung, wenn Sie den Server wieder aktivieren.

Applikationsname

Der Applikationsname ist der Name der OPC UA-Applikation und gilt für den Server und den Client. Der Name wird angezeigt unter "OPC UA > Allgemein":

- Die Voreinstellung für den Applikationsnamen lautet: "SIMATIC.S7-1500.OPC-UA.Application:PLC_1".
- Die Voreinstellung setzt sich aus "SIMATIC.S7-1500.OPC-UA.Application:" und dem Namen der CPU zusammen, wie er unter "Allgemein > Produktinformation > Name" gewählt wurde, hier "PLC_1".
- Über diesen Applikationsnamen identifiziert sich der OPC UA-Server gegenüber einem Kommunikationspartner (OPC UA Client), z. B. wenn ein OPC UA-Client den Discovery-Service nutzt, um erreichbare Server zu ermitteln.
- Den angezeigten Applikationsnamen verwendet der OPC UA-Client der CPU beim Verbindungsaufbau zu einem OPC UA-Server. D. h. die CPU trägt diesen Applikationsnamen automatisch als "ApplicationName" für die Anweisung "OPC_UA_Connect" ein (Variable vom Typ "OPC_UA_SessionConnectInfo" am Parameter "SessionConnectInfo" der Anweisung "OPC_UA_Connect").
  Daher müssen Sie beim Programmieren der Anweisung "OPC_UA_Connect" den "ApplicationName" mit einem Leerstring belegen. Mit dem Applikationsnamen können Sie z. B. den Client mit seinen Sessions (SessionNames) zu Diagnosezwecken identifizieren.

Wenn Sie den Server aktiviert haben, können Sie auch einen anderen Namen verwenden, der in Ihrem Projekt aussagekräftig ist und der die in ihrem Projekt geltenden Anforderungen z. B. nach weltweiter Eindeutigkeit erfüllt..

Das folgende Beispiel stammt von UaExpert:

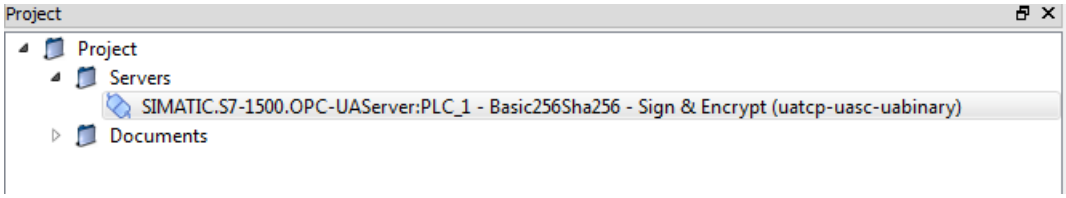

Applikationsname ändern

Um den Applikationsnamen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "OPC UA > Allgemein"
3. Tragen Sie einen aussagekräftigen Namen ein.

Beachten Sie, dass der Applikationsname auch im Zertifikat eingetragen ist (Subject Alternative Name) und Sie nach Änderung des Applikationsnamens gegebenenfalls ein zuvor erstelltes Zertifikat nochmals erstellen müssen.

Zugang zum OPC UA-Server

Server-Adressen

Der OPC UA-Server der S7-1500 CPU ist über alle internen PROFINET-Schnittstellen der CPU (ab Firmware V2.0) erreichbar, jedoch nicht über die PROFINET-Schnittstellen von CP/CM.

Bei SIMATIC S7-1500 SW Controllern ist der OPC UA-Server nur über die PROFINET-Schnittstellen erreichbar, die der Software-PLC zugewiesen sind.

Im Beispiel können über die folgenden URLs (Uniform Resource Locator) Verbindungen zu dem OPC UA-Server der CPU aufgebaut werden:

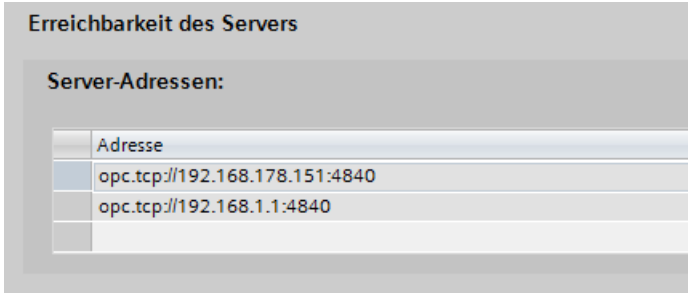

Die URLs gliedern sich folgendermaßen:

- Protokollkennung "opc.tcp://"
- IP-Adresse
  - 192.168.178.151
    Die IP-Adresse, über die OPC UA-Server aus dem Ethernet-Subnetz 192.168.178 erreichbar ist.
  - 192.168.1.1
    Die IP-Adresse, über die OPC UA-Server aus dem Ethernet-Subnetz 192.168.1 erreichbar ist.
- TCP-Portnummer
  - Voreinstellung: 4840 (Standardport)
    Die Portnummer kann geändert werden, unter "OPC UA > Server > Einstellungen > Port".

Dynamische IP-Adressen

Im folgenden Beispiel ist die IP-Adresse der PROFINET-Schnittstelle [X2] noch nicht festgelegt.

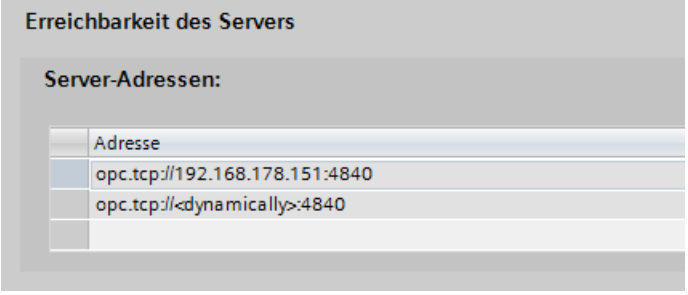

In der Tabelle erscheint der Platzhalter "<dynamically>".

Die IP-Adresse dieser PROFINET-Schnittstelle wird später am Gerät gesetzt, z. B. über das Display der CPU.

Standard-SIMATIC-Server-Schnittstelle aktivieren

Wenn die Option "Standard-SIMATIC-Server-Schnittstelle aktivieren" aktiviert ist, dann stellt der OPC UA-Server der CPU die freigegebenen PLC-Variablen und Server-Methoden den Clients zur Verfügung, wie es in der OPC UA-Spezifikation festgelegt ist.

In der Voreinstellung ist diese Option aktiviert.

Lassen Sie die Option aktiviert, damit OPC UA-Clients die Möglichkeit haben, sich automatisch mit dem OPC UA-Server der CPU zu verbinden und Daten auszutauschen.

Wenn Sie diese Option nicht aktivieren, müssen Sie die Server-Schnittstelle über den Eintrag "OPC UA-Kommunikation" in der Projektnavigation hinzufügen. Diese Schnittstelle wird dann als OPC UA-Server-Schnittstelle verwendet, siehe AUTOHOTSPOT.

Abwärtskompatible Datentyp-Definitionen nach OPC UA-Spezifikation ≤ V1.03

Die OPC UA-Spezifikation (<= V1.03) definiert Mechanismen, um mit Hilfe von TypeDictionaries Datentyp-Definitionen, z. B. für benutzerdefinierte Strukturen (UDTs) von einem Server auslesen zu können.

Sie können in den OPC UA-Server-Eigenschaften der CPU einstellen, ob die CPU diese abwärtskompatiblen Datentyp-Definitionen nach OPC UA-Spezifikation ≤ V1.03 für die Standard-SIMATIC-Server-Schnittstelle erzeugt oder nicht.

Da TypeDictionaries komplex sind und zu großen OPC UA-XML-Dateien (Server-Schnittstellen) führen, die der Client interpretieren muss, wurde mit OPC UA Spezifikation V1.04 eine einfachere Lösung eingeführt (Attribut "DataTypeDefinition" am DataType-Knoten). Wenn Ihr Client die OPC UA-Spezifikation ab V1.04 unterstützt, dann deaktivieren Sie die Option.

Allgemeine Einstellungen des OPC UA-Servers

TCP-Port für OPC UA

Standardmäßig verwendet OPC UA den TCP-Port 4840. Sie können jedoch auch einen anderen Port wählen. Eingaben von 1024 bis 49151 sind möglich. Sie müssen jedoch darauf achten, dass keine Konflikte mit anderen Anwendungen entstehen. Den gewählten Port müssen OPC UA-Clients beim Verbindungsaufbau verwenden.

Im folgenden Beispiel wurde der Port 48400 gewählt:

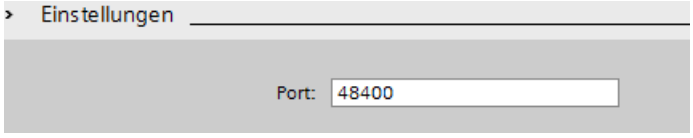

Einstellungen für Sessions

- Maximales Timeout für Sessions
  In diesem Feld legen Sie fest, wie lange die Zeitspanne höchstens sein darf, bis der OPC UA-Server eine Session ohne Datenaustausch abbaut.
  Mögliche Werte zwischen 1 und 600000 Sekunden.
- Maximale Anzahl OPC UA-Sessions
  In diesem Feld legen Sie fest, wie viele Sessions der OPC UA-Server der CPU höchstens aufbaut und gleichzeitig betreibt.
  Die maximale Anzahl der Sessions ist abhängig von der Leistungsfähigkeit der CPU. Jede Session bindet Ressourcen.

Maximale Anzahl registrierter Knoten

In diesem Feld legen Sie fest, wie viele Knoten (Nodes) der OPC UA-Server höchstens registriert.

Die maximale Anzahl der registrierten Knoten ist abhängig von der Leistungsfähigkeit der CPU und wird beim Projektieren des Feldinhalts angezeigt (Mauszeiger in das Feld setzen). Jede Registrierung bindet Ressourcen.

Hinweis

Keine Fehlermeldung beim Versuch, mehr Knoten zu registrieren als die projektierte maximale Anzahl registrierbarer Knoten

Wenn ein Client zur Laufzeit mehr Knoten registrieren will als die projektierte maximale Anzahl registrierbarer Knoten, dann registriert der Server der S7-1500 CPU nur die projektierte maximale Anzahl. Der Server liefert dem Client ab der projektierten maximalen Anzahl registrierbarer Knoten die regulären String-Node-Ids zurück, sodass der Geschwindigkeitsvorteil durch Registrierung für diese Knoten entfällt. Der Client erhält keine Fehlermeldung.

Achten Sie beim Projektieren auf eine ausreichende Reserve; berücksichtigen Sie dazu die maximale Anzahl registrierbarer Knoten (z. B. über die technischen Daten der CPU).

Weitere Informationen

Welche Ports die verschiedenen Dienste für die Datenübertragung über TCP und UDP verwenden und was bei der Verwendung von Routern und Firewalls zu beachten ist, finden Sie im FAQ (https://support.industry.siemens.com/cs/ww/de/view/8970169).

Einstellungen des Servers für Subscriptions

Subscription statt zyklisches Abfragen

Eine Alternative zum zyklischen Abfragen einer PLC-Variablen (Polling) ist das Beobachten dieses Werts. Nutzen Sie dazu eine Subscription (Abonnement): Wenn sich der Wert von PLC-Variablen geändert hat, informiert der Server den Client. Siehe "Der OPC UA-Client".

Ein Server überwacht meist sehr viele PLC-Werte. Deshalb sendet der Server in regelmäßigen Abständen Nachrichten (Notifikation) an den Client, in denen die neuen Werte der PLC-Variablen enthalten sind.

Wie oft sendet der Server Nachrichten?

Beim Anlegen einer Subscription gibt der OPC UA-Client seinen Wunsch an, in welchen Abständen der OPC UA-Client bei Wertänderung die neuen Werte erhalten möchte. Um die Kommunikationslast durch OPC UA zu begrenzen, legen Sie einen zeitlichen Mindestabstand für die Nachrichten fest. Dazu verwenden Sie die Parameter für das kleinste Sendeintervall und das kleinste Abtastintervall.

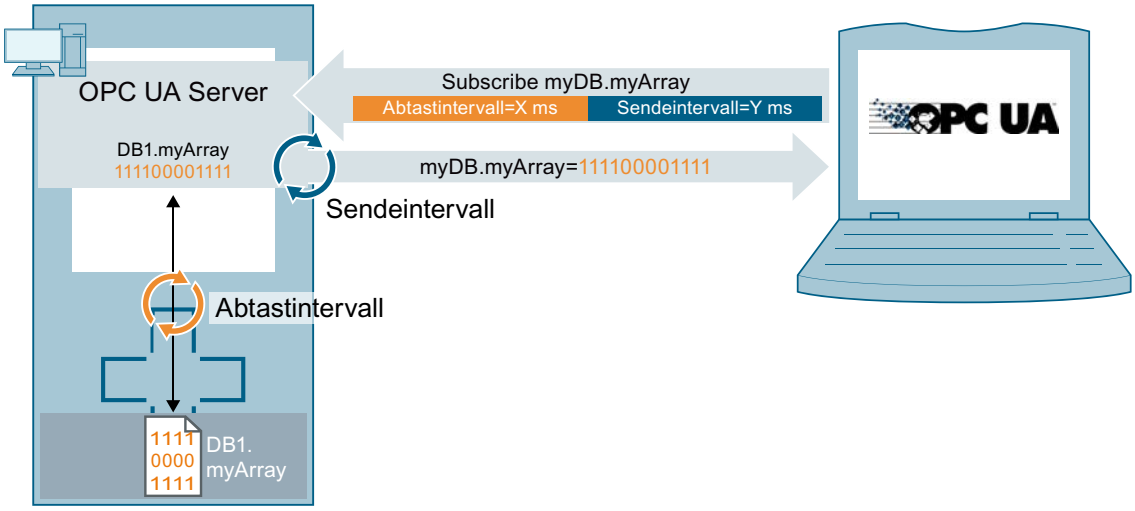

Kleinstes Sendeintervall

Bei "Kleinstes Sendeintervall" stellen Sie die zeitlichen Abstände (Intervalle) ein, in denen der Server bei Wertänderung eine Nachricht mit den neuen Werten an den Client schickt.

Im folgenden Bild wird als "Kleinstes Abtastintervall" der Wert 250 ms verwendet. Als "Kleinstes Sendeintervall" ist der Wert von 200 ms eingetragen.

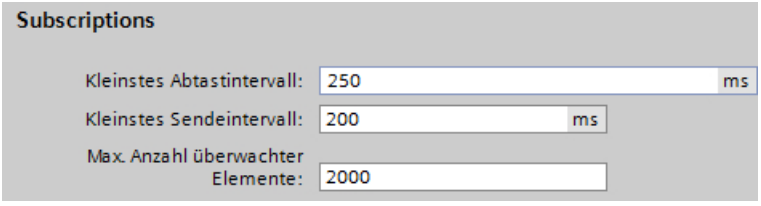

Im Beispiel sendet der OPC UA-Server bei Wertänderung alle 200 ms eine neue Nachricht, falls der OPC UA-Client eine Aktualisierung fordert.

Wenn der OPC UA-Client eine Aktualisierung alle 1000 ms fordert, dann sendet der OPC UA-Server auch nur einmal in 1000 ms (eine Sekunde) eine Nachricht mit den neuen Werten.

Wenn der OPC UA-Client eine Aktualisierung alle 100 ms fordert, dann sendet der Server trotzdem nur alle 200 ms (kleinstes Sendeintervall).

Kleinstes Abtastintervall

Bei "Kleinstes Abtastintervall" stellen Sie die zeitlichen Abstände (Intervalle) ein, in denen der OPC UA-Server den Wert einer CPU-Variablen erfasst und mit dem bisherigen Wert vergleicht, um eine Wertänderung festzustellen.

Wenn Sie das Abtastintervall kleiner gewählt haben als das Sendeintervall und ein OPC UA-Client für bestimmte PLC-Variablen eine hohe Abtastrate fordert, dann können pro Sendeintervall zwei oder mehr Werte anfallen.

In diesem Fall schreibt der OPC UA-Server die Wertänderungen in die Warteschlange und sendet nach Ablauf des Sendeintervalls alle Wertänderungen an den Client. Wenn mehr Wertänderungen im Sendeintervall anfallen als in die Warteschlange passen, dann

überschreibt der OPC UA-Server die ältesten Werte (abhängig von der eingestellten "Discard Policy", die Option "Discard Oldest" muss in diesem Fall aktiviert sein). Die aktuellsten Werte werden an den Client gesendet.

Maximale Anzahl überwachter Elemente (Monitored Items)

In diesem Feld legen Sie die maximale Anzahl an Elemente fest, die der OPC UA-Server der CPU gleichzeitig auf Wertänderung überwacht.

Die Überwachung bindet Ressourcen. Die maximale Anzahl überwachter Elemente ist abhängig von der verwendeten CPU.

Weitere Informationen

Informationen zu den Systemgrenzen des OPC UA-Servers der S7-1500 CPUs (Firmware V2.0 und V2.1) hinsichtlich Subscriptions, Abtastintervalle und Sendeintervalle entnehmen Sie folgendem FAQ (https://support.industry.siemens.com/cs/ww/de/view/109755846).

Bei der Verwendung von Subscriptions geben im Fehlerfall bestimmte Statuscodes Auskunft über den aufgetretenen Fehler. Informationen zu Ursachen und Abhilfen bei auftretenden Statuscodes im OPC UA Client finden Sie in der Liste der Fehlercodes in der Online-Hilfe zu STEP 7 (TIA Portal) oder im folgenden FAQ (https://support.industry.siemens.com/cs/ww/de/view/109755860).

Handling der Client- und Server-Zertifikate

Eine gesicherte Verbindung zwischen OPC UA-Server und einem OPC UA-Client kommt nur dann zu Stande, wenn sich der Server gegenüber dem Client ausweisen kann. Dazu dient das Zertifikat des Servers.

Zertifikat des OPC UA-Servers

Wenn Sie den OPC UA-Server aktiviert und die Sicherheitshinweise bestätigt haben, erzeugt STEP 7 automatisch das Zertifikat für den Server und speichert es im lokalen Zertifikate-Verzeichnis der CPU. Dieses Verzeichnis können Sie mit dem lokalen Zertifikatsmanager der CPU einsehen und verwalten (Zertifikate exportieren oder löschen).

Das folgende Bild zeigt den lokalen Zertifikatsmanager der CPU mit dem automatisch erzeugten Zertifikat für den OPC UA-Server:

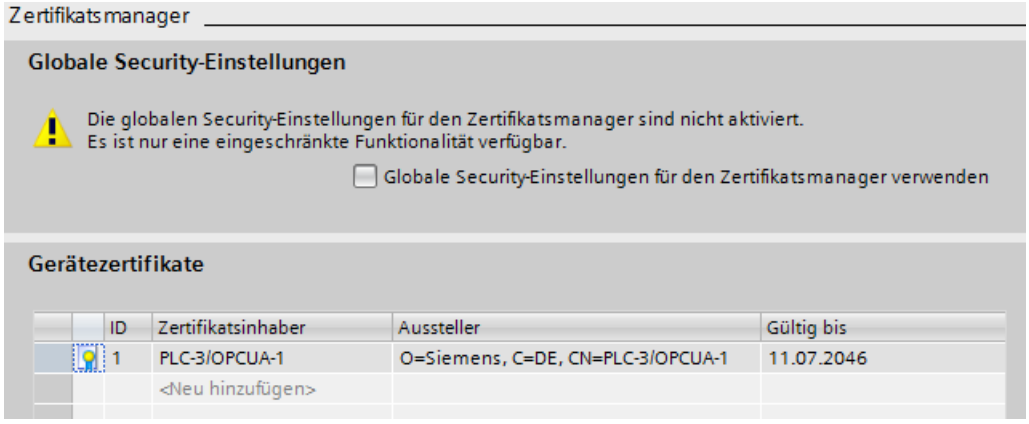

Bild 1-29 Lokaler Zertifikatsmanager der CPU

Alternativ können Sie auch selbst ein Server-Zertifikat erzeugen.

Das Zertifikat des Servers wird während des Aufbaus einer Verbindung vom Server zum Client übertragen, der Client überprüft das Zertifikat.

Der Client-Anwender entscheidet, ob er dem Zertifikat des Servers vertraut

Auf der Client-Seite muss nun der Anwender entscheiden, ob er dem Server-Zertifikat vertraut. Wenn der Anwender dem Server-Zertifikat vertraut, dann speichert der Client das Server-Zertifikat in seinem Verzeichnis, das die vertrauenswürdigen Server-Zertifikate enthält.

Das folgende Beispiel zeigt einen Dialog des Clients "UA Sample Client". Wenn der Anwender auf die Schaltfläche "Ja" klickt, vertraut der Client dem Server-Zertifikat:

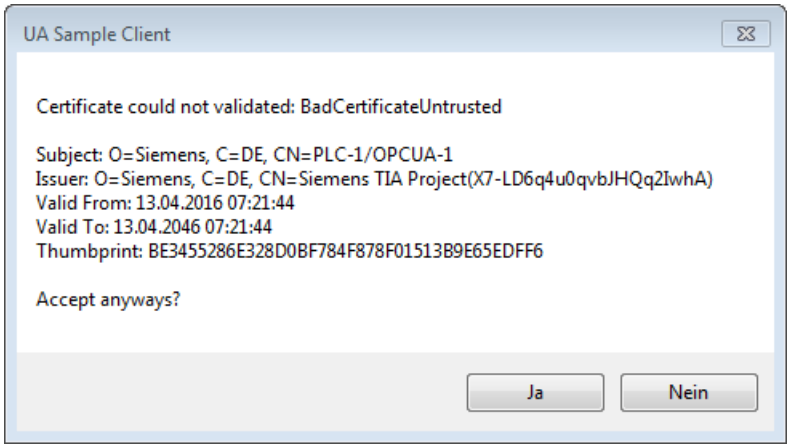

Bild 1-30 Dialog des Clients "UA Sample Client"

Siehe auch

Server-Zertifikate mit STEP 7 erzeugen (Seite 1242)

Nachrichten gesichert übertragen

Woher kommt das Zertifikat eines Clients?

Wenn Sie UA-Clients von Herstellern oder der OPC Foundation verwenden, wird bei der Installation oder beim ersten Programmaufruf automatisch ein Client-Zertifikat erzeugt. Diese Zertifikate müssen Sie über den globalen Zertifikatsmanager in STEP 7 importieren und für die jeweilige CPU verwenden (wie oben gezeigt).

Wenn Sie selbst einen OPC UA-Client programmieren, dann können Sie Zertifikate programmtechnisch erstellen, siehe "Instanz-Zertifikat für den Client". Oder Sie erzeugen Zertifikate mit Tools, zum Beispiel mit OpenSSL oder dem Zertifikate-Generator der OPC Foundation:

- Wie Sie bei OpenSSL vorgehen, lesen Sie hier: "PKI-Schlüsselpaare und Zertifikate selbst erzeugen".
- Wie Sie mit dem Zertifikate-Generator der OPC Foundation arbeiten, lesen Sie hier: "Selbst-signierte Zertifikate erzeugen".

Client-Zertifikate dem Server bekanntmachen

Client-Zertifikate müssen Sie dem Server zur Verfügung stellen, damit eine gesicherte Verbindung aufgebaut werden kann.

Dazu gehen Sie folgendermaßen vor:

1. Aktivieren Sie im lokalen Zertifikatsmanager des Servers die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". Dadurch ist der globale Zertifikatsmanager verfügbar.
   Sie finden diese Option in den Eigenschaften der CPU, die als Server dient, unter "Schutz & Security > Zertifikatsmanager".
   Falls dieses Projekt noch nicht geschützt ist, dann klicken Sie in der "Projektnavigation" von STEP 7 unter "Security-Einstellungen > Einstellungen" auf die Schaltfläche "Dieses Projekt schützen" und melden Sie sich an.
   STEP 7 zeigt nun in der "Projektnavigation" unter "Security-Einstellungen" der Eintrag "Globale Security-Einstellungen" an.
2. Doppelklicken Sie auf "Globale Security-Einstellungen".
3. Doppelklicken Sie auf "Zertifikatsmanager".
   STEP 7 öffnet den globalen Zertifikatsmanager.
4. Klicken Sie auf das Register "Vertrauenswürdige Zertifikate".
5. Klicken Sie mit der rechten Maustaste im Register auf eine freie Fläche (nicht auf ein Zertifikat).
6. Wählen Sie aus dem Kontextmenu den Eintrag "Importieren".
   Der Dialog zum Importieren von Zertifikaten wird angezeigt.
7. Wählen Sie das Client-Zertifikat aus, dem der Server vertrauen soll.
8. Klicken Sie auf die Schaltfläche "Öffnen", um das Zertifikat zu importieren.
   Das Zertifikat des Clients ist nun im globalen Zertifikatsmanager enthalten.
   Merken Sie sich die ID des gerade importierten Client-Zertifikats.
9. Klicken Sie nun in den Eigenschaften der CPU, die als Server dient, auf das Register "Allgemein".
10. Klicken Sie auf den Bereich "OPC UA > Server > Security > Secure Channel".

11. Scrollen Sie im Dialog "Secure Channel" nach unten zum Abschnitt "Vertrauenswürdige Clients".
12. Doppelklicken Sie in der Tabelle auf die leere Zeile mit "<neu hinzufügen>". In der Zeile wird eine Schaltfläche mit drei Punkten angezeigt.
13. Klicken Sie auf diese Schaltfläche.
14. Wählen Sie das Client-Zertifikat aus, das Sie importiert haben.
15. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
16. Übersetzen Sie das Projekt.
17. Laden Sie die Konfiguration in die S7-1500 CPU.

Ergebnis

Der Server vertraut nun dem Client. Wenn außerdem das Server-Zertifikat als vertrauenswürdig gilt, dann können Server und Client eine gesicherte Verbindung aufbauen.

Client-Zertifikate automatisch akzeptieren

Wenn Sie die Option "Client-Zertifikate zur Laufzeit automatisch akzeptieren" aktivieren (unterhalb der Liste "Vertrauenswürdige Clients"), dann akzeptiert der Server alle Client-Zertifikate.

ACHTUNG
Einstellung nach der Inbetriebnahme Um Sicherheitsrisiken zu vermeiden, deaktivieren Sie die Option "Client-Zertifikate zur Laufzeit automatisch akzeptieren" wieder nach der Inbetriebnahme.

Security-Einstellungen des Servers konfigurieren

Das folgende Bild zeigt die verfügbaren Security-Einstellungen des Servers zum Signieren und Verschlüsseln von Nachrichten.

Bild 1-31 Security-Einstellungen des Servers konfigurieren

Per Voreinstellung wird ein Server-Zertifikat erstellt, welches SHA256-Signierung nutzt. Die folgenden Security Policies sind freigegeben:

- Keine
  Ungesicherter Endpoint

  Hinweis

  Nicht erwünschte Security Policies deaktivieren

  Wenn Sie bei den Secure-Channel-Einstellungen des S7-1500 OPC UA-Servers alle Security Policies aktiviert haben (Voreinstellung) - also auch den Endpunkt "Keine Security" - dann ist der Datenverkehr zwischen Server und Client auch ungesichert möglich (weder signiert noch verschlüsselt). Die Identität des Clients bleibt bei "Keine Security" unbekannt. Jeder OPC UA-Client kann sich dann mit dem Server verbinden, unabhängig von sämtlichen noch folgenden Security-Einstellungen.

  Achten Sie bei der Projektierung des OPC UA-Servers darauf, dass nur Security Policies aktiviert sind, die mit dem Schutzkonzept für Ihre Maschine oder Anlage vereinbar sind. Alle anderen Security Policies sind zu deaktivieren.

  Empfehlung: Verwenden Sie, wenn möglich, die Einstellung "Basic256Sha256".

- Basic128Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.

- Basic128Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Basic256Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Basic256Sha256 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Sha256 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.

Um eine Security-Einstellung frei zu geben, klicken Sie auf das Kästchen in der jeweiligen Zeile.

Hinweis

Wenn Sie die Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" nutzen, dann müssen OPC UA-Server und OPC UA-Clients "SHA256"-signierte Zertifikate verwenden.

Bei den Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" signiert die Zertifizierungsstelle von STEP 7 die Zertifikate automatisch mit "SHA256".

Security Policy "Keine Security" und Authentifizierung über Benutzername und Passwort

Folgende Kombination können Sie einstellen:

Security Policy = "Keine Security" und Authentifizierung über Benutzername und Passwort.

- Der OPC UA-Server der S7-1500 unterstützt diese Kombination. OPC UA-Clients können sich verbinden und die Authentifizierungsdaten verschlüsseln oder auch nicht.
- Der OPC UA-Client der S7-1500 CPU unterstützt ebenfalls diese Kombination: Zur Laufzeit verbindet er sich aber nur dann, wenn er die Authentifizierungsdaten verschlüsselt über die Leitung schicken kann!

Handling der Client-Zertifikate der S7-1500 CPU

Woher kommt das Zertifikat des Clients?

Wenn Sie den OPC UA-Client einer S7-1500 CPU nutzen (OPC UA-Client aktiviert), dann können Sie mit STEP 7 ab V15 für diese Clients Zertifikate erzeugen, wie in den folgenden Abschnitten beschrieben ist.

Wenn Sie UA-Clients von Herstellern oder der OPC Foundation verwenden, wird bei der Installation oder beim ersten Programmaufruf automatisch ein Client-Zertifikat erzeugt. Diese Zertifikate müssen Sie über den globalen Zertifikatsmanager in STEP 7 importieren und für die jeweilige CPU verwenden.

Wenn Sie selbst einen OPC UA-Client programmieren, dann können Sie Zertifikate programmtechnisch erstellen, siehe "Instanz-Zertifikat für den Client". Oder Sie erzeugen Zertifikate mit Tools, zum Beispiel mit OpenSSL oder dem Zertifikate-Generator der OPC Foundation:

- Wie Sie bei OpenSSL vorgehen, lesen Sie hier: "PKI-Schlüsselpaare und Zertifikate selbst erzeugen".
- Wie Sie mit dem Zertifikate-Generator der OPC Foundation arbeiten, lesen Sie hier: "Selbstsignierte Zertifikate erzeugen".

Zertifikat des OPC UA-Clients der S7-1500 CPU

Eine gesicherte Verbindung zwischen OPC UA-Server und einem OPC UA-Client kommt nur dann zu Stande, wenn der Server das Zertifikat des Clients als vertrauenswürdig einstuft.

Dazu müssen Sie dem Server das Client-Zertifikat bekanntmachen.

Die folgenden Abschnitte beschreiben, wie Sie zunächst für den OPC UA-Client der S7-1500 CPU ein Zertifikat erzeugen und es dann dem Server zur Verfügung stellen.

1. Zertifikat für den Client erzeugen und exportieren

Für eine gesicherte Verbindung müssen Sie ein Client-Zertifikat erzeugen und das Zertifikat exportieren.

Dazu gehen Sie folgendermaßen vor:

1. In der "Projektnavigation" wählen Sie die CPU aus, die als Client fungiert.
2. Doppelklicken Sie auf "Gerätekonfiguration"
3. In den Eigenschaften der CPU klicken Sie auf "Schutz & Security > Zertifikatsmanager".
4. In der Tabelle "Gerätezertifikate" doppelklicken Sie auf "<neu hinzufügen>".
   STEP 7 öffnet einen Dialog.
5. Klicken Sie auf die Schaltfläche "Hinzufügen".

6. Bei "Verwendungszweck" wählen Sie aus der Liste den Eintrag "OPC UA-Client".
   Achtung:
   Unter "Alternativer Name des Zertifikatsinhabers (SAN)" muss die IP-Adressen eingetragen sein, unter der die CPU in Ihrer Anlage erreichbar ist.
   Sie müssen also die IP-Schnittstelle der CPU konfigurieren, bevor Sie ein Client-Zertifikat erzeugen.
7. Klicken Sie auf "OK".
   STEP 7 zeigt nun das Client-Zertifikat in der Tabelle "Gerätezertifikate" an.
8. Klicken Sie mit der rechten Maustaste auf diese Zeile und wählen Sie aus dem Kontextmenü den Eintrag "Zertifikat exportieren".
9. Wählen Sie ein Verzeichnis aus, in dem Sie das Client-Zertifikat speichern.

2. Das Client-Zertifikat dem Server bekanntmachen

Das Client-Zertifikat müssen Sie dem Server zur Verfügung stellen, damit eine gesicherte Verbindung aufgebaut werden kann.

Dazu gehen Sie folgendermaßen vor:

1. Aktivieren Sie im lokalen Zertifikatsmanager des Servers die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". Dadurch ist der globale Zertifikatsmanager verfügbar.
   Sie finden diese Option in den Eigenschaften der CPU, die als Server dient, unter "Schutz & Security > Zertifikatsmanager".
   Falls dieses Projekt noch nicht geschützt ist, dann klicken Sie in der "Projektnavigation" von STEP 7 unter "Security-Einstellungen > Einstellungen" auf die Schaltfläche "Dieses Projekt schützen" und melden Sie sich an.
   STEP 7 zeigt nun in der "Projektnavigation" unter "Security-Einstellungen" der Eintrag "Globale Security-Einstellungen" an.
2. Doppelklicken Sie auf "Globale Security-Einstellungen".
3. Doppelklicken Sie auf "Zertifikatsmanager".
   STEP 7 öffnet den globalen Zertifikatsmanager.
4. Klicken Sie auf das Register "Vertrauenswürdige Zertifikate".
5. Klicken Sie mit der rechten Maustaste im Register auf eine freie Fläche (nicht auf ein Zertifikat).
6. Wählen Sie aus dem Kontextmenu den Eintrag "Importieren".
   Der Dialog zum Importieren von Zertifikaten wird angezeigt.
7. Wählen Sie das Client-Zertifikat aus, dem der Server vertrauen soll.
8. Klicken Sie auf die Schaltfläche "Öffnen", um das Zertifikat zu importieren.
   Das Zertifikat des Clients ist nun im globalen Zertifikatsmanager enthalten.
   Merken Sie sich die ID des gerade importierten Client-Zertifikats.
9. Klicken Sie nun in den Eigenschaften der CPU, die als Server dient, auf das Register "Allgemein".
10. Klicken Sie auf den Bereich "OPC UA > Server > Security > Secure Channel".
11. Scrollen Sie im Dialog "Secure Channel" nach unten zum Abschnitt "Vertrauenswürdige Clients".

12. Doppelklicken Sie in der Tabelle auf die leere Zeile mit "<neu hinzufügen>". In der Zeile wird eine Schaltfläche mit drei Punkten angezeigt.
13. Klicken Sie auf diese Schaltfläche.
14. Wählen Sie das Client-Zertifikat aus, das Sie importiert haben.
15. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
16. Übersetzen Sie das Projekt.
17. Laden Sie die Konfiguration in die S7-1500 CPU.

Ergebnis

Der Server vertraut nun dem Client. Wenn außerdem das Server-Zertifikat als vertrauenswürdig gilt, dann können Server und Client eine gesicherte Verbindung aufbauen.

Server-Zertifikate mit STEP 7 erzeugen

Die folgende Beschreibung zeigt die Vorgehensweise zur Erzeugung neuer Zertifikate mit STEP 7 und gilt prinzipiell für verschiedene Verwendungen der Zertifikate. Je nachdem, über welchen Bereich der CPU-Eigenschaften Sie den folgenden Dialog starten, stellt STEP 7 den passenden Verwendungszweck bereits ein - hier "OPC UA-Client & -Server".

Empfehlung: Um die volle Funktionalität für die Security des OPC UA-Servers zu nutzen, verwenden Sie die globalen Security-Einstellungen.

Die globalen Security-Einstellungen aktivieren Sie in den CPU-Eigenschaften, Bereich "Schutz & Security > Zertifikatsmanager".

Server-Zertifikate individuell anpassen

STEP 7 erzeugt automatisch ein Zertifikat für den OPC UA-Server der S7-1500, wenn Sie den Server aktivieren (siehe "OPC UA-Server aktivieren (Seite 1227)"). Dabei verwendet STEP 7 für die Parameter des Zertifikats voreingestellte Werte. Wenn Sie die Parameter ändern wollen, dann gehen Sie folgendermaßen vor:

1. Klicken Sie in den Eigenschaften der CPU unter "Allgemein > OPC UA > Server > Security > Secure Channel > Server-Zertifikat" auf die Drei-Punkte-Schaltfläche. Ein Dialog wird eingeblendet, der lokal vorhandene Zertifikate anzeigt.
2. Klicken Sie auf die Schaltfläche "Hinzufügen".

3. Der Dialog zum Erzeugen neuer Zertifikate wird angezeigt (folgendes Bild). Die Werte für ein Beispiel sind bereits eingetragen:

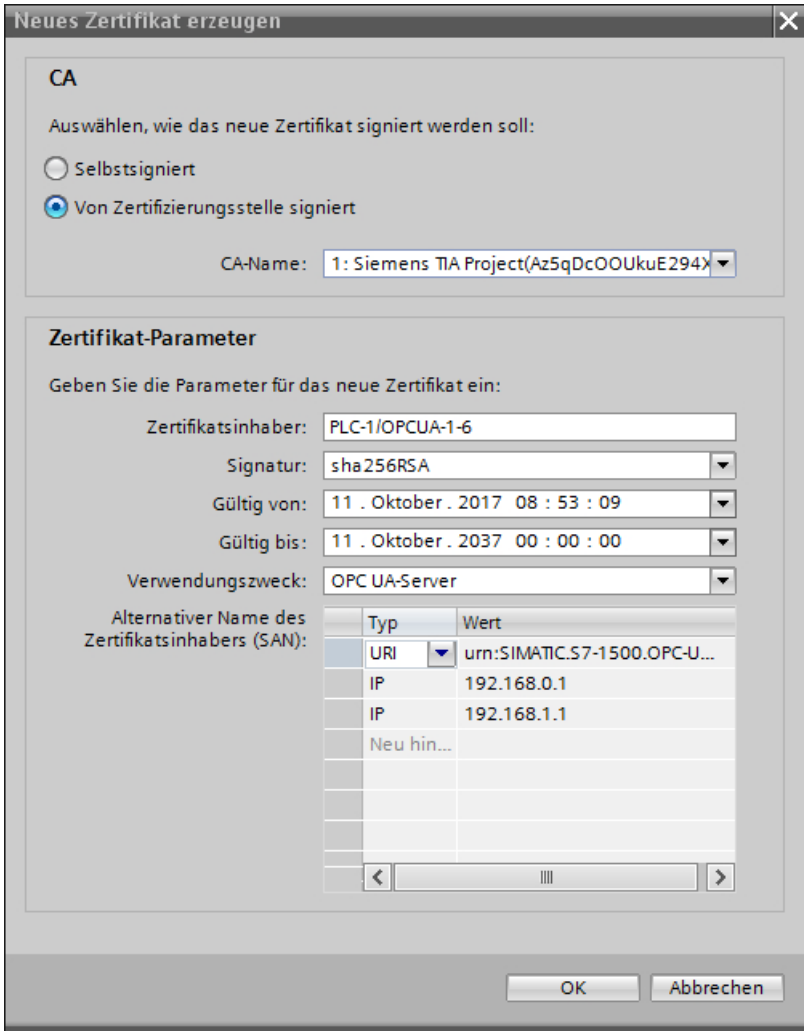

Bild 1-32 Server-Zertifikate individuell anpassen

4. Verwenden Sie andere Parameter, falls dies nach den Sicherheitsvorgaben in Ihrem Unternehmen oder des Auftraggebers erforderlich ist.

Erläuterung der Felder für die Zertifikatserzeugung

- CA
  Wählen Sie aus, ob das Zertifikat selbst-signiert sein soll oder von einem der CA-Zertifikate des TIA Portals. Die Zertifikate sind unter "Zertifikate bei OPC UA" beschrieben. Wenn Sie ein Zertifikat erzeugen wollen, das von einem der CA-Zertifikate des TIA-Portals signiert sein soll, dann muss das Projekt geschützt sein und Sie müssen als Benutzer mit den erforderlichen Funktionsrechten angemeldet sein. Informationen hierzu erhalten Sie unter "Grundlagen der Benutzerverwaltung im TIA Portal".
- Zertifikatsinhaber
  Die Voreinstellung setzt sich zusammen aus dem Namen des Projekts und "\OPCUA-1". Im Beispiel lautet der Projekt-Name "PLC1". In den Eigenschaften der CPU stellen Sie den Projektnamen ein unter "Allgemein > Projektinformation > Name". Behalten Sie die Voreinstellung bei oder tragen Sie bei "Zertifikatsinhaber" einen anderen Namen für den OPC UA-Server ein, der in Ihrem Projekt aussagekräftiger ist.
- Signatur
  Wählen Sie hier das Hash- und Verschlüsselungsverfahren aus, das beim Signieren des Server-Zertifikats verwendet werden soll. Die folgenden Einträge sind verfügbar:
  - "sha1RSA",
  - "sha256RSA".
- Gültig von
  Tragen Sie hier das Datum und die Uhrzeit ein für den Beginn der Gültigkeit des Server-Zertifikats.
- Gültig bis
  Tragen Sie hier das Datum und die Uhrzeit ein für das Ende der Gültigkeit des Server-Zertifikats. Achten Sie darauf, dass das Zertifikat nicht nur ein Jahr oder wenige Jahre gültig ist. Im Beispiel ist das Zertifikat 30 Jahre gültig. Aus Sicherheitsgründen sollten Sie allerdings das Zertifikat in viel kürzeren Abständen erneuern. Die lange Gültigkeit überlässt aber Ihnen die Entscheidung, wann dazu der passende Zeitpunkt ist, etwa wenn eine Anlage gewartet werden muss.

- Verwendungszweck
  Voreingestellt ist "OPC UA-Client & -Server". Behalten Sie diese Voreinstellung für den OPC UA-Server bei. Der Dialog "Neues Zertifikat erzeugen" kann von mehreren Stellen aus in STEP 7 aufgerufen werden. Wenn Sie zum Beispiel diesen Dialog für den Webserver der CPU aufrufen, dann wird unter "Verwendungszweck" "Webserver" eingetragen. In der Klappliste zum Verwendungszweck sind die folgenden Einträge verfügbar:
  - "OPC UA-Client"
  - "OPC UA-Client & -Server"
  - "OPC UA-Server"
  - "TLS"
  - "Webserver"
- Alternativer Name des Zertifikatsinhabers
  Im Beispiel oben ist eingetragen: "URI:urn:SIMATIC.S7-1500.OPC-UAServer:PLC1,IP: 192.168.178.151,IP:192.168.1.1". Gültig wäre auch der folgende Eintrag: "IP: 192.168.178.151, IP: 192.168.1.1". Wichtig ist, dass hier die IP-Adressen eingetragen sind, über die der OPC UA-Server der CPU erreichbar ist (siehe "Zugang zum OPC UA-Server (Seite 1229)"). Dadurch können OPC UA-Clients überprüfen, ob tatsächlich eine Verbindung zum OPC UA-Server der S7-1500 aufgebaut werden soll, oder möglicherweise ein Angreifer versucht, dem OPC UA-Client von einem anderen PC aus manipulierte Werte zuzusenden.

Security-Einstellungen des OPC UA-Server bearbeiten

OPC UA verwendet sichere Verbindungen zwischen Client und Server. Dabei überprüft OPC UA die Identität der Kommunikationspartner. Für die Authentifizierung von Client und Server nutzt OPC UA Zertifikate gemäß X.509-V3 der ITU (International Telecommunication Union).

Ausnahme: Bei der Security Policy "Keine Security" wird keine sichere Verbindung aufgebaut.

Security-Einstellungen des Servers konfigurieren

Das folgende Bild zeigt die verfügbaren Security-Einstellungen des Servers zum Signieren und Verschlüsseln von Nachrichten.

Die Security Policies sind nach den verwendeten Algorithmen benannt. Beispiel: "Basic256Sha256 - Signieren & Verschlüsseln" bedeutet: Gesicherter Endpoint unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.

Bild 1-33 Security-Einstellungen des Servers konfigurieren

Per Voreinstellung wird ein Server-Zertifikat erstellt, welches SHA256-Signierung nutzt. Die folgenden Security Policies sind freigegeben:

- Keine
  Ungesicherter Endpoint

  Hinweis

  Nicht erwünschte Security Policies deaktivieren

  Wenn Sie bei den Secure-Channel-Einstellungen des S7-1500 OPC UA-Servers alle Security Policies aktiviert haben (Voreinstellung) - also auch den Endpunkt "Keine Security" - dann ist der Datenverkehr zwischen Server und Client auch ungesichert möglich (weder signiert noch verschlüsselt). Die Identität des Clients bleibt bei "Keine Security" unbekannt. Jeder OPC UA-Client kann sich dann mit dem Server verbinden, unabhängig von sämtlichen noch folgenden Security-Einstellungen.

  Achten Sie bei der Projektierung des OPC UA-Servers darauf, dass nur Security Policies aktiviert sind, die mit dem Schutzkonzept für Ihre Maschine oder Anlage vereinbar sind. Alle anderen Security Policies sind zu deaktivieren.

  Empfehlung: Verwenden Sie, wenn möglich, die Einstellung "Basic256Sha256".

- Basic128Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.

- Basic128Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.

- Basic256Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Basic256Sha256 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Sha256 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.

Um eine Security-Einstellung freizugeben, klicken Sie auf das Kästchen in der jeweiligen Zeile.

Hinweis

Wenn Sie die Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" nutzen, dann müssen OPC UA-Server und OPC UA-Clients "SHA256"-signierte Zertifikate verwenden.

Bei den Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" signiert die Zertifizierungsstelle von STEP 7 die Zertifikate automatisch mit "SHA256".

Security Policy "Keine Security" und Authentifizierung über Benutzername und Passwort

Folgende Kombination können Sie einstellen:

Security Policy = "Keine Security" und Authentifizierung über Benutzername und Passwort.

- Der OPC UA-Server der S7-1500 unterstützt diese Kombination. OPC UA-Clients können sich verbinden und die Authentifizierungsdaten verschlüsseln oder auch nicht.
- Der OPC UA-Client der S7-1500 CPU unterstützt ebenfalls diese Kombination: Zur Laufzeit verbindet er sich aber nur dann, wenn er die Authentifizierungsdaten verschlüsselt über die Leitung schicken kann!

Siehe auch

Authentifizierung des Benutzers (Seite 1248)

Authentifizierung des Benutzers

Arten der Benutzer-Authentifizierung

Sie können beim OPC UA-Server der S7-1500 einstellen, wie sich ein Benutzer des OPC UA-Clients legitimieren muss, wenn er auf den Server zugreifen will.

Dazu gibt es die folgenden Möglichkeiten:

- **Gast-Authentifizierung**
  Der Anwender muss seine Berechtigung nicht nachweisen (anonymer Zugang). Der OPC UA-Server überprüft nicht die Berechtigung des Client-Anwenders.
  Wenn Sie diese Art der Benutzer-Authentifizierung nutzen wollen, dann wählen Sie unter "OPC UA > Server > Security > Benutzer-Authentifizierung" die Option "Gast-Authentifizierung aktivieren".

  Hinweis

  Um die Security zu erhöhen, sollten Sie den Zugriff auf den OPC UA-Server nur mit Benutzer-Authentifizierung erlauben!

- **Authentifizierung über Benutzername und Passwort**
  Der Anwender muss seine Berechtigung nachweisen (kein anonymer Zugang). Der OPC UA-Server überprüft, ob der Client-Anwender berechtigt ist, auf den Server zuzugreifen. Als Nachweis gilt der Benutzername mit dem richtigen Passwort.
  Wenn Sie diese Art der Benutzer-Authentifizierung nutzen wollen, dann wählen Sie unter "OPC UA > Server > Security > Benutzer-Authentifizierung" die Option "Authentifizierung über Benutzername und Passwort aktivieren".
  Deaktivieren Sie die Gast-Authentifizierung.
  Tragen Sie in der Tabelle "Benutzerverwaltung" die Anwender ein.
  Klicken Sie dazu jeweils auf den Eintrag "<Neuen Benutzer hinzufügen>". Ein neuer Benutzer mit einem automatisch vergebenem Namen wird angelegt. Sie können den Benutzernamen editieren und das für den Benutzernamen zugehörige Passwort eingeben. Sie können maximal 21 Benutzer hinzufügen.

- **Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts**
  Wenn Sie diese Option aktivieren, dann wird die Benutzerverwaltung des geöffneten Projekts auch für die Benutzer-Authentifizierung des OPC UA-Servers verwendet: Bei OPC UA sind dann dieselben Benutzernamen und Passwörter gültig wie im aktuellen Projekt.
  Um die Benutzerverwaltung des Projekts zu aktivieren, gehen Sie folgendermaßen vor:
  - Klicken Sie in der "Projektnavigation" auf "Security-Einstellungen > Einstellungen".
  - Klicken Sie auf Schaltfläche "Dieses Projekt schützen".
  - Tragen Sie Ihren Benutzernamen und Ihr Passwort ein.
  - Unter "Security-Einstellungen > Benutzer und Rollen" tragen Sie weitere Benutzer ein.

- Wenn Sie in Ihrem Projekt einen weiteren OPC UA-Server projektieren, dann aktivieren Sie in dem Projekt ebenfalls die Option "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren". Dadurch ist eine erneute Eingabe von Benutzernamen und Passwörtern unnötig.

Siehe auch

Security-Einstellungen des OPC UA-Server bearbeiten (Seite 1245)

Authentifizierung des Benutzers (Seite 1309)

Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 1249)

Benutzer und Rollen mit OPC UA-Funktionsrechten

Folgende Optionen zur Benutzer-Authentifizierung greifen auf zentrale Projekteinstellungen für Projektbenutzer zurück:

- Beim Server:
  Beim Parametrieren der CPU-Eigenschaften (OPC UA > Server > Security > Benutzer-Authentifizierung). Dort ist es die Option "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren"
- Beim Client:
  Beim Konfigurieren der Client-Schnittstelle (Register "Konfiguration", Bereich "Security"). Dort ist es die Option "Benutzer (TIA Portal - Security-Einstellungen)"

Voraussetzung

Um die Security-Einstellungen bearbeiten zu können, muss das Projekt geschützt sein und Sie sind mit ausreichenden Rechten, z. B. als Administrator, angemeldet.

Einstellungen in der Projektnavigation > "Security-Einstellungen"

Die zentralen Benutzereinstellungen und Rollen erreichen Sie im geschützten Projekt in der Projektnavigation, Bereich "Security-Einstellungen". Hier definieren Sie zentral Benutzer mit ihrem Benutzernamen, Passwort sowie Funktionsrechten. Diese Einstellungen können Sie an anderer Stelle einfach wiederverwenden.

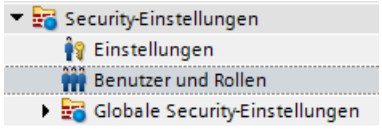

Wiederverwenden zentraler Security-Einstellungen

Beispiele für andere Stellen der Wiederverwendung sind:

- Benutzer-Auswahl für die Benutzer-Authentifizierung beim OPC UA-Server
  Bei dieser Einstellung teilen Sie dem Server mit, welcher Client (Benutzer) mit welchem Benutzernamen und welchem Passwort überhaupt auf den Server zugreifen darf.
- Benutzer-Auswahl für die OPC UA-Client-Authentifizierung
  Bei dieser Einstellung teilen Sie dem Client mit, mit welchem Benutzernamen und welchem Passwort er sich als Client beim Server authentifiziert.

Die Einstellungen für den Client und den Server müssen zusammenpassen: Der Benutzername und das Passwort, mit dem der Client sich anmeldet, muss beim Server eingerichtet und mit den entsprechenden Berechtigungen versehen sein.

Funktionsrechte für Server und Client

Für die Benutzer der Client-Funktionalität und für die Benutzer der Server-Funktionalität bei einer S7-1500 CPU müssen auch die entsprechenden Funktionsrechte für den Client bzw. für den Server aktiviert sein. Es reicht nicht aus, nur Benutzernamen und Passwort zentral zu hinterlegen.

Ein Beispiel erläutert diese Art der Rechteverwertung.

1. Sie definieren im Bereich "Security-Einstellungen > Benutzer und Rollen" eine neue Rolle im Register "Rollen" mit dem Namen z. B. "PLC-opcua-role-all-inclusive".
   Tipp: Möglicherweise wird das Register durch ein Informationsfenster verdeckt ("Der aktuelle Status wurde noch nicht überprüft..."). Schließen Sie in dem Fall zunächst das Informationsfenster.
2. Im Bereich "Kategorien von Funktionsrechten" navigieren Sie zu den Runtime-Rechten, dann zu den CPU-Funktionsrechten und markieren darunter die CPU, deren Funktionsrechte Sie einstellen wollen, z. B. PLC_2.
3. Im Bereich "Funktionsrechte" finden Sie folgende Funktionsrechte:
   - OPC UA Server-Zugriff
     Dieses Funktionsrecht wirkt am OPC UA-Server der S7-1500 CPU. Nur wenn diese Option markiert ist, hat ein Benutzer des Servers der CPU PLC_2, dem die Rolle "PLC-opcua-role-all-inclusive" zugewiesen ist, folgendes Recht: Er erzwingt für den Aufbau einer Session mit dem Server vom Client die Authentifizierung mit einem der zentral definierten (und in die CPU geladenen) Benutzernamen und zugehörigen Passwörtern.
   - Benutzer-Authentifizierung des OPC UA-Clients
     Dieses Funktionsrecht wirkt am OPC UA-Client der S7-1500 CPU (an den Client-Anweisungen). Nur wenn diese Option markiert ist, kann der Benutzer des Clients der CPU PLC_2, dem die Rolle "PLC-opcua-role-all-inclusive" zugewiesen ist, den entsprechenden Benutzernamen und das Passwort nutzen, um sich für den Aufbau einer Session mit einem Server zu authentifizieren.

4. Die Rolle "PLC-opcua-role-all-inclusive" müssen Sie noch den entsprechenden Benutzern zuweisen (Register "Benutzer" im Bereich "Security-Einstellungen" in der Projektnavigation).

Siehe auch

Authentifizierung des Benutzers (Seite 1248)

Authentifizierung des Benutzers (Seite 1309)

Lizenzen für den OPC UA-Server

Runtime-Lizenzen

Für den Betrieb des OPC UA-Servers der S7-1500 CPU ist eine Lizenz erforderlich. Der Typ der erforderlichen Lizenz ist abhängig von der Leistung der jeweiligen CPU. Die folgenden Lizenz-Typen werden unterschieden:

- SIMATIC OPC UA S7-1500 small (erforderlich für CPU 1511, CPU 1512, CPU 1513, ET 200SP CPUs, CPU 1515SP PC)
- SIMATIC OPC UA S7-1500 medium (erforderlich für CPU 1515, CPU 1516, Softwarecontroller CPU 1507, CPU 1516pro-2PN)
- SIMATIC OPC UA S7-1500 large (erforderlich für CPU 1517, CPU 1518)

Der erforderliche Lizenz-Typ wird angezeigt unter "Eigenschaften > Allgemein > Runtime-Lizenzen > OPC-UA > Typ der benötigten Lizenz":

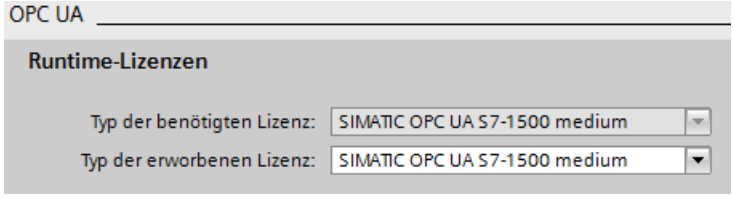

Um den Erwerb der erforderlichen Lizenz zu bestätigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in den Eigenschaften der CPU auf "Runtime-Lizenzen > OPC UA".
2. Wählen Sie in der Klappliste bei "Typ der erworbenen Lizenz" die notwendige Lizenz aus.

Methoden auf dem OPC UA-Server bereitstellen

Wissensertes zu Server-Methoden

Anwenderprogramm für Server-Methoden bereitstellen

Auf dem OPC UA-Server einer S7-1500 CPU (ab Firmware V2.5) haben Sie die Möglichkeit, Methoden über Ihr Anwenderprogramm bereitzustellen. Diese Methoden können von OPC UA-Clients genutzt werden, um z. B. einen Fertigungsauftrag über den Methodenaufruf von der S7-1500 CPU zu starten.

OPC UA-Methoden, eine Realisierung von "Remote Procedure Calls", bieten einen effizienten Mechanismus für die Interaktionen zwischen verschiedenen Kommunikationsteilnehmern. Der Mechanismus liefert sowohl eine Auftragsbestätigung als auch Rückgabewerte, sodass Sie auf die Ausprogrammierung von Handshaking-Mechanismen verzichten können.

Mit OPC UA-Methoden können Sie z. B. Daten konsistent ohne Triggerbits/Handshaking übertragen oder bestimmte Aktionen auf der Steuerung auslösen.

Wie funktioniert eine OPC UA-Methode?

Eine OPC UA-Methode funktioniert im Prinzip wie ein Knowhow geschützter Funktionsbaustein, der von einem externen OPC UA-Client zur Laufzeit aufgerufen wird.

Der OPC UA-Client "sieht" lediglich die definierten Ein- und Ausgänge. Das Innere des Funktionsbausteins, die Methode oder der Algorithmus, bleibt dem externen OPC UA-Client verborgen. Der OPC UA-Client bekommt eine Rückmeldung über die erfolgreiche Ausführung und Rückgabewerte, die der Funktionsbaustein (Methode) liefert. Oder eine Fehlermeldung bei nicht erfolgreicher Ausführung.

Sie haben als Programmierer die vollständige Kontrolle und Verantwortung darüber, in welchem Programmkontext die OPC UA-Methode abläuft.

Regeln für die Programmierung einer Methode und Laufzeitverhalten

- Sorgen Sie dafür, dass die über die OPC UA-Methode gelieferten Rückgabewerte konsistent zu den vom OPC UA-Client gelieferten Eingabewerten sind.
- Berücksichtigen Sie die Regeln zur Namensvergabe und Aufbau von Parametern sowie die verwendbaren Datentypen (siehe Beschreibung der OPC UA-Server-Anweisungen).
- Verhalten zur Laufzeit: Der OPC UA-Server nimmt **einen** Aufruf pro Instanz an. Erst wenn dieser Aufruf vom Anwenderprogramm abgearbeitet wurde oder Timeout hatte, ist diese Methoden-Instanz wieder für andere OPC UA-Clients aufrufbar.

Im Folgenden wird die prinzipielle Vorgehensweise gezeigt, mit der Sie ein Anwenderprogramm als Server-Methode implementieren.

Implementierung einer Server-Methode

Ein Programm (Funktionsbaustein) zur Implementierung einer Server-Methode hat folgenden Aufbau:

1. **Aufruf der Server-Methode abfragen mit OPC_UA_ServerMethodPre**
   In Ihrem Anwenderprogramm (d. h. in Ihrer Server-Methode) rufen Sie zunächst die Anweisung "OPC_UA_ServerMethodPre" auf.
   Diese Anweisung hat folgende Aufgaben:
   - Mit dieser Anweisung fragen Sie beim OPC UA-Server der CPU nach, ob Ihre Server-Methode von einem OPC UA-Client aufgerufen wurde.
   - Wenn die Methode aufgerufen wurde und die Server-Methode über Eingangsparameter verfügt, dann erhält Ihre Server-Methode nun die Eingangsparameter.
     Die Eingangsparameter der Server-Methode stammen vom aufrufenden OPC UA-Client.
2. **Server-Methode bearbeiten**
   In diesem Abschnitt der Server-Methode stellen Sie das eigentliche Anwenderprogramm zur Verfügung.
   Sie haben die gleichen Möglichkeiten wie in anderen Anwenderprogrammen (zum Beispiel Zugriff auf andere Funktionsbausteine oder auf globale Datenbausteine).
   Wenn die Server-Methode Eingangsparameter verwendet, stehen Ihnen die Eingangsparameter der Server-Methode zur Verfügung.
   Dieser Abschnitt der Server-Methode sollte nur dann ausgeführt werden, wenn ein OPC UA-Client die Server-Methode aufgerufen hat.
   Nach der erfolgreichen Ausführung der Methode setzen Sie die Ausgangsparameter der Server-Methode, falls die Server-Methode Ausgangsparameter besitzt.
3. **Server-Methode beantworten mit OPC_UA_ServerMethodPost**
   Um die Server-Methode abzuschließen, rufen Sie die Anweisung "OPC_UA_ServerMethodPost" auf.
   Geben Sie der Anweisung "OPC_UA_ServerMethodPost" über die Parameter die Information mit, ob das Anwenderprogramm abgearbeitet ist oder nicht.
   Wenn das Anwenderprogramm erfolgreich ausgeführt wurde, wird der OPC UA-Server über die einsprechenden Parameter informiert. Der OPC UA-Server sendet nun die Ausgangsparameter der Server-Methode an den OPC UA-Client.

Rufen Sie die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" immer pärchenweise auf, unabhängig davon, ob das Anwenderprogramm zwischen den beiden Anweisungen abgearbeitet oder im nächsten Zyklus fortgesetzt wird.

Ein Beispiel für die Implementierung einer Server-Methode finden Sie in der Onlinehilfe von STEP 7.

Einbindung der Server-Methode

Die folgende Grafik zeigt, wie ein OPC UA-Client (A) die Server-Methode "Cool" aufruft:

Die CPU führt im zyklischen Anwenderprogramm die Instanz "Cool1" der Server-Methode "Cool" aus ⑥.

Die CPU fragt zunächst mit der Anweisung "OPC_UA_ServerMethodPre" nach ④, ob ein OPC UA-Client die Server-Methode "Cool" aufgerufen hat ①.

- Wenn die Server-Methode nicht aufgerufen wurde, kehrt die Programmausführung über ④ und ⑥ direkt zum zyklischen Anwenderprogramm zurück. Die CPU setzt das zyklische Anwenderprogramm nach "Cool1" fort.
- Wenn die Server-Methode aufgerufen wurde, gelangt diese Information über ④ zurück zur Server-Methode Cool. In der Server-Methode Cool wird nun die eigentliche Funktionalität ausgeführt, siehe "<Methoden-Funktionalität>" in der Grafik.
  Anschließend teilt die Server-Methode über die Anweisung "OPC_UA_ServerMethodPost" ⑤ der Firmware (B) mit, dass die Anweisung ausgeführt wurde ③.
  Die Firmware sendet diese Information über ② an den aufrufenden OPC UA-Client (A) zurück.
  Die CPU setzt das zyklische Anwenderprogramm nach "Cool1" fort.

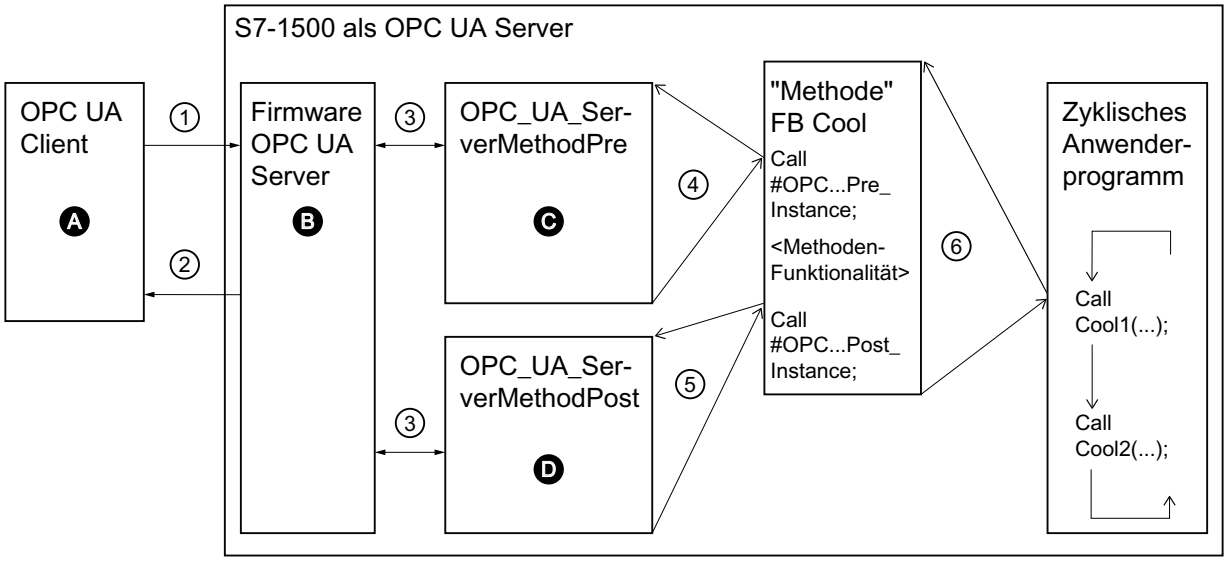

A	Aufruf der Server-Methode und Management der "Done"-Information (Methode beendet)
①	Asynchroner Aufruf der Server-Methode
②	Asynchrone "Done"-Information der aufgerufenen Methode (Methode beendet)
B	Warten auf OPC UA-Client-Aufrufe, Management von Aufrufen in der Warteschlange, "Done"-Information aus dem zyklischen Anwenderprogramm an den OPC UA-Client weiterleiten
③	Datentransfer vom OPC UA-Server zur Methoden-Instanz des Anwenderprogramms und umgekehrt
C	Prüfen, ob Methode aufgerufen wurde. Wenn ja, dann Eingangsdaten vom OPC UA-Server zur Methoden-Instanz des Anwenderprogramms weiterreichen und der Methoden-Instanz zurückmelden, dass die Methode aufgerufen wurde ("called")
④	Synchroner Aufruf der Anweisung OPC_UA_ServerMethodPre als Multiinstanz mit Angabe des Speicherbereichs für die Eingangsdaten vom OPC UA-Server. Der Return-Value informiert, ob die Methode vom OPC UA-Client aufgerufen worden ist.
⑤	Prüfen, ob die Methode beendet wurde oder noch aktiv ist ("busy").
D	Prüfen, ob die Methode beendet wurde. Wenn ja, werden die Ausgangsdaten der Methoden-Instanz zum OPC UA Server weitergeleitet und der Methoden-Instanz zurückgemeldet, dass die Methode beendet ist. Der OPC UA-Server wird darüber informiert.
⑥	Aufruf des Methoden-FB (hier: FB Cool) mit der gewünschten Instanz und den Prozessparametern

Informationen zu den Server-Anweisungen

Die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" sind detailliert beschrieben in der Hilfe zu den Anweisungen > Kommunikation > OPC UA > OPC UA-Server.

Siehe auch

Randbedingungen zum Einsatz von Server-Methoden (Seite 1255)

Randbedingungen zum Einsatz von Server-Methoden

Zulässige Datentypen

Wenn Sie Server-Methoden bereitstellen, dann achten Sie auf folgende Regel:

- Ordnen Sie die Datentypen zu wie unten gezeigt (SIMATIC-Datentyp - OPC UA-Datentyp). Andere Zuordnungen sind nicht zugelassen.

STEP 7 prüft nicht die Einhaltung dieser Regel und verhindert nicht eine falsche Zuordnung. Sie sind für die regelkonforme Auswahl und Zuordnung der Datentypen verantwortlich.

Die aufgelisteten Datentypen können Sie auch z. B. als Elemente von Strukturen/UDTs für Eingangs- und Ausgangsparameter von selbst erstellten Server-Methoden (UAMethod_InParameters und UAMethod_OutParameters) verwenden.

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32
ULINT	UInt64
REAL	Float
LREAL	Double
LDT	DateTime
WSTRING	String

SIMATIC-Datentyp	OPC UA-Datentyp
DINT	Enumeration (Encoding Int32) und alle davon abgeleiteten Datentypen
Anwenderdefinierter Datentyp erforderlich (UDT, user-defined data type) Der Anwenderdefinierte Datentyp muss mit dem Prefix "Union_" angelegt werden, z. B. "Union_MyDatatype". Das erste Element (Selector) in diesem UDT muss den Datentyp "UDINT" besitzen.	UNION und alle davon abgeleiteten Datentypen

Anzahl implementierbarer Server-Methoden und Anzahl der Argumente

Wenn Sie Server-Methoden über Ihr Anwenderprogramm implementieren, dann ist die Anzahl nutzbarer Methoden je nach CPU-Typ begrenzt, siehe folgende Tabelle.

Technisches Datum	CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)	CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)	CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)
Maximale Anzahl nutzbarer Server-Methoden bzw. max. Anzahl Server-Methoden-Instanzen (Anweisungen OPC_UA_ServerMethodPre, OPC_UA_ServerMethodPost)	20	50	100
Maximale Anzahl Argumente pro Methode (Mehr als die angegebene Anzahl Argumente ist projektierbar und ladbar in die CPU, allerdings kann ein OPC UA-Client die Methode dann nicht aufrufen.)	20	20	20

Fehlermeldung bei Überschreitung

Wenn die maximale Anzahl von Server-Methoden überschritten ist, melden die Anweisungen OPC_UA_ServerMethodPre bzw. OPC_UA_ServerMethodPost den Fehlercode 0xB080_B000 (TooManyMethods).

Versorgung von strukturierten Datentypen mit geschachtelten Arrays

Wenn ein strukturierter Datentyp (Struct/UDT) ein Array enthält, stellt der OPC UA-Server keine Information über die Länge dieses Arrays bereit.

Falls Sie eine solche Struktur z. B. als Eingangs- oder Ausgangsparameter einer Server-Methode verwenden, dann müssen Sie sicherstellen, dass das geschachtelte Array beim Aufruf der Methode mit der richtigen Länge versorgt wird.

Wenn Sie diese Regel nicht berücksichtigen, schlägt die Methode fehl mit dem Fehlercode "BadInvalidArgument".

OPC UA-Server-Schnittstelle projektieren

Was ist eine Server-Schnittstelle?

Definition

Eine Server-Schnittstelle fasst zusammengehörende Knoten eines OPC UA-Adressraums einer CPU zu einer Einheit zusammen, sodass eine bestimmte Sicht auf diese CPU für OPC UA-Clients bereitgestellt wird.

1. Beispiel:

Eine CPU steuert eine Spritzgießmaschine.

In diesem Beispiel enthält eine Server-Schnittstelle alle OPC UA-Knoten der CPU,

- die Sie mit einem OPC UA-Client lesen können, um Informationen über diese Spritzgießmaschine zu erhalten (aus lesbaren PLC-Variablen),
- die Sie mit einem OPC UA-Client schreiben können, um Werte in die Spritzgießmaschine zu übertragen (in schreibbare PLC-Variablen),
- die Sie mit einem OPC UA-Client aufrufen können, um Funktionen der Spritzgießmaschine zu starten (über Server-Methoden).

Diese Server-Schnittstelle erlaubt die Sicht auf eine CPU, die als Steuerung einer Spritzgießmaschine dient.

Für Spritzgießmaschinen definiert die Companion-Spezifikation "Euromap" eine ganze Reihe von OPC UA-Knoten, die Sie in einer Server-Schnittstelle zusammenfassen können.

Andere OPC UA-Knoten der CPU sind nicht in dieser Sicht enthalten.

Das erhöht den Überblick.

Das Kapitel "Server-Schnittstelle anlegen (Seite 1258)" zeigt, wie Sie eine Server-Schnittstelle erzeugen und dieser Server-Schnittstelle OPC UA-Knoten hinzufügen, die zusammengehören.

Als Beispiel für zusammengehörende OPC UA-Knoten dient die "Euromap77".

2. Beispiel

Anlagen- oder Kunden-spezifische Aufgaben

Sie können auch selbst eine Server-Schnittstelle definieren, die zum Beispiel den Anforderungen in einem kundenspezifischen Projekt entspricht.

Dazu legen Sie in einer XML-Datei alle erforderlichen OPC UA-Knoten fest.

Verwenden Sie zu diesem Zweck ein Tool, zum Beispiel das kostenlose Programm "SiOME" von Siemens, Erläuterungen und Download-Link finden Sie hier (https://support.industry.siemens.com/cs/ww/de/view/109755133).

Server-Schnittstelle anlegen

Server-Schnittstellen anlegen

Um eine Server-Schnittstelle zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Projektansicht im TIA Portal
2. Wählen Sie die CPU aus, die Sie als OPC UA-Server nutzen.
3. Klicken Sie auf "OPC UA-Kommunikation > Server-Schnittstellen".
4. Doppelklicken Sie auf "Neue Server-Schnittstelle hinzufügen".
   STEP 7 erzeugt eine neue Server-Schnittstelle, nennt sie "Server-Schnittstelle_1" und öffnet den Editor für Server-Schnittstellen.
5. Ändern Sie den Namen der neuen Server-Schnittstelle, sodass er in Ihrem Projekt aussagekräftig ist.
6. Klicken Sie rechts oben in der Ansicht für Server-Schnittstellen auf die Schaltfläche "Companion-Spezifikation importieren".
   STEP 7 zeigt den Dialog "Importieren" an.
7. Wählen Sie eine XML-Datei aus, die eine Instanz (oder mehrere Instanzen) einer Companion-Spezifikation enthält.
   Wie Sie eine solche XML-Datei anlegen, beschreibt das Kapitel "OPC UA Companion Spezifikationen verwenden (Seite 1260)".
   STEP 7 importiert die XML-Datei und stellt die neue Server-Schnittstelle im Editor dar.
   Das folgende Bild zeigt einen Ausschnitt aus einer Server-Schnittstelle, die eine Instanz der Companion-Spezifikation "Euromap" verwendet:

Schnittstelle exportieren | Konsistenzcheck | Partnerspezifikation importieren

OPC UA-Server-Schnittstelle

Name des Knotens	Knotentyp	Zugriffsstufe	Beschreibung	Lokaldaten	Datentyp
IMM_manufacturer_01234	Object				
InjectionUnits	Object				
InjectionUnit_1	Object				
TemperatureZones	Object				
BarrelTemperatureZone_1	Object				
ActualTemperature	Double	RD	The ActualT...	InjectionUnits.i...	LReal
EngineeringUnits	EUInformation	RD	Show Place...		
EURange	Range	RD	Range for t...		
NominalTemperature	Double	RD	The nomina...	InjectionUnits.i...	LReal
EngineeringUnits	EUInformation	RD			
EURange	Range	RD			
Index	UInt32	RD	The Index P...	InjectionUnits.i...	UDInt
IsPresent	Boolean	RD	TRUE if pres...	InjectionUnits.i...	Bool
Name	String	RD	The Name P...	InjectionUnits.i...	WString
TemperatureConfigura...	TemperatureConf...	RD	The Temper...	InjectionUnits.i...	DInt

Informationen zur Server-Schnittstelle

Der Editor für Server-Schnittstellen ist als Tabelle aufgebaut und stellt eine Reihe von Informationen über die importierte Nodeset-Datei (OPC UA-XML-Datei) bereit.

Die Tabelle stellen die folgenden Angaben bereit:

- Name des Knotens
  Der oberste Knoten trägt im Beispiel den Namen "IMM_manufacturer_01234".
  Dieser Name steht für die Spritzgießmaschine als Ganzes. Er ist der Name der hier verwendeten Instanz der Companion-Spezifikation "Euromap".
  Gemäß der Companion-Spezifikation soll der Instanzname mit "IMM" beginnen; dann folgt der Name des Herstellers der Spritzgießmaschine; am Ende wird die Seriennummer der Maschine angefügt. Dadurch lässt sich eine Maschine eindeutig identifizieren.
  Der Name dieses Knoten wird vom Anwender vergeben.
  Die Namen aller anderen (unterlagerten) Knoten sind durch die Spezifikation festgelegt (im Beispiel oben durch die Companion-Spezifikation "Euromap"). Diese Knotennamen dürfen vom Anwender nicht geändert werden. Das sichert eine einheitliche Sicht auf alle Spritzgießmaschinen, die sich an die Spezifikation halten.
- Knotentyp
  Typ des Knotens. Der Typ ist durch die verwendete Companion-Spezifikation festgelegt.
  STEP 7 zeigt einen Knotentyp in der Tabelle rot an, wenn in der importierten XML-Datei keine Typ-Definition enthalten ist.
  In diesem Fall finden Sie in den Eigenschaften der Server-Schnittstelle unter "Namespaces" die referenzierten, aber fehlenden Namensräume.
  Importieren Sie die referenzierten XML-Dateien, in denen die fehlenden Namensräume (und alle darin definierten Datentypen) enthalten sind, in einer neu anzulegenden Server-Schnittstelle.
- Zugriffstufe
  Knoten (Variablen) können nur lesbar sein (RD) oder les- und schreibbar (RD/WR).
  Die Server-Methoden einer Spezifikation sind immer aufrufbar. Wenn sie nicht aufrufbar wären, wären sie nicht in die Spezifikation aufgenommen worden.
- Beschreibung
  Die importierte XML-Datei kann Beschreibungen der Knoten in mehreren Sprachen enthalten.
  Wenn Sie eine solche Datei importiert haben, dann zeigt STEP 7 die Beschreibung in der Projektsprache an.
  Falls in der XML-Datei keine Beschreibung in der Projektsprache vorhanden ist, dann wird die englische Beschreibung angezeigt.
- Lokaldaten
  STEP 7 zeigt den Datenbaustein an, der dem Knoten in der CPU zugeordnet ist (Mapping).
  Von diesem Datenbaustein liest die CPU den Wert des OPC UA-Knotens.
  Der OPC UA-Server stellt dann diesen Wert für OPC UA-Clients zur Verfügung.
  Wenn ein Datenbaustein rot angezeigt wird, dann ist der angegebene Datenbaustein in der CPU nicht vorhanden.
  In diesem Fall müssen Sie den fehlenden Datenbaustein in der CPU (im Anwenderprogramm) anlegen und mit einem Wert versorgen.
  Alternative: Sie ändern in der XML-Datei das Mapping des OPC UA-Knotens auf einen lokalen Datenbaustein (oder auf eine Variable einer Variablentabelle Ihres Anwenderprogramms).
  Wie Sie das Mapping ändern, beschreibt das Kapitel "OPC UA Companion Spezifikationen verwenden (Seite 1260)".
- Datentyp
  Das ist der Datentyp des Datenbausteins in der CPU, von dem der Wert eines OPC UA-Knotens gelesen wird, oder dem ein Wert zugeschrieben wird.

Konsistenzcheck

Sie haben die Möglichkeit, die Konsistenz der Server-Schnittstelle zu prüfen. Dabei prüft STEP 7, ob die Knoten (z. B. Variablen) in der importierten Nodeset-Datei (OPC UA-XML-Datei) korrekt gemappt sind.

1. Markieren Sie den Koten, den Sie prüfen wollen
2. Klicken Sie auf die Schaltfläche "Konsistenzcheck".

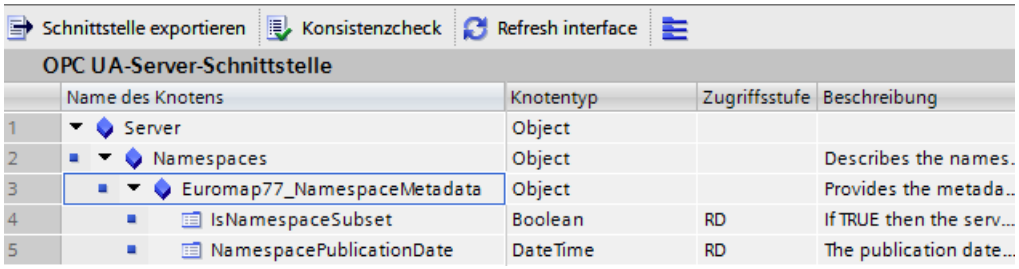

Der Konsistenzcheck prüft das Mapping der Lokaldaten gegen die angezeigte Server-Schnittstelle:

– Entsprechen die Namen der PLC-Variablen im Programm der CPU denen der importierten Server-Schnittstelle?

– Entsprechen die Datentypen der PLC-Variablen den entsprechenden Knoten in der Server-Schnittstelle? STEP 7 zeigt entsprechende Warnungen und Fehler an, wenn die oben genannten Bedingungen nicht erfüllt sind.

OPC UA Companion Spezifikationen verwenden

Einleitung

OPC UA ist universell einsetzbar: Der Standard selbst macht z. B. keine Aussagen darüber, wie PLC-Variablen zu benennen sind. Auch steht es im Ermessen des einzelnen Nutzers (Anwendungsentwicklers), Server-Methoden zu programmieren und zu benennen, die über OPC UA aufrufbar sind.

Informationsmodellierung und Standardisierung für Geräte und Branchen

Für gleichartige Anwendungen bietet es sich an, mit dem "OPC UA-Baukasten" sein Geräteinterface oder Maschineninterface zu standardisieren.

Viele Gremien und Arbeitskreise haben die Standardisierung vorangetrieben und unterschiedliche Companion Spezifikationen erarbeitet.

In diesen Spezifikationen ist festgelegt:

- Mit welchen Objekten, Methoden und Variablen ist ein typisches Gerät oder eine typische Maschine zu beschreiben.
- Welcher Namespace ist für die genannten Objekte vorgesehen.

Maschinen werden dabei typischerweise in funktionale bzw. technologische Einheiten strukturiert und diese Einheiten standardisiert.

Den Betreibern von Maschinen und Anlagen bieten Companion Spezifikationen den Vorteil einer einheitlichen Schnittstelle. So sind zum Beispiel alle RFID-Reader, welche die Spezifikation AutoID einhalten, auf die gleiche Weise integrierbar. Das heißt, alle RFID-Reader, die konform der Spezifikation AutoID sind, lassen sich auf die gleiche Weise von OPC UA-Clients ansprechen, unabhängig vom Hersteller der RFID-Reader.

Ein anderes Beispiel für eine Companion Spezifikation aus dem Bereich Spritzgießmaschinen ist die Euromap 77 Companion Spezifikation.

Das folgende Kapitel beschreibt am Beispiel von Euromap 77, wie Sie eine Companion Spezifikation in STEP 7 (TIA Portal) übernehmen und dafür PLC-Variablen anlegen.

Beispiel Euromap 77

Euromap 77 standardisiert den Datenaustausch zwischen Spritzgießmaschinen und dem übergelagerten MES (manufacturing execution system). Dadurch kann das MES alle unterlagerten Spritzgießmaschinen gleich anbinden.

Die einheitliche Datenschnittstelle erleichtert die Integration von Spritzgießmaschinen in eine Anlage.

Companion Spezifikation verwenden: Übersicht

Die Euromap 77 ist in der OPC UA-XML-Datei "Opc_Ua.EUROMAP77.NodeSet2.xml" beschrieben.

Hinweis

Euromap 77, Euromap 83 und OPC UA for Devices (DI)

Mit dem Release Candidate 2 wurde ein Teil der Definitionen von Euromap 77 nach Euromap 83 übertragen. Deshalb müssen Sie auch die OPC UA-Server-Schnittstelle von Euromap 83 importieren.

"OPC UA for Devices" ist ein allgemein gültiges Informationsmodell für die Konfiguration von Hardware- und Softwarekomponenten. Das Informationsmodell dient auch als Basis für weitere Companion Standards und wird daher ebenfalls importiert.

Sie erhalten die OPC UA-XML-Dateien hier:

Euromap77 (http://www.euromap.org/euromap77)

Euromap83 (http://www.euromap.org/euromap83)

OPC UA for Devices (https://opcfoundation.org/UA/schemas/DI/)

Die Euromap-OPC UA-XML-Dateien definieren die Schnittstelle des OPC UA-Servers einer Spritzgießmaschine, die konform zum Euromap 77 Companion Standard ist.

Um den OPC UA-Server der S7-1500 CPU nach dem Euromap 77 Standard zu modellieren, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine XML-Datei, in der Sie eine Instanz des Typs "IMM_MES_InterfaceType" anlegen.
   "IMM_MES_InterfaceType" ist der oberste Knoten in der Euromap 77: Dieser Datentyp ist direkt vom OPC UA-Datentyp "BaseObjectType" abgeleitet.
   Unterhalb von "IMM_MES_InterfaceType" sind alle Variablen und Methoden der Euromap 77 (83) definiert.
   Wie Sie im Einzelnen vorgehen, ist unter "Schritt 1" beschrieben.
2. Ordnen Sie den Variablen und Methoden der Euromap 77 (dem Informationsmodell der Euromap 77) PLC-Variablen und FB-Instanzen (Server-Methoden) Ihrer S7-1500 CPU zu.
   Wie Sie im Einzelnen vorgehen, ist unter "Schritt 2" beschrieben.
3. Importieren Sie die XML-Datei als "Server-Schnittstelle".
   Wie Sie im Einzelnen vorgehen, ist unter "Schritt 3" beschrieben.
   Übersetzen Sie das STEP 7-Projekt.
   Laden Sie das Projekt in die CPU, die als Steuerung für eine Spritzgießmaschine fungiert.
4. Legen Sie in Ihrem STEP 7-Projekt die PLC-Variablen und Server-Methoden an, denen Sie unter Punkt 2 Variablen und Methoden der Euromap 77 zugeordnet haben.
   Die PLC-Variablen müssen kompatible Datentypen besitzen, siehe unten "Mapping der Datentypen".
   Erlauben Sie den Lese- und Schreibzugriff von OPC UA-Clients auf diese PLC-Variablen entsprechend Euromap 77.
   Legen Sie die PLC-Variablen z. B. in einem Datenbaustein an.
   Wie Sie im Einzelnen vorgehen, ist unter "Schritt 4" beschrieben.

Ergebnis: Die Variablen und Server-Methoden nach Euromap 77 sind im Adressraum des OPC UA-Servers Ihrer CPU für OPC UA-Clients verfügbar.

Schritt 1: Instanz erstellen

Im folgenden ist beschrieben, wie Sie das kostenlose Programm "SiOME" nutzen, den "Siemens OPC UA Modelling Editor".

Sie können mit SiOME eine OPC UA-XML-Datei erstellen, die eine Server-Schnittstelle (ein Informationsmodell) beschreibt.

Um die neue Schnittstelle in den OPC UA-Server einer S7-1500-CPU zu laden und zu veröffentlichen, importieren Sie die Server-Schnittstelle in das STEP 7-Projekt, siehe Kapitel "Server-Schnittstelle anlegen (Seite 1258)".

Wenn das Projekt in die CPU geladen ist, steht die neue Server-Schnittstelle für OPC UA-Clients bereit.

Andere Tools zum Entwerfen von Informationsmodellen

Sie können auch mit anderen Tool zum Entwerfen von Informationsmodellen arbeiten, um ein Server-Schnittstelle zu erstellen, z.B. mit dem Programm "UaModeler" von Unified Automation. Sie können das Programm "UaModeler" von hier (https://www.unified-automation.com/downloads/opc-ua-development.html)laden.

Die Tools zum Entwerfen von Informationsmodellen werden ständig weiterentwickelt. Nutzen Sie daher für die Handhabung immer die vom Hersteller zur Verfügung gestellte Dokumentation.

SiOME

Mit Hilfe von SiOME (Siemens OPC UA Modeling Editor), ein Tool zur Umsetzung von OPC UA Companion Spezifikationen, entwerfen Sie Informationsmodelle/Adressräume für Ihren OPC UA-Server und können neue Typen und Instanzen von OPC UA-Nodes erstellen.

Ferner können Sie SiOME dazu nutzen, um UA-Variablen auf PLC-Variablen und UA-Methoden auf PLC-Funktionsbausteine (Instanzen) zu mappen.

Download-Link und Erläuterungen zu SIOME finden Sie hier (https://support.industry.siemens.com/cs/ww/de/view/109755133).

Um eine XML-Datei mit einer Instanz von "IMM_MES_InterfaceType" anzulegen, gehen Sie folgendermaßen vor:

1. Laden Sie die Dateien "Opc_Ua.EUROMAP77.NodeSet2.xml" und "Opc_Ua_EUROMAP83_NodeSet2.xml" von der Euromap-Webseite (siehe oben).
2. Laden Sie die Datei "Opc.Ua.Di.NodeSet2.xml" von der OPC Foundation-Website. Euromap77/83 ist eine OPC UA Companion Spezifikation. d. h. sie baut auf OPC UA auf und verwendet Datentypen, die in der Datei "Opc.Ua.Di.NodeSet2.xml" definiert sind.
3. Starten Sie SiOME.
4. Importieren Sie zunächst den Namensraum "http://opcfoundation.org/UA/DI/"
   Dazu klicken Sie im Bereich "Information model" auf die Schaltfläche "Import XML".

   Wählen Sie die Datei "Opc.Ua.Di.NodeSet2".xml" aus und klicken Sie auf die Schaltfläche "Öffnen", um die Datei zu importieren.
   SiOME importiert die XML-Datei und zeigt im Bereich "Namespaces" den Namensraum "http://opcfoundation.org/UA/DI/" an.
   Der Standard-Namensraum "http://opfoundation.org/UA/" ist in SiOME immer vorhanden und muss nicht importiert werden.
5. Importieren Sie nun den Namensraum "http://www.euromap.org/euromap83/"
   Dazu klicken Sie wieder im Bereich "Information model" auf die Schaltfäche "Import XML".
   Wählen Sie die Datei "Opc_Ua.EUROMAP83.NodeSet2.xml" aus.
   SiOME importiert die XML-Datei und zeigt im Bereich "Namespaces" nun auch den Namensraum "http://www.euromap.org/euromap83/" an.
6. Importieren Sie nun den Namensraum "http://www.euromap.org/euromap77/"
   Dazu klicken Sie wieder im Bereich "Information model" auf die Schaltfläche "Import XML".
   Wählen Sie die Datei "Opc_Ua.EUROMAP77.NodeSet2.xml" aus.
7. **Eigenen Namensraum definieren und Instanz bilden**
   Bislang haben Sie Euromap77 importiert.
   Nun verwenden Sie die Companion-Spezifikation für eine Spritzgießmaschine, d. h, Sie bilden eine Instanz von Euromap77 (eine Verwendung).
   Klicken Sie dazu im Bereich "Namespaces" mit der rechten Maustaste auf "Models" und wählen Sie "Add Model".
   SiOME öffnen den Dialog "Add Model".

8. Klicken Sie auf "New Model", um ein neues Model erstellen zu können.
   SiOME zeigt nun den Schalter vor "New Model" grün an:

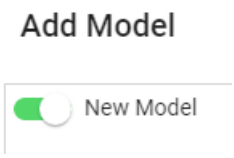

9. Fügen Sie der XML-Datei einen eigenen Namensraum hinzu.
   Dazu verwenden Sie einen eindeutigen Namen.
   Im Beispiel tragen Sie "YourCompany.com" ein:

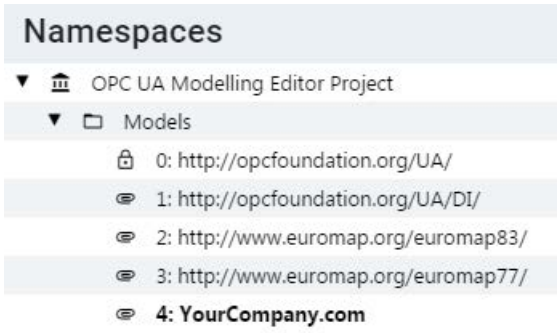

10. **Neue Instanz anlegen**
    Legen Sie nun eine neue Instanz der Euromap77 an. Damit verwenden Sie die Euromap77 für eine Spritzgießmaschine.
    Dazu klicken Sie im Bereich "Information model" mit der rechten Maustaste auf "Objects".
    SiOME zeigt ein Kontextmenu an.

11. Klicken Sie auf "Add Instance", um eine neue Instanz hinzuzufügen.
    SiOME zeigt den Dialog "Add Instance" an.
    Fügen Sie bei "Name" einen aussagekräftigen Namen für Ihre Instanz ein, d. h. für die Maschine, für die Sie Euromap77 verwenden.
    Im Beispiel tragen Sie "IMM_Manufacturer_01234" ein.
    Bei "TypeDefinition" wählen Sie "IMM_MES_InterfaceType".
    Klicken Sie auf "OK".
    SiOME zeigt die neue Instanz im Bereich "Information model" unter "Objects" an:

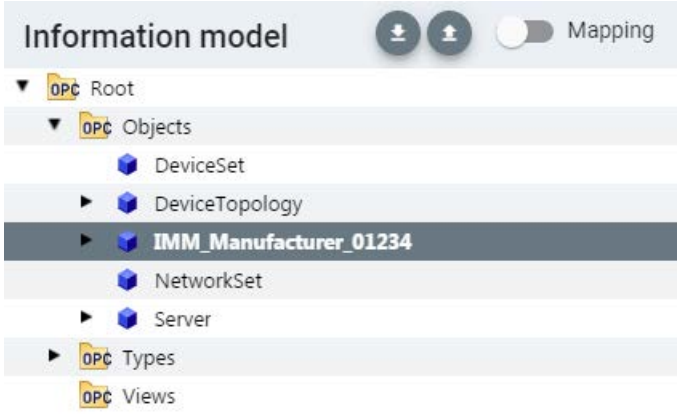

12. Klicken Sie auf den schwarzen Pfeil vor ""IMM_Manufacturer_01234", um die Instanz zu öffen:

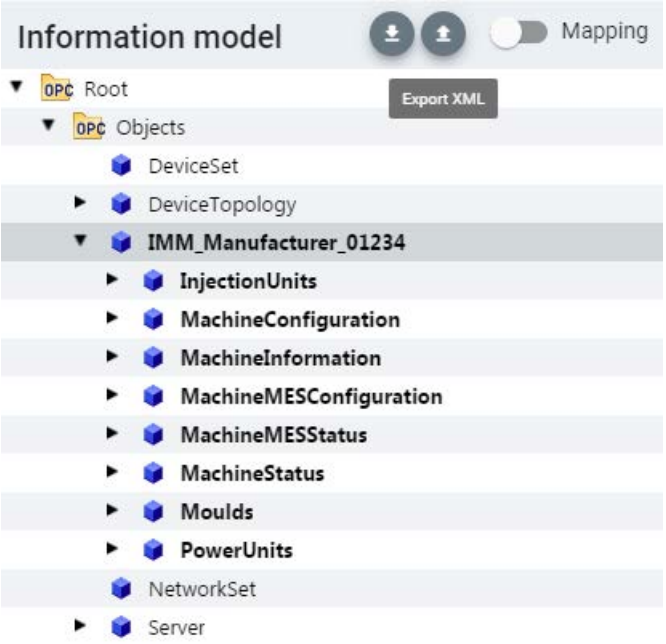

13. Legen Sie nun eine neue Instanz für "InjectionUnits" an.
    Dazu klicken Sie mit der rechten Maustaste auf "InjectionUnits".
    SiOME zeigt den Dialog "Add Instance" an.
    Fügen Sie bei "Name" einen aussagekräftigen Namen für Ihre Instanz ein.
    Im Beispiel tragen Sie "InjectionUnit_1" ein.
    Bei "TypeDefinition" wählen Sie "InjectionUnitType".
    Klicken Sie auf "OK".

14. Legen Sie nun jeweils eine neue Instanz für "Moulds" und "PowerUnits" an, wie oben für "InjectionUnits"gezeigt.

15. Speichern Sie nun die XML-Datei.
    Dazu exportieren Sie die Datei.
    Klicken Sie im Bereich "Information model" auf die Schaltfläche "Export XML".

    SiOME blendet den Dialog "Export XML" ein.

16. Lassen Sie die Option "Include mappings" nicht aktiviert, da in der Datei noch keine Mappings enthalten sind (UAVariablen auf PLC-Variablen, UAMethoden auf Server-Methoden der CPU).
    Die importieren Namensräume sind für Euromap77 erforderlich. Deshalb stellt SiOME sie grau dar.
    Klicken Sie auf die Schaltfläche "OK".
    SiOME blendet einen Dialog ein.

17. Wählen Sie einen aussagekräftigen Namen für die XML-Datei und speichern Sie die Datei.

Ergebnis:

Sie haben nun eine XML-Datei erstellt, die die Companion Spezifikation "Euromap77" und die von Ihnen hinzugefügten Instanzen enthält. Damit verwenden Sie die Euromap 77.

Schritt 2: PLC-Variablen zuordnen

Die folgende Beschreibung zeigt am Beispiel einer Variablen, wie Sie den Variablen und Methoden der Euromap 77 bestimmte PLC-Variablen und Server-Methoden zuordnen.

Der einfache Weg

Wir empfehlen, auch hier das Tool SiOME zu verwenden. In SiOME müssen Sie dazu das Mapping freischalten:

Im Bereich "TIA Portal" wählen Sie dann die Datenbausteine aus, deren Variablen Sie den OPC UA Variablen des Informationsmodells zuordnen ("mappen") wollen.

Die Variablen aus dem TIA Portal-Projekt können Sie einfach mit dem OPC UA-Variablen per Drag & Drop verschalten.

Um Methoden zu verknüpfen, müssen Sie die Instanzdatenbausteine der entsprechenden TIA Portal-Funktionsbausteine per Drag & Drop mit den Methoden des Informationsmodells verknüpfen.

Beim Export wählen Sie die Opition "Include mappings", da hier das Informationsmodell für einen OPC UA Server benötigt wird.

Die Optionwird nicht benötigt, wenn Sie das Informationsmodell für einen OPC UA Client benötigen.

Die detaillierte Beschreibung finden Sie beim Anwendungsbeispiel im Online Support (SiOME (https://support.industry.siemens.com/cs/ww/de/view/109755133)).

PLC-Variablen manuell zuordnen

Falls Sie die Daten nicht mit SiOME zuordnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie mit einem Editor die Datei "demo.xml", die Sie im "Schritt 1" erstellt haben. Suchen Sie in der XML-Datei, die UA-Variable mit der NodeId="ns=1;s=MyIMM_MES_Interface.InjectionUnits.NodeVersion". Dieser Variablen ordnen Sie im nächsten Schritt eine PLC-Variable zu.
2. Um dieser OPC UA-Variablen eine PLC-Variable zuzuordnen, fügen Sie in das XML-Element der OPC UA-Variablen eine Extension ein. In diese Extension wiederum fügen Sie ein Element des Typs "<si:VariableMapping>" ein:

```
<UAVariable DataType="String" ParentNodeId="ns=1;s=MyIMM_MES_Interface.InjectionUnits" NodeId=
"ns=1;s=MyIMM_MES_Interface.InjectionUnits.NodeVersion" BrowseName="NodeVersion">
    <DisplayName>NodeVersion</DisplayName>
    <References>
        <Reference ReferenceType="HasProperty" IsForward="false">
        ns=1;s=MyIMM_MES_Interface.InjectionUnits</Reference>
        <Reference ReferenceType="HasTypeDefinition">i=68</Reference>
    </References>
    <Extensions>
        <Extension>
            <si:VariableMapping>"MyIMM_MES_Interface"."InjectionUnits.NodeVersion"
            </si:VariableMapping>
        </Extension>
    </Extensions>
    <Value>
</UAVariable>
```

3. Das XML-Element "<si:VariableMapping>" ist im XML-Namespace "http://www.siemens.com/OPCUA/2017/SimaticNodeSetExtensions" definiert. Deshalb müssen Sie dem XML-Element "<UANodeSet>" diesen Namensraum hinzufügen, z. B. einmalig durch folgende Codezeile zu Beginn der XML-Datei:

   ```
   <UANodeSet xmlns:si="http://www.siemens.com/OPCUA/2017/SimaticNodeSetExtensions"
   ```

   Wenn Sie den Namensraum nicht hinzufügen, dann ist das Element "<si:VariableMapping>" in der Datei nicht bekannt.
4. Speichern Sie die Datei "demo.xml" und schließen Sie den Editor.

Exkurs: PLC-Methoden zuordnen

Neben Variablen können Sie auch Methoden einer FB-Instanz (Anwenderprogramm bzw. Funktionsbaustein als Repräsentation der Methode) zuordnen.

Um einer OPC UA-Methode eine FB-Instanz zuzuordnen, müssen Sie ebenfalls eine Extension nach dem unten gezeigten Muster in der OPC UA-XML-Datei ergänzen. Beachten Sie, dass das Suffix ".Method" ohne Anführungszeichen dem Instanznamen angehängt werden muss.

In der Extension fügen Sie ein Element des Typs "<si:MethodMapping>" ein.

Die Eigenschaften einer OPC UA-Methode mit BrowseName "InputArguments" und "OutputArguments" sind OPC UA-Variablen und werden **nicht** zugeordnet.

```
<UAMethod ParentNodeId="ns=1;i=5003" NodeId=
"ns=1;s=MyIMM_MES_Interface.MachineConfiguration.SetServerTime" BrowseName="4:SetServerTime"
MethodDeclarationId="ns=2;i=7005">
    <DisplayName>SetServerTime</DisplayName>
    <References>
        <Reference ReferenceType="HasProperty">
        ns=1;s=MyIMM_MES_Interface.MachineConfiguration.SetServerTime.InputArguments</Reference>
        <Reference ReferenceType="HasComponent" IsForward="false">ns=1;i=5003</Reference>
    </References>
    <Extensions>
        <Extension>
            <si:MethodMapping>"MachineConfiguration.SetServerTime_IDB".Method</si:MethodMapping>
        </Extension>
    </Extensions>
</UAMethod>
```

Schritt 3: Server-Schnittstellen importieren

Um eine OPC UA-XML-Datei als "Server-Schnittstelle" zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das STEP 7-Projekt.
2. Klicken Sie auf "OPC UA-Kommunikation > Server-Schnittstellen".
3. Doppelklicken Sie auf "Server-Schnittstelle importieren".
4. Im Dialog "Importieren" wählen Sie die Datei, die Sie als Server-Schnittstelle importieren wollen.
   Im Beispiel ist das die Datei "demo.xml".
5. Klicken Sie auf die Schaltfläche "Import".
6. Importieren Sie auch die Dateien "Opc.Ua.Di.NodeSet2.xml", "Opc_Ua.EUROMAP77.NodeSet2.xml" und "Opc_Ua.EUROMAP83.NodeSet2.xml".
   Die Datei "demo.xml" bezieht sich auf die erstgenannten Dateien.
   Verwenden Sie stets aktuelle Versionen dieser Dateien, da STEP 7 keine fehlerhaften OPC UA-XML-Dateien übersetzt.

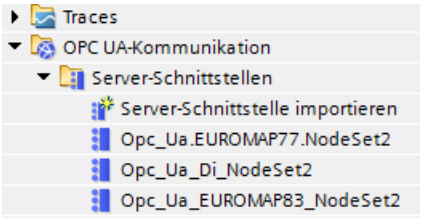

Die oben nach Euromap 77 Spezifikation erzeugte Abbildung der Maschine ist nun im Adressraum des OPC UA-Servers vorhanden.

Schritt 4: Im STEP 7-Projekt PLC-Variablen und Server-Methoden anlegen

Im Beispiel haben Sie der UA-Variablen mit der NodeId="ns=1;s=MyIMM_MES_Interface.InjectionUnits.NodeVersion" die PLC-Variable "MyIMM_MES_Interface"."InjectionUnits.NodeVersion" zugeordnet, siehe "Schritt 2: PLC-Variablen zuordnen".

Um die benötigte PLC-Variable mit STEP 7 (TIA Portal) anzulegen, gehen Sie folgendermaßen vor:

1. Erzeugen Sie einen Datenbaustein "MyIMM_MES_Interface".
2. Erzeugen Sie das DB-Element "InjectionUnits.NodeVersion". Verwenden Sie den kompatiblen SIMATIC-Datentyp zum OPC UA-Datentyp. Die UA-Variable des Beispiels besitzt den OPC UA-Datentyp "String" (DataType="String"). Der kompatible SIMATIC-Datentyp ist WSTRING.

MyIMM_MES_Interface

	Name	Datentyp	Startwert	Erreichbar aus HMI/OPC UA	Schreibbar aus HMI/OPC UA
	Static			☐	☐
	InjectionUnits.NodeVersion	WString	WSTRING#'V2.5'	☑	☐

3. Übersetzen Sie das Projekt.
4. Laden Sie das Projekt in die CPU.

Import von exportieren OPC UA-XML-Dateien in eine S7-1500 CPU

Beachten Sie folgenden Hinweis, wenn Sie Server-Schnittstellen importieren, die aus dem OPC UA-XML-Export einer S7-1500 stammen.

Hinweis

Gesperrter Import für Namensraum "http://www.siemens.com/simatic-s7-opcua"

Sie können keine Server-Schnittstelle mit dem Namensraum "http://www.siemens.com/simatic-s7-opcua" in eine S7-1500 CPU importieren, da dieser Namensraum für S7-1500 CPUs reserviert (Standard-SIMATIC Server-Schnittstelle) und für den Import gesperrt ist.

Wenn Sie eine Server-Schnittstelle mit dem Namensraum "http://www.siemens.com/simatic-s7-opcua" importieren wollen, dann öffnen Sie die zu importierende Server-Schnittstelle (OPC UA-XML-Datei) und ändern den Namensraum an den entsprechenden Stellen. Die so geänderte Datei können Sie dann importieren.

Integrität der OPC UA-XML-Dateien

OPC UA-XML-Dateien repräsentieren den Server-Adressraum. Diese Dateien werden z. B. im Kontext von OPC UA Companion Spezifikationen von Ihnen nach Anpassung an die Applikation als Server-Schnittstelle importiert, mit der Hardware-Konfiguration in die S7-1500 CPU geladen und getestet.

⚠ WARNUNG

Keine Prüfung von importierten OPC UA-XML-Dateien

Schützen Sie diese OPC UA-XML-Dateien vor nicht autorisierten Manipulationen, da STEP 7 die Integrität dieser Dateien nicht prüft.

Empfehlung

Um bei einer Erweiterung bzw. Anpassung des Server-Adressraums Risiken zu minimieren, gehen Sie folgendermaßen vor:

1. Schützen Sie das Projekt (Projektnavigation: Security-Einstellungen > Einstellungen).
2. Exportieren Sie vor der Erweiterung oder Anpassung die entsprechende Server-Schnittstelle.
3. Überarbeiten Sie diese OPC UA-XML-Datei.
4. Importieren Sie die Datei erneut als Server-Schnittstelle.

Mapping der Datentypen

Die folgende Tabelle zeigt den kompatiblen SIMATIC-Datentyp zum jeweiligen OPC UA-Datentyp.

Ordnen Sie die Datentypen zu wie unten gezeigt (SIMATIC-Datentyp - OPC UA-Datentyp). Andere Zuordnungen sind nicht zugelassen. STEP 7 prüft nicht die Einhaltung dieser Regel und verhindert nicht eine falsche Zuordnung. Sie sind für die regelkonforme Auswahl und Zuordnung der Datentypen verantwortlich.

Die aufgelisteten Datentypen können Sie auch z. B. als Elemente von Strukturen/UDTs für Eingangs- und Ausgangsparameter von selbst erstellten Server-Methoden (UAMethod_InParameters und UAMethod_OutParameters) verwenden.

Tabelle 1-89 Mapping der Datentypen

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32
ULINT	UInt64
REAL	Float
LREAL	Double
LDT	DateTime
WSTRING	String

SIMATIC-Datentyp	OPC UA-Datentyp
DINT	Enumeration (Encoding Int32) und alle davon abgeleiteten Datentypen
Anwenderdefinierter Datentyp erforderlich (UDT, user-defined data type) Der anwenderdefinierte Datentyp muss mit dem Präfix "Union_" angelegt werden, z. B. "Union_MyDatatype", siehe Beispiel unter der Tabelle. Das erste Element (Selector) in diesem UDT muss den Datentyp "UDINT" besitzen.	UNION und alle davon abgeleiteten Datentypen

Anwenderdefinierter Datentyp für UNION erforderlich

Das folgende Bild zeigt die Variable "MyVariable", die den Datentyp "Union_MyDatatype" besitzt.

Dieser SIMATIC-Datentyp entspricht einer OPC UA-Variaben mit dem Datentyp UNION.

Das Bild zeigt ein Beispiel für die Deklaration: Bei Selector = 1, nimmt die Union einen ByteArray auf, bei Selector = 2 einen WString.

Name	Datentyp
Static	
Selector	UDInt
ByteArray	Array[0..1] of Byte
WString	WString[42]

Weitere OPC UA Basisdatentypen nutzen

Neben den im Abschnitt "Mapping von Datentypen" aufgeführten OPC UA-Datentypen und deren Entsprechungen auf SIMATIC-Seite gibt es noch folgende OPC UA-Basisdatentypen, die Sie ebenfalls nutzen können:

- OpcUa_NodeId
- OpcUa_QualifiedName
- OpcUa_Guid
- OpcUa_LocalizedText
- OpcUa_ByteString
- OpcUa_XmlElement

Voraussetzung für die Nutzung der oben genannten Basisdatentypen als Variablen im Anwenderprogramm: Die Basisdatentypen müssen als zusammengesetzte Datentypen vorliegen, die genauso strukturiert sind wie die entsprechenden OPC UA-Basisdatentypen.

- OpcUa_NodeId und OpcUa_QualifiedName liegen als Systemdatentyp vor; deshalb können Sie diese Datentypen für einzelne Variablen aber auch als Elemente einer Struktur nutzen.
- Für die übrigen Basisdatentypen müssen Sie einen PLC-Datentyp entsprechend der OPC UA-Spezifikation anlegen, und anschließend als Element in einer Struktur verwenden, damit die Datentypen der Elemente aufgelöst werden können. Wie der PLC-Datentyp jeweils aussehen muss, ist im Folgenden für jeden einzelnen Basisdatentyp beschrieben. Ein Beispiel für eine Datenstruktur, in der z. B. der UDT "LocalizedText" verwendet wird, ist "EUInformation". EUInformation enthält Informationen zu EngineeringUnits. Ein Beispiel zur Umsetzung der Datenstruktur EUInformation finden Sie am Schluss der PLC-Datentypen-Beschreibungen.

Systemdatentyp "OPC_UA_NodeId"

Für den OPC UA Basisdatentyp "OpcUa_NodeId" entnehmen Sie der folgenden Tabelle die Bedeutung der Parameter. OPC_UA_NodeId nutzen Sie zur Identifizierung eines Knotens im OPC UA-Server.

Parameter	S7-Datentyp	Bedeutung
NamespaceIndex	UINT	Namensraumindex des Knotens im OPC UA-Server. Ein Knoten kann zum Beispiel eine Variable sein.
Identifier	WSTRING[254]	Die Bezeichnung für den Knoten (Objekt oder Variable) ist abhängig vom Identifier-Typ: • Numerischer Identifier: Der Knoten wird mit einer Zahl bezeichnet, zum Beispiel "12345678". • String-Identifier: Der Knoten wird mit einem Namen bezeichnet, zum Beispiel "MeineVariable". Groß- und Kleinschreibung wird unterschieden.
IdentifierType	UDINT	Typ des Identifiers • 0: Numerischer Identifier • 1: String-Identifier • 2: GUID • 3: Opaque

Systemdatentyp "OPC_UA_QualifiedName"

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_QualifiedName":

Name	S7-Datentyp	Bedeutung
NamespaceIndex	UINT	Der Namespaceindex des Namens.
Name	WSTRING[64]	Name des Knotens oder der Variablen.

UDT "Guid"

Legen Sie für den Basisdatentyp "Guid" folgenden PLC-Datentyp an. Die exemplarisch eingesetzten Defaultwerte können Sie auch anders setzen.

Guid

	Name	Data type	Default value
1	Data1	UDInt	16#11223344
2	Data2	UInt	16#5566
3	Data3	UInt	16#7788
4	Data4	ULInt	16#99AABBCCDDEEFF11

UDT "LocalizedText"

Legen Sie für den Basisdatentyp "LocalizedText" folgenden PLC-Datentyp an:

LocalizedText

	Name	Data type	Default value
1	EncodingByte	Byte	16#3
2	Locale	WString[5]	WSTRING#'de-DE'
3	Text	WString[255]	WSTRING#'Text'

Das EncodingByte gibt an, welche Felder (Locale bzw. Text) vorhanden sind:

EncodingByte	Bedeutung
0	Die Felder Locale und Text sind leer
1	Das Feld Locale hat Inhalt, das Feld Text ist leer
2	Das Feld Locale ist leer, das Feld Text hat Inhalt
3	Die Felder Locale und Text haben Inhalt

UDT "ByteString"

Legen Sie für den Basisdatentyp "ByteString" folgenden PLC-Datentyp an; hier ist z. B. ein ByteString-Array mit 12 Elementen gewählt:

ByteString			
	Name	Data type	Default value
1	ByteString	Array[0..11] of Byte	
2	ByteString[0]	Byte	16#0
3	ByteString[1]	Byte	16#0
4	ByteString[2]	Byte	16#0
5	ByteString[3]	Byte	16#0
6	ByteString[4]	Byte	16#0
7	ByteString[5]	Byte	16#0
8	ByteString[6]	Byte	16#0
9	ByteString[7]	Byte	16#0
10	ByteString[8]	Byte	16#0
11	ByteString[9]	Byte	16#0
12	ByteString[10]	Byte	16#0
13	ByteString[11]	Byte	16#0

UDT "XmlElement"

Ein XmlElement ist ein serialisiertes XML-Fragment (UTF-8-String).

Legen Sie für den Basisdatentyp "XmlElement" folgenden PLC-Datentyp an:

XmlElement			
	Name	Data type	Default value
1	XmlElement	WString	WSTRING#''

Beispiel: Struktur von EUInformation mit UDT "LocalizedText"

Velocity	"EUInformation"		
NamespaceUri	WString[255]	WSTRING#'http://yourorganization.org'	Identifies the organization (company, standards organization) that defines the EUInformation
UnitId	DInt	1	Identifier for programmatic evaluation. –1 is used if a unitId is not available.
DisplayName	"LocalizedText"		
Encoding...	Byte	16#3	
Locale	WString[5]	WSTRING#'en-EN'	
Text	WString[255]	WSTRING#'m/s'	
Description	"LocalizedText"		
Encoding...	Byte	16#3	
Locale	WString[5]	WSTRING#'en-EN'	
Text	WString[255]	WSTRING#'meter per second'	

MinimumSamplingInterval-Attribut von Variablen

Neben "Value", "DataType" und "AccessLevel" können Sie in der XML-Datei, die den Server-Adressraum repräsentiert, auch das Attribut "MinimumSamplingInterval" für eine Variable setzen.

Das Attribut gibt an, wie schnell der Server den Variablenwert abtasten kann.

Der OPC UA-Server der S7-1500 CPU geht folgendermaßen mit den Werten für MinimumSamplingInterval um:

- Negative Werte und Werte größer als 4294967 werden auf -1 gesetzt; das bedeutet: Default Einstellung des Servers soll für das Sampling Intervall zur Abtastung genutzt werden. Dieses ist durch das Publishing Intervall der Subscription definiert, siehe OPC UA Spezifikation Part 4.
- Kommazahlen werden auf drei Nachkommastellen gerundet.

Siehe auch

Unified_Automation_Download (https://www.unified-automation.com/downloads/opc-ua-development.html)

Fehlende Namensräume

Einleitung

In einer Nodeset-Datei (OPC UA-XML-Datei) befinden sich möglicherweise Verweise (Referenzen) auf andere OPC UA-XML-Dateien, die Typdefinitionen in weiteren Namensräumen zur Verfügung stellen. Sie müssen sämtliche benötigten Definitionen für das Informationsmodell importieren, damit die Server-Schnittstelle ordnungsgemäß funktioniert.

Beispiel für Liste von Namensräumen (NamespaceUris) in einer OPC UA-XML-Datei

```
<NamespaceUris>
    <Uri>http://www.siemens.com/SimaticIdent/RF600R/</Uri>
    <Uri>http://opcfoundation.org/UA/DI/</Uri>
    <Uri>http://opcfoundation.org/UA/AutoID/</Uri>
</NamespaceUris>
```

Weitere Namensräume importieren

Wenn nach dem Import einer OPC UA-XML-Datei noch Namensräume fehlen, dann erzeugt STEP 7 eine Fehlermeldung.

In diesem Fall importieren Sie die OPC UA-XML-Dateien mit den fehlenden Namensräumen in eine neue Server-Schnittstelle. Sie müssen eine neue Server-Schnittstelle anlegen, da sonst die weitere importierte XML-Datei die aktuelle Companion Spezifikation (das aktuelle Informationsmodell) überschreiben würde.

Die fehlenden Namensräume werden in den Eigenschaften der Server-Schnittstelle rot gefärbt angezeigt.

Nach dem Import der XML-Dateien müssen Sie auf die Schaltfläche "Schnittstelle aktualisieren" klicken, damit die Ansicht aktualisiert wird.

Namensräume

Namensräume

Namensraum URI	Verfügbare Version	Benötigte Version
http://www.siemens.com/SimaticIdent/RF600R/	-	-
http://opcfoundation.org/UA/DI/	-	-
http://opcfoundation.org/UA/AutoID/	-	-

Schreib- und Leserechte für CPU-Variablen koordinieren

Definition von Schreib- und Leserechten im Informationsmodell (OPC UA-XML)

Im OPC UA-Informationsmodell regelt das Attribut "AccessLevel" den Zugriff auf Variablen.

AccessLevel ist bitweise definiert:

Bit 0 = CurrentRead und Bit 1 = CurrentWrite. Die Bedeutung der Bitkombinationen ergibt sich wie folgt:

- AccessLevel = 0: kein Zugriff
- AccessLevel = 1: read only
- AccessLevel = 2: write only
- AccessLevel = 3: read+write

Beispiel für die Vergabe von Schreib- und Leserechten (read+write)

```
<UAVariable NodeId="ns=3;s=&quot;Data_block_2&quot;.&quot;Static_1&quot;"
BrowseName="3:Static_1"
ParentNodeId="ns=3;s=&quot;Data_block_2&quot;"
DataType="INT"
AccessLevel="3">
    <DisplayName>Static_1</DisplayName>
```

Definition von Schreib- und Leserechten in STEP 7

Beim Definieren von Variablen legen Sie die Zugriffsrechte fest mit den Eigenschaften "Erreichbar aus HMI/OPC UA“ und "Schreibbar aus HMI/OPC UA".

Beispiel für die Vergabe von Schreib- und Leserechten

Name	Data type	Accessible from HMI/OPC UA	Writable from HMI/OPC UA
Static_1	Int	☑	☑
<Add new>		☑	☑

Zusammenspiel der Schreib- und Leserechte

Wenn Sie eine OPC UA-Server-Schnittstelle importiert haben und in dieser OPC UA-XML-Datei sind AccessLevel-Attribute gesetzt, dann resultieren die Schreib- und Leserechte aus der Regel: Wirksam sind die geringsten Zugriffsrechte aus beiden Einstellungen.

Beispiel

- AccessLevel = 1 (read only) in der OPC UA-Server-Schnittstelle
- Sowohl "Erreichbar aus HMI/OPC UA“ als auch "Schreibbar aus HMI/OPC UA" ist in der PLC-Variablentabelle aktiviert

Ergebnis: Die Variable wird nur gelesen.

Regeln

Wenn Schreibrechte erforderlich sind:

- AccessLevel = 2 oder 3
- "Schreibbar aus HMI/OPC UA" aktiviert

Wenn Leserechte erforderlich sind:

- AccessLevel = 1 (AccessLevel 3 ist auch möglich, aber irreführend. Die Einstellung suggeriert, ein OPC UA-Client hat Schreib- und Leserechte)
- "Erreichbar aus HMI/OPC UA“ aktiviert, "Schreibbar aus HMI/OPC UA" deaktiviert

Wenn weder Lese- noch Schreibrechte eingeräumt werden sollen (kein Zugriff):

- AccessLevel = 0
- "Erreichbar aus HMI/OPC UA“ deaktiviert
  Um jeglichen Zugriff zu sperren, muss nur eine der beiden Bedingungen erfüllt sein. Überdenken Sie in diesem Fall, ob die Variable in der OPC UA-Server-Schnittstelle überhaupt benötigt wird.

Zugriffstabelle

"Erreichbar aus HMI/OPC UA" muss gesetzt sein, damit überhaupt ein Zugriff per OPC UA möglich ist. "Schreibbar aus HMI/OPC UA" muss gesetzt sein, damit ein OPC UA-Client auf eine Variable / ein DB-Element schreiben kann.

Der Tabelle entnehmen Sie das resultierende Zugriffsrecht.

Tabelle 1-90 Zugriffstabelle

OPC UA-XML	STEP 7 (TIA Portal) z. B. Variablentabelle		
AccessLevel	Erreichbar aus HMI/ OPC UA	Schreibbar aus HMI/ OPC UA	Resultierendes Zugriffs-recht
0	x	x	Kein Zugriff
x	0	x	Kein Zugriff
1	aktiviert	x	Read only
2	aktiviert	deaktiviert	Kein Zugriff
3	aktiviert	deaktiviert	Read only
2	aktiviert	aktiviert	Write only
3	aktiviert	aktiviert	Read+write

(x = don't care)

Siehe auch

Konsistenz von CPU-Variablen (Seite 1278)

Konsistenz von CPU-Variablen

Attribut "AccessLevelEx" erweitert Zugriffseigenschaften

Der OPC UA-Server der S7-1500 CPU unterstützt ab Firmware Version V2.6 neben dem Attribut "AccessLevel" (siehe Schreib- und Leserechte für CPU-Variablen koordinieren (Seite 1276)) auch das Attribut "AccessLevelEx", das neben den bereits erläuterten Bits für den Lese- und Schreibzugriff zusätzlich Auskunft über die Konsistenz einer OPC UA-Variable gibt. Das neue Attribut wurde mit Version V1.04 der OPC UA-Spezifikation eingeführt (Part 3, Address Space Model).

Konsistenz-Eigenschaften auslesen

Im OPC UA-Informationsmodell des OPC UA-Servers definiert das Attribut "AccessLevelEx" den Zugriff auf Variablen.

AccessLevelEx ist bitweise definiert, die hier relevanten Bits sind folgende:

- Bit 0 = CurrentRead
- Bit 1 = CurrentWrite
- Bit 2 bis 7 sind nicht relevant für den OPC UA-Server einer S7-1500 CPU

Die Bedeutung der Bitkombinationen ist im Abschnitt zu den Lese- und Schreibrechten erläutert.

Zusätzlich kommen folgende Bits für die Konsistenzeigenschaften hinzu:

- Bit 8 = NonatomicRead; das Bit ist gesetzt, wenn die Variable nicht konsistent gelesen werden kann. Bei Lese-Konsistenz der Variablen ist Bit 8=0.
- Bit 9 = NonatomicWrite; ist gesetzt, wenn die Variable nicht konsistent geschrieben werden kann. Bei Schreib-Konsistenz der Variablen, bzw. wenn kein schreibender Zugriff gewährt ist, dann ist Bit 9=0.

Beispiele

Eine OPC UA-Variable (Struktur) ist lesbar und schreibbar; aber inkonsistent für lesenden und für schreibenden Zugriff.

Daraus folgt: Die Bits 0, 1, 8 und 9 sind gesetzt: AccessLevelEx = "771" (1+2+256+512).

Eine andere Struktur ist nur lesbar.

Daraus folgt: Die Bits 0 und 8 sind gesetzt, Bit 1 und Bit 9 sind nicht gesetzt: AccessLevelEx = "257" (1+0+256+0).

Behandlung des Attributs im Server

Das Attribut "AccessLevelEx" steht nur im OPC UA-Server zur Verfügung. Das Attribut ist nicht in einer Nodeset-Datei (XML-Export-Datei) vorhanden.

Allerdings nimmt das Attribut "AccessLevel", das exportiert wird, die Information von "AccessLevelEx" mit auf, siehe nächsten Abschnitt.

Export

Beim XML-Export der Standard-Simatic-Server-Schnittstelle setzt der Server das Attribut "AccessLevel", das im Unterschied zur V1.03-Spezifikation in V1.04 auf 32 Bit erweitert wurde, auf den Wert von Attribut "AccessLevelEx".

Import

Beim Import einer Nodeset-Datei (z. B. aus einem Export einer Server-Schnittstelle) setzt die S7-1500 CPU das Attribut "AccessLevelEx" nach ihrem eigenen Wissen um die Konsistenz des importierten Datentyps, siehe nächsten Abschnitt. Der importierte Wert wird ignoriert.

Konsistenz von Datentypen an der Server-Schnittstelle

Die Konsistenz von Variablen (im Sprachgebrauch von OPC UA: "atomicity") innerhalb eines Programmzyklus einer S7-1500 CPU ist an den Knoten der Server-Schnittstelle sichergestellt für folgende Datentypen:

- BOOL, BYTE, WORD, DWORD, LWORD
- SINT, INT, LINT, DINT, USINT, UINT, ULINT, UDINT
- REAL, LREAL
- DATE, LDT, TIME, LTIME, TIME_OF_DAY, LTIME_OF_DAY, S5TIME
- CHAR, WCHAR
- Ebenfalls konsistent sind Systemdatentypen bzw. Hardware-Datentypen, die auf den oben genannten Datentypen basieren.
  Beispiel: HW_ANY, von UINT (UInt16) abgeleitet.

Tipp: Wenn Sie im Adressraum der S7-1500 CPU browsen (z. B. mit dem OPC UA Client UaExpert), dann finden Sie die konsistenten Datentypen unter Types > BaseDataType > Enumeration/Number/String.

Variablen von folgenden Datentypen sind **nicht** konsistent (im Sprachgebrauch von OPC UA: „nonatomic“):

- SIMATIC-Strukturen sind generell nicht konsistent. D. h. alle Variablen, die z. B. unbenannte Strukturen sind oder einen UDT-Datentyp haben, sind nicht konsistent.
- Systemdatentypen wie z. B. DTL, IEC_Counter, IEC_TIMER etc.- das sind Datentypen, die von Strukturen abgeleitet sind.

Tipp: Wenn Sie im Adressraum der S7-1500 CPU browsen (z. B. mit dem OPC UA Client UaExpert), dann finden Sie Datentypen, die auf Strukturen basieren, unter Types > BaseDataType > Structure.

Hinweise zu Mengengerüsten bei Nutzung von Server-Schnittstellen

Wenn Sie OPC UA-Server-Schnittstellen verwenden, dann müssen Sie abhängig von Leistungsklasse der S7-1500 CPU Obergrenzen für folgende Objekte berücksichtigen:

- Anzahl der Server-Interfaces
- Anzahl der OPC UA-Knoten (Nodes)
- Datenmenge der Ladeobjekte
- Falls Sie Methoden implementiert haben: Anzahl Server-Methoden bzw. Server-Methoden-Instanzen

Mengengerüste für OPC UA-Server-Schnittstellen und Methoden

In folgender Tabelle sind die Mengengerüste der S7-1500 CPUs dokumentiert, die auch beim Übersetzen und Laden einer Konfiguration berücksichtigt werden.

Eine Verletzung der Mengengerüste wird mit einer Fehlermeldung quittiert.

Tabelle 1-91 Mengengerüste für OPC UA-Server-Schnittstellen

Technisches Datum	CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)	CPU 1505 (S/SP/SP F/ SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)	CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)
Nutzung von importierten Companion Spezifikationen (Informationsmodelle)			
• Maximale Anzahl OPC UA-Server-Schnittstellen	10	10	10
• Maximale Anzahl von OPC UA Knoten (Nodes) in benutzerdefinierten Server-Schnittstellen	1000	5000	30000
• Maximale Größe ladbarer OPC UA-Server-Schnittstellen	1024 KB	5120 KB	15360 KB
Bereitstellung von Methoden			
Maximale Anzahl nutzbarer Server-Methoden bzw. max. Anzahl Server-Methoden-Instanzen (Anweisungen OPC_UA_ServerMethodPre, OPC_UA_ServerMethodPost)	20	50	100

S7-1500 CPU als OPC UA-Client nutzen

Übersicht und Voraussetzungen

Mit STEP 7 (TIA Portal) können Sie ab der Version 15.1 einen OPC UA-Client projektieren und programmieren, der PLC-Variablen in einem OPC UA-Server lesen kann. Ferner ist es möglich, zu einem OPC UA-Server neue Werte für PLC-Variablen zu übertragen. Außerdem können Sie in Ihrem Anwenderprogramm Methoden aufrufen, die ein OPC UA-Server bereitstellt. Sie verwenden dazu in Ihrem Anwenderprogramm die neuen Anweisungen für OPC UA-Clients.

Die Anweisungen des OPC UA-Clients sind angelehnt an den Standard "PLCopen OPC-UA Client for IEC61131-3".

Spezifikation PLCopen

Mit diesen standardisierten Anweisungen können Sie in Ihrem Anwenderprogramm OPC UA-Client-Funktionen nutzen, die in einer S7-1500 CPU ausgeführt werden.

Außerdem ist es mit geringen Anpassungen möglich, dieses Anwenderprogramm auch in Steuerungen anderer Hersteller auszuführen, sofern diese Hersteller ebenfalls die OPC UA Spezifikation "PLCopen OPC-UA Client for IEC61131-3" umsetzten.

Komfortable Editoren in STEP 7

Für die Parametrierung der Anweisungen für OPC UA-Clients steht Ihnen im TIA Portal ein komfortabler Editor zur Verfügung - die Verbindungsparametrierung (Seite 1227).

Außerdem verfügt STEP 7 ab Version 15.1 auch über einen Editor für Client-Schnittstellen (Seite 1287).

Dieses Kapitel beschreibt, wie Sie mit diesen Editoren arbeiten.

Zunächst soll gezeigt werden, wie Sie mit dem Schnittstellen-Editor eine neue Schnittstelle anlegen und konfigurieren, da Sie eine solche Schnittstelle für die spätere Verbindungsparametrierung benötigen.

Die Beschreibung verwendet zur besseren Verständlichkeit ein Beispiel, siehe Beschreibung des Beispiels (Seite 1285).

Voraussetzung

Um den Client der S7-1500 CPU zu nutzen, müssen Sie den Client aktivieren:

1. Wählen Sie in den Eigenschaften der CPU den Bereich "OPC UA > Client".
2. Aktivieren Sie die Option "OPC UA-Client aktivieren".

Wenn Sie den Client nicht aktivieren, kommt kein Verbindungsaufbau zustande. Sie erhalten eine entsprechende Fehlermeldung an den Anweisungen, z.B. "OPC_UA_Connect".

Informationen zum Applikationsnamen, der auch für den Server und den Client gilt, finden Sie hier.

Übersicht

Um den Editor und die Verbindungsparametrierung zu verwenden, gehen Sie folgendermaßen vor:

1. Legen Sie zunächst eine Client-Schnittstelle an. Fügen Sie der Schnittstelle PLC-Variablen und -Methoden hinzu, auf die Sie zugreifen möchten ("Erster Schritt (Seite 1287)").
2. Parametrieren Sie anschließend die Verbindung zum OPC UA-Server (Zweiter Schritt (Seite 1302)).
3. Schließlich verwenden Sie die parametrierte Verbindung bei den OPC UA-Client-Anweisungen (Dritter Schritt (Seite 1310)).

Wissenswertes zu den Client-Anweisungen

Die S7-1500 CPU als OPC UA-Client ermöglicht Ihnen, mit standardisierten OPC UA Client-Anweisungen die Kommunikation für folgende Aufgaben zu steuern:

- Lesen/Schreiben von Variablen des OPC UA-Servers
- Methoden im OPC UA-Server aufrufen

Optional verwendbare Anweisungen erlauben Ihnen, den Status der Verbindung zwischen OPC UA-Client und OPC UA-Server oder entlang eines bekannten Hierarchie Knotens zu ermitteln.

Standardisierter Ablauf der OPC UA Kommunikation

Der Ablauf der Kommunikation und damit die Reihenfolge der Anweisungen folgt einem Muster, das im Folgenden veranschaulicht ist.

Ablaufreihenfolge für Lese- bzw. Schreibvorgang

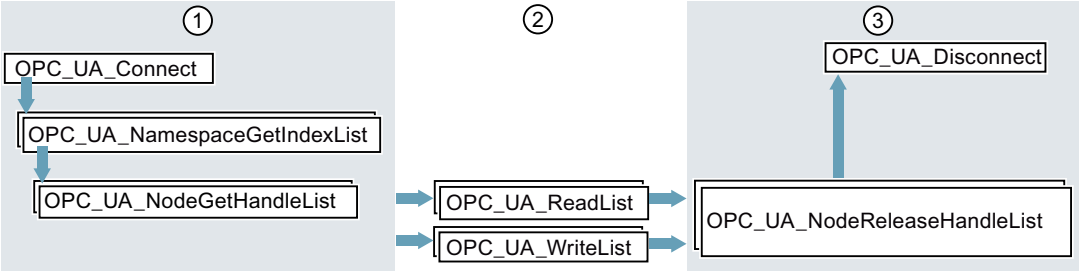

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum "Clean-up" nach durchgeführten Lese- oder Schreibvorgängen

Ablaufreihenfolge für Methodenaufruf im OPC UA-Server

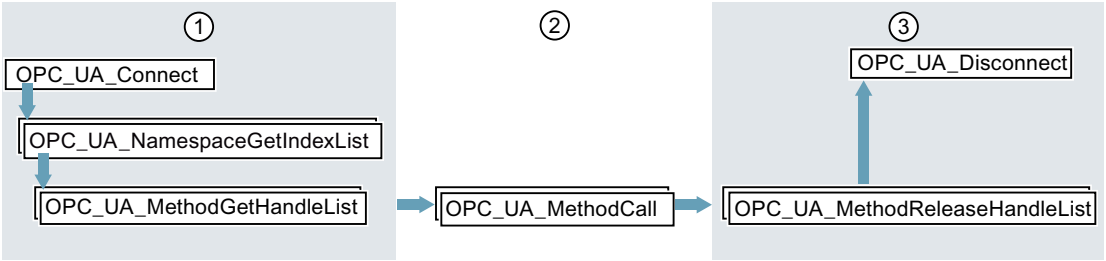

① Anweisungen zur Vorbereitung der Methodenaufrufe

② Methodenaufrufe

③ Anweisungen zum "Clean-up" nach durchgeführten Methodenaufrufen

Optionale Anweisungen zum Auslesen des Status einer Verbindung zwischen OPC UA-Client und OPC UA-Server und zum Auslesen kompletter Pfade aus dem OPC UA-Server.

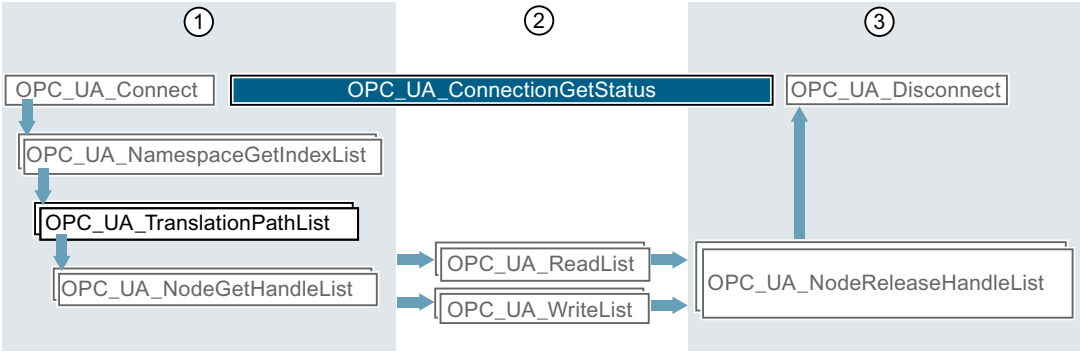

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge mit eingeschobener Anweisung zum Erfragen z. B. der NodeIds von Knoten des OPC UA-Servers.

② Parallel zu anderen Anweisungen können Sie zwischen Verbindungsaufbau und Verbindungsabbau den Verbindungsstatus ermitteln

③ Anweisungen zum "Clean-up"

Komfortable Editoren in STEP 7

Die OPC UA-Client Anweisungen sind detailliert im Referenzteil (STEP 7 Informationssystem) beschrieben. Für die Parametrierung der Anweisungen steht Ihnen im TIA Portal ein komfortabler Editor zur Verfügung - die Verbindungsparametrierung (Seite 1302).

Wir empfehlen für den ersten Programmentwurf, mit der Verbindungsparametrierung zu beginnen und bei Bedarf weitere Anweisungen zu nutzen und das Programm manuell zu optimieren.

Informationen zu den Client-Anweisungen

Die Client-Anweisungen sind detailliert beschrieben in der Hilfe zu den Anweisungen > Kommunikation > OPC UA > OPC UA-Client.

Anzahl gleichzeitig nutzbarer Client-Anweisungen

Gleichzeitig nutzbare OPC UA-Client-Anweisungen

Folgende Grenzen gelten für die gleichzeitige Nutzung von OPC UA-Client-Anweisungen:

Tabelle 1-92 Mengengerüste für OPC UA-Client-Anweisungen

OPC UA-Anweisung	Max. Anzahl für CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)	Max. Anzahl für CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)	Max. Anzahl für CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)
OPC_UA_Connect	4	10	40
OPC_UA_NamespaceGetIndexList	4	10	40
OPC_UA_NodeGetHandleList	4	10	40
OPC_UA_MethodGetHandleList	4	10	40
OPC_UA_TranslatePathList	4	10	40
OPC_UA_ReadList	20 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	50 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	200 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)
OPC_UA_WriteList	20	50	200
OPC_UA_MethodCall	20	50	200
OPC_UA_NodeReleaseHandleList	4	10	40
OPC_UA_MethodReleaseHandleList	4	10	40
OPC_UA_Disconnect	4	10	40
OPC_UA_ConnectionGetStatus	4	10	40

Maximale Anzahl nutzbarer OPC UA-Client-Schnittstellen

Wenn Sie mit Hilfe der Verbindungsparametrierung OPC UA-Client Schnittstellen anlegen, dann ist die Anzahl von Client-Schnittstellen auf max. 40 begrenzt.

Sie legen die OPC UA-Client-Schnittstellen an, indem Sie im Bereich "OPC UA-Kommunikation" in der Projektnavigation auf das Symbol "Neue Client-Schnittstelle hinzufügen" doppelklicken.

Die maximale Anzahl von OPC UA-Client-Schnittstellen ist unabhängig davon, ob Sie die CPU zusätzlich als OPC UA-Server nutzen.

Beispiel-Konfiguration für OPC UA

In den folgenden Kapiteln ist beschrieben, wie Sie den Editor für Client-Schnittstellen sowie die Verbindungsparametrierung nutzen.

Die Beschreibung geht von einem konkreten Beispiel aus: In der Anlage arbeiten zwei S7-1500 CPUs: Eine CPU dient als OPC UA-Client, die andere als OPC UA-Server.

Selbstverständlich können Sie auch Steuerungen, Sensoren oder IT-Systeme anderer Hersteller als OPC UA-Clients oder -Server verwenden. Gerade der Datenaustausch zwischen unterschiedlichen Systemen (Interoperabilität) ist ein großer Vorteil von OPC UA.

Verbindungsparametrierung anhand eines Beispiels:

Die Anlage produziert in einer Fertigungslinie Rohlinge.

Die folgenden Steuerungen werden eingesetzt:

1. Als Controller der Fertigungslinie dient eine S7-1511 CPU.
   Der Controller wird im Beispiel "**Productionline**" genannt.
   Der OPC UA-Server der Steuerung ist aktiviert.
   Die CPU besitzt im Beispiel die IP-Adresse 192.168.1.1.
   Diese CPU veröffentlicht über den OPC UA-Server die Werte der folgenden Variablen:
   - **NewProduct**
     Die Variable besitzt den Datentyp "Bool".
     Wenn die Variable den Wert TRUE besitzt, dann hat die Fertigungslinie einen Rohling bearbeitet.
     Der Rohling steht für die Abholung bereit.
   - **ProductNumber**
     Diese Variable enthält die Identnummer des Rohlings.
     Die Variable besitzt den Datentyp "Int".
   - **Temperature**
     Diese Variable enthält Temperaturwerte, die während der Fertigung des Rohlings erfasst wurden.
     Die Variable ist ein Array mit Elementen des Datentyps "Real".

   Außerdem stellt diese CPU die folgende beschreibbare Variable bereit:
   - **ProductionEnabled**
     Die Variable wird vom OPC UA-Client gesetzt.
     Die Variable besitzt den Datentyp "Bool".
     Wird der Wert auf TRUE gesetzt, dann ist die Fertigungslinie freigegeben und darf Rohlinge herstellen.

   Außerdem stellt diese CPU über den OPC UA-Server die folgende Methode bereit:
   - **OpenDoor**.
     Damit können OPC UA-Clients veranlassen, dass eine Zugangstür zur Fertigungslinie geöffnet wird.
2. Eine S7-1516 CPU kontrolliert die Zusammenarbeit mit anderen Fertigungslinien.
   Diese CPU wird im Beispiel "**Supervisor**" genannt.
   Der OPC UA-Client dieser CPU ist aktiviert.
   Über OPC UA kann diese CPU die Variablen NewProduct und ProductNumber lesen, die Variable ProductionEnabled setzen sowie die Methode OpenDoor aufrufen.
   Diese CPU besitzt im Beispiel die IP-Adresse 192.168.1.2.

Das folgende Bild zeigt das Beispiel in der Netzsicht des TIA Portals:

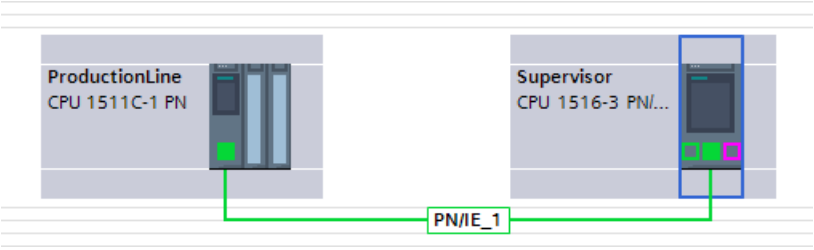

Client-Schnittstellen anlegen

Das TIA Portal verfügt ab der Version 15.1 über einen Editor für Client-Schnittstellen.

In einer Client-Schnittstelle fassen Sie alle PLC-Variablen zusammen, die Sie von einem OPC UA-Server lesen oder schreiben wollen.

Außerdem enthält die Client-Schnittstelle alle Methoden, die der OPC UA-Server bereitstellt und die Sie mit Ihrem Anwenderprogramm, das als OPC UA-Client fungiert, aufrufen wollen.

Wenn Sie eine Client-Schnittstelle anlegen, dann erstellt STEP 7 auch Datenbausteine für die Parametrierung der Verbindung zu dem OPC UA-Server, von dem Sie Daten lesen oder zu dem Sie Daten schreiben wollen.

Maximale Anzahl von Client-Schnittstellen

Sie können maximal 40 Client-Schnittstellen anlegen.

Editor für Client-Schnittstellen

Um eine Client-Schnittstelle zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Projektansicht im TIA Portal
2. Wählen Sie im Bereich "Geräte" die CPU aus, die Sie als OPC UA-Client nutzen wollen.
3. Klicken Sie auf "OPC UA-Kommunikation > Client-Schnittstellen".

4. Doppelklicken Sie auf "Neue Client-Schnittstelle hinzufügen".
   STEP 7 erzeugt eine neue Client-Schnittstelle und zeigt diese im Editor an:

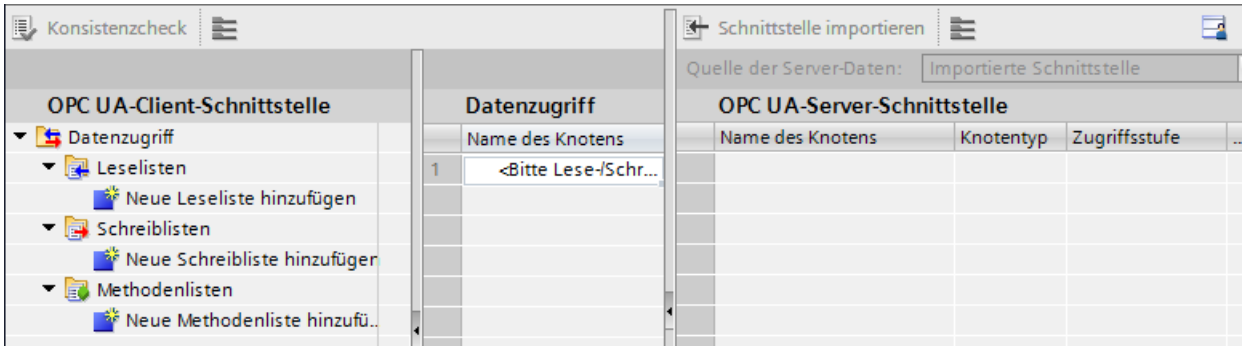

   STEP 7 nennt die neue Schnittstelle "Client-Schnittstelle_1". Wenn es schon eine "Client-Schnittstelle_1" gibt, dann erhält die neue Schnittstelle die Bezeichnung "Client-Schnittstelle_2" usw.
   Außerdem erzeugt STEP 7 die folgenden Datenbausteine:

   – Client-Schnittstelle_1_Configuration
     Der Datenbaustein enthält bereits alle Systemdatentypen, die für die Anweisungen des OPC UA-Clients benötigt werden.
     Wenn Sie die Verbindung zum OPC UA-Server parametrieren, wird dieser Datenbaustein gefüllt.
     Sie parametrieren eine Verbindung in den Eigenschaften der Client-Schnittstelle, siehe: Beispiel-Konfiguration für OPC UA (Seite 1285).

   – Client-Schnittstelle_1_Data
     Ein Datenbaustein für die PLC-Variablen, die Sie von einem OPC UA-Server lesen oder schreiben wollen.
     Diesen Datenbaustein verwenden Sie in Ihrem Anwenderprogramm.
     Aktuell ist dieser Datenbaustein noch leer.

5. Verwenden Sie einen aussagekräftigen Namen für die neue Client-Schnittstelle.
   Im Beispiel wählen Sie "Productionline".
   Dadurch ändern sich auch die Namen der zugehörigen Datenbausteine in:

   – Productionline_Data

   – Productionline_Configuration

6. Um eine OPC UA-Server-Schnittstelle zu importieren, klicken Sie rechts oben im Editor auf die Schaltfläche "Schnittstelle importieren".
   Dadurch können Sie eine XML-Datei importieren, welche die Server-Schnittstelle eines OPC UA-Servers beschreibt.
   Alternative: Sie ermitteln online die Server-Schnittstelle eines verbundenen OPC UA-Servers, siehe: Server-Schnittstelle online ermitteln (Seite 1296).

7. STEP 7 zeigt einen Dialog an, mit dem Sie eine XML-Datei auswählen.
   Diese XML-Datei beschreibt eine Server-Schnittstelle (Adressraum eines OPC UA-Servers).
   Eine Server-Schnittstelle ist die Zusammenfassung aller PLC-Variablen und Server-Methoden, die ein OPC UA-Server veröffentlicht.
   Auf die Server-Schnittstelle können OPC UA-Clients zugreifen:
   - PLC-Variablen lesen
   - PLC-Variablen schreiben
   - Server-Methoden aufrufen
   Für das Erstellen einer Server-Schnittstelle für einen OPC UA-Server, siehe: Server-Schnittstelle anlegen (Seite 1258).

8. Erzeugen Sie eine Leseliste in dieser Client-Schnittstelle.
   Dazu gehen Sie folgendermaßen vor:
   - Klicken Sie im linken Abschnitt des Editors auf "Neue Leseliste hinzufügen".
     STEP 7 fügt eine neue Liste hinzu mit dem Namen "Leseliste_1".
     Für das Beispiel ändern Sie den Namen in "ReadProduct"
   - Fügen Sie nun der neuen Leseliste die PLC-Variablen hinzu, die Sie vom OPC UA-Server lesen wollen.
     Im Beispiel werden in der Leseliste "ReadProduct" die Variablen "NewProduct" und "ProductNumber" hinzugefügt.
     Wählen Sie die Variable "NewProduct" im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Variable "NewProduct" per Drag & Drop zur Leseliste "ReadProduct" im mittleren Feld des Editors. Verfahren Sie genau so mit der Variable "ProductNumber".

   Das folgende Bild zeigt das rechte Feld des Editors:

OPC UA-Server-Schnittstelle

Name des Knotens	Knotentyp	Zugriffsstufe
Productionline	Object	
DataBlocksGlobal	Folder	
Data_for_OPC_UA_Clients	Object	
NewProduct	Boolean	RD
ProductNumber	Int16	RD
Temperature	Array of Fl...	RD
Data_from_OPC_UA_Clients	Object	
ProductionEnabled	Boolean	RD/WR
DataBlocksInstance	Folder	
OpenDoor_DB	Object	
Static	Folder	
Method	Method	

Alternative:
Sie können eine neue Leseliste auch erzeugen, indem Sie im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") einen Knoten des Typs Object oder Folder auswählen und dann per Drag & Drop zu "Neue Leseliste hinzufügen" im linken Feld des Editors ziehen. Die neue Leseliste enthält dann alle PLC-Variablen des gezogenen Knotens.
Im Beispiel wählen Sie das Objekt "Data_for_OPC_UA_Clients" aus, das die Variablen "NewProduct" und "ProductNumber" enthält. STEP 7 erzeugt die neue Schreibliste "Data_for_OPC_UA_Clients". Außerdem enthält das Objekt auch die Variable "Temperature". Löschen Sie die Variable "Temperature" aus der Leseliste. Der OPC UA-Server soll die Variable "Temperature" nicht abfragen.
Ändern Sie den Namen der Leseliste in "ReadProduct".
Das folgende Bild zeigt den Inhalt der Leseliste:

ReadProduct

Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID
NewProduct	BOOL	RD	http://www.siemen...
ProductNumber	INT	RD	http://www.siemen...

9. Falls Sie PLC-Variablen neue Werte zuweisen wollen, erzeugen Sie eine Schreibliste in dieser Client-Schnittstelle.
   Dazu gehen Sie folgendermaßen vor:
   - Klicken Sie im linken Abschnitt des Editors auf "Neue Schreibliste hinzufügen".
     STEP 7 fügt eine neue Liste hinzu mit dem Namen "Schreibliste_1".
     Für das Beispiel ändern Sie den Namen in "WriteStatus".
   - Fügen Sie nun der neuen Schreibliste alle Variablen des OPC UA-Servers hinzu, denen Sie neue Werte zuweisen wollen.
     Im Beispiel fügen Sie der Schreibliste "WriteStatus" die Variable "ProductionEnabled" hinzu.
     Wählen Sie die Variable im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Variable per Drag & Drop zur Schreibliste im mittleren Feld des Editors.

   Alternative:
   Sie können eine neue Schreibliste auch erzeugen, indem Sie im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") einen Knoten des Typs Object oder Folder auswählen und dann per Drag & Drop zu "Neue Schreibliste hinzufügen" im linken Feld des Editors ziehen. Die neue Schreibliste enthält dann alle Variablen des betreffenden Knotens.
   Im Beispiel wählen Sie das Objekt "Data_from_OPC_UA_Clients" aus, das die Variable "ProductionEnabled" enthält. STEP 7 erzeugt die neue Schreibliste "Data_from_OPC_UA_Clients". Ändern Sie den Namen in "WriteStatus".
   Das folgende Bild zeigt den Inhalt der Schreibliste:

WriteStatus

Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID
ProductionEnabled	BOOL	RD/WR	http://www.siemen...

10. Falls Sie eine Methode dieses OPC UA-Servers aufrufen wollen, erzeugen Sie eine neue Methodenliste.
    Dazu gehen Sie folgendermaßen vor:
    - Klicken Sie im linken Abschnitt des Editors auf "Neue Methodenliste hinzufügen".
      STEP 7 fügt eine neue Liste hinzu mit dem Namen "Methodenliste_1".
      Für das Beispiel ändern Sie den Namen in "CallOpenDoor".
    - Fügen Sie nun der neuen Methodenliste eine Methode des OPC UA-Servers hinzu.
      In diesem Beispiel fügen Sie der Methodenliste "CallOpenDoor" die Methode "OpenDoor" hinzu.
      Wählen Sie die Methode im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Methode per Drag & Drop zur Methodenliste im mittleren Feld des Editors.

    Alternative:
    Sie können eine neue Methodenliste auch erzeugen, indem Sie im rechten Feld des Editors (OPC UA-Server-Schnittstelle) eine Methode (Knoten des Typs Object) auswählen und dann per Drag & Drop zu "Neue Methodenliste hinzufügen" im linken Feld des Editors ziehen. Die neue Methodenliste enthält dann die Methode des betreffenden Knotens.
    Das folgende Bild zeigt den Inhalt der Methodenliste:

    CallOpenDoor

Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID
Method			http://www.siemen...

    Falls Sie eine weitere Methode des OPC UA-Servers aufrufen wollen, dann Sie müssen Sie eine neue Methodenliste anlegen. Jede Methodenliste enthält nur eine Methode.

11. Übersetzen Sie das Projekt.
    Dazu wählen Sie das Projekt aus und klicken dann in der Symbolleiste auf das folgende Symbol:

STEP 7 übersetzt das Projekt und aktualisiert die Datenbausteine, die zu der Client-Schnittstelle "ProductionLine" gehören.

Hinweis

STEP 7 überschreibt beim Übersetzen sämtliche Daten in den Datenbausteinen, die zur Client-Schnittstelle gehören. Aus diesem Grund sollten Sie diese Datenbausteine weder manuell ergänzen noch korrigieren.

Hinweis

Umbenennen von Knotennamen (DisplayNames)

Sie können in Leselisten, Schreiblisten und Methodenlisten den Namen eines Knotens über das Kontextmenü umbenennen. Im OPC UA Sprachgebrauch ist das der "DisplayName".

Wenn Sie den Namen des Knotens einer Methodenliste umbenennen und der Knoten bereits in einem programmierten Baustein für den Methodenaufruf "OPC_UA_MethodCall" verwendet wird, führt die Übersetzung des Projekts zu Konsistenzfehlern: Die UDTs der Methode werden bei der Übersetzung mit dem geänderten Namen neu erzeugt. Die im Programm verwendeten Referenzen auf die Methode stimmen dann nicht mehr.

Um die Konsistenzfehler zu beheben, können Sie entweder die Änderung des Namens der Methode in der Client-Schnittstelle rückgängig machen oder Sie navigieren zu dem Methodenaufruf und weisen dort unter "Eigenschaften > Bausteinparameter" (Register "Konfiguration") die betreffenden Parameter erneut zu.

Datenbausteine der Client-Schnittstelle

Folgende Datenbausteine gehören zur Client-Schnittstelle "Productionline":

- **Productionline_Configuration**
  Ein Datenbaustein für die Konfiguration.
  In dem Beispiel heißt dieser Datenbaustein "Productionline_Configuration".
  Der Datenbaustein enthält bereits alle Systemdatentypen, die für die Anweisungen des OPC UA-Clients benötigt werden.
  Außerdem enthält der Datenbaustein allgemeine Vorgabenwerte für die Parametrierung der Verbindung zu einem OPC UA-Server.
  Wenn Sie mit der Verbindungsparametrierung arbeiten, wird dieser Datenbaustein gefüllt.
- **Produktionline_Data**
  Ein Datenbaustein für die PLC-Variablen, die Sie in dem Client-Schnittstellen-Editor eingegeben haben.
  In dem Beispiel heißt dieser Datenbaustein "Productionline_Data".
  Das folgende Bild zeigt den Datenbaustein.

Static	
▼ ReadProduct	Struct
▼ Variable	"Productionline.ReadProduct"
NewProduct	Bool
ProductNumber	Int
▼ NodeStatusList	Array[0..1] of DWord
NodeStatusList[0]	DWord
NodeStatusList[1]	DWord
▼ TimeStamps	Array[0..1] of LDT
TimeStamps[0]	LDT
TimeStamps[1]	LDT
▼ WriteStatus	Struct
▼ Variable	"Productionline.WriteStatus"
ProductionEnabled	Bool
▼ NodeStatusList	Array[0..0] of DWord
NodeStatusList[0]	DWord
▼ CallOpenDoor	Struct
▼ MethodStatusList	Array[0..0] of DWord
MethodStatusList[0]	DWord
▶ Method	Struct

Den Datenbaustein "ProductionLine_Data" verwenden Sie in Ihrem Anwenderprogramm und greifen auf die gelesenen Werte der PLC-Variablen "NewProduct" und "ProductNumber" zu. Im folgenden Abschnitt ist dieser Sachverhalt beispielhaft erläutert.

PLC-Variablen der Client-Schnittstelle lesen und schreiben

Beispiel: Wert "ProductNumber" lesen

In einem SCL-Programm schreiben Sie zum Beispiel:

```
#MyLocalVariable :=
"ProductionLine_Data".ReadProduct.Variable.ProductNumber;
```

Damit weisen Sie z. B. der lokalen Variablen "#MyLocalVariable" die Nummer des Rohlings zu, der gerade in der Fertigungslinie hergestellt wurde.

Voraussetzungen:

- Eine Verbindung besteht zum OPC UA-Server der CPU, welche die Fertigungslinie steuert
- Der OPC UA-Client hat die aktuellen Werte gelesen

Deshalb überprüfen Sie, ob ein gelesener Wert gültig (valide) ist:

- Prüfen Sie, ob der Wert in "ProductionLine_Data".ReadProduct.NodeStatusList[1] gleich 0 ist
- Optional: Prüfen Sie, wann dieser Wert vom OPC UA-Server gelesen wurde. Dieser Wert steht in "ProductionLine_Data".ReadProduct.TimeStamps[1]. Wenn keine Zeitstempel angefordert werden, verringert sich die Kommunikationslast.

Beispiel: Wert "ProductEnabled" schreiben

Mit dem Datenbaustein übertragen Sie zum OPC UA-Server neue Werte für PLC-Variablen, in dem Beispiel für die Variable "ProductionEnabled".

Mit der folgenden Zuweisung geben Sie in der Beispielanlage die Fertigungslinie frei:
```
"ProductionLine_Data".WriteStatus.Variable.ProductionEnabled :=
TRUE;
```

Die Freigabe ist aber nur dann erfolgreich, wenn folgende Voraussetzungen erfüllt sind:

- Eine Verbindung besteht zum OPC UA-Server der CPU, welche die Fertigungslinie steuert
- Aktuelle Werte werden über den OPC UA-Client geschrieben

Konsistenzcheck

Zu guter Letzt prüfen Sie die Konsistenz der Lese-/Schreib- bzw. Methodenliste.

1. Markieren Sie die Liste, die Sie prüfen wollen
2. Klicken Sie auf die Schaltfläche "Konsistenzcheck" über dem Bereich "OPC UA-Client-Schnittstelle"

Eine fehlerfreie Zuordnung der Variablen bzw. Methoden zu den korrespondierenden Elementen der Server-Schnittstelle wird durch ein grünes Häkchen angezeigt.

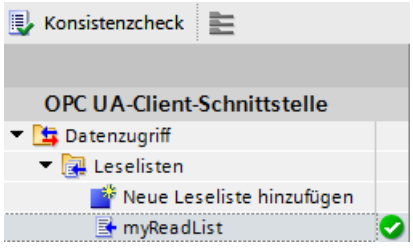

Sie können davon ausgehen, dass der Datenaustausch zwischen Client und Server sowie Methodenaufrufe zur Laufzeit ohne Probleme funktionieren.

Im Fehlerfall wird eine Liste im Inspektorfenster angezeigt. Aus dieser Liste heraus können Sie zum jeweiligen Fehler springen.

STEP 7 prüft beim Konsistenzcheck:

- Sind alle Elemente, die Sie in der jeweiligen Liste verwenden, auch im Server vorhanden?
- Stimmen die verwendeten Datentypen überein?
- Bei Methoden: Stimmen Anzahl, Namen, Reihenfolge und Datentypen von Methoden-Argumenten überein?

Server-Schnittstelle online ermitteln

Mit STEP 7 (TIA Portal) können Sie die Schnittstelle eines OPC UA-Servers online ermitteln. Damit bringen Sie in Erfahrung, welche Variablen eines verbundenen OPC UA-Server Sie mit OPC UA-Clients lesen oder setzen (schreiben) können. Außerdem erfahren Sie, welche Server-Methoden der OPC UA-Server für OPC UA-Clients bereitstellt.

Wenn Sie offline arbeiten, können Sie die Schnittstelle des OPC UA-Servers über eine OPC UA XML-Datei erstellen. In der OPC UA XML-Datei ist der Adressraum des Servers beschrieben, siehe: OPC UA-XML-Datei exportieren (Seite 1226).

Online Server-Schnittstellen ermitteln

Um eine Server-Schnittstelle online zu ermitteln, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation von STEP 7 die CPU aus, die als OPC UA-Client projektiert ist (im Beispiel Supervisor).
2. Wählen Sie die Client-Schnittstelle aus (im Beispiel OPC UA-Kommunikation > Client-Schnittstellen > Productionline).
   Falls noch keine Client-Schnittstelle angelegt wurde, dann doppelklicken Sie auf "Neue Client-Schnittstelle hinzufügen".
3. Doppelklicken Sie auf die ausgewählte Client-Schnittstelle.
   Der Editor für Client-Schnittstellen wird angezeigt:

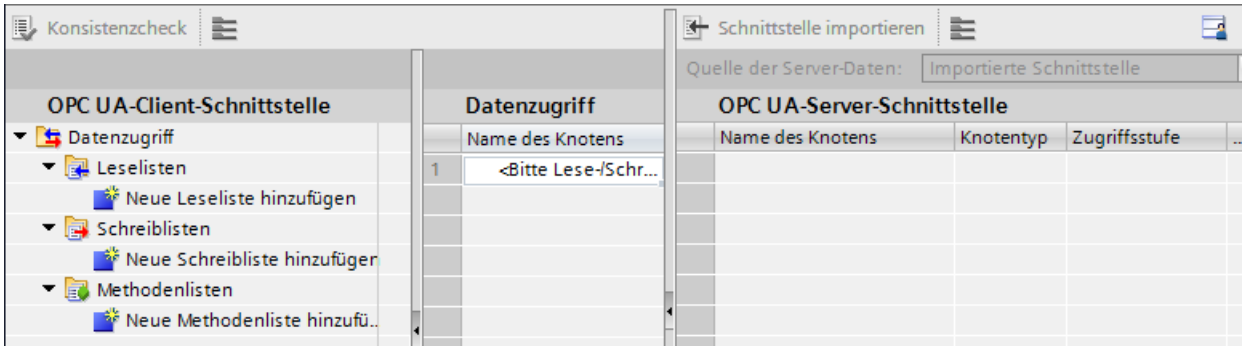

4. Klicken Sie im linken Feld des Editors auf "Neue Leseliste hinzufügen", "Neue Schreibliste hinzufügen" oder auf "Neue Methodenliste hinzufügen".
5. Wählen Sie im rechten Feld des Editors bei "Quelle der Server-Daten" als Datenquelle "Online[]":

6. Klicken Sie auf die Schaltfläche "Online-Zugänge".
   STEP 7 zeigt den Dialog "Mit OPC UA-Server verbinden":

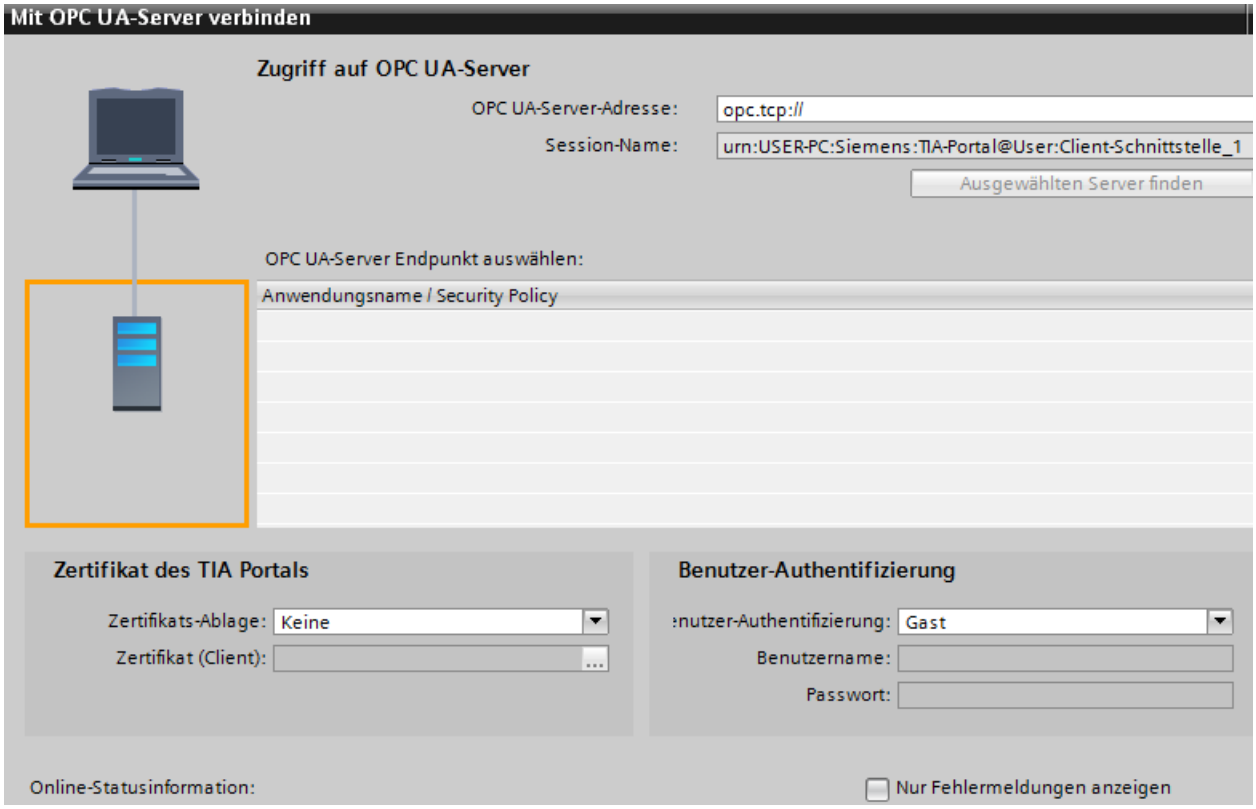

   Tipp: Wenn Sie erstmals eine Online-Verbindung zu einem OPC UA-Server aufbauen, nutzen Sie die Schaltfläche "Online-Zugänge". Beim erneuten Verbinden nach einem Verbindungsabbruch wählen Sie die Schaltfläche "Mit Online-Server verbinden" neben dem Auswahlfeld "Online".
   Tragen Sie oben rechts die IP-Adresse des OPC UA-Servers ein, dessen Server-Schnittstelle Sie online ermitteln wollen (im Beispiel 192.168.1.1).

7. Klicken Sie auf die Schaltfläche "Ausgewählten Server finden".
   STEP 7 baut eine Verbindung zum OPC UA-Server auf und ermittelt alle Sicherheitseinstellungen (Server-Endpoints), die der Server bereithält
   STEP 7 zeigt die Endpunkte als Liste an:

8. Klicken Sie auf den Endpunkt, den Sie für eine Verbindung von STEP 7 zum OPC UA-Server nutzen wollen.

9. Wollen Sie eine gesicherte Verbindung verwenden?
   – Falls Sie einen sicheren Endpunkt ausgewählt haben, dann wählen Sie jetzt bei "Zertifikats-Ablage" den Eintrag "TIA Portal" aus.
     Und bei "Zertifikat (Client)" wählen Sie ein Client-Zertifikat für Ihren PC aus, auf dem STEP 7 (TIA Portal) gerade ausgeführt wird.
     Wenn noch kein Client-Zertifikat für Ihren PC vorliegt, dann können Sie hier im TIA Portal ein Client-Zertifikat erzeugen.
     Um ein Zertifikat für Ihren PC zu erzeugen, gehen Sie folgendermaßen vor:
     - Klicken Sie auf die Schaltfläche im Eingabefeld "Zertifikat (Client)".
     - Klicken Sie auf die Schaltfläche "Hinzufügen".
     - Bei "Zertifikatsinhaber" tragen Sie "STEP 7 (TIA Portal)" ein.
     - Bei "Verwendungszweck" wählen Sie den Eintrag "OPC UA-Client".
     - Bei "Alternativer Name des Zertifikatsinhabers (SAN)" tragen Sie unter "Wert" die IP-Adresse Ihres PCs ein, auf dem Sie gerade STEP 7 (TIA Portal) ausführen. Überschreiben Sie die bereits eingetragene IP-Adresse.
     - Falls Ihr PC eine zweite IP-Adresse verwendet, so tragen Sie diese ebenfalls ein. Wenn Ihr PC keine zweite IP-Adresse verwendet, dann löschen Sie die zweite bereits eingetragene IP-Adresse.
     - Klicken Sie auf die Schaltfläche "OK".
   – Falls Sie keinen sicheren Endpunkt ausgewählt haben, dann behalten Sie die Voreinstellung ("Keine") bei.
10. Wie wollen Sie sich anmelden?
   – Falls Sie sich als Gast beim OPC UA-Server anmelden möchten, dann übernehmen Sie die Voreinstellung bei "Benutzer-Authentifizierung".
   – Falls Sie sich mit Benutzernamen und Passwort anmelden möchten, dann wählen Sie "Benutzername und Passwort".
     Verwenden Sie den Benutzernamen und das Passwort, das bei der Konfiguration des OPC UA-Servers hinterlegt wurde, in den Eigenschaften der CPU, unter "Allgemein > OPC UA > Server > Security > Benutzer-Authentifizierung > Benutzerverwaltung".
11. Klicken Sie auf die Schaltfläche "Verbinden".
   Bei einer gesicherten Verbindung erscheint noch eine Meldung, dass Sie das Server-Zertifikat akzeptieren müssen, damit die sichere Verbindung zustande kommt. Im Meldungsfenster können Sie sich die weiteren Details zum Server-Zertifikat über einen Link anzeigen lassen.
   Dieses Standard-Windows-Fenster liefert nur Informationen zum Server-Zertifikat. Wenn Sie auf über die Schaltfläche zum Installieren des Server-Zertifikats klicken, wird das Server-Zertifikat aber nicht in den Zertifikatsspeicher des TIA Portals aufgenommen, d.h. beim nächsten Verbindungsversuch werden Sie erneut aufgefordert, das Server-Zertifikat zu akzeptieren.
   STEP 7 baut anschließend eine Verbindung zum OPC UA-Server auf und zeigt wieder den Editor für Client-Schnittstellen an.
   Im rechten Feld des Editors stellt STEP 7 die oberste Ebene des Adressraums des OPC UA-Servers dar:

12. Klicken Sie auf das kleine schwarze Dreieck neben "Objects".
   STEP 7 zeigt nun auch die Ebene unterhalb von Objects an.

13. Klicken Sie auf das kleine schwarze Dreieck neben "Productionline".
    STEP 7 zeigt nun auch die Ebene unterhalb von Productionline an.
14. Öffnen Sie nun weitere unterlagerte Ordner:

OPC UA-Server-Schnittstelle

Name des Knotens	Knotentyp	Zugriffsstufe
Server	Object	
DeviceSet	Object	
Productionline	Object	
Counters	Object	
DataBlocksGlobal	Object	
Icon	ImagePNG	RD
Data_from_OPC_UA_Clients	Object	
ProductionEnabled	Boolean	RD/WR
Data_for_OPC_UA_Clients	Object	
ProductNumber	Int16	RD
Temperature	Array of Float	RD
NewProduct	Boolean	RD
DataBlocksInstance	Object	
Icon	ImagePNG	RD
OpenDoor_DB	Object	
DeviceManual	String	RD

Alternative:
Um die Server-Schnittstelle in einem Schritt zu öffnen, klicken Sie auf das folgende Symbol:

Und STEP 7 zeigt nun die vollständig aufgeklappte Server-Schnittstelle an.
Wenn Sie nochmals auf dieses Symbol klicken, dann zeigt STEP 7 wieder nur die oberste Ebene des Adressraums des Servers an.

Mehrsprachige Texte nutzen

Im Editor für Client-Schnittstellen importieren Sie mit den OPC UA-XML-Dateien (Informationsmodelle) auch Texte, die in verschieden Sprachen angezeigt werden können. Die Mehrsprachigkeit ist optional, jeder Knoten (Node) kann hinsichtlich der angebotenen Sprachen unterschiedlich definiert sein.

In der XML-Datei sind das folgende Felder, die für unterschiedlichen Sprachen vorbereitet sein können:

- DisplayName
- Description

Beispiel für mehrsprachig definierte Texte in einer OPC UA-XML-Datei

In der dargestellten XML-Datei sind z. B. der Displayname und die Description sowohl mit einem "Default"-Text und mehreren lokalisierten Texten eingetragen.

- Default-Text ist jeweils der erste Eintrag ohne Lokalisierungsinformation
- Lokalisierter Text ist jeweils der Text hinter "Locale=" gefolgt von einem Sprachkürzel, z. B. "it-IT" für italienisch

```
<UAVariable NodeId="ns=3;i=6070" BrowseName="3:EngineeringRevision" ParentNodeId="ns=3;i=1002"
DataType="String">
    <DisplayName>EngineeringRevision</DisplayName>
    <DisplayName Locale="en-US">EngineeringRevision</DisplayName>
    <DisplayName Locale="de-DE">Revisionsstand</DisplayName>
    <Description>Revision Level of the engineering environment.</Description>
    <Description Locale="en-US">Revision Level of the engineering environment.</Description>
    <Description Locale="de-DE">Revisionsstand der Engineeringumgebung.</Description>
    <Description Locale="fr-FR">Niveau de révision de l'environnement d'ingénierie.</Description>
    <Description Locale="it-IT">Livello di revisione dell'ambiente di ingegneria.</Description>
<References>
    <Reference ReferenceType="HasTypeDefinition">i=68</Reference>
    <Reference ReferenceType="HasModellingRule">i=78</Reference>
</References>
</UAVariable>
```

Anzeige mehrsprachiger Texte

Beim Importieren einer Server-Schnittstelle werden die verfügbaren mehrsprachigen Texte intern gespeichert und auch mit dem Projekt in eine CPU geladen.

Den Text aus der OPC UA-XML-Datei zeigt der Client-Editor in den Spalten "Name des Knotens" (entspricht "DisplayName") und "Beschreibung" (entspricht "Description") an.

Welche Sprache für einen Knoten angezeigt wird, hängt von folgenden kaskadierenden Regeln ab:

- Wenn der Knoten Text in der aktuell verwendeten Editiersprache enthält, wird der Text auch in der Editiersprache angezeigt.
  (Einstellung der Editiersprache: In der Projektnavigation wählen Sie den Bereich "Sprachen & Ressourcen > Projektsprachen")
- Wenn der Knoten keinen Text in der Editiersprache enthält, aber dort ein Default-Text definiert ist (ohne Sprachkürzel), dann wird der Default-Text angezeigt.
- Spalte "Name des Knotens": Wenn auch kein Default-Text definiert ist, aber ein Text in irgendeiner anderen Sprache, dann wird der DisplayName-Text in der ersten vorliegenden Sprache angezeigt. Für Beschreibungstexte (Description) gilt diese Regel nicht.
- Wenn keine der oben genannten Bedingungen erfüllt ist, wird kein Text angezeigt.

Schnittstelle importieren

OPC UA-Server-Schnittstelle

	Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID	Beschreibung
5	Server	Object		http://opcfoundation....	
6	ServerStatus	ServerStat..	RD	http://opcfoundation....	The current status of
7	StartTime	UtcTime	RD	http://opcfoundation....	

Wenn Sie die Editiersprache wechseln, wechselt auch der mehrsprachige Text nach den oben erläuterten Regeln in der importierten Schnittstelle.

Die Knoten können Sie dann in die entsprechenden Listen (Leseliste, Schreibliste, Methodenliste) per Drag&Drop übernehmen.

In den Listen (Leseliste, Schreibliste, Methodenliste) ist kein Sprachwechsel möglich.

Übernahme der angezeigten Beschreibungstexte als Kommentar in PLC-Datentypen

Wenn Sie das Programm übersetzen, erzeugt STEP 7 automatisch PLC-Datentypen (UDTs) für jede Leseliste, für jede Schreibliste und für Eingänge bzw. Ausgänge jeder Methode. Diese UDTs haben jeweils ein Element für jeden Knoten.

Die UDTs übernehmen den Beschreibungstext als Kommentar nach den oben erläuterten Regeln. STEP 7 erzeugt den Kommentar nur in einer Sprache, wie auch die Texte in der OPC UA-Server-Schnittstelle nur in einer Sprache angezeigt werden können.

Regeln für den Zugriff auf Strukturen

Im Folgenden sind die Regeln für den Zugriff auf Strukturen erläutert. Beachten Sie diese Regeln beim Lesen und Schreiben von Werten kompletter Strukturen, die ein OPC UA-Server zur Verfügung stellt.

Wie der Client der S7-1500 CPU auf Strukturen zugreift

Der OPC UA-Client der S7-1500 CPU nutzt für den performanten Zugriff auf Strukturen zur Laufzeit weder TypeDictionaries noch DataTypeDefinition-Attribute, die ein Server zur Auflösung dieser Strukturen anbietet.

Daher sind die Möglichkeiten des OPC UA-Clients zur Überprüfung von Strukturelementen zur Laufzeit des Anwenderprogramms im Client begrenzt.

Regeln für den Zugriff auf Strukturen

Wenn Sie zur Projektierung der Lese- und Schreiblisten die Client-Schnittstellen verwenden (Verbindungsparametrierung) und die PLC-Datentypen passend zum importierten bzw. online ermittelten Adressmodell des Servers zuweisen, dann funktionieren Lese- und Schreibzugriffe auf Strukturen zur Laufzeit problemlos.

Denn die Projektierung mithilfe der Client-Schnittstelle sorgt automatisch dafür, dass die Reihenfolge und die Datentypen der Strukturelemente client- und server-seitig aufeinander abgestimmt sind.

Empfehlung: Aktualisieren Sie eine S7-1500 CPU (als Server) auf die aktuellen Firmware-Version (z. B. V2.0 > V2.5.2 oder höher).

Zur Laufzeit überprüft der OPC UA-Client nur die Gesamtlänge des übermittelten Werts; detailliertere Prüfungen sind nicht möglich.

Für die Zuweisung von OPC UA-Strukturen zu PLC-Variablen bzw. DB-Variablen gelten die Mapping-Regeln (siehe Mapping von Datentypen (Seite 1197)). Nicht aufgeführte Datentypen (z. B. OPC UA Byte-Strings) werden nicht unterstützt.

Beispiel für eine fehlerfreie Zuweisung von Strukturelementen

In der importierten Nodeset-Datei (XML-Export) ist die Struktur folgendermaßen definiert:

opcUaStruct	Object
allOk	Boolean
myOPCstruct1	myOPCUAstruct
varA	Int64
VarB	Byte
nestedStructZ	"myOPCUAstruct"."nestedStructZ"
varD	Float
varE	Double
varC	Double
DeviceManual	String

Die in der Leseliste abgebildete Struktur stimmt sowohl in der Reihenfolge als auch von den zugewiesenen Datentypen mit dem entsprechenden Knoten der Nodeset-Datei überein.

myOPCstruct1	myOPCUAstruct
varA	LINT
VarB	USINT
nestedStructZ	"myOPCUAstruct"."nestedStructZ"
varD	REAL
varE	LREAL
varC	LREAL

Wenn die Struktur sich jetzt auf dem Server ändert, z. B. varA und varB werden vertauscht, und die Leseliste im Client bleibt gleich, dann stimmt die Zuweisung nicht mehr:

- Die Gesamtlänge der Daten bleibt gleich (es hat sich nur die Reihenfolge geändert)
- Der Aufbau der Struktur ist bei Client und Server unterschiedlich!

⚠ WARNUNG

Keine Fehlermeldung bei unterschiedlichem Strukturaufbau zwischen Client und Server

Wenn die Strukturen bei Client und Server nicht übereinstimmen, dann erzeugt diese Regelverletzung möglicherweise keinen Fehler beim Übersetzen und auch nicht zur Laufzeit.

Achten Sie darauf, dass sich die projektierten Zuweisungen bei Strukturen zur Laufzeit nicht ändern. Projektieren Sie bei Bedarf die Zuweisung in den Lese- und Schreiblisten neu!

Verbindungsparametrierung nutzen

Verbindungen anlegen und parametrieren

Mit den Anweisungen für OPC UA-Clients erstellen Sie ein Anwenderprogramm, das Daten mit einem OPC UA-Server austauscht. Dafür sind eine Reihe von Systemdatentypen erforderlich.

Um die Arbeit mit diesen Systemdatentypen zu vereinfachen, verfügt STEP 7 (TIA Portal) ab der Version 15.1 über eine Verbindungsparametrierung für OPC UA-Clients.

Die Verbindungsparametrierung ist eine Option, die Sie nutzen können, aber nicht müssen. Sie können die erforderlichen Systemdatentypen auch manuell anlegen.

Damit die Beschreibung nachvollziehbar wird, verwenden wir ein Beispiel, siehe Beschreibung des Beispiels (Seite 1285).

Verbindungsparametrierung öffnen

Um die Verbindung zu einem OPC UA-Server zu parametrieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Bereich "OPC UA-Kommunikation" in der Projektnavigation auf die Client-Schnittstelle, die Sie parametrieren wollen.
   Für die Beispiel-Konfiguration: Doppelklicken Sie auf die Client-Schnittstelle "Productionline":

   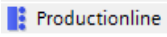

   Wie Sie eine Client-Schnittstelle anlegen, ist im Kapitel "Client-Schnittstelle anlegen (Seite 1287)" beschrieben.
2. Klicken Sie auf das Register "Eigenschaften" (Inspektorfenster), falls das Register nicht bereits angezeigt wird.
   STEP 7 zeigt nun die Verbindungsparametrierung für die Anweisungen des OPC UA-Clients an.
   Das Register "Allgemein" ist geöffnet.
3. Klicken Sie auf das Register "Konfiguration".

Verbindungsparameter einstellen

Im Register "Konfiguration" stellen Sie die Verbindung zum OPC UA-Server ein.

1. Wählen Sie einen aussagekräftigen Namen für die Session. Für das Beispiel wählen Sie den Namen "OPC UA Connection to Productionline".
2. Tragen Sie im Feld "Adresse" die IP-Adresse des OPC UA-Servers ein, zu dem Ihr Anwenderprogramm, das als OPC UA-Client arbeitet, eine Verbindung aufbauen soll.
   In der Beispielkonfiguration besitzt die CPU, die die Fertigungslinie steuert, die IP-Adresse "192.168.1.1". Es soll eine Verbindung zu dem OPC UA-Server dieser CPU aufgebaut werden, deshalb tragen Sie diese IP-Adresse im Feld "Adresse" ein.
3. Falls der OPC UA-Server nicht den Standard-Port 4840 verwendet, müssen Sie hier die Portnummer einfügen.
   Tragen Sie in das Feld zum Beispiel die Nummer 48040 ein, falls der OPC UA-Server, zu dem Sie eine Verbindung aufbauen wollen, diese Portnummer verwendet.
   Für das Beispiel übernehmen wir die Voreinstellung "4840", da der OPC UA-Server des Beispiels über den Port 4840 erreichbar ist.

4. Tragen Sie einen Pfad innerhalb des OPC UA-Servers ein, um den Zugriff auf diesen Pfad zu beschränken.
   Wenn Sie einen Pfad angeben, wird dieser automatisch an den Eintrag "ServerEndpointUrl" im Konfigurations-DB für die Client-Schnittstelle eingetragen. Der Eintrag besteht dann aus den Komponenten "OPC-Schema Präfix", "IP-Adresse", "Port-Nummer" und "Server-Pfad", z. B.: "opc.tcp://192.168.0.10:4840/Beispiel/Pfad".
   Die Angabe ist optional. Einige Server bauen jedoch nur dann eine Verbindung auf, wenn ein Server-Pfad angegeben ist.
5. Außerdem übernehmen Sie die Voreinstellungen für Session-Timeout (30 Sekunden) und Monitoring-Time (5 Sekunden).
   Das folgende Bild zeigt die Verbindungsparameter des Beispiels:

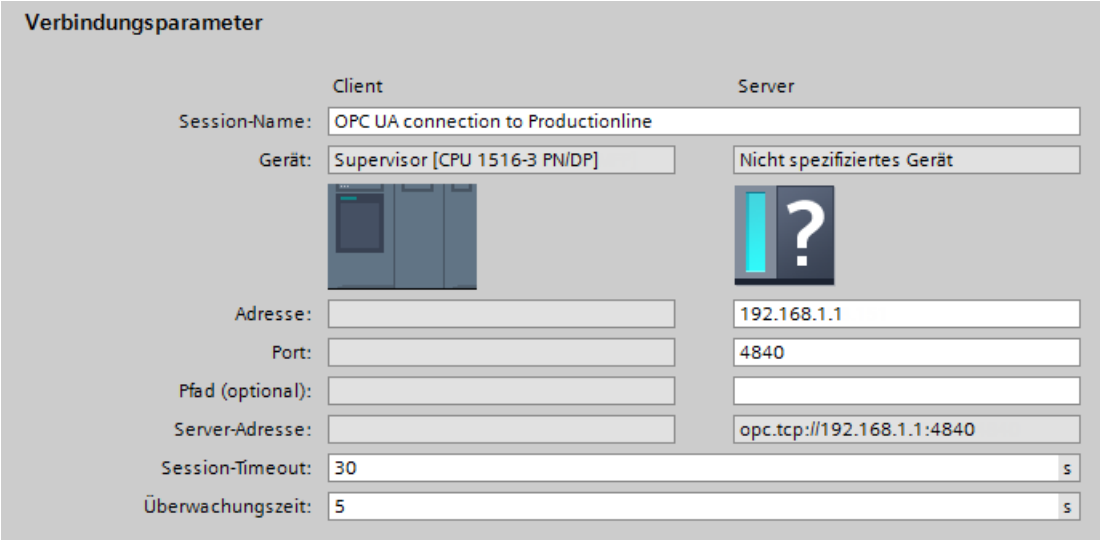

Security Parameter einstellen

1. Klicken Sie auf den Bereich "Security" im Register "Konfiguration".
   In diesem Bereich befinden sich alle Sicherheitseinstellungen für die Verbindung zum OPC UA-Server.

Folgende Einstellungen sind möglich:

Bereich "Allgemein"

Security-Modus:

Wählen Sie aus der Klappliste den Sicherheitsmodus, den die Verbindung zum OPC UA-Server erfüllen muss.

Wenn der Server den gewählten Modus nicht erfüllt, dann wird keine Session aufgebaut.

Die folgenden Einstellungen stehen zur Wahl:

- Keine Security: Keine gesicherte Verbindung!
- Signieren: OPC UA-Server und -Client signieren die Datenübertragung (alle Nachrichten): Manipulationen sind so erkennbar.
- Signieren & Verschlüsseln: OPC UA-Server und -Client signieren und verschlüsseln die Datenübertragung (alle Nachrichten):

Security Policy:

Stellen Sie die Verschlüsselungstechniken für das Signieren und Verschlüsseln von Nachrichten ein.

Die folgenden Einstellungen sind möglich:

- Keine Security
- Basic128Rsa15
- Basic256
- Basic256Sha256

Wenn Sie eine gesicherte Verbindung konfigurieren, dann müssen Sie die folgenden Punkte beachten:

- Für eine gesicherte Verbindung ist ein Zertifikat für den Client erforderlich.
- Dieses Client-Zertifikat müssen Sie dem Server bekanntmachen.

Wie Sie dazu vorgehen, ist im Kapitel "Handling der Client- und Server-Zertifikate (Seite 1234)" unter "Zertifikat des OPC UA-Clients" beschrieben.

Bereich "Zertifikate"

Client-Zertifikat:

Das Zertifikat bestätigt die Authentizität des OPC UA-Clients.

Um ein Zertifikat auszuwählen, klicken Sie dazu auf das folgende Symbol

STEP 7 zeigt eine Liste mit Zertifikaten an.

Wählen Sie davon das Zertifikat aus, das Sie dem Server bekanntgemacht haben.

Klicken Sie auf das Symbol mit dem grünen Häkchen:

Oder erzeugen Sie ein neues Zertifikat. Dazu klicken Sie auf das Symbol "Hinzufügen".

Wenn Sie ein neues Zertifikat erzeugen, dann müssen Sie dieses Zertifikat dem Server bekanntmachen.

Bereich "Benutzer-Authentifizierung"

Bei Benutzer-Authentifizierung sind die folgenden Einstellungen möglich:

- Gast
- Benutzername und Passwort
- Benutzer (TIA Portal - Security-Einstellungen)

Weitere Informationen siehe Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 1249).

Sprachen einstellen

UA-Variablen vom Typ String können bei OPC UA lokalisiert werden, d. h. Texte (Werte für die UA-Variable) können beim Server in verschiedenen Landessprachen vorliegen. Zum Beispiel können für DisplayName (Name des Knotens) und Description (Beschreibung) lokalisierte Texte vorliegen.

Im Bereich "Sprachen" des Registers "Konfiguration" haben Sie die Möglichkeit, die Sprache der vom Server zurückgelieferten Texte in folgender Weise zu beeinflussen:

Geben Sie im Bereich "Sprachen" eine Reihenfolge von Sprachen ein, die der Server beim Verbindungsaufbau dem Client übermittelt.

Die Sprache bzw. die damit verbundene Lokale ID ("Sprachkürzel"), die Sie in der ersten Zeile eingeben, ist die vom Client bevorzugte Sprache.

- Wenn der Server die UA-Variable in der gewünschten Sprache liefern kann, dann wird sie dem Client übermittelt.
- Wenn der Server die UA-Variable nicht in der gewünschten Sprache liefern kann, dann prüft er, ob er die UA-Variable in der Sprache liefern kann, die Sie in der zweiten Zeile eingegeben haben (1. Ersatzsprache).
- So arbeitet der Server die Liste ab und wenn er weder die gewünschte Sprache noch eine Ersatzsprache liefern kann, dann liefert er die Default-Sprache.

Siehe auch

Handling der Client-Zertifikate der S7-1500 CPU (Seite 1306)

Handling der Client-Zertifikate der S7-1500 CPU

Woher kommt das Zertifikat des Clients?

Wenn Sie den OPC UA-Client einer S7-1500 CPU nutzen (OPC UA-Client aktiviert), dann können Sie mit STEP 7 ab V15.1 für diese Clients Zertifikate erzeugen wie in den folgenden Abschnitten beschrieben ist.

Wenn Sie UA-Clients von Herstellern oder der OPC Foundation verwenden, wird bei der Installation oder beim ersten Programmaufruf automatisch ein Client-Zertifikat erzeugt. Diese Zertifikate müssen Sie über den globalen Zertifikatsmanager in STEP 7 importieren und für die jeweilige CPU verwenden.

Wenn Sie selbst einen OPC UA-Client programmieren, dann können Sie Zertifikate programmtechnisch erstellen. Oder Sie erzeugen Zertifikate mit Tools, zum Beispiel mit OpenSSL oder dem Zertifikate-Generator der OPC Foundation:

- Wie Sie bei OpenSSL vorgehen, lesen Sie hier: "PKI-Schlüsselpaare und Zertifikate selbst erzeugen (Seite 1209)".
- Wie Sie mit dem Zertifikate-Generator der OPC Foundation arbeiten, lesen Sie hier: "Selbstsignierte Zertifikate erzeugen (Seite 1208)".

Zertifikat des OPC UA-Clients der S7-1500 CPU

Eine gesicherte Verbindung zwischen OPC UA-Server und einem OPC UA-Client kommt nur dann zu Stande, wenn der Server das Zertifikat des Clients als vertrauenswürdig einstuft.

Dazu müssen Sie dem Server das Client-Zertifikat bekanntmachen.

Die folgenden Abschnitte beschreiben, wie Sie zunächst für den OPC UA-Client der S7-1500 CPU ein Zertifikat erzeugen und es dann dem Server zur Verfügung stellen.

1. Zertifikat für den Client erzeugen und exportieren

Für eine gesicherte Verbindung müssen Sie ein Client-Zertifikat erzeugen und - falls Server und Client sich in unterschiedlichen Projekten befinden - das Zertifikat exportieren.

Wenn Client und Server sich im selben Projekt befinden, entfällt das Exportieren des Client-Zertifikats und das anschließende Importieren.

Voraussetzungen

Die IP-Schnittstelle der CPU ist konfiguriert, eine IP-Adresse vorhanden.

Hintergrund: Unter "Alternativer Name des Zertifikatsinhabers (SAN)" wird die IP-Adressen eingetragen, unter der die CPU in Ihrer Anlage erreichbar ist.

OPC UA-Client-Zertifikat erzeugen

Die einfachste Möglichkeit, für eine S7-1500 CPU ein Client-Zertifikat zu erzeugen, ist über die Projektierung einer Client-Schnittstelle.

Die Projektierung der Client-Schnittstelle sieht die Auswahl oder Erzeugung eines Client-Zertifikats vor, siehe Verbindungen anlegen und parametrieren (Seite 1302).

Alternativ können Sie das Client auch folgendermaßen erzeugen:

1. In der "Projektnavigation" wählen Sie die CPU aus, die als Client fungiert.
2. Doppelklicken Sie auf "Gerätekonfiguration"
3. In den Eigenschaften der CPU klicken Sie auf "Schutz & Security > Zertifikatsmanager".
4. In der Tabelle "Gerätezertifikate" doppelklicken Sie auf "<neu hinzufügen>".
   STEP 7 öffnet einen Dialog.
5. Klicken Sie auf die Schaltfläche "Hinzufügen".
6. Bei "Verwendungszweck" wählen Sie aus der Liste den Eintrag "OPC UA-Client".
7. Klicken Sie auf "OK".
   STEP 7 zeigt nun das Client-Zertifikat in der Tabelle "Gerätezertifikate" an.
8. Falls der Server in einem anderen Projekt liegt: Klicken Sie mit der rechten Maustaste auf diese Zeile und wählen Sie das Kontextmenü "Zertifikat exportieren".
9. Wählen Sie ein Verzeichnis aus, in dem Sie das Client-Zertifikat speichern.

2. Das Client-Zertifikat dem Server bekanntmachen

Das Client-Zertifikat müssen Sie dem Server zur Verfügung stellen, damit eine gesicherte Verbindung aufgebaut werden kann.

Dazu gehen Sie folgendermaßen vor:

1. Falls der Client in einem anderen Projekt konfiguriert wurde und Sie das Client-Zertifikat dort erstellt und exportiert haben:
   - Aktivieren Sie im lokalen Zertifikatsmanager des Servers die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". Dadurch ist der globale Zertifikatsmanager verfügbar.
     Sie finden diese Option in den Eigenschaften der CPU, die als Server dient, unter "Schutz & Security > Zertifikatsmanager".
   - Falls dieses Projekt noch nicht geschützt ist, dann klicken Sie in der Projektnavigation von STEP 7 unter "Security-Einstellungen > Einstellungen" auf die Schaltfläche "Dieses Projekt schützen" und melden Sie sich an.
     STEP 7 zeigt nun in der Projektnavigation unter "Security-Einstellungen" der Eintrag "Globale Security-Einstellungen" an.
   - Doppelklicken Sie auf "Globale Security-Einstellungen".
   - Doppelklicken Sie auf "Zertifikatsmanager".
     STEP 7 öffnet den globalen Zertifikatsmanager.
   - Klicken Sie auf das Register "Gerätezertifikate".
   - Klicken Sie mit der rechten Maustaste im Register auf eine freie Fläche (nicht auf ein Zertifikat).
   - Wählen Sie das Kontextmenü "Importieren".
     Der Dialog zum Importieren von Zertifikaten wird angezeigt.
   - Wählen Sie das Client-Zertifikat aus, dem der Server vertrauen soll.
   - Klicken Sie auf die Schaltfläche "Öffnen", um das Zertifikat zu importieren.
     Das Zertifikat des Clients ist nun im globalen Zertifikatsmanager enthalten. Merken Sie sich die ID des gerade importierten Client-Zertifikats.
2. Klicken Sie nun in den Eigenschaften der CPU, die als Server dient, auf das Register "Allgemein".
3. Klicken Sie auf den Bereich "OPC UA > Server > Security > Secure Channel".
4. Scrollen Sie im Dialog "Secure Channel" nach unten zum Abschnitt "Vertrauenswürdige Clients".
5. Doppelklicken Sie in der Tabelle auf die leere Zeile mit "<neu hinzufügen>". In der Zeile wird eine Schaltfläche mit drei Punkten angezeigt.
6. Klicken Sie auf diese Schaltfläche.
7. Wählen Sie das vorbereitete Client-Zertifikat aus.
8. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
9. Übersetzen Sie das Projekt.
10. Laden Sie die Konfiguration in die S7-1500 CPU (Server).

Ergebnis

Der Server vertraut nun dem Client. Wenn außerdem das Server-Zertifikat als vertrauenswürdig gilt, dann können Server und Client eine gesicherte Verbindung aufbauen.

Authentifizierung des Benutzers

Sie können in der OPC UA-Client-Schnittstelle der S7-1500 einstellen, wie sich ein Benutzer des OPC UA-Clients legitimieren muss, wenn er auf den Server zugreifen will. Dazu müssen Sie in der Projektnavigation der gewünschten S7-1500 CPU unter "OPC UA-Kommunikation > Client-Schnittstellen" die entsprechende Client-Schnittstelle auswählen und im Inspektorfenster unter "Eigenschaften > Konfiguration > Security" die Art der Benutzer-Authentifizierung auswählen.

Arten der Benutzer-Authentifizierung

Für die Benutzer-Authentifizierung stehen Ihnen folgenden Möglichkeiten zur Verfügung:

- Gast
  Der Benutzer muss seine Berechtigung nicht nachweisen (anonymer Zugang). Die CPU erzeugt zu diesem Zweck für den Anwender eine anonyme Session und der OPC UA-Server überprüft nicht die Berechtigung des Client-Anwenders.
- Benutzername und Passwort
  Der Benutzer muss seine Berechtigung nachweisen (kein anonymer Zugang). Der OPC UA-Server überprüft, ob der Client-Anwender berechtigt ist, auf den Server zuzugreifen. Als Nachweis gilt der Benutzername mit dem richtigen Passwort. Diese Eingaben können nicht von der Client-Schnittstelle überprüft werden, daher werden hier alle Werte als gültig akzeptiert.

  Hinweis

  Benutzername und Passwort speichert STEP 7 unverschlüsselt im Datenbaustein/ Instanzdatenbaustein. Empfehlung: Verwenden Sie die Benutzer-Authentifizierung "Benutzer (TIA Portal - Security-Einstellungen)".

- Benutzer (TIA Portal - Security-Einstellungen)
  Sie können einen Benutzernamen aus der Liste der im Projekt eingetragenen Benutzer für die Authentifizierung eintragen. Die Namen der eingetragenen Benutzer des aktuellen Projekts finden Sie in der Benutzerverwaltung in der Projektnavigation unter "Security-Einstellungen > Benutzer und Rollen". Dort können Sie auch weitere Benutzer eintragen. Sie können auch einen Namen eintragen, der nicht in der Benutzerverwaltung des Projekts steht oder auch das Feld leer lassen. Dies ist dann notwendig, wenn der entsprechende Benutzername erst zur Laufzeit aus einer anderen Quelle kommt, beispielsweise über HMI oder von einem anderen OPC UA-Client.

Security Policy "Keine Security" und Authentifizierung über Benutzername und Passwort

Folgende Kombination können Sie einstellen:

Security Policy = "Keine Security" und Authentifizierung über Benutzername und Passwort.

- Der OPC UA-Server der S7-1500 unterstützt diese Kombination. OPC UA-Clients können sich verbinden und die Authentifizierungsdaten verschlüsseln oder auch nicht.
- Der OPC UA-Client der S7-1500 CPU unterstützt ebenfalls diese Kombination: Zur Laufzeit verbindet er sich aber nur dann, wenn er die Authentifizierungsdaten verschlüsselt über die Leitung schicken kann!

Folge: Bei folgender Konfiguration kann keine Verbindung zur Laufzeit hergestellt werden:

- S7-1500 als OPC UA-Client
- OPC UA-Server, der bei eingestellter Security Policy "Keine Security" (= "none") keine Verschlüsselung der Authentifizierungsdaten unterstützt

Siehe auch

Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 1249)

Parametrierte Verbindung nutzen

Einleitung

Dieses Kapitel zeigt, wie Sie eine parametrierte Verbindung bei OPC UA-Anweisungen verwenden (Dritter Schritt).

Voraussetzungen

- Sie haben eine Client-Schnittstelle angelegt und dieser Schnittstelle PLC-Variablen und -Methoden hinzugefügt, siehe (Erster Schritt (Seite 1287)).
- Sie haben eine Verbindung zu einem OPC UA-Server parametriert (Zweiter Schritt (Seite 1302)).

Übersicht

Um Daten von einem OPC UA-Server zu lesen oder Daten zu einem OPC UA-Server zu schreiben, verwenden Sie die folgenden Anweisungen:

- OPC_UA_Connect
- OPC_UA_NamespaceGetIndexList
- OPC_UA_NodeGetHandleList
- OPC_UA_ReadList oder OPC_UA_WriteList
- OPC_UA_NodeReleaseHandleList
- OPC_UA_Disconnect

Reihenfolge der OPC UA-Anweisungen

Das folgende Bild zeigt die Reihenfolge, in der die OPC UA-Anweisungen in einem Anwenderprogramm aufgerufen werden, um damit PLC-Variablen zu lesen oder zu schreiben:

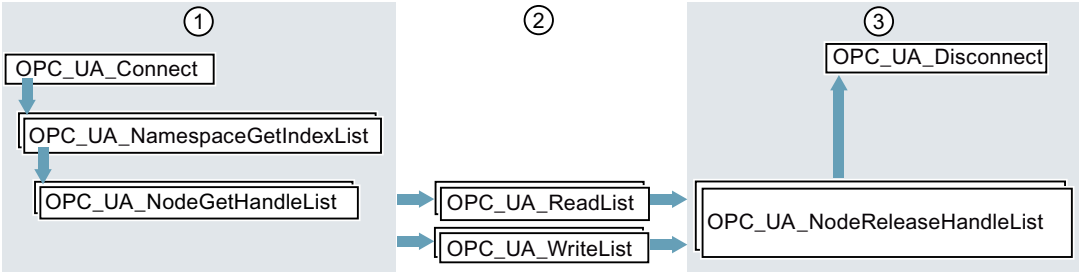

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum "Clean-up" nach durchgeführten Lese- oder Schreibvorgängen
Die Anweisung "OPC_UA_NodeReleaseHandleList" kann entfallen, wenn anschließend gleich "OPC_UA_Disconnect" aufgerufen wird.

STEP 7 (TIA Portal) versorgt die Parameter dieser Anweisungen automatisch, wenn Sie eine Client-Schnittstelle und eine parametrierte Verbindung zu einem OPC UA-Server verwenden.

Wie Sie dabei vorgehen, zeigt das folgende Kapitel.

Client-Schnittstelle und parametrierte Verbindung verwenden

Um eine parametrierte OPC UA-Verbindung zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie Ihr Anwenderprogramm im TIA Portal.
2. Ziehen Sie per Drag & Drop die Anweisung "**UA_Connect**" in den Programmeditor.
   Sie finden die Anweisung unter "Anweisungen > Kommunikation > OPC UA" im TIA Portal.

3. Wählen Sie eine Aufrufoption für die Anweisung
   Das Beispiel verwendet eine Multi-Instanz.
   STEP 7 stellt die Anweisung im Programmeditor dar.
   Der Editor für die Programmiersprache Funktionsplan (FUP) verwendet die folgende Darstellung:

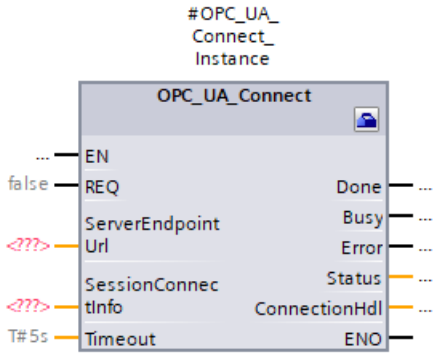

   Der Editor für die Programmiersprache Kontaktplan (KOP) zeigt die Anweisung ähnlich an.

4. Klicken Sie im Editor für FUP oder KOP auf das Symbol für einen Werkzeugkasten.
   Das Symbol befindet sich im Kopf der Anweisung:

   Falls Sie den Editor für AWL oder SCL verwenden: Klicken Sie auf das kleine grüne Rechteck unter dem ersten Zeichen des Instanznamens:

```
#OPC_UA_Connect_Instance
```

   Das Beispiel (Seite 1285) verwendet "#OPC_UA_Connect_Instance" als Instanzname.
   STEP 7 stellt nun die Eigenschaften der Anweisung in einem eigenen Dialog dar.

5. Wählen Sie bei "Client interface" das Client-Interface aus, das Sie für die Anweisung verwenden wollen.
   Im Beispiel wählen wir das Client-Interface "ProductionLine".
   STEP 7 verschaltet nun das Client-Interface "ProductionLine" mit den Parametern der Anweisung OPC_UA_Connect:

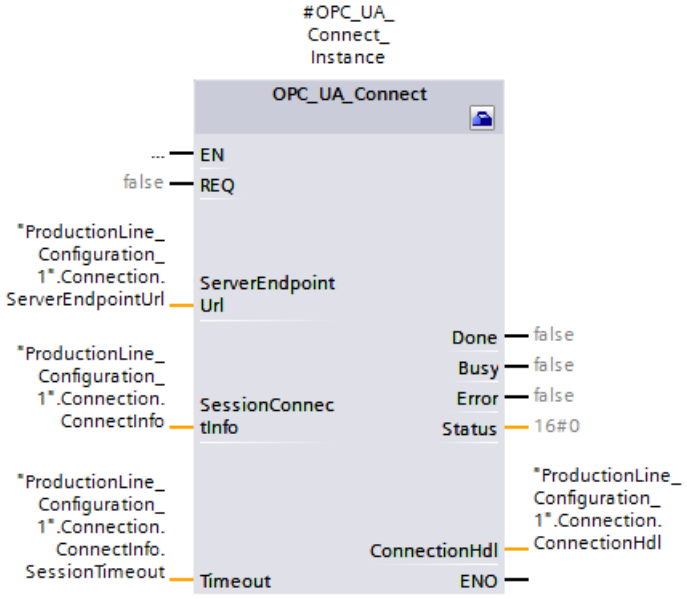

   "ProductionLine" ist das Interface, das der OPC UA-Client des Beispiels (Seite 1285), für den Datenaustausch mit dem OPC UA-Server "ProductionLine" verwendet.

6. Ziehen Sie per Drag & Drop die Anweisung "**UA_NamespaceGetIndexList**" in den Programmeditor.
   Sie finden die Anweisung unter "Anweisungen > Kommunikation > OPC UA" im TIA Portal.
   Wählen Sie die Aufrufoption "Multi-Instanz"
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie das Client-Interface aus, das Sie verwenden wollen (Im Beispiel "ProductionLine").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung "OPC_UA_NamespaceGetIndexList":

7. Ziehen Sie per Drag & Drop die Anweisung "**UA_NodeGetHandleList**" in den Programmeditor.
   Wählen Sie die Aufrufoption "Multi-Instanz".
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie das Client-Interface aus, das Sie verwenden wollen. Das Beispiel nutzt das Client-Interface "ProductionLine".
   Wählen Sie unter "Data access > Read/Writelist" die Leseliste aus, die Sie verwenden wollen (im Beispiel die Leseliste "Product").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung "OPC_UA_NodeGetHandleList":

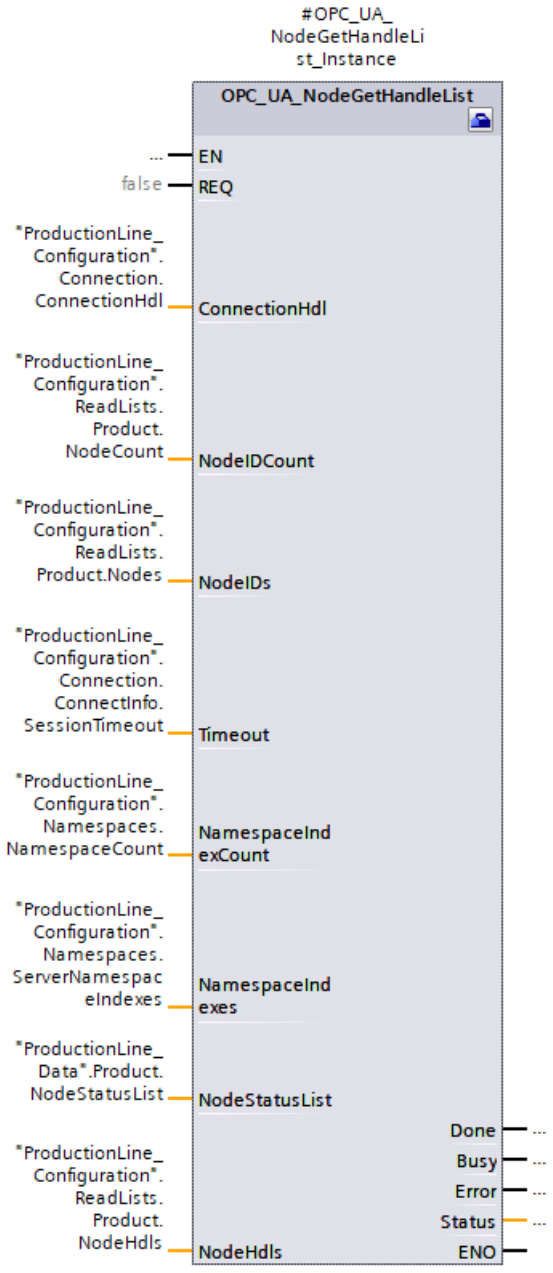

Falls Sie Daten zu einem OPC UA-Server schreiben wollen, dann wählen Sie unter "Data access > Read/Writelist" die Schreibliste aus, die Sie verwenden wollen (im Beispiel wäre das die Schreibliste "ProductionStatus").

8. Ziehen Sie per Drag & Drop die Anweisung "**UA_ReadList**" in den Programmeditor.
   Wählen Sie die Aufrufoption "Multi-Instanz"
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie das Client-Interface aus, das Sie verwenden wollen. Das Beispiel nutzt das Client-Interface "ProductionLine".
   Wählen Sie unter "Data access > Readlist" die Leseliste aus, die Sie verwenden wollen (im Beispiel die Leseliste "Product").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung OPC_UA_ReadList.
   Falls Sie Daten zu einem OPC UA-Server schreiben wollen, dann verwenden Sie die Anweisung "**OPC_UA_Write**" und wählen Sie unter "Data access > Writelist" die Liste der Variablen, die Sie zum Server senden wollen (im Beispiel die Schreibliste "ProductionStatus").
9. Falls Sie in Ihrem Anwenderprogramm mit unterschiedlichen Leselisten oder Schreiblisten arbeiten sollten, die Sie programmgesteuert verwenden, ziehen Sie per Drag&Drop die Anweisung "**UA_NodeReleaseHandleList**" in den Programmeditor.
   Wählen Sie das Client-Interface aus, das Sie verwenden wollen.
   Wählen Sie nun eine Lese- oder Schreibliste aus, die Sie frei geben wollen: Geben Sie nur Lese-/ oder Schreiblisten frei, die Sie selten nutzen, da das erneute Registrieren zeitintensiv ist.
   Wiederholen Sie dann die Schritte ab Schritt 7 mit der Anweisung "UA_NodeGetHandleList".
10. Ziehen Sie per Drag & Drop die Anweisung "**UA_Disconnect**" in den Programmeditor.
    Wählen Sie die Aufrufoption "Multi-Instanz".
    Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
    Wählen Sie das Client-Interface aus, das Sie verwenden wollen. Das Beispiel nutzt das Client-Interface "ProductionLine".
    STEP 7 verschaltet nun automatisch alle Parameter der Anweisung OPC_UA_Disconnect.

Unterstützte Anweisungen

Bei den folgenden Anweisungen versorgt STEP 7 automatisch die Parameter, wenn Sie ein Client-Interface und eine parametrierte Verbindung zu einem OPC UA-Server nutzen:

- OPC_UA_Connect
- OPC_UA_NamespaceGetIndexList
- OPC_UA_NodeGetHandleList
- OPC_UA_MethodGetHandleList
- OPC_UA_MethodReleaseHandleList
- OPC_UA_ReadList
- OPC_UA_WriteList
- OPC_UA_MethodCall
- OPC_UA_NodeReleaseHandleList
- OPC_UA_Disconnect

1.4.1.16 Webserver konfigurieren

Wissenswertes zum Webserver

Einführung

Der Webserver ermöglicht Ihnen die Überwachung und Verwaltung der CPU durch berechtige Nutzer über ein Netzwerk. Auswertungen und Diagnose sind somit über große Entfernungen möglich. Dazu ist nur ein Webbrowser erforderlich.

Meldungen und Statusinformationen werden auf HTML-Seiten angezeigt.

Webbrowser

Für den Zugriff auf die HTML-Seiten der CPU benötigen Sie einen Webbrowser.

Die Information, mit welchen Webbrowsern und Webbrowser-Versionen der Webserver getestet wurde, erhalten Sie in den entsprechenden Handbüchern, die unten aufgeführt und verlinkt sind.

Hinweis

Wenn Sie über einen Kommunikationsprozessor (CP) auf den Webserver der CPU zugreifen, stellen Sie sicher, dass der Cache (Temporäre Internetdateien) in Ihrem Browser aktiviert ist. Wählen Sie in den Cache-Einstellungen Ihres Browsers die Option "Automatisch".

Deaktivierter Cache und andere Einstellungen als die Option "Automatisch" in Ihrem Browser können zu langen Zugriffszeiten und unvollständiger Darstellung führen.

Informationen über den Webserver auslesen

Aus der CPU sind die im Folgenden aufgezählten Informationen auslesbar. Die Verfügbarkeit der jeweiligen Webseiten hängt von der CPU und ihrer Firmware-Version ab.

Dokumentation zu S7-300 (http://support.automation.siemens.com/WW/view/de/12996906)

Dokumentation zu S7-400 (http://support.automation.siemens.com/WW/view/de/44444467)

Dokumentation zu S7-1500 (http://support.automation.siemens.com/WW/view/de/59193560)

Im Folgenden finden Sie eine Übersicht der Informationen, die über die einzelnen Webseiten des Webservers der CPU S7-1500 angeboten werden.

Information	Beschreibung
Startseite mit allgemeinen CPU-Informationen	Die Startseite liefert eine Übersicht der generellen Informationen der CPU, den CPU-Namen, den CPU-Typ und Basisinformationen zum aktuellen Betriebszustand.
Diagnose: Identifikationsinformationen	Anzeige der statischen Identifikationsinformationen, wie die Seriennummer, Artikelnummer und Versionsnummern.
Diagnose: Speicher	Aktuelle Werte zum derzeit benutzten Speicherplatz
Inhalt des Diagnosepuffers	Anzeige des Diagnosepuffer-Inhalts mit den jüngsten Einträgen zuerst.
Baugruppenzustand	Anzeige, ob die Komponenten einer Station in Ordnung sind, ob z. B. Wartungsanforderungen vorliegen oder Komponenten nicht erreichbar sind.
Firmware-Update	Als Benutzer mit entsprechenden Zugriffsrechten haben Sie die Möglichkeit, die Firmware von CPUs, des Displays, oder einzelner zentraler oder dezentraler Module mithilfe einer Update-Datei zu aktualisieren.
Meldungen	Anzeige von projektorientierten Meldetexten mit Filter und Sortiermöglichkeit.
Informationen zur Kommunikation	Anzeige der Kommunikationsverbindungen, Anzeige der Ressourcen sowie Schnittstellenparameter und Portstatistik.
Topologie	Anzeige der Topologie mit den Ansichten grafisch, tabellarisch und Statusübersicht für vernetzte PROFINET-Teilnehmer.
Variablenstatus	Anzeige des Status von Operanden des Anwenderprogramms zum Beobachten der Werte. Tipp: Beim Verlassen der Variablenstatus-Seite werden die vorgenommenen Eingaben nicht gespeichert. Wenn Sie dieselben eingegebenen Variablen später erneut beobachten wollen, erzeugen Sie zu der Seite "Variablenstatus" ein Lesezeichen in Ihrem Webbrowser. Sonst müssen Sie die Variablen bei Wiederaufruf der Seite erneut eingeben.
Beobachtungstabellen	Anzeige der mit STEP 7 erzeugten Beobachtungstabellen mit der Option "Webserver".
Anwenderseiten (falls solche Webseiten/Webapplikationen projektiert und geladen wurden)	Die Anwenderseiten liefern eine Liste der Webseiten mit kundenspezifischen Webapplikationen. Mit dem ersten Aufruf der Anweisung "WWW" im Anwenderprogramm wird der Link zur Anwenderseite auf dieser Webseite angezeigt. Durch einen Klick auf den Link wird die Anwenderseite in einem neuen Fenster gestartet.

Information	Beschreibung
Filebrowser	S7-1500: Der Filebrowser listet Dateien und Verzeichnisse auf, die sich auf der SIMATIC Memory Card befinden. Dateien können abhängig vom Dateityp heruntergeladen, gelöscht, umbenannt oder hochgeladen werden. Verzeichnisse können nur erstellt, gelöscht oder umbenannt werden. Auf die Verzeichnisse "DataLogs" und "Backups" besteht über den Filebrowser nur lesender Zugriff. Systemdateien werden im Filebrowser nicht angezeigt und können somit nicht geändert oder gelöscht werden. Systemdateien sind die Auftragsdatei und alle speziellen Verzeichnisse inklusive deren Inhalt, auf die von der Auftragsdatei verwiesen wird.
Data Logs	Anzeige und Abruf der über das Anwenderprogramm erzeugten DataLogs. Achtung: Wenn Sie große DataLog Dateien über diese Webseite öffnen, können sich die Bearbeitungszeiten der DataLog-Anweisungen, die diese DataLog-Datei bearbeiten, stark erhöhen! Die Bearbeitungszeit der Prioritätsklasse (OB, in dem die DataLog-Anweisung aufgerufen wird) erhöht sich zwar nicht, aber der Ausgangsparameter BUSY ist für die Dauer der Bearbeitung gesetzt!
Online-Sicherung	S7-1500 (ab Firmware V2.0): Projektierung der CPU sichern und wiederherstellen. Mit den entsprechenden Zugriffsrechten können Sie die Projektierung einer CPU über den Webserver sichern. Bei Bedarf können Sie zu einem späteren Zeitpunkt diese Projektierung wiederherstellen. Sie können beliebig viele Sicherungen anlegen und so unterschiedliche Projektierungen für eine CPU vorhalten.
Motion Diagnose	S7-1500 (ab Firmware V2.0): Zeigt Ihnen Status, Fehler, Technologie-Alarme und die aktuellen Werte von Technologieobjekten (TOs) an, z. B. Drehzahlachse (TO_SpeedAxis), Positionierachse (TO_PositioningAxis), Gleichlaufachse (TO_SynchronousAxis), Externer Geber (TO_ExternalEncoder), Messtaster (TO_MeasuringInput), Nocken (TO_OutputCam), Nockenspur (TO_CamTrack). Auf der Webseite "Motion Diagnose" finden Sie Detailinformationen zu den projektierten Technologieobjekten in den Ansichten Diagnose und Service-Übersicht.
Trace	Anzeige von Messungen, die auf der SIMATIC Memory Card gespeichert sind.

Web-Zugriff auf die CPU über PG/PC

Um auf den Webserver zuzugreifen, gehen Sie folgendermaßen vor:

1. Verbinden Sie den Client (PG, PC) über die PROFINET-Schnittstelle mit der CPU.
   Wenn Sie nicht über die PROFINET-Schnittstelle der CPU zugreifen, sondern über die Schnittstelle eines CPs (z. B. CP 1543-1), dann bauen sich die Webseiten langsamer auf als beim Anschluss über die Schnittstelle der CPU. Achten Sie darauf, dass in diesem Fall die Cache-Einstellungen (Temporäre Internetdateien) des Browsers auf "Automatisch" stehen.
2. Öffnen Sie den Webbrowser.
   Tragen Sie im Feld "Adresse" des Webbrowsers die IP-Adresse der CPU ein in der Form http://ww.xx.yy.zz (beispielhafte Eingabe: http://192.168.3.141).
   Die Startseite der CPU wird geöffnet. Von der Startseite aus können Sie zu den weiteren Informationen navigieren.
   Wenn die Startseite nicht angezeigt wird, vergewissern Sie sich, dass der Webserver in der Projektierung aktiviert wurde und die Projektierung geladen wurde.

Web-Zugriff auf eine S7-300/400-CPU über HMI-Geräte und mobile Geräte

Der Webserver unterstützt auch den Terminal-Service von Windows, so dass neben dem Einsatz von PG und PC auch Thin-Client-Lösungen mit mobilen Geräten (z. B. PDA, MOBIC T8) und robusten Vorort-Stationen (z. B. SIMATIC MP370 mit der Option ThinClient/MP) unter Windows CE realisierbar sind.

Um auf den Webserver zuzugreifen, gehen Sie folgendermaßen vor:

1. Verbinden Sie den Client (HMI-Gerät, PDA) über die PROFINET-Schnittstelle mit der CPU.
2. Öffnen Sie den Webbrowser.
   Tragen Sie im Feld "Adresse" des Webbrowsers die IP-Adresse der CPU ein in der Form http://ww.xx.yy.zz /basic (beispielhafte Eingabe: http://192.168.3.141/basic).
   Die Startseite der CPU wird geöffnet. Von der Startseite aus können Sie zu den weiteren Informationen navigieren.

Für HMI-Geräte mit dem Betriebssystem Windows CE, kleiner V 5.x, werden die Informationen der CPU in einem speziell für Windows CE entwickelten Browser verarbeitet. In diesem Browser werden die Informationen in vereinfachter Form dargestellt.

Hinweis

SIMATIC Micro Memory Card bei Einsatz des Webservers (S7-300)

Die Projektierungsdaten für den Webserver werden auf der SIMATIC Micro Memory Card gespeichert. Deshalb wird empfohlen den Einsatz einer SIMATIC Micro Memory Card mit mindestens 512 KB.

Sie können den Webserver auch ohne gesteckte SIMATIC Micro Memory Card nutzen. Bedingungen für den Betrieb ist, dass Sie der CPU eine IP-Adresse zugewiesen haben.

- Der Inhalt des Diagnosepuffers wird in hexadezimalem Code angezeigt.
- Startseite, Identifikations- und Kommunikations-Informationen und Variablenstatus werden als Klartext angezeigt.
- Folgende Anzeigen bleiben leer:
  - Topologie
  - Meldungen
  - Baugruppenzustand
- Die automatische Seitenaktualisierung ist auch ohne SIMATIC Micro Memory Card bzw. ohne Projektierung aktiviert.

Web-Zugriff auf eine S7-1500-CPU über HMI-Geräte und mobile Geräte

Für HMI-Geräte mit dem vorinstallierten Betriebssystem Windows CE (kleiner V 5.x) werden die Informationen der CPU in einem speziell für Windows CE entwickelten Browser verarbeitet. In diesem Browser werden die Informationen in vereinfachter Form dargestellt.

Der Webserver bietet Webseiten mit reduzierter Komplexität an, die auf Geräten mit kleinen Bildschirmen und geringer Rechnerleistung nutzbar sind.

Voraussetzung für den Zugriff mit mobilen Geräten:

- WLAN Anschluss für die CPU (WLAN Access Point, z. B. SCALANCE W788-1RR oder SCALANCE W784-1)
- aktivierter Webserver auf der CPU
- Für das mobile Gerät muss WLAN aktiviert sein und eine Verbindung zum Access Point aufgebaut sein.

Für ein mobiles Gerät ist die Vorgehensweise identisch wie bei anderen Geräten (PG/PC). Im Browser geben Sie die IP-Adresse der CPU-Schnittstelle ein und laden das Zertifikat herunter und installieren es anschließend.

Sicherheit

Beachten Sie, dass Sie die CPU durch verschiedene Techniken vor Kompromittierung schützen müssen, z. B. durch Einschränkung des Netzwerkzugriffs und Verwendung von Firewalls.

Der Webserver bietet folgende Sicherheitsfunktionen:

- Zugriff über das sichere Übertragungsprotokoll HTTPS
- Projektierbare Nutzerberechtigung über Benutzerliste

Siehe auch

Voraussetzungen für den Web-Zugriff (Seite 1322)

Servicedaten auslesen

Der Webserver einer S7-1500-CPU bietet Ihnen die Möglichkeit, Servicedaten abzuspeichern. Neben dem Inhalt des Diagnosepuffers enthalten Servicedaten noch weitere Informationen über den internen Zustand der CPU. Sollte ein nicht lösbares Problem mit der CPU auftreten, haben Sie so die Möglichkeit, dem Service&Support Team die Servicedaten zukommen zu lassen.

Vorgehen

1. Geben Sie in die Adressleiste Ihres Webbrowsers die folgende Adresse ein: "http://<CPU IP address>/save_service_data.html", z. B. "http://192.168.3.141/save_service_data.html"
2. Auf Ihrem Bildschirm erscheint die Ansicht der Servicedaten-Seite mit einer Schaltfläche zum Speichern der Servicedaten.

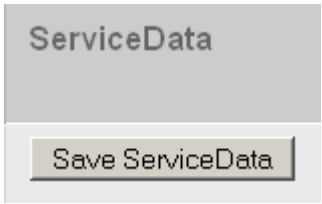

3. Speichern Sie die Servicedaten durch Klicken auf "Save ServiceData" lokal auf Ihrem Anzeigegerät.

Ergebnis

Die Daten werden in eine .dmp-Datei mit folgender Namenskonvention gespeichert: "<MLFB><Seriennummer><Zeitstempel>.dmp". Den Dateinamen können Sie nachträglich ändern.

Standard-Webseiten

Voraussetzungen für den Web-Zugriff

Im Folgenden werden die Voraussetzungen für den Zugriff auf Standard-CPU-Webseiten erläutert sowie die Auswirkungen fehlender bzw. vorhandener Projektierungsinformation dargestellt.

Voraussetzung

Der Webserver muss gestartet sein.

Der Webserver startet erst dann, wenn Sie in den Eigenschaften der CPU, Bereich "Webserver" den Webserver aktiviert haben.

Beachten Sie Folgendes:

- Die Webseiten werden standardmäßig über eine unsichere Verbindung übertragen und sind nicht vor Angriffen Dritter sicher. Wenn Sie die Webseiten verschlüsselt an den Browser übertragen wollen, dann aktivieren Sie die Option "Zugriff nur über HTTPS zulassen" in den CPU-Eigenschaften. Beachten Sie, dass die URL der CPU in diesem Fall mit https:// beginnt.
- Der Zugriff über HTTPS funktioniert nur, wenn in der CPU die Uhrzeit gestellt wurde.
- Für weitergehenden Zugriffsschutz legen Sie in den CPU-Eigenschaften, Bereich "Webserver", Benutzer an.
- Die volle Funktionalität der Webseiten zum Baugruppenzustand, Topologie und Meldungen erfordert bei S7-300/400-CPUs ein Generieren und Laden der Systemdiagnosedaten (Report System Error). Nach Aktivierung des Webservers erscheint daher die Abfrage, ob die Systemdiagnose aktiviert werden soll. Diese Abfrage sollten Sie mit "Ja" quittieren.

Anmelden

Um lesend auf die Standard-Webseiten zugreifen zu können, ist kein Anmelden notwendig. Zum Ausführen bestimmter Aktionen wie das Ändern des Betriebszustandes der CPU oder für Schreibzugriffe muss ein Benutzer angemeldet sein. Die Eingabefelder zum Anmelden sind in der linken oberen Ecke auf jeder Standard-Webseite zu finden.

Wenn Sie sich als Benutzer anmelden, müssen Sie Benutzername und Passwort dort eingeben. Benutzername und Passwort werden in den Eigenschaften der CPU, Bereich "Webserver", festgelegt.

Web-Zugriff ohne Projektierungsinformation

Ohne Projektierung sind nur grundlegende Informationen auf den Standard-Webseiten verfügbar. Diese Informationen stehen in verschiedenen Sprachen zur Verfügung:

- Diagnosepuffer-Einträge werden ohne Projektierungsinformationen nicht als Klartext angezeigt, sondern nur als Hexadezimalcode.
- Falls der Webserver der CPU Variablentabellen (Beobachtungstabellen) unterstützt: Die Variablentabellen-Seite zeigt ohne Projektierung keine Variablen an. Einen Variablenstatus können Sie immer abfragen.

Web-Zugriff mit Projektierungsinformation

In der Projektierungsinformation sind Texte in den ausgewählten Landessprachen enthalten:

- Texte für Diagnosepuffer-Einträge als Kurz-/Langtext
- Bei S7-300/400-CPUs: Meldetexte; z. B. der Meldebausteine AlarmS/SQ, AlarmD/DQ
- Falls Variablentabellen (Beobachtungstabellen) unterstützt werden: Variablentabellen mit den darin enthaltenen Symbolen und Kommentaren

Die Texte für Diagnosepuffer-Einträge und Meldungen werden im Betrieb in der Landesprache angezeigt, die Sie im Webbrowser eingestellt haben.

Siehe auch

Einstellungen für den Betrieb (Seite 1324)

Einstellungen für den Betrieb

Einstellungen für den Betrieb

Um den Webserver einer CPU nutzen zu können und um das Verhalten des Webservers zu bestimmen, müssen Sie in der Netzsicht, Gerätesicht oder Topologiesicht die CPU markieren.

Nehmen Sie dann folgende Einstellungen im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" vor:

- Webserver aktivieren
- Webzugriff beschränken auf sicheres Hypertext-Übertragungsprotokoll (HTTPS)
- Automatische Aktualisierung aktivieren
- Mehrsprachigkeit (Projektsprachen für Webser und CPU-Display einstellen)
- Benutzerliste ergänzen
- Für S7-300/400-CPUs: Anzeigeklassen der Meldungen auswählen

Jede PROFINET-Schnittstelle einer S7-1500-CPUs bzw. eines PROFINET-CPs oder eines PROFINET-CMs kann ebenfalls für den Zugriff "von außen" auf den Webserver freigegeben oder gesperrt werden. Am einfachsten verwalten Sie diesen Zugriff zentral in den Eigenschaften der CPU. Klicken Sie dazu im Register "Eigenschaften" des Inspektorfensters auf den Bereich "Webserver > Übersicht der Schnittstellen" und aktivieren Sie die Schnittstellen, über die ein Zugriff erlaubt ist.

Siehe auch

Voraussetzungen für den Web-Zugriff (Seite 1322)

Webserver aktivieren

Webserver aktivieren

In der Grundeinstellung ist der Webserver deaktiviert. Damit die CPU Webseiten anzeigt, müssen Sie das entsprechende Optionskästchen aktivieren.

Diese Einstellung finden Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver"; Optionskästchen "Webserver auf dieser Baugruppe aktivieren".

Wenn der Webserver aktiviert wurde, erscheint für S7-300/400-CPUs eine Abfrage, ob die Systemdiagnose (Report System Errors) aktiviert werden soll. Klicken Sie auf "Ja", wenn die notwendigen Bausteine und Texte zu "Report System Errors" mitgeneriert werden sollen, so dass die Meldetexte im Betrieb im Webbrowser zur Verfügung stehen. Bei S7-1200/1500-CPUs ist die Systemdiagnose immer aktiviert (nicht abschaltbar).

Bei S7-1500-CPUs mit mehreren PROFINET-Schnittstellen bzw. wenn CPs mit PROFINET-Schnittstellen konfiguriert werden, müssen Sie zusätzlich die Schnittstellen aktivieren, über die ein Webserver-Zugriff erlaubt sein soll. Diese Einstellung finden Sie unter "Eigenschaften > Allgemein > Webserver > Übersicht der Schnittstellen".

Automatische Aktualisierung aktivieren

Automatische Aktualisierung aktivieren

Bei aktiviertem Optionskästchen "Automatische Aktualisierung" werden die Webseiten automatisch im unter "Aktualisierungsintervall" festgelegten Zeitabstand aktualisiert. Von der Aktualisierung ausgenommen ist die Webseite "Identifikation".

Um die automatische Aktualisierung zu aktivieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie unter "Eigenschaften > Allgemein > Webserver > Automatische Aktualisierung" das Optionskästchen "Automatische Aktualisierung aktivieren".
2. Geben Sie das Aktualisierungsintervall an.

Hinweis

Aktualisierungszeit

Das eingestellte Aktivierungsintervall ist die kürzeste Aktualisierungszeit.
Größere Datenmengen oder mehrere HTTP-/HTTPS-Verbindungen erhöhen die Aktualisierungszeit.

Landessprache für Web einstellen

Landessprache für Web einstellen

Um die Landessprachen für sprachabhängige Texte auszuwählen, die in die CPU geladen werden sollen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Inspektorfensters zum Bereich "Mehrsprachigkeit".
2. Aktivieren Sie die benötigten Sprachen für die Anzeige im Webbrowser in der ersten Spalte; die Anzahl auswählbarer Sprachen ist CPU-spezifisch.
3. Weisen Sie den Sprachen des Displays/des Webservers jeweils die gewünschte Projektsprache zu.
   Die zuzuweisenden Projektsprachen müssen aktiviert sein und die entsprechenden Texte (Übersetzungen) im Projekt vorliegen. Die Projektsprachen-Auswahl finden Sie in der Projektnavigation unter "Sprachen & Ressourcen".
   Durch diese Zuweisungen werden z. B. Anzeigetexte, die nicht in der CPU vorhanden sind, beim Laden in die CPU in der gewählten Sprache (mit Sprach- und Ländercode) geladen.

Projektsprachen	Sprachen des CPU-Displays/des Webservers
Deutsch (Deutschland)	Deutsch
Deutsch (Deutschland)	Englisch
Deutsch (Deutschland)	Französisch
Deutsch (Deutschland)	Spanisch
Deutsch (Deutschland)	Italienisch
Deutsch (Deutschland)	Japanisch
Deutsch (Deutschland)	Chinesisch (vereinfacht)
Deutsch (Deutschland)	Koreanisch
Deutsch (Deutschland)	Russisch
Deutsch (Deutschland)	Türkisch
Deutsch (Deutschland)	Portugiesisch (Brasilien)

Hinweis

Wenn Sie den Webserver aktivieren und keine Sprache zuweisen, werden Meldungen und Diagnoseinformationen in hexadezimalem Code angezeigt.

Benutzerverwaltung

Benutzerverwaltung

Die Benutzerverwaltung bietet Ihnen folgende Möglichkeiten:

- Benutzer anlegen
- Ausführungsrechte festlegen
- Passwörter vergeben

Mit dieser Zuordnung stehen Benutzern ausschließlich die Optionen zur Verfügung, die den Ausführungsrechten fest zugeordnet sind.

- Wenn ein Benutzer projektiert ist, kann dieser nach Anmeldung mit seinem Passwort entsprechend seiner Zugriffsrechte auf die Webseiten zugreifen.
- Je nach verwendeter CPU und Firmware können Sie unterschiedliche Benutzerrechte vergeben. Rechte, die Ihre CPU nicht unterstützt, können nicht angewählt werden.
- In der Benutzerliste ist voreingestellt ein Benutzer mit Namen "Jeder" angelegt, welcher minimale Zugriffsrechte besitzt: Lesender Zugriff auf Intro- und Startseite. Der Benutzer "Jeder" ist ohne Vergabe eines Passworts festgelegt.
- Als nicht angemeldeter Benutzer greifen Sie voreingestellt immer als Benutzer "Jeder" auf den Webserver zu. Dabei ist nicht relevant, ob Sie zusätzliche Benutzer projektiert haben. Sie können dem Benutzer "Jeder" allerdings alle in STEP 7 verfügbaren Zugriffsberechtigungen zuordnen. Da der Benutzer "Jeder" in STEP 7 ohne Zuweisung eines Passwortes festgelegt ist, achten Sie darauf, welche Zugriffsberechtigungen Sie diesem Benutzer zuweisen. Einzelne Berechtigungen, wie etwa die Möglichkeit zum Ändern des Betriebszustands, können ein Sicherheitsrisiko darstellen.
  Beim Zuweisen sicherheitsrelevanter Berechtigungen empfehlen wir: Projektieren Sie einen Benutzer und vergeben Sie immer ein Passwort!

Empfehlungen zur Passwortvergabe

- Vergeben Sie bei der Projektierung der Benutzer sichere Passwörter.
  Ein sicheres Passwort wird beispielsweise nur für eine einzige Anwendung verwendet, ist mehr als 8 Zeichen lang und besteht aus Groß- und Kleinbuchstaben sowie Sonderzeichen und Ziffern (?!+%$1234...).
- Verwenden Sie keine gängigen Zeichenfolgen der Computertastatur oder Wörter aus dem Wörterbuch.
- Ändern Sie das Passwort in regelmäßigen Abständen.Aktivieren Sie wenn möglich immer die Option "Zugriff nur über HTTPS zulassen" sobald Sie mindestens einem Benutzer ein Passwort zugewiesen haben.

Server Security

Um das Übertragungsprokoll HTTPS nutzen zu können, ist bei SIMATIC S7-1500 CPUs ab Firmware V2.0 ein Webserver-Zertifikat zu erstellen und der CPU zuzuweisen.

Das Webserver-Zertifikat ist auch Voraussetzung für die Benutzerverwaltung des Webservers (Passwort-Schutz).

Voraussetzung

Sie haben die Eigenschaften der CPU (Inspektorfenster) geöffnet.

Vorgehen

Die Einstellungen für die Erzeugung eines Webserver-Zertifikats befinden sich in den CPU-Eigenschaften im Bereich "Webserver > Security".

In diesem Bereich können Sie Zertifikate erstellen bzw. auswählen. Dabei wird unterschieden, ob Sie den Zertifikatsmanagers in den globalen Security-Einstellungen verwenden oder nicht.

Zur Vorgehensweise finden Sie weiterführende Informationen zum Zertifikatsmanager unter "Siehe auch".

Erstellung von Webserver-Zertifikaten abhängig von CPU-Firmware-Version

Bei SIMATIC S7-1500 CPUs ab Firmware V2.0 müssen Sie das Zertifikat für den Webserver der CPU mit STEP 7 selbst erstellen und in den Eigenschaften der CPU dem Webserver zuweisen. Mit dem Laden der Hardware-Konfiguration in die CPU wird dieses Zertifikat automatisch mitgeladen.

Das STEP 7-Projekt selbst hat ein Stammzertifikat (CA, Certification Authority), mit dem das Webserver-Zertifikat signiert werden kann. Hierfür ist die Anmeldung für den Zertifikatsmanager in den globalen Security-Einstellungen erforderlich.

Ausnahme:

Wenn Sie die Firmware einer SIMATIC S7-1500 CPU bzw. ET 200SP mit Firmwarestand < V2.0 auf einen Firmwarestand ≥ V2.0 aktualisieren, wird automatisch von der CPU ein gültiges Server-Zertifikat generiert und verwendet. Dasselbe gilt für den Ersatzteilfall, in dem eine neuere CPU eine CPU mit Firmwarestand < V2.0 ersetzt.

In diesen Fällen wird nicht das Webserver-Zertifikat von STEP 7 genutzt.

Zugriff nur über HTTPS

Zugriff nur über HTTPS

HTTPS dient zur Verschlüsselung und Authentifizierung der Kommunikation zwischen Browser und Webserver.

Die Webseiten werden standardmäßig über eine unsichere Verbindung übertragen und sind nicht vor Angriffen Dritter geschützt. Wenn Sie Webseiten und Anmeldeinformationen ausschließlich gesichert an den Browser übertragen wollen, dann aktivieren Sie in den CPU-Eigenschaften die Option "Zugriff nur über HTTPS zulassen". Beachten Sie, dass die URL der CPU in diesem Fall mit "https://" beginnt.

Beachten Sie: Voraussetzung für die Nutzung des sicheren Übertragungsprotokolls "HTTPS" ist ein gültiges Webserver-Zertifikat in der CPU.

Voraussetzungen für Zugriff über HTTPS

Für einen fehlerfreien HTTPS-Zugriff von Ihrem Webbrowser auf den Webserver der CPU ist Folgendes erforderlich:

- In der CPU muss die aktuelle Uhrzeit eingestellt sein.
  Allgemein gilt: Wenn Sie Secure Communication nutzen (z. B. HTTPS, Secure OUC, OPC UA), dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.
- Die IP-Adresse der CPU muss vergeben sein.
- Im Webbrowser sollte ein von der CPU angebotenes, gültiges Stammzertifizierungsstellen-Zertifikat installiert sein.
  Wenn kein Zertifikat installiert ist, wird eine Warnung angezeigt, mit der Empfehlung, die Seite nicht zu benutzen. Um die Seite zu sehen, müssen Sie explizit eine "Ausnahme hinzufügen".
  Ein gültiges Stammzertifizierungsstellen-Zertifikat (Certification Authority) erhalten Sie als Download auf der Webseite "Intro" des Webservers unter "Zertifikat herunterladen".
  Wie Sie das Zertifikat in Ihren Webbrowser installieren, finden Sie in der Hilfe Ihres jeweiligen Webbrowsers und im FAQ (https://support.industry.siemens.com/cs/ww/de/view/103528224) mit der Beitrags-ID 103528224 auf der Internetseite des Service&Support.

Hinweis

Um sich vor Manipulation von außen zu schützen, führen Sie den Download des Zertifikats ausschließlich in einer garantiert unkompromittierbaren Umgebung durch.

Für jedes Anzeigegerät, das Sie verwenden wollen, muss die Installation des Stammzertifizierungsstelllen-Zertifikats einmal durchgeführt werden.

Beispiel für Zugriff auf Webseiten mit HTTPS-Protokoll

Um mit HTTPS-Protokoll auf die Webseiten der CPU zuzugreifen, geben Sie die URL in folgender Form ein: https://ww.xx.yy.zz, wobei ww.xx.yy.zz die IP-Adresse der CPU ist.

Beispielhafte Eingabe: https://192.168.3.141

Eine verschlüsselte Verbindung erkennen Sie am Schloss-Symbol in der Statusleiste der Webseite.

Zugriffsschutz

Die mithilfe des Zertifikats erstellte verschlüsselte Verbindung verhindert zwar das Abhören oder Verfälschen der Kommunikation, stellt aber keinen Zugriffsschutz dar. Schützen Sie deshalb Ihre CPU durch entsprechende Projektierung in der Benutzerverwaltung vor unberechtigtem Zugriff.

Anzeigeklassen der Meldungen

Anzeigeklassen der Meldungen

In der Grundeinstellung sind alle Anzeigeklassen der Meldungen aktiviert. Die Meldungen zu den ausgewählten Anzeigeklassen werden im Betrieb auf der Webseite "Meldungen" angezeigt. Die Meldungen zu den nicht ausgewählten Anzeigeklassen erhalten Sie nicht als Klartext, sondern als hexadezimalen Code.

So projektieren Sie die Anzeigeklassen:

- für "Report System Errors" im Inspektorfenster unter "Eigenschaften > Allgemein > Systemdiagnose"
- für bausteinbezogene Meldungen im Meldungseditor

Hinweis

Speicherbedarf verringern

Sie können den Speicherbedarf für die Webserver-Konfiguration verringern, indem Sie nur die Anzeigeklassen der Meldungen auswählen, die genutzt werden sollen.

Anwenderseiten erstellen und laden

Wissenswertes zu Anwenderseiten

Konzept

Das Konzept der Anwenderseiten bzw. Anwenderdefinierten Webseiten ermöglicht Ihnen, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen. Der Webserver der CPU stellt diese Funktion zur Verfügung.

Hinsichtlich der Gestaltung und Funktionalität der anwenderdefinierten Webseiten sind Sie nicht auf spezielle Tools angewiesen. Sie können die Seiten mit CSS im Layout anpassen, mit Javascript dynamischen Inhalt zur Verfügung stellen oder ein beliebiges Framework zur Produktion der Webseiten verwenden. Die Gesamtheit der Dateien, die der Webserver verarbeitet, wird auch "Webapplikation" genannt.

Webapplikation und Anwenderprogramm

Mit Hilfe von speziellem HTML-Code in anwenderdefinierten Webseiten (AWP-Kommandos) können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an das Anwenderprogramm der CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen.

Mit Scriptanweisungen (z. B. Javascript) können Sie Ihre Webseiten optimieren, z. B Inhalte dynamisch ändern oder Benutzereingaben validieren.

Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung WWW (SFC 99) im Anwenderprogramm aufrufen.

- Wenn keine Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, z. B. wenn eine Webseite nur statische Informationen anbietet, ist nur eine Initialisierung im Anwenderprogramm notwendig.
- Wenn ein einfacher Datenaustausch zwischen PLC-Variablen und Variablen der Webapplikation notwendig ist, z. B. um den Inhalt von PLC-Variablen anzuzeigen oder in eine PLC-Variable einen Wert zu schreiben, müssen Sie die Syntax zum Lesen und Schreiben von Variablen zu berücksichtigen. Im Anwenderprogramm ist in diesem Fall ebenfalls nur eine Initialisierung z. B. im Anlauf-OB erforderlich.
- Wenn eine weitergehende Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, müssen Sie neben der Synchronisierung zwischen Webserver und Anwenderprogramm auch Status- und Steuerinformation aus dem Web-Control-DB hantieren. Das ist z. B. der Fall bei Benutzereingaben, die über den Webbrowser an den Webserver übergeben werden, um von der CPU ausgewertet zu werden. Im Unterschied zum einfachen Datenaustausch beeinflusst das Anwenderprogramm unmittelbar den Zeitpunkt, zu dem die angeforderte Webseite an den Webbrowser zurückgegeben wird. In diesem Fall ist es notwendig, dass Sie sich mit dem Konzept der manuellen Fragmente und den Strukturen des Web-Control-DBs auseinandersetzen.

Einbindung der HTML-Quellen über das TIA Portal

Die Parameter für eine Einbindung der HTML-Quellen im TIA Portal finden Sie in den Eigenschaften der jeweiligen CPU mit Webserver:

Parameter	Bedeutung
HTML-Verzeichnis	Verzeichnis der HTML-Quellen für die Webapplikation. Sie können den Pfad direkt eingeben oder über die Schaltfläche neben dem Eingabefeld zu dem Verzeichnis navigieren. Es ist vorteilhaft, wenn Sie Sie Ihre Webseiten mit relativen Pfaden einbinden. Dazu müssen Sie das Verzeichnis, in dem Sie die HTML-Quellen für die Webapplikation speichern, im STEP 7-Projektverzeichnis eingefügen, z. B. in ein neues Verzeichnis "Webpages". Als HTML-Verzeichnis geben Sie dann den relativen Pfadnamen an; im Beispiel ".\Webpages". Vorteile der Verwendung relativer Pfade: • Die anwenderdefinierten Webseiten werden gemeinsam mit dem STEP 7-Projekt archiviert. • Die Webseiten werden über den Menübefehl "Projekt > Speichern unter..." mit in den neuen Pfad kopiert. • Beim Kopieren des Projekts über den Windows Explorer in einen anderen Pfad entfällt die Anpassung des HTML-Verzeichnis-Pfades zur Generierung neuer Web-DBs. • Mehrere Steuerungen in einem Projekt, die die gleichen HTML-Seiten verwenden, haben auch die gleiche HTML-Verzeichnis-Pfadangabe.
Start-HTML-Seite	Pfad zur Start-HTML-Seite. Sie können den Pfad direkt eingeben oder über die Schaltfläche neben dem Eingabefeld zu der HTML-Seite zu navigieren. Die Start-HTML-Seite ist die HTML-Seite, die beim Start der Webapplikation aufgerufen werden soll.
Applikationsname	Optionaler Name für die Applikation. Dieser Name dient zur weiteren Unterteilung bzw. Gruppierung der Webseiten. Wenn ein Applikationsname vorhanden ist, wird die URL in folgendem Format angezeigt: "http://a.b.c.d/awp/<Applikationsname>/<Seitenname>.html"

Initialisierung

Anwenderdefinierte Webseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren, um die Webapplikation in die CPU laden zu können. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Datenbausteine, die codierte Webseiten enthalten, werden "Fragment-DBs" genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin anwenderdefinierte Webseiten codiert sind. Durch die Anweisung "WWW" (SFC 99) z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die anwenderdefinierten Webseiten über einen Webbrowser erreichbar.

Synchronisierung

Wenn das Anwenderprogramm mit den anwenderdefinierten Webseiten Daten austauschen bzw. interagieren soll, muss die Anweisung WWW (SFC 99) im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragendenden Webbrowser zusammenstellen und zurücksenden

In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer angeforderten Webseite dem Webserver mitgeteilt werden.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Webserver S7-1500 (http://support.automation.siemens.com/WW/view/de/59193560)

Überblick Vorgehensweise

Grundlegendes

In diesem Kapitel ist die grundsätzliche Vorgehensweise schrittweise erläutert, um anwenderdefinierte Webseiten zu erstellen, zu laden und in der Betriebsphase zu nutzen.

Die folgende Grafik stellt den Ablauf zur Erstellung und Anzeige von anwenderdefinierten Webseiten vereinfacht dar:

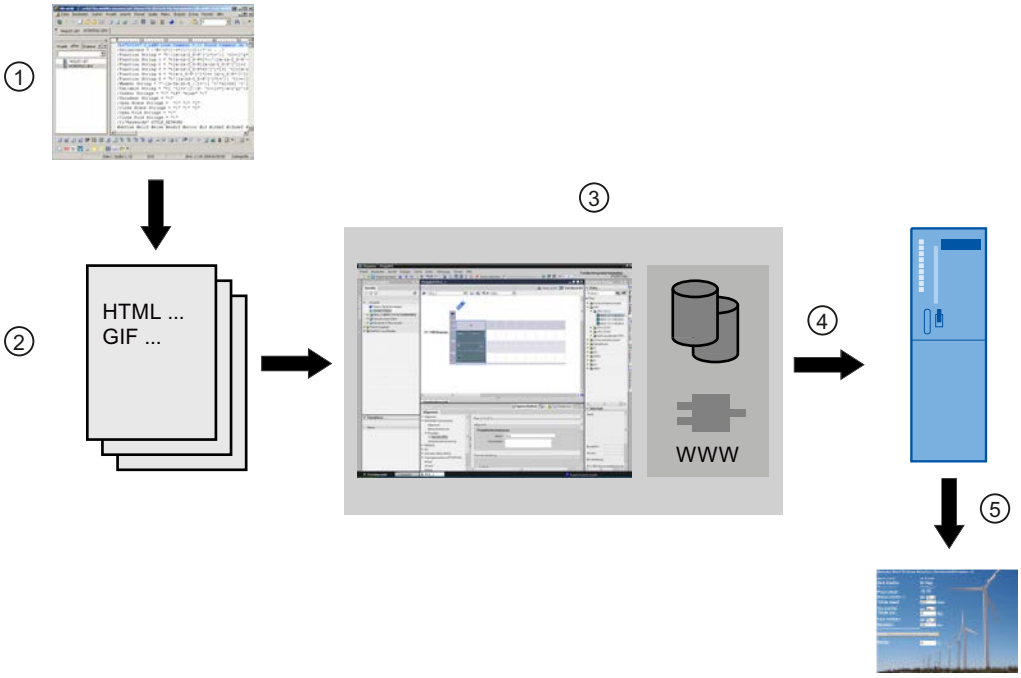

① Programmieren einer Webapplikation (evtl. unter Zuhilfenahme von geeigneten Tools, gegebenenfalls mit enthaltenen AWP-Kommandos für dynamische Seiten).

② Webapplikation besteht aus einzelnen Quelldateien, z. B. *.html, *.gif, *.js, ...

③ Mit STEP 7:

- Generieren der Datenbausteine (Web-Control-DB und Fragment-DBs) aus Quelldateien. Die DBs enthalten Metainformation und die komplette Webapplikation inklusive der Bilder und den dynamischen und statischen Teilen der Webapplikation. Die DBs werden unter "Systembausteine" in der Projektnavigation abgelegt.
- Die Anweisung "WWW" (SFC 99) im Anwenderprogramm aufrufen. Die Anweisung initialisiert den Webserver der CPU für eine Webapplikation.
- Bei Bedarf Interaktion zwischen Webserver und Anwenderprogramm ausprogrammieren

④ Laden der Bausteine in die CPU.

⑤ Webseite im Browser aufrufen. Der Aufruf der Webseiten der CPU erfolgt über die Eingabe der IP-Adresse der CPU.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Webseiten erstellen

Zur Erstellung der anwenderdefinierten Webseiten bieten sich Webdesign-Tools diverser Anbieter an. Grundsätzlich sollen die Webseiten nach den Konventionen des W3C (World Wide Web Consortium) programmiert und gestaltet werden. Im Webserver der CPU findet keine Überprüfung nach W3C Kriterien statt.

Regeln

- Das Tool muss in der Lage sein, den HTML-Code direkt zu editieren, damit Sie AWP-Kommandos in der HTML-Seite einfügen können.
  In der CPU werden lediglich die AWP-Kommandos geparst und z. B. durch Werte aus dem Anwenderprogramm/Prozessabbild der CPU ersetzt.
- Dateien, die AWP-Kommandos enthalten, müssen UTF-8 codiert sein. In den Metadaten der HTML-Seite setzen Sie daher das Attribut charset auf UTF-8 und speichern die Datei UTF-8-codiert.
- Dateien, die AWP-Kommandos enthalten, dürfen folgende Sequenz nicht enthalten:]]
- Dateien, die AWP-Kommandos enthalten, dürfen außerhalb von "Variable-lesen-Bereichen" (:=<Variablenname>:) folgende Sequenz nicht enthalten: :=
  Tipp: Ersetzen Sie das erste Zeichen einer verbotenen Sequenz durch seine Zeichencodierung; für den Doppelpunkt z. B. :.

Ein kleines Beispiel für eine anwenderdefinierte Webseite soll den grundsätzlichen Aufbau zeigen.

Voraussetzung

- Die CPU muss über einen Webserver verfügen und der Webserver der CPU muss aktiviert sein
- Um als Nutzer von anwenderdefinierten Webseiten schreibend auf PLC-Variablen zugreifen zu können, müssen Sie als "admin" angemeldet sein.
- Für das Beispiel unten müssen für PLC-Variablen, die auf der Webseite angezeigt werden sollen, PLC-Variablen definiert sein. Für die erste verwendete PLC-Variable "Tank_below_max" ist das hier dargestellt.

		Name	Datentyp	Adresse
1		Tank_below_max	Bool	%I0.0

Anwenderdefinierte Webseiten erstellen

Der folgende Code für eine Beispielwebseite liest Werte aus dem Prozessabbild und stellt diese in einer Tabelle dar.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/
TR/html4/strict.dtd">
<html>
  <head>
   <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
   <title>Mix</title>
  </head>
   <body>
    <h1>Mix</h1>
    <h2> Actual State </h2>
    <table border="1">
     <tr>
      <th>Variable</th>
      <th>State</th>
```

```
      </tr>
      <tr>
       <td>Tank below max</td>
       <td>:="Tank_below_max":</td>
      </tr>
      <tr>
       <td>Tank above min</td>
       <td>:="Tank_above_min":</td>
      </tr>
     </table>
    </body>
</html>
```

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Anwenderseite als Startseite und Empfehlungen zur Größe

Anwenderseite als Startseite definieren

Anwenderseite als Startseite definieren

Neben der voreingestellten Introseite können Sie auch die Startseite Ihrer Anwenderseiten als Startseite des Webservers festlegen.

Voraussetzungen

- Sie haben in STEP 7 einen Benutzer projektiert, dem Sie mindestens das Recht "... Anwenderseiten aufrufen" zugewiesen haben.
- Die CPU befindet sich im Betriebszustand RUN.

Vorgehensweise

Um die Anwenderseiten in STEP 7 als Startseite des Webservers zu definieren, gehen Sie wie folgt vor:

1. Wählen Sie die CPU in der Gerätekonfiguration aus.
2. Rufen Sie die Einstellungen im Inspektorfenster der CPU unter "Eigenschaften > Allgemein > Webserver" auf.
3. Wählen Sie im Bereich "Einstiegsseite" unter "Einstiegsseite auswählen" den Eintrag "AWP1".

Wenn Sie jetzt im Browser die IP-Adresse der CPU eingeben, wird automatisch eine Verbindung zu Ihren Anwenderseiten hergestellt.

Wenn Sie die Webseiten der CPU wieder erreichen wollen, dann verlinken Sie die Webseiten von Ihren Anwenderseiten aus z. B. über die URL "http://a.b.c.d./Portal/Portal.mwsl?

PriNav=Start" oder "https://a.b.c.d/Portal/Portal.mwsl?PriNav=Start". Die Angabe "a.b.c.d" steht dabei exemplarisch für die IP-Adresse der projektierten CPU.

Beispiel der Verlinkung in HTML:
```
<a href="/Portal/Portal.mwsl?PriNav=Start">SIMATIC-Webseiten</a>
```

Hinweis

Wenn Sie Ihre Anwenderseite als Startseite des Webservers festlegen, werden alle Direktzugriffe auf die Webseiten der CPU blockiert. Dies gilt auch für von Ihnen gespeicherte Lesezeichen für die Webseiten der CPU sowie die Seite zum Auslesen der Servicedaten.

Servicedaten auslesen

Wenn Sie Ihre Anwenderseite als Startseite des Webservers festlegen, wird der Direktzugriff auf die Seite zum Auslesen der Servicedaten ebenfalls blockiert.
Wollen Sie im Servicefall weiterhin Servicedaten über den Webserver auslesen, so verlinken Sie die Servicedaten-Seite direkt aus Ihrer Anwenderseite.
Wie bei den Webseiten der CPU verlinken Sie die Servicedaten-Seite z. B. über die URL "http://a.b.c.d/save_service_data" oder "https://a.b.c.d/save_service_data", die Angabe "a.b.c.d" steht dabei exemplarisch für die IP-Adresse der projektierten CPU.

Beispiel der Verlinkung in HTML:
```
<a href="/save_service_data">Servicedaten</a>
```

Verweis

Weitere Informationen zum Thema Anwenderseite als Startseite finden Sie im FAQ mit der Beitrags-ID 67184104 auf der Internetseite des Online Supports (https://support.industry.siemens.com/cs/ww/de/view/67184104).

Empfehlungen zur Größe von Webseiten

Einfluss der Webseitengröße auf Speicherplatzverbrauch und Performance

Umfangreiche HTML-Seiten, insbesondere solche, die viele Bilder enthalten, benötigen viel Speicherplatz im Ladespeicher. Beachten Sie folgende Tipps:

- Um ausreichend Ladespeicher zur Verfügung zu stellen, wählen Sie eine SIMATIC Memory Card mit genügend Speicherplatz.
- Begrenzen Sie das Datenvolumen der HTML-Seiten auf 1 Mbyte.
  Wenn das Datenvolumen der HTML-Seiten größer ist als 1 Mbyte, können Performance-Einbußen auftreten, da im Cache nur 1 Mbyte Daten zwischengespeichert werden.
- Begrenzen Sie die Größe einer Datei einer HTML-Seite auf 512 kbyte.
  Wenn eine einzelne Datei einer HTML-Seite größer ist als 512 kbyte, können Probleme beim Versenden der Datei vom Webserver an den Browser auftreten.

Lassen Sie sich die Dateigrößen im Datei-Explorer des Verzeichnisses anzeigen, in dem die Webapplikation liegt. Reduzieren Sie wenn erforderlich und möglich die Dateigrößen.

AWP-Kommandos für die Steuerung von Webseiten

AWP-Kommandos in Webseiten verwenden

Mit AWP-Kommandos (Automation Web Programming) deklarieren Sie die Schnittstelle zwischen Ihren Anwenderseiten (Webapplikation, z. B. eine einfache HTML-Seite) und den CPU-Daten.

Für die Entwicklung von Anwenderseiten bzw. Webapplikationen sind Sie nur den Beschränkungen unterworfen, die Webbrowser vorgeben. Mit dem Anwenderprogramm in der CPU kontrollieren Sie in einer der Programmiersprachen von STEP 7, welche CPU-Daten zu welcher Zeit im Webbrowser des Betrachters angezeigt werden.

Mit Hilfe von AWP-Kommandos, die Sie innerhalb der HTML-Seite notieren, werden Daten für die beabsichtigte Interaktion zwischen Webapplikation und Anwenderprogramm vereinbart.

AWP-Kommandos fügen Sie als HTML-Kommentare mit einer speziellen Syntax in HTML-Seiten ein. Mithilfe von AWP-Kommandos lassen sich folgende Features umsetzen:

- PLC-Variablen lesen
- PLC-Variablen schreiben
- Spezialvariablen lesen
- Spezialvariablen schreiben
- Enum-Typen definieren
- Variablen Enum-Typen zuweisen
- Fragmente definieren
- Fragmente importieren

Syntax von AWP-Kommandos

Ein AWP-Kommando beginnt mit "`<!-- AWP_`" und endet mit "` -->`". In Java-Script-Dateien sollten die Kommandos zusätzlich von Javascript-Kommentaren eingeschlossen werden ("`/*...*/`").

Notationsregeln für PLC-Variablennamen innerhalb eines AWP-Kommandos

Die AWP-Kommandos "AWP_In_Variable" und "AWP_Out_Variable" enthalten ein Name-Attribut und optional ein Use-Attribut. Diesen Attributen ist ein PLC-Variablenname zugeordnet, über den PLC-Variablen im Browser geschrieben bzw. gelesen werden. Für den Umgang mit PLC-Variablennamen im HTML-Code gibt es folgende Regeln:

- PLC-Variablen müssen in Anführungszeichen (" ... ") eingeschlossen werden.
- In AWP-Kommandos verwendete PLC-Variablen müssen zusätzlich von einfachen Anführungszeichen ("' ... '") oder mit umgekehrtem Schrägstrich (Backslash) maskierten Anführungszeichen umschlossen werden ("\" ... \"").
- Wenn der PLC-Variablenname das Zeichen \ (umgekehrter Schrägstrich) oder * (Sternchen) enthält, dann muss dieses Zeichen mit der Escape-Sequenz \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden. Beispiele siehe unten.

- Wenn der PLC-Variablenname im AWP-Kommando zusätzlich mit einfachen Anführungszeichen eingeschlossen ist und innerhalb des Namens kommt das einfache Anführungszeichen (') vor, muss es auch durch die Escape-Sequenz \' als normales Zeichen gekennzeichnet werden.
- Wenn im AWP-Kommando eine absolute Adresse (Eingang, Ausgang, Merker) verwendet wird, dann wird sie von einfachen Anführungszeichen umschlossen.

PLC-Variable	PLC-Variable im HTML-Code
"Velocity"	<!-- AWP_In_Variable Name='"Velocity"' -->
	<!-- AWP_In_Variable Name="\"Velocity\"" -->
"abc\de"	<!-- AWP_In_Variable Name='"abc\\de"' -->
"abc'de"	<!-- AWP_In_Variable Name='"abc\'de"' -->
"abc'de"	<!-- AWP_In_Variable Name="abcde" Use'"abc\'de"' -->
"DB name".tag	<!-- AWP_In_Variable Name='"DB name".tag' -->
"Plc1".Data[1].typeDataStruct.value	<!-- AWP_In_Variable Name='"Plc1".Data[1].typeDataStruct.value' -->
-	<!-- AWP_Out_Variable Name=’flag1’ Use='M0.0' -->

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Variablen lesen

Anwenderdefinierte Webseiten können PLC-Variablen lesen.

Die PLC-Variable muss durch einen PLC-Variablennamen spezifiziert sein.

Diese Out-Variablen (Ausgaberichtung von der Steuerung aus betrachtet) werden mit der im folgenden beschriebenen Syntax an beliebiger Stelle im HTML-Text eingefügt.

Syntax

```
:=<varname>:
```

Diese Referenzen werden im Betrieb vom Webserver durch die jeweils aktuellen Werte der PLC-Variablen ersetzt.

`<varname>` kann eine einfache globale PLC-Variable sein, aber auch ein kompletter Variablenpfad zu einem Strukturelement.

Notationsregeln für PLC-Variablennamen

- PLC-Variablen im HTML-Code werden mit Anführungszeichen (") umschlossen, wenn sie in der Variablentabelle definiert sind. Bei Datenbaustein-Variablen wird der Name des Datenbausteins mit Anführungszeichen umschlossen. Wenn spezielle Zeichen in den Strukturelementen des Datenbausteins verwendet werden wie z. B. der Punkt (.) oder Leerzeichen, muss auch dieser Teil in Anführungszeichen eingeschlossen sein.
- Anführungszeichen werden nicht verwendet bei absoluten Adressen von Eingängen, Ausgängen oder Merkern.

PLC Variable	PLC-Variable im HTML-Code
"DB_name".var_name	:="DB_name".var_name:
"DB_name"."var.name"	:="DB_name"."var.name":
"memory"	:="memory":
-	:=E0.0:
	:=A0.0:
	:=MW100:
	:=%MW100:
"My_Data_Block".flag1	<!-- AWP_Out_Variable Name='flag1' Use='"My_Data_Block".flag1' -->
	...
	:=flag1:

- Wenn der PLC-Variablenname die Zeichen : (Doppelpunkt) oder \ (umgekehrter Schrägstrich) enthält, dann müssen diese Zeichen mit der Escape-Sequenz \: bzw. \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden.

PLC Variable	PLC-Variable im HTML-Code
"abc:de"	:="abc\:de":
"abc\de"	:="abc\\de":

- Spezielle Zeichen "<, &, >"
  Falls diese Zeichen im Variablennamen vorkommen (z. B. "a<b"), kann es zu Darstellungsproblemen kommen.
  Vermeiden Sie Ausdrücke wie z. B. :="a<b": in der HTML-Seite.
  Um Darstellungsproblemen vorzubeugen, verwenden Sie z. B. ein AWP-Kommando mit einem Use-Ausdruck nach dem unten stehenden Muster. Das Use-Attribut definiert die PLC-Variable mit den problematischen Zeichen, das Name-Attribut definiert den Namen ohne problematische Zeichen, wie er in der HTML-Seite verwendet wird.

PLC Variable	PLC-Variable im HTML-Code
"a<b"	<!-- AWP_Out_Variable Name='simplename' Use='"a<b"' -->
	...
	:=simplename:

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Besonderheiten bei PLC-Variablen vom Typ String und Character

Lesen von Variablen vom Typ String und Character

Im Folgenden werden diese Arten von Hochkommata zur Erklärung verwendet: einfaches Anführungszeichen ('), doppeltes Anführungszeichen (").

Ab Firmware V1.6 gibt die CPU bei der Funktion "PLC-Variablen lesen" Variablen vom Typ String oder Character eingeschlossen in einfache Anführungszeichen an den Browser aus. Zum Beispiel:

- String-Variable "Varname".MyString = ABC
- Sie lesen die Variable in HTML über die Funktion :="Varname".MyString:
- Der Webserver gibt die Zeichenfolge 'ABC' an den Browser aus

Nutzung von String- oder Character-Variablen in Ausdrücken

Sie verwenden auf Ihrer HTML-Seite einen Ausdruck, in dem die Zeichenfolge zum Lesen einer Variable in Anführungszeichen eingeschlossen ist, z. B. in Formularen.

Möglicher verwendeter HTML-Code:
```
<input type="text" name="appfield" value="myvalue">
```

Wenn Sie in diesem Ausdruck den angezeigten Wert für das Attribut "value" aus einer PLC-Variable lesen sieht der HTML-Code folgendermaßen aus:
```
<input type="text" name="appfield" value=":="Varname".MyString:">
```

Durch das Lesen der PLC-Variable wird vom Webserver der Wert 'ABC' ausgegeben. In HTML stellt sich der Code dann wie folgt dar:
```
<input type="text" name="appfield" value=" 'ABC' ">
```

Wenn Sie in Ihrem HTML-Code statt doppelter Anführungszeichen einfache Anführungszeichen zum Einschließen der Attribute verwendet haben, dann liefert der Webserver den Inhalt der Variablen eingeschlossen in je zwei einfache Anführungszeichen an den Browser. Dies führt dazu, dass der Browser den Inhalt der String- oder Character-Variable nicht ausgibt, da zwei aufeinander folgende einfache Anführungszeichen je eine in sich geschlossene Sequenz bilden. Die zu lesenden Werte befinden sich zwischen diesen Sequenzen und werden vom Browser nicht ausgegeben.

Beachten Sie hierbei insbesondere, dass die Zeichenfolge doppeltes Anführungszeichen mit zwei einfachen Anführungszeichen nicht identisch ist, auch wenn diese augenscheinlich identisch aussehen.

Hinweis

Die Anpassung des Codes erfolgt nicht automatisch bei einem Update auf eine Firmware ab V1.6.
Passen Sie Ihren HTML-Code an, wenn Sie einfache Anführungszeichen zum Einschließen von Attributen bei der Funktion "PLC-Variablen lesen" verwendet haben.

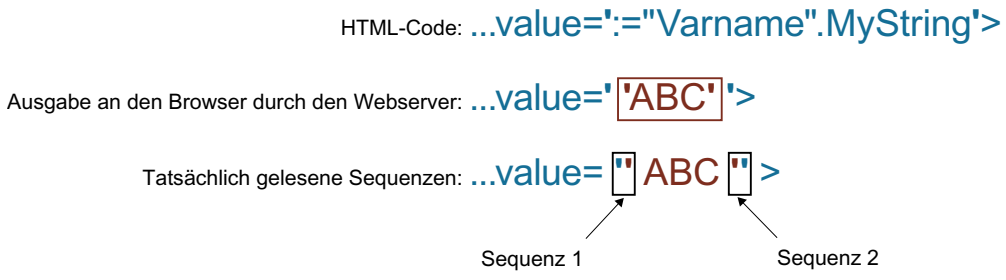

Variablen schreiben

Anwenderdefinierte Webseiten können Daten in die CPU schreiben.

Voraussetzung ist ein AWP-Kommando, das die zu schreibende PLC-Variable identifiziert.

Die PLC-Variable muss ferner durch einen PLC-Variablennamen spezifiziert sein.

Die In-Variablen (Eingaberichtung von der Steuerung aus betrachtet) werden auf der Browserseite gesetzt. Dies kann beispielsweise in einem Formular erfolgen.

Die Variablen werden entweder im HTTP-Header (per Cookie oder POST-Methode) oder in der URL (GET-Methode) vom Browser im HTTP-Request gesetzt und dann vom Webserver in die jeweilige PLC-Variable geschrieben.

Syntax

Damit die In-Variablen in die CPU geschrieben werden können, ist es notwendig, die Variablen durch eine explizite AWP-Anweisung zu definieren:

```
<!-- AWP_In_Variable Name='"<Varname1>"' Name='"<Varname2>"'
Name='"<Varname3>"' -->
```

In einer Anweisung können - wie oben - mehrere Variablen definiert werden.

Für den Fall, dass der Name der Variable, die Sie für die Webapplikation verwenden, nicht mit dem Namen der PLC-Variable identisch ist, kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen.

```
<!-- AWP_In_Variable Name='<Varname_Webapp>' Use='<PLC_Varname>'
```

Beispiel

Das AWP-Kommando "AWP_In_Variable" ist besonders für die Behandlung von Formularen unverzichtbar.

```
<form method="post" action="/awp/appl/x.html">
 <p>
  <input name='"var1"' type="text">
  <input value="set" name='"Button1"' type="submit">
 </p>
</form>
```

Im oben definierten Formular wird per "post" die Variable "var1" und "Button1" zum Webserver übertragen. Die Variable "var1" setzt der Anwender im Formularfeld. Die Variable "Button1" hat den fixen Wert "set". Damit die Variablen "var1" und "Button1" in der CPU geschrieben werden können, muss im gleichen Fragment folgende Anweisung aufgenommen werden:

```
<!-- AWP_In_Variable Name='"var1"' Name='"Button1"' -->
```

Da globale Variablen mit Anführungszeichen umschlossen werden, muss der Name im AWP-Kommando mit Apostrophen oder mit maskierten Anführungszeichen (\") eingeschlossen sein.

```
<!-- AWP_In_Variable Name='"Info".par1' -->
```

Bedingungen für den Schreibzugriff im Betrieb

Damit ein Anwender von einer anwenderdefinierten Webseite aus auf PLC-Variablen schreiben kann, muss folgende Voraussetzung erfüllt sein:

- Ein Benutzer wurde eingerichtet und ist angemeldet.

Diese Regel gilt prinzipiell für alle schreibenden Zugriffe von Webseiten auf eine CPU.

Siehe auch

Benutzerverwaltung (Seite 1326)

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Spezialvariablen

Bei den Spezialvariablen handelt es sich hauptsächlich um die so genannten HTTP-Variablen die in den Definitionen des World Wide Web Consortium (W3C) festgelegt sind. Auch für Cookies und Server-Variablen werden Spezialvariablen verwendet.

Die AWP-Kommandos zum Lesen und Schreiben von Spezialvariablen unterscheiden sich nur durch zusätzliche Parameter von den AWP-Kommandos zum Lesen und Schreiben normaler Variablen.

Spezialvariable lesen

Der Webserver kann PLC-Variablen lesen und sie an Spezialvariablen im HTTP-Response-Header übergeben. Sie können z. B. eine URL für eine Umleitung zu einer anderen Webseite lesen und an die Spezialvariable HEADER:Location übergeben mit Hilfe der Spezialvariable HEADER:Location.

Folgende Spezialvariablen können gelesen werden:

Name	Beschreibung
COOKIE_VALUE:name	Wert des Cookies mit dem Namen: "name"
COOKIE_EXPIRES:name	Ablaufzeit des Cookies mit dem Namen: "name" in Sekunden (muss zuvor gesetzt worden sein).
HEADER:Status	HTTP-Statuscode (wenn kein anderer Wert gesetzt wurde wird Statuscode 302 zurückgegeben).
HEADER:Location	Pfad für die Weiterleitung auf eine andere Seite. Statuscode 302 muss gesetzt sein.
HEADER:Retry-After	Zeit in der der Service voraussichtlich nicht verfügbar ist. Statuscode 503 muss gesetzt sein.
HEADER: …	Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden.

Mit dem AWP-Kommando "AWP_Out_Variable" wird festgelegt, welche PLC-Variablen im HTTP-Header an den Webbrowser übertragen werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_Out_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Parameterbeschreibung

- Name: Typ und Name der Spezialvariablen
- Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen.

Beispiel:

```
<!-- AWP_Out_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Spezialvariable schreiben

Prinzipiell können alle vom Webbrowser in den HTTP-Header geschriebenen HTTP-Variablen vom Anwenderprogramm der CPU ausgewertet werden. Beispiele für Variablentypen:

Name	Beschreibung
HEADER:Accept-Language	Akzeptierte bzw. bevorzugte Sprache
HEADER:Authorization	Berechtigungsnachweis für eine angeforderte Ressource
HEADER:Host	Host und Port der angeforderten Ressource
HEADER:User-Agent	Informationen zum Browser
HEADER: …	Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden
SERVER:current_user_id	Zeigt an, ob ein Benutzer angemeldet ist (current_user_id=0: kein Benutzer angemeldet)
SERVER:current_user_name	Benutzername des angemeldeten Benutzers
SERVER:GET	Request-Methode ist GET
SERVER:POST	Request-Methode ist POST
COOKIE_VALUE:name	Wert des Cookies mit dem Namen: "name"

Mit dem AWP-Kommando "AWP_In_Variable" wird festgelegt, welche Spezialvariablen im Anwenderprogramm der CPU ausgewertet werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_In_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Parameterbeschreibung:

Name: Typ und Name der Spezialvariablen

Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter Use die Zuordnung zu einer PLC-Variablen erfolgen.

Beispiele:

```
<!-- AWP_In_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablennamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='COOKIE_VALUE:siemens' Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablennamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='"COOKIE_VALUE:siemens"' -->
```

Die HTTP-Header-Variable wird in die gleichnamige PLC-Variable geschrieben.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Enumerationstypen

Enumerationstypen (Enums)

Mit Hilfe von Enums können numerische Werte aus dem PLC-Programm in Texte umgewandelt werden und umgekehrt. Die Zuordnung der numerischen Werte kann auch für mehrere Sprachen erfolgen.

Enums anlegen

Geben Sie ein AWP-Kommando mithilfe folgender Syntax am Anfang der HTML-Datei ein:

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Für z. B. deutsche Werte abzulegen als HTML-Datei im Ordner "de" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"an", 1:"aus", 2:"Störung"' -->
```

Für z. B. englische Werte abzulegen als HTML-Datei im Ordner "en" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -->
```

Enums zuordnen

Die Zuordnung der Variablen aus dem Anwenderprogramm zu den einzelnen Enum-Texten erfolgt über ein eigenes AWP-Kommando:

```
<!-- AWP_Enum_Ref Name="<VarName>" Enum="<EnumTypeName>" -->
```

Dabei ist <VarName> der symbolische Name aus dem Anwenderprogramm und <EnumTypeName> der zuvor festgelegte Name des Enum-Typs.

Hinweis

In jedem Fragment, in dem eine PLC-Variable Enum-Texte referenziert, muss diese PLC-Variable mit dem entsprechenden AWP-Kommando dem Enum-Typ-Namen zugeordnet werden.

Beachten Sie, dass zwischen einer Enum-Zuordnung und Enum-Verwendung kein AWP-Kommando zum Import von Fragmenten steht, da durch diesen Import die Enum-Zuordnung in einem anderen Fragment liegt als die Enum-Verwendung.

Beispiel

Es ist ein Enum-Typ "zustand" mit den Werten "0" und "1" definiert. "0" entspricht "aus", "1" entspricht "ein":

```
<!-- AWP_Enum_Def Name="zustand" Values='0:"aus", 1:"ein"' -->
```

Im HTML-Code der auszugebenden Webseite ist der folgende Code enthalten:

```
<!-- AWP_Enum_Ref Name="betriebszustand" Enum="zustand" -->
:=betriebszustand:
```

Das Anzeigeergebnis ist je nach Wert der Variable "betriebszustand" nun nicht mehr "0" oder "1", sondern "aus" oder "ein".

Vereinfachte Nutzung von Enumerationstypen

Bei S7-1200-CPUs ab Firmware-Version 4 ist es möglich, Enumerationen direkt in AWP-Kommandos zum Lesen und Schreiben von PLC-Variablen zu verwenden.

Sie legen wie im vorigen Abschnitt beschrieben Enums an und können dann mit Lese- und Schreib-AWP-Kommandos die Werte nutzen.

Enums anlegen

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Enums in AWP-Schreib- und Lese-Kommandos nutzen

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

Beispiel zum Lesen von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_Out_Variable Name='"Alarm"'
Enum="AlarmEnum" -->...<p>The current value of "Alarm"
is :="Alarm":</p>
```

Wenn der Wert von "Alarm" in CPU "2" ist, wird auf der HTML-Seite folgender Text angezeigt:

'The current value of "Alarm" is Tank is empty' weil die Enum-Definition den String "Tank is empty" dem numerischen Wert 2 zuordnet.

Beispiel zum Schreiben von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_In_Variable Name='"Alarm"'
Enum='AlarmEnum' -->...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /></p>
</form>
```

Weil die Enum-Definition den String "Tank is full" dem numerischen Wert "1" zuordnet, wird der Wert "1" in die PLC-Variable "Alarm" geschrieben.

Arrays verwenden

Der Webserver stellt die AWP-Kommandos AWP_Start_Array und AWP_End_Array zum Zugriff auf alle Werte eines Arrays zur Verfügung.

Nur eindimensionale Arrays werden unterstützt.

Multidimensionale Array der Form array[x][y] werden nicht unterstützt.

Syntax

```
<!-- AWP_Start_Array Name='"<DB name>".<array name>' -->
... Inhalt des Arrays, verwendete Schlüsselwörter: ArrayIndex und
value..
<!-- AWP_End_Array -->
```

Parameter

<Name>	Definiert den Namen des Arrays, auf dessen Elemente Sie zugreifen wollen. Sie benötigen den DB-Namen und den Namen des Arrays entsprechend der in STEP 7 definierten Datenbausteinstruktur. Der Name muss in einfachen oder doppelten Anführungszeichen stehen. Der DB-Name steht in doppelten Anführungszeichen.
<ArrayIndex>	Index eines Array-Elements
<value>	Wert eines Array-Elements

Beispiel

Das Beispiel liest alle Elemente der Struktur "MyArray" im Datenbausteine "DB_Name" von der CPU und zeigt den Index und die Werte der Variablen auf der anwenderdefinierten Webseite.

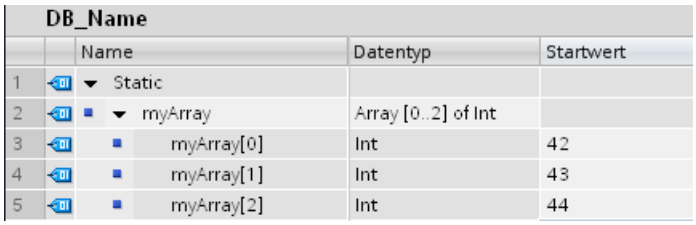

DB_Name

	Name	Datentyp	Startwert
1	Static		
2	myArray	Array [0..2] of Int	
3	myArray[0]	Int	42
4	myArray[1]	Int	43
5	myArray[2]	Int	44

```
<!-- AWP_Start_Array Name='"DB_Name".MyArray' -->
Index: :=ArrayIndex: Wert: :=value:
<!-- AWP_End_Array -->
```

Der oben gezeigte Code erzeugt folgende Anzeige:

Index: 1 Wert: 42

Index: 2 Wert: 43

Index: 3 Wert: 44

Darstellung von Arrays vom Datentyp Bool bei S7-1500

Die Ausgabe von Arrays vom Typ Bool wird immer auf die nächsten vollen 8 Bit aufgefüllt. Diese Besonderheit tritt lediglich bei Bool-Arrays auf.

Beispiel:

```
"DB_1".bitArray ist ein Bool-Array mit 5 Elementen.
<!-- AWP_Start_Array Name='"DB_1".bitArray' -->
:=ArrayIndex: -> :=value:
<!-- AWP_End_Array -->
```

Ausgabe:

0 -> Wert aus "DB_1".bitArray[0]

1 -> Wert aus "DB_1".bitArray[1]

2 -> Wert aus "DB_1".bitArray[2]

3 -> Wert aus "DB_1".bitArray[3]

4 -> Wert aus "DB_1".bitArray[4]

5 -> 0

6 -> 0

7 -> 0

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Strukturen verwenden

Der Webserver stellt AWP-Kommandos zum Zugriff auf Strukturen zur Verfügung, um auf die Werte einer PLC-Variablen vom Datentyp STRUCT zuzugreifen.

Syntax

```
<!-- AWP_Start_Struct Name='"<DB name>".<struct name>' -->
... Inhalt der Struktur ...
<!-- AWP_End_Struct -->
```

Parameter

<Name>	Definiert den Namen der Struktur, auf dessen Elemente Sie zugreifen wollen. Sie benötigen den DB-Namen und den Namen der Struktur entsprechend der in STEP 7 definierten Datenbausteinstruktur. Der Name muss in einfachen oder doppelten Anführungszeichen stehen. Der DB-Name steht in doppelten Anführungszeichen.

Beispiel

Das Beispiel liest Elemente der Struktur "MyStruct" im Datenbausteine "DB_Name" von der CPU und zeigt den Wert der Variable auf der anwenderdefinierten Webseite.

DB_Name

	Name	Datentyp	Startwert
1	Static		
2	MyStruct	Struct	
3	A	Int	0
4	B	Word	16#0
5	C	Bool	false

```
<!-- AWP_Start_Struct Name='"DB_Name".MyStruct' -->
:=A:
:=B:
:=C:
<!-- AWP_End_Struct -->
```

Der oben gezeigte Code entspricht folgenden Kommandos:

```
:="DB_Name".MyStruct.A:
:="DB_Name".MyStruct.B:
:="DB_Name".MyStruct.C:
```

Regeln für Arrays und Strukturen

Arrays und Strukturen können verschachtelt sein.

Regeln

AWP_Start_Array- und AWP_End_Array-Kommandos müssen paarweise verwendet werden. Sie dürfen sich nicht überlappen.

Zulässige Konstruktion:

AWP_Start_Array

AWP_Start_Struct

AWP_End_Struct

AWP_End_Array

Nicht zulässige Konstruktion (Überlappung):

AWP_Start_Array

AWP_Start_Struct

AWP_End_Array

AWP_End_Struct

Definieren von Fragmenten

Fragmente

Unter Fragmenten versteht man einzeln von der CPU zu bearbeitende "Sinnabschnitte" einer Webseite.

Fragmente sind meist ganze Seiten, können aber auch Einzelelemente sein, wie Dateien (z. B. Bilder) oder ganze Dokumente.

Fragmente definieren

```
<!-- AWP_Start_Fragment Name="<Name>" [Type="<Type>"] [ID="<Id>"]
[Mode=<Mode>]-->
```

Der Beginn eines Fragmentes wird mit diesem Kommando festgelegt. Ein Fragment erstreckt sich bis zum Beginn des nächsten Fragments oder bis zum Ende der Datei.

- <Name> Gibt den Namen des Fragments an.
  Der Name muss mit einem Buchstaben [a-zA-Z] oder einem Unterstrich (_) beginnen. Nach diesem ersten Zeichen können Buchstaben, Unterstriche oder Zahlen [0-9] folgen.
- <Type> Gibt den Typ des Fragments an.
  - "manual" Das Anwenderprogramm wird über den Request für ein Fragment benachrichtigt, ein Eingriff in die zurückzuliefernde Webseite durch das Anwenderprogramm ist möglich.
  - "automatic" Die Seite wird automatisch bearbeitet (default).
- <id> Für das Fragment kann eine numerische ID vorgegeben werden. Wenn keine ID vergeben wird, dann wird dem Fragment automatisch eine ID zugewiesen. Bei manuellen Seiten (<Type>=manual) kann das Fragment im Anwenderprogramm der CPU mit dieser ID angesprochen werden.

  Hinweis

  Setzen Sie die ID niedrig an, da die höchste ID die Größe des Web-Control-DBs beeinflusst.

- <Mode> Fragmente unterstützen die beiden Modi visible und hidden.
  - "visible" Das Fragment ist ein Teil der Webseite. Dieser Mode ist voreingestellt und kann auch weggelassen werden.
  - "hidden" Das Fragment ist nicht Teil der Webseite. Das Fragment wird aber im Web-DB gespeichert und steht dem Anwenderprogramm zum Einfügen in eine angeforderte Webseite zur Verfügung. Über einen Austausch der Fragment-ID (Variable Web-Control-DB.fragment_index) fügen Sie ein "hidden"-Fragment in die angeforderte Webseite ein.

Das Input Dokument wird durch das "AWP_Start_Fragment"-Kommando vollständig in Fragmente aufgeteilt. Daher ist ein "AWP_End_Fragment" nicht notwendig.

Ohne ein Start-Fragment-Kommando wird eine Datei als ein Fragment abgebildet, der Fragmentname leitet sich aus dem Dateinamen ab. Wird eine Datei in mehrere Fragmente zerlegt (durch "AWP_Start_Fragment"), so muss die Datei mit dem "AWP_Start_Fragment"-Kommando beginnen.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Importieren von Fragmenten

Sie haben die Möglichkeit, in einer HTML-Seite ein Fragment zu deklarieren und dieses Fragment in anderen Webseiten zu importieren.

Beispiel

Ein Firmen-Logo soll auf allen Webseiten einer Webapplikation angezeigt werden.

Der HTML-Code für das Fragment, das das Firmen-Logo anzeigt, existiert nur einmal. Sie können das Fragment so oft importieren und in so vielen HTML-Dateien importieren wie erforderlich.

Syntax

```
<!-- AWP_Import_Fragment Name = "<name>"-->
```

- <name> ist der Name des zu importierenden Fragments.

Beispiel

HTML-Code innerhalb einer Webseite, der ein Fragment deklariert:

```
<!-- AWP_Start_Fragment Name = "My_Company_Logo" -->
<p><img src = "compay_logo.jpg"></p>
```

Beispiel

HTML-Code innerhalb einer anderen Webseite, der das deklarierte Fragment importiert:

```
<!-- AWP_Import_Fragment Name = "My_Company_Logo" -->
```

Datenbausteine generieren und laden

Voraussetzung

- Sie haben alle benötigten Quelldateien (*.html, *.js, *.png, ...) für die Webapplikation erstellt.
- Die Quelldateien liegen in einem Ordner; und zwar nur die Quelldateien, die für die Webapplikation benötigt werden! Es dürfen keine anderen Dateien in diesem Ordner liegen.

Hinweis

Länge von Dateinamen und Variablennamen

Wenn Sie eine umfangreiche Webapplikation mit vielen Dateien und Verzeichnissen haben, scheitert die Generierung der Web-Datenbausteine möglicherweise. In diesem Fall wird die Generierung mit der Meldung "Überlauf Textliste..." abgebrochen. Ursache sind systeminterne Größenbeschränkungen für Verwaltungsinformationen, die im Web-Datenbaustein gespeichert werden.

Abhilfe: Verwenden Sie kurze Dateinamen und kurze Variablennamen.

Vorgehen

Um aus den Quelldateien für anwenderdefinierte Webseiten in STEP 7 in Datenbausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU z. B. in der Gerätekonfiguration.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" die Eigenschaften für anwenderdefinierte Webseiten.
3. Als "HTML-Quelle" wählen Sie den Ordner, wo sich die Quelldateien für die Webapplikation befinden.
4. Als Start-HTML-Seite tragen Sie die HTML-Seite ein, die beim Start der Webapplikation geöffnet werden soll.
5. Wenn erforderlich, tragen Sie einen Applikationsnamen ein.
6. Als "Dateien mit dynamischem Inhalt" ergänzen Sie die Reihe von Dateinamenserweiterungen, falls erforderlich. Sie tragen nur solche Dateinamenserweiterungen ein, die auch AWP-Kommandos enthalten.
7. Die Nummer für den Web-Control-DB und für die Fragment-DB-Startnummer können Sie beibehalten, wenn Sie nicht bereits durch ihr Anwenderprogramm belegt werden.
8. Klicken Sie auf die Schaltfläche "Erzeugen", um DBs aus den Quelldateien zu erzeugen. Die erzeugten Datenbausteine werden in der Projektnavigation im Ordner "Systembausteine" (Unterordner "Webserver") abgelegt.
9. Markieren Sie in der Netzsicht die CPU, die geladen werden soll, und wählen Sie den Befehl "Laden in Gerät" im Menü "Online", um die Bausteine zu laden. Implizit wird vor dem Laden die Übersetzung der Bausteine angestoßen.
   Falls bei diesem Prozess Fehler gemeldet werden, müssen Sie diese Fehler beheben, bevor Sie die Konfiguration laden können.

Aufbau des PLC-Programms

Ihr Anwenderprogramm muss die Anweisung " WWW" aufrufen, damit die Webapplikation, d. h. die anwenderdefinierten Webseiten, überhaupt auf den Standard-Webseiten der CPU verfügbar sind und aufgerufen werden können.

Der Web-Control-DB, den Sie aus den Quelldateien generiert haben, ist der Eingangsparameter (CTRL_DB) für die Anweisung "WWW". Der Web-Control-DB referenziert den Inhalt der anwenderdefinierten Webseiten, der in den Fragment-DBs codiert ist, und enthält Status- und Steuerinformationen.

Aufruf der Anweisung "WWW" im Anlaufprogramm

Wenn Sie angeforderte Webseiten nicht durch das Anwenderprogramm beeinflussen wollen, dann genügt ein einmaliger Aufruf der Anweisung "WWW" in einem Analuf-OB. Die Anweisung initialisiert die Kommunikation zwischen dem Webserver und der CPU.

Aufruf der Anweisung "WWW" im zyklischen Programm

Der Aufruf der Anweisung "WWW" kann auch in einem zyklisch bearbeiteten OB (z. B. dem OB 1) erfolgen. Dies bietet den Vorteil, dass auf Anfragen des Webservers im Anwenderprogramm eingegangen werden kann. Dazu müssen manuelle Fragmente verwendet werden.

In diesem Fall müssen Sie Informationen aus dem Web-Control-DB auswerten, um die angeforderte Webseite bzw. das angeforderte Fragment zu identifizieren. Andererseits müssen Sie im Anwenderprogramm ein Bit setzen, um nach Bearbeitung der Webseiten-Anforderung die vom Webserver auszuliefernde Webseite explizit freizugeben.

Der Aufbau des Web-Control-DBs ist im folgenden Abschnitt beschrieben.

Web-Control-DB

Der Web-Control-DB (Standardmäßig DB 333) wird von STEP 7 erzeugt und enthält Information zum Aufbau der Anwenderseiten, zum Status der Kommunikation und eventuell aufgetretenen Fehlern.

Neben dem Web-Control-DB werden noch weitere so genannte Fragment-DBs erzeugt. Diese Fragment-DBs (es kann auch nur ein Fragment-DB sein) werden im Web-Control-DB referenziert. Die Fragment-DBs enthalten die in Fragmenten codierten Webseiten und Mediadaten, z. B. Bilder. Der Inhalt der Fragment-DBs ist nicht durch das Anwenderprogramm änderbar. Sie werden automatisch angelegt und dienen nur der Datenhaltung.

Auf die Status- und Steuervariablen des Web-Control-DB greifen Sie symbolisch zu.

Im Folgenden sind die Variablen des Web-Control-DBs aufgeführt, die Sie für die Statusauswertung und für die Steuerung der Interaktion benötigen.

Aus dem Web-Control-DB können Sie zweierlei Informationen gewinnen:

- Globale Statusinformation: Nicht an eine konkrete Webseitenanforderung (Request) gebunden.
- Request Status- und Steuerinformationen: Informationen zu anstehenden Requests.

Globale Statusinformation

"WEB-Control_DB".commandstate.init	Aktiviert und initialisiert die Webapplikation.
"WEB-Control_DB".commandstate.deactivate	Deaktiviert die Webapplikation.
"WEB-Control_DB".commandstate.inititializing	Die Webapplikation wird initialisiert (Web-Control-DB lesen, etc.).
"WEB-Control_DB".commandstate.error	Webapplikation konnte nicht initialisiert werden. Der Grund ist in "WEB-Control_DB".commandstate.last_error codiert.
"WEB-Control_DB".commandstate.deactivating	Die Webapplikation wird beendet.
"WEB-Control_DB".commandstate.initialized	Die Webapplikation wurde initialisiert und ist bereit.
"WEB-Control_DB".commandstate.last_error	Wertetabelle der möglichen Fehler siehe nächste Tabelle.

Last_error	Beschreibung
1	Fragment-DB ist inkonsistent (passt nicht zum Web-Control-DB).
2	Eine Webapplikation mit diesem Namen existiert schon.
3	Speicherproblem beim Initialisieren im Webserver.
4	Die Daten im Web-Control-DB sind inkonsistent.
5	Ein Fragment-DB ist nicht verfügbar (nicht geladen).
6	Ein Fragment-DB hat keine AWP-Kennung.
7	Das Enum Fragment (enthält die Texte und Informationen für die Enum-Typen) ist nicht verfügbar.
8	Ein über die Command-Flags im Web-Control-DB angeforderte Aktion ist im aktuellen Zustand nicht erlaubt.
9	Webapplikation ist nicht initialisiert (wenn keine neue Initialisierung nach Deaktivierung).
10	Webserver ist deaktiviert.
...	Last_error wird zurückgesetzt, wenn die Webapplikation erfolgreich initialisiert wurde.

Request-Statusinformation

Die Request-Statusinformation ist gebunden an einen von vier möglichen Requests, x = [1 … 4].

"WEB-Control_DB".requesttab[x].idle	Es gibt nichts zu tun.
"WEB-Control_DB".requesttab[x].waiting	Das Anwenderprogramm muss auf einen Request aus einem manuellen Fragment reagieren und die Weiterverarbeitung im Webserver explizit anstoßen.
"WEB-Control_DB".requesttab[x].sending	Der Webserver ist mit der Bearbeitung des Requests/Fragments beschäftigt.
"WEB-Control_DB".requesttab[x].aborting	Die TCP-Verbindung wird vom Webserver geschlossen.

Request-Steuerinformation

Die Request-Steuerinformation ist gebunden an einen von vier möglichen Requests, x = [1 … 4].

"WEB-Control_DB".requesttab[x].continue	Gibt das gerade bearbeitete Fragment zum Senden frei. Die Bearbeitung des nächsten Fragments wird angestoßen.
"WEB-Control_DB".requesttab[x].repeat	Gibt das gerade bearbeitete Fragment zum Senden frei. Das Fragment wird anschließend erneut bearbeitet.
"WEB-Control_DB".requesttab[x].abort	Schließt die TCP-Verbindung.
"WEB-Control_DB".requesttab[x].finish	Gibt das gerade bearbeitete Fragment zum Senden frei. Stoppt die weitere Bearbeitung von Fragmenten (beendet den Request).

Beispiel:

Die Variable für den DB ist: "WEB-Control_DB". Ob bei der Initialisierung der Webapplikation Fehler aufgetreten sind, können Sie anhand des Bits "WEB-Control_DB".commandstate.error im Anwenderprogramm abfragen.

Für den Fall, dass ein Fehler aufgetreten ist, können Sie den Fehler über den Wert von "WEB-Control_DB".commandstate.last_error analysieren.

Interaktion mit dem Anwenderprogramm

Mithilfe von manuellen Fragmenten erreichen Sie, dass das Anwenderprogramm synchron auf Browsereingaben reagiert, sodass die zurückzugebende Webseite vom Anwenderprogramm aufbereitet werden kann.

Fragmenttyp

Um im Anwenderprogramm direkt auf die empfangenen Daten reagieren zu können muss für das Fragment, das die Daten schreibt, der Fragmenttyp "manual" (für "manuelle Seiten") verwendet werden:

```
<!-- AWP_Start_Fragment Name="testfrag" ID="1" Type="manual" -->
```

Das Übertragen der Werte an den Webserver der CPU erfolgt bei automatischen und manuellen Seiten immer auf die gleiche Weise:

Beispiel:

```
<form method="POST" action="">
<p>
<input type="submit" value="Set new value">
<input type="text" name='"Velocity"' size="20">
</p>
</form>
```

Anwenderprogramm für manuelle Fragmente

Im Anwenderprogramm der CPU muss bei der Verwendung manueller Seiten die Anweisung "WWW" (SFC 99) zyklisch aufgerufen werden.

Um auf im Browser eingegebene Werte regieren zu können, ist es notwendig, den Request, den die manuelle Seite an den Webserver stellt, im Anwenderprogramm auszuwerten. Dazu muss der Web-Control-DB (z. B. DB 333) zyklisch auf anstehende Requests untersucht werden. Im Abschnitt "requesttab" des Web-Control-DBs befindet sich das Array, das vier Requests verwaltet. Jedes Element des Arrays enthält dabei, in einer Struktur, Informationen zum jeweiligen Request.

Ein einfaches AWL-Programmbeispiel zeigt, wie anhand der Variablen des Web-Control-DBs auf anstehende Requests geprüft wird.

```
      U     "WEB-Control_DB".requesttab[1].waiting
      SPBN  R2
      L     "WEB-Control_DB".requesttab[1].fragment_index
      L     1
      SPA   PdRq                        // request pending
R2:   U     "WEB-Control_DB".requesttab[2].waiting
      SPBN  R3
      L     "WEB-Control_DB".requesttab[2].fragment_index
      L     2
      SPA   PdRq                        // request pending
R3:   U     "WEB-Control_DB".requesttab[3].waiting
      SPBN  R4
      L     "WEB-Control_DB".requesttab[3].fragment_index
      L     3
      SPA   PdRq                        // request pending
R4:   U     "WEB-Control_DB".requesttab[4].waiting
      SPBN  NoRq
      L     "WEB-Control_DB".requesttab[4].fragment_index
      L     4
      SPA   PdRq                        // request pending
NoRq: L     0
      T     #frag_index
      T     #req_index
      SPA   Exit
PdRq: T     #req_index
      POP
      T     #frag_index
Exit: BE
```

Für den Fall, dass ein Request vorliegt, schreibt dieser AWL-Programmabschnitt die Fragment-ID in die Variable #frag_index und die Request-Nr. (Wertebereich 1-4) in die Variable #req_index.

Mit den so gewonnenen Informationen kann die im Request übertragene Information jetzt im AWL-Programm für jede Fragment-ID gesondert bearbeitet werden (z. B. Plausibilitätsprüfung).

Wenn die Bearbeitung des Requests durch das AWL-Programm abgeschlossen ist, muss der Request beantwortet und der entsprechende Eintrag unter "requesttab" des Web-Control-DBs (z. B. DB 333) wieder zurückgesetzt werden.

Ein einfaches AWL-Programmbeispiel zur Beantwortung von Requests:

```
SET
       L     #req_index
      SPL   Con0
      SPA   Con0
      SPA   Con1
      SPA   Con2
      SPA   Con3
      SPA   Con4
Con0: SPA   Exit
Con1: =     "WEB-Control_DB".requesttab[1].continue
      SPA   Exit
Con2: =     "WEB-Control_DB".requesttab[2].continue
      SPA   Exit
Con3: =     "WEB-Control_DB".requesttab[3].continue
      SPA   Exit
Con4: =     "WEB-Control_DB".requesttab[4].continue
      SPA   Exit
Exit: BE
```

Prinzipieller Ablauf eines Browser-Requests mit Interaktion des Anwenderprogramms

Das folgende Bild zeigt den vereinfachten, prinzipiellen Ablauf vom Webbrowser-Request über die Auswirkungen auf Web-Control-DB-Inhalte und die notwendigen Aktionen des Anwenderprogramms bis zum Zurückschicken der aufbereiteten Webseite (Response).

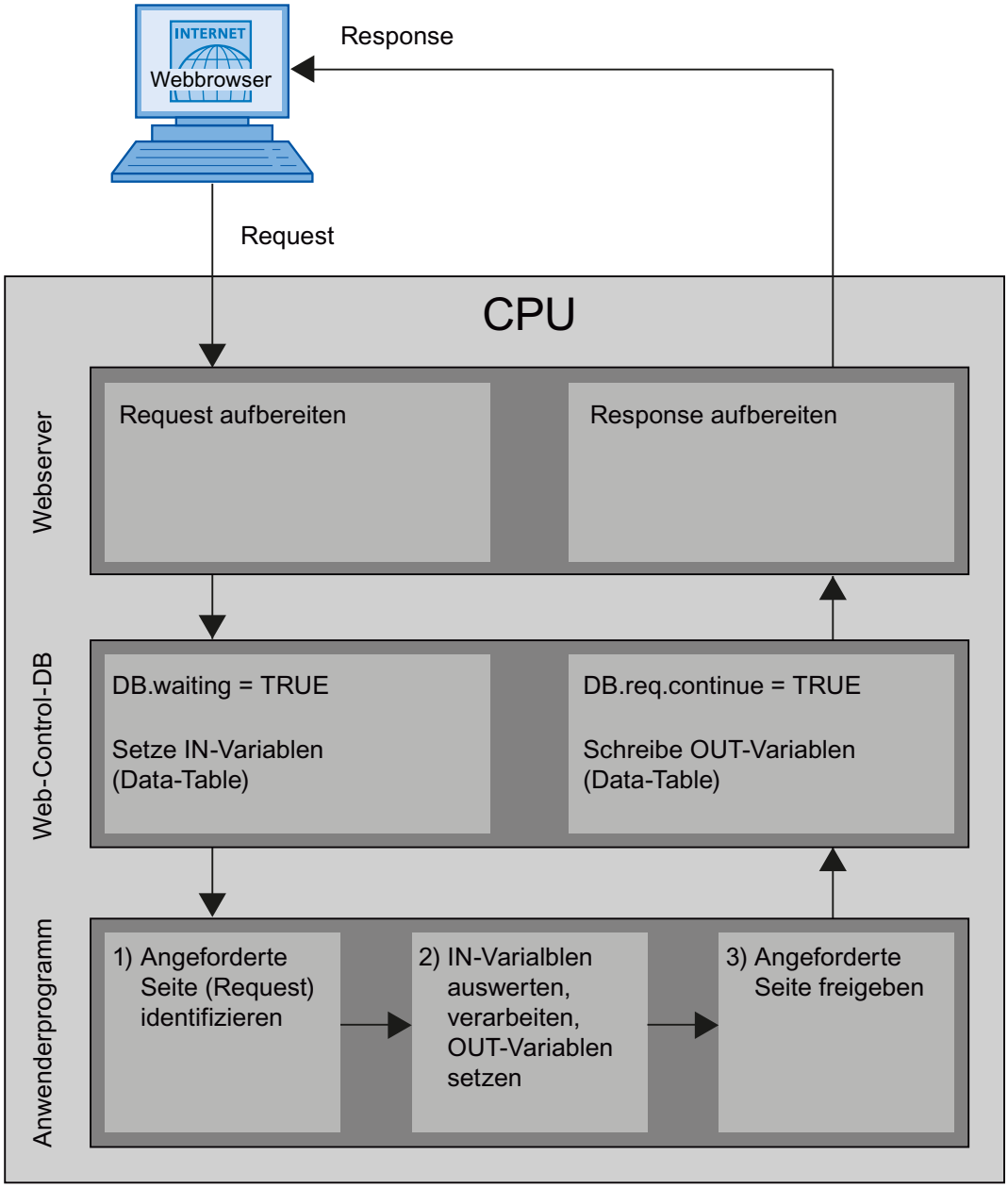

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Anzeigen benutzerdefinierter Webseiten im Browser

Webseiten im Browser anzeigen

Die Webseiten rufen Sie von den Standard-Webseiten des Webbrowsers auf.

Die Standard-Webseiten verfügen neben den anderen Links in der Navigationsleiste einen Link auf "Anwenderseiten".

Wenn Sie auf den Link "Anwenderseiten" klicken, öffnet ihr Webbrowser die Webseite, die Sie als Start-HTML-Seite konfiguriert haben.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Anwenderdefinierte Webseiten mehrsprachig erstellen

Sie haben die Möglichkeit, Ihre anwenderdefinierten Webseiten in jeweils unterschiedlichen Landessprachen zur Verfügung zu stellen.

Voraussetzungen

Die sprachabhängigen HTML-Seiten sind in einer Ordnerstruktur abzulegen, die Ordner mit den jeweiligen Sprachkürzeln enthält:

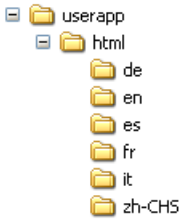

Festgelegte Sprachkürzel

Die Sprachkürzel "de", "en", "fr", "es", "it" und "zh-CHS" sind festgelegt. Weitere Sprachordner bzw. anders bezeichnete Sprachordner werden nicht unterstützt.

Weitere Ordner in derselben Ordnerhierarchie für andere Dateien können Sie bei Bedarf anlegen; z. B. einen Ordner "img" für Bilder und einen Ordner "script" für Javascript-Dateien.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Sprachumschaltung für anwenderdefinierte Webseiten

Voraussetzungen

Die HTML-Seiten befinden sich in den vordefinierten Sprachordnern, z. B. HTML-Seiten mit deutschem Text im Ordner "de", HTML-Seiten mit englischem Text im Ordner "en".

Konzept der Sprachumschaltung

Die Sprachumschaltung basiert auf einem vordefinierten Cookie mit Namen "siemens_automation_language". Wenn das Cookie auf den Wert "de" gesetzt wird, schaltet der Webserver bei der nächsten Webseiten-Anforderung bzw. Webseiten-Aktualisierung auf die Webseite aus dem Ordner "de".

Analog dazu schaltet der Webserver auf die Webseite aus dem Ordner "en" um, wenn das Cookie den Wert "en" annimmt.

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Beispiel für eine Sprachumschaltung

Das Beispiel ist folgendermaßen aufgebaut:

- In den beiden Sprachordnern "de" und "en" befinden sich die sprachabhängigen HTML-Dateien gleichen Namens, z. B. "langswitch.html". Die anzuzeigenden Texte innerhalb der beiden Dateien sind entsprechend den Ordernamen deutsch bzw. englisch.
- Zusätzlich befindet sich ein Ordner "script" in der Ordnerstruktur, in dem sich die Javascript-Datei "lang.js" befindet. In diese Datei sind Funktionen ausgelagert, die für die Sprachumschaltung benötigt werden.

Aufbau der Datei "langswitch.html" (Ordner "de")

Im Header der Datei sind die Metadaten "Content-Language", charset und Pfad zur Javascriptdatei festgelegt.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

Im Body der Datei ist die Sprachauswahl mithilfe des HTML-Elements "select" realisiert. Das select-Element leitet eine Auswahlliste ein und enthält die Optionen "de", beschriftet mit "Deutsch" und "en", beschriftet mit "Englisch"; "de" ist vorselektiert.

Mit Hilfe des event-handlers "onchange" wird die Funktion "DoLocalLanguageChange(this)" aufgerufen. Der Parameter "this" übergibt das select-Objekt mit der selektierten Option an diese Funktion. "onchange" führt bei jeder Änderung der Option zum Aufruf der Funktion.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" selected >Deutsch</option>
                <option value="en" >Englisch</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Aufbau der Datei "langswitch.html" (Ordner "en")

Der Header der HTML-Datei mit englischem Text ist analog zur HTML-Datei mit deutschem Text aufgebaut.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

Im Body der Datei ist die Sprachauswahl ebenfalls mit Hilfe des HTML-Elements "select" realisiert. Im Unterschied zur deutschen HTML-Datei ist die englische Option vorselektiert und der Text bzw. die Labels sind englisch.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" >German</option>
                <option value="en" selected >English</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Aufbau der Datei "lang.js" (im Ordner "script")

In der Javascript-Datei ist die Funktion " DoLocalLanguageChange" definiert, die ihrerseits die Funktion "SetLangCookie" mit dem Sprachauswahl-Wert aufruft. SetLangCookie setzt Cookie-Name und Cookie-Wert zusammen und setzt das Cookie schließlich über die entsprechende document.cookie-Eigenschaft. Damit der Webserver auf das Setzen des cookies mit der Anzeige der gewünschten Sprache reagiert, muss anschließend die Webseite neu geladen werden (top.window.location.reload).

```
function DoLocalLanguageChange(oSelect) {
        SetLangCookie(oSelect.value);
        top.window.location.reload();
    }

 function SetLangCookie(value) {
        var strval = "siemens_automation_language=";
            // this is the cookie by which the webserver
            // detects the desired language
            // this name is required by the webserver
        strval = strval + value;
        strval = strval + "; path=/ ;";
            // set path to the application, since otherwise
           // path would be set to the requesting page would not get
the cookie
            /*
            use expiration if this cookie should live longer
            than the current browser session
            var now      = new Date();
            var endttime = new Date(now.getTime() + expiration);
            strval = strval + "; expires=" + endttime.toGMTString()
+ ";";
            */
        document.cookie = strval;
    }
```

Siehe auch

Weiterführende Informationen und Beispielanwendungen (Seite 1362)

Weiterführende Informationen und Beispielanwendungen

Informationen und Beispiele für Anwenderdefinierte Seiten

Beispiele und weiterführende Erläuterungen finden Sie im Siemens Industrie Online-Support.

- Das Funktionshandbuch über den Webserver einer S7-1500-CPU finden Sie hier (http://support.automation.siemens.com/WW/view/de/62046437?Datakey=47071380).
- Ein Beispiel für den Zugriff auf den Webserver einer S7-1200 mit einem iPad und zum Einbinden der Bibliothek jQuery finden Sie hier (http://support.automation.siemens.com/WW/view/de/62046437?Datakey=47071380).

- Ein Applikationsbeispiel für die Erstellung einer Anwenderdefinierten Webseite finden Sie hier (http://support.automation.siemens.com/WW/view/de/58862931).
- Eine Beschreibung von Möglichkeiten, Anwenderdefinierte Webseiten mit HTML5-Code oder JavaScript zu aktualisierten, finden Sie hier (http://support.automation.siemens.com/WW/view/de/62543256).
- Eine Übersicht zum Thema Industrial Security finden Sie hier (http://support.automation.siemens.com/WW/view/de/50203404).

Einfache Beispiele für spezielle Anwendungen

Für die unterschiedlichen oben beschriebenen Features stellen wir Ihnen dokumentierte Beispiele mit Beispiel-HTML-Seiten und - je nach Anwendung - mit zugehörigem TIA Projekt/Anwenderprogramm zur Verfügung, siehe hier (https://support.industry.siemens.com/cs/ww/de/view/68011496).

Neben grundlegenden Beschreibungen und Projekten finden Sie dort auch einfache Beispiele für den Webserver von SIMATIC S7-1200/S7-1500.

Siehe auch

Webserver S7-1500 (http://support.automation.siemens.com/WW/view/de/59193560)

Quelldateien für Webserver-Anwenderseiten in STEP 7 (TIA Portal) nutzen

Bei der Migration von STEP 7-Projekten werden die Quellen für die Webserver-Anwenderseiten wie z. B. HTML-Seiten und Grafiken nicht automatisch in das neue Projekt im TIA Portal übernommen.

Nach der Migration müssen Sie die Quellen Ihrer Anwenderseiten in Ihr Projekt integrieren. Gehen Sie folgendermaßen vor:

1. Legen Sie die Quelldaten in ein Verzeichnis, das vom TIA Portal erreichbar ist.
2. Geben Sie in den Eigenschaften der CPU unter "Webserver > Anwenderseiten" die Pfade zu den Quelldateien an.
3. Übersetzen Sie anschließend die Hardware-Konfiguration.

1.4.2 Automatisierungssysteme konfigurieren

1.4.2.1 Arbeitsweise von S7-1200 CPUs

Betriebszustände

Grundlagen zu den Betriebszuständen von S7-CPUs

Einführung

Betriebszustände beschreiben das Verhalten der CPU. Folgende Betriebszustände sind möglich:

- ANLAUF
- RUN
- STOP

In diesen Betriebszuständen ist die CPU kommunikationsfähig, z. B. über die PN/IE-Schnittstelle.

Weitere Betriebszustände

Wenn die CPU nicht betriebsbereit ist, befindet sie sich in einem der beiden folgenden Betriebszustände:

- Spannungslos, d. h. die Netzspannung ist ausgeschaltet.
- Defekt, d. h. es ist ein interner Fehler aufgetreten.
  Wenn der Zustand "Defekt" durch einen Firmware-Fehler hervorgerufen wurde, wird dieser Zustand durch die Status-LEDs der CPU signalisiert (siehe Beschreibung zur CPU). Gehen Sie folgendermaßen vor, um die Ursache zu ermitteln:
  - Schalten Sie den Netzschalter aus und wieder ein.
  - Lesen Sie den Diagnosepuffer aus, wenn die CPU anläuft und senden die Daten zur Analyse an den Customer Support.

  Wenn die CPU nicht anläuft, tauschen Sie sie aus.

Siehe auch

Betriebszustand "STOP" (Seite 1370)

Betriebszustand "RUN" (Seite 1369)

Betriebszustand "ANLAUF" (Seite 1366)

Betriebszustandsübergänge

Übersicht

Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge von S7-1200-CPUs:

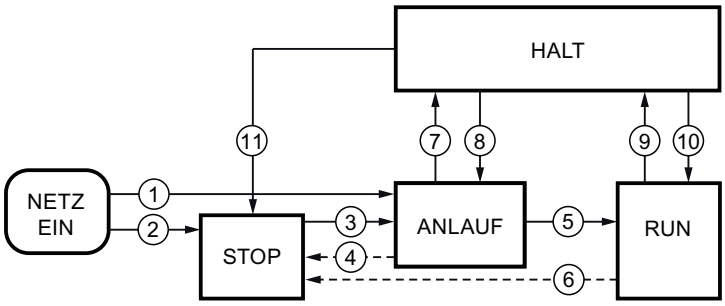

Die folgende Tabelle zeigt die Bedingungen, unter welchen die Betriebszustände wechseln:

Nr.	Betriebszustandsübergang	Bedingungen
①	NETZ EIN → ANLAUF	Die CPU geht nach dem Einschalten in den Betriebszustand "ANLAUF", wenn: • die Anlaufart "Warmstart" eingestellt ist und • die Hardware-Konfiguration und die Programmbausteine konsistent sind. Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten.
②	NETZ EIN → STOP	Bei Einstellung der Anlaufart "Kein Anlauf" geht die CPU nach dem Einschalten der Versorgungsspannung in den Betriebszustand "STOP". Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten.
③	STOP → ANLAUF	Die CPU geht in den Betriebszustand "ANLAUF", wenn: • die CPU vom Programmiergerät aus auf "RUN" gesetzt wird und • die Hardware-Konfiguration und die Programmbausteine konsistent sind.
④	ANLAUF → STOP	Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": • Während des Anlaufs wird ein Fehler erkannt. • Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anlauf-OB bearbeitet.
⑤	ANLAUF → RUN	Wenn der Anlauf erfolgreich ist, wechselt die CPU in "RUN".
⑥	RUN → STOP	Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": • Ein Fehler wird erkannt, der die Weiterarbeit verhindert. • Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt. • Ein STOP-Befehl wird im Anwenderprogramm bearbeitet.

Betriebszustand "ANLAUF"

Grundlagen zum Betriebszustand "ANLAUF"

Funktion

Bevor die CPU nach dem Einschalten mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, wird ein Anlaufprogramm bearbeitet.

Im Anlaufprogramm können Sie durch entsprechende Programmierung von Anlauf-OBs bestimmte Initialisierungs-Variablen für Ihr zyklisches Programm festlegen. Die Anzahl von Anlauf-OBs ist nicht vorgeschrieben. Sie haben die Möglichkeit, keinen Anlauf-OB, einen Anlauf-OB oder mehrere Anlauf-OBs zu programmieren.

Parametrieren des Anlaufverhaltens

Sie können festlegen, ob die CPU im Betriebszustand STOP verbleiben soll, oder ob ein Warmstart durchgeführt wird. Darüber hinaus können Sie das Verhalten im Anlauf (RUN oder vorheriger Betriebszustand) in der Gruppe "Anlauf" der CPU-Eigenschaften parametrieren.

Besonderheiten

Beachten Sie für den Betriebszustand "ANLAUF" folgende Punkte:

- Die Anlauf-OBs werden abgearbeitet. Unabhängig von der gewählten Anlaufart werden alle programmierten Anlauf-OBs bearbeitet.
- Keine zeitgesteuerte Programmbearbeitung möglich.
- Alarmgesteuerte Programmbearbeitung beschränkt auf:
  - OB 82 (Diagnosealarm)
- Die Ausgänge auf Baugruppen sind gesperrt.
- Das Prozessabbild wird nicht aktualisiert; direkter Peripheriezugriff auf Eingänge ist aber möglich.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 1364)

Anlauf-OBs (Seite 1432)

Neustart (Warmstart) (Seite 1367)

Neustart (Warmstart)

Funktion

Beim Neustart (Warmstart) werden alle nicht remanenten Merker gelöscht und nicht remanente DB-Inhalte auf Startwerte aus dem Ladespeicher zurückgesetzt. Remanente Merker und remanente DB-Inhalte bleiben erhalten.

Die Programmbearbeitung beginnt mit dem ersten Anlauf-OB.

Auslösen von Neustart (Warmstart)

In folgenden Fällen können Sie über einen Menübefehl vom PG aus einen "Neustart (Warmstart)" durchführen:

- Die CPU ist im Betriebszustand "STOP".
- Nach dem Urlöschen
- Nach dem Laden eines konsistenten Programms und einer konsistenten Hardware-Konfiguration im Betriebszustand "STOP" der CPU.

Bei "NETZ-EIN" wird ein "Neustart (Warmstart)" ausgelöst, wenn Sie das Anlaufverhalten folgendermaßen parametriert haben:

- Anlaufart "Warmstart - RUN" (unabhängig vom Betriebszustand der CPU vor NETZ-AUS).
- "Warmstart - Betriebsart vor NETZ-AUS" (abhängig vom Betriebszustand der CPU vor NETZ-AUS. Die CPU muss vorher im RUN gewesen sein.)

Siehe auch

Remanente Speicherbereiche (Seite 1376)

Anlauftätigkeiten

Übersicht

Die folgende Tabelle zeigt, welche Tätigkeiten die CPU beim Anlauf durchführt:

Tätigkeiten in der Bearbeitungsreihenfolge	bei Neustart (Warmstart)
Nicht remanente Merker löschen	ja
Alle Merker löschen	nein
Prozessabbild der Ausgänge löschen	ja
Bearbeitung der Anlauf-OBs	ja
Prozessabbild der Eingänge aktualisieren	ja
Ausgänge freigeben nach Betriebszustandsübergang in "RUN"	ja

Ablauf

Das folgende Bild zeigt die Tätigkeiten der CPU in den Betriebszuständen "STOP", "ANLAUF" und "RUN".

Den Zustand der Peripherieausgänge im ersten Zyklus des Anwenderprogramms können Sie durch folgende Maßnahmen festlegen:

- Parametrierbare Ausgabebaugruppen verwenden, um Ersatzwerte ausgeben zu können oder um den letzten Wert zu halten.
- Ausgänge in Anlauf-OBs vorbesetzen.

Während des Anlaufs werden alle Alarm-Ereignisse in eine Warteschlange gestellt, um sie dann im Betriebszustand RUN abzuarbeiten. Im Betriebszustand RUN können zu jeder Zeit Prozessalarme verarbeitet werden.

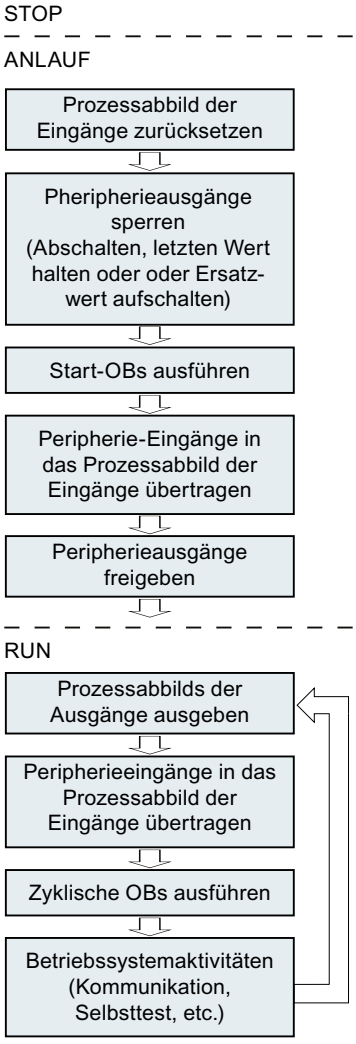

Besonderheiten beim Anlauf

Verhalten bei Sollausbau ungleich Istausbau

Der Sollausbau wird repräsentiert durch die in die CPU geladene projektierte Konfiguration. Der Istausbau ist der tatsächliche Ausbau des Automatisierungssystems.

Wenn Sollausbau und Istausbau voneinander abweichen, geht die CPU trotzdem zunächst in RUN.

Abbrechen eines Anlaufs

Wenn während des Anlaufs Fehler auftreten, wird der Anlauf abgebrochen und die CPU bleibt in "STOP".

Ein Anlauf wird unter folgenden Bedingungen nicht durchgeführt oder abgebrochen:

- Wenn eine ungültige SD-Karte gesteckt ist.
- Wenn keine Hardware-Konfiguration geladen wurde.

Siehe auch

Übersicht über die CPU-Eigenschaften (Seite 1390)

Betriebszustand "RUN"

Funktion

Im Betriebszustand "RUN" erfolgt die zyklische, zeit- und alarmgesteuerte Programmbearbeitung:

- Das Prozessabbild der Ausgänge wird ausgegeben.
- Das Prozessabbild der Eingänge wird gelesen.
- Das Anwenderprogramm wird abgearbeitet.

Der aktive Austausch von Daten zwischen S7-1200 CPUs über Open User Communication ist nur im Betriebszustand "RUN" möglich.

Abarbeiten des Anwenderprogramms

Nachdem die CPU die Eingänge gelesen hat, wird das zyklische Programm beginnend mit der ersten Anweisung bis zur letzten Anweisung abgearbeitet.

Wenn Sie eine Mindestzykluszeit parametriert haben, dann beendet die CPU den Zyklus erst nach Ablauf der Mindestzykluszeit, auch wenn das Anwenderprogramm in kürzerer Zeit durchlaufen wurde.

Um sicherzustellen, dass das zyklische Programm in einer festgelegten Zeit durchlaufen wird, ist eine Zyklusüberwachungszeit eingestellt, die Sie nach Ihren Anforderungen anpassen können. Wenn das zyklische Programm nicht innerhalb dieser Zeit durchlaufen ist, reagiert das System mit einem Zeitfehler.

Weitere Ereignisse wie z. B. Prozessalarme oder Diagnosealarme können den zyklischen Programmfluss unterbrechen und die Zykluszeit verlängern.

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 1364)

Ereignisse und OBs (Seite 1380)

Betriebszustand "STOP"

Funktion

Im Betriebszustand "STOP" wird das Anwenderprogramm nicht bearbeitet. Alle Ausgänge werden deaktiviert bzw. reagieren wie parametriert: sie liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.

Die CPU prüft folgende Punkte:

- Hardware, z. B. ob alle Baugruppen verfügbar sind
- Ob für die CPU die Standardeinstellungen gelten oder Parametersätze vorliegen
- Ob die Randbedingungen für das programmierte Anlaufverhalten stimmen

Siehe auch

Grundlagen zu den Betriebszuständen von S7-CPUs (Seite 1364)

Grundlagen zum Urlöschen

Funktion

Das Urlöschen der CPU ist nur im Betriebszustand STOP möglich.

Beim Urlöschen wird die CPU in einen so genannten "Anfangszustand" versetzt. Das bedeutet:

- Eine bestehende Online-Verbindung zwischen Ihrem PG/PC und der CPU wird abgebaut.
- Der Arbeitsspeicher-Inhalt sowie remanente und nichtremanente Daten werden gelöscht.
- Der Diagnosepuffer, die Uhrzeit, die IP-Adresse, die Hardware-Konfiguration und laufende Force-Aufträge bleiben erhalten.
- Anschließend wird der Ladespeicher (Code- und Datenbausteine) in den Arbeitsspeicher kopiert. Das hat zur Folge, dass die Datenbausteine keine Aktualwerte mehr haben, sondern ihre projektierten Startwerte.

Speicherbereiche

Wissenswertes zu SIMATIC Memory Cards

Funktion der SIMATIC Memory Card

Die SIMATIC Memory Card für eine S7-1200 ist eine mit dem Windows Betriebssystem kompatible und von Siemens vorformatierte SD-Speicherkarte für das Anwenderprogramm der CPU.

ACHTUNG
Formatieren der SIMATIC Memory Card Sie dürfen nur Dateien und Ordner löschen. Wenn Sie die SIMATIC Memory Card mit Windows-Mitteln formatieren, z. B. mit einem handelsüblichen Kartenleser, machen Sie die SIMATIC Memory Card als Speichermedium für eine S7-CPU unbrauchbar.

Einstellen des Kartentyps

Sie können die SIMATIC Memory Card als Übertragungskarte, als Programmkarte oder als Firmware-Update-Karte verwenden.

Zur Einstellung des Kartentyps stecken Sie die SIMATIC Memory Card in den Kartenleser des Programmiergeräts und wählen den Ordner "Card Reader/USB-Speicher" in der

Projektnavigation. In den Eigenschaften der markierten Memory Card bestimmen Sie den Kartentyp:

- Programm
  Bei der Verwendung als Programmkarte können Sie das Anwenderprogramm auf die SIMATIC Memory Card laden. Dadurch wird der interne Ladespeicher des Geräts durch die SIMATIC Memory Card ersetzt und der interne Ladespeicher wird gelöscht. Das Anwenderprogramm ist in diesem Fall vollständig von der SIMATIC Memory Card lauffähig. Wird die SIMATIC Memory Card mit dem Anwenderprogramm entfernt, ist kein Programm mehr vorhanden.
- Übertragung
  Bei der Verwendung als Übertragungskarte können Sie das Anwenderprogramm von der SIMATIC Memory Card in den internen Ladespeicher der CPU übertragen. Die SIMATIC Memory Card können Sie danach entfernen.
- Firmware Card
  Auf einer SIMATIC Memory Card kann Firmware für S7-1200 Baugruppen gespeichert werden. Daher ist es möglich mit Hilfe einer speziell vorbereiteten SIMATIC Memory Card ein Firmware Update durchzuführen. Ebenso kann eine Sicherungskopie einer Baugruppen-Firmware auf der SIMATIC Memory Card gespeichert werden.

Objekte aus dem Projekt auf SIMATIC Memory Card übertragen

Wenn die SIMATIC Memory Card im PG bzw. im externen Kartenleser steckt, können Sie folgende Objekte aus der Projektnavigation auf die SIMATIC Memory Card übertragen:

- Einzelne Bausteine (Mehrfachselektion möglich)
  In diesem Fall wird ein konsistentes Übertragen angeboten, d. h. die Abhängigkeiten der Bausteine untereinander durch Bausteinaufrufe wird berücksichtigt.
- PLC
  In diesem Fall werden alle ablaufrelevanten Objekte, u. a. Bausteine und die Hardware-Konfiguration auf die SIMATIC Memory Card übertragen - wie beim Laden.

Um die Übertragung durchzuführen, können Sie die Objekte per Drag&Drop übertragen oder Sie verwenden den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben" im Menü "Projekt".

Objekte von der SIMATIC Memory Card in das Projekt übertragen

Einzelne Bausteine (Mehrfachselektion möglich) übertragen Sie per Drag&Drop in das Projekt. Eine Hardware-Konfiguration ist nicht von der SIMATIC Memory Card ins Projekt übertragbar.

Firmware aktualisieren mit einer SIMATIC Memory Card

Die neuesten Firmware-Dateien erhalten Sie aus dem Internet über die Seiten des Service & Support:

http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/de/34143537)

Speichern Sie die Firmware-Dateien auf der Festplatte und stecken Sie die SIMATIC Memory Card in den Card Reader Ihres Programmiergeräts.

Um die Datei auf der SIMATIC Memory Card zu speichern, markieren Sie in der Projektnavigation die SIMATIC Memory Card im Ordner "Card Reader/USB-Speicher". Wählen Sie das Kontextmenü "Card Reader/USB-Speicher > Firmware-Update Memory Card erstellen".

Folgen Sie anschließend den Anweisungen im Service & Support-Portal zur Durchführung des Firmware-Updates mit ihrer CPU.

Durch die Aktualisierung der Firmware erhält die CPU einen neuen Firmwarestand. Wenn Sie die CPU im Projekt verwendet haben, müssen Sie durch einen Gerätetausch die bereits projektierte CPU durch die CPU mit dem neuen Firmwarestand offline nachziehen und das Programm bzw. die Konfiguration anpassen und anschließend laden.

Siehe auch

Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen (Seite 1456)

Ladespeicher

Funktion

Jede CPU besitzt einen internen Ladespeicher. Die Größe dieses internen Ladespeichers ist von der jeweiligen CPU abhängig.

Dieser interne Ladespeicher kann extern durch den Einsatz von Memory Cards ersetzt werden. Ohne gesteckte Memory Card verwendet die CPU den internen Ladespeicher, mit gesteckter Memory Card verwendet die CPU die Memory Card als Ladespeicher.

Kopieren vom internen Ladespeicher in den externen Ladespeicher

Wenn Sie in den Schutzeigenschaften der Gerätekonfiguration der CPU die Einstellung "Kopieren vom internen Ladespeicher in den externen Ladespeicher deaktivieren" nicht auswählen, ermittelt die CPU, welche Art von Memory Card Sie stecken:

- Leere Memory Card: Eine leere Memory Card hat keine Auftragsdatei (S7_JOB.S7S). Wenn Sie eine leere Memory Card stecken, fügt die CPU eine Programmauftragsdatei hinzu. Sie kopiert dann den internen Ladespeicher in den externen Ladespeicher (die Programmdatei auf der Memory Card) und löscht den internen Ladespeicher.
- Leere Programmkarte: Eine leere Programmkarte enthält eine Programmauftragsdatei, die leer ist. In diesem Fall kopiert die CPU den internen Ladespeicher in den externen Ladespeicher (die Programmdatei auf der Memory Card) und löscht den internen Ladespeicher.

Wenn Sie in den Schutzeigenschaften der Gerätekonfiguration der CPU die Einstellung "Kopieren vom internen Ladespeicher in den externen Ladespeicher deaktivieren" ausgewählt haben, verhält sich die CPU wie folgt:

- Leere Memory Card: Eine leere Memory Card hat keine Auftragsdatei (S7_JOB.S7S). Wenn Sie eine leere Memory Card stecken, führt die CPU nichts durch. Sie erstellt keine Programmauftragsdatei und sie kopiert nicht den internen Ladespeicher in den externen Ladespeicher (die Programmdatei auf der Memory Card). Sie löscht nicht den internen Ladespeicher.
- Leere Programmkarte: Eine leere Programmkarte enthält eine Programmauftragsdatei, die leer ist. In diesem Fall führt die CPU keine Aktion durch. Sie kopiert nicht den internen Ladespeicher in den externen Ladespeicher (die Programmdatei auf der Memory Card). Sie löscht nicht den internen Ladespeicher.

Wenn Sie eine Programmkarte, Übertragungskarte oder eine Karte mit einer Firmware-Aktualisierung in die CPU stecken, wirkt sich die Konfigurationseinstellung "Kopieren vom internen Ladespeicher in den externen Ladespeicher deaktivieren" nicht darauf aus, wie die CPU die Memory Card auswertet.

Arbeitsspeicher

Funktion

Der Arbeitsspeicher ist ein nicht remanenter Speicherbereich und nimmt die für den Programmablauf relevanten Teile des Anwenderprogramms auf. Das Anwenderprogramm wird ausschließlich im Arbeits- und Systemspeicher bearbeitet.

Systemspeicher

Systemspeicherbereiche

Funktion

Der Systemspeicher enthält die Speicherelemente, die jede CPU dem Anwenderprogramm zur Verfügung stellt, z. B. das Prozessabbild der Ein- und Ausgänge und Merker.

Durch Verwendung der entsprechenden Operationen adressieren Sie in Ihrem Programm die Daten direkt in den jeweiligen Operandenbereich.

Die folgende Tabelle zeigt die Operandenbereiche des Systemspeichers:

Operandenbereich	Beschreibung	Zugriff über Einheiten der folgenden Größe:	S7- Notation
Prozessabbild der Ausgänge	Zu Beginn des Zyklus schreibt die CPU die Werte aus dem Prozessabbild der Ausgänge in die Ausgabebaugruppen.	Ausgang (Bit)	A
		Ausgangsbyte	AB
		Ausgangswort	AW
		Ausgangsdoppelwort	AD

Operandenbereich	Beschreibung	Zugriff über Einheiten der folgenden Größe:	S7- Notation
Prozessabbild der Eingänge	Während des Zyklus liest die CPU die Eingänge aus den Eingabebaugruppen und speichert die Werte in das Prozessabbild der Eingänge.	Eingang (Bit)	E
		Eingangsbyte	EB
		Eingangswort	EW
		Eingangsdoppelwort	ED
Merker	Dieser Bereich stellt Speicherplatz für im Programm errechnete Zwischenergebnisse zur Verfügung.	Merker (Bit)	M
		Merkerbyte	MB
		Merkerwort	MW
		Merkerdoppelwort	MD
Datenbaustein	Datenbausteine speichern Informationen für das Programm. Sie können entweder so definiert sein, dass alle Codebausteine auf sie zugreifen können (globale DBs), oder sie sind einem bestimmten FB oder SFB zugeordnet (Instanz-DB). Voraussetzung: Baustein-Attribut "Optimierter Bausteinzugriff" nicht aktiviert.	Datenbit	DBX
		Datenbyte	DBB
		Datenwort	DBW
		Datendoppelwort	DBD
Lokaldaten	Dieser Speicherbereich nimmt die temporären Lokaldaten eines Bausteins für die Dauer der Bearbeitung dieses Bausteins auf. Voraussetzung: Baustein-Attribut "Optimierter Bausteinzugriff" nicht aktiviert. Empfehlung: symbolisch auf Lokaldaten (Temp) zugreifen.	Lokaldatenbit	L
		Lokaldatenbyte	LB
		Lokaldatenwort	LW
		Lokaldatendoppelwort	LD
Peripheriebereich Eingänge	Die Peripheriebereiche der Ein- und Ausgänge erlauben direkten Zugriff auf zentrale und dezentrale Ein- und Ausgabebaugruppen.	Peripherieeingangsbit	<Variable>:P
		Peripherieeingangsbyte	
		Peripherieeingangswort	
		Peripherieeingangsdoppelwort	
Peripheriebereich Ausgänge		Peripherieausgangsbit	
		Peripherieausgangsbyte	
		Peripherieausgangswort	
		Peripherieausgangsdoppelwort	

Siehe auch

Diagnosepuffer (Seite 1378)

Grundlagen zu den Prozessabbildern (Seite 1376)

Zugriff auf die Peripherieadressen (Seite 1379)

Remanente Speicherbereiche

Remanente Speicherbereiche

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie bestimmte Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Ein remanenter Speicherbereich ist ein Bereich, dessen Inhalt bei einem Neustart (Warmstart), d. h. nach Ausschalten der Versorgungsspannung und nach Einschalten bei einem Übergang von STOP nach RUN erhalten bleibt.

Als remanent können Sie die folgenden Daten definieren:

- Merker: Die genaue Breite des remanenten Speicherbereichs können Sie für Merker in der PLC-Variablentabelle oder im Belegungsplan festlegen.
- Variablen eines Funktionsbausteins (FB): In der Schnittstelle eines FBs können Sie einzelne Variablen als remanent definieren, wenn für diesen Baustein der optimierte Bausteinzugriff aktiviert ist. Wenn der optimierte Bausteinzugriff für einen FB nicht aktiviert ist, können Remanenzeinstellungen nur im zugeordneten Instanz-Datenbaustein vorgenommen werden.
- Variablen eines globalen Datenbausteins: In einem globalen Datenbaustein können Sie abhängig von der Einstellung für den Zugriff entweder einzelne oder alle Variablen des Bausteins als remanent definieren:
  - Baustein mit optimiertem Zugriff: Remanenz ist für jede einzelne Variable einstellbar.
  - Baustein mit Standardzugriff: Remanenzeinstellung gilt für alle Variablen des DBs; entweder sind alle Variablen remanent oder keine Variable ist remanent.

Siehe auch

Neustart (Warmstart) (Seite 1367)

Prozessabbild der Ein- und Ausgänge

Grundlagen zu den Prozessabbildern

Funktion

Werden im Anwenderprogramm die Operandenbereiche Eingänge (E) und Ausgänge (A) angesprochen, dann werden nicht die Signalzustände auf den digitalen Signalbaugruppen abgefragt oder geändert, sondern ein Speicherbereich im Systemspeicher der CPU. Dieser Speicherbereich wird als Prozessabbild bezeichnet.

Vorteile des Prozessabbilds

Der Zugriff auf das Prozessabbild hat gegenüber dem direkten Zugriff auf die Ein- und Ausgabebaugruppen den Vorteil, dass der CPU für die Dauer der zyklischen Programmbearbeitung ein konsistentes Abbild der Prozesssignale zur Verfügung steht. Wenn sich während der Programmbearbeitung ein Signalzustand auf einer Eingabebaugruppe ändert, bleibt der Signalzustand im Prozessabbild bis zur Prozessabbildaktualisierung im nächsten Zyklus erhalten. Dadurch können Sie ein Eingangssignal in Ihrem Programm mehrmals abfragen und erhalten immer eine konsistente Eingangsinformation.

Außerdem benötigt der Zugriff auf das Prozessabbild wesentlich weniger Zeit als der direkte Zugriff auf die Signalbaugruppen, weil sich das Prozessabbild im internen Speicher der CPU befindet.

Aktualisieren der Prozessabbilder

Ablauf

Die Speicherbereiche der Prozessabbilder werden vom Betriebssystem zyklisch aktualisiert, wenn nicht durch Konfiguration etwas anderes festgelegt wurde. Die Aktualisierung der Prozessabbilder der Ein- und Ausgänge erfolgt in dieser Reihenfolge:

1. Die internen Aufgaben des Betriebssystems werden ausgeführt.
2. Das Prozessabbild aller Ausgänge (PAA) wird auf die Ausgänge der Baugruppen geschrieben.
3. Der Zustand aller Eingänge wird ins Prozessabbild der Eingänge (PAE) gelesen.
4. Das Anwenderprogramm mit allen darin aufgerufenen Bausteinen wird abgearbeitet.

Das Betriebssystem steuert selbstständig das Schreiben des Prozessabbilds der Ausgänge auf die Ausgänge der Baugruppen und das Einlesen des Prozessabbilds der Eingänge.

Besonderheiten

Sie haben die Möglichkeit, über einen direkten Peripheriezugriff direkt auf Eingänge bzw. Ausgänge zuzugreifen.

- Wenn eine Anweisung direkt auf einen Ausgang zugreift und die Ausgangsadresse liegt im Prozessabbild der Ausgänge, dann wird das Prozessabbild des betreffenden Ausgangs nachgeführt.
- Wenn eine Anweisung direkt auf einen Ausgang zugreift und die Ausgangsadresse liegt **nicht** im Prozessabbild der Ausgänge, dann wird das Prozessabbild des betreffenden Ausgangs **nicht** nachgeführt.

Beispiel für einen normalen Zugriff auf die Peripherie über das Prozessabbild

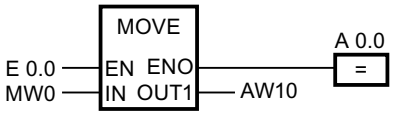

Aktualisiere AW10 im Peripheriebereich Ausgänge mit dem Wert von MW0.

Peripheriezugriffsfehler (PZF) bei Prozessabbild-Aktualisierung

Bei einem Fehler während der Prozessabbild-Aktualisierung (Peripheriezugriffsfehler) reagiert die CPU mit der voreingestellten Systemreaktion "Ignorieren".

Siehe auch

Anfangsadresse einer Baugruppe (Seite 1378)

Zugriff auf die Peripherieadressen (Seite 1379)

Anlauftätigkeiten (Seite 1367)

Diagnosepuffer

Funktion

Der Diagnosepuffer ist Teil des Systemspeichers der CPU. Er enthält die von der CPU und diagnosefähigen Baugruppen erkannten Fehler. Dazu gehören u. a. die folgenden Ereignisse:

- Jeder Betriebszustandswechsel der CPU (z. B. NETZEIN, Übergang in den Betriebszustand STOP, Übergang in den Betriebszustand RUN)
- Jeder Diagnosealarm

Für die S7-1200-CPU hat der Diagnosepuffer eine Kapazität von 50 Einträgen, wovon bei einem NETZ-AUS-NETZ-EIN-Übergang die letzten (jüngsten) 10 Einträge erhalten bleiben.

Die Einträge können nur durch Rücksetzen der CPU auf Werkseinstellungen gelöscht werden.

Mittels der Online- und Diagnosesicht können Sie den Inhalt des Diagnosepuffers auslesen.

Siehe auch

Grundlagen zum Diagnosepuffer (Seite 3527)

Peripheriedatenbereich

Anfangsadresse einer Baugruppe

Definition

Die Anfangsadresse ist die niedrigste Byteadresse einer Baugruppe. Sie stellt die Anfangsadresse des Nutzdatenbereichs der Baugruppe dar.

Baugruppen-Anfangsadressen konfigurieren

Die Zuordnung zwischen Adressen, die im Anwenderprogramm verwendet werden, und den Baugruppen erfolgt durch die Konfiguration der Baugruppen.

Die beim Platzieren der Baugruppen automatisch vergebenen Anfangsadressen können Sie in den Eigenschaften der Baugruppe ändern (Gruppe "E/A-Adressen...").

Sie haben außerdem Möglichkeit einzustellen, ob die Adressen im Prozessabbild liegen oder nicht.

Zugriff auf die Peripherieadressen

Peripherieadressen

Wenn Sie eine Baugruppe in der Gerätesicht stecken, liegen deren Nutzdaten im Prozessabbild der S7-1200 CPU (Voreinstellung). Die CPU übernimmt automatisch den Datenaustausch zwischen Baugruppe und Prozessabbild-Bereich beim Aktualisieren der Prozessabbilder.

Wenn das Programm direkt auf die Baugruppe und nicht über das Prozessabbild zugreifen soll, dann müssen Sie die Peripherieadresse durch ":P" ergänzen.

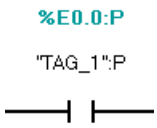

Das ist der Fall z. B. bei einem zeitkritischen Programm, in dem die Ausgänge noch im selben Zyklus beeinflusst werden sollen.

Grundlagen zur Programmbearbeitung

Englische Bezeichnung der OB-Typen

Englische Bezeichnung der OB-Typen

Beim Anlegen eines OB im Dialog "Neuen Baustein hinzufügen" können die OB-Typen aus technischen Gründen nur in Englisch angezeigt werden – unabhängig von der eingestellten Oberflächensprache. Um Ihnen die Handhabung zu erleichtern, werden in der folgenden Tabelle den englischen Begriffen die Bezeichnungen in der eingestellten Oberflächensprache gegenübergestellt.

Englische Bezeichnung des OB-Typs im Dialog "Neuen Baustein hinzufügen"	Bezeichnung im Informationssystem
Program cycle	Zyklus-OB, OB für zyklische Programmbearbeitung, Zyklisches Programm
Startup	Anlauf-OB
Time delay interrupt	Verzögerungsalarm-OB
Cyclic interrupt	Weckalarm-OB
Hardware interrupt	Prozessalarm-OB

Englische Bezeichnung des OB-Typs im Dialog "Neuen Baustein hinzufügen"	Bezeichnung im Informationssystem
Time error interrupt	Zeitfehler-OB
Diagnostic error interrupt	Diagnosealarm-OB
Pull or plug of modules	Ziehen/Stecken-OB
Rack or station failure	Baugruppenträgerausfall-OB
Time of day	Uhrzeitalarm-OB
Status	Statusalarm-OB
Update	Update-Alarm-OB
Profile	OB für hersteller- bzw. profilspezifischen Alarm
MC-Interpolator	MC-Interpolator-OB
MC-Servo	MC-Servo-OB
MC-PreServo	MC-PreServo-OB
MC-PostServo	MC-PostServo-OB

Ereignisse und OBs

Ereignisse und OBs

Das Betriebssystem der S7-1200-CPUs basiert auf Ereignissen. Es sind zwei Arten von Ereignissen zu unterscheiden:

- Ereignisse, die einen OB starten können
- Ereignisse, die keinen OB starten können

Ein Ereignis, das einen OB starten kann, hat nach seinem Auftreten diese Reaktion zur Folge:

- Falls Sie dem Ereignis einen OB zugeordnet haben, wird dieser OB aufgerufen. Wenn momentan ein Aufruf dieses OB nicht möglich ist, wird das Ereignis in eine seiner Priorität entsprechende Warteschlange eingereiht.
- Falls Sie dem Ereignis keinen OB zugeordnet haben, wird die voreingestellte Systemreaktion durchgeführt.

Ein Ereignis, das keinen OB starten kann, hat nach seinem Auftreten die für die zugehörige Ereignisklasse voreingestellte Systemreaktion zur Folge.

Der Ablauf des Anwenderprogramms basiert damit auf Ereignissen, der Zuordnung von OBs zu den Ereignissen und auf dem Code, der sich innerhalb der OBs befindet oder von dort aus aufgerufen wird.

Die folgende Tabelle gibt einen Überblick über diejenigen Ereignisse, die einen OB starten können, samt zugehörigen Ereignisklassen und OBs. Sie ist nach der voreingestellten OB-Priorität sortiert. Dabei entspricht 1 der niedrigsten Priorität.

Ereignisklasse	OB-Nr.	OB-Anzahl	Startereignis	OB-Priorität (Voreinstellung)
Zyklisches Programm	1, >= 123	>= 1	Anlaufende oder Ende des letzten Zyklus-OB	1
Anlauf	100, >= 123	>=0	STOP-RUN-Übergang	1

Ereignisklasse	OB-Nr.	OB-Anzahl	Startereignis	OB-Priorität (Voreinstellung)
Uhrzeitalarm	>= 10	Max. 2	Startzeitpunkt wurde erreicht	2
Verzögerungsalarm	>= 20	Max. 4	Verzögerungszeit abgelaufen	• 3 • ab Firmware-Stand V4.0: 3 bei OB20, 4 bei OB21, 5 bei OB22, 6 bei OB23, 3 bei OB123 bis 32767
Weckalarm	>= 30		Äquidistanter Zeitabschnitt abgelaufen	• 8 • ab Firmware-Stand V4.0: 8 bei OB30, 9 bei OB31, 10 bei OB32, 11 bei OB33, 12 bei OB34, 13 bei OB35, 14 bei OB36, 16 bei OB37, 17 bei OB38, 7 bei OB123 bis 32767
Prozessalarm	>= 40	Max. 50 (mehr verwendbar durch DETACH und ATTACH)	• Steigende Flanke (max. 16) • Fallende Flanke (max. 16)	18
			• HSC: Zählwert= Referenzwert (max. 6) • HSC: Zählrichtung geändert (max. 6) • HSC: Externes Reset (max. 6)	18
Statusalarm	55	0 oder 1	CPU hat Statusalarm empfangen	4
Update-Alarm	56	0 oder 1	CPU hat Update-Alarm empfangen	4
Hersteller- bzw. profilspezifischer Alarm	57	0 oder 1	CPU hat hersteller- bzw. profilspezifischen Alarm empfangen	4
Diagnosealarm	82	0 oder 1	Baugruppe hat Fehler erkannt	5
Ziehen/Stecken-Alarm	83	0 oder 1	Ziehen/Stecken von Baugruppen der dezentralen Peripherie	6
Baugruppenträgerfehler	86	0 oder 1	Fehler im Ein-/Ausgabesystem der dezentralen Peripherie	6
Zeitfehler	80	0 oder 1	• Zyklusüberwachungszeit überschritten • Aufgerufener OB wird noch ausgeführt • Uhrzeitalarm verpasst • Uhrzeitalarm verpasst während STOP • Warteschlange übergelaufen • Alarmverlust durch zu hohe Alarmlast	22

Die folgende Tabelle beschreibt diejenigen Ereignisse, die keinen OB-Start zur Folge haben, und die jeweilige Reaktion des Betriebssystems. Sie ist nach der Priorität der Ereignisse sortiert.

Ereignisklasse	Ereignis	Ereignis-Priorität	Systemreaktion
Ziehen/Stecken von zentralen Baugruppen	Ziehen/Stecken einer Baugruppe	21	STOP
Peripheriezugriffsfehler bei Prozessabbildaktualisierung	Peripheriezugriffsfehler bei Prozessabbildaktualisierung	22	Ignorieren
Programmierfehler	Programmierfehler in einem Baustein, für den Sie die vom Betriebssystem bereitgestellten Systemreaktionen verwenden (Hinweis: Falls Sie die lokale Fehlerbehandlung aktiviert haben, wird die im Baustein programmierte Fehlerbehandlung wirksam.)	23	RUN
Peripheriezugriffsfehler	Peripheriezugriffsfehler in einem Baustein, für den Sie die vom Betriebssystem bereitgestellten Systemreaktionen verwenden (Hinweis: Falls Sie die lokale Fehlerbehandlung aktiviert haben, wird die im Baustein programmierte Fehlerbehandlung wirksam.)	24	RUN
Zyklusüberwachungszeit 2 mal überschritten	Zyklusüberwachungszeit 2 mal überschritten	27	STOP

Zuordnung zwischen OBs und Ereignissen

Mit Ausnahme des Zyklischen Programms und des Anlaufprogramms kann einem Ereignis nur genau ein OB zugeordnet werden. In manchen Ereignisklassen wie z. B. bei den Prozessalarmen kann ein und derselbe OB aber mehreren Ereignissen zugeordnet sein.

Die Zuordnung zwischen OB und Ereignis erfolgt bei der Konfigurierung der Hardware. Eine einmal getroffene Zuordnung können Sie zur Laufzeit mit den Anweisungen ATTACH und DETACH wieder ändern.

OB-Priorität und Ablaufverhalten

Die S7-1200-CPUs unterstützen die Prioritäten 1 (niedrigste Priorität) bis 27 (höchste Priorität). Ein OB erhält die Priorität seines Startereignisses.

Grundsätzlich gilt, dass OBs prioritätsgesteuert bearbeitet werden: Die OBs mit höchster Priorität werden zuerst bearbeitet. Ereignisse gleicher Priorität werden in der Reihenfolge ihres Auftretens bearbeitet.

Ab dem Firmwarestand V4.0 der S7-1200-CPUs können Sie in der Gerätekonfiguration bei den Eigenschaften der CPU festlegen, ob die OBs unterbrechbar sind oder nicht. Diese Parametrierung wirkt sich auf alle OBs mit Ausnahme der Zyklus-OBs aus; diese sind nämlich immer unterbrechbar.

Für S7-1200-CPUs mit Firmwarestand < V4.0 gilt:

- Jeder OB mit einer Priorität >= 2 unterbricht das zyklische Programm.
- Ein OB der Priorität 2 bis 25 kann von keinem Ereignis der Priorität 2 bis 25 unterbrochen werden. Das gilt auch dann, wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt. Diese Ereignisse werden zu einem späteren Zeitpunkt bearbeitet.
- Ein aufgetretener Zeitfehler (Priorität 26) unterbricht jeden anderen OB.

Für S7-1200-CPUs ab Firmwarestand V4.0 gilt:.

Wenn Sie die OBs als nicht unterbrechbar parametrieren, dann wird ein OB stets zu Ende bearbeitet, und zwar auch dann, wenn während seiner Laufzeit ein Ereignis höherer Priorität auftritt. Das bedeutet im Einzelnen:

- Jeder OB mit einer Priorität >= 2 unterbricht das zyklische Programm.
- Ein OB der Priorität 2 bis 25 kann von keinem Ereignis unterbrochen werden. Das gilt auch dann, wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt, also auch für einen Zeitfehler. Diese Ereignisse werden zu einem späteren Zeitpunkt bearbeitet.

Wenn Sie die OBs als unterbrechbar parametrieren und während der Laufzeit eines OB ein Ereignis höherer Priorität auftritt, dann wird der laufende OB unterbrochen, und der zum aufgetretenen Ereignis gehörige OB wird bearbeitet. Nachdem dieser beendet wurde, wird der unterbrochene OB weiter bearbeitet. Das bedeutet im Einzelnen:

- Jeder OB mit einer Priorität >= 2 unterbricht das zyklische Programm.
- Ein OB der Priorität 2 bis 25 kann von jedem Ereignis, dessen Priorität höher als die des laufenden OB ist, unterbrochen werden. Das gilt auch für einen aufgetretenen Zeitfehler: Ein aufgetretener Zeitfehler (Priorität 26) unterbricht jeden OB.

OB-Startinformation

Manche OBs haben eine Startinformation, manche nicht. Darauf wird bei der Beschreibung des jeweiligen OB näher eingegangen.

Siehe auch

Ereignisbasierte Programmbearbeitung (Seite 1383)

Ereignisbasierte Programmbearbeitung

OB-Priorität und Ablaufverhalten

Die S7-1200-CPUs unterstützen die Prioritäten 1 (niedrigste Priorität) bis 27 (höchste Priorität). Ein OB erhält die Priorität seines Startereignisses.

Ein Alarm-OB kann ausschließlich von einem Zeitfehler-OB unterbrochen werden. Dies gilt auch dann, wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt. Mit Ausnahme des Zeitfehler-OB kann also nur ein einziger Alarm-OB aktiv sein.

Tritt während der Bearbeitung eines Alarm-OB ein weiteres Ereignis auf, wird dieses Ereignis in eine seiner Priorität entsprechende Warteschlange eingereiht. Die Startereignisse einer

Warteschlange werden zu einem späteren Zeitpunkt in der Reihenfolge ihres Auftretens abgearbeitet.

Programmbearbeitung in der CPU

Die zyklischen OBs werden durch Alarm-OBs unterbrochen.

Reaktionen auf Ereignisse, die einen Alarm-OB starten:

- Bei CPUs bis Firmware-Version V3: Alarm-OBs können nur durch Zeitfehler-OBs unterbrochen werden.
- Bei CPUs ab Firmware Version V4: Alarm-OBs können durch Alarm-OBs höherer Priorität unterbrochen werden.

Das folgende Bild zeigt den grundsätzlichen Ablauf für den Fall, dass Alarm-OBs nicht unterbrechbar sind (Verhalten bis Firmware-Version V3):

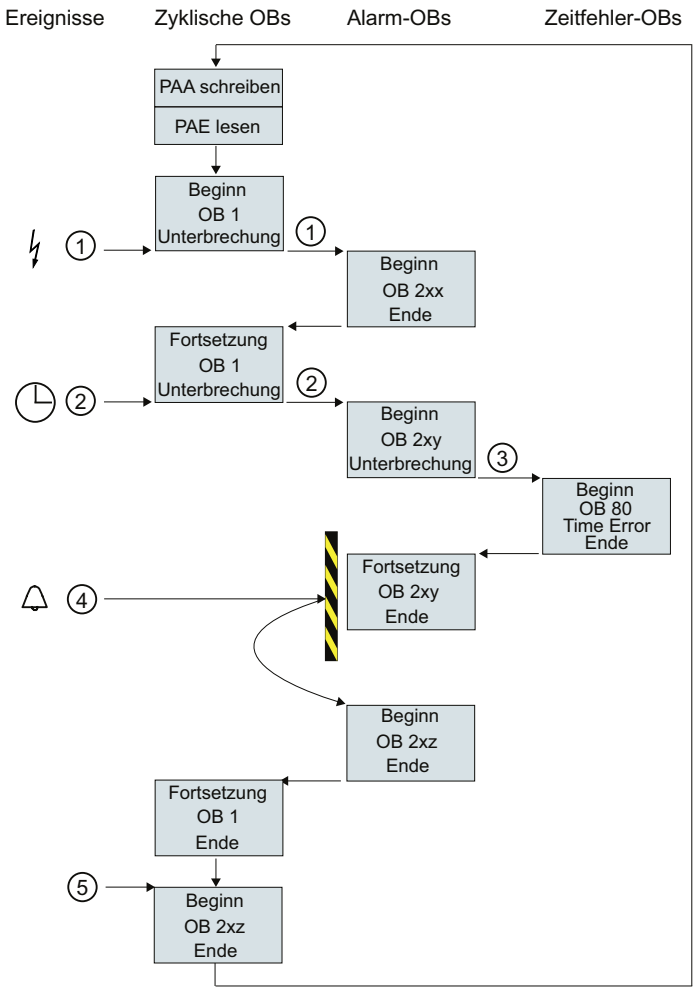

① und ② Ein Ereignis (z. B. ein Prozessalarm) ruft einen zugeordneten OB auf.

Ein aufgerufener OB mit allen unterlagerten Bausteinen wird ohne Unterbrechung abgearbeitet. Weil kein weiteres Ereignis ansteht, das einen OB-Start zur Folge hat, wird nach der Alarmbearbeitung die Bearbeitung des zyklischen OBs fortgesetzt

③ Ein Alarm-OB kann nur durch einen Zeitfehler-OB (OB 80) unterbrochen werden.

④ Während der Alarmbearbeitung tritt ein weiteres alarmauslösendes Ereignis auf.

Reaktion bei CPUs bis einschließlich Firmware-Version V3:

Dieses neue Ereignis wird in eine Warteschlange eingereiht. Erst wenn die Bearbeitung des aktuellen Alarm-OBs abgeschlossen ist, rufen die Ereignisse in der Warteschlange hintereinander die jeweils zugeordneten OBs auf, und zwar nach folgenden Regeln:

- Ereignisse werden nach ihrer Priorität bearbeitet (mit der höchsten Priorität beginnend).
- Ereignisse der gleichen Priorität werden in chronologischer Reihenfolge bearbeitet.

Reaktion bei CPUs ab Firmware-Version V4:

Bei CPUs ab Firmware-Version V4 stellen Sie die Unterbrechbarkeit mit einem CPU-Parameter ein. Voreingestelltes Verhalten: OBs sind unterbrechbar. In diesem Fall gilt: Wenn das neue Ereignis eine höhere Priorität hat als der aktuell laufende OB, dann unterbricht der durch das neue Ereignis gestartete OB den aktuell laufenden OB.

Wenn Sie die Option deaktivieren, sind die Alarm-OBs nicht unterbrechbar.

⑤ Die zyklischen OBs werden hintereinander abgearbeitet.

Hinweise zur Warteschlange

- Für jede Prioritätsklasse (aufzurufende OBs gleicher Priorität) existiert eine eigene Warteschlange. Jede dieser Warteschlangen hat eine vorgegebene Größe.
- Wenn eine Warteschlange gefüllt ist und ein weiteres Ereignis eintritt, dann wird dieses neue Ereignis verworfen, es ist damit verloren. Gleichzeitig wird ein "Zeitfehler-Ereignis" erzeugt. In der Startinfo des Zeitfehler-OBs (OB 80) wird die Information zum fehlerverursachenden OB mitgegeben. Im Zeitfehler-OB kann eine entsprechende Reaktion programmiert werden, z. B. eine Meldung erzeugt werden.

Beispiel für ein Prozessalarm-Ereignis

Am Beispiel eines Prozessalarm-auslösenden Moduls wird die Funktionsweise der ereignisorientierten Bearbeitung des Programms in der S7-1200-CPU beschrieben.

Prozess-Ereignisse und ihre Priorität

Prozess-Ereignisse sind Ereignisse, die von der Peripherie (z. B. einem digitalen Eingang) ausgelöst werden und in der Folge in der S7-1200-CPU den Aufruf eines zugeordneten OBs bewirken. OBs, die einem Prozess-Ereignis zugeordnet sind, werden daher Prozessalarm-OBs genannt.

Beispiele für Prozess-Ereignisse und deren Priorität:

- Prozess-Ereignis "steigende Flanke" bzw. "fallende Flanke" einer alarmauslösenden Baugruppe: Der daraufhin gestartete Prozessalarm-OB hat immer die Priorität 5.
- Prozess-Ereignisse vom schnellen Zähler
  - Zählwert entspricht dem Referenzwert
  - Zählrichtung ändern
  - Externer Reset des schnellen Zählers
    Der daraufhin gestartete Prozessalarm-OB hat immer die Priorität 6.

Das folgende Bild zeigt den zeitlichen Ablauf der Prozessalarm-Bearbeitung: Bei unmittelbar aufeinanderfolgenden Prozessalarm-Ereignissen wird das zweite Prozessalarm-auslösende Ereignis so lange in der Warteschlange zurückgehalten, bis der erste OBx abgearbeitet ist. Erst wenn der OBx abgearbeitet ist, kann das nächste Prozessalarm-auslösende Ereingnis den zugehörigen OBx starten. Weitere Prozessalarm-auslösende Ereignisse werden nach diesem Prinzip ebenfalls in die Warteschlange eingereiht.

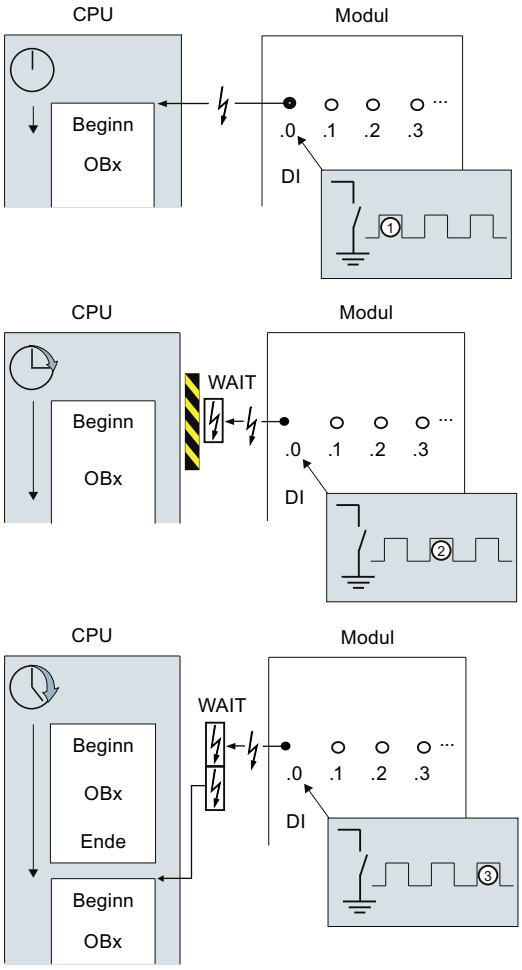

Prozessalarm-Bearbeitung

① Ein Prozessalarm-auslösendes Ereignis, z. B. steigende Flanke am Eingang, ruft einen OB auf, der diesem Ereignis zugeordnet ist.

② Wenn ein neues Prozessalarm-auslösendes Ereignis eintritt, während der OB bearbeitet wird, dann wird dieses Ereignis in eine Warteschlange eingereiht.

③ Das neue Prozessalarm-auslösende Ereignis startet den Prozessalarm-OB, der diesem Ereignis zugeordnet ist.

Zuordnen des alarmauslösenden Ereignisses

Das alarmauslösende Ereignis ordnen Sie in der Gerätesicht, Eigenschaften der Eingänge, einem OB zu.

- Ein alarmauslösendes Ereignis kann genau nur einem OB zugeordnet werden.
- Ein OB kann mehreren alarmauslösenden Ereignissen zugeordnet sein.
  Sie können z. B. sowohl steigende als auch fallende Flanke ein und demselben Alarm-OB zuordnen, um bei jeder Änderung des Eingangssignals unmittelbar auf die selbe Art und Weise reagieren zu können.
- Der gestartete OB kann einen Zyklus-OB bei jeder Anweisung unterbrechen. Konsistenz bei Datenzugriffen ist sichergestellt bis zur Doppelwort-Größe.
- Je nach Baugruppe können verschiedene alarmauslösende Ereignisse parametriert werden, z. B. steigende Flanke sowie fallende Flanke am Eingang.
- Die Zuordnung zwischen alarmauslösendem Ereignis und gestartetem OB legen Sie beim Projektieren des alarmauslösenden Moduls fest. Mit der Anweisung DETACH können Sie aber im gestarteten Prozessalarm-OB diese Zuordnung lösen und mit der Anweisung ATTACH demselben Ereignis einen anderen OB zuweisen. Auf diese Weise können Sie flexibel auf externe Prozessignale reagieren.

Einstellen des Betriebsverhaltens

Eigenschaften der Baugruppen ändern

Standardeinstellungen

Alle parametrierbaren Hardware-Komponenten sind bei Lieferung auf Werte eingestellt, die für Standardanwendungen geeignet sind. Mit diesen Standardwerten können Sie die Hardware-Komponenten ohne weitere Einstellungen einsetzen.

Sie können jedoch das Verhalten und die Eigenschaften der Hardware-Komponenten parametrieren und damit auf Ihre Erfordernisse und die Gegebenheiten Ihrer Anlage einstellen. Parametrierbare Hardware-Komponenten sind z. B. CPUs, Kommunikationsmodule sowie einige analoge und digitale Signalmodule.

Einstellen und Laden der Parameter

Wenn Sie in der Geräte- oder Netzsicht eine Hardware-Komponente selektiert haben, können Sie die Eigenschaften im Inspektorfenster einstellen. Beim Speichern einer Gerätekonfiguration mit ihren Parametern werden Daten erzeugt, die in die CPU geladen werden müssen. Von dort werden diese Daten beim Anlauf in die zugehörigen Baugruppen übertragen.

Eigenschaften der CPUs

Von besonderer Bedeutung für das Systemverhalten sind die Eigenschaften der CPUs. Bei einer CPU können Sie z. B. einstellen:

- Schnittstellen
- Eingänge und Ausgänge
- Schnelle Zähler
- Impulsgeneratoren
- Anlaufverhalten
- Uhrzeit
- Schutzstufe
- Merker für System- und Takt
- Zykluszeit
- Kommunikationslast

Was einstellbar ist und in welchen Wertebereichen, wird durch die Eingabemöglichkeiten vorgegeben. Nicht editierbare Felder sind ausgegraut, bzw. erscheinen nicht im Eigenschaftsfenster.

Voraussetzung

Sie haben die Hardware-Komponenten, deren Eigenschaften Sie ändern wollen, bereits auf einem Baugruppenträger angeordnet.

Vorgehen

Um die Eigenschaften und Parameter der Hardware-Komponenten zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die Hardware-Komponente oder Schnittstelle, die Sie bearbeiten möchten.
2. Bearbeiten Sie die Einstellungen für das gewählte Objekt:
   - In der Geräteübersicht können Sie z. B. Adressen und Namen bearbeiten.
   - Im Inspektorfenster stehen Ihnen zusätzliche Einstellmöglichkeiten zur Verfügung.

Sie müssen Ihre Eingabe nicht bestätigen, die geänderten Werte werden sofort übernommen.

CPU-Eigenschaften

Übersicht über die CPU-Eigenschaften

Übersicht

Die folgende Tabelle zeigt die Eigenschaften der CPU im Überblick:

Gruppe	Eigenschaften	Beschreibung
Allgemein	Projektinformationen	Allgemeine Angabe zur Beschreibung der gesteckten CPU. Bis auf die Steckplatznummer können Sie die Angaben ändern.
	Kataloginformationen	Nur lesbare Informationen aus dem Hardware-Katalog für diese CPU.
	Identification & Maintenance	Zur Hinterlegung einsatzspezifischer Informationen wie dem Namen der Anlage und dem Einbauort.
	Prüfsummen	Zur Prüfung der Identität bzw. Integrität von PLC-Programmen. Bausteine im Bausteinordner und Textlisten werden beim Übersetzen automatisch mit eindeutigen Prüfsummen gekennzeichnet. Sie können leicht feststellen, ob das Programm, das aktuell auf der CPU läuft, dasselbe Programm ist, das Sie vor längerer Zeit geladen haben, oder ob das Programm zwischenzeitlich geändert wurde. Um die Prüfsumme zur Laufzeit im Programm auszulesen, steht die Anweisung "GetChecksum" zur Verfügung. Siehe: Vergleich von PLC-Programmen anhand von Prüfsummen
PROFINET-Schnittstelle	Allgemein	Name und Kommentar für diese PROFINET-Schnittstelle. Der Name ist auf 110 Zeichen begrenzt.
	Ethernet-Adressen	Auswahl, ob die PROFINET-Schnittstelle vernetzt ist. Wenn bereits Subnetze im Projekt angelegt wurden, stehen Sie zur Auswahl in der Klappliste. Ansonsten kann mit der Schaltfläche "Neues Subnetz hinzufügen" ein neues Subnetz angelegt werden. Unter IP-Protokoll stehen Angaben zur IP-Adresse, Subnetzmaske und IP-Router-Verwendung im Subnetz. Wenn ein IP-Router verwendet wird, ist die Angabe der IP-Adresse des IP-Routers notwendig. Siehe: Adress- und Namensvergabe für PROFINET-Geräte
	Erweiterte Optionen	Name, Kommentar und weitere Einstellmöglichkeiten des Ports der Ethernet-Schnittstelle.
	Uhrzeitsynchronisation	Einstellungen zur Uhrzeitsynchronisation im NTP-Uhrzeitformat. Das NTP (network time protocol) ist ein allgemeines Verfahren zur Synchronisation von Systemuhren in lokalen und globalen Netzwerken. Beim NTP-Verfahren sendet die Schnittstelle der CPU in regelmäßigen Zeitabständen Uhrzeitanfragen (im Client-Modus) an NTP-Server im Subnetz (LAN), deren Adressen hier parametriert werden müssen. Anhand der Antworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und synchronisiert. Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg. Die Genauigkeit ist abhängig von der Qualität des verwendeten NTP-Servers.

Gruppe	Eigenschaften	Beschreibung
DI#/DO#	Allgemein	Name und Kommentar für die integrierten Digitaleingänge der CPU.
	Digitaleingänge	Für die Digitaleingänge sind Eingangsverzögerungen einstellbar. Die Eingangsverzögerungen lassen sich gruppenweise einstellen (jeweils für 4 Eingänge). Für jeden Digitaleingang ist die Erkennung einer steigenden und einer fallenden Flanke aktivierbar. Diesem Ereignis kann ein Name und ein Prozessalarm zugeordnet werden. Abhängig von der CPU können an verschiedenen Eingängen Impulsabgriffe aktiviert werden. Wenn der Impulsabgriff aktiviert ist, werden auch die Flanken von Impulsen erkannt, die kürzer sind als die Zykluszeit des Programms.
	Digitalausgänge	Für alle Digitalausgänge ist die Reaktion bei Betriebszustandswechsel von RUN nach STOP einstellbar: Der Zustand kann entweder eingefroren werden (entspricht einem letzten Wert halten) oder Sie stellen einen Ersatzwert ein ("0" oder "1")
	E/A-Adressen	Der Adressraum der Eingabe- und Ausgabeadressen sowie das Prozessabbild werden festgelegt.
AI#	Allgemein	Name und Kommentar für die integrierten Analogeingänge der CPU.
	Analogeingänge	In der Rauschminderung werden durch die eingestellte Integrationszeit die Störfrequenzen der angegebenen Frequenz (in Hz) unterdrückt. Die Kanaladresse, Messart, Spannungsbereich, Glättung und Überlaufdiagnose sind in der Gruppe "Kanal #" einzustellen. Messart und Spannungsbereich sind fest eingestellt auf Spannung, 0 bis 10 V. Durch die Glättung von Analogwerten wird ein stabiles Analogsignal für die Weiterverarbeitung zur Verfügung gestellt. Die Glättung der Analogwerte ist sinnvoll bei langsamen Messwertänderungen, z. B. bei Temperaturmessungen. Die Messwerte werden mittels digitaler Filterung geglättet. Die Glättung wird erreicht, indem die Baugruppe Mittelwerte aus einer festgelegten Anzahl von gewandelten (digitalisierten) Analogwerten bildet. Die parametrierte Stufe (gering, mittel, stark) bestimmt die Anzahl der Analogsignale, die zur Mittelwertbildung herangezogen wird. Wenn die Überlaufdiagnose aktiviert ist, wird bei Überlauf ein Diagnoseereignis erzeugt.
	E/A-Adressen	Der Adressraum der Eingabeadressen sowie das Prozessabbild werden festgelegt.
Schnelle Zähler (HSC)	Schnelle Zähler (HSC)#	Schnelle Zähler werden typischerweise als Antrieb für Zählwerke eingesetzt. Siehe: Schnelle Zähler konfigurieren (Seite 1403)

Gruppe	Eigenschaften	Beschreibung
Impulsgeneratoren (PTO/PWM)	PTO#/PWM#	Ein Impulsgenerator wird aktiviert und kann mit Projektinformationen versorgt werden. Für die Parametrierung eines aktivierten Impulsgenerators wird die Verwendung als PWM (Pulse Width Modulation; "Impulsdauermodulation") oder als PTO (Puls Train Output; "Impulsfolge") eingestellt. Für den PWM müssen Ausgangsquelle, Zeitbasis, Impulsdauerformat, Zykluszeit und anfängliche Impulsdauer festgelegt werden. Als Hardwareausgang ist ein Impulsausgang vorgegeben. Der PWM-Ausgang wird von der Anweisung CTRL_PWM gesteuert, siehe CTRL_PWM. Für den PTO muss die Ausgangsquelle angegeben werden. Als Hardwareausgänge sind ein Impulsausgang und ein Richtungsausgang vorgegeben. Ein PTO wird zusammen mit einem HSC in der Zählart "Bewegungsachse" betrieben und vom Technologieobjekt Motion Control gesteuert (siehe Stichwort "Motion Control S7-1200") In den E/A-/Diagnoseadressen wird die Hardware-Kennung angezeigt und bei Auswahl der Funktion PWM kann der Adressraum der Ausgabeadressen und das Prozessabbild ausgewählt werden.

Gruppe	Eigenschaften	Beschreibung
Anlauf	Anlauf nach NETZ-EIN	Einstellung des Anlaufverhaltens nach einem NETZ-AUS/NETZ-EIN- Übergang. Siehe: Grundlagen zum Betriebszustand "ANLAUF" (Seite 1366)
	Vergleich Sollausbau zu Istausbau	Legt das Anlaufverhalten der CPU fest für den Fall, dass der tatsächliche Aufbau der S7-1200-Station nicht dem konfigurierten Aufbau entspricht: • Anlauf der CPU nur bei Kompatibilität • Anlauf der CPU auch bei Unterschieden In der Einstellung "Anlauf der CPU nur bei Kompatibilität" muss ein Modul auf einem konfigurierten Steckplatz kompatibel zum konfigurierten Modul sein. Kompatibel heißt, dass das vorhandene Modul hinsichtlich seiner Anzahl Ein-/Ausgänge passt und hinsichtlich seiner elektrischen und funktionalen Eigenschaften übereinstimmen muss. Ein kompatibles Modul muss ein konfiguriertes Modul vollständig ersetzen können; es darf mehr können, aber nicht weniger. Beispiel für die Einstellung "Anlauf der CPU nur bei Kompatibilität": Eine CPU kann ein kompatibler Ersatz für eine CPU des selben Typs mit einem niedrigeren Firmwarestand sein. Ein Eingangsmodul mit 32 Digitaleingängen kann ein kompatibler Ersatz für ein Signalmodul mit 16 Digitaleingängen sein. Die CPU läuft an, wenn das konfigurierte Modul oder ein kompatibles Modul steckt. Wenn ein nicht-kompatibles Modul steckt, läuft die CPU nicht an. Beispiel für die Einstellung "Anlauf der CPU auch bei Unterschieden": Anstelle eines konfigurierten Digitaleingabemoduls wird ein Analogausgabemodul gesteckt oder es ist kein Modul auf diesem Steckplatz vorhanden. Obwohl die konfigurierten Eingänge nicht erreichbar sind, läuft die CPU an. Beachten Sie, dass in diesen Fall das Anwenderprogramm nicht ordungsgemäß funktionieren kann und treffen Sie entsprechende Maßnahmen!
	Parametrierungszeit für zentrale und dezentrale Peripherie	Legt einen maximalen Zeitraum (Standard: 60000 ms) fest, in dem die zentrale und dezentrale Peripherie hochlaufen muss. (Die CMs und CPs werden während des Anlaufs von der CPU mit Spannung und Kommunikationsparametern versorgt. Diese Parametrierungszeit gestattet einen Zeitraum, während dessen die an CM oder CP angeschlossenen E/A-Module hochlaufen müssen.) Die CPU geht in RUN, sobald die zentrale und dezentrale Peripherie hochgelaufen und betriebsbereit ist, unabhängig vom Parameter "Parametrierungszeit für zentrale und dezentrale Peripherie". Ist die zentrale und dezentrale Peripherie während dieses Zeitraums nicht hochgelaufen, geht die CPU ohne die zentrale und dezentrale Peripherie in RUN.
Zyklus	Maximale Zykluszeit und Mindestzykluszeit.	Festlegen einer maximalen Zykluszeit oder einer festen Mindestzykluszeit. Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, dann geht die CPU in den Betriebszustand STOP. Siehe: Zykluszeit und Zyklusüberwachungszeit (Seite 1396)

Gruppe	Eigenschaften	Beschreibung
Kommunikationslast	Maximaler Anteil des Zyklus für die Kommunikation (in%)	Steuert in einem gewissen Rahmen die Dauer von Kommunikationsprozessen, die immer auch die Zykluszeit verlängern. Kommunikationsprozesse können z. B. sein: Datenübertragung zu einer anderen CPU oder Laden von Bausteinen (über PC angestoßen). Siehe: Zyklusbelastung durch Kommunikation (Seite 1398)
System- und Taktmerker	Systemmerkerbits und Taktmerkerbits	Systemmerker verwenden Sie für folgende Abfragen: • Ist der aktuelle Zyklus der erste Zyklus nach NETZ-AUS/NETZ-EIN? • Haben sich Diagnosezustandsänderungen gegenüber dem vorangegangenen Zyklus ergeben? • Abfrage auf "1" (high) • Abfrage auf "0" (low) Taktmerker wechseln periodisch in festgelegen Zeitabständen ihren Wert. Siehe: Verwenden von Systemmerkern (Seite 1416) Siehe: Verwenden von Taktmerkern (Seite 1417)

Gruppe	Eigenschaften	Beschreibung
Webserver	-	Aktiviert und konfiguriert die Webserverfunktion. Siehe: Webserver aktivieren (Seite 1325)
	Automatische Aktualisierung	Schickt die angeforderte Webseite mit aktuellen CPU-Daten periodisch zum Webbrowser. Die Periodendauer geben Sie unter "Aktualisierungsintervall" ein. Die automatische Aktualisierung ist nur aktivierbar, wenn der Webserver aktiviert ist. Siehe: Automatische Aktualisierung aktivieren (Seite 1325)
	Benutzerverwaltung	Hier werden Benutzernamen mit ihren Zugriffsrechten eintragen. • Jeder: Vordefinierter Benutzer, dessen Zugriffsrechte Sie über eine Klappliste vergeben. Der Benutzer "Jeder" ist ein Benutzer, der sich beim Web-Zugriff nicht mit Passwort anmeldet. • User#: Editierbarer Name für einen Benutzer, für den Sie Zugriffsrechte und jeweils ein Passwort vergeben. Sobald Sie in eine Zeile unterhalb von "Jeder" klicken, wird ein voreingestellter Name angezeigt, den Sie ändern können. Jedem Benutzer können Sie verschiedene Zugriffsrechte über eine Klappliste zuweisen. Es gibt Zugriffsrechte, die miteinander gekoppelt sind. Beispiel: Wenn Sie das Zugriffsrecht "Firmware-Update durchführen" aktivieren, dann werden automatisch die Zugriffsrechte "Betriebszustand ändern" und "... die Diagnose abfragen" aktiviert. Diese beiden Rechte sind notwendig, um eine Firmware-Update vom Webserver aus durchführen zu können. Siehe: Benutzerliste (Seite 1326)
	Beobachtungstabellen	Ermöglicht das Anlegen von Beobachtungstabellen und legt den Zugriff darauf fest.
	Anwenderdefinierte Webseiten	Ermöglicht den Zugriff auf frei gestaltete Webseiten der CPU mit einem Webbrowser. Siehe: Wissenswertes zu Anwenderseiten (Seite 1330)
	Einstiegsseite	Ermöglicht die Auswahl der Einstiegsseite.
	Übersicht der Schnittstellen	Tabellarische Darstellung aller Baugruppen mit ihren Ethernet-Schnittstellen zur Webserver-Funktionalität für dieses Gerät. Hier können Sie für jede Ethernet-Schnittstelle des Geräts (CPU, CP, CM) den Zugriff auf den Webserver über die jeweilige Schnittstelle erlauben bzw. nicht erlauben.
Oberflächen-Sprachen	-	Wenn Sie Übersetzungen von Projekttexten für den Webserver nutzen wollen, müssen Sie hier den Sprachen für den Webserver jeweils die gewünschte Projektsprache zuweisen (Anzahl ist CPU-abhängig). Sie legen damit die Projektsprachen fest, die beim Zugriff auf den Webserver angezeigt werden. Beispiel: Sie weisen einer Sprache "Deutsch" des Webservers die Projektsprache "Deutsch (Luxemburg)" durch Auswahl aus der Klappliste zu. Damit die Projektsprache in der Klappliste auswählbar ist, muss sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" aktiviert sein.
Uhrzeit	Ortszeit und Sommerzeit	Einstellung der Zeitzone, in der die CPU betrieben wird und Einstellung der Sommerzeit/Winterzeit-Umstellung.

Gruppe	Eigenschaften	Beschreibung
Schutz & Security	Schutz und Passwort für Lese-/Schreibzugriff	Einstellen des Lese-/Schreibschutzes und des Passworts für den Zugriff auf die CPU. Siehe: Einstellmöglichkeiten für die Schutzstufe (FW V1 bis V3) (Seite 1418) Siehe: Einstellmöglichkeiten für den Schutz (FW ab V4) (Seite 1419)
	Verbindungsmechanismen	Aktivieren des Zugriffes über eine PUT/GET-Kommunikation.
	Security-Ereignis	Aktivieren von Sammelmeldungen bei Security-Ereignissen und Festlegen eines Überwachungszeitraums (Intervall).
	Externer Ladespeicher	Deaktiviert das Kopieren vom internen zum externen Ladespeicher (Seite 1373).
Konfigurationssteuerung	-	Aktiviert die Konfigurationssteuerung. Damit wird in einem gewissen Rahmen eine Konfigurationsänderung im Anwenderprogramm ermöglicht. Siehe: Konfigurationssteuerung
Verbindungsressourcen	-	Bietet einen Überblick über die zugewiesenen reservierten und dynamischen Ressourcen für die Verbindungen der CPU. In der Online-Sicht werden auch die aktuell verwendeten Ressourcen angezeigt. Siehe: Verbrauch von Verbindungsressourcen
Adressübersicht	-	Tabellarische Darstellung aller von der CPU verwendeten Adressen für die integrierten Ein/Ausgänge sowie für die gesteckten Module. Adressen, die von keinem Modul verwendet werden, sind als Lücke dargestellt. Die Ansicht lässt sich filtern nach • Eingabeadressen • Ausgabeadressen • Adresslücken

Siehe auch

Prozessalarm-OBs parametrieren (Seite 1454)

Zugriff auf die Peripherieadressen (Seite 1379)

Besonderheiten beim Anlauf (Seite 1369)

Zykluszeit und Zyklusüberwachungszeit

Funktion

Die Zykluszeit ist die Zeit, die das Betriebssystem für die zyklische Programmbearbeitung sowie für alle diesen Zyklus unterbrechende Programmteile benötigt. Die Programmbearbeitung kann unterbrochen werden durch:

- Zeitfehler und 2xMaxCycleTime-Fehler
- Systemtätigkeiten, z. B. Prozessabbildaktualisierung

Die Zykluszeit (TZ) ist daher nicht für jeden Zyklus gleich lang.

Das folgende Bild zeigt ein Beispiel für unterschiedliche Zykluszeiten (TZ1 ≠ TZ2) für S7-1200-CPUs:

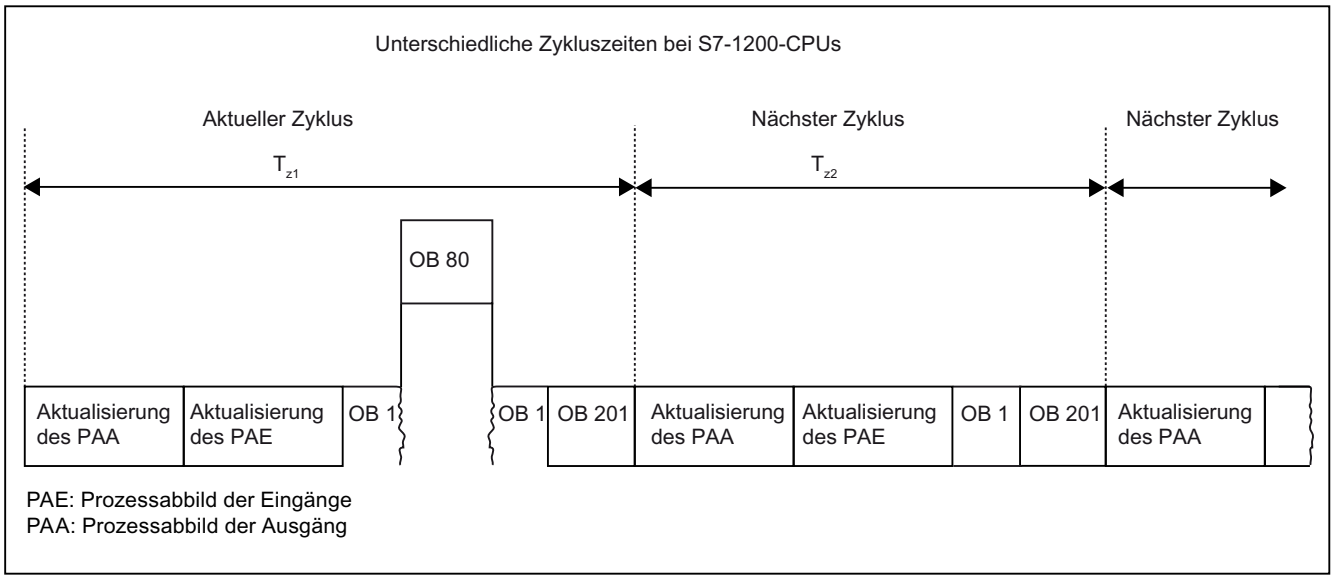

Im aktuellen Zyklus wird der hier verwendete zyklische OB (z. B. OB 1) durch einen Zeitfehler (z. B. OB 80) unterbrochen. Im Anschluss des zyklischen OBs wird der nächste Zyklus-OB 201 abgearbeitet.

Zyklusüberwachungszeit

Das Betriebssystem überwacht die Laufzeit des zyklischen Programms auf eine projektierbare Obergrenze, die maximale Zykluszeit. Sie können durch Aufruf der Anweisung RE_TRIGR an jeder Stelle Ihres Programms diese Zeitüberwachung neu starten.

Wenn das zyklische Programm die Zyklusüberwachungszeit überschreitet, versucht das Betriebssystem den Zeitfehler-OB (OB 80) zu starten. Wenn der OB nicht vorhanden ist, verhält sich die S7-1200-CPU wie folgt:

- CPUs mit Firmware-Stand V1.x bis V3.x: Die CPU bleibt im RUN.
- CPUs ab Firmware-Stand V4.x: Die CPU geht in den Betriebszustand STOP.

Neben der Überwachung der Laufzeit auf Überschreitung der Zyklusüberwachungszeit wird die Einhaltung einer minimalen Zykluszeit garantiert. Das Betriebssystem verzögert dabei den Beginn eines neuen Zyklus so lange, bis die minimale Zykluszeit erreicht ist. Während dieser Wartezeit werden neue Ereignisse und Betriebssystemdienste bearbeitet.

Wenn die Zyklusüberwachungszeit ein zweites Mal überschritten wird, z. B. während der Zeitfehler-OB bearbeitet wird (2xMaxCycleTime-Fehler), geht die CPU in den Betriebszustand STOP.

Siehe auch

Zeitfehler-OB (Seite 1443)

Zyklusbelastung durch Kommunikation

Funktion

Die Zykluszeit der CPU kann sich durch Kommunikationsprozesse verlängern. Zu diesen Kommunikationsprozessen gehören z. B.:

- Datenübertragung zu einer anderen CPU
- Über ein Programmiergerät (PG) angestoßenes Laden von Bausteinen

Die Dauer dieser Kommunikationsprozesse können Sie mit dem CPU-Parameter "Zyklusbelastung durch Kommunikation" in einem gewissen Rahmen steuern.

Neben den Kommunikationsprozessen verlängern auch Testfunktionen die Zykluszeit. Dabei kann über den Parameter "Zyklusbelastung durch Kommunikation" Einfluss auf die Dauer genommen werden.

Wirkungsweise des Parameters

Über den Parameter "Zyklusbelastung durch Kommunikation" geben Sie den Prozentsatz der gesamten CPU-Verarbeitungsleistung an, der den Kommunikationsprozessen zur Verfügung stehen soll. Diese Verarbeitungsleistung wird der Kommunikation ständig von der CPU zugewiesen. Wenn die Kommunikation diese Verarbeitungsleistung nicht benötigt, steht sie der Programmbearbeitung zur Verfügung.

Auswirkung auf die tatsächliche Zykluszeit

Durch die Verwendung des Parameters "Zyklusbelastung durch Kommunikation" verlängert sich die Zykluszeit des zyklischen Organisationsbausteins (z. B. OB 1) um einen Faktor, der sich nach der folgenden Formel berechnen lässt:

$$\frac{100}{100 - \text{“Zyklusbelastung durch Kommunikation“}}$$

Die Formel berücksichtigt dabei den Einfluss von asynchronen Ereignissen wie Prozess- oder Weckalarmen auf die Zykluszeit nicht.

Wenn sich die Zykluszeit durch Kommunikationsprozesse verlängert, können auch mehr asynchrone Ereignisse innerhalb der Zykluszeit des zyklischen Organisationsbausteins auftreten. Dies verlängert den Zyklus zusätzlich. Die Verlängerung ist davon abhängig, wie viele Ereignisse auftreten und wie lange ihre Bearbeitung dauert.

Beispiel 1 – keine zusätzlichen asynchronen Ereignisse:

Bei Einstellung des Parameters "Zyklusbelastung durch Kommunikation" auf 50 % kann die Zykluszeit des zyklischen Organisationsbausteins auf das Doppelte ansteigen.

Beispiel 2 – zusätzliche asynchrone Ereignisse:

Bei einer reinen Zykluszeit von 500 ms kann sich durch eine Kommunikationslast von 50 % eine tatsächliche Zykluszeit von bis zu 1000 ms ergeben, unter der Voraussetzung, dass die CPU immer genügend Kommunikationsaufträge zum Bearbeiten hat. Wenn nun parallel dazu alle 100 ms ein Weckalarm mit 20 ms Bearbeitungszeit abläuft, dann würde sich dieser ohne Kommunikationslast mit insgesamt 5*20 ms = 100 ms auf den Zyklus verlängernd auswirken, d. h. die tatsächliche Zykluszeit wäre 600 ms. Da ein Weckalarm auch die Kommunikation unterbricht, wirkt er sich bei 50 % Kommunikationslast mit 10 * 20 ms auf die Zykluszeit aus, d. h. in diesem Fall beträgt die tatsächliche Zykluszeit nicht 1000 ms, sondern 1200 ms.

Hinweis

Beachten Sie folgende Hinweise:

- Überprüfen Sie die Auswirkungen einer Wertänderung des Parameters "Zyklusbelastung durch Kommunikation" im Anlagenbetrieb.
- Die Kommunikationslast muss beim Einstellen der minimalen Zykluszeit berücksichtigt werden, da es sonst zu Zeitfehlern kommen kann.

Empfehlungen

- Vergrößern Sie den Wert nur dann, wenn die CPU hauptsächlich zu Kommunikationszwecken eingesetzt wird und das Anwenderprogramm zeitunkritisch ist.
- In allen anderen Fällen sollten Sie den Wert nur verringern.

Uhrzeitfunktionen

Grundlagen zu den Uhrzeitfunktionen

Alle S7-1200 CPUs sind mit einer internen Uhr ausgestattet. Die Pufferung unterstützt die Anzeige der korrekten Uhrzeit für eine Unterbrechung der Spannungsversorgung für bis zu 10 Stunden.

Uhrzeitformat

Die Uhr zeigt immer die Uhrzeit mit einer Auflösung von 1 Millisekunde und das Datum mit Wochentag an. Die durch die Sommerzeit bedingte Zeitumstellung wird berücksichtigt.

Stellen und Lesen der Uhrzeit

Uhrzeit stellen und lesen mit Anweisungen

Sie können Uhrzeit und Datum der CPU-Uhr über folgende Anweisungen im Anwenderprogramm stellen, starten und lesen:

- Uhrzeit stellen: "WR_SYS_T"
- Uhrzeit lesen "RD_SYS_T"
- Lokalzeit lesen "RD_LOC_T"
- Zeitzone setzen "SET_TIMEZONE"

Manuelle Einstellung

Das Lesen und Einstellen der Uhrzeit können Sie auch manuell in der Online- und Diagnosesicht unter "Funktionen > Uhrzeit einstellen" durchführen.

Parametrieren der Uhr

Parameter der Uhr

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Uhrzeitsynchronisation über NTP-Server aktivieren
  Aktivieren Sie das Optionskästchen, wenn die interne Uhr mit dem NTP-Synchronisationsverfahren synchronisiert werden soll.
- Netzwerk-Uhrzeitserver
  Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren.
- Aktualisierungsintervall
  Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen.

Siehe auch

NTP-Verfahren - Projektierung (Seite 3217)

Schnelle Zähler

Allgemeines zu schnellen Zählern

Einführung

Schnelle Zähler werden typischerweise als Antrieb für Zählwerke eingesetzt, bei denen eine Welle, die mit einer konstanten Drehzahl läuft, mit einem Winkelschrittgeber versehen ist. Der Winkelschrittgeber sorgt für eine bestimmte Anzahl von Zählwerten pro Umdrehung sowie für einen Rücksetzimpuls einmal pro Umdrehung. Der bzw. die Taktgeber und der Rücksetzimpuls des Winkelschrittgebers liefern die Eingänge für den schnellen Zähler.

Den verschiedenen S7-1200 CPUs stehen eine unterschiedliche Anzahl von schnellen Zählern zu Verfügung:

S7-1200 CPU	Anzahl HSCs	Bezeichnung HSC
CPU 1211C	3 (mit digitalem Signalboard 4)*	HSC1…3 (und HSC5)*
CPU 1212C	4 (mit digitalem Signalboard 5)*	HSC1…4 (und HSC5)*
CPU 1214C CPU 1215C CPU 1217C	6	HSC1…6

* mit DI2/DO2 Signalboard

Funktionsweise

Der erste von mehreren voreingestellten Werten wird in den schnellen Zähler geladen. Die gewünschten Ausgänge werden für die Zeitspanne aktiviert, während der aktuelle Wert des Zählers kleiner als der voreingestellte Wert ist. Der Zähler wird so eingerichtet, dass ein Interrupt auftritt, wenn der aktuelle Wert des Zählers gleich dem voreingestellten Wert ist oder wenn der Zähler zurückgesetzt wird.

Wenn der aktuelle Wert gleich dem voreingestellten Wert ist und es zu einem Interrupt-Ereignis kommt, dann wird ein neuer voreingestellter Wert geladen und der nächste Signalzustand für die Ausgänge gesetzt. Tritt ein Interrupt-Ereignis auf, weil der Zähler zurückgesetzt wird, dann werden der erste voreingestellte Wert und die ersten Signalzustände der Ausgänge gesetzt und der Zyklus wiederholt.

Da die Interrupts in einer sehr viel geringeren Geschwindigkeit auftreten als der schnelle Zähler zählt, kann eine präzise Steuerung der schnellen Operationen mit relativ geringem Einfluss auf den gesamten Zyklus des Automatisierungssystems implementiert werden. Da Sie Interrupts bestimmten Interrupt-Programmen zuordnen können, kann jede neue Voreinstellung in einem getrennten Interrupt-Programm geladen werden, damit so der Zustand einfach gesteuert werden kann.

Hinweis

Sie können auch alle Interrupt-Ereignisse in einem einzigen Interrupt-Programm bearbeiten.

Zählalgorithmen der unterschiedlichen Zähler

Alle Zähler arbeiten auf die gleiche Weise, aber nicht jeder schnelle Zähler unterstützt alle Zählalgorithmen. Es gibt vier grundlegende Zählalgorithmen:

- Einphasenzähler mit interner Richtungssteuerung
- Einphasenzähler mit externer Richtungssteuerung
- Zweiphasenzähler mit 2 Takteingängen
- A/B-Zähler

Jeder schnelle Zähler kann mit oder ohne Rücksetzeingang verwendet werden. Wird der Rücksetzeingang aktiviert, setzt dieser den aktuellen Wert zurück. Der aktuelle Wert bleibt solange zurückgesetzt, bis der Rücksetzeingang deaktiviert wird.

Siehe auch

Schnelle Zähler konfigurieren (Seite 1403)

Abhängigkeiten der Zählerart und Zählereingänge (Seite 1402)

Abhängigkeiten der Zählerart und Zählereingänge

Allgemeines zu Zählerart und Zählereingängen

Den schnellen Zählern können Sie Zählerarten und Zählereingänge und darüber die Funktionen wie Taktgeber, Richtungssteuerung und Rücksetzen zuweisen. Dabei gelten folgende Regeln:

- Ein Eingang kann nicht für zwei verschiedene Funktionen verwendet werden.
- Wird ein Eingang nicht von der aktuellen Zählerart des definierten schnellen Zählers benötigt, kann er für andere Zwecke genutzt werden.

Setzen Sie beispielsweise HSC1 in Zählerart 1 ein, in der die Eingänge E0.0 und E0.3 benötigt werden, können Sie E0.1 für Flankeninterrupts oder für HSC2 verwenden.

Setzen Sie beispielsweise HSC1 und HSC5 ein, werden bei den Zählerarten Zählen und Frequenz immer die Eingänge E0.0 (HSC1) und E1.0 (HSC5) verwendet. Daher stehen dann diese beiden Eingänge beim Betrieb der Zähler für keine anderen Funktionen zur Verfügung.

Zusätzliche Eingänge stehen Ihnen bei Verwendung eines digitalen Signalboards zur Verfügung.

Übersicht über die Abhängigkeiten von Zählerart und Zählereingängen

Zählerart	Beschreibung	Eingänge		
	HSC1	E0.0 (CPU) E4.0 (Signalboard)	E0.1 (CPU) E4.1 (Signalboard)	E0.3 (CPU) E4.3 (Signalboard)
	HSC2	E0.2 (CPU) E4.2 (Signalboard)	E0.3 (CPU) E4.3 (Signalboard)	E0.1 (CPU) E4.1 (Signalboard)
	HSC3*	E0.4 (CPU)	E0.5 (CPU)	E0.7 (CPU)
	HSC4 (nur CPU 1212/14/15/17C)	E0.6 (CPU)	E0.7 (CPU)	E0.5 (CPU)
	HSC5 (nur CPU 1214/15/17C)**	E1.0 (CPU) E4.0 (Signalboard)	E1.1 (CPU) E4.1 (Signalboard)	E1.2 (CPU) E4.3 (Signalboard)
	HSC6 (nur CPU 1214/15/17C)**	E1.3 (CPU)	E1.4 (CPU)	E1.5 (CPU)
Zählen / Frequenz	Einphasenzähler mit interner Richtungssteuerung	Taktgeber	-	-
Zählen				Rücksetzen
Zählen / Frequenz	Einphasenzähler mit externer Richtungssteuerung	Taktgeber	Richtung	-
Zählen				Rücksetzen
Zählen / Frequenz	Zweiphasenzähler mit 2 Takteingängen	Taktgeber vorwärts	Taktgeber rückwärts	-
Zählen				Rücksetzen
Zählen / Frequenz	A/B-Zähler	Taktgeber A	Taktgeber B	-
Zählen				Rücksetzen
Bewegungsachse	Impulsgeneratoren PWM/PTO	HSC1 und HSC2 unterstützen die Zählart Bewegungsachse für die Impulsgeneratoren PTO1 und PTO2: • HSC1 wertet für PTO1 den Ausgange A0.0 für die Anzahl der Impulse aus • HSC2 wertet für PTO2 den Ausgange A0.2 für die Anzahl der Impulse aus Als Ausgang für die Bewegungsrichtung wird A0.1 verwendet.		

* HSC3 ist für CPU 1211 nur ohne Rücksetz-Eingang verwendbar

** HSC5 ist bei Verwendung eines Signalboards DI2/DO2 auch für CPU 1211/12 nutzbar

Siehe auch

Allgemeines zu schnellen Zählern (Seite 1401)

Schnelle Zähler konfigurieren (Seite 1403)

Schnelle Zähler konfigurieren

Voraussetzung

Eine S7-1200 CPU ist in die Hardwarekonfiguration eingefügt.

Vorgehen

Um einen schnellen Zähler zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht eine S7-1200 CPU.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Schnelle Zähler (HSC)" auf den gewünschten schnellen Zähler:
   - CPU 1211C: HSC1 bis HSC3 (mit Signalboard DI2/DO2 auch HSC5)
   - CPU 1212C: HSC1 bis HSC4 (mit Signalboard DI2/DO2 auch HSC5)
   - CPU 1214C / 1215C / 1217C: HSC1 bis HSC6
3. Aktivieren Sie den schnellen Zähler in der Parametergruppe "Allgemein" mit dem entsprechenden Optionskästchen.

Hinweis

Wenn Sie eine CPU 1211C oder CPU 1212C mit Signalboard DI2/DO2 verwenden, können Sie auch den schnellen Zähler HSC5 aktivieren.

Hinweis

Wenn Sie die Impulsgeneratoren aktivieren und als PTO betreiben (zur Ansteuerung von Schrittmotoren), dann werden schnelle Zähler benötigt.

Bei Firmware-Versionen ab V3.0 nützt die CPU dazu interne (zusätzliche) HSCs.

Bei Firmware-Versionen kleiner V3.0 verwendet die CPU HSC1 für PTO1 und HSC2 für PTO2. Die Zähler stehen dann nicht mehr für andere Zählaufgaben zur Verfügung.

Bei "Projektinformation" können Sie einen Namen und einen Kommentar zum Zähler eingeben.

4. Definieren Sie die Funktionsweise des Zählers in der Parametergruppe "Funktion":
   - Zählart: Wählen Sie über die Klappliste eine der folgenden Funktionen des Zählers aus:
     - Zählen (Bestimmung der Anzahl von Impulsen)
     - Periodendauer (Messung der Periodendauer von Impulsen)
     - Frequenz (Messung der Frequenz von Impulsen)
     - Motion Control (Zähler für Bewegungssteuerung verwenden).
   - Betriebsphase: Wählen Sie über die Klappliste aus, wie der Zähler betrieben wird:
     - Einphasig (ein Takteingang fürs Zählen, Vorwärts- und Rückwärtszählen einstellbar über Richtungseingang)
     - Zweiphasig (ein Takteingang fürs Vorwärtszählen, ein Takteingang fürs Rückwärtszählen)
     - A/B-Zähler (zwei Takteingänge mit phasenverschobenen Impulsen, einfacher Zählwert)
     - A/B-Zähler vierfach (zwei Takteingänge mit phasenverschobenen Impulsen, vierfacher Zählwert)
   - Eingangsquelle (nur in Firmware-Versionen kleiner 4.0): Wählen Sie als Eingangsquelle für die Zählimpulse einen CPU-Eingang oder einen Eingang eines Signalboards aus.
   - Zählrichtung wird angegeben von: Wenn als Betriebsphase "Einphasig" eingestellt ist, können Sie wählen, ob das Anwenderprogramm die Zählrichtung vorgibt (Anweisung CTRL_HSC, Parameter DIR und NEW_DIR) oder ein Digitaleingang.
   - Anfängliche Zählrichtung: Wenn bei der Zählrichtung das Anwenderprogramm als Richtungssteuerung eingestellt ist, können Sie die Zählrichtung bei Zählbeginn auswählen.
   - Frequenzmessperiode: Ist als Zählart die Frequenz oder die Periodendauer eingestellt, so können Sie über die Klappliste die Dauer der Frequenzmessperioden auswählen.
5. In der Parametergruppe "Auf Anfangswerte zurücksetzen" legen Sie die Anfangswerte und Rücksetzbedingung des Zählers fest:
   - Anfänglicher Zählerwert: Geben Sie einen Startwert für den Zähler vor.
   - Anfänglicher Referenzwert: Geben Sie einen Referenzwert vor.
     Mit diesem Referenzwert wird der aktuelle Zählerstand verglichen: Wenn der aktuelle Zählerwert gleich dem Referenzwert ist, wird ein Ereignis ausgelöst (wenn unter "Ereigniskonfiguration" aktiviert).

   Sie können hier auch einstellen, ob der Zähler einen Rücksetzeingang verwendet und der Zähler den aktuellen Zählwert bei High- oder Low-Pegel am Rücksetzeingang zurücksetzt.
6. In der Parametergruppe "Ereigniskonfiguration" sind die folgenden Ereignisse aktivierbar:
   - Zählerwert gleich Referenzwert
   - Externes Rücksetzen
   - Richtungswechsel

   Jedem dieser Ereignisse können Sie unter "Prozessalarm" einen Prozessalarm-OB (Hardware interrupt) zuordnen. Oder Sie ordnen jedem Ereignis im Anwenderprogramm (Anweisung "ATTACH") einen Prozessalarm-OB zu.
   Ergebnis: Wenn ein Ereignis ausgelöst wird, führt die CPU den zugeordneten Prozessalarm-OB aus (gemäß der konfigurierten Priorität).
   Die Namen der Ereignisse und Prozessalarme sind veränderbar.

7. In der Parametergruppe "Hardwareeingänge" sind bereits Hardwareeingänge eingetragen. Sie können sie übernehmen. Oder Sie klicken bei "Taktgeber vorwärts", "Taktgeber rückwärts" oder "Rücksetzeingang" auf die Schaltfläche "..." und wählen einen Eingang aus.
   Hinweis: Bei früheren Firmware-Versionen (kleiner V4.0) sind die Hardwareeingänge vorgegeben und nicht änderbar.
8. Im Bereich "E/A-Adressen" sind die folgenden Parameter einstellbar:
   – Anfangsadresse: Die CPU schreibt den Zählerstand in den Adressbereich mit der eingestellten Anfangsadresse (Datentyp DInt).
   – Organisationsbaustein: Zuordnung dieses Adressbereichs zu einem Organisationsbaustein.
   – Prozessabbild: Zuordnung des Adressbereichs zu einem Teilprozessabbild (TPA). Voreingestellt ist die automatische Aktualisierung (TPA0).

Ergebnis

Sie haben nun die Parameter des schnellen Zählers an Ihre Projekterfordernisse angepasst.

Siehe auch

Allgemeines zu schnellen Zählern (Seite 1401)

Abhängigkeiten der Zählerart und Zählereingänge (Seite 1402)

Punkt-zu-Punkt-Kommunikation

Überblick über die Punkt-zu-Punkt-Kommunikation

Die PtP-Kommunikation ist eine Kommunikation über eine serielle Schnittstelle, welche standardisierte UART-Datenübertragung (Universal Asynchronous Receiver Transmitter) verwendet. Die S7-1200 verwendet Kommunikationsmodule mit einer RS232- oder RS485-Schnittstelle, um die PtP-Kommunikation aufzubauen.

Funktionen der Punkt-zu-Punkt-Kommunikation

Die Punkt-zu-Punkt-Kommunikation (PtP) bietet eine große Menge an Anwendungsmöglichkeiten:

- Direktes Senden von Informationen zu einem externen Gerät, wie z. B. einem Drucker oder einem Barcode-Lesegerät
- Empfangen von Informationen von externen Geräten, wie z. B. Barcode-Lesern, RFID-Lesern, Kameras und optische Systeme von Drittherstellern sowie von vielen anderen Geräten.
- Austausch von Informationen mit Drittherstellergeräten, wie z. B. GPS-Geräten, Radio-Modems und vielen mehr

Das Freeport-Protokoll

Die S7-1200 unterstützt das Freeport-Protokoll für zeichenbasierte serielle Kommunikation. Mit Hilfe der Freeport-Kommunikation kann das Datenübertragungsprotokoll vollständig über das Anwenderprogramm konfiguriert werden.

Siemens stellt Bibliotheken mit Freeport-Kommunikations-Funktionen zur Verfügung, die Sie im Anwenderprogramm verwenden können:

- USS Drive Protocol
- Modbus RTU Master Protocol
- Modbus RTU Slave Protocol

Siehe auch

Kommunikations-Port konfigurieren (Seite 1408)

Verwenden von RS232- und RS485-Kommunikationsmodulen

Kommunikationsmodule mit RS232- und RS485-Schnittstellen

In einer S7-1200 CPU können zwei unterschiedliche Kommunikationsmodule verwendet werden:

- RS232-Kommunikationsmodul
- RS485-Kommunikationsmodul

Die Kommunikationsmodule können mit der S7-1200 CPU über den linksseitigen I/O-Kanal verbunden werden. Das Stecken von bis zu drei Modulen ist möglich.

Eigenschaften der Kommunikationsmodule

Die Kommunikationsmodule zeichnen sich durch folgende Eigenschaften aus:

- Unterstützung des Freeport-Protokolls
- Konfiguration über das Anwenderprogramm mit Hilfe von erweiterten Anweisungen und Bibliotheksfunktionen

Kommunikations-Port konfigurieren

Kommunikations-Port konfigurieren

Nachdem Sie ein Kommunikationsmodul mit einer RS232- oder RS485-Schnittstelle gesteckt haben, müssen Sie zunächst die Schnittstellenparameter einstellen. Die Schnittstelle parametrieren Sie entweder in den Eigenschaften der Schnittstelle oder Sie steuern die Schnittstellenparameter über das Anwenderprogramm mit Hilfe der PORT_CFG Anweisung. Die folgende Beschreibung bezieht sich auf die grafische Projektierung.

Hinweis

Wenn Sie die Port-Einstellung über das Anwenderprogramm ändern, werden die Einstellungen der grafischen Projektierung überschrieben.

Beachten Sie weiterhin, dass über das Anwenderprogramm vorgenommene Einstellungen im Falle eines Spannungsverlusts nicht erhalten bleiben.

Voraussetzung

- Ein Kommunikationsmodul ist bereits gesteckt.
- Sie befinden sich in der Gerätesicht.

Vorgehen

Um den Kommunikations-Port zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der grafischen Darstellung innerhalb der Gerätesicht die Schnittstelle. Im Inspektorfenster werden die Eigenschaften der Schnittstelle angezeigt.
2. Wählen Sie in der Bereichsnavigation des Inspektorfensters die Gruppe "Port-Konfiguration".
   Die Einstellungen des Ports werden angezeigt.
3. Wählen Sie aus der Klappliste "Übertragungsgeschwindigkeit" die Geschwindigkeit für die Datenübertragung. Beachten Sie bei frei programmierter Kommunikation den Einfluss der Übertragungsgeschwindigkeit auf die Umschaltzeit.
4. Wählen Sie aus der Klappliste "Parität" die Art der Erkennung fehlerhaft übertragener Informationswörter aus.
5. Wählen Sie aus der Klappliste "Datenbits" aus, ob ein Zeichen aus acht Bit oder sieben Bit besteht.
6. Wählen Sie aus der Klappliste "Stoppbit" aus, mit wie vielen Bits das Ende eines übertragenen Worts gekennzeichnet werden soll.

7. Wählen Sie aus der Klappliste "Flusskontrolle" aus, durch welches Verfahren ein reibungsloser Datenstrom zwischen Sender und Empfänger sichergestellt werden soll. Dieser Parameter ist nur für die RS232-Schnittstelle einstellbar.
   - Geben Sie in das Feld "XON-Zeichen" einen HEX-Wert ein, bei dessen Auftreten die Nachrichtenübertragung fortgesetzt wird. Dieser Parameter ist nur für die softwaregesteuerte Datenflusskontrolle einstellbar.
   - Geben Sie in das Feld "XOFF-Zeichen" einen HEX-Wert ein, bei dessen Auftreten die Nachrichtenübertragung für die parametrierte Wartezeit ausgesetzt werden soll. Dieser Parameter ist nur für die softwaregesteuerte Datenflusskontrolle einstellbar.
8. Geben Sie in das Feld "Wartezeit" eine Zeit in ms ein, die nach dem Ende einer Nachricht gewartet werden muss, bevor die nächste Übertragung starten kann.

Hinweis

Sie können die Schnittstelle auch in der Netzsicht projektieren. Hierzu müssen Sie aber zuerst über die tabellarische Netzsicht das Kommunikationsmodul auswählen und anschließend im Inspektorfenster die Schnittstelle auswählen. Danach können Sie wie oben beschrieben fortfahren.

Siehe auch

Einstellen der Datenflusskontrolle (Seite 1409)

Einstellen der Datenflusskontrolle

Datenflusskontrolle

Datenflusskontrolle ist ein Verfahren, mit dem für ein ausgeglichenes Sende- und Empfangsverhalten gesorgt wird. Optimalerweise gehen durch die intelligente Steuerung keine Daten verloren. Sie stellt sicher, dass ein Gerät nicht mehr Informationen versendet als der empfangende Partner verarbeiten kann.

Für die Datenflusskontrolle gibt es zwei Verfahren:

- Hardwaregesteuerte Datenflusskontrolle
- Softwaregesteuerte Datenflusskontrolle

Bei beiden Methoden müssen die DSR-Signale der Kommunikationspartner beim Beginn der Übertragung aktiv sein. Sind die DSR-Signale inaktiv, wird nicht mit der Übertragung begonnen.

Das RS232-Kommunikationsmodul kann mit beiden Verfahren umgehen. Das RS485-Kommunikationsmodul unterstützt keine Datenflusskontrolle.

Hardwaregesteuerte Datenflusskontrolle

Hardwaregesteuerte Datenflusskontrolle erfolgt über die Signale Request-to-send (RTS) und Clear-to-send (CTS). Mit dem RS232-Kommunikationsmodul wird das RTS-Signal über den Ausgang von Pin 7 übertragen. Das CTS-Signal wird über Pin 8 empfangen.

Ist die hardwaregesteuerte Datenflusskontrolle aktiviert, wird das RTS-Signal immer dann aktiv gesetzt, wenn Daten gesendet werden. Gleichzeitig wird das CTS-Signal überwacht, um festzustellen, ob das empfangende Gerät Daten annehmen kann. Ist das CTS-Signal aktiv, kann das Modul Daten solange Daten übertragen bis das CTS-Signal inaktiv wird. Wird das CTS-Signal inaktiv, muss die Datenübertragung für die parametrierte Wartezeit ausgesetzt werden. Ist das CTS-Signal auch nach der parametrierten Wartezeit noch inaktiv, wird die Datenübertragung abgebrochen und ein Fehler an das Anwenderprogramm zurückgemeldet.

Datenflusskontrolle durch Hardware-Handshaking

Wird die Datenflusskontrolle durch Hardware-Handshaking geregelt, setzt das sendende Gerät das RTS-Signal standardmäßig aktiv. Ein Gerät, wie z. B. ein Modem, kann in diesem Fall zu jeder Zeit Daten übertragen. Es wartet nicht auf das CTS-Signal des Empfängers. Das sendende Gerät überwacht selbst seine eigene Übertragung indem es nur eine begrenzte Anzahl an Nachrichten-Frames (Zeichen) sendet, beispielsweise um einen Overflow im Empfangspuffer zu verhindern. Kommt es dennoch im Empfangspuffer zu einem Overflow, muss das übertragende Gerät die Nachricht zurückhalten und einen Fehler an das Anwenderprogramm zurückmelden.

Softwaregesteuerte Datenflusskontrolle

Softwaregesteuerte Datenflusskontrolle verwendet bestimmte Zeichen innerhalb der Nachrichten, welche die Übertragung steuern. Diese Zeichen sind ASCII-Zeichen, die unter XON und XOFF parametriert werden.

XOFF gibt an, wann eine Übertragung ausgesetzt werden muss. XON gibt an, wann eine Übertragung fortgesetzt werden kann.

Empfängt das sendende Gerät das XOFF-Zeichen, muss es für die parametrierte Wartezeit mit dem Senden aussetzen. Wird nach der parametrierten Wartezeit das XON-Zeichen gesendet, wird die Übertragung fortgeführt. Kommt nach der Wartezeit kein XON-Zeichen, wird ein Fehler an das Anwenderprogramm zurückgegeben.

Software-Datenflusskontrolle erfordert Full-Duplex-Kommunikation, da der empfangende Partner das XON-Zeichen während der laufenden Übertragung senden muss.

Siehe auch

Kommunikations-Port konfigurieren (Seite 1408)

Konfiguration der Nachrichtenübertragung

Frei programmierbare Kommunikation

Sie können den Datenverkehr zwischen einem Kommunikationsmodul und einem extern über die serielle Schnittstelle angeschlossenen Gerät frei steuern. Dafür müssen Sie selbst ein Kommunikationsprotokoll definieren. In der frei programmierbaren Kommunikation werden ASCII- und Binärprotokolle zur Nachrichtenübertragung unterstützt.

Innerhalb des Kommunikationsprotokolls müssen Sie festlegen, anhand welcher Kriterien Anfang und Ende einer übertragenen Nachricht im laufenden Datenstrom erkannt werden sollen.

Die frei programmierbare Kommunikation kann nur im Betriebszustand RUN aktiviert werden. Beim Versetzen in den Betriebszustand STOP wird die frei programmierbare Kommunikation angehalten.

Festlegen des Kommunikationsprotokolls

Das Kommunikationsprotokoll können Sie folgendermaßen festlegen:

- Mit Hilfe des Anwenderprogramms
  - Das Verhalten beim Senden von Daten wird über die Anweisung SEND_CFG gesteuert.
  - Das Verhalten beim Empfangen von Daten wird über die Anweisung RCV_CFG gesteuert.
- Mit Hilfe der grafischen Parametrierung im Inspektorfenster

Hinweis

Wenn Sie das Kommunikationsprotokoll über das Anwenderprogramm ändern, werden die Einstellungen der grafischen Projektierung überschrieben.

Beachten Sie, dass über das Anwenderprogramm vorgenommene Einstellungen im Falle eines Spannungsverlusts nicht erhalten bleiben.

Siehe auch

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 1411)

Festlegen der Einstellungen für das Senden (Seite 1413)

Festlegen des Nachrichtenanfangs (Seite 1414)

Festlegen des Nachrichtenendes (Seite 1415)

Frei programmierbare Kommunikation mit RS232-Geräten

RS232/PPI-Multi-Master-Kabel und frei programmierbare Kommunikation mit RS232-Geräten

Mit Hilfe des RS232/PPI-Multi-Master-Kabels und der frei programmierbaren Kommunikation können Sie an die Kommunikationsmodule der S7-1200 viele Geräte anschließen, die mit dem RS232-Standard kompatibel sind. Das Kabel muss jedoch zunächst für den Modus "PPI/frei programmierbare Kommunikation" eingestellt werden.

Einstellungen am Kabel

Die Schalter am Kabel müssen folgendermaßen eingestellt werden:

- Schalter 5 muss auf 0 stehen
- Schalter 6 stellt entweder den lokalen Modus (DCE) oder den entfernten Modus (DTE) ein:
  - Schalter auf 0 für den lokalen Modus
  - Schalter auf 1 für den entfernten Modus

Umschalten zwischen Sende- und Empfangsmodus

Das RS232/PPI-Multi-Master-Kabel ist im Sendemodus, wenn Daten von der RS232-Schnittstelle an die RS485-Schnittstelle gesendet werden. Das Kabel ist im Empfangsmodus, wenn es sich im Ruhezustand befindet oder wenn Daten von der RS485-Schnittstelle an die RS232-Schnittstelle gesendet werden. Das Kabel schaltet sofort vom Empfangs- in den Sendemodus, wenn es Zeichen in der RS232-Sendeleitung erkennt.

Unterstützte Übertragungsgeschwindigkeiten

Das RS232/PPI-Multi-Master-Kabel unterstützt Übertragungsgeschwindigkeiten zwischen 1200 Baud und 115,2 kBaud. Mit Hilfe der DIP-Schalter am PC/PPI-Kabel kann das RS232/PPI-Multi-Master-Kabel auf die gewünschte Übertragungsgeschwindigkeit eingestellt werden.

Die folgende Tabelle zeigt die Schalterstellungen für die entsprechenden Übertragungsgeschwindigkeiten:

Übertragungsgeschwindigkeit	Umschaltzeit	Einstellungen (1 = oben)
115200 bit/s	0,15 ms	110
57600 bit/s	0,3 ms	111
38400 bit/s	0,5 ms	000
19200 bit/s	1,0 ms	001
9600 bit/s	2,0 ms	010
4800 bit/s	4,0 ms	011
2400 bit/s	7,0 ms	100
1200 bit/s	14,0 ms	101

Das Kabel kehrt zum Empfangsmodus zurück, wenn sich die RS232-Sendeleitung für eine bestimmte Zeit, die als Umschaltzeit des Kabels definiert ist, im Ruhezustand befindet. Die eingestellte Übertragungsgeschwindigkeit beeinflusst die Umschaltzeit, wie in der Tabelle dargestellt.

Einfluss der Umschaltzeit

Bei der Arbeit mit einem RS232/PPI-Multi-Master-Kabel in einem System, in dem auch die frei programmierte Kommunikation eingesetzt wird, muss das Programm die Umschaltzeit aus folgenden Gründen beachten:

- Die Kommunikationsbaugruppe reagiert auf Meldungen, die von dem RS232-Gerät gesendet werden.
  Nachdem die Kommunikationsbaugruppe vom RS232-Gerät eine Anforderung empfangen hat, muss sie das Senden der Reaktionsmeldung für einen Zeitraum verzögern, der größer oder gleich der Umschaltzeit des Kabels ist.
- Das RS232-Gerät reagiert auf Meldungen, die von der Kommunikationsbaugruppe gesendet werden.
  Nachdem die Kommunikationsbaugruppe vom RS232-Gerät eine Reaktionsmeldung empfangen hat, muss sie das Senden der nächsten Anforderungsmeldung für einen Zeitraum verzögern, der größer oder gleich der Umschaltzeit des Kabels ist.

In beiden Situationen enthält das RS232-PPI-Multi-Master-Kabel durch die Verzögerung genügend Zeit, um vom Sendemodus in den Empfangsmodus zu wechseln, damit die Daten von der RS485-Schnittstelle an die RS232-Schnittstelle gesendet werden können.

Siehe auch

Konfiguration der Nachrichtenübertragung (Seite 1410)

Festlegen der Einstellungen für das Senden (Seite 1413)

Festlegen des Nachrichtenanfangs (Seite 1414)

Festlegen des Nachrichtenendes (Seite 1415)

Festlegen der Einstellungen für das Senden

Senden von Nachrichten

Sie können zwischen einzelnen Nachrichten Pausen programmieren.

Die folgende Tabelle zeigt, welche Pausen vorgegeben werden können:

Parameter	Definition
RTS ON Verzögerung	Sie können die Zeit parametrieren, die nach der Sendeanforderung RTS (Request to send) vergehen muss, ehe die tatsächliche Datenübertragung beginnt.
RTS OFF Verzögerung	Sie können die Zeit parametrieren, die nach der vollständigen Übertragung verstreichen soll, ehe das RTS-Signal deaktiviert wird.

Parameter	Definition
Sende Pause zu Meldungsbeginn	Sie können festlegen, dass nach dem Beginn jeder Nachrichtenübertragung zusätzlich eine Pause gesendet wird, wenn die RTS ON-Verzögerungszeit verstrichen ist. Die Pausenzeit wird in Bitzeiten angegeben.
Sende Idle Line nach einer Pause	Sie können festlegen, dass nach einer parametrierten Pause zu Meldungsbeginn, das Signal "Idle Line" ausgegeben wird und die Leitung somit als "unbeschäftigt" signalisiert wird. Um den Parameter zu aktivieren, muss "Sende Pause zu Meldungsbeginn" aktiviert sein. Die Dauer des "Idle Line"-Signals wird in Bitzeiten angegeben.

Siehe auch

Festlegen des Nachrichtenanfangs (Seite 1414)

Festlegen des Nachrichtenendes (Seite 1415)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 1411)

Festlegen des Nachrichtenanfangs

Erkennen des Nachrichtenanfangs

Um dem Empfänger zu signalisieren, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt, müssen im Übertragungsprotokoll Kriterien festgelegt werden, die das Ende und den Anfang einer Nachricht kennzeichnen.

Ist ein Kriterium erfüllt, welches den Nachrichtenbeginn anzeigt, wird mit dem Durchsuchen des Datenstroms nach Kriterien für das Nachrichtenende begonnen.

Zum Erkennen des Nachrichtenbeginns gibt es zwei unterschiedliche Verfahren:

- Mit beliebigem Zeichen beginnen:
  Jedes beliebige Zeichen kann den Beginn einer Nachricht definieren. Dieses Verhalten ist voreingestellt.
- Mit spezieller Bedingung beginnen:
  Der Nachrichtenbeginn wird anhand von festgelegten Bedingungen erkannt.

Bedingungen für das Erkennen des Nachrichtenbeginns

Die folgende Tabelle zeigt die verschiedenen Möglichkeiten, wie Sie den Beginn einer Nachricht definieren können:

Parameter	Definition
Erkenne Nachrichtenbeginn bei Line Break	Der Empfänger erkennt einen Line Break, wenn der Empfangsdatenstrom für mehr als ein Zeichen Länge unterbrochen wird. Ist dies der Fall, wird durch den Line Break der Nachrichtenbeginn festgelegt.
Erkenne Nachrichtenbeginn bei Idle Line	Der Beginn einer Nachricht wird erkannt, wenn die Übertragungsleitung zum Senden für eine bestimmte Zeit (angegeben in Bitzeiten) im Zustand "Idle" ist und anschließend von einem Ereignis, wie z. B. dem Empfang eines Zeichens, gefolgt wird.

Parameter	Definition
Erkenne Nachrichtenbeginn mit einzelnem Zeichen	Der Beginn wird einer Nachricht wird beim Auftreten eines bestimmten Zeichens erkannt. Das Zeichen können Sie als HEX-Wert eingeben..
Erkenne Nachrichtenbeginn mit einer Zeichenkette	Der Beginn einer Nachricht wird erkannt, wenn eine der festgelegten Zeichenfolgen im Datenstrom auftritt. Sie können bis zu vier Zeichensequenzen mit jeweils bis zu fünf Zeichen festlegen.

Die einzelnen Bedingungen lassen sich beliebig verknüpfen.

Siehe auch

Festlegen der Einstellungen für das Senden (Seite 1413)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 1411)

Festlegen des Nachrichtenendes

Erkennen des Nachrichtenendes

Um dem Empfänger zu signalisieren, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt, müssen im Übertragungsprotokoll Kriterien festgelegt werden, die das Ende und den Anfang einer Nachricht kennzeichnen.

Zum Erkennen des Nachrichtenendes gibt es insgesamt sechs unterschiedliche Verfahren, die sich beliebig verknüpfen lassen. Die folgende Tabelle zeigt die verschiedenen Einstellmöglichkeiten:

Parameter	Definition
Erkenne Nachrichtenende durch Nachrichten-Zeitüberschreitung	Das Nachrichtenende wird automatisch erkannt, wenn eine vorgegebene Maximaldauer für eine Nachricht überschritten wird. Es sind Werte von 0 bis 65535 ms einstellbar.
Erkenne Nachrichtenende durch Antwort-Zeitüberschreitung	Das Nachrichtenende wird erkannt, wenn nach dem Übertragen von Daten in der vorgegebenen Zeit keine Antwort erfolgt. Es sind Werte von 0 bis 65535 ms einstellbar.
Erkenne Nachrichtenende durch Zeitüberschreitung innerhalb der Zeichen	Das Nachrichtenende wird erkannt, wenn eine vorgegebene Zeit in Bitzeiten zwischen zwei Zeichen überschritten wird. Es sind Werte von 0 bis 2500 Bitzeiten einstellbar. Die S7-1200 CPU gibt eine maximale Zeit von acht Sekunden vor, auch wenn der eingestellte Wert mehr als acht Sekunden Dauer ergibt.
Erkenne Nachrichtenende durch maximale Länge	Das Nachrichtenende wird erkannt, wenn eine maximale Länge der Nachricht überschritten wird. Es sind Werte von 1 bis 1023 Zeichen einstellbar.

Parameter	Definition
Lese Nachrichtenlänge aus Nachricht	Die Nachricht selbst enthält Informationen über die Länge der Nachricht. Das Nachrichtenende wird erreicht, wenn der Wert, der aus der Nachricht entnommen wurde, erreicht wird. Welche Zeichen für die Auswertung der Nachrichtenlänge herangezogen werden sollen, wird mit den folgenden Parametern festgelegt: • Offset des Längenfeldes in der Nachricht Der Wert bestimmt die Position des Zeichens in der Nachricht, welches für die Bestimmung der Nachrichtenlänge herangezogen werden soll. Es sind Werte von 1 bis 1022 Zeichen einstellbar. • Größe des Längenfeldes Der Wert gibt an, wie viele Zeichen ab der ersten Auswertungsposition zur Bestimmung der Nachrichtenlänge herangezogen werden sollen. Es sind Werte von 0, 1, 2 und 4 Zeichen einstellbar. • Den Daten folgendes Längenfeld (gehört nicht zur Nachrichtenlänge) Der Wert gibt die Anzahl der Bytes nach dem Längenfeld an, die bei der Auswertung der Nachrichtenlänge nicht berücksichtigt werden sollen. Es sind Werte von 0 bis 255 Zeichen einstellbar.
Erkenne Nachrichtenende mit einer Zeichenkette	Das Ende einer Nachricht wird erkannt, wenn die festgelegte Zeichenfolgen im Datenstrom auftritt. Sie können bis zu fünf Zeichen innerhalb der Zeichenkette festlegen, die überprüft werden sollen. Wenn die festgelegten Zeichen an der richtigen Stelle innerhalb der Nachricht auftreten, wird das Nachrichtenende erkannt. Um das Nachrichtenende beispielsweise dann zu erkennen, wenn Zeichen 1 und Zeichen 3 einen bestimmten Wert besitzen, müssen die Optionskästchen für Zeichen 1 und Zeichen 3 aktiviert und ein Zeichenwert eingetragen sein.

Siehe auch

Festlegen der Einstellungen für das Senden (Seite 1413)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 1411)

Verwenden von Systemmerkern

Systemmerker

Ein Systemmerker ist ein Merker mit definierten Werten.

Welches Merkerbyte der CPU zum Systemmerkerbyte wird, bestimmen Sie bei der Parametrierung des Systemmerkers.

Nutzen

Systemmerker können Sie im Anwenderprogramm verwenden, um z. B. Programmteile nur im ersten Zyklus nach dem Anlauf auszuführen oder die Diagnose bei Änderung des Diagnosezustands auszuwerten. Zwei Systemmerker sind konstant 1 bzw. konstant 0.

Bits des Systemmerkerbytes

Die folgende Tabelle zeigt die Bedeutung der Systemmerker:

Bit des Systemmerkerbytes	7	6	5	4	3	2	1	0
Bedeutung	Reserviert (=0)	Reserviert (=0)	Reserviert (=0)	Reserviert (=0)	=0	=1	=1 bei Änderung des Diagnosestatus	=1 während des Anlaufs und im ersten Zyklus nach dem Anlauf, sonst 0

Hinweis

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

Verwenden von Taktmerkern

Taktmerker

Ein Taktmerker ist ein Merker, der seinen Binärzustand periodisch im Puls-Pausen-Verhältnis 1:1 ändert.

Welches Merkerbyte der CPU zum Taktmerkerbyte wird, bestimmen Sie bei der Parametrierung des Taktmerkers.

Nutzen

Taktmerker können Sie im Anwenderprogramm verwenden, um z. B. Leuchtmelder mit Blinklicht anzusteuern oder periodisch wiederkehrende Vorgänge anzustoßen, etwa das Erfassen eines Istwerts.

Mögliche Frequenzen

Jedem Bit des Taktmerkerbytes ist eine Frequenz zugeordnet. Die folgende Tabelle zeigt die Zuordnung:

Bit des Taktmerkerbytes	7	6	5	4	3	2	1	0
Periodendauer (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frequenz (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Hinweis

Taktmerker laufen asynchron zum CPU-Zyklus, d. h. in langen Zyklen kann sich der Zustand des Taktmerkers mehrfach ändern.

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

Schutz

Einstellmöglichkeiten für die Schutzstufe (FW V1 bis V3)

Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Schutzstufen der S7-1200 CPUs V1 bis V3 nutzen.

Auswirkungen der eingestellten Schutzstufe

Sie können zwischen folgenden Schutzstufen wählen:

- Kein Schutz: Dies entspricht dem voreingestellten Verhalten. Sie können kein Passwort eingeben. Lesender und schreibender Zugriff ist immer erlaubt.
- Schreibschutz: Es ist nur lesender Zugriff möglich. Sie können keine Daten in der CPU ändern und keine Bausteine und keine Konfiguration laden. Vom Schreibschutz ausgenommen sind HMI-Zugriffe und Kommunikation zwischen CPUs. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes erforderlich.
- Schreib-/Leseschutz: Es ist im Bereich "Erreichbare Teilnehmer" und im Projekt für Teilnehmer, die online geschaltet sind, kein schreibender und kein lesender Zugriff möglich. Lediglich der CPU-Typ und die Identifikationsdaten können in der Projektnavigation unter "Erreichbare Teilnehmer" angezeigt werden. Die Anzeige von Online-Informationen oder von Bausteinen unter "Erreichbare Teilnehmer" bzw. im Projekt für Online geschaltete Teilnehmer ist möglich.
  Vom Schreibschutz ausgenommen sind HMI-Zugriffe und Kommunikation zwischen CPUs. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes erforderlich.

Verhalten einer Passwort-geschützten CPU im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Falle eines Passwortschutzes werden Sie zur Eigabe eines Passworts aufgefordert.

Beispiel: Die Baugruppe wurde mit Schreibschutz parametriert und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht mit Passwort anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird. Wenn das Projekt geschlossen wird, erlischt die Zugangsberechtigung ebenfalls.

Hinweis

Funktionen zur Prozessführung, -beobachtung und -kommunikation können Sie nicht einschränken.

Einige Funktionen werden durch ihre Verwendung als Online-Daten dennoch geschützt. RUN/STOP in der Taskcard "Online-Tools" oder "Uhrzeit stellen" in der Online- und Diagnosesicht sind daher schreibgeschützt.

Einstellmöglichkeiten für den Schutz (FW ab V4)

Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Zugriffsstufen der S7-1200 CPUs ab V4 nutzen.

S7-1200 CPUs bieten verschiedene Zugriffsstufen, um den Zugang zu bestimmten Funktionen einzuschränken.

Die Parametrierung der Zugriffsstufen erfolgt in einer Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Zugriffsstufe geben an, welche Operationen maximal möglich sind, ohne das Passwort dieser Zugriffsstufe zu kennen. Möchten Sie die Funktionen nicht markierter Optionskästchen nutzen, ist die Eingabe eines Passwortes notwendig.

ACHTUNG
Projektierung einer Zugriffsstufe ersetzt nicht den Know-how-Schutz Die Parametrierung von Zugriffsstufen verhindert unrechtmäßige Änderungen an der CPU, indem die Rechte zum Download eingeschränkt werden. Bausteine auf der Memory Card sind jedoch nicht schreib- oder lesegeschützt. Um den Code von Bausteinen auf der Memory Card zu schützen, verwenden Sie den Know-how-Schutz.

Voreingestelltes Verhalten

Voreingestellt ist die Zugriffsstufe "Vollzugriff (kein Schutz)". Jeder Nutzer kann die Hardware-Konfiguration und die Bausteine lesen und verändern. Ein Passwort ist nicht parametriert und wird auch für den Online-Zugriff nicht benötigt.

Die Zugriffsstufen im Einzelnen

Bei einer S7-1200 CPU können Sie folgende Zugriffsstufen parametrieren:

- Vollzugriff (kein Schutz): Die Hardware-Konfiguration und die Bausteine können von jedem gelesen und verändert werden.
- Lesezugriff: Mit dieser Zugriffsstufe ist ohne Angabe des Passwortes nur lesender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich, d. h. Sie können Hardware-Konfiguration und Bausteine ins Programmiergerät laden. Möglich ist außerdem der HMI-Zugang und Zugriff auf Diagnosedaten.
  Sie können ohne Eingabe des Passwortes keine Bausteine und keine Hardware-Konfiguration in die CPU laden. Außerdem sind ohne Passwort schreibende Testfunktionen und Firmware-Updates **nicht** möglich.
- HMI-Zugriff: Mit dieser Zugriffsstufe ist ohne Angabe des Passwortes nur der HMI-Zugang und der Zugriff auf Diagnosedaten möglich.
  Sie können ohne Angabe des Passwortes weder Bausteine und Hardware-Konfiguration in die CPU laden noch von der CPU Bausteine und Hardware-Konfiguration ins Programmiergerät laden. Außerdem ist ohne Passwort Folgendes **nicht** möglich: schreibende Testfunktionen, Wechsel des Betriebszustands (RUN/STOP) und Firmware-Update.
- Kein Zugriff (kompletter Schutz): Wenn die CPU komplett geschützt ist, dann ist weder lesender noch schreibender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich. Auch der HMI-Zugriff ist nicht möglich. Die Server-Funktion für PUT/GET-Kommunikation ist in dieser Zugriffsstufe deaktiviert (nicht änderbar).
  Durch die Legitimation mit dem Passwort erhalten Sie wieder Vollzugriff auf die CPU.

Verhalten einer Passwort-geschützten Baugruppe im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Falle eines Passwortschutzes werden Sie zur Passworteingabe aufgefordert.

Beispiel: Die Baugruppe wurde mit Lesezugriff parametriert und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird.

Jede Zugriffsstufe lässt auch ohne Eingabe eines Passwortes den uneingeschränkten Zugriff auf bestimmte Funktionen zu, z. B. Identifikation über die Funktion "Erreichbare Teilnehmer".

Zugriffsstufen parametrieren

Im Folgenden erfahren Sie, wie Sie für eine S7-1200 CPU ab V4 eine Zugriffsstufe parametrieren und Passwörter eingeben.

Für eine S7-1200 CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Zugriffsstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Zugriffsstufe".

Beispiel

Sie wählen die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede der in der Tabelle darüber liegenden Zugriffsstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Vollzugriff (kein Schutz)) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Lesezugriff) wirkt, als wäre die CPU schreibgeschützt. Nutzer, die dieses Passwort kennen, haben nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (HMI-Zugriff) wirkt, als wäre die CPU schreib- und lesegeschüzt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

Vorgehen

Um die Zugriffsstufen für eine S7-1200 CPU zu parametrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften der Baugruppe im Inspektorfenster.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Schutz".
   Eine Tabelle mit den möglichen Zugriffsstufen wird im Inspektorfenster angezeigt.

Zugriffsstufe	Zugriff			Zugriffserlaubnis	
Schutz	HMI	Lesen	Schreiben	Passwort	Bestätigung
Vollzugriff (kein Schutz)	✓	✓	✓		
Lesezugriff	✓	✓			
HMI-Zugriff	✓				
Kein Zugriff (kompletter Schutz)					

3. Aktivieren Sie die gewünschte Zugriffsstufe in der ersten Spalte der Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Schutzstufe zeigen Ihnen, welche Operationen noch möglich sind, ohne das Passwort einzugeben.

4. Vergeben Sie in der Spalte "Passwort" in der ersten Zeile ein Passwort für den Vollzugriff. Wiederholen Sie zum Schutz vor Fehleingaben das gewählte Passwort in der Spalte "Bestätigung".
   Achten Sie darauf, dass das Passwort ausreichend sicher ist, d. h. dass es kein durch eine Maschine erkennbares Muster besitzt!
   Die Eingabe eines Passwortes in der ersten Zeile "Vollzugriff (kein Schutz)" ist obligatorisch und ermöglicht dem Passwort-Kenner uneingeschränkten Zugriff auf die CPU, unabhängig von der gewählten Schutzstufe.
5. Weisen Sie weiteren Zugriffsstufen nach Bedarf weitere Passwörter zu, falls die gewählte Schutzstufe das erlaubt.
6. Laden Sie die Hardware-Konfiguration, damit die Zugriffsstufe wirksam wird.

Ergebnis

Die Hardware-Konfiguration und die Bausteine sind entsprechend der eingestellten Zugriffsstufe vor unberechtigtem Zugriff geschützt. Wenn eine Operation aufgrund der parametrierten Zugriffsstufe nicht ohne Passwort ausgeführt werden kann, wird ein Dialog zur Eingabe eines Passwortes aufgeblendet.

Einschränken von Kommunikationsdiensten

Einführung

Die CPU kann Server für eine Reihe von Kommunikationsdiensten sein, d. h. ohne dass Sie für die CPU Verbindungen projektiert und programmiert haben, können andere Kommunikationsteilnehmer auf CPU-Daten zugreifen.

Damit entfällt für die lokale CPU als Server die Möglichkeit, die Kommunikation zu den Clients kontrollieren.

Ob diese Art der Kommunikation im Betrieb für die lokale CPU zulässig ist oder nicht, wird durch den Parameter "Verbindungsmechanismen" im Bereich "Schutz" der CPU-Parameter bestimmt.

Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben

In der Voreinstellung ist die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner (...) erlauben" deaktiviert. In diesem Fall ist lesender und schreibender Zugriff auf CPU-Daten nur möglich bei Kommunikationsverbindungen, die eine Projektierung bzw. Programmierung sowohl für die lokale CPU als auch für den Kommunikationspartner voraussetzen. Zugriffe über BSEND/BRCV-Anweisungen sind z. B. möglich.

Verbindungen, für die die lokale CPU nur Server ist (d. h. für die lokale CPU ist keine Projektierung/Programmierung der Kommunikation zum Kommunikationspartner vorhanden), sind damit im Betrieb der CPU nicht möglich; z. B.

- bei PUT/GET-, FETCH/WRITE- oder FTP-Zugriffen über Kommunikationsmodule
- bei PUT/GET-Zugriffen von anderen S7-CPUs
- bei HMI-Zugriffen, die über PUT/GET-Kommunikation realisiert sind

Wenn Sie den Client-seitigen Zugriff auf CPU-Daten erlauben wollen, d. h. wenn Sie die Kommunikationsdienste der CPU nicht einschränken wollen, dann aktivieren Sie die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben".

Sammelmeldungen bei Security-Ereignissen

Wissenswertes zu Sammelmeldungen bei Security-Ereignissen

Security-Ereignisse im Diagnosepuffer

Die CPU legt Informationen zu wichtigen Vorgängen und Ereignissen im Diagnosepuffer ab.

Die folgenden Security-Ereignisse (Ereignistypen) führen jeweils zu einem Eintrag im Diagnosepuffer der S7-1200:

- Online-Gehen, mit richtigem oder falschem Passwort
- Manipulierte Kommunikationsdaten erkannt
- Manipulierte Daten auf Memory Card erkannt
- Manipulierte Firmware-Update-Datei erkannt
- Geänderte Schutzstufe (Zugriffsschutz) in CPU geladen
- Passwort-Legitimierung eingeschränkt oder freigegeben (per Anweisung oder CPU-Display)
- Online-Zugriff verweigert wegen Überschreitung der Anzahl gleichzeitig möglicher Zugriffe
- Zeitüberschreitung bei Inaktivität einer bestehenden Online-Verbindung
- Anmelden am Webserver mit richtigem oder falschem Passwort
- Sicherung der CPU erstellen (Backup)
- Wiederherstellen der CPU-Projektierung (Restore)
- Im Anlauf:
  - Projekt auf der SIMATIC Memory Card hat sich geändert (die SIMATIC Memory Card ist dieselbe)
  - SIMATIC Memory Card wurde ausgetauscht

Sammelmeldung für Security-Ereignisse

Um zu verhindern, dass der Diagnosepuffer von sehr vielen identischen Security-Ereignissen "überschwemmt" wird, können Sie ab STEP 7 V13 Servicepack 1 parametrieren, dass diese Ereignisse als Sammelmeldung in den Diagnosepuffer eingetragen werden. Pro Interval (Überwachungszeitraum) erzeugt die CPU dann nur noch eine Sammelmeldung je Ereignistyp.

Beispiel: Brute-Force-Attacken

Bei Brute-Force-Attacken versuchen Angreifer, Zugang zur CPU zu erlangen, indem sie systematisch sehr viele der möglichen Passwortkombinationen ausprobieren. Dadurch erhält die CPU innerhalb von Sekunden sehr viele Login-Anfragen und der große Ansturm

identischer Einzel-Einträge im Diagnosepuffer führt zum Verlust wichtiger Diagnoseinformationen. Sammelmeldungen sorgen hier für Abhilfe.

Funktionsweise bei Sammelmeldungen

Die CPU trägt die ersten drei Ereignisse eines Ereignistyps sofort in den Diagnosepuffer ein. Alle danach folgenden Security-Ereignisse dieses Typs ignoriert sie.

Am Ende des Überwachungszeitraums (Intervalls) erzeugt die CPU eine Sammelmeldung, in der sie den Ereignistyp und die Häufigkeit dieses Ereignisses während des abgelaufenen Intervalls einträgt.

Treten diese Security-Ereignisse auch in den folgenden Intervallen auf, so generiert die CPU nur noch jeweils eine Sammelmeldung pro Folgeintervall.

Im Diagnosepuffer hinterlässt eine Attacke folgendes Muster: Drei Einzelmeldungen gefolgt von einer Reihe von Sammelmeldungen. Diese Reihe kann aus zwei, drei oder mehr Sammelmeldungen bestehen, das ist abhängig von der bewählten Überwachungszeit und der Dauer der Attacke.

Da die Meldungen im Diagnosepuffer einen Zeitstempel enthalten, lässt sich die Dauer der Attacke bestimmen.

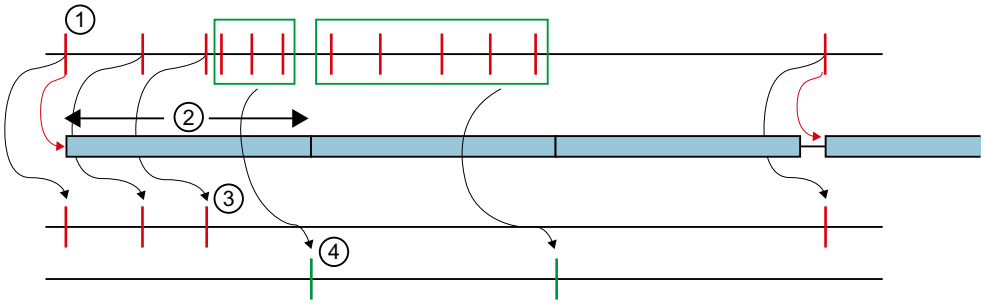

① Security-Ereignisse eines Typs, zum Beispiel Login-Versuche mit ungültigem Passwort.

② Intervall (Überwachungszeitraum): Wenn ein Security-Ereignis eines Typs zum ersten Mal auftritt (oder erneut auftritt), wird die Überwachungszeit gestartet (oder erneut gestartet).

③ Einzelmeldungen: Die drei ersten Security-Ereignisse eines Typs werden sofort in den Diagnosepuffer eingetragen. Ab dem vierten Security-Ereignis dieses Typs erzeugt die CPU nur mehr Sammelmeldungen.

Tritt nach einer Pause von mindestens einem Intervall wieder ein Security-Ereignis dieses Typs auf, trägt die CPU eine Einzelmeldung in den Diagnosepuffer ein und startet den Überwachungszeitraum erneut.

④ Sammelmeldungen: Die CPU erzeugt nach drei Security-Ereignissen nur noch eine Sammelmeldung als Zusammenfassung aller weiteren Security-Ereignisse in diesem Intervall. Treten diese Security-Ereignisse auch in den folgenden Intervallen auf, so generiert die CPU jeweils nur eine Sammelmeldung pro Folgeintervall.

Sammelmeldung bei Security-Ereignissen projektieren

Voraussetzungen

- STEP 7 V14
- S7-1200-CPU ab Firmware V4.2

Vorgehen

Um Sammelmeldungen bei Security-Ereignissen zu projektieren, gehen Sie in folgendermaßen vor:

1. Klicken Sie in der Netzsicht auf das Symbol der CPU.
   Die Eigenschaften der CPU werden im Inspektorfenster angezeigt.
2. Navigieren Sie zum Bereich "Schutz" > "Security-Ereignis".
3. Klicken Sie auf "Security-Ereignis".
4. Aktivieren Sie die Option "Security-Ereignisse bei hohem Nachrichtenaufkommen zusammenfassen", um Sammelmeldungen bei Security-Ereignissen einzuschalten.
5. Stellen Sie die Dauer eines Intervalls (Überwachungszeitraum) ein, voreingestellt sind 20 Sekunden.

Siehe auch

Wissenswertes zu Sammelmeldungen bei Security-Ereignissen (Seite 1423)

SNMP deaktivieren

Das Netzwerk-Management-Protokoll SNMP (Simple Network Management Protocol) ist ein Protokoll, das verschiedene Dienste und Tools zur Erkennung und Diagnose der Netzwerktopologie nutzen. SNMP nutzt das Transportprotokoll UDP. Das SNMP kennt zwei Netzkomponenten, den SNMP-Manager und den SNMP-Client. Der SNMP-Manager überwacht die Netzwerkknoten. Die SNMP-Clients sammeln in den einzelnen Netzwerkknoten verschiedene netzwerkspezifische Informationen und legen Sie in strukturierter Form in der MIB (Management Information Base) ab. Mit Hilfe dieser Daten können verschiedene Dienste und Tools eine ausführliche Netzwerkdiagnose durchführen.

Unter bestimmten Voraussetzungen kann es sinnvoll sein, SNMP zu deaktivieren. Beispiele:

- Die Sicherheitsrichtlinien in Ihrem Netzwerk lassen den Einsatz von SNMP nicht zu.
- Sie verwenden eine eigene SNMP-Lösung, z. B. über eigene Kommunikationanweisungen.

Wenn Sie SNMP für ein Gerät deaktivieren, dann stehen Ihnen verschiedene Möglichen zur Diagnose der Netzwerktopologie (z.B. über das PRONETA-Tool oder über den Webserver der CPU) nicht mehr zur Verfügung.

Voraussetzung

- CPU S7-1200 ab V4.1
- CPU S7-1500 ab V1.8

SNMP deaktivieren

Um SNMP zu deaktivieren, gehen Sie folgendermaßen vor:

1. Legen Sie in STEP 7 einem Datenbaustein an, der die Struktur des Datensatzes $F003_H$ enthält. Die folgende Tabelle zeigt die Struktur des Datensatzes $F003_H$.

Byte	Element	Kodierung	Erläuterung
0-1	BlockID	$F003_H$	Header Die Datensatzlänge wird ab dem Byte 4 "Version" gezählt.
2-3	BlockLenght	8	
4	Version	01_H	
5	Subversion	00_H	
6-7	Reserviert	-	-
8-11	SNMP-Steuerung	Deaktivieren/Aktivieren von SNMP	Wenn Sie SNMP deaktivieren wollen, dann tragen den Wert 0 ein. Wenn Sie SNMP aktivieren wollen, dann tragen Sie den Wert 1 ein.

2. Übertragen Sie den Datensatz $F003_H$ im Anlauf-OB (OB100) mit der Anweisung WRREC (Datensatz Schreiben) an die CPU. Nutzen Sie als Hardware-ID eine integrierte Schnittstelle der CPU.

SNMP deaktivieren - vollständiges Beispiel

Einleitung

Das Beispiel gilt für S7-1500 und S7-1200.

Um den Programmcode in Ihr Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Kopieren Sie den gesamten Programmcode in die Zwischenablage.
2. Öffnen Sie einen Texteditor (z. B. "Editor").
3. Fügen Sie den Inhalt der Zwischenablage in den Texteditor ein mit Strg+V.Speichern Sie das Dokument als scl-Datei ab, z. B. SNMP_DEACT.scl.
4. Öffnen Sie Ihr Projekt in STEP 7.
5. Importieren Sie scl-Datei als externe Quelle.
6. Erzeugen Sie den Anlauf-OB und die Datenbausteine.
7. Rechtsklicken Sie auf die scl-Datei und wählen Sie im Kontextmenü "Baustein aus Quelle generieren".

Programmbeispiel mit WRREC-Aufruf im OB 100

```
DATA_BLOCK "Deactivate_SNMP"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
NON_RETAIN
 VAR
   snmp_deactivate : Bool;
   snmp_record : Struct
     BlockID : UInt;
     BlockLength : UInt;
     "Version" : USInt;
     Subversion : USInt;
     Reserved : UInt;
     SNMPControl : UDInt;
   END_STRUCT;
   snmp_done : Bool;
   snmp_error : Bool;
   snmp_Status : DWord;
 END_VAR

BEGIN
  snmp_deactivate := true;
  snmp_record.BlockID := 16#f003;
  snmp_record.BlockLength := 8;
  snmp_record."Version" := 1;
END_DATA_BLOCK

ORGANIZATION_BLOCK "Startup"
TITLE = "Complete Restart"
{ S7_Optimized_Access := 'TRUE' }
VERSION : 0.1
BEGIN
  REPEAT
    "WRREC_DB_1"(REQ := "Deactivate_SNMP".snmp_deactivate,
                 ID := "Local~PROFINET-Schnittstelle_1",
                 INDEX := 16#b071,
                 DONE => "Deactivate_SNMP".snmp_done,
                 ERROR => "Deactivate_SNMP".snmp_error,
                 STATUS => "Deactivate_SNMP".snmp_Status,
                 RECORD := "Deactivate_SNMP".snmp_record);
   UNTIL "Deactivate_SNMP".snmp_done OR "Deactivate_SNMP".snmp_error
   END_REPEAT;
END_ORGANIZATION_BLOCK

DATA_BLOCK "WRREC_DB_1"
{OriginalPartName := 'WRREC';
 VersionGUID := 'bc169451-58cd-44a3-855b-3f78cc0623c8';
 S7_Optimized_Access := 'TRUE' }
AUTHOR : SIMATIC
FAMILY : DP
NAME : WRREC
VERSION : 1.0
```

```
NON_RETAIN
WRREC
BEGIN
END_DATA_BLOCK
```

Symbolische und numerische Bezeichnung von Anweisungen

Beschreibung

Die Anweisungen aus der Task Card setzen sich aus Funktionen (FCs), Funktionsbausteinen (FBs), Systemfunktionen (SFCs) und Systemfunktionsbausteinen (SFBs) zusammen, die intern durch Nummern bezeichnet werden.

Die folgenden Tabellen zeigen die Zuordnungen der numerischen und symbolischen Bezeichnung.

Funktionsbausteine (FBs)

Numerische Bezeichnung	Symbolische Bezeichnung
FB 105	TC_CONFIG
FB 110	Port_Config
FB 111	Send_Config
FB 112	Receive_Config
FB 113	Send_P2P
FB 114	Receive_P2P
FB 115	Receive_Reset
FB 116	Signal_Get
FB 117	Get_Features
FB 118	Set_Features
FB 163	TC_SEND
FB 164	TC_RECV
FB 165	TC_CON
FB 166	TC_DISCON
FB 804	SET_TIMEZONE
FB 1030	TSEND_C
FB 1031	TRCV_S
FB 1071	USS_DRIVE
FB 1080	MB_COMM_LOAD
FB 1081	MB_MASTER
FB 1082	MB_SLAVE
FB 1084	MB_CLIENT
FB 1085	MB_SERVER
FB 1100	MB_Halt
FB 1101	MC_Home
FB 1102	MC_MoveAbsolute

Numerische Bezeichnung	Symbolische Bezeichnung
FB 1103	MC_MoveJog
FB 1104	MC_MoveRelative
FB 1105	MC_MoveVelocity
FB 1107	MC_Power
FB 1108	MC_Reset
FB 1110	MC_MoveInterrupt
FB 1111	MC_ChangeDynamik
FB 1112	MC_CommandTable
FB 1113	MC_MoveLinearAbs_2D
FB 1114	MC_MoveLinearRel_2D
FB 1115	MC_MoveCircular_2D
FB 1130	PID_Compact
FB 1134	PID_3Step
FB 1140	HSC
FB 2040	RecipeCreate
FB 2041	RecipeOpen
FB 2042	RecipeRead
FB 2043	RecipeWrite
FB 2044	RecipeAppend
FB 2045	RecipeClose

Funktionen (FCs)

Numerische Bezeichnung	Symbolische Bezeichnung
FC 2 (1)	CONCAT
FC 4 (1)	DELETE
FC 11 (1)	FIND
FC 17 (1)	INSERT
FC 20 (1)	LEFT
FC 21 (1)	LEN
FC 22 (1)	LIMIT
FC 25 (1)	MAX
FC 26 (1)	MID
FC 27 (1)	MIN
FC 31 (1)	REPLACE
FC 32 (1)	RIGHT
FC 36 (1)	ENCO
FC 36 (1)	SEL
FC 37	DECO
FC 800	LED
FC 801	IM_DATA
FC 802	DeviceStates

Numerische Bezeichnung	Symbolische Bezeichnung
FC 803	ModuleStates
FC 1070	USS_PORT
FC 1072	USS_RPM
FC 1073	USS_WPM
(1) MC7+ Anweisung	

Systemdatentypen (SDTs)

Numerische Bezeichnung	Symbolische Bezeichnung
SDT 99	WWW_CDB
SDT 513	CONDITIONS
SDT 581	Send_Conditions
SDT 582	Receive_Conditions

Systemfunktionsbausteine (SFBs)

Numerische Bezeichnung	Symbolische Bezeichnung
SFB 0 (1)	CTU
SFB 1 (1)	CTD
SFB 2 (1)	CTUD
SFB 3 (1)	TP
SFB 4 (1)	TON
SFB 5 (1)	TOF
SFB 27	START_OB
SFB 52	RDREC
SFB 53	WRREC
SFB 54	RALRM
SFB 105	T_CONFIG
SFB 106	TDIAG
SFB 107	TRESET
SFB 110	PORT_CFG
SFB 111	SEND_CFG
SFB 112	RCV_CFG
SFB 113	SEND_PTP
SFB 114	RCV_PTP
SFB 115	SGN_GET
SFB 116	SGN_SET
SFB 117	RCV_RST
SFB 120	CTRL_HSC
SFB 122	CTRL_PWM
SFB 124	CTRL_HSC_EXT
SFB 140	DataLogCreate

Numerische Bezeichnung	Symbolische Bezeichnung
SFB 141	DataLogOpen
SFB 142	DateLogWrite
SFB 143	DataLogClear
SFB 144	DataLogClose
SFB 145	DataLogDelete
SFB 146	DataLogNewFile

Systemfunktionen (SFCs)

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 7	DP_PRAL
SFC 11	DPSYC_FR
SFC 13	DPNRM_DG
SFC 14	DPRD_DAT
SFC 16	RD_OBINF
SFC 23	DEL_DB
SFC 28	SET_TINT
SFC 29	CAN_TINT
SFC 30	ACT_TINT
SFC 31	QRY_TINT
SFC 32	SRT_DINT
SFC 33	CAN_DINT
SFC 34	QRY_DINT
SFC 41	DIS_AIRT
SFC 42	EN_AIRT
SFC 43	RE_TRIGR
SFC 45	D_ACT_DP
SFC 46	STP
SFC 82	CREA_DBL
SFC 83	READ_DBL
SFC 84	WRIT_DBL
SFC 86	CREATE_DB
SFC 89	RST_EVOV
SFC 99	WWW
SFC 101	RTM
SFC 117	GET_DIAG
SFC 124	ATTR_DB
SFC 140	IO2MOD
SFC 143	RD_ADDR
SFC 154	RD_LOC_T
SFC 154	DPWR_DAT
SFC 161	WR_LOC_T

Numerische Bezeichnung	Symbolische Bezeichnung
SFC 180	ID2LOG
SFC 181	LOG2ID
SFC 182	ID2GEO
SFC 190	SET_CINT
SFC 191	QRY_CINT
SFC 192	ATTACH
SFC 193	DETACH
MC7+ Anweisung	GET_ERROR
MC7+ Anweisung	GET_ERR_ID

Organisationsbausteine

Anlauf-OBs

Beschreibung

Sie können die Randbedingungen für das Anlaufverhalten Ihrer CPU bestimmen, z. B. die Initialisierungswerte für "RUN". Dazu schreiben Sie ein Anlaufprogramm. Das Anlaufprogramm besteht aus einem oder mehreren Anlauf-OBs (OB-Nummern 100 oder >= 123).

Das Anlaufprogramm wird einmalig beim Übergang vom Betriebszustand "STOP" in den Betriebszustand "RUN" bearbeitet. Für das Anlaufprogramm stehen weder aktuelle Werte aus dem Prozessabbild der Eingänge zur Verfügung noch können Ausgänge gesetzt werden.

Nach der vollständigen Bearbeitung der Anlauf-OBs wird das Prozessabbild der Eingänge eingelesen und das zyklische Programm wird gestartet.

Für die Ausführung des Anlaufprogramms besteht keine zeitliche Begrenzung. Daher ist die Zyklusüberwachungszeit nicht aktiv. Die Verwendung von zeit- oder alarmgesteuerten Organisationsbausteinen ist nicht möglich.

Startinformation

Ein Anlauf-OB hat die folgende Startinformation:

Variable	Datentyp	Beschreibung
LostRetentive	BOOL	= 1, falls remanente Datenbereiche verloren gegangen sind
LostRTC	BOOL	= 1, falls die Echtzeituhr verloren gegangen ist

Siehe auch

Betriebszustand "ANLAUF" (Seite 1366)

Ereignisse und OBs (Seite 1380)

Zyklus-OBs

Einführung

Damit die Programmbearbeitung starten kann, muss im Projekt mindestens ein Zyklus-OB vorhanden sein. Das Betriebssystem ruft diesen Zyklus-OB einmal in jedem Zyklus auf und startet damit die Bearbeitung des Anwenderprogramms. Sie können mehrere Zyklus-OBs (OB-Nummern >= 123) verwenden. Bei Verwendung mehrerer Zyklus-OBs werden diese nacheinander aufgerufen, und zwar in der Reihenfolge ihrer OB-Nummern. Der Zyklus-OB mit der kleinsten OB-Nummer wird zuerst aufgerufen.

Die Zyklus-OBs haben die Prioritätsklasse 1. Das entspricht der niedrigsten Priorität aller OBs. Das zyklische Programm kann also von Ereignissen jeder anderen Ereignisklasse unterbrochen werden.

Programmieren zyklischer Programmbearbeitung

Sie programmieren die zyklische Programmbearbeitung, indem Sie Ihr Anwenderprogramm in die Zyklus-OBs und die darin aufgerufenen Bausteine schreiben.

Die erste zyklische Programmbearbeitung beginnt, sobald das Anlaufprogramm fehlerfrei beendet wurde. Nach dem Ende jeder zyklischen Programmbearbeitung beginnt der Zyklus erneut.

Ablauf der zyklischen Programmbearbeitung

Ein Zyklus der Programmbearbeitung umfasst folgende Schritte:

1. Das Betriebssystem startet die Zyklusüberwachungszeit.
2. Das Betriebssystem schreibt die Werte aus dem Prozessabbild der Ausgänge in die Ausgabebaugruppen.
3. Das Betriebssystem liest den Zustand der Eingänge an den Eingabebaugruppen aus und aktualisiert das Prozessabbild der Eingänge.
4. Das Betriebssystem bearbeitet das Anwenderprogramm und führt die im Programm angegebenen Operationen aus.
5. Am Ende eines Zyklus kann das Betriebssystem weitere anstehende Aufgaben ausführen, z. B. Laden und Löschen von Bausteinen oder Aufruf anderer Zyklus-OBs.
6. Anschließend kehrt die CPU zum Zyklusanfang zurück und startet die Zykluszeitüberwachung erneut.

Siehe auch: Prozessabbild der Ein- und Ausgänge (Seite 1376)

Unterbrechungsmöglichkeiten

Die zyklische Programmbearbeitung kann durch folgende Ereignisse unterbrochen werden:

- Alarm
- STOP-Befehl, ausgelöst durch
  - Bedienung vom Programmiergerät
  - Anweisung "STP"
- Netzspannungsausfall
- Auftreten eines Geräte- oder Programmfehlers

Startinformation

- keine
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
first_scan	BOOL	= TRUE im ersten Aufruf dieses OB bei: • Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
remanence	BOOL	= TRUE, falls remanente Daten verfügbar sind

Siehe auch

Ereignisse und OBs (Seite 1380)

Organisationsbausteine für die alarmgesteuerte Programmbearbeitung

Uhrzeitalarm-OBs

Funktion

Organisationsbausteine für Uhrzeitalarme (OB-Nummern >=123) können wie folgt bearbeitet werden:

- Einmalig zu einem vorgegebenen Zeitpunkt (Datum mit Uhrzeit)
- Periodisch mit vorgegebenem Startzeitpunkt und den folgenden Zeitabständen:
  - Minütlich
  - Stündlich
  - Täglich
  - Wöchentlich
  - Monatlich
  - Jährlich
  - Am Monatsende

Uhrzeitalarm-OBs werden daher verwendet, um Teile des Anwenderprogramms zeitgesteuert ablaufen zu lassen.

Zustände für Uhrzeitalarme

Die folgende Tabelle enthält die möglichen Zustände eines Uhrzeitalarms sowie deren Bedeutung.

Zustand	Bedeutung
Storniert	Die einmalige Bearbeitung ist bereits erfolgt, oder das Startereignis eines noch nicht bearbeiteten Uhrzeitalarms wurde mit der erweiterten Anweisung CAN_TINT gelöscht.
Gestellt	Sie haben den Zeitpunkt bzw. Startzeitpunkt seiner Bearbeitung festgelegt.
Aktiviert	Sie haben festgelegt, ob er einmalig oder periodisch bearbeitet werden soll, und haben im Falle periodischer Bearbeitung den Zeitabstand festgelegt.

Regeln für Uhrzeitalarme

Für die Verwendung von Uhrzeitalarmen gelten folgende Regeln:

- Ein Uhrzeitalarm kann nur bearbeitet werden, wenn er gestellt und aktiviert wurde und ein zugehöriger Organisationsbaustein im Anwenderprogramm vorhanden ist.
- Die Startzeitpunkte periodischer Uhrzeitalarme müssen einem realen Datum entsprechen. Zum Beispiel ist die monatliche Wiederholung eines Organisationsbausteins, dessen erstmalige Bearbeitung am 31. Januar erfolgt, nicht möglich. In diesem Fall würde der OB nur in den Monaten gestartet, die 31 Tage haben.

- Ein Uhrzeit-Alarm, der während des Anlaufs durch Aufruf der erweiterten Anweisung ACT_TINT aktiviert wird, wird erst nach der Beendigung des Anlaufs bearbeitet.
- Nach jedem Anlauf der CPU müssen Sie gestellte Uhrzeitalarme neu aktivieren.

Stellen und Aktivieren eines Uhrzeitalarms-OB

Bevor ein Uhrzeitalarm ausgelöst und dadurch der zugeordnete Uhrzeitalarm-OB vom Betriebssystem bearbeitet werden kann, müssen Sie den Alarm stellen und aktivieren. Dabei gibt es folgende Möglichkeiten:

Uhrzeitalarm stellen	Uhrzeitalarm aktivieren
Mittels Projektierung	Mittels Projektierung
Mittels Projektierung	Durch Aufruf der erweiterten Anweisung ACT_TINT
Durch Aufruf der erweiterten Anweisung SET_TINTL	Durch Aufruf der erweiterten Anweisung ACT_TINT

Hinweis

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB einmalig bearbeitet werden soll, darf der Startzeitpunkt nicht in der Vergangenheit liegen (bezogen auf die Echtzeituhr der CPU).

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB periodisch bearbeitet werden soll, der Startzeitpunkt aber in der Vergangenheit liegt, dann wird der Uhrzeitalarm-OB bei dem nächsten fälligen Zeitpunkt nach der aktuellen Uhrzeit bearbeitet.

Abfragen des Status eines Uhrzeitalarms

Um den Status eines Uhrzeitalarms abzufragen, rufen Sie die erweiterte Anweisung QRY_TINT auf.

Stornieren eines Uhrzeitalarms

Sie können Uhrzeitalarme, die noch nicht bearbeitet wurden, mit der erweiterten Anweisung CAN_TINT stornieren.

Stornierte Uhrzeitalarme können Sie mit der erweiterten Anweisung SET_TINTL wieder stellen und mit der erweiterten Anweisung ACT_TINT aktivieren.

Bedingungen, die Uhrzeitalarm-OBs beeinträchtigen

Da ein Uhrzeitalarm nur in bestimmten Intervallen auftritt, können bestimmte Bedingungen die Funktion der zugehörigen OBs während der Bearbeitung Ihres Programms beeinträchtigen. Die folgende Tabelle zeigt einige dieser Bedingungen und beschreibt die Wirkung auf die Bearbeitung eines Uhrzeitalarm-OB.

Bedingung	Ergebnis
Im Anwenderprogramm wird die erweiterte Anweisung CAN_TINT aufgerufen.	Das Betriebssystem löscht das Startereignis (Datum und Uhrzeit) des Uhrzeitalarms. Sie müssen den Uhrzeitalarm erneut stellen und aktivieren, wenn der zugehörige Uhrzeitalarm-OB wieder aufgerufen werden soll.
Beim Synchronisieren oder Korrigieren der Systemuhr der CPU wurde die Uhrzeit vorgestellt. Dabei wurde der Startzeitpunkt für einen Uhrzeitalarm-OB übersprungen.	Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf und trägt das Startereignis, die Nummer und die Priorität des ersten übersprungenen Uhrzeitalarm-OB in dessen Startinformation ein. Nach Beendigung des OB 80 bearbeitet das Betriebssystem den übersprungenen Uhrzeitalarm-OB genau einmal.
Beim Synchronisieren oder Korrigieren der Systemuhr der CPU wurde die Uhrzeit zurückgestellt. Die korrigierte Uhrzeit liegt vor dem Startzeitpunkt eines bereits ausgeführten Uhrzeitalarm-OB.	Der Uhrzeitalarm-OB wird wiederholt.
Ein Uhrzeitalarm-OB wird noch bearbeitet, als das Startereignis für seine nächste Bearbeitung auftritt.	Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf. Nach dessen Bearbeitung und der Weiterbearbeitung des aktuellen Uhrzeitalarm-OB wird die angeforderte OB-Bearbeitung nachgeholt.

Startinformation

Ein Uhrzeitalarm-OB hat die folgende Startinformation:

Variable	Datentyp	Beschreibung
CaughtUp	BOOL	= 1, falls der OB-Aufruf wegen vorgestellter Uhrzeit nachgeholt wird
SecondTime	BOOL	= 1, falls der OB wegen zurückgestellter Uhrzeit ein zweites Mal aufgerufen wird (genauer: falls der geplante Zeitpunkt der aktuellen OB-Bearbeitung früher oder gleich dem geplanten Zeitpunkt der vorhergehenden OB-Bearbeitung liegt). Hinweis: SecondTime wird nur einmal gesetzt.

Statusalarm-OB

Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den Statusalarm-OB auf, wenn es einen Statusalarm von einem DP-Master oder IO-Controller empfangen hat. Dies kann der Fall sein, wenn ein Modul eines Slaves seinen Betriebszustand wechselt, z. B. von "RUN" nach "STOP". Genauere Informationen zu den Ereignissen, die einen Statusalarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

Der Statusalarm-OB hat die folgende Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Adresse der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm

Siehe auch

Ereignisse und OBs (Seite 1380)

Update-Alarm-OB

Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den Update-Alarm-OB auf, wenn es einen Update-Alarm von einem DP-Master oder IO-Controller empfangen hat. Dies kann der Fall sein, wenn Sie an einem Steckplatz eines Slaves oder Devices eine Parameteränderung durchgeführt haben. Genauere Informationen zu den Ereignissen, die einen Update-Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

Der Update-Alarm-OB hat die folgende Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Adresse der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm

Siehe auch

Ereignisse und OBs (Seite 1380)

OB für hersteller- bzw. profilspezifische Alarme

Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den OB57 auf, wenn es einen hersteller- oder profilspezifischen Alarm von einem DP-Master oder IO-Controller empfangen hat. Genauere Informationen zu den Ereignissen, die einen solchen Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

Struktur der Startinformation

Der OB für hersteller- bzw. profilspezifische Alarme hat die folgende Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Adresse der Alarm auslösenden Komponente
Slot	UINT	Steckplatz-Nr. der Alarm auslösenden Komponente
Specifier	WORD	Alarm Specifier aus dem Alarm-Telegramm

Siehe auch

Ereignisse und OBs (Seite 1380)

Verzögerungsalarm-OBs

Beschreibung

Ein Verzögerungsalarm-OB wird nach einer parametrierbaren Verzögerungszeit vom Betriebssystem gestartet. Die Verzögerungszeit beginnt nach Aufruf der Anweisung SRT_DINT abzulaufen.

Sie können bis zu vier Verzögerungsalarm-OBs oder Weckalarm-OBs in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Weckalarm-OBs, können Sie noch maximal zwei Verzögerungsalarm-OBs in Ihr Programm einfügen.

Sie können die Bearbeitung eines Verzögerungsalarms, der noch nicht gestartet wurde, mit der Anweisung CAN_DINT verhindern.

Funktionsweise der Verzögerungsalarm-OBs

Nach Ablauf der Verzögerungszeit, die Sie zusammen mit einer OB-Nummer und einer Kennung an die Anweisung SRT_DINT übergeben haben, startet das Betriebssystem den zugehörigen OB.

Wenn Sie einen Verzögerungsalarm-OB verwenden möchten, dann müssen Sie die folgenden Aufgaben ausführen:

- Sie müssen die Anweisung SRT_DINT aufrufen.
- Sie müssen den Verzögerungsalarm-OB als Teil Ihres Programms in die CPU laden.

Die Verzögerungszeit wird mit einer Genauigkeit von 1 ms gemessen. Eine Verzögerungszeit kann sofort nach ihrem Ablauf wieder gestartet werden.

Verzögerungsalarm-OBs werden nur bearbeitet, wenn die CPU im Betriebszustand "RUN" ist. Ein Neustart (Warmstart) löscht alle Startereignisse von Verzögerungsalarm-OBs.

Das Betriebssystem ruft den Verzögerungsalarm-OB auf, wenn eines der folgenden Ereignisse auftritt:

- Wenn das Betriebssystem einen OB zu starten versucht, der nicht geladen ist und dessen Nummer Sie beim Aufruf der Anweisung SRT_DINT angegeben haben.
- Wenn das nächste Startereignis für einen Verzögerungsalarm auftritt, bevor die Bearbeitung des zugehörigen Verzögerungsalarm-OB beendet ist.

Sie können Verzögerungsalarme mit Hilfe der Anweisungen DIS_AIRT und EN_AIRT sperren und wieder freigeben.

Hinweis

Wenn Sie nach der Ausführung von SRT_DINT ein Alarm mit DIS_AIRT sperren, wird dieser Alarm erst nach der Freigabe mit EN_AIRT ausgeführt. Die Verzögerungszeit verlängert sich entsprechend.

Startinformation

- keine
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
sign	WORD	Anwenderkennung: Eingangsparameter SIGN aus dem Aufruf der Anweisung "SRT_DINT"

Hinweis

Variable "sign"

Die Variable "sign" gibt es erst seit der Firmware-Version V4.1 der S7-1200-CPUs. Siehe Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen (Seite 1456)

Siehe auch

Ereignisse und OBs (Seite 1380)

Weckalarm-OBs

Beschreibung

Weckalarm-OBs dienen dazu, Programme in periodischen Zeitabständen unabhängig von der zyklischen Programmbearbeitung zu starten. Die Startzeitpunkte eines Weckalarm-OBs werden über das Zeitraster und die Phasenverschiebung angegeben.

Das Zeitraster definiert dabei die Abstände, in denen der Weckalarm-OB gestartet wird und ist ein ganzzahliges Vielfaches des Grundtakts von 1 ms. Die Phasenverschiebung ist die Zeit, um die der Startzeitpunkt gegenüber dem Grundtakt verschoben wird. Diese Verschiebung können Sie bei der Verwendung mehrerer Weckalarm-OBs dazu nutzen, einen gleichzeitigen

Startzeitpunkt zu verhindern, falls die Zeitraster der Weckalarm-OBs ein gemeinsames Vielfaches aufweisen.

Als Zeitraster können Sie eine Zeitdauer zwischen 1 ms und 60000 ms vorgeben.

Sie können bis zu vier Weckalarm-OBs oder Verzögerungsalarm-OBs in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Verzögerungsalarm-OBs, können Sie noch maximal zwei Weckalarm-OBs in Ihr Programm einfügen.

Hinweis

Die Laufzeit jedes Weckalarm-OB muss deutlich kleiner sein als sein Zeitraster. Falls ein Weckalarm-OB noch nicht beendet ist, aber wegen des abgelaufenen Zeittakts erneut zur Bearbeitung ansteht, wird der Zeitfehler-OB gestartet. Anschließend wird der Fehler verursachende Weckalarm nachgeholt oder verworfen.

Beispiel für die Verwendung der Phasenverschiebung

Sie haben in Ihr Programm zwei Weckalarm-OBs eingefügt:

- Weckalarm-OB1
- Weckalarm-OB2

Für Weckalarm-OB1 haben Sie ein Zeitraster von 20 ms und für Weckalarm-OB2 ein Zeitraster von 100 ms eingestellt. Nach Ablauf des Zeitrasters von 100 ms erreicht Weckalarm-OB1 zum fünften Mal seinen Startzeitpunkt, Weckalarm-OB2 zum ersten Mal. Um die Weckalarm-OBs dennoch zeitversetzt zu bearbeiten, geben Sie für einen der beiden Weckalarm-OBs eine Phasenverschiebung ein.

Startinformation

- keine
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
first_scan	BOOL	= TRUE im ersten Aufruf dieses OB • Beim Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
event_count	INT	Anzahl der verlorenen Startereignisse seit dem letzten Start dieses OB

Siehe auch

Weckalarm-OBs parametrieren (Seite 1453)

Ereignisse und OBs (Seite 1380)

Prozessalarm-OBs

Beschreibung

Sie können Prozessalarm-OBs einsetzen, um auf bestimmte Ereignisse zu reagieren. Dabei können Sie einem Ereignis, das einen Alarm auslöst, nur genau einen Prozessalarm-OB zuordnen. Einem Prozessalarm-OB können hingegen mehrere Ereignisse zugeordnet sein.

Prozessalarme können durch Schnelle Zähler und Eingangskanäle ausgelöst werden. Für jeden Schnellen Zähler und Eingangskanal, der einen Prozessalarm auslösen soll, ist die Parametrierung der folgenden Eigenschaften erforderlich:

- Das Prozessereignis, das den Prozessalarm auslösen soll (z. B. die Änderung der Zählrichtung eines Schnellen Zählers)
- Die Nummer des Prozessalarm-OB, der diesem Prozessereignis zugeordnet wird

Sie können bis zu 50 voneinander unabhängige Prozessalarm-OBs (OB-Nummern >= 123) in Ihrem Programm verwenden.

Funktionsweise eines Prozessalarm-OB

Nach Auslösen eines Prozessalarms identifiziert das Betriebssystem den Kanal des Eingangs oder den Schnellen Zähler und ermittelt den zugeordneten Prozessalarm-OB.

Wenn kein anderer Alarm-OB aktiv ist, wird der ermittelte Prozessalarm-OB aufgerufen. Wird bereits ein anderer Alarm-OB ausgeführt, wird der Prozessalarm in die Warteschlange seiner Prioritätsklasse eingereiht. Die Quittierung des Prozessalarms erfolgt nach Beendigung des zugeordneten Prozessalarm-OB.

Wenn in der Zeit zwischen der Identifikation und der Quittierung eines Prozessalarms auf demselben Modul erneut ein Prozessereignis auftritt, dann gilt:

- Tritt das Ereignis bei dem Kanal auf, der den aktuellen Prozessalarm ausgelöst hat, wird kein weiterer Prozessalarm ausgelöst. Ein weiterer Prozessalarm kann erst dann ausgelöst werden, wenn der aktuelle Prozessalarm quittiert ist.
- Tritt das Ereignis bei einem anderen Kanal auf, wird ein Prozessalarm ausgelöst.

Prozessalarm-OBs werden nur im Betriebszustand "RUN" der CPU aufgerufen.

Startinformation

- keine
- Optimierte Startinformation:

Name	Datentyp	Bedeutung
Laddr	HW_IO	HW-Kennung des Moduls, das den Prozessalarm auslöst
USI	WORD	Kennung für zukünftige Erweiterungen (nicht anwenderrelevant)

Name	Datentyp	Bedeutung
IChannel	USINT	Nummer des Kanals, der den Prozessalarm auslöst
EventType	BYTE	Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke) Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen.

Siehe auch

Prozessalarm-OBs parametrieren (Seite 1454)

Ereignisse und OBs (Seite 1380)

Zeitfehler-OB

Beschreibung

Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf, wenn eines der folgenden Ereignisse auftritt:

- Das zyklische Programm überschreitet die Zyklusüberwachungszeit.
- Der aufgerufene OB wird derzeit noch ausgeführt (möglich bei Verzögerungsalarm-OBs und Weckalarm-OBs).
- Ein Uhrzeitalarm wurde verpasst wegen Vorstellen der Uhrzeit um mehr als 20 Sekunden.
- Ein Uhrzeitalarm wurde verpasst während STOP.
- Eine Warteschlange für Alarm-OBs ist übergelaufen.
- Ein Alarm geht durch zu hohe Alarmlast verloren.

Wenn Sie keinen Zeitfehler-OB programmiert haben, verhält sich die S7-1200-CPU wie folgt:

- CPUs mit Firmware-Stand V1.x bis V3.x: Die CPU bleibt im RUN.
- CPUs ab Firmware-Stand V4.x:
  - Die CPU geht beim Überschreiten der Zyklusüberwachungszeit in STOP.
  - Bei allen anderen Startereignissen des Zeitfehler-OB bleibt die CPU im RUN.

Das zweimalige Überschreiten der Zyklusüberwachungszeit führt nicht zum Aufruf eines OB, sondern zum STOP der CPU. Sie können die zweite Überschreitung vermeiden, indem Sie mit der Anweisung RE_TRIGR die Zyklusüberwachung der CPU neu starten.

Sie können nur einen Zeitfehler-OB in Ihrem Programm verwenden.

Startinformation

Der Zeitfehler-OB hat die folgende Startinformation:

Variable	Datentyp	Beschreibung
fault_id	BYTE	• 0x01: Zyklusüberwachungszeit überschritten • 0x02: Aufgerufener OB wird noch ausgeführt • 0x05: Abgelaufener Uhrzeitalarm durch Uhrzeitsprung • 0x06: Abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN • 0x07: Warteschlange übergelaufen • 0x09: Alarmverlust durch zu hohe Alarmlast
csg_OBnr	OB_ANY	Nummer des zum Fehlerzeitpunkt bearbeiteten OB
csg_prio	UINT	Priorität des zum Fehlerzeitpunkt bearbeiteten OB

Siehe auch

Ereignisse und OBs (Seite 1380)

Zykluszeit und Zyklusüberwachungszeit (Seite 1396)

Diagnosealarm-OB

Beschreibung

Sie können für diagnosefähige Module den Diagnosealarm freigeben, damit das Modul Änderungen am Peripheriezustand erkennt. Dadurch löst das Modul in folgenden Fällen einen Diagnosealarm aus:

- Es liegt eine Störung vor (kommendes Ereignis)
- Es liegt keine Störung mehr vor (gehendes Ereignis)

Wenn kein anderer Alarm-OB aktiv ist, dann wird der Diagnosealarm-OB (OB 82) aufgerufen. Wird bereits ein anderer Alarm-OB ausgeführt, wird der Diagnosealarm in die Warteschlange seiner Prioritätsgruppe eingereiht.

Sie können nur einen Diagnosealarm-OB in Ihrem Programm verwenden.

Startinformation

Der Diagnosealarm-OB hat die folgende Startinformation:

Variable	Datentyp	Beschreibung
IO_state	WORD	Enthält den Peripheriezustand des diagnose-fähigen Moduls.
laddr	HW_ANY	HW-Identifikation
channel	UINT	Kanalnummer
multi_error	BOOL	= 1, falls mehr als ein Fehler vorhanden ist

Variable IO_state

Die folgende Tabelle zeigt die möglichen Peripheriezustände, die in der Variable IO_state enthalten sein können:

IO_state	Beschreibung
Bit 0	Konfiguration korrekt: = 1, falls die Konfiguration korrekt ist = 0, falls die Konfiguration nicht mehr korrekt ist
Bit 4	Fehler: = 1, falls eine Störung vorliegt, z. B. ein Drahtbruch = 0, falls die Störung nicht mehr vorliegt
Bit 5	Konfiguration nicht korrekt: = 1, falls die Konfiguration nicht korrekt ist = 0, falls die Konfiguration wieder korrekt ist
Bit 7	Peripherie nicht erreichbar: = 1, falls ein Peripheriezugriffsfehler aufgetreten ist In diesem Fall enthält laddr die Hardwarekennung der Peripherie mit dem Zugriffsfehler. = 0, falls die Peripherie wieder erreichbar ist
Bits 8 bis 15	reserviert (immer 0)

Siehe auch

Ereignisse und OBs (Seite 1380)

Ziehen/Stecken-OB

Beschreibung

Das Betriebssystem der S7 1200-CPU ruft den Ziehen/Stecken-OB (OB 83) auf, wenn ein projektiertes und nicht deaktiviertes Modul oder Submodul der dezentralen Peripherie (PROFIBUS, PROFINET, AS-i) gezogen oder gesteckt wird.

Hinweis

Das Ziehen bzw. Stecken eines zentralen Moduls führt zum STOP der CPU.

Startinformation

Der Ziehen/Stecken-OB hat die folgende Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung des betroffenen Moduls bzw. Submoduls
Event_Class	BYTE	• B#16#38: (Sub-) Modul gesteckt • B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar
Fault_ID	BYTE	Fehlercode (mögliche Werte: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58)

Die folgende Tabelle zeigt, welches Ereignis den Start des Ziehen/Stecken-OB verursacht hat.

ev_class (B#16# ...)	fault_id (B#16# ...)	Bedeutung
39	51	Modul gezogen
39	54	Submodul gezogen
38	54	Submodul gesteckt, das dem parametriertem Submodul entspricht
38	55	Submodul gesteckt, das dem parametrierten Submodul nicht entspricht
38	56	Submodul gesteckt, jedoch Fehler bei Parametrierung
38	57	Submodul oder Modul gesteckt, jedoch mit Störung oder Wartung
38	58	Submodul-Zugriffsfehler beseitigt

Siehe auch

Ereignisse und OBs (Seite 1380)

Baugruppenträgerfehler-OB

Beschreibung

Das Betriebssystem der S7 1200-CPU ruft den OB86 in den folgenden Fällen auf:

- Der Ausfall eines DP-Mastersystems bzw. eines PROFINET IO-Systems wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines DP-Slaves bzw. eines IO-Devices wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines Teils der Submodule eines PROFINET I-Device wird erkannt.

Struktur der Startinformation

Der Baugruppenträgerfehler-OB hat die folgende Startinformation:

Name	Datentyp	Bedeutung
LADDR	HW_IO	Hardware-Kennung des fehlerhaften Hardware-Objekts
Event_Class	BYTE	• B#16#32: Aktivierung eines DP-Slaves bzw. eines IO-Devices • B#16#33: Deaktivierung eines DP-Slaves bzw. eines IO-Devices • B#16#38: gehendes Ereignis • B#16#39: kommendes Ereignis
Fault_ID	BYTE	Fehlercode (mögliche Werte: B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)

Die folgende Tabelle zeigt, welches Ereignis den Start des OB86 verursacht hat.

Ev_class B#16# ...	Fault_id B#16# ...	Bedeutung
39	C3	Ausfall eines DP-Mastersystems
39/38	C4	Ausfall/Wiederkehr eines DP-Slaves
38	C5	Wiederkehr eines DP-Slaves, Slave jedoch gestört
38	C6	Erweiterungsgerätewiederkehr, jedoch Fehler bei Baugruppenparametrierung
38	C7	Wiederkehr eines DP-Slaves, jedoch Fehler bei Baugruppenparametrierung
38	C8	Wiederkehr einer DP-Slaves, jedoch Abweichung Soll-/Istausbau
32/33	C9	Aktivierung/Deaktivierung eines DP-Slaves mit der Anweisung "D_ACT_DP"
39	CA	Ausfall eines PROFINET IO-Systems
39/38	CB	Ausfall/Wiederkehr eines PROFINET IO-Devices
38	CC	Wiederkehr eines PROFINET IO-Devices mit Störung oder Wartung
38	CD	Wiederkehr eines PROFINET IO-Devices, Sollausbau weicht von Istausbau ab
38	CE	Wiederkehr eines PROFINET IO-Devices, Fehler bei der Baugruppenparametrierung
32/33	CF	Aktivierung/Deaktivierung eines IO-Devices mit der Anweisung "D_ACT_DP"
39/38	F8	Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Devices
38	F9	Wiederkehr eines Teils der Submodule eines PROFINET I-Devices mit Device-Konfigurationsunterschied

Siehe auch

Ereignisse und OBs (Seite 1380)

MC-Servo-OB

Beschreibung

Wenn Sie ein Technologieobjekt mit Antriebsanbindung über PROFIdrive/Analogausgang anlegen, wird automatisch der Organisationsbaustein "MC-Servo [OB91]" angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Motion Control-Applikationszyklus aufgerufen.

Der MC-Servo-OB ist schreibgeschützt. Der Inhalt lässt sich nicht verändern.

Innerhalb des MC-Servo-OB werden die Algorithmen der Lageregelung aller auf der CPU projektierten Technologieobjekte für Motion Control berechnet.

Der Applikationszyklus und die Priorität des Organisationsbausteins lassen sich entsprechend Ihren Anforderungen an die Regelgüte und die Systembelastung einstellen.

Applikationszyklus

Den Applikationszyklus, in dem der MC-Servo-OB aufgerufen wird, lässt sich in den Eigenschaften des Organisationsbausteins einstellen.

Der MC-Servo-OB wird zyklisch mit dem angegebenen Applikationszyklus aufgerufen.

Den Applikationszyklus müssen Sie groß genug wählen, um alle Technologieobjekte in einem Zyklus zu bearbeiten. Wenn die Bearbeitungszeit der Technologieobjekte größer ist als der Applikationszyklus, treten Überläufe auf.

Um Störungen im Programmablauf auf der CPU zu vermeiden, stellen Sie den Applikationszyklus abhängig von der Anzahl der eingesetzten Achsen folgendermaßen ein:

Applikationszyklus = Anzahl der Achsen × 2 ms

Folgende Tabelle zeigt den resultierenden Applikationszyklus beispielhaft entsprechend der Anzahl der Achsen:

Anzahl der Achsen	Applikationszyklus
1	2 ms
2	4 ms
4	8 ms
8	16 ms

Der Antrieb SINAMICS G120 aktualisiert das Antriebsprozessbild alle 4 ms. Um die Regelung zu verbessern, stellen Sie den Applikationszyklus des MC-Servo-OB auf 4 ms oder ein ganzes Vielfaches von 4 ms.

Siehe auch

MC-PreServo-OB (Seite 1449)

MC-PostServo-OB (Seite 1449)

MC-Interpolator-OB (Seite 1450)

MC-Servo-OB parametrieren (Seite 1455)

MC-PreServo-OB

Beschreibung

Der Organisationsbaustein MC-PreServo [OB67] ist programmierbar und wird in dem Applikationszyklus aufgerufen, der am MC-Servo-OB (Seite 1448) konfiguriert ist. MC-PreServo [OB67] wird direkt vor dem MC-Servo [OB91] aufgerufen.

Über den Organisationsbaustein lässt sich der konfigurierte Applikationszyklus (Angabe in µs) auslesen.

Struktur der Startinformation

Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB beim Übergang von STOP nach RUN
PIP_Input	BOOL	TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell
PIP_Output	BOOL	TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen
IO_System	USINT	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
Event_Count	INT	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).
Synchronous	BOOL	Reserviert
CycleTime	UDINT	Anzeige des am MC-Servo-OB konfigurierten Applikationszyklus in µs

MC-PostServo-OB

Beschreibung

Der Organisationsbaustein MC-PostServo [OB95] ist programmierbar und wird in dem Applikationszyklus aufgerufen, der am MC-Servo-OB (Seite 1448) konfiguriert ist. MC-PostServo [OB95] wird direkt nach dem MC-Servo [OB91] aufgerufen.

Über den Organisationsbaustein lässt sich der konfigurierte Applikationszyklus (Angabe in µs) auslesen.

Struktur der Startinformation

Optimierte Startinformation:

Name	Datentyp	Bedeutung
Initial_Call	BOOL	=TRUE im ersten Aufruf dieses OB beim Übergang von STOP nach RUN
PIP_Input	BOOL	TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell

Name	Datentyp	Bedeutung
PIP_Output	BOOL	TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen
IO_System	USINT	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
Event_Count	INT	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).
Synchronous	BOOL	Reserviert
CycleTime	UDINT	Anzeige des am MC-Servo-OB konfigurierten Applikationszyklus in µs

MC-Interpolator-OB

Beschreibung

Wenn Sie ein Technologieobjekt mit Antriebsanbindung über PROFIdrive/Analogausgang anlegen, wird automatisch der Organisationsbaustein "MC-Interpolator [OB91]" angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Motion Control-Applikationszyklus des MC-Servo-OB (Seite 1448) aufgerufen.

Der MC-Interpolator-OB ist schreibgeschützt. Der Inhalt lässt sich nicht verändern.

Innerhalb des MC-Interpolator-OB erfolgt die Auswertung der Motion Control-Anweisungen sowie die Überwachung und die Sollwertgenerierung für alle auf der CPU projektierten Technologieobjekte (Motion Control).

Der MC-Interpolator-OB wird am Ende der Bearbeitung des MC-Servo-OB aufgerufen. Das Taktverhältnis zwischen MC-Interpolator-OB und MC-Servo-OB ist immer 1:1.

Ablaufverhalten

Den Applikationszyklus des MC-Servo-OB müssen Sie groß genug wählen, um alle Technologieobjekte für Motion Control in einem Applikationszyklus zu bearbeiten. Wenn der Applikationszyklus nicht eingehalten wird, treten Überläufe auf. Beachten Sie bei Überläufen folgendes Verhalten:

- Die CPU toleriert maximal drei aufeinander folgende Überläufe des MC-Interpolator-OB.
- Die Bearbeitung eines MC-Interpolator-OB darf maximal durch einen Aufruf des MC-Servo-OB unterbrochen werden.

Bausteinparameter von Organisationsbausteinen

Grundlagen zu Bausteinparametern

Einführung

Einige Organisationsbausteine (OBs) besitzen Eigenschaften, über die ihr Verhalten oder ihre Zuordnung zu bestimmten Ereignissen gesteuert werden kann. Diese Eigenschaften können Sie durch Parametrierung beeinflussen.

Übersicht

Für folgende Organisationsbausteine können Sie die Eigenschaften parametrieren:

- Uhrzeitalarm-OBs
- Weckalarm-OBs
- Prozessalarm-OBs
- MC-Servo-OB

Siehe auch

Prozessalarm-OBs parametrieren (Seite 1454)

Weckalarm-OBs parametrieren (Seite 1453)

Uhrzeitalarm-OBs parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines Uhrzeitalarm-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Uhrzeitalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Uhrzeitalarm".

Überblick über die einstellbaren Parameter

Sie können folgende Parameter einstellen:

- Ausführung
- Startdatum und Uhrzeit
- Optionsfelder "Lokalzeit" und "Systemzeit"

Parameter "Ausführung"

Über die Klappliste "Ausführung" geben Sie vor, mit welcher Periode der Uhrzeitalarm -OB ausgeführt werden soll. Die Zeitintervalle beziehen sich auf die Einstellungen bei "Startdatum" und "Uhrzeit".

Folgende Werte von "Ausführung" sind möglich:

- nie
- einmalig
- minütlich
- stündlich
- täglich
- wöchentlich
- monatlich
- jährlich
- am Monatsende

Hinweis

Beim Wert "am Monatsende" ist der unter "Startdatum" angegebene Wert irrelevant.

Parameter "Startdatum" und "Uhrzeit"

Hier stellen Sie ein, zu welchem Zeitpunkt der Uhrzeitalarm-OB zum ersten Mal ausgeführt werden soll.

Beispiel: Startdatum = 05.07.2013, Uhrzeit =11:16

Abhängig vom Wert des Parameters "Ausführung" erzeugt die CPU periodisch weitere Uhrzeitalarme. Die Startzeit bezieht sich je nach Einstellung entweder auf die Lokalzeit oder auf die koordinierte Weltzeit UTC.

Hinweis

Wenn Sie für den Parameter "Ausführung" "monatlich" einstellen, sind der 29., der 30. und der 31. als Startdatum nicht möglich. Wenn der Uhrzeitalarm-OB am Monatsende gestartet werden soll, wählen Sie stattdessen "am Monatsende" beim Parameter "Ausführung".

"Lokalzeit" oder "Systemzeit"

Hier stellen Sie ein, auf welche Zeit sich die Startzeit des Uhrzeitalarm-OB bezieht:

- "Lokalzeit": Die Startzeit bezieht sich auf die eingestellte Zeitzone der CPU.
- "Systemzeit": Die Startzeit bezieht sich auf die koordinierte Weltzeit UTC (Universal Time Coordinated).

Weckalarm-OBs parametrieren

Einführung

Mit Weckalarm-OBs können Sie Programme in gleichmäßigen Zeitabständen starten. Dazu müssen Sie für jeden verwendeten Weckalarm-OB ein Zeitraster und eine Phasenverschiebung eingeben.

Sie können bis zu vier Weckalarm-OBs oder Verzögerungsalarm-OBs (OB-Nummern >= 200) in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Verzögerungsalarm-OBs, können Sie noch maximal zwei Weckalarm-OBs in Ihr Programm einfügen.

Hinweis

Wenn Sie mehrere Weckalarm-OBs parametrieren, müssen Sie jedem Weckalarm-OB eine andere Zykluszeit oder Phasenverschiebung geben, um einer gleichzeitigen Ausführung bzw. einer Warteschlange vorzubeugen. Beim Erstellen eines Weckalarm-OB wird die Zykluszeit 100 und die Phasenverschiebung 0 als Startwert eingetragen.

Vorgehen

Um für einen Weckalarm-OB ein Zeitraster und eine Phasenverschiebung einzugeben, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Weckalarm-OB.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Dialog "<Name des Weckalarm-OB>" wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf die Gruppe "Weckalarm".
   Die Eingabefelder für das Zeitraster und die Phasenverschiebung werden angezeigt.
5. Geben Sie das Zeitraster und die Phasenverschiebung ein.
6. Bestätigen Sie Ihre Eingaben mit "OK".

Siehe auch

Grundlagen zu Bausteinparametern (Seite 1451)

Weckalarm-OBs (Seite 1440)

Prozessalarm-OBs parametrieren

Einführung

Für jeden Eingangskanal und Schnellen Zähler, der einen Prozessalarm auslösen soll, müssen Sie das entsprechende Ereignis aktivieren und folgende Parameter vergeben:

- Ereignisnamen
- Nummer des Prozessalarm-OB, der diesem Prozessereignis zugeordnet wird

Die Parametrierung von Prozessalarmen erfolgt in den Eigenschaften des entsprechenden Geräts. Sie können bis zu 50 Prozessalarm-OBs parametrieren.

Den zu parametrierenden Prozessalarm-OB können Sie entweder vor oder während der Aktivierung eines Ereignisses erstellen.

Vorgehen

Um einen Prozessalarm-Ereignis zu projektieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Befehl "Geräte & Netze".
   Der Hardware- und Netzwerkeditor wird in der Netzsicht geöffnet.
2. Wechseln Sie zur Gerätesicht.
3. Wenn das Inspektorfenster in der Gerätesicht geschlossen ist, aktivieren Sie im Menü "Ansicht" das Optionskästchen "Inspektorfenster".
   Das Inspektorfenster wird geöffnet.
4. Klicken Sie auf das Register "Eigenschaften".
5. Selektieren Sie in der Gerätesicht die Baugruppe, für die Sie einen Prozessalarm parametrieren möchten.
6. Aktivieren Sie das entsprechende Ereignis, das einen Prozessalarm auslösen soll, z. B. eine steigende Flanke.

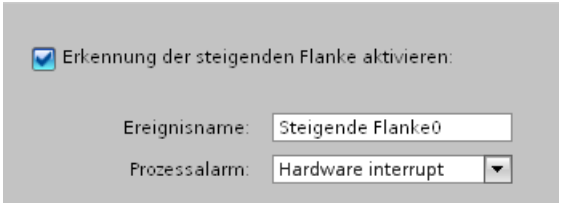

7. Geben Sie einen Ereignisnamen ein.

8. Wählen Sie aus der Klappliste "Prozessalarm" einen bestehenden Prozessalarm-OB aus oder legen Sie einen neuen Prozessalarm-OB an. Falls Sie zuvor noch keinen Prozessalarm-OB angelegt haben, klicken Sie in der Klappliste auf die Schaltfläche "Neuen Baustein hinzufügen".
   Die Startinformation des entsprechenden Prozessalarm-OBs mit allen Angaben zum alarmauslösenden Ereignis wird aktualisiert.

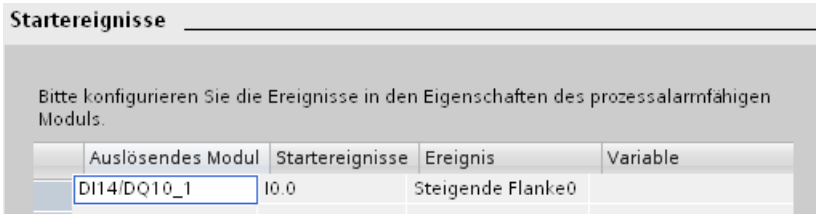

9. Wenn Sie weitere Prozessalarme parametrieren möchten, wiederholen Sie die Schritte 5 bis 8.

Für das Ereignis, das durch den eindeutigen Ereignisnamen gekennzeichnet ist, wird automatisch eine Systemkonstante vom Datentyp Event_HwInt angelegt. Die Systemkonstanten werden in der Standard-Variablentabelle angezeigt.

Siehe auch

Grundlagen zu Bausteinparametern (Seite 1451)

Prozessalarm-OBs (Seite 1442)

Ereignisse und OBs (Seite 1380)

MC-Servo-OB parametrieren

Vorgehen zur Einstellung der Parameter

Um die Parameter eines MC-Servo-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten MC-Servo-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Zykluszeit".

Überblick über die einstellbaren Parameter

In der Gruppe "Zykluszeit" wählen Sie aus, mit welchem Applikationszyklus der MC-Servo-OB im Ablaufsystem der SIMATIC S7-1200 zyklisch aufgerufen wird.

Im Feld "Applikationszyklus" stellen Sie den Zyklus ein, in dem der MC-Servo-OB aufgerufen wird.

Siehe auch

Grundlagen zu Bausteinparametern (Seite 1451)

MC-Servo-OB (Seite 1448)

Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen

CPUs und Engineering Software zum Projektieren der CPUs werden ständig weiterentwickelt, um die Leistungsfähigkeit und Sicherheit zu verbessern. Auf diese Weise entstehen neue Versionen, die im Zusammenspiel der Komponenten einige Besonderheiten aufweisen. Die folgenden Abschnitte beschreiben Besonderheiten der S7-1200-CPUs mit Firmware-Version V4 im Vergleich zu den Firmware-Versionen V1 bis V3.

Für eine detaillierte Gegenüberstellung des Funktionsumfangs lesen Sie das Systemhandbuch S7-1200 (Beschreibung neuer Anweisungen, neuer Organisationsbausteine und erweiterte Konfigurationsmöglichkeiten).

Erforderliche Engineering Software

S7-1200-CPUs V4 sind mit STEP 7 ab V13 projektierbar.

Kompatibilität zwischen Memory-Card-Inhalten und Firmware-Version der CPU

Memory Cards (Übertragungskarte oder Programmkarte) mit Konfiguration und Programm für eine S7-1200-CPU V1, V2 oder V3 funktionieren nicht in einer S7-1200-CPU V4.
Memory Cards mit Konfiguration und Programm für eine S7-1200-CPU V4 funktionieren nicht in einer S7-1200-CPU V1, V2 oder V3.

Sie müssen eine S7-1200-CPU-V1-V3-Konfiguration in eine S7-1200-CPU-V4-Konfiguration ändern (Gerätetausch) und anschließend in die CPU laden. Bei S7-1200-CPU V1-V2 ist ein stufenweiser Gerätetausch erforderlich (siehe unten).

Wenn Sie die Memory Card in eine CPU mit einer inkompatiblen Firmware-Version stecken, dann läuft die CPU nicht an. Wenn Sie eine Memory Card für eine CPU V1, V2 oder V3 in eine S7-1200-CPU V4 stecken, gibt diese CPU einen Versionsfehler aus.

Online-Gehen und Laden

Wenn Sie eine S7-1200-CPU mit der Firmware-Version V1, V2 oder V3 mit STEP 7 projektiert haben, dann muss die CPU, zu der Sie online gehen wollen bzw. die Sie laden wollen, ebenfalls eine dieser Firmware-Versionen haben. Sie können nicht mit einer projektierten S7-1200-CPU V1, V2 oder V3 zu einer S7-1200-CPU V4 online gehen.

Umgekehrt können Sie nicht mit einer projektierten S7-1200-CPU V4 zu einer S7-1200-CPU V1, V2 oder V3 online gehen oder diese CPU laden.

Bestehende CPU ersetzen

Sie können eine projektierte S7-1200-CPU V1, V2 oder V3 durch eine neue CPU mit einer Firmware-Version größer oder gleich 4 ersetzen. Im Fall einer vorliegenden S7-1200-CPU V1 oder V2 müssen Sie zunächst den Umweg über den Gerätetausch mit einer S7-1200-CPU V3 einschlagen: Ein direkter Tausch mit einer S7-1200-CPU V4 ist nicht möglich.

1. S7-1200-CPU V1 (V2) > S7-1200-CPU V3
2. S7-1200-CPU V3 > S7-1200-CPU V4

Einen Gerätetausch können Sie rückgängig machen (Befehl "Rückgängig" im Menü "Bearbeiten), solange Sie die Konfiguration nicht laden. Wenn Sie die Konfiguration für die

neue CPU-Firmware-Version ("V4-Konfiguration") in die CPU laden, ist es nicht mehr möglich, zu einer Version V3 zurückzukehren.

Speichern Sie daher das bestehende Projekt z. B. mit einer V3-Konfiguration als Projektarchiv, um später darauf zurückgreifen zu können.

Besonderheiten beim Gerätetausch (V3 > V4):

- Das Unterbrechungsverhalten von Alarm-OBs bleibt gleich; sie sind als nicht unterbrechbar parametriert. Das ist auch das voreingestellte Verhalten bei S7-1200-CPUs V1-V3. Bei S7-1200-V4-CPUs ist das Alarm-OB-Unterbrechungsverhalten parametrierbar. Wenn Sie eine S7-1200-CPU V4 direkt aus dem Hardware-Katalog in die Netzsicht ziehen, dann ist diese Option aktiviert (Alarm-OBs sind unterbrechbar)!
- Das Verhalten bei PUT/GET-Zugriff entfernter Partner bleibt gleich, der Zugriff ist erlaubt. Das ist auch das voreingestellte Verhalten bei S7-1200-CPUs V1-V3. Bei S7-1200-V4-CPUs ist der Zugriff über PUT/GET-Kommunikation durch entfernten Partner parametrierbar (Bereich "Schutz" der CPU-Parameter). Wenn Sie eine S7-1200-CPU V4 direkt aus dem Hardware-Katalog in die Netzsicht ziehen, dann ist der Zugriff nicht erlaubt und muss explizit freigegeben werden!
- Das Wording für die Schutzstufen ändert sich, die Wirkung der Einstellungen bleibt aber gleich. Zusätzlich ist die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" wählbar.
- Die Webserver-Einstellungen für Aktivierung des Webservers und HTTP/HTTPS-Einstellungen werden übernommen. Zusätzlich haben Sie die Möglichkeit, Benutzer anzulegen und ihnen spezifische Rechte zuzuweisen (Bereich Webserver > Benutzerverwaltung der CPU-Parameter). Ohne Parmetrierung von Benutzern hat ein Webserver-Nutzer nur Zugriff auf die Standard-Webseiten. Eine S7-1200-CPU V4 unterstützt nicht mehr den Benutzer "admin" und dessen Passwort.
- Aus Kompatibilitätsgründen ändern sich die Schnittstellen verwendeter OBs durch den Gerätetausch nicht. Verwendete OBs können Sie also weiterverwenden. Falls Sie OB-Eigenschaften verwenden wollen, die im neuen Firmware-Stand der S7-1200-CPU hinzugekommen sind bzw. geändert wurden, müssen Sie den entsprechenden OB neu anlegen.

Kommunikation mit HMI-Geräten

Wenn Sie an eine S7-1200-CPU V4 ein HMI-Gerät anschließen, achten Sie auf die geeignete Version der Runtime-Software des HMI-Gerätes.

Gegebenenfalls müssen Sie die neueste HMI-Runtime-Version über die Engineering Software WinCC übertragen.

Sie müssen die HMI-Projektierung neu übersetzen und auf das HMI-Gerät laden, damit die CPU-HMI-Kommunikation im Betrieb funktioniert.

Siehe auch

Wissenswertes zu SIMATIC Memory Cards (Seite 1371)

1.4.2.2 Baugruppen adressieren

Adressierung von Baugruppen

Einführung

In der Geräteübersicht sehen Sie in den Spalten E-Adresse und A-Adresse die Adressen oder Adressbereiche der Baugruppen. Daneben gibt es noch weitere Adressen, die im Folgenden erläutert werden.

E/A-Adresse (Peripherieadresse)

E/A-Adressen (Ein/Ausgabeadressen) werden dazu benötigt, um im Anwenderprogramm Eingänge zu lesen bzw. Ausgänge zu setzen.

Ein- und Ausgangsadressen werden automatisch beim Stecken von Baugruppen auf den Baugruppenträger vergeben. Die Adresse des ersten Kanals bildet die Anfangsadresse einer Baugruppe. Die Adressen der übrigen Kanäle ergeben sich aus dieser Anfangsadresse. Das Adressende ergibt sich aus der baugruppenspezifischen Adresslänge.

Teilnehmeradresse (z. B. Ethernet-Adresse)

Teilnehmeradressen sind Adressen von programmierbaren Baugruppen (Industrial Ethernet-Adressen). Sie werden benötigt, um die verschiedenen Teilnehmer eines Subnetzes zu adressieren, z. B. um ein Anwenderprogramm in eine CPU zu laden.

HW-Kennung zur Identifikation von Baugruppen bzw. Submodulen

Zusätzlich zu den E-Adressen und A-Adressen wird automatisch eine HW-Kennung (HW-ID) vergeben, die zur Adressierung und Identifizierung der Baugruppe verwendet wird. Auch Submodule (Einheiten einer Baugruppe) z. B. ein integrierter Zähler erhalten eine solche HW-Kennung.

Die HW-Kennung besteht aus einer Ganzzahl und wird vom System bei Diagnosemeldungen mit ausgegeben, um die fehlerhafte Baugruppe bzw. das fehlerhafte Submodul zu lokalisieren.

Außerdem verwenden Sie die HW-Kennung für eine Reihe von Anweisungen, um die entsprechende Baugruppe zu adressieren.

Die HW-Kennung ist nicht änderbar.

Beispiel: Schneller Zähler der S7-1200 CPU identifizieren

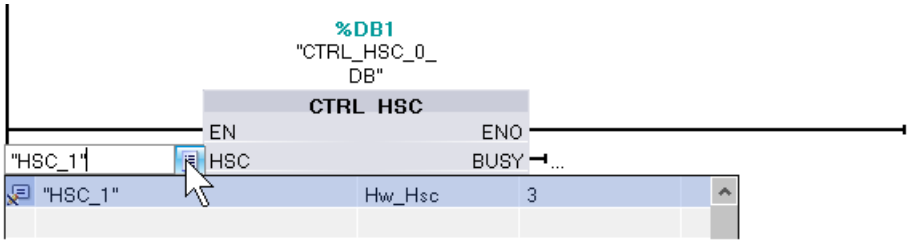

Die HW-Kennung wird automatisch beim Stecken von Komponenten in der Geräte- oder Netzsicht vergeben und bei den PLC-Variablen (Register Systemkonstanten) eingetragen. Ein Name für die HW-Kennung wird ebenfalls automatisch vergeben. Die Systemkonstanten der PLC-Variablen sind ebenfalls nicht änderbar.

Siehe auch

Ein- und Ausgangsadressen festlegen (Seite 1459)

Adressen einer Verwendungsstelle im Programm zuweisen (Seite 1460)

Einführung zum Laden einer Konfiguration (Seite 3190)

Inspektorfenster (Seite 24)

Ein- und Ausgangsadressen festlegen

Ein- und Ausgangsadressen werden automatisch voreingestellt. Sie können aber die Adresszuweisung nachträglich verändern.

Alle Adressen von Baugruppen liegen im Prozessabbild-Bereich. Das Prozessabbild wird automatisch zyklisch aktualisiert.

Voraussetzung

Sie befinden sich in der Gerätesicht.

Vorgehen

Um den voreingestellten Adressbereich zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Gerätesicht auf die Baugruppe, deren Anfangsadresse Sie einstellen wollen.
2. Navigieren Sie im Inspektorfenster unter "Eigenschaften" zu den "E/A-/Adressen".
3. Geben Sie unter "Anfangsadresse" die neue gewünschte Anfangsadresse ein.
4. Drücken Sie <Return> oder klicken auf ein beliebiges anderes Objekt um den geänderten Wert zu übernehmen.

Wenn Sie eine ungültige Adresse eingegeben haben, wird eine Meldung mit der nächsten freien Adresse eingeblendet.

Hinweis

Sie können die Adressen auch direkt in der Geräteübersicht ändern.

Siehe auch

Eigenschaften und Parameter bearbeiten (Seite 64)

Ein- und Ausgangsadressen in der Adressübersicht (Seite 66)

Adressen einer Verwendungsstelle im Programm zuweisen

Sie können Adressen von den E/A-Kanälen der Baugruppen direkt den Verwendungsstellen im Programm oder einer Variablentabelle zuordnen.

Voraussetzung

- Die Gerätesicht des Hardware- und Netzwerkeditors ist geöffnet.
- Die Zoomstufe in der Gerätesicht muss auf mindestens 200 % eingestellt sein, damit Sie die einzelnen E/A-Kanäle sehen können.
- Das Anweisungsfenster des Programmiereditors oder eine Variablentabelle ist geöffnet.

Vorgehen

Um E/A-Kanäle von Baugruppen den Verwendungsstellen im Programm oder einer Variablentabelle zuzuordnen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Gerätesicht zu der Baugruppe mit dem gewünschten E/A-Kanal.
2. Ziehen Sie mit gedrückter Maustaste die gewünschte E/A-Adresse auf die entsprechende Verwendungsstelle des Bausteins oder in die Variablentabelle.

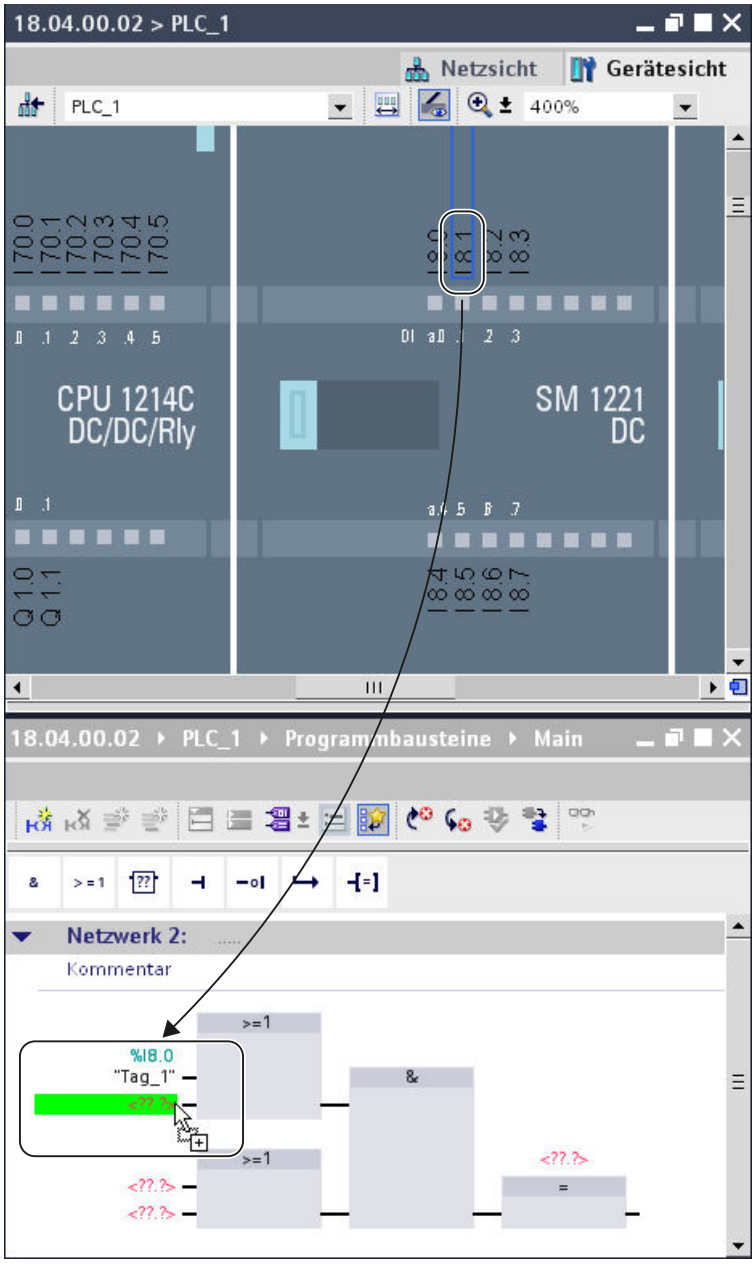

Die Adresse der Baugruppe wird der Verwendungsstelle im Programm zugewiesen, bzw. als Variable in die Variablentabelle eingetragen.

Hinweis

Die Variable für einen Ein- oder Ausgang eines Bausteins kann auch auf den Ein- oder Ausgang einer Baugruppe gezogen werden, um die Variable mit dem E/A-Kanal der Baugruppe zu verknüpfen.

1.4.2.3 Signalboard

Signalboard in CPU stecken

Einführung

Signalboards bieten Ihnen bei S7-1200 CPUs die Möglichkeit, die Anzahl der CPU-eigenen Ein- und Ausgänge zu erhöhen. Signalboards finden Sie wie alle anderen Hardware-Komponenten im Hardware-Katalog. Sie stecken Signalboards nicht wie andere Baugruppen in den Baugruppenträger, sondern direkt in einen CPU-eigenen Steckplatz.

Bei der Verwendung eines Signalboards ist folgendes zu beachten:

- Jede CPU kann nur ein Signalboard aufnehmen.
- Ein Signalboard kann nur gesteckt werden, wenn der Steckplatz der CPU frei ist.

Um ein Signalboard in eine CPU zu stecken, gibt es folgende Möglichkeiten:

- Doppelklick auf ein Signalboard im Hardware-Katalog bei freiem Steckplatz in der CPU
- Drag & Drop aus dem Hardware-Katalog auf einen freien Steckplatz in der CPU
- Kontextmenü eines Signalboards im Hardware-Katalog für "Kopieren" und "Einfügen"

Voraussetzung

- Der Hardware-Katalog ist geöffnet.
- Die S7-1200 CPU hat einen freien Steckplatz für das Signalboard.

Signalboard in CPU stecken

Um ein Signalboard in eine CPU zu stecken, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu dem gewünschten Signalboard.
2. Selektieren Sie das ausgewählte Signalboard.
3. Ziehen Sie das Signalboard mit Drag & Drop in den freien Steckplatz der CPU.

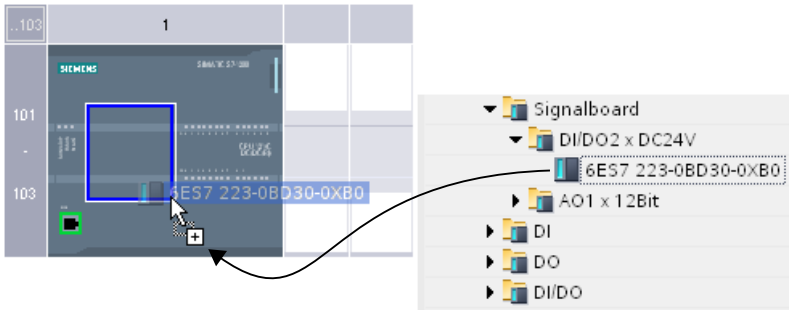

Sie haben nun das Signalboard in den Steckplatz der CPU gesteckt.

Wenn Sie sich in der Netzsicht befinden, können Sie auch ein Signalboard mit Drag & Drop auf ein Gerät ziehen. Wenn die CPU einen leeren Steckplatz für ein Signalboard hat, wird das Signalboard automatisch in diesen Steckplatz gesteckt.

1.4.2.4 Konfigurationen für Webserver

Wissenswertes zum Webserver

Einführung

Der Webserver gibt Ihnen die Möglichkeit, Ihre CPU über das Internet oder das firmeninterne Intranet zu beobachten. Auswertungen und Diagnose sind somit über große Entfernungen möglich.

Meldungen und Statusinformationen werden auf HTML-Seiten angezeigt.

Webbrowser

Für den Zugriff auf die HTML-Seiten der CPU benötigen Sie einen Webbrowser.

Die Information, mit welchen Webbrowsern und Webbrowser-Versionen der Webserver getestet wurde, erhalten Sie im entsprechenden Handbuch (http://support.automation.siemens.com/WW/view/de/36932465).

Informationen über den Webserver auslesen

Aus der CPU sind die im Folgenden aufgezählten Informationen auslesbar. Die Verfügbarkeit der jeweiligen Webseiten hängt von der CPU und ihrer Firmware-Version ab.

Ab Firmware-Version 4 sind die Webseiten in mehreren Sprachen verfügbar.

Seite/Information	Beschreibung
Einführung	Einstiegsseite für die Standard-Webseiten
Startseite mit allgemeinen CPU-Informationen	Die Startseite liefert eine Übersicht der generellen Informationen der CPU, den CPU-Namen, den CPU-Typ und Basisinformationen zum aktuellen Betriebszustand.
Diagnose	Anzeige detaillierter Information zur CPU einschließlich Seriennummer, Artikelnummer und Versionsnummern. Anzeige des Diagnosepuffer-Inhalts mit den jüngsten Einträgen zuerst.
Modulinformationen	Anzeige der Identifikiationsinformation, ob die zentral gesteckten Komponenten einer Station in Ordnung sind, ob z. B. Wartungsanforderungen vorliegen oder Komponenten nicht erreichbar sind. Ab FW-Version 4 ist ein Firmware-Update über diese Webseite möglich.
Kommunikation	Anzeige der Kommunikationsverbindungen bei offener Kommunikation (OUC), Anzeige der Ressourcen sowie Adressparameter.
Diagnosepuffer	Anzeige des Diagnosepuffer-Inhalts.
Variablen	Anzeige des Status von Operanden des Anwenderprogramms zum Beobachten und Ändern der Werte.
Dateibrowser (ab FW-Version 4)	Datenarchive im CSV-Format zum Übertragen auf die Festplatte des PGs. Die Datenarchive werden mit Data Log-Anweisungen im Anwenderprogramm angelegt und mit Daten befüllt. Ab FW-Version 4 haben Sie über diese Webseite Zugriff auf Dateien des internen Ladespeichers und des externen Ladespeichers (Memory Card), z. B. auf den Inhalt der Verzeichnisse "DataLogs" und "Recipes".
Anmeldung	Anmeldung als Benutzer oder Abmeldung.
Anwenderseiten (falls anwenderdefinierte Webseiten projektiert und geladen wurden)	Die Anwenderwebseiten liefern eine Liste der Webseiten mit kundenspezifischen Webapplikationen.

Web-Zugriff auf die CPU über PG/PC

Um auf den Webserver zuzugreifen, gehen Sie folgendermaßen vor:

1. Verbinden Sie den Client (PG, PC) über die PROFINET-Schnittstelle mit der CPU.
2. Öffnen Sie den Webbrowser.
   Tragen Sie im Feld "Adresse" des Webbrowsers die IP-Adresse der CPU ein in der Form http://ww.xx.yy.zz (beispielhafte Eingabe: http://192.168.3.141).
   Die Startseite der CPU wird geöffnet. Von der Startseite aus können Sie zu den weiteren Informationen navigieren.

Standard-Webseiten

Voraussetzungen für den Web-Zugriff

Im Folgenden werden die Voraussetzungen für den Zugriff auf Standard-CPU-Webseiten erläutert sowie die Auswirkungen fehlender bzw. vorhandener Projektierungsinformation dargestellt.

Voraussetzung

Der Webserver muss gestartet sein.

Der Webserver startet erst dann, wenn Sie in den Eigenschaften der CPU, Abschnitt "Webserver" den Webserver aktiviert haben.

Beachten Sie Folgendes:

Die Webseiten werden standardmäßig über eine unsichere Verbindung übertragen und sind nicht vor Angriffen Dritter sicher. Wenn Sie die Webseiten verschlüsselt an den Browser übertragen wollen, dann verwenden Sie die URL https://, gefolgt von der IP-Adresse der CPU.

Anmelden

Um lesend auf die Standard-Webseiten zu zugreifen, ist kein Anmelden notwendig. Zum Ausführen bestimmter Aktionen wie das Ändern des Betriebszustandes der CPU oder für Schreibzugriffe muss ein Benutzer als angemeldet sein.

S7-1200-CPUs bis FW-Version V3:

Für die oben genannten Aktionen müssen Sie als Benutzer "admin" angemeldet sein. Die Eingabefelder zum Anmelden sind in der linken oberen Ecke auf jeder Standard-Webseite zu finden.

Wenn Sie sich als Benutzer "admin" anmelden, müssen Sie Benutzername und Passwort dort eingeben.

Name: admin.

Passwort: konfiguriertes CPU-Passwort (für passwortgeschützte CPU).

S7-1200-CPUs ab FW-Version 4:

Sie können die Namen von Benutzern und die Passwörter frei wählen (CPU-Parameter "Webserver", Bereich "Benutzerverwaltung").

Benutzern weisen Sie Rechte zu, z. B. das Recht, Diagnose abzufragen oder die Firmware zu aktualisieren.

Javascript und Cookies

Die Standard-Webseiten nutzen Javascript und Cookies. Sie sollten beides in Ihrem Webbrowser freigegeben haben.

Wenn Javascript nicht freigegeben ist, bestehen z. B. folgende Einschränkungen:

- Daten von Standard-Webseiten werden nicht automatisch aktualisiert.
- Sie können sich nicht als Benutzer anmelden.
- Felder können nicht sortiert werden (Modulinformation)

Wenn Cookies nicht freigegeben sind, können Sie sich nicht anmelden.

Siehe auch

Zugriff für HTTPS (Seite 1467)

Einstellungen für den Betrieb

Einstellungen für den Betrieb

Um den Webserver einer S7-1200-CPU nutzen zu können, müssen Sie in der Netzsicht bzw. Gerätesicht die CPU markieren und folgende Einstellungen im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" vor:

- Webserver aktivieren
- Zugriffe auf die CPU einschränken auf das Übertragungsprotokoll HTTPS (verschlüsselte Übertragung)
  Ein Zugriff über Port 80 wird dann gesperrt. Kommunikation ist nur über Port 443 möglich.

- Automatische Aktualisierung der Webseiten aktivieren
  Das Aktualisierungsintervall ist voreingestellt und nicht änderbar. Webseiten mit veränderlichen Inhalten (z. B. Statusinformationen oder Diagnoseinformationen) aktualisiert die CPU in regelmäßigen Abständen.
- Benutzer anlegen und verwalten
  Den Benutzern stehen ausschließlich die Optionen zur Verfügung, die den Zugriffsrechten fest zugeordnet sind.
  Je nach verwendeter CPU und Firmware können Sie unterschiedliche Benutzerrechte vergeben. Rechte, die Ihre CPU nicht unterstützt, können nicht aktiviert werden. In der Benutzerliste ist voreingestellt ein Benutzer mit Namen "Jeder" mit minimalen Zugriffsrechten, die Sie aber erweitern können. Wer ohne Passworteingabe den Webserver nutzt, hat die Zugriffrechte des Benutzers "Jeder".
  Sie haben die Möglichkeit, weitere Benutzer mit unterschiedlichen Zugriffsrechten zu parametrieren. Diese Benutzer müssen sich mit dem parametrierten Benutzernamen und Passwort anmelden.

  ⚠ WARNUNG

  Unautorisierter Zugriff auf die CPU über den Webserver

  Unautorisierter Zugriff auf die CPU oder das Ändern von PLC-Variablen auf ungültige Werte kann zu Unterbrechungen des von der CPU gesteuerten Prozesses führen und Tod, schweren körperlichen Schaden bzw. Sachschaden verursachen!

  Weil die Aktivierung des Webservers autorisierten Personen erlaubt, z. B. Betriebszustände zu ändern, auf CPU-Daten schreibend zuzugreifen oder die Firmware zu aktualisierten, empfehlen wir, folgende Sicherheitsmaßnahmen zu berücksichtigen:

  - Schränken Sie wenn möglich Zugriffe auf das HTTPS-Protokoll ein.
  - Legen Sie Benutzer an mit sicheren Passwörtern. Ein sicheres Passwort wird z. B. nur für eine einzige Anwendung verwendet, ist mehr als 8 Zeichen lang und besteht aus Groß- und Kleinbuchstaben sowie Sonderzeichen und Ziffern (?!+%$1234...). Des Weiteren sollten keine gängigen Zeichenfolgen der Computertastatur oder Wörter aus dem Wörterbuch verwendet werden.
    Ändern Sie das Passwort in regelmäßigen Abständen.
  - Erweitern Sie nicht die Rechte des Benutzers "Jeder".
  - Prüfen Sie PLC-Variablen im Anwenderprogramm und schränken den Wertebereich auf zulässige Bereiche ein, da Nutzer über den Webserver ungültige Werte setzen können.

Zugriff für HTTPS

Zugriff über HTTPS

HTTPS dient zur Verschlüsselung und Authentifizierung der Kommunikation zwischen Browser und Webserver.

Um die Datenübertragung zwischen Browser und CPU über das HTTPS-Protokoll durchzuführen, geben Sie die URL in der Form https://ww.xx.yy.zz in die Adresszeile Ihres Webbrowsers ein, wobei ww.xx.yy.zz für die IP-Adresse der CPU steht.

Für einen fehlerfreien HTTPS-Zugriff auf die CPU benötigen Sie ein gültiges und installiertes Zertifikat.

Ist kein Zertifikat installiert, wird eine Warnung angezeigt, mit der Empfehlung, die Seite nicht zu benutzen. Um die Seite sehen zu können, müssen Sie explizit eine "Ausnahme hinzufügen".

Ein gültiges Zertifikat (Certification Authority) "SIMATIC CONTROLLER" erhalten Sie als Download von der Webseite "Intro" unter "Download certificate". Wie Sie ein Zertifikat installieren, finden Sie in der Hilfe Ihres jeweiligen Webbrowsers.

Auf Daten des CPU-Speichers zugreifen

Um auf Daten, die sich im internen oder externen Ladespeicher der CPU befinden, können Sie über eine Standard-Webseite zugreifen.

- Bei S7-1200-CPUs bis einschließlich FW-Version 3 nutzen Sie die Webseite "Data Logs". Von dieser Webseite übertragen Sie Data Logs von der CPU auf ein Laufwerk Ihres PCs.
- Bei S7-1200-CPUs ab FW-Version 4 nutzen Sie die Webseite "File Browser". Von dieser Webseite übertragen Sie Daten z. B. aus den Ordnern "Data Logs" oder "Recipes" auf ein Laufwerk Ihres PCs.

Abhängig vom Dateityp und den Zugriffsrechten, die Sie für Web-Server-Benutzer parametriert haben, können die Dateien heruntergeladen, gelöscht, umbenannt oder hochgeladen werden. Die Verzeichnisse selbst können nur erstellt, gelöscht oder umbenannt werden.

Beispiel: Data Logs

Zum Öffnen eines Data Logs klicken Sie auf den Link des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

Besonderheit: Data Logs sind im U.S. amerikanischen CSV-Format gespeichert. Sie können die Datei nur mit der U.S.-Version von Microsoft Excel direkt öffnen. Mit anderen Länderversionen von Microsoft Excel müssen Sie die Datei importieren und dabei im Import-Assistenten als Trennzeichen "Komma" wählen.

Herunterladen eines Data Logs

Zum Herunterladen eines Data Logs klicken Sie auf das Download-Symbol des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

Herunterladen und Leeren bzw. Löschen eines Data Logs

Für eine CPU mit der FW-Version bis V3.0:

Zum Herunterladen und Löschen der aktuellen Einträge des Data Logs müssen Sie angemeldet sein. Klicken Sie dazu auf das Symbol "Herunterladen und Löschen" des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

Für eine CPU ab der FW-Version V4.0:

Um das Data Log zurückzusetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die CSV-Datei, z. B. mit Excel.
2. Löschen Sie die Zeilen zwischen Überschriftenzeile und der Zeile mit dem Eintrag "//END", falls diese Zeile existiert.
3. Speichern Sie die Datei auf einem Laufwerk Ihres PCs.
4. Löschen Sie auf der Webseite "File Browser" das Data Log (d. h. die CSV-Datei) und laden die vorbereitete CSV-Datei über die Schaltfläche "Upload file" der Webseite "File Browser" in die CPU.

Weitere Informationen finden Sie im Systemhandbuch zur S7-1500-CPU.

Anwenderseiten erstellen und laden

Wissenswertes zu Anwenderseiten

Konzept

Das Konzept der Anwenderseiten bzw. Anwenderdefinierten Webseiten ermöglicht Ihnen, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen. Der Webserver der CPU stellt diese Funktion zur Verfügung.

Hinsichtlich der Gestaltung und Funktionalität der anwenderdefinierten Webseiten sind Sie nicht auf spezielle Tools angewiesen. Sie können die Seiten mit CSS im Layout anpassen, mit Javascript dynamischen Inhalt zur Verfügung stellen oder ein beliebiges Framework zur Produktion der Webseiten verwenden. Die Gesamtheit der Dateien, die der Webserver verarbeitet, wird auch "Webapplikation" genannt.

Webapplikation und Anwenderprogramm

Mit Hilfe von speziellem HTML-Code in anwenderdefinierten Webseiten (AWP-Kommandos) können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an das Anwenderprogramm der CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen.

Mit Scriptanweisungen (z. B. Javascript) können Sie Ihre Webseiten optimieren, z. B Inhalte dynamisch ändern oder Benutzereingaben validieren.

Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung WWW (SFC 99) im Anwenderprogramm aufrufen.

- Wenn keine Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, z. B. wenn eine Webseite nur statische Informationen anbietet, ist nur eine Initialisierung im Anwenderprogramm notwendig.
- Wenn ein einfacher Datenaustausch zwischen PLC-Variablen und Variablen der Webapplikation notwendig ist, z. B. um den Inhalt von PLC-Variablen anzuzeigen oder in eine PLC-Variable einen Wert zu schreiben, müssen Sie die Syntax zum Lesen und Schreiben von Variablen zu berücksichtigen. Im Anwenderprogramm ist in diesem Fall ebenfalls nur eine Initialisierung z. B. im Anlauf-OB erforderlich.
- Wenn eine weitergehende Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, müssen Sie neben der Synchronisierung zwischen Webserver und Anwenderprogramm auch Status- und Steuerinformation aus dem Web-Control-DB hantieren. Das ist z. B. der Fall bei Benutzereingaben, die über den Webbrowser an den Webserver übergeben werden, um von der CPU ausgewertet zu werden. Im Unterschied zum einfachen Datenaustausch beeinflusst das Anwenderprogramm unmittelbar den Zeitpunkt, zu dem die angeforderte Webseite an den Webbrowser zurückgegeben wird. In diesem Fall ist es notwendig, dass Sie sich mit dem Konzept der manuellen Fragmente und den Strukturen des Web-Control-DBs auseinandersetzen.

Einbindung der HTML-Quellen über das TIA Portal

Die Parameter für eine Einbindung der HTML-Quellen im TIA Portal finden Sie in den Eigenschaften der jeweiligen CPU mit Webserver:

Parameter	Bedeutung
HTML-Verzeichnis	Verzeichnis der HTML-Quellen für die Webapplikation. Sie können den Pfad direkt eingeben oder über die Schaltfläche neben dem Eingabefeld zu dem Verzeichnis navigieren. Es ist vorteilhaft, wenn Sie Sie Ihre Webseiten mit relativen Pfaden einbinden. Dazu müssen Sie das Verzeichnis, in dem Sie die HTML-Quellen für die Webapplikation speichern, im STEP 7-Projektverzeichnis eingefügen, z. B. in ein neues Verzeichnis "Webpages". Als HTML-Verzeichnis geben Sie dann den relativen Pfadnamen an; im Beispiel ".\Webpages". Vorteile der Verwendung relativer Pfade: • Die anwenderdefinierten Webseiten werden gemeinsam mit dem STEP 7-Projekt archiviert. • Die Webseiten werden über den Menübefehl "Projekt > Speichern unter..." mit in den neuen Pfad kopiert. • Beim Kopieren des Projekts über den Windows Explorer in einen anderen Pfad entfällt die Anpassung des HTML-Verzeichnis-Pfades zur Generierung neuer Web-DBs. • Mehrere Steuerungen in einem Projekt, die die gleichen HTML-Seiten verwenden, haben auch die gleiche HTML-Verzeichnis-Pfadangabe.
Start-HTML-Seite	Pfad zur Start-HTML-Seite. Sie können den Pfad direkt eingeben oder über die Schaltfläche neben dem Eingabefeld zu der HTML-Seite zu navigieren. Die Start-HTML-Seite ist die HTML-Seite, die beim Start der Webapplikation aufgerufen werden soll.
Applikationsname	Optionaler Name für die Applikation. Dieser Name dient zur weiteren Unterteilung bzw. Gruppierung der Webseiten. Wenn ein Applikationsname vorhanden ist, wird die URL in folgendem Format angezeigt: "http://a.b.c.d/awp/<Applikationsname>/<Seitenname>.html"

Initialisierung

Anwenderdefinierte Webseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren, um die Webapplikation in die CPU laden zu können. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Datenbausteine, die codierte Webseiten enthalten, werden "Fragment-DBs" genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin anwenderdefinierte Webseiten codiert sind. Durch die Anweisung "WWW" (SFC 99) z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die anwenderdefinierten Webseiten über einen Webbrowser erreichbar.

Synchronisierung

Wenn das Anwenderprogramm mit den anwenderdefinierten Webseiten Daten austauschen bzw. interagieren soll, muss die Anweisung WWW (SFC 99) im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragendenden Webbrowser zusammenstellen und zurücksenden

In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer angeforderten Webseite dem Webserver mitgeteilt werden.

Überblick Vorgehensweise

Grundlegendes

In diesem Kapitel ist die grundsätzliche Vorgehensweise schrittweise erläutert, um anwenderdefinierte Webseiten zu erstellen, zu laden und in der Betriebsphase zu nutzen.

Die folgende Grafik stellt den Ablauf zur Erstellung und Anzeige von anwenderdefinierten Webseiten vereinfacht dar:

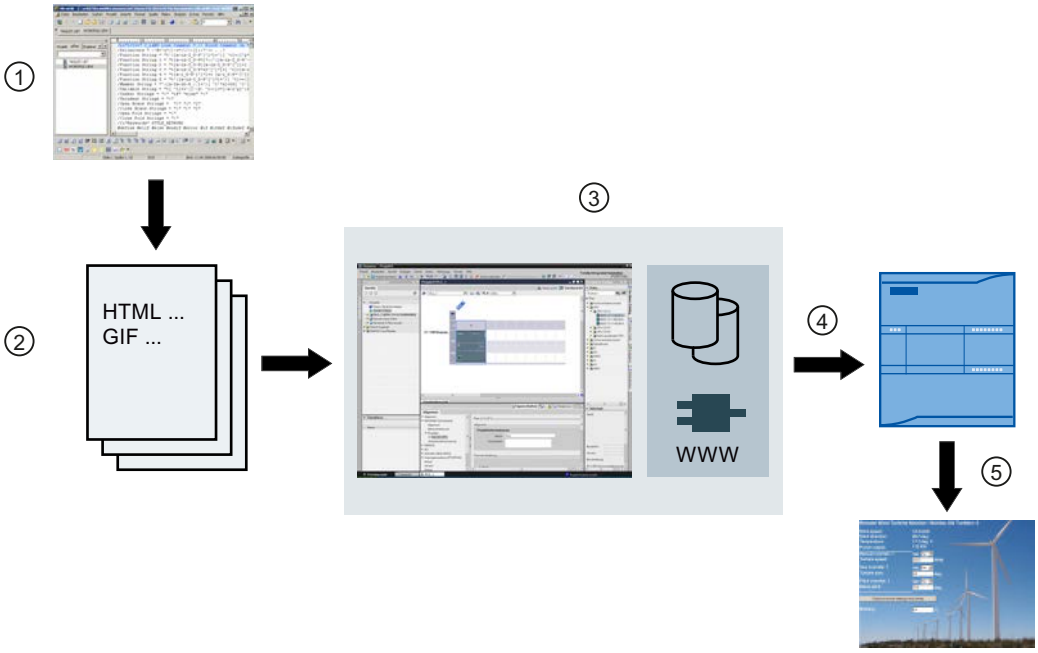

① Programmieren einer Webapplikation (evtl. unter Zuhilfenahme von geeigneten Tools, gegebenefalls mit enthaltenen AWP-Kommandos für dynamische Seiten).

② Webapplikation besteht aus einzelnen Quelldateien, z. B. *.html, *.gif, *.js, ...

③ Mit STEP 7:

- Generieren der Datenbausteine (Web-Control-DB und Fragment-DBs) aus Quelldateien. Die DBs enthalten Metainformation und die komplette Webapplikation inklusive der Bilder und den dynamischen und statischen Teilen der Webapplikation. Die DBs werden unter "Systembausteine" in der Projektnavigation abgelegt.
- Die Anweisung "WWW" im Anwenderprogramm aufrufen. Die Anweisung initialisiert den Webserver der CPU für eine Webapplikation.
- Bei Bedarf Interaktion zwischen Webserver und Anwenderprogramm ausprogrammieren

④ Laden der Bausteine in die CPU.

⑤ Webseite im Browser aufrufen. Der Aufruf der Webseiten der CPU erfolgt über die Eingabe der IP-Adresse der CPU.

Weitere Informationen

Weitere Informationen und Beispiele zum Webserver der S7-1200 finden Sie im Internet (http://support.automation.siemens.com/WW/view/de/36932465).

Webseiten erstellen

Zur Erstellung der anwenderdefinierten Webseiten bieten sich Webdesign-Tools diverser Anbieter an. Grundsätzlich sollen die Webseiten nach den Konventionen des W3C (World Wide Web Consortium) programmiert und gestaltet werden. Im Webserver der CPU findet keine Überprüfung nach W3C Kriterien statt.

Regeln

- Das Tool muss in der Lage sein, den HTML-Code direkt zu editieren, damit Sie AWP-Kommandos in der HTML-Seite einfügen können.
  In der CPU werden lediglich die AWP-Kommandos geparst und z. B. durch Werte aus dem Anwenderprogramm/Prozessabbild der CPU ersetzt.
- Dateien, die AWP-Kommandos enthalten, müssen UTF-8 codiert sein. In den Metadaten der HTML-Seite setzen Sie daher das Attribut charset auf UTF-8 und speichern die Datei UTF-8-codiert.
- Dateien, die AWP-Kommandos enthalten, dürfen folgende Sequenz nicht enthalten:]]
- Dateien, die AWP-Kommandos enthalten, dürfen außerhalb von "Variable-lesen-Bereichen" (:=<Variablenname>:) folgende Sequenz nicht enthalten: :=
  Tipp: Ersetzen Sie das erste Zeichen einer verbotenen Sequenz durch seine Zeichencodierung; für den Doppelpunkt z. B. :.

Ein kleines Beispiel für eine anwenderdefinierte Webseite soll den grundsätzlichen Aufbau zeigen.

Voraussetzung

- Die CPU muss über einen Webserver verfügen und der Webserver der CPU muss aktiviert sein
- Um als Nutzer von anwenderdefinierten Webseiten schreibend auf PLC-Variablen zugreifen zu können, müssen Sie als "admin" angemeldet sein.
- Für das Beispiel unten müssen für PLC-Variablen, die auf der Webseite angezeigt werden sollen, PLC-Variablen definiert sein. Für die erste verwendete PLC-Variable "Tank_below_max" ist das hier dargestellt.

		Name	Datentyp	Adresse
1		Tank_below_max	Bool	%I0.0

Anwenderdefinierte Webseiten erstellen

Der folgende Code für eine Beispielwebseite liest Werte aus dem Prozessabbild und stellt diese in einer Tabelle dar.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/
TR/html4/strict.dtd">
<html>
  <head>
   <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
   <title>Mix</title>
  </head>
   <body>
    <h1>Mix</h1>
    <h2> Actual State </h2>
    <table border="1">
     <tr>
      <th>Variable</th>
      <th>State</th>
```

```
      </tr>
      <tr>
       <td>Tank below max</td>
       <td>:="Tank_below_max":</td>
      </tr>
      <tr>
       <td>Tank above min</td>
       <td>:="Tank_above_min":</td>
      </tr>
     </table>
    </body>
</html>
```

AWP-Kommandos für die Steuerung von Webseiten

AWP-Kommandos in Webseiten verwenden

Mit AWP-Kommandos (Automation Web Programming) deklarieren Sie die Schnittstelle zwischen Ihren Anwenderseiten (Webapplikation, z. B. eine einfache HTML-Seite) und den CPU-Daten.

Für die Entwicklung von Anwenderseiten bzw. Webapplikationen sind Sie nur den Beschränkungen unterworfen, die Webbrowser vorgeben. Mit dem Anwenderprogramm in der CPU kontrollieren Sie in einer der Programmiersprachen von STEP 7, welche CPU-Daten zu welcher Zeit im Webbrowser des Betrachters angezeigt werden.

Mit Hilfe von AWP-Kommandos, die Sie innerhalb der HTML-Seite notieren, werden Daten für die beabsichtigte Interaktion zwischen Webapplikation und Anwenderprogramm vereinbart.

AWP-Kommandos fügen Sie als HTML-Kommentare mit einer speziellen Syntax in HTML-Seiten ein. Mithilfe von AWP-Kommandos lassen sich folgende Features umsetzen:

- PLC-Variablen lesen
- PLC-Variablen schreiben
- Spezialvariablen lesen
- Spezialvariablen schreiben
- Enum-Typen definieren
- Variablen Enum-Typen zuweisen
- Fragmente definieren
- Fragmente importieren

Syntax von AWP-Kommandos

Ein AWP-Kommando beginnt mit "`<!-- AWP_`" und endet mit "` -->`". In Java-Script-Dateien sollten die Kommandos zusätzlich von Javascript-Kommentaren eingeschlossen werden ("`/*...*/`").

Notationsregeln für PLC-Variablennamen innerhalb eines AWP-Kommandos

Die AWP-Kommandos "AWP_In_Variable" und "AWP_Out_Variable" enthalten ein Name-Attribut und optional ein Use-Attribut. Diesen Attributen ist ein PLC-Variablenname zugeordnet, über den PLC-Variablen im Browser geschrieben bzw. gelesen werden. Für den Umgang mit PLC-Variablennamen im HTML-Code gibt es folgende Regeln:

- PLC-Variablen müssen in Anführungszeichen (" ... ") eingeschlossen werden.
- In AWP-Kommandos verwendete PLC-Variablen müssen zusätzlich von einfachen Anführungszeichen ('" ... "') oder mit umgekehrtem Schrägstrich (Backslash) maskierten Anführungszeichen umschlossen werden ("\" ... \"").
- Wenn der PLC-Variablenname das Zeichen \ (umgekehrter Schrägstrich) oder * (Sternchen) enthält, dann muss dieses Zeichen mit der Escape-Sequenz \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden. Beispiele siehe unten.
- Wenn der PLC-Variablenname im AWP-Kommando zusätzlich mit einfachen Anführungszeichen eingeschlossen ist und innerhalb des Namens kommt das einfache Anführungszeichen (') vor, muss es auch durch die Escape-Sequenz \' als normales Zeichen gekennzeichnet werden.
- Wenn im AWP-Kommando eine absolute Adresse (Eingang, Ausgang, Merker) verwendet wird, dann wird sie von einfachen Anführungszeichen umschlossen.

PLC-Variable	PLC-Variable im HTML-Code
"Velocity"	<!-- AWP_In_Variable Name='"Velocity"' -->
	<!-- AWP_In_Variable Name="\"Velocity\"" -->
"abc\de"	<!-- AWP_In_Variable Name='"abc\\de"' -->
"abc'de"	<!-- AWP_In_Variable Name='"abc\'de"' -->
"abc'de"	<!-- AWP_In_Variable Name="abcde" Use'"abc\'de"' -->
"DB name".tag	<!-- AWP_In_Variable Name='"DB name".tag' -->
"Plc1".Data[1].typeDataSt-ruct.value	<!-- AWP_In_Variable Name='"Plc1".Data[1].typeDataStruct.value' -->
-	<!-- AWP_Out_Variable Name=’flag1’ Use='M0.0' -->

Siehe auch

Variablen lesen (Seite 1475)

Variablen schreiben (Seite 1477)

Spezialvariablen (Seite 1478)

Variablen lesen

Anwenderdefinierte Webseiten können PLC-Variablen lesen.

Die PLC-Variable muss durch einen PLC-Variablennamen spezifiziert sein.

Diese Out-Variablen (Ausgaberichtung von der Steuerung aus betrachtet) werden mit der im folgenden beschriebenen Syntax an beliebiger Stelle im HTML-Text eingefügt.

Syntax

```
:=<varname>:
```

Diese Referenzen werden im Betrieb vom Webserver durch die jeweils aktuellen Werte der PLC-Variablen ersetzt.

`<varname>` kann eine einfache globale PLC-Variable sein, aber auch ein kompletter Variablenpfad zu einem Strukturelement.

Notationsregeln für PLC-Variablennamen

- PLC-Variablen im HTML-Code werden mit Anführungszeichen (") umschlossen, wenn sie in der Variablentabelle definiert sind. Bei Datenbaustein-Variablen wird der Name des Datenbausteins mit Anführungszeichen umschlossen. Wenn spezielle Zeichen in den Strukturelementen des Datenbausteins verwendet werden wie z. B. der Punkt (.) oder Leerzeichen, muss auch dieser Teil in Anführungszeichen eingeschlossen sein.
- Anführungszeichen werden nicht verwendet bei absoluten Adressen von Eingängen, Ausgängen oder Merkern.

PLC Variable	PLC-Variable im HTML-Code
"DB_name".var_name	:="DB_name".var_name:
"DB_name"."var.name"	:="DB_name"."var.name":
"memory"	:="memory":
-	:=E0.0:
	:=A0.0:
	:=MW100:
	:=%MW100:
"My_Data_Block".flag1	<!-- AWP_Out_Variable Name='flag1' Use='"My_Data_Block".flag1' -->
	...
	:=flag1:

- Wenn der PLC-Variablenname die Zeichen : (Doppelpunkt) oder \ (umgekehrter Schrägstrich) enthält, dann müssen diese Zeichen mit der Escape-Sequenz \: bzw. \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden.

PLC Variable	PLC-Variable im HTML-Code
"abc:de"	:="abc\:de":
"abc\de"	:="abc\\de":

- Spezielle Zeichen "<, &, >"
  Falls diese Zeichen im Variablennamen vorkommen (z. B. "a<b"), kann es zu Darstellungsproblemen kommen.
  Vermeiden Sie Ausdrücke wie z. B. :="a<b": in der HTML-Seite.
  Um Darstellungsproblemen vorzubeugen, verwenden Sie z. B. ein AWP-Kommando mit einem Use-Ausdruck nach dem unten stehenden Muster. Das Use-Attribut definiert die PLC-Variable mit den problematischen Zeichen, das Name-Attribut definiert den Namen ohne problematische Zeichen, wie er in der HTML-Seite verwendet wird.

PLC Variable	PLC-Variable im HTML-Code
"a<b"	<!-- AWP_Out_Variable Name='simplename' Use='"a<b"' --> ... :=simplename:

Siehe auch

AWP-Kommandos in Webseiten verwenden (Seite 1474)

Variablen schreiben

Anwenderdefinierte Webseiten können Daten in die CPU schreiben.

Voraussetzung ist ein AWP-Kommando, das die zu schreibende PLC-Variable identifiziert.

Die PLC-Variable muss ferner durch einen PLC-Variablennamen spezifiziert sein.

Die In-Variablen (Eingaberichtung von der Steuerung aus betrachtet) werden auf der Browserseite gesetzt. Dies kann beispielsweise in einem Formular erfolgen.

Die Variablen werden entweder im HTTP-Header (per Cookie oder POST-Methode) oder in der URL (GET-Methode) vom Browser im HTTP-Request gesetzt und dann vom Webserver in die jeweilige PLC-Variable geschrieben.

Syntax

Damit die In-Variablen in die CPU geschrieben werden können, ist es notwendig, die Variablen durch eine explizite AWP-Anweisung zu definieren:

```
<!-- AWP_In_Variable Name='<PLC_Varname1>' Name='<PLC_Varname2>'
Name='<PLC_Varname3>' -->
```

In einer Anweisung können - wie oben - mehrere Variablen definiert werden.

Der konkrete PLC-Variablenname wird dabei in doppelten Anführungszeichen geschrieben; z. B. <PLC_Varname1> = "myVar".

Für den Fall, dass der Name der Variable, die Sie für die Webapplikation verwenden, nicht mit dem Namen der PLC-Variable identisch ist, kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen:

```
<!-- AWP_In_Variable Name='<Webapp_Varname>' Use='<PLC_Varname>'
```

Beispiel

Das AWP-Kommando "AWP_In_Variable" ist besonders für die Behandlung von Formularen unverzichtbar.

```
<form method='post' action='/awp/appl/x.html'>
 <p>
  <input name='"var1"' type='text'>
  <input value='set' name='Button1' type='submit'>
 </p>
</form>
```

Im oben definierten Formular wird mit der HTTP-Request-Methode "post" die Variable "var1" zum Webserver übertragen. Die Variable "var1" setzt der Anwender im Formularfeld. Die Variable 'Button1' hat den Wert 'set', wird aber für die CPU nicht benötigt. Damit die Variable "var1" in der CPU geschrieben werden kann, muss im gleichen Fragment folgende Anweisung aufgenommen werden:

```
<!-- AWP_In_Variable Name='"var1"' -->
```

Da PLC-Variablen mit doppelten Anführungszeichen (") umschlossen werden, muss der Name im AWP-Kommando mit einfachen Anführungszeichen (') oder mit maskierten doppelten Anführungszeichen (\") eingeschlossen sein. Um die zahlreichen Escape-Sequenzen zu vermeiden, empfehlen wir die Verwendung von einfachen Anführungszeichen.

```
<!-- AWP_In_Variable Name=’"Info".par1’ -->
<!-- AWP_In_Variable Name="\"Info".par1\"" -->
```

Bedingungen für den Schreibzugriff im Betrieb

Damit ein Anwender von einer anwenderdefinierten Webseite aus auf PLC-Variablen schreiben kann, muss folgende Voraussetzung erfüllt sein:

Der Anwender muss mit Rechte zum Ändern von Variablen haben. Der Webserver ignoriert die Befehle, wenn der Benutzer keine Änderungsrechte hat.

Diese Regel gilt prinzipiell für alle schreibenden Zugriffe von Webseiten auf eine CPU.

Siehe auch

Voraussetzungen für den Web-Zugriff (Seite 1465)

AWP-Kommandos in Webseiten verwenden (Seite 1474)

Spezialvariablen

Bei den Spezialvariablen handelt es sich hauptsächlich um die so genannten HTTP-Variablen die in den Definitionen des World Wide Web Consortium (W3C) festgelegt sind. Auch für Cookies und Server-Variablen werden Spezialvariablen verwendet.

Die AWP-Kommandos zum Lesen und Schreiben von Spezialvariablen unterscheiden sich nur durch zusätzliche Parameter von den AWP-Kommandos zum Lesen und Schreiben normaler Variablen.

Spezialvariable lesen

Der Webserver kann PLC-Variablen lesen und sie an Spezialvariablen im HTTP-Response-Header übergeben. Sie können z. B. eine URL für eine Umleitung zu einer anderen Webseite lesen und an die Spezialvariable HEADER:Location übergeben mit Hilfe der Spezialvariable HEADER:Location.

Folgende Spezialvariablen können gelesen werden:

Name	Beschreibung
COOKIE_VALUE:name	Wert des Cookies mit dem Namen: "name"
COOKIE_EXPIRES:name	Ablaufzeit des Cookies mit dem Namen: "name" in Sekunden (muss zuvor gesetzt worden sein).

Name	Beschreibung
HEADER:Status	HTTP-Statuscode (wenn kein anderer Wert gesetzt wurde wird Statuscode 302 zurückgegeben).
HEADER:Location	Pfad für die Weiterleitung auf eine andere Seite. Statuscode 302 muss gesetzt sein.
HEADER:Retry-After	Zeit in der der Service voraussichtlich nicht verfügbar ist. Statuscode 503 muss gesetzt sein.
HEADER: …	Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden.

Mit dem AWP-Kommando "AWP_Out_Variable" wird festgelegt, welche PLC-Variablen im HTTP-Header an den Webbrowser übertragen werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_Out_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Parameterbeschreibung

- Name: Typ und Name der Spezialvariablen
- Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen.

Beispiel:

```
<!-- AWP_Out_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Spezialvariable schreiben

Prinzipiell können alle vom Webbrowser in den HTTP-Header geschriebenen HTTP-Variablen vom Anwenderprogramm der CPU ausgewertet werden. Beispiele für Variablentypen:

Name	Beschreibung
HEADER:Accept-Language	Akzeptierte bzw. bevorzugte Sprache
HEADER:Authorization	Berechtigungsnachweis für eine angeforderte Ressource
HEADER:Host	Host und Port der angeforderten Ressource
HEADER:User-Agent	Informationen zum Browser
HEADER: …	Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden
SERVER:current_user_id	Zeigt an, ob ein Benutzer angemeldet ist (current_user_id=0: kein Benutzer angemeldet)
SERVER:current_user_name	Benutzername des angemeldeten Benutzers
SERVER:GET	Request-Methode ist GET
SERVER:POST	Request-Methode ist POST

Name	Beschreibung
COOKIE_VALUE:name	Wert des Cookies mit dem Namen: "name"

Mit dem AWP-Kommando "AWP_In_Variable" wird festgelegt, welche Spezialvariablen im Anwenderprogramm der CPU ausgewertet werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_In_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Parameterbeschreibung:

Name: Typ und Name der Spezialvariablen

Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter Use die Zuordnung zu einer PLC-Variablen erfolgen.

Beispiele:

```
<!-- AWP_In_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablennamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='COOKIE_VALUE:siemens' Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablennamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='"COOKIE_VALUE:siemens"' -->
```

Die HTTP-Header-Variable wird in die gleichnamige PLC-Variable geschrieben.

Siehe auch

AWP-Kommandos in Webseiten verwenden (Seite 1474)

Enumerationstypen

Enumerationstypen (Enums)

Mit Hilfe von Enums können numerische Werte aus dem PLC-Programm in Texte umgewandelt werden und umgekehrt. Die Zuordnung der numerischen Werte kann auch für mehrere Sprachen erfolgen.

Enums anlegen

Geben Sie ein AWP-Kommando mithilfe folgender Syntax am Anfang der HTML-Datei ein:

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Für z. B. deutsche Werte abzulegen als HTML-Datei im Ordner "de" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"an", 1:"aus", 2:"Störung"' -->
```

Für z. B. englische Werte abzulegen als HTML-Datei im Ordner "en" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -->
```

Enums zuordnen

Die Zuordnung der Variablen aus dem Anwenderprogramm zu den einzelnen Enum-Texten erfolgt über ein eigenes AWP-Kommando:

```
<!-- AWP_Enum_Ref Name="<VarName>" Enum="<EnumTypeName>" -->
```

Dabei ist <VarName> der symbolische Name aus dem Anwenderprogramm und <EnumTypeName> der zuvor festgelegte Name des Enum-Typs.

Hinweis

In jedem Fragment, in dem eine PLC-Variable Enum-Texte referenziert, muss diese PLC-Variable mit dem entsprechenden AWP-Kommando dem Enum-Typ-Namen zugeordnet werden.

Beachten Sie, dass zwischen einer Enum-Zuordnung und Enum-Verwendung kein AWP-Kommando zum Import von Fragmenten steht, da durch diesen Import die Enum-Zuordnung in einem anderen Fragment liegt als die Enum-Verwendung.

Beispiel

Es ist ein Enum-Typ "zustand" mit den Werten "0" und "1" definiert. "0" entspricht "aus", "1" entspricht "ein":

```
<!-- AWP_Enum_Def Name="zustand" Values='0:"aus", 1:"ein"' -->
```

Im HTML-Code der auszugebenden Webseite ist der folgende Code enthalten:

```
<!-- AWP_Enum_Ref Name="betriebszustand" Enum="zustand" -->
:=betriebszustand:
```

Das Anzeigeergebnis ist je nach Wert der Variable "betriebszustand" nun nicht mehr "0" oder "1", sondern "aus" oder "ein".

Vereinfachte Nutzung von Enumerationstypen

Bei S7-1200-CPUs ab Firmware-Version 4 ist es möglich, Enumerationen direkt in AWP-Kommandos zum Lesen und Schreiben von PLC-Variablen zu verwenden.

Sie legen wie im vorigen Abschnitt beschrieben Enums an und können dann mit Lese- und Schreib-AWP-Kommandos die Werte nutzen.

Enums anlegen

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Enums in AWP-Schreib- und Lese-Kommandos nutzen

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

Beispiel zum Lesen von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_Out_Variable Name='"Alarm"'
Enum="AlarmEnum" -->...<p>The current value of "Alarm"
is :="Alarm":</p>
```

Wenn der Wert von "Alarm" in CPU "2" ist, wird auf der HTML-Seite folgender Text angezeigt:

'The current value of "Alarm" is Tank is empty' weil die Enum-Definition den String "Tank is empty" dem numerischen Wert 2 zuordnet.

Beispiel zum Schreiben von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_In_Variable Name='"Alarm"'
Enum='AlarmEnum' -->...
```

```
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /></p>
</form>
```

Weil die Enum-Definition den String "Tank is full" dem numerischen Wert "1" zuordnet, wird der Wert "1" in die PLC-Variable "Alarm" geschrieben.

Definieren von Fragmenten

Fragmente

Unter Fragmenten versteht man einzeln von der CPU zu bearbeitende "Sinnabschnitte" einer Webseite.

Fragmente sind meist ganze Seiten, können aber auch Einzelelemente sein, wie Dateien (z. B. Bilder) oder ganze Dokumente.

Fragmente definieren

```
<!-- AWP_Start_Fragment Name="<Name>" [Type="<Type>"] [ID="<Id>"]
[Mode=<Mode>]-->
```

Der Beginn eines Fragmentes wird mit diesem Kommando festgelegt. Ein Fragment erstreckt sich bis zum Beginn des nächsten Fragments oder bis zum Ende der Datei.

- <Name> Gibt den Namen des Fragments an.
  Der Name muss mit einem Buchstaben [a-zA-Z] oder einem Unterstrich (_) beginnen. Nach diesem ersten Zeichen können Buchstaben, Unterstriche oder Zahlen [0-9] folgen.
- <Type> Gibt den Typ des Fragments an.
  - "manual" Das Anwenderprogramm wird über den Request für ein Fragment benachrichtigt, ein Eingriff in die zurückzuliefernde Webseite durch das Anwenderprogramm ist möglich.
  - "automatic" Die Seite wird automatisch bearbeitet (default).

- <id> Für das Fragment kann eine numerische ID vorgegeben werden. Wenn keine ID vergeben wird, dann wird dem Fragment automatisch eine ID zugewiesen. Bei manuellen Seiten (<Type>=manual) kann das Fragment im Anwenderprogramm der CPU mit dieser ID angesprochen werden.

  Hinweis

  Setzen Sie die ID niedrig an, da die höchste ID die Größe des Web-Control-DBs beeinflusst.

- <Mode> Fragmente unterstützen die beiden Modi visible und hidden.
  - "visible" Das Fragment ist ein Teil der Webseite. Dieser Mode ist voreingestellt und kann auch weggelassen werden.
  - "hidden" Das Fragment ist nicht Teil der Webseite. Das Fragment wird aber im Web-DB gespeichert und steht dem Anwenderprogramm zum Einfügen in eine angeforderte Webseite zur Verfügung. Über einen Austausch der Fragment-ID (Variable Web-Control-DB.fragment_index) fügen Sie ein "hidden"-Fragment in die angeforderte Webseite ein.

Das Input Dokument wird durch das "AWP_Start_Fragment"-Kommando vollständig in Fragmente aufgeteilt. Daher ist ein "AWP_End_Fragment" nicht notwendig.

Ohne ein Start-Fragment-Kommando wird eine Datei als ein Fragment abgebildet, der Fragmentname leitet sich aus dem Dateinamen ab. Wird eine Datei in mehrere Fragmente zerlegt (durch "AWP_Start_Fragment"), so muss die Datei mit dem "AWP_Start_Fragment"-Kommando beginnen.

Importieren von Fragmenten

Sie haben die Möglichkeit, in einer HTML-Seite ein Fragment zu deklarieren und dieses Fragment in anderen Webseiten zu importieren.

Beispiel

Ein Firmen-Logo soll auf allen Webseiten einer Webapplikation angezeigt werden.

Der HTML-Code für das Fragment, das das Firmen-Logo anzeigt, existiert nur einmal. Sie können das Fragment so oft importieren und in so vielen HTML-Dateien importieren wie erforderlich.

Syntax

```
<!-- AWP_Import_Fragment Name = "<name>"-->
```

- <name> ist der Name des zu importierenden Fragments.

Beispiel

HTML-Code innerhalb einer Webseite, der ein Fragment deklariert:

```
<!-- AWP_Start_Fragment Name = "My_Company_Logo" -->
<p><img src = "compay_logo.jpg"></p>
```

Beispiel

HTML-Code innerhalb einer anderen Webseite, der das deklarierte Fragment importiert:

```
<!-- AWP_Import_Fragment Name = "My_Company_Logo" -->
```

Datenbausteine generieren und laden

Voraussetzung

- Sie haben alle benötigten Quelldateien (*.html, *.js, *.png, ...) für die Webapplikation erstellt.
- Die Quelldateien liegen in einem Ordner; und zwar nur die Quelldateien, die für die Webapplikation benötigt werden! Es dürfen keine anderen Dateien in diesem Ordner liegen.

Hinweis

Länge von Dateinamen und Variablennamen

Wenn Sie eine umfangreiche Webapplikation mit vielen Dateien und Verzeichnissen haben, scheitert die Generierung der Web-Datenbausteine möglicherweise. In diesem Fall wird die Generierung mit der Meldung "Überlauf Textliste..." abgebrochen. Ursache sind systeminterne Größenbeschränkungen für Verwaltungsinformationen, die im Web-Datenbaustein gespeichert werden.

Abhilfe: Verwenden Sie kurze Dateinamen und kurze Variablennamen.

Vorgehen

Um aus den Quelldateien für anwenderdefinierte Webseiten in STEP 7 in Datenbausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU z. B. in der Gerätekonfiguration.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" die Eigenschaften für anwenderdefinierte Webseiten.
3. Als "HTML-Quelle" wählen Sie den Ordner, wo sich die Quelldateien für die Webapplikation befinden.
4. Als Start-HTML-Seite tragen Sie die HTML-Seite ein, die beim Start der Webapplikation geöffnet werden soll.

5. Wenn erforderlich, tragen Sie einen Applikationsnamen ein.
6. Als "Dateien mit dynamischem Inhalt" ergänzen Sie die Reihe von Dateinamenserweiterungen, falls erforderlich. Sie tragen nur solche Dateinamenserweiterungen ein, die auch AWP-Kommandos enthalten.
7. Die Nummer für den Web-Control-DB und für die Fragment-DB-Startnummer können Sie beibehalten, wenn Sie nicht bereits durch ihr Anwenderprogramm belegt werden.
8. Klicken Sie auf die Schaltfläche "Erzeugen", um DBs aus den Quelldateien zu erzeugen. Die erzeugten Datenbausteine werden in der Projektnavigation im Ordner "Systembausteine" (Unterordner "Webserver") abgelegt.
9. Markieren Sie in der Netzsicht die CPU, die geladen werden soll, und wählen Sie den Befehl "Laden in Gerät" im Menü "Online", um die Bausteine zu laden. Implizit wird vor dem Laden die Übersetzung der Bausteine angestoßen.
   Falls bei diesem Prozess Fehler gemeldet werden, müssen Sie diese Fehler beheben, bevor Sie die Konfiguration laden können.

Datenbausteine generieren und laden

Voraussetzung

- Sie haben alle benötigten Quelldateien (*.html, *.js, *.png, ...) für die Webapplikation erstellt.
- Die Quelldateien liegen in einem Ordner; und zwar nur die Quelldateien, die für die Webapplikation benötigt werden! Es dürfen keine anderen Dateien in diesem Ordner liegen.

Hinweis

Länge von Dateinamen und Variablennamen

Wenn Sie eine umfangreiche Webapplikation mit vielen Dateien und Verzeichnissen haben, scheitert die Generierung der Web-Datenbausteine möglicherweise. In diesem Fall wird die Generierung mit der Meldung "Überlauf Textliste..." abgebrochen. Ursache sind systeminterne Größenbeschränkungen für Verwaltungsinformationen, die im Web-Datenbaustein gespeichert werden.

Abhilfe: Verwenden Sie kurze Dateinamen und kurze Variablennamen.

Vorgehen

Um aus den Quelldateien für anwenderdefinierte Webseiten in STEP 7 in Datenbausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU z. B. in der Gerätekonfiguration.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" die Eigenschaften für anwenderdefinierte Webseiten.
3. Als "HTML-Quelle" wählen Sie den Ordner, wo sich die Quelldateien für die Webapplikation befinden.
4. Als Start-HTML-Seite tragen Sie die HTML-Seite ein, die beim Start der Webapplikation geöffnet werden soll.

5. Wenn erforderlich, tragen Sie einen Applikationsnamen ein.
6. Als "Dateien mit dynamischem Inhalt" ergänzen Sie die Reihe von Dateinamenserweiterungen, falls erforderlich. Sie tragen nur solche Dateinamenserweiterungen ein, die auch AWP-Kommandos enthalten.
7. Die Nummer für den Web-Control-DB und für die Fragment-DB-Startnummer können Sie beibehalten, wenn Sie nicht bereits durch ihr Anwenderprogramm belegt werden.
8. Klicken Sie auf die Schaltfläche "Erzeugen", um DBs aus den Quelldateien zu erzeugen. Die erzeugten Datenbausteine werden in der Projektnavigation im Ordner "Systembausteine" (Unterordner "Webserver") abgelegt.
9. Markieren Sie in der Netzsicht die CPU, die geladen werden soll, und wählen Sie den Befehl "Laden in Gerät" im Menü "Online", um die Bausteine zu laden. Implizit wird vor dem Laden die Übersetzung der Bausteine angestoßen.
   Falls bei diesem Prozess Fehler gemeldet werden, müssen Sie diese Fehler beheben, bevor Sie die Konfiguration laden können.

Aufbau des PLC-Programms

Ihr Anwenderprogramm muss die Anweisung " WWW" aufrufen, damit die Webapplikation, d. h. die anwenderdefinierten Webseiten, überhaupt auf den Standard-Webseiten der CPU verfügbar sind und aufgerufen werden können.

Der Web-Control-DB, den Sie aus den Quelldateien generiert haben, ist der Eingangsparameter (CTRL_DB) für die Anweisung "WWW". Der Web-Control-DB referenziert den Inhalt der anwenderdefinierten Webseiten, der in den Fragment-DBs codiert ist, und enthält Status- und Steuerinformationen.

Aufruf der Anweisung "WWW" im Anlaufprogramm

Wenn Sie angeforderte Webseiten nicht durch das Anwenderprogramm beeinflussen wollen, dann genügt ein einmaliger Aufruf der Anweisung "WWW" in einem Analuf-OB. Die Anweisung initialisiert die Kommunikation zwischen dem Webserver und der CPU.

Aufruf der Anweisung "WWW" im zyklischen Programm

Der Aufruf der Anweisung "WWW" kann auch in einem zyklisch bearbeiteten OB (z. B. dem OB 1) erfolgen. Dies bietet den Vorteil, dass auf Anfragen des Webservers im Anwenderprogramm eingegangen werden kann. Dazu müssen manuelle Fragmente verwendet werden.

In diesem Fall müssen Sie Informationen aus dem Web-Control-DB auswerten, um die angeforderte Webseite bzw. das angeforderte Fragment zu identifizieren. Andererseits müssen Sie im Anwenderprogramm ein Bit setzen, um nach Bearbeitung der Webseiten-Anforderung die vom Webserver auszuliefernde Webseite explizit freizugeben.

Der Aufbau des Web-Control-DBs ist im folgenden Abschnitt beschrieben.

Web-Control-DB

Der Web-Control-DB (Standardmäßig DB 333) wird von STEP 7 erzeugt und enthält Information zum Aufbau der Anwenderseiten, zum Status der Kommunikation und eventuell aufgetretenen Fehlern.

Neben dem Web-Control-DB werden noch weitere so genannte Fragment-DBs erzeugt. Diese Fragment-DBs (es kann auch nur ein Fragment-DB sein) werden im Web-Control-DB referenziert. Die Fragment-DBs enthalten die in Fragmenten codierten Webseiten und Mediadaten, z. B. Bilder. Der Inhalt der Fragment-DBs ist nicht durch das Anwenderprogramm änderbar. Sie werden automatisch angelegt und dienen nur der Datenhaltung.

Auf die Status- und Steuervariablen des Web-Control-DB greifen Sie symbolisch zu.

Im Folgenden sind die Variablen des Web-Control-DBs aufgeführt, die Sie für die Statusauswertung und für die Steuerung der Interaktion benötigen.

Aus dem Web-Control-DB können Sie zweierlei Informationen gewinnen:

- Globale Statusinformation: Nicht an eine konkrete Webseitenanforderung (Request) gebunden.
- Request Status- und Steuerinformationen: Informationen zu anstehenden Requests.

Globale Statusinformation

"WEB-Control_DB".commandstate.init	Aktiviert und initialisiert die Webapplikation.
"WEB-Control_DB".commandstate.deactivate	Deaktiviert die Webapplikation.
"WEB-Control_DB".commandstate.inititializing	Die Webapplikation wird initialisiert (Web-Control-DB lesen, etc.).
"WEB-Control_DB".commandstate.error	Webapplikation konnte nicht initialisiert werden. Der Grund ist in "WEB-Control_DB".commandstate.last_error codiert.
"WEB-Control_DB".commandstate.deactivating	Die Webapplikation wird beendet.
"WEB-Control_DB".commandstate.initialized	Die Webapplikation wurde initialisiert und ist bereit.
"WEB-Control_DB".commandstate.last_error	Wertetabelle der möglichen Fehler siehe nächste Tabelle.

Last_error	Beschreibung
1	Fragment-DB ist inkonsistent (passt nicht zum Web-Control-DB).
2	Eine Webapplikation mit diesem Namen existiert schon.
3	Speicherproblem beim Initialisieren im Webserver.
4	Die Daten im Web-Control-DB sind inkonsistent.
5	Ein Fragment-DB ist nicht verfügbar (nicht geladen).
6	Ein Fragment-DB hat keine AWP-Kennung.
7	Das Enum Fragment (enthält die Texte und Informationen für die Enum-Typen) ist nicht verfügbar.
8	Ein über die Command-Flags im Web-Control-DB angeforderte Aktion ist im aktuellen Zustand nicht erlaubt.

Last_error	Beschreibung
9	Webapplikation ist nicht initialisiert (wenn keine neue Initialisierung nach Deaktivierung).
10	Webserver ist deaktiviert.
...	Last_error wird zurückgesetzt, wenn die Webapplikation erfolgreich initialisiert wurde.

Request-Statusinformation

Die Request-Statusinformation ist gebunden an einen von vier möglichen Requests, x = [1 … 4].

"WEB-Control_DB".requesttab[x].idle	Es gibt nichts zu tun.
"WEB-Control_DB".requesttab[x].waiting	Das Anwenderprogramm muss auf einen Request aus einem manuellen Fragment reagieren und die Weiterverarbeitung im Webserver explizit anstoßen.
"WEB-Control_DB".requesttab[x].sending	Der Webserver ist mit der Bearbeitung des Requests/Fragments beschäftigt.
"WEB-Control_DB".requesttab[x].aborting	Die TCP-Verbindung wird vom Webserver geschlossen.

Request-Steuerinformation

Die Request-Steuerinformation ist gebunden an einen von vier möglichen Requests, x = [1 … 4].

"WEB-Control_DB".requesttab[x].continue	Gibt das gerade bearbeitete Fragment zum Senden frei. Die Bearbeitung des nächsten Fragments wird angestoßen.
"WEB-Control_DB".requesttab[x].repeat	Gibt das gerade bearbeitete Fragment zum Senden frei. Das Fragment wird anschließend erneut bearbeitet.
"WEB-Control_DB".requesttab[x].abort	Schließt die TCP-Verbindung.
"WEB-Control_DB".requesttab[x].finish	Gibt das gerade bearbeitete Fragment zum Senden frei. Stoppt die weitere Bearbeitung von Fragmenten (beendet den Request).

Beispiel:

Die Variable für den DB ist: "WEB-Control_DB". Ob bei der Initialisierung der Webapplikation Fehler aufgetreten sind, können Sie anhand des Bits "WEB-Control_DB".commandstate.error im Anwenderprogramm abfragen.

Für den Fall, dass ein Fehler aufgetreten ist, können Sie den Fehler über den Wert von "WEB-Control_DB".commandstate.last_error analysieren.

Interaktion mit dem Anwenderprogramm

Mithilfe von manuellen Fragmenten erreichen Sie, dass das Anwenderprogramm synchron auf Browsereingaben reagiert, sodass die zurückzugebende Webseite vom Anwenderprogramm aufbereitet werden kann.

Fragmenttyp

Um im Anwenderprogramm auf die empfangenen Daten zu reagieren, muss für das Fragment, das die Daten schreibt, der Fragmenttyp "manual" (für "manuelle Seiten") verwendet werden:

```
<!-- AWP_Start_Fragment Name="testfrag" ID="1" Type="manual" -->
```

Das Übertragen der Werte an den Webserver der CPU erfolgt bei automatischen und manuellen Seiten immer auf die gleiche Weise:

Beispiel:

```
<form method="POST" action="">
 <p>
  <input type="submit" value="Set new value">
  <input type="text" name='"Velocity"' size="20">
 </p>
</form>
```

Anwenderprogramm für manuelle Fragmente

Im Anwenderprogramm der CPU muss bei der Verwendung manueller Seiten die Anweisung "WWW" zyklisch aufgerufen werden.

Um auf im Browser eingegebene Werte zu regieren, ist es notwendig, den Request, den die manuelle Seite an den Webserver stellt, im Anwenderprogramm auszuwerten. Dazu muss der Web-Control-DB (z. B. DB 333) zyklisch auf anstehende Requests untersucht werden. Im Abschnitt "requesttab" des Web-Control-DBs befindet sich das Array, das vier Requests verwaltet. Jedes Element des Arrays enthält dabei, in einer Struktur, Informationen zum jeweiligen Request.

Ein einfaches Programmbeispiel zeigt, wie anhand der Variablen des Web-Control-DBs auf anstehende Requests geprüft wird.

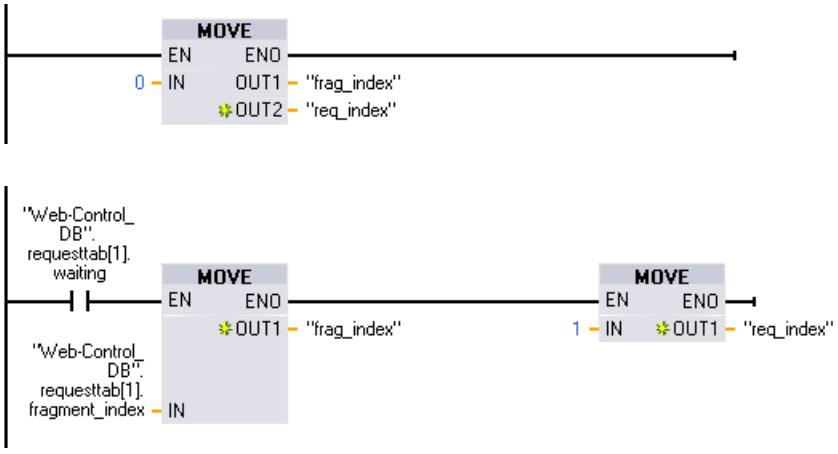

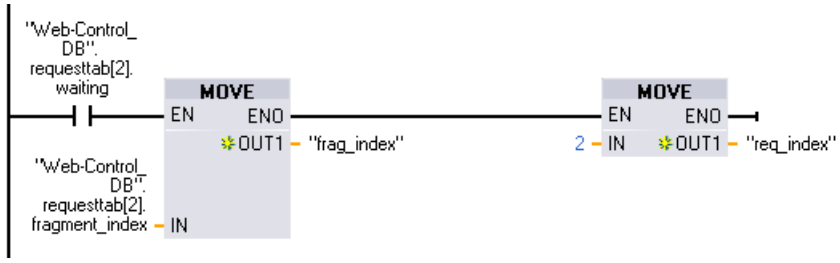

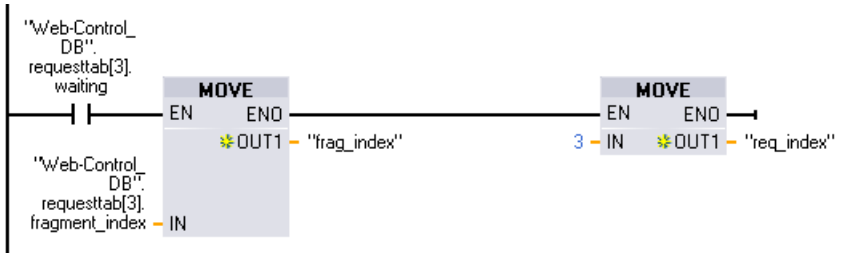

Für den Fall, dass ein Request vorliegt, schreibt dieser Programmabschnitt die Fragment-ID in die Variable #frag_index und die Request-Nr. (Wertebereich 1-4) in die Variable #req_index.

Mit den so gewonnenen Informationen kann die im Request übertragene Information jetzt im Programm für jede Fragment-ID gesondert bearbeitet werden (z. B. Plausibilitätsprüfung).

Wenn die Bearbeitung des Requests durch das Programm abgeschlossen ist, muss der Request beantwortet und der entsprechende Eintrag unter "requesttab" des Web-Control-DBs (z. B. DB 333) wieder zurückgesetzt werden.

Ein einfaches Programmbeispiel zur Beantwortung von Requests:

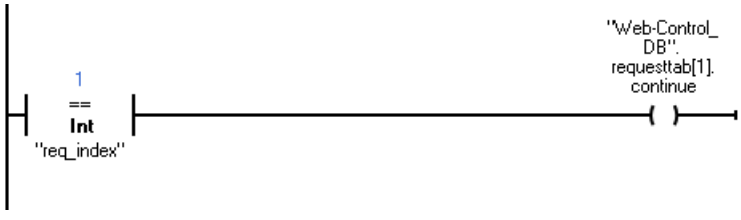

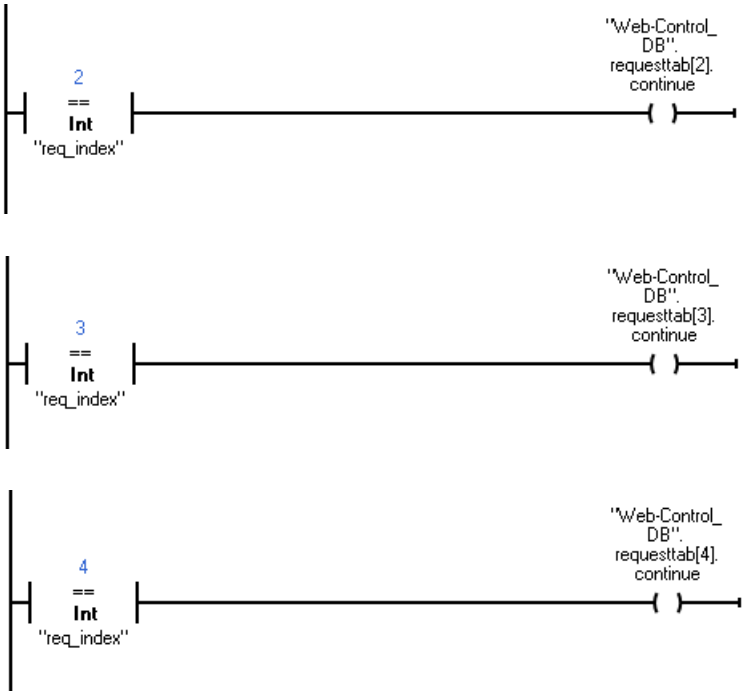
2
==
Int
"req_index"
"Web-Control_
DB".
requesttab[2].
continue
3
==
Int
"req_index"
"Web-Control_
DB".
requesttab[3].
continue
4
==
Int
"req_index"
"Web-Control_
DB".
requesttab[4].
continue

Prinzipieller Ablauf eines Browser-Requests mit Interaktion des Anwenderprogramms

Das folgende Bild zeigt den vereinfachten, prinzipiellen Ablauf vom Webbrowser-Request über die Auswirkungen auf Web-Control-DB-Inhalte und die notwendigen Aktionen des Anwenderprogramms bis zum Zurückschicken der aufbereiteten Webseite (Response).

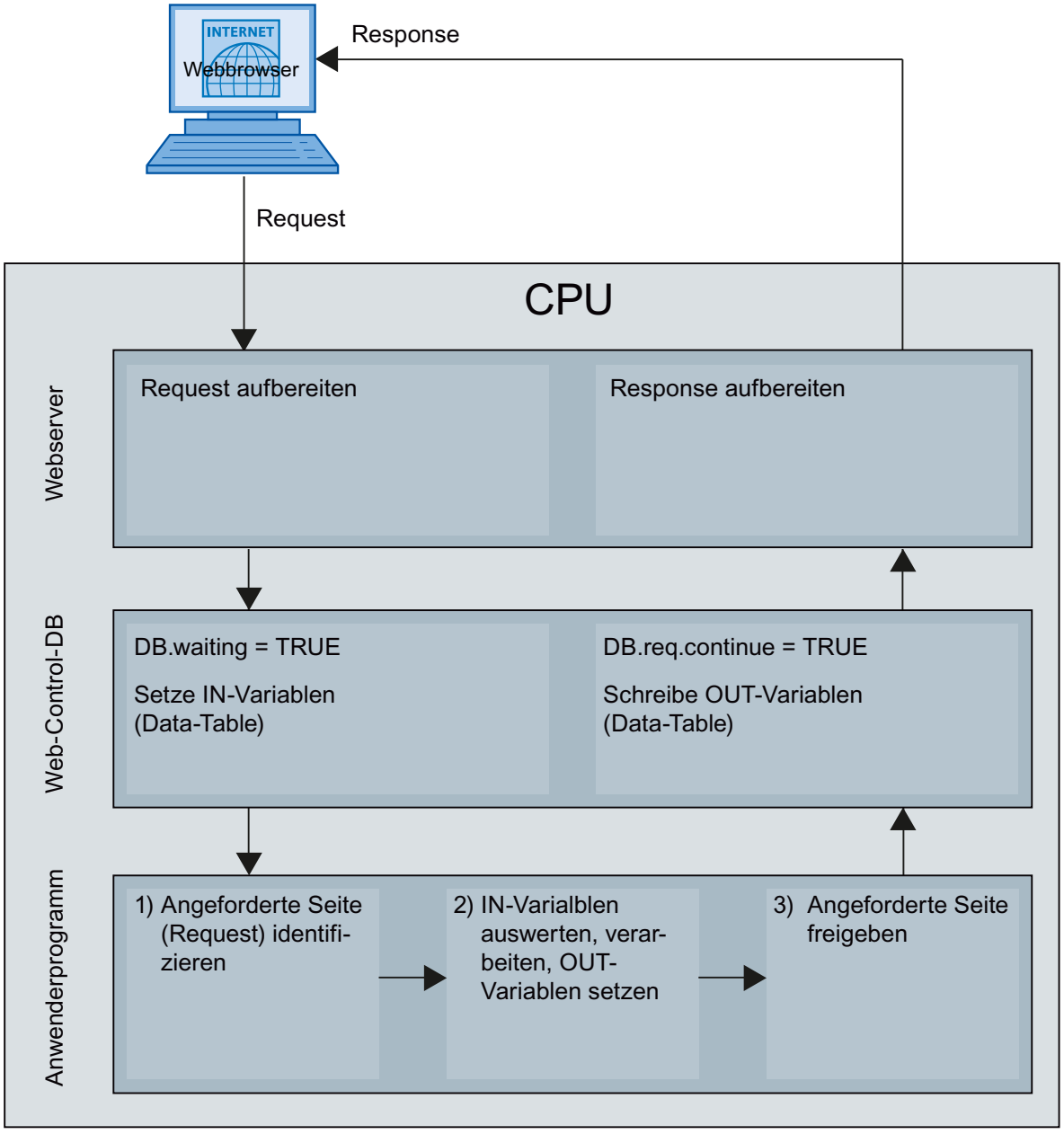

Anzeigen benutzerdefinierter Webseiten im Browser

Webseiten im Browser anzeigen

Die Webseiten rufen Sie von den Standard-Webseiten des Webbrowsers auf.

Die Standard-Webseiten verfügen neben den anderen Links in der Navigationsleiste einen Link auf "Anwenderseiten".

Wenn Sie auf den Link "Anwenderseiten" klicken, öffnet ihr Webbrowser die Webseite, die Sie als Start-HTML-Seite konfiguriert haben.

Anwenderdefinierte Webseiten mehrsprachig erstellen

Sie haben die Möglichkeit, Ihre anwenderdefinierten Webseiten in jeweils unterschiedlichen Landessprachen zur Verfügung zu stellen.

Voraussetzungen

Die sprachabhängigen HTML-Seiten sind in einer Ordnerstruktur abzulegen, die Ordner mit den jeweiligen Sprachkürzeln enthält:

Festgelegte Sprachkürzel

Die Sprachkürzel "de", "en", "fr", "es", "it" und "zh" sind festgelegt. Weitere Sprachordner bzw. anders bezeichnete Sprachordner werden nicht unterstützt.

Weitere Ordner in derselben Ordnerhierarchie für andere Dateien können Sie bei Bedarf anlegen; z. B. einen Ordner "img" für Bilder und einen Ordner "script" für Javascript-Dateien.

Einstiegsseite auswählen

Anwenderseite als Startseite definieren

Neben der voreingestellten Introseite können Sie auch die Startseite Ihrer Anwenderseiten als Startseite des Webservers festlegen.

Um die Anwenderseiten in STEP 7 als Startseite des Webservers zu definieren, gehen Sie wie folgt vor:

1. Wählen Sie die CPU in der Gerätekonfiguration aus.
2. Rufen Sie die Einstellungen im Inspektorfenster der CPU unter "Eigenschaften > Allgemein > Webserver" auf.
3. Wählen Sie im Bereich "Einstiegsseite" unter "Einstiegsseite auswählen" den Eintrag "AWP1".

Wenn Sie jetzt im Browser die IP-Adresse der CPU eingeben, wird automatisch eine Verbindung zu Ihren Anwenderseiten hergestellt.

Wenn Sie die Webseiten der CPU wieder erreichen wollen, dann verlinken Sie die Webseiten von Ihren Anwenderseiten aus z. B. über die URL "http://a.b.c.d./Portal/Portal.mwsl?PriNav=Start" oder "https://a.b.c.d/Portal/Portal.mwsl?PriNav=Start". Die Angabe "a.b.c.d" steht dabei exemplarisch für die IP-Adresse der projektierten CPU.

Beispiel der Verlinkung in HTML:
```
<a href="/Portal/Portal.mwsl?PriNav=Start">SIMATIC-Webseiten</a>
```

Hinweis

Wenn Sie Ihre Anwenderseite als Startseite des Webservers festlegen, werden alle Direktzugriffe auf die Webseiten der CPU blockiert. Dies gilt auch für von Ihnen gespeicherte Lesezeichen für die Webseiten der CPU sowie die Seite zum Auslesen der Servicedaten.

Servicedaten auslesen

Wenn Sie Ihre Anwenderseite als Startseite des Webservers festlegen, wird der Direktzugriff auf die Seite zum Auslesen der Servicedaten ebenfalls blockiert.
Wollen Sie im Servicefall weiterhin Servicedaten über den Webserver auslesen, so verlinken Sie die Servicedaten-Seite direkt aus Ihrer Anwenderseite.
Wie bei den Webseiten der CPU verlinken Sie die Servicedaten-Seite z. B. über die URL "http://a.b.c.d/save_service_data" oder "https://a.b.c.d/save_service_data", die Angabe "a.b.c.d" steht dabei exemplarisch für die IP-Adresse der projektierten CPU.

Beispiel der Verlinkung in HTML:
```
<a href="/save_service_data">Servicedaten</a>
```

Sprachumschaltung für anwenderdefinierte Webseiten

Voraussetzungen

Die HTML-Seiten befinden sich in den vordefinierten Sprachordnern, z. B. HTML-Seiten mit deutschem Text im Ordner "de", HTML-Seiten mit englischem Text im Ordner "en".

Konzept der Sprachumschaltung

Die Sprachumschaltung basiert auf einem vordefinierten Cookie mit Namen "siemens_automation_language". Wenn das Cookie auf den Wert "de" gesetzt wird, schaltet der Webserver bei der nächsten Webseiten-Anforderung bzw. Webseiten-Aktualisierung auf die Webseite aus dem Ordner "de".

Analog dazu schaltet der Webserver auf die Webseite aus dem Ordner "en" um, wenn das Cookie den Wert "en" annimmt.

Beispiel für eine Sprachumschaltung

Das Beispiel ist folgendermaßen aufgebaut:

- In den beiden Sprachordnern "de" und "en" befinden sich die sprachabhängigen HTML-Dateien gleichen Namens, z. B. "langswitch.html". Die anzuzeigenden Texte innerhalb der beiden Dateien sind entsprechend den Ordnernamen deutsch bzw. englisch.
- Zusätzlich befindet sich ein Ordner "script" in der Ordnerstruktur, in dem sich die Javascript-Datei "lang.js" befindet. In diese Datei sind Funktionen ausgelagert, die für die Sprachumschaltung benötigt werden.

Aufbau der Datei "langswitch.html" (Ordner "de")

Im Header der Datei sind die Metadaten "Content-Language", charset und Pfad zur Javascriptdatei festgelegt.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

Im Body der Datei ist die Sprachauswahl mithilfe des HTML-Elements "select" realisiert. Das select-Element leitet eine Auswahlliste ein und enthält die Optionen "de", beschriftet mit "Deutsch" und "en", beschriftet mit "Englisch"; "de" ist vorselektiert.

Mit Hilfe des event-handlers "onchange" wird die Funktion "DoLocalLanguageChange(this)" aufgerufen. Der Parameter "this" übergibt das select-Objekt mit der selektierten Option an diese Funktion. "onchange" führt bei jeder Änderung der Option zum Aufruf der Funktion.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" selected >Deutsch</option>
                <option value="en" >Englisch</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Aufbau der Datei "langswitch.html" (Ordner "en")

Der Header der HTML-Datei mit englischem Text ist analog zur HTML-Datei mit deutschem Text aufgebaut.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
```

```
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

Im Body der Datei ist die Sprachauswahl ebenfalls mit Hilfe des HTML-Elements "select" realisiert. Im Unterschied zur deutschen HTML-Datei ist die englische Option vorselektiert und der Text bzw. die Labels sind englisch.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" >German</option>
                <option value="en" selected >English</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Aufbau der Datei "lang.js" (im Ordner "script")

In der Javascript-Datei ist die Funktion " DoLocalLanguageChange" definiert, die ihrerseits die Funktion "SetLangCookie" mit dem Sprachauswahl-Wert aufruft. SetLangCookie setzt Cookie-Name und Cookie-Wert zusammen und setzt das Cookie schließlich über die entsprechende document.cookie-Eigenschaft. Damit der Webserver auf das Setzen des Cookies mit der Anzeige der gewünschten Sprache reagiert, muss anschließend die Webseite neu geladen werden (top.window.location.reload).

```
function DoLocalLanguageChange(oSelect) {
        SetLangCookie(oSelect.value);
        top.window.location.reload();
    }

 function SetLangCookie(value) {
        var strval = "siemens_automation_language=";
            // this is the cookie by which the webserver
            // detects the desired language
            // this name is required by the webserver
        strval = strval + value;
        strval = strval + "; path=/ ;";
            // set path to the application, since otherwise
            // path would be set to the requesting page
            // would not get the cookie.
            // The path for user defined applications follows this
sample:
            // path=/awp/<application name>/<pagename>
```

```
                // example: path=/awp/myapp/myappstartpage.htm
                //(where myapp is the name of the web application
                // entered in the web server properties of the cpu)
                /*
                use expiration if this cookie should live longer
                than the current browser session
                var now      = new Date();
                var endttime = new Date(now.getTime() + expiration);
                strval = strval + "; expires=" + endttime.toGMTString()
+ ";";
                */
            document.cookie = strval;
    }
```

Weitere Beispiele

Flaschen mit unterschiedlichen Getränken füllen

Dieses Beispiel beschreibt eine Abfüllanlage (http://support.automation.siemens.com/WW/view/de/94681612), die über eine Anwenderdefinierte Webseite überwacht und gesteuert werden kann.

Es ist möglich, die Art der abzufüllenden Getränke über eine Rezeptur zu bestimmen.

Für die Steuerung und Visualisierung der Anlage ist kein Panel (HMI) erforderlich.

Ein Webbrowser dient zum Anzeigen der Webseite.

Die Webseite wird regelmäßig mit aktuellen Prozesswerten versorgt.

Dazu dient ein unsichtbarer Inlineframe (iframe), der regelmäßig vom Webserver neu gesendet wird. Mit dem Inlineframe gelangen aktuelle Prozesswerte zum Browser. Per JavaScript werden die Prozesswerte aus dem Inlineframe gelesen und in die Hauptseite eingefügt.

Zudem können Anwender über die Webseite Funktionen in der CPU aufrufen, zum Beispiel das Abfüllen von Flaschen starten oder stoppen oder Rezepturen importieren oder exportieren (siehe Link Anwendung Abfüllanlage (http://support.automation.siemens.com/WW/view/de/94681612)).

Einen Tank füllen und leeren

Das Beispiel beschreibt einen Tank für Flüssigkeiten, (http://support.automation.siemens.com/WW/view/de/58862931)der über eine Anwenderdefinierte Webseite überwacht und gesteuert werden kann. Der Anwender steuert über Schaltflächen den Zulauf in den Tank und den Ablauf aus dem Tank.

Der aktuelle Füllstand wird angezeigt, ebenso wie die letzten zehn Füllstände mit den jeweiligen Zeitangaben.

Die Seiten "Overview" und "Data" lesen in festen Abständen die aktuellen Prozesswerte ein, wobei der Webserver die Seiten jeweils komplett neu an den Browser sendet.

Die Seite "DataOpti" nützt dagegen einen Inlineframe, um lediglich die aktuellen Füllstände des Tanks vom Webserver neu zu erhalten (siehe Anwendung Tanksimulation (http://support.automation.siemens.com/WW/view/de/58862931)).

1.4.2.5 Konfigurationssteuerung (Optionenhandling) für Geräte

Wissenswertes zur Konfigurationssteuerung (Optionenhandling)

Funktionsprinzip

Durch die Konfigurationssteuerung ist es mit einer S7-1200 ab Firmware-Version 4.1 möglich, den Aufbau einer Steuerung zu konfigurieren und von dieser Konfiguration abweichende Varianten (Optionen) zu betreiben.

- In einem Masterprojekt mit einer Maximalkonfiguration (Stationsmaster) sind alle Module konfiguriert, die in einer Reihe von ähnlichen Anlagenteilen oder Maschinen benötigt werden.
- Im Anwenderprogramm des Masterprojekts sind verschiedene Stationsoptionen für verschiedene Anlagenteile oder Maschinen sowie die Auswahl einer Stationsoption vorgesehen. Eine Stationsoption nutzt z. B. nur einen Teil der konfigurierten Module und diese Module sind in abweichender Reihenfolge gesteckt.
- Vor Ort wählt ein Bediener eine Stationsoption für eine konkrete Anlage aus. Er muss dazu das Projekt nicht ändern und damit auch keine geänderte Konfiguration laden.

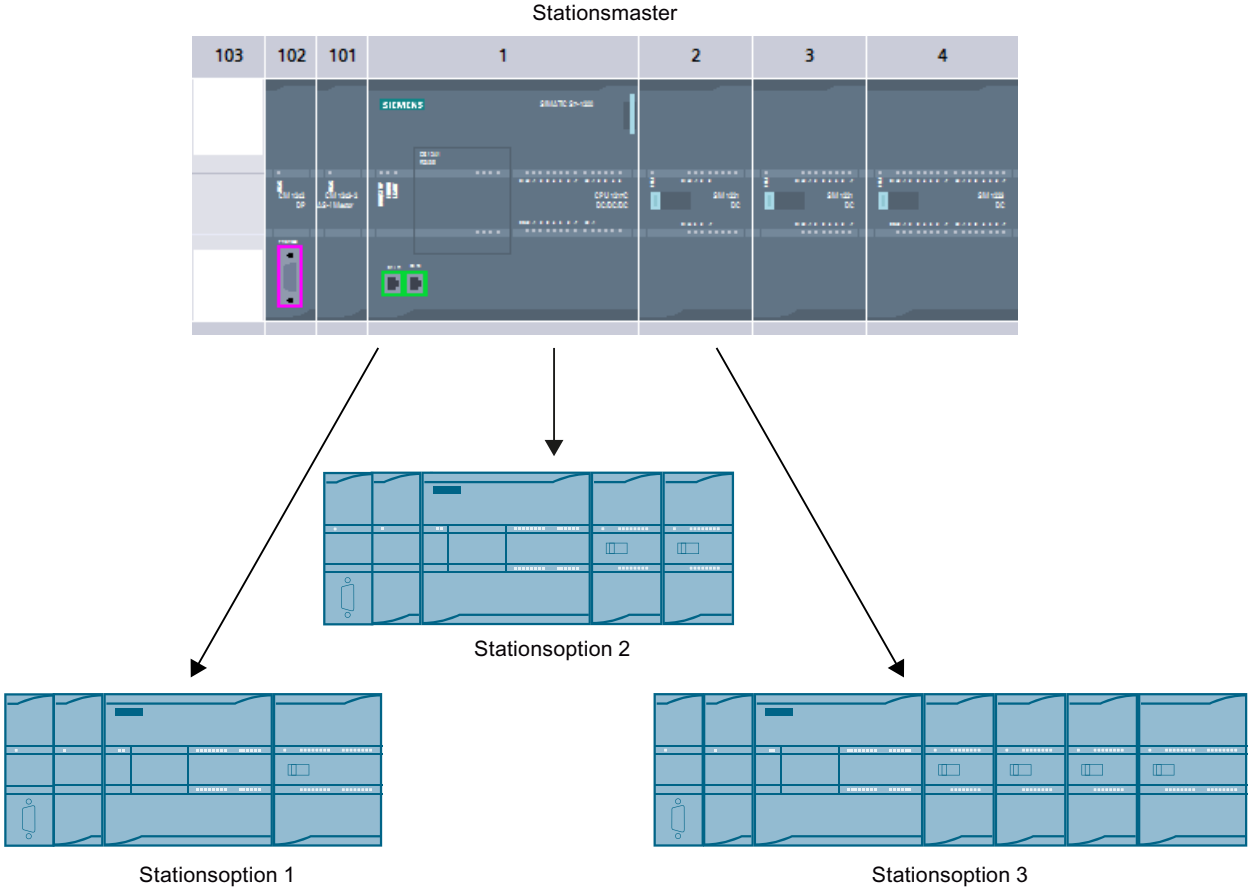

Der CPU wird durch einen von Ihnen programmierten Steuerdatensatz im Anlaufprogramm mitgeteilt, welche Module abweichend von der Projektierung im realen Aufbau fehlen oder welche Module sich abweichend von der Projektierung auf einem anderen Steckplatz befinden. Auf die Parametrierung der Module hat die Konfigurationssteuerung keinen Einfluss.

Durch die Konfigurationssteuerung können Sie flexibel den zentralen Aufbau variieren, solange sich die reale Konfiguration aus der projektierten Konfiguration eines Maximalausbaus ableiten lässt.

Im Folgenden ist beschrieben, wie Sie die Konfigurationssteuerung aktivieren (CPU-Parametrierung) und wie der erforderliche Datensatz aufgebaut sein muss.

Voraussetzung

- STEP 7 ab Version V13 SP1
- CPU S7-12XX ab Firmware-Version 4.1. Module, welche die Funktion "Konfigurationssteuerung" unterstützen, haben auch in der Beschreibung (Infotext) des Hardware-Katalogs den Eintrag "Konfigurationssteuerung".
- Empfehlung: Bevor Sie ein neues Programm mit geändertem Steuerdatensatz laden, führen Sie Urlöschen durch. Durch diese Maßnahme werden inkonsistente Zustände vermieden, die durch einen eventuell vorhandenen, inkompatiblen Steuerdatensatz entstehen.
- Der Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf "Anlauf der CPU auch bei Unterschieden" (Voreinstellung).

Erforderliche Schritte

1. Aktivieren Sie bei der Projektierung der CPU den Parameter "Umkonfigurieren des Geräts über Anwenderprogramm ermöglichen" (Bereich "Konfigurationssteuerung").

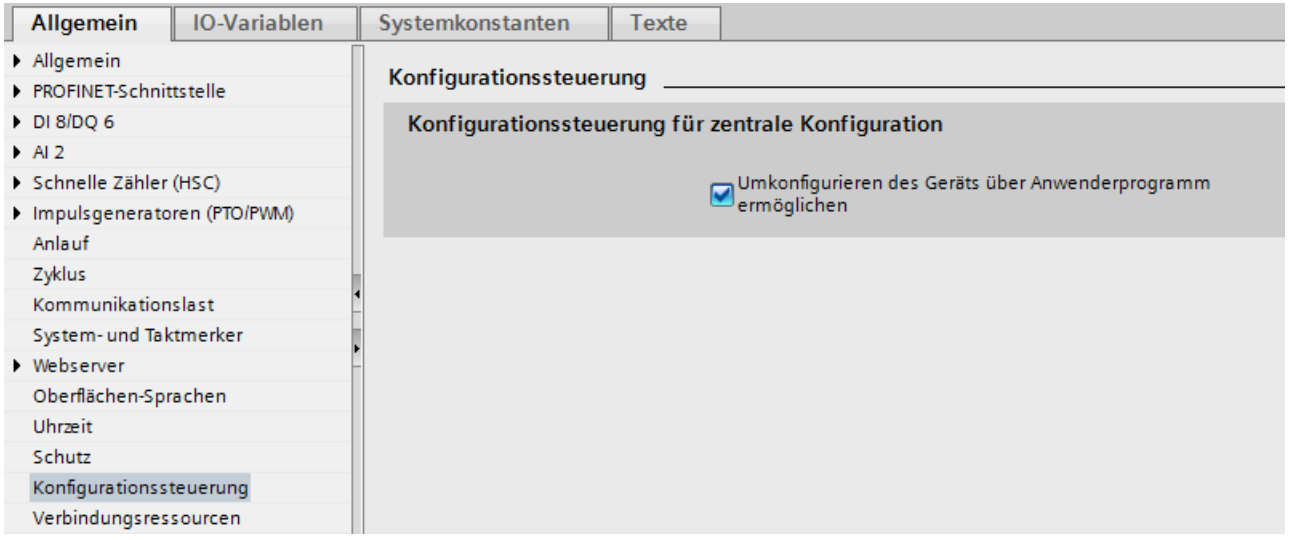

2. Erstellen Sie einen Steuerdatensatz (z. B. in einem Datenbaustein) gemäß der aktuellen Konfiguration nach dem unten beschriebenen Muster für den Steuerdatensatz. Der Steuerdatensatz hat die Nummer 196. Wenn Sie den Steuerdatensatz als ganzen Baustein der Anweisung WRREC übergeben wollen (Eingangsparameter RECORD), dann beachten Sie, dass Sie zunächste einen PLC-Datentyp erstellen müssen, der die Struktur des Steuerdatensatzes enthält und der Datenbaustein auf diesem PLC-Datentyp basiert.

ConfDB

		Name	Datentyp	Startwert	...	...	...	...	Kommentar
1		Static							
2		ConfigControl	Struct			☑	☑		
3		Block_length	USInt	9		☑	☑		Length of control data record, including header
4		Block_ID	USInt	196		☑	☑		Data record number
5		Version	USInt	5		☑	☑		
6		Subversion	USInt	0		☑	☑		
7		Slot_1	USInt	16#FF		☑	☑		Assignment for CPU annex card/Actual annex card
8		Slot_2	USInt	16#FF		☑	☑		Configured slot 2/ Assigned "real" slot
9		Slot_3	USInt	16#FF		☑	☑		Configured slot 3 / Assigned "real" slot
10		Slot_4	USInt	16#FF		☑	☑		Configured slot 4 / Assigned "real" slot
7		Slot_5	USInt	16#FF		☑	☑		Configured slot 5 / Assigned "real" slot
8		Slot_6	USInt	16#FF		☑	☑		Configured slot 6 / Assigned "real" slot
9		Slot_7	USInt	16#FF		☑	☑		Configured slot 7 / Assigned "real" slot
10		Slot_8	USInt	16#FF		☑	☑		Configured slot 8 / Assigned "real" slot
7		Slot_9	USInt	16#FF		☑	☑		Configured slot 9 / Assigned "real" slot
8		Slot_101	USInt	16#FF		☑	☑		Configured slot 101 / Assigned "real" slot
9		Slot_102	USInt	16#FF		☑	☑		Configured slot 102 / Assigned "real" slot
10		Slot_103	USInt	16#FF		☑	☑		Configured slot 103 / Assigned "real" slot

3. Übertragen Sie den Steuerdatensatz im Anlaufprogramm zur CPU.
Die Konfigurationssteuerung für die zentral gesteckten Module wirkt nur beim Betriebszustandswechsel der CPU von STOP nach RUN. Rufen Sie daher im Anlauf-OB die erweiterte Anweisung WRREC (Datensatz schreiben) auf und übertragen Sie den erstellten Steuerdatensatz an die CPU, siehe nächsten Abschnitt.

Wenn im Anlauf-OB kein gültiger Steuerdatensatz übertragen wird, dann ist die Steuerung nicht betriebsbereit. Die CPU kehrt in diesem Fall vom Anlauf zurück in den STOP-Zustand.

Steuerdatensatz im Anlaufprogramm übertragen

Die CPU bearbeitet die Anweisung WRREC zur Übertragung des Steuerdatensatzes asynchron. Sie müssen daher WRREC in einer Schleife im Anlauf-OB wiederholt aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz übertragen ist.

Tipp: Verwenden Sie zur Programmierung der Schleife die Programmiersprache SCL mit der Anweisung REPEAT ... UNTIL.

```
REPEAT
    "WRREC_DB"(REQ := "start_config_control",
               ID := 33,
               INDEX := 196,
               LEN := "conf_LEN",
               DONE => "conf_DONE",
               BUSY => "conf_BUSY",
               ERROR => "conf_ERROR",
               RECORD := "ConfDB".ConfigControl,
               STATUS => "conf_STATUS");
UNTIL NOT "conf_BUSY"
END_REPEAT;
```

Im Folgenden finden Sie Erläuterungen zu einzelnen Bausteinparametern, die Sie im Kontext Konfigurationssteuerung mit bestimmten Werten versorgen müssen. Zu den übrigen Bausteinparametern siehe auch WRREC:

Parameter	Erläuterung
ID	HW-Kennung; ist immer 33 (dezimal) bei Konfigurationssteuerung für zentral angeordnete Module.
INDEX	Datensatznummer; ist immer 196 (dezimal) bei Konfigurationssteuerung für zentral angeordnete Module.
RECORD	Zu übertragender Steuerdatensatz. Zum Aufbau des Steuerdatensatzes siehe Abschnitt "Steuerdatensatz". Tipp: Der Bausteinparameter "RECORD" der Anweisung WRREC (ab V1.1) ist vom Datentyp "VARIANT" und verlangt daher eine Variable mit Datentyp. Wenn Sie den Steuerdatensatz in einem Datenbaustein ablegen, muss dieser Datenbaustein folglich auch einen Datentyp haben. Der erzeugte Datenbaustein darf nicht vom Typ "Global-DB" sein, sondern muss von einem Anwenderdatentyp abgeleitet sein. Gehen Sie wie folgt vor: 1. Erzeugen Sie zunächst einen neuen PLC-Datentyp (Anwenderdatentyp) mit der Struktur des Steuerdatensatzes und benennen ihn z. B. "ConfDB". 2. Erzeugen Sie einen neuen Datenbaustein. Wählen Sie als Typ für diesen Datenbaustein den neu erstellten Anwenderdatentyp, z. B. "ConfDB".

In den grafischen Programmiersprachen setzen Sie die Schleife mithilfe von Anweisungen zur Programmsteuerung um.

Beispiel in FUP: Mit der Anweisung LABEL (Sprungmarke) und mit der Anweisung JMP (Springe bei VKE=1) programmieren Sie eine Schleife.

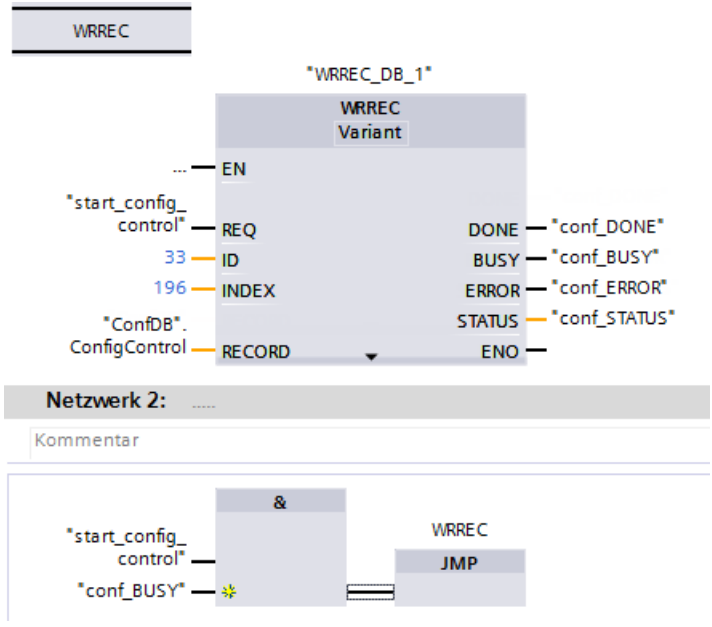

Anordnung der Module

Die nachfolgende Tabelle zeigt die Zuordnung der Steckplatznummern:

Steckplatz	Module	Bemerkung
1	Signalboard, Kommunikationsboard, Batterieboard	Steckplatz auf Vorderseite der CPU
2 - 9	Signalmodule	Steckplätze rechts von der CPU
101 - 103	Kommunikationsmodule	Steckplätze links von der CPU

Steuerdatensatz

Für die Konfigurationssteuerung wird ein Steuerdatensatz 196 definiert, der eine Steckplatzzuordnung enthält.

Es gelten folgende Kodierungen:

0	Modul ist in der Hardware-Konfiguration enthalten aber nicht in der aktuellen Konfiguration genutzt.
1 bis 9, 101 bis 103	Aktueller Steckplatz des Moduls.
16#FF (255)	In der Hardware-Konfiguration befindet sich kein Modul auf diesem Steckplatz.

Byte	Element	Kodierung	Erläuterung
0	Blocklänge	4 + Anzahl der Steckplätze	Header
1	Block-ID	196	
2	Version	5 (für zentrale Peripherie)	
3	Subversion	0	
4	Zuordnung des CPU-Erweiterungsboards	Erweiterungsboard, 0 oder 16#FF	Steuerelement Beschreibt in jedem Element, welcher reale Steckplatz im Gerät dem projektierten Steckplatz zugeordnet ist. Der Aufbau eines Steuerelements ist im folgenden Abschnitt beschrieben.
5	Zuordnung projektierter Steckplatz 2	Realer Steckplatz, 0 oder 16#FF	
...	...	...	
12	Zuordnung projektierter Steckplatz 9	Realer Steckplatz, 0 oder 16#FF	
13	Zuordnung projektierter Steckplatz 101	Realer Steckplatz oder 16#FF	Im Unterschied zu Signalmodulen muss der reale Steckplatz der Kommunikationsmodule dem konfigurierten Steckplatz entsprechen.
14	Zuordnung projektierter Steckplatz 102	Realer Steckplatz oder 16#FF	
15	Zuordnung projektierter Steckplatz 103	Realer Steckplatz oder 16#FF	

Aufbau eines Steuerelements

Ein Steuerelement enthält die Information, welches Modul auf welchem Steckplatz steckt.

Die Byte-Nummern repräsentieren die projektierten Steckplätze in aufsteigender Reihenfolge (siehe oben):

- Byte 4 steht für den projektierten Steckplatz des Erweiterungsboards
- Byte 5 bis Byte 9 stehen für die projektierten Steckplätze 2 bis 9
- Byte 13 bis Byte 15 stehen für die projektierte Steckplätze 101 bis 103

Welchen Wert Sie im jeweiligen Byte eintragen müssen, ergibt sich aus folgender Regel:

- Wenn das Modul im realen Aufbau vorhanden ist, tragen Sie die reale Steckplatznummer des Moduls ein.
  - Beispiel 1: Das Modul vom projektierten Steckplatz 2 befindet sich auf Steckplatz 2. Tragen Sie im Byte 5 (= projektierter Steckplatz 2) den Wert 2 (= tatsächlicher Steckplatz) ein.
  - Beispiel 2: Das Modul vom projektierten Steckplatz 3 befindet sich auf Steckplatz 2. Tragen Sie im Byte 6 (= projektierter Steckplatz 3) den Wert 2 (= tatsächlicher Steckplatz) ein.
- Wenn das Modul konfiguriert ist, aber im realen Aufbau fehlt, dann tragen Sie 0 im Byte für den projektierten Steckplatz ein.
- Wenn sich in der Hardware-Konfiguration kein Modul auf diesem Steckplatz befindet, dann tragen Sie 16#FF (255) im Byte für den projektierten Steckplatz ein.

Regeln

Beachten Sie folgende Regeln:

- Die Konfigurationssteuerung unterstützt nicht die Positionsänderung von Kommunikationsmodulen. Die Steckplatzeinträge im Steuerdatensatz für die Steckplätze 101 bis 103 müssen den realen Positionen der Module entsprechen oder durch den Eintrag von 16#FF (255) als nicht in der Hardware-Konfiguration vorhanden definiert sein.
- Es sind keine Steckplatzlücken in der Konfiguration erlaubt. Wenn zum Beispiel in der realen Konfiguration ein Signalmodul in Steckplatz 4 steckt, müssen auch die Steckplätze 2 und 3 realen Konfiguration belegt sein. Das selbe gilt für die Steckplätze 101 bis 103. Wenn ein Kommunikationsmodul in der realen Konfiguration auf Steckplatz 102 steckt, muss in der realen Konfiguration auch ein Kommunikationsmodul in Steckplatz 101 stecken.
- Wenn Sie die Konfigurationssteuerung aktiviert haben, ist die CPU ohne Steuerdatensatz nicht betriebsbereit. Wenn im Anlauf-OB kein gültiger Steuerdatensatz übertragen wird, kehrt die CPU vom Anlauf zurück in den STOP-Zustand. Die zentrale Peripherie wird in diesem Fall nicht initialisiert. Im Diagnosepuffer wird die Ursache für den Betriebszustand STOP eingetragen.
- Verwenden Sie zur Adressierung der Anweisung WRREC die HW-Kennung 33 (dezimal, für den Bausteinparameter ID), um den Steuerdatensatz zu schreiben.
- Der Steuerdatensatz wird remanent in der CPU gespeichert, so dass bei unveränderter Konfiguration kein erneutes Schreiben des Steuerdatensatzes 196 beim Neuanlauf erforderlich ist. Vor der Inbetriebnahme empfehlen wir, die CPU urzulöschen, um einen eventuell vorhandenen Steuerdatensatz zu löschen.
- Steckplatzeinträge im Steuerdatensatz außerhalb der projektierten Sollkonfiguration ignoriert die CPU.

- Jeder reale Steckplatz darf nur einmal im Steuerdatensatz vorhanden sein.
- Ein realer Steckplatz darf nur einem projektierten Steckplatz zugeordnet werden.

Hinweis

Geänderte Konfiguration

Das Schreiben eines Steuerdatensatzes mit geänderter Konfiguration führt automatisch zu folgender Reaktion der CPU:

Urlöschen mit anschließendem Anlauf mit dieser geänderten Konfiguration.

Durch diese Reaktion wird der remanent gespeicherte ursprüngliche Datensatz 196 gelöscht und der neue Datensatz 196 remanent gespeichert.

Verhalten im Betrieb

Auswirkung der Diskrepanz zwischen projektierter Konfiguration und realer Konfiguration:

- Für die Online-Anzeige und für die Anzeige im Diagnosepuffer (Modul o.k. oder Modul fehlerhaft) wird immer die Hardware-Konfiguration herangezogen; nicht die davon abweichende reale Konfiguration.
  Beispiel: Ein Modul liefert eine Diagnose. Konfiguriert ist dieses Modul auf Steckplatz 4, real steckt es auf Steckplatz 3 (fehlendes Modul; siehe Beispiel im nächsten Kapitel). In der Online-Sicht wird ein projektierter Steckplatz 4 als fehlerhaft angezeigt, im realen Aufbau zeigt das Modul auf Steckplatz 3 über LED-Anzeige einen Fehler an.

Wenn im Steuerdatensatz Module als fehlend eingetragen sind, verhält sich das Automatisierungssystem wie folgt:

- Im Steuerdatensatz als nicht vorhanden gekennzeichnete Module liefern keine Diagnose, ihr Zustand ist immer o.k. Der Wertstatus ist o.k.
- Schreibender Direktzugriff auf die Ausgänge oder schreibender Zugriff auf das Prozessabbild der nicht vorhandenen Ausgänge: Bleibt wirkungslos; es wird kein Zugriffsfehler gemeldet.
- Lesender Direktzugriff auf die Eingänge oder lesender Zugriff auf das Prozessabbild der nicht vorhandenen Eingänge: Wert "0" wird geliefert; es wird kein Zugriffsfehler gemeldet.
- Datensatz auf nicht vorhandenes Modul schreiben: Bleibt wirkungslos; es wird kein Fehler gemeldet.
- Datensatz von nicht vorhandenem Modul lesen: Es wird ein Fehler gemeldet, da kein gültiger Datensatz zurückgeliefert werden kann.

Fehlermeldungen

Beim Schreiben des Steuerdatensatzes werden Ihnen im Fehlerfall folgende Fehlermeldungen zurückgegeben:

Tabelle 1-93 Fehlermeldungen

Fehlercode	Bedeutung
16#80B1	Unzulässige Länge; Die Längenangabe im Datensatz 196 ist nicht korrekt.
16#80B5	Konfigurationssteuerung nicht parametriert.
16#80E2	Datensatz wurde im falschen OB-Kontext übertragen. Der Datensatz muss im Anlaufprogramm übertragen werden.
16#80B8	Parameterfehler; Modul meldet ungültige Parameter.

Siehe auch

Systemhandbuch S7-1200 (http://support.automation.siemens.com/WW/view/de/89851659)

Beispiel für eine Konfigurationssteuerung

Im folgenden wird eine Konfiguration projektiert, bestehend aus CPU und 3 Signalmodulen.

Das Modul auf Steckplatz 3 ist in der ersten Ausbaustufe nicht vorhanden und wird von der Konfigurationssteuerung "ausgeblendet".

In der zweiten Ausbaustufe wird das zunächst ausgeblendete Modul auf dem letzten Steckplatz angeordnet. Der hinzugekommene Steckplatz wird über einen geänderten Steuerdatensatz der CPU bekannt gemacht.

Istkonfiguration mit fehlendem Modul

Die projektierte Konfiguration enthält alle Module, die in einem Endausbau vorhanden sein dürfen.

Das Modul, das in der projektierten Konfiguration auf Steckplatz 3 steckt, fehlt im realen Ausbau. Daher muss der Steckplatz 3 im Steuerdatensatz entsprechend mit "FF_H" (= nicht vorhanden) kodiert werden.

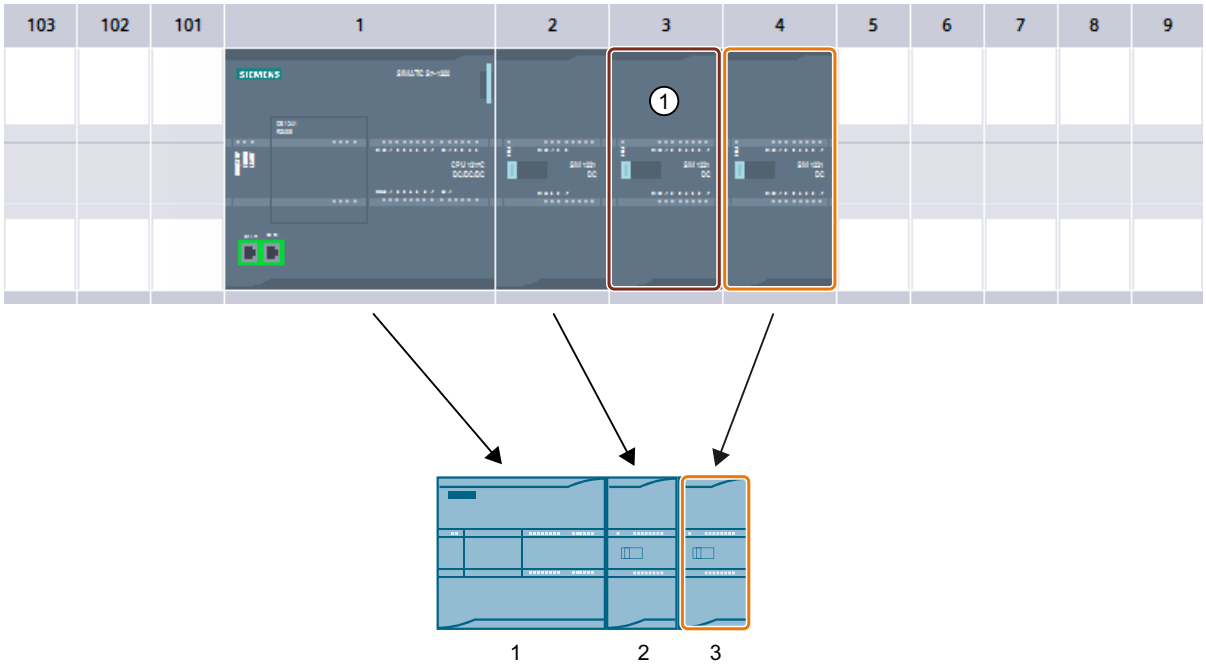

ControlDataRecord

		Name	Data type	Start value	Comment
1		Static			
2		ConfigControl	Struct		
3		Block_length	USInt	16	Length of control data record, including header
4		Block_ID	USInt	196	Data record number
5		Version	USInt	5	
6		Subversion	USInt	0	
7		Slot_1	USInt	255	Assignment for CPU annex card/Actual annex ca...
8		Slot_2	USInt	2	Configured slot 2 / Assigned ①al" slot
9		Slot_3	USInt	0	Configured slot 3 / Assigned "real" slot
10		Slot_4	USInt	3	Configured slot 4 / Assigned "real" slot
11		Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
12		Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
13		Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
14		Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
15		Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
16		Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
17		Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
18		Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

① Modul fehlt im realen Aufbau

Istkonfiguration mit nachträglich hinzugefügtem Modul

Das in der projektierten Konfiguration vorhandene Modul auf Steckplatz 3 wird im realen Aufbau "hinten dran gehängt"; als letztes Modul auf Steckplatz 4 gesteckt.

Der Steuerdatensatz wird entsprechend angepasst.

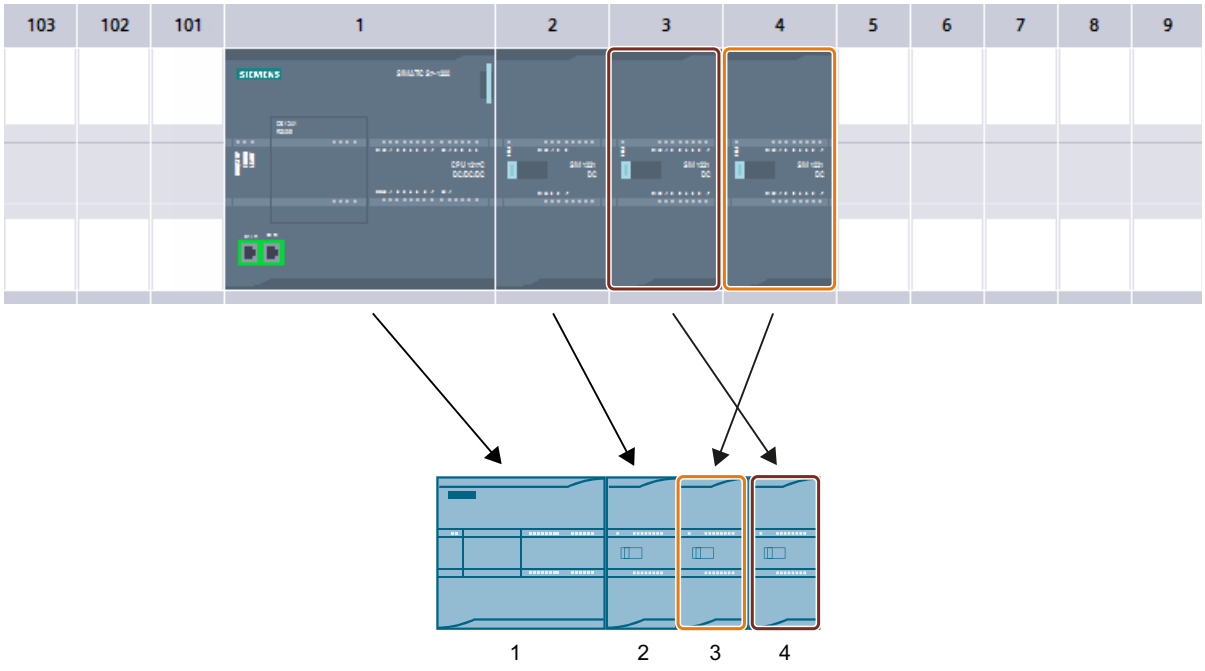

ControlDataRecord

	Name	Data type	Start value	Comment
1	Static			
2	ConfigControl	Struct		
3	Block_length	USInt	16	Length of control data record, including header
4	Block_ID	USInt	196	Data record number
5	Version	USInt	5	
6	Subversion	USInt	0	
7	Slot_1	USInt	255	Assignment for CPU annex card/Actual annex card
8	Slot_2	USInt	2	Configured slot 2/ Assigned "real" slot
9	Slot_3	USInt	4	Configured slot 3 / Assigned "real" slot
10	Slot_4	USInt	3	Configured slot 4 / Assigned "real" slot
11	Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
12	Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
13	Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
14	Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
15	Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
16	Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
17	Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
18	Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

1.4.2.6 Weitere Konfigurationen

Konfigurieren von weiteren Funktionen

Das Automatisierungssystem S7-1200 verfügt über zahlreiche weitere Funktionen, die als integrierte CPU-Funktionen nutzbar sind oder über steckbare Module (z. B. Kommunikationsmodule) zur Verfügung stehen. Die Beschreibung finden Sie über die unten aufgeführten Links.

1.4.3 PC-Systeme konfigurieren

1.4.3.1 Übersicht

Wissenswertes zu PC-Systemen und PC-Stationen

Definition PC-Systeme

PC-Systeme gibt es in verschiedensten Ausführungen, die sich flexibel mit Software- und Hardwarekomponenten ausbauen lassen. Hier behandeln wir PC-Systeme mit einem Software Controller, der einen PC (z. B. IPC) zu einer Steuerung macht - ähnlich einem modularen Controller wie einer CPU S7-300/400 oder einer CPU S7-1500.

Auf einem PC-System sind die PC-Station und mögliche weitere Komponenten installiert.

Definition PC-Station

Eine PC-Station ist eine Softwarekomponente, die SIMATIC Softwareprodukte und die Verwendung von Schnittstellen auf einem PC verwaltet. Zusätzlich regelt sie die Kommunikation der SIMATIC Produkte auf einem PC untereinander.

Die PC-Station wird von den SIMATIC Softwareprodukten SIMATIC NET PC-Software, WinAC RT oder den SIMATIC S7-1500 Software Controllern vorausgesetzt und auf dem Ziel-PC installiert.

SIMATIC HMI Produkte wie WinCC RTX Advanced können mit verschiedenen Versionen der PC-Station zusammen arbeiten.

Ab Version V2.x kann die PC-Station auch ohne Software Controller installiert werden.

Wahl der Version einer PC-Station

Das Anpassen der Version Ihrer PC-Station während der Konfiguration passiert durch die im PC-System konfigurierten Softwarekomponenten automatisch.

Folgendes Szenario zeigt Ihnen einen möglichen Versionswechsel der PC-Station an:

- Sie haben einen IPC aus dem Hardware-Katalog gewählt, der PC-Station V2.x-fähig ist.
  - Im Ausgangszustand des IPCs ist die PC-Station V1.0 vorbelegt.
  - Solange Sie Komponenten in das PC-System stecken, die mit der PC-Station V1.0 kompatibel sind, bleibt es bei PC-Station V1.0.
  - Erst wenn Sie die S7-1500-Variante des Software Controllers in den IPC stecken, passt sich die Version der PC-Station automatisch auf V2.x an.
  - Wenn Sie **nur** eine WinCC Runtime stecken, wechselt die Anzeige "PC-Station" auf "nicht installiert". Auf dem PC ist keine Installation der PC-Station erforderlich.
  - Wenn Sie eine WinCC Runtime stecken und es ist bereits ein Software Controller gesteckt, bleibt die Version der PC-Station unverändert.

Hinweis

Wenn Sie inkompatible Software-Produkte in das PC-System ziehen, dann wird Ihnen beim übersetzen der Station eine Fehlermeldung angezeigt.

Um welche Version der PC-Station es sich handelt, erkennen Sie in seinen Eigenschaften: Bei markierter PC-Station finden Sie die Einstellungen im Inspektorfenster im Bereich "PC-Station".

Verweis

Weitere Informationen zum Betrieb eines Software Controllers mit einer PC-Station finden Sie im Gerätehandbuch SIMATIC ET 200SP Open Controller CPU 1515SP PC (https://support.industry.siemens.com/cs/de/de/view/109248384), im Bedienhandbuch SIMATIC S7-1500 CPU 150xS (https://support.industry.siemens.com/cs/de/de/view/109249299) und im Projektierungshandbuch SIMATIC NET PC-Software PC-Stationen in Betrieb nehmen (https://support.industry.siemens.com/cs/de/de/view/109488960).

Software-Komponenten einer PC-Station

Eine PC-Station beinhaltet eine Reihe projektierbarer Software-Komponenten, die beim Laden in die PC-Station übernommen werden. Dabei gibt es verschiedene Anwendungen, die unterschiedliche Auswirkungen auf die Erzeugung und Hantierung der Projektierdaten haben.

- PC-Station V2.x:
  - WinCC RT Advanced, ab V13 SP1
  - S7-1500 Software Controller
- PC-Station V1.0:
  - Standard-Applikation
  - OPC-Server, als zentraler Kommunikationsdienstleister
  - WinCC RT Advanced
  - WinCC RT Professional
  - Win CC V7
  - WinAC RTX
  - Kommunikationsprozessoren
- Nicht installiert:
  - WinCC RT Advanced
  - WinCC RT Professional
  - WinCC Client

Entsprechend der genutzten Dienste projektieren Sie spezifische Eigenschaften dieser Komponenten.

WinCC RT Advanced

WinCC RT Advanced ist eine PC-basierte Bedien- und Beobachtungslösung für Einplatzsysteme im maschinennahen Bereich. Die WinCC RT Software ist bei den SIMATIC HMI-Bediengeräten (z. B. Basic/Comfort/Mobile Panel) an Bord und erlaubt den Datenaustausch über HMI-Verbindungen.

Wenn Sie mehrere HMI-Verbindungen zu einer CPU einrichten möchten, verwenden Sie z. B.:

- die PROFINET- und PROFIBUS DP-Schnittstellen der PC-Station
- CPs und CMs mit den entsprechenden Schnittstellen

Weitere Kommunikationsmöglichkeiten:

- OPC UA-Client oder als Server (alle HMI-Varianten)
- Kommunikation zwischen HMI-Systemen über das Intranet/Internet
  - Lesender und schreibender Zugriff auf Variablen

S7-1500 Software Controller

Der SIMATIC S7-1500 Software Controller ist eine PC-basierte Steuerung.

- Diese bietet dieselbe Funktionalität wie CPUs des Automatisierungssystems SIMATIC S7-1500.
- Software Controller bieten die Funktionalität einer speicherprogrammierbaren Steuerung (SPS) in einer PC-basierten Echtzeitumgebung.
- Um Ihr Anwenderprogramm zu entwickeln, nutzen Sie mit der PC-basierten Steuerung dieselben Programmiersprachen, Programmstruktur und Programmieroberfläche (STEP 7) wie für Hardware-Steuerungen.
- Für den SIMATIC S7-1500 Software Controller können Sie das gleiche Anwenderprogramm wie für eine Hardware-Steuerung nutzen. Die Software Controller CPU verfügt über eine Display-Applikation, die auf dem PC ausgeführt wird. Mit der Display-Applikation ermöglicht eine Anzeige des Betriebszustands der CPU. Sie können mit der Display-Applikation - analog zum Display einer Hardware-CPU - Diagnose- und Inbetriebnahme-Aufgaben durchführen.
- Der Software Controller nutzt die Schnittstellen des PC für PROFINET und PROFIBUS: Windows-unabhängige Nutzung von PROFINET oder PROFIBUS zum Betrieb von dezentraler Peripherie. Dabei sind je nach verwendeter Schnittstellenhardware folgende Funktionen möglich:
  - PROFINET IO RT oder PROFINET IO IRT
  - PROFIenergy
  - I-Device
  - PROFIBUS DP-Master
- Kommunikation (SIMATIC Kommunikation, Open User Communication, OPC UA) mit Windows Anwendungen oder externen Geräten über die Windows-Schnittstellen des PCs. Die Kommunikationsdienste richten sich nach eingesetzter Baugruppe.

Die Software Controller, basierend einer S7-1500, gibt es in den folgenden Ausführungen:

- Als SIMATIC ET 200SP Open Controller CPU 1515SP PC und als CPU 1515SP PC incl. HMI in der ET 200SP-Bauform, mit bereits installiertem und vorkonfiguriertem SIMATIC S7-1500 Software Controller. Zentral erweiterbar durch ET 200SP Peripheriemodule.
- Als Software Controller CPU 1507S und CPU 1508S für die PC-basierte Automatisierung. Der SIMATIC S7-1500 Software Controller macht die Vorteile des Standard Controllers SIMATIC S7-1500 auf hoch performanten Industrie-PCs verfügbar.

Standard-Applikation

Mit Standard-Applikationen kommunizieren Sie über Ihre PC-Baugruppe mit anderen Applikationen und Geräten.

Folgende Kommunikationsdienste können Sie je nach eingesetzter PC-Baugruppe projektieren:

- S7-Verbindungen
- DP-Dienste

OPC-Server als zentraler Kommunikationsdienstleister

Um mit einem Automatisierungsgerät (z. B. mit einer SIMATIC S7-1500) zu kommunizieren, verwenden die OPC-Clients die Schnittstelle zu einem OPC-Server.

Zur Anpassung des OPC-Servers projektieren Sie dessen Eigenschaften. Die eingestellten oder geänderten Parameter werden nach dem Laden der Projektierdaten in die PC-Station und dem Start des OPC-Servers wirksam.

Folgende Kommunikationsdienste können Sie je nach eingesetzter PC-Baugruppe nutzen:

- Alle Verbindungsarten
- DP-Dienste (DPV0)

WinAC RTX

Die SIMATIC WinAC (Windows Automation Center) ist eine PC-basierten Steuerungen und bieten dieselbe Funktionalität wie SIMATIC S7-300/400 CPUs (Hardware-Steuerungen). Die WinAC RTX wird genutzt zur Lösung von PC-typischen Applikationen (z. B. MS Office und selbst erstellte Anwenderprogramme) und Automatisierungsaufgaben in einem System.

Eine Ansteuerung von dezentrale Peripherie erfolgt über:

- PROFIBUS
  - Die Anbindung der Peripherie über PROFIBUS DP erfolgt über die integrierte DP-Schnittstelle der SIMATIC PCs oder über einen Kommunikationsprozessoren.
- PROFINET
  - Die Anbindung der Peripherie über PROFINET erfolgt über eine integrierte Ethernet-Schnittstelle des SIMATIC IPCs oder über einen Kommunikationsprozessor

Der optimale Zugriff auf die Daten für z. B. Bildverarbeitung, Messwerterfassung, etc. erfolgt über das WinAC-Option Open Development Kit (ODK)

Folgende Kommunikationsdienste können Sie je nach eingesetzter PC-Baugruppe nutzen:

- PG/OP-Kommunikation
- S7-Kommunikation
- Open User Communication (OUC)
- Prozessdatenzugriff über OPC

Kommunikationsprozessoren

Der Anschluss von SIMATIC Steuerungen über Kommunikationsprozessoren ermöglicht die direkte Integration der Steuerungen in industrielle Netzwerke.

Folgende Kommunikationsdienste können Sie je nach eingesetzter PC-Baugruppe nutzen:

- OPC UA-Client oder als Server
- Offene Kommunikation (TCP/IP und UDP): Multicast bei UDP
- PG/OP-Kommunikation: netzübergreifend durch S7-Routing
- S7-Kommunikation (Client, Server)

- IT-Kommunikation (HTTP(S), E-Mail)
- PROFINET IO-Controller mit Echtzeiteigenschaften (RT und IRT)
- IP-Adressvergabe über DCP
- Shared Device
- PROFIBUS DP
- PG/OP-Kommunikation

WinCC RT Professional

WinCC RT Professional ist ein PC-basiertes Bedien- und Beobachtungssystem für das Visualisieren und Bedienen von Prozessen, Fertigungsabläufen, Maschinen und Anlagen – von einfachen Einplatzsystem bis hin zu Mehrplatzsystemen und standortübergreifenden Lösungen mit Web-Clients.

Die WinCC RT Software ist bei den SIMATIC HMI-Bediengeräten (z. B. Basic/Comfort/Mobile Panel) an Bord und erlaubt den Datenaustausch über HMI-Verbindungen.

Wenn Sie mehrere HMI-Verbindungen zu einer CPU einrichten möchten, verwenden Sie z. B.:

- die PROFINET- und PROFIBUS DP-Schnittstellen der PC-Station
- CPs und CMs mit den entsprechenden Schnittstellen

Weitere Kommunikationsmöglichkeiten:

- OPC-Server
- OLE DB Server
- WinCC WebNavigator
- WinCC DataMonitor

WinCC Client

Je nach Anforderung kann ein WinCC Einplatzsystem zu einem leistungsfähigen Client-/Server-System ausgebaut werden. Auf diese Weise können mehrere koordinierte Bedien- und Beobachtungsstationen in einem gemeinsamen Verbund mit vernetzten Automatisierungssystemen betrieben werden.

Verweis

Weitere Informationen zu WinCC RT finden Sie in der Online-Hilfe zu WinCC.

Weitere Informationen zum S7-1500 Software Controller und zu WinAC RTX finden Sie im Kapitel Wissenswertes zu PC-Systemen (Seite 1510).

Weitere Informationen zur Bedeutung und Handhabung der PC-Applikationen und zum OPC-Server finden Sie im Projektierungshandbuch SIMATIC NET - Industrielle Kommunikation (https://support.industry.siemens.com/cs/de/de/view/61630799) und im Projektierungshandbuch SIMATIC NET - PC-Stationen in Betrieb nehmen (https://support.industry.siemens.com/cs/de/de/view/109488960).

Weitere Informationen zu den Komminikationsprozessoren finden Sie auf den Seiten von Service&Support (http://w3.siemens.com/mcms/industrial-communication/de/ie/systemanschluss/advanced-controller/Seiten/CPs-fuer-advanced-controller.aspx?tabcardname=CP%20für%20S7-400).

Ein Anwendungsbeispiel zum Wechsel auf die neue Software Controller Generation finden Sie im Leitfaden zur Migration PC-basierter Steuerungen: von SIMATIC WinAC RTX nach SIMATIC S7-1500 Software Controller und TIA Portal (https://support.industry.siemens.com/cs/document/109478804/leitfaden-zur-migration-pc-basierter-steuerungen%3A-von-simatic-winac-rtx-nach-simatic-s7-1500-software-controller-und-tia-portal?dti=0&lc=de-WW).

Verbesserungen von PC-Station V2.x gegenüber PC-Station V1.0

Im Folgenden sind die Gemeinsamkeiten und Verbesserungen zwischen beiden PC-Stationen stichpunktartig aufgeführt.

Gemeinsamkeiten der PC-Station V2.x und V1.0

PC-Station V2.x und PC-Station V1.0 haben folgende Gemeinsamkeiten:

- Einfache Installation aller benötigten Softwarekomponenten über ein Setup (z. B. S7-1500 Software Controller, WinCC Advanced RT, ...)
- Bestellbar sind auch PC-Systeme mit vorinstalliertem Software Controller bzw. mit vorinstalliertem Runtime Advanced.

Verbesserungen von PC-Station V2.2 gegenüber PC-Station V2.1

Im Vergleich zum PC-Station V2.1 bietet eine PC-Station V2.2 folgende Verbesserung:

- Laden von Projektdaten in eine Konfigurationsdatei

Verbesserungen von PC-Station V2.1 gegenüber PC-Station V2.0

Im Vergleich zum PC-Station V2.0 bietet eine PC-Station V2.1 folgende Verbesserung:

- Zugriff über einen Webserver möglich (Seite 1534)

Verbesserungen von PC-Station V2.0 gegenüber PC-Station V1.0

Im Vergleich zum PC-Station V1.0 bietet ein PC-Station V2.x folgende Verbesserungen:

- Laden der PC-Station-Konfiguration vergleichbar einfach wie bei einem modularen Controller S7-1500; keine vorherige Konfiguration des PC-Systemaufbaus am IPC nötig.
- Integrierte PC Hardware Diagnose eingebunden in die Systemdiagnose, z. B. um den Zustand von PC-Komponenten wie Lüfter oder Festplatte zu diagnostizieren.

Regeln für die Konfiguration

Regeln und Besonderheiten für die Konfiguration eines PC-Systems

Wegen der Kombinationsmöglichkeiten von Software- und Schnittstellenkarten mit verschiedenen IPC-Varianten existieren komplexe Abhängigkeiten.

Wir empfehlen, die PC-Station einfach nach Ihren Anforderungen zu konfigurieren und dann zu übersetzen. STEP 7 nimmt weitgehend notwendige Anpassungen vor, z. B. die Einstellung des Stationstyps. Wenn eine Kombination nicht übersetzbar ist, dann werden Sie durch eine Fehlermeldung auf die Verletzung einer Regel hingewiesen und erhalten Auskunft über Behebungsmöglichkeiten.

In den folgenden Abschnitten erhalten Sie Informationen zu den Grundregeln:

- Wenn ein IPC für einen S7-1500 Software Controller geeignet ist, dann können Sie auch eine WinAC RTX stecken. Gültig für die meisten IPCs.
- Folgende Komponenten sind für den S7-1500 Software Controller geeignet bzw. nicht geeignet:
  - Der Open Controller (CPU 1515SP PC) ist für den S7-1500 Software Controller geeignet.
  - Der IPC2x7D und der Embedded Controller (S7-mEC 31) sind nicht für den S7-1500 Software Controller geeignet.
- Gerätetausch zwischen S7-1500 Software Controller und WinAC RTX ist nicht möglich.
- Sie können jeweils nur einen Software Controller in ein PC-System stecken.
- Weder ein OPC-Server noch eine Applikation (Benutzeranwendung) sind zusammen mit einem S7-1500 Software Controller nutzbar. Eine installierte PC-Station V2.x verhindert die Installation des OPC-Servers bzw. der Applikation auf einem IPC.
  Beide Komponenten verwenden Sie für eine PC-Station V1.0.

- Zuweisen von Schnittstellen: Die Zuweisung einer Schnittstelle "Keine oder eine andere Windows-Einstellung" ist nur für Onboard-Schnittstellen sinnvoll. Den Schnittstellen von konfigurierten CPs/CMs sind zunächst der PC-Station zugewiesen, da diese in der Regel von den konfigurierten Komponenten der PC-Station genutzt werden.
- Die grafische Ansicht einer PC-Station in STEP 7 ist identisch in der Netzansicht und in der Gerätesicht.
  Die "Steckplätze", auf denen sich die Komponenten eines PC-Systems befinden, sind nicht nummeriert. Wenn Sie per Drag & Drop Komponenten aus dem Hardware-Katalog stecken, platziert STEP 7 diese automatisch: Links die Hardware Komponenten (z. B. CPs), rechts die Software-Komponenten (z. B. einen Software Controller).
  Detaillierte Informationen z. B. zum Index (das ist die Lokalisierungsinformation ähnlich einem Steckplatz im PC-System) finden Sie nur in der Geräteübersichtstabelle der Gerätesicht.
  - In der Geräteübersichtstabelle ist der Index 125 fest für das PC-Station vergeben (bei PC-Station V1.0 auch als "Stationmanager" bezeichnet). Dieser Index wird in der ersten Zeile der Geräteübersichtstabelle angezeigt.
  - Für den Open Controller CPU 1515SP PC sind zusätzlich zur Spalte "Index" die Spalten "Baugruppenträger", "Steckplatz" sowie "E-Adresse" und "A-Adresse" vorhanden. Diese Spalten enthalten die entsprechenden Informationen für gesteckte ET 200SP Module.
  - Gesteckte SW-Komponenten und CPs erhalten ihren Index in der Reihenfolge, in der diese in der Gerätesicht gesteckt werden. Einen bestimmten Index können Sie nur durch Drag & Drop vom Hardware Katalog auf die entsprechende Zeile in die Geräteübersichtstabelle erzwingen. Dies hat keine Auswirkungen auf die Darstellung in der grafischen Gerätesicht.
  - Onboard-Schnittstellen, die keiner Software Komponente zugewiesen sind, finden Sie ab Index 100. Sobald Sie eine Onboard-Schnittstelle zugewiesen haben, erhält diese den nächsten freien Index, z. B. 4. Auf diese Weise erscheinen zugewiesene Schnittstellen immer am Anfang der Geräteübersichtstabelle.

IPCs, CPs und Software Komponenten für PC-Station V2.x

Folgende IPCs sind geeignet für PC-Station V2.x:

- IPC227E, IPC277E
- IPC427D, IPC477D
- IPC427E, IPC477E
- ab IPC547G (ohne Software Controller CPU 1505SP,CPU 1507S oder CPU 1508S)
- IPC627D, IPC647D, IPC677D
- IPC827D, IPC847D (nicht mit fehlersicheren Varianten des Software Controllers)

Folgende CPs sind geeignet für PC-Station V2.x (nur als Schnittstelle des S7-1500 Software Controllers CPU 1507S verwendbar):

- IE allgemein für RPOFINET IO mit RT
- CP 1625 für PROFINET IO RT
- CP 1625 für PROFINET IO IRT

- CP5622 für PROFIBUS DP
- CP5623 für PROFIBUS DP

Folgende Software Komponenten sind geeignet für ein PC-Station V2.x:

- Software Controller CPU 1507S
- Software Controller CPU 1505SP
- Software Controller CPU 1507S
- Software Controller CPU 1508S
- WinCC RT Advanced ab V13 SP1

Vorgehen um ein PC-System zu konfigurieren

Einführung

Im Folgenden ist erläutert, wie Sie in der Netzsicht aus dem Hardware Katalog ein PC-System auswählen. Auf dieses PC-System ziehen Sie aus dem Hardware-Katalog die gewünschten Module, die automatisch angeordnet werden.

Auswahl der Komponente im Hardware-Katalog

Im Hardware Katalog wird jede Komponente als Ordner angezeigt. Wenn Sie diesen Ordner öffnen, sehen Sie die unterschiedlichen Versionen der ausgewählten Komponente mit ihren jeweiligen Artikelnummern.

Nachfolgend ist beispielhaft gezeigt, wie Sie in der Netzsicht ein PC-System anlegen.

Voraussetzung

- Der Hardware Katalog ist geöffnet.
- Sie befinden sich in der Netzsicht.

Vorgehen

Um ein PC-System mit einem gewünschten Software Controller zu konfigurieren, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware Katalog auf den Ordner "PC-Systeme".
2. Öffnen Sie den Ordner mit dem gesuchten Industrie-PC-Typ. Sie sehen alle Typen des angewählten IPC-Typs.
3. Selektieren Sie den ausgewählten IPC-Typ mit einem Mausklick.
4. Ziehen Sie mit Drag & Drop den IPC aus dem Hardware Katalog in die Netzsicht.
5. Wechseln Sie im Hardware Katalog zum Ordner "SIMATIC Controller Application".

6. Wählen Sie den gewünschten Software Controller:
   - CPU 1507S oder CPU 1508S
   - SIMATIC WinAC RTX
7. Ziehen Sie den gewünschten Software Controller mit Drag & Drop auf dieses PC-System.
8. Parametrieren Sie die Eigenschaften des PC-Systems und des Software Controllers.

Hinweis

Voraussetzungen für das Laden

Damit das Laden der Hardwarekonfiguration möglich ist, muss dem Software Controller oder der PC-Station eine Schnittstelle zugewiesen sein!

Wie Sie vorgehen, um Schnittstellen zuzuweisen, finden Sie in folgenden Kapiteln: Wissenswertes zum Schnittstellen zuweisen (Seite 1557) und Schnittstellen für die Kommunikation zuweisen (Seite 1559).

Installationsvoraussetzungen abhängig von konfigurierten Komponenten

Die konfigurierten Komponenten eines PC-Systems in STEP 7 bestimmen, welche Installationsvoraussetzungen für das Zielsystem gelten, z. B. für einen IPC. Diese Installationsvoraussetzungen müssen erfüllt sein, damit ein Laden der Konfiguration erfolgreich ausgeführt wird und damit Sie mit STEP 7 online auf das PC-System zugreifen können. Die jeweils geltenden (minimalen) Installationsvoraussetzungen zeigt STEP 7 bei markiertem PC-System im Inspektorfenster, Bereich "PC-Station", an.

Beispiel: Nur WinCC RT konfiguriert

Wenn nur WinCC RT konfiguriert ist, dann muss auf dem PC-System "nicht installiert" sein im Sinne von "keine PC-Station installiert".

Daher zeigt STEP 7 die Option "nicht installiert" an.

Beispiel: WinAC RTX konfiguriert

Wenn der Software Controller WinAC RTX konfiguriert ist, dann muss auf dem PC-System SIMATIC WinAC RTX 2010 (im Folgenden "PC-Station V1.0" genannt) installiert sein. Die SIMATIC WinAC RTX 2010 bringt als Installationskomponente die PC-Station bzw. S7-RTM mit, im Folgenden "PC-Station V1.0" genannt.

STEP 7 stellt automatisch diese Version der PC-Station ein, sobald eine WinAC RTX konfiguriert wurde.

Beispiel: CPU 1507S Software Controller konfiguriert

Wenn der Software Controller CPU 1507S konfiguriert ist, dann muss auf dem PC-System der SIMATIC S7-1500 Software Controller installiert sein. Dieser bringt die Installationskomponente PC-Station mit, im Folgenden "PC-Station V2.x" genannt.

STEP 7 stellt automatisch diese Version der PC-Station ein, sobald ein S7-1500 Software Controller konfiguriert wurde:

- PC-Station V2.4 (für Software Controller V2.6 ab TIA Portal V15 SP1)
- PC-Station V2.3 (für Software Controller V2.5 ab TIA Portal V15)
- PC-Station V2.2 (für Software Controller V2.1 ab TIA Portal V14 SP1)
- PC-Station V2.1 (für Software Controller V2.0 ab TIA Portal V14)
- PC-Station V2.0 (für Software Controller V1.x ab TIA Portal V13 SP1)

Verweis

Weitere Informationen zu den Installationsvoraussetzungen der jeweiligen PC-Station-Version finden Sie im Kapitel Laden mit STEP 7 (Seite 1523).

Besonderheiten der Diagnose beim PC-System V2.x

Besonderheiten der Diagnose beim PC-System V2.x

Das Diagnosekonzept für ein PC-System V2.x ist hierarchischen aufgebaut wie bei einer modularen CPU. Sie können auf einen Blick erkennen, ob die Station fehlerfrei ist oder ob z. B. eine unterlagerte Komponente einen Fehler hat.

Einziger Unterschied: Sie haben eine Hierarchiestufe mehr als bei modularen CPUs, das PC-System ist dem Software Controller übergeordnet. Das PC-System hat einen eigenen Hardware Status (eigene Diagnose).

Das folgende Bild zeigt die Diagnose des kompletten PC-Systems in der Projektnavigation. Die PC-Station selbst ist in Ordnung (grüner Haken), hat aber unterlagert eine Komponente mit Fehler (Ausrufezeichen im grünen Haken). Die Symbole geben eine erste Auskunft über gestörte oder fehlende Module in der unterlagerten Struktur.

Um eine genauere Fehlerbeschreibung anzuzeigen, doppelklicken Sie in der Projektnavigation auf den Eintrag "Online & Diagnose".

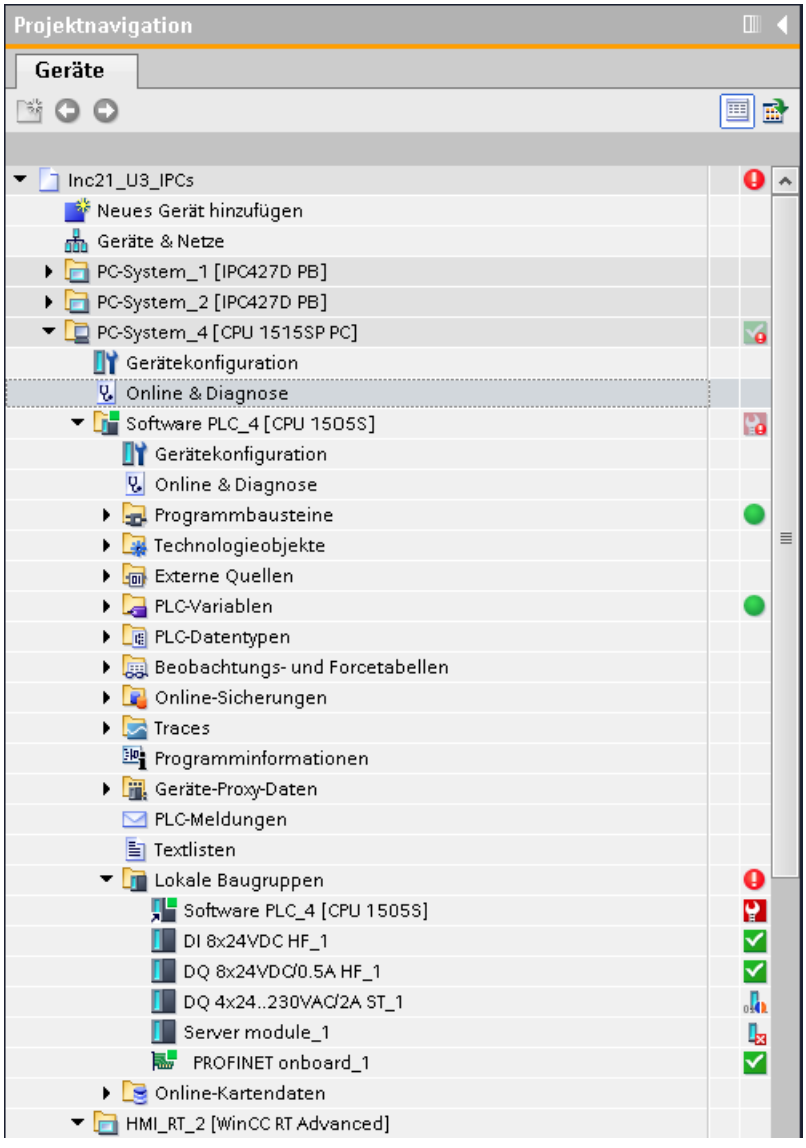

Bild 1-34 Anzeige der Störungen in der Projektnavigation

Das folgende Bild zeigt die beiden Einträge in der Projektnavigation zum Start der Online- und Diagnosesicht des PC-Systems. Einzelheiten entnehmen Sie der Bildlegende.

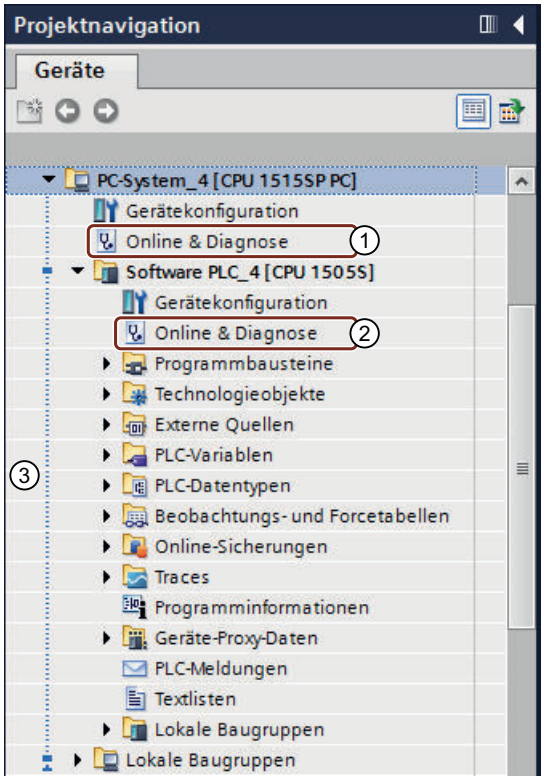

① Hardware Diagnose des PC-Systems (bei IPCs z. B. Lüfter, Temperaturen, Spannungen)

② Standard Diagnose (wie bei S7-1500 CPU)

③ Optische Klammer für die Komponenten der PC-Station

Bild 1-35 Projektnavigation "Online & Diagnose"

Weitere Informationen

Weitere Informationen zum Thema "Diagnose" und die Legende mit der Bedeutung der einzelnen Symbole finden Sie im Funktionshandbuch Diagnose (http://support.automation.siemens.com/WW/view/de/59192926).

Laden mit STEP 7

PC-Station V2.x mit STEP 7 laden

Im Folgenden sind die Besonderheiten für das Laden eines PC-System V2.x mit STEP 7 aufgeführt.

Voraussetzung

- Auf dem PC-System sind die entsprechenden Software-Komponenten installiert (z. B. SIMATIC S7-1500 CPU 1507S).
- Wenn erforderlich, installieren Sie auf dem PC-System auch die entsprechende Version der Setup-DVD zur SIMATIC WinCC Runtime Advanced.

- Die Hardware Komponente SIMATIC IPC ist physikalisch über Ethernet mit dem PC verbunden, auf dem STEP 7 installiert ist.
- Der Software Controller ist gestartet.

Weitere Einstellungen an der PC-Station sind nicht erforderlich, da durch das Laden des Projekts das PC-System V2.x komplett konfiguriert wird.

Hinweis

Empfohlene Schnittstellen zum Laden

- Nutzen Sie zum Laden eines SIMATIC IPC eine "PN/IE"-Schnittstelle, die nicht dem Software Controller als PROFINET-Schnittstelle zugeordnet werden soll.
  Empfohlen wird die "X1"-Schnittstelle.
- Nutzen Sie zum Laden einer CPU 1515SP PC die "X2"-Schnittstelle.

Vorgehen

Um das STEP 7 Projekt zu laden, gehen Sie folgendermaßen vor:

1. Markieren Sie das PC-System in der Gerätesicht.
2. Wählen Sie den Kontextmenübefehl "Laden in Gerät".
   Der Dialog "Erweitertes Laden" wird geöffnet.
3. Konfigurieren Sie die Einstellungen zur Schnittstelle.
4. Klicken Sie auf die Schaltfläche "Laden", um den Ladevorgang zu starten.
   Wenn das PC-System im Dialog "Erweitertes Laden" nicht ausgeführt wird, dann kann das an einer fehlenden Schnittstellen-Zuweisung liegen.

Ergebnis

Das Projekt wird stufenweise geladen. Ein Dialog zeigt den Ladefortschritt an.

- Wenn Sie Änderungen an der PC-Systemkonfiguration vorgenommen haben (z. B. Schnittstellen-Zuweisung, LED, NVRAM oder Index), dann wird zunächst die PC-Systemkonfiguration geladen. Wenn danach ein Neustart erforderlich ist, zeigt STEP 7 eine entsprechende Meldung an. Sie können an dieser Stelle das Laden beenden, um z. B. zunächst den Software Controller zu parametrieren und später zu laden. Bereits geladene Konfigurationen werden nur dann erneut geladen, wenn sie sich geändert haben.
- Wenn Sie nach dem Laden der PC-Systemkonfiguration das Laden nicht beenden, sondern warten, startet das PC-System neu und lädt automatisch die übrigen Komponenten wie die Konfiguration des Software Controllers und WinCC Advanced.

Ergebnis des Ladevorgangs überprüfen

Nach erfolgreicher Beendigung des Ladevorgangs erscheint die CPU-Verknüpfung im Startmenü von Windows mit dem Namen, den Sie in den Einstellungen der CPU in STEP 7 vergeben haben.

Auch im Display der CPU ist der in STEP 7 vergebene Name sichtbar.

PC-Station V1.0 mit STEP 7 laden

Im Folgenden sind die Besonderheiten für das Laden eines PC-Systems V1.0 mit STEP 7 aufgeführt.

Voraussetzung

- Auf dem PC-System ist die DVD SIMATIC WinAC RTX installiert.
- Jedes zu ladende Gerät ist über einen Online-Zugang erreichbar.
- Die PC-Station (Runtime-PC) ist über ein Netzwerk mit der Projektierungsstation gekoppelt. Die PC-Station ist z. B. durch das Setzen der Parameter in der Erstkonfiguration als Kommunikationsteilnehmer erreichbar.
- Eine bestehende Verbindung wird z. B. durch die Funktion "Erreichbare Teilnehmer" geprüft.
- Die Ziel-PC-Station muss mit dem Komponenten-Konfigurator konfiguriert sein.

Projektierung übertragen

Um eine erstellte Projektierung auf die PC-Station zu übertragen, gibt es folgende Möglichkeiten:

- Online-Betrieb
  - vernetzt (Engineering Station mit Runtime-PC vernetzt); Zugangspunkt: "S7ONLINE (STEP 7)"
  - lokal (Projektierstation mit PC-Station identisch); Zugangspunkt: "PC-internal (lokal)"
- Offline-Betrieb (Engineering Station und Runtime-PC getrennt) - XDB-Datei und WinAC-Programmdatei (*.wld) erzeugen und importieren

Online-Betrieb

Der Online-Betrieb ermöglicht das direkte Laden der Projektierungsdaten:

- Laden über an ein Netzwerk (MPI, PROFIBUS oder Ethernet) angeschlossene PC-Station
- Laden auf die lokale PC-Station, die gleichzeitig als Projektierungsstation genutzt wird

Vorgehen: Online-Betrieb - vernetzt

1. Markieren Sie die zu ladende Station im STEP 7.
2. Laden Sie die Projektierdaten über "Online" > "Laden in Gerät".

Hinweis

Für PG-Betrieb

Achten Sie darauf, dass Sie über "PG/PC-Schnittstelle einstellen" die korrekte Schnittstelle einstellen (Zugangspunkt "S7ONLINE (STEP 7)").

Vorgehen: Online-Betrieb - lokal

Die Konfigurationsdaten werden in diesem Fall direkt über eine PC-interne Verbindung übertragen.

1. Markieren Sie die zu ladende Station im STEP 7.
2. Laden Sie die Projektierdaten über "Online" > "Laden in Gerät".

Offline-Betrieb

In dieser Betriebsart wird für die Datenübertragung eine Datei vom Typ XDB benötigt. Das Projektierungssystem generiert eine XDB-Datei für jede projektierte PC-Station.

Den Ablageort der Konfigurationsdatei entnehmen Sie den Eigenschaften des Objekts "PC-Station" im Projektierungssystem. Auf der PC-Station kann diese Datei mit dem "Komponenten-Konfigurator" importiert werden.

Weitere Informationen zum Offline-Betrieb finden Sie im Kapitel "PC-Station V1.0 über Datenträger laden (Seite 1526)".

Verweis

Weitere Informationen finden Sie im Projektierungshandbuch "NET PC-Software PC-Stationen in Betrieb nehmen - Anleitung und Schnelleinstieg (https://support.industry.siemens.com/cs/de/de/view/109488960)".

Laden über Datei

PC-Station V1.0 über Datei laden

Im Folgenden sind die Besonderheiten für das Laden einer PC-Station V1.0 über Datenträger aufgeführt.

Um die komplette Projektierung zu laden, haben Sie zwei Möglichkeiten:

- Möglichkeit 1: Konfiguration über eine XDB-Datei erzeugen und importieren (Seite 1526)
- Möglichkeit 2: Programm mit Hilfe der Memory Card erstellen und in den Software Controller laden (Seite 1528)

Konfiguration über eine XDB-Datei erzeugen und importieren

Hinweis

Es darf kein WinAC RTX installiert sein.

Vorgehen

Um eine PC-Station offline über eine XDB-Datei zu erzeugen und anschließend zu importieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus dem Hardware-Katalog unter "PC-Systems" ein Gerät für eine PC-Station in die Netzsicht.
2. Selektieren Sie die PC-Station.
3. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein" den Namen der PC-Station nach Ihren Bedürfnissen.
4. Ziehen Sie aus dem Hardware-Katalog benötigte weitere Komponenten mit Drag & Drop in die PC-Station. Eine eingefügte Komponente wird auf einen neuen Index gesteckt.

   Hinweis

   Sie können den Index einer selektierten Komponente im Inspektorfenster unter "Eigenschaften > Allgemein > Positionsnummer" ändern. Dadurch können Lücken in der Indexnummerierung entstehen. Diese Lücken werden in der grafischen Darstellung der Gerätesicht nicht angezeigt. Eine Anzeige der Lücken sehen Sie aber im tabellarischen Bereich der Gerätesicht.

5. Selektieren Sie die PC-Station.
6. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > XDB-Konfiguration" das Optionskästchen "XDB-Datei generieren".
   In der XDB-Datei werden die Verbindungsdaten und Adressen für CPs und Applikationen gespeichert.
7. Legen Sie den Ablagepfad unter "Pfad der XDB-Datei" fest.
8. Selektieren Sie die PC-Station.
9. Wählen Sie im Kontextmenü den Befehl "Übersetzen > ...".
   Beim Speichern und Übersetzen der Konfiguration einer PC-Station werden Systemdaten und die XDB-Konfigurationsdatei erzeugt. Diese können Sie anschließend in das Zielsystem laden bzw. diese muss installiert werden.

   Hinweis

   Der Ort der Konfigurationsdatei wird auf der Ziel-PC-Station über das Programm "PG-PC-Schnittstelle einstellen" eingestellt (im Register "STEP 7 Projektierung").

10. Kopieren Sie die XDB-Datei auf einen Datenträger, auf den die Zielstation Zugriff hat.
11. Importieren Sie die XDB-Datei über den Komponenten-Konfigurator auf Ihrer Ziel PC-Station.

Programm mit Hilfe der Memory Card erstellen und in den Software Controller laden

Vorgehen

Die Konfigurationsdaten, die aktuellen Werte der DBs und das STEP 7-Anwenderprogramm können in wld-Dateien archiviert, wiederverwendet und weitergegeben werden. Die Archivdatei (*.wld) speichert nicht die Konfiguration der PC-Station im Komponenten-Konfigurator.

Voraussetzung hierfür ist, dass sich der Controller im Betriebszustand STOP befindet.

Um eine Archivdatei anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt > Memory Card-Datei" den Befehl "Neu...".
   Ein Dialog wird geöffnet.
2. Wählen Sie das gewünschte Verzeichnis aus, auf dem Sie die Datei erstellen möchten.
3. Geben Sie den Dateinamen und den Pfad an.
4. Bestätigen Sie mit "Erstellen".

   Hinweis

   Sie können eine Archivdatei auch in der WinAC über den Menübefehl "Datei > Archivieren von CPU" erstellen.

5. Wählen Sie im Menü "Projekt > Memory Card-Datei" den Befehl "Öffnen...", um die erstellte Archivdatei in der Projektnavigation anzuzeigen.
6. Wählen Sie das Verzeichnis aus, auf dem die Archivdatei mit der Dateiendung (*.wld) liegt.
7. Die Archivdatei erscheint in der Projektnavigation unter "Card Reader/USB-Speicher".
8. Kopieren Sie die wld-Datei auf einen Datenträger, auf den die Zielstation Zugriff hat.
9. Um die Archivdatei wiederherzustellen, wählen Sie den Menübefehl "Datei > Laden auf CPU".
10. Wählen Sie die Archivdatei mit der Erweiterung *.wld, die Sie wiederherstellen möchten.
11. Bestätigen Sie mit "OK". Das STEP 7-Anwenderprogramm und die Konfiguration für den Software Controller werden neu geladen.

PC-Station V2.x über Datei laden

Die Möglichkeit die Anlagenkonfiguration des PC-Systems in einer Konfigurationsdatei zu speichern und zu transportieren, bietet folgende Vorteile:

- aktualisieren von großen Anlagen, ohne TIA Portal
- einfache Bereitstellung von Programm- und Konfigurations-Updates
- kein anlagennahes Update mehr erforderlich
- keine spezielle Software erforderlich

Konfigurationsdatei erstellen

Konfigurationsdatei erstellen

Aus dem TIA Portal wird die gesamte Projektierung Ihrer PC-Station in einer Konfigurationsdatei gespeichert. Die Daten können wiederverwendet und weitergegeben werden. Die Konfigurationsdatei hat die Endung *.psc.

Um eine Konfigurationsdatei anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt > PC-Station Konfigurationsdatei" den Befehl "Neu > PC-Station Konfigurationsdatei".
2. Geben Sie im geöffneten Dialog "Memory Card Datei erstellen" den Dateinamen an. Um Fehlermeldungen zu vermeiden, achten Sie auf richtige Eingaben:
   - Kurze, eindeutige Namen verwenden
   - Name darf maximal 255 Zeichen haben
   - Name darf kein Leerzeichen enthalten
   - Nur erlaubte Zeichen verwenden; erlaubten Zeichen sind Buchstaben und Ziffern, sowie die Sonderzeichen '-' und '_'.
3. Wählen Sie das gewünschte Verzeichnis aus, in dem Sie die Datei erstellen möchten. Um Fehlermeldungen zu vermeiden, achten Sie auch auf richtige Eingaben, wie unter Punkt 2.
4. Bestätigen Sie mit "Erstellen".

ACHTUNG
Sichern von Daten gegen Zugriff von Dritten Die Verantwortung für den sicheren Transport von Daten liegt vollständig beim Kunden.

Ergebnis

In der Projektnavigation unter "Card Reader/USB-Memory" wird ein Ordner "Memory Card-Datei" erstellt mit folgender Struktur:

- PC-System-Konfigurationsdatei
  Diese Datei enthält die PC-System-Konfigurationsdatei. Die Informationen zeigen den Dateinamen und die Pfadinformationen an, z. B.: LW:\PC-SystemConfiguration01.psc
  - Icon "PC-Systeminfo anzeigen"
    Mit einem Doppelklick auf das Icon werden in einem Editor alle projekt-, geräte- und modulrelevanten Informationen über die geladene Konfiguration angezeigt. Wenn weitere Daten geladen werden, können Sie sich über die Schaltfläche "Aktualisieren" die neuesten Metadaten anzeigen lassen.
  - Ordner mit dem bereits in der Projektnavigation vergebenem Stationsnamen, z. B. PC-System_1.
    Dieser Ordner enthält die Konfiguration des PC-Systems.

Projektdaten in Konfigurationsdatei laden

Um Daten in die PC-System-Konfigurationsdatei zu laden, haben Sie die folgenden Möglichkeiten:

- Projektdaten per drag&drop oder copy&paste auf eine Memory Card laden
- Projektdaten auf eine Memory Card schreiben

Voraussetzung

- In dem STEP 7-Projekt ist ein PC-System mit PC-Station V2.2 oder höher konfiguriert. Für Failsafe ist ein PC-System mit PC-Station V2.3 oder höher konfiguriert.
- In der Projektnavigation ist eine *.psc Datei angelegt und geöffnet.
- Verwenden Sie in Ihrer Projektierung eine CPU1515SP PC2, Microbox, Nanobox "E", IPC547G oder einen IPC xxxD, dann ist das nur in Verbindung mit einer PC-Station Version ≥ V2.2 möglich.

Hinweis

Besonderheit

Das Kopieren einer einzelnen Software-CPU in die *psc-Datei ist nicht möglich.

Beachten Sie auch die Hinweise zum Konfigurations-Datei Import aus dem Failsafe-Handbuch "Safety Projektieren und Programmieren (https://support.industry.siemens.com/cs/ww/de/view/54110126)".

Ein Anwendungsbeispiel zum Thema "Projektdaten in Konfigurationsdatei laden" finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/109759142).

Projektdaten auf eine Memory Card-Datei laden

Um Projektdaten auf eine Memory Card-Datei zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation die Projektdaten, die Sie laden möchten, auf die Memory Card.
   Falls notwendig, werden die Projektdaten übersetzt.
2. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion". Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
4. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

oder:

1. Selektieren Sie in der Projektnavigation den Ordner "PC-System".
2. Klicken Sie mit der rechten Maustaste auf die Selektion und wählen Sie im Kontextmenü den Befehl "Kopieren". Alternativ können Sie auch die Tastenkombination <Strg+C> verwenden.

3. Klicken Sie in die Memory Card-Datei mit der rechten Maustaste auf die Dateiebene "*.psc" oder den Ordner "PC-System und wählen Sie im Kontextmenü den Befehl "Einfügen". Alternativ können Sie auch die Tastenkombination <Strg+V> verwenden.
   Alle anderen Ebenen sind gesperrt. Falls notwendig, werden die Projektdaten übersetzt.
4. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion". Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

oder:

1. Selektieren Sie in der Projektnavigation den Ordner "PC-System".
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben".
   Der Dialog "Memory Card auswählen" wir geöffnet.
3. Wählen Sie eine Memory Card aus.
   Im unteren Bereich des Dialogs wird eine Schaltfläche mit einem grünen Hacken aktiv.
4. Klicken Sie auf die Schaltfläche mit dem grünen Hacken.
   Falls notwendig, werden die Projektdaten übersetzt.
5. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
6. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion". Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
7. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Ergebnis

Die psc-Datei enthält die Konfiguration für alle Komponenten in entsprechenden Unterordnern. Der Name des Ordners wird in den aktuellen PC-Systemnamen geändert.

Hinweis

Prüfen der Datei auf Vollständigkeit

Prüfen Sie die psc-Datei in Ihrem TIA Portal auf Vollständigkeit, denn außerhalb des TIA Portals ist die Datei nicht mehr bearbeitbar.

Existierende Konfigurationsdateien öffnen

Konfigurationsdatei öffnen

Um eine Konfigurationsdatei in der Projektnavigation anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "PC-Station Konfigurationsdatei > Öffnen > PC-Station Konfigurationsdatei.
2. Wählen Sie das Verzeichnis aus, auf dem die psc-Datei liegt.

In der Projektnavigation unter "Card Reader/USB-Memory" erscheint die Memory Card-Datei mit den genannten Inhalten.

Konfigurationsdatei auf Zielsystem importieren

Im Folgenden sind die Werkzeuge, mit denen Sie das Zielsystem ohne TIA Portal in Betrieb nehmen, aufgeführt:

- Kommandozeilen-Werkzeug
- PC-Stationsanzeige im Infobereich

Voraussetzungen

- Sie haben Administrator-Rechte.
- Ein Software Controller mit der Version 2.5 oder höher ist installiert.
- Der Zugriff auf den Open Controller CPU 1515 SP PC2 und/oder einem IPC ist möglich.
- Die psc-Datei liegt in einem lokalen Verzeichnis oder ist auf einem Medium, z. B. einem USB-Stick vorhanden.
- Für Failsafe: Sie sind in der Windowsbenutzergruppe "Failsafe Operators" aufgenommen.

Überblick der unterstützten Kommandozeilen-Befehle

Die folgende Tabelle gibt einen Überblick über die unterstützten Kommandozeilen-Befehle:

Befehl	Erläuterung
PCSystem_Control.exe /Help	Zeigt den Hilfetext im Kommandozeilen-Editor an.
PCSystem_Control.exe /HelpExitCode	Zeigt den Hilfetext zu den Errorcodes im Kommandozeilen-Editor an.
PCSystem_Control.exe /PrintConfig <file.psc*>	Zeigt die Informationen zu den Komponenten an; Ausgabeformat: Standard
PCSystem_Control.exe /PrintConfig <file.psc*> /xml	Zeigt die Informationen zu den Komponenten an; Ausgabeformat: XML-Format
PCSystem_Control.exe /ImportConfig <file.psc*>	Der Aufruf startet den vollständigen Import der psc-Datei.

Befehl	Erläuterung
PCSystem_Control.exe /GetStatus /ImportConfig	Zeigt den aktuellen Status, aller Komponenten, des Importprozesses an; Ausgabeformat: XML-Format
PCSystem_Control.exe /GetStatus /SimaticComponents	Zeigt die Informationen über alle installierten Simatic-Komponenten an; Ausgabeformat: XML-Format

* Der Parameter <file.psc> steht für den kompletten Pfad und den Dateinamen, z. B. C:\Data\Test.psc

Ergebnis

Der Import der Daten läuft im Hintergrund.

Ist ein Neustart projektiert, erfolgt dieser automatisch nach der Beendigung des Imports. Ist in der importierten Projektierung eine Software-CPU enthalten, bleibt diese nach dem Neustart im STOP.

Vorgehensweise bei Fehlermeldungen

Tritt bei einem Kommando-Befehl ein Fehler auf, haben Sie die Möglichkeit die Bit-Nummer des Fehlers zu ermitteln und eine genaue Fehlerbeschreibung zu erhalten.

Gehen Sie folgendermaßen vor:

1. Um den Fehlercode zu erhalten geben Sie im Kommandozeilen-Werkzeug das Kommando "echo %errorlevel%" ein.
2. Über dieses Kommando erhalten Sie folgendes Ergebnis:
   - Ist der Fehlercode = 0, liegt kein Fehler vor.
   - Ist der Fehlercode > 0, liegt ein Fehler vor und wird als Dezimalzahl ausgegeben.
   - Für Failsafe: Fehlercode 20899 bzw. 0x51A3, Import war erfolgreich.
3. Die Dezimalzahl muss in eine Binärzahl umgerechnet werden.
4. Um eine Übersicht der Bit-Nummer mit Fehlerbeschreibung zu erhalten, geben Sie im Kommandozeilen-Werkzeug das Kommando „HelpExitCode“ ein.

Beispiel

Beispiel für Umrechnung von Zahlensystemen:

- Dezimalzahl: 288
- Binärzahl: 100100000

Bit-Nummer mit der zugehörigen Fehlerbeschreibung:

Ergebnis: Den Fehler können Sie an Bit-Nummer 5 "Err_Net_45_Full=1" und Bit-Nummer 8 "Err_IIS_Running=1" ablesen.

Webserver

Einstellungen des Webserver im Zielsystem

Die Funktionalität des Webserver ist ab PC-Station V2.1 verfügbar. Im Folgenden sind die Einstellungen und Besonderheiten für die Nutzung des Webservers aufgeführt.

Voraussetzung

Folgende Software-Voraussetzungen sind notwendig, um den vollen Funktionsumfang des Webserver zu nutzen:

- PC-Station ≥ V2.3:
  - Windows 10 IoT Enterprise (64Bit) ist installiert
- PC-Station ≤ V2.2:
  - Windows 7 Embedded ist installiert
  - Windows Ultimate 7 incl. dem Internetinformationsdienstes (IIS) ≥ V7.0 ist installiert (siehe Kapitel Installieren des Internetinformationsdienstes (Seite 1535))
- .NET-Version ≥ 4.5 ist installiert

Hinweis

Die Software-Installationen sind VOR der Installation der PC-Station durchzuführen.

Prinzip

Der Webserver ist eine Web-Applikation und Bestandteil der Microsoft Internet Information Server-Installation. Während der Installation integriert sich der Webserver in die Standard-Website unter dem vorinstallierten IIS, mit Zugang über den Port 80.
Der Webserver kommt zum Einsatz, wenn die URL-Adresse: /simatic oder /simatic/ enthält.

Beispiel

Beispiele, wie Sie den Webserver erreichen:

- http://IP-Adresse/simatic/ oder http://IP-Adresse/simatic
- http://computername/simatic/ oder http://computername/simatic

Verweis

Weitere Informationen zum Webserver finden Sie im Funktionshandbuch Webserver (https://support.industry.siemens.com/cs/de/de/view/59193560).

Aktivieren des Internetinformationsdienstes für Windows 7 und Windows10

Aktivieren des Internetinformationsdienstes (IIS) für Windows 7

Im Folgenden zeigen wir Ihnen, wie Sie den Internetinformationsdienst (IIS) aktivieren und welche Einstellungen dazu notwendig sind.

Hinweis

Die zusätzliche Installation des IIS ist nur notwendig, wenn auf Ihrem Rechner Windows 7 Ultimate installiert ist.

Voraussetzung

Windows 7 Ultimate ist installiert.

Vorgehen

Um den Internetinformationsdienst (IIS) zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Menü-Symbol "Start".
2. Wählen Sie "Systemsteuerung > Programme und Funktionen > Windows-Funktionen aktivieren oder deaktivieren".
3. Bestätigen Sie die Meldung zur Benutzerkontosteuerung mit "Ja".
4. Navigieren Sie zu dem Menüpunkt: "Internetinformationsdienst" und nehmen folgende Einstellungen vor:

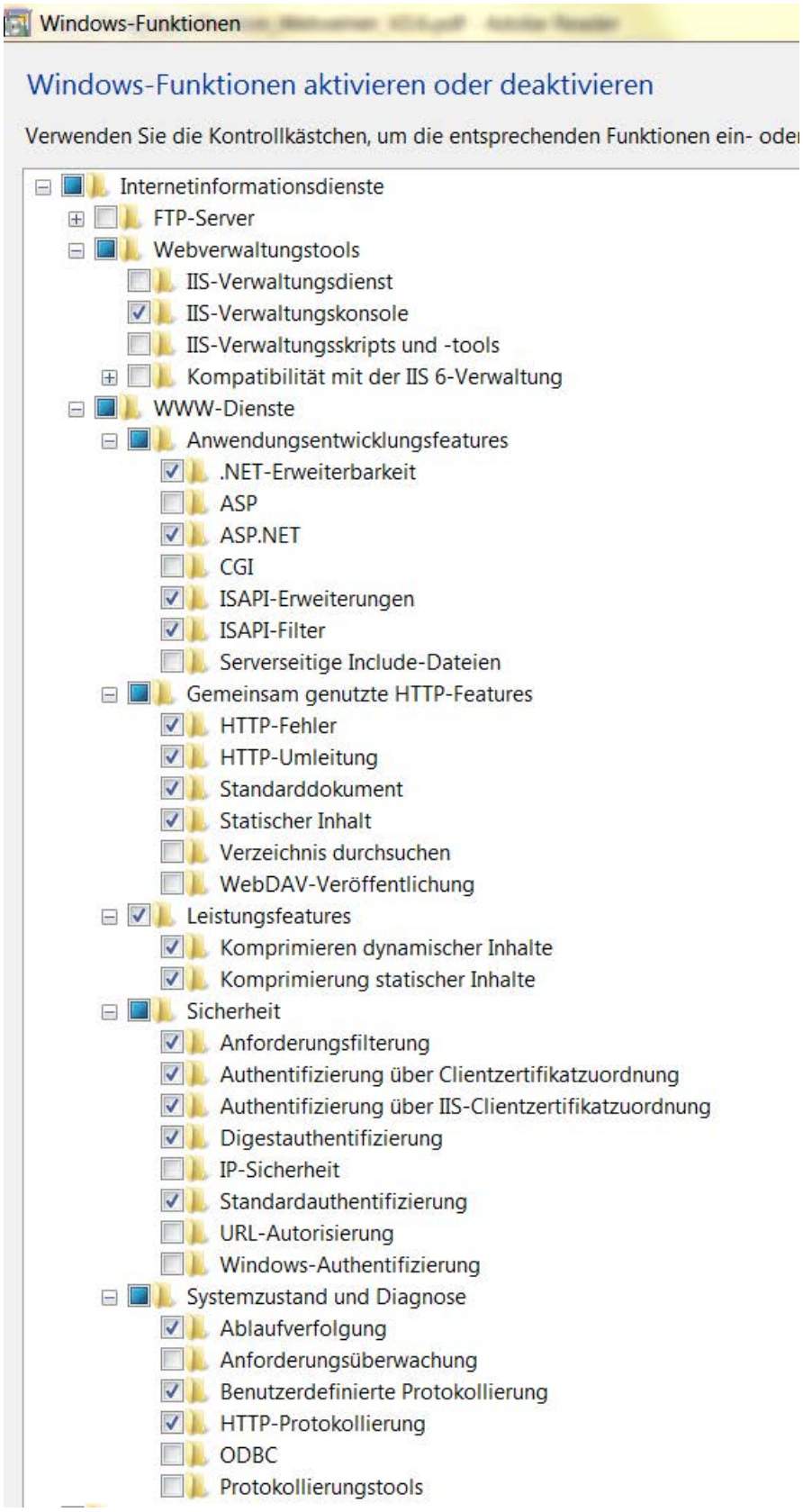
Windows-Funktionen
Windows-Funktionen aktivieren oder deaktivieren
Verwenden Sie die Kontrollkästchen, um die entsprechenden Funktionen ein- oder
Internetinformationsdienste
FTP-Server
Webverwaltungstools
IIS-Verwaltungsdienst
IIS-Verwaltungskonsole
IIS-Verwaltungsskripts und -tools
Kompatibilität mit der IIS 6-Verwaltung
WWW-Dienste
Anwendungsentwicklungsfeatures
.NET-Erweiterbarkeit
ASP
ASP.NET
CGI
ISAPI-Erweiterungen
ISAPI-Filter
Serverseitige Include-Dateien
Gemeinsam genutzte HTTP-Features
HTTP-Fehler
HTTP-Umleitung
Standarddokument
Statischer Inhalt
Verzeichnis durchsuchen
WebDAV-Veröffentlichung
Leistungsfeatures
Komprimieren dynamischer Inhalte
Komprimierung statischer Inhalte
Sicherheit
Anforderungsfilterung
Authentifizierung über Clientzertifikatzuordnung
Authentifizierung über IIS-Clientzertifikatzuordnung
Digestauthentifizierung
IP-Sicherheit
Standardauthentifizierung
URL-Autorisierung
Windows-Authentifizierung
Systemzustand und Diagnose
Ablaufverfolgung
Anforderungsüberwachung
Benutzerdefinierte Protokollierung
HTTP-Protokollierung
ODBC
Protokollierungstools

Bild 1-36 Einstellungen für den Internetinformationsdienst

Probleme beim Zugriff auf den Webserver

Mögliche Ursachen und deren Abhilfemaßnahme finden Sie hier:

- das Aktivieren des Internetinformationsdienstes erfolgte nach der Installation von .NET
- auf dem Computer wird ein 64-Bit-Version von Windows ausgeführt

Um das Problem zu beheben, ist eine nachträgliche Registrierung einer .NET-Version im Internetinformationsdienst (IIS) notwendig. Führen Sie dazu im CMD-Fenster von Windows folgenden Befehl aus: C:\Windows\Microsoft.NET \Framework64\v4.0.30319>aspnet_regiis.exe –ir

Aktivieren des Internetinformationsdienstes (IIS) für Windows 10

Im Folgenden zeigen wir Ihnen, wie Sie den Internetinformationsdienst (IIS) aktivieren und welche Einstellungen dazu notwendig sind.

Hinweis

Die zusätzliche Installation des IIS ist nur notwendig, wenn auf Ihrem Rechner Windows 10 IoT Enterprise (64Bit) installiert ist.

Voraussetzung

Windows 10 IoT Enterprise (64Bit) ist installiert.

Vorgehen

Um den Internetinformationsdienst (IIS) zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Menü-Symbol "Start".
2. Wählen Sie "Programme und Funktionen" > "Windows-Features aktivieren oder deaktivieren".

3. Bestätigen Sie die Meldung zur Benutzerkontosteuerung mit "Ja".
4. Navigieren Sie zu dem Menüpunkt: "Internetinformationsdienst" und nehmen folgende Einstellungen vor:

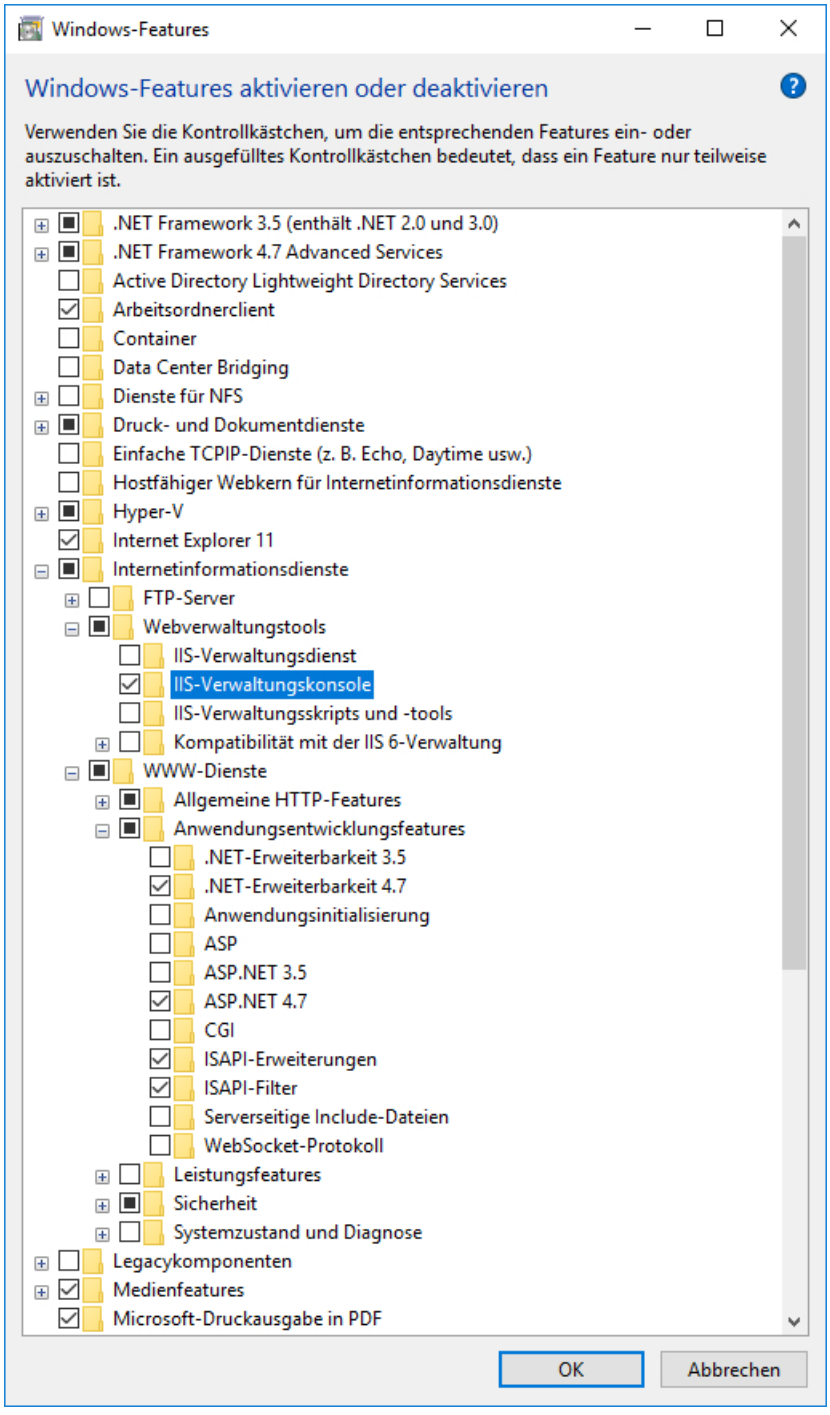

Probleme beim Zugriff auf den Webserver

Mögliche Ursachen und deren Abhilfemaßnahme finden Sie hier:

- das Aktivieren des Internetinformationsdienstes erfolgte nach der Installation von .NET
- auf dem Computer wird ein 64-Bit-Version von Windows ausgeführt

Um das Problem zu beheben, ist eine nachträgliche Registrierung einer .NET-Version im Internetinformationsdienst (IIS) notwendig. Führen Sie dazu im CMD-Fenster von Windows folgenden Befehl aus: C:\Windows\Microsoft.NET \Framework64\v4.0.30319>aspnet_regiis.exe –ir

Konfiguration des Webservers im TIA Portal

Im Folgenden ist das Vorgehen für die Konfiguration des Webservers im TIA Portal beschreiben.

Vorgehen

Um den Webserver nutzen zu können, gehen Sie folgendermaßen vor:

1. Selektieren Sie das gewünschte PC-System.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" folgende Einstellungen aus:
   - Webserver auf dieser Baugruppe aktivieren
   - Automatische Aktualisierung aktivieren
   - Benutzer anlegen und verwalten

Verweis

Weitere Informationen zum Webserver finden Sie im Funktionshandbuch Webserver (https://support.industry.siemens.com/cs/de/de/view/59193560).

Webserver über Kommandozeilen-Befehle einrichten

Bei der Installation des PC-Systems wird folgendes Kommandozeilen-Werkzeug mitinstalliert: PCSystem_Control.exe.

Das Tool ermöglicht Ihnen Informationen über den Webserver abzufragen und folgende Aufgaben zu erledigen:

- Webserver-Registrierung abfragen und überprüfen
- Webserver registrieren
- Webserver-Registrierung rückgängig machen
- Webserver-Einstellungen nachträglich ändern

Voraussetzung

Der Nutzer hat Administrator-Rechte.

Überblick der unterstützten Kommandozeilen-Befehle

Die folgende Tabelle gibt einen Überblick über die unterstützten Kommandozeilen-Befehle:

Befehl	Erläuterung
PCSystem_Control.exe/Help	Zeigt den Hilfetext im Kommandozeilen-Editor an.
PCSystem_Control.exe /HelpExitCode	Zeigt den Hilfetext zu den Errorcodes im Kommandozeilen-Editor an.
PCSystem_Control.exe/Help/RegWebserver	Zeigt den Hilfetext zu diesem Aufruf an.
PCSystem_Control.exe/Help/UnregWebserver	
PCSystem_Control.exe/Help/VerifyWebserver	
PCSystem_Control.exe/RegWebserver/	Der Dispatcher-Webserver registriert sich in den Internetinformationsdienst (IIS).
PCSystem_Control.exe/RegWebserver/HTTP	Der Aufruf erwartet eine vorhandene Bindung für das HTTP-Protokoll auf den Standard Port "80".
PCSystem_Control.exe/RegWebserver/HTTPS	Der Aufruf erwartet eine vorhandene Bindung für das HTTPS-Protokoll mit dem Standard Port "443".
PCSystem_Control.exe/RegWebserver/HTTP/HTTPS	Der Aufruf erwartet eine vorhandene Bindung für das HTTP-Protokoll mit dem Standard Port "80" und für das HTTPS-Protokoll mit dem Standard Port "443".
PCSystem_Control.exe/RegWebserver/Sitename"..."	Der Aufruf erwartet eine vorhandene Webseite im IIS mit dem übergebenen Namen.
PCSystem_Control.exe/UnregWebserver	Die Registrierung des Webservers in den IIS wird rückgängig gemacht.
PCSystem_Control.exe/VerifyWebserver	Zeigt an, ob der Benutzer eine Registrierung des Webserver in den IIS machen kann oder ob noch Vorbedingungen nötig sind.

Vorgehensweise bei Fehlermeldungen

Tritt bei einem Kommando-Befehl ein Fehler auf, haben Sie die Möglichkeit die Bit-Nummer des Fehler zu ermitteln und eine genaue Fehlerbeschreibung zu erhalten.

Gehen Sie folgendermaßen vor:

1. Um den Fehlercode zu erhalten geben Sie im Kommandozeilen-Werkzeug das Kommando "echo %errorlevel%" ein.
2. Über dieses Kommando erhalten Sie folgendes Ergebnis:
   - ist der Fehlercode = 0, liegt kein Fehler vor
   - ist der Fehlercode > 0, liegt ein Fehler vor und wird als Dezimalzahl ausgegeben
3. Die Dezimalzahl muss in eine Binärzahl umgerechnet werden.
4. Um eine Übersicht der Bit-Nummer mit Fehlerbeschreibung zu erhalten, geben Sie im Kommandozeilen-Werkzeug das Kommando „HelpExitCode“ ein.

Beispiel

Hier ein Beispiel für Umrechnung und einer Bit-Nummer mit der zugehörigen Fehlerbeschreibung:

- Dezimalzahl: 288
- Binärzahl:100100000

Ergebnis: Den Fehler können Sie an Bit-Nummer 5 "Err_Net_45_Full=1" und Bit-Nummer 8 "Err_IIS_Running=1" ablesen.

Zugriff über HTTPS

Voraussetzungen

Für einen fehlerfreien HTTPS-Zugriff gelten folgende Voraussetzungen:

- die aktuelle Uhrzeit ist eingestellt
- die Firewall ist passend konfiguriert
- ein gültiges Zertifikat ist installiert
  - Ist kein Zertifikat installiert, wird eine Empfehlung angezeigt, die Seite nicht zu benutzen. Des Weiteren haben Sie die Möglichkeit sich ein eigenes Zertifikat zu erstellen.

Eingabe und Beispiel

Um mit HTTPS-Protokoll auf die Webseiten zuzugreifen, geben Sie die URL in folgender Form ein:

- https://ww.xx.yy.zz, wobei ww.xx.yy.zz
  Beispielhafte Eingabe: https://192.168.3.141

Eine verschlüsselte Verbindung erkennen Sie am Schloss-Symbol in der Statusleiste der Webseite

Verweis

Weitere Informationen wie Sie sich ein eigenes Zertifikat erstellen finden Sie im Kapitel Erstellen eines eigenen Zertifikats (Seite 1541).

Erstellen eines eigenen Zertifikats

Sollte keine Certification Authority (CA) in Ihrem Unternehmen vorhanden sein, so können Sie die in diesem Kapitel beschriebene Vorgehensweise verwenden. Die Erstellung der Schlüsseldateien erfolgt mit Hilfe des Programms "OpenSSL". Falls Sie auf Ihrem PC noch kein OpenSSL installiert haben, können Sie dieses Programm über folgende Internetseite kostenlos herunterladen und installieren: http://openssl.org/ (http://openssl.org/)

Vorgehen

Zum Erstellen eines eigenen Zertifikats, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Anwendung "Internetinformationsdienste-Manager".
2. Klicken Sie auf den Rechner-Namen.
3. Doppelklicken Sie in der Ansicht "Feature" auf "Serverzertifikate".
   Das Verzeichnis "Serverzertifikate" wird geöffnet.
4. Klicken Sie im Bereich "Aktionen" auf "Selbstsigniertes Zertifikat erstellen".
5. Geben Sie auf der Seite "Selbstsigniertes Zertifikat erstellen" im Feld "Anzeigename für das Zertifikat" einen Name für das Zertifikat ein.
6. Klicken Sie "OK".
   Das neue Zertifikat wird in der Übersicht "Zertifikatserver" angezeigt.
7. Wechseln Sie wieder in den "Internetinformationsdienste-Manager".
8. Wählen Sie in dem Verzeichnis "Seiten" die Webseite aus, in der die neue HTTPS-Verbindung erzeugt werden soll.
9. Klicken Sie in der rechten Seite auf "Seite bearbeiten > Bindung".
   Das Dialogfeld mit bereits verfügbaren Webseiten wird geöffnet.
10. Wählen Sie die den Verbindungstyp "https" aus und klicken Sie auf "hinzufügen".
    Der Dialog mit den Verbindungseigenschaften öffnet sich.
11. Wählen Sie den Protokolltyp " HTTPS" und die Portnummer 443 aus und fügen Sie in der Auswahlbox "SSL-Zertifikate" das selbst erzeugte Zertifikat aus.
12. Klicken Sie auf "OK" um die neue Verbindung zu erstellen.

1.4.3.2 Integration von IPCs ins TIA Portal

PC-Systeme können Sie neben Steuerungs- und Visualisierungsaufgaben auch für folgende Anwendungen nutzen:

- Datengateway zum übergelagerten MES
- anlagennaher Datenserver
- Hardware-Plattform für SCADA Systeme
- Industrielle Workstations (z. B. an Prüfplätzen)
- Prozess- & Datenanalyse vor Ort

Integration von SIMATIC IPCs ins TIA Portal

Durch die Integration der SIMATIC PC-Station (IPC im TIA Portal) wird ein PC-System in eine Anlagenkonfiguration (TIA -Projekt) integriert, mit folgendenVorteilen:

- Schnellere Projektierung
  - Projektierung per Drag & Drop aus dem Hardwarekatalog
  - Detaillierte Anzeige aller Schnittstellen – Passend zur Konfiguration des SIMATIC IPCs
  - Übergreifende Netzwerkansicht inkl. Anzeige der IP-Adressen
  - Feingranulare Status-Diagnose der Netzwerktopologie
- Vereinfachte Inbetriebnahme
  - Zentraler Überblick aller IP-Adressen
    Ausschließen von Konflikten durch doppelte Vergabe
    Einfaches Finden des SIMATIC IPCs (inkl. Computer-Name und IP-Adresse) über “Erreichbare Teilnehmer”
  - Status-Diagnose innerhalb des TIA Portals
    Diagnose der Ethernet / PROFINET Verbindung zum SIMATIC IPC aus der Projektierung
    Einfache und schnelle Fehlerlokalisierung, z. B. Topologiefehler im Netzwerk, Geräte nicht erreichbar
    Portgranulare Diagnose (inkl. des in der Topologie vergebenen Ports am SCALANCE Switch)
  - Übersicht des gesamten Netzwerks
    Umfassende Informationen zu Topologie, Strukturen und Hierarchien
- Integration der Hardware Diagnose von IPCs im TIA Portal
  - Feingranulare Status-Diagnose der Netzwerktopologie, z. B. Topologiefehler im Netzwerk, Geräte nicht erreichbar
  - Präventive Wartung z. B. von Festplatten oder Lüftern
  - Frühzeitiges Reagieren auf ein Über-/Unterschreiten der zulässigen Betriebstemperatur
  - Zentraler Überblick zu allen SIMATIC IPCs in einer Anlage
  - Anzeige der Hardware Diagnose im WinCC Diagnostic Viewer
    Anzeige der PC-Diagnosen in der laufenden Anlage
    Präventive Wartung

Systemvoraussetzungen

Für die Installation einer PC-Station in einen IPC, müssen folgende Voraussetzungen erfüllt sein:

Voraussetzungen

- Sie haben Administrationsrechte auf Ihrem Computer.
- Es sind min. 400 MB freier Speicherplatz vorhanden.

Hardware-Voraussetzungen

Folgende IPCs werden unterstützt:

- SIMATIC IPC2x7E
- SIMATIC IPC4x7D
- SIMATIC IPC4x7E
- SIMATIC IPC547G
- SIMATIC IPC6x7D
- SIMATIC IPC8x7D

Hinweis

Achten Sie in Ihrer Konfiguration auf die Kompatibilitäten von IPCs

Wird die Fehlermeldung, die auf eine nicht passende Konfiguration hinweist ignoriert, kann es zu Fehlerdiagnosen im TIA Portal kommen.

Wählen Sie immer den zu Ihrer Konfiguration passenden IPC aus.

Software-Voraussetzungen

- Die SIMATIC IPC Software kann mit folgenden Betriebssystemen eingesetzt werden:
  - Windows 10 Enterprise IoT (für CPU 1515SP PC2, alle IPCs; Ausnahme: IPC4x7D)
  - Windows Embedded Standard 7 E/P SP1 32 Bit
  - Windows Embedded Standard 7 E/P SP1 64 Bit
  - Windows 7 SP1 32 Bit
  - Windows 7 SP1 64 Bit
- Folgende Diagnose-Software ist installiert:
  - DiagBase >= V1.5.0 oder
  - DiagMonitor >= V4.5.0

Einschränkungen

- Folgende Filter sind deaktiviert:
  - UWF ((für CPU 1515SP PC2, alle IPCs; Ausnahme: IPC4x7D)
  - EWF
  - FBWF
- Folgende Komponenten dürfen nicht installiert sein:
  - PC Station classic (S7-RTM)
  - CPU 150xS
  - SIMATIC NET
  - WinAC RTX

Installation

Speichermedien für die Installation

Über folgende Medien kann das Setup installiert werden:

- USB-Stick
- DVD
- Festplatte
- Netzlaufwerk

Installation

Das Setup wird durch Ausführen der "start.exe" im Setup-Ordner gestartet.

Gehen Sie dazu wie folgt vor:

1. Doppelklicken Sie die "start.exe".
2. Wählen Sie die gewünschten Produkteigenschaften für die Installation aus.
3. Folgen Sie während der Installation den Anweisungen auf dem Bildschirm.

Ergebnis

Nach erfolgreicher Installation sind folgende Komponenten installiert:

- PC Station V2.x
- S7 Device Drivers
- SIMATIC NET CP Manager Plus
- PC-Station Webserver

Des Weiteren wird im Infobereich der Taskleiste von Windows beim Betrieb der CPU auch ein Symbol für den Dienst der PC-Station angezeigt.

Verweis

Weitere Informationen zu den Installationsvoraussetzungen konfigurierter Komponenten finden Sie im Kapitel Vorgehen um ein PC-System zu konfigurieren (Seite 1519).

PC-Stationsanzeige im Infobereich

Im Infobereich der Taskleiste von Windows wird beim Betrieb der CPU auch ein Symbol für den Dienst der PC-Station angezeigt. Das Symbol zeigt unter anderem den aktuellen Zustand des Dienstes der PC-Station an und gibt Ihnen die Möglichkeit Konfigurationen vorzunehmen.
Ein Rechtsklick auf das Symbol im Infobereich öffnet das Kontextmenü der PC-Station.

Zustände des Infobereichsymbols

Der Zustand des Symbols für den Dienst der PC-Station im Infobereich der Taskleiste ändert sich sobald sich der Betriebszustand der PC-Station ändert.
Das Infobereichsymbol kann folgende Zustände anzeigen:

RUN	STOP

Konfigurationsmöglichkeiten über das Symbol des Service "Station Manager"

Das Symbol des Dienstes der PC-Station im Infobereich der Taskleiste gibt Ihnen über das Kontextmenü folgende Konfigurationsmöglichkeiten:

- Import > Konfiguration importieren
  Dieser Kontextmenübefehl bewirkt, dass die Metadaten einer ausgewählten PC-System-Konfigurationsdatei geöffnet und angezeigt werden. Der Benutzer startet hiermit auch den Importvorgang der psc-Datei. Die Funktion wird für F-CPUs nicht unterstützt.
- Konfiguration
  - Bestehende Konfiguration löschen
    Für diese Konfigurationsmöglichkeit benötigen Sie Administratorrechte.
    Dieser Kontextmenübefehl bewirkt, dass die Konfiguration gelöscht wird.
    Diese Funktion ist bei einer F-CPU defaultmäßig deaktiviert. Explizit Berechtigte müssen der Benutzergruppe "Failsafe Operators" hinzugefügt werden um sie individuell zu berechtigen. Des Weiteren bleiben bei einer F-CPU die projektierte Passwörter nach dem Löschen der Konfiguration erhalten. Diese Passwörter werden für einen späteren Download wieder benötigt. Um die Passwörter zurückzusetzen, laden Sie ein Projekt ohne Passwörter.
  - Änderung des Speicherpfads der Konfigurationsdaten
    Für diese Konfigurationsmöglichkeit benötigen Sie Administratorrechte.
    Wenn Sie z. B. eine Partition mit einem erweiterten Schreibfilter (EWF) schützen, sind dadurch auch die Konfigurations- und Diagnosedaten schreibgeschützt.
    Legen Sie die Diagnosedaten in einem nicht schreibgeschützten Bereich der Festplatte ab. Sie können auch die Konfigurationsdaten, die die Projektierung enthalten, in einem nicht schreibgeschützten Bereich der Festplatte ablegen.

- Neustart aller Dienste der PC-Station
  Für diese Konfigurationsmöglichkeit benötigen Sie Administratorrechte.
  Dieser Kontextmenübefehl bewirkt, dass alle Dienste der PC-Station neu gestartet werden.
- Beenden
  Dieser Kontextmenübefehl bewirkt, dass das PC-Station Panel beendet wird. Das Symbol für den Dienst der PC-Station im Infobereich der Taskleiste wird ausgeblendet. Starten Sie das PC-Station Panel erneut über folgenden Eintrag im Startmenü von Windows: **Siemens Automation > SIMATIC > PC Station > “PC Station”**

Reparieren/Deinstallation

Sie haben die Möglichkeit, das installierte Programm zu reparieren oder zu deinstallieren.

Reparieren

Um das Programm zu reparieren gehen Sie wie folgt vor:

1. Navigieren Sie zu dem Pfad auf dem Ihr Setup installiert ist.
2. Doppelklicken Sie auf die "start.exe".
3. Markieren Sie die "TIA -Integration für SIMATIC IPC" mit einem Haken und klicken Sie auf "Weiter".
4. Wählen Sie den Button "Reparieren".
5. Klicken Sie auf weiter und folgen Sie den Anweisungen auf dem Bildschirm.

Deinstallieren

Um das Programm zu deinstallieren haben Sie zwei Möglichkeiten.

Über die Systemsteuerung

1. Wählen Sie in der Windows Startleiste **Start > Systemsteuerung > Run Advertised Programs**
2. Wählen Sie das Programm "TIA -Integration für SIMATIC IPC" aus.
3. Klicken Sie auf "Deinstallieren".
4. Folgen Sie während der Deinstallation den Anweisungen auf dem Bildschirm.

Über die "start.exe":

1. Navigieren Sie zu dem Pfad auf dem Ihr Setup installiert ist.
2. Doppelklicken Sie auf die "start.exe".
3. Markieren Sie die "TIA -Integration für SIMATIC IPC" mit einem Haken und klicken Sie auf "Weiter".
4. Wählen Sie den Button "Deinstallieren".
5. Klicken Sie auf weiter und folgen Sie den Anweisungen auf dem Bildschirm.

Einschränkungen bei Abbruch

Wenn das Setup abgebrochen wird, dann müssen alle Installationsschritte erneut erfolgen.

1.4.3.3 Grundkonfiguration des Open Controllers

Konfiguration anlegen

Sie haben im TIA Portal ein neues Projekt erstellt.

Um die Konfiguration im TIA Portal anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Neues Gerät hinzufügen".
2. Wählen Sie unter "PC-Systeme > Open Controller" die "CPU 1515SP PC (+HMI)" aus.
3. Wählen Sie die entsprechende Version aus.
   In der Gerätesicht wird der konfigurierte Open Controller angezeigt.
4. Auf dem Open Controller sind folgende Komponenten zu sehen:
   – Onboard-Schnittstelle X2 (GB-Ethernet Windows-Schnittstelle), die direkt der PC-Station zugeordnet ist (1 Port)
   – wechselbarer Busadapter X1, der direkt dem Software-Controller zugeordnet ist (2 Ports)
   – auf der rechte Seite (hinter den gesteckten Modulen im Baugruppenträger): der projektierter Software-Controller CPU 1505SP und die WinCC Runtime Advanced.
5. Fügen Sie das Servermodul ein.
   Das Servermodul am rechten Ende des Aufbaus bildet den Abschluss der CPU mit den Peripheriemodulen.

Einstellungen der IP-Adressen:

Im Inspektorfenster unter "Eigenschaften":

- Onboard-Schnittstelle X2: die projektierte, vorgegebene IP-Adresse ist identisch mit der Windows- IP-Adresse die bei der Erstinbetriebnahme auf dem Open Controller eingestellt ist.
- Busadapter X1: die projektierte, vorgegebene IP-Adresse ist identisch mit der IP-Adresse die bei der Erstinbetriebnahme im Panel des Software-Controller (Display-Applikation) eingestellt worden ist.

Wichtige Eigenschaften des Software-Controllers

Ändern Sie bei Bedarf die Eigenschaften im Inspektorfenster unter "Eigenschaften":

- Anlaufart auswählen
- Systemdiagnose
- Speicherort für remanente Daten einstellen
- Kopierschutz einrichten
- LEDs der Hardware nutzen

- Webserver konfigurieren
- Schnittstellen für die Kommunikation zuweisen

HMI-Verbindung herstellen

1. Klicken Sie in der Gerätesicht mit der rechten Maustaste auf die WinCC RT Advanced.
2. Starten Sie den Bediengeräte-Assistenten.
3. Übernehmen Sie alle Voreinstellungen.
   Der Assistent erstellt Ihnen Systembilder mit der dazu passenden Navigation.
4. Wechseln Sie in der Netzsicht auf Verbindungsansicht.
5. Klicken Sie auf die WinCC RT Advanced.
6. Ziehen Sie mit gedrückter Maustaste von der Software WinCC RT Advanced auf den S7-1500 Software-Controller (z. B. CPU 1505SP).
   Eine Netzwerkverbindung zwischen den beiden Geräten wird hergestellt.

Projekt in das Zielsystem laden

Beim erstmaligen Laden der Projektierung wird die komplette PC-Station geladen.

1. Wählen Sie in der Gerätesicht das gesamte PC-System aus.
2. Klicken Sie auf die Schaltfläche "Download".
3. Nehmen Sie folgende Einstellungen vor:
   - Verbindungsart
   - Schnittstelle des PGs festlegen
   - X2-Schnittstelle auf dem Open Controller festlegen

Hinweis

Der erste TIA Portal-Download **muss** über die Schnittstelle "X2" erfolgen.

Der Name der Schnittstelle X2 PN/IE (LAN) darf im TIA Portal nur ASCII-Zeichen enthalten, z. B. PROFINET_2.

Laden und übersetzen Sie das Projekt. Die Hardware-Projektierung und der erste Download sind nun abgeschlossen.

WinCC RT Advanced

WinCC Runtime Advanced beinhaltet alle wesentlichen Funktionen für das Bedienen und Beobachten von Maschinen oder Anlagen.

Verweis

Weitere Informationen zur Konfiguration und zum Laden finden Sie in Gerätehandbuch ET 200SP Open Controller CPU 1515SP PC (https://support.industry.siemens.com/cs/de/de/view/109248384).

Weitere Informationen zu WinCC RT Advanced finden Sie im Systemhandbuch WinCC Advanced V13.0 SP1 (https://support.industry.siemens.com/cs/de/de/view/109091876).

1.4.3.4 Parametrierung

Hochfahren des Software Controllers (Anlauf-Parameter)

Konfiguration der Anlaufart der CPU

Zusätzlich zu den Standard-Anlaufparametern, z. B. "Anlauf nach NETZ-EIN" haben Sie bei einer CPU 150xS die Möglichkeit, das Startverhalten zu parametrieren. Die CPU kann auf zwei verschiedene Arten gestartet werden (NETZ-EIN):

- Manueller Start durch die Schaltfläche "Power" auf dem Display der CPU
- Automatischer Start beim Starten des PCs (empfohlen)

Mit der Option "Automatisches Starten nach dem Hochfahren des PCs" in den Eigenschaften des Software Controllers bestimmen Sie, wie die CPU startet.

Wir empfehlen die Option "Automatisches Starten nach dem Hochfahren des PCs" zu aktivieren, da sonst der Software Controller nach einem Start des PC-Systems nicht automatisch hochfährt. Nur ein hochgefahrener Software Controller kann geladen werden.

Wenn die Option deaktiviert ist, müssen Sie die CPU vor einem Ladevorgang manuell über das Display der CPU starten.

Hinweis

BIOS-Speichertest bei SIMATIC IPCs

PCs bieten die Möglichkeit eines Speichertests. Einige Hardware Tests, wie der Speichertest, sind im BIOS-Setup-Programm standardmäßig deaktiviert und werden im Anlauf des PCs übersprungen. Dadurch wird der Boot-Vorgang beschleunigt.

Wenn Sie die CPU auf einem SIMATIC IPC oder CPU 1515SP PC nutzen, darf der BIOS-Speichertest nicht aktiviert sein.

Die Standard-Anlaufparameter unterscheiden sich nicht von den Anlaufparametern der CPUs S7-1500.

Anlaufart einstellen

Um die Anlaufart einzustellen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7.
2. Öffnen Sie Ihr Projekt.
3. Wechseln Sie in die Projektansicht.
4. Öffnen Sie die Gerätesicht.
5. Selektieren Sie die CPU.

6. Wählen Sie im Inspektorfenster im Register "Eigenschaften > Allgemein" den Bereich "Anlauf".
7. Konfigurieren Sie das Anlaufverhalten Ihrer CPU.

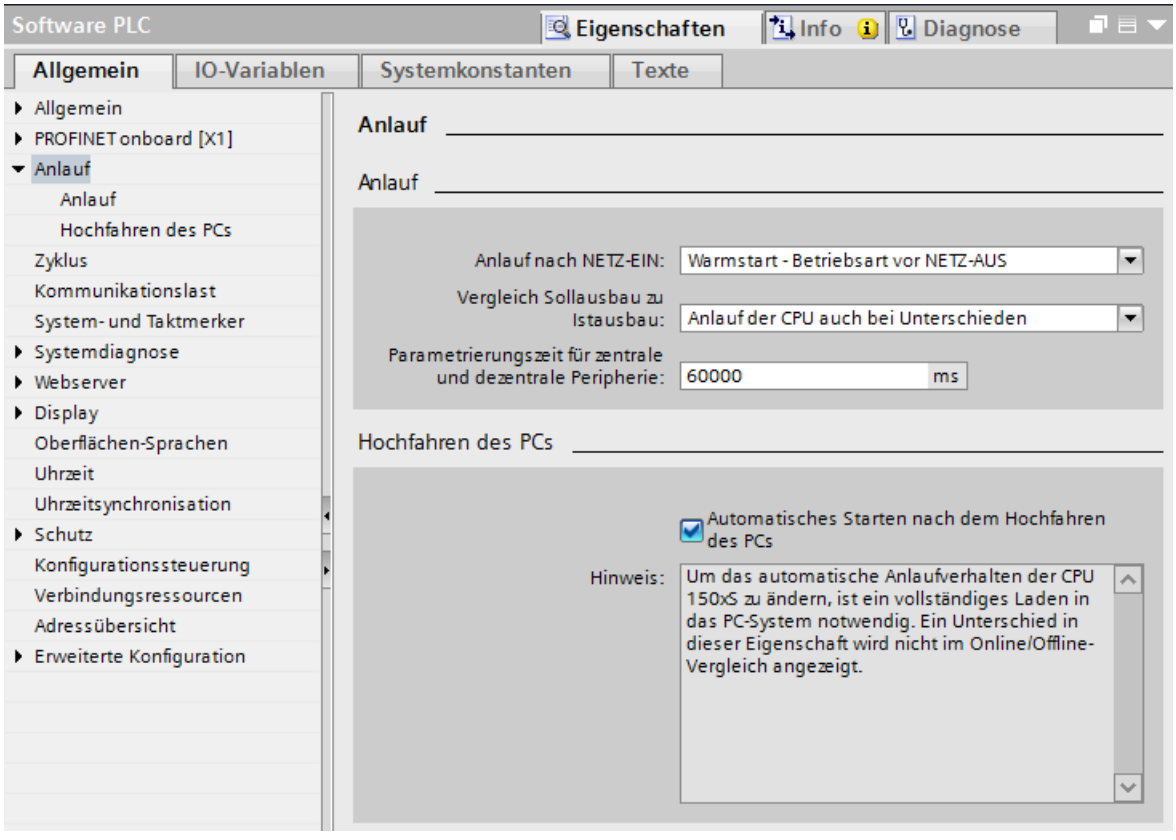

8. Laden Sie das Projekt in die CPU, indem Sie das PC-System selektieren, bevor Sie den Ladevorgang starten.

Ergebnis

Wenn Sie zusätzlich zu den Einstellungen zur Anlaufart auch an der Option "Automatisches Starten nach dem Hochfahren des PCs" Änderungen vornehmen, wird die CPU vor dem Ladevorgang automatisch beendet. Zu Beginn des Ladevorgangs startet die CPU wieder im Betriebszustand STOP.

Das Projekt wird geladen. Die neuen Einstellungen zur Anlaufart sind aktiv.

Besonderheiten bei der Parametrierung des Zugriffsschutzes

Zugriffsschutz in STEP 7 parametrieren

Die Parametrierung des Zugriffsschutzes erfolgt über die Eigenschaften des PC-Systems, in dem der Software Controller steckt.

In den Eigenschaften der CPU sind die Zugriffsstufen nur lesbar. Um in die Eigenschaften des PC-Systems zu gelangen, klicken Sie auf den Pfeil. Hier können Sie die Zugriffsstufen des PC-Systems definieren und für die gesamte Anwendung überschreiben.

Hinweis

Parametrierung des Zugriffsschutzes für das gesamte PC-System

Die Parametrierung des Zugriffsschutzes erfolgt anders als bei einer Hardware-CPU nicht direkt in den Eigenschaften der CPU. Dadurch wird sichergestellt, dass einheitliche Schutzstufen-Passwörter für alle Komponenten eines PC-Systems konfiguriert werden.

Anders als bei einer Hardware-CPU des Automatisierungssystems S7-1500 kann die CPU nicht mit einem separaten Display Passwort gegen unbefugte Zugriffe geschützt werden. Da die CPU auch über Remote-Zugang gesteuert werden kann, nutzt sie die Zugriffsschutz-Passwörter aus STEP 7, um den Zugriffsschutz auf das Display zu gewährleisten.

Verweis

Weitere Informationen zum Schutzkonzept in Verbindung mit dem Display-Zugriff finden Sie im Bedienhandbuch SIMATIC S7-1500 CPU 150xS (https://support.industry.siemens.com/cs/de/de/view/109249299).

Erweiterte Konfiguration für Software Controller

Wenn Sie in der Gerätesicht oder Netzsicht einen Software Controller S7-150xS markieren, finden Sie im Inspektorfenster einen Bereich "Erweiterte Konfiguration". Im Folgenden sind die Parameter unter "Erweiterte Konfiguration" erläutert bzw. Sie finden Verweise auf die Erläuterung dieser Parameter.

PC-Kommunikationsschnittstelle

Bei der PC-Kommunikationsschnittstelle handelt es sich um eine Schnittstelle, die Ihnen unabhängig von einem PROFIBUS- oder PROFINET-Subnetz Zugriff über eine von Windows kontrollierte Schnittstelle auf z. B. ein Büronetzwerk bietet. Über diese Schnittstelle kommuniziert Ihr Anwenderprogramm mit anderen Geräten oder Sie rufen über diese Schnittstelle Informationen über den Webserver des Software Controllers ab:

Zugriff auf den Webserver

Wie bei den Hardware-Schnittstellen auch müssen Sie den Zugriff über die PC-Kommunikationsschnittstelle aktivieren, um über diese Schnittstelle auf den Webserver des Software Controllers zugreifen zu können.

Das folgende Bild zeigt den Webserver-Zugriff vom Webbrowser des PC-Systems über die PC-Kommunikationsschnittstelle.

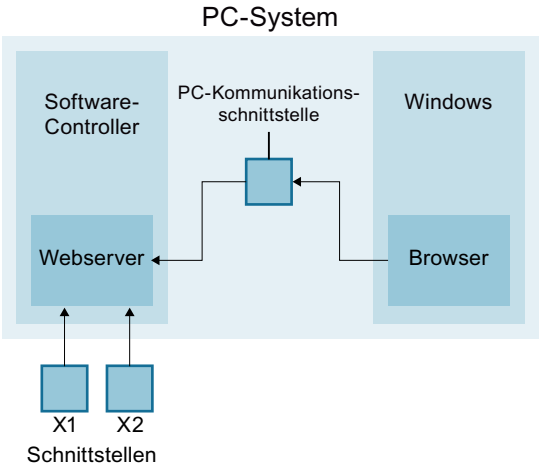

Kommunikation über Open User Communication (OUC)

Über die PC-Kommunikationsschnittstelle können Sie auch Ethernet-Schnittstellen des PCs nutzen.
Im Wesentlichen wird Ihnen bei der Projektierung nur der Name der Host-Schnittstelle und die Hardware-ID angezeigt.
In den Eigenschaften einer Kommunikationsanweisung (z. B. TSEND_C) greifen Sie auf die Eigenschaften der PC-Kommunikationsschnittstelle zu und nutzen den Bereich "Verbindungsdaten", um auf die Adressparameter für die programmierte OUC Kommunikation zuzugreifen. Die Hardware-ID der PC-Kommunikationsschnittstelle wird in die Verbindungsdaten eingetragen.

Index

Der Index einer Komponente im PC-System entspricht der Lage der Komponente (z. B. einer Hardware-Ressource wie eine Schnittstelle). Es handelt sich um eine Art "Steckplatzadresse am PC-internen Subnetz" (Softbus).

In der grafischen Ansicht wird der Index einer Software- oder Hardware-Komponente nicht angezeigt.

Bei markiertem Software Controller in der Netzsicht oder Gerätesicht können Sie den Index in den Eigenschaften des Software Controllers unter "Erweiterte Konfiguration" ablesen bzw. ändern. Den Index können Sie auch in der Geräteübersichtstabelle in der Gerätesicht ändern.

Verweis

Informationen zur Nutzung der LEDs der Hardware finden Sie hier (Seite 1555).

Informationen zu den Einstellungen zum Speicherort für remanente Daten finden Sie hier (Seite 1554).

Speicherort für remanente Daten einstellen

Der S7-1500 Software Controller (CPU) bietet die Möglichkeit, Daten beim Beenden und im Spannungsausfall remanent im Massenspeicher des PCs oder im integrierten NVRAM zu speichern. Die Art der Datenspeicherung stellen Sie in den Eigenschaften der CPU ein.

Hinweis

Datenverlust nach Ändern der Speicherart

Wenn Sie die Speicherart ändern, dann werden beim Laden der geänderten Konfiguration die aktuellen remanenten Daten und die Inhalte des Diagnosepuffers gelöscht.

Voraussetzung

Die Projektsicht ist geöffnet.

Die Gerätesicht ist geöffnet.

Vorgehen

Um die Speicherart zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU.
2. Wählen Sie im Inspektorfenster im Register "Eigenschaften" den Bereich "Erweiterte Konfiguration".
   - Aktivieren Sie das Optionsfeld "PC-Massenspeicher" zum Speichern der remanenten Daten auf dem Massenspeicher Ihres PCs.
   - Aktivieren Sie das Optionsfeld "NVRAM der PC-Plattform" zum Speichern der remanenten Daten im integrierten NVRAM Ihres PCs.

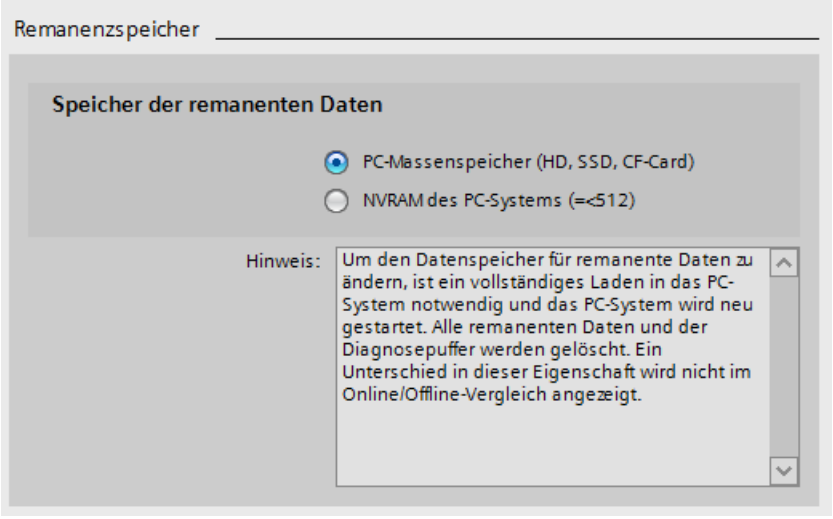

3. Um die Art der Datenspeicherung in STEP 7 abschließend zu ändern, laden Sie das Projekt erneut in das PC-System.

Hinweis

Änderung der Konfiguration laden

Wenn Sie die Speicherart zum Speichern der remanenten Daten in STEP 7 ändern, müssen Sie zur Änderung der Konfiguration das komplette PC-System laden. Wenn Sie die Speicherart wechseln, startet die CPU während des Ladevorgangs neu. Ohne einen Neustart wird die Speicherart trotz des Ladevorgangs nicht geändert.

Um den erforderlichen Neustart der CPU zu gewährleisten, müssen Sie den PC laden und anschließend neu starten.

Siehe auch

Erweiterte Konfiguration für Software Controller (Seite 1552)

LEDs der Hardware nutzen

Der S7-1500 Software Controller (CPU) kann seinen Status auf den LEDs der Hardware-Plattform anzeigen, auf der er installiert ist. Diese Funktionalität stellen Sie in den Eigenschaften der CPU in STEP 7 ein.

Hinweis

Zeitgleicher Zugriff mehrerer Komponenten

Achten Sie darauf, dass nicht mehrere konkurrierende Komponenten zeitgleich Zugriff (z. B. DiagBase und CPU) auf die Hardware-LEDs haben.

Voraussetzung

Ihr Projekt ist geöffnet.

Sie befinden sich in der Projektansicht.

Vorgehen

Um die LEDs der Hardware-Plattform für die CPU zu nutzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Netzsicht oder die Gerätesicht.
2. Markieren Sie die CPU.

3. Wählen Sie im Inspektorfenster im Register "Eigenschaften > Allgemein" den Bereich "Erweiterte Konfiguration > Hardware-LED".

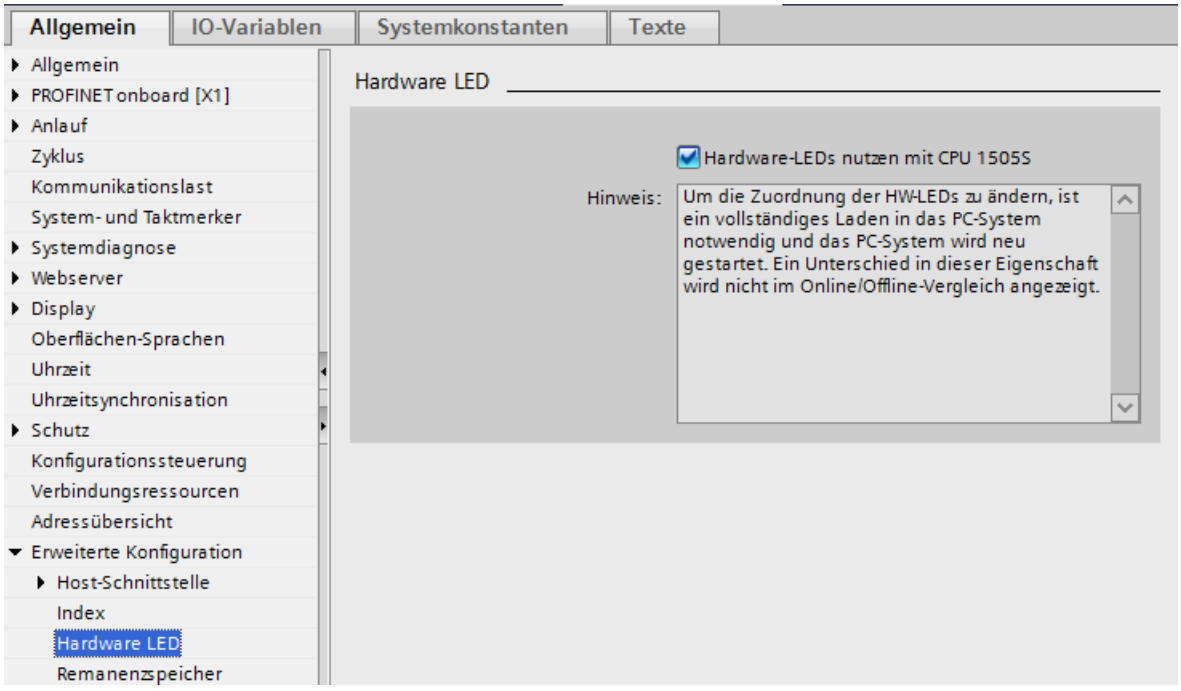

4. Aktivieren Sie die Option "Hardware-LEDs nutzen mit CPU 150xS".

Wie im Hinweis unter der Option beschrieben ist, müssen Sie das PC-System anschließend vollständig laden und neu starten.

Ergebnis

Der PC wird heruntergefahren und die CPU beendet. Das Projekt wird geladen.

Der PC wird automatisch neu gestartet. Die CPU startet im Betriebszustand STOP. Der Ladevorgang wird fortgesetzt. Die CPU wird nach Beendigung des Ladevorgangs in den Betriebszustand RUN versetzt.

Besonderheit

Wenn ein Online/Offline-Unterschied hinsichtlich der Eigenschaft "Hardware-LEDs nutzen" besteht, wird dieser Unterschied beim Online/Offline-Vergleich nicht angezeigt.

Siehe auch

Erweiterte Konfiguration für Software Controller (Seite 1552)

Wissenswertes zum Schnittstellen zuweisen

Ab STEP 7 V13 SP1 werden Beziehungen zwischen einer Software-Komponente wie z. B. einem Software Controller und ihrer Schnittstellen zur Außenwelt, z. B. einer PROFINET-Schnittstelle, durch Bezugslinien dargestellt.

Darstellung von Beziehungen

Wenn nichts markiert wurde, sind alle Schnittstellen und Software-Komponenten mit einer grauen Linie verbunden. Die Linien haben am Ende mit einem rechteckigen Stecker. Diese Linien zeigen die Beziehung zwischen Software-Komponenten (z. B. S7-1500 Software Controller, WinCC RTX oder OPC Server) und Schnittstellen (PROFINET, PROFIBUS, ...) an.

Wenn Sie eine Schnittstelle markieren, erkennen Sie zugeordnete Software-Komponenten an der blauen Linie, die sich deutlich von den grauen Linien abhebt.

Schnittstellen ohne Beziehungslinien bzw. mit grauen Bezugslinien sind nicht zugeordnet.

Beispiel 1: Schnittstelle dem Software Controller (CPU) zuweisen

Eine blaue Bezugslinie zwischen einer markierten Schnittstelle und einem S7-1500 Software Controller bedeutet, dass die Schnittstelle dem S7-1500 Software Controller zugeordnet ist und nur von ihm verwendet werden kann. In diesem Fall kann der Software Controller als IO-Controller ein PROFINET IO-System an dieser Schnittstelle betreiben.

Nur bei dieser Art der Schnittstellenzuweisung ist es möglich, die IP-Adresse und den Gerätenamen der PROFINET-Schnittstelle zu konfigurieren und in das PC-System zu laden.

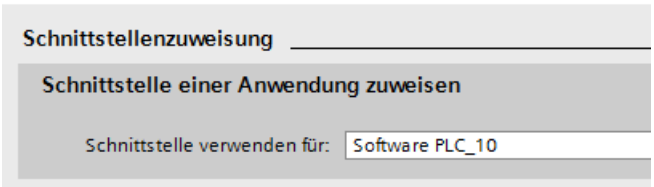

Beispiel 2: Schnittstelle der SIMATIC PC-Station zuweisen

Eine blaue Bezugslinie von einer markierten Onboard-Schnittstelle zu mehrere Software-Komponenten bedeutet, dass diese Schnittstelle der PC-Station zugeordnet ist und damit von allen Software-Komponenten verwendet werden kann. Über diese Schnittstelle kann z. B. PG-Kommunikation betrieben werden.

Bei dieser Art der Schnittstellenzuweisung können Sie zwar die IP-Adresse der Schnittstelle editieren, diese Einstellung wird aber nicht in der PC-System geladen! Die IP-Adresse muss unter Windows im PC-System eingestellt werden. Sinnvoll ist der Abgleich der IP-Adressen trotzdem, um z. B. im Dialog "Erweitertes Laden" genau nur diese PC-Station angeboten zu bekommen (Filterfunktion).

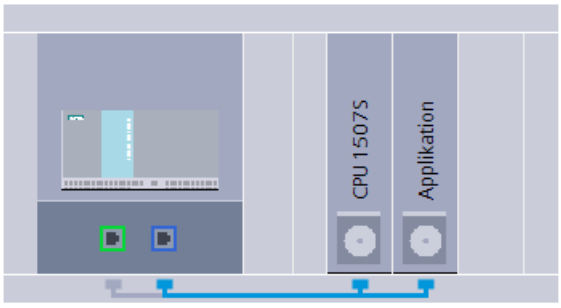

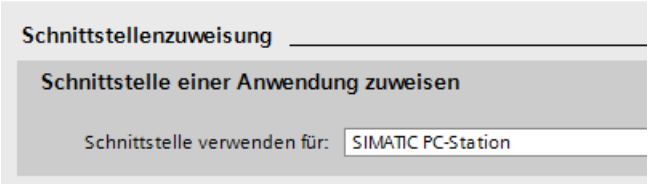

Beispiel 3: Keine oder eine andere Windows-Einstellung

Wenn Sie einer Onboard-Schnittstelle die Option "Keine oder eine andere Windows-Einstellung" zuweisen, dann ist keine Beziehung zu den gesteckten Software-Komponenten möglich und daher auch nicht sichtbar. Lediglich Windows Applikationen können über diese Schnittstelle verfügen; diese sind aber in der Konfiguration nicht sichtbar. Ebenso wenig sichtbar in der Konfiguration ist die IP-Adresse der Schnittstelle.

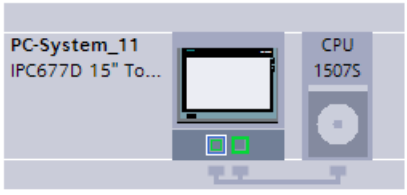

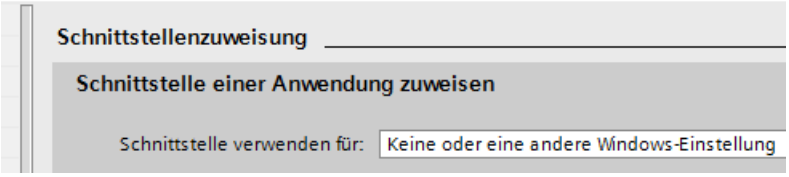

Anzeige der Lage der Schnittstelle

Unterhalb des Bereichs "Schnittstellenzuweisung" im Inspektorfenster wird die Lage der Schnittstelle angezeigt. Je nach Zuordnung zur PC-Station oder zum Software Controller wird der Index bzw. die Hardware-Ressource/der Schnittstellentyp angezeigt.

Schnittstellen für die Kommunikation der PC-Station V2.x zuweisen

Schnittstellenarten

Die PC-Station verwendet die Ethernet-Schnittstellen des PC für die Kommunikation. Über diese Ethernet-Schnittstellen sind folgende Kommunikationsdienste nutzbar:

- Engineering-Zugang zur PC-Station und den installierten Komponenten wie S7-1500 Software Controller oder WinCC RT Advanced
- HMI-Verbindungen zur PC-Station oder zum Software Controller
- Datenkommunkiation (S7-Kommunikation des Software Controllers)
- HMI-Verbindungen von z. B. WinCC RT Advanced zu externen Steuerungen
- Geroutete S7-Verbindungen durch den PC

Kommunikation zwischen Geräten

Um projektierte Kommunikation zwischen den Komponenten auf dem PC und weiteren SIMATIC Geräten zu ermöglichen, ist eine Zuweisung der Schnittstelle zur PC-Station erforderlich. Folgende Ethernet-Schnittstellen können der PC-Station V2.x zugewiesen werden:

- Onboard-Ethernet-Schnittstellen, z. B. X1 und X2, der SIMATIC IPC (nicht: Onboard CP 1616)
- Onboard X2-Schnittstelle der CPU 1515SP PC

Es können maximal 2 Onboard-Schnittstellen zugewiesen werden.

Engineeringfunktionen sind mit allen Ethernet-Schnittstellen des PC möglich, sofern sie unter Windows betrieben werden, auch wenn diese nicht der PC-Station zugewiesen wurden.
Für eine einfache Inbetriebnahme ist es jedoch erforderlich, dass mindestens eine Schnittstelle zugewiesen wird:

- SIMATIC PC: X1 Schnittstelle
- CPU 1515SP PC: X2 Schnittstelle
- keine Verwendung von externen CPs

Hinweis

Dem S7-1500 Software Controller können auch Schnittstellen für die exklusive Nutzung zugewiesen werden. Diese stehen dann für die PC-Station nicht zur Verfügung.

Vorgehen

Um die Schnittstellen für die Kommunikation dem PC-System zuzuweisen, gehen Sie wie folgt vor:

1. Selektieren Sie das PC-System.
2. Markieren Sie die gewünschte integrierte Schnittstelle in der Gerätesicht.
3. Wählen Sie im Inspektorfenster den Bereich "Allgemein > Schnittstellenzuweisung".
4. Weisen Sie die Schnittstelle der "SIMATIC-PC-Station" zu.
5. Übersetzen Sie das Projekt mit "Bearbeiten > Übersetzen".
6. Laden Sie die Hardware-Konfiguration des PC-Systems in das Zielgerät.

Ergebnis

Nach dem Ladevorgang können die Schnittstellen für konfigurierte Kommunikationsdienste verwendet werden.

Verweis

Weitere Information zu den Besonderheiten bei der Konfiguration von Schnittstellen mit dem S7-1500 Software Controller finden Sie im Bedienhandbuch SIMATIC S7-1500 CPU 150xS (https://support.industry.siemens.com/cs/de/de/view/109249299).

SIMATIC NET Version

Informationen zur SIMATIC NET Version finden Sie beim Service&Support unter der Beitrags-ID:63098071 (https://support.industry.siemens.com/cs/document/63098071?lc=de-WW).

SIMATIC NET Version für Onboard-Schnittstellen

Die SIMATIC NET Version für die Onboard-Schnittstellen finden Sie folgendermaßen:

1. Markieren Sie die Onboard-Schnittstelle.
2. Wählen Sie im Inspektorfenster den Bereich "Allgemein > Kataloginformationen".
   - Für ein **PC-System V2.x** stellt STEP 7 automatisch die Version der PC-Station ein, sobald ein S7-1500 Software Controller konfiguriert wurde. Diese Einstellung sollte nicht geändert werden. Es kann kein SIMATIC NET verwendet oder projektiert werden.
   - Für ein **PC-System V1.0** wird die aus dem Hardware Katalog bekannte neueste Version der SIMATIC NET CD angezeigt; z. B. V8.2, siehe auch Beschreibungstext. Die tatsächlich verwendete SIMATIC NET Version ist STEP 7 nicht bekannt, da Sie das PC-System V1.0 mit S7-RTM (Stationmanager) von der SIMATIC NET CD installieren und konfigurieren.

SIMATIC NET Version für die PC-Station

Wenn Sie ein PC-System in der Netzsicht markieren und im Inspektorfenster den Bereich "PC-Station" wählen, sind die verschiedenen PC-System-Versionen in der Klappliste wählbar.

- Option "PC-System V1.0" gewählt: Es wird **keine** SIMATIC NET Version angezeigt, da die tatsächlich verwendete SIMATIC NET Version STEP 7 nicht bekannt ist, siehe oben.
- Option "PC-System V2.x" gewählt: Es kann kein SIMATIC NET verwendet oder projektiert werden.

1.4.3.5 Besonderheiten zum S7-modular Embedded Controller

Besonderheiten zum S7-modular Embedded Controller

Empfehlung

Für Neuprojekte empfehlen wir auf die neue Produktgeneration mit S7-1500 Software Controllern umzusteigen.

Verweis

Weitere Informationen zum S7-modularen Embedded Controller finden Sie in der Betriebsanleitung SIMATIC Embedded Automation S7-modular Embedded Controller (https://support.industry.siemens.com/cs/de/de/view/37971572).

Einführung und Prinzipielles Vorgehen

Einleitung

S7-modular Embedded Controller (S7-mEC) sind PCs in S7-300-Bauform, mit vorinstallierter und vorkonfigurierter Software:

- Betriebssystem Windows Embedded Standard
- WinAC RTX (F)
- SIMATIC NET
- WinCC Runtime (nur bei EC31-HMI/RTX)

Ethernet-Schnittstellen des S7-modular Embedded Controller

S7-modular Embedded Controller hat zwei integrierte Industrial Ethernet-Schnittstellen:

- Die Schnittstelle X1 PN LAN P1 / X1 PN LAN P2 kann als PROFINET-Schnittstelle konfiguriert werden.
- Die Schnittstelle X2 P1 PN/IE LAN ist unter IE General der PC-Station in Index 3 zugewiesen und für die Industrial Ethernet-Kommunikation vorkonfiguriert.

Hinweis

Wenn Sie auf S7-mEC über Remote Desktop Protokoll zugreifen, dann stellen Sie sicher, dass sich S7-mEC und der Engineering-PC / das PG im selben Ethernet-Subnetz befinden.

Für den Zugriff auf S7-mEC über Remote Desktop Protokoll müssen Sie den Netzanschluss IE General verwenden (in der Regel Schnittstelle X2 P1 PN/IE LAN).

Prinzipielles Projektieren und programmieren des S7-mEC

Da S7-mEC vorkonfigurierte Systeme sind, gehen Sie folgendermaßen vor:

1. Erstellen Sie ein Projekt.
2. Fügen Sie eine PC-Station ein.
3. Konfigurieren Sie die Hardware.
4. Erstellen Sie ein S7-Programm.
5. Projektieren Sie bei EC31-HMI/RTX die Visualisierung in WinCC.
6. Laden Sie die Projektierungsdaten in WinLC RTX EC (Download).

EC31-HMI/RTX: Visualisierung projektieren

Nur bei EC31-HMI/RTX

Das Visualisierungsprojekt für EC31-HMI/RTX erstellen Sie auf dem Engineering-PC mit SIMATIC WinCC.

Voraussetzung

- WinCC ist auf dem Engineering-PC installiert.
- Ein WinCC RT-Objekt ist in der PC-Station eingefügt.

Vorgehen

Um ein Visualisierungsprojekt für EC31-HMI/RTX zu erstellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätesicht der PC-Station des EC31-HMI/RTX.
2. Selektieren Sie das WinCC RT-Objekt.

3. Wählen Sie aus den Befehl "Objekt öffnen".
   WinCC wird gestartet.
4. Erstellen und projektieren Sie die notwendigen Elemente: Bilder, Meldungen, Variablen, usw. mit Hilfe der Editoren von WinCC.
5. Speichern und testen Sie die Visualisierung.
6. Transferieren Sie das Projekt auf das Zielsystem EC31-HMI/RTX.

Projektierungsdaten in das Zielsystem laden

Lademöglichkeiten

Sie können folgende Projektierungsdaten aus STEP 7 in den Embedded Controller laden:

- gesamte PC-Station (z. B. einschließlich IE Allgemein)
- nur Konfigurationsdaten der WinLC RTX / WinLC RTX F
- nur S7-Programm der WinLC RTX / WinLC RTX F

Bei EC31-HMI/RTX müssen Sie die Projektierungsdaten aus WinCC transferieren.

Voraussetzung

- Der erweiterte Schreibfilter (EWF) auf dem Embedded Controller muss deaktiviert sein, wenn die gesamte PC-Station geladen werden soll.
- Der Embedded Controller ist mit dem PG/PC über Industrial Ethernet verbunden. Die PG/PC-Schnittstelle ist auf TCP/IP eingestellt.
- **Nur EC31-HMI/RTX:** Das Visualisierungsprojekt ist in WinCC erstellt.
- Für folgende Funktionen muss der Embedded Controller mit dem Engineering-System über einen Port der X1 PN LAN-Schnittstelle verbunden sein:
  - PROFINET IO Gerätenamen vergeben
  - PROFINET CBA: Verschaltungs-Download und Online-Funktionen

Vorgehen

1. Deaktivieren Sie EWF auf dem EC31-RTX / EC31-RTX F / EC31-HMI/RTX (wenn noch nicht geschehen). Nach dem Deaktivieren müssen Sie den Embedded Controller ausschalten und erneut starten.
2. Starten Sie WinLC.
3. Aktivieren Sie die PC-Station im Komponenten-Konfigurator (Schaltfläche "Station freigeben").
4. Schalten Sie den Embedded Controller in den Betriebszustand STOP.
5. Laden Sie die Hardware-Konfiguration, das S7-Programm und die Konfigurationsdaten in das Zielsystem.

Fertigstellen für den Betrieb

Um EC31-RTX / EC31-HMI/RTX für den Betrieb fertig zu stellen, gehen Sie folgendermaßen vor:

1. Versetzen Sie den erweiterten Schreibfilter für das Laufwerk C:\ in den Zustand "aktiviert".

ACHTUNG
Ausfall des Flash-Speichers Wenn Sie den erweiterten Schreibfilter im Zustand "deaktiviert" lassen, dann kann dies aufgrund des ständigen Schreibzugriffs des Betriebssystems zu einem frühen Ausfall des Flash-Speichers führen. Mit der Aktivierung des erweiterten Schreibfilters werden die Anzahl der Schreibzugriffe auf die Betriebssystem-Partition minimiert und somit die Lebenszeit des Flash-Speichers erhöht.

2. Dokumentieren Sie alle besonderen Einstellungen (z. B. Änderungen der IP-Adressen), die Sie während der Entwicklung der Anwendung vorgenommen haben. Legen Sie sie zusammen mit Ihrem Projekt ab.
3. Sichern Sie Ihre Einstellungen über ein Backup.

1.4.4 CPs und Applikationen für PC-Stationen konfigurieren

1.4.4.1 CPs für PC-Station 1.0

PC-Station

SIMATIC NET PC Software für PC-Kommunikationsprozessoren

SIMATIC NET PC Software für PC-Kommunikationsprozessoren

Für die folgenden Kommunikationsprozessoren wird je nach Betriebssystem die Verwendung der nachfolgend genannten Versionen der SIMATIC NET PC Software empfohlen:

- CP 1613, CP 1613 A2, CP 1623, CP 1628
- CP 5613, CP 5613 A2, CP 5614, CP 5614 A2, CP 5623, CP 5624

Verwenden Sie abhängig vom Betriebssystem folgende Versionen der SIMATIC NET PC Software:

- Windows 8 (Professional, Enterprise), 32 und 64 Bit sowie Windows Server 2012
  DVD "SIMATIC NET PC Software" V12
- Windows 7 (Professional, Enterprise, Ultimate), 32 und 64 Bit sowie Windows Server 2008 R2
  DVD "SIMATIC NET PC Software" V12
- Windows XP, Windows Server 2003 und Windows Server 2008 32 Bit
  CD "SIMATIC NET PC Software" Edition 2008 + SP6, Versionsnummer V7.1.6

Steckplatzregeln für eine PC-Station

Einführung

Komponenten wie CPs, PC-basierende Anwenderprogramme für die Verbindungsprojektierung und OPC-Server können Sie auf Steckplatz (Index) 1 bis 32 stecken. Für PC-Stationen gelten bestimmte Steckplatzregeln.

Stecken eines OPC-Servers

In PC-Stationen können OPC-Server für OPC-Clients gesteckt werden, die auf Variablen entfernter Automatisierungssysteme über projektierte Verbindungen zugreifen:

- In eine PC-Station kann jeweils maximal ein OPC-Server für OPC-Clients gesteckt werden.
- Ist ein OPC-Server in einer PC-Station gesteckt, kann diesem ein CP zugeordnet werden.

Anstelle des OPC-Servers oder zusätzlich zu einem OPC-Server können Sie auch Standard-Applikationen verwenden.

Stecken von Komponenten

Folgende Regeln gelten für die Indexnummerierung und das Stecken von Komponenten in die PC-Station:

- Die Indexnummern 1 bis 32 können vergeben werden.
- Die erste gesteckte Komponente bekommt immer Index 1.
- Weitere gesteckte Komponenten erhalten die nächste freie Indexnummer.
- Beim Löschen einer Komponente bleiben die Indexnummern der anderen Komponenten unverändert. Dadurch können Lücken in der Indexnummerierung entstehen.

- Das Einfügen von Komponenten zwischen bereits vorhandene Komponenten ist erlaubt.
- Für das Einfügen von Komponenten bei vorhandener Belegung gilt:
  - Wenn freie Indexnummern zwischen zwei Komponenten vorhanden sind, wird der erste freie Index vergeben.
  - Wenn keine freien Indexnummern zwischen zwei Komponenten vorhanden sind, wird der Index aller folgenden Komponenten bis zum ersten freien Index um je eine Nummer erhöht.
    Eine Komponente mit dem Index 32 kann auf diese Weise in die Ablage nicht gesteckter Baugruppen verdrängt werden. Auf diese Weise können Sie die PC-Station erweitern, ohne dass Ihnen bei einer Überbelegung Komponenten mit wichtigen Einstellungen verloren gehen.

Stecken von Submodulen

WinLC RTX unterstützt als Submodul

- maximal eine CP 5611/21-Karte oder eine integrierte CP 5611/21-Schnittstelle
- maximal eine PROFINET Schnittstelle (CP 1616, CP 1604 o.ä.)
- maximal vier Submodule insgesamt. Von den vier Submodulen kann es sich um eine beliebige Anzahl von CP 5613-Kommunikationsschnittstellen handeln.

Wenn eine Komponente Submodule hat, gelten folgende Regeln:

- Ein Controller enthält eine feste Anzahl freier Substeckplätze.
- Freie Substeckplätze werden in der grafischen und tabellarischen Gerätesicht angezeigt.
- Submodule können nur auf freie Substeckplätze gesteckt werden.
- Beim Löschen eines Submoduls entsteht ein freier Substeckplatz.
- Substeckplätze belegen keine Indexnummer.

Ethernet-PC-CPs

Subnetzadressierung bei CP 1613, CP 1623 und CP 1628

CP 1613, CP 1623 und CP 1628 sind Kommunikationsbaugruppen mit Mikroprozessor. Zur sicheren Abwicklung von Kommunikationsverbindungen werden diese auf der Baugruppe bearbeitet. Für Diagnosezwecke (SNMP, DCP) wird der Protokollstack in Ihrem PC verwendet. Um beiden Protokollstacks (IE- und NDIS-Zugang) Kommunikation zu den gleichen Partnern zu ermöglichen, wird empfohlen, beide Protokollstacks einer Baugruppe in das gleiche Subnetz zu legen.

Ethernet-PC-CPs ziehen/stecken

Ethernet-PC-CPs ziehen/stecken

Ethernet-PC-Baugruppen, z. B. USB-Netzwerkkarten, müssen bereits bei Systemstart gesteckt und aktiviert sein. Wenn Ethernet-Baugruppen deaktiviert oder bei Systemstart nicht vorhanden sind oder während des Betriebs gezogen und wieder gesteckt werden, zeigt die Funktionalität "Erreichbare Teilnehmer" in STEP 7 nicht alle Teilnehmer an. Dies betrifft auch den Wechsel von Docking-Stationen, wenn die Ethernet-Baugruppe nicht im Notebook sondern in den Docking-Stationen steckt.

Aktivieren Sie deaktivierte Ethernet-Baugruppen über "Systemsteuerung > Netzwerkverbindung" und starten Sie das System erneut. Starten Sie das System außerdem erneut, wenn Ethernet-Baugruppen im laufenden Betrieb gesteckt wurden oder die Docking-Station mit Ethernet-Baugruppe im laufenden Betrieb gewechselt wurde.

OPC-Konfiguration (Ethernet CP)

Bezug

Parametrierung der OPC-Konfiguration beim PC Ethernet-CP in der Parametergruppe "Eigenschaften > Allgemein > OPC-Konfiguration ".

Die Parametergruppe ist vorhanden, wenn in der PC-Station ein OPC-Server konfiguriert ist.

Option "Zugriff über den OPC-Server"

Aktivieren Sie die Baugruppe für den Betrieb über einen im Gerät projektierten OPC-Server.

Option "Automatikbetrieb des PROFINET IO-Controllers"

Indem Sie diese Option wählen, wird der Betriebszustand des PROFINET IO-Controllers automatisch auf OPERATE gesetzt.

Im Betriebszustand OPERATE hat der PROFINET IO-Controller Zugriff auf das angeschlossene PN IO-System.

Die Option ist dann wählbar, wenn die Option "Zugriff über den OPC-Server" gewählt wurde.

- Option einschalten (Default-Einstellung)
  Wenn Sie die Option wählen, dann wird der PROFINET IO-Controller in den Betriebszustand OPERATE gesetzt, sobald ein Zugriff auf ein IO-Device erfolgt. Dieser Zugriff kann beispielsweise über die OPC-Item-Schnittstelle oder die IO-Base-Anwenderprogrammierschnittstelle erfolgen.
  Weitere Aufrufe zur Initialisierung des Betriebszustandes müssen nicht vorausgegangen sein.
- Option ausschalten
  Der PROFINET IO-Controller muss vor einem Datenzugriff in den Betriebszustand OPERATE gesetzt werden. Verwenden Sie dazu entsprechende Initialisierungsaufrufe einer Schnittstelle im Anwenderprogramm.
  Schnittstelle im Anwenderprogramm kann eine OPC-Item-Schnittstelle oder eine IO-Base-Anwenderprogrammierschnittstelle sein.

Ethernet-Schnittstelle

Uhrzeitsynchronisation

Uhrzeitsynchronisation

Hier wird eingestellt, ob der CP Uhrzeittelegramme bearbeiten soll oder nicht.

Je nachdem, wo der Uhrzeitmaster angeordnet sein soll, werden die beiden folgenden Optionen angeboten

- Master (Uhrzeitsender)
  Uhrzeitnachrichten können aus der PC-Station kommen und in das Subnetz (LAN) weitergeleitet werden. Eine PC-Applikation, die die Uhrzeitmasterfunktion ausführt, übergibt die Uhrzeitinformation per Auftrag an den CP.
- Slave (Uhrzeitempfänger)
  Uhrzeitnachrichten können vom Subetz (LAN) kommen und in die Station weitergeleitet werden. Eine PC-Applikation kann mit einem entsprechenden Auftrag die übermittelte Uhrzeit lesen.

Wenn Sie in Ihrer Station mehrere CPs betreiben, sollten Sie den Fluß der Uhrzeittelegramme abhängig vom Uhrzeitmaster bedenken, da Sie über die hier beschriebene Funktion Uhrzeittelegramme von einem Netz in ein anderes Netz einkoppeln können.

Wenn in einer Station mehrere CPs vorhanden sind, die an das selbe Netz angeschlossen sind, darf nur **einer** dieser CPs die Uhrzeitsynchronisations-Nachrichten weiterleiten.

Erweiterte Optionen

Keep-Alive-Verbindungsüberwachung

Bezug

Parametrierung besonderer Einstellungen für die Ethernet-Schnittstelle in der Parametergruppe "PROFINET/Ethernet-Schnittstelle > Erweiterte Optionen > Schnittstellen-Optionen"

Keep-Alive-Verbindungsüberwachung

Wenn der Keep-Alive-Mechanismus aktiviert ist, werden bei Verbindungen über TCP/IP Lebenszeichentelegramme gesendet. Dadurch ist gewährleistet, dass Verbindungen nach dem Ausfall eines Kommunikationspartners beendet und die Verbindungsressourcen freigegeben werden.

Je nach Baugruppentyp werden Ihnen für die Einstellung zwei Möglichkeiten angeboten:

- Verbindungsüberwachung ein-/ausschaltbar mit fester Intervallzeit
  Sie können hier den Mechanismus zur Verbindungsüberwachung ein- und ausschalten. Als Intervallzeit, mit der Lebenszeichentelegramme (Keep Alive) an den Partner einer Kommunikationsverbindung gesendet werden, sind bei der Baugruppe 30 Sekunden fest eingestellt.
  - Keep Alive ausgeschaltet: 0 Sekunden
  - Keep Alive eingeschaltet: Intervallzeit fest auf 30 Sekunden
- Einstellbare Intervallzeit für Überwachungstelegramme
  Sie können hier die Intervallzeit in Sekunden einstellen, mit der Lebenszeichentelegramme an den Kommunikationspartner gesendet werden:
  - Wertebereich: 0 .. 999999
  - Funktion ausgeschaltet: 0
  - Vorbelegung: 30

Hinweise / Empfehlungen:

Beachten Sie, dass der Keep-Alive-Mechanismus dazu führen kann, dass unterlagerte Verbindungen (z. B. eine ISDN-Telefonverbindung) aufrecht erhalten werden, obwohl keine eigentlichen Nutzdaten übertragen werden. Wenn dies unerwünscht ist, dann gehen Sie wie folgt vor:

- Schalten Sie den Überwachungsmechanismus aus (Intervallzeit = 0)
  oder
- Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung nach dem Ausbleiben von Nutzdaten beendet wird, bevor ein Keep-Alive-Telegramm gesendet wird.

Hinweis

Baugruppen anderer Hersteller

Beachten Sie bei Baugruppen von anderen Herstellern deren Angaben zu den Keep-Alive-Einstellungen.

Hinweise zur Verbindungsaufbau-Überwachung

Zusammenhang zur Verbindungsprojektierung

Die hier einstellbare Zeitüberwachung steht wie folgt im Zusammenhang mit der in der Verbindungsprojektierung einstellbaren Zeitüberwachung: gültig ist der kleinere Wert.

In der Verbindungsprojektierung ist die Zeitüberwachung unter "Eigenschaften > Details zum Verbindungsaufbau" projektierbar.

Web

Webserver im PC-CP

Bezug

Projektierung für die Zugriffe auf den Webserver im PC-CP in der Parametergruppe "Webserver"

Option "Webserver auf dieser Baugruppe aktivieren"

Der CP stellt Ihnen für den Zugriff mittels Webbrowser über Port 80 die Funktion eines Webservers zur Verfügung. Hierzu werden bestimmte HTML-Seiten mit CP-Informationen und Diagnosefunktionen in einen Speicherbereich des CP abgelegt.

Aktivieren Sie die Option, um Zugriff auf diese HTML-Seiten zu erhalten.

- Option aktiviert
  Aktivieren Sie das Optionskästchen, damit nach dem Laden der Projektierungsdaten der Webserver des CP gestartet wird. Über einen Web-Browser können Sie dann alle in Web-Seiten vom CP bereitgestellte Informationen aus dem CP auslesen.
- Option deaktiviert
  Bei deaktiviertem Optionskästchen können Sie nur eingeschränkt auf Informationen zugreifen, die der CP in Web-Seiten zur Verfügung stellt. Das Deaktivieren kann bei bestimmten Betriebsfällen geeignet sein, zusätzliche Ressourcen auf dem CP freizuhalten. Folgende Seiten werden nicht angezeigt:
  - Diagnosepuffer
  - Baugruppenzustand
  - Topologie

Automatische Aktualisierung

Bei aktiviertem Optionskästchen werden die Webseiten automatisch in dem unter "Aktualisierungsintervall" festgelegten Zeitabstand aktualisiert.

Von der automatischen Aktualisierung ausgenommen ist die Webseite "Identifikation".

Auswählen der Sprachen

Wählen Sie die Landessprachen für sprachabhängige Texte, die in den CP geladen werden sollen. Die Anzahl wählbarer Sprachen ist CP-abhängig. Sprachabhängig sind z. B. die Texte im Diagnosepuffer.

Beachten Sie außerdem, dass die Sprachen, die Sie hier auswählen im Projekt installiert sein müssen. Die Projektsprachen-Auswahl finden Sie in der Projektnavigation unter "Sprachen & Ressourcen"

Wurden die von Ihnen ausgewählten Sprachen zuvor nicht installiert, dann können vom Webserver nur Texte in Ihrer Standardsprache angezeigt werden.

Einstellmöglichkeiten für den Schutz

Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Zugriffsstufen des CP nutzen.

CPs wie z. B. der CP 1626 bieten verschiedene Zugriffsstufen, um den Zugang zu bestimmten Funktionen einzuschränken.

Die Parametrierung der Zugriffsstufen erfolgt in einer Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Zugriffsstufe geben an, welche Operationen maximal möglich sind, ohne das Passwort dieser Zugriffsstufe zu kennen. Möchten Sie die Funktionen nicht markierter Optionskästchen nutzen, ist die Eingabe eines Passwortes notwendig.

ACHTUNG
Projektierung einer Zugriffsstufe ersetzt nicht den Know-how-Schutz Die Parametrierung von Zugriffsstufen verhindert unrechtmäßige Änderungen an dem CP, indem die Rechte zum Download eingeschränkt werden. Bausteine auf der Memory Card sind jedoch nicht schreib- oder lesegeschützt. Um den Code von Bausteinen auf der Memory Card zu schützen, verwenden Sie den Know-how-Schutz.

Voreingestelltes Verhalten

Voreingestellt ist die Zugriffsstufe "Vollzugriff (kein Schutz)". Jeder Nutzer kann die Hardware-Konfiguration und die Bausteine lesen und verändern. Ein Passwort ist nicht parametriert und wird auch für den Online-Zugriff nicht benötigt.

Die Zugriffsstufen im Einzelnen

Bei einem CP können Sie folgende Zugriffsstufen parametrieren:

- Vollzugriff (kein Schutz): Die Hardware-Konfiguration und die Bausteine können von jedem gelesen und verändert werden.
- Lesezugriff: Mit dieser Zugriffsstufe ist ohne Angabe des Passwortes nur lesender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich, d. h. Sie können ohne Eingabe des Passwortes keine Bausteine und Hardware-Konfiguration in den CP laden. Außerdem sind ohne Passwort schreibende Testfunktionen und Firmware-Updates nicht möglich.
- Kein Zugriff (kompletter Schutz): Wenn der CP komplett geschützt ist, dann ist weder lesender noch schreibender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich.

Durch die Legitimation mit dem Passwort erhalten Sie wieder Vollzugriff auf den CP.

Verhalten eines Passwort-geschützten CP im Betrieb

Der Schutz des CP ist wirksam, nachdem die Einstellungen in den CP geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Falle eines Passwortschutzes werden Sie zur Passworteingabe aufgefordert.

Beispiel: Die Baugruppe wurde mit Lesezugriff parametriert und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird.

Jede Zugriffsstufe lässt auch ohne Eingabe eines Passwortes den uneingeschränkten Zugriff auf bestimmte Funktionen zu, z. B. Identifikation über die Funktion "Erreichbare Teilnehmer".

Zugriffsschutz parametrieren

Im Folgenden erfahren Sie, wie Sie für einen CP eine Zugriffsstufe parametrieren und Passwörter eingeben.

Für einen CP können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Zugriffsstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Zugriffsstufe".

Beispiel:

Sie wählen die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" für den CP und geben für jede der in der Tabelle darüber liegenden Zugriffsstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist der CP komplett geschützt.

Für Nutzer, die eines der parametrierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Vollzugriff (kein Schutz)) wirkt, als wäre der CP ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf den CP.
- Das Passwort in Zeile 2 (Lesezugriff) wirkt, als wäre der CP schreibgeschützt. Trotz Passwort-Kenntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf den CP.
- Das Passwort in Zeile 3 (Kein Zugriff (Kompletter Schutz)) wirkt, als wäre der CP schreib- und lesegeschützt. Nutzer, die dieses Passwort kennen, haben nur lesenden Zugriff auf den CP.

Vorgehen

Gehen Sie folgendermaßen vor, um die Zugriffsstufen für einen CP zu parametrieren:

1. Öffnen Sie die Eigenschaften des CP im Inspektorfenster.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Schutz".
   Eine Tabelle mit den möglichen Zugriffsstufen wird im Inspektorfenster angezeigt.
3. Wählen Sie die gewünschte Zugriffsstufe in der ersten Spalte der Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Schutzstufe zeigen Ihnen, welche Operationen noch möglich sind, ohne das Passwort einzugeben.
4. Wenn Sie eine andere Zugriffsstufe als "Vollzugriff gewählt haben:
   - Vergeben Sie in der Spalte "Passwort" in der ersten Zeile (Vollzugriff) ein Passwort für den Vollzugriff.
   - Wiederholen Sie zum Schutz vor Fehleingaben das gewählte Passwort in der Spalte "Passwort bestätigen".
   - Achten Sie darauf, dass das Passwort ausreichend sicher ist, d. h. dass es kein durch eine Maschine erkennbares Muster besitzt!
   - Die Eingabe eines Passwortes in der ersten Zeile "Vollzugriff (kein Schutz)" ist obligatorisch und ermöglicht dem Passwort-Kenner uneingeschränkten Zugriff auf den CP, unabhängig von der gewählten Schutzstufe.
5. Weisen Sie weiteren Zugriffsstufen nach Bedarf weitere Passwörter zu, falls die gewählte Zugriffsstufe das erlaubt.
6. Laden Sie die Hardware-Konfiguration, damit die Zugriffsstufe wirksam wird.

Ergebnis

Die Hardware-Konfiguration und die Bausteine sind entsprechend der eingestellten Zugriffsstufe vor unberechtigtem Online-Zugriff geschützt. Wenn eine Operation aufgrund der parametrierten Zugriffsstufe nicht ohne Passwort ausgeführt werden kann, wird ein Dialog zur Eingabe eines Passwortes aufgeblendet.

Kundenspezifische Anpassungen für CP 1626 ab V1.0

Allgemein

Bei den "Kundenspezifischen Anpassungen" kann ein Gerätehersteller die SIEMENS-spezifischen Angaben durch beliebige selbstgewählte Angaben ersetzen. Diese Angaben werden auch in die GSDML-Datei exportiert.

PROFINET Vendor ID und PROFINET Device ID

Diese Parameter werden bei der PI (PROFIBUS International) herstellerspezifisch beantragt und können dann hier eingegeben werden.

Artikelnummer

In diesem Eingabefeld können Sie eine beliebige Artikelnummer eingeben.

Produktfamilie, Herstellername, Produktname und Software-Version

Diese Parameter werden zusätzlich in WEB-Oberflächen und im SNMP-Protokoll verwendet und können hier eingegeben werden.

GSD-Bitmap-Datei

In diesem Eingabefeld können Sie den Namen der GSD-Bitmap-Export-Datei festlegen, die Sie durch die Eingabe erstellen. Das Bitmap-Format ist bereits per Default vorgegeben und muss nicht eingegeben werden.

RT- oder IRT-Funktionalität zuordnen

Sie können ab der Version V1.1 zwischen den beiden Optionen "RT" und "IRT" wählen. Die ausgewählte Option wird unabhängig von der Einstellung bei "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller" im Dialog "Betriebsart" mit der GSDML-Datei exportiert.

Kundenspezifische Anpassungen für CP 1604/CP 1616 ab V2.7

Allgemein

Bei den "Kundenspezifischen Anpassungen" kann ein Gerätehersteller die SIEMENS-spezifischen Angaben durch beliebige selbstgewählte Angaben ersetzen. Diese Angaben werden auch in die GSDML-Datei exportiert.

PROFINET Vendor ID und PROFINET Device ID

Diese Parameter werden bei der PI (PROFIBUS International) herstellerspezifisch beantragt und können dann hier eingegeben werden.

Artikelnummer

In diesem Eingabefeld können Sie eine beliebige Artikelnummer eingeben.

Produktfamilie, Herstellername, Produktname und Software-Version

In diesen Eingabefeldern können Sie beliebige Namen eingeben.

GSD-Bitmap-Datei

In diesem Eingabefeld können Sie den Namen der GSD-Bitmap-Export-Datei festlegen, die Sie durch die Eingabe erstellen. Das Bitmap-Format ist bereits per Default vorgegeben und muss nicht eingegeben werden.

RT- oder IRT-Funktionalität zuordnen

Sie können ab der Version V2.8 zwischen den beiden Optionen "RT" und "IRT" wählen. Die ausgewählte Option wird unabhängig von der Einstellung bei "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller" im Dialog "Betriebsart" mit der GSDML-Datei exportiert.

PROFIBUS-PC-CPs

Betrieb von PROFIBUS-CPs in Rechnern mit STEP 7 V5.5

Betrieb von PROFIBUS-CPs in Rechnern mit STEP 7 V5.5

Wenn auf einem Rechner sowohl STEP 7 V5.5 als auch STEP 7 Professional installiert sind und STEP 7 V5.5 deinstalliert wird, dann muss die Setup-Repair-Funktion von STEP 7 Professional ausgeführt werden, bevor die folgenden Kommunikationsprozessoren als Netzwerkadapter beim Firmware-Laden genutzt werden können:

- CP 561x, CP 5611 A2, CP 5612, CP 5613 A2, CP 5614 A2
- CP 5621, CP 5622, CP 5623, CP 5624
- CP 5711

Firmware-Laden bei S7-300/400-PROFIBUS-CPs

Firmware-Laden bei S7-300/400-PROFIBUS-CPs über CP 5612 oder CP 5622

Falls die für das Firmware-Laden erforderliche Schnittstellenparametrierung "FWL" oder "FWL_FAST_LOAD" fehlt, gehen Sie wie folgt vor:

1. Öffnen Sie das Programm "PG/PC-Schnittstelle einstellen". Sie finden das Programm in der Systemsteuerung Ihres Windows-Betriebssystems.
2. Wählen Sie die Schnittstellenparametrierung, abhängig vom Betriebssystem, aus und kopieren Sie diese anschließend:
   - Bei 32-Bit Betriebssystem:
     "CP5612(PROFIBUS)" / "CP5622(PROFIBUS)"
   - Bei 64-Bit-Betriebsystem:
     "CP5612.PROFIBUS.1" / "CP5622.PROFIBUS.1"
3. Geben Sie der Kopie je nach Bedarf eine der folgenden Bezeichnungen:
   - Bei 32-Bit-Betriebsystem:
     CP5612(FWL) / CP5622(FWL)
     CP5612(FWL_FAST_LOAD) / CP5622(FWL_FAST_LOAD)
   - Bei 64-Bit-Betriebsystem:
     "CP5612.FWL.1" / "CP5622.FWL.1"
     "CP5612.FWL_FAST_LOAD.1" / "CP5622.FWL_FAST_LOAD.1"
4. Öffnen Sie die Eigenschaften des Eintrages und nehmen Sie im Register "PROFIBUS" folgende Einstellungen vor:
   - Aktivieren Sie das Optionskästchen bei "PG/PC ist einziger Master am Bus".
   - Geben Sie der Adresse den Wert "0".
   - Geben Sie der Übertragungsgeschwindigkeit den Wert 187,5 kbit/s (bzw. für FWL_FAST_LOAD den Wert 1,5 Mbit/s).
   - Ändern Sie die "Höchste Teilnehmeradresse" auf den Wert 126.
   - Ändern Sie das Profil auf "Universell (DP/FMS)".
5. Beenden Sie das Programm "PG/PC-Schnittstelle einstellen".

Ergebnis:

Die konfigurierte Schnittstellenparametrierung wird erstellt und steht danach im Firmware-Lader zur Verfügung.

LSAP-Konfiguration

Bezug

Parametrierung der LSAP-Konfiguration beim PC PROFIBUS-CP in der Parametergruppe "Eigenschaften > Allgemein > LSAP-Konfiguration".

LSAPs reservieren

Reservieren Sie LSAPs (Local Service Access Point) für andere auf dem CP genutzte Protokolle.

Beispiel FDL-Verbindungen: das FDL-Protokoll verwendet LSAPs, die nicht gleichzeitig von anderen Protokollen verwendet werden dürfen.

Markieren Sie hierzu die zum LSAP gehörende Nummer in der vorliegenden Liste.

FDL-Verbindungen

LSAPs, die bereits in FDL-Verbindungen verwendet wurden, werden als deaktiviert angezeigt.

Blenden Sie den lokalen LSAP und den Partner-LSAP in der Verbindungstabelle ein, um prüfen zu können, welche Verbindung welchen LSAP verwendet.

Standardmäßig werden diese Werte nicht angezeigt. Klappen Sie zum Einblenden das Kontextmenü in der Kopfleiste der Verbindungstabelle auf. Klicken Sie den Menüpunkt "Anzeigen/Ausblenden" und wählen Sie in der Liste "Lokaler LSAP" und "Partner-LSAP" aus.

Die LSAP werden außerdem in den Verbindungseigenschaften unter "Adressdetails" angezeigt. Hier können Sie die verwendeten LSAP bei Bedarf auch ändern.

OPC-Konfiguration (PROFIBUS CP)

Bezug

Parametrierung der OPC-Konfiguration beim PC PROFIBUS-CP in der Parametergruppe "Eigenschaften > Allgemein > OPC-Konfiguration".

Die Parametergruppe ist vorhanden, wenn in der PC-Station ein OPC-Server konfiguriert ist.

Option "Zugriff über den OPC-Server"

Aktivieren Sie die Baugruppe für den Betrieb über einen im Gerät projektierten OPC-Server.

Option "DP-Master automatisch in Operate setzen"

Indem Sie diese Option wählen, startet der DP-Betrieb automatisch mit dem Hochlauf des OPC-Servers.

Aktivitätsüberwachung des CP aktiviert - Option "DP-Zeitüberwachung"

Greift der OPC-Server (aufgrund eines Fehlers) nicht mehr auf die DP-Schnittstelle zu, reagiert die Baugruppe bei eingeschalteter Aktivitätsüberwachung wie folgt: an alle DP-Slaves, die dem OPC-Server zugeordnet sind, werden Daten mit dem Wert 0 (Ersatzwerte) gesendet.

Einstellbarer Parameter:

- Timeout des DP-Watchdog
  Bei eingeschalteter Aktivitätsüberwachung erfolgt die Übertragung der Ersatzwerte, wenn nach der hier eingestellten Zeit kein Zugriff auf die DP-Schnittstelle mehr erfolgt.
  Default: 6000 ms

Option "Diagnoseüberwachung" aktiviert

Wenn die Diagnoseüberwachung aktiviert ist, überwacht der OPC-Server bei Zugriffen auf die Ein-/Ausgänge der DP-Slaves selbständig das Diagnoseflag und wertet bei DP Norm-Slaves Diagnosedaten aus. Werden in den Diagnosedaten Fehler oder Probleme erkannt, wird der Zugriff auf die Ein-/Ausgänge der DP-Slaves für die hier eingestellte Sperrzeit unterbunden; Lese- oder Schreibaufträge werden dann mit Fehler quittiert.

Einstellbarer Parameter:

- Sperrzeit für Read / Write
  Default: 500 ms

Fehlerwartezeit für DP C1 Aufträge

Legen Sie hier für DPC1-Dienste (azyklische Übertragung als DP-Master Klasse 1) eine Fehlerwartezeit fest.

Default: 15000 ms

OPC-Serverfunktionen für DP Master Klasse 2

Sie können hier die OPC-Serverfunktionen für DP Master Klasse 2 aktivieren. Die Einstellungen hier betreffen die Datensatzfunktionen gemäß DP Master Klasse 2.

Es wird eine 2-stufige Option angeboten:

- Datensatzfunktionen aktivieren
  Mit den Datensatzfunktionen können gerätespezifische Informationen ausgelesen oder Parameter geschrieben werden. Mit dieser Option aktivieren Sie den OPC-Server, der diese Funktion unterstützt.
- im PROFIdrive Modus betreiben
  Indem Sie diese Option zusätzlich wählen, wird der PROFIdrive-Bus-Server aktiviert. Damit sind die Datensatzfunktionen nur im speziellen PROFIdrive-Modus nutzbar.

Einstellbarer Parameter:

- Timeout des Verbindungsaufbaus
  Wird die Anforderung zum Aufbau einer Verbindung vom Kommunikationspartner nicht innerhalb der hier eingestellten Zeit beantwortet, so werden OPC-Aufträge mit einer Fehlermeldung abgebrochen.
  Da je nach Aufbau des Netzes der erfolgreiche Verbindungsaufbau längere Zeit in Anspruch nehmen kann, sollte der Verbindungsaufbau-Timeout größer sein als der Auftragstimeout (siehe Fehlerwartezeit für asynchrone Aufträge).
  Default: 15000 ms
- Fehlerwartezeit für asynchrone Aufträge
  Wird ein Lese- oder Schreibauftrag an ein Partnergerät nicht innerhalb der hier eingestellten Zeit beantwortet, so werden die ausstehenden OPC-Aufträge mit einer Fehlermeldung abgebrochen.
  Default: 15000 ms

Konfiguration der LDB-Datei

Bezug

Parametrierung der LDB-Konfiguration beim PC PROFIBUS-CP in der Parametergruppe "Eigenschaften > Allgemein > LDB-Konfiguration".

Hinweis

Version der SIMATIC NET PC Software

Die LDB-Datei kann nur für CPs verwendet werden, die mit der SIMATIC NET PC Software bis zur Version Mai 2000 betrieben werden.

LDB-Datei erzeugen (gilt nur für CPs mit Firmware < V6.0 SP5 bzw. bis SIMATIC NET PC Software Mai 2000)

Die Konfiguration des DP-Mastersystems muss beim Speichern und Übersetzen in eine LDB -Datei (Ladbare Datenbasis) exportiert werden. Diese Datei enthält die DP-Projektinformationen. Sie wird von den DP-Systemanschlüssen für PG/PC wie SOFTNET oder CP 5614 in den PC-Stationen benötigt.

Hinweis

LDB-Datei und Ermittlung der Ein-/Ausgangsadressen der DP-Module

Die Adressen für die Ein-/Ausgangsbereiche der DP-Slaves werden anhand der PROFIBUS-Adresse der DP-Slaves und auf Basis der eingefügten DP-Module bündig in aufsteigender Reihenfolge automatisch ermittelt und in der LDB-Datei abgelegt.

Adressenangaben bzw. Adressenänderungen für die Ein-/Ausgangsadressen der DP-Slaves im hier projektierten DP-Mastersystem bleiben daher ohne Auswirkung auf die in der LDB-Datei abgelegten Adressinformationen.

Vernetzte PC-Konfigurationen

Konfigurationsdialog

Bezug

Sie wählen bei selektierter PC-Station den Menübefehl "Online > PC-Station online konfigurieren".

Mit dem aufgeblendeten Dialog übertragen Sie die gewählte Konfiguration für eine PC-Station oder zeigen deren aktuelle Konfiguration an.

Voraussetzungen

Auf Ihrem Zielrechner und Konfigurier-Rechner muss die DVD "SIMATIC NET PC-Software" installiert sein.

Sofern auf dem Konfigurier-Rechner oder dem Zielrechner mehr als eine Kommunikationsbaugruppe vorhanden ist, müssen Sie zunächst die jeweilige Baugruppe für den Konfiguriervorgang festlegen. Hierzu steht das Programm "SIMATIC Shell" zur Verfügung.

Wenn Sie diese Festlegung für den Konfigurier-Rechner getroffen haben, oder wenn nur eine Kommunikationsbaugruppe vorhanden ist, wird die lokale Netzwerkverbindung bereits angezeigt und ist hier nicht änderbar.

Sofern Sie in Ihrem STEP 7-Projekt den Zielrechner ausgewählt haben und eine eindeutige Zuordnung zu den erreichbaren Rechnern gegeben ist, sind auch die Felder "Erreichbarer Zielrechner" und "Zielrechner" entsprechend vorbelegt und müssen nicht verändert werden.

Hinweis

SIMATIC Shell

Das Programm "SIMATIC Shell" wird zusammen mit den SIMATIC NET PC-Software-Produkten installiert. Beachten Sie die Dokumentation zur SIMATIC NET PC-Software.

Gehen Sie so vor:

1. Wählen Sie aus der Klappliste die Netzwerkverbindung aus, über die der Zielrechner angesprochen werden soll.
2. Betätigen Sie die Schaltfläche "Aktualisieren", um die erreichbaren Rechner neu zu ermitteln.
3. Wählen Sie in der Liste "Erreichbare Rechner" den gewünschten Zielrechner aus oder geben Sie den Namen des Zielrechners direkt in das Feld "Zielrechner" ein. Die Bezeichnung des Zielrechners ist identisch mit dem Computername unter Windows.

4. Wählen Sie nun zwischen folgenden Vorgängen:
   – die aktuelle Konfiguration des gewählten Zielrechners anzeigen: Betätigen Sie hierzu die Schaltfläche "Anzeigen..."
   – die projektierte Konfiguration in den Zielrechner laden: Betätigen Sie hierzu die Schaltfläche "Konfigurieren..."
5. Beachten Sie nach erfolgter Konfiguration eventuelle Meldungen.

Netzwerkverbindung und Zielrechner zuordnen

Wenn Sie die gegebenenfalls vorbelegten Zuordnungen nicht verwenden möchten oder keine Zuordnung vorbelegt ist, gehen Sie zunächst so vor:

1. Deaktivieren Sie gegebenenfalls die Option "Benutze konfigurierten Zielrechner".
2. Wählen Sie aus der Klappliste die Netzwerkverbindung aus, über die der Zielrechner angesprochen werden soll.
3. Betätigen Sie die Schaltfläche "Aktualisieren", um die erreichbaren Rechner neu zu ermitteln.
4. Wählen Sie in der Liste "Erreichbare Rechner" den gewünschten Zielrechner aus. Alternativ geben Sie den Namen des Zielrechners direkt in das Feld "Zielrechner" ein. Die Bezeichnung des Zielrechners ist identisch mit dem Computername unter Windows.

Zielrechner konfigurieren

Gehen Sie weiter so vor:

1. Wählen Sie zwischen folgenden Vorgängen:
   – Die aktuelle Konfiguration des gewählten Zielrechners anzeigen: Betätigen Sie hierzu die Schaltfläche "Anzeigen..."
   – Die projektierte Konfiguration in den Zielrechner laden: Betätigen Sie hierzu die Schaltfläche "Konfigurieren..."
2. Beachten Sie nach erfolgter Konfiguration eventuell auftretende Meldungen.

Tipp:

Sie können die für die Schaltflächen beschriebenen Funktionen auch über die rechte Maustaste erreichen.

Allgemeine Hinweise

Beachten Sie, dass nach dem Start des Konfigurier-Dialoges Änderungen auf dem lokalen Rechner nicht erfasst werden. Gültig sind die Einstellungen oder Konfigurationsdaten, die beim Start erkannt wurden. Betroffen sind folgende Aktionen:

- Das Aktivieren oder Deaktivieren der Netzwerkadapter.
- Änderungen in der STEP 7-Konfiguration der PC-Station.

Dialog "Konfiguration Zielrechner"

In diesem Dialogfeld wird die aktuelle Konfiguration mit Baugruppen und Applikationen angezeigt.

Sie haben hier die Möglichkeit, eine aktuelle Konfiguration mit einer projektierten Konfiguration zu vergleichen.

Station

Angezeigt wird hier der Stationsname.

Der Stationsname kann mit dem Computer-Name des Zielrechners identisch sein, wenn die Zielstation nach der Installation der SIMATIC NET PC Software noch nicht mit einem neuen Namen konfiguriert wurde. Beim Hochlauf wird zunächst der Computer-Name als Stationsname übernommen.

Siehe auch

Statussymbole beim Konfigurieren (Seite 1584)

Konfigurationsdialog (Seite 1580)

Dialog "Konfigurieren: gewählter Zielrechner"

In diesem Dialogfeld wird die aktuell projektierte Konfiguration für den gewählten Zielrechner mit Baugruppen und Applikationen angezeigt.

Die Übertragung zur Erst-Konfiguration oder Konfigurationsänderung können Sie von hier aus anstoßen.

Beim Öffnen des Dialoges wird überprüft, ob eine Übertragung der projektierten Konfiguration auf den Zielrechner plausibel und damit durchführbar ist.

Station

Angezeigt wird hier der Stationsname.

Der Stationsname kann mit dem Computer-Name des Zielrechners identisch sein, wenn die Zielstation nach dem Hochlauf noch nicht mit einem neuen Namen konfiguriert wurde. Beim Hochlauf wird zunächst der Computer-Name als Stationsname übernommen.

Projektierte Konfiguration prüfen und annehmen

Sie können hier die projektierte Konfiguration prüfen, bevor Sie die Datenübertragung endgültig durchführen.

In der Informationszeile unterhalb der Konfigurationsliste wird angezeigt, ob die Konfiguration durchgeführt werden kann. Die Konfiguration setzt eine Übereinstimmung der vorhandenen (physikalischen) Konfiguration mit der projektierten Konfiguration voraus.

Beachten Sie bitte die Symbole, die Sie auf Besonderheiten hinweisen.

Nachdem Sie die Anzeigen überprüft und die Übernahme angestoßen haben (Schaltfläche OK), werden Sie nochmals aufgefordert, die Übernahme zu bestätigen.

Wann die Konfiguration abgelehnt wird

Die Konfiguration kann in folgenden Fällen nicht übertragen werden:

- Projektierte Baugruppen sind in der PC-Station nicht vorhanden.
- Projektierte Baugruppen können nicht eindeutig zugeordnet werden (siehe Zuordnungsregeln weiter unten).
- Die projektierte Konfiguration überschreitet das für die PC-Station zulässige Mengengerüst.
- Es kann keine Verbindung zur PC-Station aufgebaut werden.
  Dies kann daran liegen, dass eine in der PC-Station konfigurierte Firewall den Zugang für den von STEP 7 benutzten Port 102 sperrt. Geben Sie in der PC-Station den Port 102 für TCP frei.

Beachten Sie bitte die Hinweise bzw. Meldungen in den Feldern "Fehler" bzw. "Status"! Sie erhalten beispielsweise Warnungen, wenn die Zuordnung zu einer Änderung der bisherigen Konfiguration führen würde (Beispiel: eine bereits konfigurierte Baugruppe wird auf einen anderen Index verschoben).

Durch Doppelklicken auf den Meldungstext können Sie ein Fenster mit weitergehenden Informationen zur betreffenden Meldung öffnen.

Zuordnungsregeln

Beim Import wird wie folgt geprüft, ob eine Zurodnung der Baugruppen und damit eine Übernahme möglich ist:

- Die vorgegebenen Baugruppen müssen vorhanden sein.
- Wenn mehrere Baugruppen (bespielsweise PROFIBUS-CPs) gefunden werden, wird versucht, eine Zuordnung vorzunehmen:
  - auf Basis des für die Baugruppe vergebenen Parametrierstringes (in STEP 7 vergeben)
  - auf Basis der Adresse (PB-Adresse; MAC-/IP-Adresse)
  - auf Basis der Kompatiblität (Funktionsumfang)

  Hinweis
  Softnet-CPs

  Beachten Sie, dass in der Konfiguration nur jeweils 1 Softnet-CP (PROFIBUS bzw. Ind.Ethernet) vorhanden sein darf!

Siehe auch

Statussymbole beim Konfigurieren (Seite 1584)

Konfigurationsdialog (Seite 1580)

Statussymbole beim Konfigurieren

Spalte"Status"

Beachten Sie jeweils die Zusatzinformationen in der Spalte "Ursachen".

Fehler: Komponente nicht zuordenbar / übernehmbar / nicht installiert

Warnung: Zuordnung wurde angepasst bzw. ist nicht eindeutig.

Informationszeile unterhalb der Konfigurationsliste

In der Informationszeile wird eine Sammelmeldung ausgegeben, die Sie über den Importvorgang informiert:

Konfiguration nicht durchführbar

Konfiguration durchführbar; es liegen aber Warnungen vor: Zuordnung einzelner Komponenten wurde angepasst bzw. ist nicht eindeutig.

Konfiguration durchführbar

Siehe auch

Dialog "Konfigurieren: gewählter Zielrechner" (Seite 1582)

Protokollunabhängige / Netzunabhängige Einstellungen bei PC-CPs

XDB-Konfiguration

Bezug

Parametrierung der XDB-Konfiguration bei einer PC-Station in der Parametergruppe "Eigenschaften > XDB-Konfiguration ".

XDB-Konfiguration

- Kompatibilität - Option "S7RTM ist installiert"
  Wenn Sie eine neue PC-Station konfigurieren, ist das Optionskästchen aktiviert. In diesem Fall wird auf der Ziel-PC-Station ein "Runtime-Stationmanager" (S7RTM) vorausgesetzt. Die Ziel-PC-Station muss mit dem Komponenten-Konfigurator (SIMATIC NET PC Software ab ca. 7/2001) konfiguriert sein.
  Wenn das Optionskästchen deaktiviert ist, werden Konfigurationsdateien erzeugt, die von "älteren" PC-Stationen interpretiert werden können.
- Speicherort der Konfigurationsdatei
  Die Konfiguration mit den Verbindungsbeschreibungen für die SIMATIC PC-Station wird beim Speichern und Übersetzen in einer Konfigurationsdatei (*.XDB) gespeichert. Diese Datei müssen Sie in die entsprechende PC-Station installieren.
  Mit der Schaltfläche "Durchsuchen" können Sie den voreingestellten Pfad für die Konfigurationsdatei ändern.
  Die relative Pfadangabe (.\...) bezieht sich auf das Verzeichnis, in dem das aktuelle Projekt gespeichert wird (<Install-Dir>\S7proj\<Projekt>\XDB). Wenn Sie einen absoluten Pfad eingeben (z. B. C:\...), dann beachten Sie, dass der Zugriff nur so lange funktioniert, wie das Projekt in der aktuellen Umgebung verbleibt.

Zuordnung der Schnittstellenparametrierung

Prüfung der Schnittstellenparametrierung

Indem Sie die PC-Kommunikationskarte aus dem Hardware-Katalog übernehmen und im Baugruppenträger plazieren, wird eine passende Schnittstellenparametrierung vergeben.

Die gültige Parametrierung wird anhand der im Baugruppenkatalog vorhandenen Informationen zum Baugruppentyp (Name der Baugruppe und Anzahl projektierbarer Baugruppen des selben Typs) sowie der bereits projektierten Baugruppen (Zuweisung der Board-Nummer) erkannt.

Wenn mehrere CPs des selben Typs in der Station projektiert werden, werden die Baugruppen-Bezeichnungen durch eine zusätzliche fortlaufende Nummer – die Board-Nummer - ergänzt. Über die angebotene Klappliste können Sie die zulässigen Einträge erkennen und ggf. auswählen.

In besonderen Fällen, wenn Sie z.B. die Baugruppenanordnung identischer Baugruppen in der PC-Station vertauschen, kann es erforderlich sein, die Schnittstellenzuweisung hier

anzupassen; gemeint ist hierbei das Vertauschen der Hardware – nicht das Vertauschen des "virtuellen" Steckplatzes in HW Konfig.

Hinweis

Schnittstellenparametrierung im Zielsystem (PC-Station)

Achten Sie darauf, daß im Zielsystem (PC-Station) dieselbe Schnittstellenparametrierung benutzt wird, wie sie diese hier angegeben haben.. Andernfalls haben Sie dort nach dem Laden der Konfigurationsdatei einen Schnittstellenkonflikt.

Hinweis

Überprüfung auf dem Zielsystem

Verwenden Sie zur Überprüfung auf dem Zielsystem (PC-Station) die Funktion PG/PC-Schnittstelle einstellen. Sie finden diese in der Windows Systemsteuerung. Im Register "Zugriffsweg" sehen Sie, welche Schnittstellen konfiguriert sind und ob ggf. Konflikte bestehen.

1.4.4.2 Applikationen für PC-Station 1.0

Benutzeranwendung OPC-Server

OPC-Server

Protokollspezifische allgemeine Einstellungen

Allgemein

Zykluszeit

Auswirkung der einstellbaren Zykluszeit

Die Zykluszeit beinflusst, wie oft der OPC-Server in seinem Cache die Werte der OPC-Items über einen neuen Kommunikationsauftrag aktualisiert und auf Änderungen überprüft. Die Zykluszeit hat somit Einfluss auf die Performance, auf die Kommunikationslast und die Prozessorlast.

Zu kleine Werte für die Zykluszeit erhöhen unnötig die Prozessorlast!

Zusammenhang zwischen Zykluszeit und Aktualisierungszeit (UpdateRate)

Der über das Anwenderprogramm einstellbare Parameter "Aktualisierungszeit" (UpdateRate) legt das kleinste Zeitintervall für eine Überprüfung der Werte der OPC-Items einer aktiven OPC-Gruppe fest. Frühestens nach Ablauf dieses Zeitintervalls prüft der Server, ob sich der Wert der OPC-Items verändert hat. Auf diese Weise wird die Kommunikationslast begrenzt.

Die vom OPC-Server für SIMATIC NET verwendeten Aktualisierungsraten (Update Rate) sind Vielfache der hier bei der Projektierung festgelegten Zykluszeit. Die kleinste mögliche Aktualisierungsrate (minimale UpdateRate) entspricht der Zykluszeit.

Zusammenhang zwischen den protokollspezifisch eingestellten Zykluszeiten

Da der SIMATIC NET OPC-Server Variablen verschiedener Protokolle gleichzeitig verwenden

kann, ergibt sich die minimale UpdateRate des OPC-Servers als der größte Wert der für die aktiven Protokolle (das sind die, für die Verbindungen projektiert sind) projektierten Zykluszeit.

Beispiel:

Für S7 ist eine Zykluszeit von 100 ms projektiert. Es wird kein anderes Protokoll verwendet. Als minimale Aktualisierungszeit (UpdateRate) wird vom OPC-Server der Wert 100 ms verwendet. Nun wird zusätzlich das DP-Protokoll genutzt, für das eine Zykluszeit von 50 ms projektiert ist. Folge: Der OPC-Server verwendet nun für alle Protokolle als minimale UpdateRate 50 ms.

Anmerkung: Das beschriebene Verhalten resultiert aus den Anforderungen der OPC-Norm. Die OPC-Norm schreibt vor, dass Datenänderungen nicht schneller gemeldet werden dürfen, als dies durch die minimale Aktualisierungszeit (UpdateRate) vorgegeben wird.

tatsächliche Aktualisierungszeit (UpdateRate)

Die im Betrieb erreichbare tatsächliche Aktualisierungszeit (UpdateRate) hängt von verschiedenen Faktoren ab; hierzu gehören:

- unterlagertes Kommunikationsprotokoll
- Kommunikationspartner
- Anzahl und Art der Items

Dadurch kann die erreichbare tatsächliche Aktualisierungszeit (UpdateRate) größer sein, als der hier projektierte Minimalwert.

Ändert sich der Prozesswert mehrfach innerhalb des tatsächlichen Aktualisierungsintervalls, werden nicht alle Wertänderungen an die Applikation weitergegeben.

Anmerkung:

Der Datendurchsatz bei asynchronen und synchronen Aufträgen hängt nicht von den Parametern UpdateRate und der in der Projektierung festgelegten Zykluszeit ab.

Zugriffsschutz

Zugriffsschutz aktivieren

Bezug

Einstellungen zu OPC-Server / Parametergruppe "Eigenschaften > Allgemein"; protokollspezifischer Knoten ist angewählt.

Bedeutung von Zugriffsrechten für OPC-Variablen

Über den OPC-Server können protokollspezifisch die Zugriffsrechte auf einzelne Variablen oder Variablengruppen festgelegt werden. Durch die Angabe von Zugriffsrechten können Sie beispielsweise das Überschreiben einer steuerungsinternen Berechnungsvariablen verhindern.

- Standardrechte
  Für nicht spezifisch benannte OPC-Items können Sie hier einen Standardwert festlegen:
  - ReadWrite: Lesen und Schreiben
  - Read: Lesen
  - Write: Schreiben
  - None: Kein Zugriff
- OPC-Item-spezifische Zugriffsrechte
  Wenn Sie diese Option wählen, können Sie im zugehörenden Eingabefeld für bestimmte OPC-Items explizit Zugriffsrechte vergeben.

Siehe auch

Syntax - OPC-Item-spezifische Zugriffsrechte vergeben (Seite 1588)

Syntax - OPC-Item-spezifische Zugriffsrechte vergeben

Sie können für eine oder mehrere OPC-Items Zugriffsrechte eintragen.

Die für die übrigen OPC-Items definierten Standardrechte werden dadurch nicht berührt.

Die Syntax für OPC-Item-spezifische Zugriffsrechte ist wie folgt definiert:

<OPCItem>=<Rechte>

mit:

- <OPCItem>
  Angabe eines oder mehrerer OPC-Items entsprechend der in der OPC-Dokumentation definierten Syntax. Die Verwendung von Aliasnamen ist möglich.
  Folgende Platzhalter können verwendet werden:
  * eine beliebige Anzahl irrelevanter Zeichen
  ? genau ein Zeichen ist irrelevant
- <Rechte>

ReadWrite	Lesen und Schreiben
Read	Lesen
Write	Schreiben
None	Kein Zugriff

Regeln bei der Auswertung

- Die hier spezifisch vergebenen Rechte haben Vorrang vor den Standardrechten, die Sie unter dem Eingabebereich Zugriffsschutz vergeben haben.
- Die hier spezifisch vergebenen Rechte werden in der hier eingegebenen Reihenfolge ausgewertet. Bei mehrfacher Zuweisung zu einem OPC-Item ist immer die letzte Zuweisung gültig.
  Beispiel:
  DP:[CP 5611]Slave040_AB*=RW
  DP:[CP 5611]Slave040_AB1=R
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=W
  DP:[CP 5611]Slave040_AB1*=R
  Aufgrund dieses Eintrages sind u.a. folgende Zugriffsrechte wirksam:
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=R
  Tipp: Beachten Sie bitte die ausführliche Beschreibung der Syntax in der unten genannten OPC-Dokumentation. Sie finden dort eine Reihe von Beispielen - auch für Anwendungen wie PROFINET IO.
- Führende Nullen in Items und in den hier getroffenen Definitionen für die Zugriffsrechte werden grundsätzlich ignoriert.
  Das folgende Beispiel verdeutlicht den hieraus resultierenden Vorteil:
  Sie projektieren folgendes Zugriffsrecht: Slave004_AB0=R
  Damit haben Sie implizit beispielsweise die folgenden Items mit entsprechendem Zugriffsrecht versehen:
  - Item 1: Slave00004_AB000
  - Item 2: Slave4_AB00,1 (Anmerkung: die Syntax mit ",1" gibt der OPC-Scout vor)

  Für die Verwendung der Platzhalterzeichen "*" und "?" ergibt sich hieraus:
  Verwenden Sie diese Platzhalterzeichen nicht an Stellen, an denen im Item eine führende Null vorangestellt sein kann!
  Beispiel:
  Die Definition Slave0?9M06_AB1=R muss durch folgende beide Definitionen ersetzt werden:
  - Slave?9M06_AB1=R
  - Slave9M06_AB1=R

Weitere Informationen

Beachten Sie die ausführliche Beschreibung der Syntax in der OPC-Dokumentation. Sie finden dort eine Reihe von Beispielen - auch für Anwendungen wie PROFINET IO.

Führende Nullen in Items - Beispiel mit Platzhalterzeichen:

Sie verwenden folgendes Item:

- Slave00004_AB000

Eine beim Vergleich fehlschlagende Definition für das Zugriffsrecht wäre:

- Slave0?4_AB0=R

Begründung:

Beim Vergleich wird im Item auf ein führende "0" erkannt; das Item wird also reduziert auf: Slave4_AB

Aufgrund des Plathalterzeichens "?" wird in der Definition die vorangestellte "0" beim Vergleich nicht mehr als führende "0" erkannt. Die Definition für das Zugriffsrecht wird daher nicht reduziert und der Vergleich schlägt somit fehl.

Abhilfe:

Um die gewünschte Rechtevergabe für dieses Beispiel zu erreichen, müßten Sie also stattdessen die beiden folgenden Definitionen treffen:

- Slave?4_AB0=R
- Slave4_AB0=R

S7-spezifische Einstellungen

OPC-Symbole

Symbole verwenden

Bezug

Einstellungen zu OPC-Server / Parametergruppe "Eigenschaften > S7 > OPC-Symbole".

Symbole verwenden (ab OPC-Server V6.1)

Sie können festlegen, ob aus den für eine CPU projektierten Symboltabellen Symbole in die Systemdaten für den OPC-Server aufgenommen werden sollen. Dies ist Voraussetzung dafür, dass Anwenderapplikationen (OPC-Client) symbolische Variablenzugriffe über den OPC-Server absetzen können.

Es werden die Symboltabellen derjenigen CPUs herangezogen, zu denen für den OPC-Server S7-Verbindungen projektiert sind. Es werden hierbei nur die Symbole berücksichtigt, die sich auf Datenbausteine (DB) beziehen.

Die hier wählbaren Alternativen bedeuten:

- Keine
  In die Systemdaten für den OPC-Server werden keine Symbole aus STEP 7 übernommen. Es gelten die für die OPC-Items unter Zugriffsschutz projektierten Zugriffsrechte.
- Alle
  In die Systemdaten für den OPC-Server werden alle relevanten Symbole aus STEP 7 übernommen.
  Die Zugriffsrechte für alle symbolischen Zugriffe auf die zugeordneten Variablen werden durch diese Auswahl auf Read/Write (RW) gesetzt. Sie haben Vorrang vor den unter dem Bereich Zugriffsschutz vergebenen Zugriffsrechten für die ausgewählten OPC_Items.

- Konfigurieren
  In die Systemdaten für den OPC-Server werden ausgewählte und speziell konfigurierte Symbole aus STEP 7 übernommen. Über die Schaltfläche "Konfigurieren" gelangen Sie in den Dialog "Symbol-Konfiguration", in dem Sie die Symbole auswählen und konfigurieren können.
- Feldelemente zur Laufzeit sichtbar (wählbar ab OPC-Server V8.0)
  Sie können hier entscheiden, ob die Elemente von Arrays zur Laufzeit im OPC-Server sichtbar sein sollen. Betroffen sind die Felder (Arrays), die als Symbole konfiguriert sind. Standardmäßig ist die Option deaktiviert, da die Sichtbarmachung von Feldelementen zu einer hohen zeitlichen Belastung im OPC-Server führen kann. Bei älteren OPC-Server-Versionen (< V8.0) sind die Feldelemente grundsätzlich sichtbar, eine Deaktivierung ist dort nicht möglich.

Hinweis

Konsistenz

Wenn Sie die Option "Alle" wählen, werden beim Speichern und übersetzen die aktuellen CPU Symboltabellen berücksichtigt.

Wenn Sie die Option "Konfigurierte" verwenden, müssen Sie beachten, dass nachträgliche Änderungen in den Symboltabellen folgende Auswirkungen in den Systemdaten für den OPC-Server haben:

- Neu hinzugekommene Symbole werden nicht berücksichtigt
- Ausgewählte Symbole, die nachträglich in der Symboltabelle entfernt wurden, können nicht gespeichert werden.

Hinweis

Bit-Arrays

Symbole, die Daten vom Typ "Bit-Array" enthalten, werden nicht in die Systemdaten des OPC-Servers übernommen!

Hinweis

zur Vorgehensweise

Während der hier beschriebenen Symbol-Übernahme darf kein FB , in dem Symbole verwendet werden, in KOP / AWL / FUP geöffnet sein. Es werden sonst nicht alle Symbole bei der Übernahme eingelesen.

Symbol-Konfiguration

Bezug

Einstellungen zu OPC-Server / Parametergruppe "Eigenschaften > S7 > OPC-Symbole".

Aufruf der Symbolkonfiguration über die Schaltfläche "Symbole konfigurieren".

Übersicht

Sie haben in diesem Dialog die Möglichkeit, diejenigen Symbole auszuwählen und zu konfigurieren, die in die Systemdaten des OPC-Servers übernommen werden sollen.

Relevant für die Vorlage der Symbole sind die CPUs, zu denen für den OPC-Server S7-Verbindungen projektiert sind.

Im Navigationsbereich links werden die entsprechenden Stationen und CPUs sowie ggf. Strukturelemente angezeigt. Im Inhaltsbereich rechts werden die für die angewählte CPU oder das angewählte Strukturelement definierten Symbole angezeigt. Zu jedem Symbol werden der zugewiesene Datentyp sowie die aktuell konfigurierten Eigenschaften angegeben. Diese Eigenschaften sind im unteren Bereich des Dialoges konfigurierbar.

Hinweis

Beachten Sie, dass die Anzeige in der Spalte "Zugriff" auch den Wert "mixed(Default-Zugriffsrecht)" beinhalten kann. Dies wird angezeigt, wenn das Zugriffsrecht in einem untergeordneten Element ungleich des Default-Zugriffrechts ist. Das Default-Zugriffsrecht wird bei neu erstellten untergeordneten Elementen als Standard festgelegt.

Symbole markieren und konfigurieren

1. Wählen Sie zunächst im Navigationsbereich die Symbolgruppe einer CPU oder eines Strukturelementes aus.
2. Markieren Sie dann ein oder mehrere Symbole mit dem Mauscursor im Inhaltsbereich.
3. Konfigurieren Sie die Eigenschaften der gewählten Symbole im unteren Bereich des Dialogs gemäß den nachfolgend beschriebenen Möglichkeiten.

Anzeige "Elemente im aktiven Zweig"

Die Anzeige hat folgende Bedeutung:

- Symbole: Anzahl der insgesamt im gewählten Zweig oder im gewählten Strutkurelement vorhandenen Elemente (ein angewähltes Strukturelement wird hierbei mitgezählt).
- Array-Elemente: Anzahl der Elemente im Array
  - Wertangabe links: Anzahl der als sichtbar markierten Elemente
  - Wertangabe rechts: Gesamtzahl Elemente

Konfigurierbare Eigenschaften

- Option "Sichtbar"
  Indem Sie die Option abwählen, wird das markierte Symbol vom Ladevorgang in den OPC-Server ausgeschlossen.
- Option "Nur sichtbare Symbole laden"
  Wenn Sie die Option wählen, werden ausschließlich die als "sichtbar" markierten Symbole in die Systemdaten des OPC-Servers übernommen und sind damit für den OPC-Client sichtbar. Die Option reduziert die Ladezeit in den OPC-Server dann erheblich, wenn viele Symbole als "nicht sichtbar" markiert sind.

- Zugriffsrecht
  Den ausgewählten Symbolen können Sie über die Klappliste folgende Zugriffsrechte zuordnen:
  - ReadWrite: Lesen und Schreiben
  - Read: Lesen
  - Write: Schreiben
  - None: Kein Zugriff
    Die hier gewählten Zugriffsrechte haben Vorrang vor den für die OPC-Items unter Zugriffsschutz projektierten Zugriffsrechten.

  Hinweis

  Beachten Sie, dass die Einstellung bei "Zugriffsrecht" nicht exportiert und somit auch nicht importiert wird, sondern immer die aktuelle Einstellung angezeigt wird.

- Verbindung
  Wählen Sie in der Klappliste aus den projektierten Verbindungen die S7-Verbindung aus, über die der symbolische Variablenzugriff erfolgen soll.
- Eingabefelder "Edit EU Lo / Hi" - Toleranzbereich für Messwerte festlegen (Grenzwerte EU Lo und EU Hi)
  Legen Sie hier für ein ausgewähltes OPC-Item Grenzwerte für die Ermittlung eines Toleranzbereiches fest. Die Eingabefelder sind unter folgender Voraussetzung aktiv:
  - Sie haben nur ein einzelnes Symbol selektiert;
  - Dem Symbol ist ein elementarer Datentyp zugeordnet, der eine Zahlendarstellung zulässt (INT, DINT, REAL, BYTE, WORD, DWORD).
    Anmerkung:
    Die Eingabe ist nur für OPC-Server ab Version 6.4 möglich.
    Siehe auch Grenzwerte für EU Lo und EU Hi eingeben (Seite 1594)

Anzeige "Zugriffspfad"

Für das im Navigationsbereich angewählte Objekt wird hier der Ablagepfad im aktuellen STEP 7-Projekt angezeigt.

Besonderheiten

- Bit-Arrays werden nicht übernommen
  Symbole, die Daten vom Typ "Bit-Array" enthalten, werden nicht in die Systemdaten des OPC-Servers übernommen. Diese Einschränkung gilt, obwohl diese Symbole zur Auswahl hier angezeigt werden. Diese Einschränkung gilt auch unabhängig davon, wie Sie die Option "Sichtbar" (siehe oben) für diese Symbole einstellen.
- Darstellung der Datentypen "Array of UDT" und "Array of Struct" ist unterschiedlich
  Bei Symbolen, die den zusammengesetzten Datentyp "Array of UDT" oder "Array of Struct" enthalten, wird die Struktur des Datentyps "UDT" oder "Struct" im linken Bereich des Dialoges nur einmal dargestellt. Dass es sich um ein Array handelt, ist aufgrund der Angabe "Array" im Feld "Datentyp"im rechten Bereich erkennbar.
  Es wird nur das erste Array-Element mit seinen Adressen und dadurch die Adress-Zuordnung der einzelnen Elemente des Arrays angezeigt.
  Im Gegensatz zu dieser Darstellungsform werden in anderen Werkzeugen wie beispielsweise dem Symboldatei-Konfigurator die Array-Elemente mehrfach angezeigt. Prinzipiell handelt es sich jedoch um identische Datenstrukturen.
  Beachten Sie, dass die Einstellung für die oben beschriebene Option "Sichtbar" für alle Elemente eines Arrays wirksam ist; also auch für die nicht dargestellten Array-Elemente.

Grenzwerte für EU Lo und EU Hi eingeben

Toleranzbereich für Messwerte eingeben (Obere und untere Grenzwerte EU Lo und EU Hi)

Legen Sie hier für ein ausgewähltes OPC-Item Grenzwerte für die Ermittlung eines Toleranzbereiches fest (zur Bedeutung siehe weiter unten).

Angebbar sind Zahlenwerte als Ganzzahl, Fließkomma oder Exponentialform ("Ingenieurs-Notation")

Hinweis: Als Separator ist das Zeichen "." anstelle des Zeichens "," zu verwenden.

- EU Lo (Engineering Unit Low)
  Untergrenze
- EU Hi (Engineering Unit High)
  Obergrenze

Es muss gelten: EU Lo <= EU Hi

Ein leer gelassenes Eingabefeld wird automatisch mit "0" belegt.

Beachten Sie die vom Datentyp abhängigen Wertebereiche: Wertebereich der Datentypen (Seite 1596)

- Beispiele:
  - EU Lo = 10 EU Hi = 25
  - EU Lo = 4.5 EU Hi = 14.5
  - EU Lo = 1.2e-2 EU Hi = 2.5e-2
- Anmerkungen:
  - die Angabe für den Toleranzereich ("Percent Deadband" = prozentualer Toleranzbereich) erfolgt jeweils für eine Gruppe von OPC-Items.
  - die Eingabe ist nur für OPC-Server ab Version 6.4 möglich.

Bedeutung von unterer Grenzwert - EU Lo und oberer Grenzwert - EU Hi

Für Messwerte können Toleranzbereiche für die Messwert-Erfassung angegeben werden. Innerhalb eines solchen Toleranzereiches ("Percent Deadband" = prozentualer Toleranzbereich) kann sich ein Messwerte ändern, ohne dass die Änderung gemeldet wird.

Hierzu wird gemäß der folgenden Formel aus den angegebenen Parametern ein relativer Wert ermittelt. Dieser Wert besagt, um wieviele Einheiten eine physikalische Größe nach oben oder unten maximal abweichen darf, ohne dass eine Wertänderung gemeldet wird.

A = PD / 100 x (EU HI – EU Lo)
---

mit:

A = Betrag der Abweichung (nach oben oder unten)

PD = Percent Deadband / %

EU Lo = Unterer Grenzwert

EU Hi = Oberer Grenzwert

Beispiel

Sie gehen bei einer Temperaturmesskurve davon aus, dass im Bereich zwischen 30°C und 40°C der für den Messwert typische Toleranzbereich gegeben ist. Als prozentualen Toleranzbereich ("Percent Deadband") legen Sie hierbei 10% fest.

Daher wählen sie folgende Werte:

EU Lo = 30

EU Hi = 40

PD = 10%

Daraus ergibt sich: A = 10 / 100 x (40 – 30) = 1

Eine Temperaturänderung wird also dann als Änderung gemeldet, wenn sie mehr als 1°C über oder unter dem zuvor erfassten Messwert liegt.

Siehe auch

Symbol-Konfiguration (Seite 1591)

Wertebereich der Datentypen

Es gelten folgende Wertebereiche für die verwendeten Datentypen:

BYTE= 0 .. 256

INT= -32768 ... 32767

WORD= 0 ... 65535

DWORD= 0 ... 4294967295.0

DINT= -2147483648 ... 2147483647.0

REAL POS UPPER = 3.402823e+38

REAL POS UNDER = 1.175495e-38

REAL NEG UPPER = -1.175495e-38(negative obere Grenze)

REAL NEG UNDER = -3.402823e+38 (negative untere Grenze)

Siehe auch

Grenzwerte für EU Lo und EU Hi eingeben (Seite 1594)

Syntax - OPC-Item-spezifische Zugriffsrechte vergeben

Sie können für eine oder mehrere OPC-Items Zugriffsrechte eintragen.

Die für die übrigen OPC-Items definierten Standardrechte werden dadurch nicht berührt.

Die Syntax für OPC-Item-spezifische Zugriffsrechte ist wie folgt definiert:

<OPCItem>=<Rechte>

mit:

- <OPCItem>
  Angabe eines oder mehrerer OPC-Items entsprechend der in der OPC-Dokumentation definierten Syntax. Die Verwendung von Aliasnamen ist möglich.
  Folgende Platzhalter können verwendet werden:
  * eine beliebige Anzahl irrelevanter Zeichen
  ? genau ein Zeichen ist irrelevant
- <Rechte>

ReadWrite	Lesen und Schreiben
Read	Lesen
Write	Schreiben
None	Kein Zugriff

Regeln bei der Auswertung

- Die hier spezifisch vergebenen Rechte haben Vorrang vor den Standardrechten, die Sie unter dem Eingabebereich Zugriffsschutz vergeben haben.
- Die hier spezifisch vergebenen Rechte werden in der hier eingegebenen Reihenfolge ausgewertet. Bei mehrfacher Zuweisung zu einem OPC-Item ist immer die letzte Zuweisung gültig.
  Beispiel:
  DP:[CP 5611]Slave040_AB*=RW
  DP:[CP 5611]Slave040_AB1=R
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=W
  DP:[CP 5611]Slave040_AB1*=R
  Aufgrund dieses Eintrages sind u.a. folgende Zugriffsrechte wirksam:
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=R
  Tipp: Beachten Sie bitte die ausführliche Beschreibung der Syntax in der unten genannten OPC-Dokumentation. Sie finden dort eine Reihe von Beispielen - auch für Anwendungen wie PROFINET IO.
- Führende Nullen in Items und in den hier getroffenen Definitionen für die Zugriffsrechte werden grundsätzlich ignoriert.
  Das folgende Beispiel verdeutlicht den hieraus resultierenden Vorteil:
  Sie projektieren folgendes Zugriffsrecht: Slave004_AB0=R
  Damit haben Sie implizit beispielsweise die folgenden Items mit entsprechendem Zugriffsrecht versehen:
  - Item 1: Slave00004_AB000
  - Item 2: Slave4_AB00,1 (Anmerkung: die Syntax mit ",1" gibt der OPC-Scout vor)

  Für die Verwendung der Platzhalterzeichen "*" und "?" ergibt sich hieraus:
  Verwenden Sie diese Platzhalterzeichen nicht an Stellen, an denen im Item eine führende Null vorangestellt sein kann!
  Beispiel:
  Die Definition Slave0?9M06_AB1=R muss durch folgende beide Definitionen ersetzt werden:
  - Slave?9M06_AB1=R
  - Slave9M06_AB1=R

Weitere Informationen

Beachten Sie die ausführliche Beschreibung der Syntax in der OPC-Dokumentation. Sie finden dort eine Reihe von Beispielen - auch für Anwendungen wie PROFINET IO.

OPC-Meldungen

OPC-Meldungen konfigurieren

Bezug

Einstellungen zu OPC-Server / Parametergruppe "Eigenschaften > S7 > OPC-Meldungen".

Sprache / Meldetext-Spracheinstellung

Für OPC-Meldungen legen Sie fest, welche Sprachversionen der Textbibliothek für Meldetexte (Baustein- und Symbolbezogene Meldungen) auf den OPC-Server geladen werden sollen.

Über die beiden Optionskästchen können Sie folgendes einstellen:

- Standardsprache (Festgelegt als Standardsprache für Anzeigegeräte):
  Wenn die für Meldungen eingestellte Standardsprache eine der folgenden STEP 7-Standardsprachen ist: Deutsch (Deutschland), Englisch (USA), Französisch (Frankreich), Spanisch (traditionelle Sortierung), Italienisch (Italien). Ansonsten ist das Optionskästchen deaktiviert.
- Englisch (USA)
  (falls Englisch (USA) als Standardsprache installiert ist)
  Wenn die Standardsprache für Anzeigegeräte ohnehin schon Englisch (USA) ist, dann ist das Optionskästchen deaktiviert.

Sie können eine oder beide Optionen wählen. Falls Sie beide Optionen wählen, wird geprüft, ob im Projekt unter "Sprache für Anzeigegeräte" eine mehrsprachige Textablage gewählt wurde. Ist dies der Fall, werden beim Download in die PC-Station die entsprechenden Textbibliotheken geladen.

Zeitstempel für Meldungen

Wählen Sie unter dieser Parametergruppe aus, von welcher Instanz der Zeitstempel einer Meldung gesetzt werden soll. Indem Sie die Zeitstempelung der CPU überlassen, können Sie Meldungen exakt mit dem Zeitpunkt ihres Auftretens erfassen. Durch die mögliche Angabe eines Zeitoffset können Sie auch Meldungen, die beispielsweise in Anlagen mit unterschiedlichen Zeitzonen auftreten, in einem Erfassungssystem mit einer gemeinsame Bezugszeit registrieren.

- CPU-Zeitstempel
  Meldungen werden mit dem Zeitstempel der meldenden CPU versehen. Ausnahme: StatepathEvents sind lokal erzeugte Meldungen und werden mit der PC-Zeit (UTC) versorgt.
- CPU-Zeit + Offset
  Meldungen werden mit dem Zeitstempel der meldenden CPU versehen; dem Zeitstempel wird ein sekundengenau einstellbarer Zeitversatz (Offset) hinzuaddiert. Ausnahme: StatepathEvents sind lokal erzeugte Meldungen und werden mit der PC-Zeit (UTC) versorgt.
  Zeitversatz-Spannweite: -23:59:59 bis + 23:59:59
- PC-Zeit (UTC)
  Meldungen werden beim Eintreffen mit dem Zeitstempel (UTC) der empfangenden PC-Station (OPC-Server) versehen.

Zuordnungstabelle OPC-Meldungen

Zuordnungstabelle OPC-Meldungen

Bezug

Einstellungen zu OPC-Server / Parametergruppe "Eigenschaften > S7 > Zuordungstabelle OPC-Meldungen".

Bedeutung

Es werden die im Projekt unter "Gemeinsame Daten > Meldeklassen" konfigurierten Meldeklassen in einer Tabelle vorgelegt. Den Meldeklassen ist standardmäßig die OPC-Meldekategorie "OffNormal" zugewiesen. Ordnen Sie bei Bedarf eine andere OPC-Meldekategorie zu.

S7-Verbindungen für den Zugriff auf optimierte Datenbausteine mit S7-1200/S7-1500

Für den Datenzugriff auf optimierte Datenbausteine des OPC-Servers V12 auf nicht typsichere Datenbausteine, Merker, Eingänge, Ausgänge, Zähler und Timer einer SIMATIC S7-1500 und S7-1200 können Sie eine S7-Verbindung zur CPU projektieren.

S7-Verbindungen für den Zugriff auf optimierte Datenbausteine in der S7-1200/S7-1500 werden nur über Industrial Ethernet angeboten.

Voraussetzungen

- STEP 7 Professional ab V12 SP1
- OPC-Server ab V12
- S7-1500 oder S7-1200 ab V4.0

Verbindungsnamen für OPC-Server V12 und S7-Verbindungen

Beim Anlegen einer S7-Verbindung für einen OPC-Server schlägt STEP 7 einen gültigen S7-Verbindungsnamen vor. Sie haben jederzeit die Möglichkeit, den S7-Verbindungsnamen zu ändern. Einige Zeichen und Verbindungsnamen sind systemseitig belegt und können für die Vergabe des Verbindungsnamens nicht verwendet werden. STEP 7 prüft automatisch auf unzulässige Zeichen im Verbindungsnamen.

Verbindungstabelle

In der tabellarischen Netzsicht im Register "Verbindungen" werden die von Ihnen angelegten Verbindungen und deren Parameter in der Verbindungstabelle angezeigt. Sie haben in der Spalte "Partner" die Möglichkeit, den Kommunikationspartner zum OPC-Server V12 zu ändern.

Parameter der besonderen Verbindungseigenschaften

Im Inspektorfenster der ausgewählten einseitigen S7-Verbindung unter "Eigenschaften > Konfiguration > Besondere Verbindungseigenschaften" sind die Optionskästchen "Einseitig" und "Aktiver Verbindungsaufau" aktiviert. Sie können in der Parametergruppe "Besondere Verbindungseigenschaften" keine Änderungen vornehmen.

Adressdetails

Im Inspektorfenster der ausgewählten einseitigen S7-Verbindung unter "Eigenschaften > Konfiguration > Adressdetails" sehen Sie bei den Verbindungsressourcen die TSAPs der Kommunikationspartner. Der lokale TSAP ist folgendermaßen aufgebaut:

SNOPCCxxxxyyyyyy

- SNOPCC = SIMATIC NET OPC Configured connection
- xxxx = Index des OPC-Servers V12
- yyyyyy = Verbindungs-ID der S7-Verbindung

Der Partner-TSAP ist "SIMATIC-ROOT-OTH".

Sie können in der Parametergruppe "Adressdetails" keine Änderungen vornehmen.

OPC-Parameter

Im Inspektorfenster der ausgewählten einseitigen S7-Verbindung unter "Eigenschaften > Konfiguration > OPC" haben Sie folgende Projektierungsmöglichkeiten:

- Aktivierung des optimierten Zugriffs
- Projektierung der Puffergröße für den optimierten Zugriff auf Variablen
- Administration des optimierten Zugriffs auf Variablen über die Eingabe des Zugangspasswords

Puffergröße:

Eine Änderung der voreingestellten Puffergröße ist zweckmäßig in speziellen Anwendungsfällen mit besonders großem Mengengerüst von Datenvariablen auf eine CPU (S7-1500 oder S7-1200 V4.0). Der Zugriff auf nebeneinanderliegende Datenvariablen innerhalb eines Datenbereichs wird zur Leistungsverbesserung optimiert. Optimale Ergebnisse werden mit Puffergrößen erzielt, welche zumindest die meisten nebeneinanderliegenden Datenvariablen umfasst, auf die regelmäßig und häufig zugegriffen wird.

Beispiel: Aus dem Datenbaustein "db10" werden regelmäßig viele Datenvariablen zwischen den Bytes 1000 und 1999 gelesen, in seltenen Fällen auch eine Datenvariable im Byte 30000. Eine Puffergröße von 1000 Bytes ist in diesem Fall ausreichend.

Verbindungsressourcen

Bei projektierten S7-Verbindungen für einen OPC-Server ab V12 zum Zwecke des Datenzugriffs auf optimierte Datenbausteine auf eine PLC S7-1500 oder S7-1200 ab V4.0, werden Ihnen die hierfür benötigten Verbindungsressourcen auf der PLC-Seite angezeigt. Sie finden die Anzeige der benötigten Verbindungsressourcen bei den Eigenschaften der entsprechenden CPU im Inspektorfenster unter "Eigenschaften > Allgemein > Verbindungsressourcen".

Für S7-Verbindungen für den Datenzugriff auf optimierte Datenbausteine eines OPC-Server ab V12 werden PLC-seitig die Resourcen einer solchen S7-Verbindung nicht der Zeile "S7-Kommunikation" zugeordnet, sondern den Zeilen "Sonstige Kommunikation" (S7-1500) bzw. "Freie verfügbare Verbindungen" (S7-1200 ab V4.0). Bei einer projektierten S7-Verbindung für einen OPC Server V12 zu einer S7-1500 oder S7-1200 ab V4.0 werden mehr als eine Verbindungsressource benötigt.

OPC-Server tauschen

Über das Kontextmenü "Gerät tauschen" eines selektierten OPC-Server haben Sie die Möglichkeit, einen OPC-Server V12, welcher Datenzugriff auf optimierte Datenbausteine auf S7-1500 oder S7-1200 V4.0 ermöglicht, gegen einen OPC-Server < V12 und umgekehrt zu tauschen.

Sofern für den zu tauschenden OPC-Server bereits S7-Verbindungen projektiert wurden, erhalten Sie während des Austauschens der OPC-Server einen Hinweis, dass die existierenden S7-Verbindungen gelöscht werden. Diese müssen Sie nach erfolgtem Tausch erneut anlegen.

Anzeige einer einseitigen S7-Verbindung zwischen einem OPC-Server V12 und einer S7-1500 oder S7-1200 ab V4.0

Sobald ein OPC-Server V12 einen erfolgreichen Auftrag für den Datenzugriff auf optimierte Datenbausteine zu einer S7-1500 oder S7-1200 ab V4.0 erteilt hat, wird diese Verbindung in der Online-Verbindungstabelle der entsprechenden PLCs in STEP 7 angezeigt.

Ebenso werden solche Verbindungen auch im Inspektorfenster "Verbindungsinformation" der selektierten Online-CPU angezeigt: hier werden die PLC-seitig verwendeten online-Verbindungsressourcen angezeigt.

Redundante OPC UA-Server

Redundante OPC UA-Server - Grundlagen

Transparente OPC UA-Redundanzserver

Die folgende Darstellung zeigt, wie die OPC UA-Server bei der transparenten OPC UA-Redundanzlösung in eine Redundanzgruppe eingebettet sind und entsprechend projektiert werden.

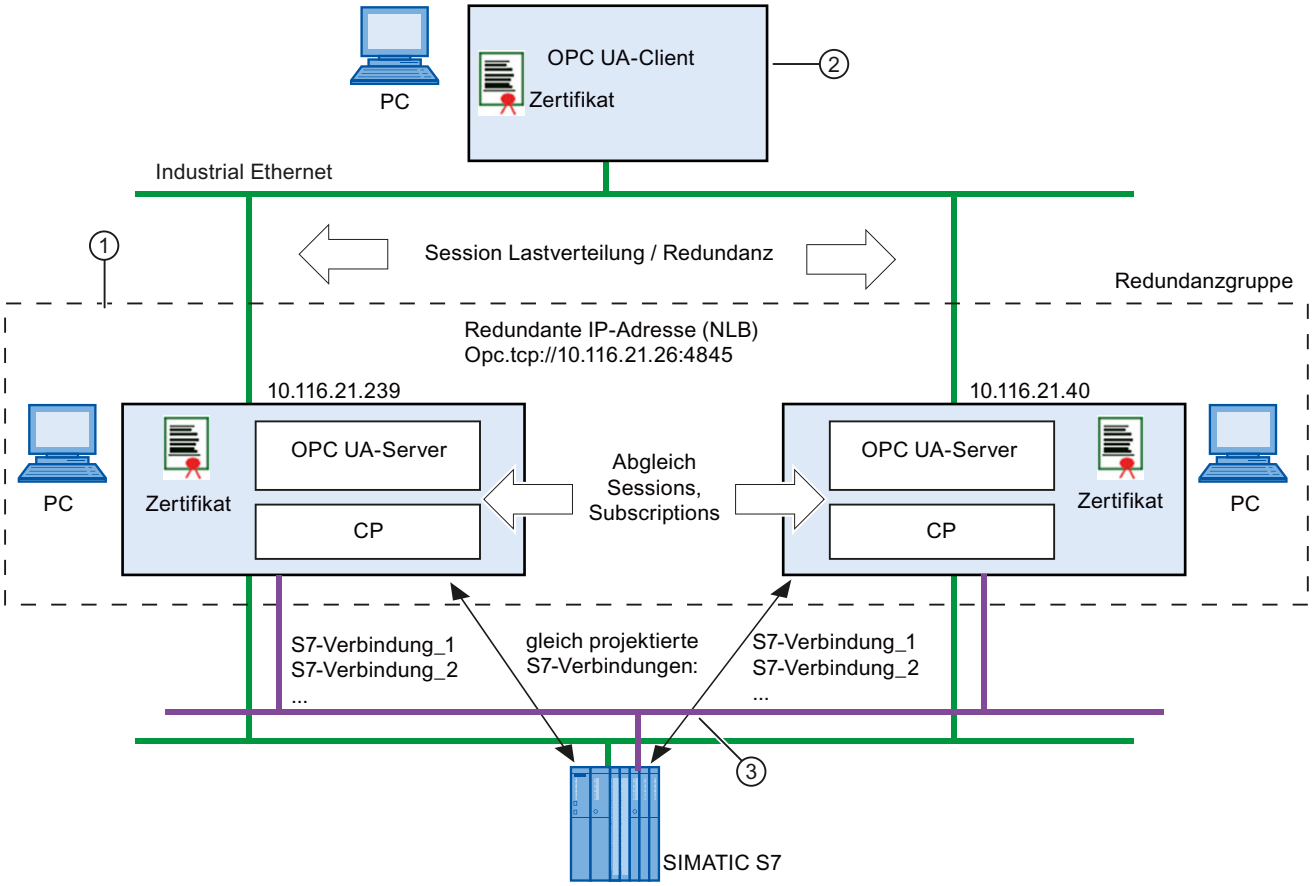

① Sie konfigurieren jeweils eine PC-Station für jeden redundanten OPC UA-Server. Die einem OPC UA-Client zugeordneten redundanten PC-Stationen bilden eine Redundanzgruppe. Diese Zuordnung erfolgt über die einheitliche Projektierung der "IP-Adresse der Redundanzgruppe" bei den OPC UA-Servern.

② Der OPC UA-Client, welcher die Redundanz nutzt, befindet sich in einer übergeordneten PC-Station.

Die übergeordnete PC-Station muss für die Verwendung der OPC UA-Redundanz im STEP 7-Projekt nicht konfiguriert werden.

③ Die Kommunikation zwischen den OPC UA-Servern und den Automatisierungsgeräten erfolgt über mehrfach angelegte, identisch konfigurierte S7-Verbindungen.

Je nach Konfiguration können S7-Verbindungen über Ind.Ethernet oder über PROFIBUS genutzt werden.

SIMATIC NET OPC-Server als S7 OPC UA-Server für transparente OPC UA-Redundanz parametrieren

Die vom übergeordneten OPC UA-Client genutzte transparente Redundanz erreichen Sie, indem Sie bei der Projektierung prinzipiell wie folgt vorgehen:

1. Um PC-Stationen mit redundantem SIMATIC NET OPC UA-Server zu betreiben, legen Sie jeweils eine Komponente vom Typ OPC-Server >= V8.1 an. Für diese OPC-Server aktivieren Sie die S7-UA-Redundanz.
2. Sie übernehmen die virtuelle IP-Adresse. Diese virtuelle IP-Adresse haben Sie zuvor im Netzwerklastenausgleich-Manager (NLB) für beide PC-Stationen festgelegt. Die virtuelle IP-Adresse wird auf die realen IP-Adressen des OPC UA-Servers abgebildet.
3. Sie legen einen Port fest, über den die beiden OPC UA-Server zum Zwecke des Datenabgleiches kommunizieren können;
4. Sie projektieren für die Kommunikation mit den angeschlossenen Automatisierungssystemen für beide OPC UA-Server identische S7-Verbindungen;
5. Den PC-Stationen werden automatisch erzeugte Authentifizierungszertifikate zugewiesen. Bei Bedarf weisen Sie den beiden redundanten PC-Stationen neu erzeugte Authentifizierungszertifikate zu.

Konsistenzprüfung

Beim Speichern und Übersetzen der Konfiguration findet eine Konsistenzprüfung statt. In Bezug auf die OPC UA-Redundanz finden folgende Prüfungen statt:

- Die Server-zu-Server Adressen innerhalb einer Redundanzgruppe müssen eindeutig sein. Ebenso darf innerhalb eines Projektes eine gemeinsame IP-Adresse einer Redundanzgruppe auch nur einmal verwendet werden.
- Jedes Mitglied einer Redundanzgruppe muss über dieselben S7-Verbindungen mit denselben Verbindungsnamen und denselben Kommunikationspartnern verfügen.
- Der Hardwareausbau der PC-Stationen in den Redundanzgruppen muss identisch sein. D.h. für die Redundanzfunktion genutzte Baugruppen müssen in den redundanten PC-Stationen identisch vorhanden sein.
- Die Parameter der S7-Verbindungen müssen identisch sein; hierzu gehören Verbindungssaufbau, Meldungen, Optimierungen, PDU-Größe, Anzahl paralleler Aufträge, Überwachungszeiten.
- Die S7-Einstellungen der OPC UA-Server müssen identisch sein; hierzu gehören Zykluszeit, Zugriffsrechte, S7-Alarme und S7-Symbolik.

Hinweis

Empfehlung: Permanent aufgebaute S7-Verbindungen

Es wird dringend empfohlen, aktiv-permanent aufgebaute S7-Verbindungen auf den OPC UA-Servern zu projektieren. Sie erzielen damit geringere Umschaltzeiten bei einem Serverausfall.

Vorgehensweise / Empfehlung

Die Parametrierungen der redundanten OPC UA-Server müssen zueinander passen. Beim Speichern und Übersetzen der Konfiguration findet eine entsprechende Konsistenzprüfung statt.

Indem Sie in STEP 7 die Möglichkeiten des Kopierens nutzen, reduzieren Sie die Gefahr inkonsistenter Angaben. Sie kopieren entweder die gesamte PC-Station oder nur den OPC-Server. Wenn Sie bereits S7-Verbindungen projektiert haben, werden diese ebenfalls mitkopiert. Sie müssen den Verbindungen in der kopierten PC-Station noch den jeweiligen Verbindungspartner zuweisen.

Siehe auch

S7-UA-Redundanz konfigurieren (Seite 1605)

S7-UA-Redundanz konfigurieren

Bezug

Parametrierung redundanter OPC UA-Server in der Parametergruppe "Eigenschaften > Allgemein > S7-UA-Redundanz

Transparente OPC UA-Redundanz

Sie konfigurieren in dieser Parametergruppe die PC-Station als redundanten OPC UA-Server für den Betrieb mit transparenter OPC UA-Redundanz. Sie verwenden hierzu den SIMATIC NET OPC-Server mit Version 8.1

Einem OPC UA-Client zugeordnete redundante PC-Stationen gehören zu einer Redundanzgruppe.

Gehen Sie in dieser Parametergruppe wie folgt vor:

1. Wählen Sie die Option "Redundanz aktivieren".
2. Tragen Sie die IP-Adresse der Redundanzgruppe ein. Sie haben dazu folgende Möglichkeiten:
   - Sie geben die IP-Adresse direkt ein bzw. übernehmen oder ändern die vorgeschlagene IP-Adresse.
     Wählen Sie diese Möglichkeit dann, wenn eine Redundanzgruppe neu angelegt wird.
   - Sie wählen die Schaltfläche "Zuordnen..."
     Wählen Sie diese Möglichkeit, um den OPC-Server einer bereits angelegten Redundanzgruppe zuzuordnen.

   Hinweis

   Beachten Sie nach der Eingabe oder Zuordnung die Tabelle "IP-Adressen der anderen Mitglieder der Redundanzgruppe" am Ende der Parametergruppe. Sie erkennen, welche OPC-Server der selben Redundanzgruppe zugeordnet sind.

3. Legen Sie den Redundanz-Port für die Kommunikation mit dem OPC UA-Client fest; übernehmen Sie gegebenenfalls die Default-Einstellung.
4. Legen Sie die Parameter für die Server zu Server-Kommunikation fest.
5. Prüfen Sie bei Bedarf das Authentifizierungszertifikat, das dem OPC-Server zugewiesen ist über die Schaltfläche "Anzeigen...".

Parameter / Eingabebereiche / Schaltflächen

- Option "Redundanz aktivieren"
  Indem Sie die Option aktivieren, schalten Sie die Parametriermöglichkeit für die Funktionalität OPC UA-Server frei.
- Schaltfläche "Zuordnen..."
  Sie öffnen den Dialog, in dem die bereits angelegten Redundanzgruppen wählbar sind.
- Parameter "IP-Adresse der Redundanzgruppe"
  Virtuelle IP-Adresse , über die der OPC UA-Client den redundanten OPC UA-Server (die Redundanzgruppe) erreicht.
  Diese virtuelle IP-Adresse legen Sie im Netzwerklastenausgleich-Manager (NLB) für beide PC-Stationen fest. Die virtuelle IP-Adresse wird auf die realen IP-Adressen des OPC UA-Servers abgebildet.
- Parameter "Redundanz-Port"
  Port, über den die Kommunikation mit dem OPC UA-Client erfolgt.
  Default: 4845
- Parameter "Server-zu-Server-Port"
  Port, über den die Kommunikation zwecks Synchronisation mit den OPC UA-Servern in der Redundanzgruppe erfolgt.
  Default: 4846
- Parameter "Server-zu-Server Schnittstelle"
  Wählen Sie hier die Schnittstelle zum Subnetz aus, über die die Synchronisation mit mit den OPC UA-Servern in der Redundanzgruppe erfolgt.
- Eingabebereich "Zertifikat" / Schaltfläche "Anzeigen..."
  STEP 7 erzeugt automatisch ein Zertifikat, das dem OPC UA-Client beim Verbindungsaufbau übermittelt wird. Der OPC UA-Client verwendet dieses zur Authentifizierung bei den darauffolgenden Kommunikationsvorgängen.
  Das Zertifikat wird außerdem zur Übermittlung der Schlüssel zur Datenverschlüsselung verwendet.
  Der private Schlüssel verbleibt beim Server und dient zur Verschlüsselung. Der Client kann dann mit dem zugehörigen öffentlichen Schlüssel, den er vom Server mit dem Zertifikat erhalten hat, die verschlüsselten Nachrichten des Servers entschlüsseln.
  Das aktuell gültige Zertifikat können Sie hier anzeigen und prüfen.
  Anmerkung:
  Die Zertifikatsanzeige erkennt nicht, ob bereits ein Zertifikat angezeigt wird. Mit jeder Betätigung der Schaltfläche "Anzeigen..." wird eine neue Zertifikatsanzeige geöffnet.
- Eingabebereich "Zertifikat" / Schaltfläche "Aktualisieren"
  Sie veranlassen STEP 7, ein neues Zertifikat zu erstellen.

IP-Adressen der anderen Mitglieder der Redundanzgruppe

Anzeige der anderen Mitglieder in der Redundanzgruppe. Zu einer Redundanzgruppe gehören die PC-Stationen, denen dieselbe "IP-Adresse der Redundanzgruppe" zugewiesen ist.

Hinweis

Maximale Anzahl Mitglieder einer Redundanzgruppe

In der aktuellen Version von STEP 7 V12 werden **2** OPC UA-Server innerhalb einer Redundanzgruppe unterstützt.

Siehe auch

Redundante OPC UA-Server - Grundlagen (Seite 1603)

S7-UA-Redundanzgruppe zuordnen

Bezug

Parametrierung redundanter OPC UA-Server in der Parametergruppe "Eigenschaften > Allgemein > S7-UA-Redundanz > S7 UA-Redundanzmaster auswählen

Eine vorhandene S7-UA-Redundanzgruppe zuordnen

Die Tabelle zeigt die im STEP 7-Projekt vorhandenen und wählbaren Redundanzgruppen mit Ihren IP-Adressen.

Sofern Sie dem aktuellen OPC-Server eine neue IP-Adresse zugewiesen haben, erscheint stellvertretend für diese Zuordnung folgender Eintrag:

- "(eigene Redundanzgruppe <IP-Adresse>)"

Gehen Sie wie folgt vor:

1. Markieren Sie die gewünschte Redundanzgruppe;
2. Bestätigen Sie den Dialog über die Schaltfläche "OK".

Damit wird die IP-Adresse der gewählten Redundanzgruppe in die Parametergruppe "S7-UA-Redundanz" übernommen.

Siehe auch

Redundante OPC UA-Server - Grundlagen (Seite 1603)

PLC-Meldungen für OPC-Server

SIMATIC NET OPC Server - Empfehlungen zur Projektierung von CPU-Meldungen

Wenn Sie PLC-Meldungen zu einem OPC-Server einer PC-Station projektieren, dann wird von STEP 7 deren konsistente Projektierung nicht überprüft. Um eine fehlerfreie Funktion der PLC-Meldungen zu erreichen, beachten Sie die folgenden Empfehlungen:

- Projektieren Sie für den Empfang von Meldungen grundsätzlich nur eine alarmfähige S7-Verbindung pro PLC.
- Verwenden Sie pro Projekt durchgängig nur eine Sprache für Meldetexte. Die Verwendung von mehrsprachigen Meldetexten wird nicht unterstützt.
- Verzichten Sie bei allen Meldungen auf die Eingabe von anwenderspezifischen Quellen (Zusatztext 1). Bei Verwendung der Default-Quellen ist die Eindeutigkeit der Meldungen automatisch sichergestellt.

Editor für OPC-Programmmeldungen

Allgemein

Der Editor für OPC-Programmmeldungen steht ab OPC-Server V14 zur Verfügung. Er hilft Ihnen dabei, OPC-Programmmeldungen einer S7-Verbindung zwischen einer PC-Station bzw. OPC-Server und einer S7-1500/S7-1500S so zu konfigurieren, dass die Darstellung in der anwenderspezifischen Ansicht z. B. Ihrer Anlagentopologie entspricht. Dabei können die Bereiche, Quellen und Conditions in der anwenderspezifischen Ansicht hinzugefügt, umbenannt oder gelöscht werden, und können zudem in verschiedenen Sprachen verwaltet werden. Die Struktur und die Namen der OPC-Programmmeldungen in der Standardansicht sind von der Baumstruktur aus dem STEP 7-Projekt vorgegeben. Die Meldetexte der OPC-Programmmeldungen werden in der STEP 7-Projektierung der S7-1500/S7-1500S konfiguriert. Die erstellte Strukturierung der OPC-Programmmeldungen in der anwenderspezifischen Ansicht wird im OPC-UA-Client mit Browsing-Schnittstelle identisch angezeigt, nachdem Sie die Projektierung der PC-Station übersetzt und geladen haben.

Standardansicht

In der Standardansicht werden die verfügbaren OPC-Programmmeldungen angezeigt. Bei hoher Anzahl von OPC-Programmmeldungen ist eine Abstimmung zwischen PLC-Projektierung und OPC-Projektierung zur übersichtlichen Verwaltung empfehlenswert.

Voraussetzung ist eine S7-Verbindung, in deren OPC-Eigenschaften die Optionskästchen "Verbindung dauerhaft aufrecht erhalten" und "Programmeldungen empfangen" aktiviert sind

("Netzsicht" > Register "Verbindungen" > "S7-Verbindung" auswählen > Register "Allgemein" > "OPC" > "Verbindungsaufbau" bzw. "Meldungen").

Hinweis

Wenn Sie nach der Konfiguration im Editor bei der S7-Verbindung die Optionskästchen "Verbindung dauerhaft aufrecht erhalten" bzw. "Programmeldungen empfangen" bei den OPC-Eigenschaften deaktivieren, werden die bis dahin konfigurierten OPC-Programmmeldungen in der Standardansicht solange nicht angezeigt, bis die Optionen wieder aktiviert werden.

Hinweis

Wenn Sie keine OPC-Programmmeldungen in der anwenderspezifischen Ansicht konfigurieren und das STEP 7-Projekt in die PC-Station laden, werden die OPC-Programmmeldungen im OPC-UA-Client identisch zur vorgegebenen Baumstruktur des STEP 7-Projekts angezeigt.

Anwenderspezifische Ansicht

In der anwenderspezifischen Ansicht haben Sie die Möglichkeit, OPC-Programmmeldungen nach Ihren Anforderungen zu strukturieren. Über das Kontextmenü bei "S7OPTAREAS:" definieren Sie die Bereiche und Quellen entsprechend Ihrer Anlagentopologie und fügen die Conditions in die entsprechenden Quellen ein.

Zwischen der Standardansicht und der anwenderspezifischen Ansicht wird über Pfeilsymbole die grafische Zuordnung der Conditions angezeigt. Wenn eine Condition in der anwenderspezifischen Ansicht selektiert wird, wird die Herkunft und das Ziel der Condition über das Pfeilsymbol angezeigt. Je nachdem, welcher Knoten selektiert ist bzw. inwieweit die Baumstrukturen aufgeklappt oder zugeklappt sind, ändert sich die Position des Pfeilsymbols. Bei zugeklappten Baumstrukturen können die Pfeilsymbole zusammengefasst angezeigt werden. Es gibt folgende Möglichkeiten für die grafische Zuordnung der Conditions durch Pfeilsymbole:

- Die Baumstrukturen sind in beiden Ansichten komplett aufgeklappt. Durch Selektion einer Condition wird die direkte 1:1-Zuordnung zwischen Herkunft und Ziel der Condition angezeigt.
- Der Bereich oder die Quelle, in der die Condition (Herkunft) enthalten ist, ist in der Standardansicht zugeklappt. Es wird eine Condition (Ziel) in der anwenderspezifischen Ansicht selektiert. Das Pfeilsymbol zeigt vom Bereich bzw. der Quelle aus der Standardansicht auf die konfigurierte Condition (Ziel) in der anwenderspezifischen Ansicht.
- Der Bereich oder die Quelle, in der die konfigurierte Condition (Ziel) enthalten ist, ist in der anwenderspezifischen Ansicht zugeklappt. Es wird eine Condition (Herkunft) in der Standardansicht selektiert. Das Pfeilsymbol zeigt vom Bereich bzw. der Quelle aus der anwenderspezifischen Ansicht auf die Condition (Herkunft) in der Standardansicht.
- Es wird eine Condition in der Standardansicht selektiert, die nicht in der anwenderspezifischen Ansicht konfiguriert ist. Kein Pfeilsymbol wird angezeigt, da keine Zuordnung erfolgt oder möglich ist.

Sobald Sie eine Condition in der anwenderspezifischen Ansicht konfiguriert haben, ist diese in der Standardansicht ausgegraut. Wenn Sie eine Condition aus der anwenderspezifischen Ansicht löschen, ist diese in der Standardansicht nicht mehr ausgegraut und somit an anderer Stelle in der anwenderspezifischen Ansicht wieder verwendbar.

Hinweis

Wenn Sie OPC-Programmmeldungen aus der S7-1500/S7-1500S löschen, werden diese in der Standardansicht nicht mehr angezeigt. In der anwenderspezifischen Ansicht sind diese OPC-Programmmeldungen rot markiert und sind inkonsistent. Sie können die inkonsistenten OPC-Programmmeldungen nicht mehr verwenden und müssen sie manuell löschen.

Hinweis

Alle in der PLC projektierten OPC-Programmmeldungen, die nicht in der anwenderspezifischen Ansicht konfiguriert werden, werden dennoch beim Laden in die PC-Station mitgeladen. Im OPC-UA-Client sind diese nicht im Editor konfigurierten OPC-Programmmeldungen in einem gesonderten Ordner enthalten. Der Name dieses Ordners entspricht dem in der PLC-Projektierung verwendeten Namen.

Bedeutung der verwendeten Icons

Im Editor für OPC-Programmmeldungen gibt es folgende Icons, die zur Darstellung von Bereichen, Quellen und Conditions in beiden Ansichten verwendet werden:

Bereich:

Jeder Stationsname (z. B. Station1), Gerätename (z. B. Controller1) oder Gruppen (z. B. Gruppe1) wird in der Standardansicht als ein Bereich dargestellt. Die sich daraus aufbauende Baumstruktur wird von der STEP 7-Projektstruktur vorgegeben.

Der Bereich muss zuerst in der anwenderspezifischen Ansicht angelegt werden, da zu Beginn kein Bereich vorhanden ist. Es können mehrere Bereiche definiert werden.

Quelle:
Die Quelle in der Standardansicht besteht aus Stationsname (z. B. Station1), Gerätename (z. B. Controller1) und dem symbolischen Namen des Datenbausteins der PLC-Meldung (z. B. Motor1).

Eine Quelle muss in der anwenderspezifischen Ansicht zu einem Bereich hinzugefügt werden. Es können mehrere Quellen zu einem Bereich hinzugefügt werden.

Condition:
Die Condition in der Standardansicht besteht aus dem Namen der konfigurierten OPC-Programmmeldung. Beispiel: alarm_motor1

Conditions werden per Drag & Drop aus der Standardansicht in die gewünschte Quelle innerhalb der anwenderspezifischen Ansicht gezogen.

Register "Allgemein"

Im Register "Allgemein" werden die Eigenschaften "Condition", "OPC Severity" und "OPC Event-Typ" der selektierten OPC-Programmmeldung angezeigt. Das Eingabefeld "Condition" ist editierbar.

Register "Allgemein" > "OPC-Meldungsinstanzen"

Im Register "Allgemein" > "OPC-Meldungsinstanzen" erhalten Sie eine tabellarische Übersicht über diverse Eigenschaften der OPC-Programmmeldung. Diese Eigenschaften sind an dieser Stelle nur lesbar und können nicht verändert werden.

Folgende Eigenschaften der OPC-Programmmeldungen werden angezeigt:

- Name
  Gibt den Namen der OPC-Programmmeldung an.
- ID
  Gibt die ID der OPC-Programmmeldung an.
- PLC-Name
  Gibt den Namen der S7-1500/S7-1500S an, bei der die OPC-Programmmeldungen projektiert sind.
- Meldetext
  Gibt die Meldetexte an, die bei der S7-1500/S7-1500S projektiert sind.
- Condition
  Gibt den Namen der Condition an.
- OPC Severity
  Gibt die OPC Severity der OPC-Programmmeldung an.
- Bereich
  Gibt den Namen des Bereichs der OPC-Programmmeldung an.

Dialogfeld "Sprachen & Ressourcen"

Sie können im Dialogfeld "Sprachen & Ressourcen" die Editiersprache und die Referenzsprache in allen Sprachen einstellen, die von "STEP 7 Professional (TIA Portal)" unterstützt werden. Sie haben dadurch z. B. die Möglichkeit, in deutscher Sprache zu editieren und zu speichern, und wenn Sie anschließend auf englische Sprache umstellen, können Sie die Namen der OPC-Meldungen in englische Sprache umbenennen und speichern. Je nachdem, welche Sprache Sie nun einstellen, werden Ihnen die Namen der OPC-Programmmeldungen immer in der "richtigen" Sprache angezeigt. Neue, noch nicht übersetzte Namen der OPC-Programmmeldungen in der anwenderspezifischen Ansicht, werden mit den Zeichen "<" und ">" umschlossen. In der Standardansicht werden für alle Sprachen dieselben Namen verwendet, so wie es bei der Projektierung in der PLC definiert wurde.

Sie können über "Projektnavigation" > "Sprachen & Ressourcen" > "Projektsprachen" weitere Projektsprachen hinzufügen.

Bei z. B. "Projektnavigation" > "Sprachen & Ressourcen" > "Projekttexte" können alle Projekttexte, also auch die Meldetexte der Conditions, angezeigt und editiert werden. Im selben Inspektorfenster können Projekttexte auch importiert bzw. selektierte Projekttexte, hier Conditions, exportiert werden.

Hinweis

Für die Übersetzung der Namen der OPC-Programmmeldungen in die verschiedenen Sprachen, wird für die einfachere Handhabung die beschriebene Import- bzw. Export-Funktion empfohlen.

Typische Vorgehensweise bei der ersten Verwendung des Editors

Gehen Sie wie folgt vor:

1. Legen Sie OPC-Programmmeldungen in der S7-1500/S7-1500S für das STEP 7-Projekt an. Bei der S7-Verbindung müssen die OPC-Eigenschaften "Verbindung dauerhaft aufrecht erhalten" und "Programmeldungen empfangen" aktiviert sein.
2. Öffnen Sie den Editor für OPC-Programmmeldungen durch Doppelklicken auf "Editor für OPC-Programmmeldungen" in der Baumstruktur der Projektnavigation beim entsprechenden OPC-Server des STEP 7-Projekts.
3. Wählen Sie in der anwenderspezifischen Ansicht über das Kontextmenü bei "S7OPTAREAS:" den Befehl "Bereichsknoten hinzufügen" aus.
4. Vergeben Sie dem neu angelegten Bereich einen beliebigen Namen (verbotene Zeichen: "\" und Steuerzeichen). Sie können dem Bereich, entsprechend Ihrer Anlagentopologie, weitere Bereiche unterordnen.
5. Wählen Sie über das Kontextmenü bei dem entsprechenden Bereich den Befehl "Quelle hinzufügen" aus.
6. Vergeben Sie der Quelle einen beliebigen Namen (verbotene Zeichen: Steuerzeichen).
7. Ziehen Sie per Drag & Drop die Conditions aus der Standardansicht an die gewünschte Stelle in der anwenderspezifischen Ansicht.
8. Vergeben Sie der Condition einen beliebigen Namen (verbotene Zeichen: Steuerzeichen).

9. Legen Sie gegebenenfalls im Dialogfeld "Sprachen & Ressourcen" die Editier- und Referenzsprache fest. Achten Sie dabei darauf, dass die nicht übersetzten Namen in spitzen Klammern (<>) übersetzt werden.
10. Speichern und übersetzen bzw. laden Sie die neue Konfiguration in die PC-Station und in die PLC.
11. Öffnen Sie einen OPC-UA-Client, der eine Browsing-Schnittstelle hat.
12. Aktivieren und Beobachten Sie die konfigurierten OPC-Programmmeldungen.

Hinweis

Wenn Sie einen bereits angelegten Knoten selektieren, können Sie die Tasten "x", "+" und "-" im Ziffernblock verwenden, um das Öffnen oder Schließen der untergeordneten Baumstruktur zu beschleunigen.

- Multiplikationstaste "x": Öffnet die komplette untergeordnete Baumstruktur des selektierten Knotens.
- Additionstaste "+": Öffnet nur den aktuell selektierten Knoten.
- Subtraktionstaste "-": Schließt die komplette untergeordnete Baumstruktur des selektierten Knotens.

Dies funktioniert sowohl in der Standardansicht als auch in der anwenderspezifischen Ansicht.

Typische Vorgehensweise bei weiteren Verwendungen des Editors

1. Öffnen Sie den Editor für OPC-Programmmeldungen durch Doppelklicken auf "Editor für OPC-Programmmeldungen" in der Baumstruktur der Projektnavigation beim entsprechenden OPC-Server des STEP 7-Projekts.
2. Analysieren Sie die Ist-Situation der vorhandenen Konfiguration.
3. Entscheiden Sie, was Sie umkonfigurieren möchten.
4. Nehmen Sie die Änderungen entsprechend vor.
5. Speichern und übersetzen bzw. laden Sie die neue Konfiguration in die PC-Station und in die PLC.
6. Öffnen Sie einen OPC-UA-Client, der eine Browsing-Schnittstelle hat.
7. Aktivieren und Beobachten Sie die konfigurierten OPC-Programmmeldungen.

Konsistenzprüfung

Die Konfiguration im Editor für OPC-Programmmeldungen wird auf Konsistenz geprüft, bevor sie auf die PC-Station geladen wird. Dabei können folgende Kompilierfehler bzw. -Warnungen ausgelöst werden:

Kompilierfehler bzw. -Warnung	Behebung
Fehler: Die OPC-Programmmeldungen der angegebenen PLC sind über mehrere Verbindungen angebunden.	Die OPC-Programmmeldungen dürfen nur über eine S7-Verbindung angebunden sein.
Fehler: Die Kombination von der angegebenen Quelle und Condition ist in einer Sprache nicht eindeutig.	Die Kombination von Quelle und Condition muss eindeutig konfiguriert sein.

Kompilierfehler bzw. -Warnung	Behebung
Fehler: Die OPC-Programmmeldung im angegebenen Pfad enthält ungültige Zeichen in der angegebenen Sprache.	Es dürfen nur erlaubte Zeichen verwendet werden.
Fehler: Die OPC-Programmmeldung im angegebenen Pfad enthält mindestens ein Bereichsknoten, eine Quelle oder eine Condition mit nicht sichtbaren ANSI Code-Zeichen in der angegebenen Sprache.	Es dürfen nur erlaubte Zeichen verwendet werden.
Fehler: Für die OPC-Programmmeldung im angegebenen Pfad ist mindestens ein Bereichsknoten, eine Quelle oder eine Condition nicht in die angegebene Sprache übersetzt.	Für die verwendeten Sprachen müssen alle Übersetzungen vorhanden sein.
Fehler: Die OPC-Programmmeldung im angegebenen Pfad ist nicht mehr gültig, da keine S7-Verbindung mit gültigen Alarmeigenschaften zu der angegebenen PLC existiert.	Es ist eine entsprechende S7-Verbindung anzulegen oder die genannte OPC-Programmmeldung muss in der anwenderspezifischen Ansicht gelöscht werden.
Fehler: Die OPC-Programmmeldung im angegebenen Pfad ist nicht mehr gültig, da die zugehörige PLC-Meldungsinstanz nicht mehr existiert.	Die genannte OPC-Programmmeldung muss in der anwenderspezifischen Ansicht gelöscht werden.
Warnung: Der angegebene Meldetext ist in der angegebenen Sprache nicht übersetzt.	Der vorgegebene Text muss in der angegebenen Sprache angepasst werden.
Warnung: Die angegebene OPC-Programmmeldung im angegebenen Pfad ist nicht in der anwenderspezifischer Ansicht konfiguriert, aber der OPC-Server verwendet mindestens eine OPC-Programmmeldung, die in der anwenderspezifischen Ansicht konfiguriert wurde.	Gegebenenfalls die in der Warnung angegebene OPC-Programmmeldung in der anwenderspezifischen Ansicht konfigurieren.

Hinweis

Angezeigte Kompilierfehler müssen behoben werden, bevor Sie die Konfiguration in die PC-Station laden können.

Kompilier-Warnungen können gegebenenfalls ignoriert werden, d. h. die Konfiguration kann ohne Behebung der Warnung in die PC-Station geladen werden.

1.4.5 Konfigurationen für Punkt-zu-Punkt-Kopplungen (S7-1500)

1.4.5.1 Konfigurationen für Punkt-zu-Punkt-Kopplungen

Einleitung

Übersicht der Kommunikationsmodule

Automatisierungssysteme umfassen verschiedenste Komponenten. Dazu gehören auch Kommunikationsmodule. Eine einfache Möglichkeit des Datenaustausches bietet die serielle Kommunikation über Punkt-zu-Punkt-Verbindungen.

Durch die Einstellung der Kommunikationsparameter auf unterer Ebene des ISO-Schichtenmodells (siehe Kapitel Übertragungssicherheit (Seite 1620)) ist eine Anpassung an unterschiedlichste Kommunikationspartner möglich.

Die Kommunikation über Punkt-zu-Punkt-Kopplung bei S7-1500, ET 200MP und ET 200SP erfolgt ausschließlich über Kommunikationsmodule (CM) mit seriellen Schnittstellen.

SIMATIC S7 biete eine Reihe von Modulen, die für diesen Einsatzzweck die physikalische Schnittstelle und grundlegende Protokollmechanismen zur Verfügung stellen.

- RS232: Eine Schnittstelle, die durch zusätzliche Begleitsignale die Kommunikation zwischen den Partnern koordinieren kann.
- RS422/RS485: Eine Schnittstelle, die durch die Verwendung von Differenzspannungen als Übertragungstechnik längere Leitungen zulässt und durch eine Busphysik (RS485) auch Strukturen mit mehr als 2 Teilnehmern ermöglicht.

Um Daten von der CPU zu den jeweiligen Modulen zu übertragen, stehen Anweisungen zur Verfügung, die die Koordination zwischen CPU und CM (Communication module) übernehmen. Sie teilen dem Anwenderprogramm eine erfolgreiche Übertragung oder den Empfang neuer Daten mit (In Systemen ohne eine SIMATIC CPU muss der Anwender die Funktion dieser Anweisungen selbst programmieren).

Funktion und Einsatz der PtP-Kommunikationsmodule sind in diesem Funktionshandbuch beschrieben.

Übersicht über die Komponenten und Bestellnummern

Tabellarische Übersicht über die Kommunikationsmodule und deren Einsetzbarkeit

Kommunikationsmodul	S7-1500	ET 200MP	ET 200SP	Artikelnummer
CM PtP RS232 BA [1]	X	X	-	6ES7540-1AD00-0AA0
CM PtP RS422/485 BA	X	X	-	6ES7540-1AB00-0AA0
CM PtP RS232 HF [2]	X	X	-	6ES7541-1AD00-0AB0
CM PtP RS422/485 HF	X	X	-	6ES7541-1AB00-0AB0
CM PtP (ET 200SP)	-	-	X	6ES7137-6AA00-0BA0

[1] BA = Basic

[2] HF = High Feature

Übersicht der Komponenten und Schnittstellen

Tabellarische Übersicht über die Kommunikationsmodule und ihre Funktionen.

Kommunikationsmodul	Schnittstelle	Protokolle					Anschlusstechnik	
		Freeport	3964(R)	Modbus Master	Modbus Slave	USS-Master	D-Sub 9-polig	D-Sub 15-polig
CM PtP RS232 BA	RS232	X	X	-	-	X	X	-
CM PtP RS422/485 BA	RS422	X	X	-	-	X	-	X
	RS485	X	-	-	-	X	-	X
CM PtP RS232 HF	RS232	X	X	X	X	X	X	-
CM PtP RS422/485 HF	RS422	X	X	X	X	X	-	X
	RS485	X	-	X	X	X	-	X

Kommunikationsmodul	Schnittstelle	Protokolle					Anschlusstechnik
CM PtP (ET 200SP)	RS232	X	X	X	X	X	ET 200SP BaseUnit 1)
	RS422 2)	X	X	X	X	X	
	RS485	X	-	X	X	X	

1) BaseUnit mit Klemmen statt D-Sub; Belegung abhängig von der Übertragungsphysik

2) Das Kommunikationsmodul CM PtP kann im RS422-Betrieb auch für Mehrpunkt-Kopplung eingesetzt werden

Übersicht der Komponenten und Datenübertragungsraten

Die Kommunikationsmodule können mit unterschiedlichen Datenübertragungsraten Daten senden und empfangen. Die Zuordnung zu den einzelnen Kommunikationsmodulen finden Sie in der folgenden Tabelle.

Kommunikationsmodul	Datenübertragungsrate in bit/s										
	300	600	1200	2400	4800	9600	19200	38400	57600	76800	115200
CM PtP RS232 BA	X	X	X	X	X	X	X	-	-	-	-
CM PtP RS422/485 BA	X	X	X	X	X	X	X	-	-	-	-
CM PtP RS232 HF	X	X	X	X	X	X	X	X	X	X	X
CM PtP RS422/485 HF	X	X	X	X	X	X	X	X	X	X	X
CM PtP (ET 200SP)	X	X	X	X	X	X	X	X	X	X	X

Übersicht der Komponenten und Empfangspuffergrößen

Jedes Kommunikationsmodul verfügt über einen Puffer, um empfangene Telegramme zwischen zu speichern. Die Zuordnung der maximalen Größe eines einzelnen Telegramms, sowie die Größe des Speichers zu den einzelnen Kommunikationsmodulen finden Sie in der folgenden Tabelle.

Baugruppe	Empfangspuffergröße kByte	Max. Telegrammlänge kByte	Pufferbare Telegramme
CM PtP RS232 BA	2	1	255
CM PtP RS422/485 BA	2	1	255
CM PtP RS232 HF	8	4	255
CM PtP RS422/485 HF	8	4	255
CM PtP (ET 200SP)	4	2	255

Begleitsignale und Datenflusskontrolle

- Software-Datenflusskontrolle mit XON/XOFF
  Die Datenflusskontrolle mit XON/XOFF ist beim Protokoll Freeport über die Schnittstellen RS232 und RS422 möglich.
- Hardware-Datenflusskontrolle mit RTS/CTS
  Die Datenflusskontrolle mit RTS/CTS ist beim Protokoll Freeport über die Schnittstelle RS232 möglich.
- Automatische Bedienung der Begleitsignale
  Die Bedienung der RS232-Begleitsignale ist bei den Protokollen Freeport, Modbus-Master und Modbus-Slave über die Schnittstelle RS232 möglich. (Nur verfügbar, wenn Hardware-Datenflusskontrolle aktiviert ist.)

Protokolle der Kommunikationsmodule

Abhängig von den eingesetzten Kommunikationsmodulen können Sie eine Kommunikationsverbindung mit unterschiedlichen Protokollen aufbauen:

- Freeport: Übertragung von ASCII-Zeichenketten ohne festgelegtes Protokoll-Format
- 3964(R): Kommunikation zwischen speicherprogrammierbaren Steuerungen (Master/Master-Kommunikation)
- Modbus RTU: Kommunikation zwischen speicherprogrammierbaren Steuerungen (Master/Slave-Kommunikation). Das Kommunikationsmodul kann sowohl Master als auch Slave sein.
- USS: Kommunikation zwischen einer speicherprogrammierbaren Steuerung und einem Antrieb (Master/Slave-Kommunikation). Die Kommunikation ist auf die Anforderungen in der Antriebstechnologie zugeschnitten. Das Kommunikationsmodul kann nur Master sein.

Übersicht der Bearbeitungsschritte

Punkt-zu-Punkt-Kopplung

Zum Austausch von Daten zwischen zwei oder mehr Kommunikationspartnern stehen unterschiedliche Möglichkeiten der Vernetzung zur Verfügung. Die Punkt-zu-Punkt-Kopplung zwischen zwei Kommunikationspartnern ist der einfachste Fall des Informationsaustausches.

Das Kommunikationsmodul (CM) bildet die Schnittstelle zwischen einer Speicherprogrammierbaren Steuerung und einem Kommunikationspartner. Die Übertragung der Daten erfolgt bei der Punkt-zu-Punkt-Kopplung mit dem Kommunikationsmodul seriell.

Konfigurieren/Parametrieren

Das Konfigurieren des Kommunikationsmoduls umfasst die Anordnung des Kommunikationsmoduls in der Gerätekonfiguration von STEP 7 (TIA Portal) sowie die Einstellungen der spezifischen Parameter der Protokolle im Eigenschaftendialog des Kommunikationsmoduls (statische Konfiguration).

Programmieren

Die Programmierung umfasst die programmtechnische Anbindung des Kommunikationsmoduls an die zugehörige CPU über das Anwenderprogramm. Die Programmierung des Kommunikationsmoduls nehmen Sie mit STEP 7 (TIA Portal) vor.

Die Kommunikation zwischen CPU, Kommunikationsmodul und einem Kommunikationspartner erfolgt über Anweisungen. Für das Automatisierungssysteme S7-1500 und S7-1200 stehen Ihnen eine Reihe von Anweisungen zur Verfügung mit deren Hilfe Sie im Anwenderprogramm die Kommunikation anstoßen und steuern sowie die Konfiguration zur Laufzeit beeinflussen können (dynamische Konfiguration).

Informationen hierzu finden Sie unter Übersicht der Anweisungen (Seite 1618) und in der Online-Hilfe von STEP 7 (TIA Portal).

Übersicht der Anweisungen

Hinweis

CPU-Mengengerüste

Die Punkt-zu-Punkt Anweisungen kommunizieren mit den Kommunikationsmodulen durch Lesen oder Schreiben von Datensätzen.
Beim Einsatz der Anweisungen sind daher das Mengengerüst der jeweiligen CPU für das Lesen und Schreiben von Datensätzen zu beachten.
Wenn mehrere Anweisungen auf einer CPU gleichzeitig Datensätze lesen oder schreiben sollen, so müssen diese Anweisungen evtl. durch das Anwenderprogramm zeitlich versetzt aufgerufen werden.

Übersicht über die Anweisungen

Auf dem Kommunikationsmodul erfolgt die Umsetzung der Übertragungsprotokolle. Über das Protokoll wird die Schnittstelle des Kommunikationsmoduls an die Schnittstelle des Kommunikationspartners angepasst.

Die Kommunikation zwischen CPU, Kommunikationsmodul und einem Kommunikationspartner erfolgt über spezielle Anweisungen und die Protokolle, die die entsprechenden Kommunikationsmodule unterstützen.

Die Anweisungen bilden die Softwareschnittstelle zwischen der CPU und dem Kommunikationsmodul. Sie müssen zyklisch aus dem Anwenderprogramm aufgerufen werden. Eine Datenübertragung erfolgt über mehrere Zyklen.

Die Anweisungen sind Bestandteil von STEP 7 (TIA Portal). Sie finden die Anweisungen in der Task Card "Anweisungen" unter Kommunikation > Kommunikationsprozessor. Sie gelten für alle genannten Kommunikationsmodule, sofern diese die gewünschte Funktion unterstützen.

Tabelle 1-94 Anweisungen für PtP

Anweisung	Bedeutung
Send_P2P	Die Anweisung Send_P2P ermöglicht Ihnen, Daten an einen Kommunikationspartner zu senden.
Receive_P2P	Die Anweisung Receive_P2P ermöglicht Ihnen, Daten von einem Kommunikationspartner zu empfangen.
Receive_Reset	Die Anweisung Receive_Reset ermöglicht Ihnen, den Empfangspuffer des Kommunikationsmoduls zu löschen.
Port_Config	Die Anweisung Port_Config ermöglicht Ihnen, grundlegende Schnittstellenparameter dynamisch zu parametrieren.
Send_Config	Die Anweisung Send_Config (Sendeparametrierung) ermöglicht Ihnen, serielle Sendeparameter eines Ports dynamisch zu parametrieren.
Receive_Config	Die Anweisung Receive_Config (Empfangsparametrierung) ermöglicht Ihnen, serielle Empfangsparameter eines Ports dynamsich zu parametrieren.
P3964_Config	Die Anweisung P3964_Config (Protokollprojektierung) ermöglicht Ihnen, die Parameter der Prozedur 3964(R) dynamisch zu projektieren.
Signal_Get	Die Anweisung Signal_Get ermöglicht Ihnen, die RS232 Begleitsignale zu lesen.
Signal_Set	Die Anweisung Signal_Set ermöglicht Ihnen, die RS232 Begleitsignale zu setzen.
Get_Features	Die Anweisung Get_Features ermöglicht Ihnen, vom Kommunikationsmodul unterstützte erweiterte Funktionen zu lesen.
Set_Features	Die Anweisung Set_Features ermöglicht Ihnen, vom Kommunikationsmodul unterstützte erweiterte Funktionen zu aktivieren.

Tabelle 1-95 Anweisungen für Modbus

Anweisung	Bedeutung
Modbus_Master	Die Anweisung Modbus_Master ermöglicht Ihnen, als Modbus-Master über den PtP-Port zu kommunizieren.
Modbus_Slave	Die Anweisung Modbus_Slave ermöglicht Ihnen, als Modbus-Slave über den PtP-Port zu kommunizieren.
Modbus_Comm_Load	Die Anweisung Modbus_Comm_Load ermöglicht Ihnen, den Port des Kommunikationsmoduls für Modbus-RTU konfigurieren.

Tabelle 1-96 Anweisungen für USS

Anweisung	Bedeutung
USS_Port_Scan	Die Anweisung USS_Port_Scan ermöglicht Ihnen die Kommunikation über das USS-Netzwerk.
USS_Drive_Control	Die Anweisung USS_Drive_Control ermöglicht Ihnen, Daten mit dem Antrieb auszutauschen.
USS_Read_Param	Die Anweisung USS_Read_Param ermöglicht Ihnen, Parameter aus dem Antrieb auszulesen.
USS_Write_Param	Die Anweisung USS_Write_Param ermöglicht Ihnen, Parameter im Antrieb zu ändern.

Grundlagen der seriellen Kommunikation

Serielle Datenübertragung

Bei der seriellen Datenübertragung werden die einzelnen Bits eines Zeichens einer zu übertragenden Information in einer festgelegten Reihenfolge nacheinander übertragen.

Bidirektionaler Datenverkehr - Betriebsart

Beim bidirektionalen Datenverkehr unterscheidet man beim Kommunikationsmodul zwei Betriebsarten:

- Halbduplexbetrieb
  Die Daten werden zwischen einem oder mehreren Kommunikationspartnern abwechselnd in beide Richtungen übertragen. Halbduplexbetrieb bedeutet, dass zu einem Zeitpunkt ein Kommunikationspartner sendet und der andere Kommunikationspartner empfängt. Dabei wird eine Leitung abwechselnd für Senden bzw. Empfangen verwendet.
- Vollduplexbetrieb
  Die Daten werden zwischen einem oder mehreren Kommunikationspartnern gleichzeitig ausgetauscht, es kann zu einem Zeitpunkt also sowohl gesendet als auch empfangen werden. Dabei wird eine Leitung für Senden und eine Leitung für Empfangen benötigt.

Asynchrone Datenübertragung

Bei den Kommunikationsmodulen erfolgt die serielle Datenübertragung asynchron. Der sogenannte Zeitrastergleichlauf (festes Zeitraster bei der Übertragung einer festen Zeichenfolge) wird nur während der Übertragung eines Zeichens aufrechterhalten. Jedem zu übertragenden Zeichen geht ein Synchronisierungsimpuls, auch als Startbit bezeichnet, voraus. Die Länge der Startbitübertragung legt den Takt fest. Das Ende des Zeichentransfers bilden ein oder zwei Stopbits.

Vereinbarungen

Neben Start- und Stopbit sind weitere Vereinbarungen zwischen den beiden Kommunikationspartnern für eine serielle Datenübertragung nötig. Darunter fallen:

- Datenübertragungsgeschwindigkeit
- Telegrammanfangs- und Endekriterien (z. B. Zeichenverzugszeit)
- Parität
- Anzahl der Datenbits (7 oder 8 Bit/Zeichen)
- Anzahl der Stopbits (1 oder 2)

Übertragungssicherheit

Die Übertragungssicherheit spielt bei der Übertragung von Daten und bei der Wahl des Übertragungsverfahrens eine wichtige Rolle. Allgemein gilt, je mehr Schichten des Referenzmodells durchlaufen werden, desto höher ist die Übertragungssicherheit.

Einordnung der vorhandenen Protokolle

Die Einordnung der vorhandenen Protokolle des Kommunikationsmoduls in das Referenzmodell können Sie der folgenden Abbildung entnehmen.

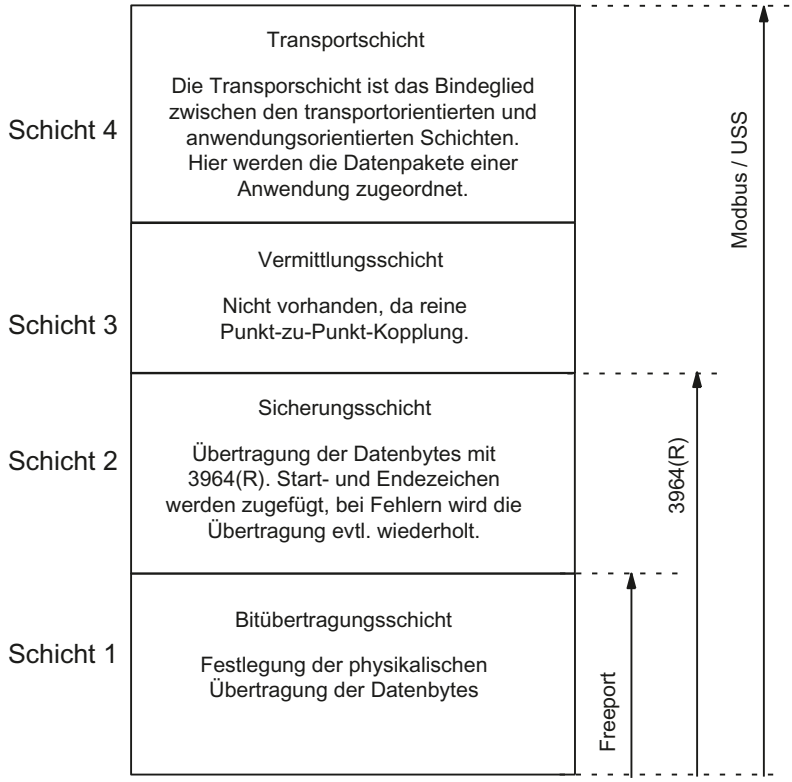

Bild 1-37 Einordnung der vorhandenen Protokolle des Kommunikationsmoduls in das Referenzmodell

Übertragungssicherheit bei Freeport

Übertragungssicherheit bei Verwendung von Freeport:

- Dadurch, dass bei einem Datentransport mit Freeport neben der Verwendung des Paritätsbits keine weiteren Maßnahmen zur Datensicherung erfolgen, ist eine Datenübertragung mit dem Freeport zwar sehr effizient, was den Datendurchsatz anbelangt, ein abgesicherter Datentransport ist aber nicht gegeben. Durch Parametrierung von Telegrammanfangs- und Telegrammendebedingungen kann bereits eine gewisse Datensicherheit erreicht werden.
- Durch die Verwendung des Paritätsbits wird das Kippen eines Bits in einem zu übertragenden Zeichen abgesichert. Kippen zwei oder mehr Bits eines Zeichens, so kann nicht mehr garantiert werden, dass diese Fehler erkannt werden.

- Soll die Übertragungssicherheit erhöht werden, so kann dies z. B. durch Einführung einer Prüfsumme, einer Längenangabe eines Telegramms oder durch parametrierbare Endebedingungen erfolgen. Diese Maßnahmen müssen durch den Anwender realisiert werden.
- Eine weitere Steigerung der Datensicherheit kann durch die Einführung von Quittungstelegrammen auf Sende– oder Empfangstelegramme erfolgen. Dies ist bei hochwertigen Protokollen zur Datenkommunikation realisiert (ISO–7–Schichten–Referenzmodell).

Übertragungssicherheit bei 3964(R)

Das Paritätsbit dient der Erhöhung der Datensicherheit; es ergänzt die Anzahl der übertragenen Datenbits je nach Parametrierung auf eine gerade oder ungerade Anzahl.

Durch die Verwendung des Paritätsbits wird das Kippen eines Bits in einem zu übertragenden Zeichen abgesichert. Kippen zwei oder mehr Bits eines Zeichens, so können diese Fehler nicht mehr sicher erkannt werden.

Falls "keine" Parität eingestellt ist, wird kein Paritätsbit übertragen. Dies reduziert die Übertragungssicherheit.

Je nachdem, ob Sie mit oder ohne Blockprüfzeichen Daten übertragen wollen, unterscheidet man zwischen:

- Datenübertragung ohne Blockprüfzeichen: 3964
  Die Übertragungssicherheit wird durch einen festgelegten Telegrammaufbau, Telegrammabbau und Telegrammwiederholungen erreicht.
- Datenübertragung mit Blockprüfzeichen: 3964R
  Die hohe Übertragungssicherheit wird durch einen festgelegten Telegrammaufbau und Telegrammabbau, Telegrammwiederholungen sowie die Mitführung des Blockprüfzeichens (BCC) erreicht.

Bei Beschreibung und Hinweisen, die sich auf beide Datenübertragungsarten beziehen, verwenden wir in diesem Handbuch die Bezeichnung 3964(R).

Übertragungssicherheit bei Modbus und USS

Das Paritätsbit dient der Erhöhung der Übertragungssicherheit; es ergänzt die Anzahl der übertragenen Datenbits je nach Parametrierung auf eine gerade oder ungerade Anzahl.

Durch die Verwendung des Paritätsbits wird das Kippen eines Bits in einem zu übertragenden Zeichen abgesichert. Kippen zwei oder mehr Bits eines Zeichens, so kann dieser Fehler nicht mehr sicher erkannt werden.

Falls "keine" Parität eingestellt ist, wird kein Paritätsbit übertragen. Dies reduziert die Übertragungssicherheit.

Bei Modbus wird zudem die zyklische Redundanzprüfung (CRC: cyclic redundancy check) verwendet. Dabei wird vor der Datenübertragung für jeden Datenblock der Nutzdaten zusätzliche Redundanz in Form eines sogenannten CRC-Werts angefügt. Dieser ist ein nach einem bestimmten Verfahren berechneter Prüfwert, mit dessen Hilfe man eventuelle während Übertragung aufgetretene Fehler erkennen kann.

Bei USS wird zudem ein Blockprüfzeichen BCC (block check character) verwendet. Während des Empfangs wird das Blockprüfzeichen gebildet und nach dem Einlesen des gesamten

Telegramms mit dem empfangenen BCC verglichen. Stimmen diese nicht überein, wird das Telegramm nicht ausgewertet. (Wird ein Zeichen falsch übertragen, wird ein Fehler sicher erkannt. Wird eine gerade Anzahl an Zeichen falsch übertragen worden, so kann ein Fehler nicht mehr sicher erkannt werden.)

RS232-Betrieb

Der RS232-Betrieb wird von folgenden Kommunikationsmodulen unterstützt:

- CM PtP RS232 BA
- CM PtP RS232 HF
- CM PtP (ET 200SP)

Beim RS232-Betrieb erfolgt die Datenübertragung über zwei Leitungen. Es steht jeweils eine Leitung für die Senderichtung und die Empfangsrichtung zur Verfügung. Es kann gleichzeitig gesendet und empfangen werden (Vollduplex).

RS232-Signale

Auf dem Kommunikationsmodul sind bei Einsatz der RS232-Physik neben den Signalen TXD, RXD und GND weitere RS232-Signale vorhanden:

TXD	Ausgang	Sendedaten (Transmitted Data); Schnittstelle sendet
RXD	Eingang	Empfangsdaten (Received Data); Schnittstelle empfängt
GND		Betriebserde (Ground); potentialfrei
DCD	Eingang	Empfangssignalpegel (Data Carrier detect); Trägersignal bei Anschluss eines Modems. Der Kommunikationspartner signalisiert, dass er einlaufende Daten erkennt.
DTR	Ausgang	Endgerät bereit (Data terminal ready); DTR auf "ON": Kommunikationsmodul eingeschaltet, betriebsbereit DTR auf "OFF": Kommunikationsmodul nicht eingeschaltet, nicht betriebsbereit
DSR	Eingang	Betriebsbereitschaft (Data set ready); DSR auf "ON": Kommunikationspartner signalisiert Betriebsbereitschaft DSR auf "OFF": Kommunikationspartner nicht eingeschaltet, nicht betriebsbereit
RTS	Ausgang	Sendeteil einschalten (Request to send); RTS auf "ON": Kommunikationsmodul sendebereit; signalisiert dem Kommunikationspartner, dass Daten zum Senden bereit liegen RTS auf "OFF": Kommunikationsmodul nicht sendebereit

CTS	Eingang	Sendebereitschaft (Clear to send); Kommunikationspartner kann Daten vom Kommunikationsmodul empfangen (Antwort auf RTS = ON des Kommunikationsmoduls) CTS auf "ON": Signalisiert dem Kommunikationspartner Empfangsbereitschaft CTS auf "OFF": Signalisiert dem Kommunikationspartner "Nicht empfangsbereit"
RI	Eingang	Ankommender Ruf bei Anschluss eines Modems (Ring Indicator)

Nach dem Einschalten des Kommunikationsmoduls befinden sich die Ausgangssignale im Zustand OFF (inaktiv).

Die Bedienung der Steuersignale DTR/DSR und RTS/CTS können Sie über die Oberfläche des Kommunikationsmoduls parametrieren.

Die RS232-Signale können nicht beeinflusst werden bei:

- parametrierter Datenflusskontrolle "Hardware RTS immer geschaltet" (entspricht der automatischen Bedienung der Begleitsignale)
- parametrierter Datenflusskontrolle "Hardware RTS immer ON" (das entspricht der Hardware Flusskontrolle mit RTS/CTS)
- parametrierter Datenflusskontrolle "Hardware RTS immer ON, DTR/DSR ignorieren"

Siehe hierzu auch Kapitel Handshake-Verfahren (Seite 1632).

Steckleitungen

Zur Verbindung mit einem Kommunikationspartner, der ebenfalls über einen 9-poligen Sub-D-Stiftstecker verfügt, stehen folgende Standard-Steckleitungen in verschiedenen Längen zur Verfügung:

Artikelnummer	6ES7902-1AB00-0AA0	6ES7902-1AC00-0AA0	6ES7902-1AD00-0AA0
Produkttyp-Bezeichnung	S7 Steckleitung RS232		
Leitungslänge	5 m	10 m	15 m

In der folgenden Tabelle finden Sie die Pinbelegung des 9-poligen Sub-D-Stiftsteckers des jeweiligen Kommunikationsmoduls.

Stiftstecker*	Pin	Bezeichnung	Eingang/ Ausgang	Benötigt/optional bei Selbst-anfertigung
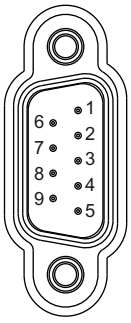	1	DCD	Eingang	Optional
	2	RXD	Eingang	Benötigt
	3	TXD	Ausgang	Benötigt
	4	DTR	Ausgang	Optional
	5	GND	—	Benötigt
	6	DSR	Eingang	Optional
	7	RTS	Ausgang	Optional
	8	CTS	Eingang	Optional
	9	RI	Eingang	Optional

* Ansicht von vorn

Die Leitung oder der Stecker der genannten Steckleitungen stehen nicht als Einzelartikel zur Verfügung. Falls Sie Steckleitungen selbst anfertigen, müssen Sie darauf achten, dass nicht beschaltete Eingänge beim Kommunikationspartner möglicherweise auf Ruhepotential gelegt werden müssen.

Beachten Sie, dass Sie nur geschirmte Steckergehäuse verwenden dürfen. Der Kabelschirm muss beidseitig großflächig mit dem Steckergehäuse verbunden sein.

⚠ VORSICHT

Kabelschirm nie mit GND verbinden

Verbinden Sie niemals Kabelschirm und GND miteinander, da die Schnittstellen zerstört werden können. GND muss in jedem Fall auf beiden Seiten verbunden werden (Pin 5), da sonst ebenfalls eine Zerstörung der Schnittstellenmodule möglich ist.

Im folgenden Bild ist die Steckleitung für eine Punkt-zu-Punkt-Kopplung zwischen einem Kommunikationsmodul und einem Kommunikationspartner dargestellt.

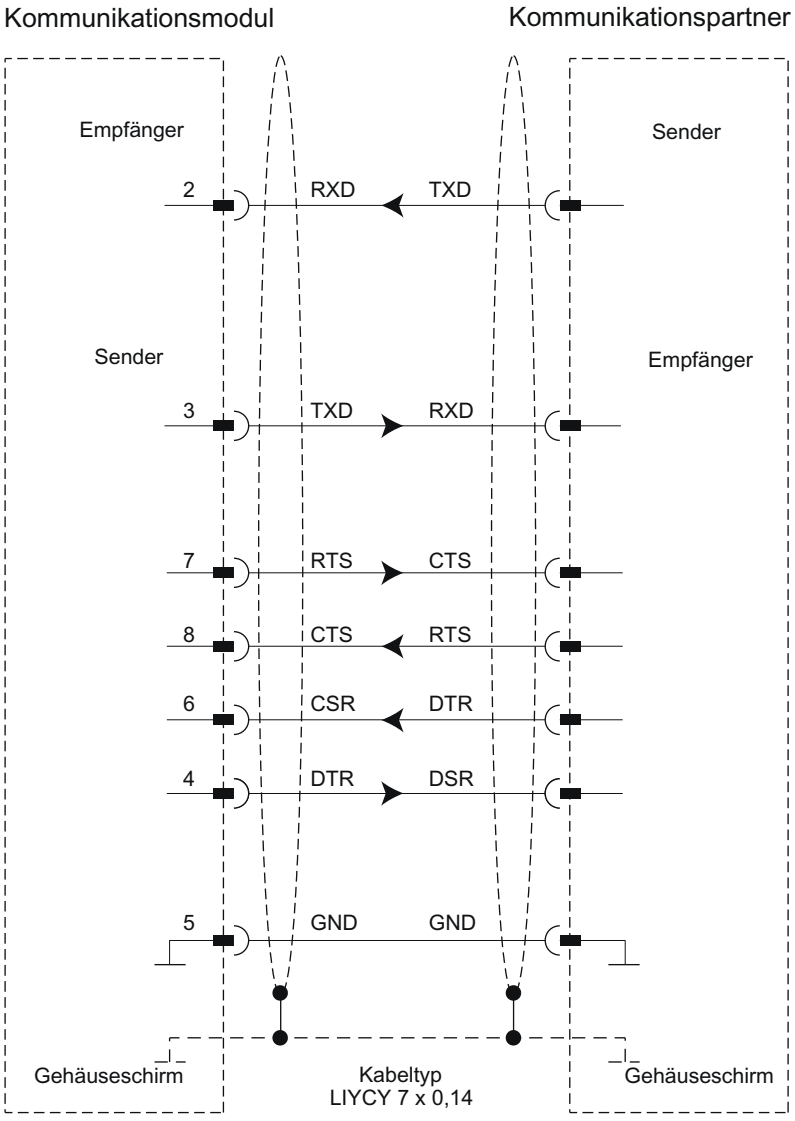

RS422-Betrieb

Der RS422-Betrieb wird von folgenden Kommunikationsmodulen unterstützt:

- CM PtP RS422/485 BA
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Bei RS422-Betrieb erfolgt die Datenübertragung über zwei Leitungspaare (Vierdraht-Betrieb). Es steht jeweils ein Leitungspaar für die Senderichtung und die Empfangsrichtung zur Verfügung. Es kann gleichzeitig gesendet und empfangen werden (Vollduplex).

Jeder Kommunikationspartner muss simultan eine Sende- und Empfangseinrichtung betreiben können.

Die Daten können zwischen zwei oder mehreren Kommunikationspartnern gleichzeitig ausgetauscht werden. Bei RS422 Mehrpunktbetrieb darf immer nur ein Slave zu einem Zeitpunkt senden.

Betriebsarten der Schnittstelle

In der folgenden Tabelle sind die Betriebsarten der Schnittstelle kommunikationsmodul- und protokollübergreifend zusammengestellt.

Das Kommunikationsmodul kann im RS422-Betrieb in folgenden Topologien eingesetzt werden:

- Kopplung zwischen zwei Teilnehmern: Punkt-zu-Punkt-Kopplung
- Kopplung zwischen mehreren Teilnehmern: Mehrpunkt-Kopplung (nur verfügbar bei CM PtP (ET 200SP))

Betriebsart	Beschreibung
Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)	In dieser Betriebsart sind beide Teilnehmer gleichberechtigt.
Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master)	Das Kommunikationsmodul kann als Mehrpunkt Master eingesetzt werden.
Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)	Das Kommunikationsmodul kann als Mehrpunkt Slave eingesetzt werden.

Bei einer Mehrpunkt Master/Slave-Topologie im RS422-Betrieb gilt:

- Der Sender des Master wird mit den Empfängern aller Slaves zusammengeschaltet.
- Die Sender der Slaves werden mit dem Empfänger des Master zusammengeschaltet.
- Die Vorbelegung erfolgt beim Empfänger des Master und bei dem Empfänger eines Slave. Alle anderen Slaves arbeiten ohne Vorbelegung.

RS422-Signale

Auf dem Kommunikationsmodul sind bei Einsatz der RS422-Physik folgende Signale vorhanden:

T (A) -	Ausgang	Sendedaten (Transmitted Data)
T (B) +	Ausgang	Sendedaten (Transmitted Data)
R (A) -	Eingang	Empfangsdaten (Received Data)
R (B) +	Eingang	Empfangsdaten (Received Data)
GND		Betriebserde (Ground); potentialfrei

Steckleitungen

Zur Verbindung mit einem Kommunikationspartner, der ebenfalls über eine 15-polige Sub-D-Buchse verfügt, stehen folgende Standard-Steckleitungen in verschiedenen Längen zur Verfügung:

Artikelnummer	6ES7902-3AB00-0AA0	6ES7902-3AC00-0AA0	6ES7902-3AG00-0AA0
Produkttyp-Bezeichnung	S7 Steckleitung RS422		
Leitungslänge	5 m	10 m	50 m

In der folgenden Tabelle finden Sie die Anschlussbelegung der 15-poligen Sub-D-Buchse des jeweiligen Kommunikationsmoduls.

Buchse*	Pin	Bezeichnung	Eingang/Ausgang
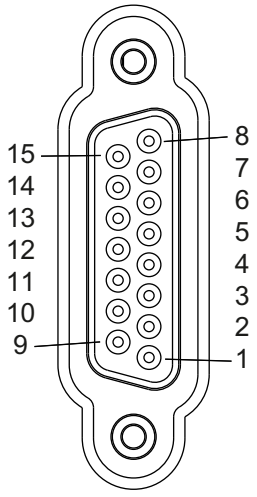	1	-	-
	2	T (A) -	Ausgang
	3	-	-
	4	R (A) -	Eingang
	5	-	-
	6	-	-
	7	-	-
	8	GND	-
	9	T (B) +	Ausgang
	10	-	-
	11	R (B) +	Eingang
	12	-	-
	13	-	-
	14	-	-
	15	-	-

* Ansicht von vorn

Die Leitung oder der Stecker der genannten Steckleitungen stehen nicht als Einzelartikel zur Verfügung. Falls Sie Steckleitungen selbst anfertigen, müssen Sie darauf achten, dass nicht beschaltete Eingänge beim Kommunikationspartner möglicherweise auf Ruhepotential gelegt werden müssen.

Beachten Sie, dass Sie nur geschirmte Steckergehäuse verwenden dürfen. Der Kabelschirm muss beidseitig großflächig mit dem Steckergehäuse verbunden sein.

⚠ VORSICHT

Kabelschirm nie mit GND verbinden

Verbinden Sie niemals Kabelschirm und GND miteinander, da die Schnittstellen zerstört werden können. GND muss in jedem Fall auf beiden Seiten verbunden werden (Pin 8), da sonst ebenfalls eine Zerstörung der Schnittstellenmodule möglich ist.

Im folgenden Bild ist die Steckleitung für eine Punkt-zu-Punkt-Kopplung zwischen einem Kommunikationsmodul und einem Kommunikationspartner dargestellt.

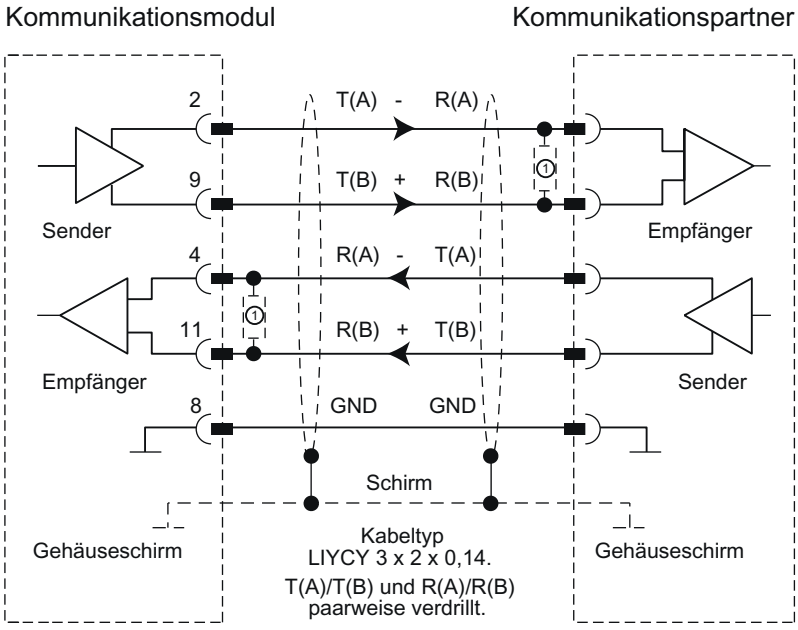

① Bei Leitungslängen größer 50 m müssen Sie für einen störungsfreien Datenverkehr einen Abschlußwiderstand von ca. 330 Ω auf der Empfängerseite einlöten.

Hinweis

Bei dem verwendeten Kabeltyp sind für ein Kommunikationsmodul als Kommunikationspartner folgende Längen möglich: max. 1200 m bei 19200 Baud, max. 500 m bei 38400 Baudmax. 250 m bei 115200 Baud.

RS485-Betrieb

Der RS485-Betrieb wird von folgenden Kommunikationsmodulen unterstützt:

- CM PtP RS422/485 BA
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Bei RS485-Betrieb erfolgt die Datenübertragung über ein Leitungspaar (Zweidraht-Betrieb). Das Leitungspaar steht abwechselnd für die Senderichtung und die Empfangsrichtung zur Verfügung. Es kann entweder gesendet oder empfangen werden (Halbduplex). Nach einem Sendevorgang wird sofort auf Empfang umgeschaltet (Empfangsbereitschaft). Sobald erneut ein Sendeauftrag eintrifft, wird wieder auf Senden umgeschaltet.

Betriebsarten der Schnittstelle

In der folgenden Tabelle sind die Betriebsarten der Schnittstelle kommunikationsmodul- und protokollübergreifend zusammengestellt.

Betriebsart	Beschreibung
Halbduplex (RS485) Zweidraht-Betrieb	Betriebsart für Punkt-zu-Punkt-Kopplung oder Mehrpunkt-Kopplung (Multipoint) im Zweidraht-Betrieb. Das Kommunikationsmodul kann sowohl Master als auch Slave sein.

Falls Sie Freeport im RS485-Betrieb (Halbduplex, Zweidrahtbetrieb) betreiben, müssen Sie im Anwenderprogramm dafür sorgen, dass immer nur ein Teilnehmer sendet. Wenn gleichzeitig gesendet wird, werden die Telegramme verfälscht.

Bei Modbus ist automatisch gewährleistet, dass immer nur ein Teilnehmer sendet.

Umschaltzeiten für RS485-Kommuniktionsmodul im Halbduplexbetrieb

Die maximale Umschaltzeit zwischen Senden und Empfangen beträgt 0,1 ms.

RS485-Signale

Auf dem Kommunikationsmodul sind bei Einsatz der RS485-Physik folgende Signale vorhanden:

R (A)/T (A) -	Eingang/Ausgang	Empfangs-/Sendedaten (Received/Transmitted Data)
R (B)/T (B) +	Eingang/Ausgang	Empfangs-/Sendedaten (Received/Transmitted Data)
GND		Betriebserde (Ground); potentialfrei

Steckleitungen

In der folgenden Tabelle finden Sie die Anschlussbelegung der 15-poligen Sub-D-Buchse des jeweiligen Kommunikationsmoduls.

Buchse*	Pin	Bezeichnung	Eingang/Ausgang
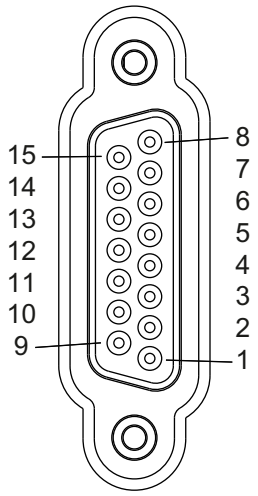	1	-	-
	2	-	-
	3	-	-
	4	R (A)/T (A) -	Ein-/Ausgang
	5	-	-
	6	-	-
	7	-	-
	8	GND	-
	9	-	-
	10	-	-
	11	R (B)/T (B) +	Ein-/Ausgang
	12	-	-
	13	-	-
	14	-	-
	15	-	-

* Ansicht von vorn

Beim Anfertigen der Steckleitungen müssen Sie darauf achten, dass nicht beschaltete Eingänge beim Kommunikationspartner möglicherweise auf Ruhepotential gelegt werden müssen.

Beachten Sie, dass Sie nur geschirmte Steckergehäuse verwenden dürfen. Der Kabelschirm muss beidseitig großflächig mit dem Steckergehäuse verbunden sein.

⚠ VORSICHT

Kabelschirm nie mit GND verbinden

Verbinden Sie niemals Kabelschirm und GND miteinander, da die Schnittstellen zerstört werden können. GND muss in jedem Fall auf beiden Seiten verbunden werden (Pin 8), da sonst ebenfalls eine Zerstörung der Schnittstellenmodule möglich ist.

Im folgenden Bild ist die Steckleitung für eine Punkt-zu-Punkt-Kopplung zwischen einem Kommunikationsmodul und einem Kommunikationspartner dargestellt.

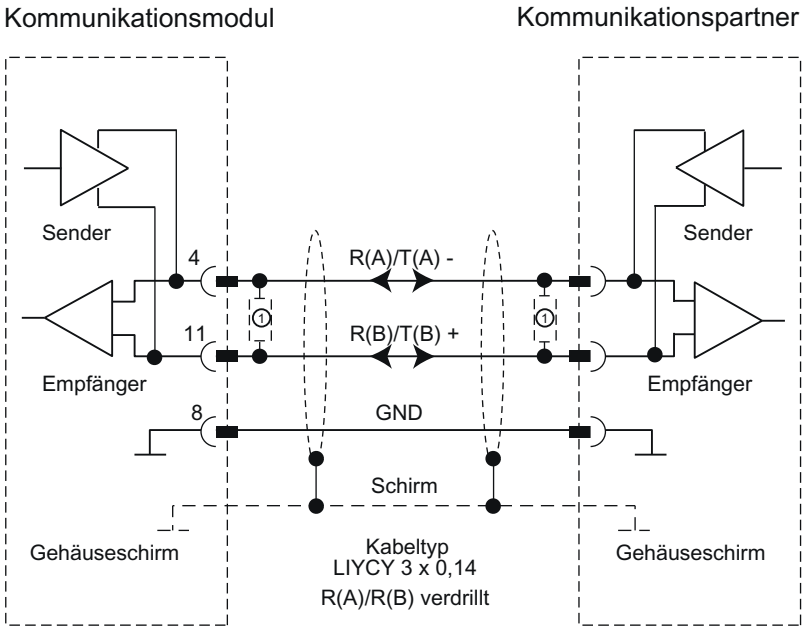

① Bei Leitungslängen größer 50 m müssen Sie für einen störungsfreien Datenverkehr einen Abschlußwiderstand von ca. 330 Ω auf der Empfängerseite einlöten.

Handshake-Verfahren

Einleitung

Handshake-Verfahren steuern den Datenfluss zwischen zwei Kommunikationspartnern. Durch die Verwendung von Handshake-Verfahren wird vermieden, dass bei unterschiedlich schnell arbeitenden Geräten Daten bei der Übertragung verloren gehen.

Grundsätzlich lassen sich die folgenden Verfahren unterscheiden:

Tabelle 1-97 Übersicht der Verfahren und Schnittstellen

Verfahren	RS232	RS422	RS485
Software-Datenflusskontrolle XON/XOFF	X	X	-
Hardware-Datenflusskontrolle RTS/CTS	X	-	-
Automatische Bedienung der Begleitsignale	X	-	-

Software-Datenflusskontrolle

Die Realisierung der Software-Datenflusskontrolle auf dem Kommunikationsmodul wird folgendermaßen durchgeführt:

- XON/XOFF
  - Sobald das Kommunikationsmodul durch Parametrierung in die Betriebsart "XON/XOFF" gebracht wurde, sendet es das Zeichen XON und erlaubt so dem Kommunikationspartner zu senden.
  - Bei Erreichen der parametrierten maximalen Telegrammanzahl bzw. 16 Zeichen bevor der Empfangspuffer überläuft sendet das Kommunikationsmodul das Zeichen XOFF und fordert damit den Kommunikationspartner auf, dass Senden zu unterbrechen. Sendet der Kommunikationspartner trotzdem weiter, wird bei Überlauf des Empfangspuffers eine Fehlermeldung generiert. Die empfangenen Daten des letzten Telegramms werden verworfen.
  - Sobald ein Telegramm durch die CPU abgeholt wurde und der Empfangspuffer wieder aufnahmebereit ist, sendet das Kommunikationsmodul das Zeichen XON.
  - Empfängt das Kommunikationsmodul während eines Sendevorgangs das Zeichen XOFF, unterbricht das Kommunikationsmodul den Sendevorgang bis vom Kommunikationspartner wieder ein XON empfangen wird. Wird nach einer bestimmten parametrierbaren Zeit kein XON empfangen, wird der Sendevorgang abgebrochen und eine entsprechende Fehlermeldung ausgegeben.

Hinweis

Die Zeichen für XON und XOFF sind parametrierbar (beliebiges ASCII-Zeichen).

Bei der Parametrierung der Software-Datenflusskontrolle XON/XOFF dürfen die Nutzdaten keines der parametrierten XON- oder XOFF-Zeichen enthalten.

Hardware-Datenflusskontrolle

Hinweis

Bei der Parametrierung "Hardware RTS immer ON, DTR/DSR ignorieren" müssen die Signale DTR/DSR nicht verdrahtet werden.

Bei der Parametrierung "Hardware RTS immer ON" müssen Sie unbedingt eine Vollverdrahtung der verwendeten Schnittstellensignale vornehmen. Stellen Sie sicher, dass das lokale RTS (out) mit dem CTS (in) des Kommunikationspartners und das lokale CTS mit dem RTS des Kommunikationspartners verbunden ist. Entsprechend muss das lokale DTR mit dem DSR des Kommunikationspartners und das lokale DSR mit dem DTR des Kommunikationspartners verbunden sein.

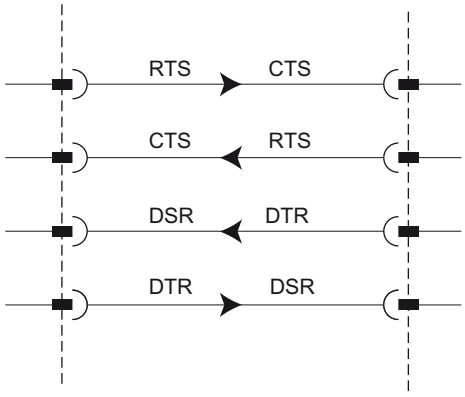

Bild 1-38 Verdrahtung der Schnittstellensignale

- Hardware RTS immer ON, DTR/DSR ignorieren
  - Sobald das Kommunikationsmodul durch Parametrierung in eine Betriebsart mit "Hardware RTS immer ON" gebracht wurde, signalisiert es dem Kommunikationspartner durch RTS = ON seine Betriebsbereitschaft.
  - Bei Erreichen der parametrierten maximalen Telegrammanzahl bzw. 16 Zeichen bevor der Empfangspuffer überläuft wird RTS auf OFF gesetzt.
    Sendet der Kommunikationspartner trotzdem weiter, wird bei Überlauf des Empfangspuffers eine Fehlermeldung generiert. Die empfangenen Daten des letzten Telegramms werden verworfen.
  - Sobald ein Telegramm durch die CPU abgeholt wurde und der Empfangspuffer wieder aufnahmebereit ist, wird RTS wieder auf ON gesetzt.
  - Erfolgt während des Sendevorgangs ein Wechsel von CTS auf OFF, so unterbricht das Kommunikationsmodul den Sendevorgang bis CTS wieder auf ON gesetzt wird. Wird nach einer parametrierbaren Zeit CTS nicht wieder auf ON gesetzt, wird der Sendevorgang abgebrochen und eine entsprechende Fehlermeldung ausgegeben.
- Hardware RTS immer ON
  Die Betriebsart "Hardware RTS immer ON" entspricht der Betriebsart "Hardware RTS immer ON, DTR/DSR ignorieren", es muss jedoch zusätzlich DTR und DSR verdrahtet werden.
  - Sobald das Kommunikationsmodul durch Parametrierung in eine Betriebsart mit "Hardware RTS immer ON" gebracht wurde, signalisiert es dem Kommunikationspartner mit dem Setzen von DTR = ON und RTS = ON generell seine Betriebsbereitschaft.
  - Bei Erreichen der parametrierten maximalen Telegrammanzahl bzw. 16 Zeichen bevor der Empfangspuffer überläuft wird RTS auf OFF gesetzt.
    Sendet der Kommunikationspartner trotzdem weiter, wird bei Überlauf des Empfangspuffers eine Fehlermeldung generiert. Die empfangenen Daten des letzten Telegramms werden verworfen.
  - Sobald ein Telegramm durch die CPU abgeholt wurde und der Empfangspuffer wieder aufnahmebereit ist, wird RTS wieder auf ON gesetzt.
  - Erfolgt während des Sendevorgangs ein Wechsel von CTS auf OFF, so unterbricht das Kommunikationsmodul den Sendevorgang bis CTS wieder auf ON gesetzt wird. Wird nach einer parametrierbaren Zeit CTS nicht wieder auf ON gesetzt, wird der Sendevorgang abgebrochen und eine entsprechende Fehlermeldung ausgegeben.
  - Bei einem Wechsel von DSR = ON nach OFF wird ein laufender Sendeauftrag mit einer Fehlermeldung abgebrochen.

Automatische Bedienung der Begleitsignale

- **Hardware RTS immer geschaltet**
  Die Realisierung des "Hardware RTS immer geschaltet" auf dem Kommunikationsmodul wird folgendermaßen durchgeführt:
  - Sobald das Kommunikationsmodul durch Parametrierung in die Betriebsart mit "Hardware RTS immer geschaltet" gebracht wurde, setzt es die Leitungen RTS auf OFF und DTR auf ON (Kommunikationsmodul betriebsbereit).
    Das Senden von Telegrammen ist erst möglich, nachdem die Leitung DSR auf ON gesetzt wurde. Solange DSR auf OFF gesetzt bleibt, werden keine Daten über die RS232-Schnittstelle gesendet. Ein Sendeauftrag wird mit einer entsprechenden Fehlermeldung abgebrochen.
  - Steht ein Sendeauftrag an, wird RTS auf ON gesetzt und die parametrierte RTS ON-Verzögerung gestartet. Nach Ablauf der Datenausgabezeit wird überprüft, ob der Kommunikationspartner CTS auf ON gesetzt hat. Ist dies der Fall, werden die Daten über die RS232-Schnittstelle gesendet.
  - Wird innerhalb der RTS ON-Verzögerung die Leitung CTS nicht auf ON gesetzt, oder erfolgt innerhalb des Sendevorgangs ein Wechsel von CTS auf OFF wird der Sendeauftrag abgebrochen und eine entsprechende Fehlermeldung generiert.
  - Nach dem Senden der Daten wird nach Ablauf der parametrierten RTS OFF-Verzögerung die Leitung RTS auf OFF gesetzt. Es erfolgt kein Warten auf den Wechsel von CTS auf OFF.
  - Ein Empfang von Daten über die RS232-Schnittstelle ist immer möglich. Droht der Empfangspuffer des Kommunikationsmoduls überzulaufen, erfolgt keine Reaktion.
  - Bei einem Wechsel von DSR = ON auf OFF wird ein laufender Sendeauftrag mit einer Fehlermeldung abgebrochen.

  Hinweis

  Die "RTS ON-Verzögerung" sollten Sie so dimensionieren, dass der Kommunikationspartner in Empfangsbereitschaft gehen kann, bevor die Zeit abläuft.

  Die "RTS OFF-Verzögerung" sollten Sie so einstellen, dass der Kommunikationspartner die letzten Zeichen des Telegramms vollständig empfangen kann, bevor RTS und damit der Sendewunsch weggenommen wird.

  Hinweis

  Bei der Parametrierung einer automatischen Bedienung der RS232-Signale ist das Steuern von RTS und DTR über die entsprechende Anweisung nicht möglich!

Zeitdiagramm

Das folgende Bild zeigt den zeitlichen Ablauf eines Sendeauftrags bei parametrierter Datenflusskontrolle "Hardware RTS immer geschaltet":

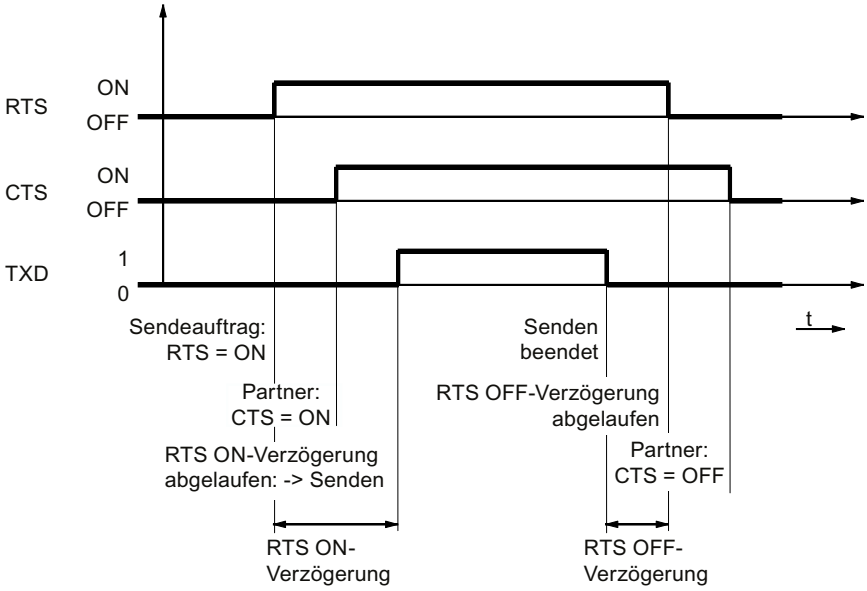

Bild 1-39 Zeitdiagramm bei Hardware RTS immer geschaltet

Weitere Informationen

Hinweis

Bei folgenden Einstellungen wird die Bedienung von DTR/DSR bzw. RTS/CTS vom Kommunikationsmodul übernommen:

- Hardware RTS immer ON, DTR/DSR ignorieren
- Hardware RTS immer ON
- Hardware RTS immer geschaltet

Konfigurieren/Parametrieren

Konfigurieren/Parametrieren eines Kommunikationsmoduls

In den folgenden Kapiteln finden Sie Erläuterungen zu folgenden Protokollen und deren Parametern:

- Kommunikation über Freeport (Seite 1638)
- Kommunikation über 3964(R) (Seite 1647)
- Kommunikation über Modbus RTU (Seite 1652)
- Kommunikation über USS (Seite 1658)

Diese Informationen benötigen Sie, um die Parametrierung und im Weiteren die Programmierung der Kommunikation entsprechend des eingesetzten Protokolls vornehmen zu können.

Die Konfiguration und Parametrierung erfolgt in der Gerätesicht von STEP 7 (TIA Portal) und im Eigenschaftendialog des Kommunikationsmoduls. Einige Konfigurationen können auch zur Laufzeit mithilfe der entsprechenden "Config"-Anweisungen (Port_Config, Send_Config, Receive_Config, P3964_Config) geändert werden.

Vorgehen zum Einrichten einer PtP-Kommunikation

Die Vorgehensweise ist unabhängig vom eingesetzten Kommunikationsmodul.

1. Konfigurieren Sie in der Gerätesicht des Hardware-Editors von STEP 7 (TIA Portal) einen S7-1500-Aufbau mit CPU und Kommunikationsmodul.
2. Nehmen Sie im Bereich "Allgemein", im Register "Eigenschaften" die Parametrierung der Schnittstelle des Kommunikationsmoduls vor (Protokoll, Protokollparameter, Adressen).

Kommunikation über Freeport

Vorgehen zum Aufbau einer seriellen Verbindung mit Freeport

Voraussetzungen

- Die Hardware ist aufgebaut und es besteht eine elektrische Verbindung zum Koppelpartner.
- Das Projekt ist in STEP 7 (TIA Portal) angelegt und die CPU ist in der Hardware-Konfiguration eingefügt.

Vorgehen - Hardware-Konfiguration

1. Fügen Sie das Kommunikationsmodul CM PtP in die Hardware-Konfiguration ein.
2. Stellen Sie die Kommunikationsparameter entsprechend dem Koppelpartner ein:
   z. B. Übertragungsgeschwindigkeit, Zeichenrahmen, Telegrammanfang und Telegrammende
   Diese Parameter werden bei jedem CPU-Start an das Kommunikationsmodul CM PtP übertragen.

Vorgehen - Programmierung

1. Legen Sie die Datenstruktur an, die die zu übertragenden Daten enthalten soll.

Senden von Daten

1. Fügen Sie die Anweisungen aus der Bibliothek PtP Communication ein: Send_P2P zum Senden von Daten
2. Verschalten Sie die Eingangs- und Ausgangsparameter der Anweisung, z. B.:
   – HWID aus den Systemvariablen am Eingang PORT
   – Datenstruktur mit zu sendenden Daten am Eingang BUFFER

   Hinweis: Im Betrieb wird jede steigende Flanke am Eingang REQ einmal den spezifizierten Datenbereich senden. Der Baustein muss so oft aufgerufen werden, bis durch DONE angezeigt wird, dass die Daten an die Baugruppe übertragen wurden.
   Im Fehlerfall wird durch einmaliges Setzen von ERROR und einer entsprechenden Information in STATUS angezeigt, dass die Daten nicht übertragen wurden.

Empfangen von Daten:

1. Fügen Sie die Anweisungen aus der Bibliothek PtP Communication ein: Receive_P2P zum Senden von Daten
2. Verschalten Sie die Eingangs- und Ausgangsparameter der Anweisung, z. B.:
   – HWID aus den Systemvariablen am Eingang PORT
   – Datenstruktur zur Ablage der zu empfangenden Daten am Eingang BUFFER

   Hinweis: Im Betrieb wird durch einen High-Pegel am Ausgang NDR angezeigt, dass neue Daten empfangen und im spezifizierten Datenbereich abgelegt wurden. Der Baustein muss so lange aufgerufen werden, bis NDR=TRUE. Dann können die empfangenen Daten ausgewertet und der RECEIVE_P2P wieder aufrufen werden.

Optionale Ergänzungen

- Anweisungen, die mit _Config enden, können optional verwendet werden, um im laufenden Betrieb vom Anwenderprogramm die Parameter der Hardware-Konfiguration zu ändern. Die Änderungen werden nicht in der Hardware-Konfiguration gespeichert. Sie werden beim nächsten Neustart wieder überschrieben.
- Die Anweisungen Signal_Set und Signal_Get können verwendet werden, um die RS232-Begleitsignale individuell zu steuern, falls eine automatische Bedienung nicht geeignet ist.

Datenübertragung mit Freeport

Einleitung

Freeport ist ein frei programmierbares telegrammbasiertes Protokoll, das auch als ASCII-Protokoll bekannt ist.

Das Freeport-Protokoll steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen dem Kommunikationsmodul und einem Kommunikationspartner. Das Freeport-Protokoll beinhaltet die Bitübertragungsschicht (Schicht 1).

Mit dem Freeport-Protokoll können Nachrichten mit beliebigem Aufbau (alle Zeichen von 00H bis FFH (bei Zeichenrahmen mit 8 Datenbits) bzw. von 00H bis 7FH (bei Zeichenrahmen mit 7 Datenbits) gesendet und empfangen werden.

Sowohl für die Sende- als auch für die Empfangsrichtung sind die Start- und Endekriterien eines Telegramms zu parametrieren. Start- und Endekriterien können unterschiedlich parametriert werden.

Für die Kommunikation mit einem Kommunikationspartner stehen ihnen Anweisungen zur Verfügung (siehe Überblick über die PtP-Programmierung (Seite 1661)).

Daten senden mit Freeport

Festlegen der Einstellungen für das Senden

Um eine Nachricht zu senden, muss dem Partner Anfang und Ende einer Nachricht mitgeteilt werden. Diese Einstellungen können in der Hardware-Konfiguration fest eingestellt oder zur Laufzeit mit der Anweisung Send_CFG angepasst werden. Sie können zwischen folgenden Möglichkeiten wählen bzw. diese auch kombinieren:

- Break vor Telegrammstart senden
  Sie können festlegen, dass zu Beginn jeder Nachrichtenübertragung zusätzlich ein Break gesendet wird, wenn die RTS ON-Verzögerungszeit verstrichen ist.
  Die Dauer des Signals "Break" wird in Bitzeiten angegeben.
  Erfolgt die Einsynchronisation durch andere Mechanismen, kann das Einhalten der Sendepause abgeschaltet werden.
- Idle Line senden
  Sie können festlegen, dass zu Beginn jeder Nachrichtenübertragung, zusätzlich das Signal "Idle Line" ausgegeben wird.
  Die Dauer des Signals "Idle Line" wird in Bitzeiten angegeben.
  Erfolgt die Einsynchronisation durch andere Mechanismen, kann das Einhalten der Sendepause abgeschaltet werden.
- RTS ON-Verzögerung
  Sie können die Zeit parametrieren, die nach der Sendeanforderung RTS (Request to send) vergehen muss, ehe die tatsächliche Datenübertragung beginnt (nur RS232).
- RTS OFF-Verzögerung
  Sie können die Zeit parametrieren, die nach der vollständigen Übertragung verstreichen soll, ehe das RTS-Signal deaktiviert wird (nur RS232).

- Senden bis einschließlich Endezeichen
  Sie können die Anzahl der Endezeichen (1 oder 2) und deren Wert parametrieren. Es wird bis zu dem/den Endezeichen gesendet, unabhängig von der gewählten Telegrammlänge. Das Endekennzeichen muss in den zu sendenden Daten enthalten sein. Es werden die Daten nur bis zum Endekennzeichen einschließlich gesendet, auch wenn eine größere Datenlänge angegeben ist.
- Anzahl angehängter Zeichen
  Eingabe der Anzahl angehängter Zeichen. Es wird bis zur parametrierten Länge gesendet. Das/die Endekennzeichen werden automatisch angehängt. Je nach Anzahl der Endekennzeichen werden 1 bis 5 Zeichen mehr zum Partner geschickt, als an der Anweisung angegeben.

Hinweis

Falls Sie "Break vor Telegrammstart senden", "Idle Line senden" und "RTS ON-Verzögerung" kombinieren, werden diese in der Reihenfolge "RTS ON-Verzögerung", "Break vor Telegrammstart senden" und "Idle Line senden" bearbeitet.

Daten empfangen mit Freeport

Festlegen des Nachrichtenanfangs

Bei der Datenübertragung mit Freeport können Sie zwischen mehreren verschiedenen Startkriterien wählen. Das Startkriterium legt fest, wann ein Telegramm beginnt. Ist ein Kriterium erfüllt, welches den Nachrichtenanfang anzeigt, wird mit dem Durchsuchen des Datenstroms nach Kriterien für das Nachrichtenende begonnen. Wählen Sie hier die Einstellungen aus, die den Eigenschaften des sendenden Kommunikationspartners entsprechen.

Für das Erkennen des Nachrichtenbeginns gibt es zwei unterschiedliche Verfahren:

- **Mit beliebigem Zeichen beginnen**
  Jedes beliebige Zeichen kann den Beginn einer Nachricht definieren (Voreinstellung). D. h. das erste Zeichen zu Beginn einer Kommunikation oder nach einem erkannten Telegrammende wird als das erste Zeichen einer Nachricht erkannt.
- **Mit spezieller Bedingung beginnen**
  Der Nachrichtenbeginn wird anhand der folgenden festgelegten Bedingungen erkannt.
  - **Nach Erkennung eines Line Break**
    Der Telegrammanfang wird erst akzeptiert, wenn zuvor ein Break empfangen wurde, d. h. der Partner muss zwingend vor dem Senden eines Telegramms ein Break senden.
  - **Nach Erkennung einer Idle Line**
    Der Telegrammanfang wird erst akzeptiert, wenn die parametrierte Idle Line-Dauer abgelaufen ist. Das setzt eine Mindestpause zwischen 2 Telegrammen voraus.
  - **Nach Erkennung des Startzeichens**
    Der Telegrammanfang wird erkannt, wenn das parametrierte Startzeichen erkannt wird.
  - **Nach Erkennung einer oder mehrerer Anfangssequenzen**
    Der Telegrammanfang wird erkannt, wenn die parametrierte, bis zu 5 Zeichen umfassende, Zeichenfolge erkannt wird. Es können bis zu 4 Anfangsequenzen parametriert werden. Die bis zu 5 Zeichen umfassenden Anfangssequenzen können auch "don't care characters" enthalten.

Beispiel:

Tabelle 1-98 Parametrierte Startbedingungen

Startbedingung	1. Zeichen	2. Zeichen	3. Zeichen	4. Zeichen	5. Zeichen
1	0x68	xx	xx	0x68	xx
2	0x10	0xaa	xx	xx	xx
3	0xdc	0xaa	xx	xx	xx
4	0xe5	xx	xx	xx	xx
:					

Folgende Nachricht wurde empfangen: 68 10 aa 68 bb 10 aa 16

Die Auswertung der Startkriterien beginnt mit dem Empfang des 1. Zeichens 0x68.
Das 2. und das 3. Zeichen sind beliebig.
Mit dem Empfang des 4. Zeichens (zweites 0x68) ist die 1. Startbedingung erfüllt und die weitere Auswertung der Nachricht beginnt.

Festlegung des Nachrichtenendes

Bei der Datenübertragung mit dem Freeport-Protokoll können Sie zwischen mehreren verschiedenen Endekriterien wählen. Das Endekriterium legt fest, wann ein Telegramm vollständig empfangen wurde.

Die einstellbaren Endekriterien sind:

- Erkenne Nachrichtenende durch Nachrichten-Zeitüberschreitung
- Erkenne Nachrichtenende durch Antwort-Zeitüberschreitung
- Nach Ablauf der Zeichenverzugszeit (Voreinstellung)
- Nach Empfang einer festen Telegrammlänge
- Nach Empfang einer maximalen Zeichenanzahl
- Lese Nachrichtenlänge aus Nachricht
- Nach Empfang einer Endesequenz

Nachrichten-Zeitüberschreitung

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Zeit für die Übertragung eines Telegramms abgelaufen ist. Die Zeitmessung beginnt mit der Erfüllung des Startkriteriums.

Antwort-Zeitüberschreitung

Mit der Antwortzeit wird das Antwortverhalten des Kommunikationspartners überwacht. Wird nach Abschluss eines Sendevorgangs kein gültiger Telegrammanfang erkannt, wird der Sendevorgang mit einer entsprechenden Meldung quittiert.

Das eigentliche Endekriterium muss zusätzlich parametriert werden.

Ablauf der Zeichenverzugszeit

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die eingestellte Maximalzeit zwischen aufeinanderfolgenden Zeichen überschritten wird (Zeichenverzugszeit). Der Wert wird in Bitzeiten angegeben.

Die Zeichenverzugszeit muss in diesem Fall so eingestellt werden, dass sie zwischen zwei aufeinanderfolgenden Telegrammen sicher abläuft. Sie sollte aber so groß sein, dass bei Sendepausen des Koppelpartners innerhalb eines Telegramms nicht fälschlicherweise das Telegrammende erkannt wird.

Feste Telegrammlänge

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Telegrammlänge erreicht ist.

Falls die Zeichenverzugszeit (sofern aktiviert) abläuft, bevor die feste Telegrammlänge erreicht ist, erfolgt eine Fehlermeldung und das Telegramm wird verworfen.

Beachten Sie Folgendes, wenn die Telegrammlänge der empfangenen Zeichen nicht mit der festen parametrierten Telegrammlänge übereinstimmt:

- Alle Zeichen, die nach Erreichen der festen parametrierten Telegrammlänge empfangen werden, werden solange verworfen bis ein neues Startkriterium erkannt wird.
- Ist ein anderes (aktiviertes) Endekriterium erfüllt, bevor die feste parametrierte Telegrammlänge erreicht wird, so erfolgt eine Fehlermeldung und das Telegramm wird verworfen.

Maximale Zeichenzahl

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Anzahl von Zeichen empfangen wurde.

Diese Einstellung kann mit den Einstellungen für "Zeichenverzugszeit" kombiniert werden. Das Telegramm gilt ebenfalls als fehlerfrei empfangen, wenn eine andere Endbedingung eintritt, unabhängig davon, ob die maximale Zeichenzahl erreicht wurde.

Beachten Sie Folgendes, wenn die Telegrammlänge der empfangenen Zeichen nicht mit der parametrierten maximalen Telegrammlänge übereinstimmt:

- Alle Zeichen, die nach Erreichen der parametrierten maximalen Zeichenzahl empfangen werden, werden solange verworfen bis ein neues Startkriterium (z. B. "Idle line") erkannt wird.
- Ist ein anderes (aktiviertes) Endekriterium erfüllt, bevor die parametrierte maximale Zeichenzahl erreicht wird, so wird dieser "Teil des Telegramms" als gültiges Telegramm gewertet und es wird auf ein neues Startkriterium gewartet. Alle vor der Erfüllung eines neuen Startkriteriums empfangenen Zeichen werden verworfen.

Hinweis

Ist kein weiteres Endekriterium aktiviert, so verhalten sich Feste Telegrammlänge und Maximale Zeichenzahl gleich.

Nachrichtenlänge in der Nachricht

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die im Empfangstelegramm mitgesendete Telegrammlänge erreicht wird.

Welche Zeichen für die Auswertung der Nachrichtenlänge herangezogen werden sollen, wird mit den folgenden Parametern festgelegt:

- **Offset des Längenfeldes in der Nachricht**
  Der Wert bestimmt die Position des Zeichens in der Nachricht, welches für die Bestimmung der Nachrichtenlänge herangezogen werden soll.
  Abhängig von der Puffergröße sind Werte von 0 bis 4095 Zeichen einstellbar.
- **Größe des Längenfeldes**
  Der Wert gibt an, wie viele Zeichen ab der ersten Auswertungsposition zur Bestimmung der Nachrichtenlänge herangezogen werden sollen.
  Es sind Werte von 1, 2 und 4 Zeichen einstellbar.
- **Anzahl nicht zur Längenangabe zählender Zeichen**
  Anzahl der Zeichen, die an das Telegramm angehängt sind, jedoch nicht zur Länge des Telegramms zählen. Der Wert gibt die Anzahl der Bytes am Ende des Telegramms an, die bei der Auswertung der Nachrichtenlänge nicht berücksichtigt werden sollen.
  Es sind Werte von 0 bis 255 Zeichen einstellbar.

Beispiel:

Parametrierungen für "Nachrichtenlänge in der Nachricht"

Offset des Längenfeldes in der Nachricht:	3. Byte (als Offset muss "2" parametriert werden)
Größe des Längenfeldes:	1 Byte
Anzahl nicht zur Längenangabe zählender Zeichen:	3 Byte

Nachricht					Anzahl nicht zur Längenangabe zählender Zeichen		
Startzeichen	Adresse	Längenfeld			Checksumme	Endezeichen	
Byte 1	Byte 2	Byte 3	Byte ...	Byte X	Byte X+1	Byte X+2	Byte X+3

Endesequenz

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Endesequenz (max. 5 Zeichen) empfangen wird. Die bis zu 5 Zeichen umfassenden Endesequenz kann auch "don't care characters" enthalten. Die empfangenen Daten werden inklusive Endesequenz von der CPU übernommen.

Wenn mit Endesequenz gearbeitet wird, ist die Übertragung nicht codetransparent, und es muss ausgeschlossen werden, dass die Endekennung/en in den Nutzdaten des Anwenders enthalten sind.

Hinweis

Telegrammendesequenz

Bei nur 1 Endzeichen **muss** der Eintrag zwingend in der 5. Zeile erfolgen.

Bei 2 Endezeichen **müssen** die Einträge in der 4. und der 5. Zeile erfolgen (keine Lücken).

Entsprechendes gilt für die Verwendung weiterer Zeichen.

Codetranparenz

Codetransparenz

Codetransparent bedeutet, dass in den Nutzdaten alle beliebigen Zeichenkombinationen vorkommen dürfen, ohne dass das Endekriterium erkannt wird.

Die Codetransparenz der Prozedur hängt von der Wahl des parametrierten Endekriteriums und der Flusskontrolle ab:

- Mit festgelegter Endesequenz oder Verwendung der Flusskontrolle XON/XOFF
  - nicht codetransparent
- Endekriterium Zeichenverzugszeit, feste Telegrammlänge, maximale Telegrammlänge, Nachrichtenzeitüberschreitung oder Antwortzeitüberschreitung und Nachrichtenlänge in der Nachricht
  - codetransparent

Empfangspuffer

Empfangspuffer der Baugruppe

Die Kommunikationsmodule verfügen über einen Empfangspuffer, der empfangene Telegramme zwischenspeichert, bis sie an die CPU übertragen werden. Der Empfangspuffer ist als Ringpuffer ausgeführt, d. h. die Telegramme werden solange in der Reihenfolge des Empfangs an die CPU übertragen, bis der Empfangspuffer voll ist . Werden danach weitere Telegramme empfangen, wird das jeweils älteste Telegramm überschrieben. Wurde "Überschreiben verhindern" parametriert, so wird bei vollem Empfangspuffer eine entsprechende Meldung generiert. Alle weiteren Telegramme werden solange verworfen, bis der Empfangspuffer wieder aufnahmebereit ist.

Bei der Parametrierung können Sie angeben, ob der Empfangspuffer im Anlauf gelöscht werden soll. Zusätzlich können Sie den Wertebereich (1 bis 255) für die Anzahl der gepufferten Empfangstelegramme angeben.

Der Empfangspuffer der Baugruppe umfasst, abhängig vom eingesetzten Kommunikationsmodul, bis zu 8 kByte (siehe Kapitel Einleitung (Seite 1614)). Die maximale Telegrammlänge beträgt 4 kByte. Damit kann jedes Kommunikationsmodul mindestens zwei Telegramme puffern.

Wenn Sie immer nur das neueste Telegramm zur CPU übertragen wollen, müssen Sie für die Anzahl der gepufferten Telegramme den Wert "1" parametrieren und den Überschreibschutz deaktivieren.

Hinweis

Wird das ständige Auslesen der Empfangsdaten im Anwenderprogramm für eine Zeit lang ausgesetzt, kann es beim erneuten Anfordern der Empfangsdaten dazu kommen, dass vom Kommunikationsmodul zunächst ein altes Telegramm und dann erst das neueste Telegramm von der CPU empfangen wird. Das alte Telegramm wurde dann zum Zeitpunkt der Unterbrechung bereits aus dem Empfangspuffer des Kommunikationsmoduls übernommen und für die Übertragung zur CPU vorbereitet.

Kommunikation über 3964(R)

Vorgehen zum Aufbau einer seriellen Verbindung mit 3964(R)

Voraussetzungen

- Die Hardware ist aufgebaut und es besteht eine elektrische Verbindung zum Koppelpartner.
- Das Projekt ist in STEP 7 (TIA Portal) angelegt und die CPU ist in der Hardware-Konfiguration eingefügt.

Vorgehen - Hardware-Konfiguration

1. Fügen Sie das Kommunikationsmodul CM PtP in die Hardware-Konfiguration ein.
2. Stellen Sie die Kommunikationsparameter entsprechend dem Koppelpartner ein:
   z. B. Übertragungsgeschwindigkeit, Zeichenrahmen, Telegrammanfang und Telegrammende
   Diese Parameter werden bei jedem CPU-Start an das Kommunikationsmodul CM PtP übertragen.

Vorgehen - Programmierung

1. Legen Sie die Datenstruktur an, die die zu übertragenden Daten enthalten soll.

Senden von Daten:

1. Fügen Sie die Anweisungen aus der Bibliothek PtP Communication ein: Send_P2P zum Senden von Daten
2. Verschalten Sie die Eingangs- und Ausgangsparameter der Anweisung, z. B.:
   – HWID aus den Systemvariablen am Eingang PORT
   – Datenstruktur mit zu sendenden Daten am Eingang BUFFER

   Hinweis: Im Betrieb wird jede steigende Flanke am Eingang REQ einmal den spezifizierten Datenbereich senden. Der Baustein muss so oft aufgerufen werden, bis durch DONE angezeigt wird, dass die Daten an die Baugruppe übertragen wurden.
   Im Fehlerfall wird durch einmaliges Setzen von ERROR und einer entsprechenden Information in STATUS angezeigt, dass die Daten nicht übertragen wurden.

Empfangen von Daten:

1. Fügen Sie die Anweisungen aus der Bibliothek PtP Communication ein: Receive_P2P zum Senden von Daten
2. Verschalten Sie die Eingangs- und Ausgangsparameter der Anweisung, z. B.:
   – HWID aus den Systemvariablen am Eingang PORT
   – Datenstruktur zur Ablage der zu empfangenden Daten am Eingang BUFFER

   Hinweis: Im Betrieb wird durch einen High-Pegel am Ausgang NDR angezeigt, dass neue Daten empfangen und im spezifizierten Datenbereich abgelegt wurden. Der Baustein muss so lange aufgerufen werden, bis NDR=TRUE. Dann können die empfangenen Daten ausgewertet und der RECEIVE_P2P wieder aufrufen werden.

Optionale Ergänzungen

- Anweisungen, die mit _Config enden, können optional verwendet werden, um im laufenden Betrieb vom Anwenderprogramm die Parameter der Hardware-Konfiguration zu ändern. Die Änderungen werden nicht in der Hardware-Konfiguration gespeichert. Sie werden beim nächsten Neustart wieder überschrieben.
- Die Anweisungen Signal_Set und Signal_Get können verwendet werden, um die RS232-Begleitsignale individuell zu steuern, falls eine automatische Bedienung nicht geeignet ist.

Datenübertragung mit der Prozedur 3964(R)

Einleitung

Die Prozedur 3964(R) steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen dem Kommunikationsmodul und einem Kommunikationspartner. Die Prozedur 3964(R) beinhaltet neben der Bitübertragungsschicht (Schicht 1) die Sicherungsschicht (Schicht 2).

Für die Kommunikation mit einem Kommunikationspartner stehen ihnen Anweisungen zur Verfügung (siehe Überblick über die PtP-Programmierung (Seite 1661)).

Steuerzeichen

Einleitung

Die Prozedur 3964(R) fügt bei der Datenübertragung den Informationsdaten Steuerzeichen hinzu (Sicherungsschicht). Durch diese Steuerzeichen kann der Kommunikationspartner kontrollieren, ob die Daten vollständig und fehlerfrei bei ihm angekommen sind.

Die Steuerzeichen der Prozedur 3964(R)

Die Prozedur 3964(R) wertet die folgenden Steuerzeichen aus:

STX	Start of Text	Anfang der zu übertragenden Zeichenfolge	02H
DLE	Data Link Escape	Datenübertragungsumschaltung	10H
ETX	End of Text	Ende der zu übertragenden Zeichenfolge	03H
NAK	Negative Acknowledge	negative Rückmeldung	15H
BCC	Block Check Character (nur bei 3964R)	Blockprüfzeichen	

BCC wird automatisch im Kommunikationsmodul gebildet und überwacht. Das Blockprüfzeichen wird nicht als Telegramminhalt an die CPU übertragen.

Hinweis

Wird innerhalb eines Telegramms als Informationszeichen das Zeichen DLE übertragen, so wird dieses zur Unterscheidung vom Steuerzeichen DLE beim Verbindungsaufbau und -abbau auf der Sendeleitung doppelt gesendet (DLE-Verdopplung). Der Empfänger macht die DLE-Verdopplung wieder rückgängig.

Priorität

Bei der Prozedur 3964(R) muss einem Kommunikationspartner eine höhere und dem anderen Kommunikationspartner eine niedrigere Priorität zugeordnet sein. Wenn beide Kommunikationspartner gleichzeitig mit dem Verbindungsaufbau beginnen, dann stellt der Partner mit niedriger Priorität seinen Sendeauftrag zurück.

Blockprüfzeichen

Blockprüfzeichen

Beim Übertragungsprotokoll 3964R wird die Datensicherheit durch ein zusätzlich gesendetes Blockprüfzeichen (BCC = Block Check Character) erhöht.

Das Blockprüfzeichen ist die gerade Längsparität (EXOR-Verknüpfung aller Datenbytes) eines gesendeten bzw. empfangenen Blocks. Die Bildung beginnt mit dem ersten Nutzdatenbyte (1.

Byte des Telegramms) nach dem Verbindungsaufbau und endet nach dem Zeichen DLE ETX beim Verbindungsabbau.

Hinweis

Bei einer DLE-Verdopplung wird das Zeichen DLE zweimal in die BCC-Bildung einbezogen.

Daten senden mit 3964(R)

Verbindungsaufbau beim Senden

Zum Aufbau der Verbindung sendet die Prozedur 3964(R) das Steuerzeichen STX. Antwortet der Kommunikationspartner vor Ablauf der Quittungsverzugszeit (QVZ) mit dem Zeichen DLE, geht die Prozedur in den Sendebetrieb über.

Antwortet der Kommunikationspartner mit NAK, einem beliebigen anderen Zeichen (außer DLE oder STX) oder die Quittungsverzugszeit verstreicht ohne Reaktion, wiederholt die Prozedur den Verbindungsaufbau. Nach der parametrierten Anzahl vergeblicher Aufbauversuche bricht die Prozedur den Verbindungsaufbau ab und sendet das Zeichen NAK an den Kommunikationspartner. Das Kommunikationsmodul gibt eine entsprechende Fehlermeldung aus.

Daten senden

Gelingt der Verbindungsaufbau, werden die im Ausgabepuffer des Kommunikationsmoduls enthaltenen Nutzdaten mit den gewählten Übertragungsparametern an den Kommunikationspartner gesendet (ein in den Nutzdaten erkanntes DLE wird während des Sendevorgangs verdoppelt.). Der Kommunikationspartner überwacht den zeitlichen Abstand der ankommenden Zeichen. Der Abstand zwischen zwei Zeichen darf nicht mehr als die Zeichenverzugszeit (ZVZ) betragen. Die Überwachung der Zeichenverzugszeit beginnt unmittelbar nach erfolgtem Verbindungsaufbau.

Sendet der Kommunikationspartner während einer laufenden Sendung das Zeichen NAK, bricht die Prozedur den Block ab und wiederholt ihn in der oben beschriebenen Weise, beginnend mit dem Verbindungsaufbau. Bei einem anderen Zeichen wartet die Prozedur zunächst auf den Ablauf der Zeichenverzugszeit und sendet anschließend NAK, um den Kommunikationspartner in den Ruhezustand zu bringen. Danach beginnt die Prozedur das Senden erneut mit dem Verbindungsaufbau STX.

Verbindungsabbau beim Senden

Nach dem Senden des Pufferinhalts fügt die Prozedur die Zeichen DLE, ETX und nur bei 3964R die Blockprüfsumme BCC als Endekennung an und wartet auf ein Quittungszeichen. Sendet der Kommunikationspartner innerhalb der Quittungsverzugszeit das Zeichen DLE, wurde der Datenblock fehlerfrei übernommen. Antwortet der Kommunikationspartner mit NAK, einem beliebigen anderen Zeichen (außer DLE), einem gestörten Zeichen oder die Quittungsverzugszeit verstreicht ohne Reaktion, beginnt die Prozedur das Senden erneut mit dem Verbindungsaufbau STX.

Nach Ablauf der parametrierten Anzahl von Übertragungsversuchen bricht die Prozedur das Verfahren ab und sendet NAK an den Kommunikationspartner. Das Kommunikationsmodul gibt eine entsprechende Fehlermeldung aus.

Daten empfangen mit 3964(R)

Verbindungsaufbau beim Empfangen

Im Ruhezustand, wenn kein Sendeauftrag zu bearbeiten ist, wartet die Prozedur auf den Aufbau der Verbindung durch den Kommunikationspartner.

Steht beim Verbindungsaufbau mit STX kein freier Empfangspuffer zur Verfügung, wird eine Wartezeit gestartet (Wartezeit = Quittungsverzugszeit - 10 ms, jedoch maximal 400 ms). Liegt nach dieser Zeit noch kein freier Empfangspuffer vor, erfolgt eine Fehlermeldung. Die Prozedur sendet das Zeichen NAK und geht wieder in den Ruhezustand zurück. Andernfalls sendet die Prozedur das Zeichen DLE und empfängt die Daten in der oben beschriebenen Weise.

Die Quittungsverzugszeit sollte bei beiden Kommunikationspartnern auf den gleichen Wert eingestellt sein.

Empfängt die Prozedur im Ruhezustand ein beliebiges Zeichen (außer STX oder NAK), wartet sie auf den Ablauf der Zeichenverzugszeit (ZVZ) und sendet dann das Zeichen NAK. Das Kommunikationsmodul gibt eine entsprechende Fehlermeldung aus.

Daten empfangen

Nach gelungenem Verbindungsaufbau werden die ankommenden Empfangszeichen im Empfangspuffer abgelegt. Werden zwei aufeinanderfolgende Zeichen DLE empfangen, wird nur ein Zeichen DLE in den Empfangspuffer übernommen.

Nach erfolgtem Verbindungsaufbau und jedem Empfangszeichen wird während der Zeichenverzugszeit auf das nächste Zeichen gewartet. Verstreicht die Zeichenverzugszeit ohne Empfang, wird das Zeichen NAK an den Kommunikationspartner gesendet. Das Kommunikationsmodul gibt eine entsprechende Fehlermeldung aus. Anschließend wird eine Wiederholung erwartet.

Treten während des Empfangs Übertragungsfehler auf (Rahmenfehler, Paritätsfehler usw.), so wird bis zum Verbindungsabbau weiter empfangen und dann NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet. Kann der Block auch nach der spezifizierten Anzahl von Übertragungsversuchen nicht fehlerfrei empfangen werden oder wird die Wiederholung vom Kommunikationspartner nicht innerhalb einer Blockwartezeit von 4 s gestartet, bricht die Prozedur den Empfang ab. Das Kommunikationsmodul meldet die erste fehlerhafte Übertragung und den endgültigen Abbruch.

Verbindungsabbau beim Empfangen

Erkennt die Prozedur 3964 die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei empfangenen Block an den Kommunikationspartner. Bei einem Empfangsfehler wird NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet.

Erkennt die Prozedur 3964R die Zeichenfolge DLE ETX BCC, beendet sie den Empfang. Sie vergleicht das empfangene Blockprüfzeichen BCC mit der intern gebildeten Längsparität. Ist

das Blockprüfzeichen korrekt und kein anderer Empfangsfehler aufgetreten, sendet die Prozedur 3964R DLE und kehrt zurück in den Ruhezustand. Das Kommunikationsmodul informiert die Steuerung, dass neue Empfangsdaten vorliegen.

Bei fehlerhaftem BCC oder einem anderen Empfangsfehler wird NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet.

Kommunikation über Modbus RTU

Vorgehen zum Aufbau einer seriellen Verbindung mit Modbus RTU

Voraussetzungen

- Die Hardware ist aufgebaut und es besteht eine elektrische Verbindung zum Koppelpartner.
- Das Projekt ist in STEP 7 (TIA Portal) angelegt und die CPU ist in der Hardware-Konfiguration eingefügt.

Vorgehen - Hardware-Konfiguration

1. Fügen Sie das Kommunikationsmodul CM PtP in die Hardware-Konfiguration ein.
2. Wählen Sie das Protokoll Modbus RTU aus.
   Hinweis: Bei Modbus RTU werden die meisten Kommunikationsparameter mithilfe der Anweisung Modbus_Comm_Load beim Anlauf der CPU eingestellt.

Vorgehen - Programmierung

1. Legen Sie die Datenstruktur an, die die zu übertragenden Daten enthalten soll.
2. Binden Sie die Anweisung Modbus_Comm_Load in den zyklischen Ablauf zur Parametrierung des Kommunikationsmoduls ein.
3. Verschalten Sie die HWID aus den Systemvariablen am Eingang PORT.
4. Rufen Sie die Anweisung so oft auf, bis am Ausgang DONE die erfolgreiche Ausführung angezeigt wird. Rufen Sie die Anweisung danach nicht wieder auf, es sei denn, sie möchten die Kommunikationsparameter ändern.

Betrieb als Modbus Master:

1. Fügen Sie die Anweisung Modbus_Master aus der Bibliothek MODBUS (RTU) ein.
2. Verschalten Sie die Datenstruktur mit den zu sendenden Daten am Eingang BUFFER.
3. Verschalten Sie den Instanz-DB der Anweisung Modbus_Master am Eingang MB_DB des Modbus_Comm_Load.
   Hinweis: Im Betrieb wird jede steigende Flanke am Eingang REQ einmal den spezifizierten Auftrag abarbeiten. Der Baustein muss so oft aufgerufen werden, bis durch DONE angezeigt wird, dass die Daten an die Baugruppe übertragen wurden.
   Im Fehlerfall wird durch einmaliges Setzen von ERROR und einer entsprechenden Information in STATUS angezeigt, dass die Daten nicht übertragen wurden.

Betrieb als Modbus Slave:

1. Fügen Sie die Anweisung Modbus_Slave aus der Bibliothek MODBUS (RTU) ein.
2. Verschalten Sie die Datenstruktur mit den Modbus Halteregistern.
3. Geben Sie die Modbus Slave-Adresse am Eingang MB_ADDR an.
4. Verschalten Sie den Instanz-DB der Anweisung Modbus_Master am Eingang MB_DB des Modbus_Comm_Load.
   Hinweis: Im Betrieb wird durch einen High-Pegel am Ausgang NDR angezeigt, dass neue Daten empfangen und im spezifizierten Datenbereich abgelegt wurden.

Überblick über Modbus-Kommunikation

Modbus RTU-Kommunikation

Modbus RTU (Remote Terminal Unit) ist ein Standardprotokoll für die Kommunikation im Netzwerk und verwendet die elektrische RS232- oder RS422/485-Verbindung für die serielle Datenübertragung zwischen Modbus-Geräten im Netzwerk.

Modbus RTU nutzt ein Master/Slave-Netzwerk, in dem die gesamte Kommunikation von einem einzigen Master-Gerät ausgelöst wird, während die Slaves lediglich auf die Anforderung des Masters reagieren können. Der Master sendet eine Anforderung an eine Slave-Adresse und nur diese Slave-Adresse antwortet auf den Befehl (Ausnahme: Broadcast Telegramme auf Slave-Adresse 0, die nicht von den Slaves quittiert werden).

Zum Einsatz kommt eine codetransparente, asynchrone Halbduplexprozedur. Die Datenübertragung erfolgt ohne Handshake.

Einordnen in die Systemumgebung

Die nachfolgende Modbus Beschreibung bezieht sich auf den Einsatz der entsprechenden Kommunikationsmodule.

- CM PtP RS232 HF
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Funktion der Kopplung

Mit den entsprechenden Kommunikationsmodulen und den zugehörigen Anweisungen wird eine Kommunikationsverbindung zwischen einem remoten Modbus-Steuerungssystem und einer SIMATIC S7 ermöglicht.

Zur Übertragung wird das GOULD - MODBUS - Protokoll im RTU-Format verwendet.

Bei der Kommunikation zwischen einem als Modbus-Slave betriebenen Kommunikationsmodul und einem Master-System können die Funktionscodes 01, 02, 03, 04, 05, 06, 08, 15 und 16 verwendet werden (siehe Funktionscodes (Seite 1657)).

Wird ein SIMATIC S7 Kommunikationsmodul als Modbus-Master betrieben, stehen zusätzlich die Funktionscodes 11 und 12 zur Verfügung.

SIMATIC S7 als Modbus-Slave

Der Master hat bei der Übertragung die Initiative, das Kommunikationsmodul arbeitet als Slave.

Ein Telegrammverkehr von Slave zu Slave ist nicht möglich.

Die Anweisung Modbus_Slave stellt die Daten entsprechend der Abbildungsvorschrift auf einen SIMATIC-Datenbereich zur Verfügung oder legt sie ab.

SIMATIC S7 als Modbus-Master

Das Kommunikationsmodul initiiert als Master die Übertragung und wartet nach Ausgabe des Anforderungstelegramms die parametrierte Antwortüberwachungszeit auf ein Antworttelegramm vom Slave. Antwortet der Slave nicht, wiederholt der Master, entsprechend der Parametrierung, die Anforderung bevor er eine Fehlermeldung zurückgibt.

Telegrammaufbau

Der Datenverkehr "Master-Slave" bzw. "Slave-Master" beginnt mit der **Slave-Adresse**, gefolgt vom **Funktionscode**. Dann werden die Daten übertragen. Der Aufbau des Datenfeldes ist abhängig vom verwendeten Funktionscode. Am Ende des Telegramms wird der CRC-Check übertragen.

ADDRESS	FUNCTION	DATA	CRC-CHECK
Byte/Wort	Byte	n Byte	2 Byte

ADDRESS	Modbus-Slave-Adresse • Standardadresse: 1 bis 247 (Byte) • Erweiterte Stationsadresse: 1 bis 65535 (Wort)
FUNCTION	Modbus- Funktionscodes (Seite 1657)
DATA	Telegrammdaten: vom Funktionscode abhängige Verwaltungs- und Nettodaten
CRC-CHECK	Telegramm-Prüfsumme

Slave-Adresse

Die Slave-Adresse kann im Bereich von 1 bis 247 (Byte) bzw. 1 bis 65535 (Wort) liegen. Mit der Adresse wird ein definierter Slave am Bus angesprochen.

Broadcast Message

Mit der Slave-Adresse Null spricht der Master alle Slaves am Bus an.

Broadcast-Messages sind nur in Verbindung mit den schreibenden Funktionscodes 05, 06, 15 bzw. 16 zulässig.

Bei einer Broadcast-Message wird vom Slave kein Antworttelegramm gesendet.

Datenfeld DATA

Im Datenfeld DATA werden die funktionscode-spezifischen Daten übertragen:

- Bytecount, Coil_Startaddress, Register_Startaddress; Number_of_Coils, Number_of_Registers, ...
  Siehe hierzu "Funktionscodes (Seite 1657)".

CRC-Check

Den Telegrammabschluss bildet die zwei Byte lange CRC-16-Prüfsumme, gebildet nach dem Polynom $x^{16} + x^{15} + x^2 + 1$.

Zuerst wird das Low-Byte, dann das High-Byte übertragen.

Telegrammende

Das Telegrammende wird erkannt, wenn während der Zeit, die dreieinhalb Zeichen zur Übermittlung benötigen (3,5-fache Zeichenverzugszeit), keine Übertragung stattfindet (siehe Modbus Protocol Reference Guide).

Dieser Telegrammende-TIME_OUT ist somit abhängig von der Datenübertragungsrate und wird in Bitzeiten angegeben (intern fest codiert sind 35 Bitzeiten; an der Anweisung können additiv weitere Bitzeiten parametriert werden).

Nach Eintreffen des Telegrammende-TIME_OUT wird das vom Koppelpartner empfangene Modbus-Telegramm ausgewertet und formal geprüft.

Exception Responses

Wenn der Slave einen Fehler im Anforderungstelegramm des Masters erkennt, z. B. Registeradresse nicht erlaubt, setzt er das höchstwertige Bit im Funktionscode des Antworttelegramms.

Nachfolgend wird ein Byte Exception Code übertragen, das die Fehlerursache beschreibt.

Sinn und Bedeutung der genannten Parameter entnehmen Sie der Beschreibung des "GOULD MODICON Modbus Protocol" (nicht Bestandteil dieser Dokumentation).

Exception Code Telegramm

Das Exception Code Telegramm vom Slave hat folgenden Aufbau:

- z. B. Slave-Adresse 5, Funktionscode 5, Exception Code 2

Antworttelegramm vom Slave EXCEPTION_CODE_xx:

05H	Slave-Adresse
85H	Funktionscode
02H	Exception Code (1...7)
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Wenn der Treiber ein Exception Code Telegramm empfängt, wird der laufende Auftrag mit Fehler beendet.

Folgende Fehlercodes sind gemäß Modbus-Spezifikation definiert:

Fehlercode	Bedeutung nach Modbus-Spezifikation	Ursache - Kurzbeschreibung *
1	Illegal function	Unzulässiger Funktionscode
2	Illegal data address	Unzulässige Daten-Adresse beim Slave
3	Illegal data value	Unzulässiger Daten-Wert beim Slave
4	Failure in associated device	Interner Fehler beim Slave
5	Acknowledge	Funktion wird ausgeführt
6	Busy, Rejected message	Slave ist nicht empfangsbereit
7	Negative acknowledgement	Funktion nicht ausführbar
* Detaillierte Ursache beim Slave untersuchen.		

RS232-Betrieb

Der RS232-Betrieb ist mit folgenden Kommunikationsmodulen möglich:

- CM PtP RS232 HF
- CM PtP (ET 200SP)

Informationen zum RS232-Betrieb finden Sie in Kapitel RS232-Betrieb (Seite 1623).

Informationen zur Hardware-Datenflusskontrolle und zur Automatischen Bedienung der Begleitsignale finden Sie im Kapitel Handshake-Verfahren (Seite 1632).

RS422/485-Betrieb

Der RS422/485-Betrieb ist mit folgenden Kommunikationsmodulen möglich:

- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Informationen zum RS422/485-Betrieb finden Sie in den Kapiteln RS422-Betrieb (Seite 1626) und RS485-Betrieb (Seite 1629).

FAQ

Weitere Informationen hierzu finden Sie im folgenden FAQ im Siemens Industry Online Support:

- Beitrags-ID 68202723 (https://support.industry.siemens.com/cs/ww/de/view/68202723)
- Beitrags-ID 58386780 (https://support.industry.siemens.com/cs/ww/de/view/58386780)

Funktionscodes

Verwendete Funktionscodes

Der Funktionscode definiert die Bedeutung des Telegramms. Durch den Funktionscode ist auch der Telegrammaufbau festgelegt.

Folgende Funktionscodes werden vom Kommunikationsmodul unterstützt:

Funtionscode	Funktion gemäß MODBUS-Spezifikation	Bereich
01	Read Coil Status	1 bis 2000 Bit/Anforderung
02	Read Input Status	1 bis 2000 Bit/Anforderung
03	Read Holding Registers	1 bis 124/125 Bit/Anforderung (124 bei erweiterter Stationsadresse)
04	Read Input Registers	1 bis 124/125 Bit/Anforderung (124 bei erweiterter Stationsadresse)
05	Force Single Coil	1 Bit/Anforderung
06	Preset Single Register	1 Wort/Anforderung
08 *	Loop Back Test	Slave Status lesen bzw. Ereigniszähler im Slave rücksetzen
11 *	Fetch Communications Event Counter (nur Master)	-
15	Force Multiple Coils	1 bis 1968 Bits/Anforderung
16	Preset Multiple Registers	1 bis 123 Bit/Anforderung

* Diagnoseinformationen für die Kommunikation des Slave

Modbus-Funktionscode 00 sendet eine Broadcast-Meldung an alle Slaves (ohne Slave-Antwort).

Zuordnung der Modbus- zu den SIMATIC-Adressen

Die folgende Tabelle zeigt die Zuordnung der Modbus-Adressen zu den SIMATIC-Adressen.

Modbus				S7-1500	
FC [1]	Funktion	Deklaration	Adressbereich	Deklaration	CPU-Adresse
01	Bits lesen	Output	1 - 9999	Prozessabbild Ausgänge	A0.0 - A1249.6
02	Bits lesen	Input	10001 - 19999	Prozessabbild Eingänge	E0.0 - E1249.6
03 [2]	Wörter lesen	Holding Register	40001 - 49999 oder 400001 - 465535	DW0 - DW19998 oder DW0 - DW131068	Der M-Adressbereich ist CPU-abhängig
04	Wörter lesen	Input	30001 - 39999	Prozessabbild Eingänge	EW0 - EW19996
05 [2]	Bits schreiben	Output	1 - 9999	Prozessabbild Ausgänge	A0.0 - A1248.7
06	Wörter schreiben	Holding Register	40001 - 49999 oder 400001 - 465535	DW0 - DW19998 oder DW0 - DW131068	Der M-Adressbereich ist CPU-abhängig

Modbus				S7-1500	
FC [1]	Funktion	Deklaration	Adressbereich	Deklaration	CPU-Adresse
15	Bits schreiben	Output	1 - 9999	Prozessabbild Ausgänge	A0.0 - A1249.6
16 [2]	Wörter schreiben	Holding Register	40001 - 49999 oder 400001 - 465535	DW0 - DW19998 oder DW0 - DW131068	Der M-Adressbereich ist CPU-abhängig

1) FC = Funktionscode

2) Ob mit den FC's 03, 05 und 16 Daten- oder Merkerbereiche in der SIMATIC CPU adressiert werden, wird mit dem Wert von HR_Start_Offset bestimmt.

Kommunikation über USS

Vorgehen zum Aufbau einer seriellen Verbindung mit Modbus USS

Voraussetzungen

- Die Hardware ist aufgebaut und es besteht eine elektrische Verbindung zum Koppelpartner.
- Das Projekt ist in STEP 7 (TIA Portal) angelegt und die CPU ist in der Hardware-Konfiguration eingefügt.

Vorgehen - Hardware-Konfiguration

1. Fügen Sie das Kommunikationsmodul CM PtP in die Hardware-Konfiguration ein.
2. Wählen Sie das Protokoll Freeport aus und stellen Sie die Kommunikationsparameter ein.
   Hinweis: Die USS-Funktionalität wird durch die Anweisungen realisiert.

Vorgehen - Programmierung

1. Fügen Sie die Anweisung USS_Port_Scan aus der Bibliothek USS Communication ein.
2. Verschalten Sie die HWID aus den Systemvariablen am Eingang PORT.
3. Fügen Sie die Anweisung USS_Drive_Control aus der Bibliothek USS Communication ein.
4. Verschalten Sie die Datenstruktur USS_DB im Instanz-DB der Anweisung USS_Drive_Control an den Eingang USS_DB der Anweisung USS_Port_Scan. Die Datenstruktur enthält die zu übertragenden Daten für alle Antriebe.
5. Fügen Sie für jede weitere Achse, die über diese USS-Schnittstelle angebunden werden soll, jeweils einen weiteren Aufruf der Anweisung USS_Drive_Control ein.
   Verwenden jedes Mal den gleichen Instanz-DB. Die Unterscheidung erfolgt mit Hilfe der USS-Adresse, die Sie am Eingang DRIVE der Anweisung USS_Drive_Control spezifizieren. So stehen Ihnen für jeden Antrieb die Steuer- und die Rückmelde-Daten an den Parametern des jeweiligen Aufrufs zur Verfügung.

Überblick über USS-Kommunikation

Einordnen in die Systemumgebung

Die nachfolgende USS-Beschreibung bezieht sich auf den Einsatz der entsprechenden Kommunikationsmodule.

- CM PtP RS232 BA
- CM PtP RS422/485 BA
- CM PtP RS232 HF
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Einleitung

Das USS®-Protokoll (Universelles-serielles-Schnittstellen-Protokoll) ist ein einfaches serielles Datenübertragungsprotokoll, das auf die Anforderungen in der Antriebstechnologie zugeschnitten ist.

Das USS-Protokoll definiert eine Zugriffstechnik entsprechend dem Master-Slave-Prinzip für die Kommunikation über einen seriellen Bus. An den Bus können ein Master und bis zu 16 Antriebe (Slaves) angeschlossen werden. Die einzelnen Antriebe werden vom Master mit Hilfe eines Adresszeichens im Telegramm gewählt. Ein Antrieb kann nie senden, ohne zuvor vom Master angestoßen worden zu sein. Es ist also keine direkte Datenübertragung zwischen einzelnen Antrieben möglich. Die Kommunikation arbeitet im Halbduplexbetrieb. Die Master-Funktion kann nicht übertragen werden.

Die Antriebstechnik fordert für die zu bewältigenden Steuerungs- und Regelungsaufgaben definierte Reaktionszeiten und damit einen streng zyklischen Telegrammverkehr:

Der Master sendet kontinuierlich Telegramme (Auftragstelegramme) an die Antriebe und erwartet jeweils ein Antworttelegramm vom angesprochenen Antrieb.

Ein Antrieb muss ein Antworttelegramm senden, wenn

- er ein Telegramm fehlerfrei empfangen hat und
- er in diesem Telegramm adressiert wurde.

Ein Antrieb darf nicht senden, wenn diese Bedingungen nicht erfüllt sind oder der Antrieb im Broadcast angesprochen wurde.

Für den Master besteht die Verbindung zu den betreffenden Antrieben dann, wenn er nach einer definierten Bearbeitungszeit (Antwortverzugszeit) vom Antrieb ein Antworttelegramm erhält.

Telegrammaufbau

Jedes Telegramm beginnt mit dem Startzeichen (STX), gefolgt von der Längenangabe (LGE) und dem Adressbyte (ADR). Danach folgt das Datenfeld. Das Telegramm endet mit dem Blockprüfzeichen (BCC). Zur Telegrammlänge zählen die Nutzdaten (Anzahl n), das Adressbyte (ADR) und das Datensicherungszeichen (BCC).

STX	LGE	ADR	1	2	...	N	BCC

Bei Daten, die aus einem Wort (16 Bit) bestehen, wird zuerst das High-Byte, gefolgt vom Low-Byte gesendet. Bei Doppelwortdaten wird dementsprechend zuerst das High-Word und anschließend das Low-Word gesendet. Die Längenangabe eines Telegramms erfolgt in Byte.

Datenverschlüsselung

Die Daten werden folgendermaßen verschlüsselt:

- STX: 1 Byte, Textbeginn, 02H
- LGE: 1 Byte, enthält die Telegrammlänge als Binärzahl
- ADR: 1 Byte, enthält die Slave-Adresse und den Telegrammtyp, binärcodiert
- Datenfelder: Je ein Byte; Inhalt abhängig vom Auftrag
- BCC: 1 Byte, Blockprüfzeichen

Datenübertragungsprozedur

Der Master sorgt für die zyklische Datenübertragung in Telegrammen. Der Master adressiert alle Slave-Teilnehmer nacheinander mit einem Auftragtelegramm. Die angesprochenen Teilnehmer antworten mit einem Antworttelegramm. Entsprechend der Master-Slave-Prozedur muss der Slave das Antworttelegramm an den Master senden, nachdem er das Auftragtelegramm empfangen hat. Erst dann kann der Master den nächsten Slave ansprechen.

Datenfeld im Telegrmm

Das Datenfeld ist in zwei Bereiche unterteilt: Parameter (PKW) und Prozessdaten (PZD).

STX	LGE	ADR	Parameter (PKW)	Prozessdaten (PZD)	BCC

- **Parameterbereich** (PKW)
  Der PKW-Bereich wickelt die Parameterübertragung zwischen zwei Kommunikationspartnern ab (z. B. Steuerung und Antrieb). Dies umfasst beispielsweise das Lesen und Schreiben von Parameterwerten und das Lesen von Parameterbeschreibungen und zugehörigem Text. Die PKW-Schnittstelle beinhaltet im Allgemeinen Aufträgen für Bedienung und Anzeigen, Wartung und Diagnose.
- **Prozessdatenbereich** (PZD)
  Der PZD-Bereich umfasst Signale, die für die Automatisierung erforderlich sind:
  - Steuerworte und Sollwerte vom Master an den Slave
  - Statusworte und Istwerte vom Slave an den Master

  Die Inhalte des Parameterbereichs und des Prozessdatenbereichs werden von den Slave-Antrieben definiert.
  Weitere Informationen hierzu finden Sie in der Dokumentation des Antriebs.

Funktionsübersicht

Ablauf der Übertragung

Die Anweisungen wickeln die Datenübertragung zyklisch mit bis zu 16 Antriebs-Slaves ab. Es ist pro Antrieb jeweils nur ein Auftrag aktiv.

Leistungsmerkmale:

- Erstellung von Datenbereichen für die Kommunikation je nach Buskonfiguration
- Ausführung und Überwachung von PKW-Aufträgen
- Überwachung des Gesamtsystems und Fehlerbehandlung
- Kommunikation mit der CPU
- Zugriff auf die Funktionen des Antriebs
- Lesen der Antriebsparameter
- Schreiben der Antriebsparameter

Programmieren - Kommunikation über Anweisungen

Überblick über die PtP-Programmierung

Datenaustausch über Freeport- bzw. 3964 (R)-Kommunikation

Die Sendedaten müssen Sie im Anwenderprogramm der zugehörigen CPU in Datenbausteinen oder im Merkerbereich bereitstellen. Für die Empfangsdaten steht im Kommunikationsmodul ein Empfangspuffer zur Verfügung. In der CPU wird ein entsprechender Datenbaustein eingerichtet.

Im Anwenderprogramm der CPU übernehmen die folgenden Anweisungen den Datentransfer zwischen CPU und Kommunikationsmodul.

- Send_P2P
- Receive_P2P

Mit der Anweisung Receive_Reset kann der Empfangspuffer gelöscht werden.

Dynamische Projektierung über das Anwenderprogramm

Alternativ bzw. ergänzend zu den in Kapitel Konfigurieren/Parametrieren eines Kommunikationsmoduls (Seite 1637) beschriebenen Parametrierung der Schnittstelle des Kommunikationsmoduls kann es in bestimmten Anwendungsbereichen vorteilhaft sein, die Kommunikation dynamisch, d. h. über eine spezifische Applikation programmgesteuert, einzurichten.

Alle im Eigenschaftendialog des Kommunikationsmoduls durchgeführten Parametrierungen können auch zur Laufzeit mithilfe einer der folgenden "Config"-Anweisung geändert werden:

Port_Config, Send_Config, Receive_Config, P3964_Config

Programmaufrufe für die PtP-Kommunikation - Ablauf

Die folgende Grafik veranschaulicht die Funktion der PtP-Anweisungen für den Ablauf der Kommunikation zwischen Anwenderprogramm und Kommunikationspartner.

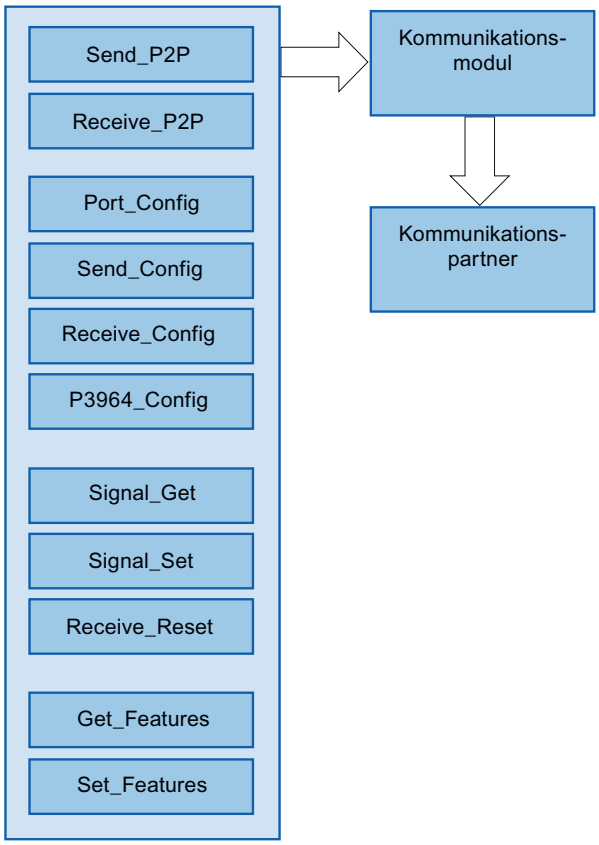

PtP-Anweisungen

Anwendung	Anweisung	Beschreibung
Datenaustausch zwischen CPU, Kommunikationsmodul und Kommunikationspartner (Kommunikation)	Send_P2P	Mit der Anweisung Send_P2P (Punkt-zu-Punkt-Daten senden) können Sie Daten an den Kommunikationspartner senden. Zum Senden von Daten mit dem Freeport-Protokoll rufen Sie die Anweisung Send_P2P auf. Sie müssen die Anweisung so lange zyklisch aufrufen, bis Sie eine entsprechende Quittung an den Ausgangsparametern der Anweisung erhalten. Hinweis: Bei der Parametrierung der Datenflusskontrolle XON/XOFF dürfen die Nutzdaten keines der parametrierten XON- oder XOFF-Zeichen enthalten. Die Default-Einstellungen sind DC1 = 11H für XON und DC3 = 13H für XOFF.
	Receive_P2P	Mit der Anweisung Receive_P2P (Punkt-zu-Punkt-Daten empfangen) können Sie die Nachrichten abholen, die im Kommunikationsmodul von einem Kommunikationspartner empfangen wurden. Zum Empfangen von Daten mit dem Freeport-Protokoll rufen Sie zyklisch die Anweisung Receive_P2P auf. Die Anweisung zeigt am Parameter NDR an, ob neue Empfangsdaten vorliegen. Um dem Empfänger zu signalisieren, wann die Übertragung einer Nachricht beginnt und wann die Nachrichtenübertragung beendet ist, müssen im Freeport-Protokoll Kriterien festgelegt werden, die den Anfang und das Ende einer Nachricht kennzeichnen.
Löschung des Empfangspuffers	Receive_Reset	Die Anweisung Receive_Reset (Empfangspuffer löschen) ermöglicht Ihnen, den Empfangspuffer des Kommunikationsmoduls zu löschen.
Dynamische Parametrierung der Schnittstelle oder des Protokolls (optional)	Port_Config	Mit der Anweisung Port_Config (Port-Konfiguration) können Sie grundlegende Schnittstellenparameter wie die Datenübertragungsgeschwindigkeit, Parität und Datenflusskontrolle über Ihr Anwenderprogramm dynamisch parametrieren.
	Send_Config	Mit der Anweisung Send_Config (Sendeparametrierung) können Sie serielle Sendeparameter wie RTS ON-/RTS OFF-Verzögerung für eine PtP-Kommunikationsschnittstelle dynamisch parametrieren.
	Receive_Config	Mit der Anweisung Receive_Config (Empfangsparametrierung) könne Sie serielle Empfangsparameter eines Kommunikationsmoduls dynamisch parametrieren. Diese Anweisung parametriert die Bedingungen, die den Beginn und das Ende einer empfangenen Nachricht kennzeichnen
	P3964_Config	Mit der Anweisung P3964_Config (Protokollkonfiguration) können Sie Protokollparameter der Prozedur 3964(R) wie Zeichenverzugszeit, Priorität und Blockcheck über ihr Programm dynamisch parametrieren.
Bedienung der RS232-Begleitsignale	Signal_Get	Mit der Anweisung Signal_Get (RS232-Signale abrufen) können Sie die aktuellen Zustände der RS232-Signale lesen.
	Signal_Set	Mit der Anweisung Signal_Set (RS232-Signale abrufen) können Sie die Zustände der RS232-Signale DTR und RTS setzen.
Modbus CRC-Unterstützung und Diagnosealarm aktiviern	Get_Features	Mit der Anweisung Get_Features (Erweiterte Funktionen holen) können Informationen über die Modbus-Unterstützung und zur Generierung von Diagnosealarmen abgerufen werden.
	Set_Features	Mit der Anweisung Set_Features (Erweiterte Funktionen einstellen) können Sie, sofern vom Modul unterstützt, die Generierung von Diagnosealarmen aktiviert werden.

Vorgehen zum Einrichten von Freeport- bzw. 3964(R)-Kommunikation

Voraussetzung: Die Konfiguration und Parametrierung einer CPU und eines Kommunikationsmoduls in der Gerätesicht und im Eigenschaften-Dialog des Kommunikationsmoduls sind abgeschlossen.

1. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den Main (OB1) durch Doppelklick. Der Programmeditor öffnet sich.
2. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation" die Anweisungen Send_P2P und Receive_P2P und ziehen Sie sie per Drag & Drop in ein Netzwerk des Main (OB1).
3. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
4. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Überblick über die Modbus-Programmierung

Programmaufrufe für die Modbus-Kommunikation - Ablauf

Die folgende Grafik veranschaulicht die Funktion der Modbus-Anweisungen für den Ablauf der Kommunikation zwischen Anwenderprogramm und Modbus-Gerät. (Die Anweisungen Send_P2P, Receive_P2P und die Config-Anweisungen werden unterlagert benötigt).

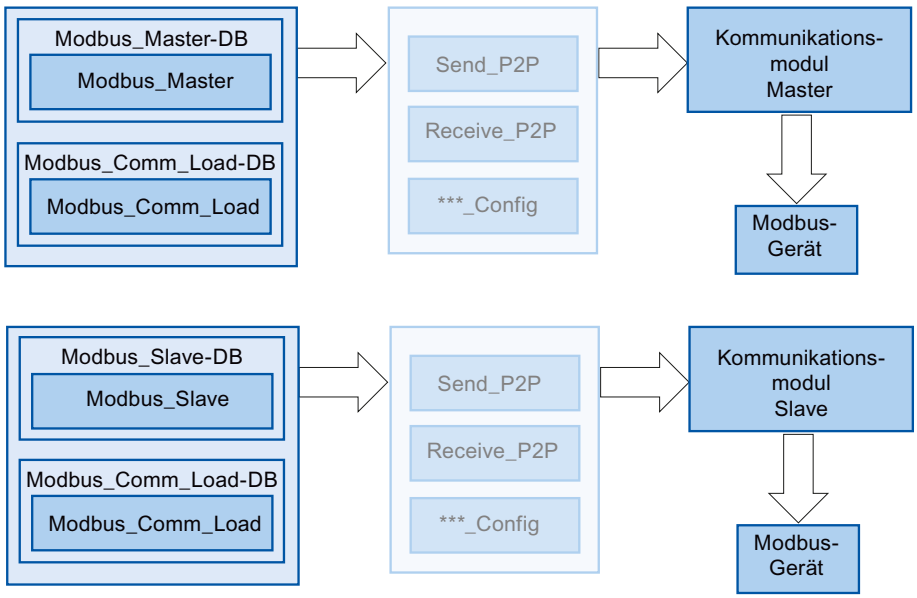

Modbus-Anweisungen

Anwendung	Anweisung	Beschreibung
Datenaustausch zwischen Anwenderprogramm und Modbus-Gerät (Kommunikation)	Modbus_Master	Die Anweisung Modbus_Master ermöglicht Ihnen, als Modbus-Master über den PtP-Port zu kommunizieren. Mit der Modbus_Master-Anweisung kann die CPU als Modbus-RTU-Master-Gerät für die Kommunikation mit einem oder mehreren Modbus-Slave-Geräten eingesetzt werden.
	Modbus_Slave	Die Anweisung Modbus_Slave ermöglicht Ihnen, als Modbus-Slave über den PtP-Port zu kommunizieren. Mit der Modbus_Slave-Anweisung kann die CPU als Modbus-RTU-Slave-Gerät für die Kommunikation mit einem Modbus-Master-Gerät eingesetzt werden.
Parametrierung der Schnittstelle und des Protokolls(obligatorisch)	Modbus_Comm_Load	Die Anweisung Modbus_Comm_Load ermöglicht Ihnen, den Port des Kommunikationsmoduls für Modbus-RTU zu konfigurieren. Eine Ausführung von Modbus_Comm_Load ist erforderlich, um PtP-Portparameter wie Datenübertragungsgeschwindigkeit, Parität und Flusskontrolle einzurichten. Nachdem die Schnittstelle für das Modbus RTU-Protokoll konfiguriert ist, kann sie nur von der Anweisung Modbus_Master oder der Anweisung Modbus_Slave verwendet werden.

Hinweis

Alternative Verwendung von Modbus_Slave und Modbus_Master

Ein Kommunikationsmodul kann entweder als Master oder als Slave betrieben werden.

Vorgehen zum Einrichten von Modbus-Kommunikation

Voraussetzung: Die Konfiguration und Parametrierung einer CPU und eines Kommunikationsmoduls in der Gerätesicht und im Eigenschaften-Dialog des Kommunikationsmoduls sind abgeschlossen.

1. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den Main (OB1) durch Doppelklick. Der Programmeditor öffnet sich.
2. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation" die Anweisungen für Modbus-Kommunikation entsprechend Ihrer Aufgabenstellung und ziehen Sie sie per Drag & Drop in ein Netzwerk des Main (OB1):
   - Die Anweisung Modbus_Comm_Load konfiguriert den Port des CM für die Modbus-Kommunikation.
     Die Anweisung Modbus_Comm_Load muss solange im Main (OB1) aufgerufen werden, bis DONE (oder ERROR) gemeldet wird.
   - Die Anweisung Modbus_Master wird eingesetzt für die Modbus-Master-Funktionalität.
   - Die Anweisung Modbus_Slave wird eingesetzt für die Modbus-Slave-Funktionalität.
3. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
4. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Siehe auch

Übersicht der Anweisungen (Seite 1618)

Überblick über die USS-Programmierung

Programmaufrufe für die USS-Kommunikation - Ablauf

Die folgende Grafik veranschaulicht die Funktion der USS-Anweisungen für den Ablauf der Kommunikation zwischen Anwenderprogramm und USS-Antrieb. (Die Anweisungen Send_P2P, Receive_P2P und die Config-Anweisungen werden unterlagert benötigt).

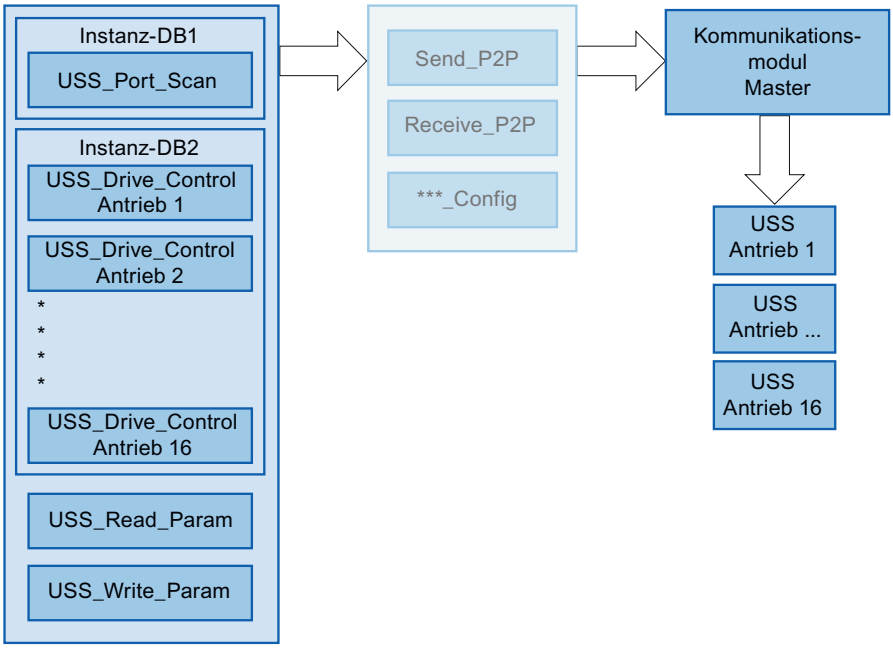

USS-Anweisungen

Anwendung	Anweisung	Beschreibung
Datenkommunikation zwischen CPU, Kommunikationsmodul und USS-Antrieb	USS_Port_Scan	Die Anweisung USS_Port_Scan ermöglicht Ihnen die Kommunikation über ein Kommunikationsmodul mit bis zu 16 Antrieben über ein USS-Netzwerk (muss zyklisch aufgerufen werden). Die Anweisung USS_Port_Scan steuert die Kommunikation zwischen CPU und den Antrieben über den PtP-Kommunikationsport. Bei jedem Aufruf dieser Funktion wird eine Kommunikation mit einem Antrieb bearbeitet. Die Anweisung USS_Port_Scan wird einmal benötigt. Da die meistern Antriebe über eine parametrierbare interne Funktion verfügen, die anhand eines Timeout die Integrität der Kommunikation überwacht, sollte die Anweisung USS_Port_Scan aus einem zeitgesteuerten OB aufgerufen werden.
Daten mit USS-Antrieb auszutauschen	USS_Drive_Control	Die Anweisung USS_Drive_Control ermöglicht Ihnen, die Sendedaten für einen Antrieb aufzubereiten und die Empfangsdaten anzuzeigen. Die Ein- und Ausgänge der Anweisung entsprechen den Zuständen und den Bedienfunktionen des Antriebs. Die Anweisung USS_Drive_Control muss je Antrieb einmal aufgerufen werden. Für alle Aufrufe der Anweisung USS_Drive_Control zu einem USS-Netzwerk wird nur ein gemeinsamer Instanz-DB benötigt. Verschalten Sie alle Aufrufe der Anweisung USS_Drive_Control für ein USS-Netzwerk mit dem selben Instanz-DB. Die Anweisung USS_Drive_Control sollte aus dem zyklischen Main (OB1) des Hauptprogramms aufgerufen werden.
Parameter im USS-Antrieb auslesen oder ändern	USS_Read_Param	Die Anweisung USS_Read_Param ermöglicht Ihnen, Parameter aus dem Antrieb auszulesen. Mit der Anweisung USS_Read_Param werden die Betriebsparameter des Antriebs gelesen, die die internen Funktionen des Antriebs steuern. Die Anweisung USS_Read_Param sollte aus dem zyklischen Main (OB1) des Hauptprogramms aufgerufen werden.
	USS_Write_Param	Die Anweisung USS_Write_Param ermöglicht Ihnen, Parameter im Antrieb zu ändern. Die Anweisung USS_Write_Param sollte wird aus dem zyklischen Main (OB1) des Hauptprogramms aufgerufen werden.

Vorgehen zum Einrichten von USS-Kommunikation

Voraussetzung: Die Konfiguration und Parametrierung einer CPU und eines Kommunikationsmoduls in der Gerätesicht und im Eigenschaften-Dialog des Kommunikationsmoduls sind abgeschlossen.

1. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie den gewünschten zeitgesteuerten OB durch Doppelklick. Der Programmeditor öffnet sich.
2. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation" die Anweisung USS_Port_Scan und ziehen Sie sie per Drag & Drop in ein Netzwerk eines zeitgesteuerten OB.
   Die Anweisung USS_Port_Scan ermöglicht die Kommunikation über das USS-Netzwerk.
3. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den Main (OB1) durch Doppelklick. Der Programmeditor öffnet sich.

4. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation" die Anweisungen für USS-Kommunikation entsprechend Ihrer Aufgabenstellung und ziehen Sie sie per Drag & Drop in ein Netzwerk des Main (OB1):
   - Die Anweisung USS_Drive_Control dient dem Datenaustausch mit dem Antrieb.
   - Die Anweisung USS_Read_Param dient dem Auslesen von Parametern aus dem Antrieb.
   - Die Anweisung USS_Write_Param dient dem Ändern von Parametern im Antrieb.
5. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
6. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Siehe auch

Übersicht der Anweisungen (Seite 1618)

Anlauf und Diagnose

Anlaufverhalten

Betriebszustandsübergänge

Nach dem Anlauf des Kommunikationsmoduls werden alle Daten zwischen CPU und dem Kommunikationsmodul über Anweisungen ausgetauscht.

CPU-STOP	Bei einer laufenden Datenübertragung Kommunikationsmodul - CPU, wird sowohl ein Sende- als auch ein Empfangsauftrag abgebrochen.
CPU-RUN	Im Zustand RUN der CPU ist der Sende– und Empfangsbetrieb gewährleistet. Durch entsprechende Parametrierung im Eigenschaften Dialog des Kommunikationsmoduls können Sie den Empfangspuffer auf dem Kommunikationsmodul im CPU-Anlauf automatisch löschen.

Aus Sicht des Kommunikationsmoduls gibt es keine weiteren Betriebszustände/ Betriebszustandsübergänge.

Diagnosefunktionen

Einleitung

Die Diagnosefunktionen des Kommunikationsmoduls erlauben Ihnen eine schnelle Lokalisierung aufgetretener Fehler. Folgende Diagnosemöglichkeiten stehen Ihnen zur Verfügung:

Diagnose über die Anzeigeelemente des Kommunikationsmoduls	Die Anzeigeelemente informieren Sie über den Betriebszustand bzw. über mögliche Fehlerzustände des Kommunikationsmoduls. Die Anzeigeelemente geben Ihnen einen ersten Überblick über aufgetretene interne bzw. externe Fehler sowie schnittstellenspezifische Fehler. Informationen hierzu finden Sie im Gerätehandbuch des entsprechenden Kommunikationsmoduls.
Diagnose über den STATUS-Ausgang der Anweisungen	Für eine Fehlerdiagnose besitzen die Anweisungen einen STATUS-Ausgang, der Ihnen Aussagen zu Kommunikationsfehlern zwischen Kommunikationsmodul und CPU liefert. Den STATUS-Parameter können Sie im Anwenderprogramm auswerten (er steht für genau einen Zyklus an).
Diagnosealarm	Das Kommunikationsmodul kann auf der ihm zugeordneten CPU einen Diagnosealarm auslösen. Das Kommunikationsmodul stellt Diagnoseinformation zur Verfügung. Die Auswertung dieser Informationen erfolgt über das Anwenderprogramm oder durch Auslesen des CPU-Diagnosepuffers.

Diagnosealarme

Die Anzeige der Diagnosen erfolgt als Klartext in STEP 7 (TIA Portal) über die Online- und Diagnosesicht. Die Fehlercodes können Sie über das Anwenderprogramm auswerten.

Folgende Diagnosen können gemeldet werden:

- Fehler (9_H)
- Parametrierfehler (10_H)
- Drahtbruch (109_H)

1.4.6 Konfigurationen für Punkt-zu-Punkt-Kopplungen (S7-300, S7-400)

1.4.6.1 Konfigurationen für Punkt-zu-Punkt-Kopplungen

Einleitung

Übersicht der Komponenten und Schnittstellen

Inhalt

Tabellarische Übersicht über die Baugruppenvarianten und ihre Funktionen.

Für CP 340/341/440/441 und ET 200S 1SI

Baugruppe	Schnittstelle	Protokolle						
		ASCII	3964(R)	RK512	Drucker-treiber	Modbus Master	Modbus Slave	USS-Master
CP 340	RS232 C	X	X	-	X	-	-	-
CP 340	20 mA TTY	X	X	-	X	-	-	-
CP 340 1)	RS422	X	X	-	X	-	-	-
	RS485	X	-	-	-	-	-	-
CP 341	RS232 C	X	X	X	X	X 2)	X 2)	-
CP 341	20 mA TTY	X	X	X	X	X 2)	X 2)	-
CP 341 1)	RS422	X	X	X	X	X 2)	X 2)	-
	RS485	X	-	-	X	X 2)	X 2)	-
CP 440	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-
CP 441-1	RS232 C	X	X	-	X	-	-	-
CP 441-1	20 mA TTY	X	X	-	X	-	-	-
CP 441-1 1)	RS422	X	X	-	X	-	-	-
	RS485	X	-	-	X	-	-	-
CP 441-2	RS232 C	X	X	X	X	X 2)	X 2)	-
CP 441-2	20 mA TTY	X	X	X	X	X 2)	X 2)	-
CP 441-2 1)	RS422	X	X	X	X	X 2)	X 2)	-
	RS485	X	-	-	X	-	-	-
ET 200S 1SI für ASCII und 3964(R)	RS232 C	X	X	-	-	-	-	-
	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-
ET 200S 1SI für Modbus und USS	RS232 C	-	-	-	-	X	X	X
	RS422	-	-	-	-	X	X	-
	RS485	-	-	-	-	X	X	X
CPU 313C-2 PtP	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-

Baugruppe	Schnittstelle	Protokolle						
		ASCII	3964(R)	RK512	Drucker-treiber	Modbus Master	Modbus Slave	USS-Master
CPU 314C-2 PtP	RS422	X	X	X	-	-	-	-
	RS485	X	-	-	-	-	-	-

1) Die Unterscheidung zwischen RS422 und RS485 wird im Parametrierdialog eingestellt.

2) Für das Modbus-Protokoll wird ein ladbarer Treiber mit einem Dongle benötigt.

Begleitsignale und Datenflusskontrolle

Die Bedienung der RS232C - Begleitsignale ist bei den Protokollen ASCII, Modbus Master und Modbus Slave über die Schnittstelle RS232C möglich.

Die Datenflusskontrolle mit RTS/CTS ist bei den Protokollen ASCII und Druckertreiber über die Schnittstelle RS232C möglich.

Die Datenflusskontrolle mit XON/XOFF ist bei den Protokollen ASCII und Druckertreiber über die Schnittstellen RS232C, 20mA TTY und RS422 möglich.

Übersicht über die Anweisungen

Übersicht über die Anweisungen

Die Kommunikation zwischen CPU, Kommunikationsbaugruppe und einem Kommunikationspartner erfolgt über die Anweisungen und die Protokolle der Kommunikationsbaugruppe.

Die Anweisungen bilden die Softwareschnittstelle zwischen der CPU und der Kommunikationsbaugruppe. Sie müssen zyklisch aus dem Anwenderprogramm aufgerufen werden.

Auf der Kommunikationsbaugruppe erfolgt die Umsetzung der Übertragungsprotokolle. Über das Protokoll wird die Schnittstelle der Kommunikationsbaugruppe an die Schnittstelle des Kommunikationspartners angepasst.

Tabelle 1-99 Anweisungen für CP 340

Anweisung	Bedeutung
P_SEND	Die Anweisung P_SEND ermöglicht Ihnen, einen gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner zu senden.
P_RCV	Die Anweisung P_RCV ermöglicht Ihnen, Daten von einem Kommunikationspartner zu empfangen und in einem Datenbaustein zu hinterlegen.
P_PRINT	Die Anweisung P_PRINT ermöglicht Ihnen, einen Meldetext mit bis zu 4 Variablen auf einen Drucker auszugeben.
P_RESET	Die Anweisung P_RESET ermöglicht Ihnen, den internen Empfangspuffer des CP 340 zurückzusetzen.
V24_STAT_340	Die Anweisung V24_STAT ermöglicht Ihnen, die Signalzustände an der RS232C-Schnittstelle des CP 340-RS232C zu lesen.
V24_SET_340	Die Anweisung V24_SET ermöglicht Ihnen, die Ausgänge an der RS232C-Schnittstelle des CP 340–RS232C zu setzen/rückzusetzen.

Tabelle 1-100 Anweisungen für CP 341

Anweisung	Bedeutung
P_SND_RK	Mit der Anweisung P_SND_RK können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner senden oder (bei Verwendung von RK512) Daten vom Kommunikationspartner holen.
P_RCV_RK	Mit der Anweisung P_RCV_RK können Sie Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen oder (bei Verwendung von RK512) Daten für den Kommunikationspartner bereitstellen.
P_PRINT341	Die Anweisung P_PRINT341 ermöglicht ihnen, einen Meldetext mit bis zu 4 Variablen auf einen Drucker auszugeben.
V24_STAT	Die Anweisung V24_STAT ermöglicht Ihnen, die Signalzustände an der RS232C-Schnittstelle des CP 341-RS232C zu lesen.
V24_SET	Die Anweisung V24_SET ermöglicht Ihnen, die Ausgänge an der RS232C-Schnittstelle des CP 341-RS232C zu setzen/rückzusetzen.

Tabelle 1-101 Anweisungen für CPU 313C-2 PTP

Anweisung	Bedeutung
SEND_PTP	Mit der Anweisung SEND_PTP können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner senden.
RCV_PTP	Mit der Anweisung RCV_PTP können Sie Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen.
RES_RCVB	Mit der Anweisung RES_RCVB können Sie den Empfangspuffer der Baugruppe zurücksetzen.

Tabelle 1-102 Anweisungen für CPU 314C-2 PTP

Anweisung	Bedeutung
SEND_PTP	Mit der Anweisung SEND_PTP können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner senden.
RCV_PTP	Mit der Anweisung RCV_PTP können Sie Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen.
RES_RCVB	Mit der Anweisung RES_RCVB können Sie den Empfangspuffer der Baugruppe zurücksetzen.
SEND_RK	Mit der Anweisung SEND_RK können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins mit dem RK512 Protokoll an einen Kommunikationspartner senden.
FETCH_RK	Mit der Anweisung FETCH_RK können Sie mit dem RK512 Protokoll Daten vom Kommunikationspartner anfordern
SERVE_RK	Mit der Anweisung SERVE_RK können Sie mit dem RK512 Protokoll Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen oder Daten für den Kommunikationspartner bereitstellen.

Tabelle 1-103 Anweisungen für CP 440

Anweisung	Bedeutung
SEND_440	Mit der Anweisung SEND_440 können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner senden.
RECV_440	Mit der Anweisung RECV_440 können Sie Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen.
RES_RECV	Mit der Anweisung RES_RECV können Sie den Empfangspuffer des CP 440 zurücksetzen.

Tabelle 1-104 Anweisungen für CP 441

Anweisung	Bedeutung
V24_STAT_441	Die Anweisung V24_STAT_441 ermöglicht Ihnen, die Signalzustände an der RS232C-Schnittstelle des CP 441 zu lesen.
V24_SET_441	Die Anweisung V24_SET_441 ermöglicht Ihnen, die Ausgänge an der RS232C-Schnittstelle des CP 441 zu setzen/rückzusetzen.
Das Automatisierungssystem S7-400 stellt Ihnen eine Reihe von Anweisungen zur Verfügung, die im Anwenderprogramm die Kommunikation zwischen CPU und dem Kommunikationsprozessor CP 441 anstoßen und steuern. Diese Anweisungen sind fest im Speicher der CPU hinterlegt (siehe unten Abschnitt " Anweisungen des Automatisierungssystems S7-400").	

Tabelle 1-105 Anweisungen für ET 200S 1SI

Anweisung	Bedeutung
S_SEND	Die Anweisung S_SEND ermöglicht Ihnen, einen gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner zu senden.
S_RCV	Die Anweisung S_RCV ermöglicht Ihnen, Daten von einem Kommunikationspartner zu empfangen und in einem Datenbaustein zu hinterlegen.
S_VSTAT	Die Anweisung S_VSTAT ermöglicht Ihnen, die Signalzustände an der RS 232C-Schnittstelle der Baugruppe ET 200S 1SI zu lesen.
S_VSET	Die Anweisung S_VSET ermöglicht Ihnen, die Ausgänge der RS 232C-Schnittstelle der Baugruppe ET 200S 1SI zu setzen/rückzusetzen.
S_XON	Mit der Anweisung S_XON können Sie zusätzliche Parameter einstellen, falls die Baugruppe für die Flusskontrolle XON/XOFF parametriert wurde.
S_RTS	Mit der Anweisung S_RTS können Sie zusätzliche Parameter einstellen, falls die Baugruppe für die Flusskontrolle RTS/CTS parametriert wurde.
S_V24	Mit der Anweisung S_V24 können Sie zusätzliche Parameter einstellen, falls die Baugruppe für die automatische Bedienung der V.24-Signale parametriert wurde.
S_MODB	Die Anweisung S_MODB ermöglicht es Ihnen eine Kommunikationsverbindung mit einem Modbus-Master-Steuerungssystem aufzubauen und die Modbus-Adressen in SIMATIC-Speicherbereiche umzusetzen.
S_USST	Die Anweisung S_USST wickelt die Übertragung der Netzdaten (PZD- und evtl. PKW-Daten) an die Slaves je nach der verwendeten Netzdatenstruktur ab.

Anweisung	Bedeutung
S_USSR	Die Anweisung S_USSR wickelt den Empfang der Netzdaten (PZD- und evtl. PKW-Daten) von den Slaves je nach der verwendeten Netzdatenstruktur ab.
S_USSI	Wird diese Anweisung S_USSI bei Anlauf des S7-Systems aufgerufen, werden Kommunikationsprozessor-DB, Netzdaten-DB und Parametrierungs-DB, die für die Kommunikation erforderlich sind, generiert.

Tabelle 1-106 Anweisungen für Modbus

Anweisung	Bedeutung
MODB_341	Mit der Anweisung MODB_341 und dem dazugehörenden Treiber wird eine Kommunikationsverbindung zwischen einem Modbus-Master-Steuerungssystem und der Kommunikationsbaugruppe CP 341 "modbusfähiges" Slavesystem ermöglicht.
MODB_441	Mit der Anweisung MODB_441 und dem dazugehörenden Treiber wird eine Kommunikationsverbindung zwischen einem Modbus-Master-Steuerungssystem und der Kommunikationsbaugruppe CP 441-2 "modbusfähiges" Slavesystem ermöglicht..

Anweisungen des Automatisierungssystems S7-400

In der folgenden Tabelle finden Sie die Anweisungen des Automatisierungssystems S7-400, die für die Kommunikation zwischen CPU und CP 441 zur Verfügung stehen.

Tabelle 1-107 Anweisungen des Automatisierungssystems S7-400

Anweisung	Bedeutung
BSEND	Mit der Anweisung BSEND können Sie Daten aus einem S7-Datenbereich an einen Kommunikationspartner mit fester Zielangabe senden.
BRCV	Mit der Anweisung BRCV können Sie Daten von einem Kommunikationspartner empfangen und in einen S7-Datenbereich übergeben.
GET	Nur bei RK512: Mit der Anweisung GET können Sie Daten von einem Kommunikationspartner holen.
PUT	Nur bei RK512: Mit der Anweisung PUT können Sie Daten an einen Kommunikationspartner mit dynamisch änderbarer Zielangabe senden.
PRINT	Mit der Anweisung PRINT können Sie einen Meldetext mit bis zu vier Variablen auf einen Drucker ausgeben.
STATUS	Mit der Anweisung STATUS können Sie den Status einer Kommunikationsverbindung abfragen.

Weitere Informationen

Eine ausführliche Beschreibung der Anweisungen des Automatisierungssystems S7-400 und deren Parameter finden Sie unter: Übersicht über die Anweisungen zur S7-Kommunikation

Basisinformationen

Serielle Übertragung eines Zeichens

Einleitung

Zum Austausch von Daten zwischen zwei oder mehr Kommunikationspartnern stehen unterschiedliche Möglichkeiten der Vernetzung zur Verfügung. Die Punkt-zu-Punkt-Kopplung zwischen zwei Kommunikationspartnern ist der einfachste Fall des Informationsaustausches.

Punkt-zu-Punkt-Kopplung

Bei der Punkt-zu-Punkt-Kopplung bildet die Kommunikationsbaugruppe die Schnittstelle zwischen einer Speicherprogrammierbaren Steuerung und einem Kommunikationspartner. Die Übertragung der Daten erfolgt bei der Punkt-zu-Punkt-Kopplung mit der Kommunikationsbaugruppe seriell.

Auf der Kommunikationsbaugruppe erfolgt eine Umsetzung auf die Adressierungsmechanismen der ausgewählten Übertragungsprozedur. Deshalb endet die Punkt-zu-Punkt-Verbindung bereits auf der Kommunikationsbaugruppe und nicht wie bei anderen Verbindungen auf dem Kommunikationspartner.

Serielle Datenübertragung

Bei der seriellen Datenübertragung werden die einzelnen Bits eines Bytes einer zu übertragenden Information in einer festgelegten Reihenfolge nacheinander übertragen.

Uni-/Bidirektionaler Datenverkehr

Die Datenübertragung mit dem Kommunikationspartner wird über die serielle Schnittstelle der Kommunikationsbaugruppe selbständig abgewickelt. Die Kommunikationsbaugruppe ist hierfür mit verschiedenen Treibern ausgestattet.

- unidirektionaler Datenverkehr:
  - Drucker-Treiber
- bidirektionaler Datenverkehr:
  - ASCII-Treiber
  - Prozedur 3964(R)
  - Rechnerkopplung RK512
  - Modbus/USS

Die Datenübertragung über die serielle Schnittstelle wird von der Kommunikationsbaugruppe je nach Schnittstellenphysik und dem angewählten Treiber abgewickelt.

Unidirektionaler Datenverkehr - Druckerausgabe

Bei der Druckerausgabe (Drucker-Treiber) werden n Byte Nutzdaten auf einen Drucker ausgegeben. Ein Empfang von Zeichen findet nicht statt. Ausgenommen davon sind einzelne Steuerzeichen zur Datenflusskontrolle (z. B. XON/XOFF).

Bidirektionaler Datenverkehr - Betriebsarten

Beim bidirektionalen Datenverkehr unterscheidet man bei der Kommunikationsbaugruppe zwei Betriebsarten:

- Halbduplexbetrieb (Prozedur 3964(R), ASCII-Treiber, RK512, Modbus/USS)
  Die Daten werden zwischen einem oder mehreren Kommunikationspartnern abwechselnd in beide Richtungen übertragen. Halbduplexbetrieb bedeutet, dass zu einem Zeitpunkt entweder gesendet oder empfangen wird. Die Ausnahme hiervon können einzelne Steuerzeichen zur Datenflusskontrolle (z. B. XON/XOFF) bilden, die auch während eines Sende-/Empfangsbetriebs empfangen/gesendet werden können.
- Vollduplexbetrieb (ASCII-Treiber)
  Die Daten werden zwischen einem oder mehreren Kommunikationspartnern gleichzeitig ausgetauscht, es kann zu einem Zeitpunkt also sowohl gesendet und empfangen werden. Jeder Kommunikationspartner muss simultan eine Sende- und Empfangseinrichtung betreiben können.

Beim Schnittstellenmodul X27 (RS422/485) kann mit RS485-Einstellung (2-Draht) nur im Halbduplexbetrieb gearbeitet werden.

Asynchrone Datenübertragung

Bei der Kommunikationsbaugruppe erfolgt die serielle Datenübertragung asynchron. Der sogenannte Zeitrastergleichlauf (festes Zeitraster bei der Übertragung einer festen Zeichenfolge) wird nur während der Übertragung eines Zeichens aufrechterhalten. Jedem zu übertragenden Zeichen geht ein Synchronisierungsimpuls, auch als Startbit bezeichnet, voraus. Die Länge der Startbitübertragung legt den Takt fest. Das Ende des Zeichentransfers bildet das Stoppbit.

Vereinbarungen

Neben Start- und Stoppbit sind weitere Vereinbarungen zwischen den beiden Kommunikationspartnern für eine serielle Datenübertragung nötig. Darunter fallen:

- Übertragungsgeschwindigkeit (Baudrate)
- Zeichen- und Quittungsverzugszeit
- Parität
- Anzahl der Datenbits
- Anzahl der Stopbits

Zeichenrahmen

Die Daten zwischen Kommunikationsbaugruppe und einem Kommunikationspartner werden über die serielle Schnittstelle in einem 10-, 11- oder 12-Bit-Zeichenrahmen übertragen. Für jeden Zeichenrahmen stehen mehrere Datenformate zur Verfügung. Parametrieren Sie das gewünschte Format der Datenübertragung im Eigenschaftendialog der Baugruppe. Abhängig von der jeweiligen Kommunikationsbaugruppe steht nicht jede der theoretisch möglichen Zeichenrahmen (-Kombinationen) zur Verfügung.

10-Bit-Zeichenrahmen

Im folgenden Bild sind beispielhaft drei Datenformate des 10-Bit-Zeichenrahmens dargestellt.

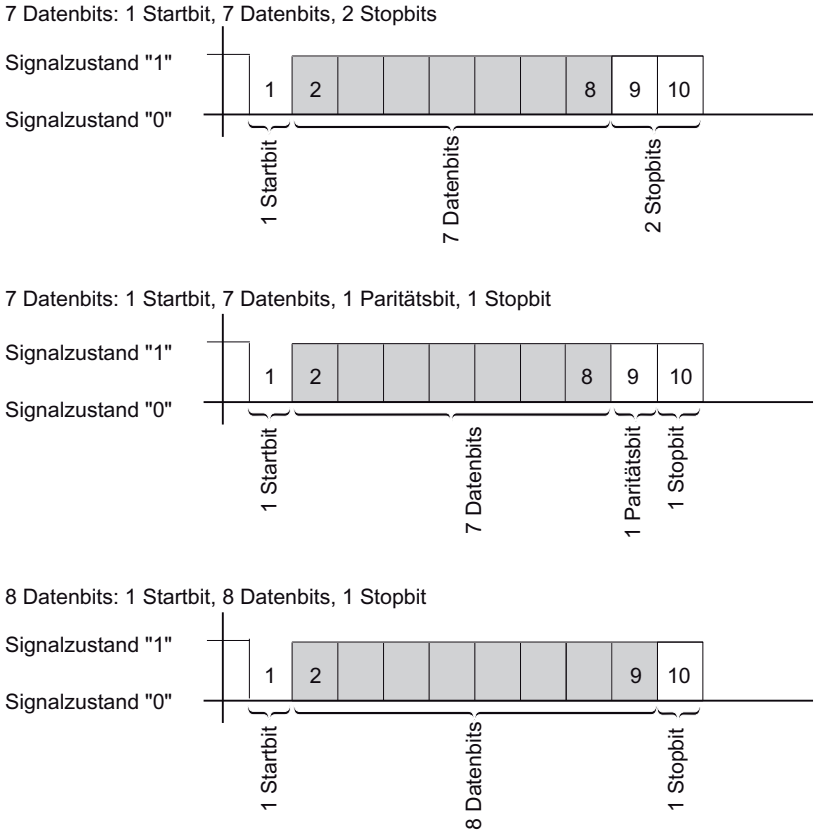

Bild 1-40 10-Bit-Zeichenrahmen

11-Bit-Zeichenrahmen

Im folgenden Bild sind beispielhaft drei Datenformate des 11-Bit-Zeichenrahmens dargestellt.

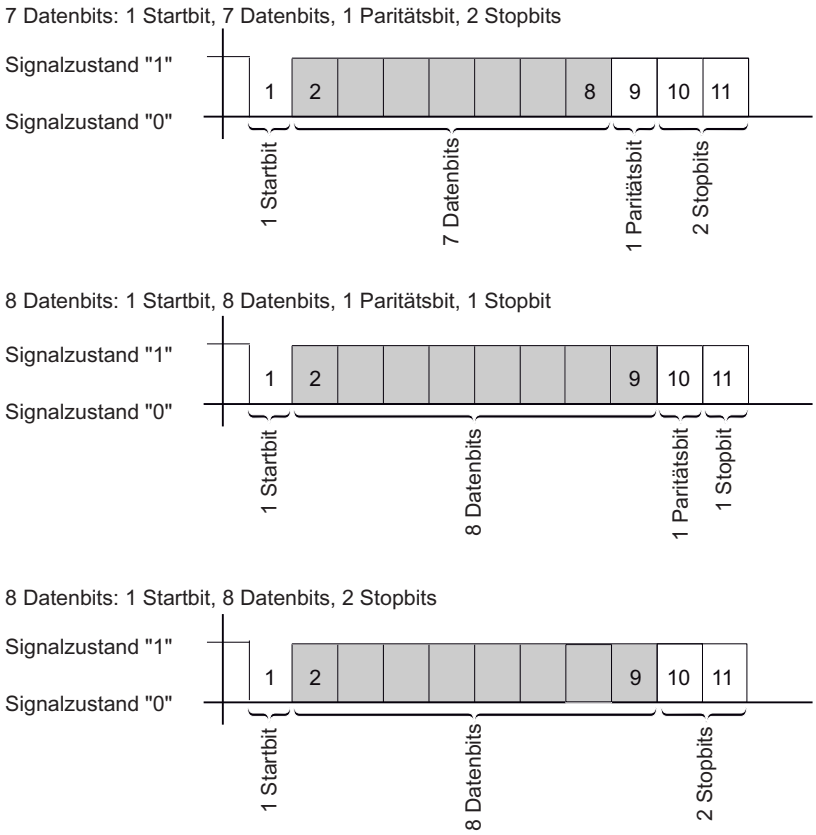

Bild 1-41 11-Bit-Zeichenrahmen

Zeichenverzugszeit

Im folgenden Bild ist der maximal zulässige zeitliche Abstand zwischen zwei empfangenen Zeichen innerhalb eines Telegramms = Zeichenverzugszeit dargestellt.

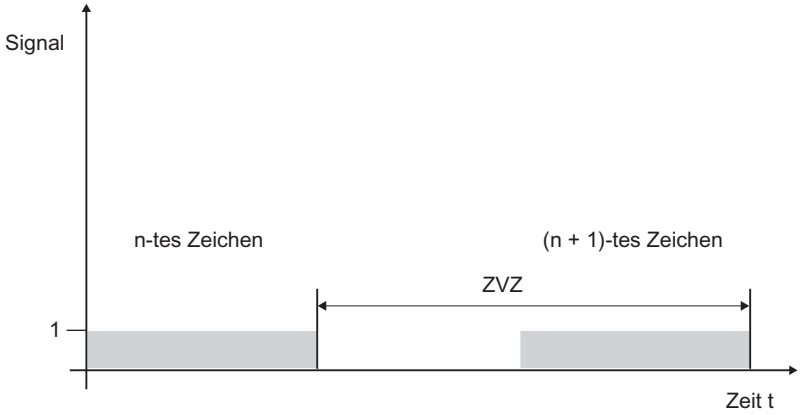

Bild 1-42 Zeichenverzugszeit

Übertragungssicherheit

Die Übertragungssicherheit spielt bei der Übertragung von Daten und bei der Wahl des Übertragungsverfahrens eine wichtige Rolle. Allgemein gilt, je mehr Schichten des Referenzmodells durchlaufen werden, desto höher ist die Übertragungssicherheit.

Einordnung der vorhandenen Protokolle

Die Einordnung der vorhandenen Protokolle der Kommunikationsbaugruppe in das Referenzmodell können Sie der folgenden Abbildung entnehmen:

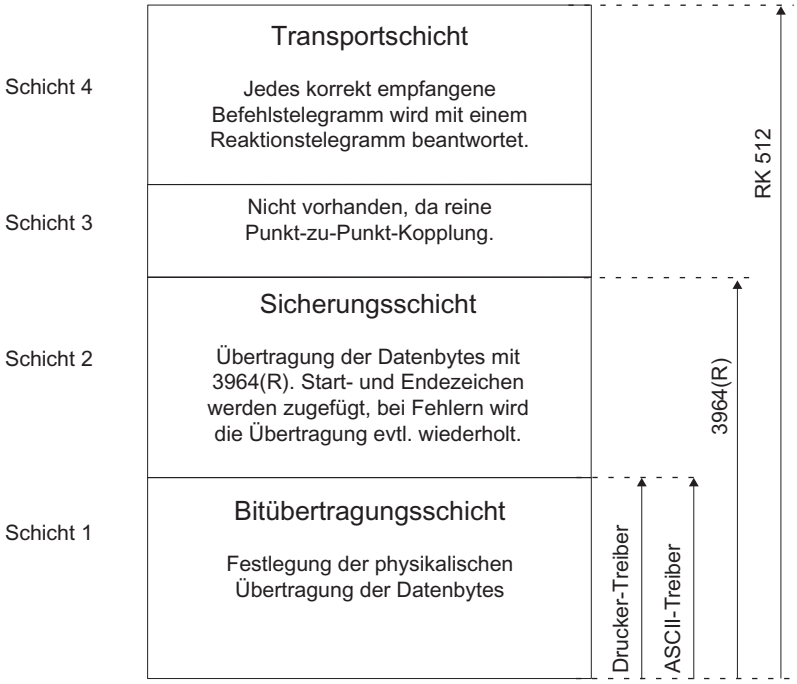

Bild 1-43 Einordnung der vorhandenen Protokolle der Kommunikationsbaugruppe in das Referenzmodell

Zuordnung der Protokolle zu den Kommunikationsbaugruppen

Protokoll	CP 340	CP 341	CPU 313C-2 PtP	CPU 314C-2 PtP	CP 440	CP 441	ET 200S 1SI
Drucker-Treiber	●	●				●	
ASCII-Treiber	●	●	●	●	●	●	●
Prozedur 3964(R)	●	●	●	●	●	●	●
Rechnerkopplung RK512		●		●		● 2)	
Modbus Master		● 1)				● 1) 2)	●

Protokoll	CP 340	CP 341	CPU 313C-2 PtP	CPU 314C-2 PtP	CP 440	CP 441	ET 200S 1SI
Modbus Slave		● [1]				● [1] [2]	●
USS Master							●

[1] Für das Modbus-Protokoll wird ein ladbarer Treiber mit einem Dongle benötigt.

[2] nur CP 441-2

Übertragungssicherheit beim Drucker-Treiber

Datensicherheit bei Verwendung des Drucker-Treibers:

- Beim Datentransport mit dem Drucker-Treiber ist keine Maßnahme zur Datensicherheit vorgesehen.
- Um zu verhindern, dass beim Überlauf des Drucker-Empfangspuffers Daten verloren gehen, können Sie mit Datenflusskontrolle (XON/XOFF, RTS/CTS) arbeiten.
- Bei der Druckerausgabe wird das BUSY-Signal des Druckers ausgewertet. Das BUSY-Signal kommt an der Kommunikationsbaugruppe als CTS-Signal an und wird auch wie CTS (siehe ASCII-Treiber) ausgewertet. Beachten Sie, dass Sie bei Flusskontrolle mit CTS/RTS am Drucker die Polarität des BUSY-Signals auf CTS = "OFF" einstellen müssen.

Übertragungssicherheit beim ASCII–Treiber

Datensicherheit bei Verwendung des ASCII–Treibers:

- Dadurch, dass bei einem Datentransport mit dem ASCII–Treiber neben der Verwendung des Paritätsbits (kann auch abhängig von der Einstellung des Zeichenrahmens abgewählt werden) keine weiteren Maßnahmen zur Datensicherung erfolgen, ist eine Datenübertragung mit dem ASCII–Treiber zwar sehr effizient, was den Datendurchsatz anbelangt, ein abgesicherter Datentransport ist nicht gegeben.
- Durch die Verwendung des Paritätsbits wird das Kippen eines Bits in einem zu übertragenden Zeichen abgesichert. Kippen zwei oder mehr Bits eines Zeichens, so kann dieser Fehler nicht mehr erkannt werden.
- Soll die Übertragungssicherheit erhöht werden, so kann dies durch Einführung einer Prüfsumme und Längenangabe eines Telegramms erfolgen. Diese Maßnahmen müssen durch den Anwender realisiert werden.
- Eine weitere Steigerung der Datensicherheit kann durch die Einführung von Quittungstelegrammen auf Sende– oder Empfangstelegramme erfolgen. Dies ist bei hochwertigen Protokollen zur Datenkommunikation realisiert (ISO–7–Schichten–Referenzmodell).

Übertragungssicherheit bei 3964(R)

Erhöhte Datensicherheit durch Verwendung der Prozedur 3964(R):

- Die Hamming–Distanz bei 3964(R) beträgt 3. Die Hamming–Distanz ist ein Maß für die Sicherheit einer Datenübertragung.
- Durch die Prozedur 3964(R) wird eine hohe Übertragungssicherheit auf der Übertragungsleitung gewährleistet. Die hohe Übertragungssicherheit wird durch einen festgelegten Telegrammauf– und –abbau sowie die Mitführung des Blockprüfzeichens (BCC) erreicht.

Je nachdem, ob Sie mit oder ohne Blockprüfzeichen Daten übertragen wollen, unterscheidet man zwischen

- Datenübertragung ohne Blockprüfzeichen: **3964**
- Datenübertragung mit Blockprüfzeichen: **3964(R)**

Bei Beschreibungen und Hinweisen, die sich auf beide Datenübertragungsarten beziehen, verwenden wir in diesem Handbuch die Bezeichnung 3964(R).

Übertragungssicherheit bei RK512

Sehr hohe Datensicherheit durch Verwendung der RK512:

- Die Hamming–Distanz bei RK512 mit 3964R beträgt 4. Die Hamming–Distanz ist ein Maß für die Sicherheit einer Datenübertragung.
- Durch die Verwendung der Rechnerkopplung RK512 wird eine hohe Übertragungssicherheit auf der Übertragungsleitung gewährleistet (weil bei RK512 die Prozedur 3964R zum Datentransport benutzt wird).
- Es ist eine Weiterverarbeitung im Kommunikationspartner sichergestellt (weil der RK512–Interpreter die zusätzliche Längenangabe im Kopf auswertet und nach Ablage der Daten im Zieldatenbereich des Kommunikationspartners ein Quittungstelegramm über den erfolgreichen/misslungenen Datentransport generiert.).
- Die Rechnerkopplung RK512 gewährleistet die richtige Verwendung der Prozedur 3964R und die Auswertung/Ergänzung der Längenangabe sowie die Generierung der Reaktionstelegramme selbständig. Dies ist kein Anwenderhandling! Sie müssen lediglich die positive/negative Abschlussquittung auswerten.

Übertragungssicherheit bei Modbus/USS

Datensicherheit bei Verwendung des Modbus/USS-Treibers.

- Die Hamming–Distanz bei Modbus/USS beträgt 4. Die Hamming–Distanz ist ein Maß für die Sicherheit einer Datenübertragung.
- Durch die Verwendung von Modbus/USS wird eine hohe Übertragungssicherheit auf der Übertragungsleitung gewährleistet (durch einen CRC16 - Blockcheck bei Modbus und einen XOR - Blockcheck bei USS).
- Es ist eine Weiterverarbeitung im Kommunikationspartner sichergestellt, weil sowohl Modbus als auch USS eine zusätzliche Längenangabe im Kopf auswerten und ein (zeitüberwachtes) Quittungstelegramm über den erfolgreichen/misslungenen Datentransport generiert wird.

Datenübertragung mit ASCII-Treiber

Datenübertragung mit dem ASCII-Treiber

Zuordnung der Protokolle zu den Kommunikationsbaugruppen siehe: Übersicht der Komponenten und Schnittstellen (Seite 1670)

Einleitung

Der ASCII-Treiber steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen dem Kommunikationsprozessor und einem Kommunikationspartner. Der ASCII-Treiber beinhaltet die Bitübertragungsschicht (Schicht 1).

Der Aufbau der Telegramme wird dadurch offengehalten, dass der S7-Anwender das komplette Sendetelegramm an den Kommunikationsprozessor übergibt. Für die Empfangsrichtung ist das Endekriterium eines Telegramms zu parametrieren. Der Aufbau der Sendetelegramme kann sich vom Aufbau der Empfangstelegramme unterscheiden.

Mit dem ASCII-Treiber können Daten mit beliebigem Aufbau (alle druckbaren ASCII-Zeichen ebenso wie alle anderen Zeichen von 00 bis FFH (bei Zeichenrahmen mit 8 Datenbits) bzw. von 00 bis 7FH (bei Zeichenrahmen mit 7 Datenbits) gesendet und empfangen werden.

Daten senden mit ASCII-Treiber

Daten senden

Beim Senden geben Sie die Anzahl der zu übertragenden Nutzdaten-Bytes beim Aufruf der jeweiligen Sendeanweisung am Längenparameter an. Weitere Informationen hierzu finden Sie in Kapitel "Übersicht über die Anweisungen (Seite 1671)".

Wenn Sie mit dem Endekriterium "Ablauf der Zeichenverzugszeit" arbeiten, hält der ASCII-Treiber auch beim Senden eine Pause zwischen zwei Telegrammen ein. Sie können die Sende-Anweisung jederzeit aufrufen, der ASCII-Treiber beginnt mit der Ausgabe aber erst, wenn seit dem letzten gesendeten Telegramm eine Zeit größer der parametrierten Zeichenverzugszeit vergangen ist.

Hinweis

Bei der Parametrierung der Flusskontrolle XON/XOFF dürfen die Nutzdaten keines der parametrierten XON- oder XOFF-Zeichen enthalten. Die Default-Einstellungen sind DC1 = 11H für XON und DC3 = 13H für XOFF.

Wenn Sie mit dem Endekriterium "Endezeichen" arbeiten, können Sie zwischen 3 Möglichkeiten wählen:

- Senden bis zum Endekennzeichen einschließlich
  Das Endekennzeichen muss in den zu sendenden Daten enthalten sein. Es werden die Daten nur bis zum Endekennzeichen einschließlich gesendet, auch wenn an der Anweisung eine größere Datenlänge angegeben ist.
- Senden bis zur an der Anweisung parametrierten Länge
  Es werden die Daten bis zu der an der Anweisung parametrierten Länge gesendet. Das letzte Zeichen muss das Endezeichen sein.
- Senden bis zu der an der Anweisung parametrierten Länge und automatisches Anhängen der/des Endekennzeichen/s
  Es werden die Daten bis zu der an der Anweisung parametrierten Länge gesendet. Zusätzlich werden das/die Endezeichen automatisch angehängt. D. h. die Endekennzeichen dürfen nicht in den zu sendenden Daten enthalten sein. Je nach Anzahl der Endekennzeichen werden 1 oder 2 Zeichen mehr zum Partner geschickt, als von der Anweisung angegeben.

Wenn Sie mit dem Endekriterium "feste Telegrammlänge" arbeiten, wird in Senderichtung die Anzahl von Daten übertragen, die Sie am Längenparameter der Anweisung angegeben haben. In Empfangsrichtung, d. h. im Empfangs-DB wird die Anzahl von Daten eingetragen, die Sie beim Empfänger im Eigenschaftendialog der Baugruppe über den Parameter "feste Telegrammlänge" eingetragen haben. Um einwandfreien Datenverkehr zu gewährleisten, sollten beide Parameter identisch gewählt werden. Beim Senden wird zwischen zwei Telegrammen eine Pause in der Länge der Zeichenverzugszeit (ZVZ) eingehalten, damit sich der Partner einsynchronisieren (Erkennen des Telegrammanfangs) kann.

Erfolgt die Einsynchronisation durch andere Mechanismen, kann das Einhalten der Sendepause im Eigenschaftendialog der Baugruppe abgeschaltet werden.

Ablauf beim Senden

Im folgenden Bild finden Sie die Abläufe beim Senden.

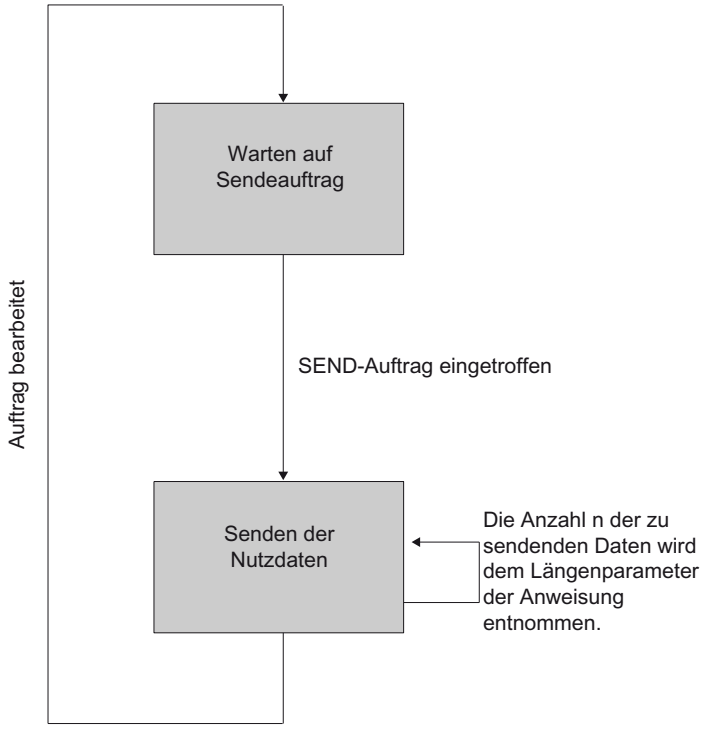

Bild 1-44 ASCII-Treiber Ablaufschema beim Senden

Daten empfangen mit ASCII-Treiber

Wählbare Endekriterien

Bei der Datenübertragung mit dem ASCII-Treiber können Sie zwischen drei verschiedenen Endekriterien wählen. Das Endekriterium legt fest, wann ein Telegramm vollständig empfangen wurde. Die einstellbaren Endekriterien sind:

- Ablauf der Zeichenverzugszeit
  Das Telegramm hat weder eine feste Länge noch definierte Endezeichen, das Ende eines Telegramms ist durch eine Pause auf der Leitung (Ablauf der Zeichenverzugszeit) festgelegt.
- Empfang der/des Endezeichen(s)
  Am Ende des Telegramms stehen ein oder zwei definierte Endezeichen.
- Empfang einer festen Zeichenanzahl
  Die Länge der Empfangstelegramme ist immer gleich.

Codetransparenz

Die Codetransparenz der Prozedur hängt von der Wahl des parametrierten Endekriteriums und der Flusskontrolle ab:

- Mit ein oder zwei Endezeichen
  - nicht codetransparent
- Endekriterium Zeichenverzugszeit oder feste Telegrammlänge
  - codetransparent
- Bei Verwendung der Flusskontrolle XON/XOFF ist kein codetransparenter Betrieb möglich.

Codetransparent bedeutet, dass in den Nutzdaten alle beliebigen Zeichenkombinationen vorkommen dürfen, ohne dass das Endekriterium erkannt wird.

Endekriterium Ablauf der Zeichenverzugszeit

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die Zeichenverzugszeit abgelaufen ist. Die empfangenen Daten werden von der CPU übernommen.

Die Zeichenverzugszeit muss in diesem Fall so eingestellt werden, dass sie zwischen zwei aufeinanderfolgenden Telegrammen sicher abläuft. Sie sollte aber so groß sein, dass bei Sendepausen des Koppelpartners innerhalb eines Telegramms nicht fälschlicherweise das Telegrammende erkannt wird.

Im folgenden Bild finden Sie die Abläufe beim Empfangen mit dem Endekriterium "Ablauf der Zeichenverzugszeit".

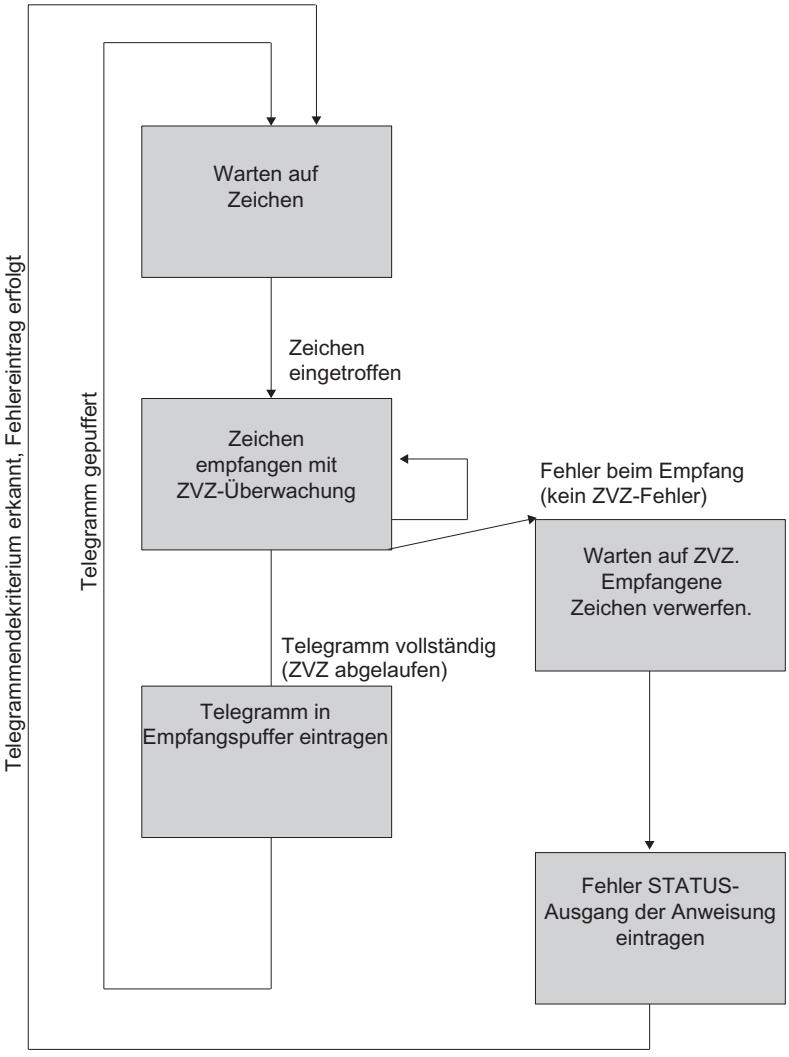

Bild 1-45 Ablaufschema beim Empfangen mit Endekriterium "Ablauf der Zeichenverzugszeit"

Endekriterium Endezeichen

Beim Empfang von Daten wird das Telegrammende erkannt, wenn das/die parametrierten Endezeichen empfangen werden. Die empfangenen Daten werden inklusive Endezeichen von der CPU übernommen.

Der Ablauf der Zeichenverzugszeit während des Empfangs führt zur Beendigung des Empfangs. Es erfolgt eine Fehlermeldung, und das Telegrammfragment wird verworfen.

Wenn mit Endezeichen gearbeitet wird, ist die Übertragung nicht codetransparent, und es muss ausgeschlossen werden, dass die Endekennung/en in den Nutzdaten des Anwenders enthalten sind.

Beachten Sie Folgendes, wenn im empfangenen Telegramm das letzte Zeichen nicht das Endezeichen ist:

- Endezeichen im Telegramm an beliebiger Stelle enthalten:
  Alle Zeichen inklusive des Endezeichens werden in den Empfangs-DB eingetragen. Die Zeichen, die hinter dem Endezeichen stehen, werden
  - verworfen, wenn am Ende des Telegramms die Zeichenverzugszeit (ZVZ) abläuft.
  - mit dem nächsten Telegramm verschmolzen, wenn ein neues Telegramm empfangen wird, bevor die Zeichenverzugszeit abgelaufen ist.
- Endezeichen im Telegramm nicht enthalten:
  Das Telegramm wird
  - verworfen, wenn am Ende des Telegramms die Zeichenverzugszeit (ZVZ) abläuft.
  - mit dem nächsten Telegramm verschmolzen, wenn ein neues Telegramm empfangen wird, bevor die Zeichenverzugszeit abgelaufen ist.

Im folgenden Bild finden Sie die Abläufe beim Empfangen mit dem Endekriterium "Endezeichen".

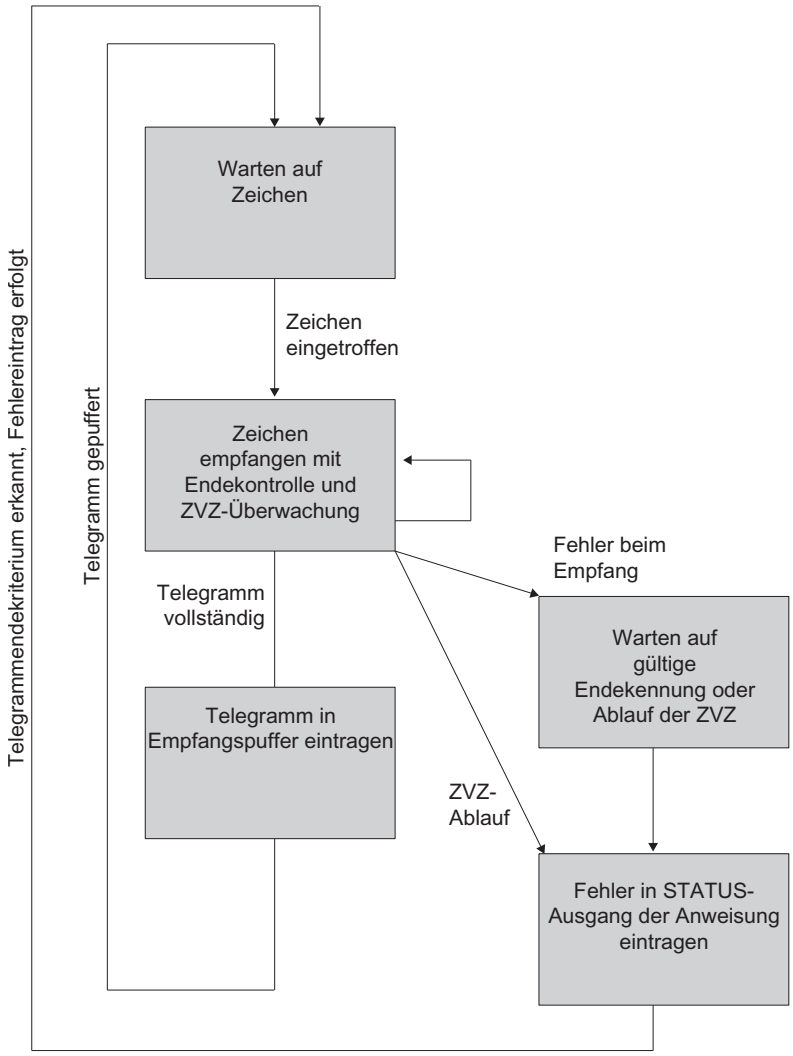

Bild 1-46 Ablaufschema beim Empfangen mit Endekriterium "Endezeichen"

Endekriterium feste Telegrammlänge

Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Anzahl von Zeichen empfangen wurde. Die empfangenen Daten werden von der CPU übernommen.

Der Ablauf der Zeichenverzugszeit vor Erreichen der parametrierten Zeichenanzahl führt zur Beendigung des Empfangs. Es erfolgt eine Fehlermeldung und das Telegrammfragment wird verworfen.

Beachten Sie Folgendes, wenn die Telegrammlänge der empfangenen Zeichen nicht mit der festen parametrierten Telegrammlänge übereinstimmt:

- Telegrammlänge der empfangenen Zeichen größer als die feste parametrierte Telegrammlänge:
  Alle Zeichen, die nach Erreichen der festen parametrierten Telegrammlänge empfangen werden, werden
  - verworfen, wenn am Ende des Telegramms die Zeichenverzugszeit (ZVZ) abläuft.
  - mit dem nächsten Telegramm verschmolzen, wenn ein neues Telegramm empfangen wird, bevor die Zeichenverzugszeit abgelaufen ist.
- Telegrammlänge der empfangenen Zeichen kleiner als die feste parametrierte Telegrammlänge:
  Das Telegramm wird
  - verworfen, wenn am Ende des Telegramms die Zeichenverzugszeit (ZVZ) abläuft.
  - mit dem nächsten Telegramm verschmolzen, wenn ein neues Telegramm empfangen wird, bevor die Zeichenverzugszeit abgelaufen ist.

Im folgenden Bild finden Sie die Abläufe beim Empfangen mit dem Endekriterium "feste Telegrammlänge".

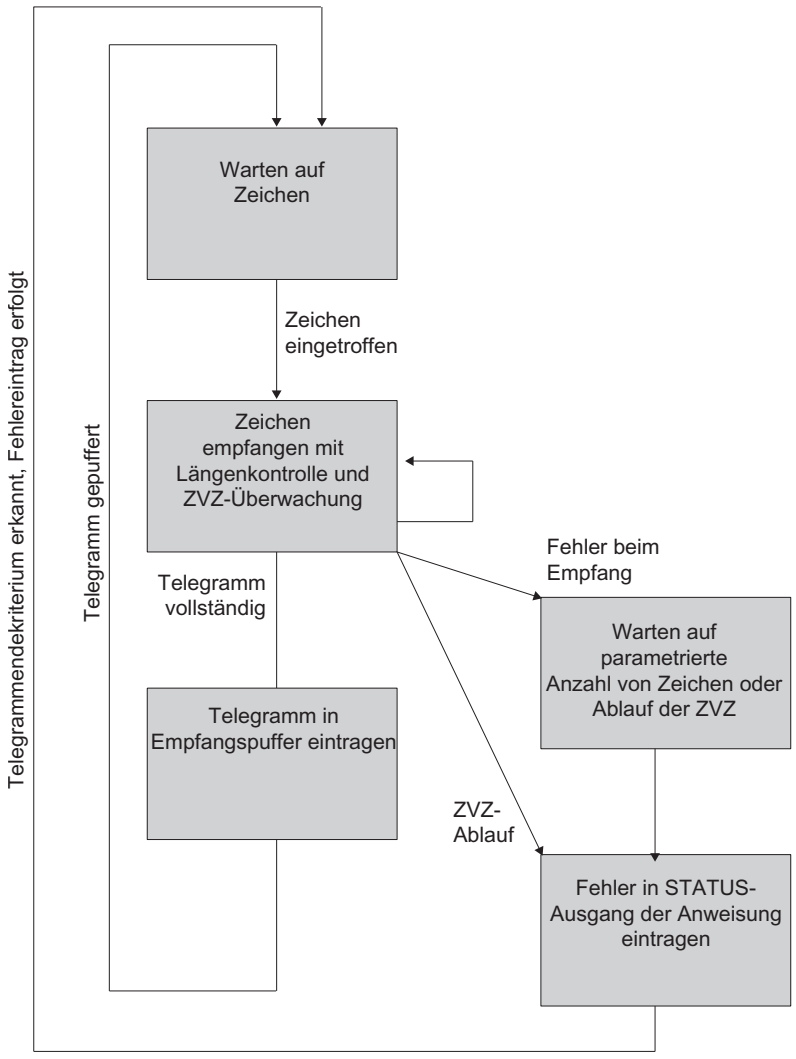

Bild 1-47 Ablaufschema beim Empfangen mit Endekriterium "feste Telegrammlänge"

Empfangspuffer der Baugruppe

Der Empfangspuffer der Baugruppe umfasst, abhängig von der eingesetzten Kommunikationsbaugruppe, bis zu 4096 Byte.

Bei der Parametrierung können Sie angeben ob:

- CP und CPU 31xC-2 PtP
  der Empfangspuffer im Anlauf gelöscht werden soll und ein Überschreiben von Daten im Empfangspuffer verhindert werden soll. Zusätzlich können Sie den Wertebereich (1 bis 250) für die Anzahl der gepufferten Empfangstelegramme angeben.
- ET 200S 1SI
  der Empfangspuffer im Anlauf gelöscht werden soll und ob ein Überschreiben von Daten im Empfangspuffer verhindert werden soll. Zusätzlich können Sie das Puffern der empfangenen Telegramme aktivieren oder sperren.

Der Empfangspuffer auf der Baugruppe ist ein Ringpuffer:

- Werden mehrere Telegramme in den Empfangspuffer der Baugruppe eingetragen, gilt: Es wird immer das älteste Telegramm von der Baugruppe an die CPU übertragen.
- Wenn Sie immer nur das neueste Telegramm zur CPU übertragen wollen, müssen Sie:
  - CP und CPU 31xC-2 PtP
    für die Anzahl der gepufferten Telegramme den Wert "1" parametrieren und den Überschreibschutz deaktivieren.
  - ET 200S 1SI
    die dynamischen Telegramme sperren und den Überschreibschutz ausschalten.

Hinweis

Wird das ständige Auslesen der Empfangsdaten im Anwenderprogramm für eine Zeit lang ausgesetzt, kann es beim erneuten Anfordern der Empfangsdaten dazu kommen, dass von der Baugruppe erst alte und dann erst das neueste Telegramm von der CPU empfangen wird.

Die alten Telegramme waren bei der Unterbrechung auf dem Weg zwischen Baugruppe und CPU unterwegs, bzw. wurden von der Anweisung bereits empfangen.

RS485-Betrieb (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI)

RS485-Betrieb

Wenn Sie den ASCII-Treiber im RS485-Betrieb (Halbduplex, Zweidrahtbetrieb) betreiben, müssen Sie im Anwenderprogramm dafür sorgen, dass immer nur ein Teilnehmer sendet. Wenn gleichzeitig gesendet wird, wird das Telegramm verfälscht.

Umschaltzeiten für RS485-Modul im Halbduplexbetrieb

Die maximale Umschaltzeit zwischen Senden und Empfangen beträgt 1 ms.

RS232-Betrieb (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI)

RS232C-Begleitsignale

Auf den Baugruppen sind bei Einsatz der RS232C-Physik folgende RS232C-Begleitsignale vorhanden:

DCD	(Eingang)	Data Carrier detect; Datenträger erkannt
DTR	(Ausgang)	Data terminal ready; Baugruppe betriebsbereit
DSR	(Eingang)	Data set ready; Kommunikationspartner betriebsbereit
RTS	(Ausgang)	Request to send; Baugruppe sendebereit

CTS	(Eingang)	Clear to send; Kommunikationspartner kann Daten von der Baugruppe empfangen (Antwort auf RTS = ON der Kommunikationsbaugruppe)
RI	(Eingang)	Ring Indicator; Rufzeichen

Nach dem Einschalten der Baugruppe befinden sich die Ausgangssignale im Zustand OFF (inaktiv).

Die Bedienung der Steuersignale DTR/DSR und RTS/CTS können Sie im Eigenschaftendialog der Baugruppe parametrieren oder über Anweisungen im Anwenderprogramm steuern.

RS232C-Begleitsignale bedienen

Die RS232C-Begleitsignale können bedient werden:

- bei parametrierter automatischen Bedienung aller RS232C-Begleitsignale
- bei parametrierter Datenflusskontrolle (RTS/CTS)
- über die Anweisungen V24_STAT und V24_SET

Hinweis

Bei der Parametrierung einer automatischen Bedienung der RS232C-Begleitsignale ist weder eine Datenflusskontrolle mit RTS/CTS noch ein Steuern von RTS und DTR über die Anweisung V24_SET möglich! Bei der Parametrierung einer Datenflusskontrolle mit RTS/CTS ist ein Steuern von RTS über die Anweisung V24_SET nicht möglich! Das Lesen aller RS232C-Begleitsignale über die Anweisung V24_STAT ist hingegen immer möglich.

Die folgenden Abschnitte erläutern das prinzipielle Handling bei der Steuerung und Auswertung der RS 232C-Begleitsignale.

Automatische Bedienung der Begleitsignale

Die Realisierung der automatischen Bedienung der RS232C-Begleitsignale auf der Kommunikationsbaugruppe wird folgendermaßen durchgeführt:

- Sobald die Baugruppe durch Parametrierung in eine Betriebsart mit automatischer Bedienung der RS232C-Begleitsignale gebracht wurde, setzt sie die Leitungen RTS auf OFF und DTR auf ON (Kommunikationsbaugruppe betriebsbereit).
  Das Senden und Empfangen von Telegrammen ist erst möglich, nachdem die Leitung DTR auf ON gesetzt wurde. Solange DTR auf OFF gesetzt bleibt, werden keine Daten über die RS232C-Schnittstelle empfangen. Ein Sendeauftrag wird mit einer entsprechenden Fehlermeldung abgebrochen.
- Steht ein Sendeauftrag an, wird RTS auf ON gesetzt und die parametrierte Datenausgabewartezeit gestartet. Nach Ablauf der Datenausgabezeit und CTS = ON werden die Daten über die RS232C-Schnittstelle gesendet.
- Wird beim Senden innerhalb der Datenausgabewartezeit die Leitung CTS nicht auf ON gesetzt, oder erfolgt innerhalb des Sendevorgangs ein Wechsel von CTS auf OFF wird der Sendeauftrag abgebrochen und eine entsprechende Fehlermeldung generiert.
- Nach dem Senden der Daten wird nach Ablauf der parametrierten RTS-Wegnahmezeit die Leitung RTS auf OFF gesetzt. Es erfolgt kein Warten auf den Wechsel von CTS auf OFF.

- Ein Empfang von Daten über die RS232C-Schnittstelle ist möglich, sobald die Leitung DSR auf ON gesetzt ist. Droht der Empfangspuffer der Kommunikationsbaugruppe überzulaufen, erfolgt keine Reaktion der Baugruppe.
- Bei einem Wechsel von DSR = ON auf OFF wird sowohl ein laufender Sendeauftrag als auch das Empfangen von Daten mit einer Fehlermeldung abgebrochen. Im Diagnosepuffer der Baugruppe wird die Meldung "DSR = OFF (automat. Bedienung d. V24-Signale)" eingetragen.

Hinweis

Bei der Parametrierung einer automatischen Bedienung der RS232C-Begleitsignale ist weder eine Datenflusskontrolle mit RTS/CTS noch ein Steuern von RTS und DTR über die Anweisung V24_SET möglich!

Die "RTS-Wegnahmezeit" ist im Eigenschaftendialog der Baugruppe so einzustellen, dass der Kommunikationspartner die letzten Zeichen des Telegramms vollständig empfangen kann, bevor RTS und damit der Sendewunsch weggenommen wird. Die "Datenausgabewartezeit" ist so zu dimensionieren, dass der Kommunikationspartner in Empfangsbereitschaft gehen kann, bevor die Zeit abläuft.

Zeitdiagramm

Das Bild zeigt den zeitlichen Ablauf eines Sendeauftrags.

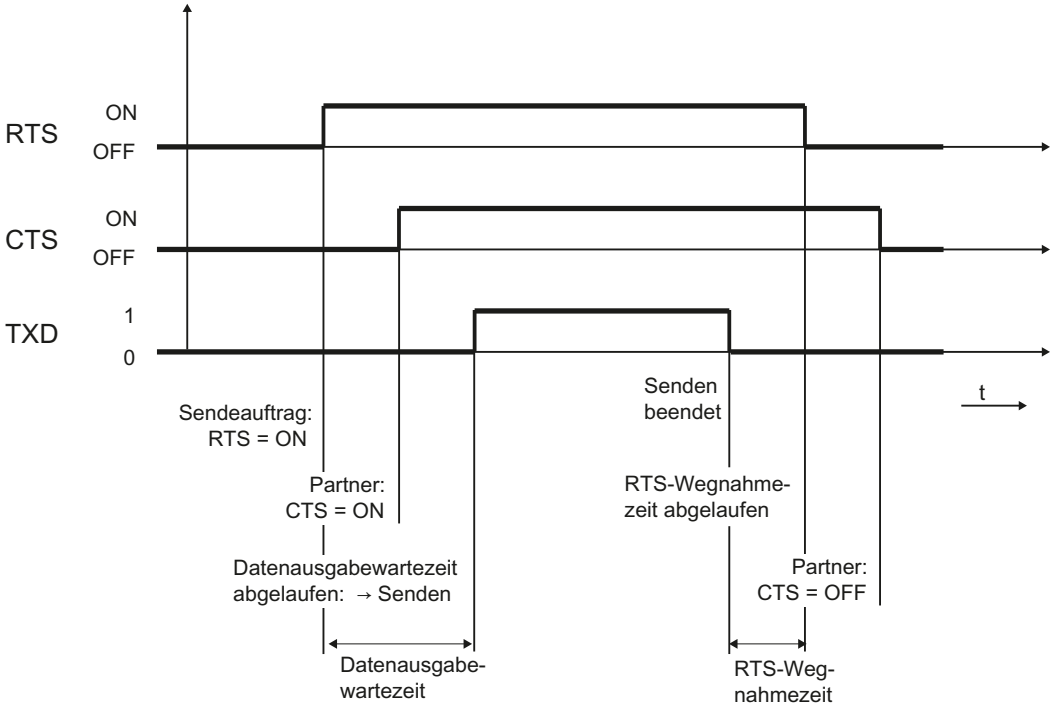

Bild 1-48 Zeitdiagramm bei automatischer Bedienung der RS232C–Begleitsignale

Aufgaben der Anweisung V24_STAT/SET

Mittels der Anweisung V24_STAT kann der Zustand von jedem RS232C-Begleitsignal ermittelt werden. Mit der Anweisung V24_SET ist ein Steuern der Ausgangssignale DTR und RTS möglich.

Softwarehandshake/Hardwarehandshake

Einleitung

Handshake-Verfahren steuern den Datenfluss zwischen zwei Kommunikationspartnern. Durch die Verwendung von Handshake-Verfahren wird vermieden, dass bei unterschiedlich schnell arbeitenden Geräten Daten bei der Übertragung verloren gehen.

Handshake-Verfahren

Grundsätzlich lassen sich die folgenden zwei Verfahren unterscheiden:

- Softwarehandshake (z. B. XON/XOFF)
- Hardwarehandshake (z. B. RTS/CTS)

Die Realisierung der Datenflusskontrolle auf der Kommunikationsbaugruppe wird folgendermaßen durchgeführt:

- Sobald die Baugruppe durch Parametrierung in eine Betriebsart mit Flusskontrolle gebracht wurde, sendet sie das Zeichen XON bzw. setzt die Leitung RTS auf ON.
- Bei Erreichen der parametrierten Telegrammanzahl bzw. 50 Zeichen bevor der Empfangspuffer überläuft (Größe des Empfangspuffers: 4096 Byte) sendet die Baugruppe das Zeichen XOFF bzw. setzt die Leitung RTS auf OFF. Sendet der Kommunikationspartner trotzdem weiter, wird bei Überlauf des Empfangspuffers eine Fehlermeldung generiert. Die empfangenen Daten des letzten Telegramms werden verworfen.
- Sobald ein Telegramm durch die S7-CPU abgeholt wurde und der Empfangspuffer aufnahmebereit ist, sendet die Baugruppe das Zeichen XON bzw. setzt die Leitung RTS auf ON.
- Empfängt die Baugruppe das Zeichen XOFF bzw. wird das Steuersignal CTS auf OFF gesetzt, unterbricht die Baugruppe den Sendevorgang. Wird nach einer bestimmten parametrierbaren Zeit kein XON empfangen bzw. CTS nicht auf ON gesetzt, wird der Sendevorgang abgebrochen und eine entsprechende Fehlermeldung (0708H) im Fehlermeldebereich der Baugruppe eingetragen.

Hinweis

Bei der Parametrierung einer Datenflusskontrolle mit RTS/CTS müssen Sie unbedingt eine Vollverdrahtung der verwendeten Schnittstellensignale in der Steckerverbindung vornehmen. Bei der Parametrierung einer Datenflusskontrolle mit RTS/CTS ist ein Steuern von RTS über die Anweisung V24_SET nicht möglich!

Datenübertragung mit 3964(R)

Datenübertragung mit der Prozedur 3964(R)

Zuordnung der Protokolle zu den Kommunikationsbaugruppen siehe: Übersicht der Komponenten und Schnittstellen (Seite 1670)

Einleitung

Die Prozedur 3964(R) steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen dem Kommunikationsprozessor und einem Kommunikationspartner. Die Prozedur 3964(R) beinhaltet neben der Bitübertragungsschicht (Schicht 1) die Sicherungsschicht (Schicht 2).

Anlauf

Im folgenden Bild finden Sie die Abläufe beim Anlauf der Prozedur 3964(R).

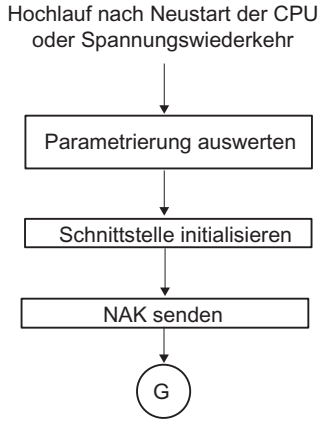

Bild 1-49 Ablaufschema beim Anlauf der Prozedur 3964(R)

Steuerzeichen

Einleitung

Die Prozedur 3964(R) fügt bei der Datenübertragung den Informationsdaten Steuerzeichen hinzu (Sicherungsschicht). Durch diese Steuerzeichen kann der Kommunikationspartner kontrollieren, ob die Daten vollständig und fehlerfrei bei ihm angekommen sind.

Die Steuerzeichen der Prozedur 3964(R)

Die Prozedur 3964(R) wertet die folgenden Steuerzeichen aus:

- **STX** Start of Text; Anfang der zu übertragenden Zeichenfolge
- **DLE** Data Link Escape; Datenübertragungsumschaltung
- **ETX** End of Text; Ende der zu übertragenden Zeichenfolge

- **BCC** Block Check Character (nur bei 3964(R)); Blockprüfzeichen
- **NAK** Negative Acknowledge; negative Rückmeldung

Hinweis

Wird als Informationszeichen das Zeichen DLE übertragen, so wird dieses zur Unterscheidung vom Steuerzeichen DLE beim Verbindungsaufbau und -abbau auf der Sendeleitung doppelt gesendet (DLE-Verdopplung). Der Empfänger macht die DLE-Verdopplung wieder rückgängig.

Priorität

Bei der Prozedur 3964(R) muss dem einen Kommunikationspartner eine höhere und dem anderen Kommunikationspartner eine niedrigere Priorität zugeordnet sein. Wenn beide Kommunikationspartner gleichzeitig mit dem Verbindungsaufbau beginnen, dann stellt der Partner mit niedriger Priorität seinen Sendeauftrag zurück.

Blockprüfsumme

Blockprüfsumme

Beim Übertragungsprotokoll 3964(R) wird die Datensicherheit durch ein zusätzlich gesendetes Blockprüfzeichen (BCC = Block Check Character) erhöht.

Telegramm:

STX	Daten	DLE ETX	BCC
02H →	30H → 31H → 32H →	10H → 03H →	20H

30	=	0011	0000
31	=	0011	0001
XOR	=	0000	0001
32	=	0011	0010
XOR	=	0011	0011
10	=	0001	0000
XOR	=	0010	0011
03	=	0000	0011
XOR	=	0010	0000
BCC	→	2	0

Bild 1-50 Blockprüfsumme

Die Blockprüfsumme ist die gerade Längsparität (EXOR-Verknüpfung aller Datenbytes) eines gesendeten bzw. empfangenen Blocks. Die Bildung beginnt mit dem ersten Nutzdatenbyte (1.

Byte des Telegramms) nach dem Verbindungsaufbau und endet nach dem Zeichen DLE ETX beim Verbindungsabbau.

Hinweis

Bei einer DLE-Verdopplung wird das Zeichen DLE zweimal in die BCC-Bildung einbezogen.

Daten senden mit 3964(R)

Ablauf der Datenübertragung beim Senden

Im folgenden Bild ist der Ablauf der Datenübertragung beim Senden mit der Prozedur 3964(R) dargestellt.

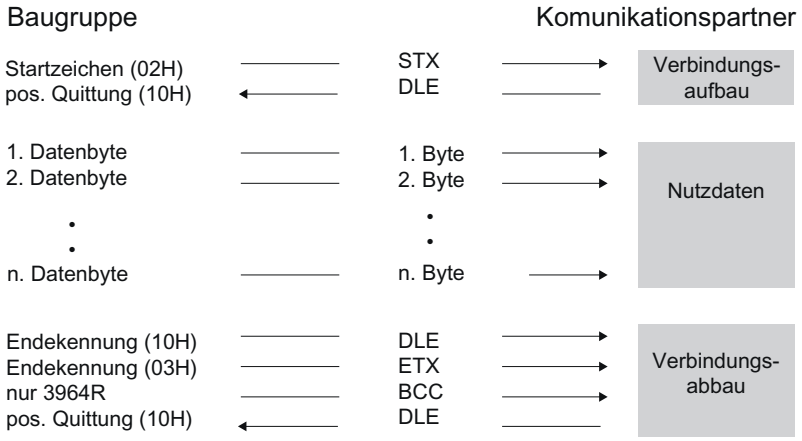

Bild 1-51 Datenverkehr beim Senden mit der Prozedur 3964(R)

Verbindungsaufbau beim Senden

Zum Aufbau der Verbindung sendet die Prozedur 3964(R) das Steuerzeichen STX. Antwortet der Kommunikationspartner vor Ablauf der Quittungsverzugszeit (QVZ) mit dem Zeichen DLE, geht die Prozedur in den Sendebetrieb über.

Antwortet der Kommunikationspartner mit NAK, einem beliebigen anderen Zeichen (außer DLE oder STX) oder die Quittungsverzugszeit verstreicht ohne Reaktion, wiederholt die Prozedur den Verbindungsaufbau. Nach der parametrierten Anzahl vergeblicher Aufbauversuche bricht die Prozedur den Verbindungsaufbau ab und sendet das Zeichen NAK an den Kommunikationspartner. Die Kommunikationsbaugruppe trägt in ihrem Fehlermeldebereich eine entsprechende Fehlernummer ein.

Daten senden

Gelingt der Verbindungsaufbau, werden die im Ausgabepuffer des Kommunikationsprozessors enthaltenen Nutzdaten mit den gewählten Übertragungsparametern an den Kommunikationspartner gesendet. Dieser überwacht den zeitlichen Abstand der ankommenden Zeichen. Der Abstand zwischen zwei Zeichen darf nicht mehr als die Zeichenverzugszeit (ZVZ) betragen.

Sendet der Kommunikationspartner während einer laufenden Sendung das Zeichen NAK, bricht die Prozedur den Block ab und wiederholt ihn in der oben beschriebenen Weise, beginnend mit dem Verbindungsaufbau. Bei einem anderen Zeichen wartet die Prozedur zunächst auf den Ablauf der Zeichenverzugszeit und sendet anschließend NAK, um den Kommunikationspartner in den Ruhezustand zu bringen. Danach beginnt die Prozedur das Senden erneut mit dem Verbindungsaufbau STX.

Verbindungsabbau beim Senden

Nach dem Senden des Pufferinhalts fügt die Prozedur die Zeichen DLE, ETX und nur bei 3964R die Blockprüfsumme BCC als Endekennung an und wartet auf ein Quittungszeichen. Sendet der Kommunikationspartner innerhalb der Quittungsverzugszeit das Zeichen DLE, wurde der Datenblock fehlerfrei übernommen. Antwortet der Kommunikationspartner mit NAK, einem beliebigen anderen Zeichen (außer DLE), einem gestörten Zeichen oder die Quittungsverzugszeit verstreicht ohne Reaktion, beginnt die Prozedur das Senden erneut mit dem Verbindungsaufbau STX.

Nach Ablauf der parametrierten Anzahl von Übertragungsversuchen, den Datenblock zu senden, bricht die Prozedur das Verfahren ab und sendet NAK an den Kommunikationspartner. Die Kommunikationsbaugruppe meldet den Fehler im Fehlermeldebereich.

Abläufe beim Senden mit der Prozedur 3964(R)

Im folgenden Bild finden Sie die Abläufe beim Senden mit der Prozedur 3964(R).

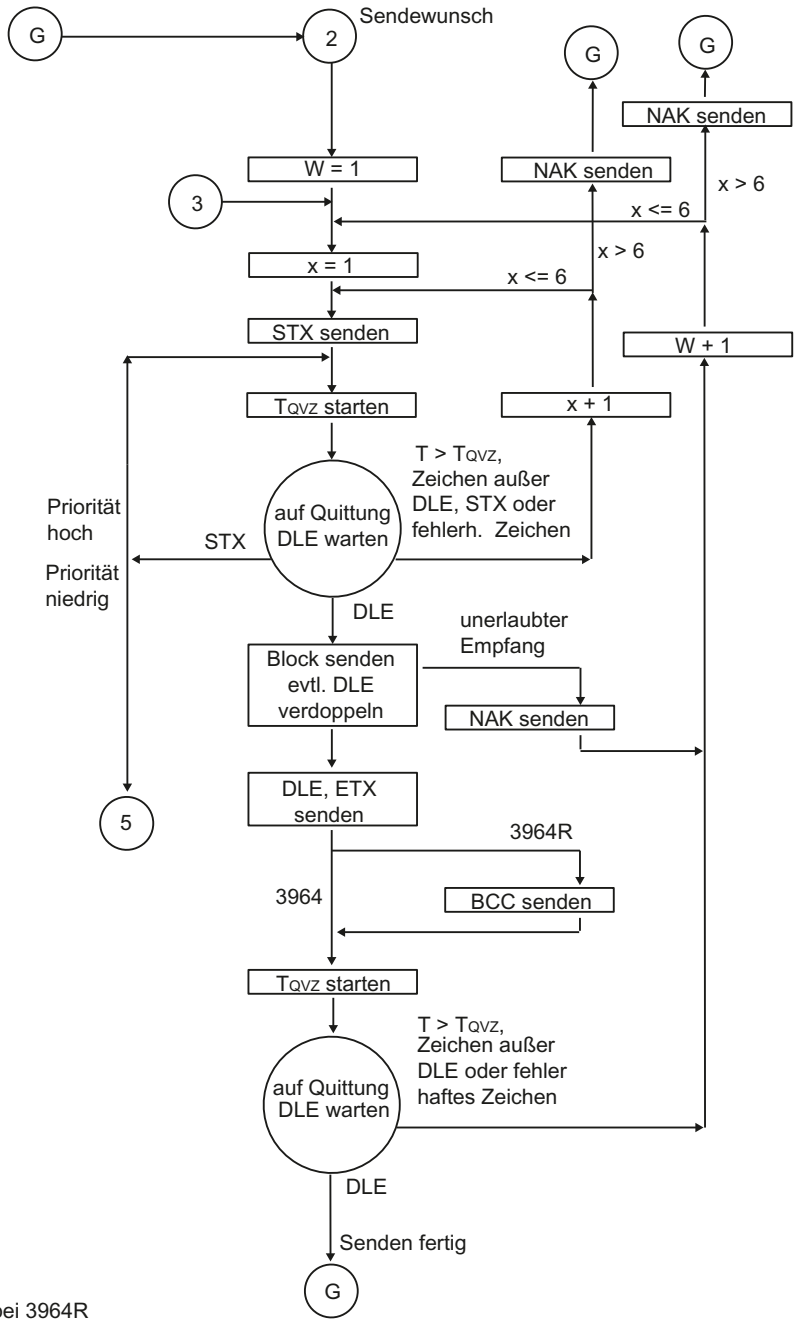

BCC nur bei 3964R
x = Zähler Aufbauversuche
TQVZ=500 ms (3964R TQVZ=2s)
W = Zähler Übertragungsversuche
bei Leitungsbruch BREAK sofort in Grundzustand

Bild 1-52 Ablaufschema beim Senden mit der Prozedur 3964(R)

Daten empfangen mit 3964(R)

Ablauf der Datenübertragung beim Empfangen

Im folgenden Bild ist der Ablauf der Datenübertragung beim Empfangen mit der Prozedur 3964(R) dargestellt.

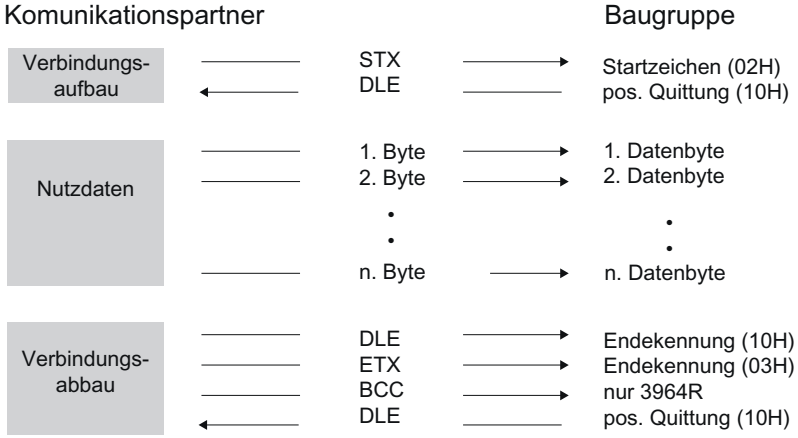

Bild 1-53 Datenverkehr beim Empfangen mit Prozedur 3964(R)

Hinweis

Die Prozedur 3964(R) sendet, sobald sie betriebsbereit ist, einmal das Zeichen NAK an den Partner, um den Kommunikationspartner in den Ruhezustand zu bringen.

Verbindungsaufbau beim Empfangen

Im Ruhezustand, wenn kein Sendeauftrag zu bearbeiten ist, wartet die Prozedur auf den Aufbau der Verbindung durch den Kommunikationspartner.

Steht beim Verbindungsaufbau mit STX kein leerer Empfangspuffer zur Verfügung, wird eine Wartezeit von 400 ms gestartet. Liegt nach dieser Zeit noch kein leerer Empfangspuffer vor, meldet das Systemprogramm den Fehler im Fehlermeldebereich der Baugruppe. Die Prozedur sendet ein Zeichen NAK und geht wieder in den Ruhezustand zurück. Andernfalls sendet die Prozedur das Zeichen DLE und empfängt die Daten in der oben beschriebenen Weise.

Empfängt die Prozedur im Ruhezustand ein beliebiges Zeichen (außer STX oder NAK), wartet sie auf den Ablauf der Zeichenverzugszeit (ZVZ) und sendet dann ein Zeichen NAK. Die Kommunikationsbaugruppe meldet den Fehler im Fehlermeldebereich der Baugruppe.

Daten empfangen

Nach gelungenem Verbindungsaufbau werden die ankommenden Empfangszeichen im Empfangspuffer abgelegt. Werden zwei aufeinanderfolgende Zeichen DLE empfangen, wird nur ein Zeichen DLE in den Empfangspuffer übernommen.

Nach jedem Empfangszeichen wird während der Zeichenverzugszeit auf das nächste Zeichen gewartet. Verstreicht die Zeichenverzugszeit ohne Empfang, wird das Zeichen NAK an den

Kommunikationspartner gesendet. Die Kommunikationsbaugruppe meldet den Fehler im Fehlermeldebereich der Baugruppe. Die Prozedur 3964(R) stößt keine Wiederholung an.

Treten während des Empfangens Übertragungsfehler auf (verlorenes Zeichen, Rahmenfehler, Paritätsfehler usw.), so wird bis zum Verbindungsabbau weiter empfangen und dann NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet. Kann der Block auch nach der im statischen Parametersatz spezifizierten Anzahl von Übertragungsversuchen nicht fehlerfrei empfangen werden oder wird die Wiederholung vom Kommunikationspartner nicht innerhalb einer Blockwartezeit von 4 s gestartet, bricht die Prozedur den Empfang ab. Die Kommunikationsbaugruppe meldet die erste fehlerhafte Übertragung und den endgültigen Abbruch im Fehlermeldebereich der Baugruppe.

Verbindungsabbau beim Empfangen

Erkennt die Prozedur 3964 die Zeichenfolge DLE ETX, beendet sie den Empfang und sendet DLE für einen fehlerfrei empfangenen Block an den Kommunikationspartner. Bei einem Empfangsfehler wird NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet.

Erkennt die Prozedur 3964R die Zeichenfolge DLE ETX BCC, beendet sie den Empfang. Sie vergleicht das empfangene Blockprüfzeichen BCC mit der intern gebildeten Längsparität. Ist das Blockprüfzeichen korrekt und kein anderer Empfangsfehler aufgetreten, sendet die Prozedur 3964R DLE und kehrt zurück in den Ruhezustand. Bei fehlerhaftem BCC oder einem anderen Empfangsfehler wird NAK an den Kommunikationspartner gesendet. Anschließend wird eine Wiederholung erwartet.

Prozedur 3964(R) Empfangen

In den folgenden Bildern finden Sie die Abläufe beim Empfangen mit der Prozedur 3964(R).

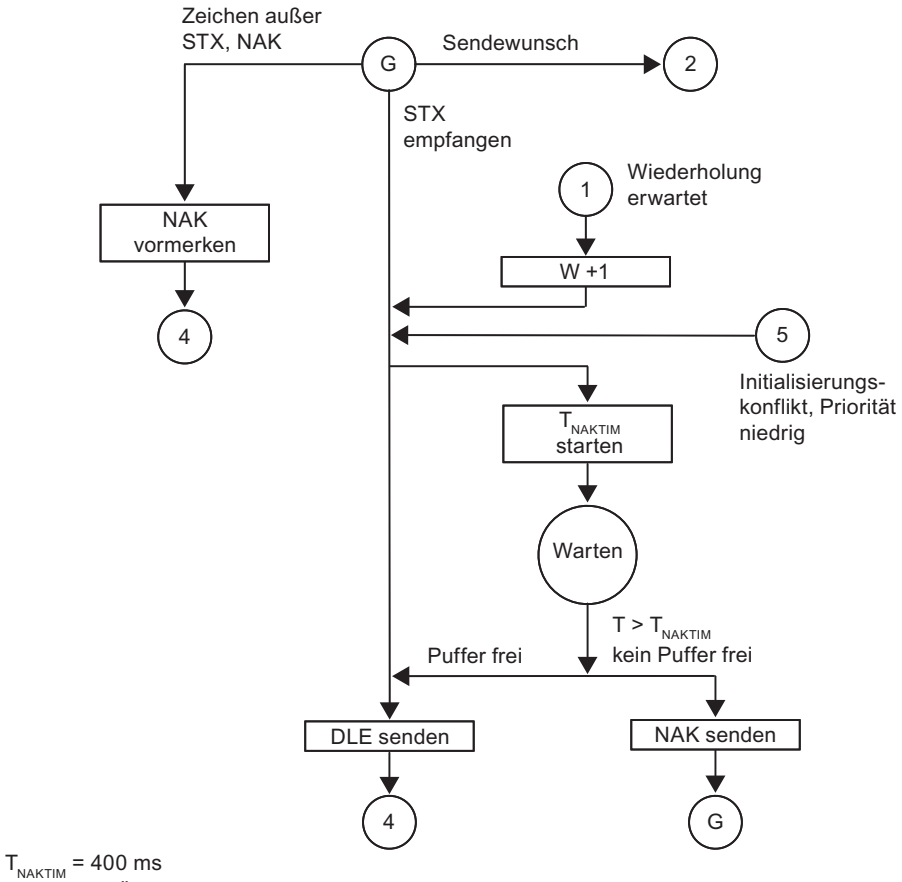

T_{NAKTIM} = 400 ms
W = Zähler Übertragungsversuche
bei Leitungsbruch BREAK sofort in Grundzustand

Bild 1-54 Ablaufschema beim Empfangen mit der Prozedur 3964(R) (Teil 1)

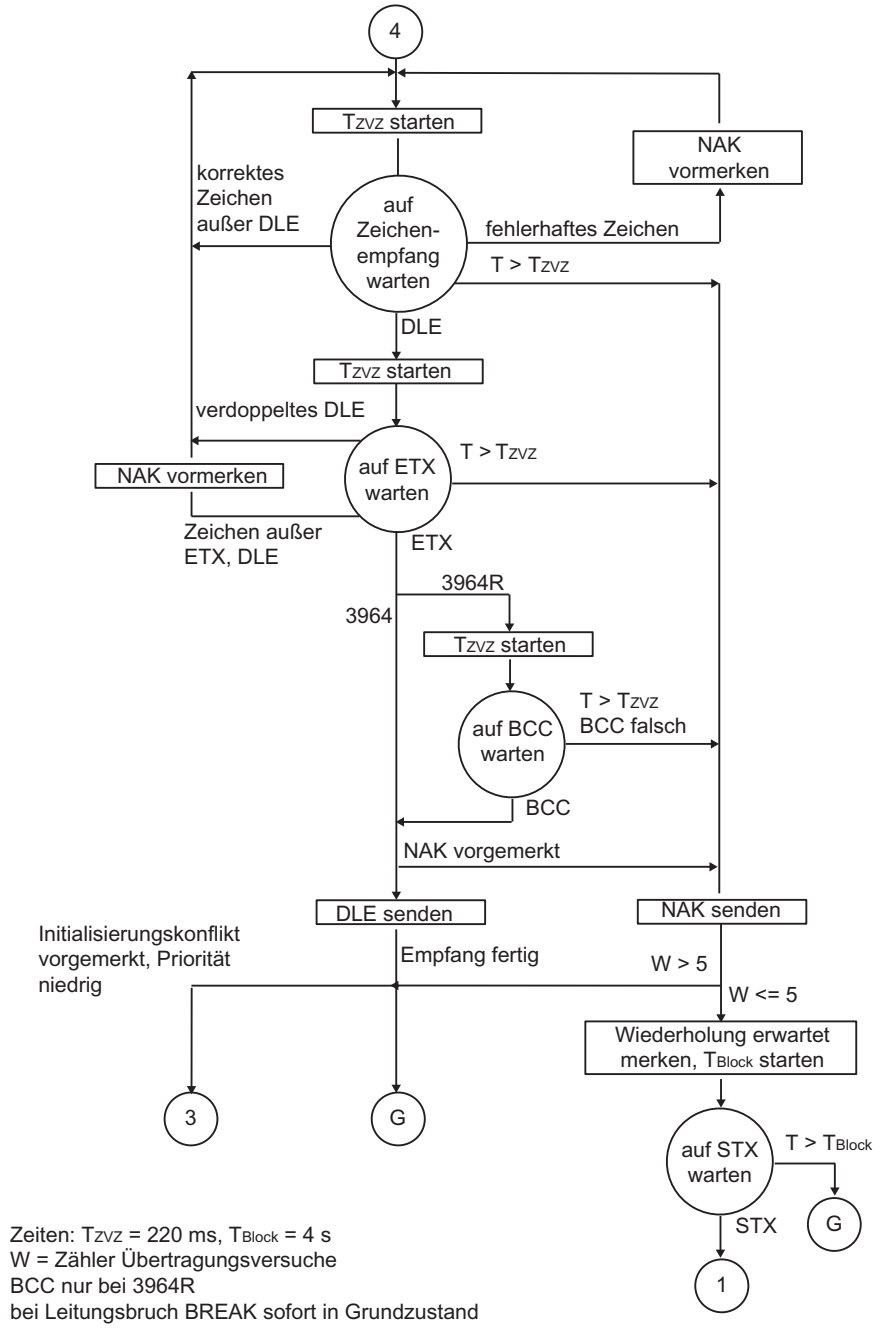

Bild 1-55 Ablaufschema beim Empfangen mit der Prozedur 3964(R) (Teil 2)

Datenübertragung mit RK512

Datenübertragung mit der Rechnerkopplung RK512

Zuordnung der Protokolle zu den Kommunikationsbaugruppen siehe: Übersicht der Komponenten und Schnittstellen (Seite 1670)

Einleitung

Die Rechnerkopplung RK512 steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen der Kommunikationsbaugruppe und einem Kommunikationspartner.

Im Unterschied zur Prozedur 3964(R) enthält die Rechnerkopplung RK512 neben der Bit-Übertragungsschicht (Schicht 1) und der Sicherungsschicht (Schicht 2) zusätzlich die Transportschicht (Schicht 4). Die Rechnerkopplung RK512 bietet darüber hinaus eine höhere Datensicherheit und erweiterte Adressierungsmöglichkeiten.

Reaktionstelegramm

Die Rechnerkopplung RK512 beantwortet jedes korrekt empfangene Befehlstelegramm mit einem Reaktionstelegramm an den Auftraggeber (Transportschicht). Damit kann der Absender prüfen, ob seine Daten fehlerfrei bei der CPU des Kommunikationspartners angekommen sind bzw. ob seine angeforderten Daten bei der CPU des Kommunikationspartners verfügbar sind.

Befehlstelegramm

Befehlstelegramme sind entweder SEND-Telegramme oder FETCH-Telegramme.

SEND-Telegramm

Bei einem SEND-Telegramm sendet die Kommunikationsbaugruppe ein Befehlstelegramm mit Nutzdaten und der Kommunikationspartner antwortet mit einem Reaktionstelegramm ohne Nutzdaten.

FETCH-Telegramm

Bei einem FETCH-Telegramm sendet die Kommunikationsbaugruppe ein Befehlstelegramm ohne Nutzdaten und der Kommunikationspartner antwortet mit einem Reaktionstelegramm mit Nutzdaten.

Folgetelegramm

Überschreitet die Datenmenge 128 Bytes, so werden bei SEND- und FETCH-Telegrammen automatisch Folgetelegramme gesendet.

Telegrammkopf

Jedes Telegramm bei RK512 beginnt mit einem Telegrammkopf. Er kann Telegrammkennungen, Angaben über Datenziel, -quelle und eine Fehlernummer enthalten.

Aufbau des Telegrammkopfs

In der folgenden Tabelle finden Sie den Aufbau des Telegrammkopfs des Befehlstelegramms.

Tabelle 1-108 Aufbau des Telegrammkopfs des Befehlstelegramms (RK512)

Byte	Bedeutung
1	Die Telegrammkennung bei Befehlstelegrammen (00H), bei Folge-Befehlstelegrammen (FFH)
2	Telegrammkennung (00H)
3	'A' (41H) für SEND-Auftrag mit Ziel-DB oder 'O' (4FH) für SEND-Auftrag mit Ziel-DX oder 'E' (45H) für FETCH-Auftrag
4	zu übertragende Daten sind aus (beim Senden ist nur 'D' möglich): 'D' (44H) = Datenbaustein 'E' (45H) = Eingangsbytes 'A' (41H) = Ausgangsbytes 'M' (4DH) = Merkerbytes 'Z' (5AH) = Zählerzellen 'T' (54H) = Zeitzellen
5 und 6	Datenziel bei SEND/PUT-Auftrag bzw. Datenquelle bei GET-Auftrag z. B. Byte 5 = DB-Nr., Byte 6 = DW-Nr. (Die RK512-Adressierung beschreibt Datenquelle und -ziel mit Wortgrenzen. Die Umrechnung auf Byteadressen in SIMATIC S7 erfolgt automatisch.)
7 und 8	Länge High-Byte Länge der zu übertragenden Daten je nach Typ in Byte oder Länge Low-Byte Wörtern
9	Bytenummer des Koppelmerkers; falls Sie keinen Koppelmerker angegeben haben, dann steht hier FFH.
10	**Bit 0 bis 3:** Bitnummer des Koppelmerker, falls Sie keinen Koppelmerker angegeben haben, dann trägt das Protokoll hier FH ein. **Bit 4 bis 7:** CPU-Nummer (Zahl von 1 bis 4); wenn Sie keine CPU-Nr., aber einen Koppelmerker angegeben haben, dann steht hier 0H; wenn Sie keine CPU-Nr. und keinen Koppelmerker angegeben haben, dann steht hier FH.

Die Buchstaben in Byte 3 und 4 sind ASCII-Zeichen.

Der Telegrammkopf des Folge-Befehlstelegramms besteht nur aus den Bytes 1 bis 4.

Reaktionstelegramm

Nachdem das Befehlstelegramm übertragen wurde, erwartet die RK512 innerhalb einer festen Überwachungszeit ein Reaktionstelegramm des Kommunikationspartners. Die Länge der Überwachungszeit beträgt unabhängig von der Übertragungsgeschwindigkeit (Baudrate) 20 s.

Abhängig von der verwendeten PtP Baugruppe kann diese Überwachungszeit durch Parametrierung des Anwenders im Eigenschaftendialog der Baugruppe reduziert werden.

Durch Auswahl der Option "baudratenabhängig" wird dann mit den nachfolgend aufgeführten maximalen Wartezeiten überwacht:

Tabelle 1-109 Baudratenabhängige Überwachungszeit auf Reaktionstelegramm

Baudrate	Überwachungszeit
• 300 Baud	10 s
• 600 Baud	7 s
• 1200 Baud	5 s
• ab 38400 Baud	3 s

Das "gegraute" Feld "maximale Wartezeit" dient nur der Anzeige der verwendeten Überwachungszeit und ist nicht editierbar!

Aufbau und Inhalt des Reaktionstelegramms

Das Reaktionstelegramm besteht aus 4 Bytes und enthält Informationen über den Verlauf des Auftrages.

Byte	Bedeutung
1	Telegrammkennung bei Reaktionstelegrammen (00H), bei Folge-Reaktionstelegrammen (FFH)
2	Telegrammkennung (00H)
3	belegt mit 00H
4	Fehlernummer des Kommunikationspartners im Reaktionstelegramm: • **00H** wenn bei Übertragung kein Fehler aufgetreten ist • **> 00H** Fehlernummer Die Fehlernummer im Reaktionstelegramm bewirkt automatisch den Eintrag einer Fehlernummer im Fehlermeldebereich der Baugruppe.

Daten senden mit RK512

Ablauf der Datenübertragung beim Senden

Im folgenden Bild ist der Ablauf der Datenübertragung beim Senden mit einem Reaktionstelegramm bei der Rechnerkopplung RK512 dargestellt.

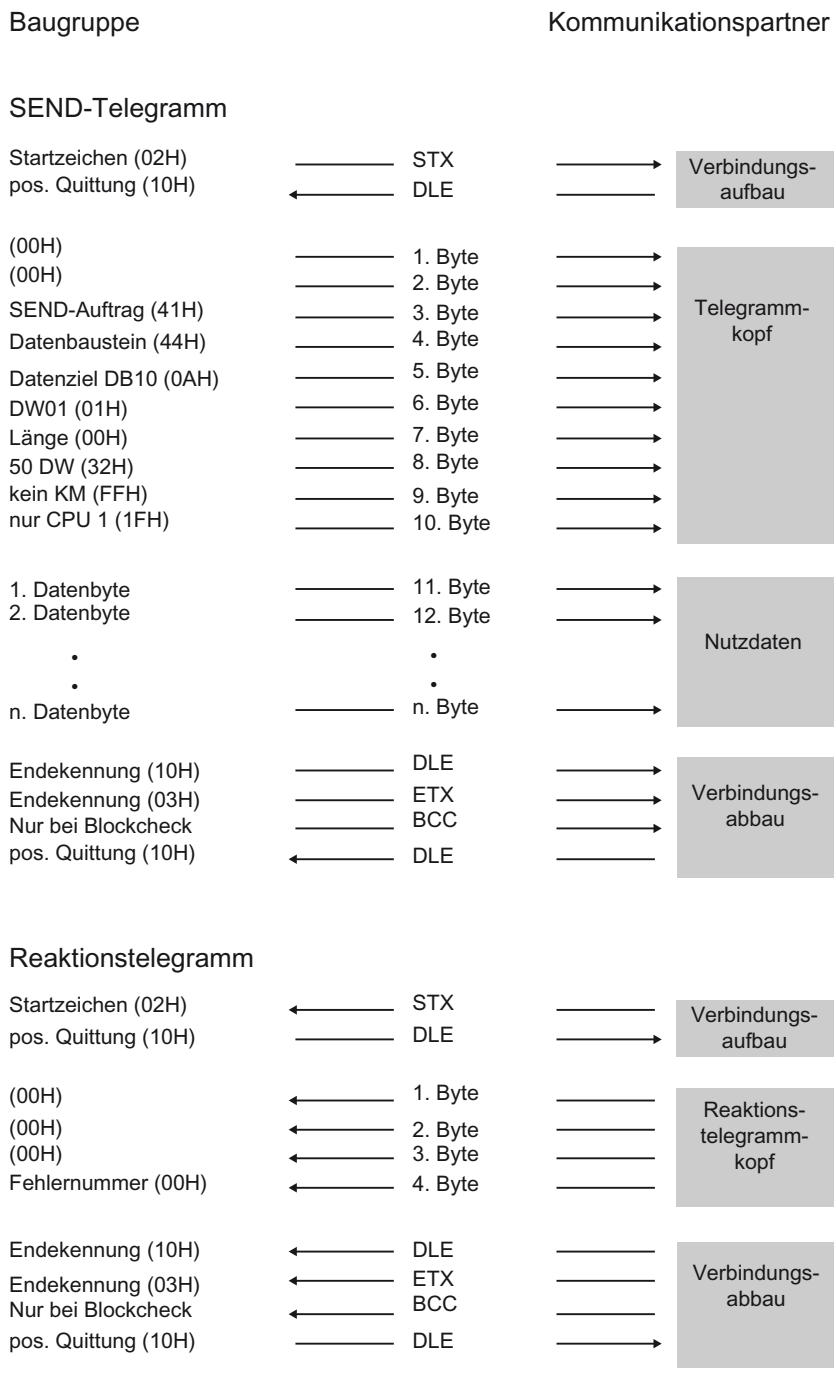

Bild 1-56 Datenverkehr beim Senden mit Reaktionstelegramm

Daten senden

Der SEND-Auftrag läuft in folgender Reihenfolge ab:

- Aktiver Partner
  Sendet ein SEND-Telegramm ab. Dieses enthält den Telegrammkopf und Daten.
- Passiver Partner
  Empfängt das Telegramm, überprüft den Telegrammkopf sowie die Daten und quittiert mit einem Reaktionstelegramm nach Übergabe der Daten an die CPU.
- Aktiver Partner
  Empfängt das Reaktionstelegramm und beendet den SEND-Auftrag.

Hinweis

Falls die Nutzdaten des SEND-Auftrags nicht fehlerfrei von der CPU entgegengenommen wurden oder ein Fehler im Telegrammkopf aufgetreten ist, trägt der Kommunikationspartner eine Fehlernummer ins 4. Byte des Reaktionstelegramms ein. Bei Protokollfehlern erfolgt kein Eintrag im Reaktionstelegramm.

Folge-SEND-Telegramme

Ein Folge-SEND-Auftrag wird gestartet, wenn die Datenmenge 128 Bytes überschreitet. Der Ablauf entspricht dem des SEND-Auftrags.

Werden mehr als 128 Bytes gesendet, werden diese automatisch in einem oder mehreren Folgetelegramm(en) übertragen. Jedes Folge-SEND-Telegramm wird mit einem Folge-Reaktionstelegramm quittiert.

Im folgenden Bild ist der Ablauf der Datenübertragung beim Senden eines Folge-SEND-Telegramms mit einem Folge-Reaktionstelegramm dargestellt.

Das Folge-Reaktionstelegramm auf das letzte Folge-SEND-Telegramm wird erst nach erfolgreicher Übergabe der gesamten Nutzdaten an die CPU gesendet.

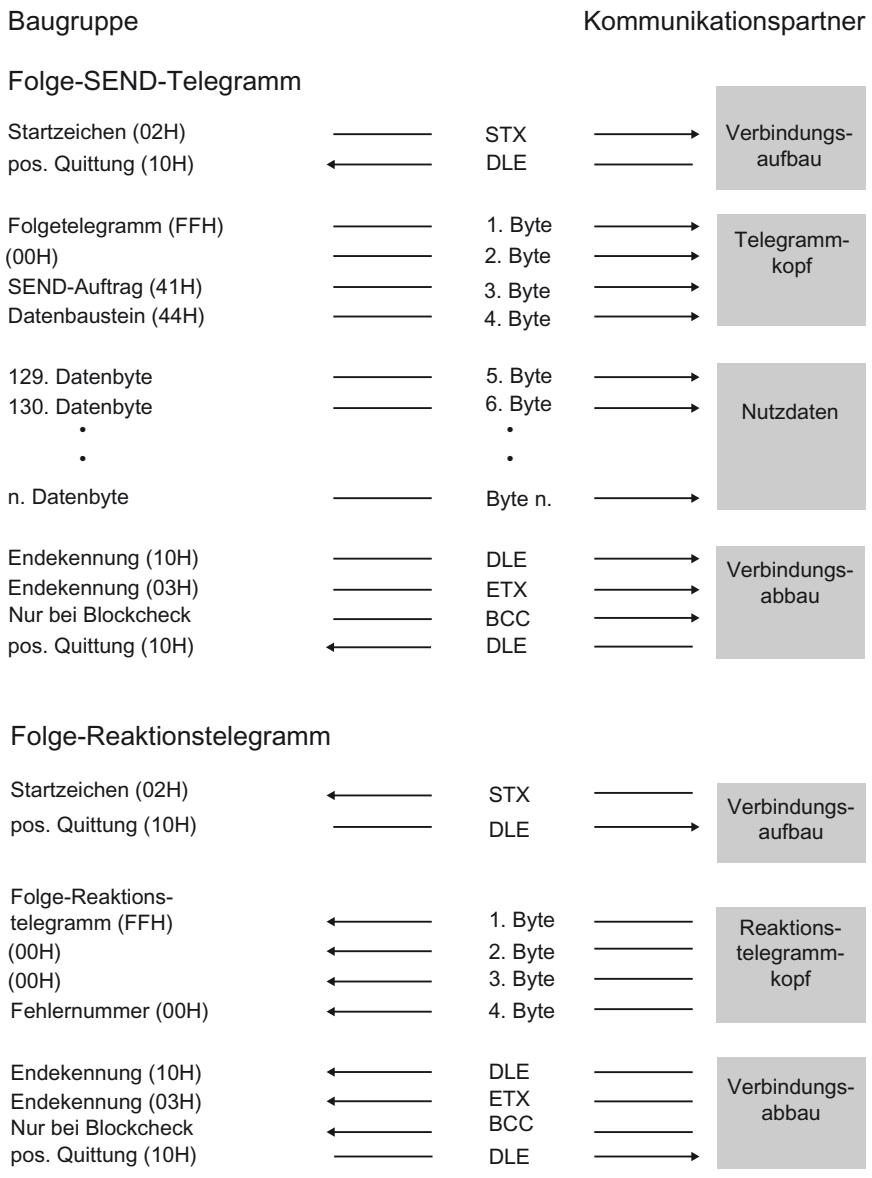

Bild 1-57 Ablauf eines Folge-SEND-Telegramms mit einem Folge-Reaktionstelegramm

Daten holen mit RK512

Ablauf beim Daten holen mit RK512

Im folgenden Bild ist der Ablauf der Datenübertragung beim Datenholen mit einem Reaktionstelegramm bei der Rechnerkopplung RK512 dargestellt.

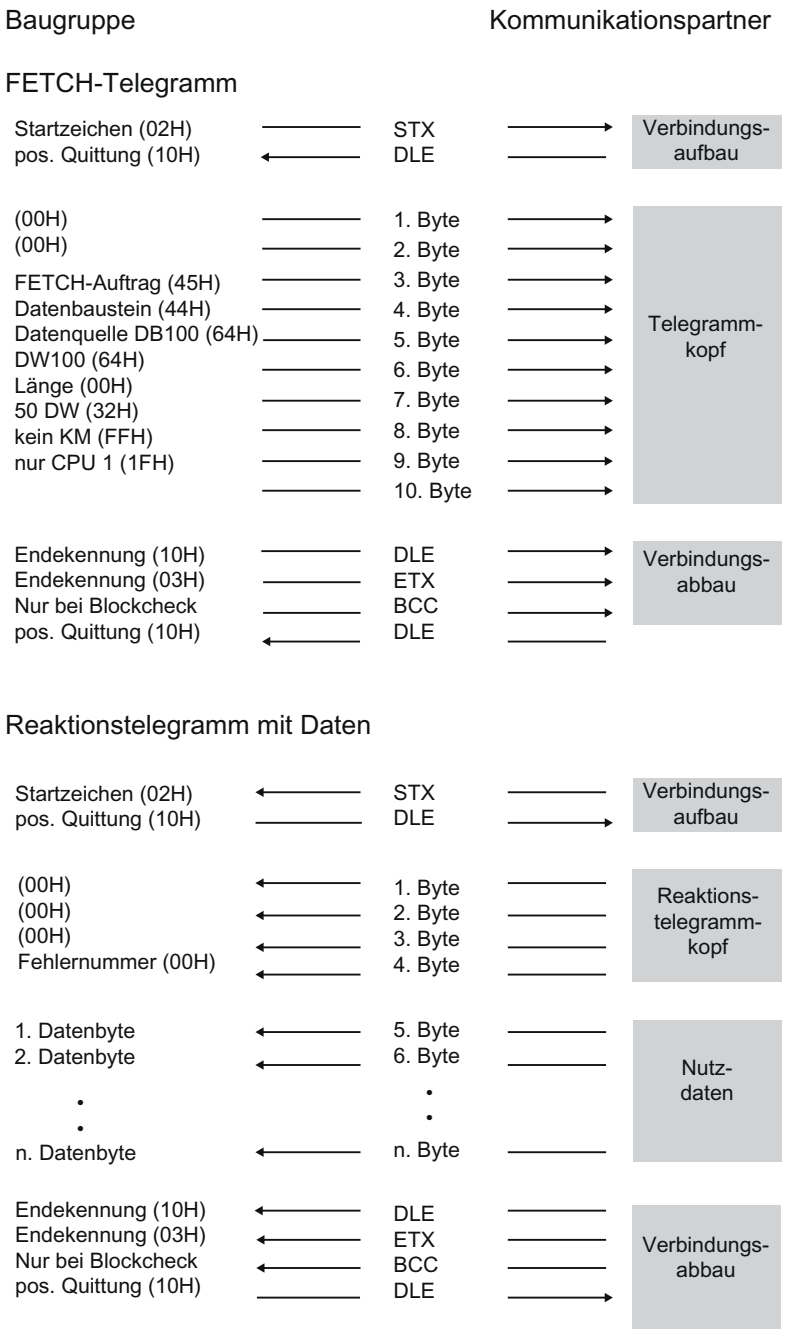

Bild 1-58 Datenverkehr beim Holen mit Reaktionstelegramm

Daten holen

Der FETCH-Auftrag läuft in folgender Reihenfolge ab:

- Aktiver Partner
  Sendet ein FETCH-Telegramm ab. Dieses enthält den Telegrammkopf.
- Passiver Partner
  Empfängt das Telegramm, überprüft den Telegrammkopf, holt die Daten aus der CPU und quittiert mit einem Reaktionstelegramm. Dieses enthält die Daten.
- Aktiver Partner
  Empfängt das Reaktionstelegramm, übergibt die im Reaktionstelegramm empfangenen Nutzdaten an die CPU und beendet den FETCH-Auftrag.

Bei einer Fehlernummer (ungleich 0) im 4. Byte enthält das Reaktionstelegramm keine Daten.

Werden mehr als 128 Bytes angefordert, werden diese automatisch in einem bzw. mehreren Folgetelegramm(en) geholt. Jedes Folge FETCH-Telegramm wird mit einem Folge-Reaktionstelegramm (mit weitere Nutzdaten) quittiert.

Nach Empfang des letzten Folge-Reaktionstelegramms (mit dem letzten Nutzdatenblock) werden die gesamten Nutzdaten an die CPU übergeben und der FETCH-Auftrag beendet.

Hinweis

Falls das FETCH-Telegramm nicht fehlerfrei von der CPU bedient werden konnte oder ein Fehler im Telegrammkopf aufgetreten ist, trägt der Kommunikationspartner eine Fehlernummer ins 4. Byte des Reaktionstelegramms ein. Bei Protokollfehlern erfolgt kein Eintrag im Reaktionstelegramm.

Folge-FETCH-Telegramm

Im folgenden Bild ist der Ablauf der Datenübertragung beim Holen von Daten mit einem Folge-Reaktionstelegramm dargestellt.

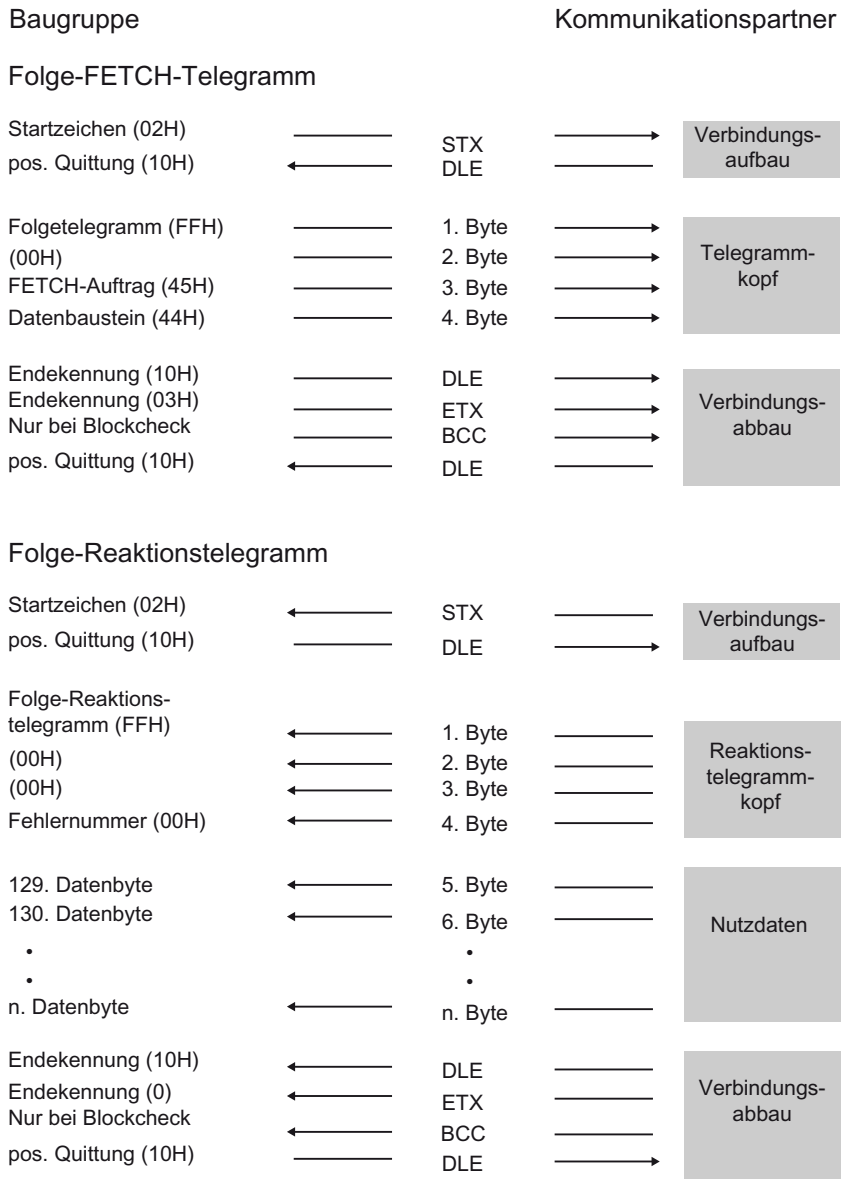

Bild 1-59 Ablauf eines Folge-FETCH-Telegramms mit einem Folge-Reaktionstelegramm

Quasi-Vollduplexbetrieb

Quasi-Vollduplex bedeutet: Die Partner können zu beliebigen Zeitpunkten Befehls- und Reaktionstelegramme senden, außer wenn der andere Partner gerade sendet. Die maximale Schachtelungstiefe für Befehls- und Reaktionstelegramme beträgt "1". Ein weiteres Befehlstelegramm kann also erst bearbeitet werden, wenn das vorige mit einem Reaktionstelegramm beantwortet wurde.

Unter Umständen kann - wenn beide Partner senden wollen - vor dem Reaktionstelegramm ein SEND- oder FETCH-Telegramm des Partners übertragen werden. Beispielsweise dann, wenn in den Ausgabepuffer der Kommunikationsbaugruppe vor dem Reaktionstelegramm ein SEND- oder FETCH-Telegramm des Partners eingetragen wurde.

Im folgenden Bild wird das Folge-Reaktionstelegramm zum ersten SEND-Telegramm dann erst nach dem SEND-Telegramm des Partners gesendet.

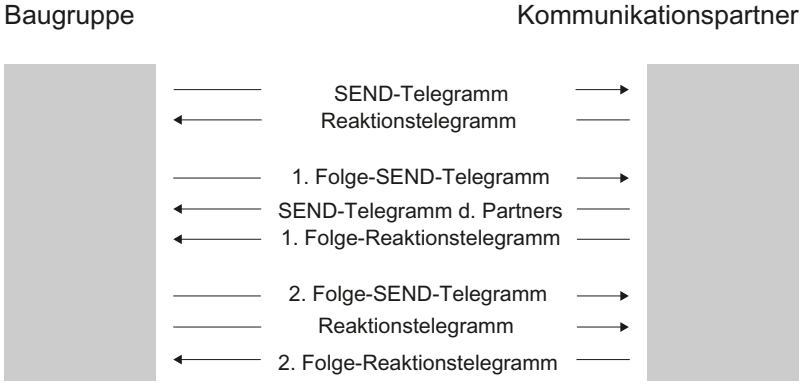

Bild 1-60 Quasi-Vollduplexbetrieb

RK512 CPU-Aufträge

Im folgenden Bild finden Sie die Abläufe der Rechnerkopplung RK512 durch CPU-Aufträge.

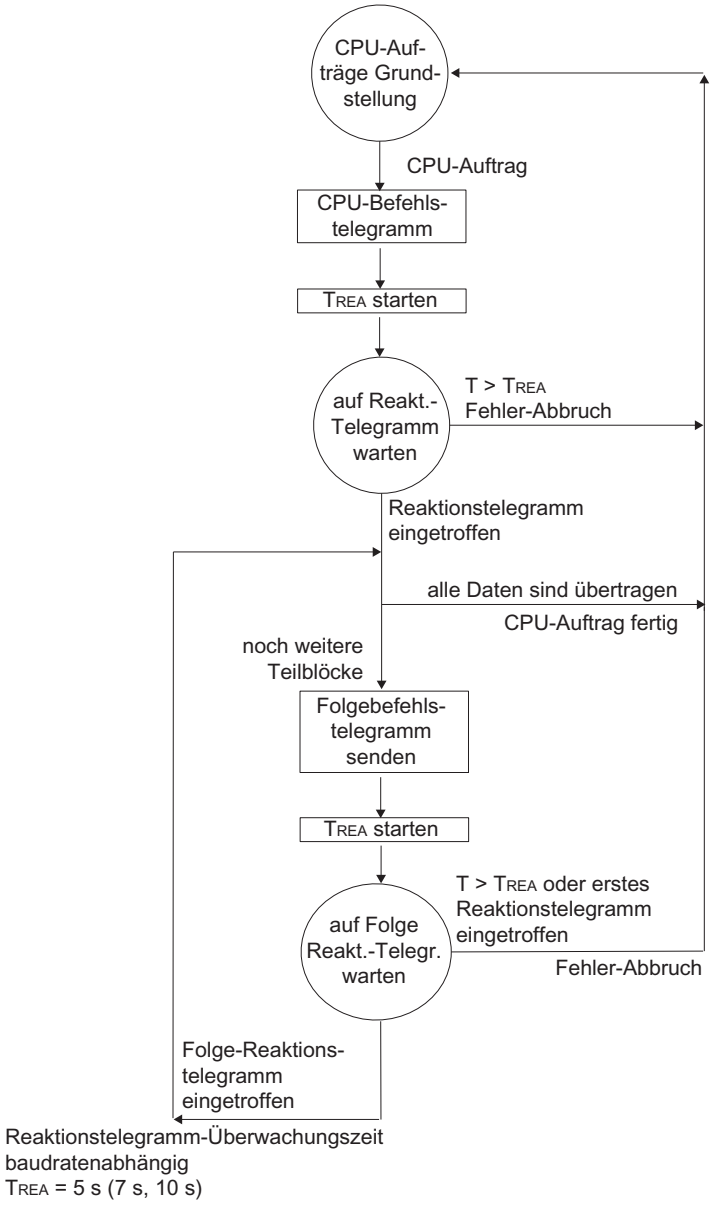

Bild 1-61 Ablauf der Datenübertragung mit der RK512 durch CPU-Aufträge

RK512 Partner-Aufträge

Im folgenden Bild finden Sie die Abläufe der Rechnerkopplung RK512 durch Partner-Aufträge.

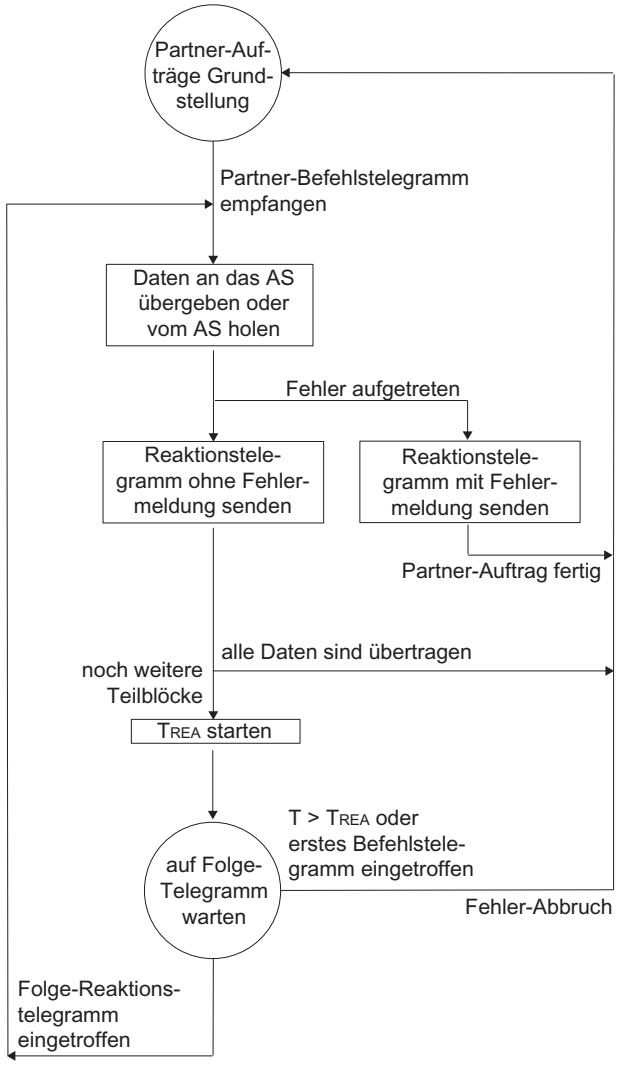

Reaktionstelegramm-Überwachungszeit baudratenabhängig
T_{REA} = 5 s (7 s, 10 s)
AS = Automatisierungssystem

Bild 1-62 Ablaufschema bei der Datenübertragung mit der RK512 durch Partner-Aufträge

Datenübertragung mit Drucker-Treiber

Datenübertragung mit dem Drucker-Treiber

Zuordnung der Protokolle zu den Kommunikationsbaugruppen siehe: Übersicht der Komponenten und Schnittstellen (Seite 1670)

Einleitung

Mit dem Drucker-Treiber können Sie Meldetexte mit Uhrzeit- und Datumsangabe auf einen Drucker ausgeben. Sie können so einfache Prozesse kontrollieren, Fehler- oder Störmeldungen ausdrucken oder z. B. auch Anweisungen für das Bedienpersonal vorgeben.

Der Drucker-Treiber beinhaltet die Bitübertragungsschicht (Schicht 1).

Meldetexte und Parameter für Druckerausgabe

Im Eigenschaftendialog der Baugruppe projektieren Sie die Meldetexte und legen Sie die Parameter (Seitenlayout, Zeichensatz, Steuerzeichen) für die Druckerausgabe fest. Meldetexte und Parameter für die Druckerausgabe werden im Anlauf der Kommunikationsbaugruppe zusammen mit den Baugruppenparametern zur Kommunikationsbaugruppe übertragen.

- **Meldetexte:**
  Sie können Meldetexte mit Variablen und Steueranweisungen (wie z. B. für Fett-, Schmal-, Breit- oder Kursivschrift und Unterstrichen) projektieren. Jeder Meldetext wird bei der Projektierung mit einer Nummer versehen. Die Druckerausgabe eines bestimmten Meldetextes erfolgt bei den CP 441-Baugruppen durch die Angabe einer Referenz (auf die Speicherzelle, welche die Meldetext-Nr. beinhaltet) an den Sendeparametern SD_1 bis SD_4 der Anweisung. Bei den CP 34x-Baugruppen ist sie durch die Angabe der Meldetextnummer in einem Formatstring beim Aufruf der Anweisung festgelegt.
  Projektierte Meldetexte können Sie auch in einer externen Text-Datei (SDB-Name; max. 8 Zeichen ASCII) speichern und mit einem geeigneten Editor bearbeiten. Auf diese Weise können die Texte z. B. übersetzt und anschließend wieder eingelesen werden.
  Aufbau der Meldetextdatei:
  <Versionsnummer>
  <Nummer der Meldung>:<Inhalt der Meldung>
- **Seitenlayout:**
  Für das Seitenlayout können Sie die Seitenränder, mögliche Zeilenumbrüche sowie Kopf- und Fußzeilen projektieren.
- **Zeichensatz:**
  Mit einer Zeichenwandeltabelle wird der ANSI-Zeichensatz von STEP 7 auf den Druckerzeichensatz umgesetzt. Eine zum Druckertyp vorgeschlagene Zeichenwandeltabelle können Sie verändern, um z. B. nationale Sonderzeichen aufzunehmen.
- **Steuerzeichen:**
  Über eine Steuerzeichentabelle können Sie die Steueranweisungen im Meldetext für die Druckeremulation zum Ein- und Ausschalten von Fett-, Schmal-, Breit- oder Kursivschrift und Unterstrichen verändern und um weitere Steuerzeichen erweitern.

Variablen

In einem Meldetext können bis zu 4 Variablen (3 + Meldetextnummer) eingeblendet werden. Variablenwerte können von der CPU auf die Kommunikationsbaugruppe übertragen werden. Als Variablen können eingeblendet werden:

- Rechenwerte des Anwenderprogramms (z. B.: Füllstände),
- Datum- und Uhrzeit,

- Strings (Stringvariable)
- weitere Meldetexte.

Für jede Variable muss im projektierten Meldetext oder im Formatstring eine Konvertierungsanweisung angegeben werden, in der die Bedeutung und das Ausgabeformat des Variablenwertes verschlüsselt sind.

Formatstring

Durch den Formatstring definieren Sie die Darstellung und die Zusammensetzung eines Meldetextes. Der Formatstring kann bestehen aus:

- Text (alle druckbaren Zeichen, z. B: Um ... Uhr wurde der Füllstand ... l erreicht!)
- Konvertierungsanweisungen für Variablen (z. B. %N = Zeiger auf die Meldetextnummer x, wobei x ein Variablenwert ist (siehe unten Beispiel 2))
  Für jede Variable muss genau eine Konvertierungsanweisung im Formatstring oder im projektierten Meldetext vorhanden sein. Die Konvertierungsanweisungen werden entsprechend ihrer Reihenfolge auf die Variablen angewendet.
- Steueranweisungen mit Steuerzeichen für Fett-, Schmal-, Breit-, Kursivschrift, Unterstrichen (z. B. \B = Fettdruck ein) bzw. mit von Ihnen zusätzlich definierten Steuerzeichen

Sie können zusätzliche Steuerzeichen nutzen, wenn Sie diese in die Steuerzeichentabelle im Eigenschaftendialog der Baugruppe eintragen und die Kommunikationsbaugruppe neu parametrieren.

Weitere Funktionen

Zusätzlich zur Ausgabe von Meldetexten können Sie noch folgende Funktionen für die Druckerausgabe nutzen. Die Ausführung einer dieser Funktionen geben Sie ebenfalls im Formatstring an.

- Seitennummer stellen (Formatstring = %P)
- neue Seite beginnen (Formatstring = \F)
- mit/ohne Zeilenumbruch drucken (\x am Ende des Formatstrings)

Beachten Sie, dass standardmäßig nach jeder Ausgabe ein Zeilenvorschub durchgeführt wird.

Beispiele

Beispiel 1: Um "17.30 " Uhr wurde der Füllstand "200" l erreicht!

Formatstring = Um %Z Uhr wurde der Füllstand %i l erreicht!
Variable 1 = Uhrzeit
Variable 2 = Stand

Beispiel 2: Der Kammerdruck "fällt"

Formatstring = %N %S
Variable 1 = 17 (Meldetext Nr. 17: Der Kammerdruck ...)
Variable 2 = Verweis auf String (Stringvariable: ... fällt)

Beispiel 3: (Seitennummer auf 10 stellen)

Formatstring = %P
Variable 1 = 10 (Seitennummer: 10)

Druckerausgabe

Zur Ausgabe von n Byte Nutzdaten auf einen Drucker sind der Formatstring und die Variablen des Meldetextes beim Aufruf der Anweisung als Parameter anzugeben.

Während der Ausgabe findet eine Druckaufbereitung der Daten statt. Die Druckaufbereitung erfolgt gemäß der Parametrierung im Eigenschaftendialog der Baugruppe (Seitenlayout, Zeichensatz, Steuerzeichen, etc.).

Ein Empfangen von Zeichen findet bei der Druckerausgabe nicht statt. Ausgenommen davon sind Steuerzeichen zur Flusskontrolle, wenn diese entsprechend parametriert wurden. Es werden keine eventuell eintreffenden Zeichen übernommen.

Meldetext ausgeben

Im folgenden Bild finden Sie die Abläufe bei der Druckerausgabe.

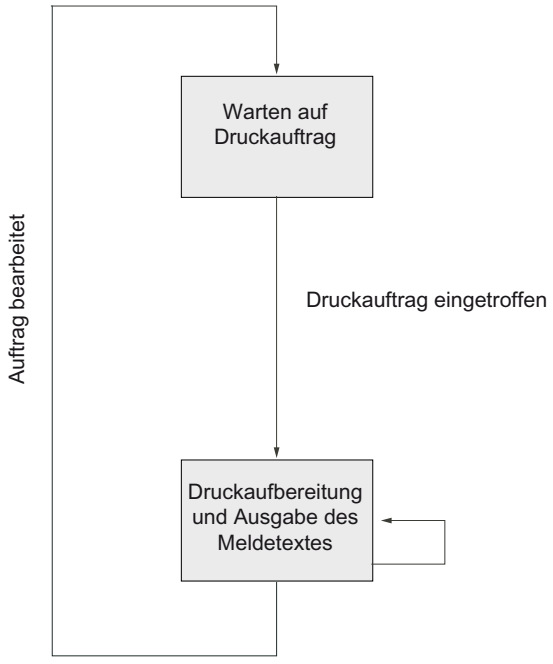

Bild 1-63 Ablaufschema bei der Druckerausgabe

Softwarehandshake/Hardwarehandshake

Handshakeverfahren steuern den Datenfluss zwischen zwei Kommunikationspartnern. Durch die Verwendung von Handshakeverfahren wird vermieden, dass bei unterschiedlich schnell arbeitenden Geräten Daten bei der Übertragung verloren gehen.

Auch bei der Druckerausgabe können Sie die Ausgabedaten mit Datenflusskontrolle senden. Grundsätzlich lassen sich die folgenden zwei Verfahren unterscheiden:

- Softwarehandshake (z. B. XON/XOFF)
- Hardwarehandshake (z. B. RTS/CTS)

Die Realisierung der Datenflusskontrolle auf der Kommunikationsbaugruppe bei Druckerausgabe wird folgendermaßen durchgeführt:

- Sobald die Kommunikationsbaugruppe durch Parametrierung in die Betriebsart mit Flusskontrolle gebracht wurde, sendet er das Zeichen XON bzw. setzt die Leitung RTS auf ON.
- Empfängt die Kommunikationsbaugruppe das Zeichen XOFF bzw. wird das Steuersignal CTS auf OFF gesetzt, unterbricht die Kommunikationsbaugruppe die Ausgabe von Zeichen. Wird nach einer bestimmten parametrierbaren Zeit kein XON empfangen bzw. CTS nicht auf ON gesetzt, wird die Druckerausgabe abgebrochen und eine entsprechende Fehlermeldung (0708H) am STATUS-Ausgang der Anweisung generiert.

Hinweis

Wenn Sie mit Flusskontrolle RTS/CTS parametrieren, müssen Sie unbedingt eine Vollverdrahtung der verwendeten Schnittstellensignale in der Steckerverbindung vornehmen.

BUSY-Signal

Die Kommunikationsbaugruppe wertet das Steuersignal "BUSY" des Druckers aus. Der Drucker meldet dem CP 34x1 seine Empfangsbereitschaft,

- bei CP 34x-20mA-TTY: mit Strom auf RxD-Leitung.
- bei CP 34x–RS232C und CP 34x-RS422/485: mit Signal CTS = "ON".

Hinweis

Wenn Sie mit Flusskontrolle RTS/CTS parametrieren, müssen Sie am Drucker die Polarität des BUSY-Signals wie folgt einstellen:

- BUSY-Signal: CTS = "OFF"

Beachten Sie, dass manche Drucker das BUSY-Signal mit Hilfe des DTR-Signals anzeigen. In diesem Fall müssen Sie im Verbindungskabel zur Kommunikationsbaugruppe eine entsprechende Verdrahtung vornehmen.

Endeerkennung eines Telegramms

Endekriterien

Bei der Datenübertragung mit dem ASCII–Treiber können Sie zwischen drei verschiedenen Endekriterien beim Datenempfang wählen:

- **Ablauf der Zeichenverzugszeit**
- **Empfang der/des Endezeichen(s)**
- **Empfang einer festen Zeichenanzahl**

Protokollparameter

In der folgenden Tabelle finden Sie die Beschreibung der Protokollparameter.

Tabelle 1-110 Protokollparameter (ASCII–Treiber)

Parameter	Beschreibung	Wertebereich	Defaultwert
Endeerkennung eines Empfangs–telegramms	Festlegung, welches Kriterium das Ende von Telegrammen bestimmt.	• Nach Ablauf der Zeichenverzugszeit • Nach Empfang der/des Endezeichen(s) • Nach Empfang einer festen Zeichenanzahl	Nach Ablauf der Zeichenverzugszeit
Endezeichen 1 [1]	Code der ersten Endekennnung.	• bei 7 Datenbits: 0 bis 7FH (Hex) [2] • bei 8 Datenbits: 0 bis FFH (Hex) [2]	3 (03H = ETX)
Endezeichen 2 [1]	Code der zweiten Endekennung, sofern angewählt.	• bei 7 Datenbits: 0 bis 7FH (Hex) [2] • bei 8 Datenbits: 0 bis FFH (Hex) [2]	0

[1] Nur bei Endekriterium Endezeichen einstellbar.
[2] Je nachdem, ob Sie für den Zeichenrahmen 7 oder 8 Datenbits parametrieren.

Weitere Informationen

Eine ausführliche Beschreibung des Datenempfangs mit dem ASCII-Treiber finden Sie in Kapiel Daten empfangen mit ASCII-Treiber (Seite 1684).

Datenflusskontrolle und Begleitsignale

Einleitung

Die Kommunikationsbaugruppe ermöglicht Ihnen eine Punkt-zu-Punkt-Kopplung mit SIMATIC-Baugruppen und mit Fremdprodukten.

Unterstützte Funktionen der Schnittstellen

In Abhängigkeit von der eingesetzten Schnittstelle können unterschiedliche Funktionen der Treiber genutzt werden:

Tabelle 1-111 Funktionen der Kommunikationsbaugruppe in Abhängigkeit von der eingesetzten Schnittstelle

	RS232C	20mA-TTY	X27 (RS422/485)	
			RS422*	RS485*
Funktion CP 441				
Prozedur 3964(R)	●	●	●	-
Rechnerkopplung RK512	●	●	●	-
ASCII-Treiber:	●	●	●	●
• Bedienung der RS232C-Begleitsignale	●	-	-	-
• Steuern/Lesen der RS232C- Begleitsignale mit Anweisungen	●	-	-	-
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-
Drucker-Treiber:	●	●	●	●
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-
Funktion CP 440				
Prozedur 3964(R)			●	-
ASCII-Treiber:			●	●
• Datenflusskontrolle mit XON/XOFF			●	-
Funktion CP 341				
Prozedur 3964(R)	●	●	●	-
Rechnerkopplung RK512	●	●	●	-
ASCII-Treiber:	●	●	●	●
• Bedienung der RS232C-Begleitsignale	●	-	-	-
• Steuern/Lesen der RS232C- Begleitsignale mit Anweisungen	●	-	-	-
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-
Drucker-Treiber:	●	●	●	●
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-
Funktion CP 340				
Prozedur 3964(R)	●	●	●	-
ASCII-Treiber:	●	●	●	●
• Bedienung der RS232C-Begleitsignale	●	-	-	-
• Steuern/Lesen der RS232C- Begleitsignale mit Anweisungen	●	-	-	-
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-

	RS232C	20mA-TTY	X27 (RS422/485)	
			RS422*	RS485*
Drucker-Treiber:	●	●	●	●
• Datenflusskontrolle mit RTS/CTS	●	-	-	-
• Datenflusskontrolle mit XON/XOFF	●	●	●	-

Funktion ET 200S 1SI				
Prozedur 3964(R)	●		●	-
ASCII-Treiber:	●		●	●
• Bedienung der RS232C-Begleitsignale	●		-	-
• Steuern/Lesen der RS232C- Begleitsignale mit Anweisungen	●		-	-
• Datenflusskontrolle mit RTS/CTS	●		-	-
• Datenflusskontrolle mit XON/XOFF	●		●	-
* Die Unterscheidung zwischen RS422 und RS485 erfolgt durch Parametrierung.				

Einsatzmöglichkeiten der Kommunikationsbaugruppen

Abhängig von der eingesetzten Kommunikationsbaugruppe können Sie eine Punkt-zu-Punkt-Kopplung mit unterschiedlichen Siemens-Baugruppen und mit Fremdprodukten aufbauen:

- SIMATIC S5 über 3964(R)-Treiber oder RK512 mit entsprechender Schnittstellenbaugruppe auf S5-Seite
- Siemens BDE-Terminals ES 2-Familie über 3964(R)-Treiber
- MOBY I (ASM 420/421, SIM), MOBY L (ASM 520) und Erfassungsstation ES 030K über 3964R-Treiber
- PCs über Prozedur 3964(R) (hierfür existieren Entwicklungstools zur Programmierung auf PC: PRODAVE DOS 64R (6ES5 897-2UD11) für MS-DOS, PRODAVE WIN 64R (6ES5 897-2VD01) für Windows oder dem ASCII-Treiber)
- Barcodeleser über 3964(R)-Treiber oder ASCII-Treiber
- SPS von anderen Herstellern über 3964(R)-, ASCII-Treiber oder RK512
- weitere Geräte mit einfachen Protokollstrukturen über entsprechende Protokollanpassung mit dem ASCII-Treiber
- weitere Geräte, die ebenfalls über 3964(R)-Treiber oder RK512 verfügen
- Drucker (HP-Deskjet, HP-Laserjet, Postscript, Epson, IBM)

Betriebsarten der Schnittstelle

Mögliche Betriebsarten der Schnittstelle

In der folgenden Tabelle sind die Betriebsarten der Schnittstelle, Kommunikationsbaugruppen- und protokollübergreifend zusammengestellt.

Betriebsart	Beschreibung
Halbduplex (RS485) Zweidraht-Betrieb	Betriebsart für Punkt-zu-Punkt-Kopplung oder Mehrpunkt-Kopplung (Multipoint) im Zweidraht-Betrieb. Die Kommunikationsbaugruppe kann sowohl Master als auch Slave sein. Bei RS485-Betrieb erfolgt die Datenübertragung über zwei Leitungen (Zweidraht-Betrieb). Die zwei Leitungen stehen abwechselnd für die Senderichtung und die Empfangsrichtung zur Verfügung. Es kann entweder gesendet oder empfangen werden (Halbduplex). Nach einem Sendevorgang wird sofort auf Empfang umgeschaltet (Umschaltzeit maximal 1 ms).
Vollduplex (RS422) Vierdraht-Betrieb	Bei RS422-Betrieb erfolgt die Datenübertragung über vier Leitungen (Vierdraht-Betrieb). Es stehen jeweils zwei Leitungen für die Senderichtung und die Empfangsrichtung zur Verfügung. Es kann gleichzeitig gesendet und empfangen werden (Vollduplex). Die Daten werden zwischen einem oder mehreren Kommunikationspartnern gleichzeitig ausgetauscht, d.h. es kann zu einem Zeitpunkt sowohl gesendet als auch empfangen werden. Jeder Kommunikationspartner muss simultan eine Sende- und Empfangseinrichtung betreiben können.
Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung) [1)]	Die Kommunikationsbaugruppe kann in im RS422-Betrieb in folgenden Topologien eingesetzt werden: • Kopplung zwischen zwei Teilnehmer: Punkt-zu-Punkt-Kopplung • Kopplung zwischen vielen Teilnehmer: Mehrpunkt-Kopplung Die Kommunikationsbaugruppe kann sowohl als Master als auch als Slave eingesetzt werden.

Betriebsart	Beschreibung
Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) [1]	Die Kommunikationsbaugruppe kann im RS422-Betrieb in folgenden Topologien eingesetzt werden: • Kopplung zwischen zwei Teilnehmern: Punkt-zu-Punkt-Kopplung • Kopplung zwischen vielen Teilnehmer: Mehrpunkt-Kopplung Die Kommunikationsbaugruppe kann sowohl als Mehrpunkt Master als auch als Mehrpunkt Slave eingesetzt werden. Bei einer Master/Slave-Topologie im RS422-Betrieb gilt: • Der Sender des Masters wird mit den Empfängern aller Slaves zusammengeschaltet. • Die Sender der Slaves werden mit dem Empfänger des Masters zusammengeschaltet. • Die Vorbelegung erfolgt beim Empfänger des Masters und bei dem Empfänger eines Slaves. Alle anderen Slaves arbeiten ohne Vorbelegung.
Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) [1]	Die Kommunikationsbaugruppe kann im RS422-Betrieb in folgenden Topologien eingesetzt werden: • Kopplung zwischen zwei Teilnehmern: Punkt-zu-Punkt-Kopplung • Kopplung zwischen vielen Teilnehmer: Mehrpunkt-Kopplung Die Kommunikationsbaugruppe kann sowohl als Mehrpunkt Master als auch als Mehrpunkt Slave eingesetzt werden. Bei einer Master/Slave-Topologie im RS422-Betrieb gilt: • Der Sender des Masters wird mit den Empfängern aller Slaves zusammengeschaltet. • Die Sender der Slaves werden mit dem Empfänger des Masters zusammengeschaltet. • Die Vorbelegung erfolgt beim Empfänger des Masters und bei dem Empfänger eines Slaves. Alle anderen Slaves arbeiten ohne Vorbelegung.

[1] nur CP 440

Vorbelegung der Empfangsleitung

X27 (RS422/485)-Schnittstelle

In der folgenden Tabelle finden Sie die Beschreibung der Parameter für das X27 (RS422/485)-Schnittstellenmodul. RS485-Betrieb ist mit den Prozeduren 3964(R) und RK512 und dem Drucker nicht möglich.

Tabelle 1-112 X27 (RS422/485)-Schnittstellenmodul

Parameter	Beschreibung	Wertebereich	Defaultwert
Vorbelegung der Empfangsleitung	**keine**: Einstellung nur sinnvoll für busfähige Sondertreiber.	keine	Signal R(A) 5V, Signal R(B) 0V (Break-Erkennung) (2)
	Signal R(A) 5V, Signal R(B) 0V (Break-Erkennung): Bei dieser Vorbelegung ist bei "Vollduplex (RS 422) Vierdraht-Betrieb" Break-Erkennung möglich.	Signal R(A) 5V, Signal R(B) 0V (Break-Erkennung) (1)	
	Signal R(A) 0V, Signal R(B) 5V: Diese Vorbelegung entspricht dem Ruhezustand (kein Sender aktiv) bei "Halbduplex (RS 485) Zweidraht-Betrieb". Bei dieser Vorbelegung ist keine Break-Erkennung möglich.	Signal R(A) 0V, Signal R(B) 5V	

(1) Nur bei "Vollduplex (RS422) Vierdraht-Betrieb"

(2) Nur bei "Vollduplex (RS422) Vierdraht-Betrieb"; bei "Halbduplex (RS485) Zweidraht-Betrieb" ist die Default-Einstellung Signal R(A) 0V, Signal R(B) 5V

Vorbelegung der Empfangsleitung

Das Bild zeigt die Beschaltung des Empfängers an der X27 (RS422/485)-Schnittstelle:

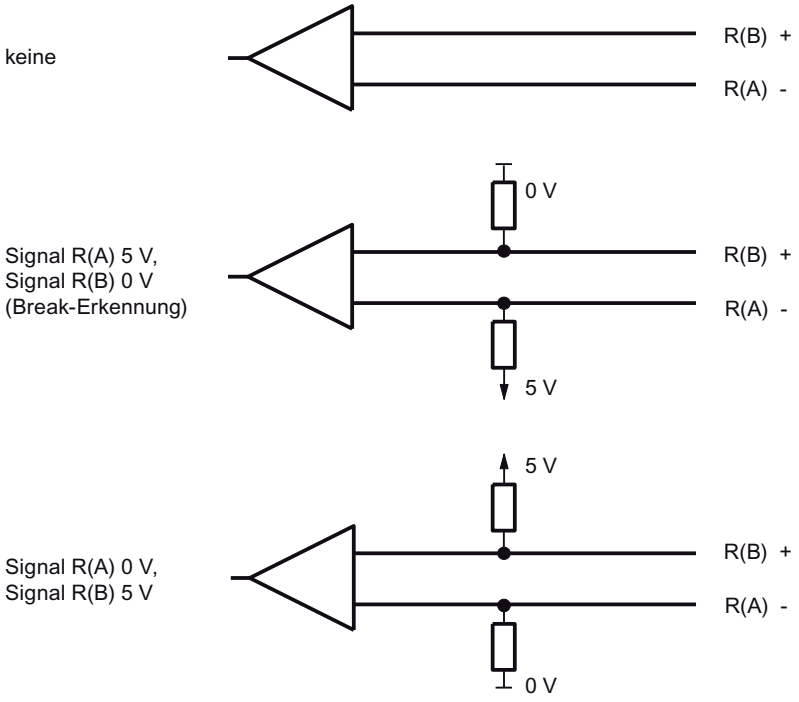

Bild 1-64 Beschaltung des Empfängers an der X27 (RS422/485)–Schnittstelle

CP 440 in einer Mehrpunkt-Topologie mit den Protokollen RS422/485

Einsatzmöglichkeiten

Der CP 440 kann in der Betriebsart RS422 bzw. RS485 in verschiedenen Topologien eingesetzt werden.

Unterschieden wird zwischen Kopplungen mit

- zwei Teilnehmern (**Punkt-zu-Punkt**) und
- vielen Teilnehmern (**Mehrpunkt/Multipoint**).

Dabei kann der CP 440 eingesetzt werden als

- **Master** oder
- **Slave**.

RS422- bzw. RS485-Betrieb bei Punkt-zu-Punkt Verbindungen

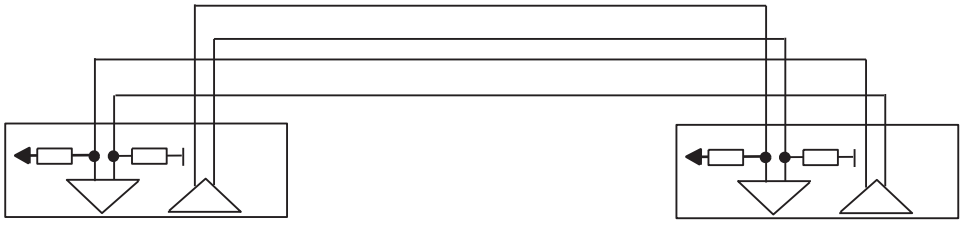

Bild 1-65 RS422 Punkt-zu-Punkt

Bild 1-66 RS485 Punkt-zu-Punkt

Bei einer **Master/Slave Topologie** muss im Anwenderprogramm ein entsprechendes Telegramm ausprogrammiert werden. Beispiel: Der Master sendet an alle Slaves ein Telegramm mit einer Adressinformation. Die Slaves hören alle mit und vergleichen die Adresse mit ihrer eigenen. Bei Gleichheit sendet der angesprochene Slave seine Antwort.

Die Sender aller Slaves müssen in der Lage sein, sich hochohmig zu schalten.

RS422-Betrieb bei Mehrpunkt-Verbindungen

Bei einer Master/Slave Topologie im RS422-Betrieb

- wird der Sender des Masters mit den Empfängern aller Slaves zusammengeschaltet,
- werden die Sender der Slaves mit dem Empfänger des Masters zusammengeschaltet,
- erfolgt die Vorbelegung nur beim Empfänger des Masters und bei dem Empfänger eines Slaves. Alle anderen Slaves arbeiten ohne Vorbelegung.

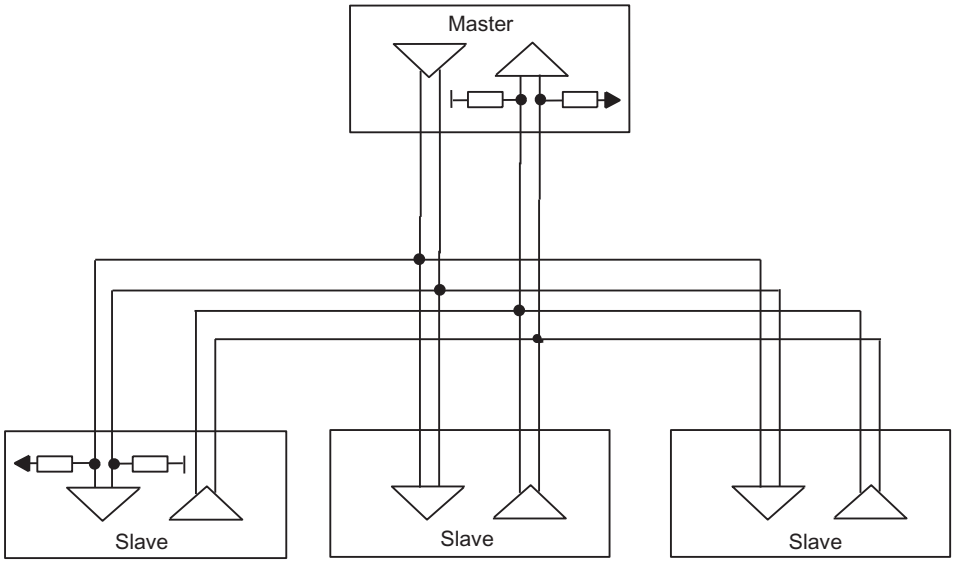

Bild 1-67 RS422 Mehrpunkt (Multipoint)

RS485-Betrieb bei Mehrpunkt-Verbindungen

Bei einer Topologie im RS485-Betrieb

- wird das Leitungspaar für die Sende-/Empfangsleitung aller Teilnehmer zusammengeschaltet,
- erfolgt die Vorbelegung nur beim Empfänger eines Teilnehmers. Alle anderen Baugruppen arbeiten ohne Vorbelegung.

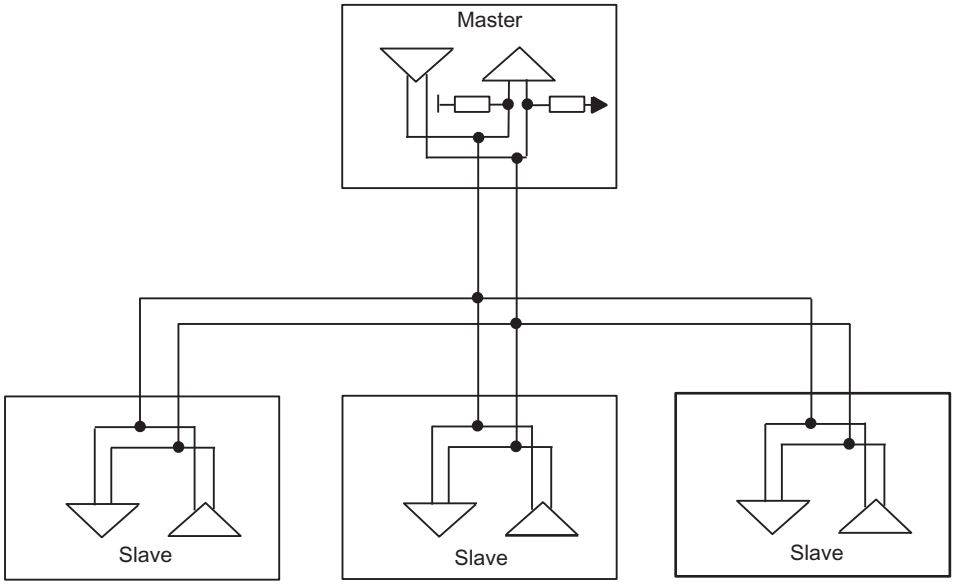

Bild 1-68 RS485 Mehrpunkt (Multipoint)

Die für die unterschiedlichen Topologien notwendigen Einstellungen erfolgen im Eigenschaftendialog der Baugruppe in der Maske "Schnittstelle".

Hinweis

Wenn Sie den ASCII-Treiber im RS422-Mehrpunktbetrieb oder im RS485-Betrieb bearbeiten, müssen Sie im Anwenderprogramm dafür sorgen, dass immer nur ein Teilnehmer sendet. Wenn gleichzeitig gesendet wird, wird das Telegramm verfälscht.

Empfangspuffer

Empfangspuffer auf der Baugruppe

In der folgenden Tabelle finden Sie die Beschreibung der Parameter für den Baugruppen-Empfangspuffer.

Tabelle 1-113 Empfangspuffer der Baugruppe - Prozedur 3964(R) / RK512 / ASCII-Treiber

Parameter	Beschreibung	Wertebereich	Defaultwert
Baugruppen-Empfangspuffer im Anlauf löschen	Der Baugruppen-Empfangspuffer wird im Anlauf der CPU (STOP-RUN-Übergang) nicht gelöscht.	nein (nicht veränderbar)	nein
CPU-Empfangsfach verwenden (nur CP 441)	Sie können angeben, ob auf der CPU ein Empfangsfach eingerichtet werden soll. Sie müssen ein Empfangsfach einrichten, wenn Sie im Anwenderprogramm der CPU für die Baugruppe keine Anweisung BRCV programmiert haben. Wenn Sie einen BRCV programmiert haben, dann müssen Sie diesen Parameter deaktivieren, da sonst Daten nicht vom BRCV bearbeitet werden, sondern im hier definierten Empfangsfach abgelegt werden.	• ja • nein	nein
DB-Nummer (1)	Nummer des Datenbausteins für das Empfangsfach auf der CPU	1 bis 65535 (je nach CPU)	1
(1) Nur bei "CPU-Empfangsfach verwenden" = "ja". (2) Nur wenn "Gepufferte Empfangstelegramme" = "1".			

Tabelle 1-114 Empfangspuffer der Baugruppe - zusätzlich beim ASCII-Treiber

Parameter	Beschreibung	Wertebereich	Defaultwert
Gepufferte Empfangs-telegramme	Sie können die Anzahl der Empfangstelegramme angeben, die im Empfangspuffer der Kommunikationsbaugruppe gepuffert werden sollen. Wenn Sie hier "1" parametrieren und den nachfolgenden Parameter "Überschreiben verhindern" deaktivieren und die Empfangsdaten im Anwenderprogrammm zyklisch auslesen, wird immer ein aktuelles Telegramm zur CPU übertragen.	1 bis 250	250
Dynamische Telegramme ET 200S 1SI	Zum Empfangen von Meldungen können Sie angeben, ob nur eine Meldung gepuffert werden soll, oder ob die Meldungen dynamisch gepuffert werden sollen. Durch Aktivieren von dynamischen Telegrammen kann die Baugruppe mehrere Meldungen mit unterschiedlichen Längen puffern. Bei dem Puffer handelt es sich um einen Ringpuffer. Ist der Puffer voll, wird die älteste Meldung überschrieben, es sei denn, der Parameter "Überschreiben des Puffers verhindern" ist aktiviert. In diesem Fall wird die neueste Meldung verworfen. In beiden Fällen zeigt ein Diagnosealarm an, dass Daten verloren gegangen sind.	• Aktiviert • Gesperrt	Aktiviert
Überschreiben verhindern	Wenn der Parameter "Gepufferte Empfangstelegramme" auf "1" eingestellt ist, können Sie diesen Parameter deaktivieren. Sie lassen damit ein Überschreiben des gepufferten Empfangstelegramms zu.	• ja • nein (2)	ja
(1) Nur bei "CPU-Empfangsfach verwenden" = "ja". (2) Nur wenn "Gepufferte Empfangstelegramme" = "1".			

Modbus-Kommunikation

Überblick über Modbus-Kommunikation

Einordnen in die Systemumgebung

Die nachfolgende Modbus Beschreibung bezieht sich auf den Einsatz des entsprechenden nachladbaren Treibers auf den Kommunikationsprozessoren CP 341 (S7 300) und CP 441-2 (S7 400), sowie auf den Betrieb des ET 200S 1SI Moduls in der Modbus/USS-Variante.

Funktion der Kopplung

Mit den entsprechend ausgestatteten Kommunikationsbaugruppen und den zugehörigen Anweisungen wird eine Kommunikationsverbindung zwischen einem remoten Modbus-Steuerungssystem (z. B. Modicon-Steuerungen oder Honeywell TDC3000) und einer SIMATIC S7 ermöglicht.

Zur Übertragung wird das GOULD - MODBUS -Protokoll im RTU-Format verwendet.

Bei der Kommunikation zwischen einer als Modbus-Slave betriebenen Kommunikationsbaugruppe und einem Master-System können die Funktionscodes 01, 02, 03, 04, 05, 06, 08, 15 und 16 verwendet werden.

Wird eine SIMATIC S7 Kommunikationsbaugruppe als Modbus-Master betrieben, stehen zusätzlich die Funktionscodes 07, 11 und 12 zur Verfügung.

Modbus-Protokoll

Das Modbus-Protokoll ist ein Kommunikationsprotokoll, das auf einer Master/Slave- bzw. Client/Server-Architektur basiert.

Zum Einsatz kommt eine codetransparente, asynchrone Halbduplexprozedur. Die Datenübertragung erfolgt ohne Handshake.

SIMATIC S7 als Modbus-Slave

Der Master hat bei der Übertragung die Initiative, der CP/die S7-CPU arbeitet als Slave.

Ein Telegrammverkehr von Slave zu Slave ist nicht möglich.

SIMATIC S7 als Modbus-Master

Die Kommunikationsbaugruppe initiiert als Master die Übertragung und wartet nach Ausgabe des Anforderungstelegramms die parametrierte Antwortüberwachungszeit auf ein Antworttelegramm vom Slave.

Telegrammaufbau

Der Datenverkehr "Master-Slave" bzw. "Slave-Master" beginnt mit der **Slave-Adresse**, gefolgt vom **Funktionscode**. Dann werden die Daten übertragen. Der Aufbau des Datenfeldes ist abhängig vom verwendeten Funktionscode. Am Ende des Telegramms wird der CRC-Check übertragen.

ADDRESS	FUNCTION	DATA	CRC-CHECK
Byte	Byte	n Byte	2 Byte

ADDRESS	MODBUS-Slave-Adresse
FUNCTION	MODBUS-Funktions-Code
DATA	Telegrammdaten: Byte_Count, Coil_Number, Data
CRC-CHECK	Telegramm-Prüfsumme

Slave-Adresse

Die Slave-Adresse kann im Bereich von 1 bis 255 liegen. Mit der Adresse wird ein definierter Slave am Bus angesprochen.

Broadcast Message

Mit der Slave-Adresse Null spricht der Master alle Slaves am Bus an.

Broadcast-Messages sind nur in Verbindung mit den schreibenden **Funktionscodes 05, 06, 15 bzw. 16** zulässig.

Bei einer Broadcast-Message wird vom Slave kein Antworttelegramm gesendet.

Funktionscode

Der Funktionscode definiert die Bedeutung des Telegramms. Durch den Funktionscode ist auch der Telegrammaufbau festgelegt. Folgende Funktionscodes werden von der Kommunikationsbaugruppe unterstützt:

Funtionscode	Funktion gemäß MODBUS-Spezifikation
01	Read Coil Status
02	Read Input Status
03	Read Holding Registers
04	Read Input Registers
05	Force Single Coil
06	Preset Single Register
07	Read Exception Status (nur Master)
08	Loop Back Test
11	Fetch Communications Event Counter (nur Master)
12	Fetch Communications Event Log (nur Master)
15	Force Multiple Coils
16	Preset Multiple Registers

Datenfeld DATA

Im Datenfeld DATA werden die funktionscode-spezifischen Daten übertragen:

- Bytecount, Coil_Startaddress, Register_Startaddress; Number_of_Coils, Number_of_Registers, ...
  Siehe hierzu Kapitel "Funktionscodes (Seite 1744)".

CRC-Check

Den Telegrammabschluss bildet die zwei Byte lange CRC-16-Prüfsumme, gebildet nach dem Polynom $x^{16} + x^{15} + x^2 + 1$.

Zuerst wird das Low-Byte, dann das High-Byte übertragen.

Telegrammende

Der ladbare Treiber erkennt das Telegrammende, wenn während der Zeit, die dreieinhalb Zeichen zur Übermittlung benötigen (3,5-fache Zeichenverzugszeit), keine Übertragung stattfindet (siehe MODBUS Protocol Reference Guide).

Dieser Telegrammende-TIME_OUT ist somit abhängig von der Datenübertragungsrate.

Datenübertragungsrate	TIME_OUT
115200 bit/s	0,3 ms
76800 bit/s	0,5 ms
57600 bit/s	0,7 ms
38400 bit/s	1 ms
19200 bit/s	2 ms
9600 bit/s	4 ms
4800 bit/s	8 ms
2400 bit/s	16 ms
1200 bit/s	32 ms
600 bit/s	64 ms
300 bit/s	128 ms

Im "Normal-Betrieb" wird nach Eintreffen des Telegrammende-TIME_OUTs das vom Koppelpartner empfangene Modbus-Telegramm ausgewertet und formal geprüft.

Bei "Störungsunterdrückung" wird das Telegrammende durch ein formal richtiges Empfangstelegramm mit korrektem CRC-Code erkannt.

Exception Responses

Wenn der Slave einen Fehler im Anforderungstelegramm des Masters erkennt, z. B. Registeradresse nicht erlaubt, setzt er das höchstwertige Bit im Funktionscode des Antworttelegramms.

Nachfolgend wird ein Byte Fehlercode, Exception Code, übertragen, das die Fehlerursache beschreibt.

Sinn und Bedeutung der genannten Parameter entnehmen Sie der Beschreibung "GOULD MODICON Modbus Protocol".

Exception Code Telegramm

Das Fehlercode-Antworttelegramm vom Slave hat folgenden Aufbau:

- z. B. Slave-Adresse 5, Funktionscode 5, Exception Code 2

Antworttelegramm vom Slave EXCEPTION_CODE_xx:

05H	Slave-Adresse
85H	Funktionscode
02H	Exception Code (1...7)
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Wenn der Treiber ein Fehlercode-Antworttelegramm empfängt, wird der laufende Auftrag mit Fehler beendet.

Zusätzlich wird in den Fehlermeldebereich eine dem empfangenen Fehlercode (Exception Code 1-7) entsprechende Fehlernummer eingetragen.

Es erfolgt kein Eintrag in den Ziel-Datenbaustein.

Folgende Fehlercodes sind gemäß MODBUS-Spezifikation definiert:

Fehlercode	Bedeutung nach MODBUS-Spezifikation	Ursache - Kurzbeschreibung *
1	Illegal function	Unzulässiger Funktionscode
2	Illegal data address	Unzulässige Daten-Adresse beim Slave
3	Illegal data value	Unzulässiger Daten-Wert beim Slave
4	Failure in associated device	Interner Fehler beim Slave
5	Acknowledge	Funktion wird ausgeführt
6	Busy, Rejected message	Slave ist nicht empfangsbereit
7	Negative acknowledgement	Funktion nicht ausführbar
* Detaillierte Ursache beim Slave untersuchen.		

RS232C-Begleitsignale

Auf der Kommunikationsbaugruppe sind bei Einsatz des RS232C-Schnittstellenmoduls folgende RS232C-Begleitsignale vorhanden:

DCD	(Eingang)	Data carrier detect	Datenträger erkannt
DTR	(Ausgang)	Data terminal ready	Kommunikationsbaugruppe betriebsbereit
DSR	(Eingang)	Data set ready	Kommunikationspartner betriebsbereit
RTS	(Ausgang)	Request to send	Kommunikationsbaugruppe sendebereit
CTS	(Eingang)	Clear to send	Kommunikationspartner kann Daten von der Kommunikationsbaugruppe empfangen (Antwort auf RTS = ON des CP)
RI	(Eingang)	Ring indicator	Rufzeichen

Nach dem Einschalten der Kommunikationsbaugruppe befinden sich die Ausgangssignale im Zustand OFF (inaktiv).

Die Bedienung der Steuersignale DTR/DSR und RTS/CTS können Sie im Eigenschaften Dialog der Baugruppe parametrieren oder über Anweisungen im Anwenderprogramm steuern.

RS232C-Begleitsignale bedienen

Die RS232C-Begleitsignale können bedient werden:

- Bei parametrierter automatischen Bedienung aller RS232C-Begleitsignale.
- Über die entsprechenden Anweisungen. Weitere Informationen hierzu finden Sie in Kapitel "Übersicht über die Anweisungen (Seite 1671)".

Hinweis

Bei der Parametrierung einer automatischen Bedienung der RS232C-Begleitsignale ist das Steuern von RTS und DTR über die entsprechende Anweisung (siehe Kapitel "Übersicht über die Anweisungen (Seite 1671)") nicht möglich!

Das Lesen aller RS232C-Begleitsignale über die entsprechende Anweisung ist hingegen immer möglich.

Die folgenden Abschnitte erläutern das prinzipielle Handling bei der Steuerung und Auswertung der RS232C-Begleitsignale.

Automatische Bedienung der Begleitsignale

Die Realisierung der automatischen Bedienung der RS232C-Begleitsignale auf der Kommunikationsbaugruppe wird folgendermaßen durchgeführt:

- Sobald die Kommunikationsbaugruppe durch Parametrierung in eine Betriebsart mit automatischer Bedienung der RS232C-Begleitsignale gebracht wurde, setzt er die Leitungen RTS auf OFF und DTR auf ON (Kommunikationsbaugruppe betriebsbereit).
- Das Senden und Empfangen von Telegrammen ist erst möglich, nachdem die Leitung DTR auf ON gesetzt wurde. Solange DTR auf OFF gesetzt bleibt, werden keine Daten über die RS232C-Schnittstelle empfangen. Ein Sendeauftrag wird mit einer entsprechenden Fehlermeldung abgebrochen.
- Wenn ein Sendeauftrag ansteht, wird RTS auf ON gesetzt und die parametrierte Datenausgabewartezeit gestartet. Nach Ablauf der Datenausgabezeit und CTS = ON werden die Daten über die RS232C-Schnittstelle gesendet.
- Wenn beim Senden innerhalb der Datenausgabewartezeit die Leitung CTS nicht auf ON gesetzt wird oder innerhalb des Sendevorgangs ein Wechsel von CTS auf OFF erfolgt, wird der Sendeauftrag abgebrochen und eine entsprechende Fehlermeldung generiert.
- Nach dem Senden der Daten wird nach Ablauf der parametrierten RTS-Wegnahmezeit die Leitung RTS auf OFF gesetzt. Es wird nicht auf den Wechsel von CTS auf OFF gewartet.
- Ein Empfang von Daten über die RS232C-Schnittstelle ist möglich, sobald die Leitung DSR auf ON gesetzt ist. Droht der Empfangspuffer der Kommunikationsbaugruppe überzulaufen, erfolgt keine Reaktion der Kommunikationsbaugruppe.
- Bei einem Wechsel von DSR = ON auf OFF wird sowohl ein laufender Sendeauftrag als auch das Empfangen von Daten mit einer Fehlermeldung abgebrochen. Im Diagnosepuffer der Kommunikationsbaugruppe wird die Meldung "DSR = OFF" eingetragen (automatische Bedienung der V24-Signale).

Hinweis

Bei der Parametrierung einer automatischen Bedienung der RS232C-Begleitsignale ist das Steuern von RTS und DTR über die entsprechende Anweisung nicht möglich! Weitere Informationen hierzu finden Sie in Kapitel "Übersicht über die Anweisungen (Seite 1671)".

Hinweis

Stellen Sie die "RTS-Wegnahmezeit" im Eigenschaften Dialog der Baugruppe so ein, dass der CP die letzten Zeichen des Telegramms vollständig empfangen kann, bevor RTS und damit der Sendewunsch weggenommen wird. Dimensionieren Sie die "Datenausgabewartezeit" so, dass der Kommunikationspartner in Empfangsbereitschaft gehen kann, bevor die Zeit abläuft.

Zeitdiagramm

Das folgende Bild zeigt den zeitlichen Ablauf eines Sendeauftrags:

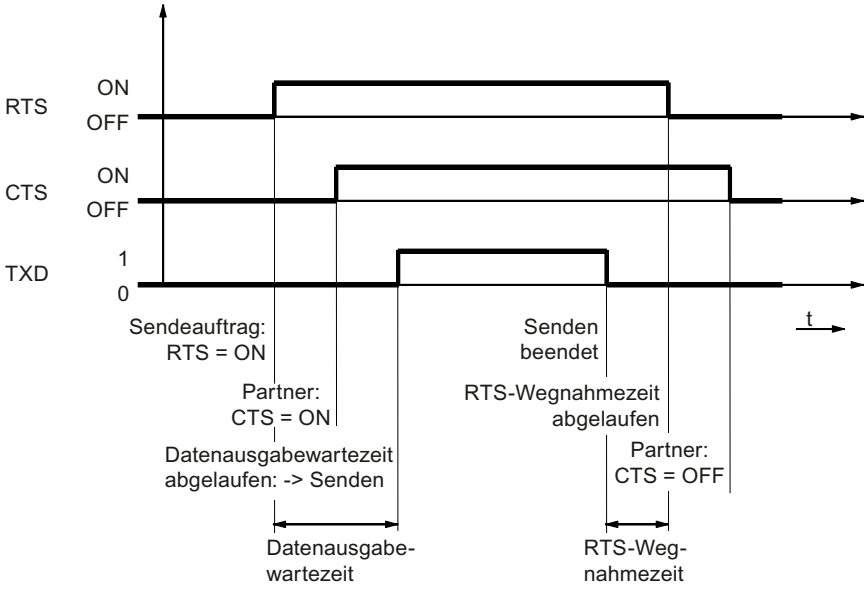

Bild 1-69 Zeitdiagramm bei automatischer Bedienung der RS232C Begleitsignale

Modbus Master

CP 341

Datenübergabe von der CPU an die Kommunikationsbaugruppe mit P_SND_RK (CP 341)

Aktivierung

Die Ausführung eines MODBUS-Funktionscodes wird mit der Anweisung **P_SND_RK** mit einer **Flanke** am Eingang **REQ** aktiviert.

Am Parameter SF geben Sie 'S' für SEND an.

An LADDR wird die logische Baugruppenadresse angegeben.

Als Bereichstyp der Partner-CPU muss 'X' für erweiterte Datenbausteine angegeben werden. Für die übrigen Parameter der Partner-CPU (R_...) müssen keine Werte angegeben werden.

Dadurch werden die zur Ausführung des Funktionscodes nötigen Parameter an den Treiber übergeben.

Datenquelle

Bei Aktivierung des P_SND_RK wird der mit den Parametern **DB_NO** und **DBB_NO** angegebene **Quell-Datenbereich** mit der Länge **LEN** an die Kommunikationsbaugruppe übertragen.

Längenangabe

Die Länge **LEN** ist abhängig vom verwendeten Funktionscode.

Funktionscode	Länge LEN in Byte
01	6
02	6
03	6
04	6
05	6
06	6
07	2
08	6
11	2
12	2
15	>6
16	>6

Wenn bei den einzelnen Funktionscodes die aufgelisteten Datenmengen nicht übergeben werden, wird der Auftrag nicht ausgeführt und der P_SND_RK mit einer Flanke am Ausgang ERROR beendet.

SEND Quell-DB

Im Quell-Datenbereich müssen die zur Ausführung eines Funktionscodes benötigten Parameter als Nutzdaten eingetragen sein.

Der detaillierte Aufbau des jeweiligen P_SND_RK-Quell-DB ist bei der Beschreibung der einzelnen Funktionscodes im Kapitel "Funktionscodes (Seite 1744)" dargestellt.

Telegrammgenerierung

Die Anforderungs-Telegramme an den Slave werden anhand der übergebenen P_SND_RK-Quell-Daten aufgebaut und von der Kommunikationsbaugruppe gesendet.

Zunächst überprüft der Treiber, ob die am P_SND_RK angegebene Länge LEN mit der Länge für diesen Funktionscode korrespondiert.

Wenn nicht, wird der Auftrag nicht ausgeführt und mit einer Flanke am Ausgang ERROR des P_SND_RK beendet.

Bei Verwendung von anderen Funktionscodes als den aufgeführten wird der aktivierte Auftrag ebenfalls nicht ausgeführt und mit ERROR am P_SND_RK beendet.

Die im Anforderungs-Telegramm enthaltenen Elemente "Bytezähler" und "CRC-Check" werden von der Kommunikationsbaugruppe gebildet, ein Eintrag in den P_SND_RK-Quell-DB ist nicht erforderlich.

Auftragsabschluss bei schreibenden Funktionen

Bei schreibenden Funktionscodes wird der aktivierte P_SND_RK nach dem fehlerfreien Empfang des Antworttelegramms beendet. Dies wird dem SIMATIC-Anwenderprogramm mit einer Flanke am Ausgang **DONE** des P_SND_RK mitgeteilt.

Wenn beim Telegrammverkehr **Fehler** erkannt wurden oder der Slave mit einem **Fehlercode**-Antworttelegramm antwortet, wird dies durch eine Flanke am Ausgang **ERROR** gemeldet.

Auftragsabschluss bei lesenden Funktionen

Bei lesenden Funktionen wird der aktivierte P_SND_RK nach dem fehlerfreien Empfang des Antworttelegramms **und** vollständiger Übergabe der empfangenen Daten an die CPU beendet.

Dies wird dem SIMATIC-Anwenderprogramm mit einer Flanke am Ausgang **DONE** des P_SND_RK mitgeteilt.

Zu diesem Zeitpunkt sind die empfangenen Daten also bereits in der CPU vorhanden.

Wenn beim Telegrammverkehr **Fehler** erkannt wurden oder der Slave mit einem **Fehlercode**-Antworttelegramm antwortet, wird dies durch eine Flanke am Ausgang **ERROR** gemeldet.

In diesem Fall werden keine Empfangsdaten an die CPU übergeben.

STATUS-Eintrag bei Auftragsende

In den Fällen, in denen ein Auftrag mit **ERROR** am P_SND_RK beendet wird, erfolgt ein zusätzlicher Eintrag eines Fehlercodes im Parameter **STATUS**.

Anhand dieses Fehlercodes kann die genaue Fehlerursache ermittelt werden

Datenübergabe von der Kommunikationsbaugruppe an die CPU mit P_RCV_RK (CP 341)

Voraussetzung

Alle **lesenden** Funktionscodes benötigen einen P_RCV_RK.

Datenziel

Bei aktivierter Empfangsbereitschaft der Anweisung P_RCV_RK übernimmt dieser die empfangenen Daten von der Kommunikationsbaugruppe und trägt sie in das durch die Parameter **DB_N0** und **DBB_N0** spezifizierte Datenziel ein.

Anzeige des Daten-Empfangs

Der Empfang von Daten in der CPU wird dem Anwender durch eine Flanke am Ausgang **NDR** mitgeteilt.

Zu diesem Zeitpunkt steht im Parameter **LEN** die Länge des empfangenen Datenblocks.

Der Abschluss des gesamten Modbus-Auftrags kann am Ausgang DONE der Anweisung P_SND_RK erkannt werden.

Verhalten im Fehlerfall

Bei Empfangs- oder Übergabefehlern werden keine Daten an die CPU übergeben. Die Anweisung P_SND_RK wird in diesem Fall mit einer Flanke am Ausgang ERROR beendet.

P_RCV_RK Ziel-DB

Die bei einem lesenden Funktionscode empfangenen Nutzdaten werden in den P_RCV_RK-Zielbereich eingetragen.

Der detaillierte Aufbau des jeweiligen P_RCV_RK-Ziel-DB ist bei der Beschreibung der einzelnen Funktionscodes im Kapitel " Funktionscodes (Seite 1744) " dargestellt.

Die Länge der eingetragenen Daten wird am Parameter LEN des P_RCV_RK angezeigt.

CP 441-2

Datenübergabe von der CPU an die Kommunikationsbaugruppe mit BSEND (CP 441-2)

Kommunikationsverbindung

Der **Parameter ID** beschreibt die eindeutige Kommunikationsverbindung zu einem Kommunikationspartner. Hier muss die lokale ID aus der Verbindungsprojektierung angegeben werden

Bausteinbeziehung

Der **Parameter R_ID** beschreibt die eindeutige Bausteinbeziehung innerhalb einer Kommunikationsverbindung.

Bei diesem Treiber können für die R_ID am BSEND **beliebige** Werte **0..255** eingetragen werden.

Bei lesenden Aufträgen muss am zugehörigen BRCV die gleiche R_ID wie beim BSEND parametriert sein.

Aktivierung

Die Ausführung eines MODBUS-Funktionscodes wird mit einer Anweisung **BSEND** mit einer **Flanke** am Eingang **REQ** aktiviert.

Dadurch werden die zur Ausführung des Funktionscodes nötigen Parameter an den Treiber übergeben.

Datenquelle

Bei Aktivierung des BSEND wird der am Parameter **SD_1** angegebene **Quell-Datenbereich** mit der Länge **LEN** an die Kommunikationsbaugruppe übertragen.

Längenangabe

Die Länge **LEN** ist abhängig vom verwendeten Funktionscode.

Funktionscode	Länge LEN in Byte
01	6
02	6
03	6
04	6
05	6
06	6
07	2
08	6
11	2
12	2
15	>6
16	>6

Wenn bei den einzelnen Funktionscodes die aufgelisteten Datenmengen nicht übergeben werden, wird der Auftrag nicht ausgeführt und der BSEND mit einer Flanke am Ausgang ERROR beendet.

BSEND Quell-DB

Im Quell-Datenbereich müssen die zur Ausführung eines Funktionscodes benötigten Parameter als Nutzdaten eingetragen sein.

Der detaillierte Aufbau des jeweiligen BSEND-Quell-DB ist bei der Beschreibung der einzelnen Funktionscodes in Kapitel "Funktionscodes (Seite 1744)" dargestellt.

Telegrammgenerierung

Die Anforderungs-Telegramme an den Slave werden anhand der übergebenen BSEND-Quell-Daten aufgebaut und von der Kommunikationsbaugruppe gesendet.

Zunächst überprüft der Treiber, ob die am BSEND angegebene Länge LEN mit der Länge für diesen Funktionscode korrespondiert.

Wenn nicht, wird der Auftrag nicht ausgeführt und mit einer Flanke am Ausgang ERROR des BSEND beendet.

Bei Verwendung von anderen Funktionscodes als den aufgeführten wird der aktivierte Auftrag ebenfalls nicht ausgeführt und mit ERROR am BSEND beendet.

Die im Anforderungs-Telegramm enthaltenen Elemente "Bytezähler" und "CRC-Check" werden von der Kommunikationsbaugruppe gebildet, ein Eintrag in den BSEND-Quell-DB ist nicht erforderlich.

Auftragsabschluss bei schreibenden Funktionen

Bei schreibenden Funktionscodes wird der aktivierte BSEND nach dem fehlerfreien Empfang des Antworttelegramms beendet.

Dies wird dem SIMATIC-Anwenderprogramm mit einer Flanke am Ausgang **DONE** des BSEND mitgeteilt.

Wenn beim Telegrammverkehr **Fehler** erkannt wurden oder der Slave mit einem **Fehlercode**-Antworttelegramm antwortet, wird dies durch eine Flanke am Ausgang **ERROR** gemeldet.

Auftragsabschluss bei lesenden Funktionen

Bei lesenden Funktionen wird der aktivierte BSEND nach dem fehlerfreien Empfang des Antworttelegramms **und** vollständiger Übergabe der empfangenen Daten an die CPU beendet.

Dies wird dem SIMATIC-Anwenderprogramm mit einer Flanke am Ausgang **DONE** des BSEND mitgeteilt.

Zu diesem Zeitpunkt sind die empfangenen Daten also bereits in der CPU vorhanden.

Wurden beim Telegrammverkehr **Fehler** erkannt oder antwortete der Slave mit einem **Fehlercode**-Antworttelegramm, wird dies durch eine Flanke am Ausgang **ERROR** gemeldet.

In diesem Fall werden keine Empfangsdaten an die CPU übergeben.

Eintrag in den Fehlermeldebereich bei Auftragsende

In den Fällen, in denen ein Auftrag mit **ERROR** am BSEND beendet wird, erfolgt ein zusätzlicher Eintrag eines Fehlercodes in den Fehlermeldebereich.

Anhand dieses Fehlercodes kann die genaue Fehlerursache ermittelt werden.

Datenübergabe von der Kommunikationsbaugruppe an die CPU mit BRCV (CP 441-2)

Kommunikationsverbindung

Der **Parameter ID** beschreibt die eindeutige Kommunikationsverbindung zu einem Kommunikationspartner. Hier muss die lokale ID aus der Verbindungsprojektierung angegeben werden.

Bausteinbeziehung

Der **Parameter R_ID** beschreibt die eindeutige Bausteinbeziehung innerhalb einer Kommunikationsverbindung.

Alle **lesenden** Funktionscodes benötigen einen BRCV.
Die am **BRCV** parametrierte **R_ID** muss die **gleiche R_ID** wie beim zugehörigen BSEND aufweisen, mit der dieser Auftrag aktiviert wurde (beliebige Werte 0 ... 255).

So können im SIMATIC-Anwenderprogramm mehrere BSEND-/BRCV-Paare programmiert werden.

Die vom Modbus-Slave empfangenen Antwort-Telegramme werden dann in unterschiedliche Zielbereiche abgelegt, in Abhängigkeit von der bei diesem Auftrag verwendeten R_ID.

Datenziel

Bei aktivierter Emfangsbereitschaft der Anweisung BRCV übernimmt dieser die empfangenen Daten von der Kommunikationsbaugruppe und trägt sie in das am Parameter **RD_1** angegebene Datenziel ein. Das Datenziel ist somit variabel.

Anzeige des Daten-Empfangs

Der Empfang von Daten in der CPU wird dem Anwender durch eine Flanke am Ausgang **NDR** mitgeteilt.

Zu diesem Zeitpunkt steht im Parameter **LEN** die Länge des empfangenen Datenblocks.

Der Abschluss des gesamten Modbus-Auftrags kann am Ausgang DONE der Anweisung BSEND erkannt werden.

Verhalten im Fehlerfall

Bei Empfangs- oder Übergabefehlern werden keine Daten an die CPU übergeben. Der BSEND wird in diesem Fall mit einer Flanke am Ausgang ERROR beendet.

BRCV Ziel-DB

Die bei einem lesenden Funktionscode empfangenen Nutzdaten werden in den BRCV-Zielbereich eingetragen.

Der detaillierte Aufbau des jeweiligen BRCV-Ziel-DB ist bei der Beschreibung der einzelnen Funktionscodes im Kapitel "Funktionscodes (Seite 1744)" dargestellt.

Die Länge der eingetragenen Daten wird am Parameter LEN des BRCV angezeigt.

ET 200S 1SI

Datenübertragung beim Modbus-Master ET 200S 1SI

Einleitung

Die Datenübergabe zwischen Baugruppe und CPU erfolgt mittels der Anweisung S_SEND und S_RCV.

Anweisung S_SEND: Daten an einen Kommunikationspartner senden

Für die Ausführung eines Modbus-Master-Auftrags müssen die Anweisungen S_SEND und S_RCV aktiviert werden. Die Anweisung S_SEND wird mit einer Flanke am Eingang REQ aktiviert, wenn Daten an die Baugruppe ausgegeben werden sollen. Die Anweisung S_RCV wird mit EN_R=1 für den Empfang von Daten von der Baugruppe vorbereitet. Ein S_RCV ist bei allen lesenden Funktionscodes notwendig. Das folgende Bild zeigt das Gesamtverhalten der Parameter S_SEND und S_RCV, wenn ein Modbus-Auftrag ausgeführt wird.

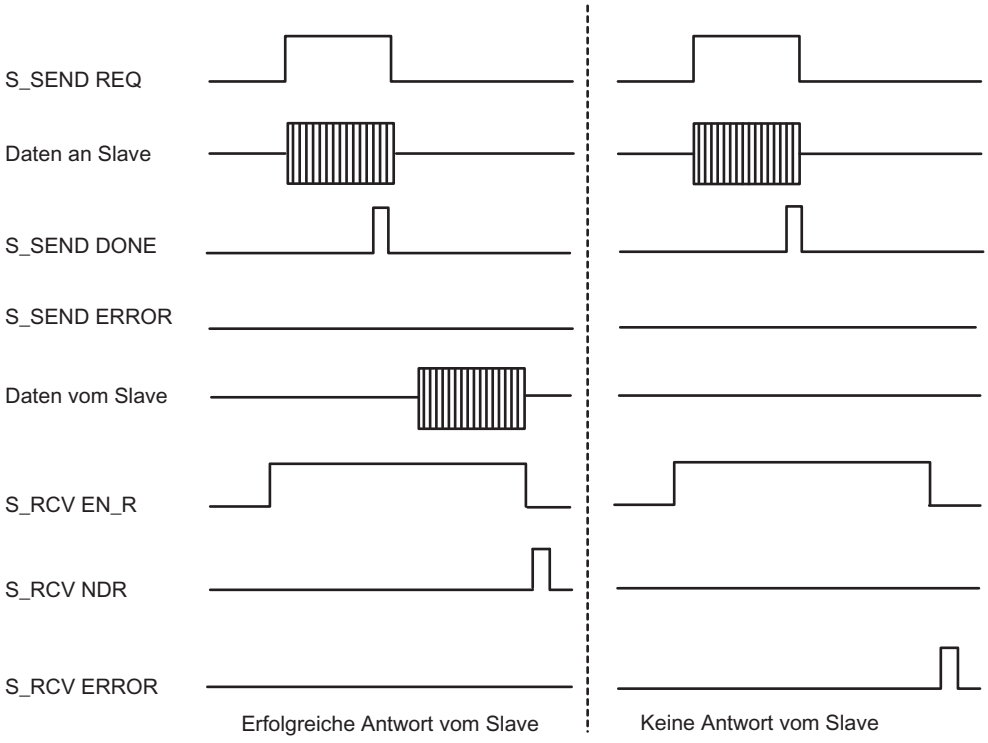

Bild 1-70 Zeitdiagramm eines Modbus-Auftrags

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung S_SEND kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zur Baugruppe abgebrochen und der S_SEND in den Grundzustand versetzt. Daten, die die Baugruppe bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

An dem Parameter LADDR wird die Adresse der anzusprechenden seriellen Schnittstellenbaugruppe ET 200S Modbus/USS angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET des S_SEND ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE zurückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Modbus-Master-Leseauftrag

Da die Schnittstelle zwischen dem Anwenderprogramm und der Schnittstellenbaugruppe halbduplex arbeitet, müssen Sie folgendes beachten:

Nach einem positiv quittierten Modbus-Master-Leseauftrag müssen Sie zuerst mit der Anweisung S_RCV die Empfangsdaten von der Schnittstellenbaugruppe abholen, bevor Sie einen neuen Modbus-Master-Sendeauftrag starten.

Modbus Slave

Initialisierung der Kommunikationsbaugruppe

Einleitung

Die Modbus--Kommunikationsanweisungen (MODB_341, MODB_441 und S_MODB) für den ladbaren Modbus Slave-Treiber müssen in der SIMATIC S7-CPU im zyklischen Teil aufgerufen werden.

Die Modbus-Kommunikationsanweisung initialisiert den CP und führt die Modbus-Funktionen aus, die der Treiber nicht selbständig durchführen kann. Die Modbus-Kommunikationsanweisung muss auch dann im Anwenderprogramm aufgerufen werden, wenn diese Funktionscodes nicht vom Modbus Master-System verwendet werden.

Die Kommunikation zwischen CP und der Anweisung erfolgt über CPU-Betriebssystem-Funktionen und die Anweisungen BSEND (CP 441-2) bzw. P_SND_RK und P_RCV_RK (CP 341), die von der Anweisung aufgerufen wird.

Anlauf, Initialisierung

Nach jedem Neustart oder Wiederanlauf der CPU muss eine Initialisierung der Modbus-Kommunikationsanweisung durchgeführt werden.

Die Initialisierung wird mit einer steigenden Flanke am Eingang CP_START aktiviert.

Die Anweisung löscht zunächst den Instanz-DB, liest die Operandenbereiche E, A, M, T, Z aus der CPU aus und legt diese im Instanz-DB ab. Dadurch können die Schreib-Anforderungen vom Modbus Master-System auf Bereichs-Überschreitung geprüft werden.

Mit einem Sendeauftrag wird dem CP die Nummer des Instanz-DB und der bisher erfolgreiche Verlauf der Initialisierung mitgeteilt.

Wenn die Initialisierung mit Fehler beendet wurde, ist keine Modbus-Kommunikation möglich. Alle Anforderungen vom Modbus Master-System werden mit einem Exception-Code-Telegramm beantwortet.

Instanz-DB

Alle für die Modbus-Kommunikationsanweisung relevanten Daten liegen in einem Instanz-Datenbaustein. Dieser Datenbaustein ist zugleich Instanz-DB (Multi-Instanzen) für die verwendeten Anweisungen und Arbeitsbereich für die Modbus-Kommunikationsanweisung. Ein weiterer Datenbereich ist nicht erforderlich.

Die Modbus-Kommunikationsanweisung arbeitet nur mit dem Instanz-DB und mit Lokaldaten.

Auf den Instanz-DB darf nur lesend zugegriffen werden.

Überwachungszeit

Nach Netz-Ein benötigt der CP für den Hardware- und Speichertest mehrere Sekunden, bis er betriebsbereit ist. Die während dieser Zeit von der Modbus-Kommunikationsanweisung durchgeführten Initialisierungsversuche werden mit Fehler beendet. Deshalb wird der Initialisierungsauftrag während der Überwachungszeit mehrfach wiederholt.

Zeitintervall SYSTAT lesen (nur CP 441-2)

Da ein regelmäßiges SYSTAT-Lesen jeden bzw. jeden zweiten AG-Zyklus den CP und den K-Bus unnötig belasten und somit den Datendurchsatz reduzieren würde, kann ein Zeitintervall zum Auslesen des SYSTAT eingestellt werden (nur relevant für CP 441-2 (MODB_441)).

Funktionscodes

Modbus Master (RTU)

Funktionscode 01 - Read Output Status

Funktion

Diese Funktion ermöglicht das Lesen einzelner Bits vom Slave.

Start-Adresse

Der Parameter **Bit-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Bit-Anzahl

Als **Bit-Anzahl**, Number of coils, ist jeder Wert zwischen **1** und **2040** erlaubt (bei ET 200S 1SI Modbus bis 2008).

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#1	Funktionscode
+2.0	bit_startadr	WORD	W#16#0040	Bit-Startadresse
+4.0	bit_anzahl	INT	16	Bit-Anzahl

Beispiel

Anforderungstelegramm FUNCTION 01:

05H	Slave-Adresse
01H	Funktionscode
00H	Bit-Startadresse "High"
40H	Bit-Startadresse "Low"
00H	Bit-Anzahl "High"
10H	Bit-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 01:

05H	Slave-Adresse
01H	Funktionscode
02H	Bytezähler
01H	<Daten>
17H	<Daten>
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#1701	Daten

Die Daten des Antworttelegramms werden vom Treiber **wortweise** in den Ziel-DB eingetragen.

Das 1. empfangene Byte wird als Low-Byte des 1. Worts "data[1]" eingetragen, das 3. empfangene Byte als Low-Byte des 2. Worts "data[2]", usw.

Wenn weniger als 9 Bit gelesen werden oder wenn nur noch ein Low-Byte gelesen wurde, wird in das verbleibende High-Byte des letzten Worts der Wert **00H** eingetragen.

Funktionscode 02 - Read Input Status

Funktion

Diese Funktion ermöglicht das Lesen einzelner Bits vom Slave.

Start-Aresse

Der Parameter **Bit-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Bit-Anzahl

Als **Bit-Anzahl**, Number of coils, ist jeder Wert zwischen **1** und **2040** erlaubt (bei ET 200S 1SI Modbus bis 2008).

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#2	Funktionscode
+2.0	bit_startadr	WORD	W#16#0120	Bit-Startadresse
+4.0	bit_anzahl	INT	24	Bit-Anzahl

Beispiel

Anforderungstelegramm FUNCTION 02:

05H	Slave-Adresse
02H	Funktionscode
01H	Bit-Startadresse "High"
20H	Bit-Startadresse "Low"
00H	Bit-Anzahl "High"
18H	Bit-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 02:

05H	Slave-Adresse
02H	Funktionscode
03H	Bytezähler
04H	<Daten>
26H	<Daten>
48H	<Daten>
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#2604	Daten
+2.0	data[2]	WORD	W#16#0048	Daten

Die Daten des Antworttelegramms werden vom Treiber **wortweise** in den Ziel-DB eingetragen.

Das 1. empfangene Byte wird als Low-Byte des 1. Wortes "data[1]" eingetragen, das 3. empfangene Byte als Low-Byte des 2. Wortes"data[2]", usw.

Wenn weniger als 9 Bit gelesen werden oder wenn nur noch ein Low-Byte gelesen wurde, wird in das verbleibende High-Byte des letzten Worts der Wert **00H** eingetragen.

Funktionscode 03 - Read Output Registers

Funktion

Diese Funktion ermöglicht das Lesen einzelner Register vom Slave.

Start-Adresse

Der Parameter **Register-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Register-Anzahl

Es können **1** bis **maximal 127 Register** (bei ET 200S 1SI Modbus max. 125 Register) gelesen werden (1 Register = zwei Byte).

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#3	Funktionscode
+2.0	reg_startadr	WORD	W#16#0040	Register-Startadresse
+4.0	reg_anzahl	INT	2	Register-Anzahl

Beispiel

Anforderungstelegramm FUNCTION 03:

05H	Slave-Adresse
03H	Funktionscode
00H	Register-Startadresse "High"
40H	Register-Startadresse "Low"
00H	Register-Anzahl "High"
02H	Register-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 03:

05H	Slave-Adresse
03H	Funktionscode
04H	Bytezähler
21H	Registeradresse 40H Daten "High"
23H	Registeradresse 40H Daten "Low"
25H	Registeradresse 41H Daten "High"
27H	Registeradresse 41H Daten "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#2123	Daten
+2.0	data[2]	WORD	W#16#2527	Daten

Funktionscode 04 - Read Input Registers

Funktion

Diese Funktion ermöglicht das Lesen einzelner Register vom Slave.

Start-Adresse

Der Parameter **Register-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Register-Anzahl

Es können **1** bis **maximal 127 Register** (bei ET 200S 1SI Modbus max. 125 Register) gelesen werden (1 Register = zwei Byte).

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#4	Funktionscode
+2.0	reg_startadr	WORD	W#16#0050	Register-Startadresse
+4.0	reg_anzahl	INT	3	Register-Anzahl

Beispiel

Anforderungstelegramm FUNCTION 04:

05H	Slave-Adresse
04H	Funktionscode
00H	Register-Startadresse "High"
50H	Register-Startadresse "Low"
00H	Register-Anzahl "High"
03H	Register-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 04:

05H	Slave-Adresse
04H	Funktionscode
04H	Bytezähler
31H	Registeradresse 50H Daten "High"
32H	Registeradresse 50H Daten "Low"
33H	Registeradresse 51H Daten "High"
34H	Registeradresse 51H Daten "Low"
35H	Registeradresse 52H Daten "High"
36H	Registeradresse 52H Daten "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#3132	Daten
+2.0	data[2]	WORD	W#16#3334	Daten
+4.0	data[3]	WORD	W#16#3536	Daten

Funktionscode 05 - Force Single Coil

Funktion

Mit dieser Funktion kann beim Slave ein einzelnes Bit gesetzt oder gelöscht werden.

Bit-Adresse

Der Parameter **Bit-Adresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Bit-Status

Als **Bit-Status** sind folgende zwei Werte zulässig:

- FF00H = Bit Setzen
- 0000H = Bit Löschen

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#5	Funktionscode
+2.0	bit_address	WORD	W#16#0019	Bit-Adresse
+4.0	bit_state	WORD	W#16#FF00	Bit-Status

Beispiel

Anforderungstelegramm FUNCTION 05:

05H	Slave-Adresse
05H	Funktionscode
00H	Bit-Adresse "High"
19H	Bit-Adresse "Low"
FFH	Bit Setzen
00H	

xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 05:

05H	Slave-Adresse
05H	Funktionscode
00H	Bit-Adresse "High"
19H	Bit-Adresse "Low"
FFH	Bit-Status "High"
00H	Bit-Status "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Das Antworttelegramm wird nicht in einen Empfangs-DB eingetragen.

Funktionscode 06 - Preset Single Register

Funktion

Mit diesem Befehl kann ein Slave-Register mit einem neuen Wert überschrieben werden.

Register-Adresse

Der Parameter **Register-Adresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Register-Wert

Als **Register-Wert** kann jeder beliebige Wert verwendet werden.

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#6	Funktionscode
+2.0	reg_address	WORD	W#16#0180	Register-Adresse
+4.0	reg_value	WORD	W#16#3E7F	Register-Wert

Beispiel

Anforderungstelegramm FUNCTION 06:

05H	Slave-Adresse
06H	Funktionscode
01H	Register-Adresse "High"
80H	Register-Adresse "Low"
3EH	Register-Wert "High"
7FH	Register-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 06:

05H	Slave-Adresse
06H	Funktionscode
01H	Register-Adresse "High"
80H	Register-Adresse "Low"
3EH	Register-Wert "High"
7FH	Register-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Das Antworttelegramm wird nicht in einen Empfangs-DB eingetragen.

Funktionscode 07 - Read Exception Status

Funktion

Mit diesem Funktionscode können 8 Ereignis-Bits vom angeschlossenen Slave gelesen werden.

Die Anfangsbitnummer der Ereignis-Bits ist durch das angeschlossene Gerät festgelegt und muss somit nicht vom SIMATIC-Anwenderprogramm vorgegeben werden

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#7	Funktionscode

Beispiel

Anforderungstelegramm FUNCTION 07:

05H	Slave-Adresse
07H	Funktionscode
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 07:

05H	Slave-Adresse
07H	Funktionscode
3EH	<Daten>
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#3Exx	Daten

Das einzelne Byte des Antworttelegramms wird vom Treiber im **High-Byte** im Ziel-DB data[1] eingetragen.

Das Low-Byte von data[1] bleibt unverändert.

Als Länge am Parameter LEN der Empfangsanweisung wird der Wert 1 angezeigt.

Funktionscode 08 - Loop Back Diagnostic Test

Funktion

Diese Funktion dient zur Überprüfung der Kommunikations-Verbindung.

Bei diesem Funktionscode wird nur der **Diagnostic Code 0000** unterstützt !

Diagnostic-Code

Für den Parameter **Diagnostic-Code** ist nur der Wert 0000 zulässig.

Test-Wert

Als **Test-Wert** kann jeder beliebige Wert verwendet werden.

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#8	Funktionscode
+2.0	diag_code	WORD	W#16#0000	Diagnostic-Code
+4.0	test_value	WORD	W#16#A5C3	Test-Wert

Beispiel

Anforderungstelegramm FUNCTION 08:

05H	Slave-Adresse
08H	Funktionscode
00H	Diagnostic-Code "High"
00H	Diagnostic-Code "Low"
A5H	Test-Wert "High"
C3H	Test-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 08:

05H	Slave-Adresse
08H	Funktionscode
00H	Diagnostic-Code "High"
00H	Diagnostic-Code "Low"
A5H	Test-Wert "High"
C3H	Test-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Der Slave muss das vom Master empfangene Anforderungstelegramm als Echo unverändert zurücksenden.

Das Antworttelegramm wird nicht in einen RCV-DB eingetragen.

Funktionscode 11 - Fetch Communications Event Counter

Funktion

Mit diesem Funktionscode kann ein 2 Byte **"Status-Word"** und ein 2 Byte **"Event-Counter"** vom Slave gelesen werden.

Sinn und Bedeutung der genannten Parameter sind der Beschreibung "GOULD MODICON Modbus Protocol" zu entnehmen

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#0B	Funktionscode

Beispiel

Anforderungstelegramm FUNCTION 11:

05H	Slave-Adresse
0BH	Funktionscode
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 11:

05H	Slave-Adresse
0BH	Funktionscode
FEH	Status-Word "High"
DCH	Status-Word "Low"
01H	Event-Counter "High"
08H	Event-Counter "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#FEDC	Status-Word
+2.0	data[2]	WORD	W#16#0108	Event-Counter

Hinweis

Einsatz mit der dezentralen Peripherie ET 200S 1SI

Beachten Sie, dass beim Einsatz mit der dezentralen Peripherie ET 200S 1SI das komplette Antworttelegramm des Slave, d. h. mit Modbus Adresse und Funktionscode (zusätzlich 2 Byte), im RCV-Ziel-DB abgelegt wird. Dadurch verschiebt sich der Offset für das Statuswort (Offset 2.0) und für den Ereigniszähler (Offset 4.0).

Funktionscode 12 - Fetch Communications Event Log

Funktion

Mit diesem Funktionscode kann vom Slave Folgendes gelesen werden.

- 2 Byte **"Status-Word"**
- 2 Byte **"Event-Counter",**
- 2 Byte **"Message-Counter** und
- 64 Byte **"Event-Bytes"**

Sinn und Bedeutung der genannten Parameter sind der Beschreibung "GOULD MODICON Modbus Protocol" zu entnehmen.

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#0C	Funktionscode

Beispiel

Anforderungstelegramm FUNCTION 12:

05H	Slave-Adresse
0CH	Funktionscode
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 12:

05H	Slave-Adresse
0CH	Funktionscode
46H	Bytezähler

87H	Status-Word "High"
65H	Status-Word "Low"
01H	Event-Counter "High"
08H	Event-Counter "Low"
02H	Message-Counter "High"
20H	Message-Counter "Low"
01H	Event-Byte 1
12H	Event-Byte 2
:	:
C2H	Event-Byte 63
D3H	Event-Byte 64
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

RCV-Ziel-DB

Inhalt des RCV-Zielbereichs:

Adresse	Name	Typ	Aktualwert	Kommentar
+0.0	data[1]	WORD	W#16#8765	Status-Word
+2.0	data[2]	WORD	W#16#0108	Event-Counter
+4.0	data[3]	WORD	W#16#0220	Message-Counter
+6.0	bytedata[1]	BYTE	B#16#01	Event-Byte 1
+7.0	bytedata[2]	BYTE	B#16#12	Event-Byte 2
:	:			:
+68.0	bytedata[63]	BYTE	B#16#C2	Event-Byte 63
+69.0	bytedata[64]	BYTE	B#16#D3	Event-Byte 64

Funktionscode 15 - Force Multiple Coils

Funktion

Mit diesem Funktionscode können bis zu 2040 Bits (bei ET 200S 1SI Modbus max. 1744 Bits) im Slave geändert werden

Start-Adresse

Der Parameter **Bit-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Bit-Anzahl

Als **Bit-Anzahl**, Number of coils, ist jeder Wert zwischen **1** und **2040** (bei ET 200S 1SI Modbus max. 1744 Bits) erlaubt.

Hiermit wird festgelegt, wie viele Bits im Slave überschrieben werden sollen.

Der im Anforderungstelegramm enthaltene Parameter "Bytezähler" wird vom Treiber aufgrund des übergebenen Parameters "Bit-Anzahl" gebildet.

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#0F	Funktionscode
+2.0	bit_startadr	WORD	W#16#0058	Bit-Startadresse
+4.0	bit_anzahl	INT	10	Bit-Anzahl
+6.0	coil_state[1]	WORD	W#16#EFCD	Status Coil 5FH..58H/57H..50H

Beispiel

Anforderungstelegramm FUNCTION 15:

05H	Slave-Adresse
0FH	Funktionscode
00H	Bit-Adresse "High"
50H	Bit-Adresse "Low"
00H	Bit-Anzahl "High"
0AH	Bit-Anzahl "Low"
02H	Bytezähler
CDH	Status Coil 50H..57H
EFH	Status Coil 58H..59H
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 15:

05H	Slave-Adresse
0FH	Funktionscode
00H	Bit-Adresse "High"
50H	Bit-Adresse "Low"
00H	Bit-Anzahl "High"
0AH	Bit-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Das Antworttelegramm wird nicht in einen Empfangs-DB eingetragen.

Funktionscode 16 - Preset Multiple Registers

Funktion

Der Funktionscode 16 ermöglicht es, mit einem Anforderungstelegramm bis zu **127 Register** (bei ET 200S 1SI Modbus max. 109 Register) im Slave zu überschreiben.

Start-Adresse

Der Parameter **Register-Startadresse** wird vom Treiber nicht überprüft und unverändert gesendet.

Register-Anzahl

Es können **1** bis **maximal 127 Register** (bei ET 200S 1SI Modbus max. 109 Register) gelesen werden (1 Register = zwei Byte).

Der im Anforderungstelegramm enthaltene Parameter "Bytezähler" wird vom Treiber aufgrund des übergebenen Parameters "Register-Anzahl" gebildet.

SEND-Quell-DB

Aufbau des SEND-Quellbereichs:

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	address	BYTE	B#16#5	Slave-Adresse
+1.0	function	BYTE	B#16#10	Funktionscode
+2.0	reg_startadr	WORD	W#16#0060	Register-Startadresse
+4.0	reg_anzahl	INT	3	Register-Anzahl
+6.0	reg_data[1]	WORD	W#16#41A1	Register-Daten
+8.0	reg_data[2]	WORD	W#16#42A2	Register-Daten
+10.0	reg_data[3]	WORD	W#16#43A3	Register-Daten

Beispiel

Anforderungstelegramm FUNCTION 16:

05H	Slave-Adresse
10H	Funktionscode
00H	Register-Adresse "High"
60H	Register-Adresse "Low"
00H	Register-Anzahl "High"
03H	Register-Anzahl "Low"
06H	Bytezähler
41H	<reg_data[1]> "High"
A1H	<reg_data[1]> "Low"
42H	<reg_data[2]> "High"

A2H	<reg_data[2]> "Low"
43H	<reg_data[3]> "High"
A3H	<reg_data[3]> "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm vom Slave FUNCTION 16:

05H	Slave-Adresse
10H	Funktionscode
00H	Register-Adresse "High"
60H	Register-Adresse "Low"
00H	Register-Anzahl "High"
03H	Register-Anzahl "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Das Antworttelegramm wird nicht in den Empfangs-DB eingetragen.

Modbus Slave (RTU)

Funktionscode 01 - Read Coil (Output) Status

Funktion

Diese Funktion ermöglicht das Lesen einzelner Bits aus den nachfolgend aufgeführten SIMATIC-Speicherbereichen durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_address	bit_number	CRC

Antworttelegramm

ADDR	FUNC	Byte_count n	n-Byte DATA	CRC

start_address

Die MODBUS-Bit-Adresse "**start_address**" wird vom Treiber wie folgt interpretiert:

Der Treiber prüft, ob sich "start_address" innerhalb eines der Bereiche befindet, die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 01, 05, 15**" angegeben wurden (von/bis: Merker, Ausgänge, Zeiten, Zähler).

Wenn sich die MODBUS-Bit-Adresse **start_address** in einem dieser Bereiche befindet,	dann erfolgt der Zugriff auf diesen **SIMATIC-Speicherbereich**	
Von aaaaa bis bbbbb	Ab Merker	M uuuuu.0
Von ccccc bis ddddd	Ab Ausgang	A ooooo.0
Von eeeee bis fffff	Ab Zeit	T ttttt
Von ggggg bis hhhhh	Ab Zähler	Z zzzzz

Die Adresse für den Zugriff, die Adress-Umsetzung, wird dann folgendermaßen berechnet:

Zugriff beginnend mit SIMATIC-		Konversionsformel			
Merkerbyte	=	((start_address	– aaaaa)	/ 8)	+ uuuuu
Ausgangsbyte	=	((start_address	– ccccc)	/ 8)	+ ooooo
Zeit	=	((start_address	– eeeee)	/ 16)	+ ttttt
Zähler	=	((start_address	– ggggg)	/ 16)	+ zzzzz

Zugriff auf "Merker" und "Ausgänge"

Beim Zugriff auf die SIMATIC-Bereiche "Merker" und "Ausgänge" wird die verbleibende Rest-Bit_Nummer berechnet und dazu verwendet, um das entsprechende Bit innerhalb des ersten/letzten Merkerbytes oder Ausgangsbytes zu adressieren.

Zugriff auf "Zeiten" und "Zähler"

Bei der Adressberechnung müssen folgende Ergebnisse ohne Rest durch 16 teilbar sein:

- (start_address - eeeee)
- (start_address - ggggg)

Nur wortweiser Zugriff beginnend an der Wortgrenze ist möglich.

bit_number

Als **bit_number**, number of coils, sind Werte zwischen **1** und **2040** (bei ET 200S 1SI Modbus max. 1768 Bits) erlaubt. Diese Anzahl von Bits wird gelesen.

Zugriff auf "Zeiten" und "Zähler"

Beim Zugriff auf die SIMATIC-Bereiche "Zeiten" und "Zähler" muss "bit_number" durch 16 teilbar sein (nur wortweiser Zugriff).

Beachten Sie, dass bei Einsatz des CP 341 maximal 16 "Zeiten" bzw. "Zähler" gelesen werden können.

Hinweis

Beachten Sie die CPU-spezifischen Einschränkungen.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-115 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 01, 05 und 15

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
Von *0* bis *2047*	Ab Merker	M *1000.0*
Von *2048* bis *2559*	Ab Ausgang	A *256.0*
Von *4096* bis *4607*	Ab Zeit	T *100*
Von *4608* bis *5119*	Ab Zähler	Z *200*

Anforderungstelegramm FUNCTION 01:

05H	Slave-Adresse ADDR
01H	Funktionscode FUNC
00H	start_address "High"
40H	start_address "Low"
00H	bit_number "High"
20H	bit_number "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 01:

05H	Slave-Adresse ADDR
01H	Funktionscode FUNC
04H	Byte_count
01H	<DATA 1> M 1008.0 - M 1008.7
17H	<DATA 2> M 1009.0 - M 1009.7
02H	<DATA 3> M 1010.0 - M 1010.7
18H	<DATA 4> M 1011.0 - M 1011.7
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_address" 0040 Hex (64 dezimal) befindet sich im Bereich "Merker":

```
Merkerbyte  =  ((start_address  - aaaaa)  / 8)  + uuuuu
            =  ((64             - 0)      / 8)  + 1000
            =  1008 ;
```

Die verbleibende Rest-Bit_Nummer ergibt sich zu:

```
Rest-Bit_Nr.   =  ((start_address  - aaaaa)  % 8)  [Modulo 8]
               =  ((64             - 0)      % 8)
               =  0 ;
```

Es erfolgt ein Zugriff beginnend mit dem Merker M 1008.0 bis einschließlich M 1011.7.

Bit-Anzahl:

Die MODBUS-Bit-Anzahl "bit_number" 0020 Hex (32 dezimal) bedeutet, dass 32 Bit = 4 Byte gelesen werden sollen.

Weitere Beispiele

Nachfolgend sind einige weitere Zugriffs-Beispiele in tabellarischer Form aufgelistet.

Allen Beispielen liegt die obige Bereichseinteilung zu Grunde.

start_address		Zugriff in SIMATIC beginnend					->	mit
Hex	Dezimal	(Dezimal)						
0000	0	Merker	((0	- 0)	/ 8)	+ 1000	->	M 1000.0
0021	33	Merker	((33	- 0)	/ 8)	+ 1000	->	M 1004.1
0400	1024	Merker	((1024	- 0)	/ 8)	+ 1000	->	M 1128.0
0606	1542	Merker	((1542	- 0)	/ 8)	+ 1000	->	M 1192.6
0840	2112	Ausgang	((2112	- 2048)	/ 8)	+ 256	->	A 264.0
09E4	2532	Ausgang	((2532	- 2048)	/ 8)	+ 256	->	A 316.4
1010	4112	Zeiten	((4112	- 4096)	/ 16)	+ 100	->	T 101
10C0	4288	Zeiten	((4288	- 4096)	/ 16)	+ 100	->	T 112
1200	4608	Zähler	((4608	- 4608)	/ 16)	+ 200	->	Z 200
13E0	5088	Zähler	((5088	- 4608)	/ 16)	+ 200	->	Z 230

Funktionscode 02 - Read Input Status

Funktion

Diese Funktion ermöglicht das Lesen einzelner Bits aus den nachfolgend aufgeführten SIMATIC-Speicherbereichen durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_address	bit_number	CRC

Antworttelegramm

ADDR	FUNC	Byte_count n	n-Byte DATA	CRC

start_address

Die MODBUS-Bit-Adresse "**start_address**" wird vom Treiber wie folgt interpretiert:

Der Treiber prüft, ob sich "start_address" innerhalb eines der Bereiche befindet, die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 02**" eingegeben wurden (von/bis: Merker, Eingänge).

Wenn sich die MODBUS-Bit-Adresse im Bereich **start_address** befindet,	dann erfolgt der Zugriff auf den **SIMATIC-Speicher-bereich**	
Von kkkkk bis lllll	Ab Merker	M vvvvv.0
Von nnnnn bis rrrrr	Ab Eingang	E sssss.0

Die Adresse für den Zugriff, die Adress-Umsetzung, wird dann folgendermaßen berechnet:

Zugriff beginnend mit SIMATIC-		Konversionsformel
Merkerbyte	=	((start_address – *kkkkk*) / 8) + *vvvvv*
Eingangsbyte	=	((start_address – *nnnnn*) / 8) + *sssss*

Zugriff auf "Merker" und "Eingänge"

Beim Zugriff auf die SIMATIC-Bereiche "Merker" und "Eingänge" wird die verbleibende Rest-Bit_Nummer berechnet und dazu verwendet, um das entsprechende Bit innerhalb des ersten/letzten Merkerbytes oder Eingangsbyte zu adressieren.

bit_number

Als **bit_number**, number of coils, ist jeder Wert zwischen **1** und **2040** (bei ET 200S 1SI Modbus max. 1768 Bits) erlaubt. Diese Anzahl von Bits wird gelesen.

Hinweis

Bitte beachten Sie die CPU-spezifischen Einschränkungen.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-116 **Umsetzung der MODBUS-Adressierung bei Funktionscode FC 02**

MODBUS-Adresse im Übertra-gungstelegramm	SIMATIC-Speicherbereich	
von 0 bis 4095	ab Merker	M 2000.0
von 4096 bis 5119	ab Eingang	E 128.0

Anforderungstelegramm FUNCTION 02:

05H	Slave-Adresse ADDR
02H	Funktionscode FUNC
10H	start_address "High"
30H	start_address "Low"
00H	bit_number "High"
18H	bit_number "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 02:

05H	Slave-Adresse ADDR
02H	Funktionscode FUNC
03H	Byte_count
12H	<DATA 1> E 134.0 - E 134.7
34H	<DATA 2> E 135.0 - E 135.7
56H	<DATA 3> E 136.0 - E 136.7
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_address" 1030 Hex (4144 dezimal) befindet sich im Bereich "Eingänge":

Eingangsbyte	=	((start_address	- nnnnn)	/ 8)	+ sssss
	=	((4144	- *4096*)	/ 8)	+ *128*
	=	134 ;			

Die verbleibende Rest-Bit_Nummer ergibt sich zu:

Rest-Bit_Nr.	=	((start_address	- nnnnn)	% 8)	[Modulo 8]
	=	((4144	- 4096)	% 8)	
	=	0 ;			

Es erfolgt ein Zugriff beginnend mit dem Eingang E 134.0 bis einschließlich E 136.7.

Bit-Anzahl:

Die MODBUS-Bit-Anzahl "bit_number" 0018 Hex (24 dezimal) bedeutet, dass 24 Bit = 3 Byte gelesen werden sollen.

Weitere Beispiele

Nachfolgend sind einige weitere Zugriffs-Beispiele in tabellarischer Form aufgelistet.

Allen Beispielen liegt die obige Bereichseinteilung zu Grunde.

start_address		Zugriff in SIMATIC beginnend					->	mit
Hex	Dezimal	(Dezimal)						
0000	0	Merker	((0	- 0)	/ 8)	+ 2000	->	M 2000.0
0071	113	Merker	((113	- 0)	/ 8)	+ 2000	->	M 2014.1
0800	2048	Merker	((2048	- 0)	/ 8)	+ 2000	->	M 2256.0
0D05	3333	Merker	((3333	- 0)	/ 8)	+ 2000	->	M 2416.5
1000	4096	Eingang	((4096	- 4096)	/ 8)	+ 128	->	E 128.0
10A4	4260	Eingang	((4260	- 4096)	/ 8)	+ 128	->	E 148.4

Funktionscode 03 - Read Output Registers

Funktion

Diese Funktion ermöglicht das Lesen von Datenworten aus einem Datenbaustein durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_register	register_number	CRC

Antworttelegramm

ADDR	FUNC	Byte_count n	n/2-Register DATA (High,Low)	CRC

start_register

Die MODBUS-Register-Adresse "**start_register**" wird vom Treiber wie folgt interpretiert:

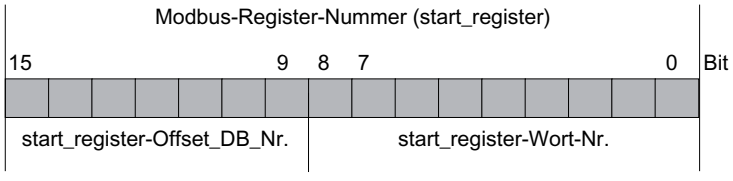

Zur weiteren Adress-Bildung verwendet der Treiber die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 03, 06, 16**" eingegebene "Basis-DB-Nummer" (ab DB xxxxx).

Die Adresse für den Zugriff, die Adress-Umsetzung, wird dann folgendermaßen berechnet:

Zugriff auf SIMATIC-		Konversionsformel
Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer *xxxxx* + start_register-Offset_DB_Nr.)
Datenwort DBW	=	(start_register-Wort_Nr. * 2)

Berechnungs-Formel für start_register

Wenn der resultierende DB bekannt ist, der gelesen werden soll, kann die im Master-System benötigte MODBUS-Adresse **start_register** nach folgender Formel berechnet werden:

start_register = ((Resultierender DB - Basis-DB-Nummer) * 512) + (Datenwort_DBW / 2)

Dabei dürfen nur geradzahlige Datenwort-Nummern zu Grunde gelegt werden.

register_number

Als register_**number**, number of registers, ist jeder Wert zwischen **1** und **127** (bei ET 200S 1SI Modbus max. 110 Register) erlaubt. Diese Anzahl von Registern wird gelesen. Dabei ist folgende Regel zu beachten:

$(\text{register_number})_{max}$ = 512 - start_register

Hinweis

Beachten Sie die CPU-spezifischen Einschränkungen.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-117 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 03, 06 und 16

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
0	ab Datenbaustein (Basis-DB-Nummer)	DB 800

Anforderungstelegramm FUNCTION 03:

05H	Slave-Adresse ADDR
03H	Funktionscode FUNC
00H	start_register "High"
50H	start_register "Low"

00H	register_number "High"
02H	register_number "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 03:

05H	Slave-Adresse ADDR
03H	Funktionscode FUNC
04H	Byte_count
87H	<DATA 1> DBW 160 "High"
65H	<DATA 2> DBW 160 "Low"
43H	<DATA 3> DBW 161 "High"
21H	<DATA 4> DBW 161 "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_register" 0050 Hex (80 dezimal) wird interpretiert:

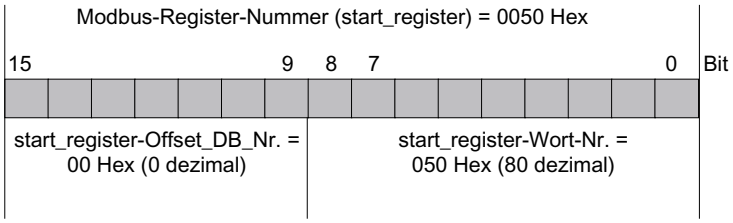

Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer xxxx + start_register-Offset_DB_Nr.)
	=	(800 + 0)
	=	800;

Datenwort DBW	=	(start_register-Wort_Nr. * 2)
	=	(80 * 2)
	=	160;

Es erfolgt ein Zugriff auf den DB 800, Datenwort DBW 160.

Register-Anzahl:

Die MODBUS-Register-Anzahl "register_number" 0002 Hex (2 dezimal) bedeutet, dass 2 Register = 2 Datenworte gelesen werden.

Weitere Beispiele

Nachfolgend sind einige weitere Zugriffs-Beispiele in tabellarischer Form aufgelistet.

			start_register				
start_register		Basis-DB-Nr.	Offset-DB_Nr.	Wort-Nummer		Resultierender DB	DBW
Hex	Dezimal	Dezimal	Dezimal	Hex	Dezimal	Dezimal	Dezimal
0000	0	800	0	000	0	800	0
01F4	500	800	0	1F4	500	800	1000
0200	512	800	1	000	0	801	0
02FF	767	800	1	0FF	255	801	510
0300	768	800	1	100	256	801	512
03FF	1023	800	1	1FF	511	801	1022
0400	1024	800	2	000	0	802	0

Funktionscode 04 - Read Input Registers

Funktion

Diese Funktion ermöglicht das Lesen von Datenworten aus einem Datenbaustein durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_register	register_number	CRC

Antworttelegramm

ADDR	FUNC	Byte_count n	n/2-Register DATA (High,Low)	CRC

start_register

Die MODBUS-Register-Adresse "**start_register**" wird vom Treiber wie folgt interpretiert:

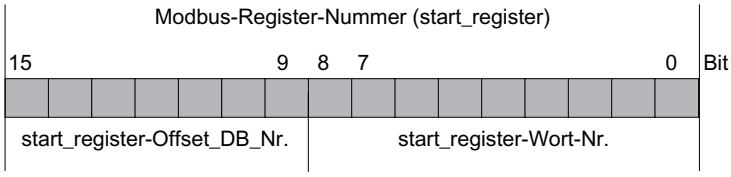

Zur weiteren Adress-Bildung verwendet der Treiber die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 04**" angegebene "Basis-DB-Nummer" (ab DB xxxxx).

Die Adress-Berechnung für den Zugriff, Adress-Umsetzung, erfolgt dann zweistufig auf folgende Weise:

Zugriff auf SIMATIC-		Konversionsformel
Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer *xxxxx* + start_register-Offset_DB_Nr.)
Datenwort DBW	=	(start_register-Wort_Nr. * 2)

Berechnungs-Formel für start_register

Wenn der resultierende DB bekannt ist, der gelesen werden soll, kann die im Master-System benötigte MODBUS-Adresse **start_register** nach folgender Formel berechnet werden:

start_register = ((Resultierender DB - Basis-DB-Nummer) * 512) + (Datenwort_DBW / 2)

Nur geradzahlige Datenwort-Nummern sind zulässig.

register_number

Als register_**number**, number of registers, ist jeder Wert zwischen **1** und **127** (bei ET 200S 1SI Modbus max. 110 Register) erlaubt. Diese Anzahl von Registern wird gelesen. Beachten Sie dabei folgende Regel:

$(\text{register_number})_{max}$ = 512 - start_register

Hinweis

Beachten Sie die CPU-spezifischen Einschränkungen.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-118 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 04

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
0	Ab Datenbaustein (Basis-DB-Nummer)	DB 900

Anforderungstelegramm FUNCTION 04:

05H	Slave-Adresse ADDR
04H	Funktionscode FUNC

02H	start_register "High"
C0H	start_register "Low"
00H	register_number "High"
03H	register_number "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 04:

05H	Slave-Adresse ADDR
04H	Funktionscode FUNC
06H	Byte_count
A1H	<DATA 1> DBW 384 "High"
A2H	<DATA 2> DBW 384 "Low"
A3H	<DATA 3> DBW 385 "High"
A4H	<DATA 4> DBW 385 "Low"
A5H	<DATA 5> DBW 386 "High"
A6H	<DATA 6> DBW 386 "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_register" 02C0 Hex (704 dezimal) wird interpretiert:

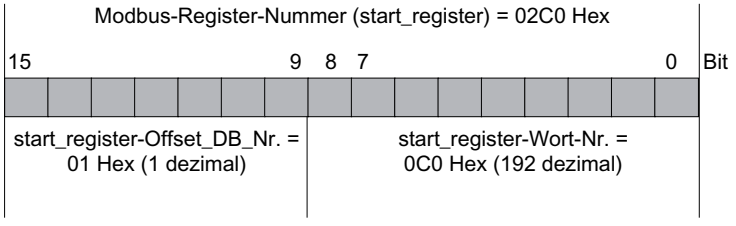

Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer xxxx + start_register-Offset_DB_Nr.)
	=	(900 + 1)
	=	901;

Datenwort DBW	=	(start_register-Wort_Nr. * 2)
	=	(192 * 2)
	=	384;

Es erfolgt ein Zugriff auf den DB 901, Datenwort DBW 384.

Register-Anzahl:

Die MODBUS-Register-Anzahl "register_number" 0003 Hex (3 dezimal) bedeutet, dass 3 Register = 3 Datenworte gelesen werden.

Weitere Beispiele

Nachfolgend sind einige weitere Zugriffs-Beispiele in tabellarischer Form aufgelistet.

			start_register				
start_register		Basis-DB-Nr.	Offset-DB_Nr.	Wort-Nummer		Resultierender DB	DBW
Hex	dezimal	dezimal	dezimal	Hex	dezimal	dezimal	dezimal
0000	0	900	0	000	0	900	0
0064	100	900	0	064	100	900	200
00C8	200	900	0	0C8	200	900	400
0190	400	900	0	190	400	900	800
1400	5120	900	10	000	0	910	0
1464	5220	900	10	064	100	910	200
14C8	5320	900	10	0C8	200	910	400

Funktionscode 05 - Force Single Coil

Funktion

Diese Funktion ermöglicht das Schreiben eines Bits in den nachfolgend aufgeführten SIMATIC-Speicherbereichen der CPU durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	coil_address	DATA-on/off	CRC

Antworttelegramm

ADDR	FUNC	coil_address	DATA-on/off	CRC

coil_address

Die MODBUS-Bit-Adresse "**coil_address**" wird vom Treiber wie folgt interpretiert:

Der Treiber prüft, ob sich "coil_address" innerhalb eines der Bereiche befindet, die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 01, 05, 15**" eingegeben wurden (von / bis : Merker, Ausgänge, Zeiten, Zähler).

Befindet sich die MODBUS-Bit-Adresse **coil_address** im Bereich	erfolgt der Zugriff auf den **SIMATIC-Speicherbereich**	
von aaaaa bis bbbbb	ab Merker	M uuuuu.0
von ccccc bis ddddd	ab Ausgang	A ooooo.0

Die Adress-Berechnung für den Zugriff (Adress-Umsetzung) erfolgt dann folgendermaßen:

Zugriff beginnend mit SIMATIC		Konversionsformel			
Merkerbyte	=	((coil_address	- *aaaaa*)	/ 8)	+ *uuuuu*
Ausgangsbyte	=	((coil_address	- *ccccc*)	/ 8)	+ *ooooo*

Zugriff auf "Merker" und "Ausgänge"

Beim Zugriff auf die SIMATIC-Bereiche "Merker" und "Ausgänge" wird die verbleibende Rest-Bit_Nummer berechnet und dazu verwendet, um das entsprechende Bit innerhalb des Merker- oder Ausgangsbytes zu adressieren.

Zugriff auf "Zeiten" und "Zähler"

Ein Zugriff auf die SIMATIC-Bereiche Zeiten und Zähler ist bei diesem Funktionscode FC 05 nicht zulässig und wird vom Treiber mit einem Fehlertelegramm abgewiesen.

DATA-on/off

Als **DATA-on/off** sind folgende zwei Werte zulässig:

- FF00H -> Bit Setzen
- 0000H -> Bit Löschen.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-119 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 01, 05, 15

MODBUS-Adresse im Übertragungstelegramm	**SIMATIC-Speicherbereich**	
von 0 bis 2047	ab Merker	M 1000.0
von 2048 bis 2559	ab Ausgang	A 256.0

Anforderungstelegramm FUNCTION 05:

05H	Slave-Adresse ADDR
05H	Funktionscode FUNC
08H	coil_address "High"
09H	coil_address "Low" A257.1
FFH	DATA-on/off "High"
00H	DATA-on/off "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 05:

05H	Slave-Adresse ADDR
05H	Funktionscode FUNC
08H	coil_address "High"
09H	coil_address "Low" A257.1
FFH	DATA-on/off "High"
00H	DATA-on/off "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "coil_address" 0809 Hex (2057 dezimal) befindet sich im Bereich "Ausgänge":

```
Ausgangsbyte   =  ((coil_address   - ccccc)  / 8)   + ooooo
               =  ((2057           - 2048)   / 8)   + 256
               =   257 ;
```

Die verbleibende Rest-Bit_Nummer ergibt sich zu:

```
Rest-Bit_Nr.   =  ((coil_address   - ccccc)  % 8)   [Modulo 8]
               =  ((2057           - 2048)   % 8)
               =   1 ;
```

Es erfolgt ein Zugriff auf den Ausgang A 257.1.

Weitere Beispiele

Weitere Zugriffs-Beispiele auf Merker und Ausgänge können bei FC 01 nachgeschlagen werden.

Funktionscode 06 - Preset Single Register

Funktion

Diese Funktion ermöglicht das Schreiben eines Datenworts in einem Datenbaustein der CPU durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_register	DATA-value (High,Low)	CRC

Antworttelegramm

ADDR	FUNC	start_register	DATA-value (High,Low)	CRC

start_register

Die MODBUS-Register-Adresse "**start_register**" wird vom Treiber wie folgt interpretiert:

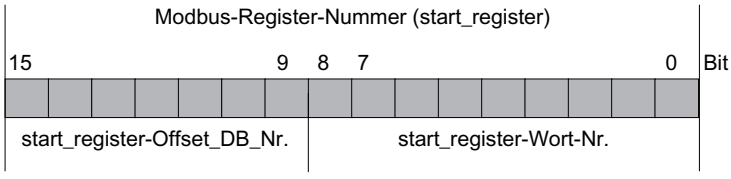

Zur weiteren Adress-Bildung verwendet der Treiber die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 03, 06, 16**" eingegebene "Basis-DB-Nummer" ab DB xxxxx.

Die Adress-Berechnung für den Zugriff, die Adress-Umsetzung, erfolgt dann zweistufig auf folgende Weise:

Zugriff auf SIMATIC-		Konversionsformel
Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer xxxxx + start_register-Offset_DB_Nr.)
Datenwort DBW	=	(start_register-Wort_Nr. * 2)

Berechnungs-Formel für start_register

Wenn der resultierende DB bekannt ist, in den geschrieben werden soll, kann die im Master-System benötigte MODBUS-Adresse **start_register** nach folgender Formel berechnet werden:

start_register = ((Resultierender DB - Basis-DB-Nummer) * 512) + (Datenwort_DBW / 2)

Nur geradzahlige Datenwort-Nummern sind zulässig.

DATA-value

Als DATA-**value**, Register-Wert, kann jeder beliebige Wert verwendet werden.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-120 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 03, 06 und 16

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
0	ab Datenbaustein (Basis-DB-Nummer)	DB 800

Anforderungstelegramm FUNCTION 06:

05H	Slave-Adresse ADDR
06H	Funktionscode FUNC
01H	start_register "High"
80H	start_register "Low" DBW 768
2BH	DATA-value "High"
1AH	DATA-value "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 06:

05H	Slave-Adresse ADDR
06H	Funktionscode FUNC
01H	start_register "High"
80H	start_register "Low" DBW 768
2BH	DATA-value "High"
1AH	DATA-value "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_register" 0180 Hex (384 dezimal) wird interpretiert:

Modbus-Register-Nummer (start_register) = 0180 Hex

15						9	8	7							0	Bit
0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
start_register-Offset_DB_Nr. = 00 Hex (0 dezimal)							start_register-Wort-Nr. = 180 Hex (384 dezimal)									

Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer xxxx + start_register-Offset_DB_Nr.)
	=	(800 + 0) = 800;

Datenwort DBW	=	(start_register-Wort_Nr. * 2)
	=	(384 * 2) = 768;

Es erfolgt ein Zugriff auf den DB 800, Datenwort DBW 768.

Weitere Beispiele

Weitere Zugriffs-Beispiele siehe bei FC 03.

Funktionscode 08 - Loop Back Diagnostic Test

Funktion

Diese Funktion dient zur Überprüfung der Kommunikations-Verbindung.

Sie hat keine Auswirkung auf die S7-CPU und die Anwenderprogramme und Anwenderdaten. Das empfangene Telegramm wird vom Treiber selbständig an das Master-System zurückgesendet.

Anforderungstelegramm

ADDR	FUNC	Diagnostic-Code (High,Low)	Test-Data	CRC

Antworttelegramm

ADDR	FUNC	Diagnostic-Code (High,Low)	Test-Data	CRC

Diagnostic-Code

Es wird nur der **Diagnostic Code 0000** unterstützt !

Test-Data

Beliebiger 16 Bit-Wert.

Anwendungs-Beispiel

Anforderungstelegramm FUNCTION 08:

05H	Slave-Adresse ADDR
08H	Funktionscode FUNC
00H	Diagnostic-Code "High"
00H	Diagnostic-Code "Low"
A5H	Test-Wert "High"
C3H	Test-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 08:

05H	Slave-Adresse ADDR
08H	Funktionscode FUNC
00H	Diagnostic-Code "High"
00H	Diagnostic-Code "Low"
A5H	Test-Wert "High"
C3H	Test-Wert "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Funktionscode 15 - Force Multiple Coils

Funktion

Diese Funktion ermöglicht das Schreiben mehrerer Bits in den nachfolgend aufgeführten SIMATIC-Speicherbereichen durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_address	quantity	byte_count n	n-DATA	CRC

Antworttelegramm

ADDR	FUNC	start_address	quantity	CRC

start_address

Die MODBUS-Bit-Adresse "**start_address**" wird vom Treiber wie folgt interpretiert:

Der Treiber prüft, ob sich "start_address" innerhalb eines der Bereiche befindet, die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 01, 05, 15**" eingegeben wurden (von/bis: Merker, Ausgänge, Zeiten, Zähler).

Wenn sich die MODBUS-Bit-Adresse im Bereich **start_address** befindet,	dann erfolgt der Zugriff auf den **SIMATIC-Speicherbereich**	
Von aaaaa bis bbbbb	Ab Merker	M uuuuu.0
Von ccccc bis ddddd	Ab Ausgang	A ooooo.0

Die Adress-Berechnung für den Zugriff, die Adress-Umsetzung, erfolgt dann folgendermaßen:

Zugriff beginnend mit SIMATIC-		Konversionsformel
Merkerbyte	=	((start_address - aaaaa) / 8) + uuuuu
Ausgangsbyte	=	((start_address - ccccc) / 8) + ooooo

Zugriff auf "Merker" und "Ausgänge"

Beim Zugriff auf die SIMATIC-Bereiche "Merker" und "Ausgänge" wird die verbleibende Rest-Bit_Nummer berechnet und dazu verwendet, um das entsprechende Bit innerhalb des Merker- oder Ausgangsbytes zu adressieren.

Zugriff auf "Zeiten" und "Zähler"

Ein Zugriff auf die SIMATIC-Bereiche Zeiten und Zähler ist bei diesem Funktionscode FC 15 nicht zulässig und wird vom Treiber mit einem Fehlertelegramm abgewiesen.

Quantity

Als **quantity**, Bit-Anzahl, ist jeder Wert zwischen **1** und **2040** (bei ET 200S 1SI Modbus max. 1976) erlaubt.

Hinweis

Bitte beachten Sie die CPU-spezifischen Einschränkungen.

DATA

Im **DATA-Feld** sind beliebige Werte als Bit-Zustände enthalten.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-121 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 01, 05 und 15

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
Von 0 bis 2047	Ab Merker	M 1000.0
Von 2048 bis 2559	Ab Ausgang	A 256.0

Aktion:

Das MODBUS-Master-System möchte die Merker M 1144.0 ... M 1144.7 und M 1145.0 ... M 1145.3 mit folgenden Bit-Zuständen beschreiben:

Merker M 1144	7	6	5	4	3	2	1	0	Bit
	ON	ON	OFF	OFF	ON	ON	OFF	ON	
Merker M 1145	7	6	5	4	3	2	1	0	Bit
	-	-	-	-	ON	OFF	OFF	ON	

Anforderungstelegramm FUNCTION 15:

05H	Slave-Adresse ADDR
0FH	Funktionscode FUNC
04H	start_address "High"
80H	start_address "Low" (M 1144.0 ...)
00H	quantity "High"
0CH	quantity "Low" (12 Bits)
02H	bytecount
CDH	Zustand coil (M 1144.0 ... M 1144.7)
09H	Zustand coil (M 1145.0 ... M 1145.3)
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 15:

05H	Slave-Adresse ADDR
0FH	Funktionscode FUNC
04H	start_address "High"
80H	start_address "Low"

00H	quantity "High"
0CH	quantity "Low"
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "coil_address" 0480 Hex (1152 dezimal) befindet sich im Bereich "Merker":

```
Merkerbyte    =  ((start_address  - aaaaa)  / 8)   + uuuuu
              =  ((1152           - 0)      / 8)   + 1000
              =   1144 ;
```

Die verbleibende Rest-Bit_Nummer ergibt sich zu:

```
Rest-Bit_Nr.  =  ((start_address  - aaaaa)  % 8)  [Modulo 8]
              =  ((1152           - 0)      % 8)
              =   0 ;
```

Es erfolgt ein Zugriff auf Merker beginnend bei M 1144.0.

Weitere Beispiele

Weitere Zugriffs-Beispiele auf Merker und Ausgänge können bei FC 01 nachgesehen werden.

Funktionscode 16 - Preset Multiple Registers

Funktion

Der Funktionscode ermöglicht das Schreiben von mehreren Datenworten in einem Datenbaustein der SIMATIC-CPU durch das MODBUS-Master-System.

Anforderungstelegramm

ADDR	FUNC	start_register	quantity	byte_count n	n-DATA (High,Low)	CRC

Antworttelegramm

ADDR	FUNC	start_register	quantity	CRC

start_register

Die MODBUS-Register-Adresse "**start_register**" wird vom Treiber wie folgt interpretiert:

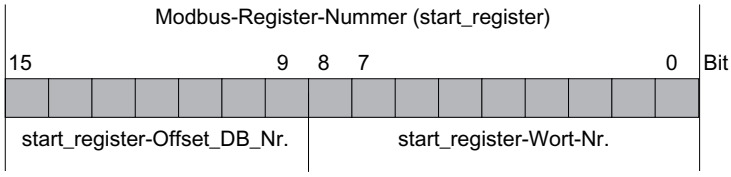

Zur weiteren Adress-Bildung verwendet der Treiber die bei der Parametrierung in der Maske "**Umsetzung der MODBUS-Adressierung bei FC 03, 06, 16**" eingegebene "Basis-DB-Nummer" ab DB xxxxx.

Die Adress-Berechnung für den Zugriff, die Adress-Umsetzung, erfolgt dann zweistufig auf folgende Weise:

Zugriff auf SIMATIC-	Konversionsformel
Datenbaustein DB (resultierender DB)	= (Basis-DB-Nummer *xxxxx* + start_register-Offset_DB_Nr.)
Datenwort DBW	= (start_register-Wort_Nr. * 2)

Berechnungs-Formel für start_register

Wenn der resultierende DB bekannt ist, in den geschrieben werden soll, kann die im Master-System benötigte MODBUS-Adresse **start_register** nach folgender Formel berechnet werden:

start_register = ((Resultierender DB - Basis-DB-Nummer) * 512) + (Datenwort_DBW / 2)

Nur geradzahlige Datenwort-Nummern sind zulässig.

Quantity

Als **quantity**, Register-Anzahl, ist jeder Wert zwischen **1** und **127** (bei ET 200S 1SI Modbus max. 123) erlaubt. Dabei ist folgende Regel zu beachten:

$(\text{Quantity})_{max}$ = 512 - start_register

Hinweis

Bitte beachten Sie die CPU-spezifischen Einschränkungen.

DATA (High,Low)

Als **DATA (High,Low)** (Register-Wert) kann jeder beliebige Wert verwendet werden.

Anwendungs-Beispiel

Beispiel für die Parametrierung:

Tabelle 1-122 Umsetzung der MODBUS-Adressierung bei Funktionscode FC 03, 06, 16

MODBUS-Adresse im Übertragungstelegramm	SIMATIC-Speicherbereich	
0	ab Datenbaustein (Basis-DB-Nummer)	DB 800

Aktion:

Das MODBUS-Master-System will die Datenworte DBW 100, DBW 102, DBW 104 des DB 800 mit den Werten CD09 Hex, DE1A Hex, EF2B Hex beschreiben.

Anforderungstelegramm FUNCTION 16:

05H	Slave-Adresse ADDR
10H	Funktionscode FUNC
00H	start_register "High"
32H	start_register "Low" DBW 100
00H	quantity "High"
03H	quantity "Low" (3 Register)
06H	bytecount
CDH	register value -High (DBW100)
09H	register value -Low
DEH	register value -High (DBW102)
1AH	register value -Low
EFH	register value -High (DBW104)
2BH	register value -Low
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Antworttelegramm FUNCTION 16:

05H	Slave-Adresse ADDR
10H	Funktionscode FUNC
00H	start_register "High"
32H	start_register "Low"
00H	quantity "High"
03H	quantity "Low" (3 Register)
xxH	CRC-Prüfcode "Low"
xxH	CRC-Prüfcode "High"

Adress-Berechnung:

Die MODBUS-Adresse "start_register" 0032 Hex (50 dezimal) wird interpretiert:

Modbus-Register-Nummer (start_register) = 0032 Hex

15						9	8	7							0	Bit
0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	
start_register-Offset_DB_Nr. = 00 Hex (0 dezimal)							start_register-Wort-Nr. = 32 Hex (50 dezimal)									

Datenbaustein DB (resultierender DB)	=	(Basis-DB-Nummer xxxx + start_register-Offset_DB_Nr.)
	=	(800 + 0) = 800;

Datenwort DBW	=	(start_register-Wort_Nr. * 2)
	=	(50 * 2) = 100;

Es erfolgt ein Zugriff auf den DB 800, Datenwort DBW 100.

Weitere Beispiele

Weitere Zugriffs-Beispiele siehe bei FC 03.

Fehlermeldebereich

Nummernschema Ereignisklasse/Ereignisnummer

Bit-Nr.	15		13	12			8	7						0
	Reserve			Ereignisklasse				Ereignisnummer (Fehlernummer)						

Ereignisklassen

Ereignisklasse	Bedeutung
2	Fehler bei der Initialisierung
5	Fehler bei Bearbeitung eines CPU-Auftrags
8	Empfangsfehler
14 ($0E_H$)	Allgemeiner Verarbeitungsfehler
30 ($1E_H$)	Fehler während der Kommunikation zwischen serieller Schnittstelle und CPU.

Ereignisklasse / Ereignisnummer

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0201	Keine (gültige) Parametrierung vorhanden.	Versorgen Sie die Baugruppe mit korrekten Parametern. Überprüfen Sie ggf. die ordnungsgemäße Installation der Anlage.
0502	Auftrag unzulässig in diesem Betriebszustand der seriellen Schnittstellenbaugruppe ET 200S Modbus/USS (Beispiel: Geräteschnittstelle ist nicht parametriert).	Werten Sie den Diagnosealarm aus, und beheben Sie den Fehler entsprechend.
050E	Ungültige Telegrammlänge	Das Sendetelegramm ist länger als 224 Byte. Der Sendeauftrag wurde von der Baugruppe ET 200S Modbus/USS abgebrochen. Wählen Sie eine kleinere Telegrammlänge.
0518	• Übertragungslänge beim Senden ist zu groß (> 4 kByte) • Die Übertragungslänge beim SEND ist zu klein.	Parameter LEN am SEND überprüfen.
0530	Modbus-Master-Sendeauftrag abgelehnt, da die Antwort des Koppelpartners auf einen vorherigen lesenden Modbus-Master-Sendeauftrag noch nicht abgerufen wurde.	Nach einem erfolgreichen lesenden Modbus-Master-Sendeauftrag müssen Sie zuerst die Antwort des Koppelpartners aus dem Modul auslesen, bevor Sie einen neuen Modbus-Master-Sendeauftrag starten.
0551	Rahmenablauffehler bei der Kommunikation zwischen der seriellen Schnittstellenbaugruppe ET 200S Modbus/USS und der CPU. Der Fehler ist beim Übertragen eines empfangenen Telegramms der seriellen Schnittstellenbaugruppe ET 200S SI in der CPU aufgetreten.	Die Baugruppe und die CPU haben die Übertragung abgebrochen. Wiederholen Sie den Empfangsauftrag. Die serielle Schnittstellenbaugruppe ET 200S Modbus/USS sendet die empfangene Meldung erneut.
0806	Zeichenverzugszeit ZVZ überschritten	Fehler beim Partnergerät oder Störungen auf der Übertragungsleitung beseitigen.
080A	Überlauf des Empfangspuffers beim Master während des Empfangs des Antworttelegramms.	Prüfen Sie die Protokolleinstellungen des Slave.
080C	Bei einem Zeichen wurde ein Übertragungsfehler erkannt (Paritätsfehler, Overrun-Fehler, Stoppbitfehler (Frame)).	Prüfen Sie, ob Störungen auf die Übertragungsleitung einwirken. Ändern Sie gegebenenfalls den Systemaufbau bzw. die Leitungsverlegung. Prüfen Sie, ob die Protokollparameter Datenübertragungsrate, Datenbitanzahl, Parität, Stoppbitanzahl bei Kommunikationsbaugruppe und Koppelpartner gleich eingestellt sind.
080D	BREAK Die Empfangsleitung zum Partnergerät ist unterbrochen.	Stellen Sie die Verbindung zwischen den Geräten her oder schalten Sie das Partnergerät ein. Prüfen Sie bei TTY-Betrieb, ob im Ruhezustand Linienstrom fließt. Überprüfen und ändern Sie gegebenenfalls bei einer RS422/485 (X27)-Verbindung die Vorbelegung der 2-Draht Empfangsleitung R(A), R(B).
0810	Paritätsfehler: Es liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsrate, Parität und Stopbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0811	Zeichenrahmenfehler: Es liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsrate, Parität und Stopbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0812	Nachdem die serielle Schnittstelle CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder entsorgen Sie die serielle Schnittstelle schneller.
0830	Die Antwortüberwachungszeit nach Senden eines Anforderungstelegramms ist abgelaufen, ohne dass der Beginn des Antworttelegramms erkannt wurde.	Prüfen Sie, ob die Übertragungsleitung gestört ist (gegebenenfalls Schnittstellen-Analyzer verwenden). Prüfen Sie, ob die Protokollparameter Datenübertragungsrate, Datenbitanzahl, Parität, Stoppbitanzahl bei Kommunikationsbaugruppe und Koppelpartner gleich eingestellt sind. Prüfen Sie, ob die im Eigenschaften Dialog der Baugruppe parametrierte Antwortüberwachungszeit groß genug eingestellt ist. Prüfen Sie, ob die angegebene Slave-Adresse vorhanden ist.
0831	Das erste Zeichen des Antworttelegramms vom Slave ist ungleich der im Anforderungstelegramm gesendeten Slave-Adresse (bei Betriebsart "Normal").	Ein falscher Slave hat geantwortet. Prüfen Sie, ob die Übertragungsleitung gestört ist (gegebenenfalls Schnittstellen-Analyzer verwenden).
0832	Überlauf des Empfangspuffers in der Kommunikationsbaugruppe bei Empfang des Antworttelegramms.	Überprüfen Sie die Protokolleinstellungen beim Slave.
0833	Bit- oder Registeranzahl bei Funktionscode FC 15/16 und Telegrammelement byte_count passen nicht zusammen.	Korrigieren Sie die Bit-/Registeranzahl oder byte_count.
0834	Unzulässige Bit-Codierung für "Bit setzen"/Bit rücksetzen" erkannt.	Verwenden Sie für FC05 nur die Codierungen 0000Hex oder FF00Hex.
0835	Unzulässiger Diagnose-Subcode (ungleich 0000Hex) bei Funktionscode FC 08 "Loop Back Test" erkannt.	Verwenden Sie für FC08 nur den Subcode 0000Hex.
0836	Der intern gebildete Wert der CRC-16-Prüfsumme stimmt nicht mit der empfangenen CRCPrüfsumme überein.	Überprüfen Sie die Bildung der CRC-Püfsumme beim Modbus-Mastersystem.
0837	Telegrammablauffehler: Das Modbus-Mastersystem sendete ein neues Anforderungstelegramm ehe vom Treiber das letzte Antworttelegramm übertragen wurde.	Erhöhen Sie das Timeout auf das Slave-Antworttelegramm beim Modbus-Mastersystem.
0850	Länge des Empfangtelegramms größer als 224 Byte bzw. größer als die parametrierte Telegrammlänge.	Telegrammlänge des Partners anpassen.
0E01	Fehler bei der Initialisierung des treiberspezifischen SCC-Prozesses.	Treiber neu parametrieren und laden.
0E02	Fehler bei Anlauf des Treibers: Falscher SCC-Prozess aktiv (SCC-Treiber). Der Treiber ist mit diesem SCC-Treiber nicht funktionsfähig.	Treiber neu parametrieren und laden.

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0E03	Fehler bei Anlauf des Treibers: Falscher Datenübergabe-Prozess aktiv (Schnittstelle zu den Anweisungen). Der Treiber ist mit diesem Datenübergabe-Prozess nicht funktionsfähig.	Treiber neu parametrieren und laden.
0E04	Fehler bei Anlauf des Treibers: Unzulässiges Schnittstellenmodul Der Treiber kann mit dem parametrierten Schnittstellenmodul nicht betrieben werden.	Parametrierung überprüfen und korrigieren.
0E05	Fehler bei Treiber-Dongle: Es ist kein Dongle gesteckt oder der gesteckte Dongle ist fehlerhaft. Der Treiber ist nicht betriebsbereit.	Überprüfen Sie, ob in der Kommunikationsbaugruppe ein Treiber-Dongle gesteckt ist. Wenn ein Dongle gesteckt ist, ersetzen Sie den fehlerhaften Dongle durch einen korrekten Dongle.
0E06	Fehler bei Treiber-Dongle: Der Dongle hat keinen gültigen Inhalt. Der Treiber ist nicht betriebsbereit.	Besorgen Sie sich bei Ihrer SIEMENS Niederlassung, von der Sie den Treiber erhalten haben, einen korrekten Dongle.
0E10	Interner Fehler Prozedur: Default-Zweig im Prozedur-Automat.	Neuanlauf der Kommunikationsbaugruppe (Netz_EIN)
0E11	Interner Fehler Prozedur: Default-Zweig bei Prozedurzustand Senden/Empfang.	Neuanlauf de Kommunikationsbaugruppe (Netz_EIN)
0E12	Interner Fehler Aktiv-Automat: Default-Zweig	Neuanlauf der Kommunikationsbaugruppe (Netz_EIN)
0E13	Interner Fehler Passiv-Automat: Default-Zweig	Neuanlauf der Kommunikationsbaugruppe (Netz_EIN)
0E20	Die Anzahl der Datenbits bei dieser Kopplung muss 8 betragen. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E21	Der parametrierte Multiplikationsfaktor für die Zeichenverzugszeit ist nicht im Bereich von 1 bis 10. Der Treiber arbeitet mit der Standardeinstellung von 1.	Korrigieren Sie die Parametrierung des Treibers.
0E22	Die parametrierte Betriebsart des Treibers ist unzulässig. Es ist "Normal"- oder "Störungsunterdrückungs"-Betrieb anzugeben. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E23	Unzulässiger Wert für die Antwortüberwachungszeit parametriert: Zulässige Werte 5 … 65500ms. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E24	Es wurden unzulässige Grenzen für schreibende Zugriffe parametriert. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E25	Es wurde eine unzulässige "von/bis"-Kombination bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 01, 05, 15" parametriert. (Bereich Merker, Ausgänge, Zeiten, Zähler). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0E26	Es wurde eine unzulässige "von/bis"-Kombination bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 02" parametriert. (Bereich Merker, Eingänge). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E27	Es wurde eine Überlappung bei den "von/bis"-Kombinationen bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 01, 05, 15" parametriert. (Bereich Merker, Ausgänge, Zeiten, Zähler). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E28	Es wurde eine Überlappung bei den "von/bis"-Kombinationen bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 02" parametriert. (Bereich Merker, Eingänge). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E2E	Beim Lesen der Schnittstellenparameter-Datei ist ein Fehler aufgetreten. Der Treiber ist nicht betriebsbereit.	Neuanlauf der Kommunikationsbaugruppe (Netz_EIN).
0E30	Interner Fehler Übergabe an CPU: Unerwartete Quittung Passiv.	Kann bei sporadischem Auftreten ignoriert werden.
0E31	Time Out bei Übergabe von Daten an die CPU	Schnittstelle CP-CPU überprüfen.
0E32	Fehler bei Übergabe von Daten mit RCV an die CPU: Ursache analysieren - Detailfehler steht vor diesem Eintrag im SYSTAT.	Schnittstelle CP-CPU überprüfen.
0E33	Interner Fehler bei Datenübergabe an CPU: Unzulässiger Automatenzustand	Schnittstelle CP-CPU überprüfen.
:		
0E38	Beim Zugriff auf einen der SIMATIC-Bereiche "Merker, Ausgang, Zeiten, Zähler, Eingang" mit dem Funktionscode FC 01 oder FC 02 ist ein Fehler aufgetreten: Z. B. Eingang nicht vorhanden, oder Leseversuch über das Bereichsende hinaus.	Prüfen Sie, ob der angesprochene SIMATIC-Bereich vorhanden ist und ob nicht versucht wurde, über das Bereichsende hinaus zuzugreifen.
0E39	Beim Zugriff auf den SIMATIC-Bereich "Datenbaustein" mit dem Funktionscode FC 03, 04, 06, 16 ist ein Fehler aufgetreten: Datenbaustein nicht vorhanden oder zu kurz.	Prüfen Sie, ob der angesprochene Datenbaustein vorhanden und ausreichend lang ist.
0E3A	Bei der Ausführung eines Schreibauftrags mit dem Funktionscode FC 05, 15 ist ein Fehler aufgetreten: Instanz-Datenbaustein der Modbus - Anweisung nicht vorhanden oder zu kurz.	Prüfen Sie, ob der an der Modbus-Kommunikationsanweisung parametrierte Instanz-DB vorhanden und ausreichend lang ist.
0E3B	TimeOut (warten auf Quittung) bei der Ausführung eines Schreibauftrags durch die Modbus-Kommunikations- Anweisung.	Verbindungsprojektierung sowie Schnittstelle CP-CPU überprüfen (BSEND), gegebenenfalls Modbus -Kommunikationsanweisung neu laden.
0E3C	Unzulässiger Auftrag bei diesem Treiber.	Nur SEND, RCV, STATUS (nur bei CP 441-2) zulässig
0E40	Beim Parameter LEN an der Anweisung SEND wurde ein zu kleiner Wert angegeben.	Die Mindestlänge ist 2 Byte.

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0E41	Beim Parameter LEN an der Anweisung SEND wurde ein zu kleiner Wert angegeben. Bei dem übergebenen Funktionscode ist eine größere Länge nötig.	Die Mindestlänge ist bei diesem Funktionscode 6 Byte.
0E42	Der übergebene Funktionscode ist nicht zulässig.	Nur die im Kapitel " Funktionscodes (Seite 1744) " aufgelisteten Funktionscodes sind zulässig.
0E43	Die Slave-Adresse 0 (= Broadcast) ist bei diesem Funktionscode nicht zulässig.	Verwenden Sie Slave-Adresse 0 nur bei den hierfür geeigneten Funktionscodes.
0E44	Der Wert des übergebenen Parameters "Bit-Anzahl" ist nicht im Bereich 1... 2040.	Der Parameter "Bit-Anzahl" muss im Bereich 1 ... 2040 liegen.
0E45	Der Wert des übergebenen Parameters "Register-Anzahl" ist nicht im Bereich 1 ... 127.	Der Parameter "Register-Anzahl" muss im Bereich 1 ... 127 liegen.
0E46	Funktionscode 15 oder 16: Der Wert des übergebenen Parameters "Bit-Anzahl" bzw. "Register-Anzahl" ist nicht im Bereich 1 ... 2040 bzw. 1 ... 127.	Der Parameter "Bit-Anzahl" bzw. "Register-Anzahl" muss im Bereich 1 ... 2040 bzw. 1 ... 127 liegen.
0E47	Funktionscode 15 oder 16: Der Parameter LEN an der Anweisung SEND korrespondiert nicht mit dem übergebenen Parameter "Bit-Anzahl" bzw. "Register-Anzahl". Der Parameter LEN ist zu klein.	Vergrößern Sie den Parameter LEN am SEND, bis eine ausreichende Anzahl von Nutzdaten an die Kommunikationsbaugruppe übergeben wird. Aufgrund von "Bit-Anzahl" bzw. "Register-Anzahl" muss eine größere Anzahl von Nutzdaten an die Kommunikationsbaugruppe übergeben werden.
0E48	Funktionscode 05: Die im SEND-Quell-DB angegebene Codierung für "Bit-Setzen" (FF00H) oder "Bit-Löschen" (0000H) ist falsch.	Es sind nur die Codierungen FF00H und 0000H erlaubt.
0E49	Funktionscode 08: Der im SEND-Quell-DB angegebene "Diagnostic Code" ist falsch.	Es ist nur der "Diagnostic Code" 0000H erlaubt.
0E50	Bei den wortorientierten SIMATIC-Bereichen Zeiten/Zähler ist die sich aus der Modbus-Adresse ergebende Rest-Bitnummer ≠ 0.	Nur Modbus-Adressen verwenden, bei denen sich gültige Bit-Nummern ergeben.
0E51	Die empfangene Modbus-Adresse liegt außerhalb der parametrierten "von/bis"-Bereiche.	Als Adressangabe im Anforderungstelegramm nur Adressen verwenden, die vorher bei der Parametrierung definiert wurden.
0E52	• Überschreitung einer SIMATIC-Bereichsgrenze bei Zugriffsversuch durch das Modbus-Master-System: Resultierende DB-Nr. < 1 • Schreib-Zugriff auf einen nicht freigegebenen Bereich (Parametrierung) • Schreibzugriff auf den Instanz-DB der Kommunikations-Anweisung	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.
0E53	Überschreitung einer SIMATIC-Bereichsgrenze bei Zugriffsversuch durch das Modbus-Master-System. Z. B. Überlauf bei der Bildung der resultierenden DB-Nr. (> 65535)	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.
0E54	Zugriff über das parametrierte Bereichsende hinaus, oder Zugriff über das SIMATIC-Bereichsende hinaus.	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0E55	Ein Schreib-Zugriff auf diesen SIMATIC-Speicherbereich ist nicht erlaubt.	Schreibende Zugriffe nur auf die SIMATIC-Datenbereiche Merker, Ausgänge durchführen.
0E56	Ein Betrieb der Kopplung ist nicht möglich, da die Kommunikationsanweisung nicht läuft.	Rufen Sie die Modbus-Kommunikations- Anweisung im STEP 7-Anwenderprogramm zyklisch auf. Führen Sie gegebenenfalls noch einmal eine Initialisierung der Kommunikations- Anweisung durch.
0E57	Bei der Abarbeitung des Modbus-Funktionscodes ist in der Kommunikations- Anweisung ein Fehler aufgetreten.	Analysieren Sie die genaue Ursache.
0E4A	Die Länge für diesen Funktionscode ist größer als die maximale Länge.	Im Handbuch finden Sie die maximalen Längenangaben zu jedem Funktionscode.
0E4F	CP 441: Die an der Anweisung SEND angegebene R_ID ist bei diesem Treiber unzulässig.	Nur R_IDs 0 ... 255 (00 00 00 00 ... 00 00 00 FFH) verwenden.
	CP 341: Die an der Anweisung SEND_RK angegebene R_TYP ist bei diesem Treiber unzulässig.	Als R_TYP muss "X" eingetragen werden.
0E50	Slave-Adresse falsch: Die empfangene Slave-Adresse ist ungleich der gesendeten Slave-Adresse.	Ein falscher Slave hat geantwortet. Prüfen Sie, ob die Übertragungsleitung gestört ist (gegebenenfalls Schnittstellen-Analyzer verwenden).
0E51	Funktionscode falsch: Der im Antworttelegramm empfangene Funktionscode ist ungleich dem gesendeten Funktionscode.	Überprüfen Sie das Slave-Gerät.
0E52	Byte-Underflow: Es wurden weniger Zeichen empfangen, als sich durch den Bytezähler des Antworttelegramms ergeben würden, bzw. als bei diesem Funktionscode erwartet werden.	Überprüfen Sie das Slave-Gerät.
0E53	Byte-Overflow: Es wurden mehr Zeichen empfangen, als sich durch den Bytezähler des Antworttelegramms ergeben würden, bzw. als bei diesem Funktionscode erwartet werden.	Überprüfen Sie das Slave-Gerät.
0E54	Bytezähler zu klein: Der im Antworttelegramm empfangene Bytezähler ist zu klein.	Überprüfen Sie das Slave-Gerät.
0E55	Bytezähler falsch: Der im Antworttelegramm empfangene Bytezähler ist falsch.	Überprüfen Sie das Slave-Gerät.
0E56	Echo falsch: Die vom Slave geechoten Daten des Antworttelegramms (Bit-Anzahl, ...) sind ungleich den im Anforderungstelegramm gesendeten Daten.	Überprüfen Sie das Slave-Gerät.
0E57	CRC-Check fehlerhaft: Bei der Überprüfung der CRC-16-Checksumme des Antworttelegramms vom Slave wurde ein Fehler festgestellt.	Überprüfen Sie das Slave-Gerät.
0E61	Antworttelegramm mit Exception Code 01: Illegal function	Siehe "Handbuch des Slave-Geräts"
0E62	Antworttelegramm mit Exception Code 02: Illegal data address	Siehe "Handbuch des Slave-Geräts"

Fehler-code(W#16 #...)	Beschreibung	Abhilfe
0E63	Antworttelegramm mit Exception Code 03: Illegal data value	Siehe "Handbuch des Slave-Geräts"
0E64	Antworttelegramm mit Exception Code 04: Failure in associated device	Siehe "Handbuch des Slave-Geräts"
0E65	Antworttelegramm mit Exception Code 05: Acknowledge	Siehe "Handbuch des Slave-Geräts"
0E66	Antworttelegramm mit Exception Code 06: Busy, Rejected message	Siehe "Handbuch des Slave-Geräts"
0E67	Antworttelegramm mit Exception Code 07: Negative Acknowledgement	Siehe "Handbuch des Slave-Geräts"
1E0D	"Auftragsabbruch wegen Neustart, Wiederanlauf oder Reset"	
1E0E	Statischer Fehler bei Aufruf der Anweisung DP_RDDAT. Der Returnwert RET_VAL der Anweisung wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E0F	Statischer Fehler bei Aufruf der Anweisung DP_WRDAT. Der Returnwert RET_VAL der Anweisung wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E10	Statischer Fehler bei Aufruf der Anweisung RD_LGADR. Der Returnwert RET_VAL der Anweisung wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E11	Statischer Fehler bei Aufruf der Anweisung RDSYSST. Der Returnwert RET_VAL der Anweisung wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E20	Parameter außerhalb des Bereichs.	Geben Sie für den Funktionsbaustein einen Parameter im zulässigen Bereich ein.
1E41	Anzahl der am Parameter LEN der Anweisung angegebenen Bytes unzulässig.	Halten Sie den Wertebereich von 1 bis 256 Byte ein.

USS-Kommunikation

Überblick über USS-Kommunikation

Einordnen in die Systemumgebung

Die nachfolgende USS Beschreibung bezieht sich auf den Betrieb des ET 200S 1SI Moduls in der Modbus/USS - Variante.

Einleitung

Das USS-Protokoll ist ein einfaches serielles Datenübertragungsprotokoll, das auf die Anforderungen in der Antriebstechnologie zugeschnitten ist.

Das USS-Protokoll definiert eine Zugriffstechnik entsprechend dem Master-Slave-Prinzip für die Kommunikation über einen seriellen Bus. An den Bus können ein Master und bis zu 31 Slaves angeschlossen werden. Master und Slaves bilden ein USS-Netzwerk. Die zwischen Master und Slaves ausgetauschten Daten werden als Netzdaten bezeichnet. Der Netzdatenbaustein enthält diese Daten für den Master. Die einzelnen Slaves werden vom Master mit Hilfe eines Adresszeichens im Telegramm gewählt. Ein Slave kann nie senden, ohne zuvor vom Master angestoßen worden zu sein. Es ist also keine direkte Datenübertragung zwischen einzelnen Slaves möglich. Die Kommunikation arbeitet im Halbduplexbetrieb. Die Master-Funktion kann nicht übertragen werden. Das USS-System besitzt nur einen Master.

Telegrammaufbau

Jedes Telegramm beginnt mit dem Startzeichen (STX), gefolgt von der Längenangabe (LGE) und dem Adressbyte (ADR). Danach folgt das Datenfeld. Das Telegramm endet mit dem Blockprüfzeichen (BCC).

STX	LGE	ADR	1	2	...	N	BCC

Bei Daten im Netzdatenbaustein, die aus einem Wort (16 Bit) bestehen, wird zuerst das High-Byte, gefolgt vom Low-Byte gesendet. Bei Doppelwortdaten wird dementsprechend zuerst das High-Word und anschließend das Low-Word gesendet.

Das Protokoll identifiziert keine Tasks in den Datenfeldern.

Datenverschlüsselung

Die Daten werden folgendermaßen verschlüsselt:

- STX: 1 Byte, Textbeginn, 02H
- LGE: 1 Byte, enthält die Telegrammlänge als Binärzahl
- ADR: 1 Byte, enthält die Slave-Adresse und den Telegrammtyp, binärcodiert
- Datenfelder: Je ein Byte, Inhalt sind taskabhängig
- BCC: 1 Byte, Blockprüfzeichen

Datenübertragungsprozedur

Der Master sorgt für die zyklische Datenübertragung in Telegrammen. Der Master adressiert alle Slave-Teilnehmer nacheinander mit einem Task-Telegramm. Die angesprochenen Teilnehmer antworten mit einem Antworttelegramm. Entsprechend der Master-Slave-Prozedur muss der Slave das Antworttelegramm an den Master senden, nachdem er das Task-Telegramm empfangen hat. Erst dann kann der Master den nächsten Slave ansprechen.

Allgemeiner Aufbau des Netzdatenbausteins

Der Netzdatenbaustein ist in zwei Bereiche unterteilt: Parameter (PKW) und Prozessdaten (PZD).

STX	LGE	ADR	Parameter (PKW)	Prozessdaten (PZD)	BCC

- **Parameterbereich** (PKW)
  Der PKW-Bereich wickelt die Parameterübertragung zwischen zwei Kommunikationspartnern ab (z.B. Steuerung und Antrieb). Dies umfasst beispielsweise das Lesen und Schreiben von Parameterwerten und das Lesen von Parameterbeschreibungen und zugehörigem Text. Die PKW-Schnittstelle beinhaltet im allgemeinen Tasks für Bedienung und Anzeigen, Wartung und Diagnose.
- **Prozessdatenbereich** (PZD)
  Der PZD-Bereich umfasst Signale, die für die Automatisierung erforderlich sind:
  - Steuerworte und Sollwerte vom Master an den Slave
  - Statusworte und Istwerte vom Slave an den Master

  Die Inhalte des Parameterbereichs und des Prozessdatenbereichs werden von den Slave-Antrieben definiert. Weitere Informationen hierzu finden Sie in der Dokumentation des Antriebs.

USS Master

Funktionsübersicht

Ablauf der Netzdatenübertragung

Die Anweisungen wickeln die Netzdatenübertragung zyklisch mit bis zu 31 Antriebs-Slaves entsprechend der in der Polling-Liste (Parametrierungs-DB) angegebenen Reihenfolge ab. Es ist pro Slave jeweils nur ein Auftrag aktiv. Die Netzdaten für jeden Slave werden vom Anwender in einem Datenbaustein (Netzdatenbaustein) abgelegt und von dort abgerufen. Sie werden entsprechend der Programmdefinition in der Polling-Liste über einen anderen Datenbereich (Kommunikationsprozessor-DB) zum Kommunikationsprozessor übertragen und von dort abgerufen.

Zwei Funktionsaufrufe sind für diese Prozedur erforderlich (eine Sende- und eine Empfangsanweisung). Eine weitere Funktion unterstützt das Erzeugen und Vorbesetzen der für die Kommunikation erforderlichen Datenbausteine.

Leistungsmerkmale:

- Erstellung von Datenbereichen für die Kommunikation je nach Buskonfiguration
- Vorbesetzung der Polling-Liste
- Telegrammaufbau entsprechend der USS-Spezifikation
- Netzdatenaustausch kann entsprechend der erforderlichen Netzdatenstruktur parametriert werden
- Ausführung und Überwachung von PKW-Aufträgen

- Abwicklung von Parameteränderungsberichten
- Überwachung des Gesamtsystems und Fehlerbehebung

Verschiedene Netzdatenstrukturen können genutzt werden, um Netzdaten zu senden. D. h. der Inhalt der Netzdaten kann je nach Einsatzfall gewählt werden.

Je nach der gewählten Struktur verfügen die Netzdaten über einen PZD-Bereich für die Prozessdaten und einen PKW-Bereich für die Parameterverarbeitung.

Im PKW-Bereich kann der Master Parameterwerte lesen und schreiben und der Slave kann Parameteränderungen über Parameteränderungsberichte anzeigen.

Der PZD-Bereich enthält für die Prozesssteuerung erforderliche Signale, z.B. Steuerworte und Sollwert vom Master an den Slave und Statusworte und Istwerte vom Slave an den Master.

Die ordnungsgemäße Reihenfolge von Anweisungsaufrufen ist: S_USST, S_SEND, S_RCV, S_USSR. Dies ist wichtig, weil die Ausgänge der Anweisungen S_SEND und S_RCV nur im aktuellen Zyklus des Automatisierungssystems gültig sind.

Das folgende Bild zeigt den Datenverkehr zwischen dem Anwenderprogramm und dem USS Slave.

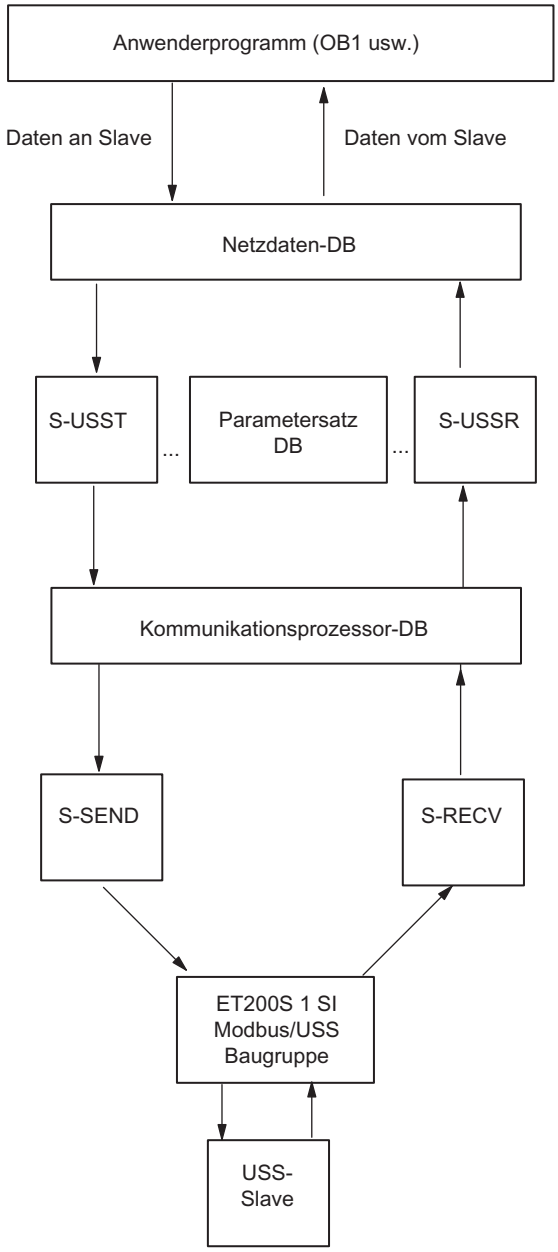

Bild 1-71 Datenverkehr zwischen dem Anwenderprogramm und dem USS Slave

Weitere Informationen

Informationen zu den verwendeten Datenbausteinen finden Sie hier:

- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT

Anlauf & Diagnose

Anlaufverhalten

Anlaufverhalten

Der Anlauf der Kommunikationsbaugruppe unterteilt sich in folgende Phasen:

- Initialisierung (NETZ-EIN der Kommunikationsbaugruppe)
- Parametrierung

Initialisierung

Sobald die Kommunikationsbaugruppe an Spannung liegt, wird nach Durchlauf eines Hardwaretestprogramms die Firmware auf der Kommunikationsbaugruppe für den Betrieb vorbereitet.

Parametrierung

Bei der Parametrierung empfängt die Kommunikationsbaugruppe die dem aktuellen Steckplatz zugeordneten Baugruppenparameter. Die Kommunikationsbaugruppe ist jetzt betriebsbereit.

Betriebszustandsübergänge

Nach dem Anlauf der Kommunikationsbaugruppe werden alle Daten zwischen CPU und der Kommunikationsbaugruppe über Anweisungen ausgetauscht. Das Betriebszustandsübergangsverhalten der Kommunikationsbaugruppe hängt dabei vom Betriebszustand der CPU ab.

CPU-STOP	Eine laufende Datenübertragung Kommunikationsbaugruppe -CPU, sowohl Sende- als auch Empfangsauftrag, wird abgebrochen und ein Neustart der Verbindung eingeleitet.
CPU-RUN	Im Zustand RUN der CPU ist ein uneingeschränkter Sende– und Empfangsbetrieb möglich. In den ersten Anweisungs-Durchläufen nach CPU-Neustart werden die Kommunikationsbaugruppe und die jeweilige Anweisung synchronisiert.
Neustart der CPU	Bei Neustart der CPU werden die Aufträge auf der CPU zurückgesetzt, d. h. alle laufenden Aufträge zwischen CPU und Kommunikationsbaugruppe werden automatisch abgebrochen.
Wiederanlauf der CPU	Bei Wiederanlauf der CPU werden die Aufträge weiterbearbeitet. Durch entsprechende Parametrierung im Eigenschaften Dialog der Baugruppe können Sie den Empfangspuffer auf der Kommunikationsbaugruppe im CPU-Anlauf automatisch löschen.

Diagnosefunktionen

Einleitung

Die Diagnosefunktionen der Kommunikationsbaugruppe erlauben Ihnen eine schnelle Lokalisierung aufgetretener Fehler. Folgende Diagnosemöglichkeiten stehen Ihnen zur Verfügung:

Diagnose über die Anzeigeelemente der Kommunikationsbaugruppe	Die Anzeigeelemente informieren Sie über den Betriebszustand bzw. über mögliche Fehlerzustände der Kommunikationsbaugruppe. Die Anzeigeelemente geben Ihnen einen ersten Überblick über aufgetretene interne bzw. externe Fehler sowie schnittstellenspezifische Fehler.
Diagnose über den STATUS-Ausgang der Anweisungen	Für eine Fehlerdiagnose besitzen die Anweisungen einen STATUS-Ausgang, die Ihnen Aussagen zu Kommunikationsfehlern zwischen Kommunikationsbaugruppe und CPU liefern. Den STATUS-Parameter können Sie im Anwenderprogramm auswerten.
Diagnose über den Fehlermeldebereich SYSTAT [1]	Über die Programmierung der Anweisung STATUS im Anwenderprogramm können Sie den Status einer Schnittstelle abfragen. Durch Lesen des SYSTAT erhalten Sie detaillierte Aussagen zu Fehlern/Ereignissen, die bei der Kommunikation zwischen der Kommunikationsbaugruppe und der zugeordneten CPU und dem angeschlossenen Kommunikationspartner auf dieser Schnittstelle aufgetreten sind.
Diagnose über die Fehlernummern im Reaktionstelegramm	Wenn Sie mit der Rechnerkopplung RK512 arbeiten und bei einem SEND- oder FETCH-Telegramm beim Kommunikationspartner ein Fehler auftritt, sendet der Kommunikationspartner ein Reaktionstelegramm mit einer Fehlernummer im 4. Byte.
Diagnose über den Diagnosepuffer der Kommunikationsbaugruppe	Alle Fehler/Ereignisse werden auch in den Diagnosepuffer [2] der Kommunikationsbaugruppe eingetragen. Wie beim CPU-Diagnosepuffer können Sie sich auch die anwenderrelevanten Informationen des CP-Diagnosepuffers im Klartext anzeigen lassen.
Diagnosealarm	Die Kommunikationsbaugruppe kann auf der ihr zugeordneten CPU einen Diagnosealarm auslösen. Die Kommunikationsbaugruppe stellt 4 Bytes Diagnoseinformation zur Verfügung. Die Auswertung dieser Informationen erfolgt über das Anwenderprogramm oder durch Auslesen des CPU-Diagnosepuffers.

[1] nur CP 441-1 und CP 441-2

[2] CP 340, ET 200S 1SI Module und CPU31xC-2 PtP haben keinen Diagnosepuffer

1.4.7 TeleControl

1.4.7.1 Telecontrol-Module konfigurieren

Projektierung

Kommunikationsarten

In dieser Parametergruppe aktivieren Sie die Kommunikationsarten, die Sie für die jeweilige Baugruppe verwenden möchten.

Um das Risiko unerlaubter Zugriffe auf die Station zu minimieren, müssen Sie die Kommunikationsdienste, welche Sie mit der Baugruppe nutzen möchten, einzeln aktivieren.

Sie können alle Optionen deaktiviert lassen, wenn Sie sie nicht benötigen, beispielsweise wenn Sie einen CP nur zum Versenden von SMS oder für die Open User Communication verwenden.

Parametergruppe "Kommunikationsarten"

- **Telecontrol-Kommunikation aktivieren**
  Gibt in der Baugruppe die Telecontrol-Kommunikation frei.

  Hinweis

  Volle Telecontrol-Funktionalität nur bei aktivierten Security-Funktionen

  Für folgende Funktionen müssen Sie die Security-Funktionen aktivieren:
  - Versenden von E-Mails über die Telecontrol-Funktionalität
  - Nutzung des Protokolls "TeleControl Basic" (generell)
  - Nutzung der DNP3-Security-Funktionen
  - Nutzung von Zertifikaten

  Hinweis

  Verlust von Projektierungsdaten bei Wechsel des Telecontrol-Protokolls

  Wenn Sie bei einer projektierten Baugruppe das Protokoll wechseln, gehen protokollspezifische Projektierungsdaten verloren. Dies sind unter anderem:
  - Partnerprojektierung
  - Datenpunktprojektierung
  - Projektierte Nachrichten (E-Mails, SMS)
  - Projektierte Uhrzeitsynchronisation
  - Projektierte VPN-Partner

  Abhängig vom Baugruppentyp können folgende Protokolle eingesetzt werden:

  – TeleControl Basic *
    Aktiviert die Kommunikation mit dem Telecontrol-Server.

  – ST7
    Aktiviert die Kommunikation mit einem Leitsystem, welches das ST7-Protokoll unterstützt.

  – DNP3 *
    Aktiviert die Kommunikation mit bis zu vier DNP3-Mastern.

  – IEC 60870-5-104 *
    Aktiviert die Kommunikation mit bis zu vier IEC-Mastern.

  * Zur Telecontrol-Kommunikation über SINEMA Remote Connect siehe nächster Abschnitt.
  Zur Projektierung der Telecontrol-Verbindungen für die Protokolle ST7, DNP3 und IEC siehe Kapitel AUTOHOTSPOT.
  Die Beschreibung wichtiger Parameter und Funktionen der Telecontrol-Protokolle finden Sie im Kapitel AUTOHOTSPOT.

- **Online-Funktionen aktivieren**
  Gibt in der Baugruppe den Zugang zur CPU für die Online-Funktionen frei (Diagnose, Projektdaten laden etc.). Bei aktivierter Funktion kann von der Engineering-Station über die Baugruppe auf die CPU zugegriffen werden.
  Wenn die Option deaktiviert ist, dann haben Sie mit den Online-Funktionen über die Baugruppe keinen Zugriff auf die CPU. Die Online-Diagnose der CPU mit direktem Anschluss an die Schnittstelle der CPU ist jedoch weiterhin möglich.

- **S7-Kommunikation aktivieren**

Gibt die Funktionen der S7-Kommunikation mit der zugeordneten SIMATIC-CPu und das S7-Routing frei.
Wenn Sie S7-Verbindungen mit der betreffenden Station projektieren, die über die Baugruppe laufen, dann müssen Sie diese Option aktivieren.
S7-Routing wird von folgenden Modulen unterstützt:

– CP 1243-1, CP 124x-7, CP 1243-8
  Ab CP-Firmware V2.1 mit CPU ≥ V4.2
– CP 1542SP-1, CP 1542SP-1 IRC, CP 1543SP-1
  Ab CP-Firmware V1.0 mit CPU ≥ V2.0

Fernwartung mit SINEMA Remote Connect (SINEMA RC)

Die Applikation "SINEMA Remote Connect" (SINEMA RC) steht für Fernwartungszwecke zur Verfügung.

SINEMA RC verwendet OpenVPN zur Verschlüsselung der Daten. Zentrum der Kommunikation ist der SINEMA RC-Server, über den die Kommunikation zwischen den Teilnehmern läuft und der die Konfiguration des Kommunikationssystems verwaltet.

Einsetzbare Module und Telecontrol-Protokolle

Folgende CPs unterstützen Kommunikation über SINEMA RC:

- CP 1243-1, CP 1243-7, CP 1243-8
  Ab Firmware V3.1
- CP 1542SP-1 IRC, CP 1543SP-1
  Ab Firmware V2.0

VPN-fähige Telecontrol-CPs können auch die Telecontrol-Kommunikation über den SINEMA RC-Server abwickeln. Folgende Telecontrol-Protokolle unterstützen Kommunikation über SINEMA RC:

- TeleControl Basic
- DNP3
- IEC 60870-5-104

Anwendungen und Projektierung

Für die Nutzung von SINEMA Remote Connect und die Telecontrol-Kommunikation über SINEMA RC nehmen Sie folgende Einstellungen vor:

1. Kommunikation über SINEMA RC: Parametergruppe "Security > VPN"
   Aktivieren Sie VPN, damit die weiteren Parameter sichtbar werden.
2. Telecontrol-Kommunikation über SINEMA RC: Parametergruppe "Kommunikationsarten"
   Die Kommunikation über SINEMA RC muss zuvor aktiviert sein.

Aus der Kombination von Telecontrol-Kommunikation und der Nutzung von SINEMA RC ergeben sich folgende Anwendungsfälle:

- (1) Keine Telecontrol-Kommunikation und kein SINEMA RC - CP nur für Netzwerktrennung.
- (2) CP nur für Fernwartung über SINEMA RC
- (3) CP nur für Telecontrol-Kommunikation

- (4) CP nutzt Telecontrol-Kommunikation, SINEMA RC aber nur für Fernwartung.
- (5) CP nutzt SINEMA RC für Telecontrol-Kommunikation und Fernwartung.

Die Tabelle gibt einen Überblick über die Anwendungsfälle mit den Parameter-Einstellungen.

Tabelle 1-123 Anwendungsfälle und zu aktivierende Parameter

Anwendungsfall	Parameter-Einstellungen * (Parameter abgekürzt)		
	SRC	TC	TC + SRC
(1)	Aus	Aus	Aus
(2)	Ein	Aus	Aus
(3)	Aus	Ein	Aus
(4)	Ein	Ein	Aus
(5)	Ein	Ein	Ein

***** Erläuterung der Parameter-Abkürzungen:

SRC - Security > VPN (aktiviert) > "VPN-Verbindungstyp":
"Automatische OpenVPN-Konfiguration über SINEMA Remote Connect Server"

TC - Kommunikationsarten > Telecontrol-Kommunikation aktiviert

TC + SRC - Kommunikationsarten >
"Telecontrol-Kommunikation über SINEMA Remote Connect aktivieren"

Telecontrol-Kommunikation über SINEMA Remote Connect

Projektierungsschritte

Gehen Sie bei der Projektierung des CP für die Nutzung der Telecontrol-Kommunikation über SINEMA RC folgendermaßen vor:

1. Aktivieren Sie in der Parametergruppe "Kommunikationsarten" die Telecontrol-Kommunikation und wählen Sie eines der oben genannten Protokolle aus.
   Die Option zur Kommunikation über SINEMA RC ist noch nicht sichtbar.
2. Wechseln Sie in die Parametergruppe "Security" und aktivieren Sie die Security-Funktionen.
   (In der Parametergruppe "Kommunikationsarten" erscheint die SINEMA RC-Option deaktiviert und gegraut.)
3. Öffnen Sie die Parametergruppe "Security > VPN" und aktivieren Sie VPN.
4. Wählen Sie beim Parameter "VPN-Verbindungstyp" die Option "Automatische OpenVPN-Konfiguration über SINEMA Remote Connect Server" aus, wenn diese noch nicht vorbelegt ist.
   (In der Parametergruppe "Kommunikationsarten" wird die SINEMA RC-Option bedienbar.)
5. Wechseln Sie in die Parametergruppe "Kommunikationsarten" und setzen Sie das Häkchen an der Option "Telecontrol-Kommunikation über SINEMA Remote Connect aktivieren".
   Der CP ist jetzt vorbereitet für die Telecontrol-Kommunikation über SINEMA Remote Connect.

6. Nehmen Sie die weitere Projektierung des SINEMA RC-Servers unter "Security > VPN" vor.
7. Importieren Sie das Zertifikat des SINEMA RC-Servers unter "Security > Zertifikatsmanager".

Projektierung des Verbindungsaufbaus

Der Verbindungsaufbau wird in der Parametergruppe "Security > VPN > Optionale Einstellungen" über den Parameter "Verbindungsart" projektiert.

- **Aktualisierungsintervall**
  Über den Parameter stellen Sie das Intervall ein, in dem der CP die Konfiguration beim SINEMA RC-Server abfragt.
  Beachten Sie bei der Einstellung 0 (null), dass Änderungen der Konfiguration des SINEMA RC-Servers dazu führen können, dass vom CP keine Verbindung mehr zum SINEMA RC-Server aufgebaut werden kann.
- **"Verbindungsart"**
  Die beiden Optionen des Parameters haben folgende Auswirkung:
  - Permanent
    Die Option ist vorgesehen für dauerhafte Kommunikation.
    Dies ist beispielsweise der Fall bei Telecontrol-Kommunikation über SINEMA RC. Der CP und der Kommunikationspartner des CP müssen im SINEMA RC-Server projektiert werden.
  - PLC-Trigger
    Die Option ist vorgesehen für sporadische Kommunikation des CP über den SINEMA RC-Server.
    Diese Option können Sie nutzen, wenn Sie sporadisch temporäre Verbindungen zwischen dem CP und der Engineering-Station (PC mit dem STEP 7-Projekt des CP) aufbauen möchten, beispielsweise für Service-Fälle.
- **PLC-Variable für Verbindungsaufbau**
  Der CP baut bei ausgewählter Option "PLC-Trigger" eine Verbindung auf, wenn die PLC-Variable den Wert 1 annimmt. Im laufenden Betrieb kann die PLC-Variable bei Bedarf gesetzt werden, beispielsweise über ein HMI-Panel.
  Beim Rücksetzen der PLC-Variable auf 0 wird die Verbindung wieder abgebaut.

Grundeinstellungen

Projektierung

Die Parametergruppe steht nur bei einigen Kommunikationsmodulen des Systems SINAUT ST7 zur Verfügung.

Zu den unterstützten Modulen siehe Kapitel Erweiterung von SINAUT-Projekten (Seite 1900).

Telecontrol-Konfiguration

Hier legen Sie für das Kommunikationsmodul das Verfahren der Projektierung der Teilnehmer- und SINAUT-Verbindungsdaten fest.

- Projektieren
  Die genannten Projektierungsdaten werden in den Parametergruppen des Moduls und im Verbindungseditor projektiert.
- Importieren
  Die genannten Projektierungsdaten werden für eine Proxy-Baugruppe in STEP 7 V5 projektiert.
  Die STEP 7 V5-Projektierungsdaten der Proxy-Baugruppe werden für das Modul in STEP 7 Professional importiert.
  Zum Importieren und zum Umfang der importierten Daten siehe Kapitel Partnerprojektierung importieren (Protokoll ST7) (Seite 1810).

Uhrzeitsynchronisation

Uhrzeitsynchronisation bei Telecontrol-CPs

Hinweis

Empfehlung: Uhrzeitsynchronisation nur durch 1 Modul

Lassen Sie die Uhrzeit der Station von einer externen Uhrzeitquelle nur durch ein einziges Modul der Station synchronisieren, um innerhalb der Station eine konsistente Uhrzeit vorzuhalten.

Wenn die CPU die Uhrzeit vom CP übernimmt, dann deaktivieren Sie die Uhrzeitsynchronisation der CPU.

Module und Synchronisationsverfahren

Bei Verwendung einer externen Uhrzeitquelle kann die S7-Station die aktuelle Uhrzeit sowohl über die CPU als auch über einen Telecontrol-CP beziehen.

Eine Weiterleitung der Uhrzeit von der Station an das Subnetz findet bei S7-1200 / ET 200SP nicht statt.

- **Uhrzeitsynchronisation der CPU**
  Als Synchronisationsverfahren kann bei der CPU nur NTP ausgewählt werden.
  Zu den Möglichkeiten zum Stellen der CPU-Uhrzeit über einen Telecontrol-CP siehe unten.
- **Uhrzeitsynchronisation von Telecontrol-CPs**
  Folgende Synchronisationsverfahren stehen für die CPs zur Auswahl:
  - **NTP**
    NTP kann auch verwendet werden, wenn die Telecontrol-Kommunikation deaktiviert ist. In diesem Fall wird der Telecontrol-CP als erweiterte PROFINET-Schnittstelle der CPU eingesetzt.
    Die Adresse des NTP-Servers kann auch als URL eingegeben werden, bspw. <ntp.server.com>.
    Bei aktivierten Security-Funktionen kann auch das Verfahren "NTP (secure)" verwendet werden.
  - **Uhrzeit vom Partner**
    Wenn die Telecontrol-Kommunikation aktiviert ist, übernimmt der CP die Uhrzeit automatisch vom Kommunikationspartner.
  - **Uhrzeit von der CPU**
    Alle CMs/CPs der Station werden durch die Uhrzeit der CPU synchronisiert.

Uhrzeitweiterleitung vom CP an die CPU (S7-1200)

Hinweis

Uhrzeitweiterleitung an die CPU

Abhängig von der Firmware-Version der beteiligten Module wird die Uhrzeit des CP unterschiedlich an die CPU weitergeleitet:

- Optionale Weiterleitung der CP-Zeit an die CPU über eine PLC-Variable
- Obligatorische Weiterleitung der CP-Zeit über den Rückwandbus an die CPU

Die Weiterleitung der CP-Zeit an die CPU ist abhängig von der Firmware-Version des CP und der CPU. Beachten Sie das nachfolgende Verhalten.

- **CP-Firmware ≤ V2.1.5**
  Mit dieser Firmware-Version kann die CP-Uhrzeit optional der CPU über eine PLC-Variable zur Verfügung gestellt werden. Wenn diese PLC-Variable zyklisch von der CPU gelesen wird, übernimmt die CPU die CP-Zeit.
  Zur PLC-Variable siehe Parametergruppe "Kommunikation mit der CPU" des CP.
- **CP-Firmware ≥ V2.1.7 und CPU-Firmware ≥ V4.2**
  Wenn beide Module in einer Station eine der genannten Firmware-Versionen aufweisen, wird die Uhrzeit des CP automatisch an die CPU weitergleitet. Da die CPU automatisch die CP-Zeit übernimmt, benötigen Sie die Weiterleitungsoption über die PLC-Variable nicht mehr.
  Wenn bei der CPU unter "PROFINET-Schnittstelle > Uhrzeitsynchronisation" die Option "CPU synchronisiert die Module des Geräts" aktiviert ist, werden alle intelligenten Module der Station mit der CPU-Zeit synchronisiert.

Uhrzeitsynchronisation über NTP

NTP-Verfahren

Beim NTP-Verfahren sendet die Baugruppe in regelmäßigen Zeitabständen Uhrzeitanfragen an einen oder mehreren NTP-Server. Aus den Antworten der Server wählt das Modul die genaueste Uhrzeit aus.

Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

NTP verwendet generell UTC (Universal Time Coordinated). Dies entspricht GMT (Greenwich Mean Time).

NTP-Server und Synchronisationsintervall

Die IP-Adresse mindestens eines NTP-Servers muss projektiert werden.

Das Synchronisationsintervall legt den Zyklus der Uhrzeitanfragen am NTP-Server fest. Abhängig vom ausgewählten Modul kann der Zyklus aus einer Klappliste ausgewählt oder innerhalb eines Wertebereichs zwischen 10 Sekunden und 1 Tag (86400 Sekunden) projektiert werden.

Option "Uhrzeitsynchronisation von nicht synchronisierten NTP-Servern annehmen"

Wenn die Option vom ausgewählten Modul unterstützt wird, gilt das folgende Verhalten:

- Aktiviert
  Das Modul nimmt die Uhrzeit auch von nicht synchronisierten NTP-Servern mit Stratum ≥ 15 an.
- Deaktiviert
  Wenn das Modul ein Uhrzeit-Telegramm von einem nicht synchronisiertem NTP-Server mit Stratum ≥ 15 empfängt, wird die Uhrzeit nicht danach gestellt.

Ethernet-Schnittstelle

Verbindungsaufbau-Verzögerung

Hinweis

Wenn der Partner nicht erreichbar ist, dann kann ein Verbindungsaufbau über das Mobilfunknetz mehrere Minuten dauern. Dies kann vom jeweiligen Netz und der aktuellen Netzlast abhängen.

Vertragsabhängig können pro Verbindungsaufbauversuch Kosten entstehen.

Parameter "Verbindungsaufbau-Verzögerung"

Die Verbindungsaufbau-Verzögerung ist für Verbindungen der Telecontrol-Kommunikation die Wartezeit zwischen wiederholten Verbindungsaufbauversuchen des CP, wenn der Telecontrol-Server nicht erreichbar ist oder die Verbindung abgebrochen ist. Diese Wartezeit dient dazu, bei Verbindungsproblemen nicht andauernd in kurzen Abständen Verbindungsaufbauversuche zu starten.

Projektiert wird ein Basiswert für die Wartezeit bis zum nächsten Verbindungsaufbauversuch. Angefangen beim Basiswert wird die aktuelle Wartezeit nach jeweils 3 erfolglosen Versuchen verdoppelt bis zu einem Maximalwert von 900 s. Wertebereich für den Basiswert: 10 .. 300 s

Beispiel: Der Basiswert 20 ergibt folgende Intervalle (Wartezeiten) zwischen den Versuchen, eine Verbindung wieder aufzubauen:

- dreimal 20 s
- dreimal 40 s
- dreimal 80 s
- etc. bis max. 900 s

Hinweis

Besonderheit beim CP 1242-7 (6GK7242-7KX30-0XE0) zusammen mit TELECONTROL SERVER BASIC (V2)

Wenn ein zweiter Telecontrol-Server oder ein zweiter Router des Telecontrol-Servers projektiert ist, dann versucht der CP beim 4. Mal, sich mit dem zweiten Partner zu verbinden. Wenn der zweite Partner auch nicht erreichbar ist, dann versucht der CP beim 7. Mal, sich wieder mit dem ersten Partner zu verbinden und so weiter.

Serielle Schnittstelle

Aktivierung der seriellen Schnittstelle des CP 1243-8 IRC (V3) für das Protokoll ST7

Serielle Schnittstelle des CP 1243-8 IRC

Wenn Sie den CP aus dem Katalog in das Projekt einfügen, ist die serielle Schnittstelle des CP in der Netzsicht von STEP 7 nicht sichtbar.

Bei Nutzung des Kommunikationsarten "DNP3" und "IEC" ist die serielle Schnittstelle nicht nutzbar.

Nach Aktivierung der Kommunikationsart "ST7" wird die serielle Schnittstelle sichtbar, ist aber nicht projektierbar. Für die Nutzung der seriellen Schnittstelle des CP ist ein TS-Modul erforderlich.

Gehen Sie folgendermaßen vor, um das TS-Modul einzufügen und dadurch die serielle Schnittstelle für die ST7-Kommunikation zu aktivieren:

1. Öffnen Sie aus der Netzsicht die Gerätesicht des CP (Doppelklick auf die Baugruppe).
   Die Gerätesicht öffnet sich.
   Gleichzeitig ändert sich die Katalogansicht. Unter anderem erscheint das Verzeichnis "Kommunikationsmodule".
2. Klappen Sie das Verzeichnis "Kommunikationsmodule" auf:
   > Industrial Remote Communication > TS Modules
3. Klappen Sie das Unterverzeichnis des erforderlichen TS-Moduls auf.
4. Selektieren Sie das TS-Modul.
   Auf dem Symbol des CP in der Gerätesicht erscheint ein senkrechter blauer Strich.
5. Ziehen Sie nun das TS-Modul auf den blauen Strich des CP in der Gerätesicht.
6. Das TS-Modul mit einer seriellen Schnittstelle wird in der Station eingefügt.
   Gleichzeitig wird in der Netzsicht eine neue WAN-Schnittstelle beim CP eingefügt.

Sie können die seriellen Schnittstelle direkt in der Gerätesicht beim TS-Adapter oder unter der WAN-Schnittstelle des CP in der Netzsicht projektieren.

Parameter von WAN-Netzen (Standleitung / Wählnetz)

Schnittstellen-Betriebsmodus

Der Betriebsmodus der seriellen Schnittstelle gibt den Modus des Datentransfers an.

Maximal eine serielle Schnittstelle eines Kommunikationsmoduls darf im DMA-Modus betrieben werden.

Wertebereich:

- Interrupt (Block)
  Dieser Betriebsmodus gilt für den Sende- und Empfangs-Betrieb. Der Standardmodus Interrupt (Block) ist prinzipiell für alle Verbindungen geeignet. Pro Block werden 4 Zeichen übertragen. Danach erfolgt ein Interrupt. Die empfangenen Zeichen werden erst nach dem Eintreffen eines kompletten Daten-Telegramms geprüft.
- DMA
  Dieser Betriebsmodus gilt für den Sende- und Empfangs-Betrieb. Der DMA-Modus sollte für Verbindungen mit hoher Baudrate bzw. hohem Datenverkehr genutzt werden, jedoch nicht für GSM-Netze.
- Interrupt (Einzelzeichen)
  Dieser Betriebsmodus wird nur in Empfangs-Richtung verwendet. In Sende-Richtung wird weiterhin mit dem Block-Modus gearbeitet. Der Interrupt-Modus ist für sehr schlechte Leitungen geeignet. Pro übertragenem Zeichen wird ein Interrupt ausgelöst und jedes Zeichen sofort nach dem Erhalt analysiert, wodurch Übertragungsfehler sehr gut diagnostiziert werden können. Diese Betriebsart ist sicherer als der Blockmodus, aber auch langsamer.

Übertragungskriterium

Nur für Schnittstellen mit dem Verbindungsmodus "Station" / "Knotenstation".

Das Übertragungskriterium steuert den Verbindungsaufbau für die Übertragung bedingt spontaner Daten-Telegramme. Dies reduziert die Anzahl der Verbindungsaufbauversuche.

Wertebereich:

- Standardbedingungen
  Es wird keine Verbindung wegen vorliegender bedingt spontaner Daten-Telegramme aufgebaut.
  Nur in folgenden Fällen wird eine Verbindung zum Senden unbedingt spontaner Daten-Telegramme aufgebaut:
  - Drohender Überlauf des Sendepuffers
  - Verbindungsaufbau durch den Kommunikationspartner
- Füllgrad
  Erst wenn der projektierte Füllgrad des Sendepuffers für bedingt spontane Daten-Telegramme überschritten wird, baut das Modul eine Verbindung auf.
  In das Eingabefeld geben Sie den Füllgrad (%) des Sendepuffers ein, bei dessen Überschreitung das Modul eine Verbindung aufbaut.
- Zeitpunkt
  Das Modul sendet bedingt spontane Daten-Telegramme zu einer projektierbaren Tageszeit, projektierbar über "Stunden" / "Minuten".
- Zeitraster
  Das Modul sendet bedingt spontane Daten-Telegramme zyklisch in einem projektierbaren Zeitraster, projektierbar über "Stunden" / "Minuten".

Polling-Überwachungszeit

Nur für WAN-Netze vom Typ Standleitung

Das Kommunikationsmodul überwacht den Telegrammverkehr über WAN. Wenn das Modul nach Ablauf einer gewissen Zeit nicht aufgerufen wird, gibt eine Meldung an seine CPU, dass die Zentrale gestört ist. Es gilt das folgende Verhalten:

- Polling-Überwachungszeit projektiert
  Nach Ablauf der projektierten Zeit gibt das Modul eine Meldung an seine lokale CPU.
- Polling-Überwachungszeit nicht projektiert
  Wenn keine Überwachungszeit projektiert ist, gibt das Modul eine Störungsmeldung nach folgenden intern berechneten Zeiten aus.
  - Bei einer Übertragungsgeschwindigkeit von 9.600 Bit/s: Nach 4 Sekunden ohne Telegrammverkehr
  - Bei einer Übertragungsgeschwindigkeit von 1.200 Bit/s: Nach 32 Sekunden ohne Telegrammverkehr

Webserver

Der Webserver des Moduls

Das Kommunikationsmodul stellt die HTML-Seiten seines Webserver für den Zugriff eines PC mit Webbrowser zur Verfügung. Die HTML-Seiten des WBM (Web Based Management) liefern Konfigurations- und Diagnose-Informationen des Moduls.

Allgemein

- **Webserver auf dieser Baugruppe aktivieren**
  Bei aktivierter Option wird die Datenverarbeitung im Webserver des Moduls aktiviert und der Zugriff per HTTP/HTTPS ermöglicht.
  Bei Modulen mit mehreren Schnittstellen müssen Sie zusätzlich die Schnittstellen für den Zugriff festlegen (siehe unten).
- **Zugriff nur über HTTPS zulassen**
  Erlaubt den Zugriff auf den Webserver nur mit dem gesicherten Protokoll HTTPS.
  Bei aktivierter Option kann nur eine gesicherte HTTPS-Verbindung über den geöffneten Port 443 der Kommunikationsbaugruppe aufgebaut werden.
  Bei deaktivierter Option wird eine HTTP-Verbindung über Port 80 der Kommunikationsbaugruppe aufgebaut werden.

Automatische Aktualisierung

- **Automatische Aktualisierung aktivieren**
  Aktiviert das automatische Aktualisieren der angezeigten Werte.
  Bei deaktivierter Option werden nur die Werte zum Zeitpunkt des Verbindens mit dem Webserver angezeigt.
- **Aktualisierungsintervall**
  Wählen Sie das Intervall, in dem Sie eine Aktualisierung der angezeigten Werte wünschen.
  Voreinstellung: 30. Zulässiger Bereich: 5...999

Übersicht der Schnittstellen

Über die Optionskästchen legen Sie fest, für welche Schnittstellen der Zugriff auf das WBM des Moduls freigegeben wird.

Die Einstellungen zur Aktivierung der Schnittstellen für den Webserver-Zugriff werden in den beiden Parametergruppen "Webserver" und "Ethernet-Schnittstelle > Zugriff auf den Webserver" wechselseitig in die jeweils andere Parametergruppe übernommen.

Wenn keine Schnittstelle für den Zugriff aktiviert ist, kann nicht auf das WBM des Moduls zugegriffen werden.

Partnerstationen

Partnerprojektierung importieren (Protokoll ST7)

Die Import-Optionen stehen nur bei Kommunikationsmodulen zur Verfügung, die das Protokoll ST7 verwenden und die zuvor als Proxy in STEP 7 V5 projektiert wurden.

Voraussetzung für den Import von Projektierungsdaten aus STEP 7 V5 ist die Auswahl des Projektierungsverfahrens "Importieren" in der Parametergruppe "Grundeinstellungen", siehe Kapitel Projektierung (Seite 1802).

Zu den unterstützten Modulen siehe Kapitel Erweiterung von SINAUT-Projekten (Seite 1900).

Parametergruppen

Proxy-Import

- Partnerprojektierung importieren
  Über diese Schaltfläche importieren Sie die Konfigurationsdatei (SDBs) aus dem STEP 7 V5-Projekt.
  Die Konfigurationsdatei muss im Dateisystem der Engineering-Station liegen.
- Partnerprojektierung zurücksetzen
  Über diese Schaltfläche löschen Sie alle Projektierungsdaten des Moduls, die Sie zuvor aus einer Textdatei über die Schaltfläche "Partnerprojektierung importieren" importiert haben. Damit setzen Sie die Partnerprojektierungsdaten zurück auf den Stand einer nicht projektierten Baugruppe.
- Importierte Konfigurationsdatei
  Zeigt den Namen der Textdatei mit den Projektierungsdaten an, Sie zuletzt geladen haben.
- Letzter Import
  Zeigt den Zeitpunkt an, an dem Sie die Konfigurationsdatei importiert haben.

Beachten Sie:

- Bei einem zweiten Import werden die Daten des ersten Imports überschrieben.
  Prüfen Sie bei einem zweiten Import immer die Konfiguration, die Verbindungen und die Datenpunkte des Moduls.
- Bei Umschaltung von "Importieren" nach "Projektieren" im Anschluss an einen Import werden alle importierten ST7-Verbindungen inklusive der zugehörigen Uhrzeitdaten gelöscht. Projektierte Datenpunkte verlieren ihren Partner.

Partnerstationen

Die Tabelle zeigt nach dem Import die aus STEP 7 V5 importierten Daten der Kommunikationspartner an:

- Teilnehmernummer
- Teilnehmer-ID
- Teilnehmer-Zusatzkennung
- Anzahl der projektierten Partner
- Teilnehmernummern der Partner

Importierte Projektierungsdaten

Importierte Daten

Beim Import der Projektierungsdaten einer Proxy-Baugruppe aus STEP 7 V5 in die kompatible Baugruppe in STEP 7 Professional werden die ST7-spezifischen Teilnehmerdaten und die SINAUT-Verbindungen importiert.

Die in STEP 7 V5 erzeugte Importdatei enthält die Daten der folgenden SDBs:

- SDB 3202: WAN data
  Daten einer seriellen Schnittstelle können nicht in einen CP 1542SP-1 IRC übernommen werden, siehe auch Kapitel Erweiterung von SINAUT-Projekten (Seite 1900).
- SDB 3203: Subscriber data
- SDB 3207: S7 connection data
- SDB 3208: Ethernet data

Datenpunkt-Informationen / SINAUT-Objekte

Das Verhalten unterscheidet sich nach der in STEP 7 V5 verwendeten TD7-Version:

- TD7onTIM
  Es werden keine Informationen in Datenpunkte umgewandelt.
- TD7onCPU
  Beim Import in eine TIM 1531 IRC werden aus den Informationen der in STEP 7 V5 parametrierten Bausteine die TD7onCPU-Bausteine der CPU 1500 angelegt.
  Wenn Sie den Import zurücknehmen ("Partnerprojektierung zurücksetzen", siehe oben), bleiben die TD7onCPU-Bausteine erhalten.

Daten der Uhrzeitsynchronisation

Daten der Uhrzeitsynchronisation werden aus STEP 7 V5 übernommen.

Bei folgenden Modulen können die Uhrzeitdaten einer seriellen Schnittstelle anschließend in STEP 7 Professional verändert werden:

- CP 1243-8 IRC ab Firmware-Version V3
- TIM 1531 IRC

Teilnehmer-Zusatzkennung

Nach dem Import der Projektierungsdaten wird unter Anderem die "Teilnehmer-Zusatzkennung" (32 Bit) angezeigt. Abhängig vom Teilnehmertyp des Partners enthält die Teilnehmer-Zusatzkennung folgende Informationen:

- CPU (TD7)
  Zustand von Bit 0:
  - 0: CPU sendet Betriebszustand nicht an Partner.
  - 1: CPU sendet Betriebszustand an Partner.

  Die restlichen Bits sind für die CPU nicht belegt.
- TIM (ST7)

Bit-Nr.	Bedeutung
Bit 0	DCF77-Uhr • 0: Nicht vorhanden • 1: Vorhanden
Bit 1	Verbindungsaufbau vom Kommunikationsmodul zur CPU • 0: Nicht erlaubt • 1: Erlaubt
Bit 2	Serielle Schnittstelle 1 (WAN-Treiber) • 0: Schnittstelle nicht vorhanden • 1: Schnittstelle vorhanden
Bit 3	Serielle Schnittstelle 2 (WAN-Treiber) • 0: Schnittstelle nicht vorhanden • 1: Schnittstelle vorhanden
Bit 4...15	Nicht belegt
Bit 16...19	Serielle Schnittstelle 1: Netztyp • 0: Standleitung • 1: Spontannetz (Mobilfunknetz) • 2: Wählnetz • 3...4: Nicht belegt • 5: MSC-Netz • 6...15: Nicht belegt
Bit 20...21	Serielle Schnittstelle 1: Netzknotentyp • 0: Zentrale-Station • 1: Knotenstation • 2: Station
Bit 22	Serielle Schnittstelle 1: Telegrammformat • 0: FT1.2 • 1: FT2
Bit 23	Nicht belegt

Bit-Nr.	Bedeutung
Bit 24...27	Serielle Schnittstelle 2: Netztyp • 0: Standleitung • 1: Spontannetz (Mobilfunknetz) • 2: Wählnetz • 3...4: Nicht belegt • 5: MSC-Netz • 6...15: Nicht belegt
Bit 28...29	Serielle Schnittstelle 2: Netzknotentyp • 0: Zentrale-Station • 1: Knotenstation • 2: Station
Bit 30	Serielle Schnittstelle 2: Telegrammformat • 0: FT1.2 • 1: FT2
Bit 31	Nicht belegt

Telecontrol-Server (Protokoll Telecontrol Basic)

Adressierung des doppelten oder redundanten Telecontrol-Servers

- **TCSB V3: Adressierung des TCSB-Redundanzverbundes**
  Für CPs ab Firmware-Version V2 mit Protokoll "Telecontrol Basic"
  In dem LAN in der Zentrale, an dem die TCSB-Server-PCs und der DSL-Router (z. B. SCALANCE M) angeschlossen sind, wird den beiden Server-PCs durch das Network Load Balancing (NLB) des Rechner-Betriebssystems eine gemeinsame virtuelle IP-Adresse vergeben.
  Diese IP-Adresse wird abhängig vom Netzaufbau projektiert:
  - Wenn CPs ohne DSL-Router angeschlossen sind, dann muss in den CPs die durch das NLB vergebene virtuelle Adresse als IP-Adresse des Telecontrol-Servers projektiert werden.
  - Bei Verwendung eines DSL-Routers wird für die Adressierung des redundanten Telecontrol-Servers in den Stationen eine einzige IP-Adresse projektiert, die öffentliche Adresse des DSL-Routers.
    Stellen Sie die Port-Weiterleitung im DSL-Router so ein, dass die öffentliche IP-Adresse (externes Netz) auf die virtuelle IP-Adresse der TCSB-Server-PCs (internes Netz) geführt wird. Nur die öffentliche IP-Adresse ist aus dem Internet erreichbar. Die Station bekommt somit keine Information, mit welchem der beiden Rechner des Redundanzverbundes sie verbunden ist.

  Wenn Sie eine zweite IP-Adresse projektieren, dann müssen Sie sicherstellen, dass TCSB über die IP-Adresse eines zweiten Routers erreichbar ist.
- **TCSB V2: Adressierung des Haupt- und Ersatz-Telecontrol-Servers**
  Nur für CP 1242-7 (6GK7 242-7KX30-0XE0)
  Mit TCSB V2 können zwei unabhängige Server-PCs installiert werden. Nur wenn diese über zwei Router an das Internet angeschlossen sind, muss auch die Adresse des Ersatz-Telecontrol-Servers projektiert werden.

Kommunikation mit der CPU

CPU-Abtastzyklus

Eingangs-Datenpunkte werden dem Lesezyklus der CPU in der Datenpunktprojektierung im Register "Allgemein > Lesezyklus" zugeordnet.

Aufbau des CPU-Abtastzyklus

Der Zyklus, mit dem die Übertragungs-Baugruppe (CP / TIM) den Speicherbereich der CPU abtastet, ist aus den folgenden Phasen aufgebaut:

- **Hochpriore Leseaufträge (Schneller Zyklus)**
  Für alle Datenpunkte mit der Zuordnung "Schneller Zyklus" werden die PLC-Variablen in jedem Abtastzyklus gelesen.
  In der Regel ist es ausreichend, nur schnell zu erfassende Daten wie Alarme und Wischermeldungen sowie Befehls-, Sollwert- und Parameter-Objekte für die 1-aus-n-Kontrolle dem schnellen Zyklus zuzuordnen.
- **Schreibaufträge**
  In jedem Zyklus werden die Werte einer bestimmten Anzahl spontaner Schreibaufträge in die CPU geschrieben.
  - Telecontrol-CP / TIM 1500:
    Die Anzahl der Variablen, die pro Zyklus geschrieben werden, wird für die Übertragungs-Baugruppe in der Parametergruppe "Kommunikation mit der CPU" mit dem Parameter "Max. Anzahl der Schreibaufträge" festgelegt. Diejenigen Variablen, deren Anzahl diesen Wert übersteigt, werden dann im nächsten oder einem der folgenden Zyklen geschrieben.
- **Niederpriore Leseaufträge - anteilig (Normaler Zyklus)**
  Für Datenpunkte mit der Zuordnung "Normaler Zyklus" werden die Werte ihrer PLC-Variablen anteilig in jedem Abtastzyklus gelesen.
  - Telecontrol-CP / TIM 1500:
    Die Anzahl der Variablen, die pro Zyklus gelesen werden, wird für die Übertragungs-Baugruppe in der Parametergruppe "Kommunikation mit der CPU" mit dem Parameter "Max. Anzahl der Leseaufträge" festgelegt. Die Variablen, die diesen Wert übersteigen und somit in einem Zyklus nicht gelesen werden, werden dann im nächsten oder einem der folgenden Zyklen gelesen.
- **Zykluspausenzeit**
  Diese Wartezeit zwischen zwei Abtastzyklen dient dazu, anderen Prozessen, die auf die CPU zugreifen, genügend Zeit zu reservieren.

Dauer des CPU-Abtastzyklus

Da für den Zyklus keine feste Zeit projektiert werden kann und da den einzelnen Phasen keine feste Anzahl an Objekten zugeordnet ist, ist die Dauer des Abtastzyklus variabel und kann sich dynamisch verändern.

Partnerstatus und Wegestatus

Die nachfolgend beschriebenen Parameter finden Sie bei Modulen mit der Kommunikationsart "ST7" in folgender Parametergruppe:

"Kommunikation mit der CPU" > "Partnerstatus"

Folgende Status werden überwacht und in PLC-Variablen geschrieben:

- **Partnerstatus**
  Erreichbarkeit eines Kommunikationspartners
- **Wegestatus**
  Zustand der Verbindungen mit einem Kommunikationspartner

Partner / Partnername

Hier legen Sie die Kommunikationspartner des selektierten Moduls fest, deren Erreichbarkeits- und Wegestatus der Telecontrol-Verbindungen überwacht werden sollen. Partner können CPUs oder eine Leistellen-Applikation (bspw. ST7cc) sein.

In der Tabelle sind alle Partner auswählbar, mit denen eine Telecontrol-Verbindung projektiert wurde.

PLC-Variablen bei den verschiedenen TIM-Typen

Die PLC-Variablen, in welche die Status geschrieben werden, unterscheiden sich je nach TIM-Typ:

- **TIM 4R-IE**
  - **Partnerstatus**
    Der Partnerstatus wird in einer PLC-Variable vom Typ Bool (DB, Merker, Ausgang) abgelegt.
    Zur Kodierung siehe unten.
  - **Wegestatus**
    Der Wegestatus wird in einer PLC-Variable vom Typ Byte (DB, Merker, Ausgang) abgelegt.
    Zur Belegung der PLC-Variable siehe unten.
- **TIM 1531 IRC / CP 1243-8 IRC**
  - **Partnerstatus / Wegestatus**
    Die beiden Status werden in einer gemeinsamen PLC-Variable vom Typ Word oder UInt (DB, Merker, Ausgang) abgelegt.
    Die beiden Status belegen jeweils ein Byte der PLC-Variable. Die Kodierung ist für Byte 0 (Partnerstatus) und Byte 1 (Wegestatus) getrennt beschrieben.

Partnerstatus

- **TIM 4R-IE**
  Das Bit für den Partnerstatus ist folgendermaßen kodiert (Wertebereich):
  - 0: Partner nicht erreichbar
  - 1: Partner erreichbar
- **TIM 1531 IRC / CP 1243-8 IRC**
  Byte 0 der gemeinsamen PLC-Variable für Partnerstatus / Wegestatus kodiert Informationen zur Erreichbarkeit des Kommunikationspartners, zu vorhandenen Verbindungen und Verbindungswegen und zum Zustand des Sendepuffers der TIM.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Wegeredundanz	**Verbindungsmodus**	**Temporäre Verbindung ****	*(Reserviert)*	**Telegrammspeicher ***		**Wegestatus**	**Partnerstatus**
0: Keine Redundanz 1: Redundanz vorhanden	0: Permanent 1: Temporär	0: Partner nicht erreichbar 1: Partner erreichbar *	-	0: Sendepuffer OK 1: Speicherbelegung > 90% 3: Überlauf (Speicherbelegung 100%)		0: Nicht alle Wege erreichbar 1: Alle Wege erreichbar	0: Partner nicht erreichbar 1: Partner erreichbar

***** Partner, die temporäre Verbindungen unterstützen, werden als "erreichbar" gesetzt, wenn der Partner selbst die Verbindung abbaut.
****** Zum Verhalten des Telegrammspeichers siehe Abschnitt "Sendepuffer" im Kapitel Prozessabbild, Übertragungsart, Ereignisklassen (Seite 1863).

Wegestatus

Byte 1 zeigt aus Sicht der lokalen TIM den Status des Verbindungswegs (projektierte Verbindung) zum Partner an.

Maximal können 2 Wege (Haupt- und Ersatzweg) zu einem Partner projektiert werden, siehe Kapitel Telecontrol-Verbindungen anlegen (Seite 1827).

Die beiden Verbindungswege müssen auf einer lokalen TIM beginnen oder enden.

Die PLC-Variable zeigt folgendes an:

- Die Wege, über die der Partner erreichbar ist.
- Der aktuell genutzte Weg
- Die TIM-Schnittstelle, über die der Hauptweg projektiert wurde.
- Die TIM-Schnittstelle, über die der Ersatzweg projektiert wurde.

Der Weg einer Verbindung wird angegeben als Kombination aus den benutzten Schnittstellen der TIM und dem Status des Wegs.

- **Byte-Belegung**
  Das Byte ist folgendermaßen belegt:
  - Zwei Bits für die Schnittstelle des Hauptwegs
  - Zwei Bits für die Schnittstelle des Ersatzwegs
  - Zwei Bits für den Status des Hauptwegs
  - Zwei Bits für den Status des Ersatzwegs

Bit 6 + 7	Bit 4 + 5	Bit 2 + 3	Bit 0 + 1
Projektierte Schnittstelle		Wegestatus	
Nr. für Ersatzweg	Nr. für Hauptweg	Ersatzweg (2. Weg)	Hauptweg (1. Weg)

- **Projektierte Schnittstelle**
  Die Schnittstellen der TIM-Baugruppen werden in der nachfolgenden Tabelle von 0 .. 3 durchnummeriert und haben folgende Bedeutung:
  - **"0"**
    TIM 4R-IE: Ethernet-Schnittstelle 1 (X3)
    TIM 1531 IRC: Ethernet-Schnittstelle 1 (X1)
    CP 1243-8 IRC: Ethernet-Schnittstelle (X1)
  - **"1"**
    TIM 4R-IE: Ethernet-Schnittstelle 2 (X4)
    TIM 1531 IRC: Ethernet-Schnittstelle 2 (X2)
    CP 1243-8 IRC: WAN-Schnittstelle (X2)
  - **"2"**
    TIM 4R-IE: WAN-Schnittstelle 1 (X1)
    TIM 1531 IRC: Ethernet-Schnittstelle 3 (X3)
  - **"3"**
    TIM 4R-IE: WAN-Schnittstelle 2 (X2)
    TIM 1531 IRC: WAN-Schnittstelle 1 (X4)

  Für den CP 1243-8 IRC sind die Schnittstellen-Nummern "2" und "3" nicht relevant.

Zustand Bit 5 / Bit 7	Zustand Bit 4 / Bit 6	Bedeutung
0	0	Kodierung für Schnittstelle "0"
0	1	Kodierung für Schnittstelle "1"
1	0	Kodierung für Schnittstelle "2"
1	1	Kodierung für Schnittstelle "3"

- **Wegestatus**
  - Hauptweg = 1. Weg, Kodierung durch Bit 0 und 1
  - Ersatzweg = 2. Weg, Kodierung durch Bit 2 und 3

Zustand Bit 1 / Bit 3	Zustand Bit 0 / Bit 2	Bedeutung Bit 1 / Bit 3	Bedeutung Bit 0 / Bit 2
0	0	Weg nicht aktuell	Teilnehmer nicht erreichbar
0	1	Weg nicht aktuell	Teilnehmer erreichbar
1	0	Weg aktuell	Teilnehmer nicht erreichbar
1	1	Weg aktuell	Teilnehmer erreichbar

Beispiel für Kodierungsmöglichkeiten des Wegestatus

Die folgende Tabelle zeigt ein Beispiel für die Kodierungsmöglichkeiten des Wegestatus bei der TIM 1531 IRC.

Die Abkürzungen für die Schnittstellen haben folgende Bedeutung:

- IE1 = Ethernet-Schnittstelle 1 (X1)
- IE2 = Ethernet-Schnittstelle 2 (X2)
- IE3 = Ethernet-Schnittstelle 3 (X3)
- WAN = Serielle Schnittstelle (X4)

Gleiche Kodierung der projektierten Schnittstelle für den Haupt- und den Ersatzweg bedeutet, dass keine Wegeredundanz vorliegt (nur 1 Schnittstelle projektiert). Der Wegestatus wird dann über die Bits des Hauptwegs (1. Weg) ausgegeben.

Tabelle 1-124 Kodierungsbeipiel für den Wegestatus

Projektierte Schnittstelle		Wegestatus	
Kodierung für Ersatzweg	Kodierung für Hauptweg	Ersatzweg (2. Weg)	Hauptweg (1. Weg)
0 0	0 0 = Kodierung für IE1	Irrelevant (nicht redundant)	Status IE1
0 0	0 1 = Kodierung für IE2	Status IE1	Status IE2
0 0	1 0 = Kodierung für IE3	Status IE1	Status IE3
0 0	1 1 = Kodierung für WAN	Status IE1	Status WAN
0 1	0 0	Status IE2	Status IE1
0 1	0 1	Irrelevant (nicht redundant)	Status IE2
0 1	1 0	Status IE2	Status IE3
0 1	1 1	Status IE2	Status WAN
1 0	0 0	Status IE3	Status IE1
1 0	0 1	Status IE3	Status IE2
1 0	1 0	Irrelevant (nicht redundant)	Status IE3
1 0	1 1	Status IE3	Status WAN
1 1	0 0	Status WAN	Status IE1
1 1	0 1	Status WAN	Status IE2
1 1	1 0	Status WAN	Status IE3
1 1	1 1	Irrelevant (nicht redundant)	Status WAN

Bedienereingaben-Überwachung

Bedienereingaben-Überwachung

Die Bedienereingaben-Überwachung dient der Überwachung von Bedienereingaben (Taster etc.) zur Absicherung gegen Fehleingaben.

Für überwachte Bedienereingaben wird eine minimale und maximale Dauer festgelegt. Der Überwachungsstatus wird in der "PLC-Variable für Bedienereingaben-Überwachung" (Byte) gespeichert.

Die PLC-Variable kann verwendet werden für Eingaben über folgende Datenpunkttypen:

- Befehlsausgang (Cmd01B_S)
- Sollwertausgang (Set01W_S)
- Parameter-Eingaben (Par12D_S)

Die Bedienereingaben-Überwachung wird vor allem dann empfohlen, wenn die Befehlseingabe über Digitaleingänge erfolgt, z. B. über daran angeschlossenen Taster. Dies gilt auch für den Fall, dass Sollwert- und Parameter-Eingaben über einen Trigger zur Übertragung gebracht werden und diese Triggerung über einen Digitaleingang erfolgt, d. h. auch hier beispielsweise über einen Taster.

Überwachung von Falscheingaben (Eingabezeit)

Für Eingaben über Digitaleingänge kann mit Hilfe der Bedienereingaben-Überwachung eine erhöhte Absicherung gegen Falscheingaben erreicht werden. Die nachfolgenden Zeiten und Anzeige-Bits sind nur relevant für Bedienereingaben über Digitaleingänge.

- Min. Eingabezeit
  Für die Eingaben lässt sich eine minimale Eingabezeit festgelegen, d. h. der jeweils betätigte Taster muss solange gedrückt bleiben, bis die eingestellte Mindestdauer abgelaufen ist. Die versehentliche Betätigung eines Tasters führt dann nicht zu einer ungewollten Übertragung. Wann die Minimale Eingabezeit abgelaufen ist und der Taster losgelassen werden kann, zeigt die "PLC-Variable für Bedienereingaben-Überwachung" in Bit 0 "Eingabe OK" an (siehe unten).
  Empfohlen wird eine Dauer von mindestens 1 Sekunde.
  Wenn Sie den Parameter nicht benötigen, dann projektieren Sie 0 (null).
- Max. Eingabezeit
  Neben einer Mindestdauer kann auch eine minimale Eingabezeit für Digitaleingänge projektiert werden. Damit können klemmende Taster oder defekte Digitaleingänge, die ein Dauer-1-Signal liefern, rechtzeitig erkannt werden. Derartige Fehler werden als "Eingabe-Fehler" in Bit 1 der PLC-Variable geschrieben.
  Weitere Hardware-Eingaben werden nicht verarbeitet, solange das Bit "Eingabe-Fehler" gesetzt ist.
  Wenn Sie den Parameter nicht benötigen, dann projektieren Sie 0 (null).

Überwachung auf 1-aus-n-Fehler

Generell für alle Bedienereingaben über Digitaleingänge, Merker oder Datenbausteine liefert die PLC-Variable den Fehlerstatus "1-aus-n-Fehler" in Bit 2. Der Fehlerstatus wird gesetzt, wenn gleichzeitig mehr als 1 Bit gesetzt wird.

Der Fehler wird in folgenden Fällen ausgegeben:

- 1-aus-n-Fehler bei "Cmd01B_S"
  Im Eingangs-Byte des Befehls-Datenpunkttyps "Cmd01B_S" wurde mehr als 1 Bit gesetzt. Zur Erhöhung der Sicherheit bei der Befehlseingabe darf bei diesem Objekt prinzipiell nur jeweils 1 Bit gesetzt sein. Wenn zwei oder mehr Bits gleichzeitig gesetzt sind, wird die Befehlseingabe abgelehnt.
- 1-aus-n-Fehler bei allen Befehls-, Sollwert- und Parameter-Eingaben im schnellen Lesezyklus
  Wenn für die Eingabe von Befehlen, Sollwerten und Parametern eine erhöhte Sicherheit gewünscht wird, sollten alle Objekte, mit denen diese Daten gesendet werden, dem schnellen Lesezyklus zugeordnet werden. Alle Befehls-, Sollwert- und Parameter-Objekte im schnellen Lesezyklus werden dann einer 1-aus-n-Prüfung unterzogen.
  Am Ende des schnellen Zyklus wird geprüft, ob nur bei einem einzigen der erfassten Objekte eine Befehls-, Sollwert- oder Parameter-Eingabe vorliegt. Nur dann wird die entsprechende Eingabe verarbeitet und übertragen. Steht mehr als nur eine Eingabe gleichzeitig an, werden die Eingaben abgelehnt. Ein neuer Befehl, Sollwert oder Parameter wird erst wieder bearbeitet, wenn vorher in mindestens einem schnellen Zyklus keine Eingabe erfasst wurde.

Hinweis

Rücksetzen des Befehls-Bits

Wenn Befehle über ein Merker- oder Daten-Byte eingegeben werden oder eine Sollwert oder Parameter-Eingabe über ein Merker- oder Daten-Bit freigegeben wird (Triggersignal), dann wird das jeweils gesetzte Befehls-Bit oder Triggersignal automatisch auf Null zurückgesetzt.

Wenn aber ein 1-aus-n-Fehler erkannt wird, dann werden die Befehls-Bits nicht automatische zurückgesetzt. Das Rücksetzen ist dann über das Anwenderprogramm vorzunehmen.

Belegung des Status-Byte der "PLC-Variable für Bedienereingaben-Überwachung"

Tabelle 1-125 Bit-Belegung des Status-Byte

Bit:	.7 *	.6 *	.5 *	.4 *	.3 *	.2	.1	.0
Status:	-	-	-	-	-	1-aus-n-Fehler	Eingabe-Fehler	Eingabe OK
Bei Wert:	0	0	0	0	0	1	1	1
* Nicht belegte Bits werden mit 0 beschrieben								

Schutz

Schutz der TIM 1531 IRC

Schutz

Über die Tabelle "Schutz" können Sie verschiedene Zugriffsstufen projektieren, um den Zugang zu den Projektierungsdaten der Baugruppe einzuschränken.

Die grünen Haken in der Spalte "Zugriff" zeigen an, welche Operationen freigegeben sind, ohne das Passwort dieser Zugriffsstufe zu kennen.

Wenn Sie nicht freigegebene Operationen nutzen möchten, dann ist die Eingabe des projektierten Passworts notwendig.

Ausnahme:
Jede Zugriffsstufe lässt auch ohne Eingabe eines Passworts den Zugriff auf die Identifikationsdaten (Name, Adressen ...) der Baugruppe über die Funktion "Erreichbare Teilnehmer" zu.

Voreingestelltes Verhalten

Voreingestellt ist die Zugriffsstufe "Vollzugriff (kein Schutz)". Jeder Nutzer kann die Projektierungsdaten der TIM lesen und verändern. Ein Passwort ist nicht projektiert und wird auch für den Online-Zugriff nicht benötigt.

Die Zugriffsstufen im Einzelnen

Sie können folgende Zugriffsstufen projektieren:

- Vollzugriff (kein Schutz)
  Die Hardware-Konfiguration und die Bausteine können von jedem gelesen und verändert werden.
- Lesezugriff
  Ohne Angabe des Passworts ist nur lesender Zugriff möglich.
  Ohne Eingabe des Passworts können Sie keine Daten (Projektierung, Firmware, Zertifikate) in die Baugruppe laden und keine schreibenden Testfunktionen ausführen.
- Kein Zugriff (kompletter Schutz)
  Weder lesender noch schreibender Zugriff auf die Baugruppe ist möglich.

Durch Legitimation mit dem Passwort erhalten Sie wieder Vollzugriff auf die Baugruppe.

Verhalten einer Passwort-geschützten Baugruppe im Betrieb

Der Schutz der Baugruppe ist wirksam, nachdem die Baugruppe geladen wurden.

Die durch Passwort geschützten Funktionen können nur von der Engineering-Station (STEP 7-PC) ausgeführt werden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Fall eines Passwortschutzes werden Sie zur Passworteingabe aufgefordert.

Beispiel:
Die Baugruppe wurde mit Lesezugriff projektiert und Sie möchten eine neue Firmware laden. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das projektierte Passwort eingegeben werden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird.

Siehe auch

Zugriffsschutz projektieren (Seite 1823)

Zugriffsschutz projektieren

Für die Baugruppe können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in der Tabelle eingegeben, sodass jedem Passwort genau eine Zugriffsstufe zugeordnet ist. Wie das Passwort wirkt, steht in der Spalte "Zugriffsstufe".

Beispiel (Kein Zugriff):

Sie wählen die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" und geben für die beiden in der Tabelle darüber liegenden Zugriffsstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die Baugruppe komplett geschützt.

Für Nutzer, die eines der Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile "Vollzugriff (kein Schutz)" bewirkt, dass die Baugruppe für Nutzer, die dieses Passwort kennen, ungeschützt ist. Sie haben uneingeschränkten Zugriff auf die Baugruppe.
- Das Passwort in Zeile "Lesezugriff" bewirkt Schreibschutz der Baugruppe. Nutzer, die dieses Passwort kennen, haben lesenden Zugriff auf die Baugruppe.

Vorgehen

Gehen Sie folgendermaßen vor, um die Zugriffsstufen in der Tabelle "Schutz" zu projektieren:

1. Wählen Sie die gewünschte Zugriffsstufe in der ersten Spalte der Tabelle aus. Die grünen Haken in den Spalten rechts der jeweiligen Schutzstufe zeigen Ihnen, welche Operationen noch möglich sind, ohne das Passwort einzugeben.
2. Wenn Sie eine andere Zugriffsstufe als "Vollzugriff gewählt haben:
   - Vergeben Sie in der Zelle "Passwort" in der ersten Zeile (Vollzugriff) ein Passwort für den Vollzugriff.
   - Wiederholen Sie zum Schutz vor Fehleingaben das gewählte Passwort.

   Achten Sie darauf, dass das Passwort ausreichend sicher ist, d. h. dass es kein durch eine Maschine erkennbares Muster besitzt!
   Die Eingabe eines Passwortes in der Zeile "Vollzugriff (kein Schutz)" ist obligatorisch und ermöglicht dem Passwort-Kenner uneingeschränkten Zugriff auf die Baugruppe, unabhängig von der gewählten Schutzstufe.

3. Weisen Sie weiteren Zugriffsstufen nach Bedarf weitere Passwörter zu, falls die gewählte Zugriffsstufe das erlaubt.
4. Laden Sie die Projektierungsdaten in die Baugruppe, damit die Zugriffsstufe wirksam wird.

Ergebnis

Die Projektierungsdaten sind entsprechend der eingestellten Zugriffsstufen vor unberechtigtem Online-Zugriff geschützt. Wenn eine Operation aufgrund der projektierten Zugriffsstufe nicht ohne Passwort ausgeführt werden kann, wird ein Dialog zur Eingabe des Passworts aufgeblendet.

Telecontrol-Verbindungen

Telecontrol-Verbindungen

Telecontrol-Verbindungen

Für die Fernwirk-Kommunikation werden Telecontrol-Beziehungen zwischen den beteiligten Kommunikationsmodulen benötigt. Abhängig vom Modultyp und der Firmware-Version nehmen Sie die Projektierung in folgenden Parametergruppen vor:

- Parametergruppe "Partnerstationen"
  oder
- Editor "Netzwerkdaten"

Projektierung in der Parametergruppe "Partnerstationen"

Für S7-1200-Telecontrol-CPs bis V2.1, die nur als Station fungieren, werden die Beziehungen zur Zentrale-Station bzw. zum Master in der Parametergruppe "Partnerstationen" projektiert. Hier legen Sie die Kommunikationspartner der CPs fest.

Alle weiteren Einstellungen, die für die Kommunikation mit der Zentrale-Station benötigt werden, werden aus den weiteren Projektierungsdaten der CPs herangezogen und müssen für die Verbindungen nicht extra projektiert werden.

Hinweis

Ausnahme: CP 1243-8 IRC V2.1 (ST7)

Ab STEP 7 V15.1 können Sie einen physischen CP mit Firmware V2.1 im STEP 7-Projekt mit der Firmware-Version V3.0 projektieren:

1. Fügen Sie im STEP 7-Projekt einen CP mit der Firmware-Version V3.0 ein.
2. Aktivieren Sie in der Parametergruppe "Allgemein > Kataloginformation" des CP die Option "Kompatibilitätsmodus V2.1".

Anschließend können Sie den CP in Telecontrol-Verbindungen, die Sie über den Editor "Netzwerkdaten" projektieren, verwenden.

Die V3.0-Projektierungsdaten können Sie in den CP V2.1 laden.

Folgende Funktionen werden bei aktivierter Option nicht unterstützt:

- Umschaltung des Protokolltyps
- Uhrzeit von lokaler Station und über NTP empfangen
- "Kommunikation mit der CPU > CP-Diagnose"
- Online & Diagnose > "Funktionen > Servicedaten speichern"

Projektierung im Editor "Netzwerkdaten"

Für alle übrigen Telecontrol-Module, die nicht in die oben genannte Gruppe fallen, projektieren Sie Telecontrol-Verbindungen im Editor "Netzwerkdaten".

Der Editor "Netzwerkdaten" wird für folgende Protokolle und Module genutzt:

- ST7
  Module ohne importierte Projektierungsdaten:
  - CP 1243-8 IRC ab Firmware V3.0
  - CP 1542SP-1 IRC ab Firmware V2.0
  - TIM 1531 IRC ab Firmware V1.0
  - TIM 3V-IE / TIM 4R-IE ab Firmware-Versionen V2.7
- DNP3
  - CP 1243-1 / CP 1243-8 IRC ab Firmware V3.0
  - CP 1542SP-1 IRC ab Firmware V2.0
  - TIM 1531 IRC ab Firmware V2.0
  - TIM 3V-IE DNP3 / TIM 4R-IE DNP3 ab Firmware V3.2
- IEC 60870-5
  - CP 1243-1 / CP 1243-8 IRC ab Firmware V3.0
  - CP 1542SP-1 IRC ab Firmware V2.0
  - TIM 1531 IRC ab Firmware V2.0

Telecontrol-Verbindungen anlegen

Öffnen des Editors "Netzwerkdaten" > Register "Telecontrol"

Gehen Sie zum Öffnen des Editors folgendermaßen vor:

1. Öffnen Sie die Netzsicht des Projekts.
   Rechts finden Sie den eingeklappten Editor "Netzwerkdaten".

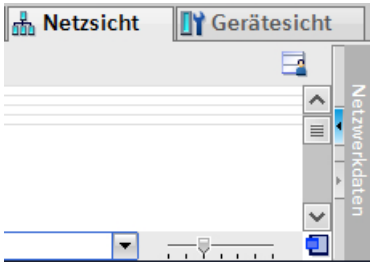

2. Öffnen Sie den Editor "Netzwerkdaten" über das Pfeilsymbol.
   Der Editor wird mit mehreren Registern eingeblendet, links das Register "Netzübersicht".
3. Ziehen Sie den Editor soweit heraus, bis das Register "TeleControl" erscheint.
   Dieses Register ist weiter unterteilt in folgende Register:
   - ST7
   - DNP3
   - IEC

   Wählen Sie je nach genutztem Protokoll das entsprechende Register, um die Telecontrol-Verbindungen zu projektieren.

Anzeigen und Ein-/Ausblenden von Spalten

In der Tabelle "Telecontrol-Verbindungen" können Sie die Spalten anzeigen oder ausblenden, anordnen und die Spaltenbreite optimieren. Klicken Sie mit der rechten Maustaste auf einen Spaltenkopf, um zu dem Kontextmenü zu gelangen.

- Spalten anordnen
  Wenn Sie auf einen Spaltenkopf klicken, können Sie die Spalte mit gedrückter linker Maustaste innerhalb der Tabelle verschieben.
- Anzeigen/Verbergen
  Über diese Funktion des Kontextmenüs können Sie einzelne Spalten ein- oder ausblenden. Damit lässt sich die Übersichtlichkeit der Tabelle erhöhen.
- Alle Spalten anzeigen
  Blendet alle Spalten der Tabelle ein.
- Breite optimieren / Breite aller Spalten optimieren
  Über diese Kontextmenüs optimieren Sie die Breite der selektierten Spalte bzw. aller Spalten der Tabelle.
  Die Spaltenbreite passt sich an den breitesten Eintrag in dieser Spalte an.

Anlegen von Telecontrol-Verbindungen

Gehen Sie zum Anlegen einer Telecontrol-Verbindung folgendermaßen vor:

1. Doppelklicken Sie in der Tabelle "Telecontrol-Verbindungen" in ein Feld der ersten freien Zeile.
   Eine neue leere Verbindung wird angelegt. Der Verbindungsname wird vorbelegt.
2. Wählen Sie den Startpunkt der Verbindung aus der Klappliste im Feld "Startpunkt" aus.
   Die Klappliste des Feldes zeigt die möglichen Startpunkte an.
3. Klicken Sie zum Ändern des Verbindungsnamens in das Feld "Name".
   Siehe auch nachfolgender Abschnitt.
4. Projektieren Sie in den weiteren Feldern die erforderlichen Parameter, wie bspw. Schnittstellen und Endpunkt der Verbindung.

Manche Felder der Tabelle sind editierbar, in anderen projektieren Sie den Parameter über eine Klappliste.

Felder mit fehlender oder fehlerhafter Projektierung werden rot hinterlegt, im Beispiel der Abbildung "TC_Connection_3".

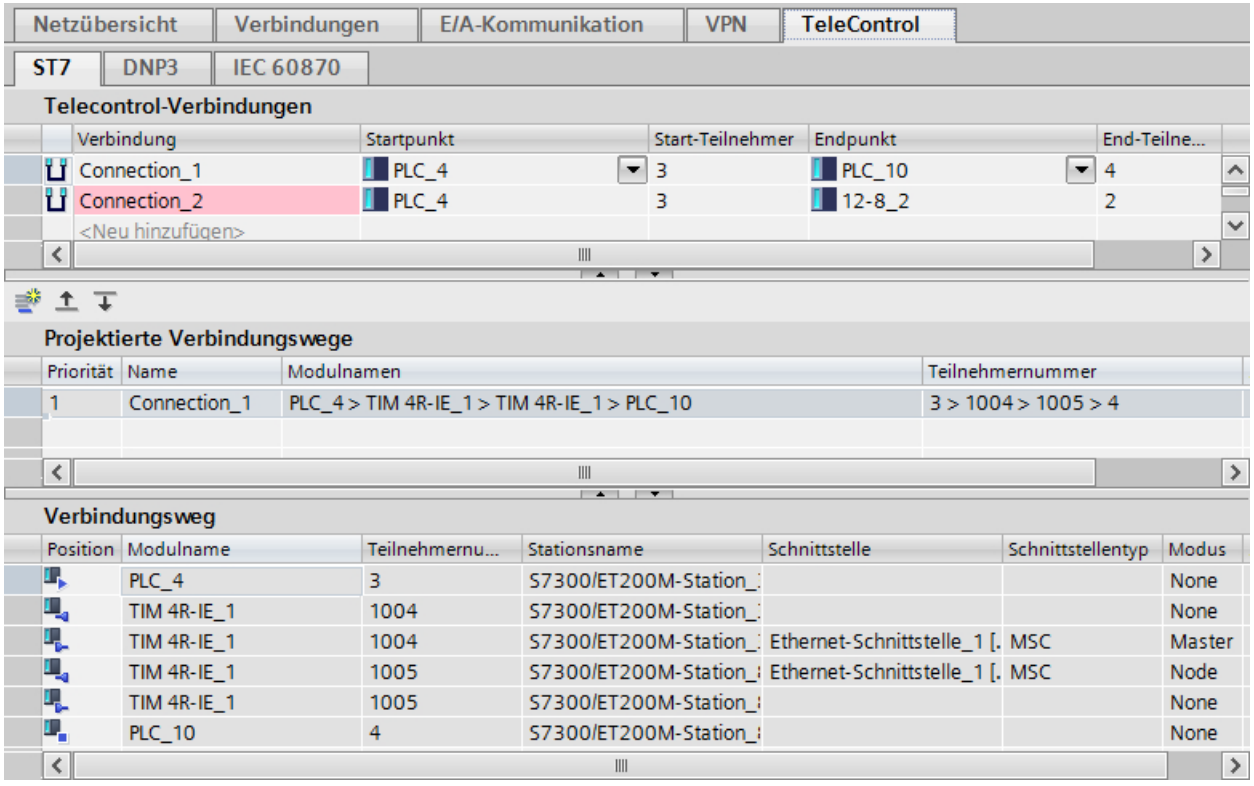

Bild 1-72 Editor "Netzwerkdaten", Register "Telecontrol > ST7", Beispiel von ST7-Verbindungen

In den Registern "DNP3" und "IEC" werden die Verbindungen etwas unterschiedlich angezeigt, siehe Protokoll-spezifische Kapitel.

Namen der Verbindungen

Sie können die vorbelegten Namen der Verbindungen anpassen.

Erlaubt sind max. 129 Zeichen aus folgenden ASCII-Zeichensätzen (Nummern dezimal):

- **Nr. 32..126**
  Leerzeichen , ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
- **Nr. 128, 130..140, 142, 145..156, 158..159, 161..172**
  € , ƒ „ … † ‡ ˆ ‰ Š ‹ Œ Ž ‘ ’ “ ” • – — ˜ ™ š › œ ž Ÿ ¡ ¢ £ ¤ ¥ ¦ § ¨ © ª « ¬
- **Nr. 174..255**
  ® ¯ ° ± ² ³ ´ µ ¶ ¹ º » ¼ ½ ¾ ¿ À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß à á â ã ä å æ ç è é ê ë ì í î ï ð ñ ò ó ô õ ö ÷ ø ù ú û ü ý þ ÿ

Festlegung des Verbindungswegs

Nach dem Anlegen einer Verbindung steht in der Regel noch nicht der tatsächliche Verbindungsverlauf fest. Insbesondere bei größeren Netzen sind häufig mehrere Verbindungswege möglich.

Legen Sie den tatsächlichen Verbindungsweg der angelegten Telecontrol-Verbindung über das Symbol "Neuen Verbindungsweg hinzufügen" fest. Der Dialog "Verbindungswege hinzufügen" öffnet sich.

Weitere Details finden Sie in den nachfolgenden Protokoll-spezifischen Kapiteln der Telecontrol-Verbindungen.

Symbole im Dialog "Verbindungswege hinzufügen"

Bei der Prüfung der projektierten Verbindungen unterstützt Sie die Tabelle "Verbindungsweg". Für jede projektierte Verbindung wird hier der detaillierte Verbindungsweg angezeigt.

In der Spalte "Position" wird ein Stationssymbol mit einer Kennung für den Verbindungspunkt angezeigt:

Die Farbe der Kennung kennzeichnet die Gültigkeit des Verbindungspunkts:

- Blau: Gültiger Verbindungspunkt
- Rot: Ungültiger Verbindungspunkt

Symbol	Bedeutung
	Startpunkt
	Knoten - Eingang
	Knoten - Ausgang
	Endpunkt
	Beispiele für ungültige Verbindungspunkte

Fehleranzeigen

Fehlerhafte Verbindungspunkte, Netze oder Parameter werden in den Tabellen rot hinterlegt.

Ursachen für fehlerhafte Verbindungen sind beispielsweise:

- Startpunkt und Endpunkt sind identisch.
- Die Verbindung läuft über ein unzulässiges Netz.
- Die Verbindung läuft über einen unzulässigen Teilnehmer.

Beispiel für die Darstellung:

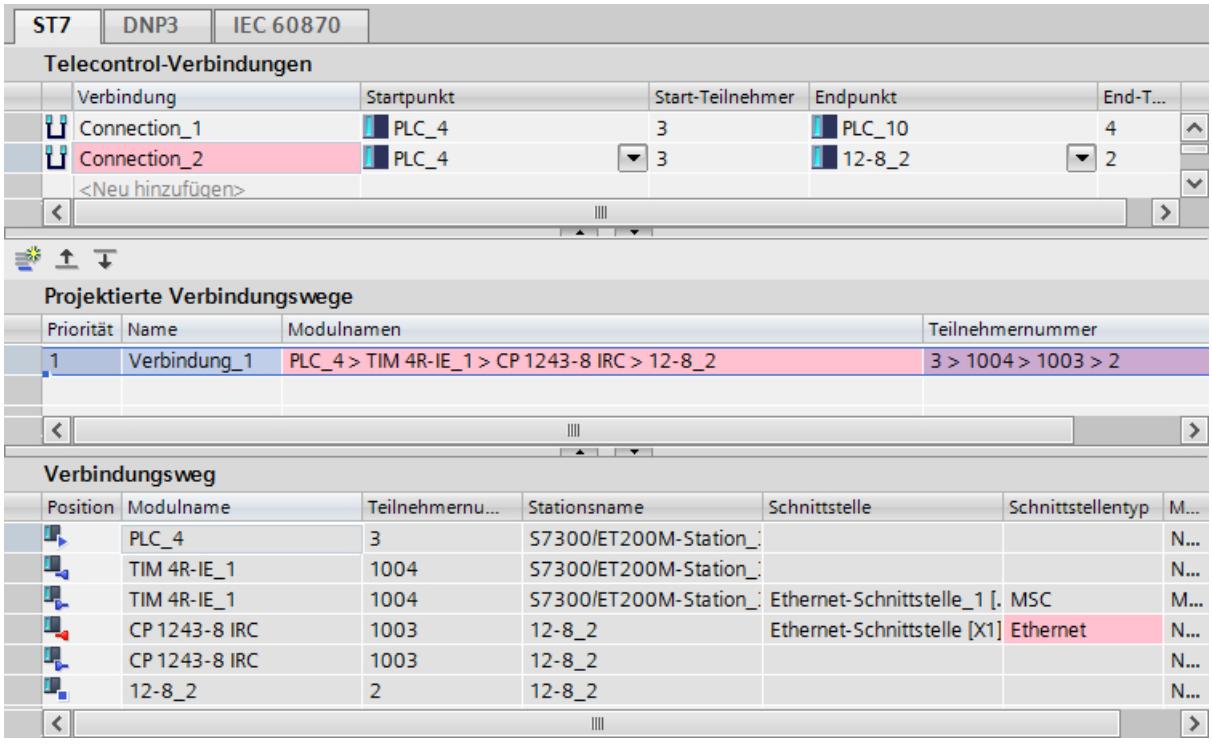

Bild 1-73 Redundante Verbindung mit fehlerhaft projektiertem Verbindungsweg (Route_2)

Der in der Abbildung angezeigte Weg "Route_2" einer ST7-Verbindung wurde über ein unzulässiges MSC-Netz angelegt. Die Anbindung der PC-Station an das MSC-Netz wurde anschließend gelöscht, sodass die Telecontrol-Verbindung über nicht mehr vorhandene Teilnehmer läuft.

Allgemeine Regeln für die Verbindungsprojektierung

Beachten Sie folgende allgemeine Regeln für die Verbindungsprojektierung:

- Eine Verbindung muss über ein einziges Subnetz laufen.
  Wenn ein "Fremdgerät" als Partner in einem anderen Subnetz liegt, dann ist der Netzübergang zwischen den Subnetzen der Endpunkt für die Verbindungsprojektierung.
- Alle Abschnitte einer Telecontrol-Verbindung müssen mit dem gleichen Telecontrol-Protokoll projektiert werden.
- Eine Verbindung über ein inkonsistentes Netz ist ungültig.
  Beispiele:
  - Teilnehmer mit inkompatiblen Netzknotentypen (Bsp.: 2 Master)
  - Verbindungen über Knoten, die nicht als Knotenstation projektiert sind.
  - Teilnehmer mit inkompatiblen Modems
  - Inkompatible Einstellungen von zwei Modems in einer Verbindung
  - Inkompatible Einstellungen zwischen Modem- und Netz-Parametern
  - Zwei Verbindungen zu einem Partner über die selbe Schnittstelle eines Moduls sind nicht erlaubt.
  - Eine Verbindung über ein Netz, in der zwei analoge Wählnetz-Modems (bspw. MD3) über 1200 Bit/s / halbduplex / AT-Modus miteinander kommunizieren, ist unzulässig.

Weitere Protokoll-spezifische Regeln finden Sie in den nachfolgenden Kapiteln.

Löschen ungültiger oder redundanter Verbindungen

Im Fall von unzulässigen oder unerwünschten redundanten Verbindungen müssen Sie einen Verbindungsweg löschen:

1. Selektieren Sie in der Tabelle "Projektierte Verbindungswege" den nicht erwünschten Verbindungsweg.
2. Klicken Sie auf das Kontextmenü "Löschen".

Register "Eigenschaften" der Verbindung

Wenn Sie in der Tabelle "Telecontrol-Verbindungen" im Editor "Netzwerkdaten" eine Verbindung selektieren, werden im Register "Eigenschaften" des Inspektorfensters weitere Parametergruppen zu dieser Verbindung angezeigt. Hier können Sie die Verbindungsprojektierung prüfen und ggf. korrigieren und weitere Eigenschaften projektieren.

Parametergruppen:

- **Allgemein**
  Zeigt die grundlegenden Eigenschaften der Telecontrol-Verbindung.
  Den Verbindungsnamen und die Teilnehmernummer / Stationsadresse des Start- und Endpunkts können Sie verändern. Dies wirkt sich auf die Projektierungsdaten der Teilnehmer aus.
- **Weitere Protokoll-spezifische Parameter**
  Die Beschreibung finden Sie nachfolgend.

ST7-Verbindungen

ST7-spezifische Regeln für die Verbindungsprojektierung

Beachten Sie folgende spezielle Regeln für die Verbindungsprojektierung:

- Eine MSC-Verbindung kann nur über die erste Ethernet-Schnittstelle [X1] einer TIM oder über die serielle Schnittstelle eines Moduls mit der Einstellung "IP-basiert" laufen.
- Verbindungen zwischen einer PC-Applikation und einem Modul dürfen nur vom Netztyp "Neutral" sein (S7-Verbindungen).
- Für Verbindungen zwischen einer redundanten PC-Applikation und einem Modul müssen Sie zwei Verbindungen anlegen.
  Legen Sie für eine redundante ST7cc-Leitstelle in STEP 7 zwei PC-Stationen mit je einer ST7cc-Applikation an (siehe unten).
  Nur so können Sie von jeder Applikation eine Verbindung projektieren.

Anlegen von ST7-Verbindungen

Für die Kommunikation zwischen zwei ST7-Stationen über klassische WAN-Netze (Standleitung, Wählnetz) benötigen Sie mindestens eine Telecontrol-Verbindung.

Im Feld "Startpunkt" werden folgende projektierbaren Startpunkte angezeigt:

- S7-Stationen: Zugeordnete CPU
- PC-Stationen: Applikation (bspw. ST7cc / ST7sc)

Wählen Sie für Verbindungen zwischen einer Applikation im Zentrale-PC und einer S7-Station immer die Applikation als Startpunkt aus, da sie nur in der Liste der Startpunkte auswählbar ist.

Nach dem Klicken auf das Symbol "Neuen Verbindungsweg hinzufügen" ( ) öffnet sich der Dialog "Verbindungswege hinzufügen", in dem die möglichen Verbindungswege gesucht werden. Die gefundenen Verbindungswege werden in der Tabelle "Selektieren Sie einen Verbindungsweg..." angezeigt.

Bei ST7-Verbindungen befindet sich unterhalb der Tabelle "Telecontrol-Verbindungen" die Tabelle "Projektierte Verbindungswege". Diese enthält zu jeder selektierten Verbindung den hinzugefügten Verbindungsweg.

Bei redundanten Verbindungen werden beide Wege angezeigt.

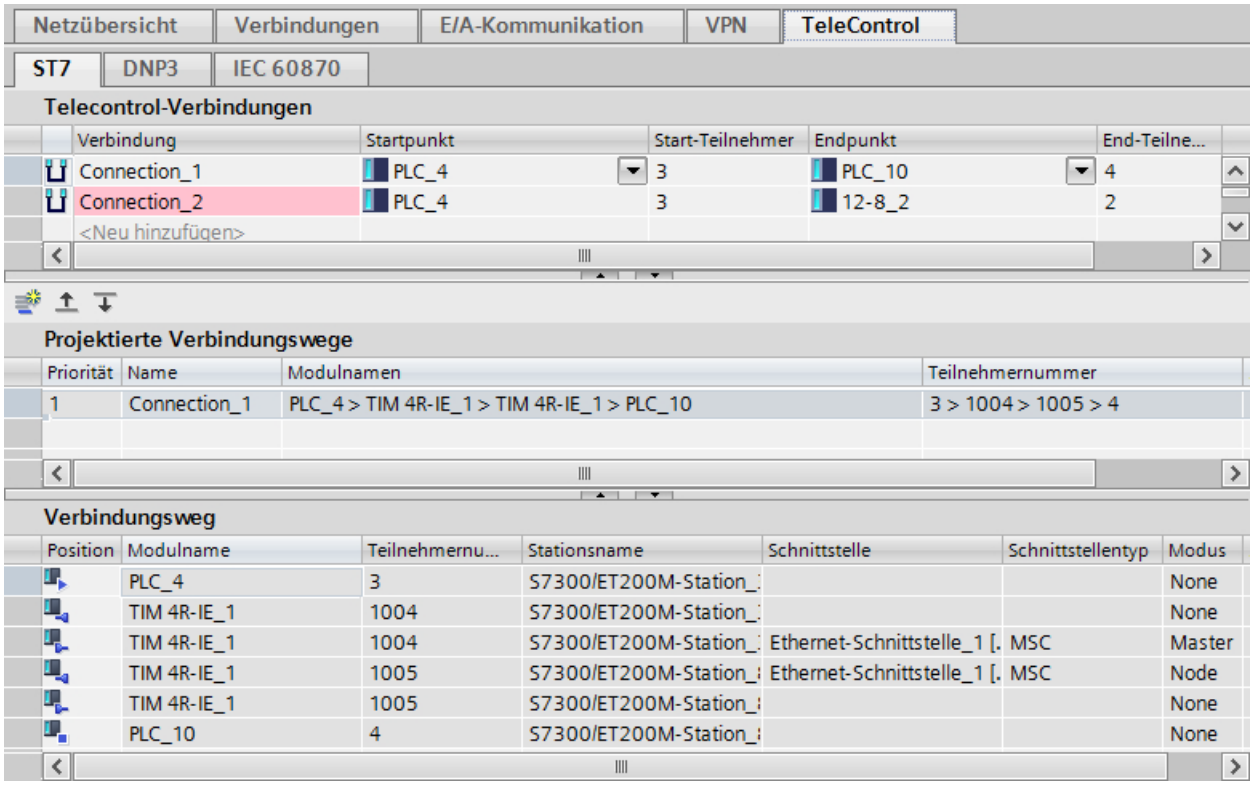

Bild 1-74 Editor "Netzwerkdaten", Register "Telecontrol"

In der dritten Tabelle "Verbindungsweg" werden Details zum selektierten Weg angezeigt.

Festlegung desVerbindungswegs

Legen Sie den tatsächlichen Verbindungsweg der angelegten Telecontrol-Verbindung folgendermaßen fest.

1. Selektieren Sie in der Tabelle "Telecontrol-Verbindungen" eine Verbindung.
2. Klicken Sie auf das Symbol "Neuen Verbindungsweg hinzufügen".
   Der Dialog "Verbindungswege hinzufügen" öffnet sich.
   Die möglichen Verbindungswege werden automatisch gesucht, erkennbar am Laufbalken unten im Dialog. Status und Ergebnis der Suche werden unten im Feld "Informationen" ausgegeben.
   Die gefundenen Verbindungswege werden in der Tabelle "Selektieren Sie einen Verbindungsweg..." angezeigt.

3. Selektieren Sie einen der möglichen Verbindungswege.
   Zum selektierten Verbindungsweg werden Details in der Tabelle "Verbindungsweg" eingeblendet.
   Die Tabelle "Vorschau" enthält weitere Details über alle Verbindungspunkte, die an dem selektiertenVerbindungswegbeteiligt sind.
   – Wenn mehrere Verbindungen in der Tabelle "Selektieren Sie einen Verbindungsweg..." eingeblendet sind, können Sie einen oder zwei redundante Verbindungen für die Auswahl des zu nutzenden Verbindungswegs auswählen.
   – Wenn keine Verbindung in der Tabelle angezeigt wird, liegt ein Projektierungsfehler in den zugehörigen Stationen oder Netzen vor.
     Schließen Sie in diesem Fall den Dialog über die Schaltfläche "Schließen" und vervollständigen Sie die Projektierung.
4. Selektieren Sie einen Verbindungsweg und klicken Sie auf "Hinzufügen", um diesen Weg als projektierte Verbindung dem Projekt hinzuzufügen.
   Unter "Informationen" wird angezeigt, ob die Verbindung hinzugefügt wurde oder ob sie bereits projektiert ist.
5. Selektieren Sie gegebenenfalls einen zweiten Verbindungsweg und klicken Sie auf "Hinzufügen", um diesen Weg als redundante projektierte Verbindung hinzuzufügen.
6. Schließen Sie den Dialog über die Schaltfläche "Schließen", wenn die hinzugefügten Verbindungswege den Projektvorgaben entsprechen.

Anlegen redundanter ST7-Verbindungen

Wenn in der Netzprojektierung verschiedene Netze zwischen zwei Teilnehmern angelegt sind, dann werden die verschiedenen Verbindungswege angezeigt:

> Dialog "Verbindungswege hinzufügen" > Tabelle "Selektieren Sie einen Verbindungsweg ..."

Wenn Sie nur einen Verbindungsweg nutzen möchten, dann selektieren Sie nur den gewünschten Weg und klicken auf "Hinzufügen".

Wenn Sie zwei redundante Verbindungswege für die ST7-Verbindung nutzen möchten, dann selektieren Sie beide nacheinander und klicken jeweils danach auf "Hinzufügen".

Nach dem Schließen des Dialogs "Verbindungswege hinzufügen" werden zu dieser Verbindung beide Wege in der Tabelle "Projektierte Verbindungen" angezeigt.

Der erste von beiden ist der Hauptweg und der zweite der Ersatzweg. Dies wird in der Spalte "Priorität" der Tabelle "Projektierte Verbindungen" angezeigt:

1. Hauptweg
2. Ersatzweg

Die Priorität der beiden Verbindungswege (Haupt- / Ersatzweg) können Sie nachträglich ändern. Hierzu dienen die Pfeiltasten:

Register "Eigenschaften" der Verbindungen

Zur Parametergruppe "Allgemein" siehe Kapitel Telecontrol-Verbindungen anlegen (Seite 1827).

- **Redundantes ST7cc/sc**
  Wenn Sie eine redundante Leitstelle in Ihrem Projekt verwenden (bspw. ST7cc oder ST7sc), dann müssen Sie zunächst zwei PC-Stationen und die einzelnen Verbindungen zwischen dem Modul und den beiden Applikationen anlegen.
  Erst im Anschluss legen Sie hier den Redundanzverbund für die Verbindung zum Redundanzverbund fest.
  - Redundanten ST7cc/sc-Partner hinzufügen
  - Aktivieren Sie die Option für jede der beiden Verbindungen zwischen der redundanten Applikation und dem Modul.
  - Redundante Applikation
    Wählen Sie aus der Klappliste "Redundante Applikation" die gewünschte Applikation aus.
- **Zielteilnehmer-Eigenschaften**
  Die Parametergruppe bezieht sich auf den Zielteilnehmer, der als Endpunkt der Verbindung projektiert wurde.
  - **Generalabfrage-Überwachungszeit**
    Dies ist die Zeit, die ein Zielteilnehmer (ST7-fähige Kommunikationsbaugruppe) zur vollständigen Beantwortung einer Generalabfrage (GA) der Zentrale maximal benötigen darf.
    Wenn die GA-Antwort nach Ablauf der Überwachungszeit noch nicht vollständig beim abfragenden Partner eingetroffen ist, wird eine Meldung in den Diagnosepuffer des Moduls eingetragen.
    Die Generalabfrage-Überwachungszeit beträgt grundsätzlich 900 Sekunden.
  - **Telegramme an Zielteilnehmer mit Zeitstempel**
    Unabhängig von der Projektierung werden alle Telegramme an den Zielteilnehmer mit Zeitstempel gesendet.

DNP3- und IEC-Verbindungen

Kommunikationsmöglichkeiten

Kommunikationswege

Folgende Kommunikationswege sind möglich:

- Verbindungen zwischen Master und Station
- Redundante Verbindungen
  Wenn zwei Teilnehmer über unterschiedliche Wege oder Netze erreichbar sind, können Sie zum Aufbau von Wegeredundanz maximal zwei Verbindungen zwischen den Teilnehmern anlegen.
- Direkte Kommunikation
  Bei der direkten Kommunikation kommunizieren Stationen direkt miteinander, ohne dass die Telegramme von einer Zentrale oder Station vermittelt werden. Zwei Stationen können somit direkt miteinander kommunizieren.
  Die Funktion legen Sie bei der Projektierung eines Datenpunkts über den Parameter "Master-Funktion" fest. Als weitere Voraussetzung muss in der Telecontrol-Verbindung die Option "Spontanes Melden" aktiviert sein.
  Die Funktion wird von folgenden Modulen unterstützt:
  - S7-1200-Telecontrol-CPs ab Firmware V3.1
  - CP 1542SP-1 IRC ab Firmware V2.0
  - TIM 1531 IRC ab Firmware V2.0
  - TIM 3V-IE DNP3 / TIM 4R-IE DNP3 ab Firmware V3.2

Verbindungen projektieren

Protokollspezifische Regeln für die Verbindungsprojektierung

Beachten Sie folgende spezielle Regeln für die Verbindungsprojektierung:

- Sie können Verbindungen zwischen Teilnehmern anlegen, die im STEP 7-Projekt projektiert sind.
  Die Adress- und Schnittstellen-Daten beider Teilnehmer stehen in der Verbindungstabelle zur Verfügung.
- Sie können Verbindungen zwischen einem Teilnehmer anlegen, der in STEP 7 projektiert ist, und einem "Fremdgerät", das nicht in STEP 7 projektiert ist.
  Der Teilnehmer, der nicht in STEP 7 projektiert ist, wird im Feld "Endpunkt" der Verbindungstabelle als "Fremdgerät" eingeblendet. Die Adressdaten dieses Teilnehmer projektieren Sie über die realen Adressdaten (IP-Adresse / Telefonnummer).
- Sie können einfache und redundante Verbindungen zu Teilnehmern in anderen Subnetzen projektieren, die über einen Netzübergang erreichbar sind. Hierzu stehen die entsprechenden Felder in der Verbindungstabelle zur Verfügung.

Festlegung des einfachen oder redundanten Verbindungswegs

Legen Sie den tatsächlichen Verbindungsweg der angelegten Telecontrol-Verbindung folgendermaßen fest.

1. Selektieren Sie in der Tabelle "Telecontrol-Verbindungen" eine Verbindung.
2. Klicken Sie auf das Symbol "Neuen Verbindungsweg hinzufügen".
   Der Dialog "Verbindungswege hinzufügen" öffnet sich.
   Die möglichen Verbindungswege werden automatisch gesucht, erkennbar am Laufbalken unten im Dialog.
   - Status und Ergebnis der Suche werden unten im Feld "Informationen" ausgegeben.
   - Die gefundenen Verbindungswege werden in der oberen Tabelle "Selektieren Sie einen Verbindungsweg..." angezeigt.
   - Zu einem selektierten Verbindungsweg werden Details in der Tabelle "Verbindungsweg" eingeblendet.
   - Bei Selektion eines Verbindungswegs wird in der Tabelle "Vorschau" angezeigt, welche Verbindungspunkte des selektierten Verbindungswegs in den Verbindungseditor übernommen werden, wenn Sie auf "Hinzufügen" klicken.
3. Selektieren Sie den bzw. die gewünschten Verbindungswege.
   - Wenn ein oder mehrere Verbindungswege in der oberen Tabelle angezeigt werden, dann selektieren Sie den gewünschten Verbindungsweg und klicken auf "Hinzufügen". Unter "Informationen" wird angezeigt, ob der Verbindungsweg hinzugefügt wurde oder ob er bereits projektiert ist.
     Wenn Sie eine redundante Verbindung nutzen möchten, dann selektieren Sie einen zweiten Weg und klicken auf "Hinzufügen".
     Schließen Sie den Dialog über die Schaltfläche "Schließen", wenn die hinzugefügten Verbindungswege den Projektvorgaben entsprechen.
   - Wenn keine Verbindung in der Tabelle angezeigt wird, liegt ein Projektierungsfehler in den zugehörigen Stationen oder Netzen vor.
     Schließen Sie in diesem Fall den Dialog über die Schaltfläche "Schließen" und vervollständigen Sie die Projektierung.

Schnittstellen-spezifische Projektierung der Verbindungs-Abschnitte

ST7 | DNP3 | IEC 60870

Telecontrol-Verbindungen

Verbindung	Startpunkt	Start-Teilnehmer	Start-Schnittstelle	Endpunkt	End-Teilnehmer	Partnerliste
*	15_	*		*	*	
_1	15_4	1	TIM_4 - Ethernet..	15_6	3	3
_3	15_6	3	TIM_2 - Ethernet.	15_4	1	1
_5	15_4	1	TIM_4 - Ethernet..	15_3_node	2	2,3
_6	15_6	3	TIM_2 - Ethe..	15_3_n...	2	1,2
_7	15_3_node	2	TIM_3_node - Et..	15_6	3	3

Bild 1-75 Editor "Netzwerkdaten", Register "Telecontrol > IEC...", Beispiel von Verbindungs-Abschnitten

In der Spalte "Startpunkt" ist der Filter "15_" gesetzt. Die Bedeutung wird in der jeweiligen Verbindungstabelle beschrieben (siehe unten).

Hinweis

Abschnitt für mehrere Verbindungen

Ein Verbindungsabschnitt zwischen zwei Teilnehmern, der für mehrere Verbindungen verwendet wird, taucht in der Verbindungstabelle nur einmal auf.

DNP3- und IEC-Verbindungen können über mehrere Teilnehmer laufen. Ein Verbindungsabschnitt zwischen zwei Teilnehmern kann auch für mehrere Verbindungen genutzt werden.

Da Sie für die einzelnen Verbindungsabschnitte individuelle Einstellungen projektieren können, werden die Verbindungsabschnitte zwischen den Schnittstellen zweier Teilnehmer jeweils in separaten Zeilen angezeigt.

In der Abbildung ist Station "15_3_node" eine Knotenstation, die in der Anlagenhierarchie zwischen der Zentrale (15_4) und der Station (15_6) liegt. Die Besonderheiten von Knotenstationen sind beim Parameter "Partnerliste" in der jeweiligen Verbindungstabelle beschrieben (siehe unten).

Parameter der DNP3-Verbindungen

Die Beschreibung der Parameter finden Sie im Kapitel Verbindungstabelle - DNP3 (Seite 1839).

Parameter der IEC-Verbindungen

Die Beschreibung der Parameter finden Sie im Kapitel Verbindungstabelle - IEC (Seite 1843).

Register "Eigenschaften" der Verbindung

- **Allgemein**
  Siehe Kapitel Telecontrol-Verbindungen anlegen (Seite 1827).
- **DNP3-Security-Optionen**
  Siehe Kapitel DNP3-Security-Optionen (Seite 1912).
- **Optionen 1. Weg / Optionen 2. Weg**
  Siehe Kapitel Optionen (IEC) (Seite 1847).
- **Fremdgeräte-Parameter**
  Siehe Kapitel Fremdgerät-Parameter (Seite 1847).

Verbindungstabelle - DNP3

Filter

Die erste Zeile unter dem Tabellenkopf enthält eine Filterfunktion, mit der Sie die Auswahl der projektierbaren Teilnehmer und Verbindungsmöglichkeiten einschränken können. Dadurch werden die Kombinationsmöglichkeiten verringert und die Übersichtlichkeit erhöht.

Wenn Sie einige Verbindungsabschnitte angelegt haben, aktivieren Sie den Filter, indem Sie einen sich wiederholenden Namen oder Teilnamen in die Filterzelle eintippen. Die Zelle wird farbig hinterlegt, vergleiche Abbildung im Kapitel Verbindungen projektieren (Seite 1836).

Beispiel:
Sie haben Verbindungen mit den Startpunkten "1200", "PLC_1" und "PLC_2" angelegt.
Bei Eingabe von "PLC_" in die Filterzelle werden in der Tabelle nur die Abschnitte eingeblendet, deren Startpunkte mit diesem Teil-String beginnen: "PLC_1" und "PLC_2"

Gesetzte Filter in mehreren Spalten multiplizieren sich.

Die Auswahl "*" blendet alle vorhandenen Verbindungsabschnitte ein.

Der Filter ist anwendbar auf folgende Spalten:

- Verbindungsname
- Startpunkt / Start-Teilnehmer
- Endpunkt / End-Teilnehmer

Parameter

Soweit Parameter bereits durch die Projektierung belegt sind, werden die Werte in die jeweiligen Spalten übernommen.

- **Name**
  Sie können den vorbelegten Namen des Verbindungsabschnitts zwischen zwei Teilnehmern anpassen.
  Siehe hierzu Kapitel Telecontrol-Verbindungen anlegen (Seite 1827).
- **Startpunkt**
  Wählen Sie aus der Klappliste den gewünschten Startpunkt der Verbindung aus.
  – Startpunkt einer Verbindung ist jeweils eine CPU.
- **Start-Teilnehmer**
  Stationsadresse des Startpunkts
- **Start-Schnittstelle**
  Schnittstelle des Startpunkt-Moduls, über welche die Verbindung läuft.
- **Start-Schnittstelle (red.)**
  Schnittstelle des Startpunkt-Moduls, über welche die redundante Verbindung läuft.

- **Endpunkt**
  Wählen Sie den Endpunkt der Verbindung aus.
  Endpunkte einer Verbindung können sein:
  - Eine CPU
  - Ein Fremdgerät
    Den Netzknotentyp von Fremdgeräten projektieren Sie in der Parametergruppe der Verbindung, siehe Kapitel Fremdgerät-Parameter (Seite 1847).
- **End-Teilnehmer**
  Stationsadresse des Endpunkts
- **End-Schnittstelle**
  Schnittstelle des Endpunkt-Moduls, über welche die Verbindung läuft.
  Bei einem Fremdgerät, das nicht im STEP 7-Projekt projektiert ist, müssen Sie die IP-Adresse manuell eintragen.
- **End-Schnittstelle (red.)**
  Schnittstelle des Endpunkt-Moduls, über welche die redundante Verbindung läuft.
- **Partnerliste**
  Bei Auswahl eines Partners (Endpunkts), der im STEP 7-Projekt liegt, wird dessen Stationsadresse automatisch in die Partnerliste eingetragen.

Hinweis

Manuelle Eintragungen

In folgenden Fällen müssen Sie die Stationsadressen manuell nachtragen:

- **Fremdgerät**
  Bei einem Fremdgerät, das nicht im STEP 7-Projekt projektiert ist, müssen Sie die Stationsadresse manuell eintragen.
- **Knotenstation**
  Die Partnerliste eines Verbindungsabschnitts, der Daten von bzw. zu weiteren Stationen überträgt, muss alle Partner enthalten. Dies ist bei Verbindungsabschnitten der Fall, bei denen ein Teilnehmer eine Knotenstation ist. Hier müssen Sie die Stationsadressen der weiteren Partner manuell nachtragen.
  Die Stationsadressen werden Komma-separiert eingetragen.

Beispiel für eine Knotenstation, siehe auch Abbildung in Kapitel AUTOHOTSPOT. Sie projektieren drei Stationen:

- Master
  Name: "15_4", Stationsadresse: "1"
- Knotenstation
  Name: "15_3_node", Stationsadresse: "2"
- Station
  Name: "15_6", Stationsadresse: "3"

Der Verbindungsabschnitt "_5" läuft vom Master zur Knotenstation. Die Stationsadresse "2" der Knotenstation wird automatisch in die Partnerliste eingetragen.
Da über den Verbindungsabschnitt aber auch die Daten von und zur Station "15_6" übertragen werden sollen, müssen Sie die Stationsadresse "3" manuell nachtragen, vergleiche Abbildung.
Gleiches gilt für den Verbindungsabschnitt "_6" zwischen Station und Knotenstation, über den auch die Daten vom und zum Master übertragen werden sollen.

- **Port-Nr.**
  Relevant für Fremdgerät (Master / Station)
  Nummer des Listener-Ports des Partners
  Bei Modulen des STEP 7-Projekts wird der Wert aus der Projektierung übernommen. Er ist änderbar.
  Bei Fremdgeräten müssen Sie die Portnummer eintragen.
  Wertebereich: 0 ... 65535
  Vorbelegung: 20000
- **Partnerüberwachung**
  Relevant für alle Teilnehmertypen
  Wenn das Stations-Modul innerhalb der projektierten Zeit kein Lebenszeichen vom Master auf Applikationsebene empfängt, stuft es die Verbindung als gestört ein und baut die Verbindung ab.
  Das Master-Modul erwartet nach dem Senden von Daten innerhalb der projektierten Zeit eine Antwort von der Station.
  Bei 0 (Null) ist die Funktion abgeschaltet.
  Siehe Parameter "Partnerüberwachungszeit", Kapitel Übertragungseinstellungen - DNP3 (Seite 1906).
- **Transport-Prot.**
  Relevant für alle Teilnehmertypen
  Wählen Sie das Transport-Protokoll aus:
  - TCP
  - UDP
- **Spontan**
  Übertragungsmodus für Ereignisse
  Bei Modulen des STEP 7-Projekts wird der Wert aus der Projektierung übernommen und ist nicht änderbar.
  Zum Parameter siehe Kapitel Einstellungen DNP3-Station (Seite 1909).
- **Ereignis-Polling**
  Relevant für Master, Fremdgerät (Master)
  Das "Ereignis-Polling-Intervall" legt fest, in welchem Zyklus Ereignisse der Station von der DNP3-Zentrale abgefragt werden. Das Intervall wird als Vielfaches des "Polling-Basisintervall" des Masters angegeben, siehe Kapitel Einstellungen DNP3-Master (Seite 1909).
- **Klasse-0-Polling-Intervall**
  Relevant für Master, Fremdgerät (Master)
  Das Klasse-0-Polling-Intervall legt fest, in welchem Zyklus Klasse-0-Daten aus dem Abbildspeicher der Station durch Generalabfragen von der DNP3-Zentrale abgefragt werden. Das Intervall wird als Vielfaches des Parameters "Polling-Basisintervall" des Masters angegeben, siehe Kapitel Einstellungen DNP3-Master (Seite 1909).
  Der bei der Station projektierte Wert wird an die Zentrale übermittelt und dort gespeichert.
  Wertebereich: 0 ... 65535
  Vorbelegung: 1
  Bei 0 (Null) ist die Funktion abgeschaltet, Klasse-0-Daten werden nicht zyklisch übertragen.

- **Max. Polling-Dauer**
  Relevant für Master, Fremdgerät (Master)
  Gibt die maximale Zeitspanne an, welche die Zentrale diese Station durchgehend aufrufen darf. Auch wenn in der Station nach dieser Zeit noch Daten zur Übertragung anstehen, werden die Aufrufe der Zentrale abgebrochen. Damit steht die Zentrale wieder anderen Stationen zur Verfügung.
  Der bei der Station projektierte Wert wird an die Zentrale übermittelt und dort gespeichert.
  Wertebereich: 0 ... 65535
  Vorbelegung: 10
  Bei 0 (Null) ist die Funktion abgeschaltet, d. h. die Aufrufzeitspanne ist unbegrenzt.
- **Polling-Modus**
  Relevant für Master, Fremdgerät (Master)
  Hier legen Sie den Modus fest, nach dem die Zentrale die Station aufruft.
  Der bei der Station projektierte Wert wird an die Zentrale übermittelt und dort gespeichert.
  Wertebereiche:
  - Zyklisch
    Die Station wird zyklisch aufgerufen. Die Zeitdauer des Polling-Zyklus wird aus dem Parameter "Klasse-0-Polling-Intervall" berechnet, siehe oben.
  - Nach Anlauf
    Die Station wird nur nach dem ersten Anlaufen und nach einem Neustart aufgerufen. Wenn für eine Station keine spontane Übertragung aktiviert ist, werden bei Auswahl dieser Option keine Daten im laufenden Betrieb übertragen.
- **Konformitätsstufe**
  Relevant für alle Teilnehmertypen
  Angabe der vom Partner unterstützten DNP3-Konformitätsstufe (DNP3 implementation level): Level 1, 2, 3, 4 und Level 4+
  Die als Level 4+ bezeichnete Konformitätsstufe beinhaltet den Funktionsumfang von Level 4 und zusätzlich die Unterstützung weiterer Datentypen / Variationen, vergleiche Datenpunkttypen im Kapitel Datenpunkttypen (Seite 1857).
- **Temporär**
  Relevant für Partner mit temporären Verbindungen (bspw. RTU)
  Aktivieren Sie die Option, wenn der Partner temporäre Verbindungen unterstützt, die er selbst abbauen kann. Der Verbindungsabbau durch den Partner wird dann nicht als Verbindungsabbruch interpretiert.
  Der Partner kann beispielsweise eine RTU3000C sein, bei welcher die Option "Temporäre Verbindung" aktiviert ist.
- **Polling-Basisintervall**
  Relevant für Master, Fremdgerät (Master)
  Basisintervall für die Dauer des Polling-Zyklus der Zentrale (siehe "Polling-Modus" oben) und zur Berechnung der Parameter "Ereignis-Polling-Intervall" sowie "Klasse-0-Polling-Intervall".
  Zum Parameter siehe Kapitel Einstellungen DNP3-Master (Seite 1909).
- **Max. Anzahl Ereignisse pro Abfrage**
  Relevant für Master, Fremdgerät (Master)
  Maximale Anzahl an Ereignissen, die nach einem Aufruf durch die Zentrale im Antworttelegramm der Station gesendet werden dürfen. (Bei 0 keine Begrenzung).
  Zum Parameter siehe Kapitel Einstellungen DNP3-Master (Seite 1909).

- **Übertragungsform**
  Relevant für Station, Fremdgerät (Station)
  Legt die Übertragungsreihenfolge von Ereignissen an die DNP3-Zentrale fest.
  - Typ-spezifisch
    Gebündelte Übertragung nach Datentypen
  - Chronologisch
    Chronologische Übertragung nach Zeitstempel

  Zum Parameter siehe Kapitel Übertragungseinstellungen - DNP3 (Seite 1906).
- **Partner-Adresse**
  Relevant für Fremdgerät (Master / Station)
  IP-Adresse (Ethernet) oder Telefonnummer (Wählnetz) des Partners

Parameter für redundante Verbindungswege

Im Fall, dass redundante Verbindungswege projektiert sind, werden diese ebenso projektiert wie die Hauptwege.

Die Parameter der redundanten Verbindungswege sind ausgezeichnet durch den folgenden Suffix:

- *** (red)**

Die Parameter der redundanten Verbindungswege haben die entsprechenden Funktionen wie diejenigen für den Hauptweg. Zur Bedeutung siehe oben.

Verbindungstabelle - IEC

Filter

Die erste Zeile unter dem Tabellenkopf enthält eine Filterfunktion, mit der Sie die Auswahl der projektierbaren Teilnehmer und Verbindungsmöglichkeiten einschränken können. Dadurch werden die Kombinationsmöglichkeiten verringert und die Übersichtlichkeit erhöht.

Wenn Sie einige Verbindungsabschnitte angelegt haben, aktivieren Sie den Filter, indem Sie einen sich wiederholenden Namen oder Teilnamen in die Filterzelle eintippen. Die Zelle wird farbig hinterlegt, vergleiche Abbildung im Kapitel Verbindungen projektieren (Seite 1836).

Beispiel:
Sie haben Verbindungen mit den Startpunkten "1200", "PLC_1" und "PLC_2" angelegt.
Bei Eingabe von "PLC_" in die Filterzelle werden in der Tabelle nur die Abschnitte eingeblendet, deren Startpunkte mit diesem Teil-String beginnen: "PLC_1" und "PLC_2"

Gesetzte Filter in mehreren Spalten multiplizieren sich.

Die Auswahl "*" blendet alle vorhandenen Verbindungsabschnitte ein.

Der Filter ist anwendbar auf folgende Spalten:

- Verbindungsname
- Startpunkt / Start-Teilnehmer
- Endpunkt / End-Teilnehmer

Parameter

Soweit Parameter bereits durch die Projektierung belegt sind, werden die Werte in die jeweiligen Spalten übernommen.

- **Name**
  Sie können den vorbelegten Namen des Verbindungsabschnitts zwischen zwei Teilnehmern anpassen.
  Siehe hierzu Kapitel Telecontrol-Verbindungen anlegen (Seite 1827).
- **Startpunkt**
  Wählen Sie aus der Klappliste den gewünschten Startpunkt der Verbindung aus.
  - Startpunkt einer Verbindung ist jeweils eine CPU.
- **Start-Teilnehmer**
  Stationsadresse des Startpunkts
- **Start-Schnittstelle**
  Schnittstelle des Startpunkt-Moduls, über welche die Verbindung läuft.
- **Endpunkt**
  Wählen Sie den Endpunkt der Verbindung aus.
  Endpunkte einer Verbindung können sein:
  - Eine CPU
  - Ein Fremdgerät
    Den Netzknotentyp von Fremdgeräten projektieren Sie in der Parametergruppe der Verbindung, siehe Kapitel Fremdgerät-Parameter (Seite 1847).
- **End-Teilnehmer**
  Stationsadresse des Endpunkts
- **End-Schnittstelle**
  Schnittstelle des Endpunkt-Moduls, über welche die Verbindung läuft.
  Bei einem Fremdgerät, das nicht im STEP 7-Projekt projektiert ist, müssen Sie die IP-Adresse manuell eintragen.

- **Partnerliste**
  Bei Auswahl eines Partners (Endpunkts), der im STEP 7-Projekt liegt, wird dessen Stationsadresse automatisch in die Partnerliste eingetragen.

Hinweis

Manuelle Eintragungen

In folgenden Fällen müssen Sie die Stationsadressen manuell nachtragen:

- **Fremdgerät**
  Bei einem Fremdgerät, das nicht im STEP 7-Projekt projektiert ist, müssen Sie die Stationsadresse manuell eintragen.
- **Knotenstation**
  Die Partnerliste eines Verbindungsabschnitts, der Daten von bzw. zu weiteren Stationen überträgt, muss alle Partner enthalten. Dies ist bei Verbindungsabschnitten der Fall, bei denen ein Teilnehmer eine Knotenstation ist. Hier müssen Sie die Stationsadressen der weiteren Partner manuell nachtragen.
  Die Stationsadressen werden Komma-separiert eingetragen.

Beispiel für eine Knotenstation, siehe auch Abbildung in Kapitel AUTOHOTSPOT. Sie projektieren drei Stationen:

- Master
  Name: "15_4", Stationsadresse: "1"
- Knotenstation
  Name: "15_3_node", Stationsadresse: "2"
- Station
  Name: "15_6", Stationsadresse: "3"

Der Verbindungsabschnitt "_5" läuft vom Master zur Knotenstation. Die Stationsadresse "2" der Knotenstation wird automatisch in die Partnerliste eingetragen.
Da über den Verbindungsabschnitt aber auch die Daten von und zur Station "15_6" übertragen werden sollen, müssen Sie die Stationsadresse "3" manuell nachtragen, vergleiche Abbildung.
Gleiches gilt für den Verbindungsabschnitt "_6" zwischen Station und Knotenstation, über den auch die Daten vom und zum Master übertragen werden sollen.

- **Port-Nr.**
  Relevant für Fremdgerät (Master / Station)
  Nummer des Listener-Ports des Partners
  Bei Modulen des STEP 7-Projekts wird der Wert aus der Projektierung übernommen. Er ist änderbar.
  Bei Fremdgeräten müssen Sie die Portnummer eintragen.
  Wertebereich: 0 ... 65535
  Vorbelegung: 2404

- **Partnerüberwachung**
  Relevant für alle Teilnehmertypen
  Wenn das Stations-Modul innerhalb der projektierten Zeit kein Lebenszeichen vom Master auf Applikationsebene empfängt, stuft es die Verbindung als gestört ein und baut die Verbindung ab.
  Das Master-Modul erwartet nach dem Senden von Daten innerhalb der projektierten Zeit eine Antwort von der Station.
  Bei 0 (Null) ist die Funktion abgeschaltet.
  Siehe Parameter "Partnerüberwachungszeit", Kapitel IEC-spezifische Parameter (Seite 1917).
- **Spontan**
  Übertragungsmodus für Ereignisse
  Bei Modulen des STEP 7-Projekts wird der Wert aus der Projektierung übernommen und ist nicht änderbar.
  Zum Parameter siehe Kapitel IEC-spezifische Parameter (Seite 1917).
- **Polling-Modus**
  Relevant für Master, Fremdgerät (Master)
  Hier legen Sie den Modus fest, nach dem die Zentrale die Station aufruft.
  Der bei der Station projektierte Wert wird an die Zentrale übermittelt und dort gespeichert.
  Wertebereiche:
  - Zyklisch
    Die Station wird zyklisch aufgerufen. Die Zeitdauer des Polling-Zyklus wird aus dem Parameter "Klasse-0-Polling-Intervall" berechnet, siehe oben.
  - Nach Anlauf
    Die Station wird nur nach dem ersten Anlaufen und nach einem Neustart aufgerufen. Wenn für eine Station keine spontane Übertragung aktiviert ist, werden bei Auswahl dieser Option keine Daten im laufenden Betrieb übertragen.
- **Temporär**
  Relevant für Partner mit temporären Verbindungen (bspw. RTU)
  Aktivieren Sie die Option, wenn der Partner temporäre Verbindungen unterstützt, die er selbst abbauen kann. Der Verbindungsabbau durch den Partner wird dann nicht als Verbindungsabbruch interpretiert.
  Der Partner kann beispielsweise eine RTU3000C sein, bei welcher die Option "Temporäre Verbindung" aktiviert ist.

Parameter für redundante Verbindungswege

Im Fall, dass redundante Verbindungswege projektiert sind, werden diese ebenso projektiert wie die Hauptwege.

Die Parameter der redundanten Verbindungswege sind ausgezeichnet durch den folgenden Suffix:

- *** (red)**

Die Parameter der redundanten Verbindungswege haben die entsprechenden Funktionen wie diejenigen für den Hauptweg. Zur Bedeutung siehe oben.

Fremdgerät-Parameter

Fremdgerät-Parameter

Nur gültig für Partner, die nicht im STEP 7-Projekt projektiert werden.

- **Partner-Stationsadresse /**
  Stationsadresse (red.)
  Stationsadresse (ASDU-Adresse) des Fremdgeräts, das über eine Verbindung bzw. über einen redundanten Verbindungsweg erreichbar ist.
- **Netzknotentyp Fremdgerät /**
  Netzknotentyp Fremdgerät (red.)
  Legen Sie den Netzknotentyp des Fremdgeräts, das über eine Verbindung bzw. über einen redundanten Verbindungsweg erreichbar ist, fest:
  - Zentrale-Station
    (Master)
  - Knotenstation
    Bei Modulen, die als Knotenstation fungieren, gilt:
    Die Schnittstelle in Richtung Zentrale wird als "Knotenstation" projektiert.
    Die Schnittstelle in Richtung unterlagertes Netz wird als "Zentrale" projektiert.
  - Station

Optionen (IEC)

Die folgenden Parameter finden Sie in den Parametergruppen "Optionen 1. Weg" / "Optionen 2. Weg" der IEC-Verbindungen.

Aufruf-Intervalle

Die nachfolgenden Parameter legen für die Station die Intervalle von speziellen Aufrufen des Masters fest (cause of transmission 20 - 41).

Alle Parameter werden als Vielfaches des "Polling-Basisintervalls" projektiert (siehe Master).

- **Intervall für Generalabfrage**
  Legt das Intervall fest, mit dem Generalabfragen des Masters beantwortet werden.
- **Intervall für Gruppenabfrage**
  Legt das Intervall fest, mit dem die jeweilige Gruppenabfrage des Masters beantwortet werden.
- **Intervall für Zähler-Generalabfrage**
  Legt das Intervall fest, mit dem Zähler-Generalabfragen des Masters beantwortet werden.
- **Intervall für Zähler-Gruppenabfrage**
  Legt das Intervall fest, mit dem die jeweilige Zähler-Gruppenabfrage des Masters beantwortet werden.

Die Einstellung, ob auf eine Generalabfrage geantwortet wird, und die Zuordnung zu einer Gruppenabfrage legen Sie für jeden einzelnen Datenpunkt in der Datenpunktprojektierung fest.

Datenpunkt-Editor

Projektierung der Datenpunkte und Nachrichten

Vorbemerkung: TIM-Baugruppen mit SINAUT ST7

TIM-Baugruppen, die das Fernwirkprotokoll SINAUT ST7 nutzen, können für die Abwicklung der Fernwirkaufgaben alternativ zu den Datenpunkten (TD7onTIM) die Bausteinbibliothek "Telecontrol ST7" (TD7onCPU) verwenden.

Dieses Kapitel gilt für die Variante "TD7onTIM".

Datenpunkt-bezogene Kommunikation mit der CPU

Zur Übertragung von Nutzdaten zwischen Station und Kommunikationspartner ist bei den Telecontrol-Modulen mit Datenpunktprojektierung keine Programmierung von Programmbausteinen erforderlich. Die Datenbereiche im Speicher der CPU, welche für die Kommunikation mit dem Kommunikationspartner vorgesehen sind, werden Datenpunkt-bezogen in der Baugruppe projektiert. Dabei ist jeder Datenpunkt mit einer PLC-Variable oder der Variable eines Datenbausteins verknüpft.

Voraussetzung: Angelegte PLC-Variablen und/oder Datenbausteine (DBs)

Voraussetzung für die Projektierung der Datenpunkte sind die entsprechend angelegten PLC-Variablen bzw. DBs in der CPU.

Die PLC-Variablen für die Datenpunktprojektierung können in der Standard-Variablentabelle oder in einer benutzerdefinierten Variablentabelle angelegt werden. Alle PLC-Variablen, welche für die Datenpunktprojektierung verwendet werden sollen, müssen mit dem Attribut "Sichtbar in HMI" ausgezeichnet sein.

Adressbereiche der PLC-Variablen sind Eingangs-, Ausgangs- oder Merkerbereiche in der CPU.

Die Formate und S7-Datentypen der PLC-Variablen, welche mit den protokollspezifischen Datenpunkttypen des CP kompatibel sind, finden Sie im Kapitel Datenpunkttypen (Seite 1857).

Zugriff auf die Speicherbereiche der CPU

Die Werte der von den Datenpunkten referenzierten PLC-Variablen bzw. DBs werden gelesen und durch den CP an den Kommunikationspartner übertragen.

Vom Kommunikationspartner empfangene Daten werden durch den CP über die PLC-Variablen bzw. DBs in die CPU geschrieben.

Projektierung der Datenpunkte und Nachrichten in STEP 7

Die Projektierung der Datenpunkte nehmen Sie in STEP 7 im Datenpunkt- und Nachrichten-Editor vor. Sie können die beiden Editoren alternativ öffnen über:

- Selektion der Kommunikationsbaugruppe in der Netzsicht oder in der Gerätesicht Kontextmenü "Datenpunkt- und Nachrichten-Editor öffnen"
- Über die Projektnavigation:
  Projekt > Verzeichnis der Station > Lokale Module > gewünschte Kommunikationsbaugruppe

Bild 1-76 Öffnen des Datenpunkt- und Nachrichten-Editors

Durch Doppelklick auf den Eintrag öffnet sich der Datenpunkt- bzw. Nachrichten-Editor.

Nach Öffnen des Editorfensters können Sie über die beiden Einträge rechts oben über der Tabelle zwischen dem Datenpunkt- bzw. Nachrichten-Editor umschalten.

Bild 1-77 Umschaltung zwischen den zwei Editoren

Anlegen von Objekten

Bei geöffnetem Datenpunkt- bzw. Nachrichten-Editor legen Sie ein neues Objekt (Datenpunkt / Nachricht) an, indem Sie in die erste Tabellenzeile mit dem gegrauten Eintrag "<Objekt hinzufügen>" doppelklicken.

Ein vorbelegter Name wird in die Zelle geschrieben. Den Namen können Sie nach Ihren Bedürfnissen anpassen, er muss aber innerhalb des CP eindeutig sein.

		Name	PLC-Variable
1		DataPoint	"Tag_1-BI"
2		DataPoint_1	"Tag_2-BQ"
3		DataPoint_2	"Tag_1-BI"

Bild 1-78 Datenpunkt-Tabelle

Die weiteren Eigenschaften eines jeden Objekts projektieren Sie über die Klapplisten der weiteren Tabellenspalten und über die unten eingeblendeten Parameterfelder.

Datenpunkte ihrer Datenquelle zuordnen

Einen neuen Datenpunkt ordnen Sie anschließend seiner Datenquelle zu. Je nach Datentyp des Datenpunkts kommt als Datenquelle eine PLC-Variable in Betracht.

Für die Zuordnung haben Sie folgende Möglichkeiten:

- Klicken Sie auf das Tabellensymbol in der Zelle der Spalte "PLC-Tag".
  Alle projektierten PLC-Variablen und die Variablen der angelegten Datenbausteine werden eingeblendet. Wählen Sie mit der Maus oder der Tastatur die gewünschte Datenquelle aus.
- Klicken Sie auf das Symbol ...
  Eine Auswahlliste der projektierten PLC-Tags und der Bausteine wird aufgeblendet. Wählen Sie aus der entsprechenden Tabelle die gewünschte Datenquelle aus.
- Geben Sie in dem Namensfeld der PLC-Variable einen Teil des Namens der gewünschten Datenquelle ein.
  Alle projektierten PLC-Variablen und Variablen der Datenbausteine, deren Namen die eingegebenen Buchstaben enthalten, werden eingeblendet.

  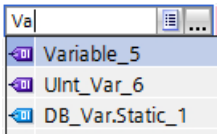

  Wählen Sie die gewünschte Datenquelle aus.

Hinweis

Zuordnung von Parameterwerten zu PLC-Variablen

Die hier beschriebenen Mechanismen gelten auch, wenn Sie den Wert eines Parameters einer PLC-Variable zuordnen müssen. Die Eingabefelder für die PLC-Variable (Bsp.: PLC-Variable für Partnerstatus, unterstützen die hier beschriebenen Funktionen zur Auswahl der PLC-Variable.

Anordnen von Spalten und Zeilen, Ein-/Ausblenden von Spalten

Wie bei vielen anderen Programmen können Sie auch im Datenpunkt- bzw. Nachrichten-Editor die Spalten anordnen und die Tabelle nach Ihren Bedürfnissen sortieren:

- Spalten anordnen
  Wenn Sie auf einen Spaltenkopf mit gedrückter linker Maustaste klicken, können Sie die Spalte verschieben.
- Objekte sortieren
  Wenn Sie kurz mit der linken Maustaste auf einen Spaltenkopf klicken, können Sie die Objekte der Tabelle aufsteigend bzw. absteigend nach den Einträgen dieser Spalte sortieren. Die Sortierung wird über einen Pfeil im Spaltenkopf angezeigt.
  Nach absteigender Sortierung einer Spalte lässt sich die Sortierung durch wiederholten Klick auf den Spaltenkopf wieder ausschalten.
- Spaltenbreite anpassen
  Diese Funktion erreichen Sie über folgende Aktionen:
  - Über das Kontextmenü, das sich bei Klicken mit der rechten Maustaste auf einen Spaltenkopf öffnet:
    "Breite optimieren", "Breite aller Spalten optimieren"
  - Wenn Sie den Cursor in die Nähe der rechten Begrenzung eines Spaltenkopfs führen, erscheint das folgende Symbol:

    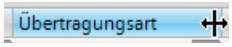

    Doppelklicken Sie in diesem Moment auf den Spaltenkopf. Die Spaltenbreite passt sich dem breitesten Eintrag in dieser Spalte an.
- Spalten ein-/ausblenden
  Diese Funktion erreichen Sie über das Kontextmenü, das sich bei Klicken mit der rechten Maustaste auf einen Spaltenkopf öffnet.

Datenpunkte und Nachrichten kopieren

Wie bei vielen anderen Programmen können Sie auch im Datenpunkt- bzw. Nachrichten-Editor Objekte kopieren und einfügen.

Wenn Sie mit der rechten Maustaste in die Zeile eines Objekts in der Tabelle klicken, dann erreichen Sie die genannten Funktion über das Kontextmenü:

- Ausschneiden
- Kopieren
- Einfügen
  Einfügen können Sie ausgeschnittene oder kopierte Objekte innerhalb der Tabelle oder in der ersten freien Zeile unterhalb der Tabelle.
  Sie können ausgeschnittene oder kopierte Objekte auch in Tabellen anderer Kommunikationsmodule vom gleichen Typ und mit gleichem Telecontrol-Protokoll einfügen.
- Löschen

Bei gedrückter <Strg>-Taste können Sie mehrere Zeilen selektieren, die nicht zusammenhängen.

Bei gedrückter <Shift>-Taste können Sie den Anfang und das Ende eines zusammenhängenden Bereichs selektieren.

Automatisches Ausfüllen von Zellen (nur ST7-Datenpunkte)

Das Automatische Ausfüllen von Zellen steht bei ST7-Datenpunkten in folgenden Funktionsvarianten zur Verfügung:

- **Überschreiben**
- **Einfügen**

Das automatische Ausfüllen bietet sich insbesondere für folgende Anwendungsfälle an:

- Sie haben ST7-Datenpunkte kopiert.
  Sie möchten die Zellen einer Spalte mit dem gleichen Wert oder fortlaufend auf- bzw. absteigend befüllen.
- Sie haben ST7-Datenpunkte angelegt.
  Sie möchten weitere Datenpunkte vom gleichen Typ hinzufügen.

Gehen Sie zum automatischen Ausfüllen folgendermaßen vor:

1. Selektieren Sie für die Funktion eine oder mehrere Zellen (Start-Zellen) einer Spalte.
2. Bewegen Sie den Cursor auf die rechte untere Ecke der Auswahl und ziehen Sie den Cursor nach unten auf die Ziel-Zellen.

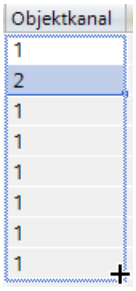

   Es öffnet sich ein Dialogfeld, in dem Sie die gewünschte Funktionsvariante auswählen.

3. Selektieren Sie die gewünschte Option "Überschreiben" oder "Einfügen" und klicken Sie auf "OK".

Überschreiben

Gehen Sie je nach gewünschtem Inhalt der Ziel-Zellen folgendermaßen vor:

- Selektieren Sie eine einzige Start-Zelle, wenn Sie die Ziel-Zellen mit dem Wert der Start-Zelle überschreiben möchten.
- Selektieren Sie mindestens zwei Start-Zellen mit numerisch aufsteigenden Werten, wenn Sie die Ziel-Zellen mit numerisch aufsteigenden Werten überschreiben möchten. Diese Variante ist oben unter Schritt 2 abgebildet.
  Wenn Sie dagegen zwei Zellen mit unterschiedlichem Textinhalt selektieren (bspw. PLC-Variable Bool[1] und DWord[1]), dann werden die Ziel-Zellen nur mit dem ersten Wert überschrieben.
  Sie selektieren mehrere Zellen wie beim Kopieren:
  - <Strg>-Taste: Selektion mehrerer Zellen. Die Zellen müssen aufeinander folgen.
  - <Shift>-Taste: Selektion eines zusammenhängenden Bereichs
- Selektieren Sie mehrere Start-Zellen mit einem beliebigen Werteblock, wenn Sie die Ziel-Zellen mit diesem Werteblock überschreiben möchten.
  Beispiel: Sie selektieren drei Zellen mit den Werten 4, 12 und 7. Die Ziel-Zellen werden dann fortlaufend mit Blöcken von 4, 12 und 7 überschrieben.

Ausnahme: Beim Namen der Datenpunkte gelten besondere Regeln, siehe unten.

Bei Selektion der Start-Zellen gemäß der Abbildung oben werden die Ziel-Zellen nach dem Ziehen des Cursors mit folgenden Werten überschrieben.

Objektkanal
1
2
3
4
5
6
7
8

Sie können Ziel-Zellen auch numerisch absteigend überschreiben, wenn die beiden Start-Zellen absteigende Werte enthalten oder wenn Sie zwei Zellen mit den höchsten Nummern selektieren und den Cursor aufwärts ziehen. Bei absteigendem Überschreiben ist der kleinste Wert die 1, da 0 ein unzulässiger Wert ist.

Das Überschreiben kann in folgenden Spalten angewendet werden:

- Name
  Beim Überschreiben der Namen von Datenpunkten werden keine fortlaufenden Namen vergeben. Es wird aber für jeden Datenpunkt ein eindeutiger Name vergeben.
- PLC-Variable
  Voraussetzung: Die Ziel-Zellen sind Datenpunkttypen zugeordnet, die zur Start-Zelle kompatibel sind.
- Datenpunkttyp
  Voraussetzung: Die Datenpunkte der Ziel-Zellen verwenden die gleiche PLC-Variable.
- Übertragungsart
  Voraussetzung: Die Datenpunkte der Ziel-Zellen verwenden die gleiche PLC-Variable und haben die gleiche Übertragungsrichtung (Eingang bzw. Ausgang).

- Objektkanal
  Es gelten die Regeln für die Vergabe der Kanalnummern, siehe Kapitel ST7-Datenpunkte (Seite 1866).
- Objektnummer
  Es gelten die Regeln für die Vergabe der Objektnummern.

Einfügen

Diese Option fügt am Ende der Tabelle neue Zeilen ein.

- Selektieren Sie eine oder mehrere Zellen und ziehen Sie anschließend den Cursor auf weitere Zellen.
  Die Gesamtzahl der nach dem Ziehen des Cursors selektierten Zellen wird eingefügt.

Die Funktion wird in folgenden Spalten unterstützt:

- Name
- PLC-Variable
- Datenpunkttyp
- Übertragungsart
- Objektkanal
- Objektnummer

Datenpunkte exportieren und importieren

Um das Engineering größerer Anlagen zu erleichtern, können Sie die Datenpunkte einer projektierten Baugruppe exportieren und in weitere Baugruppen im Projekt importieren. Dies bietet Vorteile vor allem bei Projekten mit vielen gleichen oder ähnlichen Stationen bzw. Baugruppen.

Die Export-/Import-Funktion erreichen Sie alternativ:

- Über die beiden Symbole über der Datenpunktliste
- Über das Kontextmenü der Baugruppe, wenn Sie diese in der Netz- oder Geräte-Sicht selektieren.

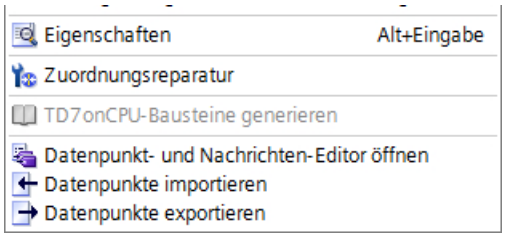

Bild 1-79 Teil des Kontextmenüs eines Telecontrol-Moduls

Die Datenpunktinformationen eines Moduls werden beim Export in genau eine CSV-Datei gespeichert.

Export

Beim Aufruf der Exportfunktion öffnet sich der Exportdialog. Hier selektieren Sie den oder die Baugruppen des Projekts, deren Datenpunktinformationen exportiert werden sollen. Bei Bedarf können Sie die Datenpunkte aller Baugruppen des Projekts auf ein Mal exportieren.

Im Exportdialog können Sie den Speicherort im Dateiverzeichnis auswählen. Wenn Sie die Daten einer Baugruppe exportieren, können Sie auch den vorbelegten Dateinamen ändern.

Beim Export aus mehreren Baugruppen werden die Dateien mit vorbelegten Namen aus Stationsname und Baugruppen-Name gebildet.

Die Datei selbst enthält neben den Datenpunktinformationen folgende Angaben:

- Baugruppen-Name
- Baugruppen-Typ
- CPU-Name
- CPU-Typ

Editieren der Export-Dateien

Sie können die Datenpunktinformationen in einer exportierten CSV-Datei editieren. Dadurch können Sie diese Datei als Projektierungsvorlage für viele andere Stationen verwenden.

Wenn Sie ein Projekt mit vielen gleichartigen Stationen haben, dann können Sie die CSV-Datei mit den Datenpunkten einer fertig projektierten Baugruppe für weitere, noch nicht projektierte Stationen kopieren und einzelne Parameter an die jeweilige Station anpassen. Damit ersparen Sie sich die Projektierung der Datenpunkte für jede Baugruppe in STEP 7. Statt dessen importieren Sie einfach die kopierte und angepasste CSV-Datei in die anderen, gleichartigen Baugruppen. Die geänderten Parameterwerte der CSV-Datei werden beim Import dieser Datei in eine andere Baugruppe in die Datenpunkt-Projektierung dieses Moduls übernommen.

Die Zeilen der CSV-Datei haben folgende Inhalte:

- Zeile 1: ,Name,Type,
  Diese Zeile darf nicht geändert werden.
- Zeile 2: PLC,<CPU-Name>,<CPU-Typ>,
  Bedeutung: PLC (Bezeichnung der Stationsklasse), CPU-Name, CPU-Typ
  Nur die Elemente <CPU-Name> und <CPU-Typ> dürfen geändert werden.
  Der CPU-Typ muss exakt dem Namen der CPU im Katalog entsprechen.
- Zeile 3: Module,<Baugruppen-Name>,<Baugruppen-Typ>,
  Bedeutung: Module (Bezeichnung der Modulklasse), Baugruppen-Typ, Baugruppen-Name
  Nur die Elemente <Baugruppen-Name> und <Baugruppen-Typ> dürfen geändert werden.
  Sein Sie sorgfältig beim Ändern der Baugruppen-Namen, wenn Sie Datenpunkte in mehrere Baugruppen importieren möchten (siehe unten).
  Der Baugruppen-Typ muss exakt dem Namen der Baugruppe im Katalog entsprechen.
- Zeile 4: Parameternamen (englisch) der Datenpunkte
  Diese Zeile darf nicht geändert werden.
- Zeilen 5..n: Werte der Parameter gemäß Zeile 4 der einzelnen Datenpunkte
  Die Parameterwerte dürfen Sie für die jeweilige Station ändern.

Import in eine Baugruppe

Stellen Sie vor dem Importieren der Datenpunkte sicher, dass die entsprechend benötigten PLC-Variablen für die Datenpunkte angelegt sind.

Beachten Sie, dass beim Import einer CSV-Datei alle in der Baugruppe vorhandenen Datenpunkte gelöscht und durch die importierten Datenpunkte ersetzt werden.

Selektieren Sie einen CP und wählen Sie die Importfunktion aus dem Kontextmenü der Baugruppe. Es öffnet sich der Importdialog, in dem Sie die gewünschte CSV-Datei im Dateiverzeichnis auswählen.

Wenn die Informationen zur Zuordnung der einzelnen Datenpunkte zu ihrer jeweiligen PLC-Variable mit der Zuordnung in der ursprünglichen Baugruppe entsprechen, werden die Datenpunkte den jeweiligen PLC-Variablen zugeordnet.

Wenn Sie Datenpunkte in einen CP importieren, aber einige benötigte PLC-Variablen in der CPU noch nicht angelegt sind, dann können die entsprechenden Datenpunktinformationen nicht zugeordnet werden. In diesem Fall können Sie fehlende PLC-Variablen nachträglich anlegen und diesen danach die importierten Datenpunktinformationen zuordnen. Hierfür steht die Funktion "Zuordnungsreparatur" zur Verfügung (siehe unten).

Wenn sich die Namen der PLC-Variablen in der Baugruppe, in welche importiert wird, von derjenigen Baugruppe, welche exportiert hat, unterscheiden, dann können die entsprechenden Datenpunkte nicht ihren PLC-Variablen zugeordnet werden.

Import in mehrere Baugruppen

Sie können die Datenpunkte von mehreren Baugruppen in die Baugruppen eines anderen Projekts importieren. Selektieren Sie hierzu im Importdialog über die Steuerungstaste alle benötigten CSV-Dateien.

Stellen Sie vor dem Importieren der Datenpunkte sicher, dass sowohl die entsprechenden Stationen mit gleichnamigen CPUs, gleichnamigen Baugruppen und die gleichnamigen PLC-Variablen angelegt sind.

Beim Import werden anhand der Baugruppen-Namen in den CSV-Dateien die entsprechenden Stationen des Projekts gesucht. Wenn eine Zielstation nicht im Projekt vorhanden ist oder die Baugruppe einen unterschiedlichen Namen hat, wird der Import der entsprechenden CSV-Datei ignoriert.

Beschränkungen des Imports von Datenpunkten

In folgenden Fällen wird der Import von Datenpunkten abgebrochen:

- Ein vom Modul benötigtes Attribut fehlt in der zu importierenden CSV-Datei.
  Beispiel: Wenn ein zu importierender Datenpunkt einen Uhrzeit-Trigger verwendet, dann wird der Import abgebrochen, wenn für das Modul keine Uhrzeitsynchronisation projektiert wurde.
- Das vom Modul verwendete Telecontrol-Protokoll unterscheidet sich vom ursprünglichen Modul.
  Module mit gleichem Telecontrol-Protokoll sind untereinander kompatibel:

Nur beim Import in mehrere Module:

- Der Import wird abgebrochen, wenn sich ein Modul- oder CPU-Name von den Daten der CSV-Datei unterscheidet.

Zuordnungsreparatur

Wenn Sie in einer Station, in die Sie importieren möchten, die PLC-Variablen anders benannt haben als in derjenigen Station, aus der die CSV-Datei exportiert wurde, dann geht die Zuordnung zwischen Datenpunkt und PLC-Variable beim Import verloren.

Sie haben dann die Möglichkeit, entweder vorhandene PLC-Variablen passend umzubenennen oder fehlende PLC-Variablen hinzuzufügen. Anschließend können Sie die Zuordnung zwischen nicht zugeordneten Datenpunkten und PLC-Variablen reparieren. Die Funktion erreichen Sie entweder über das Kontextmenü des CP (siehe oben) oder über das folgende Symbol links oben im Datenpunkteditor:

Wenn von der Reparaturfunktion zu einem Datenpunkt eine PLC-Variable mit passendem Namen gefunden wird, wird die Zuordnung wieder hergestellt. Dabei wird jedoch nicht der Datentyp der Variable geprüft.

Prüfen Sie nach der Zuordnungsreparatur auf jeden Fall, ob die neu zugeordneten PLC-Variablen richtig sind.

Datenpunkttypen

Bei der Projektierung der vom Kommunikationsmodul zu übertragenden Nutzdaten wird jeder Datenpunkt als protokollspezifischer Datenpunkttyp angelegt.

Nachfolgend werden die protokollspezifischen Datenpunkttypen mit den jeweils kompatiblen S7-Datentypen aufgelistet.

Die Spalte "Richtung" gibt die Übertragungsrichtung an:

- "in": Beobachtungsrichtung
- "out": Steuerungsrichtung

Bei ST7-Protokoll ist die Übertragungsrichtung aus dem Objektnamen ablesbar.

Hinweis

Auswirkung der Änderung von Arrays für Datenpunkte

Bei der nachträglichen Änderung eines Arrays muss der Datenpunkt neu angelegt werden.

Datenpunkttypen des Protokolls "TeleControl Basic"

Tabelle 1-126 Unterstützte Datenpunkttypen und kompatible S7-Datentypen

Format (Speicherbedarf)	Datenpunkttyp	Richtung	S7-Datentypen	Operandenbereich
Bit	Digitaleingang	in	Bool	I, Q, M, DB
	Digitalausgang	in	Bool	Q, M, DB

Format (Speicherbedarf)	Datenpunkttyp	Richtung	S7-Datentypen	Operandenbereich
Byte	Digitaleingang	in	Byte, Char, USInt	I, Q, M, DB
	Digitalausgang	out	Byte, Char, USInt	Q, M, DB
Ganzzahl mit Vorzeichen (16 Bit)	Analogeingang	in	Int	I, Q, M, DB
	Analogausgang	out	Int	Q, M, DB
Zähler (16 Bit)	Zählereingang	in	Word, UInt	I, Q, M, DB
Ganzzahl mit Vorzeichen (32 Bit)	Analogeingang	in	DInt	Q, M, DB
	Analogausgang	out	DInt	Q, M, DB
Zähler (32 Bit)	Zählereingang	in	UDInt, DWord	I, Q, M, DB
Gleitpunktzahl mit Vorzeichen (32 Bit)	Analogeingang	in	Real	Q, M, DB
	Analogausgang	out	Real	Q, M, DB
Gleitpunktzahl mit Vorzeichen (64 Bit)	Analogeingang	out	LReal	Q, M, DB
	Analogausgang	out	LReal	Q, M, DB
Datenblock (1 .. 64 Byte)	Daten	in / out	ARRAY [1)]	DB
	Daten	in / out	ARRAY [1)]	DB

[1)] Zu den möglichen Formaten des Datentyps ARRAY siehe nachfolgender Abschnitt.

Datenblock (ARRAY)

Mit dem Datentyp ARRAY können zusammenhängende Speicherbereiche bis zu einer Größe von 64 Byte übertragen werden. Kompatible Komponenten von ARRAY sind folgende S7-Datentypen:

- Byte, USInt (insgesamt bis zu 64 pro Datenblock)
- Char (insgesamt bis zu 64 pro Datenblock) - CP ab Firmware-Version 2.1.77
- Int, UInt, Word (insgesamt bis zu 32 pro Datenblock)
- DInt, UDInt, DWord (insgesamt bis zu 16 pro Datenblock)

Bei einer nachträglichen Änderung des Arrays muss der Datenpunkt neu angelegt werden.

Zeitstempel im UTC-Format

Zeitstempel werden von den OPC-Server-Applikationen im UTC-Format (48 Bit) ausgegeben und beinhalten Millisekunden.

Datenpunkttypen des Protokolls "ST7"

Die Richtung der Datenübertragung können Sie der Spalte "Objekt" entnehmen:

- Sendende Objekte haben das Suffix "_S".
- Empfangende Objekte haben das Suffix "_R".

Tabelle 1-127 Unterstützte Datenpunkttypen und kompatible S7-Datentypen

Format (Speicherbedarf)	Datenpunkttyp	S7-Datentyp	Operandenbereich	Datenpunkte (Kanäle) pro Objekt	Objekt
Bit	Digitaleingang	Bool	I, Q, M, DB	8	Bin08X_S
	Digitalausgang	Bool	Q, M, DB	8	Bin08X_R

Format (Speicherbedarf)	Datenpunkttyp	S7-Datentyp	Operandenbereich	Datenpunkte (Kanäle) pro Objekt	Objekt
Byte	Digitaleingang	Byte, USInt [3]	I, Q, M, DB	4	Bin04B_S
	Digitalausgang	Byte, USInt [3]	Q, M, DB	4	Bin04B_R
	Befehlsausgang	Byte, USInt [3]	Q, M, DB	1	Cmd01B_R
	Befehlseingang	Byte, USInt [3]	Q, M, DB	1	Cmd01B_S
Ganzzahl mit Vorzeichen (16 Bit)	Analogeingang	Int	I, Q, M, DB	4	Ana04W_S
	Mittelwert-Eingang	Int	I, Q, M, DB	4	Mean04W_S
	Analogausgang	Int	Q, M, DB	4	Ana04W_R
	Mittelwert-Ausgang	Int	Q, M, DB	4	Mean04W_R
	Sollwert-Ausgang [1]	Int	Q, M, DB	1	Set01W_R
	Sollwert-Eingang [1]	Int	Q, M, DB	1	Set01W_S
Zähler (16 Bit)	Zählereingang	UInt, Word [3]	I, Q, M, DB	1	Cnt01D_S
	Zählereingang	UInt, Word [3]	I, Q, M, DB	4	Cnt04D_S
	Zählerausgang	UDInt, DWord [3]	I, Q, M, DB	1	Cnt01D_R
	Zählerausgang	UDInt, DWord [3]	I, Q, M, DB	4	Cnt04D_R
Gleitpunktzahl (32 Bit)	Analogeingang	Real [3]	M, DB	4	Ana04R_S
	Analogausgang	Real [3]	Q, M, DB	4	Ana04R_R
Datenblock (4 .. 48 Byte)	Daten-Eingang	ARRAY [1...12] of DInt / UDInt / DWord / Real [2] [3]	DB	12	Dat12D_S
	Daten-Ausgang		DB	12	Dat12D_R
	Parameter-Ausgang [1]		DB	12	Par12D_R
	Parameter-Eingang [1]		DB	12	Par12D_S

[1] Siehe unten, Abschnitt "1: Rückspiegelung"

[2] Siehe unten, Abschnitt "2: Datenblock"

[3] USInt, UInt, Real und UDInt werden von S7-300/400 nicht unterstützt.

1: Rückspiegelung (TIM 3V-IE / TIM 4R-IE)

Die Rückspiegelung der aktuellen lokalen Daten an den Partner können Sie bei folgenden Datenpunkttypen projektieren:

- Sollwert-Ein-/Ausgang (Set01W_R / Set01W_S)
- Parameter-Ein-/Ausgang (Par12D_R / Par12D_S)

Bei Aktivierung der Option "Sollwert-Eingabemodus des Partners" werden die lokalen Werte auf Änderungen überwacht und bei Wertänderung an den Partner übertragen.

Die Funktion projektieren Sie über die Parameter des Registers "Lokale Sollwerteingabe", siehe Kapitel Register "Lokale Sollwerteingabe" (Seite 1872).

2: Datenblock

Mit dem Datentyp ARRAY können Datenblöcke aus zusammenhängenden Speicherbereichen mit einer Größe von 4 .. 48 Byte übertragen werden.

Kompatible Komponenten von ARRAY sind DInt, UDInt, DWord oder Real Die Komponenten innerhalb eines Arrays müssen gleichartig sein.

Format der Zeitstempel

Zum Format der Zeitstempel siehe Handbuch der Baugruppe.

Datenpunkttypen des Protokolls "DNP3"

Tabelle 1-128 Unterstützte Datenpunkttypen, DNP3-Objektgruppen, Varianten und kompatible S7-Datentypen

Format (Speicherbedarf)	Datenpunkttyp CP [Datenpunkttyp TIM]	DNP3-Objektgruppe [variations]	Richtung	S7-Datentypen		Operandenbereich
				CP	TIM	
Bit	Binary Input	**1** [1, 2]	in	Bool	Bool	I, Q, M, DB
	Binary Input Event	**2** [1, 2]	in	Bool	Bool	I, Q, M, DB
	Binary Output [1]	**10** [2]	out	Bool	Bool	Q, M, DB
	Binary Output Event [1]	**11** [1, 2]	out	Bool	Bool	Q, M, DB
	Binary Command	**12** [1]	out	Bool	Byte	Q, M, DB
Integer (16 Bit)	Counter	**20** [2]	in	UInt, Word	Word	I, Q, M, DB
	Frozen Counter [2]	**21** [2, 6]	in	UInt, Word	Word	I, Q, M, DB
	Counter Event	**22** [2, 6]	in	UInt, Word	Word	I, Q, M, DB
	Frozen Counter Event [3]	**23** [2, 6]	in	UInt, Word	Word	I, Q, M, DB
	Analog Input	**30** [2]	in	Int	Int	I, Q, M, DB
	Analog Input Event	**32** [2]	in	Int	Int	I, Q, M, DB
	Analog Output Status [4]	**40** [2]	out	Int	Int	Q, M, DB
	Analog Output	**41** [2]	out	Int	Int	Q, M, DB
	Analog Output Event [4]	**42** [2, 4]	out	Int	Int	Q, M, DB
Integer (32 Bit)	Counter	**20** [1]	in	DWord	DWord	I, Q, M, DB
	Frozen Counter [2]	**21** [1, 5]	in	DWord	DWord	I, Q, M, DB
	Counter Event	**22** [1, 5]	in	DWord	DWord	I, Q, M, DB
	Frozen Counter Event [3]	**23** [1, 5]	in	DWord	DWord	I, Q, M, DB
	Analog Input	**30** [1]	in	DInt	-	Q, M, DB
	Analog Input Event	**32** [1]	in	DInt	-	Q, M, DB
	Analog Output Status [4]	**40** [1, 3]	out	DInt	DWord	Q, M, DB
	Analog Output	**41** [1]	out	DInt	DWord	Q, M, DB
	Analog Output Event [4]	**42** [1]	out	DInt	DWord	Q, M, DB
Gleitpunktzahl (32 Bit)	Analog Input	**30** [5]	in	Real	-	Q, M, DB
	Analog Input Event	**32** [5, 7]	in	Real	-	Q, M, DB
	Analog Output Status [4]	**40** [3]	out	Real	-	Q, M, DB
	Analog Output	**41** [3]	out	Real	-	Q, M, DB
	Analog Output Event [4]	**42** [5, 7]	out	Real	-	Q, M, DB
Gleitpunktzahl (64 Bit)	Analog Input	**30** [6]	in	LReal	-	Q, M, DB
	Analog Input Event	**32** [6, 8]	in	LReal	-	Q, M, DB
	Analog Output	**41** [4]	out	LReal	-	Q, M, DB
	Analog Output Event [4]	**42** [6, 8]	out	LReal	-	Q, M, DB
Datenblock (1...64 Byte) [5]	Octet String / Octet String Output	**110** [-]	in, out	[5]	[5]	DB
	Octet String Event [5]	**111** [-]	in, out	[5]	[5]	DB

[1] Diese Objektgruppe ist im Datenpunkteditor von STEP 7 projektierbar über die Ersatz-Objektgruppe 12.

[2] Diese Objektgruppe ist im Datenpunkteditor von STEP 7 projektierbar über die Ersatz-Objektgruppe 20.

[3] Diese Objektgruppe ist im Datenpunkteditor von STEP 7 projektierbar über die Ersatz-Objektgruppe 22.

[4] Diese Objektgruppe ist im Datenpunkteditor von STEP 7 projektierbar über die Ersatz-Objektgruppe 41.

[5] Mit diesen Datenpunkttypen können zusammenhängende Speicherbereiche bis zu einer Größe von 64 Byte übertragen werden. Kompatibel sind alle S7-Datentypen mit einer Größe zwischen 1 und 64 Byte.

Zu Tabellenfußnoten [1], [2], [3], [4]: Projektierung von Datenpunkten über Ersatz-Objektgruppen

Die Ausgangs-Datenpunkttypen der folgenden Objektgruppen sind über die oben genannten Ersatz-Objektgruppen projektierbar:

- 10 [2]
- 11 [1, 2]
- 21 [1, 2, 5, 6]
- 23 [1, 2, 5, 6]
- 40 [1, 2, 3]
- 42 [1, 2, 4, 5, 6, 7, 8]

Verwenden Sie für die Projektierung im DNP3-CP die jeweils angegebene Ersatz-Objektgruppe.

Ordnen Sie den jeweiligen Datenpunkt im Master über den in STEP 7 projektierbaren Datenpunktindex zu. Der Datenpunkt des DNP3-CP wird dann dem entsprechenden Datenpunkt im Master zugeordnet.

Beispiel für die Projektierung des Datenpunkts Binary Output (10 [2])
Der Datenpunkt wird wie folgt projektiert:
Im DNP3-CP als Binary Command (12 [1])
Im Master als Binary Output (10 [2])

Bei den Datenpunkttypen Binary Output Event (11) und Analog Output Event (42) müssen Sie zusätzlich die Rückspiegelung aktivieren, siehe nachfolgender Abschnitt.

Projektierung der Rückspiegelung bei Output Events (Objektgruppen 11 und 42)

Die Datenpunkttypen Binary Output Event (Objektgruppe 11) und Analog Output Event (Objektgruppe 42) legen Sie zunächst, wie oben beschrieben, als Datenpunkte der Objektgruppen 12 bzw. 41 an.

Die lokalen Werte dieser beiden Objektgruppen können auf Änderungen überwacht werden und die Änderungen an den Master übertragen werden. Die Änderung eines lokalen Werts kann bspw. durch eine Handbedienung vor Ort verursacht werden.

Damit der Wert, der durch lokale Ereignisse oder Eingriffe verursacht wird, an den Master übertragen werden kann, benötigt der jeweilige Datenpunkt einen Rückspiegelungskanal. Die Rückspiegelungsfunktion projektieren Sie in der Datenpunktprojektierung über die Option "Wertüberwachung", Register "Allgemein".

Beachten Sie, dass Sie für die Rückspiegelungsfunktion die lokalen Werte in der Steuerung auf die betreffende PLC-Variable des Datenpunkts rangieren müssen.

Zeitstempel der Daten beim DNP3-Protokoll im UTC-Format

Zeitstempel werden im UTC-Format (48 Bit) übertragen und beinhalten Millisekunden.

Datenpunkttypen des Protokolls "IEC 60870-5"

Tabelle 1-129 Unterstützte Datenpunkttypen, IEC-Typen und kompatible S7-Datentypen

Format (Speicher-bedarf)	Datenpunkttyp	IEC-Typ	Richtung	S7-Datentypen	Operandenbe-reich
Bit	Single-point information	<1>	in	Bool	I, Q, M, DB
	Single-point information with time tag CP56Time2a [1]	<30>	in	Bool	I, Q, M, DB
	Single command	<45>	out	Bool	Q, M, DB
	Single command with time tag CP56Time2a [1]	<58>	out	Bool	Q, M, DB
	Double command with time tag CP56Time2a [1]	<59>	out	Bool	DB [2]
Byte	Step position information	<5>	in	Byte, USInt	I, Q, M, DB
	Step position information with time tag CP56Time2a [1]	<32>	in	Byte, USInt	I, Q, M, DB
	Regulating step command with time tag CP56Time2a [1]	<60>	out	Byte, USInt	DB [2]
Integer (16 Bit)	Measured value, normalized value	<9>	in	Int	I, Q, M, DB
	Measured value, normalized value with time tag CP56Time2a [1]	<34>	in	Int	I, Q, M, DB
	Measured value, scaled value	<11>	in	Int	I, Q, M, DB
	Measured value, scaled value with time tag CP56Time2a [1]	<35>	in	Int	I, Q, M, DB
	Set point command, normalized value	<48>	out	Int	Q, M, DB
	Set point command, scaled value	<49>	out	Int	Q, M, DB
	Set point command, normalized value with time tag CP56Time2a [1]	<61>	out	Int	Q, M, DB
	Set point command, scaled value with time tag CP56Time2a [1]	<62>	out	Int	Q, M, DB
Integer (32 Bit)	Bitstring of 32 bits	<7>	in	UDInt, DWord	I, Q, M, DB
	Bitstring of 32 bits with time tag CP56Time2a [1]	<33>	in	UDInt, DWord	I, Q, M, DB
	Integrated totals	<15>	in	UDInt, DWord	I, Q, M, DB
	Integrated totals with time tag CP56Time2a [1]	<37>	in	UDInt, DWord	I, Q, M, DB
	Bitstring of 32 bits	<51>	out	UDInt, DWord	Q, M, DB
	Bitstring of 32 bits with time tag CP56Time2a - control direction [1]	<64>	out	UDInt, DWord	Q, M, DB
Gleitpunktzahl (32 Bit)	Measured value, short floating point number	<13>	in	Real	Q, M, DB
	Measured value, short floating point number with time tag CP56Time2a [1]	<36>	in	Real	Q, M, DB
	Set point command, short floating point number	<50>	out	Real	Q, M, DB
	Set point command, short floating point with time tag CP56Time2a [1]	<63>	out	Real	Q, M, DB

Format (Speicher-bedarf)	Datenpunkttyp	IEC-Typ	Richtung	S7-Datentypen	Operandenbe-reich
Datenblock (1...2 Bit) [2]	Double-point information	<3>	in	[2]	DB
	Double-point information with time tag CP56Time2a [1]	<31>	in	[2]	DB
	Double command	<46>	out	[2]	DB
	Regulating step command	<47>	out	[2]	DB
	Double command with time tag CP56Time2a [1]	<59>	out	[2]	DB
	Regulating step command with time tag CP56Time2a [1]	<60>	out	[2]	DB
Datenblock (1...32 Bit) [3]	Bitstring of 32 bits [3]	<7>	in	[3]	DB
	Bitstring of 32 bits with time tag CP56Time2a [1] [3]	<33>	in	[3]	DB
	Bitstring of 32 bits [3]	<51>	out	[3]	DB
	Bitstring of 32 bits with time tag CP56Time2a - control direction [1] [3]	<64>	out	[3]	DB

[1] Zum Format der Zeitstempel siehe nachfolgender Abschnitt.

[2] Legen Sie für diese Datenpunkttypen einen Datenbaustein mit einem Array von genau 2 Bool an.

[3] Mit diesen Datenpunkttypen können zusammenhängende Speicherbereiche bis zu einer Größe von 32 Bit übertragen werden. Kompatibel ist nur der S7-Datentyp Bool.

Zeitstempel der Daten beim IEC-Protokoll

Zeitstempel werden gemäß der IEC-Spezifikation im Format "CP56Time2a" übertragen. Beachten Sie, dass nur die ersten 3 Bytes für Millisekunden und Minuten übertragen werden.

Prozessabbild, Übertragungsart, Ereignisklassen

Einleitung

Einige der nachfolgend beschriebenen Funktionen unterscheiden sich in Abhängigkeit des vom jeweiligen Kommunikationsmodul oder vom verwendeten Protokoll.

Speicherung von Werten

Generell werden die Werte von allen Datenpunkten im Abbildspeicher der Baugruppe gespeichert. Werte im Abbildspeicher werden erst nach Aufruf durch die Zentrale bzw. den DNP3- / IEC-Master übertragen.

Ereignisse werden zusätzlich im Sendepuffer gespeichert und können spontan übertragen werden.

Der Abbildspeicher, das Prozessabbild des Moduls

Der Abbildspeicher ist das Prozessabbild des Kommunikationsmoduls. Im Abbildspeicher werden alle aktuellen Werte der projektierten Datenpunkte gespeichert. Neue Werte eines Datenpunkts überschreiben den zuletzt gespeicherten Wert im Abbildspeicher.

Die Werte werden nur unter folgenden Bedingungen übertragen:

- Nach Abfrage des Kommunikationspartners, siehe "Übertragung nach Aufruf" im Abschnitt "Übertragungsarten" unten.
- Zusammen mit einem Daten-Telegramm aus dem Sendepuffer, das sofort übertragen werden muss.

Der Sendepuffer

Der Sendepuffer des Kommunikationsmoduls ist der Speicher für die einzelnen Werte von Datenpunkten, die als Ereignis projektiert sind (Telegrammspeicher). Die Größe des Sendepuffers entnehmen Sie dem Handbuch der jeweiligen Baugruppe.

Die Kapazität des Sendepuffers wird für alle aktivierten Partner gleichmäßig aufgeteilt.

Wenn die Verbindung mit einem Kommunikationspartner unterbrochen ist, bleiben die einzelnen Werte der Ereignisse durch die Pufferung erhalten. Bei wiederkehrender Verbindung werden die gepufferten Werte gesendet. Der Telegrammspeicher arbeitet chronologisch, das heißt, die ältesten Daten werden zuerst gesendet (FIFO-Prinzip).

Wenn ein Daten-Telegramm an den Kommunikationspartner übertragen wurde, dann werden die übertragenen Daten aus dem Sendepuffer gelöscht.

Wenn Daten für längere Zeit nicht übertragen werden können und der Sendepuffer droht überzulaufen, gilt das folgende Protokoll-abhängige Verhalten:

- Telecontrol Basic, ST7
  Das Zwangsabbildspeicherverfahren
  Bei Erreichen eines gewissen Füllgrades des Sendepuffers schaltet das Modul in das Zwangsabbildspeicherverfahren um. Der Füllgrad zur Umschaltung in das Zwangsabbildspeicherverfahren beträgt:
  - Telecontrol Basic: 80 % Speicherbelegung des Sendepuffers
  - ST7: 90 % Speicherbelegung des Sendepuffers

  Neue Werte von Datenpunkten, die als Ereignis projektiert sind, werden im Zwangsabbildspeicherverfahren nicht mehr zusätzlich in den Sendepuffer eingetragen, sondern sie überschreiben ältere, bereits vorhandene Werte im Abbildspeicher.
  Bei wiederkehrender Verbindung mit dem Kommunikationspartner schaltet das Modul wieder in das Sendepufferverfahren, sobald 50 % Füllgrad des Sendepuffers unterschritten werden.
- DNP3 / IEC
  Bei Erreichen von 100 % Füllgrad des Sendepuffers werden keine weiteren Werte mehr gespeichert, bis der Füllgrad wieder unter 100 % sinkt.

Speicherung der Datenpunktwerte

Generell werden die Werte von Datenpunkten im Abbildspeicher der Baugruppe gespeichert und erst nach Abfrage durch den Kommunikationspartner übertragen.

Ereignisse werden zusätzlich im Sendepuffer gespeichert und können spontan übertragen werden.

Über den Parameter "Übertragungsart" (siehe unten) werden Datenpunkte als statischer Wert oder als Ereignis projektiert:

- **Statischer Wert (kein Ereignis)**
  Statische Werte werden in den Abbildspeicher (Prozessabbild des CP) eingetragen.
  Statische Werte entsprechen folgenden Klassen:
  - DNP3: Class 0
  - IEC: Klasse 2
- **Ereignis**
  Die Werte von Datenpunkten, welche als Ereignis projektiert sind, werden ebenfalls in den Abbildspeicher des Kommunikationsmoduls eingetragen. Der Wert des Ereignisses wird spontan an den Kommunikationspartner gesendet, wenn dies durch den Master freigegeben ist.
  Zusätzlich werden die Werte von Ereignissen in den Sendepuffer des Kommunikationsmoduls eingetragen.
  Ereignisse entsprechen folgenden Klassen:
  - DNP3: Class 1 / 2 / 3
  - IEC: Klasse 1

Übertragungsarten

Folgende Übertragungsarten sind möglich:

- **Übertragung nach Aufruf (class 0)**
  Der jeweils aktuelle Wert des Datenpunkts wird in den Abbildspeicher des CP eingetragen. Neue Werte eines Datenpunkts überschreiben den zuletzt gespeicherten Wert im Abbildspeicher.
  Nach Aufruf durch den Kommunikationspartner wird der zu diesem Zeitpunkt aktuelle Wert übertragen.
- **Getriggert (Ereignisse)**
  Die Werte von Datenpunkten, welche als Ereignis projektiert sind, werden in den Abbildspeicher und zusätzlich in den Sendepuffer des CP eingetragen.
  Die Werte von Ereignissen werden in folgenden Fällen gespeichert:
  - Die jeweils projektierten Triggerbedingungen sind erfüllt (Datenpunktprojektierung > Register "Trigger", siehe unten)
  - Der Wert eines Statusbits der Statuskennungen des Datenpunkts ändert sich, siehe Kapitel Statuskennungen der Datenpunkte (Seite 1869).
    Beispiel:
    Wenn der Wert eines als Ereignis projektierten Datenpunkts beim Anlauf der Station durch erstmaliges Lesen der CPU-Daten aktualisiert wird, dann ändert sich der Status "RESTART" dieses Datenpunkts (Bit-Zustandswechsel 1 → 0). Dies führt zur Erzeugung eines Ereignisses.

  Die projektierbaren Ereignisklassen der verschiedenen Protokolle werden in den nachfolgenden Abschnitten beschrieben.

Ereignisklassen bei der Übertragungsart "Getriggert"

Abhängig vom verwendeten Protokoll stehen folgende Ereignisklassen zur Verfügung:

- TeleControl Basic / ST7
  - Jeder Wert getriggert
    Jede Wertänderung wird in chronologischer Reihenfolge in den Sendepuffer eingetragen.
  - Aktueller Wert getriggert
    Nur der jeweils letzte, aktuelle Wert wird in den Sendepuffer eingetragen. Er überschreibt den dort zuvor gespeicherten Wert.
- DNP3
  Die Auswertung der nachfolgenden Klassifizierung muss vom Master vorgenommen werden.
  - Ereignisklasse 1
    Klasse gemäß DNP3-Protokoll: Class 1
    Jede Wertänderung wird in chronologischer Reihenfolge in den Sendepuffer eingetragen.
  - Ereignisklasse 2
    Klasse gemäß DNP3-Protokoll: Class 2
    Jede Wertänderung wird in chronologischer Reihenfolge in den Sendepuffer eingetragen.
  - Ereignisklasse 3
    Klasse gemäß DNP3-Protokoll: Class 3
    Nur der jeweils aktuelle Wert zu dem Zeitpunkt, an dem die Trigger-Bedingung erfüllt ist, wird in den Sendepuffer eingetragen und überschreibt den dort zuletzt gespeicherten Wert.
- IEC
  Beide nachfolgende Ereignisklassen entsprechen der Anwender-Datenklasse 1 des IEC-Protokolls
  - Jeder Wert getriggert
    Jede Wertänderung wird in chronologischer Reihenfolge in den Sendepuffer eingetragen.
  - Aktueller Wert getriggert
    Nur der jeweils aktuelle Wert zu dem Zeitpunkt, an dem die Trigger-Bedingung erfüllt ist, wird in den Sendepuffer eingetragen und überschreibt den dort zuletzt gespeicherten Wert.

Für das Auslösen der ereignisgesteuerten Übertragung stehen verschiedene Trigger-Typen zur Verfügung, siehe Kapitel Register "Trigger" (Seite 1875).

ST7-Datenpunkte

Objekte der TD7-Software

Die Telecontrol-Kommunikation von Baugruppen, welche das Fernwirkprotokoll SINAUT ST7 verwenden, wird über die TD7-Software abgewickelt.

Die zu übertragenen Daten der SINAUT-Objekte werden alternativ projektiert bzw. parametriert:

- Über Datenpunkte (TD7onTIM)
- Über Programmbausteine der Bibliothek "Telecontrol ST7" (TD7onCPU)

Zu den beiden Varianten der TD7-Software siehe Kapitel Die TD7-Software (Seite 1895).

Daten-Objekte

Das Senden und Empfangen der Prozessdaten wird mithilfe von standardisierten Daten-Objekten projektiert. Entsprechend den zwei Übertragungsrichtungen unterscheiden sie sich in:

- **Sende-Objekte (.._S)**
  Daten-Objekte für das Erfassen und Senden von Daten
  Ihre Namen haben die Endung "_S" (send).
- **Empfangs-Objekte (.._R)**
  Daten-Objekte für das Empfangen und Ausgeben von Daten
  Ihre Namen haben die Endung "_R" (receive).

Die Namen der Daten-Objekte sind bei TD7onTIM und TD7onCPU identisch. Bezüglich der Funktionen sind sie miteinander kompatibel.

Objektnummern

Jedes Daten-Objekt hat eine Objektnummer. Beim Anlagen neuer Datenpunkte werden die Nummern in der Voreinstellung aufsteigend vergeben. Die Nummern können verändert werden.

Regeln:

- Die Objektnummern eines Moduls müssen eindeutig sein.
- Die kleinste Objektnummer ist 1.

Objektkanäle

Jedes Daten-Objekt hat einen oder mehrere Objektkanäle, siehe Kapitel Datenpunkttypen (Seite 1857).

Die Sende- und Empfangs-Kanäle der Daten-Objekte sind jeweils für die Verarbeitung eines einzelnen Prozesswerts zuständig, z. B. für das Aufbereiten und Senden eines Analogwertes oder das Empfangen und Ausgeben eines Befehls-Bytes.

Jedes Daten-Objekt enthält ein oder mehrere Sende- oder Empfangs-Kanäle. Anzahl und Art der Sende- und Empfangs-Kanäle pro Daten-Objekt können nicht verändert werden. Das Daten-Objekt "Ana04W_S" beispielsweise hat 4 Sende-Kanäle des Typs "Analogwert senden".

Bei Objekten mit mehreren Kanälen des gleichen Typs müssen nicht alle Kanäle aktiviert werden, wenn sie nicht benötigt werden.

Regeln:

- Die Kanalnummern eines Objekts müssen mit 1 beginnen und fortlaufend ohne Lücke vergeben werden.
- Alle Kanäle eines Objekts müssen den gleichen Datenpunkttyp haben.
- Alle Kanäle eines Objekts müssen den gleichen Partner haben.

DNP3- und IEC-Datenpunkte

Allgemein

Die wichtigsten Parameter im ersten Register des Datenpunkteditors finden Sie auch in der Datenpunkt-Tabelle.

- **Datenquelle**
  - Name: Eindeutiger Name des Datenpunkts
  - Zur Zuordnung der PLC-Variable siehe Kapitel Projektierung der Datenpunkte und Nachrichten (Seite 1848).
  - Zum Datenpunkttyp siehe Kapitel Datenpunkttypen (Seite 1857).
  - Zum Datenpunktindex siehe Kapitel Regeln zur Projektierung des Datenpunktindex (Seite 1870).
- **Master-Funktion**
  - Option aktiviert: Die Werte des Datenpunkts werden wie bei einem Master versendet:
    - Eingangs-Datenpunkte werden empfangen.
    - Ausgangs-Datenpunkte werden gesendet.
  - Option deaktiviert: Der Datenpunkt kann für 'Direkte Kommunikation' zwischen Stationen verwendet werden:
    - Eingangs-Datenpunkte können empfangen und gesendet werden.
    - Ausgangs-Datenpunkte können gesendet und empfangen werden.
- **Wertüberwachung**
  Nur bei DNP3-Befehlen und Ausgängen
  Bei aktivierter Option wird der aktuelle Wert des Datenpunkts, der vom Master gesetzt werden kann, von der Station an den Master zurückgespiegelt. Weiterhin werden die Trigger-Optionen des Datenpunkts aktivierbar, um das Auslösen des Versendens zu projektieren.
- **Übertragungsart**
  Zur Übertragungsart siehe Kapitel Prozessabbild, Übertragungsart, Ereignisklassen (Seite 1863).
- **Lesezyklus**
  Nur bei Eingängen
  Zum Lesezyklus siehe Kapitel Lesezyklus (Seite 1871).

- **Antwort auf Generalabfrage**
  Aktiviert den Datenpunkt für die Antwort auf eine Generalabfrage. Bei deaktivierter Funktion wird der Wert des Datenpunkts nach einer Generalabfrage nicht an den Kommunikationspartner gesendet.
- **Zuordnung zu Gruppenabfrage**
  Nur bei IEC-Datenpunkten
  Ordnet den Datenpunkt einer Gruppenabfrage zu.
  Bei Gruppenabfragen des Masters auf die betreffende Gruppe wird der Wert des Datenpunkts an den Master gesendet.

Statuskennungen der Datenpunkte

Statuskennungen

Die in den nachfolgenden Tabellen aufgelisteten Statuskennungen der Datenpunkte werden zusammen mit dem Wert in jedem Daten-Telegramm an den Kommunikationspartner übertragen. Sie können vom Kommunikationspartner ausgewertet werden.

Der CP 1243-8 IRC liefert keine Statuskennungen.

Die Statuskennungen unterscheiden sich nach dem verwendeten Protokoll.

Die Einträge der Tabellenzeile "Bedeutung" beziehen sich auf den jeweiligen Eintrag in der Tabellenzeile "Bit-Zustand".

TeleControl Basic

Die Status-Bits (siehe Tabelle) werden von TCSB abhängig von deren Zustand in den OPC quality code umgesetzt.

- Quality = BAD
  Bit 2 oder 7 = 1
- Quality = UNCERTAIN
  Bit 1 oder 3 oder 5 = 1
- Quality = GOOD
  Bits 1 und 2 und 3 und 5 und 6 = 0

Tabelle 1-130 Bit-Belegung des Status-Byte 0

Bit	7	6	5	4	3	2	1	0
Flag-Name	-	NON_ EXISTENT	Substituted	LOCAL_ FORCED	CARRY	OVER_ RANGE	RESTART	ONLINE
Bedeutung	-	Datenpunkt nicht vorhanden oder S7-Adresse nicht erreichbar	Ersatzwert	Vor-Ort-Bedienung	Zählwertüberlauf vor Lesen des Werts	Grenzwert der Analogwert-Vorverarbeitung über- / unterschritten	Wert nach Start noch nicht aktualisiert	Wert ist gültig
Bit-Zustand	*(immer 0)*	1	1	1	1	1	1	1

DNP3

Die Statuskennungen entsprechen folgenden Elementen der Spezifikation:

OBJECT FLAGS - DNP3 Specification, Volume 6, Data Object Library - Part 1

Tabelle 1-131 Bit-Belegung des Status-Byte

Bit	7	6	5	4	3	2	1	0
Flag-Name	-	-	-	LOCAL_ FORCED	DISCONTI NUITY	OVER_ RANGE	RESTART	ONLINE
Bedeutung	-	-	-	Vor-Ort-Be-dienung	Zählwert-überlauf vor Lesen des Werts	Grenzwert der Analog-wert-Vorver-arbeitung über- / unter-schritten	Wert nach Start noch nicht aktuali-siert	Wert ist gültig
Bit-Zustand	*(immer 0)*	*(immer 0)*	*(immer 0)*	1	1	1	1	1

IEC 60870-5

Die Statuskennungen entsprechen folgenden Elementen der Spezifikation:

Quality descriptor - IEC 60870 Part 5-101

Tabelle 1-132 Bit-Belegung des Status-Byte

Bit	7	6	5	4	3	2	1	0
Flag-Name	-	-	SB substituted	-	CY carry	OV overflow	NT not topical	IV invalid
Bedeutung	-	-	Ersatzwert	-	Zählwert-überlauf vor Lesen des Werts	Wertebereich überschrit-ten, Analog-wert	Wert nicht ak-tualisiert	Wert ist gültig
Bit-Zustand	*(immer 0)*	*(immer 0)*	1	*(immer 0)*	1	1	1	0

Regeln zur Projektierung des Datenpunktindex

Projektierung des Datenpunktindex

Nachfolgend finden Sie für die einzelnen Protokoll-Typen die Regeln für die Projektierung des Datenpunktindex.

TeleControl Basic

Innerhalb eines CP müssen die Indices der Datenpunktklassen folgenden Regeln entsprechen:

- Eingang
  Der Index eines Datenpunkts vom Typ Eingang muss über alle Datenpunkttypen (Digitaleingänge, Analogeingänge etc.) eindeutig sein.
- Ausgang
  - Ein Datenpunkts vom Typ Ausgang kann den gleichen Index haben wie ein Datenpunkt vom Typ Eingang.
  - Mehrere Datenpunkte vom Typ Ausgang können den gleichen Index haben.

Hinweis

Datenpunkte für die Querkommunikation mit einem CP in einer anderen S7-Station

Beachten Sie, dass bei der Querkommunikation die Indices der beiden korrespondierenden Datenpunkte (Datenpunktpärchen) beim sendenden und beim empfangenden CP identisch sein müssen.

DNP3

In einem CP müssen Datenpunktindices jeweils innerhalb einer der folgenden Objektgruppen eindeutig sein:

- Binary Input / Binary Input Event
- Binary Output / Binary Command
- Counter / Counter Event
- Analog Input / Analog Input Event
- Analog Output
- Octet String / Octet String Event

Die Indices zweier Datenpunkte in verschiedenen Objektgruppen können identisch sein.

IEC 60870-5-104

In einem CP müssen die Datenpunktindices eindeutig sein.

Doppelt vergebene Datenpunktindices werden bei der Konsistenzprüfung als Fehler gemeldet und verhindern das Übersetzen des Projekts.

Lesezyklus

Eingangs-Datenpunkte werden dem Lesezyklus der CPU in der Datenpunktprojektierung im Register "Allgemein > Lesezyklus" zugeordnet.

Aufbau des CPU-Abtastzyklus

Der Zyklus, mit dem die Übertragungs-Baugruppe (TIM) den Speicherbereich der CPU abtastet, ist aus den folgenden Phasen aufgebaut:

- **Hochpriore Leseaufträge (Schneller Zyklus)**
  Für alle Datenpunkte mit der Zuordnung "Schneller Zyklus" werden die PLC-Variablen in jedem Abtastzyklus gelesen.
  In der Regel ist es ausreichend, nur schnell zu erfassende Daten wie Alarme und Wischermeldungen sowie Befehls-, Sollwert- und Parameter-Objekte für die 1-aus-n-Prüfung dem schnellen Zyklus zuzuordnen.
  Zur 1-aus-n-Prüfung siehe Glossar.
- **Schreibaufträge**
  In jedem Zyklus werden die Werte einer bestimmten Anzahl spontaner Schreibaufträge in die CPU geschrieben.
  Die Anzahl der Variablen, die pro Zyklus geschrieben werden, wird für die Übertragungs-Baugruppe in der Parametergruppe "Kommunikation mit der CPU" mit dem Parameter "Max. Anzahl der Schreibaufträge" festgelegt. Diejenigen Variablen, deren Anzahl diesen Wert übersteigt, werden dann im nächsten oder einem der folgenden Zyklen geschrieben.
- **Niederpriore Leseaufträge - anteilig (Normaler Zyklus)**
  Für Datenpunkte mit der Zuordnung "Normaler Zyklus" werden die Werte ihrer PLC-Variablen anteilig in jedem Abtastzyklus gelesen.
  Die Anzahl der Variablen, die pro Zyklus gelesen werden, wird für die Übertragungs-Baugruppe in der Parametergruppe "Kommunikation mit der CPU" mit dem Parameter "Max. Anzahl der Leseaufträge" festgelegt. Die Variablen, die diesen Wert übersteigen und somit in einem Zyklus nicht gelesen werden, werden dann im nächsten oder einem der folgenden Zyklen gelesen.
- **Zykluspausenzeit**
  Diese Wartezeit zwischen zwei Abtastzyklen dient dazu, anderen Prozessen, die auf die CPU zugreifen, genügend Zeit zu reservieren.

Register "Lokale Sollwerteingabe"

Sollwerte und deren Rückspiegelung

Über die nachfolgend beschriebenen Optionen können Sie lokal eingegebene Werte von Sollwert-Ausgängen erfassen und die aktuellen Werte bei Änderung an den Partner zurückspiegeln. Die Änderung eines lokalen Werts kann bspw. durch Handbedienung vor Ort verursacht werden.

Die Funktionen stehen für folgende Module zur Verfügung:

- TIM 3V-IE / TIM 3V-IE Advanced
- TIM 4R-IE

Die Funktionen können Sie bei folgenden Datenpunkttypen projektieren:

- Sollwert-Eingang (Set01W_S)
- Sollwert-Ausgang (Set01W_R)

- Parameter-Eingang (Par12D_S)
- Parameter-Ausgang (Par12D_R)

Für die Funktionen müssen bei beiden Kommunikationspartner die entsprechenden Eingangs- und Ausgang-Datenpunkte angelegt sein. Rangieren Sie für die Rückspiegelungsfunktion den lokalen Wert in der Steuerung auf die betreffende PLC-Variable des Datenpunkts.

Die Parameter für den lokalen Sollwert-Eingabemodus und die Rückspiegelung sind optional. Falls Sie diese nicht benötigen, lassen Sie sie deaktiviert.

Parameter

Die Liste zeigt die Funktionen der Datenpunkttypen sowie die Register und Parametergruppen, in denen die Funktionen projektiert werden.

- **Sollwert-/Parameter-Eingang**
  Register "Allgemein" > "Datenquelle":
  - Sollwert an Partner senden
    Der Sollwert ist abgelegt in der PLC-Variable.

  Register "Lokale Sollwerteingabe" > "Empfangen":
  - Empfang des vom Partnerobjekt mitgesendeten Sollwert-Eingabemodus 'Lokale Eingabe'
    Der Eingabemodus wird an die PLC-Variable "Sollwert-Eingabemodus" (Bool) übergeben.
  - Empfang des Rückspiegelwerts vom Partner
    Der vom Partner zurückgespiegelte Wert wird an die PLC-Variable "Rückspiegelwert" übergeben.
- **Sollwert-/Parameter-Ausgang**
  Register "Allgemein" > "Datenquelle":
  - Sollwert vom Partner empfangen
    Der Sollwert wird an die PLC-Variable übergeben.

  Register "Lokale Sollwerteingabe" > "Senden":
  - Senden des lokalen Sollwert-Eingabemodus 'Lokale Eingabe' an das Partnerobjekt
    Die Information, ob der lokal vergebene Sollwert oder der vom Partner empfangene Sollwert gültig ist, wird aus der PLC-Variable "Sollwert-Eingabemodus" gelesen:
    - 0: Der empfangene Sollwert ist gültig.
    - 1: Der lokal eingegebene Sollwert ist gültig.
  - Senden des lokal eingegebenen Sollwerts an den Partner
    Der Sollwert wird aus der PLC-Variable "Lokaler Sollwert" gelesen.

Lokale Sollwerteingabe - Empfangen

Die Parametergruppe ist bei sendenden Datentypen (*_S) aktiv.

Folgende Parameter stehen für lokale Sollwerte des Partners zur Verfügung:

- Sollwert-Eingabemodus
  Über den Parameter wird die Rückmeldung vom Partner-Datenpunkt über dessen Sollwert-Eingabemodus empfangen.
  Die Information wird in der PLC-Variable (Bool) gespeichert.
  - 0 = Datenpunkt des Partners im Modus 'Sollwertempfang'
  - 1 = Datenpunkt des Partners im Sollwert-Eingabemodus 'lokal'

  Der Parameter dient nur der Signalisierung. Durch diesen Parameter und durch den entsprechenden Parameter des Partner-Datenpunkts wird beim Partner keine Verriegelung der mit der lokalen Sollwertvorgabe bewirkt. Die Verriegelung muss durch das Anwenderprogramm realisiert werden.
  Nach Anlauf der CPU oder der Partner-CPU oder nach Wiederkehr der Verbindung wird über eine Generalabfrage sichergestellt, dass am Parameter der aktuell gültige Zustand des Partners angezeigt wird.
- Rückspiegelwert
  Der Partner-Datenpunkt, der den Sollwert empfängt, spiegelt den dort aktuell gültigen Sollwert zurück, wenn dort der Parameter "Sollwert-Status" aktiviert ist. Dieser rückgespiegelte Sollwert wird hier zur Anzeige gebracht.
  Wenn das Partner-Objekt auf 'lokal' steht und dort eine neue Eingabe gemacht wird, dann wird der dort geänderte Sollwert hier angezeigt, wenn am Partner-Objekt der Parameter "Lokaler Sollwert" aktiviert ist.
  Nach Anlauf der CPU oder der Partner-CPU oder nach Wiederkehr der Verbindung wird über eine Generalabfrage sichergestellt, dass am Parameter der gültige Sollwert des Partners angezeigt wird.

Lokale Sollwerteingabe - Senden

Die Parametergruppe ist bei empfangenden Datentypen (*_R) aktiv.

Folgende Parameter stehen für lokale Sollwerte der eigenen Station zur Verfügung:

- Sollwert-Eingabemodus
  Über diesen Eingang kann die Information signalisiert werden, ob der lokal vorgegebene Sollwert oder der vom Partner gesendete Sollwert gültig ist.
  Die Information wird in der PLC-Variable (Bool) gespeichert.
  - 0 = Der Sollwert des Partners ist gültig.
  - 1 = Der lokal eingegebene Sollwert ist gültig.
    Ein vom Partner (z. B. der Zentrale) gesendeter Sollwert kann vom Objekt auch dann angenommen werden, wenn der Parameter gesetzt ist (True).

  Der Parameter dient nur der Information. Eine Verriegelung mit der remoten Sollwertvorgabe muss im Anwenderprogramm realisiert werden.
- Lokaler Sollwert
  Bei aktiviertem Parameter wird der lokal eingegebene Sollwert aus der PLC-Variable gelesen und an den Sende-Datenpunkt des Partners zurückgespiegelt.

Register "Trigger"

Trigger

Datenpunkte werden über den Parameter "Übertragungsart" als statischer Wert oder als Ereignis projektiert.

Das Speichern eines Ereignis-Werts im Sendepuffer (Telegrammspeicher) wird über verschiedene Trigger-Typen ausgelöst:

- **Schwellenwert-Trigger**
  Der Wert des Datenpunkts wird gespeichert, wenn dieser eine bestimmte Schwelle erreicht. Die Schwelle wird als Differenz zu dem zuletzt gespeicherten Wert berechnet, siehe Kapitel Schwellenwert-Trigger (Seite 1876).
- **Zeit-Trigger**
  Der Wert des Datenpunkts wird in einem projektierbaren Zeitraster oder zu einer bestimmten Uhrzeit gespeichert.
- **Ereignis-Trigger (Trigger-Variable)**
  Der Wert des Datenpunkts wird gespeichert, wenn ein projektierbares Trigger-Signal ausgelöst wird. Als Trigger-Signal wird der Flankenwechsel (0 → 1) einer Trigger-Variable ausgewertet, die vom Anwenderprogramm gesetzt wird. Für jeden Datenpunkt kann bei Bedarf eine separate Trigger-Variable projektiert werden.
  Rücksetzen der Trigger-Variable im Merkerbereich / DB:
  Wenn der Speicherbereich der Trigger-Variable im Merkerbereich oder in einem Datenbaustein liegt, dann wird die Trigger-Variable mit der Übertragung des Werts auf Null zurückgesetzt. Dies kann bis zu 500 Millisekunden dauern.

Hinweis

Schnelles Setzen von Triggern

Trigger dürfen nicht schneller als in einem Mindestabstand von 500 Millisekunden gesetzt werden. Dies gilt auch für Hardware-Trigger (Eingabebereich).

Hinweis

Hardware-Trigger

Hardware-Trigger müssen Sie über das Anwenderprogramm zurücksetzen.

Übertragungszeitpunkt

Ob der Wert eines Ereignisses nach Auslösen des Triggers sofort oder zeitversetzt an den Kommunikationspartner übertragen wird, hängt vom verwendeten Protokoll und den Einstellungen ab:

- DNP3 / IEC
  Bei diesen Protokollen hängt die spontane Übertragung davon ab, ob spontanes Senden bzw. asymmetrische Kommunikation im Netz möglich ist.
  Bei DNP3-Modulen ist dies über den "Übertragungsmodus" in der Parametergruppe "Übertragungseinstellungen - DNP3" der Schnittstelle einstellbar.
- TeleControl Basic / ST7
  Hier legen Sie legen Sie den Übertragungszeitpunkt mit dem Parameter "Übertragungsmodus" am Datenpunkt fest.

Übertragungsmodus

Der Übertragungsmodus eines Daten-Telegramms wird im Register "Trigger" des Datenpunkts eingestellt. Mit der Option legen Sie fest, ob Daten-Telegramme von Ereignissen sofort oder zeitversetzt gesendet werden:

- Spontan
  Der Wert wird sofort übertragen.
- Bedingt spontan
  Der Wert wird erst übertragen, wenn eine der folgenden Bedingungen erfüllt ist:
  - Der Kommunikationspartner fragt die Station ab.
  - Der Wert eines anderen Ereignisses mit dem Übertragungsmodus "Spontan" wird übertragen.
  - Der Füllgrad des Sendepuffers hat 80 % seiner maximalen Kapazität erreicht.

Schwellenwert-Trigger

Hinweis

Schwellenwert-Trigger: Berechnung erst nach Analogwert-Vorverarbeitung

Beachten Sie, dass die Analogwert-Vorverarbeitung vor der Prüfung auf einen projektierten Schwellenwert und vor der Berechnung des Schwellenwerts durchgeführt wird.

Dies hat Auswirkungen auf den Wert, der beim Schwellenwert-Trigger projektiert wird.

Hinweis

Kein Schwellenwert-Trigger bei projektierter Mittelwertbildung

Bei projektierter Mittelwertbildung kann für das betreffende Analogwertereignis kein Schwellenwert-Trigger projektiert werden.

Zum zeitlichen Ablauf der Analogwert-Vorverarbeitung siehe Kapitel Analogwert-Vorverarbeitung (Seite 1878).

Schwellenwert-Trigger

Funktion

Bei Abweichung des Prozesswerts um den Betrag des Schwellenwerts wird der Prozesswert gespeichert.

Für die Berechnung der Schwellenwert-Abweichung werden zwei Verfahren angewendet:

- Absolutes Verfahren
  Bei Binär- und Zählwerten sowie bei Analogwerten mit projektierter Mittelwertbildung wird für die Berechnung der Schwellenwert-Abweichung das absolute Verfahren angewendet.
- Integratives Verfahren
  Bei Analogwerten ohne projektierte Mittelwertbildung wird für die Berechnung der Schwellenwert-Abweichung das integrierende Verfahren angewendet.
  Bei der integrierenden Schwellenwertberechnung wird nicht der absolute Betrag der Abweichung des Prozesswerts vom zuletzt gespeicherten Wert ausgewertet sondern die integrierte Abweichung.

Absolutes Verfahren

Für jeden Binärwert geprüft, ob der aktuelle (eventuell geglättete) Wert außerhalb des Schwellenwertbandes liegt. Das jeweils aktuelle Schwellenwertband ergibt sich aus dem zuletzt gespeicherten Wert und dem Betrag des projektierten Schwellenwerts:

- Obergrenze des Schwellenwertbandes: Letzter gespeicherter Wert + Schwellenwert
- Untergrenze des Schwellenwertbandes: Letzter gespeicherter Wert - Schwellenwert

Sobald der Prozesswert die Ober- oder Untergrenze des Schwellenwertbandes erreicht, wird der Wert gespeichert. Der neue gespeicherte Wert dient als Basis für die Berechnung des neuen Schwellenwertbandes.

Integratives Verfahren

Die integrierende Schwellenwertberechnung arbeitet mit einem zyklischen Vergleich des integrierten aktuellen Werts mit dem zuletzt gespeicherten Wert. Der Berechnungszyklus, in dem die beiden Werte verglichen werden, beträgt 500 Millisekunden.
(Anmerkung: Der Berechnungszyklus ist nicht zu verwechseln mit dem Abtastzyklus der CPU-Speicherbereiche).

Die Abweichungen des aktuellen Prozesswerts werden in jedem Berechnungszyklus aufaddiert. Erst wenn der aufsummierte Wert den projektierten Wert des Schwellenwert-Triggers erreicht, wird der Trigger gesetzt und ein neuer Prozesswert in den Sendepuffer eingetragen.

Das Verfahren wird am folgenden Beispiel erläutert, bei dem ein Schwellenwert von 2,0 projektiert ist.

Tabelle 1-133 Beispiel für die integrierende Berechnung eines mit 2,0 projektierten Schwellenwerts

Zeit [s] (Berechnungs-Zyklus)	Im Sendepuffer gespeicherter Prozesswert	Aktueller Prozesswert	Absolute Abweichung vom gespeicherten Wert	Integrierte Abweichung
0	**20,0**	**20,0**	0	0
0,5		20,3	+0,3	0,3

Zeit [s] (Berechnungs-Zyklus)	Im Sendepuffer gespeicherter Prozesswert	Aktueller Prozesswert	Absolute Abweichung vom gespeicherten Wert	Integrierte Abweichung
1,0		19,8	-0,2	0,1
1,5		20,2	+0,2	0,3
2,0		20,5	+0,5	0,8
2,5		20,3	+0,3	1,1
3,0		20,4	+0,4	1,5
3,5	**20,5**	**20,5**	+0,5	**2,0**
4,0		20,4	-0,1	-0,1
4,5		20,1	-0,4	-0,5
5,0		19,9	-0,6	-1,1
5,5		20,1	-0,4	-1,5
6,0	**19,9**	**19,9**	-0,6	**-2,1**

Bei dem im Beispiel dargestellten Verlauf des Prozesswerts wird der mit 2,0 projektierte Schwellenwert-Trigger zweimal ausgelöst:

- Zum Zeitpunkt 3,5 s: Der Betrag der integrierten Abweichung liegt bei 2,0. Der neue im Sendepuffer gespeicherte Prozesswert beträgt 20,5.
- Zum Zeitpunkt 6,0 s: Der Betrag der integrierten Abweichung liegt bei 2,1. Der neue im Sendepuffer gespeicherte Prozesswert beträgt 19,9.

Wenn in diesem Beispiel eine Abweichung des Prozesswerts von ca. 0,5 das Auslösen des Triggers veranlassen sollte, dann müsste bei dem dargestellten Verhalten des Prozesswerts ein Schwellenwert von ca. 1,5 ... 2,5 projektiert werden.

Analogwert-Vorverarbeitung

CPs mit Datenpunkt-Projektierung unterstützen die Analogwert-Vorverarbeitung. Für Analogwert-Datenpunkte können einige oder alle der nachfolgend beschriebenen Funktionen projektiert werden.

Voraussetzungen und Restriktionen

Voraussetzungen für die Projektierung der Vorverarbeitungsoptionen und gegenseitige Restriktionen finden Sie im Abschnitt der jeweiligen Funktion.

Hinweis

Restriktionen aufgrund von projektierten Triggern

Die Analogwert-Vorverarbeitungsoptionen "Fehlerunterdrückungszeit", "Grenzwertberechnung" und "Glättung" werden nicht durchgeführt, wenn kein Schwellenwert-Trigger für den betreffenden Datenpunkt projektiert ist. In diesen Fällen wird der gelesene Prozesswert des Datenpunkts vor Ablauf des Vorverarbeitungszyklus der Schwellenwert-Berechnung (500 ms) in den Abbildspeicher des CP eingetragen und transparent übertragen.

Ablauf der Analogwert-Vorverarbeitungsoptionen

Die Werte von Analogeingängen, die als Ereignis projektiert sind, werden im CP nach folgendem Schema bearbeitet:

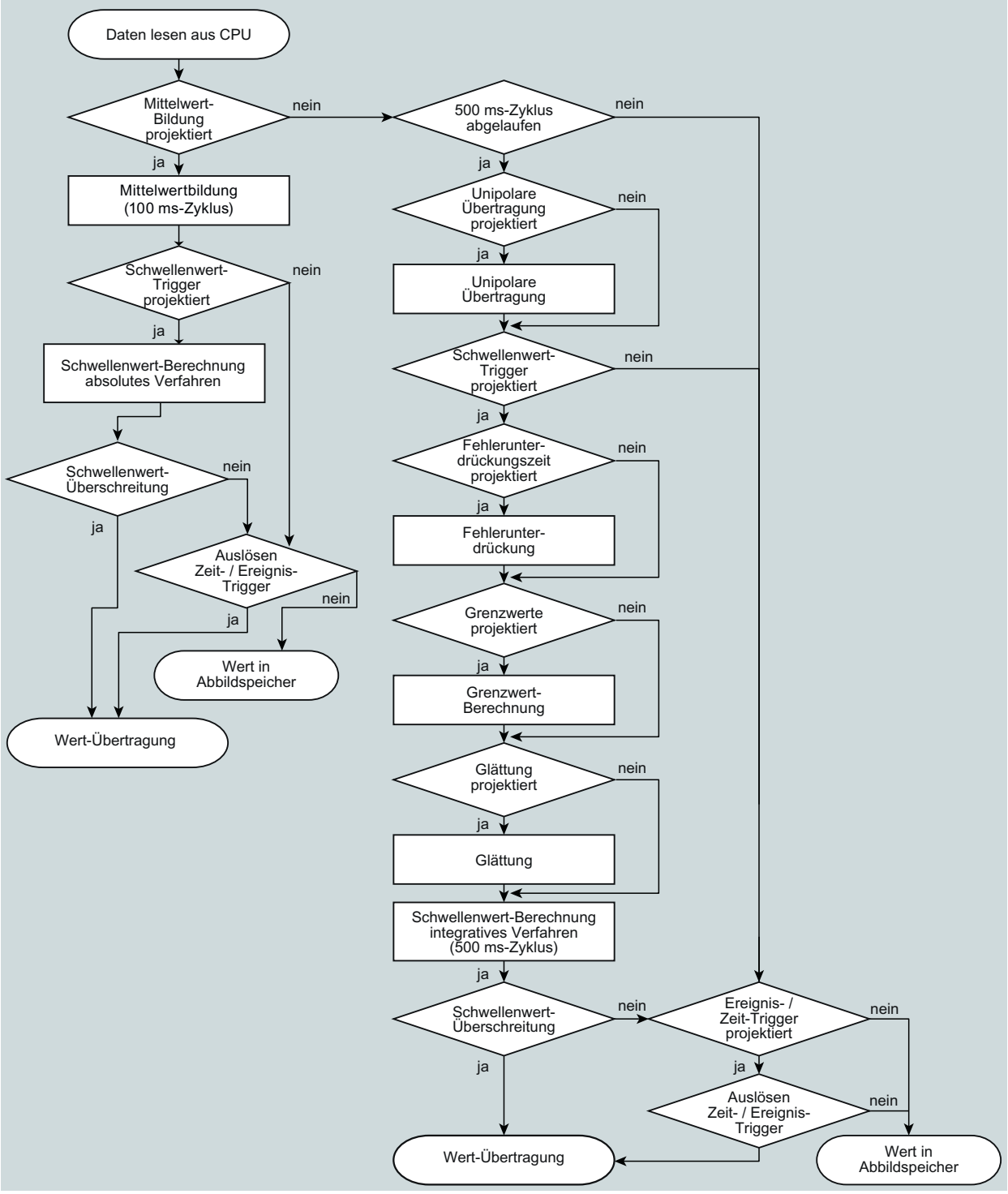

Bild 1-80 Ablauf der Analogwert-Vorverarbeitung

Der 500-Millisekunden-Zyklus wird durch die integrative Schwellenwert-Berechnung aufgesetzt. In diesem Zyklus werden die Werte auch bei Aktivierung der folgenden Vorverarbeitungsoptionen gespeichert:

- Unipolare Übertragung
- Fehlerunterdrückungszeit
- Grenzwertberechnung
- Glättung

Mittelwertbildung

Hinweis

Eingeschränkte Vorverarbeitungsoptionen bei projektierter Mittelwertbildung

Wenn Sie für ein Analogwertereignis die Mittelwertbildung projektieren, dann stehen folgende Vorverarbeitungsoptionen nicht zur Verfügung:

- Unipolare Übertragung
- Fehlerunterdrückungszeit
- Glättung

Funktion

Mit diesem Parameter werden erfasste Analogwerte als Mittelwerte übertragen.

Bei aktivierter Mittelwertbildung ist die Projektierung eines Zeit-Triggers sinnvoll.

Die aktuell anstehenden Werte eines Analogwert-Datenpunkts werden im 100-Millisekunden-Zyklus gelesen und aufsummiert. Die Anzahl der gelesenen Werte pro Zeiteinheit ist abhängig vom Lesezyklus der CPU und vom CPU-Abtastzyklus des CP.

Aus den aufsummierten Werten wird der Mittelwert gebildet, sobald die Übertragung über einen Trigger ausgelöst wird. Danach wird die Aufsummierung zur Bildung des nächsten Mittelwerts neu gestartet.

Der Mittelwert wird auch dann gebildet, wenn die Übertragung des Analogwerttelegramms durch einen Aufruf des Kommunikationspartners ausgelöst wird. Die Dauer der Mittelungsperiode ist dann die Zeit von der letzten Übertragung (z. B. über den Trigger ausgelöst) bis zum Zeitpunkt des Aufrufs. Auch nach dieser Übertragung wird die Aufsummierung zur Bildung des nächsten Mittelwertes neu gestartet.

Eingabebaugruppen: Überlaufbereich / Unterlaufbereich

Sobald ein Wert im Überlauf- bzw. Unterlaufbereich erfasst wird, wird die Mittelwertbildung abgebrochen. Für die laufende Mittelungsperiode wird der Wert 32767 / $7FFF_h$ bzw. -32768 / 8000_h als ungültiger Mittelwert gespeichert und im nächsten Daten-Telegramm übertragen.

Danach wird eine neue Mittelwertbildung gestartet. Wenn der Analogwert dann immer noch im Überlauf- oder Unterlaufbereich liegt, wird wieder einer der beiden genannten Werte als ungültiger Mittelwert gespeichert und bei der nächsten Übertragungsauslösung gesendet.

Hinweis

Fehlerunterdrückungszeit > 0 projektiert

Wenn Sie eine Fehlerunterdrückungszeit projektiert haben und danach die Mittelwertbildung aktivieren, dann wird der Wert der Fehlerunterdrückungszeit ausgegraut aber nicht mehr angewendet. Die Fehlerunterdrückungszeit wird bei aktivierter Mittelwertbildung intern auf 0 (null) gesetzt.

Unipolare Übertragung

Restriktionen

Unipolare Übertragung kann nicht gleichzeitig mit der Mittelwertbildung projektiert werden. Die Aktivierung der unipolaren Übertragung wird bei Aktivierung der Mittelwertbildung unwirksam.

Funktion

Bei der unipolaren Übertragung werden negative Werte auf Null korrigiert. Dies kann gewünscht sein, wenn Werte aus dem Untersteuerungsbereich nicht als reale Messwerte übertragen werden sollen.

Ausnahme: Bei Prozessdaten von Eingabebaugruppen wird der Wert -32768 / 8000_h für Drahtbruch eines Life-Zero-Eingangs übertragen.

Bei einem Software-Eingang dagegen werden alle Werte kleiner Null auf Null korrigiert.

Fehlerunterdrückungszeit

Voraussetzungen für die Funktion

Projektierung des Schwellenwert-Triggers für diesen Datenpunkt

Restriktionen

Die Fehlerunterdrückungszeit kann nicht gleichzeitig mit der Mittelwertbildung projektiert werden. Ein projektierter Wert wird bei Aktivierung der Mittelwertbildung unwirksam.

Funktion

Typischer Anwendungsfall für diesen Parameter ist die Unterdrückung von Spitzenstromwerten beim Anlauf leistungsstarker Motoren, die sonst als Störung an die Leitstelle gemeldet würden.

Die Übertragung eines Analogwerts im Überlauf ($7FFF_h$) oder Unterlauf (8000_h) wird für die Dauer der angegebenen Zeitspanne unterdrückt. Erst nach Ablauf der Fehlerunterdrückungszeit wird der Wert von $7FFF_h$ bzw. 8000_h übertragen, falls er noch ansteht.

Wenn der Wert vor Ablauf der Fehlerunterdrückungszeit in den Bemessungsbereich zurückfällt, wird der aktuelle Wert übertragen.

Eingabebaugruppen

Die Unterdrückung ist abgestimmt auf Analogwerte, die als Rohwerte direkt von S7-Analog-Eingabebaugruppen erfasst werden. Diese Baugruppen liefern für alle Eingangsbereiche die genannten Werte für den Überlauf- bzw. Unterlaufbereich, auch für Life-Zero-Eingänge.

Ein Analogwert im Überlaufbereich (32767 / $7FFF_h$) oder Unterlaufbereich (-32768 / 8000_h) wird für die Dauer der Fehlerunterdrückungszeit nicht übertragen. Dies gilt auch für Life-Zero-Eingänge. Erst nach Ablauf der Fehlerunterdrückungszeit wird der Wert im Über- bzw. Unterlaufbereich übertragen, falls er noch ansteht.

Empfehlung zu Fertigwerten, die von der CPU vorverarbeitet wurden:

Wenn von der CPU vorverarbeitete Fertigwerte im Merkerbereich oder in einem Datenbaustein bereitgestellt werden, dann ist eine Unterdrückung nur möglich bzw. sinnvoll, wenn die Fertigwerte im Überlauf bzw. Unterlauf ebenfalls die genannten Werte von 32767 / $7FFF_h$ bzw. -32768 / 8000_h annehmen. Wenn dies nicht der Fall ist, dann sollte der Parameter für vorverarbeitete Werte nicht projektiert werden.

Für bereits in der CPU vorverarbeitete Fertigwerte können die Grenzen für den Überlauf/ Unterlauf frei vergeben werden.

Glättungsfaktor

Voraussetzungen für die Funktion

Projektierung des Schwellenwert-Triggers für diesen Datenpunkt

Restriktionen

Der Glättungsfaktor kann nicht gleichzeitig mit der Mittelwertbildung projektiert werden. Ein projektierter Wert wird bei Aktivierung der Mittelwertbildung unwirksam.

Funktion

Schnell schwankende Analogwerte können mit Hilfe der Glättungsfunktion beruhigt werden.

Die Glättungsfaktoren werden wie bei S7-Analogeingabebaugruppen nach folgender Formel berechnet.

$$y_n = \frac{x_n + (k-1)\ y_{n-1}}{k}$$

mit
y_n = Geglätteter Wert im aktuellen Zyklus n
y_{n-1} = Geglätteter Wert im vorhergehenden Zyklus n-1
x_n = Erfasster Wert im aktuellen Zyklus n
k = Glättungsfaktor

Als Glättungsfaktor sind folgende Werte für die Baugruppe projektierbar.

- 1 = Keine Glättung
- 4 = Schwache Glättung
- 32 = Mittlere Glättung
- 64 = Starke Glättung

Grenzwert 'tief' setzen / Grenzwert 'hoch' setzen

Voraussetzungen für die Funktion

- Projektierung des Schwellenwert-Triggers für diesen Datenpunkt
- PLC-Variable im Operandenbereich Merker oder Datenbereich
  Der Analogwert-Datenpunkt muss mit einer PLC-Variable im Merker- oder Datenbereich (Datenbaustein) verknüpft sein. Für PLC-Variablen von Hardware-Baugruppen (Operandenbereich Eingang) ist die Grenzwertprojektierung nicht möglich.

Für Messwerte, die bereits in der CPU vorverarbeitet wurden, ist die Projektierung von Grenzwerten nicht sinnvoll.

Funktion

In diesen beiden Eingabefeldern können Sie jeweils einen Grenzwert in Richtung Messbereichsanfang bzw. in Richtung Messbereichsende setzen. Die Grenzwerte können Sie bspw. auch als Messbereichsanfang bzw. Messbereichsende auswerten.

Statuskennung "OVER_RANGE" / "overflow"

Bei Protokollen, die Statuskennungen unterstützen, wird bei Unter- bzw. Überschreitung des Grenzwerts die Statuskennung des Datenpunkts für Messbereichsüberschreitung gesetzt, nachfolgend als Kennung "OV" bezeichnet. Die Statuskennungen sind im Kapitel Statuskennungen der Datenpunkte (Seite 1869) beschrieben.

Das Bit "OV" der Statuskennung des Datenpunkts wird bei der Übertragung des betreffenden Analogwerts wie folgt gesetzt:

- Grenzwert 'hoch':
  - Bei Überschreitung des Grenzwerts: OV = 1
  - Bei anschließender Unterschreitung des Grenzwerts: OV = 0
- Grenzwert 'tief':
  - Bei Unterschreitung des Grenzwerts: OV = 1
  - Bei anschließender Überschreitung des Grenzwerts: OV = 0

Projektierung des Grenzwerts

Der Grenzwert wird als ganze Dezimalzahl projektiert. Der Wertebereich orientiert sich am Wertebereich des Rohwerts von Analogeingabebaugruppen.

Bereich	Rohwert (16 bit)		Baugruppenausgang [mA]			Messbereich [%]
	Dezimal	Hexadezimal	0 .. 20 (unipolar)	-20 .. +20 (bipolar)	4 .. 20 (life zero)	
Überlauf	32767	7FFF	> 23,515	> 23,515	> 22,810	> 117,593
Übersteuerungs-Bereich	32511 ... 27649	7EFF ... 6C01	23,515 ... 20,001	23,515 ... 20,001	22,810 ... 20,001	117,593 ... 100,004
Nennbereich (unipolar / life zero)	27648 ... 0	6C00 ... 0000	20 ... 0		20 ... 4	100 ... 0

Bereich	Rohwert (16 bit)		Baugruppenausgang [mA]			Messbereich [%]
	Dezimal	Hexadezimal	0 .. 20 (unipolar)	-20 .. +20 (bipolar)	4 .. 20 (life zero)	
Nennbereich (bipolar)	27648 ... 0 ... -27648	6C00 ... 0000 ... 9400		20 ... 0 ... -20		100 ... 0 ... -100
Untersteuerungs-Bereich (unipolar / life zero)	-1 ... -4864	FFFF ... ED00	-0,001 ... -3,518		3,999 ... 1,185	-0,004 ... -17,59
Untersteuerungs-Bereich (bipolar)	-27649 ... -32512	93FF ... 8100		-20,001 ... -23,516		-100,004 ... -117,593
Unterlauf / Drahtbruch	-32768	8000	< -3,518		< 1,185	< -17,593

Beachten Sie: Die Eingabe des Werts 0 (null) wird als deaktivierter Grenzwert interpretiert.

Empfehlung für schnell schwankende Analogwerte:

Wenn der Analogwert schnell schwankt, kann es bei projektierten Grenzwerten sinnvoll sein, den Analogwert zuvor zu glätten.

Masken

Maskierung einzelner Bits im Byte von Binäreingängen (Bin04B_S)

In der Parametergruppe "Masken" werden drei Optionen zur Übertragung von Binärwerten angeboten. Die entsprechenden Bits werden im Hexadezimal-Format maskiert.

Durch Eingabe hexadezimaler Werte kann Bit-individuell festgelegt werden, ob bestimmte Bits keine Übertragung auslösen oder welche Bits in einem anderen Übertragungsmodus übertragen werden.

- **Alarm-Maske**
  Maskierte Bits bewirken bei Änderung eine unbedingt spontane Übertragung. Es werden Änderungen von 0 nach 1 und von 1 nach 0 ausgewertet.
  Die Maskierung "Alarm-Maske" ist nur sinnvoll, wenn der Datenpunkt über ein Wählnetz mit aktivierter Option "Bedingt spontan" übertragen wird.
- **Sendepufferprinzip-Maske**
  Maskierte Bits bewirken bei Änderung eine Übertragung nach dem Sendepufferprinzip. Es werden Änderungen von 0 nach 1 und von 1 nach 0 ausgewertet.
  Die Sendepufferprinzip-Maske ist nur sinnvoll bei Datenpunkten mit aktivierter Option "Abbildspeicher".
- **Sperr-Maske**
  Maskierte Bits werden bei der Änderungskontrolle nicht berücksichtigt. Dies bewirkt, dass Änderungen der maskierten Bits für diesen Datenpunkt keine Übertragung auslösen. Ein maskiertes Bit hat im Daten-Telegramm grundsätzlich den Wert 0.

Beispiel für die Byte-Belegung

Die folgende Tabelle zeigt die Maskierung der Bits entsprechend des im Eingabefeld der Parameter eingetragenen Hexadezimalwerts "A3".

Tabelle 1-134 Byte-Belegung

Bit:	.7	.6	.5	.4	.3	.2	.1	.0
Maskiert	1	0	1	0	0	0	1	1
Hex-Maske	A				3			

Die mit Wert 1 belegten Bits sind maskiert, d. h. die Bits Nr. 0, 1, 5 und 7 bewirken bei den beschriebenen Masken die jeweilige Funktion.

Befehlsausgang

Verarbeitungsoptionen von Befehlsausgängen

Parameter

Für Datenpunkte in Stationen, die Befehle einer Zentrale-Station bzw. eines Masters ausgeben, können folgende Ausgabeoptionen alternativ aktiviert werden:

- LATCH_ON/OFF
  Die Funktion verriegelt einen Befehlsausgang dauerhaft auf den Wert 0 oder 1.
  Beachten Sie:
  Der verriegelte Wert wird erst durch einen neuen Befehl aufgehoben. Alternativ kann der Befehl vom Anwenderprogramm zurückgesetzt werden.
- PULSE_ON/OFF
  Die Funktion wertet die Anzahl und Länge der Signale (Pulse) von Befehlen aus einer Zentrale aus.

Datenpunkttypen

Die Ausgabeoptionen sind für folgende Datenpunkttypen projektierbar:

- ST7
  Befehlsausgang (Cmd01B_R)
- IEC
  Single command
- DNP3
  Binary Command Output

Bezug zu Spezifikationen

Die Ausgabeoptionen entsprechen folgenden Teilen der Spezifikation:

- IEC
  Qualifier of command - IEC 60870-5-101, Teil 1
- DNP3
  Control Code, Operation Type field (OP Type) - Volume 6, Part 2, Objects

Protokoll-abhängige Verarbeitung

ST7

- LATCH_ON/OFF
  Die Funktion verriegelt den Wert eines Befehls wie oben beschrieben.
- PULSE_ON/OFF
  Die Funktion wertet für einen einzigen Puls die "Befehlsausgangs-Ausgabezeit" aus (siehe unten).

IEC

- LATCH_ON/OFF
  Die Funktion verriegelt den Wert eines Befehls wie oben beschrieben.
- PULSE_ON/OFF
  Die Funktion wird vom Master über einen "Qualifier of command" kodiert. Folgende Kodierungen werden vom Kommunikationsmodul in der Station ausgewertet:
  - QU (Type 1.1) <0> no additional definition
    Korrespondierender Parameter beim Modul: "Puls-Steuerung"
  - QU (Type 1.1) <1> short pulse duration
    Korrespondierender Parameter beim Modul: "Kurze Pulsdauer"
  - QU (Type 1.1) <2> long pulse duration
    Korrespondierender Parameter beim Modul: "Lange Pulsdauer"
  - S/E (Type 6) <0> execute
    Korrespondierender Parameter beim Modul: "Befehlsmodus"
  - S/E (Type 6) <1> select
    Korrespondierender Parameter beim Modul: "Befehlsmodus"

DNP3

Die beiden Funktionen werden vom Master über den Control Code des Objekts kodiert.

- LATCH_ON/OFF
  Die Funktion verriegelt einen Befehlsausgang dauerhaft auf den Wert 0 oder 1.
  Die Station verarbeitet den Befehl nach dem Complementary Latch Model.
- PULSE_ON/OFF
  Die Funktion wertet in Befehls-Telegrammen eines Masters den Control Code "PULSE_ON" aus.
  Der Master sendet die Informationen "Trip-Close Count", "On-time", "Off-time" und "Pulse count" mit dem Befehls-Telegramm an die Station.
  Der Stations-Datenpunkt wertet diese Informationen aus und prüft diese auf Konistenz mit den Parametern "Max. Pulsdauer", "Pulsdauer-Ersatzzeit" und "Max. Pulsanzahl".
  Die TIM 1531 IRC verarbeitet den Befehl nach dem Activation Model.

Das "Clear"-Feld (CR) wird nicht unterstützt.

Der Stations-Datenpunkt wertet die Kodierung des Control Codes folgendermaßen ausgewertet.

Tabelle 1-135 Dekodierung des Control Codes durch den Befehls-Datenpunkt der Station

Empfangener Control Code			Reaktion des Stations-Datenpunkts	
Control Code	Trip-Close Code	Operation Type Field	Aktivierte Option: PULSE_ON	Aktivierte Option: LATCH_ON / OFF
0x01	NUL	PULSE_ON	Der Ausgang wird für die Dauer von "On-time" auf 1 gesetzt.	Der Befehl wird abgelehnt.
0x03	NUL	LATCH_ON	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird für die Dauer von "On-time" auf 1 gesetzt.	Der Ausgang wird auf 1 gesetzt.
0x04	NUL	LATCH_OFF	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird für die Dauer von "On-time" auf 1 gesetzt.	Der Ausgang wird auf 0 gesetzt.
0x41	CLOSE	PULSE_ON	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird für die Dauer von "On-time" auf 1 gesetzt.	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird auf 1 gesetzt.
0x81	TRIP	PULSE_ON	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird für die Dauer von "On-time" auf 1 gesetzt.	• CP Der Befehl wird abgelehnt. • TIM 1531 IRC Der Ausgang wird auf 0 gesetzt.

Hinweis

Rückspiegelung

Wenn der Stations-Datenpunkt den aktuellen Wert an den Master zurückspiegeln soll, dann aktivieren Sie im Register "Allgemein" des Datenpunkts die Option "Wertüberwachung".

Parameter

Name: **Control Code**
Wertebereich:
- PULSE_ON
- LATCH_ON/OFF

Erläuterung: Ausgabeoption des Befehlsausgangs. Zur Bedeutung siehe oben.

Name: **Puls-Steuerung**
(nur Master)
- DNP3: Trip-Close Code
- IEC: Qualifier of command

Wertebereich: DNP3:
- NUL
- CLOSE
- TRIP

IEC:
- Keine Vorgaben
  Zu den weiteren Qualifiern siehe Parameter "Kurze Pulsdauer / Lange Pulsdauer" unten.

Erläuterung: Trip-Close Code für den Control Code "PULSE_ON". Der Datenpunkt überträgt den projektierten Code an den korrespondierenden Datenpunkt der Station. Zur Bedeutung siehe oben.

Name: **Puls-EIN-Dauer**
(nur Master)
Wertebereich: 0 ... 4294967295
Vorbelegung: 10
Erläuterung: Zeitdauer (Millisekunden) für die "On-time". Der Datenpunkt überträgt die projektierte Zeit an den korrespondierenden Datenpunkt der Station.

Name: **Puls-AUS-Dauer**
(nur Master)
Wertebereich: 0 ... 4294967295
Vorbelegung: 0
Erläuterung: Zeitdauer (Millisekunden) für die "Off-time". Der Datenpunkt überträgt die projektierte Zeit an den korrespondierenden Datenpunkt der Station.

Name: **Puls-Anzahl**
(nur DNP3-Master)
Wertebereich: 1 ... 255

Vorbelegung: 1

Erläuterung: Anzahl der Pulse, die der Datenpunkt an den korrespondierenden Datenpunkt der Station überträgt.

Name: **Max. Pulsanzahl**

Wertebereich:
- DNP3: 0 ... 255
- ST7 und IEC: Feste Belegung mit "1"

Vorbelegung: 1

Erläuterung: Überwacht die vom Master gesendete Anzahl der Pulse (Count). Wenn die vom Master empfangene Anzahl der Pulse den hier projektierten Wert übersteigt, wird der Befehl verworfen.

Bei 0 (null) ist die Überwachung abgeschaltet.

Name: **Max. Pulsdauer (s)**

(nur DNP3)

Wertebereich: 0 ... 65535

Vorbelegung: 0

Erläuterung: Überwacht die vom Master im Befehls-Telegramm übertragene Pulsdauer (On-time).

Wenn die Pulsdauer den hier projektierten Wert überschreitet, wird der Befehl vom Stations-Modul verworfen, sofern keine "Pulsdauer-Ersatzzeit" projektiert ist.

Bei Überschreitung der "Max. Pulsdauer" und projektierter "Pulsdauer-Ersatzzeit" oder bei einer empfangenen Pulsdauer von 0 wird die "Pulsdauer-Ersatzzeit" verwendet.

Name: **Pulsdauer-Ersatzzeit (s)**

(nur DNP3)

Wertebereich: 0 ... 65535

Vorbelegung: 0

Erläuterung: Ersatzwert für die Pulsdauer

Der Ersatzwert wird in folgenden Fällen verwendet:

- Die vom Master empfangene Pulsdauer übersteigt den unter "Max. Pulsdauer" projektierten Wert.
- Die empfangene Pulsdauer ist gleich 0 ist.

Bei 0 (null) wird kein Ersatzwert angewendet. Der Befehl wird verworfen.

Wenn "Max. Pulsdauer" und "Pulsdauer-Ersatzzeit" beide mit Null projektiert sind, wird jeder Befehl verworfen.

Name: **Kurze Pulsdauer (s)**

(nur IEC)

Wertebereich: 0 ... 65535

Vorbelegung: 0

Erläuterung: Befehle vom Master mit dem Qualifier of command (QOC) = <1> (short pulse duration) werden vom Kommunikationsmodul für die hier projektierte Zeitdauer ausgegeben.

Bei 0 (null) werden Befehle mit QOC = <1> vom Modul verworfen.

Name: **Lange Pulsdauer (s)**

(nur IEC)

Wertebereich: 0 ... 65535

Vorbelegung: 0

Erläuterung: Befehle vom Master mit dem Qualifier of command (QOC) = <2> (long pulse duration) werden vom Kommunikationsmodul für die hier projektierte Zeitdauer ausgegeben.

Bei 0 (null) werden Befehle mit QOC = <2> vom Modul verworfen.

Name: **Befehlsausgangs-Ausgabezeit (s)**

(nur ST7)

Wertebereich: 0 ... 65535

Vorbelegung: 0

Erläuterung: Zeitdauer, für die der Befehl ausgegeben wird. Die Zeitdauer gilt für alle 8 Kanäle des SINAUT-Objekts.

Nach Ablauf der Zeit wird der Ausgang automatisch zurückgesetzt.

Bei Null wird ein gesetzter Befehlsausgang nicht automatisch zurückgesetzt. In diesem Fall müssen Sie den Ausgang über das Anwenderprogramm zurücksetzen.

Name: **Befehlsmodus**

(nur DNP3 und IEC)

Wertebereich:
- Direkte Ausführung
- Anwahl und Ausführung

Vorbelegung: Direkte Ausführung

Erläuterung:

- Direkte Ausführung
  Der Befehl wird sofort an die CPU der Station zur Ausführung übertragen.
- Anwahl und Ausführung
  Ablauf:
  - Auslösen des Befehls im Master-Modul
    Das "Anwahl"-Telegramm wird von der Zentrale an das Kommunikationsmodul der Station übertragen.
  - Die Station quittiert den Empfang.
  - Der Master-Datenpunkt sendet das Ausführungs-Telegramm nach Empfang der Quittung der Station.
  - Die Station leitet den Befehl erst dann an die CPU weiter, wenn sie innerhalb der projektierten "Max. Zeit zwischen Select und Operate" das "Ausführungs"-Telegramm des Masters empfängt.
    Die Station darf zwischen Anwahl und Ausführung kein anderes Daten-Telegramm empfangen.

  Hinweis: "Max. Zeit zwischen Select und Operate" wird in den Übertragungseinstellungen der jeweiligen Schnittstelle projektiert.
  Bezug zu den Spezifikationen:
  - DNP3
    "Select and Operate" wird über die Funktions-Codes 03 und 04 realisiert.
    "Direct Operate" wird über die Funktions-Codes 05 und 06 realisiert.
  - IEC (Select and Execute)
    Das Kommunikationsmodul quittiert den Empfang der Anwahl-ASDU mit dem Qualifier "S/E (Type 6) <1> select".
    Der Master sendet nach Empfang der Quittung die Ausführungs-ASDU mit dem Qualifier "S/E (Type 6) <1> execute".

Nachrichten-Editor

Bearbeitungsstatus von Nachrichten (E-Mails / SMS)

Statuskennung aktivieren / Externe Status

Bei aktivierter Option schreibt der CP einen Bearbeitungsstatus zur gesendeten E-Mail / SMS in eine PLC-Variable.

Die gelieferten Status haben folgende Bedeutung:

Tabelle 1-136 SMS: Bedeutung der hexadezimal ausgegebenen Statuskennung

Status	Bedeutung
0000	Übertragung fehlerfrei abgeschlossen
0001	Fehler bei der Übertragung; mögliche Ursachen: • SIM-Karte nicht gültig • Kein Netz • Falsche Zielrufnummer (Nummer nicht erreichbar)

Tabelle 1-137 E-Mails: Bedeutung des hexadezimal ausgegebenen Bearbeitungsstatus

Status	Bedeutung
0000	Übertragung fehlerfrei abgeschlossen
82xx	Sonstige Fehlermeldung vom E-Mail-Server Bis auf die führende "8" entspricht der Status der dreistelligen Fehlernummer des Protokolls SMTP.
8401	Kein Kanal verfügbar Mögliche Ursache: Es besteht bereits eine E-Mail-Verbindung über den CP. Eine zweite Verbindung kann nicht parallel eingerichtet werden.
8403	Es konnte keine TCP/IP-Verbindung zum SMTP-Server aufgebaut werden.
8405	Der SMTP-Server hat die Login-Anfrage verweigert.
8406	Ein interner SSL-Fehler oder ein Problem mit der Struktur des Zertifikats wurde durch den SMTP-Client festgestellt.
8407	Anfrage zur Verwendung von SSL wurde verweigert.
8408	Der Client konnte kein Socket zur Erstellung einer TCP/IP-Verbindung zum Mail-Server ermitteln.
8409	Über die Verbindung kann nicht geschrieben werden. Mögliche Ursache: Durch den Kommunikationspartner wurde ein Reset der Verbindung durchgeführt oder die Verbindung wurde abgebrochen.
8410	Über die Verbindung kann nicht gelesen werden. Mögliche Ursache: Durch den Kommunikationspartner wurde die Verbindung abgebaut oder die Verbindung wurde abgebrochen.
8411	Senden der E-Mail fehlgeschlagen. Ursache: Speicherplatz war nicht ausreichend, um den Sendevorgang durchzuführen.
8412	Konfigurierter DNS-Server konnte den angegebenen Domain-Namen nicht auflösen.
8413	Aufgrund eines internen Fehlers im DNS-Subsystem konnte der Domain-Name nicht aufgelöst werden.
8414	Als Domain-Name wurde eine leere Zeichenkette angegeben.
8415	Ein interner Fehler ist im Curl-Modul aufgetreten. Ausführung wurde abgebrochen.
8416	Ein interner Fehler ist im SMTP-Modul aufgetreten. Ausführung wurde abgebrochen.
8417	Anfrage an SMTP auf bereits verwendetem Kanal oder ungültige Kanal-ID. Ausführung wurde abgebrochen.
8418	Senden der E-Mail wurde abgebrochen. Mögliche Ursache: Überschreitung der Ausführungszeit.
8419	Der Kanal wurde unterbrochen und kann nicht verwendet werden, bevor die Verbindung abgebaut wird.

Status	Bedeutung
8420	Zertifikatskette vom Server konnte nicht mit dem Root-Zertifikat des CP verifiziert werden.
8421	Interner Fehler aufgetreten. Ausführung wurde gestoppt.
8450	Aktion nicht ausgeführt: Mailbox nicht verfügbar / nicht erreichbar. Versuchen Sie es später noch einmal.
84xx	Sonstige Fehlermeldung vom E-Mail-Server Bis auf die führende "8" entspricht der Status der dreistelligen Fehlernummer des Protokolls SMTP.
8500	Syntax-Fehler: Kommando unbekannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann sein, dass der E-Mail-Server das Authentifizierungsverfahren LOGIN nicht unterstützt. Versuchen Sie, E-Mails ohne Authentifizierung zu versenden (kein Benutzername).
8501	Syntax-Fehler. Überprüfen Sie die folgenden Projektierungsdaten: Nachrichtenkonfiguration > Nachrichtenparameter: • Empfängeradresse ("An" bzw. "Cc").
8502	Syntax-Fehler. Überprüfen Sie die folgenden Projektierungsdaten: Nachrichtenkonfiguration > Nachrichtenparameter: • E-Mail-Adresse (Absender)
8535	SMTP-Authentifizierung unvollständig. Überprüfen Sie in der CP-Projektierung die Parameter "Benutzername" und "Passwort".
8550	SMTP-Server kann nicht erreicht werden. Sie haben keine Zugriffsrechte. Überprüfen Sie die folgenden Projektierungsdaten: • CP-Projektierung > E-Mail-Projektierung: – Benutzername – Passwort – E-Mail-Adresse (Absender) • Nachrichtenkonfiguration > Nachrichtenparameter: – Empfängeradresse ("An" bzw. "Cc").
8554	Übertragung fehlgeschlagen
85xx	Sonstige Fehlermeldung vom E-Mail-Server Bis auf die führende "8" entspricht der Status der dreistelligen Fehlernummer des Protokolls SMTP.

Protokoll-spezifische Parameter

TeleControl Basic

CP-Identifikation

Bezug

CPs mit Nutzung des Telecontrol-Protokolls "TeleControl Basic".

Parametergruppe "CP-Identifikation"

Eine Projektnummer für das gesamte STEP 7-Projekt

Wenn Sie bei einem CP die Projektnummer in der Parametergruppe "CP-Identifikation" ändern, dann wird dieser Parameter bei allen CPs mit Nutzung des Protokolls "TeleControl Basic" im gesamten STEP 7-Projekt verändert.

TCSB wertet Projektnummern von 1 ... 2000 aus.

Projektierung der Stationsnummer

Für jede Station im STEP 7-Projekt, die einen CP mit Nutzung des Protokolls "TeleControl Basic" verwendet, müssen Sie eine eindeutige Stationsnummer projektieren.

TCSB wertet Stationsnummern von 1 ... 8000 aus.

Zugangs-ID

Die in der Parametergruppe "CP-Identifikation" angezeigte Zugangs-ID wird aus den hexadezimalen Werten von Projektnummer, Stationsnummer und Steckplatz gebildet. Der Parameter vom Typ DWORD ist folgendermaßen belegt:

- Bit 0 - 7: Steckplatz
- Bit 8 - 20: Stationsnummer
- Bit 21 - 31: Projektnummer

Adress- und Authentifizierungsdaten

IP-Adresse des CP

Da der CP generell die Verbindung mit TCSB aufbaut, kann für den CP eine dynamische IP-Adresse durch den Internet-Dienstbetreiber vergeben werden.

Adress- und Authentifizierungsdaten für die Kommunikation mit TCSB

Folgende Informationen werden bei der STEP 7-Projektierung des CP für die Kommunikation mit TCSB benötigt:

- Parameter in der Parametergruppe "Partnerstationen"
  - Partner-IP-Adresse
    IP-Adresse oder Host-Name des DSL-Routers, über den der Telecontrol-Server an das Internet angeschlossen ist.
    Eine feste IP-Adresse wird empfohlen.
  - Partner-Port (Port-Nummer des Listener-Ports von TCSB)
- Parameter in der Parametergruppe "Security > CP-Identifikation"
  - Projektnummer
  - Stationsnummer
  - Passwort (zur Authentifizierung)

Quittierung

Quittierung bei TeleControl Basic

Der Empfang eines gesendeten Telegramms wird folgendermaßen quittiert.

- Telecontrol-Kommunikation
  Von TCSB empfangene Telegramme werden vom CP sofort quittiert.
  Vom CP gesendete Telegramme werden von TCSB quittiert.
- Querkommunikation
  Empfangene Telegramme werden vom CP sofort quittiert. Das Quittungstelegramm wird vom Telecontrol-Server an den Ziel-CP weitergeleitet.
  Für gesendete Telegramme gilt dies entsprechend in umgekehrter Richtung.

SINAUT ST7

Die TD7-Software

Aufgaben der TD7-Software

Die Software "SINAUT TD7" wickelt in Kommunikationsmodulen, die das Fernwirkprotokoll SINAUT ST7 nutzen, die Kommunikation und die Prozessdatenübertragung zwischen den Teilnehmern ab. Nur mithilfe der TD7-Software kann Datenkommunikation über WAN-Verbindungen zwischen ST7-fähigen CPs und TIM-Baugruppen realisiert werden.

TD7 sorgt für eine änderungsgesteuerte Prozessdatenübertragung zwischen Station und Leitstelle (z. B. ST7cc), zwischen einzelnen Stationen und zwischen Kommunikationsmodul und CPU. Ausfälle von Verbindungen, von CPUs oder der Leitstelle werden angezeigt. Nach Störungsbehebung und im Anlauf von CPUs oder der Leitstelle werden die Daten automatisch aktualisiert.

Varianten der TD7-Software

Die TD7-Software gibt es in zwei Varianten. In Kommunikationsmodulen, die beide TD7-Varianten unterstützen, kann nur eine Variante alternativ verwendet werden.

- **TD7onTIM**
  TD7onTIM wird über die Datenpunkte des Moduls projektiert. Im Produktivbetrieb läuft sie im Kommunikationsmodul ab.
  Die Funktionen und Parameter sind in den Kapiteln zur Projektierung der Datenpunkte beschrieben.
  Die Variante ist in allen ST7-fähigen Modulen einsetzbar.
  Vorteile dieser Variante sind:
  - Leichte Projektierbarkeit
  - Unabhängigkeit der TD7-Software vom CPU-Programm
  - Entlastung der CPU bei geringen Reserven des CPU-Speichers
- **TD7onCPU**
  TD7onCPU ist als Programmbaustein-Bibliothek ausgeführt. Die Bausteine werden in der CPU, die der TIM zugeordnet ist, verwendet und parametriert. Die Software läuft im Produktivbetrieb in der CPU ab.
  TD7onCPU ist bei folgenden Modulen einsetzbar:
  - TIM 1531 IRC
  - TIM 3V-IE Advanced
  - TIM 4R-IE / TIM 4R-IE Standalone

  Vorteile dieser Variante sind:
  - Höhere Geschwindigkeit bei Datenänderung
    Änderungen werden mit jedem CPU-Zyklus erfasst.
  - Größere Auswahl an Daten-Typicals für die Zentrale und die Station

Die TD7onCPU-Bibliotheken

TD7onCPU steht in zwei Baustein-Bibliotheken für folgende SIMATIC-Familien zur Verfügung:

- S7-1500
  Die Bausteine sind auf einer S7-1500 lauffähig.
- S7-300 / 400
  Die Bausteine sind auf einer S7-300 und auf einer S7-400-CPU lauffähig. Ausnahmen werden explizit angegeben.
  Außer für die Prozessdatenübertragung über WAN eignet sich das Paket auch für die lokale Kommunikation zwischen CPUs, wenn diese über MPI miteinander verbunden sind. Auch hier werden die lokalen Verbindungen und CPUs ständig überwacht und es wird eine automatische Aktualisierung nach Anlauf oder Störungsbehebung durchgeführt.

Die Beschreibung der TD7onCPU finden Sie im Kapitel AUTOHOTSPOT.

Kommunikation zwischen SINAUT-Objekten

Kommunikation zwischen Objekten bzw. Typicals

- **Objekt / Typical**
  Der Begriff "Objekt" bezeichnet unter SINAUT ST7 die Art der Darstellung und Behandlung von Prozessgrößen wie Meldungen, Analogwerte, Befehle, Motoren, Ventile, Regler usw. Die TD7-Software verarbeitet grundsätzlich "Objekte".
  Objekte in der TD7-Welt bestehen aus Verarbeitungsvorschriften in Form von Programmbausteinen – Typicals genannt.
  Ein Objekt besteht aus folgenden Komponenten:
  - Ein Funktionsbaustein (FB)
  - Ein Datenbaustein (DB)
    Der DB ist der Objekt-Datensatz, welcher als Instanz-DB dem FB zugeordnet ist.

Ein Objekt bzw. ein Typical setzt sich immer aus einem Prozess-Anteil und einem Bedienungs- und Beobachtungs-Anteil (Operator-Anteil) zusammen, die in unterschiedlichen Teilnehmern des SINAUT-Netzes arbeiten. Um ihre vorbestimmte Funktion bearbeiten zu können, müssen die beiden Anteile miteinander kommunizieren.

Diese beiden Anteile werden in folgenden Teilnehmertypen eingesetzt und haben folgende Aufgaben:

- **Operator-Typicals**
  Verwendung in Zentrale-Stationen (Operator-Seite)
  Diese Bausteine schicken Sollwerte, Parameter, Befehle und organisatorische Anweisungen an die Prozess-Typicals.
- **Prozess-Typicals**
  Verwendung in Stationen (Prozess-Seite)
  Diese Bausteine verarbeiten Prozesswerte und geben Prozessdaten, Alarm- und Zustandsmeldungen sowie organisatorische Informationen an die Operator-Typicals zurück.

Die Daten, welche das Objekt beschreiben, werden zwischen den beiden Kommunikationspartnern ausgetauscht. Die Daten liegen im Objekt-Datensatz in den Daten-Indizes 0...n. Umfang und Zusammensetzung dieses Datenbereichs sind abhängig vom jeweiligen Typical. Es kann aus mehreren identischen oder aus einer Kombination von unterschiedlichen Datentypen bestehen. Im Objekt-Datensatz auf der Prozess- und Operator-Seite ist die Datenstruktur von zwei zusammengehörenden Typicals identisch.

Der Datenaustausch muss nicht zwangsweise in beiden Richtungen verlaufen.

Objekt-Adressierung

- **Teilnehmernummer**
  Jede CPU mit TD7-Software bekommt eine netzweit eindeutige SINAUT-Teilnehmernummer, die im Bereich von 1 bis 32000 vergeben werden kann.
- **Objektnummer**
  Jedes Typical, das in einer dieser CPUs aufgerufen wird, hat einen Instanz-DB, dessen Nummer identisch mit der SINAUT-Objektnummer ist.

Mit Hilfe der SINAUT-Teilnehmernummer und der SINAUT-Objektnummer kann eine eindeutige Adressierung für die Kommunikation zwischen den zusammengehörenden Typicals realisiert werden.

Alle Datenpunkt-Typicals sind Funktionsbausteine (FBs). Beim Aufruf eines FB ist ein Instanz-DB anzugeben. Die Nummer dieses Instanz-DB ist identisch mit der Objekt-Nummer des Datenpunkt-Objekts. Auf Sende- und Empfangsseite muss diese Objekt-Nummer nicht identisch sein.

Jedes Typical hat zur Festlegung der Kommunikationsbeziehung die folgenden Parameter:

- **Partnernummer**
  "PartnerNo"
  Teilnehmernummer des Partners, mit dem kommuniziert wird.
- **Partnerobjektnummer**
  "PartnerObcetcNo"
  Objektnummer (= Instanz-DB-Nr.) bei diesem Partner
- **Index**
  Für die Ablage wird noch die im Daten-Telegramm enthaltene Indexnummer berücksichtigt.

Telegrammarten

Neben dem Austausch der Prozess- und Operator-Daten werden auch organisatorische Informationen (Org.-Telegramme) zwischen beiden Seiten übertragen. Folgende Telegrammarten (TA) werden unterschieden:

- TA 0: Spontane Org.-Telegramme
- TA 1: Abgefragte Org.-Telegramme
- TA 2: Spontane Daten-Telegramme
- TA 3: Abgefragte Daten-Telegramme

Der Datenfluss der Org.-Telegramme sowie die dafür vorgesehen Datenbereiche im Objekt-Datensatz (Org.-Indizes) sind im obigen Bild nicht dargestellt.

Adressparameter beim Protokoll "ST7"

Adressierung der Kommunikationspartner und Netzknoten

Die Kommunikation beim Protokoll "SINAUT ST7" kann über verschiedene Wege und Subnetze laufen. Zur Adressierung der einzelnen Teilnehmer im ST7-Netz dienen die folgenden zwei Parameter:

- Teilnehmernummer
  Die Teilnehmernummer ist für jeden Teilnehmer in einem STEP 7-Projekt eindeutig. Folgende Teilnehmer benötigen eine Teilnehmernummer:
  - Kommunikationsmodul
    ST7-fähige Baugruppen (TIM, CP)
  - CPU
    Die lokale CPU, welche der Kommunikationsbaugruppe zugeordnet ist, erhält als Endpunkt einer Telecontrol-Verbindung über das Protokoll ST7 eine Teilnehmernummer.
    Bei einem CP und einer TIM 3V-IE / TIM 4R-IE ist die lokale CPU automatisch über das Rack zugeordnet.
    Der TIM 1531 IRC ordnen Sie die CPU in der Parametergruppe "Teilnehmernummern" der TIM zu.
  - Applikation des Zentrale-PC
    Die Applikation, welche Sie in der PC-Station der Zentrale projektieren.

  Für die TIM und die zugeordnete CPU wird die Teilnehmernummer in der Parametergruppe "Teilnehmernummern" der TIM projektiert.
  Für die Applikation des Zentrale-PC wird die Teilnehmernummer in der Telecontrol-Verbindung projektiert: Editor "Netzwerkdaten" > Register "Telecontrol"
- WAN-Adresse
  Für jede vernetzte serielle Schnittstelle in einem klassischen ST7-Netz wird eine WAN-Adresse vergeben. Sie ist im jeweiligen WAN-Netz eindeutig.
  Da Telecontrol-Verbindungen über mehrere Netzknoten und Knotenstationen laufen können, ist die netzweit eindeutige Adressierung der beteiligten Schnittstellen erforderlich.
  Die WAN-Adresse wird in der Parametergruppe "WAN-Parameter" der seriellen Schnittstelle projektiert.

Adressierung der Daten

Die zu übertragenen Prozessdaten werden als Datenpunkte projektiert, siehe Kapitel ST7-Datenpunkte (Seite 1866).

Quittierung

Quittierung bei ST7

- Kommunikation mit der Zentrale
  - Von der Zentrale gesendete Daten-Telegramme werden vom CP beim Empfang sofort quittiert.
  - Vom Kommunikationsmodul gesendete Daten-Telegramme werden von der Zentrale quittiert.
- Querkommunikation zwischen Stationen
  Daten-Telegramme werden wie bei der Kommunikation zwischen Station und Zentrale quittiert.
  Quittungstelegramme des empfangenden Moduls werden bei der Querkommunikation nicht von der Zentrale an das sendende Kommunikationsmodul weitergeleitet.

Erweiterung von SINAUT-Projekten

Erweiterung bestehender SINAUT-Projekte

SINAUT-Projekte mit TIM-Baugruppen für die SIMATIC-Familien S7-300 und 400, welche in STEP 7 V5 projektiert wurden, können mit neueren Baugruppen erweitert werden, die in STEP 7 Professional projektiert werden.

Hierzu werden in STEP 7 V5-Projekten TIM-Baugruppen als Platzhalter (Proxies) für Module konfiguriert, welche in STEP 7 Professional projektiert werden. Die Projektierungsdaten (SDBs) werden aus STEP 7 V5 exportiert und in STEP 7 Professional importiert.

Voraussetzung für den Import von Projektierungsdaten aus STEP 7 V5 ist die Auswahl des Projektierungsverfahrens "Importieren" in der Parametergruppe "Grundeinstellungen" , siehe Kapitel Projektierung (Seite 1802).

Die Erweiterung von SINAUT-Projekten wird durch folgende Baugruppen unterstützt:

Tabelle 1-138 Modul-Migration von STEP 7 V5 nach STEP 7 Basic / Professional (TIA Portal)

Modulfunktion für STEP 7 V5-Projekt-Erweiterung			S7-1200/1500-Modul in STEP 7 Basic / Professional	
TIM (Funktion) für Erweiterung	Zu verwendender Proxy im Katalog		Kompatible Baugruppe	Benötigte STEP 7-Version
TIM 3V-IE Advanced	PROXY CP1243-8 IRC	⇒	CP 1243-8 IRC	STEP 7 Basic
TIM 3V-IE Advanced	PROXY CP1243-8 IRC	⇒	CP 1542SP-1 IRC	STEP 7 Professional
TIM 4R-IE	PROXY TIM 1531 IRC	⇒	TIM 1531 IRC	STEP 7 Professional
TIM 4R-IE Stand-alone	PROXY TIM 1531 IRC	⇒	TIM 1531 IRC	STEP 7 Professional

Hinweis

TIM 4R-IE Stand-alone für S7-400 wird zu TIM 1531 IRC

Eine in STEP 7 V5 benötigte TIM 4R-IE Stand-alone, welche einer CPU-400 zugeordnet ist, muss bei der Erweiterung klassischer SINAUT-Projekte in STEP 7 Professional durch eine TIM 1531 IRC ersetzt werden.

Nur in neuen Projekten, die ausschließlich in STEP 7 Professional projektiert werden, kann eine TIM 4R-IE Stand-alone angelegt werden.

Hinweis

Keine serielle Schnittstelle für CP 1542SP-1 IRC

Für einen CP 1542SP-1 IRC darf in STEP 7 V5 keine serielle Schnittstelle projektiert werden, da der CP keine serielle Schnittstelle besitzt.

Voraussetzungen und Modul-Aktualisierung

Voraussetzungen für den Import sind:

- **STEP 7 V5-Projekt**
  Eine Konfigurationsdatei aus einem konsistenten STEP 7 V5-Projekt
- **Firmware-Version**
  Eine der folgenden Firmware-Versionen ist für das Kommunikationsmodul erforderlich:
  - TIM 1531 IC: V2.0
  - TIM 1531 IC: V1.1
    Eine TIM V1.1 kann durch eine TIM V2.0 ausgetauscht werden, nicht umgekehrt.
    In STEP 7 wird das Verfahren der "Telecontrol-Konfiguration" auf "Projektieren" gesetzt.
    Vorhandene Projektierungsdaten werden übernommen.
  - CP 1542SP-1 IRC: V2.0
    Ein CP V1.0 kann durch einen CP V2.0 ausgetauscht werden, nicht umgekehrt.
    In STEP 7 wird das Verfahren der "Telecontrol-Konfiguration" auf "Projektieren" gesetzt.
    Vorhandene Projektierungsdaten werden übernommen.
  - CP 1243-8 IRC: V3.1
    Übernahme der Daten aus STEP 7 V5 mit Änderungsmöglichkeiten (siehe unten, Abschnitt "Importierte Projektierungsdaten")
    Ein CP V3.0 kann durch einen CP V3.1 ausgetauscht werden, nicht umgekehrt.
  - CP 1243-8 IRC: V2.1
    Übernahme der Daten aus STEP 7 V5 ohne Änderungsmöglichkeit
    Ein CP V2.1 kann durch einen CP V3.1 ausgetauscht werden, nicht umgekehrt. Der eingesetzte CP muss den Hardware-Erzeugnisstand 2 haben.
    In STEP 7 wird das Verfahren der "Telecontrol-Konfiguration" auf "Importieren" gesetzt.
    Vorhandene Projektierungsdaten werden übernommen.

    Hinweis

    Deaktivierung der Partner-Projektierungsdaten bei Migration von STEP 7 V13 SP1

    Wenn Sie einen CP 1243-8 IRC V2.1 von STEP 7 V13 SP1 nach STEP 7 V15 migrieren, dann werden die Partner-Projektierungsdaten deaktiviert. Sie erhalten eine Meldung beim Übersetzen des Projekts.

    Importieren Sie in diesem Fall die Partner-Projektierungsdaten erneut in das Modul.

  - TIM 3V... / TIM 4R...: V2.5.5

- **ST7-Verbindungen**
  Für ST7-Verbindungen zwischen einer TIM 1531 IRC und einer CPU:
  - Die Firmware der CPU muss mindestens die Version V2.5 haben.
  - Die TIM und die CPU müssen vor dem Import in STEP 7 Professional vernetzt sein.
  - Bei Projektierung von TD7onCPU darf die CPU der TIM in STEP 7 Professional nicht zugeordnet sein.

  Für ST7-Verbindungen zwischen einer TIM 1531 IRC und einer PC-Applikation:
  - Die TIM und die PC-Station müssen vor dem Import in STEP 7 Professional vernetzt sein.
- **CPU 1500 / PC-Station**
  Für eine CPU 1500 oder eine PC-Station mit SINAUT-Applikation als ST7-Kommunikationspartner muss die IP-Adresse im STEP 7 V15-Projekt der in STEP 7 V5 projektierten Adresse entsprechen.

Projektierung in STEP 7 V5

Die Beschreibung der Projektierung finden Sie in den Handbüchern der betreffenden Module.

DNP3

Adressierung

Für die Projektierung und die Inbetriebnahme des Kommunikationsmoduls sind folgende Informationen erforderlich:

Adressinformationen des Kommunikationsmoduls

Für die Adressierung müssen folgende Parameter für das Modul projektiert werden:

- Stationsadresse (DNP-Adresse auf der Sicherungsschicht)
- Adresse, je nach Netztyp:
  - IP-Adresse und Subnetzmaske; alternativ: IP-Adresse eines DHCP-Servers
  - Telefonnummer (für Wählnetz)
  - WAN-Adresse (für Standleitung)
- Nummer des eigenen Listener-Ports der Station (für Verbindungsanfragen vom Master)

Adressinformationen des Masters

Folgende Informationen des Masters werden für die Projektierung des Moduls benötigt:

- Stationsadresse (DNP-Adresse des Masters auf der Sicherungsschicht)
- Adresse des Masters, je nach Netztyp:
  - IP-Adresse / Subnetzmaske + Port-Nummer des Listener-Ports des Masters
    oder
    Durch DNS auflösbarer Name
    (Sie benötigen die DNS-Server-Adresse; der DNS-Server muss durch das Modul erreichbar sein.)
  - Telefonnummer (für Wählnetz)
  - WAN-Adresse (für Standleitung)

Redundante DNP3-Master

Beide Geräte eines redundanten Masters haben eine identische Stationsadresse.

Sie benötigen nur unterschiedliche IP-Adressen bzw. Host-Namen.

Konfigurationen mit Verbindungen über das Internet: VPN-Verbindungen

Bei Verbindungen, die über das Internet laufen, können dynamische IP-Adressen verwendet werden.

Um Kommunikation in beide Richtungen zu ermöglichen und damit die Daten geschützt übertragen werden, wird eine Verbindung mit VPN-Tunnel benötigt. Hierfür bieten sich Security-Baugruppen der Reihen SCALANCE S oder SCALANCE M an.

Berücksichtigen Sie bei der Projektierung folgende Punkte:

- Die Master-IP-Adresse projektieren Sie wie üblich.
- Bei der Projektierung der Schnittstelle des Moduls projektieren Sie die IP-Adresse des Routers.
- Die VPN-Projektierung mit SCALANCE S/M nehmen Sie sowohl für die Stationsseite als auch für die Seite der Leitstelle in STEP 7 vor.

Verbindungsaufbau

Verbindungsaufbau bei DNP3

Beim DNP3-Protokoll baut der Master die Verbindung auf (Aufrufbetrieb / Polling).

Wenn eine aufgebaute Verbindung unterbrochen wird, dann versucht ein Master-Modul, die Verbindung wieder aufzubauen.

Hinweis

Verbindungsunterbrechung durch Mobilfunknetzwerkbetreiber

Beachten Sie bei der Nutzung von Mobilfunkdiensten, dass bestehende Verbindungen von Mobilfunk-Netzwerkbetreibern zu Wartungszwecken unterbrochen werden können.

Spontanbetrieb

Durch Aktivierung des Parameters "Spontanes Melden" der Parametergruppe "Spontane Antworten" in den "Einstellungen DNP3-Station" einer Schnittstelle wird die unaufgeforderte Übertragung von Daten einer Station freigegeben. In diesem Fall können Wählnetz-Verbindungen durch das Kommunikationsmodul als Client aufgebaut werden, um Daten von Ereignisses spontan zu senden.

Direkte Kommunikation zwischen Stationen

Zwischen zwei Stationen ist direkte Kommunikation möglich, wenn bei den Stationen der Spontanbetrieb aktiviert ist.

Verbindungsaufbau bei Open User Communication und PG/OP-Kommunikation

Bei der Open User Communication ist in einer S7-Station die CPU der Verbindungspartner.

Verbindungen werden aufgebaut, sobald die entsprechenden Programmbausteine in der CPU aufgerufen werden.

Dies gilt auch für den Fall, dass eine andere S7-Station Daten sendet. In diesem Fall werden von der Empfängerstation die entsprechenden Empfangsbausteine aufgerufen.

Quittierung beim DNP3-Protokoll

Quittierung bei DNP3

Die grundlegenden Quittiermechanismen werden für die Sicherungsschicht projektiert, siehe Kapitel Übertragungseinstellungen - DNP3 (Seite 1906).

Ein Stationsmodul quittiert eine Anfrage des Masters durch seine Antwort-Telegramme.

Die Quittierung von spontanen Telegrammen, die ein Stationsmodul an den Master sendet, wird in der Parametergruppe "Einstellungen DNP3-Station" projektiert, siehe Kapitel Einstellungen DNP3-Station (Seite 1909).

Übertragungseinstellungen - DNP3

Übertragungseinstellungen - DNP3

- **Max. Zeit zwischen Select und Operate**
  Max. Zeitraum (Sekunden) zwischen Befehls-Vorauswahl (Select) und Befehls-Eingabe (Operate)
  - Zentrale-Station:
    Wenn in der Zentrale nach einer Befehls-Vorauswahl innnerhalb des projektierten Zeitraums keine Befehlseingabe folgt, muss der Befehl erneut angewählt werden. Die doppelte Eingabepflicht vermindert das Risiko einer versehentlichen Bedienung in der Zentrale.
  - Station:
    Damit ein empfangener Befehl an die CPU übertragen und damit wirksam wird, darf die Station zwischen Select und Operate kein weiteres Befehls-Telegramm empfangen.

  Wertebereich: 1..65535
  Voreinstellung: 1
  Wenn ein Master ein Objekt mit dem Funktions-Code 7 "DIRECT_OPERATE" an die Station sendet, dann wertet die Station den hier projektierten Zeitraum nicht aus.
- **Übertragungsform**
  Legt die Übertragungsform von Ereignissen fest.
  - Typ-spezifisch
    Dies ist die DNP3-typisch gebündelte Übertragung von Ereignissen nach Datentypen. Zuerst werden die vorhandenen binären Ereignisse, dann alle Analogwerte und zuletzt alle Zählwertänderungen übertragen. Hierdurch werden die Telegramme etwas kompakter und die Übertragung effizienter.
  - Chronologisch
    In diesem Modus werden Ereignisse streng chronologisch übertragen. Die in der DNP3-Spezifikation vorgesehene optimierende Zusammenfassung zu Blöcken gleichen Datentyps (siehe oben) findet nicht statt. Dieser Modus dient in erster Linie Leitsystemen, welche Ereignisse streng chronologisch archivieren.

  Vorbelegung: Typ-spezifisch

- **Verbindungsüberwachungszeit (Applikationsschicht)**
  (Application layer confirmation timeout)
  Zeitraum (Sekunden), in dem das Modul ein Lebenszeichen auf der Applikationsschicht von seinem Partner erwartet.
  - Master (Zentrale-Station):
    Zeitraum, in dem der Master ein Lebenszeichen von der Station erwartet.
    Bei Überschreitung der projektierten Zeit sendet der Master ein Telegramm an die Station. Wenn dieses Telegramm nicht von der Station quittiert wird, stuft der Master die Station als nicht erreichbar ein, baut die Verbindung ab und anschließend wieder auf.
    Befehle:
    Wenn eine Verbindung mit einer Station unterbrochen oder gestört ist, werden Befehls-Telegramme maximal dreimal wiederholt, bevor sie vom Master verworfen werden.
  - Station:
    In der Station wird der Parameter für die Berechnung der Wartezeiten beim Verbindungsaufbau verwendet.

  Bei redundanten Wegen gilt: Bei Ausbleiben der Quittung wird der Übertragungsweg als gestört eingestuft.
  Wertebereich: 0 ... 65535 s
  Vorbelegung: 5
  Bei 0 (Null) ist die Funktion abgeschaltet.

- **Partnerüberwachungszeit**
  Wenn das Stations-Modul innerhalb der projektierten Zeit kein Lebenszeichen vom Master auf Applikationsebene empfängt, stuft es die Verbindung als gestört ein und baut die Verbindung ab.
  Das Master-Modul erwartet nach dem Senden von Daten innerhalb der projektierten Zeit eine Antwort von der Station.
  Wertebereich: 0 ... 65535
  Vorbelegung: 0
  Bei 0 (Null) ist die Funktion abgeschaltet.

Sicherungsschicht

- **Telegrammwiederholung**
  Maximale Anzahl an Wiederholungsversuchen, ein Telegramm zu versenden, wenn auf der Sicherungsschicht keine Quittung vom Partner empfangen wird.
  - Bei seriellen Verbindungen können Wiederholungen sinnvoll sein, wenn die Verbindungsqualität nicht gut ist.
  - Bei Ethernet-Verbindungen wird empfohlen, den Wert 0 zu projektieren, da erforderliche Wiederholungen durch die Protokoll-Implementierung durchgeführt werden.

  Wertebereich: 0 ... 255
  Voreinstellung: 0
  Bei 0 (Null) ist die Funktion abgeschaltet.

- **Quittierung**
  Bedingung, unter welcher das Modul eine Empfangsbestätigung (Quittung) vom Partner anfordert (Versenden des Funktions-Codes "CONFIRMED_USER_DATA"):
  - Nie
    Es werden keine Quittungen angefordert.
  - Immer
    Es werden immer Quittungen angefordert.
  - Bei Segmentierung
    Quittungen werden nur angefordert bei Daten-Telegrammen, die aufgrund ihrer Länge in mehrere Segmente geteilt werden.

  Bei Ethernet-Verbindungen empfiehlt die DNP3-Spezifikation, keine Quittungsanforderungen auf der Sicherungsschicht zu verwenden.
  Bei störungsanfälligen Verbindungen, z. B. Funkstrecken, kann die Projektierung von Quittungsanforderungen auf der Sicherungsschicht sinnvoll sein.

- **Quittungsüberwachungszeit**
  Zeitraum (Sekunden), in dem vom Partner eine Bestätigung (Quittung) auf der Sicherungsschicht erwartet wird. Wenn im projektierten Zeitraum keine Quittung empfangen wird, wird die Übertragung der Daten wiederholt, siehe Parameter "Telegrammwiederholung" oben.
  Wertebereich: 0...65535
  Voreinstellung: 2
  Der vorbelegte Wert von 2 Sekunden muss in der Regel nur bei langsamen seriellen Verbindungen erhöht werden.
  Bei 0 (Null) ist die Funktion abgeschaltet.

Einstellungen DNP3-Master

Master-spezifische DNP3-Parameter

Folgende Parameter finden Sie in der Parametergruppe "Einstellungen DNP3-Master" der auf den Netzknotentyp "Zentrale-Station" eingestellten Schnittstellen des Kommunikationsmoduls.

- **Polling-Basisintervall**
  Hier legen Sie das Basisintervall für Stationsaufrufe durch die Zentrale fest.
  Wertebereich: 0 ... 65535 Sekunden
  Vorbelegung: 30
  Bei 0 (Null) ist die Funktion abgeschaltet. Es findet kein zyklisches Polling statt, auch nicht für die nachfolgend aufgeführten Parameter, deren Berechnung auf dem Polling-Basisintervall beruht.
  Basis für weitere Parameter
  Das Polling-Basisintervall wird für die Berechnung der folgenden Parameter im Verbindungseditor (Editor "Netzwerkdaten > Telecontrol > DNP3") verwendet:
  - Ereignis-Polling-Intervall
  - Klasse-0-Polling-Intervall
  - Zeitdauer des Polling-Zyklus, siehe Parameter "Polling-Modus".
- **Max. Anzahl Ereignisse pro Abfrage**
  Maximale Anzahl an Ereignissen, die nach einem Aufruf durch die Zentrale im Antworttelegramm der Station gesendet werden dürfen.
  Wertebereich: 0 ... 65535
  Vorbelegung: 0
  Bei 0 (Null) ist die Funktion abgeschaltet (keine Begrenzung).

Einstellungen DNP3-Station

DNP3-Parameter der Station / Knotenstation

Folgende Parameter finden Sie in der Parametergruppe "Einstellungen DNP3-Station" der auf den Netzknotentyp "Station" bzw. "Knotenstation" eingestellten Schnittstellen des Kommunikationsmoduls.

Überwachungsparameter

- Störungs-Bit
  Die Station kann Bit 1.6 des Internal Indication Byte (IIN1.6) im Application Response Header verwenden, um dem Master eine Störung der CPU anzuzeigen.
  Bei aktivierter Option wird das Bit auf "1" gesetzt, wenn sich die CPU im Zustand STOP befindet.
- IP-Adress-Prüfung
  Der Parameter dient der Sicherheit beim Aufbau einer Verbindung zwischen Station und Master.
  - IP-Adresse prüfen
    Bei aktivierter Option prüft die Station bei einer Verbindungsanfrage zusätzlich zur Stationsadresse des Partners dessen IP-Adresse.
    Wenn die IP-Adresse des Partners nicht mit der Adresse eines projektierten Partners der Station übereinstimmt, lehnt die Station die Verbindung ab.
  - IP-Adresse nicht prüfen
    Die Station prüft keine IP-Adressen bei eingehenden TCP-Verbindungsanfragen.

Ereignis-Eigenschaften

- **Ereignisklasse für Abbildspeicher**
  Auswahl einer Ereignisklasse, bei denen nur die jeweils letzten, aktuellen Werte im Sendepuffer gespeichert werden. Ältere Ereignisse werden im Sendepuffer überschrieben (Abbildspeicherverfahren).
  Der Parameter steht bei der TIM 3V.. / TIM 4R.. nicht zur Verfügung. Diese Module arbeiten nach der Einstellung "Standard".
  - Klasse-x-Ereignisse
    Von Ereignissen der selektierten Klasse wird nur der jeweils aktuelle Wert im Sendepuffer gespeichert (Abbildspeicherverfahren).
  - Standard
    Alle Werte von Ereignissen der Klassen 1 und 2 werden im Sendepuffer gespeichert. Nur von Ereignissen der Klasse 3 wird der jeweils aktuelle Wert im Sendepuffer gespeichert (Abbildspeicherverfahren).

  Details zur Funktion des Abbildspeichers und Sendepuffers sowie zu den Übertragungsmöglichkeiten finden Sie im Kapitel Prozessabbild, Übertragungsart, Ereignisklassen (Seite 1863).
- **Puffer für Ereignisse Klasse 1 / 2 / 3**
  Hier legen Sie für jede der drei Ereignisklassen fest, ab welcher Ereignisanzahl die gespeicherten Ereignisse an den Kommunikationspartner gesendet werden.
  Beachten Sie, dass sich die maximale Größe des Sendepuffers auf alle angeschlossenen DNP3-Zentralen aufteilt.
  Zulässiger Bereich: 1 ... 255
- **Verzögerungszeit Ereignisse Klasse 1 / 2 / 3**
  Hier legen Sie für jede der drei Ereignisklassen in Sekunden fest, wie lange die Ereignisse maximal im Sendepuffer gespeichert werden dürfen, bevor sie an den Kommunikationspartner gesendet werden.
  Voraussetzung ist, dass spontane Übertragung projektiert ist (siehe unten).
  Zulässiger Bereich: 0 ... 65535
  Bei 0 (Null) ist die Funktion abgeschaltet.

Knotenstation

Dieser Parameter steht nur bei Schnittstellen zur Verfügung, bei denen der Netzknotentyp auf "Knotenstation" eingestellt ist.

- **Antwort ohne aktuelles Stations-Abbild**
  Bei einer Schnittstelle mit dem Netzknotentyp "Knotenstation" führt das Kommunikationsmodul der Knotenstation für jede unterlagerte DNP3-Station das Stations-Prozessabbild im Abbildspeicher.
  - Option aktiviert
    Das Modul antwortet auf einen Klasse-0-Aufruf auch dann, wenn das Prozessabbild im Abbildspeicher der Knotenstation nicht aktuell ist. Bei einem Aufruf durch die Zentrale werden die Daten aller angeschlossenen Stationen an die Zentrale gesendet, auch wenn die Daten einer oder mehrerer Stationen nicht aktuell sind.
  - Option deaktiviert
    Das Modul antwortet auf einen Klasse-0-Aufruf nur, wenn das Prozessabbild im Abbildspeicher der Knotenstation aktuell ist.

Spontanes Senden

- **Spontanes Melden**
  Übertragungsmodus für Ereignisse
  - Ja
    Ereignis-Telegramme werden sofort übertragen.
    Bei dieser Option baut die Station selbständig eine Verbindung mit dem Master auf. Wenn das spontane Senden in der Zentrale deaktiviert ist, dann sendet die Zentrale den Funktions-Code "DISABLE_UNSOLICITED" (Versenden von spontanen Telegrammen durch die Station nicht erlaubt). In diesem Fall speichert die Station die Ereignisse, ohne sie sofort zu versenden.
  - Nein
    Ereignis-Telegramme werden erst übertragen, wenn Daten vom Master abgefragt werden.

  Für eine konsistente Projektierung sollten Sie bei der Station und der Zentrale die gleiche Einstellung wählen.

  Hinweis

  Einstellung bei Standleitungen

  Bei Anschluss einer seriellen Schnittstelle an eine DNP3-Standleitung mit der Einstellung "halbduplex" müssen Sie den Parameter auf "Nein" setzen.

  Zur Vermeidung von Kollisionen sollten Sie diese Einstellung auch bei Anschluss an Vollduplex-Multidrop-Standleitungen verwenden.

- **Max. Anzahl Wiederholungen**
  Maximale Anzahl der Wiederholung von spontanen Telegrammen, wenn vom Master keine Quittung empfangen wird. Nach Erreichen der projektierten Anzahl baut die Station die Verbindung ab.
  Wertebereich: 0...255
  Voreinstellung: 3
- **Überwachungszeit für spontane Telegramme (Unsolicited Confirmation Timeout period)**
  Zeitraum (Sekunden), in dem die Station vom Master eine Quittierung spontaner Telegramme erwartet. Wenn im projektierten Zeitraum keine Quittung empfangen wird, wird die Übertragung der Daten wiederholt.
  Erst wenn die Station die Quittung vom Master empfängt, werden die gesendeten Telegramme aus dem Sendepuffer gelöscht.
  Wertebereich: 1...65535
  Voreinstellung: 5

DNP3-Security-Optionen

Authentifizierung und Schlüsselaustausch

Bei aktivierter Funktion (Secure Authentication) authentifizieren sich DNP3-Master und Station durch einen geheimen Schlüssel, dem Pre-shared Key.

Mithilfe des gemeinsamen Pre-shared Key werden nach dem ersten Verbindungsaufbau zwischen Master und Station Sitzungsschlüssel vereinbart, die nachfolgend zyklisch erneuert

werden. Die Initiative für die Erneuerung der Sitzungsschlüssel geht in der Regel vom Master aus.

Die Kriterien für die Schlüsselerneuerung werden in folgenden Parametern festgelegt.

- Schlüsselaustauschintervall
- Authentifizierungsanfragen vor Schlüsselaustausch

Sobald eine dieser beiden Bedingungen erfüllt ist, wird der Sitzungsschlüssel erneuert.

Parameter

- **DNP3-Security-Optionen aktivieren**
  Aktivieren Sie die Option, wenn Sie Secure Authentication nutzen möchten.
- **IKE-Modus**
  Auswahl des Modus zum Schlüsselaustausch. Wertebereich:
  - Aggressive Mode
    Der Aggressive Mode ist etwas schneller, überträgt aber die Identität unverschlüsselt.
  - Main Mode
    Der Main Mode ist der Standard-Modus.

  Voreinstellung: Aggressive Mode
- **Security-Statistik**
  Gibt an, ob die Statistik von Security-Ereignissen an den Master gesendet werden. Security-Ereignisse sind Authentifizierungsanfragen an das Modul. Bei Aktivierung der Option werden alle Authentifizierungsanfragen mit Datum, Zeit und Ergebnis im Modul gespeichert und zur weiteren Auswertung an den Master gesendet.
  Wertebereich:
  - Keine Security-Statistik senden
  - Security-Statistik senden

  Voreinstellung: Keine Security-Statistik senden
- **SHA-1-Verriegelung**
  Einstellung, ob der CP den Secure Hash-Algorithmus SHA-1 verwenden darf, wenn als Secure Hash-Algorithmus "SHA-256" projektiert wurde und der Master kein SHA-256 unterstützt.
  Wertebereich:
  - SHA-1-Modus nicht erlaubt
    Das Modul darf kein SHA-1 verwenden. Wenn der Master kein SHA-256 unterstützt, kommt in diesem Fall keine Verbindung zustande.
  - SHA-1-Modus erlaubt
    Das Modul darf SHA-1 verwenden, wenn der Master kein SHA-256 unterstützt.

  Voreinstellung: SHA-1-Modus nicht erlaubt

- **Secure Hash-Algorithmus (SHA)**
  Auswahl des Secure Hash Algorithm (SHA)
  Wertebereich:
  - SHA-1
  - SHA-256

  Voreinstellung: 256
- **Key-Wrap-Algorithmus**
  Auswahl des Advanced Encryption Standard (AES)
  Wertebereich:
  - AES-128
  - AES-256

  Voreinstellung: AES-128
- **Schlüssellänge**
  Angabe der Länge des Pre-shared Key in Byte.
  In Abhängigkeit des Key-Wrap-Algorithmus werden folgende Längen verwendet:
  - Für AES-128: 16 Byte
  - Für AES-256: 32 Byte
- **Max. Anzahl an Statistik-Anfragen**
  Wenn die hier projektierte Anzahl an Statistik-Anfragen eines Masters innerhalb des Schlüsselaustauschintervalls überschritten wird, trägt das Modul eine Meldung in den Diagnosepuffer der CPU ein.
  Wertebereich: 2...255. Voreinstellung: 5
- **Authentifizierungsanfragen vor Schlüsselaustausch**
  Maximale Anzahl an Authentifizierungsanfragen der Station beim Master. Wenn diese Anzahl erreicht ist, wird der Sitzungsschlüssel erneuert.
  Wertebereich: 1...10000. Voreinstellung: 1000
  Empfehlung: Stellen Sie die Anzahl bei der Station doppelt so hoch ein wie beim Master.
- **Schlüsselaustauschintervall**
  Zeitraum, nach dem der Schlüssel zwischen Station und Master neu ausgetauscht wird. Das Intervall muss bei beiden Kommunikationspartnern aufeinander abgestimmt sein.
  Wertebereich: 0...65535 min. Bei 0 (Null) wird der Schlüssel nie getauscht (Funktion abgeschaltet). Voreinstellung: 15 min.
  Empfehlung: Stellen Sie das Schlüsselaustauschintervall bei der Station doppelt so hoch ein wie beim Master.

- **Authentifizierungs-Überwachungszeit**
  Maximale Wartezeit für die Antwort vom Master auf eine Authentifizierungsanfrage der Station.
  Die Überschreitung der Wartezeit wird von der Station als Fehler gewertet. Die Station generiert in diesem Fall ein Security-Ereignis und sendet dieses an den Master.
  Wertebereich: 1... 65535 s. Voreinstellung: 5
- **Pre-shared Key**
  Der Pre-shared Key kann auf zwei Wegen projektiert werden:
  - Manuelle Projektierung
    Geben Sie den Pre-shared Key in STEP 7 manuell als Hexadezimalwert ein.
  - Import als Datei
    Importieren Sie den Pre-shared Key aus dem Dateisystem der Engineering-Station, wenn der Pre-shared Key vom Master oder einem anderen System erzeugt worden ist.

  Der Pre-shared Key eines Stationsmoduls muss identisch sein mit dem Pre-shared Key des Masters.

IEC 60870-5

Adressierung

Für die Projektierung und die Inbetriebnahme des Kommunikationsmoduls sind folgende Informationen erforderlich:

Adressinformationen des Kommunikationsmoduls

Für die Adressierung müssen folgende Parameter für das Modul projektiert werden:

- ASDU-Adresse (Adresse des Informationsobjekts auf der Sicherungsschicht)
- Adresse, je nach Netztyp:
  - IP-Adresse und Subnetzmaske; alternativ: IP-Adresse eines DHCP-Servers
  - Telefonnummer (für Wählnetz)
  - WAN-Adresse (für Standleitung)
- Nummer des eigenen Listener-Ports der Station (für Verbindungsanfragen vom Master)

Adressinformationen des Masters

Folgende Informationen des Masters werden für die Projektierung des Moduls benötigt:

- ASDU-Adresse (Adresse des Informationsobjekts des Masters auf der Sicherungsschicht)
- Adresse des Masters, je nach Netztyp:
  - IP-Adresse / Subnetzmaske + Port-Nummer des Listener-Ports des Masters
    oder
    Durch DNS auflösbarer Name
    (Sie benötigen die DNS-Server-Adresse; der DNS-Server muss durch das Modul erreichbar sein.)
  - Telefonnummer (für Wählnetz)
  - WAN-Adresse (für Standleitung)

Redundante IEC-Master

Beide Geräte eines redundanten Masters haben eine identische ASDU-Adresse.

Sie benötigen nur unterschiedliche IP-Adressen bzw. Host-Namen.

Konfigurationen mit Verbindungen über das Internet: VPN-Verbindungen

Bei Verbindungen, die über das Internet laufen, können dynamische IP-Adressen verwendet werden.

Um Kommunikation in beide Richtungen zu ermöglichen und damit die Daten geschützt übertragen werden, wird eine Verbindung mit VPN-Tunnel benötigt. Hierfür bieten sich Security-Baugruppen der Reihen SCALANCE S oder SCALANCE M an.

Berücksichtigen Sie bei der Projektierung folgende Punkte:

- Die Master-IP-Adresse projektieren Sie wie üblich.
- Bei der Projektierung der Schnittstelle des Moduls projektieren Sie die IP-Adresse des Routers.
- Die VPN-Projektierung mit SCALANCE S/M nehmen Sie sowohl für die Stationsseite als auch für die Seite der Leitstelle in STEP 7 vor.

Quittierung beim IEC-Protokoll

Quittier-Mechanismen beim Protokoll IEC 60870-5-104

Projektierung: Schnittstelle des Moduls > "Erweiterte Optionen" > Übertragungseinstellungen - IEC 60870-5"

Das Modul schickt mit jedem gesendeten Daten-Telegramm eine fortlaufende Sendefolgenummer mit. Das Daten-Telegramm bleibt zunächst im Sendepuffer des Moduls gespeichert.

Beim Empfang schickt der Master die Sendefolgenummer aus diesem oder (bei Empfang mehrerer Daten-Telegramme) dem letzten Daten-Telegramm als Quittung an das Modul

zurück. Das Modul speichert die vom Master zurückgeschickte Sendefolgenummer als Empfangsfolgenummer und verwendet sie als Quittung.

Daten-Telegramme, deren Sendefolgenummer gleich oder kleiner der aktuellen Empfangsfolgenummer ist, werden als erfolgreich übertragen gewertet und aus dem Sendepuffer des Moduls gelöscht.

Parameter:

- **k: Differenz Sendefolgenummer N(S) zu Empfangsfolgenummer N(R)**
  Maximale Anzahl an unquittierten Daten-Telegrammen (I-APDUs) als maximale Differenz zwischen Sendefolgenummer N(S) und Empfangsfolgenummer N(R).
  Wenn k erreicht ist und t_1 noch nicht abgelaufen ist, sendet das Modul solange keine Daten-Telegramme, bis alle gesendeten Daten-Telegramme vom Master quittiert sind.
  Wenn k erreicht ist und t_1 abgelaufen ist, wird die TCP-Verbindung abgebaut.
- **w: Max. Anzahl unquittierter Datentelegramme**
  Maximale Anzahl an empfangenen Daten-Telegrammen (I-APDUs), nach der das älteste vom Master empfangene Daten-Telegramm quittiert werden muss.

Empfehlungen der Spezifikation:

- w sollte nicht größer sein als 2/3 k.
- Empfohlener Wert für k: 12
- Empfohlener Wert für w: 8

IEC-spezifische Parameter

Die folgenden Parameter finden Sie in den Parametergruppen zu den Master- bzw.- Stations-Einstellungen.

Einige Parameter werden auch im Verbindungseditor der Telecontrol-Verbindungen angezeigt und sind dort teilweise editierbar.

IEC-Master

Folgende Parameter finden Sie in der Parametergruppe "Einstellungen IEC-Master" der auf den Netztyp IEC und den Netzknotentyp "Zentrale-Station" eingestellten Schnittstellen des Kommunikationsmoduls.

- **Polling-Basisintervall**
  Hier legen Sie das Basisintervall für Stationsaufrufe durch die Zentrale fest.
  Wertebereich: 0 ... 65535 Sekunden
  Vorbelegung: 30
  Bei 0 (Null) ist die Funktion abgeschaltet. Es findet kein zyklisches Polling statt, auch nicht für die nachfolgend aufgeführten Parameter, deren Berechnung auf dem Polling-Basisintervall beruht.
  Das Basisintervall wird für die Berechnung der folgenden Parameter in der Verbindungsprojektierung verwendet:
  - Intervall für Generalabfrage
  - Intervall für Zähler-Generalabfrage
  - Intervall für Gruppenabfrage
  - Intervall für Zähler-Gruppenabfrage
- **Max. Anzahl Ereignisse pro Aufruf**
  Maximale Anzahl an Ereignissen, die nach einem Aufruf durch die Zentrale im Antworttelegramm der Station gesendet werden dürfen.
  Wertebereich: 0 ... 65535
  Vorbelegung: 0
  Bei 0 (Null) ist die Funktion abgeschaltet (keine Begrenzung).

IEC-Station

Folgende Parameter finden Sie in der Parametergruppe "Einstellungen IEC-Station" der betreffenden Schnittstellen des Kommunikationsmoduls.

Stations-Parameter

- Übertragungsmodus
  Der Parameter legt fest, ob ASDUs (Ereignisse) dieser Station spontan an die Zentrale gesendet werden dürfen.
  Wertebereich:
  - Spontane Übertragung
    Das Modul darf spontane ASDUs senden (cause of transmission <3>).
  - Keine spontane Übertragung
    Das Modul sendet keine spontanen ASDUs.

  Vorbelegung: Spontan

  Hinweis

  Kollisionen bei Vollduplex-Standleitungen

  Bei Anschluss einer seriellen Schnittstelle an eine Standleitung mit der Verbindungsart "halbduplex" müssen Sie Spontane Übertragung deaktivieren.

  Zur Vermeidung von Kollisionen sollten Sie Spontane Übertragung auch bei Anschluss an Vollduplex-Multidrop-Standleitungen deaktivieren.

- Polling-Modus
  Hier legen Sie den Modus fest, nach dem die Zentrale die Station aufruft.
  Der bei der Station projektierte Wert wird an die Zentrale übermittelt und dort gespeichert.
  Wertebereiche:
  - Zyklisch
    Die Station wird zyklisch aufgerufen.
    Die Zykluszeit wird vom Parameter "Klasse-0-Polling-Intervall" des Masters bestimmt.
  - Nach Anlauf / Verbindungsaufbau
    Die Station wird nur nach dem ersten Anlaufen und nach einem Neustart aufgerufen.
- Partnerüberwachungszeit
  Wenn das Kommunikationsmodul der Station innerhalb der projektierten Zeit kein Lebenszeichen vom Master empfängt, stuft es die Verbindung zum Master als gestört ein.
  Das Modul baut die Verbindung ab und versucht, die Verbindung neu aufzubauen.
  Wertebereich: 0 ... 65535
  Vorbelegung: 0
  Bei 0 (Null) ist die Funktion abgeschaltet.

Weitere Parameter

- Allgemeine IEC-spezifische Parameter
  Siehe "Übertragungseinstellungen - IEC 60870-5" der jeweiligen Schnittstelle des Moduls.
- Antwort auf Generalabfrage / Zuordnung zu Gruppenabfrage
  In der Datenpunktprojektierung, siehe Kapitel DNP3- und IEC-Datenpunkte (Seite 1868).
- Aufruf-Intervalle
  In der Verbindungsprojektierung, siehe Kapitel Optionen (IEC) (Seite 1847).

Zusatzinformationen zu Mobilfunk-CPs

CP 1242-7 (6GK7 242-7KX30-0XE0)

Betriebsarten des CP 1242-7

Betriebsarten des CP

Der CP 1242-7 (6GK7 242-7KX30-0XE0) ermöglicht einer S7-1200, als GPRS-Station mit einer Zentrale oder anderen entfernten Stationen über das GSM-Netz zu kommunizieren. Für die Kommunikation über GPRS wird der CP auf eine der folgenden Betriebsarten eingestellt:

- Telecontrol
  Diese Betriebsart des CP ermöglicht der GPRS-Station den Datenaustausch mit folgenden Partnern:
  - Kommunikation mit dem Telecontrol-Server
    Diese Betriebsart des CP ermöglicht der GPRS-Station den Datenaustausch mit einem Telecontrol-Server.
    Der Telecontrol-Server ist ein am Internet angeschlossener PC mit der Applikation "TCSB". Er befindet sich in der Regel in der Zentrale und dient der Überwachung und Steuerung der entfernten GPRS-Stationen. Über die integrierte OPC-Schnittstelle können Daten mit dem OPC-Client eines zentralen Leitsystems ausgetauscht werden. Der Telecontrol-Server-PC wird nicht in STEP 7 projektiert. Die Applikation "TCSB" hat eine eigene Projektierungsoberfläche.
  - Kommunikation mit einer anderen entfernten GPRS-Station
    Die Daten werden über den Telecontrol-Server geleitet.
  - Kommunikation mit einer Engineering-Station (für TeleService)

  Die Kommunikation mit dem Telecontrol-Server erfolgt über das GSM-Netz und das Internet.
  Voraussetzungen für diese Betriebsart sind eine SIM-Karte mit freigeschaltetem GPRS-Dienst und ein für den CP erreichbarer Telecontrol-Server.
- GPRS direkt
  Diese Betriebsart des CP dient der direkten Kommunikation zwischen entfernten Stationen über das GSM-Netz. Es wird kein Telecontrol-Server benötigt.
  Damit Netzteilnehmer in öffentlichen Funknetzen direkt erreichbar sind, müssen diese über eine feste Adresse angesprochen werden können. Hierzu werden SIM-Karten mit fester IP-Adresse eingesetzt, die es ermöglichen, die Stationen direkt zu adressieren.
  Die möglichen Kommunikationsdienste und Sicherheitsfunktionen (z. B. VPN) hängen dabei vom Angebot des Netzwerkbetreibers ab.
  Mögliche Kommunikationspartner der Station mit CP 1242-7 in der Betriebsart "GPRS direkt" sind:
  - Ein für den CP über eine IP-Adresse erreichbarer Teilnehmer (S7-Station mit CP 1242-7)
  - Eine Engineering-Station (für TeleService)

Siehe auch

Verbindungsaufbau beim CP 1242-7 (Seite 1921)

Verbindungsaufbau beim CP 1242-7

Verbindungsmodi des CP 1242-7 (6GK7 242-7KX30-0XE0)

- Betriebsart "GPRS direkt"
  In der Betriebsart "GPRS direkt" gibt es keine unterschiedlichen Verbindungsmodi.
- Betriebsart "Telecontrol"
  Der CP kann für folgende Verbindungsmodi projektiert werden.
  - Verbindungsmodus "Permanent"
    Es besteht eine dauerhafte TCP-Verbindung zum Telecontrol-Server. Nach dem Verbindungsaufbau besteht eine dauerhafte TCP-Verbindung zum Telecontrol-Server, auch wenn nicht permanent Daten übertragen werden.
  - Verbindungsmodus "Temporär"
    Eine Verbindung zum Telecontrol-Server wird nur bei Bedarf aufgebaut.

Wenn eine TCP-Verbindung aufgebaut ist, erfolgt das Versenden von Prozessdaten, sobald die Telecontrol-Anweisungen in der CPU aufgerufen werden.

Eine Verbindung wird immer durch den CP aufgebaut. Wenn eine vom CP aufgebaute Verbindung unterbrochen wird, dann versucht der CP selbständig, die Verbindung wieder aufzubauen.

Verbindungsaufbau bei permanenten Stationen (Betriebsart "Telecontrol")

In der Betriebsart "Telecontrol" wird der Aufbau der permanenten Verbindung zum Telecontrol-Server beim Anlauf der Station aufgebaut. Bei Verbindungsunterbrechung kann der Aufbau der Verbindung durch eine Weck-SMS (siehe unten) veranlasst werden.

Verbindungsaufbau bei temporären Stationen (Betriebsart "Telecontrol")

Bei temporären Stationen kann ein Verbindungsaufbau durch folgende Ereignisse ausgelöst werden:

- Ereignis in der lokalen CPU, das programmtechnisch ausgewertet werden müssen. Programmtechnisch sind zwei Fälle zu unterscheiden:
  - Ereignisse, die zu einem einmaligen Verbindungsaufbau führen (z. B. Alarme oder Kommandos eines Bedieners).
  - Ablauf eines Zeitintervalls, das zu zyklischem Verbindungsaufbau führt (z. B. einmal täglich zur Datenübertragung)
- Anforderung durch einen Kommunikationspartner (OPC-Client oder S7-Station)
  Die Anforderung des Kommunikationspartners führt zu einem Verbindungsaufbau.
- Anforderung durch eine Engineering-Station für TeleService
  Die Anforderung, welche vom Telecontrol-Server bzw. TeleService-Gateway vermittelt wird, muss programmtechnisch nicht ausgewertet werden.

- Weck-SMS des Telecontrol-Servers
  Die Weck-SMS kann spontan am Telecontrol-Server ausgelöst werden. Auch ein zyklisches Versenden kann im Telecontrol-Server projektiert werden.
- Weckruf eines Telefons
  Der Weckruf kann von einem Telefon verschickt werden, dessen Rufnummer im STEP 7-Projekt autorisiert ist. Das Telefon muss die CLIP-Funktion (Übertragung der eigenen Rufnummer) unterstützen.
  Der Verbindungsaufbau mit dem (Haupt-) Telecontrol-Server wird veranlasst.
- Weck-SMS eines Telefons
  Die Weck-SMS kann von einem Telefon verschickt werden, dessen Rufnummer im STEP 7-Projekt autorisiert ist. Das Telefon muss die CLIP-Funktion (Übertragung der eigenen Rufnummer) und das Versenden von SMS unterstützen.
  Der Verbindungsaufbau mit dem in der SMS spezifizierten Telecontrol-Server wird veranlasst.

Beim Wecken einer temporären Station werden alle Daten übertragen, wenn diese sich seit der letzten Datenübertragung geändert haben.

Weckruf / Weck-SMS in der Betriebsart "Telecontrol"

Zum Weckruf bzw. zur Weck-SMS siehe Kapitel Wecken von Mobilfunk-CPs (Seite 1923).

Verbindungsaufbau in der Betriebsart "GPRS direkt"

In der Betriebsart "GPRS direkt" wird ein Verbindungsaufbau durch folgende Ereignisse ausgelöst:

- Ereignis in der lokalen CPU, das programmtechnisch ausgewertet wird.
- Anforderung durch einen Kommunikationspartner (keine Engineering-Station)
  Die Anforderung des Kommunikationspartners wird programmtechnisch durch Aufruf der Telecontrol-Anweisungen ausgewertet.
- Anforderung durch eine Engineering-Station für TeleService
  Die Anforderung, welche vom Telecontrol-Server bzw. TeleService-Gateway vermittelt wird, muss programmtechnisch nicht ausgewertet werden.

Wecken von Mobilfunk-CPs

Weckberechtigung durch "Autorisierte Rufnummern"

Voraussetzung dafür, dass der CP eine Weck-SMS oder einen Weckruf akzeptiert, ist die Autorisierung des sendenden Kommunikationspartners mithilfe seiner Rufnummer. Diese Rufnummern werden in STEP 7 beim CP in der Liste "Autorisierte Rufnummern" projektiert.

Hinweis

"Autorisierte Rufnummern" im STEP 7-Projekt

- Eine hier eingetragene Rufnummern berechtigt den Absender, der diese Rufnummer mit überträgt, einen Verbindungsaufbau auszulösen.
- Wenn in der Liste ein Stern (*) eingetragen wird, dann akzeptiert der CP SMS von allen Absendern.
- Wenn die Liste leer ist, dann kann der CP nicht zum Verbindungsaufbau geweckt werden.

Weck-SMS

Abhängig von der Verbindungsart und dem auslösenden Server bzw. dem vermittelnden TeleService-Server muss in der Weck-SMS folgender Text übertragen werden:

- Für Telecontrol-Verbindungen:
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Telecontrol-Server:
    `TELECONTROL`
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Haupt-Telecontrol-Server:
    `TELECONTROL MAIN`
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Ersatz-Telecontrol-Server:
    `TELECONTROL BACKUP`

  Die Projektierung der Telecontrol-Server für den GPRS-CP wird in STEP 7 unter "Telecontrol-Schnittstelle > Betriebsart > Haupt- bzw. Ersatz-Telecontrol-Server" vorgenommen.

  Hinweis

  Wecken durch ein Mobiltelefon

  - In einer Weck-SMS kann einer der oben aufgeführten Texte verwendet werden.
  - Durch einen Weckruf verbindet sich die Station immer mit dem Haupt-Telecontrol-Server.

- Für TeleService-Verbindungen:
  - Text für die Weck-SMS für den Aufbau einer Verbindung über den ersten projektierten TeleService-Server:
    `TELESERVICE`
    oder
    `TELESERVICE 1`
  - Text für die Weck-SMS für den Aufbau einer Verbindung über den zweiten projektierten TeleService-Server:
    `TELESERVICE 2`

  Die Projektierung der TeleService-Server für den GPRS-CP wird in STEP 7 unter "Telecontrol-Schnittstelle > TeleService-Berechtigung > 1. bzw. 2. TeleService-Server" vorgenommen.

Bevorzugte Mobilfunknetze

Auswahl der bevorzugten Mobilfunknetze

Zur Auswahl, in welche Netze sich ein Mobilfunk-CP bevorzugt einwählen soll, werden folgende Optionen angeboten:

- Automatisch
  Der CP wählt sich nach Vorgaben der SIM-Karte mit höchster Priorität in das Mobilfunknetz des Vertrags-Netzwerkbetreibers ein. Wenn die Einwahl in das Vertrags-Netz nicht erfolgreich ist, wählt sich der CP in andere Mobilfunknetze ein, mit denen der Vertrags-Netzwerkbetreiber Roaming-Verträge abgeschlossen hat und deren Zugangsdaten auf der SIM-Karte hinterlegt sind.
- Nur Vertragsnetz
  Der CP wählt sich nur in das Mobilfunknetz des Vertrags-Netzwerkbetreibers ein, dessen SIM-Karte im CP gesteckt ist. Kein Roaming.
- Vertragsnetz und alternative Netze
  Der CP wählt sich in erster Priorität in das Vertragsnetz ein. Wenn die Einwahl in das Vertrags-Netz nicht erfolgreich ist, wählt sich der CP in alternative Mobilfunknetze ein, die in der Liste der bevorzugten Netzwerkbetreiber mit abnehmender Priorität eingetragen werden.
  Die alternativen Netze werden in die Liste als "Public Land Mobile Network" (PLMN) eingetragen. PLMN setzt sich zusammen aus Mobile Country Code (MCC) und Mobile Network Code (MNC).
  Beispiel: 26276
  Dies ist das PLMN für das Testnetz der Siemens AG mit MCC = 262 und MNC = 76.

Online-Funktionen

Laden von Gerät beim CP 1243-8 IRC (V2.1)

Einschränkungen beim Laden aus einem CP 1243-8 IRC

Beim Laden der Projektierungsdaten aus einem CP 1243-8 IRC mithilfe der Online-Funktionen werden folgende Daten nicht vollständig geladen:

- Partnerstationsdaten
- Konfigurations- und Partnerdaten der Datenpunkte
- Referenzen zu PLC-Variablen mit Bezug zu Kommunikationspartnern

Um das STEP 7-Projekt nach dem Laden aus den Geräten vollständig nutzen zu können, müssen Sie die ursprünglichen Exportdateien aus STEP 7 V5 mit den Projektierungsdaten der Proxies in die einzelnen CP 1243-8 IRC laden (Partnerprojektierung importieren).

TeleService über Mobilfunk

Laden über TeleService

Das nachfolgende Verhalten gilt für folgende Mobilfunk-CPs:

- CP 1242-7 (6GK7 242-7KX30-0XE0)
- CP 1242-7 GPRS V2 (6GK7 242-7KX31-0XE0)
- CP 1243-7 LTE-EU (6GK7 243-7KX30-0XE0)
- CP 1243-7 LTE-US (6GK7 243-7SX30-0XE0)

Verbindungsressourcen bei TeleService

Die Funktion TeleService belegt in der Engineering-Station eine Verbindungsressource.

Die Funktion Laden in oder von Gerät während einer TeleService-Sitzung belegt in der Engineering-Station eine zweite Verbindungsressource.

Laden in Gerät

Führen Sie beim Mobilfunk-CP die Funktion "Laden in Gerät" über eine TeleService-Verbindung nur folgendermaßen aus:

1. Selektieren Sie in STEP 7 den CP.
2. Wählen Sie das Menü "Online" > "Laden in Gerät".
3. Wählen Sie im aufgeblendeten Dialog "Erweitertes Laden" die TeleService-Schnittstelle aus.
4. Laden Sie die Projektdaten über den Dialog "Erweitertes Laden".

Laden von Gerät

Die Funktion "Laden von Gerät" über eine TeleService-Verbindung wird von den Mobilfunk-CPs zusammen mit folgenden TeleService-Server-Applikationen unterstützt:

- TeleControl Server Basic ab Version V3
- TeleService Gateway ab Version V3

TeleService über Telecontrol

Kommunikationsweg bei TeleService über Telecontrol

Bei TeleService über Telecontrol wird die Verbindung zwischen Engineering-Station und entfernter S7-Station immer über einen TeleService-Server als Vermittler geführt.

Die Anforderung des Verbindungsaufbaus wird durch die Engineering-Station ausgelöst und durch den TeleService-Server weitergeleitet. Je nach verfügbaren Verbindungswegen zur Station leitet der TeleService-Server die Anfrage über LAN oder durch eine Weck-SMS an die

Station weiter. Der TeleControl-CP in der S7-1200-Station baut die Verbindung über den TeleService-Server mit der Engineering-Station auf.

Als Vermittler (TeleService-Server) fungieren alternativ:

- Ein Telecontrol-Server
  Der Telecontrol-Server kann ein separater PC sein oder als Applikation "TCSB" auf der Engineering-Station installiert sein.
- Ein TeleService-Gateway
  Ein TeleService-Gateway wird eingesetzt, wenn kein Telecontrol-Server zur Verfügung steht.

Der TeleService-Server kann über LAN oder Internet mit der Engineering-Station verbunden sein, von der aus die TeleService-Funktion aufgerufen wird.

Voraussetzungen für TeleService über Telecontrol

- Ein TeleService-Server
- Das STEP 7-Projekt mit den erforderlichen Stationen

Hinweis zur Projektierung

Der TeleService-Server wird nicht in STEP 7 projektiert.

Fernverbindung über Telecontrol aufbauen

Voraussetzungen für TeleService über Telecontrol

Zu den Voraussetzungen und Mechanismen siehe Kapitel TeleService über Telecontrol (Seite 1926).

Starten einer TeleService-Sitzung über Telecontrol

Starten Sie TeleService über Telecontrol folgendermaßen:

1. Selektieren Sie im Projekt auf der autorisierten Engineering-Station die entfernte S7-Station, zu der Sie eine TeleService-Verbindung über WAN aufbauen möchten.
2. Öffnen Sie den Dialog "Online verbinden" alternativ über folgende Möglichkeiten:
   - Schaltfläche "Online verbinden"
   - Kontextmenü "Online verbinden" (rechte Maustaste)
   - Menü "Online" > "Online verbinden"

   Der Dialog "Online verbinden" öffnet sich.
3. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" den Schnittstellentyp "TeleService über Telecontrol" aus.
4. Wählen Sie in der Klappliste "PG/PC-Schnittstelle" die Option "TeleService board" aus, wenn diese nicht automatisch angezeigt wird.

5. Klicken Sie auf das "Verbinden"-Symbol neben der Klappliste "PG/PC-Schnittstelle".
   Der Dialog "Fernverbindung über Telecontrol aufbauen" öffnet sich.
6. Nehmen Sie die erforderlichen Einstellungen in diesem Dialog vor.
   Details hierzu finden Sie in den Tooltip-Kaskaden von STEP 7.

Folgende Angaben sind für einen erfolgreichen Verbindungsaufbau erforderlich:

Notwendige Angaben für den Verbindungsaufbau mit der S7-Station

Die nachfolgenden Angaben sind im Dialog "Fernverbindung aufbauen" erforderlich:

- IP-Adresse oder DNS-Name des Telecontrol-Servers
- TCP-Port-Nummer des Telecontrol-Servers oder des DSL-Routers, über den die Verbindung zwischen Engineering-Station und entfernter S7-Station läuft.
- Server-Passwort der ES zur Authentifizierung der Engineering-Station beim Telecontrol-Server
  Nur erforderlich, wenn in der Applikation "TCSB" ein gruppenspezifische Passwort projektiert wurde.
- TeleService-Benutzername
  Siehe CP-Projektierung in STEP 7.
- TeleService-Passwort
  Siehe CP-Projektierung in STEP 7.
- Zugangs-ID des CP
  Nur erforderlich, wenn mehrere TeleControl-CPs in der Station vorhanden sind. Siehe CP-Projektierung in STEP 7.

Status

Verbindungszustände bei TeleService über Telecontrol

Die nachfolgend beschriebenen Verbindungszustände können im Dialog "Fernverbindung über Telecontrol aufbauen" angezeigt werden.

Beim Öffnen des Dialogs

- Nicht verbunden
  Es besteht keine Verbindung zur entfernten S7-Station. Der Verbindungsaufbau wurde noch nicht gestartet.

Beim Verbindungsaufbau

Wenn der Verbindungsaufbau durch Klick auf die Schaltfläche "Verbinden" gestartet wurde, dann werden bei erfolgreichem Verbindungsaufbau folgende Zustände der Reihe nach angezeigt:

- Verbinden mit Telecontrol-Server
  Die Engineering-Station verbindet sich mit dem Telecontrol-Server.
- Warten auf S7-Station
  Die Weck-SMS wurde an die entfernte Station abgesetzt. Warten auf Antwort der Station.

- Authentifizierung an der S7-Station
  Die S7-Station hat über GPRS und Internet eine IP-Verbindung mit der Engineering-Station aufgebaut und prüft die empfangenen Anmelde- und Authentifizierungsdaten.
- Verbunden
  Die Station hat die Verbindung mit der Engineering-Station erfolgreich aufgebaut.

Bei nicht erfolgreichem Verbindungsaufbau

Bei nicht erfolgreichem Verbindungsaufbau können folgende Zustände angezeigt werden:

- Telecontrol-Server nicht erreichbar
  Mögliche Ursachen:
  - Die Verbindung zwischen Engineering-Station und Telecontrol-Server ist unterbrochen.
  - Der Telecontrol-Server ist ausgeschaltet.
- Server-Passwort falsch
  Ursache: Im Dialog wurde das falsche Server-Passwort zur Anmeldung und Authentifizierung am Telecontrol-Server eingegeben.
- S7-Station meldet sich nicht
  Mögliche Ursachen:
  - Die Kommunikation zwischen Telecontrol-Server und Station ist gestört.
  - Die Verbindung zwischen Mobilfunknetz und Internet ist gestört.
  - Die Internetverbindung ist gestört.
  - Der Telecontrol-Server konnte keine Weck-SMS verschicken.
  - Der CP hat keine Weck-SMS empfangen.
  - Der SMS-Absender wurde nicht in der Liste der autorisierten Weck-Rufnummern projektiert.
- TeleService-Benutzername oder TeleService-Passwort falsch
  Mögliche Ursachen:
  - Im Dialog wurde der falsche TeleService-Benutzername oder das falsche TeleService-Passwort zur Authentifizierung am CP eingegeben.
  - Der TeleService-Benutzername oder das TeleService-Passwort wurde in STEP 7 nicht projektiert.
- Alle TeleService-Zugangspunkte sind belegt.
- Der CP ist dem Telecontrol-Server nicht bekannt.
  Ursache: Der CP stammt aus einem STEP 7-Projekt, das nicht dem Projekt des Telecontrol-Servers entspricht.
- Keine Ressourcen für TeleService im CP vorhanden: Bitte kontaktieren Sie die Hotline.
- Protokollfehler
  Ursache: Inkompatible Daten oder Daten von inkompatiblem Teilnehmer. Bitte kontaktieren Sie die Hotline.

Baustein-Bibliothek Telecontrol ST7

Bibliothek anlegen

Lieferform der Bausteinbibliothek

Die Programmbausteine der TD7onCPU-Bibliothek werden als Globale Bibliothek für STEP 7 genutzt.

Die Bibliothek wird archiviert und gepackt im Internet unter der folgenden Adresse zur Verfügung gestellt:
Link: (https://support.industry.siemens.com/cs/ww/de/view/109755374)

Der Name des Datei-Archivs hat den folgenden Aufbau: TD7_<Version>.zip

Anlegen der Bibliothek

Gehen Sie zur Nutzung der Bibliothek in STEP 7 folgendermaßen vor:

1. Legen Sie im STEP 7-Installationsverzeichnis Ihrer Engineering-Station ein geeignetes Verzeichnis an, beispielsweise mit dem Namen "Telecontrol ST7".
2. Speichern Sie die gepackte und archivierte Bibliothek (*.zip) in dieses Verzeichnis.
3. Entpacken Sie das Dateiarchiv mit den Mitteln des Betriebssystems.
   Die Bibliothek liegt jetzt als archivierte globale Bibliothek unter dem Namen "Telecontrol ST7.zal15" vor.
4. Öffnen Sie das STEP 7-Projekt.
5. Wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek dearchivieren".
   Der Dialog "Archivierte globale Bibliothek dearchivieren" wird geöffnet.
6. Wählen Sie die Archivdatei aus.
7. Empfehlung: Öffnen Sie die globale Bibliothek schreibgeschützt.
   In der Voreinstellung ist die Option "Schreibgeschützt öffnen" aktiviert.
8. Klicken Sie auf Öffnen.
   Der Dialog "Ordner suchen" wird geöffnet. Wählen Sie das Zielverzeichnis aus, an dem die archivierte globale Bibliothek entpackt wird.
9. Klicken Sie auf "OK".
   Die Bibliothek wird entpackt und geöffnet.

Sie finden die Bibliothek im STEP 7-Projekt in der Bibliotheksansicht (Task Card) in der Palette "Globale Bibliotheken" unter dem Namen "Telecontrol ST7_<Version>".

Aufbau der Bibliothek

Aufbau der globalen Bibliothek

Die geöffnete Bibliothek enthält folgende Verzeichnisse:

- Typen
  - PLC data types (UDT)
    Hier befinden sich die von den Bibliotheken verwendeten PLC-Datentypen (UDTs).
  - System data types (SDT)
    Hier befinden sich die von den Bibliotheken verwendeten Systemdatentypen.
- Kopiervorlagen
  Hier befinden sich die folgenden Programmbausteine:
  - CPU300/400
    Programmbausteine für die S7-300/400-CPU
  - CPU1500
    Programmbausteine für die S7-1500-CPU

Kopiervorlagen

Die beiden Bibliotheken unter "Kopiervorlagen" für die S7-300/400 und die S7-1500 sind jeweils in folgende Ordner unterteilt:

- Data point typicals
  Datenpunkt-Typicals (Bausteine) für die Übertragung von Daten
- Optional blocks
  Optionale Bausteine, die Sie bei Bedarf verwenden können.
- System blocks
  Bausteine, die für die Basiskommunikation verwendet werden.
  Je nach verwendeter SIMATIC-Familie finden Sie folgende Hilfsbausteine für die Kommunikation zwischen TIM und CPU in separaten Ordnern:
  - BCom
    Bausteine für die Bearbeitung des Kommunikationsfachs der S7-400 und S7-1500
  - PCom
    Bausteine für die Bearbeitung des Kommunikationsfachs einer S7-300 mit P-Bus (CPU ohne Partyline)
  - XCom
    Bausteine für die Bearbeitung des Kommunikationsfachs einer S7-300 mit X-Verbindungen

Generieren, Verwenden und Aufrufen der Bausteine

TD7onCPU-Bausteine erzeugen

Nachdem Sie die TD7-Bibliothek in der Palette "Globale Bibliotheken" unter dem Namen "Telecontrol ST7" geöffnet haben, können Sie die Bausteine für die Basiskommunikation für eine oder mehrere CPUs generieren.

Voraussetzungen

Folgende Voraussetzungen müssen vor dem Generieren der Bausteine erfüllt sein:

- Die Kommunikationsmodule müssen untereinander vernetzt sein.
- Eine TIM 1531 IRC muss über Ethernet mit einer CPU vernetzt sein.
- Eine TIM 1531 IRC darf keiner CPU zugeordnet sein.
  In der Parametergruppe "Teilnehmernummern" muss das Feld "Selektierte CPU" leer sein.
- In der TIM, deren CPU TD7onCPU nutzen soll, darf kein Datenpunkt angelegt sein.
- Im Nachrichten-Editor der TIM darf noch keine Nachricht angelegt sein.
- Zwischen den beteiligten Kommunikationspartnern müssen gültige Telecontrol-Verbindungen angelegt sein.
- Die Bibliothek "Telecontrol ST7" muss in der Task Card "Globale Bibliotheken" geöffnet sein.

Abweichende Voraussetzungen bei Modulen mit importierten Projektierungsdaten

Beachten Sie weitere Voraussetzungen bei Kommunikationsmodulen, deren SINAUT-Projektierung aus einem STEP 7 V5-Projekt importiert wurde. Bei diesen Modulen ist in der Parametergruppe "Grundeinstellungen > Projektierung" die Option "Telecontrol-Konfiguration" auf "Importieren" gesetzt.

Für diese Module gelten abweichende Voraussetzungen für das Generieren der Bausteine:

- Zwischen den beteiligten Kommunikationspartnern müssen keine ST7-Telecontrol-Verbindungen angelegt werden.
  Die entsprechenden Informationen sind in den importierten Projektierungsdaten hinterlegt.
- Die CPU des Kommunikationsmoduls muss die gleiche IP-Adresse haben wie im STEP 7 V5-Projekt. Die CPU wird dem Kommunikationsmodul über ihre IP-Adresse zugeordnet.

Vorgehensweise für das Generieren

Generieren Sie die Bausteine für die Basiskommunikation folgendermaßen:

1. Selektieren Sie die gewünschte TIM.
2. Wählen Sie den Kontextmenübefehl (rechte Maustaste) "TD7onCPU-Bausteine generieren".
3. Wählen Sie im Folgedialog eine oder mehrere CPUs aus und klicken Sie auf "Generieren...".

Die TD7onCPU-Bausteine werden in den selektierten CPUs unter den Programmbausteinen in einem neuen Verzeichnis "TD7onCPU" angelegt und übersetzt.

Bausteine verwenden

- System blocks
  Die erforderlichen Bausteine für die Basiskommunikation werden automatisch beim Generieren in der CPU angelegt.
  Diese Bausteine müssen Sie nicht manuell in der CPU verwenden.
  Die Systembausteine erledigen zentrale Aufgaben wie Anlauf, Überwachung der Verbindungen, Erreichbarkeit der Verbindungspartner, Eintragen bzw. Entnehmen von Daten in das Sendefach bzw. aus dem Empfangsfach der Kommunikations-DBs, Generalabfrage, Zeitführung, Kommunikationsabwicklung usw.
  Kommunikations-DBs:
  - S7-300/400
    Für jede Verbindung wird automatisch ein eigener Kommunikations-DB angelegt: Er enthält das Sende- und Empfangsfach sowie alle Daten, die für die Steuerung und Überwachung dieser Verbindung benötigt werden.
  - S7-1500
    Für die CPU 1500 gibt es nur einen Kommunikations-DB "BConnectData". In diesem DB ist im Array "BConnection" für jede lokale Verbindung je eine Verbindungsinstanz abgelegt.
- Data point typicals
  Die Datenpunkt-Typicals müssen Sie je nach Kommunikationsaufgaben in der jeweiligen CPU verwenden.
  Ziehen Sie die einzelnen Bausteine mit der Maus aus der Globalen Bibliothek in das Anwenderprogramm der CPU. Der zugehörige Instanz-Datenbaustein wird automatisch beim Verwenden eines Datenpunkt-Typicals angelegt.
  - Dat256D_S / Dat256D_R
    Kopieren Sie für diese Typicals auch den UDT "TransmitBlock" aus der Bibliothek (Typen > PLC data types (UDT)) in das Verzeichnis "PLC-Datentypen" der CPU. Der UDT wird vom Typical nicht automatisch referenziert.
- Optional blocks
  Verwenden Sie auch die optionalen Bausteine je nach Kommunikationsaufgaben der Station in der CPU.
  Ziehen Sie die einzelnen Bausteine mit der Maus aus der Globalen Bibliothek in das Anwenderprogramm der CPU.

Aufruf im Anwenderprogramm

- System blocks
  Nur die beiden folgenden FCs müssen im Anwenderprogramm aufgerufen werden:
  - StartUp
    Rufen Sie den FC im Anlauf-OB auf. Der Baustein hat keine Parameter.
    Für den Aufruf des FC StartUp müssen Sie zuvor den Anlauf-OB (OB100) anlegen.
  - BasicTask
    Rufen Sie den FC im zyklischen OB auf.
    Der DB BasicData wird automatisch angelegt. Der DB enthält alle zentral benötigten Daten, u. a. die Buchführung über alle Kommunikationspartner sowie über alle zu verwaltenden Verbindungen.
    Zu den verschiedenen FCs "BasicTask" siehe Kapitel Hinweis zu den FCs "BasicTask_*" (Seite 1934).
- Data point typicals
  Rufen Sie je nach Kommunikationsaufgaben alle benötigten Datenpunkt-Typicals im Anwenderprogramm auf.
- Optional blocks
  Rufen Sie je nach Kommunikationsaufgaben alle benötigten optionalen Bausteine im Anwenderprogramm auf.

Zum Aufruf der Bausteine siehe auch Kapitel Das zyklische SINAUT-Programm (Seite 1935).

Hinweis

Fehleranzeige zur Bibliothekskonformität

In der Parametergruppe "Übersetzung" von Bausteinen kann unter dem Eintrag "Bibliothekskonformität" ein Fehler angezeigt werden (Aufruf von Einzelinstanzen). Dies wird durch den Aufruf globaler Datenbausteine verursacht, bspw. dem DB BasicData.

Da die Bibliothek "Telecontrol ST7" keinen "Know-How-Schutz" besitzt, können Sie diese Fehleranzeigen ignorieren.

Hinweis zu den FCs "BasicTask_*"

Varianten des FC "BasicTask"

In der TD7-Bausteinbibliothek "Telecontrol ST7" für STEP 7 Professional gibt es verschiedene Varianten des FC "BasicTask".

Abhängig von der SIMATIC-Familie und dem Rückwandbus haben die FCs einen unterschiedlichen Suffix am Ende des Bausteinnamens:

- BasicTask_B
  Für S7-1500-CPU
- BasicTask_B
  Für S7-400-CPU

- BasicTask_X
  Für S7-300-CPU mit Partyline
- BasicTask_P
  Für S7-300-CPU ohne Partyline

Partyline-CPUs sind: CPU 312.. / 313.. / 314.. / 315.. bis CPU 315-2 DP und C7-Geräte.

Zur Bausteinbeschreibung siehe Kapitel FC BasicTask_* (Seite 2056).

Bezeichnung "BasicTask" für drei FCs

In den nachfolgenden Kapiteln wird an vielen Stellen Bezug auf den jeweiligen FC genommen. Da die Beschreibung der verschiedenen Bausteine meist für mehrere SIMATIC-Familien gültig ist, werden die drei FCs nachfolgend als "BasicTask" bezeichnet. Im jeweiligen Kontext ist die FC-Variante für den betreffenden CPU-Typ gemeint.

Aufbau des Anwenderprogramms für TD7onCPU

Das zyklische SINAUT-Programm

Einleitung

Die Struktur der TD7-Bausteine im Anwenderprogramm (OB1) wird nachfolgend erläutert.

Die TD7-Bausteine müssen in jedem OB1-Zyklus bearbeitet werden. Halten Sie die Aufruf-Reihenfolge der Bausteine ein, soweit nicht anders angegeben.

Sonstige anwenderspezifische Teile des Programms können vor oder nach den TD7-Bausteinen in den OB1 eingebunden werden oder, soweit sinnvoll, innerhalb der TD7-Bausteine.

Sie können den SINAUT-spezifischen Teil im OB1 strukturieren, indem Sie ihn in unterlagerten FCs aufrufen.

Alle Datenpunkt-Typicals sind FBs. Beim Aufruf eines FB ist ein Instanz-DB anzugeben. Die Nummer dieses Instanz-DB ist identisch mit der Objekt-Nummer des Datenpunkt-Objekts.

Struktur der TD7-Bausteine im OB1

Zyklischer OB1	
BasicTask	Am Anfang ist immer der jeweilige FC-BasicTask aufzurufen. Er übernimmt die SINAUT-Basisaufgaben, die grundsätzlich benötigt werden.
Optionale SINAUT Basis-Funktionen	Rufen Sie im Anschluss an FC BasicTask folgende optionale Bausteine (FCs) auf, wenn Sie diese benötigen: • ListGenerator Generierung von Adresslisten für Empfangstelegramme mit unvollständiger Ziel-Adresse • TimeTask Stellt die Uhrzeit zur Verfügung
	Rufen Sie im weiteren Programmablauf bei Bedarf weitere optionale FCs auf, z. B.: • Trigger Zeitabhängiges Starten von Anwenderprogrammen und Daten-Telegrammen • PartnerStatus Anzeige des Zustands von Partnern • PartnerMonitor Erweiterte teilnehmerspezifische Anzeigen und Steuermöglichkeiten • PulseCounter Zählimpulserfassung • PathStatus Anzeige des Status des Verbindungswegs zu einem Partner **Hinweis:** Die beiden folgenden Bausteine dürfen nicht explizit aufgerufen werden, da sie über eine interne Schnittstelle aktiviert werden. • TestCopy • TestcopyDB
Datenpunkt-Typicals	Rufen Sie im weiteren Programmablauf die Datenpunkt-Typicals zum Senden und Empfangen von Daten auf. Die Reihenfolge der einzelnen Typicals ist beliebig. Die Anzahl der aufzurufenden Typicals und die benötigten Typen sind abhängig von den Datenmengen und Datentypen, die zu senden bzw. zu empfangen sind. **Hinweis:** Die Instanz-DBs dieser FBs werden über die Partner-Objektnummer des Kommunikationspartners referenziert.

Beispiel eines TD7-Programms für eine Station

Zyklischer OB1	
BasicTask	Am Anfang des zyklischen Programms ist immer der jeweilige FC BasicTask aufzurufen. Generell stehen Daten nach dem Empfang zur Weiterbearbeitung bereit. Der FC hat nur den einen Parameter "UserFC". Normalerweise kann die 0 angegeben werden. Möchten Sie empfangene Daten anwenderspezifisch weiterbearbeiten, dann geben Sie hier die Nummer eines FC an, in dem das Programm für diese Weiterverarbeitung enthalten ist. Über diese Schnittstelle können Sie empfangene Daten auch chronologisch in einen Archivspeicher kopieren.
TimeTask	Optional kann direkt im Anschluss an den FC BasicTask der FC TimeTask aufgerufen werden. Der FC hat keine Parameter. Der FC TimeTask ist einzubinden, wenn Sie die Uhrzeit benötigen. Damit können Daten-Telegramme zeitgestempelt werden. Sie können die Uhrzeit aber auch nutzen, um Programmteile zu einem bestimmten Zeitpunkt zu starten oder Daten-Telegramme zeitabhängig übertragen. Dafür wird dann der nachfolgende FC Trigger benötigt. Voraussetzung für die Nutzung des FC TimeTask ist, dass die CPU von einer lokalen TIM-Baugruppe mit der Uhrzeit versorgt wird. Dies legen Sie über die Projektierung fest.
Trigger	Der FC Trigger kann optional eingebaut werden. Der FC setzt sein Ausgang für die Dauer von einem OB1-Zyklus, wenn der am FC parametrierte Zeitpunkt oder das Zeitraster erreicht ist. Der FC kann mehrmals eingebunden werden, wenn mehrere Zeitpunkte oder verschiedene Zeitraster benötigt werden. Voraussetzung für die Nutzung des FC ist, dass zuvor der FC TimeTask im OB1-Programm aufgerufen wird (siehe oben) und die CPU-Zeit schon einmal gestellt worden ist.
PartnerStatus	Der FC PartnerStatus kann optional eingebaut werden. Der FC zeigt für maximal 8 Kommunikationspartner die Erreichbarkeit an.
ListGenerator	• ListGenerator300 für S7-300-CPU • ListGenerator400 für S7-400-CPU • ListGenerator1500 für S7-1500-CPU Der FC kann optional eingebaut werden. Der FC wird benötigt, wenn die Station Telegramme empfängt, die keine oder eine unvollständige Ziel-Adresse enthalten. Dies passiert, wenn in einem Partner bei Datenpunkt-Typicals auf die Parametrierung der Ziel-Adresse ganz oder teilweise verzichtet wurde. ("PartnerNo" und "PartnerObjectNo" wurden nicht ausgefüllt; damit wird an alle bekannten Ziel-Teilnehmer übertragen.)
PulseCounter PathStatus PartnerMonitor	Bei Bedarf

Zyklischer OB1	
Datenpunkt-Typicals	Im Anschluss an die FCs für SINAUT-Basisaufgaben werden die Datenpunkt-Typicals zum Senden und Empfangen von Daten aufgerufen. Die Reihenfolge der einzelnen Typicals ist beliebig. Die Anzahl der aufzurufenden Typicals und die benötigten Typen sind abhängig von den Datenmengen und Datentypen, die zu senden bzw. zu empfangen sind. Für eine Station gilt typischerweise: • Gesendet werden: – Binäre Informationen wie Zustandsmeldungen und Alarme – Analogwerte – Zählwerte • Empfangen werden: – Befehle – Sollwerte, Grenzwerte, Parameter
Bin04B_S	Ein oder mehrere FBs für die Erfassung und Übertragung von binären Informationen wie Meldungen, Alarmen usw.
Ana04W_S	Ein oder mehrere FBs für die Erfassung und Übertragung von Analogwerten
Cnt01D_S / Cnt04D_S	Ein oder mehrere FBs für die Erfassung und Übertragung von Zählwerten Voraussetzung für die Nutzung der FBs ist, dass der FC PulseCounter in einem Weckalarm-OB eingebunden ist, z. B. in den OB35. Dieser FC sorgt im Hintergrund für die zeitgesteuerte Erfassung der Zählimpulse.
Cmd01B_R	Ein oder mehrere FBs für den Empfang und die Ausgabe von Befehlen
Set01W_R / Par12D_R	Ein oder mehrere FBs für den Empfang und die Ausgabe von Sollwerten, Grenzwerten oder Parametern
Dat12D_S / Dat256D_S	Ein oder mehrere FBs für die Erfassung und Übertragung von 12 Daten-Doppelworten mit beliebigem Informationsinhalt Bei diesen Typicals entfällt die datenspezifische Verarbeitung und Änderungskontrolle. Dies ist Aufgabe des Anwenderprogramms. Optional kann eine Änderungskontrolle aktiviert werden, die bei jeder Bit-Änderung eine Übertragung auslöst.

TD7-Programme für Zentralen und Knotenstationen

Prinzipiell sehen die TD7-Programme von Zentralen und Knotenstationen gleich aus wie für eine Station. Für die Sende-Typicals einer Station werden in einer Zentrale die entsprechenden Empfangs-Typicals verwenden.

In einer Knotenstationen werden entsprechend der Übertragungsrichtung sowohl Sende-Typicals als auch die entsprechenden Empfangs-Typicals verwendet.

In der Zentrale bietet es sich an, das OB1-Programm nach Stationen geordnet zu strukturieren, d. h. alle Daten-Typicals, die zur gleichen Station gehören, in einem FC aufzurufen. Sehr übersichtlich wird es, wenn die Nummer des FC mit der Teilnehmernummer der Station identisch ist.

Weckalarm-OB

Einleitung

Bringen Sie in einer CPU nur dann zeitgesteuerte TD7-Bausteine ein, wenn in dieser CPU schnelle Zählimpulse erfasst werden müssen, die innerhalb eines OB1-Zyklus aufgrund einer zu langen Zykluszeit nicht sicher erfasst werden könnten.

Die Zählimpulse werden über eine beliebige Digital-Eingabebaugruppe erfasst. Um die Impulse sicher zu erfassen, müssen die genutzten digitalen Eingänge in einem festen Zeitraster auf Änderung abgefragt werden. Dabei richtet sich das Zeitraster nach der Dauer des kürzesten Zählimpulses. Die minimale Zählimpulsdauer darf 50 ms betragen. Für die Dauer der Pause gilt das gleiche. Daraus ergibt sich eine maximale Zählfrequenz von 10 Hz.

Das Zeitraster, in dem die Zählimpulserfassung durchgeführt werden soll, muss etwa die Hälfte der Zählimpulsdauer betragen, d.h. bei 50 ms in einem Raster von ca. 25 ms.

Für diese zeitgesteuerte Zählimpulserfassung ist bei einer S7-300-CPU der OB35 zu programmieren, bei einer S7-400-CPU einer der zur Auswahl stehenden Weckalarm-OBs OB30 bis OB38, bei einer S7-1500-CPU ein Weckalarm-OB mit einer Nummer von 30..38 / ≥ 123.

Alle Weckalarm-OBs haben ein voreingestelltes Zeitintervall, bei OB35 z. B. 100 ms. Dieses lässt sich in 1 ms-Schritten abändern. Die Einstellung eines Weckalarm-OB auf z. B. 25 ms ist somit möglich.

Das Zeitintervall für einen Weckalarm-OB ändern Sie über den Eigenschaftendialog (S7-400/1500), bei S7-300 über den Eigenschaftendialog der CPU.

Tabelle 1-139 Aufruf des FC PulseCounter im Weckalarm-OB

OB35	
Pulse Counter	Ein oder mehrere FC PulseCounter können für die Erfassung von Zählimpulsen eingebunden werden. Der FC PulseCounter bearbeitet bis zu 8 Impulseingänge einer beliebigen Digitaleingabe. Die erfassten Zählimpulse werden in programmierbare SIMATIC-Zähler aufsummiert. Diese greifen auf die Funktionsbausteine zurück, welche die Zählwert-Telegramme zusammenstellen (FB-Cnt0x_S). Anwenderspezifische Weckalarm-Funktionen, die unabhängig vom SINAUT-Programm benötigt werden, können an beliebiger Stelle in den Weckalarm-OB eingebunden werden.

Programmierfehler-OB

Gültigkeit

Nur für S7-300/400

Funktion

Wenn in einer S7-300/400-CPU ein Baustein aufgerufen wird, der nicht vorhanden ist, geht die CPU im Normalfall in STOP. Im Diagnosepuffer wird angezeigt, welcher FB, FC oder DB gefehlt hat. Sie können dann den fehlenden Baustein nachladen und die CPU erneut starten.

Wenn Sie aber nicht möchten, dass die CPU bei einem fehlenden Baustein in STOP geht oder nur dann in STOP geht, wenn bestimmte Bausteintypen oder Bausteinnummern fehlen, dann können Sie das von Ihnen gewünschte Verhalten im OB121 festlegen.

Schon wenn Sie OB121 als leeren Baustein in die CPU geladen haben, reicht dies, um die CPU immer weiter laufen zu lassen, wenn ein Baustein fehlt. Wenn Sie selektiver entscheiden möchten, wann die CPU weiter laufen bzw. in STOP gehen soll, dann bringen Sie den OB121 in das Anwenderprogramm ein.

In Verbindung mit SINAUT ST7 ist es möglich, dass eine CPU in STOP geht, wenn sie von einer anderen CPU Daten empfängt, die sie nicht oder noch nicht kennt. Dies ist beispielsweise der Fall, wenn Sie in einer Station ein Datenpunkt-Typical hinzufügen und mit einer kompletten Zieladresse versehen (Ziel-Teilnehmer-Nr. plus Ziel-Objekt-Nr.). Die angegebene Ziel-Objekt-Nr. kann in folgendem Fall zum STOP beim Ziel-Teilnehmer führen:

- Sobald Sie ein neues Datenpunkt-Typical in einer Station installiert haben, werden die Daten zum Ziel übertragen.
- Wenn in der Ziel-CPU noch kein zugehöriges Empfangs-Typical eingebaut ist, ist auch die Ziel-Objekt-Nr. (= Instanz-DB des Empfangs-Typicals) nicht vorhanden.

Ergebnis ist, dass die CPU in STOP geht, sobald die Daten empfangen werden.

Damit dies nicht passiert, wird empfohlen, den FC ST7ObjectTest in OB121 aufzurufen.

Tabelle 1-140 Aufruf des FC ST7ObjectTest im Programmierfehler-OB

OB121	
ST7ObjectTest	Der Aufruf des FC ST7ObjectTest in OB121 verhindert einen CPU-STOP, wenn die CPU Daten mit unbekannter Ziel-Objekt-Nr. empfängt. Andere Aufrufe können unabhängig vom Aufruf des FC ST7ObjectTest an beliebiger Stelle im OB121 eingebunden werden. FC ST7ObjectTest hat einen Parameter "StopInOtherCases". Hier können Sie angeben, was in anderen Fällen passieren soll (STOP oder Weiterlaufen), wenn OB121 aufgerufen wurde, weil ein anderer Datenbaustein oder ein FB oder FC fehlt.

Typen

Die folgenden Datentypen werden in Bausteinen der S7-300/400 und der S7-1500 verwendet.

PLC data types (UDT)

Verwendete UDTs

Der folgende PLC-Datentyp (UDT) wird von der Bausteinbibliothek verwendet:

- TransmitBlock
  UDT für die Datenübertragung zwischen CPU und TIM
  Kopieren Sie den UDT bei Verwendung des Typicals "Dat256D_S / Dat256D_R" in das Verzeichnis "PLC-Datentypen" der jeweiligen CPU. Der UDT wird vom Typical nicht automatisch referenziert.
  Der UDT benötigt keine Parametrierung.

System data types (SDT)

Verwendete SDTs

Die folgenden Systemdatentypen (SDTs) werden vom DB BasicData verwendet und beim Generieren / Übersetzen der TD7-Bausteine automatisch in das Verzeichnis "PLC-Datentypen" der CPU übernommen:

- ConnectionDescription
  Der SDT ist Bestandteil des DB "BasicData" und vom Datentyp "Array[1..1] of ConnectionDescription". Er enthält folgende Angaben zur Verbindungsbeschreibung:
  - Nummer des Kommunikations-DB
  - Verbindungstyp
  - CFB-Nummer im Fall einer S7-Verbindung
  - Teilnehmernummer der lokalen TIM
- SubscriberObject
  Der SDT ist Bestandteil des DB "BasicData" und vom Datentyp "Array[1..1] of SubscriberObject". Er enthält folgende Angaben zum Kommunikationspartner:
  - Teilnehmernummer des Partners
  - Array-Index der Verbindungsbeschreibung
  - Teilnehmertyp
  - Angabe, ob der Partner lokal oder remot ist.

Kopiervorlagen (S7-300/400, S7-1500)

Gültigkeit

In den Kopiervorlagen der globalen Bibliothek befinden sich die folgenden Programmbausteine:

- CPU300/400
  Programmbausteine für die S7-300/400-CPU
- CPU1500
  Programmbausteine für die S7-1500-CPU

Die Bausteine sind weitgehend identisch. Soweit nicht anders erwähnt, gilt die Beschreibung gleichermaßen für die Bausteine der S7-300/400 und S7-1500.

Wenn einzelne Bausteine nicht für alle genannten SIMATIC-Familien gültig sind oder wenn sich Funktionen für die SIMATIC-Familien unterscheiden, wird ausdrücklich darauf hingewiesen.

Data point typicals

Typen und Übersicht der Datenpunkt-Typicals

Typen der Datenpunkt-Typicals

Datenpunkt-Typicals bearbeiten einen oder mehrere Datenpunkte vom gleichen Informationstyp, z. B. 4 Byte binäre Information, 4 Analogwerte oder 1 Byte Befehle usw.

Beim Aufruf eines Datenpunkt-Typicals (FB) ist der jeweils korrespondierende Instanz-DB anzugeben, in welchen die Daten geschrieben bzw. aus dem die zu übertragenden Daten gelesen werden.

Übertragungsrichtung der Typicals

Ein Datenpunkt-Typical für einen bestimmten Informationstyp und eine Informationsmenge gibt es immer in zwei Ausprägungen:

- Ein Typical zum Erfassen und Senden
- Ein Typical zum Empfangen und Ausgeben

Bei der Nutzung der Datenpunkt-Typicals wird daher nach der Übertragungsrichtung unterschieden:

- Sendende Typicals
  Sendende Typicals verarbeiten Daten und versenden diese an den remoten Partner.
  Sie haben die Endung "_S". Beispiele: Bin04B_S, Ana04W_S
- Empfangende Typicals
  Empfangende Typicals empfangen die Daten von ihrem remoten Partner.
  Sie haben die Endung "_R". Beispiele: Bin04B_R, Ana04W_R

Bei der Datenübertragung ist bei den beiden kommunizierenden Partnern immer ein korrespondierendes Typical-Pärchen beteiligt.

Beispiele:

- Beobachtungsrichtung
  - Die Station versendet binäre Daten mit dem Typical Bin04B_S.
  - Die Zentralestation empfängt die Daten mit dem Typical Bin04B_R.
- Steuerungsrichtung
  - Die Zentralestation versendet einen Befehl mit dem Typical Cmd01B_S.
  - Die Station empfängt den Befehl mit dem Typical Cmd01B_R.

Struktur der Typical-Namen

Hinweis

Keine SMS-Bausteine

Anders als in der Bausteinbibliothek für STEP 7 V5 stehen in der globalen Bibliothek "Telecontrol ST7" keine Bausteine für das Senden von SMS zur Verfügung.

Das Senden von E-Mails und oder SMS projektieren Sie in STEP 7 Professional im TIA Portal über den Nachrichteneditor, siehe Kapitel Projektierung der Datenpunkte und Nachrichten (Seite 1848).

Die 8-stelligen Namen der Datenpunkt-Typicals werden nach folgendem Schema vergeben:

Tabelle 1-141 Struktur der 8-stelligen Typical-Namen

1	2	3	4	5	6	7	8
Datenpunkt-Typ			**Anzahl der Daten**		**Datenformat**	**Frei**	**S** Sende-Funktion
Bin = Binär-Information			(Anzahl vom Typ gemäß Zeichen 6)		**X** = Bit	(Unterstrich)	**R** Empfangs-Funktion
Ana = Analogwert			Ausnahme: Die Anzahl der Daten bei "Dat256D_x" belegt drei Spalten.		**B** = Byte		**FS *** Senden (Forced Send)
Cnt = Zählwert					**W** = Wort		**FR *** Empfangen (Forced Receive)
Cmd = Befehl					**D** = Doppelwort		
Set = Sollwert, Parameter					**R** = Real		
Par = Parameter							
Dat = Daten (beliebige Mischung von Informationstypen)							

***** Bausteine mit den Suffixen "_FS" und "_FR" stehen nur für die S7-1500 zur Verfügung.

Übersicht der Datenpunkt-Typicals

Die nachfolgende Tabelle gibt eine Übersicht der Datenpunkt-Typicals.

Tabelle 1-142 Übersicht der Datenpunkt-Typicals

Datenformat	Symbolischer Baustein-Name	Funktion
Binärwert-Typicals		
Byte	Bin04B_S	4 Byte Binärwerte senden
Byte	Bin04B_R	4 Byte Binärwerte empfangen
Analogwert-Typicals		
Int	Ana04W_S	4 Analogwerte senden
Int	Ana04W_R	4 Analogwerte empfangen
Real	Ana04R_S	4 Analogwerte als Gleitpunktzahl senden (nur CPU 1500)
Real	Ana04R_R	4 Analogwerte als Gleitpunktzahl empfangen (nur CPU 1500)
Zählwert-Typicals		
UInt, Word *	Cnt01D_S	1 Zählwert senden
UInt, Word *	Cnt01D_R	1 Zählwert empfangen
UInt, Word *	Cnt04D_S	4 Zählwerte senden
UInt, Word *	Cnt04D_R	4 Zählwerte empfangen
Befehls-Typicals		
Byte, USInt *	Cmd01B_S	1 Byte Befehle (1-aus-8) senden
Byte, USInt *	Cmd01B_FS	1 Byte Befehle senden (ohne 1-aus-8-Kontrolle)
Byte, USInt *	Cmd01B_R	1 Byte Befehle (1-aus-8) empfangen
Byte, USInt *	Cmd01B_FR	1 Byte Befehle empfangen (ohne 1-aus-8-Kontrolle)
Sollwert- und Parameter-Typicals		
Word	Set01W_S / Set01W_FS	1 Sollwert senden und aktuellen vor-Ort-Sollwert empfangen
Word	Set01W_R	1 Sollwert empfangen und aktuellen vor-Ort-Sollwert senden
ARRAY [1...12] of DInt / UDInt / DWord / Real *	Par12D_S / Par12D_FS	Max. 12 Doppelworte mit Parametern senden und aktuelle vor-Ort-Parameter empfangen
ARRAY [1...12] of DInt / UDInt / DWord / Real *	Par12D_R	Max. 12 Doppelworte mit Parametern empfangen und aktuelle vor-Ort-Parameter senden
Typicals für variable Datentypen und Mengen		
ARRAY [1...12] of DInt / UDInt / DWord / Real *	Dat12D_S	Max. 12 Doppelworte mit beliebigen Daten senden
ARRAY [1...12] of DInt / UDInt / DWord / Real *	Dat12D_R	Max. 12 Doppelworte mit beliebigen Daten empfangen

Datenformat	Symbolischer Baustein-Name	Funktion
DWord	Dat256D_S	Max. 256 Doppelworte mit beliebigen Daten senden
DWord	Dat256D_R	Max. 256 Doppelworte mit beliebigen Daten empfangen

***** USInt, UInt und UDInt werden von S7-300/400 nicht unterstützt.

Wiederkehrende Parameter

Beschreibung der Typical-Parameter

Bei der nachfolgenden Beschreibung werden identische Datenpunkt-Typicals, die sich nur in der Anzahl der zu bearbeitenden Datenpunkte unterscheiden, gemeinsam beschrieben.

In den Tabellen der Datenpunkt-Typicals finden Sie zu den einzelnen Parametern folgende Angaben:

- Parameter
  Name des Parameters
- Deklaration
  Parameterart
  - INPUT
    Eingangsparameter
  - OUTPUT
    Ausgangsparameter
  - IN_OUT
    Durchgangsparameter
- Datentyp
  Unterstützter Datentyp für diesen Parameter
- Wertebereich
- Vorbelegung
  Voreingestellter Wert des Parameters
  Wenn Sie einzelne Parameter eines Datenpunkt-Typicals nicht parametrieren, wird der vorbelegte Wert verwendet.
- Erläuterung
  Beschreibung der Funktion und der Typical-spezifischen Eigenschaften des Parameters

Von vielen Typicals verwendete Parameter

Die folgenden Parameter werden von vielen Typicals der TD7onCPU-Bibliothek verwendet. Diese Parameter werden einmal an dieser Stelle beschrieben und in den nachfolgenden Kapiteln der einzelnen Datenpunkt-Typicals nicht jedes Mal wiederholt.

In Abhängigkeit der Verwendung der Typicals werden manche Parameter unterschiedlich parametriert. Beachten Sie die nachfolgend beschriebene Typical-Verwendung.

PartnerNo

Parameter: **PartnerNo**
Deklaration: INPUT
Datentyp: INT
Wertebereich: 0 / 1 ... 32000
Vorbelegung: 0
Erläuterung: Teilnehmernummer des Partners

Anzugeben ist die Teilnehmernummer des Partners, mit dem der Baustein kommuniziert.

- Bei einem Prozess-Typical ist dies normalerweise die Teilnehmernummer der Zentralestation oder der Applikation der Leitstelle (bspw. ST7cc).
- Bei einem Operator-Typical ist dies normalerweise die Teilnehmernummer einer Station.

Auswirkungen des Werts 0 (null) auf verschiedene Typical-Klassen

- **Sendende Prozess-Typicals**
  (Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Dat12D_S, Dat256D_S)
  Beim Wert 0 werden die Daten an alle Teilnehmer gesendet, zu denen eine ST7-Verbindung projektiert ist. Der Parameter "PartnerObjectNo" wird in diesem Fall vom Prozess-Typical automatisch mit dem Wert Null gesendet.
  Wenn PartnerNo nicht in der Buchführung (DB BasicData) gefunden wird, dann wird ein entsprechender Eintrag in den Diagnosepuffer eingetragen (Ereignis-ID B101). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
  Hinweise
  - Bei fehlender "PartnerObjectNo" muss in der Partner-CPU in jedem Fall eine Liste vorhanden sein, aus der die fehlende Objekt-Nr. hervorgeht, siehe FC ListGenerator1500/300/400 (Seite 2028).
  - Einsatz des Bausteins in einer Knotenstation
    Wenn die CPU der Knotenstation sowohl Verbindungen zu übergeordneten Teilnehmern als auch zu unterlagerten Stationen unterhält, wird ein Daten-Telegramm mit "PartnerNo" = 0 an alle Teilnehmer in Richtung Zentrale und in Richtung Stationen übertragen.
- **Empfangende Operator-Typicals**
  (Bin04B_R, Ana04W_R, Cnt01D_R/Cnt04D_R, Dat12D_R, Dat256D_R)
  Der Wert 0 ist nicht zulässig!
  Bei PartnerNo < 1 bzw. > 32000 wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B100).
  Wenn der parametrierte Wert zulässig und korrekt ist, aber "PartnerNo" nicht in der Buchführung (DB BasicData) gefunden wird, dann wird ebenfalls ein Eintrag in den Diagnosepuffer eingetragen (Ereignis-ID B101). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
  Wenn die CPU ein Daten-Telegramm für dieses Typical empfängt, wird überprüft, ob die in dem Daten-Telegramm stehende Quell-Teilnehmer-Nr. identisch ist mit der hier parametrierten "PartnerNo". Wenn die beiden Teilnehmernummern unterschiedlich sind, dann wird die empfangene Information verworfen und eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B130).

- **Sendende Operator-Typicals**
  (Cmd01B_S, Set01W_S, Par12D_S)
  Der Wert 0 ist nicht zulässig!
  Bei "PartnerNo" < 1 bzw. > 32000 wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B100).
  Wenn der parametrierte Wert zulässig und korrekt ist, aber "PartnerNo" nicht in der Buchführung (DB BasicData) gefunden wird, dann wird ebenfalls ein Eintrag in den Diagnosepuffer eingetragen (Ereignis-ID B101). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
- **Empfangende Prozess-Typicals**
  (Cmd01B_R, Set01W_R, Par12D_R)
  Parametrieren Sie den Wert 0, wenn das Typical von mehr als einem Partner Daten empfangen soll, z. B. wenn es Daten von mehreren Leitstellen empfangen soll.
  Wenn die CPU Daten für dieses Typical empfängt und "PartnerNo" > 0 ist, dann wird überprüft, ob die in dem Daten-Telegramm stehende Quell-Teilnehmer-Nr. identisch ist mit der hier parametrierten "PartnerNo". Wenn beide unterschiedlich sind, wird die empfangene Information verworfen und eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B130). Diese Prüfung unterbleibt, wenn "PartnerNo" = 0 ist. Unabhängig vom Absender wird jedes an das Typical adressierte Daten-Telegramm an das Typical weitergegeben.
  Wenn "PartnerNo" > 0 ist und diese Nummer nicht in der Buchführung (DB-BasicData) gefunden wird, dann wird ein Eintrag in den Diagnosepuffer eingetragen (Ereignis-ID B101). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
  Hinweise
  - Bei "PartnerNo" = 0 muss sichergestellt sein, dass jeder Partner die Daten mit einer vollständigen Zieladresse sendet (Ziel-Teilnehmer-Nr. und Ziel-Objekt-Nr.).
  - Einsatz des Bausteins in einer Knotenstation
    Wenn die CPU der Knotenstation sowohl Verbindungen zu übergeordneten Teilnehmern als auch zu unterlagerten Stationen unterhält, wird ein Daten-Telegramm mit PartnerNo = 0 an alle Teilnehmer in Richtung Zentrale und in Richtung Stationen übertragen.

PartnerObjectNo

Parameter:	**PartnerObjectNo**
Deklaration:	INPUT
Datentyp:	INT
Wertebereich:	0 / 1 ... 32000
Vorbelegung:	0
Erläuterung:	Objekt-Nr. beim Partner Nummer des Objekts (= DB-Nummer) beim Partner, mit dem der Baustein kommuniziert.

Auswirkungen des Werts 0 (null) auf verschiedene Typical-Klassen

- Sendende Prozess-Typicals
  (Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Dat12D_S, Dat256D_S)
  Die Parametrierung von 0 ist sinnvoll, falls beim vorangegangenen Parameter PartnerNo = 0 parametriert wurde. Bei fehlender "PartnerObjectNo" muss in der Partner-CPU in jedem Fall eine Liste vorhanden sein, aus der die fehlende Objekt-Nr. hervorgeht (siehe FC-ListGenerator).
  Wenn der Partner eine ST7cc-Leitstelle ist, dann kann an diesem Baustein die Angabe der "PartnerObjectNo" entfallen, da es bei ST7cc keine DBs als Ziel-Objekte gibt. ST7cc dekodiert seine Daten ausschließlich anhand der im Daten-Telegramm enthaltenen Quelladresse.
- Empfangende Operator-Typicals
  (Bin04B_R, Ana04W_R, Cnt01D_R/Cnt04D_R, Dat12D_R, Dat256D_R)
  Der Wert 0 ist nicht zulässig!
  Bei falscher Parametrierung (< 1 bzw. > 32000) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B102). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
  Wenn die CPU Daten für das hier parametrierte Objekt empfängt, wird überprüft, ob die in dem Daten-Telegramm stehende Quell-Objekt-Nr. identisch ist mit der hier parametrierten "PartnerObjectNo". Wenn beide unterschiedlich sind, dann wird die empfangene Information verworfen. Eine Fehlermeldung wird in den Diagnosepuffer eingetragen (Ereignis-ID B131).
- Sendende Operator-Typicals
  (Cmd01B_S, Set01W_S, Par12D_S)
  Der Wert 0 ist nicht zulässig!
  Bei falscher Parametrierung (< 1 bzw. > 32000) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B102). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.
- Empfangende Prozess-Typicals
  (Cmd01B_R, Set01W_R, Par12D_R)
  Die Parametrierung von 0 ist in den folgenden Fällen erforderlich:
  - Der Partner ist keine S7-CPU, d. h. es gibt keine DB-Nummer als Objekt. Dies ist z. B. der Fall, wenn der Partner eine ST7cc-Leitstelle ist.
  - Es gibt mehr als einen Partner (PartnerNo = 0), von dem das Typical Daten empfangen soll. Bei diesen Partnern werden die zugehörigen Objekte dann im Allgemeinen unterschiedliche Nummern haben, d. h. es kann hier keine eindeutige Nummer angegeben werden.

  Wenn die CPU Daten für das hier parametrierte Objekt empfängt und "PartnerObjectNo" ist > 0, dann wird überprüft, ob die in dem Daten-Telegramm stehende Quell-Objekt-Nr. identisch ist mit der hier parametrierten "PartnerObjectNo". Wenn beide unterschiedlich sind, dann wird die empfangene Information verworfen. Eine Fehlermeldung wird in den Diagnosepuffer eingetragen (Ereignis-ID B131).
  Diese Prüfung unterbleibt, wenn "PartnerObjectNo" = 0 ist. Unabhängig vom Absender-Objekt wird jedes an das Objekt adressierte Daten-Telegramm an das empfangende Objekt weitergegeben.

Enabled

Parameter:	**Enabled**	
Deklaration:	INPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	TRUE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Freigabe der Bausteinbearbeitung

- Bei freigegebener Bearbeitung werden sämtliche Funktionen des Bausteins durchlaufen.
- Bei nicht freigegebener Bearbeitung unterscheidet sich das Verhalten:

Nicht freigegebene Bearbeitung

- **Bei sendenden Prozess-Typicals**
  (Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S,
  Set01W_R, Par12D_R,
  Dat12D_S, Dat256D_S)
  Bei nicht freigegebener Bearbeitung ist der Baustein nur noch auf organisatorischer Ebene kommunikationsfähig, d. h. Org.-Telegramme können gesendet und empfangen werden.
  Beachten Sie: Eine Abfrage von einem Operator-Typical wird zwar beantwortet, das Antwort-Telegramm beinhaltet aber die zum Zeitpunkt der Sperre gültigen Daten.
  Hinweis
  Das hier beschriebene Verhalten gilt nicht für Cmd01B_R, siehe Befehls-Typical Cmd01B_R / Cmd01B_FR (Seite 1974)!
- **Bei empfangenden Operator-Typicals**
  (Bin04B_R, Ana04W_R, Cnt01D_R/Cnt04D_R,
  Set01W_S, Par12D_S,
  Dat12D_R, Dat256D_R)
  Bei nicht freigegebener Bearbeitung ist der Baustein nur noch auf organisatorischer Ebene kommunikationsfähig, d. h. Org.-Telegramme können gesendet und empfangen werden.
  Beachten Sie: Eine Abfrage kann noch gesendet und die Antwort empfangen werden, die empfangene Information wird jedoch nicht an die Ausgänge ausgegeben. Die relevanten Ausgänge finden Sie in der Beschreibung der jeweiligen Datenpunkt-Typicals.
  Hinweis
  Das hier beschriebene Verhalten gilt nicht für Cmd01B_S, siehe Befehls-Typical Cmd01B_S / Cmd01B_FS (Seite 1971)!

ImageMemory

Parameter:	**ImageMemory**
Deklaration:	INPUT

Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	TRUE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7
Verwendung in Typicals:	Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Set01W_R, Par12D_R, Dat12D_S	
Erläuterung:	Abbildspeicherprinzip für die spontane Datenübertragung	

- TRUE
  Die Daten werden nach dem Abbildspeicherprinzip übertragen.
  Das Abbildspeicherprinzip sorgt für eine speicherschonende Ablage von Daten-Telegrammen und führt auf der WAN-Strecke zu einem möglichst geringen Datenverkehr. Die Vorbelegung TRUE ist in den meisten Fällen die richtige Wahl.
- FALSE
  Die Daten werden nach dem Sendepufferprinzip übertragen.
  Der Sendepufferprinzip wird nur bei Datenpunkten benötigt, deren einzelne Daten-Änderungen gespeichert und zum Partner übertragen werden sollen, beispielsweise Alarme mit Zeitstempel.

Conditional

Parameter:	**Conditional**	
Deklaration:	INPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	TRUE	
Adressbereich:	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7
Verwendung in Typicals:	Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Set01W_R, Par12D_R, Dat12D_S	
Erläuterung:	Bedingt spontane Datenübertragung Informationen finden Sie nachfolgend beim Parameter "Unconditional".	

Unconditional

Parameter:	**Unconditional**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE
Vorbelegung:	FALSE

Adressbereich:	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7
Verwendung in Typicals:	Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Set01W_R, Par12D_R, Dat12D_S, Dat256D_S	

Erläuterung: Unbedingt spontane Datenübertragung

Mit den beiden Parametern "Conditional" und "Unconditional" bestimmen Sie, ob ein Daten-Telegramm bei Änderung des Werts sofort (unbedingt spontan) oder erst zu einem späteren Zeitpunkt (bedingt spontan) vom Modul übertragen werden soll.

Die beiden Parameter sind folgendermaßen zu parametrieren:

- **Bedingt spontane Übertragung (nicht unbedingt sofort)**
  - Conditional = TRUE
  - Unconditional = FALSE
- **Unbedingt spontane Übertragung (sofort)**
  - Conditional = FALSE
  - Unconditional = TRUE

Die Vorbelegung der beiden Parameter ist so gewählt, dass ein Daten-Telegramm nicht sofort übertragen wird.

Die Entscheidung, ob ein Daten-Telegramm sofort oder später übertragen werden soll, stellt sich nur bei Wählnetzen.

Bei einem Wählnetz müssen Sie fallweise entscheiden, ob bei Wertänderung des Datenpunkts die sofortige Übertragung gewünscht ist und damit sofort eine Verbindung aufgebaut wird. Dies kann beispielsweise bei Datenpunkten mit Alarmen gewünscht sein.

Bei einer Standleitung wird immer sofort übertragen, auch wenn die Kombination von "Conditional" und "Unconditional" auf "nicht sofort" eingestellt ist. Bei Standleitungen brauchen Sie keine Änderungen der beiden Parameter vorzunehmen.

Permanent

Hinweis

Parameter irrelevant

Der Parameter (INPUT, Bool) taucht nur bei Bausteinen für S7-300/400 auf.

Er wird funktional bei S7-300/400 nicht mehr unterstützt, der Wert ist immer = FALSE.

TimeStamp

Parameter:	**TimeStamp**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE

Vorbelegung:	FALSE
Verwendung in Typicals:	Bin04B_S, Ana04W_S, Cnt01D_S/Cnt04D_S, Dat12D_S, Dat256D_S, Set01W_R, Par12D_R
Erläuterung:	Zeitstempel

- TRUE
  Das Daten-Telegramm wird mit Zeitstempel übertragen.
  Voraussetzung ist, dass in der CPU eine von der lokalen TIM bereitgestellte Zeit verfügbar ist. Für weitere Details dazu siehe die Beschreibung zum FC TimeTask.
  Hinweis
  Beachten Sie bei Ana04W_S die Abhängigkeit des Zeitstempels vom Parameter "MeanValueGeneration", siehe Analogwert-Typical Ana04W_S (Seite 1959).
- FALSE
  Das Daten-Telegramm wird ohne Zeitstempel übertragen.

Zum Format der Zeitstempel siehe Kapitel Zeitstempel (Seite 1953).

Hinweis

Reinitialisierung

Dieser Parameter erfordert eine Reinitialisierung der CPU.

Empfehlung:

Ändern Sie den Parameter zur Laufzeit oder nach dem Neustart nicht mehr!

NewData

Parameter:	**NewData**	
Deklaration:	OUTPUT	
Datentyp:	BOOL (Dat256D_R : DWORD)	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Verwendung in Typicals: Bin04B_R, Ana04W_R, Cnt01D_R/Cnt04D_R, Cmd01B_R, Set01W_S, Set01W_R, Par12D_S, Par12D_R, Dat12D_R, Dat256D_R

Bei Dat256D_R belegt der Ausgang ein Doppelwort mit je einem Status pro Segment.

Erläuterung: Neue Daten empfangen

Der Ausgang NewData ist für anwenderspezifische Weiterverarbeitungen vorgesehen, z. B. um gezielt auf den Empfang von neuen Daten zu reagieren.

Jedes Mal, wenn der Baustein neue Daten empfangen und an die jeweils Typical-spezifischen Ausgänge ausgegeben hat, wird NewData für einen OB1-Zyklus auf TRUE gesetzt.

Die spezifischen Ausgänge finden Sie in der Beschreibung der einzelnen Datenpunkt-Typicals.

Bei den Datenpunkt-Typicals Set01W_R und Par12D_R wird "NewData" auch für einen OB1-Zyklus auf TRUE gesetzt, wenn im Zustand Local = 1 ein neuer Wert lokal eingegeben wird.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Zeitstempel

Format der SINAUT-Zeitstempel

Bei vielen Datenpunkt-Typicals können Sie über den Parameter "TimeStamp" bestimmen, ob die Daten des Objekts mit Zeitstempel übertragen werden soll.

Bei den empfangenden Datenpunkt-Typicals gibt es aber keinen Ausgangsparameter, über den Sie den empfangen Zeitstempel ausgeben können. Der Zeitstempel wird lediglich in dem Instanz-DB, den Sie beim Aufruf des betreffenden Empfangs-Typicals angegeben haben, abgespeichert. Zur Weiterverarbeitung des Zeitstempels müssen die Daten über das Anwenderprogramm aus dem DB ausgelesen werden.

Der Zeitstempel wird in zwei Doppelworten abgelegt, die bei allen Objekt-DBs den gleichen Namen haben:

Name des Doppelworts	Inhalt
RecTimeStamp_1	Jahr, Monat, Tag und Stunde
RecTimeStamp_2	Minute, Sekunde, Millisekunde und Zeit-Status

Mit Ausnahme des Halb-Bytes mit dem Zeit-Status sind Datum und Uhrzeit im BCD-Format kodiert.

Tabelle 1-143 Belegung der Struktur der Zeitstempel

Name des Doppelwortes	Byte-Nr.	Inhalt	
		High nibble	Low nibble
RecTimeStamp_1	0	Jahr * 10	Jahr * 1
	1	Monat * 10	Monat * 1
	2	Tag * 10	Tag * 1
	3	Stunde * 10	Stunde * 1

Name des Doppelwortes	Byte-Nr.	Inhalt	
		High nibble	Low nibble
RecTimeStamp_2	0	Minute * 10	Minute * 1
	1	Sekunde * 10	Sekunde * 1
	2	Millisekunde * 100	Millisekunde * 10
	3	Millisekunde * 1	Zeit-Status

Tabelle 1-144 Bit-Belegung des Halb-Byte "Zeit-Status" (Low nibble von Byte 3 von RecTimeStamp_2)

Bit-Nr.	Wert	Bedeutung
0	0	Zeit ist ungültig
	1	Zeit ist gültig
1	0	Winterzeit (Normalzeit)
	1	Sommerzeit
2		Nicht belegt
3		Nicht belegt

Je nach Typical belegen die genannten Zeit-Doppelworte unterschiedliche Adressen. Sehen Sie im Instanz-DB oder im Deklarationskopf des FB nach, unter welcher absoluter Adresse die beiden Doppelworte ausgelesen werden können.

Komfortabler geht es, wenn Sie den Instanz-DBs einen symbolischen Namen geben. Für das Auslesen können sie dann die symbolische Adressierung nutzen. In diesem Fall müssen Sie sich um die tatsächlichen absoluten Adressen keine Gedanken machen. Diese werden automatisch von STEP 7 eingesetzt. Das nachfolgende Beispiel erläutert diese Vorgehensweise.

Beispiel

Symbolische Name des Instanz-DB: `ObjectDB27`

Das STEP 7-Programm zum Auslesen von Datum und Uhrzeit und Ablage in DB20 ab Datenbyte 100 könnte dann, in AWL programmiert, wie folgt aussehen:

```
L "ObjectDB27".RecTimeStamp_1
T DB 20.DBD 100
L "ObjectDB27".RecTimeStamp_2
T DB 20.DBD 104
```

Analogwert-Typical Ana04R_S

Gültigkeit

Nur S7-1500

Funktion

4 Analogwerte als 32-Bit-Gleitpunktzahl senden

Ana04R_S überträgt die 4 Analogwerte als Momentanwerte. Zum Zeitpunkt der Übertragung wird der aktuell anstehende Analogwert erfasst und zum Partner übertragen.

Hinweis

Gemeinsame Bearbeitung der vier Analogwerte

Die Verarbeitungsparameter wie Schwellenwert, Glättungsfaktor usw. sind nur einmal am Typical vorhanden. Diese Parameter gelten für alle 4 Analogwerte gemeinsam, d. h. eine individuelle Einstellung der Parameter pro Analogwert ist nicht möglich. Pro Typical sollten deshalb nur solche Analogwerte erfasst werden, die identisch bearbeitet werden sollen.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional
TimeStamp

Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **ThresholdIntegration**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Schwellenwertbearbeitung nach dem Integrationsprinzip

Bei diesem Parameter kann angegeben werden, ob bei der Schwellenwertbearbeitung das Integrationsprinzip angewendet werden soll.

In der Vorbelegung FALSE wird der Schwellenwert ohne Integration berechnet. Sowohl auf der Fernleitung als auch im lokalen Verkehr zwischen CPU und TIM ist in diesem Fall mit einem geringeren Datenaufkommen zu rechnen.

Parameter: **ZeroLimitation**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE

Vorbelegung: TRUE

Erläuterung: Nullwertbegrenzung

Bei aktiviertem Parameter werden negative Werte unterdrückt und durch den Wert 0 ersetzt.

Parameter: **TriggerInput**

Deklaration: INPUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Auslöse-Eingang

Über den Flankenwechsel 0 → 1 des Eingangs "TriggerInput" kann die getriggerte Übertragung des Analogwert-Telegramms zu einem gewünschten Zeitpunkt ausgelöst werden.

Beispiel:
Eine zeitgesteuerte Analogwertübertragung mit Zeitstempel zur Versorgung eines Analogwertarchivs in der Leitstelle.

Achten Sie darauf, den Parameter "ImageMemory" auf FALSE einzustellen, damit diese Daten-Telegramme mit Zeitstempel beim Speichern auf der Stations-TIM nicht überschrieben werden.

Für die zeitgesteuerte Auslösung einer Übertragung über "TriggerInput" bietet sich die Anwendung des Bausteins FC Trigger an.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen. Die Übertragung sollte dann basierend auf den Schwellenwert-Parametern "ThresholdValue" und "ThresholdIntegration" ausgelöst werden.

Im eigentlichen Sinne wird über "TriggerInput" nur indirekt eine Übertragung ausgelöst. Beim Flankenwechsel 0 → 1 an "TriggerInput" wird das Daten-Telegramm mit seinen aktuellen Werten zusammengestellt und an die lokale TIM übergeben. Diese ist dann für die eigentliche Übertragung an den Partner zuständig. Bei Standleitungen oder Funknetzen wird umgehend übertragen. Bei einer Wählverbindung kann es sein, dass das Daten-Telegramm zunächst auf der TIM gespeichert wird und erst zu einem späteren Zeitpunkt zur Übertragung gelangt. Grund kann beispielsweise sein, dass das Daten-Telegramm als "bedingt spontan" gekennzeichnet ist, siehe Parameter "Conditional".

Parameter: **AnalogInput_1 ... _4**

Deklaration: INPUT

Datentyp: REAL

Wertebereich: Siehe Adressbereich

Vorbelegung: 0.0

Adressbereich:	Merker-Adresse	MD0 ... MDn
	Daten-Adresse	LD0 ... LDn DBm.DBD0 ... DBDn

Erläuterung: Analog-Eingangswort

Für jeden Analogwert, der im Daten-Telegramm zu übertragen ist, kann einzeln angegeben werden, woher diese analoge Information vom FB zu entnehmen ist. Adessen eines Datenbausteins und des Merkerbereichs können dabei beliebig gemischt werden.

Parameter, die Sie nicht benötigen, lassen Sie einfach offen. Für diese Analogeingänge wird im Daten-Telegramm die 0 übertragen.

Parameter: **SamplingPeriod**

Deklaration: INPUT

Datentyp: INT

Wertebereich: 0 ... 32767 [ms]

Vorbelegung: 500

Erläuterung: Erfassungsintervall für die Analogeingänge in Millisekunden

Das Erfassungsintervall wird für folgende Parameter benötigt:

- Schwellenwertbildung nach dem Integrationsprinzip (ThresholdIntegration)
- Glättung des Analogeingangswertes (SmoothingFactor)

Der Wert muss in jedem Fall so groß gewählt werden, dass sichergestellt ist, dass innerhalb der Verschlüsselungszeit des Analogeingangs ein neuer Wert erfasst wurde. Es ist mindestens die Verschlüsselungszeit der eingesetzten Analog-Eingabebaugruppe bei der jeweils gewählten Auflösung (8 ... 15 Bit) anzugeben.

Weiterhin muss der Wert so großzügig gewählt, dass Analogwerte auch bei der höchsten Auflösung und bei den Analogbaugruppen mit der größten Anzahl an Eingängen erfasst werden.

Wenn für "SamplingPeriod" eine zu kurze Zeit angegeben wird, dann kann es zu einem Überlauf des intern geführten Summenzählers führen.

Parameter: **ThresholdValue**

Deklaration: INPUT

Datentyp: REAL

Wertebereich: +1.175495e-38 ... +3.402823e+38

Vorbelegung: 1.0

Erläuterung: Schwellenwert

Ohne Parametrierung wird der vorbelegte Wert 1.0 angewendet.

Besonderheit bei "ThresholdValue" = 0: Es wird keine Änderungskontrolle auf Basis des Schwellenwerts angewendet. Das Analogwert-Telegramm wird nur noch in den folgenden Fällen übertragen:

- Anstoß über den Eingang "TriggerInput", also typischerweise eine zeit- oder ereignisgesteuerte Auslösung.
- Generalabfrage der Station oder Einzelabfrage des Daten-Telegramms

Parameter: **SmoothingFactor**
Deklaration: INPUT
Datentyp: REAL
Wertebereich: 1.0 (keine Glättung)
4.0 (schwache Glättung)
32.0 (mittlere Glättung)
64.0 (starke Glättung)
Vorbelegung: 4.0
Erläuterung: Glättungsfaktor

Mit Hilfe des Glättungsfaktors können schnell schwankende Analogwerte je nach Faktor mehr oder weniger stark beruhigt werden. Dadurch kann das Schwellenwertband (ThresholdValue) u. U. kleiner gewählt werden.

Die Glättungsfaktoren sind identisch mit den Glättungsfaktoren, die für einige S7-Analogeingabebaugruppen parametrierbar sind. Die Glättung wird nach gleicher Formel wie auf der Eingabebaugruppe berechnet:

$$y_n = \frac{x_n + (k-1)\ y_{n-1}}{k}$$

y_n = Geglätteter Wert im aktuellen Zyklus n
y_{n-1} = Geglätteter Wert im vorhergehenden Zyklus n-1
x_n = Erfasster Wert im aktuellen Zyklus n
k = Glättungsfaktor

Analogwert-Typical Ana04R_R

Gültigkeit

Nur S7-1500

Funktion

4 Analogwerte als 32-Bit-Gleitpunktzahl empfangen

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **AnalogOutput_1 … _4**
Deklaration: OUTPUT

Datentyp	REAL	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0.0	
Adressbereich:	Merker-Adresse	MD0 ... MDn
	Daten-Adresse	LD0 ... LDn DBm.DBD0 ... DBDn
Erläuterung:	Es kann einzeln parametriert werden, wohin die vom FB empfangenen Analogwerte auszugeben sind. Adressen eines Datenbausteins und im Merkerbereich können dabei beliebig gemischt werden. Parameter, die Sie nicht benötigen, lassen Sie einfach offen.	
Parameter:	**NewData** Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).	
Typical-spezifisches Verhalten:	Jedes Mal, wenn der FB neue Daten empfangen und an die Ausgänge "AnalogOutput_1" bis "AnalogOutput_4" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt.	

Analogwert-Typical Ana04W_S

Funktion

4 Analogwerte als 16-Bit-Werte senden

Ana04W_S überträgt die 4 Analogwerte alternativ:

- Als Momentanwerte
  Zum Zeitpunkt der Übertragung wird der aktuell anstehende Analogwert erfasst und zum Partner übertragen.
- Als Mittelwerte
  Der anstehende Analogwert wird in einem parametrierbaren Zeitraster aufsummiert. Zum Zeitpunkt der Übertragung wird aus dem aufsummierten Wert ein Mittelwert gebildet und zum Partner übertragen.

Hinweis

Gemeinsame Bearbeitung der vier Analogwerte

Die Verarbeitungsparameter wie Schwellenwert, Glättungsfaktor usw. sind nur einmal am Typical vorhanden. Diese Parameter gelten für alle 4 Analogwerte gemeinsam, d. h. eine individuelle Einstellung der Parameter pro Analogwert ist nicht möglich. Pro Typical sollten deshalb nur solche Analogwerte erfasst werden, die identisch bearbeitet werden sollen.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional

Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **TimeStamp**

Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Typical-spezifische Funktionen: In der Einstellung TRUE hängt der Zeitstempel von der Einstellung des Parameters "MeanValueGeneration" ab:

- MeanValueGeneration = FALSE
  Im Daten-Telegramm werden Momentanwerte übertragen.
  Der Zeitstempel im Daten-Telegramm ist identisch mit dem Zeitpunkt der Erfassung der im Daten-Telegramm enthaltenen Momentanwerte.
- MeanValueGeneration = TRUE
  Das Daten-Telegramm enthält Mittelwerte.
  Der Zeitstempel ist identisch mit dem Zeitpunkt, zu dem die Mittelungsperiode abgeschlossen wurde.
  Der Anfangszeitpunkt der Mittelungsperiode ist nicht im Daten-Telegramm enthalten. Diese ist aber identisch mit dem Zeitstempel des zuvor übertragenen Mittelwert-Telegramms.

Parameter: **ThresholdIntegration**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Schwellenwertbearbeitung nach dem Integrationsprinzip

Bei diesem Parameter kann angegeben werden, ob bei der Schwellenwertbearbeitung das Integrationsprinzip angewendet werden soll.

In der Vorbelegung FALSE wird der Schwellenwert ohne Integration berechnet. Sowohl auf der Fernleitung als auch im lokalen Verkehr zwischen CPU und TIM ist in diesem Fall mit einem geringeren Datenaufkommen zu rechnen.

Wenn "MeanValueGeneration" = TRUE (Analogwerte werden als Mittelwerte übertragen), dann ist der Parameter "ThresholdIntegration" ohne Bedeutung.

Parameter: **ZeroLimitation**
Deklaration: INPUT
Datentyp: BOOL

Wertebereich: TRUE / FALSE
Vorbelegung: TRUE
Erläuterung: Nullwertbegrenzung

Bei aktiviertem Parameter werden negative Werte unterdrückt und durch den Wert 0 ersetzt.

Parameter: **TriggerInput**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Auslöse-Eingang

Über den Flankenwechsel 0 → 1 des Eingangs "TriggerInput" kann die getriggerte Übertragung des Analogwert-Telegramms zu einem gewünschten Zeitpunkt ausgelöst werden.

Beispiel:
Eine zeitgesteuerte Analogwertübertragung mit Zeitstempel zur Versorgung eines Analogwertarchivs in der Leitstelle.

Achten Sie darauf, den Parameter "ImageMemory" auf FALSE einzustellen, damit diese Daten-Telegramme mit Zeitstempel beim Speichern auf der Stations-TIM nicht überschrieben werden.

Falls der Baustein Mittelwerte bildet, wird über den Eingang "TriggerInput" die Dauer der Mittelungsperiode bestimmt. Bei jeder Übertragung, die über diesen Eingang angestoßen wird, wird die laufende Periode abgeschlossen und die neue angefangen. Der zeitliche Abstand zwischen dem zweimaligen Triggern eines Daten-Telegramms bestimmt die Dauer der Mittelungsperiode.

Für die zeitgesteuerte Auslösung einer Übertragung über "TriggerInput" bietet sich die Anwendung des Bausteins FC Trigger an.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen. Die Übertragung sollte dann basierend auf den Schwellenwert-Parametern "ThresholdValue" und "ThresholdIntegration" ausgelöst werden.

Im eigentlichen Sinne wird über "TriggerInput" nur indirekt eine Übertragung ausgelöst. Beim Flankenwechsel 0 → 1 an "TriggerInput" wird das Daten-Telegramm mit seinen aktuellen Werten/Mittelwerten zusammengestellt und an die lokale TIM übergeben. Diese ist dann für die eigentliche Übertragung an den Partner zuständig. Bei Standleitungen oder Funknetzen wird umgehend übertragen. Bei einer Wählverbindung kann es sein, dass das Daten-Telegramm zunächst auf der TIM gespeichert wird und erst zu einem späteren Zeitpunkt zur Übertragung gelangt. Grund kann beispielsweise sein, dass das Daten-Telegramm als "bedingt spontan" gekennzeichnet ist, siehe Parameter "Conditional".

Parameter:	**MeanValueGeneration**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE
Vorbelegung:	FALSE

Erläuterung: Mittelwertbildung

Bei aktiviertem Parameter werden die zu erfassenden Analogwerte als Mittelwerte übertragen.

Wenn die Mittelwertbildung gewählt wird, wird der aktuell anstehende Analogwert zyklisch erfasst und aufsummiert. Der Erfassungszyklus ist abhängig vom Parameter "SamplingPeriod" (z.B. 500 ms, siehe auch die Erläuterung zu diesem Parameter). Aus den aufsummierten Werten wird der Mittelwert gebildet, sobald über den Eingang "TriggerInput" eine Übertragung ausgelöst wird. Danach wird die Aufsummierung zur Bildung des nächsten Mittelwertes neu gestartet.

Der Mittelwert wird auch dann gebildet, wenn die Übertragung des Analogwert-Telegramms durch eine General- oder Einzelabfrage ausgelöst wird. Die Dauer der Mittelungsperiode ist dann die Zeit von der letzten Übertragung (z.B. über "TriggerInput" ausgelöst) bis zum Zeitpunkt der General- oder Einzelabfrage. Auch nach dieser Übertragung wird die Aufsummierung zur Bildung des nächsten Mittelwertes neu gestartet.

Falls der erfasste Analogwert im Über- oder Untersteuerungsbereich (7FFFH bzw. 8000H) liegt, kann dieser Wert entweder sofort für die Mittelwertbildung berücksichtigt werden oder er wird für eine bestimmte Zeit für die Mittelwertbildung ausgeblendet. Das jeweils gewünschte Verhalten wird über den Parameter "FaultSuppressionTime" bestimmt:

- **FaultSuppressionTime = 0**
  Sobald ein Wert im Über- oder Untersteuerungsbereich erfasst wird, führt dies zum sofortigen Abbruch der Mittelwertbildung. Für die laufende Mittelungsperiode wird der Wert $7FFF_H$ bzw. 8000_H als ungültiger Mittelwert gespeichert und bei der nächsten Auslösung des Analogwert-Telegramms übertragen. Danach wird eine neue Mittelwertbildung gestartet. Wenn der Analogwert dann immer noch im Über- oder Untersteuerungsbereich liegt, wird auch dieser Wert wieder sofort als ungültiger Mittelwert gespeichert und bei der nächsten Auslösung zur Übertragung gebracht.
- **FaultSuppressionTime > 0**
  Wenn der erfasste Analogwert im Über- oder Untersteuerungsbereich liegt, werden die fehlerhaften Werte nicht in die Mittelwertbildung einbezogen und zwar maximal für die Dauer von "FaultSuppressionTime". Wenn diese Zeitdauer überschritten wird, dann wird der Wert $7FFF_H$ bzw. 8000_H als ungültiger Mittelwert gespeichert und bei der nächsten Auslösung des Analogwert-Telegramms übertragen. In jeder neuen Mittelungsperiode wird identisch verfahren, d.h. fehlerhafte Werte werden wiederum für die Dauer von "FaultSuppressionTime" unterdrückt.
  Mit der Dauer von "FaultSuppressionTime" lässt sich indirekt der prozentuale Anteil an ungültigen Werten pro Mittelungsperiode bestimmen. Wenn der Mittelwert z. B. alle 15 Minuten gebildet wird und "FaultSuppressionTime" ist auf 5 Minuten eingestellt, dann wird der Mittelwert nur dann als ungültig übertragen, wenn in der laufenden Mittelungsperiode mehr als 1/3 der erfassten Analogwerte im Über- oder Untersteuerungsbereich liegen.

Parameter: **AnalogInput_1 ... _4**

Deklaration: INPUT

Datentyp:	WORD	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0 (W#16#0)	
Adressbereich:	Peripherie-Worte	PEW0 ... PEWn
	Merker-Worte	MW0 ... MWn
	Daten-Worte	LW0 ... LWn DBm.DBW0 ... n

Erläuterung: Analog-Eingangswort

Für jeden Analogwert, der im Daten-Telegramm zu übertragen ist, kann einzeln angegeben werden, woher diese analoge Information vom FB zu entnehmen ist. Peripherieworte von Analog-Eingabebaugruppen, Datenworte eines Datenbausteins und Merkerworte können dabei beliebig gemischt werden.

Parameter, die Sie nicht benötigen, lassen Sie einfach offen. Für diese Analogeingänge wird im Daten-Telegramm die 0 übertragen.

Parameter: **SamplingPeriod**

Deklaration: INPUT

Datentyp: INT

Wertebereich: 0 ... 32767 [ms]

Vorbelegung: 500

Erläuterung: Erfassungsintervall für die Analogeingänge in Millisekunden

Das Erfassungsintervall wird für folgende Parameter benötigt:

- Schwellenwertbildung nach dem Integrationsprinzip (ThresholdIntegration)
- Glättung des Analogeingangswertes (SmoothingFactor)
- Mittelwertbildung (MeanValueGeneration)

Der Wert muss in jedem Fall so groß gewählt werden, dass sichergestellt ist, dass innerhalb der Verschlüsselungszeit des Analogeingangs ein neuer Wert erfasst wurde. Es ist mindestens die Verschlüsselungszeit der eingesetzten Analog-Eingabebaugruppe bei der jeweils gewählten Auflösung (8 ... 15 Bit) anzugeben.

Weiterhin muss der Wert so großzügig gewählt, dass Analogwerte auch bei der höchsten Auflösung und bei den Analogbaugruppen mit der größten Anzahl an Eingängen erfasst werden.

Wenn Mittelwerte gebildet werden, sollte SamplingPeriod nicht kleiner als 500 ms sein. Werden Mittelwerte über sehr lange Perioden gebildet, muss die Zeit erhöht werden, und zwar:

- Mittelungsperiode 12 h: SamplingPeriod = 1000 [ms]
- Mittelungsperiode 24 h: SamplingPeriod = 2000 [ms]

Wenn für "SamplingPeriod" eine zu kurze Zeit angegeben wird, dann kann es zu einem Überlauf des intern geführten Summenzählers führen. Der maximale Wert von 2 147 483 647 eines Doppel-Integers darf nicht überschritten werden. Wenn ein Überlauffehler festgestellt wird, dann wird für die laufende Mittelungsperiode der ungültige Mittelwert von 8000_H übertragen.

Parameter: **ThresholdValue**
Deklaration: INPUT
Datentyp: INT
Wertebereich: 0 / 1 ... 32767
Vorbelegung: 270
Erläuterung: Schwellenwert

Berücksichtigen Sie bei der Angabe des Schwellenwerts den Verschlüsselungsbereich der Analogwerte. Rohwerte von S7-Analogeingaben werden immer im Bereich von 0 ... 27648 verschlüsselt (= 0 ... 100 %) oder + 27648 (= + 100%). Je nach Auflösung der Analogeingabe macht der Wert dabei Sprünge von 128 (bei 8-Bit-Auflösung) oder 1 (bei 15-Bit-Auflösung). Wenn die erfassten Analogwerte einen anderen Verschlüsselungsbereich haben, dann geben Sie einen daran orientierten Schwellenwert an.

Ohne Parametrierung wird der vorbelegte Wert von 270 angewendet. Dies entspricht ungefähr 1% vom normalen S7-Analog-Rohwertbereich.

Besonderheit bei "ThresholdValue" = 0

Es wird keine Änderungskontrolle auf Basis des Schwellenwerts angewendet. Das Analogwert-Telegramm wird nur noch in den folgenden Fällen übertragen:

- Anstoß über den Eingang "TriggerInput", also typischerweise eine zeit- oder ereignisgesteuerte Auslösung.
- Generalabfrage der Station oder Einzelabfrage des Daten-Telegramms
- In dem Fall, dass der Analogwert in den Über- oder Untersteuerungsbereich ($7FFF_H$ bzw. 8000_H) wandert (eventuell nach Ablauf der bei "FaultSuppressionTime" parametrierten Unterdrückungszeit).

Wenn die Analogwerte als Mittelwerte übertragen werden (MeanValueGeneration = TRUE), ist der Parameter "ThresholdValue" ohne Bedeutung.

Parameter: **SmoothingFactor**
Deklaration: INPUT
Datentyp: INT
Wertebereich: 1 (keine Glättung)
4 (schwache Glättung)
32 (mittlere Glättung)
64 (starke Glättung)

Vorbelegung: 1

Erläuterung: Glättungsfaktor

Wenn die Analogwerte als Mittelwerte übertragen werden (MeanValueGeneration = TRUE), ist der Parameter "SmoothingFactor" ohne Bedeutung.

Mit Hilfe des Glättungsfaktors können schnell schwankende Analogwerte je nach Faktor mehr oder weniger stark beruhigt werden. Dadurch kann das Schwellenwertband (ThresholdValue) u. U. kleiner gewählt werden.

Die Glättungsfaktoren sind identisch mit den Glättungsfaktoren, die für einige S7-Analogeingabebaugruppen parametrierbar sind. Die Glättung wird nach gleicher Formel wie auf der Eingabebaugruppe berechnet:

$$y_n = \frac{x_n + (k-1)\ y_{n-1}}{k}$$

y_n = Geglätteter Wert im aktuellen Zyklus n

y_{n-1} = Geglätteter Wert im vorhergehenden Zyklus n-1

x_n = Erfasster Wert im aktuellen Zyklus n

k = Glättungsfaktor

Parameter: **FaultSuppressionTime**

Deklaration: INPUT

Datentyp: INT

Wertebereich: 0 ... 32767

Vorbelegung: 0

Erläuterung: Störungs-Unterdrückungszeit in Sekunden

Die Übertragung eines im Über- oder Untersteuerungsbereich ($7FFF_H$ bzw. 8000_H) befindlichen Analogwerts wird für die Dauer der angegebenen Zeitspanne unterdrückt. Erst nach Ablauf dieser Zeit wird der Wert von $7FFF_H$ bzw. 8000_H übertragen, falls er noch ansteht. Fällt der Wert vor Ablauf der Zeit wieder in den Bereich unterhalb von $7FFF_H$ oder oberhalb von 8000_H zurück, wird der Wert sofort wieder normal übertragen. Die Unterdrückungszeit wird beim nächsten erfassten $7FFF_H$ bzw. 8000_H wieder mit voller Dauer gestartet.

Ein typischer Anwendungsfall ist die zeitweise Unterdrückung von Stromwerten, die beim Einschalten von leistungsstarken Pumpen und Motoren auftreten können. Der Analogeingang wird hier eventuell mehrfach übersteuert. Die Unterdrückung soll verhindern, dass diese Werte im Leitstellensystem als gestört markiert werden.

Die Unterdrückung ist abgestimmt auf Analogwerte, die als Rohwerte von den S7-Analog-Eingabebaugruppen erfasst werden. Diese Baugruppen liefern für alle Eingangsbereiche die genannten Werte für den Über- bzw. Untersteuerungsbereich (auch für Life-Zero-Eingänge). Bei bereitgestellten Fertigwerten ist eine Störungsunterdrückung nur dann möglich, wenn diese bei Über- oder Untersteuerung ebenfalls die genannten Werte von $7FFF_H$ bzw. 8000_H annehmen. Ist dies nicht der Fall, braucht der Parameter nicht ausgefüllt zu werden.

Der Parameter kann auch in Verbindung mit der Mittelwertbildung zur zeitweisen Unterdrückung der Werte $7FFF_H$ bzw. 8000_H verwendet werden (siehe Parameter MeanValueGeneration).

Wenn kein Parameter angegeben wird, gilt die Vorlegung von 0 Sekunden. Ein erfasster Wert von $7FFF_H$ bzw. 8000_H wird dann bei erstmaliger Erkennung sofort übertragen bzw. bei Mittelwertbildung als ungültiger Mittelwert für die aktuelle Mittelungsperiode festgehalten.

Analogwert-Typical Ana04W_R

Funktion

4 Analogwerte als 16-Bit-Werte empfangen

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **AnalogOutput_1 … _4**
Deklaration: OUTPUT
Datentyp WORD

Default: TRUE

Erläuterung 0 (W#16#0)

Wertebereich:		
	Peripherie-Worte	PAW0 ... PAWn
	Merker-Worte	MW0 ... MWn
	Daten-Worte	LW0 ... LWn
		DBm.DBW0 ... n

Es kann einzeln parametriert werden, wohin die vom FB empfangenen Analogwerte auszugeben sind. Peripherieworte von Analog-Ausgabebaugruppen, Datenworte eines Datenbausteins und Merkerworte können dabei beliebig gemischt werden.

Parameter, die Sie nicht benötigen, lassen Sie einfach offen.

Parameter: **NewData**

Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Typical-spezifisches Verhalten: Jedes Mal, wenn der FB neue Daten empfangen und an die Ausgänge "AnalogOutput_1" bis "AnalogOutput_4" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt.

Binärwert-Typical Bin04B_S

Funktion

4 Byte Binärinformationen senden

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional
TimeStamp

Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **InputByte_1 … _4**

Deklaration: INPUT

Datentyp: BYTE

Wertebereich: Siehe Adressbereich

Vorbelegung: 0 (B#16#0)

Adressbereich:	Eingangs-Bytes	EB0 ... EBn PEB0 ... PEBn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n

Erläuterung: Eingangs-Byte

Geben Sie den Speicherbereich (1 bis 4 Bytes) der binären Informationen an, die vom FB übertragen werden sollen.

Eingangs-Bytes aus dem Prozessabbild der Eingänge, Peripherie-Bytes von Digitaleingabebaugruppen, Daten-Bytes eines Datenbausteins und Merker-Bytes können gemischt werden.

Parameter, die Sie nicht benötigen, lassen Sie einfach offen.

Für nicht parametrierte Bytes wird der Wert 0 (Null) übertragen.

Parameter: **DisableMask**

Deklaration: INPUT

Datentyp: DWORD

Wertebereich: 0 ... 2147483647

- Als 32-Bit-Dualzahl
  2#0 ... 2#11111111_11111111_11111111_11111111
- Als 32-Bit-Hexadezimalzahl
  DW#16#0 ... DW#16#FFFF_FFFF

Vorbelegung: 0 (2#0)

Erläuterung: Sperrmaske

- Geben Sie für jeden Eingang, der gesperrt werden soll, an der entsprechenden Stelle im Bit-Muster eine 1 ein.
- Geben Sie für die anderen Eingänge eine 0 ein.

Ein gesperrter Eingang hat bei der Übertragung den Wert 0 (Null).

Zur Zuordnung der 32 Eingänge von "InputByte_1" bis "InputByte_4" zu den 32 Bits der Sperrmaske siehe nachfolgende Tabelle.

	InputByte_1								**InputByte_2**								**InputByte_3**								**InputByte_4**							
Bit	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0
2#	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
DW#16#	_				_				_				_				_				_				_				_			

Parameter: **InversionMask**

Deklaration: INPUT

Datentyp: DWORD

Wertebereich: 0 ... 2147483647

- Als 32-Bit-Dualzahl
  2#0 ... 2#11111111_11111111_11111111_11111111
- Als 32-Bit-Hexadezimalzahl
  DW#16#0 ... DW#16#FFFF_FFFF

Vorbelegung: 0 (2#0)

Erläuterung: Invertiermaske

Die Invertierung von Eingangssignalen ist beispielsweise sinnvoll bei einer Mischung von Gebern, die nach dem Ruhestrom- bzw. nach dem Arbeitsstrom-Prinzip arbeiten.

- Geben Sie für jeden Eingang, der invertiert werden soll, an der entsprechenden Stelle im Bit-Muster eine 1 ein.
- Geben Sie für die anderen Eingänge eine 0 ein.

Zur Zuordnung der 32 Eingänge von "InputByte_1" bis "InputByte_4" zu den 32 Bits der Invertiermaske siehe nachfolgende Tabelle.

	InputByte_1								InputByte_2								InputByte_3								InputByte_4							
Bit	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0
2#	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
DW#16#	_				_				_				_				_				_				_				_			

Binärwert-Typical Bin04B_R

Funktion

4 Byte Binärinformationen empfangen

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **OutputByte_1 … _4**
Deklaration: OUTPUT
Datentyp: BYTE
Wertebereich: Siehe Adressbereich
Vorbelegung: 0 (B#16#0)

Adressbereich:	Ausgangs-Bytes	AB0 ... ABn PAB0 ... PABn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n

Erläuterung: Ausgangs-Byte

Geben Sie den Speicherbereich (1 bis 4 Bytes) der binären Informationen an, in den die binären Informationen ausgegeben werden sollen.

Ausgangs-Bytes aus dem Prozessabbild der Ausgänge, Peripherie-Bytes direkt auf Digitalausgabebaugruppen, Daten-Bytes eines Datenbausteins und Merker-Bytes können gemischt werden.

Parameter, die Sie nicht benötigen, lassen Sie einfach offen.

Zum Auslesen der mit den Daten empfangenen Zeitstempel per Anwenderprogramm siehe Kapitel Zeitstempel (Seite 1953).

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Der Ausgang "NewData" wird immer dann für einen OB1-Zyklus auf TRUE gesetzt, wenn der FB neue Daten empfangen und an die Ausgangs-Bytes "OutputByte_1" bis "OutputByte_4" ausgegeben hat.

Befehls-Typical Cmd01B_S / Cmd01B_FS

Funktion

1 Byte Befehle mit 1-aus-8-Prüfung senden

Die 1-aus-8-Prüfung wird vom Datenpunkt-Typical durchgeführt.

Die 1-aus-n-Prüfung wird vom FC Safe durchgeführt.

Hinweis

FC Safe erforderlich

Mit Cmd01B_S können nur dann Daten übertragen werden, wenn am Ende des zyklischen Programms der FC Safe eingebunden ist.

Parameter

Parameter: **PartnerNo**

PartnerObjectNo

Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter:	**Enabled**
	Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).
Typical-spezifisches Verhalten:	Freigabe der Bausteinbearbeitung
	Bei nicht freigegebener Bearbeitung wird vom FB nur noch geprüft, ob die Sperre aufgehoben wurde. Der Baustein ist in diesem Zustand auch nicht auf organisatorischer Ebene kommunikationsfähig, weil er prinzipiell keine Org.-Telegramme senden kann.

Parameter:	**CommandInputByte_HW**	
Deklaration:	INPUT	
Datentyp:	BYTE	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0 (B#16#0)	
Adressbereich:	Eingangs-Bytes	EB0 ... EBn PEB0 ... PEBn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n
Erläuterung:	Befehls-Eingangs-Byte für Hardware-Eingabe	
	Dieses Befehls-Eingangs-Byte ist speziell für die Eingabe von Befehlen über Hardware, d. h. über Digitaleingänge vorgesehen. Eingabe über Merker- oder Datenbytes ist zwar ebenfalls möglich, Sie müssen dann aber für das Rücksetzen des am Eingangs-Byte anstehenden Befehls sorgen, z. B. bei der Hardware-Eingabe durch das Loslassen der Befehlstaste.	
	Wenn eine Eingabe erkannt wird und bei der 1-aus-8- und 1-aus-n-Prüfung kein Fehler festgestellt wird und der zentrale Freigabe-Merker gesetzt ist, dann wird der Befehl zur Übertragung gebracht. Dieser wird vom FC Safe automatisch nach Ablauf einer dort parametrierten Zeitverzögerung gesetzt (siehe FC Safe, Parameter "InputDelayTime").	
	Wird ein 1-aus-8- oder 1-aus-n-Fehler festgestellt, dann wird der eingegebene Befehl nicht weiter bearbeitet. Ein neuer Befehl wird erst wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU bei diesem oder einem anderen Befehlseingabe-Baustein mit einer Hardware-Eingabe kein Hardware-Befehl erfasst wurde.	
	Der FB trägt einen erkannten 1-aus-8- oder 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B171 bzw. B172). Der Fehlerzustand wird außerdem über den Ausgang "InputError" des FC Safe zur Anzeige gebracht (siehe FC Safe, Parameter "InputError") und zwar so lange, wie der Fehler ansteht.	

Name:	**CommandInputByte_SW**
Deklaration:	IN_OUT
Datentyp	BYTE
Wertebereich:	Siehe Adressbereich

Vorbelegung:	0 (B#16#0)	
Adressbereich:	Merker-Bytes	MB0 ... MBn
	Daten-Bytes	DBm.DBB0 ... n
	Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden.	
Erläuterung	Befehls-Eingangsbyte für Software-Eingabe	

Dieses Befehls-Eingangsbyte ist speziell für die Eingabe von Befehlen über Software, d. h. über Anwenderprogramm oder Operator Panel (OP), vorgesehen. Wenn eine Eingabe erkannt wird und wenn bei der 1-aus-8- und 1-aus-n-Prüfung kein Fehler festgestellt wird, dann wird der Befehl am Eingangsbyte zurückgesetzt und zur Übertragung gebracht. Der zentrale Freigabe-Merker wird hier nicht berücksichtigt, weil dieser nur für die Befehlseingabe über Hardware vorgesehen ist (siehe "CommandInputByte_HW").

Wenn ein 1-aus-8- oder 1-aus-n-Fehler festgestellt wird, dann wird der eingegebene Befehl nicht weiter bearbeitet. Ein neuer Befehl wird erst wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU bei diesem oder einem anderen Befehlseingabe-Baustein mit Software-Eingabe kein Software-Befehl erfasst wurde.

Der FB trägt einen erkannten 1-aus-8- oder 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B171 bzw. B172). Außerdem werden im zentralen Datenbaustein BasicData entsprechende Fehlerbits gesetzt, die von der Software abgefragt werden können. Weitere Details hierzu finden Sie in der Beschreibung des FC Safe.

Prinzipiell ist es möglich, in jedem OB1-Zyklus einen neuen Befehl am "CommandInputByte_SW" anzulegen. Es darf aber pro OB1-Zyklus nur ein Befehl sein, und zwar gesehen über alle Befehlseingabe-Bausteine mit Software-Eingabe (1-aus-n-Prüfung). Ein 'Leerzyklus' zwischen zwei aufeinander folgenden Softwarebefehlen ist also nicht notwendig.

Hinweis

CommandInputByte_HW / CommandInputByte_SW

Wenn die gleichen Befehle sowohl über Hardware als auch über Software eingegeben werden sollen, können die beiden Befehlseingänge "CommandInputByte_HW" und "CommandInputByte_SW" auch parallel genutzt werden.

Wenn ein Befehl aber gleichzeitige über beide Eingangsbytes eingegeben wird, wird er nur dann entgegengenommen, wenn sowohl über die Hardware- als auch über die Software-Eingabe exakt der gleiche Befehl eingegeben wird. Die Hardware-Eingabe wird dann weiter bearbeitet.

In allen anderen Fällen wird die Eingabe abgewiesen und eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B170). Der Fehlerzustand wird außerdem über den Ausgang InputError des FC Safe zur Anzeige gebracht. Im zentralen Datenbaustein BasicData werden entsprechende Fehlerbits gesetzt, die von der Software abgefragt werden können (siehe FC Safe).

Befehls-Typical Cmd01B_R / Cmd01B_FR

Funktion

1 Byte Befehle im 1-aus-8-Format empfangen

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **Enabled**
Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Typical-spezifisches Verhalten: Freigabe der Bausteinbearbeitung

Bei nicht freigegebener Bearbeitung wird vom FB nur noch geprüft, ob die Sperre aufgehoben wurde. Eventuell noch empfangene Befehle werden nicht ausgegeben. Der FB ist in diesem Zustand auch nicht auf organisatorischer Ebene kommunikationsfähig, weil Cmd01B_R prinzipiell keine Org.-Telegramme senden oder empfangen kann.

Wenn der "Enabled"-Eingang über einen Schalter bedienbar gemacht sein sollte, dann hat diese Vor-Ort-Sperre zur Folge, dass empfangene Befehle nicht mehr ausgegeben werden. Weil der Baustein aber nicht in der Lage ist, Org.-Telegramme zu senden, kann er diese Vor-Ort-Sperre nicht selbst an den Partner zurückmelden. Dies müssten Sie mit einem anderen Typical machen, z. B. Bin04B_S.

Parameter: **MultipleOutput**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Erläuterung: Gleichzeitige Ausgabe von mehreren Befehlen zulässig

Bei diesem Parameter kann angegeben werden, ob mehrere (nacheinander empfangene) Befehle gleichzeitig ausgegeben werden dürfen, d.h. es ist festzulegen, wie der Baustein sich verhalten soll, wenn ein neuer Befehl empfangen wird und ein vorher empfangener Befehl noch ausgegeben werden soll.

Voraussetzung: Die Befehlsausgabezeit ist noch nicht abgelaufen und das Anwenderprogramm hat diesen Befehl noch nicht zurückgesetzt.

- FALSE
  Es ist keine Mehrfachausgabe zulässig. Der neu empfangene Befehl überschreibt das Ausgangs-Byte. Ein eventuell noch anstehender Befehl wird damit auf 0 zurückgesetzt, es sei denn, der neue Befehl ist identisch mit dem alten.
- TRUE
  Mehrfachausgabe ist zulässig. Ein neu empfangener Befehl wird in das aktuelle Ausgangsbyte geschrieben. Die Befehlsausgabezeit wird neu gestartet und gilt für alle anstehenden Befehle.

Parameter: **CommandOutputTime**

Deklaration: INPUT

Datentyp: INT

Wertebereich: 0 ... 500

Vorbelegung: 500

Erläuterung: Befehlsausgabezeit für die Befehlsausgänge in Millisekunden

Die angegebene Zeit gilt für alle Befehlsausgänge.

Wenn mehrere Ausgänge gleichzeitig gesetzt sein dürfen (MultipleOutput = TRUE), dann wird die Ausgabezeit mit jedem neu empfangenen Befehl erneut aufgezogen. Das bedeutet ein Nach-Triggern für bereits anstehende Befehle. Erst bei Ablauf der Ausgabezeit werden alle Befehlsausgänge gleichzeitig zurückgenommen.

Beim Wert 0 wird ein gesetzter Befehlsausgang nicht vom Befehls-Typical zurückgenommen. Dies müssen Sie über das Anwenderprogramm machen.

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Der Ausgang "NewData" wird immer dann für einen OB1-Zyklus auf TRUE gesetzt, wenn der FB neue Daten empfangen und an das Ausgangs-Byte "CommandOuputByte" ausgegeben hat.

Parameter: **CommandOutputByte**

Deklaration: IN_OUT

Datentyp: BYTE

Wertebereich: Siehe Adressbereich

Vorbelegung:	0 (B#16#0)	
Adressbereich:	(Prozessabbild)-Ausgangs-Bytes	AB0 ... ABn
	Merker-Bytes	MB0 ... MBn
	Daten- Bytes	DBm.DBB0 ... n
	Da der Parameter ein Durchgangsparameter ist (Deklaration IN_OUT), ist eine direkte Peripherie-Ausgabe des Befehls-Bytes an PAB0 ... PABn nicht zulässig! Auch die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden.	
Erläuterung:	Befehls-Ausgangsbyte	
	Damit die Befehlsausgänge sowohl vom Befehls-Typical selbst als auch vom Anwenderprogramm zurückgesetzt werden können (bei Ausgabezeit = 0), wurde der Parameter als Durchgangsparameter (IN_OUT) deklariert.	

Zählwert-Typicals Cnt01D_S / Cnt04D_S

Funktion

- Cnt01D_S: 1 Zählwert (32 Bit) senden
- Cnt04D_S: 4 Zählwerte (32 Bit) senden
  Beachten Sie, dass der Parameter "DifferenceValue" zur Differenzwert-Bildung nur für alle vier Zählwerte gleichzeitig aktiviert werden kann.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional
TimeStamp
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter:	**GeneralTriggerCommand**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE

Vorbelegung: FALSE

Erläuterung: Umspeicher-Sammelbefehl

Der Umspeicher-Sammelbefehl gehört als zentraler Systemmerker zu den organisatorischen SINAUT-Systembefehlen.

Setzen Sie den Parameter auf TRUE, wenn die Zählwertübertragung durch einen Umspeicher-Sammelbefehl ausgelöst werden soll.

- Wenn die Ziel-Teilnehmer-Nr. (PartnerNo) = 0 ist (Übertragung an alle), wird der Umspeicher-Sammelbefehl berücksichtigt.
  Beim Erkennen des Umspeicher-Sammelbefehls wird der aktuell aufsummierte Zählwert zur Übertragung gebracht, unabhängig von sonstigen Auslösern für eine Übertragung. In diesem Zählwert wird das Umspeicher-Bit invertiert.
- Wenn am Typical eine explizite Ziel-Teilnehmer-Nr. (PartnerNo > 0) parametriert ist, wird der Umspeicher-Sammelbefehl im betreffenden Teilnehmer-Objekt in der zentralen Buchführung ausgewertet.

Sie können die Parameter "GeneralTriggerCommand" und "TriggerInput" parallel nutzen. In diesem Fall wird die Übertragung sowohl bei Flankenwechsel 0 → 1 an "TriggerInput" als auch beim Empfang eines Umspeicher-Sammelbefehls ausgelöst.

Parameter: **TriggerInput**

Deklaration: INPUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Auslöse-Eingang

Über den Flankenwechsel 0 → 1 des Eingangs "TriggerInput" kann die getriggerte Übertragung zu einem gewünschten Zeitpunkt ausgelöst werden, unabhängig von sonstigen Kriterien für eine Übertragung. Es wird der aktuell aufsummierte Zählwert zur Übertragung gebracht. In diesem Zählwert wird das Umspeicher-Bit (siehe oben) invertiert.

Beispiel:
Eine zeitgesteuerte Übertragung mit Zeitstempel zur Versorgung eines Archivs in der Leitstelle.

Sie können die Parameter "GeneralTriggerCommand" und "TriggerInput" parallel nutzen. In diesem Fall wird die Übertragung sowohl bei Flankenwechsel 0 → 1 an "TriggerInput" als auch beim Empfang eines Umspeicher-Sammelbefehls ausgelöst.

In der Einstellung FALSE wird kein Umspeichern und keine Übertragung über den Eingang "TriggerInput" ausgelöst.

Parameter: **Counter_1 (Cnt01D_S)**
Counter_1 ... _4 (Cnt04D_S)
Deklaration: INPUT
Datentyp: COUNTER
Wertebereich: 0 ... 32767
- Z0 als Platzhalter
  oder
- Z1 ... Zn
  n ist abhängig vom CPU-Typ.

Vorbelegung: -
Erläuterung: Nummer des SIMATIC-Zählers

Angabe des SIMATIC-Zählers, in den die Zählimpulse zeitgesteuert eingezählt wurden. Das Einzählen wird im Hintergrund über den FC PulseCounter durchgeführt, der in einem Weckalarm-OB aufgerufen wird (z. B. im OB35). Siehe auch Kapitel FC PulseCounter (Seite 2040) und Weckalarm-OB (Seite 1939).

Der Datentyp COUNTER kann nicht mit einem Wert vorbelegt werden.

Wenn Sie den Platzhalter Z0 parametrieren, wird der zugehörige Zählwert nicht bearbeitet.

Parameter: **DifferenceValue**
Deklaration: INPUT
Datentyp: INT
Wertebereich: 0 ... 31767
Vorbelegung: 0
Erläuterung: Differenzwert
- Bei Parametrierung eines Werts zwischen 1 und 31767 wird der Zählwert übertragen, sobald die Differenz zwischen aktuellem und zuletzt übertragenem Zählwert den hier angegebenen Betrag erreicht.
- Bei Parametrierung des vorbelegten Werts 0 wird ein Zählwert nur in den folgenden Fällen übertragen:
  – Bei Flankenwechsel 0 → 1 am Eingang "TriggerInput"
  – Bei Empfang eines Umspeicherbefehls, wenn "GeneralTriggerCommand" = TRUE.

Wählen Sie den Differenzwert in Abhängigkeit der maximalen Impulsrate pro Sekunde.

Wählen Sie den Wert nicht zu klein, damit der Zählwert nicht ständig an die TIM übergeben wird. Dies würde den Kommunikationsweg zur CPU und die Sendewarteschlange der CPU belasten.

Hinweis zu Cnt04D_S
Am Typical ist dieser Verarbeitungsparameter zur Differenzwert-Bildung nur einmal vorhanden. Er gilt für alle 4 Zählwerte gemeinsam. Eine individuelle Einstellung des Parameters pro Zählwert ist nicht möglich. Bei Nutzung des Parameters sollten deshalb pro Typical nur Zählwerte erfasst werden, die identisch bearbeitet werden können.

Zählwert-Typicals Cnt01D_R / Cnt04D_R

Funktion

- Cnt01D_S: 1 Zählwert (32 Bit) empfangen
- Cnt04D_S: 4 Zählwerte (32 Bit) empfangen

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **BCD_Format**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: TRUE
Erläuterung: Zählwertausgabe in BCD-Format

- Bei aktiviertem Parameter wird der empfangene Zählwert am Ausgang "CountedValueOutput_n" als positiver BCD-Wert ausgegeben.
- Bei deaktiviertem Parameter wird der Zählwert als positiver 32-Bit-Integer-Wert ausgegeben.

Zu den unterschiedlichen Wertebereichen der beiden Formate siehe Parameter "CountedValueOutput_n".

Beim Überschreiten des maximal darstellbaren Zählwerts fängt der Zählwert wieder bei 0 an und wird im positiven Zahlenbereich weitergezählt.

Hinweis zu Cnt04D_R
Der Parameter ist nur einmal am Typical vorhanden und gilt für alle 4 Zählwerte gemeinsam. Eine individuelle Einstellung pro Zählwert ist nicht möglich. Bei Nutzung des Parameters sollten deshalb pro Typical nur Zählwerte mit identischem Ausgabeformat ausgegeben werden.

Parameter: **CntValInvalid**
Deklaration: OUTPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE

Adressbereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Zählwert ungültig

Beachten Sie bei der Auswertung von "CntValInvalid", dass das Bit eventuell nur einen OB1-Zyklus lang gesetzt wird.

Der Ausgang "CntValInvalid" zeigt an, ob der zuletzt empfangene Zählwert ungültig war. Bei "Cnt04D_R" gilt dies als Sammelanzeige für alle 4 Zählwerte, siehe Hinweis unten.

Praktisch gesehen zeigt der Ausgang den Gültigkeitsstatus des zuletzt empfangenen Zählwerts in invertierter Form an.

Der Ausgang dient folgenden Zwecken:

- Fehleranzeige
- Signal für anwenderspezifische Weiterverarbeitungen
  Sie können beispielsweise auf die Nicht-Aktualität reagieren, indem Sie den an "CountedValueOutput_n" ausgegebenen Zählwert um eventuell verlorengegangene Zählimpulse korrigieren.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Hinweis zu Cnt04D_R
Obwohl alle 4 Zählwerte im Daten-Telegramm ein eigenes Status-Bit führen, wird für den Status am Ausgang "CntValInvalid" nur das Bit des ersten Zählwerts im zuvor empfangenen Daten-Telegramm ausgewertet.
Dieser Status gilt aber für alle 4 Zählwerte.
(Alle Zählwerte im Daten-Telegramm haben immer den gleichen Status.)

Parameter:	**RestoreStatus**
Deklaration:	OUTPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE
Vorbelegung:	FALSE

Adressbereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Status des Umspeicher-Bits im empfangenen Zählwert

Der Ausgang "RestoreStatus" zeigt den aktuellen Status des Umspeicher-Bits aus dem zuletzt empfangenen Zählwert-Telegramm an.

Sie können den Ausgang für anwenderspezifische Weiterverarbeitungen nutzen.

Beispiel:
Sie können nur dann auf die Information an "CountedValueOutput_n" zugreifen, wenn am Ausgang "RestoreStatus" einen Wechsel feststellt wird, d. h wenn der Zählwert aufgrund einer Umspeicherung empfangen wurde , beispielsweise bei einer zeitgesteuerten Umspeicherung vor Ort.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Hinweis zu Cnt04D_R
Obwohl alle 4 Zählwerte im Daten-Telegramm ein eigenes Umspeicher-Bit führen, wird für den Status am Ausgang "RestoreStatus" nur das Umspeicher-Bit des ersten Zählwerts im zuvor empfangenen Daten-Telegramm ausgewertet. Dieser Status gilt aber für alle 4 Zählwerte.
(Alle Zählwerte im Daten-Telegramm werden immer zusammen umgespeichert.)

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Der Ausgang "NewData" wird immer dann für einen OB1-Zyklus auf TRUE gesetzt, wenn der FB neue Daten empfangen und an den Ausgang / an die Ausgänge "CountedValueOutput_1" bis "CountedValueOutput_4" ausgegeben hat.

Parameter: **CountedValueOutput_1 (Cnt01D_S)**
CountedValueOutput_1 ... _4 (Cnt04D_S)

Deklaration: IN_OUT

Datentyp: DWord, UDInt

Wertebereich:
- Integer: 0 ... 2 147 483 647
- BCD: 0 ... 9 999 999

Vorbelegung: 0

Adressbereich:	Ausgang (DWORD)	QD0 … QDn
	Merker (DWORD)	MD0 … MDn
	Daten (DWORD)	DBm.DBB0 … n

Da der Parameter ein Durchgangsparameter ist (Deklaration IN_OUT) ist eine direkte Peripherie-Ausgabe des Zählwerts an PAD0 ... PADn nicht zulässig!

Auch die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden.

Erläuterung: Zählwertausgang

Vom Zählwert-Typical wird immer der neu gebildete Differenzwert (Differenz zwischen neuem und zuletzt empfangenem Zählwert) auf den aktuell am Zählwertausgang ausgegebenen Wert aufaddiert.

Der Zählwertausgang umfasst ein Doppelwort, in dem der Zählwert im BCD-Format oder als 32-Bit-Integer-Wert abgelegt ist (abhängig vom Parameter "BCD_Format", siehe oben).

Der Zählwert wird immer als positive Zahl ausgegeben. Bei Überschreitung des maximal darstellbaren Zählwerts fängt der Zählwert wieder bei 0 an und es wird im positiven Zahlenbereich weitergezählt.

Da der Parameter ein Durchgangsparameter ist (IN_OUT) kann der Wert am Zählwertausgang zu jeder Zeit über das Anwenderprogramm auf 0 oder einen anderen Wert zurückgesetzt werden.

Daten-Typical Dat12D_S

Funktion

Max. 12 Doppelworte mit beliebigem Dateninhalt senden

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (DWORD, DINT, REAL), es ist aber auch eine Mischung von anderen Datentypen zulässig, die zusammen wieder ein Doppelwort ergeben, beispielsweise:

- 4 Bytes
- 2 Worte
- 2 Bytes + 1 Wort

Das Senden des Datenbereichs kann auf zwei Arten ausgelöst werden:

- Über eine Änderungskontrolle
  Die Daten werden übertragen, sobald sich ein Bit ändert ("SendOnChange" = TRUE).
- Über das Anwenderprogramm
  Die Übertragung kann durch Flankenwechsel 0 → 1 am Eingang "TriggerInput" ausgelöst werden.
  Für eine zeitgesteuerte Übertragung bietet sich der FC Trigger an.

Zusätzlich kann über "SendAll" festgelegt werden, ob bei einer Übertragung immer alle Daten gesendet werden sollen oder nur die geänderten Daten-Doppelworte.

Hinweis

Doppelwort-Grenzen beachten

Wenn geänderte Daten übertragen werden und der Datenbereich Werte in Doppelwortformat enthält, dann achten Sie darauf, dass die Doppelwort-Werte tatsächlich in einem der max. 12 Doppelworte des zu erfassenden Datenbereichs liegen.

Eine Verteilung auf zwei aufeinander folgende Daten-Doppelworte könnte dazu führen, dass nur ein Wort des Doppelwort-Werts (High- oder Low-Wort) übertragen wird, weil nur darin sich etwas verändert hat. Das fehlende Wort kann in diesem Fall beim empfangenden Partner zu einem Datenfehler führen (gilt für ST7cc, nicht für eine S7-CPU).

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional
TimeStamp
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **SendOnChange**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Senden bei Änderung

In der Einstellung TRUE führt der Baustein eine Änderungskontrolle innerhalb des erfassten Datenbereichs "DataInput" durch. Der Baustein prüft, ob sich mindestens ein Bit geändert hat. Bei Änderung wird automatisch eine Übertragung des Datenbereichs veranlasst. Ob dabei der ganze Bereich zur Übertragung kommt oder nur der geänderte Teil, legen Sie über den Parameter "SendAll" fest.

In der Einstellung FALSE müssen Sie die Übertragung über den Eingangsparameter "TriggerInput" auslösen.

Parameter:	**TriggerInput**	
Deklaration:	INPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Auslöse-Eingang

Über den Flankenwechsel 0 → 1 des Eingangs "TriggerInput" kann die getriggerte Übertragung des Daten-Telegramms zu einem gewünschten Zeitpunkt ausgelöst werden.

Beispiel:
Eine zeitgesteuerte Analogwertübertragung mit Zeitstempel zur Versorgung eines Analogwertarchivs in der Leitstelle.

Achten Sie darauf, den Parameter "ImageMemory" auf FALSE einzustellen, damit diese Daten mit Zeitstempel beim Speichern auf der Stations-TIM nicht überschrieben werden.

Für die zeitgesteuerte Auslösung einer Übertragung über "TriggerInput" bietet sich die Anwendung des Bausteins FC Trigger an.

Im eigentlichen Sinne wird über "TriggerInput" nur indirekt eine Übertragung ausgelöst. Beim Flankenwechsel 0 → 1 an "TriggerInput" wird das Daten-Telegramm mit seinen aktuellen Werten zusammengestellt und an die lokale TIM übergeben. Diese ist dann für die eigentliche Übertragung an den Partner zuständig. Bei Standleitungen oder Funknetzen wird umgehend übertragen. Bei einer Wählverbindung kann es sein, dass das Daten-Telegramm zunächst auf der TIM gespeichert wird und erst zu einem späteren Zeitpunkt zur Übertragung gelangt. Grund kann beispielsweise sein, dass das Daten-Telegramm als "bedingt spontan" gekennzeichnet ist, siehe Parameter "Conditional".

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen. Sie sollten dann aber den Parameter "SendOnChange" auf TRUE stellen, damit die Datenübertragung automatisch bei jeder Änderung veranlasst wird.

Sie können für das Auslösen der Datenübertragung auch eine Kombination von "SendOnChange" plus "TriggerInput" wählen. Dann wird sowohl bei Feststellung einer Änderung als auch bei jedem Flankenwechsel von 0 nach 1 am Eingang "TriggerInput" eine Übertragung ausgelöst.

Wenn Sie weder "SendOnChange" noch "TriggerInput" für die Auslösung der Datenübertragung nutzen, dann wird die Übertragung nur bei einer Einzelabfrage dieses Datenobjekts oder im Rahmen einer Generalabfrage durchgeführt.

Parameter:	**SendAll**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE

Vorbelegung: TRUE

Erläuterung: Bei jeder Übertragung alle Daten senden

Mit dem Parameter wird eingestellt, ob der Baustein immer alle Daten des mit "DataInput" angegebenen Bereichs übertragen soll oder nur die geänderten. Die Übertragung kann dabei aufgrund der aktivierten Änderungskontrolle ausgelöst worden sein (SendOnChange = TRUE), oder über "TriggerInput".

- SendAll = TRUE
  Immer alle Daten senden
- SendAll = FALSE
  Nur die geänderten Daten senden
  Ausnahme:
  Wenn "SendAll" = FALSE gesetzt ist, die Übertragung über "TriggerInput" ausgelöst wird und sich zu diesem Zeitpunkt keine Daten geändert haben, wird der komplette Bereich übertragen. Das entspricht für diesen Ausnahmefall "SendAll" = TRUE.
  Wenn nur der geänderte Datenbereich übertragen wird ("SendAll" = FALSE), dann besteht dieser Bereich aus dem ersten und dem letzten Doppelwort, in denen eine Änderung festgestellt wurde, sowie alle dazwischen liegenden Worte, auch wenn diese sich nicht geändert haben.

Bei einer Einzelabfrage dieses Datenobjekts oder im Rahmen einer Generalabfrage werden immer alle Datenworte des bei "DataInput" angegebenen Bereichs übertragen.

Parameter: **DataInput**

Deklaration: INPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:
`P#DB20.DBX 100.0 DWORD 4`

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Daten-Eingangsbereich

Der ANY-Pointer adressiert den Datenbereich, in dem die zu erfassenden Daten bereitstehen. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Daten-Doppelworten variieren.

Zu Inhalt und Formaten siehe Abschnitt "Funktion" oben.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Daten-Typical Dat12D_R

Funktion

Max. 12 Doppelworte mit beliebigem Dateninhalt empfangen

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (DWORD, DINT, REAL), es ist aber auch eine Mischung von anderen Datentypen zulässig, die zusammen wieder ein Doppelwort ergeben, beispielsweise:

- 4 Bytes
- 2 Worte
- 2 Bytes + 1 Wort

Dat12D_R legt die empfangenen Daten ohne weitere Aufbereitung in dem bei "DataOutput" angegebenen Datenbereich ab. Sie müssen die empfangenen Daten über das Anwenderprogramm auswerten und verarbeiten.

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **DataOutput**
Deklaration: INPUT
Datentyp: ANY
Wertebereich: Siehe Adressbereich
Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)
Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:
```
P#DB20.DBX 100.0 DWORD 4
```
Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Daten-Ausgabebereich

Der ANY Pointer adressiert den Datenbereich, in dem die empfangenen Daten abgespeichert werden. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Doppelworten variieren.

Zu Inhalt und Formaten siehe Abschnitt "Funktion" oben.

Dat12D_R legt die empfangenen Daten ohne weitere Aufbereitung in dem bei "DataOutput" angegebenen Datenbereich ab. Sie müssen die empfangenen Daten über das Anwenderprogramm auswerten und verarbeiten.

Wenn vom Partnerobjekt Dat12D_S nur geänderte Daten übertragen werden, ist es möglich, dass nur ein Teil des Daten-Ausgangsbereichs neu beschrieben wird. Dies ist derjenige Bereich, in dem auf der Erfassungsseite Änderungen festgestellt wurden.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Parameter:	**NewData**
Erläuterung:	Neue Daten empfangen Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945). Jedes Mal, wenn der Baustein neue Datenwerte vom Partnerobjekt empfangen und an das Ausgabefeld "DataOutput" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt.

Daten-Typical Dat256D_S

Funktion

Maximal 256 Doppelworte mit beliebigem Dateninhalt senden

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (z. B. DINT, REAL usw.). Es ist auch eine Kombination aus anderen Formaten zulässig, die zusammen wieder ein Doppelwort ergeben, z. B.

- 32 Bool
- 4 Bytes
- 2 Worte
- Jede Kombination wie 2 Bytes plus 1 Wort etc.

Hinweis

Doppelwort-Grenzen beachten

Wenn geänderte Daten übertragen werden und der Datenbereich Werte in Doppelwortformat enthält, dann achten Sie darauf, dass die Doppelwort-Werte tatsächlich in einem der max. 256 Doppelworte des zu erfassenden Datenbereichs liegen.

Eine Verteilung auf zwei aufeinander folgende Daten-Doppelworte könnte dazu führen, dass nur ein Wort des Doppelwort-Werts (High- oder Low-Wort) übertragen wird, weil nur darin sich etwas verändert hat. Das fehlende Wort kann in diesem Fall beim empfangenden Partner zu einem Datenfehler führen (gilt für ST7cc, nicht für eine S7-CPU).

Das Senden des Datenbereichs kann auf zwei Arten ausgelöst werden:

- Über eine Änderungskontrolle
  Die Daten werden übertragen, sobald sich ein Bit ändert ("SendOnChange" = TRUE).
- Über das Anwenderprogramm
  Die Übertragung kann durch Flankenwechsel 0 → 1 am Eingang "TriggerInput" ausgelöst werden.
  Für eine zeitgesteuerte Übertragung bietet sich der FC Trigger an.

Zusätzlich kann über "SendAll" festgelegt werden, ob bei einer Übertragung immer alle Daten gesendet werden sollen oder nur die geänderten Daten-Doppelworte.

Bei S7-300-CPUs mit X-Kommunikation beträgt die maximale Länge eines Daten-Telegramms 76 Byte. 1024 Byte Nettodaten werden über einen seriellen Übertragungsprozess übertragen,

der aus einer Sequenz von mindestens 22 Daten-Telegrammen (Segmenten) besteht. Jedes Daten-Telegramm, außer dem letzten, enthält ein Segment aus 48 Byte Nettodaten des Eingangsdatenbereichs.

Um bei aktiviertem Parameter "SendAll" oder während einer General- bzw. Einzelabfrage Datenkonsistenz sicherzustellen, werden die Daten in aufeinanderfolgenden Segmenten übertragen. Während des Übertragungsprozesses wird der Status über "SendAllBusy" angezeigt. Beim Empfänger wird der Status am Ausgang "DataStatus" angezeigt.

Hinweis

TriggerInput - SendAllBusy

Wenn "TriggerInput" mit "SendAllBusy" = TRUE ausgelöst wird, dann führt dies bei erneutem Auslösen der Übertragung zur Fehlermeldung "DataLoss" (Status im Telegrammkopf).

Nur wenn "SendAllBusy" = FALSE gesetzt ist, wird der Flankenwechsel 0 → 1 bei "TriggerInput" ausgelöst.

Wenn die Übertragung unterbrochen wird, dann wird "SendAllError" angezeigt. Außerdem erfolgt ein Eintrag im Diagnosepuffer mit der Ereignis-ID B14DTD7_Diagnostics.

Bei unvollständiger Übertragung wird der Datenstatus beim Empfänger ebenfalls "ungültig". Dies wird beim Empfänger am den Parameter DataStatus angezeigt. Außerdem wird ein Eintrag mit der Ereignis-ID B13BTD7_Diagnostics in den Diagnosepuffer geschrieben.

Hinweis

Verfügbarkeit des Partners

Wenn der Status des Partners von "verfügbar" nach "nicht verfügbar" wechselt, wird die Übertragung sämtlicher Daten sofort gestoppt. Alle Objektdaten werden aus dem TIM-Puffer gelöscht. Dies kann zu Datenverlust führen.

Sobald der Partner wieder verfügbar ist, wird durch die automatische Generalabfrage sichergestellt, dass die Daten des Partners bei der nächsten Übertragung wieder aktuell werden.

Hinweis

Dat256D_S und Dat256D_R benötigen den UDT "TransmitBlock".

Kopieren Sie den UDT bei Verwendung des Typicals aus der globalen Bibliothek in das Verzeichnis "PLC-Datentypen" der CPU. Der UDT wird vom Typical automatisch aus dem Bausteinverzeichnis der CPU referenziert, aber nicht aus der globalen Bibliothek.

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Unconditional
TimeStamp
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **SendOnChange**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Senden bei Änderung

In der Einstellung TRUE führt der Baustein eine Änderungskontrolle innerhalb des erfassten Datenbereichs "DataInput" durch. Der Baustein prüft, ob sich mindestens ein Bit geändert hat. Bei Änderung wird automatisch eine Übertragung des Datenbereichs veranlasst. Ob dabei der ganze Bereich zur Übertragung kommt oder nur der geänderte Teil, legen Sie über den Parameter "SendAll" fest.

In der Einstellung FALSE müssen Sie die Übertragung über den Eingangsparameter "TriggerInput" auslösen.

Parameter: **TriggerInput**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Auslöse-Eingang

Über den Flankenwechsel 0 → 1 des Eingangs "TriggerInput" kann die getriggerte Übertragung des Daten-Telegramms zu einem gewünschten Zeitpunkt ausgelöst werden.

Beispiel:
Eine zeitgesteuerte Analogwertübertragung mit Zeitstempel zur Versorgung eines Analogwertarchivs in der Leitstelle.

Für die zeitgesteuerte Auslösung einer Übertragung über "TriggerInput" bietet sich die Anwendung des Bausteins FC Trigger an.

Im eigentlichen Sinne wird über "TriggerInput" nur indirekt eine Übertragung ausgelöst. Beim Flankenwechsel 0 → 1 an "TriggerInput" wird das Daten-Telegramm mit seinen aktuellen Werten zusammengestellt und an die lokale TIM übergeben. Diese ist dann für die eigentliche Übertragung an den Partner zuständig. Bei Standleitungen oder Funknetzen wird umgehend übertragen. Bei einer Wählverbindung kann es sein, dass das Daten-Telegramm zunächst auf der TIM gespeichert wird und erst zu einem späteren Zeitpunkt zur Übertragung gelangt. Grund kann beispielsweise sein, dass das Daten-Telegramm als "bedingt spontan" gekennzeichnet ist, siehe Parameter "Conditional".

Wählen Sie geeignete Auslösezeitpunkte, damit die Daten auf der TIM nicht durch Pufferüberlauf überschrieben werden (zu lange Zeitabstände).

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen. Sie sollten dann aber den Parameter "SendOnChange" auf TRUE stellen, damit die Datenübertragung automatisch bei jeder Änderung veranlasst wird.

Sie können für das Auslösen der Datenübertragung auch eine Kombination von "SendOnChange" plus "TriggerInput" wählen. Dann wird sowohl bei Feststellung einer Änderung als auch bei jedem Flankenwechsel von 0 nach 1 am Eingang "TriggerInput" eine Übertragung ausgelöst.

Wenn Sie weder "SendOnChange" noch "TriggerInput" für die Auslösung der Datenübertragung nutzen, dann wird die Übertragung nur bei einer Einzelabfrage dieses Datenobjekts oder im Rahmen einer Generalabfrage durchgeführt.

Übertragen Sie keine Analogwerte, bei denen der Parameter "SendOnChange" = TRUE gesetzt ist, ohne vorher die Prozessdaten der Vorverarbeitung zu unterziehen. Nähere Angaben dazu finden Sie unter dem Analogwert-Typical Ana04W_S, Parameter "ThresholdValue".

Parameter: **SendAll**

Deklaration: INPUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: TRUE

Erläuterung: Bei jeder Übertragung alle Daten senden

Mit dem Parameter wird eingestellt, ob der Baustein immer alle Daten des mit "DataInput" angegebenen Bereichs übertragen soll oder nur die geänderten. Die Übertragung kann dabei aufgrund der aktivierten Änderungskontrolle ausgelöst worden sein (SendOnChange = TRUE), oder über "TriggerInput".

- SendAll = TRUE
  Immer alle Daten senden
- SendAll = FALSE
  Nur die geänderten Daten senden
  Ausnahme:
  Wenn "SendAll" = FALSE gesetzt ist, die Übertragung über "TriggerInput" ausgelöst wird und sich zu diesem Zeitpunkt keine Daten geändert haben, wird der komplette Bereich übertragen. Das entspricht für diesen Ausnahmefall "SendAll" = TRUE.

Bei einer Einzelabfrage dieses Datenobjekts oder im Rahmen einer Generalabfrage werden immer alle Datenworte des bei "DataInput" angegebenen Bereichs übertragen.

Parameter: **DataInput**

Deklaration: INPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...256 ab Byte-Nummer yy

Beispiel:
`P#DB20.DBX 100.0 DWORD 200`

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Daten-Eingangsbereich

Der ANY-Pointer adressiert den Datenbereich, in dem die zu erfassenden Daten bereitstehen. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 256 Daten-Doppelworten variieren. Zu den möglichen Doppelwortformaten siehe Abschnitt "Funktion" oben.

Bei falscher Parametrierung (Null-Pointer, Länge > 256, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Datenkonsistenz:
Wenn ein zu übertragendes Datensegment aus höchstens 48 Byte besteht, dann ist Datenkonsistenz bei der Übertragung gegeben.

Parameter: **SendAllBusy**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Baustein in Bearbeitung mit "SendAll" = TRUE

Dieser Ausgang zeigt an, dass der Baustein momentan die bei "DataInput" festgelegten Daten überträgt. Der Vorgang wird entweder durch eine remote Einzel- oder Generalabfrage aktiviert oder durch einen lokalen internen oder externen Trigger gestartet.

Wenn "SendAll" auf TRUE gesetzt ist, dann wird die Übertragung aller Daten entweder durch interne Änderungskontrolle (SendOnChange = TRUE) oder über den externen "TriggerInput" (Flankenwechsel 0 → 1) ausgelöst.

Der Flankenwechsel 0 → 1 hat bei externem "TriggerInput" keinerlei Auswirkung, solange "SendAllBusy" TRUE anzeigt. Der Flankenwechsel 0 → 1 von "TriggerInput" wirkt erst bei "SendAllBusy" = FALSE.

Parameter: **SendAllError**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Fehler bei Verarbeitung von "SendAll"

"SendAllError" wird auf TRUE gesetzt, wenn die Verbindung während der Verarbeitung von "SendAll" unterbrochen wird. Außerdem wird ein Eintrag in den Diagnosepuffer geschrieben.

"SendAllError" bleibt bis zu dessen Rücksetzen über das Anwenderprogramm oder bis zum nächsten Wiederanlauf der CPU auf TRUE.

Daten-Typical Dat256D_R

Funktion

Max. 256 Doppelworte mit beliebigem Dateninhalt empfangen

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (z. B. DINT, REAL usw.), es ist aber auch eine Mischung von anderen Formaten zulässig, die zusammen wieder ein Doppelwort ergeben, z. B.:

- 32 Bool
- 4 Bytes
- 2 Worte
- Jede Kombination wie 2 Bytes plus 1 Wort etc.

Hinweis

Doppelwort-Grenzen beachten

Wenn geänderte Daten übertragen werden und der Datenbereich Werte in Doppelwortformat enthält, dann achten Sie darauf, dass die Doppelwort-Werte tatsächlich in einem der max. 256 Doppelworte des zu erfassenden Datenbereichs liegen.

Eine Verteilung auf zwei aufeinander folgende Daten-Doppelworte könnte dazu führen, dass nur ein Wort des Doppelwort-Werts (High- oder Low-Wort) übertragen wird, weil nur darin sich etwas verändert hat. Das fehlende Wort kann in diesem Fall beim empfangenden Partner zu einem Datenfehler führen (gilt für ST7cc, nicht für eine S7-CPU).

Dat256D_R legt die empfangenen Daten ohne weitere Aufbereitung in dem mit "DataOutput" angegebenen Datenbereich ab. Werten Sie die empfangenen Daten über das Anwenderprogramm aus.

Bei S7-300-CPUs mit X-Kommunikation beträgt die maximale Länge eines Daten-Telegramms 76 Byte (netto 48 Byte). 1024 Byte Nettodaten werden über einen seriellen Übertragungsprozess übertragen, der aus einer Sequenz von mindestens 22 Daten-Telegrammen (Segmenten) besteht. Jedes Daten-Telegramm, außer dem letzten, enthält ein Segment aus 48 Byte Nettodaten des Ausgangsdatenbereichs.

Jeder Empfang eines erkannten Datensegments wird über einen entsprechenden Status (Bit 1 bis 22) des Ausgangsparameters "NewData" angezeigt.

Wenn eine Änderung im Datensegment erkannt wurde, wird in "NewData" außerdem das Status-Bit 0 für einen CPU-Zyklus auf TRUE gesetzt. Dadurch kann festgestellt werden, welches Segment des Ausgangsdatenbereichs sich geändert hat.

Hinweis

Beim Empfangen einer Sequenz aus mehreren Datensegmenten (Daten-Telegrammen) werden im Parameter "NewData" die Status-Bits 1 bis 22 nacheinander auf TRUE gesetzt und bleiben solange auf TRUE, bis das letzte Segment empfangen worden ist.

Wenn ein Datensegment (Daten-Telegramm) nicht Teil einer empfangenen Sequenz ist (SendAll = FALSE), dann verbleibt der Status für nur einen einzigen CPU-Zyklus auf TRUE.

Um bei "SendAll" = TRUE oder bei einer General- bzw. Einzelabfrage Datenkonsistenz sicherzustellen, wird der Datenbereich in aufeinander folgenden einzelnen Segmenten aktualisiert.

Während des Empfangsprozesses wird der Status über das Ausgangs-Byte "DataStatus" (Status "SequenceState") angezeigt. Wenn die Empfangssequenz erfolgreich abgeschlossen wurde, ist der Datenausgangsbereich auf dem aktuellen Stand und Konsistenz der Ausgangsdaten ist gegeben. Dies wird durch "DataStatus" (Status "DataValid" = TRUE) angezeigt.

Hinweis

Es kann keine Konsistenz der Datensegmente oder der Grenzsegmente garantiert werden, wenn der Parameter "SendAll" beim Sender auf FALSE gesetzt wurde.

Der Empfang einer Sequenz kann aus folgenden Gründen gestört werden:

- Die Empfangssequenz wurde unterbrochen, wenn die Kommunikation zum Partner während einer laufenden Sequenz ausfällt (Ereignis-ID B13BTD7_Diagnostics).
- Die Überwachungszeit wurde überschritten. Es konnten nicht alle Segmente in der Zeit empfangen werden, die beim Parameter "MonitoringTime" eingestellt wurde (Ereignis-ID B13CTD7_Diagnostics).
- Andere Empfangsfehler treten auf (Ereignis-ID B13DTD7_Diagnostics), z. B.:
  - Während einer fehlerfreien laufenden Sequenz wird eine neue Empfangssequenz registriert.
  - Während einer laufenden Sequenz wird ein spontanes Segment (Daten-Telegramm) empfangen.

Hinweis

Dat256D_S und Dat256D_R benötigen den UDT "TransmitBlock".

Kopieren Sie den UDT bei Verwendung des Typicals aus der globalen Bibliothek in das Verzeichnis "PLC-Datentypen" der CPU. Der UDT wird vom Typical automatisch aus dem Bausteinverzeichnis der CPU referenziert, aber nicht aus der globalen Bibliothek.

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **SingleRequest**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: TRUE

Adressbereich:	Eingang	I 0.0 … I n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Eine Einzelabfrage wird an das Partner-Objekt gesendet.

Bei Verfügbarkeit des Partners können Sie eine Einzelanfrage an das Partnerobjekt senden. Wenn eine Antwort zurückgesendet wird, dann wird die Information an den in "DataOutput" angegebenen Datenbereich weitergeleitet.

Bezüglich der Übertragungssequenzen gibt es Prioritäten:

- Niedrigste Priorität: TriggerInput
  Eine laufende Übertragung, die beispielsweise durch "TriggerInput" beim Sender ausgelöst wurde, kann durch eine Einzelabfrage oder durch eine Generalabfrage unterbrochen werden.
- Mittlere Priorität: Einzelabfrage
  Eine laufende Übertragung, die beispielsweise durch "TriggerInput" beim Sender ausgelöst wurde, kann durch eine Einzelabfrage oder eine Generalabfrage unterbrochen werden.
  Die unterbrochene bzw. neu gestartete Abfrage führt ohne Fehlermeldung zu einem Neustart der laufenden Sequenzen.
- Höchste Priorität: Generalabfrage
  Eine Generalabfrage kann sich selbst oder eine Einzelabfrage unterbrechen. Die unterbrochene bzw. neu gestartete Abfrage führt ohne Fehlermeldung zu einem Neustart der laufenden Sequenzen.

Die durch eine Einzel- oder Generalabfrage unterbrochene bzw. neu gestartete Abfrage führt ohne Fehlermeldung zu einem Neustart der laufenden Sequenzen. Wenn eine Sequenz erfolgreich abgeschlossen wurde, verbleibt der Status "DataValid" des Ausgangs-Bytes "DataStatus" auf TRUE.

Die Zeitdauer der Antwort auf die Einzelabfrage wird durch den Parameter "MonitoringTime" ausgewertet.

Hinweis:
Konsistenz des Datenausgangsbereichs über Segmentgrenzen hinweg ist nur dann gegeben, wenn die Empfangssequenz erfolgreich abgeschlossen wurde.

Parameter: **MonitoringTime**

Deklaration: INPUT

Datentyp: INT

Wertebereich: 0 (keine Begrenzung) / 1 … 32000 (Sekunden)

Vorbelegung: 0

Erläuterung: Höchstdauer einer kompletten Antwort auf eine Einzelabfrage

Bei jedem Start einer Einzelabfrage (siehe Parameter SingleRequest) wird der hier angegebene Zeitraum bei "SingleRequest" aktiviert.

Wenn ein Wert über 0 eingegeben ist und die Zeit für die Antwort-Sequenz überschritten wird, dann wird ein Fehler über das Ausgangs-Byte "DataStatus" (Status-Bits "SequenceState") angezeigt. Außerdem wird ein Eintrag in den Diagnosepuffer geschrieben (Ereignis-ID B13CTD7_Diagnostics).

Bei jedem Start einer Einzelabfrage wird "MonitoringTime" neu aktiviert.

Parameter: **DataOutput**

Deklaration: INTPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...256 ab Byte-Nummer yy

Beispiel:

```
P#DB20.DBX 100.0 DWORD 200
```

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Daten-Ausgangsbereich

Der ANY-Pointer adressiert den Datenbereich, in dem die empfangenen Daten abgespeichert werden. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 256 Daten-Doppelworten variieren. Zu den möglichen Doppelwortformaten siehe Abschnitt "Funktion" oben.

Dat256D_R legt die empfangenen Daten ohne weitere Aufbereitung in dem bei "DataOutput" angegebenen Datenbereich ab. Sie müssen die empfangenen Daten über das Anwenderprogramm auswerten und verarbeiten.

Wenn vom Partnerobjekt Dat256D_S nur geänderte Daten übertragen werden, ist es möglich, dass nur ein Teil des Daten-Ausgangsbereichs neu beschrieben wird. Dies ist derjenige Bereich, in dem auf der Erfassungsseite Änderungen festgestellt wurden.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Parameter: **NewData**

Deklaration: OUTPUT

Datentyp: DWORD

Wertebereich: Siehe Adressbereich

Vorbelegung: 0 (DW#16#0)

Adressbereich:	Ausgang (DWORD)	QD0 … QDn PQD0 … PQDn
	Merker (DWORD)	MD0 … MDn LD0 … LDn
	Daten (DWORD)	DBm.DBB0 … n

Erläuterung: Neue Daten empfangen

Immer wenn der Baustein neue Datensegmente empfängt, erscheint bei "NewData" die Anzeige der Status-Bits 1 bis 22 entsprechend der empfangenen Datensegmente 1 bis 22.

Wenn mindestens ein Datensegment der empfangenen Daten Änderungen enthält, wird Bit 0 von "NewData" für einen OB1-Zyklus auf TRUE gesetzt.

Beim Empfangen einer Sequenz aus mehreren Datensegmenten (Daten-Telegrammen) werden im Parameter "NewData" die Status-Bits 1 bis 22 nacheinander auf TRUE gesetzt und bleiben solange auf TRUE, bis das letzte Segment empfangen worden ist.

Wenn ein Datensegment (Daten-Telegramm) nicht Teil einer empfangenen Sequenz ist ("SendAll" = FALSE), dann verbleibt der Status für nur einen einzigen CPU-Zyklus auf TRUE.

Der Ausgang ist für anwenderspezifische Weiterverarbeitungen vorgesehen, z. B. um gezielt auf den Empfang von neuen Daten zu reagieren.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **DataStatus**

Deklaration: OUTPUT

Datentyp: BYTE

Wertebereich: Siehe Adressbereich

Vorbelegung: 0 (B#16#0)

Adressbereich:	Ausgang	QB0 … QBn PQB0 … PQBn
	Merker	MB0 … MBn LB0 … LBn
	Daten	DBm.DBB0 … n

Erläuterung: Aktualitäts-Status eines empfangenen Datensegments

Bei jedem laufenden Empfang einer Sequenz wird der momentane Zustand durch das Ausgangs-Byte "DataStatus" angezeigt:

- Wenn der Empfang der Sequenz erfolgreich beendet wurde, dann ist der Datenausgangsbereich "DataOutput" auf dem aktuellen Stand. Das Status-Bit "DataValid" wird auf TRUE gesetzt.
  Wenn beim Sender "SendAll" auf TRUE gesetzt ist, dann ist Datenkonsistenz gegeben.
- Wenn der laufende Empfang einer Sequenz gestört wird, dann ist der Datenausgangsbereich "DataOutput" nicht aktuell und "DataStatus" zeigt einen Fehler an.
  - Das Status-Bit "DataValid" wird auf FALSE gesetzt und es wird ein Eintrag in den Diagnosepuffer geschrieben.
  - Der Status "SequenceState" zeigt den Fehler an (siehe Tabelle).

Tabelle 1-145 Bit-Belegung von "DataStatus"

Bit	Name	Wert	Bedeutung
0	**DataValid**	FALSE	Daten ungültig
		TRUE	Daten gültig
1 ... 5	reserviert	- (FALSE)	Nicht belegt
7, 6	**SequenceState**	0	Kein laufender Empfang oder Empfang ohne Fehler beendet.
		1	Erstes Segment einer Sequenz empfangen
		2	Zweites oder höheres Segment einer Sequenz empfangen
		3	Fehler: • Übertragungssequenz abgebrochen • Überwachungszeit überschritten • Anderer Empfangsfehler

Parameter-Typical Par12D_S / Par12D_FS

Funktion

1 bis 12 Parameterwerte (je 1 Doppelwort) senden und die aktuellen vor Ort gültigen Parameterwerte vom Partner empfangen

Die 1-aus-n-Prüfung wird vom FC Safe durchgeführt.

Hinweis

FC Safe erforderlich

Mit Par12D_S können nur dann Daten übertragen werden, wenn am Ende des zyklischen Programms der FC "Safe" eingebunden ist, siehe Kapitel FC Safe (Seite 2015).

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (DWORD, DINT, REAL), es ist aber auch eine Mischung von anderen Datentypen zulässig, die zusammen wieder ein Doppelwort ergeben, beispielsweise:

- 4 Bytes
- 2 Worte
- 2 Byte + 1 Wort

Beachten Sie hierzu den Hinweis "Wort-Grenzen beachten" unten.

Der zu übertragende Datenbereich wird beim Parameter "ParameterInput" in Form eines Any-Pointers angegeben. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Doppelworten variieren. Der zum Partner übertragene Datenbereich oder die beim Partner lokal eingegebenen Parameterwerte werden von dort zurück übertragen und hier am Parameter "ReturnedParameter" ausgegeben. Auch dieser Ausgabebereich (Any-Pointer) muss in einem Datenbaustein liegen und in seiner Länge identisch sein mit der bei "ParameterInput" angegebenen Länge.

Für "ParameterInput" und "ReturnedParameter" werden normalerweise getrennte Datenbereiche angegeben. Dann ist jederzeit ersichtlich, was zuletzt eingegeben wurde und was aktuell vor Ort gültig ist. Für beide Parameter kann aber auch den gleichen Datenbereich angegeben werden. Beide Bereiche überlappen sich also zu 100% und sind dann immer abgeglichen. Ein Unterschied zwischen dem, was zuletzt eingegeben wurde und dem, was lokal gültig ist, ist in diesem Fall nicht mehr ersichtlich. Wenn die zurückgespiegelten Werte nicht benötigt werden, kann auf die Parametrierung von "ReturnedParameter" verzichtet werden.

Auch wenn für "ParameterInput" und "ReturnedParameter" getrennte Bereiche parametriert werden, kann dafür gesorgt werden, dass der Eingabebereich "ParameterInput" immer mit den zurückgespiegelten Werten von "ReturnedParameter" abgeglichen wird. Dies können Sie entweder von Fall zu Fall manuell über den Eingang "ApplyRemoteParamMan" durchführen oder automatisch, indem Sie den Parameter "ApplyRemoteParamAuto" auf TRUE setzen.

Am Partnerobjekt, das die Parameter empfängt, kann auch eine örtliche Parametereingabe vorgenommen werden. Das Partnerobjekt muss dann am Eingangs-Parameter "Local" auf 'lokal' gestellt werden (siehe Baustein Par12D_R). Der aktuelle Zustand am Eingangs-Parameter "Local" wird vom Partner-Objekt gemeldet und hier am Ausgang "LocalOperation" angezeigt. Solange das Partnerobjekt auf 'lokal' steht, werden dort keine Parameter von anderen Teilnehmern akzeptiert.

Das Senden des bei "ParameterInput" definierten Datenbereichs kann über folgende Parameter ausgelöst werden:

- EnterInput
  Diesen Eingangsparameter sollten Sie nutzen, wenn der bei "ParameterInput" definierte Datenbereich über Hardware (digitale und analoge Eingabebaugruppen) versorgt wird. "EnterInput" ist dann über einen Digitaleingang mit einer Taste in einem Pult oder Tafel zu verbinden. Durch Drücken dieser Taste wird die Übertragung der eingegebenen Werte ausgelöst.
  Es wird der komplette bei "ParameterInput" definierte Datenbereich übertragen.
- ContinuousEnterFunct
  Setzen Sie den Parameter auf TRUE, wenn Sie die Parameter über Software eingeben, bspw. über ein Operator Panel (OP). Es wird eine ständige Änderungskontrolle durchgeführt. Sobald eine Änderung in dem bei "ParameterInput" definierten Datenbereich festgestellt wird, führt dies zu einer Übertragung derjenigen Daten-Doppelworte, die sich gegenüber der letzten Übertragung geändert haben.
  Es werden nur die geänderten Daten übertragen (siehe Hinweis "Geänderte Datenbereiche" unten).

- Release
  Nutzen Sie diesen Eingangsparameter, wenn Sie die Parameter über Software eingeben, bspw. über ein OP. Der Eingang "Release" sollte dann über eine Funktionstaste des OP bedient werden. Wenn am Eingang "Release" 1-Signal erkannt wird, dann wird auf Änderung kontrolliert. Es werden aus dem bei "ParameterInput" angegebenen Datenbereich die Daten-Doppelworte übertragen, die sich gegenüber der letzten Übertragung geändert haben.
  Es werden nur die geänderten Daten übertragen (siehe Hinweis "Geänderte Datenbereiche" unten).
- RetransmitAll
  Nutzen Sie diesen Eingangsparameter, wenn Sie die Parameter über Software eingeben, bspw. über ein OP. Der Eingang "RetransmitAll" sollte dann über eine Funktionstaste des OP bedient werden. Wenn am Eingang "RetransmitAll" 1-Signal erkannt wird, dann wird der bei "ParameterInput" parametrierte Datenbereich ohne Änderungskontrolle übertragen. Es wird der komplette bei "ParameterInput" definierte Datenbereich übertragen.

Hinweis

Geänderte Datenbereiche

Wenn nur der geänderte Datenbereich übertragen wird, dann besteht dieser Bereich aus dem ersten und dem letzten Doppelwort, in denen eine Änderung festgestellt wurde, und aus allen dazwischen liegenden Worten, auch wenn sich diese nicht geändert haben.

Beispiel:
Der einzulesende Bereich hat eine Länge von 10 Doppelworten. Im aktuellen Übertragungsfall wurden Änderungen im zweiten, fünften und achten Doppelwort festgestellt. Übertragen wird der Bereich vom 2. bis zum 8. Doppelwort.

Wort-Grenzen beachten

Wenn nur geänderte Daten übertragen werden und der Datenbereich Werte in Doppelwortformat enthält, dann achten Sie darauf, dass die Doppelwort-Werte tatsächlich in einem der max. 12 Doppelworte des zu erfassenden Datenbereichs liegen.

Eine Verteilung auf zwei aufeinander folgende Daten-Doppelworte könnte dazu führen, dass nur ein Wort des Doppelwort-Werts (High- oder Low-Wort) übertragen wird, weil nur darin sich etwas verändert hat. Das fehlende Wort kann in diesem Fall beim empfangenden Partner zu einem Datenfehler führen (gilt für ST7cc, nicht für eine S7-CPU).

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter:	**EnterInput**	
Deklaration:	INPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Übernahme-Eingang

Über diesen Eingang kann die Übertragung der am Eingang "ParameterInput" anstehenden Parameterwerte flankengesteuert ausgelöst werden.

Ein Wechsel an "EnterInput" wird nur dann berücksichtigt, wenn "ContinuousEnterFunct" = FALSE ist. Wenn diese Bedingung erfüllt ist, werden beim Flankenwechsel 0 → 1 die an "ParameterInput" eingegebenen Parameterwerte übernommen und zur Übertragung gebracht. Es wird nicht auf Änderung geprüft. Es wird immer der komplette bei "ParameterInput" angegebene Datenbereich übertragen.

Diese Art der Übertragungsauslösung eignet sich für die Eingabe über eine Hardware, bspw. über eine Tafel oder ein Pult. Weitere Hinweise zu diesem und verwandten Parametern finden Sie oben im Abschnitt "Funktion".

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Sicherheitsprüfungen:

- Die eingelesenen Parameter werden nur dann zur Übertragung gebracht, wenn bei der 1-aus-n-Prüfung kein Fehler festgestellt wird und wenn der zentrale Freigabe-Merker gesetzt ist. Dieser wird vom FC Safe automatisch nach Ablauf einer dort parametrierten Zeitverzögerung gesetzt (siehe FC Safe, Parameter "InputDelayTime"). Der Eingabebereich wird vom FB erst dann wieder eingelesen, wenn an EnterInput mindestens ein OB1-Zyklus lang 0-Signal erkannt wurde.
- Wenn bei der Hardware-Eingabe ein 1-aus-n-Fehler festgestellt wird, dann werden die eingegebenen Parameter nicht weiter bearbeitet. Neue Parameter werden erst wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU keine Hardware-Eingabe über einen Befehls-, Sollwert- oder Parameter-Baustein erfasst wurde.
  Der FB trägt den erkannten 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B172). Solange der Fehler ansteht, wird der Fehlerzustand über den Ausgang "InputError" des FC Safe zur Anzeige gebracht (siehe FC Safe, Parameter "InputError").

Parameter: **ContinuousEnterFunct**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Kontinuierliche Änderungsprüfung

Über diesen Parameter kann festgelegt werden, ob die am Eingang "ParameterInput" anstehenden Parameterwerte ständig eingelesen und auf Änderung geprüft werden sollen. Die Änderungsprüfung wird durch Vergleich mit den zuletzt übertragenen Werten durchgeführt. Übertragen werden nur die geänderten Werte. Wenn mehrere Änderungen erkannt werden, dann sendet der Baustein den Datenbereich, in dem sich alle geänderten Parameterwerte befinden.

Der geänderte Datenbereich, der übertragen wird, besteht aus dem ersten und dem letzten Doppelwort, in denen eine Änderung festgestellt wurde, sowie allen dazwischen liegenden Worten, auch wenn diese sich nicht geändert haben.

Eine erneute Übertragung der Parameterwerte kann über den Eingang "RetransmitAll" ausgelöst werden (siehe unten), ohne dass die Parametereingabe geändert wird.

Diese Art der Übertragungsauslösung eignet sich für den Fall, dass die Parameterwerte von einer Software in den Eingabebereich ParameterInput eingetragen werden, kann aber auch für die Parametereingabe über ein Operator Panel (OP) genutzt werden.

Weitere Hinweise finden Sie oben im Abschnitt "Funktion".

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Sicherheitsprüfungen:

Die eingelesenen Parameter werden nur dann zur Übertragung gebracht, wenn bei der 1-aus-n-Prüfung kein Fehler festgestellt wird.

- Bei der Hardware-Eingabe (siehe EnterInput) muss ein Leerzyklus erkannt werden, bevor vom Baustein neue Parameterwerte übertragen werden können.
- Bei der Software-Eingabe können in jedem OB1-Zyklus neue Parameterwerte zur Übertragung gebracht werden. Voraussetzung ist, dass in diesem Zyklus keine weitere Software-Eingabe bei einem anderen Baustein anliegt. Anderenfalls würde auf 1-aus-n-Fehler erkannt werden. Wenn bei der Software-Eingabe ein 1-aus-n-Fehler festgestellt wird, dann werden die eingegebenen Parameter nicht weiter bearbeitet. Der FB trägt den erkannten 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B172). Neue Parameter werden erst wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU keine Software-Eingabe bei einem Befehls-, Sollwert- oder Parameter-Bausteinen erfasst wurde.

Parameter: **ApplyRemoteParamAuto**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Erläuterung: Automatischer Abgleich des Eingabebereichs mit dem Rückspiegelbereich

Mit dem Parameter werden automatisch alle Parameterwerte aus dem Rückspiegelbereich "ReturnedParameter" in den Eingabebereich "ParameterInput" kopiert.

Zusätzlich werden die zurückgespiegelten Parameterwerte in das Sendefach des Kommunikations-DB geschrieben.

Der automatische Abgleich wird immer dann durchgeführt, wenn neue Daten vom Partnerobjekt (Par12D_R) empfangen wurden.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **ParameterInput**

Deklaration: INPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:

```
P#DB20.DBX 100.0 DWORD 4
```

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Parameter-Eingangsbereich

Der ANY-Pointer adressiert den Datenbereich, in dem die zu erfassenden Parameterwerte bereitstehen. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Daten-Doppelworten variieren.

Zu den Inhalten und Formaten siehe Abschnitt "Funktion" oben.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Die Art und Weise wie die an "ParameterInput" anstehenden Parameter bearbeitet werden, ist davon abhängig, ob es sich um eine Hardware- oder eine Software-Eingabe handelt und wie die Übertragung dieses Datenbereichs ausgelöst wird. Weitere Information hierzu finden Sie im Abschnitt "Funktion" oben.

Parameter: **ReturnedParameter**

Deklaration: OUTPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:
`P#DB20.DBX 100.0 DWORD 4`

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, das der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Parameter-Ausgabebereich

Das Partnerobjekt, das die Parameterwerte empfängt, meldet die dort gültigen Parameterwerte zurück. Diese Werte werden hier am Ausgang "ReturnedParameter" zur Anzeige gebracht. Wenn das Partnerobjekt auf 'lokal' steht und dort eine neue Eingabe gemacht wird, dann werden die vor Ort geänderten Parameter hier am "ReturnedParameter" angezeigt.

Der ANY-Pointer definiert den Datenbereich in dem die empfangenen Parameterwerte ausgegeben werden. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Daten-Doppelworte variieren. Die Länge muss identisch sein mit der bei ParameterInput parametrierten Länge.

Nach Anlauf der eigenen oder der Partner-CPU oder nach Wiederkehr der Verbindung wird über eine automatische Generalabfrage sichergestellt, dass an "ReturnedParameter" die aktuell vor Ort gültigen Parameter angezeigt werden.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Bei falscher Parametrierung (Datenbereich kein Datenbaustein, Länge größer 12 oder Länge abweichend von der bei ParameterInput parametrierten Länge) wird eine entsprechende Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der FB wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben worden ist.

Parameter: **LocalOperation**

Deklaration: OUTPUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Rückmeldung vom Partner-Objekt: Objekt steht vor Ort auf lokale Bedienung.

Am Partnerobjekt, das die Parameter empfängt, kann auch eine örtliche Sollwerteingabe vorgenommen werden. Das Partnerobjekt Par12D_R muss dann am Eingangs-Parameter "Local" auf 'lokal' gestellt werden. Der aktuelle Zustand am Eingangs-Parameter "Local" wird vom Partnerobjekt gemeldet und hier am Ausgang "LocalOperation" angezeigt.

Nach Anlauf der eigenen oder der Partner-CPU oder nach Wiederkehr der Verbindung wird automatisch über eine Generalabfrage sichergestellt, dass an "LocalOperation" der vor Ort aktuell gültige Zustand angezeigt wird.

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Jedes Mal, wenn der Baustein neue Daten empfangen und an den Ausgang "ReturnedSetpoint" oder "LocalOperation" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt.

Parameter: **Release**

Deklaration: IN_OUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!

Erläuterung: Trigger-Eingang zum Senden der aktuell anstehenden Parameterwerte

Diesen Eingangsparameter können sie nutzen, wenn der Parameter über Software eingegeben wird, bspw. über ein Operator Panel (OP). "Release" sollte dann über eine Funktionstaste des OP bedient werden. Sie können dann am OP zunächst mehrere Parameter eingeben. Erst wenn Sie die Release-Funktionstaste betätigen, werden die Parameter übertragen.

Nur beim Signal 1 am Eingang Release wird eine Änderungskontrolle durchgeführt. Diejenigen Daten-Doppelworte aus dem bei ParameterInput parametrierten Datenbereich, die sich gegenüber der letzten Übertragung geändert haben, werden übertragen.

Wenn Sie immer den gesamten bei "ParameterInput" angegebenen Datenbereich übertragen möchten und nicht nur die geänderten Parameterwerte, dann sollten Sie anstelle von "Release" den Eingangsparameter "RetransmitAll" nutzen.

Der Eingang "Release" wird automatisch zurückgesetzt. Deshalb sollten nur Merker oder Datenbits als Eingang angegeben werden. Bei einem Digitaleingang würde das automatische Rücksetzen nicht funktionieren.

Sicherheitsprüfungen: Es werden die gleichen Sicherheitsprüfungen vorgenommen wie bei ContinuousEnterFunct, siehe oben.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **RetransmitAll**

Deklaration: IN_OUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: TRUE

Adressbereich:	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!

Erläuterung: Trigger-Eingang zum erneuten Senden des gesamten bei "ParameterInput" angegebenen Datenbereichs

Diesen Eingangsparameter können sie nutzen, wenn der Parameter über Software eingegeben wird, bspw. über ein Operator Panel (OP). "RetransmitAll" sollte dann über eine Funktionstaste des OP bedient werden. Wenn das Signal 1 am Eingang "RetransmitAll" erkannt wird, wird der gesamte bei "ParameterInput" angegebene Datenbereich übertragen. Eine Änderungskontrolle wird nicht durchgeführt.

Der Eingang "RetransmitAll" wird automatisch zurückgesetzt. Deshalb sollten nur Merker oder Datenbits als Eingang angegeben werden. Bei einem Digitaleingang würde das automatische Rücksetzen nicht funktionieren. Da keine Änderungskontrolle durchgeführt wird, wäre die Folge eine ständige Übertragung aller Parameterwerte, solange der Eingang 1-Signal hat.

"RetransmitAll" kann als zusätzliche Option zu "Release" oder "ContinuousEnterFunct" verwendet werden, wenn neue Parameterwerte eingegeben wurden und diese nicht an den Partner übertragen werden konnten (Verbindungsstörung, Partnerobjekt steht auf 'lokal'). In diesem Fall können Sie über "RetransmitAll" eine nochmalige Übertragung des gesamten bei "ParameterInput" angegebenen Datenbereichs auslösen. Alle Änderungen, die zuvor eingegeben wurden und beim Partner noch nicht vorhanden sind, sind konsistent enthalten.

"RetransmitAll" kann auch als alleiniger Übertragungsauslöser genutzt werden, wenn immer alle und nicht nur die jeweils geänderten Eingaben übertragen werden sollen. Man kann in diesem Fall "RetransmitAll" anstelle von "Release" nutzen, bei dem nur die geänderten Parameterwerte übertragen werden.

Sicherheitsprüfungen: Es werden die gleichen Sicherheitsprüfungen vorgenommen wie bei "ContinuousEnterFunct", siehe oben.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **ApplyRemoteParamMan**

Deklaration: IN_OUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!
Erläuterung:	Trigger-Eingang zum Abgleich des Eingabebereichs mit dem Rückspiegelbereich Der Eingang löst einmalig einen Abgleich des Eingabebereichs ParameterInput mit dem Rückspiegelbereich "ReturnedParameter" aus. Dabei werden alle Parameterwerte aus dem Rückspiegelbereich "ReturnedParameter" in den Eingabebereich "ParameterInput" kopiert. Zusätzlich wird das Sendefach des Kommunikations-DB mit den zurückgespiegelten Parameterwerten abgeglichen. "ApplyRemoteParamMan" wird automatisch zurückgesetzt. Deshalb sollten nur Merker oder Datenbits als Eingang angegeben werden. Bei einem Digitaleingang würde das automatische Rücksetzen nicht funktionieren. Die Folge wäre einen ständigen Abgleich, solange der Parameter 1-Signal hat. Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter-Typical Par12D_R

Funktion

1 bis12 Parameterwerte (je 1 Doppelwort) empfangen oder lokal eingeben und die aktuellen vor Ort gültigen Parameterwerte an Partner zurücksenden

Der Inhalt pro Doppelwort kann ein Wert in Doppelwortformat sein (DWORD, DINT, REAL), es ist aber auch eine Mischung von anderen Datentypen zulässig, die zusammen wieder ein Doppelwort ergeben, beispielsweise:

- 4 Bytes
- 2 Worte
- 2 Bytes + 1 Wort

Beachten Sie hierzu den Hinweis "Wort-Grenzen beachten" unten.

Der zu übertragende Datenbereich wird beim Parameter "ParameterOutput" in Form eines Any-Pointers angegeben. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Doppelworten variieren.

Der Baustein bietet aber auch die Möglichkeit, die Parameterwerte lokal einzugeben. Der Eingabebereich dazu wird beim Parameter "LocalParameterInput" ebenfalls als Any-Pointer angegeben, muss in einem Datenbaustein liegen und in seiner Länge identisch sein mit der bei Parameter "ParameterOutput" angegebenen Länge.

Der Baustein bearbeitet nur den geänderten Datenbereich. Im Zuge einer General- oder Einzelabfrage wird dagegen der komplette Parametersatz übertragen bzw. zurückgespiegelt.

Das stoßfreie Umschalten zwischen den Betriebsarten "Local" und "Remote" ist gewährleistet.

Hinweis

Geänderte Datenbereiche

Der geänderte Datenbereich besteht aus dem ersten und dem letzten Doppelwort, in denen eine Änderung festgestellt wurde, und aus allen dazwischen liegenden Worten, auch wenn sich diese nicht geändert haben.

Beispiel:
Der einzulesende Bereich hat eine Länge von 10 Doppelworten. Im aktuellen Übertragungsfall wurden Änderungen im zweiten, fünften und achten Doppelwort festgestellt. Übertragen wird der Bereich vom 2. bis zum 8. Doppelwort.

Wort-Grenzen beachten

Wenn nur geänderte Daten übertragen werden und der Datenbereich Werte in Doppelwortformat enthält, dann achten Sie darauf, dass die Doppelwort-Werte tatsächlich in einem der max. 12 Doppelworte des zu erfassenden Datenbereichs liegen.

Eine Verteilung auf zwei aufeinander folgende Daten-Doppelworte könnte dazu führen, dass nur ein Wort des Doppelwort-Werts (High- oder Low-Wort) übertragen wird, weil nur darin sich etwas verändert hat. Das fehlende Wort kann in diesem Fall beim empfangenden Partner zu einem Datenfehler führen (gilt für ST7cc, nicht für eine S7-CPU).

Hinweis

DB mit Standardzugriff

Der Baustein hat Parameter vom Typ "ANY". Lassen Sie daher in den Eigenschaften des DB das Attribut "Optimierter Bausteinzugriff" deaktiviert.

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
ImageMemory
Conditional
Unconditional
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **Local**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Lokale Parametereingabe freigegeben

Freigabe der lokalen Parametereingabe des bei "LocalParameterInput" angegebenen Datenbereichs

Ein vom Partner (Zentrale) gesendeter Sollwert wird vom Objekt nicht angenommen, solange "Local" = TRUE ist.

Der aktuelle Zustand des Eingangs "Local" wird zum Partner übertragen.

Stoßfreie Umschaltung:

- Wenn von "Local" = 0 auf 1 geschaltet wird, werden die zuletzt am Ausgang "ParameterOutput" ausgegebenen Werte solange beibehalten, bis über den lokalen Eingabebereich "LocalParameterInput" neue Parameterwerte eingegeben werden.
- Wenn von "Local" = 1 auf 0 zurückgeschaltet wird, werden die zuletzt am Ausgang "ParameterOutput" ausgegebenen Werte solange beibehalten, bis der Baustein vom Partner neue Parameterwerte empfängt.

Beachten Sie den Hinweis beim Parameter "ContinuousEnterFunct".

Sonderfall:

Sie können die Parameterwerte bei der lokalen Eingabe auch direkt in den bei "ParameterOutput" angegebenen Ausgabebereich eintragen. Sie verzichten dann entweder auf die Angabe eines Eingabebereichs bei "LocalParameterInput" oder Sie geben sowohl für "LocalParameterInput" als auch "ParameterOutput" den gleichen Datenbereich an.

Diese Art der Parametereingabe kann über den Eingang "Local" nicht verhindert werden. Unabhängig vom Zustand des Parameters "Local" werden die in den Ausgabebereich eingetragenen Werte sofort vom Baustein an den Partner gesendet.

Lokale Parametereingaben können damit unabhängig vom Status des Eingangs "Local" vorgenommen werden. "Local" hat nur noch einen Einfluss auf die Annahme der Parameter, die vom Partner gesendet werden:

- Local = 0
  Die vom Partner gesendeten Parameter werden angenommen und an den Datenbereich "ParameterOutput" ausgegeben.
- Local = 1
  Vom Partner gesendete Parameter werden abgewiesen.

Die Eingänge "Release" und "ContinuousEnterFunct" haben in dem beschriebenen Sonderfall keine Funktion.

Eine Statusänderung des Parameters "Local" wird vom Modul immer im Sendepufferprinzip übertragen, auch wenn der Parameter "ImageMemory" = TRUE ist. Damit wird sichergestellt, dass beim Partner der optional mögliche Abgleich zwischen Ein- und Ausgabebereich immer korrekt durchgeführt wird (siehe Par12D_S, Parameter "ApplyRemoteParamMan" und "ApplyRemoteParamAuto").

Parameter: **ContinuousEnterFunct**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Vorbelegung: FALSE
Erläuterung: Kontinuierliche lokale Parametererfassung

Über diesen Parameter legen Sie fest, ob die im Eingangsbereich "LocalParameterInput" anstehenden Werte ständig eingelesen und auf Änderung geprüft werden. Die Änderung wird durch Vergleich mit den aktuellen Werten am Ausgang "ParameterOutput" geprüft.

Änderungen im Eingangsbereich werden sofort in den Ausgangsbereich kopiert und zum Partnerobjekt übertragen. Übertragen werden nur die geänderten Werte. Wenn es mehrere Änderungen gibt, dann sendet der Baustein denjenigen Datenbereich, in dem sich alle geänderten Parameterwerte befinden.

Die Einstellung "ContinuousEnterFunct" = TRUE wird nur dann wirksam, wenn folgende Voraussetzungen erfüllt sind:

- Bei "LocalParameterInput" ist ein Eingabebereich definiert und dieser ist nicht identisch mit dem bei "ParameterOutput" abgegebenen Ausgabebereich.
  und
- Am Eingang "Local" steht 1-Signal an (= TRUE).

In der Einstellung "ContinuousEnterFunct" = TRUE werden beim Erkennen des Signals 1 am Eingang "Local" die bei "LocalparameterInput" anstehenden Werte sofort übernommen und an den Ausgang "ParameterOutput" ausgegeben. Bedingung ist, dass sich die lokalen Eingangswerte zu diesem Zeitpunkt von den aktuell ausgegebenen Werten unterscheiden.

Diese Art der Erfassung von lokalen Werten eignet sich für den Fall, dass die Werte von einer Software in den Eingabebereich "LocalParameterInput" eingetragen werden.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **LocalParameterInput**
Deklaration: INPUT
Datentyp: ANY
Wertebereich: Siehe Adressbereich
Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:
`P#DB20.DBX 100.0 DWORD 4`

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, dass der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Lokaler Parameter-Eingangsbereich

Der ANY-Pointer adressiert den Datenbereich, in dem die zu erfassenden Parameterwerte bereitstehen. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Daten-Doppelworte variieren. Die Länge muss identisch sein mit der bei "ParameterOutput" angegebenen Länge.

Zu Inhalt und Formaten siehe Abschnitt "Funktion" oben.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **ParameterOutput**

Deklaration: INPUT

Datentyp: ANY

Wertebereich: Siehe Adressbereich

Vorbelegung: P#P 0.0 VOID 0
(Null-Pointer)

Adressbereich: P#DBxx.DBX yy.0 DWORD zz

- xx: Datenbaustein-Nummer 1...32767
- yy: Byte-Nummer
- zz: Anzahl Doppelworte 1...12 ab Byte-Nummer yy

Beispiel:

```
P#DB20.DBX 100.0 DWORD 4
```

Achten Sie bei der Eingabe des Pointers auf die Punkte und die Leerzeichen!

Beachten Sie, das der vorbelegte Wert (Null-Pointer) nicht zulässig ist. Es muss ein Pointer mit einer realen Adresse angegeben werden.

Erläuterung: Parameter-Ausgabebereich

Der ANY Pointer adressiert den Datenbereich, in dem die lokal eingegebenen oder die vom Partner empfangenen Parameterwerte ausgegeben werden. Dieser Datenbereich muss in einem Datenbaustein liegen und kann in der Länge zwischen 1 und 12 Doppelworten variieren.

Zu Inhalt und Formaten siehe Abschnitt "Funktion" oben.

Par12D_R legt die empfangenen Daten ohne weitere Aufbereitung in dem bei "ParameterOutput" angegebenen Datenbereich ab. Sie müssen die empfangenen Daten über das Anwenderprogramm auswerten und verarbeiten.

Wenn vom Partnerobjekt Par12D_S nur geänderte Daten übertragen werden, ist es möglich, dass nur ein Teil des Daten-Ausgangsbereichs neu beschrieben wird. Dies ist derjenige Bereich, in dem auf der Erfassungsseite Änderungen festgestellt wurden.

Bei falscher Parametrierung (Null-Pointer, Länge > 12, Datenbereich kein DB) wird eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B114, [Info2/3] = 11). Die CPU geht nicht in STOP. Der Baustein wird danach aber solange nicht mehr bearbeitet, bis der Fehler behoben ist.

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Jedes Mal, wenn der Baustein neue Parameterwerte vom Partnerobjekt empfangen und an das Ausgabefeld "ParameterOutput" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt. Dies gilt auch für den Fall einer neuen lokalen Eingabe im Zustand "Local" = 1.

Parameter: **Release**

Deklaration: IN_OUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: FALSE

Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!

Erläuterung: Eingang zur Übernahme der lokalen Parametereingabe

Über diesen Eingang kann die Übernahme der am Parametereingang "LocalParameterInput" anstehenden Parameterwerte flankengesteuert ausgelöst werden.

Ein Wechsel von 0 nach 1 am Eingang "Release" wird nur dann berücksichtigt, wenn folgende Voraussetzungen erfüllt sind:

- Beim Parameter "LocalParameterInput" ist ein Eingabebereich angegeben und dieser ist nicht identisch mit dem bei "ParameterOutput" angegebenen Ausgabebereich.
  und
- Der Eingang "Local" steht auf TRUE.

Sie können "Release" für Parametereingaben über Software nutzen, bspw. über ein Operator Panel (OP). Der Eingang "Release" sollte dann über eine Funktionstaste des OP bedient werden. Sie können dann am OP zunächst mehrere Parameter eingeben. Erst wenn Sie die Release-Funktionstaste betätigen, werden die Parameterwerte eingelesen und auf Änderung geprüft.

Die Änderung wird durch Vergleich mit den aktuellen Parameterwerten am Ausgang "ParameterOutput" geprüft. Änderungen im Eingangsbereich werden dann sofort in den Ausgangsbereich kopiert und zum Partnerobjekt übertragen.

Übertragen werden nur die geänderten Werte. Wenn es mehrere Änderungen gibt, dann sendet der Baustein denjenigen Datenbereich, in dem sich alle geänderten Parameterwerte befinden.

Der Eingang "Release" wird automatisch zurückgesetzt. Als Eingang kann außer einem Merker oder Datenbit auch ein Digitaleingang angegeben werden. Bei einem Digitaleingang würde das automatische Rücksetzen nicht funktionieren. Negative Auswirkungen hat dies aber nicht. Die Erfassung über "Release" wird flankengesteuert ausgelöst, das heißt nur einmal.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

FC Safe

Funktion

Der FC Safe sorgt für eine gesicherte Eingabe von Befehlen und Sollwerten durch Anwendung der 1-aus-n-Kontrolle.

Der FC Safe prüft Eingaben für folgende Datenpunkt-Typicals:

- Cmd01B_S
- Set01W_S
- Par12D_S

Wenn eine Eingabe ansteht, dann prüft der FC, ob im aktuellen OB1-Zyklus nur eine Eingabe ansteht, und erteilt daraufhin dem einlesenden Baustein die Freigabe. Sobald zwei (oder mehr) Eingaben innerhalb eines OB1-Zyklus anstehen, werden beide Eingaben ignoriert.

In jeder CPU, in der Befehle und/oder Sollwerte erfasst werden, muss der FC Safe im OB1 einmalig als Abschluss aller Befehls- und Sollwert-FBs aufgerufen werden.

Der FC hat eine separate Überwachung für Eingaben über verschiedene Systemspeicherbereiche. Sie werden in Hardware-Eingaben und Software-Eingaben unterteilt:

- Hardware-Eingaben
  - Eingabebaugruppen (E)
  - Peripheriebereich Eingänge (PE)
- Software-Eingaben
  - Merker (M)
  - Datenbausteine (DB)
  - Lokaldaten (L)
  - Operator Panels

Die beiden Eingabearten "Hardware-Eingaben" und "Software-Eingaben" werden vom Baustein separat geprüft. Der FC erteilt die Freigabe für Hardware- und Software-Eingaben getrennt. Pro Eingabeart darf nur eine einzige Befehls- oder Sollwerteingabe erkannt werden.

Wenn gleichzeitig eine einzige Hardware-Eingabe und eine einzige Software-Eingabe anstehen, werden beide Eingaben freigegeben.

Für Hardware-Eingaben gilt folgende zusätzliche Bedingung: Die Eingabe muss für die Dauer von "InputDelayTime" konstant anstehen. Nur wenn ein eingegebener Befehl oder Sollwert für diese Zeit unverändert ansteht und während dieser Zeit keine weitere Befehls- oder Sollwerteingabe erkannt wird, wird die Bausteinbearbeitung freigegeben.

Die eigentliche Zusammenstellung des Befehls- bzw. Sollwert-Telegramms wird von demjenigen Baustein durchgeführt, der den Befehl bzw. Sollwert eingelesen hat.

Für die Hardware-Eingabe bietet der FC Safe noch die folgenden zwei Anzeigebits an:

- InputOK
  Anzeige der Freigabe von Hardware-Befehlen und Sollwerten
- InputError
  Anzeige von Eingabefehlern bei Hardware-Befehlen und Sollwerten

FC Safe zeigt einen Befehlsausgabefehler, der in einer Station erkannt wurde, über den folgenden Ausgang an:

- GlobalCmdOutputError
  Sammelmeldung: Befehlsausgabefehler in einer Station

Parameter

Name:	**InputDelayTime**
Deklaration:	INPUT

Datentyp: INT

Wertebereich:
- 0
  Wert für nicht benötigten Parameter
- 1 ... 32000
  Wertebereich für Verzögerungszeit

Erläuterung: Verzögerungszeit in ms für Befehle und Sollwerte, die über Hardware eingegeben werden.

Empfohlen wird eine Verzögerungszeit von mindestens 1000 ms.

Name: **MaxInputTime**

Deklaration: INPUT

Datentyp: INT

Wertebereich:
- 0
  Wert für nicht benötigten Parameter
- 1 ... 32000
  Wertebereich für Überwachungszeit

Erläuterung: Überwachungszeit in s für Befehle und Sollwerte, die über Hardware eingegeben werden.

Empfohlen wird eine Überwachungszeit von mindestens 30 s.

Wenn Sie den Parameter nicht benötigen, dann geben Sie 0 (null) an.

Name: **ResetError**

Deklaration: INPUT

Datentyp: BOOL

Wertebereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Eingang zum Rücksetzen des Ausgangs GlobalCmdOutputError.

Wenn Sie den Parameter nicht benötigen, dann geben Sie ein Merker- oder Datenbit an, das immer 0-Signal führt.

Name: **InputOK**

Deklaration:
- S7-300
- S7-1500
- OUTPUT
- IN_OUT

Datentyp: BOOL

Wertebereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Anzeige der Freigabe von Hardware-Befehlen und Sollwerten

Führt 1-Signal, sobald die Freigabe für die aktuelle Eingabe erteilt wurde, das heißt, wenn der Hardware-Befehl oder der Sollwert korrekt eingegeben wurde.

Das Anzeige-Bit erlischt mit der Rücknahme der Eingabe, d. h. sobald die Befehlstaste losgelassen wird bzw. sobald die Sollwert-Eingabetaste am Eingang "EnterInput" losgelassen wird.

Wenn Sie den Parameter nicht benötigen, dann geben Sie ein Merker- oder Datenbit im Speicherbereich der Lokaldaten an.

Name: **InputError**

Deklaration:	• S7-300	• OUTPUT
	• S7-1500	• IN_OUT
Datentyp:	BOOL	
Wertebereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Anzeige von Eingabefehlern bei Hardware-Befehlen und Sollwerten

Der Ausgang führt 1-Signal, sobald innerhalb der Überwachungszeit "MaxInputTime" einer der folgenden Hardware-Eingabefehler erkannt wird:

- Zwei oder mehr Befehls- und/oder Sollwerteingaben wurden zur gleichen Zeit festgestellt.
- Bei einem der Eingänge wurde über längere Zeit 1-Signal erkannt (bspw. wenn der Eingang defekt ist).

Wenn Sie den Parameter nicht benötigen, dann geben Sie ein Merker- oder Datenbit im Speicherbereich der Lokaldaten an.

Name: **GlobalCmdOutputError**

Deklaration:	• S7-300	• OUTPUT
	• S7-1500	• IN_OUT
Datentyp:	BOOL	

Wertebereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Sammelmeldung: In einer Station wurde ein Befehlsausgabefehler festgestellt.

Ein Befehlsausgabefehler gibt es auf der Empfangsseite nur in folgenden Fällen:

- Der Inhalt der beiden Befehls-Bytes im empfangenen Telegramm ist nicht identisch.
- Mehr als ein Bit im Befehls-Byte ist gesetzt (1-aus-8-Fehler).

Wenn ein derartiger Fehler erkannt wird, dann wird dies von der Station mittels einem Org.-Telegramm an den Teilnehmer geschickt, der die Befehle gesendet hat. Der FC Safe beim sendenden Teilnehmer zeigt die Störung am Ausgang "GlobalCmdOutputError" an.

Bei erkanntem Fehler bleibt der Ausgang solange auf 1-Signal stehen, bis Sie über den Eingang "ResetError" das Rücksetzen des Sammelsignals anfordern.

Wenn Sie den Parameter nicht benötigen, dann geben Sie ein Merker- oder Datenbit im Speicherbereich der Lokaldaten an.

Sollwert-Typical Set01W_S / Set01W_FS

Funktion

1 Sollwert als 16-Bit-Wert senden und vor-Ort-Sollwert vom Partner empfangen

Die 1-aus-n-Prüfung wird vom FC Safe durchgeführt.

Hinweis

FC Safe erforderlich

Mit Set01W_S können nur dann Daten übertragen werden, wenn am Ende des zyklischen Programms der Baustein FC Safe eingebunden ist, siehe Kapitel FC Safe (Seite 2015).

Parameter

Parameter: **PartnerNo**
PartnerObjectNo
Enabled
Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter: **EnterInput**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE

Vorbelegung:	FALSE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Übernahme-Eingang für Hardware-Sollwert

Über diesen Eingang kann die Übernahme des am Sollwerteingang "SetpointInput" anstehenden Sollwerts flankengesteuert ausgelöst werden.

Ein Wechsel am Eingang "EnterInput" wird nur dann berücksichtigt, wenn "ContinuousEnterFunct" = FALSE ist. Wenn diese Bedingung erfüllt ist, wird beim Flankenwechsel 0 → 1 der an "SetpointInput" eingegebene Sollwert übernommen und zur Übertragung gebracht. Dies gilt auch dann, wenn der neu eingegebene Sollwert identisch mit dem zuletzt übertragenen Sollwert ist.

Diese Art der Sollwertübernahme eignet sich für die Eingabe über Hardware, bspw. über eine Tafel oder ein Pult.

Sie kann auch für die Sollwerteingabe über ein Operator Panel (OP) genutzt werden. In diesem Fall muss vom OP die Übernahme durch eine Funktionstaste ausgelöst werden.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **ContinuousEnterFunct**

Deklaration: INPUT

Datentyp: BOOL

Wertebereich: TRUE / FALSE

Vorbelegung: TRUE

Erläuterung: Kontinuierliche Sollwertübernahme für Software-Sollwert

Über diesen Parameter legen Sie fest, ob der am Sollwerteingang "SetpointInput" anstehende Sollwert ständig eingelesen und auf Änderung geprüft werden soll. Auf Änderung wird durch Vergleich mit dem zuletzt übertragenen Sollwert geprüft.

Diese Art der Sollwertübernahme eignet sich für die Eingabe über eine entsprechende Software. Sie kann aber auch für die Sollwerteingabe über ein Operator Panel (OP) genutzt werden, falls beim OP keine separate Funktionstaste vorgesehen ist, über welche die Übernahme ausgelöst werden kann.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter: **SetpointInput**

Deklaration: INPUT

Datentyp: WORD

Wertebereich: Siehe Adressbereich

Vorbelegung: 0 (W#16#0)

Adressbereich:	Eingangs-Worte	EW0 ... Ewn PEW0 ... PEWn
	Merker-Worte	MW0 ... MWn LW0 ... LWn
	Daten-Worte	DBm.DBW0 ... n

Erläuterung: Sollwert-Eingang

Die Art und Weise wie der an SetpointInput anstehende Sollwert bearbeitet wird, ist davon abhängig, ob es sich um eine Hardware- oder eine Software-Eingabe handelt. Welche Eingabe-Art vorliegt, das bestimmen Sie über den Parameter "ContinuousEnterFunct":

- **ContinuousEnterFunct = FALSE (Hardware-Eingabe)**
  Ein an "SetpointInput" anstehender Sollwert wird nur eingelesen, solange an "EnterInput" das Signal 1 ansteht. Der eingelesene Sollwert wird anschließend zur Übertragung gebracht, wenn bei der 1-aus-n-Prüfung kein Fehler festgestellt wird und wenn der zentrale Freigabe-Merker gesetzt ist. Dieser wird vom FC Safe automatisch nach Ablauf einer dort parametrierten Zeitverzögerung gesetzt (siehe FC Safe, Parameter "InputDelayTime").
  Ein weiterer Sollwert wird vom FB erst dann wieder eingelesen, wenn an EnterInput mindestens ein OB1-Zyklus lang das Signal 0 erkannt wurde.
  Wenn bei der Hardware-Eingabe ein 1-aus-n-Fehler festgestellt wird, dann wird der eingegebene Sollwert nicht weiter bearbeitet. Ein neuer Sollwert wird erst dann wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU keine Hardware-Eingabe über einen Befehls-, Sollwert- oder Parameter-Baustein erfasst wurde.
  Der FB trägt den erkannten 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B172). Solange der Fehler ansteht, wird der Fehlerzustand über den Ausgang "InputError" des FC Safe zur Anzeige gebracht (siehe FC Safe, Parameter "InputError").
- **ContinuousEnterFunct = TRUE (Software-Eingabe)**
  Ein an SetpointInput anstehender Sollwert wird kontinuierlich eingelesen und auf Änderung geprüft. Auf Änderung wird durch Vergleich mit dem zuletzt übertragenen Sollwert geprüft. Bei jeder Änderung wird der Sollwert sofort übertragen, wenn bei der 1-aus-n-Prüfung kein Fehler festgestellt wird.
  - Bei der Hardware-Eingabe (siehe EnterInput) muss ein Leerzyklus erkannt werden, bevor vom Baustein ein neuer Sollwert übertragen werden kann.
  - Bei der Software-Eingabe kann in jedem OB1-Zyklus ein neuer Sollwert zur Übertragung gebracht werden. Voraussetzung ist, dass in diesem Zyklus keine weitere Software-Eingabe bei einem anderen Baustein anliegt. Anderenfalls würde auf 1-aus-n-Fehler erkannt werden.
    Eine erneute Übertragung des Software-Sollwerts kann über den Eingang "SendSoftSetpoint" ausgelöst werden, ohne dass die Sollwerteingabe geändert wird (siehe unten).
    Wenn bei der Software-Eingabe ein 1-aus-n-Fehler festgestellt wird, dann wird der eingegebene Sollwert nicht weiter bearbeitet.
    Ein neuer Sollwert wird erst wieder eingelesen, wenn vorher ein OB1-Zyklus lang in der CPU keine Software-Eingabe (Befehl oder Sollwert) erfasst wurde. Der Baustein trägt erkannte 1-aus-n-Fehler in den Diagnosepuffer ein (Ereignis-ID B172). Außerdem werden im zentralen Datenbaustein BasicData entsprechende Fehlerbits gesetzt, die von der Software abgefragt werden können. Weitere Details hierzu finden Sie in der Beschreibung des FC Safe.

Parameter:	**ReturnedSetpoint**	
Deklaration:	OUTPUT	
Datentyp:	WORD	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0 (W#16#0)	
Adressbereich:	Ausgangs-Worte	AW0 ... AWn PAW0 ... PAWn
	Merker-Worte	MW0 ... MWn LW0 ... LWn
	Daten-Worte	DBm.DBW0 ... n
Erläuterung:	Ausgang für zurückgespiegelten Sollwert	

Das Partnerobjekt, das den Sollwert empfängt, meldet den dort aktuell gültigen Sollwert zurück. Dieser Wert wird hier am Ausgang "ReturnedSetpoint" zur Anzeige gebracht.

Wenn das Partnerobjekt auf 'lokal' steht und dort eine neue Eingabe gemacht wird, dann wird der vor Ort geänderte Sollwert hier an "ReturnedSetpoint" angezeigt.

Nach Anlauf der eigenen oder der Partner-CPU oder nach Wiederkehr der Verbindung wird über eine automatisch ausgelöste Generalabfrage sichergestellt, dass an "ReturnedSetpoint" der aktuell vor Ort gültige Sollwert angezeigt wird.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter:	**LocalOperation**	
Deklaration:	OUTPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Ausgang	A 0.0 ... A n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7
Erläuterung:	Rückmeldung vom Partnerobjekt: Das Objekt steht vor Ort auf lokale Bedienung	

Am Partnerobjekt, das den Sollwert empfängt, kann auch eine örtliche Sollwerteingabe vorgenommen werden. Das Partnerobjekt Set01W_R muss dann am Eingangs-Parameter "Local" auf 'lokal' gestellt werden. Der aktuelle Zustand am Eingangs-Parameter "Local" wird vom Partnerobjekt gemeldet und hier am Ausgang "LocalOperation" angezeigt.

Nach Anlauf der eigenen oder der Partner-CPU oder nach Wiederkehr der Verbindung wird automatisch über eine Generalabfrage sichergestellt, dass an "LocalOperation" der vor Ort aktuell gültige Zustand angezeigt wird.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter:	**NewData**	
Erläuterung:	Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945). Jedes Mal, wenn der Baustein neue Daten empfangen und an den Ausgang "ReturnedSetpoint" oder "LocalOperation" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt.	

Parameter:	**SendSoftSetpoint**	
Deklaration:	IN_OUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7
	Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!	
Erläuterung:	Trigger-Eingang zum erneuten Senden des letzten Software-Sollwerts Zu weiteren Details siehe Parameter "SetpointInput". Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.	

Sollwert-Typical Set01W_R

Funktion

1 Sollwert in der Station als 16-Bit-Wert empfangen oder lokal eingeben und lokalen vor-Ort-Sollwert an Zentrale zurücksenden

Parameter

Parameter:	**PartnerNo** **PartnerObjectNo** **Enabled** **ImageMemory** **Conditional** **Unconditional** **TimeStamp** Zur Beschreibung siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Parameter:	**Local**
Deklaration:	INPUT

Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Freigabe der lokalen Sollwerteingabe des bei "LocalSetpointInput" angegebenen Datenbereichs

Ein vom Partner (Zentrale) gesendeter Sollwert wird vom Objekt nicht angenommen, solange "Local" = TRUE ist.

Der aktuelle Zustand des Eingangs "Local" wird zum Partner übertragen, und zwar zusammen mit einer Kopie des Sollwerts, der aktuell am Ausgang "SetpointOutput" ausgegeben wird (Sollwertspiegelung).

Stoßfreie Umschaltung:

- Wenn von "Local" = 0 auf 1 geschaltet wird, wird der zuletzt am Ausgang "SetpointOutput" ausgegebene Werte solange beibehalten, bis über den lokalen Eingabebereich "LocalSetpointInput" ein neuer Sollwert eingegeben wird.
- Wenn von "Local" = 1 auf 0 zurückgeschaltet wird, wird der zuletzt am Ausgang "SetpointOutput" ausgegebene Wert solange beibehalten, bis der Baustein vom Partner einen neuen Sollwert empfängt.

Beachten Sie den Hinweis beim Parameter "ContinuousEnterFunct".

Parameter:	**EnterInput**	
Deklaration:	INPUT	
Datentyp:	BOOL	
Wertebereich:	TRUE / FALSE	
Vorbelegung:	FALSE	
Adressbereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung: Übernahme-Eingang für die lokale Sollwerteingabe

Der Parameter löst flankengesteuert die Übernahme des am Sollwerteingang "LocalSetpointInput" anstehenden Sollwerts aus.

Ein Wechsel am Eingang "EnterInput" wird nur dann berücksichtigt, wenn am Eingangsparameter "Local" der Wert TRUE ansteht und wenn "ContinuousEnterFunct" = FALSE. Wenn diese Bedingungen erfüllt sind, wird beim Wechsel am Eingang "EnterInput" von 0 nach 1 der am "LocalSetpointInput" eingegebene Sollwert übernommen und am Ausgang "SetpointOutput" ausgegeben.

Diese Art der Sollwertübernahme eignet sich für Hardware-Eingaben, bspw. über eine Tafel, ein Pult oder ein Operator Panel (OP). Von einem OP muss die Übernahme durch eine Funktionstaste ausgelöst werden.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter:	**ContinuousEnterFunct**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE
Vorbelegung:	FALSE
Erläuterung:	Kontinuierliche lokale Sollwerterfassung Wenn "Local" = TRUE und "ContinuousEnterFunct" = TRUE wird der am Sollwerteingang "LocalSetpointInput" anstehende Sollwert ständig eingelesen und auf Änderung geprüft. Der an "LocalSetpointInput" eingelesene Sollwert wird am Ausgang "SetpointOutput" ausgegeben, falls sich der lokale Eingangswert zu diesem Zeitpunkt vom zuletzt zurückgespiegelten Sollwert unterscheidet. Wenn am Sollwerteingang "LocalSetpointInput" ein Array erkannt wird, wird dieses sofort (ohne Änderungsprüfung) am Ausgang "SetpointOutput" ausgegeben. Diese Art der Sollwertübernahme eignet sich für Software-Eingaben. Sie kann aber auch für Sollwerteingabe über ein Operator Panel (OP) genutzt werden, falls beim OP keine Funktionstaste vorgesehen ist. Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter:	**LocalSetpointInput**	
Deklaration:	INPUT	
Datentyp:	WORD	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0 (W#16#0)	
Adressbereich:	Eingangs-Worte	EW0 ... EWn PEW0 ... PEWn
	Merker-Worte	MW0 ... MWn LW0 ... LWn
	Daten-Worte	DBm.DBW0 ... n
Erläuterung:	Lokaler Sollwert-Eingang Ein am "LocalSetpointInput" anstehender Wert wird nur übernommen, wenn "Local" = TRUE. Wenn diese Bedingung erfüllt ist, dann ist die Art und Weise, wie der anstehende Sollwert bearbeitet wird, davon abhängig, ob es sich um eine Hardware- oder eine Software-Eingabe handelt. Die Eingabe-Art wird über den Parameter "ContinuousEnterFunct" bestimmt:	

- ContinuousEnterFunct = FALSE
  Hardware-Eingabe
  Ein an "LocalSetpointInput" anstehender Sollwert wird nur eingelesen, wenn an "EnterInput" ein Signalwechsel 0 → 1 festgestellt wird. Der lokal eingegebene Sollwert wird über den Ausgang "SetpointOutput" ausgegeben und an den Partner übertragen.
  Ein weiterer Sollwert wird vom Baustein erst dann wieder eingelesen, wenn an "EnterInput" mindestens ein OB1-Zyklus lang das Signal 0 erkannt wurde.
- ContinuousEnterFunct = TRUE
  Software-Eingabe
  Ein an "LocalSetpointInput" anstehender Sollwert wird kontinuierlich eingelesen und auf Änderung geprüft. Die Änderung wird durch Vergleich mit dem zuletzt gültigen Sollwert geprüft, das heißt mit dem Wert, der als Sollwert-Rückmeldung abgespeichert ist. Bei jeder Änderung wird der Sollwert sofort über den bei "SetpointOutput" parametrierten Ausgang ausgegeben und an den Partner übertragen.
  Bei Software-Eingaben kann in jedem OB1-Zyklus ein neuer Sollwert eingegeben werden.
  Bei Hardware-Eingaben muss ein Leerzyklus erkannt werden, bevor vom Baustein ein neuer Sollwert eingelesen werden kann.

Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.

Parameter:	**SetpointOutput**	
Deklaration:	OUTPUT	
Datentyp:	WORD	
Wertebereich:	Siehe Adressbereich	
Vorbelegung:	0 (W#16#0)	
Adressbereich:	Ausgangs-Worte	AW0 ... AWn PAW0 ... PAWn
	Merker-Worte	MW0 ... MWn LW0 ... LWn
	Daten-Worte	DBm.DBW0 ... n
Erläuterung:	Sollwert-Ausgangswort Der vom Partnerobjekt gesendete oder der an "LocalSetpointInput" lokal eingegebene Sollwert wird an den Ausgang von "SetpointOutput" ausgegeben.	

Parameter: **NewData**

Erläuterung: Zu Deklaration, Datentyp, Wertebereich, Vorbelegung und Funktion siehe Kapitel Wiederkehrende Parameter (Seite 1945).

Jedes Mal, wenn der Baustein einen neuen Sollwert vom Partnerobjekt empfangen und an den Ausgang "SetpointOutput" ausgegeben hat, wird der Ausgang "NewData" für einen OB1-Zyklus auf TRUE gesetzt. Dies gilt auch für den Fall einer neuen lokalen Sollwert-Eingabe (Local = 1).

Optional blocks

FC ListGenerator1500/300/400

Den FC ListGenerator gibt es in drei Varianten mit folgenden symbolischen Namen:

- Für S7-300
  ListGenerator300
- Für S7-400
  ListGenerator400
- Für S7-1500
  ListGenerator1500

Funktion

Der FC ListGenerator wird in einer CPU benötigt, in der Daten empfangen werden, die entweder keine oder keine vollständige Ziel-Adresse enthalten. Wesentlich ist das Fehlen der Ziel-Objekt-Nr., denn diese zeigt auf den Instanz-DB, in den die Empfangsinformation abzulegen ist.

Fehlende oder unvollständige Ziel-Adressen können vorkommen, wenn diese in der Station nicht oder nicht vollständig parametriert wurden. Dies ist bei Typicals zulässig, die binäre Information, Analogwerte oder Zählwerte senden. Wenn diese Typicals Daten an mehr als ein Ziel senden sollen, dann wird bei ihnen keine Zieladresse vergeben. Aufgrund der fehlenden Zielangabe wird das Sende-Telegramm dann automatisch an alle Ziele übertragen, zu denen bei der Projektierung eine Verbindung eingerichtet wurde. Diese Daten werden dann beim jeweiligen Ziel ohne Ziel-Adresse empfangen.

Hinweis

Ergänzung der Ziel-Teilnehmer-Nr.

Zu sendende Daten-Telegramme ohne Zieladresse werden von der sendenden TIM mit der Ziel-Teilnehmer-Nr. vervollständigt, eventuell auch mit mehreren, wenn mehrere Ziele versorgt werden müssen.

Die TIM belegt das Adressfeld für die Ziel-Objekt-Nr. mit dem Wert 0, denn dazu fehlt der TIM die entsprechende Information. Sie kennt nur die Ziel-Teilnehmer, zu denen von ihr aus eine Verbindung projektiert wurde.

Auf der Empfangsseite steht im Daten-Telegramm dann zwar die Ziel-Teilnehmer-Nr., aber die Ziel-Objekt-Nr. ist 0.

Falls die Ziel-Objekt-Nr. nicht im empfangenen Daten-Telegramm enthalten ist, greift der FC Distribute, der für die Verteilung der Empfangs-Telegramme zuständig ist, auf eine Objekt-Verweisliste zu.

Anhand der in jedem Daten-Telegramm enthaltenen Quell-Adresse (Quell-Teilnehmer-Nr. + Quell-Objekt-Nr.) sucht FC Distribute in der Liste nach einem Eintrag, der zu der betreffenden Quell-Adresse die fehlende Ziel-Objekt-Nr. angibt, d. h. die Nummer des lokalen Instanz-DB.

Diese Objekt-Verweisliste wird vom FC ListGenerator zusammengestellt. Der FC hat keine Parameter. Er wird im zyklischen Anwenderprogramm (OB1) im Anschluss an den FC BasicTask eingebunden.

Für die Zusammenstellung der Liste nutzt der FC ListGenerator die Adressen, die bei den empfangenden Typicals parametriert sind. Bei diesen Typicals sind die Angaben "PartnerNo" und "PartnerObjectNo" zwingend erforderlich. Diese Parameter sind identisch mit der Quell-Adresse in dem zugehörigen Empfangs-Telegramm. Da das Typical außerdem die Nummer seines Instanz-DB kennt, sind alle Adressen, die für einen Eintrag in die Verweisliste erforderlich sind, beim Typical bekannt.

Im Anlauf veranlasst der FC ListGenerator, dass alle empfangenden Typicals die bei ihnen parametrierte Adresse mit der Nummer des Instanz-DB in die Verweisliste eintragen. Die Objekt-Verweisliste braucht somit nicht extra parametriert zu werden, sie wird einfach aus der bereits gemachten Parametrierung der Empfangs-Typicals aufbereitet und ist somit immer konsistent.

Arbeitsweise

Der FC ListGenerator stellt die Listen nach dem Anlauf in drei aufeinanderfolgenden OB1-Zyklen zusammen:

- Im ersten Zyklus wird festgestellt, wieviele Einträge in der ersten und eventuell zweiten Objekt-Verweisliste benötigt werden. Die beteiligten Typicals zählen in diesem Durchlauf nur einen entsprechenden Zähler hoch.
- Im zweiten Zyklus generiert FC ListGenerator den Datenbaustein für die erste und eventuell zweite Objekt-Verweisliste in der erforderlichen Länge und besetzt alle Datenworte darin mit 0. Im gleichen Zyklus tragen alle beteiligten Typicals ihre Adressen und die Nummer des zugehörigen Instanz-DB in die Liste ein.
- Im dritten und letzten Zyklus sortiert FC ListGenerator alle Einträge in aufsteigender Reihenfolge. Die sortierte Ablage beschleunigt im laufenden Betrieb den Suchvorgang in der Liste.

Bei der Generierung des Datenbausteins geht der FC ListGenerator wie folgt vor:

Wenn noch keine Liste angelegt ist, wird eine freie DB-Nummer gesucht. Ausgehend von der Nummer des DB BasicData wird die Nummer des nächstniedrigen freien DB genommen.

Wenn bereits eine Liste vorhanden ist, prüft FC ListGenerator, ob der vorhandene DB lang genug ist für die aktuell erforderliche Anzahl von Verweisen. Wenn die Länge reicht, wird der Inhalt mit 0 vorbesetzt und anschließend erneut mit Adressen beschrieben und sortiert.

Wenn der vorhandene Datenbaustein zu kurz ist, dann wird bei den verschiedenen SIMATC-Familien unterschiedlich verfahren:

- Bei S7-300:
  Es wird ein neuer DB generiert. Der alte DB bleibt im Speicher vorhanden, denn bei S7-300 gibt es keine Löschfunktion für Datenbausteine.
  Bei der 300-CPU müssen Sie die alten DBs mit dem PG löschen.
  Anmerkung:
  Ohne das Generieren eines neuen DB müssten Sie diesen DB mit dem PG löschen. Wenn zu wenig Speicher in der CPU vorhanden wäre, um einen neuen DB zu generieren, müssten Sie den vorhandenen DB bereits vor dem neuen Anlauf löschen.
- Bei S7-400
  Der vorhandene DB wird gelöscht, der Speicher komprimiert und der DB anschließend mit der gleichen Nummer in der neuen Länge generiert.
  Bei der 400-CPU können Sie den Speicher evtl. manuell komprimieren oder die CPU neu laden.
- Bei S7-1500
  Der vorhandene DB wird gelöscht und mit der gleichen Nummer in der neuen Länge generiert.
  Bei der 1500-CPU wird der Speicher automatisch komprimiert, ebenso wie bei der 300-CPU.

Wenn FC Listgenerator keinen DB mehr generieren kann, wird eine Fehlermeldung in den Diagnosepuffer der CPU geschrieben:

- 0xB107 "Fehler bei der Generierung der Objektverweisliste"
  - DB[Info1]
    DB[Info1] kann nicht erzeugt werden.
  - Ursache: [Info2].
    In Info2 wird der Rückgabewert der SFC Create_DB-Funktion ausgegeben, lesen Sie bitte dort nach.

FC PartnerMonitor

Funktion

Der FC PartnerMonitor hat folgende Funktionen:

- Er zeigt für einen SINAUT-Teilnehmer einige wichtige Statusinformationen an (siehe Parameter "PartnerStatus").
- Über den FC kann eine Generalabfrage des Teilnehmers ausgelöst werden, außer an eine Leitstelle (bspw. ST7cc/ST7sc).
- Es lässt sich eine Dauerverbindung mit dem Teilnehmer auf- und abbauen.

Der FC kann im zyklischen Anwenderprogramm (im OB1) an beliebiger Stelle aufgerufen werden.

Wenn mehr als ein Teilnehmer überwacht und gesteuert werden soll, ist eine entsprechende Anzahl des FC PartnerMonitor im Anwenderprogramm einzubinden.

Ein SINAUT-Teilnehmer (Partner) kann nur eine ST7-CPU oder ein ST7cc sein, mit dem eine Verbindung projektiert wurde. TIMs können mit dem FC PartnerMonitor nicht überwacht oder gesteuert werden.

Hinweis

FC PartnerMonitor in einer Station

Der FC PartnerMonitor kann auch in einer Station eingesetzt werden. Die Steuereingänge zum Auf- und Abbau einer Dauerverbindung sind dann aber nicht nutzbar. Diese funktionieren nur von der Zentrale aus, wenn die lokale TIM eine Zentrale-TIM ist.

Parameter

Name: **PartnerNo**
Deklaration: INPUT
Datentyp: INT
Wertebereich: 1 ... 32000 [ms]
Erläuterung SINAUT-Teilnehmer-Nr. des zu überwachenden und steuernden Teilnehmers

Wenn die parametrierte "PartnerNo" nicht in der Buchführung (DB BasicData) gefunden wird, dann wird (nur im Anlauf) ein Eintrag in den Diagnosepuffer geschrieben (Ereignis-ID B101). Die CPU geht nicht in STOP.

Für einen korrekt parametrierten Teilnehmer wird dessen Zustand im Ausgangswort "PartnerStatus" angezeigt und die Steuereingänge werden bearbeitet.

Für einen unbekannten Teilnehmer wird die Bearbeitung solange ausgesetzt, bis der Fehler behoben ist. Das Ausgangswort "PartnerStatus" hat solange den Inhalt 0.

Name: **MaxConnectTime**
Deklaration: INPUT
Datentyp: INT
Wertebereich:
- 0 (keine Begrenzung)
- 1 ... 480 [Minuten]

Erläuterung Maximale Dauer einer Dauerverbindung

Die hier angegebene Zeit (> 0) wird beim Start einer Dauerverbindung aktiviert (siehe Parameter "PermanentCall_On").

Wenn die Zeit abläuft, bevor die Dauerverbindung zurückgenommen wurde, dann wird diese automatisch abgebrochen. Die Zeit wird neu getriggert, solange am Eingang "PermanentCall_On" 1-Signal ansteht.

Die hier angegeben Zeit gilt sowohl für eine Dauerverbindung im Wählnetz als auch für eine Dauerverbindung (Daueraufruf) auf einer Standleitung.

Name: **PartnerStatus**
Deklaration: OUTPUT

Datentyp:	WORD	
Wertebereich:	Ausgangs-Worte	AW0 ... AWn PAW0 ... PAWn
	Merker-Worte	MW0 ... MWn LW0 ... LWn
	Daten-Worte	DBm.DBW0 ... n
Erläuterung	Ausgangswort zur Anzeige des Zustandes des zu überwachenden Teilnehmers Falls Sie den Parameter nicht benötigen, lassen Sie ihn einfach offen.	

Bedeutung der Statusbits im Ausgangs-Wort "PartnerStatus"

Bit	.0	Zustand des Teilnehmers
	0	0 = Teilnehmer gestört
	1	1 = Teilnehmer o.k.

Bit	.1	Zustand der redundanten Verbindung
	0	0 = redundante Verbindung gestört
	1	1 = alle Verbindungen o.k.

Bit	.3	.2	Zustand der Generalabfrage (GA)
	0	0	0 = GA fertig ohne Fehler
	0	1	1 = GA gestartet
	1	0	2 = GA-Anfang empfangen
	1	1	3 = GA fertig mit Fehler (GA unvollständig oder nicht durchführbar, z. B. wegen Störung des Teilnehmers)

Bit	.6	.5	.4	Zustand der Wählverbindung
	0	0	0	0 = keine Verbindung
	0	0	1	1 = abgehender Ruf aktiviert
	0	1	0	2 = ankommender Ruf steht
	0	1	1	3 = abgehender Ruf steht
	1	0	0	4 = Dauerverbindung angemeldet
	1	0	1	5 = Dauerverbindung steht
	1	1	0	6 = Dauerverbindung abgemeldet

Bit	.7	Zustand der Wählverbindung
	0	0 = keine Wählverbindungsprüfung im Hintergrund
	1	1 = Wählverbindungsprüfung im Hintergrund aktiv

Bit	.8	Zustand des Daueraufrufs (bei Standleitung)
	0	0 = kein Daueraufruf
	1	1 = Daueraufruf aktiv

Bit	.9	Zustand der WAN-Verbindungsressourcen
*)	0	0 = Ausreichende Ressourcen beim Partner
	1	1 = Ressourcenmangel beim Partner

Bit	.10	Uhrzeitstatus
	0	0 = Datum/Uhrzeit beim Partner nicht vorhanden/nicht o.k.
	1	1 = Datum/Uhrzeit beim Partner OK

Bit	.11	Uhrzeitsynchronisation
	0	0 = Die Partner-CPU hat bei der letzten Synchronisierung eine plausible Zeit empfangen, oder es wurde nach Anlauf noch keine Uhrzeit empfangen.
	1	1 = Die Partner-CPU hat eine unplausible Uhrzeit empfangen; die zuletzt gültige Uhrzeit wird weiter geführt. (Anzeige nur mit TimeTask > V1.6)

Hinweis

Durchgangsparameter

Die nachfolgenden Parameter sind Durchgangsparameter (Deklaration IN_OUT):

- GeneralRequest
- PermanentCall_On
- PermanentCall_Off

Die Angabe von Lokalmerkern ist bei diesem Parametertyp problematisch und sollte deshalb nicht verwendet werden!

Name: **GeneralRequest**
Deklaration: IN_OUT
Datentyp: BOOL

Wertebereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7
Erläuterung	Eingang zum Auslösen einer Generalabfrage des bei PartnerNo angegebenen Teilnehmers Bei 1-Signal an diesem Eingang wird eine Generalabfrage des Teilnehmers ausgelöst, falls zu diesem Zeitpunkt noch keine Abfrage für diesen Teilnehmer läuft. Der Eingang wird danach automatisch vom FC zurückgesetzt. Falls ein Eingang einer Digitaleingabe angegeben wird (E 0.0 ... E n.7), dann müssen Sie selbst das am Eingang anstehende Signal zurücknehmen. Nehmen Sie das Signal vor Beendigung der aktuell laufenden Generalabfrage zurück, damit nicht sofort eine weitere Generalabfrage ausgelöst wird.	

Name:	**PermanentCall_On**
Deklaration:	IN_OUT
Datentyp:	BOOL

Wertebereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung Eingang zum Auslösen einer Dauerverbindung mit dem bei PartnerNo angegebenen Teilnehmer

Bei 1-Signal an diesem Eingang wird eine Dauerverbindung mit dem Teilnehmer aufgebaut, falls zu diesem Zeitpunkt noch keine Dauerverbindung mit diesem Teilnehmer steht. Der Eingang wird danach automatisch vom FC zurückgesetzt. Falls ein Eingang einer Digitaleingabe angegeben wird (E 0.0 ... E n. 7), dann müssen Sie das am Eingang anstehende Signal spätestens vor dem Abbau der bestehenden Dauerverbindung zurücknehmen.

Bei 1-Signal am Eingang "PermanentCall_On" wird außerdem die bei "MaxConnectTime" parametrierte Zeit gestartet, falls diese größer 0 ist.

Je nach dem, ob der Teilnehmer über eine Wählverbindung oder über eine Standleitung erreichbar ist, wird der Befehl zum Aufbau der Dauerverbindung wie folgt bearbeitet und am Ausgang "PartnerStatus" angezeigt:

- **Bei einer Wählverbindung:**
  Von der Zentrale-TIM wird eine Wählverbindung mit dem betreffenden Teilnehmer hergestellt und, unabhängig vom Datenverkehr, solange aufrecht gehalten, bis der Befehl zum Abbau gegeben wird.
  Der aktuelle Zustand der Dauerverbindung wird im Ausgangswort PartnerStatus mit den Bits 4 ... 6 angezeigt (siehe Parameter PartnerStatus).
- **Bei einer Standleitung:**
  In diesem Fall verkehrt die Zentrale-TIM im Aufrufbetrieb (Polling) mit den Stationen. Eine Dauerverbindung wird hier durch einen sogenannten Daueraufruf des beteffenden Teilnehmers verwirklicht. Dabei handelt es sich um einen intermittierenden Aufruf des Teilnehmers, d.h. die anderen Teilnehmer am Standleitungsnetz werden weiterhin aufgerufen, aber nach jedem Aufruf eines 'normalen' Teilnehmers wird zunächst wieder der bevorzugte Teilnehmer aufgerufen.
  Der aktuelle Zustand des Daueraufrufs zeigt das Bit 8 im Ausgangs-Parameter "PartnerStatus".

Besonderheit bei Stationen:

Von einer Station aus kann keine Dauerverbindung aufgebaut werden. Beim Einsatz des FC PartnerMonitor in einer Station ist die Nutzung dieses Steuereingangs also nicht möglich.

Name:	**PermanentCall_Off**
Deklaration:	IN_OUT
Datentyp:	BOOL

Wertebereich:	Eingang	E 0.0 ... E n.7
	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung Abbau einer bestehenden Dauerverbindung

Der Eingang dient zum Auslösen des Abbaus einer bestehenden Dauerverbindung mit dem bei "PartnerNo" angegebenen Teilnehmer.

Bei 1-Signal an diesem Eingang wird eine Dauerverbindung mit dem Teilnehmer abgebaut, falls zu diesem Zeitpunkt eine Dauerverbindung mit diesem Teilnehmer steht. Der Eingang wird danach automatisch vom FC zurückgesetzt. Falls ein Eingang einer Digitaleingabe angegeben wird (E 0.0 ... E n.7), müssen Sie das am Eingang anstehende Signal über das Anwenderprogramm zurückzunehmen. Dies sollten Sie spätestens vor dem Aufbau einer erneuten Dauerverbindung tun.

Je nach dem, ob der Teilnehmer über eine Wählverbindung oder über eine Standleitung erreichbar ist, wird der Befehl zum Abbau der Dauerverbindung wie folgt bearbeitet und am Ausgang "PartnerStatus" angezeigt:

- **Bei einer Wählverbindung:**
  Von der Zentrale-TIM wird die bestehende Wählverbindung abgebaut, jedoch erst dann, wenn keine Daten mehr zum Übertragen anstehen.
  Der aktuelle Zustand der Dauerverbindung wird im Ausgangswort PartnerStatus mit den Bits 4 ... 6 angezeigt (siehe Parameter PartnerStatus).
- **Bei einer Standleitung:**
  Die Zentrale-TIM löscht den Vermerk für den Daueraufruf des betreffenden Teilnehmers. Der Aufrufzyklus wird für alle angeschlossenen Teilnehmer im normalen Modus fortgesetzt.
  Der aktuelle Zustand des Daueraufrufs zeigt das Bit 8 im Ausgangs-Parameter "PartnerStatus".
  Bei einer Standleitung kann ein bestehender Daueraufruf auch gelöscht werden, indem ein Daueraufruf für einen anderen Teilnehmer an die Zentrale-TIM übergeben wird. Der vorhandene Auftrag wird durch den neuen ersetzt.

Besonderheit bei Stationen:

Von einer Station aus kann keine Dauerverbindung auf- oder abgebaut werden. Beim Einsatz des FC PartnerMonitor in einer Station ist die Nutzung dieses Steuereingangs also nicht möglich.

FC PartnerStatus

Funktion

Der FC PartnerStatus kann für maximal 8 SINAUT-Teilnehmer den aktuellen Status 'o.k.' oder 'gestört' anzeigen.

Der FC kann im zyklischen Anwenderprogramm (im OB1) an beliebiger Stelle aufgerufen werden.

Wenn Sie mehr als 8 Teilnehmer überwachen möchten, dann binden Sie entsprechend viele FCs PartnerStatus im Anwenderprogramm ein.

Partner kann eine ST7-CPU oder ein ST7cc sein, mit dem eine Verbindung projektiert wurde, oder eine lokale TIM.

Im Ausgangs-Byte "PartnerStatus" ist pro Teilnehmer ein Bit reserviert, das den Zustand des betreffenden Teilnehmers wie folgt anzeigt:

- FALSE (0):
  - Teilnehmer gestört
  - Zugehöriger Eingangsparameter nicht genutzt (= 0)
  - Teilnehmer unbekannt
- TRUE (1): Teilnehmer OK

Parameter

Name:	**Partner1 ... Partner8**
Deklaration:	INPUT
Datentyp:	INT
Wertebereich:	• 0 Wert für nicht benötigten Parameter • 1 ... 32000 Nummer des zu überwachenden Teilnehmers
Erläuterung	SINAUT-Teilnehmer-Nr. des zu überwachenden Teilnehmers. Wenn eine parametrierte Teilnehmernummer nicht in der Buchführung (DB BasicData) gefunden wird, dann wird (nur im Anlauf) ein Eintrag in den Diagnosepuffer geschrieben (Ereignis-ID B101). Die CPU geht nicht in STOP. Für korrekt parametrierte Teilnehmer wird deren Zustand im Ausgangs-Byte "PartnerStatus" angezeigt. Für unbekannte Teilnehmer wird keine Bearbeitung durchgeführt, bis der Fehler behoben ist. Ihre Zustands-Bits stehen auf 0.

Name:	**PartnerStatus**	
Deklaration:	OUTPUT	
Datentyp:	BYTE	
Wertebereich:	Ausgangs-Bytes	AB0 ... ABn PAB0 ... PABn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n
Erläuterung:	Anzeige des Zustandes der zu überwachenden Teilnehmer	

Zuordnung der Statusbits im Ausgangs-Byte "PartnerStatus" in Abhängigkeit von den Parametern "Partner1" ... "Partner8":

Bit	.7	.6	.5	.4	.3	.2	.1	.0
Partner	8	7	6	5	4	3	2	1

Wert:

- 0 = Partner gestört, nicht parametriert oder unbekannt.
- 1 = Partner OK

FC PathStatus

Funktion

Der Baustein (FC) zeigt aus Sicht der lokalen TIM den Wege-Status zu einem Partner an.

Maximal können 2 Wege (Haupt- und Ersatzweg) zu einem Partner projektiert werden. Dabei müssen beide Wege auf einer lokalen TIM beginnen oder enden.

Der Baustein zeigt folgendes an:

- Die Wege, über die der Partner erreichbar ist.
- Der aktuell genutzte Weg
- Die TIM-Schnittstelle, über die der Hauptweg projektiert wurde.
- Die TIM-Schnittstelle, über die der Ersatzweg projektiert wurde.

Der Weg einer Verbindung wird angegeben als Kombination aus den benutzten Schnittstellen der TIM und dem Status des Wegs.

Im Ausgangs-Byte PathStatus sind folgende Bits reserviert:

- Zwei Bits für die Schnittstelle des Hauptwegs
- Zwei Bits für die Schnittstelle des Ersatzwegs
- Zwei Bits für den Status des Hauptwegs
- Zwei Bits für den Status des Ersatzwegs

Der FC kann im zyklischen Anwenderprogramm (OB1) nach dem Aufruf des FC BasicTask an beliebiger Stelle aufgerufen werden.

Wenn der Wege-Status zu mehr als einem Teilnehmer angezeigt werden soll, dann sind entsprechend viele FC-Aufrufe im Anwenderprogramm zu programmieren.

Parameter

Name:	**Partner**
Deklaration:	INPUT
Datentyp:	INT
Wertebereich:	1 ... 32000 (Teilnehmernummer)
Erläuterung:	SINAUT-Teilnehmer-Nr. des Partners

Name:	**PathStatus**	
Deklaration:	OUTPUT	
Datentyp:	BYTE	
Wertebereich:	Ausgangs-Bytes	AB0 ... ABn PAB0 ... PABn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n
Erläuterung:	Anzeige des Wege-Status der Verbindung zum Partner	

PathStatus - Kodierung

Tabelle 1-146 Kodierung der Statusbits im Ausgangs-Byte "PathStatus"

Bit 6 + 7	Bit 4 + 5	Bit 2 + 3	Bit 0 + 1
Projektierte Schnittstelle		**Wege-Status**	
Nr. für Ersatzweg	Nr. für Hauptweg	Ersatzweg (2. Weg)	Hauptweg (1. Weg)

Projektierte Schnittstelle (Bit 4..7)

Die TIM-Schnittstellen sind im Baustein dezimal von 0 .. 3 durchnummeriert, siehe Spalte "Nr." in der Tabelle. Die Tabelle zeigt die Kodierung der Bits für die verschiedenen TIM-Typen.

Tabelle 1-147 Kodierung von Bit 4 + 5 (Schnittstelle des Hauptwegs) bzw. von Bit 6 + 7 (Schnittstelle des Ersatzwegs)

Zustand Bit 5 (7)	Zustand Bit 4 (6)	Schnittstellen			
		Nr.	TIM 3V-IE	TIM 4R-IE	TIM 1531 IRC
0	0	0	Ethernet "IE1" (X2)	Ethernet "IE1" (X3)	Ethernet "IE1" (X1)
0	1	1	-	Ethernet "IE2" (X4)	Ethernet "IE2" (X2)
1	0	2	Seriell "WAN1" (X1)	Seriell "WAN1" (X1)	Ethernet "IE3" (X3)
1	1	3	-	Seriell "WAN2" (X2)	Seriell "WAN" (X4)

Wege-Status (Bit 0..3)

- Hauptweg = 1. Weg (Bit 0 + 1)
- Ersatzweg = 2. Weg (Bit 2 + 3)

Tabelle 1-148 Zustandstabelle: Kodierung von Bit 0 + 1 bzw. von Bit 2 + 3

Zustand Bit 1 (3)	Zustand Bit 0 (2)	Bedeutung Bit 1	Bedeutung Bit 0
0	0	Weg nicht aktuell	Teilnehmer nicht erreichbar
0	1	Weg nicht aktuell	Teilnehmer erreichbar
1	0	Weg aktuell	Teilnehmer nicht erreichbar
1	1	Weg aktuell	Teilnehmer erreichbar

FC PulseCounter

Funktion

Der FC PulseCounter ist für die Zählimpulserfassung zuständig.

Maximal 8 Impulsfolgen werden über Digitaleingänge erfasst und mit Hilfe von SIMATIC-Zählern den Funktionsbausteinen zugeführt, welche die Zählwert-Telegramme zusammenstellen (Cnt01D_S, Cnt04D_S).

Die Erfassung der Zählimpulse erfolgt zeitgesteuert. Dazu ist der FC PulseCounter in einem Weckalarm-OB einzubinden, z. B. OB35. Das Aufrufintervall des Weckalarm-OB ist auf die Impulsdauer der Zählimpulse abzustimmen. Weitere Angaben zur Zählimpulserfassung über den Weckalarm-OB finden Sie im Kapitel Weckalarm-OB (Seite 1939).

Parameter

Name:	**InByte**	
Deklaration:	INPUT	
Datentyp:	BYTE	
Wertebereich:	Eingangs-Bytes	PEB0 ... PEBn
	Merker-Bytes	MB0 ... MBn LB0 ... LBn
	Daten-Bytes	DBm.DBB0 ... n
	Falls ein Eingangs-Byte einer Digitaleingabe angegeben wird, muss dies die Adresse des Peripherie-Bytes (PEB) direkt von den Digitaleingabebaugruppen sein. Nur durch den direkten Zugriff kann der aktuelle Zustand des Zähleingangs sicher erfasst werden. Beim Auslesen aus dem Prozessabbild der Eingänge (EB) könnten Zählimpulse unerkannt bleiben.	
Erläuterung:	Eingangs-Byte für Zählimpulse Die Eingänge für die Zählimpulserfassung können Byte-weise parametriert werden.	

Name:	**EnableMask**
Deklaration:	INPUT
Datentyp:	BYTE
Wertebereich:	B#16#00 ... B#16#FF

Erläuterung Freigabemaske für die Zähleingänge

Über diesen Parameter kann in Form einer Bitmaske angegeben werden, an welchen Eingängen im Eingangs-Byte tatsächlich Zählimpulse angeschlossen sind. Für jedes Bit in der Bitmaske gilt:

- 0 = Eingangsbit für die Erfassung gesperrt
- 1 = Eingangsbit für die Erfassung freigegeben

Die Eingabe der Maske ist nur in Hexadezimal-Format B#16#00 bis B#16#FF erlaubt.

Die Eingabe als 8-Bit-Binärzahl von 2#0 bis 2#1111 1111 ist beim Datentype BYTE nicht möglich.

Zuordnung der Bits in der Maske zu den Eingängen im Eingangs-Byte "InByte":

InByte	.7	.6	.5	.4	.3	.2	.1	.0
EnableMask B#16#	0 … F					0 … F		

Beispiel: EnableMask: B#16#83

Freigegeben sind: Eingänge .7, .1 und .0

Gesperrt sind: Eingänge .6 bis .2

Name: **CntIn_0 ... CntIn_7**

Deklaration: INPUT

Datentyp: COUNTER

Wertebereich: Z0 oder Z1 ... Zn
(n ist CPU-abhängig)

Erläuterung Impulszähler

Zu jedem der freigegebenen Zähleingängen ist beim entsprechenden Parameter "CntIn_0" ... "CntIn_7" ein SIMATIC-Zähler anzugeben. Mit jedem erfassten Impuls wird der SIMATIC-Zähler erhöht.

Die hier parametrierten Zähler sind bei den eigentlichen Zählwert-Funktionsbausteinen "Cnt01D_S" und "Cnt04D_S" als Eingangszähler anzugeben (Parameter "Counter_1 ... _4"). Diese Funktionsbausteinen lesen den jeweils zugeordneten Zähler aus und setzen ihn dann zurück.

Als Platzhalter für nicht benötigte Parameter wird empfohlen, den Zähler Z0 anzugeben.

Beispiel für die Parametrierung von "CntIn_0" ... "CntIn_7", ausgehend von "EnableMask" = B#16#83:

CntIn_0 : = Z10
CntIn_1 : = Z11
CntIn_2 : = Z0
CntIn_3 : = Z0
CntIn_4 : = Z0
CntIn_5 : = Z0
CntIn_6 : = Z0
CntIn_7 : = Z12

FC ST7ObjectTest

Gültigkeit

S7-300/400-CPU

Funktion

Der Aufruf des FC ST7ObjectTest im Programmierfehler-OB121 verhindert einen CPU-STOP, wenn die CPU Daten mit unbekannter Ziel-Objekt-Nr. empfängt.

FC ST7ObjectTest überprüft, weshalb OB121 aufgerufen wurde, d. h. welcher Bausteintyp fehlt.

- Wenn der fehlende Baustein ein Datenbaustein ist und wenn dieser Datenbaustein ein Instanz-DB eines SINAUT-Objekts ist, dann geht die CPU nicht in STOP.
- Wenn kein SINAUT Instanz-DB fehlt, sondern ein anderer Baustein (DB, FB, FC), dann können Sie beim Parameter "StopInOtherCases" die Reaktion festlegen:
  - CPU geht in STOP.
  - CPU läuft weiter.

Für weitere Einzelheiten zum Programmierfehler-OB121 sowie zu Hintergrundinformation bezüglich des Einsatzes von FC ST7ObjectTest siehe Kapitel Programmierfehler-OB (Seite 1939).

Parameter

Name:	**StopInOtherCases**
Deklaration:	INPUT
Datentyp:	BOOL
Wertebereich:	TRUE / FALSE
Erläuterung:	CPU soll bei anderen Fehlerfällen in STOP gehen. Zu Details des Parameters siehe oben, Abschnitt "Funktion". • TRUE: CPU geht in STOP. • FALSE: CPU läuft weiter.

FC TestCopy

Funktion

Mithilfe des FC TestCopy kann der Datenverkehr zwischen ST7-Teilnehmern auszugsweise oder auch vollständig mitgeschrieben werden. Über einzustellende Suchmasken im Steuerfeld des DB TestCopyData können gezielt einzelne Telegrammarten ausgefiltert werden, welche dann aus dem Sende- bzw. Empfangspuffer zur weiteren Auswertung in den DB TestCopyData kopiert werden. Zu Details siehe unten.

Sende- und Empfangs-Telegramme werden alle in dem gleichen DB TestCopyData abgespeichert. Dadurch ist die chronologische Reihenfolge der mitgeschriebenen Sende- und Empfangs-Telegramme einfach nachzuvollziehen.

Voraussetzungen

Zur Nutzung der TestCopy-Funktion muss das Anwenderprogramm folgende Bedingungen erfüllen:

- Die Funktion FC TestCopy muss in der CPU vorhanden sein.
- Der DB TestCopyData muss in ausreichender Länge in der CPU vorhanden sein.
  - Kopieren Sie dazu den DB TestCopyData (DB99) aus der TD7-Bibliothek in Ihre CPU.
  - Ändern Sie im DB bei Bedarf die Länge des Pufferbereichs, indem Sie in DW40 den Array "TestCopyBuffer", der vorbelegte Länge von [0..240] WORD hat, entsprechend vergrößern oder verkleinern.
- Machen Sie im jeweiligen Kommunikations-DB der CPU 300/400 (BComData / XComData / PComData), dessen Sende- und/oder Empfangs-Telegramme Sie mitschreiben möchten, folgende Einträge:
  - Tragen Sie in DW32 (TestCopyDBNo) des Kommunikations-DB die Nummer des DB TestCopyData ein.
  - Tragen Sie in DW34 (TestCopyFCNo) des Kommunikations-DB die Nummer des FC TestCopy ein.

  Gehen Sie beim Kommunikations-DB der CPU 1500 genauso vor. Die Variablen im DB BConnectData haben die gleichen Namen.

Einbinden von FC TestCopy in das Anwenderprogramm

Wenn die oben genannten Voraussetzungen erfüllt sind, wird die Testfunktion zyklisch durch den jeweiligen Kommunikations-FB der CPU bearbeitet.

Der FC TestCopy darf nicht im Anwenderprogramm aufgerufen werden.

Beobachten der mitgeschriebenen Daten

Nutzen Sie die vorgefertigte Beobachtungstabelle "TestCopyMonitor", die Sie aus der TD7-Bibliothek in das Verzeichnis "Beobachtungs- und Forcetabellen" der CPU kopieren können.

Wenn die Einstellungen auch nach CPU-Start erhalten bleiben sollen, dann lassen sich diese auch direkt in den Startwerten der einzelnen BConnection-Instanzen des DB BConnectData hinterlegen.

DB TestCopyData

Struktur des DB TestCopyData

Bereiche des DB TestCopyData

Der DB für die TestCopy-Funktion ist in folgende Bereiche eingeteilt (nach Offset im DB):

- **0 ... 27: Anwender-Schnittstelle**
  Schnittstelle zur Einstellung der TestCopy-Betriebsart und Funktionen. Der Bereich unterteilt sich in:
  - 1 ... 13
    Filtereinstellungen für RecvCopy-Funktion und Anzahl gezählter Empfangs-Telegramme
  - 15 ... 25
    Filtereinstellungen für SendCopy-Funktion und Anzahl gezählter Sende-Telegramme
- **28: Fehleranzeige**
- **31 ... 39: Interne Verwaltungszeiger**
- **40 ... (Vorbelegung: 523): Pufferbereich**
  Pufferbereich zur Ablage von Telegrammen, die den Filterkriterien entsprechen. Richten Sie den Pufferbereich als Array [0...xxxx] of WORD ein.

Die nachfolgende Tabelle zeigt die Struktur des DB TestCopyData.

Datentyp / (Offset)		Variablen-Name	Format	Erläuterung
Anwender-Schnittstelle				
DBB	0	OperationMode	BYTE	Betriebsart
DBW	12	Recv_TgramCounter	INT	Anzahl der kopierten Empfangs-Telegramme
DBW	26	Send_TgramCounter	INT	Anzahl der kopierten Sende-Telegramme
RecvCopy-Funktion				
DBB	1	Recv_TgrmType	BYTE	Empfangs-Filter: Telegrammart (TA)
DBW	2	Recv_DestSubscr	INT	Empfangs-Filter: Ziel-Teilnehmer-Nr.
DBW	4	Recv_DestObject	INT	Empfangs-Filter: Ziel-Objekt-Nr.
DBW	6	Recv_SourceSubscriber	INT	Empfangs-Filter: Quell-Teilnehmer-Nr.
DBW	8	Recv_SourceObject	INT	Empfangs-Filter: Quell-Objekt-Nr.
DBW	10	Recv_StartIndex	INT	Empfangs-Filter: Start-Index-Nr.
DBB	14	SpareDBB14	BYTE	*Reserve*
SendCopy-Funktion				
DBB	15	Send_TgrmType	BYTE	Sende-Filter: Telegrammart (TA)
DBW	16	Send_DestSubscr	INT	Sende-Filter: Ziel-Teilnehmer-Nr.
DBW	18	Send_DestObject	INT	Sende-Filter: Ziel-Objekt-Nr.

Datentyp / (Offset)		Variablen-Name	Format	Erläuterung
DBW	20	Send_SourceSubscriber	INT	Sende-Filter: Quell-Teilnehmer-Nr.
DBW	22	Send_SourceObject	INT	Sende-Filter: Quell-Objekt-Nr.
DBW	24	Send_StartIndex	INT	Sende-Filter: Start-Index-Nr.
Fehleranzeige				
DBB	28	FC_RetVal	BYTE	Fehlerinformation: 0 = Kein Fehler 1 = DB TestCopyData zu kurz 10 = Unbekannte Betriebsart
DBB	29	SpareDBB29	BYTE	*Reserve*
DBB	30	SpareDBB30	BYTE	*Reserve*
Interne Verwaltungszeiger				
DBB	31	TestCopyStatus	BYTE	Status-Byte für TestCopy-Betrieb
DBB	32	TestCopyCmdByte	BYTE	Befehlsbyte für TestCopy-Betrieb
DBB	33	TestCopyDelCount	BYTE	Schleifenzähler für Testcopy-Löschfunktion
DBW	34	NextFreeCopyByte	INT	Adresse des nächsten freien Test-CopyBuffer-Byte
DBD	36	StartTimeSFC64	DINT	SFC64-Zeit beim Start des Kopier-vorgangs
Pufferbereich				
DBB	40	TestCopyBuffer[0]	BYTE	Kopierbereich, Byte 0
DBB	41	TestCopyBuffer[1]	BYTE	Kopierbereich, Byte 1
DBB	42	TestCopyBuffer[2]	BYTE	Kopierbereich, Byte 2
DBB	43	TestCopyBuffer[3]	BYTE	Kopierbereich, Byte 3
DBB	n	TestCopyBuffer[n]	BYTE	Kopierbereich, Byte n

Aufbau eines kopierten Telegrammblocks

Ein Telegrammblock kann mehrere Telegramme enthalten. Die Telegramme werden im DB TestCopyData nach folgenden Regeln abgelegt:

1. Der erste Eintrag stellt die Zeitdifferenz in Millisekunden (7 Dekaden BCD plus Vorzeichen) seit der letzten Vorwahl einer Betriebsart > 0 dar.
2. Es folgt die Trennkennung AAAA bei gesendeten Telegrammen, EEEE bei empfangenen Telegrammen.
3. Ablage des ersten Telegramms aus dem Telegrammblock
4. Trennkennung AAAA bzw. EEEE
5. Ablage des letzten Telegramms aus dem Telegrammblock
6. Block-Endekennung FFFF

Beispiel

- Alle empfangenen Telegramme sollen in DB TestCopyData gespeichert werden.
- Kommunikation über X-Bausteine, d. h. max 76 Byte pro Empfangsblock.

- Quelle für FC TestCopy ist das Empfangsfach des DB XComData.
- Der aktuelle Empfangsblock enthält 3 Telegramme.

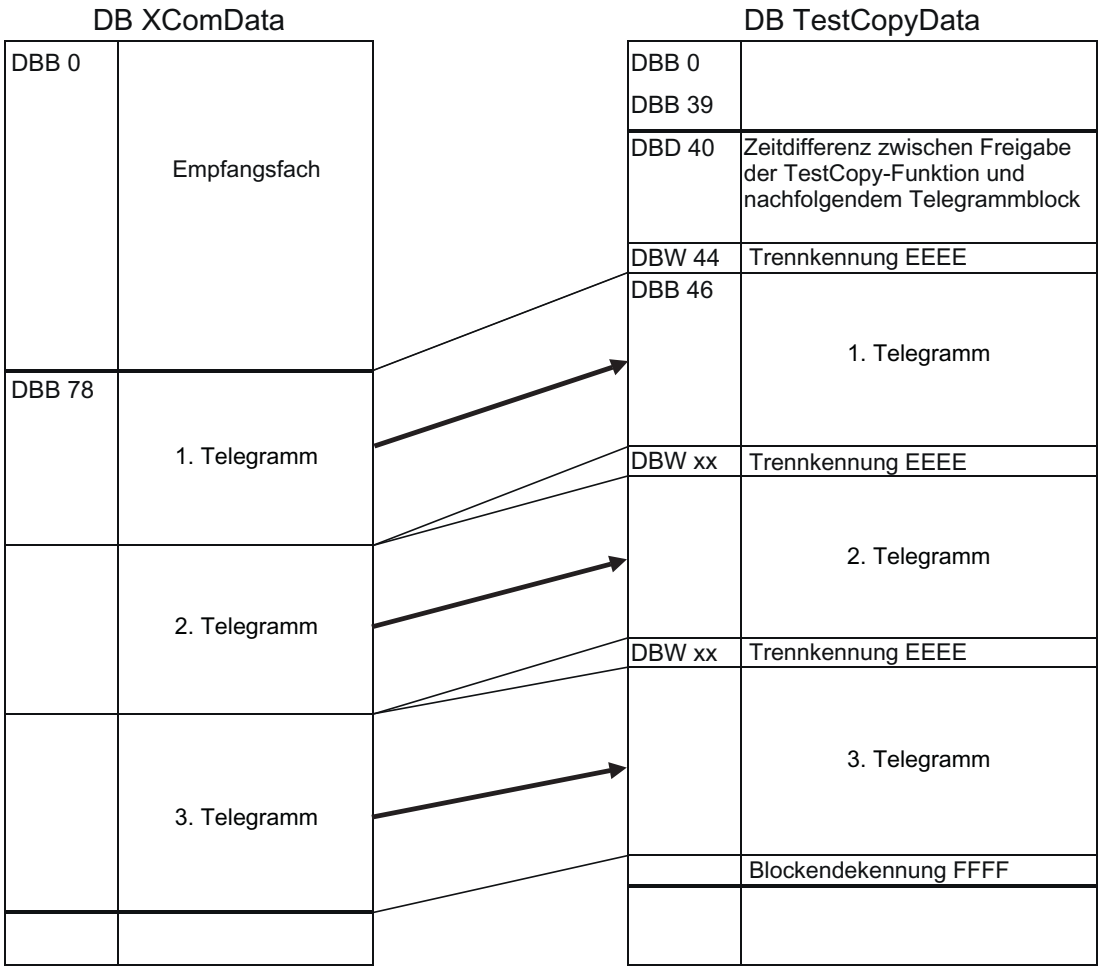

Bild 1-81 Beispiel für die Befüllung des DB TestCopyData

Längenberechnung

Für die Bestimmung der Mindestlänge des DB TestCopyData verwendet FC TestCopy folgende Parameter:

Parameter	Parametername	Länge
Länge Kommunikationsfach	$Len_{ComBuffer}$	76 oder 202 Byte *
Minimale Telegrammlänge	$Len_{MinTgrm}$	14 Byte
Offset Verwaltungsbereich	Offset	40 Byte
Länge Zeitdifferenz	Len_{dt}	4 Byte
Länge Blocktrenner	$Len_{Trenner}$	2 Byte

* 76 bei X-Kommunikation, 202 bei B-Kommunikation

Die Formel, mit der die eigentliche Berechnung durchgeführt wird, ist für X-Kommunikation und B-Kommunikation gleich. Die Ergebnisse unterscheiden sich lediglich aufgrund der unterschiedlichen Länge des Kommunikationsfachs bei der X- bzw. B-Kommunikation:

- **Länge bei X-Kommunikation**

Len_{Min_Xcom}	$=Len_{ComBuffer} + Offset + Len_{dt} + (Len_{ComBuffer} / Len_{MinTgrm} + 1) * Len_{Trenner}$
	= 76 + 40 + 4 + (76 / 14 + 1) * 2
	= 120 + 12 = 132 Byte minimal

- **Länge bei B-Kommunikation**

Len_{Min_Bcom}	$=Len_{ComBuffer} + Offset + Len_{dt} + (Len_{ComBuffer} / Len_{MinTgrm} + 1) * Len_{Trenner}$
	= 202 + 40 + 4 + (202 / 14 + 1) * 2
	= 236 + 40 = 276 Byte minimal

Wenn FC TestCopy feststellt, dass DB TestCopyData nicht die ermittelte Mindestlänge aufweist, dann wird eine entsprechende Fehlermeldung in das Datenbyte DBB28 eingetragen.

Bedienung von TestCopyMonitor

Betriebsarten und Übertragungsrichtungen von DB TestCopyData

In Byte 0 (OperationMode) des DB TestCopyData wird die Betriebsart des FC kodiert. Die Werte 0 ...3 kennzeichnen die Betriebsart:

- Betriebsart 0
  Funktion gesperrt
- Betriebsart 1
  Telegrammeintrag ab Anfang des DB TestCopyData
- Betriebsart 2
  DB TestCopyData endlos als Umlaufpuffer beschreiben
- Betriebsart 3
  Einmaliges Füllen des DB TestCopyData, dann Betriebsart 0 einstellen.

Die detaillierte Kodierung finden Sie in der Tabelle unten.

Zusammen mit der Betriebsart geht die Übertragungsrichtung der protokollierten Daten in die Kodierung von DBB0 ein. Dabei gilt folgende Zuordnung für die Kodierung:

- Bit 0...3
  Betriebsarten (0, 1 , 2, 3) für die Richtung "RecvCopy" (Empfangs-Telegramme mitschreiben)
- Bit 4...7
  Betriebsarten (0, 1 , 2, 3) für die Richtung "SendCopy" (Sende-Telegramme mitschreiben)

Bit	.7	.6	.5	.4	.3	.2	.1	.0
Richtung	SendCopy				RecvCopy			

Die Funktionen "SendCopy" und "RecvCopy" können einzeln oder gleichzeitig aktiviert werden, allerdings ist nur eine gemeinsame Betriebsart für beide Kommunikationsrichtungen möglich.

Wenn im niederwertigen Halb-Byte eine Betriebsart > 0 eingestellt ist, dann ist diese immer maßgebend für beide Kommunikationsrichtungen.

Nur falls die Betriebsart 0 in Bit 0...3 (RecvCopy) kodiert ist, gilt der Wert in Bit 4...7 (SendCopy).

Ausnahme:
Zum Löschen des DB TestCopyData muss DBB0 mit FF beschrieben werden, 0F reicht nicht aus.

Beispiele für die Kodierung von DBB0:

- 00_h
  Keine TestCopy-Funktion aktiviert
- 03_h
  Nur RecvCopy-Funktion (OperationMode = 3), keine SendCopy-Funktion.
- 30_h
  Nur SendCopy-Funktion (OperationMode = 3), keine RecvCopy-Funktion.
- 33_h
  RecvCopy-Funktion und SendCopy-Funktion (OperationMode = 3)
- FF_h
  Inhalt von DB TestCopyData löschen

Filtereinstellungen über TestCopyMonitor

Bedienen Sie FC TestCopy über eine Beobachtungstabelle, die in vorgefertigter Form als "TestCopyMonitor" in der TD7-Bibliothek vorhanden ist.

Folgende Einstellungen sind mit Hilfe der Beobachtungstabelle im DB TestCopy möglich:

Name (Symbol)	Zulässige Werte (hex.)	Bedeutung
OperationMode	00	Betriebsart 0: Funktion gesperrt
	11	Betriebsart 1: Telegrammeintrag immer ab Anfang des DB TestCopyData
	22	Betriebsart 2: DB TestCopyData endlos als Umlaufpuffer beschreiben
	02 20	• Beschreiben für SendCopy-Funktion • Beschreiben für RecvCopy-Funktion
	33	Betriebsart 3: Einmaliges Füllen des DB TestCopyData, dann Betriebsart 0 einstellen.
	03 30	• Füllen für SendCopy-Funktion • Füllen für RecvCopy-Funktion
	FF	Löschen des kompletten DB TestCopyData und auf Voreinstellung zurücksetzen

Als Ausgabewert liefert der FC TestCopy in DBW12 (Recv_TgramCounter) des DB TestCopyData einen Zählwert für die seit Einstellen der Betriebsart 1, 2, oder 3 eingetroffenen und zu den Filterkriterien passenden Empfangs-Telegramme.

In DBW26 (Sendcv_TgramCounter) wird die Anzahl der Sende-Telegramme eingetragen.

In DBB28 erhalten Sie einen Rückgabewert, der eine Aussage über aufgetretene Fehler bei der Bearbeitung des FC macht:

- FC_RetVal = 0
  Kein Fehler
- FC_RetVal = 1
  Der angegebene DB TestCopyData ist zu kurz.
- FC_RetVal = 10d
  Die eingetragene Betriebsart in DBB0 ist nicht definiert.

Zusätzlich stehen folgende spezielle Filter für die RecvCopy- bzw. SendCopy-Funktion zur Verfügung:

Name (Symbol) *	Zulässige Werte (hex.)	Bedeutung
Xxxx_TgrmType	FF	Alle Telegrammarten (TA) in DB TestCopyData kopieren
	00	Nur spontane Org.-Telegramme kopieren (TA = 0)
	11	Nur abgefragte Org.-Telegramme kopieren (TA = 1)
	22	Nur spontane Daten-Telegramme kopieren (TA = 2)
	33	Nur abgefragte Daten-Telegramme kopieren (TA = 3)
	01	Org.-Telegramme mit TA = 0 oder 1 kopieren
	23	Daten-Telegramme mit TA = 2 oder 3 kopieren
	Beliebige Kombination	Beliebige 0-, 1-, 2-, 3-Kombinationen kopieren
Xxxx_DestSubscr	Alle zulässigen Teilnehmer	Filter auf die Ziel-Teilnehmer-Nr. im Telegramm
	-1	Alle Telegramme unabhängig von der Ziel-Teilnehmer-Nr. kopieren
Xxxx_DestObject	Alle zulässigen Objekte	Filter auf die Ziel-Objekt-Nr. im Daten-Telegramm
	-1	Alle Daten-Telegramme unabhängig von der Ziel-Objekt-Nr. kopieren
Xxxx_SourceSubscr	Alle zulässigen Teilnehmer	Filter auf die Quell-Teilnehmer-Nr. im Telegramm
	-1	Alle Telegramme unabhängig von der Quell-Teilnehmer-Nr. kopieren
Xxxx_SourceObject	Alle zulässigen Objekte	Filter auf die Quell-Objekt-Nr. im Daten-Telegramm
	-1	Alle Daten-Telegramme unabhängig von der Quell-Objekt-Nr. kopieren

Name (Symbol) *	Zulässige Werte (hex.)	Bedeutung
Xxxx_StartIndex	Alle zulässigen Indices	Filter auf Start-Index-Nr. im Daten-Telegramm
	-1	Alle Daten-Telegramme unabhängig von Start-Index-Nr. kopieren

* "Xxxx": Platzhalter für "Recv" bzw. "Send"

Die gewünschten Filter-Variablen lassen sich über die Variablennamen des TestCopy-DB einfach ergänzen.

Hinweise zur Bedienung

Beim Wechsel von einer Betriebsart zur nächsten wird der Inhalt des DB nicht gelöscht. Es werden lediglich interne Zeiger und Telegrammzähler im Verwaltungsbereich des DB TestCopyData zurückgesetzt. Es wird daher empfohlen, bei Betriebsartenwechsel über die Löschfunktion "FF" zu gehen, um den Telegrammpufferbereich mit 0 vorzubesetzen. Dies erleichtert die Lesbarkeit der kopierten Telegrammblöcke.

Wenn Sende- und Empfangs-Telegramme kopiert werden sollen, dann muss im linken Halb-Byte des Parameters "OperationMode" die gleiche Betriebsart wie im rechten Halb-Byte eingetragen werden.

Es gilt das folgende Schema für Betriebsart 0, 1, 2 und 3:

- Wenn der Puffer gelöscht werden soll, dann muss immer FF_h eingetragen werden.
- Ein getrenntes Löschen der Empfangs- und Sende-Telegramme ist nicht möglich.

FC TimeTask

Funktion

Der FC TimeTask sorgt für eine kontinuierliche Datum- und Uhrzeitführung in einer CPU. Der FC hat keine Parameter.

Binden Sie den FC im Anwenderprogramm (im OB1) im Anschluss an den FC BasicTask ein.

Der FC TimeTask setzt voraus, dass die CPU von einer lokalen TIM synchronisiert wird. Aktivieren Sie die Uhrzeitsynchronisation bei der betreffenden TIM-Baugruppe.

Nach dem Anlauf der CPU liefert die TIM erstmalig das Datum und die aktuelle Uhrzeit über ein Org.-Telegramm. Danach wird die Uhrzeitsynchronisation in dem Zeitintervall durchgeführt, das bei der Projektierung der TIM festgelegt wurde. Für die Uhrzeitsynchronisation am MPI/Partyline wird ein Intervall von einer Minute empfohlen. Der FC TimeTask stellt die Uhr der CPU mit der von der TIM erhaltenen Uhrzeit.

Der FC liest die Uhrzeit in jedem OB1-Zyklus aus. Die gelesene Uhrzeit wird in den ersten beiden Doppelworten des DB BasicData abgelegt und mit Gültigkeit (bzw. Ungültigkeit) sowie mit Sommer- bzw. Winterzeit gekennzeichnet.

Vom DB BasicData entnehmen alle Bausteine die aktuelle Zeit, falls sie diese benötigen. Dies machen z. B. die Datenpunkt-Typicals zur Zeitstempelung ihrer Daten oder der FC Trigger

zum Prüfen, ob ein am FC parametrierter Zeitpunkt erreicht wurde oder ein vorgegebenes Zeitraster abgelaufen ist. Diese Uhrzeit steht auch dem Anwenderprogramm zur Verfügung.

Tabelle 1-149 Belegung der Datenworte mit Datum, Uhrzeit und Zeitstatus

CurrentDate	Datenbyte 0	Jahr * 10	Jahr * 1
	Datenbyte 1	Monat * 10	Monat * 1
	Datenbyte 2	Tag * 10	Tag * 1
	Datenbyte 3	Stunde * 10	Stunde * 1
CurrentTime	Datenbyte 4	Minute * 10	Minute * 1
	Datenbyte 5	Sekunde * 10	Sekunde * 1
	Datenbyte 6	Millisekunden * 100	Millisekunden * 10
	Datenbyte 7	Millisekunden * 1	Zeitstatus

Tabelle 1-150 Belegung des Halb-Bytes "Zeitstatus"

0	0 = Datum/Uhrzeit ungültig 1 = Datum/Uhrzeit gültig
1	0 = Winterzeit (Normalzeit) 1 = Sommerzeit
2	(nicht belegt)
3	(nicht belegt)

Ob Datum/Uhrzeit gültig ist, kann außer aus dem Zeitstatus auch anhand des Datenbits 16.1 "CpuClockOk" festgestellt werden. Sobald die Uhrzeit in der CPU gültig ist, wird dieses Bit vom FC TimeTask auf 1 gesetzt. Im Anwenderprogramm kann dieses Bit direkt unter dem symbolischen Namen "BasicData.CpuClockOk" abgefragt werden.

FC Trigger

Funktion

Der FC setzt zu einem parametrierbaren Zeitpunkt oder in einem vorgegebenen Zeitraster einen Ausgang (Merker, Datenbit oder Digitalausgang).

Der FC nimmt diesen Ausgang nach einem OB1-Zyklus wieder zurück.

Der FC kann im zyklischen Anwenderprogramm (OB1) an beliebiger Stelle aufgerufen werden, auch mehrfach.

Wenn das Durchlaufen eines Programmteils oder einer Software-Funktion mithilfe von FC Trigger angestoßen werden soll, wird empfohlen, FC Trigger direkt vor Ausführung dieser Funktion aufzurufen. Anwendungen für den Anstoß von Funktionen aufgrund des vom FC Trigger gesetzten Merkers können beispielsweise sein:

- Durchlaufen einer Funktion
- Aufruf eines Bausteins
- Anstoß einer Zählwertübertragung alle 2 Stunden

Falls mehrere Funktionen zum gleichen Zeitpunkt aktiviert werden sollen, kann dies mit einem Baustein FC Trigger realisiert werden, wenn alle Funktionen den gleichen vom FC gesetzten Merker abfragen. Dies funktioniert allerdings nur, wenn die angestoßenen Funktionen diesen Merker nicht selbst rücksetzen.

Abhilfe, wenn angesteuerte Bausteine den Merker zurücksetzen:

- Sie rufen den FC Trigger öfters auf, jedes Mal mit gleicher Zeit, aber anderem Ausgangsmerker.
- Sie vervielfältigen nach dem Aufruf von FC Trigger den gesetzten Ausgangsmerker in entsprechend viele weitere Merker.

Der FC greift auf die SINAUT-Uhrzeit in den ersten beiden Daten-Doppelworte des DB BasicData zu. Diese werden ständig versorgt, wenn ein FC TimeTask im Anwenderprogramm eingebaut ist und dieser von einer lokalen TIM regelmäßig synchronisiert wird. Der FC Trigger vergleicht die bei ihm parametrierte Zeit nur dann mit der aktuellen Uhrzeit, wenn die Uhrzeit im DB BasicData, Daten-Byte 7 (Zeitstatus-Byte, Bit 0 = 1) als gültig gekennzeichnet ist.

Die Genauigkeit, mit der FC Trigger arbeitet, hängt von der Genauigkeit der Uhrzeit und von der OB1-Zykluszeit ab.

Wenn die OB1-Zykluszeit kleiner 1 Sekunde ist (dies wird die Regel sein), dann wird der Ausgang exakt zum programmierten Sekundenwert mit der Unschärfe der OB1-Zykluszeit von kleiner 1 s gesetzt.

Wenn die OB1-Zykluszeit größer 1 Sekunde ist, dann arbeitet der FC mit einer Toleranz von 4 Sekunden. Wenn der FC zu spät, aber noch innerhalb von 4 Sekunden nach dem parametrierten Zeitpunkt bearbeitet wird, wird der Ausgang noch gesetzt.

Der am FC parametrierte Flankenmerker "Flag" wird gleichzeitig mit dem Ausgang gesetzt und 5 Sekunden nach dem parametrierten Zeitpunkt zurückgesetzt.

Für den Flankenmerker darf kein Platzhalter-Parameter verwendet werden und er darf nicht vom Anwenderprogramm zurückgesetzt werden.

Beispiele für die Parametrierung des FC Trigger finden Sie weiter unten.

Parameter

Name: **IntervalMode**
Deklaration: INPUT
Datentyp: BOOL
Wertebereich: TRUE / FALSE
Erläuterung Zeitpunkt / Zeitraster

- FALSE = Zeitpunkt
- TRUE = Zeitraster

Beispiele zur Parametrierung von Zeitpunkt oder Zeitraster finden Sie im Anschluss an dieser Erläuterung der Parameter.

Name: **Hour_Minute**
Deklaration: INPUT

Datentyp: WORD
Erläuterung Angabe der Werte für Stunden und Minuten.
Weitere Erläuterung: Siehe Parameter "Month_Year".

Name: **Second_Day**
Deklaration: INPUT
Datentyp: WORD
Erläuterung Angabe der Werte für Sekunden und Tag.
Weitere Erläuterung: Siehe Parameter "Month_Year".

Name: **Month_Year**
Deklaration: INPUT
Datentyp: WORD
Wertebereich:
- 00 ... 99
- FF

Erläuterung Angabe der Werte für Monat und Jahr
Jeder Parameter ist zweigeteilt. Jeder der beiden Werte pro Parameter wird mit zwei Ziffern als BCD-codierter Wert angegeben.
- Mit den beiden ersten Ziffern wird der Wert für Stunden, Sekunden bzw. Monat festgelegt.
- Mit den zwei anderen Ziffern wird der Wert für Minuten, Tag bzw. Jahr festgelegt.

Für nicht benötigte Parameterteile ist FF zu parametrieren.
Welche Parameter zulässig sind, hängt vom jeweiligen Parameter ab und vom Parameter "IntervalMode". Weitere Angaben hierzu finden Sie im Anschluss an die Erläuterung der Parameter.

Name: **TriggerOutput**
Deklaration: OUTPUT
Datentyp: BOOL

Wertebereich:	Ausgang	A 0.0 ... E n.7
	Merker	M 0.0 ... M n.7 L 0.0 ... L n.7
	Datenbit	DBm.DBX 0.0 ... n.7

Erläuterung Trigger-Ausgang
Der Ausgang wird beim Erreichen des bei "Hour" bis "Year" parametrierten Zeitpunkts bzw. Zeitrasters für die Dauer von einem OB1-Zyklus gesetzt.

Name: **Flag**
Deklaration: IN_OUT
Datentyp: BOOL

Wertebereich:	Merker	M 0.0 ... M n.7
	Datenbit	DBm.DBX 0.0 ... n.7
	Der Parameter ist ein Durchgangsparameter (Deklaration IN_OUT). Die Angabe von Lokalmerkern ist bei diesem Parameter problematisch und sollte deshalb nicht verwendet werden.	
Erläuterung	Flankenmerker zum Ausgang "TriggerOutput"	
	Es darf kein Platzhalter-Merker angegeben werden!	
	Der Flankenmerker darf nicht vom Anwenderprogramm zurückgesetzt werden.	

Beispiele für die Parametrierung eines Zeitpunkts oder Zeitrasters

IntervalMode = FALSE (oder 0)

Der FC arbeitet nach dem Zeitpunkt-Prinzip. Beim Erreichen des parametrierten Zeitpunkts wird der Ausgang "TriggerOutput" für einen OB1-Zyklus gesetzt.

Alle Zeit-Parameter können für die Angabe des Zeitpunktes verwendet werden. Nicht benötigte Parameter sind mit FF zu belegen. Sie werden bei der Prüfung auf Erreichen des Zeitpunkts nicht berücksichtigt.

Auch wenn "IntervalMode" = 0 ist, lassen sich in gewissen Bereichen Zeitraster einstellen, siehe nachfolgende Beispiele.

Zulässige Wertebereiche für die Zeit-Parameter:

Stunden	00-23	Tag	01-31
Minuten	00-59	Monat	01-12
Sekunden	00-59	Jahr	00-99

Beispiele:

- IntervalMode = FALSE
  Der Ausgang "TriggerOutput" wird einmalig am 04.02.91 um 06:45:12 gesetzt:
  - Hour_Minute : W#16#0645
  - Second_Day : W#16#1204
  - Month_Year : W#16#0291
- IntervalMode = FALSE
  Der Ausgang "TriggerOutput" wird jeden Tag um 06:00:00 gesetzt:
  - Hour_Minute : W#16#0600
  - Second_Day : W#16#00FF
  - Month_Year : W#16#FFFF

- IntervalMode = FALSE
  Der Ausgang "TriggerOutput" wird an jedem 1. eines Monats um 06:00:00 gesetzt:
  - Hour_Minute : W#16#0600
  - Second_Day : W#16#0001
  - Month_Year : W#16#FFFF
- IntervalMode = FALSE
  Der Ausgang "TriggerOutput" wird jedes Jahr am 1. Oktober um 06:00:00 gesetzt:
  - Hour_Minute : W#16#0600
  - Second_Day : W#16#0001
  - Month_Year : W#16#10FF

IntervalMode = TRUE (oder 1)

Der FC arbeitet nach dem Zeitraster-Prinzip. Beim Erreichen des parametrierten Zeitwerts oder eines Vielfachen davon wird der Ausgang "TriggerOutput" für einen OB1-Zyklus gesetzt.

Nur die Angaben für Stunden, Minuten und Sekunden sind relevant. Die Datum-Parameter werden nicht berücksichtigt. Ein Zeitraster kann außerdem nur in Stunden oder Minuten oder in Sekunden parametriert werden. Alle nicht benötigten Zeit-Parameter sind mit FF zu belegen.

Folgende Zeitraster sind zulässig:

- Stunden: 01, 02, 03, 04, 06, 08, 12, 24
- Minuten: 01, 02, 03, 04, 05, 06, 10, 12, 15, 20, 30, 60
- Sekunden: 10, 12, 15, 20, 30, 60

Beispiele:

- IntervalMode : TRUE
  Der Ausgang "TriggerOutput" wird gesetzt, und zwar:
  - Hour_Minute : W#16#06FF (alle 6 Stunden)
  - Second_Day : W#16#FFFF (um 00:00:00, 06:00:00, 12:00:00 Uhr und ...)
  - Month_Year : W#16#FFFF (... um 18:00:00 Uhr)
- IntervalMode : TRUE
  Der Ausgang "TriggerOutput" wird gesetzt, und zwar:
  - Hour_Minute : W#16#FF30
  - Second_Day : W#16#FFFF (um 00:00:00, 00:30:00, 01:00:00 Uhr und ...)
  - Month_Year : W#16#FFFF (... um 01:30:00, 02:00:00, 02:30:00 Uhr usw.)

Fehlermeldung im Anlauf

Der FC überprüft die Parameter Hour_Minute, Second_Day und Month_Year in jedem Zyklus auf Einhaltung der zulässigen Wertebereiche. Was jeweils zulässig ist, ist auch vom Parameter "IntervalMode" abhängig.

Bei falscher Parametrierung wird nur im Anlauf eine Fehlermeldung in den Diagnosepuffer eingetragen (Ereignis-ID B113). Die CPU geht nicht in STOP. Der FC prüft danach solange die Parameter, ohne Fehlermeldungen auszugeben, bis der Fehler behoben ist.

Die Diagnosemeldung liefert eine genaue Bezeichnung des fehlerhaften Parameters (laufende Nummer des Parameters, d. h. 2, 3 oder 4). Die Ursachen für die Diagnosemeldung sind abhängig vom Parameter "IntervalMode".

IntervalMode = FALSE (oder 0)

Die zulässigen Wertebereiche für die Parameter Stunden, Minuten, Sekunden, Tag, Monat und Jahr wurden nicht eingehalten. Außer FF dürfen parametriert werden:

Stunden	00-23	Tag	01-31
Minuten	00-59	Monat	01-12
Sekunden	00-59	Jahr	00-99

"IntervalMode" = TRUE (oder 1)

In diesem Fall kann der Fehler zwei verschiedene Ursachen haben:

- Die zulässigen Wertebereiche für die Parameter Stunden, Minuten und Sekunden wurden nicht eingehalten. Außer FF dürfen parametriert werden:
  - Stunden: 01, 02, 03, 04, 06, 08, 12, 24
  - Minuten: 01, 02, 03, 04, 05, 06, 10, 12, 15, 20, 30, 60
  - Sekunden: 10, 12, 15, 20, 30, 60
- Ein Zeitraster kann nur in Stunden oder Minuten oder Sekunden parametriert werden. Die beiden nicht genutzten Parameter müssen mit FF beschrieben sein. Auch wenn für alle drei genannten Parameter FF ausgefüllt wurde, liegt ein Fehler vor.

System blocks

FC BasicTask_*

Bausteinvarianten

Den Baustein gibt es in folgenden Varianten für die verschiedenen CPU-Typen:

- **BasicTask_B**
  Für S7-1500-CPU
  Der Baustein für die S7-1500-CPU ist nicht für die S7-400-CPU nutzbar.
- **BasicTask_B**
  Für S7-400-CPU
  Der Baustein für die S7-400-CPU ist nicht für die S7-1500-CPU nutzbar.
- **BasicTask_X**
  Für S7-300-CPU mit Partyline
- **BasicTask_P**
  Für S7-300-CPU ohne Partyline

Zu "Partyline" siehe Glossar.

Die Aufgaben der verschiedenen FCs "BasicTask_*" für die Abwicklung der grundsätzlichen Kommunikationsaufgaben sind im jeweiligen CPU-Typ gleich. Der einzige Parameter des FC ist jeweils identisch.

Beim Generieren der TD7onCPU wird in der jeweiligen CPU automatisch der richtige FC angelegt.

Funktion

Der Baustein wird in jeder CPU benötigt. Er übernimmt folgende Aufgaben:

- Zentrale Aufgaben im Anlauf
- Die Abarbeitung aller Kommunikationsfächer
- Zentrale organisatorische Aufgaben wie das Starten, Überwachen und Beantworten von Generalabfragen.

Rufen Sie den FC BasicTask als ersten Baustein im OB1 auf.

Parameter

Name:	**UserFC**
Deklaration:	INPUT
Datentyp:	INT

Wertebereich:
- 1 ... 32000
  Nummer des FC
  Die maximal mögliche Nummer ist abhängig von der CPU.
- 0
  Ersatzwert, falls kein FC für den angegebenen Zweck vorhanden ist.

Erläuterung: Nummer eines Anwender-FC zur spezifischen Weiterverarbeitung der empfangenen Daten

Falls ein FC angegeben ist, wird dieser FC bei allen empfangenen Daten automatisch vom Anwenderprogramm aufgerufen. Zum Zeitpunkt des Aufrufs steht das Empfangs-Telegramm noch im Empfangsfach des Kommunikations-DB.

Das Programm im Anwender-FC kann das Empfangs-Telegramm aus dem Empfangsfach auslesen und beliebig weiterverarbeiten, z. B. die empfangenen Daten in ein Zwischenpuffer schreiben.

Über das Anwenderprogramm können aus dem DB BasicData über die Variable "CurrentComDB" in DW60 die erforderlichen Informationen zum Kommunikations-DB ausgelesen werden. CurrentComDB enthält folgende Information:

- S7-1500
  Index der aktuellen BConnection-Instanz im DB BConnectData
- S7-300/400
  Nummer des aktuellen Kommunikations-DB

Im geöffneten Kommunikations-DB kann über die Variable "CurrentReceivedMessage" in DW10 der Anfang des aktuellen Empfangs-Telegramms im Empfangsfach gefunden werden.

DB BasicData

Dieser Datenbaustein übernimmt die zentrale Datenhaltung. Er enthält Informationen, die an zentraler Stelle für alle Bausteine bereitgehalten werden müssen. Der Datenbaustein enthält u. a. die Teilnehmerbuchführung und die Verbindungsbeschreibungen.

Der DB BasicData wird automatisch in der erforderlichen Länge generiert, mit den Teilnehmer- und Verbindungs-spezifischen Daten vorbesetzt und anschließend in das Bausteinverzeichnis der CPU abgespeichert.

DB BasicData ist in jeder CPU einmal vorhanden.

Hinweis

Nummer des DB BasicData

Der DB BasicData hat in der TD7-Bibliothek die Nummer DB127 und wird bei der Generierung dieses DB für die verschiedenen CPUs auch unter dieser Nummer dort abgelegt. Prinzipiell wäre es zwar möglich, die Nummer zu ändern, doch ist dies mit relativ viel Aufwand verbunden und kann zu Fehlern bei der weiteren Erstellung des Anwenderprogramms führen.

Es wird daher empfohlen, die DB-Nummer 127 möglichst für den DB BasicData freizuhalten.

FC Create

Hilfsbaustein für die Zusammenstellung von zu sendenden Daten und deren Eintrag in die relevanten Instanzen der Sendefächer (Array SendBuffer) im jeweiligen Kommunikations-DB. Dies sind folgende DBs:

- S7-1500
  DB BConnectData > BConnection[n]
- S7-400
  DB BComDataXX
- S7-300
  - DB XComDataXX
    oder
  - DB PComDataXX

XX ist Platzhalter für die laufende Nummer (01, 02, ...).

Der FC Create wird von den Datenpunkt-Typicals für Daten- und organisatorische Telegramme benötigt und vom FC BasicTask nur für organisatorische Telegramme.

FC Diagnose / Diagnostics

Name des Bausteins:

- Für S7-300: FC Diagnose
- Für S7-1500: FC Diagnosis

Hilfsbaustein zum Eintrag von SINAUT-Systemmeldungen in den Diagnosepuffer der CPU.

FC Distribute

Hilfsbaustein zur Verteilung der im Empfangsfach stehenden Daten an die zuständigen Datenpunkt-Typicals oder an die Teilnehmer-Objekte in der Teilnehmerbuchführung.

FC Search

Hilfsbaustein für folgende Suchaufgaben:

- Suchen der Anfangsadresse eines Teilnehmer-Objekts innerhalb der Teilnehmerbuchführung
- Suchen der lokalen Objekt-Nr. (Instanz-DB) aus einer der beiden Objekt-Verweislisten für ein Empfangs-Telegramm mit unvollständiger Ziel-Adresse

Der Hilfsbaustein wird von fast allen Bausteinen benötigt.

FC Startup

Der Baustein wird in jeder CPU benötigt. Er muss im Anlaufprogramm OB100 eingebunden werden.

Der Baustein hat die Aufgabe, im DB BasicData den Anlaufmerker zu setzen und den zugehörigen Flankenmerker zurückzunehmen, falls dieser noch ansteht.

Der Baustein hat keine Parameter.

System blocks - BCom

FB BCom

Gültigkeit

- S7-1500
- S7-400

Der Baustein für die S7-400-CPU ist nicht für die S7-1500-CPU nutzbar und umgekehrt. Die Aufgaben der beiden FCs sind gleich.

Funktion

Hilfsbaustein für FC BasicTask_B zur Bearbeitung eines Kommunikationsfachs vom Typ DB BConnectData (S7-1500) bzw. DB BComData (S7-400).

Über den DB wird eine projektierte PBK-Verbindung (Programmierte Baustein-Kommunikation) mithilfe der SFBs "BSEND" und "BRCV" abgewickelt.

Der FB BCom sorgt auch dafür, dass Empfangs-Telegramme unmittelbar an die zuständigen Empfangsobjekte in der CPU verteilt werden. Dazu ruft der FB BCom den FC Distribute als Hilfsbaustein auf.

DB BComData

Gültigkeit

S7-400

Funktion

Instanz-Datenbaustein für den Kommunikationsbaustein FB BCom. Der Instanz-DB stellt das Kommunikationsfach dar und enthält unter anderem:

- Ein Empfangsfach (ReceiveBuffer)
- Ein Sendefach (SendBuffer)

Weiterhin enthält er zentrale Daten, die zur Steuerung und Verwaltung der über dieses Kommunikationsfach geführten PBK-Verbindung benötigt werden.

Der Datenbaustein wird in jeder CPU benötigt, in dem der FB BCom eingesetzt wird. Falls von der CPU aus mehrere PBK-Verbindungen unterhalten werden, wird der DB mehrfach benötigt.

Der DB BComData wird automatisch in der erforderlichen Länge generiert, mit den verbindungsspezifischen Daten vorbesetzt und anschließend in das Bausteinverzeichnis der CPU abgespeichert.

FB BConnect

Gültigkeit

S7-1500

Funktion

Hilfsbaustein für FC BasicTask zur Bearbeitung der Verbindungen zu den Kommunikationspartnern.

DB BConnectData

Gültigkeit

S7-1500

Funktion

Instanz-Datenbaustein für den Kommunikationsbaustein FB BCom. Der Instanz-DB stellt das Kommunikationsfach dar und enthält unter anderem:

- Ein Empfangsfach (ReceiveBuffer)
- Ein Sendefach (SendBuffer)

Weiterhin enthält er zentrale Daten, die zur Steuerung und Verwaltung der über dieses Kommunikationsfach geführten PBK-Verbindung benötigt werden.

Der Datenbaustein wird in jeder CPU benötigt, in dem der FB BCom eingesetzt wird. Falls von der CPU aus mehrere PBK-Verbindungen unterhalten werden, wird der DB mehrfach benötigt.

Der DB BComData wird automatisch in der erforderlichen Länge generiert, mit den verbindungsspezifischen Daten vorbesetzt und anschließend in das Bausteinverzeichnis der CPU abgespeichert.

System blocks - PCom

FB PCom

Gültigkeit

S7-300-CPU ohne Partyline

Der FB PCom wird nur bei der Kommunikation über den P-Bus verwendet. Dies betrifft die Kommunikation zwischen einer TIM und einer CPU mit P-Bus.

Funktion

Hilfsbaustein für FC BasicTask_P zur Bearbeitung eines Kommunikationsfachs vom Typ DB PComData mit Hilfe der SFCs WR_REC und RD_REC.

Empfangs-Telegramme werden auch unmittelbar an die zuständigen Empfangsobjekte in der CPU verteilt. Dazu ruft der FB PCom den FC Distribute als Hilfsbaustein auf.

DB PComData

Instanz-Datenbaustein für den Kommunikationsbaustein FB PCom. Der Instanz-DB stellt das Kommunikationsfach dar und enthält:

- Ein Empfangsfach (ReceiveBuffer)
- Ein Sendefach (SendBuffer)

Weiterhin enthält er zentrale Daten, die zur Steuerung und Verwaltung der über dieses Kommunikationsfach geführten Verbindung benötigt werden.

Der Datenbaustein wird in jeder CPU benötigt, in dem der FB PCom eingesetzt wird. Falls von der CPU aus mehrere entsprechende Verbindungen unterhalten werden, wird der DB mehrfach benötigt.

System blocks - XCom

FB XCom

Gültigkeit

S7-300-CPU mit Partyline

Funktion

Hilfsbaustein für FC BasicTask_X zur Bearbeitung eines Kommunikationsfachs vom Typ DB XComData.

Über DB BComData wird eine projektierte Verbindung (PBK-Verbindung) mithilfe der SFCs "X_SEND" und "X_RCV" abgewickelt.

Der FB XCom sorgt auch dafür, dass Empfangs-Telegramme unmittelbar an die zuständigen Empfangsobjekte in der CPU verteilt werden. Dazu ruft der FB XCom den FC Distribute als Hilfsbaustein auf.

DB XComData

Instanz-Datenbaustein für den Kommunikationsbaustein FB XCom. Der Instanz-DB stellt das Kommunikationsfach dar und enthält:

- Ein Empfangsfach (ReceiveBuffer)
- Ein Sendefach (SendBuffer)

Weiterhin enthält er zentrale Daten, die zur Steuerung und Verwaltung der über dieses Kommunikationsfach geführten X-Verbindung benötigt werden.

Der Datenbaustein wird in jeder CPU benötigt, in dem der FB XCom eingesetzt wird. Falls von der CPU aus mehrere X-Verbindungen unterhalten werden, wird der DB mehrfach benötigt.

Der DB XComData wird automatisch in der erforderlichen Länge generiert, mit den verbindungsspezifischen Daten vorbesetzt und anschließend in das Bausteinverzeichnis der CPU abgespeichert.

1.4.8 SCALANCE X, W und M

1.4.8.1 SCALANCE X / W / M projektieren

Rechtlicher Hinweis

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung qualifiziertem Personal gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. **Qualifiziertes Personal** ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Hinweis zum Download der Konfigurationsdaten

Beim lokalen Download auf das SCALANCE-Gerät wird das werkseitig voreingestellte Passwort nicht akzeptiert.

- Ändern Sie das werkseitig voreingestellte Passwort über WBM oder CLI.

SCALANCE X projektieren

Systemfunktionen

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den IE-Switches. Beachten Sie, dass in der Online-Hilfe alle Funktionen beschrieben sind. Abhängig von Ihrem IE-Switch stehen Ihnen manche Funktionen nicht zur Verfügung.

Technische Änderungen sind vorbehalten

		SCA-LANCE XB-200	SCA-LANCE XC-200	SCA-LANCE XP-200	SCA-LANCE XF-200BA	SCA-LANCE XR-300 WG	SCA-LANCE XM-400/ SCA-LANCE XR-500
Informationen	ARP-Tabelle	✓	✓	✓	✓	✓	✓
	Log-Tabelle	✓	✓	✓	✓	✓	✓
	Ethernet-Statistiken	✓	✓	✓	✓	✓	✓
	Diagnose (Temperatur)	-	✓	✓	✓	✓	✓
System-Funktionen	DNS	-	-	-	-	-	✓
	SMTP-Client	✓	✓	✓	✓	✓	✓
	DHCP-Client	✓	✓	✓	✓	✓	✓
	DHCP-Server	✓ 2)	✓	✓	✓	✓ 2)	✓
	SNMP	✓	✓	✓	✓	✓	✓
	Manuelle Zeiteinstellung	✓	✓	✓	✓	✓	✓
	DST	-	✓	✓	✓	-	✓
	SNTP	✓	✓	✓	✓	✓	✓
	NTP	✓	✓	✓	✓	✓	✓
	SIMATIC Time Client	✓	✓	✓	✓	✓	✓
	PTP	-	-	-	-	-	✓ 1)
	Auto-Logout	✓	✓	✓	✓	✓	✓
	Syslog-Client	✓	✓	✓	✓	✓	✓
	Fehlerkontrolle	✓	✓	✓	✓	✓	✓
	PROFINET	✓	✓	✓	✓	✓	✓
	EtherNet/IP	✓	✓	✓	✓ 4)	✓	-
	PLUG	-	✓	✓	✓	-	✓
	Port-Diagnose	✓	✓	✓	✓	✓	✓

		SCA-LANCE XB-200	SCA-LANCE XC-200	SCA-LANCE XP-200	SCA-LANCE XF-200BA	SCA-LANCE XR-300 WG	SCA-LANCE XM-400/ SCA-LANCE XR-500
Layer 2-Funktionen	Sendeprioritäten	-	✓	✓	✓	-	-
	CoS-Zuordnung	✓	✓	✓	✓	✓	✓
	DSCP-Zuordnung	✓	✓	✓	✓	✓	✓
	QoS-Priorisierung	✓	✓	✓	✓	✓	✓
	CoS Port-Neuzuordnung	-	✓	✓	✓	-	-
	Lastkontrolle	✓	✓	✓	✓	✓	✓
	GVRP	-	✓	✓	✓	-	✓
	Base Bridge-Modus (VLAN)	✓	✓	✓	✓	✓	-
	Port-basiertes VLAN	✓	✓	✓	✓ 4)	✓	✓
	Private VLAN	-	✓	✓	-	-	✓
	Provider Bridge (Bridge-Modus)	-	✓	✓	✓	-	✓
	Switch-Port VLAN Trunk	-	✓	✓	✓ 4)	-	✓
	Protokoll-basierte VLAN-Gruppe	-	-	-	-	-	✓
	Protokoll-basierter VLAN-Port	-	-	-	-	-	✓
	IPv4 Subnetz-basiertes VLAN	-	-	-	-	-	✓
	Port-basiertes Mirroring	✓	✓	✓	✓	✓	✓
	RSPAN	-	-	-	-	-	✓
	VLAN-basiertes Mirroring	-	-	-	-	-	✓
	MAC Flow-Mirroring	-	-	-	-	-	✓
	IP Flow-Mirroring	-	-	-	-	-	✓
	Dynamic MAC Aging	✓	✓	✓	✓	✓	✓
	Ringredundanz	✓	✓	✓	✓	✓	✓
	MRP-Interconnection	-	-	-	-	-	✓
	H-Sync-Unterstützung	-	✓	✓	✓ 4)	-	-
	S2-Geräte	-	✓	✓	✓ 4)	-	-
	CiR/H-CiR-Unterstützung	-	✓	✓	✓ 4)	-	-
	Standby (HRP)	✓	✓	✓	✓ 4)	✓	✓
	Observer (HRP)	-	✓	✓	✓	-	✓
	Link Check	✓	✓	-	✓ 4)	✓	✓
	Spanning Tree	✓	✓	✓	✓	✓	✓

		SCA-LANCE XB-200	SCA-LANCE XC-200	SCA-LANCE XP-200	SCA-LANCE XF-200BA	SCA-LANCE XR-300 WG	SCA-LANCE XM-400/ SCA-LANCE XR-500
	RSTP	✓	✓	✓	✓	✓	✓
	MSTP	-	✓	✓	-	-	✓
	Enhanced Passive Listening Compatibility	✓	✓	✓	✓	✓	✓
	Ring mit RSTP	-	✓	✓	✓	-	✓
	RSTP+	-	-	-	-	-	✓
	Loop Detection	✓	✓	✓	✓	✓	✓
	Link Aggregation	-	✓	✓	✓ 4)	-	✓
	DCP-Weiterleitung	✓	✓	✓	✓	✓	✓
	LLDP	✓	✓	✓	✓	✓	✓
	Unicast-Filter	✓	✓	✓	✓	✓	✓
	Gesperrte Ports	✓	✓	✓	✓	✓	✓
	Unicast-Learning	✓	✓	✓	✓	✓	✓
	Unicast-Blocking	✓	✓	✓	✓	✓	✓
	Multicast-Gruppen	✓	✓	✓	✓	✓	✓
	IGMP	✓	✓	✓	✓	✓	✓
	GMRP	-	✓	✓	✓	-	✓
	Multicast-Blocking	✓	✓	✓	✓	✓	✓
	Broadcast-Blocking	✓	✓	✓	✓	✓	✓
	RMON	✓	✓	✓	✓	✓	✓
	RMON-History	-	✓	✓	✓	✓	✓
Layer 3-Funktionen	NAT/NAPT	-	✓	✓	-	-	✓ 3)
	Statische Routen	-	-	-	-	-	✓ 3)
	Route Maps	-	-	-	-	-	✓ 3)
	DHCP Relay Agent	✓	✓	✓	✓	✓	✓ 3)
	DHCPv6 Relay Agent	-	-	-	-	-	✓ 3)
	VRRP	-	-	-	-	-	✓ 3)
	VRRPv3	-	-	-	-	-	✓ 3)
	OSPFv2	-	-	-	-	-	✓ 3)
	OSPFv3	-	-	-	-	-	✓ 3)
	RIPv2	-	-	-	-	-	✓ 3)
	RIPng	-	-	-	-	-	✓ 3)
	IGMP	-	-	-	-	-	✓ 3)
	PIM	-	-	-	-	-	✓ 3)
	MSDP	-	-	-	-	-	✓ 3)
	Single-Hop Inter-VLAN-Routing	-	✓	✓	-	-	-
	Gemeinsame Agent-Adresse	-	✓	✓	-	-	-

		SCALANCE XB-200	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XR-300WG	SCALANCE XM-400/ SCALANCE XR-500
Security-Funktionen	Rollen	-	-	-	-	-	✓
	Gruppen	-	-	-	-	-	✓
	Passwörter	✓	✓	✓	✓	✓	✓
	RADIUS-Authentifizierung	✓	✓	✓	✓	✓	✓
	MAC-Authentifizierung	-	✓	✓	✓	-	✓
	Guest VLAN	-	✓	✓	✓	-	✓
	802.1X Reauthentifizierung	✓	✓	✓	✓	✓	✓
	MAC ACL	-	-	-	-	-	✓
	IP ACL	-	-	-	-	-	✓
	Management ACL	✓	✓	✓	✓	✓	✓

[1] nur SCALANCE XR528-6M und SCALANCE XR552-12M ab Firmware-Version 5.0

[2] eingeschränkt

[3] integriert oder mittels KEY-PLUG

[4] nicht bei DNA-Geräten

Verfügbarkeit der Hardware-Ausstattung

Die nachfolgende Tabelle zeigt die Hardware-Ausstattung der IE-Switches.

Technische Änderungen sind vorbehalten

	SCALANCE XB-200	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XR-300WG	SCALANCE XM-400/ SCALANCE XR-500
Modularität	-	-	-	-	-	✓ [1]
C-PLUG	-	✓	✓	✓	-	✓
KEY-PLUG	-	-	-	-	-	✓
Combo Ports	-	-	-	-	✓	✓
Stecktransceiver-Steckplätze	-	✓	-	-	-	✓
SELECT/SET-Taster	-	✓	✓	-	-	✓
RESET-Taster	✓	-	✓	-	✓	-
Meldekontakt	-	✓	✓	✓	-	✓
Serielle Schnittstelle	✓	✓	✓	-	✓	✓
Out-of-Band-Schnittstelle	-	-	-	-	-	✓

	SCA-LANCE XB-200	SCA-LANCE XC-200	SCA-LANCE XP-200	SCA-LANCE XF-200BA	SCA-LANCE XR-300WG	SCA-LANCE XM-400/ SCA-LANCE XR-500
Anzeigemodi	-	✓	✓	-	-	✓
NFC	-	✓	-	-	-	✓ [2]
Power over Ethernet	-	-	✓ [3]	-	-	✓

[1] SCALANCE XM-400 über Extender, SCALANCE XR-500 über Medienmodule

[2] nur SCALANCE XM-400

[3] Kennzeichnung "PoE" im Gerätenamen

Mengengerüste

Mengengerüst des Geräts

In der folgenden Tabelle ist das Mengengerüst für das Web Based Management und das Command Line Interface der Geräte aufgeführt.

Die Nutzbarkeit verschiedener Funktionen ist vom Gerätetyp und einem eventuell gesteckten KEY-PLUG abhängig.

	Konfigurierbare Funktion	Maximale Anzahl					
		SCALANCE XB-200	SCALANCE XR-300 WG	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XM-400/ SCALANCE XR-500
System	Maximale Framegröße (Ingress)	1632 Byte	1632 Byte	1632 Byte	1632 Byte	1632 Byte	9194 Byte (konfigurierbar)
	DNS-Server manual (IPv4/ IPv6)	-	-	-	-	-	3
	DNS-Server learned (IPv4)	-	-	-	-	-	2
	DNS-Server insgesamt	-	-	-	-	-	5
	Syslog-Server	3	3	3	3	3	3
	E-Mail-Server	3	3	3	3	3	3
	DHCP-Pools	16 [1)]	28 [1)]	24	24	24	24
	IPv4-Adressen pro DHCP-Pool	1	1	24	24	24	24
	IPv4-Adressen, die der DHCP-Server verwaltet (dynamisch + statisch)	16 [1)]	28 [1)]	576	576	576	576
	DHCP statische Zuordnungen pro DHCP-Pool	-	-	24	24	24	24
	SNMP-Trapempfänger	10	10	10	10	10	10
	SNTP-Server	1	1	1	1	1	2
	NTP-Server	1	1	1	1	1	3
	Agent-/TIA-Schnittstellen [2)]	1	1	1	1	1	1

	Konfigurierbare Funktion	Maximale Anzahl					
		SCALANCE XB-200	SCALANCE XR-300 WG	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XM-400/ SCALANCE XR-500
Layer 2	Virtuelle LANs (portbased; inclusive VLAN 1)	17	17	257	257	257	257
	Protokoll-basierte VLAN-Gruppen	-	-	-	-	-	12
	Protokoll-basierte VLAN-Gruppen pro Port	-	-	-	-	-	12
	IPv4 Subnetz-basierte VLANs	-	-	-	-	-	150
	IPv6 Subnetz-basierte VLANs	-	-	-	-	-	150
	Private VLAN	-	-	1	1	-	1
	Primary PVLANs	-	-	1	1	-	1
	Secondary Isolated PVLANs	-	-	24	24	-	24
	Secondary Community PVLANs	-	-	256	256	-	256
	Mirroring-Sessions	1	1	1	1	1	7
	VLANs, deren Datenverkehr auf einen Monitor-Port gespiegelt werden kann	-	-	-	-	-	255
	RSPAN-Sessions	-	-	-	-	-	1
	Standby-Ports	1	1	1	1	1	5
	Multiple Spanning Tree-Instanzen	-	-	4	4	-	16
	Linkaggregationen oder Etherchannels mit je max. 8 Ports pro Aggregation	-	-	8	8	4	8
	Ports in einer Linkaggregation	-	-	8	8	4	8
	Unicast Filtering	128	128	128	128	128	128
	Statische MAC-Adressen in der FDB (Forward Database)	128	128	128	128	128	256
	Multicast-Adressen ohne aktives GMRP	512	512	512	512	512	512
	Multicast-Adressen mit aktiviertem GMRP	-	-	50	50	50	50
	VLANs, deren Datenverkehr auf einen Monitor-Port gespiegelt werden kann	-	-	-	-	-	255

	Konfigurierbare Funktion	Maximale Anzahl					
		SCALANCE XB-200	SCALANCE XR-300 WG	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XM-400/ SCALANCE XR-500
Layer 3	IP-Schnittstellen	1	1	24	24	1	127
	Einträge in der HW-Routing-Tabelle	-	-	-	-	-	4096
	Statische Routen	-	-	-	-	-	100
	Mögliche Routen zum gleichen Ziel	-	-	-	-	-	8
	DHCP Relay Agent-Schnittstellen	1	1	24	24	1	127
	DHCP Relay Agent Server	4	4	4	4	4	4
	NAT-Schnittstellen	-	-	1	1	-	5
	VRRP Router-Schnittstellen (nur VLAN-Schnittstellen)	-	-	-	-	-	52
	OSPF Areas pro Gerät	-	-	-	-	-	5
	OSPF Area Range Einträge pro OSPF Area (Intra-Area Summary)	-	-	-	-	-	3
	OSPF-Schnittstellen	-	-	-	-	-	40
	OSPF-Schnittstellen pro OSPF Area	-	-	-	-	-	40
	OSPF Virtual Links (innerhalb eines autonomen Systems)	-	-	-	-	-	8
	OSPFv3-Nachbarn	-	-	-	-	-	300
	OSPFv3-Nachbarn pro Schnittstelle	-	-	-	-	-	8
	OSPFv3-Routen	-	-	-	-	-	1500
	OSPFv2-Schnittstelle Authentifizierungsschlüssel	-	-	-	-	-	200 (40 Schnittstellen mit je 5 Schlüsseln)
	OSPFv2 Virtual Link Authentifizierungsschlüssel	-	-	-	-	-	40 (8 Virtual Links mit je 5 Schlüsseln)
	PIM-Multicast-Routen pro Gerät [4)]	-	-	-	-	-	1000
	PIM-Komponenten	-	-	-	-	-	1

	Konfigurierbare Funktion	Maximale Anzahl					
		SCALANCE XB-200	SCALANCE XR-300WG	SCALANCE XC-200	SCALANCE XP-200	SCALANCE XF-200BA	SCALANCE XM-400/ SCALANCE XR-500
	Rendezvous Punkte	-	-	-	-	-	3
	Kandidaten für Rendezvous Punkte	-	-	-	-	-	3
	Statische Rendezvous Punkte	-	-	-	-	-	3
Security	Rollen	1	1	1	1	1	29
	Benutzer	30	30	30	30	30	18
	Gruppen	-	-	-	-	-	32
	IP-Adressen von RADIUS-Servern	4	4	4	4	4	4
	Gleichzeitige MAC-Authentifizierungen (authentifiziert und geblockt) pro Gerät [3]	4000	4000	4000	4000	4000	2000
	Gleichzeitige MAC-Authentifizierungen (authentifiziert und geblockt) pro Port (konfigurierbar) [3]	100	100	100	100	100	100
	Management ACLs (Zugriffsregeln für das Management)	10	10	10	10	10	10
	Regeln für Port ACL MAC	-	-	-	-	-	128
	Ingress- und Egress-Regeln für Port ACL MAC (gesamt)	-	-	-	-	-	364
	Regeln für Port ACL IP	-	-	-	-	-	128
	Ingress- und Egress-Regeln für Port ACL IP (gesamt)	-	-	-	-	-	364
	Regeln für VLAN ACL IP	-	-	-	-	-	128

[1] Beim SCALANCE XB-200 und SCALANCE XR-300WG ist die Anzahl der DHCP-Pools und verwaltbaren IPv4-Adressen von der Anzahl der Ports abhängig. Die Anzahl der Ports entspricht der maximalen Anzahl an DHCP-Pools und verwaltbaren IPv4-Adressen.

[2] Hierbei handelt es sich um eine IP-Schnittstelle.

[3] Wenn die maximale Anzahl der MAC-Authentifizierungen pro Gerät überschritten wird, werden alle MACAuthentifizierungen des Ports zurückgesetzt, an dem der Wert überschritten wurde.

[4] Die maximale Anzahl an PIM-Multicast-Routen pro Gerät setzt sich wie folgt zusammen: SSM-Streams + SM-Streams + bidirektionale Streams (von IGMP-Joins) + bidirektionale Routen (von RP-Multicastgruppen) Abhängig vom Aufbau des PIM-Netzwerks (hierarchischer Aufbau) können beim Bidirektionalen Multicast eine größere Anzahl an Streams unterstützt werden.

Wissenswertes

IP-Adressen

IPv4 / IPv6

Was sind die wesentlichen Unterschiede?

	IPv4	IPv6
IP-Konfiguration	• DHCP-Server • Manuell	• Stateless Address Autoconfiguration (SLAAC): zustandslose Autokonfiguration über NDP (Neighbor Discovery Protocol) – Erstellt eine Link-lokale Adresse für jede Schnittstelle, die keinen Router auf dem Link benötigt. – Prüft die Einmaligkeit der Adresse auf dem Link, die keinen Router auf dem Link benötigt. – Legt fest, ob die globalen Adressen über einen zustandslosen Mechanismus, einen zustandbehafte Mechanismus oder über beide Mechanismen bezogen werden. (Erfordert einen Router auf dem Link.) • Manuell • DHCPv6 (Zustandbehaftet)
Verfügbare IP-Adressen	32-Bit: 4, 29 * 10^9 Adressen	128-Bit: 3,4 * 10^{38} Adressen
Adressformat	Dezimal: 192.168.1.1 mit Port: 192.168.1.1:20	Hexadezimal: 2a00:ad80::0123 mit Port: [2a00:ad80::0123]:20
Loopback	127.0.0.1	::1
IP-Adressen pro Schnittstelle	4 IP-Adressen	Mehrere IP-Adressen • LLA: Eine link lokale-Adresse (automatisch gebildet) fe80::/128 pro Schnittstelle • ULA: Mehrere Unique lokal Unicast-Adressen pro Schnittstelle • GUA: Mehrere Globale Unicast-Adressen pro Schnittstelle
Header	• Checksumme • variable Länge • Fragmentierung im Header • keine Sicherheit	• Überprüfung auf höherer Schicht • fest vorgeschriebene Größe • Fragmentierung im Extension Header
Fragmentierung	Host und Router	nur Endpunkt der Kommunikation
Quality of Service	Type of Service zur Priorisierung (ToS)	Die Priorisierung wird im Header-Feld "Traffic Class" angegeben.
Telegrammarten	Broadcast, Multicast, Unicast	Multicast, Unicast, Anycast

	IPv4	IPv6
Identifizierung von DHCP-Clients/-Server	Client-ID: • MAC-Adresse • DHCP Client ID • System Name • PROFINET-Stationsname • IAID und DUID	DUID + IAID(s) = genau eine Schnittstelle des Hosts DUID = DHCP Unique Identifier Identifiziert Server und Clients eindeutig und sollte sich nicht ändern, auch nicht beim Austausch von Netzwerkkomponenten! IAID = Identity Association Identifier Mindestens eine pro Schnittstelle wird vom Client generiert und bleibt bei Neustart des DHCP-Clients unverändert Drei Verfahren zum Ermitteln der DUID • DUID-LLT • DUID-EN • DUID-LL
DHCP	über UDP mittels Broadcast	über UDP mittels Unicast RFC 3315, RFC 3363 **Stateful DHCPv6** Zustandsbehaftete Konfiguration bei der die IPv6-Adresse und die Konfigurationseintellungen übermittelt werden. Dabei werden vier DHCPv6-Nachrichten zwischen Client und Server ausgetauscht: 1. SOLICIT: Gesendet durch den DHCPv6-Client, um DHCPv6-Server zu lokalisieren. 2. ADVERTISE Die verfügbaren DHCPv6-Server antworten darauf. 3. REQUEST Der DHCPv6-Client fordert eine IPv6-Adresse und die Konfigurationseinstellungen beim DHCPv6-Server an. 4. REPLY Der DHCPv6-Server sendet die IPv6-Adresse und die Konfigurationseinstellungen. Wenn Client und Server die Funktion "Rapid commit" unterstützen, wird das Verfahren auf zwei DHCPv6-Nachrichten SOLICIT und REPLY gekürzt. **Stateless DHCPv6** Beim zustandslosen DHCPv6 werden nur die Konfigurationseinstellungen übermittelt. **Präfix-Delegation** Der DHCPv6-Server delegiert das Verteilen von IPv6-Präfixe an den DHCPv6-Client. Der DHCPv6-Client wird auch als PD-Router bezeichnet.
Auflösung von IP-Adressen in Hardware-Adressen	ARP (Address Resolution Protocol)	NDP (Neighbor Discovery Protocol)

IPv4-Adresse

Aufbau einer IPv4-Adresse

Adressklassen

IP-Adressbereich	Max. Anzahl der Netzwerke	Max. Anzahl Hosts/Netzwerk	Klasse	CIDR
1.x.x.x bis 126.x.x.x	126	16777214	A	/8
128.0.x.x bis 191.255.x.x	16383	65534	B	/16
192.0.0.x bis 223.255.255.x	2097151	254	C	/24
224.0.0.0 - 239.255.255.255	Multicast-Anwendungen		D	
240.0.0.0 - 255.255.255.255	reserviert für zukünftige Anwendungen		E	

Eine IP-Adresse besteht aus 4 Byte. Jedes Byte wird dezimal dargestellt und ist durch einen Punkt vom vorherigen getrennt. Es ergibt sich also folgender Aufbau, wobei für XXX eine Zahl zwischen 0 und 255 zu setzen ist:

XXX.XXX.XXX.XXX

Die IP-Adresse besteht aus zwei Teilen, der Netzwerkadresse und der Endteilnehmeradresse. Dadurch ist es möglich, verschiedene Teilnetze zu bilden. Abhängig davon, welche Bytes der IP-Adresse als Netzwerkadresse und welche als Endteilnehmeradresse genutzt werden, kann eine IP-Adresse einer bestimmten Adressklasse zugeordnet werden.

Subnetzmaske

Die Bits der Endteilnehmer-Adresse können für die Bildung von Subnetzen verwendet werden. Dabei stellen die führenden Bits die Adresse des Subnetzes dar, die restlichen Bits werden als Adresse des Rechners im Subnetz interpretiert.

Ein Subnetz wird durch die Subnetzmaske definiert. Der Aufbau der Subnetzmaske entspricht dem einer IP-Adresse. Ist in der Subnetzmaske an einer Bitposition eine "1" gesetzt, gehört das Bit an der entsprechenden Stelle in der IP-Adresse zur Subnetzadresse, andernfalls zur Adresse des Rechners.

Beispiel für ein Klasse B-Netz:

Die Standard-Subnetz-Adresse für Klasse B-Netze ist 255.255.0.0, es stehen also die letzten beiden Bytes für die Festlegung eines Subnetzes zur Verfügung. Wenn 16 Teilnetze definiert werden sollen, muss das dritte Byte der Subnetzadresse auf 11110000 (Binärdarstellung) gesetzt werden. In diesem Fall ergibt sich die Subnetzmaske 255.255.240.0.

Um festzustellen, ob zwei IP-Adressen zum gleichen Subnetz gehören, werden auf die beiden IP-Adressen und die Subnetzmaske eine bitweise UND-Verknüpfung angewendet. Wenn beide Verknüpfungen das gleiche Ergebnis haben, gehören beide IP-Adressen zum gleichen Subnetz, wie z. B. 141.120.246.210 und 141.120.252.108.

Außerhalb des lokalen Netzwerks ist die beschriebene Aufteilung der Endteilnehmer-Adresse ohne Bedeutung, dort ist für die Paketvermittlung nur die IP-Adresse in ihrer Gesamtheit von Interesse.

Hinweis

In der Bit-Darstellung der Subnetzmaske müssen die "Einsen" linksbündig gesetzt sein, d. h. es dürfen keine "Nullen" zwischen den "Einsen" stehen.

IPv6-Adressen

IPv6-Begriffe

Netzknoten

Ein Netzknoten ist ein Gerät, das über ein oder mehrere Schnittstellen an ein oder mehrere Netzwerke angeschlossen ist.

Router

Ein Netzknoten, der IPv6-Pakete weiterleitet.

Host

Ein Netzknoten, der einen Endpunkt für IPv6-Kommunikationsbeziehungen darstellt.

Link

Als Link wird in der IPv6-Terminologie eine direkte Layer 3-Verbindung innerhalb eines IPv6-Netzwerks bezeichnet.

Nachbar

Zwei Netzknoten werden als Nachbarn bezeichnet, wenn sie sich auf dem gleichen Link befinden.

IPv6-Schnittstelle

Physische oder logische Schnittstelle, an der IPv6 aktiviert ist.

Path MTU

Maximal zulässige Paketgröße auf einem Pfad, von einem Absender zu einem Empfänger.

Path MTU Discovery

Mechanismus zur Bestimmung der maximal zulässigen Paketgröße entlang des gesamten Pfades von einem Absender zu einem Empfänger.

LLA

Link lokale-Adresse FE80::/10

Sobald IPv6 auf der Schnittstelle aktiviert wird, wird automatisch eine link lokale-Adresse gebildet. Nur erreichbar für Knoten, die sich auf dem gleichen Link befinden.

ULA

Unique lokal-Adresse

Definiert in der RFC 4193. Über diese Adresse ist das IPv6-Schnittstelle im LAN erreichbar.

GUA

Globale Unicast-Adresse. Über diese Adresse ist IPv6-Schnittstelle z. B. über das Internet erreichbar.

Schnittstellen-ID

Die Schnittstellen-ID wird mit dem EUI-64 Verfahren oder manuell gebildet.

EUI-64

Extended Unique Indentifier (RFC 4291); Verfahren zur Bildung der Schnittstellen-ID. Bei Ethernet wird die Schnittstellen-ID aus der MAC-Adresse der Schnittstelle gebildet. Teilt die MAC-Adresse in den herstellerspezifischen Teil (OUI) und den netzwerkspezifischen Teil (NIC) und fügt zwischen den beiden Teilen FFFE ein.

Beispiel:

MAC-Adresse = AA:BB:CC:DD:EE:FF

OUI = AA:BB:CC

NIC = DD:EE:FF

EUI-64 = OUI + **FFFE** + NIC = AA:BB:CC:**FF:FE**:DD:EE:FF

Scope

Definiert die Reichweite der IPv6-Adresse.

Aufbau einer IPv6-Adresse

Adressformat IPv6-Notation

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vierstelligen Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt.

Beispiel:

fd00:0000:0000:ffff:02d1:7d01:0000:8f21

Regeln/Vereinfachungen:

- Wenn ein oder mehr Blöcke den Wert 0 haben, ist eine verkürzte Schreibweise möglich. Die Adresse fd00:**0000:0000**:ffff:02d1:7d01:0000:8f21 kann verkürzt auch wie folgt notiert werden:
  fd00**::**ffff:02d1:7d01:0000:8f21
  Damit Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden.
- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.
  Die Adresse fd00:0000:0000:ffff:**02d1**:7d01:0000:8f21 kann verkürzt auch wie folgt notiert werden:
  fd00**::**ffff:**2d1**:7d01:0000:8f21
- Dezimalnotation mit Punkt-Schreibweise
  Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.
  Beispiel: Die IPv6-Adresse fd00::ffff.125.1.0.1 ist äquivalent zu fd00::ffff:7d01:1

Aufbau IPv6-Adresse

Das IPv6-Protokoll unterscheidet drei Adressarten: Unicast , Anycast und Multicast. Der folgende Abschnitt beschreibt den Aufbau der globalen Unicast-Adressen.

IPv6-Präfix		Suffix
Globales Präfix: n Bit	Subnetz-ID m Bit	Schnittstelle-ID 128 - n - m Bits
Zugewiesener Adress-bereich	Beschreibung des Standorts, auch Subnetzpräfix oder Sub-netz	Eindeutige Zuordnung des Host im Netz. Die ID wird aus der MAC-Adresse gene-riert.

Das Präfix für die link local-Adresse ist immer fe80:0000:0000:0000. Das Präfix wird verkürzt und wie folgt notiert: fe80::

IPv6-Präfix

Festgelegt in: RFC 4291

Das IPv6-Präfix stellt die Subnetzkennung dar.

Präfixe und IPv6-Adressen werden auf dieselbe Weise angegeben wie bei der CIDR-Notation (Classless Inter-Domain Routing) für IPv4.

Aufbau

IPv6-Adresse / Präfixlänge

Beispiel

IPv6-Adresse: 2001:0db8:1234::1111/48

Präfix: 2001:0db8:1234::/48

Schnittstelle-ID: ::1111

Eingabe und Darstellung

Die Eingabe von IPv6-Adressen ist in den zuvor beschriebenen Notationen möglich. IPv6-Adressen werden immer in der hexadezimalen Notation angezeigt.

PROFINET

PROFINET

PROFINET ist ein offener Ethernet-Standard (IEC 61158/61784) für die industrielle Automatisierung basierend auf Industrial Ethernet. PROFINET nutzt existierende IT-Standards und ermöglicht eine durchgängige Kommunikation von der Feldebene bis in die Leitebene sowie ein anlagenweites Engineering. Weitere Eigenschaften von PROFINET sind:

- Nutzung von TCP/IP
- Automatisierung von Applikationen mit Echtzeit-Bedarf
  - Real-Time (RT)-Kommunikation
  - Isochronous Real-Time (IRT)-Kommunikation
- Nahtlose Integration von Feldbus-Systemen

PROFINET konfigurieren Sie unter "System > PROFINET (Seite 2248)".

PROFINET IO

Im Rahmen von PROFINET ist PROFINET IO ein Kommunikationskonzept für die Realisierung modularer, dezentraler Applikationen. Die Umsetzung von PROFINET IO wird durch den PROFINET-Standard für Automatisierungsgeräte (IEC 61158-x-10) realisiert.

EtherNet/IP

EtherNet/IP

EtherNet/IP (Ethernet/Industrial Protocol) ist ein offener Industriestandard für industrielles Echtzeit-Ethernet, basierend auf TCP/IP und UDP/IP. Mit EtherNet/IP wird Ethernet um das Common Industrial Protocol (CIP) auf der Anwendungsschicht erweitert. In EtherNet/IP werden die unteren Schichten des OSI-Referenzmodells von Ethernet mit den Übertragungs-, Vermittlungs-, Netzwerk- und Transportfunktionen übernommen.

EtherNet/IP konfigurieren Sie unter "System > EtherNet/IP (Seite 2250)".

Common Industrial Protocol

Das Common Industrial Protocol (CIP) ist ein Anwendungsprotokoll der Automatisierung, das den Übergang der Feldbusse in industrielles Ethernet und in IP-Netze unterstützt. Dieses Industrieprotokoll benutzen Feldbusse/Industrienetzwerke wie DeviceNet, ControlNet und EtherNet/IP in der Anwendungsschicht als Schnittstelle zwischen der deterministischen Feldbus-Welt und der Automatisierungsapplikation (Steuerung, E/A, HMI, OPC, ...). Das CIP liegt oberhalb der Transportschicht und erweitert die reinen Transportdienste um Kommunikationsdienste für die Automatisierungstechnik. Dazu gehören Dienste für den zyklischen, den zeitkritischen und den ereignisgesteuerten Datenverkehr. CIP unterscheidet zwischen den zeitkritischen E/A-Nachrichten (implicit messages) und individuellen Frage/Antwort-Telegrammen zur Konfiguration und Datenerfassung (explicit messages). CIP ist objekt-orientiert; alle von außen "sichtbaren" Daten sind in Form von Objekten zugänglich. CIP hat eine gemeinsame Konfigurationsgrundlage: EDS (Electronic Data Sheet).

Electronic Data Sheet

Electronic Data Sheet (EDS) ist ein elektronisches Datenblatt zur Beschreibung von Geräten.

Das für den EtherNet/IP-Betrieb benötigte EDS finden Sie unter "System > Laden & Speichern".

VLAN

VLAN

Netzwerkdefinition unabhängig von der räumlichen Lage der Teilnehmer

VLAN (Virtual Local Area Network) teilt ein physikalisches Netzwerk in mehrere logische Netzwerke, die voneinander abgeschirmt sind. Hierbei werden Geräte zu logischen Gruppen zusammengefasst. Nur Teilnehmer des gleichen VLANs können sich untereinander adressieren. Da auch Multicast- und Broadcast-Telegramme nur innerhalb des jeweiligen VLANs weitergeleitet werden, wird von Broadcast-Domänen gesprochen.

Daraus ergibt sich als besonderer Vorteil von VLANs eine geringere Netzlast für die Teilnehmer bzw. Netzsegmente anderer VLANs.

Für die Kennung, welches Paket welchem VLAN zugeordnet ist, wird das Telegramm um 4 Byte erweitert (VLAN-Tagging). Diese Erweiterung enthält neben der VLAN-ID auch Prioritätsinformationen.

Möglichkeiten der VLAN-Zuordnung

Es gibt verschiedene Möglichkeiten der Zuordnung zu VLANs:

- Port-basiertes VLAN
  Jedem Port eines Geräts wird eine VLAN-ID zugewiesen. Port-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".
- Protokoll-basiertes VLAN
  Jedem Port eines Geräts wird eine Protokollgruppe zugewiesen. Protokoll-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > Protokoll-basierter VLAN-Port"

- IPv4 Subnetz-basiertes VLAN
  Der IPv4-Adresse des Geräts wird eine VLAN-ID zugewiesen. Subnetz-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > IPv4 Subnetz-basiertes VLAN".
- IPv6 Präfix-basiertes VLAN
  Der IPv6-Adresse des Geräts wird eine VLAN-ID zugewiesen. Präfix-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > IPv6 Präfix-basiertes VLAN".

Abarbeitung der VLAN-Zuordnung

Wenn auf dem Gerät mehrere VLAN-Zuordnungen angelegt sind, werden die Zuordnungen in folgender Reihenfolge abgearbeitet:

1. IPv4 Subnetz-basiertes VLAN/IPv6 Präfix-basiertes VLAN
2. Protokoll-basiertes VLAN
3. Port-basiertes VLAN

Das Telegramm wird zuerst nach der IP-Adresse untersucht. Wenn im Register "IPv4 Subnetz-basiertes VLAN" bzw. "IPv6 Präfix-basiertes VLAN" eine Regel zutrifft, wird das Telegramm in das entsprechende VLAN gesendet. Wenn keine Regel zutrifft, wird das Telegramm nach seinem Protokolltyp untersucht. Trifft im Register "Protokoll-basiertes VLAN-Port" eine Regel zu, wird das Telegramm in das entsprechende VLAN gesendet. Wenn keine Regel zutrifft, wird das Telegramm über das Port-basierte VLAN gesendet. Die Regeln für das Port-basierte VLAN werden im Register "Port-basiertes VLAN" festgelegt.

VLAN-Tagging

Erweiterung der Ethernet-Telegramme um vier Byte

Für CoS (Class of Service, Telegrammpriorisierung) und für VLAN (Virtuelles Netzwerk) wurde in der Norm IEEE 802.1Q die Erweiterung der Ethernet-Telegramme um das VLAN-Tag festgelegt.

Hinweis

Durch das VLAN-Tag erhöht sich die zulässige Gesamtlänge des Telegramms von 1518 auf 1522 Byte.
Es muss geprüft werden, ob die Endteilnehmer im Netz diese Länge / diesen Telegrammtyp verarbeiten können. Ist dies nicht der Fall, dürfen an diese Teilnehmer nur Telegramme mit der Standardlänge gesendet werden.

Die zusätzlichen 4 Bytes befinden sich im Header des Ethernet-Telegramms zwischen der Quelladresse und dem Ethernet-Typ-/Längenfeld:

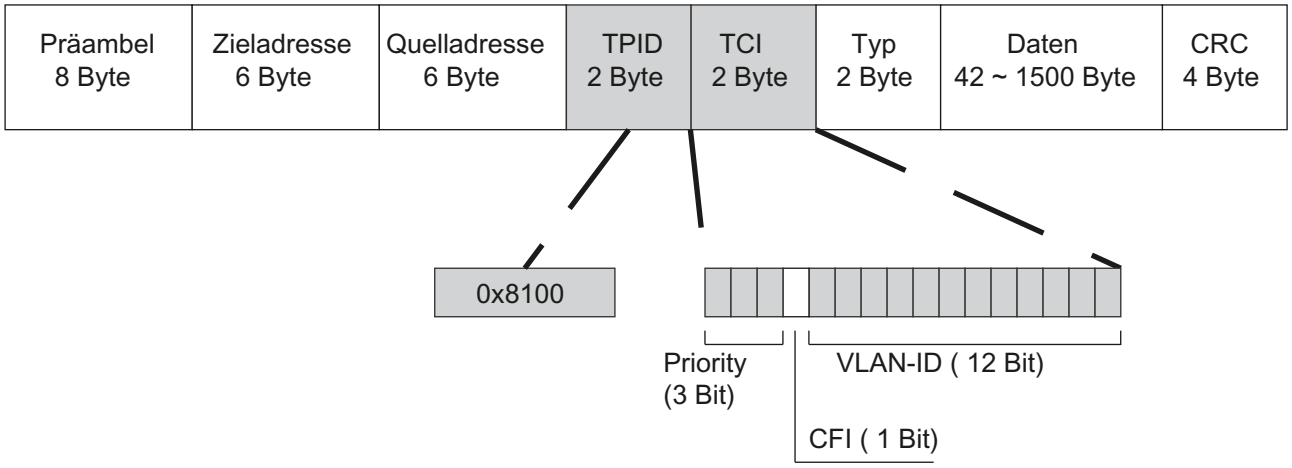

Bild 1-82 Aufbau des erweiterten Ethernet-Telegramms

Die zusätzlichen Bytes beinhalten den Tag Protocol Identifier (TPID) und die Tag Control Information (TCI).

Tag Protocol Identifier (TPID)

Die ersten 2 Bytes bilden den Tag Protocol Identifier (TPID) und sind fest mit 0x8100 belegt. Dieser Wert gibt an, dass das Datenpaket VLAN-Informationen oder Prioritätsangaben beinhaltet.

Tag Control Information (TCI)

Die 2 Bytes der Tag Control Information (TCI) beinhalten folgende Informationen:

CoS-Priorisierung

In dem getaggten Telegramm gibt es 3 Bits für die Priorität, die auch als Class of Service (CoS) bezeichnet werden, siehe auch IEEE 802.1Q.

CoS-Bits	Priorität	Art des Datenverkehrs
000	0 (niedrigste)	Background (Hintergrund)
001	1	Best Effort
010	2	Excellent Effort
011	3	Critical Applications (Kritische Anwendungen)
100	4	Video, < 100 ms Verzögerung (Latenz und Jitter)
101	5	Voice (Sprache), < 10 ms Verzögerung (Latenz und Jitter)
110	6	Internetwork Control
111	7 (höchste)	Network Control

Die Priorisierung der Datenpakete setzt eine Warteschlange in den Komponenten voraus, in der sie die Datenpakete mit der niedrigeren Priorität puffern können.

Das Gerät besitzt mehrere parallele Warteschlangen, in denen die verschieden priorisierten Telegramme abgearbeitet werden. Standardmäßig werden zuerst die Telegramme mit der höchsten Priorität abgearbeitet. Dieses Verfahren gewährleistet auch bei einem hohen

Datenaufkommen, dass die Telegramme mit der höchsten Priorität auf jeden Fall gesendet werden.

Canonical Format Identifier (CFI)

Der CFI wird für die Kompatibilität zwischen Ethernet und Token Ring benötigt.
Die Werte haben folgende Bedeutung:

Wert	Bedeutung
0	Das Format der MAC-Adresse ist kanonisch. Bei kanonischer Darstellung der MAC-Adresse wird das niederwertigste Bit zuerst übertragen. Standardeinstellung für Ethernet-Switches.
1	Das Format der MAC-Adresse ist nicht kanonisch.

VLAN-ID

Im 12 Bit-Datenfeld können bis zu 4096 VLAN-IDs gebildet werden. Dabei gelten folgende Festlegungen:

VLAN-ID	Bedeutung
0	Das Telegramm beinhaltet nur Prioritätsinformation (Priority Tagged Frames) und keine gültige VLAN-Kennung.
1 - 4094	Gültige VLAN-Kennung, das Telegramm ist einem VLAN zugeordnet, es kann zusätzlich auch Prioritätsinformationen beinhalten.
4095	Reserviert

Private VLAN

Mit einem Private VLAN (PVLAN) können Sie die Layer 2-Broadcastdomäne eines VLANs unterteilen.

Ein Private VLAN besteht aus folgenden Einheiten:

- Einem Primary Private VLAN (Primary PVLAN)
  Das VLAN, das unterteilt wird, wird Primary Private VLAN genannt.
- Secondary Private VLANs (Secondary PVLAN)
  Secondary PVLANs existieren nur innerhalb eines Primary PVLANs. Jedes Secondary PVLAN hat eine spezifische VLAN-ID und ist mit dem Primary PVLAN verbunden. Secondary PVLANs werden in folgende Typen unterteilt:
  - Isolated Secondary PVLAN
    Geräte innerhalb eines Isolated Secondary PVLANs können nicht über Layer 2 miteinander kommunizieren.
  - Community Secondary PVLAN
    Geräte innerhalb eines Community Secondary PVLANs können über Layer 2 direkt miteinander kommunizieren. Die Geräte können nicht mit Geräten in anderen Communities des PVLANs über Layer 2 kommunizieren.

Hinweis

VLAN-ID bei Secondary PVLANs

Wenn Sie auf unterschiedlichen IE-Switches die gleiche VLAN-ID für Secondary PVLANs verwenden, können die Endgeräte in diesen Secondary PVLANs switchübergreifend über Layer 2 miteinander kommunizieren.

Hinweis

Private VLAN-Funktionalität und RADIUS-Authentifizierung

Wenn die VLAN-Zuweisung über RADIUS-Authentifizierung für einen oder mehrere Ports eines VLAN aktiviert ist, sollten Sie dieses VLAN nicht zusätzlich als Private VLAN konfigurieren.

Die Private VLAN-Funktionalität in Zusammenhang mit der VLAN-Zuweisung über RADIUS-Authentifizierung kann zu einem inkonsistenten Systemzustand führen.

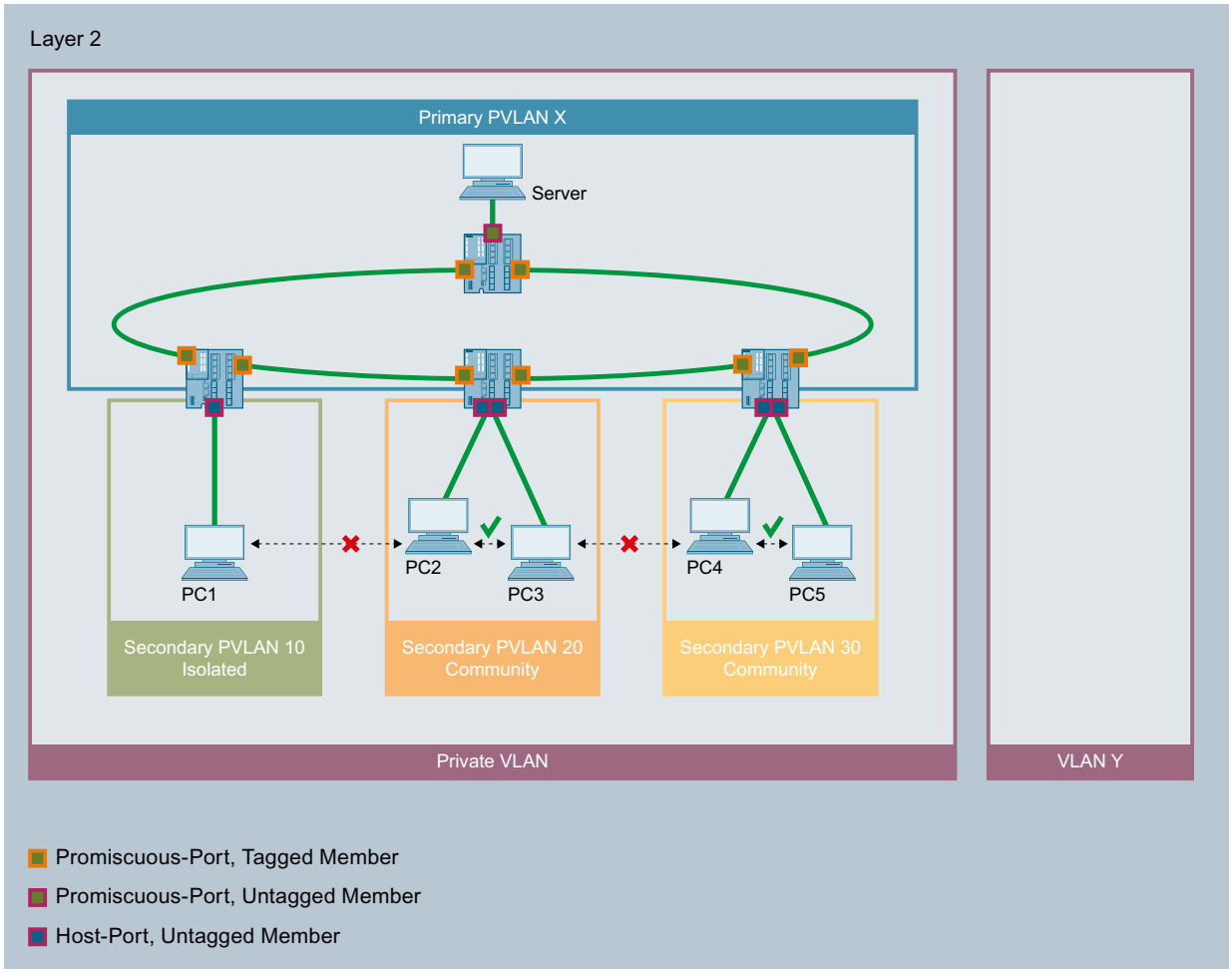

In diesem Beispiel sind die Ports der IE-Switches, die sie mit anderen IE-Switches verbinden, Promiscuous-Ports. Diese Netzwerkports sind Tagged Member in allen PVLANs: Primary PVLAN und allen Secondary PVLANs.

Die Ports, an denen die PCs angeschlossen sind, sind Host-Ports. Die Host-Ports sind jeweils Untagged Member im Primary PVLAN und in ihrem entsprechenden Secondary PVLAN.

Der Port, an dem der Server angeschlossen ist, ist ein Promiscuous Port. Dieser Promiscuous Port ist Untagged Member in allen PVLANs: Primary PVLAN und allen Secondary PVLANs.

In diesem Beispiel können alle PCs mit dem Server kommunizieren. Der Server kann mit allen PCs kommunizieren. Der PC1 kann mit keinem anderen PC kommunizieren. Die PCs innerhalb eines Community Secondary PVLANs können miteinander kommunizieren, jedoch nicht mit den PCs in einem anderen Secondary PVLAN.

VLAN-Tunnel

Mit der Funktion Q-in-Q VLAN-Tunnel ist es möglich, den Datenverkehr von verschiedenen Customer-Netzwerken mittels VLAN-Tunnel über ein Provider-Netzwerk weiterzuleiten. Dabei kann jedes Customer-Netzwerk über die volle Anzahl an möglichen VLANs verfügen.

Ein VLAN-Tunnel wird zwischen Provider-Switches aufgebaut, die an den Grenzen eines Provider-Netzwerks konfiguriert sind. Ein Provider-Switch verfügt über die folgenden Typen von Ports:

- Access-Port
  Über einen Access-Port ist der Provider-Switch mit einem Customer-Netzwerk verbunden.
  - Eingehender Datenverkehr
    Der eingehende Datenverkehr an einem Access-Port wird behandelt, als wäre er ungetagged ①. Alle eingehenden Telegramme werden um ein Tag mit der Port-VID des Access-Ports erweitert ②. Bei bereits getaggten Telegrammen bedeutet dies, dass sie um ein zweites 802.1Q-Tag erweitert werden ③, das äußere VLAN-Tag.

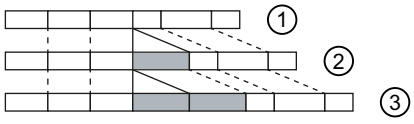

  - Ausgehender Datenverkehr
    Bei dem ausgehenden Datenverkehr an einem Access-Port wird das äußere VLAN-Tag wieder entfernt.
- Core-Port
  Über einen Core-Port ist der Provider-Switch mit einem Provider-Netzwerk verbunden. Core-Ports sind Mitglieder im Port-VLAN des Access-Ports oder mit dem Port-Typ "Switch-Port VLAN Trunk" konfiguriert.

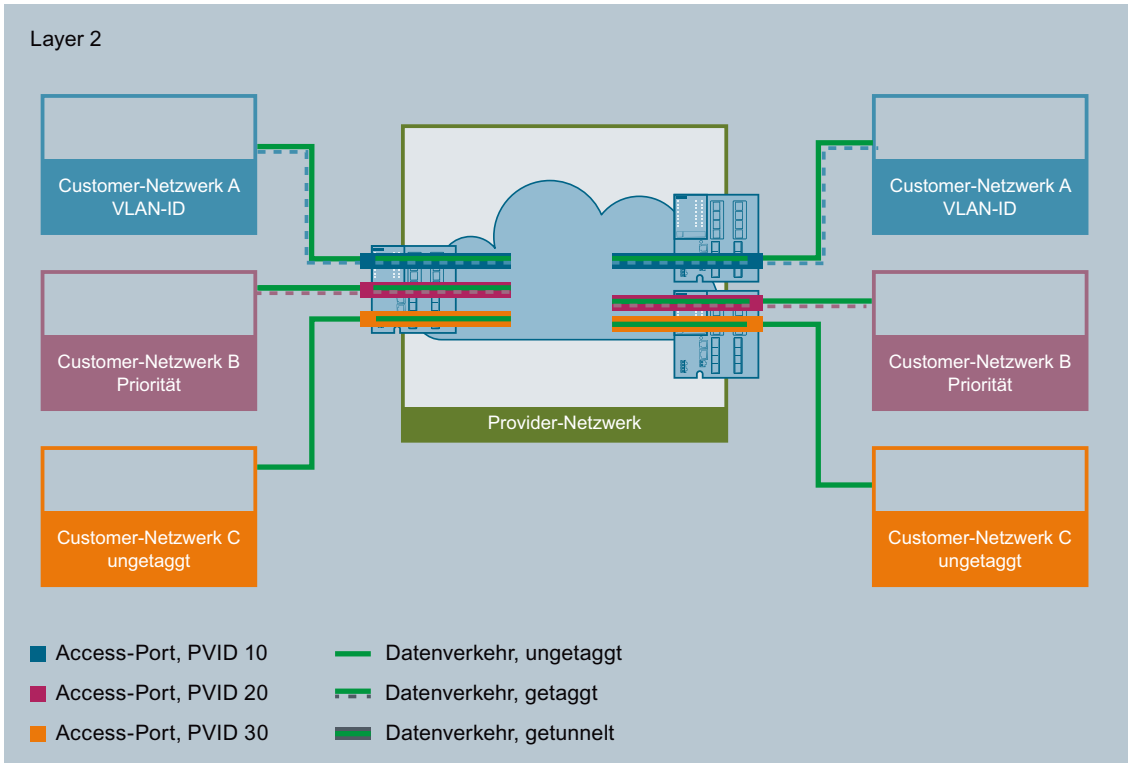

In diesem Beispiel werden der Datenverkehr aus den Custmer-Nertzwerken A, B und C mittels VLAN-Tunnel über das Provider-Netzwerk weitergeleitet. Die Telegramme aus dem Customer-Netzwerk A sind mit einer VLAN-ID getaggt. Die Telegramme aus dem Customer-Netzwerk B sind mit einer Priorität getaggt. Die Telegramme aus dem Customer-Netzwerk C sind ungetaggt.

Wenn die Telegramme den entsprechenden Access-Port erreichen, werden sie um ein Tag mit der Port-VID des Access-Ports erweitert und durch das Provider-Netzwerk getunnelt. Sobald die Telegramme das Provider-Netzwerk velassen, wird das äußere VLAN-Tag (PVID) wieder entfernt. Die Telegramme werden in ihrer ursprünglichen Form weitergeleitet. Die Priorität des Telegramms bleibt dabei erhalten.

SNMP

Einleitung

Mit Hilfe des Simple Network Management Protocol (SNMP) überwachen und steuern Sie Netzwerkkomponenten, z. B. Router oder Switches, von einer zentralen Station aus. SNMP regelt dabei die Kommunikation zwischen den überwachten Geräten und der Überwachungsstation.

Aufgaben von SNMP:

- Überwachung von Netzwerkkomponenten
- Fernsteuerung und Fernparametrierung von Netzwerkkomponenten
- Fehlererkennung und Fehlerbenachrichtigung

In den Versionen v1 und v2c verfügt SNMP über keine Sicherheitsmechanismen. Jeder Nutzer im Netzwerk kann mit geeigneter Software auf die Daten zugreifen und auch Parametrierungen verändern.

Für die einfache Steuerung von Zugriffsrechten ohne Sicherheitsaspekte werden Community-Strings verwendet.

Der Community-String wird zusammen mit der Anfrage übertragen. Wenn der Community-String korrekt ist, antwortet der SNMP-Agent und sendet die geforderten Daten. Wenn der Community-String nicht korrekt ist, verwirft der SNMP-Agent die Anfrage. Für Lese- und Schreibrechte definieren Sie verschiedene Community-Strings. Die Community-Strings werden in Klartext übertragen.

Standardwerte der Community-Strings:

- public
  besitzt nur Leserechte
- private
  besitzt Lese- und Schreibrechte

Hinweis

Da es sich bei den SNMP-Community Strings um einen Zugriffsschutz handelt, verwenden Sie nicht die Standardwerte "public" oder "private". Ändern Sie diese Werte nach der Erst-Inbetriebnahme.

Weitere einfache Schutzmechanismen auf Geräteebene:

- Allowed Host
  Dem überwachten System sind die IP-Adressen der überwachenden Systeme bekannt.
- Read Only
  Wenn Sie einem überwachten Gerät "Read Only" zuweisen, können Überwachungsstationen nur Daten auslesen, aber nicht ändern.

SNMP-Datenpakete sind nicht verschlüsselt und können einfach mitgelesen werden.

Die zentrale Station wird auch als Management-Station bezeichnet. Auf den zu überwachenden Geräten ist ein SNMP-Agent installiert, mit dem die Management-Station Daten austauscht.

Die Management-Station sendet Datenpakete folgenden Typs:

- GET
  Anfordern eines Datensatzes vom SNMP-Agent
- GETNEXT
  Ruft den nächsten Datensatz auf.
- GETBULK (verfügbar ab SNMPv2c)
  Fordert mehrere Datensätze auf einmal an, z. B. mehrere Zeilen einer Tabelle.
- SET
  Beinhaltet Parametrierungsdaten für das entsprechende Gerät.

Der SNMP-Agent sendet Datenpakete folgenden Typs:

- RESPONSE
  Der SNMP-Agent sendet die vom Manager angeforderten Daten zurück.
- TRAP
  Wenn ein bestimmtes Ereignis eintritt, sendet der SNMP-Agent eigenständig Traps.

SNMPv1/v2c/v3 verwenden UDP (User Datagram Protocol) und nutzen die UDP-Ports 161 und 162. Die Beschreibung der Daten erfolgt in einer Management Information Base (MIB).

SNMPv3

SNMPv3 führt gegenüber den Vorgängerversionen SNMPv1 und SNMPv2c ein umfangreicheres Sicherheitskonzept ein.

SNMPv3 unterstützt:

- Vollständig verschlüsselte Benutzerauthentifizierung
- Verschlüsselung des gesamten Datenverkehrs
- Zugriffskontrolle der MIB-Objekte auf Benutzer-/Gruppenebene

Mit der Einführung von SNMPv3 können Sie Benutzerkonfigurationen nicht mehr ohne Weiteres auf andere Geräte übertragen, z. B. indem Sie eine Konfigurationsdatei laden oder den C-PLUG austauschen.

Das SNMPv3-Protokoll verwendet gemäß des Standards eine eindeutige SNMP-Engine-ID als internen Bezeichner für einen SNMP-Agenten. Diese ID muss im Netzwerk eindeutig sein. Sie wird verwendet, um die Zugangsdaten von SNMPv3-Benutzern zu authentifizieren und zu verschlüsseln.

Abhängig davon, ob Sie die Funktion "SNMPv3 Benutzermigration" aktiviert oder deaktiviert haben, wird die SNMP-Engine-ID unterschiedlich generiert.

Einschränkung bei der Verwendung der Funktion

Verwenden Sie die Funktion "SNMPv3 Benutzermigration" nur, um im Ersatzteilfall Ihre konfigurierten SNMPv3-Benutzer auf ein Ersatzgerät zu übertragen.
Verwenden Sie die Funktion nicht, um konfigurierte SNMPv3-Benutzer auf mehrere Geräte zu übertragen. Wenn Sie eine Konfiguration mit angelegten SNMPv3-Benutzern in mehrere Geräte laden, verwenden diese Geräte hierdurch die gleiche SNMP-Engine-ID. Wenn Sie diese Geräte im gleichen Netzwerk verwenden, widerspricht Ihre Konfiguration dem SNMP-Standard.

Kompatibilität mit Vorgängerprodukten

Sie können SNMPv3-Benutzer nur auf ein anderes Gerät übertragen, wenn Sie die Benutzer als migrierbare Benutzer erstellt haben. Um einen migrierbaren Benutzer zu erstellen, muss die Funktion "SNMPv3 Benutzermigration" aktiviert sein, wenn Sie den Benutzer erstellen.

RCDP

Sie konfigurieren RCDP (RUGGEDCOM Discovery Protocol) auf der Seite "System > Konfiguration".

Wenn RCDP auf dem Gerät entsprechend konfiguriert ist, können Sie mit dem RUGGEDCOM EXPLORER über die Out-of-Band-Schnittstelle auf das Gerät zugreifen.

RUGGEDCOM EXPLORER

Mit dem RUGGEDCOM EXPLORER können Sie Geräte in einem Netzwerk erkennen und folgende Grundeinstellungen vornehmen:

- IP-Adresse
- Subnetzmaske
- Default-Gateway
- Systemname
- Gerätestandort
- Kontaktperson

Der RUGGEDCOM EXPLORER ist auf jedem LAN-basierten MS-Windows-PC lauffähig.

Sie finden weitere Informationen zum RUGGEDCOM EXPLORER hier:

- Auf den Internetseiten des Siemens Industry Online Support
- Auf den Internetseiten der RUGGEDCOM-Produkte

Redundanzverfahren

Spanning Tree

Vermeidung von Schleifenbildung bei redundanten Verbindungen

Das Spanning Tree-Verfahren ermöglicht es, Netzwerkstrukturen aufzubauen, bei denen es mehrere Verbindungen zwischen zwei IE-Switches/-Bridges gibt. Ein Spanning Tree verhindert, dass es zu einer Schleifenbildung im Netz kommt, indem er genau einen Pfad zulässt und die anderen (redundanten) Ports für den Datenverkehr deaktiviert. Bei einer Unterbrechung können die Daten über einen alternativen Pfad gesendet werden. Die Funktionalität des Spanning Tree-Verfahrens basiert auf dem Austausch von Konfigurations- und Topologieänderungs-Telegrammen.

Definition der Netztopologie durch Konfigurationstelegramme

Die Geräte tauschen zur Berechnung der Topologie untereinander Konfigurationstelegramme aus, sogenannte BPDUs (Bridge Protocol Data Units). Mit diesen Telegrammen wird die Root Bridge ausgewählt und die Netztopologie erstellt. Darüber hinaus bewirken BPDU-Telegramme den Statuswechsel der Root-Ports.

Die Root Bridge ist die Bridge, die das Spanning Tree-Verfahren für alle beteiligten Komponenten steuert.

Nachdem die Root Bridge festgelegt ist, bestimmt jedes Gerät einen Root-Port. Der Root-Port ist der Port mit den geringsten Pfadkosten zur Root Bridge.

Verhalten bei Veränderungen der Netztopologie

Wenn Teilnehmer zu einem Netz hinzukommen oder wegfallen, kann das Auswirkungen auf die optimale Wegewahl der Datenpakete haben. Um diese Änderungen zu berücksichtigen, versendet die Root Bridge in regelmäßigen Abständen Konfigurationsmeldungen. Der Zeitabstand zwischen zwei Konfigurationsmeldungen lässt sich mit dem Parameter "Hello Time" einstellen.

Aktualität der Konfigurationsinformation

Mit dem Parameter "Max Age" legen Sie das maximale Alter von Konfigurationsinformationen fest. Erhält eine Bridge Konfigurationsinformationen, die älter sind als in "Max Age" festgelegt, verwirft sie diese Meldung und veranlasst eine Neuberechnung der Wege.

Neue Konfigurationsinformationen werden von einer Bridge jedoch nicht sofort, sondern erst nach dem im Parameter "Forward Delay" festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.

RSTP, MSTP, CIST

Rapid Spanning Tree Protocol (RSTP)

Ein Nachteil des STP ist, dass sich das Netz bei einer Störung oder einem Geräteausfall rekonfigurieren muss: Die Geräte beginnen erst im Moment der Unterbrechung, neue Pfade auszuhandeln. Dieser Vorgang dauert bis zu 30 Sekunden. Aus diesem Grunde wurde STP zum "Rapid Spanning Tree Protocol" (RSTP, IEEE 802.1w) erweitert. Dies unterscheidet sich vom STP im Wesentlichen dadurch, dass die Geräte bereits zum Zeitpunkt des ungestörten Betriebs Informationen über Alternativrouten sammeln, die sie sich dann nicht erst beschaffen müssen, wenn eine Störung eingetreten ist. Damit lässt sich die Rekonfigurationszeit für ein RSTP-gesteuertes Netz auf wenige Sekunden reduzieren.
Das wird durch folgende Funktionen erreicht:

- Edge-Ports (Endteilnehmer-Port)
  Edge-Ports sind Ports, die mit einem Endgerät verbunden sind.
  Ein Port, der als Edge-Port definiert ist, wird direkt nach einem Verbindungsaufbau aktiviert. Wenn an einem Edge-Port eine Spanning Tree-BPDU empfangen wird, verliert der Port die Rolle als Edge-Port und nimmt wieder am (R)STP teil. Wird nach Ablauf einer Zeitspanne (3x Hello-Time) kein BPDU-Telegramm mehr empfangen, geht der Port wieder in den Edge-Port-Status über.
- Punkt-zu-Punkt (direkte Kommunikation zweier benachbarter Geräte)
  Durch die direkte Kopplung der Geräte kann eine Zustandsänderung (Umkonfiguration der Ports) ohne Verzögerungen durchgeführt werden.
- Alternativ-Port (Ersatz für den Root-Port)
  Es ist ein Ersatz für den Root-Port konfiguriert. Bei einem Verbindungsverlust zur Root-Bridge kann das Gerät deshalb ohne Verzögerung durch Neukonfiguration eine Verbindung über den Alternativ-Port aufbauen.

- Reaktion auf Ereignisse
  Ein Rapid Spanning Tree reagiert auf Ereignisse, beispielsweise einen Verbindungsabbruch, ohne Verzögerung. Es müssen also keine Zeitgeber wie beim Spanning Tree abgewartet werden.
- Zähler maximale Bridge-Sprünge
  Anzahl der Bridge-Sprünge, die ein Paket maximal ausführen darf, bevor es automatisch ungültig wird.

Prinzipiell werden also beim Rapid Spanning Tree für viele Parameter Alternativen vorkonfiguriert oder bestimmte Eigenschaften der Netzstruktur berücksichtigt, um die Rekonfigurationszeit zu verkürzen.

Multiple Spanning Tree Protocol (MSTP)

Das Multiple Spanning Tree Protocol (MSTP) ist eine Weiterentwicklung des Rapid Spanning Tree Protocols. Es bietet u. a. die Möglichkeit, mehrere RSTP-Instanzen innerhalb verschiedener VLANs oder VLAN-Gruppen zu betreiben und so z. B. Pfade, die das einfache Rapid Spanning Tree Protocol für den Datenverkehr global sperren würde, innerhalb einzelner VLANs verfügbar zu machen.

Common and Internal Spanning Tree (CIST)

CIST bezeichnet die intern vom Switch verwendete Instanz, die im Prinzip einer internen RSTP-Instanz gleicht.

HRP

HRP - High Speed Redundancy Protocol

HRP bezeichnet ein Redundanz-Verfahren für Netze in Ring-Topologie. Die Switches sind über Ringports miteinander verbunden. Einer der Switches wird zum Redundanzmanager (RM) konfiguriert. Die anderen Switches sind Redundanz-Clients. Mit Testtelegrammen prüft der Redundanzmanager den Ring auf Unterbrechungsfreiheit. Der Redundanzmanager sendet Testtelegramme über die Ringports und prüft deren Empfang am jeweils anderen Ringport. Die Redundanz-Clients leiten die Testtelegramme weiter.

Wenn die Testtelegramme des RM bei einer Unterbrechung des Rings nicht mehr am anderen Ringport ankommen, schaltet der RM seine beiden Ringports durch und informiert die Redundanz-Clients umgehend über den Wechsel. Die Rekonfigurationszeit nach Unterbrechung des Rings beträgt maximal 300 ms.

Standby-Redundanz

Standby-Redundanz ist ein Verfahren, bei dem Ringe, die jeder für sich durch High-Speed Redundancy gesichert sind, redundant gekoppelt werden. Im Ring wird ein Master-/Slave-Gerätepaar konfiguriert, das sich gegenseitig über seine Ringports überwacht. Der Datenverkehr wird im Fehlerfall von einer Ethernet-Verbindung (Standby-Port des Master bzw. Standby-Server) zu einer anderen Ethernet-Verbindung (Standby-Port des Slave) umgeleitet.

Voraussetzungen

HRP

- HRP wird in Ringtopologien mit bis zu 50 Geräten unterstützt. Eine Überschreitung der Geräteanzahl kann zum Ausfall des Datenverkehrs führen.
- Für HRP dürfen im Ring nur Geräte verwendet werden, die diese Funktion auch unterstützen.
- Geräte, die HRP nicht unterstützen, müssen über spezielle HRP-fähige Geräte an den jeweiligen Ring angebunden werden. Diese Verbindung ist bis zum Ring nicht redundant.
- Alle Geräte müssen über ihre Ringports miteinander verbunden sein. Dabei sind Multimodeverbindungen bis 3 km und Singlemodeverbindungen bis 26 km zwischen zwei IE-Switches möglich. Bei größeren Entfernungen kann es zu einer Verlängerung der angegebenen Rekonfigurationszeit kommen.
- Ein Gerät im Ring muss durch Auswahl der Einstellung "HRP-Manager" als Redundanzmanager konfiguriert werden. Bei allen übrigen Geräten im Ring muss entweder die Betriebsart "HRP-Client" oder die Betriebsart "Automatic Redundancy Detection" aktiviert werden.
- Die Standby-Ports müssen im Spanning Tree deaktiviert sein.
- Sie konfigurieren HRP über Web Based Management, Command Line Interface oder SNMP.

Standby-Redundanz

- Bei Standby-Koppelpartnern muss HRP fest eingestellt sein.
- Die Ports der Standby-Koppelpartner müssen im Spanning Tree deaktiviert sein.
- Sie konfigurieren die Standby-Redundanz über Web Based Management, Command Line Interface oder SNMP.

Standby

Allgemeines

SCALANCE X Switches unterstützen neben der Ringredundanz innerhalb eines Ringes auch die redundante Kopplung von Ringen oder offenen Netzsegmenten (Linien). Bei der redundanten Kopplung werden Ringe über zwei Ethernet-Verbindungen miteinander gekoppelt. Hierzu wird in einem Ring ein Master-/Slave-Gerätepaar konfiguriert, das sich gegenseitig überwacht und den Datenverkehr im Fehlerfall von der im Regelfall genutzten Master-Ethernet-Verbindung zur Ausweich-(Slave-)Ethernet-Verbindung umleitet.

Standby-Redundanz

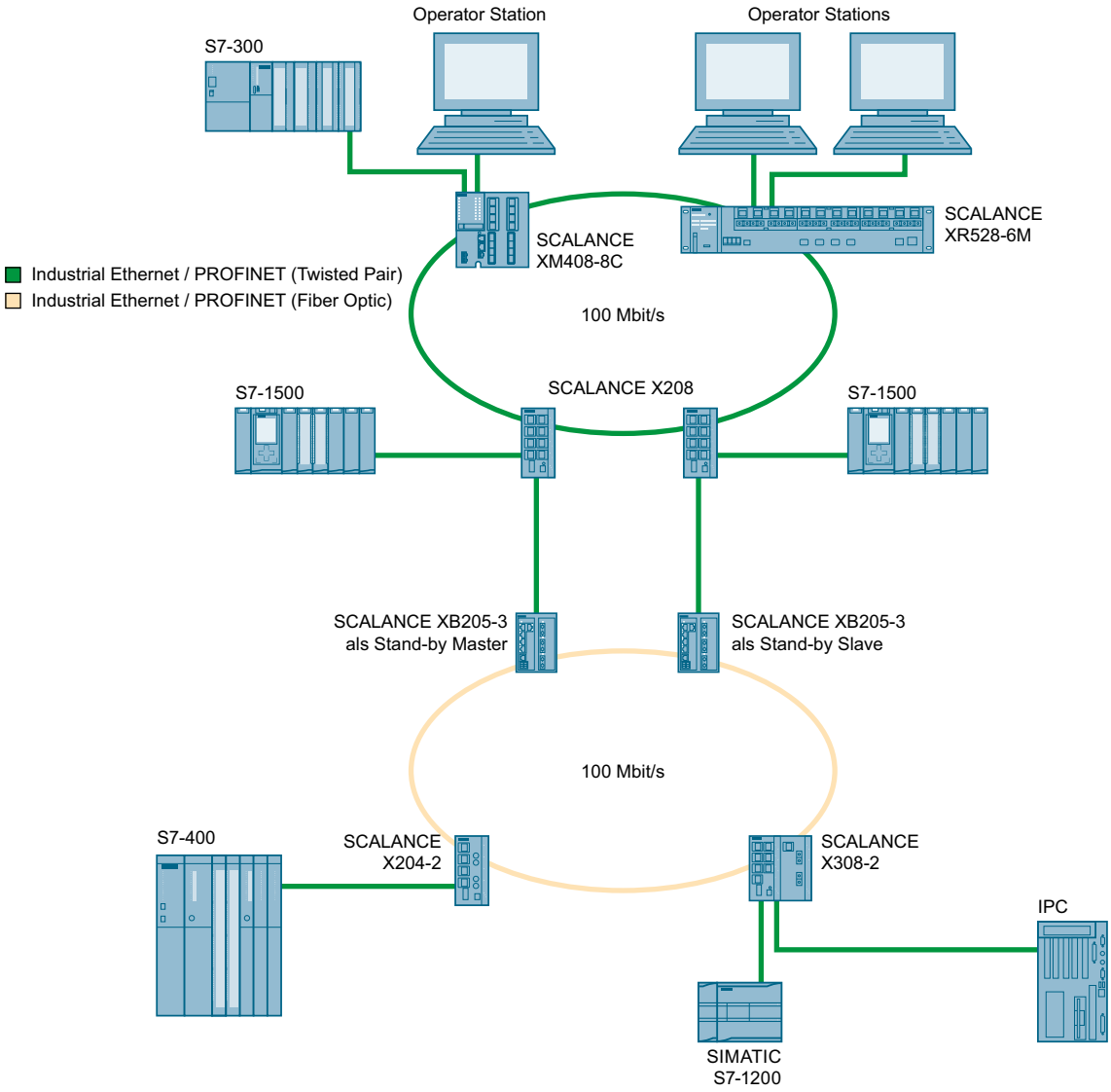

Bild 1-83 Beispiel einer redundanten Kopplung von zwei Ringen

Für eine redundante Kopplung, wie im Bild dargestellt, müssen zwei Geräte innerhalb eines Netzsegments als Standby-Redundanz-Switches projektiert werden. Netzsegmente sind hier Ringe mit einem Redundanzmanager. An die Stelle der Ringe können dabei auch Netzsegmente in Linien-Form treten.

Die beiden per Projektierung verbundenen Standby-Redundanz-Switches tauschen Datentelegramme miteinander aus und synchronisieren damit ihren Betriebsstatus (ein Gerät wird Master und das andere Slave). Im fehlerfreien Zustand ist nur beim Master die Koppelstrecke zum anderen Netzsegment aktiv. Fällt diese Koppelstrecke aus (z.B. infolge eines Link-Down oder eines Geräteausfalls), so aktiviert der Slave seine Koppelstrecke, solange der Fehler ansteht.

Kopplung mehrerer HRP-Netzsegmente

Wenn Sie mehrere HRP-Ringe oder Linien über Standby-Redundanz verbinden, müssen sich der Standby-Master und der Standby-Slave in einem geschlossenen Netzsegment befinden. Dieses Netzsegment darf keinesfalls offen, d. h. eine Linie, sein.

Standby-Master und -Slave in einem geschlossenen Netzsegment

Wenn die Verbindung zwischen Standby-Master und -Slave unterbrochen wird, können die beiden Geräte weiterhin über die redundante Strecke des HRP-Redundanzmanagers kommunizieren.

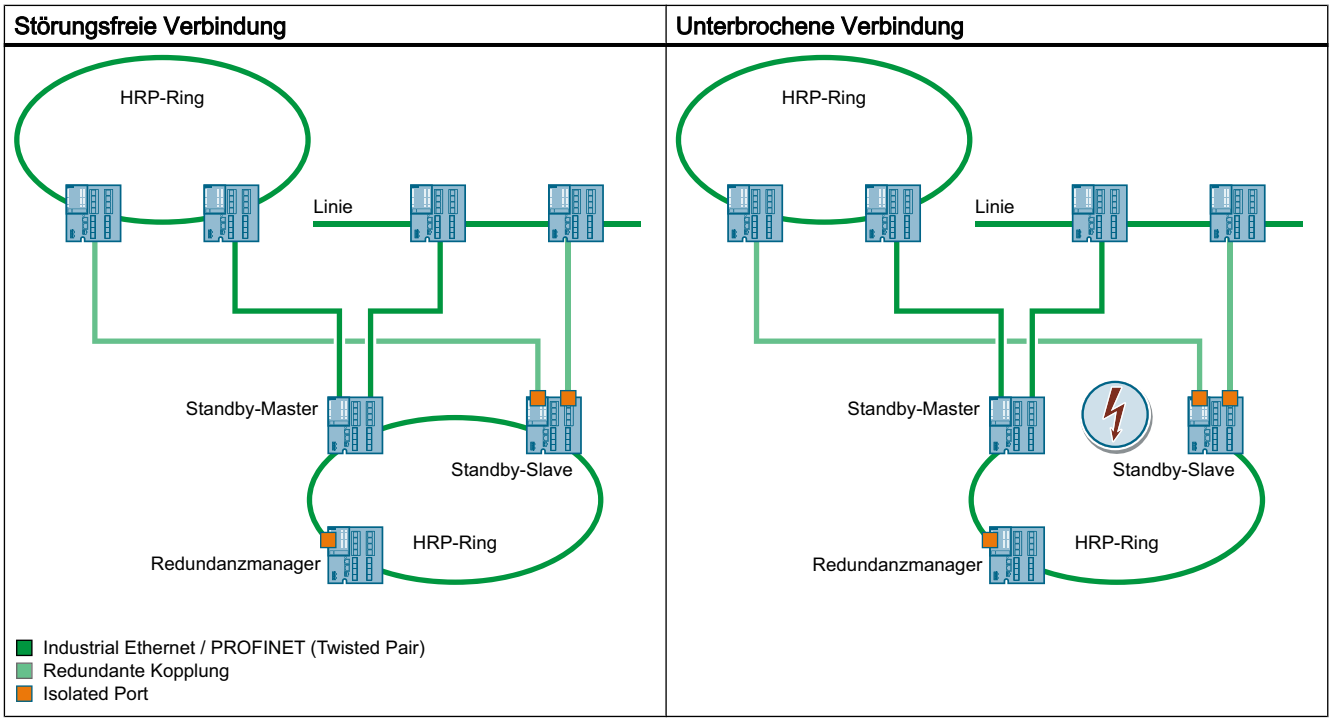

Standby-Master und -Slave in einem offenen Netzsegment

Wenn die Verbindung zwischen Standby-Master und -Slave unterbrochen wird, können die beiden Geräte nicht mehr kommunizieren. Dies führt zu einer Schleife über die gekoppelten Netzsegmente.

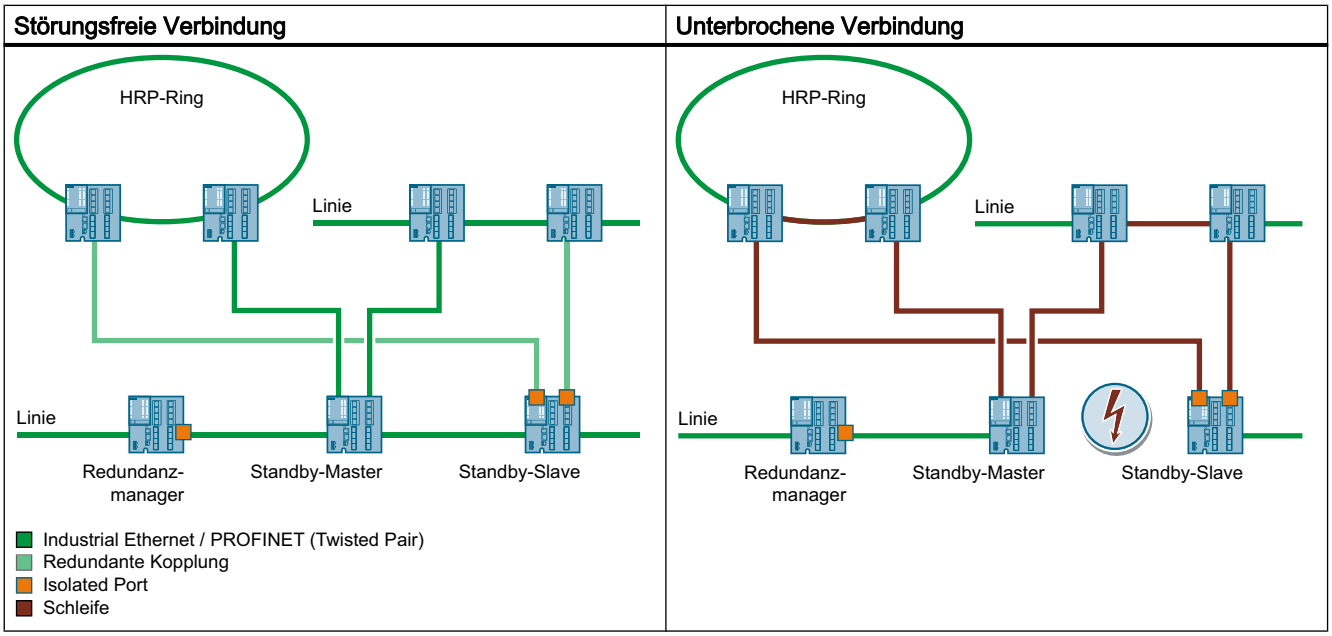

Link Check

Überwachung optischer Verbindungen im Ring

Bei optischen Übertragungsstrecken sind Störungen möglich, bei denen die optische Verbindung nicht vollständig unterbrochen wird, sondern nur sporadisch Telegramme verloren gehen. Solche Störungen können z. B. durch defekte optische Kabel, verschmutzte Steckverbindungen oder Gerätedefekte verursacht werden.

Der Redundanzmanager eines HRP- oder MRP-Rings mit optischen Verbindungen erkennt bei einer solchen Störung einen "Non-Recoverable Ring Error". Der Redundanzmanager kann die Störung nicht beheben, indem er den Ring schließt. Ein Schließen des Rings würde in diesem Fall zu kreisenden Telegrammen führen.

Mit der Funktion Link Check können Sie die Übertragungsqualität optischer Strecken innerhalb eines HRP- oder MRP-Rings überwachen, gestörte Übertragungsstrecken identifizieren und unter bestimmten Bedingungen abschalten. Wenn die gestörte Strecke abgeschaltet ist, kann der Redundanzmanager den Ring schließen und die Kommunikation wiederherstellen.

Funktionsweise von Link Check

Verhalten bei einer ungestörten Verbindung

Wenn Sie Link Check an zwei verbundenen Ring-Ports aktivieren, tauschen die beiden Verbindungspartner auf diesen Ports zyklisch Link Check-Testtelegramme aus. Die Telegramme, die der eine Verbindungspartner erhält, sendet er an den anderen zurück.

Wenn die Geräte ihre versendeten Telegramme vom Verbindungspartner zurückerhalten, ist die Verbindung für Link Check vorbereitet. Daraufhin erhöhen die Verbindungspartner die Sendefrequenz der Link Check-Testtelegramme und die eigentliche Verbindungsüberwachung ist aktiv.

Verhalten bei einer Störung

Bei aktiver Verbindungsüberwachung können Sie die Anzahl der gesendeten und empfangenen Link Check-Testtelegrammen auf der Seite "Information > Redundanz > Link Check" einsehen. Anhand dieser Statistik können Sie bereits kleinere Störungen erkennen, bei denen die Störung noch nicht zum Abschalten der Übertragungsstrecke durch Link Check führt.

Link Check erkennt eine Verbindung als fehlerhaft und schaltet diese ab, wenn innerhalb bestimmter Zeiträume zu viele Testtelegramme verloren gehen. Link Check verwendet dabei mehrere Zeitintervalle, um sowohl ein sprunghaftes Auftreten von Fehlern als auch eine kontinuierliche, niedrige Fehlerrate erkennen zu können.

Ein Port, der durch Link Check abgeschaltet wurde, muss zurückgesetzt werden, um wieder kommunizieren zu können. Hierzu haben Sie 2 Möglichkeiten:

- Ziehen Sie das Verbindungskabel und stecken Sie es wieder.
- Setzen Sie die Funktion auf beiden Verbindungspartnern über die Schaltfläche "Zurücksetzen" zurück. Dies muss bei beiden Geräten innerhalb von 30 s erfolgen.

Hinweis

Wenn Sie die Schaltfläche "Zurücksetzen" verwenden, kann es zu kurzfristiger Schleifenbildungen und damit zum Ausfall des Datenverkehrs kommen. Die Schleife wird automatisch wieder aufgelöst.

Wenn das in Ihrem Anwendungsfall nicht akzeptabel ist, setzen Sie Link Check zurück, indem Sie das Verbindungskabel ziehen und wieder stecken.

Nach dem Zurücksetzen von Link Check wird die Funktion auf dem Port erneut gestartet und die Statistik wird zurückgesetzt.

Projektierung über einen PROFINET IO-Controller

Wenn MRP über einen PROFINET IO-Controller projektiert wird, können Sie die Funktion Link Check per WBM oder CLI für die optischen Ringports der 1. MRP-Ring-Instanz zuschalten.

Wenn eine neue Projektierung übertragen wird, wird Link Check auf allen Ports automatisch deaktiviert, die nicht als Ringports der 1. MRP-Ring-Instanz konfiguriert wurden.

Hinweis

Ereignisse, die die Funktion Link Check betreffen, werden nur indirekt durch PROFINET IO gemeldet. Falls die MRP-Diagnosealarme aktiviert sind und ein Ringport durch LinkCheck deaktiviert wird, wird von Profinet IO eine Fehlermeldung generiert, dass diese Verbindung nicht mehr besteht.

MRP

MRP - Media Redundancy Protocol

Das Verfahren "MRP" arbeitet konform zum Media Redundancy Protocol (MRP), das in folgender Norm spezifiziert ist:

IEC 62439-2 Ausgabe 1.0 (2010-02) Industrial communication networks - High availability automation networks Part 2: Media Redundancy Protocol (MRP)

Die Rekonfigurationszeit nach Unterbrechung des Rings beträgt maximal 200 ms.

Topologie

Die folgende Abbildung zeigt eine mögliche Topologie für Geräte in einem Ring mit MRP.

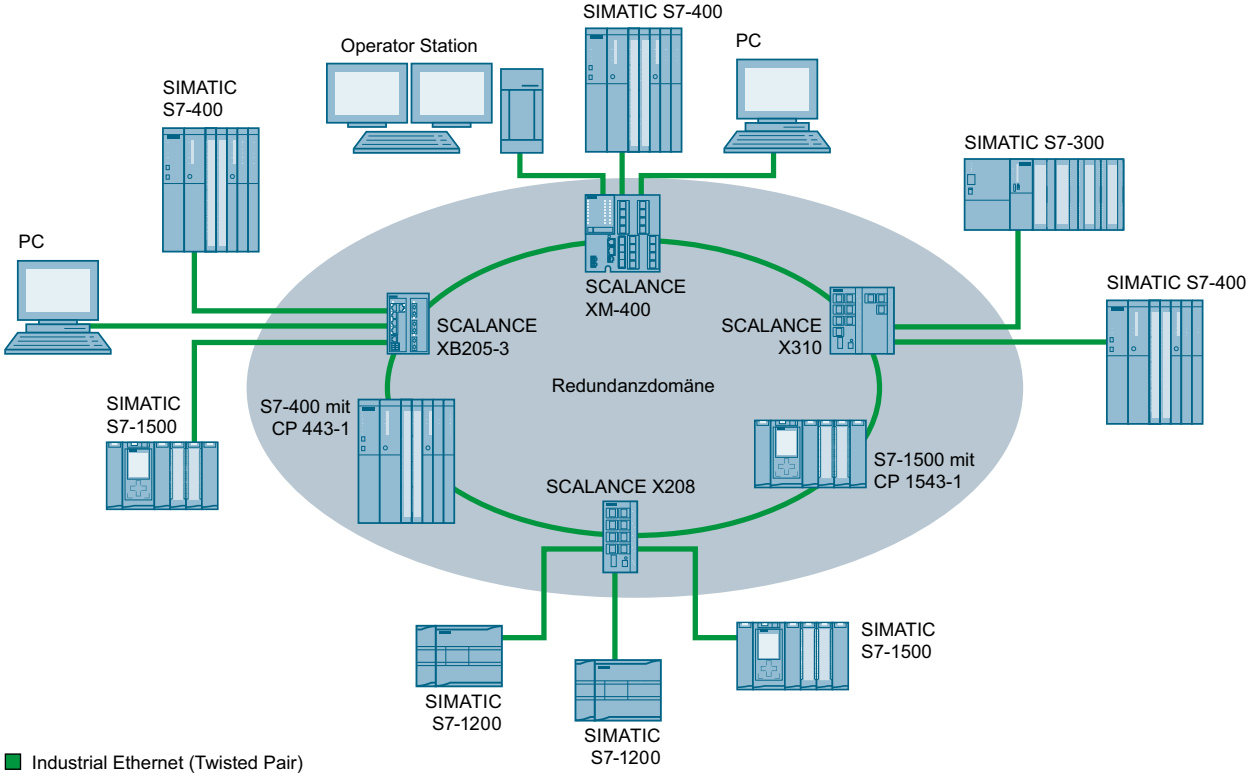

Bild 1-84 Beispiel einer Ringtopologie mit dem Medienredundanzverfahren MRP

Für die Ringtopologie mit Medienredundanz nach dem Verfahren MRP gelten folgende Regeln:

- Alle innerhalb der Ringtopologie verbundenen Geräte sind Mitglieder der gleichen Redundanz-Domäne.
- Ein Gerät im Ring ist Redundanzmanager.
- Alle anderen Geräte im Ring sind Redundanz-Clients.

Nicht MRP-fähige Geräte können über einen Switch SCALANCE X oder einen PC mit MRP-fähigem CP an den Ring angebunden werden.

Voraussetzungen

Voraussetzungen für den störungsfreien Betrieb mit dem Medienredundanzverfahren MRP sind:

- MRP wird in Ringtopologien mit bis zu 50 Geräten unterstützt.
  Eine Überschreitung der Geräteanzahl kann zum Ausfall des Datenverkehrs führen.
- Der Ring, in dem Sie MRP einsetzen wollen, darf nur aus Geräten bestehen, die diese Funktion unterstützen.
  Dies sind beispielsweise einige der Industrial Ethernet Switches SCALANCE X, einige der Kommunikationsprozessoren (CPs) für die SIMATIC S7 und PG/PC oder Nicht-Siemens-Geräte, die diese Funktion unterstützen.
- Alle Geräte müssen über ihre Ringports miteinander verbunden sein.
  Dabei sind Multimodeverbindungen bis 3 km und Singlemodeverbindungen bis 26 km zwischen zwei IE-Switches SCALANCE X möglich. Bei größeren Entfernungen kann es zu einer Verlängerung der angegebenen Rekonfigurationszeit kommen.
- Bei allen Geräten im Ring muss "MRP" aktiviert sein.
- Die Verbindungseinstellungen (Übertragungsmedium / Duplex) müssen für alle Ringports auf Vollduplex und mindestens 100 Mbit/s eingestellt sein. Andernfalls kann es zum Ausfall des Datenverkehrs kommen.
  - STEP 7: Setzen Sie im Eigenschaftendialog aller am Ring beteiligten Ports die Verbindung im Register "Optionen" auf "Automatische Einstellung".
  - WBM: Bei Projektierung über Web Based Management werden die Ringports automatisch auf Autonegotiation eingestellt.

Hinweis

Geräteanzahl

Außer in PROFINET IO-Anlagen, wurden Topologien mit bis zu 100 IE-Switches SCALANCE X-200 und SCALANCE X-300 erfolgreich getestet.

Projektierung im WBM

Rolle

Die Auswahl der Rolle ist von den folgenden Einsatzfällen abhängig:

- Sie wollen MRP in einer Ringtopologie nur mit Siemens-Geräten einsetzen:
  - Wählen Sie bei mindestens einem Gerät im Ring "Automatic Redundancy Detection" bzw. "MRP Auto-Manager" aus.
  - Wählen Sie bei allen anderen Geräten im Ring "MRP-Client" oder "Automatic Redundancy Detection" aus.
- Sie wollen MRP in einer Ringtopologie einsetzen, die auch Nicht-Siemens-Geräte enthält:
  - Wählen Sie bei genau einem Gerät im Ring die Rolle "MRP Auto-Manager" aus.
  - Wählen Sie bei allen anderen Geräten der Ringtopologie die Rolle "MRP-Client" aus.

  Hinweis

  Die Verwendung von "Automatic Redundancy Detection" ist beim Einsatz von Nicht-Siemens-Geräten nicht möglich.

- Sie projektieren die Geräte in einer MRP-Ringtopologie teilweise über WBM und teilweise über Step 7:
  - Wählen Sie bei den Geräten, die Sie über WBM projektieren, für alle Geräte "MRP-Client" aus.
  - Wählen Sie bei den Geräten, die Sie über Step 7 projektieren, für genau ein Gerät "Manager" oder "Manager (Auto)" und für alle anderen Geräte "MRP-Client" aus.

  Hinweis

  Wenn einem Gerät über Step 7 die Rolle "Manager" zugeordnet wird, muss allen anderen Geräten im Ring die Rolle "MRP-Client" zugeordnet werden. Wenn es in einem Ring ein Gerät mit der Rolle "Manager" und ein Gerät mit der Rolle "Manager (Auto)"/"MRP Auto-Manager" gibt, kann es zu kreisenden Frames und damit zum Ausfall des Netzwerks kommen.

Projektierung

Im WBM projektieren Sie MRP auf folgenden Seiten:

- Konfiguration (Seite 2265)
- Ring (Seite 2295)

Projektierung in STEP 7

Projektierung in STEP 7

Wählen Sie zur Projektierung in STEP 7 die Parametergruppe "Medienredundanz" an der PROFINET-Schnittstelle.

Stellen Sie folgende Parameter zur MRP-Konfiguration des Geräts ein:

- Domäne
- Rolle
- Ringport
- Diagnosealarme

Diese Einstellungen werden nachfolgend beschrieben.

Hinweis

Gültige MRP-Projektierung

Stellen Sie bei der MRP-Projektierung in STEP 7 sicher, dass alle Geräte im Ring eine gültige MRP-Projektierung besitzen, bevor Sie den Ring zusammenschließen. Andernfalls kann es zu kreisenden Frames und damit zum Ausfall des Netzwerks kommen.

Ein Gerät im Ring müssen Sie als "Redundanzmanager" konfigurieren, alle anderen Geräte im Ring als "Client".

Hinweis

Werkseinstellungen beachten

Bei folgenden fabrikneuen und auf Werkseinstellungen gesetzten IE-Switches ist MRP deaktiviert und Spanning Tree aktiviert:

- SCALANCE XB-200 (EtherNet/IP-Varianten)
- SCALANCE XC-200 (EtherNet/IP-Varianten)
- SCALANCE XP-200 (EtherNet/IP-Varianten)
- SCALANCE XR-300WG
- SCALANCE XM-400
- SCALANCE XR-500

Um eine PROFINET-Konfiguration mit MRP in eines der genannten Geräte zu laden, deaktivieren Sie auf dem Gerät zunächst Spanning Tree.

Hinweis

Umkonfiguration nur bei geöffnetem Ring

Öffnen Sie zunächst den Ring, bevor Sie

- die MRP-Rolle ändern oder
- Ringports umkonfigurieren.

Hinweis

Neustart und Wiederanlauf

Die MRP-Einstellungen sind auch nach einem Neustart des Geräts oder nach einem Spannungsausfall und Wiederanlauf wirksam, sofern der Spannungsausfall nicht innerhalb von 90 Sekunden nach der Konfigurationsänderung stattfindet.

Hinweis

Priorisierter Hochlauf

Wenn Sie MRP in einem Ring projektieren, dann können Sie in den beteiligten Geräten in PROFINET-Applikationen die Funktion "Priorisierter Hochlauf" nicht nutzen.

Wenn Sie die Funktion "Priorisierter Hochlauf" nutzen wollen, dann müssen Sie MRP in der Projektierung deaktivieren.

Setzen Sie in der STEP 7-Projektierung des betreffenden Geräts die Rolle auf "Nicht Teilnehmer des Rings".

Domäne

Einfache MRP-Ringe

Wenn Sie einen einzelnen MRP-Ring konfigurieren wollen, belassen Sie in der Klappliste "Domain" den werkseitig vorbelegten Eintrag "mrpdomain-1".

Alle Geräte, die in einem Ring mit MRP projektiert werden, müssen der gleichen Redundanz-Domäne angehören. Ein Gerät kann nicht mehreren Redundanz-Domänen in einem Einfachring angehören.

Mehrere MRP-Ringe

Wenn Sie mehrere einzelne MRP-Ringe konfigurieren, werden die Ringteilnehmer über den Parameter "Domain" den einzelnen Ringen zugeordnet. Stellen Sie für alle Geräte innerhalb eines Rings die gleiche Domäne ein. Stellen Sie für die unterschiedlichen Ringe unterschiedliche Domänen ein. Geräte, die nicht zum gleichen Ring gehören, müssen unterschiedliche Domänen haben.

Wenn Sie MRP-Mehrfachringe projektieren möchten, wählen sie als zentralen Redundanzmanager für bis zu vier Ringe ein mehrfachringfähiges Gerät aus. Legen Sie für alle Ring-Instanzen unterschiedliche Domänen fest und ordnen Sie diese den zugehörigen

Ringports des Redundanzmanagers zu. Projektieren sie die übrigen Geräte als Clients. Dabei muss bei allen Geräten innerhalb eines Rings die gleiche Domäne eingestellt werden.

Hinweis

Geeignete Geräte für MRP-Mehrfachringe

Als Redundanzmanager, der mehrere Ringe verbindet, können Sie alle Geräte der folgenden Produktlinien einsetzen:

- SCALANCE X-300 ab Firmware-Version V4.0
- SCALANCE X-400 SCALANCE
  - X408-2 ab Firmware-Version V4.0
  - X414-3E ab Firmware-Version V3.10

Hinweis

Geeignete Geräte für MRP-Interconnection

Als Medienredundanz Interconnection Manager und Medienredundanz Interconnection Client können Sie alle Geräte der folgenden Produktlinien einsetzen:

- SCALANCE XM-400 ab Firmware-Version V6.2
- SCALANCE XR-500 ab Firmware-Version V6.2

Rolle

Hinweis

Umkonfiguration nur bei geöffnetem Ring!

Die Auswahl der Rolle ist von den folgenden Einsatzfällen abhängig.

- Sie wollen MRP in einer Topologie mit **einem Ring** nur mit Siemens-Geräten einsetzen und keine Diagnosealarme überwachen:
  Ordnen Sie alle Geräte der Domäne "mrpdomain-1" und der Rolle "Manager (Auto)" zu. Das Gerät, welches im Betrieb tatsächlich die Rolle des Redundanzmanagers übernimmt, wird unter Siemens-Geräten automatisch ausgehandelt.
- Sie wollen MRP in einer Topologie mit **mehreren Ringen** nur mit Siemens-Geräten einsetzen und keine Diagnosealarme überwachen:
  - Ordnen Sie allen Instanzen des Geräts, das die Ringe verbindet, die Rolle "Manager" zu.
  - Wählen Sie bei allen anderen Geräten der Ringtopologie die Rolle "Client".

- Sie wollen MRP in einer Ringtopologie einsetzen, die auch Nicht-Siemens-Geräte enthält, oder Sie wollen Diagnosealarme zum MRP-Zustand von einem Gerät erhalten (siehe "Diagnosealarme"):
  - Ordnen Sie genau einem Gerät im Ring die Rolle "Manager (Auto)" zu.
  - Wählen Sie bei allen anderen Geräten der Ringtopologie die Rolle "Client".
- Sie wollen MRP deaktivieren:
  Wählen Sie die Option "Nicht Teilnehmer des Rings", wenn Sie das Gerät nicht innerhalb einer Ringtopologie mit MRP betreiben wollen.

Hinweis

Rolle beim Rücksetzen auf Werkseinstellungen

Bei fabrikneuen und auf Werkseinstellungen gesetzten Siemens-Geräten ist folgende MRP-Rolle eingestellt:

- "Manager (Auto)"
  - CPs
- "Automatic Redundancy Detection"
  - SCALANCE X-200
  - SCALANCE XB-200 (PROFINET-Varianten)
  - SCALANCE XC-200 (PROFINET-Varianten)
  - SCALANCE XF-200BA
  - SCALANCE XP-200 (PROFINET-Varianten)
  - SCALANCE X-300
  - SCALANCE X-400

Wenn Sie im Ring ein Nicht-Siemens-Gerät als Redundanzmanager betreiben, kann dies zum Ausfall des Datenverkehrs führen.

Bei folgenden fabrikneuen und auf Werkseinstellungen gesetzten IE-Switches ist MRP deaktiviert und Spanning Tree aktiviert:

- SCALANCE XB-200 (EtherNet/IP-Varianten)
- SCALANCE XC-200 (EtherNet/IP-Varianten)
- SCALANCE XP-200 (EtherNet/IP-Varianten)
- SCALANCE XR-300WG
- SCALANCE XM-400
- SCALANCE XR-500

Ringport 1 / Ringport 2

Wählen Sie hier jeweils den Port aus, den Sie als Ringport 1 bzw. als Ringport 2 projektieren möchten.

Bei Geräten mit mehr als 8 Ports sind gegebenenfalls nicht alle Ports als Ringport auswählbar.

Die Klappliste zeigt für jeden Gerätetyp die Auswahl der möglichen Ports an. Wenn die Ports werkseitig festgelegt sind, dann sind die Felder gegraut.

ACHTUNG
Ringports beim Rücksetzen auf Werkseinstellungen Mit dem Rücksetzen auf Werkseinstellungen werden auch die Ringport-Einstellungen zurückgesetzt. Wenn vor dem Rücksetzen andere Ports als Ringports verwendet wurden, kann bei entsprechendem Anschluss ein zuvor korrekt konfiguriertes Gerät kreisende Frames und damit den Ausfall des Datenverkehrs verursachen.

Hinweis

Umkonfiguration nur bei geöffnetem Ring

Öffnen Sie zunächst den Ring, bevor Sie die Ringports eines Ring-Managers umkonfigurieren.

Diagnosealarme

Aktivieren Sie die Option "Diagnose Alarme", wenn Diagnosealarme zum MRP-Zustand in der lokalen CPU ausgegeben werden sollen.

Folgende Diagnosealarme können gebildet werden:

- Verdrahtungs- bzw. Port-Fehler
  Bei folgenden Fehlern an den Ringports werden Diagnosealarme generiert:
  - Verbindungsabbruch an einem Ringport
  - Ein Nachbar des Ringports unterstützt nicht MRP.
  - Ein Ringport ist mit einem Nicht-Ringport verbunden.
  - Ein Ringport ist mit dem Ringport einer anderen MRP-Domäne verbunden.
- Statuswechsel Aktiv/Passiv (nur Redundanzmanager)
  Wenn sich in einem Ring der Status ändert (Aktiv/Passiv), wird ein Diagnosealarm generiert.

Parametrierung der Redundanz nicht durch STEP 7 vorgegeben (Alternative Redundanz)

Diese Option betrifft alle SCALANCE X-Switches. Wählen Sie diese Option, wenn die Eigenschaften zur Medienredundanz durch alternative Mechanismen wie WBM, CLI oder SNMP parametriert werden sollen.

Wenn Sie diese Option aktivieren, bleiben bestehende Redundanzeinstellungen erhalten und werden nicht überschrieben. Die Parameter im Feld "MRP-Konfiguration" werden daraufhin zurückgesetzt und gegraut dargestellt. Die Einträge sind dann ohne Bedeutung.

MRP-Interconnection

Das Verfahren MRP-Interconnection ist eine Erweiterung von MRP und ermöglicht die redundante Kopplung von zwei oder mehr MRP-Ringen in nicht echtzeitfähigen Netzwerken. MRP-Interconnection ist - wie MRP - in der Norm IEC 62439-2 spezifiziert. Mit MRP-Interconnection ist eine sehr schnelle Rekonfiguration möglich, die Rekonfigurationszeit beträgt typischerweise weniger als 200 Millisekunden.

Topologie

In der folgenden Grafik ist die redundante Kopplung von zwei MRP-Ringen dargestellt. MRP-Interconnection unterstützt darüber hinaus auch noch weitere Topologien, die allerdings von der aktuellen Firmware noch nicht unterstützt werden.

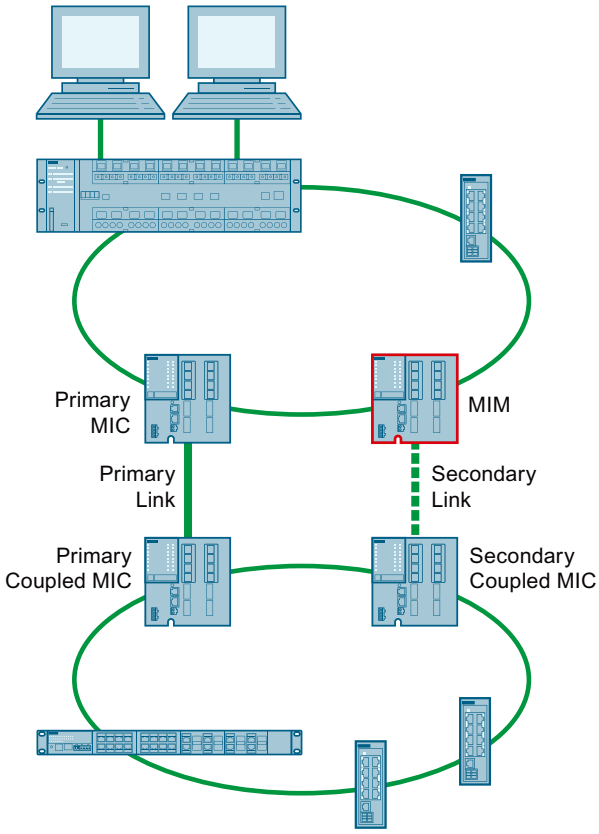

Bild 1-85 Redundante Verbindung von zwei MRP-Ringen mit MRP-Interconnection

Funktionsprinzip

Voraussetzung für MRP-Interconnection ist, dass MRP in allen beteiligten Ringen eingesetzt wird. Für die beiden MRP-Interconnection-Verbindungen sind vier Geräte erforderlich:

- Ein Medienredundanz Interconnection Manager (MIM, in der Grafik mit roter Kontur dargestellt)
- Drei Medienredundanz Interconnection Clients (Primary MIC, Primary Coupled MIC und Secondary Coupled MIC)

Da jedes Gerät Teil eines MRP-Rings ist, übernimmt jedes Gerät zusätzlich eine der für MRP definierten Rollen, also MRC oder MRM.

Abhängig vom Verbindungsstatus der Interconnection-Ports senden der MIC Statusmeldungen (Link-Up oder Link-Down) an den MIM. Interconnection-Ports sind Ports, zwischen denen die primäre oder die sekundäre Verbindung besteht. Dadurch ist der MIM immer über den Verbindungsstatus zwischen dem Primary MIC und dem Primary Coupled MIC ("Primäre Verbindung") sowie seiner eigenen Verbindung zum Secondary Coupled MIC ("Sekundäre Verbindung") informiert. Im regulären Betrieb erfolgt der Datenaustausch zwischen den beiden Ringen über die primäre Verbindung und der MIM blockiert seinen Interconnection-Port. Wird dem MIM ein Link-Down der primären Verbindung signalisiert, schaltet er seinen Interconnection-Port in den Status "Forwarding" und der Datenaustausch zwischen den beiden Ringen erfolgt über die sekundäre Verbindung zwischen MIM und Secondary Coupled MIC.

Hinweis

Geeignete Geräte für MRP-Interconnection

Der Interconnection Manager, der Interconnection Client sowie alle anderen Geräte in den Ringen müssen MRP-Interconnection unterstützen. Das ist bei folgenden Geräten gegeben:

- SCALANCE XM-400 ab Firmware-Version V6.2
- SCALANCE XR-500 ab Firmware-Version V6.2

Ergänzende Dokumentation

Ergänzende Beschreibungen zur Projektierung von MRP-Interconnection finden Sie im Projektierungshandbuch WBM von SCALANCE XM-400/SCALANCE XR-500.

Sie finden die ergänzenden Dokumentationen hier:

- Auf dem Datenträger, der manchen Produkten beiliegt:
  - Produkt-CD / Produkt-DVD
  - SIMATIC NET Manual Collection
- Auf den Internetseiten des Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/ps/15313/man)

Parallel Redundancy Protocol

Parallel Redundancy Protocol

Das "Parallel Redundancy Protocol" (PRP) ist ein Redundanzprotokoll für Ethernet-Netzwerke. Es ist im Teil 3 des IEC 62439 Standards definiert. Dieses Redundanzverfahren ermöglicht bei Netzwerkunterbrechungen die Datenkommunikation ohne Unterbrechung/Re-konfigurationszeit aufrechtzuerhalten.

Das PRP-Verfahren wird z. B. von den Geräten der Produktlinie SCALANCE X-200RNA unterstützt.

Überlange Frames

Beim Versenden von PRP-Frames erweitert der IE-Switch das Frame um einen PRP-Trailer. Bei Frames mit maximaler Länge entsteht durch den angehängten PRP-Trailer ein überlanges Frame, das die zulässige Framelänge überschreitet (nach IEEE 802.3-Standard).

Um Datenverluste bei überlangen Frames auszuschließen, müssen alle Netzkomponenten, die sich in einem PRP-Netzwerk befinden, eine Framelänge von mindestens 1528 Bytes unterstützen.

Die Geräte, die in diesem Handbuch beschrieben sind, können in PRP-Netzwerken eingesetzt werden, siehe auch Kapitel "Mengengerüste (Seite 2068)".

Routing-Funktion

Einleitung

Der Begriff Routing beschreibt das Festlegen von Wegen (Routen) für die Kommunikation zwischen verschiedenen Netzwerken, d. h. wie kommt ein Datenpaket aus Subnetz A zu Subnetz B.

SCALANCE X unterstützt folgende Routingfunktionen:

- Statisches Routing
  Beim statischen Routing werden die Routen in der Routing-Tabelle manuell eingetragen.
- Router Redundanz
  Mit standardisiertem VRRP (Virtual Router Redundancy Protocol) wird die Verfügbarkeit wichtiger Gateways durch redundante Router gesteigert.
  - VRRPv2 (IPv4)
  - VRRPv3 (IPv4 / IPv6)
- Dynamisches Routing
  Die Einträge in der Routing-Tabelle sind dynamisch und werden fortlaufend aktualisiert. Die Einträge werden mithilfe eines der folgenden dynamischen Routing-Protokolle erstellt:
  - OSPFv2 (IPv4)
  - OSPFv3 (IPv6)
  - RIPv2 (IPv4)
  - RIPng (IPv6)
  - MSDP
  - PIM Sparse Mode
  - BiDir PIM

VRRP

VRRPv2

Router-Redundanz mit VRRP

Mit dem Virtual Router Redundancy Protocol (VRRP) kann dem Ausfall eines Routers in einem Netzwerk begegnet werden.

VRRP kann nur bei virtuellen IP-Schnittstellen (VLAN-Schnittstellen) eingesetzt werden, nicht bei Router-Ports.

Mehrere VRRP-Router in einem Netzsegment werden zu einer logischen Gruppe zusammengefasst, die einen virtuellen Router (VR) darstellt. Definiert wird die Gruppe über die virtuelle ID (VRID). In der Gruppe muss die VRID gleich sein. Für andere Gruppen kann die VRID nicht mehr verwendet werden.

Dem virtuellen Router wird eine virtuelle IP-Adresse und eine virtuelle MAC-Adresse zugeordnet. Einer der VRRP-Router innerhalb der Gruppe wird als Master-Router festgelegt. Der Master-Router hat die Priorität 255. Die anderen VRRP-Router sind Backup-Router. Der Master-Router weist seiner Netzwerk-Schnittstelle die virtuelle IP-Adresse und die virtuelle MAC-Adresse zu. Der Master-Router sendet in bestimmten Intervallen VRRP-Pakete (Advertisements) an die Backup-Router. Mit den VRRP-Paketen signalisiert der Master-Router, dass er noch funktionsfähig ist. Der Master-Router beantwortet auch die ARP-Anfragen.

Wenn der virtuelle Master-Router ausfällt, übernimmt ein Backup-Router die Aufgabe des Master-Routers. Der Backup-Router mit der höchsten Priorität wird zum Master-Router. Wenn die Priorität der Backup-Router gleich ist, entscheidet die größere MAC-Adresse. Der Backup-Router wird zum neuen virtuellen Master-Router.

Der neue virtuelle Master-Router übernimmt die virtuelle MAC- und IP-Adresse. Dadurch müssen keine Routing-Tabellen oder ARP-Tabellen aktualisiert werden. Die Folgen eines Geräteausfalls werden dadurch minimiert.

VRRP konfigurieren Sie unter "Layer 3 (IPv4)> VRRP".

VRRP3

Die Version 3 von VRRP basiert auf der Version 2.

Hinweis

- Aktivieren Sie Routing, um VRRPv3 zu nutzen.
- VRRPv3 können Sie nur in Verbindung mit VLAN-Schnittstellen verwenden. Router-Ports werden nicht unterstützt.
- Der gleichzeitige Betrieb von VRRP und VRRPv3 ist nicht möglich.
- VRRPv3 unterstützt IPv4 und IPv6. Beide können mit VRRP3 gleichzeitig konfiguriert und betrieben werden.

VRRPv3 konfigurieren Sie unter:

IPv4: Layer 3 (IPv4)> VRRPv3

IPv6: Layer 3 (IPv6)> VRRPv3

OSPF

OSPFv2

Dynamisches Routing mit OSPFv2

OSPF (Open Shortest Path First) ist ein kostenbasiertes Routingprotokoll. Zum Berechnen der kürzesten und kostengünstigsten Route wird der Short Path First-Algorithmus nach Dijkstra verwendet. Entwickelt wurde OSPF von der IETF (Internet Engineering Task Force).

Sie konfigurieren OSPFv2 unter "Layer 3 (IPv4) > OSPFv2".

OSPFv2 unterteilt ein autonomes System (AS) in verschiedene Bereiche (Areas).

Bereiche in OSPF

Folgende Bereiche gibt es:

- Backbone
  Der Backbone-Bereich ist der Bereich 0.0.0.0. Mit diesem Bereich sind alle anderen Bereiche verbunden. Der Backbone-Bereich ist entweder direkt oder über virtuelle Verbindungen mit anderen Bereichen verbunden.
  Im Backbone-Bereich sind alle Routing-Informationen vorhanden. Somit ist der Backbone-Bereich für die Weiterleitung der Information zwischen verschiedenen Bereichen verantwortlich.
- Stub Area
  Dieser Bereich enthält die Routen innerhalb seines Bereichs, innerhalb des autonomen Systems und die Standardroute aus dem autonomen System heraus. Die Ziele außerhalb dieses autonomen Systems werden der Standardroute zugeordnet.
- Totally Stubby Area
  Dieser Bereich kennt nur die Routen innerhalb seines Bereichs und die Standardroute aus dem Bereich heraus.
- Not So Stubby Area (NSSA)
  Dieser Bereich kann Pakete von anderen autonomen Systemen in die Bereiche des eigenen autonomen Systems weiterleiten (redistribute). Die Pakete werden vom NSSA-Router weiterverteilt.

Router des OSPF

OSPF unterscheidet folgende Router-Typen:

- Interner Router (IR)
  Alle OSPF-Schnittstellen des Routers sind demselben Bereich zugeordnet.
- Area Border Router (ABR)
  Die OSPF-Schnittstellen des Routers sind verschiedenen Bereichen zugeordnet. Eine OSPF-Schnittstelle ist dem Backbone-Bereich zugeordnet. Wenn möglich werden Routen zusammengefasst.

- Backbone Router (BR)
  Mindestens eine der OSPF-Schnittstellen ist dem Backbone-Bereich zugeordnet.
- Autonomous System Border Router (ASBR)
  Eine Schnittstelle des Routers ist mit einem anderen AS verbunden, z. B. einem AS, dass das Routing-Protokoll RIP nutzt.

Virtuelle Verbindung

Jeder Bereich muss mit dem Backbone-Bereich verbunden sein. In einigen Fällen ist eine direkte physikalische Verbindung nicht möglich. In diesem Fall muss ein Router des betreffenden Bereichs über eine virtuelle Verbindung mit einem Backbone-Router verbunden werden.

LSA-Typen

Innerhalb des autonomen Systems werden Pakete ausgetauscht, die Informationen zu den Verbindungen eines Routers und die Verbindungsstatusmeldung enthalten. Die Pakete werden auch als LSAs (Link State Advertisements) bezeichnet. Die LSAs werden immer vom Router zum Nachbar-Router gesendet.

Wenn es Änderungen im Netzwerk gibt, werden LSAs an alle Router im Netzwerk versendet. Die Information ist abhängig vom LSA-Typ.

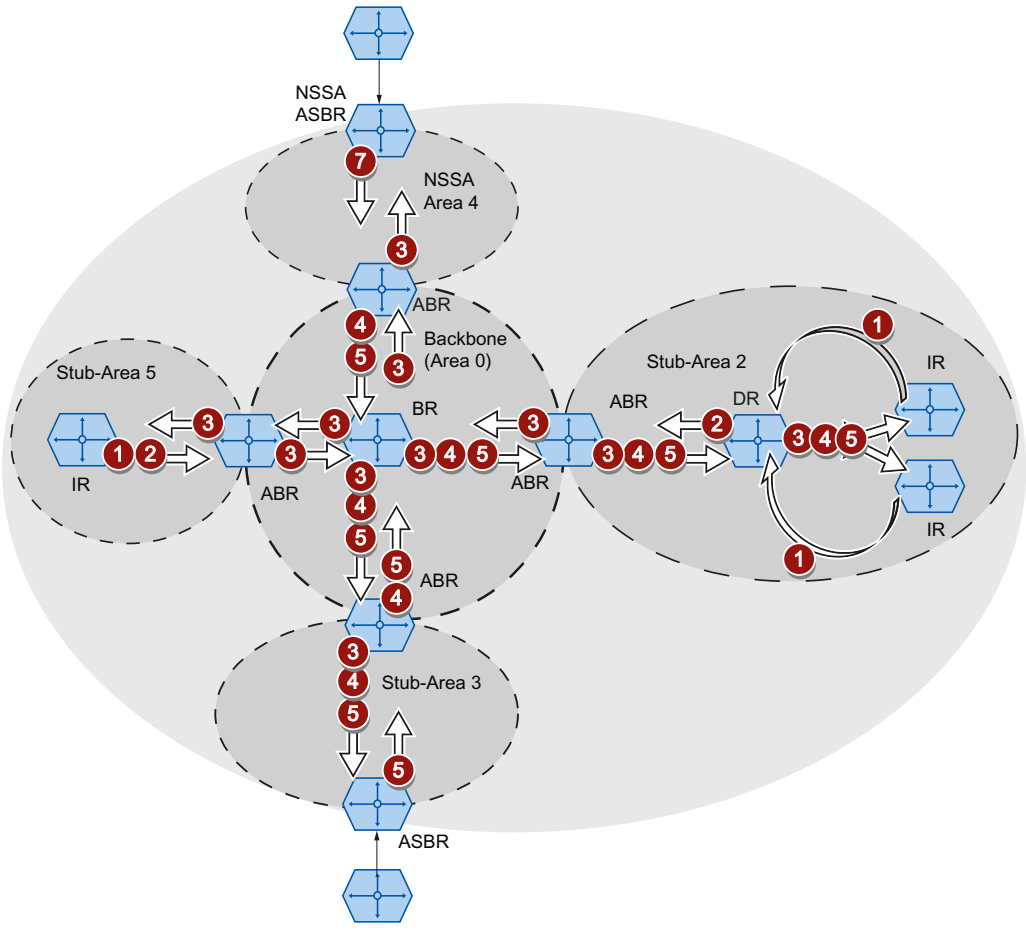

1. **Router LSA (LSA-Typ 1)**
   Der LSA-Typ 1 wird nur innerhalb eines Bereichs gesendet. Für jede aktive Verbindung des Routers, die zu dem betrachteten Bereich gehört, wird ein LSA-Typ 1 erzeugt. Der LSA-Typ 1 enthält Informationen über den Zustand und die Kosten der Verbindung, z. B. IP-Adresse, Netzmaske, Netzwerktyp
2. **Network LSA (LSA-Typ 2)**
   Der LSA-Typ 2 wird nur innerhalb eines Bereichs gesendet. Für jedes Netzwerk, das zu dem betrachten Bereich gehört, erzeugt der Router ein LSA-Typ 2. Sind in einem Netzwerk mehrere Router zusammengeschaltet, wird der LSA-Typ 2 vom designierten Router (DR) versendet. Der LSA-Typ 2 enthält u. a. die Netzadresse, die Netzmaske und eine Liste mit Routern, die mit dem Netzwerk verbunden sind

3 **Summary LSA (LSA-Typ 3 / LSA-Typ 4)**
Die Summary LSA wird vom Area Border Router erzeugt und in den Bereich gesendet.

4 Die Summary LSA enthält Informationen über Routen außerhalb des Bereichs, aber innerhalb des AS. Wenn möglich, werden die Routen zusammengefasst.

- Summary LSA (LSA-Typ 3)
  Der LSA-Typ 3 beschreibt die Routen zu den Netzwerken und propagiert die Standardroute in die Bereiche.
- AS Summary LSA (LSA-Typ 4)
  Der LSA-Typ 4 beschreibt die Routen zu den ASBR.

5 **External LSA (LSA-Typ 5 / LSA-Typ 7)**
Die External LSA wird vom ASBR erzeugt. Der LSA-Typ ist abhängig vom Bereich.

7
- AS External LSA (LSA-Typ 5)
  Der LSA-Typ 5 wird vom AS Border Router in die Bereiche des Autonomen System gesendet, außer in die Bereiche Stub und NSSA. Die LSA enthält Informationen über Routen zu einem Netzwerk in einem anderen AS. Die Routen sind entweder manuell erstellt oder extern gelernt. Der ASBR verwendet den LSA-Typ 5, um Standardrouten in den Backbone-Bereich zu verteilen.
- NSSA External LSA (LSA-Typ 7)
  Der LSA-Typ 7 wird vom AS Border Router einer NSSA erzeugt. Der Router wird auch als NSSA ASBR bezeichnet. Der LSA-Typ 7 wird nur innerhalb der NSSA gesendet. Wenn in dem LSA-Typ 7 das P-Bit = 1 ist, werden diese LSAs vom ABR in den LSA-Typ 5 umgewandelt und in den Backbone-Bereich gesendet.

Nachbarschaft aufbauen

Der Router durchläuft folgende Zustände, um eine Verbindung zum Nachbar-Router aufzubauen.

1. Attempt state / Init state
   Der Router aktiviert OSPF und beginnt Hello-Pakete zu senden und zu empfangen. Über die empfangenen Hello-Pakete erfährt der Router, welche OSPF-Router in seiner Nähe sind. Der Router überprüft den Inhalt des Hello-Pakets. In dem Hello-Paket ist auch eine Liste der Nachbar-Router (Neighbor Table) des "Absenders" enthalten.
2. Two way state
   Wenn z. B. die ID des Bereichs, der Bereichstyp und die Einstellungen der Zeiten übereinstimmen, kann eine Verbindung (adjacencies) zum Nachbar aufgebaut werden. In einem Punkt-zu-Punkt-Netzwerk wird direkt die Verbindung aufgenommen. Wenn in einem Netzwerk mehrere Nachbar-Router zu erreichen sind, wird mithilfe von Hello-Paketen der designierte Router (DR) und der Backup Designated Router (BDR) ermittelt. Der Router mit der höchsten Router-Priorität wird designierter Router. Wenn zwei Router die gleiche Router-Priorität haben, wird der Router mit der größten Router-ID zum designierten Router. Der Router nimmt Verbindung zum designierten Router auf.
3. Exchangestart state
   Die Nachbar-Router legen fest, welcher Router mit der Kommunikation beginnt. Der Router mit der höheren Router-ID wird zum Designated Router.
4. Exchange state
   Die Nachbar-Router schicken Pakete, die den Inhalt ihrer Nachbarschaftsdatenbank beschreiben. Die Nachbarschaftsdatenbank (Link state database - LSDB) enthält Informationen zur Topologie des Netzes.

5. Loading state
   Der Router vervollständigt die empfangenen Informationen. Wenn der Router noch Fragen zum Status einer bestimmten Verbindung hat, schickt er eine Anfrage (Link State Request). Der Nachbar-Router schickt eine Antwort (Link State Update). In der Antwort ist eine entsprechende LSA enthalten. Der Router bestätigt den Erhalt der Antwort (Link State Acknowledge).
6. Full State
   Der Informationsaustausch mit dem Nachbar-Router ist abgeschlossen. Die Nachbarschaftsdatenbank der Nachbar-Router ist gleich. Der Router berechnet mithilfe des Short Path First-Algorithmus eine Route zu jedem Ziel. Die Route wird in die Routingtabelle eingetragen.

Nachbarschaft überprüfen

Die Hello-Pakete werden nicht nur verwendet, um Nachbarschaftsbeziehungen aufzubauen. Mit Hello-Paketen wird die Verbindung zum Nachbar-Router überprüft, indem diese zyklisch gesendet werden. Wenn innerhalb eines bestimmten Intervalls (Dead Intervall) kein Hello-Paket empfangen wird, wird die Verbindung zum Nachbarn als "ausgefallen" markiert. Die entsprechenden Einträge werden gelöscht.

Nachbarschaftsdatenbank aktualisieren

Nachdem die Nachbarschaftsdatenbank aufgebaut ist, werden bei Änderungen in der Topologie LSAs an alle Router im Netzwerk versendet.

OSPFv3

Die Version 3 von OSPF basiert auf der Version 2 und wird nur bei IPv6 verwendet. Ein großer Teil der Routing-Mechanismen wurde übernommen. Definiert ist OSPFv3 in den RFCs 2740 und 5340.

Sie konfigurieren OSPFv3 unter "Layer 3 (IPv6) > OSPFv3".

Folgendes hat sich nicht geändert:

- Die Zustände, die ein Router durchläuft, um eine Verbindung zum Nachbar-Router aufzubauen.
- Die Bereiche: Backbone, Stub Area, Totally Stubby Area, Not So Stubby Area (NSSA)
- Die Router-Typen: Interner Router (IR), Area Border Router (ABR), Backbone Router (BR), Autonomous System Area Border Router (ASBR), Designierter Router (DR)
- Die Router-ID, die Area-ID und die ID der LSA werden im IPv4-Adressformat eingegeben: x.x.x.x

Was hat sich geändert?

Begriffe

Die Begriffe Netzwerk bzw. Subnetz werden durch Link ersetzt.

Authentifizierung

Die Authentifizierung wurde entfernt. OSPFv3 verwendet stattdessen IPsec, das in IPv6 implementiert ist.

Nachbar-Router

Die Nachbar-Router werden über die Router-ID identifiziert.

Nachbarschaftsdatenbank

Die Nachbarschaftsdatenbank (Link state database - LSDB) wird in verschiedene Geltungsbereiche aufgeteilt:

- Link-Scope LSDB
  Enthält die Link LSA
- Area-Scope LSDB
  Enthält folgende LSAs
  - Router LSA
  - Network LSA
  - Inter-Area Prefix LSA
  - Inter Area-Router LSA
  - Intra-Area Prefix LSA
- AS-Scope LSDB
  Enthält die AS External LSA

LSA-Typen

Für OSPFv3 wurden zwei neue LSA-Typen definiert.

OSPFv2		OSPFv3		Wer	Inner-halb	Beschreibung
1	Router LSA	0x2001	Router LSA	jeder Router	Bereich	Enthält keine Adressinformationen mehr. Diese sind in dem neuen LSA-Typ 2009 enthalten.
2	Network LSA	0x2002	Network LSA	DR	Bereich	Enthält keine Adressinformationen mehr. Diese sind in dem neuen LSA-Typ 2009 enthalten.
3	Summary LSA	0x2003	Inter-Area Prefix LSA	ABR	Bereich	Gleiche Funktion wie bei OSPFv2, nur umbenannt.
4	AS Summary LSA	0x2004	Inter-Area Router LSA	ABR	Bereich	Gleiche Funktion wie bei OSPFv2, nur umbenannt.
5	AS External LSA	0x4005	AS Exter-nal LSA	ASBR	AS	Gleiche Funktion wie bei OSPFv2, nur umbenannt.
7	NSSA Exter-nal LSA	0x2007	Typ-7 LSA	NSSA ASBR	NSSA	Gleiche Funktion wie bei OSPFv2, nur umbenannt.
		0x2008	Link LSA	jeder Router	Link	Die LSA wird vom Router an alle Router gesendet, die mit diesem verlinkt sind. Die LSA enthält die link local-Adresse des Routers und eine Liste mit IPv6-Präfixen, die auf dem Link konfiguriert sind.
		0x2009	Intra-Area Prefix LSA	jeder Router	Bereich	Die LSA wird nur innerhalb eines Bereichs gesendet. Die LSA enthält die IPv6-Präfixe, die mit dem Router bzw. dem Netz-werk verbunden sind.

Im Gegensatz zu OSPFv2 kann OSPFv3 unbekannte LSA-Typen weiterleiten. Bisher wurden diese gelöscht und nicht weiter verbreitet.

RIP

RIPv2

Dynamisches Routing mit RIPv2

Das Routing Information Protocol (RIPv2) wird verwendet, um Routing-Tabellen automatisch zu erstellen. RIPv2 wird in autonomen Systemen (AS) mit maximal 15 Hops eingesetzt. Es basiert auf dem Distance-Vector-Algorithmus.

RIPv2 wurde von der IETF (Internet Engineering Task Force) entwickelt und ist im RFC 2453 beschrieben.

Sie konfigurieren RIPv2 unter "Layer 3 (IPv4) > RIPv2".

Routing-Tabelle aufbauen

Da ein Router zunächst nur seine direkt angeschlossenen Netzwerke kennt, sendet er eine Anfrage an seine direkten Nachbar-Router. Als Antwort erhält er die Routing-Tabellen der Nachbar-Router. Basierend auf den erhaltenen Informationen, baut der Router eine eigene Routing-Tabelle auf.

Die Routing-Tabelle enthält Einträge für alle möglichen Ziele. Jeder Eintrag enthält die Entfernung zum Ziel und den ersten Router auf der Route.

Die Entfernung wird auch als Metrik bezeichnet. Sie gibt die Anzahl der Router an, die entlang einer Route bis zum Ziel durchlaufen werden (Hop Count). Die maximale Entfernung darf 15 Hops betragen.

Routing-Tabelle aktualisieren

Nachdem die Routing-Tabelle aufgebaut ist, schickt der Router in Intervallen von 30 Sekunden seine Routing-Tabelle über den UDP-Port 520 an jeden direkten Nachbar-Router.

Neue Routing-Informationen gleicht der Router mit seiner bestehenden Routing-Tabelle ab. Wenn die neuen Informationen kürzere Routen enthalten, werden die bestehenden Routen überschrieben. Der Router merkt sich nur die kürzeste Route zu einem Ziel.

Nachbar-Router überprüfen

Wenn ein Router länger als 180 Sekunden keine Nachrichten von einem Nachbar-Router empfängt, markiert er den Router als ungültig. Der Router vergibt für den Nachbar-Router die Metrik 16.

RIPng

RIPng (RIP next Generation) wird nur bei IPv6 verwendet und ist im RFC 2080 definiert. Wie beim RIP (IPv4) basiert RIPng auf dem Distanz-Vektor-Algorithmus von Bellman-Ford.

Im Gegensatz zu RIPv2 wird RIPng direkt an der Schnittstelle (VLAN-Schnittstelle / Router-Port) aktiviert und nicht global auf dem Gerät.

RIPng verwendet den UDP-Port 521 und RIP den UDP-Port 520.

Sie konfigurieren RIPng unter "Layer 3 (IPv6) > RIPng".

PIM

Protocol Independent Multicast (PIM) ermöglicht das Routing von Multicast-Paketen, unabhängig vom unterlagerten Routing-Protokoll, wie OSPFv2 oder statischem Routing (IPv4). PIM erweitert die Routinginformationen des im Router aktiven Unicast-Routing-Protokolls um zusätzliche Informationen für den Multicast-Betrieb.

Voraussetzungen für PIM:

- IGMP ist auf den First-Hop- und Last-Hop-Routern der Routing-Topologie aktiviert.
- PIM ist auf allen Routern der Routing-Topologie aktiviert.
- Es gibt mindestens einen Rendezvous Punkt (RP).
- In jedem Subnetz gibt es einen Designated Router (DR).
- Der DR muss auch der IGMP-Querier sein.

PIM-Netzwerk

PIM arbeitet im Sparse-Mode, der für Netzwerke mit geringer Teilnehmerdichte ausgelegt ist.

Sparse-Mode (SM)

Wenn ein Router einen Multicast empfängt, sendet er diese Information an den Rendezvous Punkt (RP). Ein Gerät, das einen Muliticast empfangen will, sendet einen IGMP-Join an die Router in seinem Subnetz. Der Designated Router (DR) des Netzwerks sendet diese Anfrage dem RP. Der RP verfügt damit über die Unicast-Adresse des Senders und eine Unicast-Adresse für den Empfänger, die des DRs.

Im Sparse-Mode wird zwischen Sender und Empfänger auf der Grundlage der Routing-Tabelle der kürzeste Pfad gewählt und die Informationen können gesendet werden.

Bidirektionaler Multicast und Designated Forwarder

Die Funktionalität "Bidirektionaler Multicast" ist eine Variante des Sparse-Mode, bei der die Informationen zwischen Sender und Empfänger immer über den Rendezvous Punkt gesendet werden.

Beim Bidirektionalen Multicast wird der DR durch den Designated Forwarder (DF) ersetzt. Wenn der DF einen IGMP-Join erhält, sendet er eine (*, G)-Meldung an den RP. Der RP ersetzt den * durch die Quelle und stellt dadurch die Verbindung her. Nur der DF leitet den Multicast weiter.

Rendezvous Punkt (RP) und Bootstrap Router (BSR)

Der Rendezvous Punkt verwaltet die Informationen über Sender und Empfänger. Sie können RPs statisch anlegen oder RP-Kandidaten definieren.

Wenn Sie RP-Kandidaten definieren, benötigen Sie einen Bootstrap Router. Der BSR koordiniert die RP-Kandidaten und bestimmt einen RP. Sie definieren BSR-Kandidaten, von denen einer der Bootstrap Router wird.

BSR-Border

Mit der Funktion "BSR-Border" können Sie ein PIM-Netzwerk in Domänen unterteilen. Zwischen den PIM-Domänen findet keine BSR-Kommunikation statt, sodass in jeder Domäne ein BSR gewählt wird. Der BSR der Domäne bestimmt einen RP für die Domäne.

MSDP

Die RPs kommunizieren über PIM-Domänengrenzen mittels MSDP und synchronisieren die Multicastgruppen, die sie empfangen. So kann ein Gerät in Domäne A einen Multicast empfangen, obwohl die Informationen über den Sender durch den RP in Domäne B verwaltet werden.

Source specific-Multicast (SSM)

PIM Source specific-Multicast ist eine weitere Variante des Sparse-Mode, die ohne Redenzvous Punkte auskommt. Über IGMPv3 teilt der Empfänger dem Router direkt die Quelle und den Multicast mit, den er empfangen möchte.

Beispiel

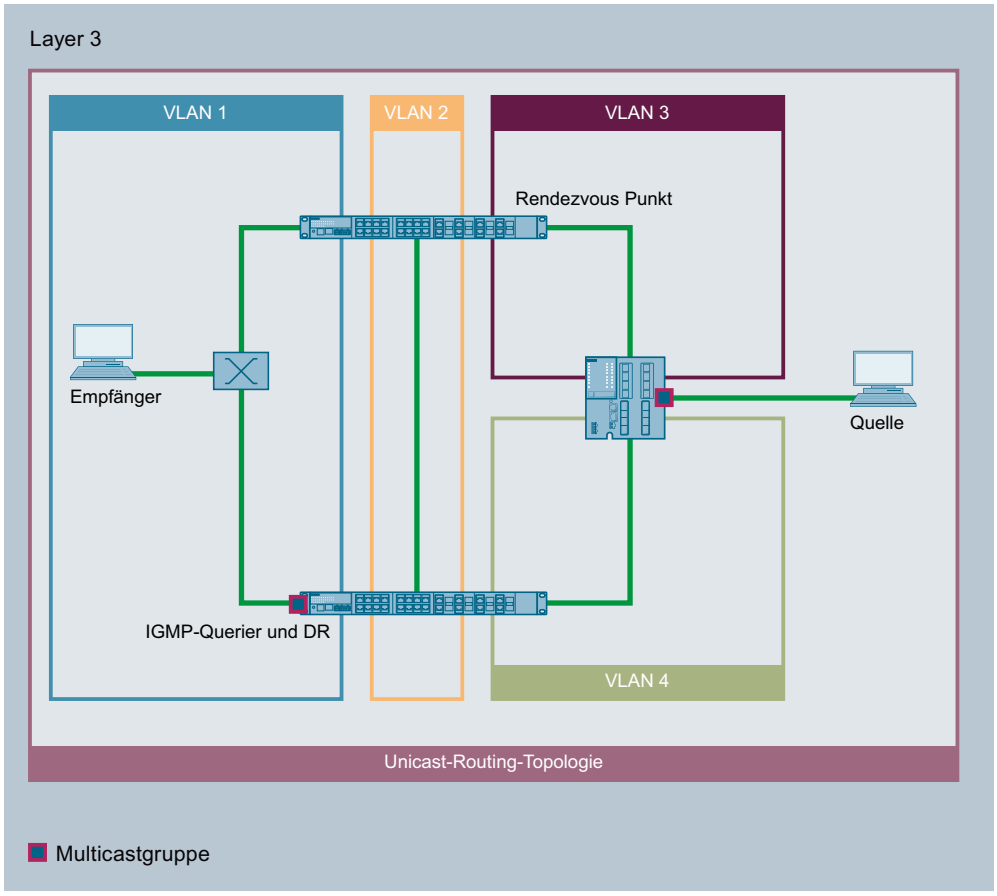

In diesem Beispiel sendet die Quelle einen Multicast an den angeschlossenen Router. Der Router sendet diese Information an den Rendezvous Punkt.

Der Empfänger sendet einen IGMP-Join an seinen IGMP-Querier und Designated Router. Der DR leitet die Anfrage an den RP weiter.

Basierend auf der Routing-Tabelle wird der kürzeste Pfad für den Multicast gewählt.

NAT/NAPT

Hinweis

NAT/NATP ist nur auf Layer 3 des ISO/OSI-Referenzmodells möglich. Zur Nutzung der NAT-Funktion müssen die Netze das IP-Protokoll verwenden.

Bei Verwendung des ISO-Protokolls, das auf Layer 2 arbeitet, ist keine Nutzung von NAT möglich.

Bei der Network Address Translation (NAT) werden IP-Subnetze in "Inside" und "Outside" unterteilt. Die Unterteilung erfolgt aus der Sicht einer NAT-Schnittstelle. Alle Netze, die über die NAT-Schnittstelle selbst erreichbar sind, gelten für diese Schnittstelle als "Outside". Alle

Netze, die über andere IP-Schnittstellen des selben Geräts erreichbar sind, gelten für die NAT-Schnittstelle als "Inside".

Wenn über eine NAT-Schnittstelle geroutet wird, werden die Quell- oder Ziel-IP-Adressen der übertragenen Datenpakete beim Übergang zwischen "Inside" und "Outside" geändert. Ob die Quell- oder Ziel-IP-Adresse geändert wird, ist von der Kommunikationsrichtung abhängig. Es wird immer die IP-Adresse des Kommunikationsteilnehmers angepasst, der sich "Inside" befindet. Abhängig von der Perspektive wird die IP-Adresse eines Kommunikationsteilnehmers als "Local" oder "Global" bezeichnet.

		Perspektive	
		Local	Global
Position	Inside	Eine tatsächliche IP-Adresse, die einem Gerät im internen Netz zugewiesen ist. Diese Adresse ist aus dem externen Netz nicht erreichbar.	Eine IP-Adresse, unter der ein internes Gerät aus dem externen Netz erreichbar ist.
	Outside	Eine tatsächliche IP-Adresse, die einem Gerät im externen Netz zugewiesen ist. Da nur "Inside"-Adressen umgesetzt werden, wird nicht zwischen Outside Local und Outside Global unterschieden.	

Beispiel

In dem Beispiel werden zwei IP-Subnetze über einen IE-Switch miteinander verbunden. Die Unterteilung erfolgt aus der Sicht der NAT-Schnittstelle 10.0.0.155. Die Kommunikation von PC2 mit PC1 wird über NAT/NAPT umgesetzt.

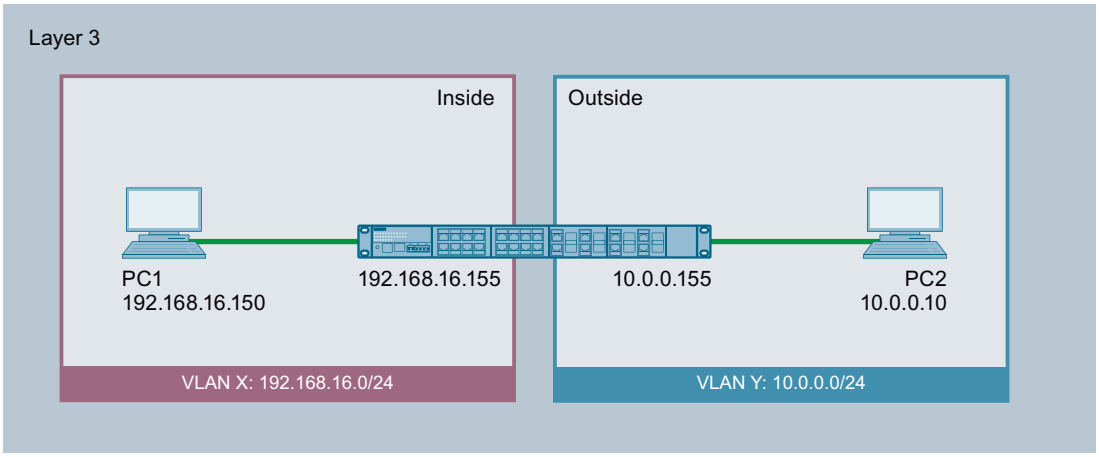

Die tatsächliche IP-Adresse von PC1 (Inside Local) wird statisch mit NAT umgesetzt. Für PC2 ist PC1 unter der Inside Global Adresse erreichbar.

		Perspektive	
		Local	Global
Position	Inside	192.168.16.150	10.0.0.7
	Outside	10.0.0.10	

Die tatsächliche IP-Adresse von PC1 (Inside Local) wird mit NAPT (Network Address and Port Translation) umgesetzt. Für PC2 ist PC1 unter der Inside Global Adresse erreichbar.

		Perspektive	
		Local	Global
Position	Inside	192.168.16.150:80	10.0.0.7:80
	Outside	10.0.0.10:1660	

Rechenkapazität

Der Paketempfang des Geräts ist aufgrund der Lastbegrenzung zur CPU auf 300 Pakete pro Sekunde limitiert. Das entspricht einem maximalen Datendurchsatz von 1,7 MBit/s. Diese Lastbegrenzung gilt nicht pro Schnittstelle, sondern generell für alle Pakete, die zur CPU gehen.

Die gesamte NAT-Kommunikation läuft über die CPU und steht damit in Konkurenz zur IP-Kommunikation, die zur CPU geht, z. B. WBM und Telnet.

Beachten Sie, dass ein großer Teil der Rechenkapazität belegt ist, wenn Sie NAT einsetzen. Dadurch kann der Zugriff über Telnet oder WBM verlangsamt werden.

NAT

Mit "Network Address Translation" (NAT) wird die IP-Adresse in einem Datenpaket durch eine andere ersetzt. NAT wird in der Regel an einem Netzübergang zwischen einem internen Netz und einem externen Netz eingesetzt.

Beim Source-NAT wird die Inside Local-Quell-Adresse eines IP-Pakets von einem Gerät im internen Netz durch ein NAT-Gerät am Netzübergang in eine Inside Global-Adresse umgeschrieben.

Beim Destination-NAT wird die Inside Global-Ziel-Adresse eines IP-Pakets von einem Gerät im externen Net durch ein NAT-Gerät am Netzübergang in eine Inside Local-Adresse umgeschrieben.

Zur Übersetzung der internen in die externe IP-Adresse und zurück pflegt das NAT-Gerät eine Übersetzungsliste. Die Adresszuordnung kann dynamisch oder statisch sein. Sie konfigurieren NAT unter "Layer 3 (IPv4) > NAT (Seite 2345)".

NAPT

Bei "Network Address Port Translation" (NAPT) werden mehrere interne IP-Adressen in die gleiche externe IP-Adresse umgeschrieben. Zur Identifikation der einzelnen Teilnehmer wird auch der Port des internen Geräts in der Übersetzungsliste des NAT-Geräts gespeichert und für die externe Adresse umgeschrieben.

Wenn mehrere interne Geräts über das NAT-Gerät eine Anfrage an die gleiche externe Ziel-IP-Adresse senden, trägt das NAT-Gerät jeweils seine eigene externe Quell-IP-Adresse in den Header dieser weitergeleiteten Telegramme ein. Da die weitergeleiteten Telegramme die gleiche externe Quell-IP-Adresse haben, ordnet das NAT-Gerät die Telegramme den Geräten über eine unterschiedliche Portnummer zu.

Wenn ein Gerät aus dem externen Netz einen Dienst im internen Netz nutzen möchte, muss die Übersetzungsliste für die statische Adresszuordnung konfiguriert werden. Sie konfigurieren NAPT unter "Layer 3 (IPv4) > NAT > NAPT (Seite 2348)".

NAT/NAPT und IP-Routing

Sie können NAT/NAPT und IP-Routing gleichzeitig aktivieren. In diesem Fall müssen Sie die Erreichbarkeit interner Adressen aus externen Netzwerken durch ACL-Regeln regulieren.

Single-Hop Inter-VLAN-Routing

Einleitung

Ein physikalisches Netzwerk wird durch VLANs in Broadcastdomänen und Subnetze unterteilt.

Geräte (Hosts) innerhalb eines VLANs können über Layer 2 direkt miteinander kommunizieren. Die Telegramme werden anhand der MAC-Adresse an das entsprechende Gerät weitergeleitet.

Geräte aus unterschiedlichen VLANs können nicht über Layer 2 direkt miteinander kommunizieren. Der Datenverkehr muss anhand der IP-Adresse geroutet werden.

Mit der Funktion Single-Hop Inter-VLAN-Routing ist es möglich, dass Geräte aus unterschiedlichen VLANs miteinander kommunizieren, ohne dass ein Router benötigt wird.

Voraussetzungen

- Der IE-Switch kann mehrere IP-Schnittstellen verwalten.
- Der IE-Switch ist Mitglied in den VLANs, zwischen denen geroutet werden soll.
- Bei den Hosts ist die IP-Adresse des VLANs als Default-Gateway eingetragen.

Single-Hop Inter-VLAN-Routing

Der IE-Switch erhält ein Telegramm und erkennt, dass es an ein Gerät in einem anderen VLAN adressiert ist. Er leitet das Telegramm an den entsprechenden Port in dem VLAN weiter.

Der IE-Switch kennt nur VLANs, mit denen er direkt verbunden ist (Connected). Beim Single-Hop Inter-VLAN-Routing ist es daher nur möglich zwischen zwei lokalen IP-Schnittstellen zu routen.

Beispiel

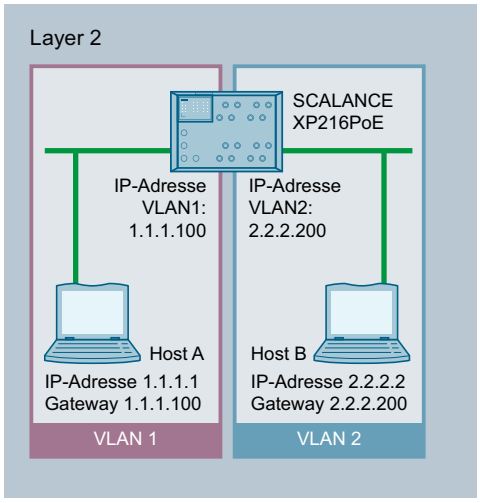

In diesem Beispiel ist der Host A über das VLAN 1 mit dem IE-Switch verbunden. Host B ist über VLAN 2 mit dem IE-Switch verbunden. Bei Host A ist die IP-Adresse von VLAN 1 als Default-Gateway eingetragen. Bei Host B ist die IP-Adresse von VLAN 2 als Default-Gateway eingetragen.

Wenn die Funktion Single-Hop Inter-VLAN-Routing auf dem SCALANCE XP216PoE aktiviert ist, können Host A und Host B miteinander kommunizieren.

Link Aggregation

Link Aggregation

Bei der Link Aggregation werden mehrere parallel verlaufende physikalische Verbindungen gleicher Übertragungsgeschwindigkeit zu einer logischen Verbindung mit höherer Übertragungsgeschwindigkeit zusammengefasst. Dieses Verfahren nach IEEE 802.3ad wird auch Port Trunking oder Channel Bundling genannt.

Link Aggregation funktioniert nur mit Vollduplex-Verbindungen gleicher Übertragungsgeschwindigkeit im Punkt-zu-Punkt-Betrieb. Dadurch wird eine Vervielfachung der Bandbreite bzw. der Übertragungsgeschwindigkeit erreicht. Wenn ein Teil der Verbindung ausfällt, wird der Datenverkehr über die verbliebenen Verbindungsteile abgewickelt.

Zur Steuerung und Kontrolle wird der Link Aggregation Control Layer (LACL) und das Link Aggregation Control Protocol (LACP) verwendet.

Quality of Service

Quality of Service (QoS) ist ein Verfahren, um die vorhandene Bandbreite in einem Netzwerk effizient zu nutzen.

QoS wird durch die Priorisierung des Datenverkehrs realisiert. Ankommende Frames werden nach einer bestimmten Priorisierung in eine Warteschlange (Queue) einsortiert und weiterverarbeitet. Dadurch wird bestimmten Frames Vorrang gewährt.

Die unterschiedlichen QoS-Verfahren beeinflussen sich gegenseitig und werden daher in folgender Reihenfolge berücksichtigt:

1. Der Switch prüft zunächst, ob es sich bei dem ankommenden Frame um ein Broadcast- oder Agent-Frame handelt.
   → Wenn die 1. Bedingung erfüllt ist, berücksichtigt der Switch die eingestellte Priorität auf der Seite "Allgemein (Seite 2268)".
   Der Switch sortiert das Frame in eine Queue ein, entsprechend der Zuordnung auf der Seite "CoS-Zuordnung (Seite 2269)".
2. Wenn die 1. Bedingung nicht erfüllt ist, prüft der Switch, ob das Frame ein VLAN-Tag enthält.
   → Wenn die 2. Bedingung erfüllt ist, prüft der Switch die Einstellungen zur Priorität auf der Seite "Allgemein (Seite 2275)". Der Switch prüft, ob für die Priorität ein anderer Wert als "Nicht überschreiben" eingestellt ist.
   Wenn eine Priorität eingestellt ist, sortiert der Switch das Frame in eine Queue ein, entsprechend der Zuordnung auf der Seite "CoS-Zuordnung (Seite 2269)".
3. Wenn die 2. Bedingung ebenfalls nicht erfüllt ist, werden die Frames entsprechend des Priorisierungsmodus weiterverarbeitet. Den Priorisierungsmodus konfigurieren Sie auf der Seite "QoS-Priorisierung (Seite 2271)".

Authentifizierung

Übersicht zur Benutzerverwaltung

Der Zugriff auf das Gerät wird durch konfigurierbare Benutzereinstellungen verwaltet. Richten Sie Benutzer mit jeweils einem Passwort zur Authentifizierung ein. Weisen Sie den Benutzern eine Rolle mit entsprechenden Rechten zu.

Die Authentifizierung von Benutzern kann entweder von dem Gerät lokal oder von einem externen RADIUS-Server durchgeführt werden. Wie die Authentifizierung erfolgen soll, konfigurieren Sie auf der Seite "Security > AAA > Allgemein".

Wenn Sie die Konfiguration eines Geräts nach TIA übertragen, werden die konfigurierten Benutzer, Rollen und Gruppen nicht übertragen.

Kompatibilität mit Vorgängerversionen

Mit der Firmware-Version 5.1 wurde die Benutzerverwaltung um den RADIUS-Autorisierungsmodus "Herstellerspezifisch" erweitert. Um die Kompatibilität mit Firmware-Versionen ≤ 5.0 zu gewährleisten, wurde die Default-Einstellung so gewählt, dass nach einem Firmware-Update weiterhin der frühere Autorisierungsmodus "Standard" verwendet wird.

Lokale Anmeldung

Die lokale Anmeldung von Benutzern durch das Gerät läuft wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät prüft, ob ein Eintrag für den Benutzer vorhanden ist:
   → Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet.
   → Wenn kein entsprechender Eintrag existiert, wird dem Benutzer der Zugriff verweigert.

Anmeldung über einen externen RADIUS-Server

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Auhentifizierung und Autorisierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können.

Anhängig davon, welchen RADIUS-Autorisierungsmodus Sie auf der Seite "Security > AAA > RADIUS-Client" eingestellt haben, wertet das Gerät unterschiedliche Informationen des RADIUS-Servers aus.

RADIUS-Autorisierungsmodus "Standard"

Wenn Sie den RADIUS-Autorisierungsmodus "Standard" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück:
     → Der Benutzer wird mit Administratorrechten angemeldet.
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt einen anderen oder gar keinen Wert für das Attribut "Service Type" an das Gerät zurück:
     → Der Benutzer wird mit Leserechten angemeldet.
   - Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
     → Dem Benutzer wird der Zugriff verweigert.

RADIUS-Autorisierungsmodus "Herstellerspezifisch"

Voraussetzung

Für den RADIUS-Autorisierungsmodus "Herstellerspezifisch" ist am RADIUS-Server Folgendes einzustellen:

- Herstellercode: 4196
- Attributnummer: 1
- Attributformat: Zeichenfolge (Gruppenname)

Vorgehen

Wenn Sie den RADIUS-Autorisierungsmodus "Herstellerspezifisch" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   Fall A: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt die dem Benutzer zugeordnete Gruppe an das Gerät zurück.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird mit den Rechten der zugeordneten Gruppe angemeldet.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird der Rolle mit den größeren Rechten zugeordnet und mit diesen Rechten angemeldet.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle angemeldet, die mit seinem Benutzeraccout verknüpft ist.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall B: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung, gibt jedoch keine Gruppe an das Gerät zurück:
   - Der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der verknüpften Rolle angemeldet.
   - Der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall C: Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
   - Dem Benutzer wird der Zugriff verweigert.

Authentifizierungsverfahren

Sie können portgranular die Authentifizierungsverfahren "802.1X" und "MAC-Authentifizierung" sowie die Option "Guest VLAN" konfigurieren.

Die Funktionen unterliegen einer hierarchischen Ordnung. Wenn alle drei Funktionen aktiviert sind, wird zunächst versucht, das Endgerät über "802.1x" zu authentifizieren. Ist diese Authentifizierung nicht erfolgreich, wird "MAC-Authentifizierung" gestartet. Ist auch diese Authentifizierung nicht erfolgreich, wird das Endgerät im "Guest VLAN" zur Kommunikation freigegeben. "Guest VLAN" kann nur eingesetzt werden, wenn mindestens ein Authentifizierungsverfahren aktiv ist.

Die beiden Authentifizierungsverfahren sind vom Endgerät abhängig. Wenn das Endgerät EAP (Extensible Authentication Protocol) unterstützt, kann es über das Verfahren "802.1X" authentifiziert werden. Wenn das Endgerät kein EAP unterstützt, kann es über "MAC-

Authentifizierung" authentifiziert werden. In diesem Fall übernimmt der IE-Switch die Rolle des Endgeräts und benutzt die MAC-Adresse des Geräts als Authentifizierungsparameter.

802.1X

Das Authentifizierungsverfahren "802.1X" läuft wie folgt ab:
Ein Endgerät, das EAP unterstützt, sendet Authentifizierungsinformationen an den IE-Switch. Der IE-Switch übermittelt die Informationen an den Authentifizierungsserver. Der Authentifizierungsserver prüft die Informationen und ermöglicht bzw. verweigert dem Endgerät den Zugriff auf das Netzwerk.

MAC-Authentifizierung

Das Authentifizierungsverfahren "MAC-Authentifizierung" läuft wie folgt ab:
Sobald der IE-Switch ein Telegramm von einem Endgerät empfängt, sendet der IE-Switch eine Anfrage an den RADIUS-Server, um dem Endgerät den Zugang zum Netzwerk zu ermöglichen oder zu verweigern.

Zuweisung eines VLANs über RADIUS oder Guest VLAN

Authentifizierung mit Änderung der VLAN-Konfiguration

Wenn ein Port bei der Authentifizierung über die Funktion "VLAN-Zuordnung von RADIUS übernehmen" oder "Guest VLAN" dynamisch einem VLAN zugewiesen wird, passiert Folgendes:

- Wenn das VLAN, das zugewiesen werden soll, nicht auf dem Gerät angelegt ist, wird die Authentifizierung abgelehnt.
- Wenn das VLAN, das zugewiesen werden soll, auf dem Gerät angelegt ist:
  - Der Port wird Untagged Member in dem zugewiesenen VLAN, wenn er es bisher nicht war.
  - Die PVID des Ports wird auf die ID des zugewiesenen VLANs geändert.

Hinweis

Wenn der Port nur einem VLAN zugeordnet sein soll, müssen Sie die VLAN-Konfiguration manuell anpassen. Alle Ports sind z. B. standardmäßig Untagged Member in "vlan 1".

Wenn die Authentifizierung zurückgenommen wird, z. B. durch einen Link-Down, werden die dynamischen Änderungen zurückgenommen:

- Der Port ist kein Member mehr in dem zugewiesenen VLAN.
- Die PVID des Ports wird auf den Wert zurückgesetzt, den sie vor der Authentifizierung hatte.

Authentifizierung ohne Änderung der VLAN-Konfiguration

Wenn bei der Authentifizierung weder durch die Funktion "VLAN-Zuordnung von RADIUS übernehmen" noch durch "Guest VLAN" ein VLAN zugewiesenen wird, bleibt die bestehende VLAN-Konfiguration des Ports unverändert.

Eigenschaften und Parameter bearbeiten

Möglichkeiten zum Bearbeiten

Zum Bearbeiten von Eigenschaften und Parameter gibt es folgende Möglichkeiten:

- Hardware- und Netzwerkeditor
  Nachdem Sie die Netzwerkkomponente eingefügt haben, können Sie die Eigenschaften und Parameter bearbeiten, z. B. den Gerätenamen. Weitere Informationen dazu finden Sie unter "Hardware- und Netzwerkeditor".
- Web Based Management (WBM)
  Die Parameter und Eigenschaften sind über mitgelieferten HTML-Seiten (WBM-Seiten) erreichbar. Jede WBM-Seite hat eine eigene Hilfeseite, die die Eigenschaften und Parameter beschreiben. Beachten Sie für weitere Informationen die Projektierungshandbücher:
  - "SCALANCE XB-200/XC-200/XP-200 Web Based Management"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter SCALANCE XB-200 (https://support.industry.siemens.com/cs/ww/de/ps/15291/man), SCALANCE XC-200 (https://support.industry.siemens.com/cs/ww/de/ps/15285/man) und SCALANCE XP-200 (https://support.industry.siemens.com/cs/ww/de/ps/21869/man).
  - "SCALANCE XM-400/XR-500 Web Based Management"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter SCALANCE XM-400 (https://support.industry.siemens.com/cs/ww/de/ps/15315/man) und SCALANCE XR-500 (https://support.industry.siemens.com/cs/ww/de/ps/15317/man).
- Command Line Interface
  Mit dem CLI können alle Konfigurationseinstellungen für das Gerät festgelegt werden. Das CLI bietet die gleichen Möglichkeiten wie das Web Based Management (WBM). Beachten Sie für weitere Informationen die Projektierungshandbücher:
  - "SCALANCE XB-200/XC-200/XP-200 Command Line Interface"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter SCALANCE XB-200 (https://support.industry.siemens.com/cs/ww/de/ps/15291/man), SCALANCE XC-200 (https://support.industry.siemens.com/cs/ww/de/ps/15285/man) und SCALANCE XP-200 (https://support.industry.siemens.com/cs/ww/de/ps/21869/man).
  - "SCALANCE XM-400/XR-500 Command Line Interface"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter SCALANCE XM-400 (https://support.industry.siemens.com/cs/ww/de/ps/15315/man) und SCALANCE XR-500 (https://support.industry.siemens.com/cs/ww/de/ps/15317/man).

Verfügbarkeit

Die Verfügbarkeit der Einstellungen ist abhängig

- vom Porttyp
- vom Projektierungsmodus

SCALANCE X unterscheidet folgende Porttypen:

- Switch-Port
- Router-Port

Folgende Projektierungsmodi gibt es:

- Offline-Projektierung
  Die Erstkonfiguration einer PC-Station können Sie offline durchführen. In diesem Modus sind nur die Einstellungen verfügbar, die keine Verbindung zum Gerät benötigen.
- Online-Projektierungsmodus
  Wenn eine Verbindung zum Gerät besteht, sind im Inspektorfenster zusätzliche Seiten verfügbar. Diese Seiten enthalten den Hinweis "Diese Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht". Bei einzelnen Einstellungen ist der Zusatz "Nur Online verfügbar" enthalten.

Eintrag anlegen und löschen

Als Beispiel wird ein Eintrag beim Syslog-Client angelegt und gelöscht. Das Vorgehen ist bei allen Seiten gültig.

Neuen Eintrag anlegen

1. Selektieren Sie in der Netz- oder Gerätesicht das Gerät.
2. Öffnen Sie im Inspektorfenster die Eigenschaften des Geräts.
3. Gehen Sie im Inspektorfenster zu System > Syslog-Client.
4. Tragen Sie bei IP-Adresse des Syslog-Servers die IP-Adresse ein.
5. Klicken Sie in die Tabelle.
6. Wählen Sie im Kontextmenü den Eintrag "Neuer Eintrag".
   In der Tabelle wird ein neuer Eintrag erstellt.

Eintrag löschen

1. Markieren Sie in der Tabelle den gewünschten Eintrag.
2. Wählen Sie im Kontextmenü den Eintrag "Löschen".

Häufig verwendete Schaltflächen

- **Aktualisieren der Anzeige mit "Aktualisieren"**
  Seiten, die aktuelle Parameter anzeigen, haben am unteren Rand die Schaltfläche "Aktualisieren". Klicken Sie auf diese Schaltfläche, wenn Sie für die angezeigte Seite aktuelle Daten vom Gerät anfordern wollen.
- **Speichern von Einstellungen für alle Ports mit "In Tabelle übernehmen"**
  Seiten, auf denen Sie mehrere Ports konfigurieren können, verfügen über 2 Tabellen. In der ersten Tabelle können Sie Einstellungen für alle Ports vornehmen und in die zweite Tabelle übernehmen. In der letzten Spalte der ersten Tabelle gibt es die Schaltfläche "In Tabelle übernehmen". Klicken Sie auf die Schaltfläche, um eingegebene Einstellungen für alle Ports zu speichern.

IP-Adresse vergeben

Konfigurationsmöglichkeiten

Im Auslieferungszustand und nach dem Zurücksetzen auf Werkseinstellungen besitzt das Gerät keine IP-Adresse.

Es gibt folgende Möglichkeiten dem Gerät eine IP-Adresse zuzuweisen:

- DHCP (Standard-Einstellung)
- Primary Setup Tool
  Beachten Sie für weitere Informationen das Projektierungshandbuch "Primary Setup Tool". Das Projektierungshandbuch finden Sie Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 19440762 (http://support.automation.siemens.com/WW/view/de/19440762).
- STEP 7
- CLI über die serielle Schnittstelle
  Beachten Sie für weitere Informationen das Projektierungshandbuch "SCALANCE XM-400/XR-500 Command Line Interface". Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Automation and Drives Service & Support (http://support.automation.siemens.com/WW/view/de/48805144/133300).

Voraussetzung

- In den Eigenschaften des Geräts ist "IP-Adresse am Gerät einstellen" aktiviert. Weitere Informationen finden Sie unter "Adressierung von PROFINET-Geräten".

SCALANCE-Gerät als IO-Device konfigurieren

Bei einem SCALANCE-Gerät, das als PROFINET IO-Device projektiert und einem IO-Controller zugeordnet ist, werden bei den Funktionen "Übersetzen" und "Laden in Gerät" nur diejenigen Daten in das SCALANCE-Gerät geladen, die auch im Web Based Management (WBM) projektierbar sind (System, Layer 2, Layer 3, Security).

Wenn Sie die Funktionen "Übersetzen" oder "Laden in Gerät" für die PROFINET IO-Device-Daten des Geräts ausführen wollen, dann selektieren Sie zuvor den zugeordneten IO-Controller.

Upload der Konfigurationsdaten bei einem PROFINET IO-Device

Bei einem SCALANCE-Gerät, das einem PROFINET IO-Controller zugeordnet ist, müssen Sie zwischen den Daten der PROFINET IO-Device-Konfiguration und der SCALANCE-Konfiguration unterscheiden:

- Die PROFINET IO-Device-Konfiguration kann nur vom PROFINET IO-Controller geliefert werden.
- Die SCALANCE-Konfiguration bezieht sich nur auf die Daten, die im Web Based Management (WBM) projektierbar sind (System, Layer 2, Layer 3, Security).

Um die gesamte Konfiguration eines SCALANCE-Geräts zu laden, gehen Sie wie folgt vor:

1. Laden Sie die PROFINET IO-Device-Konfiguration über den PROFINET IO-Controller:
   - Selektieren Sie in der Projektnavigation den PROFINET IO-Controller.
   - Wählen Sie in der Toolbar den Befehl "Online > Laden des Geräts als neue Station (Hardware und Software)".
   - Siehe auch Kapitel "Projektdaten von einem Gerät laden"
2. Laden Sie die SCALANCE-Konfiguration als lokalen Upload über das Gerät:
   - Selektieren Sie in der Netzsicht das SCALANCE-Gerät.
   - Wählen Sie im Kontextmenü des Geräts den Befehl "SCALANCE-Konfiguration > Upload auf PG/PC".
   - Siehe auch Kapitel "Geräte konfigurieren"

Informationen anzeigen

Versionen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Ausgabestände der Hardware und der Software des Geräts an.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Hardware**
  - Basic Device
    Zeigt das Grundgerät an
  - PX.X
    X.X = Port, bei dem das SFP-Modul gesteckt ist.
  - SlotX
    "X" = Steckplatznummer: In diesem Steckplatz eingestecktes Modul.
- **Name**
  Zeigt den Namen des Geräts oder des Moduls an.
- **Ausgabestand**
  Zeigt den Hardware-Ausgabestand des Geräts an.
- **Artikelnummer**
  Zeigt die Bestellnummer des Geräts oder des beschriebenen Moduls an.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Software
  - Firmware
    Zeigt die aktuelle Firmware-Version an. Wenn eine neue Firmware-Datei geladen wurde und das Gerät noch nicht neu gestartet ist, wird hier die Firmware-Version der geladenen Firmware-Datei angezeigt. Nach dem nächsten Neustart wird die geladene Firmware aktiviert und verwendet.
  - Bootloader
    Zeigt die Version der Boot-Software an, die im Gerät gespeichert ist.
- Beschreibung
  Zeigt die Kurzbeschreibung der Software an.
- Version
  Zeigt die Versionsnummer des Software-Ausgabestands an.
- Datum
  Zeigt das Erstellungsdatum des Software-Ausgabestands an.

I&M

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite beinhaltet Informationen zu gerätespezifischen Hersteller- und Wartungsdaten wie Artikelnummer, Seriennummer, Versionsnummern etc. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Zeilen:

- Hersteller-ID
  Zeigt die Herstellerkennung an.
- Artikelnummer
  Zeigt die Artikelnummer an.
- Seriennummer
  Zeigt die Seriennummer an.
- Hardware-Ausgabestand
  Zeigt den Hardware-Ausgabestand an.
- Software-Ausgabestand
  Zeigt den Software-Ausgabestand an.
- Versionszähler
  Zeigt den Versionsänderungs-Zähler an: Zähler für Versionsänderungen seit der Erst-Inbetriebnahme
- Aktualisierungsdatum
  Datum und Uhrzeit der letzten Versionsänderung

- **Funktionskennzeichen**
  Zeigt das Funktionskennzeichen (Anlagenkennzeichen) des Geräts an. Das Anlagenkennzeichen (AKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Ortskennzeichen**
  Zeigt das Ortskennzeichen des Geräts an. Das Ortskennzeichen (OKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Datum**
  Zeigt das Datum, das bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.
- **Deskriptor**
  Zeigt die Beschreibung, die bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.

ARP / Nachbar-Tabelle

ARP-Tabelle

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zuordnung von MAC-Adresse und IP-Adresse

Über das Address Resolution Protocol (ARP) erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv4-Adresse. Dieser Zuordnung wird von jedem Netzteilnehmer in seiner eigenen ARP-Tabelle gepflegt. Die Seite zeigt die ARP-Tabelle des Geräts.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IP-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

IPv6-Nachbarschaftstabelle

Zuordnung von MAC-Adresse und IPv6-Adresse

Über die IPv6-Nachbarschaftstabelle erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv6-Adresse. Diese Zuordnung wird von jedem Netzteilnehmer in seiner eigenen Nachbarschaftstabelle gepflegt..

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IPv6-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

Log-Tabelle

Protokollierung von Ereignissen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Das Gerät bietet die Möglichkeit, auftretende Ereignisse zu protokollieren, die Sie zum Teil unter "System > Ereignisse" festlegen können. So kann beispielsweise festgehalten werden, wann ein Authentifizierungsversuch fehlgeschlagen ist oder wann sich der Verbindungsstatus eines Ports geändert hat. Der Inhalt der Ereignisprotokoll-Tabelle bleibt auch nach dem Ausschalten des Geräts erhalten.

Beschreibung der angezeigten Werte

- **Severity-Filter**

Hinweis

Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

Sie können die Einträge der Tabelle nach Fehlerschwere filtern. Wählen Sie in den Optionskästchen oberhalb der Tabelle die gewünschten Einträge aus.

– Info
  Information

– Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Info" angezeigt.

– Warning
  Warnung
  Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Warning" angezeigt.

– Critical
  Kritisch
  Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

– **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis aufgetreten ist.

– **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.

– **Systemzeit**
  Zeigt das Datum und die Uhrzeit an, zu der das beschriebene Ereignis aufgetreten ist. Wenn die Systemzeit gesetzt ist, wird auch die Zeit angezeigt, bei der das Ereignis eingetreten ist.

– **Severity**
  Zeigt den Schweregrad der Meldung an.

– **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an. Eine Liste der möglichen Meldungen finden Sie im Anhang D des Projektierungshandbuches..

Fehler

Fehlerstatus

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Wenn ein Fehler auftritt, wird er auf dieser Seite angezeigt. Am Gerät werden Fehler dadurch signalisiert, dass die rote Fehler-LED leuchtet.

Gemeldet werden interne Fehler des Geräts sowie Fehler, die Sie auf folgenden Seiten konfigurieren:

- "System > Ereignisse"
- "System > Fehlerkontrolle"

Fehler des Ereignisses "Kalt-/Warmstart" können Sie durch eine Bestätigung löschen.

Die Berechnung des Fehlerzeitpunkts beginnt jeweils nach dem letzten Systemstart.

Wenn keine Fehler vorliegen, schaltet sich die Fehler-LED ab.

Beschreibung der angezeigten Werte

- **Anzahl der gemeldeten Fehler**
  Zeigt die Anzahl der gemeldeten Fehler an.
- **Zähler zurücksetzen**
  Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um den Zähler zurückzusetzen. Der Zähler wird auch durch einen Neustart zurückgesetzt.

Die Tabelle enthält die folgenden Spalten:

- **Fehlerzeitpunkt**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu dem der beschriebene Fehler aufgetreten ist.
- **Fehlerbeschreibung**
  Zeigt eine Kurzbeschreibung des aufgetretenen Fehlers an.
- **Fehlerstatus löschen**
  Wenn die Schaltfläche "Fehlerstatus löschen" aktiv ist, können Sie den Fehler löschen.
- Schaltfläche "Zähler zurücksetzen"
  Klicken Sie auf "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden durch einen Neustart zurückgesetzt.

Redundanz

Spanning Tree

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt die aktuellen Informationen zu Spanning Tree und die Einstellungen der Root-Bridge an.

Beschreibung der angezeigten Werte

- **Spanning Tree-Modus**
  Zeigt den eingestellten Modus an. Den Modus legen Sie bei "Layer 2 > Spanning Tree > Allgemein" fest.
  Folgende Werte sind möglich:
  - '-'
  - STP
  - RSTP
  - MSTP
- **Instanz-ID**
  Zeigt die Nummer der Instanz an. Der Parameter ist abhängig vom projektierten Modus.
- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root-Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root-Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root-Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root-Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096.
- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Kosten**
  Zeigt die Pfadkosten von dem Gerät bis zur Root-Bridge.
- **Bridge-Status**
  Zeigt den Status der Bridge an, z. B. ob das Gerät die Root-Bridge ist.
- **Root-Priorität regional** (nur bei MSTP verfügbar)
  Beschreibung siehe Bridge-Priorität / Root-Priorität.

- **Root-Adresse regional** (nur bei MSTP verfügbar)
  Zeigt die MAC-Adresse des Geräts an.
- **Root-Kosten regional** (nur bei MSTP verfügbar)
  Zeigt die Pfadkosten von dem Gerät bis zur regionalen Root-Bridge an.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt den Port an, über den das Gerät kommuniziert.
- **Rolle**
  zeigt den Status des Ports an. Folgende Werte sind möglich:
  - Disabled
    Der Port wurde manuell aus dem Spanning Tree entfernt und wird vom Spanning Tree nicht mehr berücksichtigt.
  - Designated
    Die Ports, die von der Root-Bridge wegführen.
  - Alternate
    Der Port mit einem alternativen Weg zu einem Netzwerksegment.
  - Backup
    Wenn ein Switch mehrere Ports zu dem gleichen Netzwerksegment hat, wird der "schlechtere" Port zum Backup-Port.
  - Root
    Der Port, der den besten Weg zur Root Bridge bietet.
  - Master
    Dieser Port zeigt zu einer Root-Bridge, die außerhalb der MST-Region liegt.
- **Status**
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt. Der Parameter ist abhängig vom projektierten Protokoll. Folgende Status sind möglich:
  - Discarding
    Der Port empfängt BPDU-Telegramme. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    Der Port empfängt und sendet BPDU-Telegramme. Der Port ist in den Spanning Tree-Algorithmus einbezogen. Andere aus- und eingehende Telegramme werden verworfen.
  - Learning
    Der Port lernt aktiv die Topologie, d. h. die Teilnehmeradressen. Andere aus- und eingehende Telegramme werden verworfen.
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz. Der Port empfängt und sendet Datentelegramme.
- **Oper. Version**
  Zeigt, welches Spanning Tree Protokoll vom Port verwendet wird.

- **Priorität**
  Kann der vom Spanning-Tree ermittelte Weg alternativ über mehrere Ports eines Geräts führen, so wird der Port mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) ausgewählt. Für die Priorität kann ein Wert von 0 bis 240 in 16er Schritten eingegeben werden. Wenn Sie einen Wert eingeben, der nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
- **Pfadkosten**
  Dieser Parameter dient zur Berechnung des zu wählenden Wegs. Es wird die Strecke mit dem geringsten Wert als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert, wird der Port mit der niedrigsten Portnummer ausgewählt.
  Ist der Wert "Kalk. Kosten" "0", so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert von "Kalk. Kosten" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000.

  Sie konfigurieren die "Kalk. Kosten" auf den Seiten "Layer 2 > Spanning Tree > CIST-Port" und "Layer 2 > Spanning Tree > MST-Port".
- **Edge-Typ**
  Zeigt den Typ der Verbindung an. Folgende Werte sind möglich:
  - Edge Port
    An diesem Port befindet sich ein Endgerät.
  - No Edge Port
    An diesem Port befindet sich ein Spanning Tree-Gerät.
- **P.t.P. Typ**
  zeigt die Art der Punkt-zu-Punkt-Verbindung an. Folgende Werte sind möglich:
  - P.t.P.
    Bei Halbduplex wird von einer Punkt-zu-Punkt-Verbindung ausgegangen.
  - Shared Media
    Bei einer Vollduplexverbindung wird nicht von einer Punkt-zu-Punkt-Verbindung ausgegangen.

VRRP-Statistiken

Einleitung

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Statistiken des VRRP-Protokolls und aller konfigurierten virtuellen Router an.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- VRID-Fehler
  Zeigt an, wie viele VRRP-Telegramme empfangen wurden, die eine nicht unterstütze VRID enthalten.
- Versionsfehler
  Zeigt an, wie viele VRRP-Telegramme empfangen wurden, die eine ungültige Versionsnummer enthalten.
- Prüfsummenfehler
  Zeigt an, wie viele VRRP-Telegramme empfangen wurden, die eine ungültige Prüfsumme enthalten.

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstelle
  Schnittstelle, auf die sich die Einstellungen beziehen.
- VRID
  Zeigt die ID des virtuellen Routers an.
- Übergang in Master-Zustand
  Zeigt an, wie häufig dieser virtuelle Router die Rolle "Master" eingenommen hat.
- Empf. Advertisements
  Zeigt an, wie häufig ein VRRP-Telegramm empfangen wurde, das eine falsche Adressliste enthalten hat.
- Advertisement-Intervall-Fehler
  Zeigt an, wie viele fehlerhafte VRRP-Telegramme empfangen wurden, deren Intervall nicht mit dem lokal gesetzten Wert übereinstimmt.
- IP TTL Fehler
  Zeigt an, wie viele fehlerhafte VRRP-Telegramme empfangen wurden, deren TTL (Time to live) Wert im IP Header nicht stimmt.
- Empf. Prio-0-Tel.
  Zeigt an, wie viele VRRP-Telegramme mit der Priorität 0 empfangen wurden. VRRP-Telegramme mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Telegramme ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- Ges. Prio-0-Tel.
  Zeigt an, wie viele VRRP-Telegramme mit der Priorität 0 versendet wurden. Telegramme mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Telegramme ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- Ungültiger Typ
  Zeigt an, wie viele fehlerhafte VRRP-Pakete empfangen wurden, deren Authentifizierungstyp nicht Typ 0 war. Typ 0 bedeutet "keine Authentifizierung".

- **Adresslistenfehler**
  Zeigt an, wie viele fehlerhafte VRRP-Pakete empfangen wurden, deren Adressliste nicht mit der lokal konfigurierten Liste übereinstimmt.
- **Ungültiger Auth.Typ**
  Zeigt an, wie viele fehlerhafte VRRP-Telegramme empfangen wurden, deren Authentifizierungstyp nicht Typ 0 war. Typ 0 bedeutet "keine Authentifizierung".
- **Abweichender Auth.-Typ**
  Zeigt an, wie viele fehlerhafte VRRP-Telegramme empfangen wurden, deren Authentifizierungstyp nicht übereinstimmt.
- **Telegrammlängenfehler**
  Zeigt an, wie viele fehlerhafte VRRP-Telegramme empfangen wurden, deren Länge nicht korrekt ist.

VRRPv3-Statistiken

Einleitung

Diese Seite zeigt die Statistiken des VRRPv3-Protokolls und aller konfigurierten virtuellen Router an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **VRID-Fehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine nicht unterstütze VRID enthalten.
- **Versionsfehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Versionsnummer enthalten.
- **Prüfsummenfehler**
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Prüfsumme enthalten.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstellen**
  Schnittstelle, auf die sich die Einstellungen beziehen.
- **VRID**
  Zeigt die ID des virtuellen Routers an. Gültige Werte sind 1 ... 255.
- **Typ**
  Zeigt die Version des IP-Protokolls an

- **Übergang in Master-Zustand**
  Zeigt an, wie häufig dieser virtuelle Router in den Zustand "Master" ging.
- **Empf. Advertisements**
  Zeigt an, wie viele VRRPv3-Paket empfangen wurden.
- **Advertisement-Intervall-Fehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Intervall nicht mit dem lokal gesetzten Wert übereinstimmt.
- **IP TTL Fehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren TTL (Time to live) Wert im IP Header nicht stimmt.
- **Empf. Prio-0-Tel**
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 empfangen wurden. VRRPv3-Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- **Ges. Prio-0-Tel**
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 versendet wurden. Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- **Ungültiger Typ**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Wert im Feld "Typ" des IP-Headers ungültig ist.
- **Adresslistenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Adressliste nicht mit der lokal konfigurierten Liste übereinstimmt.
- **Telegrammlängenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Länge nicht korrekt ist.

Ringredundanz

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Automatic Redundancy Detection (ARD) ist bei Auslieferung des Geräts voreingestellt. Soll das bisherige High Speed Redundancy Verfahren (HRP) eingesetzt werden, muss HRP konfiguriert werden.

- Rekonfigurationszeit des Telegrammverkehrs nach einer Redundanzumschaltung bei MRP: 200 ms
- Rekonfigurationszeit des Telegrammverkehrs nach einer Redundanzumschaltung bei HRP: 300 ms

Diese Seite informiert Sie über die eingestellten Werte des Gerätes.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Redundanzfunktion**
  Redundanzfunktion Die Spalte "Redundanzfunktion" zeigt die Rolle des Geräts innerhalb des Rings an:
  - Keine Ringredundanz
    Der IE-Switch arbeitet ohne Redundanz-Funktion.
  - HRP-Client
    Der IE-Switch arbeitet als HRP-Client.
  - HRP-Manager
    Der IE-Switch arbeitet als HRP-Manager.
  - MRP-Client
    Der IE-Switch arbeitet als MRP-Client.
  - MRP-Manager
    Der IE-Switch arbeitet als MRP-Manager. Über STEP 7 wurde die Rolle "Manager" für das Gerät eingestellt.
  - MRP Auto-Manager
    Der IE-Switch arbeitet als MRP-Manager. Über WBM bzw. CLI wurde die Rolle "MRP Auto-Manager" oder über STEP 7 die Rolle "Manager (Auto") eingestellt.
- **RM-Status**
  Die Spalte "RM-Status" zeigt an, ob der IE-Switch als Redundanzmanager arbeitet und ob er in dieser Funktion den Ring geöffnet oder durchgeschaltet hat.
  - Passiv
    Der IE-Switch arbeitet als Redundanzmanager und hat den Ring geöffnet, d.h. die an die Ringports angeschlossene Linie von Switches arbeitet fehlerfrei. Der Zustand "Passiv" wird auch angezeigt, wenn der IE-Switch nicht als Redundanzmanager arbeitet (Redundanzmanager deaktiviert).
  - Aktiv
    Der IE-Switch arbeitet als Redundanzmanager und hat den Ring geschlossen, d.h. die an die Ringports angeschlossene Linie von Switches ist unterbrochen (Fehlerfall). Der Redundanzmanager schaltet die Verbindung zwischen seinen Ringports durch und stellt damit wieder eine durchgehende Linientopologie her.
  - "-"
    Die Redundanzfunktion ist deaktiviert.
- **Observer-Status**
  Zeigt den aktuellen Zustand des Observers an.
- **Ring-Port 1 und Ring-Port 2**
  Die Spalten "Ring Port 1" und "Ring Port 2" zeigen die Ports an, die als Ringports verwendet werden.

- **Anz. der Wechsel zum RM-aktiv-Status**
  Zeigt an, wie oft das Gerät als Redundanzmanager in den aktiven Zustand geschaltet hat, d. h. den Ring geschlossen hat.
  Wenn die Redundanzfunktion deaktiviert ist oder das Gerät ist nicht HRP-/MRP-Manager, dann erscheint der Text "Redundanzmanager deaktiviert".
- **Max. Verzögerung der RM-Testtelegramme [ms]**
  Zeigt die maximale Verzögerungszeit für Testtelegramme des Redundanzmanagers an.
  Wenn die Redundanzfunktion deaktiviert ist oder das Gerät ist nicht HRP-/MRP-Manager, dann erscheint der Text "Redundanzmanager deaktiviert".

Standby

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Auf dieser Seite erhalten Sie Informationen zum Status des Geräts bezogen auf Standby-Redundanz. Die Textfelder dieser Seite sind nur lesbar.

Hinweis

Gerät mit höherer MAC-Adresse wird Master

Für die redundante Kopplung von HRP-Ringen werden immer zwei Geräte als Master-/Slave-Gerätepaar konfiguriert. Dies gilt auch für unterbrochene HRP-Ringe = Linien. Im fehlerfreien Zustand übernimmt immer das Gerät mit der höheren MAC-Adresse die Funktion des Masters. Wichtig ist diese Art der Zuordnung insbesondere bei einem Gerätetausch. Abhängig von den MAC-Adressen kann das bisherige Gerät mit Slave-Funktion die Rolle des Standby-Masters übernehmen.

Unter dem Reiter Standby finden Sie den Status der Standby-Funktion:

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Standby-Ports**
  Zeigt den Standby-Port an.
- **Standby-Name**
  Name der Standby-Verbindung.

- **Standby-Funktion**
  Zeigt an, ob die Standby-Verbindung aktiviert oder deaktiviert ist.
  - Master
    Das Gerät hat Verbindung zum Partnergerät und arbeitet als Master. Im fehlerfreien Betrieb ist bei diesem Gerät der Standby-Port aktiv.
  - Slave
    Das Gerät hat Verbindung zum Partnergerät und arbeitet als Slave. Im fehlerfreien Betrieb ist bei diesem Gerät der Standby-Port inaktiv.
  - Deaktiviert
    Die Standby-Kopplung ist deaktiviert. Das Gerät arbeitet weder als Master noch als Slave. Ein als Standby-Port konfigurierter Port arbeitet als normaler Port ohne Standby-Funktion.
  - Verbindungsaufbau
    Es wurde noch keine Verbindung zum Partnergerät aufgenomnen. Der Standby-Port ist inaktiv. In diesem Fall ist entweder die Projektierung auf dem Partnergerät nicht konsistent (z. B. falscher Verbindungsname, Standby-Kopplung deaktiviert) oder es liegt ein physikalischer Fehler vor (z. B. Geräteausfall, Link-Down).
  - Verbindungsverlust
    Die bestehende Verbindung zum Partnergerät wurde verloren. In diesem Fall wurde entweder die Projektierung auf dem Partnergerät geändert (z.B. anderer Verbindungsname, Standby-Kopplung deaktiviert) oder es liegt ein physikalischer Fehler vor (z.B. Geräteausfall, Link-Down).
- **Standby-Status**
  Zeigt den Status des Standby-Ports an:
  - Aktiv
    Der Standby-Port dieses Geräts ist aktiv, d. h. für den Telegrammverkehr freigeschaltet.
  - Passiv
    Der Standby-Port dieses Geräts ist inaktiv, d. h. für den Telegrammverkehr gesperrt.
  - "-"
    Die Standby-Funktion ist deaktiviert.
- **Anz. der Wechsel zum Standby-aktiv-Status**
  Zeigt an, wie oft der IE-Switch den Standby-Status vom Zustand "Passiv" in den Zustand "Aktiv" geändert hat. Wenn die Verbindung eines Standby-Ports beim Standby- Master ausfällt, wechselt der IE-Switch in den Zustand "Aktiv". Wenn die Standby-Funktion deaktiviert ist, erscheint der Text "Standby deaktiviert" in diesem Feld.
  Nach jedem Neustart des Geräts werden die Zähler automatisch zurückgesetzt.

Link Check

Überwachung optischer Verbindungen im Ring

Die Seite zeigt die folgende Informationen zu Link Check an:

- Die Ports, auf denen Sie Link Check aktivieren können
- Den aktuellen Status
- Die Statistik gesendeter und empfangener Link Check-Telegramme der überwachten Verbindungen

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Port**
  Zeigt den Port an, auf den sich die nachfolgenden Informationen beziehen. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Link Check**
  Zeigt an, ob die Funktion Link Check aktiviert oder deaktiviert ist.
- **Betriebszustand**
  Zeigt den Status der Funktion Link Check. Es gibt folgende Zustände:
  - Deaktiviert
    Die Funktion ist deaktiviert.
  - Aktiviert
    Die Funktion ist aktiviert. Der Verbindungspartner hat die Überwachung noch nicht bestätigt.
  - Running
    Die Funktion ist aktiviert. Die Verbindungsüberwachung ist aktiv. Die ausgehenden und eingehenden Testtelegramme werden gezählt und abgeglichen.
  - Fehler
    Die Funktion ist aktiviert. Link Check hat einen Fehler auf der überwachten Strecke detektiert und den Port abgeschaltet.
- **Empfangene Frames**
  Zeigt an, wie viele Link Check-Testtelegramme empfangen wurden.
- **Gesendete Frames**
  Zeigt an, wie viele Link Check-Testtelegramme gesendet wurden.

MRP-Interconnection

Redundante Kopplung von Ringen

Die Seite zeigt die konfigurierten MRP-Interconnect-Verbindungen und Detailinformationen zu den einzelnen Parametern an.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Interconnection-ID**
  Die ID der MRP-Interconnection-Verbindung.
- **Interconnection-Name**
  Der Name der MRP-Interconnection-Verbindung.
- **Interconnection-Port**
  Der Port, der für die MRP-Interconnection-Verbindung genutzt wird.
- **Port-Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Operative Rolle/Position**
  Zeigt die Rolle des Geräts an. Bei der Rolle "Client" wird außerdem die Position des Clients angezeigt. Es gibt folgende Möglichkeiten:
  - Deaktiviert
  - Manager
  - Primary Client
  - Secondary Client
- **Verbindungsstatus**
  Der Status der MRP-Interconnect-Domäne. Es gibt folgende Möglichkeiten:
  - Deaktiviert
  - Nicht definiert
  - Interconnection Offen
    Die Verbindung zwischen den primären Geräten ist geöffnet, die Ringe sind über die sekundäre Verbindung verbunden.
  - Interconnection Geschlossen
    Es besteht eine Verbindung zwischen den primären Geräten.
- **Open Zähler**
  Zeigt an, wie oft der Zustand "Interconnection Offen" nach dem letzten Zurücksetzen der Zähler eingetreten ist.
- **Open Zeit**
  Die Zeit seit der letzten Änderung des Verbindungsstatus.
- **Zähler zurücksetzen**
  Klicken Sie auf "Zähler zurücksetzen", um den Zähler zurückzusetzen. Der Zähler wird durch einen Neustart zurückgesetzt.

Ethernet-Statistiken

Schnittstellenstatistik

Informationen aus der MIB

Die Seite zeigt die Statistik aus der Schnittstellentabelle der Management Information Base (MIB).

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Byte empfangen**
  Zeigt die Anzahl der empfangenen Bytes an.
- **Byte gesendet**
  Zeigt die Anzahl der gesendeten Bytes an.
- **Unicast empfangen**
  Zeigt die Anzahl der empfangenen Unicast-Telegramme an.
- **Nicht-Unicast empfangen**
  Zeigt die Anzahl der empfangenen Telegramme an, die nicht vom Telegrammtyp Unicast sind.
- **Unicast gesendet**
  Zeigt die Anzahl der gesendeten Unicast-Telegramme an.
- **Nicht-Unicast gesendet**
  Zeigt die Anzahl der gesendeten Telegramme an, die nicht vom Telegrammtyp Unicast sind.
- **Fehler empfangen**
  Zeigt die Anzahl aller möglichen RX-Fehler an, siehe Register "Telegrammfehler".

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammlänge

Telegramme sortiert nach Länge

Diese Seite zeigt, wie viele Telegramme mit welcher Länge an jedem Port gesendet und empfangen wurden. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die angezeigten Werte werden durch RMON übermittelt.

Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link-Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.

  Hinweis
  Anzeige der Telegrammstatistik

  Beachten Sie bei der Statistik der Telegrammlängen, dass sowohl eingehende als auch ausgehende Telegramme gezählt werden.

- **Telegrammlängen**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der Telegramme entsprechend ihrer Telegrammlänge.
  Dabei wird in folgenden Telegrammlängen unterschieden:
  - 64 Byte
  - 65 - 127 Byte
  - 128 - 255 Byte
  - 256 - 511 Byte
  - 512 - 1023 Byte
  - 1024 - max

Hinweis
Datenverkehr auf geblockten Ports

Aus technischen Gründen können auf geblockten Ports Datenpakete angezeigt werden.

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammtyp

Empfangene Telegramme sortiert nach Telegrammtyp

Diese Seite zeigt, wie viele Telegramme des Typs "Unicast", "Multicast" und "Broadcast" an jedem Port empfangen wurden.

Sie können auf dieser Seite keine Konfigurationen vornehmen.

Die angezeigten Werte werden durch RMON übermittelt. Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link-Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Unicast / Multicast / Broadcast**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der eingegangenen Telegramme entsprechend der Telegrammtypen "Unicast", "Multicast" und "Broadcast".

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammfehler

Fehlerhaft empfangene Telegramme

Die Seite zeigt, wie viele fehlerhafte Telegramme pro Port empfangen wurden.

Sie können auf dieser Seite keine Konfigurationen vornehmen.

Die angezeigten Werte werden durch RMON übermittelt. Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an.
- **Fehlertypen**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der eingegangenen Telegramme entsprechend ihres Fehlertyps.
  Dabei wird in den Spalten der Tabelle nach folgenden Fehlertypen unterschieden:
  - CRC
    Pakete, deren Inhalt nicht mit der zugehörigen CRC-Prüfsumme übereinstimmt.
  - Zu kurz
    Pakete mit einer Länge kleiner als 64 Bytes.
  - Zu lang
    Pakete, die verworfen wurden, weil sie zu lang sind.
  - Fragmente
    Pakete mit einer Länge kleiner als 64 Bytes und einer falschen CRC-Prüfsumme.
  - Jabbers
    VLAN-getaggte Pakete mit einer falschen CRC Prüfsumme, die verworfen wurden, weil sie zu lang sind.
  - Kollisionen
    Erkannte Kollisionen.

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

History

Stichproben der Statistiken

Die Seite zeigt pro Port Stichproben mit Informationen aus den Statistiken an.

Auf der Seite "Layer 2 > RMON > History" können Sie einstellen, für welche Ports Stichproben genommen werden sollen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Einstellungen

- **Port**
  Wählen Sie den Port, für den die History angezeigt werden soll.

Beschreibung der angezeigten Werte

- **Einträge**
  Maximale Anzahl der Stichproben, die gleichzeitig gespeichert werden.
- **Intervall [s]**
  Intervall, nach dem der aktuelle Stand der Statistik als Stichprobe gespeichert wird.

Die Tabelle gliedert sich in folgende Spalten:

- **Stichprobe**
  Nummer der Stichprobe
- **Zeitpunkt der Stichprobe**
  Systembetriebszeit, zu der die Stichprobe genommen wurde.
- **Unicast**
  Anzahl der empfangenen Unicast-Telegramme.
- **Multicast**
  Anzahl der empfangenen Multicast-Telegramme.
- **Broadcast**
  Anzahl der empfangenen Broadcast-Telegramme.
- **CRC**
  Anzahl von Telegrammen mit fehlerhafter CRC-Prüfsumme.
- **Zu kurz**
  Anzahl der Telegramme, die kleiner als 64 Bytes sind.
- **Zu lang**
  Anzahl der Telegramme, die verworfen werden, weil sie zu groß sind.
- **Fragmente**
  Anzahl der Telegramme, die kleiner als 64 Bytes sind und eine fehlerhafte CRC-Prüfsumme haben.
- **Jabbers**
  Anzahl der Telegramme mit VLAN-Tag, die eine fehlerhafte CRC-Prüfsumme haben und verworfen werden, weil sie zu groß sind.

- **Kollisionen**
  Anzahl der Kollisionen empfangener Telegramme.
- **Auslastung**
  Auslastung des Ports während einer Stichprobe.

Unicast

Diese Seite zeigt den aktuellen Inhalt der Unicast-Filtertabelle. In dieser Tabelle sind die Quelladressen von Unicast-Adresstelegrammen aufgeführt. Einträge können entweder dynamisch erfolgen, wenn ein Teilnehmer ein Telegramm an einen Port sendet oder statisch durch Parametrierung seitens des Anwenders.

Beschreibung der angezeigten Werte

- **VLAN-ID**
  Zeigt die VLAN-ID, die dieser MAC-Adresse zugeordnet ist.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Teilnehmers, die das Gerät gelernt hat oder die der Anwender projektiert hat.
- **Status**
  Zeigt den Status jedes Adresseintrags:
  - Learnt
    Die angegebene Adresse wurde durch Empfang eines Telegramms dieses Teilnehmers gelernt und wird nach Ablauf der Aging Time wieder gelöscht, sollten keine weiteren Pakete dieses Teilnehmers empfangen werden.

    Hinweis

    Bei einem Link-Down werden gelernte MAC-Einträge gelöscht.
  - Static
    Vom Anwender projektiert. Statische Adressen sind permanent gespeichert, d.h. sie werden nach Ablauf der Aging Time oder beim Neustart des Switchs nicht gelöscht.
- **Port**
  Zeigt an, über welchen Port der Teilnehmer mit der angegebenen Adresse erreichbar ist. Vom Gerät empfangene Telegramme, deren Zieladresse mit dieser Adresse übereinstimmt, werden an diesen Port weitergegeben.

Multicast

Multicast

Diese Tabelle zeigt die aktuell in der Filtertabelle eingetragenen Multicast-Telegramme mit ihren Zielports. Die Einträge können dynamisch (das Gerät hat sie gelernt) oder statisch (der Anwender hat sie parametriert) erfolgt sein.

Beschreibung der angezeigten Werte

- VLAN-ID
  Zeigt VLAN-ID des VLANs an, dem die MAC-Multicast-Adresse zugeordnet ist.
- MAC-Adresse
  Zeigt MAC-Multicast-Adresse an, die das Gerät gelernt hat oder die der Anwender projektiert hat.
- Status
  Zeigt den Status jedes Adress-Eintrags. Dabei sind folgende Angaben möglich:
  - Statisch
    Die Adresse wurde vom Anwender statisch eingetragen. Statische Adressen sind permanent gespeichert, d.h. sie werden nicht nach Ablauf der Aging Time oder beim Neustart des Gerätes gelöscht. Sie müssen vom Anwender gelöscht werden.
  - IGMP
    Der Zielport für diese Adresse wurde über IGMP-Konfiguration ermittelt.
  - MLD
    Der Zielport für diese Adresse wurde über MLD-Konfiguration ermittelt.
  - GMRP
    Der Zielport für diese Adresse wurde über ein empfangenes GMRP-Telegramm registriert.
- Liste der Ports
  Für jeden Steckplatz gibt es eine Spalte. Innerhalb einer Spalte wird für jeden Port die Zugehörigkeit zur Multicast-Gruppe angezeigt:
  - M (Member)
    Über diesen Port werden Multicast-Telegramme gesendet.
  - R (Registered)
    Mitglied der Multicast-Gruppe, die Registrierung erfolgte über ein GMRPTelegramm.
  - I (IGMP/MLD)
    Mitglied der Multicast-Gruppe, die Registrierung erfolgte über ein IGMP/MLD-Telegramm.
  - –
    Kein Mitglied der Multicast-Gruppe. Über diesen Port werden keine Multicast-Telegramme mit der definierten Multicast-MAC-Adresse gesendet.
  - F (Forbidden)
    Kein Mitglied der Multicast-Gruppe. Außerdem darf diese Adresse nicht dynamisch über GMRP oder IGMP/MLD gelernt werden.

IGMP-Gruppen

Status der Multicast-Filtertabelle

Diese Tabelle zeigt die aktuell in der Filtertabelle eingetragenen konfigurierten und gejointen Multicastgruppen.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die IGMP-Schnittstelle an.
- **Gruppenadresse**
  Zeigt die Adresse der Multicastgruppe an, die das Gerät gelernt hat oder die der Anwender projektiert hat.
- **Modus**
  Zeigt an, ob der Empfänger angegeben hat, dass ein Multicast von einer bestimmten Quelle empfangen bzw. abgelehnt werden soll.
  - '-'
    Es wird IGMPv1 oder IGMPv2 verwendet oder der Empfänger hat keine Quellen bevorzugt bzw. ausgeschlossen.
  - Include
    Der Multicast soll von bestimmten Quellen empfangen werden.
  - Exclude
    Der Multicast soll von bestimmten Quellen nicht empfangen werden.
- **Status**
  Zeigt den Status jedes Eintrags. Dabei sind folgende Angaben möglich:
  - Statisch
    Die Multicastgruppe wurde statisch konfiguriert.
  - Dynamisch
    Die Multicastgruppe wurde über IGMP-Konfiguration gelernt.

IGMP-Gruppen

Status der Multicast-Filtertabelle

Diese Tabelle zeigt die aktuell in der Filtertabelle eingetragenen konfigurierten und gejointen Multicastgruppen.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die IGMP-Schnittstelle an.
- **Gruppenadresse**
  Zeigt die Adresse der Multicastgruppe an, die das Gerät gelernt hat oder die der Anwender projektiert hat.

- **Modus**
  Zeigt an, ob der Empfänger angegeben hat, dass ein Multicast von einer bestimmten Quelle empfangen bzw. abgelehnt werden soll.
  - '-'
    Es wird IGMPv1 oder IGMPv2 verwendet oder der Empfänger hat keine Quellen bevorzugt bzw. ausgeschlossen.
  - Include
    Der Multicast soll von bestimmten Quellen empfangen werden.
  - Exclude
    Der Multicast soll von bestimmten Quellen nicht empfangen werden.
- **Status**
  Zeigt den Status jedes Eintrags. Dabei sind folgende Angaben möglich:
  - Statisch
    Die Multicastgruppe wurde statisch konfiguriert.
  - Dynamisch
    Die Multicastgruppe wurde über IGMP-Konfiguration gelernt.

LLDP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Status der Nachbarschaftstabelle

Diese Seite zeigt den aktuellen Inhalt der Nachbarschaftstabelle. In dieser Tabelle sind die Informationen gespeichert, die der LLDP-Agent von angeschlossenen Geräten empfangen hat. Über welche Schnittstellen der LLDP-Agent Informationen empfängt bzw. versendet, legen Sie in folgendem Kapitel fest: "Layer 2 > LLDP".

Beschreibung der angezeigten Werte

Die Tabelle enthält die folgenden Spalten:

- **Systemname**
  Systemname des angeschlossenen Geräts.
- **Geräte-ID**
  Gerätekennung des angeschlossenen Geräts. Die Geräte-ID entspricht dem Gerätenamen, der über PST (STEP 7) vergeben wird. Wenn kein Gerätename vergeben ist, wird die MAC-Adresse des Geräts angezeigt.
- **Lokale Schnittstelle**
  Der Port, an dem das Gerät die Informationen empfangen hat.

- **Speicherzeit**
  Ein Eintrag bleibt für die hier angegebene Zeit im Gerät gespeichert. Wenn das Gerät in dieser Zeit keine neuen Informationen von dem angeschlossenen Gerät erhält, wird der Eintrag gelöscht.
- **Eigenschaft**
  Zeigt die Eigenschaften des angeschlossenen Geräts an:
  - Router
  - Bridge
  - Telephone
  - DOCSIS Cable Device
  - WLAN Access Point
  - Repeater
  - Station
  - Other
- **Port-ID**
  Port des Geräts, der mit dem das Gerät verbunden ist.

FMP

Überwachung optischer Strecken

Mit Fiber Monitoring können Sie optische Strecken überwachen. Die Tabelle zeigt den aktuellen Zustand der Ports.

Welche Werte überwacht werden, stellen Sie auf folgender Seite ein: "Layer 2 > FMP".

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

- **Port**
  Zeigt die verfügbaren optischen Ports an.
- **Status der Rx-Leistung**
  - disabled
    Fiber Monitoring ist deaktiviert
  - ok
    Der Wert für die Empfangsleistung der optischen Strecke liegt innerhalb der eingestellten Grenzen
  - maint. req.
    Die Strecke sollte überprüft werden.
    Es wird einq Warnung gemeldet.
  - maint. dem.
    Die Strecke muss überprüft werden.
    Es wird ein Alarm gemeldet und die Fehler-LED leuchtet.
  - link down
  - Die Verbindung ist unterbrochen.

- **Rx-Leistung**
  Zeigt den aktuellen Wert der Empfangsleistung an. Der Wert kann eine Toleranz von +/- 3 dB haben.
  Wenn keine Verbindung (Link down) besteht oder Fiber Monitoring deaktiviert ist, wird "-" angezeigt. Wenn bei dem Partner-Port Fiber Monitoring nicht aktiviert ist, wird der Wert 0.0 angezeigt..
- **Status des Leistungsabfalls**
  Um den Leistungsabfall der Verbindung überwachen zu können, braucht der Port eine Verbindung zu einem anderen Port mit aktiviertem FM.
  - disabled
    FM ist deaktiviert.
  - ok
    Der Wert für den Leistungsabfall der optischen Strecke ist in Ordnung.
  - maint. req.
    Die Strecke sollte überprüft werden.
    Es wird eine Warnung gemeldet.
  - maint. dem.
    Die Strecke muss überprüft werden.
    Es wird ein Alarm gemeldet und die Fehler-LED leuchtet.
  - idle
    Der Port hat keine Verbindung zu einem anderen Port mit aktiviertem Fiber Monitoring. Wenn 5 Zyklen lang keine Diagnoseinformationen vom optischen Port des Verbindungspartners empfangen wurden, gilt die Fiber Monitoring-Verbindung als unterbrochen. Ein Zyklus dauert 5 Sekunden.
- **Leistungsabfall [dB]**
  Zeigt den aktuellen Wert des Leistungsabfalls an. Der Wert kann eine Toleranz von +/- 3 dB haben.
  Wenn keine Verbindung (Link down) besteht, Fiber Monitoring deaktiviert ist oder der Partner-Port Fiber Monitoring nicht unterstützt, wird "-" angezeigt..

IPv4-Routing

Routing-Tabelle

Übersicht

Diese Seite zeigt die Routen an, die aktuell verwendet werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Ziel-Adresse dieser Route an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske dieser Route an.

- **Gateway**
  Zeigt das Gateway für diese Route an.
- **Schnittstelle**
  Zeigt die Schnittstelle für diese Route an.
- **Metrik**
  Zeigt die Metrik der Route an. Je größer der Wert, desto länger benötigen die Pakete zu Ihrem Ziel.
- **Routing-Protokoll**
  Zeigt an, aus welchem Routing-Protokoll der Eintrag der Routingtabelle stammt. Folgende Einträge sind möglich:
  - Verbunden: Verbundene Routen
  - Statisch: Statische Routen
  - RIP: Routen über RIP
  - OSPF: Routen über OSPF
  - Sonstige: Sonstige Routen

OSPFv2-Schnittstellen

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Konfiguration der OSPF-Schnittstelle an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt IPv4-Adresse der OSPF-Schnittstelle an
- **Area ID**
  Zeigt die Area-ID an, zu der die OSPF-Schnittstelle gehört.

- **Status der Schnittstelle**
  Zeigt den Status der Schnittstelle an:
  - Down
    Die Schnittstelle ist nicht verfügbar.
  - Loopback
    Loopback-Schnittstelle
  - Waiting
    Hochfahren und Aushandeln der Schnittstelle.
  - Point to Point
    Punkt-zu-Punkt-Verbindung
  - Designated Router
    Der Router ist ein Designated Router und erzeugt Network LSA.
  - Backup D. Router
    Der Router ist der Backup-Router für den Designated Router.
  - Other D. Router
    Die Schnittstelle ist hochgefahren. Der Router ist weder ein Designated Router noch ein Backup Designated Router.
- **OSPF-Status**
  Zeigt an, den Status von OSPF an.
  - Enabled: OSPF ist an der Schnittstelle aktiviert.
  - Disabled: OSPF ist an der Schnittstelle deaktiviert.
- **Designated Router**
  Zeigt die IPv4-Adresse des designated Routers für diese OSPF-Schnittstelle an.
- **Backup Designated Router**
  Zeigt die IPv4-Adresse des designated Backup Routers für diese OSPF-Schnittstelle an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen von OSPF an.

OSPFv2-Nachbarn

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die dynamisch erfassten Nachbar-Router an den jeweiligen Netzwerken an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv4-Adresse des Nachbar-Routers auf diesem Netzwerk an.
- **Router-ID**
  Zeigt die ID des Nachbar-Routers an. Beide Adressen können übereinstimmen.
- **Status**
  Zeigt den Status des Nachbar-Routers an. Der Status kann folgende Werte annehmen:
  - unknown
    Status des Nachbar-Routers unbekannt.
  - down
    Der Nachbar-Router ist nicht erreichbar.
  - attempt und init
    Kurzlebiger Status während der Initialisierung
  - two-way
    Beiderseitiger Empfang von Hello-Paketen. Festlegen des designierten Routers und des designierten Backup-Routers.
  - exchangestart, exchange und loading
    Status während des Austausches der LSAs
  - full
    Die Datenbank ist komplett und innerhalb des Bereichs synchron. Die Routen können nun ermittelt werden.

    Hinweis

    Normalzustand

    Wenn der Partner-Router ein designierter Router oder ein designierter Backup Router ist, ist der Status "full". Andernfalls ist der Status "two-way".
- **Assoc. Area Type**
  Zeigt die Bereichsart an, über die die Nachbarschaftsbeziehung besteht. Es gibt folgende Bereichsarten:
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Priorität**
  Zeigt die Priorität des Nachbar-Routers an. Diese ist nur während der Auswahl des designierten Routers auf einem Netzwerk von Bedeutung. Bei virtuellen Nachbar-Routern ist diese Angabe nicht relevant.
- **Unterdr. Hello-Pakete**
  Zeigt die unterdrückten Hello-Pakete zum Nachbar-Router an. Dieses Feld steht normalerweise auf "nein".

- **Hello-Queue**
  Zeigt die Länge der Warteschlange mit den noch zu übertragenden Hello-Paketen an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen an.

OSPFv2 Virtuelle Nachbarn

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die konfigurierten virtuellen Nachbar-Router an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv4-Adresse des virtuellen Nachbar-Routers auf diesem Netzwerk an.
- **Router-ID**
  Zeigt die Router-ID des virtuellen Nachbar-Routers an.
- **Status**
  Zeigt den Status des Nachbar-Routers an. Der Status kann folgende Werte annehmen:
  - unknown
    Status des Nachbar-Routers unbekannt.
  - down
    Der Nachbar-Router ist nicht erreichbar.
  - attempt und init
    Kurzlebiger Status während der Initialisierung
  - two-way
    Beiderseitiger Empfang von Hello-Paketen. Festlegen des designierten Routers und des designierten Backup-Routers.
  - exchangestart, exchange und loading
    Status während des Austausches der LSAs
  - full
    Die Datenbank ist komplett und innerhalb des Bereichs synchron. Die Routen können nun ermittelt werden.

    Hinweis

    Normalzustand

    Wenn der Partner-Router ein designierter Router oder ein designierter Backup Router ist, ist der Status "full". Andernfalls ist der Status "two-way".

- **Trans. Area-ID**
  Zeigt die ID des Bereichs an, über die die virtuelle Nachbarschaftsbeziehung besteht.
- **Unterdr. Hello-Pakete**
  Zeigt an, ob es unterdrückte Hello-Pakete zum virtuellen Nachbar-Router gibt.
  – nein: Keine unterdrückten Hello-Pakete sind vorhanden (default)
  – ja: Unterdrückte Hello-Pakete sind vorhanden.
- **Hello-Queue**
  Zeigt die Länge der Warteschlange mit den noch zu übertragenden Hello-Paketen an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen an.

OSPFv2 LSDB

Übersicht .

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Link State Database ist die zentrale Datenbank zur Verwaltung aller Verbindungen innerhalb eines Bereichs. Sie besteht aus den so genannten Link State Advertisements (LSAs). Die wichtigsten Daten dieser LSAs werden auf dieser WBM-Seite angezeigt.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Area ID**
  Zeigt die ID des Bereichs an, zu die LSA gehört. Wenn der LSA eine externe Verbindung ist, wird ein '-' angezeigt.
- **LSA-Typ**
  Zeigt den LSA-Typ an. Folgende Werte sind möglich:
  - unknown
    LSA-Typ nicht bekannt.
  - Router
    Die Router LSA (Typ 1) wird vom OSPF-Router innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über den Status aller Routerschnittstellen.
  - Network
    Die Network LSA (Typ 2) wird vom Designated Router innerhalb eines Bereichs gesendet. Die LSA enthält eine Liste mit Routern, die mit dem Netzwerk verbunden sind.
  - NSSA External
    Die NSSA External LSA (Typ 7) wird vom NSSA-ASBR innerhalb einer NSSA gesendet. Der NSSA-ASBR empfängt LSAs vom Typ 5 und konvertiert die Information in LSAs vom Typ 7. Diese LSAs kann der NSSA-Router innerhalb einer NSSA weiter senden.
  - Summary
    Die Summary LSA (Typ 3) wird vom ABR innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über Routen zu anderen Netzwerken.
  - AS Summary
    Die AS Summary LSA (Typ 4) wird vom Area Border Router innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über Routen zu anderen autonomen Systemen.
  - AS External
    Die AS External LSA (Typ 5) wird vom AS Border Router innerhalb einer autonomen Systems gesendet. Die LSA enthält Informationen über Routen von einem Netzwerk in ein anderes Netzwerk.
- **Link State ID**
  Zeigt die ID der LSA an.
- **Router ID**
  Zeigt die ID des Routers an, der diese LSA gesendet hat.
- **Sequenznummer**
  Zeigt die Sequenznummer der LSA. Mit jeder Erneuerung eines LSAs wird diese Sequenznummer um eins hoch gezählt.

RIPv2-Statistik

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Statistik der RIP-Schnittstelle an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt IP-Adresse der RIPv2-Schnittstelle an
- **Ignorierte Telegramme**
  Anzahl der empfangen RIP-Telegramme, die gelöscht und deswegen nicht beachtet wurden.
- **Ignorierte Routen**
  Anzahl der Routen gültiger RIP-Telegramme, die nicht berücksichtigt werden konnten.
- **Gesendete Updates**
  Zeigt an, wie oft der Router seine Routing-Tabelle an seine Nachbar-Router gesendet hat.

NAT-Übersetzungen

Übersicht

Diese Seite zeigt die aktiven NAT-Verbindungen an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die IP-Schnittstelle an.
- **Inside Local-Adresse**
  Zeigt die tatsächliche Adresse des Geräts an, das von extern erreichbar sein soll.
- **Inside Local-Port**
  Zeigt den Port an, der der Inside Local-Adresse zugeordnet ist.
- **Inside Global-Adresse**
  Zeigt die Adresse an, unter der das Gerät von extern erreichbar ist.
- **Inside Global-Port**
  Zeigt den Port an, der der Inside Global-Adresse zugeordnet ist.

- **Outside Local/Global-Adresse**
  Zeigt die Adresse des Kommunikationspartners an.
- **Outside Local/Global-Port**
  Zeigt den Port des externen Kommunikationspartner an.
- **Letzte Verwendung[s]**
  Zeigt den Zeitpunkt an, zu dem das letzte Paket übertragen wurde.

PIM-Schnittstellen

Übersicht

Diese Seite zeigt die PIM-Schnittstellen an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die PIM-Schnittstelle an.
- **Adresse**
  Zeigt die IP-Adresse der Schnittstelle an.
- **Query-Intervall**
  Jeder PIM-Router bzw. jede PIM-Schnittstelle sendet in dem angegebenen Intervall zyklisch Hello-Pakete. Somit lernt jeder PIM-Router seine Nachbarn kennen und der Designated Router kann festgelegt werden.
- **DR-Adresse**
  Zeigt die IP-Adresse des Designated Routers an.
- **DR-Priorität**
  Zeigt die DR-Priorität der Schnittstelle an.

PIM-Nachbarn

Übersicht

Diese Seite zeigt die die PIM-Nachbarschaftstabelle an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die PIM-Schnittstelle an. Über diese Schnittstelle sind die PIM-Router miteinander verbunden.
- **Nachbaradresse**
  Zeigt die IP-Adresse des Nachbarn an.
- **DR-Priorität**
  Zeigt die DR-Priorität der Schnittstelle an.

PIM-Routen

Übersicht

Diese Seite zeigt die PIM-Routingtabelle an.

Beschreibung der angezeigten Werte

Wählen Sie aus, welche PIM-Routen angezeigt werden sollen:

- **Alle**
- **Ausgehend**

Abhängig von Ihrer Auswahl werden in der Tabelle nicht alle Spalten anzeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Adresse der Multicastgruppe**
  Zeigt die Adresse der Multicastgruppe an.
- **Maske der Multicastgruppe**
  Zeigt die Subnetzmaske an, die das Multicastband einschränkt.
- **Quell-Adresse**
  - IP-Adresse
    Zeigt die Quell-IP-Adresse des Multicastpakets an.
  - *
    (*,G)-Meldungen beinhalten die Gruppe und die Information, die Meldung in Richtung RP weiterzureichen.
    Die Route wird über den RP geführt.
- **Eingehende Schnittstellen**
  Zeigt an, über welche Schnittstelle der Multicast empfangen wird.

- **Ausgehende Schnittstellen**
  Zeigt an, über welche Schnittstelle der Multicast weitergeleitet wird.
- **Status**
  Zeigt an, ob eine Multicastgruppe verwendet wird.
  - Forwarding
    Die Multicastgruppe wird verwendet.
  - Pruned
    Die Multicastgruppe wird nicht verwendet.

PIM RPs

Stichproben der Statistiken

Diese Seite zeigt Informationen zu Rendezvous Punkten an.

Beschreibung der angezeigten Werte

Wählen Sie aus, welche Rendezvous Punkte angezeigt werden sollen:

- **Kandidat**
  Zeigt die Schnittstellen des Geräts an, die Kandidaten für Rendezvous Punkte sind.
- **Gewählt** (Designated)
  Zeigt die Designated Router des Geräts an.
- **Statisch**
  Zeigt die statischen Rendezvous Punkte des Geräts an.

Abhängig von Ihrer Auswahl werden in der Tabelle nicht alle Spalten anzeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Adresse der Multicastgruppe**
  Zeigt die Adresse der Multicastgruppe an.
- **Maske der Multicastgruppe**
  Zeigt die Subnetzmaske an, die das Multicastband einschränkt.
- **RP-Adresse**
  Zeigt die IP-Adresse des RPs an.
- **RP-Priorität**
  Zeigt die Priorität der Schnittstelle als RP an.
- **Bidirektionaler Multicast**
  Zeigt an, ob die Funktionalität "Bidirektionaler Multicast" aktiviert oder deaktiviert ist.

PIM BSRs

Übersicht

Diese Seite zeigt Informationen zu Bootstrap-Routern (BSR) an.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Adresse des gewählten BSRs**
  Zeigt die IP-Adresse des BSRs, der innerhalb eines PIM-Netzwerks bzw. einer PIM-Domäne gewählt wurde.

Die Tabelle gliedert sich in folgende Spalten:

- **Adresse des BSR-Kandidaten**
  Zeigt die Schnittstellen des Geräts an, die Kandidaten für den BSR auf dem PIM-Router sind.
- **Priorität des BSR-Kandidaten**
  Zeigt die Priorität des BSRs an.

MSDP-Cache

Übersicht

Diese Seite zeigt die MSDP-Partner an.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Quelle**
  Zeigt die Quelle des Multicasts an.
- **Gruppe**
  Zeigt die Multicastgruppe an.
- **RP**
  Zeigt den MSDP-Partner an.
- **Betriebszeit**
  Zeigt die verbleibende Zeit an, nach der ein Eintrag aus dem SA-Cache gelöscht wird, wenn keine SA-Cache-Updates empfangen werden.

IPv6-Routing

Routing-Tabelle

Übersicht

Diese Seite zeigt die IPv6-Routen an, die aktuell verwendet werden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Ziel-Adresse dieser Route an.
- **Präfixlänge**
  Zeigt die Präfixlänge dieser Route an.
- **Gateway**
  Zeigt das Gateway für diese Route an.
- **Schnittstelle**
  Zeigt die Schnittstelle für diese Route an.
- **Metrik**
  Zeigt die Metrik der Route an. Je größer der Wert, desto länger benötigen Pakete zu Ihrem Ziel.
- **Routing-Protokoll**
  Zeigt an, aus welchem Routing-Protokoll der Eintrag der Routingtabelle stammt. Folgende Einträge sind möglich:
  - Verbunden: Verbundene Routen
  - Statisch: Statische Routen
  - RIPng: Routen über RIPng
  - OSPFv3: Routen über OSPFv3
  - Sonstige: Sonstige Routen

OSPFv3-Schnittstellen

Übersicht

Diese Seite zeigt die Konfiguration der OSPFv3-Schnittstelle an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, auf die sich die Einstellungen bezieht.
- **Area-ID**
  Zeigt die Area-ID an, zu der die OSPF-Schnittstelle gehört.

- **Status der Schnittstelle**
  Zeigt den Status der Schnittstelle an:
  - Down
    Die Schnittstelle ist nicht verfügbar.
  - Loop back
    Loop back-Schnittstelle
  - Waiting
    Hochfahren und Aushandeln der Schnittstelle.
  - Point to Point
    Punkt-zu-Punkt-Verbindung
  - Designated Router
    Der Router ist ein designierter Router und erzeugt Network LSA.
  - Backup D. Router
    Der Router ist der Backup-Router für den designierten Router.
  - S D. Router
    Die Schnittstelle ist hochgefahren. Der Router ist weder ein designierter noch designierter Backup-Router.
- **OSPF-Status**
  Zeigt den Status von OSPF an.
  - Enabled: OSPF ist an der Schnittstelle aktiviert.
  - Disabled: OSPF ist an der Schnittstelle deaktiviert.
- **Designated Router**
  Zeigt die IPv6-Adresse des designierten Routers für diese OSPFv3-Schnittstelle an.
- **Backup Designated Router**
  Zeigt die IPv6-Adresse des designierten Backup-Routers für diese OSPFv3-Schnittstelle an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen von OSPFv3 an.

OSPFv3-Nachbarn

Übersicht

Diese Seite zeigt die dynamisch erfassten Nachbar-Router an den jeweiligen Netzwerken an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die OSPFv3-Schnittstelle an, über die der Nachbar-Router erreichbar ist.
- **Router-ID**
  Zeigt die ID des Nachbar-Routers an.
- **Status**
  Zeigt den Status des Nachbar-Routers an. Der Status kann folgende Werte annehmen:
  - unknown
    Status des Nachbar-Routers unbekannt.
  - down
    Der Nachbar-Router ist nicht erreichbar.
  - attempt und init
    Kurzlebiger Status während der Initialisierung
  - two-way
    Beiderseitiger Empfang von Hello-Paketen. Festlegen des designierten Routers und des designierten Backup-Routers.
  - exchangestart, exchange und loading
    Status während des Austausches der LSAs
  - full
    Die Datenbank ist komplett und innerhalb des Bereichs synchron. Die Routen können nun ermittelt werden.

    Hinweis

    Normalzustand

    Wenn der Partner-Router ein designierter Router oder ein designierter Backup Router ist, ist der Status "full". Andernfalls ist der Status "two-way".
- **Assoc. Area Type**
  Zeigt die Bereichsart an, über die die Nachbarschaftsbeziehung besteht. Es gibt folgende Bereichsarten:
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Priorität**
  Zeigt die Priorität des Nachbar-Routers an. Diese ist nur während der Auswahl des designierten Routers auf einem Netzwerk von Bedeutung. Bei virtuellen Nachbar-Routern ist diese Angabe nicht relevant.
- **Unterdr. Hello-Pakete**
  Zeigt die unterdrückten Hello-Pakete zum Nachbar-Router an. Dieses Feld steht normalerweise auf "Nein".

- **Retrans Queue**
  Zeigt die Länge der Warteschlange mit den noch zu übertragenden Hello-Paketen an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen an.

OSPFv3 Virtuelle Nachbarn

Übersicht

Diese Seite zeigt die konfigurierten virtuellen Nachbarn an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die OSPFv3-Schnittstelle an, über die der Nachbar-Router erreichbar ist.
- **Router-ID**
  Zeigt die ID des virtuellen Nachbar-Routers an.

- **Status**
  Zeigt den Status des virtuellen Nachbar-Routers an. Der Status kann folgende Werte annehmen:
  - unknown
    Status des Nachbar-Routers unbekannt.
  - down
    Der Nachbar-Router ist nicht erreichbar.
  - attempt und init
    Kurzlebiger Status während der Initialisierung
  - two-way
    Beiderseitiger Empfang von Hello-Paketen. Festlegen des designierten Routers und des designierten Backup-Routers.
  - exchangestart, exchange und loading
    Status während des Austausches der LSAs
  - full
    Die Datenbank ist komplett und innerhalb des Bereichs synchron. Die Routen können nun ermittelt werden.

  Hinweis

  Normalzustand

  Wenn der Partner-Router ein designierter Router oder ein designierter Backup Router ist, ist der Status "full". Andernfalls ist der Status "two-way".

- **Trans. Area-ID**
  Zeigt die ID des Bereichs an, über die die virtuelle Nachbarschaftsbeziehung besteht.
- **Unterdr. Hello-Pakete**
  Zeigt die unterdrückten Hello-Pakete zum Nachbar-Router an. Dieses Feld steht normalerweise auf "Nein".
- **Hello Queue**
  Zeigt die Länge der Warteschlange mit den noch zu übertragenden Hello-Paketen an.
- **Ereignisse**
  Zeigt die Anzahl der Statusänderungen an.

OSPFv3 AS-Scope LSDB

Übersicht .

Die AS-Scope LSDB besteht aus den AS External LSAs. Die wichtigsten Daten dieser LSAs werden auf dieser Seite angezeigt

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **LSA-Typ**
  - AS External
    Die AS External LSA (Typ 0x4005) wird vom AS Border Router innerhalb einer autonomen Systems gesendet. Die LSA enthält Informationen über Routen von einem Netzwerk in ein anderes Netzwerk.
- **Link State-ID**
  Zeigt die ID der LSA an.
- **Router-ID**
  Zeigt die ID des Routers an, der diese LSA gesendet hat.
- **Sequenznummer**
  Zeigt die Sequenznummer der LSA an. Mit jeder Erneuerung eines LSAs wird diese Sequenznummer um eins hoch gezählt.

OSPFv3 Area-Scope LSDB

Übersicht .

Die Area-Scope LSDB ist die zentrale Datenbank zur Verwaltung aller Verbindungen innerhalb eines Bereichs. Sie Die wichtigsten Daten der LSAs werden auf dieser Seite angezeigt.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Area-ID**
  Zeigt die ID des Bereichs an, zu der die LSA gehört. Wenn der LSA eine externe Verbindung ist, wird ein '-' angezeigt.
- **LSA-Typ**
  Zeigt den LSA-Typ an. Folgende Werte sind möglich:
  - Router
    Die Router LSA (Typ 0x2001) wird vom OSPF-Router innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über den Status aller Routerschnittstellen. Enthält aber keine Adressinformationen mehr. Diese sind in dem neuen LSA-Typ 0x2009 enthalten.
  - Network
    Die Network LSA (Typ 0x2002) wird vom designierten Router innerhalb eines Bereichs gesendet. Die LSA enthält eine Liste mit Routern, die mit dem Netzwerk verbunden sind. Enthält aber keine Adressinformationen mehr. Diese sind in dem neuen LSA-Typ 2009 enthalten.
  - Inter-Area Prefix
    Die Inter-Area Prefix LSA (Typ 0x2003) wird vom ABR innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über Routen zu anderen Netzwerken.
  - Inter-Area Router
    Die Inter-Area Router LSA (Typ 0x2004) wird vom Area Border Routers innerhalb eines Bereichs gesendet. Die LSA enthält Informationen über Routen zu anderen autonomen Systemen.
  - Typ-7
    Die Typ-7 LSA (Typ 0x2007) wird vom NSSA-ASBR innerhalb einer NSSA gesendet. Der NSSA-ASBR empfängt LSAs vom Typ 0x4005 und konvertiert die Information in LSAs vom Typ 0x2007. Diese LSAs kann der NSSA-Router innerhalb einer NSSA weiter senden.
  - Intra-Area Prefix
    Die Intra-Area Prefix LSA (Typ 0x2009) wird nur innerhalb eines Bereichs gesendet. Sie enthält die IPv6-Präfixe, die mit dem Router bzw. das Netzwerk verbunden sind.
- **Link State-ID**
  Zeigt die ID der LSA an.
- **Router-ID**
  Zeigt die ID des Routers an, der diese LSA gesendet hat.
- **Sequenznummer**
  Zeigt die Sequenznummer der LSA an. Mit jeder Erneuerung eines LSAs wird diese Sequenz-Nummer um eins hoch gezählt.

OSPFv3 Link-Scope LSDB

Übersicht .

Die Link-Scope LSDB besteht aus den Link LSAs. Die wichtigsten Daten dieser LSAs werden auf dieser Seite angezeigt

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die OSPFv3-Schnittstelle an, zu der die Link-LSA gehört.
- **LSA-Typ**
  - Link
    Die Link LSA (Typ 0x2009) wird vom Router an alle Router gesendet, die mit diesem verlinkt sind. Sie enthält die link local-Adresse des Routers und eine Liste mit IPv6-Präfixen, die auf dem Link konfiguriert sind.
- **Link State-ID**
  Zeigt die ID der LSA an.
- **Router-ID**
  Zeigt die ID des Routers an, der diese LSA gesendet hat.
- **Sequenznummer**
  Zeigt die Sequenznummer der LSA an. Mit jeder Erneuerung eines LSAs wird diese Sequenznummer um eins hoch gezählt.

RIPng-Statistiken

Übersicht

Diese Seite zeigt die Statistik der RIPng-Schnittstellen an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die RIPng-Schnittstelle an.
- **Status**
  Zeigt den Status der RIPng-Schnittstelle an.
  - up
  - down
- **Empfangene Meldungen**
  Zeigt an, wie oft der Router Meldungen empfangen hat.
- **Empfangene Anfragen**
  Zeigt an, wie oft der Router Anfragen empfangen hat.
- **Empfangene Antworten**
  Zeigt an, wie oft der Router Antworten empfangen hat.
- **Unbekannte Befehle**
  Zeigt an, wie viele RIPng-Pakete empfangen wurden, deren Wert im Feld "Kommandos" des RIPng-Headers unbekannt ist. Bekannte Werte sind 1 für Anfragen und 2 für Antworten.
- **Andere Version**
  Zeigt an, wie viele RIPng-Pakete empfangen wurden, deren Wert im Feld "Version" nicht 1 ist.
- **Verworfene Pakete**
  Zeigt an, wie oft der Router RIPng-Pakete verworfen hat.
- **Gesendete Meldungen**
  Zeigt an, wie oft der Router Meldungen gesendet hat.
- **Gesendete Anfragen**
  Zeigt an, wie oft der Router Anfragen gesendet hat.
- **Gesendete Anworten**
  Zeigt an, wie oft der Router Antworten gesendet hat
- **Gesendete Updates**
  Zeigt an, wie oft der Router seine Routingtabelle an seine Nachbar-Router gesendet hat.

DHCP-Server

Diese Seite zeigt an, welche IPv4-Adressen vom DHCP-Server den Geräten zugeordnet wurden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

- **IP-Adresse**
  Zeigt die IPv4-Adresse an, die dem DHCP-Client zugeordnet ist.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Identifikationsmethode**
  Zeigt die Methode an, nach der der DHCP-Client identifiziert wird.
- **Identifikationswert**
  Zeigt die MAC-Adresse oder die Client-ID des DHCP-Clients an.
- **Zuordnungsmethode**
  Zeigt an, ob die IPv4-Adresse statisch oder dynamisch vergeben wurde. Die statischen Einträge konfigurieren Sie unter "System > DHCP > Statische Zuordnung".
- **Zuordnungsstatus**
  Zeigt den Status der Zuordnung an.
  - Zugeordnet
    Die Zuordnung wird verwendet.
  - Nicht verwendet
    Die Zuordnung wird nicht verwendet.
  - Wird geprüft
    Die Zuordnung wird geprüft.
  - Unbekannt
    Der Status der Zuordnung ist unbekannt.
- **Ablaufzeit**
  Zeigt an, bis wann die vergebene IPv4-Adresse noch gültig ist. Bis zu diesem Zeitpunkt muss der DHCP-Client entweder eine neue IPv4-Adresse anfordern oder die Gültigkeitsdauer der vergebenen IPv4-Adresse verlängern.

SNMP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die angelegten SNMPv3-Gruppen. Die SNMPv3-Gruppen konfigurieren Sie unter "System > SNMP".

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt den Gruppennamen an.
- **Benutzername**
  Zeigt den Benutzer an, welcher der Gruppe zugeordnet ist.

Security

Übersicht

Die Seite zeigt die Sicherheitseinstellungen und die lokalen und externen Benutzerkonten an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Dienste

Die Liste "Dienste" zeigt die Sicherheitseinstellungen an.

- **SSH-Server**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration".
  - Aktiviert: Verschlüsselter Zugriff auf das CLI
  - Deaktiviert: Kein verschlüsselter Zugriff auf das CLI
- **Webserver**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration"
  - HTTP/HTTPS: Der Zugriff auf das WBM ist über HTTP und HTTPS möglich.
  - HTTPS: Der Zugriff auf das WBM ist nur noch über HTTPS möglich.
- **SNMP**
  Die Einstellung konfigurieren Sie unter "System > SNMP > Allgemein".
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich.
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit der SNMP Version 3 möglich.
- **Management ACL**
  Die Einstellung konfigurieren Sie unter "Security > Management ACL".
  - Deaktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist deaktiviert.
  - Aktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist aktiviert, es sind jedoch keine Zugriffsregeln definiert.
  - Aktiviert: Nur eingeschränkter Zugriff
    Die Zugriffskontrolle ist aktiviert und es sind Zugriffsregeln definiert.

- **Login-Authentifizierung**
  Die Einstellung konfigurieren Sie unter "Security > AAA > Allgemein".
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
  - Lokal und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen.
    Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS mit Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
    Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.
- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie aktuell verwendet wird.

Lokale und externe Benutzerkonten

Lokale Benutzerkonten und Rollen konfigurieren Sie unter "Security > Benutzer".

Wenn Sie ein lokales Benutzerkonto anlegen, wird automatisch auch ein externes Benutzerkonto erzeugt. Bei lokalen Benutzerkonten handelt es sich um Benutzer mit jeweils einem Passwort zur Anmeldung auf dem Gerät.

In der Tabelle "Externe Benutzerkonten" wird ein Benutzer mit einer Rolle verknüpft, z. B. der Benutzer "Observer" mit der Rolle "user". In diesem Fall ist der Benutzer auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert, die zugehörige Gruppe jedoch unbekannt oder nicht vorhanden ist, prüft das Gerät, ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzerkonten" gibt. Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet. Wenn die zugehörige Gruppe auf dem Gerät bekannt ist, werden beide Tabellen ausgewertet. Dem Benutzer wird die Rolle mit den größeren Rechten zugewiesen.

Hinweis

Die Tabelle "Externe Benutzerkonten" wird nur ausgewertet, wenn Sie im RADIUS Authorisierungsmodus "Herstellerspezifisch" eingestellt haben.

Über CLI können Sie auf die externen Benutzerkonten zugreifen.

Die Tabelle "Lokale Benutzerkonten" gliedert sich in folgende Spalten:

- **Benutzerkonto**
  Zeigt den Namen des lokalen Benutzers an.
- **Rolle**
  Zeigt die Rolle des Benutzers an. Weitere Informationen zu den Funktionsrechten der Rolle erhalten Sie unter "Information > Security > Rollen".

Unterstützte Funktionsrechte

Die Seite zeigt die Funktionsrechte an, die lokal auf dem Gerät verfügbar sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

- **Funktionsrecht**
  Zeigt die Nummer des Funktionsrechts an. Den Nummern sind unterschiedliche Rechte in Bezug auf die Geräteparameter zugeordnet.
- **Beschreibung**
  Zeigt die Beschreibung des Funktionsrechts an.

Rollen

Die Seite zeigt die Rollen an, die lokal auf dem Gerät gültig sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Rolle**
  Zeigt die Rolle an.
- **Funktionsrecht**
  Zeigt die Funktionsrechte der Rolle an:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen, aber nicht verändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.
  - 0
    Hierbei handelt es sich um eine Rolle, die das Gerät intern vergibt, wenn der Benutzer nicht authentifiziert werden konnte.
    Dem Benutzer wird der Zugriff auf das Gerät verweigert.
- **Beschreibung**
  Zeigt eine Beschreibung der Rolle an.

Gruppen

Diese Seite zeigt an, welche Gruppe mit welcher Rolle verknüpft ist. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an. Der Name entspricht der Gruppe auf dem RADIUS-Server.
- **Rolle**
  Zeigt die Rolle an. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät.
- **Beschreibung**
  Zeigt die Beschreibung für die Verknüpfung an.

System-Funktionen projektieren

Konfiguration

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes. Legen Sie fest, über welche Dienste auf das Gerät zugegriffen wird. Zu einigen Diensten gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind.

Beschreibung

Die Seite enthält folgende Felder:

- **Telnet-Server**
  Aktivieren oder deaktivieren Sie den Dienst "Telnet-Server" für den unverschlüsselten Zugriff auf das CLI.
- **SSH-Server**
  Aktivieren oder deaktivieren Sie den Dienst "SSH-Server" für den verschlüsselten Zugriff auf das CLI.
- **Nur HTTPS-Server**
  Wenn diese Funktion aktiviert ist, können Sie nur noch über HTTPS auf das Gerät zugreifen.
- **DNS-Client**
  Aktivieren oder deaktivieren Sie, dass der IE-Switch als DNS-Client fungiert. Weitere Einstellungen konfigurieren Sie unter "System > DNS-Client".

- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client. Weitere Einstellungen konfigurieren Sie unter "System > SMTP-Client".
- **Syslog-Client**
  Aktivieren oder deaktivieren Sie den Syslog-Client. Weitere Einstellungen konfigurieren Sie unter "System > Syslog-Client".
- **DCP-Server**
  Legen Sie fest, ob auf das Gerät mit DCP (Discovery and Configuration Protocol) zugegriffen werden kann:
  - "-" (Deaktiviert)
    DCP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben
    Mit DCP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt
    Mit DCP können Geräteparameter zwar gelesen aber nicht verändert werden.
- **RCDP-Server**
  Legen Sie fest, ob mit RCDP (RUGGEDCOM Discovery Protocol) auf das Gerät zugegriffen werden kann:
  - "-" (Deaktiviert)
    RCDP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben
    Mit RCDP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt
    Mit RCDP können Geräteparameter zwar gelesen, aber nicht verändert werden.
- **Zeiteinstellung**
  Wählen Sie die gewünschte Einstellung aus. Folgende Einstellungen sind möglich:
  - Manuell
    Die Systemzeit wird manuell eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > Manuelle Einstellung".
  - SIMATIC Time
    Die Systemzeit wird über einen SIMATIC Zeitgeber eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SIMATIC Time Client".
  - SNTP-Client
    Die Systemzeit wird über einen SNTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SNTP-Client".
  - NTP-Client
    Die Systemzeit wird über einen NTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > NTP-Client".
  - PTP-Client (nur bei SCALANCE XR528-6M und XR552-12M)
    Die Systemzeit wird über einen PTP-Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > PTP-Client".

- **SNMP**
  Wählen Sie das gewünschte Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist mit der SNMP Version 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
- **SNMPv1/v2 schreibgeschützt**
  Aktivieren oder deaktivieren Sie den schreibenden Zugriff auf SNMP-Variablen bei SNMPv1/v2c.
- **SNMPv1 Traps**
  Aktivieren oder deaktivieren Sie das Versenden von SNMP Traps (Alarmtelegramme). Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Traps".
- **SINEMA-Konfigurationsschnittstelle**
  Wenn die SINEMA Konfigurationsschnittstelle aktiviert ist, können Sie Konfigurationen über das TIA-Portal auf den IE-Switch laden.
- **DUID-Type**
  Legen Sie fest, welcher DUID-Typ verwendet wird. Die DUID-Typen sind in der RFC 3315 definiert.
  - DUID-LLT
    DUID basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel
  - DUID-EN
    DUID wird vom Hersteller (EN = Enterprise number) vergeben
  - DUID-LL
    DUID basiert auf der Link-layer-Adresse der Schnittstelle
- **Link-layer Adresse plus Zeit (LLT)**
  Der Wert basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Unternehmensnummer des Herstellers (EN)**
  Der Wert basiert auf der Unternehmensnummer, die spezifisch für den Hersteller ist. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Link-layer Adresse (LL)**
  Die Link-layer-Adresse basiert auf der MAC-Adresse. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.

- **NFC** (nur bei SCALANCE XM-400)
  Aktivieren oder deaktivieren Sie die Funktion "NFC" (Near Field Communication).
  Weitere Informationen zu NFC finden Sie in der Betriebsanleitung "SCALANCE XM400".
- **Konfigurationsmodus**
  Wählen Sie die gewünschte Betriebsart aus. Folgende Betriebsarten sind möglich:
  - Automatisches Speichern
    Automatischer Sicherungsbetrieb. Ca. 1 Minute nach der letzten Parameteränderung oder beim Neustart des Geräts wird die Konfiguration automatisch abgespeichert.
  - Trial
    Im Konfigurationsmodus "Trial" werden Änderungen zwar übernommen aber nicht in der Konfigurationsdatei (Startup Configuration) gespeichert.
    Um Änderungen in der Konfigurationsdatei abzuspeichern, verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration". Die Schaltfläche "Schreiben der Startkonfiguration" wird eingeblendet, wenn Sie den Trial-Modus einstellen. Zusätzlich wird im Anzeigebereich die Meldung "Der Konfigurationsmodus Trial ist aktiv - Klicken Sie auf die Schaltfläche "Schreiben der Startkonfiguration" um Ihre Einstellungen zu speichern" angezeigt, sobald es ungespeicherte Änderungen gibt. Diese Meldung ist auf jeder Seite sichtbar, bis die vorgenommenen Änderungen entweder gespeichert werden oder das Gerät neu gestartet wird.

Allgemein

Gerät

Diese Seite enthält die allgemeinen Geräteinformationen.

Beschreibung

Die Seite enthält folgende Felder:

- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemuhrzeit an. Die Systemuhrzeit wird entweder vom Anwender eingestellt oder per Uhrzeittelegramm synchronisiert: entweder SINEC H1 Uhrzeittelegramm, NTP oder SNTP. (Nur lesbar)
- **Systembetriebszeit** (nur Online verfügbar)
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an. (Nur lesbar)
- **Gerätetyp** (nur Online verfügbar)
  Zeigt die Typenbezeichnung des Geräts an. (Nur lesbar)
- **Systemname**
  Sie können den Namen des Gerätes eintragen. Der eingetragene Name wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich. Der Systemname wird auch in der CLI-Eingabeaufforderung (Prompt) angezeigt. In der CLI-Eingabeaufforderung ist die Anzahl der Zeichen begrenzt. Der Systemname wird nach 16 Zeichen abgeschnitten.

- **Kontaktperson**
  Sie können den Namen einer Kontaktperson eintragen, die für die Verwaltung des Gerätes zuständig ist. Es sind maximal 255 Zeichen möglich.
- **Gerätestandort**
  Sie können den Montageort des Gerätes eintragen. Der eingetragene Montageort wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich.

Hinweis

In den Eingabefeldern werden die ASCII-Zeichen 0x20 bis 0x7e verwendet.

Koordinaten

Auf dieser Seite können Informationen über die geografischen Koordinaten (Breitengrad, Längengrad und die Höhe über dem Ellipsoid gemäß WGS84) eingetragen werden.

Ermittlung der Koordinaten

Nutzen Sie zur Ermittlung der geografischen Koordinaten des Gerätes entsprechendes Kartenmaterial.

Die geografischen Koordinaten können auch durch einen GPS-Empfänger ermittelt werden. Meist werden die geografischen Koordinaten von diesen Geräten direkt angezeigt und müssen nur noch in die Eingabefelder dieser Seite übertragen werden.

Beschreibung

Die Seite enthält folgende Eingabefelder mit einer maximalen Länge von 32 Zeichen:

- **Eingabefeld "Geographische Breite"**
  Geben Sie die nördliche oder die südliche Breite für den Standort des Gerätes ein.
  Z. B. +49° 1´ 31.67" bedeutet, dass sich das Gerät auf 49 Grad, 1 Bogenminute und 31.67 Bogensekunden nördlicher Breite befindet.
  Die südliche Breite wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben N (nördliche Breite) oder S (südliche Breite) an die Zahlenangabe anhängen (49° 1´ 31.67" N).
- **Eingabefeld "Geographische Länge"**
  Geben Sie die östliche oder die westliche Länge für den Standort des Gerätes ein.
  Z. B. +8° 20´ 58.73" bedeutet, dass sich das Gerät auf 8 Grad, 20 Bogenminuten und 58.73 Bogensekunden östliche Länge befindet.
  Die westliche Länge wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben O bzw. E (östliche Länge) oder W (westliche Länge) an die Zahlenangabe anhängen (8° 20´ 58.73" E).
- **Eingabefeld "Geographische Höhe"**
  Geben Sie die geografische Höhe über oder unter normal Null (Meereshöhe) in Metern ein.
  Z.B. 158 m bedeutet, dass sich das Gerät in einer Höhe von 158 m über normal Null befindet.
  Höhenangaben unterhalb von normal Null (z. B. am toten Meer) werden mit einem führenden Minuszeichen dargestellt.

Agent-IP

Konfiguration der IP-Adressen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Auf dieser Seite nehmen Sie die IP-Konfiguration des Geräts vor.

Bei Geräten mit mehreren IP-Schnittstellen verweist dieser Aufruf auf den Menüpunkt "Subnetze > Konfiguration" im Menü "Layer 3 (IPv4)" und die dortige Konfiguration der TIA-Schnittstelle.

Beschreibung

Die Seite enthält folgende Felder:

- **Methode der IP-Adresszuweisung**
  Zeigt an, wie die IP-Adresse zugeordnet wird.
  - Statisch
    Die IP-Adresse ist statisch. Die IP-Einstellungen tragen Sie in den Eingabefeldern "IP-Adresse" und "Subnetzmaske" ein.
  - Dynamisch (DHCP)
    Das Gerät bezieht eine dynamische IP-Adresse von einem DHCP-Server.
- **IP-Adresse**
  Tragen Sie die IP-Adresse des Geräts ein.
  Nach dem Anklicken der Schaltfläche "Einstellungen übernehmen" wird diese IP-Adresse auch in der Adresszeile des Internet-Browsers angezeigt. Sollte dies nicht automatisch erfolgen, müssen Sie die IP-Adresse manuell in die Adresszeile des Internet-Browsers eintragen.
- **Subnetzmaske**
  Tragen die Subnetzmaske des Geräts ein.
- **Standard-Gateway**
  Tragen Sie die IP-Adresse des Standard-Gateways ein, um mit Geräten in einem anderen Subnetz zu kommunizieren, z. B. Diagnosestationen, E-Mail-Server.

- Agent VLAN-ID
  Wählen Sie aus der Klappliste die VLAN ID. Sie können nur aus bereits konfigurierten VLANs auswählen.
  In dem Modus "802.1D Transparent Bridge" ist diese Klappliste ausgegraut, siehe auch "Layer 2 > VLAN > Allgemein".

  Hinweis

  Ändern der Agent VLAN-ID

  Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN-ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

- MAC-Adresse
  Zeigt die MAC-Adresse des Geräts an. Die MAC-Adresse ist hardwaregebunden und kann nicht geändert werden.

DNS

DNS-Client

Auf dieser Seite können Sie bis zu 3 DNS-Server mit IPv4- oder IPv6-Adressen manuell konfigurieren. Manuell konfigurierten DNS-Servern wird jeweils ein Index von 1 bis 3 zugeordnet. Der IE-Switch kann über DHCP 2 DNS-Server mit IPv4-Adressen lernen. Gelerneten DNS-Servern wird automatisch ein Index von 6 bis 7 zugeordnet.

Bei mehreren DNS-Servern wird durch die Reihenfolge in der Tabelle festlegen, in welcher Reihenfolge die Server angefragt werden. Der obere Server wird zuerst angefragt. Insgesamt können auf dem IE-Switch 5 DNS-Server konfiguriert sein. Manuell konfigurierte DNS-Server werden bevorzugt.

Der DNS-Server (Domain Name System) ordnet einer IP-Adresse einen Domainnamen zu, sodass ein Gerät eindeutig identifiziert werden kann.

Wenn diese Funktion aktiviert ist, kann der IE-Switch als DNS-Client mit einem DNS-Server kommunizieren. Sie haben die Möglichkeit Namen in IP-Adressfeldern einzutragen.

Hinweis

Um die Funktion "DNS-Client" zu nutzen, muss sich ein DNS-Server im Netzwerk befinden.

Beschreibung

Die Seite enthält folgende Felder:

- **DNS-Client**
  Aktivieren oder deaktivieren Sie, dass der IE-Switch als DNS-Client fungiert.
- **Verwendete DNS-Server**
  Legen Sie fest, welche DNS-Server das Gerät verwendet:
  - learned only
    Das Gerät verwendet nur die durch DHCP zugewiesenen DNS-Server.
  - manual only
    Das Gerät verwendet nur die manuell projektierten DNS-Server. Die DNS-Server müssen mit dem Internet verbunden sein. Maximal drei DNS-Server sind projektierbar.
  - all
    Das Gerät verwendet alle verfügbaren DNS-Server.
- **Adresse des DNS-Servers**
  Geben Sie die IP-Adresse des DNS-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Adresse des DNS-Servers**
  Zeigt die IP-Adresse des DNS-Servers an.
- **Erstellung**
  Zeigt an, ob der DNS-Server manuell konfiguriert wurde oder durch DHCP zugewiesen wurde.

DNS-Domaine

Auf dieser Seite können Sie bis zu 4 Domainnamen festlegen. Der primäre Domainname wird zuerst herangezogen, um einen Hostnamen aufzulösen.

Die Domainnamen 2 bis 4 können gelernt oder auf dieser Seite manuell konfiguriert werden. Bei mehreren DNS-Servern ist durch die Reihenfolge in der Tabelle festgelegt, in welcher Reihenfolge die Domainnamen herangezogen werden.

Wenn Domainnamen hinterlegt sind, haben Sie bei einigen IP-Adressfeldern die Möglichkeit, den Hostnamen einzutragen.

Beschreibung

Die Seite enthält folgende Felder:

- **Primäre Domain**
  Geben Sie den Namen der primären Domain ein. Dieser Eintrag wird zuerst herangezogen, um einen Hostnamen aufzulösen.
- **Domainname**
  Geben Sie den Namen der weiteren Domain ein.

Die Tabelle enthält folgende Spalten:

- **Domainname**
  Zeigt den Namen der weiteren Domain an.
- **Erstellung**
  Zeigt an, ob der Domainname manuell konfiguriert wurde oder durch DHCP zugewiesen wurde.

Neustart

In diesem Menü finden Sie eine Schaltfläche zum Neustart des Geräts sowie die Möglichkeit, das Gerät auf die Werkseinstellungen bzw. die Defaulteinstellungen verschiedener Profile zurückzusetzen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Neustart

Beachten Sie folgende Punkte beim Neustart eines Geräts:

- Sie können einen Neustart des Gerätes nur mit Administrator-Rechten durchführen.
- Der Neustart eines Gerätes sollte nur durch die Schaltflächen dieses Menüs oder durch die entsprechenden CLI-Befehle und nicht durch Aus- und Einschalten der Spannungsversorgung am Gerät erfolgen.
- Wenn sich das Gerät im Modus "Trial" befindet, müssen Konfigurationsänderungen vor einem Neustart manuell abgespeichert werden. Vorgenommene Änderungen werden erst nach dem Anklicken der Schaltfläche "Einstellungen übernehmen" auf der jeweiligen WBMSeite im Gerät wirksam.
- Wenn sich das Gerät im Modus "Automatisches Speichern" befindet, werden die letzten Änderungen automatisch vor einem Neustart gespeichert.

Auf Werkseinstellungen zurücksetzen

Durch das Zurücksetzen aller Einstellungen auf die Werkseinstellungen gehen auch die IP-Adresse und die Passwörter verloren. Das Gerät ist danach nur über die serielle Schnittstelle, das Primary Setup Tool oder über DHCP ansprechbar.

ACHTUNG
Ausfall des Datenverkehrs Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

Auf Defaults zurücksetzen (Profile)

Die Profile bieten eine Vorkonfiguration für verschiedene Einsatzfälle der Geräte.

Wenn Sie ein Gerät mit den Defaulteinstellungen eines Profils neustarten, werden die Einstellungen auf die Werkseinstellungen zurückgesetzt und einige Parameter so gesetzt, dass sie für einen Einsatzfall ausgelegt sind. Im Gegensatz zum Zurücksetzen auf Werkseinstellungen, bleiben Benutzer und Passwörter nach dem Neustart erhalten. Die konfigurierte IP-Adresse geht verloren, sodass das Gerät danach nur über die serielle Schnittstelle, das Primary Setup Tool oder über DHCP ansprechbar ist.

ACHTUNG
Ausfall des Datenverkehrs Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

Welche Einstellungen speziell für das Profil gesetzt wurden, wird vor dem Neustart angezeigt. Die Profile können unabhängig von der Werkseinstellung des Geräts genutzt werden.

Beschreibung

Hinweis

Beachten Sie die Auswirkungen der einzelnen Funktionen, die in den oberen Abschnitten beschrieben sind.

Für den Neustart des Geräts stehen Ihnen mit den Schaltflächen auf dieser Seite folgende Möglichkeiten zur Verfügung:

- **Neustart**
  Klicken Sie auf diese Schaltfläche, um das System neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. Bei einem Neustart wird das Gerät neu initialisiert, die interne Firmware wird neu geladen und das Gerät führt einen Selbsttest durch. Die Einstellungen der Startkonfiguration bleiben erhalten, z. B. die IP-Adresse des Geräts. Die gelernten Einträge in der Adresstabelle werden gelöscht. Sie können das Browser-Fenster geöffnet lassen, während das Gerät neu startet. Nach dem Neustart müssen Sie sich wieder neu anmelden.
- **Auf Werkseinstellungen zurücksetzen und Neustart**
  Klicken Sie auf diese Schaltfläche, um die Werkseinstellungen des Geräts wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. Die Werkseinstellungen sind vom Gerät abhängig.

Für den Neustart des Geräts mit einem vordefinierten Profil stehen Ihnen mit den Schaltflächen auf dieser Seite folgende Möglichkeiten zur Verfügung:

- PROFINET-Defaults
  Klicken Sie auf diese Schaltfläche, um die Defaulteinstellungen des PROFINET-Profils wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. In der Dialogbox werden die Einstellungen angezeigt, die speziell auf den Betrieb mit dem Protokoll PROFINET abgestimmt sind.
- EtherNet/IP-Defaults
  Klicken Sie auf diese Schaltfläche, um die Defaulteinstellungen des EtherNet/IP-Profils wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. In der Dialogbox werden die Einstellungen angezeigt, die speziell auf den Betrieb mit dem Protokoll EtherNet/IP abgestimmt sind.
- Industrial Ethernet-Defaults
  Klicken Sie auf diese Schaltfläche, um die Defaulteinstellungen des Industrial Ethernet-Profils wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. In der Dialogbox werden die Einstellungen angezeigt, die speziell auf den Betrieb im Industrial Ethernet-Umfeld abgestimmt sind.

Laden & Speichern

Dateitypen

Übersicht der Dateitypen

Dateityp	Beschreibung
Config	Diese Datei enthält die Startkonfiguration. Diese Datei enthält unter anderem die Definitionen der Benutzer, Rollen, Gruppen und Funktionsrechte. Die Passwörter sind in der Datei "Users" abgespeichert.
ConfigPack	Detaillierte Konfigurationsinformationen z. B. Startkonfiguration, Benutzer, Zertifikate und WBM-Favoriten. ZIP-Datei, die aus der Config-, Users- und LSYS-Datei besteht.
ConfigPackBackup	In dieser ZIP-Datei sind alle Konfigurationsbackups gespeichert, die Sie erstellt haben.
Debug	Diese Datei beinhaltet Informationen für den Siemens Support. Sie ist verschlüsselt und kann ohne Sicherheitsrisiko per E-Mail an den Siemens Support gesendet werden.
DebugExt	Diese Datei beinhaltet ausführlichere Informationen für den Siemens Support. Sie ist verschlüsselt und kann ohne Sicherheitsrisiko per E-Mail an den Siemens Support gesendet werden. Das Speichern der Datei kann einige Zeit in Anspruch nehmen.
EDS	Electronic Data Sheet (EDS) Elektronische Datenblätter zur Beschreibung von Geräten im EtherNet/IP-Betrieb
Firmware	Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Dateityp	Beschreibung
GSDML	PROFINET-Informationen über die Geräteeigenschaften
HTTPSCert	Voreingestellte HTTPS-Zertifikate inkl. Schlüssel Die voreingestellten und automatisch erstellten HTTPS-Zertifikate sind selbstsigniert. Es wird dringend empfohlen eigene HTTPS-Zertifikate zu erstellen und bereitzustellen. Es wird empfohlen HTTPS-Zertifikate zu verwenden, die entweder durch eine zuverlässige externe oder eine interne Zertifizierungsstelle signiert sind. Das HTTPS-Zertifikat überprüft die Identität des Geräts und regelt den verschlüsselten Datenaustausch. Zertifikate mit einem anderen Format können nicht eingespielt werden.
LogFile	Datei mit Einträgen aus der Ereignisprotokolltabelle
MIB	Private MSPS MIB-Datei
RunningCLI	Textdatei mit CLI-Befehlen Diese Datei enthält eine Übersicht der aktuellen Konfiguration in Form von CLI-Befehlen. Passwörter sind in dieser Datei wie folgt maskiert: [PASSWORD] Sie können die Textdatei herunterladen. Die Datei ist nicht dafür vorgesehen, dass Sie sie unverändert wieder hochladen.
RunningSINEMA-Config	In diesen Dateityp speichern Sie die aktuelle Konfiguration des Geräts für eine Übergabe an STEP7 Basic/Professional. Die Datei kann in STEP7 Basic/Professional importiert und auf ein Gerät mit gleicher Artikelnummer und Firmware-Version aufgespielt werden. Bevor Sie eine Datei abspeichern können, müssen Sie im WBM unter "System > Laden&Speichern > Passwörter" ein Passwort für die "RunningSINEMAConfig" vergeben. Dieses Passwort benötigen Sie auch, um die Datei in STEP7 Basic/Professional zu importieren. siehe auch "SINEMAConfig"
Script	Textdatei mit CLI-Befehlen Sie können eine Skriptdatei in einem Gerät hochladen. Die enthaltenen CLI-Befehle werden entsprechend ausgeführt.
SINEMAConfig	Über diesen Dateityp laden Sie Konfigurationsdaten, die über STEP7 Basic/Professional für eine Übergabe an das WBM exportiert wurden. Um eine Datei laden zu können, müssen Sie unter "System > Laden&Speichern > Passwörter" ein Passwort für die "SINEMAConfig" vergeben. Dieses Passwort benötigen Sie auch, um die Datei aus STEP7 Basic/Professional zu exportieren. siehe auch "RunningSINEMAConfig"
StartupInfo	Startup Logdatei Diese Datei enthält die Meldungen die während des letzten Hochlaufs im Logbook eingetragen wurden.
Users	Diese Datei enthält die Zuordnung der Benutzernamen zu den entsprechenden Passwörtern.
WBMFav	WBM Favoriten Diese Datei enthält die Favoriten, die Sie im WBM angelegt haben. Sie können diese Datei herunterladen und in anderen Geräten hochladen.

HTTP

Laden und speichern von Daten über HTTP

Diese Seite bietet die Möglichkeit, Gerätedaten in einer externen Datei auf Ihrem Client-PC zu speichern bzw. solche Daten aus einer externen Datei vom Client-PC in die Geräte zu laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Client-PC laden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.
Diese Seite ist sowohl für Verbindungen über HTTP als auch für Verbindungen über HTTPS verfügbar.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Trial-Modus/Automatisches Speichern

Im Automatic Save-Modus wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Trial-Modus werden Änderungen zwar übernommen, aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration " auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdatei abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Hochladen**
  Mit dieser Schaltfläche können Sie Dateien auf das Gerät hochladen. Die Schaltfläche ist aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird.

- **Speichern**
  Mit dieser Schaltfläche können Sie Dateien vom Gerät speichern. Die Schaltfläche ist nur aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird und die Datei auf dem Gerät vorhanden ist.
- **Löschen**
  Mit dieser Schaltfläche können Sie Dateien vom Gerät löschen. Die Schaltfläche ist nur aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird und die Datei auf dem Gerät vorhanden ist.

Hinweis

Löschen Sie nach einem Firmware-Update den Cache Ihres Internet-Browsers.

TFTP

Laden und Speichern von Daten über einen TFTP-Server

Auf der Seite können Sie den TFTP-Server und die Dateinamen konfigurieren. Weiter können Sie Gerätedaten in einer externen Datei auf einem TFTP-Server speichern bzw. solche Daten aus einer externen Datei vom TFTP-Server in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von einem TFTP-Server laden.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Modus Trial/Automatisches Speichern

Im Modus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Modus "Trial" werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration" auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden.

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung

- **Adresse des TFTP-Servers**
  Tragen Sie die IP-Adresse oder den FQDN des TFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des TFTP-Servers**
  Tragen Sie hier den Port des TFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 69 entsprechend Ihren spezifischen Anforderungen ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis
  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktion** (nur Online verfügbar)
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp, z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - Datei speichern
    Mit dieser Auswahl speichern Sie eine Datei auf den TFTP-Server.
  - Datei hochladen
    Mit dieser Auswahl laden Sie eine Datei vom TFTP-Server, z. B. Zertifikate, die für den Aufbau einer gesicherten Verbindung notwendig sind.

SFTP

Laden und Speichern von Daten über einen SFTP-Server

SFTP (SSH File Transfer Protocol) überträgt die Dateien verschlüsselt. Auf dieser Seite konfigurieren Sie die Zugangsdaten für den SFTP-Server.

Weiter können Sie Gerätedaten in einer externen Datei auf Ihrem Client-PC speichern bzw. solche Daten aus einer externen Datei vom PC in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Admin-PC laden.

Zudem lassen sich auf dieser Seite die Zertifikate laden, die für den Aufbau einer gesicherten VPN-Verbindung notwendig sind.

Firmware

Die Firmware ist signiert und verschlüsselt. Es ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden.

X509-Zertifikate

Folgende Dateitypen können in das Gerät geladen werden:

- .crt, pem, zip: Maximale Dateinamenlänge 255 Zeichen
- .p12: Maximale Dateinamenlänge 248 Zeichen

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Adresse des SFTP-Servers**
  Geben Sie die IP-Adresse oder den FQDN des SFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des SFTP-Servers**
  Geben Sie den Port des SFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 22 entsprechend Ihren spezifischen Anforderungen ändern.
- **SFTP Benutzer**
  Geben Sie den Benutzer für den Zugriff auf den SFTP-Server ein. Vorausgesetzt, auf dem SFTP-Server ist ein Benutzer mit den entsprechenden Rechten angelegt.
- **SFTP Passwort**
  Geben Sie das Passwort für den Benutzer ein
- **SFTP Passwort bestätigen**
  Bestätigen Sie das Passwort.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis

  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktionen**
  Wählen Sie aus der Klappliste die Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp. z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - **Datei speichern**
    Mit dieser Auswahl speichern Sie eine Datei auf dem SFTP-Server.
  - **Datei hochladen**
    Mit dieser Auswahl laden Sie eine Datei vom SFTP-Server.

Passwörter

Es gibt Dateien, deren Zugriff passwortgeschützt ist. Um die Datei erfolgreich ins Gerät zu laden, geben Sie auf der WBM-Seite das für die Datei festgelegte Passwort ein.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Typ**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Aktiviert**
  Wenn aktiviert, wird das Passwort verwendet. Nur aktivierbar, wenn das Passwort konfiguriert ist.
- **Passwort**
  Geben Sie das Passwort für die Datei ein.
- **Passwort bestätigen**
  Bestätigen Sie das Passwort.
- **Status**
  Zeigt an, ob die aktuellen Einstellungen zur Datei auf dem Gerät passen.
  - gültig
    Die Einstellung "Aktiviert" ist aktiviert und das Passwort passt zu dem Zertifikat.
  - ungültig
    Doe Einstellung "Aktiviert" ist aktiviert, aber das Passwort passt nicht zu dem Zertifikat oder es ist noch kein Zertifikat geladen.
  - '-'
    Das Passwort kann nicht ausgewertet werden oder wird noch nicht verwendet. Das Optionskästchen "Aktiviert" ist nicht aktiviert.

Ereignisse

Konfiguration

Auf dieser Seite legen Sie fest, wie das Gerät auf Systemereignisse reagiert. Durch Aktivieren der entsprechenden Optionen legen Sie fest, wie das Gerät bei Ereignissen reagiert. Klicken Sie zum Aktivieren oder Deaktivieren der Optionen in die entsprechenden Einstellungen der jeweiligen Spalte.

Beschreibung

- **Verhalten des Meldekontakts**
  Wählen Sie das Verhalten des Meldekontakts aus. Es gibt folgende Möglichkeiten:
  - Standard
    Standardeinstellung für den Meldekontakt. Ein auftretender Fehler wird durch die Fehler-LED angezeigt und der Meldekontakt wird geöffnet. Wenn der Fehlerzustand nicht mehr besteht, erlischt die Fehler-LED und der Meldekontakt wird geschlossen.
  - Benutzerdefiniert
    Die Funktion des Meldekontakts ist unabhängig vom auftretenden Fehler. Der Meldekontakt kann durch Benutzeraktionen beliebig geöffnet oder geschlossen werden.
- **Status des Meldekontakts**
  Wählen Sie aus der Klappliste den Zustand des Meldekontakts aus. Folgende Zustände sind möglich:
  - Geschlossen
    Meldekontakt ist geschlossen.
  - Geöffnet
    Meldekontakt ist geöffnet.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **E-Mail/Trap/Log-Tabelle/Syslog/Fehler**
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Ereignis**
  Die Spalte "Ereignis" enthält folgende Werte:
  - Kalt-/Warmstart
    Das Gerät wurde eingeschaltet oder vom Anwender neu gestartet.
  - Link Change
    Dieses Ereignis tritt nur auf, wenn der Port-Status überwacht wird und sich entsprechend geändert hat, siehe "System > Fehlerkontrolle > Link Change".
  - Authentifizierungsfehler
    Dieses Ereignis tritt beim Versuch eines Zugriffs mit fehlerhaftem Passwort auf.
  - RMON-Alarm
    Ein Alarm oder ein Ereignis ist im Zusammenhang mit der Fernüberwachung des Systems aufgetreten.
  - Umschalten der Spannungsversorgung
    Dieses Ereignis tritt nur auf, wenn die Spannungsversorgungsleitungen 1 und 2 überwacht werden. Es zeigt an, dass ein Wechsel auf Leitung 1 bzw. auf Leitung 2 stattgefunden hat, siehe "System > Fehlerkontrolle > Spannungsversorgung".
  - Statusänderung RM
    Der Redundanzmanager hat eine Unterbrechung oder Wiederherstellung des Rings erkannt und hat die Strecke um- bzw. zurückgeschaltet.
  - Änderung im Spanning Tree
    Die STP- bzw. RSTP- oder MSTP-Topologie hat sich geändert.
  - Änderung im Fehlerstatus
    Der Fehlerstatus hat sich geändert. Der Fehlerstatus kann sich auf die aktivierte Portüberwachung, auf das Ansprechen der Meldekontakte oder die Spannungsüberwachung beziehen.
  - Statusänderung Standby
    Ein Gerät mit aufgebauter Standby-Verbindung (Master oder Slave) hat die Koppelstrecke zum anderen Ring (Standby-Port) aktiviert oder deaktiviert. Der Datenverkehr wurde von einer Ethernet-Verbindung (Standby-Port des Master) zu der anderen Ethernet-Verbindung (Standby-Port des Slave) umgeleitet.
  - Statusänderung VRRP (Nur bei Routing über VRRP)
    Der Zustand des virtuellen Routers hat sich geändert
  - Loop Detection
    Es wurde eine Schleife im Netzsegment erkannt.
  - Diagnosealarme
    Der Diagnosewert hat eine bestimmte Grenze unter- bzw. überschritten.
  - Statusänderung OSPF
    Der Status von OSPF hat sich geändert.
  - Statusänderung 802.1X Port-Authentifizierung
    Dieses Ereignis tritt bei 802.1X-Authentifizierungen auf.
  - Statusänderung PoE
    Der Zustand von PoE hat sich geändert.

- Statusänderung FMP
  Der Wert der Empfangsleistung bzw. des Leistungsabfalls hat eine bestimmte Grenze unterschritten.
- Link Check
  Es wurde eine Störungen bei einer optischen Übertragungsstrecke erkannt.
- CLI Skript-Datei
  Es wurde ein Fehler in der CLI Skript-Datei erkannt.
- NTP (gesichert)
  Bei der Nutzung von Secure NTP ist ein Fehler aufgetreten, z. B. wurde ein Schlüssel mit falscher Länge angegeben.
- Änderung in der Konfiguration
  Die Konfiguration wurde persistent gespeichert.
- MRP-Interconnection Statusänderung
  In der MRP-Interconnection-Domäne wurde von der primären auf die sekundäre Verbindung umgeschaltet oder umgekehrt.
- Service-Informationen
  Für bestimmte Ereignisse werden auch ohne Konfiguration Einträge in der Log-Tabelle erstellt. Für diese Ereignisse können Sie hier weitere Folgeaktionen konfigurieren (E-Mail, Trap, Syslog).

- **E-Mail**
  Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- **Trap**
  Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.
- **Log-Tabelle**
  Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle, siehe "Information > Log-Tabelle".
- **Syslog**
  Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- **Fehler**
  Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf

Severity-Filter

Stellen Sie auf dieser Seite die Schwellwertstufen für das Versenden von Systemereignisbenachrichtigungen ein.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Client-Typ**
  Wählen Sie den Client-Typ, für den Sie die Einstellungen vornehmen:
  - E-Mail
    Versand von Systemereignismeldungen per E-Mail
  - Log-Tabelle
    Eintragen von Systemereignissen in die Log-Tabelle.
  - Syslog
    Eintragen von Systemereignissen in die Syslog-Datei
  - WLAN-Authentifizierung Log
    Weiterleitung der Einträge aus dem WLAN Authentication Log an den Systemprotokoll-Server.
    Nur bei SCALANCE W 770 und SCALANCE W780 konfigurierbar
- **Severity**
  Wählen Sie die gewünschte Stufe aus. Folgende Einstellungen sind möglich:
  - Info
    Systemereignisse werden ab dem Severity-Level "Info" bearbeitet.
  - Warning
    Systemereignisse werden ab dem Severity-Level "Warning" bearbeitet.
  - Critical
    Systemereignisse werden ab dem Severity-Level "Critical" bearbeitet.

SMTP-Client

Netzüberwachung durch E-Mails

Das Gerät bietet die Möglichkeit, beim Auftreten eines Alarmereignisses automatisch eine E-Mail (z.B. an den Netzwerkadministrator) zu senden. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Damit kann für Netze mit wenigen Teilnehmern eine einfache zentrale Netzüberwachung auf Basis eines E-Mail-Systems aufgebaut werden. Bei eintreffenden E-Mail-Störmeldungen kann über die Identifikation des Absenders per Internet-Browser das WBM gestartet werden, um weitere Diagnoseinformationen auszulesen.

Auf dieser Seite können Sie bis zu drei SMTP-Server und die dazugehörigen E-Mail-Adressen konfigurieren.

Beschreibung

Die Seite enthält folgende Felder:

- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client.
- **E-Mail-Adresse des Absenders**
  Geben Sie den Absendernamen ein, der in der E-Mail angegeben werden soll, z. B. den Gerätenamen. Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **Test-E-Mail senden**
  Verschicken Sie eine Test-E-Mail, um Ihre Konfiguration zu prüfen.
- **SMTP-Port**
  Geben Sie den Port ein, über den Ihr SMTP-Server erreichbar ist.
  Werkseinstellung: 25
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **SMTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SMTP-Servers ein.

Die Tabelle enthält folgende Spalten:

- **SMTP-Server-Adresse**
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SMTP-Servers.
- **E-Mail-Adresse des Empfängers**
  Geben Sie die E-Mail-Adresse ein, an die das Gerät im Fehlerfall eine E-Mail sendet.

Syntax von E-Mail-Adressen

Folgenden Bedingungen gelten für E-Mail-Adressen:

- Alphanumerische Zeichen sind erlaubt.
- Folgende Sonderzeichen sind erlaubt:
  - @
  - _ (Unterstrich)
  - - (Bindestrich)
  - . (Punkt)
- Es muss ein @-Zeichen vorkommen.
- Es darf nur ein @-Zeichen vorkommen.
- Die Zeichen "@" und "." dürfen nicht das erste bzw. letzte Zeichen sein.

Hinweis

Je nach Eigenschaften und Konfiguration des SMTP-Servers kann es notwendig sein, das Eingabefeld "E-Mail-Adresse des Absenders" anzupassen. Informieren Sie sich beim Administrator des SMTP-Servers.

DHCPv4

DHCP-Client

Einstellung des DHCP-Modus

Wenn das Gerät als DHCP-Client konfiguriert ist, startet es eine DHCP-Anfrage. Das Gerät erhält vom DHCP-Server als Antwort eine IPv4-Adresse zugewiesen. Der Server verwaltet einen Adressbereich, aus welchem er IPv4-Adressen vergibt. Es ist auch möglich den Server so zu konfigurieren, dass der Client auf seine Anfrage immer dieselbe IPv4-Adresse zugewiesen bekommt.

Beschreibungen

- **DHCP-Client Konfigurationsanfrage (Opt. 66, 67)**
  Aktivieren Sie diese Option, wenn der DHCP Client die Optionen 66 und 67 dazu verwenden soll, eine Konfigurationsdatei herunterzuladen und diese dann zu aktivieren.
- **DHCP-Modus**
  Wählen Sie die DHCP-Betriebsart aus. Folgende Modi sind möglich:
  - über MAC-Adresse
    Die Identifikation läuft über die MAC-Adresse ab.
  - über DHCP-Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab.
  - über Systemname
    Die Identifikation läuft über den Systemnamen ab. Ist der Gerätename 255 Zeichen lang, dann wird das letzte Zeichen nicht zur Identifikation benutzt.
  - über PROFINET-Gerätename
    Die Identifikation läuft über den PROFINET-Gerätenamen der Station ab.
  - über IAID und DUID
    Damit kann sich der DHCP-Client an DHCP-Servern anmelden, die den Parallelbetrieb von IPv4-und IPv6 unterstützten.
    Die Identifikation läuft über die IAID und die DUID ab und bezeichnet genau eine IP-Schnittstelle des Geräts bezeichnet.
    IAID (Interface Association Identifier): Für jede IP-Schnittstelle wird mindestens eine IAID generiert. Die IAID bleibt bei Neustart des DHCP-Clients unverändert.
    DUID (DHCP Unique Identifier): Identifiziert Server und Clients eindeutig und gilt für alle IP-Schnittstellen des Geräts. Die DUID bleibt bei Neustart unverändert. Es sei denn, der Benutzer ändert diese.

- **DUID-Typ**
  Legen Sie fest, welcher DUID-Typ verwendet wird. Die DUID-Typen sind in der RFC 3315 definiert.
  - DUID-LLT
    DUID basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel
  - DUID-EN
    DUID wird vom Hersteller (EN = Enterprise number) vergeben
  - DUID-LL
    DUID basiert auf der Link-layer-Adresse der Schnittstelle
- **Link-layer Adresse plus Zeit (LLT)**
  Der Wert basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Unternehmensnummer des Herstellers (EN)**
  Der Wert basiert auf der Unternehmensnummer, die spezifisch für den Hersteller ist. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Link-layer Adresse (LL)**
  Die Link-layer-Adresse basiert auf der MAC-Adresse. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.
- **DHCP**
  Aktivieren oder deaktivieren Sie den DHCP-Client für die entsprechende Schnittstelle.
- **IAID-Wert**
  Wert mit dem sich die Schnittstelle (DHCP-Client) am DHCP-Server identifiziert.

DHCP-Server

Das Gerät können Sie als DHCP-Server betreiben. Damit ist es möglich, den angeschlossenen Geräten automatisch IP-Adressen zuzuweisen. Die IP-Adressen werden entweder dynamisch aus einem von Ihnen vergebenen Adressband (Pool) verteilt oder es wird eine bestimmte IP-Adresse einem bestimmten Gerät zugewiesen.

Hinweis

DHCP-Server-Zuordnungen löschen

Wenn Sie ein IPv4-Adressband deaktivieren bzw. löschen oder den DHCP-Server aus- und wieder einschalten, werden die DHCP-Server-Zuordnungen gelöscht, siehe "Information > DHCP-Server".

Der Aufbau dieser Seite ist davon abhängig, über wie viele VLAN-IP-Schnittstellen das Gerät verfügt.

Voraussetzung

Die angeschlossenen Geräte sind so konfiguriert, dass diese die IP-Adresse von einem DHCP-Server beziehen.

Geräte mit einer VLAN-IP-Schnittstelle

Auf dieser Seite legen Sie IPv4-Adressen fest, die über bestimmte Ports vergeben werden.

Beschreibung

Die Seite enthält folgende Felder:

- **DHCP-Server**
  Aktivieren oder deaktivieren Sie den DHCP-Server auf dem Gerät.

  Hinweis

  Damit keine Konflikte mit IPv4-Adressen entstehen, darf im Netzwek nur ein Gerät als DHCP-Server konfiguriert sein.

- **Adresse vor dem Anbieten mit ICMP-Echo prüfen**
  Wenn aktiviert, prüft der DHCP-Server, ob die IP-Adresse schon vergeben ist. Dazu sendet der DHCP-Server ICMP-Echomeldungen (Ping) an die IPv4-Adresse. Wenn keine Antwort zurückkommt, wird die IPv4-Adresse vergeben.

  Hinweis

  Bei statischen Zuordnungen wird diese Prüfung nicht durchgeführt.

  Hinweis

  Wenn es in Ihrem Netzwerk Geräte gibt, bei denen der Echo-Dienst standardmäßig deaktiviert ist, kann es zu Konflikten bei den IPv4-Adressen kommen. Um dies zu vermeiden, vergeben Sie diesen Geräten eine IPv4-Adresse, die außerhalb des IPv4-Adressbands liegt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an. Wenn Sie auf die Schaltfläche "Erstellen" klicken, wird eine neue Zeile mit einer eindeutigen Nummer (Pool-ID) angelegt.
- **Port**
  Legen Sie fest, über welchen Port die IPv4-Adresse dieses DHCP-Pools vergeben werden sollen.
  Voraussetzung für die Vergabe ist, dass die IPv4-Adresse der Schnittstelle im Subnetz des IPv4-Adressbandes liegt. Wenn das nicht der Fall ist, vergibt die Schnittstelle keine IPv4-Adressen.

- **Aktivieren**
  Legen Sie fest, ob diese IPv4-Adresse verwendet wird.

  Hinweis

  Wenn Sie die IPv4-Adresse aktivieren, werden deren Einstellungen in diesem DHCPRegister ausgegraut und sind nicht mehr editierbar.

- **IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die über den angegebenen Port vergeben werden soll.
- **Subnetz**
  Tragen Sie die Subnetzmaske passend zu der IPv4-Adresse ein. Verwenden Sie die CIDRSchreibweise.
- **Gültigkeitsdauer [Sek]**
  Legen Sie fest, für wie viele Sekunden die vergebene IPv4-Adresse gültig bleibt. Nachdem die Gültigkeitsdauer zur Hälfte abgelaufen ist, kann der DHCP-Client die vergebene IPv4-Adresse verlängern. Nach Ablauf der gesamten Zeitdauer muss der DHCP-Client eine neue IPv4-Adresse anfordern.

Geräte mit mehreren VLAN-IP-Schnittstellen

Auf dieser Seite legen Sie das Adressband fest, aus dem das angeschlossene Gerät eine beliebige IP-Adresse erhält. Die statische Zuordnung der IP-Adressen konfigurieren Sie unter "Statische Zuordnung".

Beschreibung

- **DHCP-Server**
  Aktivieren oder deaktivieren Sie den DHCP-Server auf dem Gerät.

  Hinweis

  Damit keine Konflikte mit IPv4-Adressen entstehen, darf im Netzwek nur ein Gerät als DHCP-Server konfiguriert sein.

  Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, beachten Sie das Kapitel "Layer 3 (IPv4/IPv6) > VRRP/VRRPv3 > Router".

- **Adresse vor dem Anbieten mit ICMP-Echo prüfen**
  Wenn aktiviert, prüft der DHCP-Server, ob die IP-Adresse schon vergeben ist. Dazu sendet der DHCP-Server ICMP-Echomeldungen (Ping) an die IPv4-Adresse. Wenn keine Antwort zurückkommt, wird die IPv4-Adresse vergeben.

  Hinweis

  Bei statischen Zuordnungen wird diese Prüfung nicht durchgeführt.

  Hinweis

  Wenn es in Ihrem Netzwerk Geräte gibt, bei denen der Echo-Dienst standardmäßig deaktiviert ist, kann es zu Konflikten bei den IPv4-Adressen kommen. Um dies zu vermeiden, vergeben Sie diesen Geräten eine IPv4-Adresse, die außerhalb des IPv4-Adressbands liegt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an. Wenn Sie auf die Schaltfläche "Erstellen" klicken, wird eine neue Zeile mit einer eindeutigen Nummer (Pool-ID) angelegt.
- **Schnittstelle**
  Wählen Sie eine IP-Schnittstelle oder einen Router-Port aus. Über diese Schnittstelle werden die IPv4-Adressen dynamisch vergeben.
- **Aktivieren**
  Legen Sie fest, ob dieses IPv4-Adressband verwendet wird.

  Hinweis

  Wenn Sie die IPv4-Adresse aktivieren, werden deren Einstellungen in diesem DHCPRegister ausgegraut und sind nicht mehr editierbar.

- **Subnetz**
  Tragen Sie die Subnetzmaske passend zu der IPv4-Adresse ein. Verwenden Sie die CIDRSchreibweise.
- **Untere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die den Anfang des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Obere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die das Ende des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Gültigkeitsdauer [Sek]**
  Legen Sie fest, für wie viele Sekunden die vergebene IPv4-Adresse gültig bleibt. Nachdem die Gültigkeitsdauer zur Hälfte abgelaufen ist, kann der DHCP-Client die vergebene IPv4-Adresse verlängern. Nach Ablauf der gesamten Zeitdauer muss der DHCP-Client eine neue IPv4-Adresse anfordern.

Zuordnung Port zu IP-Adresse

Auf dieser Seite ordnen Sie einem bestimmten Port genau eine IP-Adresse zu. Nachdem Sie im Register "DHCP-Server" ein IPv4-Adressband angelegt haben, wird in diesem Register eine neue Zeile angelegt. Die Konfiguration auf dieser Seite wirkt sich auf die Register "DHCP-Server" und "Port-Bereich" aus.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an. Für jedes Adressband wird eine Zeile angelegt.
- **Port**
  Wählen Sie aus der Klappliste die Einstellung. Sie haben folgende Einstellungsmöglichkeiten:
  - Px.y
    Legen Sie fest, über welchen Port die IPv4-Adresse vergeben werden sollen. Sie können nur Ports auswählen, die sich in dem entsprechenden VLAN befinden. Wenn Sie einen Port auswählen, ist im Register "Port-Bereich" nur noch dieser Port aktiviert.
  - Nicht ausgewählt
    Bei dieser Einstellung sind im Register "Port-Bereich" keine Ports oder mehr als ein Port ausgewählt. Wenn Sie die Einstellung "Nicht ausgewählt" auswählen, werden alle Ports im Register "Port-Bereich" deaktiviert.
- **IP-Adresse**
  Geben Sie eine IPv4-Adresse an. Im Register "DHCP-Server" werden die Felder "Untere IP-Adresse" und "Obere IP-Adresse" entsprechend ausgefüllt.
- **Subnetzmaske**
  Geben Sie eine dazugehörige Subnetzmaske an. Im Register "DHCP-Server" wird das Feld "Subnetz" entsprechend ausgefüllt.

Port-Bereich

Auf dieser Seite definieren Sie die Ports, über die die IPv4-Adressen eines Adressbands vergeben werden.

Nachdem Sie im Register "DHCP-Server" ein IPv4-Adressband angelegt haben, wird in diesem Register eine neue Zeile angelegt und alle Ports ausgewählt, die sich zu diesem Zeitpunkt in dem entsprechenden VLAN befinden. Wenn Sie nachträglich Ports zu dem VLAN hinzufügen, werden die Ports in diesem Register nicht automatisch aktiviert.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an. Für jedes Adressband wird eine Zeile angelegt.
- **Schnittstelle**
  Zeigt die zugeordnete VLAN-IP-Schnittstelle an.

- **Alle Ports**
  Wählen Sie aus der Klappliste die Einstellung. Sie haben folgende Einstellungsmöglichkeiten:
  - Aktiviert
    Das Optionskästchen wird bei allen Ports des entsprechenden VLANs aktiviert.
  - Deaktiviert
    Das Optionskästchen wird bei allen Ports entsprechenden VLANs deaktiviert.
  - Keine Änderung
    Die Tabelle bleibt unverändert.
- **Px.y**
  Legen Sie fest, über welche Ports die IPv4-Adressen des Adressbands vergeben werden sollen. Sie können nur Ports auswählen, die sich in dem entsprechenden VLAN befinden.

Hinweis

Auswirkungen auf andere Register

- Wenn Sie genau einen Port aktivieren, ist dieser Port im Register "Zuordnung Port zu IPAdresse" ausgewählt.
- Wenn Sie keinen Port oder mehr als einen Port aktivieren, ist im Register "Zuordnung Port zu IP-Adresse" die Einstellung "Nicht ausgewählt" ausgewählt.

DHCP-Optionen

Auf dieser Seite legen Sie fest, welche DHCP-Optionen der DHCP-Server unterstützt. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert.

Die DHCP-Optionen 1, 3, 6, 66 und 67 werden automatisch beim Erstellen des IPv4-Adressbands angelegt. Mit Ausnahme der DHCP-Option 1 sind die Optionen löschbar. Bei der DHCP-Option 1 wird automatisch die Subnetzmaske eingestellt, die Sie unter "DHCP-Server" für das Adressband eingegeben haben. Bei der DHCP-Option 3 können Sie über ein Optionskästchen die interne IPv4-Adresse des Geräts als DHCP-Parameter einstellen.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Optionswert**
  Geben Sie die Nummer der gewünschten DHCP-Option ein. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert. Die unterstützten DHCP-Optionen werden im folgenden Absatz aufgeführt.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.

- **Optionswert**
  Zeigt die Nummer der DHCP-Option an.
- **Beschreibung**
  Zeigt eine Beschreibung der DHCP-Option an.
- **Schnittstellen-IP verwenden**
  Wenn Sie das Optionskästchen aktivieren, wird die IPv4-Adresse als Default-Gateway verwendet, die der VLAN-IP-Adresse zugeordnet ist. Wenn das Optionskästchen deaktiviert ist, können Sie eine IPv4-Adresse eingeben.
- **Wert**
  Geben Sie den DHCP-Parameter ein, der dem DHCP-Client übergeben wird. Der Inhalt ist abhängig von der DHCP-Option.
  - DHCP-Option 3 (Default-Gateway):
    Geben Sie den DHCP-Parameter als IPv4-Adresse an, z. B. 192.168.100.2.
  - DHCP-Option 6 (DNS):
    Geben Sie den DHCP-Parameter als IPv4-Adresse an, z. B. 192.168.100.2. Sie können bis zu drei IPv4-Adressen durch Komma getrennt angeben.
  - DHCP-Option 12 (Hostname)
    Geben Sie den Hostnamen im String-Format an.
  - DHCP-Option 66 (TFTP-Server):
    Geben Sie den DHCP-Parameter als IPv4-Adresse an, z. B. 192.168.100.2.
  - DHCP-Option 67 (Bootfilename)
    Geben Sie den Namen der Bootdatei im String-Format an.

Unterstützte DHCP-Optionen

Folgende DHCP-Optionen werden unterstützt:

- Option 1
- Option 3
- Option 6
- Option 12
- Option 66
- Option 67

Relay Agent-Information

Auf dieser Seite definieren Sie, dass Geräten mit einer bestimmten Remote-ID und Circuit-ID die IPv4-Adressen aus einem bestimmten Adressband zugeordnet werden.

Wenn Sie für ein Adressband einen solchen Eintrag erstellen, reagieren die Ports des Adressbands nur auf DHCP-Anfragen über einen DHCP Relay Agent (Option 82). Sie können weitere Adressbänder für die gleichen VLAN-IP-Schnittstellen anlegen, sodass die Ports auf verschiedene Anfragen reagieren.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Remote-ID**
  Tragen Sie die Remote-ID ein.
- **Circuit-ID**
  Tragen Sie die Circuit-ID ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Remote-ID**
  Zeigt die Remote-ID an.
- **Circuit-ID**
  Zeigt die Circuit-ID an.

Statische Zuordnung

Auf dieser Seite definieren Sie, dass DHCP-Clients abhängig von ihrer Client-ID bzw. MAC-Adresse eine vorgegebene IPv4-Adresse zugeordnet wird.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Identifikationsmethode des Clients**
  Wählen Sie die Methode, nach der ein Client identifiziert wird.
  - Ethernet MAC
    Der Client wird über seine MAC-Adresse identifiziert.
  - Client-ID
    Der Client wird über eine frei definierte DHCP-Client-ID identifiziert.
- **Wert**
  Tragen Sie die MAC-Adresse (Ethernet MAC) oder die Client-ID (Client-ID) des Clients ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.

- **Identifikationsmethode**
  Zeigt an, ob der Client über seine MAC-Adresse oder die Client-ID identifiziert wird.
- **Wert**
  Zeigt die MAC-Adresse oder Client-ID des Clients an.
- **IP-Adresse**
  Legen Sie die IPv4-Adresse fest, die dem Client zugewiesen wird. Die IPv4-Adresse muss innerhalb des IPv4-Adressbands liegen.

Host-Optionen

Auf dieser Seite legen Sie fest, welche DHCP-Optionen der DHCP-Server unterstützt. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert.

Die DHCP-Optionen 1, 3, 6, 66 und 67 werden automatisch beim Erstellen des IPv4-Adressbands angelegt. Mit Ausnahme der DHCP-Option 1 sind die Optionen löschbar. Bei der DHCP-Option 1 wird automatisch die Subnetzmaske eingestellt, die Sie unter "DHCP-Server" für das Adressband eingegeben haben. Bei der DHCP-Option 3 können Sie über ein Optionskästchen die interne IPv4-Adresse des Geräts als DHCP-Parameter einstellen.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Client**
  Wählen Sie das Gerät aus, für das Sie eine DHCP-Option festlegen wollen.
- **Optionswert**
  Geben Sie die Nummer der gewünschten DHCP-Option ein. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert. Die unterstützten DHCP-Optionen werden im folgenden Absatz aufgeführt.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Optionswert**
  Zeigt die Nummer der DHCP-Option an.
- **Beschreibung**
  Zeigt eine Beschreibung der DHCP-Option an.

- **Schnittstellen-IP verwenden**
  Wenn Sie das Optionskästchen aktivieren, wird die IPv4-Adresse als Default-Gateway verwendet, die der VLAN-IP-Adresse zugeordnet ist. Wenn das Optionskästchen deaktiviert ist, können Sie eine IPv4-Adresse eingeben.
- **Wert**
  Geben Sie den DHCP-Parameter ein, der dem DHCP-Client übergeben wird. Der Inhalt ist abhängig von der DHCP-Option.
  - DHCP-Option 12 (Hostname)
    Geben Sie den Hostnamen im String-Format an.
  - DHCP-Option 66 (TFTP-Server):
    Geben Sie den DHCP-Parameter als IPv4-Adresse an, z. B. 192.168.100.2.
  - DHCP-Option 67 (Bootfilename)
    Geben Sie den Namen der Bootdatei im String-Format an.

DHCPv6

DHCPv6-Client

Auf dieser Seite legen Sie fest, ob der DHCPv6-Client neben den Konfigurationseinstellungen auch die IPv6-Adresse bezieht oder die Funktion "Präfix-Delegation" unterstützt.

Präfix-Delegation

Der DHCPv6-Server delegiert das Verteilen des IPv6-Präfixes an den DHCPv6-Client. Dazu muss eine IPv6-Schnittstelle auf dem DHCPv6-Client als PD-Client konfiguriert werden. Über den PD-Client bezieht der DHCPv6-Client ein IPv6-Präfix vom DHCPv6 Server und legt es unter einem Präfixname ab. Die IPv6-Schnittstellen des DHCPv6-Clients, die das Präfix übernehmen sollen, werden als sogenannte PD-Sub-Clients konfiguriert.

Die PD-Sub-Clients konfigurieren Sie unter "Layer 3 (IPv6)" > "DHCPv6-Client" > "DHCPv6 PD Sub-Client".

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Schnittstelle**
  Wählen Sie die IP-Schnittstelle aus, über die als DHCPv6-Client fungiert.
- **Aktivieren**
  Aktivieren oder deaktivieren Sie den DHCPv6-Client für die entsprechende Schnittstelle.
- **Modus**
  Legen Sie das Verfahren fest.
  - Zustandsbehaftet: Bezieht die IPv6-Adresse und die Konfigurationsdatei vom DHCPv6-Server
  - Zustandslos: Bezieht nur die Konfigurationsdatei vom DHCPv6-Server
  - Präfix-Delegation: Der DHCPv6-Client übernimmt das Verteilen des IPv6-Präfixes

- **Präfixname**
  Geben Sie den Präfixnamen ein, der vom DHCPv6-Router weiterverteilt wird. Nur beim Verfahren "Präfix-Delegation" editierbar. Unter diesem Namen legt der DHCPv6-Router das IPv6-Präfix des DHCPv6-Servers ab.
- **Rapid Commit**
  Wenn aktiviert, wird das Verfahren für die IPv6-Adresszuweisung verkürzt. Anstatt 4 DHCPv6-Nachrichten (SOLICIT, ADVERTISE , REQUEST, REPLY) werden nur 2 DHCPv6-Nachrichten (SOLICIT, REPLY) verwendet. Weitere Informationen zu den Nachrichten finden Sie in der RFC 3315.

DHCPv6 PD-Sub-Client

Auf dieser Seite legen Sie fest, welche IPv6-Schnittstellen des DHCPv6-Clients, den Präfix übernehmen sollen.

Dies geschieht indem ihnen als IP-Adresse der Präfixname gefolgt von einem Präfix und der gewünschten Präfixlänge zugewiesen wird. Hierbei werden die ersten Bits des konfigurierten Präfix durch die im Präfixnamen hinterlegten Werte ersetzt.

Aus Präfixnamen, Präfix und Präfixlänge bildet der PD-Sub-Client die IPv6-Adresse.

Beispiel:

Unter dem Präfixnamen "test" ist folgendes IPv6-Präfix abgelegt: 2001:DB8:1234::/48

Präfix = 2002:AF9:5678::1

Präfixlänge = 64

Aus diesen Einstellungen bildet der PD-Sub-Client die folgende IPv6-Adresse: 2001:DB8:1234::1

Beschreibung der angezeigten Werte

Die Seite enthält Folgendes:

- **Schnittstelle**
  Wählen Sie die Schnittstelle aus, die als PD-Sub-Client fungiert. Nur IP-Schnittstellen auswählbar.
- **Präfixname**
  Geben Sie den Präfixnamen ein, der vom PD-Router weiterverteilt wird. Unter diesem Namen legt der PD-Router das IPv6-Präfix des DHCPv6-Servers ab.
- **Präfix**
  Geben Sie die Netzwerkkennung des Hosts an.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.

Die Tabelle enthält folgende Spalten:

- **Schnittstellenname**
  Zeigt den Namen der Schnittstelle
- **Präfixname**
  Zeigt den Namen des Präfixes an.

- **Präfix**
  Zeigt das Präfix an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **Generierte Adresse** (nur Online verfügbar)
  Zeigt die IPv6-Adresse an, die aus Präfixnamen, Präfix und Präfixlänge gebildet wurde.

SNMP

Allgemein

Auf dieser Seite treffen Sie grundlegende Einstellungen für SNMP. Aktivieren Sie die Optionen abhängig von der Funktion, die Sie nutzen wollen.

Beschreibung

- **SNMP**
  Wählen Sie das SNMP-Protokoll aus. Folgende Einstellungen sind möglich:
  – "-" (Deaktiviert)
    SNMP deaktiviert.
  – SNMPv1/v2c/v3
    SNMPv1/v2c/v3 wird unterstützt.

    Hinweis

    Beachten Sie, dass SNMP in den Versionen 1 und 2c über keine Sicherheitsmechanismen verfügt.

  – SNMPv3
    Nur SNMPv3 wird unterstützt.
- **SNMPv1/v2 schreibgeschützt**
  Wenn Sie diese Option aktivieren, kann SNMPv1/v2c nur lesend auf die SNMP-Variablen zugreifen.

  Hinweis

  Community String

  Verwenden Sie aus Sicherheitsgründen nicht die Standardwerte "public" oder "private". Ändern Sie die Community Strings nach der Erst-Installation.

  Die empfohlene Mindestlänge für Community Strings sind 6 Zeichen.

- **SNMPv1/v2c Read Community String - Lesen**
  Tragen Sie den Community String für den Zugriff des SNMP-Protokolls ein.
- **SNMPv1/v2c Read/Write Community String - Lesen/Schreiben**
  Tragen Sie den Community String für den lesenden und schreibenden Zugriff des SNMP-Protokolls ein.

- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Senden von SNMPv1-Traps (Alarmtelegramme). Im Eintrag "Trap" legen Sie die IP-Adressen der Geräte fest, an die SNMPv1-Traps gesendet werden.
- **SNMPv1/v2c Trap Community String - Trap**
  Tragen Sie den Community String für das Senden von SNMPv1/v2c-Meldungen ein.
- **SNMPv3 Benutzermigration**
  - **Aktiviert**
    Wenn die Funktion aktiviert ist, wird eine SNMP-Engine-ID generiert, die migriert werden kann. Sie können konfigurierte SNMPv3-Benutzer auf ein anderes Gerät übertragen. Wenn Sie diese Funktion aktivieren und die Konfiguration des Geräts auf ein anderes Gerät laden, bleiben konfigurierte SNMPv3-Benutzer erhalten.
  - **Deaktiviert**
    Wenn die Funktion deaktiviert ist, wird eine gerätespezifische SNMP-Engine-ID generiert. Um die ID zu generieren, wird die Agent-MAC-Adresse des Geräts verwendet. Sie können diese SNMP-Benutzerkonfiguration nicht auf andere Geräte übertragen. Wenn Sie die Konfiguration des Geräts auf ein anderes Gerät laden, werden alle konfigurierten SNMPv3-Benutzer gelöscht.
- **SNMP-Engine-ID** (nur Online verfügbar)
  Zeigt die SNMP-Engine-ID an.

Traps

Beim Eintreten eines Alarmereignisses kann ein Gerät SNMP-Traps (Alarmtelegramme) an bis zu zehn verschiedene Management-Stationen gleichzeitig senden. Es werden nur bei solchen Ereignissen Traps gesendet, die unter "Ereignisse" festgelegt sind.

Hinweis

SNMP-Traps werden nur dann versendet, wenn Sie bei "SNMP >Allgemein" oder unter "System > Konfiguration" die Einstellung "SNMPv1 Traps" aktiviert haben.

Beschreibung

- **Trap-Empfängeradresse**
  Tragen Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) der Station ein, an die das Gerät SNMP-Traps sendet. Sie können bis zu zehn verschiedene Empfänger angeben.

Die Tabelle gliedert sich in folgende Spalten:

- **Trap-Empfängeradresse**
  Ändern Sie bei Bedarf die IP-Adressen oder den FQDN (Fully Qualified Domain Name) der Stationen.
- **Trap**
  Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps. Stationen, die eingetragen, aber nicht selektiert sind, erhalten keine SNMP-Traps.

v3-Gruppen

Sicherheitseinstellungen und Rechtevergabe

SNMP Version 3 bietet eine Rechtevergabe, Authentifizierung und Verschlüsselung auf Protokollebene. Das Security-Level und die Lese-/Schreibrechte werden gruppenspezifisch definiert. Für jedes Mitglied einer Gruppe gelten automatisch die entsprechenden Einstellungen.

Beschreibung

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Die maximale Länge beträgt 32 Zeichen.
- **Security-Level**
  Wählen Sie die Sicherheitsstufe (Authentifizierung, Verschlüsselung) aus, die für die gewählte Gruppe gültig ist. Es gibt folgende Möglichkeiten:
  - Keine Auth./keine Priv.
    Keine Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./keine Priv.
    Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./Priv.
    Authentifizierung aktiviert / Verschlüsselung aktiviert.

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt die definierten Gruppennamen an.

  Hinweis

  Der einmal vergebene Gruppenname und die Sicherheitsstufe können nach dem Anlegen nicht mehr geändert werden. Wenn Sie den Gruppennamen oder die Sicherheitsstufe ändern wollen, müssen Sie die Gruppe löschen und mit dem neuen Namen neu anlegen und neu konfigurieren.

- **Security-Level**
  Zeigt die konfigurierte Sicherheitsstufe an.
- **Lesen**
  Aktivieren oder deaktivieren Sie den Lesezugriff für die gewünschte Gruppe.

- **Schreiben**
  Aktivieren oder deaktivieren Sie den Schreibzugriff für die gewünschte Gruppe.

  Hinweis

  Damit der Schreibzugriff funktioniert, müssen Sie ebenfalls den Lesezugriff aktivieren.

- **Persistenz**
  Zeigt an, ob die Gruppe einem SNMPv3-Benutzer zugeordnet ist. Wenn die Gruppe keinem SNMPv3-Benutzer zugeordnet ist, wird kein automatisches Speichern ausgelöst und die konfigurierte Gruppe ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Die Gruppe ist einem SNMPv3-Benutzer zugeordnet.
  - Nein
    Die Gruppe ist keinem SNMPv3-Benutzer zugeordnet.

v3-Benutzer

Benutzerspezifische Sicherheitseinstellungen

Auf der Seite können Sie SNMPv3-Benutzer neu anlegen, ändern oder löschen. Das benutzerbasierte Sicherheitsmodell arbeitet mit dem Konzept des Benutzernamens, d. h. jedes Telegramm wird mit einer Benutzerkennung versehen. Diesen Benutzernamen und die betreffenden Sicherheitseinstellungen überprüfen sowohl der Absender wie auch der Empfänger.

Beschreibung

- **Benutzername**
  Geben Sie einen frei wählbaren Benutzernamen ein. Nach der Datenübernahme können Sie den Namen nicht mehr ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Benutzername**
  Zeigt die angelegten Benutzer an.
- **Gruppenname**
  Wählen Sie die Gruppe aus, die dem Benutzer zugeordnet wird.
- **Authentifizierungsprotokoll**
  Legen Sie das Authentifizierungsprotokoll fest, für das ein Passwort hinterlegt werden soll. Folgende Einstellungen gibt es:
  - Keine
  - MD5
  - SHA
- **Verschlüsselungsprotokoll**
  Legen Sie fest, ob ein Passwort zur Verschlüsselung mit dem DES-Algorithmus hinterlegt werden soll. Nur aktivierbar, wenn auch ein Authentifizierungsprotokoll ausgewählt wurde.

- **Authentifizierungspasswort**
  Geben Sie das Authentifizierungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Authentifizierungspasswort bestätigen**
  Bestätigen Sie das Passwort durch die Wiederholung der Eingabe.
- **Verschlüsselungspasswort**
  Geben Sie Ihr Verschlüsselungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Verschlüsselungspasswort bestätigen**
  Bestätigen Sie das Verschlüsselungspasswort durch die Wiederholung der Eingabe.
- **Persistenz**
  Zeigt an, ob der Benutzer einer SNMPv3-Gruppe zugeordnet ist. Wenn der Benutzer keiner SNMPv3-Gruppe zugeordnet ist, wird kein automatisches Speichern ausgelöst und der konfigurierte Benutzer ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Der Benutzer ist einer SNMPv3-Gruppe zugeordnet.
  - Nein
    Der Benutzer ist keiner SNMPv3-Gruppe zugeordnet.

Systemzeit

Um die Systemzeit des Geräts einzustellen, gibt es unterschiedliche Methoden. Es kann immer nur eine Methode aktiv sein. Wenn eine Methode aktiviert wird, dann wird automatisch die bisher aktivierte Methode deaktiviert.

Manuelle Zeiteinstellung

Manuelle Einstellung der Systemzeit

Auf dieser Seite stellen Sie das Datum und die Uhrzeit des Systems ein. Damit diese Einstellung verwendet wird, müssen Sie "Manuelle Zeiteinstellung" aktivieren.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Manuelle Zeiteinstellung**
  Aktivieren oder deaktivieren Sie die manuelle Zeiteinstellung. Wenn Sie die Option aktivieren, wird das Eingabefeld "Systemzeit" editierbar.
- **Systemzeit**
  Geben Sie Datum und Uhrzeit im Format "MM/DD/YYYY HH:MM:SS" ein.
- **PC-Zeit verwenden**
  Klicken Sie auf die Schaltfläche, um die Zeiteinstellung des PCs zu übernehmen.
- **Letzter Synchronisationszeitpunkt**
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
  Wenn keine Uhrzeitsynchronisation möglich war, enthält das Feld die Angabe "Datum/Zeit nicht eingestellt".

- **Letzter Synchronisationsmechanismus**
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde.
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm.
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. Die aktuelle Systemzeit sehen Sie oben rechts im Auswahlbereich des WBM. In dem Feld "Systemzeit" wird weiterhin die eingestellte Zeit angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.

DST-Übersicht

Umstellung der Sommerzeit

Auf dieser Seite können Sie neue Einträge für die Umstellung der Sommerzeit anlegen. Die Tabelle gibt Ihnen einen Überblick über die vorhandenen Einträge.

Beschreibung

- **DST-Nr.**
  Zeigt die Nummer des Eintrags an.
  Wenn Sie einen neuen Eintrag anlegen, wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Name**
  Zeigt den Namen des Eintrags an.
- **Jahr**
  Zeigt das Jahr an, für das der Eintrag angelegt wurde.
- **Anfangsdatum**
  Zeigt Monat, Tag und Uhrzeit für den Start der Sommerzeit an.
- **Enddatum**
  Zeigt Monat, Tag und Uhrzeit für das Ende der Sommerzeit an.

- **Regelmäßige Zeitpunkte der Umstellung**
  Bei einem Eintrag des Typs "Regel" wird die Zeitspanne angezeigt, bestehend aus Woche, Tag, Monat und Uhrzeit, in der die Sommerzeit aktiv ist. Bei einem Eintrag des Typs "Datum" wird ein "-" angezeigt.
- **Status**
  Zeigt der Status des Eintrags an:
  - Aktiviert
    Der Eintrag wurde korrekt angelegt.
  - Ungültig
    Der Eintrag wurde neu angelegt und Anfangs- und Enddatum sind identisch.
- **Typ**
  Zeigt an, wie die Umstellung der Sommerzeit erfolgt:
  - Datum
    Es ist ein festes Datum für die Umstellung der Sommerzeit eingetragen.
  - Regel
    Es ist eine Regel für die Umstellung der Sommerzeit definiert.

DST-Konfiguration

Umstellung der Sommerzeit konfigurieren

Auf dieser Seite können Sie die Einträge für die Umstellung der Sommerzeit konfigurieren. Durch die Umstellung auf Sommer- bzw. Winterzeit ist die Systemzeit für die lokale Zeitzone korrekt eingestellt. Sie können eine Regel für die Umstellung der Sommerzeit definieren oder ein festes Datum angeben.

Beschreibung

Hinweis

Der Inhalt dieser Seite ist abhängig davon, was Sie im Feld "Typ" auswählen.

Die Felder "DST-Nr.", "Typ" und "Name" werden immer angezeigt.

- **DST-Nr.**
  Wählen Sie die Nummer des Eintrags aus.
- **Typ**
  Wählen Sie aus, wie die Umstellung der Sommerzeit erfolgen soll:
  - Datum
    Sie können ein festes Datum für die Umstellung der Sommerzeit angeben. Diese Einstellung eignet sich für Regionen, in denen die Umstellung der Sommerzeit keiner Regel folgt.
  - Periodisch
    Sie können eine Regel für die Umstellung der Sommerzeit definieren. Diese Einstellung eignet sich für Regionen, in denen die Sommerzeit immer an einem bestimmten Wochentag beginnt bzw. endet.
- **Name**
  Geben Sie einen Namen für den Eintrag an. Der Name kann maximal 16 Zeichen lang sein.

Einstellungen bei der Auswahl "Datum"

Sie können ein festes Datum für den Beginn und das Ende der Sommerzeit angeben.

- **Jahr**
  Geben Sie das Jahr für die Umstellung der Sommerzeit an.
- **Anfangsdatum**
  Geben Sie folgende Werte für den Beginn der Sommerzeit an:
  - Tag
    Geben Sie den Tag an.
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
- **Enddatum**
  Geben Sie folgende Werte für das Ende der Sommerzeit an:
  - Tag
    Geben Sie den Tag an.
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.

Einstellungen bei der Auswahl "Regel"

Sie können eine Regel für die Umstellung der Sommerzeit erstellen.

- **Anfangsdatum**
  Geben Sie folgende Werte für den Beginn der Sommerzeit an:
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
  - Woche
    Geben Sie die Woche an.
    Sie können die erste bis vierte oder die letzte Woche des Monats auswählen.
  - Wochentag
    Geben Sie den Wochentag an.
- **Enddatum**
  Geben Sie folgende Werte für das Ende der Sommerzeit an:
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
  - Woche
    Geben Sie die Woche an.
    Sie können die 1. bis 4. oder die letzte Woche des Monats auswählen.
  - Wochentag
    Geben Sie den Wochentag an.

SNTP-Client

Uhrzeitsynchronisation im Netzwerk

Das SNTP (Simple Network Time Protocol) dient zur Zeitsynchronisation im Netzwerk. Die entsprechenden Telegramme werden von einem SNTP-Server im Netz versendet.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SNTP-Client**
  Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über SNTP.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom IE-Switch empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.

- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit. Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
    In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.
- **SNTP-Modus**
  Wählen Sie die Synchronisationsart aus. Folgende Synchronisierungsarten sind möglich:
  - Listen
    Bei diesem Modus ist das Gerät passiv und empfängt SNTP-Telegramme, die die Uhrzeit liefern. Einstellungen in den Eingabefeldern "SNTP-Server-Adresse" und "Port des SNTP-Servers" haben in diesem Modus keine Wirkung. In diesem Modus werden nur IPv4-Adressen unterstützt.
  - Poll
    Wenn Sie diesen Modus wählen, wird das Eingabefeld "Poll-Intervall[s]" zur weiteren Konfiguration eingeblendet. In diesem Modus werden die Einstellungen in den Eingabefeldern "SNTP-Server-Adresse" und "Port des SNTP-Servers" berücksichtigt. Bei dieser Synchronisationsart ist das Gerät aktiv und sendet eine Zeitabfrage an den SNTP-Server. In diesem Modus werden IPv4- und IPv6-Adressen unterstützt.
- **Poll-Intervall[s]**
  Geben Sie den Zeitabstand zwischen zwei Zeitanfragen ein. In diesem Feld geben Sie das Abfrageintervall in Sekunden an. Mögliche Werte sind 16 bis 16284 Sekunden.

- **SNTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SNTP-Servers ein.
- **Port des SNTP-Servers**
  Geben Sie den Port des SNTP-Servers ein.
  Folgende Ports sind möglich:
  – 123 (Standard-Port)
  – 1025 bis 36564
- **Primär**
  Bei dem SNTP-Server, den SIe zuerst anlegen, wird das Häckchen gesetzt. Wenn mehrere SNTP-Server angelegt sind, wird der primäre Server zuerst angefragt.

NTP-Client

Automatische Zeiteinstellung über NTP

Wenn die Uhrzeitsynchronisation über NTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **NTP-Client**
  Markieren Sie dieses Optionskästchen, um die automatische Zeitsynchronisation über NTP zu aktivieren.
- **Nur NTP (secure)-Client**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem gesicherten NTP-Server. Die Einstellung gilt für alle Severeinträge.
  Um den gesicherten NTP-Client zu aktivieren, konfigurieren Sie die Parameter für die Authentifizierung (Schlüssel-ID, Hash-Algorithmus, Schlüssel).
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom IE-Switch empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.

- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.
- **NTP-Serverindex**
  Wählen Sie den Index des NTP-Servers aus. Der Server mit dem kleinsten Index wird zuerst angefragt.
- **NTP-Server-Adresse**
  Geben Sie die IPv4-Adresse des NTP-Servers an.
- **Port des NTP-Servers**
  Geben Sie den Port des NTP-Servers an.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564
- **Poll-Intervall[s]**
  Legen Sie den Zeitabstand zwischen zwei Uhrzeitanfragen fest. Mögliche Werte sind 64 bis 1024 Sekunden.
- **Schlüssel-ID**
  Geben Sie die ID des Authentifizierungsschlüssels ein.

- **Hash-Algorithmus**
  Legen Sie das Format für den Authentifizierungsschlüssel fest.
- **Schlüssel**
  Geben Sie den Authentifizierungsschlüssel ein.
- **Schlüssel bestätigen**
  Geben Sie nochmals den Authentifizierungsschlüssel ein, um ihn zu bestätigen.

SIMATIC Time Client

Zeiteinstellung über SIMATIC Time Client

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den SIMATIC Time Client.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SIMATIC Time Client**
  Markieren Sie dieses Optionskästchen, um das Gerät als SIMATIC Time Client zu aktivieren.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemzeit an.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP

PTP-Client

Automatische Zeiteinstellung über PTP

Wenn die Uhrzeitsynchronisation über PTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

Folgende Geräte unterstützen die Uhrzeitsynchronisation über PTP:

- SCALANCE XR528-6M
- SCALANCE XR552-12M

Beschreibung

- **PTP-Client**
  Markieren Sie dieses Optionskästchen, um die automatische Zeitsynchronisation über PTP zu aktivieren.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die durch die Zeitsyschronisation im Netzwerk ermittelt wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat. Wenn keine Uhrzeitsynchronisation möglich war, enthält das Feld die Angabe "Datum/Zeit nicht eingestellt".
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde.
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP

- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit. Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.

NTP-Server

Gerät als NTP-Server konfigurieren

Auf dieser Seite konfigurieren Sie das Gerät als NTP-Server. Die anderen Geräte können über diesen NTP-Server die aktuelle Zeit abrufen. Damit sind die versorgten Geräte nicht auf eine Verbindung zu einem externen Zeitserver angewiesen.

Hinweis

Zeitsynchronisation

Damit das Gerät die angeschlossenen Geräte auf einen korrekten Zeitstand synchronisiert, konfigurieren Sie das Gerät auch als NTP-Client. Als NTP-Client bezieht das Gerät die präzise Zeit von einem externen Zeitserver und verteilt diese als NTP-Server an seine NTP-Clients.

Beschreibung

- **NTP-Server**
  Aktivieren oder deaktivieren Sie den Dienst NTP-Server.

  Hinweis

  Der SNTP-Modus "Listen" des SNTP-Clients und der NTP-Server können nicht gleichzeitig aktiviert sein.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **An allen Schnittstellen**
  Zeigt an, dass die Einstellungen für alle Schnittstellen der Tabelle 2 gültig sind.
- **Mithören**
  Wählen Sie in der Klappliste die Einstellung für alle Schnittstellen aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Schnittstellen der Tabelle 2 übernommen

Die Tabelle 2 gliedert sich in folgende Spalten

- **Schnittstelle**
  Über diese Schnittstelle wird die Zeit mittels NTP übermittelt.
- **Mithören**
  Wenn aktiviert, können die anderen Geräte über diese Schnittstelle die Uhrzeit abrufen.

Automatische Abmeldung

Einstellen der automatischen Abmeldung

Stellen Sie auf dieser Seite die Zeiten ein, nach denen bei Inaktivität des Benutzers automatisch eine Abmeldung vom WBM oder dem CLI erfolgt. Wenn Sie automatisch abgemeldet wurden, dann müssen Sie sich wieder neu anmelden.

Konfiguration

- **Web Based Management [s]**
  Tragen Sie in dieses Eingabefeld einen Wert von 60-3600 Sekunden ein. Wenn Sie den Wert 0 eintragen, ist die automatische Abmeldung deaktiviert.
- **CLI (TELNET, SSH, Serial) [s]**
  Tragen Sie in dieses Eingabefeld einen Wert von 60-600 Sekunden ein. Wenn Sie den Wert 0 eintragen, ist die automatische Abmeldung deaktiviert.

Taster

Der Taster "SELECT/SET" dient zum:

- Umschalten des Anzeigemodus,
- Zurücksetzen auf werkseitige Voreinstellungen,
- Definieren der Meldemaske und der LED-Anzeige.

Eine detaillierte Beschreibung der einzelnen Funktionen, die am Taster bedient werden können, finden Sie in der Betriebsanleitung des Geräts. Auf dieser Seite kann die Funktionalität des SELECT/SET-Tasters eingeschränkt oder komplett deaktiviert werden.

Beschreibungen

- **Auf Werkseinstellungen zurücksetzen**
  Aktivieren oder deaktivieren Sie die Funktion "Auf Werkseinstellungen zurücksetzen" am SELECT/SET-Taster.

  ⚠ VORSICHT

  Tasterfunktion "Auf Werkseinstellungen zurücksetzen" beim Hochlauf aktiv

  Wenn Sie diese Funktion in ihrer Projektierung deaktiviert haben, ist die Deaktivierung nur im laufenden Betrieb gültig. Bei einem Hochlauf, z.B. nach "Strom Aus", ist die Funktion bis zum Laden der Projektierung aktiv und das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät nach diesem Vorfall erst neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferungszustand versetzt

  ⚠ VORSICHT

  Zurücksetzen über WBM und CLI

  Wenn Sie die Funktion "Auf Werkseinstellungen zurücksetzen" in Ihrer Projektierung deaktiviert haben, ist die Deaktivierung nicht für den CLI-Befehl "Neustart" und für die WBM-Seite "System > Neustart" gültig. Das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät dann neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferzustand versetzt.

- **Meldemaske aktivieren**
  Aktivieren oder deaktivieren Sie die Funktion "Definieren der Meldemaske über die LED-Anzeige" am SELECT/SET-Taster.

Syslog-Client

Systemereignis-Agent

Syslog nach RFC 3164 wird für die Übermittlung von kurzen, unverschlüsselten Textmeldungen per UDP im IP-Netz verwendet. Dazu wird ein Syslog-Server benötigt.

Voraussetzungen für das Versenden der Log-Einträgen

- Die Syslog-Funktion ist im Gerät aktiviert.
- Die Syslog-Funktion für das jeweilige Ereignis ist aktiviert.
- In Ihrem Netz befindet sich ein Syslog-Server, der die Log-Einträge entgegen nimmt. Da es sich um eine UDP-Verbindung handelt, gibt es keine Rückmeldung an den Absender.
- Die IP-Adresse oder der FQDN (Fully Qualified Domain Name) des Syslog-Servers ist im Gerät eingetragen.

Beschreibung

Die Seite enthält folgende Felder:

- **Syslog-Client**
  Aktivieren oder deaktivieren Sie die Syslog-Funktion.
- **Adresse des Syslog-Servers**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Adresse des Syslog-Servers**
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers an.
- **Server-Port**
  Geben Sie den verwendeten Port des Syslog-Servers ein.

Ports

Übersicht

Portkonfiguration im Überblick

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an. Sie können auf dieser Seite keine Konfiguration vornehmen.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die konfigurierbaren Ports an. Der Eintrag ist ein Link. Wenn Sie auf den Link klicken, wird die entsprechende Konfigurationsseite geöffnet. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Port-Name**
  Zeigt den Namen des Ports.
- **Port-Typ** (Nur bei Routing)
  Zeigt den Typ des Ports an. Folgende Typen sind möglich:
  - Router-Port
  - Switch-Port VLAN Hybrid
  - Switch-Port VLAN Trunk
  - Switch-Port PVLAN Host
  - Switch-Port PVLAN Promiscuous
  - Switch-Port VLAN Access

- **Combo Port Medientyp**
  Die Werte sind nur bei Combo Ports verfügbar. Zeigt den Modus des Combo Ports an:
  - auto
  - rj45
  - sfp
- **Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Betriebszustand**
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig. Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
  - not present
    Bei modularen Geräten wird dieser Status angezeigt, wenn z. B. kein Medienmodul gesteckt ist.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - Up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - Down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.
- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsparameter des Ports an
- **MTU** (Maximum Transmission Unit)
  Zeigt die maximale Paketgröße an.
- **Negotiation**
  Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **Flow Ctrl. Type**
  Gibt an, ob für den Port die Flusskontrolle aktiviert oder deaktiviert ist.
- **Flow Ctrl.**
  Gibt an, ob bei diesem Port die Flusskontrolle arbeitet.

- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Geblockt durch**
  Zeigt an, warum sich der Port im Zustand "blocked" befindet:
  - -
    Die Analyse des Port-Status wird nicht unterstützt.
  - Ringredundanz
    Der Port gehört zu einem Redundanzmanager. Wenn sich der Redundanzmanager im Status "Passive" befindet, ist einer der Ring-Ports im Zustand "blocking".
  - Spanning Tree
    Der Port hat im Spanning Tree den Status "Discarding". Der Port ist Teil eines Spanning Trees, jedoch liegt er auf einem redundanten Pfad und ist für den Datenverkehr deaktiviert.
  - Loop Detection
    Es wurde ein Loop erkannt und als Reaktion auf einen Loop wurde für den Port der Zustand "disable" konfiguriert.
  - Link Check
    Es wurde eine Störungen bei einer optischen Übertragungsstrecke erkannt und als Reaktion wurde für den Port der Zustand "disable" konfiguriert.
  - Link Aggregation-Mitglied
    Der Port ist Teil einer Link Aggregation und wurde durch LACP deaktiviert.
  - Link Aggregation (LoopD)
    Der Port ist Teil einer Link Aggregation. Es wurde ein Loop erkannt und als Reaktion auf einen Loop wurde für die Link Aggregation der Zustand "disable" konfiguriert.
  - Link Aggregation (STP)
    Der Port ist Teil einer Link Aggregation. Die Link Aggregation wurde durch Spanning Tree in den Status "Discarding" geschaltet.
  - Admin down
    Für den Port ist der Status "disabled" konfiguriert, siehe "System > Ports > Konfiguration".
  - Link down
    Für den Port ist der Status "enabled" konfiguriert, aber es besteht keine Verbindung, siehe "System > Ports > Konfiguration".
  - Power down
    Für den Port ist der Status "Link down" konfiguriert, siehe "System > Ports > Configuration".
  - Standby
    Auf dem Gerät ist die Standby-Redundanz aktiviert. Der Port ist ein Standby-Port mit dem Status "Passive".

Abweichende Anzeige der Übertragungsparameter bei Combo Ports

Im Verbindungsstatus „down" entsprechen die angezeigten Übertragungsparameter nicht den eigentlichen Werten des Combo Ports. Im Verbindungsstatus „up" werden die korrekten Werte angezeigt.

Ausgangslage

Es ist ein Stecktransceiver am Combo Port gesteckt, mit folgenden Einstellungen:

- Combo Port Medientyp: auto
- Status: enabled
- Link: down

Anzeige der Übertragungsparameter

Bei 100 MBit/s-Stecktransceivern

- Ist-Verhalten: Mode: 1G HD
- Soll-Verhalten: Mode: 100M FD

Bei 1 GBit/s-Stecktransceivern

- Ist-Verhalten: Mode: 1G HD
- Soll-Verhalten: Mode: 1G FD

Konfiguration

Ports konfigurieren

Mit dieser Seite können Sie alle Ports des Geräts konfigurieren.

Beschreibung

Die Tabelle gliedert sich in folgende Zeilen:

- **Port**
  Wählen Sie den zu konfigurierenden Port aus. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Status**
  Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled
    Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled
    Der Port ist ausgeschaltet, aber die Verbindung besteht noch.
  - Link down
    Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**
  Tragen Sie hier einen Namen für den Port ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.

- **Übertragungsmodus** (Nur Online verfügbar)
  Wählen Sie die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports aus. Wenn Sie die Betriebsart auf "Auto Negotiation" stellen, werden diese Parameter automatisch mit dem angeschlossenen Endgerät ausgehandelt. Dieses muss sich hierzu ebenfalls in der Betriebsart "Auto Negotiation" befinden.

  Hinweis

  Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

  Hinweis

  "Übertragungsmodus" bei Combo Ports

  Um den "Übertragungsmodus" eines Combo Ports einstellen zu können, ändern Sie den "Combo Port Medientyp" in "rj45". Wenn für den "Combo Port Medientyp" "auto" eingestellt ist und der RJ45-Port verwendet wird, können Sie den "Übertragungsmodus" nicht einstellen.

- **Akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Die Übertragungsgeschwindigkeit kann 10 Mbit/s, 100 Mbit/s, 1000 Mbit/s oder 10Gbit/s betragen. Als Übertragungsverfahren können Vollduplex (FD) oder Halbduplex (HD) konfiguriert werden.
- **Negotiation**
  Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.
- **Flow Ctrl. Type**
  Aktivieren oder deaktivieren Sie die Flusskontrolle für den Port.

  Hinweis

  Ein-/Ausschalten der Flusskontrolle bei Auto Negotiation

  Die Flusskontrolle kann nur bei ausgeschalteter Funktion "Auto Negotiation" aktiviert oder deaktiviert werden. Die Funktion kann danach wieder aktiviert werden.

- **Flow Ctrl.**
  Zeigt an, ob bei diesem Port die Flusskontrolle arbeitet.
- **MTU**
  Tragen Sie die Paketgröße ein.

- **Port-Typ** (Nur bei Routing)
  Wählen Sie die Art des Ports aus:
  - Router-Port
    Der Port ist eine Layer 3-Schnittstelle. Er unterstützt keine Layer 2-Funktionen.
  - Switch-Port VLAN Hybrid
    Der Port sendet getaggte und ungetaggte Telegramme. Er ist nicht automatisch Mitglied eines VLANs.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.
  - Switch-Port PVLAN Host
    Host-Ports gehören zu einem Secondary PVLAN. Schließen Sie an Host-Ports Geräte an, die nur mit bestimmten Geräten des PVLANs kommunizieren sollen.
  - Switch-Port PVLAN Promiscuous
    Promiscuous-Ports gehört zu einem Primary PVLAN. Schließen Sie an Promiscuous-Ports Geräte an, die mit allen anderen Geräten des PVLANs kommunizieren sollen.
  - Switch-Port VLAN Access
    Access-Ports gehört zu einem Provider-Switch, der die Funktion Q-in-Q VLAN-Tunnel unterstützt. Schließen Sie an Access-Ports ein Customer-Netzwerk an.

- **Combo Port Medientyp**
  Legen Sie den Modus des Combo Ports fest:
  - auto
    Wenn Sie diesen Modus wählen, hat der Stecktransceiver-Port Priorität. Sobald ein Stecktransceiver gesteckt wird, wird eine bestehende Verbindung am festen RJ45-Port getrennt. Wenn kein Stecktransceiver gesteckt ist, kann eine Verbindung über den festen RJ45-Port hergestellt werden.
  - rj45
    Wenn Sie diesen Modus wählen, wird der feste RJ45-Port verwendet, unabhängig vom Stecktransceiver-Port. Wenn ein Stecktransceiver gesteckt ist, wird er deaktiviert und stromlos geschaltet.
  - sfp
    Wenn Sie diesen Modus wählen, wird der Stecktransceiver-Port verwendet, unabhängig von dem festen RJ45-Port. Wenn eine RJ45-Verbindung besteht, wird diese getrennt, da der RJ45-Port stromlos geschaltet wird.

  Die Werkseinstellung für die Combo Ports ist der Modus "auto".

  Hinweis

  Automatische Anpassung durch PROFINET-Konfiguration

  Beim Aufbau einer PROFINET-Verbindung wird auch die Einstellung des Combo Port Medientyps automatisch angepasst:
  - Wenn eine Stecktransceiver konfiguriert ist, wird der Combo Port Medientyp auf "sfp" gesetzt.
  - Wenn der feste RJ45-Port konfiguriert ist, wird der Combo Port Medientyp auf "rj45" gesetzt.

  Damit die automatische Anpassung durchgefürt werden kann, muss der Combo Port Medientyp auf "auto" gesetzt sein. Konfigurieren Sie den Combo Port Medientyp entsprechend üer das WBM oder CLI.

- **Betriebszustand**
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig. Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
  - not present
    Bei modularen Geräten wird dieser Status angezeigt, wenn z. B. kein Medienmodul gesteckt ist.

- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk an. Es gibt folgende Möglichkeiten:
  - Up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - Down
    Die Verbindung ist unterbrochen, weil z. B. das angeschlossene Gerät ausgeschaltet ist.
- **Geblockt durch**
  Zeigt an, warum sich der Port im Zustand "blocked" befindet:
  - -
    Die Analyse des Port-Status wird nicht unterstützt.
  - Ringredundanz
    Der Port gehört zu einem Redundanzmanager. Wenn sich der Redundanzmanager im Status "Passive" befindet, ist einer der Ring-Ports im Zustand "blocking".
  - Spanning Tree
    Der Port hat im Spanning Tree den Status "Discarding". Der Port ist Teil eines Spanning Trees, jedoch liegt er auf einem redundanten Pfad und ist für den Datenverkehr deaktiviert.
  - Loop Detection
    Es wurde ein Loop erkannt und als Reaktion auf einen Loop wurde für den Port der Zustand "disable" konfiguriert.
  - Link Check
    Es wurde eine Störungen bei einer optischen Übertragungsstrecke erkannt und als Reaktion wurde für den Port der Zustand "disable" konfiguriert.
  - Link Aggregation-Mitglied
    Der Port ist Teil einer Link Aggregation und wurde durch LACP deaktiviert.
  - Link Aggregation (LoopD)
    Der Port ist Teil einer Link Aggregation. Es wurde ein Loop erkannt und als Reaktion auf einen Loop wurde für die Link Aggregation der Zustand "disable" konfiguriert.
  - Link Aggregation (STP)
    Der Port ist Teil einer Link Aggregation. Die Link Aggregation wurde durch Spanning Tree in den Status "Discarding" geschaltet.
  - Admin down
    Für den Port ist der Status "disabled" konfiguriert, siehe "System > Ports > Konfiguration".
  - Link Down
    Für den Port ist der Status "enabled" konfiguriert, aber es besteht keine Verbindung, siehe "System > Ports > Konfiguration".
  - Power Down
    Für den Port ist der Status "Link down" konfiguriert, siehe "System > Ports > Konfiguration".
  - Standby
    Auf dem Gerät ist die Standby-Redundanz aktiviert. Der Port ist ein Standby-Port mit dem Status "Passive".

Veränderung der Port-Konfiguration

Klicken Sie in das entsprechende Feld, um die Konfiguration zu ändern.

Hinweis

Optische Ports arbeiten immer mit dem Übertragungsverfahren Vollduplex und mit maximaler Übertragungsgeschwindigkeit. Deshalb können Sie bei optischen Ports folgende Einstellungen nicht vornehmen:

- Automatische Konfiguration
- Übertragungsgeschwindigkeit
- Übertragungsverfahren

Hinweis

Das Gerät verhindert oder reduziert bei Überlastung eines Ports durch verschiedene Automatismen die Rückwirkung auf andere Ports und Prioritätsklassen (Class of Service). Dies kann auch bei aktivierter Flusskontrolle dazu führen, dass Telegramme verworfen werden.

Port-Überlastungen treten auf, wenn das Gerät mehr Telegramme empfängt, als es senden kann, z.B. infolge unterschiedlicher Übertragungsgeschwindigkeiten.

Fehlerkontrolle

Spannungsversorgung

Einstellungen zur Überwachung der Spannungsversorgung

Konfigurieren Sie, ob die Spannungsversorgung durch das Meldesystem überwacht werden soll. Je nach Hardware-Variante gibt es ein oder zwei Spannungsanschlüsse (Versorgung 1 / Versorgung 2). Bei redundanter Spannungsversorgung konfigurieren Sie die Überwachung für jede einzelne Zuleitung getrennt.

Wenn an einem überwachten Anschluss (Versorgung1 oder Versorgung 2) keine oder eine zu geringe Spannung anliegt, wird ein Fehler durch das Meldesystem signalisiert.

Hinweis

Die zulässigen Betriebsspannungsgrenzen entnehmen Sie der Kompaktbetriebsanleitung des Geräts.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät und kann abhängig von der Konfiguration einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Einstellungen

- **Versorgung 1**
  Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 1.
- **Versorgung 2**
  Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 2.

Link Change

Konfiguration der Fehlerüberwachung von Zustandsänderungen bei Verbindungen

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

Bei aktivierter Verbindungsüberwachung wird ein Fehler signalisiert,

- wenn an einem Port ein Link vorhanden sein soll und dieser fehlt.
- oder wenn an dem Port kein Link vorhanden sein soll und ein Link erkannt wird.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Möglichkeiten:
  - "-" (Deaktiviert)
  - Up
  - Down
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Port**
  Zeigt die verfügbaren Ports an.
- **Einstellung**
  Wählen Sie die Einstellung aus. Sie haben folgende Möglichkeiten:
  - Up
    Die Fehlerbehandlung wird beim Übergang in den aktiven Zustand des Ports ausgelöst.
    (Von "Link down" nach "Link up")
  - Down
    Die Fehlerbehandlung wird beim Übergang in den inaktiven Zustand des Ports ausgelöst.
    (Von "Link up" nach "Link down")
  - "-" (Deaktiviert)
    Die Fehlerbehandlung wird nicht ausgelöst.

Redundanz

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer redundanten Verbindung eine Fehlermeldung ausgelöst wird.

Einstellung

- **Redundanzverlust (nur HRP)**
  Aktivieren oder deaktivieren Sie die Verbindungsüberwachung. Wenn die Redundanz der Verbindung verloren geht, wird ein Fehler signalisiert.

PROFINET

Einstellungen für PROFINET

Auf dieser Seite konfigurieren Sie den Modus von PROFINET.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **PNIO-Gerätediagnose**
  Zeigt an, ob PNIO aktiviert ("On") oder deaktiviert ("Off") ist.
- **PNIO-Gerätediagnose beim nächsten Hochlauf**
  Stellen Sie ein, ob PNIO nach dem nächsten Neustart des Geräts aktiviert ("On") oder deaktiviert ("Off") sein soll.

  Hinweis

  PNIO und EtherNet/IP

  Wenn PNIO eingeschaltet wird, wird EtherNet/IP ausgeschaltet. Das Umschalten von PNIO und EtherNet/IP hat keine Auswirkungen auf DCP.

  Hinweis

  PNIO AR-Status

  Wenn eine PROFINET-Verbindung aufgebaut ist, d. h. der PNIO AR-Status "Online" ist, können Sie PNIO nicht deaktivieren.

- **PNIO AR-Status**
  Dieses Feld zeigt den Status des PROFINET IO-Verbindungsverhältnisses an, d.h. ob das Gerät mit einem PROFINET IO-Controller "Online" oder "Offline" verbunden ist.
  Online bedeutet hierbei, dass eine Verbindung zu einem PROFINET IO-Controller besteht, dass dieser seine Konfigurationsdaten auf das Gerät geladen hat und das Gerät Statusdaten zum PROFINET IO-Controller senden kann. In diesem Zustand, der auch "in Data exchange" genannt wird, sind die Parameter, die über den PROFINET IO-Controller eingestellt werden, nicht konfigurierbar.
- **PNIO-Gerätename**
  In diesem Feld erscheint der PROFINET IO-Gerätenamen gemäß der Projektierung in der HW-Konfig von STEP 7.
- **Neustart mit PROFINET Defaults**
  Klicken Sie auf diese Schaltfläche, um die Defaulteinstellungen des PROFINET IO-Profils wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. In der Dialogbox werden die Einstellungen angezeigt, die speziell auf den Betrieb mit dem Protokoll PROFINET abgestimmt sind.

ACHTUNG
Verlust der IP-Adresse Durch das Zurücksetzen der Einstellungen auf die Defaulteinstellungen eines Profils geht auch die IP-Adresse verloren. Das Gerät ist danach nur über die serielle Schnittstelle, das Primary Setup Tool oder über DHCP ansprechbar. Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

- **PNIO-Datenaustausch erlauben** (nur beim SCALANCE XB-200)
  Aktivieren oder deaktivieren Sie den PNIO-Datenaustausch
- **PNIO-Datenaustausch simulieren** (nur beim SCALANCE XB-200)
  Aktivieren oder deaktivieren Sie die Simulation des PNIO-Datenaustausches.

EtherNet/IP

EtherNet Industrial Protocol (EtherNet/IP)

Auf dieser Seite konfigurieren Sie den Modus von EtherNet/IP.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **EtherNet/IP-Gerätediagnose**
  Zeigt an, ob EtherNet/IP aktiviert ("On") oder deaktiviert ("Off") ist.
- **EtherNet/IP-Gerätediagnose beim nächsten Hochlauf**
  Stellen Sie ein, ob EtherNet/IP nach dem nächsten Neustart des Geräts aktiviert ("On") oder deaktiviert ("Off") sein soll.

  Hinweis

  EtherNet/IP und PROFINET

  Wenn EtherNet/IP eingeschaltet wird, wird PROFINET ausgeschaltet. Das Umschalten von EtherNet/IP und PROFINET hat keine Auswirkungen auf DCP.

  Hinweis

  PROFINET AR Status

  Wenn eine PROFINET-Verbindung aufgebaut ist, d. h. der PROFINET AR-Status "Online" ist, können Sie EtherNet/IP nicht aktivieren.

- **Neustart mit EtherNet/IP Defaults**
  Klicken Sie auf diese Schaltfläche, um die Defaulteinstellungen des EtherNet/IP-Profils wiederherzustellen und das Gerät neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. In der Dialogbox werden die Einstellungen angezeigt, die speziell auf den Betrieb mit dem Protokoll EtherNet/IP abgestimmt sind.

  ACHTUNG

  Verlust der IP-Adresse

  Durch das Zurücksetzen der Einstellungen auf die Defaulteinstellungen eines Profils geht auch die IP-Adresse verloren. Das Gerät ist danach nur über die serielle Schnittstelle, das Primary Setup Tool oder über DHCP ansprechbar.

  Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

PLUG

PLUG-Konfiguration

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.
Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt.

Informationen über die Konfiguration des C-PLUG / KEY-PLUG

Diese Seite liefert Detailinformationen über die Konfiguration, die im C-PLUG bzw. KEY-PLUG abgelegt ist. Darüber hinaus gibt es die Möglichkeit, den PLUG auf "Factory Default" zurückzusetzen oder mit einem neuen Inhalt zu versehen.

Hinweis

Die Aktion wird erst dann durchgeführt, wenn Sie auf die Schaltfläche "Einstellungen übernehmen" klicken.

Die Aktion kann nicht rückgängig gemacht werden.

Wenn Sie sich nach der Auswahl gegen die Ausführung entscheiden, dann klicken Sie auf die Schaltfläche "Aktualisieren". Dadurch werden die Daten dieser Seite aus dem Gerät neu ausgelesen und Ihre Auswahl wird aufgehoben.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da die sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Einstellungen

Die Tabelle gliedert sich in folgende Zeilen:

- **Status**
  Zeigt den Status des PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Es ist ein PLUG mit einer gültigen und passenden Konfiguration im Gerät vorhanden.
  - NOT ACCEPTED
    Ungültige bzw. inkompatible Konfiguration auf dem gesteckten PLUG.
  - NOT PRESENT
    Im Gerät ist kein C-PLUG oder KEY-PLUG gesteckt.
  - FACTORY
    PLUG ist gesteckt und enthält keine Konfiguration. Dieser Status wird auch angezeigt, wenn der PLUG im Betrieb formatiert wurde.
  - MISSING
    Es ist kein PLUG gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
- **Gerätegruppe**
  Zeigt an, von welcher SIMATIC NET-Produktlinie der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Gerätetyp**
  Zeigt den Gerätetyp innerhalb der Produktlinie an, von dem der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Version der Konfiguration**
  Die Version der Konfigurationsstruktur. Diese Angabe betrifft die vom Gerät unterstützten Konfigurationsmöglichkeiten und hat nichts mit der konkreten Hardware-Konfiguration zu tun. Diese Revisionsangabe ändert sich also nicht, wenn Sie Zusatzkomponenten (z.B. Module bzw. Extender) hinzufügen oder entfernen, sie kann sich aber ändern, wenn Sie ein Firmware-Update durchführen.
- **Dateisystem**
  Zeigt den Typ des Dateisystems an, das auf dem PLUG vorhanden ist.

ACHTUNG
Neues Dateisystem UBI Ab Firmware-Version 3.0 ist UBI das Standard-Dateisystem für den C-PLUG bzw. KEY-PLUG. Wird in einem solchen Gerät ein C-PLUG mit dem bisherigen Dateisystem IECP erkannt, wird dieser C-PLUG für das Dateisystem UBI formatiert und die Daten werden neu auf den C-PLUG geschrieben. Die Änderung des Dateisystems erfolgt auch nach einem Firmware-Update auf V3.0. Ein Downgrade auf die Vorgängerversion der entsprechenden Firmware ist dann problematisch. Die Firmware kann den C-PLUG bzw. KEY-PLUG weder lesen noch schreiben und auch ein "PLUG auf Werkseinstellungen zurücksetzen" ist nicht möglich.

- **Verfügbarer Speicherplatz [Byte]**
  Zeigt die maximale Speicherkapazität des Dateisystems an, das auf dem PLUG vorhanden ist.

- **Belegter Speicherplatz [Byte]**
  Zeigt den belegten Speicherplatz im Dateisystem des PLUG an.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.
- **Firmware auf PLUG**
  Wenn die Funktion aktiviert ist, wird die Firmware auf dem PLUG abgespeichert. Damit können mit dem PLUG automatische Firmware-Updates/Downgrades durchgeführt werden. In dem Feld "Info" wird angezeigt, ob die Firmware auf dem PLUG gespeichert ist oder nicht.
- **PLUG ändern**
  Wählen Sie die gewünschte Einstellung aus. Sie haben folgende Möglichkeiten, um die Konfiguration auf dem C-PLUG bzw. KEY-PLUG zu ändern:
  - Aktuelle Konfiguration auf den PLUG schreiben
    Diese Option ist nur verfügbar, wenn der Status des PLUG "NOT ACCEPTED" oder "FACTORY" ist.
    Die im internen Flash-Speicher des Gerätes vorhandene Konfiguration wird auf den PLUG kopiert.
  - PLUG auf Werkseinstellungen zurücksetzen
    Löscht alle Daten vom PLUG und führt eine Low-Level-Formatierung durch.

PLUG-Lizenz

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.
Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt.

Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurückgesetzt.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Informationen über die Lizenz des KEY-PLUG

Ein C-PLUG kann nur die Konfiguration eines Geräts speichern. Ein KEY-PLUG enthält zusätzlich zur Konfiguration eine Lizenz, die bestimmte Funktionen Ihres SIMATIC NET-Geräts freischaltet.

Diese Seite liefert Detailinformationen über die Lizenz auf dem KEY-PLUG.

Beschreibung der angezeigten Felder

- **Status**
  Zeigt den Status des KEY-PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Der im Gerät vorhandene KEY-PLUG enthält eine passende und gültigen Lizenz.
  - NOT ACCEPTED
    Die Lizenz des gesteckten KEY-PLUG ist nicht gültig.
  - NOT PRESENT
    Im Gerät ist kein KEY-PLUG gesteckt.
  - MISSING
    Es ist kein KEY-PLUG oder ein C-PLUG mit dem Status "FACTORY" gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
  - WRONG
    Der gesteckte KEY-PLUG passt nicht zum Gerät.
  - UNKNOWN
    Unbekannter Inhalt des KEY-PLUG.
  - DEFECTIVE
    Der Inhalt des KEY-PLUG ist fehlerhaft.
- **Bestellnummer**
  Zeigt die Bestellnummer des KEY-PLUG an. Es gibt den KEY-PLUG für unterschiedliche Funktionserweiterungen und für verschiedene Zielsysteme.

- **Seriennummer**
  Zeigt die Seriennummer des KEY-PLUG.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den KEY-PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Bestellnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.

Hinweis

Beim Speichern der Konfiguration wird die Information mitgespeichert, ob zu diesem Zeitpunkt ein KEY-PLUG im Gerät gesteckt war. Diese Konfiguration ist dann auch nur lauffähig, wenn ein KEY-PLUG mit der gleichen Artikelnummer / Lizenz gesteckt ist.

Ping

Erreichbarkeit einer Adresse in einem IP-Netzwerk

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mit der Ping-Funktion können Sie überprüfen, ob eine bestimmte Adresse im Netzwerk erreichbar ist.

Beschreibung

Die Seite enthält folgende Felder:

- **Zieladresse**
  Geben Sie die IPv4-, IPv6-Adresse oder den FQDN (Fully Qualified Domain Name) des Geräts ein.
- **Wiederholen**
  Tragen Sie die Anzahl der Ping-Anforderungen ein.
- **DNS-Auflösung**
  Wählen Sie aus in welchen IP-Adress-Typ ein eingegebener FQDN aufgelöst werden soll.
  - Auto
    In diesem Modus wird der IP-Adress-Typ automatisch gewählt.
  - IPv4
    Der eingegebene FQDN wird in einer IPv4-Adresse aufgelöst.
  - IPv6
    Der eingegebene FQDN wird in einer IPv6-Adresse aufgelöst.

- **Ausgehende Schnittstelle für IPv6**
  Diese Auswahl wird nur benötigt, wenn die Zieladresse eine Multicast- oder eine Link-lokale Adresse ist.
  – "-" (Werkseinstellung)
  – Wählen Sie die entsprechende IPv6-Schnittstelle aus.
- **Ping**
  Klicken Sie diese Schaltfläche, um die Ping-Funktion zu starten.
- **Ping-Ausgabe**
  Dieses Feld zeigt die Ausgabe der Ping-Funktion an.
- **Leeren**
  Klicken Sie diese Schaltfläche, um die Ping-Ausgabe zu leeren.

DCP Discovery

Geräten suchen, die über die ausgewählte Schnittstelle erreichbar sind

Auf dieser Seite können Sie eine Schnittstelle auswählen und nach den Geräten suchen, die über die Schnittstelle erreichbar sind. Die erreichbaren Geräte werden in einer Tabelle aufgelistet. In der Tabelle können Sie die Netzwerkparameter der Geräte überprüfen und anpassen. Zum Identifizieren und zum Konfigurieren der Geräte wird das Discovery Configuration Protocol (DCP) verwendet.

Hinweis

DCP Discovery

Die Funktion ist nur in dem mit der TIA-Schnittstelle assoziierten VLAN verfügbar. Die TIA-Schnittstelle konfigurieren Sie unter "Layer 3 > Subnetze > Konfiguration".

Voraussetzung:

Um die Netzwerkparameter anzupassen, benötigt DCP Schreibrechte auf dem Gerät. Wenn der Zugriff schreibgeschützt ist, sind die Netzwerkparameter nicht konfigurierbar.

Auf den SCALANCE-Geräten konfigurieren Sie den Zugriff konfigurieren unter "System > Konfiguration".

Beschreibung der angezeigten Werte

- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus.
- **Durchsuchen**
  Startet die Suche nach Geräten, die über die gewählte Schnittstelle erreichbar sind.
  Nach dem Abschluss der Suche werden die erreichbaren Geräte in der Tabelle aufgelistet.
  Die Tabelle ist auf 100 Einträge begrenzt.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt den Port an, über den das Gerät erreichbar ist.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Geräts an.
- **Gerätetyp**
  Zeigt an, zu welcher Produktlinie bzw. Produktgruppe das Gerät gehört.
- **Gerätename**
  Falls das Gerät diese Funktion unterstützt, können Sie dem Gerät einen PROFINET-Gerätenamen zuweisen.
- **IP-Adresse**
  Passen Sie bei Bedarf die IPv4-Adresse des Geräts an.
  Die IPv4-Adresse sollte innerhalb ihres Netzwerks eindeutig sein und zum Netzwerk passen. Die IPv4-Adresse 0.0.0.0 bedeutet, dass noch keine IPv4-Adresse eingestellt ist.
- **Subnetzmaske**
  Passen Sie bei Bedarf die Subnetzmaske des Geräts an.
- **Gateway-Adresse**
  Geben Sie bei Bedarf die IPv4-Adresse des Gateways an.
- **Status Gerätename**
  - Discoverd: Der eingestellte Gerätename wird verwendet.
  - Configured: Dem Gerät wurde ein neuer Gerätename zugewiesen.
- **Status IP-Adresse.**
  - Discovered/IP: Das Gerät verwendet eine statische IPv4-Adresse.
  - Discovered/DHCP: Das Gerät hat die IPv4-Adresse von einem DHCP-Server bezogen.
  - Configured: Dem Gerät wurde eine neue IPv4-Adresse zugewiesen.
- **Timeout**
  Legen Sie die Zeitdauer für das Blinken fest. Wenn die Zeit abgelaufen ist, wird das Blinken beendet.
- **Blinken**
  Lässt die Port-LEDs des ausgewählten Geräts blinken.

PoE

Allgemein

Hinweis

Der Menüpunkt "PoE" wird nur bei Geräten angezeigt, die PoE unterstützen

Einstellungen für Power over Ethernet (PoE)

Auf dieser Seite sehen Sie Informationen zu der Leistung, die der IE-Switch über PoE liefert.

Der SCALANCE X-500 stellt ein PSE (Power Sourcing Equipment) dar. Beim SCALANCE XM400 wird jede Gruppe von vier PoE-fähigen Ports als ein PSE bezeichnet. Die angezeigten Werte gelten jeweils nur für das entsprechende PSE.

Beschreibung der angezeigten Felder

- **PSE** (nur lesbar)
  Zeigt die Nummer des PSE an.
- **Maximale Leistung [W]** (nur lesbar)
  Maximale Leistung, die ein PSE für die Versorgung von PoE-Geräten zur Verfügung stellt. Der Wert „Maximale Leistung" ist beim SCALANCE XM-400 einstellbar.
- **Zugeteilte Leistung [W]** (nur lesbar)
  Summe der durch die PoE-Geräte entsprechend der "Classification" reservierten Leistung.
- **Genutzte Leistung [W]** (nur lesbar)
  Summe der von den Endgeräten genutzten Leistung.
- **Schwellenwert für Leistung [%]**
  Wenn die von den Endgeräten verbrauchte Leistung größer ist als der hier angegebene Prozentanteil, wird ein Event ausgelöst.

Power over Ethernet bei SCALANCE XM-400

Bei SCALANCE XM-400 können Sie die Funktion „Power over Ethernet" über den Port Extender PE408PoE nutzen.

PoE-Spannungsversorgung

Der Anschluss der PoE-Spannungsversorgung erfolgt extern. Pro Port Extender PE408PoE können Sie 2 PoE-Spannungsversorgungen anschließen. Jeder PE408PoE verfügt daher über 2 PSEs (Power Sourcing Equipment) mit jeweils 4 Ports.

Nummerierung der PSEs

Um die PSEs bei der Projektierung unterscheiden zu können, sind sie nummeriert:

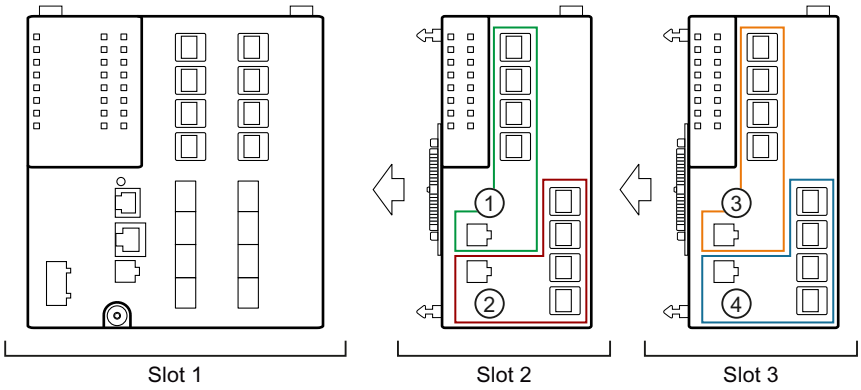

Bild 1-86 SCALANCE XM-400 mit 2 Port Extendern PE408PoE

Wenn ein PE408PoE in Slot 2 gesteckt ist, haben die dazugehörigen PSEs die Indizes 1 und 2. Wenn ein PE408PoE in Slot 3 gesteckt ist, haben die dazugehörigen PSEs die Indizes 3 und 4.

Die Nummerierung der PSEs ist an die Slots gebunden. Wenn in Slot 2 ein Port Extender ohne PoE gesteckt ist und in Slot 3 ein PE408PoE, dann haben die PSEs in Slot 3 weiterhin die Indizes 3 und 4.

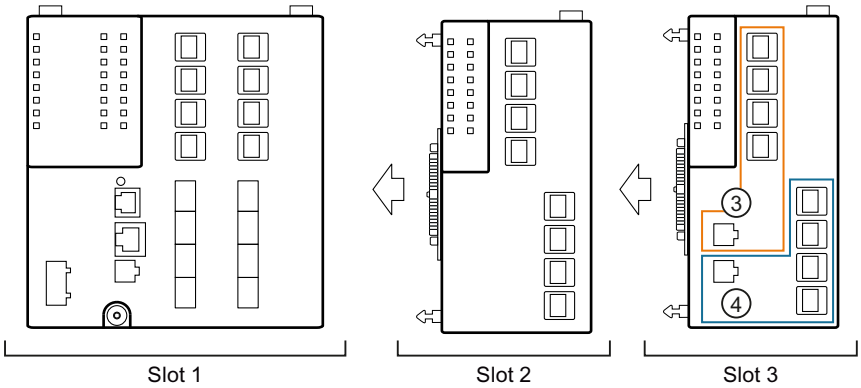

Bild 1-87 SCALANCE XM-400 mit 2 Port Extendern (1 Port Extender ohne PoE und 1 PE408PoE)

Bei einem SCALANCE XM416-4C, mit dem Sie nur einen Port Extender verbinden können, sind die Indizes der PSEs entsprechend 1 und 2.

Power over Ethernet bei SCALANCE XR528-6M und SCALANCE XR552-12M

Bei SCALANCE XR528-6M und SCALANCE XR552-12M können nur die Modulschächte 1 bis 3 mit PoE-Modulen bestückt werden.

Port

Einstellungen für die Ports

Für jeden einzelnen PoE-Port können Sie festlegen, ob eine Spannungsversorgung über Ethernet erfolgen soll. Außerdem können Sie für jeden angeschlossenen Verbraucher eine Priorität festlegen. Geräte, für die eine hohe Priorität festgelegt wurde, werden im Bedarfsfall gegenüber anderen bei der Spannungsversorgung bevorzugt. Auf dieser Seite sehen Sie Detailinformationen zu den einzelnen PoE-Ports.

Beschreibung der angezeigten Felder

Die Seite enthält zwei Tabellen. In der Tabelle 1 können Sie Einstellungen vornehmen und diese allen Ports gleichzeitig zuweisen. In der Tabelle 2 können Sie für jeden Port unterschiedliche Einstellungen vornehmen.

Die Tabelle 1 gliedert sich in folgende Spalten:

- Port
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- Einstellung
  Wählen Sie die gewünschte Einstellung aus. Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- Priorität
  Wählen Sie die gewünschte Priorität der Ports aus. Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert
- Typ
  Geben Sie eine Zeichenkette ein, die das angeschlossene Gerät näher beschreibt. Die maximale Länge beträgt 255 Zeichen.
- Benutzerdefinierte maximale Leistung verwenden
  Wählen Sie, ob die benutzerdefinierte maximale Leistung verwendet werden soll. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- Benutzerdefinierte maximale Leistung [W]
  Geben Sie die maximale Leistung an, die ein Port für die Versorgung eines angeschlossenen Geräts zur Verfügung stellt. Dieser Wert wird nur berücksichtigt, wenn das dazugehörige Optionskästchen "Benutzerdefinierte maximale Leistung verwenden" aktiviert ist. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Port
  Zeigt die konfigurierbaren PoE-Ports an. Der Port setzt sich aus der Portnummer und der Slotnummer zusammen, z. B. Port 0.1 ist Slot 0, Port 1.
- Einstellung
  Schalten Sie die PoE-Spannungsversorgung für diesen Port frei oder unterbrechen Sie sie.
- Priorität
  Wählen Sie die Priorität, mit der dieser Port bei der Spannungsversorgung berücksichtigt wird. Es gibt folgende Einstellungsmöglichkeiten mit aufsteigender Relevanz:
  - Niedrig
  - Hoch
  - Kritisch

  Wenn die Leistung der angeschlossenen Spannungsversorgung nicht ausreicht, um alle angeschlossenen Geräte zu versorgen, werden Geräte mit einer höheren Priorität vorrangig versorgt. Ist für zwei Ports die gleiche Priorität vorgegeben, wird im Bedarfsfall der Port mit der niedrigeren Nummer bevorzugt.
- Typ
  Hier können Sie eine Zeichenkette eingeben, die das angeschlossene Gerät näher beschreibt. Die maximale Länge beträgt 255 Zeichen.

- **Benutzerdefinierte maximale Leistung verwenden**
  Wenn Sie dieses Optionskästchen für einen Port aktivieren, wird die benutzerdefinierte maximale Leistung verwendet.
- **Benutzerdefinierte maximale Leistung [W]**
  Geben Sie die maximale Leistung an, die ein Port für die Versorgung eines angeschlossenen Geräts zur Verfügung stellt. Dieser Wert wird nur berücksichtigt, wenn das dazugehörige Optionskästchen "Benutzerdefinierte maximale Leistung verwenden" aktiviert ist. Die benutzerdefinierte Leistung wird mit dem Wertebereich der Klasse abgeglichen, die das angeschlossene Gerät angibt:
  - Wenn die benutzerdefinierte Leistung innerhalb der Klasse des angeschlossenen Geräts liegt, wird der benutzerdefinierte Wert verwendet.
  - Wenn die benutzerdefinierte Leistung über der Klasse des angeschlossenen Geräts liegt, wird der höchste Wert der Klasse verwendet.
  - Wenn die benutzerdefinierte Leistung unter der Klasse des angeschlossenen Geräts liegt, wird der niedrigste Wert der Klasse verwendet.

  Wenn die Leistungsaufnahme des angeschlossenen Geräts die definierte bzw. verwendete maximale Leistung überschreitet, wird das angeschlossene Gerät abgeschaltet.
- **Klassifizierung** (nur lesbar)
  Die Classification gibt die Klasse des Geräts an. Daran kann man die maximale Leistung des Geräts ablesen.
- **Status** (nur lesbar)
  Zeigt an, in welchem Zustand sich der Port befindet.
  Es gibt folgende Zustände:
  - disabled
    Die PoE-Spannungsversorgung für diesen Port ist deaktiviert.
  - delivering
    Die PoE-Spannungsversorgung für diesen Port ist aktiviert und es ist ein Gerät angeschlossen.
  - searching
    Die PoE-Spannungsversorgung für diesen Port ist aktiviert, aber es ist kein Gerät angeschlossen.

  Hinweis

  Wenn ein Gerät an einen PoE-fähigen Port angeschlossen wird, wird geprüft, ob die Leistung des Ports für das angeschlossene Gerät ausreicht. Wenn die Leistung des Ports nicht ausreicht, ist PoE zwar unter "Einstellung" aktiviert, der Port hat jedoch den Status "disabled". Der Port wurde dann vom PoE-Power Management deaktiviert.

- **Leistung [mW]** (nur lesbar)
  Zeigt die Leistung an, die der SCALANCE an diesem Port liefert.
- **Spannung [V]** (nur lesbar)
  Zeigt die Spannung an, die an diesem Port anliegt.
- **Strom [mA]** (nur lesbar)
  Zeigt den Strom an, mit dem ein Gerät an diesem Port versorgt wird.

Port-Diagnose

Kabel-Tester

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mit dieser Seite kann jeder einzelne Ethernet-Port eine unabhängige Fehlerdiagnose am Kabel durchführen. Dieser Test wird durchgeführt, ohne dass das Kabel ausgesteckt, ein Kabeltester angeschlossen und am anderen Ende ein Loopback-Modul installiert ist. Kurzschlüsse sowie Leitungsunterbrechungen können auf wenige Meter genau lokalisiert werden.

Hinweis

Bitte beachten Sie, dass dieser Test nur zulässig ist, wenn auf dem zu testenden Port keine Datenverbindung aufgebaut ist. Sollte dennoch auf dem zu testenden Port eine Datenverbindung bestehen, so wird diese kurzzeitig unterbrochen. Ein automatischer Wiederaufbau der Verbindung kann scheitern und muss dann manuell erfolgen.

Beschreibung

- **Port**
  Wählen Sie aus der Klappliste den gewünschten Port aus.
- **Test ausführen**
  Aktiviert die Fehlerdiagnose. Das Ergebnis wird in der Tabelle dargestellt.

Die Tabelle enthält folgende Spalten:

- **Paar**
  Zeigt das Adernpaar im Kabel an.

  Hinweis

  Adernpaare
  - Bei 10/100 Mbit Netzwerkkabeln werden die Adernpaare 4-5 und 7-8 nicht verwendet.
  - Bei 1000 Mbit bzw. Gigabit Ethernet werden alle 4 Adernpaare verwendet. Dabei ist die Zuordnung Adernpaar - Pinbelegung wie folgt (DIN EN 50173):
    - Paar 1 = Pin 4-5
    - Paar 2 = Pin 1-2
    - Paar 3 = Pin 3-6
    - Paar 4 = Pin 7-8
- **Status**
  Zeigt den Status der Leitung an.
- **Distanz [m]**
  Zeigt die Entfernung zum Kabelende, Kabelbruch oder zum Kurzschluss an.

SFP-Diagnose

Mit dieser Seite führen Sie für jeden einzelnen SFP-Port eine unabhängige Fehlerdiagnose durch. Dieser Test wird durchgeführt, ohne dass ein Kabel ausgesteckt, ein Kabel-Tester angeschlossen und am anderen Ende ein Loopback-Modul installiert werden muss.

Hinweis

Bitte beachten Sie, dass dieser Test nur zulässig ist, wenn auf dem zu testenden Port keine Datenverbindung aufgebaut ist. Sollte dennoch auf dem zu testenden Port eine Datenverbindung bestehen, so wird diese kurzzeitig unterbrochen. Ein automatischer Wiederaufbau der Verbindung kann scheitern und muss dann manuell erfolgen.

Beschreibung

Die Seite enthält folgende Felder:

- **Port**
  Wählen Sie aus der Klappliste den gewünschten Port aus.
- **Aktualisieren**
  Erneuert die Anzeige der Werte des eingestellten Ports. Das Ergebnis wird in der Tabelle dargestellt.

Die Werte werden in den folgenden Feldern angezeigt:

- **Name**
  Zeigt den Namen der Schnittstelle an.
- **Modell**
  Zeigt die Art der Schnittstelle an.
- **Ausgabestand**
  Zeigt den Hardware Ausgabestand des SFP an.
- **Seriell**
  Zeigt die Seriennummer des SFP an.
- **Nominale Bit-Rate [MBit/s]**
  Zeigt die Nenn-Bit-Rate der Schnittstelle an.
- **Max. Link (50.0/125um) [m]**
  Zeigt die maximale Distanz in Metern an, die mit diesem Medium möglich sind.
- **Max. Link (62.5/125um) [m]**
  Zeigt die maximale Distanz in Metern an, die mit diesem Medium möglich sind.

Die folgende Tabelle zeigt die Werte des in diesem Port eingesetzten SFP-Stecktransceivers an:

- **Temperatur [°C]**
  Zeigt die Temperatur der Schnittstelle an.
- **Spannung [V]**
  Zeigt die an der Schnittstelle anliegende Spanung an [V].
- **Strom [mA]**
  Zeigt die Stromaufnahme der Schnittstelle an [mA].

- **Rx-Leistung [mW]**
  Zeigt die Empfangsleistung der Schnittstelle an [mW].
- **Tx-Leistung [mW]**
  Zeigt die Sendeleistung der Schnittstelle an [mW].
- **Aktuell**
  Zeigt den aktuellen Wert an.
- **Niedrig**
  Zeigt den niedrigsten Wert an.
- **Hoch**
  Zeigt den höchsten Wert an.

Backup der Konfiguration

Backup

Auf dieser Seite können Sie Backups Ihrer Konfiguration speichern.

Die erstellten Backups werden unter dem Dateityp "ConfigPackBackup" gespeichert. Auf der Seite "System > Laden & Speichern > HTTP/TFTP" können Sie Konfigurationsbackups auf Ihrem Client-PC speichern bzw. von dort laden.

Beschreibung

Die Seite enthält folgende Felder:

- **Name**
  Geben Sie einen Namen für das Backup ein.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Name**
  Zeigt den Namen des Backups an.
- **Größe [kBytes]**
  Die erste Zeile "Available memory" zeigt an, wie viel Speicherplatz für Backups auf dem Gerät verfügbar ist. Wenn Sie ein Backup erstellen, verringert sich der verfügbare Speicherplatz entsprechend. In den weiteren Zeilen wird jeweils die Größe des Backups angezeigt.
- **Wiederherstellen**
  Klicken Sie auf die Schaltfläche "Wiederherstellen", um das entsprechende Backup wieder auf das Gerät zu laden.

Layer 2-Funktionen projektieren

Konfiguration

Layer 2 konfigurieren

Auf dieser Seite nehmen Sie eine Basiskonfiguration der Funktionen des Layer 2 vor. In der jeweiligen Konfigurationsseite dieser Funktionen sind detailliertere Einstellungen möglich. Auf den Konfigurationsseiten können Sie auch die Einstellungen prüfen.

Beschreibung der angezeigten Felder

- Protokoll-basiertes VLAN
  Aktivieren oder deaktivieren Sie protokoll-basiertes VLAN. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN".
- Subnetz-basiertes VLAN
  Aktivieren oder deaktivieren Sie subnetz-basiertes VLAN. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN".
- Dynamic MAC Aging
  Aktivieren oder deaktivieren Sie den Mechanismus "Aging". Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Dynamic MAC Aging".
- Redundanztyp
  Folgende Einstellungen gibt es:
  - "-" (Deaktiviert)
    Die Redundanzfunktion ist deaktiviert.
  - Spanning Tree
    Wenn Sie diese Option auswählen, legen Sie bei "Redundanzverfahren" den gewünschten Redundanzmodus fest.
  - Ring
    Aktiviert Ringredundanz. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Ringredundanz > Ring".
  - Ring mit RSTP
    Wenn Sie diese Option auswählen, ist der Kompatibilitätsmodus für Spanning Tree fest auf RSTP gesetzt. In der Klappliste "Redundanzverfahren" legen Sie den Redundanzmodus der Ringredundanz fest. Die aktuelle Einstellung können Sie in den Menüs "Ringredundanz" und "Spanning Tree" ändern.

    Hinweis

    Einschränkung bzgl. Ports bei der Option „Ring mit RSTP“

    Wenn Sie die Option "Ring mit RSTP" aktiviert haben, dürfen folgende Ports nicht im Spanning Tree enthalten sein:
    - Ring-Ports
    - Standby-Ports
    - Standby-Koppelports

- **Redundanzverfahren**
  Wenn Sie beim "Redundanztyp" "Ring" oder "Ring mit RSTP" auswählen, stehen Ihnen folgende Auswahlmöglichkeiten zur Verfügung:
  - Automatic Redundancy Detection
    Wählen Sie diese Einstellung, um eine automatische Konfiguration der Redundanzbetriebsart vorzunehmen.
    Im Modus "Automatic Redundancy Detection" stellt das Gerät automatisch fest, ob sich ein Gerät mit der Rolle "HRP Manager" im Ring befindet. Ist dies der Fall, so nimmt das Gerät die Rolle "HRP Client" ein.
    Wird kein HRP Manager gefunden, so handeln alle Geräte mit der Einstellung "Automatic Redundancy Detection" oder "MRP Auto-Manager" untereinander aus, welches Gerät die Rolle "MRP Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP Manager". Die übrigen Geräte stellen sich automatisch auf die Betriebsart "MRP Client" ein.
  - MRP Auto-Manager
    Im Modus "MRP Auto-Manager" handeln die Geräte untereinander aus, welches Gerät die Rolle "MRP Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP Manager". Die übrigen Geräte stellen sich automatisch auf die Betriebsart "MRP Client" ein.
    Im Gegensatz zur Einstellung "Automatic Redundancy Detection" sind die Geräte nicht in der Lage zu erkennen, ob ein HRP-Manager im Ring ist.

    Hinweis

    MRP-Projektierung in STEP 7

    Wenn Sie über STEP 7 die Rolle "Manager (Auto)" oder "Manager" für das Gerät einstellen, wird auf dieser WBM-Seite in beiden Fällen "MRP Auto-Manager" angezeigt. In der Anzeige im CLI wird zwischen den beiden Rollen unterschieden.

  - MRP-Client
    Das Gerät nimmt die Rolle MRP-Client ein.
  - HRP-Client
    Das Gerät nimmt die Rolle HRP-Client ein.
  - HRP-Manager
    Das Gerät nimmt die Rolle HRP-Manager ein.
    Bei der Projektierung eines HRP-Rings muss ein Gerät als HRP-Manager eingestellt werden. Bei allen übrigen Geräten muss "HRP Client" oder "Automatic Redundancy Detection" eingestellt sein.

  Wenn Sie beim "Redundanztyp" "Spanning Tree" auswählen, stehen Ihnen folgende Auswahlmöglichkeiten zur Verfügung:
  - **STP**
    Aktiviert das Spanning Tree Protocol (STP). Typische Rekonfigurationszeiten bei Spanning Tree liegen zwischen 20 und 30 Sekunden. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Spanning Tree".

- RSTP
  Aktiviert das Rapid Spanning Tree Protocol (RSTP). Wenn an einem Port ein Spanning Tree-Telegramm erkannt wird, fällt dieser Port von RSTP auf Spanning Tree zurück. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Spanning Tree".

  Hinweis

  Bei RSTP (Rapid Spanning Tree-Protokoll) kann es zu kurzzeitiger Schleifenbildung mit Telegrammverdoppelung oder zu Telegrammüberholungen kommen. Wenn das in Ihrem Anwendungsfall nicht akzeptabel sein sollte, müssen Sie das langsamere Standardverfahren Spanning Tree benutzen.

- MSTP
  Aktiviert das Multiple Spanning Tree Protocol (MSTP). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Spanning Tree".

- Standby
  Aktivieren oder deaktivieren die Funtkion "Standby-Redundanz". Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Ringredundanz > Standby".

- MRP-Interconnection
  Aktivieren oder deaktivieren Sie die Funktion MRP-Interconnection. Weitere Einstellungen finden Sie unter "Layer 2 > Ringredundanz > MRP-Interconnection".

- Passive Listening
  Aktivieren oder deaktivieren Sie Funktion "Passive Listening".

- RMON
  Wenn Sie dieses Optionskästchen aktivieren, ermöglicht Remote Monitoring (RMON), Diagnosedaten im Gerät zu sammeln, aufzubereiten und über SNMP von einer Netzwerkmanagement-Station, die ebenfalls RMON unterstützt, auszulesen. Diese Diagnosedaten, wie zum Beispiel portbezogene Lastverläufe, ermöglichen es, Probleme im Netzwerk frühzeitig zu erkennen und zu beseitigen.

- Dynamisches Multicast
  Folgende Einstellungen sind möglich:

  - "-" (Deaktiviert)

  - IGMP Snooping
    Aktiviert IGMP (Internet Group Management Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mulitcast > IGMP".

  - MLD Snooping
    Aktiviert MLD. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mulitcast > MLD".

  - IGMP und MLD Snooping
    Aktiviert IGMP und MDL.

  - GMRP
    Aktiviert GMRP (GARP Multicast Registration Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mulitcast > GMRP".

    Hinweis

    GMRP und IGMP können nicht gleichzeitig betrieben werden.

- GVRP
  Aktivieren oder deaktivieren Sie "GVRP" (GARP VLAN Registration Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > GVRP".
- Mirroring
  Aktivieren oder deaktivieren Sie die Port-Spiegelung. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > Port".
- Loop Detection
  Aktivieren oder deaktivieren Sie die Schleifenerkennung. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Loop Detection".
- PTP
  Legen Sie fest, wie das Gerät PTP-Nachrichten verarbeiten soll. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > PTP".
  - aus
    Das Gerät verarbeitet keine PTP-Nachrichten. PTP-Nachrichten werden aber nach den Regeln des Switchs weitergeleitet.
  - transparent
    Das Gerät übernimmt die Funktion einer Transparent Clock und leitet PTP-Nachrichten an andere Teilnehmer weiter, wobei es Eintragungen in das Korrekturfeld der PTP-Nachricht vornimmt.

QoS

Allgemein

Hinweis

Diese Seite gibt es nur bei den folgenden Baugruppen:

- SCALANCE XB-200
- SCALANCE XC-200
- SCALANCE XF-200BA
- SCALANCE XP-200
- SCALANCE XR-300WG

Übertragungsprioritäten

Auf dieser Seite können Sie die Prioritäten verschiedener Frames festlegen. Außerdem können Sie abhängig von der Priorität einstellen, nach welcher Methode die Abarbeitungsreihenfolge der Frames festgelegt wird. Beachten Sie hierzu auch das Kapitel "Grundlagen (Seite 2122)".

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Broadcast-Priorität**
  Legen Sie die Priorität von Broadcast-Frames fest. Der Switch sortiert das Frame entsprechend dieser Priorisierung in eine Queue ein. Die Zuordnung der Priorität zu einer Queue konfigurieren Sie auf der Seite "Layer 2 > QoS > CoS-Zuordnung".
- **Agent-Priorität**
  Legen Sie die Priorität von Agent-Frames fest. Der Switch sortiert das Frame entsprechend dieser Priorisierung in eine Queue ein. Die Zuordnung der Priorität zu einer Queue konfigurieren Sie auf der Seite "Layer 2 > QoS > CoS-Zuordnung".
- **Abarbeitungsschema**
  Wählen Sie aus, in welcher Reihenfolge die Frames in den Queues verarbeitet werden. Je höher die Queue-Nummer desto höher die Abarbeitungspriorität.
  - Strict Queueing
    Solange sich Frames mit einer hohen Priorität in der Queue befinden, werden nur diese hoch priorisierten Frames abgearbeitet.
  - Weighted Fair Queueing
    Auch wenn sich Frames mit einer hohen Priorität in der Queue befinden, werden gelegentlich Frames mit einer niedrigeren Priorität abgearbeitet.

CoS-Zuordnung

CoS-Warteschlange

Auf dieser Seite können Sie CoS-Prioritäten verschiedenen Warteschlangen (Traffic Queues) zuordnen.

Beschreibung der angezeigten Felder

- **CoS**
  Zeigt CoS-Priorität der eingehenden Pakete an.
- **Queue**
  Wählen Sie aus der Klappliste die Queue aus, die der CoS-Priorität zugeordnet wird.
  Je höher die Nummer der Queue desto höher die Abarbeitungspriorität
  Bei den Queues 1 - 6 werden gelegentlich Frames mit einer niedrigeren Priorität abgearbeitet, auch wenn sich Frames mit einer hohen Priorität in der Warteschlange befinden.
  Bei den Queues 7 - 8 werden nur Frames mit einer hohen Priorität abgearbeitet, solange sich hoch priorisierte Frames in der Warteschlange befinden.

Die Service-Klassen (CoS) sind den Queues wie folgt zugeordnet:

SCALANCE XB-200/XP-200	SCALANCE XM-400/XR-500
CoS 0 → Queue 2	CoS 0 → Queue 2
CoS 1 → Queue 1	CoS 1 → Queue 1
CoS 2 → Queue 1	CoS 2 → Queue 3
CoS 3 → Queue 2	CoS 3 → Queue 4
CoS 4 → Queue 3	CoS 4 → Queue 5
CoS 5 → Queue 3	CoS 5 → Queue 6
CoS 6 → Queue 4	CoS 6 → Queue 7
CoS 7 → Queue 4	CoS 7 → Queue 8

DSCP-Zuordnung

DSCP-Zuordnung

Auf dieser Seite können Sie DSCP-Prioritäten verschiedenen Warteschlangen (Queues) zuordnen.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **DSCP min**
  Wählen Sie aus der Klappliste den minimalen Wert für einen Bereich von DSCP-Codes aus, denen Sie eine Queue zuweisen wollen.
- **DSCP max**
  Wählen Sie aus der Klappliste den maximalen Wert für einen Bereich von DSCP-Codes aus, denen Sie eine Queue zuweisen wollen.
- **Queue**
  Wählen Sie aus der Klappliste die Weiterleitungs-Warteschlange (Sendepriorität) aus, welcher dem Bereich von DSCP-Codes zugeordnet wird.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden den DSCP-Codes im angegebenen Bereich die gewählte Weiterleitungs-Warteschlange (Sendepriorität) zugewiesen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **DSCP**
  Zeigt die DSCP-Priorität der eingehenden Frames an.
- **Queue**
  Wählen Sie aus der Klappliste die Queue aus, die der DSCP-Priorität zugeordnet wird.
  Je höher die Queue-Nummer desto höher die Abarbeitungspriorität.
  Bei den Queues 1 - 6 werden gelegentlich Frames mit einer niedrigeren Priorität abgearbeitet, auch wenn sich Frames mit einer hohen Priorität in der Warteschlange befinden.
  Bei den Queues 7 - 8 werden nur Frames mit einer hohen Priorität abgearbeitet, solange sich hoch priorisierte Frames in der Warteschlange befinden.

SCALANCE XB-200/XP-200	SCALANCE XM-400/XR-500
DSCP-Codes 0 - 15 → Queue 1	DSCP-Codes 0 - 7 → Queue 2
DSCP-Codes 16 - 31 → Queue 2	DSCP-Codes 8 - 15 → Queue 1
DSCP-Codes 32 - 47 → Queue 3	DSCP-Codes 16 - 23 → Queue 3
DSCP-Codes 48 - 63 → Queue 4	DSCP-Codes 24 - 31 → Queue 4
	DSCP-Codes 32 - 39 → Queue 5
	DSCP-Codes 40 - 47 → Queue 6
	DSCP-Codes 48 - 55 → Queue 7
	DSCP-Codes 56 - 63 → Queue 8

QoS-Priorisierung

Festlegen der Priorität

Auf dieser Seite können Sie portgranular einstellen, nach welchem Verfahren weiterzuleitende Frames priorisiert werden.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Priorisierungsmodus**
  Wählen Sie aus der Klappliste die Einstellung. Es gibt folgende Einstellmöglichkeiten:
  - Portpriorisierung
  - Priorisierung nach COS
  - Priorisierung nach DSCP
  - Priorisierung nach COS-DSCP
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die konfigurierbaren Ports an.
  Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorisierungsmodus**
  Wählen Sie aus der Klappliste den gewünschten Modus:

  Hinweis

  Die Priorisierung des Empfangsports konfigurieren Sie auf der Seite "Layer 2 > VLAN > Port-basiertes VLAN".

  Die Zuordnung der folgenden Prioritäten zu einer Queue konfigurieren Sie auf der Seite "Layer 2 > QoS > CoS-Zuordnung":

  - Empfangsport
  - VLAN-Tag
  - Broadcast- und Agent-Frame

  Die Zuordnung der DSCP-Priorisierung zu einer Queue konfigurieren Sie auf der Seite "Layer 2 > QoS > DSCP-Zuordnung".

  – Portpriorisierung
    Der Switch sortiert ankommende Frames entsprechend der Priorisierung des Empfangsports in eine Queue ein.
    Wenn ein DSCP-Wert im IP-Header vorhanden ist, wird dieser nicht berücksichtigt.
    Wenn ein VLAN-Tag vorhanden ist, wird es durch den Prioritätswert des Empfangsports ersetzt.

  – Priorisierung nach COS
    Wenn ein ankommendes Frame ein VLAN-Tag enthält, sortiert der Switch es entsprechend dieser Priorisierung in eine Queue ein.
    Wenn das Frame kein VLAN-Tag enthält, sortiert der Switch das Frame entsprechend der Priorisierung des Empfangsports in eine Queue ein.
    Wenn ein DSCP-Wert im IP-Header vorhanden ist, wird dieser nicht berücksichtigt.

  – Priorisierung nach DSCP
    Wenn ein ankommendes Frame eine DSCP-Priorisierung enthält, sortiert der Switch es entsprechend dieser Priorisierung in eine Queue ein.
    Wenn das Frame keine DSCP-Priorisierung enthält, sortiert der Switch das Frame entsprechend der Priorisierung des Empfangsports in eine Queue ein.
    Wenn das Frame ein VLAN-Tag enthält, wird es nicht berücksichtigt.

  – Priorisierung nach COS-DSCP
    Bei einem ankommenden Frame wird sequenziell geprüft, welche Priorisierung es enthält.
    Wenn das Frame eine DSCP-Priorisierung enthält, wird es wie im Modus "Priorisierung nach DSCP" behandelt.
    Wenn das Frame keine DSCP-Priorisierung enthält, prüft der Switch, ob das Frame ein VLAN-Tag enthält. Wenn es ein VLAN-Tag enthält, sortiert der Switch es entsprechend dieser Priorisierung in eine Queue ein.
    Wenn das Frame weder eine DSCP-Priorisierung noch ein VLAN-Tag enthält, sortiert der Switch das Frame entsprechend der Priorisierung des Empfangsports in eine Queue ein.

CoS Port-Neuzuordnung

Priorität beim Versenden ändern

Auf dieser Seite können Sie abhängig von der Priorität beim Empfangen eines Frames, die Priorität ändern, mit der es versendet wird.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **CoS-Neuzuordnung**
  Aktivieren oder Deaktivieren Sie, dass Frames mit geänderten Prioritäten entsprechend Tabelle 2 versendet werden.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Priorität 0 - 7**
  Die Priorität der Spalte steht für die Priorität, mit der ein Frame empfangen wird.
  - 0 - 7
    Wählen Sie die Priorität aus, mit der ein Frame versendet werden soll.
  - Keine Änderung
    Keine Änderung in Tabelle 2.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorität 0 - 7**
  Die Priorität der Spalte steht für die Priorität, mit der ein Frame empfangen wird.
  Wählen Sie in der Klappliste die Priorität aus, mit der ein Frame versendet werden soll.

Lastkontrolle

Begrenzung der Transferrate eingehender und ausgehender Daten

Auf dieser Seite konfigurieren Sie die Lastbegrenzung (maximale Anzahl von Datenpaketen pro Sekunde) für die einzelnen Ports. Sie können festlegen, für welche Kategorie von Telegrammen diese Grenzwerte gelten sollen.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Ingress Unicast(DLF)-Limit / Ingress Broadcast-Limit / Ingress Multicast-Limit / Ingress Unicast-Limit**
  Wählen Sie die gewünschte Einstellung aus.
  - Aktiviert: Aktiviert die Funktion.
  - Deaktiviert: Deaktiviert die Funktion
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert
- **Ingress-Gesamtübertragungsrate [pkts/s]**
  Legen Sie die maximale Anzahl der eingehenden Pakete fest. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **Egress-Übertragungsrate [kb/s]**
  Legen Sie die Datenrate für alle ausgehenden Telegramme fest. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Ingress Unicast (DLF)-Limit**
  Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Unicast-Telegramme mit nicht auflösbarer Adresse (Destination Lookup Failure).
- **Ingress Broadcast-Limit**
  Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Broadcast-Telegramme.
- **Ingress Multicast-Limit**
  Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Multicast-Telegramme.
- **Ingress Unicast-Limit**
  Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Unicast-Telegramme.

- **Ingress-Gesamtübertragungsrate [pkts/s]**
  Legen Sie die maximale Anzahl der eingehenden Pakete fest.
- **Egress-Übertragungsrate [kb/s]**
  Legen Sie die Datenrate für alle ausgehenden Telegramme fest.

Hinweis

Rundungen der Werte, Abweichung vom Sollwert

Beachten Sie bei der Eingabe, dass das WBM auf korrekte Werte rundet.

Sind Werte für die Ingress-Gesamtübertragungsrate und die Egress-Übertragungsrate konfiguriert, können die tatsächlichen Werte im Betrieb leicht von den eingestellten Werten abweichen.

VLAN

Allgemein

VLAN-Konfigurationsseite

Auf dieser Seite legen Sie fest, ob das Gerät Telegramme mit VLAN-Tags transparent weiterleitet (IEEE 802.1D/VLAN-unaware-Modus) oder VLAN-Informationen berücksichtigt (IEEE 802.1Q/VLAN-aware-Modus). Wenn sich das Gerät im Modus "802.1Q VLAN Bridge" befindet, können Sie VLANs definieren und die Verwendung der Ports festlegen.

Die Einstellmöglichkeiten auf dieser Seite sind abhängig davon, was Sie im Feld "Base Bridge-Modus" auswählen.

Hinweis

Ändern der "Agent VLAN ID"

Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die "Agent VLAN ID" ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

Wichtige Regeln für VLANs

Berücksichtigen Sie bei der Konfiguration und beim Betrieb Ihrer VLANs folgende Regeln:

- Telegramme mit der VLAN-ID "0" werden wie ungetaggte Telegramme behandelt, behalten jedoch ihren Prioritätswert.
- Alle Ports am Gerät senden standardmäßig Telegramme ohne VLAN-Tag, um sicher zu gehen, dass der Endteilnehmer diese Telegramme empfangen kann.
- Bei SCALANCE X-Geräten ist an allen Ports die VLAN-ID "1" voreingestellt.

- Wenn an einem Port ein Endteilnehmer verbunden ist, dann sollen ausgehende Telegramme ohne Tag versendet werden (statischer Zugriffs-Port). Wenn sich an dem Port ein weiterer Switch befindet, so ist das Telegramm mit einem Tag zu versehen (Trunk Port).
- Bei einem Trunk Port erfolgt die VLAN-Zuordnung dynamisch. Statische Konfigurationen können nur durchgeführt werden, wenn der Port zusätzlich zur Eigenschaft Trunk Port noch statisch als Member in den betreffenden VLANs eingetragen wird. Ein Beispiel für eine statische Konfiguration ist das Zuweisen von Multicastgruppen in bestimmten VLANs.

Hinweis

Bei SCALANCE XB-200, XC-200, XF-200BA, XP-200 und XR-300WG gelten die Regeln beim Base Bridge-Modus "802.1Q VLAN Bridge".

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Bridge-Modus**
  - Customer
    Wenn Sie das Gerät mit der Rolle "Customer" betreiben, verhält es sich wie ein Standard-IE-Switch.
  - Provider
    Wenn Sie das Gerät mit der Rolle "Provider" betreiben, verfügt es zusätzlich zu den Eigenschaften der Rolle "Customer" über die Möglichkeit, äußere VLAN-Tags zu verwalten. In dieser Rolle können Sie die Funktion Q-in-Q VLAN-Tunnel verwenden.
- **Base Bridge-Modus (nicht bei SCALANCE XM-400/XR-500)**
  Legen Sie die den "Base Bridge-Modus" fest.

  Hinweis

  Base Bridge-Modus wechseln

  Beachten Sie den Abschnitt "Base Bridge-Modus wechseln". In diesem Abschnitt ist beschrieben, wie sich ein Welchsel auf die bestehende Konfiguration auswirkt.

  - 802.1Q VLAN Bridge
    Stellt bei dem Gerät den Modus "VLAN-aware" ein. In diesem Modus werden VLAN-Informationen berücksichtigt.
  - 802.1D Transparent Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLANTags nicht verändert, sondern transparent weitergeleitet. Die VLAN-Priorität wird für CoS ausgewertet. Sie können in diesem Modus keine VLANs anlegen. Es ist nur ein Management-VLAN verfügbar: VLAN 1.
- **VLAN-ID**
  Tragen Sie im Eingabefeld "VLAN-ID" die VLAN-ID ein. Wertebereich: 1 ... 4094.

Die Tabelle gliedert sich in folgende Spalten:

- Selektieren
  Wählen Sie die Zeile, die Sie löschen wollen.
- VLAN-ID
  Zeigt die VLAN-ID an. Die VLAN-ID (eine Zahl zwischen 1 und 4094) kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden. Bis zu 257 VLANs können definiert werden.
- Name
  Tragen Sie einen Namen für das VLAN ein. Der Name hat nur informativen Charakter und keine Auswirkungen auf die Konfiguration. Der VLAN-Name darf maximal 32 Zeichen lang sein.
- Status
  Zeigt die Statusart des Eintrags in der internen Portfiltertabelle an. Dabei bedeutet statisch, dass die Adresse vom Anwender statisch eingetragen wurde. Die Angabe GVRP bedeutet, dass die Konfiguration über ein GVRP-Telegramm registriert wurde. Voraussetzung dafür ist allerdings, dass GVRP für das Gerät aktiviert wurde.
- Private VLAN-Typ
  Zeigt den Typ des PVLANs an.
- Primary VLAN-ID
  Zeigt bei Secondary PVLANs die ID des dazugehörigen Primary PVLANs an.

- Transparent (nur bei SCALANCE XM-400/XR-500)
  Wenn Sie diese Einstellungen aktivieren, schalten Sie ein VLAN in den transparenten Modus. Ports, die diesem VLAN als Mitglied oder ungetaggtes Mitglied zugeordnet waren, werden nun zu transparenten Ports.
  Das bedeutet:
  - Die Port-VLAN-ID des transparenten Ports wird auf die ID dieses VLANs eingestellt.
  - Ungetaggte Telegramme, die an diesen Ports empfangen werden, werden an allen anderen transparenten Ports ebenfalls ohne Tag weitergeleitet, sofern sie nicht durch eine Protokoll-, oder Subnetzregel in ein Standard-VLAN geleitet werden.
  - Telegramme, die mit VLAN-ID "0" getagged sind und an diesen Ports empfangen werden, werden an allen anderen transparenten Ports ebenfalls getagged mit VLAN-ID "0" weitergeleitet, sofern sie nicht durch eine Protokoll-, oder Subnetzregel in ein Standard-VLAN geleitet werden.
  - Telegramme, die mit der VLAN-ID des transparenten VLANs getagged sind und an diesen Ports empfangen werden, werden an allen transparenten Ports ebenfalls getagged mit der VLAN-ID des Transparent VLANs weitergeleitet.
  - Andere Telegramme werden entsprechend der normalen VLAN-Regeln weitergeleitet, wobei sich ein transparenter Port wie ein ungetaggtes Mitglied in diesem VLAN verhält.
  - Alle Ports, die nicht Mitglied oder ungetaggtes Mitglied in dem betreffenden VLAN waren, werden automatisch in den Status "Forbidden" gesetzt.
  - Solange ein VLAN als transparentes VLAN konfiguriert ist, können die Portzugehörigkeiten dieses VLANs nicht modifiziert werden.
  - Sie können nur jeweils ein VLAN als transparentes VLAN konfigurieren.
- Liste der Ports
  Legen Sie die Verwendung des Ports fest. Folgende Möglichkeiten gibt es:
  - "-"
    Der Port ist kein Mitglied des angegebenen VLANs. Bei der Neudefinition sind alle Ports mit der Kennung "-" belegt.
  - M
    Der Port ist Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden mit dem entsprechenden VLAN-Tag weitergeleitet.
  - R
    Der Port ist Mitglied des VLANs. Die Registrierung erfolgt über ein GVRP-Telegramm.
  - U (Großbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet. Von diesem Port werden Telegramme ohne VLAN-Tag gesendet.
  - u (Kleinbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs, jedoch ist das VLAN nicht als Port-VLAN konfiguriert. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet.

– F
Der Port ist kein Mitglied des angegebenen VLANs und es ist nicht möglich, dass das VLAN dynamisch über GVRP an diesem Port registriert werden kann. Wenn ein Port bei einem VLAN diese Option aufweist, dann kann er kein Mitglied dieses VLAN werden, auch dann nicht, wenn er als Trunk-Port konfiguriert wird. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".

– T
Diese Option wird nur angezeigt und kann im WBM nicht ausgewählt werden. Dieser Port ist Trunk-Port und wurde dadurch Mitglied in allen VLANs. Sie konfigurieren diese Funktion im CLI (Command Line Interface) mit Hilfe des Befehls "`switchport mode trunk`".

Base Bridge-Modus bei SCALANCE XB-200 wechseln

VLAN-unaware (802.1D Transparent Bridge) → VLAN-aware (802.1Q VLAN Bridge)

Wenn Sie den Base Bridge Mode von VLAN-unaware in VLAN-aware ändern, hat dies folgende Auswirkungen:

- Alle statischen und dynamischen Unicast-Einträge werden gelöscht.
- Alle statischen und dynamischen Multicast-Einträge werden gelöscht.
- Bei Spanning Tree können Sie die folgende Protokollkompatibilität einstellen: STP, RSTP und MSTP.

VLAN-aware (802.1Q VLAN Bridge) → VLAN-unaware (802.1D Transparent Bridge)

Wenn Sie den Base Bridge Mode von VLAN-aware in VLAN-unaware ändern, hat dies folgende Auswirkungen:

- Alle statischen und dynamischen VLAN-Informationen werden gelöscht.
- Es wird ein Management-VLAN angelegt: VLAN 1.
- Alle statischen und dynamischen Unicast-Einträge werden gelöscht.
- Alle statischen und dynamischen Multicast-Einträge werden gelöscht.
- Bei Spanning Tree können Sie die folgende Protokollkompatibilität einstellen: STP und RSTP.
- Sie können GVRP nicht nutzen.
- Sie können Guest VLAN nicht nutzen.
- Die VLAN-Zuordnung kann nicht vom RADIUS-Server übernommen werden.
- Sie können den Port-Typ nicht konfigurieren.
- Definierte Zugriffsregeln müssen für alle VLANs gültig sein. Auf der Seite "Security > Management ACL" muss für den Parameter "Zulässige VLANs" der Wert "1-4094" definiert sein.

GVRP

Konfiguration der GVRP-Funktion

Über ein GVRP-Telegramm kann sich ein anderes Gerät am Port des Geräts für eine bestimmte VID registrieren. Ein anderes Gerät kann z. B. ein Endteilnehmer oder ein Switch sein. Das Gerät kann außerdem GVRP-Telegramme über diesen Port senden. Auf dieser Seite können Sie jeden Port für die GVRP-Funktionalität aktivieren.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- GVRP
  Aktivieren oder deaktivieren Sie die Funktion "GVRP".

Die Tabelle 1 gliedert sich in folgende Spalten:

- 1. Spalte
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- Einstellung
  Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert
    Aktiviert das Senden von GVRP-Telegrammen.
  - Deaktiviert
    Deaktiviert das Senden von GVRP-Telegrammen.
  - Keine Änderung
    Keine Änderung in Tabelle 2.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Port
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- Einstellung
  Aktivieren oder deaktivieren Sie das Senden von GVRP-Telegrammen.

Port-basiertes VLAN

Verarbeitung empfangener Telegramme

Auf dieser Seite legen Sie die Konfiguration der Port-Eigenschaften für den Telegrammempfang fest.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Priorität / Port-VID / Erlaubte Telegrammtypen / Ingress Filterung**
  Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorität**
  Wählen Sie die Priorität aus, mit der ungetaggte Telegramme versehen werden.
  Die CoS-Priorität (Class of Service), die im VLAN-Tag verwendet wird. Wird ein Telegramm ohne Tag empfangen, wird ihm diese Priorität zugeordnet. Diese Priorität legt fest, wie dieses Telegramm im Vergleich zu anderen Telegrammen weiterhin bearbeitet wird. Es gibt insgesamt acht Prioritäten, mit den Werten 0 bis 7, wobei 7 der höchsten Priorität entspricht (IEEE 802.1p Port Priority).
- **Port-VID**
  Wählen Sie die VLAN-ID aus. Nur die VLAN-IDs sind wählbar, die Sie auf der Seite "VLAN > Allgemein" definiert haben.
  Wenn ein empfangenes Telegramm kein VLAN-Tag hat, so wird es um ein Tag mit der hier angegebenen VLAN-ID ergänzt und entsprechend den Regeln am Port gesendet.

- **Erlaubte Telegrammtypen**
  Legen Sie fest, welche Arten von Telegrammen akzeptiert werden. Es gibt folgende Alternativen:
  - Nur getraggte Frames
    Das Gerät verwirft alle ungetaggten Telegramme. Telegramme, die mit "0" getagged sind, werden wie ungetaggte Telegramme behandelt. Das Gerät leitet alle getaggten Telegramme weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration.
  - Nur ungetaggte und mit Priorität getaggte Frames
    Das Gerät verwirft alle getaggten Telegramme. Das Gerät leitet alle ungetaggten Telegramme sowie Telegramme mit einer Priorität (Priority Tagged Frames) weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration. Wenn Sie den Bridge-Modus "Provider" konfiguriert haben, bedeutet diese Einstellung für das Gerät, dass es alle eingehenden Telegramme als ungetaggte Telegramme behandelt.
  - Alle
    Das Gerät leitet alle Telegramme weiter
- **Ingress Filterung**
  Legen Sie fest, ob die VID von empfangenen Telegrammen ausgewertet wird. Folgende Möglichkeiten gibt es:
  - Aktiviert
    Die VLAN-ID empfangener Telegramme bestimmt die Weiterleitung: Für die Weiterleitung eines VLAN-getaggten Telegramms muss der Empfangsport Mitglied im gleichen VLAN sein. Am Empfangsport werden Telegramme aus unbekannten VLANs verworfen.
  - Deaktiviert
    Alle Telegramme werden weitergeleitet.

Protokoll-basierte VLAN-Gruppe

Einleitung

Auf dieser Seite legen Sie Gruppen fest und ordnen ihnen ein Protokoll zu.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Protokoll-basiertes VLAN**
  Aktivieren oder deaktivieren Sie die protokoll-basierte VLAN-Zuordnung.
- **Protokollwert**
  Tragen Sie den hexadezimalen Protokollwert ein.
  Einige Beispiele sind:
  - Profinet: 88:92
  - IP: 08:00
  - Novell: 81:37
  - netbios: f0:f0
  - appletalk: 80:9b
- **Gruppen-ID**
  Tragen Sie die ID der Gruppe ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Protokollwert**
  Zeigt den Protokollwert an.
- **Gruppen-ID**
  Zeigt die Gruppen-ID an.

Protokoll-basierter VLAN-Port

Einleitung

Auf dieser Seite legen Sie fest, welches Protokoll und welches VLAN dem einzelnen Port zugeordnet wird.

Beschreibung der angezeigten Felder

- **Port**
  Wählen Sie in den gewünschten Port aus. Alle verfügbaren Ports und die Link-Aggregationen werden angeboten.
- **Gruppen-ID**
  Wählen Sie in der Klappliste die Gruppen-ID aus. Die ID legen Sie auf der WBM-Seite "Layer 2 > VLAN > Protokoll-basierte VLAN-Gruppe" fest.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- **Gruppen-ID**
  Zeigt die Gruppen-ID an, die dem Port zugeordnet ist.
- **VLAN ID**
  Wählen Sie die gewünschte VLAN-ID aus, die Sie dem Port zuordnen.

IPv4 Subnetz-basiertes VLAN

Einleitung

Auf dieser Seite legen Sie fest, welche VLAN-ID dem Subnetz zugeordnet wird.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Subnetz-basiertes VLAN
  Aktivieren oder deaktivieren Sie die IPv4 subnetz-basierte VLAN-Zuordnung.
- Port
  Wählen Sie den Port aus. Alle verfügbaren Ports und die Link-Aggregationen werden angeboten.
- Subnetzadresse
  Tragen Sie die IP-Adresse des Subnetzes ein.
  Beispiel: 192.168.10.0 für das Netz 192.168.10.x mit den Teilnehmern 192.168.10.1 bis 192.168.10.254.
- Subnetzmaske
  Tragen Sie die Subnetzmaske ein.

Die Tabelle gliedert sich in folgende Spalten:

- Port
  Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- Subnetzadresse
  Zeigt das Subnetz an, das dem Port zugeordnet ist.
- Subnetzmaske
  Zeigt die Subnetzmaske an, die dem Port zugeordnet ist.
- VLAN ID
  Wählen Sie die VLAN-ID, die Sie dem Port bzw. dem Subnetz zuordnen.

IPv6 Präfix-basiertes VLAN

Einleitung

Auf dieser Seite legen Sie fest, welche VLAN-ID dem IPv6-Präfix zugeordnet wird.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Port
  Wählen Sie in der Klappliste den Port aus. Alle verfügbaren Ports und die Link Aggregationen werden angeboten.
- Präfixadresse
  Tragen Sie die IPv6-Adresse ein.

- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.
- **VLAN-ID**
  Wählen Sie die VLAN-ID aus, die Sie dem Port bzw. dem Präfix zuordnen.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Port**
  Alle verfügbaren Ports und die Link Aggregationen werden angezeigt.
- **Präfixadresse**
  Zeigt die IPv6-Adresse des Ports an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **VLAN-ID**
  Wählen Sie die VLAN-ID aus, die Sie dem Port bzw. dem Präfix zuordnen.

Private VLAN

Allgemein

Private VLAN-Konfigurationsseite

Auf dieser Seite definieren Sie die Typen der PVLANs und ordnen Seondary PVLANs einem Primary PVLAN zu.

Hinweis

Auf allen Geräten eines PVLANs müssen alle Secondary PVLANs bekannt sein. Auch wenn ein Gerät keinen Host-Port in einem Secondary PVLAN hat, muss das Secondary PVLAN auf dem Gerät bekannt sein.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- VLAN-ID
  Zeigt die VLAN-ID an.
- Private VLAN-Typ
  Legen Sie den Typ des PVLANs fest:
  - Primary
    Mit diesem Typ definieren Sie ein Primary PVLAN. Sie können in einem PVLAN nur ein Primary PVLAN definieren. Das Primary PVLAN verwendet die VLAN-ID des VLANs.
  - Isolated
    Mit diesem Typ definieren Sie ein Secondary PVLAN. In diesem Secondary PVLAN kann sich nur ein Gerät befinden. Das Secondary PVLAN hat eine spezifische VLAN-ID.
  - Community
    Mit diesem Typ definieren Sie ein Secondary PVLAN. Die Geräte in diesem Secondary PVLAN können über Layer 2 miteinander kommunizieren. Das Secondary PVLAN hat eine spezifische VLAN-ID.
- Primary VLAN-ID
  Wählen Sie bei Secondary PVLANs die VLAN-ID des Primary PVLANs aus.

IP-Schnittstellen-Zuordnung

Private VLAN-Konfigurationsseite

Auf dieser Seite legen Sie fest, aus welchen Secondary PVLANs die IP-Schnittstelle des Primary PVLANs erreichbar sein soll. Konfigurieren Sie die IP-Schnittstellen-Zuordnung für alle Funktionen, bei denen ein Endgerät aus dem Secondary PVLAN über die IP-Schnittstelle des Primary PVLANs kommunizieren muss.

Beispiele:

- Ein Endgerät im Secondary PVLANs ist als DHCP-Client konfiguriert. Es ist ein Remote DHCP-Server eingerichtet. Ein PVLAN-Switch ist als DHCP Relay Agent konfiguriert. Konfigurieren Sie eine IP-Schnittstelle im Primary PVLAN des DHCP Relay Agents. Ordnen Sie die Secondary PVLANs, in denen sich DHCP-Clients befinden, dieser IP-Schnittstelle zu.
- Ein PVLAN-Switch ist als Router konfiguriert. Konfigurieren Sie eine IP-Schnittstelle im Primary PVLAN des Routers. Ordnen Sie die Secondary PVLANs dieser IP-Schnittstelle zu, in denen sich Endgeräet befinden, die den Router als Gateway verwenden.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie ein Primary PVLAN mit einer IP-Schnittstelle aus.
- **Secondary VLAN-ID**
  Wählen Sie ein Secondary VLAN-ID aus, aus dem die IP-Schnittstelle des Primary PVLANs erreichbar sein soll.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Schnittstelle**
  Zeigt die IP-Schnittstelle an.
- **Secondary VLAN-ID**
  Zeigt die Secondary VLAN-ID des Secondary PVLANs an, aus dem die IP-Schnittstelle des Primary PVLANs erreichbar ist.

Provider Bridge

Tunnel-Ports

Konfigurationsseite für Tunnel-Ports

Auf dieser Seite aktivieren Sie die Funktion Q-in-Q VLAN-Tunnel. Telegramme, die ein Tunnel-Port empfängt, werden um ein äußeres VLAN-Tag, die PVID des Ports, erweitert.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**
  Wählen Sie aus der Klappliste die Einstellung. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert
    Aktiviert die Funktion Q-in-Q VLAN-Tunnel auf allen Ports.
  - Deaktiviert
    Deaktiviert die Funktion Q-in-Q VLAN-Tunnel auf allen Ports.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Aktivieren oder deaktivieren Sie die Funktion für diesen Port.

Mirroring

Grundlagen

Mirroring

Das Gerät bietet die Möglichkeit, ein- oder ausgehende Datenströme parallel auf andere Schnittstellen zur Analyse oder Beobachtung auszuleiten. Dabei gibt es keine Rückwirkung auf die betrachteten Datenströme. Das Verfahren wird Mirroring genannt. In diesem Menüabschnitt schalten Sie das Mirroring ein oder aus und stellen die Parameter ein.

Ports spiegeln

Einen Port spiegeln bedeutet, dass der Datenverkehr an einem Port (gespiegelter Port) des IE-Switches auf einen anderen Port (Monitor-Port) kopiert wird. Sie können einen oder mehrere Ports auf einen Monitor-Port spiegeln.

Wird am Monitor-Port ein Protokollanalysator angeschlossen, kann damit der Datenverkehr am gespiegelten Port aufgezeichnet werden, ohne dass die Verbindung dort unterbrochen wird. Dadurch ist eine rückwirkungsfreie Untersuchung des Datenverkehrs möglich. Voraussetzung hierfür ist, dass am Gerät ein freier Port als Monitor-Port zur Verfügung steht.

Allgemein

Hinweis

Es kann nicht garantiert werden, dass beim Mirroring des Datenverkehrs alle Pakete gespiegelt werden. Dies ist insbesondere abhängig von der Last auf den gespiegelten Ports und der Anzahl der Sessions. Um die höchste Genauigkeit zu erzielen, wird die Begrenzung auf eine Session empfohlen.

Mirroring Allgemein

Auf dieser Seite können Sie den Funktion "Mirroring" ein- bzw. ausschalten und die Basiseinstellungen vornehmen.

Hinweis

Das Spiegeln eines Ports funktioniert nicht über Switch-Core-Grenzen. Beachten Sie hierzu auch die Betriebsanleitung des Geräts.

Hinweis

Sie müssen die Portspiegelung ausschalten, wenn Sie an den Monitor-Port ein normales Endgerät anschließen. Dies gilt nicht für den Function Extender BUS ANALYZER Agent XM-400.

Datenrate beachten

Wenn die maximale Datenrate des gespiegelten Ports höher ist als die des Monitor-Ports, kann es zu Datenverlusten kommen und der Monitor-Port gibt nicht mehr die Abläufe am gespiegelten Port wieder. Auf einem Monitor-Port können mehrere Ports gleichzeitig gespiegelt werden.

RSPAN

Mit RSPAN (Remote Switched Port Analyzer) können Sie den Datenverkehr einer Mirroring-Session über ein VLAN an den Monitor-Port leiten. Auf dem RSPAN VLAN wird der gespiegelte Datenverkehr nicht durch andere Daten gestört.

Frames, die direkt an die Mirroring-Quelle adressiert sind, können nicht auf den RSPAN-Ziel-Port gespiegelt werden.

Einstellungen

- **Mirroring**
  Aktivieren oder deaktivieren Sie die Spiegelung des Datenverkehrs.
- **Monitor-Barrier**
  Aktivieren oder deaktivieren Sie die Einstellung, um die Kommunikation über den Monitor Port einzuschränken.

  Hinweis

  Auswirkungen von Monitor Barrier

  Wenn Sie diese Einstellung einschalten, ist das Management des Switches über den Monitor- Port nicht mehr erreichbar. Folgende portspezifische Funktionen werden geändert:
  - DCP forwarding wird ausgeschaltet
  - LLDP wird ausgeschaltet
  - Unicast- Multicast- und Broadcast-Blocking werden eingeschaltet

  Die vorherigen Zustände dieser Funktionen werden nach Beendigung von Monitor Barrier nicht wieder hergestellt. Sie werden auf die Default-Werte zurückgesetzt und müssen eventuell neu konfiguriert werden.

  Sie können diese Funktionen manuell konfigurieren, auch wenn Monitor Barrier eingeschaltet ist. Sie erlauben damit aber auch wieder den entsprechenden Datenverkehr auf dem Monitor-Port. Wenn Sie dies nicht wünschen, achten Sie darauf, dass nur der zu beobachtende Datenverkehr auf die Schnittstelle geleitet wird.

  Wird Mirroring ausgeschaltet, dann werden die genannten portspezifischen Funktionen auf die Default-Werte zurückgesetzt. Das Zurücksetzen erfolgt unabhängig davon, ob die Funktionen manuell oder automatisch durch das Einschalten von "Monitor Barrier" konfiguriert wurden.

  Hinweis

  Function Extender BUS ANALYZER Agent XM-400

  Wenn der Zielport ein Port des Function Extender BUS ANALYZER Agent XM-400 ist, dann ist die Option Monitor-Barrier immer aktivert. Das Aktivieren erfolgt unabhängig davon, ob das Optionskästchen aktiviert oder deaktiviert wurde.

- **RSPAN-VLAN-ID**
  Wählen Sie ein VLAN aus, auf dem der Datenverkehr einer Mirroring-Session ungestört übertragen wird.

Die Tabelle für die Basiseinstellungen enthält folgende Felder:

- Session-ID
  Die Session-ID wird automatisch vergeben, wenn ein neuer Eintrag angelegt wird.
- Session-Typ
  Wählen Sie den gewünschten Eintrag aus:
  - '-'
    keine
  - Port basiert
    Port-basiertes Mirroring.
  - VLAN
    VLAN-basiertes Mirroring.
  - MAC ACL
    Mirroring der MAC Access Control List.
  - IP ACL
    Mirroring der IP Access Control List.

  Hinweis

  Wenn Sie eine Session des Typs "VLAN", "MAC ACL" oder "IP ACL" angelegt haben, können Sie keine weiteren Sessions anlegen. Von dem Typ "Port-based" können Sie bis zu 7 Sessions anlegen.

  Hinweis

  Wenn der "Session Type" einer bestehenden Session geändert wird, gehen alle bisherigen Konfigurationen dieser Session verloren.

- Status
  Zeigt an, ob die Spiegelung aktiv ist oder nicht.
- Hardware-Index
  Wenn Sie aus einem VLAN mehr als einen Quell-Port für das Port-basierte Egress-Mirroring auswählen, werden unbekannte Unicast- und Multicast-Telegramme sowie Broadcast-Telegramme nur einmal an den Ziel-Port weitergeleitet. Bei mehreren Sessions sind die entsprechenden Telegramme nur in einer Session sichtbar. Sie werden nur an dem Ziel-Port mit dem niedrigsten Hardware-Index gespiegelt.
- Ziel-Port
  Wählen Sie den Ziel-Port aus, auf den in dieser Session gespiegelt wird.

  Hinweis

  Function Extender BUS ANALYZER Agent XM-400

  Wenn Sie einen Funktion Extender BUS ANALYZER Agent XM-400 mit einem Grundgerät SCALANCE XM-400 verbinden, können Sie bis zu vier Ports des Function Extender BUS ANALYZER Agent XM-400 als Zielports auswählen. Der Function Extender BUS ANALYZER Agent XM-400 unterstützt Port-basiertes, VLAN-basiertes, MAC ACL-basiertes und IP ACL-basiertes Mirroring.

- RSPAN
  Aktivieren bzw. deaktivieren Sie RSPAN für eine Session.

Port

Ports spiegeln

Sie können die Einstellungen auf dieser Seite nur dann konfigurieren, wenn auf dem Register "Allgemein" zuvor eine Session-ID mit dem Session-Type "Port-basiert" erzeugt wurde.

Beschreibung der angezeigten Felder

- Session-ID
  Wählen Sie die Session aus, die überwacht werden soll. Bis zu 7 parallele Sessions sind möglich, deren Ports sich nicht überschneiden dürfen.

Die Tabelle gliedert sich in folgende Spalten:

- Port
  Zeigt den Port an, der überwacht werden soll.
- Ingress Mirroring
  Aktivieren oder deaktivieren Sie am gewünschten Port das Mithören der eingehenden Pakete.
- Egress Mirroring
  Aktivieren oder deaktivieren Sie am gewünschten Port das Mithören der ausgehenden Pakete.

VLAN

VLAN-Quellen der Portspiegelung

Sie können die Einstellungen auf dieser Seite nur dann konfigurieren, wenn unter dem Reiter "Allgemein" zuvor eine Session ID mit dem Session Type "VLAN" erzeugt wurde.

Auf dieser Seite bestimmen Sie das VLAN, dessen eingehender Datenverkehr auf den Monitorport gespiegelt wird.

Es kann vorkommen, dass am Monitor-Port Datenpakete sichtbar sind, die nicht im definierten VLAN empfangen wurden. Diese Datenpakete kommen von Funktionen, die am Gerät aktiviert sind, z. B. SIMATIC Time Client. Um diese Datenpakete beim VLAN-Mirroring zu vermeiden, deaktivieren Sie die entsprechenden Funktionen auf dem Gerät vor einer Aufzeichnung.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Session-ID
  Wählen Sie die Session aus, die überwacht werden soll. Es ist nur eine Session möglich.
- VLAN ID
  Tragen Sie im Eingabefeld "VLAN-ID" die VLAN-ID ein. Wertebereich: 1 ... 4094

Die Tabelle "Ingress-Mirroring" gliedert sich in folgende Spalten:

- VLAN-ID
  Zeigt die VLAN-ID an, für die eingehende Telegramme gespiegelt werden. Die VLAN-ID kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.

MAC Flow

ACL-Filter für die Portspiegelung

Sie können die Einstellungen auf dieser Seite nur dann konfigurieren, wenn unter dem Reiter "Allgemein" zuvor eine Session ID mit dem Session Type "MAC ACL" erzeugt wurde.

Der MAC ACL Filter bestimmt, welche Daten am Monitor-Port bereitgestellt werden.

Beschreibung der angezeigten Felder

- Session-ID
  Wählen Sie die Session aus, die überwacht werden soll. Es ist nur eine Session möglich.

Die Tabelle gliedert sich in folgende Spalten:

- ACL-Filternummer
  Zeigt die Nummer des ACL-Filters an. Den MAC ACL-Filter konfigurieren Sie unter "Security > MAC ACL".
- Ingress Mirroring
  Zeigt an, ob eingehende Pakete gespiegelt werden.

  Hinweis

  Regeln

  Die ausgewählte Regel wird nur dann aktiv, wenn Sie für mindestens eine Schnittstelle festlegen, nach welchen ACL-Regeln die eingehenden Pakete gefiltert werden. Die Einstellungen konfigurieren Sie unter "Security > MAC ACL > Ingress-Regeln.

- Quell-MAC-Adresse
  Zeigt die MAC-Adresse des Absenders an.
- Ziel-MAC-Adresse
  Zeigt die MAC-Adresse des Empfängers an.
- Ingress-Schnittstellen
  Zeigt alle Schnittstellen, für die diese Regel gilt. Der ACL-Filter bestimmt, welche eingehenden Datenströme auf den Monitor-Port (Ziel-Port) gespiegelt werden.
- Egress-Schnittstellen
  Zeigt alle Schnittstellen an, für die diese Regel gilt..

IP Flow

ACL-Filter für die Portspiegelung

Sie können die Einstellungen auf dieser Seite nur dann konfigurieren, wenn unter dem Reiter "Allgemein" zuvor eine Session ID mit dem Session Type "IP ACL" erzeugt wurde.

Der ACL-Filter bestimmt, welche Daten am Monitor-Port bereitgestellt werden.

Beschreibung der angezeigten Felder

- Session-ID
  Wählen Sie die Session aus, die überwacht werden soll. Es ist nur eine Session möglich.

Die Tabelle gliedert sich in folgende Spalten:

- ACL-Filternummer
  Zeigt die Nummer des ACL-Filters an.
- Ingress Mirroring
  Zeigt an, ob eingehende Pakete gespiegelt werden.

Hinweis

Regeln

Die ausgewählte Regel wird nur dann aktiv, wenn Sie für mindestens eine Schnittstelle festlegen, nach welchen ACL-Regeln die eingehenden Pakete gefiltert werden. Die Einstellungen konfigurieren Sie unter "Security > IP ACL > Ingress-Regeln".

- Quell-IP-Adresse
  Zeigt die IPv4-Adresse des Absenders an.
- Quell-Subnetzmaske
  Zeigt die Subnetzmaske des Absenders an.
- Ziel-IP-Adresse
  Zeigt die IPv4-Adresse des Empfängers an.
- Ziel-Subnetzmaske
  Zeigt die Subnetzmaske des Empfängers an.
- Ingress-Schnittstellen
  Zeigt alle Schnittstellen an, für die diese Regel gilt. Der ACL-Filter bestimmt, welche eingehenden Datenströme auf den Monitor-Port (Ziel-Port) gespiegelt werden.
- Egress-Schnittstelle
  Zeigt alle Schnittstellen an, für die diese Regel gilt. Der ACL-Filter bestimmt, welche ausgehenden Datenströme auf den Monitor-Port (Ziel-Port) gespiegelt werden.

Dynamic MAC Aging

Protokolleinstellungen und Switch-Funktionalität

Das Gerät lernt automatisch die Quelladressen der angeschlossenen Teilnehmer. Diese Information wird dazu benutzt, um Datentelegramme gezielt an die betroffenen Teilnehmer weiterzuleiten. Die Netzlast für die anderen Teilnehmer wird reduziert.
Erhält ein Gerät innerhalb einer bestimmten Zeitspanne kein Telegramm, dessen Quelladresse mit einer gelernten Adresse übereinstimmt, dann löscht es die gelernte Adresse. Dieser Mechanismus wird als "Aging" bezeichnet. Durch Aging wird verhindert, dass Telegramme fehlgeleitet werden, wenn z.B. ein Endgerät (beispielsweise ein Programmiergerät) an einen anderen Port angeschlossen wird. Wenn die Option nicht aktiviert ist, löscht ein Gerät gelernte Adressen nicht automatisch.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Dynamic MAC Aging
  Aktivieren oder deaktivieren Sie die Funktion zum automatischen Aging von gelernten MAC-Adressen.
- Aging Time [s]
  Tragen Sie die Zeitspanne in Sekunden ein. Nach dieser Zeitspanne wird eine gelernte Adresse gelöscht, wenn das Gerät keine weiteren Telegramme von dieser Absenderadresse mehr empfängt. Wertebereich: 10 - 630 Sekunden

Ringredundanz

Ring

Werkseinstellungen

Die Werkseinstellung definiert als Redundanzverfahren MSTP und folgende Ring-Ports:

Geräte	Werkseinstellung Ring-Ports
SCALANCE XB208 und XB216	P0.1 und P0.2
SCALANCE XB205-3	P0.7 und P0.8
SCALANCE XB213-3	P0.15 und P0.16
SCALANCE XC206-2SFP, XC208, XC216 und XC224	P0.1 und P0.2
SCALANCE XC206-2	P0.7 und P0.8
SCALANCE XF-200BA	P1.1 und P2.1
SCALANCE XP208	P0.1 und P0.2
SCALANCE XP216	P0.10 und P0.12
SCALANCE XR324WG	P0.1 und P0.2
SCALANCE XR328-4C WG	P0.25 und P0.26

Geräte	Werkseinstellung Ring-Ports
SCALANCE XM-400	P1.1 und P1.2
SCALANCE XR-500	P0.1 und P0.2

Ringredundanz aktivieren

Sie können die Ringredundanz wie folgt aktivieren:

- über das WBM
- über das CLI
- über den SELECT/SET-Taster
- über einen PNIO-Konfigurations-Download

Einstellungen

- **Ringredundanz**
  Wenn Sie das Optionskästchen "Ringredundanz" aktivieren, schalten Sie die Ringredundanz ein. Es werden die auf dieser Seite eingestellten Ringports verwendet.
- **Ringredundanzverfahren**
  Legen Sie die Betriebsart der Ringredundanz fest. Folgende Betriebsarten stehen zur Verfügung:
  - **Automatic Redundancy Detection**
    Wählen Sie diese Einstellung, um eine automatische Konfiguration der Redundanzbetriebsart vorzunehmen.
    Im Modus "Automatic Redundancy Detection" stellt das Gerät automatisch fest, ob sich ein Gerät mit der Rolle "HRP Manager" im Ring befindet. Ist dies der Fall, so nimmt das Gerät die Rolle "HRP Client" ein. Wird kein HRP Manager gefunden, so handeln alle Geräte mit der Einstellung "Automatic Redundancy Detection" oder "MRP Auto-Manager" untereinander aus, welches Gerät die Rolle "MRP Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP Manager". Die übrigen Geräte stellen sich automatisch auf die Betriebsart "MRP Client" ein.
  - **MRP Auto Manager**
    Im Modus "MRP Auto-Manager" handeln die Geräte untereinander aus, welches Gerät die Rolle "MRP Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP Manager". Die übrigen Geräte stellen sich automatisch auf die Betriebsart "MRP Client" ein. Im Gegensatz zur Einstellung "Automatic Redundancy Detection" sind die Geräte nicht in der Lage zu erkennen, ob ein HRP-Manager im Ring ist.

    Hinweis

    MRP-Projektierung in STEP 7

    Wenn Sie über STEP 7 die Rolle "Manager (Auto)" oder "Manager" für das Gerät einstellen, wird auf dieser WBM-Seite in beiden Fällen "MRP Auto-Manager" angezeigt. In der Anzeige im CLI wird zwischen den beiden Rollen unterschieden.
  - **MRP Client**
    Das Gerät nimmt die Rolle MRP-Client ein.
  - **HRP Client**
    Das Gerät nimmt die Rolle HRP-Client ein.
  - **HRP Manager**
    Das Gerät nimmt die Rolle HRP-Manager ein. Bei der Projektierung eines HRP-Rings muss ein Gerät als HRP-Manager eingestellt werden. Bei allen übrigen Geräten muss "HRP Client" oder "Automatic Redundancy Detection" eingestellt sein.
- **Ring-Ports**
  Hier stellen Sie die Ports ein, die bei der Ringredundanz als Ringports verwendet werden sollen. Der Ringport, den Sie im linken Drop-down-Menü auswählen, ist bei HRP der "Isolated Port". Bei MRP wird beim Linkaufbau entschieden, welcher Port der "Isolated Port" wird.

- **Observer**
  Aktivieren/Deaktivieren Sie den Observer. Die Funktion "Observer" steht nur in HRP-Ringen zur Verfügung.
  Der Ringport, der im linken Drop-down-Menü ausgewählt ist, wird an den "Isolated Port" eines HRP-Managers angeschlossen.
  Der Observer überwacht Fehlfunktionen des Redundanzmanagers oder Fehlkonfigurationen eines HRP-Rings.
  Wenn der Observer aktiviert ist, dann kann er bei erkannten Fehlern den angeschlossenen Ring unterbrechen. Dazu schaltet der Observer einen Ringport in den Zustand "blocking". Wenn der Fehler aufgelöst wird, schaltet der Observer den Port wieder frei.
- **Observer neu starten** (nur Online verfügbar)
  Wenn viele Fehler schnell hintereinander auftreten, schaltet der Observer seinen Port nicht mehr selbstständig frei. Der Ringport bleibt dauerhaft im Zustand "blocking". Dies wird durch die Fehler-LED und einen Meldetext signalisiert.
  Nach der Fehlerbehebung können Sie den Port über die Schatlfläche "Observer neu starten" wieder freischalten.

Werkseinstellungen wiederherstellen

Werden die Werkseinstellungen (Reset to Factory Defaults) wiederhergestellt, dann ist die Ringredundanz deaktiviert und die werkseitigen Ringports werden ausgewählt. Dies kann kreisende Telegramme und den Ausfall des Datenverkehrs verursachen, wenn in einer vorherigen Projektierung andere Einstellungen verwendet wurden.

Zustand der Ring-Ports beim Redundanzmanager tauschen

Wenn Sie einen Redundanzmanager konfigurieren, stellen Sie den Zustand der Ring-Ports fest ein. Der erste Ring-Port geht in den Zustand "blocking" und der zweite Ring-Port in den Zustand "forwarding". Bei aktivierter Ringredundanz können Sie den Zustand dieser Ring-Ports tauschen.

Hinweis

Achten Sie darauf, dass Sie zunächst den Ring öffnen, damit es nicht zu kreisenden Telegrammen kommt.

Ring-Ports ändern

Sie können die Ring-Ports ändern, ohne den Ring zu öffnen. Um die Ring-Ports zu ändern, gehen Sie wie folgt vor:

- Wechseln Sie auf die Seite "Layer 2 > Spanning Tree > CIST Port".
- Deaktivieren Sie die Ports im Spanning Tree, die Sie als Ring-Ports konfigurieren wollen.
- Wechseln Sie auf die Seite "Layer 2 > Ringredundanz > Ring".
- Wählen Sie die neuen Ring-Ports aus.
- Stecken Sie die Kabel um.

- Wechseln Sie auf die Seite "Layer 2 > Spanning Tree > CIST Port".
- Aktivieren Sie die Ports im Spanning Tree, die keine Ring-Ports mehr sind.

Standby

Redundante Kopplung von Ringen

Die Standby-Redundanz erlaubt die redundante Kopplung von HRP-Ringen. Um eine Standby-Verbindung zu etablieren, konfigurieren Sie innerhalb eines Rings zwei benachbarte Geräte als Standby-Master bzw. Standby-Slave. Der Standby-Master und der Standby-Slave müssen über parallele Leitungen mit zwei Geräten in einem anderen Ring verbunden werden. Im ungestörten Zustand laufen Nachrichten zwischen den beiden Ringen über den Master. Wenn die Leitung des Masters gestört wird, übernimmt der Slave die Weiterleitung von Nachrichten zwischen beiden Ringen. Aktivieren sie die Standby-Redundanz für beide Standby-Partner und wählen Sie, über welche Ports das Gerät mit den zu koppelnden Ringen verbunden ist. Als "Standby Connection Name" muss für beide Partner ein eindeutiger Name im Ring vergeben werden, mit dem die beiden zusammengehörenden Geräte als Standby-Partner identifiziert werden.

Hinweis

Um die Funktion zu nutzen, muss HRP aktiviert sein.

Der Standby Manager erfordert immer einen aktivierten HRP-Client oder HRP-Manager.

Hinweis

Standby-Master-Kopplung mit optischer 100 MBit/s-Verbindung

Während der Koppelpartner eines Standby-Masters mit einer optischen 100 MBit/s-Verbindung wieder anläuft, trennen Sie die Verbindung zwischen dem Standby-Master und seinem Koppelpartner physikalisch.

Beschreibung der angezeigten Felder

- **Standby**
  Aktivieren oder deaktivieren Sie diese Funktion.

  Hinweis

  Sind zwei Geräte über Standby-Funktion gekoppelt, muss die Funktion "Standby" an beiden Geräten aktiviert sein.

- **Standby Port**
  Wählen Sie aus, welcher Port Standby-Port ist. Über den Standby-Port erfolgt die Kopplung zum anderen Ring.
  Hinweis

  Hinweis

  Standby-Ports im Spanning Tree

  Bevor Sie den Port als Standby-Port aktivieren können, müssen Sie den Port im Spanning Tree deaktivieren.

  1. Wechseln Sie auf die Seite "Layer 2 > Spanning Tree > CIST Port".

  2. Deaktivieren Sie die Ports im Spanning Tree, die Sie als Standby-Ports konfigurieren wollen.

  Der Standby-Port ist an der Umleitung des Datenverkehrs beteiligt. Im ungestörten Fall ist nur der Standby-Port des Master aktiv und übernimmt den Datenverkehr in den angeschlossenen HRP-Ring bzw. HRP-Linie.
  Wenn der Master oder die Ethernet-Verbindung (Link) eines Standby-Port des Master ausfällt, dann wird der Standby-Port des Master abgeschaltet und der Standby-Port des Slave aktiviert. Damit wird wieder eine funktionierende Ethernet-Verbindung in das angeschlossene Netzsegment (HRP-Ring bzw. HRP-Linie) hergestellt.

- **Name der Standby-Verbindung**
  Durch diesen Namen wird das Master-/Slave-Geräepaar definiert. Beide Geräe müsen im gleichen Ring liegen.
  Tragen Sie hier den Namen für die Standby-Verbindung ein. Dieser muss identisch sein mit dem beim Standby-Partner eingetragenen Namen. Der Name kann frei gewählt werden, darf im ganzen Netz jedoch nur für ein Gerätepaar verwendet werden.

- **Standby-Master-Betrieb erzwingen**
  Wenn aktiviert, wird das Gerät unabhägig von seiner MACAdresse als Standby-Master konfiguriert.
  - Wenn bei keinem der beiden Geräe, für die der Standby-Manager eingeschaltet ist, dieses Optionskätchen markiert ist, dann übernimmt im fehlerfreien Zustand das Gerät mit der höheren MAC-Adresse die Funktion des Standby-Masters.
  - Wenn diese Option bei beiden Geräen ausgewält ist, oder wenn die Eigenschaft "Standby-Master-Betrieb erzwingen" nur von einem Gerät unterstüzt wird, dann wird der Standby-Master ebenfalls anhand der MAC-Adresse ausgewält.

  Wichtig ist diese Art der Zuordnung insbesondere bei einem Gerätetausch. Abhägig von den MAC-Adressen kann das bisherige Gerät mit Slave-Funktion die Rolle des Standby-Masters üernehmen.

  Hinweis

  Wenn die Option „Standby-Master-Betrieb erzwingen“ an beiden Geräen einer Standby-Kopplung aktiviert ist, kann es zu kreisenden Telegrammen und damit dem Ausfall des Datenverkehrs kommen. Aktivieren Sie die Option „Standby-Master-Betrieb erzwingen“ nur an einem Geräe einer Standby-Kopplung.

- **Warten auf Standby-Partner**
  - Aktiviert
    Eine Standby-Verbindung wird erst aktiviert, wenn der Standby-Master und der Standby-Slave sowie deren Standby-Partner eine Verbindung aufgebaut haben. Dadurch wird sichergestellt, dass die redundante Verbindung auch wirklich zur Verfüung steht, bevor die Kommunikation üer eine Standby-Verbindung aktiviert wird.
  - Deaktiviert
    Eine Standby-Verbindung wird aktiviert, auch wenn der Standby-Master noch keine Verbindung zu dem Standby-Slave aufgebaut hat.
    Dies kann kreisende Telegramme und den Ausfall des Datenverkehrs verursachen, wenn bereits eine andere Standby-Verbindung aktiviert wurde. Mehrere Standby-Verbindungen könen z. B. durch Konfigurationsfehler entstehen, wenn bei Standby-Master und Standby-Slave unterschiedliche Standby-Verbindungs-Namen vergeben wurden.

Link Check

Voraussetzungen

- Sie können Link Check nicht auf Ports mit 10 GBit/s aktivieren.
- Sie können die Funktion Link Check nur bei optischen Ring-Ports eines HRP- oder MRP-Rings aktivieren.

Hinweis

Medienmodule tauschen: optisch → elektrisch

Wenn Sie Link Check auf einem optischen Port eines Medienmoduls betreiben, beachten Sie Folgendes:

- Link Check ist auf dem optischen Port eines Medienmoduls aktiviert.
- Sie wollen das Medienmodul gegen ein Modul ohne optische Ports austauschen.

1. Deaktivieren Sie Link Check auf den Ports des gesteckten Moduls.
2. Tauschen Sie das Medienmodul.

- Link Check muss jeweils auf zwei benachbarten Geräten (Verbindungspartnern) innerhalb eines HRP- oder MRP-Rings aktiviert werden.
- Die Ring-Ports, an denen Sie Link Check aktivieren, müssen miteinander verbunden sein.

Überwachung optischer Verbindungen im Ring

Mit der Funktion Link Check können Sie die Übertragungsqualität optischer Strecken innerhalb eines HRP- oder MRP-Rings überwachen, gestörte Übertragungsstrecken identifizieren und unter bestimmten Bedingungen abschalten. Wenn die gestörte Strecke abgeschaltet ist, kann der Redundanzmanager den Ring schließen und die Kommunikation wiederherstellen.

ACHTUNG

Stellen Sie sicher, dass die Telegramme, die von Link Check zur Überwachung der optischen Verbindungen genutzt werden, nicht durch eine Überlast an hochprioren Telegrammen im Netzwerk verdrängt werden.

Eine Überlast an hochprioren Telegrammen kann z. B. folgende Ursachen haben:

- Netzwerkschleifen, durch die es zu einer Vervielfältigung der hochprioren Telegramme kommen kann
- Ändern der Prioritäten für die Weiterleitung von Telegrammen

Hinweis

Aktivieren Sie Link Check nicht nur bei einem von zwei Verbindungspartnern. Dies kann zu fehlerhaftem Verhalten führen.

Hinweis

Wenn Link Check bei allen Geräten eines Rings zeitgleich eingeschaltet ist und mehrere Verbindungen innerhalb des Rings gestört sind, führt dies zu einer Fragmentierung des Rings.

1. Schalten Sie bei der Inbetriebnahme die Funktion Link Check für eine Verbindungsstrecke nach der anderen ein, indem Sie jeweils bei den beiden Verbindungspartnern, die an eine Strecke angeschlossen sind, Link Check aktivieren.
2. Um eine fehlerfreie Verbindung zu gewährleisten, warten Sie jeweils 1 min, bevor Sie Link Check für die nächste Verbindungsstrecke aktivieren.

Beschreibung der angezeigten Felder

Die Tabelle enthält folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Mit diesem Optionskästchen aktivieren bzw. deaktivieren Sie die Funktion Link Check für einen Port.
  Bei aktiver Verbindungsüberwachung können Sie die Anzahl der gesendeten und empfangenen Link Check-Testtelegrammen auf der Seite "Information > Redundanz > Link Check" einsehen.
- **Zurücksetzen**
  Nach dem Zurücksetzen von Link Check wird die Funktion auf dem Port erneut gestartet und die Statistik wird zurückgesetzt.
  Wenn Sie die Schaltfläche "Zurücksetzen" verwenden, muss das Zurücksetzen auf beiden Verbindungspartnern innerhalb von 30 s erfolgen.

  Hinweis

  Wenn Sie die Schaltfläche "Zurücksetzen" verwenden, kann es zu kurzfristiger Schleifenbildungen und damit zum Ausfall des Datenverkehrs kommen. Die Schleife wird automatisch wieder aufgelöst.

  Wenn das in Ihrem Anwendungsfall nicht akzeptabel ist, setzen Sie Link Check zurück, indem Sie das Verbindungskabel ziehen und wieder stecken.

MRP-Interconnection

Redundante Kopplung von Ringen

Auf dieser Seite erstelle, löschen und konfigurieren Sie MRP-Interconnect-Verbindungen.

Beschreibung der angezeigten Felder

- MRP-Interconnection
  Markieren Sie dieses Optionskästchen, um MRP-Interconnect für das Gerät zu aktivieren. Sie können MRP-Interconnection nur einschalten, wenn folgende Voraussetzungen erfüllt sind:
  - Ringredundanz ist eingeschaltet.
  - Als Ringredundanzverfahren wird "MRP-Manager" oder "MRP-Client" eingesetzt.
  - Es gibt eine aktivierte MRP-Interconnect-Verbindung.

Die Tabelle gliedert sich in folgende Spalten:

- Selektieren
  Wählen Sie die Zeile, die Sie löschen wollen.
- Interconnection-ID
  Geben Sie die ID der MRP-Interconnection-Verbindung ein. Beachten Sie bei der Vergabe der ID folgende Regeln:
  - Die Interconnection-ID darf nicht 0 sein.
  - Für alle vier Geräte, die für die Kopplung der Ringe eingesetzt werden, müssen Sie die gleiche Interconnection-ID konfigurieren.
- Interconnection-Name
  Geben Sie einen beliebigen Namen für die MRP-Interconnection-Verbindung ein. Sie können für die vier Geräte, die für die Kopplung der Ringe eingesetzt werden, auch unterschiedliche Namen festlegen. Zulässige Zeichen für diesen Namen sind die Buchstaben 'A' bis 'Z' und 'a' bis 'z' sowie die Ziffern '0' bis '9' und das Zeichen '-'. Das erste und letzte Zeichen des Namens darf kein Bindestrich sein. Der Interconnection-Name muss mindestens aus einem Zeichen bestehen und darf maximal 240 Zeichen lang sein.
- Interconnection-Port
  Wählen Sie aus dieser Klappliste den Port aus, der für die MRP-Interconnection-Verbindung genutzt wird. Beachten Sie dabei folgende Einschränkungen:
  - Der Port darf nicht abgeschaltet oder geblockt sein. Die Funktion "Unicast Blocking" darf für den Port nicht aktiviert sein.
  - Der Port darf nicht für eine Link Aggregation genutzt werden.
  - Der Port darf nicht Monitor-Port der Funktion "Mirroring" sein.
  - Der Port darf kein Spanning Tree-Port sein.
  - Der Port darf kein Ring-Port sein.
  - Der Port darf kein Router-Port sein.
  - Der Port muss Mitglied in "vlan1" sein.
  - Für den Port muss LLDP (Tx und Rx) aktiviert sein.
  - Der Port darf kein 802.1X Authenticator-Port sein.
  - Der Port darf kein 802.1X Supplicant-Port sein.

- **Warten**
  Wenn dieses Optionskästchen aktiviert ist, wird die Konfiguration aller vier Geräte einer MRP-Interconnection-Kopplung abgewartet, bevor die Datenübertragung aufgenommen wird.
- **Rolle/Position**
  Wählen Sie aus dieser Klappliste die Rolle und, bei der Rolle "Client", die Position des Clients aus. Es gibt folgende Auswahlmöglichkeiten:
  - Manager
  - Primary Client
  - Secondary Client
- **Status**
  Markieren Sie dieses Optionskästchen, um die MRP-Interconnection-Verbindung zu aktivieren. Beachten Sie dabei folgende Regeln:
  - Wenn nicht mindesten eine MRP-Interconnection-Verbindung aktiviert ist, können Sie MRP-Interconnection für das Gerät nicht einschalten.
  - Es darf maximal eine MRP-Interconnection-Verbindung aktiviert sein.

Spanning Tree

Allgemein

Allgemeine Einstellungen von MSTP

Dies ist die Basisseite zu Spanning Tree. Wählen Sie aus der Klappliste den Kompatibilitätsmodus aus. In der jeweiligen Konfigurationsseite der Funktionen sind weitere Einstellungen möglich. Je nach Kompatibilitätsmodus können Sie in der jeweiligen Konfigurationsseite die entsprechende Funktion konfigurieren.

Einstellungen

- **Spanning Tree**
  Aktivieren oder deaktivieren Sie Spanning Tree.
- **Protokollkompatibilität**
  Wählen Sie den Kompatibiltäts-Modus von MSTP aus. Z. B. wenn Sie RSTP wählen, dann verhält sich MSTP wie ein RSTP.
  Folgende Einstellungen gibt es:
  - STP
  - RSTP
  - MSTP

- **RSTP+**
  RSTP+ ermöglicht die Kopplung eines Netzsegments, in dem Spanning Tree aktiviert ist, mit einem MRP-Ring. Stellen Sie sicher, dass folgende Voraussetzungen erfüllt sind, bevor Sie dieses Kontrollkästchen aktivieren:
  - MRP muss als Redundanverfahren aktiviert sein.
  - Bei aktivierter Ringredundanz müssen Sie die Ringports für Spanning Tree deaktivieren.

  Wenn Sie RSTP+ aktivieren, werden die Ringports sowohl Teil des MRP-Rings als auch Teil des Spanning Tree-Netzsegments.
- **RSTP+ MRP-Interconnection-Domain-ID**
  Konfigurieren Sie hier die MRP-Interconnection-Domain-ID für RSTP+. Dieser Wert darf nicht mit der MRP-Interconnection-Domain-ID übereinstimmen, die für die aktive MRP-Interconnection-Verbindung konfiguriert wurde.

Siehe auch

Allgemein (Seite 2317)

MST Port (Seite 2312)

MST Allgemein (Seite 2311)

CIST Allgemein/ST Allgemein

Die Seite besteht aus folgenden Teilen:

- Der linke Teil der Seite zeigt die Konfiguration des Gerätes.
- Der mittlere Teil zeigt die Konfiguration der Root-Bridge, wie sie aus Spanning Tree-Telegrammen abgeleitet werden kann, die ein Gerät empfangen hat.
- Der rechte Teil zeigt die Konfiguration der Regionalen Root-Bridge, wie sie aus den MSTPTelegrammen abgeleitet werden kann. Die angezeigten Daten sind nur dann sichtbar, wenn auf der Seite "Allgemein" "Spanning Tree" aktiviert und bei "Protokollkompatibilität" "MSTP" eingestellt ist. Das gilt auch für den Parameter "Bridge Max Hop Count". Wenn das Gerät eine Root-Bridge ist, stimmen die Informationen des linken und des rechten Teils überein.

Beschreibung der angezeigten Felder

- **Bridge-Priorität / Root-Priorität** (Nur Online verfügbar)
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root-Bridge wird. Die Bridge mit der höchsten Priorität wird Root-Bridge. Je kleiner der Wert, desto höher die Priorität. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, wird das Gerät Root-Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root-Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein.
- **Bridge-Adresse / Root-Adresse** (Nur Online verfügbar)
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.

- **Root-Port** (Nur Online verfügbar)
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten** (Nur Online verfügbar)
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge.
- **Topologieänderungen** / **Letzte Topologieänderung** (Nur Online verfügbar)
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit dem letzten Hochlauf. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz sec hinter der Zahlenangabe
  - Minuten: Zusatz min hinter der Zahlenangabe
  - Stunde: Zusatz hr hinter der Zahlenangabe
- **Bridge Hello Time [s] / Root Hello Time [s]** (Nur Online verfügbar)
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei Konfigurationstelegrammen ist die "Hello Time". Werkseinstellung: 2 Sekunden

  Hinweis

  Die Einstellung der "Bridge Hello Time" ist nur mit der Protokollkompatibilität RSTP möglich. Wenn die Protokollkompatibilität MSTP eingestellt ist, wird der Parameter "Hello Time" auf der Seite "Layer 2 > Spanning Tree > CIST-Port" verwendet.

- **Bridge Forward Delay [s] / Root Forward Delay [s]**
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort angewendet, sondern erst nach dem im Parameter "Forward Delay" festgelegten Zeitraum. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.
  Werkseinstellung: 15 Sekunden
- **Bridge Max Age / Root Max Age** (Nur Online verfügbar)
  Wenn die BPDU älter ist als das angegebene "Max Age", wird sie verworfen.
  Werkseinstellung: 20 Sekunden
- **Root-Priorität regional** (Nur Online verfügbar)
  Beschreibung siehe Bridge-Priorität / Root-Priorität
- **Root-Adresse regional** (Nur Online verfügbar)
  Die MAC-Adresse des Gerätes.
- **Root-Kosten regional** (Nur Online verfügbar)
  Zeigt die Pfadkosten von dem Gerät bis zur regionalen Root-Bridge.
- **Bridge Max Hop Count**
  Dieser Parameter gibt an, wie viele MSTP-Teilnehmer eine BPDU passieren darf. Wird eine MSTP-BPDU empfangen, deren Hop Count den hier konfigurierten Wert übersteigt, wird sie verworfen. Der Standardwert für diesen Parameter beträgt 20.
- **Name der Region**
  Tragen Sie den Namen der MSTP-Region ein, zu der dieses Gerät gehört. Defaultmäßig ist hier die MAC-Adresse des Gerätes eingetragen. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.

- **Version der Region**
  Tragen Sie die Versionsnummer der MSTP-Region ein, in der sich das Gerät befindet. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.
- **Zähler zurücksetzen** (Nur Online verfügbar)
  Klicken Sie auf diese Schaltfläche, um die Zähler auf dieser Seite zurückzusetzen.

CIST-Port/ST-Port

Konfiguration der MSTP-CIST-Ports

In der Tabelle wird beim Aufruf der Seite der aktuelle Stand der Konfiguration der Port-Parameter angezeigt.

Klicken Sie zur Konfiguration in die entsprechenden Felder der Port-Tabelle.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Spanning Tree-Status**
  Wählen Sie die Einstellung aus. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Port ist im Spanning-Tree integriert.
  - Deaktiviert
    Port ist im Spanning-Tree nicht integriert.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Spanning Tree-Status**
  Legen Sie fest, ob der Port im Spanning-Tree integriert ist oder nicht.

Hinweis

Wenn Sie die Einstellung "Spanning Tree-Status" für einen Port deaktivieren, kann es zur Schleifenbildung kommen. Die Topologie muss beachtet werden.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind. Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240. Der Standardwert ist 128.
- **kalk. Kosten**
  Tragen Sie die Wegekostenberechnung ein. Wenn Sie den Wert "0" eintragen, wird im Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.
- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn bei "erm. Kosten" der Wert "0" ist, so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "erm. Kosten" angezeigt. Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten. Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht parametriert werden. Der Parameter "Status" ist abhängig von dem projektierten Protokoll. Beim Status ist Folgendes möglich:
  - Disabled
    Der Port empfängt nur und nimmt nicht am STP, MSTP und RSTP teil.
  - Discarding
    In der Betriebsart "Discarding" werden BPDU-Telegramme empfangen. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    In diesem Status werden sowohl BPDU-Telegramme empfangen als auch gesendet. Der Port ist in den Spanning Tree-Algorithmus einbezogen.
  - Learning
    Vorstufe zum Weiterleitungsstatus, der Port lernt aktiv die Topologie (d. h. die Teilnehmeradressen).
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz, er empfängt und sendet Datentelegramme.

- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Wechsel vom Status "Discarding" zum Status "Forwarding" an
- **Edge-Typ**
  Legen Sie die Art des Edge-Port fest. Sie haben folgende Möglichkeiten:
  – "-"
    Edge-Port ist deaktiviert. Der Port wird wie ein "no-Edge-Port" behandelt.
  – Admin
    Wählen Sie diese Option, wenn sich an diesem Port immer ein Endgerät befindet. Sonst wird bei jeder Verbindungsänderung eine Rekonfiguration des Netzwerks ausgelöst.
  – Auto
    Wählen Sie diese Option, wenn an diesem Port automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist. Beim ersten Verbindungsaufbau wird der Port wie ein "no Edge Port" behandelt.
  – Admin/Auto
    Wählen Sie diese Optionen, wenn Sie an diesem Port eine Kombination aus beiden betreiben. Beim ersten Verbindungsaufbau wird der Port als Edge Port behandelt.
- **Edge** (Nur Online verfügbar)
  Zeigt an, in welchem Status der Port ist.
  – Enabled
    An diesem Port befindet sich ein Endgerät.
  – Disabled
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.

  Bei einem Endgerät kann ein Switch ohne Rücksicht auf Spanning Tree-Telegramme. schneller den Port umschalten. Wird entgegen dieser Einstellung ein Spanning Tree-Telegramm empfangen, wechselt der Port automatisch auf die Einstellung "Deaktiviert" für Switches.
- **P.t.P.-Typ**
  Wählen Sie die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.
  – "-"
    Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
  – P.t.P.
    Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
  – Shared Media
    Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

  Hinweis

  Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

- **Hello Time**
  Tragen Sie das Intervall ein, nach der die Bridge Konfigurationstelegramme (BPDUs) sendet. Standardmäßig sind 2 Sekunden eingestellt. Wertebereich: 1-2 Sekunden

  Hinweis

  Die portspezifische Einstellung der Hello Time ist nur mit der Protokollkompatibilität MSTP möglich. Wenn die Protokollkompatibilität RSTP eingestellt ist, wird der Parameter "Bridge Hello Time" auf der Seite "Layer 2 > Spanning Tree > CIST Allgemein" verwendet.

- **Eingeschränkte Rolle**
  Bei aktiviertem Optionskästchen wird der entsprechende Port nicht als Root-Port ausgewählt, unabhängig vom Wert für die Priorität. Auch der Port mit der niedrigsten Priorität wird bei aktiviertem Optionskästchen nicht zum Root-Port. Aktivieren Sie diese Option nur dann, wenn Sie den Einfluss von Bridges außerhalb des administrierten Bereichs auf die Spanning Tree-Topologie beschränken wollen.
- **Eingeschränktes TCN**
  Bei aktiviertem Optionskästchen leitet der entsprechende Port weder erhaltene noch erkannte Topologieänderungen (Topology Change Notifications) an andere Ports weiter. Aktivieren Sie diese Option nur dann, wenn Sie den Einfluss von Bridges außerhalb des administrierten Bereichs auf die Spanning Tree-Topologie beschränken wollen.
- **Limitiertes TCN**
  Bei aktiviertem Optionskästchen akzeptiert der entsprechende Port erhaltene und erkannte Topologieänderungen, leitet Topologieänderungen aber nicht an andere Ports weiter. Sie können die Optionskästchen in dieser Spalte nur dann aktivieren, wenn folgende Voraussetzungen erfüllt sind:
  - RSTP+ muss aktiviert sein.
  - Das Optionskästchen "Eingeschränktes TCN" muss für diesen Port deaktiviert sein.

  Wenn die genannten Voraussetzungen nicht erfüllt sind, werden die Optionskästchen in dieser Spalte gegraut dargestellt.

MST Allgemein

Multiple Spanning Tree-Konfiguration

Bei MSTP können zusätzlich zu RSTP mehrere VLANs in einem LAN mit eigenen RSTP-Bäumen verwaltet werden.

Beschreibung der angezeigten Felder

- **MSTP-Instanz-ID**
  Tragen Sie die Nummer der MSTP-Instanz ein. Zulässige Werte: 1 - 64

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **MSTP-Instanz-ID**
  Zeigt die Nummer der MSTP-Instanz an.

- **Root-Adresse**
  Zeigt die MAC-Adresse der Root-Bridge an
- **Root-Priorität**
  Zeigt die Priorität der Root-Bridge an.
- **Bridge-Priorität**
  Tragen Sie in dieses Feld die Bridge-Priorität ein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 61440.
- **VLAN-ID**
  Tragen Sie die VlAN-ID ein. Sie können hier auch Bereiche mit Start-ID, "-", End-ID angeben. Mehrere Bereiche oder IDs werden durch "," separiert. Zulässige Werte: 1- 4094

MST Port

Konfiguration der Multiple Spanning Tree Port Parameter

Auf dieser Seite stellen Sie die Parameter für die Ports der konfigurierten Multiple Spanning Tree Instanzen ein.

Beschreibung der angezeigten Felder

Die Seite enthält folgendes Feld:

- **MSTP-Instanz-ID**
  Wählen Sie die ID der MSTP-Instanz aus.

Die Tabelle 1 gliedert sich in folgende Spalten

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports gültig ist.
- **MSTP-Status**
  Wählen Sie die Einstellung aus. Es gibt folgende Einstellmöglchkeiten:
  - Aktiviert
  - Deaktiviert
  - Keine Änderung: Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports und Link-Aggregationen an.
- **MSTP-Instanz-ID**
  ID der MSTP-Instanz.
- **MSTP-Status**
  Klicken Sie in das Optionskästchen, um diese Option zu aktivieren oder zu deaktivieren.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind. Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240. Werkseinstellung: 128
- **Kalk. Kosten**
  Tragen Sie in das Eingabefeld die Wege-Kostenberechnung ein. Wenn Sie hier 0 eintragen, wird im nächsten Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.
- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn bei "Kalk. Kosten" der Wert "0" ist, so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "Kalk. Kosten" angezeigt. Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten. Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht konfiguriert werden. Beim Status ist Folgendes möglich:
  - Discarding
    Der Port tauscht MSTP-Informationen aus, nimmt aber nicht am Datenverkehr teil.
  - Blocked
    Im Blocking-Modus werden BPDU-Telegramme empfangen.
  - Forwarding
    Der Port empfängt und sendet Datentelegramme.
- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Statuswechsel Discarding - Forwarding bzw. Forwarding - Discarding für einen Port an.

Enhanced Passive Listening Compatibility

Aktivieren der Funktion

Auf dieser Seite können Sie passive Listening Compatibility aktivieren.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Enhanced Passive Listening Compatibility**
  Aktivieren oder deaktivieren Sie diese Funktion für das gesamte Gerät.
- **Einstellung**
  - Aktiviert
    Aktiviert die Funktion für alle Ports des Geräts.
  - Deaktiviert
    Deaktivieren die Funktion für alle Ports des Geräts.
  - Keine Änderung
    Keine Änderung
- **In Tabelle übernehmen**
  Schreibt die Einstellung unter "Einstellung" in die nachfolgende Tabelle.

Portgranulare Tabelle:

Wenn die Funktion geräteweit aktiviert ist, aktivieren oder deaktivieren sie diese Funktion auf einzelnen Ports.

- **Port**
  Zeigt den Port des Gerätes an.
- **Einstellung**
  Aktivieren oder deaktivieren Sie die Funktion für diesen Port.

Loop Detection

Mit der Funktion "Loop Detection" legen Sie fest, für welche Ports Schleifenerkennung aktiviert werden soll. Von den betreffenden Ports werden spezielle Testtelegramme, die Loop Detection-Telegramme, gesendet. Wenn diese Telegramme wieder zum Gerät zurück gesendet werden, dann liegt eine Schleife ("Loop") vor.

Von einem "Local Loop" unter Beteiligung dieses Gerätes spricht man, wenn die Telegramme an einem anderen Port desselben Gerätes wieder empfangen werden. Wenn die ausgesendeten Telegramme wieder am gleichen Port empfangen werden, ist eine Schleife "Remote Loop" an anderen Netzkomponenten aufgetreten.

Hinweis

Eine Schleife ist ein Fehler im Netzaufbau, der beseitigt werden muss. Die Schleifenerkennung kann helfen, den Fehler schneller zu finden, behebt ihn jedoch nicht. Die Schleifenerkennung ist nicht dazu geeignet, die Netzwerkverfügbarkeit durch den gezielten Einbau von Schleifen zu erhöhen.

Hinweis

Beachten Sie, dass die Schleifenerkennung nur auf Ports möglich ist, die nicht als Ring-Port oder Standby-Port konfiguriert wurden.

Beschreibung der angezeigten Felder

- **Loop Detection**
  Aktivieren oder deaktivieren Sie die Schleifenerkennung.
- **VLAN Loop Detection**
  Aktivieren oder deaktivieren Sie die Schleifenerkennung bei VLAN.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind
- **Intervall [ms] / Schwellenwert / Timeout [s]/ Remote Reaction / Local Reaction**
  Legen Sie die gewünschten Einstellungen fest.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an.
- **Einstellung**
  Legen Sie fest, wie der Port mit Loop Detection-Telegrammen verfahren soll.

  Hinweis

  Durch die Testtelegramme entsteht zusätzliche Netzlast. Wir empfehlen, nur einzelne Switches, z. B. an den Abzweigungen vom Ring, als "sender" zu konfigurieren und die anderen als "forwarder".

  – sender
    Loop Detection-Telegramme werden ausgesendet und weitergeleitet.
  – forwarder
    Loop Detection-Telegramme von anderen Geräten werden weitergeleitet.
  – blocked
    Die Weiterleitung der Loop-Detection-Telegramme wird blockiert.
- **Intervall [ms]**
  Legt das Sendeintervall für Loop Detection-Telegramme in Millisekunden fest.
- **Schwellenwert**
  Legen Sie durch Eingabe einer Zahl fest, nach wie vielen empfangenen Loop Detection-Telegrammen von einer Schleife ausgegangen wird.
- **Timeout** [s]
  Legen Sie fest, nach wie vielen Sekunden das Gerät automatisch wieder in den Zustand wechselt, in dem es sich vor dem Loop befunden hat. Wenn der Wert "0" eingestellt ist, müssen Sie den Port nach einem Loop manuell wieder aktivieren, mit der Schaltfläche "Rücksetzen".

- **Remote Reaction**
  Legen Sie fest, wie der Port bei Auftreten einer Remote-Loop reagieren soll.
  - Keine Änderung
    Eine Schleife hat keine Auswirkungen auf den Port.
  - Deaktiviert
    Der Port wird geblockt.
- **Local Reaction**
  Legen Sie fest, wie der Port bei Auftreten eines Local Loop reagieren soll.
  - Keine Änderung
    Eine Schleife hat keine Auswirkungen auf den Port.
  - Deaktiviert
    Der Port wird geblockt.
- **Status** (Nur Online verfügbar)
  Zeigt an, ob die Schleifenerkennung für diesen Port ein- oder ausgeschaltet ist.
- **Quell-Port** (Nur Online verfügbar)
  Zeigt den Empfänger-Port des Loop-Detection-Telegramms an, das die letzte Reaktion ausgelöst hat.
- **Quell-VLAN** (Nur Online verfügbar)
  Zeigt die VLAN-ID des Loop-Detection-Telegramms an, das die letzte Reaktion ausgelöst hat.
  Voraussetzung dafür ist, dass die Einstellung "VLAN Loop Detection" aktiviert ist.
- **Rücksetzen** (Nur Online verfügbar)
  Nachdem eine Schleife im Netzwerk beseitigt wurde, klicken Sie auf diese Schaltfläche, um den Port wieder zurücksetzen.

Änderung des konfigurierten Port-Status durch Loop Detection

Die Konfiguration des Port-Status kann durch die Funktion "Loop Detection" verändert werden. Wenn der Administrator z. B. einen Port deaktiviert hat (disabled), kann der Port nach einem Geräteneustart durch "Loop Detection" wieder aktiviert werden (enabled). Der Port-Status "Link down" wird durch "Loop Detection" nicht verändert.

Auswirkungen bei der Konfiguration über STEP7

Die Konfiguration von Spanning Tree kann verändert werden, wenn Sie "Loop Detection" über STEP7 konfigurieren. Wenn Sie "Loop Detection" über STEP7 an einem Port aktivieren, wird an diesem Port automatisch Spanning Tree deaktiviert. Wenn Sie "Loop Detection" über STEP7 für den Port wieder deaktivieren, wird Spanning Tree nicht automatisch aktiviert. Aktivieren Sie Spanning Tree über WBM oder CLI.

Link Aggregation

Allgemein

Bündelung von Netzwerkverbindungen für Redundanz und höhere Bandbreite

Die Link Aggregation nach IEEE 802.3ad erlaubt es, mehrere Verbindungen zwischen benachbarten Geräten zu bündeln, um so höhere Bandbreiten zu erreichen und zusätzlich für Ausfallsicherheit zu sorgen.

Hierbei werden Ports auf beiden Partnergeräten in Verbindungsbündel eingebunden und dann die Geräte über diese Ports miteinander verbunden. Um Ports (also Links) korrekt einem Partnergerät zuzuordnen, wird das Link-Aggregation Control Protocol (LACP) aus dem Standard IEEE 802.3ad verwendet.

Bis zu 8 Link Aggregationen können definiert werden. Jeder Link Aggregation können max. 8 Ports zugewiesen werden.

Hinweis

Wenn ein Port einer Link-Aggregation zugewiesen ist, aber nicht aktiv ist (z. B. Link-Down), können die angezeigten Werte von den Werten abweichen, die für die Link-Aggregation konfiguriert wurden. Wenn der Port in der Link-Aggregation aktiv wird, werden individuelle Portkonfigurationen wie z. B. DCP-Weiterleitung mit den konfigurierten Werte der Link Aggregation überschrieben.

Beschreibung der angezeigten Felder

Auf dieser Seite werden alle konfigurierten Link Aggregationen angezeigt. Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die virtuelle Port-Nummer dieser Link Aggregation an. Die Bezeichnung wird intern von der Firmware vergeben.
- **Name der Link Aggregation**
  Zeigt den Namen des Verbindungsbündels an. Dieser Name kann bei der Konfiguration vom Benutzer angegeben werden. Der Name ist nicht zwingend notwendig, kann aber hilfreich sein, die verschiedenen Link Aggregationen zu unterscheiden.
- **MAC-Adresse**
  Zeigt die MAC-Adresse an.
- **Status**
  Aktivieren oder deaktivieren Sie die Link Aggregation.
- **MTU**
  Legen Sie die Paketgröße fest.

- LACP
  - An
    Aktiviert das Senden von LACP-Telegrammen.
  - Aus
    Deaktiviert das Senden von LACP-Telegrammen.
- Lastverteilung
  Legen Sie die Art der Verteilung von Paketen auf die einzelnen Links einer Aggregation fest.
  - Destination&Source Mac
    Die Verteilung basiert auf einer Kombination der Ziel- und Quell-MAC-Adresse.
  - Destination&Source IP-MAC
    Die Verteilung basiert auf einer Kombination der Ziel- und Quell-IP- und MAC-Adresse.
- VLAN-Modus
  Legen Sie fest, wie die Link Aggregation in einem VLAN eingetragen wird:
  - Hybrid
    Die Link Aggregation sendet getaggte und ungetaggte Telegramme. Sie ist nicht automatisch Mitglied eines VLANs.
  - Trunk
    Die Link Aggregation sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.
- Liste der Ports
  Zeigt die Ports an, die zu dieser Link Aggregation gehören. Dabei können aus der Klappliste folgende Werte gewählt werden:
  - "-" (Deaktiviert)
    Link Aggregation ist deaktiviert.
  - "a" (Aktiv)
    Der Port sendet LACP-Telegramme und nimmt nur an der Link-Aggregation teil, wenn LACP-Telegramme empfangen werden.
  - "p" (Passiv)
    Der Port nimmt nur an der Link-Aggregation teil, wenn LACP-Telegramme empfangen werden.
  - "o" (Ein)
    Der Port nimmt an der Link-Aggregation teil und sendet keine LACP-Telegramme.

Hinweis

Innerhalb einer "Link Aggregation" sind immer nur Ports folgender Konfiguration möglich:

- alle Ports mit "o"
- alle Ports mit "a" oder "p".

Hinweis

Wenn Sie einen konfigurierten Port zu einer "Link Aggregation" hinzufügen, übernimmt der Port die Konfiguration der "Link Aggregation". Wenn Sie den Port aus der "Link Aggregation" herausnehmen, werden die Einstellungen des Ports auf die Werkseinstellung zurückgesetzt.

Siehe auch

Allgemein (Seite 2305)

LACP Timeout

Konfiguration des LACP Timeouts

Im Standard IEEE 802.3ad sind für die Länge des Timeouts zwei mögliche Werte festgelegt, "Long" (90 Sekunden) und "Short" (3 Sekunden). Dieser Wert definiert, in welchem zeitlichen Abstand LACPDUs gesendet werden. Als Defaultwert ist für alle Ports der Wert "Long" konfiguriert. Um eine symetrische LACP-Konfiguration zu ermöglichen, kann wahlweise der Wert "Short" eingestellt werden. Wählen Sie für alle Ports einer Link Aggregation den gleichen Wert für den Timeout.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**
  Wählen Sie aus der Klappliste die Einstellung. Folgende Einstellungsmöglichkeiten haben Sie:
  - Short
    Der Wert für den LACP Timeout beträgt 3 Sekunden.
  - Long
    Der Wert für den LACP Timeout beträgt 90 Sekunden.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Wählen Sie den Wert "Short" oder "Long" für diesen Port.

DCP-Weiterleitung

Anwendungen

Das DCP-Protokoll wird von STEP 7 und dem PST-Tool für die Konfiguration und Diagnose verwendet. In der Werkseinstellung ist DCP auf allen Ports aktiviert, d.h. empfangene DCP-Telegramme werden auf allen Ports weitergeleitet. Mit dieser Option haben Sie die Möglichkeit das Aussenden der Telegramme für einzelne Ports auszuschalten, um z.B. einzelne Netzbereiche von der Konfiguration per PST-Tool abzuschotten, bzw. um das gesamte Netz in kleinere Teilnetze für die Konfiguration und Diagnose zu unterteilen.

Hinweis

PNIO Konfiguration

Da es sich bei DCP um ein PROFINET-Protokoll handelt ist die hier vorgenommene Konfiguration nur in dem mit der TIA-Schnittstelle assoziierten VLAN wirksam.

Auf dieser Seite werden alle Ports des Gerätes angezeigt. Hinter jedem angezeigten Port befindet sich eine Klappliste zur Funktionsauswahl.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Legen Sie fest, ob der Port DCP-Telegramme ausgangsseitig blocken oder weiterleiten soll. Sie haben die folgenden Möglichkeiten zur Auswahl:
  - Forward
    An diesem Port werden DCP-Telegramme weitergeleitet.
  - Block
    An diesem Port werden ausgangsseitig keine DCP-Telegramme weitergeleitet. Ein Empfangen ist jedoch über diesen Port weiterhin möglich.

LLDP

Bestimmung der Netzwerktopologie

LLDP (Link Layer Discovery Protocol) ist im Standard IEEE 802.AB definiert.

LLDP ist ein Verfahren zur Bestimmung der Netzwerktopologie. Netzwerkkomponenten tauschen über LLDP Informationen mit ihren Nachbargeräten aus.

Netzwerkkomponenten, die LLDP unterstützen, verfügen über einen LLDP-Agenten. Der LLDP-Agent versendet in periodischen Abständen Informationen über sich selbst und empfängt Informationen von angeschlossenen Geräten. Die empfangenen Informationen werden

Anwendungen

PROFINET benutzt das LLDP-Protokoll für die Topologie-Diagnose. In der Werkseinstellung ist LLDP für alle Ports aktiviert, d. h. es werden LLDP-Telegramme auf allen Ports gesendet und empfangen. Mit dieser Funktion haben Sie die Möglichkeit das Aussenden und/oder Empfangen pro Port ein- oder auszuschalten.

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Legen Sie fest, ob der Port LLDP-Telegramme senden oder empfangen soll. Folgende Möglichkeiten gibt es:
  - Rx
    Dieser Port kann LLDP-Telegramme nur empfangen.
  - Tx
    Dieser Port kann LLDP-Telegramme nur senden.
  - Rx & Tx
    Dieser Port kann LLDP-Telegramme empfangen und senden.
  - "-" (Deaktiviert)
    Dieser Port kann LLDP-Telegramme weder empfangen noch senden.

FMP

Voraussetzungen

- Sie können Fiber Monitoring nur bei diagnosefähigen Transceivern verwenden. Beachten Sie hierzu die Dokumentation der Geräte.
- Um die Funktion Fiber Monitoring nutzen zu können, aktivieren Sie LLDP. Die Fiber Monitoring-Informationen werden an die LLDP-Pakete angehängt.

Überwachung optischer Strecken

Mit Fiber Monitoring können Sie die Empfangsleistung und den Leistungsabfall auf optischen Strecken zwischen zwei Switches überwachen.

Wenn Sie FM an einem optischen Port aktivieren, werden Diagnoseinformationen über den Port versendet. Parallel zum Sendevorgang wird geprüft, ob ensprechende Informationen empfangen werden.

Unabhängig davon, ob der IE-Switch Diagnoseinformationen von seinem Verbindungspartner empfängt, überwacht er die am optischen Port gemessene Empfangsleistung auf die eingestellten Grenzwerte.

Wenn bei dem Verbindungspartner Fiber Monitoring aktiviert ist, übermittelt der Verbindungspartner den aktuellen Wert für die Sendeleistung des Ports an das Gerät. Das Gerät vergleicht den Wert, den es für die Sendeleistung erhalten hat, mit der tatsächlich empfangenen Leistung. Aus der Differenz von Empfangsleistung und Sendeleistung ergibt sich der Leistungsabfall auf der Strecke. Der berechnete Leistungsabfall wird ebenfalls auf die eingestellten Grenzwerte überwacht.

Wenn der Wert der Empfangsleistung bzw. der Betrag des Leistungsabfalls die eingestellten Grenzwerte unter- bzw. überschreitet, wird ein Event ausgelöst. Sie können in zwei Stufen Grenzwerte einstellen, für Meldungen mit den Severity Levels "Warning" und "Critical".

Unter "System > Ereignisse > Konfiguration" können Sie einstellen, wie der IE-Switch das Event anzeigt.

Hinweis

Wenn Sie Fiber Monitoring aktiviert haben und ein diagnosefähiger Stecktransceiver gezogen wird, wird automatisch Fiber Monitoring für diesen Port deaktiviert und die eingestellten Grenzwerte sowie ein möglicher anstehender Fehlerzustand werden gelöscht.

Beschreibung der angezeigten Felder

In der Tabelle können Sie die zu überwachenden Grenzwerte für die gemessene Empfangsleistung und den berechneten Leistungsabfall festlegen.

- **Port**
  Zeigt die verfügbaren optischen Ports an, die Fiber Monitoring unterstützen. Dies ist von den Transceivern abhängig..
- **Status**
  Aktivieren oder Deaktivieren Sie FM. Defaultmäßig ist die Funktion aktiviert.

- Rx-Leistung [dBm] Wartungsbedarf (Warning)
  Tragen Sie den Wert ein, bei dem Sie durch eine Meldung des Severity-Levels "Warning" über die Verschlechterung der Empfangsleistung informiert werden. Wenn Sie den Wert "0" eintragen, wird die Empfangsleistung nicht überwacht. Der Default-Wert ist von dem jeweiligen Transceiver abhängig. Der Default-Wert ist von dem jeweiligen Stecktransceiver abhängig.
- Rx-Leistung [dBm] Wartungsanforderung (Critical)
  Tragen Sie den Wert ein, bei dem Sie durch eine Meldung des Severity-Levels "Critical" über die Verschlechterung der Empfangsleistung informiert werden. Wenn Sie den Wert "0" eintragen, wird die Empfangsleistung nicht überwacht. Der Default-Wert ist von dem jeweiligen Transceiver abhängig.
  Der Default-Wert ist von dem jeweiligen Stecktransceiver abhängig
- Leistungsabfall [dB] Wartungsbedarf (Warning)
  Tragen Sie den Wert ein, bei dem Sie durch eine Meldung des Severity-Levels "Warning" über den Leistungsabfall der Verbindung informiert werden. Wenn Sie den Wert "0" eintragen, wird der Leistungsabfall nicht überwacht. Default-Wert: -50 dB
- Leistungsabfall [dB] Wartungsanforderung (Critical)
  Tragen Sie den Wert ein, bei dem Sie durch eine Meldung des Severity-Levels "Critical" über den Leistungsabfall der Verbindung informiert werden. Wenn Sie den Wert "0" eintragen, wird der Leistungsabfall nicht überwacht. Default-Wert: -55 dB

Unicast

Filtering

Adressfilterung

Diese Tabelle zeigt die Unicast-Adressen, die statisch durch Parametrierung des Anwenders eingetragen wurden. Auf dieser Seite definieren Sie auch die statischen Unicast-Filter.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- VLAN-ID
  Wählen Sie die VLAN-ID aus, in dem Sie eine neue MAC-Adresse statisch konfigurieren. Wenn nichts vorgegeben wird, ist "VLAN1" als Grundeinstellung parametriert.
- MAC-Adresse
  Tragen Sie hier die MAC-Adresse ein.

Die Tabelle enthält folgende Spalten:

- VLAN-ID
  Zeigt die VLAN-ID, die dieser MAC-Adresse zugeordnet ist.
- MAC-Adresse
  Zeigt die MAC-Adresse des Teilnehmers, die das Gerät gelernt hat oder die der Anwender projektiert hat.

- **Status**
  Zeigt den Status jedes Adresseintrags:
  - Static
    Vom Anwender projektiert. Statische Adressen sind permanent gespeichert, d.h. sie werden nach Ablauf der Aging Time oder beim Neustart des Switches nicht gelöscht.
  - Invalid
    Diese Werte werden nicht ausgewertet.
- **Port**
  Zeigt an, über welchen Port der Teilnehmer mit der angegebenen Adresse erreichbar ist. Vom Gerät empfangene Telegramme, deren Zieladresse mit dieser Adresse übereinstimmt, werden an diesen Port weitergegeben.

Hinweis

Für Unicast-Adressen können Sie nur **einen** Port angeben.

Gesperrte Ports

Aktivierung der Zugangskontrolle

Auf dieser Seite können Sie einzelne Ports für unbekannte Teilnehmer sperren.

Wenn die Port Lock-Funktion aktiviert ist, werden Pakete an diesem Port, die von unbekannten MAC-Adressen kommen, sofort verworfen. Die Pakete von bekannten Teilnehmern werden vom Port angenommen.
Da Ports mit aktivierter Port Lock-Funktion auch keine MAC-Adressen lernen, werden gelernte Adressen auf diesen Ports nach Aktivieren der Port Lock-Funktion automatisch ausgetragen. Der Port akzeptiert nur statische MAC-Adressen, die vorher entweder manuell oder mit der "Lernprozess starten"-Funktion und der "Lernprozess stoppen"-Funktion erstellt wurden.

Um alle angeschlossenen Teilnehmer automatisch statisch einzutragen, gibt es eine Funktion zum automatischen Lernen (siehe "Layer 2 > Unicast > Learning").

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie aus der Klappliste die Einstellung. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Aktiviert die Port Lock-Funktion.
  - Deaktiviert
    Deaktiviert die Port Lock-Funktion.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  In dieser Spalte werden alle in diesem Gerät verfügbaren Ports aufgeführt.
- **Einstellung**
  Aktivieren oder deaktivieren Sie die Zugriffssteuerung für den Port.

Learning

Lernen starten/stoppen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mithilfe des Automatischen Lernens können alle angeschlossenen Geräte automatisch in die Unicast Filter Tabelle eingetragen werden. Solange die "Lernen starten"-Funktion aktiviert ist, werden alle gelernten Unicast-Adressen sofort als statische Unicast-Einträge angelegt.
Der Lernvorgang wird erst wieder durch Klicken auf die Schaltfläche "Lernen stoppen" beendet. Auf diese Weise kann wenige Minuten oder in größeren Netzen auch mehrere Stunden lang gelernt werden, um wirklich alle Teilnehmer zu finden. Es können nur Teilnehmer gefunden werden, die während des Lernens Pakete senden.
Durch anschließendes Aktivieren der Port Lock-Funktion werden auf den entsprechenden

Ports nur noch Pakete von den nach Beendigung des Lernvorgangs bekannten Teilnehmern (statische Unicast-Einträge) angenommen.

Hinweis

Ist die Port Lock-Funktion auf einzelnen Ports bereits vor dem automatischen Lernen aktiv, werden auf diesen Ports keine Adressen gelernt. Auf diese Weise ist es möglich nur auf bestimmten Ports zu lernen. Aktivieren Sie hierzu vorher die Port Lock-Funktion auf den Ports, die keine Adressen lernen sollen.

Vorgehensweise zur Konfiguration

Adressen lernen

1. Klicken Sie auf die Schaltfläche "Lernprozess starten", um den Lernvorgang zu starten. Nach dem Starten des Lernvorgangs wird die Schaltfläche "Lernprozess starten" durch die Schaltfläche "Lernprozess stoppen" ersetzt. Das Gerät trägt nun solange die Adressen angeschlossener Geräte ein, bis Sie den Vorgang anhalten.
2. Klicken Sie auf die Schaltfläche "Lernprozess stoppen", um den Lernvorgang anzuhalten. Die Schaltfläche wird wieder durch die Schaltfläche "Lernprozess starten" ersetzt. Die gelernten Einträge werden gespeichert.

Alle statischen Unicast-Adressen löschen

1. Klicken Sie auf die Schaltfläche "Alle statischen Unicast-Adressen löschen", um alle statischen Einträge zu löschen.
   In großen Netzen mit sehr vielen Teilnehmern kann das automatische Lernen eventuell zu vielen unerwünschten statischen Einträgen führen. Um diese nicht einzeln löschen zu müssen, gibt es über diese Schaltfläche die Möglichkeit, alle statischen Einträge zu löschen. Diese Funktion ist während des automatischen Lernens deaktiviert.

   Hinweis

   Das Löschen kann je nach Anzahl der Einträge einige Zeit in Anspruch nehmen.

Blocking

Weiterleitung von unbekannten Unicast-Telegrammen sperren

Auf der Seite wird das Weiterleiten von unbekannten Unicast-Telegrammen für einzelne Ports gesperrt.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie aus der Klappliste die Einstellung. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Blocken von Unicast-Telegrammen ist aktiviert.
  - Deaktiviert
    Blocken von Unicast-Telegrammen ist deaktiviert.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Alle verfügbaren Ports werden in dieser Spalte aufgeführt. Nicht verfügbare Ports werden nicht angezeigt.
- **Einstellung**
  Aktivieren oder deaktivieren Sie das Sperren von Unicast-Telegrammen.

Multicast

Gruppen

Multicast-Anwendungen

In der Mehrzahl der Fälle wird ein Telegramm mit einer Unicast-Adresse an einen bestimmten Empfänger gesendet. Wenn eine Anwendung die gleichen Daten an mehrere Empfänger senden soll, kann das zu sendende Datenvolumen reduziert werden, indem die Daten über eine Multicast-Adresse an alle gesendet werden. Für manche Anwendungen gibt es feste Multicast-Adressen (NTP, IETF1-Audio, IETF1-Video usw.).

Reduzierung der Netzlast

Im Gegensatz zu Unicast-Telegrammen bewirken Multicast-Telegramme eine höhere Last für das Gerät. Denn generell werden Multicast-Telegramme an allen Ports versendet. Es gibt drei Möglichkeiten, die Last durch Multicast-Telegramme zu reduzieren:

- Statischer Eintrag der Adressen in die Multicast-Filtertabelle.
- Dynamischer Eintrag der Adressen durch Mithören von IGMP-Parametriertelegrammen (IGMP-Konfiguration).
- Aktive dynamische Vergabe von Adressen durch GMRP-Telegramme.

Alle genannten Verfahren haben zur Folge, dass Multicast-Telegramme nur an solche Ports versendet werden, für die eine entsprechende Adresse eingetragen ist.

Der Menüpunkt "Multicast " zeigt die aktuell in der Filtertabelle eingetragenen Multicast-Telegramme mit ihren Zielports, die der Anwender parametriert hat (statisch).

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- VLAN-ID
  Wenn Sie auf dieses Textfeld klicken, wird Ihnen eine Klappliste angeboten. Hier können Sie die VLAN ID einer neu zu projektierenden MAC-Adresse auswählen.
- MAC-Adresse
  Hier geben Sie eine neu zu projektierende MAC-Multicast-Adresse ein.

Die Tabelle gliedert sich in folgende Spalten:

- VLAN-ID
  Hier wird die VLAN ID des VLANs angezeigt, dem die MAC-Multicast-Adresse dieser Zeile zugeordnet ist.
- MAC-Adresse
  Zeigt MAC-Multicast-Adresse an, die das Gerät gelernt hat oder die der Anwender projektiert hat.
- Status
  Zeigt den Status jedes Adress-Eintrags. Die Adresse wurde vom Anwender statisch eingetragen. Statische Adressen sind permanent gespeichert, d.h. sie werden nicht nach Ablauf der Aging-Time oder beim Neustart des Geräts gelöscht. Sie müssen vom Anwender gelöscht werden.
- Liste der Ports
  Für jeden Steckplatz gibt es eine Spalte. Innerhalb einer Spalte wird für jeden Port die Zugehörigkeit zur Multicast-Gruppe angegeben. Folgende Auswahlmöglichkeiten gibt es bei der Klappliste:
  - M
    (Member) Über diesen Port werden Multicast-Telegramme gesendet.
  - F
    (Forbidden) Kein Mitglied der Multicast-Gruppe. Außerdem darf diese Adresse nicht dynamisch über GMRP oder IGMP/MLD gelernt werden.
  - -
    Kein Mitglied der Multicast-Gruppe. Über diesen Port werden keine Multicast-Telegramme mit der definierten Multicast-MAC-Adresse gesendet

IGMP

Funktion

Das Gerät unterstützt "IGMP Snooping" und die "IGMP Querier"-Funktion. Ist "IGMP Snooping" aktiviert, so werden IGMP-Telegramme (Internet Group Management Protocol) ausgewertet und mit diesen Informationen die Multicast-Filtertabelle aktualisiert. Ist zusätzlich "IGMP Querier" aktiviert, so versendet das Gerät auch IGMP-Anfragen, die bei IGMP-fähigen Teilnehmern Antworten auslösen.

IGMP Snooping Aging Time

Mit diesem Menü können Sie die Aging-Time für die IGMP-Konfiguration festlegen. Nach Ablauf dieser Zeit werden durch IGMP erzeugte Einträge aus der Adresstabelle gelöscht, wenn diese nicht durch ein neues IGMP-Telegramm aktualisiert werden.

Die Festlegung gilt für alle Ports, eine portspezifische Konfiguration ist nicht möglich.

IGMP Snooping Aging Time in Anhängigkeit des Queriers

SCALANCE XR500 als IGMP-Querier

Wenn ein SCALANCE XR500 als IGMP-Querier verwendet wird, beträgt das Query-Intervall 125 Sekunden. Stellen Sie bei der "IGMP Snooping Aging Time" mindestens 250 Sekunden ein.

Andere IGMP-Querier

Wenn ein anderer IGMP-Querier verwendet wird, sollte der Wert der "IGMP Snooping Aging Time" mindestens doppelt so groß sein wie das Query-Intervall.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- IGMP Snooping
  Aktivieren oder deaktivieren Sie IGMP-Snooping (Internet Group Management Protocol). Die Funktion aktiviert IGMP-Snooping auf allen Schnittstellen. Wenn die Funktion aktiviert ist, werden die Multicast-Adressen, die über IGMP-Snooping gelernt wurden, in die Multicast-Filtertabelle eingetragen.
- IGMP Snooping Aging Time
  Tragen Sie in dieses Feld den Wert für die Aging Time in Sekunden ein. Standardmäßig sind 300 Sekunden eingestellt. Wertebereich: 130 - 1225 Sekunden
- IGMP Querier
  Aktivieren oder deaktivieren Sie "IGMP Querier". Das Gerät sendet zyklisch IGMPAnfragen.
- IGMP Snooping Switch IP Address
  Nach dem Aktivieren des Optionskästchens "IGMP Querier" wird hier die IP-Adresse des Geräts angezeigt. Optional können Sie die IP-Adresse eines anderen Geräts eintragen, das die Funktion des IGMP Queriers übernehmen soll.

Die Tabelle gliedert sich in folgende Spalten:

- VLAN-ID
  Die VLAN-ID, für die IGMP Snooping oder IGMP Querier aktiviert werden sollen.
- IGMP Snooping
  Aktivieren Sie die Optionskästchen in dieser Spalte für solche VLANs, bei denen IGMP Snooping aktiviert sein soll. Die Festlegungen in dieser Spalte werden erst dann wirksam, wenn Sie IGMP Snooping für das Gerät aktivieren (erstes Optionskästchen auf dieser Seite).
- IGMP Querier
  Aktivieren Sie die Optionskästchen in dieser Spalte für solche VLANs, bei denen IGMP Querier aktiviert sein soll. Voraussetzung ist, dass für das entsprechende VLAN IGMP Snooping aktiviert ist.

Statische IGMPs

Funktion

Statische Multicast-Adressen werden vom Anwender in die Filtertabelle eingetragen und dort permanent gespeichert. Statische Einträge in der Filtertabelle werden, im Gegensatz zu dynamisch erzeugten Einträgen, nicht nach Ablauf der Aging-Time oder beim Neustart des Geräts gelöscht. Sie müssen vom Anwender gelöscht werden.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- VLAN-ID
  Wählen Sie aus dieser Klappliste die VLAN-ID der zu projektierenden Gruppe.
- Start-Port
  Wählen Sie aus dieser Klappliste den ersten Port eines Port-Bereichs aus, für den Sie eine Multicast-Adresse konfigurieren wollen.
- End-Port
  Hier wird zunächst die in der Klappliste "Start-Port" getroffene Auswahl angezeigt. Wählen Sie aus dieser Klappliste den letzten Port eines Port-Bereichs aus, für den Sie eine Multicast-Adresse konfigurieren wollen. Wenn Sie in dieser Klappliste keine Auswahl vornehmen, wird der Multicast-Adresse nur ein Port zugeordnet.
- MAC-Adresse
  Hier geben Sie die Multicast-IP-Adresse der Gruppe ein.

Die Tabelle gliedert sich in folgende Spalten:

- Selektieren
  Wählen Sie die Zeile, die Sie löschen wollen.
- VLAN-ID
  Hier wird die ID des VLANs angezeigt, dem die Multicast-Adresse dieser Zeile zugeordnet ist.

- **Adresse der Gruppe**
  Hier wird die Multicast-IP-Adresse der Gruppe angezeigt.
- **Liste der Ports**
  Für jeden Port gibt es eine Spalte mit einem Optionskästchen. Wenn das Optionskästchen aktiviert ist, gehört der Port zur entsprechenden Gruppe.

Vorgehensweise zur Konfiguration

Statische Multicast-Gruppe anlegen

Hinweis

Sie können keine statischen Multicast-Einträge anlegen, wenn GMRP aktiviert ist.

1. Wählen Sie aus der Klappliste "VLAN-ID" die gewünschte ID aus.
2. Legen Sie in den Klapplisten Start-Port und End-Port die Ports fest, die Mitglied der Multicast-Gruppe sein sollen.
3. Tragen Sie in das Eingabefeld "Adresse der Gruppe" die Multicast-IP-Adresse der Gruppe ein.
4. Klicken Sie die Schaltfläche "Erstellen". In der Tabelle wird ein neuer Eintrag erzeugt.
5. Sie können bestehende Gruppen verändern, indem Sie die Optionskästchen der entsprechenden Ports aktivieren bzw. deaktivieren.

GMRP

Aktivierung von GMRP

Durch Aktivieren der entsprechenden Optionen legen Sie individuell für jeden Port fest, ob GMRP angewendet wird. Wenn für einen Port "GMRP" deaktiviert ist, werden für ihn keine Registrierungen durchgeführt und er kann keine GMRP-Telegramme versenden.

Damit GMRP funktioniert, müssen Sie die Funktion global und auf den Ports aktivieren.

Beschreibung der angezeigten Felder

Die Seite enthält folgendes Feld:

- **GMRP**
  Aktivieren oder deaktivieren Sie die GMRP-Funktion.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Aktiviert das Versenden von GMRP-Telegrammen.
  - Deaktiviert
    Deaktiviert das Versenden von GMRP-Telegrammen.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  In dieser Spalte werden alle im Gerät verfügbaren Ports und auch die Link-Aggregationen angezeigt.
- **Einstellung**
  Mit Hilfe dieses Optionskästchens aktivieren oder deaktivieren Sie GMRP für den Port oder die Link-Aggregation.

Multicast Blocking

Auf der Seite wird das Weiterleiten von unbekannten Multicast-Telegrammen für einzelne Ports gesperrt.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Blocken von Multicast-Telegrammen ist aktiviert.
  - Deaktiviert
    Blocken von unbekannten Multicast-Telegrammen ist deaktiviert.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Port
  Alle verfügbaren Ports werden in dieser Spalte aufgeführt. Nicht verfügbare Ports werden nicht angezeigt.
- Einstellung
  Aktivieren oder deaktivieren Sie das Blockieren von Multicast-Telegrammen.

MLD

Mulitcast Listener Discovery

Geräte unterstützen "MLD Snooping" und die "MLD Querier"-Funktion. Ist "MLD Snooping" aktiviert, so werden MLD-Telegramme ausgewertet und mit diesen Informationen die Multicast-Filtertabelle aktualisiert. Ist zusätzlich "MLD Querier" aktiviert, so versenden IE-Switches auch MLD-Anfragen, die bei MLD-fähigen Teilnehmern Antworten auslösen.

Bei MLD Snooping werden die MLD-Telegramme nur an den vorgesehenen Multicast Listener weiter geleitet, anstatt alle Ports damit zu fluten.

Damit MLD Snooping funktioniert, müssen Sie die Funktion global und in den VLANs aktivieren.

MLD Snooping Aging Time

Auf dieser Seite können Sie die Aging-Time für die MLD-Konfiguration festlegen. Nach Ablauf dieser Zeit werden durch MLD erzeugte Einträge aus der Adresstabelle gelöscht, wenn diese nicht durch ein neues MLD-Telegramm aktualisiert werden.

Die Festlegung gilt für alle VLANS, eine VLAN-spezifische Konfiguration ist nicht möglich.

Beschreibung der angezeigten Felder

- MLD Snooping
  Aktivieren oder deaktivieren Sie MLD auf dem Gerät.
- MLD Snooping Aging Time
  Geben Sie die Zeitdauer an, nach der MLD erzeugte Einträge aus der Mulitcast-Filtertabelle gelöscht werden. Vorausgesetzt, diese werden nicht durch ein neues MLD-Paket aktualisiert.
  Die Festlegung gilt für alle VLANS, eine VLAN-spezifische Konfiguration ist nicht möglich

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle VLANs der Tabelle 2 gültig sind.
- **Snooping / Querier**
  - Aktiviert
    Die Funktion ist aktiviert.
  - Deaktiviert
    Die Funktion ist deaktiviert.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **VLAN**
  Zeigt das VLAN an, auf das sich die Einstellungen beziehen
- **Snooping**
  Aktivieren oder deaktivieren Sie MLD für das gewünschte VLAN. Wenn aktiviert, werden die MLD-Einträge in Mulitcast-Filtertabelle aufgenommen und MLD-Pakete weitergeleitet.
- **Querier**
  Aktivieren oder deaktivieren verschickt MLD-Anfragen.

Broadcast

Sperrung der Weiterleitung von Broadcast-Telegrammen

Auf dieser Seite kann das Weiterleiten von Broadcast-Telegrammen für einzelne Ports gesperrt werden.

Hinweis

Einige Kommunikationsprotokolle funktionieren nur mit Unterstützung von Broadcast. In diesen Fällen kann das Sperren zum Ausfall der Datenkommunikation führen. Sperren Sie Broadcast nur, wenn Sie sicher sind, dass Sie darauf verzichten können.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Einstellmöglichkeiten:
  Aktiviert
  Das Blocken von Broadcast-Telegrammen ist aktiviert.
  Deaktiviert
  Das Blocken von Broadcast-Telegrammen ist deaktiviert.
  Keine Änderung
  Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- **Einstellung**
  Aktivieren oder deaktivieren Sie das Sperren von Broadcast-Telegrammen.

PTP

Allgemein

Folgende Geräte unterstützen die Uhrzeitsynchronisation über PTP:

- SCALANCE XR528-6M
- SCALANCE XR552-12M

IEEE 1588 bei SCALANCE-Geräten

Die Norm IEEE 1588v2 definiert Mechanismen, mit denen eine hochpräzise Synchronisation der Uhrzeit von Geräten in einem Netzwerk erreicht wird. SCALANCE-Geräte mit einer entsprechenden Hardware-Ausstattung unterstützen die Zeitsynchronisation nach IEEE 1588v2.

Die Funktionalität ist bei diesen Geräten im Auslieferungszustand und nach einem "Auf Werkseinstellungen zurücksetzen" abgeschaltet. Um PTP nutzen zu können, müssen Sie diese Funktion aktivieren und jeden Port, der im Synchronisationspfad liegt, sowie Ports, die durch Redundanzmechanismen geblockt sind, konfigurieren. PTP ist auch nutzbar bei Redundanzmechanismen im Ring wie HRP, Standby-Kopplung von Ringen, MRP und RSTP.

Konfiguration

- **Betriebsart 1588**
  Es gibt die folgenden Einstellmöglichkeiten:
  - **aus**
    Das Gerät verarbeitet keine PTP-Nachrichten. PTP-Nachrichten werden aber nach den Regeln des Switchs weitergeleitet.
  - **transparent**
    Das Gerät übernimmt die Funktion einer Transparent Clock und leitet PTP-Nachrichten an andere Teilnehmer weiter, wobei es Eintragungen in das Korrekturfeld der PTP-Nachricht vornimmt.

TC Allgemein

Auf dieser Seite legen Sie die allgemeinen Einstellungen zu PTP fest.

1588 Transparent Clock Configuration

Delay Mechanismus

Legen Sie fest, mit welchem Delay-Mechanismus das Gerät arbeiten soll:

- End-to-End
  (Delay-Request-Response-Mechanismus wird verwendet)

  Hinweis

  Bei der End-to-End Synchronisation mit mehr als 2 Slaves können Ausreißer > 100 ns im Offset auftreten.

- Peer-to-Peer
  (Peer-Delay-Mechanismus wird verwendet)

Domainnummer

Tragen Sie hier die Domänen-Nummer für das Gerät ein. Das Gerät ignoriert PTP-Nachrichten mit einer abweichenden Domänen-Nummer. Ein SCALANCE-Gerät kann nur einer Synchronisations-Domäne zugeordnet sein.

TC-Port

Auf dieser Seite befinden sich die Port-Einstellungen zu PTP.

1588 Transparent Clock Port Parameters

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.

- **Transportmechanismus**
  Es gibt folgende Einstellmöglichkeiten:
  - Ethernet
  - UDP IPv4
  - Keine Änderung
    Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Die Portnummer. Bei modularen Geräten werden Steckplatznummer und Portnummer durch einen Punkt getrennt angezeigt. Auf SPF+-Ports wird PTP nicht unterstützt.
- **Einstellung**
  Der Port-Status. Folgende Angaben sind möglich:
  - Aktiviert
    Der Port beteiligt sich nicht am PTP.
  - Deaktiviert
    Der Port verarbeitet PTP-Nachrichten
- **Fehlerflag**
  Der Fehlerstatus im Bezug auf PTP.
  - wahr
    Es ist ein Fehler aufgetreten.
  - falsch
    An diesem Port sind keine Fehler aufgetreten.
- **Transportmechanismus**
  Wählen Sie aus, wie dieser Port den Datenverkehr von PTP-Nachrichten abwickeln soll. Sie können für die Ports eines Geräts unterschiedliche Einstellungen vornehmen, allerdings muss der entsprechende Kommunikationspartner den gewählten Transportmechanismus unterstützen. Es gibt folgende Einstellmöglichkeiten:
  - Ethernet
  - UDP IPv4

RMON

Statistik

Statistik

Auf dieser Seite können Sie festlegen, für welche Ports RMON-Statistiken angezeigt werden. Die RMON-Statistiken werden auf der Seite "Information > Ethernet-Statistiken" bei "Telegrammlängen", "Telegrammtyp" und "Telegrammfehler" angezeigt.

Einstellungen

- **RMON**
  Wenn aktiviert, ermöglicht Remote Monitoring (RMON), Diagnosedaten im Gerät zu sammeln, aufzubereiten und über SNMP von einer Netzwerkmanagement-Station, die ebenfalls RMON unterstützt, auszulesen. Diese Diagnosedaten, wie zum Beispiel portbezogene Lastverläufe, ermöglichen es, Probleme im Netzwerk frühzeitig zu erkennen und zu beseitigen.

  Hinweis

  Wenn Sie RMON deaktivieren, werden die Statistiken nicht gelöscht, sondern sie bleiben auf dem letzten Stand stehen

- **Port**
  Wählen Sie die Ports aus, für die Statistiken angezeigt werden sollen.

Die Tabelle gliedert sich in folgende Spalte:

- **Port**
  Zeigt die Ports an, für die Statistiken angezeigt werden.

History

Stichproben der Statistiken

Auf dieser Seite können Sie festlegen, ob für einen Port Stichproben der Statistiken abgespeichert werden sollen. Sie können festlegen, wie viele Einträge gespeichert werden sollen und in welchem Intervall Stichproben genommen werden sollen.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **Einträge**
  Tragen Sie die maximale Anzahl der Stichproben ein, die gleichzeitig gespeichert werden sollen. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **Intervall [s]**
  Tragen Sie ein Intervall ein, nach dem der aktuelle Stand der Statistik als Stichprobe gespeichert werden soll. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt den Port an, auf den sich die Einstellungen beziehen.
- **Einstellung**
  Aktivieren oder deaktivieren Sie die Aufzeichnung der History auf dem entsprechenden Port.
- **Einträge**
  Tragen Sie die maximale Anzahl der Stichproben ein, die gleichzeitig gespeicherten werden sollen. Die maximale Anzahl an Einträgen kann durch die Kapazität des Geräts beschränkt werden.
  Wertebereich: 1 - 65535
  Werkseinstellung: 24
- **Intervall [s]**
  Tragen Sie ein Intervall ein, nach dem der aktuelle Stand der Statistik als Stichprobe gespeichert werden soll.
  Wertebereich: 1 - 3600
  Werkseinstellung: 3600

Layer 3-Funktionen für IPv4 projektieren

Konfiguration

Die Seite enthält die Übersicht über die Layer 3-Funktionen für IPv4 des Geräts. Auf dieser Seite aktivieren oder deaktivieren Sie die gewünschten Layer 3- Funktion.

Die Funktionen "Routing", "VRRP", "RIP" und "OSPF" sind nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Routing** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  - Aktiviert
    IPv4-Routing wird aktiviert. Sie können die Funktion Routing nur aktivieren, wenn DHCP auf allen konfigurierten Schnittstellen deaktiviert ist.
  - Deaktiviert
    IPv4-Routing wird deaktiviert. Ist am Gerät IPv6-Routing aktiviert, wird diese ebenfalls deaktiviert
- **DHCP Relay Agent**
  Aktivieren oder deaktivieren Sie den DHCP Relay Agenten. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv4) > DHCP Relay Agent".
- **VRRP** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  Aktivieren oder deaktivieren Sie Routing über VRRP. Um VRRP zu verwenden, aktivieren Sie zuerst die Funktion Routing. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv4) > VRRP".

- **OSPF** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  Aktivieren oder deaktivieren Sie Routing über OSPF. Um OSPF zu verwenden, aktivieren Sie zuerst die Funktion Routing. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv4) > OSPF".
- **RIP** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  Aktivieren oder deaktivieren Sie Routing über RIP. Um RIP zu verwenden, aktivieren Sie zuerst die Funktion Routing. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv4)> RIP".
- **PIM-Routing** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  Aktivieren oder deaktivieren Sie PIM-Routing. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv4) > PIM".

Subnetze

Übersicht (IPv4)

Erstellen von Subnetzen

Diese Seite zeigt die Subnetze für die ausgewählte Schnittstelle. Wenn für eine Schnittstelle mehrere Subnetze verfügbar sind, ist der erste Eintrag dieser Schnittstelle vom Adressentyp "Primär". Alle weiteren Subnetze werden auf dieser Seite angelegt. Ein Subnetz bezieht sich immer auf eine Schnittstelle. Die Schnittstelle wird auf dem Register "Konfiguration" angelegt.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus, an die Sie ein weiteres Subnetz projektieren.
- **Loopback**
  Aktivieren oder deaktivieren Sie die Eigenschaft "Loopback" auf der IP-Schnittstelle. Wenn für eine IP-Schnittstelle die Eigenschaft "Loopback" aktiviert werden soll, darf das zugewiesene VLAN keine Member-Ports enthalten. Da die IP-Schnittstelle nicht mit physikalischen Ports verknüpft ist, ist der Betriebszustand der IP-Schnittstelle unabhängig vom Link-Status der physikalischen Ports. Die Loopback- Schnittstelle ist erreichbar, solange das Gerät mit einer entsprechenden Spannungsversorgung verbunden ist. Über die Loopback-Schnittstelle können Sie auf das Gerät zugreifen.

  Hinweis

  Eine als Loopback konfigurierte Schnittstelle muss in einem eigenen Subnetz mit einer mind. 30 Bit-Subnetzmaske liegen.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an.
- **TIA-Schnittstelle**
  Zeigt an, ob der Eintrag eine TIA-Schnittstelle ist.

- **Schnittstellenname**
  Zeigt den Namen der Schnittstelle.
- **MAC-Adresse**
  Zeigt die MAC-Adresse an.
- **IP-Adresse**
  Zeigt die IPv4-Adresse des Subnetzes an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske an.
- **Adresstyp**
  Zeigt den Adresstyp an. Folgende Werte sind möglich:
  - Primär
    Die erste IPv4-Adresse, die auf einer Schnittstelle konfiguriert wurde.
  - Sekundär
    Alle weiteren IPv4-Adressen, die auf einer Schnittstelle konfiguriert wurden.
- **Methode der IP-Adresszuweisung**
  Zeigt an, wie die IPv4-Adresse zugeordnet wird.
  - Statisch
    Die IP-Adresse ist statisch. Die IP-Einstellungen tragen Sie bei "IP-Adresse" und "Subnetzmaske" ein.
  - Automatisch (DHCP)
    Das Gerät bezieht eine dynamische IPv4-Adresse von einem DHCP-Server.

- **Status der Erkennung von Adresskollisionen**
  Wenn neue IPv4-Adressen im Netz aktiv werden, prüft diese Funktion, ob es zu Adresskollisionen kommen kann. Mit dieser Funktion können IPv4-Adressen erkannt werden, die doppelt vergeben werden sollen.
  Zeigt an, in welchem Status sich die Funktion "Erkennung von Adresskollisionen" befindet.

  Hinweis

  Die Funktion führt keine zyklische Prüfung durch.

  – Idle
    Die Schnittstelle ist nicht aktiv und besitzt keine IP-Adresse.
  – Starting
    Dieser Status bezeichnet die Anlaufphase. In dieser Phase sendet das Gerät zunächst eine Anfrage, ob es die geplante IP-Adresse bereits gibt. Wenn die Adresse noch nicht vergeben ist, sendet das Gerät die Mitteilung, dass es ab jetzt diese IP-Adresse verwendet.
  – Conflict
    Die Schnittstelle ist nicht aktiv. Die Schnittstelle versucht eine IP-Adresse zu verwenden, die bereits vergeben ist.
  – Defending
    Die Schnittstelle verwendet eine eindeutige IP-Adresse. Eine andere Schnittstelle versucht die gleiche IP-Adresse zu verwenden.
  – Active
    Die Schnittstelle verwendet eine eindeutige IP-Adresse. Es gibt keine Kollisionen.
  – Not supported
    Die Funktion zur Erkennung von Adresskollisionen wird nicht unterstützt.
  – Disabled
    Die Funktion zur Erkennung von Adresskollisionen ist deaktiviert.

- **Loopback**
  Zeigt an, ob die Eigenschaft "Loopback" aktiviert ist.

Konfiguration

Auf dieser Seite konfigurieren Sie das Subnetz.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Schnittstelle (Name)**
  Wählen Sie die gewünschte Schnittstelle aus.
- **Schnittstellenname**
  Geben Sie einen Namen für die Schnittstelle ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse der ausgewählten Schnittstelle an

- DHCP
  Aktivieren oder deaktivieren Sie den DHCP-Client für diese IPv4-Schnittstelle.

  Hinweis

  Wenn Sie das Gerät als Router mit mehreren Schnittstellen betreiben wollen, deaktivieren Sie DHCP auf allen Schnittstellen.

- IP-Adresse
  Geben Sie die IPv4-Adresse des Subnetzes ein. Die IPv4-Adresse darf nicht mehrfach verwendet werden.
- Subnetzmaske
  Geben Sie die Subnetzmaske des zu erstellenden Subnetzes ein. Subnetze an unterschiedlichen Schnittstellen dürfen sich nicht überlappen.
- Adresstyp
  Zeigt den Adresstyp an. Folgende Werte sind möglich:
  - Primär
    Das erste Subnetz der Schnittstelle.
  - Sekundär
    Alle weiteren Subnetze der Schnittstelle.
- Loopback
  Zeigt an, ob die Eigenschaft "Loopback" aktiviert ist.

- IP-Quelle
  Wenn die Option aktiviert ist, wird in den folgenden Protokollen die Loopback-Schnittstelle als Quell-IP-Adresse verwendet:
  - RADIUS
  - SNMPv1 Trap
  - Syslog
  - SNTP
  - NTP
  - SMTP
  - TFTP
  - SFTP

  Diese Option kann z. B. beim Einsatz einer Firewall oder der Authentifizierung über RADIUS hilfreich sein. Da ein Router mit mehreren IP-Schnittstellen nicht immer die gleiche IP-Schnittstelle als Quelle angibt, müssten ohne diese Einstellung auf einem RADIUS-Server oder bei einer Firewall alle IP-Schnittstellen eines Routers konfiguriert werden. Mit dieser Einstellung ist es möglich, dass eine Router für die ausgehende IP-Kommunikation eine Loopback-Schnittstelle verwendet und damit nur diese dem RADIUS-Server und der Firewall bekannt sein muss.
  Wenn die Option deaktiviert ist, wird in den Protokollen die ausgehende Schittstelle als Quelle angegeben.
- TIA-Schnittstelle
  Aktivieren oder deaktivieren Sie die Einstellung.
  Für die TIA-Schnittstelle gelten folgende Bedingungen:
  - Nur Schnittstellen mit dem Adresstyp "Primär" können als TIA-Schnittstelle aktiviert werden.
  - Es muss immer eine TIA-Schnittstelle geben.
  - Es kann nur eine TIA-Schnittstelle geben.
  - Eine TIA-Schnittstelle ist immer eine VLAN-Schnittstelle.

TIA-Schnittstelle und PROFINET-Schnittstelle

Die IP-Adresse der TIA-Schnittstelle ist mit der IP-Adresse verknüpft, die Sie in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen" finden.

Beide IP-Adressen haben immer den gleichen Wert. Wenn Sie den Wert einer IP-Adresse ändern, ändert sich die andere IP-Adresse entsprechend.

TIA-Schnittstelle

Bei der TIA-Schnittstelle handelt es sich um die Schnittstelle, über die alle PROFINET-Funktionen des IE-Switch laufen.

TIA-Schnittstelle ist die Bezeichnung der IP-Adresse für PROFINET aus der Sicht des IE-Switches.

PROFINET-Schnittstelle

Unter PROFINET-Schnittstelle nehmen Sie die IP-Einstellung des PROFINET-Geräts in TIA vor.

PROFINET-Schnittstelle ist die Bezeichnung der IP-Adresse für PROFINET aus der Sicht des TIA-Portals.

Default-Gateway

Erstellen von Subnetzen

Auf dieser Seite definieren Sie das Default-Gateway.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- Default-Gateway
  Geben Sie die IP-Adresse der Schnittstelle ein, die als Default-Gateway verwendet wird.

NAT

NAT

Auf dieser WBM-Seite legen Sie die Grundeinstellungen für NAT fest.

Beschreibung

Die Seite enthält folgende Felder:

- NAT
  Aktivieren oder deaktivieren Sie NAT/NAPT für das gesamte Gerät. Wenn aktiviert, fungiert das Gerät als NAT-Router.
- Idle Timeout [s]
  Geben Sie die gewünschte Zeitspanne ein. Das Gerät prüft zyklisch, nach Ablauf der angegebenen Zeitspanne, ob die Aging Time von TCP- und UDP-Verbindungen abgelaufen ist. Die Verbindungen, deren Aging Time seit der letzten Prüfung abgelaufen ist, werden aus der Tabelle "NAT-Übersetzungen" gelöscht.
- TCP Timeout [s]
  Geben Sie die gewünschte Aging Time für TCP-Verbindungen ein. TCP-Verbindungen werden solange gespeichert, bis für die angegebene Zeitspanne kein Datenaustausch stattgefunden hat. Abhängig von der zyklischen Prüfung nach Ablauf des Idle Timeouts werden die Verbindungen aus der Tabelle "NAT-Übersetzungen" gelöscht.
- UDP Timeout [s]
  Geben Sie die gewünschte Aging Time für UDP-Verbindungen ein. UDP-Verbindungen werden solange gespeichert, bis für die angegebene Zeitspanne kein Datenaustausch stattgefunden hat. Abhängig von der zyklischen Prüfung nach Ablauf des Idle Timeouts werden die Verbindungen aus der Tabelle "NAT-Übersetzungen" gelöscht.

- **Schnittstelle**
  Wählen Sie aus der Klappliste eine IP-Schnittstelle aus, auf der Sie NAT konfigurieren wollen. Sobald Sie eine Schnittstelle als NAT-Schnittstelle konfiguriert haben, werden alle weiteren Konfigurationen ausgehend von dieser Schnittstelle betrachtet. Das bedeutet, für diese Schnittstelle sind alle über die Schnittstelle selbst erreichbaren Netzwerke "Outside". Alle anderen Netzwerke sind "Inside".

  Hinweis

  Wenn Sie mehrere NAT-Schnittstellen auf einem Gerät konfiguriert haben, ist dadurch ein Netzwerk aus Sicht einer NAT-Schnittstelle "Outside" und aus der Sicht einer anderen NATSchnittstelle "Inside".

- **NAT**
  Aktivieren oder deaktivieren Sie NAT für eine IP-Schnittstelle.
  Im Register "Pool" wird automatisch ein Eintrag erstellt. Das Gerät ist aus dem externen Netz über die IP-Adresse der IP-Schnittstelle erreichbar.
  Wenn Sie NAT für eine IP-Schnittstelle deaktivieren und keine Konfigurationen auf der NAT-Schnittstelle vorliegen, wird der Eintrag automatisch aus der Tabelle gelöscht.
  NAPT
  Aktivieren oder deaktivieren Sie NAPT für eine IP-Schnittstelle.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.
- **NAT**
  Zeigt an, ob NAT für die ausgewählte IP-Schnittstelle aktiviert oder deaktiviert ist. NAT ist erst aktiv, wenn Sie NAT für das gesamte Gerät aktiviert haben.
- **NAPT**
  Zeigt an, ob NAPT für die ausgewählte IP-Schnittstelle aktiviert oder deaktiviert ist. NAPT ist erst aktiv, wenn Sie NAT für das gesamte Gerät aktiviert haben. Wenn Sie keine weiteren Konfigurationen für NAPT vornehmen, ist automatisch die dynamische Portumsetzung aktiv. Ein Gerät im internen Netz ist standardmäßig nicht aus einem externen Netz erreichbar. Wenn das interne Gerät in ein externes Netz kommunizieren will, werden die Inside Local- Adresse und die IP-Adresse der IP-Schnittstelle um einen Port ergänzt und dem internen Gerät als Inside Local- und Inside Global-Adresse zugeordnet. Über diese Inside Global- Adresse ist das interne Gerät aus dem externen Netz erreichbar, bis der Timer der Verbindung abgelaufen ist.

Statisch

Auf dieser WBM-Seite konfigurieren Sie statische 1:1-Adressumsetzungen.

Sie legen fest, in welche Inside Global-Adresse die Inside Local-Adresse eines Geräts umgesetzt werden soll und umgekehrt. Diese Variante erlaubt den Verbindungsaufbau in beide Richtungen. Das Gerät im internen Netz ist aus dem externen Netz erreichbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste eine NAT-Schnittstelle aus, für die Sie weitere NATKonfigurationen vornehmen wollen.
- **Inside Local-Adresse**
  Geben Sie die tatsächliche Adresse des Geräts ein, das von extern erreichbar sein soll.
- **Inside Global-Adresse**
  Geben Sie die Adresse ein, unter der das Gerät von extern erreichbar sein soll.

Die Tabelle gliedert sich in folgende Spalten:

- **1. Spalte**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Schnittstelle**
  NAT-Schnittstelle, auf die sich die Einstellung bezieht.
- **Inside Local-Adresse**
  Zeigt die tatsächliche Adresse des Geräts an, das von extern erreichbar sein soll.
- **Inside Global-Adresse**
  Zeigt die Adresse an, unter der das Gerät von extern erreichbar sein soll.

Pool

Auf dieser WBM-Seite konfigurieren Sie dynamische Adressumsetzungen.

Das Gerät ist standardmäßig nicht aus einem externen Netz erreichbar. Wenn das Gerät in ein externes Netz kommunizieren will, wird ihm dynamisch eine Inside Global-Adresse zugeordnet. Über diese Inside Global-Adresse ist das Gerät aus dem externen Netz erreichbar, bis der Timer der Verbindung abgelaufen ist.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste eine NAT-Schnittstelle aus, für die Sie weitere NATKonfigurationen vornehmen wollen.
- **Inside Global-Adresse**
  Geben Sie die Startadresse für die dynamische Zuordnung von Adressen ein, unter denen Geräte von extern erreichbar sein sollen.
- **Inside Global-Adressmaske**
  Geben Sie die Adressmaske des externen Subnetzes ein.

Die Tabelle gliedert sich in folgende Spalten:

- **1. Spalte**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.

- **Inside Global-Adresse**
  Zeigt die Startadresse für die dynamische Zuordnung von Adressen an, unter denen Geräte von extern erreichbar sein sollen.
- **Inside Global-Adressmaske**
  Zeigt die Adressmaske des externen Subnetzes an.

NAPT

Auf dieser WBM-Seite konfigurieren Sie statische Portumsetzungen.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste eine NAT-Schnittstelle aus, für die Sie weitere NATKonfigurationen vornehmen wollen.
- **Inside Local-Adresse**
  Geben Sie die tatsächliche Adresse des Geräts ein, das von extern erreichbar sein soll.
- **Dienst**
  Wählen Sie aus, für welchen Dienst die Portumsetzung gültig ist.
  Wenn Sie einen Dienst auswählen, werden in den Feldern Start-Port und End-Port der gleiche Port eingetragen. Wenn Sie den Start-Port ändern, wird der End-Port entsprechend geändert.
  Wenn Sie den Eintrag "-" auswählen, können Sie den Start- und End-Port frei eingeben.
- **Start-Port**
  Geben Sie einen Port ein.
- **End-Port**
  Abhängig von Ihrer Auswahl in der Klappliste "Dienst", können Sie einen Port eingeben oder es wird ein Port angezeigt.
  Wenn Sie in den Feldern Start-Port und End-Port unterschiedliche Ports eingeben, wird im Feld Inside Global-Port der gleiche Port-Bereich eingetragen. Ein Port-Bereich kann immer nur auf den gleichen Port-Bereich umgesetzt werden.
  Wenn Sie in den Feldern Start-Port und End-Port den gleichen Port eingeben, können Sie den Inside Global-Port frei eingeben.
- **Inside Global-Port**
  Abhängig von Ihrer Auswahl in der Klappliste "Dienst", können Sie einen Port eingeben oder es wird ein Port angezeigt.
- **Protokoll**
  Wählen Sie aus, für welches Protokoll die Portumsetzung gültig ist.
- **Beschreibung**
  Geben Sie eine Beschreibung für die Portumsetzung ein.

Die Tabelle gliedert sich in folgende Spalten:

- **1. Spalte**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Schnittstelle**
  NAT-Schnittstelle, auf die sich die Einstellung bezieht.
- **Inside Local-Adresse**
  Zeigt die tatsächliche Adresse des Geräts an, das von extern erreichbar sein soll.
- **Start-Port**
  Zeigt den Start-Port an, der der Inside Local-Adresse zugeordnet wird.
- **End-Port**
  Zeigt den End-Port an, der der Inside Local-Adresse zugeordnet wird.
- **Protokoll**
  Zeigt an, für welches Protokoll die Portumsetzung gültig ist.
- **Inside Global-Adresse**
  Zeigt die Adresse an, unter der das Gerät von extern erreichbar sein soll.
- **Inside Global-Port**
  Zeigt den Port an, der der Inside Global-Adresse zugeordnet wird.
- **Beschreibung**
  Zeigt die Beschreibung für die Portumsetzung an.

Statische Routen (IPv4)

Statische Route

Auf dieser Seite legen Sie statische IPv4-Routen an.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Zielnetzwerk**
  Tragen Sie die Netzwerkadresse des Ziels ein, das über diese Route erreichbar ist.
- **Subnetzmaske**
  Tragen Sie die dazugehörende Subnetzmaske ein.
- **Gateway**
  Tragen Sie die IPv4-Adresse des Gateways ein, über den diese Netzwerkadresse erreichbar ist.
- **Administrative Distanz**
  Tragen Sie die administrative Distanz für die Route ein. Die administrative Distanz entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt. Standardmäßig ist -1 eingestellt. Diese Einstellung bedeutet, dass die Metrik nicht gesetzt ist.
  Wertebereich: 1 - 254

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Netzwerkadresse des Ziels an.
- **Subnetzmaske**
  Zeigt die dazugehörende Subnetzmaske an.
- **Gateway**
  Zeigt die IPv4-Adresse des nächsten Gateways an.
- **Schnittstelle** (Nur Online verfügbar)
  Zeigt die Schnittstelle der Route an.
- **Administrative Distanz**
  Tragen Sie die administrative Distanz für die Route ein. Beim Erstellen der Route wird automatisch "Nicht Verwendet" eingetragen (Wert -1). Die administrative Distanz entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt. Wertebereich: 1 - 254
- **Status** (Nur Online verfügbar)
  Zeigt an, ob die Route aktiv ist oder nicht.

Route Maps

Allgemein (IPv4)

Route Maps

Mit Route Maps können Sie kontrollieren, wie Routing-Informationen weiterverarbeitet werden. Sie können Routing-Informationen filtern und festlegen, ob die Informationen weiterverarbeitet, verändert oder verworfen werden.

Route Maps arbeiten nach folgendem Prinzip:

- Routing-Informationen werden mit den Filtern der Route Maps abgeglichen.
- Der Abgleich läuft solange, bis die Filter einer Route Map mit den Eigenschaften einer Information übereinstimmen.
- Die Information wird dann entsprechend den Einstellungen der Route Map verarbeitet:
  - Die Routing-Information wird verworfen.
  - Die Eigenschaften der Routing-Information werden verändert.

Beschreibung der angezeigten Felder

- **Name**
  Tragen Sie einen Namen für die Route Map ein.
- **Sequenznummer**
  Tragen Sie eine Nummer für die Route Map ein.
  Sie können mehrere Route Maps mit gleichem Namen, aber unterschiedlichen Sequenznummern, anlegen. Die Sequenznummer bestimmt dann die Reihenfolge, in der die Route Maps abgearbeitet werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen der Route Map an.
- **Sequenznummer**
  Zeigt die Sequenznummer der Route Map an.
- **Aktion**
  Legen Sie fest, was mit den Routing-Informationen passiert, die den Einstellungen der Route Map entsprechen:
  - permit
    Die Routing-Informationen werden weiterverarbeitet, entsprechend Ihren Einstellungen auf der Seite "Einstellen".
  - deny
    Die Routing-Informationen werden verworfen.

Schnittstelle & Wert filtern (IPv4)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen für eine Route Map nach Schnittstellen, Metrik oder Tags gefiltert werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.
  Es stehen Ihnen die angelegten Route Maps zur Verfügung.
- **Typ**
  Wählen Sie aus, wonach gefiltert werden soll:
  - Schnittstelle
  - Metrik
  - Tag
  - Routen-Typ
  - Metrik-Typ
- **Schnittstelle**
  Wählen Sie eine Schnittstelle aus.
  Nur aktiv, wenn Sie bei "Typ" den Eintrag "Schnittstelle" ausgewählt haben.
- **Metrik**
  Geben Sie für die Metrik einen Wert ein.
  Dieses Feld ist nur aktiv, wenn Sie bei "Typ" den Eintrag "Metrik" ausgewählt haben.
- **Tag**
  Geben Sie für den Tag einen Wert ein.
  Dieses Feld ist nur aktiv, wenn Sie bei "Typ" den Eintrag "Tag" ausgewählt haben.

- **Routen-Typ**
  Wählen Sie den Typ der Route aus:
  - Lokal
    Die Routing-Informationen für die Route Map werden nach direkt verbundenen Routen (lokale Schnittstellen) gefiltert.
  - Remote
    Die Routing-Informationen für die Route Map werden nach gelernten oder statisch konfigurierten Routen gefiltert.

  Dieses Feld ist nur aktiv, wenn Sie in der Klappliste "Typ" den Eintrag "Routen-Typ" ausgewählt haben.
- Metrik-Typ
  Wählen Sie den Typ der Metrik aus:
  - Inter-Area
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, die aus anderen Areas gelernt werden.
  - Intra-Area
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, die aus der gleichen Area stammen.
  - type-1-external
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, deren Pfadkosten (Metrik) sich aus den externen Pfadkosten und den Pfadkosten zum ASBR zusammensetzen.
  - type-2-external
    Die Routing-Informationen für die Route Map werden nach Routen mit nur externen Pfadkosten (Metrik) gefiltert.

  Dieses Feld ist nur aktiv, wenn Sie in der Klappliste "Typ" den Eintrag "Metrik-Typ" ausgewählt haben.

Die Tabelle gliedert sich in folgende Spalten:

- **Typ**
  Zeigt den ausgewählten Typ an:
  - Schnittstelle
  - Metrik
  - Tag
  - Routen-Typ
  - Metrik-Typ
- **Wert**
  Zeigt die ausgewählte Schnittstelle, den Wert der Metrik oder des Tags an.

Ziel filtern (IPv4)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen für eine Route Map nach der Ziel-IPv4-Adresse gefiltert werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.
- **IP-Adresse**
  Geben Sie die IPv4-Adresse des Ziels ein, nach der gefiltert werden soll.
- **Subnetzmaske**
  Geben Sie die Subnetzmaske des Ziels ein, nach der gefiltert werden soll.

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv4-Adresse des Ziels an.
- **Subnetzmaske/Präfix**
  Zeigt die Subnetzmaske des Ziels an.

Next-Hop filtern (IPv4)

Auf dieser Seite legen Sie fest, ob für eine Route Map danach gefiltert werden soll, zu welchem Router die Routing-Informationen als nächstes gesendet werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.
- **IP-Adresse**
  Geben Sie die IP-Adresse des Routers ein, an den die Routing-Informationen als nächstes gesendet werden sollen.

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IP-Adresse des nächsten Routers an.

Einstellen (IPv4)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen durch eine Route Map verändert werden.

Wenn Sie z. B. nach einer bestimmten Metrik gefiltert haben, können Sie hier den Wert der Metrik verändern. Die Routing-Informationen werden dann mit dem neuen Wert weitergeleitet. Sie können nur die Informationen einer "Permit"-Route Map verändern.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen der Route Map an.
- **Sequenznummer**
  Zeigt die Sequenznummer der Route Map an.
- **Metrik**
  Geben Sie den neuen Wert für die Metrik ein, mit dem die Routing-Informationen weitergeleitet werden.
- **Tag**
  Geben Sie den neuen Wert für den Tag ein, mit dem die Routing-Informationen weitergeleitet werden.

DHCP Relay Agent

Allgemein

Wenn sich der DHCP-Server in einem anderen Netz befindet als der DHCP-Client, kann der Client den Server nicht erreichen. Der DHCP Relay Agent vermittelt zwischen DHCP-Server und DHCP-Client. Wenn Sie die Option 82 konfigurieren, erweitert der DHCP Relay Agent die Pakete an den DHCP-Server um eine Circuit-ID und eine Remote-ID.

Sie können für den DHCP Relay Agent bis zu 4 DHCP-Server angeben. Wenn ein DHCPServer nicht erreichbar ist, kann das Gerät auf einen anderen DHCP-Server ausweichen.

Beschreibung der angezeigten Felder

- **DHCP Relay Agent**
  Aktivieren oder deaktivieren Sie den DHCP Relay Agent.
- **Option 82 senden**
  Aktivieren oder deaktivieren Sie die Option 82.
- **Gemeinsame Agent-Adresse**
  Aktivieren oder deaktivieren Sie die gemeinsame Agent-Adresse. Wenn die Funktion aktiviert ist, ersetzt der Relay Agent in der DHCP-Anfrage die Adresse des Empfangsports durch die Adresse der Schnittstelle, die Sie unter "Gemeinsame Agent- Schnittstelle" konfigurieren.
- **Gemeinsame Agent-Schnittstelle**
  Der Relay Agent verwendet die IP-Adresse der hier ausgewählten Schnittstelle als Quelladresse (giaddr) in DHCP-Anfragen.
- **IP-Adresse des Servers**
  Tragen Sie die IPv4-Adresse des DHCP-Servers ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Server-IP-Adresse**
  Zeigt die IPv4-Adresse des DHCP-Servers an.

Option

Parameter des DHCP Relay Agent

Auf dieser Seite können Sie Parameter für den DHCP-Server festlegen, z. B. die Ciruit-ID. Die Circuit-ID beschreibt die Herkunft der DHCP-Anfrage, z. B. welcher Port die DHCP-Anfrage empfangen hat. Die DHCP-Server legen Sie auf dem Register "General" fest.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

Globale Konfiguration

- **Circuit-ID Router-Index**
  Aktivieren oder deaktivieren Sie die Einstellung. Wenn aktiviert, wird der erzeugten Circuit ID der Router Index hinzugefügt.
- **Circuit-ID Empfänger-VLAN-ID**
  Aktivieren oder deaktivieren Sie die Einstellung. Wenn aktiviert, wird der erzeugten Circuit ID die VLAN ID hinzugefügt.
- **Circuit-ID Empfänger-Port**
  Aktivieren oder deaktivieren Sie die Einstellung. Wenn aktiviert, wird der erzeugten Circuit ID der Empfangsport hinzugefügt.

  Hinweis

  Sie müssen mindestens eine Option auswählen.

  Weiterführende Informationen zum Router-Index (Circuit-ID Router-Index) und Port-Index (Circuit-ID Empfänger-Port) finden Sie in der IfTable über SNMP.

- **Remote-ID**
  Zeigt die Gerätekennung an.

Schnittstellen-spezifische Konfiguration

- **Schnittstelle**
  Wählen Sie die gewünschte IPv4-Schnittstelle aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an.

  Hinweis

  Wenn Sie keine Schnittstellen-spezifische Konfiguration angelegt haben, wird die globale Konfiguration mit der MAC-Adresse als Gerätekennung verwendet

- **Remote-ID-Typ**
  Wählen Sie die Art der Gerätekennung aus. Sie haben folgende Möglichkeiten:
  - IP-Adresse
    Als Gerätekennung wird die aktuelle IPv4-Adresse des Geräts verwendet.

    Hinweis

    Keine automatische Aktualisierung

    Es gibt keine Verknüpfung zwischen dem Feld "Remote-ID" und der aktuell eingestellten IP-Adresse.

    Wenn die IP-Adresse geändert wird, wird die neue IP-Adresse nicht automatisch in das Feld "Remote-ID" eingetragen. Es ändert sich nur der Wert des Feldes "Remote-ID-Typ" in "Free Text". Um wieder die aktuelle IP-Adresse zu verwenden, wählen Sie im Feld "Remote-ID-Typ" erneut "IP-Adresse" aus.

  - MAC-Adresse
    Als Gerätekennung wird die MAC-Adresse des Geräts verwendet.
  - Beliebiger Text
    Wenn Sie "Beliebiger Text" verwenden, können Sie bei "Remote-ID" den Gerätenamen als Gerätekennung eintragen.
- **Remote-ID**
  Tragen Sie den Gerätenamen ein. Nur editierbar, wenn Sie bei "Remote-ID-Typ" den Eintrag "Beliebiger Text" auswählen.
- **Circuit-ID-Type**
  Wählen Sie aus der Klappliste die Art der Circuit-ID aus. Sie haben folgende Möglichkeiten:
  - Vordefiniert
    Die Ciruit-ID wird automatisch erstellt, basierend auf Router Index, VLAN-ID oder Port.
  - Beliebige Nummer
    Wenn Sie "Beliebige Nummer" verwenden, können Sie bei "Circuit-ID" die ID eingeben.
- **Circuit-ID**
  Tragen Sie die Circuit-ID ein. Nur editierbar, wenn Sie bei "Circuit-ID-Type" den Eintrag "Beliebige Nummer" auswählen.

VRRP

Router

Mit Hilfe der Schaltfläche "Erstellen" können Sie neue virtuelle Router anlegen. Maximal sind 52 virtuelle Router konfigurierbar. Weitere Parameter konfigurieren Sie auf dem Register "Konfiguration".

Hinweis

- VRRP ist nur bei Layer 3 verfügbar.
- Aktivieren Sie "VRRP", um VRRP zu konfigurieren
- Der gleichzeitige Betrieb von VRRP und VRRPv3 ist nicht möglich
- VRRP ist nur im Zusammenhang mit VLAN-Schnittstellen nutzbar. Router-Ports werden nicht unterstützt.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **VRRP**
  Aktivieren oder deaktivieren Sie das Routing über VRRP.
- **Antworte an virtuellen Schnittstellen auf Ping-Anfragen**
  Wenn aktiviert, antworten auch die virtuellen IPv4-Adressen auf den Ping.
- **VRID-Tracking**
  Aktivieren oder deaktivieren Sie das VRID-Tracking. Wenn aktiviert, werden alle VRRP-Instanzen überwacht. Wenn sich der Status einer VRRPInstanz nach "Initialize" ändert, wird die Priorität aller VRRP-Instanzen auf den Wert "1" verringert. Wenn sich der Status der VRRP-Instanz ändert, wird die unsprüngliche Priorität aller VRRPInstanzen wiederhergestellt.
- **Schnittstelle**
  Wählen Sie die gewünschte VLAN-Schnittstelle aus, die als virtueller Router fungiert
- **VRID**
  Tragen Sie in das Eingabefeld die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden. Gültige Werte sind 1 ... 255.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, die als virtueller Router fungiert.
- **VRID**
  Zeigt die ID des virtuellen Routers an.
- **Virtuelle MAC-Adresse**
  Zeigt die virtuelle MAC-Adresse des virtuellen Routers an.

- **Primäre IP-Adresse**
  Zeigt die kleinste numerische IPv4-Adresse auf diesem VLAN an. Der Eintrag 0.0.0.0 bedeutet, dass die "Primäre" Adresse auf diesem VLAN verwendet. Ansonsten sind alle IPv4-Adressen, die auf diesem VLAN unter "Layer 3 (IPv4) > Subnetz" konfiguriert wurden, gültige Werte.
- **Router-Status** (Nur Online verfügbar)
  Zeigt den derzeitigen Zustand des virtuellen Routers an. Mögliche Werte sind:
  - Master
    Der Router ist der Master-Router und übernimmt die Routingfunktionalität für alle zugeordneten IP-Adressen.
  - Backup
    Der Router ist der Backup-Router. Wenn der Master-Router ausfällt, übernimmt der Backup-Router die Aufgaben des Master-Routers.
  - Initialize
    Der virtuelle Router wurde soeben eingeschaltet. In kurzer Zeit wird er in den Zustand "Master" oder "Backup" wechseln.
- **Master IP-Adresse** (Nur Online verfügbar)
  Zeigt die IPv4-Adresse des Master-Routers an.
- **Priorität**
  Zeigt die Priorität des virtuellen Routers an.
  Wenn eine IPv4-Adresse dem VRRP-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- **Advert. Intervall**
  Zeigt das Intervall an, in dem der Master-Router VRRP-Pakete verschickt.
- **Bevorzugen**
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - yes
    Dieser Router ist beim Rollenwechsel bevorzugt.
  - no
    Dieser Router ist beim Rollenwechsel nicht bevorzugt.

VRRP und DHCP-Server

Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, muss der DHCP-Server auf dem Master-Router konfiguriert sein. Backup-Router reagieren nicht auf DHCP-Anfragen. Stellen Sie sicher, dass der Master-Router statisch konfiguriert ist und nach einem Ausfall wieder zum Master der VRRP-Gruppe wird.

Konfiguration

Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

Hinweis

VRRP ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle / VRID**
  Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Primäre IP-Adresse**
  Wählen Sie die numerisch kleinste IPv4-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IPv4-Adresse..

  Hinweis

  Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist dann 0.0.0.0.
  Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IPv4-Adresse als Quelladresse für VRRP-Pakete genutzt wird, wählen Sie aus der Klappliste die entsprechende IPv4-Adresse aus. Ansonsten wird die numerisch kleinste IPv4-Adresse verwendet..
- **Master**
  Wenn diese Option aktiviert ist, wird die numerisch kleinste IPv4-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die vorrangige IPv4-Adresse des VRRP-Routers als virtuelle IP-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IPv4-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**
  Tragen Sie die Priorität dieses virtuellen Routers ein.
  Wenn eine IPv4-Adresse dem VRRP-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- **Advertisement-Intervall**
  Tragen Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRP-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**
  Erlauben sie die Bevorzugung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.

- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Priorität verringern**
  Geben Sie ein, um welchen Wert die Priorität der VRRP-Schnittstelle verringert wird.
- **Aktuelle Priorität**
  Zeigt die Priorität der VRRP-Schnittstelle an, nachdem die überwachte Schnittstelle in den Zustand "down" gewechselt hat.

Adressübersicht

Übersicht

Diese Seite zeigt an, welche IP-v4Adressen der virtuelle Router überwacht. Jeder virtueller Router kann maximal 10 IPv4-Adressen überwachen.

Hinweis

VRRP ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, die als virtueller Router fungiert.
- **VRID**
  Zeigt die ID dieses virtuellen Routers an.
- **Anzahl der Adressen**
  Zeigt die Anzahl der IPv4-Adressen an.
- **Zugeordnete IP-Adresse (1) ... Zugeordnete IP-Adresse (4)**
  Zeigt die Router IPv4-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IPv4-Adressen durch diesen Router übernommen.

Adresskonfiguration

Anlegen oder ändern der überwachten IPv4-Adressen

Auf dieser Seite können Sie die zu überwachenden IPv4-Adressen anlegen, ändern oder löschen. Maximal 10 IPv4-Adressen lassen sich von einem virtuellen Router überwachen.

Hinweis

VRRP ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Werte

Folgende Einstellungen sind möglich:

- **Schnittstelle / VRID**
  Wählen Sie den gewünschten virtuellen Router aus.
- **Zugeordnete IP-Adresse**
  Geben Sie die IPv4-Adresse ein, die der virtuelle Router überwachen soll. Maximal 10 IPv4-Adressen sind möglich.

Die Tabelle gliedert sich in folgende Spalte:

- **Zugeordnete IP-Adresse**
  Zeigt die IPv4-Adressen an, die der virtuelle Router überwacht.

Schnittstellenüberwachung

Einleitung

Auf dieser Seite konfigurieren Sie die Überwachung von Schnittstellen.

Wenn sich der Link einer überwachten Schnittstelle von "up" nach "down" ändert, wird die Priorität der zugeordneten VRRP-Schnittstelle verringert. Um welchen Wert die Priorität verringt wird, konfigurieren Sie auf der Seite "Layer 3 > VRRP/VRRPv3 > Konfiguration".

Wenn sich der Link der Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstelle wiederhergestellt.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste die Schnittstelle aus, die überwacht werden soll.
- **Track-ID**
  Geben Sie eine Track-ID ein.
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Track-Schnittstellenzähler**
  Geben Sie ein, wie viele der überwachten Schnittstellen in den Zustand "down" wechseln müssen, bevor die Priorität geändert wird.

Die Tabelle gliedert sich in folgende Spalten:

- Track-ID
  Zeigt die Track-ID an.
- Schnittstelle
  Zeigt die Schnittstelle an, die überwacht wird.

VRRPv3

Router (IPv4)

Einleitung

Auf dieser Seite legen Sie neue virtuelle Router an. Weitere Parameter konfigurieren Sie unter "Layer 3 (IPv4) > VRRPv3 > Konfiguration".

Hinweis

- Diese Funktion ist nur bei Layer 3 verfügbar.
- Der gleichzeitige Betrieb von VRRP und VRRPv3 ist nicht möglich.
- Aktivieren Sie das Optionskästchen "VRRPv3", um VRRPv3 zu konfigurieren.
- Sie können VRRPv3 an VLAN-Schnittstellen nutzen. Router-Ports werden nicht unterstützt.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- VRRPv3
  Aktivieren oder deaktivieren Sie das Routing über VRRPv3. Nur aktivierbar, wenn unter "Layer 3 (IPv6) > Konfiguration" "IPv6-Routing" aktiviert ist.
- Antworte an virtuellen Schnittstellen auf Ping-Anfragen
  Wenn aktiviert, antworten auch die virtuellen IPv4-Adressen auf den Ping.
- VRID-Tracking
  Aktivieren oder deaktivieren Sie das VRID-Tracking. Wenn aktiviert, werden alle VRRP-Instanzen überwacht. Wenn sich der Status einer VRRPInstanz nach "Initialize" ändert, wird die Priorität aller VRRP-Instanzen auf den Wert "1" verringert. Wenn sich der Status der VRRP-Instanz ändert, wird die unsprüngliche Priorität aller VRRPInstanzen wiederhergestellt.
- Schnittstelle
  Wählen Sie die gewünschte VLAN-Schnittstelle aus, die als virtueller Router fungiert.
- VRID
  Geben Sie die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden. Gültige Werte sind 1 ... 255.

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstelle
  Zeigt die IPv4-Schnittstelle an, die als virtueller Router fungiert.
- VRID
  Zeigt die ID des virtuellen Routers an.
- Virtuelle MAC-Adresse
  Zeigt die virtuelle MAC-Adresse des virtuellen Routers an.
- Primäre IP-Adresse
  Zeigt die kleinste numerische IPv4-Adresse auf diesem VLAN an. Der Eintrag 0.0.0.0 bedeutet, dass die "Primary" Adresse auf diesem VLAN verwendet wird. Ansonsten sind alle IPv4-Adressen, die auf diesem VLAN in dem Menü "Layer 3 (IPv4) > Subnetze" konfiguriert wurden, gültige Werte.
- Router-Status
  Zeigt den derzeitigen Zustand des virtuellen Routers an. Mögliche Werte sind:
  - Master
    Der Router ist der Master-Router und übernimmt die Routingfunktionalität für alle zugeordneten IPv4-Adressen.
  - Backup
    Der Router ist der Backup-Router. Wenn der Master-Router ausfällt, übernimmt der Backup-Router die Aufgaben des Master-Routers.
  - Initialize
    Der virtuelle Router wurde soeben eingeschaltet. In kurzer Zeit wird er in den Zustand "Master" oder "Backup" wechseln.
- Master IP-Adresse
  Zeigt die IPv4-Adresse des Master-Routers an.
- Priorität
  Zeigt die Priorität des virtuellen Routers an. Gültige Werte sind 1 ... 254. Wenn eine IPv4-Adresse dem VRRP-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- Advert.-Intervall
  Zeigt das Intervall an, in dem der Master-Router VRRPv3-Pakete verschickt.
- Unterbrechen
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - ja
    Dieser Router ist beim Rollenwechsel bevorzugt.
  - nein
    Dieser Router ist beim Rollenwechsel nicht bevorzugt.

VRRP und DHCP-Server

Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, muss der DHCP-Server auf dem Master-Router konfiguriert sein. Backup-Router reagieren nicht auf DHCP-Anfragen. Stellen Sie sicher, dass der Master-Router statisch konfiguriert ist und nach einem Ausfall wieder zum Master der VRRP-Gruppe wird.

Konfiguration (IPv4)

Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Schnittstelle / VRID**
  Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Primäre Adresse**
  Wählen Sie die vorrangige IPv4-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IPv4-Adresse.

  Hinweis

  Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist dann 0.0.0.0.
  Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IPv4-Adresse als Quelladresse für VRRP-Pakete genutzt wird, wählen Sie die entsprechende IPv4-Adresse aus. Ansonsten wird die numerisch kleinste IPv4-Adresse verwendet.

- **Master**
  Wenn aktiviert, wird die numerisch kleinste IPv4-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die numerisch kleinste IPv4-Adresse des VRRPv3-Routers als virtuelle IP-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IPv4-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**
  Gegen Sie die Priorität dieses virtuellen Routers ein. Gültige Werte sind 1-254.
  Wenn eine IPv4-Adresse dem VRRPv3-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRPv3-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRPv3-Router zum "Master".

- **Advertisement-Intervall**
  Geben Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRPv3-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**
  Erlauben sie die Bevorzugung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.
- **VRRP-Kompatibilitätsmodus**
  Wenn aktiviert, sendet und empfängt der VRRPv3-Router für konfigurierte IPv4-Adressen zusätzlich zu den VRRPv3-Telegrammen auch VRRPv2-Telegramme. Nur notwendig, wenn nicht alle VRRP-Router VRRPv3 unterstützen
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Priorität verringern**
  Geben Sie ein, um welchen Wert die Priorität der VRRPv3-Schnittstelle verringert wird.
- **Aktuelle Priorität**
  Zeigt die Priorität der VRRPv3-Schnittstelle an, nachdem die überwachte Schnittstelle in den Zustand "down" gewechselt hat.

Adressübersicht (IPv4)

Übersicht

Diese Seite zeigt an, welche IPv4-Adressen der virtuelle Router überwacht. Jeder virtueller Router kann maximal 10 IPv4-Adressen überwachen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, der als virtueller Router fungiert.
- **VRID**
  Zeigt die ID dieses virtuellen Routers an.
- **Anzahl der Adressen**
  Zeigt die Anzahl der IPv4-Adressen an.
- **Zugeordnete IP-Adresse (1) ... Zugeordnete IP-Adresse (10)**
  Zeigt die Router IPv4-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IPv4-Adressen durch diesen Router übernommen.

Adresskonfiguration (IPv4)

Anlegen oder ändern der überwachten IP-Adressen

Auf dieser Seite können Sie die zu überwachenden IPv4-Adressen anlegen, ändern oder löschen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Schnittstelle / VRID**
  Wählen Sie die ID des virtuellen Routers aus.
- **Zugeordnete IP-Adresse**
  Geben Sie die IPv4-Adresse ein, die der virtuelle Router überwachen soll. Maximal 10 IPv4-Adressen sind möglich.

Die Tabelle gliedert sich in folgende Spalten:

- **Zugeordnete IP-Adresse**
  Zeigt die IPv4-Adressen an, die der virtuelle Router überwacht.

Schnittstellenüberwachung (IPv4)

Einleitung

Auf dieser Seite konfigurieren Sie die Überwachung von Schnittstellen.

Wenn sich der Link einer überwachten Schnittstelle von "up" nach "down" ändert, wird die Priorität der zugeordneten VRRP-Schnittstelle verringert. Um welchen Wert die Priorität verringt wird, konfigurieren Sie auf der Seite "Layer 3 > VRRP/VRRPv3 > Konfiguration".

Wenn sich der Link der Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstelle wiederhergestellt.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste die Schnittstelle aus, die überwacht werden soll.
- **Track-ID**
  Geben Sie eine Track-ID ein.
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Track-Schnittstellenzähler**
  Geben Sie ein, wie viele der überwachten Schnittstellen in den Zustand "down" wechseln müssen, bevor die Priorität geändert wird.

Die Tabelle gliedert sich in folgende Spalten:

- **Track-ID**
  Zeigt die Track-ID an.
- **Schnittstelle**
  Zeigt die Schnittstelle an, die überwacht wird.

OSPFv2

Konfiguration

Einleitung

Auf dieser Seite konfigurieren Sie das Routing über OSPF.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **OSPFv2**
  Aktivieren oder deaktivieren Sie das Routing über OSPFv2.
- **Router-ID**
  Tragen Sie eine Bezeichnung einer der OSPFv2-Schnittstellen ein. Die Bezeichnung wird im IP-Adressformat eingegeben und muss nicht mit der eigenen IP-Adresse übereinstimmen. Die Router ID muss im Netzwerk eindeutig sein.
- **OSPFv2 - RFC1583-Kompatibilität**
  Aktivieren Sie die Option, wenn sie noch alte OSPFv2-Router in Betrieb haben, die nicht zu RFC 2328 kompatibel sind.

- **Border-Router**
  Zeigt den Status des OSPFv2-Routers an. Ist das lokale System aktives Mitglied in mindestens 2 Bereichen, ist es ein Area Border Router.
- **AS Border-Router**
  Legen Sie fest, ob der Router ein AS Border Router ist. Ein AS Border Router vermittelt zwischen mehreren autonomen Systemen, z. B. wenn Sie ein zusätzliches RIP Netzwerk haben. Ein AS Border Router ist auch nötig, um statische Routen hinzuzufügen und zu verteilen.
- **Neues LSA empfangen** (Nur Online verfügbar)
  Zeigt die Anzahl der LSAs an, die empfangen wurden. Aktualisierungen und eigene LSAs werden nicht gezählt.
- **Neues LSA konfiguriert** (Nur Online verfügbar)
  Zeigt die Anzahl der unterschiedlichen LSAs an, die von diesem lokalen System versendet wurden.
- **Maximale Anzahl externer LSAs**
  Um die Einträge von externen LSAs in der Datenbank zu begrenzen, tragen Sie die maximale Anzahl der External LSAs ein.
- **Exit-Intervall [s]**
  Tragen Sie das Intervall ein, nach dem der OSPFv2-Router wieder versucht aus dem Overflow-Status herauszukommen. Eine 0 bedeutet, dass der OSPFv2-Router erst nach einem Hochlauf wieder versucht den Overflow Status zu verlassen.
- **Inbound-Filter**
  Wählen Sie eine Route Map aus, die ankommende Routen filtert.
- **Weiterverteilung von Routen**
  Legen Sie fest, welche bekannten Routen über OSPFv2 weitergegeben werden. Sie können unterschiedliche Entscheidungen für die Routearten Default, Connected und Static treffen.

  Hinweis

  Die Einstellungen lassen sich nur an einem AS Border Router aktivieren. Besonders die Aktivierung der Optionen Default und Static kann zu Problemen führen, wenn sie an zu vielen Stellen im Netz aktiviert werden, z. B. Forwarding Loops

- **Route Map**
  Wählen Sie eine Route Map aus, die filtert, welche Routen über OSPFv2 weitergegeben werden.
- **Protokollspezifische Einstellungen**
  - **Default-Distanz**
    Legen Sie die administrative Distanz für das Protokoll OSPFv2 fest.
- **Route Map-spezifische Einstellungen**
  - **Route Map-Distanz**
    Wählen Sie aus der Klappliste eine Route Map aus, für die Sie die administrative Distanz festlegen wollen.
  - **Distanz**
    Konfigurieren Sie die administrative Distanz für die zuvor ausgewählre Route Map.

Areas

Übersicht

Ein autonomes Netzwerk (Autonomous System) kann in kleinere Bereiche (Areas) unterteilt werden.

Auf dieser Seite können Sie die Bereiche des Routers ansehen, anlegen, ändern oder löschen.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Werte

- **Area-ID**
  Geben Sie die Bezeichnung des Bereichs ein. Für alle Router eines Bereichs wird die Datenbank synchronisiert. Die Bereichsbezeichnung muss im Netzwerk eindeutig sein. Die Bereichsbezeichnung ist eine 32 Bit Zahl in folgendem Eingabeformat: x.x.x.x mit x = 0 ... 255
  Die Bereichsbezeichnung 0.0.0.0 ist für den Backbone-Bereich reserviert und kann nicht gelöscht werden.

Die Tabelle enthält folgende Spalten:

- **Area-ID**
  Zeigt die ID des Bereichs an.
- **Area-Typ**
  Wählen Sie in der Klappliste den Bereichstyp aus.
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Summary**
  Legen Sie fest, ob für diesen Bereich Summary LSAs erzeugt werden.
  - Summary: Summary LSAs werden in den Bereich verschickt.
  - No Summary: Summary LSAs werden nicht in den Bereich verschickt.
- **Metrik**
  Zeigt die Kosten für die OSPFv2-Schnittstelle an.
- **Aktualisierungen**
  Zeigt die Anzahl der Neuberechnungen der Routing-Tabellen an.
- **LSA-Zähler**
  Zeigt die Anzahl der LSA in der Datenbank an.

- Area BR
  Zeigt die Anzahl der erreichbaren Area Border Router (ABR) innerhalb dieses Bereichs an.
- AS BR
  Zeigt die Anzahl der erreichbaren Autonomous System Border Router (ASBR) in dieser Area an.

Area-Bereich

Anlegen einer neuen OSPFv2 Bereichsumfanges.

Auf dieser Seite können Netzwerke unter einer Area-ID zusammengefasst werden. Das Verfahren wird nur bei Area Border Routern angewandt. Damit propagiert ein Area Border-Router nach außen lediglich eine Route für zusammengefasste Adressbereiche.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Area-ID
  Wählen Sie die ID des Bereichs aus. Die ID konfigurieren Sie unter "Layer 3 > OSPFv2 > Areas".
- Subnetzadresse
  Tragen Sie die IPv4-Adresse des Netzwerks ein, das zusammengefasst wird.
- Subnetzmaske
  Tragen Sie die Subnetzmaske des Netzwerks ein, das zusammengefasst wird.

Die Tabelle enthält folgende Spalten:

- Area-ID
  Zeigt die ID des Bereichs an.
- Subnetzadresse
  Zeigt die IPv4-Adresse des Netzwerks an, das mit anderen Netzwerken zusammengefasst wird.
- Subnetzmaske
  Zeigt die Subnetzmaske des Netzwerks an, das mit anderen Netzwerken zusammengefasst wird.
- Advertise
  Aktivieren Sie diese Einstellung, um das zusammengefasste Netzwerk zu veröffentlichen.

Schnittstellen

Übersicht

Auf dieser Seite können Sie OSPFv2-Schnittstellen konfigurieren.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **IP-Adresse**
  Wählen Sie die IPv4-Adresse der OSPFv2-Schnittstelle aus.
- **Area ID**
  Wählen Sie die ID des Bereichs aus, mit der die OSPFv2-Schnittstelle verbunden ist.

  Hinweis

  Wählen Sie für die sekundäre Schnittstelle die gleiche Area-ID aus wie für die dazugehörige primäre Schnittstelle.
  Die Information, ob es sich um eine primäre oder sekundäre Schnittstelle handelt, finden Sie in der Spalte "Adresstyp" auf der Seite "Layer 3 (IPv4) > Subnetze > "Übersicht".

Die Tabelle enthält folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv4-Adresse der OSPFv2-Schnittstelle an.
- **Adresstyp**
  Es gibt zwei Adresstypen:
  - Primär
  - Sekundär

  Zeilen für sekundäre Adressen sind gegraut und zeigen die Werte der zugehörigen primären Adresse.
- **Area ID**
  Wählen Sie die ID des Bereichs aus, mit der die OSPFv2-Schnittstelle verbunden ist.
- **Passive Schnittstelle**
  Legen Sie das Verhalten der Schnittstelle fest:
  - Aktiviert
    Über diese Schnittstelle werden keine OSPFv2-Informationen versendet (z. B. Hello-Telegramme und LSDB-Updates) und gelernt.
  - Deaktiviert
    Über diese Schnittstelle werden OSPFv2-Informationen versendet und gelernt.
- **Metrik**
  Tragen Sie die Kosten für die OSPFv2-Schnittstelle ein.

- **Priorität**
  Tragen Sie die Router-Priorität ein. Die Priorität spielt nur bei der Auswahl des designierten Routers oder designierten Border Routers eine Rolle. Dieser Parameter kann auf Routern des gleichen Subnetzes unterschiedlich gewählt werden.
  Wertebereich: 0 bis 255
  Default-Einstellung: 1
- **Trans. Delay**
  Tragen Sie die gewünschte Verzögerung beim Versenden einer Verbindungsaktualisierung ein.
  Wertebereich: 1 s ... 3600 s
  Default-Einstellung: 1 s
- **Retrans. Delay**
  Tragen Sie die Zeit ein, nach der ein OSPFv2-Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
  Wertebereich: 1s ... 3600 s
  Default-Einstellung: 5 s
- **Hello-Intervall**
  Tragen Sie den Abstand zwischen zwei Hello-Paketen ein.
  Wertebereich: 1s ... 65.535 s
  Default-Einstellung: 10 s
- **Dead-Intervall**
  Tragen Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" markiert wird, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.
  Default-Einstellung: 40 s

Schnittstellen-Authentifizierung

Konfigurieren der Schnittstellen Anmeldung

Auf dieser Seite definieren Sie die Authentifizierung der Schnittstelle.

Hinweis

OSPF nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **OSPF-Schnittstelle**
  Wählen Sie die OSPFv2-Schnittstelle aus, für die Sie die Authentifizierung konfigurieren wollen.
- **Authentifizierungstyp**
  Wählen Sie die Authentifizierungsmethode der OSPF-Schnittstelle aus. Sie haben folgende Möglichkeiten:
  - Keine:
    Keine Authentifizierung
  - Einfach:
    Authentifizierung über ein unverschlüsseltes Passwort
  - MD5:
    Authentifizierung über MD5

Abschnitt "Einfache Authentifizierung"

- **Passwort**
  Geben Sie ein Passwort ein.
- **Bestätigung**
  Bestätigen Sie das eingegebene Passwort.

Abschnitt "MD5-Authentifizierung"

- **Authentifizierungsschlüssel-ID**
  Geben Sie die Kennung des MD5 Authenifizierungsschlüssels an. Tragen Sie für die MD5-Authentifizierung die ID ein, unter der das Passwort als Schlüssel verwendet wird. Da die Schlüssel-ID mit dem Protokoll übertragen wird, muss bei allen benachbarten Routern der gleiche Schlüssel unter der gleichen Schlüssel-ID gespeichert werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Authentifizierungsschlüssel-ID**
  Nur editierbar, wenn Sie die Authentifizierungsmethode MD5 einstellen. Nur dort können mehrere Schlüssel verwendet werden.
- **MD5-Schlüssel**
  Tragen Sie den MD5-Schlüssel ein.
- **MD5-Schlüssel bestätigen**
  Bestätigen Sie den eingegebenen Schlüssel.
- **Neueste Schlüssel-ID** (Nur Online verfügbar)
  Gibt an, ob der MD5-Schlüssel die neueste Schlüssel-ID ist.

Virtual Links

Übersicht

Jeder Area Border Router muss aus protokolltechnischen Gründen Zugang zum Backbone-Bereich erhalten. Wenn ein Router nicht direkt am Backbone-Bereich angeschlossen ist, wird eine virtuelle Verbindung dorthin angelegt.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Hinweis

Beachten Sie, dass beim Anlegen einer virtuellen Verbindung sowohl der Transit-Bereich als auch der Backbone-Bereich bereits konfiguriert sein müssen.

Die virtuelle Verbindung muss auf beiden Seiten gleich konfiguriert werden.

Beschreibung der angezeigten Felder

Die Seite enthält folgenden Hinweis:

- **Virtuelle Links sind nicht wirksam, solange das Gerät kein ABR ist.** (nur Online verfügbar)
  Dieser Hinweis wird angezeigt, wenn mindestens ein Virtual Link-Eintrag projektiert ist und das Gerät kein Area Border Router ist.

Die Seite enthält folgende Felder:

- **Nachbar-Router-ID**
  Tragen Sie die ID des Nachbar-Routers auf der anderen Seite der virtuellen Verbindung ein.
- **Transit Area ID**
  Wählen Sie die ID des Bereichs, der die beiden Router verbindet.

Die Tabelle enthält folgende Spalten:

- **Transit Area ID**
  Zeigt die ID an, über die die beiden Router verbunden sind.
- **Nachbar-Router-ID**
  Zeigt die ID des Nachbar-Routers auf der anderen Seite der virtuellen Verbindung an.
- **Virt. Link-Status**
  Legen Sie fest, in welchem Zustand sich die virtuelle Verbindung befindet. Folgende Zustände sind möglich:
  - down: Die virtuelle Verbindung ist inaktiv.
  - point-to-point: Die virtuelle Verbindung ist aktiv.
- **Trans. Delay**
  Tragen Sie die gewünschte Verzögerung bei dem Versenden eines Link Update Paketes ein.
  Wertebereich: 1 s bis 3600 s
  Default-Einstellung: 1 s

- **Retrans. Delay**
  Tragen Sie die Zeit ein, nach der ein Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
  Wertebereich: 1s bis 3600 s
  Default-Einstellung: 5 s
- **Hello-Intervall**
  Tragen Sie den Abstand zwischen zwei Hello-Paketen ein.
  Wertebereich: 1s bis 65.535 s
  Default-Einstellung: 10 s
- **Dead-Intervall**
  Tragen Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" gilt, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.
  Default-Einstellung: 40 s

Virtual Links-Authentifizierung

Konfigurieren der Schnittstellen-Anmeldung

Auf dieser Seite definieren Sie die Authentifizierung der Schnittstelle.

Hinweis

OSPF ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Virtual Link (Area/Nachbar)**
  Wählen Sie den virutellen Link aus, für den Sie die Authentifizierung konfigurieren wollen.
- **Authentifizierungstyp**
  Wählen Sie die Authentifizierungsmethode der OSPF-Schnittstelle aus. Sie haben folgende Möglichkeiten:
  - Keine
    Keine Authentifizierung
  - Einfach
    Authentifizierung über ein unverschlüsseltes Passwort.
  - MD5
    Authentifizierung über MD5

Abschnitt "Simple Authentication"

- **Passwort**
  Geben Sie das Passwort ein.
- **Bestätigung**
  Bestätigen Sie das eingegebene Passwort.

Abschnitt "MD5-Authentifizierung"

- Authentifizierungsschlüssel-ID
  Geben Sie die Kennung des MD5 Authenifizierungsschlüssels an. Tragen Sie für die MD5-Authentifizierung die ID ein, unter der das Passwort als Schlüssel verwendet wird.
  Da die Schlüssel-ID mit dem Protokoll übertragen wird, muss bei allen benachbarten Routern der gleiche Schlüssel unter der gleichen Schlüssel-ID gespeichert werden.

Die Tabelle gliedert sich in folgende Spalten:

- Authentifizierungsschlüssel-ID
  Nur editierbar, wenn Sie die Authentifizierungsmethode MD5 einstellen. Nur dort können mehrere Schlüssel verwendet werden.
- MD5-Schlüssel
  Tragen Sie den MD5-Schlüssel ein.
- MD5-Schlüssel bestätigen
  Bestätigen Sie den eingegebenen Schlüssel.
- Neueste Schlüssel-ID
  Gibt an, ob der MD5-Schlüssel die neueste Schlüssel-ID ist.

RIPv2

Konfiguration

Auf dieser Seite konfigurieren Sie das Routing über RIP.

Hinweis

RIPv2 ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- RIPv2
  Aktivieren oder deaktivieren Sie das Routing über RIPv2.
- Inbound-Filter
  Wählen Sie eine Route Map aus, die ankommende Routen filtert.
- Weiterverteilung von Routen
  Legen Sie fest, welche bekannten Routen über RIPv2 weitergegeben werden.
  Folgende Routenarten gibt es:
  - Static Default
  - Connected
  - Static
  - OSPF
- Route Map
  Wählen Sie eine Route Map aus, die filtert, welche Routen über RIPv2 weitergegeben werden.

- **Protokollspezifische Einstellungen**
  - **Default-Distanz**
    Legen Sie die administrative Distanz für das Protokoll OSPFv2 fest.
- **Route Map-spezifische Einstellungen**
  - **Route Map-Distanz**
    Wählen Sie aus der Klappliste eine Route Map aus, für die Sie die administrative Distanz festlegen wollen.
  - **Distanz**
    Konfigurieren Sie die administrative Distanz für die zuvor ausgewählre Route Map.

Schnittstellen

Übersicht

Auf dieser Seite können Sie RIPv2-Schnittstellen konfigurieren.

Hinweis

RIPv2 ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **IP-Adresse**
  Wählen Sie die IPv4-Adresse der RIPv2-Schnittstelle aus.

Die Tabelle enthält folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv4-Adresse der RIPv2-Schnittstelle an.
- **Updates versenden**
  Wählen Sie aus, auf welche Weise Updates gesendet werden:
  - no send
    Es werden keine Updates versendet.
  - RIPv1
    Es werden Updates für RIPv1 versendet.
  - RIPv1-compat.
    Es werden RIPv2-Updates nach den Regeln von RIPv1 als Broadcasts versendet.
  - RIPv2
    Es werden Updates für RIPv2 als Multicasts versendet.
  - RIPv1 demand/RIPv2 demand
    RIP-Pakete werden nur als Antworten auf explizites Nachfragen versendet.

- **Updates empfangen**
  Wählen Sie aus, in welcher Form empfangene Updates akzeptiert werden:
  - no receive
    Es werden keine Updates empfangen.
  - RIPv1
    Es werden nur Updates von RIPv1 empfangen.
  - RIPv2
    Es werden nur Updates von RIPv2 empfangen.
  - RIPv1/v2
    Es werden Updates von RIPv1 und RIPv2 empfangen.
- **Standard-Metrik**
  Tragen Sie die Kosten für die RIPv2-Schnittstelle ein.

IGMP

IGMP

Auf dieser Seite konfigurieren Sie IGMP (Internet Group Management Protocol). IGMP ist ein Netzwerkprotokoll, das für IP-Multicast eingesetzt wird. Beim IP-Multicasting werden IP-Pakete unter einer IP-Adresse an mehrere Clients gleichzeitig verteilt. IGMP kann dynamische und statische Multicastgruppen verwalten.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **IGMP**
  Aktivieren oder deaktivieren Sie IGMP (Internet Group Management Protocol) für das gesamte Gerät.
- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus, auf der Sie IGMP konfigurieren wollen.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an.
- **Version**
  Wählen Sie die Version von IGMP aus.
- **Query-Intervall [s]**
  Definieren Sie das Intervall, in dem der Multicastrouter IGMP-Queries versendet.
  Angabe in Sekunden
- **Max. Reaktionszeit [1/10 s]**
  Legen Sie die Zeitspanne in einem IGMP-Query fest, die der Client Zeit hat, um auf eine Anfrage zu antworten.
  Angabe in Zehntelsekunden

- **Explizite Verfolgung**
  Aktivieren oder deaktivieren Sie die Funktion "Explizite Verfolgung" für die Schnittstelle. Wenn die Funktion "Explizite Verfolgung" aktiviert ist, trackt der Multicastrouter genau, welche Clients sich bei einer Multicastgruppe anmelden (Join) und abmelden (Leave). Sobald sich der letzte Client bei einer Multicastgruppe abgemeldet hat, löscht der Multicastrouter die Multicastgruppe. Da der Multicastrouter genaue Informationen über die an- und abgemeldeten Clients hat, muss der Multicastrouter nicht nachfragen, ob noch ein Client den Multicast empfangen will.
- **Admin-Status**
  Aktivieren oder deaktivieren Sie IGMP auf der Schnittstelle.
- **Betriebszustand**
  Zeigt den Betriebszustand der Schnittstelle an, siehe auch "System > Ports > Übersicht":
  - Up
  - Down

PIM

Konfiguration

Auf dieser Seite konfigurieren Sie PIM.

Hinweis

PIM ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **PIM-Routing**
  Aktivieren oder deaktivieren Sie PIM global auf dem Gerät.
- **Statischer Rendezvous Punkt (RP)**
  Aktivieren oder deaktivieren Sie, dass das Gerät als statischer Rendezvous Punkt fungiert.
- **Bidirektionaler Multicast**
  Aktivieren oder deaktivieren Sie die Funktionalität "Bidirektionaler Multicast" global auf dem Gerät.

Schnittstelle

Übersicht

Auf dieser Seite konfigurieren Sie PIM-Schnittstellen.

Hinweis

PIM ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie eine VLAN-IP-Schnittstelle aus.
- **PIM**
  Wenn Sie dieses Optionskästchen aktivieren, ist die ausgewählte Schnittstelle PIM zugeordnet.
- **Bootstrap Router (BSR) Kandidat**
  Wenn Sie dieses Optionskästchen aktivieren, legen Sie fest, dass die Schnittstelle Kandidat für den Bootstrap-Router werden soll. Ein BSR koordiniert die Rendevous Punkte in einem PIM-Netzwerk.
- **BSR-Wert des Kandidaten**
  Geben Sie die Priorität der Schnittstelle als BSR-Kandidat ein.
  Anhand der Priorität wird festgelegt, welcher Kandidat BSR wird. Der Kandidat mit der höchsten Priorität wird BSR.
  Ist für zwei Kandidaten die gleiche Priorität vorgegeben, wird im Bedarfsfall anhand der IP-Adresse der Schnittstelle entschieden. Die Schnittstelle mit der höheren IP-Adresse wird BSR.
  Ein BSR koordiniert die Rendevous Punkte in einem PIM-Netzwerk.
- **Designated Router (DR) Priorität**
  Geben Sie die DR-Priorität der Schnittstelle ein.
  Anhand der RP-Priorität wird festgelegt, welcher Kandidat DR wird. Der Kandidat mit der höchsten Priorität wird DR.
  Ist für zwei Kandidaten die gleiche Priorität vorgegeben, wird im Bedarfsfall anhand der IP-Adresse der Schnittstelle entschieden. Die Schnittstelle mit der höheren IP-Adresse wird DR.
  Der DR leitet die IGMP-Joins an den Rendezvous Punkt weiter.
- **BSR-Border**
  Wenn aktiviert, findet über diese Schnittstelle keine BSR-Kommunikation statt. Dadurch wird ein PIM-Netzwerk in PIM-Domänen unterteilt.

RP-Statisch

Übersicht

Auf dieser Seite legen Sie fest, welche Schnittstellen statische Rendezvous Punkte werden sollen.

Konfigurieren Sie den statischen RP auf allen Geräten der PIM-Komponente.

Hinweis

PIM ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Adresse der Multicastgruppe**
  Geben Sie die Adresse der Multicastgruppe ein, für die die Schnittstelle RP werden soll.
- **Maske der Multicastgruppe**
  Geben Sie Subnetzmaske ein, die das Multicastband einschränkt.
- **RP-Adresse**
  Geben Sie die IP-Adresse der Schnittstelle ein, die RP werden soll.
- **Bidirektionaler Multicast**
  Aktivieren oder deaktivieren Sie die Funktionalität "Bidirektionaler Multicast" für eine Multicastgruppe.
  Konfigurieren Sie die Funktionalität für eine Multicastgruppe auf allen RPs gleich.
  Wenn Sie die Funktion für eine RP-Adresse aktivieren, wird die Funktion bei allen bestehenden und zukünftigen Einträgen dieser RP-Adresse aktiviert.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Adresse der Multicastgruppe**
  Zeigt die Adresse der Multicastgruppe an.
- **Maske der Multicastgruppe**
  Zeigt die Subnetzmaske an.
- **RP-Adresse**
  Zeigt die IP-Adresse des RPs an.
- **Bidirektionaler Multicast**
  Zeigt an, ob die Funktionalität "Bidirektionaler Multicast" für eine Multicastgruppe aktiviert oder deaktiviert ist.

RP-Kandidat

Übersicht

Auf dieser Seite legen Sie fest, welche Schnittstellen Kandidaten für den Rendezvous Punkt (RP) werden sollen.

Sie können innerhalb eines PIM-Netzwerks mehrere Kandidaten für den RP konfigurieren. Der BSR koordiniert die Kandidaten und bestimmt den RP.

Hinweis

PIM ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Adresse der Multicastgruppe**
  Geben Sie die Adresse der Multicastgruppe ein.
- **Maske der Multicastgruppe**
  Geben Sie Subnetzmaske ein, die das Multicastband einschränkt.
- **RP-Schnittstelle**
  Wählen Sie die Schnittstelle aus, die RP-Kandidat werden soll.
- **Bidirektionaler Multicast**
  Aktivieren oder deaktivieren Sie die Funktionalität "Bidirektionaler Multicast" für eine Multicastgruppe.
  Konfigurieren Sie die Funktionalität für eine Multicastgruppe auf allen RPs gleich.
  Wenn Sie die Funktion für eine RP-Schnittstelle aktivieren, wird die Funktion bei allen bestehenden und zukünftigen Einträgen dieser RP-Schnittstelle aktiviert.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Adresse der Multicastgruppe**
  Zeigt die Adresse der Multicastgruppe an.
- **Maske der Multicastgruppe**
  Zeigt die Subnetzmaske an.
- **RP-Kandidat**
  Zeigt die IP-Adresse des RPs an.
- **Bidirektionaler Multicast**
  Zeigt an, ob die Funktionalität "Bidirektionaler Multicast" für eine Multicastgruppe aktiviert oder deaktiviert ist.
- **RP-Priorität**
  Geben Sie die Priorität der Schnittstelle ein.
  Anhand der RP-Priorität wird festgelegt, welcher Kandidat Rendezvous Punkt wird. Der Kandidat mit der höchsten Priorität wird RP. Je kleiner der Wert, desto höher die Priorität. Ist für zwei Kandidaten die gleiche Priorität vorgegeben, wird im Bedarfsfall anhand der IP-Adresse der Schnittstelle entschieden. Die Schnittstelle mit der kleinsten IP-Adresse wird RP.

MSDP

Konfiguration

Übersicht

Auf dieser Seite können Sie MSDP konfigurieren.

MSDP (Multicast Source Discovery Protocol) wird innerhalb einer PIM-SM-Domäne in Kombination mit Anycast-RPs unterstützt. Bei Anycast-RPs handelt es sich um ein

Redundanzkonzept für RPs. Die beteiligten RPs bekommen alle die gleiche IP-Adresse und werden im Netz bekannt gemacht. Über MSDP-Sessions sind die RPs miteinander verknüpft. Die RPs synchronisieren ihre Multicastquellen über Source-Active-Nachrichten.

Konfigurieren Sie MSDP auf den Rendezvous Punkten.

Sie können MSDP nur nutzen, wenn die Funktionalität "Bidirektionaler Multicast" in dem PIM-Netzwerk deaktiviert ist.

Hinweis

MSDP ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **MSDP**
  Aktivieren oder deaktivieren Sie MSDP.
- **Originator-ID**
  Wählen Sie eine Schnittstelle zur Identifikation des MSDP-Teilnehmers aus. Die Originator-ID muss im Netzwerk eindeutig sein.
- **Cache Speicherdauer**
  Geben Sie die Zeit an, nach der ein Eintrag mit Informationen eines anderen RPs aus dem SA-Cache gelöscht wird, wenn keine SA-Cache-Updates empfangen werden. Nachdem ein Eintrag erstellt bzw. aktualisiert wurde, startet ein Timer und läuft mit der definierten Zeit rückwärts. SA-Cache-Updates werden zyklisch empfangen und setzen den Timer des Eintrags zurück. Wenn der Timer abläuft, ohne dass in dieser Zeit ein SA-Cache-Update empfangen wurde, wird der entsprechende Eintrag gelöscht.
  Wenn der Wert "0" eingestellt ist, werden keine SA-Cache-Einträge gespeichert.
- **Listener Port**
  TCP-Port, über den das Gerät mit anderen MSDP-Teilnehmern kommuniziert. Wenn Sie den Listener Port ändern, werden nur neue MSDP-Verbindungen beeinflusst, die nach dieser Änderung erstellt werden. Bestehende MSDP-Verbindungen nutzen weiterhin den vorherigen Listener Port.

Peer

Übersicht

Auf dieser Seite konfigurieren Sie MSDP-Partner.

Hinweis

MSDP ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Peer-Adresse**
  Geben Sie die IP-Adresse des MSDP-Partners ein, zu dem das Gerät eine Verbindung aufbauen soll.
- **Verbundene Schnittstelle**
  Wählen Sie die Schnittstelle aus, über die das Gerät mit dem MSDP-Partner kommuniziert.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **Peer-Adresse**
  Zeigt die Adresse des MSDP-Partners an.
- **Lokale Adresse**
  Zeigt die IP-Adresse der Schnittstelle an, über die das Gerät mit dem MSDP-Partner kommuniziert.
- **Admin-Status**
  Aktivieren oder deaktivieren Sie die Verbindung.
- **Verbindungsstatus**
  Zeigt den Status der Verbindung an.
  - Listening
    Das Gerät wartet auf die Verbindung.
  - Connecting
    Verbindungsaufbau
  - Established
    Die Verbindung ist aufgebaut.
  - Inactive
    Die Verbindung ist nicht aktiv.
  - Disabled
    Die Verbindung ist deaktiviert.
- **Keepalive-Intervall[s]**
  Definieren Sie das Intervall, in dem das Gerät Keepalive-Nachrichten an die MSDP-Partner versendet.
- **Peer-Speicherzeit-Intervall[s]**
  Definieren Sie die Zeit, nach der eine MSDP-Verbindung getrennt wird, wenn keine Keepalive-Nachrichten vom MSDP-Partner empfangen werden.
- **Connect Retry-Intervall[s]**
  Definieren Sie das Intervall, nach dessen Ablauf versucht wird, eine Verbindgung herzustellen. Wenn zu einem Partner keine MSDP-Verbindung besteht, wird nach Ablauf des Connect Retry-Intervalls versucht eine Verbindgung herzustellen.
- **TTL-Schwellenwert**
  Definieren Sie die maximale Anzahl der Router, die zu dem MSDP-Partner durchlaufen werden darf (Hop Count). Wenn der angegebene Wert überschritten wird, wird das Telegramm verworfen.

Layer 3-Funktionen für IPv6 projektieren

Konfiguration

Einleitung

Die Seite enthält die Übersicht über die Layer 3-Funktionen für IPv6. Auf dieser Seite aktivieren oder deaktivieren Sie die gewünschte Layer 3-Funktion.

Die Funktionen "IPv6 Routing", "VRRPv3" und "OSPFv3" sind nur bei Layer 3 (IPv6) verfügbar.

Beschreibung der angezeigten Felder

- **IPv6 Routing** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  - Aktiviert
    IPv6-Routing wird aktiviert. Ist am Gerät IPv4-Routing nicht aktiviert, wird dies ebenfalls mit dieser Funktion aktiviert.
  - Deaktiviert
    IPv6-Routing wird deaktiviert.
- **VRRPv3** (nur bei Geräten mit Layer 3-Lizenz verfügbar,)
  Aktivieren oder deaktivieren Sie Routing über VRRPv3. Um VRRPv3 zu verwenden, aktivieren Sie zuerst die Funktion IPv6-Routing. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv6)> VRRPv3".
- **OSPFv3** (nur bei Geräten mit Layer 3-Lizenz verfügbar)
  Aktivieren oder deaktivieren Sie Routing über OSPFv3. Weitere Einstellungen konfigurieren Sie unter "Layer 3 (IPv6) > OSPFv3".

Subnetze (IPv6)

Verbundene Subnetze

Auf dieser Seite können Sie IPv6 an der Schnittstelle aktivieren. Die Schnittstelle wird auch als IPv6-Schnittstelle bezeichnet. Eine IPv6-Schnittstelle kann mehrere IPv6-Adressen haben.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Schnittstelle**
  Wählen Sie die IP-Schnittstelle aus, für die IPv6 aktiviert wird.
- **IPv6 aktivieren**
  Aktivieren oder deaktivieren Sie IPv6 für diese Schnittstelle.

  Hinweis

  IPv6 deaktivieren

  Sobald IPv6 auf der IP-Schnittstelle aktiviert ist, kann IPv6 nur durch Entfernen der Schnittstelle deaktiviert werden

- **IPv6-Adresse**
  Geben Sie die IPv6-Adresse an. Die Eingabe ist abhängig von der gewählten Adressart.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.
- **IPv6-Adresstyp**
  Wählen Sie die Adressart aus:
  - Unicast
  - Anycast
  - Link-lokal: IPv6-Adresse ist nur auf dem Link gültig
- **Adressen-Autokonfiguration (SLAAC)**
  Aktivieren oder deaktivieren Sie den Mechanismus SLAAC (Stateless Address Auto Configuration) für die Adresskonfiguration.
  Wenn SLAAC aktiviert ist, erfolgt eine zustandslose Autokonfiguration über NDP (Neighbor Discovery Protocol).
  Wenn die Option deaktiviert ist, erfolgt die Adresskonfiguration zustandsbehaftet. Das Gerät bezieht die IPv6-Adresse und die Konfigurationsdatei von einem DHCPv6-Server.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstellenname**
  Zeigt den Namen der Schnittstelle.
- **IPv6-Adresse**
  Zeigt die IPv6-Adresse des Subnetzes an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **IPv6-Adresstyp**
  Zeigt den Adressentyp an. Folgende Werte sind möglich:
  - Unicast
  - Anycast
  - Link-lokal

- **Loopback**
  Zeigt an, ob die Eigenschaft "Loopback" aktiviert ist. Sie konfigurieren die Eigenschaft "Loopback" auf der Seite "Layer 3 (IPv4) > Subnetze > Übersicht".
- **Status der Erkennung von doppelten Adressen**
  Bei der Adressen-Autokonfiguration (SLAAC) verhindert die Funktion "Status der Erkennung von doppelten Adressen", dass IPv6-Adressen doppelt vergeben werden. Das Gerät darf bei der Autokonfiguration nur unvergebene IPv6-Adressen verwenden.
  Wenn die Funktion aktiviert ist, läuft die Prüfung über NDP automatisch ab.

  Hinweis

  Die Funktion führt keine zyklische Prüfung durch.

  Diese Spalte zeigt an, in welchem Status sich die Funktion befindet. Folgende Werte sind möglich:

  – Tentative
    Dieser Status zeigt an, dass die ausgewählte IPv6-Adresse geprüft wird. Das Gerät sendet eine Neighbor Solicitation-Nachricht an die ausgewählte IPv6-Adresse.
  – Conflict
    Dieser Status zeigt an, die IPv6-Adresse bereits verwendet wird. Das Gerät erhält in diesem Fall eine Neighbor-Advertisement-Nachricht mit der ausgewählten IPv6-Adresse zurück. Das Gerät bildet eine neue IPv6-Adresse und prüft diese erneut.
  – Complete
    Dieser Status zeigt an, dass die ausgewählte IPv6-Adresse verwendet werden kann. Das Gerät hat in diesem Fall innerhalb einer Zeitapanne keine Rückmeldung erhalten und geht davon aus, dass die IPv6-Adresse noch nicht vergeben ist.
  – Down
    Dieser Status zeigt an, dass die Schnittstelle nicht aktiv ist. Es wird keine Prüfung durchgeführt.

Statische Routen (IPv6)

Auf dieser Seite konfigurieren Sie statische IPv6-Routen.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Zielnetzwerk**
  Geben Sie die Netzwerkadresse des Ziels ein, das über diese Route erreichbar ist.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.
- **Gateway**
  Geben Sie die IPv6-Adresse des Gateways ein, über den diese Netzwerkadresse erreichbar ist.

- **Metrik**
  Geben Sie die Metrik für die Route ein. Die Metrik entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt.
  Wertebereich: 1 - 254
- **Schnittstelle**
  Legen Sie die IPv6-Schnittstelle fest, über die die Netzwerkadresse des Ziels erreicht wird.

Die Tabelle enthält folgende Spalten:

- **Zielnetzwerk**
  Zeigt die IPv6-Netzwerkadresse des Ziels an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **Gateway**
  Zeigt die IPv6-Adresse des nächsten Gateways an.
- **Schnittstelle**
  Zeigt die Schnittstelle der Route an.
- **Metrik**
  Tragen Sie die Metrik für die Route ein. Beim Erstellen der Route wird automatisch "not used" eingetragen. Die Metrik entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt.
  Wertebereich: 1 - 254
- **Status**
  Zeigt an, ob die Route aktiv ist oder nicht.

Route Maps

Allgemein (IPv6)

Route Maps

Mit Route Maps können Sie kontrollieren, wie Routing-Informationen weiterverarbeitet werden. Sie können Routing-Informationen filtern und festlegen, ob die Informationen weiterverarbeitet, verändert oder verworfen werden.

Route Maps arbeiten nach folgendem Prinzip:

- Routing-Informationen werden mit den Filtern der Route Maps abgeglichen.
- Der Abgleich läuft solange, bis die Filter einer Route Map mit den Eigenschaften einer Information übereinstimmen.
- Die Information wird dann entsprechend den Einstellungen der Route Map verarbeitet:
  - Die Routing-Information wird verworfen.
  - Die Eigenschaften der Routing-Information werden verändert.

Beschreibung der angezeigten Felder

- **Name**
  Tragen Sie einen Namen für die Route Map ein.
- **Sequenznummer**
  Tragen Sie eine Nummer für die Route Map ein.
  Sie können mehrere Route Maps mit gleichem Namen, aber unterschiedlichen Sequenznummern, anlegen. Die Sequenznummer bestimmt dann die Reihenfolge, in der die Route Maps abgearbeitet werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen der Route Map an.
- **Sequenznummer**
  Zeigt die Sequenznummer der Route Map an.
- **Aktion**
  Legen Sie fest, was mit den Routing-Informationen passiert, die den Einstellungen der Route Map entsprechen:
  - permit
    Die Routing-Informationen werden weiterverarbeitet, entsprechend Ihren Einstellungen auf der Seite "Einstellen".
  - deny
    Die Routing-Informationen werden verworfen.

Schnittstelle & Wert filtern (IPv6)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen für eine Route Map nach Schnittstellen, Metrik oder Tags gefiltert werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.
  Es stehen Ihnen die angelegten Route Maps zur Verfügung.
- **Typ**
  Wählen Sie aus, wonach gefiltert werden soll:
  - Schnittstelle
  - Metrik
  - Tag
  - Routen-Typ
  - Metrik-Typ
- **Schnittstelle**
  Wählen Sie eine Schnittstelle aus.
  Nur aktiv, wenn Sie bei "Typ" den Eintrag "Schnittstelle" ausgewählt haben.

- **Metrik**
  Geben Sie für die Metrik einen Wert ein.
  Dieses Feld ist nur aktiv, wenn Sie bei "Typ" den Eintrag "Metrik" ausgewählt haben.
- **Tag**
  Geben Sie für den Tag einen Wert ein.
  Dieses Feld ist nur aktiv, wenn Sie bei "Typ" den Eintrag "Tag" ausgewählt haben.
- **Routen-Typ**
  Wählen Sie den Typ der Route aus:
  - Lokal
    Die Routing-Informationen für die Route Map werden nach direkt verbundenen Routen (lokale Schnittstellen) gefiltert.
  - Remote
    Die Routing-Informationen für die Route Map werden nach gelernten oder statisch konfigurierten Routen gefiltert.

  Dieses Feld ist nur aktiv, wenn Sie in der Klappliste "Typ" den Eintrag "Routen-Typ" ausgewählt haben.
- Metrik-Typ
  Wählen Sie den Typ der Metrik aus:
  - Inter-Area
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, die aus anderen Areas gelernt werden.
  - Intra-Area
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, die aus der gleichen Area stammen.
  - type-1-external
    Die Routing-Informationen für die Route Map werden nach Routen gefiltert, deren Pfadkosten (Metrik) sich aus den externen Pfadkosten und den Pfadkosten zum ASBR zusammensetzen.
  - type-2-external
    Die Routing-Informationen für die Route Map werden nach Routen mit nur externen Pfadkosten (Metrik) gefiltert.

  Dieses Feld ist nur aktiv, wenn Sie in der Klappliste "Typ" den Eintrag "Metrik-Typ" ausgewählt haben.

Die Tabelle gliedert sich in folgende Spalten:

- **Typ**
  Zeigt den ausgewählten Typ an:
  - Schnittstelle
  - Metrik
  - Tag
  - Routen-Typ
  - Metrik-Typ
- **Wert**
  Zeigt die ausgewählte Schnittstelle, den Wert der Metrik oder des Tags an.

Ziel filtern (IPv6)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen für eine Route Map nach der Ziel-IPv6-Adresse gefiltert werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq.-Nr.)**
  Wählen Sie eine Route Map aus.
- **IP-Adresse**
  Geben Sie die IPv6-Adresse des Ziels ein, nach der gefiltert werden soll.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IPv6-Adresse des Ziels an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.

Next-Hop filtern (IPv6)

Auf dieser Seite legen Sie fest, ob für eine Route Map danach gefiltert werden soll, zu welchem Router die Routing-Informationen als nächstes gesendet werden.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.
- **IP-Adresse**
  Geben Sie die IP-Adresse des Routers ein, an den die Routing-Informationen als nächstes gesendet werden sollen.

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**
  Zeigt die IP-Adresse des nächsten Routers an.

Einstellen (IPv6)

Auf dieser Seite legen Sie fest, ob die Routing-Informationen durch eine Route Map verändert werden.

Wenn Sie z. B. nach einer bestimmten Metrik gefiltert haben, können Sie hier den Wert der Metrik verändern. Die Routing-Informationen werden dann mit dem neuen Wert weitergeleitet. Sie können nur die Informationen einer "Permit"-Route Map verändern.

Beschreibung der angezeigten Felder

- **Route Map (Name/Seq-Nr.)**
  Wählen Sie eine Route Map aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen der Route Map an.
- **Sequenznummer**
  Zeigt die Sequenznummer der Route Map an.
- **Metrik**
  Geben Sie den neuen Wert für die Metrik ein, mit dem die Routing-Informationen weitergeleitet werden.
- **Tag**
  Geben Sie den neuen Wert für den Tag ein, mit dem die Routing-Informationen weitergeleitet werden.

DHCPv6 Relay Agent

Schnittstellen

Ein DHCPv6-Relay-Agent leitet die DHCP-Anfragen des DHCPv6-Clients an den DHCPv6-Server weiter. Bei der Weiterleitung wird die Remote-ID in die DHCP-Anfrage eingefügt. Die Remote-ID ist die Kennung der Schnittstelle. Damit kann der DHCPv6-Agent die Antwort des DHCPv6-Servers an den gewünschnten DHCPv6-Client senden.

Beschreibung der angezeigten Felder

- **Remote ID (Opt. 37)**
  Wenn aktiviert wird die DHCP Relay Agent-Option 37 (Remote-ID) verwendet. Legen Sie bei "Remote ID Type" die Art der Kennung fest.
- **Schnittstelle**
  Wählen Sie die IPv6-Schnittstelle aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellungen beziehen.
- **Aktuelle Remote-ID**
  Zeigt an, welche Remote-ID gerade verwendet wird.

- **Remote-ID-Typ**
  Wählen Sie aus, was als Kennung für die IPv6-Schnittstelle verwendet wird. Folgende Möglichkeiten gibt es:
  - DUID
    DHCP Unique Identifier
  - Systemnamen
    Systemnamen des Geräts
  - Management IP
    Management IPv6-Adresse des Geräts
  - Benutzerdefinert
    Benutzerdefinerter Eintrag
- **Router-Index-Suffix**
  Bei aktiviertem Optionskästchen wird das Router-Index-Suffix an die Remote-ID angehängt.
- **VLAN-ID-Suffix**
  Bei aktiviertem Optionskästchen wird das VLAN-ID-Suffix an die Remote-ID angehängt.
- **Empfänger-Port-Suffix**
  Bei aktiviertem Optionskästchen wird das Empfänger-Port-Suffix an die Remote-ID angehängt.
- **DHCP Unique Identifier**
  Geben Sie den DUID (DHCP Unique Identifier) ein.
- **Benutzerdefinert**
  Geben Sie einen benutzerdefinierten Eintrag als Kennung.

Server-Adressen

Auf dieser Seite aktivieren Sie den DHCPv6 Relay Agent an der Schnittstelle. Zusätzlich legen Sie fest, welcher DHCPv6-Server darüber erreichbar ist.

Beschreibung der angezeigten Felder

- **Relay-Schnittstelle**
  Wählen Sie die Schnittstelle aus, an der der DHCPv6-Relay-Agent die DHCPv6-Anfragen entgegennimmt.
- **Server-IP-Adresse**
  Geben Sie die IPv6-Adresse des DHCPv6-Servers ein, an den die DHCPv6-Anfragen weitergeleitet werden. Die Adresse kann entweder die Unicast- oder die link-lokale Multicast-Adresse sein. Bei der link-lokalen Multicast-Adresse müssen Sie eine Schnittstelle angeben, über die der DHCPv6-Server erreichbar ist. Die Schnittstelle konfigurieren Sie unter "Layer 3 (IPv6) > DHCPv6 Relay Agent > Ausgehende Schnittstellen".

Die Tabelle gliedert sich in folgende Spalten:

- **Server-IP-Adresse**
  Zeigt die IPv6-Adresse des DHCPv6-Servers an.

Ausgehende Schnittstellen

Ausgehende Schnittstellen

Die link-lokale Multicast-Adresse (ff02::1:2) des DHCPv6-Servers ist nur von Geräten erreichbar, die auf dem selben Link sind. Auf dieser Seite legen Sie die IPv6-Schnittstelle fest, über die dieser DHCPv6-Server erreichbar ist.

Beschreibung der angezeigten Felder

- **Relay-Schnittstelle**
  Wählen Sie die IPv6Schnittstelle aus, an der der DHCPv6-Relay-Agent die DHCPv6-Anfragen entgegennimmt.
- **Server-IP-Adresse**
  Wählen Sie die IPv6-Adresse des DHCPv6-Servers ein.
- **Ausgehende Schnittstelle**
  Wählen Sie die Schnittstelle aus, über die der DHCPv6-Server zu erreichen ist.

Die Tabelle gliedert sich in folgende Spalten:

- **Server-IP-Adresse**
  Zeigt die IPv6-Adresse des DHCPv6-Servers an.
- **Ausgehende Schnittstelle**
  Zeigt die IPv6-Schnittstelle an, über die der DHCPv6-Server zu erreichen ist

VRRPv3

Router (IPv6)

Einleitung

Auf dieser Seite legen Sie neue virtuelle Router an. Maximal sind 52 virtuelle Router konfigurierbar. Weitere Parameter konfigurieren Sie unter "Layer 3 (IPv6) > VRRPv3 > Konfiguration".

Hinweis

- Diese Funktion ist nur bei Layer 3 verfügbar.
- Der gleichzeitige Betrieb von VRRP und VRRPv3 ist nicht möglich.
- Aktivieren Sie das Optionskästchen "VRRPv3", um VRRPv3 zu konfigurieren.
- Sie können VRRPv3 an VLAN-Schnittstellen nutzen. Router-Ports werden nicht unterstützt.

Beschreibung der angezeigeten Felder

- VRRPv3
  Aktivieren oder deaktivieren Sie das Routing über VRRPv3. Nur aktivierbar, wenn unter "Layer 3 (IPv6) > Konfiguration" "IPv6-Routing" aktiviert ist.
- VRID-Tracking
  Aktivieren oder deaktivieren Sie das VRID-Tracking.
  Wenn aktiviert, werden alle Schnittstellen einer VRID überwacht. Wenn sich der Link einer Schnittstelle von "up" nach "down" ändert, wird die Priorität aller VRRP-Schnittstelle mit der gleichen VRID auf den Wert "0" verringert.
  Wenn sich der Link einer Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstellen wiederhergestellt.
- Schnittstelle
  Wählen Sie die gewünschte VLAN-Schnittstelle aus, die als virtueller Router fungiert.
- VRID
  Geben Sie die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden. Gültige Werte sind 1 ... 255.

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstelle
  Zeigt die Schnittstelle an, die als virtueller Router fungiert.
- VRID
  Zeigt die ID des virtuellen Routers an.
- Virtuelle MAC-Adresse (Nur Online verfügbar)
  Zeigt die virtuelle MAC-Adresse des virtuellen Routers an.
- Primäre IP-Adresse
  Zeigt die numerisch kleinste IPv6-Adresse auf diesem VLAN an. Der Eintrag "::" bedeutet, dass die "Primäre" Adresse auf diesem VLAN verwendet wird. Ansonsten sind alle IPv6-Adressen, die auf diesem VLAN in dem Menü "Layer 3 (IPv6) >Subnetze" konfiguriert wurden, gültige Werte.
- Router-Status (Nur Online verfügbar)
  Zeigt den derzeitigen Zustand des virtuellen Routers an. Mögliche Werte sind:
  - Master
    Der Router ist der Master-Router und übernimmt die Routingfunktionalität für alle zugeordneten IPv4-Adressen.
  - Backup
    Der Router ist der Backup-Router. Wenn der Master-Router ausfällt, übernimmt der Backup-Router die Aufgaben des Master-Routers.
  - Initialize
    Der virtuelle Router wurde soeben eingeschaltet. In kurzer Zeit wird er in den Zustand "Master" oder "Backup" wechseln.
- Master IP-Adresse (Nur Online verfügbar)
  Zeigt die IPv6-Adresse des Master-Routers an.

- **Priorität** (Nur Online verfügbar)
  Zeigt die Priorität des virtuellen Routers an. Gültige Werte sind 1 ... 254.
  Die 255 erhält automatisch der aktuelle Master-Router. Alle anderen Prioritäten können frei auf die VRRPv3-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRPv3-Router zum "Master".
- **Advert.-Intervall**
  Zeigt das Intervall an, in dem der Master-Router VRRPv3-Pakete verschickt.
- **Unterbrechen**
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - yes
    Dieser Router ist beim Rollenwechsel bevorzugt.
  - no
    Dieser Router ist beim Rollenwechsel nicht bevorzugt.
- **Accept Mode**
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - yes
    Es antworten auch virtuelle IP-Adressen auf Ping-Anfragen.
  - no
    Virtuelle IP-Adressen antworten nicht auf Ping-Anfragen.

VRRP und DHCP-Server

Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, muss der DHCP-Server auf dem Master-Router konfiguriert sein. Backup-Router reagieren nicht auf DHCP-Anfragen. Stellen Sie sicher, dass der Master-Router statisch konfiguriert ist und nach einem Ausfall wieder zum Master der VRRP-Gruppe wird.

Konfiguration (IPv6)

Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **Schnittstelle / VRID**
  Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Primäre Adresse**
  Wählen Sie die vorrangige IPv6-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IPv4-Adresse.

  Hinweis

  Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist dann **::**.
  Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IPv6-Adresse als Quelladresse für VRRPv3-Pakete genutzt wird, wählen Sie die entsprechende IPv6-Adresse aus. Ansonsten wird die vorrangige IPv6-Adresse verwendet.

- **Master**
  Wenn diese Option aktiviert ist, wird die numerisch kleinste IPv6-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die vorrangige IPv6-Adresse des VRRP-Routers als virtuelle IPv6-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IPv6-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**
  Gegen Sie die Priorität dieses virtuellen Routers ein. Gültige Werte sind 1 ... 254. Den Wert 255 erhält immer der aktuelle Master-Router. Alle anderen Prioritäten können frei auf die redundanten Router verteilt werden. Je größer eine Priorität ist, desto eher wird der Router zum "Master".
- **Advertisement-Intervall**
  Geben Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRPv3-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**
  Erlauben sie die Bevorzugung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.
- **Antworte an virtuellen Schnittstellen auf Ping-Anfragen (Accept Mode)**
  Wenn aktiviert, antworten auch die virtuellen IPv6-Adressen auf den Ping.
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Priorität verringern**
  Geben Sie ein, um welchen Wert die Priorität der VRRP-Schnittstelle verringert wird.
- **Aktuelle Priorität**
  Zeigt die Priorität der VRRP-Schnittstelle an, nachdem die überwachte Schnittstelle in den Zustand "down" gewechselt hat.

Adressübersicht (IPv6)

Übersicht

Diese Seite zeigt an, welche IPv6-Adressen der virtuelle Router überwacht. Jeder virtueller Router kann maximal 10 IPv6-Adressen überwachen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, der als virtueller Router fungiert.
- **VRID**
  Zeigt die ID dieses virtuellen Routers an.
- **Anzahl der Adressen**
  Zeigt die Anzahl der IPv6-Adressen an.
- **Zugeordnete IP-Adresse (1) ... Zugeordnete IP-Adresse (10)**
  Zeigt die Router IPv6-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IPv6-Adressen durch diesen Router übernommen.

Adresskonfiguration (IPv6)

Anlegen oder Ändern der überwachten IP-Adressen

Auf dieser Seite können Sie die zu überwachenden IPv4-Adressen anlegen, ändern oder löschen. Maximal 10 IPv4-Adressen lassen sich von einem virtuellen Router überwachen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **Schnittstelle / VRID**
  Wählen Sie die ID des virtuellen Routers aus.
- **Zugeordnete IP-Adresse**
  Geben Sie die IPv6-Adresse ein, die der virtuelle Router überwachen soll. Maximal 10 IPv6-Adressen sind möglich.

Die Tabelle gliedert sich in folgende Spalten:

- **Zugeordnete IP-Adresse**
  Zeigt die IPv6-Adressen an, die der virtuelle Router überwacht.

Schnittstellenüberwachung (IPv6)

Einleitung

Auf dieser Seite konfigurieren Sie die Überwachung von Schnittstellen.

Wenn sich der Link einer überwachten Schnittstelle von "up" nach "down" ändert, wird die Priorität der zugeordneten VRRP-Schnittstelle verringert. Um welchen Wert die Priorität verringt wird, konfigurieren Sie auf der Seite "Layer 3 > VRRP/VRRPv3 > Konfiguration".

Wenn sich der Link der Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstelle wiederhergestellt.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie aus der Klappliste die Schnittstelle aus, die überwacht werden soll.
- **Track-ID**
  Geben Sie eine Track-ID ein.
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Track-Schnittstellenzähler**
  Geben Sie ein, wie viele der überwachten Schnittstellen in den Zustand "down" wechseln müssen, bevor die Priorität geändert wird.

Die Tabelle gliedert sich in folgende Spalten:

- **Track-ID**
  Zeigt die Track-ID an.
- **Schnittstelle**
  Zeigt die Schnittstelle an, die überwacht wird.

OSPFv3

Konfiguration

Einleitung .

Auf dieser Seite konfigurieren Sie das Routing über OSPFv3.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **OSPFv3**
  Aktivieren oder deaktivieren Sie das Routing über OSPFv3.
- **Router-ID**
  Geben Sie eine Bezeichnung einer der OSPFv3-Schnittstellen ein. Die Router-ID wird im IPv4-Adressformat angegeben und muss im Netzwerk eindeutig sein.
- **Border-Router**
  Zeigt den Status des OSPFv3-Routers an. Ist das lokale System aktives Mitglied in mindestens 2 Bereichen, ist es ein Area Border Router.
- **AS Border Router**
  Legen Sie fest, ob der Router ein AS Border-Router ist. Ein AS Border-Router vermittelt zwischen mehreren autonomen Systemen, z. B. wenn Sie ein zusätzliches RIP-Netzwerk haben. Ein AS Border-Router ist auch nötig, um statische Routen hinzuzufügen und zu verteilen.
- **Neues LSA empfangen**
  Zeigt die Anzahl der LSAs an, die empfangen wurden. Aktualisierungen und eigene LSAs werden nicht gezählt.
- **Neues LSA konfiguriert**
  Zeigt die Anzahl der unterschiedlichen LSAs an, die von diesem lokalen System versendet wurden.
- **Maximale Anzahl externer LSAs**
  Um die Einträge von externen LSAs in der Datenbank zu begrenzen, tragen Sie die maximale Anzahl der External LSAs ein.
- **Exit-Intervall**
  Geben Sie das Intervall ein, nach dem der OSPFv3-Router wieder versucht aus dem Overflow-Status herauszukommen. Eine 0 bedeutet, dass der OSPFv3-Router erst nach einem Hochlauf wieder versucht, den Overflow Status zu verlassen.
- **Inbound-Filter**
  Wählen Sie eine Route Map aus, die ankommende Routen filtert.

- **Weiterverteilung von Routen**
  Legen Sie fest, welche bekannten Routen über OSPFv3 weitergegeben werden. Sie können unterschiedliche Entscheidungen für die Routearten Connected, Static und RIPng treffen.

  Hinweis

  Die Optionen lassen sich nur an einem AS Border Router aktivieren. Besonders die Aktivierung der Optionen Default und Static kann zu Problemen führen, wenn sie an zu vielen Stellen im Netz aktiviert werden, z. B. Forwarding Loops.

- **Route Map**
  Wählen Sie eine Route Map aus, die filtert, welche Routen über RIPng weitergegeben werden.
- **Protokollspezifische Einstellungen**
  - **Default-Distanz**
    Legen Sie die administrative Distanz für das Protokoll OSPFv2 fest.
- **Route Map-spezifische Einstellungen**
  - **Route Map-Distanz**
    Wählen Sie aus der Klappliste eine Route Map aus, für die Sie die administrative Distanz festlegen wollen.
  - **Distanz**
    Konfigurieren Sie die administrative Distanz für die zuvor ausgewählre Route Map.

Areas

Übersicht

Ein autonomes Netzwerk (Autonomous System) kann in kleinere Bereiche (Areas) unterteilt werden.

Auf dieser Seite können Sie die Bereiche des Routers ansehen, anlegen, ändern oder löschen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Area-ID**
  Geben Sie die Bezeichnung des Bereichs ein. Für alle Router eines Bereichs wird die Datenbank synchronisiert. Die Bereichsbezeichnung muss im Netzwerk eindeutig sein. Die Bereichsbezeichnung ist eine 32-Bit-Zahl in folgendem Eingabeformat: x.x.x.x mit x = 0 ... 255
  Die Bereichsbezeichnung 0.0.0.0 ist für den Backbone-Bereich reserviert und kann nicht gelöscht werden.

Die Tabelle enthält folgende Spalten:

- **Area-ID**
  Zeigt die ID des Bereichs an.
- **Area-Typ**
  Legen Sie den Typ des Bereichs fest.
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Summary**
  Legen Sie fest, ob für diesen Bereich Summary LSAs erzeugt werden.
  - Summary: Summary LSAs werden erzeugt und in den Bereich verschickt.
  - No Summary: Summary LSAs werden nicht erzeugt und in den Bereich verschickt.
- **Metrik**
  Zeigt die Kosten für die OSPFv3-Schnittstelle an.
- **Aktualisierungen**
  Zeigt die Anzahl der Neuberechnungen der IPv6 Routing-Tabellen an.
- **LSA-Zähler**
  Zeigt die Anzahl der LSA in der Datenbank an.
- **Area BR**
  Zeigt die Anzahl der erreichbaren Area Border Router (ABR) innerhalb dieses Bereichs an.
- **AS BR**
  Zeigt die Anzahl der erreichbaren Autonomous System Border Router (ASBR) in dieser Area an.

Area-Bereich

Anlegen eines neuen OSPFv3 Area-Bereich

Auf dieser Seite können Sie bis zu vier Netzwerke unter einer Area-ID zusammengefassen. Das Verfahren wird nur bei Area Border Routern angewandt. Damit propagiert ein Area Border-Router nach außen lediglich eine Route für zusammengefasste Adressbereiche.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **Area-ID**
  Wählen Sie die ID des Bereichs. Die ID legen Sie bei "Areas" fest.
- **Präfix**
  Geben Sie die IPv6-Adresse des Netzwerks ein, das zusammengefasst wird.
- **Präfixlänge**
  Geben Sie die Anzahl der Bits ein, die dem Präfix angehören, das zusammengefasst wird.

Die Tabelle enthält folgende Spalten:

- **Area-ID**
  Zeigt die ID des Bereichs an.
- **Präfix**
  Zeigt die IPv6-Adresse des Netzwerks an, das zusammengefasst wird.
- **Präfixlänge**
  Zeigt die die Anzahl der Bits an, die dem Präfix angehören wird.
- **Advertise**
  Aktivieren Sie diese Option, um das zusammengefasste Netzwerk zu veröffentlichen.

Schnittstellen

Übersicht

Auf dieser Seite können Sie OSPFv3-Schnittstellen konfigurieren.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **Schnittstelle**
  Wählen Sie die IPv6-Schnittstelle aus, an der OSPFv3 aktiviert wird.
- **Area-ID**
  Wählen Sie die ID des Bereichs aus, mit der die OSPFv3-Schnittstelle verbunden ist.

  Hinweis

  Wählen Sie für sekundäre Schnittstelle die gleiche Area-ID aus wie für die dazugehörige primäre Schnittstelle aus.
  Die Information, ob es sich um eine primäre oder sekundäre Schnittstelle handelt, finden Sie unter "Layer 3 (IPv6)" > "Subnetze".

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die OSPFv3-Schnittstelle an.
- **Area-ID**
  Wählen Sie die ID des Bereichs aus, mit der die OSPFv3-Schnittstelle verbunden ist.
- **Passive Schnittstelle**
  Legen Sie fest, ob OPSFv3 an der Schnittstelle aktiv ist.
  - Aktiviert:
    Über diese Schnittstelle werden keine OSPFv3-Informationen versendet (z. B. Hello-Telegramme und LSDB-Updates) und gelernt.
  - Deaktiviert:
    Über diese Schnittstelle werden OSPFv3-Informationen versendet und gelernt.
- **Metrik**
  Tragen Sie die Kosten für die OSPFv3-Schnittstelle ein.
- **Priorität**
  Geben Sie die Router-Priorität ein. Die Priorität spielt nur bei der Auswahl des designierten Routers oder designierten Border Routers eine Rolle. Dieser Parameter kann auf Routern des gleichen Subnetzes unterschiedlich gewählt werden.
  Wertebereich: 0 bis 255
  Defaultwert: 1.
- **Trans.-Delay**
  Geben Sie die gewünschte Verzögerung beim Versenden einer Verbindungsaktualisierung ein.
  Wertebereich: 1 s bis 3600 s
  Default-Einstellung: 1 s
- **Retrans.-Intervall**
  Geben Sie die Zeit ein, nach der ein OSPFv3-Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
  Wertebereich: 1s bis 3600 s
  Default-Einstellung: 5 s
- **Hello-Intervall**
  Geben Sie den Abstand zwischen zwei Hello-Paketen ein.
  Wertebereich: 1s bis 65.535 s
  Default-Einstellung: 10 s
- **Dead-Intervall**
  Geben Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" markiert wird, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.
  Default-Einstellung: 60 s

Virtuelle Links

Übersicht

Jeder Area Border Router muss aus protokolltechnischen Gründen Zugang zum Backbone-Bereich erhalten. Wenn ein Router nicht direkt am Backbone-Bereich angeschlossen ist, wird eine virtuelle Verbindung dorthin angelegt.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Hinweis

Beachten Sie, dass beim Anlegen einer virtuellen Verbindung sowohl der Transit-Bereich als auch der Backbone-Bereich bereits konfiguriert sein müssen.

Eine virtuelle Verbindung muss auf beiden Seiten gleich konfiguriert werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Nachbar-Router-ID**
  Geben Sie die ID des Nachbar-Routers ein, der auf der anderen Seite der virtuellen Verbindung ist.
- **Transit Area-ID**
  Wählen Sie die ID des Bereichs, der die beiden Router verbindet.

Die Tabelle enthält folgende Spalten:

- **Transit Area ID**
  Zeigt die ID an, über die die beiden Router verbunden sind.
- **Nachbar-Router-ID**
  Zeigt die ID des Nachbar-Routers auf der anderen Seite der virtuellen Verbindung an.
- **Virt. Link Status**
  Legen Sie fest, in welchem Zustand sich die virtuelle Verbindung befindet. Folgende Zustände sind möglich:
  - down: Die virtuelle Verbindung ist inaktiv.
  - point-to-point: Die virtuelle Verbindung ist aktiv.
- **Trans. Delay**
  Geben Sie die gewünschte Verzögerung bei dem Versenden einer Verbindungsaktualisierung ein.
  Wertebereich: 1 s bis 3600 s
  Default-Einstellung: 1 s

- **Retrans.-Intervall**
  Geben Sie die Zeit ein, nach der ein Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
  Wertebereich: 1s bis 3600 s
  Default-Einstellung: 5 s
- **Hello-Intervall**
  Geben Sie den Abstand wischen zwei Hello-Paketen ein.
  Wertebereich: 1s bis 65.535 s
  Default-Einstellung: 10 s
- **Dead-Intervall**
  Geben Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" gilt, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.
  Default-Einstellung: 60 s

RIPng

Konfiguration

Auf dieser Seite konfigurieren Sie das Routing über RIPng.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

- **Inbound-Filter**
  Wählen Sie eine Route Map aus, die ankommende Routen filtert.
- **Weiterverteilung von Routen**
  Legen Sie fest, welche bekannten Routen über RIPng weitergegeben werden.
  Folgende Routenarten gibt es:
  - Connected
  - Static
  - OSPFv3
- **Route Map**
  Wählen Sie eine Route Map aus, die filtert, welche Routen über RIPng weitergegeben werden.

- Protokollspezifische Einstellungen
  - Default-Distanz
    Legen Sie die administrative Distanz für das Protokoll OSPFv2 fest.
- Route Map-spezifische Einstellungen
  - Route Map-Distanz
    Wählen Sie aus der Klappliste eine Route Map aus, für die Sie die administrative Distanz festlegen wollen.
  - Distanz
    Konfigurieren Sie die administrative Distanz für die zuvor ausgewählre Route Map

Schnittstellen

Auf dieser Seite legen Sie die Schnittstellen fest, die RIPng unterstützen.

Hinweis

Diese Funktion ist nur bei Layer 3 verfügbar.

Beschreibung der angezeigten Felder

Die Tabelle enthält folgende Spalten:

- Schnittstelle
  Zeigt die verfügbaren IPv6-Schnittstelle an.
- Admin-Status
  Aktivieren oder deaktivieren Sie RIPng-Routing für die Schnittstelle.
- Standard-Metrik
  Geben Sie die Kosten für die RIPng-Schnittstelle ein.

Security-Funktionen projektieren

Benutzerverwaltung (WBM, CLI)

Übersicht zur Benutzerverwaltung

Der Zugriff auf das Gerät wird durch konfigurierbare Benutzereinstellungen verwaltet. Richten Sie Benutzer mit jeweils einem Passwort zur Authentifizierung ein. Weisen Sie den Benutzern eine Rolle mit entsprechenden Rechten zu.

Die Authentifizierung von Benutzern kann entweder vom Gerät lokal oder von einem externen RADIUS-Server durchgeführt werden. Wie die Authentifizierung erfolgen soll, konfigurieren Sie auf der Seite "Security > AAA > Allgemein".

Hinweis

Wenn Sie die Konfiguration eines Geräts nach STEP 7 (TIA Portal) übertragen, werden die konfigurierten Benutzer, Rollen und Gruppen nicht übertragen.

Kompatibilität mit Vorgängerversionen

Mit der Firmware-Version 5.1 wurde die Benutzerverwaltung um den RADIUS-Autorisierungsmodus "Herstellerspezifisch" erweitert. Um die Kompatibilität mit Firmware-Versionen ≤ 5.0 zu gewährleisten, wurde die Default-Einstellung so gewählt, dass nach einem Firmware-Update weiterhin der frühere Autorisierungsmodus "Standard" verwendet wird.

Lokale Anmeldung

Die lokale Anmeldung von Benutzern durch das Gerät läuft wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort am Gerät an.
2. Das Gerät prüft, ob ein Eintrag für den Benutzer vorhanden ist:
   → Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet.
   → Wenn kein entsprechender Eintrag existiert, wird dem Benutzer der Zugriff verweigert.

Anmeldung über einen externen RADIUS-Server

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Auhentifizierung und Autorisierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können.

Anhängig davon, welchen RADIUS-Autorisierungsmodus Sie auf der Seite "Security > AAA > RADIUS-Client" eingestellt haben, wertet das Gerät unterschiedliche Informationen des RADIUS-Servers aus.

RADIUS-Autorisierungsmodus "Standard"

Wenn Sie den RADIUS-Autorisierungsmodus "Standard" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort am Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück:
     → Der Benutzer wird mit Lese-/Schreibrechten angemeldet.
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt einen anderen oder gar keinen Wert für das Attribut "Service Type" an das Gerät zurück:
     → Der Benutzer wird mit Leserechten angemeldet.
   - Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
     → Dem Benutzer wird der Zugriff verweigert.

RADIUS-Autorisierungsmodus "Herstellerspezifisch"

Voraussetzung

Für den RADIUS-Autorisierungsmodus "Herstellerspezifisch" ist am RADIUS-Server Folgendes einzustellen:

- Herstellercode: 4196
- Attributnummer: 1
- Attributformat: Zeichenfolge (Gruppenname)

Vorgehen

Wenn Sie den RADIUS-Autorisierungsmodus "Herstellerspezifisch" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   Fall A: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt die dem Benutzer zugeordnete Gruppe an das Gerät zurück.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird mit den Rechten der zugeordneten Gruppe angemeldet.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird der Rolle mit den größeren Rechten zugeordnet und mit diesen Rechten angemeldet.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle angemeldet, die mit seinem Benutzeraccout verknüpft ist.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall B: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung, gibt jedoch keine Gruppe an das Gerät zurück:
   - Der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der verknüpften Rolle angemeldet.
   - Der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall C: Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
   - Dem Benutzer wird der Zugriff verweigert.

Zuweisung eines VLANs über RADIUS oder Guest VLAN

Authentifizierung mit Änderung der VLAN-Konfiguration

Wenn ein Port bei der Authentifizierung über die Funktion "VLAN-Zuordnung von RADIUS übernehmen" oder "Guest VLAN" dynamisch einem VLAN zugewiesen wird, gibt es folgende Optionen:

- Wenn das VLAN, das zugewiesen werden soll, nicht auf dem Gerät angelegt ist, wird die Authentifizierung abgelehnt.
- Wenn das VLAN, das zugewiesen werden soll, auf dem Gerät angelegt ist:
  - Der Port wird Untagged Member in dem zugewiesenen VLAN, wenn er es bisher nicht war.
    → Dadurch wird ggf. die statische Konfiguration des Ports in diesem VLAN überschrieben und nicht wiederhergestellt, wenn die Authentifizierung zurückgenommen wird.
  - Die Port-VID des Ports wird auf die ID des zugewiesenen VLANs geändert.

Hinweis

Wenn der Port nur einem VLAN zugeordnet sein soll, müssen Sie die VLAN-Konfiguration manuell anpassen. Alle Ports sind z. B. standardmäßig Untagged Member in "VLAN 1".

Wenn die Authentifizierung zurückgenommen wird, z. B. durch einen Link-Down, werden die dynamischen Änderungen zurückgenommen:

- Der Port ist kein Member mehr in dem zugewiesenen VLAN.
- Die Port-VID des Ports wird auf den Wert zurückgesetzt, den sie vor der Authentifizierung hatte.

Hinweis

Wenn die Port-VID des Ports vor der Authentifizierung der zugewiesenen Port-VID entspricht, bleibt der Port Untagged Member in diesem VLAN.

Authentifizierung ohne Änderung der VLAN-Konfiguration

Wenn bei der Authentifizierung weder durch die Funktion "VLAN-Zuordnung von RADIUS übernehmen" noch durch "Guest VLAN" ein VLAN zugewiesen wird, bleibt die bestehende VLAN-Konfiguration des Ports unverändert.

Benutzer

Lokale Benutzer

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Lokale Benutzer

Auf dieser Seite erstellen Sie lokale Benutzer mit den entsprechenden Rechten.

Wenn Sie einen lokalen Benutzer anlegen oder löschen, wird diese Änderung automatisch auch in der Tabelle "Externe Benutzerkonten" durchgeführt. Wenn Sie eine Änderung explizit für die interne oder externe Benutzertabelle durchführen wollen, nutzen Sie die CLI-Befehle.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden

Beschreibung der angezeigten Felder

- **Benutzerkonto**
  Geben Sie den Namen für den Benutzer ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 255 Zeichen lang sein.
  - Er darf folgende Zeichen nicht enthalten: | ? " ; :
    Die Zeichen für Space und Delete dürfen auch nicht enthalten sein.

  Hinweis

  Benutzername nicht änderbar

  Nach dem Anlegen eines Benutzers kann der Benutzername nicht mehr geändert werden.

  Wenn ein Benutzername geändert werden soll, dann muss der Benutzer gelöscht und ein neuer Benutzer angelegt werden.

  Hinweis

  Werkseitig voreingestellter Benutzer "user"

  Ab der Firmware-Version 6.0 ist der werkseitig voreingestellte Benutzer "user" im Auslieferzustand nicht mehr verfügbar. Wenn Sie ein Gerät auf die Firmware V6.0 aktualisieren, ist der werkseitig voreingestellte Benutzer "user" zunächst noch verfügbar. Wenn Sie das Gerät auf die Werkseinstellungen zurücksetzen ("Auf Werkseinstellungen zurücksetzen und Neustart"), wird der werkseitig voreingestellten Benutzer "user" gelöscht. Sie können neue Benutzer mit der Rolle "user" anlegen.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie verwendet wird:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

  Sie konfigurieren die Passwortrichtlinie auf der Seite "Security > Passwörter > Optionen".

- **Passwort**
  Geben Sie das Passwort für den Benutzer ein. Die Stärke des Passworts ist abhängig von der eingestellten Passwortrichlinie.
- **Passwort bestätigen**
  Geben Sie das Passwort erneut ein, um es zu bestätigen.
- **Rolle**
  Wählen Sie eine Rolle aus.
  Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".

Die Tabelle enthält folgende Spalten:

Hinweis

Die voreingestellten Benutzer sowie angemeldete Benutzer können nicht gelöscht oder geändert werden.

- **Benutzeraccount**
  Zeigt den Benutzernamen an.
- **Rolle**
  Zeigt die Rolle des Benutzers an.
- **Beschreibung**
  Zeigt eine Beschreibung des Benutzekontos an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Rollen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Rollen

Auf dieser Seite erstellen Sie Rollen, die lokal auf dem Gerät gültig sind.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Rollenname**
  Geben Sie den Namen für die Rolle ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

  Hinweis
  Rollenname nicht änderbar

  Nach dem Anlegen einer Rolle kann der Rollenname nicht mehr geändert werden.

  Wenn ein Rollenname geändert werden soll, dann muss die Rolle gelöscht und eine neue Rolle angelegt werden.

Die Tabelle enthält folgende Spalten:

Hinweis
Rollen löschen

Die voreingestellten Rollen sowie zugewiesene Rollen können nicht gelöscht oder geändert werden

- **Rolle**
  Zeigt den Namen der Rolle an.
- **Funktionsrecht**
  Wählen Sie die Funktionsrechte der Rolle aus:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen aber nicht verändern.
    Benutzer mit dieser Rolle können ihr eigenes Passwort ändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.

  Hinweis
  Funktionsrecht nicht änderbar

  Wenn Sie eine Rolle zugewiesen haben, können Sie das Funktionsrecht der Rolle nicht mehr ändern.

  Wenn Sie das Funktionsrecht einer Rolle ändern wollen, gehen Sie wie folgt vor:
  1. Löschen Sie alle zugewiesenen Benutzer.
  2. Ändern Sie das Funktionsrecht der Rolle.
  3. Weisen Sie die Rolle erneut zu.

- **Beschreibung**
  Geben Sie eine Beschreibung für die Rolle ein. Bei vordefinierten Rollen wird eine Beschreibung angezeigt. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Gruppen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Benutzergruppen

Auf dieser Seite verknüpfen Sie eine Gruppe mit einer Rolle.

In diesem Beispiel wird die Gruppe "Administrators" mit der Rolle "admin" verknüpft. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert und der Gruppe "Administrators" zuordnet, erhält dieser Benutzer auf dem Gerät die Rechte der Rolle "admin".

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Der Name muss der Gruppe auf dem RADIUS-Server entsprechen. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

Die Tabelle enthält folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an.
- **Rolle**
  Wählen Sie eine Rolle aus. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät. Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".
- **Beschreibung**
  Geben Sie eine Beschreibung für die die Verknüpfung der Gruppe mit einer Rolle an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Passwörter

Passwörter

Konfiguration der Geräte-Passwörter

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Hinweis

Wenn Sie über einen RADIUS-Server angemeldet sind, können Sie keine lokalen Gerätepasswörter ändern.

Auf dieser Seite können Sie Passwörter ändern. Wenn Sie mit der Rolle "admin" angemeldet sind, können Sie die Passwörter für alle Benutzeraccounts ändern. Wenn Sie mit der Rolle "user" angemeldet sind, können Sie nur ihr eigenes Passwort ändern.

Hinweis

Wenn Sie sich das erste Mal oder nach einem "Auf Werkseinstellungen zurücksetzen und Neustart" mit dem voreingestellten Benutzer "admin" anmelden, werden Sie aufgefordert, das Passwort zu ändern.

Werksseitig sind die Passwörter bei Auslieferung des Geräts wie folgt eingestellt:

- admin: admin

Hinweis

Passwort ändern im Trial-Modus

Auch wenn Sie im Konfigurationsmodus "Trial" das Passwort ändern, wird diese Änderung sofort gespeichert.

Beschreibung der angezeigten Felder

- **Aktueller Benutzer**
  Zeigt den Benutzer an, der aktuell angemeldet ist.
- **Aktuelles Benutzerpasswort**
  Geben Sie das Passwort des aktuell angemeldeten Benutzers ein.
- **Benutzerkonto**
  Wählen Sie den Benutzer, dessen Passwort Sie ändern möchten.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie bei der Vergabe von neuen Passwörter verwendet wird.
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen
- **Neuer Name für das Admin-Benutzerkonto**
  Sie können einmalig den Namen des werksseitig voreingestellten Benutzers "admin" ändern. Danach ist ein Umbenennen von "admin" nicht mehr möglich.
- **Neues Passwort**
  Geben Sie das neue Passwort für den ausgewählten Benutzer ein. Es darf folgende Zeichen nicht enthalten:
  – § ? " ; : –
  Das Leerzeichen und das Zeichen für "Entfernen" düfen auch nicht enthalten sein.
- **Passwort bestätigen**
  Geben Sie das neue Passwort erneut ein, um es zu bestätigen.

Optionen

Auf dieser Seite legen Sie fest, welche Passwortrichtlinie bei der Vergabe von neuen Passwörter beachtet wird.

Beschreibung

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie akutell verwendet wird.
- **Neue Passwortrichtlinie**
  Wählen Sie aus der Klappliste die gewünschte Einstellung aus:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

AAA

Allgemein

Anmeldung von Netzteilnehmern

Die im Menü verwendete Bezeichnung "AAA" steht dabei für "Authentication, Authorization, Accounting". Dieses Feature dient dazu, Netzteilnehmer zu identifizieren und zuzulassen und ihnen die entsprechenden Dienste bereitzustellen.

Auf dieser Seite konfigurieren Sie die Anmeldung.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

Hinweis

Um den Login-Authentifizierung "RADIUS", "Lokal und RADIUS" oder "RADIUS und Fallback Lokal" nutzen zu können, muss ein RADIUS-Server hinterlegt und für die Benutzerauthentifizierung konfiguriert sein.

- **Login-Authentifizierung**
  Legen Sie fest, wie die Anmeldung erfolgt:
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen
  - Local und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen. Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS und Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen. Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.

RADIUS-Client

Authentifizierung über einen externen Server

Das Konzept von RADIUS basiert auf einem externen Authentifizierungsserver.

Jede Zeile der Tabelle enthält die Zugangsdaten für je einen Server. In der Suchreihenfolge wird der primäre Server zuerst angefragt. Ist der primäre Server nicht erreichbar, werden in der eingetragenen Reihenfolge sekundäre Server angefragt. Wenn keiner der Server antwortet, findet keine Authentifizierung statt.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **RADIUS-Autorisierungsmodus**
  Der RADIUS-Autorisierungsmodus legt bei der Login-Authentifizierung fest, wie bei einer erfolgreichen Authentifizierung (Seite 2123) die Rechtevergabe für die Benutzer erfolgt.
  - Standard
    In diesem Modus wird der Benutzer mit Administratorrechten angemeldet, wenn der Server für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück gibt. In allen anderen Fällen wird der Benutzer mit Leserechten angemeldet.
  - Herstellerspezifisch
    In diesem Modus ist die Rechtevergabe davon abhängig, ob und welche Gruppe der Server für den Benutzer zurück gibt und ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzeraccounts" gibt.

Die Tabelle gliedert sich in folgende Spalten:

- **Auth.-Servertyp**
  Wählen Sie aus, für welche Authentifizierungsverfahren der Server verwendet werden soll.
  - Login
    Der Server wird nur für die Login-Authentifizierung verwendet.
  - 802.1X
    Der Server wird nur für die 802.1X-Authentifizierung verwendet.
  - Login & 802.1X
    Der Server wird für beide Authentifizierungsverfahren verwendet.
- **Adresse des RADIUS-Servers**
  Tragen Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des RADIUS Servers ein.
- **Server-Port**
  Tragen Sie den Eingangs-Port auf dem RADIUS-Server ein. Standardmäßig ist der Eingangs-Port 1812 eingestellt. Der Wertebereich ist 1 ... 65535.
- **Shared Secret**
  Tragen Sie Ihre Zugangskennung ein. Der Wertebereich ist 1 ... 128 Zeichen.
- **Shared Secret bestätigen**
  Tragen Sie die Zugangskennung zur Bestätigung erneut ein.
- **Max. Retrans**
  Geben Sie hier die maximale Anzahl der Wiederholungen eines Anfrageversuchs ein. Der initiale Verbindungsversuch wird um den hier angegebenen Wert wiederholt, bevor ein anderer konfigurierter RADIUS-Server angefragt wird oder die Anmeldung für gescheitert erklärt wird. Standardmäßig sind 3 Wiederholungen eingestellt, das bedeutet 4 Verbindungsversuche. Der Wertebereich ist 1 ... 5.
- **Primärer Server**
  Legen Sie fest, ob dieser Server der primäre Server ist. Sie können aus den Optionen "ja" oder "nein" auswählen.

- **Test**
  Mit dieser Schaltfläche können Sie testen, ob der angegebene RADIUS-Server verfügbar ist oder nicht. Der Test wird einmalig durchgeführt und nicht zyklisch wiederholt.
- **Testergebnis**
  Zeigt an, ob der RADIUS-Server verfügbar ist oder nicht:
  - Fehlgeschlagen, keine Testpakete gesendet
    Die IP-Adresse ist nicht erreichbar.
    Die IP-Adresse ist erreichbar, der RADIUS-Server läuft jedoch nicht.
  - Erreichbar, das Shared Secret wurde nicht akzeptiert
    Die IP-Adresse ist erreichbar, der RADIUS-Server akzeptiert jedoch das angegebene Shared Secret nicht.
  - Erreichbar, das Shared Secret wurde akzeptiert
    Die IP-Adresse ist erreichbar und der RADIUS-Server akzeptiert das angegebene Shared Secret.

  Das Testergebnis wird nicht automatisch aktualisiert. Um das Testergebnis zu löschen, klicken Sie auf die Schaltfläche "Aktualisieren".

802.1x-Authenticator

Netzwerkzugriff einrichten

Für ein Endgerät ist der Zugang zum Netzwerk erst möglich, nachdem das Gerät die Anmeldedaten beim Authentifizierungs-Server verifiziert hat. Die Authentifizierung kann über 802.1X oder die MAC-Adresse erfolgen. Bei der Authentifizierung über 802.1X müssen sowohl das Endgerät als auch der Authentifizierungs-Server das EAP-Protokoll (Extensive Authentication Protocol) unterstützen.

Aktivierung der Authentifizierung für einzelne Ports

Durch Aktivieren der entsprechenden Optionen legen Sie individuell für jeden Port fest, ob der Netzzugriffsschutz nach IEEE 802.1x auf diesem Port aktiviert ist.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **MAC-Authentifizierung**
  Aktivieren oder Deaktivieren Sie die MAC-Authentifizierung für das Gerät.
- **Guest VLAN**
  Aktivieren oder Deaktivieren Sie die Funktion "Guest VLAN" für das Gerät.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **802.1X Auth.-Kontrolle**
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.

- 802.1x Reauthentifizierung
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- MAC-Authentifizierung
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
  MAC Auth. only on Timeout
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- VLAN-Zuordnung von RADIUS übernehmen
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- MAC-Auth. max. zugelassene Adressen
  Geben Sie an, wie viele Endgeräte gleichzeitig an dem Port angeschlossen sein dürfen. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- Guest VLAN
  Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- Guest VLAN-ID
  Geben Sie die VLAN-ID des Ports an. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- Guest VLAN max. zugelassene Adressen
  Geben Sie an, wie viele Endgeräte gleichzeitig im "Guest VLAN" zugelassen sind. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Port
  In dieser Spalte werden alle in diesem Gerät verfügbaren Ports aufgeführt.
- 802.1X Auth.-Kontrolle
  Legen Sie die Authentifizierung des Ports fest:
  - Nicht autorisieren erzwingen
    Der Datenverkehr über den Port ist gesperrt.
  - Autorisieren erzwingen
    Der Datenverkehr über den Port ist ohne Einschränkung erlaubt. Defaulteinstellung
  - Auto
    Endgeräte werden an dem Port mit dem Verfahren "802.1x" authentifiziert. Der Datenverkehr über den Port wird entsprechend des Authentifizierungsergebnisses erlaubt oder gesperrt.
- 802.1x Reauthentifizierung
  Aktivieren Sie diese Option, wenn für ein bereits authentifiziertes Endgerät zyklisch eine Reauthentifizierung durchführt werden soll.

- **MAC-Authentifizierung**
  Aktivieren Sie diese Option, wenn Endgeräte mit dem Verfahren "MAC-Authentifizierung" authentifiziert werden sollen.
  Wenn die "802.1X Auth.-Kontrolle" auf "Auto" konfiguriert ist und die "MAC-Authentifizierung" aktiviert ist, beträgt der Timeout für das Verfahren "802.1X" 5 Sekunden. Wenn für die Authentifizierung mit dem Verfahren "802.1X" an einem Port eine manuelle Eingabe erforderlich ist, reichen die 5 Sekunden nicht aus. Um die Authentifizierung über "802.1X" ausführen zu können, deaktivieren Sie die MAC-Authentifizierung an diesem Port.
- **MAC Auth. only on Timeout**
  Bei aktiviertem Optionskästchen ist eine MAC-Authentifizierung nur nach einem 802.1X-Timeout möglich, nicht jedoch nach einer fehlgeschlagenen 802.1X-Authentifizierung. Bei nicht aktiviertem Optionskästchen ist eine MAC-Authentifizierung sowohl nach einem 802.1X-Timeout als auch nach einer fehlgeschlagenen 802.1X-Authentifizierung möglich.
- **VLAN-Zuordnung von RADIUS übernehmen**
  Der RADIUS-Server übermittelt dem IE-Switch, welchem VLAN der Port angehören soll. Aktivieren Sie diese Option, wenn die Informationen des Servers berücksichtigt werden sollen.
  Der Port kann dem entsprechenden VLAN nur zugeordnet werden, wenn das VLAN auf dem Gerät angelegt ist. Die Authentifizierung (Seite 474) wird sonst abgelehnt. Wenn ein Port bei der Authentifizierung über diese Funktion dynamisch einem VLAN zugewiesen wird, ist die Zuordnung über die VLAN-ID oder den VLAN-Namen möglich. Konfigurieren Sie auf dem RADIUS-Server folgende Werte:
  - Tunnel-Type = VLAN
  - Tunnel-Medium-Type = IEEE-802
  - Tunnel-Private-Group-Id = VLAN-ID oder VLAN-Name

  Der IE-Switch unterscheidet wie folgt:
  - VLAN-ID: Der RADIUS-Server übermittelt für den Parameter "Tunnel-Private-Group-Id" einen numerischen String.
  - VLAN-Name: Der RADIUS-Server übermittelt für den Parameter "Tunnel-Private-Group-Id" einen alphanumerischen String.

- **MAC-Auth. max. zugelassene Adressen**
  - 1 ... 100
    Geben Sie an, wie viele MAC-Adressen gleichzeitig an dem Port kommunizieren dürfen.

    Hinweis

    Wenn ein Gerät mehrere MAC-Adressen verwendet, müssen alle MAC-Adressen authentifiziert werden. Hinterlegen Sie alle zu authentifizierenden MAC-Adressen auf dem RADIUS-Server. Tragen Sie die entsprechende Anzahl im Feld "MAC-Auth. max. zugelassene Adressen" ein.

  - 0
    Sie können den Wert "0" einstellen. Diese Einstellung bewirkt, dass nach der ersten erfolgreichen Authentifizierung einer MAC-Adresse der Port für alle MAC-Adressen freigeschaltet ist.
    Anwendungsfall
    Wenn Sie für einen Port den Wert "0" konfigurieren, schließen Sie an diesem Port einen WLAN-Access Point an. Nachdem sich der AP erfolgreich authentifiziert hat, sind alle MAC-Adressen auf diesem Port freigeschaltet. Alle mit dem AP verbundenen WLANClients können ohne eigene Authentifizierung an dem Port kommunizieren. Stellen Sie sicher, dass die Authentifizierung der Clients durch den AP erfolgt.
    Firmware-Kompatibilität
    In Firmware-Versionen < 6.0.2 ist der Wert "0" keine gültige Eingabe. Wenn Sie mit einer Firmware-Version ≥ 6.0.2 den Wert "0" konfigurieren und auf das Gerät eine FW-Version < 6.0.2 aufspielen, ist die Funktion MAC-Authentifizierung auf den entsprechenden Ports deaktiviert. Konfigurieren Sie die MAC-Authentifizierung auf diesen Ports neu.

- **Guest VLAN**
  Aktivieren Sie diese Option, wenn das Endgerät bei fehlgeschlagener Authentifizierung im "Guest VLAN" zugelassen werden soll.
  Der Port kann dem entsprechenden VLAN nur zugeordnet werden, wenn das VLAN auf dem Gerät angelegt ist. Die Authentifizierung (Seite 2123) wird sonst abgelehnt.
  Diese Funktion wird auch als "Authentication failed VLAN" bezeichnet.

- **Guest VLAN-ID**
  Geben Sie die VLAN-ID des Ports an.

- **Guest VLAN max. zugelassene Adressen**
  Geben Sie an, wie viele Endgeräte gleichzeitig im "Guest VLAN" zugelassen sind.

- **802.1X Auth.-Status**
  Zeigt den Status der Authentifizierung des Ports an:
  - Autorisieren
  - Nicht autorisieren

- **MAC-Auth. zurzeit zugelassene Adressen**
  Zeigt die Anzahl der momentan angeschlossen Endgeräte an.

- **MAC-Auth. zurzeit geblockte Adressen**
  Zeigt die Anzahl der momentan geblockten Endgeräte an.

- **Guest VLAN zurzeit zugelassene Adressen**
  Zeigt an, wie viele Endgeräte momentan im "Guest VLAN" zugelassen sind.

MAC ACL

Regelkonfiguration

Einleitung

Auf dieser Seite legen Sie die Zugriffsregeln für die MAC-basierte Access Control-Liste fest. Über die MAC-basierte ACL können Sie festlegen, ob Telegramme bestimmter MAC-Adressen weitergeleitet oder verworfen

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-MAC-Adresse**
  Tragen Sie die Unicast MAC-Adresse der Quelle ein.
  Ziel-MAC-Adresse
  Tragen Sie die Unicast MAC-Adresse des Ziels ein.
- **Aktion**
  Wählen Sie das Verhalten. Folgendes ist möglich:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Ingress-Schnittstelle**
  Zeigt eine Liste aller Schnittstellen an, für die diese Regel gilt.
- **Egress-Schnittstelle**
  Zeigt eine Liste aller Schnittstellen, für die diese Regel gilt.

Hinweis

Eingabe der MAC-Adressen

Sie können die Zugriffsregeln nur für Unicast MAC-Adressen konfigurieren. Nur wenn Sie bei Quell- und/oder Ziel-MAC-Adresse die Adresse "00:00:00:00:00:00" eingeben, gilt die so erstellte Regel für alle Quell- bzw. Ziel-MAC-Adressen.

Ingress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welchen ACL-Regeln eingehende Telegramme an Schnittstellen gefiltert werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fes

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie in der Klappliste die gewünschte Schnittstelle aus. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät.
- **Regel hinzufügen**
  Wählen Sie die ACL-Regel aus, die der Schnittstelle zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regelkonfiguration" fest.
- **Hinzufügen**
  Um der Schnittstelle die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von der Schnittstelle zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an.
- **Quell-MAC-Adresse**
  Zeigt die Unicast MAC-Adresse der Quelle an.
- **Ziel-MAC-Adresse**
  Zeigt die Unicast MAC-Adresse des Ziels an.
- **Aktion**
  Zeigt das Verhalten:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

Egress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden. Die ACL-Regel legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät.
- **Regel hinzufügen**
  Wählen Sie die ACL-Regel aus, die der Schnittstelle zugewiesen wird.
- **Hinzufügen**
  Um der Schnittstelle die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von der Schnittstelle zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an.
- **Quell-MAC-Adresse**
  Zeigt die Unicast MAC-Adresse der Quelle an.
- **Ziel-MAC-Adresse**
  Zeigt die Unicast MAC-Adresse des Ziels an.
- **Aktion**
  Zeigt das Verhalten an
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

IP ACL

Regelkonfiguration

Einleitung

Auf dieser Seite legen Sie die Regeln für die IP-basierte Access Controll-Liste fest. Über die IP-basierte ACL können Sie festlegen, ob Telegramme bestimmter IPv4-Adressen weitergeleitet oder verworfen werden.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- Regelnummer
  Zeigt die Nummer der ACL-Regel an. Wenn Sie einen neuen Eintrag anlegen, wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- Quell-IP-Adresse
  Geben Sie die IPv4-Adresse der Quelle ein.
- Quell-Subnetzmaske
  Geben Sie die Subnetzmaske ein, in der sich die Quelle befindet.
- Ziel-IP-Adresse
  Tragen Sie die IP-Adresse des Ziels ein.
- Ziel-Subnetzmaske
  Geben Sie die Subnetzmaske ein, in der sich das Ziel befindet.
- Aktion
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- Ingress-Schnittstelle
  Zeigt eine Liste aller Schittstellen an, für die diese Regel gilt.
- Egress-Schnittstelle
  Zeigt eine Liste aller Schittstellen an, für die diese Regel gilt.

Hinweis

Subnetzmaske für einzelne Hosts

Wenn Sie die Regel für ein einzelnes System erstellen (eine IPv4-Adresse), geben Sie die Subnetzmaske "255.255.255.255" an.

Protokollkonfiguration

Auf dieser Seite legen Sie die Regeln für Protokolle fest.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**
  Zeigt die Nummer der Protokollregel an. Beim Erstellen einer Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Protokoll**
  Wählen Sie das Protokoll aus, für das die Regel gilt.
- **Protokollnummer**
  Geben Sie eine Protokollnummer ein, um weitere Protokolle zu definieren. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "Other Protocol" eingestellt haben.
- **Quell-Port Min.**
  Geben Sie die kleinstmögliche Portnummer des Quell-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Quell-Port Max.**
  Geben Sie die größtmögliche Portnummer des Quell-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Ziel-Port Min.**
  Geben Sie die kleinstmögliche Portnummer des Ziel-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Ziel-Port Max.**
  Geben Sie die größtmögliche Portnummer des Ziel-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Nachrichtentyp**
  Geben Sie einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.
- **Nachrichtencode**
  Geben Sie einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.
- **DSCP**
  Geben Sie einen Wert zur Klassifizierung der Priorität ein. Dieses Feld ist nicht editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.

Ingress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welchen ACL-Regeln eingehende Telegramme von Schnittstellen behandelt werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Beschreibung der angezeigten Felder:

- Schnittstelle
  Wählen Sie den gewünschten Port aus: einen Port oder eine VLAN-Schnittstelle. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät. Um eine VLAN-Schnittstelle auswählen zu können, muss eine IP-Schnittstelle konfiguriert sein.

  Hinweis

  Wenn Sie eine VLAN-Schnittstelle verwenden, gilt die ACL-Regel an allen Ports, die zu dem VLAN gehören.

- **Regel hinzufügen**
  Wählen Sie die ACL-Regel aus, die dem Port zugewiesen wird.
- **Hinzufügen**
  Um dem Port die ACL-Regel fest zuzuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von dem Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an.
- **Protokoll**
  Zeigt das Protokoll an, für das die Regel gilt.
- **Protokollnummer**
  Zeigt die Protokollnummer an.
- **Quell-IP-Adresse**
  Zeigt Sie die IP-Adresse der Quelle an.
- **Quell-Subnetzmaske**
  Zeigt die Subnetzmaske an, in der sich die Quelle befindet.
- **Ziel-IP-Adresse**
  Zeigt die IP-Adresse des Ziels an.
- **Ziel-Subnetzmaske**
  Zeigt die Subnetzmaske des Ziels an..

- **Aktion**
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Quell-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Quell-Ports an.
- **Quell-Port Max.**
  Zeigt die größtmögliche Portnummer des Quell-Ports an.
- **Ziel-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Ziel-Ports an.
- **Ziel-Port Max.**
  Zeigt die größtmögliche Portnummer des Ziel-Ports an.
- **Nachrichtentyp**
  Zeigt einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen.
- **Nachrichtencode**
  Zeigt einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren.
- **DSCP**
  Zeigt den Wert zur Klassifizierung der Priorität an.

Egress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie den gewünschten Port aus: einen Port oder eine VLAN-Schnittstelle. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät. Um eine VLAN-Schnittstelle auswählen zu können, muss eine IP-Schnittstelle konfiguriert sein.

  Hinweis

  Wenn Sie eine VLAN-Schnittstelle verwenden, gilt die ACL-Regel an allen Ports, die zu dem VLAN gehören.

- **Regel hinzufügen**
  Wählen Sie die ACL-Regel, die dem Port zugewiesen wird.

- **Hinzufügen**
  Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von dem Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an.
- **Protokoll**
  Zeigt das Protokoll an, für das die Regel gilt.
- **Protokollnummer**
  Zeigt die Protokollnummer an.
- **Quell-IP-Adresse**
  Zeigt Sie die IP-Adresse der Quelle an.
- **Quell-Subnetzmaske**
  Zeigt die Subnetzmaske der Quelle an.
- **Ziel-IP-Adresse**
  Zeigt die IP-Adresse des Ziels an.
- **Ziel-Subnetzmaske**
  Zeigt die Subnetzmaske der Quelle an..
- **Aktion**
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Quell-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Quell-Ports an.
- **Quell-Port Max.**
  Zeigt die größtmögliche Portnummer des Quell-Ports an.
- **Ziel-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Ziel-Ports an.
- **Ziel-Port Max.**
  Zeigt die größtmögliche Portnummer des Ziel-Ports an.
- **Nachrichtentyp**
  Zeigt einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen.

- Nachrichtencode
  Zeigt einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren.
- DSCP
  Zeigt den Wert zur Klassifizierung der Priorität an.

Management ACL

Konfigurationsbeschreibung

Auf dieser Seite können Sie die Sicherheit Ihres Geräts erhöhen. Um festzulegen, welche Station mit welcher IPv4-Adresse auf Ihr Gerät zugreifen darf, konfigurieren Sie die IPv4-Adresse oder auch ein ganzes Adress-Band.

Sie können einstellen, mit welchen Protokollen und über welche Ports die Station auf das Gerät zugreifen darf. Sie definieren, in welchem VLAN die Station liegen darf. Damit wird gewährleistet, dass nur bestimmte Stationen innerhalb eines VLANs Zugriff das Gerät haben.

Beschreibung der angezeigten Felder

Hinweis

Wenn Sie diese Funktion aktivieren, beachten Sie Folgendes

Eine fehlerhafte Projektierung auf der Seite "Management Access Control List" kann dazu führen, dass Sie nicht mehr auf das Gerät zugreifen können. Projektieren Sie daher eine Zugriffsregel, die Ihnen den Zugriff auf das Management erlaubt, bevor Sie die Funktion aktivieren

Die Seite enthält folgende Felder:

- Management ACL
  Aktivieren oder deaktivieren Sie die Funktion. Im Auslieferungszustand ist die Funktion deaktiviert.

  Hinweis

  Wenn die Funktion deaktiviert ist, dann besteht uneingeschränkter Zugriff auf das Management des IE-Switches. Erst wenn die Funktion aktiviert ist, werden die projektierten Zugriffsregeln berücksichtigt

- IP-Adresse
  Geben Sie die IPv4-Adresse oder die Netzadresse ein, für die die Regel gelten soll. Wenn Sie die IPv4-Adresse 0.0.0.0 verwenden, gelten die Einstellungen für alle IPv4-Adressen. Wenn Sie die IPv6-Adresse :: verwenden, gelten die Einstellungen für alle IPv6-Adressen.
- Subnetzmaske
  Tragen Sie die Subnetzmaske oder die Präfixlänge ein. Die Subnetzmaske 255.255.255.255 ist für eine bestimmte IPv4-Adresse. Möchten Sie ein Subnetz zulassen, tragen Sie z. B. für ein C-Subnetz 255.255.255.0 ein. Die Subnetzmaske 0.0.0.0 gilt für alle Subnetze.

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **IP-Adresse**
  Zeigt die IPv4-Adresse an.
- **Subnetzmaske/Präfixlänge**
  Zeigt die Subnetzmaske oder die Präfixlänge an.
- **Zulässige VLANs**
  Geben Sie die Nummer des VLANs ein, in dem sich das Gerät befindet. Nur die Station kann auf das Gerät zugreifen, wenn es sich in diesem konfigurierten VLAN befindet. Bleibt dieses Eingabefeld leer, gibt es keine Einschränkung bezüglich der VLANs.
- **Out-Band**
  Legen Sie fest, ob die IPv4-Adresse über den Out-Band-Port auf den Switch zugreifen kann.
- **SNMP**
  Legen Sie fest, ob die Station (bzw. die IP-Adresse) über das Protokoll SNMP das Gerät zugreifen darf.
- **TELNET**
  Legen Sie fest, ob die Station (bzw. die IP-Adresse) über das Protokoll TELNET auf das Gerät zugreifen darf.
- **HTTP**
  Legen Sie fest, ob die Station (bzw. die IP-Adresse) über das Protokoll HTTP auf das Gerät zugreifen darf.
- **HTTPS**
  Legen Sie fest, ob die Station (bzw. die IP-Adresse) über das Protokoll HTTPS auf das Gerät zugreifen dar.
- **SSH**
  Legen Sie fest, ob die Station (bzw. die IP-Adresse) über das Protokoll SSH auf das Gerät zugreifen darf..
- **Px.y**
  LegenSie fest, ob die Station (bzw. die IP-Adresse) über diesen Port (Slot.Port) auf dieses Gerät zugreifen darf.

SCALANCE W projektieren

Schnittstellen und Systemfunktionen

SCALANCE W780 / W740

Verfügbarkeit der Schnittstellen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der physischen und der logischen Schnittstellen an. Beachten Sie, dass in dieser Tabelle alle Schnittstellen aufgelistet sind. Abhängig von der Systemfunktion stehen Ihnen manche Schnittstellen nicht zur Verfügung. Auf den WBM-Seiten können Sie nur die verfügbaren Schnittstellen auswählen.

Technische Änderungen sind vorbehalten.

	Client-Gerät W748-1 M12 W748-1 RJ45	Access Points W786-1 RJ45 W788-1 M12 W788-1 RJ45	Access Points W786-2 RJ45 W786-2IA RJ45 W786-2 SFP W788-2 M12 W788-2 M12 EEC W788-1 RJ45 W788-2 RJ45
Funkschnittstelle (WLAN)	WLAN 1	WLAN 1	• WLAN 1 (Im Client-Modus ist nur eine WLAN-Schnittstelle verfügbar) • WLAN 2
IP-Schnittstelle: LAN-Schnittstelle VLAN	P1 ManagementVLAN	P1 ManagementVLAN	P1 ManagementV-LAN
VAP-Schnittstelle [1]	-	VAP 1.Y Y = 1 ... 8	VAP X.Y X = 1 ... 2 Y = 1 ... 8
WDS-Schnittstelle [1]	_	WDS 1.Y Y = 1 ... 8	WDS X.Y X = 1 ... 2 Y = 1 ... 8
VLAN	24	24	24

[1] nur im Access Point-Modus

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Geräten. Beachten Sie, dass in diesem Projektierungshandbuch und der Online-Hilfe alle Funktionen beschrieben sind. Abhängig von der Betriebsart und dem KEY PLUG stehen Ihnen manche Funktionen nicht zur Verfügung.

Technische Änderungen sind vorbehalten.

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Informationen	Security	Inter AP Blocking	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-
	WLAN	Übersicht AP	✓	-
		Client-Liste	✓	-
		WDS-LIste	✓	-
		Überlappung AP	✓	-
		Roaming erzwingen	✓	✓
		Übersicht Client	-	✓
		Verfügbare APs	-	✓
		IP-Zuordnung	-	✓
		Hintergrund-rauschen	✓	✓
	WLAN-Statistiken	Fehler	✓	✓
		Management gesendet	✓	✓
		Management empfangen	✓	✓
		Gesendete Daten	✓	✓
		Empfangene Daten	✓	✓
	WLAN iFeatures	iREF Client-Liste	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
		iREF WDS-Liste	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
		AeroScout	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
System		PROFINET	✓	-✓
		EtherNet/IP	✓	✓
Schnittstellen	WLAN	Basic	✓	-✓
		Erweiterungen	✓	✓
		Antennen	✓	✓
		Zugelassene Kanäle	✓	✓
		802.11n	✓	✓
		AP	✓	-
		AP WDS	✓	-
		AP 802.11a/b/g-Datenraten	✓	-
		AP 802.11n-Datenraten	✓	-
		Client 802.11a/b/g-Datenraten	-	✓
		Client 802.11n-Datenraten	-	✓
		Roaming erzwingen	✓	✓
		Signalrekorder	-	✓
		Spektrumanalysator	✓	-
Layer 3	NAT	Basic	-	✓
		NAPT	-	✓

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Security	WLAN	Basic	✓	✓
		AP-Kommunikation	✓	-
		AP RADIUS-Authenticator	✓	-
		Client RADIUS-Supplicant	-	✓
		Schlüssel	✓	✓
	Inter AP Blocking	Basic	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-
		Zugelassene Adressen	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
iFeatures	iPCF		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iPCF-MC		✓ Nur Dual-APs W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iPRP		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iREF		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
	AeroScout		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-

SCALANCE W770 / W730

Verfügbarkeit der Schnittstellen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der physischen und der logischen Schnittstellen an. Beachten Sie, dass in dieser Tabelle alle Schnittstellen aufgelistet sind. Abhängig von der Systemfunktion stehen Ihnen manche Schnittstellen nicht zur Verfügung. Auf den WBM-Seiten können Sie nur die verfügbaren Schnittstellen auswählen.

Technische Änderungen sind vorbehalten.

	Client-Gerät SCALANCE W734-1 RJ45 SCALANCE W738-1 M12	Access Points SCALANCE W774-1 RJ45 SCALANCE W774-1 M12 EEC SCALANCE W778-1 M12 SCALANCE W778-1 M12 EEC
Funkschnittstelle (WLAN)	WLAN 1	WLAN 1
IP-Schnittstelle: LAN-Schnittstelle VLAN	 P1 ManagementVLAN	 P1 ManagementVLAN
VAP-Schnittstelle [1)]	-	VAP 1.Y Y = 1 ... 8
WDS-Schnittstelle [1)]	_	WDS 1.Y Y = 1 ... 8
VLAN	16	16

[1)] nur im Access Point-Modus

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Geräten. Beachten Sie, dass in diesem Projektierungshandbuch und der Online-Hilfe alle Funktionen beschrieben sind. Abhängig von der Betriebsart und dem KEY PLUG stehen Ihnen manche Funktionen nicht zur Verfügung.

Technische Änderungen sind vorbehalten.

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Informationen	Security	Inter AP Blocking	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-
	WLAN	Übersicht AP	✓	-
		Client-Liste	✓	-
		WDS-LIste	✓	-
		Überlappung AP	✓	-
		Roaming erzwingen	✓	-
		Übersicht Client	-	✓
		Verfügbare APs	-	✓
		IP-Zuordnung	-	✓
	WLAN-Statistiken	Fehler	✓	✓
		Management gesendet	✓	✓
		Management empfangen	✓	✓
		Gesendete Daten	✓	✓
		Empfangene Daten	✓	✓
	WLAN iFeatures	iREF Client-Liste	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
		iREF WDS-Liste	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
		AeroScout	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
System		PROFINET	✓	-✓
		EtherNet/IP	✓	✓

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Schnittstellen	WLAN	Basic	✓	-✓
		Erweiterungen	✓	✓
		Antennen	✓	✓
		Zugelassene Kanäle	✓	✓
		802.11n	✓	✓
		AP	✓	-
		AP WDS	✓	-
		AP 802.11a/b/g-Datenraten	✓	-
		AP 802.11n-Datenraten	✓	-
		Client 802.11a/b/g-Datenraten	-	✓
		Client 802.11n-Datenraten	-	✓
		Roaming erzwingen	✓	✓
		Signalrekorder	-	✓
		Spektrumanalysator	✓	-
Layer 3	NAT	Basic	-	✓
		NAPT	-	✓

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Security	WLAN	Basic	✓	✓
		AP-Kommunikation	✓	-
		AP RADIUS-Authenticator	✓	-
		Client RADIUS-Supplicant	-	✓
		Schlüssel	✓	✓
	Inter AP Blocking	Basic	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-
		Zugelassene Adressen	✓ W780 iFeatures (MLFB 6GK5 907-8PA00) W700 Security (MLFB 6GK5907-0PA00)	-

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
iFeatures	iPCF	iPCFv1	✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iPCF-MC		✓ Nur Dual-APs W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iPRP		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	✓ Access Point im Client-Modus: W780 iFeatures (MLFB 6GK5 907-8PA00 Client: W740 iFeatures (MLFB 6GK5 907-4PA00)
	iREF		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-
	AeroScout		✓ W780 iFeatures (MLFB 6GK5 907-8PA00)	-

SCALANCE W760 / W720

Verfügbarkeit der Schnittstellen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der physischen und der logischen Schnittstellen an. Beachten Sie, dass in dieser Tabelle alle Schnittstellen aufgelistet sind. Abhängig von der Systemfunktion stehen Ihnen manche Schnittstellen nicht zur Verfügung. Auf den WBM-Seiten können Sie nur die verfügbaren Schnittstellen auswählen.

Technische Änderungen sind vorbehalten.

	Client-Gerät W721-1 RJ45 W722-1 RJ45	Access Points W761-1 RJ45
Funkschnittstelle (WLAN)	WLAN 1	WLAN 1
IP-Schnittstelle: LAN-Schnittstelle VLAN	 P1 ManagementVLAN	 P1 ManagementVLAN
VAP-Schnittstelle [1)]	-	VAP 1.1
WDS-Schnittstelle [1)]	_	WDS 1.1
VLAN	3	3

[1)] nur im Access Point-Modus

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Geräten. Beachten Sie, dass in diesem Projektierungshandbuch und der Online-Hilfe alle Funktionen beschrieben sind. Abhängig von der Betriebsart und dem KEY PLUG stehen Ihnen manche Funktionen nicht zur Verfügung.

Technische Änderungen sind vorbehalten.

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Informationen	WLAN	Übersicht AP	✓	-
		Client-Liste	✓	-
		WDS-LIste	✓	-
		Überlappung AP	✓	-
		Roaming erzwingen	✓	-
		Übersicht Client	-	✓
		Verfügbare APs	-	✓
		IP-Zuordnung	-	✓
	WLAN-Statistiken	Fehler	✓	✓
		Management gesendet	✓	✓
		Management empfangen	✓	✓
		Gesendete Daten	✓	✓
		Empfangene Daten	✓	✓
System		PROFINET	✓	✓
		EtherNet/IP	✓	✓

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
Schnittstellen	WLAN	Basic	✓	✓
		Erweiterungen	✓	✓
		Antennen	✓	✓
		Zugelassene Kanäle	✓	✓
		802.11n	✓	✓
		AP	✓	-
		AP WDS	✓	-
		AP 802.11a/b/g-Datenraten	✓	-
		AP 802.11n-Datenraten	✓	-
		Roaming erzwingen	✓	✓
		Signalrekorder	-	✓
		Spektrumanalysator	✓	-
Layer 3	NAT	Basic	-	✓
		NAPT	-	✓
Security	WLAN	Basic	✓	✓
		AP-Kommunikation	✓	-
		AP RADIUS-Authenticator	✓	-
		Client RADIUS-Supplicant	-	✓
		Schlüssel	✓	✓

			Access Points-Modus	Access Points im Client-Modus Client-Geräte
iFeatures	iPCF		-	✓ Nur W722-1 RJ45 Dieser kann in ein WLAN mit iPCF, iPCF-MC integriert werden
	iPCF-HT		-	✓ Nur W722-1 RJ45
	iPCF-MC		-	✓ Nur W722-1 RJ45 Dieser kann in ein WLAN mit iPCF, iPCF-MC integriert werden
	iPRP		-	✓ Nur W722-1 RJ45

IPv6-Unterstützung

Unterstützung von IPv6

Folgende Systemfunktionen unterstützen keine IPv6-Adressen:

- Inter AP Blocking
- Roaming erzwingen
- IP ACL
- Management ACL

Mengengerüste

SCALANCE W780 / W740

Mengengerüst des Geräts

In der folgenden Tabelle ist das Mengengerüst für das Web Based Management und das Command Line Interface des Geräts aufgeführt.

Abhängig von Ihrem Gerät stehen Ihnen manche Funktionen nicht zur Verfügung.

	konfigurierbare Funktion		maximale Anzahl
System	Syslog-Server		3
	DNS-Server	manual (IPv4/IPv6)	3
		learned (IPv4/IPv6)	2
		insgesamt	7
	SMTP-Server		2
	SNMPv1-Trap-Empfänger		10
	SNTP-Server		2
	NTP-Server		1
	DHCP-Pools		1
	IPv4-Adressen, die der DHCP-Server verwaltet (dynamisch + statisch)		100
	DHCP statische Zuordnungen pro DHCP-Pool		20
	DHCP-Optionen		20
Schnittstellen	Zieladressen für Roaming erzwingen		10
	Verbundene Clients pro VAP		126
Layer 2	Virtuelle LANs (portbased; inklusive VLAN 1)		24
	Multiple Spanning Tree-Instanzen		16
Security	IP-Adressen von RADIUS-Servern		• AAA: 4 • WLAN: 2
	Management ACLs (Zugriffsregeln für das Management)		10
	MAC ACL Regelkonfiguration		20
	Ingress- und Egress-Regeln für MAC ACL (gesamt)		40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 80 (P1, WLAN) • Acccess Point: 680 (P1, WDS 1,Y, VAP 1,Y) • Dual Access Point: 1320 (P1, WDS X,Y, VAP X,Y)
	IP ACL Regelkonfiguration		20
	Ingress- und Egress-Regeln für Port ACL IP (gesamt)		40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 120 (P1, WLAN, ManagementVLAN) • Acccess Point: 720 (P1, WDS 1,Y, VAP 1,Y, ManagementVLAN) • Dual Access Point: 1360 (P1, WDS X,Y, VAP X,Y, ManagementVLAN)
	Benutzerrollen		28
	Benutzergruppen		32
	Benutzer		28

SCALANCE W770 / W730

Mengengerüst des Geräts

In der folgenden Tabelle ist das Mengengerüst für das Web Based Management und das Command Line Interface des Geräts aufgeführt.

Abhängig von Ihrem Gerät stehen Ihnen manche Funktionen nicht zur Verfügung.

	konfigurierbare Funktion		maximale Anzahl
System	Syslog-Server		3
	DNS-Server	manual (IPv4/IPv6)	3
		learned (IPv4/IPv6)	2
		insgesamt	7
	SMTP-Server		3
	SNMPv1-Trap-Empfänger		10
	SNTP-Server		2
	NTP-Server		1
	DHCP-Pools		1
	IPv4-Adressen, die der DHCP-Server verwaltet (dynamisch + statisch)		100
	DHCP statische Zuordnungen pro DHCP-Pool		20
	DHCP-Optionen		20
Schnitt-stellen	Zieladressen für Roaming erzwingen		10
	Verbundene Clients pro VAP		126
Layer 2	Virtuelle LANs (portbased; inklusive VLAN 1)		16
	Multiple Spanning Tree-Instanzen		16

	konfigurierbare Funktion	maximale Anzahl
Security	IP-Adressen von RADIUS-Servern	• AAA: 4 • WLAN: 2
	Management ACLs (Zugriffsregeln für das Management)	10
	MAC ACL Regelkonfiguration	20
	Ingress- und Egress-Regeln für MAC ACL (gesamt)	40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 80 (P1, WLAN) • Acccess Point: 680 (P1, WDS 1.Y, VAP 1.Y)
	IP ACL Regelkonfiguration	20
	Ingress- und Egress-Regeln für Port ACL IP (gesamt)	40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 120 (P1, WLAN, ManagementVLAN) • Acccess Point: 720 (P1, WDS 1.Y, VAP 1.Y, ManagementVLAN)
	Benutzerrollen	28
	Benutzergruppen	32
	Benutzer	28

SCALANCE W760 / W720

Mengengerüst des Geräts

In der folgenden Tabelle ist das Mengengerüst für das Web Based Management und das Command Line Interface des Geräts aufgeführt.

Abhängig von Ihrem Gerät stehen Ihnen manche Funktionen nicht zur Verfügung.

	konfigurierbare Funktion		maximale Anzahl
System	Syslog-Server		3
	DNS-Server	manual (IPv4/IPv6)	3
		learned (IPv4/IPv6)	2
		insgesamt	7
	SMTP-Server		3
	SNMPv1-Trap-Empfänger		10
	SNTP-Server		2
	NTP-Server		1
	DHCP-Pools		1
	IPv4-Adressen, die der DHCP-Server verwaltet (dynamisch + statisch)		100
	DHCP statische Zuordnungen pro DHCP-Pool		20
	DHCP-Optionen		20
Schnittstellen	Zieladressen für Roaming erzwingen		10
	Verbundene Clients pro VAP		126
Layer 2	Virtuelle LANs (portbased; inklusive VLAN 1)		3
	Multiple Spanning Tree-Instanzen		16
Security	IP-Adressen von RADIUS-Servern		• AAA: 4 • WLAN: 2
	Management ACLs (Zugriffsregeln für das Management)		10
	MAC ACL Regelkonfiguration		20
	Ingress- und Egress-Regeln für MAC ACL (gesamt)		40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 80 (P1, WLAN) • Acccess Point: 680 (P1, WDS 1.Y, VAP 1.Y)
	IP ACL Regelkonfiguration		20
	Ingress- und Egress-Regeln für IP ACL IP (gesamt)		40 pro Schnittstelle (20 Ingress-Regeln / 20 Egress-Regeln) • Client: 120 (P1, WLAN, ManagementVLAN) • Acccess Point: 720 (P1, WDS 1.Y, VAP 1.Y, ManagementVLAN) • Dual Access Point: 1360 (P1, WDS X.Y, VAP X.Y, ManagementVLAN)
	Benutzerrollen		28
	Benutzergruppen		32
	Benutzer		28

Wissenswertes

IP-Adressen

IPv4 / IPv6

Was sind die wesentlichen Unterschiede?

	IPv4	IPv6
IP-Konfiguration	• DHCP-Server • Manuell	• Stateless Address Autoconfiguration (SLAAC): zustandslose Autokonfiguration über NDP (Neighbor Discovery Protocol) – Erstellt eine Link-lokale Adresse für jede Schnittstelle, die keinen Router auf dem Link benötigt. – Prüft die Einmaligkeit der Adresse auf dem Link, die keinen Router auf dem Link benötigt. – Legt fest, ob die globalen Adressen über einen zustandslosen Mechanismus, einen zustandbehafte Mechanismus oder über beide Mechanismen bezogen werden. (Erfordert einen Router auf dem Link.) • Manuell • DHCPv6 (Zustandbehaftet)
Verfügbare IP-Adressen	32-Bit: 4, 29 * 10^9 Adressen	128-Bit: 3,4 * 10^{38} Adressen
Adressformat	Dezimal: 192.168.1.1 mit Port: 192.168.1.1:20	Hexadezimal: 2a00:ad80::0123 mit Port: [2a00:ad80::0123]:20
Loopback	127.0.0.1	::1
IP-Adressen pro Schnittstelle	4 IP-Adressen	Mehrere IP-Adressen • LLA: Eine link lokale-Adresse (automatisch gebildet) fe80::/128 pro Schnittstelle • ULA: Mehrere Unique lokal Unicast-Adressen pro Schnittstelle • GUA: Mehrere Globale Unicast-Adressen pro Schnittstelle
Header	• Checksumme • variable Länge • Fragmentierung im Header • keine Sicherheit	• Überprüfung auf höherer Schicht • fest vorgeschriebene Größe • Fragmentierung im Extension Header
Fragmentierung	Host und Router	nur Endpunkt der Kommunikation
Quality of Service	Type of Service zur Priorisierung (ToS)	Die Priorisierung wird im Header-Feld "Traffic Class" angegeben.

	IPv4	IPv6
Telegrammarten	Broadcast, Multicast, Unicast	Multicast, Unicast, Anycast
Identifizierung von DHCP-Clients/-Server	Client-ID: • MAC-Adresse • DHCP Client ID • System Name • PROFINET-Stationsname • IAID und DUID	DUID + IAID(s) = genau eine Schnittstelle des Hosts DUID = DHCP Unique Identifier Identifiziert Server und Clients eindeutig und sollte sich nicht ändern, auch nicht beim Austausch von Netzwerkkomponenten! IAID = Identity Association Identifier Mindestens eine pro Schnittstelle wird vom Client generiert und bleibt bei Neustart des DHCP-Clients unverändert Drei Verfahren zum Ermitteln der DUID • DUID-LLT • DUID-EN • DUID-LL
DHCP	über UDP mittels Broadcast	über UDP mittels Unicast RFC 3315, RFC 3363 **Stateful DHCPv6** Zustandsbehaftete Konfiguration bei der die IPv6-Adresse und die Konfigurationseintellungen übermittelt werden. Dabei werden vier DHCPv6-Nachrichten zwischen Client und Server ausgetauscht: 1. SOLICIT: Gesendet durch den DHCPv6-Client, um DHCPv6-Server zu lokalisieren. 2. ADVERTISE Die verfügbaren DHCPv6-Server antworten darauf. 3. REQUEST Der DHCPv6-Client fordert eine IPv6-Adresse und die Konfigurationseinstellungen beim DHCPv6-Server an. 4. REPLY Der DHCPv6-Server sendet die IPv6-Adresse und die Konfigurationseinstellungen. Wenn Client und Server die Funktion "Rapid commit" unterstützen, wird das Verfahren auf zwei DHCPv6-Nachrichten SOLICIT und REPLY gekürzt. **Stateless DHCPv6** Beim zustandslosen DHCPv6 werden nur die Konfigurationseinstellungen übermittelt. **Präfix-Delegation** Der DHCPv6-Server delegiert das Verteilen von IPv6-Präfixe an den DHCPv6-Client. Der DHCPv6-Client wird auch als PD-Router bezeichnet.
Auflösung von IP-Adressen in Hardware-Adressen	ARP (Address Resolution Protocol)	NDP (Neighbor Discovery Protocol)

IPv4-Adresse

Aufbau einer IPv4-Adresse

Adressklassen

IP-Adressbereich	Max. Anzahl der Netzwerke	Max. Anzahl Hosts/Netzwerk	Klasse	CIDR
1.x.x.x bis 126.x.x.x	126	16777214	A	/8
128.0.x.x bis 191.255.x.x	16383	65534	B	/16
192.0.0.x bis 223.255.255.x	2097151	254	C	/24
224.0.0.0 - 239.255.255.255	Multicast-Anwendungen		D	
240.0.0.0 - 255.255.255.255	reserviert für zukünftige Anwendungen		E	

Eine IP-Adresse besteht aus 4 Byte. Jedes Byte wird dezimal dargestellt und ist durch einen Punkt vom vorherigen getrennt. Es ergibt sich also folgender Aufbau, wobei für XXX eine Zahl zwischen 0 und 255 zu setzen ist:

XXX.XXX.XXX.XXX

Die IP-Adresse besteht aus zwei Teilen, der Netzwerkadresse und der Endteilnehmeradresse. Dadurch ist es möglich, verschiedene Teilnetze zu bilden. Abhängig davon, welche Bytes der IP-Adresse als Netzwerkadresse und welche als Endteilnehmeradresse genutzt werden, kann eine IP-Adresse einer bestimmten Adressklasse zugeordnet werden.

Subnetzmaske

Die Bits der Endteilnehmer-Adresse können für die Bildung von Subnetzen verwendet werden. Dabei stellen die führenden Bits die Adresse des Subnetzes dar, die restlichen Bits werden als Adresse des Rechners im Subnetz interpretiert.

Ein Subnetz wird durch die Subnetzmaske definiert. Der Aufbau der Subnetzmaske entspricht dem einer IP-Adresse. Ist in der Subnetzmaske an einer Bitposition eine "1" gesetzt, gehört das Bit an der entsprechenden Stelle in der IP-Adresse zur Subnetzadresse, andernfalls zur Adresse des Rechners.

Beispiel für ein Klasse B-Netz:

Die Standard-Subnetz-Adresse für Klasse B-Netze ist 255.255.0.0, es stehen also die letzten beiden Bytes für die Festlegung eines Subnetzes zur Verfügung. Wenn 16 Teilnetze definiert werden sollen, muss das dritte Byte der Subnetzadresse auf 11110000 (Binärdarstellung) gesetzt werden. In diesem Fall ergibt sich die Subnetzmaske 255.255.240.0.

Um festzustellen, ob zwei IP-Adressen zum gleichen Subnetz gehören, werden auf die beiden IP-Adressen und die Subnetzmaske eine bitweise UND-Verknüpfung angewendet. Wenn beide Verknüpfungen das gleiche Ergebnis haben, gehören beide IP-Adressen zum gleichen Subnetz, wie z. B. 141.120.246.210 und 141.120.252.108.

Außerhalb des lokalen Netzwerks ist die beschriebene Aufteilung der Endteilnehmer-Adresse ohne Bedeutung, dort ist für die Paketvermittlung nur die IP-Adresse in ihrer Gesamtheit von Interesse.

Hinweis

In der Bit-Darstellung der Subnetzmaske müssen die "Einsen" linksbündig gesetzt sein, d. h. es dürfen keine "Nullen" zwischen den "Einsen" stehen.

IPv6-Adressen

IPv6-Begriffe

Netzknoten

Ein Netzknoten ist ein Gerät, das über ein oder mehrere Schnittstellen an ein oder mehrere Netzwerke angeschlossen ist.

Router

Ein Netzknoten, der IPv6-Pakete weiterleitet.

Host

Ein Netzknoten, der einen Endpunkt für IPv6-Kommunikationsbeziehungen darstellt.

Link

Als Link wird in der IPv6-Terminologie eine direkte Layer 3-Verbindung innerhalb eines IPv6-Netzwerks bezeichnet.

Nachbar

Zwei Netzknoten werden als Nachbarn bezeichnet, wenn sie sich auf dem gleichen Link befinden.

IPv6-Schnittstelle

Physische oder logische Schnittstelle, an der IPv6 aktiviert ist.

Path MTU

Maximal zulässige Paketgröße auf einem Pfad, von einem Absender zu einem Empfänger.

Path MTU Discovery

Mechanismus zur Bestimmung der maximal zulässigen Paketgröße entlang des gesamten Pfades von einem Absender zu einem Empfänger.

LLA

Link lokale-Adresse FE80::/10

Sobald IPv6 auf der Schnittstelle aktiviert wird, wird automatisch eine link lokale-Adresse gebildet. Nur erreichbar für Knoten, die sich auf dem gleichen Link befinden.

ULA

Unique lokal-Adresse

Definiert in der RFC 4193. Über diese Adresse ist das IPv6-Schnittstelle im LAN erreichbar.

GUA

Globale Unicast-Adresse. Über diese Adresse ist IPv6-Schnittstelle z. B. über das Internet erreichbar.

Schnittstellen-ID

Die Schnittstellen-ID wird mit dem EUI-64 Verfahren oder manuell gebildet.

EUI-64

Extended Unique Indentifier (RFC 4291); Verfahren zur Bildung der Schnittstellen-ID. Bei Ethernet wird die Schnittstellen-ID aus der MAC-Adresse der Schnittstelle gebildet. Teilt die MAC-Adresse in den herstellerspezifischen Teil (OUI) und den netzwerkspezifischen Teil (NIC) und fügt zwischen den beiden Teilen FFFE ein.

Beispiel:

MAC-Adresse = AA:BB:CC:DD:EE:FF

OUI = AA:BB:CC

NIC = DD:EE:FF

EUI-64 = OUI + **FFFE** + NIC = AA:BB:CC:**FF:FE**:DD:EE:FF

Scope

Definiert die Reichweite der IPv6-Adresse.

Aufbau einer IPv6-Adresse

Adressformat IPv6-Notation

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vierstelligen Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt.

Beispiel:

fd00:0000:0000:ffff:02d1:7d01:0000:8f21

Regeln/Vereinfachungen:

- Wenn ein oder mehr Blöcke den Wert 0 haben, ist eine verkürzte Schreibweise möglich. Die Adresse fd00:**0000:0000**:ffff:02d1:7d01:0000:8f21 kann verkürzt auch wie folgt notiert werden:
  fd00**::**ffff:02d1:7d01:0000:8f21
  Damit Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden.
- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.
  Die Adresse fd00:0000:0000:ffff:**02d1**:7d01:0000:8f21 kann verkürzt auch wie folgt notiert werden:
  fd00**::**ffff:**2d1**:7d01:0000:8f21
- Dezimalnotation mit Punkt-Schreibweise
  Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.
  Beispiel: Die IPv6-Adresse fd00::ffff.125.1.0.1 ist äquivalent zu fd00::ffff:7d01:1

Aufbau IPv6-Adresse

Das IPv6-Protokoll unterscheidet drei Adressarten: Unicast , Anycast und Multicast. Der folgende Abschnitt beschreibt den Aufbau der globalen Unicast-Adressen.

IPv6-Präfix		Suffix
Globales Präfix: n Bit	Subnetz-ID m Bit	Schnittstelle-ID 128 - n - m Bits
Zugewiesener Adress-bereich	Beschreibung des Standorts, auch Subnetzpräfix oder Sub-netz	Eindeutige Zuordnung des Host im Netz. Die ID wird aus der MAC-Adresse gene-riert.

Das Präfix für die link local-Adresse ist immer fe80:0000:0000:0000. Das Präfix wird verkürzt und wie folgt notiert: fe80::

IPv6-Präfix

Festgelegt in: RFC 4291

Das IPv6-Präfix stellt die Subnetzkennung dar.

Präfixe und IPv6-Adressen werden auf dieselbe Weise angegeben wie bei der CIDR-Notation (Classless Inter-Domain Routing) für IPv4.

Aufbau

IPv6-Adresse / Präfixlänge

Beispiel

IPv6-Adresse: 2001:0db8:1234::1111/48

Präfix: 2001:0db8:1234::/48

Schnittstelle-ID: ::1111

Eingabe und Darstellung

Die Eingabe von IPv6-Adressen ist in den zuvor beschriebenen Notationen möglich. IPv6-Adressen werden immer in der hexadezimalen Notation angezeigt.

VLAN

VLAN-Tagging

Erweiterung der Ethernet-Telegramme um vier Byte

Für CoS (Class of Service, Telegrammpriorisierung) und für VLAN (Virtuelles Netzwerk) wurde in der Norm IEEE 802.1Q die Erweiterung der Ethernet-Telegramme um das VLAN-Tag festgelegt.

Hinweis

Durch das VLAN-Tag erhöht sich die zulässige Gesamtlänge des Telegramms von 1518 auf 1522 Byte.
Es muss geprüft werden, ob die Endteilnehmer im Netz diese Länge / diesen Telegrammtyp verarbeiten können. Ist dies nicht der Fall, dürfen an diese Teilnehmer nur Telegramme mit der Standardlänge gesendet werden.

Die zusätzlichen 4 Bytes befinden sich im Header des Ethernet-Telegramms zwischen der Quelladresse und dem Ethernet-Typ-/Längenfeld:

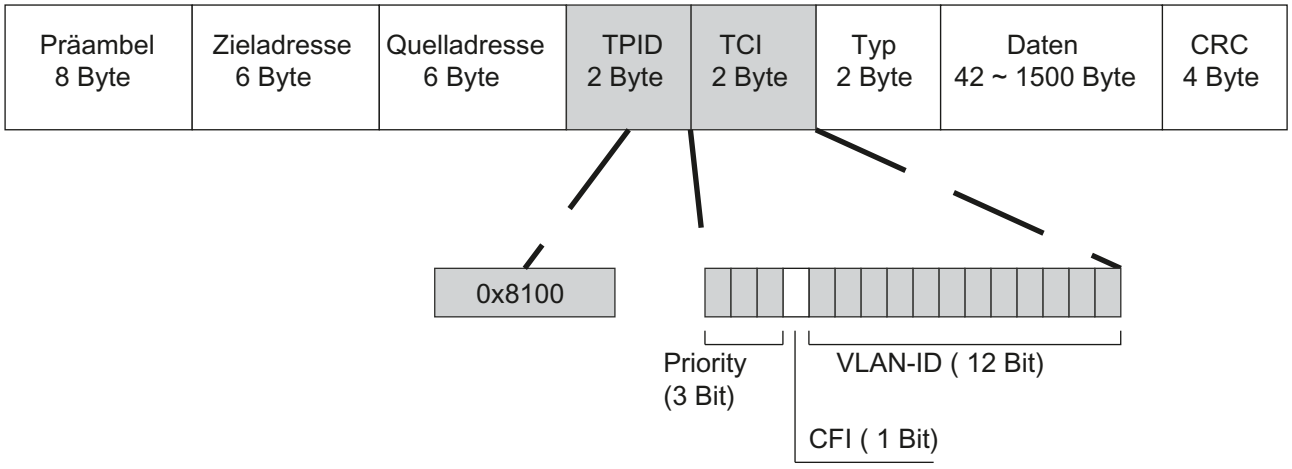

Bild 1-88 Aufbau des erweiterten Ethernet-Telegramms

Die zusätzlichen Bytes beinhalten den Tag Protocol Identifier (TPID) und die Tag Control Information (TCI).

Tag Protocol Identifier (TPID)

Die ersten 2 Bytes bilden den Tag Protocol Identifier (TPID) und sind fest mit 0x8100 belegt. Dieser Wert gibt an, dass das Datenpaket VLAN-Informationen oder Prioritätsangaben beinhaltet.

Tag Control Information (TCI)

Die 2 Bytes der Tag Control Information (TCI) beinhalten folgende Informationen:

CoS-Priorisierung

In dem getaggten Telegramm gibt es 3 Bits für die Priorität, die auch als Class of Service (CoS) bezeichnet werden, siehe auch IEEE 802.1Q.

CoS-Bits	Priorität	Art des Datenverkehrs
000	0 (niedrigste)	Background (Hintergrund)
001	1	Best Effort
010	2	Excellent Effort
011	3	Critical Applications (Kritische Anwendungen)
100	4	Video, < 100 ms Verzögerung (Latenz und Jitter)
101	5	Voice (Sprache), < 10 ms Verzögerung (Latenz und Jitter)
110	6	Internetwork Control
111	7 (höchste)	Network Control

Die Priorisierung der Datenpakete setzt eine Warteschlange in den Komponenten voraus, in der sie die Datenpakete mit der niedrigeren Priorität puffern können.

Das Gerät besitzt mehrere parallele Warteschlangen, in denen die verschieden priorisierten Telegramme abgearbeitet werden. Standardmäßig werden zuerst die Telegramme mit der höchsten Priorität abgearbeitet. Dieses Verfahren gewährleistet auch bei einem hohen Datenaufkommen, dass die Telegramme mit der höchsten Priorität auf jeden Fall gesendet werden.

Canonical Format Identifier (CFI)

Der CFI wird für die Kompatibilität zwischen Ethernet und Token Ring benötigt.
Die Werte haben folgende Bedeutung:

Wert	Bedeutung
0	Das Format der MAC-Adresse ist kanonisch. Bei kanonischer Darstellung der MAC-Adresse wird das niederwertigste Bit zuerst übertragen. Standardeinstellung für Ethernet-Switches.
1	Das Format der MAC-Adresse ist nicht kanonisch.

VLAN-ID

Im 12 Bit-Datenfeld können bis zu 4096 VLAN-IDs gebildet werden. Dabei gelten folgende Festlegungen:

VLAN-ID	Bedeutung
0	Das Telegramm beinhaltet nur Prioritätsinformation (Priority Tagged Frames) und keine gültige VLAN-Kennung.
1 - 4094	Gültige VLAN-Kennung, das Telegramm ist einem VLAN zugeordnet, es kann zusätzlich auch Prioritätsinformationen beinhalten.
4095	Reserviert

VLAN

Netzwerkdefinition unabhängig von der räumlichen Lage der Teilnehmer

VLAN (Virtual Local Area Network) teilt ein physikalisches Netzwerk in mehrere logische Netzwerke, die voneinander abgeschirmt sind. Hierbei werden Geräte zu logischen Gruppen zusammengefasst. Nur Teilnehmer des gleichen VLANs können sich untereinander adressieren. Da auch Multicast- und Broadcast-Telegramme nur innerhalb des jeweiligen VLANs weitergeleitet werden, wird von Broadcast-Domänen gesprochen.

Daraus ergibt sich als besonderer Vorteil von VLANs eine geringere Netzlast für die Teilnehmer bzw. Netzsegmente anderer VLANs.

Für die Kennung, welcher Frame welchem VLAN zugeordnet ist, wird der Frame um 4 Byte (VLAN-Tagging) erweitert. Diese Erweiterung enthält neben der VLAN-ID auch Prioritätsinformationen.

Möglichkeiten der VLAN-Zuordnung

Es gibt verschiedene Möglichkeiten der Zuordnung zu VLANs:

- Port-basiertes VLAN
  Jedem Port eines Geräts wird eine VLAN ID zugewiesen. Portbasiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN".
- Protokoll-basiertes VLAN
  Jedem Port eines Geräts wird eine Protokollgruppe zugewiesen.
- Subnetz-basiertes VLAN
  Der IP-Adresse des Geräts wird eine VLAN ID zugewiesen.

Doppel getaggter Frame (Q-in-Q)

Es gibt Geräte z. B. SCALANCE XR500, die die Funktion Q-in-Q unterstützen. Bei der Funktion Q-in-Q wird der eingehende Datenverkehr behandelt, als wäre er ungetagged. Bei bereits getaggten Frames ① bedeutet dies, dass sie um ein zweites VLAN-Tag erweitert werden, das äußere VLAN-Tag ②.

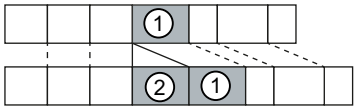

Wenn ein SCALANCE W-Gerät einen doppel getaggten Frame erhält, verwendet es die VLAN-ID aus dem äußeren VLAN-Tag ②. Der Frame wird dann an das entsprechende VLAN weitergeleitet.

MAC-basierte Kommunikation

Frames, die vom Client zum Access Point geschickt werden, enthalten als Source-MAC-Adresse immer die MAC-Adresse des WLAN-Clients. In der "Learning-Tabelle" des Access Points steht daher nur die MAC-Adresse des WLAN-Clients.

MAC Mode "Automatisch", "Manuell" und "Eigene"

Wenn die MAC-Adresse von einem Gerät adoptiert wird, das am Client angeschlossen ist (Automatisch), bzw. manuell eingestellt wird (Manuell), erreichen sowohl die MAC-basierten als auch die IP-basierten Frames für genau dieses Gerät ihr Ziel. Wenn die MAC-Adresse der Ethernet-Schnittstelle des WLAN-Clients verwendet wird (Eigene), erreichen die MAC-basierten und die IP-basierten Frames nur den WLAN-Client.

Der Access Point prüft, ob die Ziel-MAC-Adresse mit den MAC-Adressen der verbundenen Clients übereinstimmt. Da ein WLAN-Client nur eine MAC-Adresse verwenden kann, kann eine Kommunikation auf MAC-Adressen-Ebene (ISO/OSI-Schicht 2) maximal zu einem Teilnehmer hinter dem Client bzw. dem Client selbst erfolgen.

Durch IP Mapping können mehrere Teilnehmer hinter einem Client auf Basis des IP-Protokolls angesprochen werden. Die IP-Pakete werden über eine intern gehaltene Tabelle aufgeschlüsselt und an die angeschlossenen Geräte weitergeleitet.

Maximal mögliche Anzahl Ethernet-Teilnehmer mit Layer 2-Kommunikation hinter dem Client: 1

Hinweise zur Einstellung "Automatisch":

- Solange an der Ethernet-Schnittstelle kein Link besteht, nutzt das Gerät die MAC-Adresse der Ethernet-Schnittstelle, um in diesem Zustand ansprechbar zu sein. In diesem Zustand kann das Gerät über das Primary Setup Tool gefunden werden und über WBM oder CLI konfiguriert werden.
- Sobald an der Ethernet-Schnittstelle ein Link besteht, adoptiert das Gerät die Quell-MAC-Adresse des ersten empfangenen Telegramms.

Hinweis

Ab dem Zeitpunkt, an dem das Gerät eine fremde MAC-Adresse adoptiert hat (manuell oder automatisch), antwortet das Gerät nicht mehr auf Anfragen des Primary Setup Tools, wenn die Anfrage über die WLAN-Schnittstelle empfangen wird. Anfragen des PST über die Ethernet-Schnittstelle werden weiterhin beantwortet.

MAC Mode "Layer 2 Tunnel"

Der WLAN Client verwendet die MAC-Adresse der Ethernet-Schnittstelle für die WLAN-Schnittstelle.

Zusätzlich wird der Access Point über die an der Ethernet-Schnittstelle des WLAN-Clients angeschlossenen MAC-Adressen informiert. Damit ist der Eintrag der MAC-Adressen dieser

Geräte in die "Learning-Tabelle" des Access Points möglich. Der Access Point kann MAC-basierte Frames zu den Geräten hinter dem Client an den entsprechenden Client weiterleiten.

Dabei wird ähnlich wie bei WDS ein eigener Port für den L2T-Client angelegt, durch den die Ethernet-Frames ohne Änderung der Ziel-MAC-Adresse versendet werden.

Maximal mögliche Anzahl Ethernet-Teilnehmer hinter dem Client: 8

IEEE 802.11n

Überblick

Der Standard IEEE 802.11n ist eine Erweiterung des 802.11-Standards und wurde 2009 verabschiedet.
Bisherige Standards arbeiteten entweder im Frequenzband 2.4 GHz (IEEE 802.11g/b) oder im Frequenzband 5 GHz (IEEE 802.11a). IEEE 802.11n kann in beiden Frequenzbändern arbeiten.
Im Standard IEEE 802.11n sind Mechanismen in PHY- und MAC-Schichten implementiert, die den Datendurchsatz erhöhen und die Funkabdeckung verbessern.

- MIMO-Antennentechnik
- Maximum Ratio combining (MRC)
- Spatial Multiplexing
- Channel Bonding
- Frame-Aggregation
- Verkürztes Guard Intervall
- Modulation and Coding Scheme
- Datendurchsatzraten bis zu 450 Mbit/s (Brutto)
  Dies ist nicht bei allen SCALANCE W700-Geräten möglich.

MIMO-Antennentechnik

MIMO (Multiple Input - Multiple Output) basiert auf einem intelligenten Mehrantennensystem. Dabei haben der Sender und der Empfänger jeweils mehrere räumlich getrennte Antennen. Die räumlich getrennten Antennen strahlen die Datenströme zur gleichen Zeit aus. Es sind bis zu vier Datenströme möglich. Die Datenströme werden dabei räumlich verteilt ausgestrahlt und legen durch Beugung, Brechung, Fading und Reflexion (Mehrwegausbreitung) verschiedene Wege zurück. Die Mehrwegeausbreitung bewirkt, dass am Empfangsort ein komplexes, raum- und zeitabhängiges Muster als Summensignal der einzelnen Sendesignale entsteht. Dieses eindeutige Muster nutzt MIMO, indem es die in ihrer räumlichen Position charakteristischen Signale erfasst. Dabei unterscheidet sich jede Raumposition von der benachbarten. Mit der Charakterisierung der einzelnen Sender ist der Empfänger in Lage mehrere Signale voneinander zu trennen.

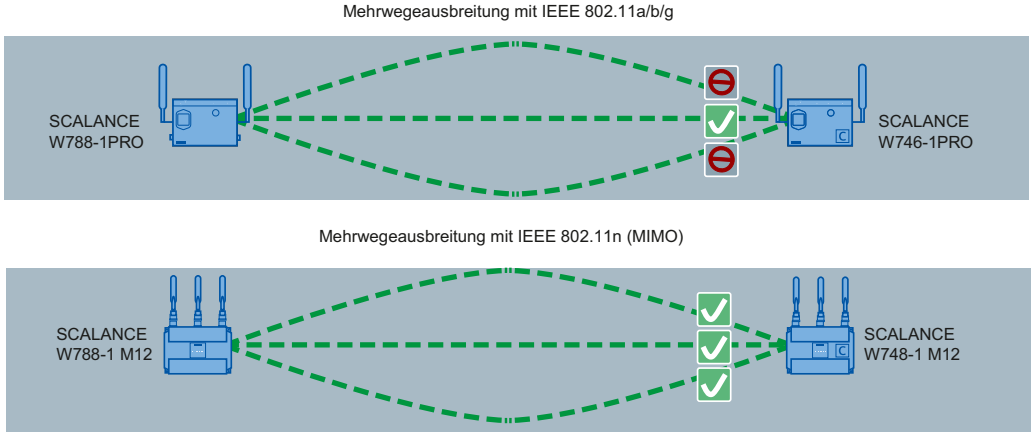

Maximum Ratio Combining (MRC)

Beim Mehrantennensystem werden die Funksignale von den einzelnen Antennen empfangen und zu einem Signal kombiniert. Zum Kombinieren der Funksignale wird das MRC-Verfahren verwendet. Das MRC-Verfahren gewichtet die Funksignale nach ihrem Signal-Rausch-Verhältnis und kombiniert die Funksignale zu einem Signal. Das Signal-Rausch-Verhältnis wird verbessert und die Fehlerrate wird verringert.

Spatial Mutliplexing

Beim räumlichen Multiplexen (Spatial Multiplexing) werden verschiedene Informationen über die gleiche Frequenz gesendet. Der Datenstrom wird auf n Sendeantennen verteilt, d. h. jede Antenne sendet nur 1/n des Datenstroms. Das Aufteilen des Datenstroms ist durch die Anzahl der Antennen beschränkt. Auf der Empfängerseite wird das Signal wieder zusammengesetzt. Durch das räumliche Multiplexen ergibt sich ein höheres Signal-Rausch-Verhältnis und ein höherer Datendurchsatz.

Channel Bonding

Bei IEEE 802.11n können Daten über zwei direkt benachbarte Kanäle übertragen werden. Die beiden 20 MHz Kanäle werden zu einem Kanal mit 40 MHz zusammengefasst. Dadurch lässt sich die Kanalbandbreite verdoppeln und der Datendurchsatz erhöhen.
Um die Kanalbündelung zu nutzen, muss der Empfänger 40 MHz-Übertragungen unterstützen. Wenn der Empfänger 40 MHz-Übertragungen nicht unterstützt, wird automatisch auf 20 MHz reduziert. Damit kann IEEE 802.11n auch mit IEEE 802.11a/b/g-Geräten kommunizieren.
Die Kanalbündelung stellen Sie auf der WBM-Seite "AP" mit dem Parameter "HT-Kanal-Bandbreite [MHz]" ein.

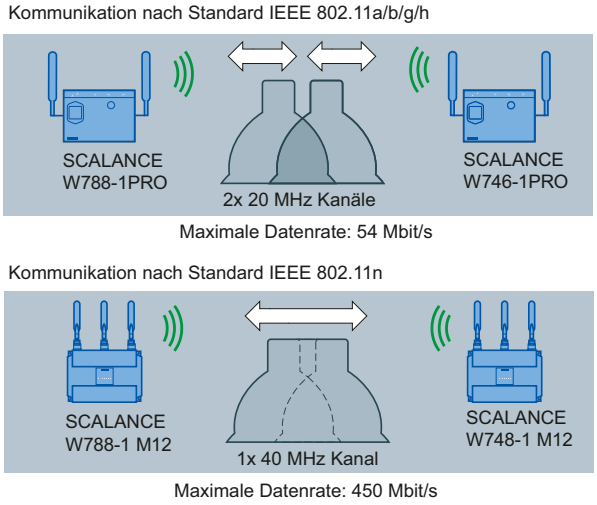

Frame Aggregation

Bei IEEE 802.11n ist es möglich Einzel-Datenpakete zu einem größeren Datenpaket zusammen zufassen, die so genannten Frame Aggregation. Es gibt zwei Arten der Frame Aggregation:

- Aggregated MAC Protocol Data Unit (A-MPDU)
  Bei A-MPDU werden mehrere MPDU-Datenpakete mit der gleichen Zieladresse zusammengefasst und als ein großes A-MPDU versendet.
- Aggregated Mac Service Data Unit (A-MSDU)
  Bei A-MSDU werden mehrere MSDU-Datenpakete der gleichen Zieladresse zusammen verkettet und versendet.

Die SCALANCE W-Geräte unterstützen beide Arten der Frame Aggregation. Die Einstellungen legen Sie auf der WBM-Seite "AP 802.11n" fest.

Verkürztes Guard-Intervall

Das Guard-Intervall verhindert, dass sich verschiedene Übertragungen vermischen. In der Nachrichtentechnik wird diese Vermischung auch als Intersymbolinterferenz (ISI) bezeichnet. Nach Ablauf der Sendezeit wird eine Sendepause (Guard Interval) eingelegt, bevor die nächste Übertragung beginnt.

Das Guard-Intervall von IEEE 802.11a/b/g beträgt 800 ns. IEEE 802.11n kann das verkürzte Guard-Intervall von 400 ns benutzen. Das Guard-Intervall legen Sie auf der WBM-Seite "AP 802.11n" fest.

Modulation and Coding Schemes

Der Standard IEEE 802.11n unterstützt unterschiedliche Datenraten. Die Datenraten basieren auf der Anzahl der Sender- und Empfangszüge (Spatial Streams), dem Modulationsverfahren und der Kanalkodierung. Die verschiedenen Kombinationen werden im Modulation and Coding Shemes beschrieben.

PROFINET

PROFINET

PROFINET ist ein offener Ethernet-Standard (IEC 61158/61784) für die industrielle Automatisierung basierend auf Industrial Ethernet. PROFINET nutzt existierende IT-Standards und ermöglicht eine durchgängige Kommunikation von der Feldebene bis in die Leitebene sowie ein anlagenweites Engineering. Weitere Eigenschaften von PROFINET sind:

- Nutzung von TCP/IP
- Automatisierung von Applikationen mit Echtzeit-Bedarf
  - Real-Time (RT)-Kommunikation
  - Isochronous Real-Time (IRT)-Kommunikation
- Nahtlose Integration von Feldbus-Systemen

PROFINET konfigurieren Sie unter "System > PROFINET (Seite 2248)".

PROFINET IO

Im Rahmen von PROFINET ist PROFINET IO ein Kommunikationskonzept für die Realisierung modularer, dezentraler Applikationen. Die Umsetzung von PROFINET IO wird durch den PROFINET-Standard für Automatisierungsgeräte (IEC 61158-x-10) realisiert.

EtherNet/IP

EtherNet/IP

EtherNet/IP (Ethernet/Industrial Protocol) ist ein offener Industriestandard für industrielles Echtzeit-Ethernet, basierend auf TCP/IP und UDP/IP. Mit EtherNet/IP wird Ethernet um das Common Industrial Protocol (CIP) auf der Anwendungsschicht erweitert. In EtherNet/IP werden die unteren Schichten des OSI-Referenzmodells von Ethernet mit den Übertragungs-, Vermittlungs-, Netzwerk- und Transportfunktionen übernommen.

EtherNet/IP konfigurieren Sie unter "System > EtherNet/IP (Seite 2250)".

Common Industrial Protocol

Das Common Industrial Protocol (CIP) ist ein Anwendungsprotokoll der Automatisierung, das den Übergang der Feldbusse in industrielles Ethernet und in IP-Netze unterstützt. Dieses Industrieprotokoll benutzen Feldbusse/Industrienetzwerke wie DeviceNet, ControlNet und EtherNet/IP in der Anwendungsschicht als Schnittstelle zwischen der deterministischen Feldbus-Welt und der Automatisierungsapplikation (Steuerung, E/A, HMI, OPC, ...). Das CIP liegt oberhalb der Transportschicht und erweitert die reinen Transportdienste um Kommunikationsdienste für die Automatisierungstechnik. Dazu gehören Dienste für den zyklischen, den zeitkritischen und den ereignisgesteuerten Datenverkehr. CIP unterscheidet zwischen den zeitkritischen E/A-Nachrichten (implicit messages) und individuellen Frage/Antwort-Telegrammen zur Konfiguration und Datenerfassung (explicit messages). CIP ist objekt-orientiert; alle von außen "sichtbaren" Daten sind in Form von Objekten zugänglich. CIP hat eine gemeinsame Konfigurationsgrundlage: EDS (Electronic Data Sheet).

Electronic Data Sheet

Electronic Data Sheet (EDS) ist ein elektronisches Datenblatt zur Beschreibung von Geräten.

Das für den EtherNet/IP-Betrieb benötigte EDS finden Sie unter "System > Laden & Speichern".

NAT / NAPT

Was ist NAT?

Mit Network Address Translation (NAT) wird die IPv4-Adresse in einem Datenpaket durch eine andere ersetzt. NAT wird in der Regel an einem Netzübergang zwischen einem privaten LAN und einem externen Netz mit global gültigen IPv4-Adressen eingesetzt. Dabei wird eine lokale IPv4-Adresse des internen LAN von einem NAT-Gerät am Netzübergang in eine externe, globale IPv4-Adresse umgeschrieben.

Zur Übersetzung der internen in die globale IPv4-Adresse pflegt das NAT-Gerät eine Übersetzungsliste. Die Adresszuordnung erfolgt automatisch. Die Adresszuordnung für NAT konfigurieren Sie unter "Layer 3 > NAT > Basic".

Was ist NAPT?

Bei "Network Address Port Translation" (NAPT) oder "Port Address Translation" (PAT) werden mehrere interne Quell-IPv4-Adressen in die gleiche externe Quell-IPv4-Adresse umgeschrieben. Zur Identifikation der einzelnen Quell-Teilnehmer wird auch der Port des Quellgeräts in der Übersetzungsliste des NAT-Gateways gespeichert und für die externe Adresse umgeschrieben.

Wenn mehrere lokale Clients über das NAT-Gateway eine Anfrage an die gleiche externe Ziel-IPv4-Adresse senden, trägt das Gateway jeweils seine eigene externe Quell-IPv4-Adresse in den Header dieser weitergeleiteten Datenpakete ein. Da die weitergeleiteten Datenpakete die

gleiche globale Quell-IPv4-Adresse haben, ordnet das NAT-Gateway die Datenpakete den Clients über eine unterschiedliche Portnummer zu.

Hinweis

NAT/NATP ist nur auf Schicht 3 des ISO/OSI-Referenzmodells möglich. Zur Nutzung der NAT-Funktion müssen die Netze das IP-Protokoll verwenden.

Bei Verwendung des ISO-Protokolls, das auf Schicht 2 arbeitet, ist keine Nutzung von NAT möglich.

Wenn ein Client aus dem globalen Netz einen Dienst im internen Netz nutzen möchte, muss die Übersetzungsliste für die statische Adresszuordnung konfiguriert werden. Die Übersetzungsliste für NAPT konfigurieren Sie unter "Layer 3 > NAT > NAPT".

iPCF / iPCF-HT / iPCF-MC

Der Funkbereich eines IWLAN-Systems kann durch den Einsatz mehrerer Access Points erweitert werden. Wird ein Client aus dem Bereich eines Access Points in den Bereich eines anderen Access Points verschoben, bleibt die Funkverbindung nach kurzer Unterbrechung bestehen (Roaming).

Werden sehr schnelle Aktualisierungszeiten benötigt, z. B. für PROFINET-Kommunikation, sind Access Points und Client-Module einzusetzen, die die proprietäre Verfahren iPCF / iPCF-HT oder iPCF-MC für schnelles Roaming und deterministischen Datenverkehr verwenden.

iPCF / iPCF-HT / iPCF-MC können nur allein betrieben werden. Eine Kombination miteinander ist nicht möglich, z. B. iPCF mit iPCF-HT oder iPCF-MC.

Funktionsprinzip

iPCF

Bei iPCF prüft der Access Point alle Teilnehmer in der Funkzelle zyklisch ab. Die Abfrage beinhaltet gleichzeitig den Downlink-Verkehr für diesen Teilnehmer. In der Antwort sendet der Teilnehmer die Uplink-Daten. Der Access Point fragt mindestens alle 5 ms einen neuen Teilnehmer an.

Die Abfrage eines Teilnehmers wird von allen anderen Teilnehmern in der Funkzelle gesehen. So kann ein Client die Qualität der Funkverbindung zum Access Point auch dann feststellen, wenn er selbst nicht mit dem Access Point kommuniziert. Empfängt der Client für eine bestimmte Zeit keine Telegramme vom Access Point, beginnt er mit der Suche nach einem neuen Access Point.

Im iPCF-Modus ist sowohl die Suche nach einem neuen Access Point als auch das Anmelden an diesem Access Point in ihrem Zeitverhalten optimiert. Es werden Handover-Zeiten von deutlich unter 50 ms erreicht.

Für eine stabile PROFINET-Kommunikation ist erforderlich, dass sich ein WLAN-Client zu jeder Zeit in einer Funkzelle mit einer Signalstärke > 60 % bzw. > –65 dBm befindet. Dies kann durch das An- und Abschalten der verschiedenen Funkzellen überprüft werden.

Das bedeutet nicht, dass der Client bei einer Signalstärke < 60 % bzw. < –65 dBm wechseln muss. Stellen Sie sicher, dass Access Points mit ausreichender Signalstärke zur Verfügung stehen.

iPCF konfigurieren Sie unter "iFeatures > iPCF > iPCF".

iPCF-HT

Wenn für iPCF ein höherer Datendurchsatz benötigt wird, wird iPCF-HT eingesetzt. Damit können Sie z. B. neben PROFINET noch Video-Daten übertragen. Das wird erzielt, in dem einzelne Datenpakete zu einem großen Datenpaket mittels Frame-Bursting (A-MPDU) effektiver übertragen werden. Es werden die Einzel-Datenpakete zusammengefasst, die für dieselbe Empfänger-Station (Client) vorgesehen sind und die gleiche Priorisierung besitzen.

iPCF-HT konfigurieren Sie unter "iFeatures > iPCF > iPCF-HT".

iPCF-MC

Für frei bewegliche Teilnehmer, die unabhängig von einer RCoax-Leitung oder gerichteten Antennen kommunizieren, sollte iPCF-MC verwendet werden. Bei iPCF-MC sucht der Client auch dann nach potenziell geeigneten Access Points, wenn er iPCF-Abfragen des Access Points erhält und die bestehende Verbindung zu einem Access Point störungsfrei funktioniert. Dadurch kann im Bedarfsfall der Wechsel zu einem anderen Access Point sehr schnell erfolgen. Im Gegensatz zu iPCF sind bei iPCF-MC die Handover-Zeiten unabhängig von der Anzahl der verwendeten Funkkanäle.

Dabei ist es notwendig, einen Access Point mit zwei Funkschnittstellen zu verwenden, einen so genannten Dual Access Point. Die eine Schnittstelle arbeitet als Management Channel und sendet kurze Telegramme („Beacon") mit administrativen Informationen (z. B. Kanaleinstellung des Data Channels und SSID). Die andere Schnittstelle (Data Channel) überträgt ausschließlich die Nutzdaten.

iPCF-MC konfigurieren Sie unter "iFeatures > iPCF > iPCF-MC".

Die folgende Grafik zeigt ein Konfigurationsbeispiel für iPCF-MC.

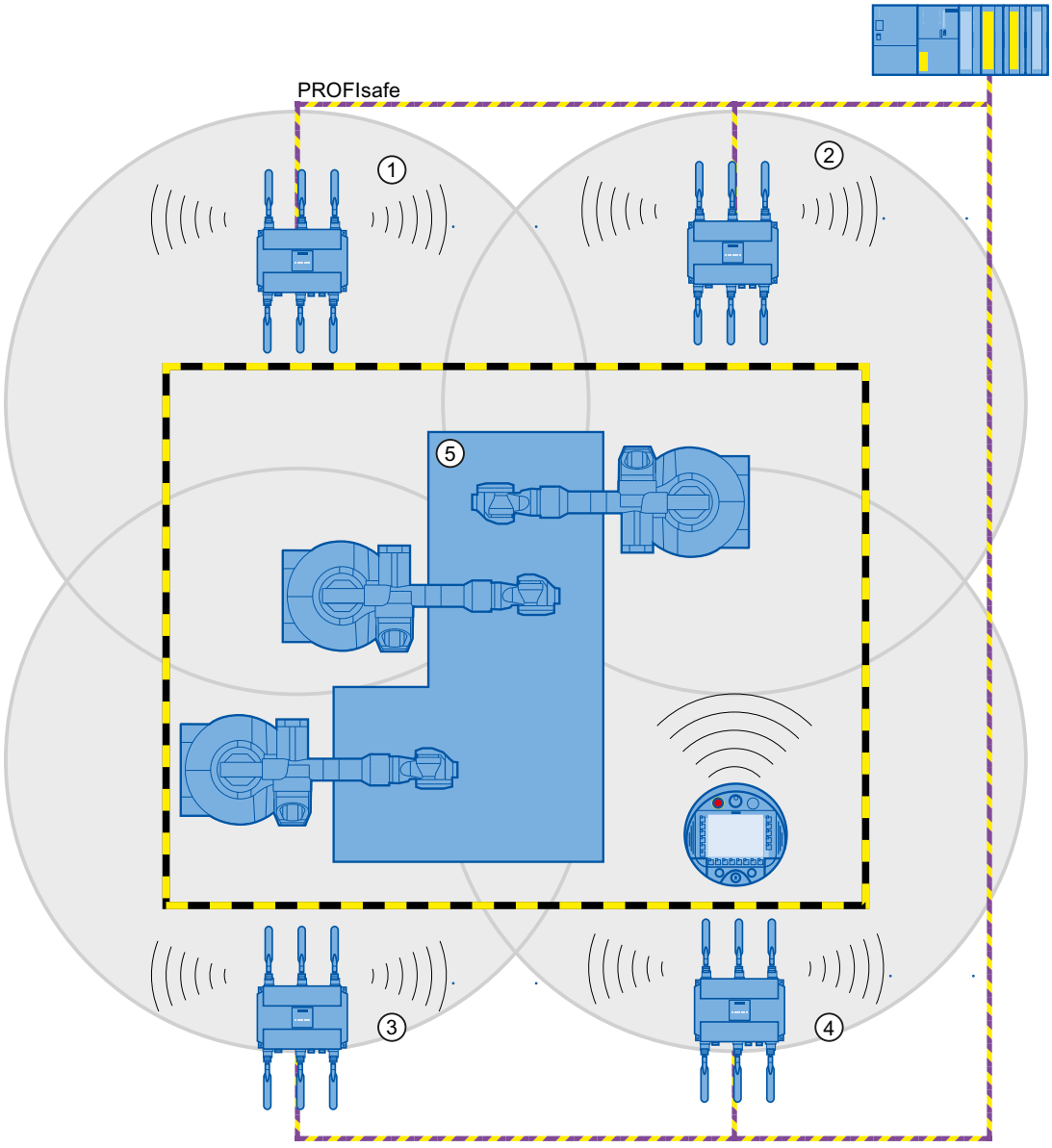

① Funkzelle des Access Point 1
② Funkzelle des Access Point 2
③ Funkzelle des Access Point 3
④ Funkzelle des Access Point 4
⑤ Anlage

Einschränkungen

- iPCF / iPCF-HT und iPCF-MC sind Eigenentwicklungen der Fa. Siemens AG und arbeiten nur mit Teilnehmern zusammen, bei denen iPCF / iPCFv2 / iPCF-MC implementiert ist.
- Bei einem Access Point mit mehreren WLAN-Schnittstellen ist es möglich, sowohl iPCF / iPCF-HT als auch Standard-WLAN gleichzeitig einzustellen.

- Access Points mit einer WLAN-Schnittstelle können nicht am iPCF-MC-Verfahren teilnehmen. iPCF ist aber möglich.
- iPCF-HT steht nur auf der WLAN-Schnittstelle 1 zur Verfügung und kann ausschließlich im 5 GHz-Band mit WLAN-Modus "(Nur) IEEE 802.11n" verwendet werden.

Voraussetzungen für iPCF-MC

iPCF-MC nutzt die beiden Funkschnittstellen des Access Point unterschiedlich: Die eine Schnittstelle arbeitet als Management-Schnittstelle und sendet alle fünf Millisekunden ein Beacon. Die andere Schnittstelle überträgt die Nutzdaten.

Folgende Voraussetzungen müssen erfüllt sein, um iPCF-MC nutzen zu können:

- Als Access Points können nur SCALANCE W700-Geräte mit zwei WLAN-Schnittstellen eingesetzt werden
- Die Daten-Schnittstelle (WLAN1) und die Management-Schnittstelle (WLAN2) müssen im gleichen Frequenzband betrieben werden und hinsichtlich ihrer Funkabdeckung übereinstimmen. iPCF-MC wird nicht funktionieren, wenn beide Funkschnittstellen mit Richtantennen ausgestattet sind, die unterschiedliche Bereiche abdecken.
- Die Management-Schnittstellen von allen Access Points, zwischen denen ein Client wechseln soll, müssen den gleichen Kanal verwenden. Ein Client scannt nur diesen einen Kanal, um erreichbare Access Points zu finden.
- Für die Management-Schnittstelle kann nicht das Übertragungsverfahren nach IEEE 802.11h (DFS) verwendet werden. Für die Datenschnittstelle ist 802.11h (DFS) möglich.
- Ein Client muss dieses Feature auf seiner WLAN-Schnittstelle unterstützen.

Siehe auch

iPCF (Seite 2677)

iPCF-MC (Seite 2682)

iREF

Funktionsweise

Wenn ein Access Point mehrere aktivierte Antennen hat, verteilt sich die Sendeleistung zu gleichen Teilen auf diese Antennen. Die Sendeleistung unterliegt hierbei länderspezifischen gesetzlichen Beschränkungen. Die maximal zulässige Leistung hängt vom Gewinn der angeschlossenen Antennen ab. Weisen die angeschlossenen Antennen unterschiedliche Gewinne auf, beschränkt maßgeblich der maximale Antennengewinn die zulässige Sendeleistung.

iREF (industrial Range Extension Function) sorgt dafür, dass der Datenverkehr vom Access Point zu jedem einzelnen Client über die jeweils am besten geeignete Antenne abläuft. Welche Antenne am Besten geeignet ist, wird vom Access Point auf Basis der RSSI-Werte empfangener Pakete ermittelt.

Unter Einbeziehung von Antennengewinn und ggf. Kabelverlusten werden Pakete jeweils nur auf denjenigen Antennen gesendet, für die clientseitig die maximale Signalstärke zu erwarten ist.

In dieser Zeit sind die anderen Antennen inaktiv und die gesetzlich zulässige Sendeleistung steht für die ausgewählte Antenne zur Verfügung. Die inaktiven Antennen wirken sich nicht beschränkend auf die zulässige Sendeleistung aus.

Damit können die Daten, insbesondere in Anwendungsfällen, bei denen MIMO nicht anwendbar oder nicht von Vorteil ist, mit der größtmöglichen Datenübertragungsrate gesendet werden.

iREF konfigurieren Sie unter "iFeatures > iREF"

ohne iREF

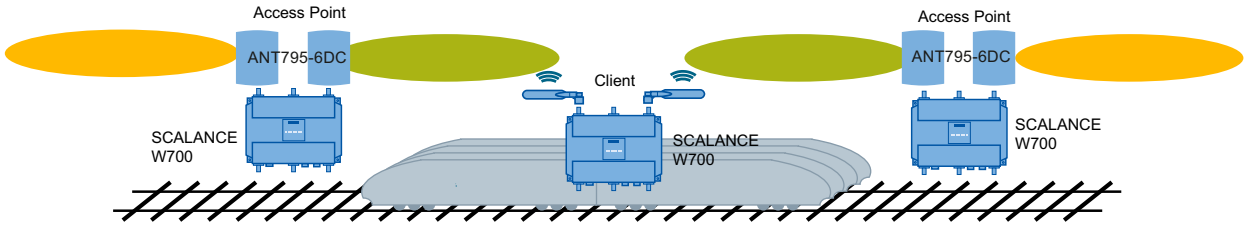

mit iREF

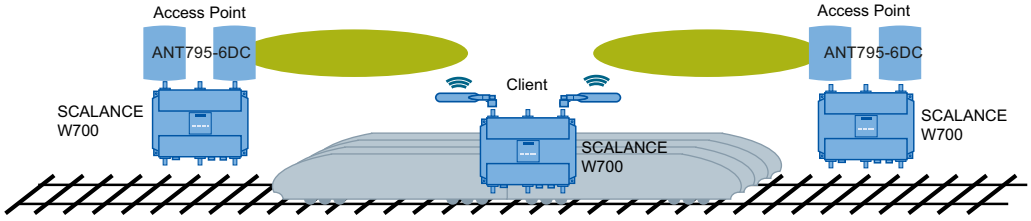

Voraussetzung

- Um iREF zu verwenden muss das SCALANCE W700-Gerät über mindestens 2 aktivierte Antennen verfügen.

Einschränkungen

- Es ist nur eine maximale Datenrate von bis zu 150 Mbps (MCS 0 - 7 oder 1x Spatial Stream) möglich
- iREF kann nicht zusammen mit anderen iFeatures benutzt werden (z.B. iPCF oder iPCF-MC)

Vorteile

- Durch die gerichtete Datenübertragung und dynamische Deaktivierung von Antennen, die nicht in Richtung des jeweiligen Clients strahlen, lassen sich Interferenzen reduzieren.
- Die Signalstärke wird verbessert, da der aktiven Antenne immer die maximal zulässige Sendeleistung zur Verfügung steht.

iPRP

Das "Parallel Redundancy Protocol" (PRP) ist ein Redundanzprotokoll für kabelgebundene Netzwerke. Es ist im Teil 3 des IEC 62439 Standards definiert.

Mit "industrial Parallel Redundancy Protocol" (iPRP) kann die PRP-Technologie in drahtlosen Netzwerken eingesetzt werden. Damit wird die Verfügbarkeit der drahtlosen Kommunikation verbessert.

Funktionsprinzip

Ein PRP-Netzwerk besteht aus zwei völlig unabhängigen Netzwerken. Wenn eines der Netzwerke gestört ist, werden die Frames ohne Unterbrechung/Rekonfiguration über das parallele redundante Netzwerk gesendet. Dazu werden die Ethernet-Frames verdoppelt und über beide Netzwerke an den Empfänger übertragen. PRP-fähige Geräte haben jeweils mindestens zwei voneinander getrennte Ethernet-Schnittstellen, die an jeweils unabhängige Netzwerke angeschlossen sind.

Bei nicht PRP-fähigen Geräten wird eine Redundancy-Box (RedBox) vorgeschaltet. Diese schafft den Zugang für so genannte Single Attached Nodes (SAN) zu den PRP-Netzen. Die RedBox verdoppelt jeden zu sendenden Ethernet-Frame und fügt einen PRP-Trailer an den Frame, der u.a. eine Sequenznummer enthält. Die RedBox sendet gleichzeitig jeweils eine Kopie des Frames in das PRP A und PRP B Netzwerk. Auf der Empfängerseite wird von der RedBox das doppelte Frame verworfen. Dafür erfordert die RedBox bestimmte Übertragungszeiten, die für Ethernet Netzwerke ausgelegt sind. Aus diesem Grund kommt es bei der Verwendung von PRP in WLAN Netzen zu doppelten und verzögerten Frames.

Durch iPRP wird diese Problematik aufgelöst und die Verwendung von PRP in WLAN mit SCALANCE W700-Geräten ermöglicht

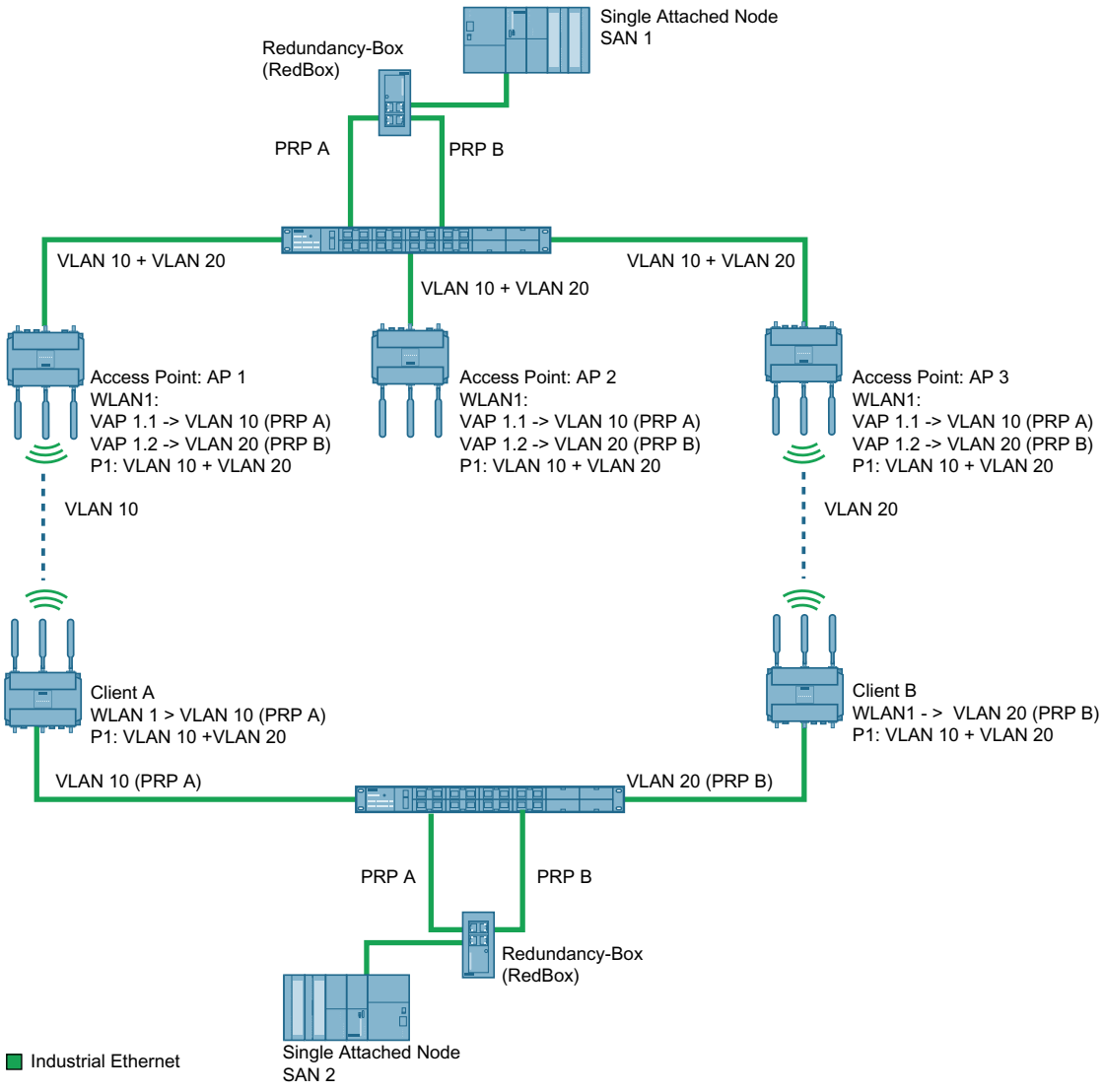

Die Access Points (AP 1, AP 2 und AP 3) und die RedBox auf der AP Seite sind über einen Switch miteinander verbunden. PRP Netzwerk A und B sind über VLANs voneinander getrennt.

Sendet SAN1 einen Frame an SAN2, wird der Frame von der RedBox auf der AP-Seite dupliziert und die beiden redundanten Frames werden über den Switch an die Access Points übertragen. Über die zwei unterschiedlichen Funkstrecken werden die redundanten PRP-Frames an die RedBox auf der Client-Seite übermittelt. Die Clients sind ebenfalls über einen Switch mit ihrer RedBox verbunden. Diese leitet den ersten ankommenden PRP-Frame weiter an SAN2 und verwirft den Zweiten.

Hinweis

An den Schnittstellen der Switche zu den SCALANCE W700-Geräten dürfen jeweils nur die VLANs konfiguriert werden, die auch an den VAP- bzw. WLAN-Schnittstellen der SCALANCE W700-Geräte eingestellt sind.

Bei iPRP kommunizieren die redundanten Partner (hier: AP1 und AP3 bzw. Client A und Client B) über einen Switch miteinander, um zu verhindern, dass die beiden redundanten PRP-Frames mit einem zu großen Zeitunterschied bei der RedBox ankommen.

Ist zum Beispiel die Kommunikation zwischen AP1 und Client A sehr langsam, wird das langsamere Frame auf der Empfangsseite verworfen.

iPRP konfigurieren Sie unter "iFeatures > iPRP (Seite 2684)".

Voraussetzung

- Der Base Bridge-Modus "802.1Q VLAN Bridge" ist eingestellt.
- Die VLANs sind angelegt.
- Access Point-Modus: Die VAP-Schnittstelle ist aktiviert.
- Client-Modus: Bei MAC-Modus ist "Layer-2-Tunnel" eingestellt.
- Je nach Konfiguration können die Clients mit jedem Access Point kommunizieren.

AeroScout

AeroScout Tags

SCALANCE W700-Geräte unterstützen Tags der Firma AeroScout. Tags sind batteriebetriebene RFID-Sensoren, die zyklisch ihre Daten als Mutlicast-Frames versenden.

AeroScout Tags verfügen unter anderem über folgende Merkmale:

- **Umgebungstemperatur**
  Ist ein Tag an einem SCALANCE W700-Gerät oder an Material angebracht, kann überwacht werden, ob eine vorgeschriebene Umgebungstemperatur eingehalten wird.
- **Bewegung**
  Damit kann ein Tag auch die Information liefern, ob er sich in Bewegung oder in Ruhe befindet. Die Bereiche Materialfluss und Handhabungstechnik bieten mögliche Einsatzfälle für diese Funktion.
- **Taster**
  Unabhängig von den zyklisch versendeten Frames kann ein Benutzer auch durch einen Tastendruck eine Benachrichtigung versenden.
- **Leuchtdiode**
  Sie liefert Informationen über den Betriebszustand des Tags.

Hinweis

Weiterführende Informationen entnehmen Sie bitte der Dokumentation der Firma AeroScout (www.aeroscout.com).

Funktionsweise

Der Tag versendet seine Daten als AeroScout-Frames. Die Tags und die Access Points kommunizieren im 2,4 GHz-Band.

Empfängt die WLAN-Schnittstelle des Access Points den AeroScout-Frame, wird dieses in ein UDP-Telegramm umgewandelt. Das SCALANCE W700-Gerät leitet das UDP-Telegramm zusammen mit der Information über die Signalstärke (RSSI) an einen PC weiter. Auf dem PC läuft die AeroScout Engine, die die enthaltenen Informationen auswertet.

Hinweis

Es ist **nicht** empfehlenswert, PROFINET-Kommunikation und AeroScout zusammen auf einer Funkschnittstelle zu betreiben.

Lokalisierungsgenauigkeit

Um eine optimale Genauigkeit bei der Lokalisierung von AeroScout Tags zu erreichen,

- empfehlen wir den Einsatz von Antennen mit omnidirektionaler Charakteristik
- sollte das Signal von mindestens drei Access Points empfangen werden.

SNMP

Einleitung

Mit Hilfe des Simple Network Management Protocol (SNMP) überwachen und steuern Sie Netzwerkkomponenten, z. B. Router oder Switches, von einer zentralen Station aus. SNMP regelt dabei die Kommunikation zwischen den überwachten Geräten und der Überwachungsstation.

Aufgaben von SNMP:

- Überwachung von Netzwerkkomponenten
- Fernsteuerung und Fernparametrierung von Netzwerkkomponenten
- Fehlererkennung und Fehlerbenachrichtigung

In den Versionen v1 und v2c verfügt SNMP über keine Sicherheitsmechanismen. Jeder Nutzer im Netzwerk kann mit geeigneter Software auf die Daten zugreifen und auch Parametrierungen verändern.

Für die einfache Steuerung von Zugriffsrechten ohne Sicherheitsaspekte werden Community-Strings verwendet.

Der Community-String wird zusammen mit der Anfrage übertragen. Wenn der Community-String korrekt ist, antwortet der SNMP-Agent und sendet die geforderten Daten. Wenn der Community-String nicht korrekt ist, verwirft der SNMP-Agent die Anfrage. Für Lese- und Schreibrechte definieren Sie verschiedene Community-Strings. Die Community-Strings werden in Klartext übertragen.

Standardwerte der Community-Strings:

- public
  besitzt nur Leserechte
- private
  besitzt Lese- und Schreibrechte

Hinweis

Da es sich bei den SNMP-Community Strings um einen Zugriffsschutz handelt, verwenden Sie nicht die Standardwerte "public" oder "private". Ändern Sie diese Werte nach der Erst-Inbetriebnahme.

Weitere einfache Schutzmechanismen auf Geräteebene:

- Allowed Host
  Dem überwachten System sind die IP-Adressen der überwachenden Systeme bekannt.
- Read Only
  Wenn Sie einem überwachten Gerät "Read Only" zuweisen, können Überwachungsstationen nur Daten auslesen, aber nicht ändern.

SNMP-Datenpakete sind nicht verschlüsselt und können einfach mitgelesen werden.

Die zentrale Station wird auch als Management-Station bezeichnet. Auf den zu überwachenden Geräten ist ein SNMP-Agent installiert, mit dem die Management-Station Daten austauscht.

Die Management-Station sendet Datenpakete folgenden Typs:

- GET
  Anfordern eines Datensatzes vom SNMP-Agent
- GETNEXT
  Ruft den nächsten Datensatz auf.
- GETBULK (verfügbar ab SNMPv2c)
  Fordert mehrere Datensätze auf einmal an, z. B. mehrere Zeilen einer Tabelle.
- SET
  Beinhaltet Parametrierungsdaten für das entsprechende Gerät.

Der SNMP-Agent sendet Datenpakete folgenden Typs:

- RESPONSE
  Der SNMP-Agent sendet die vom Manager angeforderten Daten zurück.
- TRAP
  Wenn ein bestimmtes Ereignis eintritt, sendet der SNMP-Agent eigenständig Traps.

SNMPv1/v2c/v3 verwenden UDP (User Datagram Protocol) und nutzen die UDP-Ports 161 und 162. Die Beschreibung der Daten erfolgt in einer Management Information Base (MIB).

SNMPv3

SNMPv3 führt gegenüber den Vorgängerversionen SNMPv1 und SNMPv2c ein umfangreicheres Sicherheitskonzept ein.

SNMPv3 unterstützt:

- Vollständig verschlüsselte Benutzerauthentifizierung
- Verschlüsselung des gesamten Datenverkehrs
- Zugriffskontrolle der MIB-Objekte auf Benutzer-/Gruppenebene

Mit der Einführung von SNMPv3 können Sie Benutzerkonfigurationen nicht mehr ohne Weiteres auf andere Geräte übertragen, z. B. indem Sie eine Konfigurationsdatei laden oder den C-PLUG austauschen.

Das SNMPv3-Protokoll verwendet gemäß des Standards eine eindeutige SNMP-Engine-ID als internen Bezeichner für einen SNMP-Agenten. Diese ID muss im Netzwerk eindeutig sein. Sie wird verwendet, um die Zugangsdaten von SNMPv3-Benutzern zu authentifizieren und zu verschlüsseln.

Abhängig davon, ob Sie die Funktion "SNMPv3 Benutzermigration" aktiviert oder deaktiviert haben, wird die SNMP-Engine-ID unterschiedlich generiert.

Einschränkung bei der Verwendung der Funktion

Verwenden Sie die Funktion "SNMPv3 Benutzermigration" nur, um im Ersatzteilfall Ihre konfigurierten SNMPv3-Benutzer auf ein Ersatzgerät zu übertragen.
Verwenden Sie die Funktion nicht, um konfigurierte SNMPv3-Benutzer auf mehrere Geräte zu übertragen. Wenn Sie eine Konfiguration mit angelegten SNMPv3-Benutzern in mehrere Geräte laden, verwenden diese Geräte hierdurch die gleiche SNMP-Engine-ID. Wenn Sie diese Geräte im gleichen Netzwerk verwenden, widerspricht Ihre Konfiguration dem SNMP-Standard.

Kompatibilität mit Vorgängerprodukten

Sie können SNMPv3-Benutzer nur auf ein anderes Gerät übertragen, wenn Sie die Benutzer als migrierbare Benutzer erstellt haben. Um einen migrierbaren Benutzer zu erstellen, muss die Funktion "SNMPv3 Benutzermigration" aktiviert sein, wenn Sie den Benutzer erstellen.

Spanning Tree

Vermeidung von Schleifenbildung

Das Spanning Tree-Verfahren ermittelt physikalisch redundante Netzwerkstrukturen und verhindert Schleifenbildung durch Abschalten redundanter Wege. Dazu wertet es die Entfernung, die Leistungsfähigkeit einer Verbindung oder auch Benutzervorgaben aus. Der Datenverkehr erfolgt dann ausschließlich auf den verbleibenden Verbindungswegen.

Wenn der bevorzugte Datenweg ausfällt, sucht der Spanning Tree-Algorithmus den effizientesten Weg, der mit den verbliebenen Netzteilnehmern möglich ist.

Root Bridge und Bridge Priority

Die Ermittlung der effizientesten Verbindung erfolgt immer im Bezug auf die so genannte "Root Bridge", eine Netzkomponente, die als Wurzelelement einer baumartigen Netzstruktur angesehen wird. Mit dem Parameter "Bridge Priority" können Sie die Wahl der Root Bridge beeinflussen. Der Rechner mit dem geringsten Wert für diesen Parameter wird zwangsläufig Root-Bridge. Wenn zwei Rechner den gleichen Prioritätswert haben, wird derjenige Rechner Root-Bridge, der die niedrigere MAC-Adresse hat.

Verhalten bei Veränderungen der Netztopologie

Wenn Teilnehmer zu einem Netz hinzukommen oder wegfallen hat das möglicherweise Auswirkungen auf die optimale Wegewahl der Datenpakete. Um diese Änderungen zu berücksichtigen, versendet die Root Bridge in regelmäßigen Abständen Konfigurationsmeldungen (BPDUs). Den Zeitabstand zwischen zwei Konfigurationsmeldungen können Sie mit dem Parameter "Hello Time" einstellen.

Aktualität der Konfigurationsinformation

Mit dem Parameter "Max Age" legen Sie das maximale Alter von Konfigurationsinformationen fest. Erhält eine Bridge Konfigurationsinformationen, die älter sind, als in Max Age festgelegt, verwirft Sie diese Meldung und veranlasst eine Neuberechnung der Wege.

Neue Konfigurationsinformationen werden von einer Bridge jedoch nicht sofort, sondern erst nach dem im Parameter "Forward Delay" festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.

RSTP, MSTP, CIST

Rapid Spanning Tree Protocol (RSTP)

Ein Nachteil des STP ist, dass sich das Netz bei einer Störung oder einem Geräteausfall rekonfigurieren muss: Die Geräte beginnen erst im Moment der Unterbrechung, neue Pfade auszuhandeln. Dieser Vorgang dauert bis zu 30 Sekunden. Aus diesem Grunde wurde STP zum "Rapid Spanning Tree Protocol" (RSTP, IEEE 802.1w) erweitert. Dies unterscheidet sich vom STP im Wesentlichen dadurch, dass die Geräte bereits zum Zeitpunkt des ungestörten Betriebs Informationen über Alternativrouten sammeln, die sie sich dann nicht erst beschaffen müssen, wenn eine Störung eingetreten ist. Damit lässt sich die Rekonfigurationszeit für ein RSTP-gesteuertes Netz auf wenige Sekunden reduzieren.
Das wird durch folgende Funktionen erreicht:

- Edge-Ports (Endteilnehmer-Port)
  Edge-Ports sind Ports, die mit einem Endgerät verbunden sind.
  Ein Port, der als Edge-Port definiert ist, wird direkt nach einem Verbindungsaufbau aktiviert. Wenn an einem Edge-Port eine Spanning Tree-BPDU empfangen wird, verliert der Port die Rolle als Edge-Port und nimmt wieder am (R)STP teil. Wird nach Ablauf einer Zeitspanne (3x Hello-Time) kein BPDU-Telegramm mehr empfangen, geht der Port wieder in den Edge-Port-Status über.
- Punkt-zu-Punkt (direkte Kommunikation zweier benachbarter Geräte)
  Durch die direkte Kopplung der Geräte kann eine Zustandsänderung (Umkonfiguration der Ports) ohne Verzögerungen durchgeführt werden.
- Alternativ-Port (Ersatz für den Root-Port)
  Es ist ein Ersatz für den Root-Port konfiguriert. Bei einem Verbindungsverlust zur Root-Bridge kann das Gerät deshalb ohne Verzögerung durch Neukonfiguration eine Verbindung über den Alternativ-Port aufbauen.

- Reaktion auf Ereignisse
  Ein Rapid Spanning Tree reagiert auf Ereignisse, beispielsweise einen Verbindungsabbruch, ohne Verzögerung. Es müssen also keine Zeitgeber wie beim Spanning Tree abgewartet werden.
- Zähler maximale Bridge-Sprünge
  Anzahl der Bridge-Sprünge, die ein Paket maximal ausführen darf, bevor es automatisch ungültig wird.

Prinzipiell werden also beim Rapid Spanning Tree für viele Parameter Alternativen vorkonfiguriert oder bestimmte Eigenschaften der Netzstruktur berücksichtigt, um die Rekonfigurationszeit zu verkürzen.

Multiple Spanning Tree Protocol (MSTP)

Das Multiple Spanning Tree Protocol (MSTP) ist eine Weiterentwicklung des Rapid Spanning Tree Protocols. Es bietet u. a. die Möglichkeit, mehrere RSTP-Instanzen innerhalb verschiedener VLANs oder VLAN-Gruppen zu betreiben und so z. B. Pfade, die das einfache Rapid Spanning Tree Protocol für den Datenverkehr global sperren würde, innerhalb einzelner VLANs verfügbar zu machen.

Common and Internal Spanning Tree (CIST)

CIST bezeichnet die intern vom Switch verwendete Instanz, die im Prinzip einer internen RSTP-Instanz gleicht.

SCALANCE-Gerät als IO-Device konfigurieren

Bei einem SCALANCE-Gerät, das als PROFINET IO-Device projektiert und einem IO-Controller zugeordnet ist, werden bei den Funktionen "Übersetzen" und "Laden in Gerät" nur diejenigen Daten in das SCALANCE-Gerät geladen, die auch im Web Based Management (WBM) projektierbar sind (System, Layer 2, Layer 3, Security).

Wenn Sie die Funktionen "Übersetzen" oder "Laden in Gerät" für die PROFINET IO-Device-Daten des Geräts ausführen wollen, dann selektieren Sie zuvor den zugeordneten IO-Controller.

IP-Adressvergabe

Konfigurationsmöglichkeiten

Im Auslieferungszustand und nach dem Zurücksetzen auf Werkseinstellungen besitzt das Gerät keine IP-Adresse.

Es gibt folgende Möglichkeiten dem Gerät eine IP-Adresse zuzuweisen:

- DHCP (Standard-Einstellung)
- Primary Setup Tool
  Beachten Sie für weitere Informationen das Projektierungshandbuch "Primary Setup Tool". Das Projektierungshandbuch finden Sie bei Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 19440762 (http://support.automation.siemens.com/WW/view/de/19440762).
- STEP 7
- CLI über die serielle Schnittstelle
  Beachten Sie für weitere Informationen die Projektierungshandbücher zum Command Line Interface.
  Díe Projektierungshandbücher finden Sie bei Siemens Industry Automation and Drives Service & Support im Internet unter den folgenden Beitrags-IDs.
  - SCALANCE W780/W740 Command Line Interface - Beitrags-ID 62515451 (http://support.automation.siemens.com/WW/view/de/62515451)
  - SCALANCE W770/W730 Command Line Interface - Beitrags-ID 89534887 (http://support.automation.siemens.com/WW/view/de/89534887)
  - SCALANCE W760/W720 Command Line Interface - Beitrags-ID 89534699 (http://support.automation.siemens.com/WW/view/de/89534699)

Voraussetzung

- In den Eigenschaften des Geräts ist "IP-Adresse am Gerät einstellen" aktiviert. Weitere Informationen finden Sie unter "Adressierung von PROFINET-Geräten".

Eigenschaften und Parameter bearbeiten

Möglichkeiten zum Bearbeiten

Zum Bearbeiten von Eigenschaften und Parameter gibt es folgende Möglichkeiten:

- Hardware- und Netzwerkeditor
  Nachdem Sie die Netzwerkkomponente eingefügt haben, können Sie die Eigenschaften und Parameter bearbeiten, z. B. den Gerätenamen. Weitere Informationen dazu finden Sie unter "Hardware- und Netzwerkeditor".
- Web Based Management (WBM)
  Die Parameter und Eigenschaften sind über mitgelieferten HTML-Seiten (WBM-Seiten) erreichbar. Jede WBM-Seite hat eine eigene Hilfeseite, die die Eigenschaften und Parameter beschreiben. Beachten Sie für weitere Informationen die Projektierungshandbücher zum Web Based Management.
  - SCALANCE W780/W740 Web Based Management - Beitrags-ID: 62382125 (http://support.automation.siemens.com/WW/view/de/62382125)
  - SCALANCE W770/W730 Web Based Management - Beitrags-ID: 89535155 (http://support.automation.siemens.com/WW/view/de/89535155)
  - SCALANCE W760/W720 Web Based Management - Beitrags-ID: 89535534 (http://support.automation.siemens.com/WW/view/de/89535534)
- Command Line Interface
  Mit dem CLI können alle Konfigurationseinstellungen für das Gerät festgelegt werden. Das CLI bietet die gleichen Möglichkeiten wie das Web Based Management (WBM). Beachten Sie für weitere Informationen die Projektierungshandbücher zum Command Line Interface.
  - SCALANCE W780/W740 Command Line Interface - Beitrags-ID: 62515451 (http://support.automation.siemens.com/WW/view/de/62515451)
  - SCALANCE W770/W730 Command Line Interface - Beitrags-ID: 89534887 (http://support.automation.siemens.com/WW/view/de/89534887)
  - SCALANCE W760/W720 Command Line Interface - Beitrags-ID: 89534699 (http://support.automation.siemens.com/WW/view/de/89534699)

Verfügbarkeit

Die Verfügbarkeit der Einstellungen ist abhängig

- vom Gerätemodus (Betriebsart)
- vom Projektierungsmodus
- Gerät

Gerätemodus

Bei SCALANCE W gibt es folgende Gerätemodi:

- Access Point-Modus
- Client-Modus

Projektierungsmodus

Folgende Projektierungsmodi gibt es:

- Offline-Projektierung
  Die Erstkonfiguration einer PC-Station können Sie offline durchführen. In diesem Modus sind nur die Einstellungen verfügbar, die keine Verbindung zum Gerät benötigen.
- Online-Projektierungsmodus
  Wenn eine Verbindung zum Gerät besteht und das Web Based Management gestartet ist, sind im Arbeitsbereich zusätzliche Seiten verfügbar. Diese Seiten enthalten in der Onlinehilfe den Hinweis "Diese Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht". Bei einzelnen Einstellungen ist der Zusatz "Nur Online verfügbar" enthalten. Um das Web Based Management zu starten, gibt es folgende Möglichkeiten:
  - Wählen Sie in der Bereichsnavigation "Verwaltung" und klicken Sie auf "Web Based Management".
  - Starten Sie den Webbrowser und geben Sie im Adressfeld die IP-Adresse oder die URL des Geräts ein.

Gerät

Ein Teil der Einstellungen sind abhängig vom Gerät. In der Onlinehilfe verwendete Bezeichnungen haben folgende Bedeutung:

SCALANCE W740	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W748-1 RJ45 SCALANCE W748-1 M12
SCALANCE W780	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W788-1 M12 SCALANCE W788-2 M12 SCALANCE W788-2 M12 EEC SCALANCE W788-1 RJ45 SCALANCE W788-2 RJ45 SCALANCE W786-1 RJ45 SCALANCE W786-2 RJ45 SCALANCE W786-2IA RJ45 SCALANCE W786-2 SFP
SCALANCE W770	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W774-1 RJ45
SCALANCE W730	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W734-1 RJ45
SCALANCE W760	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W761-1 RJ45
SCALANCE W720	Das beschriebene Kapitel / der beschriebene Abschnitt / die beschriebene Einstellung ist nur für folgende Geräte gültig: SCALANCE W721-1 RJ45 SCALANCE W722-1 RJ45

Access Points	SCALANCE W788-1 M12 SCALANCE W788-2 M12 SCALANCE W788-2 M12 EEC SCALANCE W788-1 RJ45 SCALANCE W788-2 RJ45 SCALANCE W786-1 RJ45 SCALANCE W786-2 RJ45 SCALANCE W786-2IA RJ45 SCALANCE W786-2 SFP SCALANCE W774-1 RJ45 SCALANCE W761-1 RJ45
Client	SCALANCE W748-1 RJ45 SCALANCE W748-1 M12 SCALANCE W734-1 RJ45 SCALANCE W721-1 RJ45 SCALANCE W722-1 RJ45

Eintrag anlegen und löschen

Als Beispiel wird ein Eintrag beim Syslog-Client angelegt und gelöscht. Das Vorgehen ist bei allen Seiten gültig.

Neuen Eintrag anlegen

1. Selektieren Sie in der Netz- oder Gerätesicht das Gerät.
2. Öffnen Sie im Inspektorfenster die Eigenschaften des Geräts.
3. Gehen Sie im Inspektorfenster zu System > Syslog-Client.
4. Tragen Sie bei IP-Adresse des Syslog-Servers die IP-Adresse ein.
5. Klicken Sie in die Tabelle. Wählen Sie im Kontextmenü den Eintrag "Neuer Eintrag". In der Tabelle wird ein neuer Eintrag erstellt.

Eintrag löschen

1. Markieren Sie in der Tabelle den gewünschten Eintrag.
2. Wählen Sie im Kontextmenü den Eintrag "Löschen".

Häufig verwendete Schaltflächen

- **Aktualisieren der Anzeige mit "Aktualisieren"**
  Seiten, die aktuelle Parameter anzeigen, haben am unteren Rand die Schaltfläche "Aktualisieren". Klicken Sie auf diese Schaltfläche, wenn Sie für die angezeigte Seite aktuelle Daten vom Gerät anfordern wollen.
- **Speichern von Einstellungen für alle Ports mit "In Tabelle übernehmen"**
  Seiten, auf denen Sie mehrere Ports konfigurieren können, verfügen über 2 Tabellen. In der ersten Tabelle können Sie Einstellungen für alle Ports vornehmen und in die zweite Tabelle übernehmen. In der letzten Spalte der ersten Tabelle gibt es die Schaltfläche "In Tabelle übernehmen". Klicken Sie auf die Schaltfläche, um eingegebene Einstellungen für alle Ports zu speichern.

Fehlermeldung beim Laden der Konfiguration

Mögliche Probleme beim Laden

Während die Konfiguration auf ein SCALANCE-Gerät geladen wird, werden im Ausgabefenster Statusmeldungen über den Fortschritt des Ladevorgangs ausgegeben.

Wenn beim Laden auf das Gerät Probleme auftreten, ist die Ursache in den meisten Fällen eine der Folgenden:

- Falsches SCALANCE-Gerät im Projekt
- Falsche Firmware-Version
- Das SCALANCE-Gerät ist nicht mit dem Projektierungs-PC verbunden.

Fehlermeldung

Folgende Fehlermeldung kann beim Laden auftreten, wenn keine Verbindung aufgebaut werden kann:

Mögliche Fehler	Mögliche Ursachen	Mögliche Abhilfe
Fehlermeldung "Das Ziel-Gerät kann momentan keinen Download der Daten akzeptieren"	Das Ziel-Gerät befindet sich gerade in einer aktiven Anwendungsrelation mit einem IO-Controller.	Deaktivieren Sie die Anwendungsrelation zwischen dem IO-Controller und dem SCALANCE-Gerät.
	Die TIA-Konfigurationsschnittstelle auf dem Ziel-Gerät ist deaktiviert.	Aktivieren Sie über WBM oder CLI die TIA-Konfigurationsschnittstelle auf dem SCALANCE-Gerät.

Informationen anzeigen

Versionen

Diese Seite zeigt die Ausgabestände der Hardware und der Software des Geräts an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

Die Tabelle 1 enthält folgende Spalten:

- **Hardware**
  - Basic Device
    Zeigt das Grundgerät an.
  - WLAN X
    Zeigt die verfügbaren WLAN-Schnittstellen an. Ein Gerät kann bis zu zwei WLAN-Schnittstellen haben.
- **Name**
  Zeigt den Namen des Geräts oder des Moduls an.
- **Ausgabestand**
  Zeigt den Hardware-Ausgabestand des Geräts an.
- **Artikelnummer**
  Zeigt die Artikelnummer des Geräts oder des beschriebenen Moduls an. Bei der Funkkarte wird nur dann ein Ausgabestand angezeigt, wenn die WLAN-Schnittstelle eingeschaltet ist.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Software**
  - Firmware
    Zeigt die aktuelle Firmware-Version an. Wenn eine neue Firmware-Datei geladen wurde und das Gerät noch nicht neu gestartet ist, wird hier die Firmware-Version der geladenen Firmware-Datei angezeigt. Nach dem nächsten Neustart wird die geladene Firmware aktiviert und verwendet.
  - Bootloader
    Zeigt die Version der Boot-Software an, die im Gerät gespeichert ist.
  - Firmware_Running
    Zeigt die Firmware-Version an, die aktuell im Gerät verwendet wird
- **Beschreibung**
  Zeigt die Kurzbeschreibung der Software an.
- **Version**
  Zeigt die Versionsnummer des Software-Ausgabestands an.
- **Datum**
  Zeigt das Erstellungsdatum des Software-Ausgabestands an.

I&M

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite beinhaltet Informationen zu gerätespezifischen Hersteller- und Wartungsdaten wie Artikelnummer, Seriennummer, Versionsnummern etc. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Zeilen:

- **Hersteller-ID**
  Zeigt die Herstellerkennung an.
- **Artikelnummer**
  Zeigt die Artikelnummer an.
- **Seriennummer**
  Zeigt die Seriennummer an.
- **Hardware-Ausgabestand**
  Zeigt den Hardware-Ausgabestand an.
- **Software-Ausgabestand**
  Zeigt den Software-Ausgabestand an.
- **Versionszähler**
  Zeigt den Versionsänderungs-Zähler an: Zähler für Versionsänderungen seit der Erst-Inbetriebnahme
- **Aktualisierungsdatum**
  Datum und Uhrzeit der letzten Versionsänderung
- **Funktionskennzeichen**
  Zeigt das Funktionskennzeichen (Anlagenkennzeichen) des Geräts an. Das Anlagenkennzeichen (AKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Ortskennzeichen**
  Zeigt das Ortskennzeichen des Geräts an. Das Ortskennzeichen (OKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Datum**
  Zeigt das Datum, das bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.
- **Deskriptor**
  Zeigt die Beschreibung, die bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.

ARP / Nachbar-Tabelle

IPv6-Nachbarschaftstabelle

Zuordnung von MAC-Adresse und IPv6-Adresse

Über die IPv6-Nachbarschaftstabelle erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv6-Adresse. Diese Zuordnung wird von jedem Netzteilnehmer in seiner eigenen Nachbarschaftstabelle gepflegt..

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IPv6-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

ARP-Tabelle

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zuordnung von MAC-Adresse und IP-Adresse

Über das Address Resolution Protocol (ARP) erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv4-Adresse. Dieser Zuordnung wird von jedem Netzteilnehmer in seiner eigenen ARP-Tabelle gepflegt. Die Seite zeigt die ARP-Tabelle des Geräts.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IP-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

Log-Tabellen

Ereignis-Log

Protokollierung von Ereignissen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Das Gerät bietet die Möglichkeit, auftretende Ereignisse zu protokollieren, die Sie zum Teil unter "System > Ereignisse" festlegen können. So kann beispielsweise festgehalten werden, wann ein Authentifizierungsversuch fehlgeschlagen ist oder wann sich der Verbindungsstatus eines Ports geändert hat. Der Inhalt der Ereignisprotokoll-Tabelle bleibt auch nach dem Ausschalten des Geräts erhalten.

Beschreibung der angezeigten Werte

- **Severity-Filter**

Hinweis

Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

Sie können die Einträge der Tabelle nach Fehlerschwere filtern. Wählen Sie in den Optionskästchen oberhalb der Tabelle die gewünschten Einträge aus.

– Info
  Information

– Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Info" angezeigt.

– Warning
  Warnung
  Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Warning" angezeigt.

– Critical
  Kritisch
  Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

– **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis aufgetreten ist.

– **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.

– **Systemzeit**
  Zeigt das Datum und die Uhrzeit an, zu der das beschriebene Ereignis aufgetreten ist. Wenn die Systemzeit gesetzt ist, wird auch die Zeit angezeigt, bei der das Ereignis eingetreten ist.

– **Severity**
  Zeigt den Schweregrad der Meldung an.

– **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an. Eine Liste der möglichen Meldungen finden Sie im Anhang D des Projektierungshandbuches..

WLAN Authentifizierungs-Log

Protokollierung von Authentifizierungsversuchen

Die Seite zeigt in tabellarischer Form Informationen zu erfolgreichen oder fehlgeschlagenen Authentifizierungsversuchen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

- **Severity Filters**
  Die Einträge der Tabelle können Sie nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.
  - Info
    Information
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Info" angezeigt.
  - Warning
    Warnungen
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Warning" angezeigt.
  - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis aufgetreten ist.
- **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der der beschriebene Fehler aufgetreten ist.
- **Systemzeit**
  Zeigt den Zeitpunkt an, zu dem der beschriebene Fehler aufgetreten ist.
- **Severity**
  Zeigt den Schweregrad der Meldung an.
- **Log-Meldugen**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an. Eine Liste der möglichen Meldungen finden Sie im Anhang D des Projektierungshandbuches.

Wenn die Systemzeit gesetzt ist, wird auch die Zeit angezeigt, bei der das Ereignis eingetreten ist.

Schaltflächen

- **Leeren**
  Klicken Sie auf diese Schaltfläche, um den Inhalt der Protokolldatei zu löschen. Es werden alle Einträge gelöscht, unabhängig davon, was Sie unter "Severity Filters" ausgewählt haben.
  Die Anzeige wird dabei ebenfalls geleert. Erst wenn nach dem Wiederherstellen der Werkseinstellungen das Gerät neu gestartet ist, wird der Neustart-Zähler zurückgesetzt.

  Hinweis

  Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

- **Alle anzeigen**
  Klicken Sie auf diese Schaltfläche, um alle Einträge auf der WBM-Seite anzuzeigen. Beachten Sie, dass das Anzeigen aller Meldungen einige Zeit beanspruchen kann.
- **Weiter**
  Klicken Sie auf diese Schaltfläche, um zur nächsten Seite zu navigieren.
- **Zurück**
  Klicken Sie auf diese Schaltfläche, um zur vorherigen Seite zu navigieren.

Fehler

Fehlerstatus

Wenn ein Fehler auftritt, wird er auf dieser Seite angezeigt. Am Gerät werden Fehler durch das Leuchten der Fehler-LED signalisiert.

Gemeldet werden interne Fehler des Geräts sowie die Fehler, die Sie auf folgenden Seiten konfigurieren:

- "System > Ereignisse"
- "System > Fehlerkontrolle"

Die Berechnung des Fehlerzeitpunkts beginnt jeweils nach dem letzten Systemstart.

Wenn keine Fehler vorliegen, schaltet sich die Fehler-LED ab.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

- **Anzahl der gemeldeten Fehler**
  Zeigt an, wie oft die Fehler-LED angegangen ist und nicht wieviele Fehler aufgetreten sind.
- **Schaltfläche "Zähler zurücksetzen"**
  Über die Schaltfläche wird die Anzahl zurückgesetzt. Der Zähler wird durch einen Neustart zurückgesetzt.

Die Tabelle enthält die folgenden Spalten:

- **Fehlerzeitpunkt**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der der beschriebene Fehler aufgetreten ist.
- **Fehlerbeschreibung**
  Zeigt eine Kurzbeschreibung des aufgetretenen Fehlers an.
- **Fehlerstatus löschen**
  Wenn die Schaltfläche "Fehlerstatus löschen" aktiv ist, können Sie den Fehler löschen, z. B. Fehler des Ereignisses "Kalt-/Warmstart".

Spanning Tree

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt die aktuellen Informationen zu Spanning Tree und die Einstellungen der Root-Bridge an.

- Wenn Spanning Tree ausgeschaltet ist, werden nur die Basisinformationen zu diesem Gerät angezeigt.
- Wenn Spanning Tree eingeschaltet ist, werden die Statusinformationen der ausgewählten Instanz und in der Tabelle die Informationen der konfigurierten Ports angezeigt. Die Informationen sind abhängig vom gewählten Spanning Tree-Modus.

Angezeigte Werte

- **Spanning Tree-Modus**
  zeigt den eingestellten Modus an. Den Modus legen Sie bei "Layer 2 > MSTP > Allgmein" fest.
  Folgende Werte sind möglich:
  - '-'
  - STP
  - RSTP
  - MSTP
- **Instanz-ID**
  Zeigt die Nummer der Instanz an. Der Parameter ist abhängig vom projektierten Modus.
- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 32768.

- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Kosten**
  Zeigt die Pfadkosten von dem Gerät bis zur Root Bridge.
- **Bridge-Status**
  Zeigt den Status der Bridge an, z. B. ob das Gerät die Root-Bridge ist
- **Root-Priorität regional** (nur bei MSTP verfügbar)
  Beschreibung siehe Bridge Priority / Root Priority
- **Root-Adresse regional** (nur bei MSTP verfügbar)
  Zeigt die MAC-Adresse des Geräts.
- **Root-Kosten regional** (nur bei MSTP verfügbar)
  Zeigt die Pfadkosten von der regionalen Root-Bridge bis zur Root-Bridge.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt den Port an, über den das Gerät kommuniziert.
- **Rolle**
  zeigt den Status des Ports an. Folgende Werte sind möglich:
  - Disabled
    Der Port wurde manuell aus dem Spanning Tree entfernt und wird vom Spanning Tree nicht mehr berücksichtigt.
  - Designated
    Die Ports, die von der Root-Bridge wegführen.
  - Alternate
    Der Port mit einem alternativen Weg zu einem Netzwerksegment
  - Backup
    Wenn ein Switch mehrere Ports zu dem gleichen Netzwerksegment hat, wird der "schlechtere" Port zum Backup-Port.
  - Root
    Der Port, der den besten Weg zur Root Bridge bietet.
  - Master
    Dieser Port zeigt zu einer Root-Bridge, die außerhalb der MST-Region liegt.

- **Status**
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt. Der Parameter ist abhängig vom projektierten Protokoll. Folgende Status sind möglich:
  - Discarding
    Der Port empfängt BPDU-Telegramme. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    Der Port empfängt und sendet BPDU-Telegramme. Der Port ist in den Spanning Tree-Algorithmus einbezogen. Andere aus- und eingehende Telegramme werden verworfen.
  - Learning
    Der Port lernt aktiv die Topologie, d. h. die Teilnehmeradressen. Andere aus- und eingehende Telegramme werden verworfen.
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz. Der Port empfängt und sendet Datentelegramme.
- **Oper. Version**
  Beschreibt die Art des Spanning Tree, in dem der Port arbeitet
- **Priorität**
  Kann der vom Spanning-Tree ermittelte Weg alternativ über mehrere Ports eines Gerätes führen, so wird der Port mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) ausgewählt. Für die Priorität kann ein Wert von 0 bis 240 in 16er Schritte eingegeben werden. Wenn Sie einen Wert eingeben, der nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
- **Pfadkosten**
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Es wird die Strecke mit dem geringsten Wert als Weg ausgewählt. Haben mehrere Ports eines Gerätes den gleichen Wert, wird der Port mit der niedrigsten Portnummer ausgewählt.
  Ist der Wert "kalk. Kosten" "0", so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert von "kalk. Kosten" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000.

- **Edge-Typ**
  Zeigt den Typ der Verbindung an. Folgende Werte sind möglich:
  - Edge Port
    An diesem Port befindet sich ein Endgerät.
  - No Edge Port
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.
- **P.t.P. Typ**
  zeigt die Art der Punkt-zu-Punkt-Verbindung an. Folgende Werte sind möglich:
  - P.t.P.
    Bei Halbduplex wird von einer Punkt-zu-Punkt-Verbindung ausgegangen.
  - Shared Media
    Bei einer Vollduplexverbindung wird nicht von einer Punkt-zu-Punkt-Verbindung ausgegangen.

Hinweis

Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten.

Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

Ethernet-Statistiken

Schnittstellenstatistik

Informationen aus der MIB

Die Seite zeigt die Statistik aus der Schnittstellentabelle der Management Information Base (MIB).

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Byte empfangen**
  Zeigt die Anzahl der empfangenen Bytes an.
- **Byte gesendet**
  Zeigt die Anzahl der gesendeten Bytes an.
- **Unicast empfangen**
  Zeigt die Anzahl der empfangenen Unicast-Telegramme an.

- **Nicht-Unicast empfangen**
  Zeigt die Anzahl der empfangenen Telegramme an, die nicht vom Telegrammtyp Unicast sind.
- **Unicast gesendet**
  Zeigt die Anzahl der gesendeten Unicast-Telegramme an.
- **Nicht-Unicast gesendet**
  Zeigt die Anzahl der gesendeten Telegramme an, die nicht vom Telegrammtyp Unicast sind.
- **Fehler empfangen**
  Zeigt die Anzahl aller möglichen RX-Fehler an, siehe Register "Telegrammfehler".

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammlänge

Telegramme sortiert nach Länge

Diese Seite zeigt, wie viele Telegramme mit welcher Länge an jedem Port gesendet und empfangen wurden. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die angezeigten Werte werden durch RMON übermittelt.

Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link-Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.

  Hinweis
  Anzeige der Telegrammstatistik

  Beachten Sie bei der Statistik der Telegrammlängen, dass sowohl eingehende als auch ausgehende Telegramme gezählt werden.

- **Telegrammlängen**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der Telegramme entsprechend ihrer Telegrammlänge.
  Dabei wird in folgenden Telegrammlängen unterschieden:
  - 64 Byte
  - 65 - 127 Byte
  - 128 - 255 Byte
  - 256 - 511 Byte
  - 512 - 1023 Byte
  - 1024 - max

Hinweis
Datenverkehr auf geblockten Ports

Aus technischen Gründen können auf geblockten Ports Datenpakete angezeigt werden.

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammtyp

Empfangene Telegramme sortiert nach Telegrammtyp

Diese Seite zeigt, wie viele Telegramme des Typs "Unicast", "Multicast" und "Broadcast" an jedem Port empfangen wurden.

Sie können auf dieser Seite keine Konfigurationen vornehmen.

Die angezeigten Werte werden durch RMON übermittelt. Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link-Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Unicast / Multicast / Broadcast**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der eingegangenen Telegramme entsprechend der Telegrammtypen "Unicast", "Multicast" und "Broadcast".

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Telegrammfehler

Fehlerhaft empfangene Telegramme

Die Seite zeigt, wie viele fehlerhafte Telegramme pro Port empfangen wurden.

Sie können auf dieser Seite keine Konfigurationen vornehmen.

Die angezeigten Werte werden durch RMON übermittelt. Auf der Seite "Layer 2 > RMON > Statistik" können Sie einstellen, für welche Ports Werte angezeigt werden sollen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an.
- **Fehlertypen**
  Die weiteren Spalten hinter der jeweiligen Portnummer enthalten die absoluten Zahlen der eingegangenen Telegramme entsprechend ihres Fehlertyps.
  Dabei wird in den Spalten der Tabelle nach folgenden Fehlertypen unterschieden:
  - CRC
    Pakete, deren Inhalt nicht mit der zugehörigen CRC-Prüfsumme übereinstimmt.
  - Zu kurz
    Pakete mit einer Länge kleiner als 64 Bytes.
  - Zu lang
    Pakete, die verworfen wurden, weil sie zu lang sind.
  - Fragmente
    Pakete mit einer Länge kleiner als 64 Bytes und einer falschen CRC-Prüfsumme.
  - Jabbers
    VLAN-getaggte Pakete mit einer falschen CRC Prüfsumme, die verworfen wurden, weil sie zu lang sind.
  - Kollisionen
    Erkannte Kollisionen.

Beschreibung der Schaltfläche

Zähler zurücksetzen

Klicken Sie auf die Schaltfläche "Zähler zurücksetzen", um alle Zähler zurückzusetzen. Die Zähler werden auch durch einen Neustart zurückgesetzt.

Learning-Tabelle

Adressfilterung

Diese Seite zeigt den aktuellen Inhalt der Learning-Tabelle. In dieser Tabelle sind die Quelladressen von Unicast-Frames aufgeführt.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- VLAN-ID
  Zeigt die VLAN ID des Teilnehmers.

  Hinweis

  Diese Spalte erscheint nur dann in der Tabelle, wenn ein VLAN konfiguriert ist.

- MAC-Adresse
  Zeigt die MAC-Adresse des Teilnehmers.
- Status
  Zeigt den Status jedes Adresseintrags:
  - dynamisch
    Die angegebene Adresse wurde durch Empfang eines Frames dieses Teilnehmers gelernt und werden nach Ablauf der Aging Time wieder gelöscht, sollten keine weiteren Frames dieses Teilnehmers empfangen werden.
  - ungültig
    Diese Werte werden nicht ausgewertet.
- Port
  Zeigt an, über welchen Port der Teilnehmer mit der angegebenen Adresse erreichbar ist. Vom Gerät empfangene Frames, deren Zieladresse mit dieser Adresse übereinstimmen, werden an diesen Port weitergegeben.

Schaltflächen

- Alle anzeigen
  Klicken Sie auf diese Schaltfläche, um alle Einträge auf der Seite anzuzeigen. Beachten Sie, dass das Anzeigen aller Meldungen einige Zeit beanspruchen kann.
- Weiter
  Klicken Sie auf diese Schaltfläche, um zur nächsten Seite zu navigieren.
- Zurück
  Klicken Sie auf diese Schaltfläche, um zur vorherigen Seite zu navigieren.

Klappliste für Seitenwechsel

Wählen Sie aus der Klappliste die gewünschte Seite aus, um zu einer bestimmten Seite zu navigieren.

IPv6 Routing-Tabelle

Übersicht

Diese Seite zeigt die IPv6-Routen an, die aktuell verwendet werden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Ziel-Adresse dieser Route an.
- **Präfixlänge**
  Zeigt die Präfixlänge dieser Route an.
- **Gateway**
  Zeigt das Gateway für diese Route an.
- **Schnittstelle**
  Zeigt die Schnittstelle für diese Route an.
- **Metrik**
  Zeigt die Metrik der Route an. Je größer der Wert, desto länger benötigen Pakete zu Ihrem Ziel.
- **Routing-Protokoll**
  Zeigt an, aus welchem Routing-Protokoll der Eintrag der Routingtabelle stammt. Folgende Einträge sind möglich:
  - Verbunden: Verbundene Routen
  - Statisch: Statische Routen
  - RIPng: Routen über RIPng
  - OSPFv3: Routen über OSPFv3
  - Sonstige: Sonstige Routen

DHCP-Server

Diese Seite zeigt an, welche IPv4-Adressen vom DHCP-Server den Geräten zugeordnet wurden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

- **IP-Adresse**
  Zeigt die IPv4-Adresse an, die dem DHCP-Client zugeordnet ist.
- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Identifikationsmethode**
  Zeigt die Methode an, nach der der DHCP-Client identifiziert wird.
- **Identifikationswert**
  Zeigt die MAC-Adresse oder die Client-ID des DHCP-Clients an.
- **Zuordnungsmethode**
  Zeigt an, ob die IPv4-Adresse statisch oder dynamisch vergeben wurde. Die statischen Einträge konfigurieren Sie unter "System > DHCP > Statische Zuordnung".
- **Zuordnungsstatus**
  Zeigt den Status der Zuordnung an.
  - Zugeordnet
    Die Zuordnung wird verwendet.
  - Nicht verwendet
    Die Zuordnung wird nicht verwendet.
  - Wird geprüft
    Die Zuordnung wird geprüft.
  - Unbekannt
    Der Status der Zuordnung ist unbekannt.
- **Ablaufzeit**
  Zeigt an, bis wann die vergebene IPv4-Adresse noch gültig ist. Bis zu diesem Zeitpunkt muss der DHCP-Client entweder eine neue IPv4-Adresse anfordern oder die Gültigkeitsdauer der vergebenen IPv4-Adresse verlängern.

SNMP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die angelegten SNMPv3-Gruppen. Die SNMPv3-Gruppen konfigurieren Sie unter "System > SNMP".

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt den Gruppennamen an.
- **Benutzername**
  Zeigt den Benutzer an, welcher der Gruppe zugeordnet ist.

Security

Übersicht

Die Seite zeigt die Sicherheitseinstellungen und die lokalen und externen Benutzerkonten an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Dienste

Die Liste "Dienste" zeigt die Sicherheitseinstellungen an.

- **SSH-Server**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration".
  - Aktiviert: Verschlüsselter Zugriff auf das CLI
  - Deaktiviert: Kein verschlüsselter Zugriff auf das CLI
- **Webserver**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration"
  - HTTP/HTTPS: Der Zugriff auf das WBM ist über HTTP und HTTPS möglich.
  - HTTPS: Der Zugriff auf das WBM ist nur noch über HTTPS möglich.
- **SNMP**
  Die Einstellung konfigurieren Sie unter "System > SNMP > Allgemein".
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich.
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit der SNMP Version 3 möglich.
- **Management ACL**
  Die Einstellung konfigurieren Sie unter "Security > Management ACL".
  - Deaktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist deaktiviert.
  - Aktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist aktiviert, es sind jedoch keine Zugriffsregeln definiert.
  - Aktiviert: Nur eingeschränkter Zugriff
    Die Zugriffskontrolle ist aktiviert und es sind Zugriffsregeln definiert.

- **Login-Authentifizierung**
  Die Einstellung konfigurieren Sie unter "Security > AAA > Allgemein".
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
  - Lokal und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen.
    Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS mit Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
    Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.
- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie aktuell verwendet wird.

Lokale und externe Benutzerkonten

Lokale Benutzerkonten und Rollen konfigurieren Sie unter "Security > Benutzer".

Wenn Sie ein lokales Benutzerkonto anlegen, wird automatisch auch ein externes Benutzerkonto erzeugt. Bei lokalen Benutzerkonten handelt es sich um Benutzer mit jeweils einem Passwort zur Anmeldung auf dem Gerät.

In der Tabelle "Externe Benutzerkonten" wird ein Benutzer mit einer Rolle verknüpft, z. B. der Benutzer "Observer" mit der Rolle "user". In diesem Fall ist der Benutzer auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert, die zugehörige Gruppe jedoch unbekannt oder nicht vorhanden ist, prüft das Gerät, ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzerkonten" gibt. Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet. Wenn die zugehörige Gruppe auf dem Gerät bekannt ist, werden beide Tabellen ausgewertet. Dem Benutzer wird die Rolle mit den größeren Rechten zugewiesen.

Hinweis

Die Tabelle "Externe Benutzerkonten" wird nur ausgewertet, wenn Sie im RADIUS Authorisierungsmodus "Herstellerspezifisch" eingestellt haben.

Über CLI können Sie auf die externen Benutzerkonten zugreifen.

Die Tabelle "Lokale Benutzerkonten" gliedert sich in folgende Spalten:

- **Benutzerkonto**
  Zeigt den Namen des lokalen Benutzers an.
- **Rolle**
  Zeigt die Rolle des Benutzers an. Weitere Informationen zu den Funktionsrechten der Rolle erhalten Sie unter "Information > Security > Rollen".

Unterstützte Funktionsrechte

Die Seite zeigt die Funktionsrechte an, die lokal auf dem Gerät verfügbar sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

- **Funktionsrecht**
  Zeigt die Nummer des Funktionsrechts an. Den Nummern sind unterschiedliche Rechte in Bezug auf die Geräteparameter zugeordnet.
- **Beschreibung**
  Zeigt die Beschreibung des Funktionsrechts an.

Gruppen

Diese Seite zeigt an, welche Gruppe mit welcher Rolle verknüpft ist. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an. Der Name entspricht der Gruppe auf dem RADIUS-Server.
- **Rolle**
  Zeigt die Rolle an. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät.
- **Beschreibung**
  Zeigt die Beschreibung für die Verknüpfung an.

Rollen

Die Seite zeigt die Rollen an, die lokal auf dem Gerät gültig sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Rolle**
  Zeigt die Rolle an.
- **Funktionsrecht**
  Zeigt die Funktionsrechte der Rolle an:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen, aber nicht verändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.
  - 0
    Hierbei handelt es sich um eine Rolle, die das Gerät intern vergibt, wenn der Benutzer nicht authentifiziert werden konnte.
    Dem Benutzer wird der Zugriff auf das Gerät verweigert.
- **Beschreibung**
  Zeigt eine Beschreibung der Rolle an.

Inter AP Blocking

Die Seite zeigt eine Liste der Geräte an, mit denen die Clients kommunizieren dürfen.

Hinweis

Die Seite ist nur verfügbar

- bei SCALANCE W780 / W770
- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.
- mit gestecktem KEY-PLUG: W780 iFeatures (MLFB 6GK5 907-8PA00) oder W700 Security (MLFB 6GK5907-0PA00)

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an, auf die sich die Einstellungen beziehen.
- **Port**
  Zeigt die VAP-Schnittstelle an, auf die sich die Einstellungen beziehen.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Geräts, mit dem der Client kommunizieren darf.
- **IP-Adresse**
  Zeigt IP-Adresse des Geräts, mit dem der Client kommunizieren darf.
- **IP-Adressauflösung**
  Zeigt die IP-Adresse, mit der die erlaubte IP-Adresse aufgelöst wird.

WLAN

Übersicht AP

Konfigurationsübersicht

Diese Seite zeigt die Einstellungen/Eigenschaften des WLAN bzw. der WLAN-Schnittstelle an.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Angezeigte Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt die verfügbaren VAP-Schnittstellen an.
- **WLAN-Modus**
  Zeigt den Übertragungsstandard an. Bei aktiviertem DFS wird nicht zusätzlich der Übertragungsstandard "802.11h" angezeigt, sondern nur der konfigurierte Übertragungsstandard "802.11a".
- **Konfigurierter Kanal**
  Zeigt den projektierten Kanal an. Wenn "Auto" angezeigt wird, sucht der Access Point selbst nach einem freien Kanal.

- **Alternativer DFS-Kanal**
  Wenn die Funktion DFS aktiviert ist, wird der projektierte Alternativ-Kanal des Access Points angezeigt.
  Wenn "Auto" angezeigt wird, sucht der Access Point selbst nach einem Alternativ-Kanal. Wenn die Funktion DFS aktiviert ist, und der Access Point vor Aufnahme der Kommunikation auf dem gewählten Kanal 60 Sekunden nach Primärnutzern sucht, wird anstatt des Kanals der Text "scanning ..." angezeigt.
- **Operativer Kanal**
  Zeigt den Kanal des Access Points an, über den der Access Point kommuniziert.
- **HT-Kanal-Bandbreite**
  Zeigt die Kanalbandbreite an.
  - 20
    Kanalbandbreite 20 MHz
  - 40 up
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darüberliegende benachbarte Kanal werden verwendet.
  - 40 down
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darunterliegende benachbarte Kanal werden verwendet.

  Hinweis

  Kanalbandbreite 40 MHz und Frequenzband 2,4 GHz

  Wenn der Access Point auf dem konfigurierten Kanal oder auf benachbarten Kanälen einen anderen Access Point entdeckt, wechselt der Access Point von der Kanalbandbreite 40 MHz auf 20 MHz. Wenn Sie auf dem Access Point einen "freien" Kanal einstellen, verwendet der Access Point die Kanalbandbreite 40 MHz.

- **iFeatures**
  Zeigt an, welche iFeatures verwendet werden.
  - -
    iFeatures werden nicht verwendet.
  - iPCF
  - iPCF-MC
  - iREF
- **Status**
  Zeigt den Status der WLAN-Schnittstelle an.
  - enabled
    WLAN-Schnittstelle ist aktiviert.
  - disabled
    WLAN-Schnittstelle ist deaktiviert.

Die Tabelle 2 gliedert sich in folgende Spalten:

- Funkschnittstelle
  Zeigt die verfügbaren WLAN-Schnittstellen in dieser Spalte an.
- Port
  Zeigt den Port des Virtuellen Access Points an.
- MAC-Adresse
  Zeigt die MAC-Adresse des Virtuellen Access Points an.
- SSID
  Zeigt die SSID an.
- Security
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.
- Status
  Zeigt den Status der WLAN-Schnittstelle an.
  - enabled
    WLAN-Schnittstelle ist aktiviert.
  - disabled
    WLAN-Schnittstelle ist deaktiviert.

Client-Liste

Angemeldete Clients

Die Seite zeigt die am Access Point angemeldeten Clients sowie Zusatzinformationen z. B. Status, Signalstärke, MAC-Adresse.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Angezeigte Werte

- Angemeldete Clients
  Zeigt die Anzahl der Clients an, die am Access Point angemeldet sind.

Die Tabelle gliedert sich in folgende Spalten:

- AID (Associated ID)
  Zeigt die Verbindungs-ID des Clients an. Wenn der Client sich über die VAP-Schnittstelle mit dem Access Point verbindet, bekommt der Client eine Verbindungs-ID zugewiesen. Die Verbindungs-ID ist innerhalb einer VAP-Schnittstelle eindeutig. Wenn sich zwei Clients an verschiedenen VAP-Schnittstellen anmelden, können beide Clients die gleiche ID erhalten.
- Funkschnittstelle
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- Port
  Zeigt die VAP-Schnittstelle an.
- Typ
  Zeigt die Client-Art an, z. B. "Sta" steht für IEEE 802.11 Standard-Client.
- MAC-Adresse
  Zeigt die MAC-Adresse des Clients an.
- Systemname
  Zeigt den Systemnamen des Clients an, wenn der Client diesen an den Access Point kommuniziert. Nicht alle Clients unterstützen diesen Parameter.
- Kanal
  Zeigt den Kanal an, über den der Client mit dem Access Point kommuniziert.
- Signalstärke [dBm]
  Zeigt die Signalstärke des verbundenen Clients in dBm an.
- Signalstärke [%]
  Zeigt die Signalstärke des verbundenen Clients in Prozent an.
- Age [s]
  Zeigt die Zeit an, die seit der letzten Aktivität des Clients vergangen ist.
- Security
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.
- WLAN-Modus
  Zeigt den Übertragungsstandard an. Bei aktiviertem DFS wird nicht zusätzlich der Übertragungsstandard "802.11h" angezeigt, sondern nur der konfigurierte Übertragungsstandard "802.11a".
- Maximale Datenrate [Mbps]
  Zeigt die maximale Datenrate in Megabits pro Sekunde an.
- Status
  Zeigt den aktuellen Status der Verbindung an, z. B Verbunden bedeutet der Client ist mit dem Access Point verbunden und ist bereit mit dem AP zu kommunizieren.

WDS-Liste

Kommunikation zwischen Access Points

Im normalen Betrieb ist der Access Point als Schnittstelle zu einem Netz eingesetzt und kommuniziert mit Clients. Es gibt allerdings auch den Anwendungsfall, dass mehrere Access Points miteinander kommunizieren müssen, z.B. zum Zweck der Reichweitenvergrößerung oder zum Aufbau eines Wireless-Backbones. Diese Betriebsart ist mit WDS (Wireless Distributed System) möglich.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Die Seite zeigt Informationen über die WDS-Verbindungen des Access Points.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt den Port an.
- **BSSID**
  Zeigt die MAC-Adresse des WDS-Partners an.
- **WDS ID**
  Zeigt den Namen des WDS-Partners an.
- **Kanal**
  Zeigt den Kanal an, über den der Access Point mit dem WDS-Partner kommuniziert.
- **Signalstärke [dBm]**
  Zeigt die Signalstärke des verbundenen Access Points in dBm an.
- **Signalstärke [%]**
  Zeigt die Signalstärke des verbundenen Access Points in Prozent an.
- **Security**
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.
- **Maximale Datenrate [Mbps]**
  Zeigt die maximale Datenrate in Megabits pro Sekunde an.
- **Status**
  Zeigt den aktuellen Zustand der WDS-Verbindung an.

Überlappung AP

Überlappende Kanäle

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Für einen optimalen Datendurchsatz ist es notwendig, dass der eingestellte Funkkanal nicht von anderen Access Points genutzt wird. Im 2,4 GHz Band (802.11b oder 802.11g) gibt es eine Überlappung der Kanäle, sodass ein Access Point nicht nur den eingestellten Kanal belegt, sondern zusätzlich die benachbarten 2-3 Kanäle. Achten Sie deshalb darauf, dass zu benachbarten Access Points ein ausreichend großer Kanalabstand besteht.

Die Seite zeigt alle Access Points, die auf dem eingestellten oder auf benachbarten Kanälen (bei 2,4 GHz) sichtbar sind. Wenn hier Einträge vorhanden sind, ist der maximale Datendurchsatz des Access Points und die Verfügbarkeit der Kommunikationsverbindung zum Access Point potenziell beeinträchtigt.

Angezeigte Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Aging Time [min]**
  Legen Sie die Lebensdauer für die Einträge in der Liste fest. Ist ein Access Point länger als die eingestellte Zeit inaktiv, wird er aus der Liste entfernt.

  Hinweis
  Ändern der Aging Time

  Die Aging Time ist eine WLAN-Einstellung. Deshalb wird bei einer Änderung die WLAN Verbindung kurz unterbrochen, um den neuen Wert zu übernehmen.

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen in dieser Spalte an.
- **Typ**
  Zeigt die Betriebsart der WLAN-Schnittstelle an.
- **SSID**
  Zeigt die SSID des Access Points an.
- **BSSID**
  Zeigt die MAC-Adresse des Access Points an.

- **Systemname**
  Zeigt den Systemnamen des Geräts an. Der Eintrag ist abhängig vom Access Point. Nicht alle Access Point unterstützen diesen Parameter.
- **Kanal**
  Zeigt den Kanal an, über den der Client mit dem Access Point kommuniziert.
- **Signalstärke [dBm]**
  Zeigt die Signalstärke des Clients in dBm an.
- **Signalstärke [%]**
  Zeigt die Signalstärke des Clients in Prozent an.
- **Age [s]**
  Zeigt die Zeit an, die seit der letzten Aktivität des Access Points vergangen ist.
- **Security**
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.
- **WLAN-Modus**
  Zeigt den Übertragungsstandard an. Bei aktiviertem DFS wird nicht zusätzlich der Übertragungsstandard "802.11h" angezeigt, sondern nur der konfigurierte Übertragungsstandard "802.11a" oder "802.11n".

Übersicht Client

Konfigurationsübersicht

Die Seite zeigt eine Übersicht der vorhandenen Clients und ihrer Konfiguration.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- bei Clients oder Access Points im Client-Modus.

Angezeigte Werte

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen in dieser Spalte an.
- **WLAN-Modus**
  Zeigt den Übertragungsstandard an.

- **MAC-Modus**
  Zeigt an, wie der Schnittstelle die MAC-Adresse zugeordnet ist.
  - Automatisch
    Der Client nimmt automatisch die Quell-MAC-Adresse des ersten Telegramms an, das er über die Ethernet-Schnittstelle empfängt.
  - Manuell
    Die Adresse wurde manuell eingetragen
  - Eigene
    Der Client verwendet die MAC-Adresse der Ethernet-Schnittstelle für die WLAN-Schnittstelle.
  - Layer-2-Tunnel
    Der Client verwendet die MAC-Adresse der Ethernet-Schnittstelle für die WLAN-Schnittstelle. Zusätzlich wird das Netzwerk über die, an der Ethernet-Schnittstelle des Clients, angeschlossenen MAC-Adressen informiert. Bis zu acht MAC-Adressen können verwendet werden.
- **MAC-Adresse**
  Zeigt die MAC-Adresse der WLAN-Schnittstelle an.
- **Operativer Kanal**
  Zeigt den Kanal des Access Points an, mit dem der Client verbunden ist.
- **HT-Kanal-Bandbreite [Mhz]**
  Zeigt die Kanalbandbreite an.
  - 20
    Kanalbandbreite 20 MHz
  - 40 up
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darüberliegende benachbarte Kanal werden verwendet.
  - 40 down
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darunterliegende benachbarte Kanal werden verwendet.

  Hinweis

  Kanalbandbreite 40 MHz und Frequenzband 2,4 GHz

  Wenn der Access Point auf dem konfigurierten Kanal oder auf benachbarten Kanälen einen anderen Access Point entdeckt, wechselt der Access Point von der Kanalbandbreite 40 MHz auf 20 MHz. Wenn Sie auf dem Access Point einen "freien" Kanal einstellen, verwendet der Access Point die Kanalbandbreite 40 MHz.

- **Verbundene BSSID**
  Zeigt die MAC-Adresse des Access Points an, mit dem der Client verbunden ist.
- **Verbundene SSID**
  Zeigt die SSID des Access Points an, mit dem der Client verbunden ist.
- **Security**
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.

- **iFeatures**
  Zeigt an, welche iFeatures verwendet werden.
  - -
    iFeatures werden nicht verwendet.
  - iPCF
  - iPCF-MC
- **Status**
  Zeigt den Status der WLAN-Schnittstelle an.
  - enabled
    WLAN-Schnittstelle ist aktiviert.
  - disabled
    WLAN-Schnittstelle ist deaktiviert.

Verfügbare APs

Verfügbare Access Points

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- bei Clients oder Access Points im Client-Modus.

Diese Seite zeigt alle Access Points an, die der Client sieht. In der Liste sind auch die Access Points enthalten, auf die sich der Client wegen seiner Konfiguration nicht verbinden kann.

Hinweis

Anzeige bei aktiviertem iPCF-Modus

Wenn der iPCF-Modus aktiv ist, ergibt sich ein anderes Anzeigeverhalten. Da hier der Client keinen Background-Scan durchführt, wird nur der Access Point angezeigt, mit dem der Client aktuell verbunden ist.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die WLAN-Schnittstelle, die den Access Point sieht.
- **SSID**
  Zeigt die SSID des Access Points an.
- **BSSID**
  Zeigt die MAC-Adresse des Access Points an.

- **Systemname**
  Zeigt den Systemnamen des Access Points an. Der Eintrag ist abhängig vom Access Point. Nicht alle Access Point unterstützen diesen Parameter.
- **Kanal**
  Zeigt den Kanal an auf dem der Access Point funkt bzw.kommuniziert.
- **Signalstärke [dBm]**
  Zeigt die Signalstärke des Access Points in dBm an.
- **Signalstärke [%]**
  Zeigt die Signalstärke des Access Points in Prozent an.
- **Typ**
  Zeigt die Betriebsart der WLAN-Schnittstelle an.
- **Security**
  Zeigt an, welches Authentifizierungsverfahren verwendet wird.
  Wenn die Authentifizierungsverfahren "Open System + Verschlüsselung" oder "Shared Key" verwendet werden, wird für beide das Authentifizierungsverfahren "Encrypted (WEP/AES)" angezeigt.
- **WLAN-Modus**
  Zeigt den Übertragungsstandard an. Bei aktiviertem DFS wird nicht zusätzlich der Übertragungsstandard "802.11h" angezeigt, sondern nur der konfigurierte Übertragungsstandard "802.11a" oder "802.11n".
- **Status**
  Zeigt den Status des Access Points an, z.B. ob der Access Point verfügbar ist.

Roaming erzwingen

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Das Gerät überwacht zyklisch die Verbindung zu IP-Adressen. Dazu sendet das Gerät in regelmäßigen Abständen Echomeldungen (Ping) an die projektierte Ziel-IP-Adresse.

Auf dieser WBM-Seite wird angezeigt, welchen aktuellen Status die Verbindung hat. Zudem wird angezeigt, ob ein Roaming ausgeführt wird.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren VAP-Schnittstellen an.
- **Zieladresse / Status**
  Zeigt an, welche Zieladresse überwacht wird und wie der Status der Verbindung ist. Die Zieladresse konfigurieren Sie unter "Schnittstellen > WLAN > Roaming erzwingen".
  - not configured: Es ist keine Zieladresse konfiguriert.
  - idle: Die Konfiguration ist unvollständig.
  - up: Die Zieladresse ist erreichbar.
  - down: Die Zieladresse nicht erreichbar.
- **IP-Adresse unerreichbar - Roaming erzwingen**
  Zeigt an, ob aktuell ein Roaming durchgeführt wird.
  - inactive: Es wird kein Roaming durchgeführt.
  - active: Es wird Roaming durchgeführt. Keine der Zieladressen ist erreichbar.

IP-Mapping-Tabelle

WLAN-Zugang für mehrere Geräte über einen Client

Hinweis

Die Seite ist nur verfügbar

- wenn eine Online-Verbindung zum Gerät besteht.
- bei Clients oder Access Points im Client-Modus.

Sie können mit einem Client mehreren Geräten den WLAN-Zugang ermöglichen, wenn Sie das sogenannte IP-Mapping einsetzen. So brauchen Sie nicht jedes Gerät mit einem eigenen WLAN-Client auszustatten. Voraussetzung ist, dass die angeschlossenen Geräte nur mit IP-Telegrammen angesprochen werden. Eine Kommunikation auf MAC-Adressen-Ebene (ISO/OSI Schicht 2) kann

- zu einer Komponente erfolgen, deren MAC-Adresse im Client konfiguriert ist,
- zu maximal acht Komponenten erfolgen, wenn bei MAC-Modus "Layer-2-Tunnel" gewählt wird.

Die Einstellung "Layer-2-Tunnel" erfüllt Forderungen von Industrie-Applikationen, bei denen MAC-Adressen basierte Kommunikation zu mehreren Geräten hinter dem Client stattfindet. Clients mit dieser Einstellung können sich nicht auf Standard-Wifi-Geräte und Access Points mit Firmware V3.0 oder älter verbinden.

Der Client unterhält eine Tabelle mit der Zuordnung von MAC-Adresse und IP-Adresse, um eingehende IP- Telegramme an die richtige MAC-Adresse zu verschicken. Die Seite zeigt die IP-Mapping-Tabelle des Geräts.

Hinweis

Anzeige der IP-Mapping Tabelle

Ist auf dem Gerät bei MAC-Modus "Layer 2-Tunnel" konfiguriert, wird die IP-Mapping-Tabelle nicht angezeigt.

Angezeigte Werte

- **MAC-Adresse**
  Die MAC-Adresse eines Gerätes, das sich, aus Sicht des Access Points, hinter dem WLAN-Client befindet.
- **IP-Adresse**
  Die IP-Adresse, die vom WLAN-Client für dieses Gerät verwaltet wird.
- **Typ**
  Für den Typ gibt es zwei Optionen:
  - system
    Die Informationen beziehen sich auf den WLAN-Client selbst.
  - learned
    Die Informationen beziehen sich auf ein Gerät, das sich hinter dem WLAN-Client befindet.

MAC-Modus

Frames, die vom Client zum Access Point geschickt werden, enthalten als Quell-MACAdresse immer die MAC-Adresse des WLAN-Clients. In der "Learning-Tabelle" des Access Points steht daher nur die MAC-Adresse des WLAN-Clients.

Sind hinter dem Client mehrere SCALANCE W700-Geräte, sollte nicht die Option "Automatisch" aktiviert sein. In diesem Falle würde die MAC-Adresse wahllos an das erste SCALANCE W700-Gerät vergeben, das sich über Ethernet meldet. Wenn zwischen dem Access Point und dem Client nur IP-Kommunikation stattfindet, kann die Default-Einstellung "Eigene" beibehalten werden. Sollen MAC-basierte Frames auch von SCALANCE W700-Geräten hinter dem Client versendet werden können, müssen Sie die Einstellungen "Manuell", "Automatisch" oder "Layer 2-Tunnel" auswählen.

Hintergrundrauschen

Auf der Seite wird das Hintergrundrauschen des Kanals angezeigt.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Anschluss**
  Zeigt die Bezeichnung des jeweiligen Antennenanschlusses an.
- **Kanal [dBm]**
  Zeigt das Hintergrundrauschen des eingestellten Kanals an.
- **Erweiterter Kanal [dBm]**
  Zeigt das Hintergrundrauschen des erweiterten Kanals (HT-40) an.

WLAN-Statistiken

Fehler

Die Seite zeigt an, wie viele fehlerhafte Telegramme pro WLAN-Schnittstelle empfangen bzw. gesendet wurden. Wenn eine erhöhte Fehleranzahl auftritt, sollten Sie die Einstellungen für die WLAN-Schnittstelle(n), den Aufbau der Geräte und die Verbindungsqualität überprüfen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

Die Tabelle Übertragungsfehler gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die WLAN-Schnittstelle an, auf die sich die Einträge beziehen.
- **Fehlertypen**
  Die weiteren Spalten hinter der jeweiligen WLAN-Schnittstelle enthalten die absoluten Zahlen der gesendeten Telegramme entsprechend ihres Fehlertyps.
  Dabei wird in den Spalten der Tabelle nach folgenden Fehlertypen unterschieden:
  - Übertragungsfehler
    Zeigt die Anzahl und den prozentualen Anteil der fehlerhaften Telegramme an, die gesendet wurden.
  - Verworfene Telegramme
    Zeigt die Anzahl und den prozentualen Anteil der Telegramme an, die verworfen wurden. Das Telegramm konnte trotz aller Wiederholungen nicht erfolgreich gesendet werden.
    Das Telegramm wurde noch nicht gesendet und der Empfänger hat sich zwischenzeitlich abgemeldet.
  - Sendewiederholungen
    Zeigt die Anzahl und den prozentualen Anteil der erfolgreich gesendeten Telegramme an, die einen oder mehrere Wiederholungen benötigt haben.

Die Tabelle "Empfangene Fehler" gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die WLAN-Schnittstelle an, auf die sich die Einträge beziehen.
- **Fehlertypen**
  Die weiteren Spalten hinter der jeweiligen WLAN-Schnittstelle enthalten die absoluten Zahlen der empfangenen Telegramme ihres Fehlertyps.
  Dabei wird in den Spalten der Tabelle nach folgenden Fehlertypen unterschieden:
  - Empfangene Fehler
    Zeigt die Anzahl und den prozentualen Anteil der fehlerhaften Telegramme an, die empfangen wurden.
  - Doppelte Telegramme
    Zeigt die Anzahl und den prozentualen Anteil der Telegramme an, die doppelt empfangen wurden.
  - Verschlüsselungsfehler
    Zeigt die Anzahl und den prozentualen Anteil der fehlerhaften verschlüsselten Telegramme an.
  - FCS-Fehler
    Zeigt die Anzahl und den prozentualen Anteil der Telegramme an, bei denen die Prüfsumme nicht korrekt war.

Gesendete Management-Telegramme

Die Seite zeigt an, wie viele Telegramme für den Anmeldevorgang bzw. Abmeldevorgang pro VAP-Schnittstelle gezählt wurden.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die VAP-Schnittstelle an, auf die sich die Einträge beziehen.
- **Telegramme**
  - Management-Telegramme
    Zeigt die Anzahl der Management-Telegramme an
  - Association-Anfragen
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten anfragenden Association-Telegramme an.
  - Association-Antworten
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten antwortenden Association-Telegramme an.
  - Disassociation-Anfragen
    Zeigt die Anzahl der für einen Abmeldevorgang relevanten anfragenden Disassociation-Telegramme an.
  - Authentication-Anfragen
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten anfragenden Authentication-Telegramme an.
  - Authentication-Antworten
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten antwortenden Authentication-Telegramme an.
  - Deauthentication-Anfragen
    Zeigt die Anzahl der für einen Abmeldevorgang relevanten Deauthentication-Telegramme an.

Empfangene Management-Telegramme

Die Seite zeigt an, wie viele Telegramme für den Anmeldevorgang bzw. Abmeldevorgang pro VAP-Schnittstelle gezählt wurden.

Hinweis

Die Seite ist nur verfügbar,

- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die VAP-Schnittstelle an, auf die sich die Einträge beziehen.
- **Telegramm**
  - Management-Telegramme
    Zeigt die Anzahl der Management-Telegramme an
  - Association-Anfragen
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten anfragenden Association-Telegramme an.
  - Association-Antworten
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten antwortenden Association-Telegramme an.
  - Disassociation Anfragen
    Zeigt die Anzahl der für einen Abmeldevorgang relevanten anfragenden Disassociation-Telegramme an.
  - Authentication-Anfragen
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten anfragenden Authentication-Telegramme an.
  - Authentication-Antworten
    Zeigt die Anzahl der für einen Anmeldevorgang relevanten antwortenden Authentication-Telegramme an.
  - Deauthentication-Anfragen
    Zeigt die Anzahl der für einen Abmeldevorgang relevanten Deauthentication-Telegramme an.

Gesendete Daten-Telegramme

Die Seite zeigt an, wie viele Telegramme pro VAP-Schnittstelle gesendet wurden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die VAP-Schnittstelle an, auf die sich die Einträge beziehen.
- **Telegrammtypen**
  Die weiteren Spalten hinter der jeweiligen VAP-Schnittstelle enthalten die absoluten Zahlen der gesendeten Telegramme entsprechend der Telegrammtypen.
  Dabei wird in den Spalten der Tabelle nach folgenden Telegrammtypen unterschieden:
  - Datentelegramme
    Zeigt die Anzahl der gesendeten Datentelegramme an.
  - Multicast/Broadcast-Telegramme
    Zeigt die Anzahl der gesendeten Multicast- und Broadcast-Telegramme an.
  - Unicast-Telegramme
    Zeigt die Anzahl der gesendeten Unicast-Telegramme an.
  - Mittlere Datenrate
    Zeigt die mittlere Datenrate der zuletzt gesendeten Datentelegramme an.

Empfangene Daten-Telegramme

Die Seite zeigt an, wie viele Telegramme pro VAP-Schnittstelle empfangen wurden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die VAP-Schnittstelle an, auf die sich die Einträge beziehen.
- **Telegrammtypen**
  Die weiteren Spalten hinter der jeweiligen VAP-Schnittstelle enthalten die absoluten Zahlen der empfangenen Telegramme entsprechend der Telegrammtypen.
  Dabei wird in den Spalten der Tabelle nach folgenden Telegrammtypen unterschieden:
  - Datentelegramme
    Anzahl der empfangenen Datentelegramme an.
  - Multicast/Broadcast-Telegramme
    Zeigt die Anzahl der empfangenen Multicast- und Broadcast-Telegramme an.
  - Unicast-Telegramme
    Zeigt die Anzahl der empfangenen Unicast-Telegramme an.
  - Mittlere Datenrate
    Zeigt die mittlere Datenrate der zuletzt empfangenen Datentelegramme an.

WLAN iFeatures

iREF Client-Liste

Die Seite zeigt über welchen Antennenanschluss die am Access Point angemeldeten Clients kommunizieren. Zusätzliche werden Informationen wie z.B. die Signalstärke und die MAC-Adresse der WLAN-Schnittstelle dargestellt.

Hinweis

Die Seite ist nur verfügbar

- bei SCALANCE W780
- bei SCALANCE W770
- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.
- mit gestecktem KEY-PLUG: W780 iFeatures (MLFB 6GK5 907-8PA00)

Angezeigte Werte

- **Angemeldete Clients**
  Zeigt die Anzahl der Clients an, die am Access Point angemeldet sind.

Die Tabelle gliedert sich in folgende Spalten:

- **AID** (Associated ID)
  Zeigt die Verbindungs-ID des Clients an. Wenn der Client sich über die VAP-Schnittstelle mit dem Access Point verbindet, bekommt der Client eine Verbindungs-ID zugewiesen. Die Verbindungs-ID ist innerhalb einer VAP-Schnittstelle eindeutig. Wenn sich zwei Clients an verschiedenen VAP-Schnittstellen anmelden, können beide Clients die gleiche ID erhalten.
- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt die VAP-Schnittstelle an.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Clients an.
- **Systemname**
  Zeigt den Systemnamen des Clients an, wenn der Client diesen an den Access Point kommuniziert. Nicht alle Clients unterstützen diesen Parameter.
- **Tx Chain**
  Zeigt den Antennenanschluss an, über den der Client mit dem Access Point kommuniziert.
- **Signalstärke [dBm]**
  Zeigt die Signalstärke des verbundenen Clients in dBm an.
- **Signalstärke [%]**
  Zeigt die Signalstärke des verbundenen Clients in Prozent an.
- **Age [s]**
  Zeigt das "Alter" des aufgelisteten Clients an.

iREF WDS-Liste

Die Seite zeigt die Access Points an, die über eine WDS-Strecke am Access Point angemeldet sind. Es werden Informationen wie z.B. die verwendete Antenne und die Signalstärke der WLAN-Schnittstelle dargestellt.

Hinweis

Die Seite ist nur verfügbar,

- SCALANCE W780
- SCALANCE W770
- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.
- mit gestecktem KEY-PLUG: Access Point W780 iFeatures (MLFB 6GK5 907-8PA00)

Angezeigte Werte

- **Verbundene WDS-Partner**
  Zeigt die Anzahl der Access Points an, die am Access Point angemeldet sind

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt die WDS-Schnittstelle an.
- **BSSID**
  Zeigt die MAC-Adresse des WDS-Partners an.
- **WDS ID**
  Zeigt den Namen des WDS-Partners an.
- **Tx Chain**
  Zeigt den Antennenanschluss an, über den die beiden Access Points miteinander kommunizieren.
- **Signalstärke [dBm]**
  Zeigt die Signalstärke des verbundenen Access Points in dBm an.
- **Signalstärke [%]**
  Zeigt die Signalstärke des verbundenen Access Points in Prozent an.

AeroScout

Hinweis

Die Seite ist nur verfügbar

- bei SCALANCE W780 und SCALANCE W774-1 RJ45
- wenn eine Online-Verbindung zum Gerät besteht.
- im Access Point-Modus.
- mit gestecktem KEY-PLUG: Access Points W780 iFeatures (MLFB 6GK5 907-8PA00)

Hinweis

AeroSout und iPCF-Modus

Die Funktion AeroScout ist nicht mit anderen iFeatures (iPCF, iPCF-MC iREF) kombinierbar. AeroScout kann ausschließlich im 2,4 GHz-Band entsprechend IEEE 802.11g, IEEE 802.11n und IEEE 802.11n-only verwendet werden.
Weiterführende Informationen entnehmen Sie bitte der Dokumentation der Firma AeroScout (www.aeroscout.com)..

Angezeigte Werte

- **Tag-Information weiterleiten**
  Zeigt die Einstellung des Management-Programms an.
  Im Management-Programm, das die AeroScout-Frames auswertet, können Sie festlegen, ob ein IWLAN-Gerät Frames weiterleitet oder nicht.

  Hinweis

  Keine Auswertung oder Bearbeitung

  Das SCALANCE W-Gerät leitet bei entsprechender Konfiguration AeroScout-Frames weiter, führt aber selbst keine Bearbeitung oder Auswertung durch. Dies erfolgt ausschließlich durch das Programm "AeroScout System Manager".

- **AeroScout-Status**
  Zeigt an, ob AreoScout aktiviert ist oder nicht. Sie können das unter "iFeatures > AeroSout" für jede WLAN-Schnittstelle des Geräts einstellen. Wenn Sie die Weiterleitung aktiviert haben, wird "active" angezeigt, andernfalls "inactive
- **Engine-Port**
  Zeigt den Port an, auf dem das Gerät UDP-Telegramme von der AeroScout Engine erwartet.
- **Antwort-Port**
  Zeigt den Port an, an dem das Gerät die empfangenen AeroScout-Frames weiterleitet.
- **Antwort IP-Adresse**
  Zeigt die IP-Adresse des PCs an, auf dem die AeroScout Engine für die Auswertung der AeroScout-Frames läuft.
- **Multicast-Addresse**
  Zeigt die Multicast-Adresse an, die der Tag verwendet. Die Multicast-Adresse wird in der AeroScout Engine konfiguriert.

- **Gesendete Bestätigungen**
  Zeigt die Anzahl der vom Gerät versendeten Bestätigungen an. Die Anzahl ist ein Maß für die empfangenen UDP-Frames.
- **Verworfene Meldungen**
  Zeigt die Anzahl der vom Gerät verworfenen AeroScout-Frames an. Ist z. B. ein AeroScout-Tag so konfiguriert, dass er auf Kanal 1 sendet, leitet das Gerät einen auf Kanal 6 empfangenen Frame nicht weiter.

iRPR

Auf dieser Seite können Sie prüfen, ob die Einstellungen für iPRP korrekt sind. Sie können z. B. sehen welches Gerät der Partner-Client ist.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Hinweis

Clients oder Access Points im Client-Modus

- SCALANCE W740, SCALANCE W734-1 RJ45 und SCALANCE W722-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Client: W740 iFeatures (MLFB 6GK5 907-4PA00)

Access Point-Modus

- SCALANCE W780 und SCALANCE W774-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die WLAN-Schnittstellen an, über die der Client mit dem Access Point verbunden ist.
- **Port** (Nur im Acces Point-Modus)
  Zeigt die VAP-Schnittstelle an, an der die iPRP-Clients angemeldet sind.
- **iPRP-Client**
  Zeigt die MAC-Adresse des iPRP-Clients
- **Aktivitätsstatus**
  Zeigt an, ob iPRP aktiv ist.
- **Partner-Client**
  Zeigt die MAC-Adresse des Partner-Clients an.
- **Partner BSS**
  Zeigt die MAC-Adresse des Access Points an, mit dem der Partner-Client verbunden ist.

- **Lösch-Telegramme gesendet**
  Zeigt die Anzahl der gesendeten iPRP-Lösch-Telegramme an, die das Gerät (Access Point / Client) an sein Partnergerät gesendet hat.
- **Lösch-Telegramme empfangen**
  Zeigt die Anzahl der empfangenen iPRP-Lösch-Telegramme an, die das Gerät (Access Point / Client) von seinem Partnergerät erhalten hat.
- **Telegramme gelöscht**
  Zeigt die Anzahl der noch nicht gesendeten Telegramme an, die aufgrund des iPRP-Lösch-Telegrammes aus der Warteschlange gelöscht wurden.
- **Scanning Sync-Status** (Nur im Client-Modus)
  Damit nicht beide Clients gleichzeitig nach einem Access Point suchen und in den Scan-Modus wechseln, synchronisieren sie sich.
  Folgende Stati kann die Synchronisation haben:
  - idle: Leerlauf. Kein Scannen.
  - requested: Anfrage an den Partner-Client, ob Scannen möglich ist.
  - pending: Scannen ist möglich. Wartet auf den Beginn des Scannens und wechselt dann in den Status "foreground" oder "background".
  - background: Background-Scan wird durchgeführt.
  - foreground: Client ist z. B. gerade angelaufen und führt einen Foreground-Scan durch.

System-Funktionen projektieren

Konfiguration

Legen Sie fest, über welche Dienste auf das Gerät zugegriffen wird. Zu einigen Diensten gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind.

Einstellungen

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes.

- **Telnet-Server**
  Aktivieren oder deaktivieren Sie den Dienst "Telnet-Server" für den unverschlüsselten Zugriff auf das CLI.
- **SSH-Server**
  Aktivieren oder deaktivieren Sie den Dienst "SSH-Server" für den verschlüsselten Zugriff auf das CLI.
- **Nur HTTPS-Server**
  Aktivieren oder deaktivieren Sie den Zugriff über HTTP.
- **DNS-Client**
  Aktivieren oder deaktivieren Sie den DNS-Client. Weitere Einstellungen konfigurieren Sie unter "System > DNS Client".
- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client. Weitere Einstellungen konfigurieren Sie unter "System > SMTP-Client".

- **Syslog-Client**
  Aktivieren oder deaktivieren Sie den Systemereignis-Client. Weitere Einstellungen konfigurieren Sie unter "System > Syslog-Client".
- **DCP-Server**
  Legen Sie fest, ob auf das Gerät mit DCP (Discovery and Configuration Protocol) zugegriffen werden kann:
  - "-" (Deaktiviert)
    DCP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben
    Mit DCP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt
    Mit DCP können Geräteparameter zwar gelesen aber nicht verändert werden.
- **Zeiteinstellung**
  Wählen Sie die gewünschte Einstellung aus. Folgende Einstellungen sind möglich:
  - Manuell
    Die Systemzeit wird manuell eingestellt.
  - SIMATIC Time
    Die Systemzeit wird über einen SIMATIC Zeitgeber eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SIMATIC Time Client".
  - SNTP-Client
    Die Systemzeit wird über einen SNTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SNTP-Client".
  - NTP-Client
    Die Systemzeit wird über einen NTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > NTP-Client".
- **SNMP**
  Wählen Sie das gewünschte Protokoll. Folgende Einstellungen sind möglich:
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist mit der SNMP Version 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
- **SNMPv1/v2 schreibgeschützt**
  Aktivieren oder deaktivieren Sie den schreibenden Zugriff auf SNMP-Variablen bei SNMPv1/v2c.
- **SNMPv1 Traps**
  Aktivieren oder deaktivieren Sie das Versenden von SNMP Traps (Alarmtelegramme). Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Traps".
- **DHCP-Client**
  Aktivieren oder deaktivieren Sie den DHCP-Client. Weitere Einstellungen konfigurieren Sie unter "System > DHCP-Client".

- **DHCPv6-Client**
  Aktivieren oder deaktivieren Sie den DHCPv6-Client.
- **SINEMA-Konfigurationsschnittstelle**
  Wenn die SINEMA Konfigurationsschnittstelle aktiviert ist, können Sie Konfigurationen über das TIA-Portal auf das Gerät laden.
- **Konfigurationsmodus**
  Wählen Sie die gewünschte Betriebsart. Folgende Betriebsarten sind möglich:
  – Automatisches Speichern
    Automatischer Sicherungsbetrieb. Ca. 1 Minute nach der letzten Parameteränderung oder beim Neustart des Geräts wird die Konfiguration automatisch abgespeichert. Zusätzlich wird im Anzeigebereich die Meldung "Die Änderungen werden automatisch in x Sekunden gespeichert. Um die Änderungen sofort zu speichern, klicken Sie auf 'Schreiben der Startkonfiguration."

    Hinweis

    Unterbrechung des Speichervorgangs

    Der Speichervorgang startet erst, nachdem der Timer in der Meldung abgelaufen ist. Die Dauer des Speichervorgangs ist vom Gerät abhängig.

    Während des Speichervorgangs wird die Meldung "Die Konfigurationsdaten werden gespeichert. Schalten Sie das Gerät nicht aus." angezeigt.

    - Schalten Sie das Gerät nicht sofort aus, nachdem der Timer abgelaufen ist.

  – Trial
    Trial-Modus. Im Trial-Modus werden Änderungen zwar übernommen aber nicht in der Konfigurationsdatei (Startup Configuration) gespeichert.
    Um Änderungen in der Konfigurationsdatei abzuspeichern, verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration". Die Schaltfläche "Schreiben der Startkonfiguration" wird eingeblendet, wenn Sie den Trial-Modus einstellen. Zusätzlich wird nach jeder Parameteränderung im Anzeigebereich folgende Meldung angezeigt: "Der Konfigurationsmodus Trial ist aktiv - Klicken Sie auf die Schaltfläche "Schreiben der Startkonfiguration" um Ihre Einstellungen zu speichern" angezeigt sobald es ungespeicherte Änderungen gibt. Diese Meldung ist auf jeder WBM-Seite sichtbar, bis die vorgenommenen Änderungen entweder gespeichert werden oder das Gerät neu gestartet wird.

Allgemein

Gerät

Diese Seite enthält die allgemeinen Geräteinformationen.

Beschreibung

Die Seite enthält folgende Felder:

- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemuhrzeit an. Die Systemuhrzeit wird entweder vom Anwender eingestellt oder per Uhrzeittelegramm synchronisiert: entweder SINEC H1 Uhrzeittelegramm, NTP oder SNTP. (Nur lesbar)
- **Systembetriebszeit** (nur Online verfügbar)
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an. (Nur lesbar)
- **Gerätetyp** (nur Online verfügbar)
  Zeigt die Typenbezeichnung des Geräts an. (Nur lesbar)
- **Systemname**
  Sie können den Namen des Gerätes eintragen. Der eingetragene Name wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich. Der Systemname wird auch in der CLI-Eingabeaufforderung (Prompt) angezeigt. In der CLI-Eingabeaufforderung ist die Anzahl der Zeichen begrenzt. Der Systemname wird nach 16 Zeichen abgeschnitten.
- **Kontaktperson**
  Sie können den Namen einer Kontaktperson eintragen, die für die Verwaltung des Gerätes zuständig ist. Es sind maximal 255 Zeichen möglich.
- **Gerätestandort**
  Sie können den Montageort des Gerätes eintragen. Der eingetragene Montageort wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich.

Hinweis

In den Eingabefeldern werden die ASCII-Zeichen 0x20 bis 0x7e verwendet.

Koordinaten

Auf dieser Seite können Informationen über die geografischen Koordinaten (Breitengrad, Längengrad und die Höhe über dem Ellipsoid gemäß WGS84) eingetragen werden.

Ermittlung der Koordinaten

Nutzen Sie zur Ermittlung der geografischen Koordinaten des Gerätes entsprechendes Kartenmaterial.

Die geografischen Koordinaten können auch durch einen GPS-Empfänger ermittelt werden. Meist werden die geografischen Koordinaten von diesen Geräten direkt angezeigt und müssen nur noch in die Eingabefelder dieser Seite übertragen werden.

Beschreibung

Die Seite enthält folgende Eingabefelder mit einer maximalen Länge von 32 Zeichen:

- **Eingabefeld "Geographische Breite"**
  Geben Sie die nördliche oder die südliche Breite für den Standort des Gerätes ein.
  Z. B. +49° 1´ 31.67" bedeutet, dass sich das Gerät auf 49 Grad, 1 Bogenminute und 31.67 Bogensekunden nördlicher Breite befindet.
  Die südliche Breite wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben N (nördliche Breite) oder S (südliche Breite) an die Zahlenangabe anhängen (49° 1´ 31.67" N).
- **Eingabefeld "Geographische Länge"**
  Geben Sie die östliche oder die westliche Länge für den Standort des Gerätes ein.
  Z. B. +8° 20´ 58.73" bedeutet, dass sich das Gerät auf 8 Grad, 20 Bogenminuten und 58.73 Bogensekunden östliche Länge befindet.
  Die westliche Länge wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben O bzw. E (östliche Länge) oder W (westliche Länge) an die Zahlenangabe anhängen (8° 20´ 58.73" E).
- **Eingabefeld "Geographische Höhe"**
  Geben Sie die geografische Höhe über oder unter normal Null (Meereshöhe) in Metern ein.
  Z.B. 158 m bedeutet, dass sich das Gerät in einer Höhe von 158 m über normal Null befindet.
  Höhenangaben unterhalb von normal Null (z. B. am toten Meer) werden mit einem führenden Minuszeichen dargestellt.

Agent IPv4

Konfiguration der IP-Adressen

Auf dieser Seite konfigurieren Sie die IPv4-Adresse für das Gerät.

Einstellungen

- **Methode der IP-Adresszuweisung**
  Zeigt an, wie die IPv4-Adresse zugeordnet wird.
  - Statisch
    Die IPv4-Adresse ist statisch. Die IP-Einstellungen tragen Sie bei"IP-Adresse" und "Subnetzmaske" ein.
  - Dynamisch (DHCP)
    Das Gerät bezieht eine dynamische IPv4-Adresse von einem DHCP-Server.
- **IP-Adresse**
  Tragen Sie die IP-Adresse des Geräts ein. Wenn Sie die IPv4-Adresse ändern, sollte sich der Webbrowser automatisch auf die neue Adresse einstellen. Wenn dies nicht geschieht, geben Sie die neue Adresse manuell im Webbrowser ein.
- **Subnetzmaske**
  Tragen Sie die Subnetzmaske des Geräts ein.
- **Standard-Gateway**
  Tragen Sie die IPv4-Adresse des Standard-Gateways ein, um mit Geräten in einem anderen Subnetz zu kommunizieren, z. B. Diagnosestationen, E-Mail-Server.

- **Agent VLAN-ID**
  Wählen Sie aus der Klappliste die VLAN ID. Die Klappliste ist nur verfügbar, wenn beim Parameter "Base Bridge Mode" "802.1 Q VLAN Bridge" eingestellt ist. Den Parameter konfigurieren Sie unter "Layer 2 > VLAN > Allgemein". Sie können nur aus bereits konfigurierten VLANs auswählen.

  Hinweis
  Ändern der Agent VLAN-ID

  Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

- **MAC-Adresse** (nur Online verfügbar)
  Zeigt die MAC-Adresse des Geräts an. Die MAC-Adresse ist hardwaregebunden und kann nicht geändert werden.

Agent IPv6

Agent IPv6

Konfiguration der IP-Adressen

Auf dieser Seite aktivieren Sie IPv6 auf dem Management VLAN. Diese VLAN-Schnittstelle wird auch als IPv6-Schnittstellen bezeichnet. Eine IPv6-Schnittstelle kann mehrere IPv6-Adressen haben.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle**
  Zeigt die VLAN-Schnittstelle an, an der IPv6 aktiviert wird.
- **IPv6 aktivieren**
  Aktivieren oder deaktivieren Sie IPv6 auf der Schnittstelle. Wenn Sie die Einstellung aktivieren und die Einstellung übernehmen, wird automatisch die link lokale-Adresse erstellt.
- **IPv6-Adresse**
  Geben Sie die IPv6-Adresse an. Die Eingabe ist abhängig von der gewählten Adressart.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.
- **IPv6-Adresstyp**
  Wählen Sie die Adressart aus:
  - Unicast
  - Link lokal: IPv6-Adresse ist nur auf dem Link gültig.

- **Adresskonfiguration**
  Legen Sie den Mechanismus für die Adresskonfiguration fest:
  - Automatisch (Default)
    Die IPv6-Adresse wird entweder über einen zustandslosen Mechanismus oder einen zustandsbehafteten Mechanismus erstellt.
  - DHCPv6
    Zustandsbehaftet: Bezieht die IPv6-Adresse und die Konfigurationsdatei vom DHCPv6-Server.
  - SLAAC (Stateless Address Auto Configuration)
    Zustandslose Autokonfiguration über NDP (Neighbor Discovery Protocol.
  - Statisch
    Geben Sie eine statische IPv6-Adresse ein.
- **Rapid Commit**
  Wenn aktiviert, wird das Verfahren für die IPv6-Adresszuweisung verkürzt. Anstatt 4 DHCPv6-Nachrichten (SOLICIT, ADVERTISE , REQUEST, REPLY) werden nur 2 DHCPv6-Nachrichten (SOLICIT, REPLY) verwendet. Weitere Informationen zu den Nachrichten finden Sie in der RFC 3315.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Schnittstellenname**
  Zeigt den Namen der VLAN-Schnittstelle an.
- **IPv6-Adresse**
  Zeigt die IPv6-Adresse an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **IPv6-Adresstyp**
  Zeigt den Adressentyp an. Folgende Werte sind möglich:
  - Unicast
  - Link lokal

IPv6 Default-Routen

Auf dieser Seite konfigurieren Sie die IPv6-Defaultroute. Die IPv6-Defaultroute ist eine IPv6-Route, die für alle IPv6-Adressen zutrifft. Das Gerät muss nur das Default-Gateway kennen und schickt alle IPv6-Pakete dahin.

Das Default-Gateway kennt entweder selbst alle Routen oder hat eine Default-Route zu einem weiteren Default-Gateway

Beschreibung

Die Seite enthält Folgendes:

- **Zielnetzwerk**
  Das Zielnetzwerk (:: oder 0:0:0:0:0:0:0:0:0:0:0:0:0:0:0:0) gilt für alle IPv6-Adressen.
- **Präfixlänge**
  Geben Sie die Anzahl der linksseitigen Bits an, die dem Präfix angehören.
- **Gateway**
  Geben Sie die IPv6-Adresse des Gateways ein, an den die IPv6-Pakete gesendet werde.
- **Metrik**
  Geben Sie die Metrik für die Route ein. Die Metrik entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Metrik-Wert benutzt.
  Wertebereich: 1 - 254
- **Schnittstelle**
  Legen Sie die Schnittstelle fest, über das die Netzwerkadresse des Ziels erreicht wird.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Zielnetzwerk**
  Zeigt die Netzwerkadresse des Ziels an.
- **Präfixlänge**
  Zeigt die Präfixlänge an.
- **Gateway**
  Zeigt die IPv6-Adresse des nächsten Gateways an.
- **Schnittstelle**
  Zeigt die Schnittstelle der Route an.
- **Administrative Distanz**
  Tragen Sie die Metrik für die Route ein. Beim Erstellen der Route wird automatisch "not used" eingetragen. Die Metrik entspricht der Güte einer Verbindung, basierend z. B. auf Geschwindigkeit oder Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Metrik-Wert benutzt.
  Wertebereich: 1 - 254
- **Status**
  Zeigt an, ob die Route aktiv ist oder nicht.

DNS-Client

Auf dieser Seite können Sie bis zu 3 DNS-Server mit IPv4- oder IPv6-Adressen manuell konfigurieren. Manuell konfigurierten DNS-Servern wird jeweils ein Index von 1 bis 3 zugeordnet.

Bei mehreren DNS-Servern wird durch Sie die Reihenfolge in der Tabelle festlegen, in welcher Reihenfolge die Server angefragt werden.

Der obere Server wird zuerst angefragt. Der DNS-Server (Domain Name System) ordnet einer IP-Adresse einen Domainnamen zu, sodass ein Gerät eindeutig identifiziert werden kann. Wenn diese Funktion aktiviert ist, kann das Gerät als DNS-Client mit einem DNS-Server kommunizieren. Sie haben die Möglichkeit Namen in IP-Adressfeldern einzutragen.

Hinweis

Um die Funktion "DNS-Client" zu nutzen, muss sich ein DNS-Server im Netzwerk befinden.

Einstellungen

- **DNS-Client**
  Aktivieren oder deaktivieren Sie, dass das Gerät als DNS-Client fungiert.
- **Verwendete DNS-Server**
  - learned only
    Das Gerät verwendet nur die durch DHCP zugewiesenen DNS-Server.
  - manual only
    Das Gerät verwendet nur die manuell projektierten DNS-Server. Die DNS-Server müssen mit dem Internet verbunden sein.
  - all
    Das Gerät verwendet alle verfügbaren DNS-Server.
- **Adresse des DNS-Servers**
  Tragen Sie die IP-Adresse des DNS-Servers ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Adresse des DNS-Servers**
  Zeigt die IP-Adresse des DNS-Servers an.
- **Erstellung**
  Hier wird angezeigt, ob der DNS-Server manuell konfiguriert wurde oder durch DHCP zugewiesen wurde.

Neustart

Auf dieser Seite finden Sie eine Schaltfläche zum Neustart des Gerätes sowie verschiedene Möglichkeiten, die Voreinstellungen des Gerätes zurückzusetzen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Hinweis

Beachten Sie folgende Punkte beim Neustart eines Gerätes:

- Sie können einen Neustart des Gerätes nur mit Administrator-Rechten durchführen.
- Der Neustart eines Gerätes sollte nur durch die Schaltflächen dieser Seite oder durch die entsprechenden CLI-Befehle und nicht durch Aus- und Einschalten der Spannungsversorgung am Gerät erfolgen.
- Vorgenommene Änderungen werden erst nach dem Anklicken der Schaltfläche "Einstellungen übernehmen" auf der jeweiligen Seite im Gerät wirksam. Wenn sich das Gerät im "Trial-Modus" befindet, müssen Konfigurationsänderungen vor einem Neustart manuell abgespeichert werden. Im Modus "Automatisch Speichern" werden die letzten Änderungen automatisch vor einem Neustart gespeichert.

Beschreibung

Für den Neustart gibt es folgende Möglichkeiten:

- **Neustart**
  Klicken Sie auf diese Schaltfläche, um das System neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. Bei einem Neustart wird das Gerät neu initialisiert, die interne Firmware wird neu geladen und das Gerät führt einen Selbsttest durch. Die gelernten Einträge in der Adresstabelle werden gelöscht. Sie können das Browser-Fenster geöffnet lassen, während das Gerät neu startet. Sie müssen sich wieder neu anmelden.
- **Auf gespeicherte Einstellungen zurücksetzen und Neustart**
  Klicken Sie diese Schaltfläche, um die werkseitigen Konfigurationseinstellungen mit Ausnahme der folgenden Parameter zurückzusetzen und einen Neustart auszuführen:
  - IP-Adressen
  - Subnetz-Maske
  - IP-Adresse des Standard-Gateways
  - DHCP Client ID
  - DHCP
  - Systemname
  - System-Aufstellungsort
  - System-Ansprechpartner
  - Betriebsart des Geräts
- **Auf Werkseinstellungen zurücksetzen und Neustart**
  Klicken Sie auf diese Schaltfläche, um die werkseitigen Konfigurationseinstellungen wiederherzustellen. Es werden auch die geschützten Voreinstellungen zurückgesetzt. Es wird ein automatischer Neustart ausgeführt.

Hinweis

Durch das Zurücksetzen auf die werkseitigen Konfigurationseinstellungen verliert das Gerät die IP-Adresse. Das Gerät ist danach nur über das Primary Setup Tool oder über DHCP ansprechbar.

Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

Verwaltung von Änderungen

Beschreibung

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mit dieser Seite legen Sie fest, wann die WLAN-Einstellungen auf dem Gerät wirksam werden. Wenn Sie eine WLAN-Einstellung ändern und die Änderung bestätigen, wird diese Änderung

übernommen und ist sofort wirksam. Hierfür wird die WLAN-Verbindung kurzzeitig unterbrochen. Somit können sie die WLAN-Verbindung zu ihrem Gerät verlieren, bevor es komplett konfiguriert ist.

Die Einstellung "Manuell Anwenden" bietet Ihnen die Möglichkeit, das Gerät erst fertig zu konfigurieren. Die Änderungen werden zwar übernommen, sind aber nicht sofort wirksam. Die Änderungen werden erst wirksam, wenn Sie mit der Schaltfläche "Einstellungen übernehmen" die Änderungen bestätigen.

Hinweis

Wenn Sie das Gerät über die WLAN-Schnittstelle konfigurieren, wird empfohlen die Einstellung "Manuell Anwenden" zu verwenden. Überprüfen Sie nochmals die Parameter, bevor Sie die Änderungen mit der Schaltfläche "Einstellungen übernehmen" bestätigen.

Einstellungen

- **Änderungsmodus**
  Wählen Sie die gewünschte Einstellung aus:
  - Automatisch anwenden
    Jede Änderung in den WLAN-Einstellungen wird übernommen und ist sofort wirksam, wenn Sie auf die Schaltfläche "Einstellungen übernehmen" klicken. Das SCALANCE W700-Gerät ist in der Grundeinstellung auf "Automatisch Anwenden" eingestellt.
  - Manuell anwenden
    Die Änderungen werden übernommen, sind aber nicht sofort wirksam. Die Änderungen werden erst wirksam, wenn Sie auf die Schaltfläche "Einstellungen übernehmen" klicken. Die Schaltfläche "Änderungen anwenden" wird eingeblendet, wenn Sie "Manuell Anwenden" einstellen. Zusätzlich wird im Anzeigebereich die Meldung "Änderungsmodus 'Manuell anwenden' aktiviert. - Klicken Sie die Schaltfläche "Einstellungen übernehmen" um die aktuelle Konfiguration zu übernehmen." angezeigt, sobald es WLAN-Änderungen gibt. Diese Meldung ist auf jeder WBM-Seite sichtbar, bis die vorgenommenen Änderungen entweder wirksam werden oder das Gerät neu gestartet wird.

    Hinweis

    Wenn die Änderungen wirksam werden, wird die WLAN-Verbindung an allen WLAN-Schnittstellen für kurze Zeit unterbrochen. Der WLAN-Treiber wird mit den neuen Einstellungen gestartet.

Laden & Speichern

Hinweis

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Dateien aus dem Gerät geladen werden können.

Übersicht der Dateitypen

Tabelle 1-151 HTTP

Dateityp	Beschreibung	Laden	Speichern	Löschen
Config	Diese Datei enthält die Startkonfiguration. Diese Datei enthält unter anderem die Definitionen der Benutzer, Rollen, Gruppen und Funktionsrechte. Die Passwörter sind in der Datei "Users" abgespeichert.	X	X	--
ConfigPack	Detaillierte Konfigurationsinformationen z. B. Startkonfiguration, Benutzer, Zertifikate, Favoriten Firmware des Geräts (wenn mitgespeichert). Nähere Informationen zur Erstellung und Benutzung des ConfigPack inkl. Firmware finden Sie in Kapitel "Instandhalten und Warten".	X	X	--
CountryList	Die Zip-Datei enthält die Länderliste als csv- und als pdf-Datei.	--	X	--
Debug	Diese Datei beinhaltet Informationen für den Siemens Support. Sie ist verschlüsselt und kann ohne Sicherheitsrisiko per E-Mail an den Siemens Support gesendet werden.	--	X	X
EDS	Electronic Data Sheet (EDS) Elektronische Datenblätter zur Beschreibung von Geräten im EtherNet/IP-Betrieb	--	X	--
Firmware	Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.	X	X	--
GSDML	Informationen über die Geräteeigenschaften (PROFINET)	--	X	--
HTTPS Cert	HTTPS-Zertifikat Maximale Dateigröße: 8192 Bit	X	X	X
LogFile	Datei mit Einträgen aus der Ereignisprotokolltabelle	--	X	--
MIB	Private MSPS MIB-Datei "Scalance_w_msps.mib"	--	X	--
RunningCLI	Textdatei mit CLI-Befehlen Diese Datei enthält eine Übersicht der aktuellen Konfiguration in Form von CLI-Befehlen. Passwörter sind in dieser Datei wie folgt maskiert: [PASSWORD] Sie können die Textdatei herunterladen. Die Datei ist nicht dafür vorgesehen, dass Sie sie unverändert wieder hochladen.	--	X	--
Script	Textdatei mit CLI-Befehlen	X	--	--
StartupInfo	Startup Logdatei Diese Datei enthält die Meldungen die während des letzten Hochlaufs im Logdatei eingetragen wurden.	--	X	--
Users	Datei mit Benutzernamen und Passwörter	X	X	--

Dateityp	Beschreibung	Laden	Speichern	Löschen
WBMFav	WBM Favoriten Diese Datei enthält die Favoriten, die Sie im WBM angelegt haben. Sie können diese Datei herunterladen und in anderen Geräten hochladen.	X	X	X
WLANAuth-log	Datei mit Einträgen aus dem WLAN Authentication Log (Informationen zu erfolgreichen oder fehlgeschlagenen Authentifizierungsversuchen)	--	X	--
WLANCert (Nur im Client-Modus)	Benutzerzertifikat. Für das Benutzerzertifikat können Sie auf der WBM-Seite "Load&Save > Password" ein Passwort festlegen. Maximale Dateigröße: 8192 Bit	X	X	X
WLAN-ServCert (Nur im Client-Modus)	Serverzertifikat Maximale Dateigröße: 8192 Bit	X	X	X
WLANSigRec (Nur im Client-Modus)	Die Zip-Datei enthält Folgendes: • csv-Datei mit den Messwerten des Signalrekorders • pdf-Datei mit den Messwerten und einer zusätzlichen grafischen Darstellung der Messwerte. Die Informationen zu den Messwerten und deren grafische Darstellung finden Sie im Kapitel "Signalrekorder".	--	X	X
WLAN-SpectrumAnalyzer (Nur im Access Point-Modus)	Die Zip-Datei enthält eine csv-Datei mit den Messwerten des Spektrumanalysators. Die Informationen zu den Messwerten und deren grafische Darstellung finden Sie im Kapitel "Spektrumanalysator".	--	X	X

Tabelle 1-152 TFTP/SFTP

Dateityp	Beschreibung	Speichern	Laden
Config	Diese Datei enthält die Startkonfiguration. Diese Datei enthält unter anderem die Definitionen der Benutzer, Rollen, Gruppen und Funktionsrechte. Die Passwörter sind in der Datei "Users" abgespeichert.	X	X
ConfigPack	Detaillierte Konfigurationsinformationen z. B. Startkonfiguration, Benutzer, Zertifikate, Firmware des Geräts (wenn mitgespeichert). Nähere Informationen zur Erstellung und Benutzung des ConfigPack inkl. Firmware finden Sie in Kapitel "Instandhalten und Warten".	X	X
CountryList	Die Zip-Datei enthält die Länderliste als csv- und als pdf-Datei.	X	--

Dateityp	Beschreibung	Speichern	Laden
Debug	Diese Datei beinhaltet Informationen für den Siemens Support.Sie ist verschlüsselt und kann ohne Sicherheitsrisiko per E-Mail an den Siemens Support gesendet werden.	X	--
EDS	Electronic Data Sheet (EDS) Elektronische Datenblätter zur Beschreibung von Geräten im EtherNet/IP-Betrieb	X	--
Firmware	Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.	X	X
GSDML	Informationen über die Geräteeigenschaften (PROFINET)	X	--
HTTPS Cert	Voreingestellte HTTPS-Zertifikate inkl. Schlüssel Die voreingestellten und automatisch erstellten HTTPS-Zertifikate sind selbst-signiert. Es wird dringend empfohlen eigene HTTPS-Zertifikate zu erstellen und bereitzustellen. Es wird empfohlen HTTPS-Zertifikate zu verwenden, die entweder durch eine zuverlässige externe oder eine interne Zertifizierungsstelle signiert sind. Das HTTPS-Zertifikat überprüft die Identität des Geräts und regelt den verschlüsselten Datenaustausch. Es gibt Dateien, deren Zugriff passwortgeschützt ist. Um die Datei erfolgreich ins Gerät zuladen, geben Sie auf der WBM-Seite "Passwörter" das für die Datei festgelegte Passwort ein. Maximale Dateigröße: 8192 Bit	X	X
LogFile	Datei mit Einträgen aus der Ereignisprotokolltabelle	X	--
MIB	Private MSPS MIB-Datei "Scalance_w_msps.mib"	X	--
RunningCLI	Textdatei mit CLI-Befehlen Diese Datei enthält eine Übersicht der aktuellen Konfiguration in Form von CLI-Befehlen. Passwörter sind in dieser Datei wie folgt maskiert: [PASSWORD] Sie können die Textdatei herunterladen. Die Datei ist nicht dafür vorgesehen, dass Sie sie unverändert wieder hochladen.	X	--
Script	Textdatei mit CLI-Befehlen Sie können eine Skriptdatei in einem Gerät hochladen. Die enthaltenen CLI-Befehle werden entsprechend ausgeführt. CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei ausgeführt werden.	--	X
StartupInfo	Startup Logdatei Diese Datei enthält die Meldungen die während des letzten Hochlaufs im Logdatei eingetragen wurden.	X	--
Users	Datei mit Benutzernamen und Passwörter	X	X

Dateityp	Beschreibung	Speichern	Laden
WBMFav	WBM Favoriten Diese Datei enthält die Favoriten, die Sie im WBM angelegt haben. Sie können diese Datei herunterladen und in anderen Geräten hochladen.	X	X
WLANAuthlog	Datei mit Einträgen aus dem WLAN Authentication Log (Informationen zu erfolgreichen oder fehlgeschlagenen Authentifizierungsversuchen)	X	--
WLANCert (Nur im Client-Modus)	Benutzerzertifikat. Für das Benutzerzertifikat können Sie auf der WBM-Seite "Load&Save > Password" ein Passwort festlegen. Maximale Dateigröße: 8192 Bit	X	X
WLANServerCert (Nur im Client-Modus)	Server-Zertifikat Maximale Dateigröße: 8192 Bit	X	X
WLANSigRec (Nur im Client-Modus)	Die Zip-Datei enthält Folgendes: • csv-Datei mit den Messwerten des Signalrekorders • pdf-Datei mit den Messwerten und einer zusätzlichen grafischen Darstellung der Messwerte. Die Informationen zu den Messwerten und deren grafische Darstellung finden Sie im Kapitel "Signalrekorder".	X	--
WLANSpectrumAnalyzer (Nur im Access Point-Modus)	Die Zip-Datei enthält eine csv-Datei mit den Messwerten des Spektrumanalysators. Die Informationen zu den Messwerten und deren grafische Darstellung finden Sie im Kapitel "Spektrumanalysator".	X	--

Hochladen und Speichern über HTTP

Laden und speichern von Daten über HTTP

Diese Seite bietet die Möglichkeit, Gerätedaten in einer externen Datei auf Ihrem Client-PC zu speichern bzw. solche Daten aus einer externen Datei vom Client-PC in die Geräte zu laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Client-PC laden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.
Diese Seite ist sowohl für Verbindungen über HTTP als auch für Verbindungen über HTTPS verfügbar.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Trial-Modus/Automatisches Speichern

Im Automatic Save-Modus wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Trial-Modus werden Änderungen zwar übernommen, aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration " auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdatei abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Hochladen**
  Mit dieser Schaltfläche können Sie Dateien auf das Gerät hochladen. Die Schaltfläche ist aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird.
- **Speichern**
  Mit dieser Schaltfläche können Sie Dateien vom Gerät speichern. Die Schaltfläche ist nur aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird und die Datei auf dem Gerät vorhanden ist.
- **Löschen**
  Mit dieser Schaltfläche können Sie Dateien vom Gerät löschen. Die Schaltfläche ist nur aktivierbar, wenn diese Funktion von dem Dateityp unterstützt wird und die Datei auf dem Gerät vorhanden ist.

Hinweis

Löschen Sie nach einem Firmware-Update den Cache Ihres Internet-Browsers.

Hochladen und Speichern über TFTP

Laden und Speichern von Daten über einen TFTP-Server

Auf der Seite können Sie den TFTP-Server und die Dateinamen konfigurieren. Weiter können Sie Gerätedaten in einer externen Datei auf einem TFTP-Server speichern bzw. solche Daten aus einer externen Datei vom TFTP-Server in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von einem TFTP-Server laden.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Modus Trial/Automatisches Speichern

Im Modus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Modus "Trial" werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration" auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden.

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung

- **Adresse des TFTP-Servers**
  Tragen Sie die IP-Adresse oder den FQDN des TFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des TFTP-Servers**
  Tragen Sie hier den Port des TFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 69 entsprechend Ihren spezifischen Anforderungen ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis

  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktion** (nur Online verfügbar)
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp, z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - Datei speichern
    Mit dieser Auswahl speichern Sie eine Datei auf den TFTP-Server.
  - Datei hochladen
    Mit dieser Auswahl laden Sie eine Datei vom TFTP-Server, z. B. Zertifikate, die für den Aufbau einer gesicherten Verbindung notwendig sind.

Hochladen und Speichern über SFTP

Laden und Speichern von Daten über einen SFTP-Server

SFTP (SSH File Transfer Protocol) überträgt die Dateien verschlüsselt. Auf dieser Seite konfigurieren Sie die Zugangsdaten für den SFTP-Server.

Weiter können Sie Gerätedaten in einer externen Datei auf Ihrem Client-PC speichern bzw. solche Daten aus einer externen Datei vom PC in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Admin-PC laden.

Zudem lassen sich auf dieser Seite die Zertifikate laden, die für den Aufbau einer gesicherten VPN-Verbindung notwendig sind.

Firmware

Die Firmware ist signiert und verschlüsselt. Es ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Modus Trial/Automatisches Speichern

Im Modus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Modus "Trial" werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration" auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden

Beschreibung

Die Seite enthält folgende Felder:

- **Adresse des SFTP-Servers**
  Geben Sie die IP-Adresse oder den FQDN des SFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des SFTP-Servers**
  Geben Sie den Port des SFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 22 entsprechend Ihren spezifischen Anforderungen ändern.
- **SFTP Benutzer**
  Geben Sie den Benutzer für den Zugriff auf den SFTP-Server ein. Vorausgesetzt, auf dem SFTP-Server ist ein Benutzer mit den entsprechenden Rechten angelegt.
- **SFTP Passwort**
  Geben Sie das Passwort für den Benutzer ein
- **SFTP Passwort bestätigen**
  Bestätigen Sie das Passwort.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.

- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis
  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktionen**
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp. z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - **Datei speichern**
    Mit dieser Auswahl speichern Sie eine Datei auf dem SFTP-Server.
  - **Datei hochladen**
    Mit dieser Auswahl laden Sie eine Datei vom SFTP-Server, z. B. Zertifikate, die für den Aufbau einer gesicherten Verbindung notwendig sind.

Passwörter

Passwort für Dateien

Es gibt Dateien, deren Zugriff passwortgeschützt ist. Um die Datei erfolgreich ins Gerät zuladen, geben Sie auf der Seite das für die Datei festgelegte Passwort ein.

Beschreibung

- **Typ**
  Zeigt den Dateityp an
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Aktiviert**
  Wenn aktiviert, wird die Datei verwendet. Nur aktivierbar, wenn das Passwort konfiguriert ist
- **Passwort**
  Geben Sie das Passwort für die Datei ein.

- **Passwort bestätigen**
  Bestätigen Sie das Passwort.
- **Status** (nur Online verfügbar)
  Zeigt an, ob das Passwort zur der Datei auf dem Gerät passen.
  - gültig
    Das Optionskästchen "Aktiviert" ist aktiviert und das Passwort passt zu der Datei.
  - ungültig
    Das Optionskästchen "Aktiviert" ist aktiviert, aber das Passwort passt nicht zu der Datei oder es ist noch keine Datei geladen.
  - '-'
    Das Passwort kann nicht ausgewertet werden oder wird noch nicht verwendet. Das Optionskästchen "Aktiviert" ist nicht aktiviert.

Ereignisse

Konfiguration

Auf dieser Seite legen Sie fest, wie das Gerät auf Systemereignisse reagiert. Durch Aktivieren der entsprechenden Optionen legen Sie fest, wie das Gerät bei Ereignissen reagiert. Klicken Sie zum Aktivieren oder Deaktivieren der Optionen in die entsprechenden Einstellungen der jeweiligen Spalte.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ereignisse der Tabelle 2 gültig sind.
- **E-Mail / Trap / Log-Tabelle / Syslog / Fehler**
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Ereignis**
  Die Spalte enthält folgende Werte:
  - Kalt-/Warmstart
    Das Gerät wurde eingeschaltet oder vom Anwender neu gestartet.
  - Link Change
    Dieses Ereignis tritt nur auf, wenn der Port-Status überwacht wird und sich entsprechend geändert hat, siehe "System > Fehlerkontrolle > Link Change".
  - Authentifizierungsfehler
    Dieses Ereignis tritt beim Versuch eines Zugriffs mit fehlerhaften Kennwort auf.
  - Umschaltung der Spannungsversorgung
    Dieses Ereignis tritt auf, wenn die Spannungsversorgungsleitungen 1 und 2 überwacht werden. Es zeigt an, dass ein Wechsel auf Leitung 1 bzw. auf Leitung 2 stattgefunden hat. Das Ereignis tritt auch auf, wenn die PoE-Spannungsversorgung ausgefallen ist, siehe "System > Fehlerkontrolle > Spannungsversorgung".
  - Änderung im Spanning Tree
    Die STP- bzw. RSTP- oder MSTP-Topologie hat sich geändert.
  - Änderung im Fehlerstatus
    Der Fehlerstatus hat sich geändert. Der Fehlerstatus kann sich auf die aktivierte Portüberwachung, auf das Ansprechen der Meldekontakte oder die Spannungsüberwachung beziehen.
  - AP-Kanalüberlappung (Nur im Access Point-Modus)
    Bei einem Eintrag in die Liste "Überlappung AP" wird diese Ereignis ausgelöst.
  - WDS (Nur im Access Point-Modus)
    Der Verbindungsstatus einer WDS-Strecke hat sich geändert.
  - DFS (Nur im Access Point-Modus)
    Dieses Ereignis tritt ein, wenn ein Radarsignal empfangen worden ist oder der DFS-Scan gestartet bzw. gestoppt wurde.
  - WLAN Allgemein (Nur im Access Point-Modus)
    Dieses Ereignis tritt ein, wenn sich die Kanalbandbreite geändert hat.
  - WLAN-Authentifizierung Log
    Weiterleitung der Einträge aus WLAN Authentication Log an den Systemprotokoll-Server.
  - WLAN De-/Authentifizierung (Nur im Client-Modus)
    Bei erfolgten oder fehlgeschlagenen WLAN-Authentifizierungsversuchen.
  - iPCF-Zykluszeit (Nur im Access Point-Modus)
    Nur verfügbar, wenn der KEY-PLUG gesteckt ist.
    Dieses Ereignis tritt ein, wenn für die eingestellte iPCF Zykluszeit zu viele Clients angemeldet sind oder in einem Zyklus nicht alle Clients erreicht wurden.
  - iPCF Poll-Größe
    Nur verfügbar, wenn der KEY-PLUG gesteckt ist.
    Dieses Ereignis tritt ein, wenn die PROFINET-Datengröße zu groß für die Übertragung ist.

  - Änderung in der Konfiguration
    Dieses Ereignis tritt ein, wenn sich die Konfiguration des Geräts geändert hat.

- E-Mail
  Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- Trap
  Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.
- Log-Tabelle
  Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle.
- Syslog
  Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- Fehler
  Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf.
- E-Mail
  Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- Trap
  Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.
- Log-Tabelle
  Das Gerät schreibt einen Eintrag in die Log-Tabelle.
- Syslog
  Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- Fehler
  Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf

Severity-Filter

Stellen Sie auf dieser Seite die Schwellwertstufen für das Versenden von Systemereignisbenachrichtigungen ein.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Client-Typ**
  Wählen Sie den Client-Typ, für den Sie die Einstellungen vornehmen:
  - E-Mail
    Versand von Systemereignismeldungen per E-Mail
  - Log-Tabelle
    Eintragen von Systemereignissen in die Log-Tabelle.
  - Syslog
    Eintragen von Systemereignissen in die Syslog-Datei
  - WLAN-Authentifizierung Log
    Weiterleitung der Einträge aus dem WLAN Authentication Log an den Systemprotokoll-Server.
    Nur bei SCALANCE W 770 und SCALANCE W780 konfigurierbar
- **Severity**
  Wählen Sie die gewünschte Stufe aus. Folgende Einstellungen sind möglich:
  - Info
    Systemereignisse werden ab dem Severity-Level "Info" bearbeitet.
  - Warning
    Systemereignisse werden ab dem Severity-Level "Warning" bearbeitet.
  - Critical
    Systemereignisse werden ab dem Severity-Level "Critical" bearbeitet.

SMTP-Client

Netzüberwachung durch E-Mails

Das Gerät bietet die Möglichkeit, beim Auftreten eines Alarmereignisses automatisch eine E-Mail (z.B. an den Netzwerkadministrator) zu senden. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Damit kann für Netze mit wenigen Teilnehmern eine einfache zentrale Netzüberwachung auf Basis eines E-Mail-Systems aufgebaut werden. Bei eintreffenden E-Mail-Störmeldungen kann über die Identifikation des Absenders per Internet-Browser das WBM gestartet werden, um weitere Diagnoseinformationen auszulesen.

Auf dieser Seite können Sie bis zu drei SMTP-Server und die dazugehörigen E-Mail-Adressen konfigurieren.

Beschreibung

Die Seite enthält folgende Felder:

- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client.
- **E-Mail-Adresse des Absenders**
  Geben Sie den Absendernamen ein, der in der E-Mail angegeben werden soll, z. B. den Gerätenamen. Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **Test-E-Mail senden**
  Verschicken Sie eine Test-E-Mail, um Ihre Konfiguration zu prüfen.
- **SMTP-Port**
  Geben Sie den Port ein, über den Ihr SMTP-Server erreichbar ist.
  Werkseinstellung: 25
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **SMTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SMTP-Servers ein.

Die Tabelle enthält folgende Spalten:

- **SMTP-Server-Adresse**
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SMTP-Servers.
- **E-Mail-Adresse des Empfängers**
  Geben Sie die E-Mail-Adresse ein, an die das Gerät im Fehlerfall eine E-Mail sendet.

Syntax von E-Mail-Adressen

Folgenden Bedingungen gelten für E-Mail-Adressen:

- Alphanumerische Zeichen sind erlaubt.
- Folgende Sonderzeichen sind erlaubt:
  - @
  - _ (Unterstrich)
  - - (Bindestrich)
  - . (Punkt)
- Es muss ein @-Zeichen vorkommen.
- Es darf nur ein @-Zeichen vorkommen.
- Die Zeichen "@" und "." dürfen nicht das erste bzw. letzte Zeichen sein.

Hinweis

Je nach Eigenschaften und Konfiguration des SMTP-Servers kann es notwendig sein, das Eingabefeld "E-Mail-Adresse des Absenders" anzupassen. Informieren Sie sich beim Administrator des SMTP-Servers.

DHCPv4

DHCP-Client

Einstellung des DHCP-Modus

Wenn das Gerät als DHCP-Client konfiguriert ist, startet es eine DHCP-Anfrage. Das Gerät erhält vom DHCP-Server als Antwort eine IPv4-Adresse zugewiesen. Der Server verwaltet einen Adressbereich, aus welchem er IPv4-Adressen vergibt. Es ist auch möglich den Server so zu konfigurieren, dass der Client auf seine Anfrage immer dieselbe IPv4-Adresse zugewiesen bekommt.

Beschreibungen

- **DHCP-Client Konfigurationsanfrage (Opt. 66, 67)**
  Aktivieren Sie diese Option, wenn der DHCP Client die Optionen 66 und 67 dazu verwenden soll, eine Konfigurationsdatei herunterzuladen und diese dann zu aktivieren.
- **DHCP-Modus**
  Wählen Sie die DHCP-Betriebsart aus. Folgende Modi sind möglich:
  - über MAC-Adresse
    Die Identifikation läuft über die MAC-Adresse ab.
  - über DHCP-Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab.
  - über Systemname
    Die Identifikation läuft über den Systemnamen ab. Ist der Gerätename 255 Zeichen lang, dann wird das letzte Zeichen nicht zur Identifikation benutzt.
  - über PROFINET-Gerätename
    Die Identifikation läuft über den PROFINET-Gerätenamen der Station ab.
  - über IAID und DUID
    Damit kann sich der DHCP-Client an DHCP-Servern anmelden, die den Parallelbetrieb von IPv4-und IPv6 unterstützten.
    Die Identifikation läuft über die IAID und die DUID ab und bezeichnet genau eine IP-Schnittstelle des Geräts bezeichnet.
    IAID (Interface Association Identifier): Für jede IP-Schnittstelle wird mindestens eine IAID generiert. Die IAID bleibt bei Neustart des DHCP-Clients unverändert.
    DUID (DHCP Unique Identifier): Identifiziert Server und Clients eindeutig und gilt für alle IP-Schnittstellen des Geräts. Die DUID bleibt bei Neustart unverändert. Es sei denn, der Benutzer ändert diese.

- **DUID-Typ**
  Legen Sie fest, welcher DUID-Typ verwendet wird. Die DUID-Typen sind in der RFC 3315 definiert.
  - DUID-LLT
    DUID basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel
  - DUID-EN
    DUID wird vom Hersteller (EN = Enterprise number) vergeben
  - DUID-LL
    DUID basiert auf der Link-layer-Adresse der Schnittstelle
- **Link-layer Adresse plus Zeit (LLT)**
  Der Wert basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Unternehmensnummer des Herstellers (EN)**
  Der Wert basiert auf der Unternehmensnummer, die spezifisch für den Hersteller ist. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Link-layer Adresse (LL)**
  Die Link-layer-Adresse basiert auf der MAC-Adresse. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.
- **DHCP**
  Aktivieren oder deaktivieren Sie den DHCP-Client für die entsprechende Schnittstelle.
- **IAID-Wert**
  Wert mit dem sich die Schnittstelle (DHCP-Client) am DHCP-Server identifiziert.

DHCP-Server

Hinweis

Die Seite ist nur verfügbar bei

- SCALANCE W780 / W770 / W740 / W730, W761-1 RJ45 und W722-1 RJ45

Das Gerät können Sie als DHCP-Server betreiben. Damit ist es möglich, den angeschlossenen Geräten automatisch IPv4-Adressen zuzuweisen. Die IPv4-Adressen werden entweder dynamisch aus einem von Ihnen vergebenen IPv4-Adressband verteilt oder es wird eine bestimmte IPv4-Adresse (statisch) einem bestimmten Gerät zugewiesen.

Auf dieser Seite legen Sie das IPv4-Adressband fest, aus dem das Gerät eine beliebige IPv4-Adresse erhält. Die statische Zuordnung der IPv4-Adressen konfigurieren Sie unter "Statische IP-Adressen".

Voraussetzung

- Im Access Point-Modus
  - Die angeschlossenen Geräte sind so konfiguriert, dass diese die IPv4-Adresse von einem DHCP-Server beziehen.
- Im Client Modus
  - Die angeschlossenen Geräte sind so konfiguriert, dass diese die IPv4-Adresse von einem DHCP-Server beziehen.
  - NAT ist aktiviert. NAT aktivieren Sie unter "Layer 3 > NAT".

Einstellungen

- **DHCP-Server**
  Aktivieren oder deaktivieren Sie den DHCP-Server auf dem Gerät.

  Hinweis

  Damit keine Konflikte mit IPv4-Adressen entstehen, darf im Netzwek nur ein Gerät als DHCP-Server konfiguriert sein.

  Hinweis
  Access Point

  Beim Access Point ist die Funktion "DHCP-Server" nur an dem Management zugeordneten VLAN (Agent VLAN ID) möglich.

- **Adresse vor dem Anbieten mit ICMP-Echo prüfen**
  Wenn aktiviert, prüft der DHCP-Server, ob die IPv4-Adresse schon vergeben ist. Dazu sendet der DHCP-Server ICMP-Echomeldungen (ping) an die IPv4-Adresse. Wenn keine Antwort zurückkommt, kann der DHCP-Server die IPv4-Adresse vergeben.

  Hinweis

  Wenn es in Ihrem Netzwerk Geräte gibt, bei denen der Echo-Dienst standardmäßig deaktiviert ist, kann es zu Konflikten bei den IPv4-Adressen kommen. Um dies zu vermeiden, vergeben Sie diesen Geräten eine IPv4-Adresse, die außerhalb des IPv4-Adressbands liegt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.

  Hinweis

  Es kann nur eine Pool ID (ID = 1) angelegt werden.

- **Aktivieren**
  Legen Sie fest, ob dieses IPv4-Adressband verwendet wird.

  Hinweis

  Wenn das IPv4-Adressband aktiviert ist, sind weder DHCP-Optionen noch statische IPv4-Adressen konfigurierbar.

- **Schnittstelle**
  Legen Sie die Schnittstelle fest, über die die IPv4-Adressen dynamisch vergeben werden. Voraussetzung für die Vergabe ist, dass die IPv4-Adresse der Schnittstelle innerhalb des IPv4-Adressbandes liegt. Wenn das nicht der Fall ist, vergibt die Schnittstelle keine IPv4-Adressen.
- **Subnetz**
  Geben Sie den Netzadressbereich ein, die den Geräten zugewiesen wird. Verwenden Sie die CIDR-Schreibweise.
- **Untere IP-Adresse**
  Geben Sie die IPv4-Adresse ein, die den Anfang des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Obere IP-Adresse**
  Geben Sie die IPv4-Adresse ein, die das Ende des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Gültigkeitsdauer [Sek]**
  Legen Sie fest, für wie viele Sekunden die vergebene IPv4-Adresse gültig bleibt. Nach Ablauf dieser Zeitdauer muss das Gerät entweder eine neue IPv4-Adresse anfordern oder die Gültigkeitsdauer der vorhandenen IPv4-Adresse verlängern.

DHCP-Optionen

Hinweis

Die Seite ist nur verfügbar bei

- SCALANCE W780 / W770 / W740 / W730, W761-1 RJ45 und W722-1 RJ45

Auf dieser Seite legen Sie fest, welche DHCP-Optionen der DHCP-Server unterstützt. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert.

Einstellungen

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Optionswert**
  Geben Sie die Nummer der gewünschten DHCP-Option ein. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert. Die DHCP-Optionen 1, 3, 6, 66 und 67 werden automatisch beim Erstellen des IPv4-Adressbands angelegt. Mit Ausnahme der Option 1 sind die Optionen löschbar.
  Bei der DHCP-Option 3 wird automatisch die interne IPv4-Adresse des Gerätes als DHCP-Parameter eingestellt.

  Hinweis

  Nicht unterstützte DHCP-Server

  Die DHCP-Optionen 50 - 60 und 255 werden nicht unterstützt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Optionswert**
  Zeigt Nummer der DHCP-Option an.
- **Schnittstellen-IP verwenden**
  Legen Sie fest, ob die interne IPv4-Adresse des Geräts verwendet wird oder nicht.
- **Wert**
  Geben Sie den DHCP-Parameter ein, der dem DHCP-Client übergeben wird. Der Inhalt ist abhängig von der DHCP-Option.
  - DHCP-Option 67 (Bootfilename)
    Geben Sie den Namen der Bootdatei im String-Format an.
  - DHCP-Optionen 3 (Router) und 6 (DNS):
    Geben Sie den DHCP-Parameter als IPv4-Adresse an, z. B. 192.168.100.2. Bei der DHCP-Option 6 können Sie mehrere IPv4-Adressen durch Komma getrennt angeben.
  - DHCP-Option 66 (TFTP-Server):
    Geben Sie den DHCP-Parameter als IPv4-Adresse, z. B. 192.168.100.2, oder den FQDN-Namen an.
  - Alle anderen DHCP-Optionen
    Geben Sie den DHCP-Parameter in Hexadezimal an, z. B. die IPv4-Adresse 192.168.100.2 entspricht "C0A86402".

Statische IP-Adressen

Hinweis

Die Seite ist nur verfügbar bei

- SCALANCE W780 / W770 / W740 / W730, W761-1 RJ45 und W722-1 RJ45

Auf dieser Seite legen Sie fest, dass Geräte mit einer bestimmten MAC-Adresse der vorgegebenen IPv4-Adresse zugeordnet werden.

Einstellungen

- **Pool-ID**
  Wählen Sie das gewünschte IPv4-Adressband aus.
- **Identifikationsmethode des Clients**
  Wählen Sie die Methode, nach der ein Client identifiziert wird.
  - Ethernet MAC
    Der Client wird über seine MAC-Adresse identifiziert.
  - Client-ID
    Der Client wird über eine frei definierte DHCP-Client-ID identifiziert.
- **Wert**
  Geben Sie die MAC-Adresse ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.

  Hinweis

  Nur die Pool-ID = 1 wird unterstützt.

- **Identifikationsmethode**
  Zeigt an, ob der Client über seine MAC-Adresse oder die Client-ID identifiziert wird.
- **Wert**
  Zeigt die MAC-Adresse an, dem die IPv4-Adresse zugeordnet wird.
- **IP-Adresse**
  Legen Sie die IPv4-Adresse fest. Die IPv4-Adresse muss zum Subnetz des IPv4-Adressbands passen.

SNMP

Allgemein

Auf dieser Seite treffen Sie grundlegende Einstellungen für SNMP. Aktivieren Sie die Optionen abhängig von der Funktion, die Sie nutzen wollen.

Beschreibung

- **SNMP**
  Wählen Sie das SNMP-Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (Deaktiviert)
    SNMP deaktiviert.
  - SNMPv1/v2c/v3
    SNMPv1/v2c/v3 wird unterstützt.

    Hinweis

    Beachten Sie, dass SNMP in den Versionen 1 und 2c über keine Sicherheitsmechanismen verfügt.

  - SNMPv3
    Nur SNMPv3 wird unterstützt.
- **SNMPv1/v2 schreibgeschützt**
  Wenn Sie diese Option aktivieren, kann SNMPv1/v2c nur lesend auf die SNMP-Variablen zugreifen.

  Hinweis

  Community String

  Verwenden Sie aus Sicherheitsgründen nicht die Standardwerte "public" oder "private". Ändern Sie die Community Strings nach der Erst-Installation.

  Die empfohlene Mindestlänge für Community Strings sind 6 Zeichen.

- **SNMPv1/v2c Read Community String - Lesen**
  Tragen Sie den Community String für den Zugriff des SNMP-Protokolls ein.
- **SNMPv1/v2c Read/Write Community String - Lesen/Schreiben**
  Tragen Sie den Community String für den lesenden und schreibenden Zugriff des SNMP-Protokolls ein.
- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Senden von SNMPv1-Traps (Alarmtelegramme). Im Eintrag "Trap" legen Sie die IP-Adressen der Geräte fest, an die SNMPv1-Traps gesendet werden.
- **SNMPv1/v2c Trap Community String - Trap**
  Tragen Sie den Community String für das Senden von SNMPv1/v2c-Meldungen ein.

- **SNMPv3 Benutzermigration**
  - **Aktiviert**
    Wenn die Funktion aktiviert ist, wird eine SNMP-Engine-ID generiert, die migriert werden kann. Sie können konfigurierte SNMPv3-Benutzer auf ein anderes Gerät übertragen. Wenn Sie diese Funktion aktivieren und die Konfiguration des Geräts auf ein anderes Gerät laden, bleiben konfigurierte SNMPv3-Benutzer erhalten.
  - **Deaktiviert**
    Wenn die Funktion deaktiviert ist, wird eine gerätespezifische SNMP-Engine-ID generiert. Um die ID zu generieren, wird die Agent-MAC-Adresse des Geräts verwendet. Sie können diese SNMP-Benutzerkonfiguration nicht auf andere Geräte übertragen. Wenn Sie die Konfiguration des Geräts auf ein anderes Gerät laden, werden alle konfigurierten SNMPv3-Benutzer gelöscht.
- **SNMP-Engine-ID** (nur Online verfügbar)
  Zeigt die SNMP-Engine-ID an.

Traps

Beim Eintreten eines Alarmereignisses kann ein Gerät SNMP-Traps (Alarmtelegramme) an bis zu zehn verschiedene Management-Stationen gleichzeitig senden. Es werden nur bei solchen Ereignissen Traps gesendet, die unter "Ereignisse" festgelegt sind.

Hinweis

SNMP-Traps werden nur dann versendet, wenn Sie bei "SNMP >Allgemein" oder unter "System > Konfiguration" die Einstellung "SNMPv1 Traps" aktiviert haben.

Beschreibung

- **Trap-Empfängeradresse**
  Tragen Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) der Station ein, an die das Gerät SNMP-Traps sendet. Sie können bis zu zehn verschiedene Empfänger angeben.

Die Tabelle gliedert sich in folgende Spalten:

- **Trap-Empfängeradresse**
  Ändern Sie bei Bedarf die IP-Adressen oder den FQDN (Fully Qualified Domain Name) der Stationen.
- **Trap**
  Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps. Stationen, die eingetragen, aber nicht selektiert sind, erhalten keine SNMP-Traps.

v3-Gruppen

Sicherheitseinstellungen und Rechtevergabe

SNMP Version 3 bietet eine Rechtevergabe, Authentifizierung und Verschlüsselung auf Protokollebene. Das Security-Level und die Lese-/Schreibrechte werden gruppenspezifisch definiert. Für jedes Mitglied einer Gruppe gelten automatisch die entsprechenden Einstellungen.

Beschreibung

- Gruppenname
  Geben Sie den Namen der Gruppe ein. Die maximale Länge beträgt 32 Zeichen.
- Security-Level
  Wählen Sie die Sicherheitsstufe (Authentifizierung, Verschlüsselung) aus, die für die gewählte Gruppe gültig ist. Es gibt folgende Möglichkeiten:
  - Keine Auth./keine Priv.
    Keine Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./keine Priv.
    Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./Priv.
    Authentifizierung aktiviert / Verschlüsselung aktiviert.

Die Tabelle gliedert sich in folgende Spalten:

- Gruppenname
  Zeigt die definierten Gruppennamen an.

  Hinweis

  Der einmal vergebene Gruppenname und die Sicherheitsstufe können nach dem Anlegen nicht mehr geändert werden. Wenn Sie den Gruppennamen oder die Sicherheitsstufe ändern wollen, müssen Sie die Gruppe löschen und mit dem neuen Namen neu anlegen und neu konfigurieren.

- Security-Level
  Zeigt die konfigurierte Sicherheitsstufe an.
- Lesen
  Aktivieren oder deaktivieren Sie den Lesezugriff für die gewünschte Gruppe.

- **Schreiben**
  Aktivieren oder deaktivieren Sie den Schreibzugriff für die gewünschte Gruppe.

  Hinweis

  Damit der Schreibzugriff funktioniert, müssen Sie ebenfalls den Lesezugriff aktivieren.

- **Persistenz**
  Zeigt an, ob die Gruppe einem SNMPv3-Benutzer zugeordnet ist. Wenn die Gruppe keinem SNMPv3-Benutzer zugeordnet ist, wird kein automatisches Speichern ausgelöst und die konfigurierte Gruppe ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Die Gruppe ist einem SNMPv3-Benutzer zugeordnet.
  - Nein
    Die Gruppe ist keinem SNMPv3-Benutzer zugeordnet.

v3-Benutzer

Benutzerspezifische Sicherheitseinstellungen

Auf der Seite können Sie SNMPv3-Benutzer neu anlegen, ändern oder löschen. Das benutzerbasierte Sicherheitsmodell arbeitet mit dem Konzept des Benutzernamens, d. h. jedes Telegramm wird mit einer Benutzerkennung versehen. Diesen Benutzernamen und die betreffenden Sicherheitseinstellungen überprüfen sowohl der Absender wie auch der Empfänger.

Beschreibung

- **Benutzername**
  Geben Sie einen frei wählbaren Benutzernamen ein. Nach der Datenübernahme können Sie den Namen nicht mehr ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Benutzername**
  Zeigt die angelegten Benutzer an.
- **Gruppenname**
  Wählen Sie die Gruppe aus, die dem Benutzer zugeordnet wird.
- **Authentifizierungsprotokoll**
  Legen Sie das Authentifizierungsprotokoll fest, für das ein Passwort hinterlegt werden soll. Folgende Einstellungen gibt es:
  - Keine
  - MD5
  - SHA
- **Verschlüsselungsprotokoll**
  Legen Sie fest, ob ein Passwort zur Verschlüsselung mit dem DES-Algorithmus hinterlegt werden soll. Nur aktivierbar, wenn auch ein Authentifizierungsprotokoll ausgewählt wurde.

- **Authentifizierungspasswort**
  Geben Sie das Authentifizierungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Authentifizierungspasswort bestätigen**
  Bestätigen Sie das Passwort durch die Wiederholung der Eingabe.
- **Verschlüsselungspasswort**
  Geben Sie Ihr Verschlüsselungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Verschlüsselungspasswort bestätigen**
  Bestätigen Sie das Verschlüsselungspasswort durch die Wiederholung der Eingabe.
- **Persistenz**
  Zeigt an, ob der Benutzer einer SNMPv3-Gruppe zugeordnet ist. Wenn der Benutzer keiner SNMPv3-Gruppe zugeordnet ist, wird kein automatisches Speichern ausgelöst und der konfigurierte Benutzer ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Der Benutzer ist einer SNMPv3-Gruppe zugeordnet.
  - Nein
    Der Benutzer ist keiner SNMPv3-Gruppe zugeordnet.

Systemzeit

Manuelle Zeiteinstellung

Manuelle Einstellung der Systemzeit

Auf dieser Seite stellen Sie das Datum und die Uhrzeit des Systems ein. Damit diese Einstellung verwendet wird, müssen Sie "Manuelle Zeiteinstellung" aktivieren.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Manuelle Zeiteinstellung**
  Aktivieren oder deaktivieren Sie die manuelle Zeiteinstellung. Wenn Sie die Option aktivieren, wird das Eingabefeld "Systemzeit" editierbar.
- **Systemzeit**
  Geben Sie Datum und Uhrzeit im Format "MM/DD/YYYY HH:MM:SS" ein.
- **PC-Zeit verwenden**
  Klicken Sie auf die Schaltfläche, um die Zeiteinstellung des PCs zu übernehmen.
- **Letzter Synchronisationszeitpunkt**
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
  Wenn keine Uhrzeitsynchronisation möglich war, enthält das Feld die Angabe "Datum/Zeit nicht eingestellt".
- **Letzter Synchronisationsmechanismus**
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde.
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm.
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. Die aktuelle Systemzeit sehen Sie oben rechts im Auswahlbereich des WBM. In dem Feld "Systemzeit" wird weiterhin die eingestellte Zeit angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.

DST-Übersicht

Umstellung der Sommerzeit

Auf dieser Seite können Sie neue Einträge für die Umstellung der Sommerzeit anlegen. Die Tabelle gibt Ihnen einen Überblick über die vorhandenen Einträge.

Beschreibung

- **DST-Nr.**
  Zeigt die Nummer des Eintrags an.
  Wenn Sie einen neuen Eintrag anlegen, wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Name**
  Zeigt den Namen des Eintrags an.
- **Jahr**
  Zeigt das Jahr an, für das der Eintrag angelegt wurde.
- **Anfangsdatum**
  Zeigt Monat, Tag und Uhrzeit für den Start der Sommerzeit an.
- **Enddatum**
  Zeigt Monat, Tag und Uhrzeit für das Ende der Sommerzeit an.
- **Regelmäßige Zeitpunkte der Umstellung**
  Bei einem Eintrag des Typs "Regel" wird die Zeitspanne angezeigt, bestehend aus Woche, Tag, Monat und Uhrzeit, in der die Sommerzeit aktiv ist. Bei einem Eintrag des Typs "Datum" wird ein "-" angezeigt.
- **Status**
  Zeigt der Status des Eintrags an:
  - Aktiviert
    Der Eintrag wurde korrekt angelegt.
  - Ungültig
    Der Eintrag wurde neu angelegt und Anfangs- und Enddatum sind identisch.
- **Typ**
  Zeigt an, wie die Umstellung der Sommerzeit erfolgt:
  - Datum
    Es ist ein festes Datum für die Umstellung der Sommerzeit eingetragen.
  - Regel
    Es ist eine Regel für die Umstellung der Sommerzeit definiert.

DST-Konfiguration

Umstellung der Sommerzeit konfigurieren

Auf dieser Seite können Sie die Einträge für die Umstellung der Sommerzeit konfigurieren. Durch die Umstellung auf Sommer- bzw. Winterzeit ist die Systemzeit für die lokale Zeitzone korrekt eingestellt. Sie können eine Regel für die Umstellung der Sommerzeit definieren oder ein festes Datum angeben.

Beschreibung

Hinweis

Der Inhalt dieser Seite ist abhängig davon, was Sie im Feld "Typ" auswählen.

Die Felder "DST-Nr.", "Typ" und "Name" werden immer angezeigt.

- **DST-Nr.**
  Wählen Sie die Nummer des Eintrags aus.
- **Typ**
  Wählen Sie aus, wie die Umstellung der Sommerzeit erfolgen soll:
  - Datum
    Sie können ein festes Datum für die Umstellung der Sommerzeit angeben. Diese Einstellung eignet sich für Regionen, in denen die Umstellung der Sommerzeit keiner Regel folgt.
  - Periodisch
    Sie können eine Regel für die Umstellung der Sommerzeit definieren. Diese Einstellung eignet sich für Regionen, in denen die Sommerzeit immer an einem bestimmten Wochentag beginnt bzw. endet.
- **Name**
  Geben Sie einen Namen für den Eintrag an. Der Name kann maximal 16 Zeichen lang sein.

Einstellungen bei der Auswahl "Datum"

Sie können ein festes Datum für den Beginn und das Ende der Sommerzeit angeben.

- **Jahr**
  Geben Sie das Jahr für die Umstellung der Sommerzeit an.
- **Anfangsdatum**
  Geben Sie folgende Werte für den Beginn der Sommerzeit an:
  - Tag
    Geben Sie den Tag an.
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
- **Enddatum**
  Geben Sie folgende Werte für das Ende der Sommerzeit an:
  - Tag
    Geben Sie den Tag an.
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.

Einstellungen bei der Auswahl "Regel"

Sie können eine Regel für die Umstellung der Sommerzeit erstellen.

- **Anfangsdatum**
  Geben Sie folgende Werte für den Beginn der Sommerzeit an:
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
  - Woche
    Geben Sie die Woche an.
    Sie können die erste bis vierte oder die letzte Woche des Monats auswählen.
  - Wochentag
    Geben Sie den Wochentag an.
- **Enddatum**
  Geben Sie folgende Werte für das Ende der Sommerzeit an:
  - Stunde
    Geben Sie die Stunde an.
  - Monat
    Geben Sie den Monat an.
  - Woche
    Geben Sie die Woche an.
    Sie können die 1. bis 4. oder die letzte Woche des Monats auswählen.
  - Wochentag
    Geben Sie den Wochentag an.

SNTP-Client

Uhrzeitsynchronisation im Netzwerk

Das SNTP (Simple Network Time Protocol) dient zur Zeitsynchronisation im Netzwerk. Die entsprechenden Telegramme werden von einem SNTP-Server im Netz versendet.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SNTP-Client**
  Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über SNTP.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom IE-Switch empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.

- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit. Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
    In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.
- **SNTP-Modus**
  Wählen Sie die Synchronisationsart aus. Folgende Synchronisierungsarten sind möglich:
  - Listen
    Bei diesem Modus ist das Gerät passiv und empfängt SNTP-Telegramme, die die Uhrzeit liefern. Einstellungen in den Eingabefeldern "SNTP-Server-Adresse" und "Port des SNTP-Servers" haben in diesem Modus keine Wirkung. In diesem Modus werden nur IPv4-Adressen unterstützt.
  - Poll
    Wenn Sie diesen Modus wählen, wird das Eingabefeld "Poll-Intervall[s]" zur weiteren Konfiguration eingeblendet. In diesem Modus werden die Einstellungen in den Eingabefeldern "SNTP-Server-Adresse" und "Port des SNTP-Servers" berücksichtigt. Bei dieser Synchronisationsart ist das Gerät aktiv und sendet eine Zeitabfrage an den SNTP-Server. In diesem Modus werden IPv4- und IPv6-Adressen unterstützt.
- **Poll-Intervall[s]**
  Geben Sie den Zeitabstand zwischen zwei Zeitanfragen ein. In diesem Feld geben Sie das Abfrageintervall in Sekunden an. Mögliche Werte sind 16 bis 16284 Sekunden.

- **SNTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SNTP-Servers ein.
- **Port des SNTP-Servers**
  Geben Sie den Port des SNTP-Servers ein.
  Folgende Ports sind möglich:
  – 123 (Standard-Port)
  – 1025 bis 36564
- **Primär**
  Bei dem SNTP-Server, den SIe zuerst anlegen, wird das Häckchen gesetzt. Wenn mehrere SNTP-Server angelegt sind, wird der primäre Server zuerst angefragt.

NTP-Client

Automatische Zeiteinstellung über NTP

Wenn die Uhrzeitsynchronisation über NTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **NTP-Client**
  Markieren Sie dieses Optionskästchen, um die automatische Zeitsynchronisation über NTP zu aktivieren.
- **Nur NTP (secure)-Client**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem gesicherten NTP-Server. Die Einstellung gilt für alle Severeinträge.
  Um den gesicherten NTP-Client zu aktivieren, konfigurieren Sie die Parameter für die Authentifizierung (Schlüssel-ID, Hash-Algorithmus, Schlüssel).
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom IE-Switch empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.

- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. In dem Feld "Aktuelle Systemzeit" wird weiterhin die Normalzeit inkl. Zeitzone angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.
- **NTP-Serverindex**
  Wählen Sie den Index des NTP-Servers aus. Der Server mit dem kleinsten Index wird zuerst angefragt.
- **NTP-Server-Adresse**
  Geben Sie die IPv4-Adresse des NTP-Servers an.
- **Port des NTP-Servers**
  Geben Sie den Port des NTP-Servers an.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564
- **Poll-Intervall[s]**
  Legen Sie den Zeitabstand zwischen zwei Uhrzeitanfragen fest. Mögliche Werte sind 64 bis 1024 Sekunden.
- **Schlüssel-ID**
  Geben Sie die ID des Authentifizierungsschlüssels ein.

- **Hash-Algorithmus**
  Legen Sie das Format für den Authentifizierungsschlüssel fest.
- **Schlüssel**
  Geben Sie den Authentifizierungsschlüssel ein.
- **Schlüssel bestätigen**
  Geben Sie nochmals den Authentifizierungsschlüssel ein, um ihn zu bestätigen.

SIMATIC Time Client

Zeiteinstellung über SIMATIC Time Client

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den SIMATIC Time Client.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SIMATIC Time Client**
  Markieren Sie dieses Optionskästchen, um das Gerät als SIMATIC Time Client zu aktivieren.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemzeit an.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
  - PTP
    Automatische Zeitsynchronisation über PTP

Auto-Logout

Stellen Sie in dieser Seite die Zeiten ein, nach denen bei Inaktivität des Benutzers automatisch eine Abmeldung vom WBM oder dem CLI erfolgt.

Wenn Sie automatisch abgemeldet wurden, dann müssen Sie sich wieder neu anmelden.

Hinweis

Keine automatische Abmeldung vom CLI

Wenn die Verbindung nach der eingestellten Zeit nicht beendet wird, prüfen Sie am Telnet Client die Einstellung des "Keep alive"- Mechanismus.

Ist die Intervallzeit kleiner als die projektierte Zeit, wird die die Verbindung aufrecht erhalten, obwohl keine Nutzdaten übertragen werden. Z. B. Sie haben bei der automatischen Abmeldung 300 Sekunden eingestellt und bei der "Keep alive"- Funktion steht 120 Sekunden. In diesem Fall wird alle 120 Sekunden ein Paket gesendet, dass die Verbindung aufrecht erhält.

- Schalten Sie den Kepp-Alive-Mechanismus aus (Intervallzeit=0)
  oder
- Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung bei Inaktivität beendet wird.

Beschreibung

- **Web Based Management [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom WBM ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.
- **CLI (TELNET, SSH) [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom CLI ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.

Syslog-Client

Systemereignis-Agent

Syslog nach RFC 3164 wird für die Übermittlung von kurzen, unverschlüsselten Textmeldungen per UDP im IP-Netz verwendet. Dazu wird ein Syslog-Server benötigt.

Voraussetzungen für das Versenden der Log-Einträgen

- Die Syslog-Funktion ist im Gerät aktiviert.
- Die Syslog-Funktion für das jeweilige Ereignis ist aktiviert.
- In Ihrem Netz befindet sich ein Syslog-Server, der die Log-Einträge entgegen nimmt. Da es sich um eine UDP-Verbindung handelt, gibt es keine Rückmeldung an den Absender.
- Die IP-Adresse oder der FQDN (Fully Qualified Domain Name) des Syslog-Servers ist im Gerät eingetragen.

Beschreibung

Die Seite enthält folgende Felder:

- **Syslog-Client**
  Aktivieren oder deaktivieren Sie die Syslog-Funktion.
- **Adresse des Syslog-Servers**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Adresse des Syslog-Servers**
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers an.
- **Server-Port**
  Geben Sie den verwendeten Port des Syslog-Servers ein.

Fehlerkontrolle

Spannungsversorgung

Hinweis

Diese Seite ist auf dem SCALANCE W786-2 SFP nicht verfügbar.

Konfigurieren Sie, ob die Spannungsversorgung durch das Meldesystem überwacht werden soll. Je nach Hardware-Variante gibt es ein oder zwei Spannungsanschlüsse (Versorgung 1 / Versorgung 2). Bei redundanter Spannungsversorgung konfigurieren Sie die Überwachung für jede einzelne Zuleitung getrennt.

Wenn an einem überwachten Anschluss (Versorgung1 oder Versorgung 2) keine oder eine zu geringe Spannung anliegt, wird ein Fehler durch das Meldesystem signalisiert.

Hinweis

Die zulässigen Betriebsspannungsgrenzen entnehmen Sie der Betriebsanleitung des Geräts.

Ein Fehler führt zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Einstellungen

- **Versorgung 1**
  Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 1.
- **Versorgung 2** (nicht verfügbar bei W760/W720)
  Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 2.

- **PoE** (nicht verfügbar bei W760/W720)
  Aktivieren oder deaktivieren Sie die Überwachung der Stromversorgung über Ethernet.
- **PROFINET-Redundanz** (nicht verfügbar bei W760/W720
  Bei den folgenden Geräten können Sie zusätzlich konfigurieren, welche Spannungsversorgung durch PROFINET überwacht werden soll:
  - SCALANCE W788-x RJ45-Varianten
  - SCALANCE W748-1 RJ45
  - SCALANCE W774-1 (RJ45- und M12-Variante)
  - SCALANCE W734-1 RJ45

Link Change

Konfiguration der Fehlerüberwachung von Zustandsänderungen bei Verbindungen

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

Bei aktivierter Verbindungsüberwachung wird ein Fehler signalisiert,

- wenn an einem Port ein Link vorhanden sein soll und dieser fehlt.
- oder wenn an dem Port kein Link vorhanden sein soll und ein Link erkannt wird.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Möglichkeiten:
  - "-" (Deaktiviert)
  - Up
  - Down
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Port**
  Zeigt die verfügbaren Ports an.
- **Einstellung**
  Wählen Sie die Einstellung aus. Sie haben folgende Möglichkeiten:
  - Up
    Die Fehlerbehandlung wird beim Übergang in den aktiven Zustand des Ports ausgelöst.
    (Von "Link down" nach "Link up")
  - Down
    Die Fehlerbehandlung wird beim Übergang in den inaktiven Zustand des Ports ausgelöst.
    (Von "Link up" nach "Link down")
  - "-" (Deaktiviert)
    Die Fehlerbehandlung wird nicht ausgelöst.

EtherNet/IP

EtherNet/IP

Auf dieser Seite konfigurieren Sie den Modus von EtherNet/IP.

Hinweis

Geräte mit zwei Ethernet-Schnittstellen

Bei Geräten mit zwei Ethernet-Schnittstellen darf nur eine der Schnittstellen (P1 oder P2) für die EtherNet/IP-Projektierung verwendet werden.

Dies betrifft folgende Geräte:

- SCALANCE W786-2 SFP .
- SCALANCE W774-1 RJ45
- SCALANCE W774-1 M12 EEC
- SCALANCE W734-1 RJ45.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **EtherNet/IP-Gerätediagnose**
  Zeigt an, ob EtherNet/IP aktiviert ("On") oder deaktiviert ("Off") ist.
- **EtherNet/IP-Gerätediagnose beim nächsten Hochlauf**
  Stellen Sie ein, ob EtherNet/IP nach dem nächsten Neustart des Geräts aktiviert ("On") oder deaktiviert ("Off") sein soll.

Hinweis

EtherNet/IP und PROFINET

Wenn EtherNet/IP eingeschaltet wird, wird PROFINET ausgeschaltet. Das Umschalten von EtherNet/IP und PROFINET hat keine Auswirkungen auf DCP.

Hinweis

PROFINET AR Status

Wenn eine PROFINET-Verbindung aufgebaut ist, d. h. der PROFINET AR-Status "Online" ist, können Sie EtherNet/IP nicht aktivieren.

PROFINET

Einstellungen für PROFINET

Diese Seite zeigt den PROFINET AR Status und den Gerätenamen an.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **PROFINET-Modus**
  Zeigt an, ob PROFINET aktiviert ("On") oder deaktiviert ("Off") ist.
- **PROFINET-Modus beim nächsten Hochlauf**
  Stellen Sie ein, ob PROFINET nach dem nächsten Neustart des Geräts aktiviert ("On") oder deaktiviert ("Off") sein soll.

Hinweis

PROFINET und EtherNet/IP

Wenn PROFINET eingeschaltet wird, wird EtherNet/IP ausgeschaltet. Das Umschalten von PROFINET und EtherNet/IP hat keine Auswirkungen auf DCP.

Hinweis

PROFINET AR Status

Wenn eine PROFINET-Verbindung aufgebaut ist, d. h. der PROFINET AR-Status "Online" ist, können Sie PROFINET nicht deaktivieren.

- **PROFINET AR-Status**
  Dieses Feld zeigt den Status des PROFINET-Verbindungsverhältnisses an, d.h. ob das Gerät mit einem PROFINET-Controller "Online" oder "Offline" verbunden ist.
  Online bedeutet hierbei, dass eine Verbindung zu einem PROFINET-Controller besteht, dass dieser seine Konfigurationsdaten auf das Gerät geladen hat und das Gerät Statusdaten zum PROFINET-Controller senden kann. In diesem Zustand, der auch "in Data exchange" genannt wird, sind die Parameter, die über den PROFINET-Controller eingestellt werden, nicht konfigurierbar.
- **PROFINET-Gerätename**
  In diesem Feld erscheint der PROFINET-Gerätenamen gemäß der Projektierung in der HW-Config von STEP 7.

Hinweis

Geräte mit zwei Ethernet-Schnittstellen

Bei Geräten mit zwei Ethernet-Schnittstellen sollte nur die Schnittstelle P1 für die PROFINET-Projektierung verwendet werden, weil LLPD-Frames nur über die Schnittstelle P 1 gesendet und empfangen werden. An der Schnittstelle P 2 werden sie geblockt und sie werden auch nicht zwischen den Schnittstellen weitergeleitet.

Dies betrifft folgende Geräte:

- SCALANCE W786-2 SFP
- SCALANCE W774-1 RJ45
- SCALANCE W774-1 M12 EEC
- SCALANCE W734-1 RJ45

SCALANCE W700 und STEP7

Die Ethernet-Schnittstelle kann von STEP7 aus projektiert werden, wenn folgende Voraussetzungen erfüllt sind:

- STEP7 V13 Update 3 mit HSP0107 oder
- STEP7 Version 5.5.4 mit GSDML Version 2.31

Darüber hinaus können die Diagnose-Funktionen genutzt werden. Für die WLAN-Schnittstelle ist eine Konfiguration mit STEP7 nicht möglich.

PROFINET für Client-Geräte

Wenn ein Client als PROFINET-Gerät eingesetzt werden soll, muss die MAC-Adresse des Clients wie folgt festgelegt werden (MAC Mode):

- **Eigene**
  Im dahinterliegenden Netz ist nur IP-Kommunikation und kein PROFINET möglich.
- **Layer-2-Tunnel**
  Der Client und die dahinterliegenden Geräte können als PROFINET-Geräte betrieben werden.

Hinweis

Wenn für einen Client "Automatisch" bzw. "Manuell" als MAC-Modus konfiguriert ist, kann dieses Gerät nicht als PROFINET-Gerät eingesetzt werden.

PLUG

Konfiguration

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite ist nur bei SCALANCE W780 / W 770 / W740 / W730 verfügbar.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.

Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt. Bei SCALANCE W werden in diesem Fall die verfügbaren Funkschnittstellen deaktiviert.

Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurückgesetzt.

Informationen über die Konfiguration des C-PLUG / KEY-PLUG

Diese Seite liefert Detailinformationen über die Konfiguration, die im C-PLUG bzw. KEY-PLUG abgelegt ist. Darüber hinaus gibt es die Möglichkeit, den PLUG auf Werkseinstellungen zurückzusetzen oder mit einem neuen Inhalt zu versehen.

Hinweis

Die Aktion wird erst dann durchgeführt, wenn Sie auf die Schaltfläche "Einstellungen übernehmen" klicken.

Die Aktion kann nicht rückgängig gemacht werden.

Wenn Sie sich nach der Auswahl gegen die Ausführung entscheiden, dann klicken Sie auf die Schaltfläche "Aktualisieren". Dadurch werden die Daten dieser Seite aus dem Gerät neu ausgelesen und Ihre Auswahl wird aufgehoben.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da die sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Einstellungen

Die Tabelle gliedert sich in folgende Zeilen:

- **Status**
  Zeigt den Status des PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Es ist ein PLUG mit einer gültigen und passenden Konfiguration im Gerät vorhanden.
  - NOT ACCEPTED
    Ungültige bzw. inkompatible Konfiguration auf dem gesteckten PLUG.
  - NOT PRESENT
    Im Gerät ist kein C-PLUG oder KEY-PLUG gesteckt.
  - FACTORY
    PLUG ist gesteckt und enthält keine Konfiguration. Dieser Status wird auch angezeigt, wenn der PLUG im Betrieb formatiert wurde.
  - MISSING
    Es ist kein PLUG gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
- **Gerätegruppe**
  Zeigt an, von welcher SIMATIC NET-Produktlinie der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Gerätetyp**
  Zeigt den Gerätetyp innerhalb der Produktlinie an, von dem der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Version der Konfiguration**
  Die Version der Konfigurationsstruktur. Diese Angabe betrifft die vom Gerät unterstützten Konfigurationsmöglichkeiten und hat nichts mit der konkreten Hardware-Konfiguration zu tun. Diese Revisionsangabe ändert sich also nicht, wenn Sie Zusatzkomponenten (z.B. Module bzw. Extender) hinzufügen oder entfernen, sie kann sich aber ändern, wenn Sie ein Firmware-Update durchführen.
- **Dateisystem**
  Zeigt den Typ des Dateisystems an, das auf dem PLUG vorhanden ist.

ACHTUNG
Neues Dateisystem UBI Ab Firmware-Version 2.0 ist UBI das Standard-Dateisystem für den C-PLUG bzw. KEY-PLUG. Wird in einem solchen Gerät ein C-PLUG mit dem bisherigen Dateisystem IECP erkannt, wird dieser C-PLUG für das Dateisystem UBI formatiert und die Daten werden neu auf den C-PLUG geschrieben. Die Änderung des Dateisystems erfolgt auch nach einem Firmware-Update auf V2.0. Ein Downgrade auf die Vorgängerversion der entsprechenden Firmware ist dann problematisch. Die Firmware kann den C-PLUG bzw. KEY-PLUG weder lesen noch schreiben und auch ein "PLUG auf Werkseinstellungen zurücksetzen" ist nicht möglich. .

- **Verfügbarer Speicherplatz [Byte]**
  Zeigt die maximale Speicherkapazität des Dateisystems an, das auf dem PLUG vorhanden ist.

- **Belegter Speicherplatz [Byte]**
  Zeigt den belegten Speicherplatz im Dateisystem des PLUG an.
- **Firmware auf PLUG**
  Wenn aktiviert, wird die Firmware auf dem PLUG abgespeichert. Damit können mit dem PLUG automatische Firmware-Updates/Downgrades durchgeführt werden.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.
  Wenn ein PLUG als PRESET-PLUG konfiguriert wurde, wird dies hier als Zusatzinformation in der ersten Zeile angezeigt. Nähere Informationen zur Erstellung und Benutzung eines PRESET-PLUG finden Sie in Kapitel "Instandhalten und Warten (Seite 363)".
- **PLUG ändern**
  Wählen Sie die gewünschte Einstellung. Sie haben folgende Möglichkeiten, um die Konfiguration auf dem C-PLUG bzw. KEY-PLUG zu ändern:
  - Aktuelle Konfiguration auf den PLUG schreiben
    Diese Option ist nur verfügbar, wenn der Status des PLUG "NOT ACCEPTED" oder "FACTORY" ist.
    Die im internen Flash-Speicher des Gerätes vorhandene Konfiguration wird auf den PLUG kopiert.
  - PLUG auf Werkseinstellungen zurücksetzen
    Löscht alle Daten vom PLUG und führt eine Low-Level-Formatierung durch.

Lizenz

ACHTUNG
C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken! Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden. Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt. Bei SCALANCE W werden in diesem Fall die verfügbaren Funkschnittstellen deaktiviert. Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurück.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Informationen über die Lizenz des KEY-PLUG

Ein C-PLUG kann nur die Konfiguration eines Geräts speichern. Ein KEY-PLUG enthält zusätzlich zur Konfiguration eine Lizenz, die bestimmte Funktionen Ihres SIMATIC NET-Geräts freischaltet.

Hinweis

Informationen nur Online verfügbar

Die Informationen auf dieser Seite können sie nur einsehen, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite ist nur bei SCALANCE W780 / W 770 / W740 / W730 verfügbar.

Angezeigte Werte

- **Status**
  Zeigt den Status des KEY-PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Der im Gerät vorhandene KEY-PLUG enthält eine passende und gültigen Lizenz.
  - NOT ACCEPTED
    Die Lizenz des gesteckten KEY-PLUG ist nicht gültig.
  - NOT PRESENT
    Im Gerät ist kein KEY-PLUG gesteckt.
  - MISSING
    Es ist kein KEY-PLUG oder ein C-PLUG mit dem Status "FACTORY" gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
  - WRONG
    Der gesteckte KEY-PLUG passt nicht zum Gerät.
  - UNKNOWN
    Unbekannter Inhalt des KEY-PLUG.
  - DEFECTIVE
    Der Inhalt des KEY-PLUG ist fehlerhaft.
- **Artikelnummer**
  Zeigt die Artikelnummer des KEY-PLUG an. Es gibt den KEY-PLUG für unterschiedliche Funktionserweiterungen und für verschiedene Zielsysteme.
- **Seriennummer**
  Zeigt die Seriennummer des KEY-PLUG.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den KEY-PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.

Hinweis

Beim Speichern der Konfiguration wird die Information mitgespeichert, ob zu diesem Zeitpunkt ein KEY-PLUG im Gerät gesteckt war. Diese Konfiguration ist dann auch nur lauffähig, wenn ein KEY-PLUG mit der gleichen Bestellnummer / Lizenz gesteckt ist. Das ist unabhängig davon, ob z. B. iFeatures konfiguriert sind.

Ping

Erreichbarkeit einer Adresse in einem IP-Netzwerk

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mit der Ping-Funktion können Sie überprüfen, ob eine bestimmte Adresse im Netzwerk erreichbar ist.

Beschreibung

Die Seite enthält folgende Felder:

- **Zieladresse**
  Geben Sie die IPv4-, IPv6-Adresse oder den FQDN (Fully Qualified Domain Name) des Geräts ein.
- **Wiederholen**
  Tragen Sie die Anzahl der Ping-Anforderungen ein.
- **DNS-Auflösung**
  Wählen Sie aus in welchen IP-Adress-Typ ein eingegebener FQDN aufgelöst werden soll.
  - Auto
    In diesem Modus wird der IP-Adress-Typ automatisch gewählt.
  - IPv4
    Der eingegebene FQDN wird in einer IPv4-Adresse aufgelöst.
  - IPv6
    Der eingegebene FQDN wird in einer IPv6-Adresse aufgelöst.
- **Ausgehende Schnittstelle für IPv6**
  Diese Auswahl wird nur benötigt, wenn die Zieladresse eine Multicast- oder eine Link-lokale Adresse ist.
  - "-" (Werkseinstellung)
  - Wählen Sie die entsprechende IPv6-Schnittstelle aus.
- **Ping**
  Klicken Sie diese Schaltfläche, um die Ping-Funktion zu starten.
- **Ping-Ausgabe**
  Dieses Feld zeigt die Ausgabe der Ping-Funktion an.
- **Leeren**
  Klicken Sie diese Schaltfläche, um die Ping-Ausgabe zu leeren.

Schnittstellen projektieren

Ethernet

Übersicht

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an.

Beschreibung

- **Port**
  Zeigt die konfigurierbaren Ports an.
- **Port-Name**
  Zeigt den Namen des Ports.
- **Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Betriebszustand** (Nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.
- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsparameter des Ports an
- **MTU**
  Zeigt die maximale Paketgröße an.
- **Negotiation**
  Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.

Konfiguration

Auf dieser Seite konfigurieren Sie die Ethernet-Ports des Geräts.

Hinweis

SCALANCE W786-2 SFP

Die beiden SFP-Ports des SCALANCE W786-2 SFP sind nicht einzeln parametrierbar und diagnostizierbar.

Einstellungen

- **Port**
  Wählen Sie die zu konfigurierenden Port aus. .
- **Status**
  Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled
    Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled
    Der Port ist ausgeschaltet, aber die Verbindung besteht noch.
  - Link down
    Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**
  Tragen Sie hier einen Namen für den Port ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Übertragungsmodus**

  Hinweis

  Der Parameter ist nicht konfigurierbar bei:
  - SCALANCE W786-2 SFP
  - SCALANCE W770
  - SCALANCE W760
  - SCALANCE W730
  - SCALANCE W720

  Der Wert ist fest auf "Auto negotiation" eingestellt.

  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Die Einstellungen für "Autonegotiation" und "Übertragungsrate" legen Sie in den Port-Optionen fest.

  Hinweis

  Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt an, welche Übertragungsgeschwindigkeit und welches Übertragungsverfahren aktuell verwendet werden.
- **Negotiation**
  Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.
- MTU ((Maximial Transmission Unit)
  Tragen Sie die Paketgröße ein, ab dem Pakete fragmentiert werden.
- **Betriebszustand**
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - Up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - Down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.

Veränderung der Port-Konfiguration

Hinweis

Optische Ports arbeiten immer mit dem Übertragungsverfahren Vollduplex und mit maximaler Übertragungsgeschwindigkeit. Deshalb können Sie bei optischen Ports folgende Einstellungen nicht vornehmen:

- Automatische Konfiguration
- Übertragungsgeschwindigkeit
- Übertragungsverfahren

Hinweis

Das Gerät verhindert oder reduziert bei Überlastung eines Ports durch verschiedene Automatismen die Rückwirkung auf andere Ports und Prioritätsklassen (Class of Service). Dies kann auch bei aktivierter Flusskontrolle dazu führen, dass Telegramme verworfen werden.

Port-Überlastungen treten auf, wenn das Gerät mehr Telegramme empfängt, als es senden kann, z.B. infolge unterschiedlicher Übertragungsgeschwindigkeiten.

WLAN

Basic

Hinweis

Um die WLAN-Schnittstelle zu konfigurieren, müssen Sie immer zuerst den Ländercode festlegen. Einige Parameter sind abhängig von der Ländereinstellung, z. B. der Übertragungsstandard.

Einstellungen

- **Ländercode**
  Wählen Sie das Land aus, in dem das Gerät eingesetzt wird.
  Sie brauchen die landesspezifischen Daten nicht zu kennen, die richtige Kanaleinteilung und Festlegung der Sendeleistung erfolgt entsprechend Ihrer Länderauswahl durch das Gerät.

  Hinweis
  Ländereinstellung

  Die richtige Ländereinstellung ist für einen zulassungskonformen Betrieb unbedingt notwendig. Die Auswahl eines vom Anwenderland abweichenden Landes kann strafrechtlich geahndet werden.

- **Gerätemodus**
  Betriebsart des Geräts. Die Betriebsart ist nur nur bei Access Points änderbar.
  Folgende Betriebsarten sind möglich:
  - AP: Access Point-Modus
  - Client: Client-Modus

  Hinweis

  Nach dem Ändern der Betriebsart, wird eine Meldung eingeblendet. Wenn Sie die Meldung mit "Ja" bestätigen, werden die Konfigurationseinstellungen in der geänderten Betriebsart aktualisiert.

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.

- **Aktiviert**
  Status der WLAN-Schnittstelle. Um die WLAN-Schnittstelle zu aktivieren, aktivieren Sie die Einstellung.

  Hinweis

  WLAN-Schnittstelle aktivieren

  Die WLAN-Schnittstellen sind im Auslieferungszustand deaktiviert. Die WLAN-Schnittstellen sind aktivierbar, nachdem die Länder- und die Antenneneinstellungen konfiguriert sind.

- **Modus der Funkschnittstelle**
  Zeigt die Betriebsart der WLAN-Schnittstelle an.
- **Frequenzband**
  Legen Sie das Frequenzband fest. Im Client-Modus ist auch ein Dual-Frequenzbetrieb möglich.
  - 2.4 GHz
  - 5 GHz
  - 2.4 GHz + 5 GHz (nur im Client-Modus)

  Hinweis

  WLAN-Schnittstellen des W786-2IA RJ45 für unterschiedliche Frequenzbänder konfigurieren

  Wenn bei diesem Gerät beide WLAN-Schnittstellen für das gleiche Frequenzband konfiguriert sind, kann es zu gegenseitiger Beeinflussung oder Störungen kommen. Dies gilt besonders bei hohem Datendurchsatz.

- **WLAN Modus 2.4 GHz/WLAN Modus 5 GHz**
  Wählen Sie für das projektierte Frequenzband den gewünschten Übertragungsstandard. Die Auswahl ist abhängig von der Ländereinstellung.
  - Auto (nur im Client-Modus)
    Der Übertragungsstandard wird automatisch ermittelt (2.4 GHz + 5 GHz).
  - 802.11a
    Der Übertragungsstandard IEEE 802.11a (5 GHz) ist eingestellt.
  - 802.11g
    Der Übertragungsstandard IEEE 802.11g (2.4 GHz) ist eingestellt. Dieser Übertragungsstandard ist abwärts kompatibel zu IEEE 802.11b.
  - 802.11n
    Der Übertragungsstandard IEEE 802.11n (2.4 GHz und 5 GHz) ist eingestellt. Dieser Übertragungsstandard ist abwärts kompatibel zu IEEE802.11a und IEEE 802.11g.
  - nur 802.11n
    Der Übertragungsstandard IEEE 802.11n (2.4 GHz und 5 GHz) ist eingestellt. Dieser Übertragungsstandard ist abwärts kompatibel zu IEEE802.11a und IEEE 802.11g.

Hinweis

Fragmentierungslänge

Wenn Sie den Übertragungsstandard "802.11n", "nur 802.11n" oder "Auto" (nur im Client-Modus) auswählen, können Sie den Schwellenwert der Fragmentierungslänge nicht einstellen, siehe "Schwellenwert der Fragmentierungslänge" unter "Schnittstellen > WLAN > Erweiterungen".

- DFS (802.11h)
  Aktiviert oder deaktiviert die Funktion "Dynamic Frequency Selection (DFS)".

  Hinweis

  RCoax 5 GHz und DFS

  In den USA und in Ländern, die der FCC (Federal Communication Commission) folgen, darf beim Betrieb mit DFS (Dynamic FrequenceSelection) das IWLAN RCoax Cable 5GHz nicht verwendet werden. Der aktuelle Stand der Zulassungen ist im Internet unter der Adresse (http://www.siemens.com/funkzulassungen) abrufbar.

  – Aktiviert
    Wenn Sie diese Funktion aktiviert haben, stehen Ihnen zusätzliche DFS-Kanäle aus dem 5 GHz-Band zur Verfügung. Diese Kanäle sind länderspezifisch und unterliegen bestimmten DFS-Vorgaben.
    Wenn der Access Point auf dem aktuellen Kanal eine Störung entdeckt, wechselt er automatisch auf einen Alternativ-Kanal und der aktuelle Kanal wird für 30 Minuten gesperrt. Die Störung kann z. B. von einem Primärbenutzer oder einer Radarinterferenz stammen.
    Bevor der Access Point die Kommunikation auf einem Kanal startet, sucht er 60 Sekunden nach Primärnutzern auf dem Kanal. In dieser Zeit sendet der Access Point keine Beacons. Wenn Signale auf dem Kanal entdeckt werden, wechselt der Access Point den Kanal und wiederholt die Prüfung. Erst wenn nach Ablauf der 60 Sekunden keine Signale von Primärnutzern erkannt wurden, sendet der Access Point auf einem Kanal. Auch während des Betriebs sucht der Access Point nach Primärnutzern.

  – Deaktiviert
    Die Funktion DFS wird nicht verwendet.

- Outdoor-Modus

  – Aktiviert
    Wenn Sie den Outdoor-Modus aktiviert haben, stehen Ihnen nur die Kanäle zur Verfügung, die für den Outdoor-Betrieb zugelassen sind.

  – Deaktiviert
    Wenn Sie den Outdoor-Modus deaktiviert haben, stehen Ihnen nur die Kanäle zur Verfügung, die für den Betrieb in einem Gebäude zugelassen sind.

- **maximale Tx-Leistung**
  Legen Sie die maximal mögliche Sendeleistung des Geräts fest. Wenn die Sendeleistung zu hoch eingestellt ist, kann das empfangene Signal beim Client übersteuert sein. Kontrollieren Sie beim Client die Empfangssignalstärke (dBm). Abhängig von den verwendeten Antennen kann eine Verminderung der Sendeleistung notwendig werden, um die gesetzlich vorgeschriebene maximale Sendeleistung nicht zu überschreiten. Eine Verminderung der Sendeleistung bewirkt eine gezielte Reduktion der Zellengröße.

  Hinweis

  Je nach Kanal und Datenrate variiert die maximal mögliche Sendeleistung. Beachten Sie für weitere Informationen zur Sendeleistung die Dokumentation "Leistungsdaten Funkkarte / Characteristics radio interface".

  Hinweis

  Werden bei Access Points mit zwei WLAN-Schnittstellen beide Schnittstellen im gleichen Frequenzbereich betrieben, kann es bei einer Sendeleistung größer 15 dBm zu Funkstörungen an einer bzw. an beiden Schnittstellen kommen.

- **Prüfung der Tx-Leistung (Nur Online verfügbar)**
  Zeigt an, ob mit den vorgenommenen Einstellungen die zulässigen Sendeleistungsbeschränkungen des eingestellten Landes verletzt werden. Einfluss auf diese Berechnung haben folgende Parameter:
  max. Tx-Leistung, Antennengewinn, zusätzliche Dämpfung.
  Es gibt folgende Anzeigen:
  - Zugelassene
    Die Kanäle können mit den aktuellen Einstellungen benutzt werden.
  - Nicht zulässig (einige Kanäle)
    Unter den Kanälen gibt es welche, bei denen die aktuelle Sendeleistung die maximal erlaubte Sendeleistung überschreitet.
  - Nicht zulässig (alle Kanäle)
    Kein zulässiger Betrieb möglich. Die Sendeleistung ist zu hoch.
  - Automatisch mit iREF gesteuert
    Von der Funktion iREF gesteuert.

Erweiterungen

Sie müssen die Einstellungen nur anpassen, wenn das Gerät mit den Standardeinstellungen nicht in der vorgesehenen Art und Weise genutzt werden kann.

Einstellungen

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Beacon Interval [ms]** (nur im Access Point-Modus)
  Legen Sie das Interval (40 - 1000 ms) fest, mit dem der Access Point Beacons sendet. Beacons sind zyklisch versendete Pakete, mit denen ein Access Point die Clients über seine Existenz informiert.

- **DTIM** (nur im Access Point-Modus)
  Das DTIM-Intervall (1-15) legt die Anzahl der Beacons fest, die versendet werden, bevor, nach dessen Ablauf der Access Point die gesammelten Pakete (Broadcast, Unicast, Multicast) an den Client sendet.
  - Wenn Sie eine "1" eintragen, überträgt der Access Point Broadcast-, Unicast- und Multicast-Pakete direkt nach jedem Beacon (empfohlene Einstellung für normale Netzwerkumgebungen).
  - Wenn Sie eine "5" eintragen, sammelt der Access Point die Pakete und sendet diese nach jedem fünften Beacon.

  Ein Vergrößern dieses Werts erlaubt den verbundenen Clients einen längeren Sleep-Modus auf Kosten einer höheren Verzögerung bei Paketen.
- **RTS/CTS Schwellenwert [Bytes]**
  RTS/CTS (Request To Send/Clear To Send) ist ein Verfahren zur Kollisionsvermeidung. Das Verfahren beruht auf dem Austausch von Statusinformationen vor dem Senden der eigentlichen Daten (Hidden node problem). Um die Netzbelastung durch den zusätzlichen Protokollverkehr zu minimieren, wird dieses Verfahren erst ab einer bestimmten Paketgröße angewendet. Die Paketgröße legen Sie mit dem Parameter "RTS/CTS Threshold" fest.
- **Schwellenwert der Fragmentierungslänge [Bytes]**
  Legen Sie die maximale Paketgröße fest, die auf der Funkstrecke übertragen wird. Größere Pakete werden vor dem Senden in kleinere aufgeteilt und nach dem Empfangen wieder zur ursprünglichen Größe zusammengesetzt. Dies kann bei sehr schlechter Übertragungsqualität vorteilhaft sein, da größere Pakete schwieriger zu übertragen sind. Jedoch bringt die Fragmentierung in kleinere Pakete einen schlechteren Datendurchsatz mit sich.

  Hinweis

  Sie können diesen Wert nur bearbeiten, wenn Sie den Übertragungsstandard "802.11g" (2,4 GHz) oder "802.11 a" (5 GHz) eingestellt haben, siehe "WLAN-Modus" unter "Schnittstellen > WLAN > Basic".

- **Hardware-Wiederholungen**
  Legen Sie die Anzahl der Hardware-Wiederholungen fest. Die Hardware-Wiederholung wird vom WLAN-Chip selbst durchgeführt, indem er versucht, ein nicht quittiertes Paket sofort zu wiederholen.
  Waren alle Hardware-Wiederholungen erfolglos, wird das Paket gelöscht.

  Hinweis

  Wenn Sie iPCF oder iPCF-MC aktiviert haben, können Sie nur bis zu 16 Hardware-Wiederholungen einstellen

- **Multi Radar Erfassung** (nur im Access Point-Modus)
  - Aktiviert
    Diese Funktion ist nur verfügbar, wenn Sie auf der Seite "Basic" die Funktion "DFS" aktiviert haben. Diese Funktion eignet sich für Anlagen mit mehreren Access Points, die über ein Ethernet-Netzwerk miteinander verbunden sind und auf dem gleichen Kanal senden. Wenn ein Access Point ein Radarsignal erkennt, verteilt er diese Information an alle verbundenen Access Points. Wenn mindestens ein weiterer Access Point das Radarsignal innerhalb von 40 ms verifiziert, werden alle verbundenen Access Points informiert. Alle Geräte, die auf diesem Kanal senden, wechseln auf einen anderen Kanal. Der Kanal ist für alle Access Points in dem Netzwerk für 30 Minuten gesperrt. Wenn Sie auf der Seite "Schnittstellen > WLAN > AP" für den Kanal "Auto" konfiguriert haben, kann die Funktion nicht zuverlässig eingesetzt werden. In diesem Fall ist die Verifizierung der Radarsignale nur möglich, wenn mindestens zwei verbundene Access Points zufällig auf dem gleichen Kanal senden. Wenn nur ein Access Point auf einem Kanal ein Signal erkennt, behandelt er es als gültiges Radarsignal.
  - Deaktiviert
    Die Funktion wird nicht verwendet. Wenn ein Access Point ein Radarsignal erkennt, wechselt er zu einem anderem Kanal. Der konfigurierte Kanal wird nicht mehr berücksichtigt.
- **Konfigurierten DFS-Kanal bevorzugen** (nur im Access Point-Modus)
  - Aktiviert
    Diese Funktion ist nur verfügbar, wenn Sie auf der Seite "Basic" die Funktion "DFS" aktiviert haben. Wenn der konfigurierte Kanal einer WLAN-Schnittstelle durch Radarerkennung gesperrt wurde und nach 30 Minuten wieder freigegeben wird, wechselt der Access Point automatisch wieder zu dem konfigurierten Kanal. Bevor der Access Point die Kommunikation auf dem konfigurierten Kanal startet, sucht er 60 Sekunden nach Primärnutzern auf dem Kanal. In dieser Zeit sendet der Access Point keine Beacons. Wenn Signale auf dem Kanal entdeckt werden, wechselt der Access Point den Kanal und wiederholt die Prüfung. Erst wenn nach Ablauf der 60 Sekunden keine Signale von Primärnutzern erkannt wurden, sendet der Access Point auf dem Kanal. Wenn Sie auf der Seite "Interfaces > WLAN > AP" für den Kanal "Auto" konfiguriert haben, hat das Gerät keinen konfigurierten Kanal, zu dem es zurückwechseln kann.
  - Deaktiviert
    Die Funktion wird nicht verwendet.

Antennen

Überblick

IWLAN-Antennen im Überblick:

Antennenart	Frequenz (GHz)	Antennen	SCALANCE W780/W740	SCALANCE W770/W730	SCALANCE W760/W720	SCALANCE W1780/W1740
omnidirektional	2.4	ANT792-6MN	●	●	●	●
	2.4 und 5	ANT795-4MA	●	●	●	●
		ANT795-4MB	●	●	●	●
		ANT795-4MC	●	●		●
		ANT795-4MD	●	●		●
		ANT795-4MX	●	●		●
		ANT795-6MN	●	●	●	●
		ANT795-6MT	●			●
		ANT795-6MP	●	●	●	●
	5	ANT793-6MN	●	●	●	●
Sektor	2.4 und 5	ANT795-6DC	●	●	●	●
	5	ANT793-6DG	●	●		●
		ANT793-6DT	●			●

Antennenart	Frequenz (GHz)	Antennen	SCALANCE W780/W740	SCALANCE W770/W730	SCALANCE W760/W720	SCALANCE W1780/W1740
direktional	2.4	ANT792-8DN	●			●
	5	ANT793-8DP	●	●	●	●
		ANT793-8DJ	●	●		●
		ANT793-8DK	●	●		●
		ANT793-8DL	●	●		●
RCoax	2.4	RCoax Leckwellen-leiter 2.4 GHz	●	●	●	●
		ANT792-4DN	●	●	●	●
	5	RCoax Leckwellen-leiter 5 GHz	●	●	●	●
		ANT793-4MN	●	●	●	●
Mobilfunk für GSM	2.4 und 5	ANT896-6MM				●

Der Antennenname gibt Aufschluss über die Eigenschaften der in der IWLAN-Antennenübersicht dargestellten Antennen:

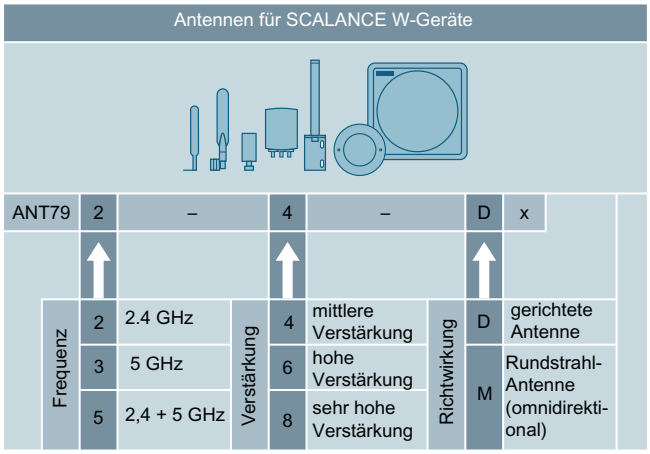

Antennen

Auf dieser Seite konfigurieren Sie die Einstellungen für die angeschlossenen externen Antennen.

Hinweis

Übertragungsstörungen ohne angeschlossene Antenne

Eine WLAN-Schnittstelle kann bis zu drei Antennenanschlüsse haben. Am ersten Anschluss muss immer eine Antenne angeschlossen sein, sobald die zugehörige WLAN-Schnittstelle eingeschaltet ist.

Nicht verwendete Anschlüsse müssen mit einem 50 Ω-Abschlusswiderstand versehen werden.

Ist keine Antenne angeschlossen, muss auch die entsprechende WLAN-Schnittstelle für RX und TX deaktiviert sein. Andernfalls kann es zu Übertragungsstörungen kommen.

Einstellungen

- **Anschluss**
  Zeigt die Bezeichnung des jeweiligen Antennenanschlusses an.
- **Antennentyp**
  Wählen Sie den Typ der externen Antenne aus, die an das Gerät angeschlossen ist. Wenn der Typ ihrer externen Antenne nicht verfügbar ist, wählen Sie den Eintrag "Benutzerdefiniert".
  Wenn Sie einen Antennenanschluss mit einem 50 Ω-Abschlusswiderstand abschließen, wählen Sie den Eintrag "Nicht verwendet (50-Ohm-Abschlusswiderstand)".

- **Antennengewinn**
  Wenn Sie bei "Antennentyp" den Eintrag "Benutzerdefiniert" wählen, tragen Sie manuell den Antennengewinn in der Einheit "dBi" ein.
  - **Antennengewinn 2.4 GHz [dBi]**
    Tragen Sie hier den Antennengewinn ein, den die Antenne im Frequenzband 2.4 GHz hat.
  - **Antennengewinn 5 GHz [dBi]**
    Tragen Sie hier den Antennengewinn ein, den die Antenne im Frequenzband 5 GHz hat.
- **Leitungslänge [m]**
  Tragen Sie die Länge der flexiblen Antennen-Anschlussleitung in Metern ein, die zwischen dem Gerät und der externen Antenne verwendet wird.
- **Zusätzliche Dämpfung [dB]**
  Tragen Sie hier die zusätzliche Dämpfung ein, die z.B. durch einen zusätzlichen Splitter verursacht wird.
- **Antennenmodus**
  Legen Sie die Verwendung der Antenne fest. Beim Antennenanschluss 1 (R1 A1 und R2 A1) ist der Eintrag nicht änderbar.
  - TX
    Nur zum Senden
  - RX
    Nur zum Empfangen
  - RX\TX
    Zum Empfangen und zum Senden

  Die folgende Tabelle zeigt, welche Kombinationen möglich sind:

R1 A1 / R2 A1	R1 A2 / R2 A2	R1 A3 / R2 A3
RX\TX	RX\TX	RX\TX
RX\TX	RX\TX	RX
RX\TX	RX	RX
RX\TX	RX\TX	TX
RX\TX	TX	TX
RX\TX	RX\TX	-- 1)
RX\TX	TX	-- 1)
RX\TX	RX	-- 1)
RX\TX	-- 1)	-- 1)

1) Antennentyp "50 Ω-Abschlusswiderstand"

Zugelassene Kanäle

Für die Kommunikation wird ein bestimmter Kanal innerhalb eines Frequenzbandes verwendet. Sie können diesen Kanal entweder fest vorgeben oder so konfigurieren, dass eine automatische Kanalauswahl erfolgt.

Einstellungen

Die Tabelle 1 enthält folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Nur zugelassene Kanäle verwenden**
  Wenn Sie die Einstellung aktivieren, schränken Sie damit die Auswahl an Kanälen ein, über die der AP bzw. der Client die Verbindung aufbauen darf.
  In den folgenden Tabellen definieren Sie an
  - welche Kanäle der AP zum Aufbau einer Funkzelle bei der Kanaleinstellung "Auto" nutzen darf.
  - auf welchen Kanälen der Client nach einem AP sucht.

  Die Tabellen sind nach den Frequenzbändern getrennt.
  Wenn deaktiviert, werden die entsprechend den Einstellungen (Ländercode, Antennen, Sendeleistung etc.) verfügbaren Kanäle benutzt.

Über den Tabellen der Frequenzbänder befindet sich jeweils folgende Einstellung:

- **Alle auswählen/abwählen**
  - Aktiviert
    Alle Kanäle werden aktiviert.
  - Deaktiviert
    Der erste gültige Kanal des Frequenzbands bleibt aktiviert. Aktivieren Sie den gewünschten Kanal.

Die Tabellen der Frequenzbänder enthalten folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Modus der Funkschnittstelle**
  Zeigt die Betriebsart an.
- **Kanalnummer**
  Um die gültigen Kanäle für das gewünschte Frequenzband festzulegen, aktivieren Sie bei der Kanalnummer das entsprechende Kontrollkästchen.
  In der Tabelle werden die zulässigen Kanäle des Landes angezeigt. Nur die gültigen Kanäle lassen sich aktivieren. Nicht gültige Kanäle sind gegraut und können nicht aktiviert werden.

  Hinweis

  Um die Kanäle festzulegen, muss die Einstellung "Nur zugelassene Kanäle verwenden" aktiviert sein.

802.11n

Eigenschaften des 802.11n

Mit dem Standard IEEE 802.11n ist es möglich Einzel-Datenpakete zu einem größeren Datenpaket zusammenzuführen, A-MPDU und A-MSDU-Datenpakete. Dadurch wird ein höherer Datendurchsatz erzielt.
Auf dieser Seite legen Sie die Einstellungen für die A-MPDU und A-MSDU-Datenpakete fest. Einige der Einstellungen sind abhängig vom eingestellten Übertragungsstandard und von der gewählten Kanalbreite.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **A-MPDU**
  Aktiviert oder deaktiviert das Versenden von A-MPDU-Datenpakete. Wenn diese Einstellung deaktiviert ist, werden nur A-MPDU-Datenpakete empfangen.
- **A-MPDU Limit [Frames]**
  Legen Sie die Anzahl der Einzel-Datenpakete fest, die in einem AMPDU-Datenpaket zusammengefasst werden.
- **A-MPDU Limit [Bytes]**
  Legen Sie die Größe für das AMPDU-Datenpaket fest, die es maximal erreichen darf.
- **A-MSDU**
  Aggregated MAC Service Data Unit (A-MSDU) Aktiviert oder deaktiviert, dass mehrere MSDUs mit der gleichen Zieladresse zu einem AMSDU gebündelt und zusammen versendet werden. Dadurch wird die Netzlast verringert. A-MSDUs eignen sich durch ihre kürzere maximale Länge eher für die Bündelung mehrerer kürzerer Frames. Wenn dieses Optionskästchen deaktiviert ist, werden A-MSDU-Datenpakete empfangen, aber nicht versendet.
- **A-MSDU Telegrammlänge [Bytes]**
  Legen Sie die Größe für das A-MSDU-Datenpaket fest, die es maximal erreichen darf.
  Wertebereich: 50 - 200 Bytes
  Defaultwert: 100 Bytes
- **Guard Interval [ns] (nur im Access Point-Modus)**
  Wählen Sie die Sendepause, die zwischen zwei übertragenen OFDM-Symbolen eingehalten werden muss.
  Folgende Einstellungen sind möglich. Die Auswahl ist abhängig vom gewählten Übertragungsstandard.
  - 400 (short)/800 (long): Die Einstellung 400 ns ist optional. Je nach Signalqualität können Pakete mit einer Sendepause von 400 ns oder 800 ns geschickt werden
  - 800 (long): Die Sendepause beträgt 800 ns.

Access Point

Hinweis

Die Seite ist nur im Access Point-Modus verfügbar.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Kanal**
  Legen Sie den Hauptkanal fest.
  Wenn sich der Access Point selbst einen freien Kanal suchen soll, verwenden Sie "Auto". Die Auswahl der Kanäle, die ein Access Point beim Aufbau einer Funkzelle nutzt, kann begrenzt werden. Aktivieren Sie bei "Zugelassene Kanäle" die Einstellung "Nur ausgewählte Kanäle verwenden". Wenn Sie einen festen Kanal nutzen wollen, wählen Sie den gewünschten Kanal aus der Klappliste.

  Hinweis

  SCALANCE W780/W740: Kanalabstand bei WLAN-Schnittstellen

  Bei Betrieb einer zweiten WLAN-Schnittstelle müssen Sie auf einen ausreichenden Kanalabstand achten.

- **Alternativer DFS-Kanal (802.11h)**
  Wenn Sie auf der Seite "Basic" die Funktion "DFS" aktiviert haben, legen Sie hier den Alternativ-Kanal fest. Wenn sich der Access Point selbst einen freien Kanal suchen soll, verwenden Sie "Auto".
  Wurde sowohl auf dem Haupt- als auch auf dem Alternativkanal ein Primärnutzer entdeckt, sucht sich der Access Point selbstständig einen freien Kanal aus.
  Wenn Sie einen festen Kanal nutzen wollen, wählen Sie den gewünschten Kanal aus der Klappliste.
- **HT-Kanal-Bandbreite [MHz]**
  Nur beim Übertragungsstandard IEEE 802.11n können Sie die Kanalbandbreite festlegen. Folgende Einstellungen sind möglich.
  - 20
    Kanalbandbreite 20 MHz
  - 40 up
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darüberliegende benachbarte Kanal werden verwendet.
  - 40 down
    Kanalbandbreite 40 MHz. Der konfigurierte Kanal und der darunterliegende benachbarte Kanal werden verwendet.

  Hinweis

  Kanalbandbreite 40 MHz und Frequenzband 2,4 GHz

  Wenn der Access Point auf dem konfigurierten Kanal oder auf benachbarten Kanälen einen anderen Access Point entdeckt, wechselt der Access Point von der Kanalbandbreite 40 MHz auf 20 MHz. Wenn Sie auf dem Access Point einen "freien" Kanal einstellen, verwendet der Access Point die Kanalbandbreite 40 MHz.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen in dieser Spalte an.
- **Verfügbare Kanäle**
  Zeigt die zulässigen Kanäle an. Die Anzeige ist abhängig von den Funkzulassungen des aktuell gewählten Landes und den Einstellungen bei "Zugelassene Kanäle".

Die Tabelle 3 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die WLAN-Schnittstelle an.
- **Port**
  Zeigt die VAP-Schnittstelle an.
- **Aktiviert**
  Um die gewünschte VAP-Schnittstelle zu nutzen, aktivieren Sie diese Einstellung.
- **SSID**
  Tragen Sie die SSID des WLANs ein. Die Länge der Zeichenkette für SSID beträgt 1 bis 32 Zeichen.
  Für die SSID werden die ASCII-Zeichen 0x20 bis 0x7e verwendet.

- **Broadcast-SSID**
  - deaktiviert
    Die SSID wird nicht mehr im Beacon-Frame des Access Points versendet. Somit ist die SSID für andere Geräte nicht sichtbar. Nur die Clients können sich mit dem Access Point verbinden, denen die SSID des Access Points bekannt ist, und die mit dieser konfiguriert sind. Die Einstellung "Beliebige SSID" muss auf diesen Clients deaktiviert sein.
  - aktiviert
    Die SSID wird im Beacon-Frame des Access Points versendet und ist für andere Geräte sichtbar. So können auch Clients sich mit dem Access Point verbinden, bei denen die Einstellung "Beliebige SSID" aktiviert ist.

Hinweis

Da bei der SSID-Übertragung keine Verschlüsselung eingesetzt wird, kann diese Funktion nur grundlegend vor unberechtigten Zugriffen schützen. Die Nutzung einer Authentifizierungsmethode (z.B. WPA2 (RADIUS), wenn nicht möglich WPA2-PSK) bietet eine höhere Sicherheit. Es muss zudem damit gerechnet werden, dass gewisse Endgeräte Probleme mit dem Zugriff auf eine versteckte SSID haben können.

- **Nur WDS**
  Wenn Sie diese Einstellung aktivieren, unterstützt der Access Point nur die Kommunikation über WDS. In der Betriebsart WDS müssen alle Access Points den gleichen Kanal benutzen.
- **WDS ID**
  Tragen Sie die WDS ID ein. Die WDS ID darf maximal 32 Zeichen lang sein.
  Um eine WDS-Verbindung aufzubauen, tragen Sie beim WDS-Partner diese WDS ID ein.
  Für die WDS ID werden die ASCII-Zeichen 0x20 bis 0x7e verwendet.

AP WDS

Im normalen Betrieb ist der Access Point als Schnittstelle zu einem Netz eingesetzt und kommuniziert mit Clients. Es gibt allerdings auch den Anwendungsfall, dass mehrere Access Points miteinander kommunizieren müssen, beispielsweise zum Zweck der Reichweitenvergrößerung oder zum Aufbau eines Wireless-Backbones. Diese Betriebsart ist mit WDS (Wireless Distributed System) möglich.

Hinweis

Dieses Register ist nur im Access Point-Modus verfügbar.

Einstellungen

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt den Port an.

- **Port aktiviert**
  Aktiviert die WDS-Schnittstelle.
- **Verbindung über**
  Legen Sie fest, über welche VAP-Schnittstelle die WDS-Verbindung hergestellt wird. Es werden sowohl die MAC-Adresse des VAPs als auch die Security-Einstellungen (z.B. WPA2) benutzt.
- **Partner ID Typ**
  Legen Sie die Art der WDS-Kommunikation fest.
  – MAC-Adresse
    Die MAC-Adresse wird verwendet. Das Eingabefeld "WDS-ID des Partner" wird gegraut. Tragen Sie bei "MAC-Adresse des Partners" die MAC-Adresse des WDS-Partners ein.
  – WDS-ID
    Die WDS-ID wird verwendet. Das Eingabefeld "MAC-Adresse des Partners" wird gegraut. Tragen Sie bei "WDS-ID des Partner" die WDS-ID des WDS-Partners ein. Verwenden Sie diese Option, wenn Sie den Access Point später mithilfe des C-PLUG oder KEY-PLUG austauschen wollen.
- **MAC-Adresse des Partners**
  Tragen Sie die MAC-Adresse des WDS-Partners ein. Nur editierbar, wenn bei "Partner ID Typ" "MAC-Adresse" eingestellt ist.
- **WDS-ID des Partner**
  Tragen Sie die WDS ID des WDS-Partners ein. Nur editierbar, wenn bei "Partner ID Typ" "WDS-ID" eingestellt ist. Für die WDS-ID sind die ASCII-Zeichen 0x20 bis 0x7e erlaubt.

Hinweis

Übereinstimmende Sicherheitseinstellungen im WDS-Betrieb

Achten Sie im WDS-Betrieb auf übereinstimmende Sicherheitseinstellungen bei allen beteiligten Geräten. Bei fehlerhaften oder nicht kompatiblen Einstellungen bei den einzelnen Geräten kann kein Datenaustausch aufgrund fehlerhafter Authentifizierung stattfinden. Vermeiden Sie die Einstellung "Auto" bei den Sicherheitseinstellungen des Basic Wizards, weil mit dieser Einstellung eine Synchronisation der Sicherheitseinstellungen zwischen den Access Points nicht möglich ist.

Hinweis

Im WDS-Betrieb gelten alle daran beteiligten Access Points folgende Einschränkungen:

- Alle Access Points, die miteinander kommunizieren sollen, müssen den gleichen Kanal, das gleiche Übertragungsverfahren und die gleiche Datenrate benutzen.
- Als Verschlüsselungsverfahren können Sie entweder WEP oder WPA(2)-PSK auswählen.
  Die Sicherheits-Einstelllungen konfigurieren Sie in der jeweils zugeordneten VAP-Schnittstelle: "Security > WLAN > Grundeinstellungen"
  Die Authentifizierung über RADIUS Server können Sie nicht für eine WDS-Verbindung nutzen.
- Im Übertragungsverfahren IEEE 802.11h ist es nicht sinnvoll, die Betriebsart WDS zu wählen. In der Betriebsart WDS müssen alle Access Points den gleichen Kanal benutzen. Wird jetzt von einem Access Point ein Signal eines Primärnutzers entdeckt, wird der Kanal automatisch gewechselt und dadurch die bestehende Verbindung unterbrochen.

AP 802.11a/b/g-Datenraten

Hinweis

Die Seite ist nur

- im Access Point-Modus verfügbar.
- konfigurierbar, wenn bei WLAN-Modus "802.11a", "802.11g" oder "802.11n" eingestellt ist.

Auf der Seite werden die verfügbaren Datenübertragungsgeschwindigkeiten für den WLAN-Modus 802.11a/b/g anzeigt. Sie können bei Bedarf die Datenübertragungsgeschwindigkeiten ändern. Ansonsten empfehlen wir, die Standardeinstellung für Datenübertragungsgeschwindigkeiten beizubehalten. Ausschließlich die gewählten Datenübertragungsgeschwindigkeiten werden dann vom Access Point zur Kommunikation mit den Clients verwendet.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Informationen beziehen.
- **Nur ausgewählte Datenraten verwenden**
  Wenn Sie die Einstellung aktivieren, können Sie die Datenübertragungsgeschwindigkeiten für die gewünschte WLAN-Schnittstelle festlegen.
  Wenn diese Einstellung deaktiviert ist, werden die Standardwerte verwendet.

Funkschnittstelle
Wählen Sie die WLAN-Schnittstellen aus, die in Tabelle 3 (Datenrate) angezeigt werden.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Alle Einstellungen für die Datenrate**
  Zeigt an, dass die Einstellung für alle Einträge der Tabelle 3 gültig ist.
- **Aktiviert / Grundeinstellung**
  Wählen Sie die gewünschte Einstellung. Wenn "No Change" ausgewählt ist, bleibt der Eintrag in der Tabelle 3 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Einträge der Tabelle 3 übernommen

Die Tabelle 3 (Datenrate) gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Informationen beziehen.
- **Datenrate [Mbps]**
  Zeigt die unterstützten Datenübertragungsgeschwindigkeiten in Megabits pro Sekunde an.

- **Aktiviert**
  Um der WLAN-Schnittstelle die gewünschte Datenübertragungsgeschwindigkeit zuzuweisen, aktivieren Sie die Einstellung.

  Hinweis

  Sie müssen mindestens eine Datenübertragungsgeschwindigkeit aktivieren.

- **Grundeinstellung**
  Um die gewünschte Datenrate als "Grundeinstellung" zu deklarieren, aktivieren Sie die Einstellung. Die Einstellung legt fest, dass ein Client die entsprechende Geschwindigkeit zwingend beherrschen muss, um sich mit dem Access Point verbinden zu können. Die Einstellung ist nur aktivierbar, wenn eine verfügbare Datenrate ausgewählt ist.

  Hinweis

  Sie müssen mindestens eine Datenübertragungsgeschwindigkeit als "Grundeinstellung" festlegen.

Voreinstellungen
Die Schaltfläche "Voreinstellungen" setzt die Auswahl der Werte konform zum Standard.

AP 802.11n-Datenraten

Hinweis

Die Seite ist nur

- im Access Point-Modus verfügbar.
- konfigurierbar, wenn bei WLAN-Modus "Nur 802.11n" oder "802.11n" eingestellt ist.

Auf der Seite werden die verfügbaren Datenübertragungsgeschwindigkeiten (MCS = Modulation and Coding Schemes) für den WLAN-Modus 802.11n anzeigt. Sie können eine beliebige Kombination dieser Datenübertragungsgeschwindigkeiten auswählen. Ausschließlich die gewählten Datenübertragungsgeschwindigkeiten werden dann vom Access Point zur Kommunikation mit den Clients verwendet.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Informationen beziehen.
- **Nur ausgewählte Datenraten verwenden**
  Wenn Sie die Einstellung aktivieren, können Sie die Datenübertragungsgeschwindigkeiten für die gewünschte WLAN-Schnittstelle festlegen.
  Wenn diese Einstellung deaktiviert ist, werden die Standardwerte verwendet.
  Standardmäßig ist die Einstellung deaktiviert.

Funkschnittstelle
Wählen Sie die WLAN-Schnittstellen aus, die in Tabelle 3 (MCS Index) angezeigt werden.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Alle Einstellungen für die Datenrate**
  Zeigt an, dass die Einstellung für alle Einträge der Tabelle 3 gültig ist.
- **Aktiviert**
  Wählen Sie die gewünschte Einstellung. Wenn "No Change" ausgewählt ist, bleibt der Eintrag in der Tabelle 3 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Einträge der Tabelle 3 übernommen.

Die Tabelle 3 (MCS Index) gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Informationen beziehen.
- **MCS-Index**
  Zeigt die unterstützten MCS Indizes an. Die angezeigten MCS Indizes sind abhängig von den Einstellungen "Antennentyp" und "Antennen-Modus". Die Einstellungen finden Sie unter "Schnittstellen > WLAN > Antennen". Wenn Sie z. B. nur eine Antenne verwenden, werden nur die MCS 0 bis 7 angezeigt.
- **Streams**
  Zeigt die maximal mögliche Anzahl von parallelen Datenströmen an, die mit dem ausgewählten MCS Index übertragen werden können.
- **Datenrate [Mbps]**
  Zeigt die unterstützten Datenübertragungsgeschwindigkeiten in Megabits pro Sekunde an. Die angezeigten Datenübertragungsgeschwindigkeiten sind abhängig von den Einstellungen "Guard Interval" und "HT-Kanal-Bandbreite". Die Einstellung "HT-Kanal-Bandbreite" finden Sie unter "Schnittstellen > WLAN > Access Point". Die Einstellung "Guard Interval" finden Sie unter "Schnittstellen > WLAN > 802.11n"
- **Aktiviert**
  Um der WLAN-Schnittstelle die gewünschte Datenübertragungsgeschwindigkeit zuzuweisen, aktivieren Sie die Einstellung.

  Hinweis

  Sie müssen mindestens einen MCS Index aktivieren.

Voreinstellungen
Die Schaltfläche "Voreinstellungen" setzt die Auswahl der Werte konform zum Standard.

Client

Netzanbindung

Auf dieser Seite legen Sie fest, wie sich das Gerät als Client mit einem Netz verbindet.

Hinweis

Die Seite ist nur verfügbar bei Clients oder Access Points im Client-Modus.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.

  Hinweis

  WLAN-Schnittstelle deaktiviert

  Die WLAN-Schnittstelle wird deaktiviert, wenn nicht mindestens eine SSID konfiguriert oder die Einstellung "Beliebige SSID" aktiviert ist.

- **MAC-Modus**
  Legen Sie fest, wie dem Client eine MAC-Adresse zugeordnet wird. Es gibt die folgenden Möglichkeiten.
  - Automatisch
    Der Client nimmt automatisch die Quell-MAC-Adresse des ersten Telegramms an, das er über die Ethernet-Schnittstelle empfängt.
  - Manuell
    Wenn Sie Manuell wählen, geben Sie die MAC-Adresse bei "MAC-Adresse" ein.
  - Eigene
    Der Client verwendet die MAC-Adresse der Ethernet-Schnittstelle für die WLAN-Schnittstelle.
  - Layer-2-Tunnel
    Der Client verwendet die MAC-Adresse der Ethernet-Schnittstelle für die WLAN-Schnittstelle. Zusätzlich wird das Netzwerk über die an der Ethernet-Schnittstelle des Clients angeschlossenen MAC-Adressen informiert. Es können bis zu acht MAC-Adressen verwendet werden.
- **MAC-Adresse**
  Wenn Sie bei "MAC-Modus" "Manuell" ausgewählt haben, geben Sie die MAC-Adresse des Clients ein.
- **Beliebige SSID**
  - Aktiviert
    Das SCALANCE W700-Gerät versucht sich im Client-Modus mit dem Access Point zu verbinden, der den Security-Einstellungen des Security-Kontext 1 entspricht. Die Clients können sich nur mit dem Access Point verbinden, bei dem die Option "Broadcast SSID" aktiviert ist.
  - Deaktiviert
    Der Client versucht sich mit dem Access Point aus der SSID-Liste zu verbinden, dessen Security-Einstellungen zu einem der definierten Security-Kontexte passt.

- Nach Roaming DHCP erneuern
  - Aktiviert
    Nach dem Wechsel zu einem anderen Access Point, wird geprüft ob die IPv4-Adresse des Clients noch gültig ist. Wenn die IPv4-Adresse ungültig ist, wird eine neue IPv4-Adresse vom DHCP-Server angefordert.
  - Deaktiviert
    Wenn der Client zu einem anderen Access Point wechselt wird die IPv4-Adresse nicht geprüft.
- Roaming-Schwellenwert
  Legen Sie die Schwelle fest, ab wann der Client zum neuen Access Point wechselt.
  - Hoch
    Wechselt erst bei deutlich höherer Feldstärke zum AP mit stärkerem Signal.
  - Mittel
    Wechselt bei mäßig höherer Feldstärke zum AP mit stärkerem Signal.
  - Niedrig
    Wechselt schon bei geringfügig höherer Feldstärke zum AP mit stärkerem Signal.
- min AP-Signalstärke
  Dem Client wird eine Signalstärke vorgegeben.Der Client muss das vom Access Point kommende Signal mindestens mit der vorgegebenen Signalstärke empfangen, um sich mit diesem Access Point zu verbinden.
  Die Signalstärke kann kurzfristig schwanken, z. B. durch Bewegung des Clients oder andere Störfaktoren. Um diese Schwankungen beim Signal herauszufiltern, wird mit einer Hysterese ein Bereich um diesen Wert festgelegt, in welchem der Client nicht den Access Point wechselt, bevor dieser Bereich unterschritten wird. Wenn das vom Access Point kommende Signal unter diesen Bereich fällt, trennt sich der Client von dem verbundenen Access Point und sucht nach einem neuen Access Point.
- Background Scan-Modus
  Während der Client mit einem Access Point verbunden ist, sucht er im Hintergrund nach weiteren Access Points, mit denen er sich gegebenenfalls verbinden kann. Legen Sie den Modus für die Suche fest.
  Folgende Möglichkeiten gibt es:
  - always
    Wenn der Background Scan-Schwellenwert unterschritten wird, sucht der Client kontinuierlich nach Access Points.
  - idle
    Wenn für eine bestimmte Zeit kein Datentransfer stattfindet, wird nach weiteren Access Points gesucht.
  - disabled
    Solange der Client verbunden ist, wird nicht nach weiteren Access Points gesucht.
- Background Scan-Intervall [ms]
  Legen Sie das Intervall fest, in dem nach weiteren Access Points gesucht wird.
- Background Scan-Schwellenwert dBm]
  Legen Sie den Schwellenwert fest. Wenn der Schwellenwert unterschritten wird, sucht der Client nach weiteren Access Points.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die WLAN-Schnittstelle an.
- **Scan-Kanäle**
  zeigt die Kanäle an, auf welchen der Client nach einem Access Point sucht. Die Anzeige ist abhängig von den Funkzulassungen des gewählten Landes und den Einstellungen unter "Zugelassene Kanäle".

Die Tabelle 3 gliedert sich in folgende Spalten:

- **Aktiviert**
  Aktiviert oder deaktiviert die jeweilige SSID.
- **Funkschnittstelle**
  Zeigt die WLAN-Schnittstelle an.
- **SSID**
  Tragen Sie die SSID des Access Points ein, mit der sich der Client verbinden soll.
  Für die SSID wird der ASCII-Code 0x20 bis 0x7e verwendet.
- **Security**
  Wählen Sie einen Security-Kontext aus. Sie erstellen und konfigurieren einen Security-Kontext unter "Security > WLAN > Basic".
  Standardeinstellung: Kontext 1

Hinweis

iPCF- bzw. iPCF-MC-Modus aktiviert

Wenn der iPCF- bzw. iPCF-MC-Modus aktiviert ist, können Sie nur den Security Context 1 auswählen.

Roaming erzwingen

Hinweis

Die Seite ist nur verfügbar

- im Access Point-Modus
- wenn eine Online-Verbindung zum Gerät besteht.

Die WLAN-Schnittstelle des Geräts muss aktiviert sein, andernfalls können die Frequenzbereiche nicht gescannt werden.

Wenn die Schnittstelle nicht mehr zur Verfügung steht (Kabelbruch, Netzkomponenten ausgefallen, Stecker gezogen), erhält ein über das Funknetz angebundener Client davon keine Kenntnis. Der Access Point kann die angemeldeten Clients zu einem Roaming zwingen, indem er die entsprechende Schnittstelle abgeschaltet.

Auf dieser Seite legen Sie fest, wann ein Roaming durchgeführt wird.

- Bei Verbindungsabbruch
  Wenn die drahtgebundene Ethernet-Schnittstelle nicht mehr zur Verfügung steht, wird die WLAN-Schnittstelle abgeschaltet. Die Clients roamen und verbinden sich dann auf einen anderen Access Point. Sobald der erste Access Point den Server wieder erreicht, schaltet er seine WLAN-Schnittstellen wieder aktiv.
- Wenn die Zieladresse nicht erreichbar ist
  Das Roaming ist von Zieladressen abhängig. Zum Überwachen sendet das Gerät in regelmäßigen Abständen Ping-Anforderung an die projektierten Zieladressen.
  - VAP-Schnittstelle überwacht durch eine Zieladresse
    Wenn von dieser Zieladresse keine Ping-Antwort empfangen wird, schaltet der Access Point die entsprechende VAP-Schnittstelle ab.
  - VAP-Schnittstelle überwacht durch mehrere Zieladressen
    Erst wenn von keiner dieser Zieladressen eine Ping-Antwort empfangen wird, schaltet der Access Point die entsprechende VAP-Schnittstelle ab. Solange mindestens eine Zieladresse erreichbar ist, bleibt die VAP-Schnittstelle aktiv.

  Der Access Point sendet an die WLAN-Clients, die über diese VAP-Schnittstelle verbunden sind, einen Disassociation-Frame. Die WLAN-Clients roamen und verbinden sich mit einer anderen VAP-Schnittstelle. Ist die Adresse wieder erreichbar, kann die Verbindung über diese VAP-Schnittstelle wieder aufgebaut werden.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Roaming bei Verbindungsabbruch erzwingen**
  Wenn aktiviert, wird beim Verbindungsabbruch über die Ethernet-Schnittstelle, die WLAN-Schnittstelle abgeschaltet.

Die Tabelle "IP-Adresse unerreichbar - Roaming erzwingen" gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Ziel-Adresse**
  Geben Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des Ziels an, dessen Erreichbarkeit geprüft wird.

  Hinweis

  Zieladresse nicht im Agent IP-Subnetz

  Wenn die Zieladresse nicht in dem Agent IP-Subnetz liegt, ist bei "Layer 2 > Agent IP" ein Gateway einzutragen.

  Base Bridge-Modus "802.1Q VLAN Bridge"

  Wenn Sie unter "Layer 2 > VLAN" den "Base Bridge-Modus" "802.1Q VLAN Bridge" konfiguriert haben, werden die Echomeldungen (Ping-Pakete) in das Management-VLAN gesendet.

- **Intervall [ms]**
  Legen Sie fest, in welchen Zeitabständen Echomeldungen (Ping-Anforderung) gesendet werden.
- **Max. Paketverlust**
  Legen Sie die maximale Anzahl der nacheinander verlorenen Ping-Antworten fest. Wenn diese Anzahl bei einer Zieladresse erreicht wird, dann gilt diese Zieladresse als nicht erreichbar (down).
- **VAP X.Y**
  Legen Sie fest, über welche VAP-Schnittstelle die Adresse überwacht wird.

Signalrekorder

Aufzeichnung des effektiven Nutzsignals

Mit dem Signalrekorder wird das effektive Nutzsignal zwischen Access Point und Client aufgezeichnet. Mithilfe dieser Daten können Sie Gebiete mit unzureichendem Nutzsignal auffinden. Der Signalrekorder kann besonders dann vorteilhaft eingesetzt werden, wenn sich der Client auf einer fest vorgegebenen Strecke bewegt.

Hinweis

Die Seite ist nur verfügbar

- bei Clients oder Access Points im Client-Modus
- wenn eine Online-Verbindung zum Gerät besteht

Die WLAN-Schnittstelle des Geräts ist aktiviert, andernfalls wird keine Aufzeichnung durchgeführt.

Beschreibung

Die Darstellung ist in zwei Bereiche aufgeteilt.

- Client
  Stellt die Messung des Clients dar.
- Access Point
  Stellt die Messung des Access Points dar, mit dem der Client aktuell verbunden ist. Vorausgesetzt die Einstellung "Bidirektionale Aufzeichnung" ist aktiviert und auf dem Access Point ist eine Firmware-Version > 6.1 installiert. Der Access Point schickt seine Daten an maximal 3 Clients, auf denen der Signalrekorder läuft. Auf weiteren Clients werden die Access Point-Daten nicht dargestellt.

Beide Bereiche enthalten je zwei Grafiken.

Die erste Grafik enthält die folgenden Elemente:

- Scrollbalken
  Mit dem Scrollbalken können Sie die gesamte Messung durchschauen. Dafür können Sie die Schaltflächen "<<" und ">>" oder die Pfeiltasten auf der Tastatur verwenden.
- Balken (links)
  In dem Balken auf der linken Seite wird das Nutzsignal vom Client / Access Point in Echtzeit angezeigt, entsprechend des abgebildeten Farbschemas. Die graue Linie zeigt das Hintergrundrauschen an.
  Wenn der Client eine iPCF-MC-Verbindung hat, wird das Nutzsignal des Managementkanals mit einer schwarzen Linie dargestellt.
- Farbschema
  Der Bereich > - 35 dBm (Blau) ist der Übersteuerungsbereich, d. h. das WLAN-Signal ist zu stark und wird übersteuert empfangen. Ab ca. - 60 dBm (Gelb) wird das WLAN-Signal schwächer.
- x-Achse
  Die x-Achse zeigt den Verlauf der Messung, in Stichproben und Sekunden.
- Verlaufsdaten
  - Client
    Die Verlaufsdaten zeigen den Wert des effektiven Nutzsignals entsprechend des abgebildeten Farbschemas. Die graue Linie zeigt das Hintergrundrauschen an.
    Wenn der Client während einer Messung den Access Point wechselt (Roaming) oder sich neu verbindet, wird dies durch eine vertikale schwarze Linie angezeigt. An der Linie werden der neue AP Systemname und die BSSID angezeigt.
    Wenn der Client während einer Messung keine Verbindung zu einem Access Point hat, wird kein Nutzsignal angezeigt. Um zu verdeutlichen, dass keine Verbindung zu einem Access Point besteht, wird die BSSID auf 00:00:00:00:00:00 gesetzt und rot dargestellt.
    Wenn der Client eine iPCF-MC-Verbindung hat, wird zusätzlich das Nutzsignal des Managementkanals mit einer schwarzen Linie dargestellt.
  - Access Point
    Die Verlaufsdaten zeigen den Wert des effektiven Nutzsignals entsprechend des abgebildeten Farbschemas. Die graue Linie zeigt das Hintergrundrauschen an.
    Wenn der Client während einer Messung den Access Point wechselt (Roaming) oder sich neu verbindet, wird dies durch eine vertikale schwarze Linie angezeigt.
    Wenn der Access Point die Einstellung "Bidirektionale Aufzeichnung" nicht unterstützt, wird kein Nutzsignal angezeigt.

Die zweite Grafik enthält folgende Elemente:

- Balken (links)
  In dem Balken auf der linken Seite werden die Übertragungsversuche und die Datenrate des Clients / Access Points angezeigt, entsprechend des abgebildeten Farbschemas.
- Farbschema
  Der Bereich > - 35 dBm (Blau) ist der Übersteuerungsbereich, d. h. das WLAN-Signal ist zu stark und wird übersteuert empfangen. Ab ca. - 60 dBm (Gelb) wird das WLAN-Signal schwächer. Die einzelnen Farben werden nochmals unterhalb der Grafik beschrieben.

- x-Achse
  Die x-Achse zeigt den Verlauf der Messung, in Stichproben und Sekunden.
- Verlaufsdaten
  - Client
    Die Verlaufsdaten zeigen die Übertragungsversuche entsprechend des abgebildeten Farbschemas. Die Übertragungsversuche werden als Balken dargestellt. Die Datenrate der gesendeten Datenpakete wird als Linie dargestellt. Wenn der Client während einer Messung den Access Point wechselt (Roaming) oder sich neu verbindet, wird dies durch eine vertikale schwarze Linie angezeigt
  - Access Point
    Die Verlaufsdaten zeigen die Übertragungsversuche entsprechend des abgebildeten Farbschemas. Die Übertragungsversuche werden als Balken dargestellt. Die Datenrate der gesendeten Datenpakete wird als Linie dargestellt.
    Wenn der Client während einer Messung den Access Point wechselt (Roaming) oder sich neu verbindet, wird dies durch eine vertikale schwarze Linie angezeigt. Wenn der Access Point die Einstellung "Bidirektionale Aufzeichnung" nicht unterstützt, wird keine Daten angezeigt

Neben den Grafiken werden folgende Werte angezeigt:

- Status
  Zeigt an, ob der Signalrekorder Werte aufzeichnet oder nicht.
- Current Sample
  Die Nummer der aktuellen Messung
- CL RX-Signal [dBm] / AP RX-Signal [dBm]
  Das effektive Nutzsignal des Clients / des Access Points in dBm
- CL NF [dBm] / AP NF [dBm]
  Das Hintergrundrauschen des Clients / des Access Points in dBm
- CL Retries [%] / AP Retries [%]
  Die Übertragungswiederholungen des Clients / des Access Points in Prozent.
- CL RSSI / AP RSSI
  Der Rohwert der RSSI (Received Signal Strength Indication) des Clients / des Access Points
- CL TX-Rate [Mbps] / AP TX-Rate [Mbps]
  Die durchschnittliche Datenrate der gesendeten Datenpakete während der aktuellen Stichprobe
- CL M-Signal [dBm]
  Wenn der Client eine iPCF-MC-Verbindung hat, wird zusätzlich das Nutzsignal des Managementkanals angezeigt.
- Roaming Counter
  Der Roaming Counter zeigt an, wie oft der Client den Access Point, während der Aufzeichnung, gewechselt hat. Nach 4 294 967 295 Wechseln wird der Zähler zurückgesetzt.
- Operative Channel
  Der aktuelle Kanal bzw. der Kanal, auf dem der Client mit dem Access Point verbunden ist
- AP System Name
  Der Systemname des Access Points

- BSSID
  Die BSSID (Basic Service Set Identification) des Access Points.
- Connected Stations
  Anzahl der Clients, die mit dem Access Point verbunden sind.
- Bidirectional Status
  Zeigt an, ob gleichzeitig die Daten des Access Points mit aufgezeichnet werden.

Einstellungen

Die Tabelle enthält folgende Spalten:

- **Funkschnittstelle**
  Zeigt an, für welche WLAN-Schnittstelle die Angaben gelten. Da ein Client über eine WLAN-Schnittstelle verfügt, gibt es in dieser Tabelle immer nur eine Zeile für "WLAN 1".
- **Zeitintervall [ms]**
  Geben Sie das Zeitintervall zwischen der Erfassung von zwei Messwerten in Millisekunden an. Der erste Messwert wird erst nach Ablauf des eingestellten Zeitintervalls angezeigt.
- **Stichproben**
  Geben Sie an, wie viele Messungen durchgeführt werden sollen.
- **Unbegrenzt**
  Wenn Sie das Optionshäkchen aktivieren, ist die Anzahl der Messungen unbegrenzt. Das Feld "Stichproben" ist ausgegraut. Der Signalrekorder läuft so lange, bis er manuell gestoppt wird oder das Gerät umkonfiguriert wird.
  Diese Option können Sie erst ab einem Zeitintervall ≥ 100 Millisekunden auswählen.
  Wenn die Aufzeichnung über 10000 Messungen beinhaltet, werden in der csv-Datei und in der PDF-Datei immer die letzten 10000 Messungen aufgelistet.
- **Bidirektionale Aufzeichnung**
  Wenn Sie die Einstellung aktivieren, werden ab einem Zeitintervall ≥ 10 Millisekunden die Werte des Access Points aufgezeichnet.
- **Starten**
  Klicken Sie die Schaltfläche in dieser Spalte, um die Aufzeichnung des Nutzsignals zu starten.

  Hinweis
  - Wenn Sie eine neue Aufzeichnung starten, wird die vorherige Aufzeichnung überschrieben.
  - Wenn die Aufzeichnung weniger als 10 Minuten gedauert hat und noch nicht beendet war (z. B. wegen Neustart oder Stromausfall), werden die Messwerte gelöscht.

  Der Signalrekorder führt alle 10 Minuten eine automatische Speicherung durch. Nach einem Neustart enthält die Aufzeichnung alle Werte bis zur letzten Speicherung.
- **Stoppen**
  Klicken Sie die Schaltfläche in dieser Spalte, um die Aufzeichnung des Nutzsignals vorzeitig zu beenden. Wenn die vorgegebene Anzahl an Messungen durchgeführt wurde, endet die Aufzeichnung des Nutzsignals automatisch.
- **Angezeigte Stichproben**
  Wählen Sie aus, wie viele Messungen in der Grafik angezeigt werden sollen.

Anwendungshinweise

Beachten Sie folgende Anwendungshinweise, um aussagekräftige Messungen mit dem Signalrekorder zu erhalten:

- Stellen Sie am Access Point eine feste Datenrate ein.
- Wenn Sie iPCF aktiviert haben, stellen Sie für die Messungen eine möglichst geringe Zykluszeit im Access Point ein.
- Es muss sichergestellt werden, dass genügend Datenkommunikation während der Messung besteht, weil die Statistik die ankommenden Datentelegramme auswertet.
- Die Messstrecke sollte 2 bis 3 mal mit den gleichen Parametern abgefahren werden, um herauszufinden, ob Einschnitte in der Signalstärke immer an der gleichen Position vorkommen.
- Punktuell sollten Messungen in einer festen Position über einen längeren Zeitraum durchgeführt werden.

Hinweis

- Wenn Sie eine neue Aufzeichnung starten, wird die vorherige Aufzeichnung überschrieben.
- Wenn das Gerät neu startet, werden die Messwerte gelöscht.

Wechseln Sie zu einem der folgenden Menüpunkte, um das Ergebnis der Aufzeichnung abzurufen:

- System > Laden & Speichern > HTTP
  Klicken Sie die Schaltfläche "Speichern" in der Tabellenzeile "WLANSigRec", um die Datei "signal_recorder_SCALANCE_W700.zip" im Dateisystem des angeschlossenen PC zu speichern.
- System > Laden & Speichern > TFTP
  Ändern Sie bei Bedarf den Dateinamen "signal_recorder_SCALANCE_W700.zip" in der Tabellenzeile "WLANSigRec". Wählen Sie in der Tabellenzeile "WLANSigRec" in der Klappliste der letzten Spalte den Eintrag "Datei speichern" und klicken Sie die Schaltfläche "Einstellungen übernehmen".

Die heruntergeladene ZIP-Datei enthält zwei Dateien mit den Ergebnissen der Aufzeichnung:

- Eine PDF-Datei
- Eine CSV-Datei

Spektrumanalysator

Hinweis

Die Seite ist nur verfügbar

- im Access Point-Modus
- wenn eine Online-Verbindung zum Gerät besteht.

Die WLAN-Schnittstelle des Geräts muss aktiviert sein, andernfalls können die Frequenzbereiche nicht gescannt werden.

Technische Angaben

Der Frequenzbereich ist abhängig von der Konfiguration.

Parameter		Wert
Amplitudengenauigkeit	In 2,4 GHz	3 dBm
	In 5 GHz	7 dBm
Auflösebandbreite		330 KHz
Min. Signalstärke		-100 dBm
Max. Signalstärke		0 dBm
Analysezeit	Bei 40 MHz	120 ms
	Bei 20 MHz	95 ms
Aktualisierungszeit		1 s

Signale eines Frequenzbereichs darstellen

Mit dem Spektrumanalysator können Sie elektromagnetische Signale eines Frequenzbereichs erkennen und darstellen. Sie können die Stärke aller Signale messen, die sich in der Umgebung des Access Points befinden.

Hinweis

Wenn der Spektrumanalysator gestartet ist, werden auf beiden WLAN-Schnittstellen alle WLAN-Verbindungen getrennt. Der Access Point sendet dann auch keine Beacons.

Hinweis

Aktivieren Sie den Spektrumanalysator nicht, wenn das Gerät produktiv läuft. Die Performance des Geräts kann beeinflusst werden.

Hinweis

Die Funktionalität des Spektrumanalysators ersetzt nicht einen dedizierten Spektrumanalysator.

Beschreibung

Die Seite enthält folgende Grafiken:

Die untere x-Achse zeigt bei allen Grafiken die Kanäle um die gewählte Mittenfrequenz an, für die Messungen durchgeführt werden. Die obere x-Achse zeigt den Frequenzbereich an. Die Anzeige der y-Achse ist von der gewählten Grafik abhängig.

- Realtime
  Die y-Achse zeigt die Signalstärke in dBm an.
  Die Grafik zeigt die Stärke aller Signale, die der Access Point in seiner Umgebung im konfigurierten Frequenzbereich empfängt.
  Die rote Linie zeigt die maximalen Werte seit dem Start der Messung an. Die weiße Linie zeigt die aktuellen Werte an. Die grüne Linie zeigt die Durchschnittswerte an.
- Spectrogram
  Die y-Achse zeigt den zeitlichen Verlauf der Messwerte von aktuell (0 s) bis zu den Werten, die vor 500 s empfangen wurden.
  Die Grafik zeigt die Stärke aller Signale, die der Access Point in seiner Umgebung im konfigurierten Frequenzbereich empfängt.
  Die Farbgebung ist von der Einstellung bei "Farbschema" abhängig.
- Density Chart
  Die y-Achse zeigt die Signalstärke in dBm an.
  Die Grafik zeigt, wie häufig Signale mit einer bestimmten Stärke im konfigurierten Frequenzbereich vorkommen.
  Der Farbverlauf geht vom geringsten Wert (0 %) in Schwarz zum höchsten Wert (100 %) in Rot.

Die Seite enthält folgende Schaltflächen:

- erweitern
  Mit diesem Icon zeigen Sie auf der Seite nur einen Grafiktyp in groß an.
- vermindern
  Mit diesem Icon wechseln Sie zurück in die Ansicht mit allen drei Grafiktypen.
- Farbschema
  Mit diesem Icon wechseln Sie das Farbschema für den Grafiktyp "Spectrogram":
  - Der Farbverlauf geht vom geringsten Wert (-100 dBm) in Schwarz zum höchsten Wert (0 dBm) in Rot.
  - Der Farbverlauf geht vom geringsten Wert (-100 dBm) in Rot zum höchsten Wert (0 dBm) in Schwarz.
- Rücksetzen
  Mit diesem Icon setzen Sie die Maximal- und Durchschnittswerte des Grafiktyps "Realtime" zurück.

Die Tabelle enthält folgende Spalten:

- Funkschnittstelle
  Zeigt an, für welche WLAN-Schnittstelle die Angaben gelten.
- Status
  Zeigt den Status der Messung. Folgende Werte sind möglich:
  - Gestopped
    Die Messung wurde angehalten.
  - Gestarted
    Die Messung läuft.

- Frequenzband
  Legen Sie das Frequenzband fest.
- Mittenfrequenz
  Wählen Sie die Mittenfrequenz aus.
  Wie viele Kanäle angezeigt werden, hängt von der Einstellung der "HT-Kanal-Bandbreite" ab, siehe Seite "Schnittstellen > WLAN > AP".
- Starten
  Klicken Sie die Schaltfläche in dieser Spalte, um die Messung zu starten.

  Hinweis

  Wenn Sie eine neue Messung starten, wird die vorherige Messung überschrieben.

- Stoppen
  Klicken Sie die Schaltfläche in dieser Spalte, um die Messung zu beenden.

Sie können das Messergebnis mithilfe des Menüs "System > Laden & Speichern > WLANSpectrumAnalyzer" als CSV-Datei speichern.

Messergebnisse

CSV-Datei

Die CSV-Datei enthält Informationen zur Konfiguration des Geräts und Detailinformationen zu allen Einzelmessungen und gliedert sich in zwei Bereiche. Der erste Bereich enthält die konfigurierten Einstellungen:

- System Name
  Den Systemnamen des Access Points
- Device IP
  Die IP-Adresse des Geräts
- Device MAC
  Die MAC-Adresse des Geräts
- Recording Interval
  Das Zeitintervall, welches zwischen der Erfassung zweier Messwerte liegt.

Der zweite Bereich ist eine Tabelle. Die Tabelle enthält für jeden Messwert Folgendes:

- Sample
  Die laufende Nummer der Messung
- Timestamp
  Der Zeitstempel
- In den folgenden Spalten werden alle Frequenzen des ausgewählten Frequenzbandes angezeigt. Nur bei den Frequenzen, für die ein Wert gemessen wurde, sind die Zellen befüllt. Die Messwerte zeigen die Signalstärke in dBm an.

Remote Capture

Auf dieser Seiten aktivieren Sie an der Schnittstelle (Ethernet, WLAN) die Funktion "Remote Capture". Die Funktion ist für die Netzwerkdiagnose über einen angeschlossen PC, z. B. um Übertragungsfehlern auf die Spur zu können.

Die Funktion können Sie auch an mehreren Schnittstellen gleichzeitig aktivieren. Wenn die Funktion aktiviert ist, lässt sich die Schnittstelle in Wireshark einbinden. Wireshark zeichnet über einen Zeitraum den Datenverkehr auf, der über die Schnittstelle fließt. Danach können Sie aus dem Mitschnitt heraus den Inhalt der Frames ansehen oder nach bestimmten Inhalten filtern.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Die Schnittstelle, auf die sich der Eintrag bezieht.
- **Aktivieren**
  Aktivieren oder deaktivieren Sie die Funktion "Remote Capture". Defaultmäßig ist die Funktion deaktiviert.

  Hinweis

  Performance

  Aktivieren Sie die Funktion nur zu Diagnosezwecke. Der erhöhte Datenverkehr könnte die Performance des Geräts beinflussen.

Die Seite enthält folgende Felder:

- **WLAN Capture-Modus (Nur im Access Point-Modus)**
  Legen Sie den Aufzeichnungsmodus für die WLAN-Schnittstelle fest:
  - Eigener Datenverkehr
    In diesem Fall werden die Frames aufgezeichnet, die vom Gerät empfangen und gesendet wurden.
    Ausnahme: Nicht angezeigt werden Datenpakete, die direkt von der Hardware behandelt werden, z. B. Hardware-Wiederholungen, Acknowledgement-Frames.
  - Gesamter Datenverkehr
    Der Access Point sendet keine Frames mehr, sondern zeichnet alle eingehenden Datenpakete auf.

    Hinweis

    Keine WLAN-Kommunikation zwischen Access Point und Clients

    Wenn die Einstellung "Gesamter Datenverkehr" verwendet wird, ist der Access Point für andere Teilnehmer nicht mehr erreichbar und verliert gegebenenfalls die verbundenen Clients.

- **Nach dem Systemstart aktivieren**
  - Deaktiviert
    Nach einem Neustart wird die Konfiguration auf die Default-Einstellung zurückgesetzt.
  - Aktiviert
    Die Konfiguration wird gespeichert und bleibt nach einem Neustart enthalten.

Schnittstelle in Wireshark einbinden

Voraussetzung:

- Wireshark V2.0.0 ist auf dem PC installiert.
- Der PC und das Gerät müssen über IP (Layer 3) erreichbar sein.

Vorgehensweise

Um den Datenverkehr z. B. der WLAN-Schnittstelle 1 in Wireshark zu analysieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie am Gerät an der WLAN-Schnittstelle die Funktion "Remote Caputre".
2. Wählen Sie bei Empfangsmodus "Eigener Datenverkehr" aus.
3. Klicken Sie auf "Einstellungen übernehmen", um die Funktion zu aktivieren.
4. Starten Sie Wireshark.
5. Klicken Sie im Menü "Aufzeichnen" auf "Optionen". Das Fenster "Wireshark - Mitschnittschnittstellen" wird geöffnet.
6. Klicken Sie auf dem Register "Eingabe" auf die Schaltfläche "Schnittstellen verwalten...". Klicken Sie im folgenden Dialog auf das Register "Entfernte Schnittstellen".
7. Um die Schnittstelle hinzuzufügen, klicken Sie im Register "Entfernte Schnittstellen" das Pluszeichen.
8. Geben Sie im folgenden Dialog bei "Host" die IPv4-Adresse des Geräts und bei "Port" 2002 ein.
9. Aktivieren Sie bei "Authentifizierung" "Keine Authentifizierung" und klicken Sie auf die Schaltfläche "OK".
10. Auf dem Register "Entfernte Schnittstellen" wird der Host und die Schnittstellen angezeigt, an denen zuvor die Funktion "Remote Capture" aktiviert wurde.
11. Wählen Sie die Schnittstelle aus und klicken Sie auf die Schaltfläche "OK".
12. Um die Aufzeichnung zu starten, klicken Sie auf "Starten". Weitere informationen zum Umgang mit dem Programm erhalten Sie bei Wireshark.

Wenn Sie mehrere Schnittstellen analysieren, können Sie für jede Schnittstelle eine Wireshark-Instanz verwenden.

Layer 2-Funktionen projektieren

VLAN

Allgemein

Auf dieser Seite legen Sie fest, ob das Gerät Telegramme mit VLAN-Tags transparent weiterleitet (IEEE 802.1D/VLAN-unaware-Modus) oder VLAN-Informationen berücksichtigt (IEEE 802.1Q/VLAN-aware-Modus). Wenn sich das Gerät im Modus "802.1Q VLAN Bridge" befindet, können Sie VLANs definieren und die Verwendung der Ports festlegen.

Hinweis

Ändern der Agent VLAN-ID

Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN-ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

Beschreibung

Die Seite enthält folgende Felder:

- **Base Bridge-Modus**
  Wählen Sie den gewünschten Modus aus. Folgende Modi sind möglich:

  Hinweis

  Base Bridge-Modus wechseln

  Beachten Sie den Abschnitt "Base Bridge-Modus wechseln". In diesem Abschnitt ist beschrieben, wie sich ein Wechsel auf die bestehende Konfiguration auswirkt.

  – 802.1 D Transparent Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLANTags nicht berücksichtigt bzw. verändert, sondern transparent weitergeleitet. Sie können in diesem Modus keine VLANs anlegen.

  – 802.1 Q VLAN Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLANTags nicht berücksichtigt bzw. verändert, sondern transparent weitergeleitet. Sie können in diesem Modus keine VLANs anlegen.
    Es ist nur ein Management-VLAN verfügbar: VLAN 1.

- **VLAN-ID**
  Tragen Sie die VLAN-ID (eine Zahl zwischen 1 und 4094) ein.
  Sie können bis zu 2 (SCALANCE W760), 8 (SCALANCE W770) oder 24 (SCALANCE W780) VLANS anlegen.

Die Tabelle gliedert sich in folgende Spalten:

- VLAN-ID
  Zeigt die VLAN ID an. Die VLAN ID (eine Zahl zwischen 1 und 4094) kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.
- Name
  Tragen Sie einen Namen für das VLAN ein. Der Name hat nur informativen Charakter und keine Auswirkungen auf die Konfiguration.
- Status
  Zeigt die Statusart des Eintrags in der internen Portfiltertabelle internen Portfiltertabellean. Dabei bedeutet statisch, dass die Adresse vom Anwender statisch eingetragen wurde. Die Angabe GVRP bedeutet, dass die Konfiguration über ein GVRP-Telegramm registriert wurde. Voraussetzung dafür ist allerdings, dass GVRP für das Gerät aktiviert wurde.
- Liste der Ports
  Legen Sie die Verwendung des Ports fest. Folgende Möglichkeiten gibt es:
  - "-"
  - Der Port ist kein Mitglied des angegebenen VLANs.
    Bei der Neudefinition sind alle Ports mit der Kennung "-" belegt.
  - M
    Der Port ist Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden mit dem entsprechenden VLAN-Tag weitergeleitet.
  - R
    Der Port ist Mitglied des VLANs. Die Registrierung erfolgt über ein GVRP-Telegramm.
  - U (Großbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet. Von diesem Port werden Telegramme ohne VLAN-Tag gesendet.
  - u (Kleinbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs, jedoch ist das VLAN nicht als Port-VLAN konfiguriert. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet.
  - F
    Der Port ist kein Mitglied des angegebenen VLANs und es ist nicht möglich, dass das VLAN dynamisch über GVRP an diesem Port registriert werden kann. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".
  - T
    Diese Option wird nur angezeigt und kann nicht ausgewählt werden. Dieser Port ist Trunk-Port und wurde dadurch Mitglied in allen VLANs. Sie konfigurieren diese Funktion im CLI (Command Line Interface) mit Hilfe des Befehls "`switchport mode trunk`" oder im WBM unter "Schnittstellen > Ethernet > Konfiguration".

Base Bridge-Modus wechseln

VLAN-unaware (802.1D Transparent Bridge) → VLAN-aware (802.1Q VLAN Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-unaware in VLAN-aware ädern, hat dies folgende Auswirkungen:

- Alle statischen und dynamischen Unicast-Einträe werden gelöcht.

VLAN-aware (802.1Q VLAN Bridge) → VLAN-unaware (802.1D Transparent Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-aware in VLAN-unaware ädern, hat dies folgende Auswirkungen:

- Alle VLAN-Konfigurationen werden gelöcht.
- Es wird ein Management-VLAN angelegt: VLAN 1.
- Alle statischen und dynamischen Unicast-Einträe werden gelöcht.

802.1Q VLAN Bridge: Wichtige Regeln für VLANs

Berücksichtigen Sie bei der Konfiguration und beim Betrieb Ihrer VLANs folgende Regeln:

- Telegramme mit der VLAN ID "0" werden wie ungetaggte Telegramme behandelt, behalten jedoch ihren Prioritäswert.
- Alle Ports am Gerä senden standardmäßg Telegramme ohne VLAN-Tag, um sicher zu gehen, dass der Endteilnehmer diese Telegramme empfangen kann.
- Die werkseitige Zuordnung der Ports finden Sie im Kapitel "VLAN".
- Wenn an einem Port ein Endteilnehmer angebunden ist, dann sollen ausgehende Telegramme ohne Tag versendet werden (statischer Zugriffs-Port). Wenn sich an dem Port ein weiterer Switch befindet, so ist das Telegramm mit einem Tag zu versehen (Trunk Port).

Port-basiertes VLAN

Verarbeitung empfangener Telegramme

Auf dieser Seite legen Sie die Konfiguration der Port-Eigenschaften für den Telegrammempfang fest.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Priorität / Port-VID / Erlaubte Telegrammtypen / Ingress Filterung**
  Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorität**
  Wählen Sie die Priorität aus, mit der ungetaggte Telegramme versehen werden.
  Die CoS-Priorität (Class of Service), die im VLAN-Tag verwendet wird. Wird ein Telegramm ohne Tag empfangen, wird ihm diese Priorität zugeordnet. Diese Priorität legt fest, wie dieses Telegramm im Vergleich zu anderen Telegrammen weiterhin bearbeitet wird. Es gibt insgesamt acht Prioritäten, mit den Werten 0 bis 7, wobei 7 der höchsten Priorität entspricht (IEEE 802.1p Port Priority).
- **Port-VID**
  Wählen Sie die VLAN-ID aus. Nur die VLAN-IDs sind wählbar, die Sie auf der Seite "VLAN > Allgemein" definiert haben.
  Wenn ein empfangenes Telegramm kein VLAN-Tag hat, so wird es um ein Tag mit der hier angegebenen VLAN-ID ergänzt und entsprechend den Regeln am Port gesendet.
- **Erlaubte Telegrammtypen**
  Legen Sie fest, welche Arten von Telegrammen akzeptiert werden. Es gibt folgende Alternativen:
  - Nur getraggte Frames
    Das Gerät verwirft alle ungetaggten Telegramme. Telegramme, die mit "0" getagged sind, werden wie ungetaggte Telegramme behandelt. Das Gerät leitet alle getaggten Telegramme weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration.
  - Nur ungetaggte und mit Priorität getaggte Frames
    Das Gerät verwirft alle getaggten Telegramme. Das Gerät leitet alle ungetaggten Telegramme sowie Telegramme mit einer Priorität (Priority Tagged Frames) weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration. Wenn Sie den Bridge-Modus "Provider" konfiguriert haben, bedeutet diese Einstellung für das Gerät, dass es alle eingehenden Telegramme als ungetaggte Telegramme behandelt.
  - Alle
    Das Gerät leitet alle Telegramme weiter
- **Ingress Filterung**
  Legen Sie fest, ob die VID von empfangenen Telegrammen ausgewertet wird. Folgende Möglichkeiten gibt es:
  - Aktiviert
    Die VLAN-ID empfangener Telegramme bestimmt die Weiterleitung: Für die Weiterleitung eines VLAN-getaggten Telegramms muss der Empfangsport Mitglied im gleichen VLAN sein. Am Empfangsport werden Telegramme aus unbekannten VLANs verworfen.
  - Deaktiviert
    Alle Telegramme werden weitergeleitet.

Dynamic MAC Aging

Protokolleinstellungen und Switch-Funktionalität

Das Gerät lernt automatisch die Quelladressen der angeschlossenen Teilnehmer. Diese Information wird dazu benutzt, um Datentelegramme gezielt an die betroffenen Teilnehmer weiterzuleiten. Die Netzlast für die anderen Teilnehmer wird reduziert.
Erhält ein Gerät innerhalb einer bestimmten Zeitspanne kein Telegramm, dessen Quelladresse mit einer gelernten Adresse übereinstimmt, dann löscht es die gelernte Adresse. Dieser Mechanismus wird als "Aging" bezeichnet. Durch Aging wird verhindert, dass Telegramme fehlgeleitet werden, wenn z.B. ein Endgerät (beispielsweise ein Programmiergerät) an einen anderen Port angeschlossen wird. Wenn die Option nicht aktiviert ist, löscht ein Gerät gelernte Adressen nicht automatisch.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Dynamic MAC Aging
  Aktivieren oder deaktivieren Sie die Funktion zum automatischen Aging von gelernten MAC-Adressen.
- Aging Time [s]
  Tragen Sie die Zeitspanne in Sekunden ein. Nach dieser Zeitspanne wird eine gelernte Adresse gelöscht, wenn das Gerät keine weiteren Telegramme von dieser Absenderadresse mehr empfängt. Wertebereich: 10 - 630 Sekunden

Spanning Tree

Allgemein

Allgemeine Einstellungen von MSTP

Dies ist die Basisseite zu Spanning Tree. Wählen Sie aus der Klappliste den Kompatibilitätsmodus aus. Standardmäßig ist Multiple Spanning Tree aktiviert.

In der jeweiligen Konfigurationsseite dieser Funktionen sind detailliertere Einstellungen möglich.

Hinweis

Client-Gerät nicht als Root-Bridge

Stellen Sie durch die Konfiguration von Prioritäten und Pfadkosten sicher, dass ein Client-Gerät nie zur Root-Bridge werden kann. Wenn ein Client-Gerät zur Root-Bridge wird, ist die Funktion von Rapid Spanning Tree nicht mehr gegeben.

Je nach Kompatibilitäts-Modus können Sie in der jeweiligen Konfigurationsseite die entsprechende Funktion konfigurieren.

Einstellungen

- **Spanning Tree**
  Aktivieren oder deaktivieren Sie Spanning Tree.

  Hinweis

  Clients oder Access Points im Client-Modus

  Wenn bei "MAC-Modus" "Automatisch" eingestellt ist, wird Spanning Tree nicht unterstützt.

- **Protokollkompatibilität**
  Wählen Sie den Kompatibiltätsmodus von Spanning Tree aus. Wenn Sie z. B. RSTP wählen, dann verhält sich Spanning Tree wie RSTP.
  Folgende Einstellungen gibt es:
  - STP
  - RSTP
  - MSTP

  Hinweis

  Wenn der iPCF-Modus aktiviert ist, werden nur die Kompatibilitäts-Modi STP und RSTP unterstützt.

CIST Allgemein

Die Seite besteht aus folgenden Teilen:

- Der linke Teil der Seite zeigt die Konfiguration des Geräts.
- Der mittlere Teil zeigt die Konfiguration der Root Bridge, wie sie aus Spanning Tree-Telegrammen abgeleitet werden kann, die ein Gerät empfangen hat.
- Der rechte Teil zeigt die Konfiguration der Regionalen Root Bridge, wie sie aus den MSTP-Telegrammen abgeleitet werden kann. Die angezeigten Daten sind nur dann sichtbar, wenn auf der Seite "Allgemein" "Spanning Tree" aktiviert und bei "Protokollkompatibilität" "MSTP" eingestellt ist. Das gilt auch für den Parameter "Bridge Max Hop Count". Wenn das Gerät eine Root Bridge ist, stimmen die Informationen des linken und des rechten Teils überein.

Einstellungen

- **Bridge-Priorität / Root-Priorität** (Nur Online verfügbar)
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root-Bridge wird. Die Bridge mit der höchsten Priorität wird Root-Bridge. Je kleiner der Wert, desto höher die Priorität. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, wird das Gerät Root-Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root-Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein.
- **Bridge-Adresse** / **Root-Adresse** (Nur Online verfügbar)
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.

- **Root-Port** (Nur Online verfügbar)
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten** (Nur Online verfügbar)
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge.
- **Topologieänderungen** / **Letzte Topologieänderung** (Nur Online verfügbar)
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit dem letzten Hochlauf. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz "sec" hinter der Zahlenangabe
  - Minuten: Zusatz "min" hinter der Zahlenangabe
  - Stunde: Zusatz "hr" hinter der Zahlenangabe
- **Bridge Hello Time / Root Hello Time** (Nur Online verfügbar)
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei solchen Telegrammen ist die Hello-Time. Der Standardwert für diesen Parameter beträgt 2 Sekunden.
- **Bridge Forward Delay [s] / Root Forward Delay [s]**
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort, sondern erst nach dem im Parameter festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.
- **Bridge Max Age / Root Max Age** (Nur Online verfügbar)
  Bridge Max Age definiert das maximale "Alter", welches eine empfangene BPDU haben darf, um vom Switch als gültig akzeptiert zu werden.

Die folgenden Werte sind nur im Access Point-Modus verfügbar:

- **Root-Priorität regional** (Nur Online verfügbar)
  Beschreibung siehe Bridge-Priorität / Root-Priorität
- **Root-Adresse regional** (Nur Online verfügbar)
  Die MAC-Adresse des Gerätes.
- **Root-Kosten regional** (Nur Online verfügbar)
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge.
- **Bridge Max Hop Count**
  Dieser Parameter gibt an, wie viele MSTP-Teilnehmer eine BPDU passieren darf. Wird eine MSTP-BPDU empfangen, deren "Bridge Max Hop Count" den hier konfigurierten Wert übersteigt, wird sie verworfen.
- **Name der Region**
  Tragen Sie den Namen der MSTP-Region ein, zu der dieses Gerät gehört. Defaultmäßig ist hier die MAC-Adresse des Gerätes eingetragen. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.
- **Version der Region**
  Tragen Sie die Versionsnummer der MSTP-Region ein, in der sich das Gerät befindet. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.

- **Layer-2 Tunnel Admin Edge Port** (Nur im Access Point-Modus verfügbar)
  Aktivieren Sie die Einstellung, wenn sich an einem Layer-2-Tunnel-Port ein Endgerät befinden kann. Andernfalls wird bei jeder Linkänderung an diesen Ports eine Rekonfiguration des Netzwerks ausgelöst. Die L2T-Clients sollten nicht miteinander verbunden sein.
- **Layer-2 Tunnel Auto Edge Port** (Nur im Access Point-Modus verfügbar)
  Aktivieren Sie die Einstellung, wenn an allen Layer-2-Tunnel-Ports automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist.

Zum Rücksetzen der Zähler auf dieser Seite steht Ihnen folgende Schalfläche zur Verfügung:

- **Zähler zurücksetzen** (Nur Online verfügbar)
  Über die Schaltfläche werden die Werte zurückgesetzt.

CIST Port

Konfiguration der MSTP-CIST-Ports

In der Tabelle wird beim Aufruf der Seite der aktuelle Stand der Konfiguration der Port-Parameter angezeigt.

Klicken Sie zur Konfiguration in die entsprechenden Felder der Port-Tabelle.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Spanning Tree-Status**
  Wählen Sie die Einstellung aus. Es gibt folgende Einstellmöglichkeiten:
  - Aktiviert
    Port ist im Spanning-Tree integriert.
  - Deaktiviert
    Port ist im Spanning-Tree nicht integriert.
  - Keine Änderung
    Tabelle 2 bleibt unverändert.
- In Tabelle übernehmen
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Spanning Tree-Status**
  Legen Sie fest, ob der Port im Spanning-Tree integriert ist oder nicht.

  Hinweis

  Wenn Sie die Einstellung "Spanning Tree-Status" für einen Port deaktivieren, kann es zur Schleifenbildung kommen. Die Topologie muss beachtet werden.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind. Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240. Der Standardwert ist 128.
- **kalk. Kosten**
  Tragen Sie die Wegekostenberechnung ein. Wenn Sie den Wert "0" eintragen, wird im Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.

- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn bei "erm. Kosten" der Wert "0" ist, so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "erm. Kosten" angezeigt. Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten. Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht parametriert werden. Der Parameter "Status" ist abhängig von dem projektierten Protokoll. Beim Status ist Folgendes möglich:
  - Disabled
    Der Port empfängt nur und nimmt nicht am STP, MSTP und RSTP teil.
  - Discarding
    In der Betriebsart "Discarding" werden BPDU-Telegramme empfangen. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    In diesem Status werden sowohl BPDU-Telegramme empfangen als auch gesendet. Der Port ist in den Spanning Tree-Algorithmus einbezogen.
  - Learning
    Vorstufe zum Weiterleitungsstatus, der Port lernt aktiv die Topologie (d. h. die Teilnehmeradressen).
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz, er empfängt und sendet Datentelegramme.

- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Wechsel vom Status "Discarding" zum Status "Forwarding" an
- **Edge-Typ**
  Legen Sie die Art des Edge-Port fest. Sie haben folgende Möglichkeiten:
  – "-"
    Edge-Port ist deaktiviert. Der Port wird wie ein "no-Edge-Port" behandelt.
  – Admin
    Wählen Sie diese Option, wenn sich an diesem Port immer ein Endgerät befindet. Sonst wird bei jeder Verbindungsänderung eine Rekonfiguration des Netzwerks ausgelöst.
  – Auto
    Wählen Sie diese Option, wenn an diesem Port automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist. Beim ersten Verbindungsaufbau wird der Port wie ein "no Edge Port" behandelt.
  – Admin/Auto
    Wählen Sie diese Optionen, wenn Sie an diesem Port eine Kombination aus beiden betreiben. Beim ersten Verbindungsaufbau wird der Port als Edge Port behandelt.
- **Edge** (Nur Online verfügbar)
  Zeigt an, in welchem Status der Port ist.
  – Enabled
    An diesem Port befindet sich ein Endgerät.
  – Disabled
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.

  Bei einem Endgerät kann ein Switch ohne Rücksicht auf Spanning Tree-Telegramme. schneller den Port umschalten. Wird entgegen dieser Einstellung ein Spanning Tree-Telegramm empfangen, wechselt der Port automatisch auf die Einstellung "Deaktiviert" für Switches.
- **P.t.P.-Typ**
  Wählen Sie die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.
  – "-"
    Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
  – P.t.P.
    Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
  – Shared Media
    Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

  Hinweis

  Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

- **Hello Time**
  Tragen Sie das Intervall ein, nach der die Bridge Konfigurationstelegramme (BPDUs) sendet. Standardmäßig sind 2 Sekunden eingestellt. Wertebereich: 1-2 Sekunden

  Hinweis

  Die portspezifische Einstellung der Hello Time ist nur mit der Protokollkompatibilität MSTP möglich. Wenn die Protokollkompatibilität RSTP eingestellt ist, wird der Parameter "Bridge Hello Time" auf der Seite "Layer 2 > Spanning Tree > CIST Allgemein" verwendet.

- **Eingeschränkte Rolle**
  Bei aktiviertem Optionskästchen wird der entsprechende Port nicht als Root-Port ausgewählt, unabhängig vom Wert für die Priorität. Auch der Port mit der niedrigsten Priorität wird bei aktiviertem Optionskästchen nicht zum Root-Port. Aktivieren Sie diese Option nur dann, wenn Sie den Einfluss von Bridges außerhalb des administrierten Bereichs auf die Spanning Tree-Topologie beschränken wollen.
- **Eingeschränktes TCN**
  Bei aktiviertem Optionskästchen leitet der entsprechende Port weder erhaltene noch erkannte Topologieänderungen (Topology Change Notifications) an andere Ports weiter. Aktivieren Sie diese Option nur dann, wenn Sie den Einfluss von Bridges außerhalb des administrierten Bereichs auf die Spanning Tree-Topologie beschränken wollen.
- **Limitiertes TCN**
  Bei aktiviertem Optionskästchen akzeptiert der entsprechende Port erhaltene und erkannte Topologieänderungen, leitet Topologieänderungen aber nicht an andere Ports weiter. Sie können die Optionskästchen in dieser Spalte nur dann aktivieren, wenn folgende Voraussetzungen erfüllt sind:
  - RSTP+ muss aktiviert sein.
  - Das Optionskästchen "Eingeschränktes TCN" muss für diesen Port deaktiviert sein.

  Wenn die genannten Voraussetzungen nicht erfüllt sind, werden die Optionskästchen in dieser Spalte gegraut dargestellt.

MST Allgemein

Multiple Spanning Tree-Konfiguration

Bei MSTP können zusätzlich zu RSTP mehrere VLANs in einem LAN mit eigenen RSTP-Bäumen verwaltet werden.

Beschreibung der angezeigten Felder

- **MSTP-Instanz-ID**
  Tragen Sie die Nummer der MSTP-Instanz ein. Zulässige Werte: 1 - 64

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die Zeile, die Sie löschen wollen.
- **MSTP-Instanz-ID**
  Zeigt die Nummer der MSTP-Instanz an.

- **Root-Adresse**
  Zeigt die MAC-Adresse der Root-Bridge an
- **Root-Priorität**
  Zeigt die Priorität der Root-Bridge an.
- **Bridge-Priorität**
  Tragen Sie in dieses Feld die Bridge-Priorität ein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 61440.
- **VLAN-ID**
  Tragen Sie die VlAN-ID ein. Sie können hier auch Bereiche mit Start-ID, "-", End-ID angeben. Mehrere Bereiche oder IDs werden durch "," separiert. Zulässige Werte: 1- 4094

MST Port

Konfiguration der Multiple Spanning Tree Port Parameter

Auf dieser Seite stellen Sie die Parameter für die Ports der konfigurierten Multiple Spanning Tree Instanzen ein.

Beschreibung der angezeigten Felder

Die Seite enthält folgendes Feld:

- **MSTP-Instanz-ID**
  Wählen Sie die ID der MSTP-Instanz aus.

Die Tabelle 1 gliedert sich in folgende Spalten

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports gültig ist.
- **MSTP-Status**
  Wählen Sie die Einstellung aus. Es gibt folgende Einstellmöglchkeiten:
  - Aktiviert
  - Deaktiviert
  - Keine Änderung: Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports und Link-Aggregationen an.
- **MSTP-Instanz-ID**
  ID der MSTP-Instanz.
- **MSTP-Status**
  Klicken Sie in das Optionskästchen, um diese Option zu aktivieren oder zu deaktivieren.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind. Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240. Werkseinstellung: 128
- **Kalk. Kosten**
  Tragen Sie in das Eingabefeld die Wege-Kostenberechnung ein. Wenn Sie hier 0 eintragen, wird im nächsten Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.
- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn bei "Kalk. Kosten" der Wert "0" ist, so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "Kalk. Kosten" angezeigt. Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten. Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht konfiguriert werden. Beim Status ist Folgendes möglich:
  - Discarding
    Der Port tauscht MSTP-Informationen aus, nimmt aber nicht am Datenverkehr teil.
  - Blocked
    Im Blocking-Modus werden BPDU-Telegramme empfangen.
  - Forwarding
    Der Port empfängt und sendet Datentelegramme.
- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Statuswechsel Discarding - Forwarding bzw. Forwarding - Discarding für einen Port an.

DCP-Weiterleitung

Anwendungen

Das DCP-Protokoll wird von STEP 7 und dem PST-Tool für die Konfiguration und Diagnose verwendet. In der Werkseinstellung ist DCP auf allen Ports aktiviert, d.h. empfangene DCP-Telegramme werden auf allen Ports weitergeleitet. Mit dieser Option haben Sie die Möglichkeit das Aussenden der Telegramme für einzelne Ports auszuschalten, um z.B. einzelne Netzbereiche von der Konfiguration per PST-Tool abzuschotten, bzw. um das gesamte Netz in kleinere Teilnetze für die Konfiguration und Diagnose zu unterteilen.

Hinweis

Leere Tabelle

Wenn Sie auf dem Gerät NAT aktiviert haben, ist die Tabelle leer bzw. wird geleert:

Einstellungen

- **Port**
  Zeigt die verfügbaren Ports an.
- **Einstellung**
  Legen Sie fest, ob der Port DCP-Telegramme ausgangsseitig blocken oder weiterleiten soll. Folgende Möglichkeiten gibt es:
  - block
    An diesem Port werden ausgangsseitig keine DCP-Telegramme weitergeleitet. Ein Empfangen ist jedoch über diesen Port weiterhin möglich.
  - forward
    An diesem Port werden DCP-Telegramme weitergeleitet.

LLDP

Bestimmung der Netzwerktopologie

LLDP (Link Layer Discovery Protocol) ist im Standard IEEE 802.AB definiert.

LLDP ist ein Verfahren zur Bestimmung der Netzwerktopologie. Netzwerkkomponenten tauschen über LLDP Informationen mit ihren Nachbargeräten aus.

Netzwerkkomponenten, die LLDP unterstützen, verfügen über einen LLDP-Agenten. Der LLDP-Agent versendet in periodischen Abständen Informationen über sich selbst und empfängt Informationen von angeschlossenen Geräten. Die empfangenen Informationen werden in der MIB gespeichert.

Anwendungen

PROFINET benutzt das LLDP-Protokoll für die Topologie-Diagnose. In der Werkseinstellung ist LLDP für alle Ports aktiviert, d. h. es werden LLDP-Telegramme auf allen Ports gesendet und empfangen. Mit dieser Funktion haben Sie die Möglichkeit das Aussenden und/oder Empfangen pro Port ein- oder auszuschalten.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Port**
  Zeigt den Port an.
- **Einstellung**
  Legen Sie die LLDP-Funktionalität fest. Folgende Möglichkeiten gibt es:
  - Tx
    Dieser Port kann LLDP-Telegramme nur senden.
  - Rx
    Dieser Port kann LLDP-Telegramme nur empfangen.
  - Rx & Tx
    Dieser Port kann LLDP-Telegramme empfangen und senden.
  - "-" (Deaktiviert)
    Dieser Port kann LLDP-Telegramme weder empfangen noch senden.

Layer 3-Funktionen für IPv4 projektieren

NAT

Basic

Hinweis

Die Seite ist nur verfügbar:

- SCALANCE W740, SCALANCE W730 und SCALANCE W722-1 RJ45
- Clients oder Access Points im Client-Modus.

Auf dieser Seite legen Sie die Grundeinstellungen für NAT fest.

Einstellungen

- **Schnittstelle**
  Wählen Sie die gewünschte Ethernet-Schnittstelle aus.
- **NAT aktivieren**
  Aktivieren oder deaktivieren Sie NAT für die Ethernet-Schnittstelle. Nicht aktivierbar, wenn am Client beim "MAC-Modus" "Layer-2-Tunnel" eingestellt ist.

- **TCP Idle Timeout [s]**
  Tragen Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die TCP-Verbindung aus der Übersetzungstabelle gelöscht.
  Der Wertebereich ist 1 bis 4294967295. Default-Einstellung: 86400 Sekunden.
- **UDP Idle Timeout [s]**
  Tragen Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die UDP-Verbindung aus der Übersetzungstabelle gelöscht.
  Der Wertebereich ist 1 bis 4294967295. Default-Einstellung: 300 Sekunden.
- **IP-Adresse der lokalen Schnittstelle**
  Tragen Sie die lokale IP-Adresse der Ethernet-Schnittstelle ein. Diese IP-Adresse ist die Gateway-Adresse des lokalen Gerätes.
- **Subnetzmaske der lokalen Schnittstelle**
  Tragen Sie die Subnetzmaske für das lokale Ethernet ein.
- **IPv6 Transparentmodus**
  Wenn aktiviert, wird der transparente Modus aktiviert. IPv6-Frames werden zwischen Ethernet und WLAN unverändert weitergeleitet. Vorausgesetzt, beim MAC-Modus ist nicht "Eigene" eingestellt und IPv6 ist ausgeschaltet. Wenn beim MAC-Modus "Manuell" eingestellt ist, müssen Sie die MAC-Adresse des IPv6-Geräts eingeben, das die IPv6-Telegramme empfängt bzw. versendet
- **IPv4- Multicast-Weiterleitung**
  Legen Sie fest, ob die eingehenden Multicast-Telegramme weitergeleitet werden.
  - Von der globalen zur lokalen Schnittstelle
    Die an der WLAN-Schnittstelle eingehenden Multicast-Telegramme werden über die Ethernet-Schnittstelle in das interne Netz weitergeleitet.
  - Von der lokalen zur globalen Schnittstelle
    Die an der lokalen Ethernet-Schnittstelle eingehenden Multicast-Telegramme werden über die WLAN-Schnittstelle in das externe Netz weitergeleitet.

NAPT

Hinweis

Die Seite ist nur verfügbar:

- SCALANCE W740, SCALANCE W730 und SCALANCE W722-1 RJ45
- Clients oder Access Points im Client-Modus.

Auf dieser Seite definieren Sie die Übersetzungsliste, für die Kommunikation vom globalen ins lokale Netz.

Einstellungen

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellungen beziehen. Nur auswählbar, wenn das Gerät mehrere Schnittstellen besitzt.
- **Traffic-Typ**
  Legen Sie fest, für welches Protokoll die Adresszuordnung gültig ist.
- **Globaler Port**
  Tragen Sie den globalen Port ein. Eingehende Telegramme mit diesem Port als Ziel-Port werden weitergeleitet.

  Hinweis

  Wenn der Port bereits durch einen lokalen Dienst z. B. Telnet belegt ist, wird eine Warnmeldung ausgegeben. Vermeiden Sie auf jeden Fall die Nutzung von TCP-Port 23 (Telnet), Port 22 (SSH), die Ports 80/443 (http/https: Erreichbarkeit des Clients mit dem WBM) und UDP-Port 161 (SNMP) als globalen Port.

- **Lokale IP-Adresse**
  Tragen Sie die IP-Adresse des Teilnehmers im lokalen Netz ein. An diese IP-Adresse wird das Telegramm weitergeleitet.
- **Lokaler Port**
  Tragen Sie die Nummer des Ports ein. Das ist der neue Ziel-Port, an den das eingehende Telegramm weitergeleitet wird. Sie können für mehrere Teilnehmer im lokalen Netz den gleichen Port angeben.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die zu löschenden Zeilen aus.
- **Schnittstelle**
  Zeigt die Schnittstelle an, auf die sich die Einstellungen beziehen.
- **Aktivieren**
  Wenn aktiviert, wird der Eintrag für die Adresszuordnung verwendet.
- **Traffic-Typ**
  Zeigt an, für welches Protokoll die Adresszuordnung gilt.
- **Dynamische globale IP-Adresse**
  Zeigt an, ob die dynamische Adressumsetzung verwendet wird.
- **Globale IP-Adresse**
  Zeigt die globale IP-Adresse an, auf die die lokale IP-Adresse umgesetzt wird.
- **Globaler Port**
  Zeigt den globalen Port an.
- **Lokale IP-Adresse**
  Zeigt die IP-Adresse des Teilnehmers im lokalen Netz an.
- **Lokaler Port**
  Zeigt die Nummer des lokalen Ports an.

Security-Funktionen projektieren

Passwörter

Passwörter

Konfiguration der Geräte-Passwörter

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Hinweis

Wenn Sie über einen RADIUS-Server angemeldet sind, können Sie keine lokalen Gerätepasswörter ändern.

Auf dieser Seite können Sie Passwörter ändern. Wenn Sie mit der Rolle "admin" angemeldet sind, können Sie die Passwörter für alle Benutzeraccounts ändern. Wenn Sie mit der Rolle "user" angemeldet sind, können Sie nur ihr eigenes Passwort ändern.

Hinweis

Wenn Sie sich das erste Mal oder nach einem "Auf Werkseinstellungen zurücksetzen und Neustart" mit dem voreingestellten Benutzer "admin" anmelden, werden Sie aufgefordert, das Passwort zu ändern.

Werksseitig sind die Passwörter bei Auslieferung des Geräts wie folgt eingestellt:

- admin: admin

Hinweis

Passwort ändern im Trial-Modus

Auch wenn Sie im Konfigurationsmodus "Trial" das Passwort ändern, wird diese Änderung sofort gespeichert.

Beschreibung der angezeigten Felder

- **Aktueller Benutzer**
  Zeigt den Benutzer an, der aktuell angemeldet ist.
- **Aktuelles Benutzerpasswort**
  Geben Sie das Passwort des aktuell angemeldeten Benutzers ein.
- **Benutzerkonto**
  Wählen Sie den Benutzer, dessen Passwort Sie ändern möchten.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie bei der Vergabe von neuen Passwörter verwendet wird.
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen
- **Neuer Name für das Admin-Benutzerkonto**
  Sie können einmalig den Namen des werksseitig voreingestellten Benutzers "admin" ändern. Danach ist ein Umbenennen von "admin" nicht mehr möglich.
- **Neues Passwort**
  Geben Sie das neue Passwort für den ausgewählten Benutzer ein. Es darf folgende Zeichen nicht enthalten:
  – § ? " ; : –
  Das Leerzeichen und das Zeichen für "Entfernen" düfen auch nicht enthalten sein.
- **Passwort bestätigen**
  Geben Sie das neue Passwort erneut ein, um es zu bestätigen.

Optionen

Auf dieser Seite legen Sie fest, welche Passwortrichtlinie bei der Vergabe von neuen Passwörter beachtet wird.

Beschreibung

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie akutell verwendet wird.
- **Neue Passwortrichtlinie**
  Wählen Sie aus der Klappliste die gewünschte Einstellung aus:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

Benutzer

Lokale Benutzer

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Lokale Benutzer

Auf dieser Seite erstellen Sie lokale Benutzer mit den entsprechenden Rechten.

Wenn Sie einen lokalen Benutzer anlegen oder löschen, wird diese Änderung automatisch auch in der Tabelle "Externe Benutzerkonten" durchgeführt. Wenn Sie eine Änderung explizit für die interne oder externe Benutzertabelle durchführen wollen, nutzen Sie die CLI-Befehle.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden

Beschreibung der angezeigten Felder

- **Benutzerkonto**
  Geben Sie den Namen für den Benutzer ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 255 Zeichen lang sein.
  - Er darf folgende Zeichen nicht enthalten: | ? " ; :
    Die Zeichen für Space und Delete dürfen auch nicht enthalten sein.

  Hinweis

  Benutzername nicht änderbar

  Nach dem Anlegen eines Benutzers kann der Benutzername nicht mehr geändert werden.

  Wenn ein Benutzername geändert werden soll, dann muss der Benutzer gelöscht und ein neuer Benutzer angelegt werden.

  Hinweis

  Werkseitig voreingestellter Benutzer "user"

  Ab der Firmware-Version 6.0 ist der werkseitig voreingestellte Benutzer "user" im Auslieferzustand nicht mehr verfügbar. Wenn Sie ein Gerät auf die Firmware V6.0 aktualisieren, ist der werkseitig voreingestellte Benutzer "user" zunächst noch verfügbar. Wenn Sie das Gerät auf die Werkseinstellungen zurücksetzen ("Auf Werkseinstellungen zurücksetzen und Neustart"), wird der werkseitig voreingestellten Benutzer "user" gelöscht. Sie können neue Benutzer mit der Rolle "user" anlegen.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie verwendet wird:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

  Sie konfigurieren die Passwortrichtlinie auf der Seite "Security > Passwörter > Optionen".
- **Passwort**
  Geben Sie das Passwort für den Benutzer ein. Die Stärke des Passworts ist abhängig von der eingestellten Passwortrichlinie.
- **Passwort bestätigen**
  Geben Sie das Passwort erneut ein, um es zu bestätigen.
- **Rolle**
  Wählen Sie eine Rolle aus.
  Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".

Die Tabelle enthält folgende Spalten:

Hinweis

Die voreingestellten Benutzer sowie angemeldete Benutzer können nicht gelöscht oder geändert werden.

- **Benutzeraccount**
  Zeigt den Benutzernamen an.
- **Rolle**
  Zeigt die Rolle des Benutzers an.
- **Beschreibung**
  Zeigt eine Beschreibung des Benutzekontos an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Rollen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Rollen

Auf dieser Seite erstellen Sie Rollen, die lokal auf dem Gerät gültig sind.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Einstellungen

- **Rollenname**
  Geben Sie den Namen für die Rolle ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

  Hinweis

  Rollenname nicht änderbar

  Nach dem Anlegen einer Rolle kann der Rollenname nicht mehr geändert werden.

  Wenn ein Rollenname geändert werden soll, dann muss die Rolle gelöscht und eine neue Rolle angelegt werden.

  Rollen löschen

  Die voreingestellten Rollen sowie zugewiesene Rollen können nicht gelöscht oder geändert werden

Die Tabelle enthält folgende Spalten:

- **Rolle**
  Zeigt den Namen der Rolle an.
- **Funktionsrecht**
  Wählen Sie die Funktionsrechte der Rolle aus:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen aber nicht verändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.

  Hinweis

  Funktionsrecht nicht änderbar

  Wenn Sie eine Rolle zugewiesen haben, können Sie das Funktionsrecht der Rolle nicht mehr ändern.

  Wenn Sie das Funktionsrecht einer Rolle ändern wollen, gehen Sie wie folgt vor:
  1. Löschen Sie alle zugewiesenen Benutzer.
  2. Ändern Sie das Funktionsrecht der Rolle.
  3. Weisen Sie die Rolle erneut zu.

- **Beschreibung**
  Geben Sie eine Beschreibung für die Rolle ein. Bei vordefinierten Rollen wird eine Beschreibung angezeigt. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Gruppen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Benutzergruppen

Auf dieser Seite verknüpfen Sie eine Gruppe mit einer Rolle.

In diesem Beispiel wird die Gruppe "Administrators" mit der Rolle "admin" verknüpft. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert und der Gruppe "Administrators" zuordnet, erhält dieser Benutzer auf dem Gerät die Rechte der Rolle "admin".

Einstellungen

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Der Name muss der Gruppe auf dem RADIUS-Server entsprechen.
  Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

Die Tabelle enthält folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an.
- **Rolle**
  Wählen Sie eine Rolle aus. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät.
  Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".
- **Beschreibung**
  Geben Sie eine Beschreibung für die die Verknüpfung der Gruppe mit einer Rolle an.

AAA

Benutzerverwaltung

Übersicht zur Benutzerverwaltung

Der Zugriff auf das Gerät wird durch konfigurierbare Benutzereinstellungen verwaltet. Richten Sie Benutzer mit jeweils einem Passwort zur Authentifizierung ein. Weisen Sie den Benutzern eine Rolle mit entsprechenden Rechten zu.

Die Authentifizierung von Benutzern kann entweder von dem Gerät lokal oder von einem externen RADIUS-Server durchgeführt werden. Wie die Authentifizierung erfolgen soll, konfigurieren Sie auf der Seite "Security > AAA > Allgemein".

Lokale Anmeldung

Die lokale Anmeldung von Benutzern durch das Gerät läuft wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät prüft, ob ein Eintrag für den Benutzer vorhanden ist:
   → Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet.
   → Wenn kein entsprechender Eintrag existiert, wird dem Benutzer der Zugriff verweigert.

Anmeldung über einen externen RADIUS-Server

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Authentifizierung und Autorisierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können.

Anhängig davon, welchen RADIUS-Autorisierungsmodus Sie auf der Seite "Security > AAA > RADIUS-Client" eingestellt haben, wertet das Gerät unterschiedliche Informationen des RADIUS-Servers aus.

RADIUS-Autorisierungsmodus "Standard"

Wenn Sie den RADIUS-Autorisierungsmodus "Standard" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück:
     → Der Benutzer wird mit Administratorrechten angemeldet.
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt einen anderen oder gar keinen Wert für das Attribut "Service Type" an das Gerät zurück:
     → Der Benutzer wird mit Leserechten angemeldet.
   - Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
     → Dem Benutzer wird der Zugriff verweigert.

RADIUS-Autorisierungsmodus "Herstellerspezifisch"

Voraussetzung

Für den RADIUS-Autorisierungsmodus "Herstellerspezifisch" ist am RADIUS-Server Folgendes einzustellen:

- Herstellercode: 4196
- Attributnummer: 1
- Attributformat: Zeichenfolge (Gruppenname)

Vorgehen

Wenn Sie den RADIUS-Autorisierungsmodus "Herstellerspezifisch" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   Fall A: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt die dem Benutzer zugeordnete Gruppe an das Gerät zurück.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird mit den Rechten der zugeordneten Gruppe angemeldet.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird der Rolle mit den größeren Rechten zugeordnet und mit diesen Rechten angemeldet.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle angemeldet, die mit seinem Benutzeraccount verknüpft ist.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall B: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung, gibt jedoch keine Gruppe an das Gerät zurück:
   - Der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der verknüpften Rolle angemeldet.
   - Der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall C: Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
   - Dem Benutzer wird der Zugriff verweigert.

Allgemein

Anmeldung von Netzteilnehmern

Die im Menü verwendete Bezeichnung "AAA" steht dabei für "Authentication, Authorization, Accounting". Dieses Feature dient dazu, Netzteilnehmer zu identifizieren und zuzulassen und ihnen die entsprechenden Dienste bereitzustellen.

Auf dieser Seite konfigurieren Sie die Anmeldung.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

Hinweis

Um den Login-Authentifizierung "RADIUS", "Lokal und RADIUS" oder "RADIUS und Fallback Lokal" nutzen zu können, muss ein RADIUS-Server hinterlegt und für die Benutzerauthentifizierung konfiguriert sein.

- **Login-Authentifizierung**
  Legen Sie fest, wie die Anmeldung erfolgt:
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen
  - Local und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen. Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS und Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen. Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.

RADIUS-Client

Authentifizierung über einen externen Server

Das Konzept von RADIUS basiert auf einem externen Authentifizierungsserver.

Jede Zeile der Tabelle enthält die Zugangsdaten für je einen Server. In der Suchreihenfolge wird der primäre Server zuerst angefragt. Ist der primäre Server nicht erreichbar, werden in der eingetragenen Reihenfolge sekundäre Server angefragt. Wenn keiner der Server antwortet, findet keine Authentifizierung statt.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **RADIUS-Autorisierungsmodus**
  Der RADIUS-Autorisierungsmodus legt bei der Login-Authentifizierung fest, wie bei einer erfolgreichen Authentifizierung die Rechtevergabe für die Benutzer erfolgt.
  - Standard
    In diesem Modus wird der Benutzer mit Administratorrechten angemeldet, wenn der Server für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück gibt. In allen anderen Fällen wird der Benutzer mit Leserechten angemeldet.
  - Herstellerspezifisch
    In diesem Modus ist die Rechtevergabe davon abhängig, ob und welche Gruppe der Server für den Benutzer zurück gibt und ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzeraccounts" gibt.

Die Tabelle gliedert sich in folgende Spalten:

- **Auth.-Servertyp**
  Wählen Sie aus, für welche Authentifizierungsverfahren der Server verwendet werden soll.
  - Login
    Der Server wird nur für die Login-Authentifizierung verwendet.
  - 802.1X
    Der Server wird nur für die 802.1X-Authentifizierung verwendet.
  - Login & 802.1X
    Der Server wird für beide Authentifizierungsverfahren verwendet.
- **Adresse des RADIUS-Servers**
  Tragen Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des RADIUS Servers ein.
- **Server-Port**
  Tragen Sie den Eingangs-Port auf dem RADIUS-Server ein. Standardmäßig ist der Eingangs-Port 1812 eingestellt. Der Wertebereich ist 1 ... 65535.
- **Shared Secret**
  Tragen Sie Ihre Zugangskennung ein. Der Wertebereich ist 1 ... 128 Zeichen.
- **Shared Secret bestätigen**
  Tragen Sie die Zugangskennung zur Bestätigung erneut ein.
- **Max. Retrans**
  Geben Sie hier die maximale Anzahl der Wiederholungen eines Anfrageversuchs ein. Der initiale Verbindungsversuch wird um den hier angegebenen Wert wiederholt, bevor ein anderer konfigurierter RADIUS-Server angefragt wird oder die Anmeldung für gescheitert erklärt wird. Standardmäßig sind 3 Wiederholungen eingestellt, das bedeutet 4 Verbindungsversuche. Der Wertebereich ist 1 ... 5.
- **Primärer Server**
  Legen Sie fest, ob dieser Server der primäre Server ist. Sie können aus den Optionen "ja" oder "nein" auswählen.

- **Test**
  Mit dieser Schaltfläche können Sie testen, ob der angegebene RADIUS-Server verfügbar ist oder nicht. Der Test wird einmalig durchgeführt und nicht zyklisch wiederholt.
- **Testergebnis**
  Zeigt an, ob der RADIUS-Server verfügbar ist oder nicht:
  - Fehlgeschlagen, keine Testpakete gesendet
    Die IP-Adresse ist nicht erreichbar.
    Die IP-Adresse ist erreichbar, der RADIUS-Server läuft jedoch nicht.
  - Erreichbar, das Shared Secret wurde nicht akzeptiert
    Die IP-Adresse ist erreichbar, der RADIUS-Server akzeptiert jedoch das angegebene Shared Secret nicht.
  - Erreichbar, das Shared Secret wurde akzeptiert
    Die IP-Adresse ist erreichbar und der RADIUS-Server akzeptiert das angegebene Shared Secret.

  Das Testergebnis wird nicht automatisch aktualisiert. Um das Testergebnis zu löschen, klicken Sie auf die Schaltfläche "Aktualisieren".

WLAN

Basic (Access Point)

Sicherheitsstufen

Um das Netzwerk zu sichern, werden Authentifizierung und Verschlüsselung verwendet. Auf dieser Seite legen Sie Sicherheitseinstellungen fest.

Hinweis

Übertragungsstandard IEEE 802.11 n

Der Übertragungsstandard IEEE 802.11 n mit der Einstellung "802.11n" oder "802.11 n only" unterstützt bei den Sicherheitseinstellungen nur WPA2 / WPA2-PSK mit AES.

iPCF, iPCF-HT oder iPCF-MC-Modus aktiviert

Wenn iPCF, iPCF-HT oder iPCF-MC aktiviert ist, wird bei den Sicherheitseinstellungen nur "Open System" mit den Verschlüsselungsverfahren WEP oder AES unterstützt.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die verfügbaren Schnittstellen an.
- **Authentifizierungstyp**
  Wählen Sie die Authentifizierungsart aus. Die Auswahl ist abhängig von der Betriebsart und dem Übertragungsstandard.
  - **Open System**
    Es wird keine Authentifizierung vorgenommen. Eine Verschlüsselung mit einem festen (nicht wechselnden) WEP-Schlüssel kann optional ausgewählt werden. Um den Schlüssel zu verwenden, aktivieren Sie "Verschlüsselung". Den WEP-Schlüssel definieren auf der Seite "Keys".
    Wenn der iPCF-Modus aktiviert ist, wird zusätzlich das Verschlüsselungsverfahren AES mit 128 Bit-Schlüssellänge unterstützt.
  - **Shared Key**
    Bei dieser Authentifizierung wird am Client und am Access Point ein fester Schlüssel hinterlegt. Dieser WEP-Schlüssel wird dann zur Authentifizierung und Verschlüsselung verwendet. Den WEP-Schlüssel definieren auf der Seite "Keys".

    Hinweis

    Bei Verwendung von "Open System" mit "Verschlüsselung" bzw. "Shared Key" muss immer Key 1 auf der Seite "Keys" gesetzt sein.

  - **WPA (RADIUS)**
    Wi-Fi Protected Access (WPA) ist eine Methode, die durch die Wi-Fi Alliance spezifiziert wird, um die Sicherheitslücken von WEP zu schließen. Dabei ist die Authentifizierung durch einen Server (802.1x) fest vorgeschrieben. Durch den dynamischen Austausch der Schlüssel bei jedem Datenframe wird eine weitere Sicherheit eingebaut.
  - **WPA-PSK**
    WPA Pre Shared Key (WPA-PSK) ist die abgeschwächte Version von WPA. Bei diesem Verfahren wird keine Authentifizierung durch einen Server durchgeführt, sondern anhand eines Passworts. Dieses Passwort wird auf dem Client wie auf dem Server manuell konfiguriert.
  - **WPA2 (RADIUS)**
    WPA2 (Wi-Fi Protected Access 2) ist die Weiterentwicklung von WPA und implementiert die Funktionen des Sicherheitsstandards IEEE 802.11i. Die WPA-Authentifizierung arbeitet jedoch mit dem RADIUS-Server.
  - **WPA2-PSK**
    WPA2-PSK basiert auf dem Standard 802.11i. Die WPA-Authentifizierung arbeitet jedoch ohne RADIUS-Server. Statt dessen wird auf jedem Client und Access Point ein WPA(2)-Schlüssel (WPA(2) Pass phrase) hinterlegt. Die WPA(2) Pass phrase wird zur Authentifizierung und weiteren Verschlüsselung verwendet.

- WPA/WPA2-Auto-PSK
  Einstellung, in der ein Access Point sowohl die Authentifizierungsart "WPA-PSK" als auch "WPA2-PSK" verarbeiten kann. Dies ist erforderlich, wenn der Access Point mit verschiedenen Clients kommuniziert, die einerseits "WPA-PSK" und andererseits "WPA2-PSK" verwenden. Bei den Clients wird das gleiche Verschlüsselungsverfahren eingesetzt.
- WPA/WPA2-Auto
  Einstellung, in der ein Access Point sowohl die Authentifizierungsart "WPA" als auch "WPA2" verarbeiten kann. Dies ist erforderlich, wenn der Access Point mit verschiedenen Clients kommuniziert, die einerseits "WPA" und andererseits "WPA2" verwenden. Bei den Clients wird das gleiche Verschlüsselungsverfahren eingesetzt

- Verschlüsselung
  Die Verschlüsselung schützt die übertragenen Daten vor Abhören und Fälschen. Sie können die Verschlüsselung nur abschalten, wenn Sie bei der Authentifizierung "Open System" gewählt haben. Alle anderen Sicherungsverfahren beinhalten sowohl Authentifizierung als auch Verschlüsselung.
- Verschlüsselungsverfahren
  Wählen Sie das Verschlüsselungsverfahren. Die Auswahl ist abhängig vom Übertragungsstandard.
  - AUTO
    Je nach Fähigkeit der Gegenstation wird entweder AES oder TKIP automatisch ausgewählt.
  - WEP
    WEP (Wired Equivalent Privacy)
    Ein symmetrisches Stromverschlüsselungsverfahren mit lediglich 40 bzw. 104 Bit langen Schlüsseln, die auf dem RC4-Algorithmus (Ron’s Code 4) basieren.
  - TKIP (Temporal Key Integrity Protocol)
    Ein symmetrisches Stromverschlüsselungsverfahren mit dem RC4-Algorithmus (Ron’s Code 4). TKIP verwendet im Gegensatz zur schwachen WEP-Verschlüsselung wechselnde Schlüssel, die von einem Hauptschlüssel abgeleitet werden. TKIP kann außerdem gefälschte Datentelegramme erkennen.
  - AES (Advanced Encryption Standard)
    Starkes symmetrisches Blockverschlüsselungsverfahren nach dem Rijndael-Algorithmus, der die Funktionen von TKIP weiter verbessert.

    Hinweis

    Um Ihre Daten besser vor Angriffen zu schützen, verwenden Sie WPA2/ WPA2-PSK mit AES.

- WPA(2)-Schlüssel
  Tragen Sie hier einen WPA(2)-Schlüssel ein. Dieser WPA(2)-Schlüssel muss sowohl auf der Client-Seite als auch dem Access Point bekannt sein und wird vom Benutzer auf beiden Seiten eingegeben.
  Für den WPA(2)-Schlüssel wird der ASCII-Code 0x20 bis 0x7e verwendet.

- **WPA(2)-Schlüssel bestätigen**
  Bestätigen Sie den oben eingegebenen WPA(2)-Schlüssel.
- **Default-Schlüssel**
  Legen Sie den WEP-Schlüssel fest, der zur Verschlüsselung der Daten verwendet wird. Den WEP-Schlüssel definieren auf der Seite "Keys".

Möglichkeiten

Welche Einstellungen möglich sind, hängt ab vom eingestellten "Authentifizierungs-Typ".

Authentifizierungs-Typ	Verschlüsselung	Verschlüsselungs- verfahren	Schlüsselherkunft
Open System	deaktiviert	--	--
Open System	aktiviert	WEP	Default-Schlüssel
Open System	aktiviert	AES[1)]	Default-Schlüssel (128-Bit)
Shared Key	aktiviert	WEP	Default-Schlüssel
WPA (RADIUS)	aktiviert	Auto/TKIP/AES	RADIUS-Server
WPA-PSK	aktiviert	Auto/TKIP/AES	WPA(2)-Schlüssel
WPA2 (RADIUS)	aktiviert	Auto/TKIP/AES	RADIUS-Server
WPA2-PSK	aktiviert	Auto/TKIP/AES	WPA(2)-Schlüssel
WPA/WPA2-AutoPSK	aktiviert	Auto/TKIP/AES	WPA(2)-Schlüssel
WPA/WPA2-Auto (RADIUS)	aktiviert	Auto/TKIP/AES	RADIUS-Server

[1)] nur bei iPCF, iPCF-HT oder iPCF-MC verfügbar

Basic (Client)

Sicherheitsstufen

Um das Netzwerk zu sichern, werden Authentifizierung und Verschlüsselung verwendet. Auf dieser Seite legen Sie Sicherheitseinstellungen fest.

Hinweis

Übertragungsstandard IEEE 802.11 n

Der Übertragungsstandard IEEE 802.11 n mit der Einstellung "802.11n" oder "802.11 n only" unterstützt bei den Sicherheitseinstellungen nur WPA2 / WPA2-PSK mit AES.

iPCF, iPCF-HT oder iPCF-MC-Modus aktiviert

Wenn der iPCF-, iPCF-HT- oder iPCF-MC-Modus aktiviert ist, wird bei dem Security Context 1 bei den Sicherheitseinstellungen nur "Open System" mit dem Verschlüsselungsverfahren AES unterstützt..

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- Security-Kontext
  Zeigt die Nummer des Eintrags an. Wenn Sie einen neuen Eintrag anlegen, wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
  Sie können bis zu 8 Security-Kontexte anlegen. Der Security-Kontext 1 kann nicht gelöscht werden.
- Authentifizierungtyp
  Wählen Sie die Authentifizierungsart aus. Die Auswahl ist abhängig von der Betriebsart und dem Übertragungsstandard.
  - Open System
    Es wird keine Authentifizierung vorgenommen. Eine Verschlüsselung mit einem festen (nicht wechselnden) WEP-Schlüssel kann optional ausgewählt werden. Um den Schlüssel zu verwenden, aktivieren Sie "Encryption". Den WEP-Schlüssel definieren auf der Seite "Keys".
    Wenn der iPCF- bzw. iPCF-MC-Modus aktiviert ist, wird nur das Verschlüsselungsverfahren AES mit 128 Bit-Schlüssellänge unterstützt.
  - Shared Key
    Bei der Shared Key Authentifizierung wird am Client und am Access Point ein fester Schlüssel hinterlegt. Dieser WEP-Schlüssel wird dann zur Authentifizierung und Verschlüsselung verwendet. Den WEP-Schlüssel definieren auf der Seite "Keys".
  - WPA (RADIUS)
    Wi-Fi Protected Access (WPA) ist eine Methode, die durch die Wi-Fi Alliance spezifiziert wird, um die Sicherheitslücken von WEP zu schließen. Dabei ist die Authentifizierung durch einen Server (802.1x) fest vorgeschrieben. Durch den dynamischen Austausch der Schlüssel bei jedem Datenframe wird eine weitere Sicherheit eingebaut.

    Hinweis

    Nehmen Sie die entsprechenden RADIUS-Einstellungen zunächst auf der Seite "Security > WLAN > Client Radius Supplicant" vor.

  - WPA-PSK
    WPA Pre Shared Key (WPA-PSK) ist die abgeschwächte Version von WPA. Bei diesem Verfahren wird keine Authentifizierung durch einen Server durchgeführt, sondern anhand eines Passworts. Dieses Passwort wird auf dem Client wie auf dem Server manuell konfiguriert.
  - WPA2 (RADIUS)
    WPA2 (Wi-Fi Protected Access 2) ist die Weiterentwicklung von WPA und implementiert die Funktionen des Sicherheitsstandards IEEE 802.11i. Die WPA-Authentifizierung arbeitet jedoch mit dem RADIUS-Server.

    Hinweis

    Nehmen Sie die entsprechenden RADIUS-Einstellungen zunächst auf der Seite "Security > WLAN > Client Radius Supplicant" vor.

- WPA2-PSK
  WPA2-PSK basiert auf dem Standard 802.11i. Die WPA-Authentifizierung arbeitet jedoch ohne RADIUS-Server. Statt dessen wird auf jedem Client und Access Point ein WPA(2)-Schlüssel (WPA(2) Pass phrase) hinterlegt. Die WPA(2) Pass phrase wird zur Authentifizierung und weiteren Verschlüsselung verwendet.
- WPA/WPA2-Auto-PSK
  Einstellung, in der ein Access Point sowohl die Authentifizierungsart "WPA-PSK" als auch "WPA2-PSK" verarbeiten kann. Dies ist erforderlich, wenn der Access Point mit verschiedenen Clients kommuniziert, die einerseits "WPA-PSK" und andererseits "WPA2-PSK" verwenden. Bei den Clients wird das gleiche Verschlüsselungsverfahren eingesetzt.
- WPA/WPA2-Auto
  Einstellung, in der ein Access Point sowohl die Authentifizierungsart "WPA" als auch "WPA2" verarbeiten kann. Dies ist erforderlich, wenn der Access Point mit verschiedenen Clients kommuniziert, die einerseits "WPA" und andererseits "WPA2" verwenden. Bei den Clients wird das gleiche Verschlüsselungsverfahren eingesetzt

- Verschlüsselung
  Die Verschlüsselung schützt die übertragenen Daten vor Abhören und Fälschen. Sie können die Verschlüsselung nur abschalten, wenn Sie bei der Authentifizierung "Open System" gewählt haben. Alle anderen Sicherungsverfahren beinhalten sowohl Authentifizierung als auch Verschlüsselung.
- Verschlüsselungsverfahren
  Wählen Sie das Verschlüsselungsverfahren. Die Auswahl ist abhängig vom Übertragungsstandard.
  - AUTO
    Je nach Fähigkeit der Gegenstation wird entweder AES oder TKIP automatisch ausgewählt.
  - WEP
    WEP (Wired Equivalent Privacy)
    Ein symmetrisches Stromverschlüsselungsverfahren mit lediglich 40 bzw. 104 Bit langen Schlüsseln, die auf dem RC4-Algorithmus (Ron’s Code 4) basieren.
  - TKIP (Temporal Key Integrity Protocol)
    Ein symmetrisches Stromverschlüsselungsverfahren mit dem RC4-Algorithmus (Ron’s Code 4). TKIP verwendet im Gegensatz zur schwachen WEP-Verschlüsselung wechselnde Schlüssel, die von einem Hauptschlüssel abgeleitet werden. TKIP kann außerdem gefälschte Datentelegramme erkennen.
  - AES (Advanced Encryption Standard)
    Starkes symmetrisches Blockverschlüsselungsverfahren nach dem Rijndael-Algorithmus, der die Funktionen von TKIP weiter verbessert.

  Hinweis

  Um Ihre Daten besser vor Angriffen zu schützen, verwenden Sie WPA2/ WPA2-PSK mit AES.

- **WPA(2)-Schlüssel**
  Tragen Sie hier einen WPA(2)-Schlüssel ein. Dieser WPA(2)-Schlüssel muss sowohl auf der Client-Seite als auch dem Access Point bekannt sein und wird vom Benutzer auf beiden Seiten eingegeben.
  Bei einem Schlüssel mit 8 bis 63 Zeichen können Sie nur folgende, lesbare ASCII-Zeichen verwenden: 0x20 - 0x7e.
  Bei einem Schlüssel mit genau 64 Zeichen können Sie folgende ASCII-Zeichen verwenden: 0 - 9, a - f und A - F.
- **WPA(2)-Schlüssel bestätigen**
  Bestätigen Sie den oben eingegebenen WPA(2)-Schlüssel.
- **Default-Schlüssel**
  Legen Sie den WEP-Schlüssel fest, der zur Verschlüsselung der Daten verwendet wird. Den WEP-Schlüssel definieren auf der Seite "Keys".

AP-Kommunikation

Kommunikationsmöglichkeiten

Auf dieser Seite legen Sie fest, welche Art von Kommunikation der Access Point erlaubt.

Hinweis

Die Seite ist nur im Access Point-Modus verfügbar.

Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **innerhalb des eigenen VAPs / mit anderen VAPs / mit dem Ethernet**
  Wählen Sie die gewünschte Einstellung. Wenn "No Change" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Zeigt die verfügbaren WLAN-Schnittstellen an.
- **Port**
  Zeigt die VAP-Schnittstelle an.

- **innerhalb des eigenen VAPs**
  - Aktiviert
    Clients, die an der gleichen VAP-Schnittstelle eines Access Point angemeldet sind, können miteinander kommunizieren.
  - Deaktiviert
    Option ist deaktiviert.
- **mit anderen VAPs**
  Nur verfügbar, wenn das Gerät mindestens 2 VAP-Schnittstellen hat.
  - Aktiviert
    Clients, die an unterschiedlichen VAP-Schnittstellen eines Access Point angemeldet sind, können miteinander kommunizieren.
  - Deaktiviert
    Option ist deaktiviert.

  Hinweis

  Bei einem Access Point muss auf allen WLAN-Schnittstellen beziehungsweise auf allen VAPs "mit anderen VAPs" aktiviert werden, um eine Kommunikation zwischen den Clients zu ermöglichen, die an verschiedenen VAP-Schnittstellen des Access Points angemeldet sind.

- **mit dem Ethernet**
  - Aktiviert
    Clients können über die Ethernet-Schnittstelle des Access Points kommunizieren.
  - Deaktiviert
    Option ist deaktiviert.

Client RADIUS Supplicant

Client Supplicant

Auf dieser Seite konfigurieren Sie die Einstellungen für die RADIUS-Autorisierung des Clients.

Hinweis

Die Seite ist nur verfügbar bei Clients oder Access Points im Client-Modus.

Hinweis

Dot1X-Benutzername und Dot1X-Benutzerpasswort

Bei WPA (RADIUS), WPA2 (RADIUS), EAP-TLS, EAP-TTLS und PEAP muss der Dot1X-Benutzername und das Dot1X-Benutzerpasswort konfiguriert werden. Bei der Einstellung "Auto" muss entweder das Zertifikat geladen sein oder der Dot1X-Benutzername und das Dot1X-Benutzerpasswort konfiguriert werden.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Security-Kontext**
  Zeigt den Security-Kontext an.
- **Dot1X Benutzername**
  Tragen Sie den Benutzernamen ein, mit dem sich der Client am RADIUS-Server anmelden.
- **Dot1X Benutzerpasswort**
  Geben Sie für den oben genannten Benutzernamen das Passwort ein. Mit dieser Kombination meldet sich der Client beim RADIUS-Server an.
  Für die Passwort-Vergabe werden die ASCII-Zeichen 0x20 bis 0x7e verwendet.
- **Dot1X Benutzerpasswort bestätigen**
  Bestätigen Sie das Passwort.
- **Dot1X Serverzertifikat**
  Legen Sie fest, ob sich der RADIUS-Server mittels Zertifikat gegenüber dem Client ausweist.
- **Dot1X EAP-Typen**
  Legen Sie die Authentifizierungsverfahren fest. Folgende Verfahren gibt es:
  - Auto
    Client bietet RADIUS-Sever alle Methoden an.
  - EAP-TLS
    Extensible Authentication Protocol - Transport Layer Security
    Nutzt Zertifikate zur Authentifizierung.
  - EAP-TTLS
    Extensible Authentication Protocol - Tunnel Transport Layer Security
    Nach Aufbau des TLS Tunnels wird MS-CHAPv2 zur inneren Authentifizierung genutzt.
  - PEAP
    Protected Extensible Authentication Protocol
    Alternativer Protokollentwurf der IETF zu EAP-TTL.

AP RADIUS-Authenticator

Konfigurieren des RADIUS-Server

Auf dieser Seite definieren Sie die RADIUS-Server sowie die RADIUS-Authentifizierung des Access Points. Sie können die Daten für zwei RADIUS-Server eingeben.

Hinweis

Die Seite ist nur im Access Point-Modus verfügbar.

Einstellungen

- **Reauthentifizierungsmodus**
  Legen Sie fest, wer die Zeit bestimmen soll, bis die Clients zu einer neuen Reauthentifizierung gezwungen werden.
  - '-' Deaktiviert
    Reauthentifizierungsmodus ist deakviert.
  - Server
    Aktiviert die Zeitverwaltung auf dem Server.
  - Lokal
    Aktiviert die lokale Zeitverwaltung. Legen Sie bei "Reauthentication Interval" den Gültigkeitszeitraum fest.
- **Reauthentifizierungs-Intervall [s]**
  Bei der lokalen Zeitverwaltung geben Sie hier den Gültigkeitszeitraum für eine Authentifizierung in Sekunden an. Die minimale Zeitdauer beträgt 1 Minute (Eingabe 60), die maximale Zeitdauer 12 Stunden (Eingabe 43200). Der Standardwert ist eine Stunde (3.600 Sekunden).

Die Tabelle gliedert sich in folgende Spalten:

- **Server-IP-Adresse**
  Tragen Sie hier die IP-Adresse des RADIUS-Servers ein.
- **Server-Port**
  Tragen Sie hier den Eingangs-Port auf dem RADIUS-Server ein.
- **Shared Secret**
  Tragen Sie das Passwort des RADIUS-Servers ein.
  Für das Passwort wird der ASCII-Code 0x20 bis 0x7e verwendet.
- **Shared Secret bestätigen**
  Bestätigen Sie das Passwort.
- **Max. Sendewiederholungen**
  Tragen Sie die maximale Anzahl der Verbindungsversuche ein.
- **Primärer Server**
  Legen Sie fest, ob dieser Server der primäre Server ist.
  - ja: Primärer Server
  - nein: Backup-Server.
- **Status**
  Mit dieser Einstellung können Sie den RADIUS-Server aktivieren oder deaktivieren

Schlüssel

Festlegung des WEP-Schlüssels

Damit Sie bei den Authentifizierungsmethoden "Open System" und "Shared Key" die Verschlüsselung aktivieren können, müssen Sie zuerst mindestens einen Schlüssel in der Schlüsseltabelle eintragen.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Schlüssel 1 - 4**
  Tragen Sie den WEP-Schlüssel bzw. den AES-Schlüssel ein. Für den WEP-Schlüssel sind Zeichen des ASCII-Code von 0x20 bis 0x7E oder hexadezimale Zeichen von 0x00 bis 0xFF zulässig.
  Wenn der iPCF- bzw. iPCF-MC-Modus aktiviert ist, wird nur das Verschlüsselungsverfahren AES mit 128 Bit-Schlüssellänge unterstützt.
  Zwischen folgenden Schlüssellängen kann gewählt werden:
  - 5 bzw. 13 ASCII- oder 10 bzw. 26 hexadezimale Zeichen (40/104 Bit)
  - 16 ASCII- oder 32 hexadezimale Zeichen (128 Bit)

  Hinweis

  Die Eingabe der hexadezimalen Zeichen erfolgt ohne vorangestelltes "0x". Mit einem Hexadezimalzeichen werden vier Bit kodiert. Die Eingaben "ABCDE" (ASCII-Zeichen) und "4142434445" (Hexadezimalzeichen) sind also gleichwertig, weil das ASCII-Zeichen "A" den Hexadezimalcode "0x41" hat.

- **Schlüssel 1 - 4 bestätigen**
  Bestätigen Sie den WEP-Schlüssel.

MAC ACL

Regelkonfiguration

Einleitung

Auf dieser Seite legen Sie die Zugriffsregeln für die MAC-basierte Access Control-Liste fest. Über die MAC-basierte ACL können Sie festlegen, ob Telegramme bestimmter MAC-Adressen weitergeleitet oder verworfen

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-MAC-Adresse**
  Tragen Sie die Unicast MAC-Adresse der Quelle ein.
  Ziel-MAC-Adresse
  Tragen Sie die Unicast MAC-Adresse des Ziels ein.

- **Aktion**
  Wählen Sie das Verhalten. Folgendes ist möglich:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Ingress-Schnittstelle**
  Zeigt eine Liste aller Schnittstellen an, für die diese Regel gilt.
- **Egress-Schnittstelle**
  Zeigt eine Liste aller Schnittstellen, für die diese Regel gilt.

Hinweis

Eingabe der MAC-Adressen

Sie können die Zugriffsregeln nur für Unicast MAC-Adressen konfigurieren. Nur wenn Sie bei Quell- und/oder Ziel-MAC-Adresse die Adresse "00:00:00:00:00:00" eingeben, gilt die so erstellte Regel für alle Quell- bzw. Ziel-MAC-Adressen.

Ingress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welchen ACL-Regeln eingehende Telegramme an Schnittstellen gefiltert werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fes

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie in der Klappliste die gewünschte Schnittstelle aus. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät.
- **Regel hinzufügen**
  Wählen Sie die ACL-Regel aus, die der Schnittstelle zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regelkonfiguration" fest.
- **Hinzufügen**
  Um der Schnittstelle die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von der Schnittstelle zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- Regelreihenfolge
  Zeigt die Reihenfolge der ACL-Regeln an.
- Regelnummer
  Zeigt die Nummer der ACL-Regel an.
- Quell-MAC-Adresse
  Zeigt die Unicast MAC-Adresse der Quelle an.
- Ziel-MAC-Adresse
  Zeigt die Unicast MAC-Adresse des Ziels an.
- Aktion
  Zeigt das Verhalten:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

Egress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden. Die ACL-Regel legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- Schnittstelle
  Wählen Sie die gewünschte Schnittstelle aus. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät.
- Regel hinzufügen
  Wählen Sie die ACL-Regel aus, die der Schnittstelle zugewiesen wird.
- Hinzufügen
  Um der Schnittstelle die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- Regel entfernen
  Wählen Sie die zu löschende ACL-Regel aus.
- Entfernen
  Um die ACL-Regel von der Schnittstelle zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- Regelreihenfolge
  Zeigt die Reihenfolge der ACL-Regeln an.
- Regelnummer
  Zeigt die Nummer der ACL-Regel an.
- Quell-MAC-Adresse
  Zeigt die Unicast MAC-Adresse der Quelle an.
- Ziel-MAC-Adresse
  Zeigt die Unicast MAC-Adresse des Ziels an.
- Aktion
  Zeigt das Verhalten an
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

IP ACL

Regelkonfiguration

Einleitung

Auf dieser Seite legen Sie die Regeln für die IP-basierte Access Controll-Liste fest. Über die IP-basierte ACL können Sie festlegen, ob Telegramme bestimmter IPv4-Adressen weitergeleitet oder verworfen werden.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- Regelnummer
  Zeigt die Nummer der ACL-Regel an. Wenn Sie einen neuen Eintrag anlegen, wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- Quell-IP-Adresse
  Geben Sie die IPv4-Adresse der Quelle ein.
- Quell-Subnetzmaske
  Geben Sie die Subnetzmaske ein, in der sich die Quelle befindet.
- Ziel-IP-Adresse
  Tragen Sie die IP-Adresse des Ziels ein.
- Ziel-Subnetzmaske
  Geben Sie die Subnetzmaske ein, in der sich das Ziel befindet.

- **Aktion**
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Ingress-Schnittstelle**
  Zeigt eine Liste aller Schittstellen an, für die diese Regel gilt.
- **Egress-Schnittstelle**
  Zeigt eine Liste aller Schittstellen an, für die diese Regel gilt.

Hinweis

Subnetzmaske für einzelne Hosts

Wenn Sie die Regel für ein einzelnes System erstellen (eine IPv4-Adresse), geben Sie die Subnetzmaske "255.255.255.255" an.

Protokollkonfiguration

Auf dieser Seite legen Sie die Regeln für Protokolle fest.

Beschreibung der angezeigten Felder

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**
  Zeigt die Nummer der Protokollregel an. Beim Erstellen einer Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Protokoll**
  Wählen Sie das Protokoll aus, für das die Regel gilt.
- **Protokollnummer**
  Geben Sie eine Protokollnummer ein, um weitere Protokolle zu definieren. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "Other Protocol" eingestellt haben.
- **Quell-Port Min.**
  Geben Sie die kleinstmögliche Portnummer des Quell-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Quell-Port Max.**
  Geben Sie die größtmögliche Portnummer des Quell-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Ziel-Port Min.**
  Geben Sie die kleinstmögliche Portnummer des Ziel-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.

- **Ziel-Port Max.**
  Geben Sie die größtmögliche Portnummer des Ziel-Ports an. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "TCP" oder "UDP" eingestellt haben.
- **Nachrichtentyp**
  Geben Sie einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.
- **Nachrichtencode**
  Geben Sie einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren. Dieses Feld ist nur editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.
- **DSCP**
  Geben Sie einen Wert zur Klassifizierung der Priorität ein. Dieses Feld ist nicht editierbar, wenn Sie bei Protokoll "ICMP" eingestellt haben.

Ingress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welchen ACL-Regeln eingehende Telegramme von Schnittstellen behandelt werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Beschreibung der angezeigten Felder:

- Schnittstelle
  Wählen Sie den gewünschten Port aus: einen Port oder eine VLAN-Schnittstelle. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät. Um eine VLAN-Schnittstelle auswählen zu können, muss eine IP-Schnittstelle konfiguriert sein.

  Hinweis

  Wenn Sie eine VLAN-Schnittstelle verwenden, gilt die ACL-Regel an allen Ports, die zu dem VLAN gehören.

- **Regel hinzufügen**
  Wählen Sie die ACL-Regel aus, die dem Port zugewiesen wird.
- **Hinzufügen**
  Um dem Port die ACL-Regel fest zuzuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von dem Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- Regelreihenfolge
  Zeigt die Reihenfolge der ACL-Regeln an.
- Regelnummer
  Zeigt die Nummer der ACL-Regel an.
- Protokoll
  Zeigt das Protokoll an, für das die Regel gilt.
- Protokollnummer
  Zeigt die Protokollnummer an.
- Quell-IP-Adresse
  Zeigt Sie die IP-Adresse der Quelle an.
- Quell-Subnetzmaske
  Zeigt die Subnetzmaske an, in der sich die Quelle befindet.
- Ziel-IP-Adresse
  Zeigt die IP-Adresse des Ziels an.
- Ziel-Subnetzmaske
  Zeigt die Subnetzmaske des Ziels an..
- Aktion
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- Quell-Port Min.
  Zeigt die kleinstmögliche Portnummer des Quell-Ports an.
- Quell-Port Max.
  Zeigt die größtmögliche Portnummer des Quell-Ports an.
- Ziel-Port Min.
  Zeigt die kleinstmögliche Portnummer des Ziel-Ports an.
- Ziel-Port Max.
  Zeigt die größtmögliche Portnummer des Ziel-Ports an.
- Nachrichtentyp
  Zeigt einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen.
- Nachrichtencode
  Zeigt einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren.
- DSCP
  Zeigt den Wert zur Klassifizierung der Priorität an.

Egress-Regeln

Einleitung

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden. Die ACL-Regeln legen Sie in der Registerkarte "Regelkonfiguration" fest.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **Schnittstelle**
  Wählen Sie den gewünschten Port aus: einen Port oder eine VLAN-Schnittstelle. Die verfügbaren Schnittstellen sind abhängig von Ihrem Gerät. Um eine VLAN-Schnittstelle auswählen zu können, muss eine IP-Schnittstelle konfiguriert sein.

  Hinweis

  Wenn Sie eine VLAN-Schnittstelle verwenden, gilt die ACL-Regel an allen Ports, die zu dem VLAN gehören.

- **Regel hinzufügen**
  Wählen Sie die ACL-Regel, die dem Port zugewiesen wird.
- **Hinzufügen**
  Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**
  Wählen Sie die zu löschende ACL-Regel aus.
- **Entfernen**
  Um die ACL-Regel von dem Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Regelreihenfolge**
  Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**
  Zeigt die Nummer der ACL-Regel an.
- **Protokoll**
  Zeigt das Protokoll an, für das die Regel gilt.
- **Protokollnummer**
  Zeigt die Protokollnummer an.
- **Quell-IP-Adresse**
  Zeigt Sie die IP-Adresse der Quelle an.
- **Quell-Subnetzmaske**
  Zeigt die Subnetzmaske der Quelle an.
- **Ziel-IP-Adresse**
  Zeigt die IP-Adresse des Ziels an.

- **Ziel-Subnetzmaske**
  Zeigt die Subnetzmaske der Quelle an..
- **Aktion**
  Wählen Sie aus, ob das Telegramm weitergeleitet oder abgelehnt werden soll, wenn es der ACL-Regel entspricht:
  - Forward
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard
    Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Quell-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Quell-Ports an.
- **Quell-Port Max.**
  Zeigt die größtmögliche Portnummer des Quell-Ports an.
- **Ziel-Port Min.**
  Zeigt die kleinstmögliche Portnummer des Ziel-Ports an.
- **Ziel-Port Max.**
  Zeigt die größtmögliche Portnummer des Ziel-Ports an.
- **Nachrichtentyp**
  Zeigt einen Nachrichtentyp ein, um das Format der Nachricht zu bestimmen.
- **Nachrichtencode**
  Zeigt einen Nachrichtencode ein, um die Funktion der Nachricht zu spezifizieren.
- **DSCP**
  Zeigt den Wert zur Klassifizierung der Priorität an.

Management ACL

Auf dieser Seite können Sie die Sicherheit Ihres Geräts erhöhen. Um festzulegen, welche Station mit welcher IP-Adresse auf Ihr Gerät zugreifen darf, konfigurieren Sie die IP-Adresse oder auch ein ganzes Adress-Band.

Sie können einstellen, mit welchen Protokollen und über welche Ports die Station auf das Gerät zugreifen darf. Sie definieren, in welchem VLAN die Station liegen darf. Damit wird gewährleistet, dass nur bestimmte Stationen innerhalb eines VLANs Zugriff auf das Gerät haben.

Hinweis

Beachten Sie, eine fehlerhafte Konfiguration kann dazu führen, dass Sie nicht mehr auf das Gerät zugreifen können.

Einstellungen

- **Management ACL**
  Aktivieren oder deaktivieren Sie die Funktion.
- **IP-Adresse**
  Geben Sie die IPv4-Adresse oder die Netzadresse ein, für welche die Regel gelten soll. Wenn Sie die IP-Adresse 0.0.0.0 verwenden, gelten die Einstellungen für alle IP-Adressen.
- **Subnetzmaske/Präfixlänge**
  Tragen Sie die Subnetzmaske oder die Präfixlänge ein.
  Die Subnetzmaske 255.255.255.255 ist für eine bestimmte IPv4-Adresse. Möchten Sie ein Subnetz zulassen, tragen Sie z. B. für ein C-Subnetz 255.255.255.0 ein. Die Subnetzmaske 0.0.0.0 gilt für alle Subnetze.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die zu löschenden Zeilen aus.
- **Regelreihenfolge**
  Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **IP-Adresse**
  Zeigt die IP-Adresse an.
- **Subnetzmaske/Präfixlänge**
  Zeigt die Subnetzmaske oder die Präfixlänge an.
- **Zulässige VLANs**
  Geben Sie die Nummer des VLANs ein, in dem sich das Gerät befindet. Nur die Station kann auf das Gerät zugreifen, wenn es sich in diesem konfigurierten VLAN befindet. Bleibt dieses Eingabefeld leer, gibt es keine Einschränkung.
- **SNMP**
  Legen Sie fest, ob die die Station bzw. die IP-Adresse über das Protokoll SNMP auf das Gerät zugreift.
- **TELNET**
  Legen Sie fest, ob die die Station bzw. die IP-Adresse über das Protokoll TELNET auf das Gerät zugreift.
- **HTTP**
  Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll HTTP auf das Gerät zugreift.
- **HTTPS**
  Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll HTTPS auf das Gerät zugreift.
- **SSH**
  Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll SSH auf auf das Gerät zugreift.
- **Px**
  Legen Sie fest, ob die Station bzw. die IP-Adresse über diesen Port auf das Gerät zugreift..

- **WLAN1** (Nur im Client-Modus)
  Legen Sie fest, ob die Station bzw. die IP-Adresse über die WLAN-Schnittstelle auf das Gerät zugreift.
- **VAPX.Y** (nur im Access Point-Modus)
  Legen Sie fest, ob die Station bzw. die IP-Adresse über die VAP-Schnittstelle auf das Gerät zugreift.
- **WDSX.Y** (nur im Access Point-Modus)
  Legen Sie fest, ob die Station bzw. die IP-Adresse über die WDS-Schnittstelle auf auf das Gerät zugreift.

Inter AP Blocking

Basic

Hinweis

- Dieses Register ist nur verfügbar.
  - SCALANCE W780 / W770
  - im Access Point-Modus
- Diese WBM-Seite ist nur mit folgenden KEY-PLUGs konfigurierbar:
  - W780 iFeatures (MLFB 6GK5 907-8PA00)
  - W700 Security (MLFB 6GK5907-0PA00)

Wann sollte Inter AP Blocking eingesetzt werden?

Die Clients, die an einem Access Point verbunden sind, können im Normalfall mit allen Geräten des Layer 2-Netzes kommunizieren.

Mit Inter AP Blocking lässt sich die Kommunikation der Clients einschränken, die mit dem Access Point verbunden sind. Nur die Geräte sind für die Clients zugänglich, deren IP-Adressen unter "Zugelassene Addressen" im Access Point konfiguriert sind. Eine Kommunikation mit anderen sich im Netz befindlichen Teilnehmern wird damit unterbunden.

Einstellungen

- **Aktualisierungsintervall [s]**
  Tragen Sie das Aktualisierungsintervall für die ARP-Tabelle ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen.
- **Port**
  Angabe der VAP-Schnittstelle, auf die sich die Einstellungen beziehen.
- **SSID**
  Angabe der SSID, auf die sich die Einstellungen beziehen.

- **Aktiviert**
  Wenn aktiviert, wird die Zugangsbeschränkung verwendet. Welche Geräte für die Clients zugänglich sind, konfigurieren Sie unter "Security > Inter AP Blocking > Erlaube IP-Adressen".
- **Blockierte unaufgeforderte ARP-Pakete**
  Wenn aktiviert, werden unaufgeforderte ARP-Pakete nicht weitergeleitet.
- **Blockierte Nicht-IP-Pakete**
  Wenn aktiviert, erfolgt kein Austausch von Nicht-IP-Paketen, z. B. Layer 2-Paketen, zwischen dem Client und den Geräten, die im Access Points als zulässige Kommunikationspartner konfiguriert sind.

Zugelassene Adressen

Auf dieser Seite legen Sie fest, welche Geräte für die Clients zugänglich sind.

Hinweis

- Dieses Register ist nur verfügbar.
  - SCALANCE W780 / W770
  - im Access Point-Modus
- Diese WBM-Seite ist nur mit folgenden KEY-PLUGs konfigurierbar:
  - W780 iFeatures (MLFB 6GK5 907-8PA00)
  - W700 Security (MLFB 6GK5907-0PA00)

Einstellungen

- **Port**
  Wählen Sie den gewünschten Port.
- **IP-Adresse**
  Tragen Sie die IP-Adresse der Geräte ein, die für den Client zugänglich sind.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Wählen Sie die zu löschenden Zeilen aus.
- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen
- **Port**
  Angabe der VAP-Schnittstelle, auf die sich die Einstellungen beziehen
- **IP-Adresse**
  IP-Adresse der Geräte, die für den Client zugänglich sind. Bei Bedarf können Sie die IP-Adresse ändern.
- **IP-Adresse des Resolvers**
  Die IP-Adresse, mit der die zugelassene IP-Adresse aufgelöst wird. Der Eintrag ist notwendig, wenn sich die Management IP-Adresse in einem anderen Subnetz befindet. Wenn bei "IP-Adress-Auflösung" die IP-Adresse "0.0.0.0" konfiguriert wird, wird zum Auflösen die Management IP-Adresse verwendet.

iFeatures projektieren

iPCF

Hinweis

iPCF bei Clients oder Access Points im Client-Modus

- SCALANCE W740, SCALANCE W734-1 RJ45 und SCALANCE W722-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Client: W740 iFeatures (MLFB 6GK5 907-4PA00)

iPCF im Access Point-Modus

- SCALANCE W780 und SCALANCE W774-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)

Wann sollte iPCF eingesetzt werden?

Der Einsatz von iPCF ist besonders empfehlenswert, wenn bei einer hohen Teilnehmerzahl ein hoher Grad an Deterministik realisiert werden soll. Dies ist zum Beispiel bei PROFINET oder anderen zyklischen Protokollen nötig.

Hinweis

Einsatz von iPCF zusammen mit anderen iFeatures

iPCF und andere iFeatures (z.B. iPCF-MC, iPRP) sind nicht miteinander kompatibel und können nicht gleichzeitig bei einem Gerät eingesetzt werden.

Einstellungen

In beiden Modi enthält die Tabelle folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen.
- **iPCF aktivieren**
  Aktivieren oder deaktivieren Sie den iPCF-Modus. Für PROFINET Kommunikation empfehlen wir, den iPCF-Modus zu aktivieren. Durch die Aktivierung von iPCF werden die vom Access Point angebotenen Datenraten angepasst. Wir empfehlen dringend, die Standardeinstellung für die Datenraten (802.11 a/b/g = 12 Mbit/s und 802.11n = MCS 2) beizubehalten.

Im Access Point-Modus enthält die Tabelle zusätzlich folgende Spalten:

- **Protokollunterstützung**
  Legen Sie fest, welches Protokoll von der WLAN-Schnittstelle priorisiert behandelt wird.
  - PROFINET
    Wenn Sie PROFINET einstellen, darf sich hinter dem Client kein PROFINET-Controller befinden.
  - EtherNET/IP
    Wenn Sie EtherNet/IP einstellen, darf sich hinter dem Client kein Scanner befinden
  - Deaktiviert
    Die Funktion ist deaktiviert.
- **iPCF-Zykluszeit [ms]**
  Wählen Sie aus der Klappliste die gewünschte Zykluszeit.
  Folgende Punkte sind bei der Einstellung der Zykluszeit zu beachten. Sonst wird eventuell keine stabile Kommunikation aufgebaut.
  - In der Anlage gibt es nur einen Access Point, d.h. die Clients bewegen sich nur in einer Funkzelle. In diesem Fall werden Aktualisierungszeiten >= 16 ms unterstützt.
  - In der Anlage gibt es mehrere Access Points, die über verschiedene Kanäle kommunizieren. Die Clients bewegen sich zwischen den Access Points (Roaming). Wählen Sie in diesem Fall Aktualisierungszeiten >= 32 ms.

  Zusätzlich zu den oben genannten Richtwerten ist zu beachten, dass die kleinste einzustellende Zykluszeit nach der Formel "2 ms * max. Teilnehmerzahl" ermittelt wird.

- Scan-Modus
  Die gewählte Einstellung wirkt sich auf das Scanverhalten der angemeldeten Clients aus. Folgende Einstellungen gibt es:
  - Alle Kanäle
    Der Client scannt alle erlaubten Kanäle und wählt den Access Points mit der besten Signalstärke aus und verbindet sich mit diesem.
  - Nächster Kanal
    Der Client scannt den nächsten Kanal aus seiner erlaubten Kanalliste. Falls dort ein Access Point vorhanden ist, verbindet er sich mit diesem. Wenn er auf diesem Kanal keinen Access Point findet, scannt er den nächsten Kanal.
- Schwellenwert der Signalqualität
  Nur konfigurierbar, wenn bei "Scan-Modus "Nächster Kanal" eingestellt ist.
  Der Access Point gibt dem Client eine Signalqualität vor. Beim Scannen muss der Client das vom Access Point kommende Signal mindestens mit der vorgegebenen Signalqualität empfangen. Nur dann wird eine Verbindung aufgebaut. Die Signalqualität wird vom Client auf Basis der RSSI-Werte (Received Signal Strength Indicator) empfangener Pakete ermittelt. Der RSSI-Wert gibt an, wie stark das ankommende Signal ist und wird im Signalrekorder angezeigt.
  Für die Signalstärke gibt es folgende Schwellenwerte:

Bereich	Signalqualität in %	Signalqualität in RSSI
1	40	20
2	50	25
3	60	30
4	70	35
5	80	40

iPCF-HT

Hinweis

iPCF bei Clients oder Access Points im Client-Modus

- SCALANCE W740, SCALANCE W734-1 RJ45 und SCALANCE W722-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Client: W740 iFeatures (MLFB 6GK5 907-4PA00)

iPCF im Access Point-Modus

- SCALANCE W780 und SCALANCE W774-1 RJ45
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)

Wann sollte iPCF-HT (High Throughput) eingesetzt werden?

Der Einsatz von iPCF ist besonders empfehlenswert, wenn bei einer hohen Teilnehmerzahl ein hoher Grad an Deterministik realisiert werden soll. Dies ist zum Beispiel bei PROFINET oder anderen zyklischen Protokollen nötig.

Hinweis

Einsatz von iPCF-HT

Die Funktion iPCF-HT

- und andere iFeatures (z. B. iPCF, iPCF-MC, iPRP) sind nicht miteinander kompatibel und können nicht gleichzeitig bei einem Gerät eingesetzt werden.
- ist nur im Frequenzband 5 GHz und mit WLAN-Modus "(Nur) IEEE 802.11n" nutzbar.
- steht nur auf der WLAN-Schnittstelle 1 zur Verfügung.

Es wird empfohlen nur einen MCS-Index zu verwenden.

Beschreibung

In beiden Modi enthält die Tabelle folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen.
- **iPCF-HT aktivieren**
  Aktivieren oder deaktivieren Sie den iPCF-HT. Wenn aktiviert, werden die vom Access Point angebotenen Datenraten angepasst. Wir empfehlen dringend, die Standardeinstellung für die Datenraten (802.11n = MCS 2) beizubehalten

Im Access Point-Modus enthält die Tabelle zusätzlich folgende Spalten:

- **Protokollunterstützung**
  Legen Sie fest, welches Protokoll von der WLAN-Schnittstelle priorisiert behandelt wird.
  - PROFINET
    Wenn Sie PROFINET einstellen, darf sich hinter dem Client kein PROFINET-Controller befinden.
  - EtherNet/IP
    Wenn Sie EtherNet/IP einstellen, darf sich hinter dem Client kein Scanner befinden.
  - Deaktiviert
    Die Funktion ist deaktiviert.
- **iPCF-HT-Zykluszeit [ms]**
  Legen Sie die Zykluszeit fest.
  Folgende Punkte sind bei der Einstellung der Zykluszeit zu beachten. Sonst wird eventuell keine stabile Kommunikation aufgebaut.
  - Der Wertebereich ist 16 - 512. Der eingestelte Wert sollte der Zykluszeit von Profinet bzw. EtherNet/IP entsprechen.
  - In der Anlage gibt es nur einen Access Point, d.h. die Clients bewegen sich nur in einer Funkzelle. In diesem Fall werden Aktualisierungszeiten >= 16 ms unterstützt.
  - In der Anlage gibt es mehrere Access Points, die über verschiedene Kanäle kommunizieren. Die Clients bewegen sich zwischen den Access Points (Roaming). Wählen Sie in diesem Fall Aktualisierungszeiten >= 32 ms.

  Zusätzlich zu den oben genannten Richtwerten ist zu beachten, dass die kleinste einzustellende Zykluszeit nach der Formel "4 ms * max. Teilnehmerzahl" ermittelt wird.

- **Scan-Modus**
  Die gewählte Einstellung wirkt sich auf das Scanverhalten der angemeldeten Clients aus. Folgende Einstellungen gibt es:
  - Alle Kanäle
    Der Client scannt alle erlaubten Kanäle und wählt den Access Points mit der besten Signalstärke aus und verbindet sich mit diesem.
  - Nächster Kanal
    Der Client scannt den nächsten Kanal aus seiner erlaubten Kanalliste. Falls dort ein Access Point vorhanden ist, verbindet er sich mit diesem. Wenn er auf diesem Kanal keinen Access Point findet, scannt er den nächsten Kanal.
- **Schwellenwert der Signalqualität**
  Nur konfigurierbar, wenn bei "Scan-Modus" "Nächster Kanal" eingestellt ist.
  Der Access Point gibt dem Client eine Signalqualität vor. Beim Scannen muss der Client das vom Access Point kommende Signal mindestens mit der vorgegebenen Signalqualität empfangen. Nur dann wird eine Verbindung aufgebaut.
  Die Signalqualität wird vom Client auf Basis der RSSI-Werte (Received Signal Strength Indicator) empfangener Pakete ermittelt. Der RSSI-Wert gibt an, wie stark das ankommende Signal ist und wird im Signalrekorder angezeigt.
  Für die Signalstärke gibt es folgende Schwellenwerte:

Bereich	Signalqualität in %	Signalqualität in RSSI
1	40	20
2	50	25
3	60	30
4	70	35
5	80	40

iPCF-MC

Voraussetzungen, um iPCF-MC ausführen zu können:

- Der Access Point verfügt über zwei WLAN-Schnittstellen (Dual AP).
- Access Point-Modus: Nur Dual APs mit KEY-PLUG W780 iFeatures (MLFB 6GK5 907-8PA00)
- Client-Modus: Client mit KEY-PLUG W740 iFeatures (MLFB 6GK5 907-4PA00)
- Die Management-Schnittstelle und die Daten-Schnittstelle müssen im gleichen Frequenzband und Mode betrieben werden und hinsichtlich ihrer Funkabdeckung übereinstimmen. iPCF-MC funktioniert nicht, wenn beide Funkschnittstellen mit Richtantennen ausgestattet sind, die unterschiedliche Bereiche abdecken.
- Die Management-Schnittstellen von allen Access Points, zwischen denen ein Client wechseln soll, müssen den gleichen Kanal verwenden. Ein Client scannt nur diesen einen Kanal, um erreichbare Access Points zu finden.
- Für die Management-Schnittstelle kann nicht das Übertragungsverfahren nach IEEE801.11h (DFS) verwendet werden. Für die Daten-Schnittstelle ist 801.11h (DFS) möglich.
- Der Client kann nicht mit "Nur zugelassene Kanäle verwenden" betrieben werden.

- "Roaming bei Verbindungsabbruch erzwingen" wird automatisch auf die zweite Schnittstelle gespiegelt.
- Für Clients gilt: Allen konfigurierten und aktiven SSIDs muss Security-Kontext 1 zugeordnet sein. Eine SSID ist aktiv, wenn auf der Seite "Schnittstellen > WLAN > Client" das dazugehörige Optionskästchen "Enabled" selektiert ist.
- In Japan kann iPCF-MC nicht aktiviert werden, wenn die Daten- oder Management-Schnittstelle eine Frequenz des 4920 MHz - 5080 MHz-Frequenzbandes nutzt.

Wann sollte iPCF-MC eingesetzt werden?

iPCF wurde entwickelt, um beim Wechsel zwischen Funkzellen ("Roaming") kurze Handover-Zeiten zu erreichen. Das Verfahren iPCF-MC ermöglicht kurze Handover-Zeiten auch bei frei beweglichen Clients und vielen Zellen bzw. einer großen Anzahl verwendeter Kanäle.

Hinweis

Einsatz von iPCF zusammen mit anderen iFeatures

Die Funktion iPCF-MC und andere iFeatures (z.B. iPCF, iPCF-HTMC, iPRP) sind nicht miteinander kompatibel und können nicht gleichzeitig bei einem Gerät eingesetzt werden.

Zuordnung der Schnittstellen

Beachten Sie, dass bei 11n-Geräten die Zuordnung der WLAN-Schnittstellen für iPCF-MC festgelegt ist.

- WLAN1: Daten-Schnittstelle
- WLAN2: Management-Schnittstelle

Einstellungen

Folgende Einstellung gibt es in beiden Modi

- **iPCF-MC aktivieren**
  Aktivieren oder deaktivieren Sie den iPCF-MC-Modus des Geräts.
  Für PROFINET-Kommunikation empfehlen wir, den iPCF-MC-Modus zu aktivieren. Durch die Aktivierung von iPCF-MC werden die vom Access Point angebotenen Datenraten angepasst.
  Wir empfehlen dringend, die Standardeinstellung für die Datenraten (802.11 a/b/g = 12 Mbit/s und 802.11n = MCS 2) beizubehalten.
- **iPCF-Zykluszeit** (nur im Access Point-Modus)
  Wählen Sie die Aktualisierungszeit aus, die für das Netzwerk projektiert ist, an das der Access Point angeschlossen ist. Der kleinste Wert für die Aktualisierungszeit beträgt 32 ms.

- **Protokollunterstützung** (nur im Access Point-Modus)
  Legen Sie fest, welches Protokoll priorisiert behandelt wird.
  - PROFINET
    Wenn Sie PROFINET einstellen, darf sich hinter dem Client kein PROFINET-Controller befinden.
  - EtherNET/IP
    Wenn Sie EtherNet/IP einstellen, darf sich hinter dem Client kein Scanner befinden.
- **Management Scan-Periode** (Nur im Client-Modus)
  Dieser Parameter legt fest, wie lang die Zeitdauer (angegeben in iPCF-Zyklen) zwischen zwei Background-Scans des Clients ist. Wenn Sie z. B. zwei auswählen, führt der Client nur bei jedem zweiten iPCF-Zyklus einen Background-Scan durch.
  Ein kleiner Wert für das Background-Scan-Intervall bildet die Grundlage für schnelles Roamen, jedoch kann damit kein hoher Datendurchsatz erreicht werden. Für einen hohen Datendurchsatz sollte ein größerer Wert gewählt werden.
- **Management Scan-Periode**
  Dieser Parameter legt fest, wie lang die Zeitdauer (angegeben in iPCF-Zyklen) zwischen zwei Background-Scans des Clients ist. Wenn Sie z. B. zwei auswählen, führt der Client nur bei jedem zweiten iPCF-Zyklus einen Background-Scan durch.
  Ein kleiner Wert für das Background-Scan-Intervall bildet die Grundlage für schnelles Roamen, jedoch kann damit kein hoher Datendurchsatz erreicht werden. Für einen hohen Datendurchsatz sollte ein größerer Wert gewählt werden.
- **Roaming-Filter** (nur im Client-Modus)
  Mit dieser Einstellung legen Sie die Anzahl der RSSI-Einzelmessungen fest, aus denen der Median ermittelt wird. Bei 5 werden die letzten 5 gemessenen RSSI-Werte betrachtet.
  - Median bei ungerader Anzahl an Messungen
    Die Werte werden aufsteigend geordnet. Der Wert, der sich genau in der Mitte befindet ist der Median.
  - Median bei gerader Anzahl an Messungen
    Die Werte werden aufsteigend geordnet. Der Median wird aus dem Durchschnitt der beiden mittleren Zahlen berechnet.

  Wenn es gelegentlich zu extremen Ausreißern bezüglich des ankommenden Signals kommt, können Sie mit diesem Roaming-Filter die gröbsten Schwankungen herausfiltern. Damit wird das vorzeitige Roaming des Clients verhindert.

iPRP

Hinweis

- SCALANCE W770/W740, SCALANCE W770/W730 und SCALANCE W722-1 RJ45

Die folgenden KEY-PLUGs sind gesteckt:

- Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)
- Client: W740 iFeatures (MLFB 6GK5 907-4PA00)

Wann sollte iPRP eingesetzt werden?

Hinweis

Einsatz von iPRP zusammen mit anderen iFeatures

iPRP und andere iFeatures (z. B. iPCF, iPCF-HT, iPCF-MC) sind nicht miteinander kompatibel und können nicht gleichzeitig bei einem Gerät eingesetzt werden.

iPRP mit Oversize-Frames (Jumbo-Frames)

Um Oversize-Frames nutzen zu können, muss bei sämtlichen Geräten im Netzwerk Oversize-Frames (Jumbo-Frames) konfiguriert sein.

Agent VLAN (Management-VLAN) bei iPRP

Als Agent-VLAN kann das jeweilige iPRP VLAN verwendet werden. Das ist davon abhängig wo sich das Gerät befindet.

- Wenn sich das Gerät im PRP-Netzwerk A oder PRP-Netzwerk B befindet, verwenden Sie als Agent VLAN das entsprechende VLAN, das PRPA oder PRPB zugeordnet ist.
- Wenn sich die Access Points in beiden PRP-Netzwerken befinden, können Sie als Agent VLAN eines der beiden VLANs verwenden. Alternativ können Sie auch andere VLANs als Agent-VLANs verwenden. Die Unterteilung in die PRP-Netzwerke A und B muss bestehen bleiben. Ein einziges Management VLAN für alle Geräte in Netzwerk A und B ist nicht ohne weiteres möglich

Mit "industrial Parallel Redundancy Protocol" (iPRP) kann die PRP-Technologie in drahtlosen Netzwerk genutzt werden. Bei iPRP werden die PRP-Frames parallel über zwei Funkstrecken übertragen. Dadurch können Störungen bei der Übertragung auf einer Funkstrecke durch die parallele Übertragung auf einer anderen kompensiert werden.

Voraussetzung

- Der Base Bridge-Modus "802.1Q VLAN Bridge" ist eingestellt
- Die VLANs sind angelegt.
- Access Point-Modus: Die VAP-Schnittstelle ist aktiviert.
- Client-Modus: Bei "MAC-Modus" ist "Layer 2-Tunnel" eingestellt.

Einstellungen

Die Seite enthält Folgendes:

- **PRP A**
  Wählen Sie in der Klappliste die VLAN-Zuordnung für PRP A aus.
- **PRP B**
  Wählen Sie in der Klappliste die VLAN-Zuordnung für PRP B aus.

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die verfügbaren Schnittstellen an.

  Hinweis

  Access Point

  Beim Access Point wird de Funktion iPRP nur an einer VAP-Schnittstelle unterstützt.

- **iPRP aktivieren**
  Aktivieren oder deaktivieren Sie iPRP für die gewünschte Schnittstelle.
- **PRP-Netzwerk**
  Legen Sie fest, in welchem PRP-Netzwerk die Schnittstelle Mitglied ist.
- **AP Funk-Redundanz** (Nur im Client-Modus)
  - Aktiviert
    Verhindert, dass sich die beiden Clients eines Client-Paares auf die gleiche WLAN-Schnittstelle des Access Points verbinden.
  - Deaktiviert
    Wenn der beste Access Point für einen Client der gleiche Access Point (gleiche WLAN-Schnittstelle) ist, wie der des Partner-Clients, wird geprüft, ob es einen weiteren Access Point gibt, dessen Signalstärke < 10dB schlechter als die des besten Access Points ist. In diesem Fall verbindet sich der Client mit diesem Access Point, ansonsten verbindet er sich mit dem gleichen, besten Access Point wie der Partner-Client.

iRef

Hinweis

- Die Seite ist nur im Access Point-Modus verfügbar.
- Die Funktion wird von SCALANCE W780 / W770 unterstützt.
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)

Wann sollte iREF eingesetzt werden?

Mit iREF können die Daten mit der hochstmöglichen Sendeleistung gesendet werden. Das ist insbesondere in Anwendungsfällen, bei denen MIMO nicht anwendbar, nicht von Vorteil ist oder bei der Verwendung von Sektoren-Antennen.

Hinweis

Einsatz von iREF zusammen mit anderen iFeatures

iREF und andere iFeatures (z.B. iPCF) sind nicht miteinander kompatibel und können nicht gleichzeitig bei einem Gerät eingesetzt werden.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen.
- **iREF aktivieren**
  Aktivieren oder deaktivieren Sie iREF für die gewünschte WLAN-Schnittstelle. Das Ergebnis wird unter "Information > WLAN >iFeatures" dargestellt.

Hinweis

Um iREF zu verwenden, müssen es mindestens zwei Antennen sein.

Die Antennen werden automatisch auf den Antennenmodus RX/TX umgestellt, sobald iREF eingeschaltet wird.

AeroScout

Hinweis

- Die Seite ist nur verfügbar
  - SCALANCE W780 / W770
  - im Access Point-Modus
- Die Einstellung ist nur mit dem folgenden KEY-PLUG konfigurierbar:
  - Access Point: W780 iFeatures (MLFB 6GK5 907-8PA00)

Hinweis

Die Funktion AeroScout ist nicht mit anderen iFeatures (iPCF,iPCF-MC, iREF) kombinierbar.

AeroScout kann ausschließlich im 2,4 GHz-Band entsprechend IEEE 802.11g, IEEE 802.11n und IEEE 802.11n-only verwendet werden.

Weiterführende Informationen entnehmen Sie bitte der Dokumentation der Firma AeroScout (www.aeroscout.com).

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Funkschnittstelle**
  Angabe der WLAN-Schnittstelle, auf die sich die Einstellungen beziehen.
- **AeroScout aktivieren**
  Aktivieren oder deaktivieren Sie die Aeroscout-Unterstützung für die gewünschte WLAN-Schnittstelle. Das Ergebnis wird unter "Informationen > WLAN iFeatures" dargestellt. Diese Seite ist nur im WBM verfügbar.

SCALANCE M projektieren

Unterstützte Geräte

Unterstützte Geräte

Die in diesem Hilfeabschnitt beschriebenen Funktionen werden von folgenden Produkten unterstützt:

- SCALANCE M874, M876 V5.0
- SCALANCE M812, M816 V5.0
- SCALANCE M826 V5.0

Allgemeine Benennung "Gerät"

Im vorliegenden Abschnitt des Informationssystems wird die Benennung "Gerät" stellvertretend für die oben genannten Produktnamen verwendet. Die Funktionsunterschiede finden Sie in Kapitel Hardware-Ausstattung und Systemfunktionen (Seite 2688). Abhängig vom ausgewählten Gerät sind auf den einzelnen Konfigurationsseiten unterschiedliche Parameter verfügbar.

Hardware-Ausstattung und Systemfunktionen

Verfügbarkeit der Hardware-Ausstattung

Die nachfolgende Tabelle zeigt die Hardware-Ausstattung der Geräte

Technische Änderungen sind vorbehalten.

	SCALANCE M874-2 SCALANCE M874-3	SCALANCE M876-3 SCALANCE M876-4	SCALANCE M812	SCALANCE M816	SCA-LANCE M826	SCALANCE M804PB
Schnitt-stellen	1 x SMA-An-tennen-an-schluss	2 x SMA-An-tennen-an-schluss	1 x DSL	1 x DSL	2 x SHDSL	1 x RS 485 (MPI/DP)
	2 x RJ45	4x RJ45	1x RJ45 (LAN)	4 x RJ45 (LAN)	4 x RJ45	2 x RJ45
Digitaler Ein-/Aus-gang	✓	✓	✓	✓	✓	✓
PLUG-Steckplatz	✓	✓	-	✓	✓	✓
SET-Tas-ter	✓	✓	✓	✓	✓	✓

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Geräten. Beachten Sie, dass in diesem Projektierungshandbuch und der Online-Hilfe alle Funktionen beschrieben sind. Abhängig dem KEY PLUG stehen Ihnen manche Funktionen nicht zur Verfügung.

Technische Änderungen sind vorbehalten.

		SCA-LANCE M87x	SCA-LANCE M812	SCA-LANCE M816	SCA-LANCE M826	SCA-LANCE M804PB
Basic Wizard	IP-Einstellungen	✓	✓	✓	✓	✓
	Geräteeinstellungen	✓	✓	✓	✓	✓
	SIM	✓	-	-	-	-
	Mobilfunkanbieter	✓	-	-	-	-
	DSL	-	✓	✓	-	-
	SHDSL	-	-	-	✓	-
	TIA Portal Cloud Connector	-	-	-	-	✓
	Zeiteinstellungen	✓	✓	✓	✓	✓
	SINEMA RC [1)]	✓	-	✓	✓	✓
	DDNS	✓	✓	✓	✓	✓
Informationen	ARP-Tabelle	✓	✓	✓	✓	✓
	Log-Tabellen	✓	✓	✓	✓	✓
	Mobilfunk	✓	-	-	-	-
	DSL	-	✓	✓	-	-
	SHDSL	✓	✓	✓	✓	-
	Redundanz	✓	✓	✓	✓	✓
	VRRPv3	✓	✓	✓	-	-
	SINEMA RC [1)]	✓	-	✓	✓	✓

		SCA-LANCE M87x	SCA-LANCE M812	SCA-LANCE M816	SCA-LANCE M826	SCA-LANCE M804PB
System	SMTP-Client	✓	✓	✓	✓	✓
	SNMP	✓	✓	✓	✓	✓
	Zeiteinstellung	✓	✓	✓	✓	✓
	Automatisches Abmelden	✓	✓	✓	✓	✓
	Syslog-Client	✓	✓	✓	✓	✓
	Fehlerkontrolle	✓	✓	✓	✓	✓
	PLUG	✓	-	✓	✓	✓
	SMS	✓	-	-	-	-
	DNS	✓	✓	✓	✓	✓
	DHCP-Client	-	-	-	✓	-
	DHCP-Server	✓	✓	✓	✓	✓
	cRSP/SRS	✓	✓	✓	✓	✓
	Proxy-Server	✓	✓	✓	✓	✓
	TIA Portal Cloud Connector	-	-	-	-	✓
	SINEMA RC[1)]	✓	-	✓	✓	✓
Schnittstellen	Ethernet	✓	✓	✓	✓	✓
	Mobilfunk	✓	-	-	-	-
	DSL	-	✓	✓	-	-
	SHDSL	-	-	-	✓	-
Layer 2	Konfiguration	✓	✓	✓	✓	✓
	VLAN	✓	✓	✓	✓	✓
	Dynamic MAC Aging	✓	✓	✓	✓	✓
	LLDP	✓	✓	✓	✓	✓
	Spanning Tree	✓	✓	✓	✓	✓
Layer 3	Statische Routen	✓	✓	✓	✓	✓
	Subnetze	✓	✓	✓	✓	✓
	Spanning Tree	✓	✓	✓	✓	✓
	NAT	✓	✓	✓	✓	✓
	VRRPv3	✓	✓	✓	-	-
Security	Passwörter	✓	✓	✓	✓	✓
	Benutzer	✓	✓	✓	✓	✓
	AAA (Authentication, Authorization, Accounting)	✓	✓	✓	✓	✓
	Zertifikate	✓	✓	✓	✓	✓
	Firewall	✓	✓	✓	✓	✓
	IPsec VPN	✓	✓	✓	✓	✓
	OpenVPN	✓	✓	✓	✓	✓

[1)] KEY-PLUG SINEMA Remote Connect 6GK5908-0PB00

Besonderheiten der Benutzerverwaltung für Security-Funktionen

Zugriff auf Security-Funktionen

Auf Security-Funktionen können Sie nur nach Aktivierung des Projektschutzes und nach erfolgreicher Anmeldung eines Benutzers zugreifen. Der angemeldete Benutzer muss zudem die erforderlichen Engineering- und Runtime-Rechte besitzen.

Grundlegende Informationen zur Benutzerverwaltung in STEP 7 finden Sie im Abschnitt "Benutzerverwaltung verwenden" des Informationssystems..

Engineering-Rechte für Security-Geräte

Diese Engineering-Rechte berechtigen zur Diagnose und Projektierung von Security-Geräten:

- Security: Security-Module mit Leserechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Geräte anzeigen..
- Security: Security-Module mit Schreibrechten: Die globalen Security-Funktionen sowie die lokalen Security-Einstellungen der Security-Geräte projektieren und die Diagnose der Security-Geräte durchführen.

Rollen mit Runtime-Rechten für SCALANCE M-800

Einigen systemdefinierten Rollen mit dem Präfix "NET" sind standardmäßig Runtime-Rechte für SCALANCE M-800 zugeordnet. Runtime-Rechte berechtigen zur Ausführung bestimmter Aufgaben auf den Geräten. Runtime-Rechte können Sie ebenso wie Engineering-Rechte nicht für systemdefinierte Rollen ändern. Für benutzerdefinierte Rollen können Sie die Runtime-Rechte beliebig ändern. Die Runtime-Rechte einer benutzerdefinierten Rolle legen Sie pro Gerät fest.

Es muss mindestens ein Benutzer vorhanden sein, dem eine Rolle mit vollständigen Runtime-Rechten für ein Security-Modul zugeordnet ist, z.B. die systemdefinierte Rolle "NET Administrator".

Den systemdefinierten Rollen mit dem Präfix "NET" sind folgende Engineering-Rechte zugeordnet:

Recht	NET Administrator	NET Standard	NET Diagnose	NET Remote Access	NET Administrator Radius	NET Radius
Projekt schreibgeschützt öffnen	x	x	x	-	-	-
Projekt mit Schreibrechten öffnen	x	x	x	-	-	-
Security-Einstellungen bearbeiten: "Einstellungen" und "Benutzer und Rollen"	x	-	-	-	-	-
Security: Security-Module mit Leserechten	x	x	x	-	x	x
Security: Security-Module mit Schreibrechten	x	x	-	-	x	-

Runtime-Rechte für SCALANCE M-800

Folgende Runtime-Rechte stehen zur Auswahl:

Recht	NET Administrator	NET Standard	NET Diagnose	NET Remote Access	NET Administrator Radius	NET Radius
Laden der Konfigurationsdateien	x	x	-	-	x	-
TIA-Portal: Diagnose des Security-Moduls durchführen	x	x	x	-	x	x

Zertifikate für Security-Funktionen

Bei der Projektierung können nur Zertifikate verwendet werden, die vom TIA Portal erzeugt wurden, beispielsweise VPN-Gruppen-Zertifikate. Nicht vom TIA Portal erzeugte Zertifikate, beispielsweise CA-Zertifikate für SINEMA Remote Connect, können nicht verwendet werden. Solche Zertifikate müssen Sie nachträglich über das WBM hinzufügen und auf das Gerät laden. Weiterführende Informationen finden Sie im zugehörigen WBM-Projektierungshandbuch.

Eigenschaften und Parameter bearbeiten

Möglichkeiten zum Bearbeiten

Zum Bearbeiten von Eigenschaften und Parameter gibt es folgende Möglichkeiten:

- Hardware- und Netzwerkeditor
  Nachdem Sie die Netzwerkkomponente eingefügt haben, können Sie die Eigenschaften und Parameter bearbeiten, z. B. den Gerätenamen. Weitere Informationen dazu finden Sie unter "Hardware- und Netzwerkeditor".
- Web Based Management (WBM)
  Die Parameter und Eigenschaften sind über mitgelieferten HTML-Seiten (WBM-Seiten) erreichbar. Jede WBM-Seite hat eine eigene Hilfeseite, die die Eigenschaften und Parameter beschreiben. Beachten Sie für weitere Informationen die Projektierungshandbücher:
  - "SCALANCE M-800 Web Based Management"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter (https://support.industry.siemens.com/cs/ww/de/ps/15982/man):
- Command Line Interface
  Mit dem CLI können alle Konfigurationseinstellungen für das Gerät festgelegt werden. Das CLI bietet die gleichen Möglichkeiten wie das Web Based Management (WBM). Beachten Sie für weitere Informationen die Projektierungshandbücher:
  - "SCALANCE M-800 Command Line Interface"
    Das Projektierungshandbuch finden Sie auf den Internetseiten des Siemens Industry Online Support unter:

Verfügbarkeit

Die Verfügbarkeit ist abhängig abhängig vom SCALANCE M-Gerät.

Folgende Projektierungsmodi gibt es:

- Offline-Projektierung
  Die Erstkonfiguration einer PC-Station können Sie offline durchführen. In diesem Modus sind nur die Einstellungen verfügbar, die keine Verbindung zum Gerät benötigen.
- Online-Diagnosemodus
  Wenn eine Online-Verbindung zum Gerät besteht, sind unter "Online & Diagnose" zusätzliche Seiten verfügbar. In der Dokumentation dieser Seiten ist der Hinweis "Diese Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht" enthalten. Bei einzelnen Einstellungen ist der Zusatz "Nur Online verfügbar" enthalten.

Eintrag anlegen und löschen

Als Beispiel wird ein Eintrag beim Syslog-Client angelegt und gelöscht. Das Vorgehen ist bei allen Seiten gültig.

Neuen Eintrag anlegen

1. Selektieren Sie in der Netz- oder Gerätesicht das Gerät.
2. Öffnen Sie im Inspektorfenster die Eigenschaften des Geräts.
3. Gehen Sie im Inspektorfenster zu System > Syslog-Client.
4. Tragen Sie bei IP-Adresse des Syslog-Servers die IP-Adresse ein.
5. Klicken Sie in die Tabelle.
6. Wählen Sie im Kontextmenü den Eintrag "Neuer Eintrag".
   In der Tabelle wird ein neuer Eintrag erstellt.

Eintrag löschen

1. Markieren Sie in der Tabelle den gewünschten Eintrag.
2. Wählen Sie im Kontextmenü den Eintrag "Löschen".

Häufig verwendete Schaltflächen

- **Aktualisieren der Anzeige mit "Aktualisieren"**
  Seiten, die aktuelle Parameter anzeigen, haben am unteren Rand die Schaltfläche "Aktualisieren". Klicken Sie auf diese Schaltfläche, wenn Sie für die angezeigte Seite aktuelle Daten vom Gerät anfordern wollen.
- **Speichern von Einstellungen für alle Ports mit "In Tabelle übernehmen"**
  Seiten, auf denen Sie mehrere Ports konfigurieren können, verfügen über 2 Tabellen. In der ersten Tabelle können Sie Einstellungen für alle Ports vornehmen und in die zweite Tabelle übernehmen. In der letzten Spalte der ersten Tabelle gibt es die Schaltfläche "In Tabelle übernehmen". Klicken Sie auf die Schaltfläche, um eingegebene Einstellungen für alle Ports zu speichern.

Fehlermeldung beim Laden der Konfiguration

Mögliche Probleme beim Laden

Während die Konfiguration auf ein SCALANCE-Gerät geladen wird, werden im Ausgabefenster Statusmeldungen über den Fortschritt des Ladevorgangs ausgegeben.

Wenn beim Laden auf das Gerät Probleme auftreten, ist die Ursache in den meisten Fällen eine der Folgenden:

- Falsches SCALANCE-Gerät im Projekt
- Falsche Firmware-Version
- Das SCALANCE-Gerät ist nicht mit dem Projektierungs-PC verbunden.

Fehlermeldung

Folgende Fehlermeldung kann beim Laden auftreten, wenn keine Verbindung aufgebaut werden kann:

Mögliche Fehler	Mögliche Ursachen	Mögliche Abhilfe
Fehlermeldung "Das Ziel-Gerät kann momentan keinen Download der Daten akzeptieren"	Das Ziel-Gerät befindet sich gerade in einer aktiven Anwendungsrelation mit einem IO-Controller.	Deaktivieren Sie die Anwendungsrelation zwischen dem IO-Controller und dem SCALANCE-Gerät.
	Die TIA-Konfigurationsschnittstelle auf dem Ziel-Gerät ist deaktiviert.	Aktivieren Sie über WBM oder CLI die TIA-Konfigurationsschnittstelle auf dem SCALANCE-Gerät.

Wissenswertes

IPv4-Adresse, Subnetzmaske und Adresse des Netzübergangs

Wertebereich für IPv4-Adresse

Die IPv4-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 141.80.0.16

Adressformat IPv4 - Notation

Eine IPv4-Adresse besteht aus 4 Byte. Jedes Byte wird dezimal dargestellt und ist durch einen Punkt vom vorherigen getrennt.

XXX.XXX.XXX.XXX

XXX steht für eine Zahl zwischen 0 und 255

Die IPv4-Adresse besteht aus zwei Teilen:

- Der Adresse des (Sub-)Netzes
- Der Adresse des Teilnehmers (im allgemeinen auch Endteilnehmer, Host oder Netzknoten genannt)

Wertebereich für Subnetzmaske

Die Subnetzmaske besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 255.255.0.0

Die 4 Dezimalzahlen der Subnetzmaske müssen in ihrer binären Darstellung von links eine Folge von lückenlosen Werten "1" und von rechts eine Folge von lückenlosen Werten "0" enthalten.

Die Werte "1" bestimmen die Netznummer innerhalb der IPv4-Adresse. Die Werte "0" die Host-Adresse innerhalb der IPv4-Adresse.

Beispiel:

richtige Werte:

255.255.0.0 D = 1111 1111.1111 1111.0000 0000.0000 0000 B

255.255.128.0 D = 1111 1111.1111 1111.1000 0000.0000 0000 B

255.254.0.0 D = 1111 1111.1111 1110.0000 0000.0000.0000 B

falscher Wert:

255.255.1.0 D = 1111 1111.1111 1111.0000 0001.0000 0000 B

Zusammenhang IPv4-Adresse und Subnetzmaske

Die erste Dezimalzahl der IPv4-Adresse (von links) bestimmt den Aufbau der Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt (für "x" steht die Host-Adresse):

Erste Dezimalzahl der IPv4-Adresse	Subnetzmaske
0 bis 127	255.x.x.x
128 bis 191	255.255.x.x
192 bis 223	255.255.255.x

Classless Inter-Domain Routing (CIDR)

CIDR ist ein Verfahren das mehrerer IPv4-Adressen zu einem Adressbereich zusammenfasst, indem eine IPv4-Adresse mit ihrer Subnetzmaske kombiniert dargestellt wird. Dazu wird an die IPv4-Adresse ein Suffix angehängt, das die Anzahl der auf 1 gesetzten Bits der Netzmaske angibt. Durch die CIDR-Notation lassen sich Routing-Tabellen reduzieren und die verfügbaren Adressbereiche besser ausnutzen.

Beispiel:

IPv4-Adresse 192.168.0.0 mit Subnetzmaske 255.255.255.0

Der Netzanteil der Adresse umfasst in der binären Darstellung 3 x 8 Bits, also 24 Bits.

Daraus ergibt sich die CIDR-Notation 192.168.0.0/24.
Der Host-Anteil umfasst in der binären Darstellung 1 x 8 Bits. Daraus ergibt sich der Adressbereich von 28, also 256 mögliche Adressen.

Wertebereich für Adresse des Netzübergangs

Die Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 141.80.0.1

Zusammenhang IPv4-Adresse und Adresse des Netzübergangs

Die IPv4-Adresse und die Adresse des Netzübergangs dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

Beispiel:

Sie haben eingegeben: für Subnetzmaske 255.255.255.0; für IPv4-Adresse 141.30.0.5 und für die Adresse des Netzübergangs 141.30.128.0. Die IPv4-Adresse und die Adresse des Netzübergangs dürfen nur in der 4. Dezimalzahl einen unterschiedlichen Wert haben. Im Beispiel ist aber die 3. Stelle schon unterschiedlich.

Im Beispiel müssen Sie also alternativ ändern:

die Subnetzmaske auf: 255.255.0.0 oder

die IPv4-Adresse auf: 141.30.128.1 oder

die Adresse des Netzübergangs auf: 141.30.0.1

VLAN

VLAN

Netzwerkdefinition unabhängig von der räumlichen Lage der Teilnehmer

VLAN (Virtuelles Local Area Network) teilt ein physikalisches Netzwerk in mehrere logische Netzwerke, die voneinander abgeschirmt sind. Hierbei werden Geräte zu logischen Gruppen zusammengefasst. Nur Teilnehmer des gleichen VLANs können sich untereinander adressieren. Da auch Multicast- und Broadcast-Telegramme nur innerhalb des jeweiligen VLANs weitergeleitet werden, wird von Broadcast-Domänen gesprochen.

Daraus ergibt sich als besonderer Vorteil von VLANs eine geringere Netzlast für die Teilnehmer bzw. Netzsegmente anderer VLANs.

Für die Kennung, welches Paket welchem VLAN zugeordnet ist, wird das Telegramm um 4 Byte erweitert, siehe VLAN-Tagging (Seite 2697). Diese Erweiterung enthält neben der VLAN-ID auch Prioritätsinformationen.

Möglichkeiten der VLAN-Zuordnung

Es gibt verschiedene Möglichkeiten der Zuordnung zu VLANs:

- Port-basiertes VLAN
  Jedem Port eines Geräts wird eine VLAN-ID zugewiesen. Port-basiertes VLAN konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".
- Protokoll-basiertes VLAN
  Jedem Port eines Geräts wird eine Protokollgruppe zugewiesen.
- Subnetz-basiertes VLAN
  Der IP-Adresse des Geräts wird eine VLAN-ID zugewiesen.

VLAN-Zuordnung am Gerät

SCALANCE M812	P1	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
SCALANCE M874 SCALANCE M804PB	P1 und P2	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
SCALANCE M876 SCALANCE M816	P1 bis P4	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät
SCALANCE M826	P1 bis P4 SHDSL 1 und SHDSL 2	vlan1 Für den Zugriff vom lokalen Netz (LAN) auf das Gerät

Die VLANs sind in verschiedenen IP-Subnetzen. Damit diese miteinander kommunizieren können, müssen im Gerät die entsprechende Route und die Firewall-Regel konfiguriert sein.

Die Zuordnung können Sie unter "Layer 2 > VLAN > Allgemein (Seite 2818)" ändern.

VLAN-Tagging

Erweiterung der Ethernet-Telegramme um vier Byte

Für CoS (Class of Service, Telegrammpriorisierung) und für VLAN (Virtuelles Netzwerk) wurde in der Norm IEEE 802.1Q die Erweiterung der Ethernet-Telegramme um das VLAN-Tag festgelegt.

Hinweis

Durch das VLAN-Tag erhöht sich die zulässige Gesamtlänge des Telegramms von 1518 auf 1522 Byte.
Es muss geprüft werden, ob die Endteilnehmer im Netz diese Länge / diesen Telegrammtyp verarbeiten können. Ist dies nicht der Fall, dürfen an diese Teilnehmer nur Telegramme mit der Standardlänge gesendet werden.

Die zusätzlichen 4 Bytes befinden sich im Header des Ethernet-Telegramms zwischen der Quelladresse und dem Ethernet-Typ-/Längenfeld:

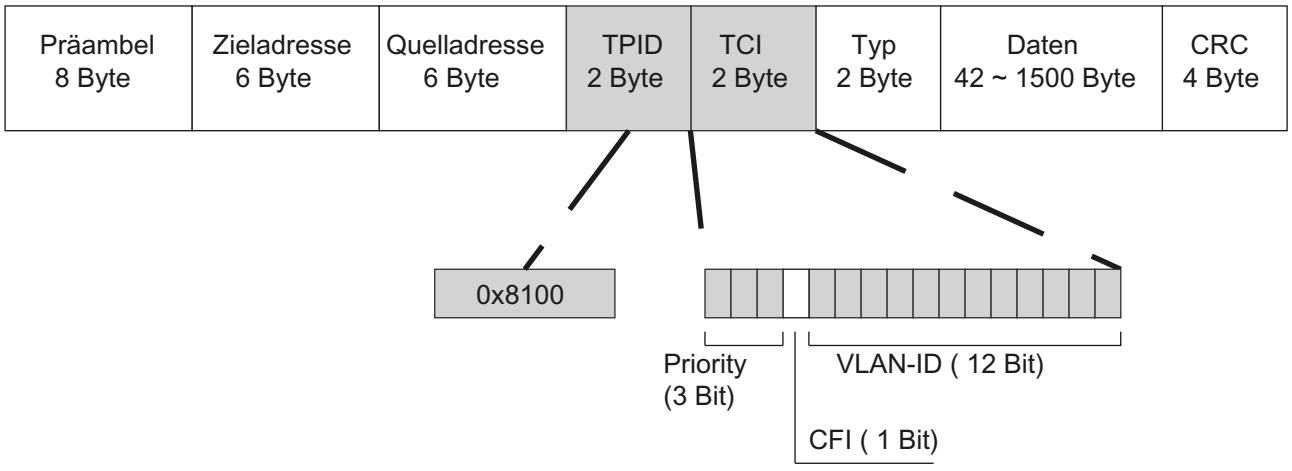

Bild 1-89 Aufbau des erweiterten Ethernet-Telegramms

Die zusätzlichen Bytes beinhalten den Tag Protocol Identifier (TPID) und die Tag Control Information (TCI).

Tag Protocol Identifier (TPID)

Die ersten 2 Bytes bilden den Tag Protocol Identifier (TPID) und sind fest mit 0x8100 belegt. Dieser Wert gibt an, dass das Datenpaket VLAN-Informationen oder Prioritätsangaben beinhaltet.

Tag Control Information (TCI)

Die 2 Bytes der Tag Control Information (TCI) beinhalten folgende Informationen:

CoS-Priorisierung

In dem getaggten Telegramm gibt es 3 Bits für die Priorität, die auch als Class of Service (CoS) bezeichnet werden, siehe auch IEEE 802.1Q.

CoS-Bits	Priorität	Art des Datenverkehrs
000	0 (niedrigste)	Background (Hintergrund)
001	1	Best Effort
010	2	Excellent Effort
011	3	Critical Applications (Kritische Anwendungen)
100	4	Video, < 100 ms Verzögerung (Latenz und Jitter)
101	5	Voice (Sprache), < 10 ms Verzögerung (Latenz und Jitter)
110	6	Internetwork Control
111	7 (höchste)	Network Control

Die Priorisierung der Datenpakete setzt eine Warteschlange in den Komponenten voraus, in der sie die Datenpakete mit der niedrigeren Priorität puffern können.

Das Gerät besitzt mehrere parallele Warteschlangen, in denen die verschieden priorisierten Telegramme abgearbeitet werden. Standardmäßig werden zuerst die Telegramme mit der höchsten Priorität abgearbeitet. Dieses Verfahren gewährleistet auch bei einem hohen Datenaufkommen, dass die Telegramme mit der höchsten Priorität auf jeden Fall gesendet werden.

Canonical Format Identifier (CFI)

Der CFI wird für die Kompatibilität zwischen Ethernet und Token Ring benötigt.
Die Werte haben folgende Bedeutung:

Wert	Bedeutung
0	Das Format der MAC-Adresse ist kanonisch. Bei kanonischer Darstellung der MAC-Adresse wird das niederwertigste Bit zuerst übertragen. Standardeinstellung für Ethernet-Switches.
1	Das Format der MAC-Adresse ist nicht kanonisch.

VLAN-ID

Im 12 Bit-Datenfeld können bis zu 4096 VLAN-IDs gebildet werden. Dabei gelten folgende Festlegungen:

VLAN-ID	Bedeutung
0	Das Telegramm beinhaltet nur Prioritätsinformation (Priority Tagged Frames) und keine gültige VLAN-Kennung.
1 - 4094	Gültige VLAN-Kennung, das Telegramm ist einem VLAN zugeordnet, es kann zusätzlich auch Prioritätsinformationen beinhalten.
4095	Reserviert

SNMP

SNMP

Einleitung

Mit Hilfe des Simple Network Management Protocol (SNMP) überwachen und steuern Sie Netzwerkkomponenten, z. B. Router oder Switches, von einer zentralen Station aus. SNMP regelt dabei die Kommunikation zwischen den überwachten Geräten und der Überwachungsstation.

Aufgaben von SNMP:

- Überwachung von Netzwerkkomponenten
- Fernsteuerung und Fernparametrierung von Netzwerkkomponenten
- Fehlererkennung und Fehlerbenachrichtigung

In den Versionen v1 und v2c verfügt SNMP über keine Sicherheitsmechanismen. Jeder Nutzer im Netzwerk kann mit geeigneter Software auf die Daten zugreifen und auch Parametrierungen verändern.

Für die einfache Steuerung von Zugriffsrechten ohne Sicherheitsaspekte werden Community-Strings verwendet.

Der Community-String wird zusammen mit der Anfrage übertragen. Wenn der Community-String korrekt ist, antwortet der SNMP-Agent und sendet die geforderten Daten. Wenn der Community-String nicht korrekt ist, verwirft der SNMP-Agent die Anfrage. Für Lese- und Schreibrechte definieren Sie verschiedene Community-Strings. Die Community-Strings werden in Klartext übertragen.

Standardwerte der Community-Strings:

- public
  besitzt nur Leserechte
- private
  besitzt Lese- und Schreibrechte

Hinweis

Da es sich bei den SNMP-Community Strings um einen Zugriffsschutz handelt, verwenden Sie nicht die Standardwerte "public" oder "private". Ändern Sie diese Werte nach der Erst-Inbetriebnahme.

Weitere einfache Schutzmechanismen auf Geräteebene:

- Allowed Host
  Dem überwachten System sind die IP-Adressen der überwachenden Systeme bekannt.
- Read Only
  Wenn Sie einem überwachten Gerät "Read Only" zuweisen, können Überwachungsstationen nur Daten auslesen, aber nicht ändern.

SNMP-Datenpakete sind nicht verschlüsselt und können einfach mitgelesen werden.

Die zentrale Station wird auch als Management-Station bezeichnet. Auf den zu überwachenden Geräten ist ein SNMP-Agent installiert, mit dem die Management-Station Daten austauscht.

Die Management-Station sendet Datenpakete folgenden Typs:

- GET
  Anfordern eines Datensatzes vom SNMP-Agent
- GETNEXT
  Ruft den nächsten Datensatz auf.
- GETBULK (verfügbar ab SNMPv2c)
  Fordert mehrere Datensätze auf einmal an, z. B. mehrere Zeilen einer Tabelle.
- SET
  Beinhaltet Parametrierungsdaten für das entsprechende Gerät.

Der SNMP-Agent sendet Datenpakete folgenden Typs:

- RESPONSE
  Der SNMP-Agent sendet die vom Manager angeforderten Daten zurück.
- TRAP
  Wenn ein bestimmtes Ereignis eintritt, sendet der SNMP-Agent eigenständig Traps.

SNMPv1/v2c/v3 verwenden UDP (User Datagram Protocol) und nutzen die UDP-Ports 161 und 162. Die Beschreibung der Daten erfolgt in einer Management Information Base (MIB).

SNMPv3

SNMPv3 führt gegenüber den Vorgängerversionen SNMPv1 und SNMPv2c ein umfangreicheres Sicherheitskonzept ein.

SNMPv3 unterstützt:

- Vollständig verschlüsselte Benutzerauthentifizierung
- Verschlüsselung des gesamten Datenverkehrs
- Zugriffskontrolle der MIB-Objekte auf Benutzer-/Gruppenebene

Mit der Einführung von SNMPv3 können Sie Benutzerkonfigurationen nicht mehr ohne Weiteres auf andere Geräte übertragen, z. B. indem Sie eine Konfigurationsdatei laden oder den C-PLUG austauschen.

Das SNMPv3-Protokoll verwendet gemäß des Standards eine eindeutige SNMP-Engine-ID als internen Bezeichner für einen SNMP-Agenten. Diese ID muss im Netzwerk eindeutig sein. Sie wird verwendet, um die Zugangsdaten von SNMPv3-Benutzern zu authentifizieren und zu verschlüsseln.

Abhängig davon, ob Sie die Funktion "SNMPv3 Benutzermigration" aktiviert oder deaktiviert haben, wird die SNMP-Engine-ID unterschiedlich generiert.

Einschränkung bei der Verwendung der Funktion

Verwenden Sie die Funktion "SNMPv3 Benutzermigration" nur, um im Ersatzteilfall Ihre konfigurierten SNMPv3-Benutzer auf ein Ersatzgerät zu übertragen.
Verwenden Sie die Funktion nicht, um konfigurierte SNMPv3-Benutzer auf mehrere Geräte zu übertragen. Wenn Sie eine Konfiguration mit angelegten SNMPv3-Benutzern in mehrere Geräte laden, verwenden diese Geräte hierdurch die gleiche SNMP-Engine-ID. Wenn Sie diese Geräte im gleichen Netzwerk verwenden, widerspricht Ihre Konfiguration dem SNMP-Standard.

Kompatibilität mit Vorgängerprodukten

Sie können SNMPv3-Benutzer nur auf ein anderes Gerät übertragen, wenn Sie die Benutzer als migrierbare Benutzer erstellt haben. Um einen migrierbaren Benutzer zu erstellen, muss die Funktion "SNMPv3 Benutzermigration" aktiviert sein, wenn Sie den Benutzer erstellen.

ICMP

Die Abkürzung ICMP steht für Internet Control Message Protocol (RFC792) und dient zum Austausch von Fehler- und Informationsmeldungen.

- Fehlermeldung
  Informiert den Absender des IP-Telegramms, dass beim Weiterleiten des Telegramms ein Fehler oder ein Parameterproblem aufgetreten ist.
- Informationsmeldung
  Kann Informationen zur Zeitmessung, zur Adressmaske, zur Erreichbarkeit des Ziels oder zum Auffinden des Routers enthalten.

Aufbau des ICMP-Datenpakets

0	4	8	12	16	20	24	28	31
ICMP-Pakettyp Art der Meldung		Code Weitere Details der Meldung		Prüfsumme				
Daten (optional)								

- **ICMP-Pakettyp**
  Die wichtigsten ICMP-Pakettypen sind:
  - Redirect
    Der Router teilt dem Host in einem seiner Subnetze mit, dass es eine bessere Route zum Ziel gibt. Auf diesen ICMP-Pakettyp wird in der folgenden Beschreibung näher eingegangen.
  - Destination Unreachable
    IP-Telegramm ist nicht zustellbar.
  - Time Exceeded
    Zeitlimit überschritten
  - Echo-Request
    Echo-Frage, besser bekannt als Ping.
- **Code**
  Der Code beschreibt den ICMP-Pakettyp genauer. Die Auswahl ist abhängig vom gewählten ICMP-Pakettyp. Bei "Destination Unreachable" ist z. B. "Code 1" Host ist nicht erreichbar.

Eine vollständige Liste der ICMP-Pakettypen und Codes finden Sie auf der Webseite von IANA (http://www.iana.org/assignments/icmp-parameters).

ICMP-Pakettyp 5 - Redirect

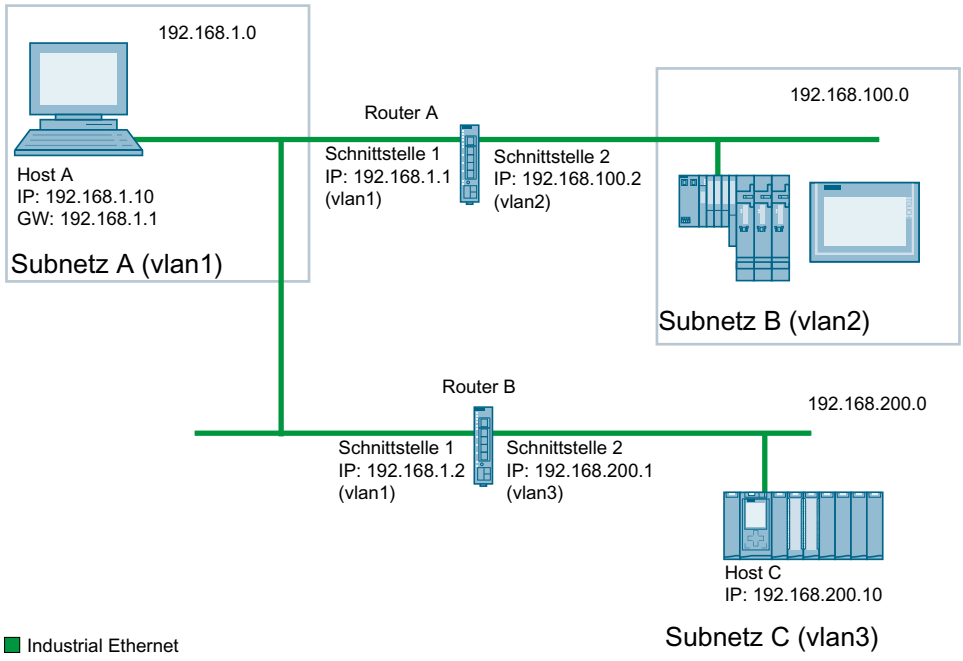

Der Host A will an den Host C ein IP-Telegramm senden. Der Host C liegt nicht im gleichen Subnetz wie der Host A. Deshalb sendet der Host A das IP-Telegramm an seinen Default-Gateway. Das Default-Gateway von Host A ist die Schnittstelle 1 des Routers A. Der Router A kann das IP-Telegramm nicht weiterleiten, da er das Zielnetzwerk nicht kennt. Über seine Routingtabelle weiß Router A jedoch, dass das Subnetz C über den Router B erreichbar ist. Der Router B verbindet das Subnetz A mit dem Subnetz C. Der Router A schickt an den Host A eine Redirect-Nachricht. Darin weist der Router A den Host A an, die IP-Telegramme an Host C zukünftig über Router B zu senden, dessen IP-Adresse in der Redirect-Nachricht enthalten ist. Das initiale IP-Telegramm wird vom Router A direkt an Router B gesendet, der es an Host C weitergibt.

Bedingungen für den Versand von Redirect-Nachrichten

- Das IP-Telegramm wird über die gleiche Schnittstelle des Routers A empfangen und gesendet.
- Die Quell-IP-Adresse (Host A) ist aus dem gleichen Subnetz wie die Next-Hop-Adresse (Router B) in der Routingtabelle.
- Das IP-Telegramm ist nicht von einer Source NAT-Regel (Masquerading, Source-NAT oder NETMAP) betroffen.
- Damit der Router A das initiale IP-Telegramm an Router B weiterleitet, wird eine Firewall Regel vlanX → vlanX benötigt.

Security-Funktionen

Benutzerverwaltung (WBM, CLI)

Übersicht zur Benutzerverwaltung

Der Zugriff auf das Gerät wird durch konfigurierbare Benutzereinstellungen verwaltet. Richten Sie Benutzer mit jeweils einem Passwort zur Authentifizierung ein. Weisen Sie den Benutzern eine Rolle mit entsprechenden Rechten zu.

Die Authentifizierung von Benutzern kann entweder von dem Gerät lokal oder von einem externen RADIUS-Server durchgeführt werden. Wie die Authentifizierung erfolgen soll, konfigurieren Sie auf der Seite "Security > AAA > Allgemein".

Lokale Anmeldung

Die lokale Anmeldung von Benutzern durch das Gerät läuft wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät prüft, ob ein Eintrag für den Benutzer vorhanden ist:
   → Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet.
   → Wenn kein entsprechender Eintrag existiert, wird dem Benutzer der Zugriff verweigert.

Anmeldung über einen externen RADIUS-Server

RADIUS (Remote Authentication Dial-In User Service) ist ein Protokoll zur Authentifizierung und Autorisierung von Benutzern durch Server, auf denen Benutzerdaten zentral abgelegt werden können.

Anhängig davon, welchen RADIUS-Autorisierungsmodus Sie auf der Seite "Security > AAA > RADIUS-Client" eingestellt haben, wertet das Gerät unterschiedliche Informationen des RADIUS-Servers aus.

RADIUS-Autorisierungsmodus "Standard"

Wenn Sie den RADIUS-Autorisierungsmodus "Standard" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück:
     → Der Benutzer wird mit Administratorrechten angemeldet.
   - Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt einen anderen oder gar keinen Wert für das Attribut "Service Type" an das Gerät zurück:
     → Der Benutzer wird mit Leserechten angemeldet.
   - Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
     → Dem Benutzer wird der Zugriff verweigert.

RADIUS-Autorisierungsmodus "Herstellerspezifisch"

Voraussetzung

Für den RADIUS-Autorisierungsmodus "Herstellerspezifisch" ist am RADIUS-Server Folgendes einzustellen:

- Herstellercode: 4196
- Attributnummer: 1
- Attributformat: Zeichenfolge (Gruppenname)

Vorgehen

Wenn Sie den RADIUS-Autorisierungsmodus "Herstellerspezifisch" eingestellt haben, läuft die Authentifizierung von Benutzern über einen RADIUS-Server wie folgt ab:

1. Der Benutzer meldet sich mit seinem Benutzernamen und Passwort bei dem Gerät an.
2. Das Gerät schickt eine Authentifizierungsanfrage mit den Anmeldedaten an den RADIUS-Server.
3. Der RADIUS-Server führt eine Prüfung durch und meldet das Ergebnis an das Gerät zurück:
   Fall A: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung und gibt die dem Benutzer zugeordnete Gruppe an das Gerät zurück.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird mit den Rechten der zugeordneten Gruppe angemeldet.
   - Die Gruppe ist auf dem Gerät bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen.
     → Der Benutzer wird der Rolle mit den größeren Rechten zugeordnet und mit diesen Rechten angemeldet.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle angemeldet, die mit seinem Benutzeraccount verknüpft ist.
   - Die Gruppe ist auf dem Gerät nicht bekannt und der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall B: Der RADIUS-Server meldet eine erfolgreiche Authentifizierung, gibt jedoch keine Gruppe an das Gerät zurück:
   - Der Benutzer ist in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der verknüpften Rolle angemeldet.
   - Der Benutzer ist nicht in der Tabelle "Externe Benutzerkonten" eingetragen:
     → Der Benutzer wird mit den Rechten der Rolle "Default" angemeldet.

   Fall C: Der RADIUS-Server meldet eine fehlgeschlagene Authentifizierung an das Gerät zurück:
   - Dem Benutzer wird der Zugriff verweigert.

Firewall

Zu den Security-Funktionen des Geräts gehört eine Stateful Inspection Firewall. Dabei handelt es sich um eine Methode der Paketfilterung bzw. Paketüberprüfung.

Die IP-Pakete werden anhand von Firewall-Regeln geprüft, in denen Folgendes festgelegt wird:

- Die erlaubten Protokolle
- IP-Adressen und Ports der erlaubten Quellen
- IP-Adressen und Ports der erlaubten Ziele

Wenn ein IP-Paket den festgelegten Parametern entspricht, dann darf es die Firewall passieren. Zusätzlich wird festgelegt, wie mit IP-Paketen verfahren wird, welche die Firewall nicht passieren dürfen.

Einfache Paketfiltertechniken benötigen pro Verbindung zwei Firewall-Regeln.

- Eine Regel für Anfragerichtung von der Quelle zum Ziel.
- Eine zweite Regel für die Antwortrichtung vom Ziel zur Quelle

Stateful Inspection Firewall

Sie müssen nur eine Firewall-Regel für die Anfragerichtung von der Quelle zum Ziel festlegen. Die zweite Regel wird implizit hinzugefügt. Der Paketfilter merkt sich, wenn z. B. Rechner "A" mit Rechner "B" kommuniziert und erlaubt nur dann Antworten darauf. Eine Anfrage von Rechner "B" ist somit ohne vorherige Anforderung durch Rechner "A" nicht möglich.

Die Firewall konfigurieren Sie unter "Security > Firewall".

Hinweis

IP-Pakete über Layer 2 (innerhalb des gleichen VLANs)

Wenn die IP-Pakete von dem Gerät über einen Switch-Port (Layer 2) gesendet werden, werden diese IP-Pakete nicht anhand von Firewall-Regeln geprüft. Die Firewall hat keine Wirkung auf Pakete, die auf der Layer 2-Ebene weitergeleitet werden.

Kommunikationsrichtungen

von	nach	Bedeutung
vlan x	vlan x	Zugriff vom IP-Subnetz vlan x und dem Gerät auf das IP-Subnetz vlan x. Beispiel: vlan1 (INT) → vlan2 (EXT) Zugriff vom lokalen IP-Subnetz und dem Gerät auf das externe IP-Subnetz.
	ppp0/usb	Zugriff vom IP-Subnetz auf die Mobilfunk-Schnittstelle des Geräts.
	Gerät	Zugriff vom IP-Subnetz auf das Gerät.
	SINEMA RC	Zugriff vom IP-Subnetz und dem Gerät auf die SINEMA RC-Verbindung.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom IP-Subnetz auf die Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.

von	nach	Bedeutung
Gerät	vlan x	Zugriff vom Gerät auf das IP-Subnetz.
	ppp0/usb	Zugriff vom Gerät auf die Mobilfunk-Schnittstelle des Geräts.
	SINEMA RC	Zugriff vom Gerät auf die SINEMA RC-Verbindung
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom Gerät auf die Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.
SINEMA RC	vlan x	Zugriff vom SINEMA RC-Server auf das IP-Subnetz.
	ppp0/usb	Zugriff vom SINEMA RC-Server auf die Mobilfunk-Schnittstelle des Geräts.
	Gerät	Zugriff vom SINEMA RC-Server auf das Gerät.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff vom SINEMA RC-Server auf die Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.
IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	vlan x	Zugriff über Tunnelpartner auf das IP-Subnetz.
	ppp0/usb	Zugriff über Tunnelpartner auf die Mobilfunk-Schnittstelle des Geräts.
	Gerät	Zugriff über Tunnelpartner auf das Gerät.
	SINEMA RC	Zugriff über Tunnelpartner auf die SINEMA RC-Verbindung.
ppp0/usb	vlan x	Zugriff von der Mobilfunk-Schnittstelle auf das IP-Subnetz.
	Gerät	Zugriff von der Mobilfunk-Schnittstelle auf das Gerät.
	SINEMA RC	Zugriff von der Mobilfunk-Schnittstelle auf die SINEMA RC-Verbindung.
	IPsec (all) IPsec <Connection Name> OpenVPN (all) OpenVPN <Connection Name>	Zugriff von der Mobilfunk-Schnittstelle auf die Tunnelpartner, die über alle VPN-Verbindungen (all) oder über eine bestimmte VPN-Verbindung <Connection Name> zu erreichen sind.

Werkseinstellung der Firewall

Dienst	Zugriff	
	von intern (vlan1) auf das Gerät	von extern (ppp0/usb0) auf das Gerät
HTTP	ja	nein
HTTPS	ja	nein
DNS	ja	nein
SNMP	ja	nein
Telnet	ja	nein
IPsec VPN	nein	ja

Dienst	Zugriff	
	von intern (vlan1) auf das Gerät	von extern (ppp0/usb0) auf das Gerät
SSH	ja	nein
DHCP	ja	ja (nur bei M826 für die Funktion DHCP-Server)
Ping	ja	nein
Systemzeit	nein	nein
SMS Relay (nur bei M87x)	ja	nein
Cloud Connector (nur bei M804PB)	ja	-

NAT

NAT (Network Address Translation) ist eine Methode IP-Adressen in Datenpaketen umzuschreiben. Damit können zwei verschiedene Netze (intern und extern) miteinander verbunden werden.

Man unterscheidet zwischen Source-NAT, bei dem die Quell-IP-Adresse umgeschrieben wird und Destination-NAT, bei dem die Ziel-IP-Adresse umgeschrieben wird.

Weitere Informationen zu NAT-Szenarien, die mit dem Gerät realisiert werden können, finden Sie unter folgender Adresse: (https://support.industry.siemens.com/cs/de/de/view/109744660)

IP-Masquerading

IP-Masquerading ist ein vereinfachtes Source-NAT. Dabei wird bei jedem ausgehenden Datenpaket, das über diese Schnittstelle gesendet wird, die Quell-IP-Adresse durch die IP-Adresse der Schnittstelle ersetzt. Das angepasste Datenpaket wird an die Ziel-IP-Adresse gesendet. Für den Ziel-Host sieht es so aus, als kämen die Anfragen immer von dem gleichen Absender. Die internen Teilnehmer sind aus dem externen Netz nicht direkt erreichbar. Mithilfe von NAPT lassen sich die Dienste der internen Teilnehmer über die externe IP-Adresse des Geräts erreichbar machen.

IP-Masquerading kann benutzt werden, wenn die internen IP-Adressen extern nicht weitergeleitet werden können oder sollen, z. B. weil die interne Netzstruktur verborgen werden soll.

Masquerading konfigurieren Sie unter "Layer 3" > "NAT" > "IP-Masquerading (Seite 2831)".

NAPT

NAPT (Network Address and Port Translation) ist eine Form des Destination-NAT und wird oft auch als Portweiterleitung (Port Forwarding) bezeichnet. Damit lassen sich Dienste der internen Teilnehmer von außen erreichbar machen, die durch IP-Masquerading oder Source-NAT versteckt sind.

Umgesetzt werden eingehende Datenpakete, die vom externen Netz kommen und an eine externe IP-Adresse des Geräts (Ziel-IP-Adresse) gerichtet sind. Die Ziel-IP-Adresse wird mit der IP-Adresse des internen Teilnehmers ersetzt. Zusätzlich zur Adressumsetzung ist auch eine Port-Umsetzung möglich.

Es gibt folgende Möglichkeiten der Port-Umsetzung:

von	zu	Verhalten
einem einzigen Port	dem gleichen Port	Wenn die Ports gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.
einem einzigen Port	einem einzigen Port	Die Telegramme werden auf den Port umgesetzt.
einem Port-Bereich	einem einzigen Port	Die Telegramme aus dem Port-Bereich werden auf den gleichen Port umgesetzt (n:1).
einem Port-Bereich	dem gleichen Port-Bereich	Wenn die Port-Bereiche gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.
einem Port-Bereich	einem anderen Port-Bereich	Die Telegramme werden auf einen beliebigen freien Port aus dem Zielbereich umgesetzt. Bei einzelnen Verbindungen wird meist auf den ersten Port im Zielbereich umgesetzt. Bei gleichzeitigen Verbindungen wird mittels einer Reih-um-Methode (round robin) auf einen freien Port im Zielbereich umgesetzt.
einem einzigen Port	einem Port-Bereich	Die Telegramme werden auf einen beliebigen freien Port aus dem Zielbereich umgesetzt. Bei einzelnen Verbindungen wird meist auf den ersten Port im Zielbereich umgesetzt. Bei gleichzeitigen Verbindungen wird mittels einer Reih-um-Methode (round robin) auf einen freien Port im Zielbereich umgesetzt.

Port Forwarding kann benutzt werden, um externen Teilnehmern den Zugriff auf bestimmte Dienste des internen Netzes zu ermöglichen, z. B. FTP, HTTP.

NAPT konfigurieren Sie unter "Layer 3" > "NAT" > "NAPT (Seite 2832)".

Source-NAT

Wie beim Masquerading wird beim Source-NAT die Quelladresse umgeschrieben. Zusätzlich können die ausgehenden Datenpakete beschränkt werden. Dazu gehören Beschränkungen auf bestimme IP-Adressen oder IP-Adressbereiche und Beschränkungen auf bestimmte Schnittstellen.

Source-NAT kann benutzt werden, wenn die internen IP-Adressen extern nicht weitergeleitet werden können oder sollen, z. B. weil ein privater IP-Adressbereich wie 192.168.x.x benutzt wird.

Source-NAT konfigurieren Sie unter "Layer 3" > "NAT" > "Source NAT (Seite 2834)".

NETMAP

Mit NETMAP ist es möglich, komplette Subnetze auf ein anderes Subnetz umzusetzen. Bei dieser Umsetzung wird der Subnetzanteil der IP-Adresse geändert und der Hostanteil bleibt bestehen. Für die Umsetzung wird bei NETMAP nur eine Regel benötigt. NETMAP kann sowohl die Quell-IP-Adresse als auch die Ziel-IP-Adresse umsetzen. Um die Umsetzung mit Destination-NAT und Source-NAT durchzuführen, wären viele Regeln notwendig. NETMAP kann auch auf VPN-Verbindungen angewendet werden.

NETMAP konfigurieren Sie unter "Layer 3" > "NAT" > "NETMAP (Seite 2835)".

NAT und Firewall

Firewall und NAT-Router unterstützen den Mechanismus "Stateful Inspection". Wenn der IP-Datenverkehr von intern nach extern freigegeben ist, können interne Teilnehmer eine Kommunikationsverbindung in das externe Netz initiieren.

Die Antworttelegramme aus dem externen Netz können den NAT-Router und die Firewall passieren, ohne dass deren Adressen in der Firewall-Regel und der NAT-Adressumsetzung zusätzlich aufgenommen werden müssen. Telegramme, die keine Antwort auf eine Anfrage aus dem internen Netz sind, werden ohne zutreffende Firewallregel verworfen.

NAT-Umsetzung und Firewall-Regeln

Beispiel für NAT-Umsetzungen

NAT-Regel							
	Typ	Quell-Schnittstelle	Ziel-Schnittstelle	Quell-IP-Subnetz	Quell-IP-Subnetzumsetzung	Ziel-IP-Subnetz	Ziel-IP-Subnetzumsetzung
①	Quelle	vlan1 (intern)	vlan2 (extern)	192.168.1.0/24	10.100.1.0/24	10.10.10.0/24	-
	Die Regel gilt für Pakete, die von vlan1 (intern) nach vlan2 (extern) gesendet werden. Bei den Paketen, die an vlan1 ankommen, wird geprüft, ob die Regel zutrifft. Wenn die Quell-IP-Adresse im Subnetz des Absenders (Quell-IP-Subnetz) und die Ziel-IP-Adresse im Subnetz des Empfängers (Quell-IP-Subnetz) liegen, wird die Quell-IP-Adresse durch die passende IP-Adresse aus dem "Quell-IP-Subnetzumsetzung" ersetzt. Der Subnetzanteil der Quell-IP-Adresse wird geändert und der Hostanteil bleibt unverändert. Ein Paket z. B. mit der Quell-IP-Adresse 192.168.1.102 wird zu 10.100.1.102 geändert. Für die Geräte, die an vlan2 angeschlossen sind, sieht es so aus, als ob die Pakete aus dem IP-Subnetz 10.100.1.0/24 gesendet werden. Damit lassen sich z. B. Überschneidungen von IP-Subnetzen auflösen. Die Regel ist nur für die Senderichtung festzulegen. Die Rückübersetzung erfolgt implizit. Wenn die Regel nicht zutrifft, werden die Pakete ohne Umsetzung weitergeleitet.						
②	Ziel	vlan2 (extern)	vlan1 (intern)	10.10.10.0/24	-	10.100.1.0/24	192.168.1.0/24
	Die Regel gilt für Pakete, die von vlan2 (extern) nach vlan1 (intern) gesendet werden. Bei den Paketen, die an vlan2 ankommen, wird geprüft, ob die Regel zutrifft. Wenn die Quell-IP-Adresse im Subnetz des Absenders Quell-IP-Subnetz) und die Ziel-IP-Adresse im Subnetz des Empfängers (Quell-IP-Subnetz) liegen, wird die Quell-IP-Adresse durch die passende IP-Adresse aus dem "Ziel-IP-Subnetzumsetzung" ersetzt. Ein Paket z. B. mit der Quell-IP-Adresse 10.10.10.102 wird zu 192.168.1.102 geändert. Die an vlan1 angeschlossenen Geräte können mit den Geräten kommunizieren, die an vlan2 angeschlossen sind. Vorausgesetzt, die entsprechende Firewallregel ist gesetzt. Die an vlan2 angeschlossenen Geräte müssen die am vlan1 angeschlossenen Geräte mit der virtuellen IP-Adresse aus dem Subnetz 10.100.1.0 adressieren.						

Firewall-Regeln für die NAT-Regeln ① und ②

Beispiel 1:

Diese IP-Paketfilter-Regeln erlauben für die angegebene Richtung den IP-Datenverkehr für alle Geräte.

NAT-Regel	IP-Paketfilter-Regeln						Beschreibung
	Aktion	Von	Nach	Quelle (Bereich)	Ziel (Bereich)	Dienst	
①	Accept	vlan1 (intern)	vlan2 (extern)	192.168.1.0/24 (Quell-IP-Subnetz)	10.10.10.0/24 (Ziel-IP-Subnetz)	all	Alle Pakete, die von vlan1 (intern) nach vlan2 (extern) gesendet werden, werden durchgelassen. Diese IP-Paketfilter-Regel gilt für die an vlan1 angeschlossenen Geräte sind.
②	Accept	vlan2 (extern)	vlan1 (intern)	192.168.1.0/24 (Ziel-IP-Subnetzumsetzung)	10.100.1.0/24 (Ziel-IP-Subnetz)	all	Alle Pakete, die von vlan2 (extern) nach vlan1 (intern) gesendet werden, werden durchgelassen.

Beispiel 2:

Diese IP-Paketfilter-Regeln schränken den IP-Datenverkehr auf ein bestimmtes Gerät ein.

NAT-Regel	IP-Paketfilter-Regeln						Beschreibung
	Aktion	Von	Nach	Quelle (Bereich)	Ziel (Bereich)	Dienst	
①	Accept	vlan1 (intern)	vlan2 (extern)	192.168.1.20/32 (Quell-IP-Subnetz)	10.10.10.0/24 (Ziel-IP-Subnetz)	all	Nur die Pakete, die von der IP-Adresse 192.168.1.20 nach vlan2 (extern) gesendet werden, werden durchgelassen.
②	Accept	vlan2 (extern)	vlan1 (intern)	192.168.1.20/32 (Ziel-IP-Subnetzumsetzung)	10.100.1.0/24 (Ziel-IP-Subnetz)	all	Nur die Pakete, die von vlan2 (extern) an die IP-Adresse 192.168.1.20 gesendet werden, werden durchgelassen.

Zertifikate

Zertifikatstypen

Das Gerät verwendet verschiedene Zertifikate, um die verschiedenen Teilnehmer zu authentifizieren.

Zertifikat		Wird verwendet in ...
CA-Zertifikat	Das CA-Zertifikat ist ein durch eine Zertifizierungsstelle, die so genannte "Certificate Authority", ausgestelltes Zertifikat, von denen die Server-, Geräte- und Gegenstellenzertifikate abgeleitet werden. Damit ein Zertifikat abgeleitet werden kann, besitzt das CA-Zertifikat einen privaten Schlüssel, der durch die Zertifizierungsstelle signiert wurde. Der Schlüsselaustausch zwischen dem Gerät und dem VPN-Gateway der Gegenstelle erfolgt automatisch beim Aufbau der Verbindung. Es ist kein manueller Austausch von Schlüsseldateien notwendig.	IPsec VPN
Serverzertifikat	Serverzertifikate werden zum Aufbau einer gesicherten Kommunikation (z. B. HTTPS, VPN ...) zwischen Gerät und einem weiteren Netzwerkteilnehmer benötigt. Bei dem Serverzertifikat handelt es sich um ein verschlüsseltes SSL-Zertifikat. Das Serverzertifikat wird von der ältesten gültigen CA abgeleitet, auch wenn dieses "außer Dienst" ist. Entscheidend ist das Gültigkeitsdatum der CA.	SINEMA RC
Gerätezertifikat	Zertifikate mit dem privaten Schlüssel (Key file), mit denen sich das Gerät ausweist.	IPsec VPN
Gegenstellenzertifikat	Zertifikate, mit denen sich das VPN-Gateway der Gegenstelle bei dem Gerät authentifiziert.	IPsec VPN

Dateitypen

Dateityp	Beschreibung
*.crt	Datei, die das Zertifikat enthält.
*.p12	Bei der PKCS12-Zertifikatsdatei wird der private Schlüssel mit dem zugehörigen Zertifikat passwortgeschützt gespeichert. Die CA erstellt für beide Gegenstellen einer VPN-Verbindung je eine Zertifikatsdatei (PKCS12) mit der Dateiendung ".p12". Diese Zertifikatsdatei enthält den öffentlichen und privaten Schlüssel der eigenen Station, das signierte Zertifikat der CA und den öffentlichen Schlüssel der CA.
*.pem	Zertifikat und Schlüssel als Base64-kodierten ASCII-Text.

VPN

Das Gerät unterstützt folgende VPN-Systeme

- IPsec VPN
- OpenVPN

IPsec VPN

Die IPsec-Verbindungen konfigurieren Sie unter "Security" > "IPsec VPN"

Bei IPsec VPN werden die Telegramme im Tunnel-Modus übertragen. Damit das Gerät einen VPN-Tunnel aufbauen kann, muss das entfernte Netz über ein VPN-Gateway als Gegenstation verfügen.

Für die VPN-Verbindungen unterscheidet das Gerät zwei Modi:

- Roadwarrior-Modus
  In diesem Modus wird entweder die Adresse der Gegenstelle fest vorgegeben oder ein IP-Bereich eingetragen, aus dem Verbindungen entgegengenommen werden. Die erreichbaren Remote-Subnetze lernt das Gerät von der Gegenstelle.
- Standard-Modus
  In diesem Modus ist die Adresse der Gegenstelle oder das Remote-Subnetz fest eingetragen. Das Gerät kann entweder als VPN-Client die VPN-Verbindung aktiv aufbauen, oder passiv auf den Verbindungsaufbau durch die Gegenstelle warten.

Das IPsec-Verfahren

Das Gerät verwendet für den VPN-Tunnel das IPsec-Verfahren im Tunnelmodus. Dabei werden die zu übertragenden Telegramme vollkommen verschlüsselt und mit einem neuen Header versehen, bevor sie zum VPN-Gateway der Gegenstelle gesendet werden. Von der Gegenstelle werden die empfangenen Telegramme entschlüsselt und an den Empfänger weitergeleitet.

Zum Absichern verwendet das IPsec-Verfahren verschiedene Protokolle:

- Der IP-Authentication-Header (AH) wickelt die Authentifizierung und Identifizierung der Quelle ab.
- Die Encapsulation Security Payload (ESP) verschlüsselt die Daten.
- Die Security Association (SA) enthält die Festlegungen, die zwischen den Partner ausgehandelt wurden, z. B. über die Lebensdauer des Schlüssels, den Verschlüsselungsalgorithmus, den Zeitraum für eine neue Authentifizierung etc.
- Das Internet Key Exchange (IKE) ist ein Schlüsselaustauschverfahren. Der Schlüsselaustausch erfolgt in zwei Phasen:
  - Phase 1
    In dieser Phase sind noch keine Sicherheitsdienste wie Verschlüsselung, Authentifizierung und Integritätsprüfung verfügbar, da die notwendigen Schlüssel und die IPsec-SA noch nicht erstellt wurden. Phase 1 dient zum Aufbau eines sicheren VPN-Tunnels für Phase 2. Dafür verhandeln die Kommunikationspartner eine ISAKMP Security Association (ISAKMP-SA), welche die notwendigen Sicherheitsdienste (verwendete Algorithmen, Authentifizierungsmethoden) definiert. Damit werden die weiteren Nachrichten und Phase 2 abgesichert.
  - Phase 2
    Phase 2 dient zur Aushandlung der benötigten IPsec-SA. Ähnlich wie bei Phase 1 wird durch das wechselseitige Anbieten eine Einigung über die Authentifizierungsmethoden, die Algorithmen und die Verschlüsselungsverfahren getroffen, um die IP-Pakete mit IPsec-AH und IPsec-ESP zu schützen.
    Geschützt wird der Nachrichtenaustausch über die ISAKMP-SA, die in Phase 1 vereinbart wurde. Durch die in Phase 1 ausgehandelte ISAKMP-SA ist die Identität der Teilnehmer sowie das Verfahren zur Integritätsprüfung bereits gegeben.

Authentifizierungsverfahren

- CA-Zertifikat, Geräte- und Gegenstellenzertifikat (digitale Signaturen)
  Die Verwendung von Zertifikaten ist ein asymmetrisches Kryptosystem, wobei jeder Teilnehmer (Gerät) über ein Schlüsselpaar verfügt. Jeder Teilnehmer besitzt einen geheimen, privaten Schlüssel und einen öffentlichen Schlüssel der Gegenstelle. Der private Schlüssel ermöglicht es, sich zu authentisieren und digitale Signaturen zu erzeugen.
- Preshared Key
  Die Verwendung eines Preshared Key ist ein symmetrisches Kryptosystem. Jeder Teilnehmer besitzt nur einen geheimen Schlüssel für die Ent- und Verschlüsselung von Datenpaketen. Die Authentifizierung erfolgt über ein gemeinsames Passwort.

Lokale-ID und Remote-ID

Die Lokale-ID und die Remote-ID werden vom IPsec genutzt, um beim Aufbau der VPN-Verbindung die Gegenstellen (VPN-Endpunkt) eindeutig zu identifizieren.

Verschlüsselungsverfahren

Folgende Verschlüsselungsverfahren werden unterstützt. Die Auswahl ist abhängig von der Phase und vom Schlüsselaustauschverfahren (IKE):

	Phase 1		Phase 2	
	IKEv1	IKEv2	IKEv1	IKEv2
3DES	x	x	x	x
AES128 CBC	x	x	x	x
AES192 CBC	x	x	x	x
AES256 CBC	x	x	x	x
AES128 CTR	-	x	x	x
AES192 CTR	-	x	x	x
AES256 CTR	-	x	x	x
AES128 CCM 16	-	x	x	x
AES192 CCM 16	-	x	x	x
AES256 CCM 16	-	x	x	x
AES128 GCM 16	-	x	x	x
AES192 GCM 16	-	x	x	x
AES256 GCM 16	-	x	x	x

x: wird unterstützt

-: wird nicht unterstützt

Default-Chiffre

Beim Verbindungsaufbau kann eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt werden. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Die Kombinationen sind abhängig von der Phase und vom Schlüsselaustauschverfahren (IKE).

Kombination			Phase 1		Phase 2	
Verschlüsselung	Authentifizierung	Schlüsselableitung	IKEv1	IKEv2	IKEv1	IKEv2
AES128	SHA1	DH Group 14	x	x	x	x
AES256	SHA512	DH Group 16	x	x	x	x
AES128 CCM 16	SHA256	DH Group 14	-	x	x	x
AES256 CCM 16	SHA512	DH Group 16	-	x	x	x
AES128	SHA1	none	-	-	x	x
AES256	SHA512	none	-	-	x	x
AES128 CCM 16	SHA256	none	-	-	x	x
AES256 CCM 16	SHA512	none	-	-	x	x

x: Kombination ist Teil der Default-Chiffre

-: Kombination ist nicht Teil der Default-Chiffre

none: Für die Phase 2 werden keine separaten Schlüssel ausgetauscht. Damit ist Perfect Forward Secrecy (PFS) deaktiviert.

Anforderungen an die VPN-Gegenstelle

Die VPN-Gegenstelle muss IPsec mit folgender Konfiguration unterstützen, um erfolgreich eine IPsec-Verbindung aufzubauen:

- Authentifizierung über Gegenstellenzertifikate, CA-Zertifikate oder Pre-Shared Key
- IKEv1 oder IKEv2
- Unterstützung von mindestens einer der folgenden DH-Gruppen: Diffie-Hellman Gruppe 1, 2, 5 und 14 - 18
- 3DES- oder AES-Verschlüsselung
- MD5, SHA1, SHA256, SHA384 oder SHA512
- Tunnel-Modus

Wenn sich die VPN-Gegenstelle hinter einem NAT-Router befindet, dann muss die Gegenstelle NAT-T unterstützen. Oder aber der NAT-Router muss das IPsec-Protokoll kennen (IPsec/VPN Passthrough).

NAT-Traversal (NAT-T)

Eventuell befindet sich zwischen dem Gerät und dem VPN-Gateway des entfernten Netzes ein NAT-Router. Nicht alle NAT-Router lassen IPsec-Telegramme passieren. Daher kann es erforderlich sein, die IPsec-Telegramme in UDP-Pakete einzukapseln, um den NAT-Router passieren zu können.

Dead Peer Detection

Voraussetzung ist, dass die VPN-Gegenstelle DPD unterstützt. DPD prüft, ob die Verbindung noch störungsfrei arbeitet oder ob es eine Unterbrechung auf der Strecke gab. Ohne DPD muss je nach Konfiguration bis zum Ablauf der SA-Lebensdauer gewartet oder die Verbindung manuell neu initiiert werden. Um zu prüfen, ob die IPsec-Verbindung noch störungsfrei arbeitet, sendet das Gerät selber DPD-Anfragen zur VPN-Gegenstelle. Wenn die VPN-Gegenstelle nicht antwortet, wird nach Ablauf einer bestimmten Zeitspanne die Verbindung zur VPN-Gegenstelle für ungültig erklärt. Die Einstellungen für DPD konfigurieren Sie bei der Phase 1.

OpenVPN

Mit OpenVPN lassen sich virtuelle private Netzwerke (VPN) aufbauen. Das Gerät kann als OpenVPN-Client eine VPN-Verbindung zu einem entfernten Netzwerk aufbauen.

Den OpenVPN-Client konfigurieren Sie unter "Security" > "OpenVPN-Client".

Die VPN-Verbindung wird über virtuelle Gerätetreiber hergestellt, dem TAP- und TUN-Device. Dabei werden virtuelle Netzwerkschnittstellen angelegt, die wie eine physische Schnittstelle des Geräts wirken und den Endpunkt des VPN-Tunnels darstellen.

Das Gerät unterstützt Folgendes:

- TUN-Device: Routing-Modus
  Die LAN-Schnittstelle und die virtuelle Netzwerkschnittstelle befinden sich in verschiedenen IP-Subnetzen. Der virtuellen Tunnelschnittstelle wird vom OpenVPN-Server eine virtuelle IP-Adresse aus einem erdachten Subnetz zugewiesen. Die IP-Pakete (Layer 3) werden zwischen der virtuellen Tunnelschnittstelle und der LAN-Schnittstelle geroutet.

Authentifizierungsverfahren

- Zertifikate: CA-Zertifikat und Gerätezertifikat
  Die Verwendung von Zertifikaten ist ein asymmetrisches Kryptosystem. Jeder Teilnehmer (Gerät) besitzt einen geheimen, privaten Schlüssel und einen öffentlichen Schlüssel der Gegenstelle. Der private Schlüssel ermöglicht es, sich zu authentisieren und digitale Signaturen zu erzeugen.
- Benutzername/Passwort
  Der Zugang wird über einen Benutzernamen und ein Passwort beschränkt.

Verschlüsselungsverfahren

Das Gerät unterstützt dabei die folgenden Verfahren:

- BF CBC
- AES128 CBC
- AES192 CBC
- AES256 CBC
- DES EDE3

VPN-Verbindungsaufbau

Das Gerät unterstützt folgende Möglichkeiten, um eine VPN-Verbindung aufzubauen.

- OpenVPN: Security > OpenVPN > Verbindungen
- IPsec VPN: Security > IPsec VPN > Verbindungen
- SINEMA RC: System > SINEMA RC

Möglichkeiten	Verwendung			Beschreibung
	OpenVPN	IPsec VPN	SINEMA RC [1]	
Starten	x	x	-	Das Gerät ist "aktiv", d. h. es wird versucht, zu einer Gegenstelle eine Verbindung herzustellen. Die Adressierung der Gegenstelle erfolgt über deren projektierte WAN-IP-Adresse oder den projektierten FQDN.
Warten	-	x	-	Das Gerät ist "passiv", d. h. es wird gewartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
Auf Anforderung	-	x	-	Das Gerät versucht bei Bedarf zu einer Gegenstelle eine Verbindung herzustellen. Die Annahme von Anfragen zum VPN-Verbindungsaufbau ist ebenfalls möglich. Für die projektierten lokalen und entfernten Subnetze wird in der Routing-Tabelle ein Eintrag angelegt. Wenn ein Teilnehmer aus einem der entsprechenden Netze versucht Datenpakete über den VPN-Tunnel zu senden, wird die VPN-Verbindung aufgebaut. Der einstellbare Timeout bewirkt, dass nach dieser Zeit ohne weitere Datenpakete der VPN-Tunnel wieder abgebaut wird.
Bei DI starten	x	x	x	Der Verbindungsaufbau wird über den digitalen Eingang (DI) gesteuert
Bei DI warten	-	x	-	
Digitaler Eingang & Weck-SMS	-	-	x	Der Verbindungsaufbau wird über den digitalen Eingang (DI) oder SMS gesteuert.
Weck-SMS	-	-	x	Der Verbindungsaufbau wird über SMS gesteuert
Bei SMS starten	x	x	-	
Auf SMS warten	-	x	-	
Auto	-	-	x	Das Gerät übernimmt die Einstellungen des SINEMA RC Server. Die Einstellungen auf dem SINEMA RC Server konfigurieren Sie unter "Fernverbindungen > Geräte". Weiterführende Informationen hierzu finden Sie in der Betriebsanleitung "SINEMA RC Server".
Permanent	-	-	x	Das Gerät baut eine VPN-Verbindung zum SINEMA RC-Server. Der VPN-Tunnel wird permanent aufrecht erhalten.

[1] KEY-PLUG SINEMA REMOTE CONNECT notwendig

Digitaler Eingang (DI)

Der Aufbau des VPN-Tunnels kann auch über den digitalen Eingang gesteuert werden, z. B. über einen Taster. Wenn der Taster schließt, liegt am digitalen Eingang Spannung an und die LED vom digitalen Eingang leuchtet auf. Die aufleuchtende LED signalisiert, dass das Signal 1 (TRUE / HIGH) anliegt. Das Signal 1 löst auf dem Gerät ein Ereignis aus, mit dem der Aufbau des VPN-Tunnels gesteuert wird. Informationen zum Anschließen und zur maximalen Strombelastbarkeit finden Sie in der Betriebsanleitung der Geräte.

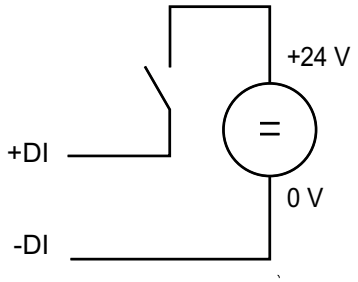

Voraussetzung

- Unter "System > Ereignisse > Konfiguration" ist beim Ereignis "Digitaler Eingang" "VPN-Tunnel" aktiviert.
  Wenn diese Einstellung nicht aktiviert ist, wird das Ereignis nicht an die VPN-Verbindung weitergegeben.

Möglichkeiten

Das Gerät unterstützt folgende Möglichkeiten zur Steuerung des VPN-Tunnels über den digitalen Eingang:

- Bei DI starten
  Wenn das Ereignis "Digitaler Eingang" eintritt, wird das Gerät "aktiv". Das Gerät versucht, zu einer Gegenstelle eine VPN-Verbindung (OpenVPN, IPsec, SINEMA RC) herzustellen. Beim SCALANCE M804PB wird der integrierte TIA Portal Cloud Connector gestartet.
- Bei DI warten
  Wenn das Ereignis "Digitaler Eingang" eintritt, ist das Gerät "passiv". Das Gerät wartet, bis der Verbindungsaufbau von der Gegenstelle (OpenVPN, IPsec) initiiert wird.
- Digitaler Eingang & Weck-SMS (nur bei M87x)
  Wenn das Ereignis "Digital In" eintritt oder das Gerät eine Wake-up SMS erhält, wird das Gerät "aktiv". Das Gerät versucht, zum SINEMA RC-Server eine VPN-Verbindung herzustellen.

SMS (nur bei M87x)

Voraussetzung

- Unter "System > SMS > Befehl-SMS" legen Sie fest, von wem eine Befehl-SMS der Klasse "System" akzeptiert wird.
- Unter "System > SMS > Ereignis-SMS" ist die Funktion aktiviert und die Telefonnummer eines Empfängers konfiguriert.
- Wenn diese Einstellung nicht aktiviert ist, wird das Ereignis nicht an die VPN-Verbindung weitergegeben.

Möglichkeiten

Das Gerät unterstützt folgende Möglichkeiten zur Steuerung des VPN-Tunnels über eine SMS:

- Bei SMS starten
  Wenn das Gerät eine Befehl-SMS erhält, wird das Gerät "aktiv". Die Befehl-SMS enthält den Namen der VPN-Verbindung (OpenVPN, IPsec). Das Gerät versucht zu der Gegenstelle eine Verbindung aufzubauen, die in der VPN-Verbindung angegeben ist.
  Format der SMS:
  Aufbau der VPN-Verbindung:
  - OpenVPN: SYS OVPN UP <Name der OpenVPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der OpenVPN-Verbindung> ein '*' eingeben, werden die OpenVPN-Verbindungen aufgebaut, die bei "Betrieb" "Bei SMS starten" eingestellt haben.
  - IPsec VPN: SYS IPSEC UP <Name der IPsec VPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der IPsec VPN-Verbindung> ein '*' eingeben, werden die IPsec VPN-Verbindungen aufgebaut, die bei "Betrieb" "Bei SMS starten" eingestellt haben.

  Abbau der VPN-Verbindung:
  - OpenVPN: SYS OVPN DOWN <Name der OpenVPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der OpenVPN-Verbindung> ein '*' eingeben, werden die OpenVPN-Verbindungen beendet, die bei "Betrieb" "Bei SMS starten" eingestellt haben.
  - IPsec VPN: SYS IPSEC DOWN <Name der IPsec VPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der IPsec VPN-Verbindung> ein '*' eingeben, werden die IPsec VPN-Verbindungen beendet, die bei "Betrieb" "Bei SMS starten" eingestellt haben.
- Auf SMS warten
  Wenn das Gerät eine Befehl-SMS erhält, ist das Gerät "passiv". Die Befehl-SMS enthält den Namen der IPsec VPN-Verbindung. Das Gerät wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird, die in derIPsec VPN-Verbindung angegeben ist.
  Format der SMS:
  Wenn Sie bei <Name der IPsec VPN-Verbindung> ein '*' eingeben, werden die IPsec VPN-Verbindungen aufgebaut, die bei "Betrieb" "Bei SMS starten" eingestellt haben.
  - IPsec VPN: SYS IPSEC UP <Name der IPsec VPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der IPsec VPN-Verbindung> ein '*' eingeben, werden die IPsec VPN-Verbindungen aufgebaut, die bei "Betrieb" "Bei SMS starten" eingestellt haben.
  - IPsec VPN: SYS IPSEC DOWN <Name der IPsec VPN-Verbindung>, siehe Befehl-SMS.
    Wenn Sie bei <Name der IPsec-VPN-Verbindung> ein '*' eingeben, werden die IPsec VPN-Verbindungen beendet, die bei "Betrieb" "Auf SMS warten" eingestellt haben.

- Digitaler Eingang & Weck-SMS
  Die Einstellungen des SINEMA RC-Servers werden ignoriert. Wenn das Ereignis "Digital In" eintritt oder das Gerät eine Weck-SMS erhält. Das Gerät versucht die VPN-Verbindung zum SINEMA RC-Server aufzubauen.
  Format der SMS: SYS SRC UP <Adresse des SINEMA RC-Servers>, siehe Befehl-SMS.
- Weck-SMS
  Die Einstellungen des SINEMA RC-Servers werden ignoriert. Wenn das eine Weck-SMS erhält, versucht das Gerät die VPN-Verbindung zum SINEMA RC-Server aufzubauen.
  Format der SMS:
  – SYS SRC UP <Adresse des SINEMA RC-Servers>, siehe Befehl-SMS.

Benachrichtigungsmöglichkeiten

Wenn sich der Status des digitalen Eingangs oder eines VPN-Tunnels (IPsec, OpenVPN, SINEMA RC) ändert, bietet das Gerät auf der Seite "Events" mehrere Möglichkeiten der Benachrichtigung.

Art der Benachrichtigung	Digitaler Eingang	VPN Tunnel	Verhalten bei Statusänderung
E-Mail	x	x	Das Gerät sendet eine E-Mail. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Voraussetzung: • Ein SMTP-Server ist eingerichtet. • Unter "System > SMTP-Client" ist die Funktion aktiviert, ein Empfänger und die IP-Adresse des SMTP-Servers konfiguriert.
Trap	x	x	Das Gerät löst einen SNMP-Trap aus. Voraussetzung: • Unter "System > Konfiguration" ist "SNMPv1 Traps" aktiviert. • Unter "System > Konfiguration > Traps" ist ein Empfänger konfiguriert, an den das Gerät die SNMP-Traps sendet.
Log-Tablle	x	x	Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle. Der Inhalt der Ereignisprotokoll-Tabelle wird unter "Information > Log Table" angezeigt.
Syslog	x	x	Das Gerät schreibt einen Eintrag auf den Syslog-Server. Voraussetzung: • Ein Syslog-Server ist eingerichtet. • Unter "System > Syslog-Client" ist die Funktion aktiviert und die IP-Adresse des Syslog-Servers konfiguriert.
Fehler-LED	x	-	Die Fehler-LED am Gerät leuchtet auf.

Art der Benachrichtigung	Digitaler Eingang	VPN Tunnel	Verhalten bei Statusänderung
Digitaler Eingang	x	x	Steuert den digitalen Ausgang an oder signalisiert die Zustandsänderung mit der LED "DO". An dem digitalen Ausgang kann ein Verbraucher angeschlossen werden. Informationen zum Anschließen finden Sie in der Betriebsanleitung der Geräte. Der Verbraucher signalisiert eine Statusänderung. 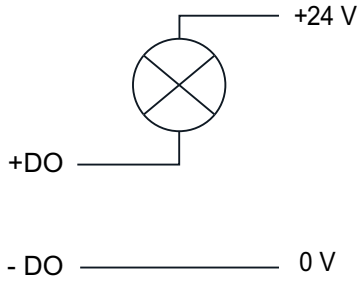 **Hinweis** Über CLI und über SNMP können Sie den digitalen Ausgang direkt ansteuern. Im WBM und CLI können Sie bei den "Events" die Verwendung des digitalen Ausgangs projektieren. Steuern Sie den digitalen Ausgang nicht direkt an, wenn Sie diesen im WBM und CLI verwenden.
SMS (nur bei M87x)	x	x	Das Gerät sendet eine SMS mit einem Standardtext. Voraussetzung: • Unter "System > SMS > Ereignis-SMS" ist die Funktion aktiviert, die Telefonnummer eines Empfängers konfiguriert.
	x	-	Das Gerät sendet eine SMS mit einem benutzerdefinierten Text. Den Text konfigurieren Sie unter "System > SMS > Ereignis-SMS". Voraussetzung: • Unter "System > SMS > Ereignis-SMS" ist die Funktion aktiviert, die Telefonnummer eines Empfängers und der Text konfiguriert. • Unter "System > Ereignisse > Konfiguration" ist beim Ereignis "Digitaler Eingang" "SMS" aktiviert.
Status aus der MIB-Variable auslesen	x	-	Über die private MIB-Variable snMspsDigitalInputLevel können Sie den Status des digitalen Eingangs auslesen: • OID der privaten MIB-Variable snMspsDigitalInputLevel: `iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).siemens(4329).industrialComProducts(20).iComPlatforms(1).simaticNet(1).snMsps(1).snMspsCommon(1).snMspsDigitalIO(39).snMspsDigitalIOObjects(1).snMspsDigitalInputTable(2).snMspsDigitalInputEntry(1).snMspsDigitalInputLevel(6)` • Werte der MIB-Variable – 1: Signal 0 am digitalen Eingang (DI) – 2: Signal 1 am digitalen Eingang (DI)

Informationen anzeigen

Versionen

Die Seite zeigt die Ausgabestände der Hardware und der Software für das Gerät an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Hardware**
  - Basic Device
    Zeigt das Grundgerät an
  - CELL (nur bei M87x)
    Zeigt das verwendete Funkmodul an.
  - xDSL (nur bei M81x)
    Zeigt das verwendete DSL-Modul
- **Name**
  Zeigt den Namen des Geräts.
- **Ausgabestand**
  Zeigt den Hardware-Ausgabestand des Geräts an.
- **Artikelnummer**
  Zeigt die Artikelnummer des Geräts an.
- **Software**
  - Firmware
    Zeigt die aktuelle Firmware-Version an. Wenn eine neue Firmware-Datei geladen wurde und das Gerät noch nicht neu gestartet ist, wird hier die Firmware-Version der geladenen Firmware-Datei angezeigt. Nach dem nächsten Neustart wird die geladene Firmware aktiviert und verwendet.
  - Bootloader
    Zeigt die Version der Boot-Software an, die im Gerät gespeichert ist.
  - Firmware_Running
    Zeigt die Firmware-Version an, die aktuell vom Gerät verwendet wird.
- **Beschreibung**
  Zeigt die Kurzbeschreibung der Software an.
- **Version**
  Zeigt die Versionsnummer des Software-Ausgabestands an.
- **Datum**
  Zeigt das Erstellungsdatum des Software-Ausgabestands an.

I&M

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite beinhaltet Informationen zu gerätespezifischen Hersteller- und Wartungsdaten wie Artikelnummer, Seriennummer, Versionsnummern etc. Sie können auf dieser Seite keine Konfigurationen vornehmen.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Zeilen:

- **Hersteller-ID**
  Zeigt die Herstellerkennung an.
- **Artikelnummer**
  Zeigt die Artikelnummer an.
- **Seriennummer**
  Zeigt die Seriennummer an.
- **Hardware-Ausgabestand**
  Zeigt den Hardware-Ausgabestand an.
- **Software-Ausgabestand**
  Zeigt den Software-Ausgabestand an.
- **Versionszähler**
  Zeigt den Versionsänderungs-Zähler an: Zähler für Versionsänderungen seit der Erst-Inbetriebnahme
- **Aktualisierungsdatum**
  Datum und Uhrzeit der letzten Versionsänderung
- **Funktionskennzeichen**
  Zeigt das Funktionskennzeichen (Anlagenkennzeichen) des Geräts an. Das Anlagenkennzeichen (AKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Ortskennzeichen**
  Zeigt das Ortskennzeichen des Geräts an. Das Ortskennzeichen (OKZ) wird bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt.
- **Datum**
  Zeigt das Datum, das bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.
- **Deskriptor**
  Zeigt die Beschreibung, die bei der Projektierung des Geräts mit HWKonfig von STEP7 angelegt wurde.

ARP-Tabelle

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zuordnung von MAC-Adresse und IP-Adresse

Über das Address Resolution Protocol (ARP) erfolgt die eindeutige Zuordnung von MAC-Adresse zu IPv4-Adresse. Dieser Zuordnung wird von jedem Netzteilnehmer in seiner eigenen ARP-Tabelle gepflegt. Die Seite zeigt die ARP-Tabelle des Geräts.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an, über die der Zeileneintrag gelernt wurde.
- **MAC-Adresse**
  Zeigt die MAC-Adresse des Ziel- oder Quellgeräts an.
- **IP-Adresse**
  Zeigt die IP-Adresse des Zielgeräts an.
- **Medientyp**
  Zeigt die Art der Verbindung.
  - Dynamisch
    Das Gerät hat die Adressdaten automatisch erkannt.
  - Statisch
    Die Adressen wurden als statische Adressen eingetragen.

Log-Tabellen

Ereignis-Log

Protokollierung von Ereignissen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Das Gerät bietet die Möglichkeit, auftretende Ereignisse zu protokollieren, die Sie zum Teil unter "System > Ereignisse" festlegen können. So kann beispielsweise festgehalten werden, wann ein Authentifizierungsversuch fehlgeschlagen ist oder wann sich der Verbindungsstatus eines Ports geändert hat. Der Inhalt der Ereignisprotokoll-Tabelle bleibt auch nach dem Ausschalten des Geräts erhalten.

Beschreibung der angezeigten Werte

- **Severity-Filter**

  Hinweis

  Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

  Sie können die Einträge der Tabelle nach Fehlerschwere filtern. Wählen Sie in den Optionskästchen oberhalb der Tabelle die gewünschten Einträge aus.

  – Info
    Information

  – Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Info" angezeigt.

  – Warning
    Warnung
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Warning" angezeigt.

  – Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorie "Critical" angezeigt.

  Die Tabelle gliedert sich in folgende Spalten:

  – **Neustart**
    Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis aufgetreten ist.

  – **Systembetriebszeit**
    Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.

  – **Systemzeit**
    Zeigt das Datum und die Uhrzeit an, zu der das beschriebene Ereignis aufgetreten ist. Wenn die Systemzeit gesetzt ist, wird auch die Zeit angezeigt, bei der das Ereignis eingetreten ist.

  – **Severity**
    Zeigt den Schweregrad der Meldung an.

  – **Log-Meldung**
    Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an. Eine Liste der möglichen Meldungen finden Sie im Anhang D des Projektierungshandbuches..

Security-Log

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt in tabellarischer Form die Ereignisse an, die bei der Kommunikation über einen gesicherten VPN-Tunnel aufgetreten sind.

Beschreibung der angezeigten Werte

- **Severity-Filter**

  Hinweis

  Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

  Die Einträge der Tabelle können Sie nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.

  – 6 - Info
    Informativ
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Info" angezeigt.
  – 4 - Warning
    Warnend
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Warning" angezeigt.
  – 2 - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis eingetreten ist.
- **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.
- **Systemzeit**
  Wenn die Systemzeit gesetzt ist, werden Datum und Uhrzeit angezeigt, bei der das Ereignis eingetreten ist. Wenn keine Systemzeit eingestellt ist, enthält das Feld die Angabe "Date/time not set"
- **Severity**
  Einordnung des Eintrags in obige Kategorien.
- **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an.

Firewall-Log

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Das Firewall-Logbuch protokolliert die Ereignisse, die an der Firewall eingetreten sind. Beim Anlegen von Firewall-Regeln können Sie festlegen, mit welcher Ereignisschwere diese protokolliert werden.

Beschreibung der angezeigten Werte

- **Severity-Filter**

  Hinweis

  Pro Severity sind maximal 400 Einträge in der Tabelle möglich. Wenn bei einer Severity die maximale Anzahl der Einträge ereicht ist, werden die ältesten Einträge dieser Severity in der Tabelle überschrieben. Die Tabelle verbleibt permanent im Speicher

  Die Einträge der Tabelle können Sie nach Schweregrad filtern. Um alle Einträge anzuzeigen, aktivieren oder deaktivieren Sie alle Parameter.

  – 6 - Info
    Informativ
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Info" angezeigt.
  – 4 - Warning
    Warnend
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Warning" angezeigt.
  – 2 - Critical
    Kritisch
    Wenn dieser Parameter aktiviert ist, werden alle Einträge der Kategorien "Critical" angezeigt.

Die Tabelle gliedert sich in folgende Spalten:

- **Neustart**
  Zählt die Anzahl der Neustarts seit dem letzten Zurücksetzen auf Werkseinstellungen und gibt an, nach welchem Neustart des Geräts das entsprechende Ereignis eingetreten ist.
- **Systembetriebszeit**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der das beschriebene Ereignis eingetreten ist.
- **Systemzeit**
  Wenn die Systemzeit gesetzt ist, werden Datum und Uhrzeit angezeigt, bei der das Ereignis eingetreten ist. Wenn keine Systemzeit eingestellt ist, enthält das Feld die Angabe "Date/time not set"

- **Severity**
  Einordnung des Eintrags in obige Kategorien.
- **Log-Meldung**
  Zeigt eine Kurzbeschreibung des eingetretenen Ereignisses an.

Fehler

Fehlerstatus

Wenn ein Fehler auftritt, wird er auf dieser Seite angezeigt. Am Gerät werden Fehler durch das Leuchten der Fehler-LED signalisiert.

Gemeldet werden interne Fehler des Geräts sowie die Fehler, die Sie auf folgenden Seiten konfigurieren:

- "System > Ereignisse"
- "System > Fehlerkontrolle"

Die Berechnung des Fehlerzeitpunkts beginnt jeweils nach dem letzten Systemstart.

Wenn keine Fehler vorliegen, schaltet sich die Fehler-LED ab.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Angezeigte Werte

- **Anzahl der gemeldeten Fehler**
  Zeigt an, wie oft die Fehler-LED angegangen ist und nicht wieviele Fehler aufgetreten sind.
- **Schaltfläche "Zähler zurücksetzen"**
  Über die Schaltfläche wird die Anzahl zurückgesetzt. Der Zähler wird durch einen Neustart zurückgesetzt.

Die Tabelle enthält die folgenden Spalten:

- **Fehlerzeitpunkt**
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an, zu der der beschriebene Fehler aufgetreten ist.
- **Fehlerbeschreibung**
  Zeigt eine Kurzbeschreibung des aufgetretenen Fehlers an.
- **Fehlerstatus löschen**
  Wenn die Schaltfläche "Fehlerstatus löschen" aktiv ist, können Sie den Fehler löschen, z. B. Fehler des Ereignisses "Kalt-/Warmstart".

DHCP-Server

Diese Seite zeigt an, welche IPv4-Adressen vom DHCP-Server den Geräten zugeordnet wurden.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

- **IP-Adresse**
  Zeigt die IPv4-Adresse an, die dem DHCP-Client zugeordnet ist.
- **Pool-ID**
  Zeigt die Nummer des IPv4-DHCP-Pools an.
- **Identifikationsmethode**
  Zeigt die Methode an, nach der der DHCP-Client identifiziert wird.
  - Remote-ID
    Zeigt die Remote-ID des DHCP-Clients an.
  - Circuit-ID
    Zeigt die Circuit-ID des DHCP-Clients an.
  - DUID
    Zeigt die DUID des DHCP-Clients an.
- **Identifikationswert**
  Zeigt die MAC-Adresse oder die Client-ID des DHCP-Clients an.
- **Zuordnungsmethode**
  Zeigt an, ob die IPv4-Adresse statisch oder dynamisch vergeben wurde. Die statischen Einträge konfigurieren Sie unter "System > DHCP > Statische Zuordnung".
- **Zuordnungsstatus**
  Zeigt den Status der Zuordnung an.
  - Zugeordnet
    Die Zuordnung wird verwendet.
  - Nicht verwendet
    Die Zuordnung wird nicht verwendet.
  - Wird geprüft
    Die Zuordnung wird geprüft.
  - Unbekannt
    Der Status der Zuordnung ist unbekannt.
- **Ablaufzeit**
  Zeigt an, wie lange die vergebene IPv4-Adresse noch gültig ist. Nachdem die Gültigkeitsdauer zur Hälfte abgelaufen ist, kann der DHCP-Client die vergebene IPv4-Adresse verlängern. Nach Ablauf der gesamten Zeitdauer muss der DHCP-Client eine neue IPv4-Adresse anfordern.

SNMP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die angelegten SNMPv3-Gruppen. Die SNMPv3-Gruppen konfigurieren Sie unter "System > SNMP".

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt den Gruppennamen an.
- **Benutzername**
  Zeigt den Benutzer an, welcher der Gruppe zugeordnet ist.

LLDP

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Status der Nachbarschaftstabelle

Diese Seite zeigt den aktuellen Inhalt der Nachbarschaftstabelle. In dieser Tabelle sind die Informationen gespeichert, die der LLDP-Agent von angeschlossenen Geräten empfangen hat. Über welche Schnittstellen der LLDP-Agent Informationen empfängt bzw. versendet, legen Sie in folgendem Kapitel fest: "Layer 2 > LLDP".

Beschreibung der angezeigten Werte

Die Tabelle enthält die folgenden Spalten:

- **Systemname**
  Systemname des angeschlossenen Geräts.
- **Geräte-ID**
  Gerätekennung des angeschlossenen Geräts. Die Geräte-ID entspricht dem Gerätenamen, der über PST (STEP 7) vergeben wird. Wenn kein Gerätename vergeben ist, wird die MAC-Adresse des Geräts angezeigt.
- **Lokale Schnittstelle**
  Der Port, an dem das Gerät die Informationen empfangen hat.
- **Speicherzeit**
  Ein Eintrag bleibt für die hier angegebene Zeit im Gerät gespeichert. Wenn das Gerät in dieser Zeit keine neuen Informationen von dem angeschlossenen Gerät erhält, wird der Eintrag gelöscht.

- **Eigenschaft**
  Zeigt die Eigenschaften des angeschlossenen Geräts an:
  - Router
  - Bridge
  - Telephone
  - DOCSIS Cable Device
  - WLAN Access Point
  - Repeater
  - Station
  - Other
- **Port-ID**
  Port des Geräts, der mit dem das Gerät verbunden ist.

Routing

Einleitung

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Diese Seite zeigt die Routen an, die aktuell verwendet werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**
  Zeigt die Zieladresse dieser Route an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske dieser Route an.
- **Gateway**
  Zeigt das Gateway für diese Route an.
- **Schnittstelle**
  Zeigt die Schnittstelle für diese Route an.

- **Metrik**
  Zeigt die Metrik der Route an. Je größer der Wert, desto länger benötigen Pakete zu Ihrem Ziel.
- **Routing-Protokoll**
  Zeigt an, aus welchem Routing-Protokoll der Eintrag der Routingtabelle stammt. Folgende Einträge sind möglich:
  - Connected: Verbundene Routen
  - Static: Statische Routen

Mobilfunk

Mobilfunk Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt einen Überblick über den aktuellen Betriebsstatus des Geräts.

Beschreibung der angezeigten Werte

Die Seite enthält Folgendes:

- **IMEI**
  Zeigt die IMEI-Nummer des verwendeten Funkmoduls an.
  Die IMEI (= International Mobile Equipment Identity) wird weltweit eindeutig vergeben.
- **SIM-Status**
  Zeigt den Status der SIM-Karte an.
  - Vorhanden
    SIM-Karte vorhanden
  - Fehlend
    SIM-Karte fehlt
- **PIN-Status**
  Zeigt den Status der PIN an.
  - PIN erforderlich
    PIN muss noch eingegeben werden.
  - Keine PIN erforderlich
    PIN wird nicht benötigt.
  - PUK erforderlich
    SIM-Karte gesperrt. Mit der PUK kann die Karte wieder entsperrt werden.
  - Falsche PIN
    Die PIN ist falsch.
  - PIN gültig
    Die PIN ist gültig.

- IMSI
  Zeigt die Teilnehmerkennung an, die auf der verwendeten SIM-Karte gespeichert ist.
- Rufnummer
  Zeigt die Rufnummer der SIM-Karte an.
  Wenn die Rufnummer nicht vorhanden ist oder nicht ausgelesen werden kann, wird "no number" angezeigt.
- Verbindungsstatus
  Zeigt an, ob eine Funkverbindung besteht, und falls ja, welche:
  – UMTS: IP-Verbindung über UMTS oder HSPA
  – GPRS: IP-Verbindung über EGPRS, GPRS oder LTE
- Telegramm Switch-Status
  Zeigt den Status der Paketvermittlung an.
- Funkzellen-ID
  Zeigt die Kennung der Funkzelle an, in welches sich das Gerät eingebucht hat.
- LAC (Location Area Code)
  Zeigt die Kennung für den aktuellen Aufenthaltsbereich des M87x innerhalb des Mobilfunknetzs an.
- Signalstärke
  Zeigt die Signalstärke an.

dBm	
< -109	Keine Verbindung
-109 ... -89	Schlechte Signalstärke
-89 …-73	Mittlere Signalstärke
> -73	Gute Signalstärke

- Mobilfunk Chip-Temperatur
  Zeigt die aktuelle Temperatur des verbauten Chips an.
- Anbieter
  Zeigt den Mobilfunkbetreiber an.
- APN
  Zeigt den verwendeten APN (Access Point Name) der Funkverbindung an.
- Externe IP-Adresse
  Zeigt die WAN-IP-Adresse an, unter der das Gerät im Mobilfunknetz zu erreichen ist. Die WAN-IP-Adresse wird dem Gerät vom Dienst des Mobilfunkbetreibers zugewiesen.
- DNS-Server
  Zeigt den oder die vom Gerät verwendeten DNS-Server an.

Signalrekorder

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Mit dem Signalrekorder wird die Signalstärke zur Funkzelle angezeigt, in der das Gerät eingebucht ist.

Beschreibung der angezeigten Werte

Die Grafik enthält die folgenden Elemente:

- Scrollbalken
  Mit dem Scrollbalken können Sie die gesamte Messung durchschauen. Dafür können Sie die Schaltflächen "<<" und ">>" oder die Pfeiltasten auf der Tastatur verwenden.
- Balken (links)
  In dem Balken auf der linken Seite wird die Signalstärke in Echtzeit angezeigt, entsprechend des abgebildeten Farbschemas.
- Farbschema
  Grün = Gut, Gelb = Mittel, Orange = Schwach, Rot = Sehr schwach
- x-Achse
  Die x-Achse zeigt den Verlauf der Messung in Sekunden.
- Verlaufsdaten
  Die Verlaufsdaten zeigen den Wert der Signalstärke entsprechend des abgebildeten Farbschemas.

Angezeigte Stichproben

Wählen Sie aus, wie viele Messwerte in der Grafik angezeigt werden sollen.

SHDSL

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt den Status der SHDSL-Schnittstelle an.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Schnittstelle**
  Zeigt die SHDSL-Schnittstellen des Geräts.
- **Negotiation**
  Folgende Werte sind möglich:
  - down not ready (Nur bei der Rolle CO)
    Zum Kommunikationspartner ist keine elektrische Verbindung gesteckt. Der Treiber wird initialisiert und ist bereit zum Aushandeln der Verbindungsparameter.
  - down ready (nur bei der Rolle CPE)
    Zum Kommunikationspartner ist keine elektrische Verbindung gesteckt. Der Treiber wird initialisiert und ist bereit zum Aushandeln der Verbindungsparameter.
  - initialising
    Die Verbindungsparameter werden ausgehandelt.
  - up data mode
    Die Verbindungsparameter sind erfolgreich ausgehandelt. Die Datenübertragung ist möglich.
  - stop down ready
    Das Aushandeln der Verbindungsparameter war nicht erfolgreich oder die Verbindung wurde unterbrochen. Neuaushandlung der Parameter steht an.
  - unknown
    Die Treiberinitialisierung noch nicht abgeschlossen. Dieser Status tritt nach dem Gerätestart auf.
- **Datenrate [kbps]**
  Zeigt die nutzbare Datenübertragungsrate an.
- **Betriebszeit**
  Zeigt die Zeit seit dem letzten Aushandeln der Verbindungsparameter bzw. der letzten Verbindungsunterbrechung an.
- **SNR [dB]**
  Zeigt das Signal-Rausch-Verhältnis an. Je größer der Wert, desto besser das Signal.
- **CRC Anomalie**
  Cyclic Redundancy Check
  Zeigt an, wie viele SHDSL-Netzwerkpakete mit fehlerhafter Prüfsumme (CRC) entdeckt wurden.
- **LOSW Defekt**
  Loss of Synchronization Word
  Zeigt an, wie oft die Synchronisation auf der SHDSL-Verbindung verloren ging.
- **Negotiation-Zähler**
  Zeigt an, wie oft die Verbindungsparameter neu ausgehandelt wurden.

DSL

DSL Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Eine ADSL-Verbindung besteht aus zwei Endpunkten, den sogenannten ADSL Transceiver Units (ATU). Die ADSL Transceiver Unit-Remote (ATU-R) ist dieses Gerät und die ADSL Transceiver Unit-Central (ATU-C) ist das Gerät in der Vermittlungsstelle.

Die Seite zeigt den aktuellen Betriebsstatus des Geräts und enthält Informationen zur ATU-C.

Beschreibung der angezeigten Werte

Die Seite enthält Folgendes:

- **Modem-Status**
  Zeigt den Status des Modems an.
  Z. B. beim Initialisieren und beim Aufbau der ADSL-Verbindung werden folgende Stati angezeigt:
  - showtime-sync
    ADSL-Verbindung ist aufgebaut. Datenverkehr möglich.
  - handshake
    ADSL-Verbindung wird geprüft und aufgebaut.
  - full-init
    Modem wird initialisiert.
  - silent
    Keine ADSL-Verbindung
- **Latenz-Typ**
  Zeigt an, welches Verfahren verwendet wird.
  - fast
    Fast Path: Kleine Antwortzeit. Geringere Störsicherheit.
  - slow
    Störsicherheit durch Interleaving. Beim Interleaving werden die Daten verschachtelt, dadurch verlängert sich die Antwortzeit.
  - unkown
    Verfahren unbekannt
- **Externe IP-Adresse**
  Zeigt die externe IP-Adresse an, unter der das Gerät zu erreichen ist. Die externe IP-Adresse wird dem Gerät vom Dienst des DSL-Anbieters zugewiesen.
- **DNS-Server**
  Zeigt den oder die vom Gerät verwendeten DNS-Server an

DSL -Datenrate

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite enthält eine Übersicht der Datenübertragungsraten für die Empfangsrichtung (Downstream) und für die Senderichtung (Upstream).

Beschreibung der angezeigten Werte

Die Seite enthält Folgendes:

- **Downstream Datenrate (kbps)**
  Zeigt die nutzbare Datenübertragungsrate für den Downstream an.
- **Downstream ATTNDR (kbps)**
  ATTNDR (Attainable data rate)
  Zeigt die maximal mögliche Datenübertragungsrate für den Downstream an.
- **Upstream Datenrate (kbps)**
  Zeigt die nutzbare Datenübertragungsrate für den Upstream an.
- **Upstream ATTNDR (kbps)**
  ATTNDR (Attainable data rate).
  Zeigt die maximal mögliche Datenübertragungsrate für den Upstream an.

DSL-Streams

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Die Seite zeigt Parameter an, die die Datenübertragungsrate beeinflusssen.

Beschreibung der angezeigten Werte

Für die Empfangsrichtung (Downstream) werden folgende Werte angezeigt:

- **Downstream Interleaver-Tiefe**
  Zeigt an, mit welcher Tiefe die Daten verschachtelt werden. Je höher die Interleaving-Tiefe, desto länger die Antwortzeit.
- **Downstream Leitungsdämpfung [dB]**
  Zeigt die Leitungsdämpfung an. Die Leitungsdämpfung ist abhängig vom Leitungsdurchmesser und der Übertragungsfrequenz.
- **Downstream Signaldämpfung [dB]**
  Zeigt die Signaldämpfung an. Je kleiner der Wert, desto besser das Signal.

- **Downstream Signal-Rausch-Verhältnis[dB]**
  Zeigt das Signal-Rausch-Verhältnis an. Je größer der Wert, desto besser das Signal.
- **Downstream Aktuell aggregierte Sendeleistung [dB]**
  Zeigt die aktuelle Sendeleistung an, die im aggregierten Betrieb erreicht wird.

Für die Senderichtung (Upstream) werden folgende Werte angezeigt

- **Upstream Interleaver -Tiefe**
  Zeigt an, mit welcher Tiefe die Daten verschachtelt werden. Je höher die Interleaving-Tiefe, desto länger die Antwortzeit.
- **Upstream Leitungsdämpfung [dB]**
  Zeigt die Leitungsdämpfung an. Die Leitungsdämpfung ist abhängig vom Leitungsdurchmesser und der Übertragungsfrequenz.
- **Upstream Signaldämpfung [dB]**
  Zeigt die Signaldämpfung an. Je kleiner der Wert, desto besser das Signal.
- **Upstream Signal-Rausch-Verhältnis [dB]**
  Zeigt das Signal-Rausch-Verhältnis an. Je größer der Wert, desto besser das Signal.
- **Upstream Aktuell aggregierte Sendeleistung [dB]**
  Zeigt die aktuelle Sendeleistung an, die im aggregierten Betrieb erreicht wird.

IPsec VPN

Die Seite zeigt den Status der aktivierten VPN-Verbindungen an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an.
- **Lokaler Host**
  Zeigt die IP-Adresse des Geräts an.
- **Lokale DN**
  Zeigt den Distinguished Name (DN) des Geräts an, der während des Verbindungsaufbaus an die Gegenstelle gemeldet wurde. Der Eintrag wird aus dem Feld "Local ID", dem Gerätezertifikat, oder der IP Adresse des Geräts übernommen.
- **Lokales Subnetz**
  Zeigt das lokale Netz an.
- **Remote-Host**
  Zeigt die IP-Adresse oder den Hostnamen der Gegenstelle an.
- **Remote-DN**
  Zeigt den Distinguished Name (DN) an, den die Gegenstelle beim Verbindungsaufbau gemeldet hat.

- **Remote Subnetz**
  Zeigt das entfernte Netz an.
- **Schlüssel-Lebensdauer**
  Zeigt an, wann die Gültigkeit des Schlüssels abläuft.
- **Status**
  Zeigt den Status der VPN-Verbindung an.

SINEMA RC

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Zeigt Informationen zum SINEMA RC-Server an.

Hinweis

Für SCALANCE S615 ist diese Funktion nur mit KEY PLUG nutzbar.

Beschreibung der angezeigten Werte

- **Status**
  Zeigt den Status der Verbindung zum SINEMA RC-Server an.
- **Name des Geräts**
  Wenn projektiert, wird der Name des Geräts angezeigt.
- **Gerätestandort**
  Wenn projektiert, wird der Standort des Geräts angezeigt
- **GSM-Nummer**
  Wenn projektiert, wird die Rufnummer des Geräts angezeigt
- **Hersteller**
  Wenn projektiert, wird der Eintrag angezeigt
- **Kommentar**
  Wenn projektiert, wird der Kommentar angezeigt
- **Verbindungsart (Server)**
  Zeigt an, welche Verbindungsart auf dem SINEMA RC-Server eingestellt ist.
- **Verbindungsart (Gerät)**
  Zeigt an, welche Verbindungsart auf dem Gerät eingestellt ist
- **Fingerabdruck**
  Zeigt den Fingerabdruck des Serverzertifikats an. Wird nur angezeigt, wenn zum Verifizieren der Fingerabdruck verwendet wird.
- **Remote-Adresse**
  Zeigt die IP-Adresse des SINEMA RC-Servers an.

- **Verbundene lokale Subnetze**
  Zeigt die IP-Adressen der lokalen Subnetze an. Wird nur angezeigt, wenn auf dem SINEMA RC-Server die Option "Verbundene lokale Subnetze" aktiviert ist. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- **Verbundene lokale Hosts**
  Zeigt die Ziel-IP-Adresse der Hosts an, die erreichbar sind.
- **Adresse Tunnel-Schnittstelle**
  Zeigt die IP-Adresse der virtuellen Tunnelschnittstelle an.
- **Verbundene Remote-Subnetze**
  Zeigt die Subnetze des SINEMA RC-Servers an, die für das Gerät erreichbar sind. Welche Subnetze für das Gerät erreichbar sind, ist von den Kommunikationsbeziehungen auf dem SINEMA RC-Server abhängig. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.

OpenVPN-Client

Die Seite zeigt den Status der aktivierten OpenVPN-Verbindungen an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der OpenVPN-Verbindung an.
- **Remote-Server**
  Zeigt die IP-Adresse oder den Hostnamen des OpenVPN-Servers an.
- **Tunnel-Schnittstelle IP**
  Zeigt die IP-Adresse der virtuellen Tunnelschnittstelle an.
- **Exportierte Subnetze**
  Zeigt die IP-Adresse der lokalen Subnetze an.
- **Geroutete Subnetze**
  Zeigt die Subnetze des Open VPN-Servers an.
- **Status**
  Zeigt den Status der OpenVPN-Verbindung an.

Redundanz

Übersicht

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein.
- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Port**
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten**
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge
- **Topologieänderungen / Letzte Topologieänderung**
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit des letzten Hochlaufs. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz "Sek." hinter der Zahlenangabe
  - Minuten: Zusatz "Min." hinter der Zahlenangabe
  - Stunden: Zusatz "Std." hinter der Zahlenangabe
- **Bridge Hello Time[s] / Root Hello Time[s]**
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei solchen Telegrammen ist die Hello-Time.
- **Bridge Forward Delay / Root Forward Delay**
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort, sondern erst nach dem im Parameter Weiterleitungsverzögerung festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben. Der
- **Bridge Max Age / Root Max Age**
  Nach Ablauf des Max Age-Timers wird die empfangene BPDU verworfen, um vom Switch als gültig akzeptiert zu werden. Der Defaultwert liegt bei 20s.

- **Bridge Max Hop Count**
  Dieser Parameter gibt an, wie viele STP-Teilnehmer eine BPDU passieren darf. Wird eine MSTP-BPDU empfangen, deren Hop Count den hier konfigurierten Wert übersteigt, wird sie verworfen.
- **Root Hop Count**
  Die Anzahl der Teilnehmer, die bis zur Root-Bridge durchlaufen werden müssen.

Spanning Tree

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Einleitung

Die Seite zeigt die aktuellen Informationen zu Spanning Tree und die Einstellungen der Root Bridge an.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- **Spanning Tree-Modus**
  Zeigt den eingestellten Modus an. Den Modus legen Sie bei "Layer 2 > Konfiguration" und bei "Layer 2 > Spanning Tree > Allgemein" fest.
  Folgende Werte sind möglich:
  - '-'
  - RSTP
- **Bridge-Priorität / Root-Priorität**
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 32768.
- **Bridge-Adresse / Root-Adresse**
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Switch an.
- Root-Kosten
  Zeigt die Pfadkosten von dem Gerät bis zur Root Bridge.
- Bridge-Status
  Zeigt den Status der Bridge an, z. B. ob das Gerät die Root-Bridge ist.

Die Tabelle gliedert sich in folgende Spalten:

- Port
  Zeigt die Schnittstellen an, über den das Gerät kommuniziert.
- Rolle
  Zeigt den Status des Ports an. Folgende Werte sind möglich:
  - Disabled
    Der Port wurde manuell aus dem Spanning Tree entfernt und wird vom Spanning Tree nicht mehr berücksichtigt.
  - Designated
    Die Ports, die von der Root-Bridge wegführen.
  - Alternate
    Der Port mit einem alternativen Weg zu einem Netzwerksegment
  - Backup
    Wenn ein Switch mehrere Ports zu dem gleichen Netzwerksegment hat, wird der "schlechtere" Port zum Backup-Port.
  - Root
    Der Port, der den besten Weg zur Root Bridge bietet.
  - Master
    Dieser Port zeigt zu einer Root-Bridge, die außerhalb der MST-Region liegt.
- Status
  Zeigt den aktuellen Status an, in dem sich die Schnittstelle befindet. Die Werte werden nur angezeigt. Der Parameter ist abhängig vom projektierten Protokoll.
  - Discarding
    Der Port empfängt BPDU-Telegramme. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    Der Port empfängt und sendet BPDU-Telegramme. Der Port ist in den Spanning Tree-Algorithmus einbezogen. Andere aus- und eingehende Telegramme werden verworfen.
  - Learning
    Der Port lernt aktiv die Topologie, d. h. die Teilnehmeradressen. Andere aus- und eingehende Telegramme werden verworfen.
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz. Der Port empfängt und sendet Datentelegramme.
- Oper. Version
  Zeigt den Kompatibilitätsmodus von Spanning Tree, der vom Port verwendet wird.
- Priorität
  Kann der vom Spanning-Tree ermittelte Weg alternativ über mehrere Ports eines Gerätes führen, so wird der Port mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) ausgewählt. Für die Priorität kann ein Wert von 0 bis 240 in 16er Schritte eingegeben werden. Wenn Sie einen Wert eingeben, der nicht durch 16 teilbar ist, wird der Wert automatisch angepasst. Der Standardwert ist 128.

- **Pfadkosten**
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Es wird die Strecke mit dem geringsten Wert als Weg ausgewählt. Haben mehrere Ports eines Gerätes den gleichen Wert, wird der Port mit der niedrigsten Portnummer ausgewählt.
  Ist der Wert im Feld "Cost Calc" "0", so wird der automatisch ermittelte Wert angezeigt. Im anderen Fall wird der Wert des Feldes "Cost Calc" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000
- **Edge-Typ**
  Zeigt den Typ der Verbindung an. Folgende Werte sind möglich:
  - Edge Port
    An diesem Port befindet sich ein Endgerät.
  - No Edge Port
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.
- **P.t.P. Type**
  zeigt die Art der Punkt-zu-Punkt-Verbindung an. Folgende Werte sind möglich:
  - P.t.P.
    Bei Halbduplex wird von einer Punkt-zu-Punkt-Verbindung ausgegangen.
  - Shared Media
    Bei einer Vollduplexverbindung wird nicht von einer Punkt-zu-Punkt-Verbindung ausgegangen.

VRRPv3-Statistiken

Einleitung

Diese Seite zeigt die Statistiken des VRRPv3-Protokolls und aller konfigurierten virtuellen Router an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung der angezeigten Werte

Folgende Felder werden angezeigt:

- VRID-Fehler
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine nicht unterstütze VRID enthalten.
- Versionsfehler
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Versionsnummer enthalten.
- Prüfsummenfehler
  Zeigt an, wie viele VRRPv3-Pakete empfangen wurden, die eine ungültige Prüfsumme enthalten.

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstellen
  Schnittstelle, auf die sich die Einstellungen beziehen.
- VRID
  Zeigt die ID des virtuellen Routers an. Gültige Werte sind 1 ... 255.
- Typ
  Zeigt die Version des IP-Protokolls an
- Übergang in Master-Zustand
  Zeigt an, wie häufig dieser virtuelle Router in den Zustand "Master" ging.
- Empf. Advertisements
  Zeigt an, wie viele VRRPv3-Paket empfangen wurden.
- Advertisement-Intervall-Fehler
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Intervall nicht mit dem lokal gesetzten Wert übereinstimmt.
- IP TTL Fehler
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren TTL (Time to live) Wert im IP Header nicht stimmt.
- Empf. Prio-0-Tel
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 empfangen wurden. VRRPv3-Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- Ges. Prio-0-Tel
  Zeigt an, wie viele VRRPv3-Pakete mit der Priorität 0 versendet wurden. Pakete mit der Priorität 0 werden versendet, wenn ein Master-Router heruntergefahren wird. Diese Pakete ermöglichen dann eine schnelle Übergabe an den entsprechenden Backup Router.
- Ungültiger Typ
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Wert im Feld "Typ" des IP-Headers ungültig ist.

- **Adresslistenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Adressliste nicht mit der lokal konfigurierten Liste übereinstimmt.
- **Telegrammlängenfehler**
  Zeigt an, wie viele fehlerhafte VRRPv3-Pakete empfangen wurden, deren Länge nicht korrekt ist.

Security

Übersicht

Die Seite zeigt die Sicherheitseinstellungen und die lokalen und externen Benutzerkonten an.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Dienste

Die Liste "Dienste" zeigt die Sicherheitseinstellungen an.

- **SSH-Server**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration".
  - Aktiviert: Verschlüsselter Zugriff auf das CLI
  - Deaktiviert: Kein verschlüsselter Zugriff auf das CLI
- **Webserver**
  Die Einstellung konfigurieren Sie unter "System > Konfiguration"
  - HTTP/HTTPS: Der Zugriff auf das WBM ist über HTTP und HTTPS möglich.
  - HTTPS: Der Zugriff auf das WBM ist nur noch über HTTPS möglich.
- **SNMP**
  Die Einstellung konfigurieren Sie unter "System > SNMP > Allgemein".
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich.
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit der SNMP Version 3 möglich.

- **Management ACL**
  Die Einstellung konfigurieren Sie unter "Security > Management ACL".
  - Deaktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist deaktiviert.
  - Aktiviert: Keine Zugriffsbeschränkung
    Die Zugriffskontrolle ist aktiviert, es sind jedoch keine Zugriffsregeln definiert.
  - Aktiviert: Nur eingeschränkter Zugriff
    Die Zugriffskontrolle ist aktiviert und es sind Zugriffsregeln definiert.
- **Login-Authentifizierung**
  Die Einstellung konfigurieren Sie unter "Security > AAA > Allgemein".
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
  - Lokal und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen.
    Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS mit Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen.
    Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.
- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie aktuell verwendet wird.

Lokale und externe Benutzerkonten

Lokale Benutzerkonten und Rollen konfigurieren Sie unter "Security > Benutzer".

Wenn Sie ein lokales Benutzerkonto anlegen, wird automatisch auch ein externes Benutzerkonto erzeugt. Bei lokalen Benutzerkonten handelt es sich um Benutzer mit jeweils einem Passwort zur Anmeldung auf dem Gerät.

In der Tabelle "Externe Benutzerkonten" wird ein Benutzer mit einer Rolle verknüpft, z. B. der Benutzer "Observer" mit der Rolle "user". In diesem Fall ist der Benutzer auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert, die zugehörige Gruppe jedoch unbekannt oder nicht vorhanden ist, prüft das Gerät, ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzerkonten" gibt. Wenn ein entsprechender Eintrag existiert, wird der Benutzer mit den Rechten der verknüpften Rolle angemeldet. Wenn die zugehörige Gruppe auf dem Gerät bekannt ist, werden beide Tabellen ausgewertet. Dem Benutzer wird die Rolle mit den größeren Rechten zugewiesen.

Hinweis

Die Tabelle "Externe Benutzerkonten" wird nur ausgewertet, wenn Sie im RADIUS Authorisierungsmodus "Herstellerspezifisch" eingestellt haben.

Über CLI können Sie auf die externen Benutzerkonten zugreifen.

Die Tabelle "Lokale Benutzerkonten" gliedert sich in folgende Spalten:

- **Benutzerkonto**
  Zeigt den Namen des lokalen Benutzers an.
- **Rolle**
  Zeigt die Rolle des Benutzers an. Weitere Informationen zu den Funktionsrechten der Rolle erhalten Sie unter "Information > Security > Rollen".

Unterstützte Funktionsrechte

Die Seite zeigt die Funktionsrechte an, die lokal auf dem Gerät verfügbar sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

- **Funktionsrecht**
  Zeigt die Nummer des Funktionsrechts an. Den Nummern sind unterschiedliche Rechte in Bezug auf die Geräteparameter zugeordnet.
- **Beschreibung**
  Zeigt die Beschreibung des Funktionsrechts an.

Gruppen

Diese Seite zeigt an, welche Gruppe mit welcher Rolle verknüpft ist. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an. Der Name entspricht der Gruppe auf dem RADIUS-Server.
- **Rolle**
  Zeigt die Rolle an. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät.
- **Beschreibung**
  Zeigt die Beschreibung für die Verknüpfung an.

Rollen

Die Seite zeigt die Rollen an, die lokal auf dem Gerät gültig sind.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Es ist von der Rolle des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Werte

Die Tabelle enthält folgende Spalten:

- **Rolle**
  Zeigt die Rolle an.
- **Funktionsrecht**
  Zeigt die Funktionsrechte der Rolle an:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen, aber nicht verändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.
  - 0
    Hierbei handelt es sich um eine Rolle, die das Gerät intern vergibt, wenn der Benutzer nicht authentifiziert werden konnte.
    Dem Benutzer wird der Zugriff auf das Gerät verweigert.
- **Beschreibung**
  Zeigt eine Beschreibung der Rolle an.

System-Funktionen projektieren

Konfiguration

Systemkonfiguration

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes.

Legen Sie fest, über welche Dienste auf das Gerät zugegriffen wird. Zu einigen Diensten gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind.

Beschreibung

Die Seite enthält die Konfigurationsübersicht über die Zugriffsmöglichkeiten des Gerätes

- **Telnet-Server**
  Aktivieren oder deaktivieren Sie den Dienst Telnet-Sever für den unverschlüsselten Zugriff auf das CLI.
- **SSH-Server**
  Aktivieren oder deaktivieren Sie den Dienst SSH-Server für den verschlüsselten Zugriff auf das CLI.
- **HTTP-Dienste**
  Legen Sie fest, wie auf das WBM zugegriffen wird:
  - Nur HTTPS
    Zugriff auf das WBM nur über HTTPS möglich.
  - HTTP/HTTPS
    Zugriff auf das WBM nur über HTTP und HTTPS möglich.
  - HTTP nach HTTPS umleiten
    Beim Zugriff über HTTP wird automatisch nach HTTPS umgeleitet.
- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client. Weitere Einstellungen konfigurieren Sie unter "System > SMTP-Client".
- **Syslog-Client**
  Aktivieren oder deaktivieren Sie den Syslog-Client. Weitere Einstellungen konfigurieren Sie unter "System > Syslog-Client".
- **DCP-Server**
  Legen Sie fest, ob auf das Gerät mit DCP (Discovery and Configuration Protocol) zugegriffen werden kann:
  - "-" (Deaktiviert)
    DCP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben
    Mit DCP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt
    Mit DCP können Geräteparameter zwar gelesen aber nicht verändert werden.

- **Zeiteinstellung**
  Wählen Sie die gewünschte Einstellung aus. Folgende Einstellungen sind möglich:
  - Manuell
    Die Systemzeit wird manuell eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > Manuelle Einstellung".
  - SIMATIC Time
    Die Systemzeit wird über einen SIMATIC Zeitgeber eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SIMATIC Time Client".
  - SNTP-Client
    Die Systemzeit wird über einen SNTP-Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SNTP-Client".
  - NTP-Client
    Die Systemzeit wird über einen NTP-Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > NTP-Client".
- **SNMP**
  Wählen Sie das gewünschte Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (SNMP deaktiviert)
    Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
  - SNMPv1/v2c/v3
    Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
  - SNMPv3
    Ein Zugriff auf die Geräteparameter ist nur mit SNMP Version 3 möglich. Weitere Einstellungen konfigurieren Sie unter " System > SNMP > Allgemein".
- **SNMPv1/v2 schreibgeschützt**
  Aktivieren oder deaktivieren Sie den schreibenden Zugriff auf SNMP-Variablen bei SNMPv1/v2c.
- **DHCP-Client**
  Aktivieren oder deaktivieren Sie den DHCP-Client. Weitere Einstellungen konfigurieren Sie unter "System > DHCP"

- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Versenden von SNMPv1-Traps (Alarmtelegramme). Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Traps".
- **Konfigurationsmodus**
  Wählen Sie die gewünschte Betriebsart. Folgende Betriebsarten sind möglich
  - Automatisches Speichern
    Automatischer Sicherungsbetrieb. Ca. 1 Minute nach der letzten Parameteränderung oder vor dem Neustart des Geräts wird die Konfiguration automatisch abgespeichert. Zusätzlich erscheint im Anzeigebereich folgende Meldung "Die Änderungen werden automatisch in x Sekunden gespeichert. Um die Änderungen sofort zu speichern, klicken Sie auf 'Schreiben der Startkonfiguration'."

    Hinweis

    Unterbrechung des Speichervorgangs

    Der Speichervorgang startet erst, nachdem der Timer in der Meldung abgelaufen ist. Die Dauer des Speichervorgangs ist vom Gerät abhängig.

    Während des Speichervorgangs wird die Meldung "Die Konfigurationsdaten werden gespeichert. Schalten Sie das Gerät nicht aus." angezeigt.

    - Schalten Sie das Gerät nicht sofort aus, nachdem der Timer abgelaufen ist.
  - Trial
    Trial-Modus. Im Trial-Modus werden Änderungen zwar übernommen aber nicht in der Konfigurationsdatei (Startup Configuration) gespeichert.
    Um Änderungen in der Konfigurationsdatei abzuspeichern, verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration". Zusätzlich wird im Anzeigebereich die Meldung "Der Konfigurationsmodus Trial ist aktiv - Klicken Sie auf die Schaltfläche "Schreiben der Startkonfiguration" um Ihre Einstellungen zu speichern" angezeigt, sobald es ungespeicherte Änderungen gibt. Diese Meldung ist auf jeder WBM-Seite sichtbar, bis die vorgenommenen Änderungen entweder gespeichert werden oder das Gerät neu gestartet wird.

Allgemein

Gerät

Diese Seite enthält die allgemeinen Geräteinformationen.

Beschreibung

Die Seite enthält folgende Felder:

- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemuhrzeit an. Die Systemuhrzeit wird entweder vom Anwender eingestellt oder per Uhrzeittelegramm synchronisiert: entweder SINEC H1 Uhrzeittelegramm, NTP oder SNTP. (Nur lesbar)
- **Systembetriebszeit** (nur Online verfügbar)
  Zeigt die Laufzeit des Geräts seit dem letzten Neustart an. (Nur lesbar)

- **Gerätetyp** (nur Online verfügbar)
  Zeigt die Typenbezeichnung des Geräts an. (Nur lesbar)
- **Systemname**
  Sie können den Namen des Gerätes eintragen. Der eingetragene Name wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich. Der Systemname wird auch in der CLI-Eingabeaufforderung (Prompt) angezeigt. In der CLI-Eingabeaufforderung ist die Anzahl der Zeichen begrenzt. Der Systemname wird nach 16 Zeichen abgeschnitten.
- **Kontaktperson**
  Sie können den Namen einer Kontaktperson eintragen, die für die Verwaltung des Gerätes zuständig ist. Es sind maximal 255 Zeichen möglich.
- **Gerätestandort**
  Sie können den Montageort des Gerätes eintragen. Der eingetragene Montageort wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich.

Hinweis

Erlaubte Zeichenn

Folgende darstellbare ASCII-Zeichen (0x20 bis 0x7e) in den Eingabfeldern Felder **"Systemname"**, **"Kontaktperson"** und **"Gerätestandort"** sind erlaubt:

- 0123456789
- A...Z a...z
- !"#$%&'()*+,-./:;<=>?@ [\]_{|}~^`

Koordinaten

Auf dieser Seite können Informationen über die geografischen Koordinaten (Breitengrad, Längengrad und die Höhe über dem Ellipsoid gemäß WGS84) eingetragen werden.

Ermittlung der Koordinaten

Nutzen Sie zur Ermittlung der geografischen Koordinaten des Gerätes entsprechendes Kartenmaterial.

Die geografischen Koordinaten können auch durch einen GPS-Empfänger ermittelt werden. Meist werden die geografischen Koordinaten von diesen Geräten direkt angezeigt und müssen nur noch in die Eingabefelder dieser Seite übertragen werden.

Beschreibung

Die Seite enthält folgende Eingabefelder mit einer maximalen Länge von 32 Zeichen:

- **Eingabefeld "Geographische Breite"**
  Geben Sie die nördliche oder die südliche Breite für den Standort des Gerätes ein.
  Z. B. +49° 1´ 31.67" bedeutet, dass sich das Gerät auf 49 Grad, 1 Bogenminute und 31.67 Bogensekunden nördlicher Breite befindet.
  Die südliche Breite wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben N (nördliche Breite) oder S (südliche Breite) an die Zahlenangabe anhängen (49° 1´ 31.67" N).
- **Eingabefeld "Geographische Länge"**
  Geben Sie die östliche oder die westliche Länge für den Standort des Gerätes ein.
  Z. B. +8° 20´ 58.73" bedeutet, dass sich das Gerät auf 8 Grad, 20 Bogenminuten und 58.73 Bogensekunden östliche Länge befindet.
  Die westliche Länge wird mit einem führenden Minuszeichen dargestellt.
  Sie können auch die Buchstaben O bzw. E (östliche Länge) oder W (westliche Länge) an die Zahlenangabe anhängen (8° 20´ 58.73" E).
- **Eingabefeld "Geographische Höhe"**
  Geben Sie die geografische Höhe über oder unter normal Null (Meereshöhe) in Metern ein.
  Z.B. 158 m bedeutet, dass sich das Gerät in einer Höhe von 158 m über normal Null befindet.
  Höhenangaben unterhalb von normal Null (z. B. am toten Meer) werden mit einem führenden Minuszeichen dargestellt.

Neustart

Auf dieser Seite finden Sie eine Schaltfläche zum Neustart des Gerätes sowie verschiedene Möglichkeiten, die Voreinstellungen des Gerätes zurückzusetzen.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Hinweis

Beachten Sie folgende Punkte beim Neustart eines Gerätes:

- Sie können einen Neustart des Gerätes nur mit Administrator-Rechten durchführen.
- Der Neustart eines Gerätes sollte nur durch die Schaltflächen dieser Seite oder durch die entsprechenden CLI-Befehle und nicht durch Aus- und Einschalten der Spannungsversorgung am Gerät erfolgen.
- Vorgenommene Änderungen werden erst nach dem Anklicken der Schaltfläche "Einstellungen übernehmen" auf der jeweiligen Seite im Gerät wirksam. Wenn sich das Gerät im "Trial-Modus" befindet, müssen Konfigurationsänderungen vor einem Neustart manuell abgespeichert werden. Im Modus "Automatisch Speichern" werden die letzten Änderungen automatisch vor einem Neustart gespeichert.

Beschreibung

Für den Neustart gibt es folgende Möglichkeiten:

- **Neustart**
  Klicken Sie auf diese Schaltfläche, um das System neu zu starten. Sie müssen den Neustart in einer Dialogbox bestätigen. Bei einem Neustart wird das Gerät neu initialisiert, die interne Firmware wird neu geladen und das Gerät führt einen Selbsttest durch. Die gelernten Einträge in der Adresstabelle werden gelöscht. Sie können das Browser-Fenster geöffnet lassen, während das Gerät neu startet. Sie müssen sich wieder neu anmelden.
- **Auf gespeicherte Einstellungen zurücksetzen und Neustart**
  Klicken Sie diese Schaltfläche, um die werkseitigen Konfigurationseinstellungen mit Ausnahme der folgenden Parameter zurückzusetzen und einen Neustart auszuführen:
  - IP-Adressen
  - Subnetz-Maske
  - IP-Adresse des Standard-Gateways
  - DHCP Client ID
  - DHCP
  - Systemname
  - System-Aufstellungsort
  - System-Ansprechpartner
  - Betriebsart des Geräts
- **Auf Werkseinstellungen zurücksetzen und Neustart**
  Klicken Sie auf diese Schaltfläche, um die werkseitigen Konfigurationseinstellungen wiederherzustellen. Es werden auch die geschützten Voreinstellungen zurückgesetzt. Es wird ein automatischer Neustart ausgeführt.

Hinweis

Durch das Zurücksetzen auf die werkseitigen Konfigurationseinstellungen verliert das Gerät seine projektierte IP-Adresse, siehe Kapitel "Voraussetzungen für den Betrieb".

- M87x, M81x: Das Gerät ist wieder über die werkseitig eingestellte IP-Adresse 192.168.1.1 zu erreichen.
- M826: Dem Gerät muss eine IP-Adresse zugeordnet werden. Das Gerät ist danach nur über das Primary Setup Tool oder über DHCP ansprechbar.

Bei entsprechendem Anschluss kann ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

Laden & Speichern

Hochladen und Speichern über TFTP

Laden und Speichern von Daten über einen TFTP-Server

Auf der Seite können Sie den TFTP-Server und die Dateinamen konfigurieren. Weiter können Sie Gerätedaten in einer externen Datei auf einem TFTP-Server speichern bzw. solche Daten aus einer externen Datei vom TFTP-Server in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von einem TFTP-Server laden.

Firmware

Die Firmware ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Hinweis

Inkompatibilität zu Firmware-Vorgängerversionen ohne/mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über die WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Modus Trial/Automatisches Speichern

Im Modus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Modus "Trial" werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration" auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden.

Austausch von Konfigurationsdaten mit STEP7 Basic/Professional über eine Datei

Über die beiden Dateitypen "RunningSINEMAConfig" und "SINEMAConfig" können Sie Konfigurationsdaten zwischen einem Gerät (WBM) und STEP7 Basic/Professional über eine Datei austauschen.

Voraussetzungen:

- Gleiche Artikelnummer
- Gleiche Firmware-Version
- Passwort
  Das Passwort vergeben Sie im WBM unter "System > Laden&Speichern > Passwörter".

Die Dateitypen können wie folgt genutzt werden:

- Zur Offline-Diagnose
  Sie können die fehlerhafte Konfiguration eines Geräts als "RunningSINEMAConfig" über das WBM speichern und in STEP7 Basic/Professional importieren. Die Diagnose in STEP7 Basic/Professional erfolgt, ohne dass eine Verbindung zu einem realen Gerät besteht. Eine korrigierte Konfiguration kann exportiert und als "SINEMAConfig" wieder über das WBM geladen werden.
- Zur Konfiguration
  Sie können ein Gerät in STEP7 Basic/Professional konfigurieren, ohne dass eine Verbindung zu einem realen Gerät besteht. Die Konfiguration kann exportiert und als "SINEMAConfig" über das WBM in das reale Gerät geladen werden.

Beschreibung

- **Adresse des TFTP-Servers**
  Tragen Sie die IP-Adresse oder den FQDN des TFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des TFTP-Servers**
  Tragen Sie hier den Port des TFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 69 entsprechend Ihren spezifischen Anforderungen ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis

  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktion** (nur Online verfügbar)
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp, z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - Datei speichern
    Mit dieser Auswahl speichern Sie eine Datei auf den TFTP-Server.
  - Datei hochladen
    Mit dieser Auswahl laden Sie eine Datei vom TFTP-Server, z. B. Zertifikate, die für den Aufbau einer gesicherten Verbindung notwendig sind.

Hochladen und Speichern über SFTP

Laden und Speichern von Daten über einen SFTP-Server

SFTP (SSH File Transfer Protocol) überträgt die Dateien verschlüsselt. Auf dieser Seite konfigurieren Sie die Zugangsdaten für den SFTP-Server.

Weiter können Sie Gerätedaten in einer externen Datei auf Ihrem Client-PC speichern bzw. solche Daten aus einer externen Datei vom PC in die Geräte laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Admin-PC laden.

Zudem lassen sich auf dieser Seite die Zertifikate laden, die für den Aufbau einer gesicherten VPN-Verbindung notwendig sind.

Firmware

Die Firmware ist signiert und verschlüsselt. Es ist sichergestellt, dass nur von Siemens erstellte Firmware in das Gerät geladen werden kann.

Konfigurationsdateien

Hinweis

Konfigurationsdateien und Modus Trial/Automatisches Speichern

Im Modus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden.
Im Modus "Trial" werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Startkonfiguration" auf der WBM-Seite "System > Konfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

CLI-Skriptdatei

Sie können bestehende CLI-Konfigurationen herunterladen (RunningCLI) und eigene CLI-Skripte hochladen (Script).

Hinweis

Das herunterladbare CLI-Skript ist nicht dafür vorgesehen, dass Sie es unverändert wieder hochladen.

CLI-Befehle zum Speichern und Laden von Dateien können nicht über die CLI-Skriptdatei (Script) ausgeführt werden

Beschreibung

Die Seite enthält folgende Felder:

- **Adresse des SFTP-Servers**
  Geben Sie die IP-Adresse oder den FQDN des SFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des SFTP-Servers**
  Geben Sie den Port des SFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 22 entsprechend Ihren spezifischen Anforderungen ändern.
- **SFTP Benutzer**
  Geben Sie den Benutzer für den Zugriff auf den SFTP-Server ein. Vorausgesetzt, auf dem SFTP-Server ist ein Benutzer mit den entsprechenden Rechten angelegt.
- **SFTP Passwort**
  Geben Sie das Passwort für den Benutzer ein
- **SFTP Passwort bestätigen**
  Bestätigen Sie das Passwort.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**
  Zeigt den Dateityp an.
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.

- **Dateiname**
  Für jeden Dateityp ist hier ein Dateiname vorgegeben.

  Hinweis
  Änderung des Dateinamens

  Sie können den in dieser Spalte vorgegebenen Dateinamen ändern. Nach dem Laden auf das Gerät kann der geänderte Dateiname auch mit dem Command Line Interface genutzt werden.

- **Aktionen**
  Wählen Sie die gewünschte Aktion aus. Die Auswahl ist abhängig vom gewählten Dateityp. z. B. können Sie die Log-Datei nur speichern.
  Folgende Aktionen sind möglich:
  - **Datei speichern**
    Mit dieser Auswahl speichern Sie eine Datei auf dem SFTP-Server.
  - **Datei hochladen**
    Mit dieser Auswahl laden Sie eine Datei vom SFTP-Server, z. B. Zertifikate, die für den Aufbau einer gesicherten Verbindung notwendig sind.

Passwörter

Passwort für Dateien

Es gibt Dateien, deren Zugriff passwortgeschützt ist. Um die Datei erfolgreich ins Gerät zuladen, geben Sie auf der Seite das für die Datei festgelegte Passwort ein.

Beschreibung

- **Typ**
  Zeigt den Dateityp an
- **Beschreibung**
  Zeigt die Kurzbeschreibung des Dateityps an.
- **Aktiviert**
  Wenn aktiviert, wird die Datei verwendet. Nur aktivierbar, wenn das Passwort konfiguriert ist
- **Passwort**
  Geben Sie das Passwort für die Datei ein.

- **Passwort bestätigen**
  Bestätigen Sie das Passwort.
- **Status** (nur Online verfügbar)
  Zeigt an, ob das Passwort zur der Datei auf dem Gerät passen.
  - gültig
    Das Optionskästchen "Aktiviert" ist aktiviert und das Passwort passt zu der Datei.
  - ungültig
    Das Optionskästchen "Aktiviert" ist aktiviert, aber das Passwort passt nicht zu der Datei oder es ist noch keine Datei geladen.
  - '-'
    Das Passwort kann nicht ausgewertet werden oder wird noch nicht verwendet. Das Optionskästchen "Aktiviert" ist nicht aktiviert.

Ereignisse

Konfiguration

Systemereignisse auswählen

Auf der Seite legen Sie fest, welche Systemereignisse wie protokolliert werden.

Folgende Meldungen werden immer in die Ereignisprotokoll-Tabelle eingetragen und sind nicht abwählbar:

- Ändern des Admin-Kennworts
- Starten des Geräts
- Betriebstatus des Geräts, z. B. ob ein PLUG vorhanden ist oder nicht
- Status unerledigter Fehler

Um diese Meldungen zusätzlich an einen Syslog-Server zu senden, aktivieren Sie beim Ereignis "System Allgemein Logs" das Optionskästchen "Syslog".

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Alle Ereignisse**
  Zeigt an, dass die Einstellungen für alle Ereignisse der Tabelle 2 gültig sind.
- **E-Mail / Trap / Log-Tabelle / Syslog / Fehler / SMS / Digitaler Ausgang / VPN-Tunnel**
  Aktivieren oder deaktivieren Sie die gewünschte Art der Benachrichtigung für alle Ereignisse. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ereignisse der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Ereignis**
  Die Spalte "Ereignis" enthält Folgendes:
  - Kalt-/Warmstart
    Das Gerät wurde eingeschaltet oder vom Anwender neu gestartet. Im Fehlerspeicher des Geräts wird ein neuer Eintrag mit der Art des durchgeführten Neustarts erzeugt.
  - Link Change
    Dieses Ereignis tritt nur auf, wenn der Port-Status überwacht wird und sich entsprechend geändert hat, siehe "System > Fehlerkontrolle > Link Change".
  - Authentifizierungsfehler
    Dieses Ereignis tritt beim Versuch eines Zugriffs mit fehlerhaftem Passwort auf.
  - Änderung des Fehlerstatus
    Der Fehlerstatus hat sich geändert. Der Fehlerstatus kann sich auf die aktivierte Portüberwachung, auf das Ansprechen des Meldekontakts oder die Spannungsüberwachung beziehen.
  - Security-Logs
    Im Sicherheitslogbuch wird eingetragen, wenn das IPsec-Verfahren für VPN angewendet
  - Firewall-Logs
    Im Firewall-Logbuch wird eingetragen, wann einzelne Firewall-Regeln angewendet wurden. Dazu muss zu den verschiedenen Firewall-Funktionen die LOG-Funktion aktiviert werden.
  - DDNS Client-Logs
    Das Ereignis tritt auf, wenn der DDNS-Client die zugewiesene IP-Adresse mit dem im DDNS-Provider registrierten Hostnamen synchronisiert.
  - System-Verbindungsstatus
    Der Verbindungsstatus hat sich geändert.
  - System Allgemein Logs
    Verbindungsaufbau, Änderung der Konfiguration.
  - Digitaler Eingang
    Das Ereignis tritt auf, wenn sich der Zustand des digitalen Eingangs geändert hat.
  - VPN-Tunnel
    Das Ereignis tritt auf, wenn sich der Zustand von VPN (IPsec, OpenVPN, SINEMA RC) geändert hat.
  - Secure NTP
    Das Ereignis tritt auf, wenn sich der Zustand vom Secure NTP-Server geändert hat
  - Änderung der Konfiguration
    Dieses Ereignis tritt ein, wenn sich die Konfiguration des Geräts geändert hat.
- **E-Mail**
  Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- **Trap**
  Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.

- Log-Tabelle
  Das Gerät schreibt einen Eintrag in die Ereignisprotokoll-Tabelle, siehe "Information > Log-Tabelle".
- Syslog
  Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- Fehler
  Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf
- SMS (nur bei M87x)
  Das Gerät sendet eine SMS. Voraussetzung ist, dass "System > SMS > Ereignis SMS" aktiviert ist und die Telefonnummer des Empfängers konfiguriert ist.
- Digitaler Ausgang
  Steuert den digitalen Ausgang an oder signalisiert die Zustandsänderung mit der LED "DO".
- VPN-Tunnel
  Steuert die Weitergabe eines Ereignisses an eine VPN-Verbindung (IPsec, OpenVPN, SINEMA RC). Solange das Ereignis anliegt, wird die VPN-Verbindung aktiv geschaltet.

Severity-Filter

Auf dieser Seite konfigurieren Sie die Fehlerschwere für das Versenden von Systemereignisbenachrichtigungen.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- Client-Typ
  Wählen Sie den Client-Typ, für den Sie die Einstellungen vornehmen:
  - E-Mail
    Versand von Systemereignismeldungen per E-Mail.
  - Log-Tabelle
    Eintragen von Systemereignissen in die Log-Tabelle.
  - Syslog
    Eintragen von Systemereignissen in die Syslog-Datei.
  - SMS (nur bei M87x)
    Versand von Systemereignismeldungen per SMS.
- Severity
  Wählen Sie die gewünschte Stufe aus. Folgende Einstellungen sind möglich:
  - Info
    Die Meldungen aller Stufen werden versendet bzw. protokolliert.
  - Warning
    Die Meldung dieser Stufe und der Stufe "critical" werden versendet bzw. protokolliert.
  - Critical
    Nur die Meldungen dieser Stufe werden versendet bzw. protokolliert.

SMTP Client

Das Gerät bietet die Möglichkeit, beim Auftreten eines Alarmereignisses automatisch eine E-Mail (z.B. an den Netzwerkadministrator) zu senden. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Damit kann für Netze mit wenigen Teilnehmern eine einfache zentrale Netzüberwachung auf Basis eines E-Mail-Systems aufgebaut werden. Bei eintreffenden E-Mail-Störmeldungen kann über die Identifikation des Absenders per Browser das WBM gestartet werden, um weitere Diagnoseinformationen auszulesen.

Auf dieser Seite können Sie bis zu drei SMTP-Server und die dazugehörigen E-Mail-Adressen konfigurieren.

Beschreibung

Die Seite enthält folgende Felder:

- **SMTP-Client**
  Aktivieren oder deaktivieren Sie den SMTP-Client.
- **E-Mail-Adresse des Absenders**
  Geben Sie den Absendernamen ein, der in der E-Mail angegeben werden soll, z. B. den Gerätenamen.
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **Test-E-Mail senden**
  Verschicken Sie eine Test-E-Mail, um Ihre Konfiguration zu prüfen.
- **SMTP-Port**
  Geben Sie den Port ein, über den Ihr SMTP-Server erreichbar ist.
  Werkseinstellung: 25
  Diese Einstellung gilt für alle konfigurierten SMTP-Server.
- **SMTP-Server-Adresse**
  Geben Sie die IP-Adresse, den FQDN (Fully Qualified Domain Name) oder den Hostnamen des SMTP-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in einer zu löschenden Zeile das Optionskästchen.
- **SMTP-Server-Adresse**
  Zeigt die IP-Adresse, den FQDN (Fully Qualified Domain Name) oder den Hostnamen des SMTP-Servers.
- **E-Mail-Adresse des Empfängers**
  Geben Sie die E-Mail-Adresse ein, an die das Gerät im Fehlerfall eine E-Mail sendet.

SNMP

Allgemein

Auf dieser Seite treffen Sie grundlegende Einstellungen für SNMP. Aktivieren Sie die Optionen abhängig von der Funktion, die Sie nutzen wollen.

Beschreibung

- **SNMP**
  Wählen Sie das SNMP-Protokoll aus. Folgende Einstellungen sind möglich:
  - "-" (Deaktiviert)
    SNMP deaktiviert.
  - SNMPv1/v2c/v3
    SNMPv1/v2c/v3 wird unterstützt.

    Hinweis

    Beachten Sie, dass SNMP in den Versionen 1 und 2c über keine Sicherheitsmechanismen verfügt.
  - SNMPv3
    Nur SNMPv3 wird unterstützt.
- **SNMPv1/v2 schreibgeschützt**
  Wenn Sie diese Option aktivieren, kann SNMPv1/v2c nur lesend auf die SNMP-Variablen zugreifen.

  Hinweis

  Community String

  Verwenden Sie aus Sicherheitsgründen nicht die Standardwerte "public" oder "private". Ändern Sie die Community Strings nach der Erst-Installation.

  Die empfohlene Mindestlänge für Community Strings sind 6 Zeichen.
- **SNMPv1/v2c Read Community String - Lesen**
  Tragen Sie den Community String für den Zugriff des SNMP-Protokolls ein.
- **SNMPv1/v2c Read/Write Community String - Lesen/Schreiben**
  Tragen Sie den Community String für den lesenden und schreibenden Zugriff des SNMP-Protokolls ein.
- **SNMPv1-Traps**
  Aktivieren oder deaktivieren Sie das Senden von SNMPv1-Traps (Alarmtelegramme). Im Eintrag "Trap" legen Sie die IP-Adressen der Geräte fest, an die SNMPv1-Traps gesendet werden.
- **SNMPv1/v2c Trap Community String - Trap**
  Tragen Sie den Community String für das Senden von SNMPv1/v2c-Meldungen ein.

- **SNMPv3 Benutzermigration**
  - **Aktiviert**
    Wenn die Funktion aktiviert ist, wird eine SNMP-Engine-ID generiert, die migriert werden kann. Sie können konfigurierte SNMPv3-Benutzer auf ein anderes Gerät übertragen. Wenn Sie diese Funktion aktivieren und die Konfiguration des Geräts auf ein anderes Gerät laden, bleiben konfigurierte SNMPv3-Benutzer erhalten.
  - **Deaktiviert**
    Wenn die Funktion deaktiviert ist, wird eine gerätespezifische SNMP-Engine-ID generiert. Um die ID zu generieren, wird die Agent-MAC-Adresse des Geräts verwendet. Sie können diese SNMP-Benutzerkonfiguration nicht auf andere Geräte übertragen. Wenn Sie die Konfiguration des Geräts auf ein anderes Gerät laden, werden alle konfigurierten SNMPv3-Benutzer gelöscht.
- **SNMP-Engine-ID** (nur Online verfügbar)
  Zeigt die SNMP-Engine-ID an.

Traps

Beim Eintreten eines Alarmereignisses kann ein Gerät SNMP-Traps (Alarmtelegramme) an bis zu zehn verschiedene Management-Stationen gleichzeitig senden. Es werden nur bei solchen Ereignissen Traps gesendet, die unter "Ereignisse" festgelegt sind.

Hinweis

SNMP-Traps werden nur dann versendet, wenn Sie bei "SNMP >Allgemein" oder unter "System > Konfiguration" die Einstellung "SNMPv1 Traps" aktiviert haben.

Beschreibung

- **Trap-Empfängeradresse**
  Tragen Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) der Station ein, an die das Gerät SNMP-Traps sendet. Sie können bis zu zehn verschiedene Empfänger angeben.

Die Tabelle gliedert sich in folgende Spalten:

- **Trap-Empfängeradresse**
  Ändern Sie bei Bedarf die IP-Adressen oder den FQDN (Fully Qualified Domain Name) der Stationen.
- **Trap**
  Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps. Stationen, die eingetragen, aber nicht selektiert sind, erhalten keine SNMP-Traps.

v3-Gruppen

Sicherheitseinstellungen und Rechtevergabe

SNMP Version 3 bietet eine Rechtevergabe, Authentifizierung und Verschlüsselung auf Protokollebene. Das Security-Level und die Lese-/Schreibrechte werden gruppenspezifisch definiert. Für jedes Mitglied einer Gruppe gelten automatisch die entsprechenden Einstellungen.

Beschreibung

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Die maximale Länge beträgt 32 Zeichen.
- **Security-Level**
  Wählen Sie die Sicherheitsstufe (Authentifizierung, Verschlüsselung) aus, die für die gewählte Gruppe gültig ist. Es gibt folgende Möglichkeiten:
  - Keine Auth./keine Priv.
    Keine Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./keine Priv.
    Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./Priv.
    Authentifizierung aktiviert / Verschlüsselung aktiviert.

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**
  Zeigt die definierten Gruppennamen an.

  Hinweis

  Der einmal vergebene Gruppenname und die Sicherheitsstufe können nach dem Anlegen nicht mehr geändert werden. Wenn Sie den Gruppennamen oder die Sicherheitsstufe ändern wollen, müssen Sie die Gruppe löschen und mit dem neuen Namen neu anlegen und neu konfigurieren.

- **Security-Level**
  Zeigt die konfigurierte Sicherheitsstufe an.
- **Lesen**
  Aktivieren oder deaktivieren Sie den Lesezugriff für die gewünschte Gruppe.

- **Schreiben**
  Aktivieren oder deaktivieren Sie den Schreibzugriff für die gewünschte Gruppe.

  Hinweis

  Damit der Schreibzugriff funktioniert, müssen Sie ebenfalls den Lesezugriff aktivieren.

- **Persistenz**
  Zeigt an, ob die Gruppe einem SNMPv3-Benutzer zugeordnet ist. Wenn die Gruppe keinem SNMPv3-Benutzer zugeordnet ist, wird kein automatisches Speichern ausgelöst und die konfigurierte Gruppe ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Die Gruppe ist einem SNMPv3-Benutzer zugeordnet.
  - Nein
    Die Gruppe ist keinem SNMPv3-Benutzer zugeordnet.

v3-Benutzer

Benutzerspezifische Sicherheitseinstellungen

Auf der Seite können Sie SNMPv3-Benutzer neu anlegen, ändern oder löschen. Das benutzerbasierte Sicherheitsmodell arbeitet mit dem Konzept des Benutzernamens, d. h. jedes Telegramm wird mit einer Benutzerkennung versehen. Diesen Benutzernamen und die betreffenden Sicherheitseinstellungen überprüfen sowohl der Absender wie auch der Empfänger.

Beschreibung

- **Benutzername**
  Geben Sie einen frei wählbaren Benutzernamen ein. Nach der Datenübernahme können Sie den Namen nicht mehr ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Benutzername**
  Zeigt die angelegten Benutzer an.
- **Gruppenname**
  Wählen Sie die Gruppe aus, die dem Benutzer zugeordnet wird.
- **Authentifizierungsprotokoll**
  Legen Sie das Authentifizierungsprotokoll fest, für das ein Passwort hinterlegt werden soll. Folgende Einstellungen gibt es:
  - Keine
  - MD5
  - SHA
- **Verschlüsselungsprotokoll**
  Legen Sie fest, ob ein Passwort zur Verschlüsselung mit dem DES-Algorithmus hinterlegt werden soll. Nur aktivierbar, wenn auch ein Authentifizierungsprotokoll ausgewählt wurde.

- **Authentifizierungspasswort**
  Geben Sie das Authentifizierungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Authentifizierungspasswort bestätigen**
  Bestätigen Sie das Passwort durch die Wiederholung der Eingabe.
- **Verschlüsselungspasswort**
  Geben Sie Ihr Verschlüsselungspasswort ein. Das Passwort muss mindestens 1 Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.

  Hinweis

  Länge des Passworts

  Als wichtige Maßnahme zur Erhöhung der Sicherheit empfehlen wir, dass das Passwort mindestens 6 Zeichen lang ist und Sonderzeichen, Groß-/Kleinschreibung sowie Zahlen enthält.

- **Verschlüsselungspasswort bestätigen**
  Bestätigen Sie das Verschlüsselungspasswort durch die Wiederholung der Eingabe.
- **Persistenz**
  Zeigt an, ob der Benutzer einer SNMPv3-Gruppe zugeordnet ist. Wenn der Benutzer keiner SNMPv3-Gruppe zugeordnet ist, wird kein automatisches Speichern ausgelöst und der konfigurierte Benutzer ist nach einem Neustart des Geräts gelöscht.
  - Ja
    Der Benutzer ist einer SNMPv3-Gruppe zugeordnet.
  - Nein
    Der Benutzer ist keiner SNMPv3-Gruppe zugeordnet.

Systemzeit

Manuelle Zeiteinstellung

Manuelle Einstellung der Systemzeit

Auf dieser Seite stellen Sie das Datum und die Uhrzeit des Systems ein. Damit diese Einstellung verwendet wird, müssen Sie "Manuelle Zeiteinstellung" aktivieren.

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Manuelle Zeiteinstellung**
  Aktivieren oder deaktivieren Sie die manuelle Zeiteinstellung. Wenn Sie die Option aktivieren, wird das Eingabefeld "Systemzeit" editierbar.
- **Systemzeit**
  Geben Sie Datum und Uhrzeit im Format "MM/DD/YYYY HH:MM:SS" ein.
- **PC-Zeit verwenden**
  Klicken Sie auf die Schaltfläche, um die Zeiteinstellung des PCs zu übernehmen.
- **Letzter Synchronisationszeitpunkt**
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
  Wenn keine Uhrzeitsynchronisation möglich war, enthält das Feld die Angabe "Datum/Zeit nicht eingestellt".
- **Letzter Synchronisationsmechanismus**
  Dieses Feld zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde.
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm.
  - PTP
    Automatische Zeitsynchronisation über PTP
- **Sommerzeit (DST)**
  Zeigt an, ob die Umstellung der Sommerzeit aktiv ist.
  - active (offset +1 h)
    Die Systemzeit wurde auf Sommerzeit umgestellt, d. h. es wird eine Stunde hinzugezählt. Die aktuelle Systemzeit sehen Sie oben rechts im Auswahlbereich des WBM. In dem Feld "Systemzeit" wird weiterhin die eingestellte Zeit angezeigt.
  - inactive (offset +0 h)
    Die aktuelle Systemzeit wird nicht verändert.

SNTP-Client

Um die Systemzeit des Geräts einzustellen, gibt es unterschiedliche Methoden. Es kann immer nur eine Methode aktiv sein. Wenn eine Methode aktiviert wird, dann wird automatisch die bisher aktivierte Methode deaktiviert.

Das SNTP (Simple Network Time Protocol) dient zur Zeitsynchronisation im Netzwerk. Die entsprechenden Telegramme werden von einem SNTP-Server im Netz versendet.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Voraussetzung

Um die SNTP-Telegramme zu empfangen, aktivieren Sie unter "Security > Firewall > Vordefinierte IPv4-Regeln" den Eintrag "Systemzeit".

Beschreibung

- **SNTP-Client**
  Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über SNTP.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom Gerät empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
- **Zeitzone**
  Geben Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.

- **SNTP-Modus**
  Wählen Sie die Synchronisationsart aus. Folgende Synchronisierungsarten sind möglich:
  - Poll
    Wenn Sie diese Protokollart wählen, werden die Eingabefelder "SNTP-Server-Adresse", "Port des SNTP Servers" und "Poll-Intervall[s]" zur weiteren Konfiguration eingeblendet. Bei dieser Synchronisationsart ist das Gerät aktiv und sendet eine Zeitabfrage an den SNTP-Server.
  - Listen
    Bei dieser Synchronisationsart ist das Gerät passiv und empfängt SNTP-Telegramme, die die Uhrzeit liefern.
- **Poll-Intervall[s]**
  Geben Sie den Zeitabstand zwischen zwei Zeitanfragen ein. In diesem Feld geben Sie das Abfrageintervall in Sekunden an. Mögliche Werte sind 16 bis 16284 Sekunden.
- **SNTP-Server-Adresse**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des SNTP-Servers an.
- **Port des SNTP-Servers**
  Geben Sie den Port des SNTP-Servers ein.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564
- **Primär**
  Bei dem SNTP-Server, den SIe zuerst anlegen, wird das Häckchen gesetzt. Wenn mehrere SNTP-Server angelegt sind, wird der primäre Server zuerst angefragt.

NTP-Client

Wenn die Uhrzeitsynchronisation über NTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Voraussetzung

Um die NTP-Telegramme zu empfangen, aktivieren Sie unter "Security > Firewall > Vordefinierte IPv4-Regeln" den Eintrag "Systemzeit".

Beschreibung

- **NTP-Client**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem NTP-Server.
- **Nur NTP (secure)-Client**
  Wenn aktiviert, erhält das Gerät die Systemzeit von einem gesicherten NTP-Server. Die Einstellung gilt für alle Severeinträge.
  Um den gesicherten NTP-Client zu aktivieren, konfigurieren Sie die Parameter für die Authentifizierung (Schlüssel-ID, Hash-Algorithmus, Schlüssel). Auf dem Secure NTP-Server müssen diese Einträge vorhanden sein.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt das aktuelle Datum und die aktuelle Normalzeit an, die vom IE-Switch empfangen wurden. Wenn Sie eine Zeitzone angeben, wird die Zeitangabe entsprechend angepasst.
- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm
- **Zeitzone**
  Geben Sie die verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit.
  Die Zeitangabe im Feld "Aktuelle Systemzeit" wird entsprechend angepasst.
- **NTP-Serverindex**
  Wählen Sie den Index des NTP-Servers aus. Der Server mit dem kleinsten Index wird zuerst angefragt.
- **NTP-Server-Adresse**
  Geben Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des NTP-Servers an.
- **Port des NTP-Servers**
  Geben Sie den Port des NTP-Servers an.
  Folgende Ports sind möglich:
  - 123 (Standard-Port)
  - 1025 bis 36564

- **Poll-Intervall**
  Legen Sie den Zeitabstand zwischen zwei Uhrzeitanfragen fest. Je größer der Zeitabstand, desto ungenauer ist die Uhrzeit des Geräts.
  Mögliche Werte sind 64 bis 2592000 Sekunden (30 Tage).
- **Schlüssel-ID**
  Geben Sie die ID des Authentifizierungsschlüssels ein.
- **Hash-Algorithmus**
  Legen Sie das Format für den Authentifizierungsschlüssel fest.
- **Schlüssel**
  Geben Sie den Authentifizierungsschlüssel ein. Die Länge ist abhängig vom Hash-Algorithmus.
  - DES: ASCII 8 Zeichen
  - MD5: ASCII 16 – 128 Zeichen
  - SHA1: ASCII 20 – 128 Zeichen
- **Schlüssel bestätigen**
  Geben Sie nochmals den Authentifizierungsschlüssel ein, um ihn zu bestätigen.

SIMATIC Time Client

Zeiteinstellung über SIMATIC Time Client

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den SIMATIC Time Client.

Hinweis

Um Zeitsprünge zu vermeiden, stellen Sie sicher, dass sich nur ein Zeitserver im Netz befindet.

Beschreibung

- **SIMATIC Time Client**
  Markieren Sie dieses Optionskästchen, um das Gerät als SIMATIC Time Client zu aktivieren.
- **Aktuelle Systemzeit** (nur Online verfügbar)
  Zeigt die aktuelle Systemzeit an.

- **Letzter Synchronisationszeitpunkt** (nur Online verfügbar)
  Zeigt an, wann die letzte Uhrzeitsynchronisation stattgefunden hat.
- **Letzter Synchronisationsmechanismus** (nur Online verfügbar)
  Zeigt an, wie die letzte Zeitsynchronisation durchgeführt wurde. Folgende Arten gibt es:
  - Nicht eingestellt
    Die Zeit wurde nicht eingestellt.
  - Manuell
    Manuelle Zeiteinstellung
  - SNTP
    Automatische Zeitsynchronisation über SNTP
  - NTP
    Automatische Zeitsynchronisation über NTP
  - SIMATIC
    Automatische Zeitsynchronisation über SIMATIC-Uhrzeittelegramm

NTP-Server

Auf dieser Seite konfigurieren Sie das Gerät als NTP-Server. Die anderen Geräte können über diesen NTP-Server die vom Gerät bereitgestellteZeit abrufen. Damit sind die versorgten Geräte nicht auf eine Verbindung zu einem externen Zeitserver angewiesen.

Hinweis

Zeitsynchronisation

Damit das Gerät die angeschlossenen Geräte auf einen korrekten Zeitstand synchronisiert, konfigurieren Sie das Gerät auch als NTP-Client. Als NTP-Client bezieht das Gerät die präzise Zeit von einem externen Zeitserver und verteilt diese als NTP-Server an seine NTP-Clients.

Voraussetzung

Um die NTP-Telegramme zu empfangen, aktivieren Sie unter "Security > Firewall > Vordefinierte IPv4-Regeln" den Eintrag "Systemzeit".

Beschreibung

Die Seite enthält folgende Felder:

- **NTP-Server**
  Aktivieren oder deaktivieren Sie den Dienst NTP-Sever

Die Tabelle 1 gliedert sich in folgende Spalten

- **Alle Schnittstellen**
  Zeigt an, dass die Einstellungen für alle Schnittstellen der Tabelle 2 gültig sind.
- **Mithören**
  Wählen Sie in der Klappliste die Einstellung für alle Schnittstellen aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Schnittstellen der Tabelle 2 übernommen

Die Tabelle 2 gliedert sich in folgende Spalten

- **Schnittstelle**
  Über diese Schnittstelle wird die Zeit mittels NTP übermittelt.
- **Mithören**
  Wenn aktiviert, können die anderen Geräte über diese Schnittstelle die Uhrzeit abrufen.

Auto-Logout

Stellen Sie in dieser Seite die Zeiten ein, nach denen bei Inaktivität des Benutzers automatisch eine Abmeldung vom WBM oder dem CLI erfolgt.

Wenn Sie automatisch abgemeldet wurden, dann müssen Sie sich wieder neu anmelden.

Hinweis

Keine automatische Abmeldung vom CLI

Wenn die Verbindung nach der eingestellten Zeit nicht beendet wird, prüfen Sie am Telnet Client die Einstellung des "Keep alive"- Mechanismus.

Ist die Intervallzeit kleiner als die projektierte Zeit, wird die die Verbindung aufrecht erhalten, obwohl keine Nutzdaten übertragen werden. Z. B. Sie haben bei der automatischen Abmeldung 300 Sekunden eingestellt und bei der "Keep alive"- Funktion steht 120 Sekunden. In diesem Fall wird alle 120 Sekunden ein Paket gesendet, dass die Verbindung aufrecht erhält.

- Schalten Sie den Kepp-Alive-Mechanismus aus (Intervallzeit=0)
  oder
- Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung bei Inaktivität beendet wird.

Beschreibung

- **Web Based Management [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom WBM ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.
- **CLI (TELNET, SSH) [s]**
  Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom CLI ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.

Taster

Funktionalität

Der SET-Taster dient zum:

- Neustart
- Laden einer neuen Firmware,
- Zurücksetzen auf Werkseinstellungen.

Eine detaillierte Beschreibung der Funktionen, finden Sie in der Betriebsanleitung des Geräts.

Auf dieser Seite kann die Funktionalität des Tasters eingeschränkt werden.

Beschreibung

Folgende Funktionalität ist möglich:

- **Neustart / Auf Werkseinstellungen zurücksetzen**
  Wenn deaktiviert, kann der SET-Taster nicht zum Neustart oder zum Zurücksetzen auf Werkseinstellungen verwendet werden.

  ⚠ VORSICHT

  Tasterfunktion "Neustart / Auf Werkseinstellungen zurücksetzen" beim Hochlauf aktiv

  Wenn Sie diese Funktion in ihrer Projektierung deaktiviert haben, ist die Deaktivierung nur im laufenden Betrieb gültig. Bei einem Hochlauf, z.B. nach "Stromaus", ist die Funktion bis zum Laden der Projektierung aktiv und das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät dann neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferungszustand versetzt.

  Weitere Informationen, wie Sie das Gerät trotz deaktivierter Funktionen auf Werkseinstellungen zurücksetzen können, finden im Projektierungshandbch Sie im Kapitel "Instandhalten und Warten".

Syslog-Client

Systemereignis-Agent

Syslog nach RFC 3164 wird für die Übermittlung von kurzen, unverschlüsselten Textmeldungen per UDP im IP-Netz verwendet. Dazu wird ein Syslog-Server benötigt.

Voraussetzungen für das Versenden der Log-Einträgen

- Die Syslog-Funktion ist im Gerät aktiviert.
- Die Syslog-Funktion für das jeweilige Ereignis ist aktiviert.

- In Ihrem Netz befindet sich ein Syslog-Server, der die Log-Einträge entgegen nimmt. Da es sich um eine UDP-Verbindung handelt, gibt es keine Rückmeldung an den Absender.
- Die IP-Adresse oder der FQDN (Fully Qualified Domain Name) des Syslog-Servers ist im Gerät eingetragen.

Beschreibung

Die Seite enthält folgende Felder:

- **Syslog-Client**
  Aktivieren oder deaktivieren Sie die Syslog-Funktion.
- **Adresse des Syslog-Servers**
  Geben Sie die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers ein.

Die Tabelle enthält folgende Spalten:

- **Adresse des Syslog-Servers**
  Zeigt die IP-Adresse oder den FQDN (Fully Qualified Domain Name) des Syslog-Servers an.
- **Server-Port**
  Geben Sie den verwendeten Port des Syslog-Servers ein.

Fehlerkontrolle

Link Change

Konfiguration der Fehlerüberwachung von Zustandsänderungen bei Verbindungen

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

Bei aktivierter Verbindungsüberwachung wird ein Fehler signalisiert,

- wenn an einem Port ein Link vorhanden sein soll und dieser fehlt.
- oder wenn an dem Port kein Link vorhanden sein soll und ein Link erkannt wird.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Es gibt folgende Möglichkeiten:
  - "-" (Deaktiviert)
  - Up
  - Down
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Port**
  Zeigt die verfügbaren Ports an.
- **Einstellung**
  Wählen Sie die Einstellung aus. Sie haben folgende Möglichkeiten:
  - Up
    Die Fehlerbehandlung wird beim Übergang in den aktiven Zustand des Ports ausgelöst.
    (Von "Link down" nach "Link up")
  - Down
    Die Fehlerbehandlung wird beim Übergang in den inaktiven Zustand des Ports ausgelöst.
    (Von "Link up" nach "Link down")
  - "-" (Deaktiviert)
    Die Fehlerbehandlung wird nicht ausgelöst.

Mobilfunk

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung der SIM-Karte eine Fehlermeldung ausgelöst wird.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **SIM-Karte**
  Zeigt die verfügbaren SIM-Karten an.
- **Einstellung**
  Wählen Sie in der Klappliste die Einstellung aus. Folgende Möglichkeiten haben Sie:
  - "-" (Deaktiviert)
    Die Fehlerbehandlung wird nicht ausgelöst.
  - SIM-Karte fehlt
    Die Fehlerbehandlung wird ausgelöst, wenn keine SIM-Karte gesteckt ist. Ein Fehler führt zum Aufleuchten der Fehler-LED am Gerät.

PLUG

Konfiguration

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.

Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt. Bei SCALANCE W werden in diesem Fall die verfügbaren Funkschnittstellen deaktiviert.

Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurückgesetzt.

Informationen über die Konfiguration des C-PLUG

Diese Seite liefert Detailinformationen über die Konfiguration, die im C-PLUG abgelegt ist. Darüber hinaus gibt es die Möglichkeit, den PLUG auf Werkseinstellungen zurückzusetzen oder mit einem neuen Inhalt zu versehen.

Hinweis

Die Aktion wird erst dann durchgeführt, wenn Sie auf die Schaltfläche "Einstellungen übernehmen" klicken.

Die Aktion kann nicht rückgängig gemacht werden.

Wenn Sie sich nach der Auswahl gegen die Ausführung entscheiden, dann klicken Sie auf die Schaltfläche "Aktualisieren". Dadurch werden die Daten dieser Seite aus dem Gerät neu ausgelesen und Ihre Auswahl wird aufgehoben.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zum Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da die sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Beschreibung

Die Tabelle gliedert sich in folgende Zeilen:

- **Status**
  Zeigt den Status des PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Es ist ein PLUG mit einer gültigen und passenden Konfiguration im Gerät vorhanden.
  - NOT ACCEPTED
    Ungültige bzw. inkompatible Konfiguration auf dem gesteckten PLUG.
  - NOT PRESENT
    Im Gerät ist kein C-PLUG oder KEY-PLUG gesteckt.
  - FACTORY
    PLUG ist gesteckt und enthält keine Konfiguration. Dieser Status wird auch angezeigt, wenn der PLUG im Betrieb formatiert wurde.
  - MISSING
    Es ist kein PLUG gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
- **Gerätegruppe**
  Zeigt an, von welcher SIMATIC NET-Produktlinie der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Gerätetyp**
  Zeigt den Gerätetyp innerhalb der Produktlinie an, von dem der C-PLUG bzw. KEY-PLUG im vorangegangenen Betrieb genutzt wurde.
- **Version der Konfiguration**
  Die Version der Konfigurationsstruktur. Diese Angabe betrifft die vom Gerät unterstützten Konfigurationsmöglichkeiten und hat nichts mit der konkreten Hardware-Konfiguration zu tun. Diese Revisionsangabe ändert sich also nicht, wenn Sie Zusatzkomponenten (z.B. Module bzw. Extender) hinzufügen oder entfernen, sie kann sich aber ändern, wenn Sie ein Firmware-Update durchführen.
- **Dateisystem**
  Zeigt den Typ des Dateisystems an, das auf dem PLUG vorhanden ist.
- **Verfügbarer Speicherplatz [Byte]**
  Zeigt die maximale Speicherkapazität des Dateisystems an, das auf dem PLUG vorhanden ist.
- **Belegter Speicherplatz [Byte]**
  Zeigt den belegten Speicherplatz im Dateisystem des PLUG an.
- **Firmware auf PLUG**
  Wenn aktiviert, wird die Firmware auf dem PLUG abgespeichert. Damit können mit dem PLUG automatische Firmware-Updates/Downgrades durchgeführt werden.

- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.
  Wenn ein PLUG als PRESET-PLUG konfiguriert wurde, wird dies hier als Zusatzinformation in der ersten Zeile angezeigt. Nähere Informationen zur Erstellung und Benutzung eines PRESET-PLUG finden Sie in Kapitel "Instandhalten und Warten (Seite 363)".
- **PLUG ändern**
  Wählen Sie die gewünschte Einstellung. Sie haben folgende Möglichkeiten, um die Konfiguration auf dem C-PLUG bzw. KEY-PLUG zu ändern:
  - Aktuelle Konfiguration auf den PLUG schreiben
    Diese Option ist nur verfügbar, wenn der Status des PLUG "NOT ACCEPTED" oder "FACTORY" ist.
    Die im internen Flash-Speicher des Gerätes vorhandene Konfiguration wird auf den PLUG kopiert.
  - PLUG auf Werkseinstellungen zurücksetzen
    Löscht alle Daten vom PLUG und führt eine Low-Level-Formatierung durch.

Lizenz

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

ACHTUNG

C-PLUG / KEY-PLUG nicht im laufenden Betrieb ziehen oder stecken!

Ein PLUG darf nur bei ausgeschaltetem Gerät entnommen oder eingesetzt werden.
Das Gerät überprüft im Sekundenabstand, ob ein PLUG gesteckt ist. Wird festgestellt, dass der PLUG entfernt wurde, erfolgt ein Neustart. War in dem Gerät ein gültiger KEY-PLUG gesteckt, wird das Gerät nach dem Neustart in einen definierten Fehlerzustand versetzt. Bei SCALANCE W werden in diesem Fall die verfügbaren Funkschnittstellen deaktiviert.

Wenn das Gerät einmal mit einem PLUG konfiguriert wurde, kann das Gerät ohne diesen PLUG nicht mehr genutzt werden. Um das Gerät wieder nutzen zu können, setzen Sie das Gerät auf Werkeinstellungen zurück.

Hinweis

Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "NOT ACCEPTED", da sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden.

Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell über "System > PLUG" gelöscht oder neu beschrieben werden.

Informationen über die Lizenz des KEY-PLUG

Ein C-PLUG kann nur die Konfiguration eines Geräts speichern. Ein KEY-PLUG enthält zusätzlich zur Konfiguration eine Lizenz, die bestimmte Funktionen Ihres SIMATIC NET-Geräts freischaltet.

Hinweis

Informationen nur Online verfügbar

Die Informationen auf dieser Seite können sie nur einsehen, wenn eine Online-Verbindung zum Gerät besteht.

Beschreibung

- **Status**
  Zeigt den Status des KEY-PLUG an. Es gibt die folgenden Möglichkeiten:
  - ACCEPTED
    Der im Gerät vorhandene KEY-PLUG enthält eine passende und gültige Lizenz.
  - NOT ACCEPTED
    Die Lizenz des gesteckten KEY-PLUG ist nicht gültig.
  - NOT PRESENT
    Im Gerät ist kein KEY-PLUG gesteckt.
  - MISSING
    Es ist kein KEY-PLUG oder ein C-PLUG mit dem Status "FACTORY" gesteckt. Im Gerät sind Funktionen konfiguriert, für die eine Lizenz erforderlich ist.
  - WRONG
    Der gesteckte KEY-PLUG passt nicht zum Gerät.
  - UNKNOWN
    Unbekannter Inhalt des KEY-PLUG.
  - DEFECTIVE
    Der Inhalt des KEY-PLUG ist fehlerhaft.
- **Artikelnummer**
  Zeigt die Artikelnummer des KEY-PLUG an. Es gibt den KEY-PLUG für unterschiedliche Funktionserweiterungen und für verschiedene Zielsysteme.
- **Seriennummer**
  Zeigt die Seriennummer des KEY-PLUG.
- **Info**
  Zeigt zusätzliche Informationen über das Gerät an, das den KEY-PLUG im vorangegangenen Betrieb genutzt hatte, z. B. Artikelnummer, Typenbezeichnung sowie die Ausgabestände von Hard- und Software. Der angezeigte Software-Ausgabestand entspricht dem Ausgabestand, in dem zuletzt die Konfiguration geändert wurde. Beim Status "NOT ACCEPTED" werden weitere Informationen zur Problemursache angezeigt.

Hinweis

Beim Speichern der Konfiguration wird die Information mitgespeichert, ob zu diesem Zeitpunkt ein KEY-PLUG im Gerät gesteckt war. Diese Konfiguration ist dann auch nur lauffähig, wenn ein KEY-PLUG mit der gleichen Artikelnummer / Lizenz gesteckt ist..

SMS

Allgemein

Damit das Gerät eine SMS versenden kann, müssen Sie eine bestimmte SMS-Zentrale (Short Message Service Center, SMS-C) Ihres Mobilfunkanbieters oder Ihres Service-Providers eintragen.

- Wenn Sie die Standard-SMS-Zentrale verwenden, müssen Sie auf dieser Seite nichts konfigurieren. Die SIM-Karte enthält bereits die richtigen Informationen.
- Wenn Sie nicht die Standard-SMS-Zentrale verwenden, können Sie mit dieser Seite die gespeicherte Rufnummer der Standard-SMS-Zentrale durch eine andere ersetzen. Das ist abhängig von Ihrem Vertrag.

Beschreibung

Die Seite enthält folgende Felder:

- **Aktuelle SMS Service·Center Rufnummer** (nur Online verfügbar)
  Zeigt die gespeicherte Rufnummer der SMS-Zentrale an.
- **SMS Service·Center Rufnummer überschreiben**
  Tragen Sie die Rufnummer der SMS-Zentrale inklusive Landesvorwahl ein. Bestätigen Sie die Eingabe, um die die Rufnummer auf der SIM-Karte zu speichern. Wird die Rufnummer bei "Aktuelle·SMS·Service·Center·Rufnummer" angezeigt, wurde die gespeicherte Rufnummer der SMS-Zentrale überschrieben und das Eingabefeld geleert.

  Hinweis

  Wenn die Änderung wirksam wird, wird die Mobilfunk-Verbindung für kurze Zeit unterbrochen. Das Ändern kann bis zu einer Minute dauern.

Ereignis-SMS

Beim Auftreten von Ereignissen kann das Gerät automatisch eine SMS versenden.

Voraussetzung:

- Unter "System > Ereignisse > Konfiguration" ist beim entsprechenden Ereignis "SMS" aktiviert.

Beschreibung

Die Seite enthält Folgendes:

- **Ereignis-SMS aktivieren**
  Wenn aktiviert, dann versendet das Gerät eine SMS.

Im Bereich "SMS-Nachrichten für Digitalen Eingang anpassen" legen Sie den SMS-Text für den digitalen Eingang fest.

- **Digitaler Eingang**
  Zeigt die verfügbaren digitalen Eingänge an.
- **Steigende Flanke**
  Geben Sie den SMS-Text ein, der gesendet wird, wenn am digitalen Eingang das Signal von 0 (LOW) nach 1 (HIGH) wechselt.
- **Fallende Flanke**
  Geben Sie den SMS-Text ein, der gesendet wird, wenn am digitalen Eingang das Signal von 1 (HIGH) nach 0 (LOW) wechselt.

Hinweis

Erlaubte Zeichen für den SMS-Text

In dem Text sind folgende Zeichen erlaubt:

- 0123456789
- A...Z a...z
- Leerzeichen
- ! % & / ( ) = * + < > ' , . -

- **Sendeoption**
  Legen Sie fest, wann der SMS-Text versendet wird.
  - Keine
    Die Funktion ist deaktiviert.
  - Beide
    In beiden Fällen wird ein SMS-Text versendet.
  - Nur steigend
    Nur der SMS-Text wird gesendet, wenn am digitalen Eingang das Signal von 0 (LOW) nach 1 (HIGH) wechselt.
  - Nur fallend
    Nur der SMS-Text wird gesendet, wenn am digitalen Eingang das Signal von 1 (HIGH) nach 0 (LOW) wechselt.
- **Rufnummer**
  Geben Sie die vollständige Telefonnummer des Empfängers ein, das heißt inklusive der Landesvorwahl.
- Die Tabelle enthält folgende Spalten:
  - **Nr.**
    Zeigt die eindeutige Nummer der Telefonnummer an.
  - **Senden**
    Legen Sie fest, ob das Gerät eine SMS an diesen Empfänger sendet.
  - **Rufnummer**
    Zeigt die Telefonnummer des Empfängers an.

Befehl-SMS

Befehle können auch per SMS abgesetzt werden. Auf dieser Seite legen Sie fest, von wem das Gerät gesteuert werden kann. Zur Identifikation wird entweder die Rufnummer oder die Absenderkennung angegeben. Die Absenderkennung kann z. B. eine IP-Adresse oder ein DNS-Name sein.

Mit einer Befehl-SMS können Sie z. B. eine VPN-Verbindung zu einer Anwendung wie dem SINEMA RC-Server aufbauen. Dazu sendet der SINEMA RC-Server eine Weck-SMS, die die notwendigen Befehle enthält. Weiterführende Informationen finden Sie im Projektierungshandbuch unter "Befehl-SMS" oder im Getting Started.

Beschreibung

Die Seite enthält Folgendes:

- **Befehl-SMS aktivieren**
  Aktivieren oder deaktivieren Sie das Empfangen von Befehl-SMS.
- **Rufnummer / Absenderkennung**
  Geben Sie die Absenderkennung oder die Rufnummer inklusive der Landesvorwahl an. Die Rufnummern können Sie mit dem Platzhalter Stern (*) beliebig viele Ziffern nach rechts ersetzen. Der Platzhalter kann nur am Ende einer Ziffernfolge stehen.
  Beispiel:
  + 49144545* steht für alle Rufnummern, die mit 0144545 beginnen.

  Hinweis

  Bitte beachten Sie, wenn Sie den Platzhalter "*" verwenden, werden Befehl-SMS von einem großen Bereich an Rufnummern akzeptiert. Es wird empfohlen eine möglichst restriktive Eingabe zu verwenden.

Die Tabelle enthält folgende Spalten:

- **Nr.**
  Zeigt die eindeutige Nummer des Eintrags an.
- **Rufnummer / Absenderkennung**
  Zeigt die Absenderkennung oder die Rufnummer an.
- **System / Relay**
  Legen Sie fest, welche Klasse (System / Relay) an Befehlen von diesem Absender akzeptiert werden.

SMS-Relay (Ausgehend)

Mit der Funktion SMS-Versand können Anwendungen SMS verschicken, die an der Ethernet-Schnittstelle des Geräts angeschlossen sind. Zum Versand einer SMS muss die Anwendung an der Ethernet-Schnittstelle eine TCP/IP-Verbindung zum Gerät aufbauen. Über diese TCP/IP-Verbindung sendet die Anwendung den Text der SMS an das Gerät, das den Text in eine SMS verpackt und verschickt.

Format des SMS-Textes

Der Text wird in einem Telegramm über die TCP/IP-Verbindung an das Gerät übermittelt, das folgendem Format entspricht:

Benutzername#Passwort#CommandCode#Seq-Num;Rufnummer;Nachricht:

`Beispiel: benutzer#passwort#105#01;0049xxxxxxxxx;mein SMS Text:`

- **Benutzername**
  Geben Sie den Benutzernamen zur Prüfung der Sendeberechtigung einer SMS ein. Maximal 10 Zeichen.
- **Passwort**
  Geben Sie das zugehörige Passwort zum Benutzernamen ein. Maximal 10 Zeichen.
- **CommandCode**
  Kommando zum SMS-Versand aus dem lokalen Netz. Dieser Wert ist fest 105 und darf nicht verändert werden.
- **Seq-Num**
  Die Sequenznummer dient der Zuordnung mehrerer Anfragen gleichzeitig. Die Funktion wird derzeit nicht unterstützt.
  Die Sequenznummer besteht aus 2 nummerischen Zeichen von 01 bis 99.
- **Rufnummer**
  Rufnummer des SMS-Empfängers mit maximal 40 Zeichen. Internationale Nummern (+49...) sind zulässig.
- **Nachricht**
  SMS-Text mit maximal 160 Zeichen

Hinweis

Erlaubte Zeichen für den SMS-Versand

In dem Text sind folgende Zeichen erlaubt:

- 0123456789
- A...Z a...z
- Leerzeichen
- ! " % & / ( ) = ? * + < > ' , . -

Beschreibung

Die Seite enthält Folgendes:

- **SMS-Relay (Ausgehend) aktivieren**
  Aktivieren oder deaktivieren Sie das Versenden von SMS aus dem lokalen Netzwerk.
- **Benutzer**
  Tragen Sie den Benutzernamen ein, der enthalten sein muss, bevor der Text per SMS versendet wird.
- **Passwort**
  Tragen Sie das Passwort ein, das enthalten sein muss, bevor der Text per SMS versendet wird.
- **Passwort bestätigen**
  Wiederholen Sie das Passwort, um es zu bestätigen.
- **Server-Portnummer**
  Tragen Sie den Port ein, auf dem der Server die SMS entgegennimmt.

SMS-Relay (Eingehend)

Anwendungen können SMS empfangen, die an der Ethernet-Schnittstelle des Geräts angeschlossen sind. Zum Empfang einer SMS muss die Anwendung an der Ethernet-Schnittstelle eine TCP/IP-Verbindung zum Gerät aufbauen. Das Gerät empfängt eine Befehl-SMS und verschickt den enthaltenen Text weiter an die Anwendung.

Format des SMS-Textes

Der Text wird in einem Telegramm über die TCP/IP-Verbindung an die Anwendungen gesendet, das folgendem Format entspricht:

Benutzername#Password#CommandCode#Seq-Num;Rufnummer;Nachricht:

```
Beispiel: benutzer#passwort#105#01;0049xxxxxxxxx;mein SMS Text:
```

- **Benutzername**
  Geben Sie den Benutzernamen zur Prüfung des Empfangs der Nachricht ein. Maximal 10 Zeichen.
- **Passwort**
  Geben Sie das zugehörige Passwort zum Benutzernamen ein. Maximal 10 Zeichen.
- **CommandCode**
  Kommando zum SMS-Versand in das lokale Netz. Dieser Wert ist fest 105.
- **Seq-Num**
  Die Sequenznummer dient der Zuordnung mehrerer Anfragen gleichzeitig. Die Funktion wird derzeit nicht unterstützt.
  Die Sequenznummer besteht aus 2 nummerischen Zeichen von 01 bis 99.
- **Rufnummer**
  Rufnummer des Absenders der Befehl-SMS mit maximal 40 Zeichen. Internationale Nummern (+49...) sind zulässig.
- **Nachricht**
  SMS-Text mit maximal 160 Zeichen

Hinweis

Erlaubte Zeichen für den SMS-Versand

In dem Text sind folgende Zeichen erlaubt:

- 0123456789
- A...Z a...z
- Leerzeichen
- ! " % & / ( ) = ? * + < > ' , . -

Beschreibung

Die Seite enthält Folgendes:

- **Verbindungsname**
  Tragen Sie einen eindeutigen Namen für die Relay-Verbindung ein.
- **IP-Adresse**
  Tragen Sie die IP-Adresse des Empfängers ein. An diese IP-Adresse wird das Telegramm gesendet.

- **Port-Nummer**
  Tragen Sie die Port ein, an den das Telegramm gesendet wird.
- **Benutzername**
  Tragen Sie den Benutzernamen zur Prüfung des Empfangs der Nachricht ein. Der Benutzernamen wird in das Telegramm eingetragen.
- **Passwort**
  Tragen Sie das zum Benutzernamen zugehörige Passwort ein.
- **Passwort bestätigen**
  Wiederholen Sie das Passwort, um es zu bestätigen.

Die Tabelle enthält folgende Spalten:

- **Verbindungsname**
  Zeigt den Namen der Relay-Verbindung an.
- **IP-Adresse**
  Zeigt die IP-Adresse des Empfängers an.
- **Portnummer**
  Zeigt den Port an.
- **Benutzername**
  Zeigt den Benutzernamen an, der im Telegramm enthalten ist.

DNS

DNS-Client

Der DNS-Server (Domain Name System) ordnet einer IP-Adresse einen Domainnamen zu, sodass ein Gerät eindeutig identifiziert werden kann.

Wenn diese Einstellung aktiviert ist, kann das Gerät als DNS-Client mit einem DNS-Server kommunizieren.

Beschreibung

- **DNS-Client**
  Aktivieren oder deaktivieren Sie, dass das Gerät als DNS-Client fungiert.
- **Verwendete DNS-Server**
  - learned only
    Das Gerät verwendet nur die durch DHCP zugewiesenen DNS-Server.
  - manual only
    Das Gerät verwendet nur die manuell projektierten DNS-Server. Die DNS-Server müssen mit dem Internet verbunden sein.
  - all
    Das Gerät verwendet alle verfügbaren DNS-Server.
- **Adresse des DNS-Servers**
  Tragen Sie die IP-Adresse des DNS-Servers ein.

Die Tabelle gliedert sich in folgende Spalten:

- Adresse des DNS-Servers
  Zeigt die IP-Adresse des DNS-Servers an.
- Erstellung
  Hier wird angezeigt, ob der DNS-Server manuell konfiguriert wurde oder durch DHCP zugewiesen wurde.

DNS-Proxy

Das Gerät stellt dem lokalen Netz einen DNS-Server zur Verfügung. Wenn Sie in der lokalen Anwendung die IP-Adresse des Gerätes als DNS-Server eintragen, beantwortet das Gerät DNS-Anfragen aus seinem Cache.

Wenn das Gerät die IP-Adresse zu einer Domain-Adresse nicht kennt, leitet es die Anfrage an einen externen DNS-Server weiter. Wie lange das Gerät eine Domain-Adresse im Cache behält, ist abhängig vom adressierten Host. Die DNS-Anfrage an einen externen DNS-Server liefert außer der IP-Adresse auch die Lebensdauer dieser Information zurück.

Beschreibung

Die Seite enthält folgende Felder:

- DNS-Proxy aktivieren
  Aktivieren oder deaktivieren Sie den Proxy des DNS-Servers.
- Cache Name Errors (NXDOMAIN)
  Aktivieren oder deaktivieren Sie das Zwischenspeichern von NXDOMAIN-Antworten. Wenn Sie die Option aktivieren, verbleiben auch die Domain-Namen im Cache, die dem DNS-Server unbekannt waren.

DDNS-Client

Der DDNS (Dynamic Domain Name System) ist ein Internetdienst, der es ermöglicht, einen festen Hostnamen als Pseudonym für eine sich dynamisch ändernde IP-Adresse einzurichten.

Der DDNS-Client synchronisiert die zugewiesene IP-Adresse mit dem im DDNS-Provider registrierten Hostnamen. Damit ist das Gerät immer unter demselben Hostnamen erreichbar.

Voraussetzung:

- Benutzernamen und Passwort, dass Sie zur Nutzung des DDNS-Dienstes berechtigt.
- Registrierter Hostname z. B. example.no-ip.com
- Der UDP-Port 53 für DNS ist freigeschaltet und wird nicht bei NAT verwendet.

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Dienst**
  Zeigt an, welche Anbieter unterstützt werden.
- **Aktiviert**
  Wenn aktiviert, meldet sich das Gerät an dem DDNS-Server an.
- **Host**
  Tragen Sie den Hostnamen ein, den Sie für das Gerät mit Ihrem DDNS-Anbieter vereinbart haben, z. B. example.no-ip-com.
- **Benutzername**
  Tragen Sie den Benutzernamen ein, mit dem sich das Gerät am DDNS-Server anmeldet.
- **Passwort**
  Tragen Sie das dem Benutzer zugeordnete Passwort ein.
- **Passwort bestätigen**
  Bestätigen Sie das Passwort.

DHCP

DHCP-Client

Wenn das Gerät als DHCP-Client konfiguriert ist, startet es eine DHCP-Anfrage. Das Gerät erhält vom DHCP-Server als Antwort eine IPv4-Adresse zugewiesen. Der Server verwaltet einen Adressbereich, aus welchem er IPv4-Adressen vergibt. Es ist auch möglich, den Server so zu konfigurieren, dass der Client auf seine Anfrage immer dieselbe IPv4-Adresse zugewiesen bekommt.

Beschreibung

Die Seite enthält folgende Felder:

- **DHCP-Client Konfigurationsanfrage (Opt. 66, 67)**
  Wenn aktiviert, verwendet der DHCP-Client die Optionen dazu, die Konfigurationsdatei (Option 67) vom TFTP-Server (Option 66) herunterzuladen. Nach dem Neustart verwendet das Gerät die Daten aus der Konfigurationsdatei.

  Hinweis

  Konfigurationsdatei und Firmware-Version

  Die Konfigurationsdatei dient zum Abspeichern und Einlesen von Konfigurationsdaten innerhalb einer Firmware-Version z. B. 4.3. Konfigurationsdateien, die mit einer Firmware-Version <4.2 erstellt wurden, können nicht auf einem Gerät mit einer Firmware-Version 4.3 eingelesen werden.

  Hinweis

  Wird eine Konfigurationsdatei heruntergeladen, so kann dies einen Neustart des Systems auslösen. Wenn sich die aktuell laufende Konfiguration und die Konfiguration in der heruntergeladenen Konfigurationsdatei unterscheiden, startet das System neu.

  Achten Sie darauf, dass in dieser Konfigurationsdatei die Option "DHCP-Client Konfigurationsanfrage (Opt. 66, 67)" nicht mehr gesetzt ist.

- **DHCP-Modus**
  Legen Sie fest, mit welcher Art von Kennung sich der DHCP-Client bei seinem DHCP-Server anmeldet:
  - über MAC-Adresse
    Die Identifikation läuft über die MAC-Adresse ab.
  - über DHCP-Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab.
  - über Systemname
    Die Identifikation läuft über den Systemnamen ab. Ist der Systemname 255 Zeichen lang, dann wird das letzte Zeichen nicht zur Identifikation benutzt.
  - über IAID und DUID
    Damit kann sich der DHCP-Client an DHCP-Servern anmelden, die den Parallelbetrieb von IPv4-und IPv6 unterstützten.
    Die Identifikation läuft über die IAID und die DUID ab und bezeichnet genau eine IP-Schnittstelle des Geräts.
    IAID (Interface Association Identifier): Für jede IP-Schnittstelle wird mindestens eine IAID generiert. Die IAID bleibt bei Neustart des DHCP-Clients unverändert.
    DUID (DHCP Unique Identifier): Identifiziert Server und Clients eindeutig und gilt für alle IP-Schnittstellen des Geräts. Die DUID bleibt bei Neustart unverändert. Es sei denn, der Benutzer ändert diese.

- **DUID-Typ**
  Legen Sie fest, welcher DUID-Typ verwendet wird. Die DUID-Typen sind in der RFC 3315 definiert.
  - DUID-LLT
    DUID basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel
  - DUID-EN
    DUID wird vom Hersteller (EN = Enterprise number) vergeben
  - DUID-LL
    DUID basiert auf der Link-layer-Adresse der Schnittstelle
- **Link-layer Adresse plus Zeit (LLT)**
  Der Wert basiert auf der Link-layer-Adresse der Schnittstelle und einem Zeitstempel. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Unternehmensnummer des Herstellers (EN)**
  Der Wert basiert auf der Unternehmensnummer, die spezifisch für den Hersteller ist. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.
- **Link-layer Adresse (LL)**
  Die Link-layer-Adresse basiert auf der MAC-Adresse. Der Wert wird nach jedem Zurücksetzen auf die Werkseinstellungen neu generiert. Bei Bedarf kann der Wert geändert werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht.
- **DHCP**
  Aktivieren oder deaktivieren Sie den DHCP-Client für die entsprechende Schnittstelle.
- **IAID-Wert**
  Wert mit dem sich die Schnittstelle (DHCP-Client) am DHCP-Server identifiziert.

DHCP-Server

Das Gerät können Sie als DHCP-Server betreiben. Damit ist es möglich, den angeschlossenen Geräten automatisch IP-Adressen zuzuweisen. Die IP-Adressen werden entweder dynamisch aus einem von Ihnen vergebenen Adressband (Pool) verteilt oder es wird eine bestimmte IP-Adresse einem bestimmten Gerät zugewiesen.

Hinweis

DHCP-Server-Zuordnungen löschen

Wenn Sie ein IPv4-Adressband deaktivieren bzw. löschen oder den DHCP-Server aus- und wieder einschalten, werden die DHCP-Server-Zuordnungen gelöscht, siehe "Information > DHCP-Server".

Voraussetzung

Die angeschlossenen Geräte sind so konfiguriert, dass diese die IP-Adresse von einem DHCP-Server beziehen.

Beschreibung

Die Seite enthält folgende Felder:

- **DHCP-Server**
  Aktivieren oder deaktivieren Sie den DHCP-Server auf dem Gerät.

  Hinweis

  Damit keine Konflikte mit IPv4-Adressen entstehen, darf im Netzwek nur ein Gerät als DHCP-Server konfiguriert sein.

  Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, beachten Sie das Kapitel "Layer 3 (IPv4/IPv6) > VRRP/VRRPv3 > Router".

- **Adresse vor dem Anbieten mit ICMP-Echo prüfen**
  Wenn aktiviert, prüft der DHCP-Server, ob die IP-Adresse schon vergeben ist. Dazu sendet der DHCP-Server ICMP-Echomeldungen (Ping) an die IPv4-Adresse. Wenn keine Antwort zurückkommt, wird die IPv4-Adresse vergeben.

  Hinweis

  Bei statischen Zuordnungen wird diese Prüfung nicht durchgeführt.

  Hinweis

  Wenn es in Ihrem Netzwerk Geräte gibt, bei denen der Echo-Dienst standardmäßig deaktiviert ist, kann es zu Konflikten bei den IPv4-Adressen kommen. Um dies zu vermeiden, vergeben Sie diesen Geräten eine IPv4-Adresse, die außerhalb des IPv4-Adressbands liegt.

Die Tabelle gliedert sich in folgende Spalten:

- **Pool-ID**
  Zeigt die Nummer des IPv4-Adressbands an.
- **Schnittstelle**
  Wählen Sie eine IP-Schnittstelle oder einen Router-Port aus. Über diese Schnittstelle werden die IPv4-Adressen dynamisch vergeben.
- **Aktivieren**
  Legen Sie fest, ob dieses IPv4-Adressband verwendet wird.

  Hinweis

  Wenn Sie die IPv4-Adresse aktivieren, werden deren Einstellungen in diesem DHCPRegister ausgegraut und sind nicht mehr editierbar.

- **Subnetz**
  Tragen Sie die Subnetzmaske passend zu der IPv4-Adresse ein. Verwenden Sie die CIDRSchreibweise.

- **Untere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die den Anfang des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Obere IP-Adresse**
  Tragen Sie die IPv4-Adresse ein, die das Ende des dynamischen IPv4-Adressbands festlegt. Die IPv4-Adresse muss innerhalb des Netzadressbereichs liegen, den Sie bei "Subnetz" konfiguriert haben.
- **Gültigkeitsdauer [Sek]**
  Legen Sie fest, für wie viele Sekunden die vergebene IPv4-Adresse gültig bleibt. Nachdem die Gültigkeitsdauer zur Hälfte abgelaufen ist, kann der DHCP-Client die vergebene IPv4-Adresse verlängern. Nach Ablauf der gesamten Zeitdauer muss der DHCP-Client eine neue IPv4-Adresse anfordern.

DHCP-Optionen

Auf dieser Seite legen Sie fest, welche DHCP-Optionen der DHCP-Server unterstützt. Die verschiedenen DHCP-Optionen sind im RFC 2132 definiert.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte Adressband aus.
- **Optionswert**
  Geben Sie die Nummer der gewünschten DHCP-Option ein.

  Hinweis

  Unterstützte DHCP-Optionen

  Die DHCP-Optionen 1, 2, 3. 4. 5, 6, 42, 66, 67 werden unterstützt.

  Die DHCP-Optionen 1, 3, 6, 66 und 67 werden automatisch beim Erstellen des IPv4-Adressbands angelegt. Mit Ausnahme der Option 1 sind die Optionen löschbar.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen
- **Pool-ID**
  Zeigt die Nummer des Adressbands an.
- **Optionswert**
  Zeigt Nummer der DHCP-Option an.

- **Schnittstellen-IP verwenden**
  Legen Sie fest, ob die interne IP-Adresse des Geräts verwendet wird oder nicht.
- **Wert**
  Geben Sie den DHCP-Parameter ein, der dem DHCP-Client übergeben wird. Der Inhalt ist abhängig von der DHCP-Option.

Wert	Optionsname		
1	Subnetzmaske	Die Subnetzmaske wird automatisch eingetragen.	Option nicht löschbar.
2	Offset-Zeit	Offset-Zeit zu der koordinierten Weltzeit UTC.	Geben Sie die Offset-Zeit in Sekunden im Hexadezimal-Format an.
3	Router	Die IPv4-Adresse für Router im Subnetz des DHCP-Clients. Wenn das Gerät selbst der Router ist, wird die IPv4-Adresse der Schnittstelle verwendet.	Sie können mehrere IPv4-Adressen durch Komma getrennt angeben.
4	Zeitserver	Die IPv4-Adresse des Zeitservers, die dem DHCP-Client zur Verfügung steht .	
5	Namenserver	Die IPv4-Adresse des Namesservers, die DHCP-Client zur Verfügung steht.	
6	DNS-Server	Die IPv4-Adresse des DNS-Servers, die dem DHCP-Client zur Verfügung steht. Wenn das Gerät selbst der DNS-Server ist, wird die IPv4-Adresse der Schnittstelle verwendet.	
42	NTP-Server	Die IPv4-Adresse des NTP-Servers, die dem DHCP-Client zur Verfügung stehen .	
66	TFTP-Server	Die IPv4-Adresse oder der Hostnamen des TFTP-Servers, die dem DHCP-Client zur Verfügung steht.	Geben Sie die Adresse des TFTP-Servers an.
67	Namen der Boodatei	Der Namen der Boodatei, die der Client vom TFTP-Server herunterlädt.	Geben Sie den Namen der Bootdatei im String-Format an.

Statische Zuordnung

Auf dieser Seite legen Sie fest, dass bestimmten Geräten eine bestimmte IP-Adresse zugewiesen wird. Die Adresszuordnung erfolgt anhand der MAC-Adresse, anhand der Client-ID oder anhand der DUID.

Beschreibung

Die Seite enthält folgende Felder:

- **Pool-ID**
  Wählen Sie das gewünschte Adressband aus.
- **Identifikationsmethode des Clients**
  Wählen Sie die Methode, nach der ein Client identifiziert wird.
  - Ethernet MAC
    Die Identifikation läuft über die MAC-Adresse ab. Tragen Sie bei "Wert" die MAC-Adresse ein. Die MAC-Adresse besteht aus sechs Bytes, die, durch Bindestriche getrennt, hexadezimal notiert werden, z. B. 00-ab-1d-df-b4-1d.
  - Client-ID
    Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab. Tragen Sie bei "Wert" die gewünschte Bezeichnung ein.
  - DUID
    Die Identifikation läuft über DUID und IAID ab. Tragen Sie bei "Wert" die gewünschte Bezeichnung ein, z. B. 00-00-01-C2-00-01-00-01-00-00-00-72-00-1B-1B-B6-32-9D.
- **Wert**
  Tragen Sie den gewünschten Wert ein. Die Eingabe ist abhängig von der gewählten Identifikationsmethode des Clients.

Hinweis

Maximal 20 Einträge sind möglich.

Die Tabelle gliedert sich in folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Pool-ID**
  Zeigt die Nummer des Adressbands an.
- **Identifikationsmethode**
  Zeigt an, mit welcher Methode sich der Client am DHCP-Server identifiziert.
- **Wert**
  Zeigt die MAC-Adresse, die Client-ID oder DUID des Clients an.
- **IP-Adresse**
  Legen Sie die IPv4-Adresse fest, die dem Client zugewiesen wird. Die IPv4-Adresse muss innerhalb des Adressbands liegen.

cRSP/SRS

Hinweis

Common Remote Service Platform (cRSP) / Siemens Remote Service (SRS) ist eine Fernwartungsplattform über die der Fernwartungszugriff durchgeführt wird.

Zur Nutzung der Plattform sind zusätzliche Serviceverträge notwendig und Randbedingungen zu beachten. Bei Interesse an cRSP / SRS wenden Sie sich an Ihren Siemens-Ansprechpartner vor Ort und besuchen Sie folgende Webseite (https://support.industry.siemens.com/de/en/sc/2281).

Auf dieser Seite konfigurieren Sie die Zugangsdaten für SRS / cRSP nach der URI-Syntax. Der Uniform Resource Identifier (URI) ist in der RFC 3986 definiert.

Beschreibung

Die Seite enthält folgende Felder:

- **DDNS für cRSP / SRS aktivieren**
  Aktivieren oder deaktivieren Sie die Nutzung von cRSP / SRS.
- **Update-Intervall**
  Geben Sie die Zeitspanne ein.
- **Serverzertifikat überprüfen**
  Wenn aktiviert, überprüft das Gerät das empfangene Serverzertifikat auf Gültigkeit.

Die Tabelle gliedert sich in folgende Spalten:

- **Index**
  Die Nummer des Eintrags.
- **Schema**
  Identifiziert die Zugriffsmethode und den Ressourcentyp.
  https: Gesicherter Zugriff auf eine Webseite.
- **Authority**
  Enthält die Adresse des Zielservers
- **Pfad**
  Enthält den Zielpfad zur Ressource. Der Zielpfad kann einem Verzeichnisnamen oder Dateinamen entsprechen.
- **Abfrage**
  Eine Anfrage kann Parameterwerte für eine Anwendung enthalten.
- **Frag**.
  Adressiert lokale Teile der Ressource, z. B. das Anker-Attribut einer Webseite.
- **Status** (nur Online verfügbar)
  Zeigt den Status des letzten cRSP / SRS-Zugriffs des Eintrages an.
- **Aktiviert**
  Wenn aktiviert, wird dieser Eintrag verwendet.

Proxy-Server

Auf dieser Seite konfigurieren Sie den Proxy-Server, der von verschiedenen Komponenten verwendet wird, z. B. SINEMA RC.

Beschreibung

- **Proxy-Name**
  Tragen Sie einen Namen für den Proxy-Server ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Name**
  Zeigt den Namen des Proxy-Servers an.
- **Adresse**
  Tragen Sie die IPv4-Adresse des Proxy-Servers ein.
- **Typ**
  Legen Sie die Art des Proxy-Servers fest.
  - HTTP: Proxy-Server nur für Zugriffe über HTTP.
  - SOCKS: Universeller Proxy-Server
- **Port**
  Geben Sie den Port ein, auf dem der Proxydienst läuft.
- **Auth.-Methode**
  Legen Sie die Authentifizierungsmethode fest.
  - None
    Ohne Authentifizierung
  - Basic
    Standardauthentifizierung. Benutzernamen und Passwort werden unverschlüsselt gesendet.
  - NTML (NT LAN Manager)
    Authentifizierung nach NTML Standard (Windows-Benutzeranmeldung)
- **Benutzername**
  Geben Sie den Benutzernamen für den Zugang zum Proxy-Server ein.
- **Passwort**
  Geben Sie das Passwort für den Zugang zum Proxy-Server ein.
- **Passwort bestätigen**
  Geben Sie nochmals das Passwort ein, um es zu bestätigen.

SINEMA RC

Auf dieser Seite konfigurieren Sie den Zugriff zum SINEMA RC-Server.

Hinweis

Diese Funktion ist nur mit KEY PLUG nutzbar.

Beschreibung

Die Seite enthält Folgendes:

- SINEMA RC aktivieren
  - Aktiviert:
    Eine Verbindung zum konfigurierten SINEMA RC-Server wird aufgebaut. Die Felder sind nicht editierbar.
  - Deaktiviert:
    Die Felder lassen sich editieren. Eine eventuell bestehende Verbindung wird abgebaut.

Bereich "Server-Einstellungen"

- SINEMA RC-Adresse
  Geben Sie die IPv4-Adresse oder den DNS-Hostnamen des SINEMA RC-Servers ein.
- SINEMA RC-Port
  Geben Sie den Port ein, über den der SINEMA RC-Servers erreichbar ist.

Bereich "Serverüberprüfung"

- Prüfungsart
  - Fingerabdruck: Die Identität des Servers wird über den Fingerabdruck verifiziert.
  - CA-Zertifikat: Die Identität des Servers wird über das CA-Zertifikat verifiziert
- Fingerabdruck
  Nur bei der Einstellung "Fingerabdruck" notwendig. Geben Sie den Fingerabdruck des Geräts ein. Der Fingerabdruck wird bei der Inbetriebnahme des SINEMA RC-Servers vergeben. Anhand des Fingerabdrucks überprüft das Gerät, ob es sich den korrekten SINEMA RC-Servers handelt. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- CA-Zertifikat
  Nur bei der Einstellung "CA-Zertifikat" notwendig. Wählen Sie das CA-Zertifikat des Servers aus, das zur Signierung des Serverzertifikats verwendet wird. Nur geladene CA-Zertifikate sind auswählbar.

Bereich "Geräteanmeldedaten"

- Geräte-ID
  Geben Sie die Geräte-ID ein. Die Geräte-ID wird beim Konfigurieren des Geräts am SINEMA RC-Server vergeben. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- Geräte-Passwort
  Geben Sie das Passwort ein, mit dem sich das Gerät am SINEMA RC-Server anmeldet. Das Passwort wird beim Konfigurieren des Geräts am SINEMA RC-Server vergeben. Weiterführende Informationen dazu finden Sie in der Betriebsanleitung zum SINEMA RC-Server.
- Geräte-Passwort bestätigen
  Wiederholen Sie das Passwort ein, um es zu bestätigen.

Bereich "Optionale Einstellungen"

- Auto·Firewall/NAT-Regeln
  - Aktiviert
    Für die VPN-Verbindung werden automatisch die Firewall und NAT-Regeln angelegt. Dabei werden die Verbindungen, die zwischen den projektierten exportierten Subnetzen und den Subnetzen, die über den SINEMA RC-Server erreichbar sind, zugelassen. Die NAT-Einstellungen werden wie im SINEMA RC-Server projektiert umgesetzt.
  - Deaktiviert
    Sie müssen selbst die Firewall und NAT-Regeln anlegen.
- Verbindungsart
  Legen Sie die Art der VPN-Verbindung fest. Weitere Informationen dazu finden Sie im Kapitel "VPN-Verbindungsaufbau".
  - Auto
    Das Gerät übernimmt die Einstellungen des SINEMA RC Server. Die Einstellungen auf dem SINEMA RC Server konfigurieren Sie unter "Fernverbindungen > Geräte". Weiterführende Informationen hierzu finden Sie in der Betriebsanleitung "SINEMA RC Server".
  - Permanent
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Das Gerät baut eine VPN-Verbindung zum SINEMA RC-Server. Der VPN-Tunnel wird permanent aufrechterhalten.
  - Weck-SMS (nur bei M87x)
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Wenn das Gerät eine Befehl-SMS (Wake-up SMS) erhält, versucht das Gerät zum SINEMA RC-Server aufzubauen. Vorausgesetzt unter "System > SMS > Befehl-SMS" ist festgelegt, von wem eine Befehl-SMS der Klasse "System" akzeptiert wird.
  - Digitaler Eingang
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Beim Eintreten des Ereignisses "Digital In" versucht das Gerät zum SINEMA RC-Server eine VPN-Verbindung aufzubauen. Vorausgesetzt ist, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Digitaler Eingang & Weck-SMS (nur bei M87x)
    Die Einstellungen des SINEMA RC-Servers werden ignoriert. Wenn das Ereignis "Digitaler Eingang" eintritt oder wenn das Gerät eine Befehl-SMS erhält, versucht das Gerät eine VPN-Verbindung zum SINEMA RC-Server aufzubauen
- Proxy verwenden
  Legen Sie fest, ob Verbindung zu dem definierten SINEMA RC-Server über einen Proxy-Server aufgebaut wird. Es sind nur die Proxy-Server auswählbar, die Sie unter "System > Proxy Server" konfiguriert haben.
- Autokonfigurations-Intervall·[min]
  Geben Sie die Zeitspanne in Minuten an, nach der Anfragen an den SINEMA RC Server gesendet werden. Mit dieser Anfragen prüft das Gerät, ob auf dem SINEMA RC Server eine neuere Firmware-Datei vorhanden ist oder ob sich die Verbindungseinstellungen geändert haben.
  Wenn Sie den Wert 0 eintragen, ist diese Funktion deaktiviert.

Schnittstellen projektieren

Ethernet

Übersicht

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an.

Beschreibung

- **Port**
  Zeigt die konfigurierbaren Ports an.
- **Port-Name**
  Zeigt den Namen des Ports.
- **Port-Typ** (Nur bei Routing)
  Zeigt den Typ des Ports an. Folgende Typen sind möglich:
  - Router-Port
  - Switch-Port VLAN Hybrid
  - Switch-Port VLAN Trunk
- **Status**
  Zeigt an, ob der Port ein- oder ausgeschaltet ist. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
- **Betriebszustand** (Nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.
- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsparameter des Ports an

- **Negotiation**
  Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.

Konfiguration

Auf dieser Seite konfigurieren Sie die Ethernet-Ports des Geräts.

Beschreibung

- **Port**
  Wählen Sie die zu konfigurierenden Port aus. .
- **Status**
  Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled
    Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled
    Der Port ist ausgeschaltet, aber die Verbindung besteht noch.
  - Link down
    Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**
  Tragen Sie hier einen Namen für den Port ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse des Ports an.
- **Übertragungsmodus**
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Wenn Sie die Betriebsart auf "Auto negotiation" stellen, werden diese Parameter automatisch mit dem angeschlossenen Endgerät oder der Netzkomponente ausgehandelt. Dieses muss sich hierzu ebenfalls in der Betriebsart "Auto negotiation" befinden.

  Hinweis

  Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

- **akt. Übertragungsmodus** (Nur Online verfügbar)
  Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Die Anzeige ist abhängig von dem eingestellten "Mode Type".
- **Negotiation**
  Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.

- **Port-Typ** (Nur bei Routing)
  Wählen Sie die Art des Ports aus:
  - Router-Port
    Der Port ist eine IP-Schnittstelle. Er unterstützt keine Layer 2-Funktionen.
  - Switch-Port VLAN Hybrid
    Der Port sendet getaggte und ungetaggte Telegramme. Er ist nicht automatisch Mitglied eines VLANs.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.
- **Betriebszustand** (Nur Online verfügbar)
  Zeigt den aktuellen Betriebszustand an. Der Betriebszustand ist vom konfigurierten "Status" und dem "Link" abhängig.
  Es gibt folgende Möglichkeiten:
  - up
    Sie haben für den Port den Status "enabled" konfiguriert und der Port hat eine gültige Verbindung zum Netzwerk.
  - down
    Sie haben für den Port den Status "disabled" oder "Link down" konfiguriert oder der Port hat keine Verbindung.
- **Link** (Nur Online verfügbar)
  Zeigt den Verbindungsstatus zum Netzwerk am. Beim Verbindungsstatus ist Folgendes möglich:
  - Up
    Der Port hat eine gültige Verbindung zum Netzwerk, es wird ein "Link Integrity Signal" empfangen.
  - Down
    Die Verbindung ist unterbrochen, weil beispielsweise das angeschlossene Gerät ausgeschaltet ist.

Mobilfunk

SIM

Für den Zugang zum Mobilfunknetz und zu den Mobilfunkdiensten sind folgende Zugangsparameter erforderlich. Die Zugangsparameter erhalten Sie von Ihrem Mobilfunkanbieter.

Beschreibung

Die Seite enthält Folgendes:

- **Mobilfunk-Schnittstelle aktivieren**
  Aktivieren oder deaktivieren Sie die Mobilfunk-Schnittstelle.
- **PIN**
  Tragen Sie die PIN der SIM-Karte ein. Die PIN erhalten Sie von Ihrem Mobilfunkanbieter. Das Gerät arbeitet auch mit PIN-losen SIM-Karten, in diesem Fall lassen Sie das Feld leer.

  Hinweis

  Bei Fehleingaben wird die SIM-Karte gesperrt

  Achten Sie darauf, die PIN korrekt einzugeben. Wenn Sie die PIN öfter als 3 Mal falsch eingeben, wird die SIM-Karte gesperrt.

  Unter "Information > Mobilfunk" können Sie den Status der SIM-Karte überprüfen. Wenn bei "SIM-Status" "PUK erforderlich" angezeigt wird, ist die SIM-Karte gesperrt.

  Zum Entsperren muss die Karte aus dem Gerät genommen und in ein Mobilfunktelefon eingesetzt werden. Durch die Eingabe der PUK kann die Karte wieder entsperrt werden. Setzen Sie sich gegebenenfalls mit Ihrem Mobilfunkbetreiber in Verbindung.

- **PIN bestätigen**
  Wiederholen Sie die PIN, um diese zu bestätigen.
- **Funkbetriebsart**
  Wählen Sie das gewünschte Mobilfunknetz aus. Folgende Optionen stehen zur Auswahl:
  - Auto (nicht bei M874-2)
    Alle Dienste. Das Gerät versucht bevorzugt mit dem schnellsten verfügbaren Mobilfunkverfahren eine Verbindung herzustellen.
  - Nur GSM
    Die Dienste EGPRS und GPRS.
    Das Gerät ignoriert das UMTS- und LTE-Netz und baut eine Verbindung zu dem GSM-Dienst auf, der vor Ort die höchste Bandbreite zur Verfügung stellt
  - Nur UMTS (nicht bei M874-2)
    Der Dienst UMTS und HSPA. Das Gerät ignoriert die Dienste EGPRS, GPRS und LTE und baut eine Verbindung im UMTS-Netz auf.
  - Nur LTE (nur beim M876-4 verfügbar)
    Der Dienst LTE. Das Gerät ignoriert die Dienste EGPRS, GPRS und UMTS und baut eine Verbindung im LTE-Netz auf.
  - Keine Datenverbindung
    Es besteht keine Datenverbindung. Der Modus eignet sich für den Fall, wenn Sie das Gerät nur zum Senden oder Empfangen von SMS verwenden möchten.

- **Authentifizierungsmethode**
  Wählen Sie das gewünschte Authentifizierungsverfahren.
  - CHAP
    Verschlüsselte Übertragung von Benutzername und Passwort über das Challenge Handshake Authentication Protocol (CHAP).
  - PAP
    Unverschlüsselte Übertragung von Benutzername und Passwort über das Password Authentication Protocol (PAP).
  - Auto
    Benutzername und Passwort werden automatisch mit einem der beiden Verfahren übertragen. CHAP hat die höhere Priorität. Wenn der Kommunikationspartner CHAP nicht unterstützt, werden Benutzername und Passwort per PAP übertragen.
- **Daten-Roaming ermöglichen**
  Wenn aktiviert, meldet sich das Gerät automatisch bei einem verfügbaren Netz an, falls das festgelegte Netz nicht erreichbar ist.

Mobilfunkanbieter

Der APN (Access Point Name) ist der Name des Zugangspunkts vom Mobilfunknetz zum Internet oder zu einem privaten Firmennetz. Abhängig vom Typ des angeschlossenen Netzes handelt es sich um einen öffentlichen oder privaten APN. Der APN wird vom Mobilfunkanbieter mitgeteilt.

Beschreibung

Die Seite enthält Folgendes:

- **Länderliste** (Nur Online verfügbar)
  Wählen Sie in der Liste das Land aus, in dem das Gerät eingesetzt wird.
- **Anbieter-Liste** (Nur Online verfügbar)
  Wählen Sie in der Liste den entsprechenden Mobilfunkanbieter aus. Die Liste ist abhängig von dem ausgewählten Land.
  Wenn der Mobilfunkbetreiber nicht in der Liste der Provider enthalten ist, wählen Sie "-" aus.
- **PLMNID**
  Jeder Mobilfunkanbieter hat eine weltweit einmalig vergebene Identifikationsnummer, die als Public Land Mobile Network ID (PLMN ID) bezeichnet wird. Sie finden die Net-ID in den Unterlagen Ihres Mobilfunkanbieters oder auf dessen Internetseiten.
  - Wenn Sie die Identifikationsnummer (Net-ID) des Mobilfunkanbieters kennen, geben Sie diese ein.
  - Wenn Sie die PLMNID nicht kennen, geben Sie "Manuell" ein.
- **APN**
  Tragen Sie den Namen des APNs ein. Den APN finden Sie in den Unterlagen Ihres Mobilfunkanbieters, auf dessen Internetseite oder Sie erfragen den APN bei der Hotline Ihres Mobilfunkbetreibers.

- **Benutzername**
  Tragen Sie den Benutzernamen ein. Einige Mobilfunkanbieter verzichten auf die Zugangskontrolle durch Benutzername und/oder Passwort. In diesem Fall lassen Sie das Feld leer.
- **Passwort**
  Tragen Sie das Passwort ein. Einige Mobilfunkanbieter verzichten auf die Zugangskontrolle durch Benutzername und/oder Passwort. In diesem Fall lassen Sie das Feld leer.
- **Passwort bestätigen**
  Wiederholen Sie das Passwort.

Die Tabelle enthält folgende Spalten:

- **PLMNID**
  Zeigt die Identifikationsnummer (Net-ID) des Mobilfunkanbieters an.
- **Name des Anbieters**
  Tragen Sie den Namen Ihres Mobilfunkanbieters ein.
- **APN**
  Tragen Sie den Namen des APNs ein. Den APN finden Sie in den Unterlagen Ihres Mobilfunkanbieters, auf dessen Internetseite oder Sie erfragen den APN bei der Hotline Ihres Mobilfunkbetreibers.
- **Benutzername**
  Tragen Sie den Benutzernamen für den APN ein. Einige Mobilfunkanbieter verzichten auf die Zugangskontrolle durch Benutzername und/oder Passwort. In diesem Fall lassen Sie das Feld leer.
- **Passwort**
  Tragen Sie das Passwort für den APN ein. Einige Mobilfunkanbieter verzichten auf die Zugangskontrolle durch Benutzername und/oder Passwort. In diesem Fall lassen Sie das Feld leer.
- **Passwort bestätigen**
  Wiederholen Sie das Passwort.
- **Aktiviert**
  Der Eintrag wird verwendet.

Verbindungsüberprüfung

Verwenden Sie diese Funktion, um sicherzustellen, dass die Verbindung zum Mobilfunknetz besteht. Auf dieser Seite legen Sie fest, welche WAN-Adressen (IPv4-Adresse oder FQDN) als Referenz für die Verbindungsüberwachung verwendet werden.

Das Gerät sendet in regelmäßigen Abständen (Ping Time Interval) Echomeldungen (Ping) an die projektierten WAN-Adressen.

Wenn Sie z. B. drei WAN-Adressen projektieren, dann sendet das Gerät an die erste WAN-Adresse einen Ping. Wenn diese WAN-Adresse erreichbar ist, sendet diese eine Antwort. Nach Ablauf des projektieren Intervalls sendet das Gerät den nächsten Ping. Wenn die erste WAN-Adresse nicht erreichbar ist, sendet das Gerät nach Ablauf einer Wartezeit (10s) einen weiteren Ping. Insgesamt werden drei Pings an die WAN-Adresse gesendet. Wenn das Gerät nach dem dritten Ping keine Antwort erhält, wird die zweite WAN-Adresse überprüft. Ist auch diese WAN-Adresse nicht erreichbar, wird die nächste WAN-Adresse überprüft.

Wenn keine der drei WAN-Adressen erreichbar ist, startet das Gerät die Mobilfunkschnittstelle neu und protokolliert den Neustart mit folgender Meldung "Detected a problem with the provider connection. Trying to reconnect to the provider. Attempt no.: x".

Das x steht für die Anzahl der Versuche. Insgesamt führt das Gerät 10 Versuche durch. Nach dem Neustart der Mobilfunkschnittstelle wartet das Gerät einige Zeit und sendet dann einen Ping an die erste WAN-Adresse. Die Wartezeit beträgt beim ersten Neustart der Mobilfunkschnittstelle 2 Minuten. Bei jedem weiteren Neustart der Mobilfunkschnittstelle verlängert sich die Wartezeit um eine Minute.

Wenn die 10 Versuche erfolglos sind, startet das Gerät neu und protokolliert den Neustart mit folgender Meldung: "Detected a problem with the provider connection. Reconnect to the provider had no effect. Device will restart now."

Hinweis

Projektieren Sie stets mehrere WAN-Adressen. Wenn Sie nur eine WAN-Adresse projektieren, kann der sporadische Ausfall der einzelnen WAN-Adresse dazu führen, dass die WAN-Verbindung ständig neu aufgebaut wird.

Beschreibung

Die Seite enthält Folgendes:

- **Ping für Verbindung aktivieren**
  Aktivieren oder deaktivieren Sie die Funktion. Nur aktivierbar, wenn mindestens eine WAN-Adresse projektiert ist.
- **Ping Zeitintervall [s]**
  Legen Sie fest, in welchem Abstand die Pings gesendet werden.
- **Ping Ziel-Adresse**
  Tragen Sie die IPv4-Adresse oder FQDN-Adresse ein, die für die Verbindungsüberwachung verwendet wird.

Die Tabelle enthält folgende Spalten:

- **Ziel-Adresse**
  Zeigt die WAN-Adresse an.

DSL

Auf dieser Seite konfigurieren Sie den DSL-Zugang und die Parameter für die virtuelle Verbindung, über die die Pakete übertragen werden.

Beschreibung

Die Seite enthält Folgendes:

- **DSL-Schnittstelle aktivieren**
  Aktivieren oder deaktivieren Sie die DSL-Schnittstelle.
- **PPPoE Passthrough aktivieren**
  - Aktiviert
    Das Gerät agiert als Modem. Die Einstellungen "Account", "Password" und "Enable Force Disconnect" sind nicht editierbar. Nur ein angeschlossenes Gerät kann die DSL-Verbindung nutzen. Dieses Gerät übernimmt auch die Authentifizierung (Einwahl) beim Provider.
  - Deaktiviert
    Das Gerät agiert als Router und meldet sich mit Benutzernamen und Passwort an. Alle angeschlossenen Geräte können die DSL-Verbindung nutzen.
- **Benutzerkonto**
  Tragen Sie den Benutzernamen ein. Den Benutzernamen erhalten Sie von Ihrem DSL-Anbieter.
- **Passwort**
  Tragen Sie das Passwort ein. Das Password erhalten Sie von Ihrem DSL-Anbieter.
- **Passwort bestätigen**
  Wiederholen Sie das Passwort, um es zu bestätigen.
- **VCI** (Virtual Channel Identifier)
  Tragen Sie die ID für den virtuellen Kanal ein. Die Einstellung erhalten Sie von Ihrem DSL-Anbieter.
- **VPI** (Virtual Path Identifier)
  Tragen Sie die ID für den virtuellen Pfad ein. Die Einstellung erhalten Sie von Ihrem DSL-Anbieter.
- **Kapselung**
  Die Datenpakete werden in das gewünschte Protokoll eingepackt.
  Folgende Möglichkeiten gibt es:
  - vc-mux (Virtual circuit - multiplexing)
  - LLC (Logical Link Control)
- **Protokoll**
  Legen Sie das Protokoll für die Internetverbindung fest. Die Information erhalten Sie von ihrem DSL-Anbieter.
  - PPPoE (Point-to-Point over Ethernet)
    Die PPP-Daten werden in einen Ethernet-Frame eingekapselt.
  - PPPoA (Point-to-Point over ATM)
    Die PPP-Daten werden in ATM AAL5 (Adaptation Layer 5) eingekapselt. Wenn Sie PPPoA auswählen, wird die Einstellung "PPPoE Passthrough aktivieren" deaktiviert.
- **Rauschabstand Delta DS:**
  Legen Sie den Abstand zum Grundrauschen fest.
  Wertebereich -5dB ... +5dB

- VLAN-ID aktivieren
  Wenn aktiviert, können Sie eine VLAN-ID konfigurieren.
- VLAN-ID
  Geben Sie die VLAN-ID an. Die Einstellung erhalten Sie von Ihrem DSL-Anbieter.
- Zwangstrennung
  Der DSL-Anbieter trennt nach einem bestimmten Zeitraum die Verbindung. Aktivieren Sie diese Option, wenn Sie die Zwangstrennung durch Ihren Provider auf eine bestimmte Uhrzeit verschieben wollen, z. B. Nachts außerhalb der üblichen Bürozeiten.
- Zeit für Zwangstrennung
  Legen Sie die Uhrzeit fest zu der Sie die Zwangstrennung des DSL-Anbieters verschieben wollen. Vorausgesetzt, im Gerät ist die korrekte Systemzeit eingestellt.
  Eingabeformat: HH:MM

SHDSL

Übersicht

Die Seite zeigt die Konfiguration der SHDSL-Schnittstellen an.

Beschreibung

Die Seite enthält Folgendes:

- PME Aggregation-Funktion aktivieren
  PME = Physical Medium Entities
  Wenn aktiviert, werden die SHDSL-Schnittstellen bzw. die 2-Drathleitungen zu einer einzigen Verbindung mit höherer Übertragungsrate zusammengefasst.

  Hinweis

  Wenn die Funktion "PME Aggregation" aktiviert ist, müssen die Schnittstellen eines Geräts jeweils die gleiche Rolle besitzen.

Die Tabelle enthält folgende Spalten:

- Schnittstelle
  Zeigt die verfügbaren SHDSL-Schnittstellen an.
- Status
  Zeigt an, ob die Schnittstelle verwendet wird oder nicht.
- Rolle
  Zeigt die Rolle der SHDSL-Schnittstelle an:
  - Central Office (CO)
    Die Schnittstelle ist eine Vermittlungsstelle.
  - Customer Premises Equipment (CPE)
    Die Schnittstelle ist ein Teilnehmerendgerät.

- **Target SNR**
  Zeigt das Signal-Rausch-Verhältnis an:
  - Zuverlässig (10 dB)
    Zuverlässige Übertragung
  - Normal (6 dB)
    Normale Übertragung
  - High-speed (0 dB)
    Hochgeschwindigkeits-Übertragung
- **Port-Typ**
  Zeigt den Betriebsmodus des Switch-Ports an:
  - Switch-Port VLAN Hybrid
    Der Port akzeptiert getaggte und ungetaggte Telegramme.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.

Konfiguration

Auf dieser Seite legen Sie die Einstellungen für die Verbindung fest und welche Rolle das Gerät übernimmt. Weiter Information zu den Einstellungen finden Sie in der ITU-T G.991.2.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle**
  Wählen Sie die Schnittstelle aus.
- **Status**
  Legen Sie fest, ob die Schnittstelle verwendet wird.
- **Port -Typ**
  Legen Sie den Betriebsmodus für den Switch-Port fest.
  - Switch-Port VLAN Hybrid
    Der Port akzeptiert getaggte und ungetaggte Telegramme.
  - Switch-Port VLAN Trunk
    Der Port sendet nur getaggte Telegramme und ist automatisch Mitglied in allen VLANs.
- **Rolle**
  Legen Sie die Rolle fest:
  - Central Office (CO)
    Die Schnittstelle ist eine Vermittlungsstelle.
  - Customer Premises Equipment (CPE)
    Die Schnittstelle ist ein Teilnehmerendgerät.

- **Vordefinierte Profile**
  Legen Sie das Profil für die Übertragung fest. Wenn Sie ein Profil verwenden, werden die nachfolgenden Parameter automatisch eingestellt.
  - Profile mit einem Übertragungsbereich
    Wenn beim Einmessen der Leitung der gemessene Wert innerhalb des Bereichs liegt, wird die SHDSL-Verbindung aufgebaut.

	Link-Datenrate min - max	
Standard	192 - 5696	Für mittlere und längere Leitungslänge
Zuverlässig	192 - 5696	Beim gleichzeitigen Betrieb von mehreren SHDSL-Verbindungen.
Erweitert	64 - 15296	Für kurze und lange Leitungslänge

  - Profile mit einer fest vorgegebenen Übertragungsrate
    Nur wenn beim Einmessen der Leitung der gemessene Wert genau der fest vorgegebenen Übertragungsrate entspricht, wird eine SHDSL-Verbindung aufgebaut.

	Link-Datenrate
Fest·(Hohe·Datenrate)	5696
Fest·(Mittlere·Datenrate)	3072
Fest·(Niedrige·Datenrate)	512

- **Erweiterter Modus**
  Aktivieren oder deaktivieren Sie den erweiterten Modus. Abhängig von der Einstellung, ändert Sie die Auswahl in den nachfolgenden Feldern.
- **PAM**
  Pulse Amplitude Modulation
  Legen Sie das Modulationsverfahren fest.
- **Line probing**
  Wenn aktiviert, wird die Leitung gemäß ITU-T G.991.2 eingemessen.
- **SNR-Modell**
  Legen Sie das Modell zum Berechnen des Signal-Rausch-Verhältnisses fest.
  - Aktuelle Bedingung
    Der Wert für Rauschen wird verwendet, der beim Einmessen der Leitung gemessen wurde.
  - Worst Case
    Der Wert für Rauschen wird angenommen, der über mehrere Leitungen eines Kabelbündels auftreten könnte.

- **Target SNR**
  Legen Sie den Übertragungsmodus für das Signal-Rausch-Verhältnis fest. Unter "Information > SHDSL" wird das aktuelle Signal-Rausch-Verhältnis angezeigt.
  - Zuverlässig (10 dB)
    Zuverlässige Übertragung
  - Normal (6 dB)
    Normale Übertragung
  - High-speed (0 dB)
    Hochgeschwindigkeits-Übertragung
- **Min. Link Datenrate [kbps]**
  Legen Sie die minimale Übertragungsrate für das Senden und das Empfangen von Daten fest.
  - Auto (0)
    Das Gerät passt die Übertragungsrate je nach Modus automatisch an.
  - 64 bis 15296
    Die Übertragungsrate ist fest vorgegeben.
- **Max. Link Datenrate [kbps]**
  Legen Sie die maximale Übertragungsrate für das Senden und das Empfangen von Daten fest.
  - Auto (0)
    Das Gerät passt die Übertragungsrate je nach Modus automatisch an.
  - 64 bis 15296
    Die Übertragungsrate ist fest vorgegeben.
- **Leistungsregulierung**
  Legen Sie die fest, ob die Sendeleistung reduziert wird.
  - Deaktiviert
    Die Funktion ist deaktiviert
  - Normal
    Die Funktion ist aktiviert.
  - Erzwingen
    Bei PBO Value geben Sie den Wert für die Dämpfung an.
- **PBO-Wert**
  Power Back Off Values
  Geben Sie den Wert für die Dämpfung an. Nur wenn der Wert erreicht wird, wird eine SHDSL-Verbindung aufgebaut.
  - Bereich: 0 - 31 dB, wobei "0" keine Dämpfung bedeutet.

Verbindungsüberprüfung

Verwenden Sie diese Funktion, um sicherzustellen, dass die Verbindung zum SHDSL-Netz besteht. Auf dieser Seite legen Sie fest, welche WAN-Adressen (IP-Adresse, FQDN) als Referenz für die Verbindungsüberwachung verwendet werden.

Das Gerät sendet in regelmäßigen Abständen (Ping Zeitintervall) Echomeldungen (Ping) an die projektierten WAN-Adressen.

Wenn Sie z. B. drei WAN-Adressen projektieren, dann sendet das Gerät an die WAN-Adressen einen Ping. Wenn nur eine der drei WAN-Adressen nicht erreichbar ist, wird das folgender Meldung protokolliert: "Could not reach remote device a.b.c.d (Failure count x)".

Dabei steht "a.b.c.d" für die IP-Adresse des Geräts und das x steht für die Anzahl der Versuche. Insgesamt führt das Gerät 5 Versuche durch.

Wenn die WAN-Adresse wieder erreichbar ist, wird dies mit folgender Meldung protokolliert: SHDSL connection check: Remote devices are reachable again.

Wenn die 5 Versuche erfolglos sind, startet das Gerät neu und protokolliert den Neustart mit folgender Meldung: "SHDSL connection check: Maximum failure count reached, restarting device."

Beschreibung

Die Seite enthält Folgendes:

- **Ping für Verbindung aktivieren**
  Aktivieren oder deaktivieren Sie die Funktion. Nur aktivierbar, wenn mindestens eine WAN-Adresse projektiert ist.
- **Ping Zeitintervall [s]**
  Legen Sie fest, in welchem Abstand die Pings gesendet werden.
- **Ping Ziel-Adresse**
  Tragen Sie die IPv4-Adresse oder FQDN-Adresse ein, die für die Verbindungsüberwachung verwendet wird.

Die Tabelle enthält folgende Spalten:

- **Ziel-Adresse**
  Zeigt die WAN-Adresse an.

Layer 2-Funktionen projektieren

Konfiguration

Layer 2 konfigurieren

Auf dieser Seite konfigurieren Sie Passive Listening.

Beschreibung

- **Passive Listening**
  Wenn aktiviert, sorgt die Funktion dafür, dass die BPDUs aus dem RSTP-Netzwerk transparent weitergeleitet werden und auch wieder zurückgelangen. Wäre dies nicht der Fall, würde es zur Schleifenbildung an der Verbindungsstelle zwischen RSTP und Ring kommen.

VLAN

Allgemein

Auf dieser Seite legen Sie fest, ob das Gerät Telegramme mit VLAN-Tags transparent weiterleitet (IEEE 802.1D/VLAN-unaware-Modus) oder VLAN-Informationen berücksichtigt (IEEE 802.1Q/VLAN-aware-Modus). Wenn sich das Gerät im Modus "802.1Q VLAN Bridge" befindet, können Sie VLANs definieren und die Verwendung der Ports festlegen.

Hinweis

Ändern der Agent VLAN-ID

Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN-ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

Beschreibung

Die Seite enthält folgende Felder:

- **Base Bridge-Modus**
  Wählen Sie den gewünschten Modus aus. Folgende Modi sind möglich:

  Hinweis

  Base Bridge-Modus wechseln

  Beachten Sie den Abschnitt "Base Bridge-Modus wechseln". In diesem Abschnitt ist beschrieben, wie sich ein Wechsel auf die bestehende Konfiguration auswirkt.

  – 802.1 D Transparent Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLANTags nicht berücksichtigt bzw. verändert, sondern transparent weitergeleitet. Sie können in diesem Modus keine VLANs anlegen.

  – 802.1 Q VLAN Bridge
    Stellt bei dem Gerät den Modus "VLAN-unaware" ein. In diesem Modus werden VLANTags nicht berücksichtigt bzw. verändert, sondern transparent weitergeleitet. Sie können in diesem Modus keine VLANs anlegen.
    Es ist nur ein Management-VLAN verfügbar: VLAN 1.

- **VLAN-ID**
  Tragen Sie die VLAN-ID (eine Zahl zwischen 1 und 4094) ein.
  Sie können bis zu 2 (SCALANCE W760), 8 (SCALANCE W770) oder 24 (SCALANCE W780) VLANS anlegen.

Die Tabelle gliedert sich in folgende Spalten:

- **VLAN-ID**
  Zeigt die VLAN ID an. Die VLAN ID (eine Zahl zwischen 1 und 4094) kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.

- **Name**
  Tragen Sie einen Namen für das VLAN ein. Der Name hat nur informativen Charakter und keine Auswirkungen auf die Konfiguration.

- Status
  Zeigt die Statusart des Eintrags in der internen Portfiltertabelle internen Portfiltertabellean. Dabei bedeutet statisch, dass die Adresse vom Anwender statisch eingetragen wurde. Die Angabe GVRP bedeutet, dass die Konfiguration über ein GVRP-Telegramm registriert wurde. Voraussetzung dafür ist allerdings, dass GVRP für das Gerät aktiviert wurde.
- Liste der Ports
  Legen Sie die Verwendung des Ports fest. Folgende Möglichkeiten gibt es:
  - "-"
  - Der Port ist kein Mitglied des angegebenen VLANs.
    Bei der Neudefinition sind alle Ports mit der Kennung "-" belegt.
  - M
    Der Port ist Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden mit dem entsprechenden VLAN-Tag weitergeleitet.
  - R
    Der Port ist Mitglied des VLANs. Die Registrierung erfolgt über ein GVRP-Telegramm.
  - U (Großbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet. Von diesem Port werden Telegramme ohne VLAN-Tag gesendet.
  - u (Kleinbuchstabe)
    Der Port ist ungetaggtes Mitglied des VLANs, jedoch ist das VLAN nicht als Port-VLAN konfiguriert. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet.
  - F
    Der Port ist kein Mitglied des angegebenen VLANs und es ist nicht möglich, dass das VLAN dynamisch über GVRP an diesem Port registriert werden kann. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".
  - T
    Diese Option wird nur angezeigt und kann nicht ausgewählt werden. Dieser Port ist Trunk-Port und wurde dadurch Mitglied in allen VLANs. Sie konfigurieren diese Funktion im CLI (Command Line Interface) mit Hilfe des Befehls "`switchport mode trunk`" oder im WBM unter "Schnittstellen > Ethernet > Konfiguration".

Base Bridge-Modus wechseln

VLAN-unaware (802.1D Transparent Bridge) → VLAN-aware (802.1Q VLAN Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-unaware in VLAN-aware ädern, hat dies folgende Auswirkungen:

- Alle statischen und dynamischen Unicast-Einträge werden gelöcht.

VLAN-aware (802.1Q VLAN Bridge) → VLAN-unaware (802.1D Transparent Bridge)

Wenn Sie den Base Bridge-Modus von VLAN-aware in VLAN-unaware ädern, hat dies folgende Auswirkungen:

- Alle VLAN-Konfigurationen werden gelöcht.
- Es wird ein Management-VLAN angelegt: VLAN 1.
- Alle statischen und dynamischen Unicast-Einträe werden gelöcht.

802.1Q VLAN Bridge: Wichtige Regeln für VLANs

Berücksichtigen Sie bei der Konfiguration und beim Betrieb Ihrer VLANs folgende Regeln:

- Telegramme mit der VLAN ID "0" werden wie ungetaggte Telegramme behandelt, behalten jedoch ihren Prioritäswert.
- Alle Ports am Gerä senden standardmäßg Telegramme ohne VLAN-Tag, um sicher zu gehen, dass der Endteilnehmer diese Telegramme empfangen kann.
- Die werkseitige Zuordnung der Ports finden Sie im Kapitel "VLAN (Seite 2696)".
- Wenn an einem Port ein Endteilnehmer angebunden ist, dann sollen ausgehende Telegramme ohne Tag versendet werden (statischer Zugriffs-Port). Wenn sich an dem Port ein weiterer Switch befindet, so ist das Telegramm mit einem Tag zu versehen (Trunk Port).

Port-basiertes VLAN

Verarbeitung empfangener Telegramme

Auf dieser Seite legen Sie die Konfiguration der Port-Eigenschaften für den Telegrammempfang fest.

Beschreibung der angezeigten Felder

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**
  Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Priorität / Port-VID / Erlaubte Telegrammtypen / Ingress Filterung**
  Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports und Link Aggregationen an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Priorität**
  Wählen Sie die Priorität aus, mit der ungetaggte Telegramme versehen werden.
  Die CoS-Priorität (Class of Service), die im VLAN-Tag verwendet wird. Wird ein Telegramm ohne Tag empfangen, wird ihm diese Priorität zugeordnet. Diese Priorität legt fest, wie dieses Telegramm im Vergleich zu anderen Telegrammen weiterhin bearbeitet wird. Es gibt insgesamt acht Prioritäten, mit den Werten 0 bis 7, wobei 7 der höchsten Priorität entspricht (IEEE 802.1p Port Priority).
- **Port-VID**
  Wählen Sie die VLAN-ID aus. Nur die VLAN-IDs sind wählbar, die Sie auf der Seite "VLAN > Allgemein" definiert haben.
  Wenn ein empfangenes Telegramm kein VLAN-Tag hat, so wird es um ein Tag mit der hier angegebenen VLAN-ID ergänzt und entsprechend den Regeln am Port gesendet.
- **Erlaubte Telegrammtypen**
  Legen Sie fest, welche Arten von Telegrammen akzeptiert werden. Es gibt folgende Alternativen:
  - Nur getraggte Frames
    Das Gerät verwirft alle ungetaggten Telegramme. Telegramme, die mit "0" getagged sind, werden wie ungetaggte Telegramme behandelt. Das Gerät leitet alle getaggten Telegramme weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration.
  - Nur ungetaggte und mit Priorität getaggte Frames
    Das Gerät verwirft alle getaggten Telegramme. Das Gerät leitet alle ungetaggten Telegramme sowie Telegramme mit einer Priorität (Priority Tagged Frames) weiter. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration. Wenn Sie den Bridge-Modus "Provider" konfiguriert haben, bedeutet diese Einstellung für das Gerät, dass es alle eingehenden Telegramme als ungetaggte Telegramme behandelt.
  - Alle
    Das Gerät leitet alle Telegramme weiter
- **Ingress Filterung**
  Legen Sie fest, ob die VID von empfangenen Telegrammen ausgewertet wird. Folgende Möglichkeiten gibt es:
  - Aktiviert
    Die VLAN-ID empfangener Telegramme bestimmt die Weiterleitung: Für die Weiterleitung eines VLAN-getaggten Telegramms muss der Empfangsport Mitglied im gleichen VLAN sein. Am Empfangsport werden Telegramme aus unbekannten VLANs verworfen.
  - Deaktiviert
    Alle Telegramme werden weitergeleitet.

Dynamic MAC Aging

Das Gerät lernt automatisch die Quelladressen der angeschlossenen Teilnehmer.

Diese Information wird dazu benutzt, um Datentelegramme gezielt an die betroffenen Teilnehmer weiterzuleiten. Die Netzlast für die anderen Teilnehmer wird reduziert.
Erhält ein Gerät innerhalb einer bestimmten Zeitspanne kein Telegramm, dessen Quelladresse mit einer gelernten Adresse übereinstimmt, dann wird die gelernte Adresse gelöscht. Dieser Mechanismus wird als "Aging" bezeichnet. Durch Aging wird verhindert, dass Telegramme fehlgeleitet werden, wenn z.B. ein Endgerät (beispielsweise ein Programmiergerät) an einen anderen Port angeschlossen wird. Wenn die Option nicht aktiviert ist, löscht ein Gerät gelernte Adressen nicht automatisch.

Beschreibung

- **Dynamic MAC Aging**
  Aktivieren oder deaktivieren Sie die Funktion zum automatischen Aging von gelernten MAC-Adressen.
- **Aging Time[s]**
  Tragen Sie die Zeitspanne in Sekunden in 15er-Schritten ein. Nach dieser Zeitspanne wird eine gelernte Adresse gelöscht, wenn das Gerät keine weiteren Telegramme von dieser Absenderadresse mehr empfängt.

Hinweis

Rundungen der Werte, Abweichung vom Sollwert

Beachten Sie bei der Eingabe der Aging Time, dass auf korrekte Werte gerundet wird. Wenn Sie einen Wert eingeben, der nicht durch 15 teilbar ist, wird der Wert automatisch abgerundet.

Spanning Tree

Allgemein

Dies ist die Basisseite zu Spanning Tree. Standardmäßig ist Rapid Spanning Tree aktiviert.

Beschreibung

Die Seite enthält folgende Felder:

- **Spanning Tree**
  Aktivieren oder deaktivieren Sie Spanning Tree.
- **Protokollkompatibilität**
  Folgende Einstellung gibt es:
  - RSTP

ST Allgemein

Die Seite besteht aus folgenden Teilen:

- Der linke Teil der Seite zeigt die Konfiguration des Geräts.

Beschreibung

Die Seite enthält folgende Felder:

- **Bridge-Priorität / Root-Priorität** (Nur Online verfügbar)
  Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root Bridge wird. Die Bridge mit der höchsten Priorität (d. h. dem kleinsten Wert für diesen Parameter) wird Root Bridge. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, dann wird das Gerät Root Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096 mit einem Wertebereich von 0 bis 32768
- **Bridge-Adresse** (Nur Online verfügbar)**/ Root-Adresse** (Nur Online verfügbar)
  Die Bridge-Adresse zeigt die MAC-Adresse des Geräts und die Root-Adresse zeigt die MAC-Adresse der Root-Bridge an.
- **Root-Port** (Nur Online verfügbar)
  Zeigt den Port an, über den der Switch mit der Root-Bridge kommuniziert.
- **Root-Kosten** (Nur Online verfügbar)
  Die Pfadkosten von diesem Gerät bis zur Root-Bridge
- **Topologieänderungen** (Nur Online verfügbar) **/ Letzte Topologieänderung** (Nur Online verfügbar)
  Die Angabe für das Gerät nennt die Zahl der Umkonfigurationen aufgrund des Spanning Tree-Mechanismus seit des letzten Hochlaufs. Für die Root-Bridge wird die Zeitdauer seit der letzten Umkonfiguration wie folgt angezeigt:
  - Sekunden: Zusatz "Sek." hinter der Zahlenangabe
  - Minuten: Zusatz "Min." hinter der Zahlenangabe
  - Stunden: Zusatz "Std." hinter der Zahlenangabe
- **Bridge Hello Time[s] / Root Hello Time[s]** (Nur Online verfügbar)
  Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei Konfigurationstelegrammen ist die "Hello Time".
- **Bridge Forward Delay[s] / Root Forward Delay[s]** (Nur Online verfügbar)
  Neue Konfigurationsinformationen werden von einer Bridge nicht sofort angewendet, sondern erst nach dem im Parameter "Forward Delay" festgelegten Zeitraum. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.
- **Bridge Max Age[s] / Root Max Age[s]** (Nur Online verfügbar)
  Wenn die BPDU älter ist als das angegebene "Max Age", wird sie verworfen.
- **Zähler zurücksetzen** (Nur Online verfügbar)
  Klicken Sie auf diese Schaltfläche, um die Zähler auf dieser Seite zurückzusetzen.

ST-Port

In der Tabelle wird beim Aufruf der Seite der aktuelle Stand der Konfiguration der Port-Parameter angezeigt.

Klicken Sie zur Konfiguration in die entsprechenden Felder der Port-Tabelle.

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Alle Ports**
  Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Spanning Tree-Status**
  Wählen Sie in der Klappliste die Einstellung für alle Ports aus. Wenn "Keine Änderung" ausgewählt ist, bleiben die Einträge der entsprechenden Spalte in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt alle verfügbaren Ports an.
- **Spanning Tree-Status**
  Legen Sie fest, ob der Port im Spanning-Tree integriert ist oder nicht.

  Hinweis

  Wenn Sie die Option "Spanning Tree Status" für einen Port deaktivieren, kann es zur Schleifenbildung kommen. Die Topologie muss beachtet werden.

- **Priorität**
  Tragen Sie die Priorität des Ports ein. Die Priorität wird nur ausgewertet, wenn die Pfadkosten gleich sind.
  Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
  Wertebereich: 0 - 240.
  Der Standardwert ist 128.
- **Kalk. Kosten**
  Tragen Sie die Wegekostenberechnung ein. Wenn Sie den Wert "0" eintragen, wird im Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.

- **Pfadkosten** (Nur Online verfügbar)
  Dieser Parameter dient zur Berechnung des zu wählenden Weges. Die Strecke mit dem geringsten Wert wird als Weg ausgewählt. Haben mehrere Ports eines Geräts den gleichen Wert bei gleichen Pfadkosten, wird der Port mit der niedrigsten Portnummer ausgewählt. Wenn im Feld "Kalk. Kosten" der Wert "0" ist, wird der automatisch ermittelte Wert angezeigt. Wenn ein Wert ungleich "0" eingetragen wird, wird der Wert des Feldes "Kalk. Kosten" angezeigt.
  Die Ermittlung der Wegekosten richtet sich maßgeblich nach der Übertragungsgeschwindigkeit. Je höher die erzielbare Übertragungsgeschwindigkeit ist, umso kleiner ist der Wert für die Wegekosten.
  Typische Werte für Wegekosten bei Rapid Spanning Tree:
  - 10.000 Mbit/s = 2.000
  - 1000 Mbit/s = 20.000
  - 100 Mbit/s = 200.000
  - 10 Mbit/s = 2.000.000

  Die Werte können aber auch individuell parametriert werden.
- **Status** (Nur Online verfügbar)
  Zeigt den momentanen Status an, in dem sich der Port befindet. Die Werte werden nur angezeigt und können nicht parametriert werden. Der Parameter "Status" ist abhängig von dem projektierten Protokoll. Folgende Werte sind möglich:
  - Disabled
    Der Port empfängt nur und nimmt nicht am STP, MSTP und RSTP teil.
  - Discarding
    In der Betriebsart "Discarding" werden BPDU-Telegramme empfangen. Andere aus- oder eingehende Telegramme werden verworfen.
  - Listening
    In diesem Status werden sowohl BPDU-Telegramme empfangen als auch gesendet. Der Port ist in den Spanning Tree-Algorithmus einbezogen.
  - Learning
    Vorstufe zum Status "Forwarding", der Port lernt aktiv die Topologie (d. h. die Teilnehmeradressen).
  - Forwarding
    Der Port ist nach der Umkonfigurationszeit aktiv im Netz, er empfängt und sendet Datentelegramme.
- **Fwd. Trans** (Nur Online verfügbar)
  Gibt die Anzahl der Wechsel vom Status "Discarding" zum Status "Forwarding" an.

- **Edge Type**
  Legen Sie die Art des "Edge Port" fest. Sie haben folgende Möglichkeiten:
  - "-"
    Edge Port ist deaktiviert. Der Port wird wie ein "no Edge Port" behandelt.
  - Admin
    Wählen Sie diese Option, wenn sich an diesem Port immer ein Endgerät befindet. Sonst wird bei jeder Verbindungsänderung eine Rekonfiguration des Netzwerks ausgelöst.
  - Auto
    Wählen Sie diese Option, wenn an diesem Port automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist. Beim ersten Verbindungsaufbau wird der Port wie ein "no Edge Port" behandelt.
  - Admin/Auto
    Wählen Sie diese Optionen, wenn Sie an diesem Port eine Kombination aus beidem betreiben. Beim ersten Verbindungsaufbau wird der Port als "Edge Port" behandelt.
- **Edge** (Nur Online verfügbar)
  Zeigt an, in welchem Status der Port ist.
  - Aktiviert
    An diesem Port befindet sich ein Endgerät.
  - Deaktiviert
    An diesem Port befindet sich ein Spanning Tree- oder Rapid Spanning Tree-Gerät.

  Bei einem Endgerät kann ein Switch ohne Rücksicht auf Spanning Tree-Telegramme schneller den Port umschalten. Wird entgegen dieser Einstellung ein Spanning Tree-Telegramm empfangen, wechselt der Port automatisch auf die Einstellung "Deaktiviert".
- **P.t.P.-Typ**
  Wählen Sie in der Klappliste die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.
  - "-"
    Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
  - P.t.P.
    Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
  - Shared Media
    Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

  Hinweis

  Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

- **P.t.P.** (Nur Online verfügbar)
  - Aktiviert
    Zeigt an, dass eine Punkt zu Punkt Verbindung besteht.
  - Deaktiviert
    Zeigt an, dass keine Punkt zu Punkt Verbindung besteht

LLDP

Bestimmung der Netzwerktopologie

LLDP (Link Layer Discovery Protocol) ist im Standard IEEE 802.AB definiert.

LLDP ist ein Verfahren zur Bestimmung der Netzwerktopologie. Netzwerkkomponenten tauschen über LLDP Informationen mit ihren Nachbargeräten aus.

Netzwerkkomponenten, die LLDP unterstützen, verfügen über einen LLDP-Agenten. Der LLDP-Agent versendet in periodischen Abständen Informationen über sich selbst und empfängt Informationen von angeschlossenen Geräten. Die empfangenen Informationen werden

Anwendungen

PROFINET benutzt das LLDP-Protokoll für die Topologie-Diagnose. In der Werkseinstellung ist LLDP für alle Ports aktiviert, d. h. es werden LLDP-Telegramme auf allen Ports gesendet und empfangen. Mit dieser Funktion haben Sie die Möglichkeit das Aussenden und/oder Empfangen pro Port ein- oder auszuschalten.

Beschreibung

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**
  Zeigt an, dass die Einstellung für alle Ports der Tabelle 2 gültig ist.
- **Einstellung**
  Wählen Sie die gewünschte Einstellung. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**
  Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**
  Zeigt die verfügbaren Ports an. Der Port setzt sich aus der Modulnummer und der Portnummer zusammen, z. B. Port 0.1 ist Modul 0, Port 1.
- **Einstellung**
  Legen Sie fest, ob der Port LLDP-Telegramme senden oder empfangen soll. Folgende Möglichkeiten gibt es:
  - Rx
    Dieser Port kann LLDP-Telegramme nur empfangen.
  - Tx
    Dieser Port kann LLDP-Telegramme nur senden.
  - Rx & Tx
    Dieser Port kann LLDP-Telegramme empfangen und senden.
  - "-" (Deaktiviert)
    Dieser Port kann LLDP-Telegramme weder empfangen noch senden.

Layer 3-Funktionen für IPv4 projektieren

Statische Routen

Statische Route

Auf dieser Seite legen Sie fest, über welche Routen ein Datenaustausch zwischen den verschiedenen Subnetzen stattfinden kann. Dynamische Routingprotokolle werden nicht unterstützt, z. B. RIP, OSPF.

Beschreibung

Die Seite enthält folgende Felder:

- Zielnetzwerk
  Tragen Sie die Netzwerkadresse des Ziels ein, das über diese Route erreichbar ist.
- Subnetzmaske
  Tragen Sie die dazugehörende Subnetzmaske ein.
- Gateway
  Tragen Sie die IPv4-Adresse des Gateways ein, über den diese Netzwerkadresse erreichbar ist.
- Schnittstelle
  Wählen Sie die Schnittstelle der Route aus.
- Administrative Distanz
  Tragen Sie die administrative Distanz für die Route ein. Die administrative Distanz entspricht der Güte einer Verbindung, z. B. Geschwindigkeit, Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Wert benutzt.
  Wenn Sie nicht eintragen, wird automatisch "nicht verwendet" eingetragen. Die administrative Distanz ist nachträglich änderbar.
  Wertebereich: 1 - 254 oder -1 für "nicht verwendet". Dabei ist 1 der Wert für die bestmögliche Route.

Die Tabelle gliedert sich in folgende Spalten:

- Zielnetzwerk
  Zeigt die Netzwerkadresse des Ziels an.
- Subnetzmaske
  Zeigt die dazugehörende Subnetzmaske an.
- Gateway
  Zeigt die IPv4-Adresse des nächsten Gateways an.
- Schnittstelle (Nur Online verfügbar)
  Zeigt die Schnittstelle der Route an.
- Administrative Distanz
  Tragen Sie die administrative Distanz für die Route ein.
- Status (Nur Online verfügbar)
  Zeigt an, ob die Route aktiv ist oder nicht.

Subnetze

Übersicht

Die Seite zeigt die Subnetze für die ausgewählte Schnittstelle. Ein Subnetz bezieht sich immer auf eine Schnittstelle und wird auf dem Register "Konfiguration" angelegt.

Beschreibung

Die Seite enthält folgendes Feld:

- **Schnittstelle**
  Wählen Sie die gewünschte Schnittstelle aus, an die Sie ein weiteres Subnetz projektieren.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die Schnittstelle an.
- **TIA-Schnittstelle**
  Zeigt die ausgewählte TIA-Schnittstelle an. Über die TIA-Schnittstelle wird definiert, auf welchem VLAN die PROFINET Funktionalitäten zur Verfügung stehen. Dies betrifft hauptsächlich die Gerätesuche mit oder über DCP-
- **Schnittstellenname**
  Zeigt den Namen der Schnittstelle.
- **MAC-Adresse**
  Zeigt die MAC-Adresse an.
- **IP-Adresse**
  Zeigt die IPv4-Adresse des Subnetzes an.
- **Subnetzmaske**
  Zeigt die Subnetzmaske.
- **Adresstyp**
  Zeigt den Adressentyp an. Folgende Werte sind möglich:
  - Primär
    Die erste IPv4-Adresse, die an der Schnittstelle konfiguriert wurde.
  - Sekundär
    Alle weiteren IPv4-Adressen, die an der Schnittstelle konfiguriert wurden.
- **Methode der IP-Adresszuweisung**
  Zeigt an, wie die IPv4-Adresse zugeordnet wird. Folgende Werte sind möglich:
  - Statisch
    Die IPv4-Adresse ist statisch. Tragen Sie die Einstellungen bei "IP-Adresse" und "Subnetzmaske" ein.
  - Dynamisch (DHCP)
    Das Gerät bezieht eine dynamische IPv4-Adresse von einem DHCPv4-Server.

- **Status der Erkennung von Adresskollisionen** (Nur Online verfügbar)
  Wenn neue IPv4-Adressen im Netz aktiv werden, prüft die Funktion "Erkennung von Adresskollisionen ", ob es zu Adresskollisionen kommen kann. Dadurch werden IPv4-Adressen erkannt, die doppelt vergeben werden sollen.

  Hinweis

  Die Funktion führt keine zyklische Prüfung durch.

  Diese Spalte zeigt an, in welchem Status sich die Funktion befindet. Folgende Werte sind möglich:

  – Idle
    Die Schnittstelle ist nicht aktiv und besitzt keine IPv4-Adresse.
  – Starting
    Dieser Status bezeichnet die Anlaufphase. In dieser Phase sendet das Gerät zunächst eine Anfrage, ob es die geplante IPv4-Adresse bereits gibt. Wenn die Adresse noch nicht vergeben ist, sendet das Gerät die Mitteilung, dass es ab jetzt diese IP-Adresse verwendet.
  – Conflict
    Die Schnittstelle ist nicht aktiv. Die Schnittstelle versucht eine IPv4-Adresse zu verwenden, die bereits vergeben ist.
  – Defending
    Die Schnittstelle verwendet eine eindeutige IPv4-Adresse. Eine andere Schnittstelle versucht die gleiche IPv4-Adresse zu verwenden.
  – Active
    Die Schnittstelle verwendet eine eindeutige IPv4-Adresse. Es gibt keine Kollisionen.
  – Not supported
    Die Funktion zur Erkennung von Adresskollisionen wird nicht unterstützt.
  – Disabled
    Die Funktion zur Erkennung von Adresskollisionen ist deaktiviert.

- **MTU**
  Zeigt die Paketgröße an.

Konfiguration

Auf dieser Seite konfigurieren Sie das Subnetz für die Schnittstelle.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle (Name)**
  Wählen Sie die gwünschte Schnittstelle aus.
- **Schnittstellenname**
  Tragen Sie den Namen für die Schnittstelle ein.
- **MAC-Adresse** (Nur Online verfügbar)
  Zeigt die MAC-Adresse der ausgewählten Schnittstelle an.

- DHCP
  Aktivieren oder deaktivieren Sie den DHCP-Client für dieses IPv4-Schnittstelle.
- IP-Adresse
  Tragen Sie die IPv4-Adresse der Schnittstelle ein. Die IPv4-Adressen dürfen nicht mehrfach verwendet werden.
- Subnetzmaske
  Tragen Sie die Subnetzmaske des zu erstellenden Subnetzes ein. Subnetze an unterschiedlichen Schnittstellen dürfen sich nicht überlappen.
- Broadcast-IP-Adresse
  Wenn eine bestimmte IP-Adresse als Broadcast-IP-Adresse des Subnetzes verwendet werden soll, dann tragen Sie diese ein. Ansonsten wird die letzte IP-Adresse des Subnetzes verwendet.
- Adresstyp
  Zeigt den Adressen Typ an. Folgende Werte sind möglich:
  - Primär
    Das erste Subnetz der Schnittstelle.
  - Sekundär
    Alle weiteren Subnetze der Schnittstelle.
- TIA-Schnittstelle
  Aktivieren oder deaktivieren Sie die Einstellung.
  Für die TIA-Schnittstelle gelten folgende Bedingungen:
  - Nur Schnittstellen mit dem Adresstyp "Primär" können als TIA-Schnittstelle aktiviert werden.
  - Es muss immer eine TIA-Schnittstelle geben.
  - Es kann nur eine TIA-Schnittstelle geben.
  - Eine TIA-Schnittstelle ist immer eine VLAN-Schnittstelle.
- MTU
  Mit MTU (Maximum Transmission Unit) wird die maximale Größe des Pakets festgelegt. Wenn die Pakete größer sind, als die eingestellte MTU, werden Sie fragmentiert.

NAT

Masquerading

Auf dieser Seite aktivieren Sie die Regeln für IP-Masquerading (Seite 2709).

Beschreibung

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht. Nur Schnittstellen mit konfiguriertem Subnetz sind verfügbar.
- **Masquerading aktivieren**
  Wenn aktiviert, wird bei jedem ausgehenden Datenpaket, das über diese Schnittstelle gesendet wird, die Quell-IP-Adresse durch die IP-Adresse der Schnittstelle ersetzt

NAPT

Auf dieser Seite konfigurieren Sie die Portweiterleitung (Seite 2709).

Beschreibung

Die Seite enthält folgende Felder:

- **Quell-Schnittstelle**
  Wählen Sie die Schnittstelle aus, auf der die Anfragen eintreffen sollen.
- **Traffic-Typ**
  Legen Sie fest, für welches Protokoll die Adresszuordnung gültig ist.
- **Schnittstellen-IP der Quell-Schnittstelle verwenden**
  Wenn aktiviert, wird bei "Ziel-IP-Adresse" die IP-Adresse der ausgewählten Schnittstelle verwendet.
- **Ziel-IP-Adresse**
  Geben Sie die Ziel-IP-Adresse ein. An dieser IP-Adresse werden die Telegramme empfangen. Nur editierbar, wenn "Schnittstellen-IP der Quell-Schnittstelle verwenden" deaktiviert ist.
- **Ziel-Port**
  Geben Sie den Ziel-Port ein. Eingehende Telegramme mit diesem Port als Ziel-Port werden weitergeleitet. Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.

- **Ziel-IP-Umsetzung**
  Geben Sie die IP-Adresse des Teilnehmers an, an den dieses Telegramm weitergeleitet wird.
- **Ziel-Port Umsetzung**
  Geben Sie die Nummer des Ports ein. Das ist der neue Ziel-Port, an den das eingehende Telegramm weitergeleitet wird. Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  Wenn der "Ziel-Port" und der "Ziel-Port Umsetzung" gleich sind, werden die Telegramme ohne Port-Umsetzung weitergeleitet.

  Hinweis

  Wenn der Port bereits durch einen lokalen Dienst z. B. Telnet belegt ist, wird eine Warnmeldung ausgegeben.
  Vermeiden Sie auf jeden Fall die Nutzung von folgenden Ports:
  - Telnet / SSH: TCP-Port 23
  - HTTP: TCP-Port 80
  - HTTPS: TCP-Port 443
  - SNMP: UDP-Port 161
  - ISAKMP: UDP-Port 500
  - IPsec Nat-T: UDP-Port 4500
  - NTP: UDP-Port 123

Die Tabelle gliedert sich in folgende Spalten:

- **Quell-Schnittstelle**
  Zeigt die Schnittstelle an, von dem die Pakete kommen müssen. Nur diese Pakete kommen für eine Portweiterleitung in Frage.
- **Traffic-Typ**
  Zeigt an, für welches Protokoll die Adresszuordnung gilt.
- **Schnittstellen-IP**
  Zeigt an, ob die IP-Adresse der Schnittstelle verwendet wird.
- **Ziel-IP**
  Zeigt die Ziel-IP-Adresse an. An dieser IP-Adresse werden die Telegramme empfangen.
- **Ziel-Port**
  Zeigt den Ziel-Port an. Eingehende Telegramme mit diesem Port als Ziel-Port werden weitergeleitet.
- **Ziel-IP-Umsetzung**
  Zeigt die IP-Adresse Teilnehmers an, an dem die Pakete weitergeleitet werden.
- **Ziel-Port Umsetzung**
  Zeigt an, auf welchen Ziel-Port übersetzt wird.

Source-NAT

Auf dieser Seite konfigurieren Sie die Regeln für Source-NAT (Seite 2709).

Hinweis

Firewallregeln bei Source-NAT

Wenn Sie für eine Source-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Quell-IP-Subnetz". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz".

Beschreibung

- **Quell-Schnittstelle / Ziel-Schnittstelle**
  Legen Sie die Richtung des Verbindungsaufbaus fest. Es werden nur Verbindungen berücksichtigt, die in dieser festgelegten Richtung aufgebaut werden.
  Zur Auswahl stehen auch die virtuellen Schnittstellen von VPN-Verbindungen:
  - VLANx: VLANs mit konfigurierten Subnetz
  - ppp0 bzw, usb0 (nur beim M876-4): WAN-Schnittstelle
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung

  Hinweis

  Wenn Sie eine NAT-Adressumsetzung in oder aus Richtung VPN-Tunnel konfigurieren, sind nur noch die beteiligten IP-Adressen der NAT-Adressumsetzungsregeln über VPN-Tunnel erreichbar.

- **Quell-IP-Adressen**
  Legen Sie fest, für welche Quell-IP-Adressen diese Source-NAT-Regel gültig ist. Nur die Pakete werden berücksichtigt, die der eingegebenen Adressen entsprechen.
  Folgende Eingaben sind möglich:
  - IP-Adresse: Gilt genau für die angegebene IP-Adresse.
  - IP-Adressbereich: Gilt für einen bestimmten IP-Adressbereich: Start-IP-Adresse "-" End-IP-Adresse an, z. B. 192.168.100.10 - 192.168.100.20
  - IP-Subnetz: Gilt für mehrere IPv4-Adressen, die zu einem IP-Adressbereich zusammengefasst werden : IP-Adresse/Anzahl Bits des Netzanteils (CIDR-Notation)
- **Schnittstellen-IP der Ziel-Schnittstelle verwenden**
  Wenn aktiviert, wird bei "Quell-IP-Umsetzung" die IP-Adresse der ausgewählten Ziel-Schnittstelle verwendet.

- **Quell-IP-Umsetzung**
  Geben Sie die IP-Adresse ein, mit der die IP-Adresse des Absenders ersetzt wird. Nur editierbar, wenn "Schnittstellen-IP der Ziel-Schnittstelle verwenden" deaktiviert ist.
- **Ziel-IP-Adressen**
  Legen Sie fest, für welche Ziel-IP-Adressen diese Source-NAT-Regel gültig ist. Nur die Pakete werden berücksichtigt, deren Ziel-IP-Adresse im Bereich der eingegebenen Adressen liegt.
  - IP-Adresse: Gilt genau für die angegebene IP-Adresse.
  - IP-Adressbereich: Gilt für einen bestimmten IP-Adressbereich: Start-IP-Adresse "-" End-IP-Adresse an, z. B. 192.168.100.10 - 192.168.100.20
  - IP-Subnetz: Gilt für mehrere IPv4-Adressen, die zu einem IP-Adressbereich zusammengefasst werden: IP-Adresse/Anzahl Bits des Netzanteils (CIDR-Notation)

Die Tabelle gliedert sich in folgende Spalten:

- Quell-Schnittstelle
  Zeigt die Quell-Schnittstelle an.
- Ziel-Schnittstelle
  Zeigt die Ziel-Schnittstelle an.
- Quell-IP-Adressen
  Zeigt die IP-Adressen der Absender an, für die eine Adressumsetzung gewünscht ist.
- Quell-IP-Umsetzung
  Zeigt die IP-Adresse an, mit der die IP-Adresse der Absender ersetzt wird.
- Ziel-IP-Adressen
  Zeigt die IP-Adressen der Empfänger an, für die eine Adressumsetzung gewünscht ist.

NETMAP

Auf dieser Seite legen Sie die Regeln für NETMAP (Seite 2709) fest. NETMAP ist ein statisches 1:1-Mapping von Netzwerkadressen, wobei der Hostanteil erhalten bleibt. Weiter Informationen dazu finden Sie im Kapitel "NAT und Firewall (Seite 2711)".

Hinweis

Firewallregeln bei Source-NAT

Wenn Sie für eine Source-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Quell-IP-Subnetz". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz".

Firewallregeln bei Destination-NAT

Wenn Sie für eine Destination-NAT-Regel eine entsprechende Firewall-Regel anlegen, verwenden Sie bei "IP-Regeln" für die "Quelle (Bereich)" die Eingabe aus "Ziel-IP-Subnetz Umsetzung". Und für die "Ziel (Bereich)" verwenden Sie die Eingabe aus "Ziel-IP-Subnetz".

Beschreibung

- Typ
  Legen Sie die Art der Adressumsetzung fest.
  - Quelle: Ersetzen der Quell-IP-Adresse
  - Ziel: Ersetzen der Ziel-IP-Adresse.
- Quell-Schnittstelle
  Legen Sie die Quell-Schnittstelle fest.
  - VLANx: VLANs mit konfigurierten Subnetz
  - ppp0 bzw, usb0 (nur beim M876-4): WAN-Schnittstelle
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- Ziel-Schnittstelle
  Legen Sie die Ziel-Schnittstelle fest.
  - VLANx: VLANs mit konfigurierten Subnetz
  - ppp0 bzw, usb0 (nur beim M876-4): WAN-Schnittstelle
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsec VPN-Verbindung
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- Quell-IP-Subnetz
  Tragen Sie das Subnetz des Absenders ein.
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- Quell-IP-Subnetz Umsetzung
  Tragen Sie Subnetz ein, mit der das Subnetz des Absenders ersetzt wird. Nur editierbar, bei den Einstellungen "SourceNAT".
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- Ziel-IP-Subnetz
  Tragen Sie das Subnetz des Empfängers ein.
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.
- Ziel-IP-Subnetz Umsetzung
  Tragen Sie Subnetz ein, mit der das Subnetz des Empfängers ersetzt wird. Nur editierbar, bei den Einstellungen "DestinationNAT".
  Das Subnetz kann auch nur ein einzelner PC, oder eine andere Untermenge des Subnetzes sein. Verwenden Sie die CIDR-Schreibweise.

Die Tabelle gliedert sich in folgende Spalten:

- Typ
  Zeigt die Richtung der Adressumsetzung an.
- Quell-Schnittstelle
  Zeigt die Quell-Schnittstelle an.
- Ziel-Schnittstelle
  Zeigt die Ziel-Schnittstelle an.
- Quell-IP-Subnetz
  Zeigt das Subnetz des Absenders an.
- Quell-IP-Subnetz Umsetzung
  Zeigt das Subnetz des Absenders an, mit der das Subnetz des Absenders ersetzt wird.
- Ziel-IP-Subnetz
  Zeigt das Subnetz des Empfängers an.
- Ziel-IP-Subnetz Umsetzung
  Zeigt das Subnetz des Empfängers an, mit der das Subnetz des Empfängers ersetzt wird.

VRRPv3

Router

Einleitung

Auf dieser Seite legen Sie neue virtuelle Router an. Weitere Parameter konfigurieren Sie unter "Layer 3 (IPv4) > VRRPv3 > Konfiguration".

Hinweis

- Sie können VRRPv3 an VLAN-Schnittstellen nutzen. Router-Ports werden nicht unterstützt.

Voraussetzung

Damit die eingehenden VRRP-Pakete an das Gerät weitergeleitet werden, ist folgende Firewallregel zu konfigurieren:

Security > Firewall > IP-Protokoll:

- Protokollname: "VRRP“
- Protokollnummer: 112

Security > Firewall > IP-Regeln:

- Protokoll: IPv4
- Aktion: Accept
- Von: <Schnittstelle>
- Nach: Gerät

- Quelle (Bereich): 0.0.0.0/0
- Ziel (Bereich): 224.0.0.18/32
- Dienste: VRRP

Beschreibung

- **VRRPv3**
  Aktivieren oder deaktivieren Sie das Routing über VRRPv3.
- **Antworte an virtuellen Schnittstellen auf Ping-Anfragen**
  Wenn aktiviert, antworten auch die virtuellen IPv4-Adressen auf den Ping.
- **VRID-Tracking**
  Aktivieren oder deaktivieren Sie das VRID-Tracking. Wenn aktiviert, werden alle VRRP-Instanzen überwacht.
  Wenn sich der Status einer VRRP-Instanz nach "Initialize" ändert, wird die Priorität aller VRRP-Instanzen auf den Wert "1" verringert.
  Wenn sich der Status der VRRP-Instanz ändert, wird die ursprüngliche Priorität aller VRRP-Instanzen wiederhergestellt.
- **Schnittstelle**
  Wählen Sie die gewünschte VLAN-Schnittstelle aus, die als virtueller Router fungiert.
- **VRID**
  Geben Sie die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**
  Zeigt die IPv4-Schnittstelle an, die als virtueller Router fungiert.
- **VRID**
  Zeigt die ID des virtuellen Routers an.
- **Virtuelle MAC-Adresse** (Nur Online verfügbar)
  Zeigt die virtuelle MAC-Adresse des virtuellen Routers an.
- **Primäre IP-Adresse**
  Zeigt die kleinste numerische IPv4-Adresse auf diesem VLAN an. Der Eintrag 0.0.0.0 bedeutet, dass die "Primäre" Adresse auf diesem VLAN verwendet wird. Ansonsten sind alle IPv4-Adressen, die auf diesem VLAN in dem Menü "Layer 3 > Subnetze" konfiguriert wurden, gültige Werte.

- **Router-Status** (Nur Online verfügbar)
  Zeigt den derzeitigen Zustand des virtuellen Routers an. Mögliche Werte sind:
  - Master
    Der Router ist der Master-Router und übernimmt die Routingfunktionalität für alle zugeordneten IPv4-Adressen.
  - Backup
    Der Router ist der Backup-Router. Wenn der Master-Router ausfällt, übernimmt der Backup-Router die Aufgaben des Master-Routers.
  - Initialize
    Der virtuelle Router wurde soeben eingeschaltet. In kurzer Zeit wird er in den Zustand "Master" oder "Backup" wechseln.
- **Master IP-Adresse** (Nur Online verfügbar)
  Zeigt die IPv4-Adresse des Master-Routers an.
- **Priorität** (Nur Online verfügbar)
  Zeigt die Priorität des virtuellen Routers an.
  Wenn eine IPv4-Adresse dem VRRP-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- **Advert.-Intervall**
  Zeigt das Intervall an, in dem der Master-Router VRRPv3-Pakete verschickt.
- **Unterbrechen**
  Zeigt die Bevorzugung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - ja
    Dieser Router ist beim Rollenwechsel bevorzugt.
  - nein
    Dieser Router ist beim Rollenwechsel nicht bevorzugt.

VRRP und DHCP-Server

Wenn Sie auf den Geräten einer VRRP-Gruppe einen DHCP-Server betreiben wollen, muss der DHCP-Server auf dem Master-Router konfiguriert sein. Backup-Router reagieren nicht auf DHCP-Anfragen. Stellen Sie sicher, dass der Master-Router statisch konfiguriert ist und nach einem Ausfall wieder zum Master der VRRP-Gruppe wird.

Konfiguration

Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

Beschreibung

Die Seite enthält Folgendes:

- **Schnittstelle / VRID**
  Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Primäre Adresse**
  Wählen Sie die vorrangige IPv4-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IPv4-Adresse.

  Hinweis

  Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist dann 0.0.0.0.
  Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IPv4-Adresse als Quelladresse für VRRP-Pakete genutzt wird, wählen Sie die entsprechende IPv4-Adresse aus. Ansonsten wird die numerisch kleinste IPv4-Adresse verwendet.

- **Master**
  Wenn aktiviert, wird die numerisch kleinste IPv4-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die numerisch kleinste IPv4-Adresse des VRRPv3-Routers als virtuelle IP-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IPv4-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**
  Gegen Sie die Priorität dieses virtuellen Routers ein. Gültige Werte sind 1-254.
  Wenn eine IPv4-Adresse dem VRRPv3-Router zugeordnet wird, die auch tatsächlich an der lokalen IPv4-Schnittstelle konfiguriert ist, wird automatisch der Wert 255 eingetragen. Alle anderen Prioritäten können frei auf die VRRPv3-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRPv3-Router zum "Master".
- **Advertisement-Intervall**
  Geben Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRPv3-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**
  Erlauben sie die Bevorzugung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.
- **VRRP-Kompatibilitätsmodus**
  Wenn aktiviert, sendet und empfängt der VRRPv3-Router für konfigurierte IPv4-Adressen zusätzlich zu den VRRPv3-Telegrammen auch VRRPv2-Telegramme. Nur notwendig, wenn nicht alle VRRP-Router VRRPv3 unterstützen
- **Track-ID**
  Wählen Sie eine Track-ID aus.
- **Priorität verringern**
  Geben Sie ein, um welchen Wert die Priorität der VRRP-Schnittstelle verringert wird.
- **Aktuelle Priorität**
  Zeigt die Priorität der VRRP-Schnittstelle an, nachdem die überwachte Schnittstelle in den Zustand "down" gewechselt hat.

Adressübersicht

Übersicht

Diese Seite zeigt an, welche IPv4-Adressen der virtuelle Router überwacht. Jeder virtueller Router kann maximal 10 IPv4-Adressen überwachen.

Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- Schnittstelle
  Zeigt die Schnittstelle an, der als virtueller Router fungiert.
- VRID
  Zeigt die ID dieses virtuellen Routers an.
- Anzahl der Adressen
  Zeigt die Anzahl der IPv4-Adressen an.
- Zugeordnete IP-Adresse (1) ...Zugeordnete IP-Adresse (10)
  Zeigt die Router IPv4-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IPv4-Adressen durch diesen Router übernommen.

Adresskonfiguration

Anlegen oder ändern der überwachten IP-Adressen

Auf dieser Seite können Sie die zu überwachenden IPv4-Adressen anlegen, ändern oder löschen.

Einstellungen

Die Seite enthält Folgendes:

- Schnittstelle / VRID
  Wählen Sie die ID des virtuellen Routers aus.
- Zugeordnete IP-Adresse
  Geben Sie die IPv4-Adresse ein, die der virtuelle Router überwachen soll.

Die Tabelle gliedert sich in folgende Spalten:

- Zugeordnete IP-Adresse
  Zeigt die IPv4-Adressen an, die der virtuelle Router überwacht.

Schnittstellenüberwachung

Einleitung

Auf dieser Seite konfigurieren Sie die Überwachung von Schnittstellen.

Wenn sich der Link einer überwachten Schnittstelle von "up" nach "down" ändert, wird die Priorität der zugeordneten VRRP-Schnittstelle verringert. Um welchen Wert die Priorität verringt wird, konfigurieren Sie auf der Seite "Layer 3 > VRRP/VRRPv3 > Konfiguration".

Wenn sich der Link der Schnittstelle wieder von "down" nach "up" ändert, wird die unsprüngliche Priorität der VRRP-Schnittstelle wiederhergestellt.

Beschreibung

Die Seite enthält folgende Felder:

- Schnittstelle
  Wählen Sie die gewünschte Schnittstelle aus, die überwacht werden soll.
- Track-ID
  Geben Sie eine Track-ID ein.
- Track-ID
  Wählen Sie eine Track-ID aus.
- Track-Schnittstellenzähler
  Geben Sie ein, wie viele der überwachten Schnittstellen in den Zustand "down" wechseln müssen, bevor die Priorität geändert wird.

Die Tabelle gliedert sich in folgende Spalten:

- Track-ID
  Zeigt die Track-ID an.
- Schnittstelle
  Zeigt die Schnittstelle an, die überwacht wird.

Security-Funktionen projektieren

Benutzer

Lokale Benutzer

Hinweis

Die Seite ist bei einigen Geräten nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Lokale Benutzer

Auf dieser Seite erstellen Sie lokale Benutzer für WBM und CLI mit den entsprechenden Rechten.

Wenn Sie einen lokalen Benutzer anlegen oder löschen, wird diese Änderung automatisch auch in der Tabelle "Externe Benutzerkonten" durchgeführt. Wenn Sie eine Änderung explizit für die interne oder externe Benutzertabelle durchführen wollen, nutzen Sie die CLI-Befehle.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden

Beschreibung

Die Seite enthält Folgendes:

- **Benutzerkonto**
  Geben Sie den Namen für den Benutzer ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 250 Zeichen lang sein.
  - Er darf folgende Zeichen nicht enthalten: § ? " ; :

  Hinweis

  Benutzername nicht änderbar

  Nach dem Anlegen eines Benutzers kann der Benutzername nicht mehr geändert werden.

  Wenn ein Benutzername geändert werden soll, dann muss der Benutzer gelöscht und ein neuer Benutzer angelegt werden.

  Hinweis

  Benutzernamen: admin, user, service, debug

  Werkseitig sind bei Auslieferung folgende Benutzernamen vordefiniert: user, admin, service, debug.
  - admin: Mit diesem Benutzernamen, können Sie das Gerät konfigurieren. Wenn Sie sich das erste Mal anmelden oder nach einem "Auf Werkseinstellungen zurücksetzen und Neustart" anmelden, werden Sie aufgefordert das vordefinierte Passwort "admin" zu ändern.
  - user, service, debug: Diese Benutzernamen sind für Servicezwecke vorbehalten.

- **Passwortrichtlinie**
  Zeigt an, welche Passwortrichtlinie verwendet wird:
  - Hoch
    Passwortlänge: mindestens 8 Zeichen, maximal 128 Zeichen
    Mindestens 1 Großbuchstabe
    Mindestens 1 Sonderzeichen
    Mindestens 1 Zahl
  - Niedrig
    Passwortlänge: mindestens 6 Zeichen, maximal 128 Zeichen

- **Passwort bestätigen**
  Geben Sie das Passwort erneut ein, um es zu bestätigen.
- **Rolle**
  Wählen Sie eine Rolle aus.
  Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".

Die Tabelle enthält folgende Spalten:

Hinweis

Die voreingestellten Benutzer sowie angemeldete Benutzer können nicht gelöscht oder geändert werden.

- **Benutzerkonto**
  Zeigt den Benutzernamen an.
- **Rolle**
  Zeigt die Rolle des Benutzers an.
- **Beschreibung**
  Zeigt eine Beschreibung des Benutzerkontos an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Rollen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Rollen

Auf dieser Seite erstellen Sie Rollen, die lokal auf dem Gerät gültig sind.

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Rollenname**
  Geben Sie den Namen für die Rolle ein. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

  Hinweis
  Rollenname nicht änderbar

  Nach dem Anlegen einer Rolle kann der Rollenname nicht mehr geändert werden.

  Wenn ein Rollenname geändert werden soll, dann muss die Rolle gelöscht und eine neue Rolle angelegt werden.

Die Tabelle enthält folgende Spalten:

Hinweis
Rollen löschen

Die voreingestellten Rollen sowie zugewiesene Rollen können nicht gelöscht oder geändert werden

- **Rolle**
  Zeigt den Namen der Rolle an.
- **Funktionsrecht**
  Wählen Sie die Funktionsrechte der Rolle aus:
  - 1
    Benutzer mit dieser Rolle können Geräteparameter lesen aber nicht verändern.
    Benutzer mit dieser Rolle können ihr eigenes Passwort ändern.
  - 15
    Benutzer mit dieser Rolle können Geräteparameter sowohl lesen als auch verändern.

  Hinweis
  Funktionsrecht nicht änderbar

  Wenn Sie eine Rolle zugewiesen haben, können Sie das Funktionsrecht der Rolle nicht mehr ändern.

  Wenn Sie das Funktionsrecht einer Rolle ändern wollen, gehen Sie wie folgt vor:
  1. Löschen Sie alle zugewiesenen Benutzer.
  2. Ändern Sie das Funktionsrecht der Rolle.
  3. Weisen Sie die Rolle erneut zu.

- **Beschreibung**
  Geben Sie eine Beschreibung für die Rolle ein. Bei vordefinierten Rollen wird eine Beschreibung angezeigt. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

Gruppen

Hinweis

Die Seite ist nur verfügbar, wenn eine Online-Verbindung zum Gerät besteht.

Benutzergruppen

Auf dieser Seite verknüpfen Sie eine Gruppe mit einer Rolle.

In diesem Beispiel wird die Gruppe "Administrators" mit der Rolle "admin" verknüpft. Die Gruppe ist auf einem RADIUS-Server definiert. Die Rolle ist lokal auf dem Gerät definiert. Wenn ein RADIUS-Server einen Benutzer authentifiziert und der Gruppe "Administrators" zuordnet, erhält dieser Benutzer auf dem Gerät die Rechte der Rolle "admin".

Hinweis

Es ist von den Rechten des angemeldeten Benutzers abhängig, welche Werte angezeigt werden.

Beschreibung der angezeigten Felder

Die Seite enthält Folgendes:

- **Gruppenname**
  Geben Sie den Namen der Gruppe ein. Der Name muss der Gruppe auf dem RADIUS-Server entsprechen. Der Name muss folgende Bedingungen erfüllen:
  - Er muss eindeutig sein.
  - Er muss zwischen 1 und 64 Zeichen lang sein.

Die Tabelle enthält folgende Spalten:

- **Gruppe**
  Zeigt den Namen der Gruppe an.
- **Rolle**
  Wählen Sie eine Rolle aus. Benutzer, die über den RADIUS-Server mit der verknüpften Gruppe authentifiziert werden, erhalten die Rechte dieser Rolle lokal auf dem Gerät. Sie können zwischen den voreingestellten und selbst definierten Rollen wählen, siehe Seite "Security > Benutzer > Rollen".
- **Beschreibung**
  Geben Sie eine Beschreibung für die die Verknüpfung der Gruppe mit einer Rolle an. Der Beschreibungstext kann bis zu 100 Zeichen lang sein.

AAA

Allgemein

Anmeldung von Netzteilnehmern

Die im Menü verwendete Bezeichnung "AAA" steht dabei für "Authentication, Authorization, Accounting". Dieses Feature dient dazu, Netzteilnehmer zu identifizieren und zuzulassen und ihnen die entsprechenden Dienste bereitzustellen.

Auf dieser Seite konfigurieren Sie die Anmeldung.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

Hinweis

Um den Login-Authentifizierung "RADIUS", "Lokal und RADIUS" oder "RADIUS und Fallback Lokal" nutzen zu können, muss ein RADIUS-Server hinterlegt und für die Benutzerauthentifizierung konfiguriert sein.

- **Login-Authentifizierung**
  Legen Sie fest, wie die Anmeldung erfolgt:
  - Lokal
    Die Authentifizierung muss lokal auf dem Gerät erfolgen.
  - RADIUS
    Die Authentifizierung muss über einen RADIUS-Server erfolgen
  - Local und RADIUS
    Die Authentifizierung kann sowohl über die im Gerät vorhandenen Benutzer (Benutzername und Passwort) als auch über einen RADIUS-Server erfolgen. Es wird zuerst in der lokalen Datenbank nach dem Benutzer gesucht. Wenn der Benutzer dort nicht vorhanden ist, wird eine RADIUS-Anfrage geschickt.
  - RADIUS und Fallback Lokal
    Die Authentifizierung muss über einen RADIUS-Server erfolgen. Nur wenn der RADIUS-Server im Netz nicht erreichbar ist, wird eine lokale Authentifizierung durchgeführt.

RADIUS-Client

Authentifizierung über einen externen Server

Das Konzept von RADIUS basiert auf einem externen Authentifizierungsserver.

Jede Zeile der Tabelle enthält die Zugangsdaten für je einen Server. In der Suchreihenfolge wird der primäre Server zuerst angefragt. Ist der primäre Server nicht erreichbar, werden in der eingetragenen Reihenfolge sekundäre Server angefragt. Wenn keiner der Server antwortet, findet keine Authentifizierung statt.

Beschreibung der angezeigten Felder

Die Seite enthält folgende Felder:

- **RADIUS-Autorisierungsmodus**
  Der RADIUS-Autorisierungsmodus legt bei der Login-Authentifizierung fest, wie bei einer erfolgreichen Authentifizierung die Rechtevergabe für die Benutzer erfolgt.
  - Standard
    In diesem Modus wird der Benutzer mit Administratorrechten angemeldet, wenn der Server für das Attribut "Service Type" den Wert "Administrative User" an das Gerät zurück gibt. In allen anderen Fällen wird der Benutzer mit Leserechten angemeldet.
  - Herstellerspezifisch
    In diesem Modus ist die Rechtevergabe davon abhängig, ob und welche Gruppe der Server für den Benutzer zurück gibt und ob es für den Benutzer einen Eintrag in der Tabelle "Externe Benutzeraccounts" gibt.

Die Tabelle gliedert sich in folgende Spalten:

- **Auth.-Servertyp**
  Wählen Sie aus, für welche Authentifizierungsverfahren der Server verwendet werden soll.
  - Login
    Der Server wird nur für die Login-Authentifizierung verwendet.
  - 802.1X
    Der Server wird nur für die 802.1X-Authentifizierung verwendet.
  - Login & 802.1X
    Der Server wird für beide Authentifizierungsverfahren verwendet.
- **Adresse des RADIUS-Servers**
  Tragen Sie die IPv4-Adresse oder den FQDN (Fully Qualified Domain Name) des RADIUS Servers ein.
- **Server-Port**
  Tragen Sie den Eingangs-Port auf dem RADIUS-Server ein. Standardmäßig ist der Eingangs-Port 1812 eingestellt. Der Wertebereich ist 1 ... 65535.
- **Shared Secret**
  Tragen Sie Ihre Zugangskennung ein. Der Wertebereich ist 1 ... 128 Zeichen.
- **Shared Secret bestätigen**
  Tragen Sie die Zugangskennung zur Bestätigung erneut ein.
- **Max. Retrans**
  Geben Sie hier die maximale Anzahl der Wiederholungen eines Anfrageversuchs ein. Der initiale Verbindungsversuch wird um den hier angegebenen Wert wiederholt, bevor ein anderer konfigurierter RADIUS-Server angefragt wird oder die Anmeldung für gescheitert erklärt wird. Standardmäßig sind 3 Wiederholungen eingestellt, das bedeutet 4 Verbindungsversuche. Der Wertebereich ist 1 ... 5.
- **Primärer Server**
  Legen Sie fest, ob dieser Server der primäre Server ist. Sie können aus den Optionen "ja" oder "nein" auswählen.

- **Test**
  Mit dieser Schaltfläche können Sie testen, ob der angegebene RADIUS-Server verfügbar ist oder nicht. Der Test wird einmalig durchgeführt und nicht zyklisch wiederholt.
- **Testergebnis**
  Zeigt an, ob der RADIUS-Server verfügbar ist oder nicht:
  - Fehlgeschlagen, keine Testpakete gesendet
    Die IP-Adresse ist nicht erreichbar.
    Die IP-Adresse ist erreichbar, der RADIUS-Server läuft jedoch nicht.
  - Erreichbar, das Shared Secret wurde nicht akzeptiert
    Die IP-Adresse ist erreichbar, der RADIUS-Server akzeptiert jedoch das angegebene Shared Secret nicht.
  - Erreichbar, das Shared Secret wurde akzeptiert
    Die IP-Adresse ist erreichbar und der RADIUS-Server akzeptiert das angegebene Shared Secret.

  Das Testergebnis wird nicht automatisch aktualisiert. Um das Testergebnis zu löschen, klicken Sie auf die Schaltfläche "Aktualisieren".

Zertifikate

Übersicht

Auf dieser Seite werden die geladenen Dateien (Zertifikate und Schlüssel) angezeigt. Folgende Möglichkeiten gibt es, um die Dateien ins Gerät zu laden:

- System > Laden & Speichern > HTTP
- System > Laden & Speichern > TFTP
- System > Laden & Speichern > SFTP

Beschreibung

- **Typ**
  Zeigt die Art der geladenen Datei an.
  - CA-Zert
    Das CA-Zertifikat ist von einer zertifizierenden Stelle (CA = Certification Authority) signiert.
  - Gerätezertifikat
  - Schlüsseldatei
  - Remote-Zert
    Gegenstellenzertifikat
- **Dateiname**
  Zeigt den Dateinamen an.
- **Status**
  Zeigt an, ob das Zertifikat gültig oder bereits abgelaufen ist.

- Antragsteller DN
  Zeigt den Namen des Antragsteller an.
- Aussteller DN
  Zeigt den Namen des Zertifikatsausstellers an.
- Ausstellungsdatum
  Zeigt den Beginn des Gültigkeitszeitraums des Zertifikats an
- Ablaufdatum
  Zeigt das Ende des Gültigkeitszeitraums des Zertifikats an.
- Verwendet
  Zeigt an, welche Funktion das Zertifikat nutzt.

Zertifikate

Das Format des Zertifikats basiert auf X.509, einem Standard der ITU-T zum Erstellen digitaler Zertifikate. In diesem Standard ist der schematische Aufbau von X.509-Zertifikaten beschrieben. Weitere Informationen dazu finden Sie im Internet unter "http://www.itu.int".

Auf dieser Seite kann der Inhalt folgender Strukturelemente angezeigt werden. Wenn in dem ausgewählten Zertifikat das Strukturelement nicht vorhanden oder befüllt ist, wird im Feld nichts angezeigt. Bestimmte Einträge sind nur editierbar, wenn Sie unterstützt werden.

Beschreibung

- Dateiname
  Wählen Sie das gewünschte Zertifikat aus.
- Typ
  Zeigt die Art der geladenen Datei an.
  - CA-Zert
    Das CA-Zertifikat ist von einer zertifizierenden Stelle (CA = Certification Authority) signiert.
  - Gerätezertifikat
  - Schlüsseldatei
  - Remote-Zert
    Gegenstellenzertifikat
- Antragsteller DN
  Zeigt den Namen des Antragssteller an.
- Aussteller DN
  Zeigt den Namen des Zertifikatsausstellers an.
- Alternativer Name des Antragsstellers
  Wenn vorhanden, wird ein alternativer Name des Antragsstellers angezeigt.
- Ausstellungsdatum
  Zeigt den Beginn des Gültigkeitszeitraums des Zertifikats an
- Ablaufdatum
  Zeigt das Ende des Gültigkeitszeitraums des Zertifikats an.

- **Seriennummer**
  Zeigt die Seriennummer des Zertifikats an.
- **Verwendet**
  Zeigt an, welche Funktion das Zertifikat nutzt.
- **Verschlüsselungs-Algorithmus**
  Zeigt an, welches kryptografisches Verfahren verwendet wird.
- **Schlüsselverwendung**
  Zeigt an, für welchen Zweck der zum Zertifikat gehörende Schlüssel verwendet wird, z. B. zum Verifizieren digitaler Signaturen.
- **Erweiterte Schlüsselverwendung**
  Zeigt an, ob der Verwendungszweck noch zusätzlich beschränkt ist, z. B. nur zum Verifizieren von Signaturen des CA-Zertifikats.
- **Schlüsseldatei**
  Zeigt die Schlüsseldatei an.
- **Zertifikatssperrliste 1. URL**
  Tragen Sie die URL ein, über die die Sperrliste abgerufen werden kann. Nur editierbar, wenn vom Zertifikat unterstützt.
- **Zertifikatssperrliste 2. URL**
  Tragen Sie eine Alternativ-URL ein. Wenn die Sperrliste über die 1. URL nicht abrufbar ist, wird die Alternativ-URL verwendet. Nur editierbar, wenn vom Zertifikat unterstützt.
- **Zertifikat**
  Zeigt den Namen des Zertifikats an.
- **Passphrase**
  Tragen Sie das Passwort für das Zertifikat ein. Nur editierbar, wenn die verschlüsselte Datei Passwort-geschützt ist.
- **Passphrase bestätigen**
  Tragen Sie das Passwort nochmals ein. Nur editierbar, wenn die verschlüsselte Datei Passwort-geschützt ist.

Firewall

Allgemein

Auf dieser Seite aktivieren Sie die Firewall.

Hinweis

Bitte beachten Sie, wenn Sie die Firewall deaktivieren, dann ist ihr internes Netz ungeschützt.

Beschreibung

Die Seite enthält Folgendes:

- **Firewall aktivieren**
  Wenn aktiviert, ist die Firewall aktiv.
- **TCP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die TCP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 86400 Sekunden
- **UDP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die UDP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 300 Sekunden
- **ICMP Idle Timeout [s]**
  Geben Sie die gewünschte Zeitspanne in Sekunden ein. Wenn kein Datenaustausch stattfindet, wird nach Ablauf dieser Zeitspanne die ICMP-Verbindung automatisch getrennt. Der Wertebereich ist 1 bis 21474836.
  Default-Einstellung: 300 Sekunden

Vordefinierte IPv4-Regeln

Die Seite enthält vordefinierte IP-Paketfilter-Regeln. Wenn Sie eigene IP-Paketfilter-Regeln anlegen, haben diese eine höhere Priorität als die vordefinierten IP-Paketfilter-Regeln.

Hier kann man einstellen, welche Dienste des Gerätes von welcher Schnittstelle/Subnetz aus erreichbar sein sollen.

Beschreibung

- **Schnittstelle**
  Schnittstelle, auf die sich die Einstellung bezieht. Die Liste der Schnittstellen/Subnetze ist dynamisch und richtet sich nach den Einstellungen aus "Layer 3 >Subnetz".
  - pppx bzw, usb0 (nur beim M876-4): Erlaubt den Zugriff von der WAN-Schnittstelle auf das Gerät.
  - VLANx: Erlaubt den Zugriff vom IP-Subnetz auf das Gerät.
- Der Zugriff auf folgende IPv4-Dienste wird erlaubt:
  - Alle
    Alle IPv4-Dienste
  - HTTP
    Zum Zugriff auf das Web Based Management.
  - HTTPS
    Zum gesicherten Zugriff auf das Web Based Management.

    Hinweis

    HTTP und HTTPS deaktiviert

    Wenn Sie HTTP und HTTPS deaktivieren, ist das WBM des Geräts nicht mehr erreichbar.

    HTTPS deaktiviert

    Wenn Sie HTTPS deaktivieren, können Sie nur noch über HTTP auf das WBM zugreifen. Vorausgesetzt, unter "System > Konfiguration > HTTP-Dienste" ist "HTTP & HTTPS" eingestellt. Ist z. B. "HTTP nach HTTPS umleiten" eingestellt, kann der Zugriff über HTTP nicht nach HTTPS umgeleitet werden. Somit ist das WBM des Geräts nicht mehr erreichbar.

  - DNS
    DNS-Anfragen an das Gerät. Nur notwendig, wenn am Gerät die Funktion "DNS-Relay" aktiv ist.
  - SNMP
    Eingehende SNMP-Verbindungen. Notwendig, um z. B. mit einem MIB-Browser auf die SNMP-Informationen des Geräts zuzugreifen.
  - Telnet
    Zum unverschlüsselten Zugriff auf das CLI.
  - SMS Relay (nur beim M87x)
    Zum Versenden von SMS aus dem lokalen Netz.
  - IPsec VPN
    Erlaubt den IKE (Internet Key Exchange) Datenverkehr vom externen Netz zum Gerät. Notwendig, wenn eine IPsec VPN-Gegenstelle eine Verbindung zu diesem Gerät herstellen soll.
  - SSH
    Zum verschlüsselten Zugriff auf das CLI.
  - DHCP
    Zugriff auf den DHCP-Server oder den DHCP-Client

- Ping
  Zugriff auf die Ping-Funktion
- Systemzeit
  Zugriff auf NTP und SNTP

IP-Dienste

Auf dieser Seite definieren Sie IP-Dienste. Mithilfe der IP-Dienst-Definitionen können Sie Firewall-Regeln definieren, die auf bestimmte Dienste angewendet werden. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Namen.

Beschreibung

Die Seite enthält Folgendes:

- **Name des Diensts**
  Tragen Sie den Namen für den IP-Dienst ein. Der Name muss eindeutig sein.

Die Tabelle enthält folgende Spalten:

- **Name des Diensts**
  Zeigt den Namen des IP-Diensts an.
- **Transport**
  Legen Sie den Protokolltyp fest.
  - UDP
    Die Regel gilt nur für UDP-Telegramme.
  - TCP
    Die Regel gilt nur für TCP-Telegramme.
- **Quell-Port (Bereich)**
  Tragen Sie den Quell-Port ein. Die Regel gilt genau für den angegebenen Port.
  - Wenn die Regel für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Regel für alle Ports gelten soll, geben Sie "*" ein.
- **Ziel-Port (Bereich)**
  Tragen Sie den Ziel-Port ein. Die Regel gilt genau für den angegebenen Port.
  - Wenn die Regel für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Regel für alle Ports gelten soll, geben Sie "*" ein.

ICMP-Dienste

Auf dieser Seite definieren Sie ICMP-Dienste. Mithilfe der ICMP-Dienst-Definitionen können Sie Firewall-Regeln definieren, die auf bestimmte Dienste angewendet werden. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Namen.

Beschreibung

Die Seite enthält Folgendes:

- **Name des Diensts**
  Tragen Sie einen Namen für den ICMP-Dienst ein. Der Name muss eindeutig sein.

Die Tabelle enthält folgende Spalten:

- **Name des Diensts**
  Zeigt den Namen des ICMP-Diensts an.
- **Protokoll**
  Zeigt die Version des ICMP-Protokolls an.
- **Typ**
  Legen Sie den ICMP-Pakettyp fest. Einige Beispiele sind:
  - Destination Unreachable
    IP-Telegramm kann nicht zugestellt werden.
  - Time Exeeded
    Zeitlimit überschritten
  - Echo-Request
    Echo-Frage, besser bekannt als Ping.
- **Code**
  Der Code beschreibt den ICMP-Pakettyp genauer. Die Auswahl ist abhängig vom gewählten ICMP-Pakettyp.
  Bei "Destination Unreachable" ist z. B. "Code 1" Host ist nicht erreichbar.

IP-Protokolle

Auf dieser Seite können Sie benutzerdefinierte Protokolle konfigurieren, z. B. IGMP für Multicast-Gruppen. Sie vergeben hierbei einen Protokollnamen und ordnen diesem die Dienstparameter zu. Bei der Projektierung der IP-Regeln verwenden Sie dann einfach diesen Protokollnamen.

Beschreibung

Die Seite enthält Folgendes:

- **Protokollname**
  Tragen Sie einen Namen für das Protokoll ein.

Die Seite enthält folgende Optionskästchen:

- **Protokollname**
  Zeigt den Protokollnamen an.
- **Protokollnummer**
  Tragen Sie die Protokollnummer ein, z. B. 2. Eine Liste der Protokollnummern finden Sie auf den Internetseiten von iana.org

IP-Regeln

Auf dieser Seite legen Sie eigene IP-Paketfilter-Regeln für die Firewall fest.

Die hier erstellten IP-Paketfilter-Regeln haben Vorrang:

- vor den vordefinierten IP-Paketfilter-Regeln (Predefined IPv4) und
- vor den IP-Paketfilter-Regeln, die aufgrund einer Verbindungsprojektierung (SINEMA RC) automatisch angelegt werden.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Selektieren**
  Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **Protokoll**
  Zeigt die Version des IP-Protokolls an.
- **Aktion**
  Wählen Sie aus, wie mit eintreffenden IP-Paketen zu verfahren ist:
  - "Accept" – Die Datenpakete dürfen passieren,
  - "Reject" – Die Datenpakete werden abgewiesen, der Absender erhält eine entsprechende Meldung,
  - "Drop" – Die Datenpakete werden ohne Rückmeldung an den Absender verworfen.
- **Von / Nach**
  Legen Sie die Kommunikationsrichtung der IP-Regel fest.
  - VLANx: VLANs mit konfiguriertem Subnetz
  - Gerät
  - SINEMA RC: Verbindung zum SINEMA RC-Server
  - IPsec: Entweder alle IPsec VPN-Verbindungen (all) oder eine spezifische IPsecVPN-Verbindung
  - pppx bzw. usb0: WAN-Schnittstelle
  - OpenVPN: Entweder alle OpenVPN-Verbindungen (all) oder eine spezifische OpenVPN-Verbindung
- **Quelle (Bereich)**
  Tragen Sie die IP-Adresse oder einen IP-Bereich ein, die IP-Pakete senden darf.
  - Wenn die Regel für einen IP-Bereich gelten soll, geben Sie den Bereich mit Startadresse "-" Endadresse an, z. B. 192.168.100.10 - 192.168.100.20.
  - Wenn die Regel für alle IP-Adressen gelten soll, geben Sie " 0.0.0.0/0" ein.
- **Ziel (Bereich)**
  Tragen Sie die IP-Adresse oder einen IP-Bereich ein, die IP-Pakete empfangen darf.
  - Wenn die Regel für einen IP-Bereich gelten soll, geben Sie den Bereich mit Startadresse "-" Endadresse an, z. B. 192.168.100.10 - 192.168.100.20.
  - Wenn die Regel für alle IP-Adressen gelten soll, geben Sie " 0.0.0.0/0" ein.
- **Dienst**
  Wählen Sie den Dienst oder den Protokollnamen aus, für den diese Regel gültig ist.

- **Log**
  Legen Sie fest, ob das Zutreffen der Regel protokolliert wird und welche Ereignisschwere der Eintrag hat.
  Folgende Einstellungen gibt es:
  - none
    Das Zutreffen wird nicht protokolliert.
  - info / warning / critical
    Das Zutreffen wird mit der gewählten Ereignisschwere protokolliert. Die Logdatei wird unter "Information" > "Log-Tabellen" > "Firewall-Log" angezeigt.
- **Reihenfolge**
  Legen Sie die Reihenfolge der Regel fest.

Hinweis

WBM über VPN-Tunnel öffnen

Um das WBM über einen VPN-Tunnel zu öffnen, legen Sie die folgende IP-Regel an:

Aktion: Accept

Von: IPsec

Nach: Gerät

Dienst: HTTPS (TCP-Port 443)

IPsec VPN

Übersicht

Um eine IPsec VPN-Verbindung zu erstellen gibt es folgende Möglichkeiten

Automatische Erstellen über die VPN-Gruppen in STEP7

Wenn Sie über den

Weiterführende Informationen zum Erstellen von VPN-Gruppen für IPsec VPN-Verbindungen finden Sie im Kapitel "IPsec VPN-Tunnel: VPN-Gruppen anlegen und zuordnen".

STEP7 über VPN-Gruppen

Alternativ können Sie die IPsec-Verbindungen über VPN-Gruppen projektieren, siehe Kapitel "IPsec VPN-Tunnel: VPN-Gruppen anlegen und zuordnen". Bei der Projektierung über VPN-Gruppen werden die Remote-Adressen und Remote-Subnetze von STEP 7 anhand der Schnittstellennamen ermittelt:

- Schnittstellenname "INT": STEP 7 behandelt die Schnittstelle als interne Schnittstelle.
- Schnittstellenname "EXT": STEP 7 behandelt die Schnittstelle als externe Schnittstelle.

STEP 7 legt beim Hinzufügen von Geräten in VPN-Gruppen ausschließlich VPN-Verbindungen zwischen den externen Schnittstellen an. Nur die lokalen Subnetze an den

internen Schnittstellen werden dadurch über VPN miteinander verbunden. Zusätzliche VPN-Verbindungen müssen Sie manuell projektieren.

Besonderheiten bei der Projektierung über VPN-Gruppen

- Die Erlaubnis zur Initiierung des Verbindungsaufbaus legen Sie bei der Zuweisung zu VPN-Gruppen fest.
- In den verbindungsspezifischen VPN-Einstellungen wird das Gerät als Verbindungspartner nicht editierbar angezeigt. Bei der Auswahl des Geräts kann die Schnittselle anderer Partnergeräte angepasst werden.
- Das Gerät verhält sich bezüglich der VPN-Betriebsart stets als Routing-Gerät.
- Die Eigenschaften einer VPN-Gruppe werden in die lokalen Eigenschaften von VPN-Gruppenteilnehmern übernommen. Die Eigenschaften einer VPN-Gruppe können in Bezug auf folgende Parameter lokal überschrieben werden:
  - Phase 1: Keying-Versuche, DPD, DPD-Zeitraum, DPD-Timeout
  - Phase 2: Lifebytes, Protokoll, Port (Bereich), Auto-Firewallregeln

Allgemein

Auf der Seite konfigurieren Sie die Grundeinstellungen für VPN.

Beschreibung

Die Seite enthält Folgendes:

- **IPsec VPN aktivieren**
  Aktivieren oder deaktivieren Sie das IPsec-Verfahren für VPN.
- **CRL-Richtlinie strikt durchsetzen**
  Wenn aktiviert, wird die Gültigkeit der Zertifikate anhand der Zertifikatssperrliste (CRL-Certificate Revocation List) überprüft. In der Zertifikatssperrliste sind die von der Zertifizierungsstelle ausgestellten Zertifikate aufgeführt, die vor ihrem gesetzten Ablaufdatum ihre Gültigkeit verloren haben. Welche Zertifikatssperrliste verwendet wird, konfigurieren Sie auf der Seite "Zertifikate (Seite 2850)".
- **NAT Keep Alive-Zeitintervall**
  Legen Sie fest, in welchen Zeitabständen Lebenszeichentelegramme (Keep Alive) gesendet werden. Befindet sich ein NAT-Gerät zwischen zwei VPN-Endpunkten, dann wird bei Inaktivität die Verbindung aus dessen dynamischer NAT-Tabelle gelöscht. Um dies zu verhindern, werden die Lebenszeichentelegramme gesendet.

Remote-Endpunkt

Auf dieser Seite konfigurieren Sie die Gegenstelle (VPN-Endpunkt).

Beschreibung

Die Seite enthält Folgendes:

- **Name Remote-Endpunkt**
  Tragen Sie einen Namen für die Gegenstelle ein und klicken Sie auf "Neuer Eintrag", um eine neue Gegenstelle zu erstellen.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der Gegenstelle an.
- **Remote-Modus**
  Legen Sie fest, welche Rolle die Gegenstellen einnimmt.
  - Roadwarrior
    Die erreichbaren Gegenstellen (Remote-Adresse) werden eingetragen. Die erreichbaren Remote-Subnetze werden von der Gegenstelle gelernt.
  - Standard
    Die erreichbare Gegenstelle (Remote-Adresse) und die erreichbaren Remote-Subnetze werden fest eingetragen.
- **Remote-Typ**
  Legen Sie die Art der Gegenstellen-Adresse fest.
  - Manuell
    Die Adresse der Gegenstelle ist bekannt. Das Gerät kann entweder als VPN-Client die VPN-Verbindung aktiv aufbauen, oder passiv auf den Verbindungsaufbau durch die Gegenstelle warten.
  - Beliebig
    Nimmt die Verbindung von Gegenstellen mit beliebiger IP-Adresse an. Das Gerät kann nur auf VPN-Verbindungen warten, aber nicht als aktiver Partner einen VPN-Tunnel aufbauen.
- **Remote-Adresse**
  Nur beim Remote-Typ "Manuell" editierbar.
  - Im Standard-Modus tragen Sie die WAN-IP-Adresse oder den DDNS-Hostnamen der Gegenstelle ein. Die Netzmaske ist immer /32
  - Im Roadwarrior-Modus können Sie entweder die Adresse der Gegenstelle vorgeben oder einen IP-Bereich eintragen, aus dem Verbindungen entgegengenommen werden.
- **Remote-Subnetz**
  - Im Standard-Modus geben Sie das erreichbare Subnetz der Gegenstelle ein. Verwenden Sie die CIDR-Schreibweise.
  - Im Roadwarrior-Modus teilt die Gegenstelle dem Gerät seinen erreichbaren Subnetze mit und das Gerät lernt diese.

- **Virtueller IP-Modus**
  Legen Sie fest, ob der Gegenstelle eine virtuelle IP-Adresse angeboten wird.
  Folgende Möglichkeiten gibt es:
  - Benutzerdefinierte IPv4
    Die virtuelle IP-Adresse ist aus dem bei "Virtuelle IP" festgelegten Band.
  - Keine
    Keine virtuelle IP-Adresse. Der VPN-Tunnel wird dynamisch zur internen IP-Adresse der Gegenstelle aufgebaut.
- **Virtuelle IP**
  Legen Sie das Subnetz fest (CIDR), aus dem die Gegenstelle eine virtuelle IP-Adresse angeboten bekommt.
  Nur editierbar, wenn bei "Virtueller IP-Modus" "Benutzerdefinierte IPv4" ausgewählt ist.

Verbindungen

Auf dieser Seite konfigurieren Sie die Grundeinstellungen für die VPN-Verbindung. Mit diesen Einstellungen kann das Gerät (lokaler Endpunkt) einen ungesicherten VPN-Tunnel zur Gegenseite aufbauen. Die Sicherheitseinstellungen legen Sie auf der Seite "Authentifizierung" fest.

Hinweis

Mehrere IPsec VPN-Verbindungen über denselben VPN-Endpunkt

Wenn Sie über denselben VPN-Endpunkt IPsec VPN-Verbindungen zu verschiedenen Remote-Subnetzen angelegt haben, ist die zuerst projektierte VPN-Verbindung (niedrigster Index) die Hauptverbindung (Parent).

Über die Hauptverbindung werden alle weiteren IPsec VPN-Verbindungen (Child) angelegt und aufgebaut. Wenn nun alle VPN-Tunnel aufgebaut sind und die Hauptverbindung beendet wird, so werden allen Child-Verbindungen unterbrochen. Nach Ablauf des DPD-Timeouts werden alle IPsec VPN-Verbindungen wieder neu über Hauptverbindung aufgebaut.

Wenn nur eine Child-Verbindung beendet wird, bleiben die Hauptverbindung und die weiteren Child-Verbindungen bestehen.

Hinweis

Wenn Sie "NETMAP" verwenden,

- werden nur Auto-Firewall-Regeln unterstützt.
- ist bei "Betrieb" die Einstellung "Auf Anforderung" nicht auswählbar

---

Beschreibung

Die Seite enthält folgende Felder:

- **Verbindungsname**
  Geben Sie einen Namen für die VPN-Verbindung ein.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an.
- **Betrieb**
  Legen Sie fest, von wem die VPN-Verbindung aufgebaut wird. Weitere Informationen dazu finden Sie unter "Technische Grundlagen > VPN-Verbindungsaufbau (Seite 2718)".
  - Deaktiviert
    Die VPN-Verbindung ist deaktiviert.
  - Starten
    Das Gerät versucht, zu der Gegenstelle eine VPN-Verbindung aufzubauen.
  - Warten
    Das Gerät wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
  - Auf Anforderung
    Die VPN-Verbindung wird bei Bedarf aufgebaut.
  - Bei DI starten
    Beim Eintreten des Ereignisses "Digitaler Eingang" versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration (Seite 2762)" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Auf DI warten
    Beim Eintreten des Ereignisses "Digitaler Eingang" wartet das Gerät, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration (Seite 2762)" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Bei SMS starten (nur bei M87x)
    Wenn das Gerät eine Befehl-SMS erhält, versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen. Vorausgesetzt, dass das Gerät eine Befehl-SMS der Klasse "System" von bestimmten Absendern akzeptiert. Die Absender konfigurieren Sie unter "System > SMS > Befehl-SMS (Seite 2789)".
  - Auf SMS warten (nur bei M87x)
    Wenn das Gerät eine Befehl-SMS erhält, wartet das Gerät, bis der Verbindungsaufbau von der Gegenstelle initiiert wird. Vorausgesetzt, dass das Gerät eine Befehl-SMS der Klasse "System" von bestimmten Absendern akzeptiert.Die Absender konfigurieren Sie unter "System > SMS > Befehl-SMS (Seite 2789).
- **Keying-Protokoll**
  Legen Sie fest, ob IKEv2 oder IKEv1 verwendet wird.
- **Remote-Endpunkt**
  Wählen Sie die gewünschte Gegenstelle aus. Nur die Gegenstellen sind projektierbar, die Sie auf der Seite "Remote-Endpunkt" konfiguriert haben.
- **Lokales Subnetz**
  Tragen Sie das lokale Subnetz ein. Verwenden Sie die CIDR-Schreibweise. Das lokale Netz kann auch nur ein einzelner PC, oder eine andere Untermenge des lokalen Netzes sein.

- **Virtuelle IP anfordern**
  Wenn aktiviert, wird beim Verbindungsaufbau eine virtuelle IP-Adresse von der Gegenstelle angefordert.
- **Timeout [Sek]**
  Nur bei der Einstellung "Auf Anforderung" notwendig. Tragen Sie die Zeitspanne ein, nach der die VPN-Verbindung getrennt wird. Wenn innerhalb dieser Zeit keine Pakete gesendet werden, wird der VPN-Verbindung automatisch getrennt.

Authentifizierung

Auf dieser Seite legen Sie fest, wie sich die VPN-Verbindungspartner gegenseitig authentisieren.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Authentifizierung**
  Wählen Sie das Authentifizierungsverfahren aus. Voraussetzung für die VPN-Verbindung ist, dass die Gegenstelle das gleiche Authentisierungsverfahren verwendet.
  - Deaktiviert
    Es ist kein Authentifizierungsverfahren gewählt. Ein Verbindungsaufbau ist nicht möglich.
  - Remote-Zert
    Für die Authentifizierung wird das Gegenstellenzertifikat verwendet. Das Zertifikat legen Sie bei "Remote-Zertifikat" fest
  - CA-Zert
    Für die Authentifizierung wird das Zertifikat der Zertifizierungsstelle verwendet. Das Zertifikat legen Sie bei "CA-Zertifikat" fest. Wenn Sie eine IPsec-Verbindung zu SINEMA RC aufbauen wollen, müssen Sie das Zertifikat nachträglich über das WBM hinzufügen und auf das Gerät laden.
  - PSK
    Für die Authentifizierung wird ein Schlüssel verwendet. Den Schlüssel konfigurieren Sie bei "PSK".
- **CA-Zertifikat**
  Wählen Sie das Zertifikat aus. Nur geladene Zertifikate sind auswählbar.
- **Lokales Zertifikat**
  Wählen Sie das Gerätezertifikat aus. Nur die Gerätezertifikat sind auswählbar, die vom TIA Portal erzeugt wurden.
- **Lokale-ID**
  Geben Sie die Lokale-ID aus dem Gegenstellenzertifikat ein. Nur wenn Sie das Gegenstellenzertifikat verwenden, können Sie das Feld leer lassen. Das Feld wird automatisch mit dem Wert aus dem Gegenstellenzertifikat befüllt.

- **Remote-Zertifikat**
  Wählen Sie das Gegenstellenzertifikat aus. Nur die Gegenstellenzertifikate sind auswählbar, die vom TIA Portal erzeugt wurden.
- **Remote-ID**
  Geben Sie den "Distinguished Name" oder "Alternate Name" aus dem Gegenstellenzertifikat ein. Nur wenn Sie das Gegenstellenzertifikat verwenden, können Sie das Feld leer lassen. Das Feld wird automatisch mit dem Wert aus dem Gegenstellenzertifikat befüllt.
- **PSK**
  Geben Sie den Schlüssel ein.
- **PSK bestätigen**
  Wiederholen Sie den Schlüssel.

Phase 1

Phase 1: Verschlüsselungsvereinbarung und Authentisierung (IKE = Internet Key Exchange)

Auf dieser Seite stellen Sie die Parameter für das Protokoll des IPsec-Schlüsselmanagements ein. Der Schlüsselaustausch erfolgt über das standardisierte Verfahren IKE, für das Sie folgende Protokollparameter einstellen können.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Default-Chiffre**
  Wenn aktiviert, wird beim Verbindungsaufbau eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Die Auswahl ist abhängig vom Schlüsselaustauschverfahren. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 2713)".
- **Verschlüsselung**
  Wählen Sie für die Phase 1 den gewünschten Verschlüsselungsalgorithmus aus. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Die Auswahl ist abhängig vom Schlüsselaustauschverfahren. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 2713)".

  Hinweis

  Die AES-Modi CCM und GCM beinhalten separate Mechanismen für die Authentisierung von Daten. Wenn Sie bei "Verschlüsselung" einen Modus AES x CCM verwenden, dann wird dieser auch zur Authentisierung verwendet. Von dem Parameter "Authentifizierung" wird dann nur noch die Pseudo-Random-Funktion abgeleitet. Damit eine VPN-Verbindung aufgebaut wird, müssen alle Geräte die gleichen Einstellungen verwenden.

- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende Verfahren werden unterstützt:
  - MD5
  - SHA1
  - SHA512
  - SHA256
  - SHA384
- **Schlüsselableitung**
  Wählen Sie die gewünschte Diffie-Hellmann-Gruppe (DH), aus der ein Schlüssel erzeugt wird. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende DH-Gruppen werden unterstützt:
  - DH Group 1
  - DH Group 2
  - DH Group 5
  - DH Group 14
  - DH Group 15
  - DH Group 16
  - DH Group 17
  - DH Group 18
- **Keying-Versuche**
  Tragen Sie die Anzahl der Wiederholungen für einen fehlgeschlagenen Verbindungsaufbau ein. Wenn Sie den Wert 0 eintragen, wird der Verbindungsaufbau unendlich oft zu versucht.
- **Lebensdauer [min]**
  Tragen Sie einen Zeitraum in Minuten ein, der die Lebensdauer der Authentisierung festlegt. Nach Ablauf der Zeit müssen sich die beteiligten VPN-Endpunkte erneut gegenseitig authentisieren und einen neuen Schlüssel erzeugen.
- **DPD**
  Wenn aktiviert, wird DPD verwendet. Mit DPD lässt sich feststellen, ob die VPN-Verbindung noch besteht oder ob sie abgebrochen ist.

  Hinweis

  Durch das Versenden der DPD-Anfragen steigt die Anzahl der gesendeten und empfangenen Daten. Dies kann zu erhöhten Kosten führen.

- **DPD-Zeitraum [Sek]**
  Tragen Sie eine Zeitspanne ein, nach der DPD-Anfragen gesendet werden. Diese Anfragen testen, ob die Gegenstelle noch verfügbar ist

- **DPD-Timeout [Sek]**
  Tragen Sie eine Zeitspanne ein. Wenn auf die DPD-Anfragen keine Antwort erfolgt, dann wird nach Ablauf dieser Zeit die Verbindung zur Gegenstelle für ungültig erklärt.
- **Aggressive Mode**
  - Deaktiviert:
    Main Mode wird verwendet.
  - Aktiviert
    Aggressive Mode wird verwendet

  Der Unterschied zwischen Main- und Aggressive Mode ist die "Identity-Protection", die im Main Mode verwendet wird. Die Identität wird im Main Mode verschlüsselt übertragen, im Aggressive Mode nicht.

Phase 2

Phase 2: Datenaustausch (ESP = Encapsulating Security Payload)

Auf dieser Seite stellen Sie die Parameter für das Protokoll des IPsec-Datenaustauschs ein. Die gesamte Kommunikation in dieser Phase erfolgt verschlüsselt über das standardisierte Sicherheitsprotokoll ESP, für das Sie folgende Protokollparameter einstellen können.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Default-Chiffre**
  Wenn aktiviert, wird beim Verbindungsaufbau eine vorgegebene Liste an den VPN-Verbindungspartner übermittelt. In der Liste sind Kombinationen aus den drei Algorithmen (Encryption, Authentication, Key Derivation) enthalten. Um eine VPN-Verbindung aufzubauen, muss der VPN-Verbindungspartner mindestens eine dieser Kombinationen unterstützen. Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 2713)".
- **Verschlüsselung**
  Wählen Sie für die Phase 2 den gewünschten Verschlüsselungsalgorithmus aus. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Weitere Informationen dazu erhalten Sie bei "IPsec VPN (Seite 2713)".

  Hinweis

  Die AES-Modi CCM und GCM beinhalten separate Mechanismen für die Authentisierung von Daten. Wenn Sie bei "Verschlüsselung" einen Modus AES x CCM oder AES x GCM verwenden, dann wird dieser auch zur Authentisierung verwendet. Von dem Parameter "Authentifizierung" wird dann nur noch die Pseudo-Random-Funktion abgeleitet.

- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende Verfahren werden unterstützt:
  - MD5
  - SHA1
  - SHA512
  - SHA256
  - SHA384
- **Schlüsselableitung**
  Wählen Sie die gewünschte Diffie-Hellmann-Gruppe (DH), aus der ein Schlüssel erzeugt wird. Nur auswählbar, wenn "Default-Chiffre" deaktiviert ist.
  Folgende DH-Gruppen werden unterstützt:
  - None: Für die Phase 2 werden keine separaten Schlüssel ausgetauscht. Damit ist Perfect Forward Secrecy (PFS) deaktivert.
  - DH Group 1
  - DH Group 2
  - DH Group 5
  - DH Group 14
  - DH Group 15
  - DH Group 16
  - DH Group 17
  - DH Group 18

  Hinweis

  Damit eine VPN-Verbindung aufgebaut wird, müssen alle Geräte die gleichen Einstellungen verwenden oder kompatible Verschlüsselungsverfahren anbieten.

- **Lebensdauer [min]**
  Tragen Sie einen Zeitraum in Minuten ein, der die Lebensdauer der vereinbarten Schlüssel festlegt. Nach Ablauf der Zeit wird der Schlüssel neu ausgehandelt.
- **Lifebytes**
  Tragen Sie das Datenlimit in Bytes ein, das die Lebensdauer der vereinbarten Schlüssel festlegt. Nach Ablauf des Datenlimits wird der Schlüssel neu ausgehandelt.
- **Protokoll**
  Legen Sie fest, für welches Protokoll die VPN-Verbindung gültig ist, z. B. UDP, TCP, ICMP. Wenn die Einstellung für alle Protokolle gelten soll, geben Sie "*" ein.

- **Port (Bereich)**
  Legen Sie den Port fest, durch den der VPN-Tunnel kommunizieren kann. Die Einstellung gilt genau für den angegebenen Port
  - Wenn die Einstellung für einen Port-Bereich gelten soll, geben Sie den Bereich mit Start-Port "-" End-Port an, z. B. 30 - 40.
  - Wenn die Einstellung für alle Ports gelten soll, geben Sie "*" ein.

  Die Einstellung hat nur bei portbasierten Protokollen Auswirkungen.
- **Auto-Firewallregeln**
  - akiviert
    Für die VPN-Verbindung werden automatisch die Firewall-Regeln angelegt.
  - deaktiviert
    Sie müssen selbst die Firewall-Regeln anlegen.

OpenVPN-Client

Allgemein

Auf dieser Seite aktivieren Sie den OpenVPN-Client.

Beschreibung

Die Seite enthält Folgendes:

- **OpenVPN-Client aktivieren**
  Aktivieren oder deaktivieren Sie den OpenVPN-Client.

Verbindungen

Auf dieser Seite konfigurieren Sie die Grundeinstellungen für die OpenVPN-Verbindung. Die Sicherheitseinstellungen legen Sie auf der Seite "Authentifizierung" fest.

Beschreibung

- **Verbindungsname**
  Geben Sie einen eindeutigen Namen für die OpenVPN-Verbindung ein.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der OpenVPN-Verbindung an.
- **Betrieb**
  Legen Sie fest, wie die Verbindung aufgebaut wird. Weitere Informationen dazu finden Sie unter "Technische Grundlagen > VPN-Verbindungsaufbau (Seite 2718)".
  - Starten
    Das Gerät versucht, zu der Gegenstelle eine VPN-Verbindung aufzubauen.
  - Bei DI starten
    Beim Eintreten des Ereignisses "Digitaler Eingang" versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen.
    Vorausgesetzt, dass das Ereignis "Digitaler Eingang" an die VPN-Verbindung weitergegeben wird. Dazu aktivieren Sie unter "System > Ereignisse > Konfiguration" beim Ereignis "Digitaler Eingang" "VPN-Tunnel".
  - Bei·SMS·starten (nur bei M87x)
    Wenn das Gerät eine Befehl-SMS erhält, versucht das Gerät zu der Gegenstelle eine VPN-Verbindung aufzubauen. Vorausgesetzt, dass das Gerät eine Befehl-SMS der Klasse "System" von bestimmten Absendern akzeptiert. Die Absender konfigurieren Sie unter "System > SMS > Befehl-SMS".
  - Deaktiviert
    Die VPN-Verbindung ist deaktiviert.
- **Verschlüsselung**
  Wählen Sie den gewünschten Verschlüsselungsalgorithmus aus.
  - AES-128-CBC (Default)
  - AES-192-CBC
  - AES-256-CBC
  - DES-EDE3
  - BF-CBC
- **Authentifizierung**
  Legen Sie das Verfahren zum Berechnen der Prüfsumme fest.
  - SHA256 (Default)
  - SHA384
  - SHA512
  - SHA224
  - SHA1
  - MD5

- **LZO Komp.**
  - Deaktiviert (-)
    Die Komprimierung ist deaktiviert. Der Server kann Komprimierung nicht wieder einschalten.
  - Nein
    Die Komprimierung ist standardmäßig deaktiviert. Der Server kann Komprimierung einschalten
  - Ja
    Die Komprimierung standardmäßig aktiviert. Der Server kann die Komprimierung ausschalten
  - Adaptiv
    Komprimierung wird standardmäßig adaptiv aktiviert. Nur wenn gut komprimierbare Daten vorliegen wird komprimiert, ansonsten wird die Kompression für eine gewisse Zeit deaktiviert.
- **Auto-Firewallregeln**
  - Aktiviert
    Für die VPN-Verbindung werden automatisch die Firewall-Regeln angelegt.
  - Deaktiviert
    Sie müssen selbst die geeigneten Firewall-Regeln anlegen.
- **NAT aktivieren**
  Mit dieser Einstellung aktivieren Sie automatisches IP-Masquerading für diese Schnittstelle. Die lokalen Geräte sind von außen nicht direkt erreichbar, sondern nur über die IP-Adresse der Schnittstelle. Die lokalen Geräte können sich aber mit den Geräten hinter dem OpenVPN-Server verbinden. Weitere Informationen zu NAT finden Sie unter "Technische Grundlagen > NAT".

Remote

Auf dieser Seite konfigurieren Sie die Gegenstelle (OpenVPN-Endpunkt). Pro Verbindung können Sie mehrere OpenVPN.Gegenstellen festlegen. Das Gerät probiert alle projektierten OpenVPN-Gegenstellen der Reihe nach aus bis ein Verbindungsaufbau erfolgreich ist.

Beschreibung

Die Seite enthält Folgendes:

- **Remote-Name**
  Geben Sie einen Namen für die OpenVPN-Gegenstelle ein.

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der OpenVPN-Gegenstelle an.
- **Verbindung**
  Wählen Sie die zugehörige Verbindung aus. Nur die Verbindungen sind projektierbar, die Sie auf der Seite "Verbindungen" konfiguriert haben.

- **Remote-Adresse**
  Geben Sie die WAN-IP-Adresse oder den DNS-Hostnamen der OpenVPN-Gegenstelle ein.
- **Port**
  Legen Sie den Port fest, durch den der OpenVPN-Tunnel kommunizieren kann. Die Einstellung gilt genau für den angegebenen Port.
- **Protokoll**
  Legen Sie fest, für welches Protokoll die OpenVPN-Verbindung genutzt werden soll.

Authentifizierung

Auf dieser Seite legen Sie fest, wie sich die VPN-Verbindungspartner gegenseitig authentisieren.

Beschreibung

Die Tabelle enthält folgende Spalten:

- **Name**
  Zeigt den Namen der VPN-Verbindung an, auf die sich die Einstellungen beziehen.
- **Methode**
  Wählen Sie das Authentifizierungsverfahren aus. Voraussetzung für die VPN-Verbindung ist, dass die Gegenstelle das gleiche Authentisierungsverfahren verwendet.
  - Deaktiviert
    Es ist kein Authentifizierungsverfahren gewählt. Ein Verbindungsaufbau ist nicht möglich.
  - Zertifikate
    Für die Authentifizierung werden Zertifikate verwendet. Die Zertifikate müssen nachträglich über das WBM hinzugefügt und auf das Gerät geladen werden.
  - Benutzename/Passwort
    Für die Authentifizierung werden Benutzername/Passwort verwendet. Das Zertifikat muss nachträglich über das WBM hinzugefügt und auf das Gerät geladen werden.
- **CA-Zertifikat**
  Wählen Sie das CA-Zertifikat über das WBM aus und laden Sie es auf das Gerät.
- **Gerätezertifikat**
  Wählen Sie das Gerätezertifikat über das WBM aus und laden Sie es auf das Gerät.
- **Benutzername**
  Geben Sie den Benutzernamen an.
- **Passwort**
  Geben Sie das Passwort an.
- **Passwort bestätigen**
  Bestätigen Sie das Passwort.

1.4.9 PROFIBUS DP konfigurieren

1.4.9.1 Konfigurieren von dezentraler Peripherie

Im folgenden Kapitel erfahren Sie, wie Sie dezentrale Peripherie konfigurieren. Außerdem erhalten Sie grundsätzliche Informationen zur Konfiguration eines DP-Mastersystems.

Grundsätzliche Informationen

- Grundlagen zum Konfigurieren eines DP-Mastersystems (Seite 2871)
- DP-Slaves im Hardware-Katalog (Seite 2873)
- Aufbau einer Konfiguration mit einfachen DP-Slaves (Mono-Mastersystem) (Seite 2874)
- Aufbau einer Konfiguration mit intelligenten Slaves (I-Slaves) (Seite 2875)
- Aufbau einer Konfiguration mit zwei Mastersystemen (Seite 2876)

Übersicht zum Konfigurieren dezentraler Peripherie

Schritt	Beschreibung
1	Legen Sie ein DP-Mastersystem an (Seite 2878).
2	Konfigurieren Sie bei Bedarf die Schnittstellen des DP-Mastersystems (Seite 2881).
3	Fügen Sie einen DP-Slave ein und konfigurieren Sie diesen (Seite 2883).
4	Weisen Sie bei Bedarf DP-Slaves einer SYNC/FREEZE-Gruppe zu (Seite 2887).

Siehe auch

DPV1-Geräte konfigurieren (Seite 2927)

Direkten Datenaustausch konfigurieren (Seite 2915)

1.4.9.2 Grundlagen zum Konfigurieren eines DP-Mastersystems

Dezentrale Peripherie

Als dezentrale Peripherie werden DP-Mastersysteme bezeichnet, die aus DP-Master und DP-Slaves bestehen, über einen Bus verbunden sind und miteinander über das Protokoll PROFIBUS-DP kommunizieren.

Konfiguration der dezentralen Peripherie

Weil DP-Master bzw. DP-Slaves unterschiedliche Geräte sein können, sind hier nur die grundsätzlichen Vorgehensweisen zum Konfigurieren erläutert. Die Konfiguration der dezentralen Peripherie ist aber weitgehend identisch mit dem Vorgehen bei der Konfiguration eines zentralen Aufbaus.

Anlegen des DP-Mastersystems in der Netzsicht

Nachdem Sie per Drag & Drop aus dem Hardware-Katalog einen DP-Master und einen DP-Slave (z. B. eine CPU 315-2 DP und eine IM 153-1) in der Netzsicht platziert haben, verbinden Sie die beiden Geräte mit einem PROFIBUS-Subnetz.

Da in der Netzsicht nur die Geräte, aber nicht die Baugruppen konfiguriert werden können, müssen Sie für die Konfiguration von DP-Master und DP-Slave in die Gerätesicht wechseln.

Konfigurieren eines DP-Slaves

Wechseln Sie in die Gerätesicht, um den DP-Slave zu konfigurieren. In der Gerätesicht bestücken Sie den Baugruppenträger des DP-Slave mit Baugruppen und parametrieren DP-Kennungen und E/A-Adressen. In der Geräteübersicht und im Inspektorfenster können Sie die Einstellungen prüfen und bearbeiten.

Steckplatznummerierung bei dezentralen Peripheriegeräten

Abhängig von dem verwendeten Typ des DP-Slaves, beginnen die Steckplätze des DP-Slave entweder mit "0" oder mit "4".

Die Steckplatz-Nummerierung von DP-Slaves, wie z. B. bei ET 200M, ist abgeleitet aus dem Aufbau einer S7-300:

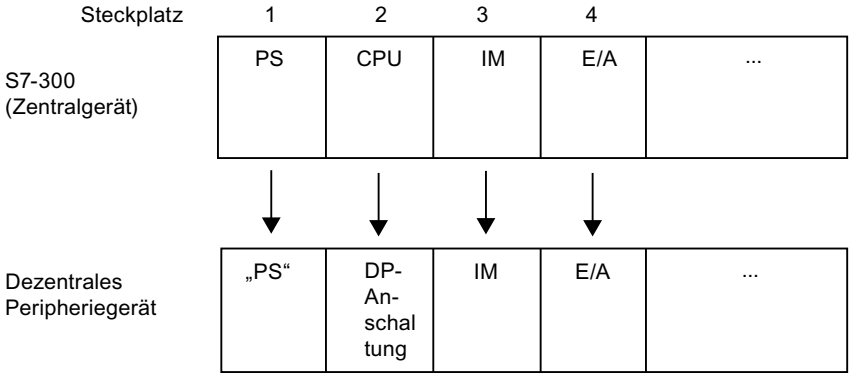

Steckplatzregeln für DP-Slaves nach S7-300

Für DP-Slaves, die nach dem Schema der S7-300 aufgebaut sind, gelten folgende Steckplatzregeln:

- Die eigentliche Peripherie (Ein-/Ausgänge) beginnt immer mit Steckplatz 4.
- Steckplatz 1 ist immer für eine Stromversorgungsbaugruppe (PS) reserviert, unabhängig davon, ob im realen Aufbau eine PS steckt oder nicht.
- Steckplatz 2 ist immer für die DP-Anschaltung reserviert.
- Steckplatz 3 ist immer für eine Erweiterungs-Anschaltung (IM) reserviert, unabhängig davon, ob ein reales Peripheriegerät erweiterbar ist oder nicht.

Dieses Schema wird auf alle modulare und kompakte DP-Slave-Typen angewendet, die auf dem Schema der S7-300 aufgebaut sind. Wichtig ist die Steckplatzzuordnung für die Auswertung von Diagnosemeldungen ("Diagnose-auslösender Steckplatz").

Weitere Informationen

Beachten Sie weitere Informationen zum Funktionsumfang, zu den Zugriffsverfahren etc. in den Handbüchern zu den jeweiligen Geräten bzw. in der Onlinehilfe zu speziellen FCs (z. B. DP-SEND und DP-RECEIVE für CP 342-5).

1.4.9.3 DP-Slaves im Hardware-Katalog

DP-Slaves im Hardware-Katalog

DP-Slaves finden Sie im Hardware-Katalog in dem Ordner "Dezentrale Peripherie". Dort finden Sie kompakte und modulare DP-Slaves:

- Kompakte DP-Slaves
  Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L
- Modulare DP-Slaves
  Anschaltungsbaugruppen mit zugeordneten S7-Baugruppen, z. B. ET 200M

Die verwendbaren DP-Slaves hängen von dem vorhandenen DP-Master und der gewünschten Funktionalität ab.

I-Slaves im Hardware-Katalog

Geräte, die Sie als intelligente DP-Slaves parametrieren können, finden Sie im Hardware-Katalog in dem Ordner "PLC". Als I-Slave können Sie beispielsweise folgende Geräte konfigurieren:

- CP 342-5 DP
  im Hardware-Katalog unter "PLC > SIMATIC S7-300 > CP > PROFIBUS".
- CPU 315-2 DP, CPU 317-2 DP, CPU 319-3 PN/DP
  im Hardware-Katalog unter "PLC > SIMATIC S7-300 > CPU".
- IM 151-7 CPU
  im Hardware-Katalog unter "PLC > SIMATIC ET200 PLC > ET 200S > CPU".

1.4.9.4 DP/DP-Koppler im Hardware-Katalog

Einführung

Ein DP/DP-Koppler dient dazu, zwei PROFIBUS DP-Netze als Netzübergang (Gateway) miteinander zu verbinden, und so Daten vom DP-Master des einen Netzes zum DP-Master des anderen Netzes zu übertragen.

Die maximale Größe der übertragbaren Daten beträgt 244 Byte Eingangsdaten und 244 Byte Ausgangsdaten.

DP/DP-Koppler im Hardware-Katalog

DP/DP-Koppler als Gateway zwischen zwei DP-Mastersystemen finden Sie im Hardware-Katalog in dem Ordner "Weitere Feldgeräte > PROFIBUS DP > Netzübergänge".

Projektieren des DP/DP-Kopplers

DP/DP-Koppler werden an beiden PROFIBUS-Netzen in jeweils eigenen Mastersystemen projektiert.

Die Ein- und Ausgangsbereiche beider Netze müssen dabei aufeinander abgestimmt sein. Die Ausgangsdaten der einen Seite des DP/DP-Kopplers werden als Eingangsdaten der jeweils anderen Seite übernommen und umgekehrt.

Siehe auch

GSD-Datei installieren (Seite 2923)

DP-Mastersystem anlegen (Seite 2878)

1.4.9.5 Konfigurationen mit PROFIBUS-DP

Konfigurationen mit "einfachen" DP-Slaves

Datenaustausch zwischen DP-Master und DP-Slave

In der Konfiguration mit einfachen DP-Slaves erfolgt der Datenaustausch zwischen DP-Master und einfachen DP-Slaves, also mit E/A-Baugruppen über den DP-Master. Der DP-Master pollt jeden projektierten DP-Slave in seiner Aufrufliste (Polling-Liste) innerhalb des DP-Mastersystems nacheinander ab und überträgt die Ausgangsdaten bzw. erhält deren Eingangswerte zurückgeliefert.

Mono-Master-System

Die Konfiguration mit nur einem DP-Master wird auch als Mono-Master-System bezeichnet. Ein einziger DP-Master ist mit seinen zugehörigen DP-Slaves an einem physikalischen PROFIBUS DP-Subnetz angeschlossen.

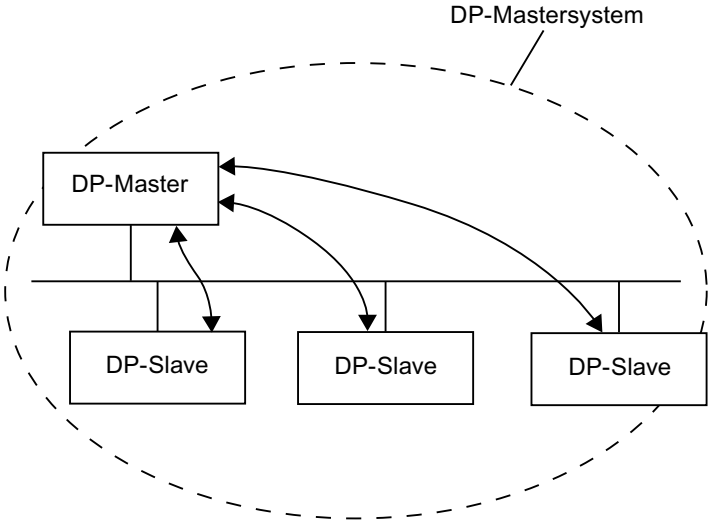

Konfigurationen mit mehreren DP-Mastersystemen

Konfigurationsarten mit zwei oder mehr DP-Mastersystemen

Mehrere DP-Mastersysteme an einem physikalischen PROFIBUS-DP Subnetz werden auch als Multi-Master-System bezeichnet.

Über den PROFIBUS-DP können diese DP-Mastersysteme auf unterschiedliche Arten verbunden werden:

- Ohne logische Verbindung der DP-Mastersysteme untereinander
- Direkter Datenaustausch DP-Slave > DP-Master
- Direkter Datenaustausch DP-Slave > I-Slave

Direkter Datenaustausch DP-Slave > DP-Master

Bei der Konfiguration mit Multi-Master-System können Eingangsdaten von DP-Slaves von DP-Mastern am gleichen physikalischen PROFIBUS-DP Subnetz gelesen werden.

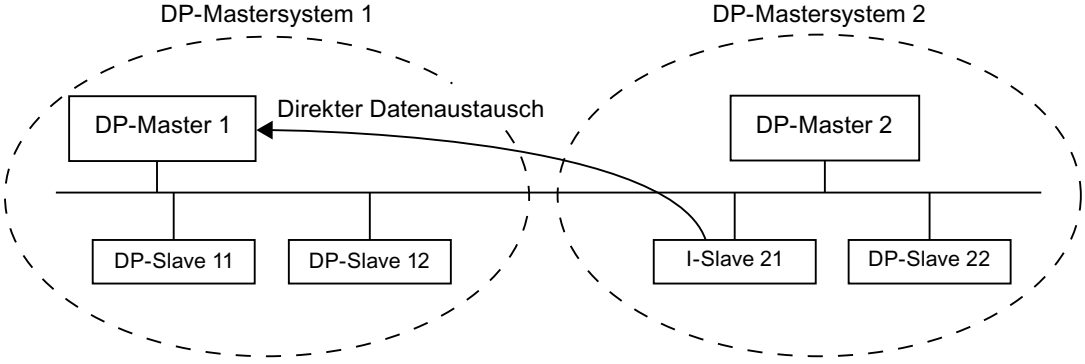

Direkter Datenaustausch DP-Slave > I-Slave

Ein intelligenter DP-Slave (I-Slave), wie z. B. eine CPU 315-2 DP, kann im Multi-Master-System Eingangsdaten von DP-Slaves auch von verschiedenen DP-Mastersystemen direkt auf seinen Eingangsdatenbereich übertragen lassen.

Prinzipiell stellen DP-Slaves ab einem bestimmten Ausgabestand ausgewählte Eingangsdaten für den Direkten Datenaustausch zur Verfügung. Diese Eingangsdaten können wiederum von intelligenten DP-Slaves weiter verwendet werden.

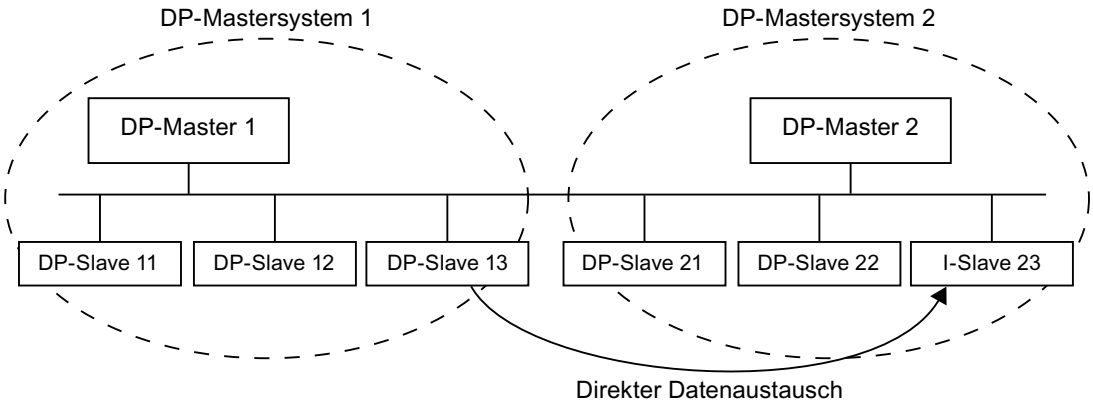

Siehe auch

Wissenswertes zum Direkten Datenaustausch (Seite 2914)

Direkten Datenaustausch konfigurieren (Seite 2915)

Konfigurationen mit intelligenten DP-Slaves

Definition

DP-Slaves mit eigenem Programm zur Vorverarbeitung werden als intelligente DP-Slaves (I-Slaves) bezeichnet. Beispiele für intelligente DP-Slaves sind:

- CPU 315-2 DP
- CPU 317-2 DP
- CPU 319-3 PN/DP

Konfigurationen mit intelligenten DP-Slaves

Über den PROFIBUS-DP können intelligente DP-Slaves auf zwei Arten konfiguriert werden:

- Datenaustausch I-Slave <> DP-Master
- Direkter Datenaustausch DP-Slave > I-Slave

Datenaustausch I-Slave <> DP-Master

Ein überlagertes Automatisierungssystem behandelt die in Teilaufgaben zerlegte Automatisierungsaufgabe. Die zu erledigenden Steuerungsaufgaben werden eigenständig und effizient als Programme der Vorverarbeitung in einer CPU abgearbeitet. Diese CPU kann in Form eines intelligenten DP-Slaves realisiert werden.

Bei Konfigurationen mit intelligenten DP-Slaves, wie z. B. einer CPU 315-2 DP, greift der DP-Master nicht auf die E/A-Baugruppen des intelligenten DP-Slaves zu, sondern nur auf den Operandenbereich der CPU des I-Slaves. Der Operandenbereich darf nicht für reale E/A-Baugruppen im I-Slave belegt werden. Diese Zuordnung muss bei der Projektierung des I-Slaves erfolgen.

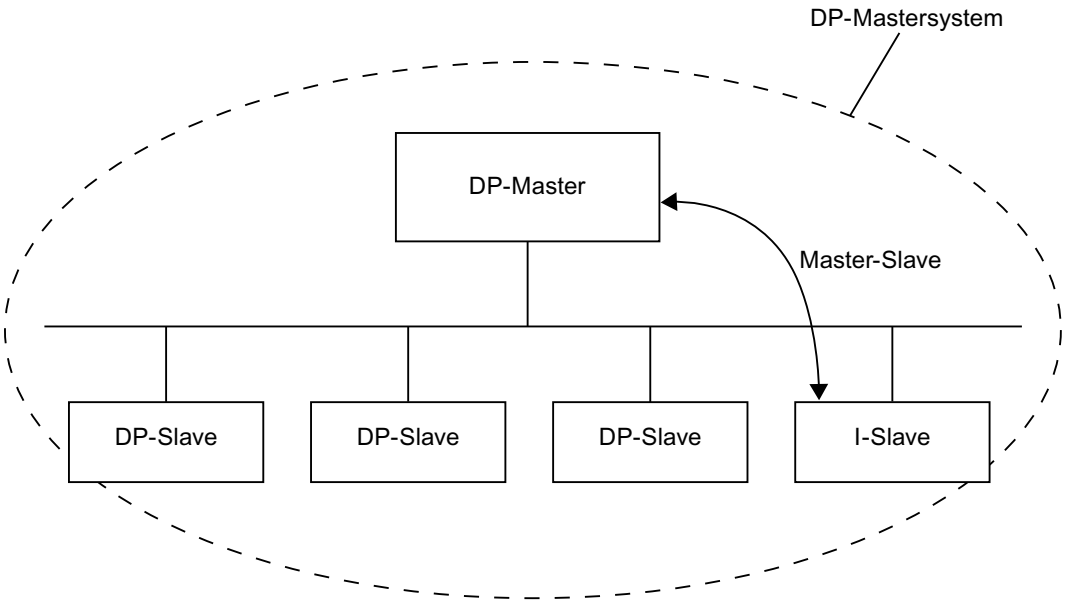

Direkter Datenaustausch DP-Slave > I-Slave

Bei dieser Konfiguration können Eingangsdaten von einfachen DP-Slaves sehr schnell zu intelligenten DP-Slaves am PROFIBUS-DP Subnetz übermittelt werden.

Dabei können prinzipiell alle einfachen DP-Slaves (ab einem bestimmten Ausgabestand) und auch intelligente DP-Slaves, ihre Eingangsdaten für den Direkten Datenaustausch zur Verfügung stellen. Als Empfänger dieser Daten können intelligente DP-Slaves verwendet werden.

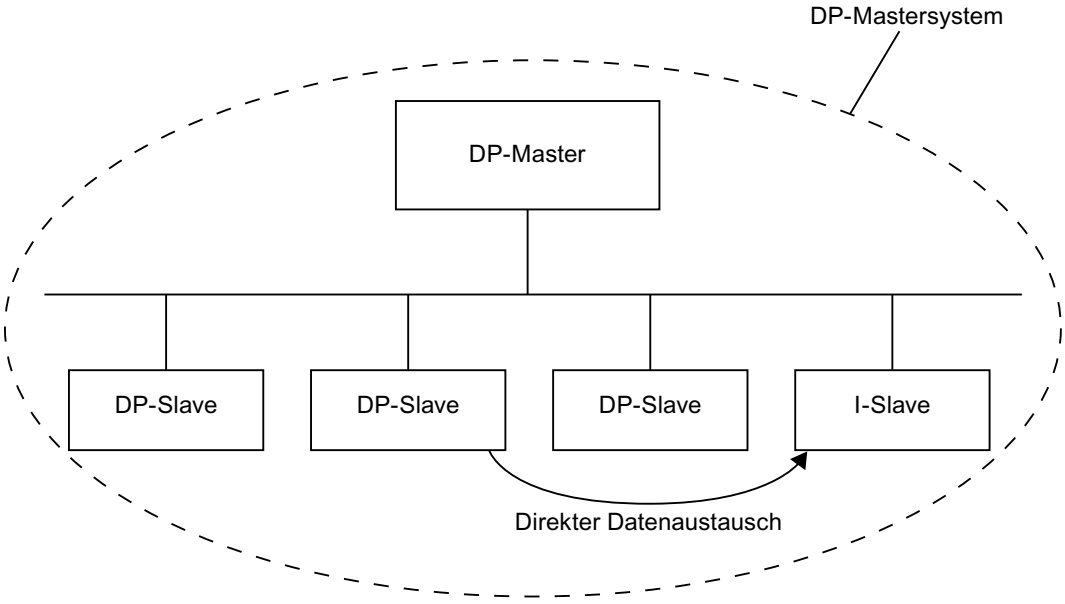

Siehe auch

I-Slave in ein DP-Mastersystem einfügen (Seite 2901)

Beispiel für das Konfigurieren einer ET 200 CPU als I-Slave (Seite 2909)

Wissenswertes zum Direkten Datenaustausch (Seite 2914)

1.4.9.6 Dezentrale Peripheriesysteme konfigurieren

DP-Mastersystem anlegen

Einführung

Um ein PROFIBUS DP-Mastersystem anzulegen, benötigen Sie einen DP-Master und wenigstens einen DP-Slave. Sobald Sie einen DP-Master mit einem DP-Slave über ihre PROFIBUS DP-Schnittstellen verbinden, erfolgt eine Master-Slave-Kopplung.

DP-Master und DP-Slave

Als DP-Master können Sie folgende Geräte mit DP-Schnittstelle einsetzen:

- CPU mit fest integrierter oder steckbarer DP-Master-Schnittstelle
- CP in Verbindung mit einer CPU
- Schnittstellenmodul, das einer CPU/FM zugeordnet ist
- Anschaltungsbaugruppe mit DP-Master-Schnittstelle

Als DP-Slave können Sie Kopfmodule der Dezentralen Peripherie mit PROFIBUS DP-Schnittstelle oder CPUs als intelligente DP-Slaves einsetzen.

Bei einigen Geräten können Sie im Inspektorfenster zur DP-Schnittstelle unter "Betriebsart" zwischen den möglichen Betriebsarten (wie "DP-Master" oder "DP-Slave") wählen. Selektieren Sie das Optionsfeld mit der benötigten Betriebsart.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

Um ein DP-Mastersystem beispielsweise mit einer CPU 319-3 PN/DP anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie als potenziellen DP-Master eine CPU 319-3 PN/DP aus dem Hardware-Katalog.
2. Ziehen Sie per Drag & Drop die CPU auf die freie Fläche in der Netzsicht.

3. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle der CPU.
4. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".

Es wird ein DP-Mastersystem mit einer CPU 319-3 PN/DP als DP-Master als einzigem Teilnehmer angelegt.

Wenn Sie die DP-Schnittstelle eines DP-Slaves mit der DP-Schnittstelle des DP-Masters verbinden, wird der DP-Slave automatisch in das DP-Mastersystem aufgenommen. Wenn zwischen DP-Master und DP-Slave noch kein Subnetz existiert, wird ein neues Subnetz zwischen DP-Master und DP-Slave erzeugt.

Um beispielsweise eine IM 155-6 DP HF als DP-Slave in das DP-Mastersystem aufzunehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die DP-Schnittstelle des DP-Masters oder DP-Slaves.
2. Ziehen Sie mit gedrückter Maustaste eine Verbindung von der selektierten DP-Schnittstelle auf die DP-Schnittstelle des gewünschten Kommunikationspartners.

oder

1. Klicken Sie auf den Hyperlink des DP-Slaves IM 155-6 DP HF.
2. Wählen Sie aus der angezeigten Liste möglicher DP-Master den gewünschten DP-Master aus.

Der DP-Slave IM 155-6 DP HF wird in das DP-Mastersystem mit der CPU 319-3 PN/DP als DP-Master aufgenommen.

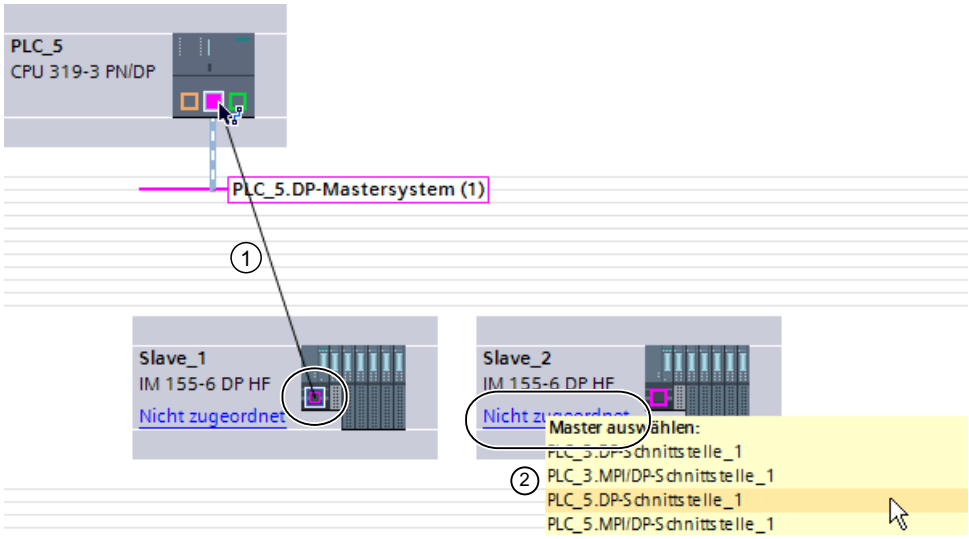

① Drag & Drop zwischen den DP-Schnittstellen von DP-Slave zu DP-Master.

② Klick auf den Link bei nicht zugeordnetem DP-Slave öffnet eine Auswahl möglicher DP-Master.

Passen Sie gegebenenfalls die Eigenschaften des PROFIBUS-Subnetzes oder des DP-Masters (z. B. PROFIBUS-Adresse) im Inspektorfenster unter "Eigenschaften" an.

Hinweis

Wenn Sie mit der Maus eine Verbindung zwischen der MPI-Schnittstelle eines DP-Masters und der PROFIBUS-Schnittstelle eines DP-Slaves ziehen, wird die MPI-Schnittstelle des DP-Masters automatisch als PROFIBUS-Schnittstelle konfiguriert und ein DP-Mastersystem angelegt.

Anzeige des DP-Masters im DP-Slave

Wenn ein DP-Slave mit einem DP-Master verbunden ist, wird der Name des DP-Masters als Hyperlink am DP-Slave angezeigt. Wenn Sie auf den Hyperlink klicken, wird der zugeordnete DP-Master selektiert.

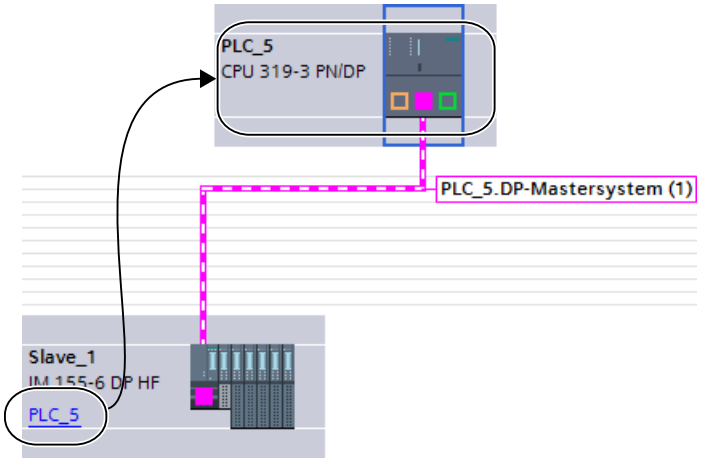

Hervorheben des DP-Mastersystems

Wenn Sie ein neues DP-Mastersystem angelegt haben, wird das DP-Mastersystem hervorgehoben dargestellt. Dadurch erkennen Sie schnell, welche Geräte zu dem DP-Mastersystem gehören. Sie können ein DP-Mastersystem auch hervorheben, indem Sie den Mauszeiger über ein Subnetz ziehen. Es werden daraufhin die Namen der vorhandenen DP-Mastersysteme angezeigt. Wenn Sie mit der Maus auf eines der angezeigten DP-Mastersysteme klicken, wird das entsprechende DP-Mastersystem hervorgehoben.

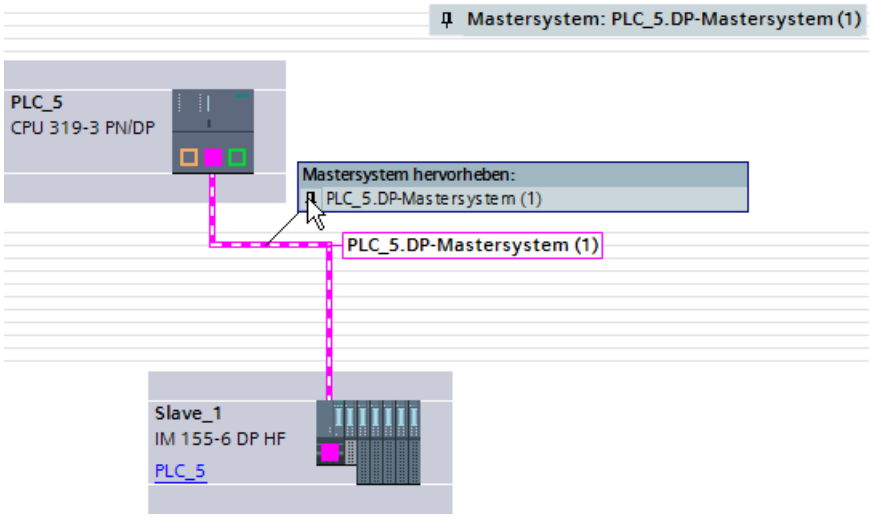

Um die Hervorhebung eines DP-Mastersystems wieder auszublenden, stehen Ihnen mehrere Möglichkeiten zu Verfügung:

- Heben Sie ein anderes DP-Mastersystem hervor.
- Klicken Sie auf die Pinnadel mit der Bezeichnung des DP-Mastersystems in der rechten oberen Ecke der Netzsicht.

Siehe auch

DP/DP-Koppler im Hardware-Katalog (Seite 2873)

Grundlagen zum Konfigurieren eines DP-Mastersystems (Seite 2871)

DP-Mastersysteme und -Schnittstellen bearbeiten

Einführung

Nachdem Sie ein DP-Mastersystem angelegt haben, haben Sie auch die Möglichkeit, das DP-Mastersystem und seine Komponenten zu trennen. Daraus können sich Subnetze mit DP-Slaves und ohne DP-Master ergeben.

Das Bearbeiten der Schnittstellen eines DP-Masters ist meist nicht notwendig.

Am DP-Mastersystem können Sie den Namen und die Nummer ändern.

Trennung von DP-Mastersystemen

Wenn Sie eine CPU mit integrierter PROFIBUS-DP-Schnittstelle oder einen PROFIBUS-CP (als intelligenter DP-Slave projektierbar) als DP-Master mit Mastersystem projektiert haben, dann können Sie das DP-Mastersystem vom DP-Master trennen. Das Gerät ist dann nicht mehr an das DP-Mastersystem angeschlossen.

Das Trennen des Subnetzes bei einem DP-Master hat zur Folge, dass das Mastersystem nicht mehr existiert, da es keinem DP-Master mehr zugeordnet ist. Die einzelnen DP-Slaves sind aber weiterhin miteinander über das Subnetz verbunden und ein projektierter direkter Datenaustausch bleibt erhalten.

Wenn Sie die DP-Slaves löschen oder vom Mastersystem trennen, bleibt das Mastersystem am DP-Master erhalten.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Es besteht ein DP-Mastersystem mit einem DP-Master und mindestens einem DP-Slave.

DP-Master vom DP-Mastersystem trennen

Um das DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Vom Mastersystem trennen".

Der selektierte DP-Master wird vom DP-Mastersystem getrennt. Es bleibt ein Subnetz mit den DP-Slaves zurück.

DP-Master dem DP-Mastersystem hinzufügen

Um einem Subnetz wieder einen DP-Master zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle eines DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".
3. Ziehen Sie das neue DP-Mastersystem auf die DP-Schnittstellen der DP-Slaves.

Der DP-Master bildet mit den DP-Slaves wieder ein DP-Mastersystem.

PROFIBUS-DP-Schnittstelle online-fähig machen

Um die integrierte MPI/DP-Schnittstelle einer CPU die aktive Teilnahme am PROFIBUS-DP zu ermöglichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Schnittstellensymbol der CPU in der Geräte- oder Netzsicht.
2. Wählen Sie im Inspektorfenster unter "MPI-Adresse" den Parameter "Schnittstellentyp: PROFIBUS" aus.

3. Ändern Sie bei Bedarf die vorgeschlagene PROFIBUS-Adresse.
4. Wählen Sie bei Bedarf und soweit vorhanden unter "Subnetz" ein schon bestehendes Subnetz aus.

Hinweis

Wenn Sie mit der Maus eine Verbindung von der PROFIBUS-Schnittstelle eines DP-Slaves zu der MPI-Schnittstelle eines DP-Masters ziehen, wird die MPI-Schnittstelle des DP-Masters automatisch als PROFIBUS-Schnittstelle konfiguriert und ein DP-Mastersystem angelegt.

Eigenschaften eines DP-Mastersystems bearbeiten

Um die Eigenschaften eines DP-Mastersystems zu bearbeiten, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger auf ein Subnetz mit DP-Mastersystem.
2. Klicken Sie in der erscheinenden Meldung über die vorhandenen DP-Mastersysteme auf das DP-Mastersystem, welches Sie bearbeiten wollen. Das DP-Mastersystem wird nun farblich hervorgehoben.
3. Klicken Sie nun auf das farblich hervorgehobene DP-Mastersystem.
4. Bearbeiten Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Attribute des DP-Mastersystems.

Hinweis

Wenn Sie ein Subnetz anklicken, ohne dass ein DP-Mastersystem hervorgehoben ist, können Sie im Inspektorfenster unter "Eigenschaften" die Eigenschaften des ganzen Subnetzes bearbeiten.

DP-Slaves am Mastersystem einfügen und konfigurieren

In der Netzsicht fügen Sie verschiedene DP-Slaves direkt per Drag & Drop oder Doppelklick aus dem Hardware-Katalog ein.

Typen von DP-Slaves

Bei der Projektierung von DP-Slaves werden diese unterschieden nach:

- Kompakte DP-Slaves
  (Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L)
- Modulare DP-Slaves
  (Anschaltungsbaugruppen mit zugeordneten S5- oder S7-Baugruppen, z. B. ET 200M)
- Intelligente DP-Slaves (I-Slaves)
  (SIMATIC S7-300 mit z. B. CP 342-5, CPU 315-2 DP oder ET 200S mit IM 151-7 CPU)

Steckplatzregeln

- In einem DP-Mastersystem verwenden Sie nur einen DP-Master und ein oder mehrere DP-Slaves.
- Sie dürfen nicht mehr DP-Slaves in einem DP-Mastersystem haben, als im DP-Master angegeben.

Hinweis

Beachten Sie bei der Konfigurierung des DP-Mastersystems die technischen Daten des DP-Masters (max. Anzahl Teilnehmer, max. Anzahl Steckplätze, max. Anzahl Nutzdaten). Möglicherweise können Sie aufgrund der Begrenzung durch Steckplätze oder Nutzdaten nicht die maximale Anzahl von Teilnehmern verwenden!

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Mastersystem ist angelegt.

DP-Slave am DP-Mastersystem einfügen

Um einen DP-Slave aus dem Hardware-Katalog am DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie einen modularen DP-Slave (z. B. ET 200M) aus dem Hardware-Katalog.
2. Ziehen Sie den DP-Slave per Drag & Drop vom Hardware-Katalog in die Netzsicht.
3. Ziehen Sie eine Verbindung von der DP-Schnittstelle des DP-Masters oder von dem hervorgehobenen DP-Mastersystem zu der DP-Schnittstelle des neuen DP-Slaves.

Es wird automatisch ein DP-Mastersystem erstellt und der DP-Slave wird automatisch als DP-Slave mit dem DP-Master verbunden.

Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog einen Doppelklick auf den gewünschten DP-Slave durchführen. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

DP-Slave vom DP-Mastersystem trennen

Um einen DP-Slave vom DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb der Netzsicht mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slaves.
2. Wählen Sie aus dem Kontextmenü die Art der Trennung vom DP-Mastersystem:
   - "Vom Subnetz trennen": Die PROFIBUS-Verbindung wird getrennt und das Gerät ist nicht mehr an das DP-Mastersystem oder ein Subnetz angeschlossen.
   - "Vom Mastersystem trennen": Der DP-Slave ist weiterhin an das Subnetz angeschlossen, aber nicht mehr dem DP-Mastersystem als DP-Slave zugeordnet.

Der selektierte DP-Slave wird vom DP-Mastersystem getrennt.

Sie können die Trennung auch direkt über das Kontextmenü des DP-Slaves durchführen:

1. Klicken Sie innerhalb der Netzsicht mit der rechten Maustaste auf den DP-Slave.
2. Wählen Sie aus dem Kontextmenü "Vom DP-Master/IO-System trennen".

Der DP-Slave ist weiterhin an das Subnetz angeschlossen, aber nicht mehr dem DP-Mastersystem als DP-Slave zugeordnet.:

DP-Slave einem neuen DP-Mastersystem zuweisen

Um einen vorhandenen DP-Slave einem neuen DP-Mastersystem zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slave.
2. Wählen Sie im Kontextmenü "Neuem Master zuweisen".
   Es ist dabei unerheblich, ob der DP-Slave bereits einem anderen DP-Mastersystem zugeordnet ist.
3. Wählen Sie aus der Auswahlliste den DP-Master aus, an dessen DP-Mastersystem Sie den DP-Slave anbinden wollen.
   Der selektierte DP-Slave wird einem neuen DP-Mastersystem zugewiesen.

Analog dazu können Sie einen DP-Slave über die Funktion "Neuem Subnetz zuweisen" auch an ein anderes Subnetz anbinden, ohne ihn dadurch an ein bestehendes DP-Mastersystem zu koppeln.

Sie können die Zuweisung auch direkt über das Kontextmenü des DP-Slaves durchführen:

1. Klicken Sie mit der rechten Maustaste auf den DP-Slave.
2. Wählen Sie im Kontextmenü "Neuem DP-Master/IO-Controller zuweisen".
3. Wählen Sie den DP-Master aus der Liste vorhandenener DP-Master aus.

Der selektierte DP-Slave wurde einem neuen DP-Mastersystem zugewiesen.

Hinweis

Wenn sich an einem Subnetz nur ein einzelner DP-Master befindet, reicht es auch aus, wenn Sie per Drag & Drop die Schnittstelle des DP-Slaves mit dem Subnetz dieses DP-Masters verbinden. Der DP-Slave wird dann sofort mit dem Subnetz des DP-Masters verbunden und der DP-Slave wird diesem DP-Master zugewiesen.

DP-Slave konfigurieren

Um einen DP-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wechseln Sie in die Gerätesicht des DP-Slaves.
2. Wählen Sie die gewünschten Baugruppen aus dem Hardware-Katalog aus. Bei modularen DP-Slaves sind die möglichen Baugruppen im Hardware-Katalog unterhalb der entsprechenden DP-Slave-"Familie" angeordnet. Nutzen Sie dabei die Filterfunktion des Hardware-Katalogs.

3. Ziehen Sie die gewünschten Baugruppen aus dem Hardware-Katalog auf den Baugruppenträger des DP-Slave.
4. Konfigurieren Sie den DP-Slave und die gesteckten Baugruppen im Inspektorfenster.

Siehe auch

DP-Mastersystem anlegen (Seite 2878)

DP-Slaves im Hardware-Katalog (Seite 2873)

Tipp: Schnelles Konfigurieren von Mastersystemen

Wenn das DP-Mastersystem viele DP-Slaves hat, weisen Sie alle per Drag & Drop platzierten DP-Slaves in einem Schritt einem Master zu.

Voraussetzung

DP-Master und DP-Slaves sind in der Netzsicht platziert.

DP-Slaves einem DP-Mastersystem zuweisen

Gehen Sie folgendermaßen vor:

1. Wählen Sie einen passenden Zoomfaktor, so dass Sie möglichst viele DP-Slaves in der Netzsicht sehen.
2. Ordnen Sie die DP-Slaves in maximal zwei Reihen an.
3. Wählen Sie mit dem Mauszeiger alle DP-Schnittstellen aus (nicht die Geräte!). Das gelingt nur, wenn Sie außerhalb des ersten DP-Slaves mit dem Ziehen des Mauszeigers beginnen und beim letzten DP-Slave die Maustaste loslassen (Auswahl mit dem Lasso).

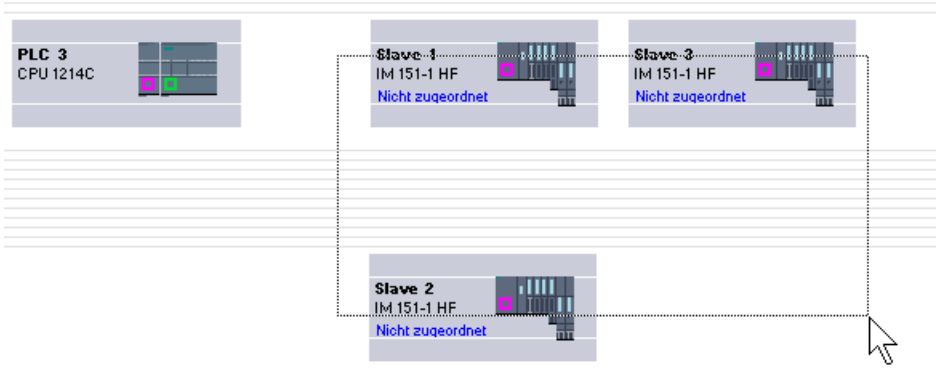

4. Wählen Sie das Kontextmenü "Neuem Master zuweisen" und markieren Sie im Folgedialog die entsprechende DP-Schnittstelle des DP-Masters.

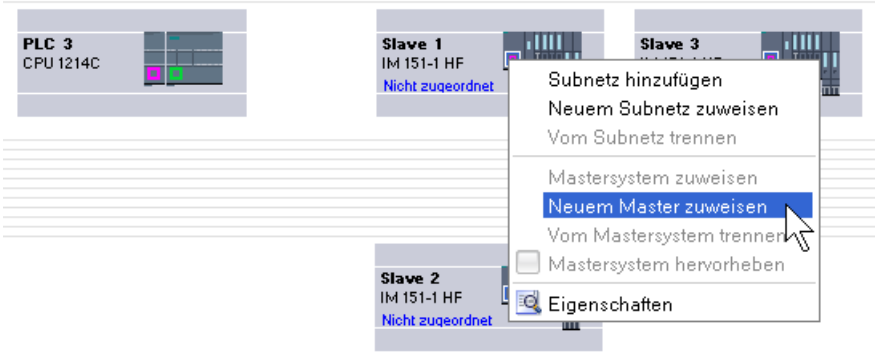

5. Die DP-Slaves werden automatisch mit dem DP-Master vernetzt und bilden mit ihm zusammen ein DP-Mastersystem.

Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog auf einen DP-Slave doppelklicken und damit schnell weitere DP-Slaves hinzufügen. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

DP-Slave SYNC/FREEZE-Gruppe zuordnen

Einführung

Ein DP-Master mit entsprechender Funktionalität kann an eine Gruppe von DP-Slaves gleichzeitig die Steuerkommandos SYNC und/oder FREEZE zur Synchronisation der DP-Slaves senden. Sie müssen dazu die DP-Slaves SYNC/FREEZE-Gruppen zuordnen.

Voraussetzung

Ein DP-Mastersystem muss angelegt sein.

Vorgehen

Um einen DP-Slave einer SYNC/FREEZE-Gruppe zuzuordnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die Schnittstelle des DP-Slaves, den Sie einer Gruppe zuordnen wollen.
2. Wählen Sie im Inspektorfenster unter der Gruppe "SYNC/FREEZE" die Optionskästchen für die gewünschten SYNC/FREEZE-Gruppen aus.

Hinweis

Jeden DP-Slave können Sie maximal einer SYNC/FREEZE-Gruppe zuordnen.

Ausnahme: Wenn Sie als DP-Master einen CP 342-5 einsetzen, dann können Sie jedem zugeordneten DP-Slave maximal acht SYNC/FREEZE-Gruppen zuweisen.

Beachten Sie dazu auch die Dokumentation zum CP 342-5.

Wissenswertes zu den Steuerkommandos SYNC und FREEZE

Mit den Steuerkommandos SYNC und FREEZE im DP-Master ist es möglich, die DP-Slaves ereignisgesteuert zu synchronisieren. Die Steuerkommandos sendet der DP-Master gleichzeitig an eine Gruppe von DP-Slaves seines Mastersystems. Nicht berücksichtigt werden DP-Slaves, die ausgefallen sind oder gerade Diagnose melden.

Voraussetzung für das Synchronisieren über Steuerkommandos ist, dass Sie die DP-Slaves den SYNC/FREEZE-Gruppen zugeordnet haben.

Für eine S7-CPU verwenden Sie die Anweisung DPSYC_FR (SFC 11) zur Synchronisation der DP-Slaves.

Wenn Sie den DP-Master selektieren, sehen Sie im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle > SYNC/FREEZE" oder "Eigenschaften > MPI/DP-Schnittstelle > SYNC/FREEZE" (bei Betrieb als PROFIBUS-Schnittstelle) eine Liste der acht SYNC/FREEZE-Gruppen.

Hinweis

Es gibt DP-Slaves, die nur SYNC-fähig oder nur FREEZE-fähig sind. Welche dieser Funktionen ein DP-Slave unterstützt, lesen Sie in den Eigenschaften der DP-Schnittstelle, Bereich "SYNC/FREEZE".

Um solch einem DP-Slave einer SYNC-Gruppe bzw. einer FREEZE-Gruppe zuordnen zu können, müssen Sie in den Eigenschaften der DP-Master-Schnittstelle die betreffende Gruppe auf die Funktion einschränken, die der DP-Slave unterstützt..

Beispiel: Bei einem DP-Slave, der nur FREEZE-fähig ist, müssen Sie das Optionskästchen "SYNC" in der SYNC/FREEZE-Gruppe des DP-Masters deaktivieren.

Steuerkommando SYNC

Mit dem Steuerkommando SYNC veranlasst der DP-Master eine Gruppe von DP-Slaves, die Zustände ihrer Ausgänge auf den momentanen Wert einzufrieren.

Bei den folgenden Telegrammen speichern die DP-Slave die Ausgangsdaten des DP-Masters. Die Zustände der Ausgänge der DP-Slaves bleiben aber unverändert.

Nach jedem neuen Steuerkommando SYNC setzt der DP-Slave seine Ausgänge auf die Werte, die er als Ausgangsdaten des DP-Masters gespeichert hat.

Die Ausgänge werden erst dann wieder zyklisch aktualisiert, wenn der DP-Master das Steuerkommando UNSYNC sendet.

Steuerkommando FREEZE

Nach Erhalt des Steuerkommandos FREEZE vom DP-Master frieren die DP-Slaves einer Gruppe den aktuellen Zustand ihrer Eingänge ein. Bei den folgenden zyklischen Telegrammen senden die DP-Slaves diese eingefrorenen Eingangsdaten an den DP-Master.

Nach jedem neuen Steuerkommando FREEZE frieren die DP-Slaves erneut den aktuellen Zustand ihrer Eingänge ein.

Der Zustand der DP-Slave-Eingänge wird erst dann wieder zyklisch an den DP-Master gesendet, wenn der DP-Master das Steuerkommando UNFREEZE sendet.

Siehe auch

DP-Mastersystem anlegen (Seite 2878)

PROFIBUS-Eigenschaften einstellen (Seite 2948)

1.4.9.7 Beispiel: ET 200S als DP-Slave konfigurieren

ET 200S konfigurieren

Einführung

DP-Slaves und IO-Devices der Familie ET 200S werden konfiguriert wie andere modulare DP-Slaves und IO-Devices. Die Module für ET 200S finden Sie im Hardware-Katalog unter "Dezentrale Peripherie".

Steckplatzregeln für die Konfiguration einer ET 200S

Für das Konfigurieren der ET 200S gelten die folgende Regeln:

- Stecken Sie die Module der ET 200S lückenlos nebeneinander.
- Steckplatz 1: nur für Powermodule PM-E oder PM-D
- Links neben einem Elektronikmodul (EM): Nur ein EM oder ein Powermodul (PM-E oder PM-D)
- Links neben einem Motorstarter (MS): Nur ein MS oder ein Powermodul PM-D, PM-D Fx (1..x..4) oder PM-X

- Links neben einem PM-X: Nur ein Motorstarter oder ein PM-D
- Erlaubt sind maximal 63 Module und ein Interfacemodul IM

Hinweis

Beachten Sie die korrekte Zuordnung von PM-E-Spannungsbereich und EM-Spannungsbereich!

Betrieb bei Sollausbau ungleich Istausbau

Dieser Parameter steuert das Verhalten im Hochlauf und beim Ziehen und Stecken von Modulen während des Betriebs (zyklischer Datenaustausch).

Optionskästchen ist aktiviert (Betrieb bei Sollausbau ungleich Istausbau ist erlaubt):

- Die Baugruppe läuft auch dann an, wenn der Sollausbau ungleich dem Istausbau ist
- Die Baugruppe bleibt nach Ziehen bzw. Stecken im zyklischen Datenaustausch (kein Stationsausfall).

Optionskästchen ist deaktiviert (Betrieb bei Sollausbau ungleich Istausbau ist verboten):

- Die Baugruppe läuft nicht an (geht nicht in den zyklischen Datenaustausch über), wenn der Sollausbau ungleich dem Istausbau ist
- Wenn Module im Betrieb gezogen bzw. gesteckt werden, dann führt das zu einem Stationsausfall und die Baugruppe nimmt nicht mehr am zyklischen Datenaustausch teil.

Beachten Sie die Bedingungen im Handbuch der jeweiligen Baugruppe, welche Randbedingungen zu einem Stationsausfall führen oder den Eintritt in den Datenaustausch verhindern, auch wenn der Betrieb bei Sollausbau ungleich Istausbau freigegeben ist!

Störfrequenzunterdrückung

Die Frequenz Ihres Wechselspannungsnetzes kann sich insbesondere bei der Messung in kleinen Spannungsbereichen und bei Thermoelementen störend auf den Messwert auswirken. Für die gesamte ET 200S stellen Sie die Frequenz der Netzspannung ein, die in Ihrer Anlage vorherrscht.

Die analogen Elektronikmodule AI stellen abhängig von der Auswahl der Störfrequenzunterdrückung ihre Integrationszeit bzw. Wandlungszeit ein.

Vergleichsstelle parametrieren

Eine Vergleichsstelle ist der Anschluss eines Thermoelements an eine Zuleitung (i. d. R. im Klemmenkasten). Die hier durch Temperatureinfluss auftretende Spannung verfälscht den Temperaturwert, den das Modul misst.

Bei ET 200S kann ein Kanal des Moduls AI RTD als Vergleichsstelle parametriert werden. Mit der von diesem Modul gemessenen Temperatur an der Vergleichsstelle können andere AI TC-Module ihre Messwerte kompensieren.

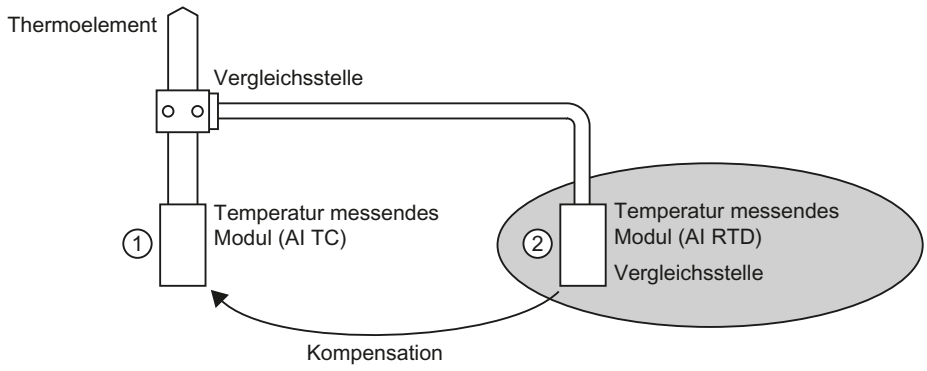

① Parametrierung des AI TC:
→ Auswahl der verwendeten Vergleichsstelle

② Parametrierung des AI RTD:
→ Aktivierung der Vergleichstelle
→ Festlegung Steckplatz und Kanal des AI RTD

Besonderheiten bei der Parametrierung von Vergleichsstellen

Die Parametrierung von Vergleichsstellen wird Ihnen exemplarisch an der Verwendung eines Widerstandsthermometers Pt 100 "Klimabereich" zum Erfassen der Vergleichsstellentemperatur gezeigt.

Um die Vergleichsstelle zu parametrieren, gehen Sie folgendermaßen vor:

1. Platzieren Sie in der Gerätesicht der ET 200S ein analoges RTD-Elektronikmodul, z. B. 2AI RTD HF. Das Modul muss die Messart Thermowiderstand (linear, 4-Leiter) und den Messbereich Pt 100 Klimabereich unterstützen.
2. Selektieren Sie die Baugruppe im Baugruppenträger.
3. Stellen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Eingänge" einen Kanal für die Vergleichsstellen-Funktion ein:
   Messart: Thermowiderstand (linear, 4-Leiter)
   Messbereich: Pt 100 Klimabereich
4. Selektieren Sie die ET 200S (d. h. das Interfacemodul).
5. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Baugruppenparameter > Vergleichsstelle" das Optionskästchen "Vergleichsstelle" und geben Sie den Steckplatz und die Kanalnummer der relevanten RTD-Baugruppe an.
6. Platzieren Sie das analoge Elektronikmodul für Temperaturmessung mittels Thermoelement (TC-Modul) und wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Eingänge > Kanal x" als Vergleichsstelle die Option "RTD".

Weitere Informationen

Beachten Sie für weitere Informationen zu den verschiedenen Typen und Einsatzmöglichkeiten von Modulen in ET 200S die Betriebsanleitung und das Gerätehandbuch "Dezentrales Peripheriesystem ET 200S".

Beachten Sie für weitere Informationen zur Analogwertverarbeitung die Dokumentation zum Dezentralen Peripheriesystem ET 200S.

Siehe auch

ET 200S im DPV1-Modus konfigurieren (Seite 2899)

Dokumentation zu ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)

Adressen packen

Einführung

DP-Slaves und IO-Devices der Familie ET 200S werden konfiguriert wie andere modulare DP-Slaves und IO-Devices. Zusätzlich zu den üblichen Funktionen modularer DP-Slaves und IO-Devices gibt es bei ET 200S die Funktion "Adressen packen":

Digitale Elektronikmodule mit einem benötigten Adressraum von 2 oder 4 Bit belegen beim Einfügen in der Gerätesicht zunächst pauschal 1 Byte. Der wirklich belegte Adressraum kann aber nach dem Konfigurieren durch die Funktion "Adressen packen" zusammengeschoben werden.

	Vorbelegung	Nach "Adressen packen"
Modul	E-Adresse	E-Adresse
2DI (2 Bit)	Byte 10	10.0...10.1
4DI (4 Bit)	Byte 11	10.2...10.5

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Eine ET 200S, z. B. IM 151-1, ist vorhanden.
- Ein paar digitale Elektronikmodule, z. B. 2DI AC120V ST befinden sich in den Steckplätzen.

Adressen packen

Um Adressen zu packen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Elektronikmodule, deren Adressen gepackt werden sollen. Für das Selektieren von mehreren Elektronikmodulen haben Sie folgende Möglichkeiten:
   – Klicken Sie die Elektronikmodule bei gleichzeitigem Drücken von <Shift> oder <Strg> einzeln an.
   – Klicken Sie außerhalb des Baugruppenträgers und ziehen Sie die Maus um die zu selektierenden Elektronikmodule.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen packen".

Es werden die Adressbereiche für Eingänge, Ausgänge und Motorstarter getrennt gepackt. In den Spalten E-Adresse und A-Adresse der Geräteübersicht sehen Sie die gepackten Adressen.

Adressbildung und Struktur bei gepackten Adressen

Wenn Sie die Funktion "Adressen packen" verwenden, werden die Adressen der selektierten Elektronikmodule nach folgenden Regeln gepackt:

- Der Anfang des Adressbereichs ist durch die niedrigste Adresse der selektierten Elektronikmodule festgelegt: X.0.
- Wenn die Bitadresse nicht "0" ist, dann wird automatisch die nächste (freie) Byteadresse herangezogen, ab welcher der markierte Bereich eingeschoben werden kann: (X+n).0.
- Wenn kein zusammenhängender Bereich mehr vorhanden ist, wird automatisch in vorhandene Adresslücken gepackt.

Elektronikmodule mit gepackten Adressen und derselben Byteadresse bilden eine Packgruppe.

Adressen entpacken

Um Adressen zu entpacken, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Elektronikmodule mit gepackter Adresse.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen entpacken".

Packgruppen der selektierten Elektronikmodule werden aufgelöst und die gepackten Adressen der entsprechenden Elektronikmodule werden entpackt.

Wenn Sie aus einer Packgruppe Elektronikmodule löschen oder verschieben oder auf einen freien Steckplatz innerhalb einer Packgruppe Elektronikmodule einfügen, wird die Packgruppe ebenfalls aufgelöst und die gepackten Adressen werden entpackt.

Die Anfangsadressen der entpackten Elektronikmodule werden auf die jeweils nächsten freien Byte-Adressen gelegt.

Besonderheiten bei Elektronikmodulen mit gepackten Adressen

Für ein Elektronikmodul mit einer gepackten Adresse gelten folgende Besonderheiten:

- Aus CPU-Sicht ist keine Steckplatzzuordnung für das Elektronikmodul möglich. Aus diesem Grund gibt die Anweisung GADR_LGC (SFC 5) für den tatsächlichen Steckplatz des Elektronikmoduls die Fehlerinformation W#16#8099 "Steckplatz nicht projektiert" aus.
- Die Anweisung LGC_GADR (SFC 49) und die SZL-ID W#16#xy91 "Baugruppenzustandsinformation" für ein Elektronikmodul können nicht ausgewertet werden.
- Das Elektronikmodul erhält eine zusätzliche Diagnoseadresse über die DPV1-Funktionalität, weil aus CPU-Sicht Alarme aufgrund der gepackten Adresse sonst nicht zugeordnet werden können.
- Der "Ziehen/Stecken-Alarm" ist nicht möglich, da sich die Funktionen "Adressen packen" und "Ziehen/Stecken-Alarm" gegenseitig ausschließen.

Optionenhandling mit Reservemodulen konfigurieren

Über das Optionenhandling können Sie die ET 200S mit PROFIBUS-Schnittstelle für zukünftige Erweiterungen (Optionen) einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling mit Reservemodulen beschrieben.

Dazu montieren, verdrahten, projektieren und programmieren Sie den geplanten Maximalausbau der ET 200S und verwenden statt der erst später erforderlichen Elektronikmodule bei der Montage zunächst preisgünstige Reservemodule (138-4AA00 oder 138-4AA10).

Hinweis

Die ET 200S kann mit der Stammverkabelung vollständig vorverdrahtet werden, da ein Reservemodul keine Verbindung zu den Klemmen des Terminalmoduls und damit zum Prozess hat.

Voraussetzung

- ET 200S Interfacemodul
  - IM 151-1 STANDARD (ab 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (ab 6ES7 151-1AB02-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Aktivieren Sie nun die nummerierten Optionskästchen für die Steckplätze, die zunächst mit Reservemodulen statt mit Elektronikmodulen bestückt sind.
3. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).

Zu einem späteren Zeitpunkt können die montierten Reservemodule durch die projektierten Module ersetzt werden, ohne die Projektierung nachholen zu müssen.

Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1 STANDARD) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheriesystem ET 200S.

Funktionsweise des Optionenhandlings im Anlauf

Wenn "Anlauf bei Soll- ungleich Istausbau" gesperrt ist, dann läuft die ET 200S auch dann an, wenn statt des projektierten Elektronikmoduls ein Reservemodul steckt und das Optionenhandling für diesen Steckplatz eingeschaltet ist.

Funktionsweise des Optionenhandlings im Betrieb

Im Betrieb unterscheidet sich die Funktionsweise des Optionenhandlings wie folgt:

- Optionenhandling für einen Steckplatz eingeschaltet:
  Auf diesem Steckplatz darf sich das Reservemodul (Option) oder das projektierte Elektronikmodul befinden. Wenn sich ein anderes Modul auf diesem Steckplatz befindet, wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).
- Optionenhandling für einen Steckplatz ausgeschaltet:
  Auf diesem Steckplatz darf sich nur das projektierte Elektronikmodul befinden. Bei jedem anderen Modul wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).

Ersatzwerte des Reservemoduls

- Ersatzwert für digitale Eingänge: 0
- Ersatzwert für analoge Eingänge: 0x7FFF

Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 2 bis 63:

- Wert des Bits = 0: Es gilt die Parametrierung des Optionenhandlings. Reservemodule sind erlaubt.
- Wert des Bits = 1: Die Parametrierung des Optionenhandlings ist aufgehoben. Reservemodule werden auf diesem Steckplatz nicht akzeptiert:

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Auf dem Steckplatz befinden sich das Reservemodul, ein falsches Modul oder ein gezogenes Modul.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

Siehe auch

Welche Module unterstützen Optionenhandling? (http://support.automation.siemens.com/WW/view/de/22564754)

Optionenhandling ohne Reservemodule konfigurieren

Über das Optionenhandling können Sie die ET 200S für zukünftige Erweiterungen (Optionen) auch ohne die Montage von Reservemodulen einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling ohne Reservemodulen beschrieben.

Hinweis

ET 200S mit PROFINET-Schnittstelle

Diese Beschreibung bezieht sich auf ET 200S mit PROFIBUS-Schnittstelle. Das Optionenhandling für ET 200S mit PROFINET-Schnittstelle funktioniert prinzipiell so wie hier beschrieben ohne Reservemodule. Statt der hier aufgeführten DP-Interfacemodule sind PN-Interfacemodule zu verwenden. Weitere Informationen zum Optionenhandling bei ET 200S mit PROFINET-Schnittstelle finden Sie in den entsprechenden Gerätehandbüchern.

Voraussetzung

- ET 200S Interfacemodul
  - IM 151-1 HIGH FEATURE (ab 6ES7151-1BA02)
  - IM 151-1 STANDARD (ab 6ES7 151-1AA05-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).
3. Konfigurieren Sie den Maximalausbau des Slaves. Die Aktivierung/Deaktivierung von Optionen wird über das Anwenderprogramm gesteuert.

Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheriesystem ET 200S.

Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 1 bis 63:

- Wert des Bits = 0: Steckplatz ist in der Istkonfiguration nicht vorhanden.
- Wert des Bits = 1: Steckplatz ist in der Istkonfiguration vorhanden.

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Steckplatz gehört zu einer nicht vorhandenen Option oder Modulstatus ist nicht in Ordnung.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

Siehe auch

Beispielapplikation für ET 200S, Optionenhandling ohne Reservemodule (http://support.automation.siemens.com/WW/view/de/29430270)

ET 200S im DPV1-Modus konfigurieren

Mit PROFIBUS DPV1 stehen Ihnen erweiterte PROFIBUS-Funktionalitäten zur Verfügung.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Master mit DPV1-Funktionalität ist vorhanden.
- Eine Master-Slave-Verbindung mit PROFIBUS ist aufgebaut.

Vorgehen

Um den DP-Slave auf DPV1 zu schalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den DP-Slave.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP Alarm Mode" die Betriebsart "DPV1".

oder

1. Selektieren Sie den DP-Master.
2. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Verbindung zwischen dem DP-Master und dem gewünschten DP-Slave.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP Alarm Mode" die Betriebsart "DPV1".

Besonderheiten

Es gibt Abhängigkeiten der Parameter untereinander, die im Folgenden dargestellt sind:

Parameter	Betriebsart DPV0	Betriebsart DPV1
Betrieb bei Soll- ungleich Istausbau	Uneingeschränkt bedienbar	Uneingeschränkt bedienbar
Diagnosealarm (OB 82)	Nicht bedienbar, nicht gesetzt	Uneingeschränkt bedienbar
Prozessalarm (OB 40 bis 47)	Nicht bedienbar, nicht gesetzt	Uneingeschränkt bedienbar
Ziehen/Stecken-Alarm (OB 83)	Nicht bedienbar, nicht gesetzt	Nur bedienbar, wenn Adressen nicht gepackt sind. Wenn Ziehen/Stecken-Alarm aktiviert wird, dann wird automatisch "Anlauf bei Soll- ungleich Istausbau" aktiviert.

Alarme bei Modulen mit gepackten Adressen

Wenn das Modul Alarme auslösen kann und die Bitadresse wegen gepackter Adressen ungleich 0 ist, dann müssen Sie im Adressdialog der ET 200S eine Diagnoseadresse vergeben.

Die Diagnoseadresse ist notwendig für die Zuordnung eines DPV1-Alarms zum Modul als Alarmauslöser. Nur wenn ein Modul diese "ungepackte" Adresse hat, kann die CPU einen Alarm zuordnen und Informationen zum Alarm in der Startinformation des Alarm-OB bzw. im Diagnosepuffer hinterlegen. Eine "gepackte" Adresse kann die CPU hierfür nicht verwenden.

Aus Sicht der Alarmverarbeitung (Alarm-OB) hat das Modul dann die zugeordnete Diagnoseadresse, für die Verarbeitung der Ein- und Ausgangsdaten im Anwenderprogramm hat das Modul die gepackten Adressen!

Hinweis

Wenn die Adressen des Moduls gepackt sind, ist der Ziehen/Stecken-Alarm für die ET 200S gesperrt!

Siehe auch

Einführung zu PROFIBUS DPV1 (Seite 2926)

DPV1-Geräte konfigurieren (Seite 2927)

ET 200S konfigurieren (Seite 2889)

Weitere Konfigurationen

Konfigurieren von weiteren Funktionen

Das dezentrale Peripheriesystem ET 200S verfügt über zahlreiche weitere Funktionen, die über steckbare Module (z. B. Kommunikationsmodule) zur Verfügung stehen. Die Beschreibungen dazu finden Sie in den Abschnitten "Technologiefunktionen einsetzen" und "Konfigurationen für Punkt-zu-Punkt-Kopplungen".

Weitere Dezentrale Peripheriesysteme

Informationen zu weiteren Dezentralen Peripheriesystemen finden Sie über weiterführende Links unter "Siehe auch".

1.4.9.8 Intelligente DP-Slaves konfigurieren

I-Slave in ein DP-Mastersystem einfügen

Einführung

Merkmal eines intelligenten DP-Slaves (I-Slave) ist, dass Ein-/Ausgangsdaten nicht unmittelbar von einem realen Ein-/Ausgang dem DP-Master zur Verfügung gestellt werden, sondern von einer vorverarbeitenden CPU. Diese vorverarbeitende CPU bildet zusammen mit dem CP den I-Slave.

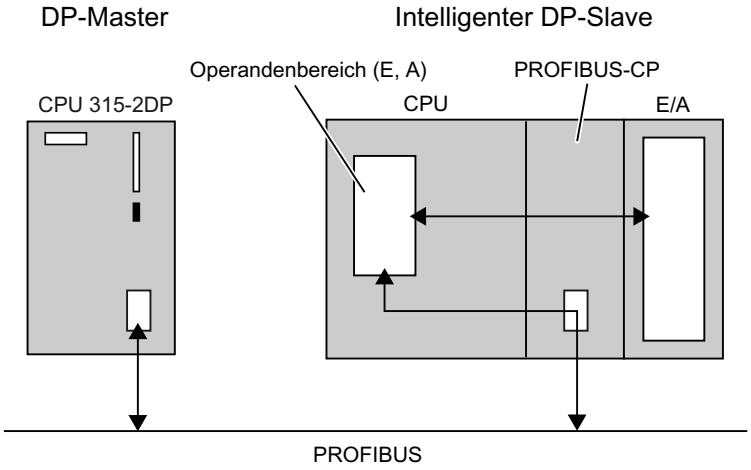

Unterschied: DP-Slave - Intelligenter DP-Slave

Bei einem DP-Slave greift der DP-Master auf die dezentralen Ein-/Ausgänge zu.

Bei einem intelligenten DP-Slave greift der DP-Master nicht auf die angeschlossenen Ein-/Ausgänge des intelligenten DP-Slaves zu, sondern auf einen Übergabebereich im E/A-Adressraum der vorverarbeitenden CPU. Das Anwenderprogramm der vorverarbeitenden CPU muss für den Austausch der Daten zwischen Operandenbereich und Ein-/Ausgängen sorgen.

Hinweis

Die konfigurierten E/A-Bereiche für den Datenaustausch zwischen DP-Master und DP-Slaves dürfen nicht von E/A-Baugruppen verwendet werden.

Anwendungen

Konfigurationen mit intelligenten DP-Slaves:

- Datenaustausch I-Slave <> DP-Master
- Direkter Datenaustausch DP-Slave > I-Slave

Vorgehen

Um einen I-Slave in ein DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus dem Hardware-Katalog zwei Geräte mit PROFIBUS-DP-Schnittstelle für die Konfiguration als DP-Master und I-Slave (z. B. CPU 317-2 DP) in die Netzsicht.
2. Schalten Sie das als I-Slave zu verwendende Gerät im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle" in die Betriebsart "Slave". Das Gerät wird nun als intelligenter DP-Slave betrieben.
3. Ziehen Sie eine Verbindungslinie zwischen den DP-Schnittstellen der beiden Geräte. Dadurch verbinden Sie den I-Slave mit einem DP-Master in einem DP-Mastersystem.

Ergebnis

Sie haben nun ein DP-Mastersystem mit einem DP-Master und einem I-Slave aufgebaut.

Beachten Sie die folgenden Beispiele für die Konfigurierung von I-Slaves:

- CP als I-Slave konfigurieren (Seite 2907)
- CPU als I-Slave konfigurieren (Seite 2908)
- DP-Slave als I-Slave konfigurieren (Seite 2909)

Siehe auch

Alarme im I-Slave (Seite 2905)

Diagnoseadressen konfigurieren (Seite 2902)

Diagnoseadressen konfigurieren

Einführung

Bei der Vergabe von Diagnoseadressen für den PROFIBUS-DP müssen Sie beachten, dass sowohl dem DP-Master als auch dem I-Slave DP-Diagnoseadressen zugeordnet sind.

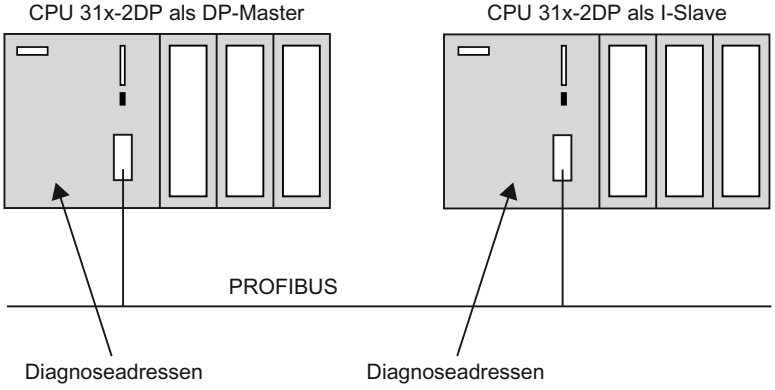

Diagnoseadressen

Bei den Diagnoseadressen wird zwischen Diagnoseadressen im DP-Master und im I-Slave unterschieden:

- Diagnoseadressen im DP-Master
  Über die dem DP-Master zugeordnete Diagnoseadressen erhält der DP-Master Auskunft über den Zustand des DP-Slaves bzw. über eine Busunterbrechung.
- Diagnoseadressen im I-Slave
  Über die dem I-Slave zugeordnete Diagnoseadressen erhält der I-Slave Auskunft über den Zustand des DP-Masters bzw. über eine Busunterbrechung.

Beim Konfigurieren der Master-I-Slave-Konfiguration vergibt STEP 7 automatisch Diagnoseadressen für die jeweiligen CPUs. Die folgende Beschreibung zeigt, wie Sie die vorgegebenen Diagnoseadressen ändern.

Tipp: Um eine Übersicht aller für die CPU vergebenen Adressen einschließlich Diagnoseadressen zu erhalten, wählen Sie im Inspektorfenster der markierten CPU den Bereich "Adressübersicht".

Diagnoseadressen der I-Slave-Kommunikation

Am I-Slave wird eine Tabelle mit den Diagnoseadressen der I-Slave-Kommunikation angezeigt. Als Bezeichner für die Diagnoseadresse wird der Name der PLC und der Transferbereich mit der niedrigsten Subslot-Nummer verwendet. Die verwendeten Diagnoseadressen können Sie in der Tabelle bearbeiten.

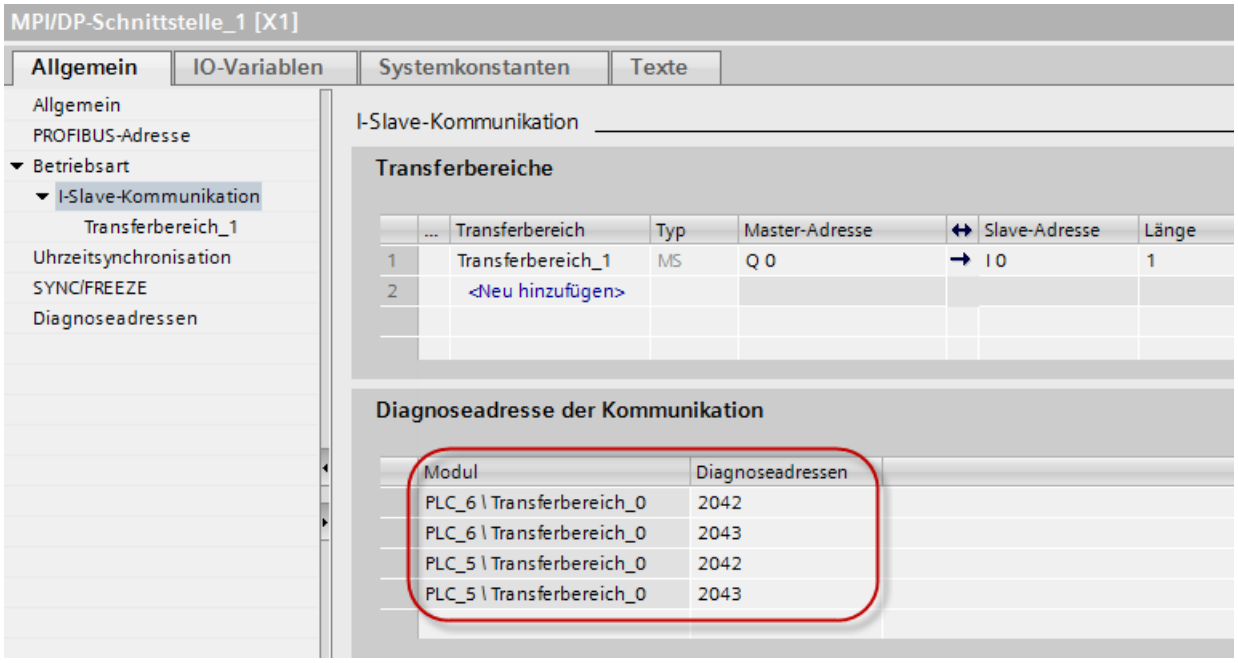

Die Tabelle wird nur angezeigt, wenn wenigstens der DP-Master oder der I-Slave eine CPU 300/400 ist:

DP-Master	I-Slave	Anzeige der Diagnoseadressen
CPU 300/400	CPU 300/400	lokale und entfernte Adressen des I-Slaves
CPU 300/400	CPU 1200/1500	entfernte Adressen des I-Slaves
CPU 1200/1500	CPU 300/400	lokale Adressen des I-Slaves
CPU 1200/1500	CPU 1200/1500	keine Anzeige

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Sie haben ein DP-Mastersystem mit DP-Master und I-Slave angelegt.

Vorgehen

Um die Diagnoseadressen für die I-Slave-Kommunikation zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den DP-Master oder den I-Slave.
2. Wählen Sie in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "I-Slave".
3. Bearbeiten Sie die Diagnoseadressen im Inspektorfenster unter "Eigenschaften > I-Slave-Kommunikation > Diagnoseadresse der Kommunikation".

Oder:

1. Selektieren Sie den I-Slave.
2. Bearbeiten Sie die Diagnoseadressen im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle > Betriebsart > I-Slave-Kommunikation > Diagnoseadresse der Kommunikation".

Sie können die Felder mit den Werten der Diagnoseadressen selektieren und bearbeiten, und zwar sowohl für den Slave als auch für den Master.

Weitere Informationen

Beachten Sie für weitere Informationen zu Diagnoseadressen die Handbücher zu den einzelnen S7-CPUs.

Alarme im I-Slave

Alarmtypen in Abhängigkeit von DPV1 und DPV0

I-Slaves können mit der erweiterten Anweisung "SALRM" (SFB 75) im zugeordneten DP-Master Alarme auslösen. Die folgende Tabelle zeigt die möglichen Alarmtypen abhängig von PROFIBUS DPV1 und DPV0:

Alarmtyp	DPV0	DPV1
Diagnosealarm (OB 82)	Ja	Ja
Prozessalarm (OB 40 bis 47)	Ja	Ja
Ziehen-/Steckenalarm (OB 83)	Ja (wenn I-Slave diesen Alarm unterstützt)	Ja
Statusalarm (OB 55)	Nein	Ja
Update-Alarm (OB 56)	Nein	Ja
Herstellerspezifischer Alarm (OB 57)	Nein	Ja

Alarmauslösende Adressen

Benutzen Sie die im Inspektorfenster des I-Slave unter "Eigenschaften > DP-Schnittstelle > Betriebsart > I-Slave-Kommunikation" konfigurierte Adressen, um mit der erweiterten Anweisung "SALRM" Alarme auszulösen. Diese Adressen sind keiner realen Baugruppe zugeordnet, sondern entsprechen virtuellen "Steckplätzen".

Die Adressen für Betriebszustandsübergänge und Gerätediagnose können Sie nicht zum Auslösen von Alarmen benutzen.

Prinzipielle Funktionsweise der Alarmerzeugung

Das Prinzip der Alarmerzeugung wird anhand eines Diagnosealarms gezeigt.

- Im I-Slave ist die Ausgangsadresse 0 für einen virtuellen Steckplatz vergeben.
- Die Ausgangsadresse 0 wird im Beispiel dazu benutzt, um im DP-Master einen Diagnosealarm (OB 82) auszulösen.

Mit dem Diagnosealarm werden per Anwenderprogramm Daten mitgegeben (AINFO). Diese Daten müssen dem prinzipiellen Aufbau der Alarmzusatzinformation entsprechen.

Folgender vereinfachter Aufbau ist z. B. möglich:

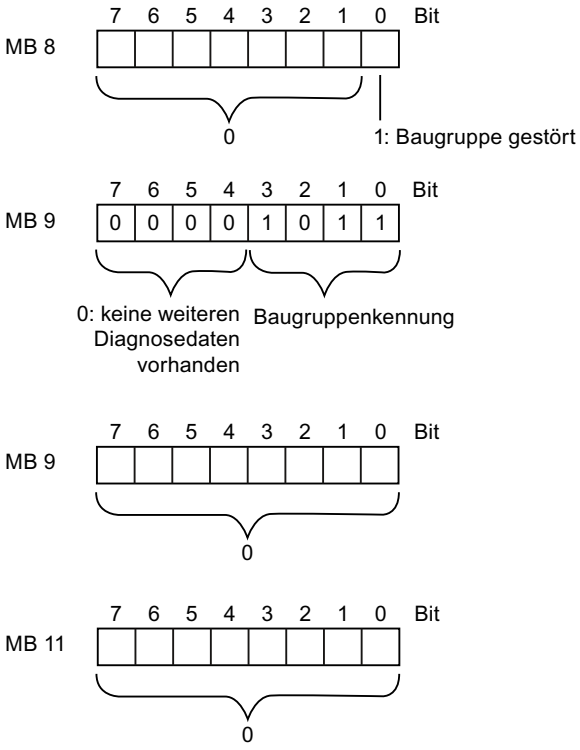

Weitere Informationen

Beachten Sie weitere Informationen zu dem oben dargestellten Aufbau im Referenzhandbuch zu den System- und Standardfunktionen für S7-300/400 im Band 2, Kap. Diagnosedaten und den in den Handbücher zu den System- und Diagnosefunktionen der S7-1500.

Hinweis

Die Alarmzusatzinformationen beeinflussen die Baugruppenzustandsdaten und die Fehler-LED "SF" des I-Slaves (S7-300/400). Ebenso werden durch den Alarm die Baugruppenzustandsdaten und Fehler-LEDs des zugeordneten DP-Masters beeinflusst. Sie müssen daher für die Zusammenstellung der Daten für die Alarmzusatzinformation die Bedeutung der Diagnose-Datensätze (Datensatz 0 und Datensatz 1) berücksichtigen!

Das folgende Bild zeigt den Ablauf der Alarmerzeugung:

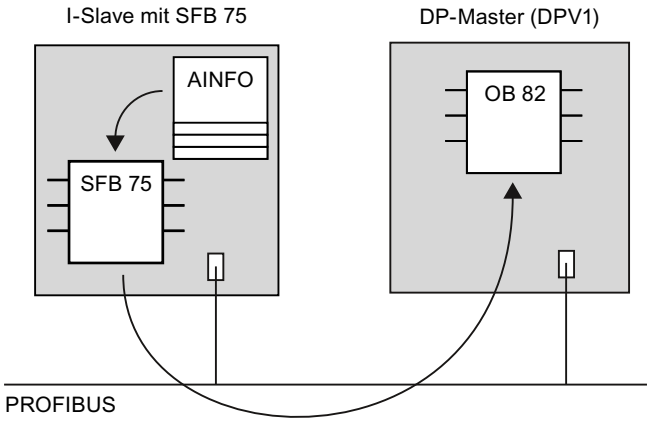

Ein Aufruf der erweiterten Anweisung "SALRM" kann wie folgt aussehen:

AWL

```
CALL   FB     SALRM , DB75
       REQ    :=M0.0
       ID     :=DW#16#8000     //Ausgangsadresse 0
       ATYPE  :=1              //Diagnosealarm
       ASPEC  :=1              //Alarm kommend, EG gestört
       LEN    :=4              //Länge Alarminfo 4 Bytes
       DONE   :=M1.3
       BUSY   :=M1.4
       ERROR  :=M1.5
       STATUS:=MD50
       AINFO  :=P#M 8.0 BYTE 4 //Bereich für Alarminfo
```

Siehe auch

Einführung zu PROFIBUS DPV1 (Seite 2926)

DPV1-Geräte konfigurieren (Seite 2927)

Beispiel für das Konfigurieren eines CP als I-Slave

In diesem Beispiel konfigurieren Sie eine Station mit einem CP 342-5 als I-Slave. Den CP 342-5 müssen Sie dafür über die Betriebsart "DP-Slave" zu einem intelligenten DP-Slave machen.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein CP 342-5 ist in der Netzsicht vorhanden.
- Der CP 342-5 wurde in der Gerätesicht mit E/A-Baugruppen bestückt.
- Ein DP-Master (CPU mit integrierter PROFIBUS-DP-Schnittstelle oder CP mit PROFIBUS-DP-Schnittstelle) ist in der Netzsicht vorhanden.

Vorgehen

Um einen CP als I-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie den CP 342-5.
2. Schalten Sie den CP 342-5 im Inspektorfenster unter "Eigenschaften > Betriebsart" in die Betriebsart "DP-Slave".
3. Ziehen Sie eine Verbindung von der DP-Schnittstelle der CP 342-5 zu der DP-Schnittstelle des DP-Masters.
4. Selektieren Sie den CP 342-5.
5. Klicken Sie im Inspektorfenster unter "Eigenschaften > Betriebsart" auf "I-Slave-Kommunikation".
6. Bearbeiten Sie die Adresszuordnung zwischen DP-Master und I-Slave.

Alternativ können Sie nach Selektion des CP 342-5 in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "I-Slave" anwählen, um dann im Inspektorfenster unter "Eigenschaften" und "I-Slave-Kommunikation" die Adresszuordnung bearbeiten.

Weitere Informationen

Beachten Sie weitere Informationen zum Datenaustausch zwischen vorverarbeitender CPU und CP 342-5 DP innerhalb des DP-Slaves im Handbuch für NCM S7 Industrial Ethernet.

Beispiel für das Konfigurieren einer CPU als I-Slave

In diesem Beispiel konfigurieren Sie eine Station mit einer CPU 31X-2 DP mit integrierter DP-Schnittstelle (z. B. CPU 315-2 DP) als I-Slave. Die CPU 31X-2 DP benötigt eine integrierte DP-Schnittstelle und muss von Ihnen über die Betriebsart "DP-Slave" zu einem intelligenten DP-Slave gemacht werden.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Eine CPU 315-2 DP ist in der Netzsicht vorhanden.
- Die CPU 315-2 DP wurde in der Gerätesicht mit E/A-Baugruppen bestückt.
- Ein DP-Master (CPU mit integrierter PROFIBUS-DP-Schnittstelle oder CP mit PROFIBUS-DP-Schnittstelle) ist in der Netzsicht vorhanden.

Vorgehen

Um eine CPU als I-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU 315-2 DP.
2. Schalten Sie die CPU 315-2 DP im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle > Betriebsart" in die Betriebsart "DP-Slave".
3. Ziehen Sie eine Verbindung von der DP-Schnittstelle der CPU 315-2 DP zu der DP-Schnittstelle des DP-Masters.

4. Selektieren Sie die CPU 315-2 DP.
5. Klicken Sie im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle > Betriebsart" auf "I-Slave-Kommunikation".
6. Bearbeiten Sie die Adresszuordnung zwischen DP-Master und I-Slave.

Alternativ können Sie nach Selektion der CPU 315-2 DP in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "I-Slave" anwählen, um dann im Inspektorfenster unter "Eigenschaften > I-Slave-Kommunikation" die Adresszuordnung bearbeiten.

Weitere Informationen

Beachten Sie weitere Informationen zu der Konfiguration von CPUs als I-Slaves in den Referenzhandbüchern der jeweiligen CPUs.

Beispiel für das Konfigurieren einer ET 200 CPU als I-Slave

In diesem Beispiel konfigurieren Sie eine ET 200S mit dem Interfacemodul IM 151-7 CPU als I-Slave.

Hinweis

Sie finden das Interfacemodul der ET 200S im Hardware-Katalog unter "PLC > SIMATIC ET 200 PLC > ET 200S CPU".

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Eine ET 200S mit IM 151-7 CPU ist in der Netzsicht vorhanden.
- Die ET 200S wurde in der Gerätesicht mit E/A-Erweiterungsmodulen bestückt.
- Ein DP-Master (CPU mit integrierter PROFIBUS-DP-Schnittstelle oder CP mit PROFIBUS-DP-Schnittstelle) ist in der Netzsicht vorhanden.

Vorgehen

Um einen DP-Slave als I-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie das IM 151-7 CPU.
2. Ziehen Sie eine Verbindung von der DP-Schnittstelle des IM 151-7 CPU zu der DP-Schnittstelle des DP-Masters.

   Hinweis

   Einige Geräte haben fest eingestellte Schnittstellen für Master oder Slave. Unter "Eigenschaften > MPI/DP-Schnittstelle" und/oder "Eigenschaften > DP-Schnittstelle" finden Sie die Einstellung der entsprechenden Betriebsart.

3. Selektieren Sie das IM 151-7 CPU.

4. Wählen Sie in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "I-Slave".
5. Bearbeiten Sie die Adresszuordnung zwischen DP-Master und I-Slave im Inspektorfenster unter "Eigenschaften >I-Slave-Kommunikation".

Weitere Informationen

Beachten Sie weitere Informationen zu der Konfiguration von DP-Slaves als I-Slaves in den Gerätehandbüchern und Betriebsanleitungen der dezentralen Peripheriesysteme ET 200.

Beispiel: I-Slave projektieren (S7-1500, ET 200SP CPU)

Voraussetzungen für das Konfigurieren eines I-Slaves

Der I-Slave besteht z. B. aus:

- Einer CPU aus S7-1500 und einem Kommunikationsmodul CM 1542-5/CP 1542-5 (STEP 7 ab V12)
- Einer CPU aus ET 200SP und einem Kommunikationsmodul CM DP (STEP 7 ab V13 SP1)

Vorgehen zum Konfigurieren eines I-Slaves

Dieser Abschnitt zeigt, wie Sie am Beispiel einer CPU 1512SP-1 PN einen I-Slave konfigurieren. Das Vorgehen für die CPUs aus S7-1500 mit CM 1542-5/CP 1542-5 sowie für die CPU 1510SP-1 PN mit CM DP unterscheidet sich nicht.

Um einen I-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus dem Hardware-Katalog eine CPU 1512SP-1 PN per Drag & Drop in die Netzsicht.
2. Öffnen Sie die Gerätesicht der CPU.
3. Doppelklicken Sie im Hardware-Katalog auf das Kommunikationsmodul CM DP. STEP 7 legt das CM DP in der Gerätesicht an.
4. Selektieren Sie die PROFIBUS-Schnittstelle des Kommunikationsmoduls CM DP.

5. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Betriebsart" und aktivieren Sie das Optionskästchen "DP-Slave".
6. In der Klappliste "Zugewiesener DP-Master" haben Sie jetzt die Möglichkeit, den DP-Master auszuwählen.
   Wenn Sie den DP-Master ausgewählt haben, werden anschließend die Vernetzung und das DP-Mastersystem zwischen beiden Geräten in der Netzsicht angezeigt.

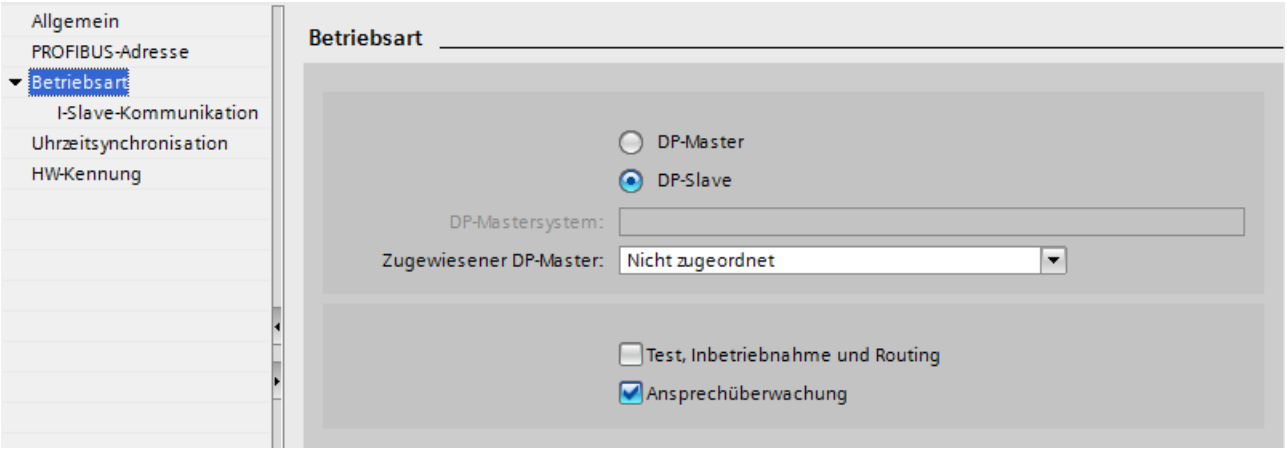

Hinweis

Betrieb über GSD-Datei

Wenn Sie einen I-Slave über GSD-Datei betreiben, dann dürfen Sie das Kontrollkästchen "Test, Inbetriebnahme und Routing" nicht aktivieren.

Legen Sie an der PROFIBUS-Schnittstelle des I-Slaves ein DP-Subnetz an.

Beispiel: Transferbereiche projektieren

Voraussetzungen für das Projektieren von Transferbereichen

- Sie haben in STEP 7 einen I-Slave konfiguriert.
- Sie befinden sich in der Gerätesicht des I-Devices und haben die PROFIBUS-Schnittstelle des Kommunikationsmoduls ausgewählt.

Vorgehen zum Projektieren von Transferbereichen

Um Transferbereiche für einen I-Slave zu projektieren, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Bereichsnavigation in den Abschnitt "Betriebsart" > "I-Slave-Kommunikation" > "Transferbereiche".
2. Legen Sie Transferbereiche an. Stellen Sie die Eigenschaften der angelegten Transferbereiche ein.

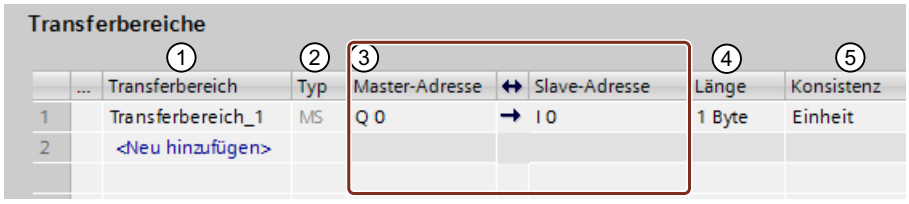

① Klicken Sie in das erste Feld der Spalte "Transferbereich". STEP 7 vergibt einen voreingestellten Namen, den Sie ändern können.

② Wählen Sie den Typ der Kommunikationsbeziehung: Aktuell können Sie nur MS für "Master-Slave-Kommunikationsbeziehung" auswählen.

③ STEP 7 vergibt die Adressen für den Transferbeich automatisch. Korrigieren Sie die Adressen, wenn erforderlich.

④ Stellen Sie die Länge und Einheit des Transferbereiches ein. Geben Sie die Länge des Transferbereichs in folgender Form an: [1...64] Byte/Wort. Beispiele: "32 Byte", "64 Wort"

⑤ Legen Sie fest, ob Transferbereich byte- bzw. wortgranular oder über die gesamte Länge konsistent zwischen DP-Master und I-Slave ausgetauscht wird.

Für jeden Transferbereich wird in der Bereichsnavigation ein einzelner Eintrag erzeugt. Wenn Sie einen dieser Einträge auswählen, können Sie die Details des Transferbereichs anpassen bzw. korrigieren und kommentieren.

Diagnose und Alarmverhalten (S7-1500, ET 200SP CPU)

Diagnose und Alarmverhalten

S7-CPUs verfügen über vielfältige Diagnose- und Alarmfunktionen, die beispielsweise Fehler oder Ausfälle von untergeordneten DP-Systemen melden können. Diese Diagnosemeldungen verringern Ausfallzeiten und erleichtern die Lokalisierung und Behebung von Fehlern.

Diagnosemöglichkeiten im übergeordneten DP-Master und im I-Slave

Dem übergeordneten DP-Master und dem I-Slave stehen folgende Diagnosemechanismen zur Verfügung:

- OB 82 (Diagnosealarm)
  Wenn der I-Slave einen Betriebszustandswechsel durchführt, dann ruft der DP-Master den OB 82 (Diagnosealarm) auf.
  Wenn der DP-Master einen Betriebszustandswechsel durchführt, dann ruft der I-Slave den OB 82 (Diagnosealarm) auf.
- OB 86 (Baugruppenträgerausfall)
  Wenn die Busverbindung zum I-Slave unterbrochen wird, dann ruft der DP-Master den OB 86 (Baugruppenträgerausfall) auf.
  Wenn die Busverbindung zum DP-Master unterbrochen wird, dann ruft der I-Slave den OB 86 (Baugruppenträgerausfall) auf.
- OB 122 (Peripheriezugriffsfehler)
  Sofern Sie für den OB 122 nicht das Attribut "Lokale Fehlerbehandlung im Baustein" eingestellt haben, gilt Folgendes:
  - Wenn die Busverbindung zum I-Slave unterbrochen wird, dann ruft der DP-Master bei Direktzugriff auf die entsprechenden Transferbereiche den OB 122 (Peripheriezugriffsfehler) auf.
  - Wenn die Busverbindung zum DP-Master unterbrochen wird, dann ruft der I-Slave bei Direktzugriff auf die entsprechenden Transferbereiche den OB 122 (Peripheriezugriffsfehler) auf.

Verhalten der Transferbereiche bei Betriebszustandsübergängen

DP-Master	I-Slave	Verhalten Eingangstransferbereiche DP-Master	Verhalten Eingangstransferbereiche I-Slave
RUN→STOP	RUN	Keine Aktualisierung des Prozessabbilds	Die Eingangstransferbereiche behalten ihre Aktualwerte. (kein Zugriffsfehler)
STOP→RUN	RUN	Die Eingangstransferbereiche werden vom zyklischen Anwenderprogramm über das Prozessabbild aktualisiert.	Die Eingangstransferbereiche werden über Prozessabbild aktualisiert.
RUN	RUN→STOP	Der I-Slave setzt die Eingangstransferbereiche im DP-Master auf "0".	Keine Aktualisierung des Prozessabbilds
RUN	STOP→RUN	Der I-Slave setzt die Eingangstransferbereiche im DP-Master auf "0". Nach dem Anlaufprogramm des I-Slave werden die Eingangstransferbereiche des DP-Masters über das Prozessabbild aktualisiert.	Die Eingangstransferbereiche werden vor der Bearbeitung des Anlaufprogramms über das Prozessabbild aktualisiert.

1.4.9.9 Direkten Datenaustausch konfigurieren

Wissenswertes zum Direkten Datenaustausch

Einführung

Der Direkte Datenaustausch (Querverkehr) ist eine weitere Form der E/A-Kommunikation, die auch über ein DP-Mastersystem hinaus wirksam sein kann. Im Direkten Datenaustausch kann der Empfänger nur Daten lesen, nicht schreiben.

In einem DP-Mastersystem steuert ein DP-Master ausschließlich die ihm zugeordneten DP-Slaves. Mit dem Direkten Datenaustausch kann ein anderer DP-Master oder DP-Slave als Empfänger am PROFIBUS-Subnetz "mithören", welche Eingangsdaten ein DP-Slave als Sender seinem DP-Master sendet. Diese Übertragung von Daten zwischen DP-Slaves bezeichnet man als Direkten Datenaustausch.

Funktionsweise

In einer Konfiguration für den Direkten Datenaustausch werden lokale Eingangs-Adressbereiche eines intelligenten DP-Slaves oder eines DP-Masters den Eingangs-Adressbereichen eines PROFIBUS-DP-Partners zugeordnet.

Über die zugeordneten Eingangs-Adressbereiche empfängt der intelligente DP-Slave oder der DP-Master die Daten von seinem PROFIBUS-DP-Partner.

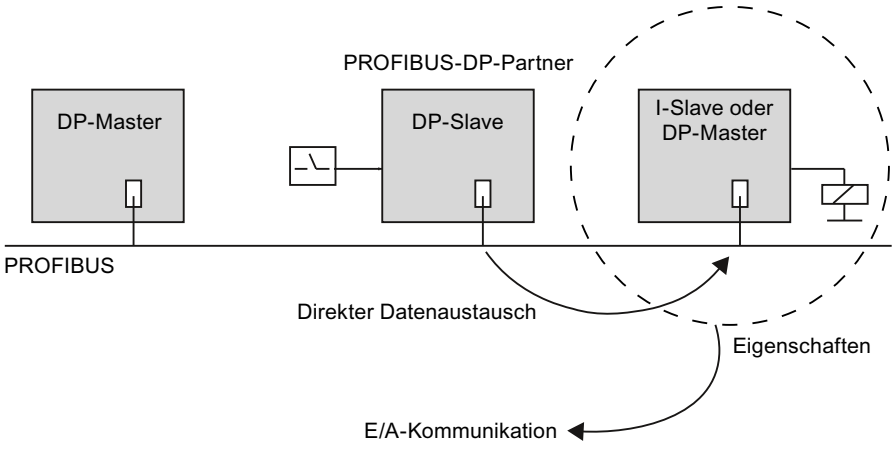

Konfigurationsmöglichkeiten

Mögliche Anwendungsfälle für den Direkten Datenaustausch sind:

- Konfiguration mit intelligenten DP-Slaves: DP-Slave > I-Slave
- Konfiguration mit zwei DP-Mastersystemen: DP-Slave > DP-Master

Anzahl anschließbarer PROFIBUS-DP-Partner

Die Summe der PROFIBUS-DP-Partner, welche direkt an einer DP-Schnittstelle angeschlossen oder mittels Direktem Datenaustausch über diese angesprochen werden, ist schnittstellenspezifisch beschränkt. An der MPI/DP-Schnittstelle können maximal 32 PROFIBUS-DP-Partner angesprochen werden.

Siehe auch

Konfigurationen mit mehreren DP-Mastersystemen (Seite 2875)

Konfigurationen mit intelligenten DP-Slaves (Seite 2876)

Direkten Datenaustausch konfigurieren

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Mastersystem mit Eingabebaugruppen am DP-Slave ist vorhanden.
- Ein DP-Master oder DP-Slave ist als "Empfänger" mit dem DP-Mastersystem verbunden.

Direkten Datenaustausch anlegen

Um den Direkten Datenaustausch anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gewünschten Kommunikationsteilnehmer (Sender oder Empfänger) durch Mausklick.
2. Ziehen Sie mit gedrückter linker Maustaste den zweiten Kommunikationspartner in das Feld "Gerät hier loslassen oder auswählen" der Spalte "Partner 2" der E/A-Kommunikationstabelle.

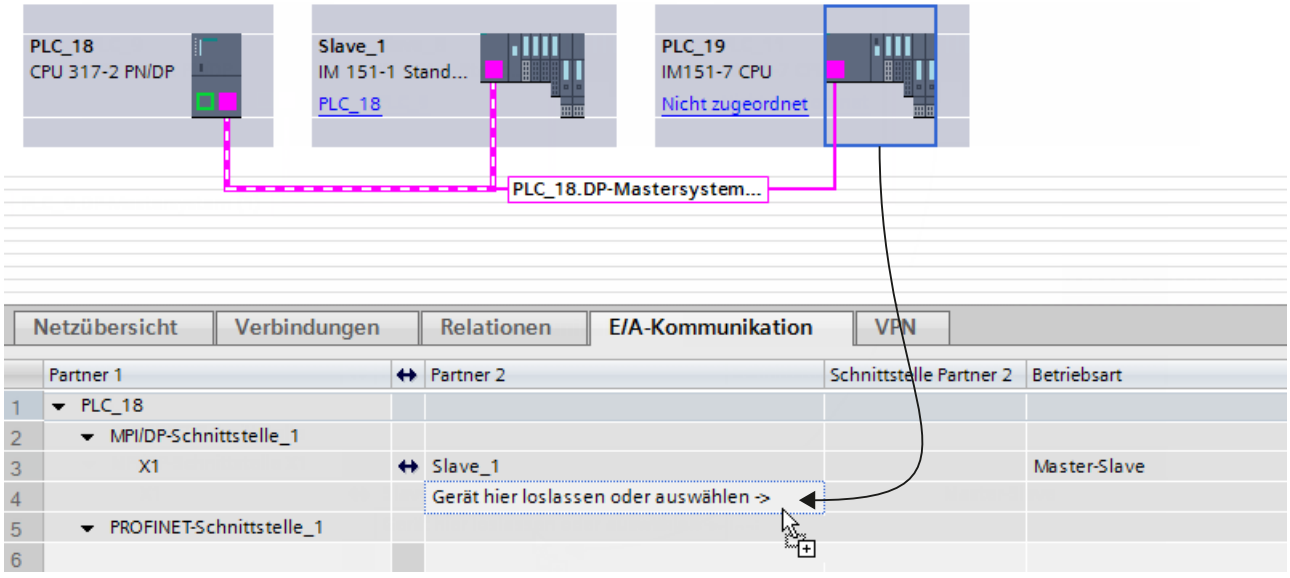

   Alternativ klicken Sie direkt in das Feld "Gerät hier loslassen..." und wählen Sie den gewünschten Verbindungspartner aus der Klappliste aus.

Sie haben nun in der Tabelle über Drag & Drop die Kommunikationsbeziehung zwischen zwei Teilnehmern angelegt. In der E/A-Kommunikationstabelle wird in einer neuen Zeile die Betriebsart "Direkter Datenaustausch" angezeigt.

Hinweis

Bei einem Direkten Datenaustausch zwischen zwei I-Slaves wird immer zuerst der Sender selektiert. Anschließend wird der Empfänger in das Feld "Gerät hier loslassen oder auswählen" gezogen oder aus der Klappliste ausgewählt. Die Reihenfolge bestimmt hier die Kommunikationsrichtung des Direkten Datenaustausches.

Wenn Sie zum Anlegen des Direkten Datenaustausches die Klappliste in der Spalte "Partner 2" der E/A-Kommunikationstabelle verwenden wollen, können Sie die Kommunikationsrichtung des Direkten Datenaustausches direkt über die Auswahl der Klappliste bestimmen.

Direkten Datenaustausch konfigurieren

Um den Direkten Datenaustausch zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Teilnehmer des direkten Datenaustausches.
2. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "Direkter Datenaustausch".
3. Selektieren Sie im Inspektorfenster "Eigenschaften > Direkter Datenaustausch".
4. Legen Sie unter "Direkter Datenaustausch" in der Transferbereichstabelle einen neuen Transferbereich an:
   Klicken Sie in der Spalte "Transferbereich" auf "<Neu hinzufügen>" und wählen Sie in der Klappliste "Typ" die Kommunikationsart "DX" für den Direkten Datenaustausch. Sie können den neuen Transferbereich auch anlegen, indem Sie nur unter "Typ" die Kommunikationsart in einer neuen Spalte einstellen. Der Transferbereich wird dann automatisch erstellt.
5. Klicken Sie unter "Direkter Datenaustausch" auf den neu angelegten Transferbereich.
   Es wird die Detailansicht des Transferbereichs geöffnet.
6. Bearbeiten Sie die Eigenschaften für Sender und Empfänger und allgemeine Parameter des Direkten Datenaustausches.

Sie haben nun den Direkten Datenaustausch konfiguriert.

Direkten Datenaustausch löschen

Um den Direkten Datenaustausch zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "Direkter Datenaustausch".
2. Drücken Sie <Entf> oder wählen Sie über das Kontextmenü den Befehl "Löschen".

Beispiel für das Konfigurieren des Direkten Datenaustauschs

Einführung

In diesem Beispiel konfigurieren Sie drei CPUs in einer Master-Slave-Konfiguration für den Direkten Datenaustausch. Danach vergeben Sie aussagekräftige Namen und projektieren die Adressbereiche.

Verwenden Sie folgende CPUs mit folgender Konfiguration:

CPU	Aufgabe	Adresse	Daten
317-2 PN/DP	DP-Master	E 200	8 Worte vom Sender
315-2 DP	Sender	A 100	8 Worte zum Master
317-2 DP	Empfänger	E 120	Erste 2 Bytes aus den 8 Worten des Senders

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Der Hardware-Katalog ist geöffnet.

PROFIBUS-DP-Mastersystem anlegen

Um das PROFIBUS-DP-Mastersystem anzulegen, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit Drag & Drop die CPUs 315-2 DP, 317-2 DP und 317-2 PN/DP aus dem Hardware-Katalog in die Netzsicht.
2. Geben Sie den CPUs aussagekräftige Namen:
   – Selektieren Sie die CPU 317-2 PN/DP.
   – Geben Sie der CPU im Inspektorfenster unter "Eigenschaften > Allgemein" den Namen "DP-Master".

   Fahren Sie entsprechend mit den anderen CPUs fort und vergeben Sie die Namen "Sender" und "Empfaenger".
3. Selektieren Sie die CPU 317-2 PN/DP und ändern Sie ggf. im Inspektorfenster unter "Eigenschaften > MPI/DP-Schnittstelle > MPI-Adresse" den Schnittstellentyp von "MPI" nach "PROFIBUS".
   Hinweis: Wenn der Schnittstellentyp der MPI/DP-Schnittstelle auf "PROFIBUS" eingestellt ist, lautet der Eintrag nicht mehr "MPI-Adresse" sondern "PROFIBUS-Adresse".
4. Selektieren Sie nacheinander die beiden CPUs 315-2 DP und 317-2 DP und wählen Sie im Inspektorfenster unter "Eigenschaften > DP-Schnittstelle > Betriebsart" die Betriebsart "Slave".
   Die beiden CPUs 315-2 DP "Sender" und 317-2 "Empfaenger" werden nun als I-Slaves betrieben.
5. Ziehen Sie mit gedrückter Maustaste die PROFIBUS-Verbindungen von der Master-CPU an die beiden Slave CPUs.

Sie haben nun ein PROFIBUS-Mastersystem mit der CPU 317-2 PN/DP als DP-Master und den CPUs 315-2 DP "Sender" und 317-2 DP "Empfaenger" als I-Slaves angelegt. In der E/A-Kommunikationstabelle der CPUs wurden neue Zeilen für die Kommunikationspartner angelegt. In der Spalte "Betriebsart" wird "I-Slave" angezeigt.

Direkten Datenaustausch anlegen

Um innerhalb des PROFIBUS-Mastersystems den Direkten Datenaustausch anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den "Sender".
2. Ziehen Sie mit gedrückter linker Maustaste den "Empfaenger" in das Feld "Gerät hier loslassen" der Spalte "Partner 2" der E/A-Kommunikationstabelle.
   Hinweis: Sie können auch die Klappliste im Feld "Gerät hier loslassen" der Spalte "Partner 2" öffnen und daraus den "Empfaenger" auswählen.

Sie haben nun zwischen den beiden I-Slaves einen Direkten Datenaustausch angelegt. In der E/A-Kommunikationstabelle der beiden I-Slaves wurde eine neue Zeile angelegt. In der Spalte "Betriebsart" wird "Direkter Datenaustausch" angezeigt.

Hinweis

Die Pfeilsymbole der E/A-Kommunikationstabelle zwischen den Spalten "Partner 1" und "Partner 2" zeigen die Richtungen der Kommunikationsbeziehungen an. Bei Master-Slave- oder Master-I-Slave-Beziehungen zeigen die Pfeile in beide Richtungen. Bei Kommunikationsbeziehungen zwischen I-Slaves oder im Direkten Datenaustausch zeigen die Pfeile vom Sender zum Empfänger.

Master-Slave-Kommunikation konfigurieren

Um die Master-Slave-Kommunikation auszuwählen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen der beiden Kommunikationspartner der Master-Slave-Kommunikation (DP-Master CPU 317-2 PN/DP oder I-Slave CPU 315-2 DP "Sender")
2. Klicken Sie in der E/A-Kommunikationstabelle in die Zeile mit dem entsprechenden zweiten Partner und der Betriebsart "I-Slave".
3. Selektieren Sie im Inspektorfenster "Eigenschaften > I-Slave-Kommunikation".

Oder:

1. Selektieren Sie den I-Slave CPU 315-2 DP "Sender".
2. Selektieren Sie im Inspektorfenster "Eigenschaften > DP-Schnittstelle > Betriebsart > I-Slave-Kommunikation".

Um die Master-Slave-Kommunikation zu konfigurieren, gehen Sie folgendermaßen vor:

1. Legen Sie unter "I-Slave-Kommunikation" in der Transferbereichstabelle einen neuen Transferbereich an:
   Klicken Sie in der Spalte "Transferbereich" auf "<Neu hinzufügen>" und wählen Sie in der Klappliste "Typ" die Kommunikationsart "MS" für die Master-Slave-Kommunikation. Sie können den neuen Transferbereich auch anlegen, indem Sie nur unter "Typ" die Kommunikationsart in einer neuen Spalte einstellen. Der Transferbereich wird dann automatisch erstellt.
2. Klicken Sie unter "I-Slave-Kommunikation" auf den neu angelegten Transferbereich.
   Es wird die Detailansicht des Transferbereichs geöffnet.
3. Machen Sie folgende Eingaben:
   für "DP-Master":
   - Adresstyp I
   - Anfangsadresse 200
   - Länge 8
   - Einheit WORD
   - Konsistenz Gesamte Länge

   für "Sender":
   - Adresstyp Q
   - Anfangsadresse 100

Nach der Angabe der Datenlänge werden die eingegebenen Slave- und Masteradressen für den benötigten Datenraum vervollständigt. Wenn Sie im Inspektorfenster auf "I-Slave-Kommunikation" klicken, sehen Sie in der Übersicht die aktualisierte Konfiguration der Master-Slave-Kommunikation.

Direkten Datenaustausch konfigurieren

Um den Direkten Datenaustausch zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen der beiden Kommunikationspartner des Direkten Datenaustausches (I-Slave CPU 315-2 DP "Sender" oder I-Slave CPU 317-2 DP "Empfaenger")
2. Klicken Sie in der E/A-Kommunikationstabelle in die Zeile mit der Betriebsart "Direkter Datenaustausch".
3. Selektieren Sie im Inspektorfenster "Eigenschaften > Direkter Datenaustausch".

Um den Direkten Datenaustausch zu konfigurieren, gehen Sie folgendermaßen vor:

1. Legen Sie unter "Direkter Datenaustausch" in der Transferbereichstabelle einen neuen Transferbereich an:
   Klicken Sie in der Spalte "Transferbereich" auf "<Neu hinzufügen>" und wählen Sie in der Klappliste "Typ" die Kommunikationsart "DX" für den Direkten Datenaustausch. Sie können den neuen Transferbereich auch anlegen, indem Sie nur unter "Typ" die Kommunikationsart in einer neuen Spalte einstellen. Der Transferbereich wird dann automatisch erstellt.
2. Klicken Sie unter "Direkter Datenaustausch" auf den neu angelegten Transferbereich.
   Es wird die Detailansicht des Transferbereichs geöffnet.
3. Machen Sie folgende Eingaben:
   für "Sender":
   - Adresse 100
   - Datenlänge 1
   - Einheit WORD

   für "Empfaenger":
   - Adresse 120

Hinweis

Wenn es sich bei dem Sender im Direkten Datenaustausch um einen normalen DP-Slave handelt, werden keine Transferbereiche konfiguriert. Stattdessen, müsste dem Sender eine Eingabebaugruppe hinzugefügt werden.

Besonderheit

Sie können prinzipiell auch eine größere Anfangsadresse als 200 einstellen, z. B. 202. Die Länge der konsistenten Daten wird automatisch angepasst. Außerdem können Sie eine kleinere Länge einstellen als der Sender vorgibt (z. B. 1 Byte).

Hinweis

Wenn beim Sender 3 oder mehr als 4 Bytes konsistente Länge eingestellt ist und die Daten mit der erweiterten Anweisung der Systemfunktion "DPWR_DAT" (SFC 15) übertragen werden, dann muss der Empfänger immer die erweiterte Anweisung "DPRD_DAT" (SFC 14) verwenden, auch wenn z. B. nur 1 Byte gelesen wird!

Wenn Sie für das Lesen von 1 Byte statt DPRD_DAT eine Lade-Operation (L EB..) verwenden, wird eine "0" eingelesen (falscher Wert).

Aufruf von DPWR_DAT im Sender (CPU 315-2 DP)

```
AWL
CALL  "DPWR_DAT"
       LADDR  :=W#16#64                //Anfangsadresse A 100
       RECORD :=P#M 10.0 BYTE 16       //Quellbereich für Nutzdaten
       RET_VAL:=MW100                  //Rueckgabewert
```

Aufruf von DPRD_DAT im Empfänger (CPU 317-2 DP)

```
AWL
CALL  "DPRD_DAT"
       LADDR  :=W#16#78                //Anfangsadresse E 120
       RET_VAL:=MW100                  //Rueckgabewert
       RECORD :=P#M 10.0 BYTE 2        //Zielbereich fuer Nutzdaten
```

1.4.9.10 GSD-Dateien verwenden

GSD-Revisionen

Wissenswertes zu GSD-Revisionen

Die Eigenschaften von DP-Slaves werden durch GSD-Dateien für Projektierungstools verfügbar gemacht. Funktionserweiterungen im Bereich der dezentralen Peripherie haben Auswirkungen auf die GSD-Spezifikation, z. B. führen sie zur Definition neuer Schlüsselwörter. Daraus ergibt sich eine Versionierung der Spezifikation. Die Version der Spezifikation, nach der sich eine GSD-Datei richtet, heißt bei GSD-Dateien "GSD-Revision". Die GSD-Revision ist als Schlüsselwort "GSD_Revision" in GSD-Dateien ab GSD-Revision 1 obligatorisch enthalten. GSD-Dateien ohne dieses Schlüsselwort werden daher von Projektierungstools als GSD-Revision "0" interpretiert.

GSD-Dateien können bis GSD-Revision 5 interpretiert werden. Damit werden DP-Slaves unterstützt, die z. B. folgende Funktionen unterstützen:

- Diagnosemeldungen für Alarmblöcke
- Taktsynchronität und Äquidistanz (Isochron-Modus)
- SYNC/FREEZE
- Uhrzeitsynchronisation für DP-Slaves

GSD-Datei installieren

Einführung

In einer GSD-Datei (Gerätestammdaten-Datei) sind alle Eigenschaften eines DP-Slaves hinterlegt. Falls Sie einen DP-Slave projektieren möchten, der nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller mitgelieferte GSD-Datei installieren. Über GSD-Dateien installierte DP-Slaves werden im Hardware-Katalog angezeigt und können dann selektiert und konfiguriert werden.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Sie haben Zugriff auf die benötigten GSD-Dateien in einem Verzeichnis auf der Festplatte.

Vorgehen

Um eine GSD-Datei zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Dateien liegen.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien eine oder mehrere Dateien aus.
4. Klicken Sie auf die Schaltfläche "Installieren". Die ausgewählten GSD-Dateien werden nun installiert.
5. Um eine Protokolldatei für die Installation zu erstellen, klicken Sie auf die Schaltfläche "Protokolldatei speichern".
   Eventuell auftretende Probleme bei der Installation lassen sich anhand der Protokolldatei nachvollziehen.
6. Klicken Sie auf "Schließen". Sie werden darüber informiert, dass die DP-Slaves aus den installierten GSD-Dateien in den Hardware-Katalog eingetragen werden. Dieser Vorgang kann ein paar Sekunden dauern.

Sie haben nun mit der Installation einer GSD-Datei den darin beschriebenen DP-Slave dem Hardware-Katalog hinzugefügt.

Ergebnis

Sie finden die neuen über GSD-Dateien installierten DP-Slaves unter "Weitere Feldgeräte > PROFIBUS DP" im Hardware-Katalog.

GSD-Dateien werden immer mit dem Projekt zusammen gespeichert, d. h. alle zur Darstellung des Gerätes relevanten Informationen (auch die Symbole) sind auch im gespeicherten Projekt verfügbar.

GSD-Datei löschen

Einführung

Sie können über GSD-Dateien installierte DP-Slaves löschen. Diese werden danach im Hardware-Katalog nicht mehr angezeigt.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Es befinden sich über GSD-Dateien installierte DP-Slaves im Hardware-Katalog.

Vorgehen

Um eine GSD-Datei zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Datei liegt.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien die zu löschende Datei aus.
4. Klicken Sie auf die Schaltfläche "Löschen".

Die ausgewählte GSD-Datei wurde gelöscht und der DP-Slave befindet sich nicht mehr im Hardware-Katalog.

Ungenutzte GSD-Dateien finden

Einführung

Wenn Sie GSD-basierte Hardware-Komponente ihrer Hardware-Konfiguration hinzufügen, dann werden unterschiedliche GSD-Dateien zu den Projektdaten hinzugefügt. Wird die GSD-basierte Hardware-Komponente aus der Konfiguration wieder gelöscht, verbleiben die GSD-Dateien im Datenbestand des Projektes. Diese GSD-Dateien werden im Projekt nicht mehr benötigt, sofern die entsprechende GSD-basierte Hardware-Komponente nicht erneut in der Hardware-Konfiguration verwendet wird. Im Projekt nicht mehr benötigte Dateien können Sie über einen Suchmechanismus finden und dann löschen.

Voraussetzung

- Es muss ein Projekt geöffnet sein.
- Das Projekt darf nicht online sein.
- Das Projekt darf nicht schreibgeschützt sein.

Vorgehen

Um ungenutzte GSD-Dateien zu finden und zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Klicken Sie im Register "GSDs im Projekt" auf die Schaltfläche "Ungenutzte GSDs finden".

   Hinweis

   Ungenutzte GSD-Dateien sind nur verfügbar, wenn mindestens eine GSD-basierte Hardware-Komponente der Konfiguration hinzugefügt und anschließend wieder aus der Konfiguration entfernt wurde. Ungenutzte GSD-Dateien in den Projektbibliotheken werden bei der Suche ebenfalls berücksichtigt. GSD-basierte Hardware-Komponenten, die noch in der Konfiguration verwendet werden, gelten nicht als ungenutzt.

   Installierte GSD-basierte Hardware-Komponenten, die im Hardware-Katalog enthalten sind, aber bisher nicht in der Hardware-Konfiguration des Projekts verwendet wurden, können keine ungenutzten GSD-Daten im Projekt abgelegt haben.

3. Wenn nicht mehr benötigte GSD-Dateien in den Projektdaten gefunden wurden, können Sie diese nun löschen. Klicken Sie dafür auf die Schaltfläche "Löschen".

Die nicht mehr benötigten GSD-Dateien werden aus den Projektdaten gelöscht. Die GSD-basierte Hardware-Komponente ist weiterhin im Hardware-Katalog verfügbar und kann bei Bedarf der Konfiguration wieder hinzugefügt werden. In diesem Fall werden die dann notwendigen GSD-Daten erneut in die Projektdaten kopiert.

Hinweis

Wenn Sie eine GSD-basierte Hardware-Komponente über "Extras > Gerätebeschreibungsdateien (GSD) verwalten > Installierte GSDs" löschen, ist diese aus dem Hardware-Katalog gelöscht. Ungenutzte GSD-Dateien dieser Hardware-Komponente können aber immer noch ein Teil der Projektdaten sein und können über die Funktion zum Finden der ungenutzten GSD-Dateien gelöscht werden.

ACHTUNG
Nicht mehr rückgängig zu machende Aktionen Wenn Sie als Aktion die ungenutzten GSD-Dateien löschen, führt dies dazu, dass der Aktionsstapel für rückgängig zu machende Aktionen gelöscht wird. Die Aktion des Löschens von ungenutzten GSD-Dateien, sowie alle vorherigen Aktionen können dann nicht mehr rückgängig gemacht werden.

GSD-basierten DP-Slave konfigurieren

DP-Slaves, die Sie über die Installation einer GSD-Datei eingefügt haben, können Sie wie üblich über den Hardware-Katalog auswählen und in der Netzsicht einfügen. Wenn Sie die Baugruppen der GSD-basierten DP-Slaves einfügen wollen, müssen Sie einige Besonderheiten beachten.

Voraussetzung

- Sie haben einen DP-Slave über eine GSD-Datei installiert.
- Sie haben die Kopfbaugruppe in der Netzsicht wie üblich eingefügt.
- Die Geräteübersicht in der Gerätesicht ist geöffnet.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

Um die Baugruppen eines GSD-basierten DP-Slaves hinzuzufügen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu den Baugruppen des GSD-basierten DP-Slaves. GSD-basierte DP-Slaves, auch DP-Normslaves genannt, befinden sich im Hardware-Katalog im Ordner "Weitere Feldgeräte".
2. Selektieren Sie die gewünschte Baugruppe.
3. Ziehen Sie die Baugruppe mit Drag & Drop auf einen freien Platz der Geräteübersicht.
4. Selektieren Sie die Baugruppe in der Geräteübersicht, um deren Parameter zu bearbeiten.

Sie haben die Baugruppe nun auf einen freien Steckplatz des GSD-basierten DP-Slaves gesteckt und können dessen Parameter bearbeiten.

Hinweis

In dem grafischen Bereich der Gerätesicht können Sie nur den GSD-basierten DP-Slave sehen. Die hinzugefügten Baugruppen von GSD-basierten DP-Slaves finden Sie immer nur in der Geräteübersicht.

Sollkonfiguration

Bei Baugruppen mit einer einstellbaren Sollkonfiguration können Sie diese im Inspektorfenster unter "Eigenschaften > Sollkonfiguration" ändern.

1.4.9.11 PROFIBUS DPV1 anwenden

Einführung zu PROFIBUS DPV1

Allgemeines

Die DPV1-Funktionalität ist eine Erweiterung der PROFIBUS-Norm für die Dezentrale Peripherie. Gegenüber der Vorgängerfunktionalität DPV0 weist die DPV1-Funktionalität einige Erweiterungen und Vereinfachungen auf:

- PROFIBUS-DPV0 behandelt den zyklischen Austausch der Daten und die Diagnose
- PROFIBUS-DPV1 behandelt den azyklischen und zyklischen Datenaustausch zuzüglich der Alarmbehandlung.

Zusätzliche Funktionalität bei DPV1-Geräten (DP-Master/DP-Slaves)

DP-Master und DP-Slaves, die DPV1 unterstützen, haben gegenüber den nur DPV0 unterstützenden Geräten folgende, zusätzliche Funktionen:

- Der azyklische Datenaustausch zwischen Master und Slave wird unterstützt (Datensatz lesen/schreiben, z. B. um einen Slave im Betrieb umzuparametrieren). Die Datensätze einer Baugruppe und den Aufbau dieser Datensätze können Sie der Dokumentation der jeweiligen Baugruppe entnehmen.
- Alarme können von einem DPV1-Slave gestellt werden, die eine Behandlung des alarmauslösenden Ereignisses in der Master-CPU sicherstellen. Auch im Betriebszustand "STOP" werden die Alarmdaten in der CPU ausgewertet (Aktualisierung des Diagnosepuffers und des Baugruppenzustands). Eine OB-Bearbeitung findet aber im STOP nicht statt.
  Neben den aus der SIMATIC bekannten Alarmen, z. B. Diagnosealarm bei ET 200M, werden jetzt auch die Alarme Statusalarm, Update-Alarm und herstellerspezifischer Alarm unterstützt.

Siehe auch

S7-Integration von DPV1-Slaves (http://support.automation.siemens.com/WW/view/de/10259221)

DPV1-Geräte konfigurieren

Einige Geräte, z. B. die DP-Anschaltung IM 151-1 HF, können zwischen DPV0 und DPV1 umgeschaltet werden.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Master mit DPV1-Funktionalität ist vorhanden.
- Eine Master-Slave-Verbindung mit PROFIBUS ist aufgebaut.

Vorgehen

Um die Betriebsart des DP-Slaves von DPV1- auf DPV0-Funktionalität umzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den DP-Slave.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Baugruppenparameter > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" zwischen den Betriebsarten "DPV0" und "DPV1".

oder

1. Selektieren Sie den DP-Master.
2. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Verbindung zwischen dem DP-Master und dem gewünschten DP-Slave.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Baugruppenparameter > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" zwischen den Betriebsarten "DPV0" und "DPV1".

Verwendung von Geräten mit unterschiedlicher PROFIBUS-Funktionalität

Bei einer gemischten Verwendung von Geräten mit DPV0- und DPV1-Funktionalität gelten folgende Regeln:

- An einer DP-Schnittstelle mit DPV1-Funktionalität können prinzipiell auch DPV0-Slaves betrieben werden.
- An einer DP-Schnittstelle mit DPV0-Funktionalität können prinzipiell auch DPV1-Slaves betrieben werden. Die DPV1-Funktionalität wird dann automatisch abgeschaltet. Herstellerspezifische Projektierungsregeln bei bestimmten DP-Slaves können allerdings zwingend einen DPV1-Betrieb fordern, so dass Sie diese DP-Slaves nicht mit dem DP-Mastersystem betreiben können!

Zuordnung der Diagnoseadresse

Bei DPV1-Slaves wird die Diagnoseadresse automatisch dem virtuellen Steckplatz 0 als Stationsstellvertreter zugeordnet.

Generell gilt folgende Zuordnung:

- Dem virtuellen Steckplatz 0 mit seiner Diagnoseadresse werden Diagnosen und Alarme zugeordnet, die sich nur auf den gesamten DP-Slave beziehen: z. B. Alarme von Baugruppen auf nicht projektierten Steckplätzen, Stationsausfall/Stationswiederkehr (OB 86 (Seite 908)).
- Den übrigen Steckplätzen mit ihrer jeweiligen Anfangsadresse werden Diagnosen und Alarme zugeordnet, die von dieser Baugruppe ausgehen (z. B. von einer DP-Anschaltung IM 153-2 auf Steckplatz 2).

Siehe auch

Alarme im I-Slave (Seite 2905)

Programmieren von DPV1-Geräten

Für die DPV1-Funktionalität gibt es OBs und SFBs/SFCs, die nicht für DPV0 anwendbar sind.

Alarm-OBs für DPV1-Ereignisse

DPV1-Slaves können Alarme auslösen. Für Diagnosealarme, Prozessalarme, Ziehen-/Steckenalarme können Sie die entsprechenden OBs verwenden, die das Betriebssystem der S7-CPUs zur Verfügung stellt.

In der folgenden Tabelle sind die Alarm-OBs für DPV1 dargestellt. Ausführliche Informationen finden Sie in der Beschreibung zu den OBs.

DPV1-Alarm	OB	Erläuterung
Statusalarm	OB 55 (Seite 880)	Der Statusalarm kann ausgelöst werden, wenn der Betriebszustand eines Gerätes bzw. einer Baugruppe wechselt, z. B. von RUN nach STOP. Eine genaue Beschreibung der Ereignisse, bei denen ein Statusalarm ausgelöst wird, können Sie der Dokumentation des jeweiligen Herstellers des DPV1-Slaves entnehmen.
Update-Alarm	OB 56 (Seite 882)	Ein Update-Alarm kann ausgelöst werden, wenn ein Steckplatz umparametriert wurde. Das kann z. B. durch lokalen oder durch Partner-Zugriff auf die Parameter verursacht worden sein. Eine genaue Beschreibung der Ereignisse, bei denen ein Update-Alarm ausgelöst wird, können Sie der Dokumentation des jeweiligen Herstellers des DPV1-Slaves entnehmen.
Herstellerspezifischer Alarm	OB 57 (Seite 885)	Das Ereignis, das den herstellerspezifischen Alarm auslöst, kann vom Hersteller eines DPV1-Slaves festgelegt werden.

Anweisungen für Zugriffe auf DPV1-Slaves

In der folgenden Tabelle sind die DPV1-Schnittstellen mit ihren Funktionen dargestellt. Ausführliche Informationen finden Sie in der Beschreibung zu den erweiterten Anweisungen.

Funktion	Schnittstelle (DPV1)	Bemerkungen
Datensatz lesen	RDREC	-
Datensatz schreiben	WRREC	-
Alarm von einem DP-Slave empfangen	RALRM	Die Anweisung muss in dem OB aufgerufen werden, der den Alarm auslöst.

Hinweis

Wenn ein DPV1-Slave verwendet wird und die DP-Schnittstelle des DP-Masters auf "DPV0" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit den Anweisungen "WR_REC"/"RD_REC" bzw. "WRREC"/"RDREC" von den E/A-Baugruppen gelesen bzw. beschrieben werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz+3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Checkliste für die Umstellung von DPV0 nach DPV1

Folgende Abschnitte des bestehenden Anwenderprogramms sind zu beachten, wenn Sie die Konfiguration bearbeitet und die Schnittstelle von "DPV0" auf "DPV1" umgestellt haben:

Funktion	Was ist zu beachten?
Adressumrechnungen	Wenn Sie im Anwenderprogramm Adressumrechnungen verwendet haben, z. B. mit den erweiterten Anweisungen der Systemfunktionen "GADR_LGC" (SFC 5), "LGC_GADR" (SFC 49), "RD_LGADR" (SFC 50), dann müssen Sie für DPV1-Slaves die Zuordnung Steckplatz <-> logische Anfangsadresse prüfen. Zusätzlich hat Steckplatz 0 eine Adresse. DPV1-Slave: Während bisher die erste E/A-Baugruppe des DP-Slaves von den erweiterten Anweisungen dem Steckplatz 4 zugeordnet wurde, ist jetzt die erste E/A-Baugruppe dem Steckplatz 1 zugeordnet (wie in der Hardware-Konfiguration sichtbar). DPV1-Slave im Hardware-Katalog integriert (z. B. ET 200M): Anschaltungsbaugruppe (Steckplatz 2) hat eigene Adresse.
Diagnose auslesen mit der Systemfunktion "DPNRM_DG"	Die ursprünglich vergebene Diagnoseadresse funktioniert auch weiterhin. Intern wird dieser Adresse der Steckplatz 0 zugeordnet. Der Diagnosedatensatz der DPV1-Slaves ist anders aufgebaut als bei DPV0 (siehe Beschreibung der DP-Slaves, bei ET 200M z. B. auch unter dem Stichwort "erweiterte Diagnose").
Datensätze lesen/schreiben	Wenn Sie mit der erweiterten Anweisung "WR_REC" (SFC 58) Datensätze zu einem DPV1-Slave übertragen oder mit der erweiterten Anweisung "RD_REC" (SFC 59) Datensätze von einem DPV1-Slave lesen und dieser im DPV1-Mode arbeitet, bewertet der DP-Master die vom Slave erhaltene Fehlerinformation wie folgt: Liegt die Fehlerinformation in den Bereichen W#16#8000 bis W#16#80FF oder W#16#F000 bis W#16#FFFF, reicht der DP-Master die Fehlerinformation an die SFC weiter. Liegt sie außerhalb, gibt er den Wert W#16#80A2 an die SFC und suspendiert den Slave. Zur Beschreibung der von DPV1-Slaves stammenden Fehlerinformationen siehe Beschreibung der Anweisung "RALRM" (SFB 54). Weitere Informationen finden Sie in den Beschreibungen zu Sprüngen in Sprachbeschreibungen, Baustein-Hilfen und Systemattributen.
Systemzustandsliste auslesen	Wenn Sie die erweiterte Anweisung "RDSYSST" benutzen, um z. B. Baugruppenzustandsinformation oder Baugruppenträger / Stationszustandsinformation auszulesen, müssen Sie auch die geänderte Bedeutung der Steckplätze und den zusätzlichen Steckplatz 0 (s. o.) berücksichtigen.

Beispiel für das Auswerten von Alarminfos

Einführung

Eine dezentral angeordnete S7-Digitaleingabebaugruppe (Anfangsadresse 288) löst einen Prozessalarm aus. Im OB 40 (Seite 878) soll mittels Aufruf der erweiterten Anweisung "RALRM" die Alarmzusatzinformationen dieser Baugruppe ausgelesen werden. Es wird geprüft, ob der erste Kanal einen Prozessalarm ausgelöst hat.

Die Alarmzusatzinformationen könnten bei S7-Baugruppen auch direkt aus der Startinformation des OB 40 ausgelesen werden. Die DPV1-Funktionalität lässt bis zu 59 Bytes Alarmzusatzinformationen zu, was aber für die Startinformation des OB 40 zu viel ist.

Auswerten von Alarminfos aus OB 40 mit "RALRM"

AWL

```
// ...
// ...
//Schalter fuer alarmausloesende Adresse (288)
    L    DW#16#120
    T    "MD10"
    CALL  "RALRM" , "DB54"
      MODE  :=1                  //Funktionsmodus: 1=alle Ausgangsparameter setzen (d. h.
F_ID ohne Wirkung)
      F_ID  :="MD10"             //Anfangsadresse des Steckplatzes, von dem ein Alarm
zugelassen wird
      MLEN  :=8                  //Max. Laenge der Alarmzusatzinfo in Bytes (z. B. fuer
Kanalstatus der Baugruppe)
      NEW   :="Alarm_neu"        //Alarm empfangen ? (ja = 1)
      STATUS:="DP_RALRM_STATUS"  //Rueckgabewert mit Funktionsergebnis/Fehlermeldung
      ID    :="Slotadresse_Alarm" //Anfangsadresse des Steckplatzes, von dem ein Alarm
empfangen wurde
      LEN   :="Laenge_Alarminfo"  //Laenge der Alarmzusatzinfo (4 Byte Kopfinfo + z. B. 4
Byte bei S7-E/A-Baugruppen)
      TINFO :=P#M 100.0 BYTE 28   //Zeiger fuer OB-Startinfo+Verwaltungsinfo: 28 Bytes ab
MB 100
      AINFO :=P#M 130.0 BYTE 8    //Zeiger fuer Zielbereich der Kopfinfo + Alarmzusatzinfo
(max. 59 Bytes)
    U    M    124.0              //Ist Eingang 1 (Bit 0) der Alarmausloeser?
    SPB  Alrm
    BEA
Alrm:    S    A   0.0            // Alarmbearbeitung
// ...
```

Beispiel für das Auswerten von Diagnosedaten

Einführung

Der Zielbereich für Diagnosedaten soll ausreichen für die Standard-Diagnose (6 Bytes), für die kennungsspezifische Diagnose (3 Bytes für 12 Steckplätze) sowie für die Auswertung der gerätespezifischen Diagnose (nur Modulstatus, das sind weitere 7 Bytes). Hierzu wird die erweiterte Anweisung "RALRM" im OB 82 (Seite 894) verwendet.

Für eine weitergehende Auswertung (kanalspezifische Diagnose) wären zusätzliche Bytes zu reservieren, sofern der DP-Slave diese Funktion unterstützt.

Auswerten von Diagnosedaten im OB 82 mit "RALRM" (SFB 54)

AWL

```
// ...
// ...
      L     120                          //Anfangsadresse festlegen fuer Geraet/Baugruppe,
      T     "Slotadresse_Diag"           //von der Diagnose geholt werden soll


      CALL  "RALRM" , "DB54"
       MODE  :="Alle_Params"             // 1 = Alle Ausgangsparameter werden gesetzt
       F_ID  :="Slotadresse_Diag"       //Anfangsadresse des Steckplatzes, von dem Diagnose
geholt werden soll
       MLEN  :=20                      //Max. Laenge der Diagnosedaten in Bytes
       NEW   :="neu"                     //irrelevant
       STATUS:="RET_VAL"                 //Funktionsergebnis, Fehlermeldung
      ID    :="Slotadresse_Alarm"      //Anfangsadresse des Steckplatzes, von dem ein Alarm
empfangen wurde
       LEN   :="Laenge_Alarminfo"        //Laenge der Alarmzusatzinfo (4 Byte Kopfinfo+16
Bytes Diagnosedaten
      TINFO :=P#M 100.0 BYTE 28        //Zeiger fuer OB-Startinfo+Verwaltungsinfo: 28 Bytes
ab MB 100
       AINFO :=P#M 130.0 BYTE 20         //Zeiger auf Zielbereich, in dem die Diagnosedaten
abgelegt werden sollen

// ...
//Struktur der abgelegten Diagnosedaten:
//   MB 130 bis MB 133: Kopfinfo (Laenge, Identifier, Steckplatz)
//   MB 134 bis MB 139: Standard-Diagnose (6 Bytes)
//   MB 140 bis MB 142: Kennungsspezifische Diagnose (3 Bytes)
//   MB 143 bis MB 149: Baugruppenstatus (7 Bytes)
// ...
      U     M    141.0                   //Steckplatz 1 mit Fehler?
      SPB   stp1
      BE

stp1: L     MB   147                     //Baugruppenstatus Steckpl. 1 bis 4 holen
      UW    W#16#3                       //Stecklpl 1 herausfiltern
      L     W#16#2                       //2-Bit-Status 'wrong module' falsche Baugruppe
gesteckt
      ==I
      S     A      0.1                   //Reaktion auf falsche Baugruppe

      L     MB   147                     //Baugruppenstatus Steckpl. 1 bis 4 holen
      UW    W#16#3                       //Stecklpl 1 herausfiltern
      L     W#16#1                       //2-Bit-Status 'invalid data' ungueltige Nutzdaten
      ==I
      S     A      0.2                   //Reaktion auf ungueltige Nutzdaten
// ...
```

Steckplatzmodell der DPV1-Slaves bei I-Slaves

Einführung

Im Folgenden geht es darum, die Zuordnung der E/A-Adressen und Diagnoseadressen zu den Steckplätzen bei DPV1 sichtbar zu machen. Insbesondere die Adressen, die keine Nutzdaten transportieren, und deren Projektierung sollen betrachtet werden.

Das Steckplatzmodell bei DPV1

Bei DPV1 ist ein DP-Slave genauso wie bei DPV0 aus Steckplätzen (Slots) aufgebaut. Die Nummern der Steckplätze sind 0, 1, ...n. Der Steckplatz 0 gilt nur bei DPV1 und hat herausragende Bedeutung, weil er Stellvertreter des gesamten DP-Slaves ist.

Stellvertreter bedeutet, dass z. B. Alarme, die vom Steckplatz 0 ausgelöst wurden, vom gesamten DP-Slave kommen und nicht von einem ganz bestimmten Steckplatz innerhalb des DP-Slaves. Diagnosen, die von diesem Steckplatz kommen, sind dem DP-Slave insgesamt zugeordnet und nicht einem einzelnen Steckplatz bzw. einer einzelnen Baugruppe.

Adressen für DP-Schnittstellen

Aus Sicht der CPU gibt es für jede ihrer Schnittstellen eine separate logische Adresse. Die Adresse der jeweiligen DP-Schnittstelle wird Ihnen auf verschiedenen Wegen angezeigt:

- Bei Selektion eines Gerätes oder einer DP-Schnittstelle im Inspektorfenster unter "Eigenschaften > PROFIBUS-Adresse (Xn)" bzw. "Eigenschaften > MPI/DP-Schnittstelle > PROFIBUS-Adresse (Xn)".
- Bei aktiviertem "Adressen anzeigen" in der Funktionsleiste in der Netzsicht: Sie sehen dann für jede Schnittstelle den Namen des evtl. angeschlossenen Subnetzes und dahinter die vergebene PROFIBUS-Adresse.

Diese Adressen haben nichts mit dem Steckplatzmodell von DP-Slaves zu tun, sondern dienen lediglich zur CPU-internen Identifikation, z. B. bei einem Ausfall der Schnittstelle. Für das Anwenderprogramm ist diese Adresse von geringer Bedeutung.

Steckplätze und Adressen für Nutzdaten

Generell gilt, dass jeder Hersteller eines DP-Slaves frei bestimmen kann, welche Daten von welchem Steckplatz kommen.

Im Gegensatz zu DPV1-Slaves mit Nutzdaten ab Steckplatz 1 gilt für DPV0-Slaves die Einschränkung, dass sich die erste E/A-Baugruppe immer auf Steckplatz 4 befindet. DP-Slaves, die über eine GSD-Datei installiert werden, können Nutzdaten ab Steckplatz 1 haben. Da dezentrale Peripheriedaten in der Regel wie zentrale Peripheriedaten über ihre Adressen angesprochen werden, müssen Nutzdaten der DPV0-Slaves ab der Anfangsadresse des Steckplatzes 4 adressiert werden. Das gilt auch für intelligente DP-Slaves.

Bei intelligenten DP-Slaves können Sie E/A-Speicherbereiche der I-Slaves über Transferbereiche der "I-Slave-Kommunikation" den E/A-Speicherbereichen des DP-Masters zuordnen. Während des Betriebs (zyklischer Datenaustausch) werden dann die Daten, die Sie im Anwenderprogramm des intelligenten DP-Slaves in dessen E/A-Speicherbereich übertragen, über den Transferbereich in den E/A-Speicherbereich des DP-Masters

übertragen. Die Tabelle für die Zuordnung der E/A-Speicherbereiche in Transferbereichen finden Sie nach Auswahl der I-Slave-Verbindung in der E/A-Kommunikationstabelle im Inspektorfenster unter "Eigenschaften > I-Slave-Kommunikation". Nach Anlegen eines Transferbereichs können Sie unter "Details des Transferbereichs" die weitere Konfiguration vornehmen.

Beim Konfigurieren der Adressen unter "Details des Transferbereichs" werden auch Steckplatznummern angezeigt. Die Steckplätze werden nicht durch reale Baugruppen belegt, wie z. B. bei ET 200M, sondern durch frei bestimmbare Längen der jeweiligen E/A-Bereiche. Man spricht in diesem Fall auch von virtuellen Steckplätzen.

Wichtig für das Verständnis der Adresszuordnungen im Transferbereich ist:

- Der intelligente DP-Slave hat zusätzlich zu seinen realen Steckplätzen virtuelle Steckplätze, die sich im E/A-Speicherbereich befinden.
- Die virtuellen Steckplätze werden wie die realen Steckplätze über logische Adressen angesprochen. Im Fall eines DP-Slaves, wie z. B. ET 200M über die Anfangsadresse einer Baugruppe, beim I-Slave über die projektierte Adresse der "I-Slave-Kommunikation".
- Die Adressen der virtuellen Steckplätze sind aus Sicht des DP-Masters andere als aus Sicht des DP-Slaves. Die Zuordnung ist projektierbar. Derselbe DP-Slave-Steckplatz wird daher i. d. R. vom DP-Master unter einer anderen Adresse angesprochen als vom DP-Slave.

Beispiel für eine Adresszuordnung bei Nutzdaten

Die folgende Tabelle zeigt die Zuordnung der virtuellen Steckplätze im Transferbereich:

Beispieladresse aus Sicht des DP-Slaves	Bedeutung für DP-Slave	Steckplatz	Bedeutung für DP-Master	Beispieladresse aus Sicht des DP-Masters
		0		
		1		
		2		
		3		
E 2	Gelesen wird über Eingangsbyte 2 das, ...	4	... was der Master in das Ausgangsbyte 4 geschrieben hat.	A 4
A 5	Was im Slave auf Ausgangsbyte 5 geschrieben wurde ...	5	... kann im Master als Eingangsbyte 6 gelesen werden	E 6
E 8	...	6	...	A 8
		...		
		35		

Hinweis

Bei geöffneter Netzsicht finden Sie die Steckplatzzuordnung bei selektierter I-Slave-Verbindung in der E/A-Kommunikationstabelle im Inspektorfenster unter "I-Slave-Kommunikation > Transferbereiche".

Adressen für Systeminformationen

Adressen für Systeminformationen werden dazu genutzt, um z. B. Diagnoseinformationen oder Informationen über Betriebszustandsübergänge hantieren zu können.

Adressen des DP-Slaves

Die Systeminformationen eines DP-Slaves sind änderbaren Adressen zugeordnet. Relevant für DPV1 sind in diesem Zusammenhang folgende Adressen:

- Adressen für Gerätediagnose:
  - Der DP-Master diagnostiziert einen Ausfall bzw. die Wiederkehr des I-Slaves.
  - Der I-Slave diagnostiziert einen Ausfall bzw. die Wiederkehr des DP-Masters.
- Adressen für Betriebszustandsübergange:
  - Der DP-Master kann einen Betriebszustandswechsel des DP-Slaves feststellen.
  - Der intelligente DP-Slave kann einen Betriebszustandswechsel des DP-Masters feststellen.

Die Adressen werden automatisch von oben nach unten vergeben, damit kein Konflikt mit Nutzdaten auftritt. Die vorgeschlagenen Adressen sollten übernommen werden, auch wenn sie änderbar sind. Wenn das Anwenderprogramm auf unterschiedlichen CPUs laufen soll, dann prüfen Sie, ob die Adressen auch im Adressbereich der "kleinsten" CPU liegen.

Die folgende Tabelle zeigt die Zuordnung der Adressen in der I-Slave-Kommunikation:

Beispieladresse aus Sicht des DP-Slaves	Bedeutung für DP-Slave	Beispieladresse aus Sicht des DP-Masters	Bedeutung für DP-Master
8189	Stationsausfall / Stationswiederkehr des DP-Masters	16381	Stationsausfall / Stationswiederkehr des DP-Slaves
8188	Betriebszustands-übergang des DP-Masters	16380	Betriebszustands-übergang des DP-Slaves

Zusammenfassung

Die Konfiguration des I-Slaves sieht mit virtuellen Steckplätzen in Transferbereichen folgendermaßen aus:

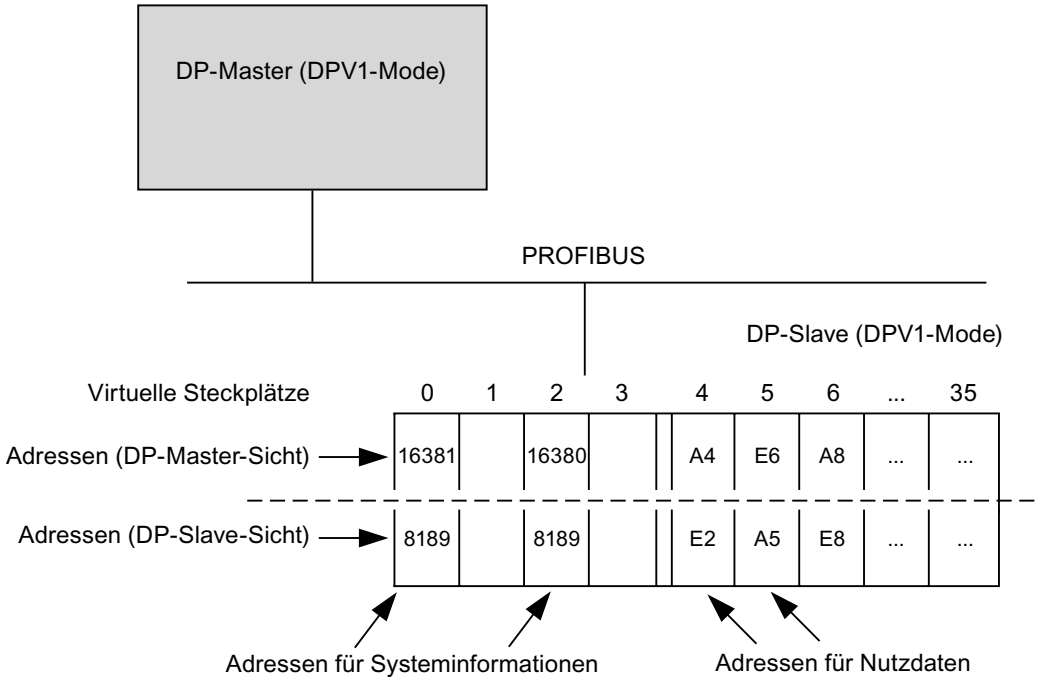

Prozessalarm auslösen mit Anweisung DP_PRAL

Mit der Systemfunktion in der erweiterten Anweisung "DP_PRAL" können Sie aus dem Anwenderprogramm der CPU des I-Slaves heraus für jede projektierte Adresse einen Prozessalarm auslösen. Das gilt sowohl für Nutzdatenadressen aus dem E/A-Bereich als auch für die Adresse des virtuellen Steckplatzes 2.

Im Anwenderprogramm des I-Slaves verwenden Sie für die Anweisung "DP_PRAL" z. B. die in der Spalte "lokal ..." projektierten E/A-Adressen.

Im Anwenderprogramm des Masters wird daraufhin ein Prozessalarm ausgelöst. Als alarmauslösende Adresse wird in der Startinformation des Prozessalarm-OB (z. B. OB 40 (Seite 878)) die Adresse mitgegeben, die Sie in der Spalte "PROFIBUS-DP-Partner" projektiert haben.

1.4.9.12 Äquidistanz und Taktsynchronität

Wissenswertes zu Äquidistanz und Taktsynchronität

Gleich lange Buszyklen bei PROFIBUS-Subnetzen

Einführung

Die PROFIBUS-Eigenschaften "Taktsynchronität" und "Äquidistanz" bilden die Grundlage für synchronisierte Bearbeitungszyklen.

Äquidistanz gewährleistet eine exakt gleich lange Zeitspanne für Buszyklen. "Gleich lange Buszyklen" bedeuten, dass der PROFIBUS DP-Master den DP-Buszyklus immer wieder nach der gleichen Zeitspanne beginnt. Die angeschlossenen Slaves erhalten deshalb ebenfalls in exakt gleich bleibenden Zeitabständen ihre Daten vom Master. Man spricht hier auch von einer "Taktung des Buszyklus".

Äquidistanz ist die Voraussetzung für Taktsynchronität.

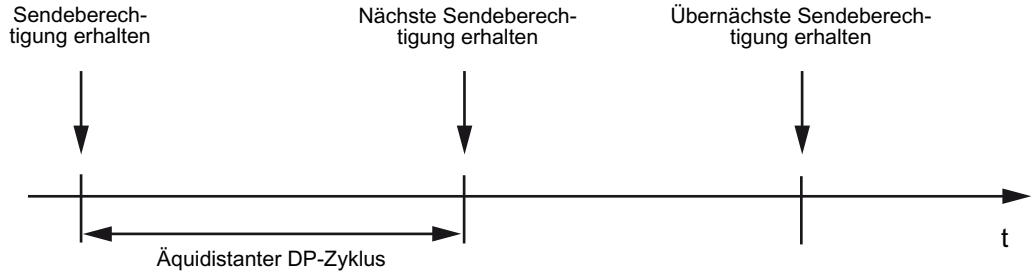

Zusammensetzung der Buszykluszeit

Das folgende Bild zeigt, wie sich die Zeit für einen Buszyklus zusammensetzt.

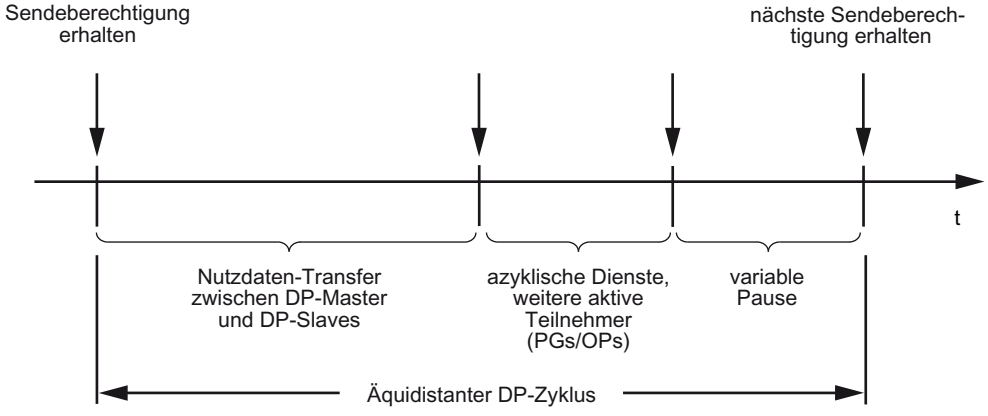

Die im Bild gezeigte "variable Pause" ist immer dann minimal, wenn noch Kommunikationsaufträge z. B. für weitere aktive Teilnehmer anstehen. Der Master, genauer der Äquidistanz-Master, steuert die Kommunikationsanteile so, dass immer die gleiche Dauer für einen Buszyklus erreicht wird.

Voraussetzungen

Folgende Voraussetzungen müssen für Äquidistanz gegeben sein:

- Der DP-Master muss die Funktion "Äquidistanz" unterstützen.
- Der Äquidistanz-Master muss ein DP-Master Klasse 1 sein. Ein PG/PC ist als Äquidistanz-Master nicht geeignet.
- Der Äquidistanz-Master ist die einzige aktive Station am PROFIBUS DP. Es darf nur ein DP-Mastersystem am PROFIBUS-Subnetz vorhanden sein. PGs bzw. PCs dürfen zusätzlich angeschlossen werden.
- Es sind keine SYNC/FREEZE-Gruppen projektiert.
- Äquidistanz ist nur bei den Busprofilen "DP" und "benutzerdefiniert" möglich.
- Es darf kein CiR (Configuration in Run) projektiert sein.
- Am PROFIBUS-Subnetz darf keine H-CPU angeschlossen sein.

Zeit für äquidistanten DP-Zyklus

Die Zeit für den äquidistanten DP-Zyklus wird automatisch errechnet:

- Aus der PROFIBUS-Projektierung (Anzahl projektierter Teilnehmer)
- Aus weiteren Informationen für die Berechnung, die optional eingegeben werden: z. B. PGs, die zu berücksichtigen sind, aber außerhalb des Projekts liegen.

Der errechnete Zeit für den äquidistanten DP-Zyklus berücksichtigt den Nutzdatenverkehr des PROFIBUS DP-Masters. Wenn Fehler auftreten, kann es zu Äquidistanzverletzungen kommen.

Die errechnete Zeit können Sie innerhalb des angezeigten Intervalls ändern.

Einfluss angeschlossener aktiver Teilnehmer (PGs/PCs und I-Slaves)

Ein PG/PC muss nur dann berücksichtigt werden, wenn es mit seiner PROFIBUS-Schnittstelle direkt am PROFIBUS angeschlossen ist. Wenn das PG über die MPI-Schnittstelle der CPU angeschlossen ist, braucht es nicht berücksichtigt zu werden, wie im folgenden Bild dargestellt ist.

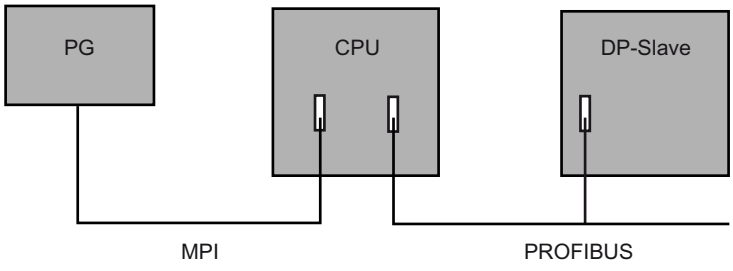

Wenn intelligente DP-Slaves, z. B. CPU 315-2 DP, angeschlossen sind, muss die Zeit für den äquidistanten DP-Zyklus großzügiger bemessen werden.

Synchronisierte Bearbeitungszyklen

PROFIBUS ermöglicht Taktsynchronität

Die wesentliche Grundlage für synchronisierte Bearbeitungszyklen bildet der äquidistante (isochrone) PROFIBUS. Er stellt eine Grundtaktung als Basis zur Verfügung. Durch die Systemeigenschaft "Taktsynchronität" wird eine Automatisierungslösung an den äquidistanten PROFIBUS gekoppelt.

Dies bedeutet:

- Das Einlesen der Eingangsdaten wird mit dem DP-Zyklus synchronisiert; alle Eingangsdaten werden zum selben Zeitpunkt eingelesen.
- Das Anwenderprogramm zur Bearbeitung der Peripheriedaten wird über die Taktsynchronalarm-OBs OB 61 bis OB 64 mit dem DP-Zyklus synchronisiert.
- Die Ausgabe der Ausgangsdaten wird mit dem DP-Zyklus synchronisiert; alle Ausgangsdaten werden gleichzeitig wirksam.
- Alle Ein- und Ausgangsdaten werden konsistent übertragen. Dies bedeutet, dass alle Daten des Prozessabbildes logisch und zeitlich zusammengehören.

Synchronisierung der Einzelzyklen

Das folgende Bild zeigt für einen Bearbeitungszyklus aus einer Reihe von taktsynchronen Einzelzyklen die Bandbreite möglicher Reaktionszeiten, die sich aufgrund der Synchronisierung des Eingangsdaten-Einlesens ergibt.

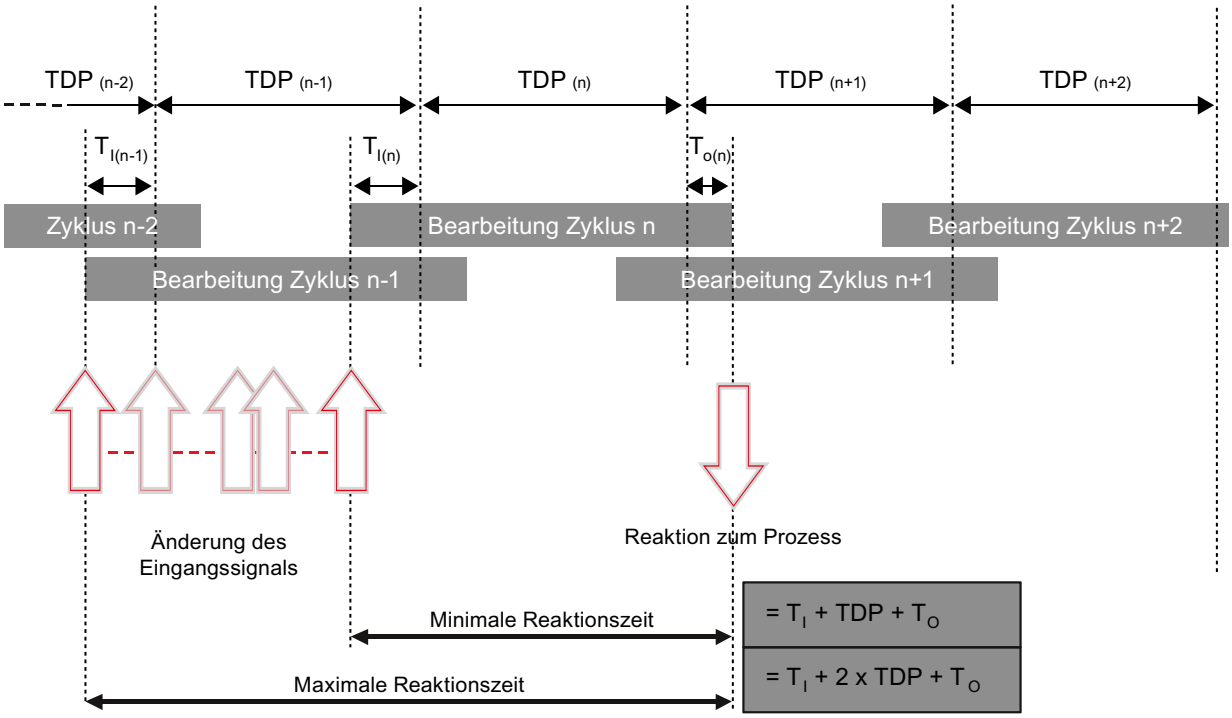

TDP	Systemtakt
Ti	Zeitpunkt des Einlesens der Eingangsdaten
To	Zeitpunkt der Ausgabe der Ausgangsdaten

Durch die Synchronisierung der Einzelzyklen wird es möglich, im Takt "n-1" die Eingangsdaten zu lesen, im Takt "n" die Daten zu übertragen und zu bearbeiten und zu Beginn des Takts "n +1" die errechneten Ausgangsdaten zu übertragen und auf die "Klemmen" zu schalten. Hiermit ergibt sich eine reale Prozessreaktionszeit von "Ti +TDP +To" bis zu "Ti + 2xTDP + To".

Mit der Systemeigenschaft "Taktsynchronität" sind die Systemdurchlaufzeiten der SIMATIC konstant; die SIMATIC ist über den Bus streng deterministisch.

Just in time

Die schnelle und zuverlässige Reaktionszeit einer Taktsynchronität begründet sich darauf, dass alle Daten just in time zur Verfügung gestellt werden. Der äquidistante (isochrone) DP-Zyklus gibt hierfür den Takt vor.

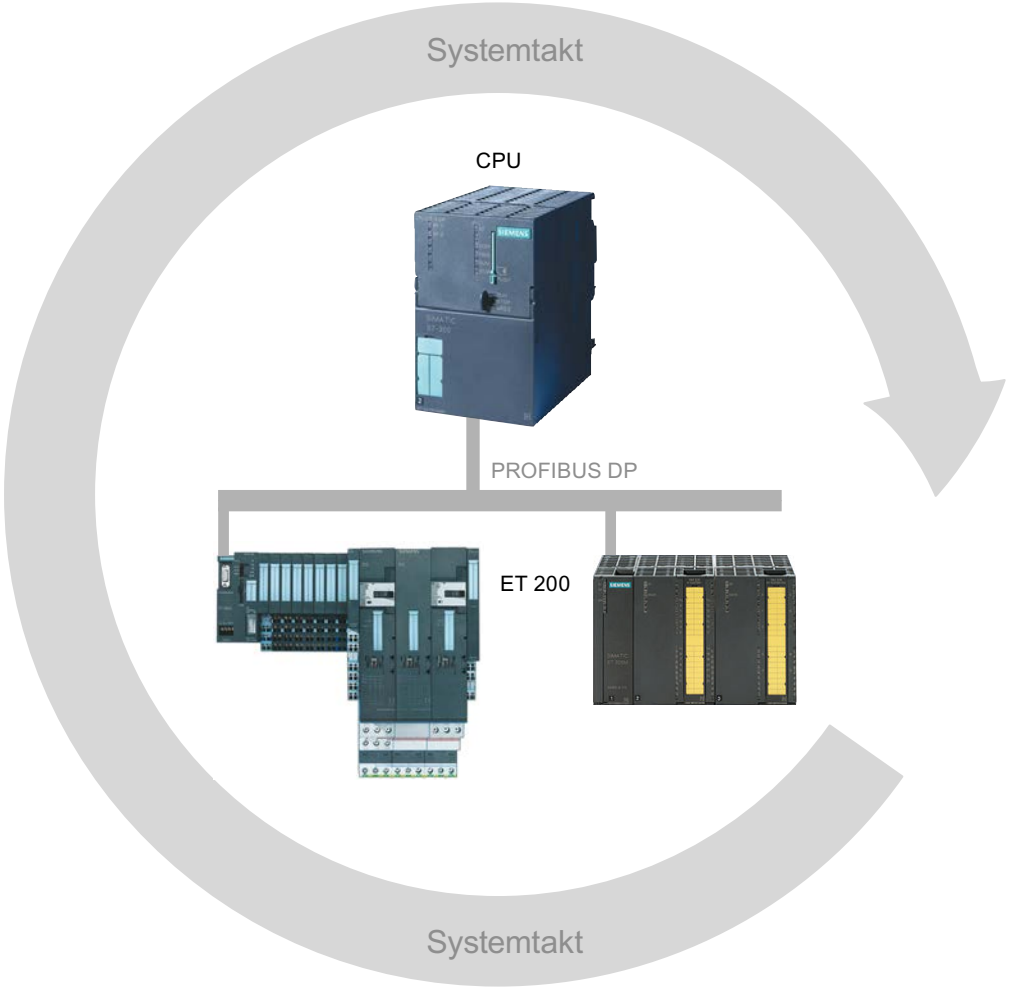

Vorlaufzeit Ti

Damit zum jeweils nächsten Beginn des DP-Zyklus alle Eingangsdaten zum Transport über den DP-Strang bereitstehen, wird der Peripherie-Einlesezyklus um eine Vorlaufzeit Ti früher gestartet. Die Vorlaufzeit Ti wird vom Anwender projektiert oder automatisch festgelegt.

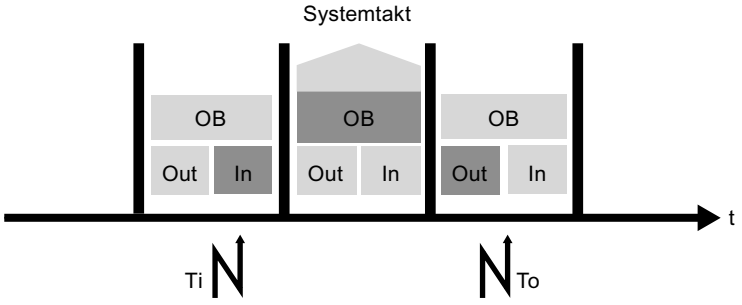

Verarbeitung der Eingangsdaten

Der PROFIBUS transportiert die Eingangsdaten über den DP-Strang zum DP-Master. Der Taktsynchronalarm-OB (OB 61, OB 62, OB 63 bzw. OB 64) wird aufgerufen. Das Anwenderprogramm im Taktsynchronalarm-OB bestimmt die Prozessreaktion und stellt die Ausgangsdaten rechtzeitig bis zum Beginn des nächsten DP-Zyklus bereit. Sie können die Länge des DP-Zyklus automatisch errechnen lassen oder selbst projektieren.

Bereitstellung der Ausgangsdaten

Für die Ausgangsdaten gilt:

- Bereitstellung rechtzeitig zu Beginn des nächsten DP-Zyklus
- Übermittlung über den DP-Strang an die DP-Slaves
- Weitergabe taktsynchron, d. h. zeitgleich zum Zeitpunkt To an den Prozess

Als reproduzierbare Reaktionszeit von der Eingangsklemme zur Ausgangsklemme ergibt sich hiermit die Summe "Ti + 2xTDP + To".

Merkmale der Taktsynchronität

Die Taktsynchronität wird durch drei wesentliche Merkmale charakterisiert:

- Das Anwenderprogramm ist mit der Peripheriebearbeitung synchronisiert, d. h. alle Vorgänge sind zeitlich aufeinander abgestimmt. Alle Eingangsdaten werden zu einem definierten Zeitpunkt erfasst. Ebenso werden die Ausgangsdaten zu einem definierten Zeitpunkt wirksam. Die Ein- und Ausgangsdaten sind bis zur Klemme auf den Systemtakt synchronisiert.
  Die Daten eines Taktes werden immer im nächsten Takt verarbeitet und im darauf folgenden Takt an den Klemmen wirksam.
- Die Ein- und Ausgangsdaten werden äquidistant (isochron) bearbeitet, d.h. Eingangsdaten werden immer in gleichen Zeitabständen eingelesen und Ausgangsdaten werden immer in gleichen Zeitabständen ausgegeben.
- Alle Ein- und Ausgangsdaten werden konsistent übertragen, d. h. alle Daten des Prozessabbildes gehören logisch und zeitlich zusammen.

Beispiel für die taktsynchrone Erfassung mehrerer Messstellen

Einführung

Mit der Systemeigenschaft "Taktsynchronität" lassen sich Messwerte und Prozessdaten in einem festen Systemtakt erfassen. Im gleichen Systemtakt erfolgt die Signalverarbeitung bis zum Schalten auf die Ausgangsklemme. Die Taktsynchronität trägt somit zu einer hohen Güte der Steuerung und damit zu einer größeren Fertigungsgenauigkeit bei. Mit der Taktsynchronität werden die möglichen Schwankungen der Prozessreaktionszeiten drastisch reduziert. Die zeitlich gesicherte Bearbeitung kann für einen höheren Maschinentakt genutzt werden.

Grundsätzlich bietet sich die Taktsynchronität überall dort an, wo Messwerte synchron erfasst werden müssen, Bewegungen koordiniert und Prozessreaktionen definiert und gleichzeitig erfolgen müssen wie im folgenden Beispiel.

Aufgabe

Innerhalb des Produktionsprozesses von Nockenwellen müssen diese zur Qualitätssicherung präzise vermessen werden.

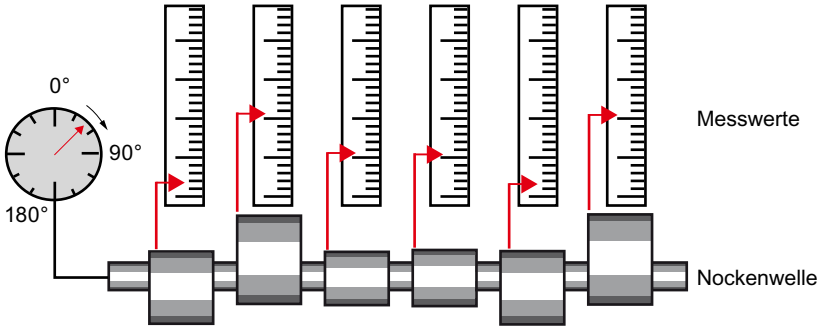

Herkömmlicher Arbeitsablauf

1. Nockenwelle drehen
2. Anhalten
3. Winkelposition messen
4. Alle Nockenauslenkungen messen
5. Weiterdrehen, Anhalten, Messen ...

Arbeitsablauf mit Taktsynchronität

Mit dem Einsatz der Systemeigenschaft "Taktsynchronität" und der damit verbundenen Gleichzeitigkeit der Messwerterfassung, kann der Messvorgang kontinuierlich erfolgen; der Zeitaufwand für den Messvorgang wird verkürzt. Dies führt zu folgendem Arbeitsablauf:

1. Nockenwelle kontinuierlich drehen
2. Während der kontinuierlichen Drehung synchron Positionen und Nockenauslenkung messen
3. Nächste Nockenwelle bearbeiten

Bei einer einzigen Drehung der Nockenwelle werden somit synchron alle Positionen der Nockenwelle und die zugehörigen Messwerte (im Bild rot) gemessen. Der Maschinentakt erhöht sich bei gleicher oder besserer Messgenauigkeit.

Regeln, Voraussetzungen und Randbedingungen

Was für die Projektierung von Taktsynchronität zu beachten ist

- Die Taktsynchronität ist nicht an optischen PROFIBUS-Netzen einsetzbar.
- Äquidistanz und Taktsynchronität ist nur mit den Busprofilen "DP" und "Benutzerdefiniert" möglich.
- Die Taktsynchronität ist nur mit den in der CPU integrierten DP-Schnittstellen möglich. Der taktsynchrone Betrieb mit CPs für PROFIBUS ist nicht möglich.
- Am taktsynchronen PROFIBUS-DP ist nur der Äquidistanz-Master als aktive Station erlaubt. OPs und PGs (bzw. PCs mit PG-Funktionalität) beeinflussen das Zeitverhalten des äquidistanten DP-Zyklus und werden deshalb nicht empfohlen.
- Eine strangübergreifende Taktsynchronität ist nicht möglich.
- Die Adressen der taktsynchronen Peripherie müssen im Adressbereich der Teilprozessabbilder liegen, da die taktsynchrone Peripherie nur in Teilprozessabbildern verarbeitet werden kann. Ohne den Einsatz von Teilprozessabbildern ist keine taktsynchrone konsistente Datenübertragung möglich. Die Einhaltung von Mengengerüsten wird überwacht, da Slave-Anzahl und Anzahl Bytes am DP-Mastersystem pro Teilprozessabbild begrenzt sind.
- Auch nicht-taktsynchrone Peripherie darf zusammen mit der taktsynchronen Peripherie im selben Teilprozessabbild liegen. Es muss ein taktsynchrones Modul/Submodul im Teilprozessabbild liegen.

- Eine volle Taktsynchronität von "Klemme" zu "Klemme" ist nur dann möglich, wenn alle in der Kette beteiligten Komponenten die Systemeigenschaft "Taktsynchronität" unterstützen. Achten Sie bei der Auswahl im Katalog auf den Eintrag "Taktsynchronität" bzw. "taktsynchrone Bearbeitung" im Informationsfeld der Baugruppe.
- Wenn Sie Taktsynchronität projektieren, dürfen Sie dem Slave keine SYNC/FREEZE-Gruppe zuordnen.

Siehe auch

DP-Slave SYNC/FREEZE-Gruppe zuordnen (Seite 2887)

Handbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045)

Konfigurieren und Parametrieren der Taktsynchronität

Zeitverhalten durch Auswahl geeigneter Topologie optimieren

Einfluss der DP-Topologie

Das zeitliche Verhalten der Taktsynchronität kann durch die Wahl einer geeigneten DP-Topologie positiv beeinflusst werden.

Trennung von taktsynchroner und nicht taktsynchroner Peripherie

Es ist möglich, taktsynchrone dezentrale Peripherie mit nicht taktsynchroner dezentraler Peripherie an einem DP-Mastersystem zu kombinieren.

Allerdings ist für eine gute Performance eine Aufteilung von taktsynchroner und nicht taktsynchroner Peripherie auf unterschiedliche DP-Mastersysteme zu empfehlen.

Werden TDs, OPs oder PGs eingesetzt, so sollten Sie diese nicht am taktsynchronen DP-Mastersystem betreiben.

Zweckmäßige Verteilung der Peripheriebaugruppen

Um die Zeiten für Ti und To möglichst kurz zu halten, sollte bei der Planung der DP-Topologie auf eine zweckmäßige Verteilung der Peripheriebaugruppen auf die Interfacemodule (ET 200M, ET 200S) geachtet werden.

Hinweis

Variieren Sie verschiedene Konfigurationen und lassen Sie sich das zeitliche Verhalten errechnen. Wählen Sie die günstigste Konfiguration aus.

Weitere Informationen

Weitere Hinweise für die Optimierung der Äquidistanzzeit finden Sie in den Betriebsanleitungen der DP-Slave-Familien ET 200M und ET 200S.

Äquidistanz und Taktsynchronität parametrieren

Äquidistanz und Taktsynchronität konfigurieren

Konfiguration für Taktsynchronität

Eine Konfiguration für Taktsynchronität besteht aus folgenden taktsynchronen Komponenten:

- CPU mit integrierter DP-Schnittstelle (z. B. CPU 319-3 PN/DP)
- DP-Anschaltungen (z. B. ET 200S-Anschaltung IM 151-1 High Feature)
- Dezentrale Ein-/Ausgabemodule (z. B. DI 2xDC24V, High Feature [131-4BB00], DO 2xDC24V/2A, High Feature [132-4BB30])

Eine aktuelle Liste taktsynchroner Komponenten finden Sie im Service & Support Portal (http://support.automation.siemens.com/WW/view/de/14747677).

Vorgehen

Um Äquidistanz und Taktsynchronität zu konfigurieren, gehen Sie grundsätzlich folgendermaßen vor:

1. Konfigurieren Sie eine taktsynchron betreibbare CPU und stellen Sie die Eigenschaften für Taktsynchronität ein:
   - Taktsynchronalarm
   - Zugeordnetes DP-Mastersystem
   - Teilprozessabbild
   - Verzögerungszeit
2. Konfigurieren Sie taktsynchrone DP-Slaves mit taktsynchronen Baugruppen und stellen Sie die Eigenschaften für Taktsynchronität zentral bei den PROFIBUS-Eigenschaften bzw. bei den einzelnen DP-Slaves (Slave-Eigenschaften) ein. Hier sind es im wesentlichen die Ti/To-Werte und wie sie berechnet werden.

   Hinweis

   Alle Eigenschaften für die taktsynchrone Peripherie können Sie zentral in den PROFIBUS-Eigenschaften einstellen. Hier haben Sie auch gleichzeitig die Übersicht über alle Slaves und deren Einstellungen am PROFIBUS.

3. Schreiben Sie das Anwenderprogramm mit Zugriffen auf die taktsynchrone Peripherie.

Eine detaillierte Beschreibung der Vorgehensweise finden Sie in den nächsten Abschnitten.

Siehe auch

CPU-Eigenschaften einstellen (Seite 2947)

PROFIBUS-Eigenschaften einstellen (Seite 2948)

Taktsynchronität von DP-Slaves einstellen (Seite 2951)

Anwenderprogramm erstellen (Seite 2951)

CPU-Eigenschaften einstellen

Die CPU-Einstellungen für Taktsynchronalarme bewirken die taktsynchrone Anbindung des Anwenderprogramms über die dafür vorgesehenen Organisationsbausteine (z. B. Taktsynchronalarm-OBs).

Voraussetzung

Die CPU muss Taktsynchronität unterstützen und die Randbedingungen schließen die Taktssynchronität nicht aus.

Vorgehen

Um die CPU-Eigenschaften einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Gerätesicht oder in der Netzsicht.
2. Wählen Sie die CPU-Eigenschaften "Taktsynchronalarme".
3. Folgende Einstellungen müssen Sie für jeden Taktsynchronalarm-OB vornehmen:
   - verwendetes DP-Mastersystem
   - gewünschtes Teilprozessabbild bzw. die gewünschten Teilprozessabbilder
   - Verzögerungszeit

Weitere Informationen zu den Teilprozessabbildern

Die zugeordneten Teilprozessabbilder müssen im Taktsynchronalarm-OB mit Aufruf der erweiterten Anweisungen "SYNC_PI" (SFC 126) und "SYNC_PO" (SFC 127) aktualisiert werden. Der Aufruf von "SYNC_PI" aktualisiert das Teilprozessabbild der Eingänge, der Aufruf von "SYNC_PO" das Teilprozessabbild der Ausgänge.

Wenn die dezentralen Peripherieadressen verschiedenen Teilprozessabbildern zugeordnet wurden, so müssen Sie die Nummern der verwendeten Teilprozessabbilder durch Kommata getrennt eingeben (z. B. 1,2,3).

Weitere Informationen zur Verzögerungszeit

- Die Verzögerungszeit ist die Zeit zwischen dem Global-Control-Telegramm und dem Start des OB 6x. In dieser Zeit wickelt der DP-Master den zyklischen Datenaustausch mit den DP-Slaves ab.

  Hinweis

  Empfehlung: Neben dem Feld zur Eingabe der Verzögerungszeit befindet sich das Optionskästchen "automatisch minimal". Wenn Sie dieses Optionskästchen aktiviert haben, brauchen Sie nach Konfigurationsänderungen die Verzögerungszeit nicht neu einzustellen!

- Die Verzögerungszeit kann nur dann eingestellt werden, wenn das Optionskästchen "automatisch minimal" deaktiviert ist.

  Hinweis

  Bei einer Verzögerungszeit von 0 startet der OB 6x direkt mit dem Global-Control-Telegramm. Ein Zugriff auf die taktsynchrone Peripherie mittels der erweiterten Anweisung "SYNC PI" (SFC 26) wird in diesem Fall mit hoher Wahrscheinlichkeit fehlschlagen, da die Peripheriedaten zur Startzeit des OBs noch nicht zur Verfügung stehen.

  Wird eine zu große Verzögerungszeit gewählt, wäre der Zugriff auf die taktsynchrone Peripherie schon früher möglich gewesen. Eine zu große Verzögerungszeit führt zu einer unnötig hohen OB-Laufzeit.

Siehe auch

Äquidistanz und Taktsynchronität konfigurieren (Seite 2946)

Regeln, Voraussetzungen und Randbedingungen (Seite 2944)

PROFIBUS-Eigenschaften einstellen

Einführung

Um die Eigenschaften der Taktsynchronität am PROFIBUS zu bearbeiten, müssen Sie die Einstellungen am Subnetz/DP-Mastersystem ändern.

Die Äquidistanz für den taktsynchronen Betrieb in einem äquidistanten PROFIBUS-Subnetz wird unter folgenden Bedingungen automatisch aktiviert:

- Zuordnung eines DP-Mastersystems zu dem Taktsynchronalarm-OB in den CPU-Eigenschaften für Taktsynchronalarme.
- Einstellung des taktsynchronen Betriebs für einen DP-Slave am PROFIBUS.

Nur wenn Sie Zeiten manuell ändern wollen, müssen Sie die Eigenschaften des PROFIBUS bearbeiten. Ansonsten werden alle Zeiten automatisch ermittelt.

Voraussetzung

Äquidistantes PROFIBUS-Subnetz.

Vorgehen

Um die Eigenschaften der Taktsynchronität am PROFIBUS zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht das PROFIBUS-Subnetz und wählen Sie im Inspektorfenster unter "Eigenschaften" die "Äquidistanz".

   Hinweis

   Achten Sie darauf, dass Sie nicht ein hervorgehobenes Mastersystem selektieren. Nur wenn das PROFIBUS-Subnetz selektiert ist, sind auch die PROFIBUS-Eigenschaften und damit auch die Taktsynchronitätseigenschaften bedienbar.

2. Stellen Sie die Eigenschaften für die Taktsynchronität ein. Erläuterungen zu den einzelnen Einstellungen finden Sie in den nächsten Abschnitten.

Eigenschaften für Taktsynchronität nicht wählbar

Wenn die Eigenschaften für Taktsynchronität nicht wählbar sind, prüfen Sie folgendes:

- Anschluss nur eines DP-Masters am PROFIBUS-Subnetz. Bei mehreren angeschlossenen DP-Mastern ist keine Taktsynchronität möglich.
- Unterstützung des taktsynchronen Betriebs durch den DP-Master (z. B. eine CPU mit integrierter DP-Schnittstelle).
- Konfigurierbarkeit der SYNC/FREEZE-Gruppen. SYNC/FREEZE-Gruppen und Taktsynchronität schließen sich gegenseitig aus.

DP-Zykluszeit und Ti/To-Werte manuell einstellen

Um die DP-Zykluszeit manuell einstellen zu können, deaktivieren Sie das Optionskästchen "DP-Zykluszeit automatisch minimal". Sie können dann einen höheren Wert als die automatisch berechnete minimale DP-Zykluszeit eingeben.

Um Ti- und To-Werte manuell einstellen zu können, deaktivieren Sie das Optionskästchen "Ti/To-Werte automatisch".

Taktsynchronität und Ti/To-Einstellungen in der Detailübersicht

Die Detailübersicht erlaubt Ihnen, für jeden geeigneten Slave bzw. für jede geeignete Baugruppe im Slave folgende Eigenschaften einzustellen:

- Taktsynchroner Betrieb
- Ti/To-Werte

Die Detailübersicht ist eine Tabelle. Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen, z. B. die Werte für TDPMin anzeigen.

Das Optionskästchen "Taktsynchroner Betrieb" für eine CPU (PLC) kann nicht manuell aktiviert bzw. deaktiviert werden. Das Optionskästchen wird automatisch aktiviert, wenn in den CPU-Eigenschaften, Abschnitt "Alarme", dem DP-Mastersystem ein Taktsynchronalarm zugeordnet wird.

Einstellmöglichkeiten für die Berechnung der Ti/To-Werte in der Detailübersicht

Für die Berechnung der Ti/To-Werte gibt es folgende Einstellmöglichkeiten:

- Vom Subnetz
  Wenn Sie den taktsynchronen Betrieb aktivieren, dann ist automatisch die Option Ti/To-Werte "Vom Subnetz" voreingestellt. Bei dieser Option haben alle DP-Slaves, die ihre Ti/To-Werte vom Netz beziehen, identische Ti/To-Zeiten; und zwar die vom PROFIBUS-Subnetz.
- Automatisch minimal
  Wenn Sie z. B. die DP-Zykluszeit manuell ändern oder Ti/To-Werte anderer Slaves manuell ändern und diese Änderung Rückwirkungen auf die Ti/To-Zeiten des Slaves hat, werden die notwendigen Anpassungen automatisch vorgenommen. Bei dieser Option können die Ti/To-Werte der Slaves voneinander abweichen.
- Manuell
  Mit dieser Option haben Sie die Möglichkeit, Ti/To-Werte für den Slave manuell einzugeben. Bei dieser Einstellung können Fehler auftreten:
  - Zeiten zu klein gewählt
  - Ti/To-Werte anderer DP-Slaves können aufgrund Ihrer Einstellung nicht mehr automatisch berechnet werden

Anzeige "-" bei Zeit Ti (To)

Wenn für Ti bzw. To kein Wert für das PROFIBUS-Subnetz angezeigt wird (Anzeige "-"), dann hat das folgende Ursache: Zur automatischen Berechnung der Ti/To-Zeit für das PROFIBUS-Subnetz wird die Schnittmenge aller Ti/To-Wertebereiche aller DP-Slaves herangezogen. Aus dieser Schnittmenge wird der kleinste Wert gewählt. Wenn keine Schnittmenge gebildet werden kann, wird "-" angezeigt. In diesem Fall kann ein DP-Slave kein Ti/To-Wert vom Netz beziehen!

Siehe auch

Äquidistanz und Taktsynchronität konfigurieren (Seite 2946)

Überlappung von Ti und To (Seite 2952)

Taktsynchronität von DP-Slaves einstellen (Seite 2951)

DP-Slave SYNC/FREEZE-Gruppe zuordnen (Seite 2887)

Taktsynchronität von DP-Slaves einstellen

Einstellungen am DP-Slave

Bis auf den äquidistanten DP-Zyklus und die Ti/To-Zeiten für das PROFIBUS-Subnetz können Sie dieselben Einstellungen in den DP-Slave-Eigenschaften vornehmen wie in den PROFIBUS-Eigenschaften.

Vorgehen

Um die Taktsynchronität für Baugruppen einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht den DP-Slave, dessen Eigenschaften Sie bearbeiten wollen.
2. Passen Sie unter "Eigenschaften > Taktsynchronisation" die Parameter Ihren Anforderungen an.

Ti/To-Zeiten für alle Slaves gleich einstellen

Das voreingestellte Verhalten nach Einfügen von DP-Slaves in die Netzsicht ist: Ti/To-Werte werden vom Netz bezogen und sind daher automatisch gleich.

Sie können bei einer bearbeiteten Konfiguration die Ti/To-Zeiten entweder manuell gleich einstellen oder alle Slaves so einstellen, dass sie ihre Ti/To-Werte vom Netz beziehen.

Siehe auch

Äquidistanz und Taktsynchronität konfigurieren (Seite 2946)

PROFIBUS-Eigenschaften einstellen (Seite 2948)

Anwenderprogramm erstellen

Vorgehen

Um das Anwenderprogramm für die Taktsynchronität zu erstellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie die benötigten Taktsynchronalarm-OBs (z. B. den OB 61).
2. Rufen Sie im Taktsynchronalarm-OB folgende erweiterten Anweisungen zum Aktualisieren des Teilprozessabbilds auf:
   - "SYNC_PI" (SFC 126) zum Aktualisieren des Teilprozessabbilds der Eingänge
   - "SYNC_PO" (SFC 127) zum Aktualisieren des Teilprozessabbilds der Ausgänge

Als Teilprozessabbild ist das in der CPU parametrierte Teilprozessabbild zu verwenden (Parameterbereich "Taktsynchronalarme").

Hinweis

Insbesondere bei sehr kurzen DP-Zykluszeiten kann folgende Situation eintreten: Die Laufzeit des Anwenderprogramms (OB 6x mit aufgerufenen SYNC_PI/SYNC_PO) ist größer als der kleinste Takt (siehe Technische Daten der CPU, Abschnitt "Taktsynchronität"). In diesem Fall müssen Sie die automatisch berechnete DP-Zykluszeit manuell vergrößern.

Siehe auch

Äquidistanz und Taktsynchronität konfigurieren (Seite 2946)

Überlappung von Ti und To

Überlappung von Ti und To

Verkürzung des Systemtakts und der Prozessreaktionszeit

Mit einer Überlappung von Ti und To wird eine Verkürzung des Systemtakts TDP und der Prozessreaktionszeit erreicht, d. h. die Zeit vom Auftreten und Erfassen eines Ereignisses über die Verarbeitung bis zur Ausgabe einer Reaktion an den Ausgängen wird kürzer.

Voraussetzung für die Überlappung von Ti und To

Für die Überlappung von Ti und To muss folgende Voraussetzung erfüllt sein:

ET 200M: Interfacemodul IM 153-2; ab Artikelnummer 6ES7 153-2BAx2-0XB0

ET 200S: Interfacemodul IM 151-1; ab Artikelnummer 6ES7 151-2BAx2-0XB0

Hinweis

Baugruppen, die sowohl Eingangswandelzeiten (TWE) als auch Ausgangswandelzeiten (TWA) besitzen (ggf. abhängig von der Betriebsart der Ein-/Ausgabebaugruppe), können nicht taktsynchron mit Überlappung von Ti und To betrieben werden. Die Überlappung von Ti und To ist nur möglich, wenn diese Baugruppen **nicht** für den taktsynchronen Betrieb aktiviert werden.

Siehe auch

PROFIBUS-Eigenschaften einstellen (Seite 2948)

Funktionsprinzip der Überlappung von Ti und To (Seite 2953)

Einstellen der Taktsynchronität mit Überlappung von Ti und To (Seite 2954)

Funktionsprinzip der Überlappung von Ti und To

Überlappende Bearbeitung

Das Funktionsprinzip bei der Überlappung von Ti und To besteht darin, dass die Peripherie-Eingabebaugruppe die Eingänge bereits einliest, während die Peripherie-Ausgabebaugruppe die Prozessreaktionen des Anwenderprogramms an die Ausgänge noch durchschaltet. Das führt zu einer Verkürzung des Systemtakts TDP und somit zu kürzeren Prozessreaktionszeiten.

DP-Buszyklus ohne Überlappung von Ti und To

Das folgende Bild stellt einen äquidistanten DP-Buszyklus mit Taktsynchronität, aber ohne Überlappung von Ti und To, dar.

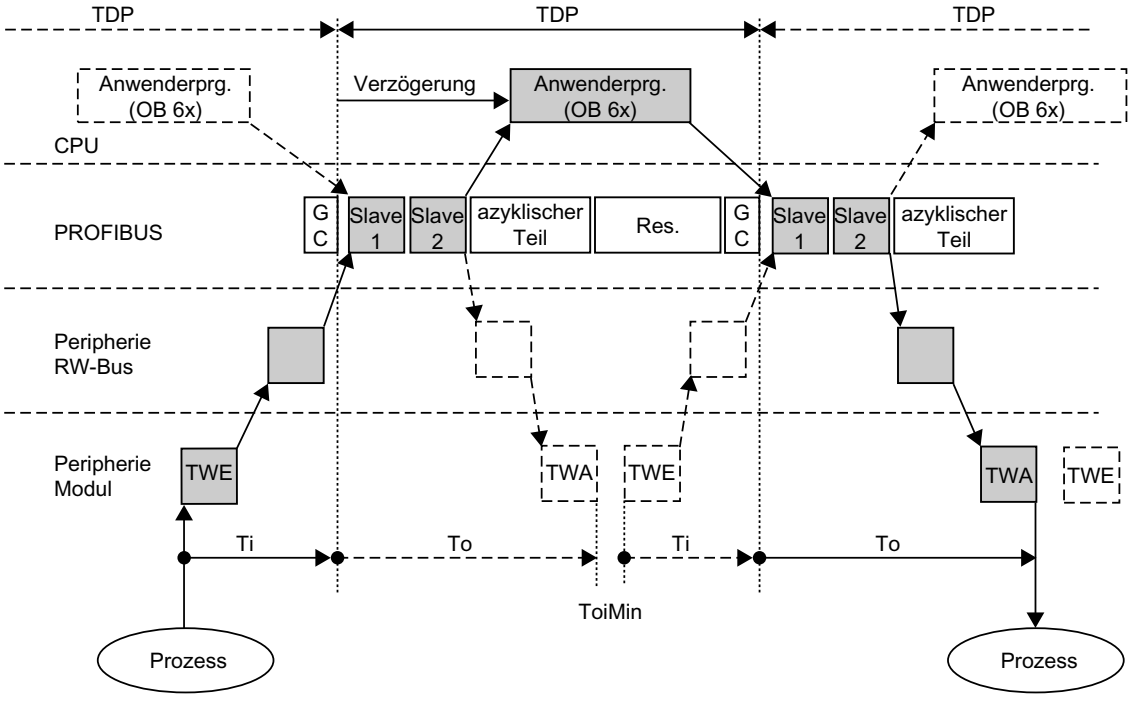

DP-Buszyklus mit Überlappung von Ti und To

Das folgende Bild stellt einen äquidistanten DP-Buszyklus mit Taktsynchronität und Überlappung von Ti und To dar.

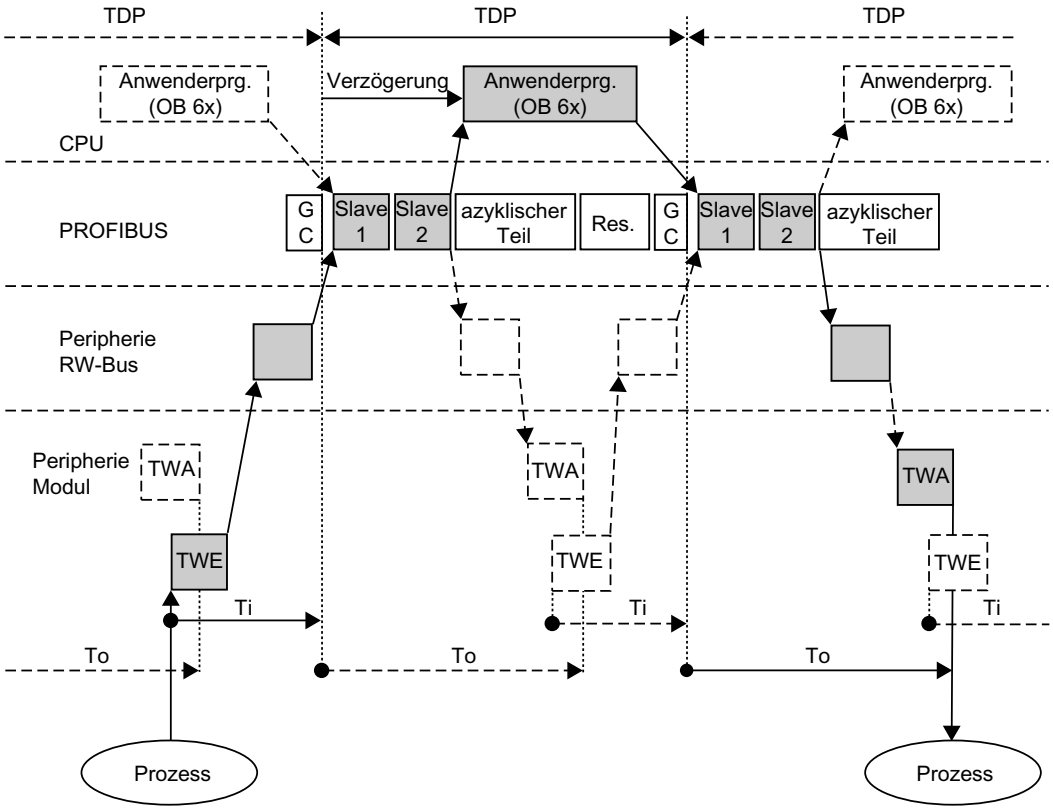

Regeln für Ti, To und TDP

Folgende Regeln gelten für Ti, To und TDP:

- Ohne Überlappung ist Ti + To ≤ TDP, anders ausgedrückt: TDP ≥ Ti + To
- Mit Überlappung kann Ti + To > TDP bzw. TDP < Ti + To sein

Siehe auch

Überlappung von Ti und To (Seite 2952)

Einstellen der Taktsynchronität mit Überlappung von Ti und To

Taktsynchronität mit Überlappung von Ti und To einstellen

Für die Taktsynchronität mit Überlappung von Ti und To müssen Sie auf folgendes achten:

Der DP-Slave, bei dem Ti/To-Überlappung möglich ist, darf seine Ti/To-Werte nicht vom PROFIBUS-Subnetz beziehen. Wählen Sie für diesen DP-Slave z. B. die Option "Ti/To-Werte automatisch".

Es werden automatisch die bestmöglichen Werte für Ti und To eingestellt, abhängig vom aktuell eingestellten äquidistanten DP-Zyklus.

Regeln für Ti und To bei Überlappung

Bei der Taktsynchronität mit Überlappung gelten folgende Regeln für Ti und To:

- Ti ≤ TDP
- To ≤ TDP

Reserven auffinden

Zum Auffinden von Reserven können Sie die zeitlichen Abläufe mit Hilfe einer Berechnungstabelle (http://support.automation.siemens.com/WW/view/de/23876584) visualisieren und mit dem Ergebnis die Abläufe in STEP 7 entsprechend optimieren.

Hinweis

Haftungsausschluss

Die Siemens AG übernimmt keine Haftung für Schäden, die durch die Benutzung der Berechnungstabelle entstehen; insbesondere nicht für Personen-, Sach- oder Vermögensschäden, die als Folgeschäden unmittelbar oder mittelbar im Zusammenhang mit der Nutzung der Berechnungstabelle in Verbindung stehen.

Für die Anwendung der Berechnungstabelle leisten wir keinen Support.

Hinweis

Beachten Sie die Regeln für die Optimierung der Äquidistanzzeit auch für die Überlappung von Ti und To!

Weitere Informationen

Weitere Informationen für die Optimierung der Äquidistanzzeit finden Sie in der Betriebsanleitung ET 200M und in der Betriebsanleitung ET 200S.

Siehe auch

Überlappung von Ti und To (Seite 2952)

1.4.10 PROFIBUS DP konfigurieren

1.4.10.1 Grundlagen zum Konfigurieren eines DP-Mastersystems

Dezentrale Peripherie

Als dezentrale Peripherie werden DP-Mastersysteme bezeichnet, die aus DP-Master und DP-Slaves bestehen, über einen Bus verbunden sind und miteinander über das Protokoll PROFIBUS-DP kommunizieren.

Firmware-Version der S7-1200-CPU

Die Nutzung der PROFIBUS-Funktionalitäten bei der S7-1200 setzt CPUs ab Firmware-Version 2.0 voraus.

Konfiguration der dezentralen Peripherie

Weil DP-Master bzw. DP-Slaves unterschiedliche Geräte sind, sind hier nur die grundsätzlichen Vorgehensweisen zum Konfigurieren erläutert. Die Konfiguration der dezentralen Peripherie ist aber weitgehend identisch mit dem Vorgehen bei der Konfiguration eines zentralen Aufbaus.

Anlegen des DP-Mastersystems in der Netzsicht

Nachdem Sie per Drag & Drop aus dem Hardware-Katalog einen DP-Master und einen DP-Slave (z. B. ein CM 1243-5 und ein CM 1243-5) in der Netzsicht platziert haben, verbinden Sie die beiden Geräte mit einem PROFIBUS-Subnetz.

Weitere Informationen

Beachten Sie weitere Informationen zum Funktionsumfang in den Handbüchern des jeweiligen Geräts.

1.4.10.2 DP-Slaves im Hardware-Katalog

DP-Slaves im Hardware-Katalog

DP-Slaves finden Sie im Hardware-Katalog in dem Ordner "Dezentrale Peripherie". Dort finden Sie kompakte und modulare DP-Slaves:

- Kompakte DP-Slaves
  Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L
- Modulare DP-Slaves
  Anschaltungsbaugruppen mit zugeordneten S7-Baugruppen, z. B. ET 200M

Die verwendbaren DP-Slaves hängen von dem vorhandenen DP-Master und der gewünschten Funktionalität ab.

I-Slaves im Hardware-Katalog

Ein DP-Slave, den Sie als intelligenten DP-Slave projektieren können, ist beispielsweise das CM 1242-5. Sie finden es im Hardware-Katalog an folgender Stelle:

- CM 1242-5
  "PLC > SIMATIC S7 1200 > Kommunikationsmodul > PROFIBUS"

1.4.10.3 DP/DP-Koppler im Hardware-Katalog

Einführung

Ein DP/DP-Koppler dient dazu, zwei PROFIBUS DP-Netze als Netzübergang (Gateway) miteinander zu verbinden, und so Daten vom DP-Master des einen Netzes zum DP-Master des anderen Netzes zu übertragen.

Die maximale Größe der übertragbaren Daten beträgt 244 Byte Eingangsdaten und 244 Byte Ausgangsdaten.

DP/DP-Koppler im Hardware-Katalog

DP/DP-Koppler als Gateway zwischen zwei DP-Mastersystemen finden Sie im Hardware-Katalog in dem Ordner "Weitere Feldgeräte > PROFIBUS DP > Netzübergänge".

Projektieren des DP/DP-Kopplers

DP/DP-Koppler werden an beiden PROFIBUS-Netzen in jeweils eigenen Mastersystemen projektiert.

Die Ein- und Ausgangsbereiche beider Netze müssen dabei aufeinander abgestimmt sein. Die Ausgangsdaten der einen Seite des DP/DP-Kopplers werden als Eingangsdaten der jeweils anderen Seite übernommen und umgekehrt.

1.4.10.4 Konfigurationen mit PROFIBUS-DP

Konfigurationen mit einfachen DP-Slaves

Datenaustausch zwischen DP-Master und DP-Slave

In der Konfiguration mit einfachen DP-Slaves erfolgt der Datenaustausch zwischen DP-Master und einfachen DP-Slaves, also mit E/A-Baugruppen über den DP-Master. Der DP-Master ruft nacheinander jeden innerhalb des DP-Mastersystems projektierten DP-Slave, der in seiner Aufrufliste (Polling-Liste) enthalten ist, ab. Er überträgt dabei die Ausgangsdaten an die Slaves und erhält deren Eingangsdaten zurückgeliefert.

Mono-Master-System

Die Konfiguration mit nur einem DP-Master wird auch als Mono-Master-System bezeichnet. Ein einziger DP-Master ist mit seinen zugehörigen DP-Slaves an einem physikalischen PROFIBUS DP-Subnetz angeschlossen.

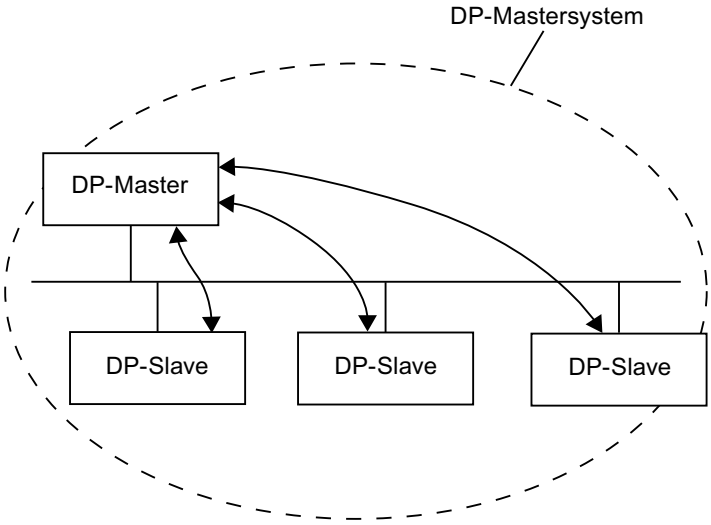

Konfigurationen mit intelligenten DP-Slaves

Definition

DP-Slaves mit eigenem Programm zur Vorverarbeitung werden als intelligente DP-Slaves (I-Slaves) bezeichnet.

Das CM 1242-5 ist ein intelligenter DP-Slave.

Datenaustausch I-Slave <> DP-Master

Ein überlagertes Automatisierungssystem behandelt die in Teilaufgaben zerlegte Automatisierungsaufgabe. Die Teilaufgaben werden in den unterlagerten Automatisierungssystemen bearbeitet. Die zu erledigenden Steuerungsaufgaben werden eigenständig und effizient als Programme der Vorverarbeitung in den CPUs abgearbeitet.

Bei Konfigurationen mit intelligenten DP-Slaves greift der DP-Master nicht auf die E/A-Baugruppen des intelligenten DP-Slaves zu, sondern nur auf den Operandenbereich der CPU des I-Slaves. Der Operandenbereich darf nicht für reale E/A-Baugruppen im I-Slave belegt werden. Diese Zuordnung muss bei der Projektierung des I-Slaves erfolgen.

Die Adressen der zwischen Master und Slave auszutauschenden Daten werden im Transferbereich des I-Slave projektiert.

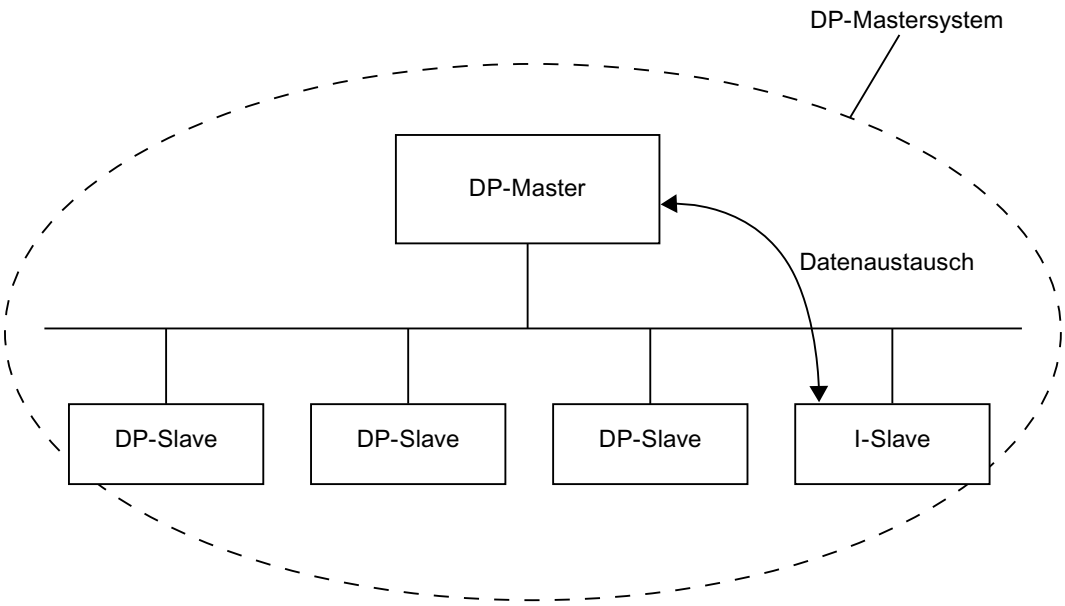

1.4.10.5 Dezentrale Peripheriesysteme konfigurieren

DP-Mastersystem anlegen

Einführung

Um ein PROFIBUS DP-Mastersystem anzulegen, benötigen Sie einen DP-Master und wenigstens einen DP-Slave. Sobald Sie einen DP-Master mit einem DP-Slave über ihre PROFIBUS DP-Schnittstellen verbinden, erfolgt eine Master-Slave-Kopplung.

DP-Master und DP-Slave

Der DP-Master muss eine PROFIBUS DP-Schnittstelle haben. Bei Verwendung einer S7-1200 CPU mit PROFINET-Schnittstelle benötigen Sie das Kommunikationsmodul CM 1243-5 mit PROFIBUS DP-Schnittstelle für den Einsatz als DP-Master.

Als DP-Slave können Sie Kopfmodule der Dezentralen Peripherie mit PROFIBUS DP-Schnittstelle oder CPUs als intelligente DP-Slaves einsetzen. Bei Verwendung einer S7-1200 CPU mit PROFINET-Schnittstelle benötigen Sie das Kommunikationsmodul CM 1242-5 mit PROFIBUS DP-Schnittstelle für den Einsatz der CPU als intelligenter DP-Slave.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

Um ein DP-Mastersystem beispielsweise mit einer CPU 1217C anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie als potenziellen DP-Master eine CPU 1217C aus dem Hardware-Katalog.
2. Ziehen Sie per Drag & Drop die CPU auf die freie Fläche in der Netzsicht.
3. Selektieren Sie das Kommunikationsmodul CM 1243-5 aus dem Hardware-Katalog.
4. Ziehen Sie das Kommunikationsmodul auf die CPU in der Netzsicht.
5. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des Kommunikationsmoduls.
6. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".

Es wird ein DP-Mastersystem mit einer CPU 1217C über CM 1243-5 als DP-Master als einzigem Teilnehmer angelegt.

Wenn Sie die DP-Schnittstelle eines DP-Slaves mit der DP-Schnittstelle des DP-Masters verbinden, wird der DP-Slave automatisch in das DP-Mastersystem aufgenommen. Wenn zwischen DP-Master und DP-Slave noch kein Subnetz existiert, wird ein neues Subnetz zwischen DP-Master und DP-Slave erzeugt.

Um beispielsweise eine CPU 1217C mit Kommunikationsmodul CM 1242-5 als intelligenten DP-Slave in das DP-Mastersystem aufzunehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die DP-Schnittstelle des DP-Masters (Schnittstelle des CM 1243-5) oder DP-Slaves (Schnittstelle des CM 1242-5).
2. Ziehen Sie mit gedrückter Maustaste eine Verbindung von der selektierten DP-Schnittstelle auf die DP-Schnittstelle des gewünschten Kommunikationspartners.

oder

1. Klicken Sie auf den Hyperlink des DP-Slaves CPU 1217C mit CM 1242-5.
2. Wählen Sie aus der angezeigten Liste möglicher DP-Master den gewünschten DP-Master aus.

Der intelligente DP-Slave CPU 1217C mit CM 1242-5 wird in das DP-Mastersystem mit CPU 1217C und CM 1243-5 als DP-Master aufgenommen.

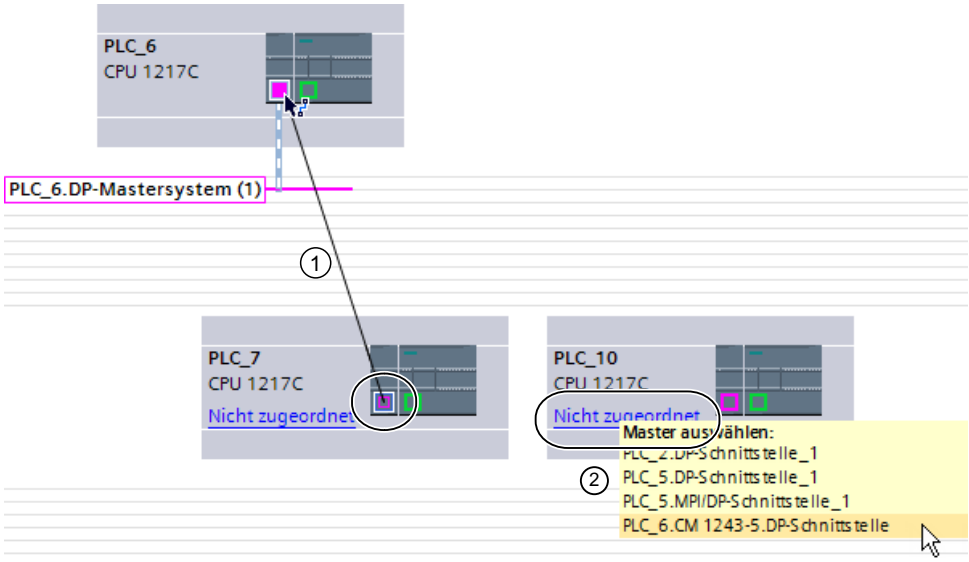

① Drag & Drop zwischen den DP-Schnittstellen von DP-Slave zu DP-Master.

② Klick auf den Link bei nicht zugeordnetem DP-Slave öffnet eine Auswahl möglicher DP-Master.

Passen Sie gegebenenfalls die Eigenschaften des PROFIBUS-Subnetzes oder des DP-Masters (z. B. PROFIBUS-Adresse) im Inspektorfenster unter "Eigenschaften" an.

Anzeige des DP-Masters im DP-Slave

Wenn ein DP-Slave mit einem DP-Master verbunden ist, wird der Name des DP-Masters als Hyperlink am DP-Slave angezeigt. Wenn Sie auf den Hyperlink klicken, wird der zugeordnete DP-Master selektiert.

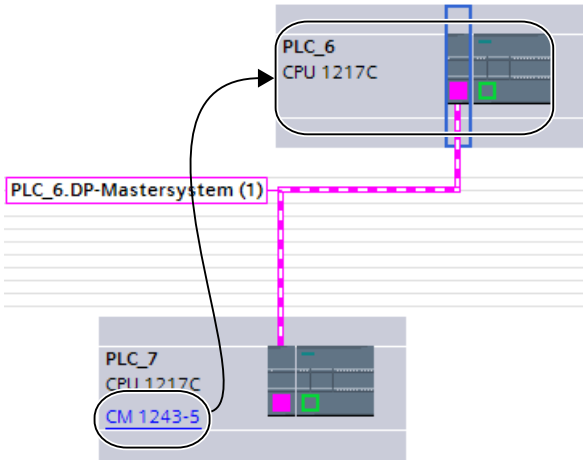

Hervorheben des DP-Mastersystems

Wenn Sie ein neues DP-Mastersystem angelegt haben, wird das DP-Mastersystem hervorgehoben dargestellt. Dadurch erkennen Sie schnell, welche Geräte zu dem DP-Mastersystem gehören. Sie können ein DP-Mastersystem auch hervorheben, indem Sie den Mauszeiger über ein Subnetz ziehen. Es werden daraufhin die Namen der vorhandenen DP-Mastersysteme angezeigt. Wenn Sie mit der Maus auf eines der angezeigten DP-Mastersysteme klicken, wird das entsprechende DP-Mastersystem hervorgehoben.

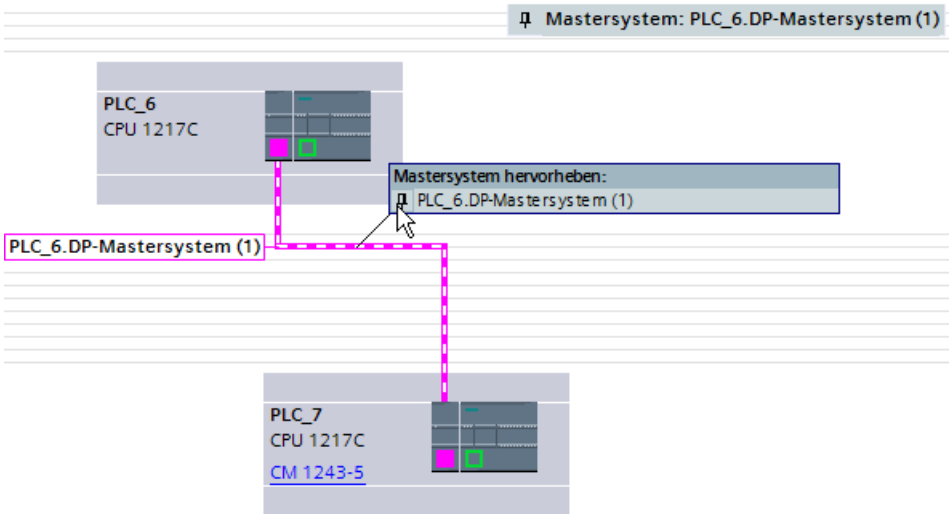

Um die Hervorhebung eines DP-Mastersystems wieder auszublenden, stehen Ihnen mehrere Möglichkeiten zu Verfügung:

- Heben Sie ein anderes DP-Mastersystem hervor.
- Klicken Sie auf die Pin-Nadel mit der Bezeichnung des DP-Mastersystems in der rechten oberen Ecke der Netzsicht.

DP-Mastersysteme und -Schnittstellen bearbeiten

Einführung

Nachdem Sie ein DP-Mastersystem angelegt haben, haben Sie auch die Möglichkeit, das DP-Mastersystem von seinen Komponenten zu trennen. Daraus können sich Subnetze mit DP-Slaves aber ohne DP-Master ergeben.

Das Bearbeiten der Schnittstellen eines DP-Masters ist meist nicht notwendig.

Am DP-Mastersystem können Sie den Namen und die Nummer ändern.

Trennung von Master oder Slaves vom DP-Mastersystemen

Wenn Sie einen PROFIBUS-CP als DP-Master mit Mastersystem projektiert haben, dann können Sie das DP-Mastersystem vom DP-Master trennen. Das Gerät ist dann nicht mehr an das DP-Mastersystem angeschlossen.

Das Trennen des Subnetzes von einem DP-Master hat zur Folge, dass das Mastersystem nicht mehr existiert, da es keinem DP-Master mehr zugeordnet ist. Die einzelnen DP-Slaves sind aber weiterhin miteinander über das Subnetz verbunden.

Wenn Sie die DP-Slaves löschen oder vom Mastersystem trennen, dann bleibt das Mastersystem am DP-Master erhalten.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Es besteht ein DP-Mastersystem mit einem DP-Master und mindestens einem DP-Slave.

DP-Master vom DP-Mastersystem trennen

Um das DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Vom Mastersystem trennen".

Der selektierte DP-Master wird vom DP-Mastersystem getrennt. Es bleibt ein Subnetz mit den DP-Slaves zurück.

DP-Master dem DP-Mastersystem hinzufügen

Um einem Subnetz wieder einen DP-Master zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle eines DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".
3. Ziehen Sie das neue DP-Mastersystem auf die DP-Schnittstellen der DP-Slaves.

Der DP-Master bildet mit den DP-Slaves wieder ein DP-Mastersystem.

Eigenschaften eines DP-Mastersystems bearbeiten

Um die Eigenschaften eines DP-Mastersystems zu bearbeiten, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger auf ein Subnetz mit DP-Mastersystem.
2. Klicken Sie in der erscheinenden Meldung über die vorhandenen DP-Mastersysteme auf das DP-Mastersystem, welches Sie bearbeiten wollen. Das DP-Mastersystem wird nun farblich hervorgehoben.
3. Klicken Sie auf das hervorgehobene DP-Mastersystem.
4. Bearbeiten Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Attribute des DP-Mastersystems.

Hinweis

Wenn Sie ein Subnetz anklicken, ohne dass ein DP-Mastersystem hervorgehoben ist, können Sie im Inspektorfenster unter "Eigenschaften" die Eigenschaften des ganzen Subnetzes bearbeiten.

DP-Slaves am Mastersystem einfügen und konfigurieren

In der Netzsicht fügen Sie verschiedene DP-Slaves direkt per Drag & Drop oder Doppelklick aus dem Hardware-Katalog ein.

Typen von DP-Slaves

Bei der Projektierung von DP-Slaves werden diese unterschieden nach:

- Kompakte DP-Slaves
  (Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L)
- Modulare DP-Slaves
  (Anschaltungsbaugruppen mit zugeordneten S5- oder S7-Baugruppen, z. B. ET 200M)
- Intelligente DP-Slaves (I-Slaves)
  (SIMATIC S7-300 mit z. B. CP 342-5, CPU 315-2 DP oder ET 200S mit IM 151-7 CPU)

Steckplatzregeln

- In einem DP-Mastersystem verwenden Sie nur einen DP-Master und einen oder mehrere DP-Slaves.
- Sie dürfen nicht mehr DP-Slaves in einem DP-Mastersystem projektieren als für den betreffenden DP-Master zulässig ist.

Hinweis

Beachten Sie bei der Konfigurierung des DP-Mastersystems die technischen Daten des DP-Masters (max. Anzahl Teilnehmer, max. Anzahl Steckplätze, max. Anzahl Nutzdaten). Möglicherweise können Sie aufgrund der Begrenzung der Nutzdaten nicht die maximale Anzahl von Teilnehmern verwenden!

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Mastersystem ist angelegt.

DP-Slave am DP-Mastersystem einfügen

Um einen DP-Slave aus dem Hardware-Katalog am DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie einen DP-Slave aus dem Hardware-Katalog.
2. Ziehen Sie den DP-Slave per Drag & Drop vom Hardware-Katalog in die Netzsicht.
3. Ziehen Sie eine Verbindung von der DP-Schnittstelle des DP-Masters oder von dem hervorgehobenen DP-Mastersystem zu der DP-Schnittstelle des neuen DP-Slave.

Es wird automatisch ein DP-Mastersystem erstellt und der DP-Slave wird automatisch mit dem DP-Master verbunden.

Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog einen Doppelklick auf den gewünschten DP-Slave durchführen. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

DP-Slave vom DP-Mastersystem trennen

Um einen DP-Slave vom DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb der Netzsicht mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slaves.
2. Wählen Sie aus dem Kontextmenü die Art der Trennung vom DP-Mastersystem:
   - "Vom Subnetz trennen": Die PROFIBUS-Verbindung wird getrennt und das Gerät ist nicht mehr an das DP-Mastersystem oder ein Subnetz angeschlossen.
   - "Vom Mastersystem trennen": Der DP-Slave ist weiterhin an das Subnetz angeschlossen, aber nicht mehr dem DP-Mastersystem als DP-Slave zugeordnet.

Der selektierte DP-Slave wird vom DP-Mastersystem getrennt.

Sie können die Trennung auch direkt über das Kontextmenü des DP-Slaves durchführen:

1. Klicken Sie innerhalb der Netzsicht mit der rechten Maustaste auf den DP-Slave.
2. Wählen Sie aus dem Kontextmenü "Vom DP-Master/IO-System trennen".

Der DP-Slave ist weiterhin an das Subnetz angeschlossen, aber nicht mehr dem DP-Mastersystem als DP-Slave zugeordnet.:

DP-Slave einem neuen DP-Mastersystem zuweisen

Um einen vorhandenen DP-Slave einem neuen DP-Mastersystem zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slave.
2. Wählen Sie im Kontextmenü "Neuem Master zuweisen".
   Es ist dabei unerheblich, ob der DP-Slave bereits einem anderen DP-Mastersystem zugeordnet ist.
3. Wählen Sie aus der Auswahlliste den DP-Master aus, an dessen DP-Mastersystem Sie den DP-Slave anbinden wollen.
   Der selektierte DP-Slave wird einem neuen DP-Mastersystem zugewiesen.

Analog dazu können Sie einen DP-Slave über die Funktion "Neuem Subnetz zuweisen" auch an ein anderes Subnetz anbinden, ohne ihn dadurch an ein bestehendes DP-Mastersystem zu koppeln.

Sie können die Zuweisung auch direkt über das Kontextmenü des DP-Slaves durchführen:

1. Klicken Sie mit der rechten Maustaste auf den DP-Slave.
2. Wählen Sie im Kontextmenü "Neuem DP-Master/IO-Controller zuweisen".
3. Wählen Sie den DP-Master aus der Liste vorhandenener DP-Master aus.

Der selektierte DP-Slave wurde einem neuen DP-Mastersystem zugewiesen.

Hinweis

Wenn sich an einem Subnetz nur ein einzelner DP-Master befindet, reicht es auch aus, wenn Sie per Drag & Drop die Schnittstelle des DP-Slaves mit dem Subnetz dieses DP-Masters verbinden. Der DP-Slave wird dann sofort mit dem Subnetz des DP-Masters verbunden und der DP-Slave wird diesem DP-Master zugewiesen.

DP-Slave konfigurieren

Um einen DP-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wechseln Sie in die Gerätesicht des DP-Slave.
2. Wählen Sie die gewünschte Baugruppen aus.
3. Konfigurieren Sie den DP-Slave im Inspektorfenster.

Tipp: Schnelles Konfigurieren von Mastersystemen

Wenn das DP-Mastersystem viele DP-Slaves hat, weisen Sie alle per Drag & Drop platzierten DP-Slaves in einem Schritt einem Master zu.

Voraussetzung

DP-Master und DP-Slaves sind in der Netzsicht platziert.

DP-Slaves einem DP-Mastersystem zuweisen

Gehen Sie folgendermaßen vor:

1. Wählen Sie einen passenden Zoomfaktor, so dass Sie möglichst viele DP-Slaves in der Netzsicht sehen.
2. Ordnen Sie die DP-Slaves in maximal zwei Reihen an.

3. Wählen Sie mit dem Mauszeiger alle DP-Schnittstellen aus (nicht die Geräte!). Das gelingt nur, wenn Sie außerhalb des ersten DP-Slaves mit dem Ziehen des Mauszeigers beginnen und beim letzten DP-Slave die Maustaste loslassen (Auswahl mit dem Lasso).

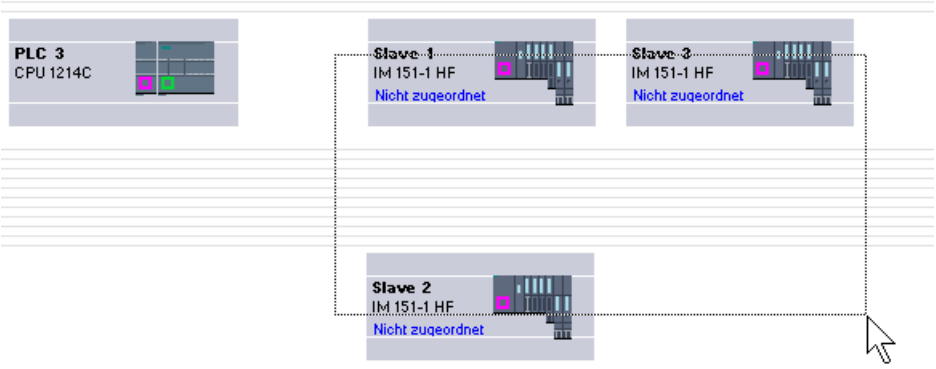

4. Wählen Sie das Kontextmenü "Neuem Master zuweisen" und markieren Sie im Folgedialog die entsprechende DP-Schnittstelle des DP-Masters.

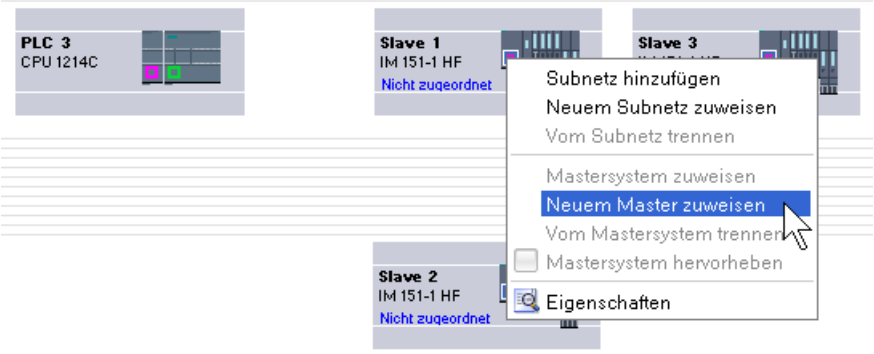

5. Die DP-Slaves werden automatisch mit dem DP-Master vernetzt und bilden mit ihm zusammen ein DP-Mastersystem.

Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog auf einen DP-Slave doppelklicken und damit schnell weitere DP-Slaves hinzugügen. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

1.4.10.6 Intelligente DP-Slaves konfigurieren

I-Slave in ein DP-Mastersystem einfügen

Einführung

Merkmal eines intelligenten DP-Slave (I-Slave) ist, dass Ein-/Ausgangsdaten nicht unmittelbar von einem realen Ein-/Ausgang dem DP-Master zur Verfügung gestellt werden, sondern von einer vorverarbeitenden CPU. Diese vorverarbeitende CPU bildet zusammen mit dem CP den I-Slave.

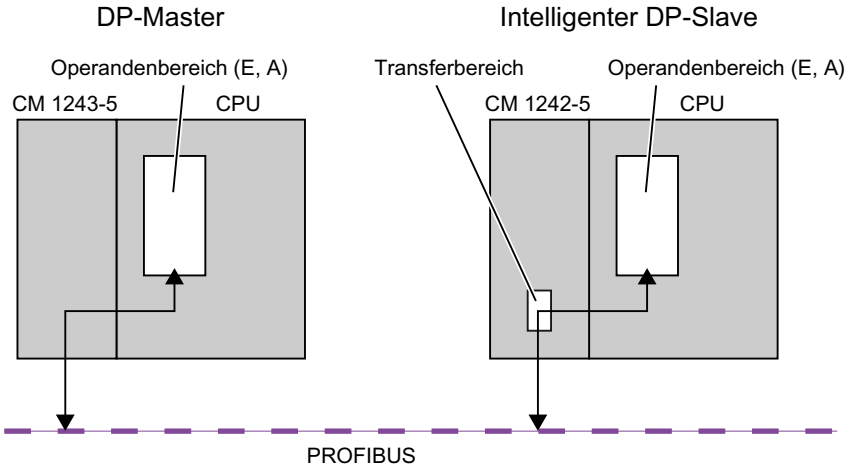

Unterschied: DP-Slave - Intelligenter DP-Slave

Bei einem DP-Slave greift der DP-Master auf die dezentralen Ein-/Ausgänge zu.

Bei einem intelligenten DP-Slave greift der DP-Master nicht auf die angeschlossenen Ein-/Ausgänge des intelligenten DP-Slaves zu, sondern auf einen Übergabebereich im E/A-Adressraum der vorverarbeitenden CPU. Das Anwenderprogramm der vorverarbeitenden CPU muss für den Austausch der Daten zwischen Operandenbereich und Ein-/Ausgängen sorgen.

Hinweis

Die konfigurierten E/A-Bereiche für den Datenaustausch zwischen DP-Master und DP-Slaves dürfen nicht von E/A-Baugruppen verwendet werden.

Anwendungen

Konfigurationen mit intelligenten DP-Slaves: Datenaustausch I-Slave <> DP-Master

Vorgehen

Um einen I-Slave in ein DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Netzsicht aus dem Hardware-Katalog einen CM 1242-5 als I-Slave in eine Station und einen CM 1243-5 als DP-Master in eine Station.
2. Ziehen Sie eine Verbindungslinie zwischen den DP-Schnittstellen der beiden Geräte. Dadurch verbinden Sie den I-Slave mit einem DP-Master in einem DP-Mastersystem. Ergebnis: Sie haben nun ein DP-Mastersystem mit einem DP-Master und einem I-Slave aufgebaut.

Zugriff auf Daten des I-Slave konfigurieren

Datenzugriff

Für den CM 1242-5 als I-Slave gilt: Die Adressen für den Übergabebereich der Daten und die Adresse für die E/A-Baugruppen im I-Slave sind unterschiedlich. Die Anfangsadresse, die eine E/A-Baugruppe belegt, kann daher nicht mehr für den Übergabespeicher verwendet werden. Wenn also der übergeordnete DP-Master auf Daten einer E/A-Baugruppe im I-Slave zugreifen soll, müssen Sie diesen Datenaustausch zwischen E/A-Baugruppe und Übergabebereich im Anwenderprogramm des I-Slave konfigurieren.

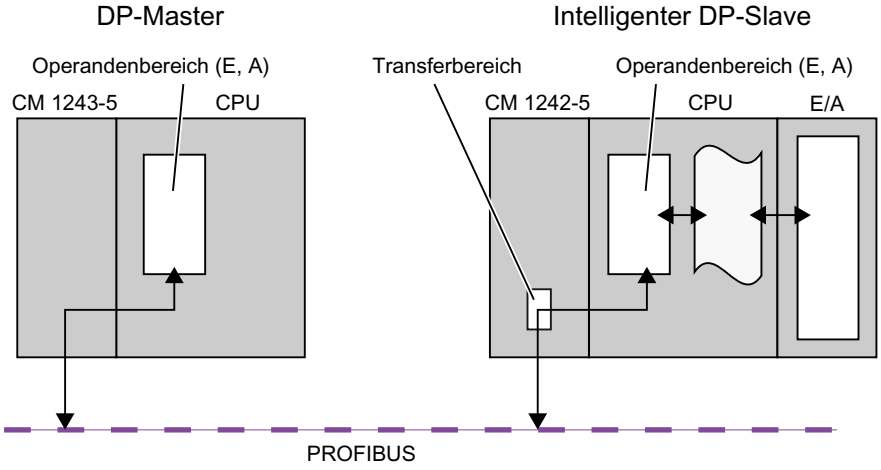

Projektierung des Übergabebereichs beim CM 1242-5 (Transferbereich)

Beim CM 1242-5 wird der Übergabebereich für den zyklischen PROFIBUS-Datenaustausch in der Parametergruppe "PROFIBUS-Schnittstelle > Betriebsart > I-Slave-Kommunikation" als Transferbereich projektiert.

Direkter Datenzugriff von CPU zu CPU

Direkter Datenzugriff von CPU zu CPU über PROFIBUS wird von den S7-1200-PROFIBUS-CMs nur über die PUT/GET-Dienste unterstützt.

1.4.10.7 DP-Slaves der Dezentralen Peripherie konfigurieren

ET 200S konfigurieren

Steckplatzregeln für die Konfiguration einer ET 200S

Für das Konfigurieren der ET 200S gelten die folgende Regeln:

- Stecken Sie die Module der ET 200S lückenlos nebeneinander.
- Steckplatz 1: nur für Powermodule PM-E oder PM-D
- Links neben einem Elektronikmodul (EM): Nur ein EM oder ein Powermodul (PM-E oder PM-D)
- Links neben einem Motorstarter (MS): Nur ein MS oder ein Powermodul PM-D, PM-D Fx (1..x..4) oder PM-X
- Links neben einem PM-X: Nur ein Motorstarter oder ein PM-D
- Erlaubt sind maximal 63 Module und ein Interfacemodul IM

Hinweis

Beachten Sie die korrekte Zuordnung von PM-E-Spannungsbereich und EM-Spannungsbereich!

Vergleichsstelle parametrieren

Eine Vergleichsstelle ist der Anschluss eines Thermoelements an eine Zuleitung (i. d. R. im Klemmenkasten). Die hier durch Temperatureinfluss auftretende Spannung verfälscht den Temperaturwert, den das Modul misst.

Bei ET 200S kann ein Kanal des Moduls AI RTD als Vergleichsstelle parametriert werden. Mit der von diesem Modul gemessenen Temperatur an der Vergleichsstelle können andere AI TC-Module ihre Messwerte kompensieren.

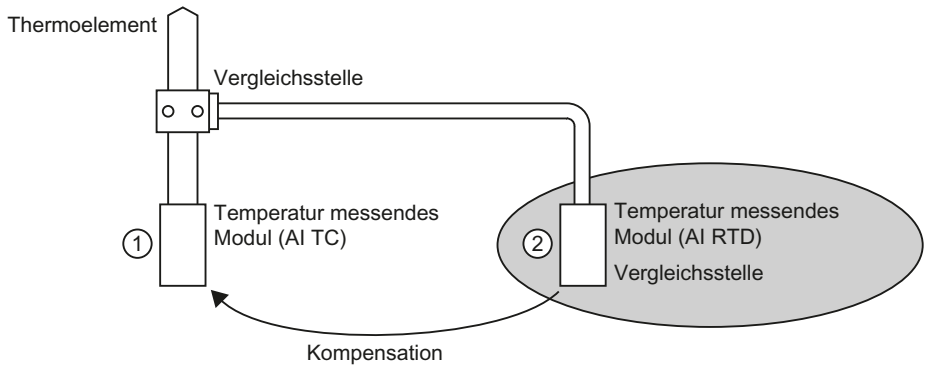

① Parametrierung des AI TC:
→ Auswahl der verwendeten Vergleichsstelle

② Parametrierung des AI RTD:
→ Aktivierung der Vergleichstelle
→ Festlegung Steckplatz und Kanal des AI RTD

Besonderheiten bei der Parametrierung von Vergleichsstellen

Die Parametrierung von Vergleichsstellen wird Ihnen exemplarisch an der Verwendung eines Widerstandsthermometers Pt 100 "Klimabereich" zum Erfassen der Vergleichsstellentemperatur gezeigt.

Um die Vergleichsstelle zu parametrieren, gehen Sie folgendermaßen vor:

1. Platzieren Sie in der Gerätesicht der ET 200S ein analoges Elektronikmodul, z. B. 2AI RTD HF.
2. Selektieren Sie die Baugruppe im Baugruppenträger.
3. Stellen Sie im Inspektorfenster unter "Eigenschaften > Eingänge" einen Kanal für die Vergleichsstellen-Funktion auf den Messbereich "RTD-4L Pt 100 Kl." ein.
4. Selektieren Sie die ET 200S.
5. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter > Vergleichsstellen" das Optionskästchen "Vergleichsstelle" und geben Sie den Steckplatz und die Kanalnummer der relevanten RTD-Baugruppe an.
6. Platzieren Sie das analoge Elektronikmodul für Temperaturmessung mittels Thermoelement (TC-Modul) und parametrieren Sie es mit der Vergleichsstellennummer des RTD-Moduls.

Weitere Informationen

Beachten Sie für weitere Informationen zu den verschiedenen Typen und Einsatzmöglichkeiten von Modulen in ET 200S die Betriebsanleitung und das Gerätehandbuch "Dezentrales Peripheriesystem ET 200S".

Beachten Sie für weitere Informationen zur Analogwertverarbeitung die Dokumentation zum Dezentralen Peripheriesystem ET 200S.

Adressen packen

Einführung

DP-Slaves und IO-Devices der Familie ET 200S werden konfiguriert wie andere modulare DP-Slaves und IO-Devices. Zusätzlich zu den üblichen Funktionen modularer DP-Slaves und IO-Devices gibt es bei ET 200S die Funktion "Adressen packen":

Digitale Elektronikmodule mit einem benötigten Adressraum von 2 oder 4 Bit belegen beim Einfügen in der Gerätesicht zunächst pauschal 1 Byte. Der wirklich belegte Adressraum kann aber nach dem Konfigurieren durch die Funktion "Adressen packen" zusammengeschoben werden.

	Vorbelegung	Nach "Adressen packen"
Modul	E-Adresse	E-Adresse
2DI (2 Bit)	Byte 10	10.0...10.1
4DI (4 Bit)	Byte 11	10.2...10.5

Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Eine ET 200S, z. B. IM 151-1, ist vorhanden.
- Ein paar digitale Elektronikmodule, z. B. 2DI AC120V ST befinden sich in den Steckplätzen.

Adressen packen

Um Adressen zu packen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Elektronikmodule, deren Adressen gepackt werden sollen. Für das Selektieren von mehreren Elektronikmodulen haben Sie folgende Möglichkeiten:
   - Klicken Sie die Elektronikmodule bei gleichzeitigem Drücken von <Shift> oder <Strg> einzeln an.
   - Klicken Sie außerhalb des Baugruppenträgers und ziehen Sie die Maus um die zu selektierenden Elektronikmodule.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen packen".

Es werden die Adressbereiche für Eingänge, Ausgänge und Motorstarter getrennt gepackt. In den Spalten E-Adresse und A-Adresse der Geräteübersicht sehen Sie die gepackten Adressen.

Adressbildung und Struktur bei gepackten Adressen

Wenn Sie die Funktion "Adressen packen" verwenden, werden die Adressen der selektierten Elektronikmodule nach folgenden Regeln gepackt:

- Der Anfang des Adressbereichs ist durch die niedrigste Adresse der selektierten Elektronikmodule festgelegt: X.0.
- Wenn die Bitadresse nicht "0" ist, dann wird automatisch die nächste (freie) Byteadresse herangezogen, ab welcher der markierte Bereich eingeschoben werden kann: (X+n).0.
- Wenn kein zusammenhängender Bereich mehr vorhanden ist, wird automatisch in vorhandene Adresslücken gepackt.

Elektronikmodule mit gepackten Adressen und derselben Byteadresse bilden eine Packgruppe.

Adressen entpacken

Um Adressen zu entpacken, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Elektronikmodule mit gepackter Adresse.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen entpacken".

Packgruppen der selektierten Elektronikmodule werden aufgelöst und die gepackten Adressen der entsprechenden Elektronikmodule werden entpackt.

Wenn Sie aus einer Packgruppe Elektronikmodule löschen oder verschieben oder auf einen freien Steckplatz innerhalb einer Packgruppe Elektronikmodule einfügen, wird die Packgruppe ebenfalls aufgelöst und die gepackten Adressen werden entpackt.

Die Anfangsadressen der entpackten Elektronikmodule werden auf die jeweils nächsten freien Byte-Adressen gelegt.

Besonderheiten bei Elektronikmodulen mit gepackten Adressen

Für ein Elektronikmodul mit einer gepackten Adresse gelten folgende Besonderheiten:

- Aus CPU-Sicht ist keine Steckplatzzuordnung für das Elektronikmodul möglich. Aus diesem Grund gibt die Anweisung GADR_LGC (SFC 5) für den tatsächlichen Steckplatz des Elektronikmoduls die Fehlerinformation W#16#8099 "Steckplatz nicht projektiert" aus.
- Die Anweisung LGC_GADR (SFC 49) und die SZL-ID W#16#xy91 "Baugruppenzustandsinformation" für ein Elektronikmodul können nicht ausgewertet werden.
- Das Elektronikmodul erhält eine zusätzliche Diagnoseadresse über die DPV1-Funktionalität, weil aus CPU-Sicht Alarme aufgrund der gepackten Adresse sonst nicht zugeordnet werden können.
- Der "Ziehen/Stecken-Alarm" ist nicht möglich, da sich die Funktionen "Adressen packen" und "Ziehen/Stecken-Alarm" gegenseitig ausschließen.

Optionenhandling mit Reservemodulen konfigurieren

Über das Optionenhandling können Sie die ET 200S mit PROFIBUS-Schnittstelle für zukünftige Erweiterungen (Optionen) einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling mit Reservemodulen beschrieben.

Dazu montieren, verdrahten, projektieren und programmieren Sie den geplanten Maximalausbau der ET 200S und verwenden statt der erst später erforderlichen Elektronikmodule bei der Montage zunächst preisgünstige Reservemodule (138-4AA00 oder 138-4AA10).

Hinweis

Die ET 200S kann mit der Stammverkabelung vollständig vorverdrahtet werden, da ein Reservemodul keine Verbindung zu den Klemmen des Terminalmoduls und damit zum Prozess hat.

Voraussetzung

- ET 200S Interfacemodul
  - IM 151-1 STANDARD (ab 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (ab 6ES7 151-1AB02-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Aktivieren Sie nun die nummerierten Optionskästchen für die Steckplätze, die zunächst mit Reservemodulen statt mit Elektronikmodulen bestückt sind.
3. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).

Zu einem späteren Zeitpunkt können die montierten Reservemodule durch die projektierten Module ersetzt werden, ohne die Projektierung nachholen zu müssen.

Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1 STANDARD) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheriesystem ET 200S.

Funktionsweise des Optionenhandlings im Anlauf

Wenn "Anlauf bei Soll- ungleich Istausbau" gesperrt ist, dann läuft die ET 200S auch dann an, wenn statt des projektierten Elektronikmoduls ein Reservemodul steckt und das Optionenhandling für diesen Steckplatz eingeschaltet ist.

Funktionsweise des Optionenhandlings im Betrieb

Im Betrieb unterscheidet sich die Funktionsweise des Optionenhandlings wie folgt:

- Optionenhandling für einen Steckplatz eingeschaltet:
  Auf diesem Steckplatz darf sich das Reservemodul (Option) oder das projektierte Elektronikmodul befinden. Wenn sich ein anderes Modul auf diesem Steckplatz befindet, wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).
- Optionenhandling für einen Steckplatz ausgeschaltet:
  Auf diesem Steckplatz darf sich nur das projektierte Elektronikmodul befinden. Bei jedem anderen Modul wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).

Ersatzwerte des Reservemoduls

- Ersatzwert für digitale Eingänge: 0
- Ersatzwert für analoge Eingänge: 0x7FFF

Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 2 bis 63:

- Wert des Bits = 0: Es gilt die Parametrierung des Optionenhandlings. Reservemodule sind erlaubt.
- Wert des Bits = 1: Die Parametrierung des Optionenhandlings ist aufgehoben. Reservemodule werden auf diesem Steckplatz nicht akzeptiert:

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Auf dem Steckplatz befinden sich das Reservemodul, ein falsches Modul oder ein gezogenes Modul.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

Siehe auch

Welche Module unterstützen Optionenhandling? (http://support.automation.siemens.com/WW/view/de/22564754)

Optionenhandling ohne Reservemodule konfigurieren

Über das Optionenhandling können Sie die ET 200S für zukünftige Erweiterungen (Optionen) auch ohne die Montage von Reservemodulen einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling ohne Reservemodulen beschrieben.

Hinweis

ET 200S mit PROFINET-Schnittstelle

Diese Beschreibung bezieht sich auf ET 200S mit PROFIBUS-Schnittstelle. Das Optionenhandling für ET 200S mit PROFINET-Schnittstelle funktioniert prinzipiell so wie hier beschrieben ohne Reservemodule. Statt der hier aufgeführten DP-Interfacemodule sind PN-Interfacemodule zu verwenden. Weitere Informationen zum Optionenhandling bei ET 200S mit PROFINET-Schnittstelle finden Sie in den entsprechenden Gerätehandbüchern.

Voraussetzung

- ET 200S Interfacemodul
  - IM 151-1 HIGH FEATURE (ab 6ES7151-1BA02)
  - IM 151-1 STANDARD (ab 6ES7 151-1AA05-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).
3. Konfigurieren Sie den Maximalausbau des Slaves. Die Aktivierung/Deaktivierung von Optionen wird über das Anwenderprogramm gesteuert.

Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheriesystem ET 200S.

Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 1 bis 63:

- Wert des Bits = 0: Steckplatz ist in der Istkonfiguration nicht vorhanden.
- Wert des Bits = 1: Steckplatz ist in der Istkonfiguration vorhanden.

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Steckplatz gehört zu einer nicht vorhandenen Option oder Modulstatus ist nicht in Ordnung.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

Siehe auch

Beispielapplikation für ET 200S, Optionenhandling ohne Reservemodule (http://support.automation.siemens.com/WW/view/de/29430270)

ET 200S im DPV1-Modus konfigurieren

Mit PROFIBUS DPV1 stehen Ihnen erweiterte PROFIBUS-Funktionalitäten zur Verfügung.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Master mit DPV1-Funktionalität ist vorhanden.
- Eine Master-Slave-Verbindung mit PROFIBUS ist aufgebaut.

Vorgehen

Um den DP-Slave auf DPV1 zu schalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den DP-Slave.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" die Betriebsart "DPV1".

oder

1. Selektieren Sie den DP-Master.
2. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Verbindung zwischen dem DP-Master und dem gewünschten DP-Slave.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" die Betriebsart "DPV1".

Besonderheiten

Es gibt Abhängigkeiten der Parameter untereinander, die im Folgenden dargestellt sind:

Parameter	Betriebsart DPV0	Betriebsart DPV1
Betrieb bei Soll- ungleich Istausbau	Uneingeschränkt bedienbar	Uneingeschränkt bedienbar
Diagnosealarm (OB 82)	Nicht bedienbar, nicht gesetzt	Uneingeschränkt bedienbar
Prozessalarm (OB 40 bis 47)	Nicht bedienbar, nicht gesetzt	Uneingeschränkt bedienbar
Ziehen/Stecken-Alarm (OB 83)	Nicht bedienbar, nicht gesetzt	Nur bedienbar, wenn Adressen nicht gepackt sind. Wenn Ziehen/Stecken-Alarm aktiviert wird, dann wird automatisch "Anlauf bei Soll- ungleich Istausbau" aktiviert.

Alarme bei Modulen mit gepackten Adressen

Wenn das Modul Alarme auslösen kann und die Bitadresse wegen gepackter Adressen ungleich 0 ist, dann müssen Sie im Adressdialog der ET 200S eine Diagnoseadresse vergeben.

Die Diagnoseadresse ist notwendig für die Zuordnung eines DPV1-Alarms zum Modul als Alarmauslöser. Nur wenn ein Modul diese "ungepackte" Adresse hat, kann die CPU einen Alarm zuordnen und Informationen zum Alarm in der Startinformation des Alarm-OB bzw. im Diagnosepuffer hinterlegen. Eine "gepackte" Adresse kann die CPU hierfür nicht verwenden.

Aus Sicht der Alarmverarbeitung (Alarm-OB) hat das Modul dann die zugeordnete Diagnoseadresse, für die Verarbeitung der Ein- und Ausgangsdaten im Anwenderprogramm hat das Modul die gepackten Adressen!

Hinweis

Wenn die Adressen des Moduls gepackt sind, ist der Ziehen/Stecken-Alarm für die ET 200S gesperrt!

1.4.10.8 GSD-Dateien verwenden

GSD-Revisionen

Wissenswertes zu GSD-Revisionen

Die Eigenschaften von DP-Slaves werden durch GSD-Dateien für Projektierungstools verfügbar gemacht. Funktionserweiterungen im Bereich der dezentralen Peripherie haben Auswirkungen auf die GSD-Spezifikation, z. B. führen sie zur Definition neuer Schlüsselwörter. Daraus ergibt sich eine Versionierung der Spezifikation. Die Version der Spezifikation, nach der sich eine GSD-Datei richtet, heißt bei GSD-Dateien "GSD-Revision". Die GSD-Revision ist als Schlüsselwort "GSD_Revision" in GSD-Dateien ab GSD-Revision 1 obligatorisch enthalten. GSD-Dateien ohne dieses Schlüsselwort werden daher von Projektierungstools als GSD-Revision "0" interpretiert.

GSD-Dateien können bis GSD-Revision 5 interpretiert werden. Damit werden DP-Slaves unterstützt, die z. B. folgende Funktionen unterstützen:

- Diagnosemeldungen für Alarmblöcke
- Taktsynchronität und Äquidistanz (Isochron-Modus)
- SYNC/FREEZE
- Uhrzeitsynchronisation für DP-Slaves

GSD-Datei installieren

Einführung

In einer GSD-Datei (Gerätestammdaten-Datei) sind alle Eigenschaften eines DP-Slaves hinterlegt. Falls Sie einen DP-Slave projektieren möchten, der nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller mitgelieferte GSD-Datei installieren. Über GSD-Dateien installierte DP-Slaves werden im Hardware-Katalog angezeigt und können dann selektiert und konfiguriert werden.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Sie haben Zugriff auf die benötigten GSD-Dateien in einem Verzeichnis auf der Festplatte.

Vorgehen

Um eine GSD-Datei zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Dateien liegen.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien eine oder mehrere Dateien aus.
4. Klicken Sie auf die Schaltfläche "Installieren". Die ausgewählten GSD-Dateien werden nun installiert.
5. Um eine Protokolldatei für die Installation zu erstellen, klicken Sie auf die Schaltfläche "Protokolldatei speichern".
   Eventuell auftretende Probleme bei der Installation lassen sich anhand der Protokolldatei nachvollziehen.
6. Klicken Sie auf "Schließen". Sie werden darüber informiert, dass die DP-Slaves aus den installierten GSD-Dateien in den Hardware-Katalog eingetragen werden. Dieser Vorgang kann ein paar Sekunden dauern.

Sie haben nun mit der Installation einer GSD-Datei den darin beschriebenen DP-Slave dem Hardware-Katalog hinzugefügt.

Ergebnis

Sie finden die neuen über GSD-Dateien installierten DP-Slaves unter "Weitere Feldgeräte > PROFIBUS DP" im Hardware-Katalog.

GSD-Dateien werden immer mit dem Projekt zusammen gespeichert, d. h. alle zur Darstellung des Gerätes relevanten Informationen (auch die Symbole) sind auch im gespeicherten Projekt verfügbar.

Siehe auch

Übersicht über den Hardware- und Netzwerkeditor (Seite 9)

GSD-Datei löschen

Einführung

Sie können über GSD-Dateien installierte DP-Slaves löschen. Diese werden danach im Hardware-Katalog nicht mehr angezeigt.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Es befinden sich über GSD-Dateien installierte DP-Slaves im Hardware-Katalog.

Vorgehen

Um eine GSD-Datei zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Datei liegt.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien die zu löschende Datei aus.
4. Klicken Sie auf die Schaltfläche "Löschen".

Die ausgewählte GSD-Datei wurde gelöscht und der DP-Slave befindet sich nicht mehr im Hardware-Katalog.

Ungenutzte GSD-Dateien finden

Einführung

Wenn Sie GSD-basierte Hardware-Komponente ihrer Hardware-Konfiguration hinzufügen, dann werden unterschiedliche GSD-Dateien zu den Projektdaten hinzugefügt. Wird die GSD-basierte Hardware-Komponente aus der Konfiguration wieder gelöscht, verbleiben die GSD-Dateien im Datenbestand des Projektes. Diese GSD-Dateien werden im Projekt nicht mehr benötigt, sofern die entsprechende GSD-basierte Hardware-Komponente nicht erneut in der Hardware-Konfiguration verwendet wird. Im Projekt nicht mehr benötigte Dateien können Sie über einen Suchmechanismus finden und dann löschen.

Voraussetzung

- Es muss ein Projekt geöffnet sein.
- Das Projekt darf nicht online sein.
- Das Projekt darf nicht schreibgeschützt sein.

Vorgehen

Um ungenutzte GSD-Dateien zu finden und zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Klicken Sie im Register "GSDs im Projekt" auf die Schaltfläche "Ungenutzte GSDs finden".

   Hinweis

   Ungenutzte GSD-Dateien sind nur verfügbar, wenn mindestens eine GSD-basierte Hardware-Komponente der Konfiguration hinzugefügt und anschließend wieder aus der Konfiguration entfernt wurde. Ungenutzte GSD-Dateien in den Projektbibliotheken werden bei der Suche ebenfalls berücksichtigt. GSD-basierte Hardware-Komponenten, die noch in der Konfiguration verwendet werden, gelten nicht als ungenutzt.

   Installierte GSD-basierte Hardware-Komponenten, die im Hardware-Katalog enthalten sind, aber bisher nicht in der Hardware-Konfiguration des Projekts verwendet wurden, können keine ungenutzten GSD-Daten im Projekt abgelegt haben.

3. Wenn nicht mehr benötigte GSD-Dateien in den Projektdaten gefunden wurden, können Sie diese nun löschen. Klicken Sie dafür auf die Schaltfläche "Löschen".

Die nicht mehr benötigten GSD-Dateien werden aus den Projektdaten gelöscht. Die GSD-basierte Hardware-Komponente ist weiterhin im Hardware-Katalog verfügbar und kann bei Bedarf der Konfiguration wieder hinzugefügt werden. In diesem Fall werden die dann notwendigen GSD-Daten erneut in die Projektdaten kopiert.

Hinweis

Wenn Sie eine GSD-basierte Hardware-Komponente über "Extras > Gerätebeschreibungsdateien (GSD) verwalten > Installierte GSDs" löschen, ist diese aus dem Hardware-Katalog gelöscht. Ungenutzte GSD-Dateien dieser Hardware-Komponente können aber immer noch ein Teil der Projektdaten sein und können über die Funktion zum Finden der ungenutzten GSD-Dateien gelöscht werden.

ACHTUNG

Nicht mehr rückgängig zu machende Aktionen

Wenn Sie als Aktion die ungenutzten GSD-Dateien löschen, führt dies dazu, dass der Aktionsstapel für rückgängig zu machende Aktionen gelöscht wird. Die Aktion des Löschens von ungenutzten GSD-Dateien, sowie alle vorherigen Aktionen können dann nicht mehr rückgängig gemacht werden.

GSD-basierten DP-Slave konfigurieren

DP-Slaves, die Sie über die Installation einer GSD-Datei eingefügt haben, können Sie wie üblich über den Hardware-Katalog auswählen und in der Netzsicht einfügen. Wenn Sie die Baugruppen der GSD-basierten DP-Slaves einfügen wollen, müssen Sie einige Besonderheiten beachten.

Voraussetzung

- Sie haben einen DP-Slave über eine GSD-Datei installiert.
- Sie haben die Kopfbaugruppe in der Netzsicht wie üblich eingefügt.
- Die Geräteübersicht in der Gerätesicht ist geöffnet.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

Um die Baugruppen eines GSD-basierten DP-Slaves hinzuzufügen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu den Baugruppen des GSD-basierten DP-Slaves. GSD-basierte DP-Slaves, auch DP-Normslaves genannt, befinden sich im Hardware-Katalog im Ordner "Weitere Feldgeräte".
2. Selektieren Sie die gewünschte Baugruppe.
3. Ziehen Sie die Baugruppe mit Drag & Drop auf einen freien Platz der Geräteübersicht.
4. Selektieren Sie die Baugruppe in der Geräteübersicht, um deren Parameter zu bearbeiten.

Sie haben die Baugruppe nun auf einen freien Steckplatz des GSD-basierten DP-Slaves gesteckt und können dessen Parameter bearbeiten.

Hinweis

In dem grafischen Bereich der Gerätesicht können Sie nur den GSD-basierten DP-Slave sehen. Die hinzugefügten Baugruppen von GSD-basierten DP-Slaves finden Sie immer nur in der Geräteübersicht.

Sollkonfiguration

Bei Baugruppen mit einer einstellbaren Sollkonfiguration können Sie diese im Inspektorfenster unter "Eigenschaften > Sollkonfiguration" ändern.

1.4.11 PROFINET IO konfigurieren

1.4.11.1 Wissenswertes zu PROFINET IO

Was ist PROFINET IO?

PROFINET IO

PROFINET als ethernet-basierter Automatisierungsstandard der PROFIBUS Nutzerorganisation e.V. (PNO) definiert ein herstellerübergreifendes Kommunikations-, Automatisierungs- und Engineering-Modell.

Zielsetzung

Zielsetzung von PROFINET ist:

- eine durchgängige Kommunikation über Feldbus und Ethernet
- eine offene, verteilte Automatisierung
- die Verwendung offener Standards

Architektur

Die PROFIBUS Nutzerorganisation e.V. (PNO) hat für die PROFINET-Architektur folgende Teilaspekte vorgesehen:

- Kommunikation zwischen Steuerungen als Komponenten in verteilten Systemen.
- Kommunikation zwischen Feldgeräten wie z. B. Peripheriegeräten und Antrieben.

Umsetzung durch Siemens

Die Forderung nach "Kommunikation zwischen Steuerungen als Komponenten in verteilten Systemen" ist durch "Component Based Automation" (CBA) umgesetzt. Mit Component Based Automation erstellen Sie eine verteilte Automatisierungslösung auf Basis vorgefertigter Komponenten und Teillösungen.

Die Forderung nach "Kommunikation zwischen Feldgeräten" ist von Siemens mit "PROFINET IO" umgesetzt. Wie bei PROFIBUS DP ist die vollständige Projektierung und Programmierung der beteiligten Komponenten mit dem Totally Integrated Automation Portal möglich.

Um die Projektierung der Kommunikation zwischen Feldgeräten mit PROFINET IO geht es in den folgenden Abschnitten.

Übersicht über die RT-Klassen

RT-Klassen in PROFINET IO

PROFINET IO ist ein skalierbares Echtzeit-Kommunikationssystem auf Basis der Ethernet-Technologie. Der skalierbare Ansatz drückt sich in verschiedenen Echtzeitklassen aus:

- **RT:** Übertragung der Daten in priorisierten Ethernet-Telegrammen, nicht taktsynchron. Die benötigte Bandbreite fällt in den Bereich der freien Bandbreite für die TCP/IP-Kommunikation.
- **IRT:** Taktsynchrone Übertragung der Daten mit hoher Stabilität für zeitkritische Anwendungen (z. B. Motion-Control). Die benötigte Bandbreite fällt in den Bereich der für zyklische Daten reservierte Bandbreite.

Je nach Gerät werden nicht alle Echtzeitklassen unterstützt.

Welche IO-Controller und IO-Devices unterstützen welche PROFINET-Funktionen?

Weitere Informationen und Übersichten

Im folgenden Beitrag (http://support.automation.siemens.com/WW/view/de/44383954) des Siemens Industry Online Supports finden Sie eine Übersicht der PROFINET IO-Controller und IO-Devices, die die folgenden PROFINET Funktionen unterstützen:

- Isochronous Real-Time-Kommunikation (IRT)
- Priorisierter Hochlauf
- Medienredundanz (MRP)
- PROFIenergy
- Shared-Device
- I-Device
- Taktsynchroner Betrieb von Prozessdaten

Die Funktionen sind in den folgenden Abschnitten erläutert, allerdings ohne die jeweilige Hardware zu nennen, die diese Funktionen unterstützen.

Im Hardware-Katalog finden Sie in der Beschreibung unterhalb der markierten Komponente ebenfalls eine Übersicht der unterstützten Funktionen.

Eine Beschreibung von PROFINET in der jeweils aktuellen STEP 7-Version finden Sie auch hier (http://support.automation.siemens.com/WW/view/de/49948856).

Anbindung bestehender Bussysteme

Kopplung von PROFINET und PROFIBUS

PROFINET IO und PROFIBUS DP können über folgende Wege miteinander gekoppelt werden:

- über Industrial Ethernet:
  Zur Verbindung der beiden Netztypen Industrial Ethernet (Leitebene) und PROFIBUS (Zellebene/Feldebene) verwenden Sie z. B. das IE/PB-Link.
- über Industrial Wireless LAN:
  PROFIBUS-Geräte können Sie z. B. drahtlos an PROFINET IO über ein Wireless-LAN/PB-Link koppeln. Dadurch können Sie bestehende PROFIBUS-Konfigurationen in PROFINET integrieren.

AS-Interface-Geräte können Sie mit einem IE/AS-i Link PN IO an die Schnittstelle eines PROFINET-Geräts koppeln. Dadurch können Sie das bereits bestehende AS-i-Netzwerk in PROFINET integrieren.

Das folgende Bild zeigt die Anbindung eines PROFIBUS-Subnetzes über ein PROFINET-Gerät mit Proxy-Funktionalität.

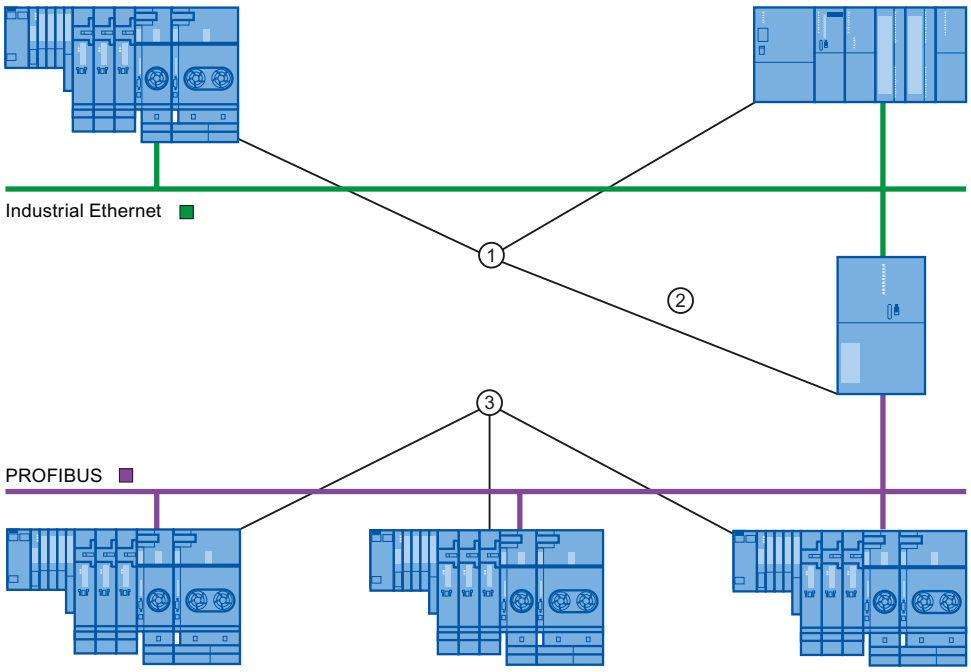

① PROFINET-Geräte
② PROFINET-Gerät mit Proxy-Funktionalität (z. B. IE/PB-Link)
③ PROFIBUS-Geräte

PROFINET-Gerät mit Proxy-Funktionalität als Stellvertreter eines PROFIBUS-Geräts

Das PROFINET-Gerät mit Proxy-Funktionalität ist der Stellvertreter eines PROFIBUS-Geräts am Ethernet. Die Proxy-Funktionalität ermöglicht es, dass ein PROFIBUS-Gerät nicht nur mit seinem Master, sondern mit allen Teilnehmern am PROFINET kommunizieren kann.

Bestehende PROFIBUS-Systeme können bei PROFINET problemlos mit Hilfe der Proxy-Funktionalität in die PROFINET-Kommunikation eingebunden werden.

Wenn Sie beispielsweise ein PROFIBUS-Gerät über einen IE/PB-Link an PROFINET anbinden, nimmt der IE/PB-Link stellvertretend für die PROFIBUS-Komponenten die Kommunikation über PROFINET auf.

Konfiguration mit IE/PB-Link PN IO

Konfiguration mit IE/PB-Link PN IO

Um PROFIBUS DP-Konfigurationen an PROFINET IO anzubinden, können Sie das IE/PB-Link PN IO verwenden.

Aus Sicht der CPU hängen die PROFIBUS-DP-Slaves am gleichen Netz wie das IE/PB-Link PN IO. Diese Slaves haben denselben Gerätenamen und IP-Adresse wie das IE/PB-Link PN IO, aber unterschiedliche Gerätenummern. Außerdem weisen sie noch eine spezifische PROFIBUS-Adresse auf.

In den Eigenschaften des IE/PB-Links werden neben den PROFINET-Gerätenummern auch die PROFIBUS-Adressen der angeschlossenen DP-Slaves angezeigt, da dieses Gerät zwei Adressierungsschemas besitzt.

Hantieren von Gerätenummern und PROFIBUS-Adressen am Mastersystem

Beim Plazieren wird dieselbe Ziffer für die PROFINET-Gerätenummer und die PROFIBUS-Adresse vergeben.

Einen Überblick über die verwendete Gerätenummern und PROFIBUS-Adressen eines IE/PB-Links bekommen Sie im Inspektorfenster unter "Eigenschaften Allgemein > PROFINET Gerätenummer ". Dort können Sie auch die Gerätenummern ändern. Außerdem können Sie einstellen, dass die Gerätenummer und die PROFIBUS-Adresse immer identisch sein sollen oder nicht. Wenn die Option "PROFINET-Gerätenummer=PROFIBUS-Adresse" aktiviert ist, entfällt für Sie das Nachführen der Gerätenummer, wenn sich die PROFIBUS-Adresse ändert.

Die PROFIBUS-Adresse können Sie in den Eigenschaften des PROFIBUS-Geräts ändern.

Einschränkungen

Am PROFIBUS-Subnetz eines IE/PB-Link gelten für DP-Slaves in der oben beschriebenen Konfiguration folgende Einschränkungen:

- Kein IE/PB-Link steckbar
- Kein DP/PA-Link steckbar
- Kein Y-Link steckbar
- Nicht CiR-fähig
- Keine redundanten Slaves steckbar
- Keine Taktsynchronisation / Äquidistanz projektierbar
- SYNC/FREEZE-Anweisungen ("DPSYC_FR") einer CPU am Ethernet-Subnetz für DP-Slaves hinter dem IE/PB-Link werden nicht unterstützt.

Siehe auch

DP-Slave über IE/PB-Link an ein PROFINET IO-System anbinden (Seite 3028)

Konfiguration mit IWLAN/PB-Link

Maximale Anzahl Devices an einem IWLAN Segment

Wenn ein Ethernet-Subnetz als Funknetz aufgebaut ist (IWLAN = Industrial Wireless LAN), dann ist zyklischer Datenaustausch zwischen IO-Controllern und IO-Devices über eine Funkstrecke möglich.

Auf der einen Seite der Funkstrecke befinden sich fest installierte Access Points (z. B. SCALANCE W 788) und auf der anderen Seite mobilen Stationen (mit z. B. IWLAN/PB-Links mit PROFIBUS Teilnehmern).

Wenn der Aktionsradius der mobilen Stationen groß ist, müssen möglicherweise mehrere Access Points (SCALANCE W 788) installiert werden. Jeder Access Point bildet mit seiner Funkreichweite ein Segment, so dass das IWLAN aus einer Reihe von Segmenten aufgebaut ist.

Die mobilen Geräte "auf der anderen Seite" der Funkstrecke mit ihren IWLAN/PB-Links können sich entlang der Segmente bewegen.

Besonderheit

Wenn mehrere IWLAN/PB-Links sich innerhalb eines Segments befinden, dann müssen sie sich die Bandbreite, die für die Funkübertragung zur Verfügung steht, teilen. Das führt zu einer Vergrößerung der Aktualisierungszeit für diese Geräte.

Beispiel

Im Beispiel unten befinden sich zwei IO-Devices (IWLAN/PB-Links) innerhalb eines Segments.

Wenn sich nicht mehr als maximal zwei IWLAN/PB-Links zur gleichen Zeit in einem IWLAN Segment aufhalten, dann tragen Sie eine "2" ein.

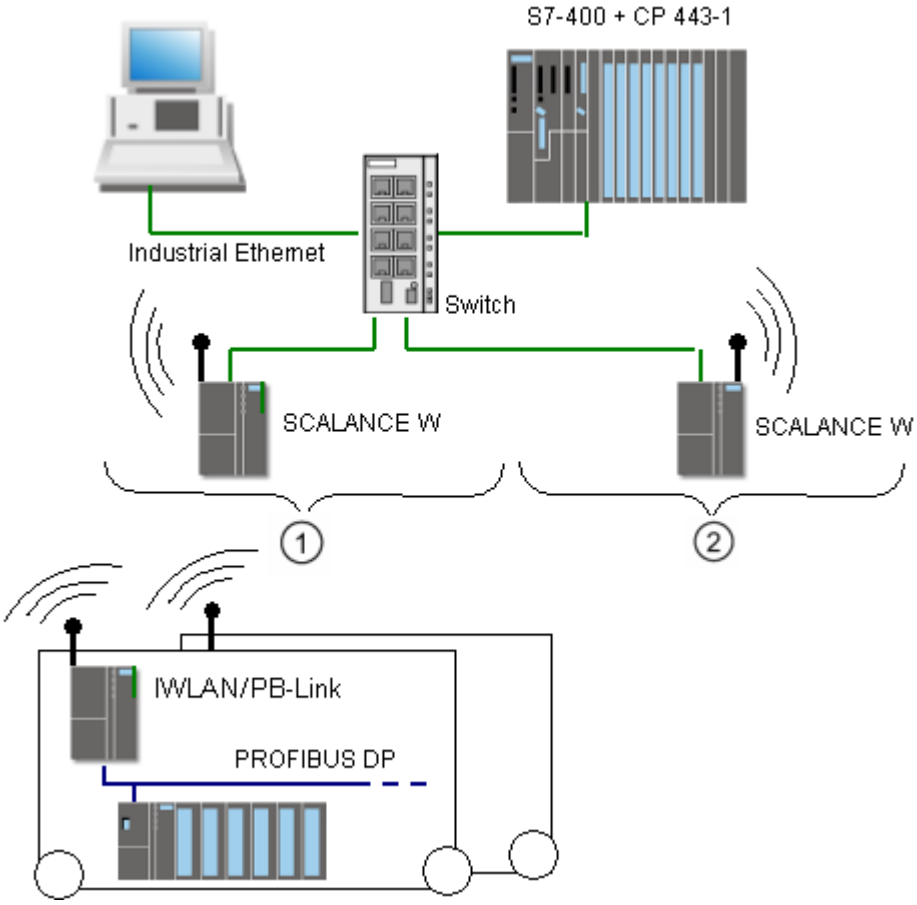

① Segment 1
② Segment 2

1.4.11.2 PROFINET IO konfigurieren

Adressierung von PROFINET-Geräten

Adressen- und Namensvergabe für PROFINET-Geräte

In diesem Kapitel erfahren Sie, welche Adress- und Namenskonventionen für PROFINET-Geräte gelten.

IP-Adressen

Alle PROFINET-Geräte beherrschen das TCP/IP-Protokoll und benötigen daher für den Betrieb am Ethernet eine IP-Adresse.

Die IP-Adressen können Sie in den Baugruppen-Eigenschaften festlegen. Wenn das Netz Teil eines bestehenden Ethernet-Firmennetzes ist, dann erfragen Sie diese Daten von Ihrem Netzwerkadministrator.

Die IP-Adressen der IO-Devices werden automatisch vergeben und normalerweise erst im Anlauf der CPU den IO-Geräten zugewiesen. Die IP-Adressen der IO-Devices haben immer dieselbe Subnetzmaske wie der IO-Controller und werden - ausgehend von der IP-Adresse des IO-Controllers - in aufsteigender Reihenfolge vergeben.

Gerätenamen

Bevor ein IO-Device von einem IO-Controller angesprochen werden kann, muss es einen Gerätenamen haben. Bei PROFINET ist diese Vorgehensweise gewählt worden, weil Namen einfacher zu handhaben sind als komplexe IP-Adressen.

Sowohl IO-Controller als auch IO-Devices haben einen Gerätenamen. Bei aktivierter Option "PROFINET-Gerätename automatisch generieren" wird der Gerätename automatisch aus dem projektierten Namen des Geräts abgeleitet (CPU, CP oder IM):

- Der PROFINET-Gerätename setzt sich aus dem Namen des Geräts (z. B. der CPU), dem Namen der Schnittstelle (nur bei mehreren PROFINET-Schnittstellen) und optional dem Namen des IO-Systems zusammen:
  <CPU-Name>.<Name der Schnittstelle>.<IO-Systemname>
  Diesen Namen können Sie nicht direkt ändern. Sie ändern den PROFINET-Gerätenamen indirekt, indem Sie den Namen der betreffenden CPU, des CPs oder der IM in den Allgemein-Eigenschaften des Moduls ändern. Der PROFINET-Gerätename wird z. B. auch in der Liste der erreichbaren Teilnehmer angezeigt. Wenn Sie den PROFINET-Gerätenamen unabhängig vom Namen des Moduls einstellen wollen, müssen Sie die Option "PROFINET-Gerätename automatisch generieren" deaktivieren.
- Aus dem PROFINET-Gerätenamen wird ein "konvertierter Name" erzeugt. Das ist der Gerätename, der dann tatsächlich in das Gerät geladen wird.
  Der PROFINET-Gerätename wird nur dann konvertiert, wenn er nicht den Regeln von IEC 61158-6-10 entspricht. Auch diesen Namen können Sie nicht direkt ändern.

Regeln für den konvertierten Namen

Im Folgenden sind die Regeln für den konvertierten Namen aufgelistet. Wenn der konvertierte Name sich **nicht** vom Namen der Baugruppen unterscheiden soll, dann muss der Name der Baugruppe diesen Regeln genügen.

- Der Name besteht aus einem oder mehreren Namensbestandteilen (engl: labels), die durch einen Punkt [.] getrennt sind.
- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein.
- Ein Namensbestandteil besteht aus den Zeichen [a-z, 0-9].
- Der Gerätename darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Der Gerätename darf nicht mit Ziffern beginnen.
- Der Gerätename darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Gerätename darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).

Beispiel für Gerätenamen

```
device-1.machine-1.plant-1.vendor
```

Wenn Sie diesen Namen z. B. für eine CPU vergeben, konvertiert STEP 7 ihn nicht, da er den oben beschriebenen Regeln entspricht.

Gerätenummer

Neben dem Gerätenamen wird beim Stecken eines IO-Devices automatisch auch eine Gerätenummer vergeben, die Sie ändern können.

Teilnehmerzahl in einem PROFINET-Subnetz

In einem PROFINET-Subnetz wird die zulässige maximale Anzahl der Teilnehmer während der Projektierung überwacht.

Siehe auch

Gerätename und IP-Adresse zuweisen (Seite 2990)

Remanenz von IP-Adressparametern und Gerätenamen (Seite 3001)

Gerätename und IP-Adresse zuweisen

Erstmaliges Zuweisen von IP-Adresse und Subnetzmaske bei einem IO-Controller

Dazu gibt es verschiedene Möglichkeiten.

Bei der Parametrierung der PROFINET-Schnittstelle müssen Sie einstellen, ob die IP-Adresse im Projekt (d. h. in der Hardware-Konfiguration) eingestellt wird oder ob die IP-Adresse am Gerät eingestellt werden soll.

Zuweisung einer IP-Adresse	Bemerkungen
Option "IP-Adresse im Projekt einstellen": Der IO-Controller erhält die IP-Adresse über das Laden der Hardware-Konfiguration, z. B. über eine der PROFINET-Schnittstellen, über die PROFIBUS-Schnittstelle oder über die MPI-Schnittstelle.	Mit dem Laden der Hardware-Konfiguration in den IO-Controller (z. B. CPU) wird auch die IP-Adresse und, falls eingestellt, der Gerätename geladen. Beispiel PROFINET-Schnittstelle: 1. Schließen Sie Ihr PG/PC am gleichen Netz an, an dem das betreffende PROFINET-Gerät angeschlossen ist. Die Schnittstelle des PGs/PCs muss auf TCP/IP (Auto) eingestellt sein. 2. Lassen Sie sich die Erreichbaren Teilnehmer anzeigen. 3. Wählen Sie das Zielgerät über seine MAC-Adresse aus laden Sie die Hardware-Konfiguration inklusive der projektierten IP-Adresse (IP-Adresse ist dann remanent hinterlegt). Wenn Ihr PROFINET-Gerät über eine MPI- oder PROFIBUS DP-Schnittstelle verfügt, dann schließen Sie Ihren PG/PC über die MPI- oder PROFIBUS DP-Schnittstelle direkt an das PROFINET-Gerät an. Beim Laden der Hardware-Konfiguration wird die projektierte IP-Adresse übernommen.
Option "Anpassen der IP-Adresse direkt am Gerät erlauben": • Online zuweisen • Zuweisen über Anwenderprogramm (Anweisung IP_CONFIG für S7-300/400, T_CONFIG für S7-1200/1500) • Über CPU-Display zuweisen (S7-1500) • Übergeordneter IO-Controller übernimmt Zuweisung (nur bei I-Devices)	Wenn Sie diese Option in den Eigenschaften der PROFINET-Schnittstelle gewählt haben, kann die IP-Adresse über den Online- und Diagnose-Editor, mit dem Primary Setup Tool, oder auch durch das Anwenderprogramm (Anweisung "IP_CONFIG") vergeben werden. Diese Option ist automatisch eingestellt, wenn Sie in den Eigenschaften des PROFINET IO-Systems die Option "Mehrfach einsetzbares IO-System" aktiviert haben (Serienmaschinen-Projekt). Achten Sie bei einer S7-1200-CPU darauf, dass der Zugriff auf die CPU nicht durch ein Passwort geschützt ist. Im Fall einer schreibgeschützten CPU kann keine IP-Adresse und auch kein Gerätename direkt am Gerät zugewiesen werden.

Inbetriebnahme einer PROFINET-Schnittstelle

Weitere Details, wie Sie eine PROFINET-Schnittstelle in Betrieb nehmen, finden Sie auch in den Betriebsanleitungen der PROFINET-Geräte der SIMATIC-Gerätefamilie.

Vergabe des Gerätenamens für IO-Devices bei aktivierter Option "Gerätetausch ohne Wechselmedium ermöglichen"

Bei IO-Controllern mit aktivierter Option "Gerätetausch ohne Wechselmedium ermöglichen" müssen Sie den IO-Devices vor Ort keine Gerätenamen zuweisen, z. B. im Fall eines Gerätetauschs. Ein anderer Anwendungsfall ist die automatische Inbetriebnahme, bei der die CPU automatisch den IO-Devices im Anlauf Gerätename und IP-Adressparameter zuweist.

Voraussetzung: Die Ports der Geräte sind verschaltet und die betreffenden Geräte unterstützten LLDP. Die Geräte sind in den Auslieferungszustand versetzt worden oder - für S7-1500-CPUs ab Version V1.5 - die Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" ist beim IO-Controller aktiviert (Bereich "Erweiterte Optionen > Schnittstellen-Optionen" der Eigenschaften der PROFINET-Schnittstelle).

Vergabe von Gerätename und Adresse bei einem IO-Device

Die folgende Illustration veranschaulicht, wie die Vergabe des Gerätenamens und der Adresse abläuft. Dieser Ablauf gilt nicht bei aktivierter Option "Gerätetausch ohne Wechselmedium ermöglichen").

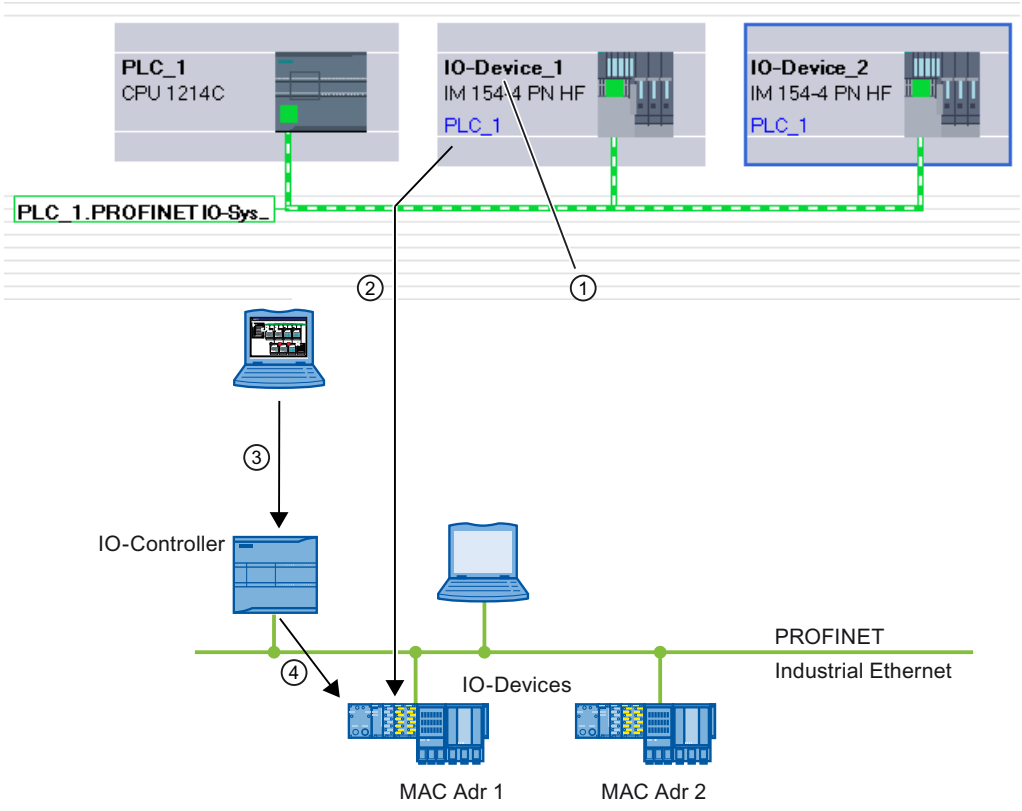

① Jedes Gerät bekommt einen Namen, STEP 7 vergibt automatisch eine IP-Adresse.

② Aus dem Namen erzeugt STEP 7 einen PROFINET-Gerätenamen, den Sie online einem IO-Device (MAC-Adresse) zuordnen und der ins Gerät geschrieben wird.

③ Die Projektierung laden Sie in den IO-Controller.

④ Der IO-Controller vergibt im Anlauf die entsprechende IP-Adresse an das IO-Device mit dem zugewiesenen PROFINET-Gerätenamen.

Gerätename und IP-Adresse ändern

Sie haben die Möglichkeit, Name und IP-Adresse nachträglich manuell zu ändern. Der Gerätename muss zunächst in der Projektierung geändert werden, um ihn anschließend über Memory Card oder online mit PG/PC dem IO-Device zuweisen zu können.

Offline mit Memory Card:

1. Legen Sie die projektierten Daten (Gerätename z. B. turbo-3) für das IO-Device auf der Micro Memory Card im PG/PC ab. Verwenden Sie dazu den Befehl "SIMATIC Card Reader > Gerätenamen auf Memory Card schreiben" im Menü "Projekt".
2. Stecken Sie dann die Micro Memory Card in das IO-Device. Das IO-Device übernimmt automatisch den projektierten Gerätenamen.

Online mit PG/PC:

1. Schließen Sie den PG/PC über die PROFINET-Schnittstelle an das Ethernet-Subnetz an.
2. Selektieren Sie in der Netzsicht das Subnetz oder IO-Device und klicken Sie auf den Befehl "Gerätenamen zuweisen":
   – Entweder im Kontextmenü des selektierten Subnetzes / IO-Devices oder
   – in der Menüleiste der grafischen Sicht auf die entsprechende Schaltfläche.
3. Im Dialog "PROFINET-Gerätename vergeben" wählen Sie die passende PG/PC-Schnittstelle aus, um sich mit dem Ethernet-Subnetz zu verbinden. In der oberen Klappliste sind alle projektierten PROFINET-Gerätenamen wählbar. Wählen Sie dort einen PROFINET-Gerätenamen aus zu markieren Sie in der unteren Tabelle das IO-Device, das diesen Gerätenamen erhalten soll. Sie können die Anzeige der Geräte in der Tabelle nach verschiedenen Kriterien filtern.
4. Über die Schaltfläche "LED blinken" können Sie das Gerät leicht identifizieren.
5. Klicken Sie auf die Schaltfläche "Name zuweisen".

Der IO-Controller erkennt das IO-Device über dessen Gerätenamen und vergibt an das IO-Device automatisch die projektierte IP-Adresse.

IP-Adressvergabe für spezielle IO-Devices

Spezielle IO-Devices, z. B. SCALANCE X, S7-300 CPs, unterstützen die Option, die IP-Adresse nicht vom IO-Controller im Anlauf zuzuweisen. In diesem Fall ist die IP-Adresse auf anderem Wege zu vergeben. Die Option heißt "Anpassen der IP-Adresse direkt am Gerät erlauben". Weitere Informationen finden Sie in dem Gerätehandbuch des betreffenden PROFINET-Geräts der SIMATIC-Gerätefamilie.

Ein weiterer Sonderfall ist die Option "Anpassen der IP-Adresse direkt am Gerät erlauben" im Abschnitt "IP-Protokoll" der Ethernet-Adresseigenschaften eines IO-Devices. Diese Option ist automatisch eingestellt, wenn beim zugehörigen PROFINET IO-System die Option "Mehrfach einsetzbares IO-System" für ein Serienmaschinen-Projekt aktiviert ist. In diesem Fall wird eine angepasste IP-Adresse erst dann durch den IO-Controller zugewiesen, wenn der IO-Controller selbst vor Ort eine IP-Adresse bekommen hat.

Voraussetzung für weitere Verfahren der Vergabe von IP-Adresse und Gerätename

Wenn das IO-Device, wie oben beschrieben, die IP-Adresse bzw. Gerätename nicht vom IO-Controller beziehen soll, gehen Sie wie folgt vor:

1. Wählen Sie die Geräte- oder Netzsicht.
2. Öffnen Sie die Eigenschaften des betreffenden PROFINET-Geräts und wählen Sie dort den Bereich "PROFINET-Schnittstelle [X1]" > "Ethernet-Adressen".
3. Wählen Sie unter "IP-Protokoll" die Option "Anpassen der IP-Adresse direkt am Gerät erlauben" bzw. unter "PROFINET" die Option "Anpassen des PROFINET-Gerätenamens direkt am Gerät erlauben".

Regeln

Wird die Option "Anpassen der IP-Adresse / des Gerätenamens direkt am Gerät erlauben" für ein PROFINET-Gerät verwendet, beachten Sie Folgendes:

- Der Subnetz-Teil der IP-Adresse des IO-Devices muss mit dem Subnetz-Teil der IP-Adresse des IO-Controllers übereinstimmen.
- Das entsprechende PROFINET-Gerät kann nicht als Netzübergang verwendet werden.

Siehe auch

Einen Namen in der Online- und Diagnosesicht vergeben, die über "Erreichbare Teilnehmer" aufgerufen wurde (Seite 3519)

Gerätetausch ohne Wechselmedium aktivieren (Seite 3026)

Beispiel für die Vergabe des Gerätenamens

In diesem Beispiel weisen Sie einem PROFINET IO-Controller und einem PROFINET IO-Device Gerätenamen zu. Die Gerätenamen sollen zur leichten Zuordnung auch den Namen des PROFINET IO-Systems enthalten.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Eine CPU 1214C (ab V2.0) ist in der Netzsicht vorhanden.
- Ein Interfacemodul IM 151-3PN ist vorhanden.
- Die PROFINET-Schnittstellen beider Baugruppen sind vernetzt.

Vorgehen

Um die Namen zu vergeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU.
   Achten Sie darauf, dass Sie nur die CPU selektiert haben und nicht das komplette Gerät!
2. Vergeben Sie im Inspektorfenster, Bereich "Allgemein", den Namen "myController".

3. Selektieren Sie das Interfacemodul.
   Achten Sie darauf, dass Sie nur das Interfacemodul selektiert haben und nicht das komplette Gerät ET 200S!
4. Vergeben Sie im Inspektorfenster, Bereich "Allgemein", den Namen "Device_1"
5. Klicken Sie mit der rechten Maustaste auf das PROFINET IO-System und wählen sie den Befehl "Eigenschaften".

6. Vergeben Sie für das IO-System den Namen "Plant_section1" und aktivieren Sie das Optionskästchen "Namen als Erweiterung für PROFINET-Gerätenamen verwenden".

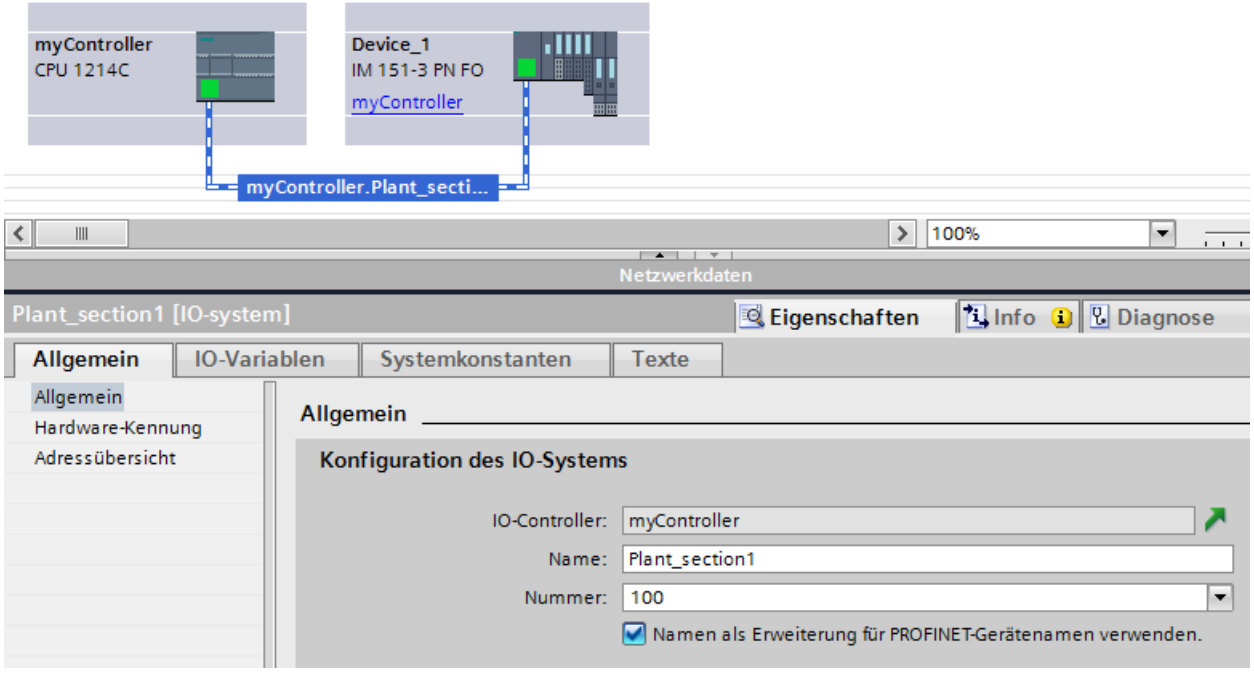

7. Die automatisch gebildeten PROFINET-Gerätenamen finden Sie bei markiertem Gerät im Inspektorfenster, Bereich "PROFINET-Schnittstelle".

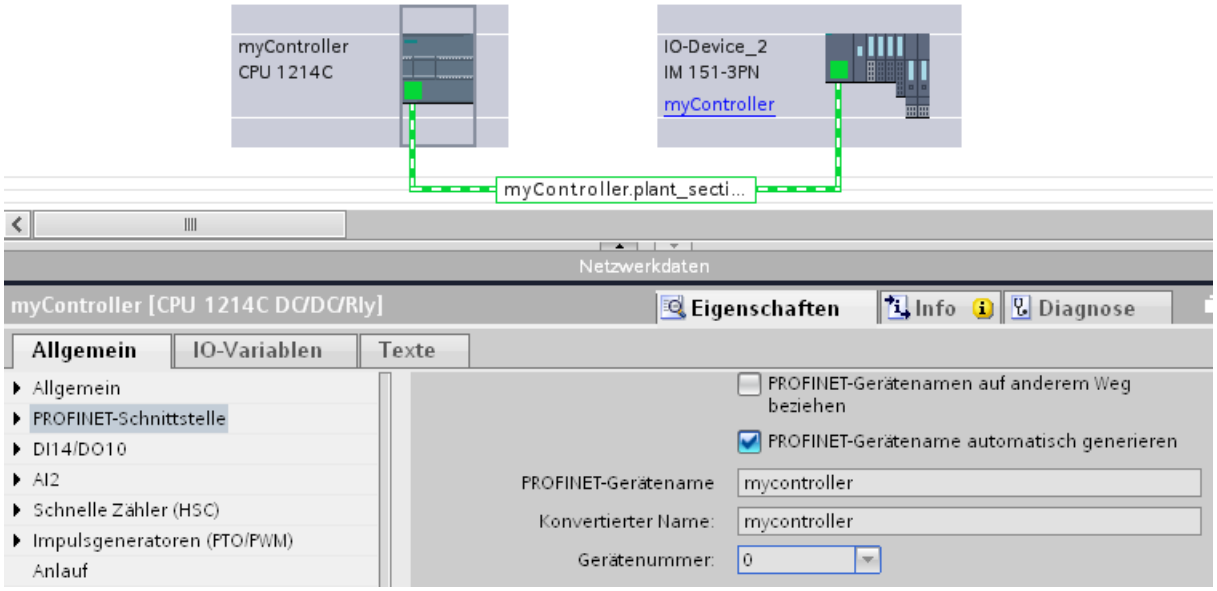

Der PROFINET-Gerätename entspricht dem Namen der Baugruppe (hier mit Erweiterung um den Namen des IO-Systems) mit dem Unterschied, dass nur Kleinschreibung verwendet wird. Hintergrund: Bei der Speicherung des Namens wird nicht zwischen Groß- und Kleinschreibung unterschieden ("case insensitive").

Wenn Sie den Gerätenamen unabhängig vom Namen der Baugruppe bestimmen wollen, müssen Sie die Option "PROFINET-Gerätename automatisch generieren" deaktivieren. In diesem Fall ist der PROFINET-Gerätename editierbar.
Darunter wird der konvertierte Name angezeigt. Das ist der Name, der automatisch aus dem PROFINET-Gerätenamen erzeugt wird und den DNS-Konventionen genügt. Wenn Sie mit STEP 7 arbeiten, benötigen Sie diesen Namen nicht. Dieser Name wird hier zur Kontrolle angezeigt und entspricht dem Namen, wie er im Gerät gespeichert wird. Wenn Sie mit anderen Tools arbeiten, die in der Lage sind, den Datenaustausch aufzuzeichnen und den tatsächlichen Gerätenamen auslesen, dann finden Sie den konvertierten Namen vor.

Weitere Besonderheiten

Bei PROFINET-Geräten mit mehreren PROFINET-Schnittstellen wird der Name der Schnittstelle an den Namen der Baugruppen angehängt, durch einen Punkt getrennt.

Beispiel:

- Name der Baugruppe: myController
- Name der Schnittstelle: Interface_1
- PROFINET-Gerätename: mycontroller.interface_1

Gerätenamen zuweisen über Kommunikationstabelle

Einführung

Sie können die offline projektierten Gerätenamen von PROFINET IO-Devices den Geräten online zuweisen. Dies können Sie im tabellarischen Bereich der Netzsicht in der Tabelle "E/A-Kommunikation" erledigen. Sie können dabei auch mehreren Geräten gleichzeitig die Gerätenamen zuweisen.

Register "Online-Zuweisung"

In der Tabelle E/A-Kommunikation finden Sie die Register "Offline-Konfiguration" und "Online-Zuweisung". Über das Register "Online-Zuweisung" können Sie die offline vergebenen PROFINET-Gerätenamen online den entsprechenden IO-Devices zuweisen. Verwenden Sie dazu die Schaltflächen "Geräte prüfen" und "Jetzt zuweisen".

Die angezeigten Objekte in der Tabelle der Register "Online-Zuweisung" sind abhängig von der Einstellung der Filterfunktion. Wenn nur ausgewählte Objekte angezeigt werden sollen, werden abhängig von der Auswahl in der Netzsicht nur Objekte des entsprechenden Kontextes angezeigt:

- PROFINET-Subnetz: Alle angeschlossenen Geräte und ihre PROFINET-Schnittstellen
- IO-System: Alle beteiligten Geräte und ihre PROFINET-Schnittstellen
- Sync-Domain: Alle beteiligten Geräte und ihre PROFINET-Schnittstellen
- Geräte: Das Gerät und die eventuell vorhandenen PROFINET-Schnittstellen
- Andere Subnetze oder Schnittstellen wie zum Beispiel MPI oder PROFIBUS werden nicht angezeigt

Wenn über die Filterfunktion die Anzeige für alle Geräte eingestellt ist, werden alle Geräte angezeigt, die über eine PROFINET-Schnittstelle verfügen, unabhängig davon, ob sie über ein PROFINET-Subnetz verbunden oder Teil eines IO-Systems sind. Geräte ohne PROFINET-Schnittstelle, zum Beispiel nur mit DP- oder MPI-Schnittstelle, werden nicht angezeigt.

Allgemeine Vorgehensweise

Für die Zuweisung von PROFINET-Gerätenamen müssen Sie zuerst die online verfügbaren IO-Devices ermitteln. Bei der Vorgehensweise kommt es darauf an, ob die MAC-Adressen unbekannt oder bereits bekannt sind. Allgemein ergibt sich daraus eine Vorgehensweise in zwei Schritten:

1. Ermitteln der online verfügbaren IO-Devices
2. Offline konfigurierte PROFINET-Gerätenamen den online verfügbaren IO-Devices zuweisen

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Es besteht eine Online-Verbindung zu den Geräten.

Vorgehen (Schritt 1)

Um online verfügbare IO-Devices aus der E/A-Kommunikationstabelle zu ermitteln, gehen Sie folgendermaßen vor:

1. Optional: Geben Sie in der Spalte "MAC-Adresse" bekannte MAC-Adressen ein. Nach jeder gültigen Eingabe wird das Kontrollkästchen unter "Gerät zuweisen" für die entsprechende Zeile aktiviert.

   Hinweis

   Sie können die MAC-Adresse in unterschiedlichen Formaten eingeben, einfügen oder importieren. Das richtige Format wird automatisch in die Zelle eingetragen. Folgende Eingaben werden unterstützt und anschließend in das benötigte Format konvertiert:
   - "08:00:06:BA:1F:20"
   - "08 00 06 BA 1F 20"
   - "080006BA1F20"

   Die im Beispiel verwendeten Formate werden automatisch konvertiert nach "08-00-06-BA-1F-20".

2. Klicken Sie auf die Schaltfläche "Geräte prüfen", um die Prüfung der online verfügbaren IO-Devices zu starten.
3. Stellen Sie im Dialogfenster die PG/PC-Schnittstelle ein und klicken Sie auf "Start".

Zwischenergebnis

Nach dem Prüfvorgang wird das Ergebnis für jedes Gerät in der Tabelle angezeigt. Gefundene Online-Daten werden automatisch in die Tabelle eingetragen und das Kontrollkästchen "Gerät zuweisen" wird in den Zeilen auf "geprüft" gesetzt, in denen eine MAC-Adresse eingegeben oder online gefunden wurde. In der Spalte "Status" wird das Ergebnis der Prüfung als Icon dargestellt.

Status	Bedeutung
	Übereinstimmung: Passendes Gerät und kompatibler Typ. Das online gefundene Gerät stimmt in PROFINET-Gerätenamen und Gerätetyp mit dem Gerät im Projekt überein. Es ist keine Aktion erforderlich.
	Bereit für Zuweisung: Der Gerätetyp des online gefundenen Gerätes stimmt mit dem Gerät im Projekt überein. Das Gerät ist bei bekannter MAC-Adresse für die Zuweisung des PROFINET-Gerätenamens bereit.
	Gerätetyp unterschiedlich: Es wurde online ein passendes Gerät gefunden, aber der Gerätetyp stimmt nicht mit dem Gerätetyp im Projekt überein. Bitte wählen Sie online ein Geräte mit passendem Gerätetyp.
	Gerät unterschiedlich: Es konnte online kein Gerät gefunden werden, welches dem Gerät im Projekt eindeutig entspricht. Bitte wählen Sie ein manuell ein passendes Gerät.
	Gerät nicht erreichbar: Das Gerät ist bei bekannter MAC-Adresse online nicht erreichbar. Bitte überprüfen Sie die Verbindung zum Gerät.

Hinweis

Das Icon "Bereit für Zuweisung" erscheint, wenn eine MAC-Adresse vorhanden ist und passende Gerätedaten gefunden wurden, aber kein PROFINET-Gerätename online gefunden wurde.

Sie können die Daten der ermittelten Geräte über deren MAC-Adresse jederzeit wieder aktualisieren. Sie müssen dazu die MAC-Adresse angeben und dann wird der Zustand des Gerätes sofort angezeigt, ohne erst das Gerät neu ermitteln zu müssen.

Vorgehen (Schritt 2)

In einer Massenoperation sollen alle offline konfigurierten PROFINET-Gerätenamen den online verfügbaren Geräten zugewiesen werden.

1. Klicken Sie auf die Schaltfläche "Jetzt zuweisen".

   Hinweis

   Die Massenoperation kann nicht rückgängig gemacht werden. In einem Dialogfenster erscheint ein entsprechender Sicherheitshinweis.

2. Klicken Sie im Dialogfenster auf "Start", um die Zuweisung der PROFINET-Gerätenamen zu beginnen.

Ergebnis

Die offline konfigurierten PROFINET-Gerätenamen werden den online verfügbaren Geräten zugewiesen. Dies betrifft die Geräte, bei deren Zeile das Kontrollkästchen unter "Gerät zuweisen" aktiviert ist, die eine MAC-Adresse haben und den Status "Bereit für Zuweisung" haben.

Import und Export der Daten

Sie können über die Import- und Export-Schaltfläche die Daten der E/A-Kommunikationstabelle für die Online-Zuweisung importieren oder exportieren:

- Bei einem Export werden die aktuell angezeigten Daten der Tabelle in eine CSV-Datei exportiert. Über die Filterfunktion der Tabelle können Sie auswählen, welche Daten exportiert werden sollen.
- Bei einem Import werden die Daten der CSV-Datei in die Tabelle geschrieben. Bei Konflikten mit bereits in der Tabelle existierenden Werten können Sie entscheiden, ob die Daten überschrieben werden soll oder der Import abgebrochen werden muss.

Gerätenamen zuweisen über Memory Card

Einführung

Sie können den Gerätenamen von PROFINET IO-Devices offline projektieren. Dazu speichern Sie einen projektierten Gerätenamen auf ein Memory Card und stecken anschließend die Memory Card in das dafür vorgesehene IO-Device.

Wenn im Fall eines Gerätedefekts ein IO-Device komplett getauscht werden muss, führt der IO-Controller automatisch eine Parametrierung und Konfigurierung des neuen Gerätes durch. Mit der Memory Card ist ein Gerätetausch ohne PG möglich.

Voraussetzung

- Das PG verfügt über einen Kartenleser für Memory Cards.
- Das IO-Device muss das Zuweisen des Gerätenamens über Memory Card unterstützen.
- Die Station mit ihrem PROFINET IO-System ist projektiert.

Vorgehen

Um einen Gerätenamen auf einer Memory Card zu speichern, gehen Sie folgendermaßen vor:

1. Stecken Sie die Memory Card in den Kartenleser.
2. Selektieren Sie das IO-Device, dem der Gerätename über Memory Card zugewiesen werden soll.
3. Wählen Sie im Menü "Projekt" den Befehl "Card Reader > Gerätenamen auf Micro Memory Card speichern".
   Wenn die Memory Card nicht leer ist, werden Sie durch eine Meldung auf diesen Umstand aufmerksam gemacht und können die Memory Card vor dem Speichern löschen.

Remanenz von IP-Adressparametern und Gerätenamen

Die Remanenz von IP-Adressparametern (IP-Adresse, Subnetzmaske, Routereinstellung) und Gerätename ist davon abhängig, wie die Adressvergabe erfolgt.

Die nicht remanente temporäre Vergabe bedeutet:

- IP-Adressparameter und Gerätename bleiben über folgenden Zeitraum gültig:
  - bis zum nächsten NETZ-AUS
  - bis zum nächsten Urlöschen
  - bis zum Beenden der Online-Verbindung (z. B. nach Laden des Programms) Nach NETZ-AUS / NETZ-EIN bzw. Urlöschen ist die CPU nur noch über die MAC-Adresse erreichbar.

Wenn die IP-Adressparameter nicht remanent sind, kann nach den oben genannten Ereignissen (z. B. nach NETZ-AUS/NETZ-EIN) keine Kommunikation mehr stattfinden, die auf dem IP-Protokoll basiert.

Die Vergabe einer temporären IP-Adresse löscht auch remanent gespeicherte IP-Adressparameter.

IP-Adressparameter und Gerätename nicht remanent vergeben

IP-Adressparameter und Gerätename sind in folgenden Fällen nicht remanent:

- Eine temporäre IP-Adresse, die nicht remanent ist, wird im Fall der Funktion "Erreichbare Teilnehmer" implizit vergeben, wenn das Gerät (z. B. CPU) noch keine IP-Adresse hat.
- Das Gerät ist ein "normaler" IO-Controller (d. h. kein I-Device) und im Anwenderprogramm (Anweisung "IP_Conf") ist festgelegt, dass die IP-Adressparameter/Gerätename nicht remanent sein sollen.

IP-Adressparameter und Gerätename remanent vergeben

IP-Adressparameter und Gerätename sind in folgenden Fällen remanent:

- In den Eigenschaften der PROFINET-Schnittstelle ist eingestellt, dass die IP-Adresseparameter im Projekt eingestellt sind (Option "IP-Adresse im Projekt einstellen").
- In den Eigenschaften der PROFINET-Schnittstelle ist eingestellt, dass die IP-Adresse am Gerät eingestellt werden soll.
  - Nach dem Laden der Projektierung werden über STEP 7 oder ein Setup-Tool wie PST die IP-Adressparameter bzw. der Gerätename vergeben (STEP 7: Online- und Diagnosefunktion "IP-Adresse zuweisen"). Die zugewiesenen IP-Adressparameter sind remanent.
  - Das Gerät ist ein "normaler" IO-Controller (d. h. kein I-Device) und im Anwenderprogramm (Anweisung "IP_Conf") ist festgelegt, dass die IP-Adressparameter/Gerätename remanent sein sollen.

Besonderheiten beim I-Device

In den Eigenschaften der PROFINET-Schnittstelle des I-Device ist eingestellt, dass die IP-Adressparameter am Gerät eingestellt werden sollen. Die IP-Adressparameter für das I-Device werden durch den übergeordneten IO-Controller zugewiesen.

- Wenn priorisierter Hochlauf eingestellt ist, sind die IP-Adressparameter remanent.
- Wenn **kein** priorisierter Hochlauf eingestellt ist, sind die IP-Adressparameter nicht remanent.

Empfehlung

Verwenden Sie, wenn möglich, die Option "IP-Adresse im Projekt einstellen" und legen Sie eine geeignete IP-Adresse fest. In diesem Fall ist die IP-Adresse remanent vergeben.

Rücksetzen von remanenten IP-Adressparametern und Gerätenamen

Remanente IP-Adressparameter und Gerätenamen werden durch die Online- und Diagnosefunktion "Rücksetzen auf Werkseinstellungen" zurückgesetzt.

Hinweis

Folgen der erneuten Vergabe von IP-Adressparametern auf bestehende IP-Adressparameter

- Durch die temporäre Vergabe von IP-Adressparametern / Gerätenamen erfolgt ein Rücksetzen von ggf. remanent gespeicherten IP-Adressparametern / Gerätenamen.
- Bei einer festen Vergabe von IP-Adressparametern / Gerätenamen werden vorher remanent gespeicherte Parameter durch die neu vergebenen Parameter ersetzt.

Hinweis

Wiederverwendung von Geräten

Führen Sie "Rücksetzen auf Werkseinstellungen" durch, bevor Sie ein Gerät mit remanenten IP-Adressparametern / Gerätenamen in andere Subnetze / Anlagen einbauen oder ins Lager legen.

Router für ein PROFINET IO-Device festlegen

Einleitung

Einen Router (auch "Standard-Gateway") benötigen Sie immer dann, wenn ein PROFINET-Gerät mit Teilnehmern kommunizieren soll, deren IP-Adressen außerhalb des eigenen IP-Subnetzes liegen. Wenn das PROFINET-Gerät ein IP-Paket an eine IP-Adresse außerhalb des eigenen IP-Subnetzes sendet, geht das IP-Paket erst mal an den parametrierten Router. Der Router überprüft wiederum die IP-Adresse. Liegt diese auch außerhalb von seinem Subnetz, dann schickt der Router das IP-Paket an den nächsten Router weiter. Das IP-Paket wird so lange weiter geroutet, bis es die Zieladresse erreicht.

S7-1500-CPUs mit mehreren PROFINET-Schnittstellen bieten wie alle S7-1500-CPUs die Möglichkeit, die IP-Adresse eines Routers zu parametrieren. Allerdings besteht die Einschränkung, dass Sie nur an einer PROFINET-Schnittstelle die IP-Adresse eines Routers eingeben können.

Für die anderen PROFINET-Schnittstellen der CPU können Sie keine IP-Adresse eines Routers mitgeben. IO-Devices, die an dieser PROFINET-Schnittstelle angeschlossen sind, übernehmen diese Einstellung. Bis einschließlich STEP 7 V14 SP1 gibt es für diese IO-Devices keine Möglichkeit, Geräte in einem anderen IP-Subnetz zu erreichen.

Ab STEP 7 V15 haben Sie die Möglichkeit, die Adresse eines Routers für ein IO-Device unabhängig von der Einstellung des IO-Controllers zu vergeben. Jetzt können Sie z. B. auch in folgenden Fällen eine Routeradresse am IO-Device einstellen:

- Sie haben keine IP-Adresse eines Routers für die Schnittstelle des zugehörigen IO-Controllers gesetzt.
- Sie haben bereits für eine andere Schnittstelle in der CPU eine Routeradresse eingestellt.

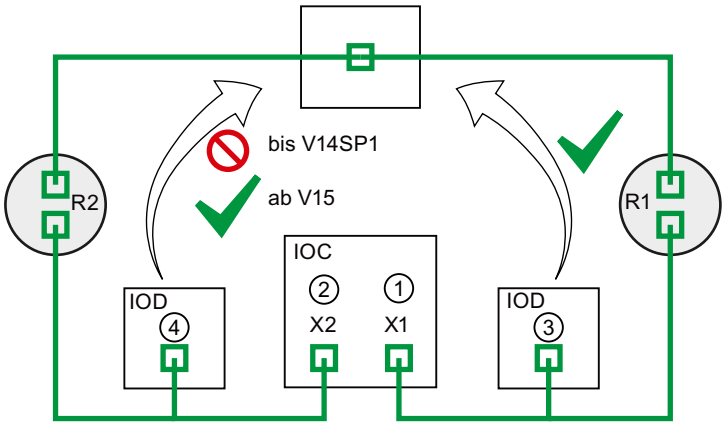

① Router R1 projektiert an PROFINET-Schnittstelle X1

② Wenn an X1 ein Router projektiert ist, dann können Sie an X2 keinen Router projektieren.

③ Weil an X1 ein Router projektiert ist, übernimmt das IO-Device die IP-Adresse des Routers R1. Das IO-Device ist von einem anderen IP-Subnetz aus erreichbar.

④ Ab STEP 7 V15 stellen Sie die IP-Adresse des Routers R2 am IO-Device ein, unabhängig von der Einstellung an Schnittstelle X2. Das IO-Device ist von einem anderen IP-Subnetz aus erreichbar.

Bild 1-90 Router für ein IO-Device festlegen

Wissenswertes zur Einstellung "Router verwenden"

Im Bereich "IP-Protokoll" der Einstellungen zur PROFINET-Schnittstelle (Ethernet-Adressen) haben Sie die Möglichkeit, die Verwendung eines Routers samt IP-Adresse des Routers zu parametrieren.

Regeln

Beachten Sie die folgenden Regeln, wenn Sie die für eine PROFINET-Schnittstelle eines IO-Controllers einen Router konfigurieren:

- Ein PROFINET IO-Gerät unterstützt genau einen Router, unabhängig von der Anzahl Schnittstellen.
- Sie können für genau eine PROFINET-Schnittstelle einen Router konfigurieren. Alle IO-Devices, die dieser PROFINET-Schnittstelle zugeordnet sind, übernehmen den konfigurierten Router vom IO-Controller.
- Für weitere PROFINET-Schnittstellen der CPU können Sie keinen Router konfigurieren. Die weiteren PROFINET-Schnittstellen nehmen als Router die IP-Adresse "0.0.0.0" an und geben diese an ihre IO-Devices weiter.

Ab STEP 7 V15 können Sie für ein IO-Device die Verwendung eines Routers konfigurieren. Damit kann das IO-Device mit einem Teilnehmer außerhalb des eigenen IP-Subnetzes

kommunizieren, unabhängig von der Einstellung der PROFINET-Schnittstelle des IO-Controllers.

Konfigurationsbeispiel: Router für ein IO-Device konfigurieren

Das folgende Beispiel zeigt eine Konfiguration, bei der Sie einen Router am IO-Device konfigurieren, damit das IO-Device IP-Adressen im übergeordneten Netz erreicht.

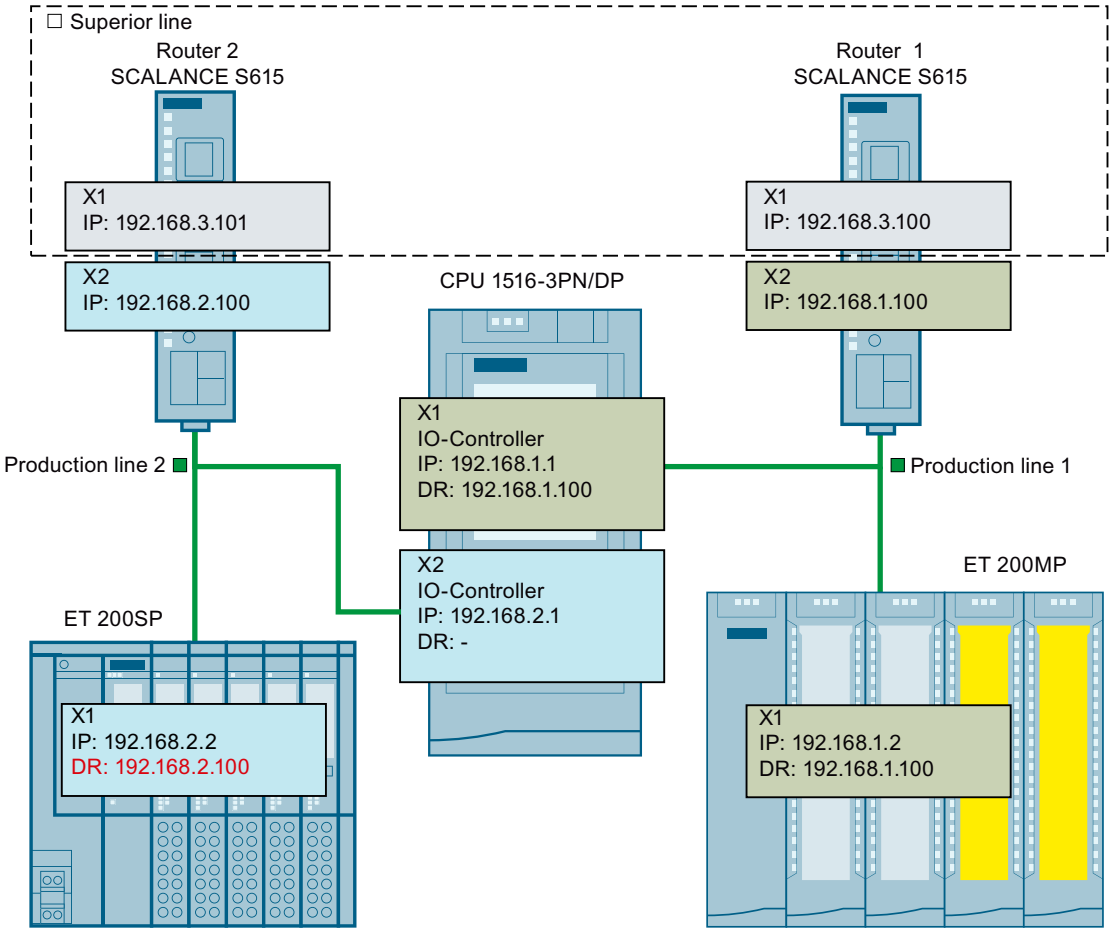

Bild 1-91 Konfigurationsbeispiel: Router für IO-Device konfigurieren

Sie haben eine CPU 1516-3PN/DP. Die beiden PROFINET-Schnittstellen X1 und X2 der CPU arbeiten in der Betriebsart "IO-Controller". Die PROFINET-Schnittstelle X1 ist mit dem Subnetz "Production line 1" verbunden. Die PROFINET-Schnittstelle X2 ist mit dem Subnetz "Production line 2" verbunden. Die beiden Subnetze "Production line 1" und "Production line 2" sind jeweils über einen Router mit dem übergeordneten Netz "Superior line" verbunden.

Für die PROFINET-Schnittstelle X1 konfigurieren Sie den Router "Router 1" mit der IP-Adresse 192.168.1.100.

Das IO-Device (ET 200SP) im Subnetz "Production line 1" übernimmt den Router vom IO-Controller.

Die PROFINET-Schnittstelle X2 können Sie keinen Router konfigurieren, weil Sie bereits für die PROFINET-Schnittstelle X1 der CPU einen Router konfiguriert haben.

Das IO-Device im Subnetz "Production line 2" bekommt keinen Router von der PROFINET-Schnittstelle X2 übergeben.

Damit das IO-Device im Subnetz "Production line 2" Teilnehmer im übergeordneten "Superior line" erreicht, konfigurieren Sie für das IO-Device den Router "Router 2" mit der IP-Adresse 192.168.2.100.

Router für den IO-Controller konfigurieren

Voraussetzung: Sie verwenden für die PROFINET-Schnittstelle die Option "IP-Adresse im Projekt einstellen".

Um einen Router für den IO-Controller in STEP 7 zu konfigurieren, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht von STEP 7 die PROFINET-Schnittstelle des IO-Controllers.
2. Navigieren Sie ins Inspektorfenster zu "Eigenschaften > Allgemein > Ethernet-Adressen".
3. Aktivieren Sie im Feld "IP-Protokoll" das Kontrollkästchen "Router verwenden".
4. Tragen Sie bei "Router-Adresse" die IP-Adresse des Routers ein.

Router für ein IO-Device konfigurieren

Voraussetzungen:

- STEP 7 ab V15
- CPU 1500 ab Firmware-Version V2.5
- IO-Device ist der PROFINET-Schnittstelle eines IO-Controllers zugeordnet. Die PROFINET-Schnittstelle des IO-Controllers verwendet die Option "IP-Adresse im Projekt einstellen".

Um einen Router für ein IO-Device in STEP 7 zu konfigurieren, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht von STEP 7 die PROFINET-Schnittstelle des IO-Devices.
2. Navigieren Sie ins Inspektorfenster zu "Eigenschaften > Allgemein > Ethernet-Adressen".
3. Deaktivieren Sie das Kontrollkästchen "Router-Einstellungen mit IO-Controller synchronisieren".
4. Aktivieren Sie das Kontrollkästchen "Router verwenden".
5. Tragen Sie bei "Router-Adresse" die IP-Adresse des Routers ein.

PROFINET IO-System anlegen

Einführung

Um ein PROFINET IO-System anzulegen, benötigen Sie einen PROFINET IO-Controller und wenigstens ein PROFINET IO-Device. Sobald Sie einen IO-Controller mit einem IO-Device über ihre PROFINET-Schnittstellen verbinden, erfolgt eine Controller-Device-Kopplung.

IO-Controller und IO-Device

Als IO-Controller können Sie folgende Geräte mit PROFINET-Schnittstelle einsetzen:

- CPU mit fest integrierter oder steckbarer PROFINET-Schnittstelle
- CP in Verbindung mit einer CPU
- Schnittstellenmodul, das einer CPU/FM zugeordnet ist
- Anschaltungsbaugruppe mit PROFINET-Schnittstelle

Als IO-Device können Sie Kopfmodule der Dezentralen Peripherie mit PROFINET-Schnittstelle oder CPUs als intelligente IO-Devices einsetzen.

Bei einigen Geräten können Sie im Inspektorfenster zur PROFINET-Schnittstelle unter "Betriebsart" zwischen den möglichen Betriebsarten (wie "IO-Controller" oder "IO-Device") wählen. Selektieren Sie das Optionsfeld mit der benötigten Betriebsart.

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

Um ein PROFINET IO-System beispielsweise mit einer CPU 1217C anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie als potenziellen IO-Controller eine CPU 1217C aus dem Hardware-Katalog.
2. Ziehen Sie per Drag & Drop die CPU auf die freie Fläche in der Netzsicht.
3. Klicken Sie mit der rechten Maustaste auf die PROFINET-Schnittstelle der CPU.
4. Wählen Sie aus dem Kontextmenü "IO-System anlegen".

Es wird ein PROFINET IO-System mit einer CPU 1217C als IO-Controller als einzigem Teilnehmer angelegt.

Wenn Sie die PROFINET-Schnittstelle eines IO-Devices mit der PROFINET-Schnittstelle des IO-Controllers verbinden, wird das IO-Device automatisch in das IO-System aufgenommen. Wenn zwischen IO-Controller und IO-Device noch kein Subnetz existiert, wird ein neues Subnetz zwischen IO-Controller und IO-Device erzeugt.

Um beispielsweise eine CPU 1217C als intelligentes IO-Device in das DP-Mastersystem aufzunehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die PROFINET-Schnittstelle des IO-Controllers oder IO-Devices.
2. Ziehen Sie mit gedrückter Maustaste eine Verbindung von der selektierten PROFINET-Schnittstelle auf die PROFINET-Schnittstelle des gewünschten Kommunikationspartners.

oder

1. Klicken Sie auf den Hyperlink des IO-Devices CPU 1217C.
2. Wählen Sie aus der angezeigten Liste möglicher IO-Controller den gewünschten IO-Controller aus.

Das intelligente IO-Device CPU 1217C wird in das PROFINET IO-System mit der CPU 1217C als IO-Controller aufgenommen.

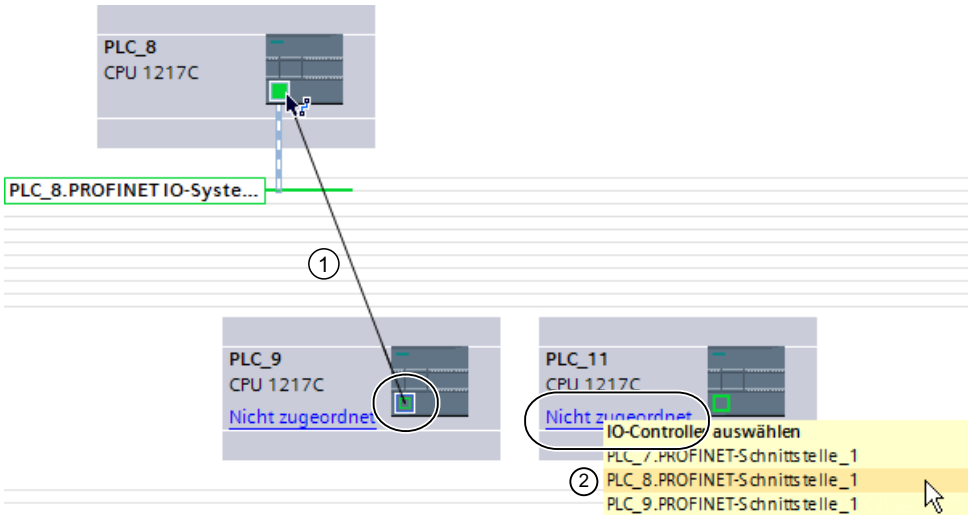

① Drag & Drop zwischen den PROFINET-Schnittstellen von IO-Device zu IO-Controller.

② Klick auf den Link bei nicht zugeordnetem IO-Device öffnet eine Auswahl möglicher IO-Controller.

Passen Sie gegebenenfalls die Eigenschaften des Ethernet-Subnetzes oder des IO-Controllers (z. B. IP-Adresse) im Inspektorfenster unter "Eigenschaften" an.

Hinweis

Wenn sich an einem Subnetz nur ein einzelner IO-Controller befindet, reicht es auch aus, wenn Sie per Drag & Drop die Schnittstelle des IO-Devices mit dem Subnetz dieses IO-Controllers verbinden. Das IO-Device wird dann sofort mit dem Subnetz des IO-Controllers verbunden und das IO-Device wird diesem IO-Controller zugewiesen.

Anzeige des IO-Controllers im IO-Device

Wenn ein IO-Device mit einem IO-Controller verbunden ist, wird der Name des IO-Controllers als Hyperlink am IO-Device angezeigt. Wenn Sie auf den Hyperlink klicken, wird der zugeordnete IO-Controller selektiert.

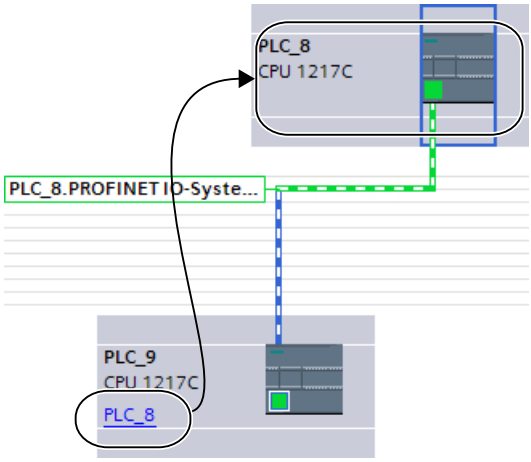

Hervorheben des PROFINET IO-Systems

Wenn Sie ein neues PROFINET IO-System angelegt haben, wird das PROFINET IO-System hervorgehoben dargestellt. Dadurch erkennen Sie schnell, welche Geräte zu dem PROFINET IO-System gehören. Sie können ein PROFINET IO-System auch hervorheben, indem Sie den Mauszeiger über ein Subnetz ziehen. Es werden daraufhin die Namen der vorhandenen PROFINET IO-Systeme angezeigt. Wenn Sie mit der Maus auf eines der angezeigten PROFINET IO-Systeme klicken, wird das entsprechende PROFINET IO-System hervorgehoben.

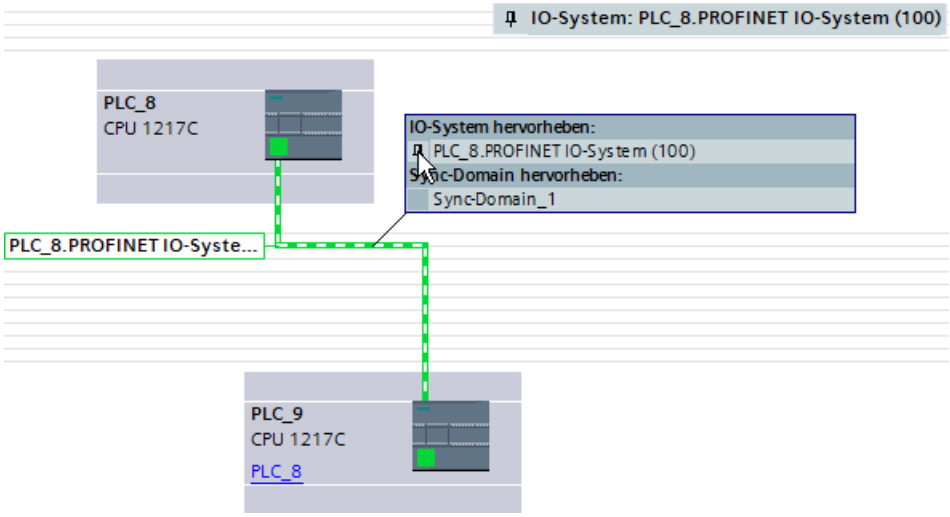

Um die Hervorhebung eines PROFINET IO-Systems wieder auszublenden, stehen Ihnen mehrere Möglichkeiten zu Verfügung:

- Heben Sie ein anderes PROFINET IO-System hervor.
- Klicken Sie auf die Pin-Nadel mit der Bezeichnung des PROFINET IO-Systems in der rechten oberen Ecke der Netzsicht.

PROFINET IO-System hantieren

Über die Kontextmenübefehle haben Sie die Möglichkeit, PROFINET IO-Systeme in der Netzsicht zu löschen, neu anzulegen oder auch die Schnittstelle mit einem anderen Subnetz zu verbinden.

Auf diese Weise korrigieren Sie eine bestehende PROFINET-Konfiguration in der Netzsicht.

Neues PROFINET IO-System für IO-Controller anlegen

Um für einen IO-Controller ein neues PROFINET IO-System anzulegen, gehen Sie folgendermaßen vor:

1. Vergewissern Sie sich, dass dem IO-Controller kein IO-System zugewiesen ist. Falls dem IO-Controller bereits ein IO-System zugewiesen ist, dann ist der Kontextmenübefehl "IO-System zuweisen" nicht aktiv.
2. Selektieren Sie die PROFINET-Schnittstelle und wählen Sie den Kontextmenübefehl "IO-System zuweisen".

Ein neues PROFINET IO-System wird am IO-Controller angelegt und sie können IO-Devices diesem IO-System zuweisen.

PROFINET IO-Devices vom PROFINET IO-System trennen

Um bereits vernetzte PROFINET IO-Devices von ihrem PROFINET IO-System zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die PROFINET-Schnittstelle eines IO-Devices.

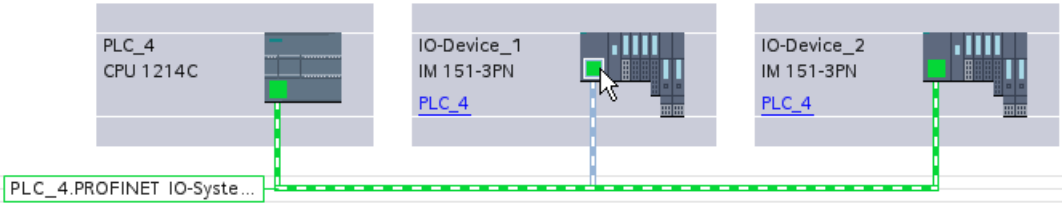

2. Wählen Sie den Kontextmenübefehl "Vom IO-System trennen".
   Das IO-Device, das diesem IO-System zugewiesen war, ist danach nicht mehr zugeordnet.

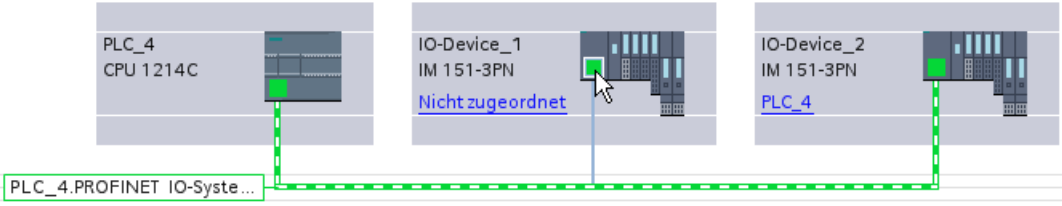

Sie haben die Möglichkeit, ein neues IO-System anzulegen und die nicht zugeordneten IO-Devices jeweils einem IO-Controller zuzuweisen.

PROFINET IO-Devices anderen IO-Controllern zuweisen

Bestehende PROFINET IO-Systeme können Sie in der Netzsicht einfach umkonfigurieren, indem Sie das IO-Device oder seine Schnittstelle selektieren und den gewünschten Befehl aus dem Kontextmenü auswählen:

Bei Auswahl der Schnittstelle:

1. Selektieren Sie die Schnittstelle eines IO-Devices und wählen Sie das Kontextmenü. Hier haben Sie folgende Möglichkeiten:
   - Das IO-Device einem neuen Subnetz zuweisen oder vom bestehenden Subnetz trennen
   - Das IO-Device einem neuen IO-Controller zuweisen
   - Das IO-Device einem neuen IO-System zuweisen oder vom bestehenden IO-System trennen.
2. Um das IO-Device einem andern IO-Controller zuzuweisen, wählen Sie den Kontextmenübefehl "Neuem IO-Controller zuweisen".
   Im Fall einer fehlenden Vernetzung wird automatisch ein Subnetz angelegt und das IO-Device dem IO-System des neuen IO-Controllers zugewiesen.

Bei Auswahl des ganzen Gerätes:

1. Selektieren Sie das IO-Device und wählen Sie das Kontextmenü. Hier haben Sie folgende Möglichkeiten:
   - Das IO-Device einem neuen IO-Controller zuweisen
   - Das IO-Device vom bestehenden Subnetz trennen
2. Um das IO-Device einem andern IO-Controller zuzuweisen, wählen Sie den Kontextmenübefehl "Neuem DP-Master/IO-Controller zuweisen".
   Im Fall einer fehlenden Vernetzung wird automatisch ein Subnetz angelegt und das IO-Device dem IO-System des neuen IO-Controllers zugewiesen.

IO-Device konfigurieren über die Hardware-Erkennung

Einleitung

Ab STEP 7 V15 haben Sie die Möglichkeit, ein real vorhandenes IO-Device zu erkennen und in Ihr Projekt zu übernehmen.

Sie erkennen das IO-Device in STEP 7 über die Funktion "Hardware-Erkennung". Ein erkanntes Gerät können Sie in Ihr Projekt übernehmen. STEP 7 fügt das IO-Device mit allen Modulen und Submodulen ein.

Beachten Sie, dass Geräte der ET 200AL nicht von der Hardware-Erkennung erkannt werden.

Hardware-Erkennung von IE/PB-Links

Die Hardware-Erkennung erkennt nur Geräte, die eine Rolle als IO-Device haben. Geräte und andere Komponenten ohne diese Rolle, zum Beispiel auch IO-Controller, I-Devices oder Geräte mit Zuordnungen zu mehreren IO-Controllern (Mehrfachzuweisung), werden bei der Hardware-Erkennung nicht berücksichtigt. IE/PB-Links sind Geräte mit Mehrfachzuweisung und werden daher bei der Hardware-Erkennung nicht angezeigt.

Gerätenamen von PN/PN-Kopplern

Bei der Hardware-Erkennung von PN/PN-Kopplern und deren anschließenden Integration in die Projektkonfiguration, müssen die Namen der PN/PN-Koppler besonders berücksichtigt werden. Diese PROFINET-Devices haben im Unterschied zu anderen IO-Devices ein zusätzliches Attribut für die automatische Generierung der Gerätenamen. Beim Hinzufügen eines PN/PN-Kopplers zu einer Konfiguration aus der Hardware-Erkennung heraus, verhält sich der Gerätename genauso, also ob der PN/PN-Koppler aus dem Hardware-Katalog der Konfiguration hinzugefügt worden wäre.

Der Gerätename des PN/PN-Kopplers wird also nach dem Hinzufügen aus der Hardware-Erkennung automatisch generiert und muss ggf. nachträglich manuell angepasst werden.

Voraussetzungen

- STEP 7 (TIA Portal) ab V15
- Neues oder vorhandenes Projekt muss geöffnet sein
- IO-Devices müssen IP-technisch erreichbar sein

Vorgehen

Um ein oder mehrere vorhandene IO-Devices in STEP 7 zu erkennen und in das Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in STEP 7 zu "Online" > "Hardware-Erkennung".
2. Klicken Sie auf "PROFINET-Devices aus Netzwerk...".
   STEP 7 öffnet das Fenster "Hardware-Erkennung der PROFINET-Devices".
3. Wählen Sie bei "PG/PC-Schnittstelle" die Schnittstelle Ihres PGs/PCs aus.
4. Klicken Sie auf "Suche starten".
   STEP 7 beginnt mit der Hardware-Erkennung. Wenn die Hardware-Erkennung abgeschlossen ist, zeigt STEP 7 alle erkannten IO-Devices an.
5. Wählen Sie die IO-Devices aus, die Sie in das Projekt übernehmen wollen, indem Sie auf das entsprechende Kontrollkästchen vor dem IO-Device klicken.
6. Klicken Sie auf "Geräte hinzufügen".
   Nach kurzer Zeit öffnet sich ein Fenster, das über den Erfolg bzw. Misserfolg der Hardware-Erkennung berichtet.

Ergebnis der Hardware-Erkennung

Bei erfolgreicher Hardware-Erkennung übernimmt STEP 7 das IO-Device mit allen Modulen und Submodulen in das Projekt.

Ein über Hardware-Erkennung konfiguriertes IO-Device verhält sich wie folgt:

- Über die "Hardware-Erkennung" konfigurierte Module sind so parametriert, als hätten Sie diese aus dem Katalog eingefügt.
- MAC-Adresse: STEP 7 übernimmt die MAC-Adresse des erkannten IO-Devices in das Projekt.
- IP-Einstellungen:
  - Wenn das erkannte IO-Device bereits eine IP-Adresse besitzt, dann übernimmt STEP 7 die IP-Adresse in das Projekt.
  - Wenn das erkannte IO-Device keine IP-Adresse besitzt, dann vergibt STEP 7 automatisch eine IP-Adresse im Projekt.

  PROFINET-Gerätenamen:
  - Wenn das erkannte IO-Device bereits einen PROFINET-Gerätenamen besitzt, dann übernimmt STEP 7 den PROFINET-Gerätenamen in das Projekt.
  - Wenn das erkannte IO-Device keinen PROFINET-Gerätenamen besitzt, dann vergibt STEP 7 automatisch einen PROFINET-Gerätenamen im Projekt.
- Über "Hardware-Erkennung" konfigurierte IO-Devices sind weder einem IP-Subnetz, noch einem IO-Controller zugeordnet.

Siehe auch

Nicht spezifizierte CPU hinzufügen (Seite 43)

Tipp: Schnelles Konfigurieren von IO-Systemen

Wenn das IO-System viele IO-Devices hat, weisen Sie alle per Drag & Drop platzierten IO-Devices in einem Schritt einem IO-Controller zu.

Voraussetzung

IO-Controller und IO-Devices sind in der Netzsicht platziert.

IO-Devices einem IO-System zuweisen

Gehen Sie folgendermaßen vor:

1. Wählen Sie einen passenden Zoomfaktor, so dass Sie möglichst viele IO-Devices in der Netzsicht sehen.
2. Ordnen Sie die IO-Devices in maximal zwei Reihen an.

3. Wählen Sie mit dem Mauszeiger alle IO-Schnittstellen aus (nicht die Geräte!). Das gelingt nur, wenn Sie außerhalb des ersten IO-Devices mit dem Ziehen des Mauszeigers beginnen und beim letzten IO-Device die Maustaste loslassen (Auswahl mit dem Lasso).

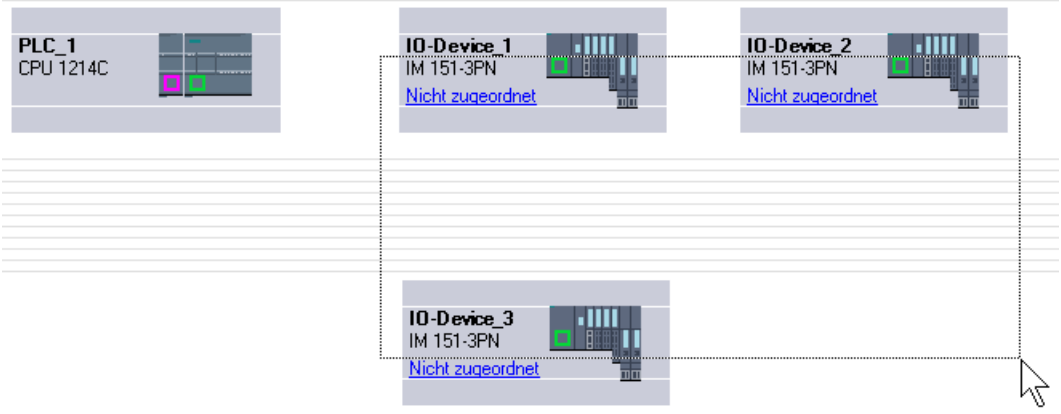

4. Wählen Sie das Kontextmenü "Neuem IO-Controller zuweisen" und markieren Sie im Folgedialog die entsprechende IO-Schnittstelle des IO-Controllers.

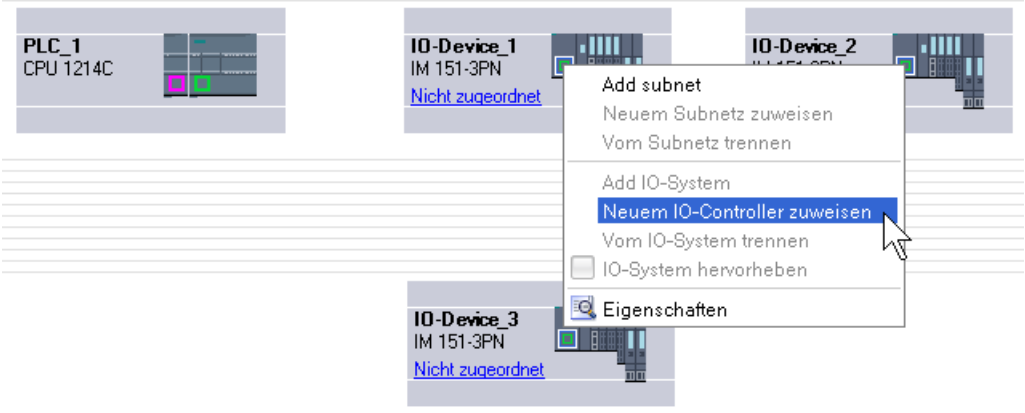

5. Die IO-Devices werden automatisch mit dem IO-Controller vernetzt und bilden mit ihm zusammen ein IO-System.

Hinweis

Bei einem hervorgehobenem IO-System können Sie im Hardware-Katalog auf ein IO-Device doppelklicken und damit schnell weitere IO-Devices hinzugügen. Das IO-Device wird dadurch automatisch in das hervorgehobene IO-System aufgenommen.

Ports verschalten

Wenn ein IO-Device einem IO-Controller zugewiesen ist, dann ist damit noch nicht festgelegt, wie die Ports untereinander verschaltet sind.

Eine Portverschaltung ist nicht notwendig, um die Ethernet/PROFINET-Funktionalität nutzen zu können, bietet aber folgende Vorteile:

- Mit der Portverschaltung wird eine Solltopologie vorgegeben. Anhand eines Online-Offline-Vergleichs ist es möglich, einen Soll-Ist-Vergleich durchzuführen bei den Geräten, die diese Funktionalität unterstützen.
- Nur bei IRT-Kommunikation: Wenn eine Portverschaltung projektiert ist, kann STEP 7 präziser die notwendige Bandbreite ermitteln. Das führt in der Regel zu einer höheren Performance.

Beachten Sie, dass durch die Verschaltung von Ports keine unzulässigen Ringstrukturen entstehen.

Sinnvoll ist die Portverschaltung nur für Geräte, die die Topologieprojektierung unterstützen.

Ports im Inspektorfenster verschalten

Um Ports miteinander zu verschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ethernet/PROFINET-Gerät oder die Ethernet/PROFINET-Schnittstelle.
2. Navigieren Sie zur Port-Eigenschaft "Portverschaltung".
   Bei markierter Ethernet/PROFINET-Schnittstelle ist diese Einstellung wie folgt im Inspektorfenster zu finden: Eigenschaften > Allgemein > Erweiterte Optionen > Port [...] > Portverschaltung.
3. Im Abschnitt "Lokaler Port" sind die Einstellungen am lokalen Port zu finden. Für Lichtwellenleiter sind hier die z. B. Kabelbezeichnungen einstellbar.
   Im Abschnitt "Partnerport" klicken Sie auf das schwarze Dreieck im Feld "Partnerport", um sich die verfügbaren Partnerports anzeigen zu lassen und auszuwählen.
4. Wenn es sich um eine Portverschaltung mit dem Medium Kupfer handelt und die Geräte IRT-Kommunikation unterstützen, können Sie noch Leitungslänge bzw. Signallaufzeit einstellen.

Wenn die Ethernet/PROFINET-Schnittstelle nicht vernetzt war, wird sie durch diese Aktion automatisch vernetzt. In den Eigenschaften des Subnetzes können Sie einstellen, ob dieses Subnetz für die Vernetzung herangezogen werden soll oder nicht.

Hinweis

Verschaltung eines elektrischen mit einem optischen Port

Wenn Sie einen elektrischen und einen optischen Port miteinander verschalten wolllen, müssen Sie zwischen RT- und IRT-Kommunikation unterscheiden:

- Bei RT-Kommunikation ist die Projektierung eines Medienkonverters nicht erforderlich.
- Bei IRT-Kommunikation müssen Sie die Verschaltung über einen Medienkonverter vornehmen.

Information über Überwachung des Partnerports

Nachdem Sie zwei Ports miteinander verschaltet haben, erhalten Sie im Abschnitt "Partnerport" in einem Textfeld eine Information über die Überwachung des Partnerports. Folgende Anzeigen sind möglich:

- Überwachung des Partnerports ist nicht möglich.
- Überwachung des Partnerports wird durchgeführt.

Wenn Sie als Partnerport z. B. einen deaktivierten Port wählen, ist dessen Überwachung nicht möglich. In diesem Fall ist die Überwachung der Solltopologie und der Signallaufzeit nicht möglich. Ein Gerätetausch ist dann nur mit einer Micro Memory Card möglich.

Siehe auch

Übersicht (Seite 314)

Übersicht über die CPU-Eigenschaften (Seite 988)

Sendetakt einstellen

Voraussetzungen zum Ändern des Sendetakts am PROFINET-Gerät

Das PROFINET-Gerät muss ein IO-Controller als Sync-Master sein oder es darf im IO-System kein IRT (Isochronous Realtime) projektiert sein.

Wenn das PROFINET-Gerät kein IO-Controller als Sync-Master ist, müssen für die Einstellung des Sendetaktes die folgenden Voraussetzungen erfüllt sein:

- Es darf am IO-System kein Gerät als Sync-Slave oder Sync-Master projektiert sein.
- Alle Geräte am IO-System müssen unsynchronisiert sein.

Wenn IRT mit einem IO-Controller als Sync-Master projektiert ist, kann der Sendetakt am Gerät oder in der Sync-Domain eingestellt werden.

Vorgehen

Um den Sendetakt am PROFINET-Gerät einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den PROFINET IO-Controller in der Geräte- oder Netzsicht.
2. Ändern Sie in den Eigenschaften der PROFINET-Schnittstelle den Wert für das kleinst mögliche Aktualisierungsintervall unter "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > IO-Kommunikation > Sendetakt".

Der Sendetakt ist für alle PROFINET-Geräte am IO-System gültig. Wenn Sie die Synchronisationsrolle auf einen anderen Wert als "Unsynchronisiert", dann können Sie den Sendetakt auch in der Sync-Domain, d. h. zentral am PROFINET IO-System einstellen:

1. Selektieren Sie das IO-System über das PROFINET-Subnetz.
2. Ändern Sie in den Eigenschaften der Sync-Domain den Wert für den Sendetakt unter "Domain-Management > Sync-Domains > Sync-Domain_1 > Sendetakt".

Aktualisierungszeit einstellen

Aktualisierungszeit

Innerhalb dieses Zeitintervalls wird ein IO-Device/IO-Controller im PROFINET IO-System vom IO-Controller/IO-Device mit neuen Daten versorgt. Die Aktualisierungszeit kann für jedes IO-Device separat projektiert werden und bestimmt den Zeitabstand, in dem Daten vom IO-Controller zum IO-Device (Ausgänge) sowie Daten vom IO-Device zum IO-Controller (Eingänge) gesendet werden.

In der Voreinstellung berechnet STEP 7 die Aktualisierungszeit automatisch für jedes IO-Device des PROFINET IO-Systems unter Berücksichtigung des auszutauschenden Datenvolumens und des eingestellten Sendetakts.

Aktualisierungszeit einstellen

Wenn Sie die Aktualisierungszeit nicht automatisch berechnen lassen möchten, können Sie die Einstellung ändern.

Um die Aktualisierungszeit im Inspektorfenster zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht oder in der Gerätesicht die PROFINET-Schnittstelle des IO-Devices.
2. Bestimmen Sie in den Eigenschaften der Schnittstelle unter "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Zyklus" die Aktualisierungszeit. Sie können die Aktualisierungszeit auf zwei verschiedenen Wegen bestimmen:
   - Automatische Berechnung der optimalen Aktualisierungszeit.
   - Selber einstellbare Aktualisierungszeit mit Auswahl verschiedener Werte aus einer Klappliste.
3. Wenn Sie das Verhältnis zwischen Sendetakt und Aktualisierungszeit konstant halten wollen, aktivieren Sie die Option "Aktualisierungszeit bei Änderung des Sendetakts anpassen".
   Mit dieser Option ist sichergestellt, dass die Aktualisierungszeit nicht kleiner eingestellt ist als der Sendetakt.

Sie können diese Einstellung auch direkt in der E/A-Kommunikationstabelle über die Klappliste in der Spalte "Aktualisierungszeit-Modus" vornehmen.

Die manuelle Einstellung des Sendetakts kann zu Fehlern führen, wenn die zur Verfügung stehende Bandbreite nicht ausreicht oder wenn andere Grenzen/Mengengerüste überschritten werden (z. B. wenn zu viele Teilnehmer projektiert sind).

Hinweis

Wenn Sie den Aktualisierungszeit-Modus von "Einstellbar" nach "Automatisch" ändern, wird eine manuell eingetragene Aktualisierungszeit durch den automatisch berechneten Wert überschrieben.

Keine Aktualisierungszeit bestimmbar

Aus den Projektierungsinformationen (Eigenschaften IO-Controller, Eigenschaften IO-Devices, Anzahl und Art der IO-Devices, Konsistenz der zyklischen Nutzdaten...) bestimmt STEP 7 den Ablauf des zyklischen Datenaustauschs. Dabei werden die zyklischen Daten in Frames gepackt und nacheinander in errechneten zeitlichen Intervallen gesendet/empfangen.

Die maximale Anzahl/Größe der Frames und die maximal zur Verfügung stehende Anzahl der Intervalle und müssen ausreichen, um alle Daten "unterzubringen". Das resultierende Sende-/Empfangsintervall muss außerdem von jedem PROFINET-Gerät unterstützt werden.

Wenn die Grenzen hinsichtlich der Anzahl zyklischer Nutzdaten/Frames oder hinsichtlich der zur Verfügung stehenden zeitlichen Intervalle überschritten wurde, kann STEP 7 keine Aktualisierungszeit bestimmen.

Auch wenn keine gemeinsame Basis für das Sende-/Empfangsintervall existiert, ist die Kalkulation der Aktualierungszeit nicht möglich.

Falls ein Hinderungsgrund für die Berechnung der Aktualisierungszeit vorliegt, meldet STEP 7 die Ursache bei der Übersetzung der Hardware-Konfiguration.

Maßnahmen zur Behebung:

- Reduzieren Sie die Anzahl von IO-Devices
- Reduzieren Sie die Anzahl von Modulen innerhalb der IO-Devices
- Bei Verwendung eines IE/PB Links: Reduzieren Sie die Anzahl von DP-Slaves hinter dem IE/PB Link
- Verwenden Sie einen leistungsfähigeren IO-Controller bzw. IE/PB Link
- Erhöhen Sie den Sendetakt
  - bei RT: in den Eigenschaften des IO-Controllers
  - bei IRT: in den Eigenschaften der Sync-Domain
- Kontrollieren Sie die Geräteeigenschaften der IO-Devices ('MinDeviceIntervall' und die möglichen Untersetzungen) wegen der gemeinsamer Basis für Sende-/Empfangsintervall. Tauschen Sie nicht passende IO-Devices aus. Die Geräteeigenschaften sind in der GSD-Datei des IO-Devices hinterlegt.

- Bei IRT-Projektierung:
  - Überprüfen Sie, ob die Ports von Sync-Master und Sync-Slaves miteinander verschaltet sind.
  - Überprüfen Sie die Reihenfolge der IO-Devices: Zwischen Sync-Master und Sync-Slave darf **kein** unsynchronisiertes Gerät geschaltet sein
    (Beispiel für fehlerhafte Projektierung: Sync-Master --- Unsynchronisiertes Gerät --- Sync-Slave).
  - Überprüfen Sie, ob Sie mehr als einen Sync-Master projektiert haben.
  - Überprüfen Sie die verbleibende Bandbreite für RT-Daten. Die für die Übertragung zur Verfügung stehende Bandbreite für RT-Daten kann durch IRT-Kommunikation am selben Ethernet-Subnetz eingeschränkt sein.
- Bei Verwendung von I-Devices:
  Möglicherweise kann der eingestellte Sendetakt nicht gemeinsam mit der vorliegenden I-Device-Konfiguration verwendet werden.
  - Konfigurieren Sie das I-Device ohne unterlagerte IO-Devices und aktivieren Sie die Einstellung "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller".
  - Ändern Sie den Sendetakt des IO-Controllers auf einen geraden Wert (... 0,250, 0,500, 1,000, ...).

Identifikation der betroffenen IO-Devices:

Sie identifizieren die betroffenen IO-Devices, für die keine Aktualisierungszeit berechnet werden kann, in der Tabelle "E/A-Kommunikation" des PROFINET IO-Systems in der Netzsicht. In der Spalte "Aktualisierungszeit" ist bei den betroffenen IO-Devices kein Wert in der Spalte "Aktualisierungszeit" eingetragen (Eintrag "-").

Ansprechüberwachungszeit einstellen

Ansprechüberwachungszeit

Für PROFINET IO-Devices können Sie eine Ansprechüberwachungszeit projektieren.

Wenn das IO-Device nicht innerhalb der Ansprechüberwachungszeit vom IO-Controller mit Ein-oder Ausgangsdaten (IO-Daten) versorgt wird, schaltet es in den sicheren Zustand.

Sie geben die Ansprechüberwachungszeit nicht direkt ein, sondern als "Anzahl akzeptierter Aktualisierungszyklen ohne IO-Daten". Das macht die Einstellung einfacher, weil die Aktualisierungszeit je nach Leistungsfähigkeit des IO-Devices bzw. je nach Einstellung kürzer oder länger sein kann.

Aus der "Anzahl akzeptierter Aktualisierungszyklen ohne IO-Daten" wird die resultierende Ansprechüberwachungszeit automatisch berechnet.

Ansprechüberwachungszeit projektieren

Um die Ansprechüberwachungszeit festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht oder in der Gerätesicht die PROFINET-Schnittstelle des IO-Devices.
2. Navigieren Sie in den Eigenschaften der Schnittstelle zu "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Zyklus".
3. Wählen Sie aus der Klappliste "Ansprechüberwachung nach # Zyklen mit fehlenden IO-Daten auslösen" die gewünschte Anzahl Zyklen aus.

Die Ansprechüberwachungszeit wird anschließend auf Grund des vorgegebenen Faktors automatisch berechnet. Sie darf nicht mehr als 1,92 Sekunden betragen.

Hinweis

Die Voreinstellung sollten Sie nur im Ausnahmefall ändern, z. B. in der Inbetriebnahmephase.

Berechnete Bandbreite für zyklische IO-Daten

Berechnete Bandbreite für zyklische IO-Daten

Das System überwacht die Einhaltung der maximal zur Verfügung stehenden Bandbreite für zyklische IO-Daten. Die maximale Bandbreite ist vom Sendetakt abhängig. Bei Sendetakten größer oder gleich 1 ms beträgt die maximale Bandbreite 0,5 ms. Bei kleineren Sendetakten verringert sich die maximal zur Verfügung stehende Bandbreite.

Die tatsächlich benötigte Bandbreite für zyklische IO-Daten wird vom System auf Basis der Anzahl der projektierten IO-Geräte und IO-Baugruppen bestimmt. Des Weiteren ist die benötigte Bandbreite von der verwendeten Aktualisierungszeit abhängig.

Im Allgemeinen erhöht sich die berechnete Bandbreite bei:

- einer größeren Anzahl von IO-Geräten
- einer größeren Anzahl von IO-Baugruppen
- kleineren Aktualisierungszeiten.

Maximale Bandbreite für zyklische IO-Daten in Abhängigkeit vom Sendetakt

Die folgende Tabelle zeigt, wie sich die maximal zur Verfügung stehende Bandbreite für zyklische IO-Daten in Abhängigkeit zum Sendetakt verhält:

Sendetakt	Maximale Bandbreite für zyklische IO-Daten
250 µs – 468,75 µs	<< 125 µs
500 µs – 968,75 µs	= Sendetakt / 2
1 – 4 ms	= 500 µs

Port-Optionen festlegen

Festlegen der Port-Optionen

Verbindungseinstellungen für den PROFINET-IO-Port ändern

Standardmäßig sind die Einstellungen für Ports einer PROFINET-Schnittstelle so festgelegt, dass im Normalfall ein problemloser Datenaustausch gewährleistet ist.

Bei Bedarf können Sie die Einstellungen ändern. Die Einstellmöglichkeiten sind im folgenden beschrieben.

Einstellmöglichkeiten Übertragungsrate/Duplex

Abhängig von dem gewählten Gerät können Sie folgende Einstellungen für "Übertragungsrate/Duplex" vornehmen:

- Automatische Einstellung
  Empfohlene Default-Einstellung des Ports. Die Übertragungseinstellungen mit dem Partnerport werden automatisch "ausgehandelt". In der Voreinstellung ist automatisch auch die Option "Autonegotiation aktivieren" aktiviert.
- TP/ITP mit x Mbit/s Vollduplex (Halbduplex)
  Einstellung der Übertragungsgeschwindigkeit und des Vollduplex/Halbduplex-Betriebs. Die Wirkung ist abhängig von der eingestellten Option "Autonegotiation aktivieren":
  - Autonegotiation aktiviert
    Sie können sowohl Crosskabel als auch Patchkabel verwenden. Bei dieser Einstellung wird der Port überwacht.
  - Autonegotiation deaktiviert
    Achten Sie darauf, dass Sie die korrekten Kabel verwenden (Patch- oder Crosskabel)! Auch bei dieser Einstellung wird der Port überwacht.
- Deaktiviert
  Abhängig vom Gerätetyp kann in der Klappliste die Option "deaktiviert" angeboten werden. Sie haben damit beispielsweise die Möglichkeit, den Zugang über einen unbenutzten Port aus Sicherheitsgründen zu unterbinden. Bei dieser Einstellung werden keine Diagnoseereignisse erzeugt.

Option "Überwachen"

Mit der Option "Überwachen" aktivieren/deaktivieren Sie die Port-Diagnose. Sie können die Überwachung nur dann aktivieren, wenn das Gerät diese Funktion unterstüzt.

Im Folgenden ist aufgeführt, was bei aktivierter Überwachung im Betrieb vom Gerät überwacht wird - unabhängig davon, ob Sie den Port verschaltet haben oder nicht:

- Der Link-Status wird überwacht, d. h. eine Diagnose wird bei Link-Down erzeugt (z. B. bei Leitungsbruch oder gezogenem Stecker). Bei Fiber Optic Ports (FO) wird die Systemreserve überwacht, so dass Sie eine gestufte Maintenance-Diagnose erhalten.
- Die Medien-Einstellungen werden überwacht, d. h. eine Diagnose wird dann erzeugt, wenn im Betrieb die lokale Port-Einstellung (Übertragungsrate/Duplex) nicht mit der Einstellung des Partner-Ports übereinstimmt.
- Wenn der Port verschaltet ist: Die Leitungslänge/Signallaufzeit wird überwacht, d. h. eine Diagnose wird dann erzeugt, wenn im Betrieb die lokal ermittelte Leitungslänge/Signallaufzeit nicht mit der vom Partner ermittelten Leitungslänge/Signallaufzeit übereinstimmt.

Überwachen deaktivieren

Eine Überwachung ist in folgenden Fällen **nicht** sinnvoll:

- Am lokalen Port ist die Option "Alternative Partner" aktiviert, z. B. für einen Docking Port einer Docking Station (Werkzeugwechsler). In diesem Fall ist eine Link-Status-Änderung ein normale Betriebsfall.
- Der Partner-Port gehört zu einem Nicht-PROFINET-Gerät. Nur Ports von PROFINET-Geräten übermitteln ihre Port-Einstellungen wie Übertragungsrate, Duplex-Einstellung etc.
- Der lokale Port ist mit einem Medienkonverter verbunden (passive Netzkomponente mit unterschiedlichen Signallaufzeiten für Ethernet-Frames).
- Am lokalen Port ist in den Boundaries-Einstellungen die Option "Ende der Topologieerkennung" aktiviert.

Überwachung der Solltopologie

MIt der Option "Überwachen" ist keine Überwachung auf Einhaltung der Solltopologie verknüpft! D. h. ob die projektierten Schnittstellen- und Port-Eigenschaften wie z. B. Gerätename, Portname und Leitungslänge mit den aktuellen Einstellungen übereinstimmen, wird nicht durch die Option "Überwachung" beeinflusst.

Die Einhaltung der Solltopologie wird automatisch geprüft, wenn Sie den lokalen Port mit einem Partnerport verschalten; der Partner-Port darf dann icht deaktiviert sein. Ob die Überwachung der Solltopologie durchgeführt wird oder nicht möglich ist, wird Ihnen in den Port-Eigenschaften angezeigt (Portverschaltung, Partnerport, siehe im Bild unten ①).

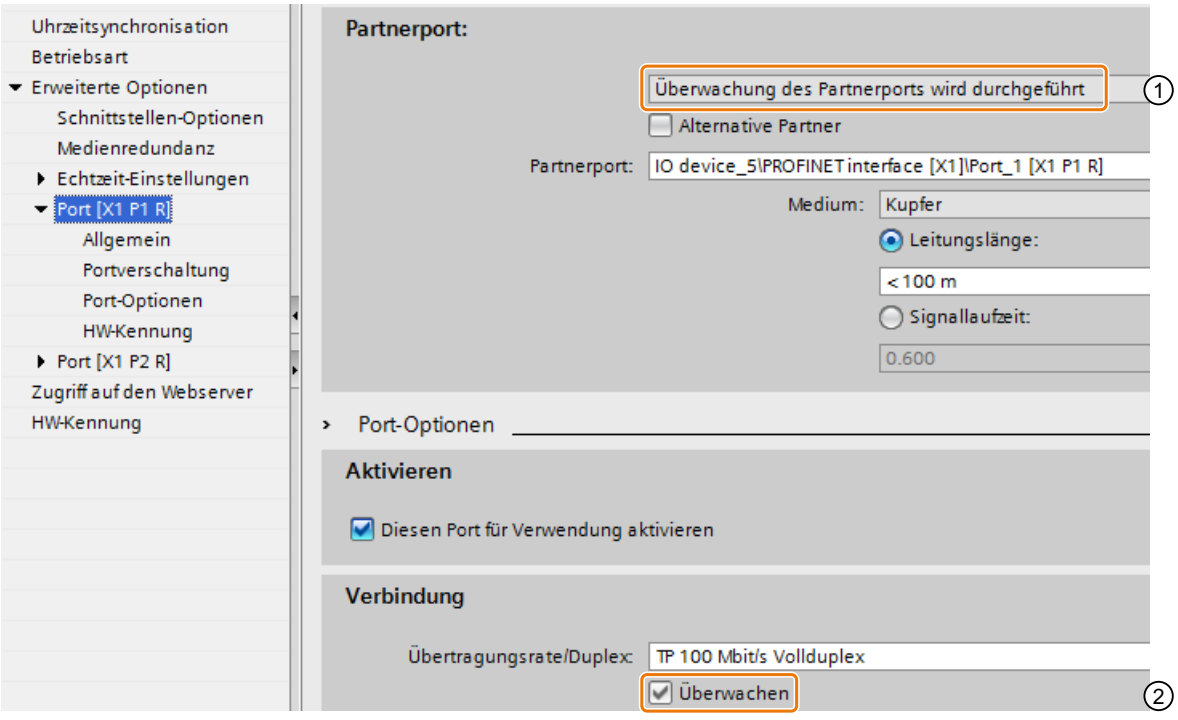

① Anzeige, ob das Gerät die tatsächlichen Eigenschaften des Partnerports gegen die projektierten Eigenschaften des Partnerports prüfen kann. Diese Einhaltung der Solltopologie wird automatisch geprüft, wenn Sie den lokalen Port mit einem Partnerport verschalten.

② Einstellung der Überwachung auf Link-Down, Medien-Einstellungen und Leitungslänge/Signallaufzeit. Der lokale Port vergleicht zur Laufzeit die eigenen Einstellungen mit den Einstellungen des Partnerports.

Option "Autonegotiation aktivieren"

Die Einstellung für Autonegotiation ist nur änderbar, wenn ein konkretes Medium (z. B. TP 100 Mbit/s Vollduplex) ausgewählt ist. Ob ein konkretes Medium eingestellt werden kann, hängt von den Eigenschaften des Moduls ab.

Ist die Autonegotiation deaktiviert, dann wird die feste Einstellung des Ports erzwungen, ähnlich wie das z. B. bei einem priorisierten Hochlaufs des IO-Devices erforderlich ist.

Sie müssen für gleiche Einstellungen beim Partnerport sorgen, da bei dieser Option die Betriebsparameter des angeschlossenen Netzes nicht erkannt werden und entsprechend die Datenübertragungsgeschwindigkeit und der Übertragungsmodus nicht optimal eingestellt werden können.

Hinweis

STEP 7 übernimmt bei einem verschalteten lokalen Port die Einstellung für den Partnerport, falls der Partnerport die Einstellung unterstützt. Falls der Partnerport die Einstellung nicht unterstützt, wird eine Fehlermeldung erzeugt.

Port-Optionen für Nutzung der GBIT PROFINET-Schnittstelle

Die PROFINET-Schnittstelle (X3) der CPU 1518-4 PN/DP unterstützt eine maximale Übertragungsrate von 1000 Mbit/s (GBIT).

Um diese Übertragungsgeschwindigkeit zu erreichen, sind folgende Voraussetzungen notwendig:

- CPU-Firmware-Stand ab V1.7.
- Die Teilnehmer am selben PROFINET-Subnetz müssen ebenfalls die Übertragungsrate 1000 Mbit/s unterstützen.
- Die Infrastruktur des Netzes (Netzwerkkabel und Dosen) muss die Kategorie CAT 5e oder besser aufweisen.
- Die Port-Optionen der PROFINET-Schnittstelle X3 müssen wie folgt eingestellt sein:
  - „Übertragungsrate/Duplex“: Automatisch
  - „Autonegotiation“: Aktiviert

Siehe auch

Verkabelungsregeln bei deaktivierter Autonegotiation (Seite 3024)

Begrenzungen am Port (Seite 3025)

Verkabelungsregeln bei deaktivierter Autonegotiation

Voraussetzung

Sie haben z. B. zur Beschleunigung der Hochlaufzeit des IO-Devices für den betreffenden Port folgende Einstellungen vorgenommen:

- Feste Übertragungsgeschwindigkeit
- Autonegotiation inkl. Autocrossing deaktiviert

Die Zeit für das Aushandeln der Übertragungsgeschwindigkeit beim Hochlauf wird somit eingespart.

Wenn Sie Autonegotiation deaktiviert haben, müssen Sie die Verkabelungsregeln beachten.

Verkabelungsregeln bei deaktivierter Autonegotiation

PROFINET-Geräte besitzen folgende zwei Arten von Ports:

Art des Ports	PROFINET-Geräte	Bemerkung
Switchport mit gekreuzter Pinbelegung	Bei IO-Devices: Port 2 Bei S7-CPUs mit 2 Ports: Port 1 und Port 2	Gekreuzte Pinbelegung bedeutet, dass die Pinbelegung der Ports für Senden und Empfangen zwischen den betreffenden PROFINET-Geräten intern vertauscht werden.
Endgeräteport mit nicht gekreuzter Pinbelegung	Bei IO-Devices: Port 1 Bei S7-CPUs mit einem Port: Port 1	-

Gültigkeit der Verkabelungsregeln

Die im folgenden Absatz beschriebenen Verkabelungsregeln gelten ausschließlich für den Fall, dass Sie eine feste Port-Einstellung vorgenommen haben.

Regeln für die Verkabelung

Mehrere IO-Devices können Sie mit einem Patchkabel (Eins-zu-eins-Verdrahtung beider Stecker) in Linie verbinden. Dazu verbinden Sie Port 2 (P2) des IO-Devices mit dem Port 1 (P1) des nächsten IO-Devices. In der folgenden Grafik ist ein Beispiel mit zwei IO-Devices dargestellt.

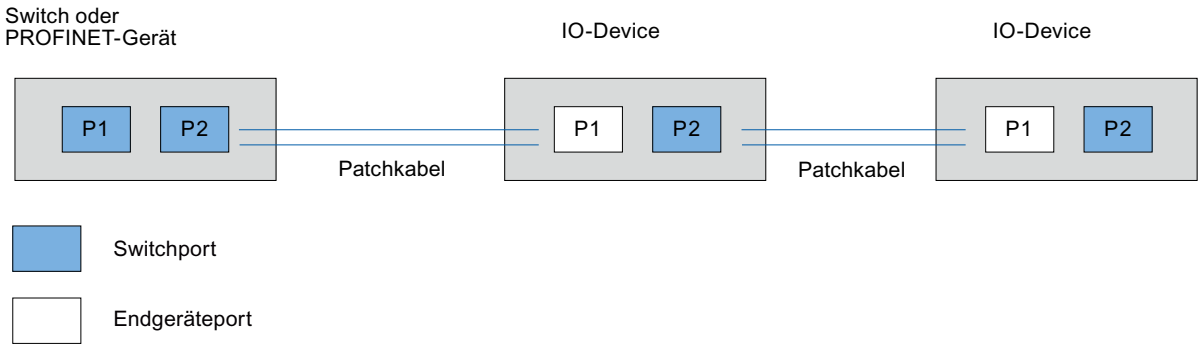

Begrenzungen am Port

Voraussetzung

Um mit Boundaries ("Begrenzungen") arbeiten zu können, muss das jeweilige Gerät Boundaries-Einstellungen unterstützen. Wenn das Gerät für PROFINET keine Boundaries-Einstellungen unterstützt, sind die entsprechenden Parameter deaktiviert, wie beispielsweise bei der CPU 1215C V3.

Boundaries aktivieren

Unter "Boundaries" versteht man Grenzen für die Übertragung bestimmter Ethernet-Frames. Es können folgende Begrenzungen an einem Port gesetzt werden:

- "Ende der Erfassung erreichbarer Teilnehmer"
  DCP-Frames zur Erfassung erreichbarer Teilnehmer werden nicht weitergeleitet. Hinter diesem Port liegende Teilnehmer werden nicht mehr unter "erreichbare Teilnehmer" in der Projektnavigation angezeigt. Hinter diesem Port liegende Teilnehmer können von der CPU nicht mehr erreicht werden.
  Für diese Funktion muss das Gerät mehr als einen Port besitzen.
- "Ende der Topologieerkennung"
  LLDP-Frames (Link Layer Discovery Protocol) zur Topologieerfassung werden nicht weitergeleitet.
- "Ende der Sync-Domain"
  Sync-Frames, die zur Synchronisierung von Teilnehmern innerhalb einer Sync-Domain übertragen werden, werden nicht weitergeleitet.
  Wenn Sie z. B. ein PROFINET-Gerät mit mehr als zwei Ports in einem Ring betreiben, dann sollten Sie das Einspeisen von Sync-Frames in den Ring durch Setzen einer Sync-Boundary (an den Ports, die nicht im Ring sind) verhindern.
  Für diese Funktion muss das Gerät mehr als einen Port besitzen.
  Weiteres Beispiel: Möchten Sie mehrere Sync-Domains nutzen, dann projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.

Einschränkungen

Folgende Einschränkungen sind zu beachten:

- Nur wenn der Port das jeweilige Feature unterstützt sind die einzelnen Optionskästchen bedienbar.
- Wenn für den Port ein Partnerport bestimmt wurde sind folgende Optionskästchen nicht bedienbar:
  - "Ende der Erfassung erreichbarer Teilnehmer"
  - "Ende der Topologieerkennung"

Gerätetausch ohne Wechselmedium aktivieren

Austausch eines IO-Devices ohne Wechselmedium

Häufig ist es notwendig, IO-Devices in Automatisierungssystemen auszutauschen. Üblicherweise wird den IO-Devices entweder über das Stecken eines Wechselmediums oder über das PG ein Gerätename zugewiesen. Der IO-Controller identifiziert das IO-Device über diesen Gerätenamen.

IO-Devices können unter bestimmten Voraussetzungen auch ohne das Stecken eines Wechselmediums (z. B. Memory Card) bzw. ohne PG ihren Gerätenamen erhalten. Hierzu analysiert der IO-Controller die Nachbarschaftsbeziehungen zwischen den einzelnen IO-Devices und dem IO-Controller über Ethernet-Mechanismen (LLDP-Protokoll; Link Layer

Discovery Protocol). Aus diesen Nachbarschaftsbeziehungen erkennt der IO-Controller, welches IO-Device ersetzt wurde, und weist ihm den projektierten Gerätenamen zu.

Voraussetzungen

- Es ist bereits eine Portverschaltung projektiert.
- Die beteiligten IO-Devices des Automatisierungssystems müssen den Gerätetausch ohne Wechselmedium unterstützen (LLDP-Protokoll).

Hinweis

Verwenden Sie als Austauschgeräte nur neue IO-Devices oder versetzen Sie bereits parametrierte IO-Devices vor der Inbetriebnahme in den Auslieferungszustand.

Bei S7-1500-CPUs ab Firmware-Version V1.5 ist ein Zurücksetzen parametrierter IO-Devices in den Auslieferungszustand nicht erforderlich. Bedingung hierfür ist, dass die Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" beim IO-Controller aktiviert ist (Bereich "Erweiterte Optionen > Schnittstellen-Optionen" der Eigenschaften der PROFINET-Schnittstelle).

Vorgehen

Um den Austausch eines IO-Devices ohne Wechselmedium zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die PROFINET-Schnittstelle des entsprechenden IO-Controllers.
2. Aktivieren Sie in den Eigenschaften der Schnittstelle unter "Erweiterte Einstellungen > Schnittstellen-Optionen" das Optionskästchen "Gerätetausch ohne Wechselmedium ermöglichen".

Die Option "Gerätetausch ohne Wechselmedium ermöglichen" erlaubt auch eine automatische Inbetriebnahme, d. h. Sie können das IO-System mit den IO-Devices in Betrieb nehmen, ohne ihnen vorab ihren Gerätenamen zuzuweisen.

IO-Devices ohne entsprechende Funktionalität

Wenn einzelne IO-Devices des Automatisierungssystems den Gerätetausch ohne Wechselmedium nicht unterstützen, wird für das IO-Device eine entsprechende Meldung ausgegeben. Es ist möglich, die Funktionalität trotzdem zu nutzen, dazu müssen aber alle folgenden Bedingungen erfüllt sein:

- Im übergeordneten IO-Controller ist die Funktion "Gerätetausch ohne Wechselmedium" aktiviert.
- Das IO-Device unterstützt das LLDP-Protokoll.
- Mindestens ein "Nachbar" des IO-Devices unterstützt
  - als IO-Controller die Funktion "Gerätetausch ohne Wechselmedium" oder
  - als IO-Device ebenfalls das LLDP-Protokoll.

Wenn Sie die Umgebung des IO-Devices so konfigurieren, dass die oben beschriebenen Bedingungen für den "Gerätetausch ohne Wechselmedium" erfüllt sind, ist die Nutzung der Funktion möglich, auch wenn das IO-Device die Funktionalität nicht direkt unterstützt.

Siehe auch

Gerätename und IP-Adresse zuweisen (Seite 2990)

Komponenenten mit der Funktion Gerätetausch ohne Wechselmedium (http://support.automation.siemens.com/WW/view/de/36752540)

DP-Slave über IE/PB-Link an ein PROFINET IO-System anbinden

Voraussetzungen

- STEP 7 ab V12
- S7-1500 CPU ab Firmware-Stand 1.7
- ET 200SP CPU ab Firmware-Stand 1.7
- S7-1500 Software Controller
- S7-300/400 CPU

Vorgehen zum Anbinden eines DP-Slaves über IE/PB-Link

Um in STEP 7 ein DP-Slave über IE/PB Link an ein PROFINET IO-System anzubinden, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus dem Hardware-Katalog eine PROFINET-CPU, z. B. 1513-1 PN, per Drag & Drop in die Netzsicht von STEP 7.
2. Ziehen Sie aus dem Hardware-Katalog ein IE/PB Link PN IO per Drag & Drop in die Netzsicht von STEP 7. Das IE/PB Link PN IO finden Sie unter Netzkomponenten > Netzübergänge > IE/PB Link PN IO.
3. Ordnen Sie das IE/PB Link PN IO der CPU zu.
4. Ziehen Sie aus dem Hardware-Katalog ein PROFIBUS-Interfacemodul, z. B. IM155-6 DP HF, per Drag & Drop in die Netzsicht.
5. Orden Sie das Interfacemodul dem IE/PB-Link zu.

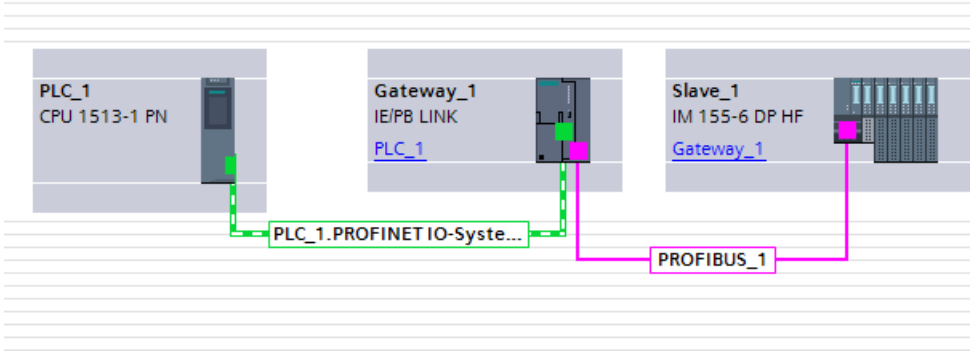

6. Selektieren Sie das IE/PB Link PN IO in der Netzsicht von STEP 7.
7. Wählen Sie im Inspektorfenster im Bereich "Netzübergang" die Option "Netzübergang als PROFINET IO-Proxy" aus.

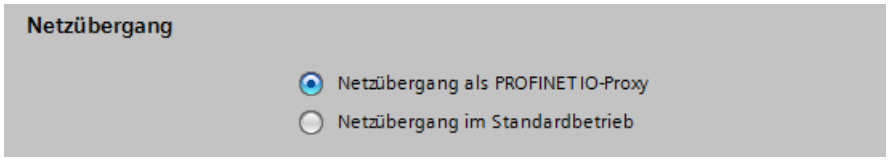

8. Im Bereich PROFINET-Gerätenummer können Sie für den DP-Slave eine PROFINET-Gerätenummer vergeben.
   Wenn Sie das Optionskästchen "Gerätenummer = PB-Adresse" aktiviert haben (Voreinstellung), dann vergibt STEP 7 automatisch die Gerätenummer entsprechend der PROFIBUS-Adresse des Slaves. Außerdem entfällt für Sie das Nachführen der Gerätenummer, wenn sich die PROFIBUS-Adresse ändert.

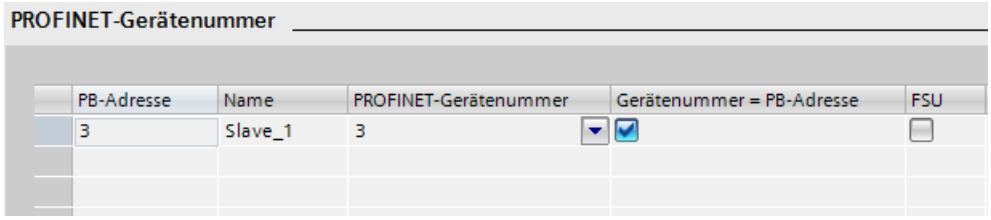

Siehe auch

Konfiguration mit IE/PB-Link PN IO (Seite 2986)

Aufbauempfehlungen zum Optimieren von PROFINET

PROFINET mit RT optimieren

PROFINET ermöglicht Ihnen Kommunikation mit hoher Performance über alle Ebenen.

Im folgenden Bild sehen Sie ein Beispiel für eine optimierte PROFINET-Topologie.

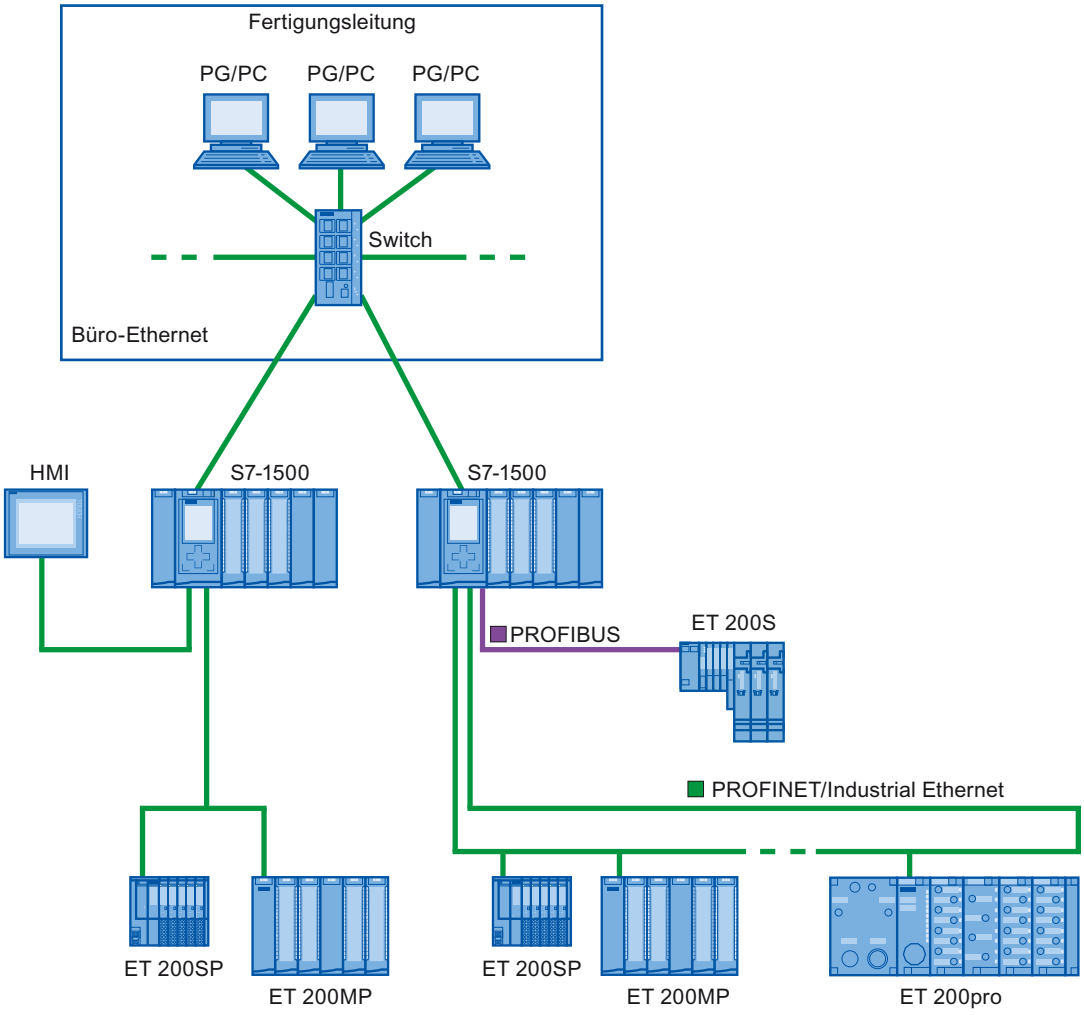

Beachten Sie beim Aufbau Ihrer PN-Netzwerktopologie, dass die verschiedenen Automatisierungsanwendungen in eigene Netzwerkzweige verteilt werden, damit auch für zukünftige Erweiterungen eine ausreichende Bandbreitenreserve zur Verfügung steht.

- Wenn Sie Standard-Ethernet-Geräte in die Netztopologie einbinden bzw. Standard-Ethernet-Kommunikation benutzen, berücksichtigen Sie die durch Standard-Ethernet bedingte Netzlast und passen Sie die Netztopologie gegebenenfalls an (max. Bandbreite 100 MBit/s).
- Nutzen Sie für die Kommunikation zu überlagerten Netzwerken mit einem hohen Datenaufkommen möglichst direkte Wege zur überlagerten Netzwerkinfrastruktur.

Beachten Sie dazu auch die Installationsrichtlinie der PROFIBUS-Nutzerorganisation.

PROFINET mit IRT aufbauen

Beachten Sie beim Aufbau und Betrieb eines PROFINET IO-Systems im IRT-Betrieb die folgenden Regeln. Sie dienen dazu, einen optimalen Betrieb Ihres PROFINET IO-Systems sicherzustellen.

- Bei der Verwendung von IRT müssen Sie die Topologie konfigurieren. Aktualisierungszeit, Bandbreite und weitere Parameter werden dadurch exakt berechnet.
- Möchten Sie mehrere Sync-Domains nutzen, projektieren Sie eine Sync-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.
- In einer Sync-Domain können Sie nur jeweils einen Sync-Master projektieren.
- Ein PROFINET IO-System darf nur einer einzigen Sync-Domain angehören.
- Wenn Sie PROFINET-Geräte in einer Sync-Domain projektieren und mit IRT synchronisieren wollen, müssen die betreffenden PROFINET-Geräte die IRT-Kommunikation unterstützen.
- Verwenden Sie nach Möglichkeit dasselbe PROFINET-Gerät als PROFINET IO-Controller und Sync-Master.
- Wird nur ein Teil der PROFINET-Geräte eines PROFINET IO-Systems synchronisiert, so beachten Sie Folgendes: Weisen Sie PROFINET-Geräten, die nicht an der IRT-Kommunikation teilnehmen, in der Sync-Domain die RT-Klasse "RT" und die Synchronisationsrolle "unsynchronisiert" zu.

Anwendungen für CPU mit mehreren PROFINET IO-Schnittstellen

- Kopplung von Maschinen: Sie haben in Ihrer Konfiguration Maschinen, die sich auf getrennten IO-Strängen befinden. Über die PROFINET IO-Schnittstellen X2 können Sie eine Echtzeitkommunikation zwischen den CPU betreiben. Nutzen Sie dazu die Funktionalitäten I-Device oder Shared-I-Device.
  Das folgende Bild zeigt eine Beispielkonfiguration, bei der 2 Maschinen über die Schnittstelle X2 mit einer I-Device-Beziehung gekoppelt sind.

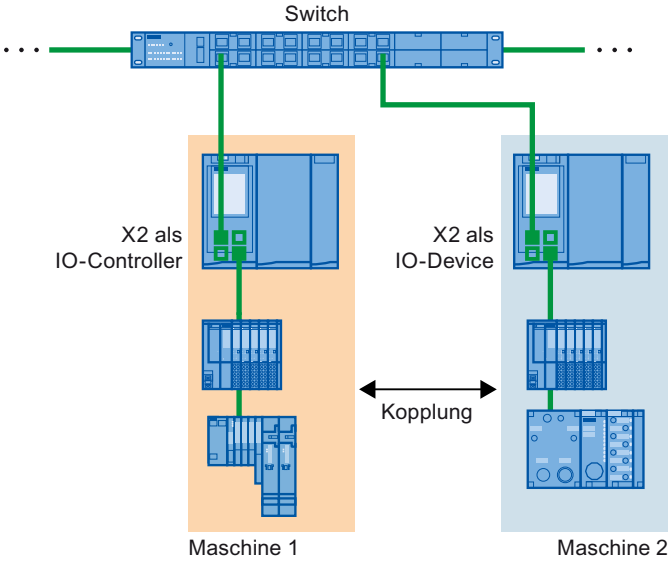

- Aufteilung nach Automatisierungsaufgaben:
  - Für Automatisierungsaufgaben mit hohen Anforderungen an Performance und Deterministik nutzen Sie PROFINET mit IRT über die Schnittstelle X1.
  - Für weitere Aufgaben, die Sie mit RT realisieren können, nutzen Sie die Schnittstelle X2.

Wenn Sie die Schnittstelle X2 bei einer der folgenden CPUs als PROFINET IO-Schnittstelle verwenden, dann kann das Einfluss auf die Performance haben:

- CPU 1515(F)-2 PN
- CPU 1515T-2 PN
- CPU 1516(F)-3 PN/DP
- CPU 1516(F)pro-2 PN

Weitere Informationen dazu finden Sie im Funktionshandbuch Zyklus- und Reaktionszeiten.

Topologische Überlagerung von IO-Systemen bei Multi-Controller-Anwendungen

Bei einer Konfiguration mit mehreren IO-Controllern addieren sich auf gemeinsam genutzten Pfaden die Netzlasten aller beteiligten PROFINET IO-Systeme auf.

Um hohe Kommunikationslasten bei Multi-Controller-Anwendungen zu vermeiden, halten Sie sich an folgende Empfehlungen:

- Vermeiden Sie Pfade, die gemeinsam von mehreren IO-Systemen genutzt werden. Das folgende Bild zeigt eine Konfiguration mit zwei PROFINET IO-Systemen, die gemeinsame Pfade nutzen.

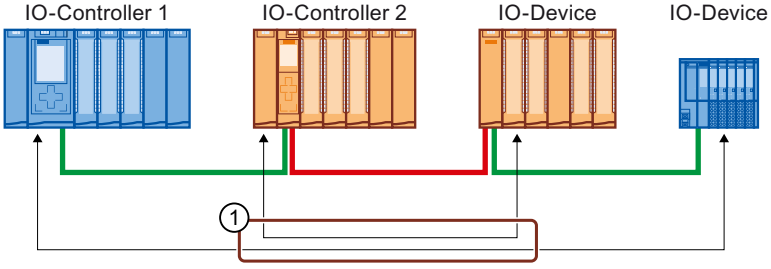

① Gemeinsam genutzter Pfad

Im folgenden Bild nutzen die beiden PROFINET IO-Systeme keine gemeinsamen Pfade.

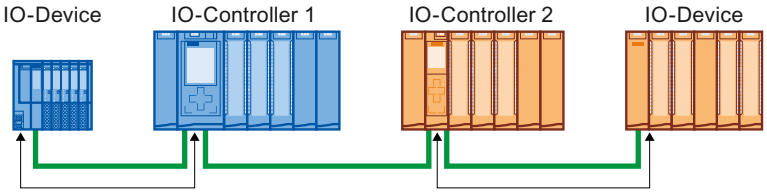

- Wenn keine Trennung möglich ist: Erhöhen Sie die Aktualisierungszeit bei den betroffenen IO-Devices.

Aufbaurichtlinien der PROFIBUS-Nutzerorganisation

Die Installationsrichtlinie finden Sie im Internet.

Begrenzung der Dateneinspeisung ins Netz

Dateneinspeisung ins Netz an PROFINET-Schnittstellen begrenzen

Die Funktion "Dateneinspeisung ins Netz begrenzen" begrenzt die Netzlast von Standard-Ethernet-Kommunikation, welche von der Schnittstelle in das Netzwerk eingespeist wird, auf einen maximalen Wert. Zyklische Echtzeitkommunikation (RT/IRT) ist davon nicht betroffen.

In einem PROFINET IO-System können kritische Netzlasten durch Standard-Ethernet-Kommunikation entstehen. Alle Geräte in einem PROFINET IO-System sollten die "Begrenzung der Dateneinspeisung ins Netz" unterstützen.

Wenn Sie Geräte verwenden, die viel "Standard-Ethernet-Kommunikation" in das PROFINET IO-System, einspeisen, dann sollten Sie die Topologie gegebenenfalls anpassen.

Abhängig von der Schnittstelle können Sie die Funktion "Dateneinspeisung ins Netz begrenzen" aktivieren bzw. deaktivieren. Wenn Sie die Schnittstelle X1 einer S7-1500 CPU als IO-Controller oder I-Device verwenden, dann ist die Funktion "Dateneinspeisung ins Netz begrenzen" immer aktiv. Beispiel: Wenn Sie die Schnittstelle X1 einer S7-1500 CPU nicht als IO-Controller oder I-Device nutzen, dann können Sie die Funktion aktivieren bzw. deaktivieren.

Nutzen von der Begrenzung der Dateneinspeisung

- Aufteilen der Bandbreite für Standard-Ethernet-Kommunikation zwischen den Geräten:
  In PROFINET-Netzen teilen sich zyklische Echtzeitkommunikation und Standard-Ethernet-Kommunikation dasselbe Netz. Das führt dazu, dass für Standard-Ethernet-Kommunikation nur eine begrenzte Bandbreite übrig bleibt. Die Begrenzung der Dateneinspeisung stellt sicher, dass die verbleibende Bandbreite für Standard-Ethernet-Kommunikation nicht von einem einzelnen Gerät besetzt wird, sondern zwischen mehreren Geräten aufgeteilt werden kann.
- Spitzen in der Dateneinspeisung glätten:
  Die Begrenzung der Dateneinspeisung glättet Spitzenlasten von Standard-Ethernet-Kommunikation (z. B. von Open User Communication, Zugriffe vom Webserver).
- Probleme in der Dateneinspeisung eingrenzen:
  Wenn Anwendungen in einem Gerät zu viele Daten erzeugen, dann werden diese Daten nicht ins PROFINET-Netz weitergeleitet. Negative Auswirkungen (z. B. Datenverlust, Kommunikationsabbruch) bleiben auf das Gerät und dessen Kommunikationspartner beschränkt. Weitere Teilnehmer werden nicht gestört.

Begrenzung der Dateneinspeisung ins Netz für eine CPU einstellen

Um die Begrenzung der Dateneinspeisung ins Netz ("Einspeiselastbegrenzung") einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 die Schnittstelle der CPU.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Erweiterte Optionen" > "Schnittstellen-Optionen".
3. Aktivieren bzw. Deaktivieren Sie das Optionskästchen "Dateneinspeisung ins Netz begrenzen".

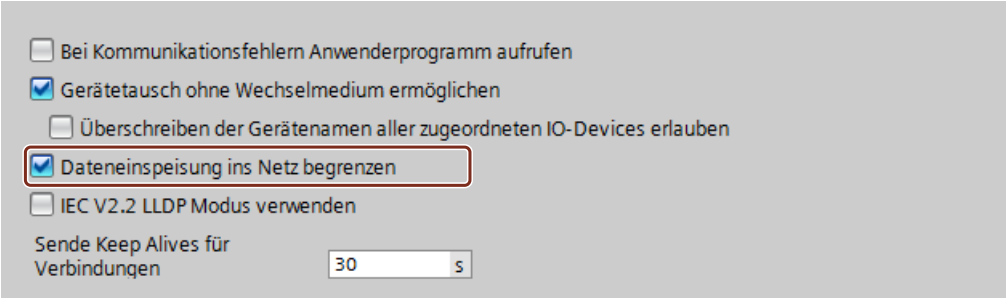

1.4.11.3 GSD-Dateien verwenden

GSD-Dateien für IO-Devices

Grundsätzliches zu GSD-Dateien von IO-Devices

Die Eigenschaften von PROFINET IO-Devices sind nicht in einer Schlüsselwort-basierten Textdatei hinterlegt (wie bei PROFIBUS DP-Slaves), sondern in einer XML-Datei, deren Struktur und Regeln von einem GSDML-Schema bestimmt wird.

Die Sprache zur Beschreibung der GSD Dateien ist GSDML (Generic Station Description Markup Language). Sie wird durch das GSDML-Schema definiert. Ein GSDML-Schema enthält Gültigkeitsregeln, die es erlauben, z. B. die Syntax einer GSD-Datei zu überprüfen. GSDML-Schemata (in Form von Schema-Dateien) beziehen Hersteller von IO-Devices von PROFIBUS International.

Funktionserweiterungen im Bereich von PROFINET IO haben Auswirkungen auf die GSDML-Spezifikation und das zugehörige Schema. Durch eine Funktionserweiterung entsteht eine neue Version der Spezifikation und des Schemas.

Namen der GSD-Dateien für IO-Devices

Der Name einer GSD-Datei für IO-Devices kann beispielsweise folgendermaßen aussehen:

"GSDML-V1.0-Siemens-ET200S-20030616.xml"

Namensbestandteil	Erläuterung
GSDML	Zeichenfolge, mit der jede GSD-Datei für IO-Devices beginnt
V1.0	Version des GSDML-Schemas
Siemens	Hersteller
ET200S	Name des Geräts
20030616	Kennzeichnung des Ausgabestands (Datum)
.xml	Dateiendung

Versionierung der GSD-Dateien für IO-Devices

Die Versionsinformation für GSD-Dateien für PROFINET IO-Devices besteht aus zwei Teilen:

- Version des GSDML-Schemas: Dadurch ist festgelegt, welchen Sprachumfang eine GSD-Datei nutzt.
- Ausgabestand in Form eines Datums: Der Ausgabestand von GSD-Dateien erhöht sich, wenn z. B eine Fehlerbeseitigung durchgeführt wurde oder eine Funktionserweiterung eingebracht wurde.

Funktionserweiterungen können zu einer neuen Version des GSDML-Schemas führen. Eine neue Version eines GSDML-Schemas kann möglicherweise nur eingeschränkt unterstützt werden.

GSD-Datei installieren

Einführung

In einer GSD-Datei (Gerätebeschreibungsdatei) sind alle Eigenschaften eines IO-Devices hinterlegt. Falls Sie ein IO-Device projektieren möchten, das nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller mitgelieferte GSD-Datei installieren. Über GSD-Dateien installierte IO-Devices werden im Hardware-Katalog angezeigt und können dann selektiert und konfiguriert werden.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Sie haben Zugriff auf die benötigten GSD-Dateien in einem Verzeichnis auf der Festplatte.

Vorgehen

Um eine GSD-Datei zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Dateien liegen.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien eine oder mehrere Dateien aus.
4. Klicken Sie auf die Schaltfläche "Installieren". Die ausgewählten GSD-Dateien werden nun installiert.
5. Um eine Protokolldatei für die Installation zu erstellen, klicken Sie auf die Schaltfläche "Protokolldatei speichern".
   Eventuell auftretende Probleme bei der Installation lassen sich anhand der Protokolldatei nachvollziehen.
6. Klicken Sie auf "Schließen". Sie werden darüber informiert, dass die IO-Devices aus den installierten GSD-Dateien in den Hardware-Katalog eingetragen werden. Dieser Vorgang kann ein paar Sekunden dauern.

Sie haben nun mit der Installation einer GSD-Datei das darin beschriebene IO-Device dem Hardware-Katalog hinzugefügt.

Ergebnis

Sie finden die neuen über GSD-Dateien installierten IO-Devices unter "Weitere Feldgeräte > PROFINET IO" im Hardware-Katalog.

GSD-Dateien werden immer mit dem Projekt zusammen gespeichert, d. h. alle zur Darstellung des Gerätes relevanten Informationen (auch die Symbole) sind auch im gespeicherten Projekt verfügbar.

Siehe auch

Übersicht über den Hardware- und Netzwerkeditor (Seite 9)

GSD-Datei löschen

Einführung

Sie können über GSD-Dateien installierte IO-Devices löschen. Diese werden danach im Hardware-Katalog nicht mehr angezeigt.

Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Es befinden sich über GSD-Dateien installierte IO-Devices im Hardware-Katalog unter "Weitere Feldgeräte > PROFINET".

Vorgehen

Um eine GSD-Datei zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Wählen Sie im Register "Installierte GSDs" das Verzeichnis aus, in dem die GSD-Datei liegt.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien die zu löschende Datei aus.
4. Klicken Sie auf die Schaltfläche "Löschen".

Die ausgewählte GSD-Datei wurde gelöscht und das IO-Device befindet sich nicht mehr im Hardware-Katalog.

Ungenutzte GSD-Dateien finden

Einführung

Wenn Sie GSD-basierte Hardware-Komponente ihrer Hardware-Konfiguration hinzufügen, dann werden unterschiedliche GSD-Dateien zu den Projektdaten hinzugefügt. Wird die GSD-basierte Hardware-Komponente aus der Konfiguration wieder gelöscht, verbleiben die GSD-Dateien im Datenbestand des Projektes. Diese GSD-Dateien werden im Projekt nicht mehr benötigt, sofern die entsprechende GSD-basierte Hardware-Komponente nicht erneut in der Hardware-Konfiguration verwendet wird. Im Projekt nicht mehr benötigte Dateien können Sie über einen Suchmechanismus finden und dann löschen.

Voraussetzung

- Es muss ein Projekt geöffnet sein.
- Das Projekt darf nicht online sein.
- Das Projekt darf nicht schreibgeschützt sein.

Vorgehen

Um ungenutzte GSD-Dateien zu finden und zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien (GSD) verwalten".
2. Klicken Sie im Register "GSDs im Projekt" auf die Schaltfläche "Ungenutzte GSDs finden".

   Hinweis

   Ungenutzte GSD-Dateien sind nur verfügbar, wenn mindestens eine GSD-basierte Hardware-Komponente der Konfiguration hinzugefügt und anschließend wieder aus der Konfiguration entfernt wurde. Ungenutzte GSD-Dateien in den Projektbibliotheken werden bei der Suche ebenfalls berücksichtigt. GSD-basierte Hardware-Komponenten, die noch in der Konfiguration verwendet werden, gelten nicht als ungenutzt.

   Installierte GSD-basierte Hardware-Komponenten, die im Hardware-Katalog enthalten sind, aber bisher nicht in der Hardware-Konfiguration des Projekts verwendet wurden, können keine ungenutzten GSD-Daten im Projekt abgelegt haben.

3. Wenn nicht mehr benötigte GSD-Dateien in den Projektdaten gefunden wurden, können Sie diese nun löschen. Klicken Sie dafür auf die Schaltfläche "Löschen".

Die nicht mehr benötigten GSD-Dateien werden aus den Projektdaten gelöscht. Die GSD-basierte Hardware-Komponente ist weiterhin im Hardware-Katalog verfügbar und kann bei Bedarf der Konfiguration wieder hinzugefügt werden. In diesem Fall werden die dann notwendigen GSD-Daten erneut in die Projektdaten kopiert.

Hinweis

Wenn Sie eine GSD-basierte Hardware-Komponente über "Extras > Gerätebeschreibungsdateien (GSD) verwalten > Installierte GSDs" löschen, ist diese aus dem Hardware-Katalog gelöscht. Ungenutzte GSD-Dateien dieser Hardware-Komponente können aber immer noch ein Teil der Projektdaten sein und können über die Funktion zum Finden der ungenutzten GSD-Dateien gelöscht werden.

ACHTUNG

Nicht mehr rückgängig zu machende Aktionen

Wenn Sie als Aktion die ungenutzten GSD-Dateien löschen, führt dies dazu, dass der Aktionsstapel für rückgängig zu machende Aktionen gelöscht wird. Die Aktion des Löschens von ungenutzten GSD-Dateien, sowie alle vorherigen Aktionen können dann nicht mehr rückgängig gemacht werden.

Ausgabestand einer GSD-Datei ändern

Ausgabestand einer GSD-Datei ändern

Sie können den Ausgabestand der GSD-Datei für IO-Device ändern:

- Nur für das aktuelle IO-Device
- Alle passenden IO-Devices des IO-Systems
- Alle passenden IO-Devices des gesamten Projekts

Zunächst werden alle verfügbaren GSD-Dateien für das aktuelle IO-Device angezeigt. Die angezeigten GSD-Dateien unterscheiden sich nur im Ausgabestand. Die aktuelle verwendete GSD-Datei ist markiert.

Voraussetzung

- Die E/A-Daten sind für alle IO-Devices, für die der Ausgabestand geändert werden soll, gleich.
- Die Artikelnummer hat sich nicht geändert.
- Die Anzahl der Submodule ist gleich.
- Die Parametrierdaten haben sich nicht verändert.
- Es darf sich kein Modul oder Submodul auf einem Steckplatz befinden, der nach der neuen GSD-Datei nicht mehr zulässig ist.

Vorgehen

Um den Ausgabestand von einem oder mehreren IO-Devices zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie das IO-Device, für dessen GSD-Datei Sie den Ausgabestand ändern möchten.
2. Klicken Sie in den Eigenschaften des IO-Device unter "Allgemein > Kataloginformation" auf die Schaltfläche "Ausgabestand ändern".
   Der Dialog "Ausgabestand ändern" wird geöffnet.
3. Selektieren Sie in der Tabelle "Verfügbare Ausgabestände" die GSD-Revision, die Sie verwenden möchten.
4. Wählen Sie unter "Markierte Ausgabestände verwenden für", für welche Geräte der Ausgabestand geändert werden soll:
   - Nur für das aktuelle IO-Device
   - Für alle passenden IO-Devices in des IO-Systems
   - Für alle passenden IO-Devices des Projekts
5. Klicken Sie auf die Schaltfläche "Übernehmen".

1.4.11.4 Spezielle PROFINET-Konfigurationen

Besonderheiten bei S7-300/400/1500 und PC-Stationen

Adressen- und Namensvergabe für PROFINET-Geräte

In diesem Kapitel erfahren Sie, welche Adress- und Namenskonventionen für PROFINET-Geräte gelten.

IP-Adressen

Alle PROFINET-Geräte beherrschen das TCP/IP-Protokoll und benötigen daher für den Betrieb am Ethernet eine IP-Adresse.

Die IP-Adressen können Sie in den Baugruppen-Eigenschaften festlegen. Wenn das Netz Teil eines bestehenden Ethernet-Firmennetzes ist, dann erfragen Sie diese Daten von Ihrem Netzwerkadministrator.

Die IP-Adressen der IO-Devices werden automatisch vergeben und normalerweise erst im Anlauf der CPU den IO-Geräten zugewiesen. Die IP-Adressen der IO-Devices haben immer dieselbe Subnetzmaske wie der IO-Controller und werden - ausgehend von der IP-Adresse des IO-Controllers - in aufsteigender Reihenfolge vergeben.

Gerätenamen

Bevor ein IO-Device von einem IO-Controller angesprochen werden kann, muss es einen Gerätenamen haben. Bei PROFINET ist diese Vorgehensweise gewählt worden, weil Namen einfacher zu handhaben sind als komplexe IP-Adressen.

Der Gerätename wird automatisch aus dem projektierten Namen des Geräts abgeleitet (CPU, CP oder IM):

- Der PROFINET-Gerätename setzt sich aus dem Namen des Geräts (z. B. der CPU), dem Namen der Schnittstelle (nur bei mehreren PROFINET-Schnittstellen) und optional dem Namen des IO-Systems zusammen:
  <CPU-Name>.<Name der Schnittstelle>.<IO-Systemname>
  Diesen Namen können Sie nicht direkt ändern. Sie ändern den PROFINET-Gerätenamen indirekt, indem Sie den Namen der betreffenden CPU, des CPs oder der IM in den Allgemein-Eigenschaften des Moduls ändern. Der PROFINET-Gerätename wird z. B. auch in der Liste der erreichbaren Teilnehmer angezeigt. Wenn Sie den PROFINET-Gerätenamen unabhängig vom Namen des Moduls einstellen wollen, müssen Sie die Option "PROFINET-Gerätename automatisch generieren" deaktivieren.
- Aus dem PROFINET-Gerätenamen wird ein "konvertierter Name" erzeugt. Das ist der Gerätename, der dann tatsächlich in das Gerät geladen wird.
  Der PROFINET-Gerätename wird nur dann konvertiert, wenn er nicht den Regeln von IEC 61158-6-10 entspricht. Auch diesen Namen können Sie nicht direkt ändern.

Gerätenummer

Neben dem Gerätenamen wird beim Stecken eines IO-Devices automatisch auch eine Gerätenummer vergeben, die Sie ändern können.

Über diese Gerätenummer kann im Anwenderprogramm ein IO-Device identifiziert werden, z. B. über die Anweisung "LOG_GEO". Der Gerätename ist im Gegensatz zur Gerätenummer im Anwenderprogramm nicht sichtbar.

Teilnehmerzahl in einem PROFINET-Subnetz

In einem PROFINET-Subnetz wird die zulässige maximale Anzahl der Teilnehmer während der Projektierung überwacht.

Wissenswertes zum Sendetakt

Sendetakt - Kleinst mögliches Aktualisierungsintervall

Der Sendetakt gibt den Zeitraum zwischen zwei aufeinanderfolgenden Kommunikationszyklen an. Er ist das kleinstmögliche Sendeintervall für den Datenaustausch. Die berechneten Aktualisierungszeiten sind ein Vielfaches des Sendetaktes.

Voraussetzungen zum Ändern des Sendetakts

Der Sendetakt wird bei unsynchronisierten Geräten, oder wenn das Gerät ein IO-Controller als Sync-Master ist, am Gerät eingestellt. Bei synchronisierten Geräten (Sync-Master, Sync-Slave) kann der Sendetakt in der Sync-Domain eingestellt werden.

Der Sendetakt ist am Gerät nicht änderbar, wenn es sich um keinen IO-Controller als Sync-Master handelt und am IO-System mindestens ein Gerät Sync-Slave oder Sync-Master ist. In diesem Fall wird nur der Sendetakt angezeigt, der in den Eigenschaften der entsprechenden Synchronisations-Domain gewählt wurde. Zum Ändern müssen Sie die Eigenschaften der entsprechenden Sync-Domain aufrufen (Schaltfläche "Domain-Einstellungen").

Der Sendetakt ist am Gerät änderbar, wenn alle Geräte am IO-System unsynchronisiert sind oder wenn das Gerät ein IO-Controller als Sync-Master ist. Die sich ergebenden Aktualisierungszeiten für die einzelnen IO-Devices sehen Sie in der IO-Kommunikationtabelle oder in den Eigenschaften der einzelnen IO-Geräte.

Einfluss des Sendetakts auf die maximal mögliche Bandbreite für zyklische IO-Daten

Die folgende Tabelle zeigt, wie der Sendetakt die maximal zur Verfügung stehende Bandbreite für zyklische IO-Daten beeinflusst:

Sendetakt	Maximal Bandbreite für zyklische IO-Daten
250 µs – 468,75 µs	<< 125 µs
500 µs – 968,75 µs	= Sendetakt / 2
1 – 4 ms	= 500 µs

Der Sendetakt beeinflusst die Aktualisierungszeit bei IO-Devices mit aktivierter Option "Automatisch" und "Einstellbar" (mit aktivierter Option "Aktualisierungszeit bei Änderung des Sendetakts anpassen").

IO-Devices mit aktivierter Option "Einstellbar" können zu Inkonsistenzen in der Projektierung führen. Sie müssen bei diesen IO-Devices gegebenenfalls die Aktualisierungszeit nachträglich anpassen.

Sendetakt einstellen

Voraussetzungen zum Ändern des Sendetakts am PROFINET-Gerät

Das PROFINET-Gerät muss ein IO-Controller als Sync-Master sein oder es darf im IO-System kein IRT (Isochronous Realtime) projektiert sein.

Wenn das PROFINET-Gerät kein IO-Controller als Sync-Master ist, müssen für die Einstellung des Sendetaktes die folgenden Voraussetzungen erfüllt sein:

- Es darf am IO-System kein Gerät als Sync-Slave oder Sync-Master projektiert sein.
- Alle Geräte am IO-System müssen unsynchronisiert sein.

Wenn IRT mit einem IO-Controller als Sync-Master projektiert ist, kann der Sendetakt am Gerät oder in der Sync-Domain eingestellt werden.

Vorgehen

Um den Sendetakt am PROFINET-Gerät einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den PROFINET IO-Controller in der Geräte- oder Netzsicht.
2. Ändern Sie in den Eigenschaften der PROFINET-Schnittstelle den Wert für das kleinst mögliche Aktualisierungsintervall unter "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > IO-Kommunikation > Sendetakt".

Der Sendetakt ist für alle PROFINET-Geräte am IO-System gültig. Wenn Sie die Synchronisationsrolle auf einen anderen Wert als "Unsynchronisiert", dann können Sie den Sendetakt auch in der Sync-Domain, d. h. zentral am PROFINET IO-System einstellen:

1. Selektieren Sie das IO-System über das PROFINET-Subnetz.
2. Ändern Sie in den Eigenschaften der Sync-Domain den Wert für den Sendetakt unter "Domain-Management > Sync-Domains > Sync-Domain_1 > Sendetakt".

Siehe auch

PROFINET IO mit IRT konfigurieren (Seite 3050)

Ungerade Sendetakte einstellen

S7-1500-CPU unterstützen sowohl gerade als auch ungerade Sendetakte.

Gerade Sendetakte: 0,25 ms, 0,5 ms, 1 ms, 2 ms, ...

Ungerade Sendetakte: 0,375 ms, 0,75 ms, 1,125 ms, …

Für IO-Devices, die den schnellen Sendetakt der CPU (d. h. des IO-Controllers) nicht unterstützen oder die keine ungeraden Sendetakte unterstützen, "adaptiert" die CPU den Sendetakt des IO-Devices. Es kann also z. B. sein, dass die CPU (IO-Controller) mit 875 µs

Sendetakt arbeitet, ein angeschlossenes IO-Devices ebenfalls mit 875 µs arbeitet, ein anderes mit 2 ms Sendetakt arbeitet.

Regeln

- Die Sendetakt-Adaption funktioniert nur bei unsynchronisierten IO-Devices (RT-Klasse "RT").
- Für synchronisierte IO-Devices (RT-Klasse "IRT") gilt ein einheitlicher Sendetakt, der in der Sync-Domain eingestellt wird. Die Aktualisierungszeit entspricht in diesem Fall dem Sendetakt.
- Für IO-Devices, deren Aktualisierungszeit-Option auf "Automatisch" eingestellt ist, wird von STEP 7 eine Aktualisierungszeit von 2 ms oder größer eingestellt, d. h. kleinere Aktualisierungszeiten als 2 ms sind bei dieser Option nicht möglich.

Beispiel

Ein PROFINET IO-Mastersystem besteht aus folgenden Komponenten:

- IO-Controller: CPU 1516-3 PN/DP (6ES7 516-3AN00-0AB0)
- IO-Device 1: ET200MP (6ES7 155-5AA00-0AB0) mit einem DI-Modul
- IO-Device 2: ET200S (6ES7 151-3BA23-0AB0 V6.0) mit einem DI-Modul

Das IO-Device 2 unterstützt den ungeraden Sendetakt (hier: 0,875 ms) nicht.

1. Ordnen Sie die IO-Devices dem IO-Controller zu.
2. Für das IO-Device 1 stellen Sie die Synchronisationsrolle "Sync-Slave" ein (Inspektorfenster, Bereich PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation).
3. Für den IO-Controller stellen Sie auf gleiche Weise die Synchronisationsrolle "Sync-Master" ein.
4. Verschalten Sie die Ports von IO-Controller und IO-Device 1 miteinander (z. B. in der Topologiesicht).
5. Das IO-Device 2 bleibt unsynchronisiert; die Aktualisierungszeit-Option ist auf "Automatisch" eingestellt.
6. In der Sync-Domain (Eigenschaften des Subnetzes) stellen Sie den Sendetakt 0,875 ms ein. Alle Teilnehmer der Sync-Domain haben dann diesen Sendetakt bzw. diese Aktualisierungszeit.

Das unsynchronisierte IO-Device 2 erhält automatisch die Aktualisierungszeit 2 ms.

Konfiguration mit externem IO-Controller

Konfiguration mit externem IO-Controller

CPs, die als externe IO-Controller verwendet werden können, beherrschen eine Vielzahl von Kommunikationsmöglichkeiten und sind daher nicht für die Verwendung als IO-Controller "vorbestimmt".

Wenn Sie einen externen IO-Controller (z. B. CP 443-1 Advanced) verwenden, dann müssen Sie nach dem Stecken des IO-Controllers ein IO-System einfügen.

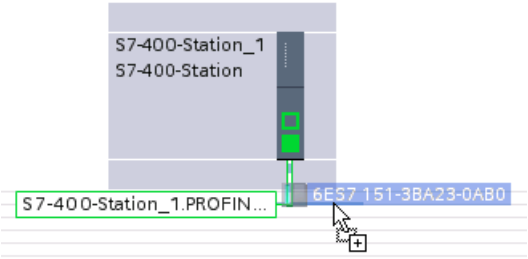

Die gewünschten IO-Devices können dann per Drag & Drop aus dem Hardware-Katalog auf dieses PROFINET IO-System gezogen werden.

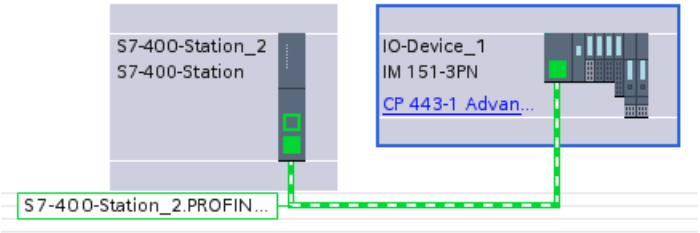

Konfiguration mit SIMATIC PC-Station

Konfiguration mit SIMATIC PC-Stationen

Eine "PC-Station" ist ein PC mit Kommunikationsbaugruppen und Software-Komponenten innerhalb einer Automatisierungslösung mit SIMATIC.

Durch Verwendung entsprechender Kommunikationsbaugruppen und Software-Komponenten können Sie eine PC-Station als PROFINET IO-Controller betreiben.

Ihre PC-Applikationen in der PC-Station haben folgende Zugangsmöglichkeiten zum PROFINET IO-Controller:

- Als OPC-Client über den OPC-Server PROFINET IO
- Direkt über die PROFINET IO-Anwenderschnittstelle (IO-Base-Programmierschnittstelle)

Zu einem Zeitpunkt kann von PC-Applikationen immer nur eine dieser Zugangsmöglichkeiten genutzt werden (Open/Close-Sequenz).

Funktionen	OPC-Server	RTE-Base Programmierschnitt-stelle
IO-Daten lesen und schreiben	Ja	Ja
Datensätze lesen und schreiben	Ja	Ja
Alarme empfangen und quittie-ren	Nein	Ja

Im folgenden Bild sehen Sie eine PC-Station mit den beschriebenen Komponenten.

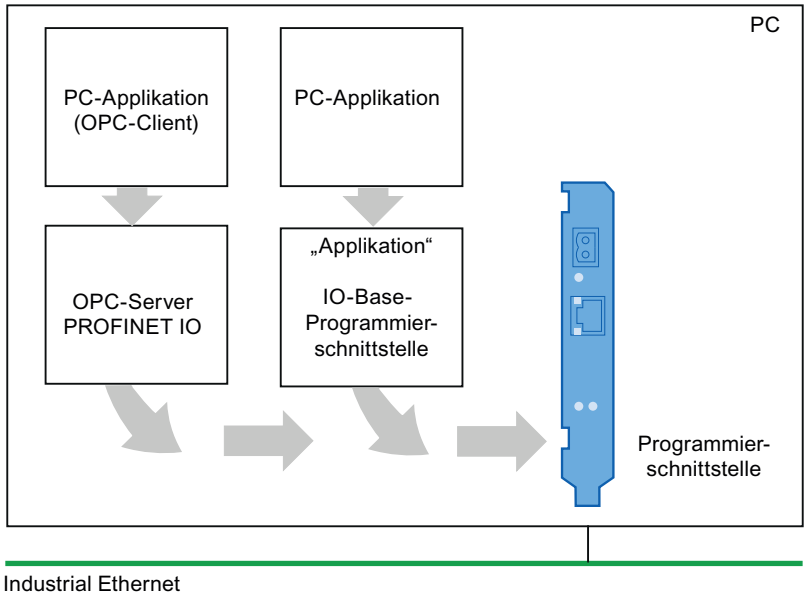

Konfiguration in der Netzsicht

In dem folgenden Bild sehen Sie eine PC-Station mit einem OPC-Server und einer PC-Applikation. Über die hinzugefügte Ethernet-Schnittstelle eines CP 1612 wird die PC-Station zu einem IO-Controller mit angeschlossenen IO-Devices. Außerdem wurde IE/PB-Link als Netzübergang von PROFINET zu PROFIBUS verwendet.

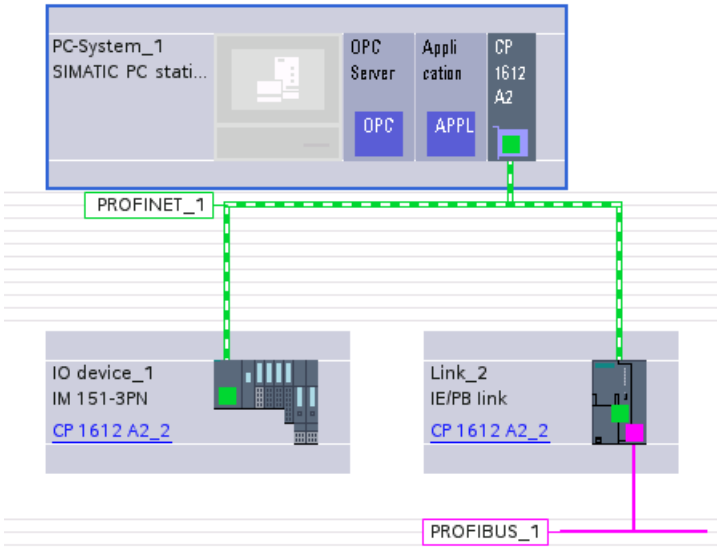

IO-Routing bei CP 1604 / 1616 / 1626 und CP 1616 onboard

CP als PROFINET IO-Router - praktische Bedeutung

Sie können eine PC-Station mit den CPs 1604 / 1616 / 1626 als PROFINET IO-Router betreiben, indem Sie den jeweiligen CP gleichzeitig als PROFINET IO-Device und PROFINET IO-Controller einsetzen. Durch dieses IO-Routing werden Prozessdaten zwischen zwei PROFINET IO-Systemen ausgetauscht. Der CP kann bei entsprechender Konfiguration so dafür sorgen, dass beispielsweise einem Gerät in der Leitebene bestimmte Eingänge oder Ausgänge in der Prozessebene zugänglich gemacht werden.

In einer typischen Konfiguration erfüllt der CP dann gleichzeitig folgende Funktionen:

- PROFINET IO-Device für einen übergeordneten PROFINET IO-Controller der Leitebene
- PROFINET IO-Controller mit angeschlossenen IO-Devices für die unterlagerte Prozessebene

Projektierung - Transferbereich "TM" (IO-Routingbereich) zuordnen

In der Projektierung wählen Sie unter "Betriebsart" die Option "IO-Device".

Damit erscheint der zusätzliche Parametrierbereich "I-Device-Kommunikation", in dem Sie die Einstellungen für das IO-Routing vornehmen können. Dort weisen Sie in der Tabelle "Transferbereich" die Datenbereiche für den Datenaustausch zum übergeordneten IO-Controller zu.

Im vorliegenden Fall haben Sie folgende Optionen in der Spalte "Typ":

- CD: Controller Device
  Wählen Sie diese Option zur Festlegung eines Transferbereiches zwischen IO-Controller und IO-Device. Beachten Sie hierzu die weiteren Informationen in der Online-Hilfe unter "Konfiguration für PROFINET IO".
- TM: IO-Routing (Transfermodul-Mapping)
  Wählen Sie diese Option für den Anwendungsfall IO-Router. Es erscheint im Inspektorfenster unter "Betriebsart / I-Device-Kommunikation " ein zusätzlicher Parametrierbereich mit dem gerade festgelegten Namen. In diesem Parametrierbereich treffen Sie die Zuordnung für das IO-Routing in der IO-Routing-Tabelle.
- F-PS: PROFIsafe
  Wählen Sie diese Option für die Konfiguration eines PROFIsafe-Transferbereiches. Der PROFIsafe-Transferbereich ist bidirektional und belegt in der Tabelle der Transferbereiche zwei Zeilen. Die Länge der PROFIsafe-Transferbereiche ist dabei für Eingang und Ausgang auf jeweils 8 Byte Nutzdaten begrenzt.

IO-Routing-Tabelle

IO-Routingbereiche repräsentieren im überlagerten IO-System die zwischen dem unterlagerten IO-System und dem überlagerten IO-System gerouteten Daten.

Tragen Sie entsprechend die Transferbereiche der IO-Devices aus dem unterlagerten IO-System in der IO-Routing-Tabelle ein.

Die Tabelle enthält folgende Spalten:

Parameter	Bedeutung
TM-Offset	Offset in Bezug auf das erste Bit im IO-Routingbereich. Die Angabe hat die Form <Byte>.<Bit>. Beispiel: 0.5
Gerät/Baugruppe (Steckplatz)	Angabe des PROFINET-Gerätenamens des IO-Devices und dessen Modul und Steckplatz. Der aufgeblendete Dialog zeigt die Transferbereiche der am unterlagerten IO-System projektierten IO-Devices.
IO-Typ	Wählen Sie den IO-Typ über die Klappliste. Nur bei einem IO-Routingbereich vom Typ "Input" ist diese Auswahl möglich, und zwar nur dann wenn als "Gerät/Baugruppe (Steckplatz)" ein Mischmodul ausgewählt ist. In diesem Fall kann ausgewählt werden, ob die Eingänge gelesen (Input) oder die Ausgänge rückgelesen (Output) werden.
Anfang	Anfangsadresse (Offset) im Datenbereich des Transferbereichs im unterlagerten IO-System. Die Angabe hat die Form <Byte>.<Bit>. Beispiel: 0.0
Ende	Endadresse (Offset) im Datenbereich des Transferbereichs im unterlagerten IO-System. Die Angabe hat die Form <Byte>.<Bit>. Beispiel: 8.0

Diagnose von PROFINET-Geräten

Diagnose von PROFINET-Geräten

Die Diagnosewege, die Ihnen für PROFIBUS DP zur Verfügung stehen, sind die selben wie bei PROFINET IO. Die Vorgehensweise ist identisch. Etwas anders verhält es sich mit der Auswertung von Diagnoseinformationen über die erweiterten Anweisungen für die dezentrale Peripherie im Anwenderprogramm einer S7-300/400-CPU.

Für PROFINET IO gilt eine herstellerübergreifende Struktur für Datensätze mit Diagnoseinformationen. Diagnoseinformationen werden nur für gestörte Kanäle gebildet.

Über die Systemzustandslisten (SZLs) und die erweiterten Anweisungen "RALRM" und "RDREC" können Sie den Zustand von PROFINET IO-Systemen und die Diagnoseinformationen einem S7-Anwenderprogramm zur Verfügung stellen:

- Um einen Gesamtüberblick über den Zustand des PROFINET IO-Systems zu bekommen, lesen Sie z. B. die SZL 0x0X91 mit "RDSYSST".
- Um Diagnosedatensätze (Records) direkt von einer gestörten Baugruppe auszulesen, verwenden Sie die erweiterte Anweisung "RDREC" (Datensatz lesen). Sie erhalten damit zustandsbezogen detaillierte Fehlerinformationen.
- Um ereignisbezogen (d. h. durch Fehler-OB ausgelöst) Diagnosedatensätze zu lesen, verwenden Sie im entsprechenden Fehler-OB die erweiterte Anweisung "RALRM" (Alarmzusatzinfo lesen).

Die erweiterten Anweisungen "RALRM" und "RDREC" sind auch für PROFIBUS DP verwendbar.

Die Information, welche SZLs und welche Diagnosedatensätze für PROFINET IO definiert sind und wie die Diagnosedatensätze aufgebaut sind finden Sie hier (http://support.automation.siemens.com/WW/view/de/19289930).

IRT-Kommunikation projektieren

Einführung: Isochronous Realtime Ethernet

Einführung: Isochronous Realtime Ethernet

IRT ist ein Übertragungsverfahren, bei dem die PROFINET-Geräte mit sehr hoher Genauigkeit synchronisiert werden.

Ein Sync-Master gibt den Takt vor, Sync-Slaves synchronisieren sich auf diesen Takt. Sowohl ein IO-Controller, als auch ein IO-Device kann die Rolle des Sync-Masters innehaben.

Sync-Master und Sync-Slaves sind immer Teilnehmer einer Sync-Domain. Innerhalb der Sync-Domain wird für die IRT-Kommunikation Bandbreite reserviert. Real-Time- und Non-Real-Time-Kommunikation (TCP/IP-Kommunikation) ist außerhalb der reservierten Bandbreite möglich.

Zeitbereiche des Kommuniktionszyklus

Der Kommunikationszyklus gliedert sich in drei Zeitbereiche auf, die in der folgenden Grafik dargestellt werden:

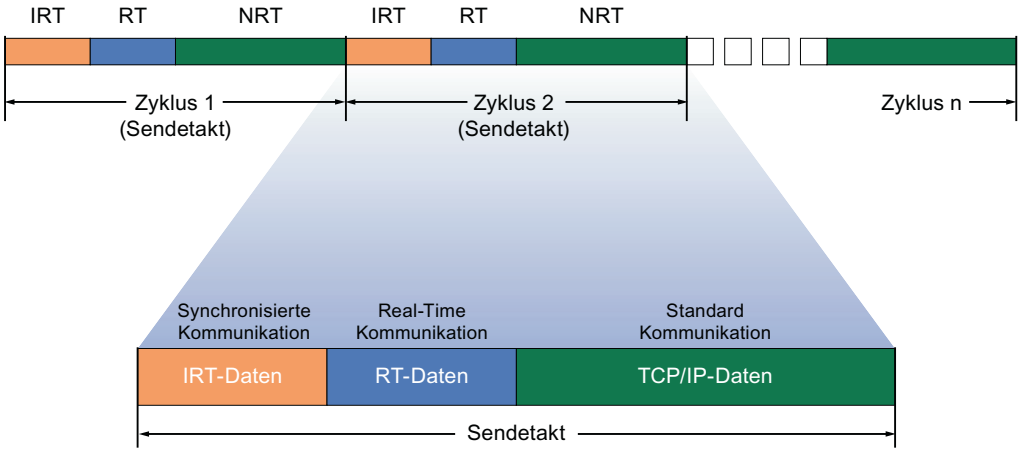

- IRT-Daten (synchronisierte Kommunikation)
  Diesen Zeitbereich können Sie abhängig vom Sendetakt in bestimmten Schritten reservieren. Innerhalb dieses Zeitbereichs werden ausschließlich IRT-Daten übertragen.
- RT-Daten (Real-Time Kommunikation)
  In diesem Zeitbereich werden die zyklischen RT-Daten übertragen. RT-Daten werden gegenüber "normalen" TCP/IP-Daten priorisiert. TCP/IP-Daten bzw. Ethernet-Telegramme können eine Priorität zwischen 1 und 7 haben. RT-Daten haben die Priorität 6.
- TCP/IP-Daten (Standard Kommunikation)
  Die Standard-Kommunikation (TCP/IP, etc.) wird schließlich in dem verbleibenden Intervall des Kommunikationszyklus übertragen.

Einsatzgebiete von IRT

PROFINET mit IRT eignet sich im Besonderen für:

- Hohe Performance und Deterministik bei großen Mengengerüsten bezüglich der Nutzdatenkommunikation (Produktivdaten)
- Hohe Performance auch bei vielen Teilnehmern in Linientopologie bezüglich der Nutzdatenkommunikation (Produktivdaten)
- Parallele Übermittlung von Produktiv- und TCP/IP-Daten über eine Leitung, auch bei hohem Datenaufkommen mit Sicherstellung der Weiterleitung der Produktivdaten durch Reservierung der Übertragungsbandbreite.

Siehe auch

Grundsätzliche Vorgehensweise zur Projektierung von IRT (Seite 3050)

PROFINET IO mit IRT konfigurieren (Seite 3050)

Sync-Domain (Seite 3052)

Grundsätzliche Vorgehensweise zur Projektierung von IRT

Vorgehensweise zur Projektierung von IRT

Wenn Sie in einer Anlage mit PROFINET IO um IRT erweitern wollen, gehen Sie in drei Schritten vor:

1. Konfigurieren Sie die PROFINET IO-Geräte. Die PROFINET-Geräte müssen IRT unterstützen.
2. Legen Sie fest, wer als Sync-Master die anderen Geräte synchronisiert. Dazu müssen Sie eine Sync-Domain mit einem Sync-Master und mindestens einem Sync-Slave projektieren.
3. Laden Sie die Projektierung in die beteiligten Geräte.

PROFINET IO-Konfiguration für IRT erstellen

Voraussetzung für die Konfiguration von IRT ist eine bestehende PROFINET IO-Konfiguration, d. h. ein oder mehrere Stationen müssen mit IO-Controller und IO-Devices konfiguriert sein.

Welche Komponenten für IRT-Kommunikation geeignet sind, können Sie in der Task Card "Information" ablesen, sobald Sie die entsprechende Komponente im Hardware-Katalog selektiert haben.

Allein durch die Projektierung eines PROFINET IO-Systems mit Komponenten, die IRT unterstützen, sind die Voraussetzungen für eine funktionierende IRT-Kommunikation gegeben.

PROFINET IO mit IRT konfigurieren

Einleitung

Wenn Sie die Genauigkeit des zyklischen Datenaustauschs über PROFINET IO mit IRT erhöhen wollen, konfigurieren Sie zuerst die PROFINET IO-Geräte. Diese PROFINET-Geräte müssen IRT unterstützen. Legen Sie danach fest, wer als Sync-Master die anderen Geräte synchronisiert. Dazu müssen Sie eine Sync-Domain mit einem Sync-Master und mindestens einem Sync-Slave konfigurieren.

Voraussetzung

- Ein IO-System mit einem IO-Controller und mindestens einem IO-Device ist vorhanden.
- Alle beteiligten Geräte unterstützen IRT.

Vorgehen

Um für ein bestehendes IO-System IRT zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PROFINET-Schnittstelle des **IO-Controllers**.
2. Navigieren Sie im Inspektorfenster zu "Erweitere Optionen> Echtzeit-Einstellungen > Synchronisation"

3. Weisen Sie dem IO-Controller unter "Synchronisationsrolle" die Rolle des Sync-Masters zu.
4. Selektieren Sie nun die PROFINET-Schnittstellen eines dazugehörigen **IO-Devices**.
5. Navigieren Sie im Inspektorfenster zu "Erweitere Optionen> Echtzeit-Einstellungen > Synchronisation".
6. Aktivieren Sie dort die RT-Klasse "IRT". Dem IO-Device wird daraufhin automatisch die Synchronisationsrolle "Sync-Slave" zugewiesen.
7. Über die Schaltfläche "Sync-Domain" können Sie ihre Einstellungen jederzeit prüfen und korrigieren.
8. Verschalten Sie die Ports zwischen Sync-Slaves und Sync-Master.

Oder

1. Heben Sie in der Netzsicht das PROFINET IO-System hervor.
2. Klicken Sie auf das PROFINET IO-System.
3. Navigieren Sie im Inspektorfenster zu dem Teilnehmer der gewünschten Sync-Domain.
4. Führen Sie alle benötigten Einstellungen direkt in den Tabellen durch:
   - Wählen Sie ein IO-System aus.
   - Stellen Sie bei dem IO-Controller die Synchronisationsrolle "Sync-Master" ein.
   - Stellen Sie bei den IO-Devices die RT-Klasse auf "IRT" um. Dadurch wird den IO-Devices automatisch die Synchronisationsrolle "Sync-Slave" zugewiesen.
5. Verschalten Sie die Ports zwischen Sync-Slaves und Sync-Master.

Sie können jetzt die Konfiguration mit PROFINET IRT in die beteiligten Geräte laden.

Hinweis

Automatisches Zuweisen eines Sync-Masters

Sie können auch einfach ein I-Device oder IO-Device als Sync-Slave projektieren, indem Sie die RT-Klasse von RT zu IRT umschalten. Der IO-Controller, der dem I-Device oder IO-Device zugewiesen wurde, wird automatisch als Sync-Master projektiert.

Haben Sie den IO-Controller als Sync-Slave projektiert, wird diese Einstellung nicht überschrieben. Der IO-Controller wird also nicht automatisch von Sync-Slave zu Sync-Master umgestellt.

Sind mehrere IO-Controller dem IO-Device als Shared Device zugewiesen, wird der IO-Controller automatisch als Sync-Master ausgewählt, dem die Schnittstelle des IO-Devices zugewiesen ist.

Hinweis

Umschaltung zwischen RT und IRT

In hierarchisch aufgebauten IO-Systemen kann IRT-Kommunikation mit RT-Kommunikation kombiniert werden, aber nicht in beiden IO-Systemen zur gleichen Zeit stattfinden: Wenn am I-Device die Option "Parametrierung der PN-Schnittstelle und deren Ports am übergeordneten IO-Controller" aktiviert ist, kann das I-Device am übergeordneten IO-Controller IRT-Kommunikation nutzen. Wenn die Option deaktiviert ist, kann das untergeordnete IO-System IRT-Kommunikation nutzen, aber am I-Device kann nicht zwischen RT und IRT umgeschaltet werden.

Sync-Domain

Sync-Domain

Für die Synchronisierung von PROFINET IO-Geräten wird eine Sync-Domain benötigt. Die Sync-Domain gewährleistet, dass alle Teilnehmer, die ihr angehören, im Gleichtakt kommunizieren.

Voraussetzung für die IRT-Kommunikation ist ein Synchronisations-Takt für alle PROFINET-Geräte in einer Sync-Domain zur Verteilung einer gemeinsamen Zeitbasis. Mit dieser Basissynchronisation wird ein Gleichlauf des Übertragungs-Zyklus der PROFINET-Geräte innerhalb einer Sync-Domain erzielt. Der Sync-Master (i. d. R. ein IO-Controller) generiert den gemeinsamen Synchronisationstakt und gibt die Zeitbasis vor, auf die sich alle anderen Sync-Slaves (i. d. R. IO-Devices) synchronisieren.

Wenn der Sync-Master ausfällt und kein redundanter Sync-Master konfiguriert wurde, dann fällt die Kommunikation der IRT-Geräte auf RT-Qualität zurück.

Abläufe im Hintergrund während der Konfigurierung eines IO-Systems

Konfigurieren Sie einen IO-Controller und vernetzen diesen mit einem Ethernet-Subnetz, wird er automatisch der Default Sync-Domain des Ethernet-Subnetzes zugewiesen. Die Default Sync-Domain ist immer vorhanden. Der IO-Controller wird unsynchronisiert betrieben.

Wenn Sie dem IO-System des IO-Controllers ein weiters IO-Device zuweisen, wird das IO-Device automatisch der Sync-Domain des IO-Controllers zugewiesen.

Siehe auch

Arbeiten mit Sync-Domains (Seite 3053)

Redundanten Sync-Master konfigurieren (Seite 3056)

Arbeiten mit Sync-Domains

Arbeiten mit Sync-Domains

Alle Komponenten, die an der IRT-Kommunikation teilnehmen, müssen einer Sync-Domain angehören.

Eine Sync-Domain ist eine Gruppe von PROFINET-Geräten, die auf einen gemeinsamen Takt synchronisiert sind. Genau ein Gerät hat die Rolle des Sync-Masters (Taktgeber), alle anderen Geräte haben die Rolle eines Sync-Slaves.

Default Sync-Domain

Wenn Sie ein Ethernet-Subnetz angelegt haben, ist automatisch eine spezielle Sync-Domain angelegt: die Default-Sync-Domain. Alle PROFINET-Geräte, die Sie für das Ethernet-Subnetz projektieren, gehören automatisch dieser Sync-Domain an.

Die folgenden Bilder verdeutlichen den Zusammenhang:

① Default-Domain; immer vorhanden

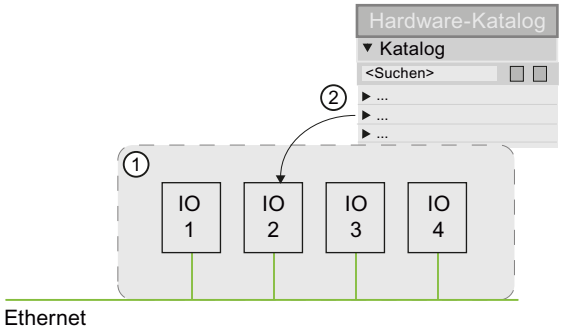

①	Default-Domain; immer vorhanden
②	Neu konfigurierte PROFINET IO-Systeme befinden sich zunächst in der Default-Domain.

Eigenschaften von Sync-Domains bearbeiten

Eigenschaften von Sync-Domains bearbeiten

Im Folgenden erfahren Sie, wie Sie den Namen und den Sendetakt einer Sync-Domain festlegen. Außerdem erfahren Sie, wie Sie eine Sync-Domain als Default-Domain festlegen.

Voraussetzung

- Mindestens ein IO-Device ist im Projekt angelegt.
- Es existiert bereits mindestens eine Sync-Domain.

Sync-Domain bearbeiten

Um den Namen einer Sync-Domain zu ändern, den Sendetakt anzupassen, oder um sie als Default-Domain festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ethernet-Subnetz in der Netzsicht, an dem die Sync-Domain angelegt ist.
2. Öffnen Sie die Eigenschaften des Ethernet-Subnetzes.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Domain-Verwaltung > Sync-Domains > <Name der Sync-Domain>".
4. Nehmen Sie die Änderungen vor:
   - Geben Sie den gewünschten Namen der Sync-Domain im Eingabefeld "Sync-Domain Name" ein.
   - Wählen Sie den Sendetakt aus der Klappliste "Sendetakt" aus.
   - Aktivieren Sie das Optionskästchen "Default-Domain", wenn Sie die Sync-Domain als Default-Domain festlegen möchten.

Bandbreiten für die verschiedenen RT-Klassen

Neben den Einstallungen für Name und Sendetakt werden Ihnen auch die verfügbaren und kalkulierten Bandbreiten für die verschiedenen RT-Klassen innerhalb einer Sync-Domain angezeigt.

Wählen Sie dazu den Abschnitt "Details" in der Bereichsnavigation unter dem Sync-Domain-Namen.

Siehe auch

Bandbreitennutzung für synchronisierte IO-Controller (Seite 3054)

Bandbreitennutzung für synchronisierte IO-Controller

Nutzung der Bandbreite

Hier stellen Sie ein, wie viel Sendezeit jeweils den zyklischen IO-Daten (Echtzeitdaten per IRT) und den übrigen Daten (Daten per TCP/IP oder UDP/IP, RT-Daten) zusteht.

Sie bestimmen damit, wie die Bandbreite (Übertragungskapazität) genützt wird.

Zyklische IO-Daten per IRT

Zyklische IO-Daten werden per IRT (Isochronous real time) verschickt.

Für IRT stehen bestimmte Zeitspannen exklusiv zur Verfügung.

Reservierte Zeitspanne für IRT

Die Dauer dieser reservierten Zeitspannen (Zeitschlitze) bestimmen Sie durch eine Auswahl in der Klappliste.

Alle übrigen Daten werden in der verbleibenden Zeit bis zum Beginn des nächsten Sendetakts übertragen.

Wählen Sie unter den folgenden Optionen:

- Maximal 25 % zyklische IO-Daten. Schwerpunkt auf nichtzyklische Daten (Daten per TCP/IP bzw. UDP/IP)
- Maximal 37,5 % zyklische IO-Daten. Schwerpunkt auf nichtzyklische Daten (Daten per TCP/IP bzw. UDP/IP)
- Maximal 50 % zyklische IO-Daten. Ausgewogenes Verhältnis
- Maximal 90 % zyklische IO-Daten. Schwerpunkt auf zyklische Daten (Daten per IRT)

Siehe auch

IRT mit höchster Performance projektieren (Seite 3168)

Teilnehmer einer Sync-Domain festlegen

Teilnehmer einer Sync-Domain festlegen

Alle Komponenten, die an der IRT-Kommunikation teilnehmen, müssen einer Sync-Domain angehören. Im Folgenden erfahren Sie, wie Sie die Teilnehmer einer Sync-Domain festlegen.

Voraussetzung

Es sind bereits PROFINET IO-Systeme projektiert sind, die sich in der Default-Domain befinden.

Einen neuen Teilnehmer in die Sync-Domain aufnehmen

Um einer Sync-Domain Teilnehmer zuzuweisen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ethernet-Subnetz in der Netzsicht, an dem die Sync-Domain angelegt ist.
2. Öffnen Sie die Eigenschaften des Ethernet-Subnetzes.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Domain-Verwaltung > Sync-Domains > <Name der Sync-Domain> > Teilnehmer".
4. Klicken Sie in der Tabelle "Station / IO-System" auf "Teilnehmer hinzufügen".
   Der Dialog "IO-System auswählen" wird geöffnet.
5. Selektieren Sie alle Teilnehmer, die Sie der Sync-Domain zuordnen wollen und klicken Sie auf "OK".

Die ausgewählten IO-Systeme werden aus der ursprünglichen Sync-Domain entfernt und der aktuellen Sync-Domain zugeordnet.

Einen Teilnehmer aus einer Sync-Domain entfernen

Um einen Teilnehmer aus einer Sync-Domain zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ethernet-Subnetz in der Netzsicht, an dem die Sync-Domain angelegt ist.
2. Öffnen Sie die Eigenschaften des Ethernet-Subnetzes.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Domain-Verwaltung > Sync-Domains > <Name der Sync-Domain> > Teilnehmer".
4. Selektieren Sie in der Tabelle "Station / IO-System" den Teilnehmer, den Sie aus der Sync-Domain entfernen möchten und drücken Sie <Entf>.

Der Teilnehmer wird gelöscht und wieder der Default-Domain zugeordnet.

Redundanten Sync-Master konfigurieren

Sie können für PROFINET IO-Geräte im IRT-Betrieb einen redundanten Sync-Master projektieren.

Beispiel: Eine S7-1500-CPU kann als Sync-Master projektiert werden, ein Scalance 6GK5 201-3BH00-2BA3 V5.0 kann als redundanter Sync-Master projektiert werden.

Funktionsweise

Ein redundanter Sync-Master gehört derselben Sync-Domain an wie der "primäre" Sync-Master. Er verhindert bei Ausfall des primären Sync-Masters den Ausfall der Kommunikation in der Sync-Domain.

Sowohl der Sync-Master als auch der redundante Sync-Master senden Sync-Telegramme. Ein Sync-Slave synchronisiert sich auf das Sync-Telegramm des Sync-Masters. Wenn der Sync-Master ausfällt, dann synchronisiert sich ein Sync-Slave automatisch auf die Sync-Telegramme des redundanten Sync-Masters.

Regeln

- Nur in einem einzigen Gerät in der Sync-Domain ist die Rolle des redundanten Sync-Masters zugewiesen.
- In der Sync-Domain sind nur IRT-Geräte vorhanden mit der Option "IRT mit hoher Performance".
- Zwischen Sync-Master und redundantem Sync-Master befinden sich maximal 2 Teilnehmer.

Vorgehen

Um einen redundanten Sync-Master zur projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht das Gerät, welches Sie als redundanten Master projektieren möchten.
2. Wählen Sie unter "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation > Synchronisationsrolle" die Option "Redundanter Sync-Master".

Regeln und Hinweise für IRT-Konfigurationen

Die folgenden Regeln und Hinweise beziehen sich auf PROFINET-Konfigurationen. Weitere Aufbauempfehlungen zur Optimierung Ihres PROFINET-Systems finden Sie hier.

PROFINET mit IRT aufbauen

Beachten Sie beim Aufbau und Betrieb eines PROFINET IO-Systems im IRT-Betrieb die folgenden Regeln. Sie dienen dazu, einen optimalen Betrieb Ihres PROFINET IO-Systems sicherzustellen.

- Bei der Verwendung von IRT müssen Sie die Topologie konfigurieren. Aktualisierungszeit, Bandbreite und weitere Parameter werden dadurch exakt berechnet.
- Möchten Sie mehrere Sync-Domains nutzen, projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.
- In einer Sync-Domain können Sie nur jeweils einen Sync-Master projektieren.
- Ein PROFINET IO-System darf nur einer einzigen Sync-Domain angehören.
- Wenn Sie PROFINET-Geräte in einer Sync-Domain projektieren und mit IRT synchronisieren wollen, müssen die betreffenden PROFINET-Geräte die IRT-Kommunikation unterstützen.
- Verwenden Sie nach Möglichkeit dasselbe PROFINET-Gerät als PROFINET IO-Controller und Sync-Master.
- Wird nur ein Teil der PROFINET-Geräte eines PROFINET IO-Systems synchronisiert, so beachten Sie Folgendes: PROFINET-Geräte, die nicht an der IRT-Kommunikation teilnehmen, platzieren Sie außerhalb der Sync-Domain.
- Übergeordnete und untergeordnete IO-Systeme können gleichzeitig RT-Kommunikation nutzen. IRT-Kommunikation kann mit RT-Kommunikation kombiniert werden, aber nicht in beiden IO-Systemen zur gleichen Zeit stattfinden: Wenn am I-Device die Option "Parametrierung der PN-Schnittstelle und deren Ports am übergeordneten IO-Controller" aktiviert ist, kann das I-Device am übergeordneten IO-Controller IRT-Kommunikation nutzen. Wenn die Option deaktiviert ist, kann das untergeordnete IO-System IRT-Kommunikation nutzen.

Siehe auch

IRT mit höchster Performance projektieren (Seite 3168)

Taktsynchronität projektieren

Was ist Taktsynchronität?

Wozu Taktsynchronität?

Wenn der öffentliche Nahverkehr so schnell führe, wie er nur könnte und seine Haltezeiten auf das absolute Minimum reduzierte, sähe der Fahrgast sehr oft nur noch die roten Schlusslichter. Die Gesamtfahrzeit wird aber von den jeweiligen Bahn-, Bus- oder U-Bahntakten bestimmt, denn mit einer fein abgestimmten Taktung geht es besser. Dies gilt auch in der dezentralen Automatisierungstechnik. Nicht nur schnelle Zyklen zählen, sondern erst die Abstimmung und Synchronisation der einzelnen Zyklen bringen den optimalen Durchsatz.

Just-In-Time

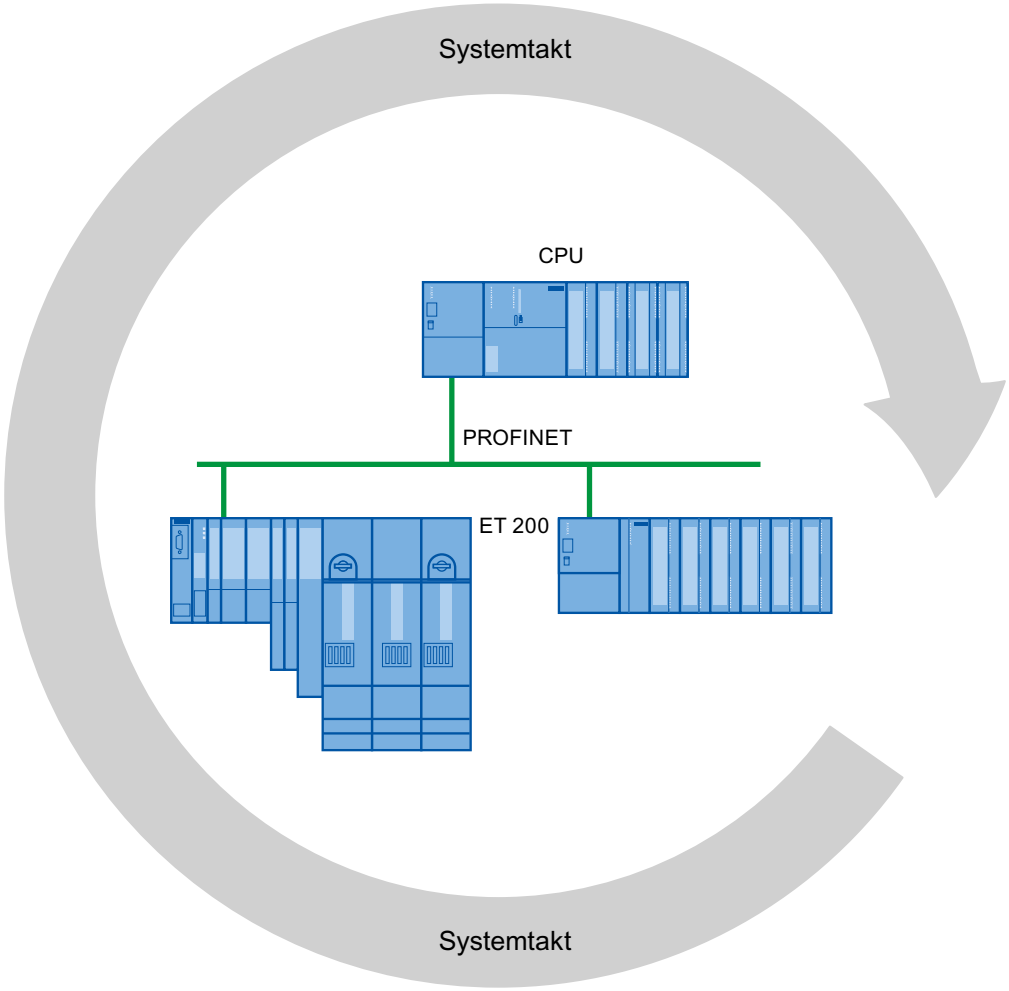

Die schnelle und zuverlässige Reaktionszeit einer Taktsynchronisation begründet sich darauf, dass alle Daten Just-In-Time zur Verfügung gestellt werden. Der äquidistante PROFINET IO-Zyklus schlägt hierfür den Takt.

Einsatz der Taktsynchronität

Einsatz der Taktsynchronität

Mit der Systemeigenschaft "Taktsynchronität" lassen sich Messwerte und Prozessdaten in einem festen Systemtakt erfassen. Im gleichen Systemtakt erfolgt die Signalverarbeitung bis zum Schalten auf die "Ausgangsklemme". Die Taktsynchronität trägt somit zu einer hohen Güte der Steuerung und damit zu einer größeren Fertigungsgenauigkeit bei. Mit der Taktsynchronität werden die möglichen Schwankungen der Prozessreaktionszeiten drastisch reduziert. Die zeitlich gesicherte Bearbeitung kann für einen höheren Maschinentakt genutzt werden.

Grundsätzlich bietet sich die Taktsynchronität überall dort an, wo Messwerte synchron erfasst werden müssen, Bewegungen koordiniert und Prozessreaktionen definiert und gleichzeitig erfolgen müssen.

Zeitlicher Ablauf der Synchronisation

Vom Einlesen der Eingangsdaten bis zum Ausgeben der Ausgangsdaten

Im Folgenden ist der prinzipielle zeitliche Ablauf aller an der Synchronisation beteiligten Komponenten erläutert:

- Taktsynchrones Einlesen der Eingangsdaten
- Transport der Eingangsdaten über das PROFINET-Subnetz zum IO-Controller (CPU)
- Weiterverarbeitung in der taktsynchronen Applikation der CPU

- Transport der Ausgangsdaten über das PROFINET-Subnetz zum ausgebenden IO-Device
- Taktsynchrones Ausgeben der Ausgangsdaten

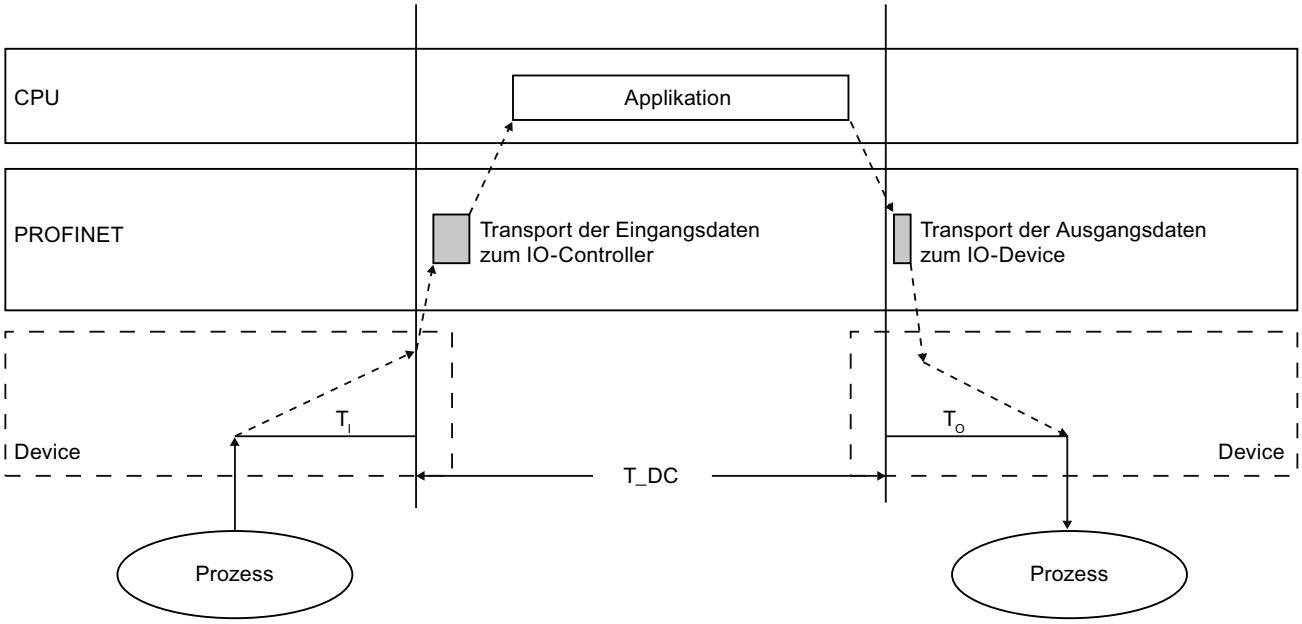

T_DC	Datenzyklus (Time_DataCycle)
T_I	Zeit zum Einlesen der Eingangsdaten
T_O	Zeit zur Ausgabe der Ausgangsdaten

Damit zum jeweils nächsten Beginn des PROFINET IO-Zyklus alle Eingangsdaten zum Transport über den PROFINET IO-Strang bereitstehen, wird der Peripherie-Einlesezyklus um eine Vorlaufzeit T_I früher gestartet. Das T_I ist das "Blitzlicht" für die Eingänge; zu diesem Zeitpunkt werden alle synchronisierten Eingänge eingelesen. T_I ist notwendig, um Analog-Digital-Umwandlung, Rückwandbuszeiten und ähnliches zu kompensieren. Die Vorlaufzeit T_I kann von STEP 7 automatisch eingestellt, aus den Einstellungen des zugehörigen Taktsynchronalarm-OBs übernommen oder vom Anwender projektiert werden. Es empfiehlt sich, die Vorlaufzeit T_I von STEP 7 automatisch vergeben zu lassen.

Der PROFINET IO-Strang transportiert die Eingangsdaten zum IO-Controller. Der Taktsynchronalarm-OB (OB 61, OB 62, OB 63 bzw. OB 64) wird nach einer parametrierbaren Verzögerungszeit aufgerufen. Das Anwenderprogramm im Taktsynchronalarm-OB bestimmt die Prozessreaktion und stellt die Ausgangsdaten rechtzeitig bis zum Beginn des nächsten Datenzyklus bereit. Die Länge des Datenzyklus wird immer vom Anwender projektiert.

T_O ist die Kompensation aus dem Rückwandbus und der Digital-Analog-Umwandlung innerhalb des Devices. Das T_O ist das "Blitzlicht" für die Ausgänge; zu diesem Zeitpunkt werden die synchronisierten Ausgänge ausgegeben. Die Zeit T_O kann von STEP 7 automatisch eingestellt, aus den Einstellungen des zugehörigen Taktsynchronalarm-OBs übernommen oder vom Anwender projektiert werden. Es empfiehlt sich, die Zeit T_O von STEP 7 automatisch vergeben zu lassen.

Taktsynchrone und nicht taktsynchrone dezentrale Peripherie

Es ist möglich, taktsynchrone dezentrale Peripherie mit nicht taktsynchroner dezentraler Peripherie an einem IO-Controller zu kombinieren.

Der Wert Ti

Wirkung von T_I

Die Wirkung von T_I wird durch das folgende Bild deutlich:

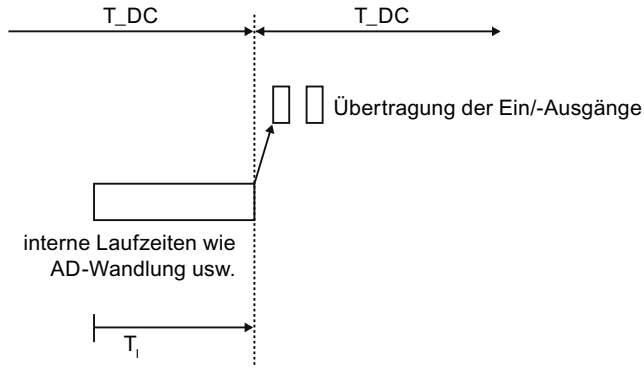

T_DC	Datenzyklus
T_I	Zeit zum Einlesen der Eingangsdaten

Ablauf

Damit zum Startzeitpunkt des neuen Systemtaktes ein konsistenter Zustand der Eingänge zum IO-Controller übertragen werden kann, muss der Einlesevorgang um die Zeit T_I vorverlegt werden. Die Zeit T_I umfasst für ein bestimmtes Eingangsmodul minimal die Signalaufbereitungs- und Wandlungszeit an den Elektronikmodulen und die Zeit zur Übertragung zum Interface Modul am IO-Device-Rückwandbus.

In der Anlage wird erreicht, dass die Werte zeitgleich eingelesen werden, indem das T_I aller taktsynchron gelesenen Eingangsmodule auf einen gleichen Wert gesetzt wird, der größer oder gleich der größten minimalen T_I aller taktsynchronen Eingangsmodule ist. STEP 7 stellt mit der Standardeinstellung sicher, dass eine gemeinsame kleinstmögliche T_I eingestellt wird.

Der Wert To

Wirkung von T_O

Die Wirkung wird durch das folgende Bild deutlich:

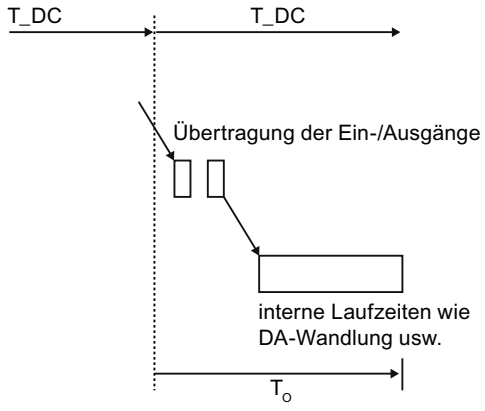

T_DC	Datenzyklus
T_O	Zeitpunkt der Ausgabe der Ausgangsdaten

Ablauf

Damit zum Startzeitpunkt des neuen Systemtaktes ein konsistenter Zustand der Ausgänge zum Prozess übertragen werden kann, erfolgt die Ausgabe an der Klemme erst zum Zeitpunkt T_O nach dem Taktschlag. Die Zeit T_O umfasst für ein bestimmtes Ausgangsmodul minimal die Zeit der Übertragung vom IO-Controller zum IO-Device (über PROFINET IO) und im IO-Device die Übertragung der Ausgänge vom Interfacemodul zum Elektronikmodul (Rückwandbus) (ggf. zzgl. Zeit für Digital-Analog-Wandlung).

Dass die Werte zeitgleich geschrieben werden wird in der Anlage erreicht, indem das T_O aller taktsynchronen Ausgangsmodule auf einen gleichen Wert gesetzt wird. Dieser Wert muss größer oder gleich der größten minimalen T_O aller taktsynchronen Ausgangsmodule sein. STEP 7 berechnet automatisch eine gemeinsame kleinstmögliche T_O.

Der Taktsynchronalarm-OB (OB 6x)

Zur taktsynchronen Anbindung des Anwenderprogramms nutzen Sie den Taktsynchronalarm OB (z. B. OB 61).

Aufruf des Taktsynchronalarm-OBs

Der Taktsynchronalarm-OB wird über den Systemtakt getriggert; genauer über eine einstellbare Verzögerungszeit nach dem Systemtakt. Die Wirkung wird durch das folgende Bild deutlich:

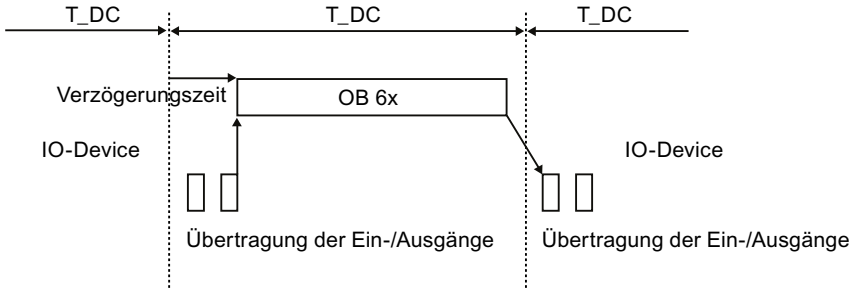

T_DC Datenzyklus

Verzögerungszeit

Für die Verzögerungszeit berechnet STEP 7 zunächst automatisch einen geeigneten Wert. Die Verzögerungszeit kompensiert die Übertragungszeit der Eingänge von den taktsynchronen IO-Devices zum IO-Controller über das PROFINET IO-Netzwerk. Die Ausführung des OB 6x ist an den Systemtakt und die eingestellte Verzögerungszeit gebunden. Die Verzögerungszeit kann ggf. auch manuell korrigiert werden. Sie können z. B. die Verzögerungszeit verkürzen, um zu Beginn des OB 6x schon Programmteile abzuarbeiten, die nicht auf die taktsynchrone Peripherie zugreifen. Sie gewinnen dadurch Laufzeit für das taktsynchrone Anwenderprogramm.

Anbindung der taktsynchronen Peripherie an den Taktsynchronalarm-OB

Auf die taktsynchrone Peripherie greifen Sie über ein Teilprozessabbild zu, d. h. die Adressen der taktsynchronen Module müssen in einem Teilprozessabbild liegen.

Zugriffe auf die taktsynchrone Peripherie sind durch die Anweisungen SYNC_PI und SYNC_PO innerhalb des Taktsynchronalarm-OBs zu programmieren.

SYNC-PI rufen Sie zu Beginn des OB 6x auf; vorausgesetzt, Sie haben für die Verzögerungszeit die automatische Einstellung gewählt. SYNC_PO rufen Sie am Ende des OB 6x auf.

T_DC, Applikationszyklus und Sendetakt

Die Zeit T_DC bezeichnet die Zeit für einen Datenzyklus, d. h. diese Zeit ist die kleinstmögliche Zeit für das Einlesen, Verarbeiten und Ausgeben der Daten (EVA-Modell). T_DC ist immer ein Vielfaches des Sendetaktes.

Wenn IRT für ein IO-Device aktiviert ist, dann ist T_DC gleich dem Sendetakt. Die Aktualisierungszeit entspricht dem Sendetakt.

Der Applikationszyklus ist ein Vielfaches des Datenzyklus. Er kann identisch mit dem Datenzyklus sein, wenn die Laufzeit des Taktsynchronalarm-OBs entsprechend kurz ist.

Programmbearbeitung nach dem EVA-Modell mit kurzer Zeit

Das EVA-Modell mit Applikationszyklus-Faktor = 1 zeichnet sich dadurch aus, dass die Bearbeitung der E-/A-Daten innerhalb eines Systemtaktes T_DC abgeschlossen wird. Mit diesem Modell erreicht man die kürzesten Reaktionszeiten.

Programmbearbeitung nach dem EVA-Modell mit langer Zeit

Kennzeichen des EVA-Modells mit Applikationszyklus-Faktor > 1

Wenn die Laufzeit des Taktsynchron-OBs länger ist als ein Datenzyklus T_DC, muss die Ausgabe um einen Applikationstakt (OB 6x) verzögert werden. Die Ausgabe erfolgt deshalb bei diesem Modell vor der Eingabe.

Die Anweisungen SYNC_PI und SYNC_PO müssen dann im Ausführungsfenster des ersten Datenzyklus (steht bei Applikationszyklus-Faktor > 1 nur dort zur Verfügung) aufgerufen werden.

Als Ausführungsfenster wird der Zeitraum bezeichnet, in dem ein Aufruf der Anweisungen SYNC_PI bzw. SYNC_PO möglich ist; d. h. nach Übertragung der zyklischen Daten über PROFINET bis kurz vor Ende des Datenzyklus T_DC.

Die sonstige Anwenderprogrammbearbeitung im OB 6x erfolgt dann im nächsten oder über die folgenden Datenzyklen.

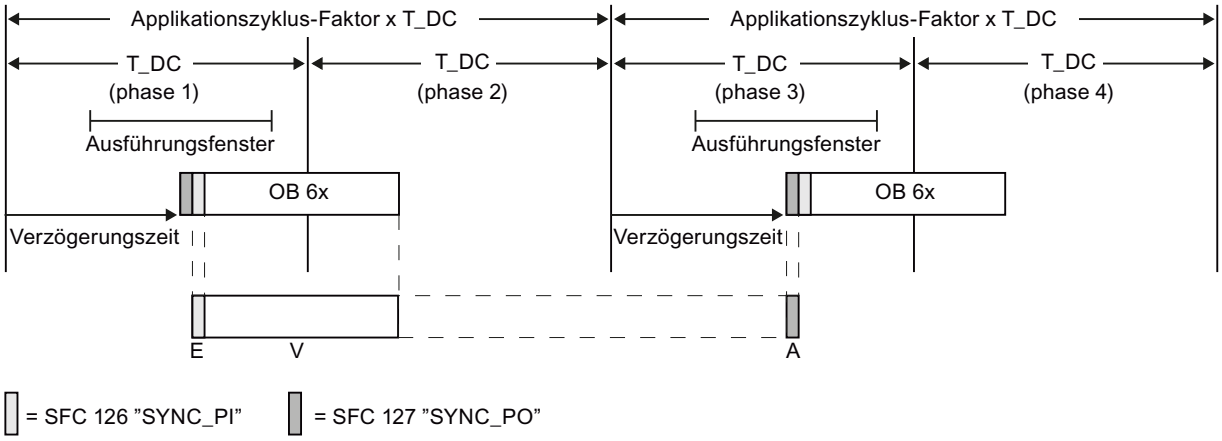

Die Abbildung zeigt den Signalverlauf mit Applikationszyklus-Faktor = 2 (Applikationszyklus = 2 x Datenzyklus) von der Erfassung (E) über die Verarbeitung (V) im IO-Controller bis zur Ausgabe (A) der Prozesswerte. STEP 7 berechnet Ihnen die Verzögerungszeit. In dieser Zeit sind die gelesenen Eingangsdaten auf dem PROFINET IO unterwegs.

Dieses Modell ist besonders für große Peripherieausbauten mit einem umfangreichen Anwenderprogramm im OB 6x geeignet. Mit diesem Modell sind für die Bearbeitung der Eingangsdaten und Ermittlung der korrespondierenden Ausgangsdaten längere Rechenzeiten möglich.

Die konstante Bearbeitungszeit von der "Eingangsklemme" zur "Ausgangsklemme" beträgt T_I + (Applikationszyklus-Faktor + 1) x T_DC + T_O. Als Prozessreaktionszeit kann T_I + (2 x Applikationszyklus-Faktor + 1) x T_DC + T_O sichergestellt werden.

Taktsynchronität projektieren

Einleitung

Im Folgenden wird die Projektierung der Taktsynchronität eines Moduls anhand der neuen dezentralen Peripheriefamilien ET 200MP als IO-Devices beschrieben. Die beschriebene Vorgehensweise ist aber auch für andere dezentrale Peripheriesysteme gültig (z. B. ET 200S oder ET 200SP).

Der IO-Controller ist eine S7-1500-CPU.

Voraussetzung

- Die Netzsicht ist geöffnet.
- Eine S7-1500-CPU ist platziert (z. B. CPU 1516-3 PN/DP).
- Ein dezentrales Peripheriegerät ist platziert (z. B. IM 155-5 PN HF) und mit der CPU vernetzt.
- Alle Voraussetzungen für eine IRT-Konfiguration sind erfüllt, d. h:
  - Die Ports der CPU und der IM sind miteinander verschaltet.
  - Die RT-Klasse der IM ist auf "IRT" eingestellt (Bereich "Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation").

Vorgehen

Um eine taktsynchrone Anbindung von Peripherie und Anwenderprogramm zu erstellen gehen Sie folgendermaßen vor:

1. Markieren Sie das dezentrale Peripheriegerät (die IM 155-5 PN HF) in der Netzsicht und wechseln Sie in die Gerätesicht.
2. Stecken Sie ein Peripheriemodul, das taktsynchron betrieben werden kann (Modul mit dem Namenszusatz "HF", z. B. DI 16x24VDC HF - 6ES7 521-1BH00).
3. Navigieren Sie im Inspektorfenster des markierten Peripheriemoduls zum Bereich "E/A-Adressen".
4. Nehmen Sie im Bereich E/A-Adressen folgende Einstellungen vor:
   - Aktivieren Sie die Option "Taktsynchroner Betrieb".
   - Wählen Sie ein Teilprozessabbild aus, z. B. Teilprozessabbild 1.
   - Klicken Sie auf die Klappliste "Zugeordneter Organsisationsbaustein" und klicken Sie auf die Schaltfläche "Objekt hinzufügen".
     Ein Dialog zur Auswahl von Organisationsbausteinen wird geöffnet.
   - Wählen Sie den OB "Synchronous Cycle" und bestätigen Sie die Auswahl mit "OK".
     Bei automatischer Nummernvergabe wird der OB 61 erzeugt und geöffnet. Im Inspektorfenster ist der Bereich "Taktsynchronität" markiert und Sie können unmittelbar mit der Einstellung von Applikationszyklus und Verzögerungszeit fortfahren sowie im Anweisungsteil mit der Programmierung des OB beginnen.

5. Fügen Sie weitere IO-Devices nach Bedarf hinzu, vernetzen und verschalten sie und passen die Konfiguration und die taktsynchronen Einstellungen an.
6. Zum Abrufen von Informationen über berechnete Bandbreiten oder zum Anpassen des Sendetakts markieren Sie das Subnetz und navigieren in den entsprechenden Bereich des Sync-Domain-Managements im Inspektorfenster.

Taktsynchronität projektieren

Einleitung

Im Folgenden wird die Projektierung der Taktsynchronität eines Moduls anhand der neuen dezentralen Peripheriefamilien ET 200MP als IO-Devices beschrieben. Die beschriebene Vorgehensweise ist aber auch für andere dezentrale Peripheriesysteme gültig (z. B. ET 200S oder ET 200SP).

Der IO-Controller ist eine S7-300/400-CPU.

Voraussetzung

- Die Netzsicht ist geöffnet.
- Eine S7-300/400-CPU ist platziert (z. B. CPU 319-3 PN/DP).
- Ein dezentrales Peripheriegerät ist platziert (z. B. IM 155-5 PN HF) und mit der CPU vernetzt.
- Alle Voraussetzungen für eine IRT-Konfiguration sind erfüllt, d. h:
  - Die Ports der CPU und der IM sind miteinander verschaltet.
  - Die Synchronisationsrolle der PROFINET-Schnittstelle ist auf "Sync-Master" eingestellt (Eigenschaften der CPU, Bereich "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation")
  - Die RT-Klasse der IM ist auf "IRT" eingestellt (Bereich "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation").

Vorgehen

Um eine taktsynchrone Anbindung von Peripherie und Anwenderprogramm zu erstellen gehen Sie folgendermaßen vor:

1. Markieren Sie die CPU und navigieren Sie im Inspektorfenster zum Bereich "Alarme > Taktsynchronalarme".
2. Markieren Sie das dezentrale Peripheriegerät (die IM 155-5 PN HF) in der Netzsicht und wechseln Sie in die Gerätesicht.
3. Stecken Sie mindestens ein Peripheriemodul, das taktsynchron betrieben werden kann (Modul mit dem Namenszusatz "HF", z. B. DI 16x24VDC HF - 6ES7 521-1BH00).
4. Weisen Sie dem Peripheriemodul ein Teilprozessabbild zu (Inspektorfenster, Bereich "E/A-Adressen").

5. Markieren Sie die IM in der Gerätesicht oder in der Netzsicht und navigieren Sie im Inspektorfenster zum Bereich "PROFINET-Schnittstelle > Erweiterte Optionen > Taktsynchronisation".
   - Aktivieren Sie die Option "Taktsynchroner Betrieb".
   - Wenn Sie den Sendetakt oder den Applikationszyklus ändern wollen, klicken Sie über auf den grünen Link neben den entsprechenden Feldern. Sie erreichen hierdurch die Sync-Domain, in der diese Einstellungen zentral geändert werden können.
   - Wählen Sie für die Ti/To-Zeiten, ob sie automatisch minimal berechnet werden sollen oder ob Sie sie manuell einstellen wollen. Nur bei manueller Einstellung sind die Ti/To-Werte änderbar.
   - In der Detailübersicht die Module: Aktivieren Sie die Option "Taktsynchroner Betrieb" für die Module, die taktsynchron betrieben werden sollen.
6. Fügen Sie weitere IO-Devices nach Bedarf hinzu, vernetzen und verschalten sie und passen die Konfiguration und die taktsynchronen Einstellungen an.
7. Zum Abrufen von Informationen über berechnete Bandbreiten oder zum Anpassen des Sendetakts markieren Sie das Subnetz und navigieren in den entsprechenden Bereich des Sync-Domain-Managements im Inspektorfenster.

Taktsynchronität am Subnetz einstellen

Einleitung

Die Eigenschaften des Subnetzes bieten neben den Sync-Domain-Eigenschaften und MRP-Eigenschaften eine Übersicht auf die Einstellungen der Taktsynchronität.

Voraussetzung

Ein PROFINET IO-System mit IO-Controller und IO-Devices ist projektiert.

Vorgehen

Um die Taktsynchronitätseigenschaften des Subnetzes anzuzeigen, gehen Sie folgendermaßen vor:

1. Markieren Sie das Subnetz.
2. Wählen Sie in den Eigenschaften den Bereich "Übersicht Taktsynchroner Betrieb".

Einstellungen in der Übersicht Taktsynchroner Betrieb

CPU	Auswahl einer CPU (IO-Controller) des Subnetzes.
Taktsynchrone Applikation	Auswahl eines OBs der gewählten CPU, der einem IO-System zugeordnet ist.
Details der taktsynchronen Applikation	

Applikationszyklus	Anzeige des Applikationszyklus (Vielfaches des Datenzyklus). Für S7-300/400 stellen Sie die Zeit in den CPU-Eigenschaften ein, für S7-1500 stellen Sie die Zeit in den Taktsynchron-OB-Eigenschaften ein.
Verzögerungszeit	Anzeige der Zeit zwischen Systemtakt und Start des Taktsynchronalarm-OBs (z. B. OB 61).
Sendetakt	Anzeige des Zeitraums zwischen zwei aufeinander folgenden Intervallen für IRT- bzw. RT-Kommunikation.
IO-System Nr.	Anzeige der Nr. des PROFINET IO-Systems, das dem OB zugeordnet ist.
Teilprozessabbild	Anzeige aller Teilprozessabbild-Nummern, die dem PROFINET IO-System zugeordnet sind.
Ti/To Werte am OB	
Ti/To-Werte automatisch	Wenn diese Option aktiviert ist, dann berechnet STEP 7 jeweils einen Ti-Wert und einen To-Wert, der für alle Module, die taktsynchron betrieben werden und dem OB zugeordnet sind, geeignet ist. In der Tabelle IO-Devices/Modules können Sie auf diese gemeinsamen Ti/To-Werte referenzieren (Ti/To-Werte = vom OB).
Zeit Ti	Zeit zum Einlesen der taktsynchronen Eingangsdaten.
Zeit To	Zeit zum Ausgeben der taktsynchronen Ausgangsdaten.
IO-Devices / Module	
Tabelle mit Ti/To-Werten:	Für jedes IO-Device/Modul im IO-Device kann einzeln der taktsynchrone Betrieb aktiviert bzw. deaktiviert werden. Ti/To Werte: Drei Einstellungen möglich: • Automatisch: Ti/To wird automatisch berechnet. • Manuell: Ti/To kann editiert werden. • Vom OB: Ti/To-Werte werden vom OB übernommen. Ti, To: nur editierbar bei Einstellung "Manuell". TiMin/TiMax: Minimale bzw. maximale Zeit für Ti. ToMin/ToMax: Minimale bzw. maximale Zeit für To. Intervall: Vorgegebenes Intervall für Ti/To-Werte.

Regeln, Voraussetzungen und Randbedingungen

Was für die Projektierung von Taktsynchronität zu beachten ist

Berücksichtigen Sie folgende Hinweise für die Projektierung der Taktsynchronität:

Komponenten und PROFINET-Schnittstellen

- Die Taktsynchronität ist nur mit den in der CPU integrierten Schnittstellen möglich. Bei S7-1500-CPUs mit zwei PROFINET-Schnittstellen ist nur die Schnittstelle X1 für Taktsynchronität nutzbar. Der taktsynchrone Betrieb mit S7-1500-CPs/S7-1500-CMs ist nicht möglich.
- Eine volle Taktsynchronität von "Klemme" zu "Klemme" ist nur dann möglich, wenn alle in der Kette beteiligten Komponenten die Systemeigenschaft "Taktsynchronität" unterstützen. Achten Sie bei der Auswahl im Katalog auf den Eintrag "Taktsynchronität" bzw. "taktsynchrone Bearbeitung" im Informationsfeld der Baugruppe.
- Wenn Sie ein IO-Device taktsynchron betreiben (d. h. indem Sie ihm z. B. die Rolle Sync-Slave zugewiesen haben), dann muss mindestens ein taktsynchron betriebenes Modul bzw. Submodul im IO-Device vorhanden sein.

Strangübergreifende Taktsynchronität

- Eine strangübergreifende Taktsynchronität ist zur Zeit nicht möglich.

Teilprozessabbilder

- Die taktsynchrone Peripherie kann nur in Teilprozessabbildern verarbeitet werden. Ohne den Einsatz von Teilprozessabbildern ist keine taktsynchrone konsistente Datenübertragung möglich. Die Einhaltung von Mengengerüsten wird überwacht, da IO-Device-Anzahl und Anzahl Bytes am IO-System pro Teilprozessabbild begrenzt sind.
- Die Adressen der taktsynchronen Peripherie müssen in einem Teilprozessabbild liegen.
- Das Teilprozessabbild für taktsynchrone Peripherie darf auch nicht-taktsynchrone Peripherie enthalten.
- S7-300/400: Mehr als ein Teilprozessabbild für einen Taktsynchronalarm-OB bzw. ein IO-System ist möglich.
- S7-1500: Genau ein Teilprozessabbild für einen Taktsynchronalarm-OB bzw. ein IO-System ist möglich. Nur Module bzw. Submodule desselben IO-Systems bzw. desselben Taktsynchronalarm-OBs dürfen im selben Teilprozessabbild liegen.

Zeiten

- Wenn der taktsynchrone Modus aktiv ist, dann ist die von STEP 7 berechnete Aktualisierungszeit in den IO-Devices immer identisch mit dem Sendetakt. Eine manuelle Änderung der Aktualisierungszeit ist nicht möglich.
- Die Zeit für den Applikationszyklus ist auf 32 ms begrenzt.
- Die Verzögerungszeit für den Taktsynchronalarm-OB muss zwischen 0 und (Datenzyklus - 1 µs) liegen.

I-Devices konfigurieren

Funktionalität I-Device

Funktionalität I-Device

Die Funktionalität "I-Device" (Intelligentes IO-Device) einer CPU erlaubt es, Daten deterministisch mit einem IO-Controller auszutauschen und somit die CPU z. B. als intelligente Vorverarbeitungseinheit von Teilprozessen einzusetzen. Das I-Device ist hierbei als IO-Device an einen "übergeordneten" IO-Controller angebunden.

Die Vorverarbeitung wird durch das Anwenderprogramm in der CPU sichergestellt. Die in zentraler oder dezentraler (PROFINET IO oder PROFIBUS DP) Peripherie erfassten Prozesswerte werden durch das Anwenderprogramm vorverarbeitet und über eine PROFINET IO-Device-Schnittstelle der CPU bzw. des CPs einer übergeordneten Station zur Verfügung gestellt.

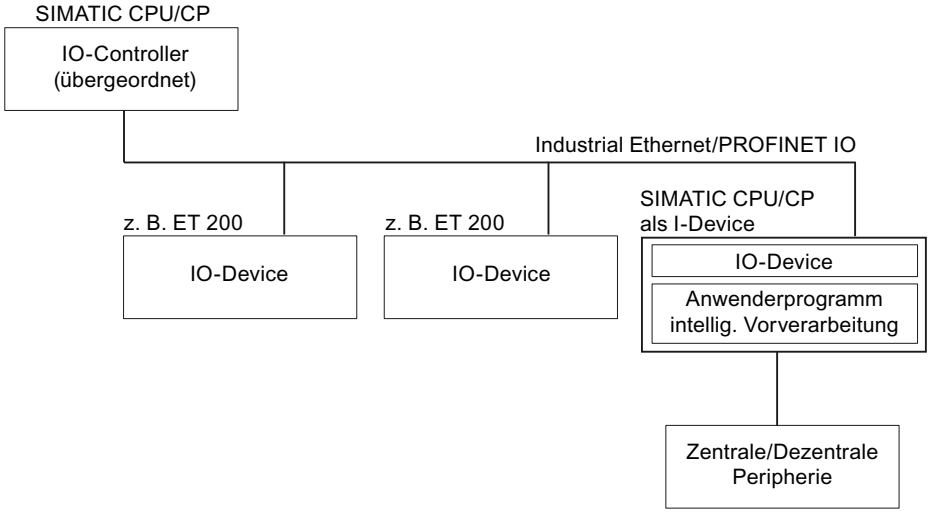

Namenskonvention "I-Device"

In der weiteren Beschreibung wird eine CPU bzw. ein CP mit der Funktionalität I-Device vereinfacht als "I-Device" bezeichnet.

Eigenschaften und Vorteile des I-Devices

Einsatzbereiche

Einsatzbereiche des I-Devices:

- Verteilte Verarbeitung
  Eine komplexe Automatisierungsaufgabe lässt sich in kleinere Einheiten/Teilprozesse aufteilen. Dadurch ergeben sich überschaubare Prozesse, die zu vereinfachten Teilaufgaben führen.
- Entkopplung von Teilprozessen
  Komplizierte, weit verteilte und umfangreiche Prozesse können durch die Verwendung von I-Devices in mehrere Teilprozesse mit überschaubaren Schnittstellen gegliedert werden. Diese Teilprozesse können ggf. jeweils in einzelnen Projekten hinterlegt werden, die später zu einem Gesamtprojekt zusammengestellt werden.
- Know-How Schutz
  Anlagenteile können statt mit einem Projekt nur noch mit einer GSD-Datei zur Schnittstellenbeschreibung des I-Devices ausgeliefert werden. Das Know-How des Anwenderprogramms muss damit nicht mehr offen gelegt werden.

Eigenschaften

Eigenschaften des I-Devices:

- Entkopplung von Projekten
  Ersteller und Nutzer eines I-Devices können komplett getrennte Automatisierungprojekte haben. Die Schnittstelle zwischen den Projekten bildet die GSD-Datei. Damit ist die Kopplung über eine standardisierte Schnittstelle zu Standard-IO-Controllern möglich.
- Echtzeitkommunikation
  Das I-Device wird einem deterministischen PROFINET IO-System über eine PROFINET IO-Schnittstelle zur Verfügung gestellt und unterstützt somit die Echtzeitkommunikation RT (Real-Time).

Vorteile

Das I-Device bietet folgende Vorteile:

- Einfache Kopplung von IO-Controllern ohne zusätzliche Software-Tools.
- Echtzeitkommunikation zwischen SIMATIC-CPUs und zu Standard IO-Controllern.
- Durch die Verteilung der Rechenleistung auf mehrere I-Devices kann die notwendige Rechenleistung der einzelnen CPUs und natürlich des IO-Controllers vermindert werden.
- Geringere Kommunikationslast durch Verarbeitung der Prozessdaten vor Ort.
- Übersichtlichkeit durch Bearbeitung der Teilaufgaben in getrennten CPUs bzw. in getrennten Projekten.

Datenaustausch zwischen über- und untergeordnetem IO-System

Einleitung

Im folgenden Kapitel wird der Datenaustausch zwischen über- und untergeordnetem IO-System dargestellt.

Transferbereiche

In den Transferbereichen werden die Daten für die Kommunikation zwischen IO-Controller und I-Device bereitgestellt. Ein Transferbereich enthält dabei eine Informationseinheit, die konsistent zwischen IO-Controller und I-Device ausgetauscht wird.

Applikationstransferbereiche sind eine Schnittstelle zum Anwenderprogramm der I-Device-CPU. Eingänge werden im Anwenderprogramm verarbeitet und Ausgänge sind das Ergebnis einer Verarbeitung im Anwenderprogramm.

Im folgenden Bild ist der Datenaustausch zwischen über- und untergeordnetem IO-System dargestellt. Die einzelnen Kommunikationsbeziehungen werden anhand der Ziffern nachfolgend erklärt.

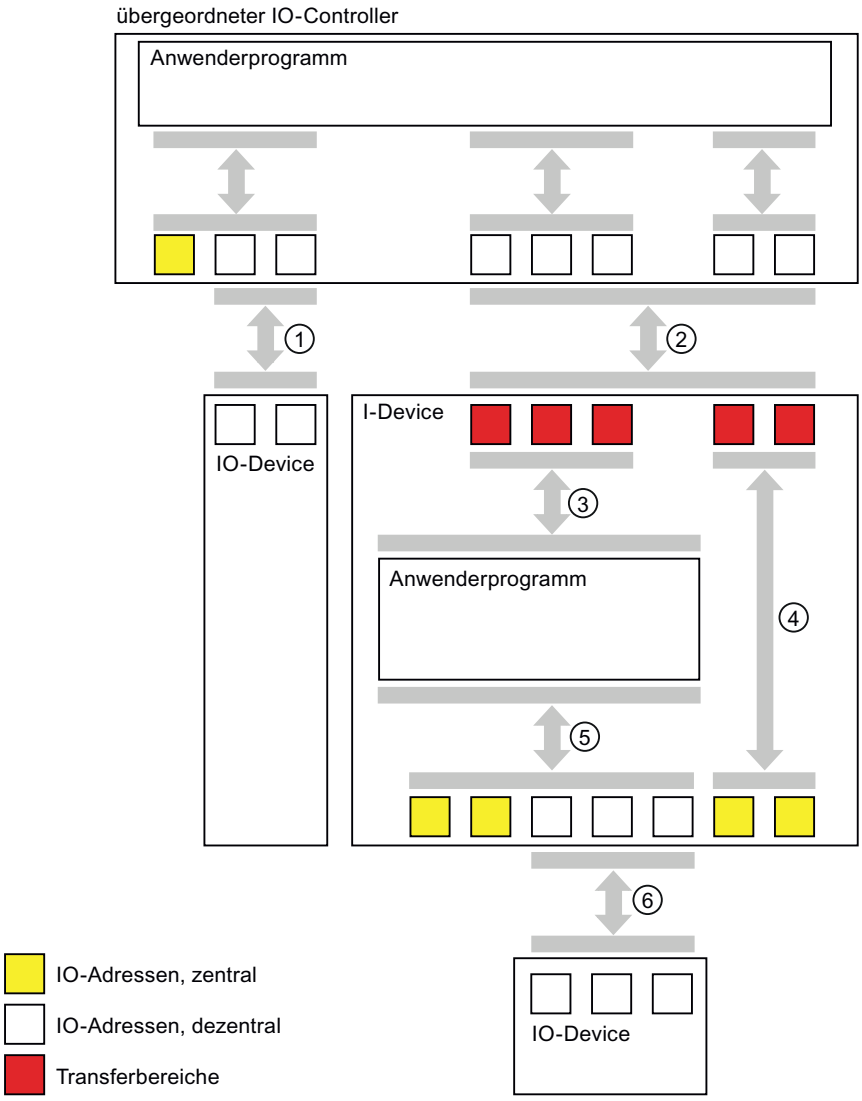

① Datenaustausch zwischen übergeordnetem IO-Controller und normalem IO-Device

Über diesen Weg tauschen IO-Controller und IO-Devices Daten über PROFINET aus.

② Datenaustausch zwischen übergeordnetem IO-Controller und I-Device

Über diesen Weg tauschen der IO-Controller und das I-Device Daten über PROFINET aus.

Der Datenaustausch zwischen einem übergeordneten IO-Controller und einem I-Device basiert auf der herkömmlichen IO-Controller / IO-Device-Beziehung.

Für den übergeordneten IO-Controller stellen die Transferbereiche des I-Devices Submodule einer vorkonfigurierten Station dar.

Die Ausgangsdaten des IO-Controllers sind die Eingangsdaten des I-Devices. Analog dazu sind die Eingangsdaten des IO-Controllers die Ausgangsdaten des I-Devices.

③ **Transferbeziehung zwischen dem Anwenderprogramm und dem Transferbereich**

Über diesen Weg tauschen Anwenderprogramm und Applikationstransferbereich Ein- und Ausgangsdaten aus.

④ **Transferbeziehung zwischen dem Transferbereich und der Peripherie des I-Devices**

Wird von der Projektierung aktuell nicht unterstützt.

⑤ **Datenaustausch zwischen dem Anwenderprogramm und der Peripherie des I-Devices**

Über diesen Weg tauschen Anwenderprogramm und zentrale / dezentrale Peripherie Ein- und Ausgangsdaten aus.

⑥ **Datenaustausch zwischen dem I-Device und einem untergeordneten IO-Device**

Über diesen Weg tauschen das I-Device und seine IO-Devices Daten aus. Die Datenübertragung erfolgt per PROFINET.

Diagnoseadressen konfigurieren

Einführung

Bei der Vergabe von Diagnoseadressen für PROFINET-IO müssen Sie beachten, dass sowohl dem IO-Controller als auch dem I-Device IO-Diagnoseadressen zugeordnet sind.

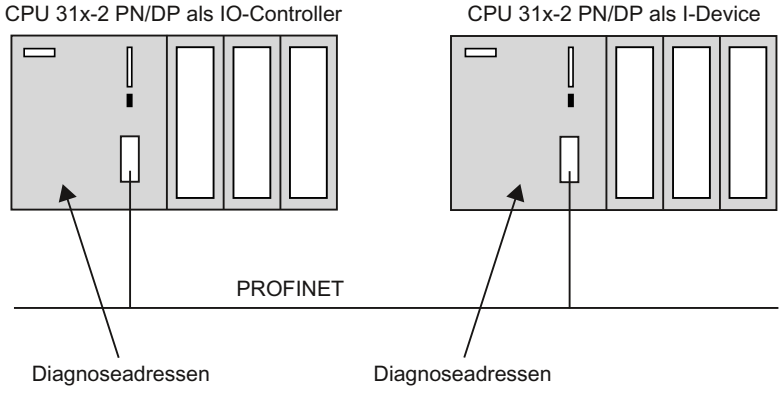

Diagnoseadressen

Bei den Diagnoseadressen wird zwischen Diagnoseadressen im IO-Controller und im I-Device unterschieden:

- Diagnoseadressen im IO-Controller
  Über die dem IO-Controller zugeordnete Diagnoseadressen erhält der IO-Controller Auskunft über den Zustand des IO-Device bzw. über eine Busunterbrechung.
- Diagnoseadressen im I-Device
  Über die dem I-Device zugeordnete Diagnoseadressen erhält das I-Device Auskunft über den Zustand des IO-Controllers bzw. über eine Busunterbrechung.

Beim Konfigurieren der Controller-I-Device-Konfiguration vergibt STEP 7 automatisch Diagnoseadressen für die jeweiligen CPUs. Die folgende Beschreibung zeigt, wie Sie die vorgegebenen Diagnoseadressen ändern.

Tipp: Um eine Übersicht aller für die CPU vergebenen Adressen einschließlich Diagnoseadressen zu erhalten, wählen Sie im Inspektorfenster der markierten CPU den Bereich "Adressübersicht".

Diagnoseadressen der I-Device-Kommunikation

Am I-Device wird eine Tabelle mit den Diagnoseadressen der I-Device-Kommunikation angezeigt. Als Bezeichner für die Diagnoseadresse wird der Name der PLC und der Transferbereich mit der niedrigsten Subslot-Nummer verwendet. Wenn Sie am I-Device bei der Betriebsart der PROFINET-Schnittstelle die Parametrierung der PN-Schnittstelle durch den übergeordneten IO-Controller aktivieren, werden auch die Diagnoseadressen für die PROFINET-Schnittstelle und deren Ports angezeigt. Die verwendeten Diagnoseadressen können Sie in der Tabelle bearbeiten.

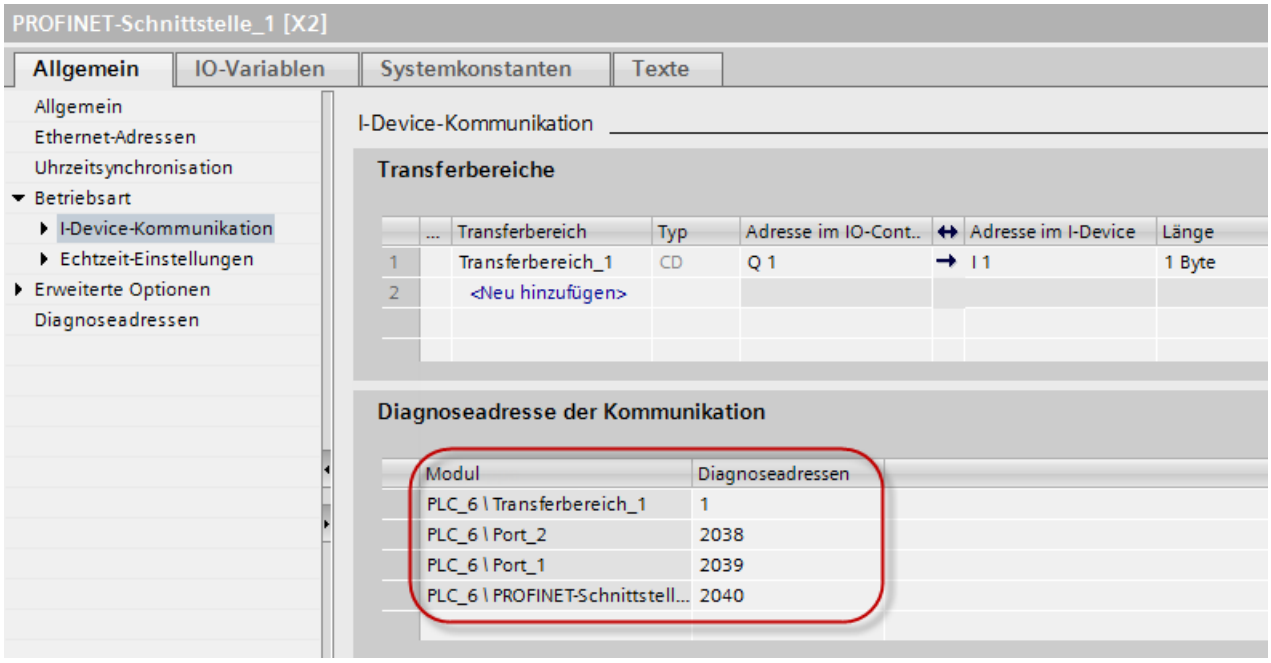

Die Tabelle wird nur angezeigt, wenn der IO-Controller eine CPU 300/400 ist:

IO-Controller	Anzeige der Diagnoseadressen
CPU 300/400	lokale Adressen des I-Devices
CPU 1200/1500	keine Anzeige

Voraussetzung

- Sie befinden sich in der Netzsicht.
- Sie haben ein IO-System mit IO-Controller und I-Device angelegt.

Vorgehen

Um die Diagnoseadressen für die I-Device-Kommunikation zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den IO-Controller oder das I-Device.
2. Wählen Sie in der E/A-Kommunikationstabelle die Zeile mit der Betriebsart "I-Device".
3. Bearbeiten Sie die Diagnoseadressen im Inspektorfenster unter "Eigenschaften > I-Device-Kommunikation > Diagnoseadresse der Kommunikation".

Oder:

1. Selektieren Sie das I-Device.
2. Bearbeiten Sie die Diagnoseadressen im Inspektorfenster unter "Eigenschaften > PROFINET-Schnittstelle > Betriebsart > I-Device-Kommunikation > Diagnoseadresse der Kommunikation".

Sie können die Felder mit den Werten der Diagnoseadressen selektieren und bearbeiten, und zwar sowohl für das Device als auch für den Controller.

Weitere Informationen

Beachten Sie für weitere Informationen zu Diagnoseadressen die Handbücher zu den einzelnen S7-CPUs.

I-Device projektieren

Einleitung

Grundsätzlich haben Sie zwei Möglichkeiten der Projektierung:

- Konfigurieren eines I-Devices innerhalbe eines Projekts
- Konfigurieren eines I-Devices, das in einem anderen Projekt verwendet wird oder das in einem anderen Engineering-System verwendet wird.

Wenn Sie ein I-Device für ein anderes Projekt bzw. für ein anderes Engineering System konfigurieren, dann ermöglicht Ihnen STEP 7 dies über den Export eines konfigurierten I-Devices in eine GSD-Datei. Die GSD-Datei importieren Sie im anderen Projekt bzw. im anderen Engineering System wie andere GSD-Dateien. In dieser GSD-Datei sind u. A. die Transferbereiche für den Datenaustausch hinterlegt.

Konfigurieren eines I-Devices innerhalb eines Projekts

1. Ziehen Sie aus dem Hardware-Katalog einen PROFINET IO-Controller per Drag & Drop in die Netzsicht.
2. Ziehen Sie aus dem Hardware-Katalog einen PROFINET IO-Controller, der auch als IO-Device parametriert werden kann, per Drag & Drop in die Netzsicht. Dieses Gerät wird als I-Device parametriert (z. B. CPU 319-3 PN/DP, FW 3.2).
3. Selektieren Sie die PROFINET-Schnittstelle des I-Devices.

4. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Betriebsart" und aktivieren Sie das Optionskästchen "IO-Device".
5. In der Klappliste "Zugewiesener IO-Controller" haben Sie jetzt die Möglichkeit, den IO-Controller auszuwählen.
   Wenn Sie den IO-Controller ausgewählt haben, wird anschließend die Vernetzung und das IO-System zwischen beiden Geräten in der Netzsicht angezeigt.
6. Mit dem Optionskästchen "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller" bestimmen Sie, ob die Schnittstelle und deren Ports vom I-Device selbst parametriert wird oder vom übergeordneten IO-Controller.
   Wenn Sie das I-Device mit einem untergeordneten IO-System betreiben, dann kann die PROFINET-Schnittstelle (z. B. Port-Parameter) des I-Devices nicht durch den übergeordneten IO-Controller parametriert werden.
7. Projektieren Sie die Transferbereiche. Das sind die Peripheriebereiche, über die das I-Device mit dem übergeordneten IO-Controller Daten austauscht. Die Transferbereiche finden Sie in der Bereichsnavigation im Abschnitt "I-Device-Kommunikation".
   - Klicken Sie in das erste Feld der Spalte "Transferbereiche". STEP 7 vergibt einen voreingestellten Namen, den Sie ändern können.
   - Wählen Sie den Typ der Kommunikationsbeziehung, beispielsweise CD für "Controller-Device-Kommunikationsbeziehung".
   - Adressen werden automatisch vorbelegt; korrigieren Sie die Adressen, wenn erforderlich und legen Sie die Länge der Transferbereichs fest, der konsistent übertragen werden soll.
8. Für jeden Transferbereich wird in der Bereichsnavigation ein einzelner Eintrag erzeugt. Wenn Sie einen dieser Einträge auswählen, können Sie die Details des Transferbereichs anpassen bzw. korrigieren und kommentieren.
   Wenn die CPU das Feature unterstützt, könne Sie auch die Option "Bidirektionales Adressmapping" aktivieren. In diesem Fall werden automatisch Adressbereiche "in Gegenrichtung" eingerichtet, so dass Eingangs- und Ausgangsbereiche von den CPUs jeweils symmetrisch belegt werden.

Konfigurieren eines I-Devices mit Hilfe einer GSD-Datei

Wenn Sie das I-Device in einem anderen Projekt verwenden oder wenn das I-Device in einem anderen Engineering System verwendet wird, dann projektieren Sie den übergeordneten IO-Controller und das I-Device grundsätzlich wie oben beschrieben. Das I-Device muss danach als GSD-Datei exportiert werden. Die so erzeugte GSD-Datei ist der Stellvertreter für das projektierte I-Device in einem anderen Projekt.

Beachten Sie dazu die Beschreibung zum Export von GSD-Dateien (Seite 3078).

Hinweis

PROFIsafe-Transfermodul vom Typ "F-PS"

Wenn Sie ein PROFIsafe-Transfermodul vom Typ "F-PS" gesteckt haben, darf das I-Device keinem IO-Controller in diesem Projekt zugewiesen werden. Es muss eine GSD-Datei exportiert werden. Mit dieser GSD-Datei können im anderen Projekt alle PROFIsafe-Parameter eingestellt werden und das I-Device an einem PROFIsafe-fähigen IO-Controller betrieben werden.

Siehe auch

I-Device als Shared Device konfigurieren (Seite 3124)

Mehrfachzuweisung von IO-Controllern (Seite 3080)

I-Device als GSD-Datei exportieren

Einleitung

Wenn Sie ein I-Device in einem anderen Projekt verwenden möchten oder wenn das I-Device in einem anderen Engineering System eingesetzt werden soll, müssen Sie das I-Device als GSD-Datei exportieren. Die GSD-Datei enthält die Beschreibung des I-Devices und kann anschließend in das neue Projekt importiert und verwendet werden.

Voraussetzung

- Ein I-Device mit I-Device-Kommunikation und Transferbereich wurde projektiert.
- Die Änderungen an der Hardware wurden übersetzt.

Vorgehen

Um ein I-Device als GSD-Datei zu exportieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie das I-Device oder dessen PROFINET-Schnittstelle im Hardware- und Netzwerkeditor.
2. Wählen Sie im Inspektorfenster an der PROFINET-Schnittstelle "Betriebsart > I-Device-Kommunikation" den Befehl "Export". Es öffnet sich der Dialog für den Export des I-Devices als GSD-Datei.

3. Sie können nun den Namen und eine Beschreibung für den I-Device-Stellvertreter festlegen und den Pfad für die Ablage der GSD-Datei verändern.

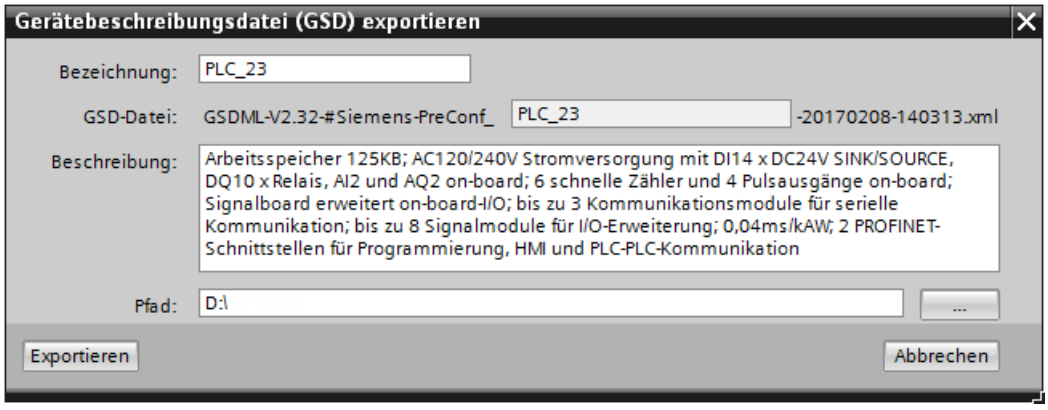

Bezeichnung: Die Bezeichnung steht für einen Text, der als Namensbestandteil der GSD-Datei verwendet wird. Mit dieser Bezeichnung können Sie ihre GSD-Datei später leichter identifizieren. Voreingestellt ist der Name der PLC aus dem aktuellen Projekt.

Hinweis

Bei einem späteren Import der GSD-Datei in ein anderes Projekt wird jedes importierte I-Device mit einer neuen Bezeichnung als neuer Eintrag im Hardware-Katalog erstellt. Wenn die Bezeichnung im Hardware-Katalog bereits existiert, wird das I-Device lediglich als weitere Version eines schon vorhandenen I-Devices hinzugefügt.

Beschreibung: Hier können Sie die Beschreibung editieren. Dieser Text wird nach dem Import der GSD-Datei in ein anderes Projekt im Hardware-Katalog bei dem I-Device im Feld "Beschreibung" angezeigt. Voreingestellt ist die Beschreibung der PLC aus dem aktuellen Projekt.
Pfad: Hier können Sie eintragen, wo die GSD-Datei gespeichert werden soll. Wählen Sie einen Pfad, den Sie leicht wiederfinden können.

4. Klicken Sie auf die Schaltfläche "Exportieren". Das I-Device wird nun als GSD-Datei exportiert.

Sie haben nun das I-Device als GSD-Datei exportiert und können es in einem anderen Projekt verwenden.

Zusammensetzung des Namens der exportierten GSD-Datei

Der Name der GSD-Datei ist nicht frei wählbar, da er einem festgelegten Schema folgen muss. Sie können nur den Teil des Namens beeinflussen, unter dem Sie nach einem Import der GSD-Datei das I-Device im Hardware-Katalog finden können. Der Aufbau des Namens für die exportierte GSD-Datei wird am folgenden Beispiel erläutert:

Der Name einer exportierten GSD-Datei für I-Devices kann beispielsweise folgendermaßen aussehen:

"GSDML-V2.32-#Siemens-PreConf_PLC_23-20170208-140313.xml"

Namensbestandteil	Editierbar?	Erläuterung
GSDML	Nein	Zeichenfolge, mit der jede GSD-Datei für I-Devices beginnt
V2.32	Nein	Version des GSDML-Schemas
Siemens	Nein	Hersteller
PreConf	Nein	Kennzeichnung für GSD-Datei aus der Exportfunktion
PLC_23	Ja	Änderbare Bezeichnung des Geräts. Unter diesem Namen kann das I-Device nach einem Import im Hardware-Katalog des anderen Projektes gefunden werden.
20170208	Nein	Kennzeichnung des Ausgabestands als Datum im Format Jahr-Monat-Tag (YYYYMMDD). Als Referenz wird das Systemdatum verwendet.
140313	Nein	Kennzeichnung des Ausgabestands als Uhrzeit im Format Stunde-Minute-Sekunde (hhmmss). Als Referenz wird die UTC-Systemzeit verwendet.
.xml	Nein	Dateiendung

Import der exportierten GSD-Datei in einem anderen Projekt

Importieren Sie die GSD-Datei mit dem exportierten I-Device, indem Sie die GSD-Datei installieren, wie an anderer Stelle beschrieben. Nachdem Sie unter "Extras > "Gerätebeschreibungsdateien (GSD) verwalten > Installierte GSDs" die GSD-Datei mit dem I-Slave ausgewählt und installiert haben, finden Sie das I-Device unter "Weitere Feldgeräte > PROFINET IO" im Hardware-Katalog unter dem Namen, den Sie als Bezeichnung im Exportdialog angegeben haben. I-Devices mit gleichem Namen werden als Versionen des selben Geräts im Hardware-Katalog eingestellt. Sie können diese unterschiedlichen Versionen über die Kennzeichnung des Ausgabestandes im Dateinamen und ggf. über eine unterschiedliche Beschreibung identifizieren. Zur Anzeige dieser Versionen nutzen Sie nach Auswahl des I-Devices die Klappliste "Version".

Ausgewählte I-Devices können wie andere Geräte des Hardware-Kataloges per Drag & Drop in die Konfiguration gezogen werden.

GSD-Dateien werden immer mit dem Projekt zusammen gespeichert, d. h. alle zur Darstellung des Gerätes relevanten Informationen (auch die Symbole) sind auch im gespeicherten Projekt verfügbar.

Siehe auch

I-Device projektieren (Seite 3076)

Mehrfachzuweisung von IO-Controllern

Einleitung

Wenn in einer S7-1500 PLC mehrere PROFINET IO-Schnittstellen vorhanden sind mit I-Device-Funktionalität, dann können Sie diesem Gerät auch mehrere IO-Controller zuweisen.

Ab STEP 7 V14 ist auch sichtbar, dass es mehr als einen übergeordneten IO-Controller für ein I-Device gibt: In der Netzsicht erscheint an solchen Geräten der Link "Mehrfachzuweisung".

Beispiele für PROFINET IO-Schnittstellen mit I-Device-Funktionalität:

- S7-1500 CPU, insbesondere solche mit zwei PROFINET IO-Schnittstellen (z. B. CPU 1518-4 PN/DP, CPU 1517-3 PN/DP unterstützen PROFINET IO (und I-Device-Funktionalität) an den Schnittstellen X1 und X2)
- CM 1542-1

Im Folgenden sind die Auswirkungen einer Mehrfachzuweisung beschrieben.

Konfigurieren eines Gerätes mit Zuordnungen zu mehreren IO-Controllern

Im Folgenden wird an einer Beispielkonfiguration gezeigt, wie sich die Zuweisung eines I-Devices zu mehreren IO-Controllern auswirkt.

1. Ziehen Sie aus dem Hardware-Katalog mehrere PROFINET IO-Controller (z. B. CPU 1518-4 PN/DP) per Drag & Drop in die Netzsicht.
2. Ziehen Sie aus dem Hardware-Katalog einen PROFINET IO-Controller, der auch als IO-Device parametriert werden kann, per Drag & Drop in die Netzsicht (z. B. CPU 1518-4 PN/DP). Sie können auch einen CM 1542-1 in diese Station einfügen und die PROFINET-Schnittstelle ebenfalls als IO-Device parametrieren.
3. Selektieren Sie die PROFINET-Schnittstelle einer Baugruppe.
4. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Betriebsart" und aktivieren Sie das Optionskästchen "IO-Device".
5. In der Klappliste "Zugewiesener IO-Controller" haben Sie jetzt die Möglichkeit, den IO-Controller auszuwählen.
6. Wiederholen Sie Schritt 3 und 4 für weitere PROFINET-Schnittstellen.
   Wenn Sie den IO-Controller bzw. die PROFINET IO-Schnittstelle des IO-Controllers ausgewählt haben, wird anschließend die Vernetzung und das IO-System zwischen beiden Geräten in der Netzsicht angezeigt.
   Sobald Sie eine weitere PROFINET IO-Schnittstelle einem anderen IO-Controller zuweisen, erscheint im Gerät der Link "Mehrfachzuweisung" statt des Namens des übergeordneten IO-Controllers.
   Wenn Sie auf diesen Link klicken, erscheint die Auswahl zugewiesener IO-Controller und Sie können durch Klick auf einen dieser IO-Controller den Fokus auf dieses Gerät setzen.

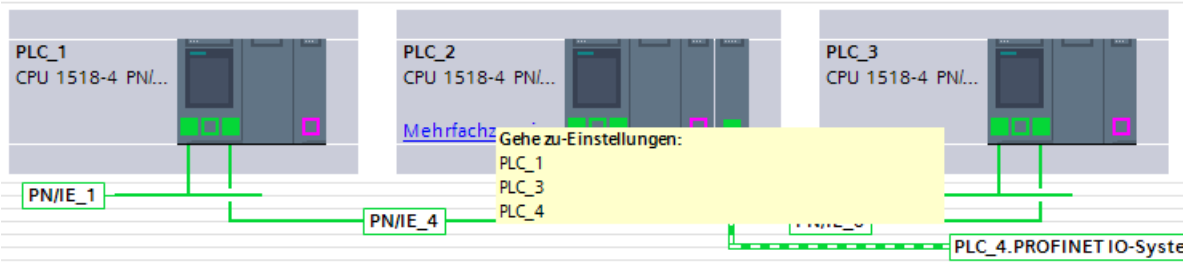

Die übrigen Einstellungen (weitere Parametrierung der IO-Schnittstelle, Projektieren der Transferbereiche) sind von dieser Funktionserweiterung nur insofern betroffen, als dass die Auswahl der Geräte bzw. IO-Systeme auch bei CPUs mit mehreren PROFINET IO-Schnittstellen erleichtert wird durch eine eindeutige Kennzeichnung.

Wenn Sie die Schnittstellen genau einem S7-Subnetz zugewiesen haben, dann existieren auf diesem Subnetz mehrere IO-Systeme.

Wenn Sie auf das Subnetz klicken, erscheint die Anzeige zur Auswahl des hervorzuhebenden IO-Systems.

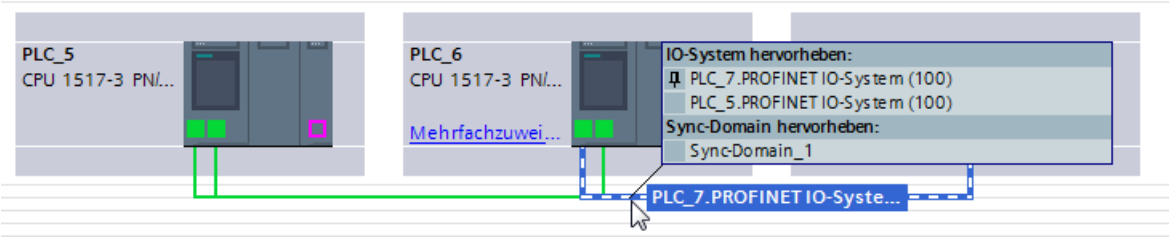

Randbedingungen beim Einsatz von I-Devices

Randbedingungen

Beim Einsatz von I-Devices sind in speziellen Fällen einige Randbedingungen zu beachten.

Bandbreite

Der Adressumfang der projektierten Transferbereiche schlägt sich auf die nutzbare Bandbreite des I-Devices nieder:

Bandbreite der Transferbereiche + Bandbreite des untergeordneten IO-Systems = gesamte am I-Device genutzte Bandbreite.

Wenn der Adressraum der Transferbereiche zu groß ist, dann bleibt für das untergeordnete IO-System nicht mehr genug Bandbreite übrig, um schnelle Aktualisierungszeiten erreichen zu können.

Tipp: Halten Sie den Adressraum der Transferbereiche so klein wie möglich.

Regeln für die RT- und IRT-Kommunikation

IO-Systeme mit I-Devices eignen sich auch für den Aufbau von Echtzeitanwendungen mit RT-Kommunikation. Im Folgenden ist ein hierarchisches System vorausgesetzt, bestehend aus einem übergeordneten IO-Controller, dem ein I-Device zugeordnet ist (übergeordnetes IO-System). Dem I-Device sind wiederum IO-Devices zugeordnet (untergeordnetes IO-System). Es gilt folgende Regel:

Sowohl das übergeordnete als auch das untergeordnete IO-System unterstützt RT-Kommunikation. Sie können zur gleichen Zeit für beide IO-Systeme RT-Kommunikation nutzen.

IRT-Kommunikation kann mit RT-Kommunikation kombiniert werden. IRT-Kommunikation kann jedoch nicht in beiden IO-Systemen zur gleichen Zeit stattfinden.

Beispiel:

- Wenn am I-Device die Option "Parametrierung der PN-Schnittstelle und deren Ports am übergeordneten IO-Controller" **aktiviert** ist, dann kann das I-Device am übergeordneten IO-Controller mit der RT-Klasse "IRT" betrieben werden. Für das untergeordneten IO-System des I-Devices ist in diesem Fall keine IRT-Kommunikation möglich.
- Wenn am I-Device die Option "Parametrierung der PN-Schnittstelle und deren Ports am übergeordneten IO-Controller" **deaktiviert** ist, dann muss das I-Device am übergeordneten IO-Controller mit der RT-Klasse "RT" betrieben werden. Für das untergeordnete IO-System ist IRT-Kommunikation möglich.

Wenn Sie das I-Device an einem IO-System mit der RT-Klasse "RT" projektieren, sind keine ungeraden Sendetakte einstellbar.

Regeln für den Datenzugriff

- Der übergeordnete IO-Controller kann auf Transferbereiche zugreifen, wenn das I-Device in RUN ist.
- In der I-Device CPU darf nur auf Applikationstransferbereiche zugegriffen werden:
  - Zugriff auf Eingangs-Applikationstransferbereiche möglich, wenn der übergeordnete IO-Controller in RUN ist.
  - Zugriff auf Ausgangs-Applikationstransferbereiche möglich, unabhängig vom Betriebszustand des übergeordneten IO-Controllers.

IP-Adressparameter und Gerätename

Wie jedes andere IO-Device benötigt auch ein I-Device IP-Adressparameter / Gerätenamen um via PROFINET kommunizieren zu können. IP-Adressparameter bestehen aus drei Teilen; der IP-Adresse selbst, der Subnetzmaske und der Adresse des Routers (Gateway).

Die IP-Adressparameter / Gerätenamen für ein I-Device können auf zwei Arten vergeben werden; über das Projekt oder direkt am Gerät.

IP-Adresse / Gerätename im Projekt einstellen:

- Die IP-Adressparameter / Gerätename werden bereits bei der Projektierung (im I-Device Projekt) in STEP 7 fest vergeben. Dies ist die Standardmethode.

IP-Adresse / Gerätename am Gerät einstellen:

- IP-Adressparameter / Gerätename via DCP: Die IP-Adressparameter / Gerätename werden per DCP (Discovery and Configuration Protocol) zugewiesen. Dies kann auf zwei Arten erfolgen:
  - Durch ein Setup-Tool wie PST oder STEP 7 (via erreichbare Teilnehmer).
  - Durch den übergeordneten IO-Controller.
- IP-Adressparameter / Gerätename via Anwenderprogramm: Die Vergabe der IP-Adressparameter erfolgt im Anwenderprogramm der I-Device CPU (über die Anweisung "IP_CONFIG").

Hinweis

Topologiedarstellung

Wenn für ein I-Device der Gerätename und die IP-Adressparameter direkt am Gerät angepasst werden, dann kann das I-Device im Topologieeditor mehrmals angezeigt werden.

Diagnose und Alarmverhalten

Diagnose und Alarmverhalten

S7-CPUs verfügen über vielfältige Diagnose- und Alarmfunktionen, die beispielsweise Fehler oder Ausfälle von untergeordneten IO-Systemen melden können. Diese Diagnosemeldungen verringern Ausfallzeiten und erleichtern die Lokalisierung und Behebung von Fehlern.

Diagnose des I-Devices im übergeordneten IO-Controller und im I-Device

Dem übergeordneten IO-Controller und der I-Device-CPU stehen folgende Mechanismen zur Verfügung, um den Zustand des jeweiligen Partners zu diagnostizieren:

- Ziehen/Stecken-OB (OB 83) für "Return-of-Submodul"-Alarm
- Programmablauffehler-OB (OB 85) für z. B. Prozessabbildtransferfehler (nur S7-300/400)
- Baugruppenträgerausfall-OB (OB 86) für Ausfall von Erweiterungsgeräten, IO-Systemen oder Dezentralen Peripheriegeräten (Stationsausfall, Stationswiederkehr)
- Peripheriezugriffsfehler-OB (OB 122), optional bei S7-1500
- Lokale Fehlerbehandlung (nur S7-1500)

Hinweis

Die Diagnosemeldungen der Peripherie können im Anwenderprogramm der I-Device CPU verarbeitet werden und von dort über Transferbereiche an den übergeordneten IO-Controller weiter gegeben werden.

Betriebszustandswechsel und Stationsausfall / Stationswiederkehr

In der folgenden Tabelle sehen Sie, welche Folgen der Betriebszustandswechel oder der Ausfall einzelner CPUs (I-Device, IO-Controller) auf die jeweils Andere(n) hat.

Beachten Sie, dass eine S7-1500-CPU bei Fehlern durch direkten Peripheriezugriff auf die Transferbereiche (z. B. weil die andere CPU in STOP geht) den OB 122 aufruft, wenn er vorhanden ist. Die CPU geht aber im Gegensatz zu einer S7-300/400-CPU nicht in STOP, wenn der OB 122 nicht vorhanden ist. Sie haben außerdem bei einer S7-1500-CPU die Möglichkeit der lokalen Fehlerbehandlung innerhalb des Bausteins, in dem der Fehler aufgetreten ist. Dieser Sachverhalt in der Tabelle verkürzt wiedergegeben durch die Formulierung "Aufruf von OB 122".

Ereignisse

Initialzustand	Ereignis	Verhalten I-Device	Verhalten übergeordneter Controller
Beide CPUs sind in RUN(übergeordneter IO-Controller und I-Device).	Die I-Device CPU geht in STOP	/	**S7-300/400/1500:** Fehlercode in RET_VAL bei Aktualisierung des Prozessabbildes mit Anweisung UPDAT_PI bzw. UPDAT_PO. Bei direktem Zugriff auf Transferbereiche zum I-Device komm es zum Aufruf von OB 122 (optional bei S7-1500). **S7-300/400**: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer zum I-Device zum Aufruf von OB 85.
I-Device-CPU ist in STOP, übergeordneter IO-Controller ist in RUN.	Die I-Device CPU läuft an	**S7-300/400/1500:** Aufruf von OB 100 (Anlauf). Aufruf von OB 83 für Eingangstransferbereiche zum übergeordneten IO-Controller. Bis zum Aufruf von OB 83 kommt es bei direktem Zugriff auf die Eingangstransferbereiche zum übergeordneten IO-Controller zu Zugriffsfehlern: Aufruf von OB 122. **S7-300/400:** Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bis zum Aufruf von OB 83 weiterhin bei systemseitigem Prozessabbildtransfer zum übergeordneten IO-Controller zum Aufruf von OB 85.	**S7-300/400/1500:** Aufruf von OB 83: für alle Transferbereiche zum I-Device. Bis zum Aufruf des OB 83 kommt es bei direktem Zugriff auf die Transferbereiche zum I-Device weiterhin zum Aufruf von OB 122. **S7-300/400:** Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bis zum Aufruf von OB 83 weiterhin bei systemseitigem Prozessabbildtransfer zum I-Device zum Aufruf von OB 85.

Initialzustand	Ereignis	Verhalten I-Device	Verhalten übergeordneter Controller
Beide CPUsind in RUN (übergeordneter IO-Controller und I-Device).	Der übergeordnete IO-Controller geht in STOP	**S7-300/400/1500:** Fehlercode in RET_VAL bei Aktualisierung des Prozessabbildes mit Anweisung UPDAT_PI bzw. UPDAT_PO. Bei direktem Zugriff auf Eingangstransferbereiche zum übergeordneten IO-Controller kommt es zum Aufruf von OB 122. Hinweis: Auf Ausgangs-transferbereiche kann weiterhin zugegriffen werden. **S7-300/400**: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Eingangstransferbereiche zum übergeordneten IO-Controller zum Aufruf von OB 85.	/
Übergeordneter IO-Controller ist in STOP, I-Device-CPU in RUN.	Der übergeordnete IO-Controller läuft an	**S7-300/400/1500**: Aufruf von OB 83 für Eingangstransferbereiche zum übergeordneten IO-Controller. Bis zum Aufruf von OB 83 kommt es bei direktem Zugriff auf die Eingangstransferbereiche zum übergeordneten IO-Controller weiterhin zum Aufruf von OB 122. **S7-300/400**: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Eingangstransferbereiche zum übergeordneten IO-Controller zum Aufruf von OB 85.	Aufruf von OB 100 (Anlauf).

Initialzustand	Ereignis	Verhalten I-Device	Verhalten übergeordneter Controller
Beide CPUs sind in RUN (übergeordneter IO-Controller und I-Device).	Stationsausfall I-Device, z.B. durch Busunterbrechung	S7-300/400/1500: Falls das I-Device ohne Busanbindung weiterläuft: Aufruf von OB 86 (Stationsausfall). Fehercode in RET_VAL bei Aktualisierung des Prozessabbildes mit Anweisung UPDAT_PI bzw. UPDAT_PO. Bei direktem Zugriff auf Transferbereiche zum übergeordneten IO-Controller kommt es zum Aufruf von OB 122 S7-300/400: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Transferbereiche zum übergeordneten IO-Controller zum Aufruf von OB 85.	S7-300/400/1500: Aufruf von OB 86 (Stationsausfall). Fehlercode in RET_VAL bei Aktualisierung des Prozessabbildes mit Anweisung UPDAT_PI bzw. UPDAT_PO. Bei direktem Zugriff auf Transferbereiche zum I-Device komm es zum Aufruf von OB 122. S7-300/400: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Transferbereiche zum I-Device zum Aufruf von OB 85.
Beide CPUs in RUN (übergeordneter IO-Controller und I-Device). Die Kommunikationsverbindung zwischen IO-Controller und I-Device ist unterbrochen (Busunterbrechung).	Die Busverbindung zwischen IO-Controller und I-Device ist wieder hergestellt und das I-Device wird vom IO-Controller wieder in den Nutzdatenverkehr aufgenommen.	S7-300/400/1500: Beachten Sie den Hinweis unter dieser Tabelle zu den Besonderheiten beim Hochlaufen des I-Devices! Aufruf von OB 86 (Stationswiederkehr). Aufruf von OB 83 für die Eingangstransferbereiche zum übergeordneten IO-Controller. Bis zum Aufruf von OB 83 kommt es bei direktem Zugriff auf die Eingangstransferbereiche zum übergeordneten IO-Controller weiterhin zum Aufruf von OB 122. S7-300/400: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Eingangstransferbereiche zum übergeordneten IO-Controller zum Aufruf von OB 85.	S7-300/400/1500: Aufruf des OB 86 (Stationswiederkehr). Bis zum Aufruf von OB 86 kommt es bei direktem Zugriff auf Transferbereiche zum I-Device zum Aufruf von OB 122. S7-300/400: Bei projektiertem OB85-Aufruf bei Peripheriezugriffsfehler kommt es bei systemseitigem Prozessabbildtransfer der Transferbereiche zum I-Device zum Aufruf von OB 85.

Hinweis

Besonderheit beim Hochlauf des I-Devices

Im Gegensatz zur Stationswiederkehrmeldung von IO-Devices im IO-Controller, die durch einen Aufruf des OB 86 komplett abgedeckt wird, ist die Stationswiederkehrmeldung eines übergeordneten IO-Controllers im I-Device in 2 Teile geteilt:

1. Aufruf des OB 86: Die Initialwerte für die Ausgänge des I-Devices werden gesetzt. Die Eingangswerte sind allerdings noch nicht gültig; sie werden erst mit dem Aufruf des OB86 im übergeordneten IO-Controller gültig.
2. Aufruf des OB 83 für jeden Eingangstransferbereich. Mit diesem Aufruf wird die Gültigkeit eines Eingangstransferbereiches angezeigt. Der Hochlauf des I-Devices ist erst abgeschlossen, wenn für die Eingangstransferbereiche der OB 83 aufgerufen wurde. Dieser Schritt kann in folgender Situationen deutlich verzögert oder gar nicht stattfinden:
   - Übergeordneter IO-Controller ist in STOP: OB 83 wird erst bei STOP → RUN-Übergang des übergeordneten IO-Controllers aufgerufen.
   - Die IRT-Kommuniktion ist gestört, z. B. weil der Sync-Master ausgefallen ist, wegen Topologiefehler etc: Der OB 83 wird erst gestartet, wenn die IRT-Kommunikation wieder zustande gekommen ist.

Siehe auch

Ziehen/Stecken-OB (OB 83) (Seite 897)

Baugruppenträgerausfall-OB (OB 86) (Seite 908)

Peripheriezugriffsfehler-OB (OB 122) (Seite 927)

Programmablauffehler-OB (OB 85) (Seite 904)

Online-Diagnose von I-Devices

Die Online-Diagnose von I-Devices unterscheidet sich prinzipiell nicht von der Online-Diagnose von "normalen" IO-Devices.

Anzeige des Diagnosezustands beim Online-Gehen

Abhängig davon, welches Gerät Sie beim Online-Gehen selektiert haben, erhalten Sie folgende Informationen:

- Sie haben den übergeordneten IO-Controller selektiert:
  In der Projektnavigation wird der Diagnosezustand für das I-Device unterhalb des IO-Controllers im Ordner "Dezentrale Peripherie" angezeigt. Für das I-Device selbst wird in der Projektnavigation kein Diagnosezustand angezeigt.
- Sie haben das I-Device, das diesem IO-Controller zugewiesen ist, selektiert.
  In der Projektnavigation wird der Diagnosezustand für das I-Device selbst angezeigt. Für das I-Device unterhalb des IO-Controllers im Ordner "Dezentrale Peripherie" wird kein Diagnosezustand angezeigt.

Online- und Diagnosesicht

Abhängig davon, welches Gerät Sie beim Starten der Online- und Diagnosesicht selektiert haben, erhalten Sie folgende Informationen:

- Sie haben den übergeordneten IO-Controller selektiert und sind online mit diesem Gerät verbunden:
  Die Online- und Diagnosesicht zeigt die Diagnosedaten des I-Devices aus Sicht des übergeordneten IO-Controllers. Die Anzeige entspricht der Anzeige eines "normalen" IO-Controllers.
- Sie haben das I-Device, das diesem IO-Controller zugewiesen ist, selektiert und sind online mit diesem Gerät verbunden:
  Die Online- und Diagnosesicht zeigt die Diagnosedaten des I-Devices aus lokaler Sicht, d. h. aus Sicht des I-Devices. Die Anzeige entspricht der Anzeige eines "normalen" IO-Devices.

Anzeige in der Liste der erreichbaren Teilnehmer

In der Liste der erreichbaren Teilnehmer wird das I-Device wie eine "normale" CPU angezeigt.

Im Betrieb wechselnde IO-Devices

Docking Station - Im Betrieb wechselnde IO-Devices

Anwendung von im Betrieb wechselnden IO-Devices ("Wechselnder Partnerport") in einer Docking-Station

Im folgenden Bild ist eine Automatisierungszelle mit einem Dockingsystem und mehreren Docking Units dargestellt.

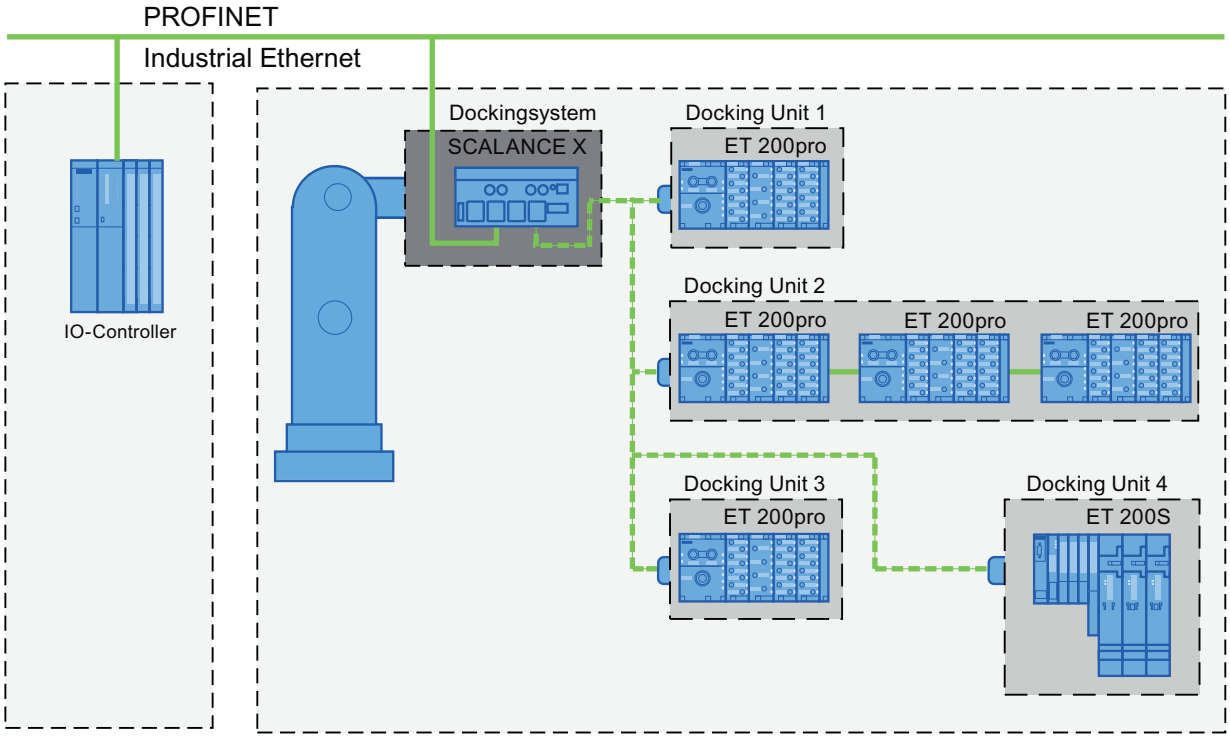

Bild 1-92 Wechselnde IO-Devices (Partnerports) in einer Docking-Station

Applikative Voraussetzungen

Folgende Punkte sind bei der Realisierung eines Dockingsystems mit im Betrieb wechselnden IO-Devices zu beachten:

- Die IO-Devices aller Docking Units sind in der Projektierung in der Voreinstellung deaktiviert.
- Es kann zu jeder Zeit immer nur eine Docking Unit aktiv sein, d. h. nur die IO-Devices einer Docking Unit können aktiviert sein. Alle IO-Devices anderer Docking Units müssen deaktiviert sein oder deaktiviert werden, bevor die IO-Devices einer Docking Unit aktiviert werden können. Dies geschieht mit der Anweisung "D_ACT_DP".

- Zur Aktivierung einer Docking Unit wird eine physikalische Verbindung zu dieser Docking Unit und dessen IO-Devices hergestellt und danach werden die IO-Devices eingeschaltet (Power-On). Zeitgleich müssen im Anwenderprogramm alle IO-Devices dieser Docking Unit per Anweisung "D_ACT_DP" aktiviert werden.
- Nach der Rückmeldung "IO-Device aktiviert" greifen Sie mit dem Befehl Direkter Peripheriezugriff auf das IO-Device zu.
- Rufen Sie der Anweisung "D_ACT_DP" zum Aktivieren und Deaktivieren des IO-Devices möglichst am Anfang des OB 1-Zyklus auf.

Einsatzgebiet von im Betrieb wechselnden IO-Devices

Die PROFINET-Funktionalität "Im Betrieb wechselnde IO-Devices" ("Wechselnder Partnerport") können Sie z. B. für den Werkzeugwechsel bei Robotern nutzen. Typische Werkzeuge sind z. B. wie folgt:

- Schweißzangen und
- Haltewerkzeuge für Fertigungsteile.

Hinweis

Anzahl der im Betrieb wechselnden IO-Devices ("Wechselnder Partnerport") - Anzahl der Docking Units

Wenn Sie kürzest mögliche Werkzeugwechselzeiten erreichen möchten, müssen Sie folgende Punkte, die von der eingesetzten CPU oder dem eingesetzten CP abhängig sind, beachten:

- Nur mit der PROFINET-Funktionalität "Priorisierter Hochlauf" projektierte IO-Devices können optimiert anlaufen. Die Anzahl der IO-Devices mit Projektierung dieser PROFINET-Funktionalität ist beschränkt.
- Es kann nur eine bestimmte Anzahl von IO-Devices gleichzeitig aktiviert werden (abhängig von den verfügbaren Ressourcen der Anweisung "D_ACT_DP"), so dass dann eine Docking Unit auch nicht mehr als die entsprechende Anzahl IO-Devices enthalten sollte. Werden hier mehr IO-Devices in einer Docking Unit betrieben, so müssen die IO-Devices nacheinander aktiviert werden, was entsprechend länger dauert.

Beispiel: Eine S7-CPU 319-3 PN/DP kann maximal 32 IO-Devices mit priorisiertem Hochlauf bedienen und kann gleichzeitig 8 IO-Devices per Anweisung "D_ACT_DP"aktivieren.

Für eine zeitlich optimale Anwendung sollte deshalb eine Docking Unit nicht mehr als 8 IO-Devices beinhalten und in allen wechselnden Docking Units sollten nicht mehr als 32 IO-Devices eingesetzt sein.

Einschränkung bei der Verschaltung

Die Verschaltung mit einem Partnerport ist in folgenden Fällen nicht möglich:

- Der Partnerport hat einen nicht passenden Leitungstyp. In diesem Fall muss ein Medienkonverter aus dem Katalog eingefügt werden.
- Der Partnerport ist gesperrt (deaktiviert).

- Die beiden zu verschaltenden Ports gehören zum gleichen Interface (nur die Verschaltung der Ports von unterschiedlichen Interfaces in einer Station ist möglich).
- Sie versuchen eine Ringverbindung mit einer Baugruppe zu erstellen, die nicht redundanzfähig ist.
- Die beiden zu verschaltenden Ports gehören zu unterschiedlichen Ethernet-Subnetzen.
- Der Port einer PROFINET-Schnittstelle eines IO-Controllers kann nicht direkt mit der Funktionalität Im Betrieb wechselnde IO-Devices ("Wechselnder Partnerport") projektiert werden.

Docking Units an einem Docking System projektieren

Docking Units an einem Docking System projektieren

Um mehrere Docking Units an einem Docking System zu projektieren, gehen Sie wie folgt vor:

1. Projektieren Sie Ihre Hardware-Konfiguration wie üblich; betrachten Sie die IO-Devices, die den wechselnden Partnern entsprechen, zunächst als normale IO-Devices.
2. Öffnen Sie im Inspektorfenster die Eigenschaften des Ports, an dem Sie wechselnde IO-Devices (wechselnde Partnerports) betreiben möchten.
3. Aktivieren Sie unter "Allgemein > PROFINET-Schnittstelle > Erweiterte Optionen > RJ45 100 Mbit Port [Xn Pm] > Portverschaltung" das Optionskästchen "Alternative Partner".
4. Fügen Sie die alternativen Partner in die Tabelle ein, wie im Kapitel "Ports verschalten" beschrieben.

Siehe auch

Ports verschalten bei wechselnden Partnerports (Seite 3092)

Hochlaufzeiten optimieren (Seite 3094)

Ports verschalten bei wechselnden Partnerports

Voraussetzung

- Alle beteiligten Geräte unterstützen Topologie-Projektierung
- Der IO-Controller, die wechselnden IO-Devices (Docking Unit) und der Switch (Dockingsystem), an dem die wechselnden IO-Devices betrieben werden, müssen dieses Feature unterstützen.
- Die Docking Unit muss mit einem Switch, der die PROFINET-Funktion "Priorisierter Hochlauf" unterstützt (z. B. aus der SCALANCE X200IRT-Familie), verbunden sein.

Beispiel

- IO-Controller: CPU 416-3 PN/DP, CPU 1516-3 PN/DP
- Dockingsystem: Scalance X-200IRT
- Docking Units (wechselnde IO-Devices): z. B. 2 x ET 200pro (IM 154-4)

Vorgehen

Bei der Beschreibung des Vorgehens wird die Konfiguration aus dem oben genannten Beispiel vorausgesetzt. Um Ports miteinander zu verschalten, gehen Sie folgendermaßen vor:

1. Verbinden Sie alle PROFINET-Schnittstellen der Geräte untereinander in der Netzsicht.
2. Selektieren Sie in der Netz- oder Gerätesicht das Dockingsystem (den Scalance X200 IRT), dessen Port Sie verschalten möchten.
3. Öffnen Sie im Inspekorfenster die Eigenschaften des Geräts.
4. Wählen Sie unter "Eigenschaften > PROFINET-Schnittstelle > Erweiterte Optionen > RJ45 100 Mbit Port [Xn Pm] > Portverschaltung" den Partnerport für die CPU (IO-Controller) in der Klappliste "Partnerport" aus.
   Die Ports der CPU und des Dockingsystems (Scalance) sind miteinander verschaltet.
5. Das Dockingsystem ist weiterhin selektiert: Wählen Sie jetzt für einen beliebigen weiteren unverschalteten Port die Partnerports für die IO-Devices aus:
   - Aktivieren Sie das Optionskästchen "Alternativer Partner".
     Eine Tabelle für die Partnerports wird angezeigt.
   - Klicken Sie innerhalb der Tabelle auf die Zeile "<Alternativen Partner hinzufügen...>, um eine Klappliste zu öffnen.
     In der Klappliste werden alle Geräte mit ihren unverschalteten Ports zur Auswahl angezeigt.
   - Wählen Sie den Port der ersten Docking Unit aus.
   - Klicken Sie auf die nächste Tabellenzeile und wählen Sie einen weiteren alternativen Partnerport aus.
   - Wiederholen Sie die oben genannten Schritte, bis alle erforderlichen Partnerports in der Tabelle der eingetragen sind.

Hinweis

Verwendung eines CP 1243-1 (WAN-CP)

Bei Verwendung einer CPU S7-1200 mit CP 1243-1 wird wie oben beschrieben vorgegangen, aber statt der PROFINET-Schnittstelle des IO-Controllers wird die Ethernet-Schnittstelle des CPs verwendet.

Hinweis

CAx-Export einer PROFINET-Konfiguration für Werkzeugwechsler-Projekte

Wenn Sie eine PROFINET-Konfiguration mit aktiviertem Optionskästchen "Alternativer Partner" haben, dann müssen Sie beim Exportieren von CAx-Daten das Folgende beachten: Die Einstellung "Alternativer Partner" wird beim Export nicht unterstützt. Das hat zur Folge, dass bei einem anschließenden Re-Import diese Einstellung bzw. die topologische Verbindung fehlt.

Verschaltung von mehreren IO-Devices mit im Betrieb wechselnden IO-Devices

IO-Devices auf einer Docking Unit, die in Linie mit im Betrieb wechselnde IO-Devices ("Alternative Partner") verschaltet werden, projektieren Sie mit dem Port des wechselnden IO-Devices wie gewohnt, d. h. das in Linie verschaltete IO-Device hat dann keinen wechselnden Partnerport.

Löschen des Partnerports

Entfernen Sie die Verschaltung des im Betrieb wechselnden Partnerports, indem Sie folgendermaßen vorgehen:

1. Klicken Sie in der Tabelle der alternativen Partner auf die Zeile, in der welcher der zu löschende alternative Partnerport eingetragen ist, und klappen Sie die Auswahlliste für die Partnerports auf (schwarzes Dreieck).
2. Wählen Sie in der Liste den Eintrag "beliebiger Partner".
   Der alternative Partnerport wird aus der Tabelle entfernt.

Siehe auch

Was ist Priorisierter Hochlauf? (Seite 3094)

Hochlaufzeiten optimieren

Was ist Priorisierter Hochlauf?

Definition

Priorisierter Hochlauf, auch "Fast Start-Up" oder "schneller Hochlauf" genannt, bezeichnet die PROFINET-Funktionalität zur Beschleunigung des Anlaufs von IO-Devices in einem PROFINET IO-System mit RT- und IRT-Kommunikation. Sie verkürzt die Zeit, die die entsprechend projektierten IO-Devices benötigen, um in folgenden Fällen wieder in den zyklischen Nutzdatenaustausch zu gelangen:

- Nach Wiederkehr der Spannungsversorgung
- Nach Stationswiederkehr
- Nach Aktivieren von IO-Devices

Vorteile

Die PROFINET-Funktionalität "Priorisierter Hochlauf" ermöglicht PROFINET IO-Anwendungen, in denen Maschinenteile oder Werkzeuge und deren IO-Devices permanent getauscht werden. Wartezeiten von mehreren Sekunden zwischen den zeitlichen Abläufen des Wiederanlaufs sind durch deren Optimierung auf ein Minimum reduziert. Dadurch beschleunigt sich der Fertigungsprozess mit wechselnden IO-Devices, z. B. in Werkzeugwechsler-Anwendungen, und ermöglicht einen größeren Durchsatz in der Produktion.

Auch in Anwendungen, bei denen es generell auf eine schnelle Hochlaufzeit der IO-Devices nach "Power-On" bzw. nach Stationsausfall/Stationswiederkehr ankommt, oder beim Aktivieren von IO-Devices bietet die PROFINET-Funktionalität "Priorisierter Hochlauf" einen erheblichen Performance-Gewinn.

Eigenschaften

Folgende Eigenschaften realisieren Sie mit der PROFINET-Funktionalität "Priorisierter Hochlauf":

- Kommunikationsbereitschaft der IO-Devices (Dezentrale Peripherie) bis zu minimal 500 ms.
- Sie können den priorisierten Hochlauf für IO-Devices (Dezentrale Peripherie) sowohl mit RT- als auch mit IRT-Kommunikation nutzen.

Hochlaufzeiten

Die Länge der Anlaufzeit eines IO-Devices (Dezentrale Peripherie) mit der PROFINET-Funktionalität "Priorisierter Hochlauf" ist von folgenden Punkten abhängig:

- Verwendete IO-Devices (Dezentrale Peripherie)
- Peripherieausbau des IO-Devices (Dezentrale Peripherie)
- Verwendete Module des IO-Devices (Dezentrale Peripherie)
- Verwendeter IO-Controller
- Verwendeter Switch
- Port-Einstellung
- Verkabelung
- Projektierte RT-Klasse des IO-Devices

Hinweis

Priorisierter Hochlauf nach erstmaligem Hochlauf

Ein Priorisierter Hochlauf der IO-Devices steht Ihnen immer erst nach der erstmaligen Parametrierung dieses IO-Devices im allerersten Anlauf des PROFINET IO-Systems zur Verfügung. Auch im Ersatzteilfall bzw. nach Reset-to-factory Settings (Zurücksetzen auf Werkseinstellung) erfolgt der erste Hochlauf als Standard-Hochlauf bei den entsprechend projektierten IO-Devices.

Hinweis

Bei folgenden Situationen kann es trotz Priorisiertem Hochlauf zu Anlaufzeiten von bis zu 8 s kommen:

- Ein IO-Device wird innerhalb von 8 s abgekoppelt und wieder angekoppelt.
- An einer Andockstelle docken als ein IO-Device mit einem bestimmten Gerätenamen und einer bestimmten IP-Konfiguration mehrere physikalische IO-Devices an (z. B. Andockstelle für fahrerloses Transportsystem).

Wenn Sie eine kürzest mögliche Anlaufzeit von 500 ms erreichen wollen, müssen Sie folgende Maßnahmen umsetzen:

- Port-Einstellung am IO-Device
- Verkabelung in Abhängigkeit der verschalteten PROFINET-Geräte
- Maßnahmen im Anwenderprogramm

Priorisierten Hochlauf einstellen

Voraussetzung

Die PROFINET-Funktionalität "Priorisierter Hochlauf" können Sie bei den IO-Devices (Dezentrale Peripherie) nur in folgenden Fällen aktivieren:

- Der verwendete IO-Controller kann ausgewählte IO-Devices beim Hochlauf priorisieren.
- Das verwendete IO-Device unterstützt die Priorisierung.

Hinweis

Priorisierter Hochlauf

Im Fall eines beschleunigten Hochlaufs (Priorisierter Hochlauf), müssen Sie besondere Bedingungen bei der Einstellung der PROFINET-Schnittstelle und der Verkabelung beachten, wenn Sie die kürzest möglichen Hochlaufzeiten erreichen möchten.

Vorgehen

1. Selektieren Sie in der Netz- oder Gerätesicht das IO-Device, dessen Hochlauf beschleunigt werden soll.
2. Öffnen Sie im Inspekorfenster die Eigenschaften des IO-Devices.
3. Gehen Sie auf "PROFINET-Schnittstelle > Erweiterte Optionen > Schnittstellen-Optionen".
4. Aktivieren Sie das Optionskästchen "Priorisierten Hochlauf aktivieren".
5. Laden Sie die Projektierung in den IO-Controller.

Anmerkung:

Bei Geräten, die eine einstellbare DP-Betriebsart aufweisen, ist das zuvor beschriebene Optionskästchen zu finden unter "PROFINET-Schnittstelle > Betriebsart". Zu diesen Gerätetypen gehört beispielsweise der CP 1616 für PC-Stationen.

Hinweis

Anzahl von IO-Devices (Dezentrale Peripherie) mit priorisiertem Hochlauf

Innerhalb eines PROFINET IO-Systems können Sie nur eine vom verwendeten IO-Controller abhängige, maximale Anzahl von IO-Devices mit der PROFINET-Funktionalität "Priorisierter Hochlauf" anlaufen lassen.

Siehe auch

Port-Einstellungen am IO-Device und IO-Controller optimieren (Seite 3097)

Port-Einstellungen am IO-Device und IO-Controller optimieren

Port-Einstellungen am IO-Device und IO-Controller optimieren

Während des Hochlaufs des IO-Devices findet im Fall einer CU-Verkabelung eine Prüfung des Übertragungsmediums und der Duplexoption statt.

Diese Prüfungen nehmen Zeit in Anspruch - durch gezielte Voreinstellung dieser Optionen kann die Zeit für die Prüfung eingespart werden. Stellen Sie sicher, dass die vorgenommenen Einstellungen den realen Gegebenheiten entsprechen (Verwendung der richtigen Kabel).

Eigenschaften von lokalem Port und Partnerport angleichen

Um die Einstellungen für den lokalen Port und den Partnerport aneinander anzugleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie jeweils zuerst den lokalen Port und anschließend den Partnerport und führen Sie die nachfolgenden Schritte für beide Ports durch.
2. Gleichen Sie in den Eigenschaften des jeweiligen Ports unter "Port-Optionen" die Einstellung "Übertragungsmedium/Duplex" an. Verwenden Sie die Einstellung "TP 100 Mbit/s Vollduplex".
3. Deaktivieren Sie unter "Port-Optionen" bei beiden Ports das Optionskästchen "Autonegotiation aktivieren".

Medienredundanz konfigurieren

Wissenswertes zur Medienredundanz

Möglichkeiten der Medienredundanz

Zur Erhöhung der Netzverfügbarkeit eines Industrial Ethernet-Netzwerks mit optischen oder elektrischen Linientopologien stehen verschiedene Möglichkeiten zur Verfügung:

- Vermaschung von Netzwerken
- Parallelschaltung von Übertragungswegen
- Zusammenschluss einer Linientopologie zu einer Ringtopologie

Medienredundanz in Ringtopologien

Teilnehmer von Ringtopologien können externe Switche und/oder die integrierten Switche von Kommunikationsbaugruppen sein.

Zum Aufbau einer Ringtopologie mit Medienredundanz müssen Sie die beiden freien Enden einer linienförmigen Netztopologie in einem Gerät zusammenführen. Der Zusammenschluss der Linientopologie zu einem Ring erfolgt über zwei Ports (Ringports) eines Geräts im Ring. Dieses Gerät ist der Redundanzmanager. Alle anderen Geräte im Ring sind Redundanz-Clients.

Das folgende Bild zeigt die ringförmige Verschaltung von IO-Devices:

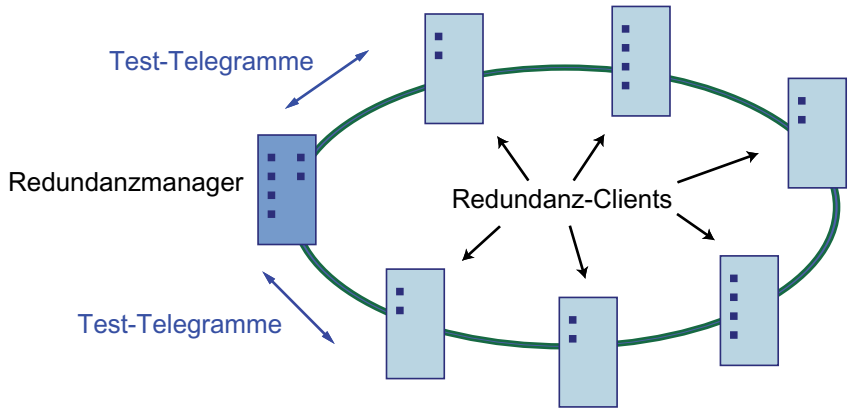

Die zwei Ringports eines Geräts sind die Ports, die in einer Ringtopologie die Verbindung zu seinen beiden Nachbargeräten herstellen. Die Auswahl und Festlegung der Ringports erfolgt in der Projektierung des jeweiligen Geräts.

Laden Sie vor dem physischen Zusammenschluss des Rings die Projektierung in die einzelnen Geräte.

Funktionen der Medienredundanz in einer Ringtopologie

Wenn der Ring an einer Stelle unterbrochen wird, dann werden die Datenwege zwischen den einzelnen Geräten automatisch rekonfiguriert. Nach der Rekonfiguration sind die Geräte in der neu entstandenen Topologie wieder erreichbar.

Im Redundanzmanager werden die 2 Ringports bei unterbrechungsfreiem Netzwerkbetrieb voneinander getrennt, damit keine Datentelegramme kreisen. Die Ringtopologie wird aus Sicht der Datenübertragung zu einer Linie. Der Redundanzmanager überwacht die Ringtopologie. Hierzu schickt er Test-Telegramme sowohl von Ringport 1 als auch von Ringport 2. Die Test-Telegramme durchlaufen den Ring in beiden Richtungen, bis sie am jeweils anderen Ringport des Redundanzmanagers ankommen.

Eine Unterbrechung des Rings kann durch Ausfall der Verbindung zwischen zwei Geräten oder durch Ausfall eines Geräts im Ring erfolgen.

Wenn die Test-Telegramme des Redundanzmanagers bei einer Unterbrechung des Rings nicht mehr zum anderen Ringport durchgeleitet werden, schaltet der Redundanzmanager seine beiden Ringports durch. Über diesen Ersatzweg wird wieder eine funktionierende Verbindung zwischen allen verbleibenden Geräten in Form einer linienförmigen Netztopologie hergestellt.

Sobald die Unterbrechung beseitigt ist, werden die ursprünglichen Übertragungswege wieder genützt, die beiden Ringports im Redundanzmanager voneinander getrennt und die Redundanz-Clients über den Wechsel informiert. Der Datenverkehr verläuft jetzt wieder wie vor der Unterbrechung.

Medienredundanzverfahren

In STEP 7 können Sie das Medienredundanzverfahren MRP (Media Redundancy Protocol) für Ringtopologien verwenden, es können bis zu 50 Geräte pro Ring teilnehmen.

Darüber hinaus gibt es auch das echtzeitfähige Medienredundanzverfahren MRPD (Media Redundancy with Planned Duplication of frames).

Rekonfigurationszeit

Die Zeit zwischen Ringunterbrechung und Wiederherstellung einer funktionsfähigen Linientopologie (mit Neuaufbau der Adresstabellen) wird Rekonfigurationszeit genannt. Sie beträgt typischerweise 200 ms.

Wenn die Ansprechüberwachungszeit eines IO-Devices kleiner ist als die Rekonfigurationszeit des Rings, besteht die Gefahr, dass die Kommunikation zum IO-Device unterbrochen wird (Stationsausfall).

Empfehlung: Wählen Sie bei der Projektierung die Ansprechüberwachungszeit eines IO-Devices größer als 200 ms (in den Eigenschaften des IO-Devices: PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen).

Siehe auch

Media Redundancy Protocol (MRP) (Seite 3100)

Medienredundanz projektieren (Seite 3102)

Media Redundancy with Planned Duplication of frames (MRPD) (Seite 3105)

MRP-Domains verwalten (Seite 3107)

Media Redundancy Protocol (MRP)

Media Redundancy Protocol (MRP)

Das Verfahren "MRP" arbeitet konform zum Media Redundancy Protocol (MRP), das in der Norm IEC 62439-2 spezifiziert ist.

Voraussetzungen

- Alle Geräte im Ring unterstützen das Verfahren MRP.
- Sie haben die unten beschriebenen Regeln für die Topologie eingehalten. STEP 7 überwacht die Regeln beim Übersetzen und gibt entspechende Meldungen aus.

Topologie

Die folgende Abbildung zeigt eine mögliche Topologie für Geräte in einem Ring mit MRP. Die Geräte innerhalb des schattierten Ovals befinden sich in der Redundanzdomäne.

Beispiel einer Ringtopologie mit dem Medienredundanzverfahren MRP:

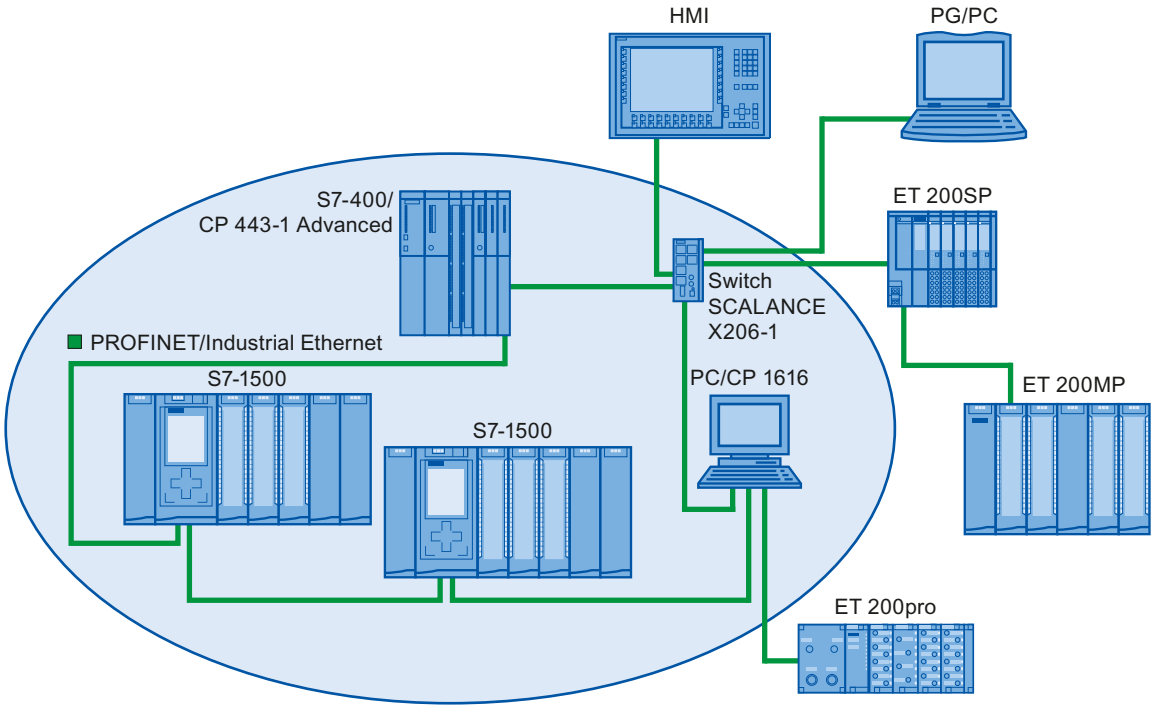

Für die Ringtopologie mit Medienredundanz nach dem Verfahren MRP gelten folgende Regeln:

- Alle Geräte müssen über ihre Ringports miteinander verbunden sein.
- Alle Geräte im Ring gehören zur selben Redundanz-Domäne.

- Ein Gerät im Ring ist Redundanzmanager.
  - Genau ein Gerät besitzt die Rolle "Manager". Kein weiteres Gerät darf die Rolle "Manager" besitzen. Oder
  - Mehrere Geräte im Ring besitzen die Rolle "Manager (auto)". Die Geräte mit der Rolle "Manager (auto)" handeln dann unter sich aus, wer tatsächlich Redundanzmanager wird. In diesem Fall darf kein Gerät die Rolle "Manager" besitzen.
- Alle anderen Geräte im Ring sind Redundanz-Clients.
- Sie können bis zu 50 Geräte zu einem Ring zusammenschließen.

Nicht MRP-fähige Geräte können beispielsweise über einen Switch SCALANCE X oder einen PC mit CP 1616 an den Ring angebunden werden.

Regel zum Laden der Geräte einer MRP-Domain

Beim Laden von Geräten einer MRP-Domain kann es zu kreisenden Frames und damit zum Ausfall des Netzwerks kommen, wenn eine ungültige MRP-Projektierung vorliegt.

Beispiel: Sie ändern die MRP-Rollen von mehreren Geräten und laden nacheinander die Konfiguration in die beteiligten Geräte. Es können Konfigurationen entstehen, die den oben genannten Regeln widersprechen, z. B. könnten Geräte mit der Rolle "Manager" und "Manager (auto)" zu einem Zeitpunkt gleichzeitig im Ring existieren.

Damit eine ungültige MRP-Konfiguration nicht zu einem Ausfall des Netzwerks führt, lösen Sie vor dem Laden den Ring. Gehen Sie folgendermaßen vor:

1. Lösen Sie den Ring.
2. Laden Sie die fehlerfreie und konsistente MRP-Projektierung aus Ihrem Projekt in alle beteiligten Geräte und stellen Sie sicher, dass sich die Geräte im Datenaustausch befinden (d. h. die Applikationsbeziehungen (ARs) sind eingerichtet).
3. Schließen Sie den Ring.

Randbedingungen

MRP und RT

RT-Betrieb ist bei der Verwendung von MRP möglich.

Hinweis

Die RT-Kommunikation wird unterbrochen (Stationsausfall), wenn die Rekonfigurationszeit des Rings größer als die gewählte Ansprechüberwachungszeit der IO-Devices ist. Wählen Sie deshalb die Ansprechüberwachungszeit der IO-Devices ausreichend groß.

MRP und IRT

Der IRT-Betrieb ist zusammen mit MRP nicht möglich.

Wenn Sie in einem Ring Medienredundanz zusammen mit IRT nutzen wollen, dann verwenden Sie ausschließlich Geräte, die das Verfahren MRPD unterstützen.

MRP und TCP/IP (Open User Communication über z. B. TSEND, Webzugriff über HTTP, ...)

Die TCP/IP-Kommunikation ist zusammen mit MRP möglich, da verlorene Datenpakete ggf. wiederholt versendet werden.

MRP und Priorisierter Hochlauf

Wenn Sie MRP in einem Ring projektieren, dann können Sie in den beteiligten Geräten in PROFINET-Applikationen die Funktion "Priorisierter Hochlauf" nicht nutzen.

Wenn Sie die Funktion "Priorisierter Hochlauf" für ein Gerät nutzen wollen, darf das Gerät nicht Teilnehmer des Ring sein.

Siehe auch

Medienredundanz projektieren (Seite 3102)

Wissenswertes zur Medienredundanz (Seite 3098)

Mehrfachringe projektieren (Seite 3108)

Medienredundanz projektieren

MRP projektieren

Um in STEP 7 eine PROFINET IO-Konfiguration mit MRP zu erstellen, gehen Sie folgendermaßen vor:

1. Erzeugen Sie in der Topologiesicht über die Portverschaltungen einen Ring. Verschalten Sie die Geräte zuerst zu einer Linietopologie. Verbinden Sie den unbelegten Port des letzten Gerätes in der Linie mit dem unbelegten Port des ersten Gerätes.
   Das folgende Beispiel zeigt eine CPU 1516-3 PN/DP und zwei Interfacemodule IM 155-6 PN HF, die in der Topologiesicht von STEP 7 zu einem Ring verschaltet sind.

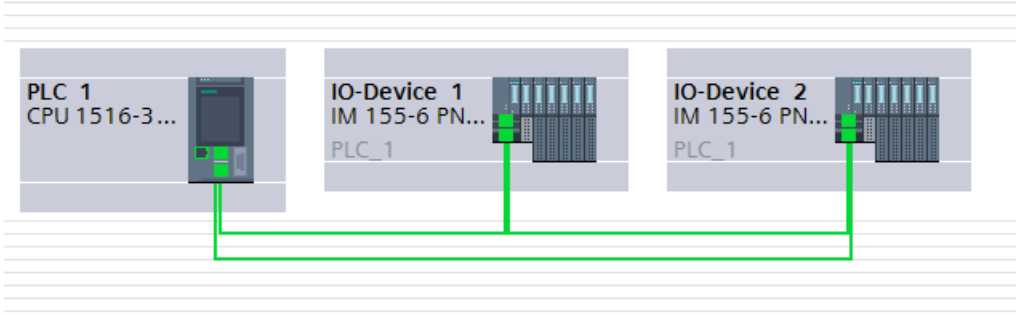

2. Selektieren Sie in der Netzsicht das PROFINET IO-System.
3. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "MRP-Domains" in das Feld "Ringverschaltung".
   Dieses Feld zeigt Ihnen alle topologischen Ringe im IO-System mit den dazugehörigen MRP-Domains.

4. Wählen Sie im Feld "Ringverschaltung" oben erzeugten Ring aus.
   Die darunter befindliche Tabelle zeigt alle PROFINET-Geräte im Ring an.
5. Stellen Sie für die PROFINET-Geräte in der Spalte "MRP-Rolle" die Medienredundanzrolle ein.
   Wenn Sie die Geräte miteinander verschalten, dann sind die MRP-Rollen zunächst bei allen Geräten auf "Nicht Teilnehmer des Rings" eingestellt. Diese Konfiguration ist nicht konsistent.
   Sie haben folgende Möglichkeiten: Die MRP-Rollen können Sie manuell zuweisen entsprechend den Regeln von MRP-Konfigurationen (siehe Enstellmöglichkeiten "Medienredundanz"). Die MRP-Rollen können Sie von STEP 7 auch automatisch zuweisen lassen, siehe nächsten Abschnitt.

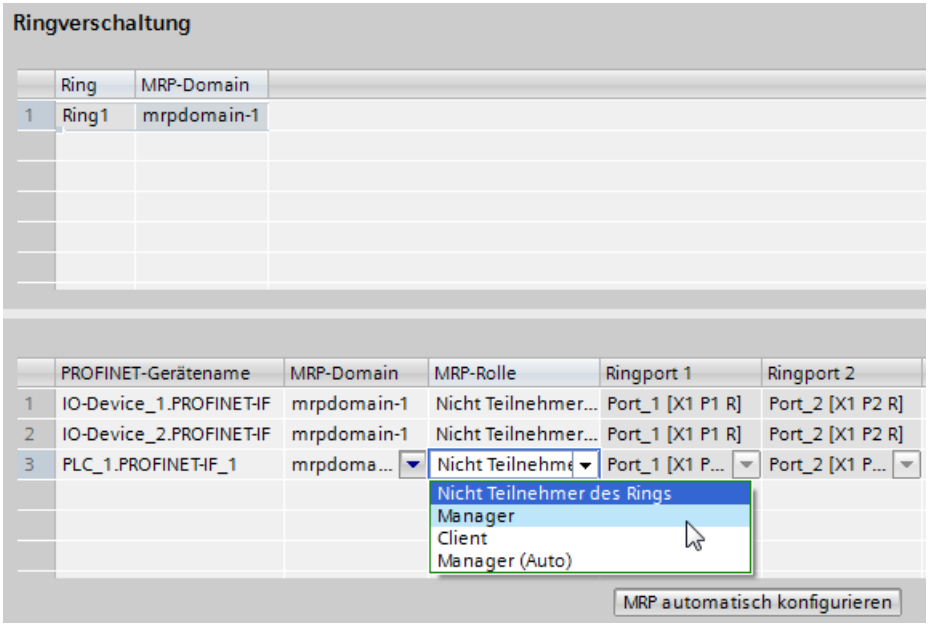

Automatische MRP-Konfiguration

Sie können für Ihre PROFINET-Geräte im Ring die Medienredundanzrollen auch automatisch zuweisen lassen.

Um die Medienredundanzrollen automatisch zuweisen zu lassen, klicken Sie auf die Schaltfläche "MRP automatisch konfigurieren". STEP 7 vergibt automatisch für jedes Gerät im Ring die Medienredundanzrolle. Nach der automatischen MRP-Konfiguration können Sie wie gewohnt in der Spalte "MRP-Rolle" Änderungen der Medienredundanzrollen vornehmen.

Einstellmöglichkeiten "Medienredundanz"

Medienredundanzrolle

Es sind je nach verwendetem Gerät die Rollen "Manager", "Manager (Auto)", "Client" und "Nicht Teilnehmer des Rings" verfügbar.

Regeln:

- Ein Ring darf genau ein Gerät mit der Rolle "Manager" haben. Es sind keine weiteren Geräte mit der Rolle "Manager" oder "Manager (Auto)" zulässig. Alle anderen Geräte dürfen nur noch die Rolle "Client" haben.
- Wenn ein Ring kein Gerät mit der Rolle "Manager" hat, dann muss der Ring mindestens über ein Gerät mit der Rolle "Manager (Auto)" verfügen. Geräte mit den Rollen "Client" dürfen beliebig vorhanden sein.

Ringport 1/Ringport 2

Wählen Sie hier jeweils den Port aus, den Sie als Ringport 1 bzw. als Ringport 2 projektieren möchten. Die Klappliste zeigt für jeden Gerätetyp die Auswahl der möglichen Ports an. Wenn die Ports werkseitig festgelegt sind, dann sind die Felder gegraut.

Wenn Sie die einstufige Inbetriebnahme nutzen (nur Siemens-Geräte mit automatisch zugewiesenen MRP-Rollen im Ring nach Klicken auf "MRP automatisch konfigurieren"), dann verwenden Sie die in STEP 7 voreingestellten Ringports.

Diagnosealarme

Wenn Diagnosealarme zum MRP-Zustand in der lokalen CPU ausgegeben werden sollen, aktivieren Sie das Optionskästchen "Diagnosealarme". Folgende Diagnosealarme können gebildet werden:

- Verdrahtungs- bzw. Portfehler
  Bei folgenden Fehlern an den Ringports werden Diagnosealarme generiert:
  - Ein Nachbar des Ringports unterstützt MRP nicht.
  - Ein Ringport ist mit einem Nicht-Ringport verbunden.
  - Ein Ringport ist mit dem Ringport einer anderen MRP-Domain verbunden.
- Unterbrechung/Wiederkehr (nur Redundanzmanager)
  Bei Unterbrechung des Rings und bei Wiederkehr der ursprünglichen Konfiguration generiert der Redundanzmanager Diagnosealarme. Das Auftreten dieser beiden Alarme innerhalb von 0,2 Sekunden deutet auf eine Unterbrechung des Rings hin.

Sie können im Anwenderprogramm auf diese Ereignisse reagieren, indem Sie eine entsprechende Reaktion im Diagnosealarm-OB (OB 82) programmieren.

Tipp: Löschen von nicht benötigten MRP-Domains

Wenn Sie nicht mehr benötigte MRP-Domains löschen wollen, weil sie z. B. keine Geräte mehr enthalten, dann wählen Sie bei markierten PROFINET IO-Systemen den Bereich "MRP-Domains":

1. Navigieren Sie zur ersten Tabelle. Das ist die Tabelle, in der Sie die Default-Domain wählen.
2. Markieren Sie die Zeile mit der zu löschenden MRP-Domain und drücken die Taste <Entf>.

Bis auf die Default-Domain können Sie jede MRP-Domain löschen.

Siehe auch

Mehrfachringe projektieren (Seite 3108)

Media Redundancy Protocol (MRP) (Seite 3100)

Wissenswertes zur Medienredundanz (Seite 3098)

MRP-Domains verwalten (Seite 3107)

Media Redundancy with Planned Duplication of frames (MRPD)

MRP-Erweiterung "Media Redundancy with Planned Duplication of frames" (MRPD)

Die MRP-Erweiterung "Media Redundancy with Planned Duplication of frames" (MRPD) bringt den Vorteil, dass beim Ausfall eines Geräts oder einer Leitung im Ring alle anderen Geräte ohne Unterbrechung und mit kurzen Aktualisierungszeiten weiter mit IO-Daten versorgt werden.

MRPD basiert auf IRT und MRP. Um Medienredundanz mit kurzen Aktualisierungszeiten zu erreichen, senden die am Ring beteiligten PROFINET-Geräte ihre Daten in beide Richtungen. Die Geräte empfangen diese Daten an beiden Ringports, dadurch entfällt die Rekonfigurationszeit des Rings.

Voraussetzungen für Medienredundanz mit MRPD

- Alle Teilnehmer des Rings müssen MRPD unterstützen, z. B. Interfacemodul IM 155-6 PN HS ab Firmwarestand V4.0, Scalance X300 ab V4.0. Auch die (End-) Geräte am Switch, welche mit einer Ring-Komponente zyklisch IRT-Daten austauschen, müssen MRPD unterstützen.
  Ob ein Gerät MRPD unterstützt, zeigt der Infotext zum Gerät im Hardware-Katalog von STEP 7 (Geräte & Netze).
- Sie haben für alle Teilnehmer des Rings MRP projektiert. Geräten, die sich nicht im Ring befinden, haben Sie die MRP-Rolle "Nicht Teilnehmer des Rings" zugewiesen.
- Sie haben für alle beteiligten Komponenten IRT projektiert.

MRPD projektieren

Sie brauchen in STEP 7 MRPD nicht explizit aktivieren. Sobald alle Voraussetzungen für MRPD erfüllt sind, ist die Funktion automatisch verfügbar.

Redundanzstufen von IO-Devices mit MRPD

Die Redundanzstufe eines IO-Devices gibt an, wie stark die Echtzeit-Kommunikation im Falle einer Netzunterbrechung zwischen einem IO-Device und dessen IO-Controller beeinflusst wird.

- Vollständige Redundanz: Keine Beeinflussung, da sich IO-Controller und IO-Device im selben Ring befinden.
- Teilweise Redundanz:
  - Wenn die Unterbrechung auf einem nicht redundanten Teil (Linie) zwischen IO-Device und IO-Controller stattfindet, dann wird die Echtzeit-Kommunikation beeinflusst.
  - Wenn die Unterbrechung auf dem redundanten Teil (Ring) stattfindet, dann wird die Echtzeit-Kommunikation nicht beeinflusst.
- Keine Redundanz: Keine redundanten Pfade zwischen IO-Device und dessen IO-Controller, Kommunikation wird immer beeinflusst.

Das folgende Bild zeigt für eine Beispielkonfiguration mit MRPD die Redundanzstufen der IO-Devices. Die drei IO-Devices im Ring und das Switch haben die Redundanzstufe "Vollständige Redundanz". Das Device 4 hat die Redundanzstufe "Teilweise Redundanz", weil die Verbindung zwischen dem Switch und dem IO-Device nicht redundant ist.

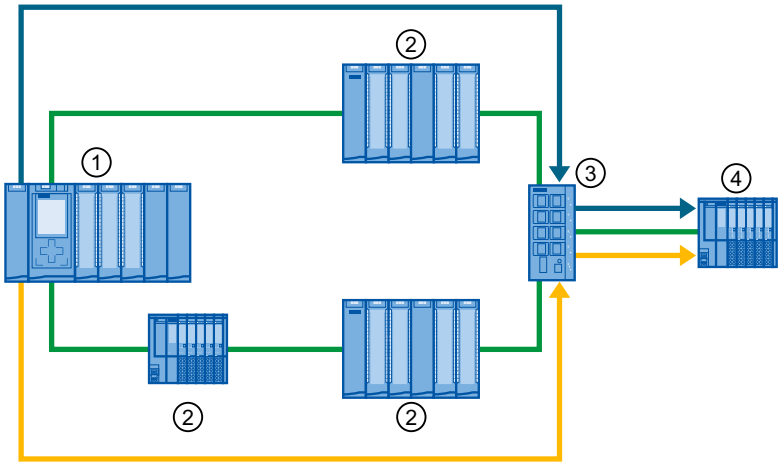

① IO-Controller im Ring
② IO-Device im Ring mit Redundanzstufe "Vollständige Redundanz"
③ Switch im Ring mit Redundanzstufe "Vollständige Redundanz"
④ IO-Device mit Stichleitung an den Ring angebunden mit Redundanzstufe "Teilweise Redundanz"

Die Redundanzstufen werden neben anderen Eigenschaften der Sync-Domain in STEP 7 angezeigt:

Domain-Managemant > Sync-Domains > ... > Teilnehmer.

Anwendungsbeispiel: Aufbau einer Ring-Topologie auf Basis „MRPD”

Über den folgenden Link können Sie sich ein Anwendungsbeispiel zum Aufbau einer Ring-Topologie auf Basis von MRPD herunterladen:

https://support.industry.siemens.com/cs/ww/de/view/109744035 (http://support.automation.siemens.com/WW/view/de/109744035)

Siehe auch

Wissenswertes zur Medienredundanz (Seite 3098)

MRP-Domains verwalten

Die Einstellungen für Medienredundanz können Sie zentral kontrollieren und einstellen.

Voraussetzungen

- Die beteiligten Komponenten müssen das Media Redundancy Protocol (MRP) unterstützen.
- Die Geräte sind konfiguriert und topologisch verschaltet.

Vorgehen

Um die MRP-Domain zu öffnen, gehen Sie folgendermaßen vor:

1. Markieren Sie eine beliebige PROFINET-Schnittstelle und wählen Sie im Inspektorfenster den Bereich "Erweiterte Optionen > Medienredundanz".
2. Klicken Sie auf die Schaltfläche "Domain-Einstellungen".

oder

1. Markieren Sie das Subnetz bzw. das IO-System.
2. Navigieren Sie im Inspektorfenster zum Bereich "PROFINET > Domain-Management > MRP-Domains".

Hier können Sie zentral festlegen, welche MRP-Domain die Default-Domain sein soll, Sie können die Namen sämtlicher Domains ändern sowie Redundanzrollen und Ringports kontrollieren.

Für jede MRP-Domain erhalten Sie eine Übersicht, wie viele PROFINET-Schnittstellen im Ring verschaltet sind, wie viele PROFINET-Schnittstellen sich außerhalb des Rings befinden und die Anzahl der Redundanz-Manager sowie der Redundanz-Clients.

Sie können sich die Geräte im Ring mit Gerätenamen, zugehöriger MRP-Domain, Medienredundanzrolle und die verwendeten Ringports gruppiert anzeigen lassen:

- MRP-Domains
  Die Liste zeigt Ihnen die vorhandenen MRP-Domains des selektierten IO Systems an.
- <Name der MRP-Domain>
  Die Liste zeigt Ihnen die Anzahl der PROFINET-Schnittstellen und der Geräte mit den jeweiligen Redundanzrollen für die ausgewählte MRP-Domain an.
- Teilnehmer
  Die Liste zeigt Ihnen alle projektierten Geräte des selektierten IO Systems an.

Siehe auch

Wissenswertes zur Medienredundanz (Seite 3098)

Medienredundanz projektieren (Seite 3102)

Mehrfachringe projektieren

Mehrfachringe

Um eine höhere Verfügbarkeit für PROFINET IO-Netze mit Sterntopologie zu erreichen, verwenden Sie Mehrfachringe.

Bei einer Mehrfachring-Konfiguration gehen mehrere PROFINET-Stränge von einem Switch aus (Sterntopologie). Die einzelnen PROFINET-Stränge führen von IO-Device zu IO-Device. Vom jeweils letzten IO-Device an den einzelnen Strängen führen PROFINET-Leitungen zurück zum Switch und bilden auf diese Weise mehrere Ringe.

Der Switch arbeitet als Manager. Der Manager muss für jeden Ring zwei Ringports besitzen. Es sind mehrere Ringe möglich. Ein SCALANCE X414 ab Firmwarestand V3.10 unterstützt z. B. bis zu 4 Ringe.

Der Manager überwacht alle Ringe einzeln: Er überprüft jeweils für einen bestimmten Ring (eine MRP-Domain), ob der Übertragungspfad intakt ist. Dazu verwendet er jeweils eine MRP-Instanz. Für jeden angeschlossenen Ring ist eine MRP-Instanz erforderlich (richtet STEP 7 automatisch ein).

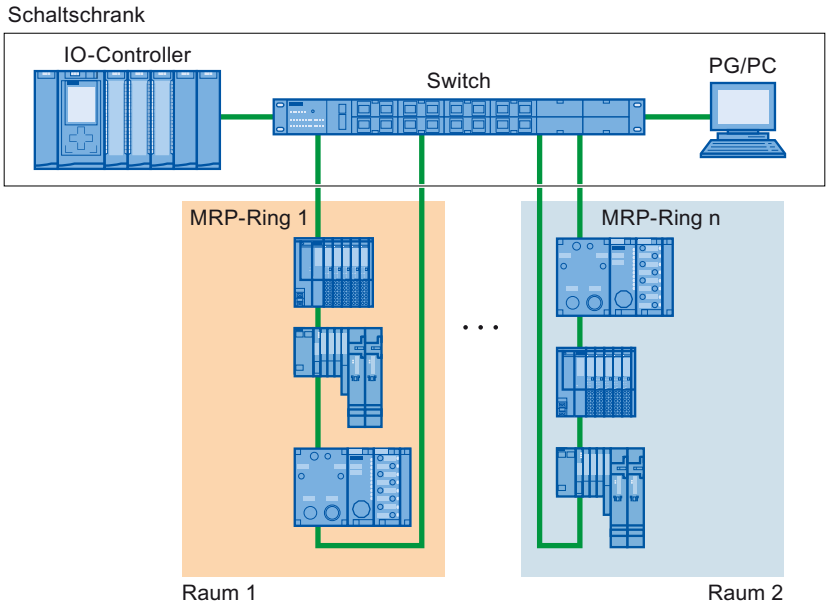

Voraussetzungen

Folgende Geräte unterstützen Mehrfachringe als Manager:

- SCALANCE X300 ab Version V4.0
- SCALANCE X414 ab Version V3.10

Regeln für die Konfiguration von Mehrfachringen

- MRP-Rolle bei Mehrfachringen:
  - Das Gerät, das allen Ringen angehört, muss in jeder Instanz die MRP-Rolle besitzen, die in der GSD-Datei im Attribut "SupportedMultipleRole" eingetragen ist.
  - Die Switche aus der Reihe SCALANCE X300 ab V4.0 (nur über GSDML) und das Switch X414 ab Version 3.10 unterstützen für Mehrfachringe die MRP-Rolle "Manager".
- Wenn das Gerät, das allen Ringen angehört, in einem Ring die Rolle "Manager" besitzt, dann dürfen in diesem Ring keine Geräte mit der Rolle "Manager (Auto)" vorhanden sein.

Mehrfachringe projektieren

Um eine MRP-Konfiguration mit Mehrfachringen zu projektieren, gehen Sie folgendermaßen vor:

1. Verschalten Sie in der Topologiesicht die Ringports der Geräte, die einer MRP-Domain angehören sollen, jeweils zu einem Ring.
2. Selektieren Sie in der Netzsicht das PROFINET IO-System.
3. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "MRP-Domains" in das Feld "Ringverschaltung".
   Dieses Feld zeigt Ihnen alle topologischen Ringe im IO-System mit den dazugehörigen MRP-Domains.
4. Markieren Sie im Feld "Ringverschaltung" einen der oben erzeugten Ringe.
   Die darunter befindliche Tabelle zeigt alle PROFINET-Geräte im markierten Ring an.
5. Stellen Sie für die PROFINET-Geräte in der Spalte MRP-Rolle die Medienredundanzrolle ein.

Voreingestellte MRP-Rollen anpassen

Wenn Sie die Geräte der Beispielkonfiguration wie unten beschrieben miteinander verschalten, dann sind die MRP-Rollen zunächst bei allen Geräten auf "Nicht Teilnehmer des Rings" eingestellt.

Diese Konfiguration ist nicht konsistent.

Sie haben folgende Möglichkeiten:

- Die MRP-Rollen können Sie manuell zuweisen entsprechend den beschriebenen Regeln für MRP-Konfigurationen. Editieren Sie dazu die Eigenschaften des PROFINET IO-Systems, Bereich "MRP-Domains > Ringverschaltung".
- Die MRP-Rollen können Sie von STEP 7 automatisch zuweisen lassen.

In beiden Fällen markieren Sie einen projektierten MRP-Ring (im Bereich "MRP-Domains > Ringverschaltung") und klicken auf die Schaltfläche "MRP automatisch konfigurieren". Wiederholen Sie den Vorgang für jeden projektierten MRP-Ring.

Anschließend sind die Ringe neuen MRP-Domains zugewiesen und die MRP-Rollen und MRP-Instanzen sind gesetzt. Die Konfiguration ist konsistent.

Beispiel für Mehrfachringe

Das folgende Bild zeigt eine Beispielkonfiguration für Mehrfachringe mit zwei Ringen ① und ②.

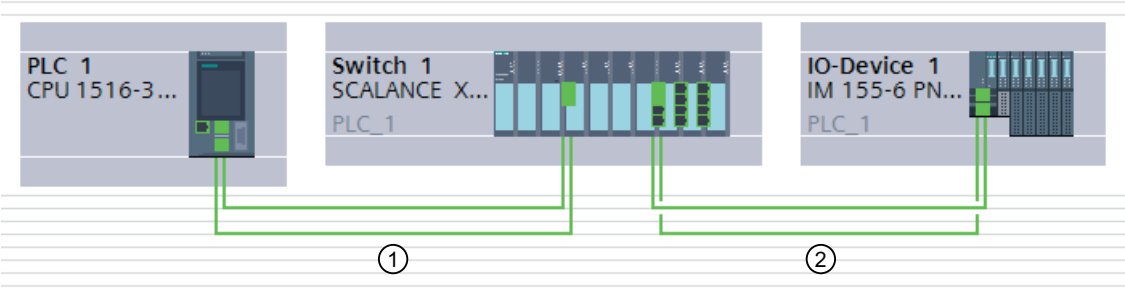

Im Beispiel gehört Switch 1 zu zwei MRP-Ringen. Ring 1 wird gebildet von Switch 1 und PLC 1, Ring 2 von Switch 1 und IO-Device 1.

Der Manager liegt im Schnittpunkt der beiden Ringe 1 und 2. Der Manager überwacht beide Ringe jeweils für sich. Dafür nutzt er zwei MRP-Instanzen.

Eine MRP-Instanz kontrolliert, ob alle Geräte im Ring 1 erreichbar sind, eine andere Instanz überwacht, ob alle Geräte im Ring 2 erreichbar sind (im Beispiel jeweils nur ein Gerät).

Sie können jede MRP-Instanz getrennt konfigurieren.

Das folgende Bild zeigt die beiden MRP-Instanzen im Manager (PROFINET Schnittstelle des Switches). Hier im Beispiel kontrolliert die MRP-Instanz 1, ob die Geräte der MRP-Domain "mrpdomain-1" erreichbar sind. Die MRP-Instanz 2 ist für die Überwachung der Geräte der MRP-Domain "mrpdomain-2" zuständig.

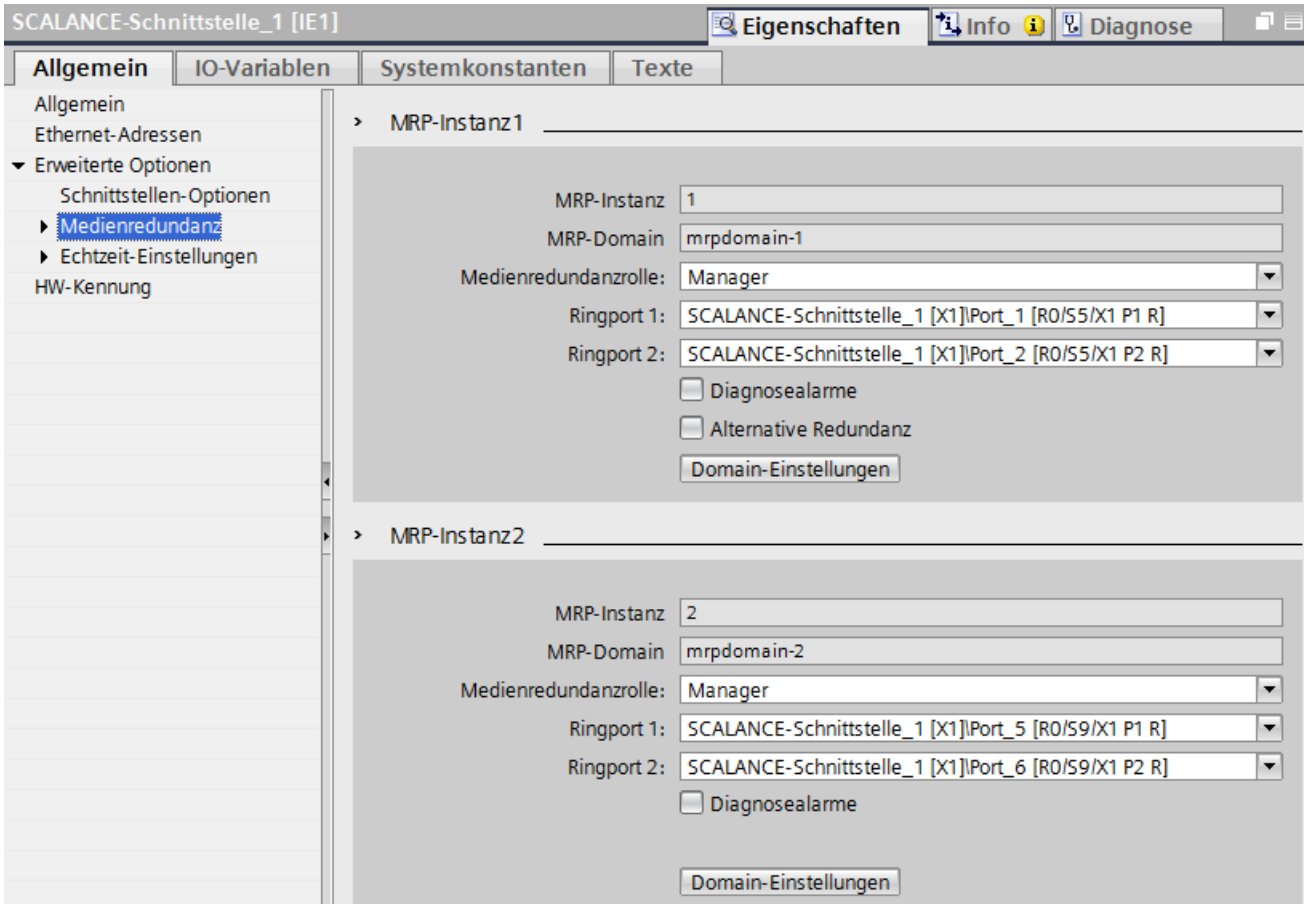

Das folgende Bild zeigt den Ring 1 (mrpdomain-1). Die Teilnehmer der mrpdomain-1 sind die PROFINET-Schnittstelle der CPU als "Client" und die MRP-Instanz 1 der PROFINET-Schnittstelle des Switches als "Manager".

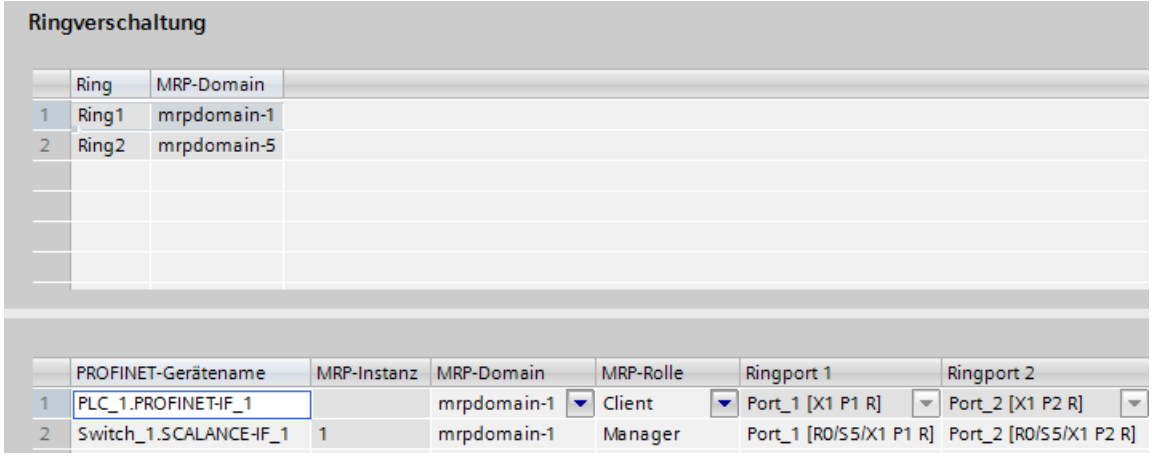

Das folgende Bild zeigt den Ring 2 (mrpdomain-2). Die Teilnehmer der mrpdomain-2 sind die PROFINET-Schnittstelle des IO-Devices als "Client" und die MRP-Instanz 2 der PROFINET-Schnittstelle des Switches als "Manager".

Ringverschaltung

	Ring	MRP-Domain
1	Ring1	mrpdomain-1
2	Ring2	mrpdomain-2

	PROFINET-Gerätename	MRP-Instanz	MRP-Domain	MRP-Rolle	Ringport 1	Ringport 2
1	IO-Device_1.PROFINET-IF		mrpdomain-2	Client	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]
2	Switch_1.SCALANCE-IF_1	2	mrpdomain-2	Manager	Port_5 [R0/S9/X1 P1 R]	Port_6 [R0/S9/X1 P2 R]

Tipp: Löschen von nicht benötigten MRP-Domains

Wenn Sie nicht mehr benötigte MRP-Domains löschen wollen, weil sie z. B. keine Geräte mehr enthalten, dann wählen Sie bei markierten PROFINET IO-Systems den Bereich „MRP-Domains“.

1. Navigieren Sie zur ersten Tabelle. Das ist die Tabelle, in der Sie die Default-Domain wählen.
2. Markieren Sie die Zeile mit der zu löschenden MRP-Domain und drücken die Taste <Entf>.

Bis auf die Default-Domain können Sie jede MRP-Domain löschen.

Siehe auch

Medienredundanz projektieren (Seite 3102)

Media Redundancy Protocol (MRP) (Seite 3100)

PROFIenergy projektieren

Übersicht PROFIenergy

Energie sparen mit PROFIenergy

PROFIenergy ist eine auf PROFINET basierende Datenschnittstelle, die es erlaubt, hersteller- und geräteunabhängig Verbraucher koordiniert und zentral gesteuert in Pausenzeiten abzuschalten. Dadurch soll dem Prozess nur die absolut notwendige Energie zu Verfügung gestellt werden. Der Großteil der Energie wird dabei vom Prozess gespart, das PROFINET-Gerät selbst trägt nur mit einigen Watt zum Einsparpotenzial bei.

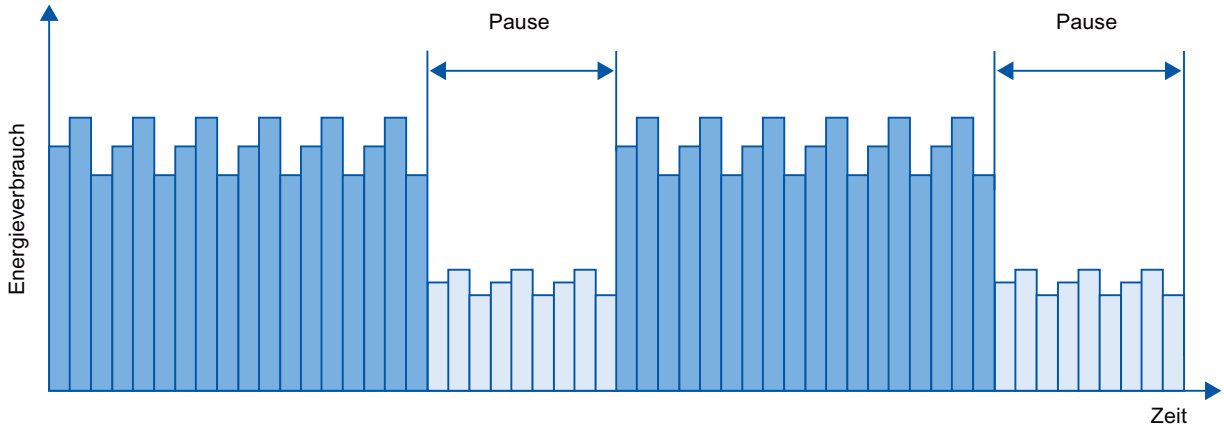

Bild 1-93 Energieeinsparung in Pausen mit PROFIenergy

Grundlegendes

Die Abschaltung der PROFINET-Geräte bzw. der Powermodule erfolgt über spezielle Kommandos im Anwenderprogramm des PROFINET IO-Controllers. Es wird keine zusätzliche Hardware benötigt, die PROFIenergy Kommandos werden direkt von den PROFINET-Geräten interpretiert.

Funktionsweise

Zu Beginn und am Ende von Pausen aktiviert oder deaktiviert der Anlagenführer die Pausenfunktion der Anlage; daraufhin sendet der IO-Controller das PROFIenergy-Kommando "Start_Pause" / "End_Pause" an die PROFINET-Geräte. Das Gerät interpretiert dann den Inhalt des PROFIenergy-Kommandos und schaltet ab / wieder an.

Über weitere PROFIenergy Funktionen können während der Pausen Geräteinformationen abgerufen werden. Diese kann der Anwender nutzen, um das "Start_Pause" / "End_Pause" Kommando rechtzeitig zu übertragen.

PROFIenergy Anweisungen für IO-Controller

Zur Steuerung und Überwachung der PROFIenergy Funktionen sind zwei Anweisungen nötig.

Mit der Anweisung PE_START_END wird auf einfachste Art und Weise der Ruhezustand der PROFINET-Geräte aktiviert oder deaktiviert. Das geschieht über eine kommende oder gehende Flanke. Die Anweisung PE_START_END bietet eine einfache Schnittstelle für die Realisierung der PROFIenergy-Kommandos Start_Pause und End_Pause.

Mit der Anweisung PE_CMD lassen sich alle PROFIenergy-Kommandos einschließlich Start_Pause und End_Pause übertragen. Mit den weiteren Kommandos kann z.B. der aktuelle Status der PROFINET-Geräte oder das Verhalten während der Pausen abgefragt werden. Die Anweisung PE_CMD ermöglicht das komfortable Handling aller PROFIenergy Funktionen.

PROFIenergy Anweisung für I-Devices

Mit der Anweisung PE_I_DEV lässt sich PROFIenergy auch auf I-Devices realisieren. Die Anweisung empfängt PROFIenergy-Kommandos auf dem I-Device und gibt sie an das Anwenderprogramm zur Bearbeitung weiter. Nach der Bearbeitung des Kommandos durch das Anwenderprogramm ruft es die Anweisung PE_I_DEV erneut auf, um die Quittung an den IO-Controller zu senden. Für diese Antworten steht Ihnen für jedes Kommando eine Hilfsanweisung zur Verfügung, die die Anweisung mit den Antwortdaten versorgt.

Die Anweisungen finden Sie in der Task Card "Anweisungen" des Programmeditors in STEP 7.

Ein Applikationsbeispiel finden Sie im Internet (http://support.automation.siemens.com/WW/view/de/41986454) im Service und Support Portal.

Projektierung und Programmierung

Die Funktionen lassen sich komfortabel in bestehende Anlagen integrieren. Für die Anwendung von PROFIenergy ist in der Regel keine Projektierung notwendig (Aushahmen unter "Siehe auch"). Es sind jedoch Ergänzungen am Anwenderprogramm nötig:

- Vor dem Kommando "Start_Pause" müssen Sie dafür sorgen, dass Ihre Anlage in einen sichereren Pause-geeigneten Zustand gebracht wird.
- Eine Zeitablaufsteuerung für den Pausenbeginn der Devices und für das rechtzeitige Wiederzuschalten der sich in der Pause befindenden Teilnehmer muss programmiert werden (in Abhängigkeit der erforderlichen Einschalt-Vorhalt-Zeiten, die das jeweilige PROFINET-Gerät verlangt).
- Die Fehlermeldungen der Anweisung PE_CMD müssen ausgewertet und die entsprechend erforderliche Reaktion programmiert werden (z.B. Abbruch oder Fortsetzung weiterer Kommandos an untergeordnete PROFINET-Geräte).

Siehe auch

PROFIenergy bei ET 200S projektieren (Seite 3115)

Anwendungsbeispiel zu PROFIenergy (https://support.industry.siemens.com/cs/de/de/view/109478388)

Energie sparen mit SIMATIC S7 und SIMATIC HMI (TIA Portal) (https://support.industry.siemens.com/cs/ww/de/view/58235225)

PROFIenergy bei ET 200S projektieren

Das Dezentrale Peripheriesystem ET 200S bietet eine Besonderheit: Sie können bei bestimmten Interfacemodulen einstellen, welche Potentialgruppen bzw. Powermodule des IO-Devices für PROFIenergy aktiviert sind.

Besonderheiten bei ET 200S HF V7.0 and ET 200S FO V7.0 Interfacemodulen

Das initiale Verhalten der Powermodule (z. B. das schaltbare Powermodul (PM-E DC24V/8A RO) können Sie für die oben genannten IO-Devices konfigurieren, so dass Sie das Gerät nicht im Anlaufprogramm mit den PROFIenergy-Anweisungen PE_Start_End_Pause in den Energiespar-Zustand bringen müssen.

Die Energiesparfunktion aktivieren Sie in den Eigenschaften des Powermoduls (Bereich Baugruppenparameter, Abschnitt "PROFIenergy").

Beachten Sie, dass für die Anwendung der PROFIenergy-Funktionen Vorkehrungen im Anwenderprogramm notwendig sind.

Regeln

Die oben beschriebene Konfigurationsmöglichkeit steht nicht bei ET 200S-Geräten zur Verfügung, die mittels GSDML-Dateien in den Hardware-Katalog aufgenommen wurden.

Siehe auch

Übersicht PROFIenergy (Seite 3113)

PROFIenergy bei I-Devices projektieren

Voraussetzung für programmgesteuerte Pausen zur Energieeinsparung bei PROFINET-Geräten ist, dass die PROFINET-Geräte das PROFIenergy-Protokoll unterstützen.

Nur wenn ein PROFINET-Gerät (IO-Device) das PROFIenergy-Protokoll unterstützt, setzt ein IO-Controller auch PE-Kommandos an dieses IO-Device ab, um z. B. Pausen zu starten oder zu beenden.

Wenn ein IO-Device das PROFIenergy-Protokoll unterstützt, dann ist diese Eigenschaft in seiner PROFINET GSD-Datei hinterlegt und steht damit einem Engineering-System für die Projektierung zur Verfügung.

Ab STEP 7 V13 Servicepack 1 haben Sie für S7-1500-CPUs als Intelligente IO-Devices (I-Devices) die Möglichkeit, für jeden Transferbereich die PROFIenergy-Unterstützung einzustellen.

Wenn Sie für einen Transferbereich die Option "PROFIenergy-Kommunikation aktivieren" aktiviert haben und die damit generierte PROFINET GSD-Datei in einem anderen Projekt importieren, können Sie dort ein I-Device als PE-Entity handhaben.

Voraussetzung

- STEP 7 ab V13 Servicepack 1
- S7-15XX-CPU ab Firmware-Version 1.7 als I-Device mit Transferbereichen projektiert
- Das Anwenderprogramm im I-Slave hantiert PROFIenergy-Kommandos
  Hintergrund: PROFIenergy-Funktionen müssen Sie bei I-Devices im Anwenderprgramm mit Hilfe der Anweisung "PE_I_DEV" und zugehöriger Hilfsbausteine ausprogrammieren; das ist anders als bei IO-Devices, bei denen diese Funktionalität über die Firmware bereitgestellt wird. Sie dürfen also nur dann für Transferbereiche die PROFIenergy-Unterstützung aktivieren, wenn auch das Anwenderprgramm im I-Slave enstprechend gestaltet ist.

PROFIenergy für Transferbereiche von I-Devices aktivieren

Um die Unterstützung für PROFIenergy zu parametrieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die PROFINET-Schnittstelle (X1) der CPU.
2. Wählen Sie in der Bereichsnavigation den gewünschten Transferbereich, z. B. Betriebsart > I-Device-Kommunikation > Transferbereich_1.
3. Aktivieren Sie das Optionskästchen "PROFIenergy-Kommunikation aktivieren".

Nach vollständiger Projektierung des I-Devices generieren Sie die GSD-Datei für das I-Device und importieren diese Datei im Projekt für den IO-Controller. Die generierte GSD-Datei ist kompatibel mit GSD-Version 2.31 und enthält einen Eintrag, der spezifiziert, dass das I-Device das PROFIenergy-Profil unterstützt.

Zum Adressieren des I-Devices z. B. für das PE-Kommando "PE_START_END" verwenden Sie die HW-Kennung des "PROFIenergy-unterstützenden" Transferbereichs im I-Device.

Zum Adressieren des IO-Controllers für das PE-Kommando "PE_I_DEV" verwenden Sie die HW-Kennung des Transferbereichs, welcher im IO-Controller mit den Daten für PROFIenergy versorgt wird.

Shared Devices konfigurieren

Wissenswertes zu Shared Devices

Funktionalität Shared Device

In größeren oder weit verteilten Anlagen werden häufig zahlreiche IO-Controller eingesetzt.

Ohne die Funktion "Shared Device" ist jedes Peripheriemodul eines IO-Devices demselben IO-Controller zugeordnet. Wenn räumlich nah beieinanderliegende Sensoren Daten an unterschiedliche IO-Controller liefern müssen, sind daher mehrere IO-Devices erforderlich.

Die Funktion "Shared Device" ermöglicht es, die Module bzw. Submodule eines IO-Devices zwischen verschiedenen IO-Controllern aufzuteilen. Durch diese Aufteilung sind flexible Automatisierungskonzepte möglich. Sie haben z. B. die Möglichkeit, räumlich naheliegende Peripheriemodule in einem IO-Device zusammenzufassen.

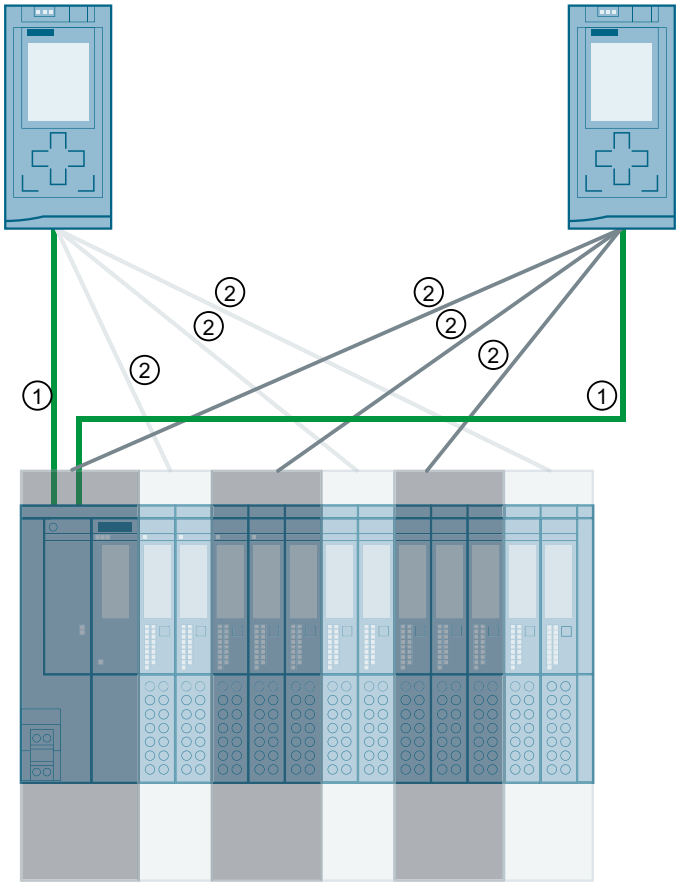

① PROFINET
② logische Zuordnung

Prinzip

Der Zugriff auf die Submodule des Shared Devices wird zwischen den einzelnen IO-Controllern aufgeteilt. Jedes Submodul des Shared Devices ist exklusiv einem IO-Controller zugeordnet.

Voraussetzung

- STEP 7 ab V12 Servicepack 1
- S7-1500 CPU ab FW 1.1 / ET 200SP CPU als IO-Controller oder S7-300/400 CPU als IO-Controller
- S7-1200 CPU ab FW 4.1
- S7-300/400 CPU (STEP 7 TIA Portal ab V13 SP1)
- IO-Device unterstützt die Funktion Shared Device, z. B. Interfacemodul IM 155-5 PN ST
- Bei GSD-Projektierung des IO-Devices: GSD-Datei für die Konfiguration des IO-Device ist installiert

Anmerkungen:

- Eine S7-1500 CPU als I-Device konfiguriert ist ab Firmware V1.5 als Shared Device einsetzbar. Das I-Device muss über einen GSD-Import in das IO-System des IO-Controllers eingebunden werden.
- Eine S7-300 CPU bzw. eine S7-400 CPU als I-Device konfiguriert ist als Shared Device einsetzbar unter folgender Voraussetzung:
  Sie müssen die PROFINET-GSD-Datei für das I-Device aus STEP 7 (ab V5.5) oder STEP 7 (TIA-Portal) exportieren und dann in STEP 7 (TIA Portal) importieren.

Projektierung des Zugriffs

Um zu erreichen, dass die Module bzw. Submodule eines IO-Devices unterschiedlichen IO-Controllern zugeordnet werden können, muss das IO-Device in mehreren Projekten vorhanden sein. Für jeden IO-Controller wird ein eigenes Projekt benötigt.

Auf welche Module bzw. Submodule der IO-Controller Zugriff hat, bestimmen Sie über den Parameter "Shared Device" des Interfacemoduls:

- Wenn der lokale IO-Controller Zugriff auf das konfigurierte Modul hat, wählen Sie aus der Liste den Namen des IO-Controllers aus.
- Wenn nicht der lokale IO-Controller, sondern der IO-Controller aus einem anderen Projekt Zugriff auf das konfigurierte Modul haben soll, wählen Sie den Eintrag "---".

Wenn jedes Modul bzw. Submodul in genau einem Projekt einem IO-Controller zugewiesen ist, dann ist die Konfiguration konsistent hinsichtlich des Zugriffs.

Modul bzw. Submodul ist anderem IO-Controller zugeordnet

Im Folgenden geht es um die Konsequenzen der Einstellung "---" des Parameters "Shared Device" aus Sicht des lokalen IO-Controllers.

Der lokale IO-Controller hat in diesem Fall keinen Zugriff auf das so konfigurierte Modul. Das bedeutet im Einzelnen:

- Kein Datenaustausch mit dem Modul bzw. Submodul
- Kein Empfang von Alarmen und Diagnosen, d. h. keine Anzeige des Diagnosezustands in der Online-Sicht
- Keine Parametrierung des Moduls bzw. Submoduls

Einstellen der Echtzeiteigenschaften

STEP 7 berechnet die Kommunikationslast und damit die resultierenden Aktualisierungszeiten. Damit eine Berechnung bei Shared-Device-Konfigurationen möglich ist, müssen Sie in dem Projekt, bei dem die PROFINET-Schnittstelle des Shared Devices dem IO-Controller zugewiesen ist, die Anzahl der projekt-externen IO-Controller eingeben.

Die maximal mögliche Anzahl von IO-Controllern für das Shared Device ist geräteabhängig. Diese Anzahl ist in der GSD-Datei des Shared Device hinterlegt.

Bei einer S7-1500 CPU als IO-Controller können Sie einen sehr kurzen Sendetakt einstellen. Der Sendetakt darf kürzer sein als der kürzeste Sendetakt, den das Shared Device unterstützt.

Das Shared Device wird in diesem Fall vom IO-Controller mit einem Sendetakt betrieben, den es unterstützt (Sendetakt-Adaption).

Beispiel:

Eine S7-1500 CPU unterstützt Sendetakte ab 0,25 ms. Ein konfiguriertes IO-Device unterstützt ebenfalls Sendetakte ab 0,25 ms, ein anderes unterstützt Sendetakte ab 1 ms. In diesem Fall haben Sie die Möglichkeit, für die CPU den kurzen Sendetakt von 0,25 ms einzustellen. Die CPU betreibt das "langsame" IO-Device mit dem Sendetakt von z. B. 1 ms.

Wenn Sie einen der IO-Controller mit Zugriff auf das Shared Device nicht mit STEP 7 konfigurieren, dann ist diese Funktion nicht möglich. In diesem Fall müssen Sie den gemeinsamen Sendetakt in den Eigenschaften des Shared Device in STEP 7 ablesen und im Engineering-Tool für den IO-Controller eingeben.

Regeln für die Konfiguration

- IO-Controller, die das Shared Device nutzen, werden in verschiedenen Projekten angelegt. In jedem Projekt müssen Sie sicherstellen, dass das Shared Device in jeder Station genau gleich konfiguriert ist. Es darf immer nur ein IO-Controller vollen Zugriff auf ein Submodul haben. Inkonsistenzen in der Konfiguration führen zum Ausfall des Shared Device.
- E/A-Adressen eines Moduls bzw. Submoduls sind nur dann editierbar, wenn ein Modul bzw. Submodul dem IO-Controller im selben Projekt zugewiesen ist.
- Das Shared Device muss in jedem Projekt dieselben IP-Parameter und denselben Gerätenamen haben.
- Der Sendetakt muss gleich sein bei allen IO-Controllern, die Zugriff auf das Shared Device haben.
- Sie S7-Subnetz-ID des Subnetzes, an dem das Shared Device angeschlossen ist, muss in allen Projekten gleich sein.
- Nur wenn die PROFINET-Schnittstelle des Shared Device dem lokalen IO-Controller zugeordnet ist, sind folgende Funktionen möglich:
  - IRT-Betrieb
  - Priorisierter Hochlauf
  - Parametrierung der Port-Eigenschaften

Randbedingungen

Weil sich eine Shared-Device-Konfiguration auf mehrere Projekte verteilt, ergeben sich folgende Randbedingungen:

- In der Adressübersicht jedes IO-Controllers, der Zugriff auf ein Shared Device hat, fehlen die Adressen von Modulen bzw. Submodulen, die nicht diesem IO-Controller zugeordnet sind.
- Die nicht zugeordneten Module bzw. Submodule gehen bei der Konsistenzprüfung nicht in die Mengengerüst-Berechnung für das Shared Device ein. Sie müssen daher selbst überprüfen, dass die maximale Anzahl von Submodulen oder die maximale Anzahl von zyklischen IO-Daten für das Shared Device nicht überschritten wird.

  Hinweis

  Wenn über Shared Device nur ein Submodul einem IO Controller zugewiesen ist, wird der Name des Submoduls nicht im Display der CPU angezeigt. Erst wenn mehr als ein Submodul zugewiesen wurde, wird die Anzahl der konfigurierten Submodule im Display der CPU angezeigt.

- Projektierungsfehler wie die Zuweisung eines Moduls bzw. Submoduls zu mehreren IO-Controllern werden nicht von STEP 7 erkannt.
- CPUs, die mit einer Shared-Device-Konfiguration geladen werden, haben keine Information darüber, dass es sich beim IO-Device um ein Shared Device handelt. Module bzw. Submodule, die anderen IO-Controllern und damit anderen CPUs zugeordnet sind, fehlen in der geladenen Konfiguration. Diese Module bzw. Submodule werden daher weder im CPU-Webserver noch im CPU-Display angezeigt.

Siehe auch

IO-Controller anzeigen, die auf Module eines Shared Device zugreifen (Seite 3491)

I-Device als Shared Device konfigurieren (Seite 3124)

Shared Device konfigurieren (GSD-Projektierung)

Im Folgenden ist beschrieben, wie Sie ein Dezentrales Peripheriesystem als Shared Device mit STEP 7 ab Version V12, Servicepack 1 konfigurieren.

Grundsätzlich ist eine "verteilte" Konfiguration mit verschiedenen Engineering-Tools für verschiedene IO-Controller-Familien möglich. Die Beschreibung der Vorgehensweise orientiert sich aber ausschließlich an STEP 7 V12, Servicepack 1. Die Beschreibung beschränkt sich auf zwei IO-Controller der Familie S7-1500, die sich ein Shared Device teilen.

Zwei Projekte werden angelegt (Shared-Device-1 und Shared-Device-2) mit jeweils einem IO-Controller (PLC1 und PLC2). Das Shared Device müssen Sie in beiden Projekten anlegen, obwohl es sich physikalisch um ein und dasselbe IO-Device handelt.

Voraussetzung

- STEP 7 ab V12 Servicepack 1
- Im Kontext von STEP 7 V12, Servicepack 1: Nur CPU 15XX ab FW 1.1 als IO-Controller projektierbar (keine CPU 31X, CPU 41X oder CPU 12XX).
- Im Kontext anderer IO-Controller, die Zugriff auf ein Shared Device haben sollen: entsprechendes Engineering-Tool, z. B. STEP 7 V5.5, Servicepack 3 für CPU 31x-3 PN/DP.
- IO-Device unterstützt Funktion Shared Device, z. B. Interfacemodul IM 155-5 PN ST V2.0.
- GSD-Datei für die Projektierung des IO-Device als Shared Device ist installiert.

Vorgehen - Projekt 1 anlegen

Um das erste Projekt mit einem Shared Device anzulegen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7.
2. Legen Sie ein neues Projekt an mit dem Namen "Shared-Device-1".
3. Fügen Sie in die Netzsicht eine CPU 1513-1 PN aus dem Hardware-Katalog ein. Vergeben Sie den Namen "PLC1".
4. Fügen Sie ein IO-Device mit der Funktion "Shared Device" aus dem Hardware-Katalog ein (Hardware-Katalog: Weitere Feldgeräte > PROFINET IO > I/O).
5. Ordnen Sie dem IO-Device den IO-Controller "PLC1" zu.
6. Doppelklicken Sie auf das IO-Device und fügen Sie alle benötigten Module und Submodule aus dem Hardware-Katalog in die Geräteübersichtstabelle ein.
7. Parametrieren Sie die Module.
8. Speichern Sie das Projekt.

Vorgehen - Projekt 2 anlegen

Um das zweite Projekt mit einem Shared Device anzulegen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7 ein weiteres mal.
   Eine neue Instanz von STEP 7 wird geöffnet.
2. Legen Sie in der neuen Instanz ein neues Projekt an mit dem Namen "Shared-Device-2".
3. Fügen Sie in die Netzsicht eine CPU 1513-1 PN ein. Vergeben Sie den Namen "PLC2".
4. Kopieren Sie das IO-Device aus dem Projekt "Shared-Device-1" und fügen es in die Netzsicht von Projekt "Shared-Device-2" ein.
5. Ordnen Sie dem IO-Device den IO-Controller "PLC2" zu.
6. Speichern Sie das Projekt.

Beide Projekte haben jetzt ein identisch aufgebautes IO-Device, das im nächsten Schritt für die verschiedenen IO-Controller-Zugriffe parametriert werden muss.

Vorgehen - Zugriffe auf das Shared Device parametrieren

Die Module und Submodule, die Sie in das Shared Device stecken, sind automatisch der lokalen CPU zugeordnet. Um die Zuordnung zu ändern, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht oder in der Gerätesicht von Projekt "Shared-Device-1" das Interfacemodul.
2. Wählen Sie den Bereich "Shared Device" im Inspektorfenster.
   Eine Tabelle zeigt für alle projektierten Module, welche CPU Zugriff auf das jeweilige Modul bzw. Submodul hat. Voreingestellt ist für alle Module und Submodule, dass die lokale CPU Zugriff hat.
3. Für alle Module bzw. Submodule, die im Adressbereich der lokalen CPU bleiben sollen, belassen Sie die Einstellung "PLC1"
   Für alle Module bzw. Submodule, die im Adressbereich der CPU aus dem Projekt "Shared-Device-2" liegen sollen ("PLC2"), wählen Sie die Einstellung "---". Das bedeutet, dass ein IO-Controller außerhalb des Projekts Zugriff auf das Modul bzw. Submodul haben soll.
4. Markieren Sie in der Netzsicht oder Gerätesicht von Projekt "Shared-Device-2" das Interfacemodul.
5. Wählen Sie den Bereich "Shared Device" im Inspektorfenster.
   Eine Tabelle zeigt für alle projektierten Module, welche CPU Zugriff auf das jeweilige Modul bzw. Submodul hat.
6. Für alle Module bzw. Submodule, die im Adressbereich der CPU aus dem Projekt "Shared-Device-1" liegen sollen ("PLC1"), wählen Sie die Einstellung "---".
7. Prüfen Sie abschließend, ob für jedes Modul bzw. Submodul in beiden Projekten die Einstellungen für den Zugriff "komplementär" sind. Das bedeutet, wenn in einem Projekt die lokale CPU Zugriff hat, muss im anderen Projekt die Option "---" eingestellt sein und umgekehrt.
   Besonderheit: Die Option "---" für die PROFINET-Schnittstelle und damit auch der Ports bewirkt, dass die zugehörigen Parameter deaktiviert und nicht änderbar sind. Parameter der PROFINET-Schnittstelle und Port-Parameter sind nur in dem Projekt editierbar, in dem die PROFINET-Schnittstelle der lokalen CPU zugeordnet ist. Unabhängig davon lassen sich die Ports in beiden Projekten verschalten.
8. Prüfen Sie, ob in allen Projekten für das Shared Device dieselben IP-Adressparameter und Gerätenamen eingestellt sind.
   Prüfen Sie, ob in allen Projekten dieselbe S7-Subnetz-ID eingestellt ist für das Subnetz, an dem das Shared Device angeschlossen ist (Eigenschaften Subnetz, Bereich "Allgemein" im Inspektorfenster).

Hinweis

Falls Sie Änderungen am Shared Device vornehmen: Führen Sie die Änderungen in jedem Projekt am Shared Device nach! Achten Sie darauf, dass immer nur ein IO-Controller Zugriff auf ein Modul bzw. Submodul hat!

Vorgehen - Echtzeit-Einstellungen anpassen

Damit sichergestellt ist, dass alle IO-Controller und Shared Devices mit dem passenden Sendetakt betrieben werden und die Aktualisierungszeiten aufgrund der Kommunikationslast richtig berechnet werden, müssen Sie folgende Einstellungen anpassen und überprüfen:

1. Wählen Sie das Projekt, dessen IO-Controller Zugriff auf die PROFINET-Schnittstelle und die Ports des Shared Devices haben.
2. Markieren Sie das Interfacemodul des Shared Devices in der Netzansicht.
3. Navigieren Sie im Inspektorfenster zum Bereich "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > IO-Zyklus".
4. Stellen Sie im Bereich "Shared Device" die Anzahl der projekt-externen IO-Controller ein. Die maximale Anzahl ist abhängig vom IO-Device (Vorgabe in GSD-Datei).
5. Für jeden IO-Controller, der Zugriff auf Module bzw. Submodule des Shared Device hat, müssen Sie denselben Sendetakt einstellen:
   - Wenn Sie den IO-Controller mit STEP 7 (TIA Portal) projektieren:
     Öffnen Sie das entsprechende Projekt.
     Markieren Sie die PROFINET-Schnittstelle des IO-Controllers.
     Wählen Sie im Inspektorfenster den Bereich "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Kommunikation" und stellen den gemeinsamen Sendetakt ein.
   - Wenn Sie den IO-Controller mit einem anderen Engineering-Tool projektieren:
     Markieren Sie die PROFINET-Schnittstelle des Shared Device in STEP 7 (TIA Portal) und lesen den Sendetakt am Shared Device ab (Bereich "Erweiterte Optionen > Echtzeit-Einstellungen")
     Geben Sie im Engineering-Tool den abgelesenen Sendetakt ein.
   - Besonderheit: Wenn Sie **alle** IO-Controller, die Zugriff auf das Shared Device haben, in STEP 7 (TIA Portal oder V5.5) projektieren, dann lassen sich auch kürzere Sendetakte am IO-Controller einstellen, als das Shared Device unterstützt (Sendetakt-Adaption).

Übersetzen und Laden

Die Konfigurationen für die verschiedenen IO-Controller müssen Sie übersetzen und nacheinander in die CPUs laden.

Aufgrund der verteilten Konfiguration mit separaten Projekten gibt STEP 7 keine Konsistenzfehler aus bei fehlerhafter Zugriffs-Parametrierung. Beispiele für fehlerhafte Zugriffs-Parametrierung:

- Mehrere IO-Controller haben Zugriff auf dasselbe Modul
- IP-Adressparameter oder Sendetakte sind nicht identisch

Diese Fehler wirken sich erst im Betrieb aus und werden z. B. als Konfigurationsfehler gemeldet.

I-Device als Shared Device konfigurieren

Im Folgenden ist der Sonderfall beschrieben, wie Sie eine S7-1500 als I-Device mit STEP 7 ab Version V13 konfigurieren und anschließend in zwei Projekten als Shared Device verwenden.

Grundsätzlich ist auch hier eine "verteilte" Konfiguration mit verschiedenen Engineering-Tools für verschiedene IO-Controller-Familien möglich. Die im Folgenden beschriebene Vorgehensweise orientiert sich an STEP 7 V13 und beschränkt sich auf eine Konfiguration mit zwei IO-Controllern der Familie S7-1500, die sich die Transferbereiche eines I-Device als Shared Device teilen. Das I-Device selbst ist ebenfalls eine S7-1500-CPU.

Zunächst ist das allgemeine Konzept beschrieben.

In den weiteren Abschnitten wird anhand eines Beispiels die schrittweise Vorgehensweise erläutert: Drei Projekte werden angelegt mit jeweils einem IO-Controller (PLC-I-Device, PLC_1 und PLC_2).

PLC-I-Device wird verwendet, um das I-Device zu konfigurieren. Die PROFINET-GSD-Variante von PLC-I-Device wird in den Projekten PLC_1 und PLC_2 verwendet, um die Transferbereiche im jeweils übergeordneten IO-Controller zuordnen zu können.

Shared-I-Device-Konzept

Zur Einführung des Shared-I-Device-Konzepts werden hier mit zwei Rollen eingeführt:

- Die Rolle eines Herstellers (z. B. Maschinenhersteller): Er konfiguriert und programmiert ein I-Device, das eine bestimmte Automatisierungsaufgabe übernimmt. Als Peripherieschnittstelle zum Betreiber der Maschine werden Transferbereiche definiert, die unterschiedlichen IO-Controllern zugeordnet werden können. Für die Anbindung an übergeordnete IO-Controller stellt der Hersteller eine PROFINET-GSD-Datei zur Verfügung und legt die Transferbereiche offen, über die auf das I-Device zugegriffen werden kann.
- Die Rolle des Betreibers: Er verwendet das I-Device als PROFINET-GSD-Variante in der Projektierung seines PROFINET IO-Systems und legt dabei die E/A-Adressen fest, unter denen die IO-Controller auf die Transferbereiche zugreifen.

Herstellersicht

Eine S7-1500-CPU parametrieren Sie als I-Device mit zentraler und dezentraler Peripherie. Wichtig sind folgende Einstellungen:

- gewünschte Transferbereiche
- Anzahl IO-Controller mit Zugriff auf dieses I-Device (bei Shared Device immer größer als 1!)

Besonderheit: Das I-Device wird ohne übergeordneten IO-Controller konfiguriert. Damit stehen nur die lokalen E/A-Adressen des Transferbereichs zur Verfügung (= "Adresse im I-Device"), um das Anwenderprogramm zur Bearbeitung der Adressen aus dem Transferbereich zur erstellen.

Das bis auf die Anbindung zum übergeordneten IO-Controller vollständig projektierte I-Device wird in die S7-1500-CPU geladen.

Aus der I-Device-Konfiguration exportieren Sie eine PROFINET-GSD-Datei.

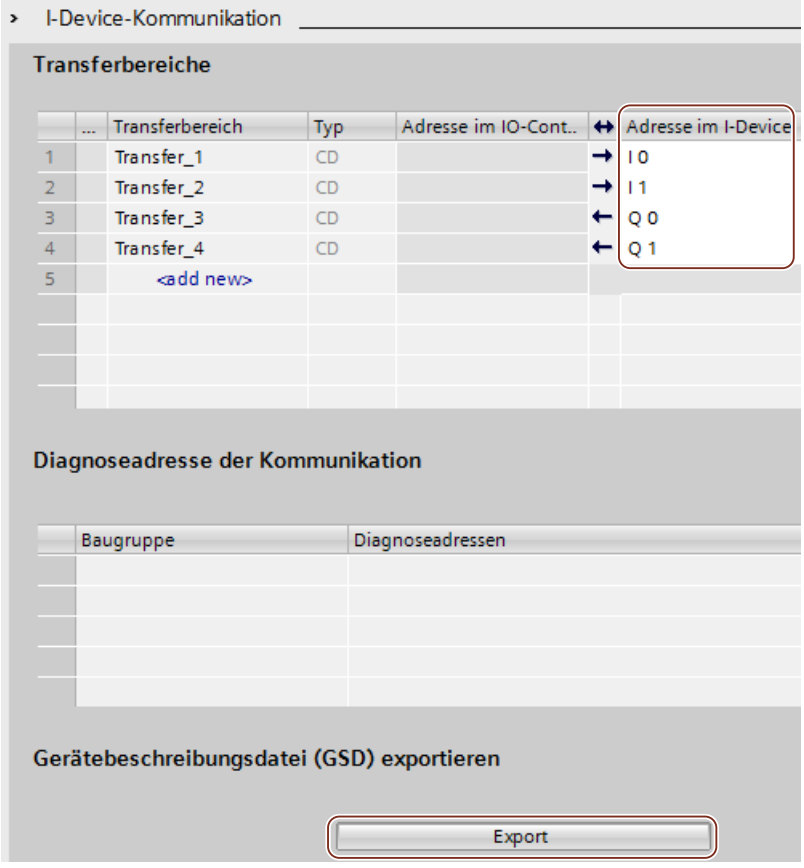

Betreibersicht

Die aus der I-Device-Konfiguration erzeugte PROFINET-GSD-Datei müssen Sie in allen Engineering-Systemen installieren, die an der Projektierung eines PROFINET IO-Systems mit diesem Shared-I-Device beteiltigt sind. Wenn alle Verwendungen dieses I-Devices mit STEP 7 V13 projektiert werden, genügt die Installation der GSD-Datei in STEP 7.

Jeder übergeordnete IO-Controller mit zugeordnetem Shared I-Device-Transferbereich in einem eigenen Projekt

Das I-Device konfigurieren Sie als GSD-Variante am PROFINET IO-System in den beteiligten Projekten. In STEP 7 V13 ist dieses I-Device nach der Installation unter "Weitere Feldgeräte > PROFINET IO > PLCs & CPs" zu finden.

Für jede Zuordnung zwischen dem Shared I-Device und einem der übergeordneten IO-Controller, die sich die Tranferbereiche teilen, benötigen Sie ein eigenes Projekt.

Das folgende Bild zeigt die Zuordung des Shared I-Device zum übergeordneten IO-Controller PLC_1.

Für jeden übergeordneten IO-Controller müssen Sie jeweils ein eigene Projekt mit demselben Shared I-Device erstellen und dem IO-Controller den Zugriff auf den gewünschten Teil-Transferbereich gewähren.

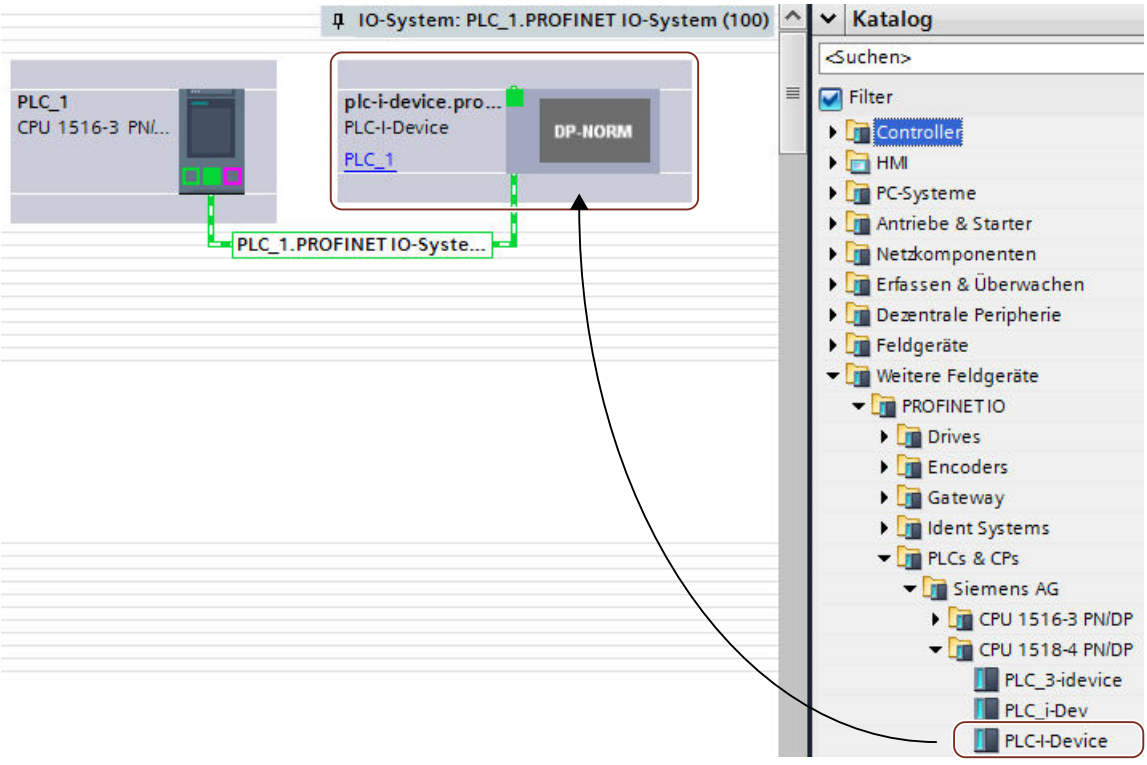

In jedem der beteiligten Projekte bestimmen Sie, welche Transferbereiche exklusiv dem übergeordneten IO-Controller zugwiesen sind (voreingestellt: alle). Die übrigen Transferbereiche stellen Sie auf "---" (nicht zugewiesen). Diese Einstellung bewirkt, dass der lokale IO-Controller keinen Zugriff auf diesen Transferbereich hat und daher in einem anderen Projekt einem anderen IO-Controller zugeordnet werden kann.

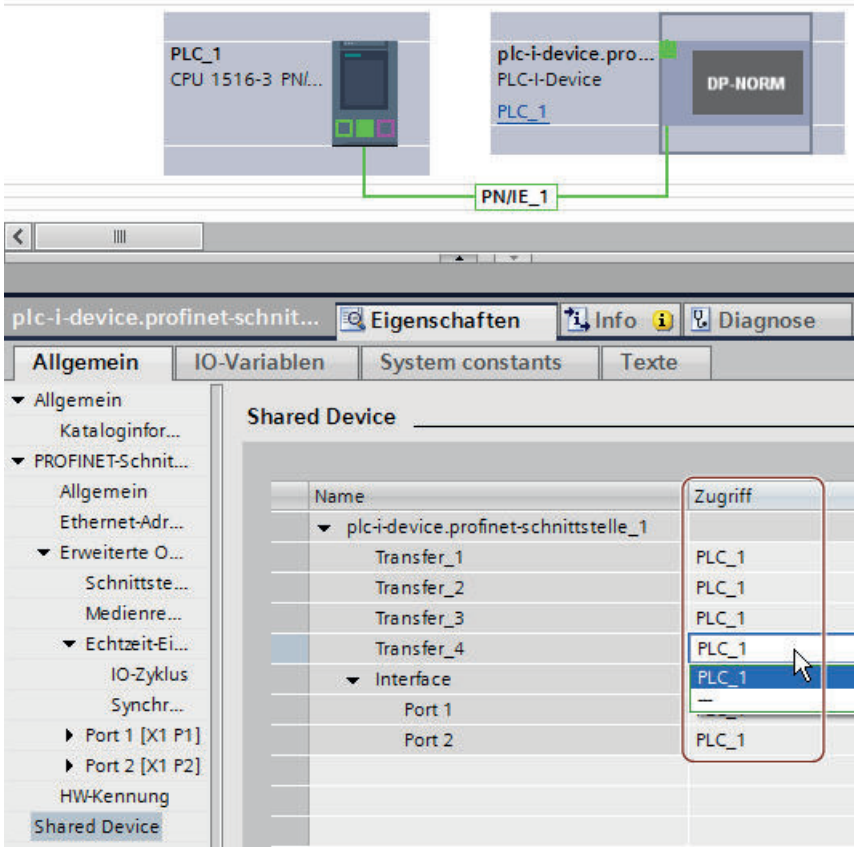

Die Adressen aus Sicht des IO-Controllers passen Sie in der Geräteübersicht an. Die Geräteübersicht öffnen Sie, indem Sie auf das I-Device doppelklicken.

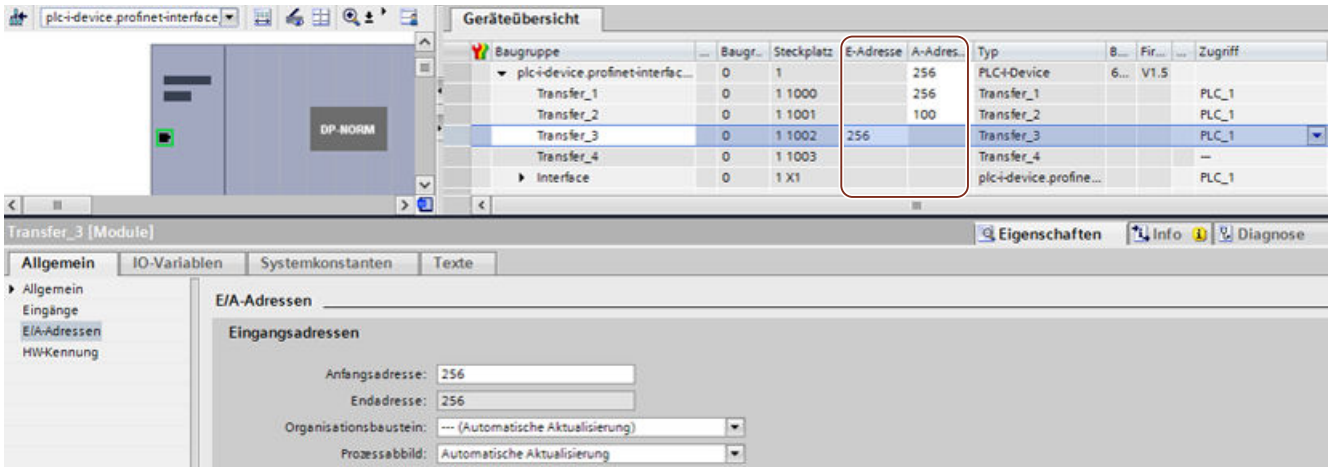

Voraussetzung

- STEP 7 ab V13
- Im Kontext von STEP 7 ab V13: Nur CPU 15XX ab FW 1.5 als Shared I-Device projektierbar (keine CPU 31X, CPU 41X oder CPU 12XX).
- Im Kontext anderer IO-Controller, die Zugriff auf ein Shared I-Device haben sollen: entsprechendes Engineering-Tool, z. B. STEP 7 V5.5, Servicepack 3 für CPU 31x-3 PN/DP.

Vorgehen - Projekt PLC-I-Device anlegen

Um das Projekt mit einem Shared-I-Device anzulegen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7.
2. Legen Sie ein neues Projekt an mit dem Namen "PLC-I-Device".
3. Fügen Sie in die Netzsicht eine CPU 1518-4 PN/DP aus dem Hardware-Katalog ein. Vergeben Sie den Namen "PLC-I-Device".
4. Doppelklicken Sie auf das IO-Device und konfigurieren Sie alle benötigten Module und Submodule.

5. Parametrieren Sie die Module.
   Speziell für die CPU sind folgende Einstellungen im Bereich der PROFINET-Schnittstelle [X1] notwendig:
   - Aktivieren Sie die Option "IO-Device" im Bereich "Betriebsart".
   - Konfigurieren Sie die Transferbereiche im Bereich "Betriebsart" > "I-Device-Konfiguration". Die Spalte "Adresse im IO-Controller" bleibt dabei leer, da kein IO-Controller zugewiesen ist.
     Hinweis: Um einen Eingangsbereich in einen Ausgangsbereich zu ändern und umgekehrt, müssen Sie in den Bereich des entsprechenden Transferbereichs navigieren.

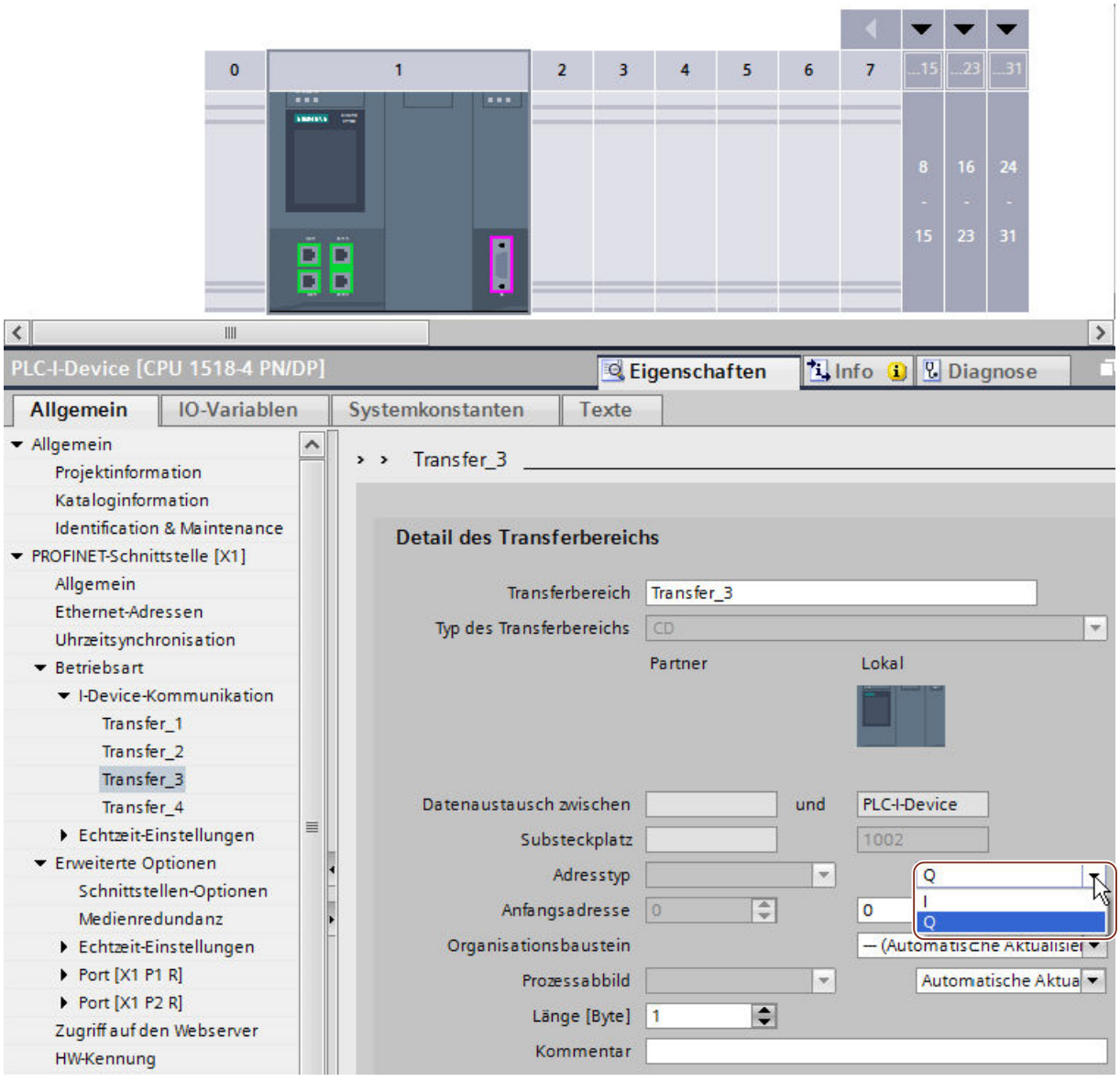

   - Wählen Sie die Anzahl der IO-Controller, die im Berieb auf das Shared-I-Device zugreifen werden (Bereich "Betriebsart" > "Echtzeit-Einstellungen", Abschnitt "Shared Device").

6. Speichern Sie das Projekt.
7. Klicken Sie auf die Schaltfläche "Export" (Bereich "Betriebsart" > "I-Device-Konfiguration", Abschnitt "Gerätebeschreibungsdatei (GSD) exportieren").
   Falls Sie den Namen im Export-Dialog nicht ändern, hat die GSD-Datei z. B. einen Namen der Form "GSDML-V2.31-#Siemens-PreConf_PLC-I-Device-20130925-123456".

Vorgehen - Projekt PLC_1 anlegen

Um das erste Projekt mit einem Shared-I-Device anzulegen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7.
2. Installieren Sie die PROFINET-GSD-Datei aus dem Export der I-Device-CPU (PLC-I-Device).
3. Legen Sie ein neues Projekt an mit dem Namen "PLC_1".
4. Fügen Sie in die Netzsicht eine CPU 1516-3 PN/DP ein. Der Name der CPU sollte "PLC_1" sein.
5. Fügen Sie das I-Device aus dem Hardware-Katalog ein (Hardware-Katalog: Weitere Feldgeräte > PROFINET IO > PLCs & CPs).
6. Ordnen Sie dem I-Device den IO-Controller "PLC_1" zu.
7. Wählen Sie in den Eigenschaften des I-Devices den Bereich "Shared Device".
   In der Tabelle sind alle Transferbereiche und die PROFINET-Schnittstelle dem lokalen IO-Controller zugeordnet (PLC_1).
8. Bestimmen Sie, auf welche Transferbereiche die CPU PLC_1 **keinen** Zugriff haben soll. Für diese Bereiche wählen Sie den Eintrag "---". Diese Transferbereiche sind für PLC_2 vorgesehen.
9. Speichern Sie das Projekt.

Vorgehen - Projekt PLC_2 anlegen

Um das zweite Projekt mit einem Shared Device anzulegen, gehen Sie folgendermaßen vor:

1. Starten Sie STEP 7 ein weiteres mal.
   Eine neue Instanz von STEP 7 wird geöffnet.
2. Legen Sie in der neuen Instanz ein neues Projekt an mit dem Namen "PLC_2".
3. Fügen Sie in die Netzsicht eine CPU 1516-3 PN/DP ein. Vergeben Sie den Namen "PLC_2".
4. Fügen Sie das I-Device aus dem Hardware-Katalog ein (Hardware-Katalog: Weitere Feldgeräte > PROFINET IO > PLCs & CPs).
5. Ordnen Sie dem I-Device den IO-Controller "PLC_2" zu.
6. Passen Sie wie im Projekt PLC_1 die Zugriffe auf die Transferbereiche an. Beachten Sie, dass keine doppelten Zuordnungen entstehen.
7. Passen Sie die Parameter von Subnetz und PROFINET-Schnittstelle an. Da es sich beim Shared-I-Device um dasselbe Gerät in verschiedenen Projekten handelt, müssen diese Daten übereinstimmen.
8. Speichern Sie das Projekt.

Beide Projekte haben jetzt ein identisch aufgebautes Shared-I-Device. Die IO-Controller-Zugriffe und die Parameter der PROFINET-Schnittstelle sollten im nächsten Schritt in den verschiedenen Projekten noch kontrolliert werden.

Zusammenfassung - Zugriffe auf das Shared Device parametrieren

Die Transferbereiche sind automatisch dem lokalen IO-Controller zugeordnet. Um die Zuordnung zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Netzsicht von Projekt "PLC_1" das Gerät "PLC_I-Device" und wählen Sie den Bereich "Shared Device".
2. Eine Tabelle zeigt für alle projektierten Transferbereiche, welche CPU Zugriff auf den jeweiligen Tranferbereich hat. Voreingestellt ist für alle Module und Submodule, dass die lokale CPU Zugriff hat.
3. Für alle Transferbereiche, die im Adressbereich der lokalen CPU bleiben sollen, belassen Sie die Einstellung "PLC_1"
   Für alle Transferbereiche, die im Adressbereich der CPU "PLC_2" aus dem Projekt "PLC_2" liegen sollen, wählen Sie die Einstellung "---". Das bedeutet, dass ein IO-Controller außerhalb des Projekts Zugriff auf den Transferbereich haben soll.
4. Verfahren Sie mit den übrigen Projekten entsprechend.
5. Prüfen Sie abschließend, ob für jedes Modul bzw. Submodul in beiden Projekten die Einstellungen für den Zugriff "komplementär" sind. Das bedeutet, wenn in einem Projekt die lokale CPU Zugriff hat, muss im anderen Projekt die Option "---" eingestellt sein und umgekehrt.
   Besonderheit: Die Option "---" für die PROFINET-Schnittstelle und damit auch der Ports bewirkt, dass die zugehörigen Parameter deaktiviert und nicht änderbar sind. Parameter der PROFINET-Schnittstelle und Port-Parameter sind nur in dem Projekt editierbar, in dem die PROFINET-Schnittstelle der lokalen CPU zugeordnet ist. Unabhängig davon lassen sich die Ports in beiden Projekten verschalten.
6. Prüfen Sie, ob in allen Projekten für das Shared Device dieselben IP-Adressparameter und Gerätenamen eingestellt sind.
   Prüfen Sie, ob in allen Projekten dieselbe S7-Subnetz-ID eingestellt ist für das Subnetz, an dem das Shared Device angeschlossen ist (Eigenschaften Subnetz, Bereich "Allgemein" im Inspektorfenster).

Hinweis

Falls Sie Änderungen am I-Device vornehmen (z. B. Anzahl oder Länge der Transferbereiche ändern):

Exportieren Sie das I-Device erneut als GSD-Datei. Installieren Sie die GSD-Datei erneut in jedem Projekt, die das I-Device als Shared Device nutzen! Achten Sie darauf, dass immer nur ein IO-Controller Zugriff auf einen Transferbereich hat!

Vorgehen - Echtzeit-Einstellungen anpassen

Damit sichergestellt ist, dass alle IO-Controller und Shared Devices mit dem passenden Sendetakt betrieben werden und die Aktualisierungszeiten aufgrund der Kommunikationslast richtig berechnet werden, müssen Sie folgende Einstellungen anpassen und überprüfen:

1. Für jeden IO-Controller, der Zugriff auf Module bzw. Submodule des Shared Device hat, müssen Sie denselben Sendetakt einstellen:
   - Wenn Sie den IO-Controller mit STEP 7 (TIA Portal) projektieren:
     Öffnen Sie das entsprechende Projekt.
     Markieren Sie die PROFINET-Schnittstelle des IO-Controllers.
     Wählen Sie im Inspektorfenster den Bereich "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Kommunikation" und stellen den gemeinsamen Sendetakt ein.
   - Wenn Sie den IO-Controller mit einem anderen Engineering-Tool projektieren:
     Markieren Sie die PROFINET-Schnittstelle des Shared Device in STEP 7 (TIA Portal) und lesen den Sendetakt am Shared Device ab (Bereich "Erweiterte Optionen > Echtzeit-Einstellungen")
     Geben Sie im Engineering-Tool den abgelesenen Sendetakt ein.
   - Besonderheit: Wenn Sie **alle** IO-Controller, die Zugriff auf das Shared-I-Device haben, in STEP 7 (TIA Portal oder V5.5) projektieren, dann lassen sich auch kürzere Sendetakte am IO-Controller einstellen, als das Shared Device unterstützt (Sendetakt-Adaption).

Übersetzen und Laden

Die Konfigurationen für die verschiedenen IO-Controller müssen Sie übersetzen und nacheinander in die CPUs laden.

Aufgrund der verteilten Konfiguration mit separaten Projekten gibt STEP 7 keine Konsistenzfehler aus bei fehlerhafter Zugriffs-Parametrierung. Beispiele für fehlerhafte Zugriffs-Parametrierung:

- Mehrere IO-Controller haben Zugriff auf dasselbe Modul
- IP-Adressparameter oder Sendetakte sind nicht identisch

Diese Fehler wirken sich erst im Betrieb aus und werden z. B. als Konfigurationsfehler gemeldet.

Siehe auch

I-Device projektieren (Seite 3076)

Wissenswertes zu Shared Devices (Seite 3116)

Modulinternes Shared Input/Shared Output (MSI/MSO)

Einleitung

Dieses Kapitel beschreibt die Funktionalität Modulinternes Shared Input/Shared Output (MSI/MSO) für die S7-1500 Peripheriemodule, die in einer ET 200MP an PROFINET betrieben werden.

Funktionalität Modulinternes Shared Input/Shared Output

Die Funktion Modulinternes Shared Input (MSI) erlaubt, dass ein Eingabemodul seine Eingangsdaten bis zu vier IO-Controllern zur Verfügung stellt. Jeder Controller kann auf dieselben Kanäle lesend zugreifen.

Die Funktion Modulinternes Shared Output (MSO) erlaubt, dass ein Ausgabemodul seine Ausgangsdaten bis zu vier IO-Controllern zur Verfügung stellt. Ein IO-Controller besitzt Schreibzugriff. Die weiteren IO-Controller können auf dieselben Kanäle lesend zugreifen.

Das folgende Bild zeigt die Funktionalität MSI/MSO.

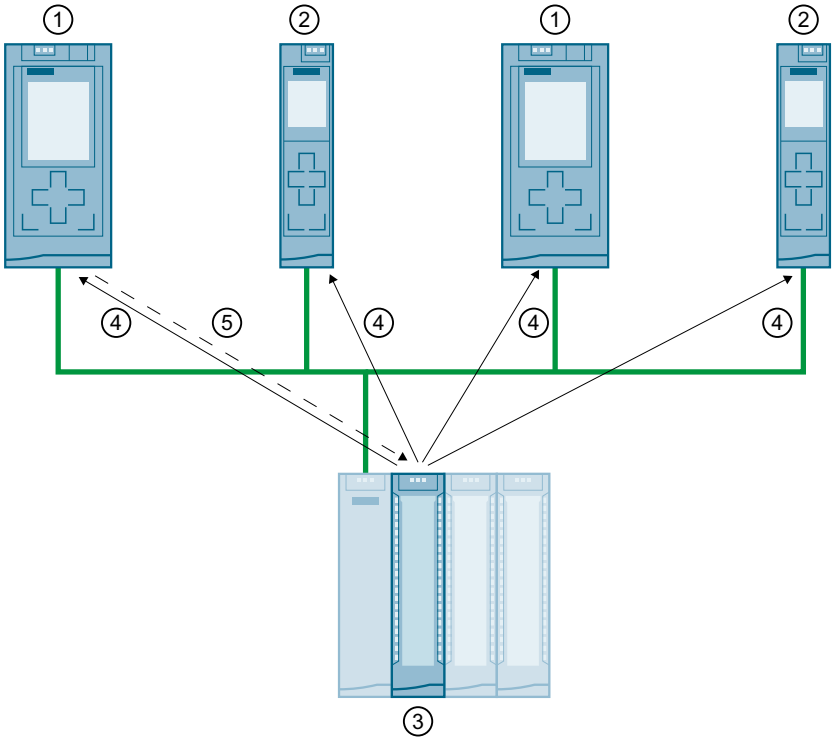

① CPU 1516-3 PN/DP als IO-Controller
② CPU 1511-1 PN als IO-Controller
③ I/O-Modul mit MSI/MSO
④ Lesezugriff auf die Kanäle des I/O-Moduls
⑤ Schreibzugriff auf die Kanäle des I/O-Moduls (nur bei MSO)

Vorteile von MSI/MSO

Modulinternes Shared Input/Modulinternes Shared Output bietet folgende Vorteile:

- Echtzeiterfassung in mehreren CPUs
- Geringere Kosten durch Einsparung an zusätzlichen IO-Devices und Modulen
- Geringerer Platzbedarf durch Einsparung an zusätzlichen IO-Devices und Modulen
- Geringere Kommunikationslast durch wegfallende CPU-CPU-Kommunikation
- Kein zusätzlicher Programmieraufwand für CPU-CPU-Kommunikation notwendig

Voraussetzungen für den Einsatz von MSI/MSO

Beachten Sie folgende Voraussetzungen:

- Der Einsatz von MSI/MSO ist nur bei PROFINET IO möglich
- Projektiersoftware: STEP 7 (TIA Portal) ab V12 SP1 mit GSD-Datei.
- Das Interfacemodul IM 155-5 PN ST und die Module unterstützen MSI/MSO ab Firmware-Version V2.0.0.

Randbedingungen beim Einsatz von MSI/MSO

Beachten Sie folgende Randbedingungen:

- Bei Modulen mit Kanalgruppen ist der Einsatz von MSI/MSO nicht möglich.
- Module mit MSI/MSO sind nicht im taktsynchronen Betrieb einsetzbar.

MSI-Submodule

Bei der MSI-Konfiguration eines Eingangsmoduls werden die Eingangswerte aller Kanäle in ein Basis-Submodul und bis zu drei weitere MSI-Submodule kopiert. Die Kanäle des Moduls sind dann mit identischen Eingangswerten im Basis-Submodul und den MSI-Submodulen vorhanden. Die Submodule können beim Einsatz des Moduls in einem Shared Device bis zu vier IO-Controllern zugewiesen werden. Jeder IO-Controller kann auf dieselben Kanäle lesend zugreifen.

Das folgende Bild zeigt ein digitales Eingabemodul mit dem Basis-Submodul und drei MSI-Submodulen. Jedes Submodul ist einem IO-Controller zugeordnet. Die Diagnose und Parametrierung des Digitalen Eingabemoduls ist über das Basis-Submodul vom IO-Controller 1 aus möglich.

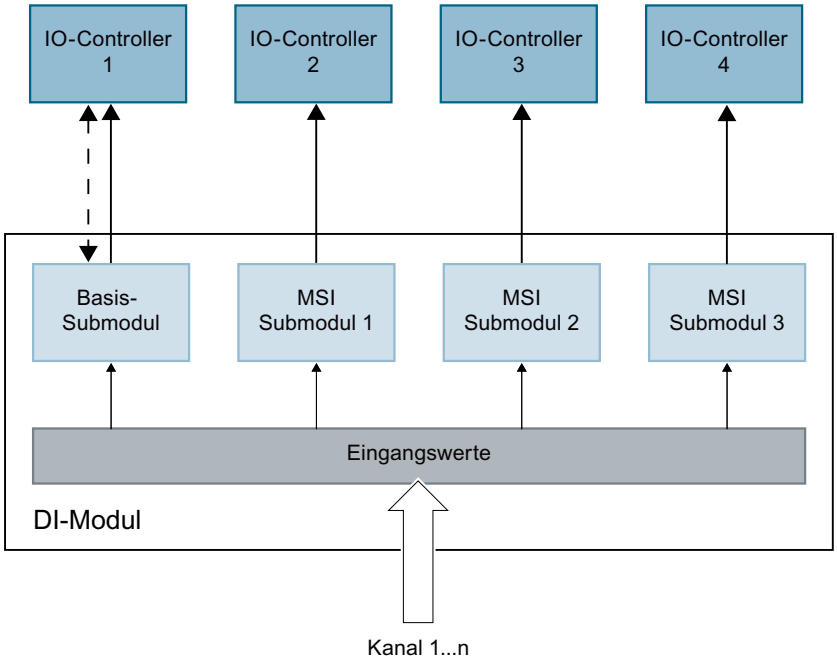

Wertstatus des MSI-Submoduls (Quality Information, QI)

Die Bedeutung des Wertstatus hängt davon ab, um welches Submodul es sich handelt.

Beim Basis-Submodul (= 1. Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist.

Bei einem MSI-Submodul (2. bis 4. Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist oder dass das Basis-Submodul noch nicht parametriert ist (nicht betriebsbereit).

MSO-Submodule

Bei der MSO-Konfiguration eines Ausgangsmoduls werden die Ausgabewerte aller Kanäle des Moduls von einem Basis-Submodul in bis zu drei weitere MSO-Submodule kopiert. Die Kanäle des Moduls sind dann mit identischen Werten im Basis-Submodul und den MSO-Submodulen vorhanden. Die MSO-Submodule können beim Einsatz des Moduls in einem Shared Device bis zu vier IO-Controllern zugewiesen werden:

- Der IO-Controller, dem das Basis-Submodul zugewiesen ist, hat schreibenden Zugriff auf die Ausgänge des Moduls. Das Basis-Submodul belegt daher Ausgangsadressen im Prozessabbild des IO-Controllers.
- Die IO-Controller, denen die MSO-Submodule zugewiesen sind, haben lesenden Zugriff auf die Ausgänge des Moduls. MSO-Submodule belegen daher Eingangsadressen im Prozessabbild der IO-Controller.

Das folgende Bild zeigt ein digitales Ausgabemodul mit dem Basis-Submodul und drei MSO-Submodulen. Jedes Submodul ist einem IO-Controller zugeordnet. Die Diagnose und Parametrierung des digitalen Ausgabemoduls ist über das Basis-Submodul vom IO-Controller 1 aus möglich.

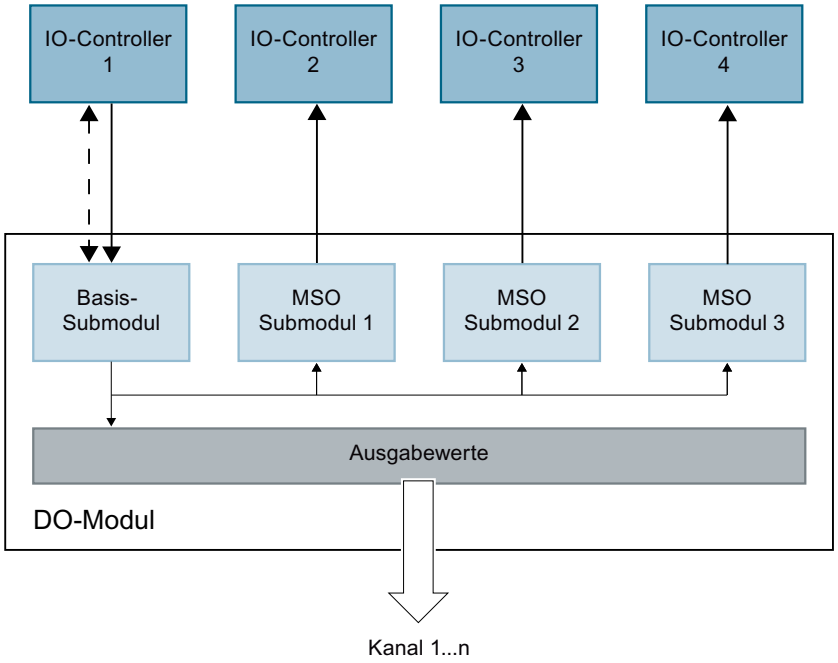

Wertstatus des MSO-Submoduls (Quality Information, QI)

Die Bedeutung des Wertstatus hängt davon ab, um welches Submodul es sich handelt.

Beim Basis-Submodul (=1. Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist.

Bei einem MSO-Submodul (=2. bis 4. Submodul) zeigt der Wertstatus 0 an, dass der Wert fehlerhaft ist oder dass einer der folgenden Fehler aufgetreten ist:

- Das Basis-Submodul ist noch nicht parametriert (nicht betriebsbereit).
- Die Verbindung zwischen IO-Controller und Basis-Submodul ist unterbrochen.
- Der IO-Controller vom Basis-Submodul ist im STOP-Zustand oder NETZ-AUS.

Peripheriemodule mit MSI/MSO-Submodulen konfigurieren

Vorraussetzung

- Projektiersoftware STEP 7 ab V12 Servicepack 1
- GSD-Datei für Peripheriemodule mit MSI/MSO.
- IO-Device unterstützt MSI/MSO (z. B. IM 155-5 PN ST ab Firmware-Version V2.0.0)

Vorgehen

1. Fügen Sie in der Netzsicht von STEP 7 ein Interfacemodul IM 155-5 PN ST ab V2.0 ein.
2. Doppelklicken Sie auf das IO-Device.
   Sie befinden sich jetzt in der Gerätesicht.
3. Platzieren Sie Module mit MSI/MSO aus dem Hardware-Katalog in einem passenden Steckplatz.
4. Konfigurieren Sie die Module, wie in der Produktinformation zum Modul beschrieben.

Serienmaschinen-Projekt erstellen

Übersicht

Einleitung

Serienmaschinen-Projekte sind STEP 7-Projekte, die ein Set von innovativen Funktionen nutzen, um flexible Automatisierungslösungen für Serienmaschinen bzw. für modular aufgebaute Maschinen einfach projektieren und in Betrieb nehmen zu können.

Eine Hardware-Konfiguration, bestehend aus einer S7-1500 CPU als IO-Controller und aus beliebigen angeschlossenen IO-Devices, repräsentiert dabei einen „PROFINET IO-System-Master“. Dieser Master ist mit einer Maximalkonfiguration projektiert, aus der sich unterschiedliche Optionen ableiten lassen für unterschiedliche Serienmaschinen mit z. B. verschiedenen Aufbauvarianten des IO-Systems.

Flexibilisierung auf allen Ebenen

Serienmaschinen-Projekte besitzen folgende zentrale Merkmale:

- Aus genau einem Projekt (IO-System-Master) mit projektierter Maximalkonfiguration lassen sich verschiedene Varianten einer Serienmaschine (IO-System-Optionen) laden. Das Serienmaschinen-Projekt deckt alle Varianten (Optionen) des IO-Systems ab.
- Eine IO-System-Option lässt sich vor Ort mit einfachen Mitteln in ein bestehendes Netzwerk einbinden.

Flexibilität ist in mehrfacher Hinsicht gegeben:

- Durch entsprechende Projektierung ist eine Anpassung der IP-Adressparameter des IO-Controllers vor Ort mit einfachen Tools möglich. Auf diese Weise lässt sich eine Serienmaschine aufwandsarm in unterschiedliche Anlagen integrieren oder mehrfach in einem Netzwerk einbinden.
  IO-Systeme mit dieser Eigenschaft bezeichnen wir als **mehrfach einsetzbare IO-Systeme**.
- Durch entsprechende Projektierung und Programmierung lassen sich vor Ort unterschiedlich aufgebaute IO-System-Optionen betreiben, die sich hinsichtlich der Auswahl eingesetzter IO-Devices oder hinsichtlich der Anordnung der IO-Devices unterscheiden.
  Da die konkrete Konfiguration des IO-Systems über das Anwenderprogramm gesteuert wird, bezeichnen wir dies als **Konfigurationssteuerung für IO-Systeme**.
- Unabhängig von den oben beschriebenen Funktionen können Sie durch entsprechende Projektierung und Programmierung in einem einzigen Projekt unterschiedliche Stationsoptionen von Zentralgeräten bzw. von dezentralen Peripheriegeräten betreiben. Die Geräte dürfen sich hinsichtlich der Auswahl und Anordnung der Module unterscheiden. Da die konkrete Konfiguration der Station über das Anwenderprogramm gesteuert wird, bezeichnen wir dies auch als **Konfigurationssteuerung**.

Mehrfach einsetzbare PROFINET IO-Systeme

Wissenswertes zu mehrfach einsetzbaren IO-Systemen

Mehrfach einsetzbare Automatisierungslösungen

Für eine flexibel nutzbare Automatisierungslösung, wie das bei Serienmaschinen der Fall ist, gibt es folgende typische Anwendungsfälle:

- Die Maschine (und damit das PROFINET IO-System) wird bei Kunden mehrfach eingesetzt.
- Die Maschine wird in verschiedenen Anlagen bei verschiedenen Kunden eingesetzt.

Dazu muss die Automatisierungslösung folgenden Anforderungen genügen:

- Ein (Konfiguration und Programm) läßt sich ohne Änderungen auf verschiedene Maschinen gleichen Typs laden.
- Vor Ort sind nur noch wenige aufwandsarme Anpassungen notwendig, um die Maschine in eine bestehende Netzwerk-Infrastruktur einzubinden.

Das folgende Bild zeigt, wie eine Automatisierungslösung mit mehrfach einsetzbarem IO-System in verschiedene Automatisierungssysteme geladen wird und anschließend ein Automatisierungssystem vor Ort an die bestehende Netzinfrastruktur angepasst wird.

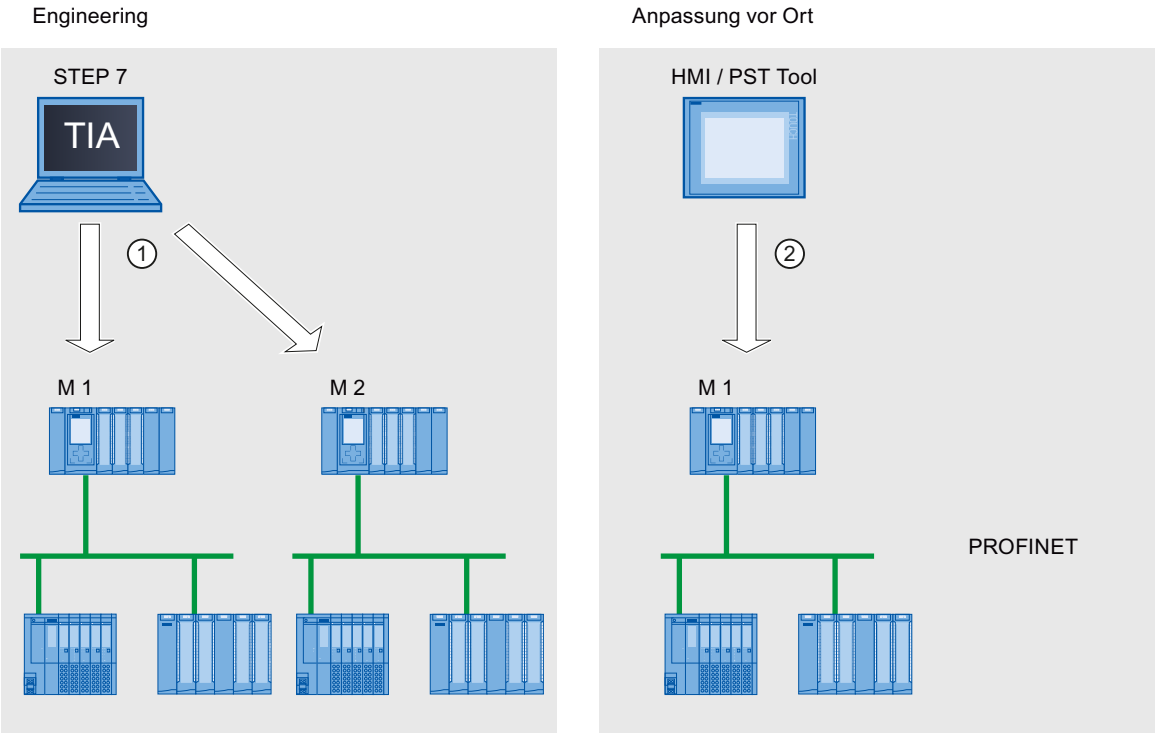

① Konfiguration mit mehrfach einsetzbarem IO-System laden

② IP-Adresse und Gerätename vor Ort am IO-Controller einstellen

Prinzip

Die Automatisierungskomponenten für eine Maschine umfassen ein PROFINET IO-System, bestehend aus einem IO-Controller (PROFINET-Schnittstelle einer CPU) und den ihm zugeordneten IO-Devices.

Mit der Einstellung "Mehrfach einsetzbares IO-System" am IO-System machen Sie ein STEP 7-Projekt zu einem "Serienmaschinen-Projekt".

Die Einstellung "Mehrfach einsetzbares IO-System" bewirkt, dass STEP 7 verschiedene Einstellungen und Prüfungen der Konfiguration vornimmt. Diese Einstellungen sorgen dafür, dass das IO-System in sich geschlossen ist und keine Abhängigkeiten zu Komponenten außerhalb des IO-Systems vorhanden sind.

Voraussetzung

- STEP 7 ab V13
- S7-1500 CPU oder ET 200SP CPU ab Firmware-Version V1.5 als IO-Controller

Regeln

Folgende Regeln gelten für ein mehrfach einsetzbares IO-System:

- Kein IO-Device darf als Shared Device konfiguriert sein.
- Die Ports der Geräte müssen verschaltet sein.
  Geräte, für die keine Portverschaltung projektierbar ist wie z. B. Interfacemodul IM 154-6 IWLAN (ET200pro PN), sind mit STEP 7 V13 nicht als IO-Device am mehrfach einsetzbaren IO-System betreibbar.

- Wenn ein IO-Device am mehrfach einsetzbaren IO-System ein I-Device ist (CPU als "intelligentes" IO-Device):
  Wenn das I-Device ein untergeordnetes IO-System hat, dann ist es zwar möglich, dieses untergeordnete IO-System an derselben PROFINET-Schnittstelle anzuschließen wie den übergeordneten IO-Controller. Allerdings ist nur eines der beiden IO-Systeme mehrfach einsetzbar nach folgender Regel:
  "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller" (am I-Device) = **ja**: Übergeordnetes IO-System ist mehrfach einsetzbar.
  "Parametrierung der PN-Schnittstelle durch übergeordneten IO-Controller" (am I-Device) = **nein**: Untergeordnetes IO-System ist mehrfach einsetzbar.

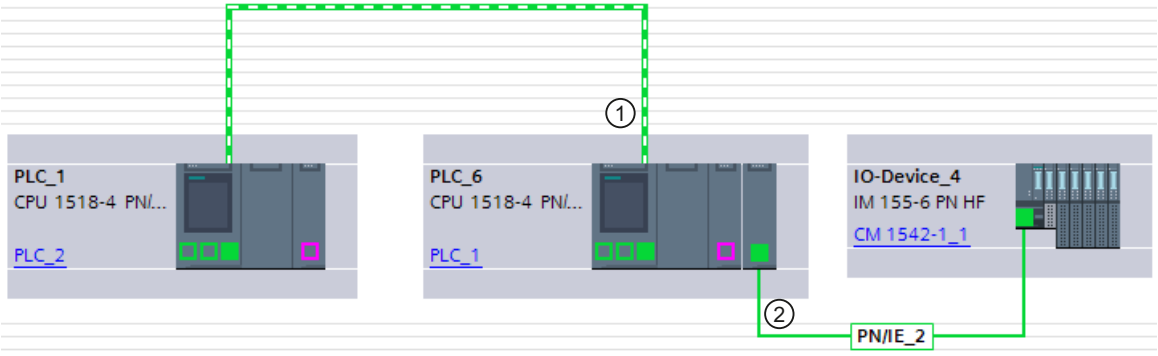

① I-Device am mehrfach einsetzbaren IO-System. Die PROFINET-Schnittstelle wird als IO-Device betrieben.

② Ein untergeordnetes IO-System am I-Device ist ebenfalls an dieser PROFINET-Schnittstelle angeschlossen.

- Wenn MRP (Media Redundancy Protocol) konfiguriert ist:
  - Alle IO-Devices am mehrfach einsetzbaren IO-System müssen derselben MRP-Domain angehören.
- Wenn IRT (Isochronous Real Time) konfiguriert ist:
  - Alle IO-Devices am mehrfach einsetzbaren IO-System müssen derselben Sync-Domain angehören.
  - Die Sync-Domain darf keine weiteren IO-Devices enthalten.
- IE/PB Links sind mit STEP 7 V13 nicht als IO-Device am mehrfach einsetzbaren IO-System betreibbar.
  Ab STEP 7 V13 SP1 sind IE/PB Links mit Firmware-Version ab V3 am mehrfach einsetzbaren IO-System betreibbar.

Projektierung

Ob eine Konfiguration mehrfach einsetzbar ist, legen Sie in den Eigenschaften des IO-Systems fest.

Alle weiteren Parametereinstellungen an den projektierten Geräten werden daraufhin von STEP 7 automatisch gesetzt und bei beim Übersetzen geprüft.

Randbedingungen

Um zu verhindern, dass ein Serienmaschinen-Projekt Abhängigkeiten zu anderen Geräten außerhalb der Maschine hat, beachten Sie Folgendes:

- Ein Serienmaschinen-Projekt besteht aus einem IO-Controller und den zugehörigen IO-Devices. Konfigurieren Sie daher im Serienmaschinen-Projekt nur eine CPU als IO-Controller und die zugehörigen IO-Devices.
- Nutzen Sie für die Kommunikation keine zweiseitigen Verbindungen, sondern nur einseitige Verbindungen bzw. unspezifizierte Verbindungen, wenn erforderlich.
  Hintergrund: Für die Projektierung der Kommunikation in einem STEP 7-Projekt bietet es sich grundsätzlich an, die IP-Adressparameter im Projekt einzustellen. Diese Vorgehensweise ist aber für mehrfach einsetzbare IO-Systeme nicht möglich, da die IP-Adressparameter vom IO-Controller und von den zugeordneten IO-Devices erst vor Ort zugewiesen werden. Zum Zeitpunkt der Projektierung sind die IP-Adressparameter also unbekannt.
  Wenn Sie trotzdem Kommunikation zu Geräten am PROFINET projektieren wollen, z. B. zu einem zentralen Koordinator, dann können Sie nur solche Kommuniktionsmechanismen verwenden, die eine dynamische Vergabe der IP-Adressparameter im Anwenderprogramm zulassen.
  Beispiel: Open User Communication
  Wenn das Gerät z. B. als aktiver Endpunkt (Initiator der Verbindung) projektiert ist, lassen sich die IP-Adressparameter z. B. in einem Datenbaustein halten. Erst zum Zeitpunkt der Inbetriebnahme versorgen Sie den Datenbaustein mit den aktuell gültigen IP-Adressparametern. Für diese dynamische Art der IP-Adressparametrierung gibt es keine Systemunterstützung, d. h. wenn Sie die Konfiguration des Systems ändern, werden die IP-Adressparameter nicht automatisch angepasst.
  Die Beschreibung zur Handhabung von Anweisungen zur Open User Communication finden Sie unter diesem Stichwort in der Online-Hilfe von STEP 7.

Siehe auch

Unspezifizierte Verbindungen (Seite 217)

Unspezifizierte Verbindung spezifizieren (Seite 245)

Anlegen einer neuen Verbindung - Übersicht (Seite 221)

Mehrfach einsetzbares IO-System konfigurieren

Voraussetzung

- S7-1500 CPU ab Firmware-Version V1.5 als IO-Controller

Vorgehen

Um ein Serienmaschinen-Projekt zu erstellen, gehen Sie folgendermaßen vor:

1. Legen Sie ein Projekt an.
2. Konfigurieren Sie eine S7-1500-CPU ab Firmware-Version V1.5 als IO-Controller.

3. Konfigurieren Sie die erforderlichen IO-Devices und weisen Sie die IO-Devices dem IO-Controller zu.
4. Projektieren Sie die Portverschaltung zwischen den Geräten.
5. Markieren Sie das IO-System, so dass Sie die Eigenschaften im Inspektorfenster bearbeiten können.
6. Aktivieren Sie das Optionskästchen "Mehrfach einsetzbares IO-System" im Bereich "Allgemein" des Inspektorfensters.

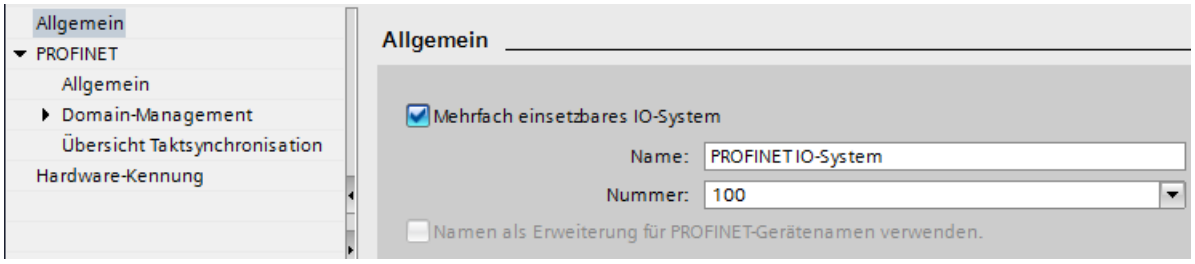

Ergebnis: Folgende Einstellungen werden von STEP 7 gesetzt:

- Der Gerätename des IO-Controllers (CPU) im Serienmaschinenprojekt ist eingestellt auf "Anpassen des PROFINET-Gerätenamens direkt am Gerät erlauben". Der IO-Controller (CPU) hat zunächst keinen PROFINET-Gerätenamen.
- Das IP-Protokoll des IO-Controllers (CPU) ist fest eingestellt auf "Anpassen der IP-Adresse direkt am Gerät erlauben". Die CPU hat zunächst keine IP-Adresse.
- Die Option "Gerätetausch ohne Wechselmedium ermöglichen" ist automatisch aktiviert. Diese Option bewirkt, dass eine automatische Inbetriebnahme möglich ist. Vor Ort entfällt dann die Zuweisung von Gerätenamen an den IO-Devices durch einen Inbetriebnehmer. Der IO-Controller weist den IO-Devices aufgrund der Solltopologie und anderer Einstellungen Gerätename und IP-Adresse im Anlauf zu.
- Der Gerätename der IO-Devices ist eingestellt auf "PROFINET-Gerätename automatisch generieren" (aus dem projektierten Namen des IO-Devices).
- Das IP-Protokoll bei den IO-Devices ist eingestellt auf "Anpassen der IP-Adresse durch den IO-Controller erlauben". Die IO-Devices haben zunächst keine IP-Adresse. Falls es sich bei einem IO-Device nicht um ein typisches dezentrales Peripheriesystem handelt (z B. ET 200-System), sondern um ein anderes Gerät wie z. B. ein HMI-Gerät, dann ändern Sie die Option auf "Anpassen der IP-Adresse direkt am Gerät erlauben", siehe unten.
- Die Gerätenummer für die IO-Devices wird automatisch zugewiesen und wird vor Ort dazu verwendet, um die IP-Adresse eindeutig zu machen.

Damit der IO-Controller später beim Betreiber den Gerätenamen anpassen kann, müssen Sie in der CPU die Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" aktivieren (Bereich "Erweiterte Optionen > Schnittstellen-Optionen" der Eigenschaften der PROFINET-Schnittstelle).

In der Voreinstellung ist diese Option deaktiviert.

Das folgende Bild zeigt die oben beschriebenen Einstellungen zur IP-Adresse und zum PROFINET-Gerätenamen.

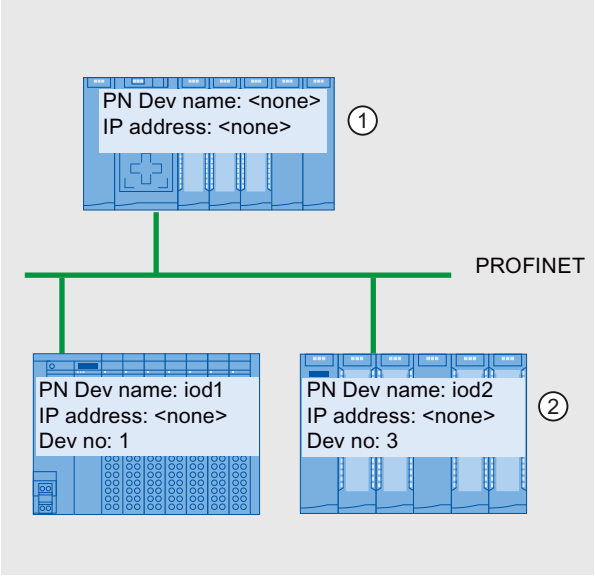

① Nach dem Laden der Konfiguration aus dem Serienmaschinen-Projekt hat der IO-Controller keinen Gerätenamen und keine IP-Adresse.

② Nach dem Laden haben die IO-Devices einen Gerätenamen und eine Gerätenummer, aber keine IP-Adresse.

Wie ein IO-Device vor Ort an eine IP-Adresse kommt

Im Folgenden sind die beiden Optionen "Anpassen der IP-Adresse durch den IO-Controller erlauben" und "Anpassen der IP-Adresse direkt am Gerät erlauben" erläutert, die grundsätzlich bei einem mehrfach einsetzbaren IO-System projektierbar sind.

Wenn Sie am IO-System die Option "Mehrfach einsetzbares IO-System" eingestellt haben, setzt STEP 7 automatisch bei den IO-Devices die Option "Anpassen der IP-Adresse durch den IO-Controller erlauben".

In diesem Fall bekommt das IO-Device eine IP-Adresse vom IO-Controller zugewiesen, die sich aus der vor Ort eingestellten IP-Adresse des IO-Controllers ergibt (siehe nächstes Kapitel). Diese Option ist sinnvoll, falls es sich bei dem IO-Device um ein Feldgerät handelt, z. B. ET 200MP, ET 200SP oder ein anderes dezentrales Peripheriesystem.

Wenn das IO-Device kein "normales" Feldgerät ist, sondern z. B. ein HMI-Gerät einem Windows-Betriebssystem, funktioniert die oben beschriebene Option "Anpassen der IP-Adresse durch den IO-Controller erlauben" nicht. In diesem Fall wählen Sie die Option "Anpassen der IP-Adresse direkt am Gerät erlauben". Die IP-Adresse müssen Sie dann vor Ort am Gerät zuweisen und selbst dafür sorgen, dass diese Adresse zu den IP-Adressen der übrigen IO-Devices und zur IP-Adresse des IO-Controller passt.

Siehe auch

IO-System vor Ort anpassen (Seite 3144)

IO-System vor Ort anpassen

Für die Anpassung der Maschine, die mit dem Serienmaschinen-Projekt geladen wurde, sind wenige Schritte zur Anpassung notwendig.

Vor Ort müssen nur Gerätename und IP-Adresse des IO-Controllers angepasst werden. Die Gerätenamen und IP-Adressen der IO-Devices ergeben sich aus diesen Anpassungen. In diesem Beispiel sind die Auswirkungen der Vor-Ort-Einstellungen für zwei konkrete Maschinenmodule beschrieben.

Die Vor-Ort-Einstellungen sind z. B. mit dem CPU-Display, Inbetriebnahme-Tools wie Primary Setup Tool (PST) oder PRONETA möglich. Sie benötigen für diese Einstellungen kein Programmiergerät mit STEP 7.

Voraussetzung

- Die Maschine wurde mit einem Serienmaschinen-Projekt geladen (siehe Mehrfach einsetzbares IO-System konfigurieren (Seite 3141)).
- Das Display ist betriebsbereit bzw. das gewünschte Tool zur Zuweisung von IP-Adresse und Gerätename ist vorhanden (z. B. PST Tool, STEP 7).
- Die Ports von IO-Controller und IO-Devices sind entsprechend der Projektierung verschaltet.

Vorgehen

Berücksichtigen Sie die Randbedingungen und Anweisungen zur Inbetriebnahme einer S7-1500! Einen Verweis zum Systemhandbuch S7-1500, in dem die Inbetriebnahme beschrieben ist, finden Sie unter "Siehe auch".

Um eine Serienmaschine vor Ort anzupassen, gehen Sie folgendermaßen vor:

1. Binden Sie die Maschine in das Netzwerk ein.
2. Schließen Sie das Gerät für die Zuweisung der IP-Adresse und Gerätenamen an die CPU an, z. B. ein PG/PC mit der entsprechenden Software.
3. Weisen Sie dem IO-Controller den gewünschten Gerätenamen und die gewünschte IP-Adresse zu.

Der IO-Controller (CPU) weist daraufhin den IO-Devices den angepassten PROFINET-Gerätenamen und eine eindeutige IP-Adresse zu.

Folgende Regeln gelten bei der Zuweisung:

- Die Gerätenamen der IO-Devices ergeben sich aus der Verkettung folgender Namensbestandteile, getrennt durch einen Punkt:
  <Projektierter Name des IO-Devices aus dem Serienmaschinen-Projekt>.<Am Gerät eingestellter Name des zugehörigen IO-Controllers>
- Die IP-Adressen der IO-Devices ergeben sich aus der vor Ort projektierten IP-Adresse des zugehörigen IO-Controllers und der Gerätenummer (Summe).

Hinweis

Achten Sie darauf, dass während der Zueisung keine doppelten IP-Adressen am Subnetz entstehen können. Der IO-Controller weist in diesem Fall keine neue IP-Adresse zu.

Im folgenden Bild ist für den IO-Controller der ersten Maschine der Gerätename "m1" und die IP-Adresse 192.168.1.10 vergeben worden.

Für die zweite Maschine ist der Gerätename "m2" und die IP-Adresse 192.168.1.20 vergeben worden.

Die sich ergebenden Gerätenamen und IP-Adressen entnehmen Sie dem Bild.

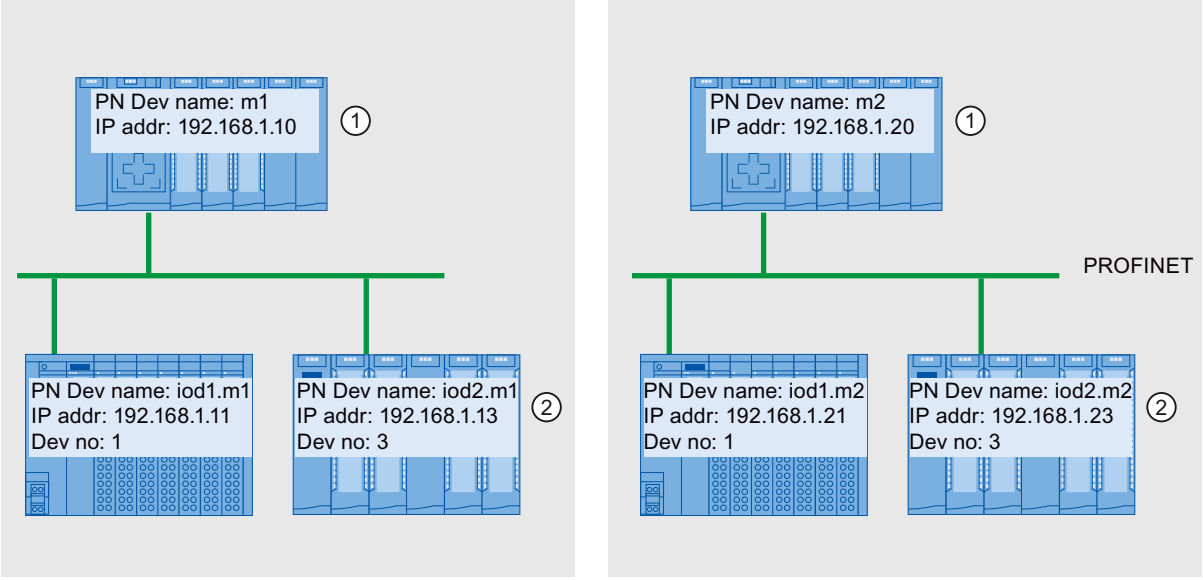

① Gerätename und der IP-Adresse am IO-Controller einstellen

② Nach dem Anlauf haben die IO-Devices einen aktualisierten Gerätenamen (<Projektierter Gerätename>.<Gerätename des IO-Controllers>) und eine angepasste IP-Adresse (= <IP-Adresse des IO-Controllers> + <Gerätenummer>)

Siehe auch

Mehrfach einsetzbares IO-System konfigurieren (Seite 3141)

Systemhandbuch S7-1500 (http://support.automation.siemens.com/WW/view/de/59191792)

Überschreiben des PROFINET-Gerätenamens erlauben

Mit S7-1500 CPUs ab Firmware-Version V1.5 haben Sie die Möglichkeit, PROFINET-Gerätenamen von IO-Devices im Anlauf der CPU zu überschreiben. Diese Option reduziert z. B. bei der automatischen Inbetriebnahme Ihren Aufwand bei Austausch eines Gerätes.

Wirkungsweise der Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben"

Bei aktivierter Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" kann der IO-Controller (CPU) die PROFINET-Gerätenamen der IO-Devices im IO-System überschreiben.

Mehrfach einsetzbare IO-Systeme können nur mit aktivierter Option betrieben werden. Der IO-Controller prüft vor dem Überschreiben, ob der Typ des IO-Devices mit dem projektierten Typ übereinstimmt.

Bei einer Teilinbetriebnahme (IO-Devices sind zum Teil nicht vorhanden bzw. überbrückt) dürfen Sie die Option nicht verwenden.

⚠ VORSICHT

Fehler bei Teilinbetriebnahme

Wenn die Gerätenamen bei einer Teilinbetriebnahme oder Falschverdrahtung falsch zugewiesen wurden, dann müssen Sie die Gerätenamen nach einer Korrektur der Verdrahtung von Hand löschen, um eine korrekte Vergabe zu erreichen.

⚠ WARNUNG

Falsche PROFINET-Gerätenamen aus der Projektierung durch falsch angeschlossene Geräte

Je nach angeschlossener Peripherie droht Lebensgefahr, schwere Verletzungsgefahr oder Sachschaden durch Fehlfunktionen, wenn durch falsch angeschlossene Geräte die falschen PROFINET-Gerätenamen eingetragen werden.

Um eine Gefährdung auszuschließen, prüfen Sie in jedem Fall bei einem Gerätetausch, ob das passende Ersatzgerät angeschlossen und die Portverschaltung der projektierten Soll-Topologie entspricht!

Wenn die Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" nicht aktiviert ist, dann kann der IO-Controller die Gerätenamen der IO-Devices nicht überschreiben. In diesem Fall müssen Sie z. B. nach einer Änderung eines PROFINET-Gerätenamens in der Projektierung oder im Fall eines Gerätetausches den PROFINET-Gerätenamen am IO-Device manuell zuweisen oder vor einer automatischen Inbetriebnahme den Gerätenamen der IO-Devices löschen.

Typische Gefahrenquelle

Beim Austausch eines IO-Devices ("Normalfall") ist weitgehend sichergestellt, dass das ausgetauschte Gerät entsprechend der projektierten Portverschaltung angeschlossen wird.

Im folgenden Bild ist der Fall dargestellt, wie die Anschlüsse zweier identisch aufgebauter PROFINET-Stränge an zwei Switchports vertauscht werden. Da der IO-Controller die Gerätenamen entsprechend der Soll-Topologie zuweist, hat der falsche Anschluss der Geräte gravierende Auswirkungen auf die Namensvergabe!

Durch die Ansteuerung unterschiedlicher Aktoren können in diesem Fall Gefahren von Anlage ausgehen.

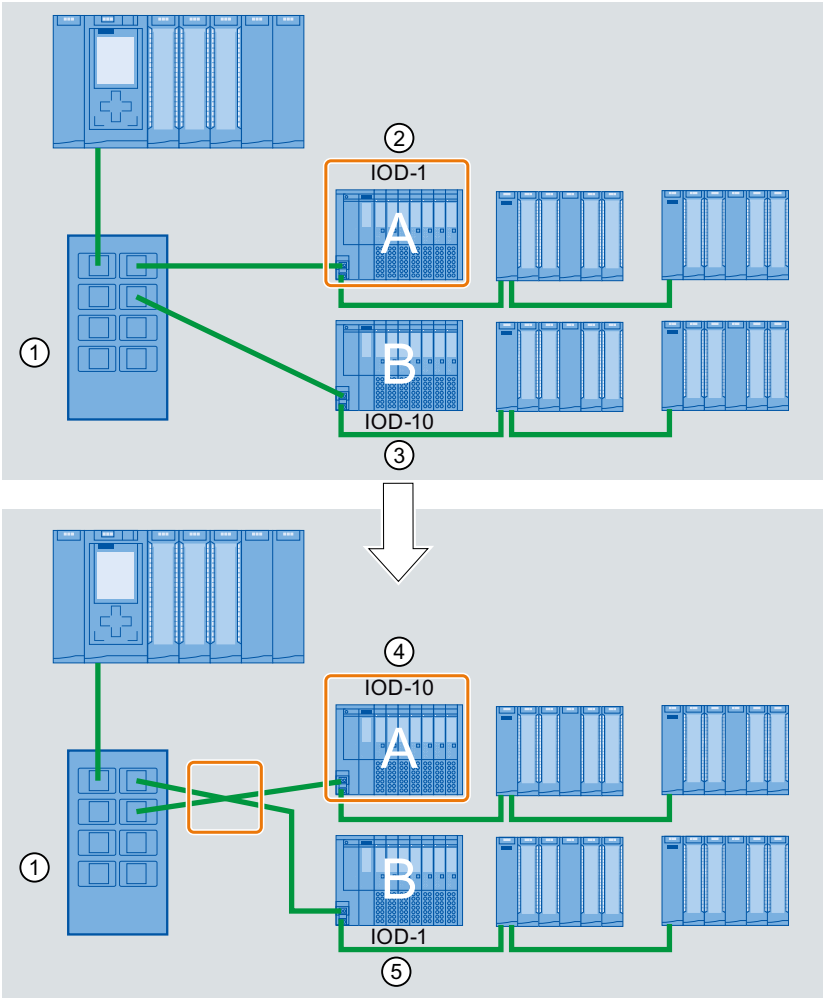

① Switch mit angeschlossenen PROFINET-Strängen
② Gerät A, Gerätename “IOD-1”: steuert Motor 1
③ Gerät B, Gerätename “IOD-10”: steuert Motor 10
④ Gerät A steuert Motor 10!
⑤ Gerät B steuert Motor 1!

Vorgehen

Um die Option "Überschreiben der Gerätenamen aller zugeordneten IO-Devices erlauben" zu ändern, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Netzsicht oder in der Gerätesicht die PROFINET-Schnittstelle der CPU, für die Sie die Option ändern wollen.
2. Wählen Sie den Bereich "Erweiterte Optionen > Schnittstellen-Optionen".
3. Ändern Sie die Option.

Konfigurationssteuerung (Optionenhandling) für PROFINET IO-Systeme

Wissenswertes zur Konfigurationssteuerung von IO-Systemen

Durch die Konfigurationssteuerung von IO-Systemen ist es möglich, aus einem Serienmaschinen-Projekt mehrere konkrete Ausprägungen einer Serienmaschine zu erzeugen.

Sie können flexibel die Konfiguration eines IO-Systems für eine bestimmte Anwendung variieren, solange sich die reale Konfiguration aus der projektierten Konfiguration ableiten lässt. Die projektierte Konfiguration stellt daher die Obermenge aller abzuleitenden realen Konfigurationen dar.

Das folgende Bild zeigt beispielhaft, wie aus einem Serienmaschinen-Projekt zwei PROFINET IO-Systeme mit unterschiedlicher Anzahl von IO-Systemen hervorgehen.

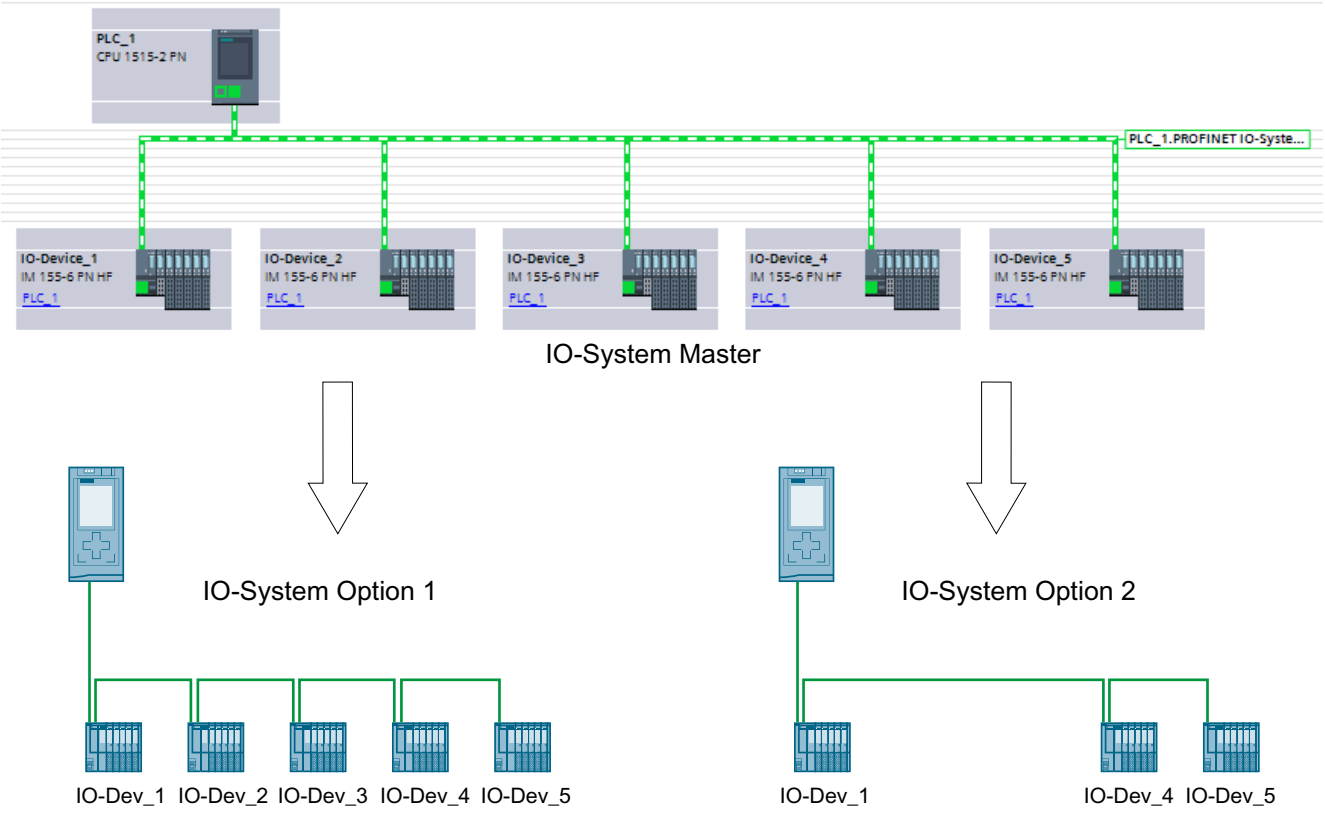

Die folgenden Abschnitte zeigen, wie Sie ein PROFINET IO-System projektieren und programmieren, damit vor Ort z. B. eine Serienmaschine ohne Projektierungssoftware in Betrieb genommen werden kann.

Konzept

Das Prinzip der Konfigurationssteuerung ist bereits auf Geräteebene für den flexiblen Einsatz von Baugruppen/Modulen bekannt ("Optionenhandling"). Sowohl für zentrale als auch für dezentrale Peripherie lassen sich verschiedene Konfigurationen aus einer Projektierung ableiten.

Mit S7-1500 CPUs ab Firmware-Version V1.7 ist dieses Prinzip auch auf der Ebene von IO-Systemen anwendbar. Sie haben die Möglichkeit, Stationen (IO-Devices) eines PROFINET IO-Systems in einer konkreten Anlage wegzulassen, hinzuzufügen oder die Reihenfolge der Stationen zu variieren.

Konfigurationssteuerung für Geräte und Konfigurationssteuerung für IO-Systeme lassen sich miteinander kombinieren; die Funktionen sind voneinander unabhängig.

Aus einem konfigurierten Maximalausbau eines IO-Systems lassen sich abweichende Varianten betreiben. Sie können in einem Serienmaschinen-Projekt einen Baukasten aus IO-Devices vorbereiten, den Sie dann flexibel für verschiedenste Ausbaustufen über die Konfigurationssteuerung anpassen.

Folgende Variationsmöglichkeiten stehen Ihnen zur Verfügung:

- Variation der Anzahl beteiligter IO-Devices
  Optionale IO-Devices für die Konfigurationssteuerung nehmen Sie in die Konfiguration auf, indem Sie einen entsprechend Datensatz mit der gewünschten Konfiguration im Anwenderprogramm übertragen.
- Variation der Reihenfolge beteiligter IO-Devices
  Sie passen die Portverschaltung der IO-Devices an die verwendete Topologie an, indem Sie einen entsprechenden Datensatz mit der gewünschten Topologie im Anwenderprogramm übertragen.

Das folgende Bild zeigt, wie Sie durch ein IO-Device, das Sie in der Netzsicht von STEP 7 als optional gekennzeichnet haben, zwei verschiedene Konfigurationen bedienen:

- Konfiguration ohne das optionale IO-Device:
  In diesem Fall müssen Sie mithilfe der Anweisung "ReconfigIOSystem" einen Datensatz an die PROFINET-Schnittstelle übertragen, der die Information enthält, dass kein optionales IO-Device in die Konfiguration aufgenommen werden soll.
- Konfiguration mit optionalem IO-Device:
  In diesem Fall müssen Sie mithilfe der Anweisung "ReconfigIOSystem" einen Datensatz an die PROFINET-Schnittstelle übertragen, der das optionale IO-Device in die Konfiguration aufnimmt.

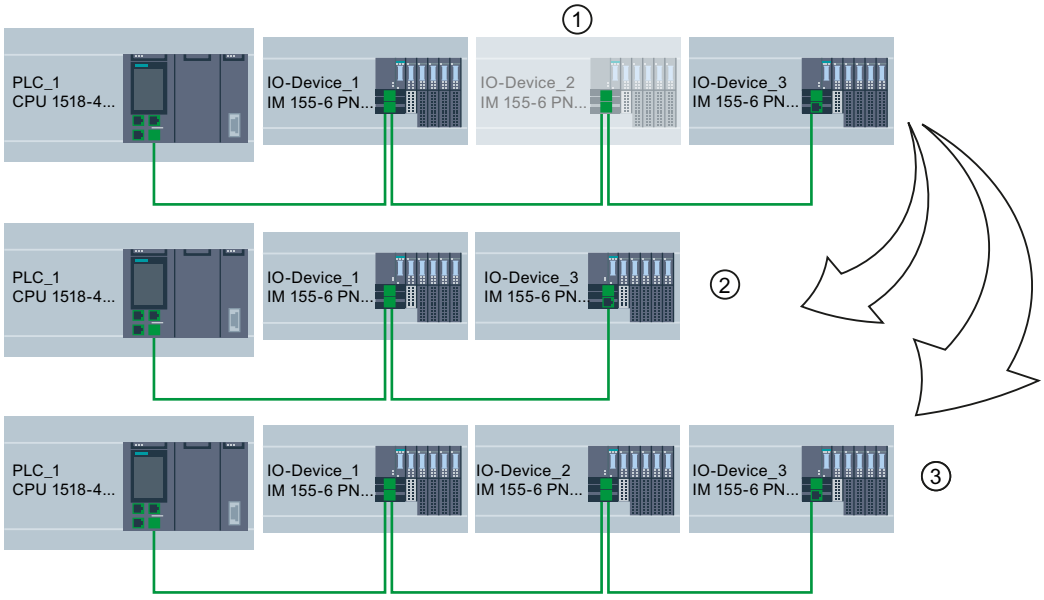

① Durch Parametrierung festgelegt: IO-Device_2 ist optionales IO-Device

② Konfiguration ohne das optionale IO-Device

③ Konfiguration mit optionalem IO-Device

Zusammenfassung: Prinzipielle Vorgehensweise

Folgende Phasen für die Umsetzung eines Serienmaschinen-Konzepts lassen sich unterscheiden:

1. Engineering-Phase: Serienmaschinen-Projekt anlegen und in konkrete Maschine oder Anlage laden:
   - Alle jemals in einer konkreten Maschine oder Anlage benötigten IO-Devices (Optionen) vollständig projektieren
   - Diejenigen IO-Devices, die in konkreten Maschinen oder Anlagen weggelassen werden, als optional parametrieren
   - Anwenderprogramm vorbereiten mit der Möglichkeit, vor Ort die gewünschte Auswahl der tatsächlich vorhandenen Konfiguration über Schalter oder HMI-Gerät treffen zu können
2. Inbetriebnahme-Phase: Konkrete Maschine oder Anlage für den Betrieb vorbereiten:
   - Maschine oder Anlage in das Netzwerk vor Ort integrieren (siehe Mehrfach einsetzbare IO-Systeme vor Ort anpassen)
   - Über projektierte Auswahlmöglichkeit den aktuell vorhandenen Ausbau des IO-Systems wählen

Optionale IO-Devices parametrieren

Voraussetzung

- S7-1500 CPU ab Firmware-Version V1.7 als IO-Controller
- STEP 7 ab V13 SP1
- Die Regeln (Seite 3162) zum Aufbau und Betrieb eines Serienmaschinen-Projekts sind berücksichtigt.

Portverschaltung für optionale IO-Devices

Ab STEP 7 V15.1 ist keine Portverschaltung für optionale IO-Devices erforderlich.

In folgenden Fällen ist eine Portverschaltung zwingend notwendig zwischen den Geräten des IO-Systems, die Sie über das Anwenderprogramm anpassen wollen.

- Sie haben IRT projektiert.
- Sie haben MRP projektiert.
- Sie verwenden STEP 7 <= V15.

Vorgehen

Um ein IO-Device als optionales IO-Device zu parametrieren, gehen Sie folgendermaßen vor:

1. Legen Sie ein Projekt an.
2. Konfigurieren Sie eine S7-1500 CPU ab Firmware-Version V1.7 als IO-Controller.
3. Konfigurieren Sie die erforderlichen IO-Devices und weisen Sie die IO-Devices dem IO-Controller zu.
4. Markieren Sie das IO-Device, das Sie als optional kennzeichnen wollen.
5. Wählen Sie den Bereich "PROFINET-Schnittstelle [X1] > Erweiterte Optionen".
6. Aktivieren Sie die Option "Optionales IO-Device"

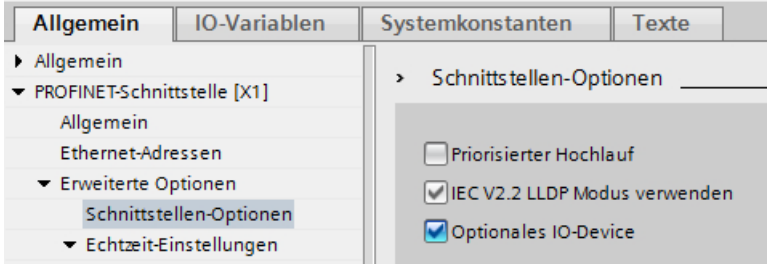

7. Wiederholen Sie Schritte 4 bis 6 für alle IO-Devices, die Sie als optional parametrieren wollen.
8. Laden Sie die Konfiguration in die CPU.

Ergebnis: Nach dem Laden dieser Konfiguration ist das Systemverhalten wie folgt:

- Die CPU ist für die Konfigurationssteuerung des IO-Systems vorbereitet.
- Alle IO-Devices sind deaktiviert.
- Unabhängig davon, ob Sie die Konfiguration im Anwenderprogramm anpassen (optionale IO-Devices hinzunehmen) oder keine Änderung an der geladenen Konfiguration vornehmen: Sie müssen im Anwenderprogramm die Anweisung "ReconfigIOSystem" aufrufen und dem System die aktuelle Konfiguration mitteilen.
  Ohne Aufruf der Anweisung "ReconfigIOSystem" ist das System nicht betriebsbereit.
  Weitere Informationen zur Vorgehensweise siehe Optionale IO-Devices im Programm aktivieren (Seite 3152).

Schnelle Parametrierung in Tabelle "E/A-Kommunikation"

Ob ein IO-Device optional ist oder nicht, können Sie auch im Register "E/A-Kommunikation" einstellen.

In einer zusätzlichen Spalte "Optionale IO-Device" ist für jedes IO-Device ein bedienbares Optionskästchen vorhanden, das anzeigt, ob ein IO-Device optinal ist oder nicht. Hier können Sie zentral die Einstellung anpassen.

Optionale IO-Devices im Programm aktivieren

Voraussetzung

- CPU S7-1500 ab Firmware-Version V1.7 als IO-Controller.
- STEP 7 V13 SP1.
- Mindestens ein IO-Device wurde als optionales IO-Device parametriert.
- Die Regeln (Seite 3162) zum Aufbau und Betrieb eines Serienmaschinen-Projekts sind berücksichtigt.

Vorgehen

Beachten Sie die Hinweise und Regeln zur Inbetriebnahme eines S7-1500-Systems in der Dokumentation zur SIMATIC S7-1500!

Die folgende Beschreibung des Vorgehens umfasst nur Schritte, die zum Verständnis der programmgesteuerten Aktivierung eines optionalen IO-Devices notwendig sind.

Um optionale IO-Devices zu aktivieren oder zu deaktivieren, in die Konfiguration aufzunehmen und im Anwenderprogramm zu aktivieren, gehen Sie folgendermaßen vor:

1. Erzeugen Sie einen Datensatz wie hier beschrieben.
2. Rufen Sie die Anweisung "ReconfigIOSystem" auf und wählen Sie den Modus 1, um alle IO-Devices zu deaktivieren.
   Wenn Sie die CPU in den Zustand STOP oder NETZ-AUS bringen, um in diesem die Anlage umzubauen (z. B. um ein optionales IO-Device hinzuzufügen), dann ist kein explizites Deaktivieren mit "ReconfigIOSystem" mit Modus 1 erforderlich. In diesem Fall, d. h. nach einem STOP-RUN-Übergang und nach einem NETZ-AUS > NETZ-EIN-Übergang, sind alle IO-Devices automatisch deaktiviert.
3. Wenn Sie die Anlage in einen sicheren Zustand gebracht haben, der einen gefahrlosen Umbau erlaubt:
   Bauen Sie die Anlage entsprechend Ihrem Anwendungsfall zusammen. Fügen Sie die erforderliche optionale IO-Devices an den Stellen ein, die Sie in der Projektierung dafür vorgesehen haben (Reihenfolge beachten!) bzw. entfernen Sie nicht mehr benötigte optionale IO-Devices.
4. Verkabeln Sie die IO-Devices.
5. Lassen Sie das S7-1500-System anlaufen und rufen erneut die Anweisung "ReconfigIOSystem" auf. Wählen Sie den Modus 2 zum Übertragen des Datensatzes CTRLREC.
6. Nach erfolgreicher Übertragung des Datensatzes rufen Sie erneut die Anweisung "ReconfigIOSystem" auf. Wählen Sie den Modus 3 zum Aktivieren aller IO-Devices, die zur aktuellen Konfiguration gehören.

Ergebnis

Die CPU aktiviert folgende IO-Devices:

- Alle IO-Devices, die Sie nicht als optionale IO-Devices parametriert haben.
- Alle optionalen IO-Devices, die im Steuerdatensatz (CTRLREC) aufgeführt sind.

Folgende IO-Devices bleiben deaktiviert:

- Docking Units (im Betrieb wechselnde IO-Devices).
- Optionale IO-Devices, die im Steuerdatensatz nicht aufgeführt sind.

Hinweis

Rufen Sie die Anweisung "ReconfigIOSystem" in allen Modi mit demselben Steuerdatensatz (CTRLREC) auf!

Wenn Sie in den unterschiedlichen Modi unterschiedliche Datensätze verwenden, führt das zu einer nicht konsistenten Anpassung der Konfiguration und damit zu schwer nachvollziehbaren Fehlermeldungen.

Beispiel: Datensatzaufbau für die Aktivierung eines IO-Devices

Das IO-Device "IO-Device_2" soll als einziges IO-Device im Anwenderprogramm aktiviert werden. Dazu benötigen Sie nur die HW-Kennung von "IO-Device_2".

Empfehlung: Nutzen Sie die Systemkonstanten der HW-Kennungen statt der absoluten Werte wie in diesem Beispiel gezeigt. Mit dieser Vorgehensweise bleibt der DB-Inhalt unbeeinflusst von Änderungen der HW-Kennungen infolge einer Änderung der Projektierung.

Der Datensatz soll in einem Datenbaustein hinterlegt werden und im Anwenderprogramm mit der Anweisung "ReconfigIOSystem" an die PROFINET-Schnittstelle des IO-Controllers übertragen werden.

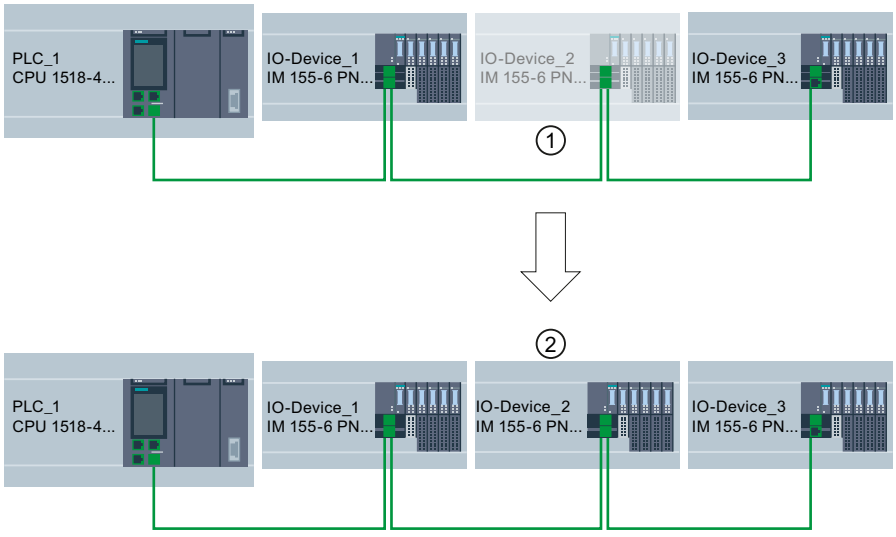

① IO-Device_2 ist als optionales IO-Device parametriert.

② Nach der Übertragung des Datensatzes und Aktivierung der Konfiguration mit der Anweisung "ReconfigIOSystem" ist IO-Device_2 in die Konfiguration aufgenommen und nimmt am Datenaustausch mit dem IO-Controller teil.

Datenbaustein erstellen

Der Steuerdatensatz ist in diesem Beispiel in einem Datenbaustein erstellt. Der Datenbaustein ist wie folgt aufgebaut:

Zeile 2: Array-Definition: Array vom Typ Word mit 4 Elementen. Als Datentyp ist Array of Word zugelassen.

Zeile 3: Version des Datensatzes (Aktuell: V1.0).

Zeile 4: Anzahl zu aktivierender optionaler IO-Devices (hier: 1).

Zeile 5: Liste der HW-Kennungen der IO-Devices, hier als Systemkonstante eingefügt.

Zeile 6: Anzahl Portverschaltungen, die im Anwenderprogramm gesetzt werden (hier: 0).

Zeile 7: Weitere Datensätze (optional).

myCTRLREC

	Name	Data type	Start value	Comment
1	Static			
2	ArrMachineConfig0	Array[0..3] of Word		Array for one optional IO device
3	ArrMachineConfig0[0]	Word	16#0100	
4	ArrMachineConfig0[1]	Word	16#0001	
5	ArrMachineConfig0[2]	Word	"IO-Device_2~IODevice"	
6	ArrMachineConfig0[3]	Word	16#0000	
7	ArrMachineConfig1	Array[0..8] of Word		

Regeln zur Aufrufreihenfolge von "ReconfigIOSystem"

- Versorgen Sie die Anweisung "ReconfigIOSystem" immer mit demselben Steuerdatensatz (Eingangsparameter "CTRLREC").
- Aufrufreihenfolge nach NETZ-AUS > NETZ-EIN-Übergang:
  - ReconfigIOSystem-Aufruf mit Mode 1 (optional).
  - ReconfigIOSystem-Aufruf mit Mode 2 (obligatorisch, auch ohne Umkonfiguration!).
  - ReconfigIOSystem-Aufruf mit Mode 3 (obligatorisch).
- Aufrufreihenfolge nach STOP > RUN-Übergang:
  - ReconfigIOSystem-Aufruf mit Mode 1 (optional).
  - ReconfigIOSystem-Aufruf mit Mode 2 (obligatorisch, wenn die Konfiguration im STOP geändert wurde. Sonst nicht notwendig).
  - ReconfigIOSystem-Aufruf mit Mode 3 (obligatorisch).
- Aufrufreihenfolge bei Umkonfiguration im RUN:
  - ReconfigIOSystem-Aufruf mit Mode 1 (obligatorisch).
  - ReconfigIOSystem-Aufruf mit Mode 2 (obligatorisch).
  - ReconfigIOSystem-Aufruf mit Mode 3 (obligatorisch).

Erläuterungen und Tipps zu den Regeln

- Wenn Sie ein als optionales IO-Device parametriertes IO-Device nicht im Steuerdatensatz bzw. Datenbaustein aufführen, dann ist dieses IO-Device nicht Teil der Konfiguration und nimmt nicht am Datenaustausch mit der CPU teil.
- Wenn Sie gar kein optionales IO-Device aktivieren und mit der geladenen Konfiguration ohne Umkonfiguration arbeiten, dann müssen Sie trotzdem die im Abschnitt oben beschriebene Vorgehensweise einhalten und den Steuerdatensatz zur CPU übertragen. Der Steuerdatensatz hat die einfache Struktur mit folgenden Variablen:
  - Version (High Byte =1, Low Byte = 0)
  - Anzahl zu aktivierender optionaler Devices = 0
  - Anzahl Portverschaltungen, die im Anwenderprogramm gesetzt werden = 0

- Nach einem STOP > RUN-Übergang und nach einem NETZ-AUS > NETZ-EIN-Übergang sind alle IO-Devices automatisch deaktiviert. Sie benötigen daher für die ordnungsgemäße Funktion der Konfigurationssteuerung keinen ReconfigIOSystem-Aufruf mit Mode 1.
  Wenn Sie Ihr Projekt als allgemeingültiges Muster für die Programmierung der Konfigurationssteuerung nutzen, empfehlen wir trotzdem den ReconfigIOSystem-Aufruf mit Mode 1 vor jeder Umkonfiguration. Sie können das Muster dann auch für Umkonfigurationen im Betriebszustand RUN verwenden.
- Inbetriebnahme von umfangreichen IO-Systemen (mehr als 8 optionale IO-Devices) bei gleichzeitiger Nutzung von IRT:
  Um die Hochlaufzeiten bei der Aktivierung der optionalen IO-Devices (ReconfigIOSystem, Mode 3) gering zu halten, folgender Tipp: Kontrollieren Sie die Gerätenummern der IO-Devices. Die Gerätenummern sollten vom IO-Controller aus in aufsteigender Reihenfolge der topologischen Verschaltung folgen. Je weiter ein IO-Device topologisch vom IO-Controller entfernt ist, d. h. je mehr IO-Devices zwischen IO-Controller und dem betrachten IO-Device liegen, desto größer sollte die Gerätenummer sein.
  Die Gerätenummern stellen Sie bei markierter PROFINET-Schnittstelle im Bereich "Ethernet-Adressen - PROFINET" im Inspektorfenster ein.
  Beispiel für die Vergabe von Gerätenummern bei einer Linientopologie:

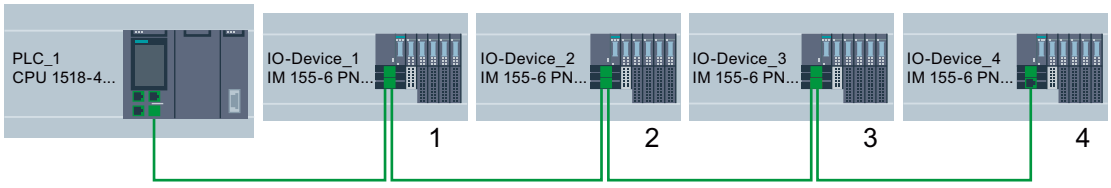

- Die CPU bearbeitet die Anweisung "ReconfigIOSystem" zur Übertagung des Steuerdatensatzes asynchron.
  Wenn Sie die Anweisung im Anlaufprogramm aufrufen, müssen Sie daher "ReconfigIOSystem" in einer Schleife wiederholt aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz übertragen ist.
  Tipp: Verwenden Sie zur Programmierung der Schleife die Programmiersprache SCL mit der Anweisung REPEAT ... UNTIL.

```
REPEAT
    "ReconfigIOSystem"(REQ := "start_config_ctrl",
                MODE := 1,
                LADDR := 64,
                CTRLREC := "myCTRLREC".ArrMachineConfig0,
                DONE => "conf_DONE",
                BUSY => "conf_BUSY",
                ERROR => "conf_ERROR",
                STATUS => "conf_STATUS");
UNTIL NOT "conf_BUSY"
END_REPEAT;
```

Weiterführende Informationen

Informationen zum grundsätzlichen Aufbau des Datensatzes und zur Verwendung der Anweisung "ReconfigIOSystem" finden Sie hier.

Siehe auch

Optionale IO-Devices parametrieren (Seite 3151)

Flexible Anordnung von IO-Devices parametrieren

Der folgende Abschnitt zeigt, wie Sie die Voraussetzungen für das Ändern der Reihenfolge von IO-Devices in einem PROFINET IO-System schaffen.

Diese Funktion ist auch mit optionalen IO-Devices möglich. Um die Darstellung einfach zu halten, ist im Folgenden eine Maximalkonfiguration ohne optionale IO-Devices dargestellt.

Konzept

Eine typische Anwendung für ein Serienmaschinen-Projekt besteht darin, aus einem Baukasten unterschiedlicher Anlagenteile eine Gesamtanlage zusammenzusetzen, die sich nur durch die unterschiedliche räumlicher Anordnung der Teile voneinander unterscheiden - z. B. bei Transportsystemen. Jeder Anlagenteil besteht aus einer funktionalen Einheit von Mechanik (Schienen oder Förderbänder) und Elektrik (Stromversorgung, IO-Device mit Peripheriemodulen, Sensoren, Aktoren, Motoren, PROFINET-Anschluss für Datenaustausch mit zentraler Steuerung ...).

Das folgende Bild zeigt, wie durch den Tausch zweier Schienensegmente ein neues Transportsystem entsteht, das mit einer vorverlegten Weiche an die örtlichen Gegebenheiten angepasst ist.

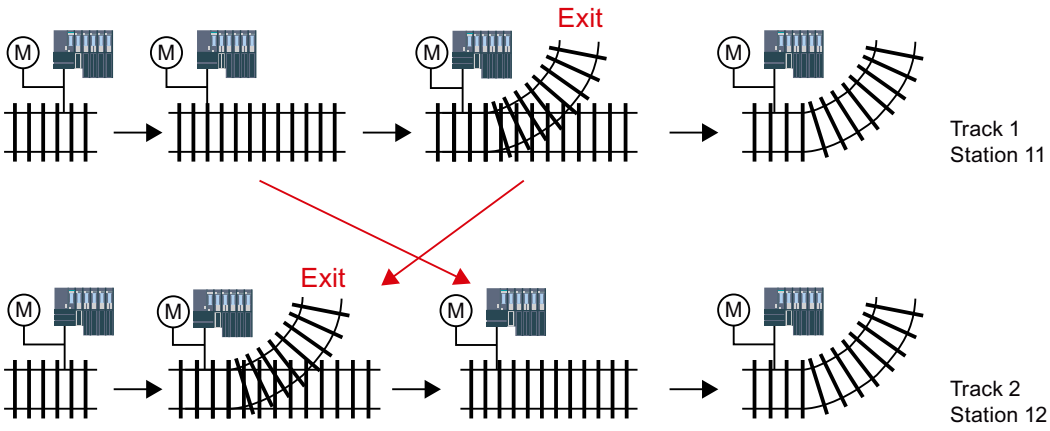

Aus Sicht der Automatisierung interessiert die Umsetzung einer flexiblen Reihenfolge der einzelnen IO-Devices in einem Projekt, das vor Ort nicht mehr geändert werden soll.

Die Reihenfolge der IO-Devices ist durch die Portverschaltung bestimmt. In den Porteigenschaften der IO-Geräte projektieren Sie jeweils den Partnerport und damit das Nachbargerät, das am jeweiligen lokalen Port angeschlossen ist. Wenn der Partnerport durch das Anwenderprogramm bestimmt werden soll, dann müssen Sie als Partnerport die Option "Partner durch Anwenderprogramm gesetzt" wählen.

Im folgenden Bild zeigt die Ausgangskonfiguration des oben gezeigten Transportsystems, das eine Änderung der Reihenfolge der angeschlossenen IO-Devices über das Anwenderprogramm zulassen soll. Im Beispiel soll die Reihenfolge von IO-Device_2 und IO-Device_3 über das Anwenderprogramm gesteuert werden.

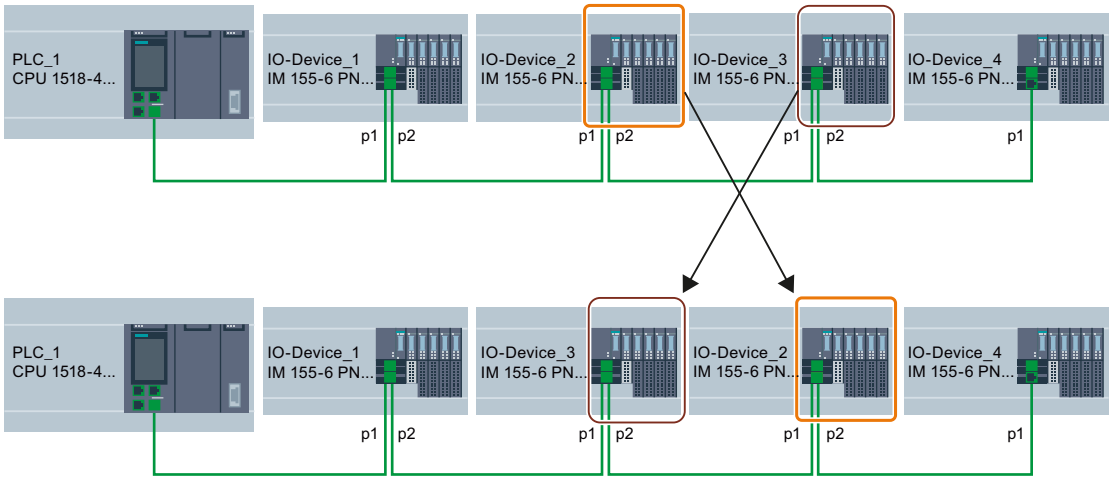

Um zu bestimmen, wie die Partnerport-Einstellungen zu wählen sind, müssen Sie für jedes Gerät und jeden Port eines Geräts notieren, welcher Partner für eine Verschaltung in Frage kommt.

- Wenn der Partner in den verschiedenen vorgesehenen Konfigurationen immer derselbe ist, dann wählen Sie den Partnerport für diesen Partner.
- Wenn die Partner in unterschiedlichen Konfigurationen variieren, dann wählen Sie "Partner durch Anwenderprogramm einstellen".

Für das Beispiel im Bild oben ergeben sich folgende Porteinstellungen:

Gerät	lokaler Port	Partnerport-Einstellung	Erläuterung
PLC_1	p1	p1 (IO-Device_1)	Partner von PLC_1 an Port 1 ist IO-Device_1 (immer)
IO-Device_1	p1	p1 (PLC_1)	Partner von IO-Device_1 an Port 1 ist PLC_1 (immer)
IO-Device_1	p2	Partner durch Anwenderprogramm gesetzt	Partner von IO-Device_1 an Port 2 ist entweder IO-Device_2 oder IO-Device_3 => Partner durch Anwenderprogramm einstellen
IO-Device_2	p1	Partner durch Anwenderprogramm gesetzt	Partner von IO-Device_2 an Port 1 ist entweder IO-Device_1 oder IO-Device_3 => Partner durch Anwenderprogramm einstellen
IO-Device_2	p2	Partner durch Anwenderprogramm gesetzt	Partner von IO-Device_2 an Port 2 ist entweder IO-Device_3 oder IO-Device_4 => Partner durch Anwenderprogramm einstellen
IO-Device_3	p1	Partner durch Anwenderprogramm gesetzt	Partner von IO-Device_3 an Port 1 ist entweder IO-Device_2 oder IO-Device_1 => Partner durch Anwenderprogramm einstellen
IO-Device_3	p2	Partner durch Anwenderprogramm gesetzt	Partner von IO-Device_3 an Port 2 ist entweder IO-Device_4 oder IO-Device_2 => Partner durch Anwenderprogramm einstellen

Gerät	lokaler Port	Partnerport-Einstellung	Erläuterung
IO-Device_4	p1	Partner durch Anwender-programm gesetzt	Partner von IO-Device_4 an Port 1 ist entweder IO-Device_3 oder IO-Device_2 => Partner durch Anwen-derprogramm einstellen
IO-Device_4	p2	beliebiger Partner	kein Partner an Port 2 vorgesehen

Voraussetzung

- S7-1500 CPU ab Firmware-Version V1.7 als IO-Controller.
- STEP 7 ab V13 SP1.
- Die Regeln (Seite 3162) zum Aufbau und Betrieb eines Serienmaschinen-Projekts sind berücksichtigt.

Vorgehen

Um den Partnerport eines Ports für eine programmgesteuerte Verschaltung einzustellen, gehen Sie folgendermaßen vor:

1. Markieren Sie die PROFINET-Schnittstelle des Geräts (IO-Controller oder IO-Device), dessen Port Sie einstellen wollen.
2. Wählen Sie in den Eigenschaften der PROFINET-Schnittstelle den Bereich "Portverschaltung" (Erweiterte Optionen > Port [...] > Portverschaltung).
3. Wählen Sie aus der Klappliste als Partnerport "Partner durch Anwenderprogramm einstellen".
4. Wiederholen Sie die Schritte 1. bis 3 für jeden Port, der über das Anwenderprogramm verschaltet werden soll.

Siehe auch

Anordnung von IO-Devices im Programm anpassen (Seite 3159)

Anordnung von IO-Devices im Programm anpassen

Voraussetzung

- CPU S7-1500 ab Firmware-Version V1.7 als IO-Controller.
- Mindestens eine Partnerport wurde als "Partner durch Anwenderprogramm gesetzt" parametriert.
- Die Regeln (Seite 3162) zum Aufbau und Betrieb eines Serienmaschinen-Projekts sind berücksichtigt.

Vorgehen

Das Vorgehen entspricht dem Vorgehen zum Aktivieren von optionalen IO-Devices.

Lediglich der Aufbau des Datensatzes muss für die programmgesteuerte Zuordung der Ports erweitert werden. Die Erweiterung ist in den nächsten Abschnitten beschrieben.

Beispiel: Datensatzaufbau für die Zuordnung von Partnerports

Für den Aufbau des Datensatzes benötigen Sie die HW-Kennungen der Ports.

Der Datensatz soll in einem Datenbaustein hinterlegt werden und im Anwenderprogramm mit der Anweisung "ReconfigIOSystem" an die PROFINET-Schnittstelle des IO-Controllers übertragen werden.

Da der Eingangsparameter RECORD der Anweisung "ReconfigIOSystem" den Datentyp VARIANT hat, müssen Sie zunächst einen Datentyp für den Datenbaustein erstellen.

In den folgenden Abschnitten ist der Aufbau des PLC-Datentyps erläutert sowie der Aufbau des darauf basierenden Datenbausteins.

Abgeleitete Konfiguration wählen

Für folgende gewählte Konfiguration wird im Folgenden gezeigt, wie der Datensatz beschaffen sein muss, damit die IO-Devices in der vorgesehenen Reihenfolge durch das Anwenderprogramm verschaltet werden.

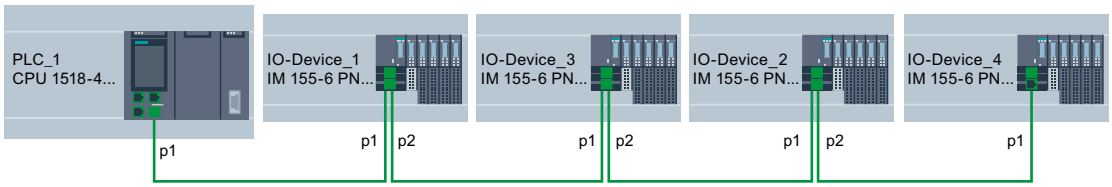

Vorausgesetzt ist die flexible Konfiguration aus dem vorangegangenen Abschnitt (Seite 3157) mit den dort beschriebenen Einstellungen für die jeweiligen Partnerports.

Die Partnerports in der konkreten, abgeleiteten Konfiguration sind festgelegt, daher ist es möglich, die HW-Kennungen der beteiligten Ports zu benennen.

Die folgende Tabelle enthält nur die Geräte, bei denen die Ports durch das Anwenderprogramm bestimmt werden können. Nur diese Geräte sind für den Aufbau des Datensatzes relevant!

Gerät	lokaler Port	Partnerport-Einstellung	Partnerport der gewählten Konfiguration
IO-Device_1	p2 = Port 2 HW-Kennung: 251	Partner durch Anwenderprogramm gesetzt	Port 1 von IO-Device_3 HW-Kennung: 261
IO-Device_2	p1 = Port 1 HW-Kennung: 281	Partner durch Anwenderprogramm gesetzt	Port 2 von IO-Device_3 HW-Kennung: 291
IO-Device_2	p2 = Port 2 HW-Kennung: 311	Partner durch Anwenderprogramm gesetzt	Port 1 von IO-Device_4 HW-Kennung: 321

Datenbaustein erstellen

Für die abgeleitete Konfiguration ist hier exemplarisch der Aufbau des Datenbausteins "DB-IO-SYSTEM-Port-Interconnections" erläutert.

Diesen Datenbaustein verwenden Sie beim Aufruf der Anweisung "ReconfigIOSystem" am Eingangsparameter "CTRLREC".

Statt der absoluten Werte für die HW-Kennungen der Ports sind hier die Systemkonstanten der HW-Kennungen verwendet.

Der Datenbaustein ist wie folgt aufgebaut:

Zeile 2: Vereinbarung eines Arrays of Word (nur dieser Datentyp ist möglich).

Zeile 3: Version des Steuerdatensatzes: V1.0.

Zeile 4: Anzahl optionaler IO-Devices: 0.

Zeile 5: Anzahl spezifizierter Portverschaltungen: 3.

Zeile 6: Portverschaltung 1, Lokaler Port.

Zeile 7: Portverschaltung 1, Partnerport.

Zeile 8: Portverschaltung 2, Lokaler Port.

Zeile 9: Portverschaltung 2, Partnerport

Zeile 10: Portverschaltung 3, Lokaler Port.

Zeile 11: Portverschaltung 3, Partnerport.

IO-SYSTEM-Port-Interconnections

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	ArrMachineConfig	Array[0..8] of Word		data record for port interconn settings
3	ArrMachineConfig[0]	Word	16#0100	
4	ArrMachineConfig[1]	Word	16#0000	
5	ArrMachineConfig[2]	Word	16#0003	
6	ArrMachineConfig[3]	Word	"IO_device_1~PROFINET_interface~Port_2"	
7	ArrMachineConfig[4]	Word	"IO_device_3~PROFINET_interface~Port_1"	
8	ArrMachineConfig[5]	Word	"IO_device_2~PROFINET_interface~Port_1"	
9	ArrMachineConfig[6]	Word	"IO_device_3~PROFINET_interface~Port_2"	
10	ArrMachineConfig[7]	Word	"IO_device_2~PROFINET_interface~Port_2"	
11	ArrMachineConfig[8]	Word	"IO_device_4~PROFINET_interface~Port_1"	

Verschaltung im Datenbaustein nicht aufgeführt

Wenn Sie in den Port-Eigenschaften den Partnerport als "Partner durch Anwenderprogramm einstellen" parametriert haben und diesen Port im Datensatz bzw. Datenbaustein nicht aufführen, dann setzt die CPU für diesen Port die Einstellung "beliebiger Partner". Wenn Sie gar keinen Datensatz übertragen, setzt die CPU diese Einstellung "beliebiger Partner" für alle programmgesteuerten Zuordnungen.

Für IRT-Konfigurationen ist die Einstellung "beliebiger Partner" nicht zulässig.

Weiterführende Informationen

Informationen zum grundsätzlichen Aufbau des Datensatzes und zur Verwendung der Anweisung "ReconfigIOSystem" finden Sie hier.

Systemverhalten und Regeln

Im Folgenden erfahren Sie, wie sich ein IO-System, dessen Konfiguration über das Anwenderprogramm gesteuert ist, im Betrieb verhält.

Außerdem sind hier die Regeln und Einschränkungen aufgeführt, die Sie für die Projektierung des Maximalausbau der Konfiguration in Ihrem Serienmaschinen-Projekt zu berücksichtigen sind.

Systemverhalten

- Systemdiagnose:
  Wenn ein IO-Device deaktivert ist, wird es aus Sicht der Systemdiagnose (Online-Sicht oder Online & Diagnose) als "deaktiviert" angezeigt.
- Topologiesicht:
  Offline-Sicht: Wie projektiert. Ports, deren Partnerport parametriert wurde als "Partner durch Anwenderprogramm einstellen", zeigen keine Verschaltung.
  Online-Sicht: Ports und Verschaltungen zu deaktivierten IO-Devices werden in einem anderen Grünton dargestellt als fehlerfreie Ports und Verschaltungen von aktivierten IO-Devices.
- Darstellung im Webserver:
  Name von Geräten wird angezeigt wie projektiert (Eigenschaften > Allgemein > Projektinformation).
  Zugewiesener PROFINET-Gerätename wird auf der Webseite "Kommunikation", Register "Parameter" für die CPU angezeigt.
  IP-Adressparameter: Aktuell zugewiesene IP-Adressparameter werden auf der Webseite "Baugruppenzustand" angezeigt.
  Topologie: Die aktuelle Topologie wird im Webserver angezeigt, wie sie sich aufgrund der Anpassung per Anwenderprogramm ergeben hat. IO-Devices, die als optional parametriert wurden, zeigt der Webser als "deaktivierte" IO-Devices an.

Regeln

Es gelten die Regeln für Serienmaschinen-Projekte wie hier (Seite 3138) beschrieben.

Für konfigurationsgesteuerte IO-Systeme gelten die folgenden zusätzlichen Regeln:

- Wenn Sie MRP (Media Redundancy Protocol) projektieren:
  Die als Ringports parametrierten Ports dürfen nicht über Anwenderprogramm verschaltet werden.
  Geräte mit Ringports (Teilnehmer einer MRP-Domain) können aber optionale IO-Devices sein.
- Wenn Sie Dockingsysteme projektieren (= im Betrieb wechselnde IO-Devices):
  Weder die Docking Station noch das erste IO-Device einer Docking Unit dürfen optionale IO-Devices sein.
  Die Ports der Docking Units dürfen nicht über Anwenderprogramm verschaltet werden.
- Wenn Sie IRT projektieren:
  Die Reihenfolge von synchronisierten IO-Devices ("IRT-Geräte") muss durch die Konfiguration festgelegt sein und darf sich in den verschiedenen Ausprägungen einer Serienmaschine nicht ändern. Die Ports von IRT-Geräten dürfen daher nicht über Anwenderprogramm verschaltet werden.
  Sie haben aber die Mögichkeit, IRT-Geräte in einer Linie als optionale IO-Devices zu parametrieren.
  Außerdem haben Sie die Möglichkeit, RT-Geräte, die z. B. über einen Switchport von dieser Linie "abgehängt" sind (siehe Bild), über das Anwenderprogramm zu verschalten.

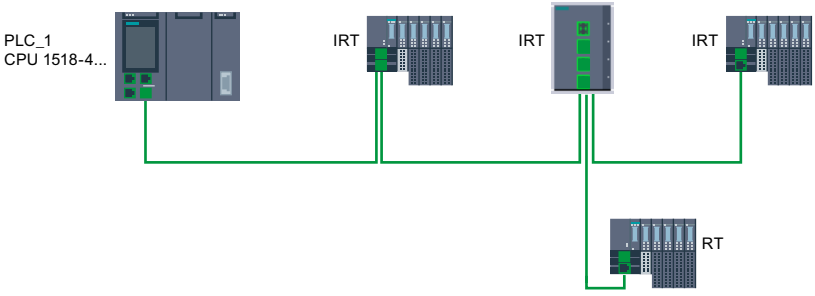

Performance Upgrade

Einführung

Performance Upgrade

Das Performance Upgrade setzt die Application Class "High Performance" der PROFINET Spezifikation V2.3 um.

Das Performance Upgrade bringt eine Reihe von Maßnahmen mit, die für PROFINET mit IRT zu den folgenden Verbesserungen führen:

- Reduzierung der Laufzeitverzögerungen in den IO-Devices
- Bandbreitenerhöhung für zyklische IO-Daten
- Verringerung des Bandbreitenverbrauchs der PROFINET-Telegramme
- Verkleinerung der Sendetakte

Die Verbesserungen durch das Performanceupgrade führen dazu, dass Sie Ihr PROFINET IO-System mit mehr Teilnehmern bei gleichem Sendetakt oder mit einem geringeren Sendetakt bei gleicher Anzahl Teilnehmern betreiben können.

Mit PROFINET war es bisher möglich, Sendetakte von 250 µs zu erreichen.

Mit dem Performance Upgrade ist es nun möglich, mit den Verfahren Fast Forwarding, Dynamic Frame Packing und Fragmentation taktsynchrone Sendetakte von bis zu 125 µs zu erreichen. Bei kurzen Sendetakten ist weiterhin eine Standard-Kommunikation möglich.

Das Performance Upgrade hilft Ihnen, Anwendungen mit hohen Anforderungen an die Geschwindigkeit und Sendetakte zu realisieren.

Dynamic Frame Packing

Dynamic Frame Packing

Bisher werden für jedes PROFINET IO-Device einzelne PROFINET IO-Frames gesendet.

Das Performanceupgrade nutzt für Linientopologien das Verfahren Dynamic Frame Packing, das auf dem Summenrahmenverfahren aufsetzt. Beim Summenrahmenverfahren enthält ein Frame die Nutzdaten für alle Teilnehmer in einer Linie. Beim Dynamic Frame Packing kommt hinzu, dass jedes IO-Device seine Nutzdaten aus dem Frame entnimmt und den Rest weiterleitet. Der Frame verkürzt sich von IO-Device zu IO-Device. Dynamic Frame Packing verbessert die Ausnutzung der Bandbreite in einer Linientopologie.

Das folgende Bild zeigt die Funktionsweise des Verfahrens Dynamic Frame Packing am Beispiel eines Frames, der die Nutzdaten für 3 IO-Devices enthält.

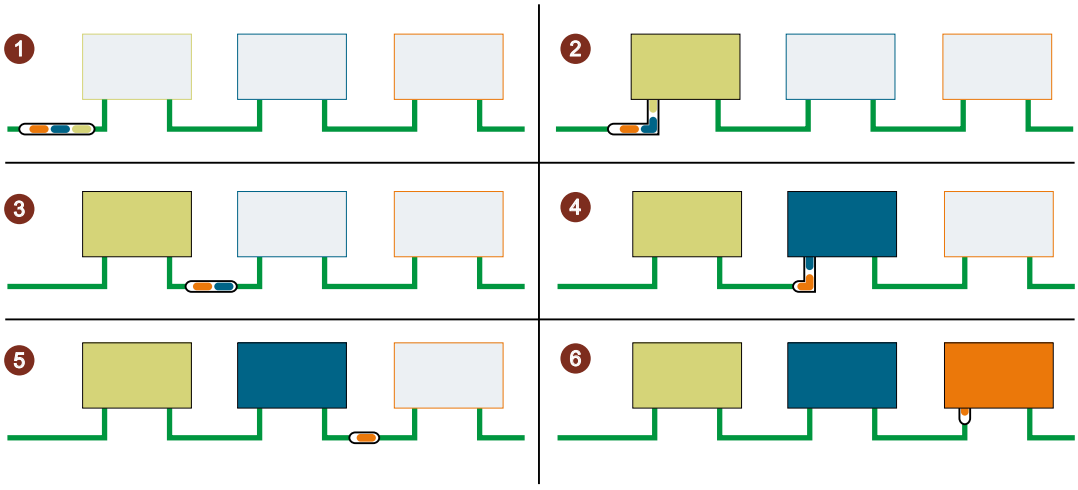

① Der PROFINET IO-Frame enthält die Nutzdaten für alle 3 IO-Devices (grün, blau, orange).

② Der PROFINET IO-Frame erreicht das erste IO-Device. Das IO-Device entnimmt seine Nutzdaten (grün) aus dem Frame und leitet den restlichen Frame weiter.

③ Der PROFINET IO-Frame enthält die Nutzdaten für zwei IO-Devices (blau, orange).

④ Der PROFINET IO-Frame erreicht das zweite IO-Device. Das IO-Device entnimmt seine Nutzdaten (blau) aus dem Frame und leitet den restlichen Frame weiter.

⑤ Der PROFINET IO-Frame enthält die Nutzdaten für ein IO-Device (orange).

⑥ Der PROFINET IO-Frame erreicht das letzte IO-Device. Das IO-Device nimmt den gesamten Frame inklusive der Nutzdaten (orange) auf.

DFP-Gruppen

Dynamic Frame Packing fasst IO-Devices, die das Performance Upgrade unterstützen, automatisch zu DFP-Gruppen zusammen. Um in einer DFP-Gruppe zusammengefasst zu werden, müssen die IO-Devices hintereinander in einer Linie liegen und dieselbe Aktualisierungszeit und Ansprechüberwachungszeit haben. Sobald eine maximale Frame-Größe für die DFP-Gruppe überschritten ist bzw. eine maximale Anzahl von Mitgliedern für eine DFP-Gruppe erreicht ist, öffnet das Dynamic Frame Packing automatisch eine neue DFP-Gruppe.

STEP 7 zeigt die DFP-Gruppen an unter "Domain-Management" > "Sync-Domains" > "Name der Sync-Domain" > "Teilnehmer" im Feld "IO-Devices an.

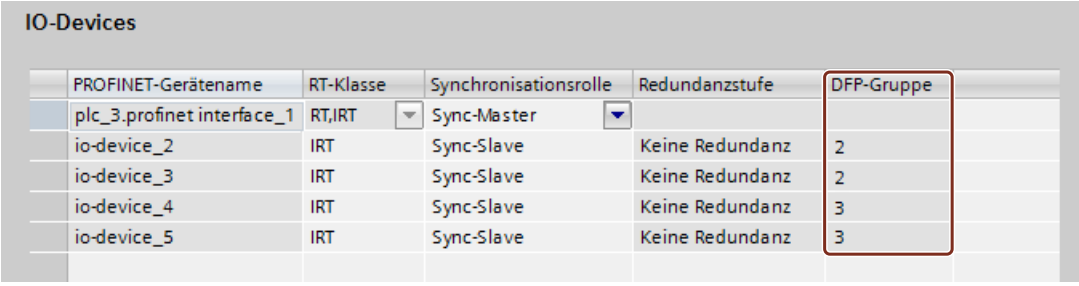

IO-Devices

PROFINET-Gerätename	RT-Klasse	Synchronisationsrolle	Redundanzstufe	DFP-Gruppe
plc_3.profinet interface_1	RT,IRT	Sync-Master		
io-device_2	IRT	Sync-Slave	Keine Redundanz	2
io-device_3	IRT	Sync-Slave	Keine Redundanz	2
io-device_4	IRT	Sync-Slave	Keine Redundanz	3
io-device_5	IRT	Sync-Slave	Keine Redundanz	3

Hinweis

Dynamic Frame Packing bei großen Ansprechüberwachungszeiten

Wenn Sie für ein IO-Device den Parameter "Akzeptierte Aktualisierungsszyklen ohne IO-Daten" auf größer 31 einstellen, dann steckt das Dynamic Frame Packing dieses IO-Device nicht in eine DFP-Gruppe.

Fragmentierung

Fragmentierung

Die Übertragung eines kompletten Standard-Ethernet-Frames mit TCP/IP-Daten dauert bis zu zu 125 µs. Dadurch lässt sich die Zykluszeit für PROFINET-IO Daten nicht beliebig reduzieren.

Das Performanceupgrade nutzt das Verfahren Fragmentierung, das die TCP/IP-Telegramme in Teiltelegramme zerlegt. Diese Teiltelegramme werden über mehrere Sendetakte hinweg zum Ziel-Device übertragen und dort wieder zum original TCP/IP-Telegramm zusammengesetzt.

Durch Fragmentierung lässt sich die Zykluszeit auf Werte kleiner 250 µs verkürzen. In den kürzeren Sendetakten ist die benötigte Bandbreite für TCP/IP-Daten kleiner. Die Bandbreitennutzung für zyklische IO-Daten steigt, d. h. Sie können mehr zyklische IO-Daten in der gleichen Zeit übertragen.

Das folgende Bild zeigt die Funktionsweise des Verfahrens Fragmentierung.

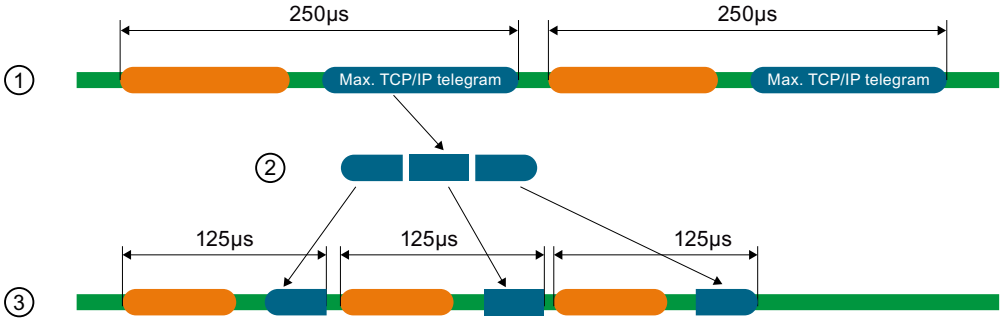

① Ein Standard-Ethernet-Frame mit TCP/IP-Daten ist mindestens 125 µs groß.

② Bei Fragmentierung wird der Standard-Ethernet-Frame in Teiltelegramme zerlegt.

③ Die Teiltelegrame werden auf mehrere kurze Sendetakte aufgeteilt.

Fast Forwarding

Fast Forwarding

Um zu entscheiden, ob ein Telegramm weitergeleitet oder selbst verwendet wird, benötigt ein PROFINET IO-Gerät die Frame-ID. Bis die Frame-ID im IO-Device vorliegt, dauerte es bisher 1440 ns.

Das Performanceupgrade nutzt das Verfahren Fast Forwarding, dabei ist die Frame-ID weiter vorn im Telegramm angeordnet. Die Durchlaufzeit im Device reduziert sich auf 320 ns.

Mit Fast Forwarding verringert sich die Durchlaufzeit der Frames in Ihrem PROFINET IO-System. Insbesondere in Linientopologien, Ringtopologien und Baumtopologien ergeben sich so entscheidende Performancevorteile.

Das folgende Bild vergleicht den Durchlauf eines PROFINET IO-Frames durch ein IO-Device mit Fast Forwarding und ohne Fast Forwarding.

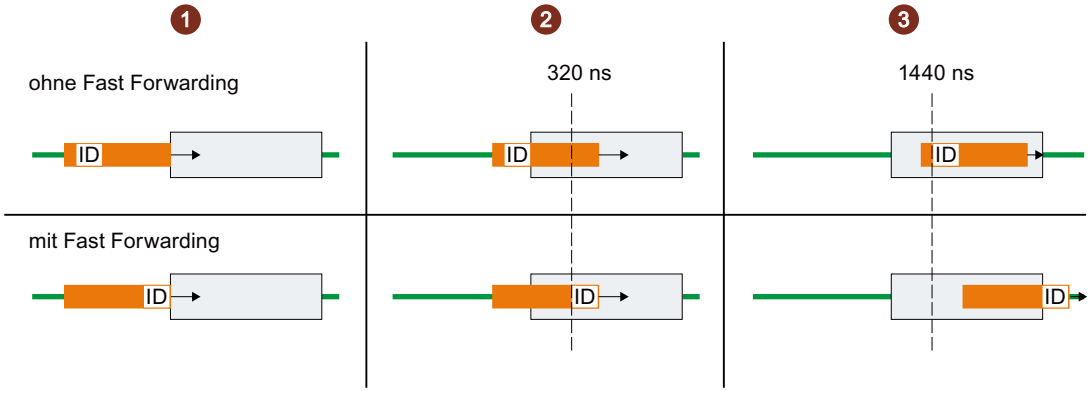

① Start:

- Beide PROFINET-Frames erreichen das IO-Device.
  Die IO-Devices beginnen, die Frames auf die Frame-ID zu überprüfen

② Nach 320 ns:

- Ohne Fast Forwarding: Das IO-Device ist noch dabei, den PROFINET-Frame auf die Frame-ID zu überprüfen.
- Mit Fast Forwarding: Das IO-Device erhält vom PROFINET-Frame die Frame-ID und leitet den Frame weiter.

③ Nach 1440 ns:

- Ohne Fast Forwarding: Das IO-Device erhält vom PROFINET-Frame die Frame-ID und leitet den Frame weiter.
- Der PROFINET-Frame mit Fast Forwarding hat gegenüber dem PROFINET-Frame ohne Fast Forwarding "Vorsprung".

IRT mit höchster Performance projektieren

High End Anwendungen mit IO-Kommunikation erfordern höchste Performance-Ansprüche an die IO-Verarbeitung, z. B. die Steuerung von Windrädern (Converter Control).

Um in Ihrem PROFINET IO-System IRT-Kommunikation mit höchster Performance zu nutzen, aktivieren Sie die Option "High Performance ermöglichen".

Wenn Sie die Option "High Performance ermöglichen" aktivieren, dann hat das folgende Auswirkungen:

- Sie können die Sendetakte 188 µs und 125 µs einstellen (bei CPU 1518).
- Sie können mehr Bandbreitennutzung für zyklische IO-Daten einstellen.
- Sie können die Option "Erlaubt Verwendung von Fast Forwarding" nutzen.

Voraussetzungen

- S7-1500 CPU ab Firmwarestand V2.0

"High Performance ermöglichen" aktivieren

Um "High Performance ermöglichen" zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie das PROFINET IO-System in der Netzsicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "Domain-Management" >"Sync-Domains" > "Name der Sync-Domain".
3. Aktivieren Sie die Option "High Performance ermöglichen".

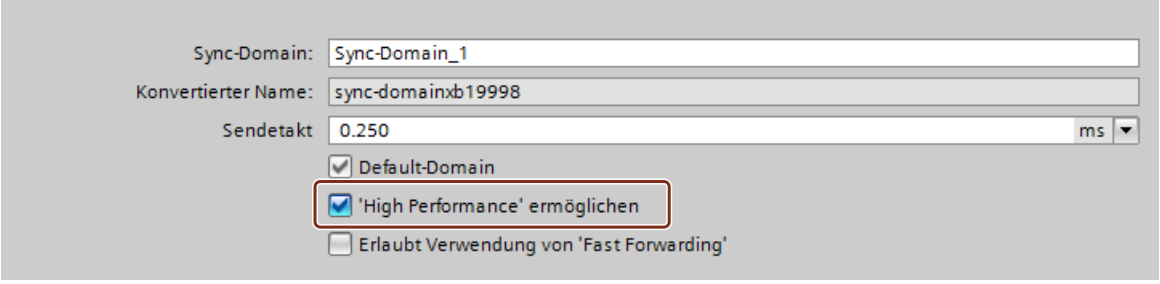

Mehr Bandbreite für zyklische IO-Daten nutzen

Voraussetzung: Sie haben "High Performance ermöglichen" aktiviert.

Um mehr Bandbreite für zyklische IO-Daten für ihr PROFINET IO-System einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie ihr IO-System in der Netzsicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "Domain-Management" >"Sync-Domains" > "Name der Sync-Domain" > "Details".
3. Wählen Sie in der Klappliste "Maximal 90% zyklische IO-Daten. Schwerpunkt auf zyklischen IO-Daten." aus.

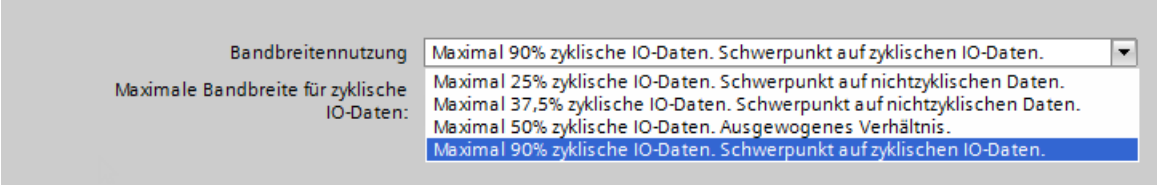

Hinweis

Bandbreitennutzung bei Taktsynchronem Betrieb

Wenn Sie das PROFINET IO-System mit Taktsynchronität betreiben, dann vermeiden Sie die Einstellung für die Bandbreitennutzung "Maximal 90% zyklische IO-Daten. Schwerpunkt auf zyklische IO-Daten".

Kleine Sendetakte einstellen

Voraussetzung: Sie haben "High Performance ermöglichen" aktiviert.

1. Selektieren Sie das PROFINET IO-System in der Netzsicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "Domain-Management" >"Sync-Domains" > "Name der Sync-Domain".
3. Wählen Sie bei "Sendetakt" in der Klappliste den Sendetakt aus.

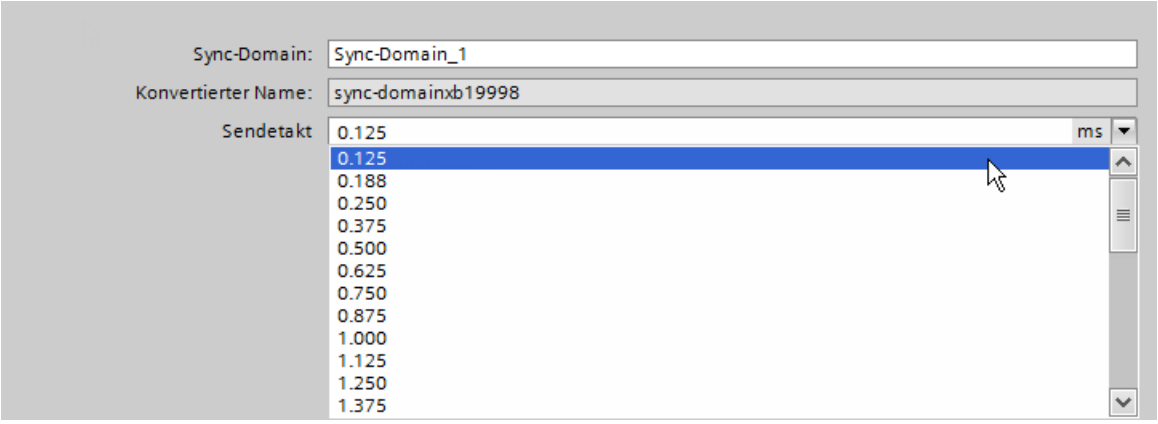

Voraussetzungen für das Verfahren Fragmentierung (CPU 1518-4 PN/DP)

Wenn Sie die folgenden Kombinationen für Sendetakt und Bandbreiteneinstellungen verwenden, dann nutzen die Geräte im IO-System das Verfahren Fragmentierung:

- Sendetakt 125 µs: immer Fragmentierung unabhängig von der Bandbreiteneinstellung.
- Sendetakt 188 µs: Fragmentierung bei den Bandbreiteneinstellungen "Maximal 50% zyklische IO-Daten. Ausgewogenes Verhältnis" und "Maximal 90% zyklische IO-Daten. Schwerpunkt auf zyklische IO-Daten".

Die PROFINET IO-Schnittstelle des Controllers unterstützt Fragmentierung, wenn alle Ports bis auf einen deaktiviert sind.

Fast Forwarding erlauben

Voraussetzungen:

- Das PROFINET IO-Device muss das Verfahren Fast Forwarding unterstützen.
- Sie haben "High Performance ermöglichen" aktiviert.
- Die PROFINET IO-Schnittstelle unterstützt Fast Forwarding, wenn alle Ports bis auf einen deaktiviert sind.

Um Fast Forwarding zu erlauben, gehen Sie folgendermaßen vor:

1. Selektieren Sie das PROFINET IO-System in der Netzsicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "PROFINET" > "Domain-Management" >"Sync-Domains" > "Name der Sync-Domain".
3. Aktivieren Sie die Option "Erlaubt Verwendung von 'Fast Forwarding'".

Hinweis

Fast Forwarding und IPv6

Der Betrieb von Fast Forwarding zusammen mit IPv6 wird nicht unterstützt.

Sobald ein IO-Device im Subnetz eine IPv6-Adresse nutzt, dürfen Sie kein "Fast Forwarding" aktivieren.

Porteinstellungen für kleine Sendetakte optimieren

Sie können die Bandbreitennutzung in Ihrem PROFINET IO-System weiter optimieren, indem Sie zwischen den Geräten Leitungen mit einer Länge < 20 m verwenden bzw. indem Sie eine kleine Signallaufzeit (max. 0,12 µs) einstellen.

Um in STEP 7 Leitungen mit einer Länge < 20 m bzw. die entsprechende Signallaufzeit zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Port in der Topologiesicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Portverschaltung" in das Feld "Partnerport".
3. Aktivieren Sie die Option "Leitungslänge" (Voreinstellung) bzw. "Signallaufzeit".
4. Wählen Sie in der Klappliste "< 20 m" aus bzw. tragen Sie die Signallaufzeit ein.

Siehe auch

Regeln und Hinweise für IRT-Konfigurationen (Seite 3057)

Bandbreitennutzung für synchronisierte IO-Controller (Seite 3054)

Beispielkonfiguration für IRT mit höchster Performance

Das folgende Bild zeigt eine Beispielkonfiguration, mit der Sie die höchste Performance erreichen können.

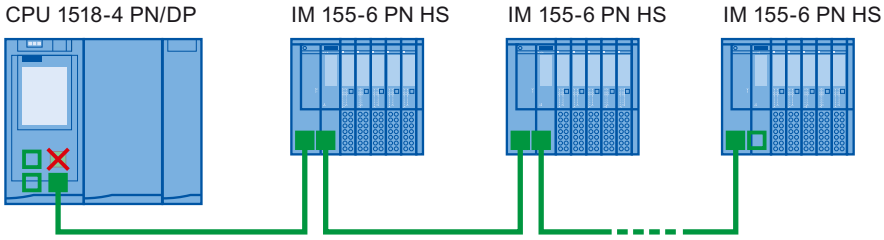

✕ Der Port 2 der Schnittstelle X1 und die Ports der Interfacemodule in der Linie verwenden die folgende Einstellung: Medium Kupfer, Leitungslänge < 20 m.

■ Der Port 2 der Schnittstelle X1 und die Ports der Interfacemodule in der Linie verwenden die folgende Einstellung: Medium Kupfer, Leitungslänge < 20 m.

□ Über die Schnittstellen X2 und X3 der CPU kann eine PG-Verbindung zur CPU hergestellt werden.

Verwenden Sie für die Sync-Domain die folgenden Einstellungen:

- Aktivieren Sie die Option "High Performance ermöglichen".
- Stellen Sie den Sendetakt auf 125 µs ein.
- Stellen die die Bandbreitennutzung auf "Maximal 90 % zyklische IO-Daten. Schwerpunkt auf zyklischen IO-Daten ein".
- Aktivieren Sie die Option "Erlaubt Verwendung von Fast Forwarding".

Standard-Ethernet-Kommunikation bei IRT mit höchster Performance

Auch in einem PROFINET IO System mit höchster Performance ist immer noch Standard-Ethernet-Kommunikation möglich. Beachten Sie, dass Sie, vom IO-Controller aus gesehen, zuerst die IRT-Teilnehmer anordnen und die Standard-Ethernet-Teilnehmer an das Ende der Linie anordnen.

Bei großem Datenaufkommen durch Standard-Ethernet-Kommunikation ist es sinnvoll, Ihr Netz durch Trennung von Standard-Ethernet-Kommunikation und zyklischer Echtzeitkommunikation zu entlasten. Beispiel: Verwenden Sie für das schnelle PROFINET IO System die Schnittstelle X1 und für die Standard-Ethernet-Kommunikation die Schnittstelle X2.

Instandhaltung mit Asset Management Datensätzen

Wissenswertes zum Asset Management bei PROFINET

Der Betrieb von Maschinen und Anlagen ohne detaillierte Kenntnis über Geräte und Assets ist kaum vorstellbar.

Die Instandhaltung braucht dazu Daten – möglichst umfassende und aktuelle Daten.

Dem Wunsch zu größerer Transparenz hinsichtlich der von Anlagenkomponenten zur Verfügung gestellten Daten ist PROFIBUS & PROFINET International (PI) nachgekommen: Die seit PROFIBUS-Zeiten bewährten Identifikations- und Maintenancedaten wurden nochmals erweitert.

Die aktuelle PROFINET-Spezifikation erfüllt diesen Wunsch durch die Definition eines speziellen Datensatzes: des Asset-Management-Records (AMR).

Ziel dieser Definition ist, dass Sie möglichst **alle** zu wartenden Komponenten einer Anlage online erfassen können - nicht nur die Komponenten, die über das PROFINET-Gerätemodell adressierbar und erreichbar sind (Device/Modul/Submodul). PROFINET erreicht nun auch Nicht-PROFINET-Komponenten!

Assets allgemein und Asset Management Datensätze

Assets sind Komponenten (Hardware und Software/Firmware) einer Maschine, z. B. eine Laserschweißmaschine, oder Anlage.

Ein großer Teil dieser Gerätekomponenten ist bereits über bewährte I&M-Funktionen bzw. entsprechende Datensätze (I&M0 bis I&M5) im PROFINET-Kontext identifizierbar: das Gerät selbst sowie seine Module und Submodule. Also alle Komponenten, die über das PROFINET-Gerätemodell adressierbar sind.

Komponenten, die **nicht** über das PROFINET-Gerätemodell adressierbar sind, deren Daten aber für Betrieb und Instandhaltung online erfassbar sein sollen, sind über **Asset Management**-Funktionen identifizierbar. Diese Asset-Management-Daten (kurz: AM-Daten) sind hinterlegt in einer definierten Struktur in einem speziellen Datensatz, dem oben erwähnten **Asset Management Record** (AMR).

Die PROFINET-Guideline "Identification & Maintenance Functions" unterscheidet hier zwischen I&M-Funktionen (I&M-Daten) und Asset Management Funktionen (AM-Daten): In den folgenden Abschnitten geht es ausschließlich um die AM-Daten.

Zu den Komponenten, die Sie durch Asset-Management-Datensätze zusätzlich online auslesen können, gehören sowohl Hardware-Komponenten wie z. B. Rückwandbusmodule eines Geräts als auch Firmware-Komponenten wie z. B. eine Antriebsregelung mit eigener Versionierung.

Anwendungsbeispiele

Sie haben durch Auslesen von Asset-Management-Records die Möglichkeit, während der Installation oder während des Betriebs z. B. folgende Informationen auszulesen:

- Sind nur frei gegebene Betriebsmittel im Einsatz (Whitelist-Check)?
- Ein Firmware-Update steht an. Verschaffen Sie sich einen schneller Überblick: Welche Geräte bzw. Komponenten sind betroffen und müssen hochgerüstet werden?

Asset Management Daten bereitstellen

Das Konzept für Asset Management von PROFINET-Geräten sieht vor, dass die Hersteller von PROFINET-Geräten dafür sorgen, dass Nicht-PROFINET-Automatisierungskomponenten über einen Assets Management Record bereitgestellt werden. Dieser Datensatz (Record) ist dem PROFINET-Gerät zugeordnet.

Im Gegensatz zu einem "Standard" IO-Device obliegt es bei I-Devices dem Projekteur des I-Device, den Asset-Management-Datensatz bereitzustellen. Assets sind in diesem Fall die zentralen Module des I-Device. Diese zentralen Module sind aus Sicht des übergeordneten IO-Controllers aus PROFINET-Sicht nicht sichtbar; der übergeordnete IO-Controller "sieht" nur die Transferbereiche, über die er IO-Daten mit dem I-Device austauscht.

Das Prinzip dieser Bereitstellung ist im nächsten Kapitel erläutert.

Asset Management Daten lesen

Der Asset-Management-Datensatz hat den Index 0xF880 und wird vom Nutzer des Datensatzes, z. B. ein Tool oder Programm zum Auswerten dieser Daten, mit Standard-PROFINET-Mechanismen ausgelesen.

Ein Anwenderprogramm im S7-1500 IO-Controller kann z. B. den AMR von einem IO-Device mit der Anweisung RDREC (Index 0xF880) auslesen.

Schreiben auf diesen Datensatz ist nicht möglich.

Weitere Informationen

Ob und in welchem Umfang ein PROFINET-Gerät Asset Management-Daten unterstützt, d. h., ob es einen AMR zur Verfügung stellt, entnehmen Sie der Dokumentation zum jeweiligen Gerät.

Inhalt und Struktur eines Asset Management Datensatzes

Grundstruktur des Asset-Management-Datensatzes

Zunächst erhalten Sie einen Überblick, wie der Datensatz generell aufgebaut ist. Die folgende Tabelle beschreibt den Rahmen, innerhalb dessen die Asset-Management-Datenblöcke eingebettet sind. Jeder Datenblock repräsentiert ein Asset, z. B. ein Terminal-Block.

Element der Datenstruktur	Bezeichnung nach IEC 61158-6-10	Kodierung	Datentyp/Länge in Bytes
Header AssetManagement-Data	BlockType	0x0035	UINT / 2
	BlockLength	Anzahl Bytes ohne Zählung der Bytes für BlockType und BlockLength	UINT / 2
	BlockVersion	0x0100	UINT / 2
AssetManagementInfo AssetManagementBlocks (n)	NumbersOfEntries	Anzahl AssetManagementBlocks	UINT / 2
	AssetManagementBlock 1	siehe nachfolgende Tabelle	
	AssetManagementBlock 2		
	...		
	AssetManagementBlock n		

Struktur von Asset-Management-Blöcken

Jeder AssetManagementBlock enthält Identifikationsdaten und Lokalisierungsinformation für ein Asset. Ein AssetManagementBlock hat eine Substruktur, die im Folgenden in Grundzügen erläutert ist.

Im Header eines AssetManagementBlocks ist kodiert, welche von den drei möglichen AM-Daten-Zusammentellungen der Datensatz enthält. Entsprechend den unterschiedlichen Gerätetypen stellen Geräte einen passenden BlockType zur Verfügung:

- Komplexe Geräte mit Informationen zur Hardware und Firmware (BlockType "AM_Fullinformation")
- Geräte mit Informationen zur Hardware (BlockType "AM_HardwareOnlyInformation")
- Geräte mit Informationen zur Firmware (BlockType "AM_FirmwareOnlyInformation")

Die Unterscheidung erlaubt eine effiziente Datenstruktur unterhalb des Headers; trotzdem kann der Datensatz eine erhebliche Größe annehmen (max. 64 kbyte, abhängig von der Anzahl Assets, den das IO-Device liefert).

Tabelle 1-153 Struktur AssetManagenentBlock

Element der Datenstruktur	Bezeichnung nach IEC 61158-6-10	Kodierung	Datentyp/Länge in Bytes
Header AssetManagementBlock	BlockType	0x0036 (AM_FullInformation) 0x0037 (AM_FirmwareOnlyInformation) 0x0038 (AM_HardwareOnlyInformation)	UINT / 2
	BlockLength	Anzahl Bytes ohne Zählung der Bytes für BlockType und BlockLength	UINT / 2
	BlockVersion	0x0100	UINT / 2
	Padding	0x0000 (Füllbyte)	USINT / 1
	Padding	0x0000 (Füllbyte)	USINT / 1

Element der Datenstruktur	Bezeichnung nach IEC 61158-6-10	Kodierung	Datentyp/Länge in Bytes
AssetManagementBlock (Aufbau abhängig vom BlockType, hier beispielhaft für AM_Fullinformation)	IM_UniqueIdentifier	Hersteller-erzeugter Unique Identifier (UUID) entsprechend ISO/IEC 9834-8. Wird genutzt als Referenzschlüssel, um dieses Asset eindeutig zu identifizieren. Beispiel: 550c5300-d34a-22b4-11d3-5533991111	Array of Byte / 16
	AM_Location	Beschreibung der Lage des Assets: Entweder steckplatzorientiert ("Slot- and SubslotNumber format") oder hierarchisch ("Twelve level tree format"). siehe nachfolgende Beschreibung	Array of Byte / 16
	IM_Annotation	Herstellerspezifische Notation Beispiel: "Terminal block, Type xyz123 ". Es werden immer 64 byte verwendet, bei kürzeren Zeichenfolgen wird mit Leerzeichen aufgefüllt.	Array of Char / 64
	IM_OrderID	Herstellerspezifische Artikelnummer Beispiel: "6ES7 131-6BF00-0BA0 ". Es werden immer 64 byte verwendet, bei kürzeren Zeichenfolgen wird mit Leerzeichen aufgefüllt.	Array of Char / 64
	AM_SoftwareRevision (nicht bei AM_HardwareOnlyInformation)	Herstellerspezifische SW-Version Beispiel: "V6.3.8 ". Es werden immer 64 byte verwendet, bei kürzeren Zeichenfolgen wird mit Leerzeichen aufgefüllt. Wenn das Asset IM_Software_Revision unterstützt, dann ist AM_Software-Revision mit Leerzeichen gefüllt.	Array of Char / 64
	AM_HardwareRevision (nicht bei AM_FirmwareOnlyInformation)	Herstellerspezifische HW-Version Beispiel: "A4 ". Es werden immer 64 byte verwendet, bei kürzeren Zeichenfolgen wird mit Leerzeichen aufgefüllt.	Array of Char / 64
	IM_Serial_Number	Herstellerspezifische eindeutige produktionsbezogene Nummer. Die Zeichen kommen aus dem sichtbaren Bereich (0x20 ... 0x7E), keine Steuerzeichen. Beispiel: "A78C-1C82 ". Es werden immer 16 byte verwendet, bei kürzeren Zeichenfolgen wird mit Leerzeichen aufgefüllt.	Array of Char / 16
	IM_Software_Revision	Software Version, folgt einem strengen Aufbau (SW-Version-Präfix, z. B.	Array of Byte / 4

Element der Datenstruktur	Bezeichnung nach IEC 61158-6-10	Kodierung	Datentyp/Länge in Bytes
	(nicht bei AM_HardwareOnlyInformation)	"V", Ziffern für Funktionale Erweiterung, Ziffern für BugFix, Ziffern für Internal Change). Beispiel: 'V' 0x01 0x2 0x3 Wenn AM_SoftwareRevision mit Leerzeichen gefüllt ist, sollten Sie IM_Software_Revision auswerten. Wenn das Asset keine SW unterstützt, ist die Kodierung 'V' 0x00 0x00 0x00.	Präfix (Zeichen "V", "R", "P", "U", oder "T"), dann 3 Zeichen "0" bis "9"
	AM_DeviceIdentification	Identifikation des Devices; die Struktur hat folgenden Aufbau: AM_DeviceIdentification.DeviceSubID (bei Siemens z. B. 0x0000) AM_DeviceIdentification.DeviceID (DeviceID vom Hersteller, 0x0000 bis 0xFFFF) AM_DeviceIdentification.VendorID (Beispiel für Siemens-Assets: 0x002A) AM_DeviceIdentification.Organization: Beispiel für Siemens Assets: 0x0000 (PROFINET)	Array of Byte / 8
	AM_TypeIdentification	Hersteller-zugeordnete Typ-Identifikation: 0x0000: Nicht spezifiziert 0x0001: Controller (PLC) 0x0002: PC based 0x0003: IO-Modul, IO-Submodul 0x0004: Kommunikationsmodul / -Submodul 0x0005: Interfacemodul/ -Submodul 0x0006: Aktive Netzkomponente 0c0007: Media attached Unit (Busadapter) 0x0100 bis 0x7FF: Herstellerspezifisch	UINT / 2
	IM_Hardware_Revision (nicht bei AM_FirmwareOnlyInformation)	Version der Hardware (0x0000 bis 0xFFFF) Beispiel: 0x0003 Wenn AM_HardwareRevision mit Leerzeichen gefüllt. ist, sollten Sie IM_Hardware_Revision auswerten.	UINT / 2

AM_Location

Asset Management bei PROFINET unterstützt zwei Formate zur Kodierung der Lage eines Assets:

- Steckplatzorientiertes Format ("Slot- and SubslotNumber format")
- Hierarchisches, d. h. Ebenenorientiertes Format ("Twelve **L**evel **T**ree format" oder kurz **LT**-Format)

Assets, die Teil des PROFINET-Geräts sind, verwenden das steckplatzorientierte Format. Diese Assets sind an die PROFINET Module und Submodule gebunden.

Assets, die sich außerhalb des PROFINET-Geräts befinden, nutzen das hierarchische Format (LT-Format) für die Kodierung der Lage eines Assets.

Diese Assets sind durch ihre Baumebene (Level Tree) lokalisiert. Die Baumebene beginnt mit Level 0. Der Wert von Level 0 gibt Auskunft über die Nähe zum PROFINET-Gerät:

- Wenn das Asset an ein Modul angeschlossen ist, das über das PROFINET-Gerätemodell adressiert werden kann, dann hat Level 0 den Wert 0. Die folgenden Levels (Level 1 bis Level 3) haben dann die Bedeutung von Steckplatznummer, Substeckplatznummer und Kanalnummer. Wenn an dieses Asset weitere Assets angebunden sind, wird der nächste Level 4 genutzt. Die Grenze ist bei Level 11 erreicht.
- Wenn das Asset zum PROFINET-Gerät gehört, aber nicht an ein Modul angeschlossen ist, das über das PROFINET-Gerätemodell adressiert werden kann, dann hat Level 0 einen Wert zwischen 1 und 0x1FF. Beispiel für so ein Asset ist eine Stromversorgung im PROFINET-Gerät. Wenn an diese Stromversorgung ein weiteres Asset angeschlossen ist, z. B. ein Sensor, dann wird die nächste Baumebene zur Lokalisierung dieses Sensors genutzt (Level 1).
- Wenn das Asset sich außerhalb des PROFINET-Gerätes befindet, aber z. B. zur Maschine gehört, in die das PROFINET-Gerät verbaut ist, dann hat Level 0 einen Wert zwischen 0x200 und 0x3FE.

Der Wert 0x3FF für eine Baumebene zeigt an, dass diese Baumebene nicht genutzt wird. D. h. es ist kein weiteres Asset mehr angeschlossen. In diesem Fall müssen alle tieferen Baumebenen bis Level 11 ebenfalls diesen Wert haben.

Beispiel AM_Location steckplatzorientiert

Ein Modulträger und die darauf befindlichen Terminal-Blöcke liefern jeweils AM-Daten. Aus dem Bild sind die Steckplatzzuordnungen ersichtlich.

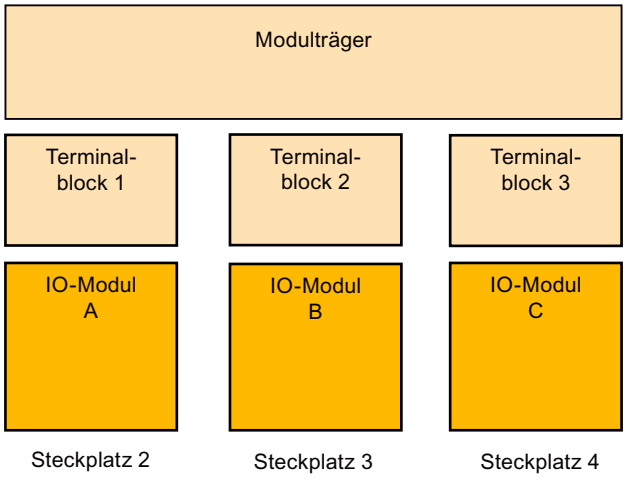

Bild 1-94 Beispiel für Assets mit steckplatzorientierter AM_Location-Kodierung

Für den Modulträger als Asset kodieren Sie AM_Location folgendermaßen:

Bit 0 – 7: AM_Location.Structure = 0x02 (Kodierung "Slot- and SubslotNumber format")

Bit 8 – 15: AM_Location.Reserved1 = 0x00 (Füllbyte)

Bit 16 – 31: AM_Location.BeginSlotNumber = 2 (Das Asset "Modulträger" beginnt ab Steckplatz 2)

Bit 32 – 47: AM_Location.BeginSubslotNumber = 0xFFFF (Das Asset umfasst alle Substeckplätze von Steckplatz 2, ansonsten geben Sie die Nr. des Substeckplatzes an, bei dem das Asset beginnt)

Bit 48 – 63: AM_Location.EndSlotNumber = 4 (Das Asset endet bei Steckplatz 4)

Bit 64 – 79: AM_Location.EndSubslotNumber = 0xFFFF (Das Asset umfasst alle Substeckplätze von Steckplatz 4, ansonsten geben Sie die Nr. des Substeckplatzes an, bei dem das Asset endet)

Bit 80 – 95: AM_Location.Reserved2 = 0x0000 (Füllbyte)

Bit 96 – 111: AM_Location.Reserved3 = 0x0000

Bit 112 – 127: AM_Location.Reserved4 = 0x0000

Beispiel AM_Location ebenenorientiert

An ein IO-Modul (Steckplatz 5, Substeckplatz 1, Kanal 1) ist ein komplexer Sensor angeschlossen. Am komplexen Sensor sind wiederum zwei einfache Sensoren angeschlossen. Das Modul ist innerhalb des PROFINET-Gerätemodells adressierbar, daher hat Level 0 den Wert 0x0000. Die nächste Ebene (Level 1) ist durch den zugeordneten Steckplatz festgelegt, dann folgen die weiteren Ebenen für Substeckplatz und Kanal und ggf. weitere unterlagerte Ebenen.

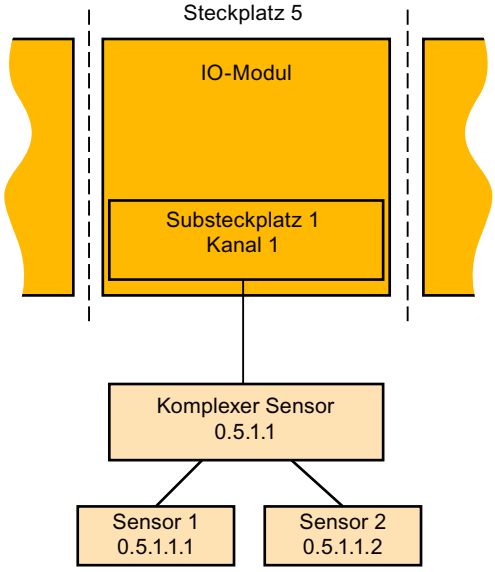

Bild 1-95 Beispiel für Assets mit hierarchischer AM_Location-Kodierung

Detaillierte Kodierung für das Beispiel:

Bit 0 – 7: AM_Location.Structure = 0x01 (LT-Format)

Bit 8 – 17: AM_Location.Level0 = 0x000 (Assets, die Modulen zugeordnet sind, haben immer den Level0-Wert 0x000)

Bit 18 – 27: AM_Location.Level1 = 0x005 (Steckplatz 5)

Bit 28 – 37: AM_Location.Level2 = 0x001 (Substeckplatz 1)

Bit 38 – 47: AM_Location.Level3 = 0x001 (Kanal 1)

Bit 48 – 57: AM_Location.Level4 = 0x3FF (Kodierung für "Level nicht verwendet")

Bit 58 – 67: AM_Location.Level5 = 0x3FF (Kodierung für "Level nicht verwendet")

...

Bit 118 – 127: AM_Location.Level11 = 0x3FF (Kodierung für "Level nicht verwendet")

Im Bild verwendete Notation für die LT-Kodierung Komplexer Sensor: 0.5.1.1

Entsprechend gilt für die übrigen Sensoren:

LT-Kodierung für einfachen Sensor 1 am komplexen Sensor: 0.5.1.1.1

LT-Kodierung für zweiten einfachen Sensor 2 am komplexen Sensor: 0.5.1.1.2

Asset-Management-Datensatz für I-Devices

Sie finden Mit STEP 7 (TIA Portal) ab V15 und mit S7-1500 CPUs ab Firmware V2.5 haben Sie die Möglichkeit, per Anwenderprogramm einen Asset-Management-Datensatz zusammenzustellen. Als I-Device projektiert, liefern diese CPUs dann einem anfordernden IO-Controller die Daten von zentral gesteckten Modulen als Assets.

Mit "S7-1500 CPUs" sind auch die CPU-Varianten S7-1500F, S7-1500T, S7-1500C, S7-1500 SW Controller, S7-1500pro CPUs und ET 200SP CPUs gemeint.

Asset-Management-Datensätze für I-Devices

I-Devices repräsentieren häufig Maschinen. Der PROFINET IO-Controller, dem das I-Device zugeordnet ist, sieht nur die PROFINET-Schnittstelle (als IO-Device projektiert) und die vom Maschinenhersteller projektierten Transferbereiche des I-Device. Die lokalen Module des I-Device sind vom IO-Controller nicht sichtbar bzw. nicht erreichbar.

Mithilfe eines Asset-Management-Datensatzes, den das Anwenderprogramm des I-Device zusammenstellt, kann der zugewiesene IO-Controller die zentralen Module als Assets des I-Device auslesen.

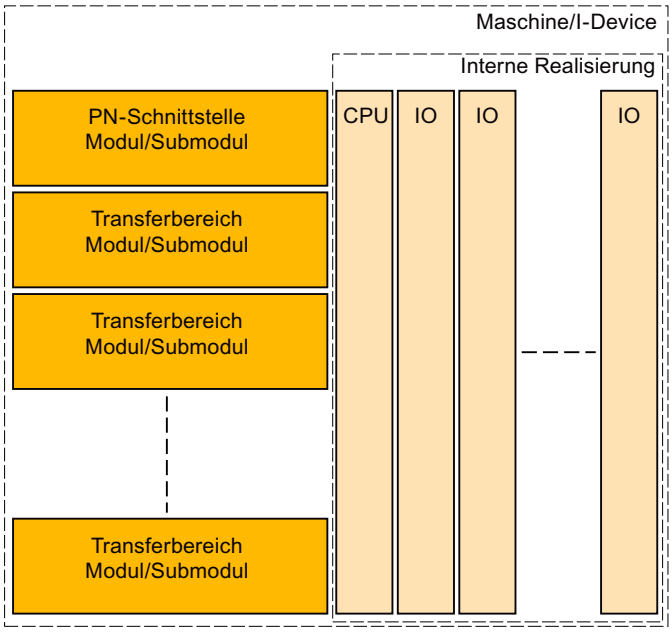

Bild 1-96 Assets eines I-Device

Voraussetzung

- S7-1500-CPU ab Firmware V2.5, als I-Device projektiert
- STEP 7 (TIA Portal) ab V15
- Falls ein IO-Controller den Asset-Management-Datensatz lesen soll:
  Der IO-Controller ist entsprechend programmiert, um einen Asset-Management-Datensatz zu lesen.
  Für einen SIMATIC IO-Controller rufen Sie z. B. eine Lese-Anweisung (RDREC) mit Datensatz-Index 0xF880 auf; die Anweisung adressiert ein beliebiges Submodul des I-Device, z. B. das erste projektierte Transferbereich-Submodul.

Grundsätzliche Vorgehensweise

Um die Voraussetzungen dafür zu schaffen, dass ein I-Device seine lokalen Module als Asset-Management-Datensatz einem anfordernden IO-Controller zur Verfügung stellen, sind grundsätzlich folgende Schritte erforderlich:

1. Einstellungen in den Eigenschaften der PROFINET-Schnittstelle der CPU vornehmen:
   - Betriebsart "IO-Device" aktivieren
   - Option "Asset Management über Anwenderprogramm aktivieren" aktivieren
     Nur wenn die Option aktiviert ist, leitet die PROFINET-Schnittstelle eine Anforderung eines IO-Controllers zum Asset-Management-Datensatz-Lesen weiter an das Anwenderprogramm des I-Device.

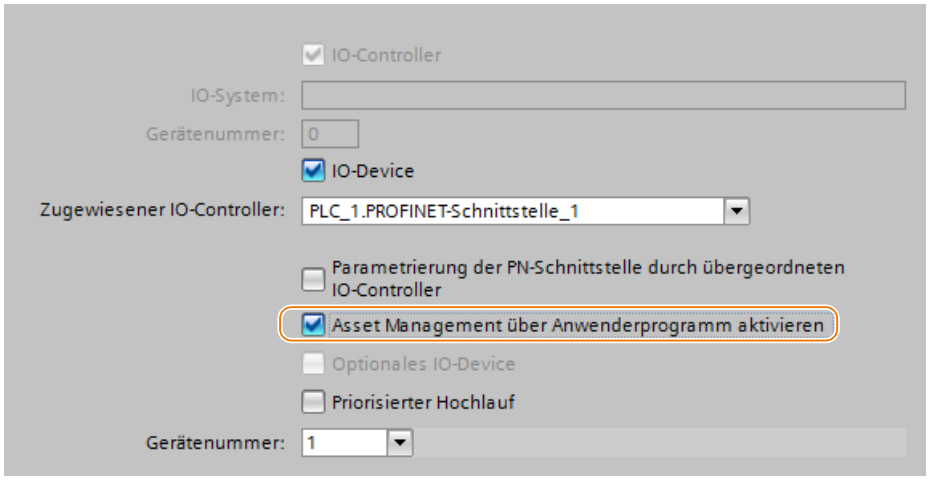

Bild 1-97 Asset Management über Datensatz aktivieren

2. Entwerfen Sie den Programmteil zur Zusammenstellung des Asset-Management-Datensatzes. Der Programmteil sammelt die erforderlichen I&M0-Daten der gesteckten zentralen Module und hinterlegt sie in den dafür vorgesehenen Feldern der Datensatzstruktur des Asset-Management-Datensatzes.
3. Entwerfen Sie den Programmteil zur Koordination der Datensatz-Bereitstellung:
   Rufen Sie dazu die Anweisung PRVREC (Provide Record) nach folgendem Muster in den entsprechenden Modi auf:
   - Zyklischer Aufruf (z. B. im Zyklus-OB) der Anweisung PRVREC mit Mode 0, um die AM-Datensatz-Anforderung zu erkennen.
   - Wenn die AM-Datensatz-Anforderung erkannt ist, muss das Programm PRVREC innerhalb von einer Sekunde quittieren, dass die Anforderung erkannt wurde. D. h., PRVREC muss mit dem Mode 2 aufgerufen werden, und zwar mit dem angeforderten AM-Datensatz. Wenn das I-Device den Zeitrahmen nicht einhält, dann quittiert das I-Device die Datensatz-Anforderung des IO-Contollers negativ!
     Besonderheit für die Parametrierung des PRVREC-Aufrufs: PRVREC müssen Sie mit **F_ID = 0** aufrufen; damit ist kodiert, dass es sich um einen IO-Device-spezfischen Datensatz handelt. Die Output-Parameter SLOT und SUBSLOT liefern daher auch den Wert 0 zurück.

– Innerhalb von 10 Sekunden muss nun der AM-Datensatz fertig gestellt sein und PRVREC mit Mode 3 aufgerufen werden (positive Antwort an den IO-Controller mit Bereitstellung des AM-Datensatzes). Wenn das I-Device den Zeitrahmen nicht einhält, dann quittiert das I-Device die Datensatz-Anforderung des IO-Contollers negativ!

Eine detaillierte Beschreibung der Anweisung PRVREC und mögliche Fehlercodes zur Auswertung der Funktion finden Sie in der Online-Hilfe von STEP 7 (TIA Portal).

Zusammenstellung des Asset-Management-Datensatzes

Sie haben verschiedene Möglichkeiten, den Asset-Management-Datensatz für ein I-Device zusammenzustellen:

- Empfehlung: Empfehlung: Der Siemens Industry Online Support stellt Ihnen eine Applikation bereit, die Ihnen beim Zusammenstellen des Asset-Management-Datensatzes hilft.
  Der Datenbereich des Asset-Management-Datensatzes ist zweigeteilt. Der erste Teil besteht aus einem automatisch ermittelten Bereich, welcher die IM0-Daten der Steckplätze des I-Device in einen Asset-Management-Block verpackt. Der zweite Teil besteht aus den anwenderspezifischen Asset-Management-Blöcken. Die anwenderspezifischen Asset-Management-Blöcke konfigurieren Sie auf Basis vorgefertigter Asset-Management-Datensatzstrukturen, befüllen Sie mit Informationen und stellt sie der Applikation zur Verfügung.
  Die Apllikation leistet folgendes:
  – Die Applikation ermittelt die nötige Größe aller Asset-Management-Blöcke und erzeugt einen neuen Datenbaustein im Ladespeicher der CPU.
  – Die Applikation parametriert den Datenbaustein nach den Vorgaben eines Asset-Management-Datensatzes und befüllt ihn mit den automatisch ermittelten und den von Ihnen übergebenen Asset-Management-Blöcken.
  – Die Applikation stellt dem übergeordneten IO-Controller diesen Asset-Management-Datensatz zur Verfügung.

  Sie finden die Applikation in diesem Anwendungsbeispiel beschrieben
- Sie erstellen den Asset-Management-Datensatz selbst.
  Im Folgenden ist beschrieben, wie Sie einen Asset-Management-Datensatz für ein I-Device selbst zusammenstellen.
  Das Konzept sieht vor, dass Sie für jedes zentral gesteckte Modul die I&M-Daten ermitteln und mit diesen Informationen den Asset-Management-Datensatz füllen. Die I&M0-Daten eines Moduls enthalten Basisinformationen zum Modul wie Herstellerkennung, Artikelnummer, Seriennummer, Hardware- und Firmware-Version. Das sind die Daten, die auch im AM-Datensatz für ein Asset benötigt werden.

I&M-Daten von zentral gesteckten Modulen ermitteln

Der zentrale Aufbau besteht aus einer optional gesteckten Stromversorgung (Steckplatz 0), gefolgt von der I-Device CPU (Steckplatz 1) und dann folgen lückenlos die weiteren Module wie z. B. Digitalmodule, Analogmodule etc. (ab Steckplatz 2).

Die I&M-Daten ermitteln Sie mit der Anweisung "Get_IM_Data" für die gesteckten Module mit Ausnahme der CPU:

Voraussetzung für die Parametrierung der Anweisung "Get_IM_Data" ist die Kenntnis der Hardware-Kennung (Input-Parameter LADDR). Die Hardware-Kennung ermitteln Sie für jeden belegten Steckplatz mit der Anweisung "GEO2LOG" (Aus dem Steckplatz die Hardware-Kennung ermitteln).

Zusammenfassung der prinzipiellen Schritte:

1. Ermitteln Sie in einer Schleife die Hardware-Kennungen der gesteckten Module mit der Anweisung "GEO2LOG".
2. Ermitteln Sie für jede ermittelte Hardware-Kennung die I&M-Daten mit der Anweisung "Get_IM_Data" und speichern Sie diese Daten in einem Datenbaustein, den Sie mit dem Eingangsparameter DATA adressieren. Verwenden Sie für die Datenablage ARRAY of BYTE; das entspricht der Beschreibung der AM-Datensatz-Inhalte im vorausgehenden Kapitel.

Mit ermittelten I&M-Daten AM-Datensatz formen

Die folgenden Ausführungen basieren auf der Beschreibung zum grundsätzlichen Aufbau des AM-Datensatzes, siehe vorangehendes Kapitel.

Weil jedes Modul einer S7-1500 Hardware- und Firmware-Informationen enthält, wählen Sie für den zugeordneten BlockType die Kodierung für "AM_Fullinformation".

Zu den verwendeten Datentypen:

- IM_Annotation, IM_OrderID, AM_SoftwareRevisen und AM_HardwareRevision: Characters (UTF-8)
- IM_SeriaNumber: Charcters ("ASCII-Zeichen") mit Ausnahme der Zeichen für DEL (0x7F)
- Keine String-Datenypen verwenden; sie erfordern zusätzliche Bytes und entsprechen daher nicht den Konventionen der PROFINET-Norm 61158-6-10 Application layer protocol specification

Für jedes Modul bilden Sie den AM_FullInformationBlock auf folgende Weise:

Tabelle 1-154 AM_FullInformationBlock für Module

Datensatz-Element	Beschreibung
IM_UniqueIdentifier	Erzeugen Sie eine (Pseudo-) Zufalls-UUID (Hashwert) entsprechend ISO 9834-8 auf folgende Weise: • Erzeugen Sie einen 8-Byte-Hashwert über die I&M0 Daten des Moduls (ab Steckplatz 2) Verwenden Sie den Algorithmus Fowler-Noll-Vo (kurz: FNV); ein Algorithmus zur Generierung von Streuwerten (Hashwerten) über Datenfelder, siehe entsprechenden Beispielcode im Internet bzw. Online-Support. • Erzeugen Sie einen 8-Byte-Hashwert über die I&M0 Daten der CPU (verwenden Sie den Algorithmus Fowler-Noll-Vo (kurz: FNV) wie oben beschrieben) • IM_UniqueIdentifier: Byte 0 bis 7: Hashwert von Modul-I&M0-Daten Byte 8 bis 15: Hashwert für CPU-I&M0-Daten Erforderliche Anpassungen an ISO 9834-8: Byte 8, Bit 7 muss auf 1 gesetzt werden und Byte 8, Bit 6 auf 0 (Ergebnis der UND-Verknüpfung mit **00**11 1111, anschließend ODER-Verknüpfung mit **10**00 0000) Byte 6, Bit 4 bis 7 müssen auf 0100 gesetzt werden (Ergebnis der UND-Verknüpfung mit 0000 1111, dann ODER-Verknüpfung mit 0001 0000) Dieser Algorithmus erzeugt, da er auf den I&M0-Daten der CPU als auch der Module basiert, einen konstante IM_UniqueIdentifier für ein einzelnes Modul. Wenn sich die Konfiguration ändert, ändert sich auch IM_UniqueIdentifier.
AM_Location	Byte 0 = 0x02 (steckplatzorientierte Kodierung), siehe Beschreibung im vorhergehenden Kapitel.
IM_Annotation	Beispiel: "S7-1500 module" und füllen die restlichen Bytes von IM_Annotation mit Leerzeichen (0x20).
IM_OrderID	Kopieren Sie 20 Bytes der I&M0-Daten des Moduls (Beginnend mit Offset 2 der I&M0-Daten), füllen Sie die restlichen 44 Bytes mit Leerzeichen (0x20)
AM_SoftwareRevision	Füllen Sie das Feld mit 64 Leerzeichen (0x20)
AM_HardwareRevision	Füllen Sie das Feld mit 64 Leerzeichen (0x20)
IM_SerialNumber	Kopieren Sie die 16 Bytes der I&M0-Daten des Moduls (Beginnend mit Offset 22 der I&M0-Daten)
IM_SoftwareRevision	Kopieren Sie 4 Bytes der I&M0-Daten des Moduls (Beginnend mit Offset 40 der I&M0-Daten)
AM_DeviceIdentification	Byte 0, 1, 2, 6, 7 = 0x00 Byte 3 = 0x2A (Vendor = Siemens) Byte 4 = 01, Byte 5 = DeviceID (z. B. CPU 15xx = 0x0E)
AM_TypeIdentification	Kopieren Sie 2 Bytes der I&M0-Daten des Moduls (Beginnend mit Offset 48 der I&M0-Daten)
IM_HardwareRevision	Kopieren Sie 2 Bytes der I&M0-Daten des Moduls (Beginnend mit Offset 38 der I&M0-Daten)

1.4.12 Buskopplung mit PN/PN Coupler

1.4.12.1 Anwendungsgebiet und Funktion

Anwendungsgebiet

Der PN/PN Coupler dient dazu, zwei Ethernet-Subnetze miteinander zu verbinden und Daten auszutauschen. Dabei können Nutzdaten über Eingangs- oder Ausgangsadressbereiche oder Datensätze verwendet werden. Die maximale Größe der übertragbaren Eingangs- und Ausgangsdaten beträgt 1024 Byte. Die Aufteilung auf Eingangsdaten und Ausgangsdaten ist beliebig, so dass z. B. 800 Byte Eingangsdaten und 200 Byte Ausgangsdaten projektiert werden können.

Der PN/PN Coupler hat als ein Gerät zwei PROFINET-Schnittstellen, die jeweils mit einem Subnetz verbunden werden.

In der Projektierung werden aus diesem einen PN/PN Coupler zwei IO-Devices gemacht, und zwar für jede Station mit ihrem Subnetz jeweils ein IO-Device. Der jeweils andere Teil des PN/PN Couplers wird als Koppel-Partner bezeichnet. Mit Abschluss der Projektierung werden die beiden Teile zusammengeführt.

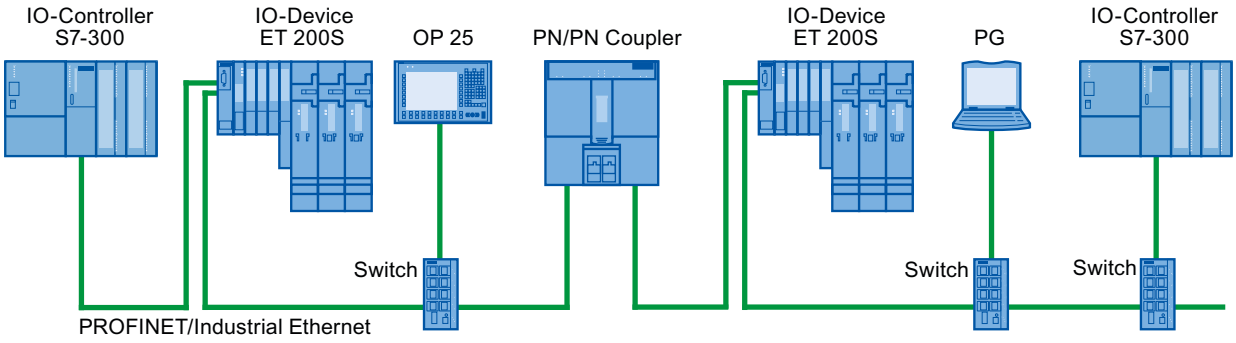

Bild 1-98 Kopplung zweier PROFINET IO-Subnetze mit einem PN/PN Coupler

Weitere Informationen

Beachten Sie für weitere Informationen zum Thema " PN/PN Coupler" den Service & Support im Internet (http://support.automation.siemens.com/WW/view/de/44319532).

1.4.12.2 Ethernet-Subnetze koppeln

Ethernet-Subnetze mit einem PN/PN Coupler koppeln

Mit dem Normgerät PN/PN Coupler können Sie Ethernet-Subnetze miteinander koppeln.

Um Ethernet-Subnetze zu koppeln, gehen Sie folgendermaßen vor:

1. Legen Sie Ihre Ethernet-Subnetze an.
2. Wählen Sie im Hardware-Katalog die Norm-Feldgeräte. Im Ordner "PROFINET IO" finden sie den PN/PN Coupler als Kopfmodul.
3. Ziehen Sie beide Komponenten X1 und X2 der gewünschten Version des PN/PN Coupler per Drag & Drop in die Netzsicht. Die Komponenten bilden ein Gerät, werden aber für die bessere Handhabung getrennt dargestellt.
4. Verbinden Sie die Ethernetschnittstelle des PN/PN Coupler X1 mit dem ersten Ethernet-Subnetz.
5. Verbinden Sie die Ethernetschnittstelle des PN/PN Coupler X2 mit dem zweiten Ethernet-Subnetz.
   Die Ethernet-Subnetze sind nun durch die beiden Komponenten des PN/PN Coupler gekoppelt.

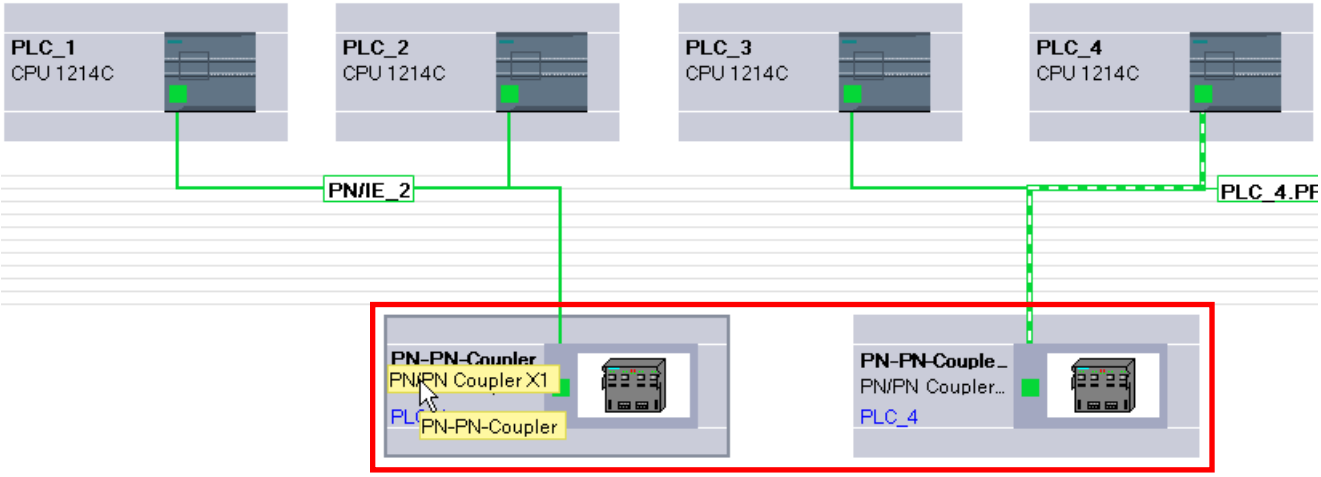

1.4.13 Externe Tools einbinden

1.4.13.1 Einbindung von S7-externen Tools

Einführung

Für das Konfigurieren dezentraler Geräte können STEP 7-externe Tools ("Device Tools") mit einer speziellen Aufrufschnittstelle (Tool Calling Interface) verwendet werden. Solche Geräte werden auch als "TCI-fähig" bezeichnet.

Diese Tools gehen in ihrem Leistungsumfang über die Möglichkeiten der GSD-Projektierung hinaus, z. B. können sie erweiterte grafische Eingabemöglichkeiten bieten.

Als dezentrale Geräte kommen in Frage:

- PROFIBUS DP-Slaves
- Baugruppen innerhalb eines DP-Slaves

- PROFINET IO-Devices
- Baugruppen innerhalb eines IO-Devices

Hinweis

Gewährleistung und Haftung

Siemens übernimmt keine Verantwortung für die per TCI (Tool Calling Interface) aufgerufene Drittsoftware (Device Tools) und für das ordnungsgemäße Zusammenspiel mit den dazugehörigen Geräten.

Voraussetzung

Die Aufrufschnittstelle des Tools genügt der TCI-Spezifikation. Über diese Aufrufschnittstelle werden Parameter und Kommandos an das dezentrale Gerät übergeben.

Solche Tools müssen über ein vom Hersteller bereitgestelltes Setup installiert werden. Ausnahme ist das Device Tool "S7-PCT" (Port Configuration Tool) für IO-Link Master-Module und IO-Link Devices, das mit STEP 7 mitgeliefert wird. Besonderheit: Nach der Installation wird das Tool nicht in der Liste der installierten Software bzw. in der Liste der Software-Produkte im Projekt angezeigt.

Die GSD-Datei des dezentralen Gerätes, das mit dem Device Tool konfiguriert werden soll, muss installiert sein.

Starten des Device Tools

Den Befehl für das Starten des Device Tools finden Sie im Kontextmenü des TCI-fähigen Gerätes im Kontextmenü der grafischen und tabellarischen Gerätesicht: "Device Tool starten".

Siehe auch

SIMATIC S7-PCT starten (Seite 3189)

GSD-Dateien verwenden (Seite 2922)

1.4.13.2 SIMATIC S7-PCT starten

Einführung

Mit STEP 7 wird das Device Tool "S7-PCT" (Port Configuration Tool) installiert.

Das Tool ermöglicht die Parametrierung der Ports von IO-Link Modulen wie 4SI IO-Link (S7-1200, ET 200S) oder 4IOL+8DI+4DO (ET 200eco PN).

Voraussetzung

Sie haben die entsprechende CPU, den DP-Slave oder das IO-Device mit einem IO-Link Modul konfiguriert.

Vorgehen

Für den Start über die grafische Gerätesicht gehen Sie folgendermaßen vor:

1. Markieren Sie das IO-Link Modul in der Gerätesicht.
2. Wählen Sie den Kontext-Menübefehl "Device Tool starten".

ODER gehen Sie für den Start über die tabellarische Geräteübersicht folgendermaßen vor:

1. Markieren Sie das IO-Link Modul in der Gerätesicht.
2. Ordnen Sie die Bereiche im Arbeitsbereich so an, dass die tabellarische Geräteübersicht sichtbar ist (befindet sich zwischen Gerätesicht und Inspektorfenster).
3. Markieren Sie die Zeile mit dem IO-Link Modul in der Geräteübersicht.
4. Wählen Sie den Kontext-Menübefehl "Device Tool starten".

Ergebnis

Das Tool startet und Sie können die Ports konfigurieren.

Siehe auch

Einbindung von S7-externen Tools (Seite 3188)

1.4.14 Konfiguration laden

1.4.14.1 Einführung zum Laden einer Konfiguration

Um ein Gerät in Betrieb zu nehmen, ist es notwendig, dass auf dem PG/PC wie auch auf den angeschlossenen Geräten identische Konfigurationen gespeichert sind. Den Abgleich zwischen PG/PC und den angeschlossen Geräten nehmen Sie über das Laden einer Konfiguration vor. Konfigurationsdaten können in zwei Richtungen geladen werden:

- vom PG/PC in ein Gerät
- vom Gerät in ein PG/PC

Siehe auch

Laden einer Konfiguration in ein Gerät (Seite 3191)

Allgemeines zum Laden in PG/PC (Seite 3192)

1.4.14.2 Laden einer Konfiguration in ein Gerät

Laden der Hardware-Konfiguration

Wenn Sie ein neues Gerät in das Projekt eingefügt und konfiguriert haben, oder wenn Sie eine bestehende Hardware-Konfiguration verändert haben, müssen Sie zunächst die aktuelle Konfiguration in das Gerät laden. Auf diese Weise ist sichergestellt, dass auf dem PG/PC und dem physikalisch vorhandenen Gerät dieselbe Konfiguration eingestellt ist. Verwenden Sie dazu den Menübefehl "Online > Laden in Gerät".

Beim erstmaligen Laden werden die Hardware-Konfiguration vollständig geladen. Bei weiteren Ladevorgängen werden nur noch Änderungen der Hardware-Konfiguration geladen.

Zum Laden der Hardware-Konfiguration stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Laden in der Geräte- oder Netzsicht
- Laden in der Projektnavigation
- Laden in einen erreichbaren Teilnehmer

⚠ WARNUNG

Ladevorgang nur in STOP durchführen

Nach dem Laden kann es bei einer Fehlparametrierung zu unerwartetem Verhalten der Maschine oder des Prozesses kommen. Eine CPU muss für den Ladevorgang zwingend in den Betriebszustand STOP gesetzt werden, um möglichen Schaden an der Anlage oder Personenschäden auszuschließen.

Besonderheiten beim Laden von taktsynchronen Applikationen

Taktsynchrone Applikationen haben einen Hardware-Konfigurations-Anteil und einen Software-Anteil.

Beispiel: Wenn Sie in der Hardware-Konfiguration die Nummer eines IO-Systems, die Verzögerungszeit oder die Zuordnung eines Teilprozessabbildes der taktsynchronen Peripherie ändern, hat das Einfluss auf die Parameter des Taktsynchronalarm-OBs und damit auf den Software-Anteil.

Bei taktsynchronen Applikationen sollten Sie immer das komplettes Projekt laden (Hardware und Software). Beim partiellen Laden (Hardware und Software getrennt zu unterschiedlichen Zeitpunkten) können Inkonsistenzen entstehen, die z. B. für die CPU ein Anlaufhindernis darstellen oder bewirken, dass die Applikation nicht taktsynchron läuft.

1.4.14.3 Laden einer Konfiguration in PG/PC

Allgemeines zum Laden in PG/PC

Einführung

Wenn Sie mit ihrem PG/PC an eine Anlage kommen und das STEP 7-Projekt, mit dem die Konfiguration dieser Anlage erstellt wurde, nicht zur Verfügung steht, dann laden Sie sich die Konfiguration in Ihr PG/PC, z. B. in ein neues Projekt. Verwenden Sie dazu den Menübefehl "Online > Laden des Geräts als neue Station (Hardware und Software)".

Das Laden eines Geräts in Ihr Programmiergerät erfolgt grundsätzlich über die Liste der erreichbaren Teilnehmer in der Projektnavigation. Sie können durch Multiselektion mehrere Geräte auf einmal in das Projekt laden.

Voraussetzungen

- Die Hardware-Konfiguration im Gerät muss im TIA Portal ab V12 erstellt sein. Eine im Gerät vorhandene Hardware-Konfiguration, die mit einer älteren Version erstellt wurde, kann nicht geladen werden und muss hochgerüstet werden.
- Im Gerät vorhandene Module aus GSD (ML), HSPs oder Service Packs müssen im TIA Portal auf dem PG/PC installiert sein.
- Ein Projekt muss geöffnet sein. Dabei kann es sich um ein neues (leeres) Projekt oder ein bereits vorhandenes Projekt handeln.
- Das geöffnete Projekt befindet sich im Offline-Modus.

Ladeumfang

Die folgende Liste zeigt eine Übersicht über die ladbaren Bestandteile einer Konfiguration:

- Das Gerät (z. B. eine CPU) mit allen Peripheriemodulen und allen Parametereinstellungen
- PROFIBUS Mastersysteme und alle PROFIBUS-relevanten Einstellungen
- PROFINET IO-Systeme und alle PROFINET-relevanten Einstellungen
- I-Devices und I-Slaves
- Einstellungen für den Direkten Datenaustausch

Nach dem Laden einer CPU werden automatisch auch alle anderen Module, die sich im Adressraum der CPU befinden, mit geladen.

Die folgenden Verbindungen werden beim Laden der Konfiguration mit geladen:

- S7-Verbindungen (auch geroutet) in gemischten PB/IE-Netzwerken, auch über IE-CP- oder PB-CM-Schnittstellen. S7-Verbindungen werden automatisch als einseitig beim Laden einer Gerätekonfiguration übernommen, auch wenn die S7-Verbindung im Ursprungsprojekt zweiseitig projektiert war. Wenn beide Verbindungspartner geladen sind, wird beim nächsten Übersetzen die Verbindung wieder zusammengeführt.
- TCP-Verbindungen über CPU-eigene Ethernet-Schnittstelle, UDP-/ISOonTCP-Verbindungen, TCP-, UDP-, ISO- und ISOonTCP-Verbindungen über IE-CP-Schnittstelle
- Verbindungen über die OUC-Verbindungsparametrierung für Projekte aus STEP 7 ab V13

Hinweis

Die in PG/PC geladene Hardware-Konfiguration ist nicht vollkommen identisch mit der ursprünglich in das Gerät geladenen Konfiguration. Beachten Sie dazu die weiteren Information zum Laden, vor allem bezüglich teilweise geladener Konfigurationdaten bei geräteübergreifender Kommunikation.

Das Laden von PC-Systemen wie zum Beispiel WinAC oder PC-based Automation ist nicht möglich.

Laden bestimmter Gerätekonfigurationen

Informationen zum Laden

Beim Laden in PG/PC werden alle eingestellten Parameter vom Gerät in das Projekt übertragen. Wenn die CPU an ein Subnetz angeschlossen ist, dann werden alle Parameter des Geräts geladen und die CPU wird in der Netzsicht als vernetzt angezeigt.

Hinweis

Auf ihre Werkseinstellungen zurückgesetzte CPUs haben keine Hardware-Konfiguration, daher wird in diesem Fall nach "Laden in PG/PC" im Menü "Online" nichts geladen.

Laden von Konfigurationen S7-300/400

Um Konflikte beim Laden von einem Gerät in ein bestehendes Projekt zu vermeiden, müssen folgende Regeln erfüllt sein:

- Eindeutigkeit bei Gerätenamen für CPUs, PROFIBUS-Slaves (DP-Slaves, I-Slaves) und PROFINET-Devices (IO-Devices, I-Devices)
- Eindeutigkeit bei der Kombination aus Netzwerkname, Subnetz-ID und IP/DP-Adresse bei Modulen

Bei Konflikten wird der Ladevorgang abgebrochen und eine Meldung informiert über die aufgetretenen Probleme. Sie können dann das Projekt entsprechend anpassen oder fehlende Bestandteile nachinstallieren und danach erneut laden.

Meldungsprojektierungen werden nicht in PG/PC geladen.

Laden von Konfigurationen S7-1200/1500

Berücksichtigen Sie beim Laden Folgendes:

- CPUs der S7-1200 mit dem Firmwarestand V1.0 werden beim Laden nicht unterstützt.
- Gerätespezifische Diagnosealarme der S7-1200 werden beim Laden nicht unterstützt. Damit die gerätespezifischen Diagnosealarme der S7-1200 auf PG/PC neu generiert werden, muss die Hardware-Konfiguration neu übersetzt werden.
- Modul-Kommentare der S7-1200/1500 werden aus dem Gerät in PG/PC geladen, wenn die Projektsprache eingestellt ist, die auch beim Laden in das Gerät verwendet wurde. Sie können das Laden der Kommentare auf Wunsch abwählen.

Hinweis

Alle Arten von PC-Systemen, wie z. B. WinAC, Embedded Controller, CP 1616 oder PC CPs, unterstützen kein Laden nach PG/PC.

Laden Dezentraler Peripherie

Folgende Funktionalitäten und Einstellungen der Dezentralen Peripherie werden geladen:

- DP-Mastersysteme/IO-Systeme mit den zugehörigen DP-Mastern/IO-Controllern (CPUs und CPs), DP-Slaves/IO-Devices, den verwendeten Modulen und deren Parametrierung und Eigenschaften wie z. B. Optionenhandling, Statusbyte oder SYNC/FREEZE
- Anbindung von Teilprozessabbildern (TPA) zu Organisationsbausteinen (OB). Gilt für die Modul- und OB-Eigenschaften
- Konfigurierte Prozessalarme mit den zugehörigen Eigenschaften
- DP-Mastersysteme mit I-Slave
- CPs als PROFIBUS I-Slave oder PROFINET I-Device
- Direkter Datenaustausch

Master-Slave-Beziehungen zwischen I-Slave/I-Device und zugeordnetem DP-Master/IO-Controller werden im Projekt nur dann hergestellt, wenn sowohl der Master als auch der Slave in das PG geladen werden. Ob Sie zuerst den DP-Master/IO-Controller oder zuerst den I-Slave/das I-Device laden, ist unerheblich. Sobald beide Geräte geladen sind, werden auch die Master-Slave/Controller-Device-Beziehungen wieder hergestellt.

Laden von Subnetzen und Teilnehmern mit MPI, PROFIBUS, Ethernet und PtP

Für das Laden von Subnetzen und Endpunkten von Verbindungen für MPI, PROFIBUS, Ethernet und PtP mit ihren jeweiligen Verbindungseigenschaften gelten folgende Besonderheiten:

- Wenn ein Gerät mit PROFIBUS-Schnittstelle geladen wird, weichen die Busparameter des Geräts zunächst von den Einstellungen im Ursprungsprojekt ab. Erst wenn alle beteiligten Geräte geladen sind und sich keine zusätzlichen Geräte am selben Bus befinden, stimmen die Busparameter wieder mit dem Ursprungsprojekt überein.
- Passive Kommunikationsteilnehmer, die als DP-Slaves oder IO-Devices nicht mit einem entsprechenden Mastersystem oder IO-System verbunden sind, nehmen nicht am Datenaustausch teil. Sie werden daher nicht geladen.
- Bei geräteübergreifenden Konfigurationen sollten alle betroffenen Teilnehmer geladen werden. Beim Übersetzen des Projekts wird für fehlende Netzteilnehmer eine Warnung ausgegeben. Auch fehlende Routing-Informationen wegen nicht geladener Kommunikationsteilnehmer werden bei der Übersetzung als Warnung angezeigt. Wird die Konfiguration wieder von PG/PC zurück auf das Gerät geladen, ergeben sich unterschiedliche Routinginformationen.

Wenn Sie nach dem Laden von Geräten in PG/PC das Projekt übersetzen, prüft STEP 7, ob alle Geräte vorhanden sind, zu denen Kommunikationsbeziehungen konfiguriert wurden. Fehlen Geräte, erhalten Sie eine Meldung über die Anzahl der fehlenden Kommunikationsteilnehmer.

ACHTUNG
Geräteübergreifene Kommunikation Wenn Sie eine Konfiguration mit geräteübergreifender Kommunikation in PG/PC laden, müssen Sie auch alle entsprechenden Netzteilnehmer in PG/PC laden. Fehlen benötigte Netzteilnehmer und die Konfiguration wird wieder zurück in das Gerät geladen, ist nicht sichergestellt, dass die geräteübergreifende Kommunikation wieder funktioniert.

Siehe auch

Dezentrale Peripheriesysteme konfigurieren (Seite 2878)

Laden von Konfigurationen mit Webserver

Informationen zum Laden

Die Hardware-Konfiguration einer CPU enthält auch die Webserver-Einstellungen. Das Laden einer Webserver-Konfiguration in PG/PC unterliegt dabei einigen Einschränkungen:

- Die Zuordnung Webserver-Sprache und Projektsprache bei S7-300/400 wird nicht geladen. Die Projekttexte werden nicht geladen und es wird die Meldung ausgegeben, dass keine Projektsprachen zugewiesen sind. Bei S7-1200/1500 CPU werden die in STEP 7 zugewiesenen Sprachen ohne Einschränkung geladen.
- Daten der Benutzerverwaltung der S7-1200/1500 können geladen, aber nicht bearbeitet werden. Sie können über ein Optionskästchen auswählen, ob Sie die vorhandenen Daten schreibgeschützt verwenden oder diese verwerfen und neue Daten eingeben möchten.
- Beobachtungstabellen des Webservers werden nicht geladen.

Die Quelldateien der anwenderdefinierten Webseiten (HTML-Seiten, Java-Scripte etc.) werden nicht geladen. Die beim Laden erzeugten Programmbausteine können nur bearbeitet werden, wenn Sie die Eigenschaften und die HTML-Seite selber eingeben.

Siehe auch

Wissenswertes zum Webserver (Seite 1463)

Laden von Konfigurationen mit PROFIBUS

Informationen zum Laden

Ein DP-Master wird in PG/PC geladen. Das DP-Mastersystem und alle angeschlossenen DP-Slaves werden in das Projekt eingefügt. Die jeweiligen Einstellungen bleiben erhalten. Wenn bereits ein passendes PROFIBUS-Subnetz angelegt ist, werden die geladenen Geräte mit PROFIBUS-Schnittstelle an das vorhandene Subnetz angeschlossen.

Voraussetzung für das Laden von DP-Mastersystemen mit Normslaves ist, dass die entsprechenden GSD-Dateien im TIA Portal installiert und im Hardware-Katalog vorhanden sind. Wenn eine benötigte GSD-Datei nicht in derselben Version wie im Gerät vorhanden ist, werden Unterschiede in der Konsistenzprüfung angezeigt.

Hinweis

Direkter Datenaustausch in einer Konfiguration wird nur geladen, wenn alle am direkten Datenaustausch beteiligten Kommunikationspartner in PG/PC geladen werden.

Taktsynchronität

Beim Laden von DP-Mastersystemen mit der aktivierten PROFIBUS-Funktionalität "Taktsynchronität", achten Sie auf Folgendes:

- Busparameter und die Einstellungen für Taktsynchronität sind nach einem Laden vom Gerät in PG/PC erst identisch, wenn alle für die Berechnung der Taktsynchronität relevanten Teilnehmer geladen sind.
- Es werden nur Mono-Master-Systeme mit Taktsynchronität unterstützt. Geladen werden also nur Konfigurationen mit nur einem DP-Master am PROFIBUS-Subnetz.

I-Slave

Beziehungen zwischen DP-Master und I-Slave werden im Projekt nur dann hergestellt, wenn sowohl der DP-Master als auch der I-Slave in das PG geladen werden. Ob Sie dabei zuerst den DP-Master oder den I-Slave laden, ist dabei unerheblich. Wenn Sie aus einem DP-Mastersystem mit verbundenem I-Slave den DP-Master laden, werden der DP-Master und seine DP-Slaves geladen. Als Platzhalter für einen beteiligten I-Slave wird ein I-Slave-Proxy geladen.

Für I-Slave-Proxy-Geräte gilt:

- DP-Master kann übersetzt und geladen werden
- Eigenschaften werden angezeigt, können aber nicht verändert werden
- Diagnose in der Projektnavigation wird nicht ausgeführt

In der Netzsicht werden Proxy-Geräte mit einem Fragezeichen-Symbol dargestellt:

Das Laden eines DP-Masters mit verbundenem I-Slave-Proxy vom PG/PC in das Gerät ist möglich.

Ersetzen von I-Slave-Proxy

Um den I-Slave im Projekt bearbeiten zu können, müssen Sie den I-Slave vom Gerät in PG/PC laden. Dadurch wird der I-Slave-Proxy durch den vollständigen I-Slave mit dessen unterlagerten DP-Slaves ersetzt.

Sie haben folgende Möglichkeiten, um den I-Slave-Proxy durch den entsprechenden I-Slave zu ersetzen:

- In der Menüleiste der Befehl "Online > Laden des Geräts als neue Station (Hardware und Software)"
- Bei selektiertem I-Slave-Proxy der Kontextmenübefehl "Laden von Gerät > Hardware und Software".
- In der Funktionsleiste die Schaltfläche "Laden von Gerät".

Hinweis

Das Ersetzen von I-Slave-Proxy-Geräten ist nur möglich, wenn der benötigte I-Slave im Hardware-Katalog vorhanden ist.

Durch ein Passwort geschützte I-Slaves werden nicht geladen.

Siehe auch

Konfigurationen mit PROFIBUS-DP (Seite 2874)

GSD-Dateien verwenden (Seite 2922)

Laden von Konfigurationen mit PROFINET

Informationen zum Laden

Wenn Sie eine CPU in der Liste der erreichbaren Teilnehmer markiert haben und Laden in PG/PC ausführen, werden alle zu diesem Gerät gehörenden IO-Controller und IO-Devices mit ihren IO-Systemen geladen. Einstellungen zur Topologie werden ebenfalls übernommen. Wenn bereits ein passendes Ethernet-Netzwerk im Projekt vorhanden ist, werden die geladenen Geräte in das vorhandene Netzwerk integriert.

Beziehungen zwischen IO-Controllern und I-Devices werden im Projekt nur dann hergestellt, wenn sowohl der IO-Controller als auch das I-Device in das PG geladen werden. Ob Sie dabei zuerst den IO-Controller oder die I-Devices laden, ist dabei unerheblich.

Unterstützte Funktionen

Folgende Funktionalitäten und Einstellungen werden geladen:

- PROFINET-Konfigurationen (RT und IRT) in IO-Systemen mit den zugehörigen IO-Controllern (CPUs und CPs), IO-Devices und den verwendeten Modulen
- Logische Adressen und Schnittstelleneigenschaften
- Portverschaltungen
- Taktsynchronität
- Sync-Domains/MRP-Domains
- Redundanzrolle "Client" oder "Manager" bei MRP-Konfigurationen

Hinweis

Leere Sync-Domains und MRP-Domains werden nicht mit hochgeladen.

GSD-basierte IO-Devices

Voraussetzung für das Laden von GSD-basierten IO-Devices ist, dass die entsprechenden GSD-Dateien im TIA Portal installiert und im Hardware-Katalog vorhanden sind. Wenn eine benötigte GSD-Datei nicht in derselben Version wie im Gerät vorhanden ist, werden Unterschiede in der Konsistenzprüfung angezeigt.

I-Device

Wenn Sie aus einem IO-System mit verbundenem I-Device den IO-Controller laden, werden der IO-Controller und seine IO-Devices geladen. Als Platzhalter für ein beteiligtes I-Device wird ein I-Device-Proxy geladen. Dem I-Device-Proxy fehlt die CPU-Parametrierung inklusive der Parametrierung des "eigenen" (untergeordneten) IO-Systems. Lediglich die Schnittstelle zum übergeordneten IO-Controllers wird geladen.

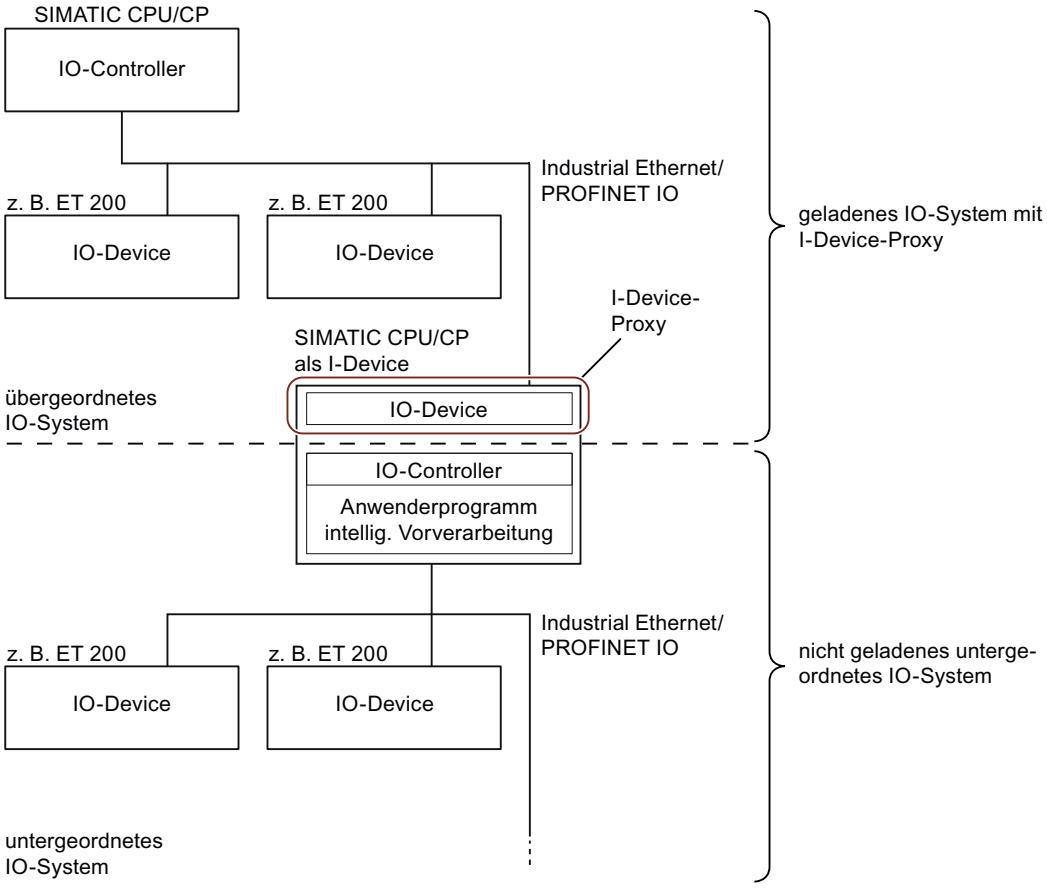

Für I-Device-Proxy-Geräte gilt:

- IO-Controller kann übersetzt und geladen werden
- Eigenschaften werden angezeigt, können aber nicht verändert werden
- Diagnose in der Projektnavigation wird nicht ausgeführt

In der Netzsicht werden Proxy-Geräte mit einem Fragezeichen-Symbol dargestellt:

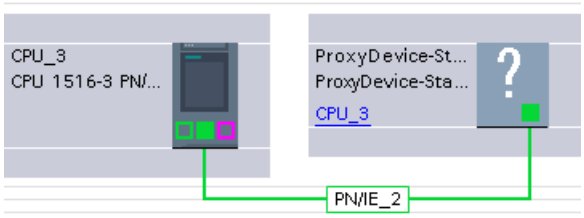

Das Laden eines IO-Controllers mit verbundenem I-Device-Proxy vom PG/PC in das Gerät ist möglich.

Ersetzen von I-Device-Proxy

Um das I-Device im Projekt bearbeiten zu können, müssen Sie das I-Device vom Gerät in PG/PC laden. Dadurch wird der I-Device-Proxy durch das vollständige I-Device mit dessen unterlagerten IO-Devices ersetzt.

Sie haben folgende Möglichkeiten, um den I-Device-Proxy durch das entsprechende I-Device zu ersetzen:

- In der Menüleiste der Befehl "Online > Laden des Geräts als neue Station (Hardware und Software)"
- Bei selektiertem I-Device-Proxy der Kontextmenübefehl "Laden von Gerät > Hardware und Software".
- In der Funktionsleiste die Schaltfläche "Laden von Gerät".

Hinweis

Das Ersetzen von I-Device-Proxy-Geräten ist nur möglich, wenn das benötigte I-Device im Hardware-Katalog vorhanden ist.

Durch ein Passwort geschützte I-Devices werden nicht geladen.

Ein I-Device-Proxy, welcher für ein SIMOTION I-Device steht, kann nicht geladen und ersetzt werden.

Konfigurationen mit IE/PB-Link

Wenn eine der folgenden Konfigurationen mit IE/PB-Link PN IO als PROFINET IO-Device vorliegt, wird die gesamte Konfiguration mit allen unterlagerten PROFIBUS-Teilnehmern geladen:

- CPU/CP der S7-300/400
- PC-Station und angeschlossenem PROFIBUS Mastersystem

Die Gesamtkonfiguration besteht aus:

- CPU
- CP-Konfiguration
- PROFINET IO-System mit angeschlossenem IE/PB-Link
- PROFIBUS Mastersystem des IE/PB-Links mit angeschlossenen DP-Slaves

Eine Beispielkonfiguration besteht aus einer S7-300 CPU mit einem CP als PROFINET IO-Controller. An dem IO-Controller ist ein IE/PB-Link als IO-Device angeschlossen. Der IE/PB-

Link pollt als PROFIBUS DP-Master einen PROFIBUS DP-Slave, zum Beispiel ET 200L .Wenn Sie die CPU aus dem Gerät in PG/PC laden, wird die Gesamtkonfiguration geladen.

Hinweis

Wenn der IE/PB-Link nicht als PROFINET IO-Proxy betrieben wird, sondern als Netzübergang im Standardbetrieb, arbeitet der IE/PB-Link als Zentralbaugruppe und kann separat geladen werden.

Laden von Shared Devices

Für das Laden von Kommentaren gilt: Wenn ein Ein-/Ausgabemodul mit der Funktion Modulinternes Shared Input (MSI) oder Modulinternes Shared Output (MSO) nur aus einem Submodul besteht, hat das Submodul keinen eigenen Kommentar. Es wird stattdessen der Kommentar des Ein-/Ausgabemoduls verwendet. Erst bei einer Aufteilung des Moduls in mehrere Submodule, haben sowohl das Ein-/Ausgabemodul und auch alle Submodule eigene Kommentarfelder.

Laden von HMI-Geräten

Informationen zum Laden

Beim Laden von HMI-Geräten in PG/PC werden folgende Fälle unterschieden:

- HMI-Geräte verbunden mit einem DP-Master als DP-Slave oder mit einem IO-Controller als IO-Device werden als DP-Slave bzw. IO-Device geladen (zum Beispiel PP 17-I PROFIsafe).
- HMI-Geräte in einem Mastersystem als I-Slave oder in einem IO-System als I-Device werden als I-Slave-Proxy bzw. I-Device-Proxy (Seite 3198) geladen (zum Beispiel SIMATIC Comfort Panels). Auf die Einstellungen des Geräte-Proxys ist nur lesender Zugriff möglich.
- HMI-Geräte werden nicht geladen, wenn sie zwar an ein Subnetz (PROFIBUS oder PROFINET), aber nicht an ein Mastersystem oder IO-System angeschlossen sind (zum Beispiel KP600 Basic color DP).

Online verbinden von geladenen Konfigurationen

Sie können mit dem geladenen Projekt oder mit geladenen Teilen des Projekts Online gehen.

Voraussetzung

Die vom Gerät in PG/PC geladene Hardware-Konfiguration wurde übersetzt. Erst nach einer Übersetzung werden die Zustände der zentralen und dezentralen Module korrekt angezeigt.

Hinweis

Wenn Sie sich vor dem Übersetzen online verbinden, wird das Diagnose-Symbol “?” angezeigt (Diagnose nicht möglich). Im Inspektorfenster unter "Info > Allgemein" wird eine entsprechende Meldung angezeigt.

Abhängigkeiten

Je nachdem, wie vollständig die Hardware-Konfiguration in das PG geladen wurde, gibt es Einschränkungen beim Online gehen und bei der Diagnose:

- Vollständig geladenes Gerät mit allen zugehörigen zentralen und dezentralen Baugruppen, wie DP-Slaves oder IO-Devices:
  Online verbinden und Diagnose ist möglich.
- Geladenes Gerät mit angeschlossenen I-Devices/I-Slaves:
  - I-Device/I-Slave ist nicht geladen: Online verbinden für das Gerät und seine Module ist möglich. Für die nicht geladenen abhängigen Bestandteile der Konfiguration werden Geräte-Proxys mit nur minimaler Diagnoseunterstützung behandelt. Der Online-Zustand wird als Icon dargestellt. In der Online- und Diagnosesicht wird die Standarddiagnose gezeigt. I&M-Daten werden nicht geladen.
  - I-Device/I-Slave ist auch geladen: Online verbinden ist für alle Geräte möglich, Diagnose wird voll unterstützt.

1.5 Meldungen anzeigen

1.5.1 Übersicht über die Meldungsanzeige

Mit der Funktion "Meldungsanzeige" können asynchrone Meldungen von Diagoseereignissen und anwenderdefinierte Diagnosemeldungen sowie Meldungen von ALARM-Anweisungen ausgegeben werden.

Bei einem Überlauf von Diagnoseereignissen wird ein Hinweis angezeigt. Sobald der Überlauf beseitigt ist, werden Meldungen des Typs NOTIFY_AP und ALARM_AP nachgemeldet. Nachgemeldet werden nur Diagnoseereignisse, die nach dem Überlauf noch im System aktiv sind. Während des Überlaufs aufgetretene und wieder gegangene Diagnoseereignisse werden nicht nachgemeldet (keine Alarmhistorie).

Außerdem können Sie von der Meldungsanzeige aus über den Kontextmenübefehl "Meldung bearbeiten" den Meldungseditor starten und Anwender-Diagnosemeldungen anlegen.

Symbole

Die folgende Tabelle zeigt die Symbole und ihre Funktionsweise:

Symbol	Funktion
Aktuelle Meldungen	Zeigt die aktuell anstehenden Meldungen an. Quittierpflichtige Meldungen werden in blauer Schrift dargestellt.
Meldungsarchiv	Zeigt die im Archiv befindlichen Meldungen an.
Meldungsarchiv vollständig leeren	Löscht alle Meldungen im Archiv (auch bei aktiviertem Filter).
Archiv exportieren	Exportiert das aktuelle Meldungsarchiv in eine Datei im xml-Format.
Mehrzeilig	Zeigt die Meldungen mehrzeilig an.
Meldungen einfrieren	Friert die Anzeige der Meldungen ein.
Meldung(en) quittieren	Bestätigt die markierte(n) Meldung(en) als gelesen. Zu quittierende Meldungen werden in blauer Schrift angezeigt.
Filter	Zeigt den Filter für die Meldungsanzeige an. Wenn Filterkriterien aktiviert sind, enthält das Symbol ein grünes Häkchen.

1.5.2 Ansicht "Aktuelle Meldungen"

Die Ansicht "Aktuelle Meldungen" ist ein Abbild des Meldequittierspeichers der angewählten CPU(s).

1.5.3 Aufbau der Meldungen in der Ansicht "Aktuelle Meldungen"

Die Ansicht "Aktuelle Meldungen" stellt ein Abbild des Meldequittierspeichers der angewählten CPUs dar. Pro anstehender Meldung wird in der Tabelle ein Eintrag angezeigt. Ereignisse einer Meldung ("Kommen", "Gehen" und "Quittiert") werden in einer Zeile angezeigt.

Aufbau der Tabelle

Alle Attribute der Meldungen können als Spalten dargestellt werden. Sie können die einzelnen Spalten ein- oder ausblenden und die Breite und Reihenfolge der Spalten ändern. Diese Einstellungen werden beim Schließen des Projekts gespeichert.

Sie können Spalten auf- oder absteigend sortieren. Nach welchen Spalten sortiert werden kann, ist davon abhängig, ob die Schaltfläche "Meldungen einfrieren" aktiviert ist oder nicht.

Die Meldungen können ein- oder mehrzeilig dargestellt werden. Bei der einzeiligen Darstellung wird nur die erste Zeile der mehrzeiligen Meldungsdaten angezeigt.

Die Meldungen können quittierpflichtig und nicht quittierpflichtig sein. Die quittierpflichtigen, aber noch nicht quittierten Alarme werden durch blaue Schrift hervorgehoben und können entweder über kontextsensitiv über das Symbol in der Funktionsleiste oder über den Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

Siehe auch

Tabelle in der Meldungsanzeige sortieren (Seite 3209)

1.5.4 Meldungsarchiv

Im Meldungsarchiv werden Meldungen nach ihrem zeitlichen Erscheinen angezeigt und archiviert. Die Archivgröße (zwischen 200 und 3000 Meldungen) können Sie über den Menübefehl "Extras > Einstellungen > Online&Diagnose" festlegen. Ist die eingestellte Archivgröße überschritten, wird jeweils die älteste vorhandene Meldung gelöscht.

Quittierpflichtige Meldungen werden in blauer Schrift angezeigt und können mit dem Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

Das Archiv wird ständig aktualisiert und muss nicht explizit gespeichert werden.

Siehe auch

Meldungsarchiv vollständig leeren (Seite 3205)

Archiv exportieren (Seite 3206)

1.5.5 Aufbau der Meldungen im Meldungsarchiv

Im Meldungsarchiv werden alle Ereignisse, die auf den angewählten CPUs aufgetreten sind, protokolliert. Dabei wird für jedes einzelne Ereignis ein neuer Eintrag erstellt und in der Tabelle als eine weitere Zeile dargestellt.

Aufbau der Tabelle

Alle Attribute der Meldungen können als Spalten dargestellt werden. Sie können die einzelnen Spalten ein- oder ausblenden und die Breite und Reihenfolge der Spalten ändern. Diese Einstellungen werden beim Schließen des Projekts gespeichert.

Sie können Spalten auf- oder absteigend sortieren. Nach welchen Spalten sortiert werden kann, ist davon abhängig, ob die Schaltfläche "Meldungen einfrieren" aktiviert ist oder nicht.

Die Meldungen können ein- oder mehrzeilig dargestellt werden. Bei der einzeiligen Darstellung wird nur die erste Zeile der mehrzeiligen Meldungsdaten angezeigt.

Die Meldungen können quittierpflichtig (quittierpflichtige Meldungen) und nicht quittierpflichtig sein. Die quittierpflichtigen, aber noch nicht quittierten Alarme werden durch blaue Schrift hervorgehoben und können entweder kontextsensitiv über das Symbol in der Funktionsleiste oder über den Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

Siehe auch

Tabelle in der Meldungsanzeige sortieren (Seite 3209)

1.5.6 Meldungsarchiv vollständig leeren

Das Archiv ist als Ringpuffer organisiert, d. h., wenn es gefüllt ist, werden für neue Meldungen die jeweils ältesten Meldungen aus dem Archiv gelöscht. Über das Symbol "Meldungsarchiv vollständig leeren" löschen Sie das gesamte Archiv, unabhängig davon, ob Sie Filterkriterien zur Anzeige bestimmter Meldungen aktiviert haben oder nicht.

Vorgehen

Um das Archiv zu leeren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Meldungsanzeige auf das Symbol "Meldungsarchiv vollständig leeren".

1.5.7 Archiv exportieren

Zum Archivieren von Meldungen können Sie das Archiv exportieren. Gehen Sie dazu folgendermaßen vor:

1. Wechseln Sie in das Meldungsarchiv.
2. Klicken Sie auf das Symbol "Archiv exportieren".
3. Wählen Sie im aufgeblendeten Dialog den Pfad aus, wohin das Archiv exportiert werden soll.

Ergebnis

Das Archiv wird als xml-Datei an die von Ihnen angegebene Stelle gespeichert.

1.5.8 Meldungen empfangen

Damit die Meldungen angezeigt werden, müssen Sie zunächst das Empfangen der Meldungen für jede CPU einstellen.

Vorgehen

Um Meldungen zu empfangen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Ordner "Online & Diagnose" der entsprechenden CPU in der Projektnavigation.
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Online-Zugänge".
3. Aktivieren Sie die Option "Meldungen empfangen".

Hinweis

Wenn Sie diese Vorgehensweise wählen, werden Meldungen erst empfangen, wenn Sie erneut eine Online-Verbindung zum Gerät hergestellt haben.

Oder:

1. Markieren Sie in der Geräte-, Netz- oder Topologiesicht die entsprechende CPU.
2. Wählen Sie im Menü "Online" oder im Kontextmenü den Befehl "Meldungen empfangen".

Oder:

1. Markieren Sie in der Projektnavigation die CPU.
2. Wählen Sie im Menü "Online" oder im Kontextmenü den Befehl "Meldungen empfangen".

Oder:

1. Wählen Sie in der Meldungsanzeige aus der Klappliste "Meldungen anzeigen" die gewünschten Geräte aus, für die Sie Meldungen empfangen möchten.
   - Wenn kein Projekt geöffnet ist, kein online-Gerät enthält oder (noch) kein Gerät ausgewählt ist, wird der Eintrag "Kein Gerät ausgewählt" angezeigt.
   - Wenn Sie auf den Projektnamen klicken, werden alle unterlagerten Geräte markiert und der Eintrag "Alle Geräte ausgewählt" wird angezeigt.
   - Wenn Sie einzelne Geräte im Projekt auswählen, wird der Eintrag "Mehrere Geräte ausgewählt" angezeigt.
2. Bestätigen Sie Ihre Auswahl, indem Sie auf den grünen Haken unterhalb der Klappliste klicken.

Hinweis

Wenn Sie eine der drei oberhalb genannten Vorgehensweisen wählen, müssen Sie zuerst eine Online-Verbindung zum Gerät hergestellt haben.

1.5.9 Meldungen einfrieren

Liefert eine Momentaufnahme der Meldungsanzeige, indem die Anzeige eingefroren wird. Wenn Sie erneut auf diese Schaltfläche klicken, wird die Meldungsaktualisierung wieder aktiviert.

Die Schaltfläche "Meldungen einfrieren" ist nicht aktiviert

Sie können die Meldungen nur nach den Spalten "Datum" und "Uhrzeit" sortieren. Diese Einstellungen werden beim Schließen des Projekts gespeichert. Abhängig von der Sortierreihenfolge werden aktuelle Meldungen am Anfang oder am Ende der Tabelle angezeigt.

Die Schaltfläche "Meldungen einfrieren" ist aktiviert

Sie können jede Spalte der Tabelle sortieren. Wenn Sie die Schaltfläche wieder deaktivieren, geht die eingestellte Sortierung jedoch wieder verloren.

1.5.10 Meldungen quittieren

Quittierpflichtige Meldungen werden in blauer Schrift angezeigt.

Vorgehen

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte(n) Meldung(en) in der Tabelle.
2. Klicken Sie auf die Schaltfläche "Meldungen quittieren".

Hinweis

Sie können mehrere zu quittierende Meldungen gleichzeitig selektieren. Halten sie dazu die Taste <Strg> gedrückt und selektieren Sie dann die gewünschten Meldungen in der Tabelle.

Ergebnis

Die ausgewählte Meldung wurde quittiert und wird in normaler Schrift angezeigt.

Hinweis

In der Ansicht "Aktuelle Meldungen" werden quittierte Meldungen, die bereits gegangen sind, nicht mehr angezeigt.

1.5.11 Status der Meldungen

Abhängig davon, ob Sie sich in der Ansicht "Aktuelle Meldungen" oder im Meldungsarchiv befinden, können die angezeigten Meldungen verschiedene Status haben.

Status der Meldungen in der Ansicht "Aktuelle Meldungen"

- K: Meldung ist gekommen
- KQ: Meldung ist gekommen und quittiert
- KG: Meldung ist gegangen

Wenn mehr Signalwechsel kommen als gesendet werden können (Signal Overflow), wird beim Status ein O angezeigt und der Status wird rot dargestellt.

Status der Meldungen im Meldungsarchiv

- keine Angabe: nur bei Meldungen, die vom PG/PC erzeugt und im Register "Archiv" angezeigt werden, z. B. Anmeldezustand, Verbindungsabbruch, Betriebszustandsänderungen
- K: Meldung ist gekommen
- Q: Meldung ist gekommen und quittiert
- G: Meldung ist gegangen
- L: Meldung wurde gelöscht

Wenn mehr Signalwechsel kommen als gesendet werden können (Signal Overflow), wird beim Status ein O angezeigt und der Staus wird rot dargestellt.

1.5.12 Tabelle in der Meldungsanzeige sortieren

Tabelle auf- oder absteigend sortieren

Um die Tabelle nach einer Spalte auf- oder absteigend zu sortieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Tabellenkopf einer Spalte, wenn Sie diese aufsteigend sortieren möchten.
2. Klicken Sie erneut auf den Tabellenkopf derselben Spalte, um die Spalte absteigend zu sortieren.
3. Klicken Sie ein drittes Mal auf den Tabellenkopf derselben Spalte, um die Sortierung wieder aufzuheben.

Die Schaltfläche "Meldungen einfrieren" ist nicht aktiviert

Sie können die Meldungen nur nach den Spalten "Datum" und "Uhrzeit" sortieren. Diese Einstellungen werden beim Schließen des Projekts gespeichert.

Die Schaltfläche "Meldungen einfrieren" ist aktiviert

Sie können jede Spalte der Tabelle sortieren. Wenn Sie die Schaltfläche wieder deaktivieren, geht die eingestellte Sortierung jedoch wieder verloren.

1.5.13 Tabelle in der Meldungsanzeige filtern

Sie können alle Spalten mit Ausnahme der Spalte "Hilfe" filtern. Dies ist sowohl in den Ansichten "Aktuelle Meldungen" und "Meldungsarchiv" als auch im aktiven Modus (die Meldungen werden ständig aktualisiert) und im Modus "Meldungen einfrieren" möglich.

Sobald ein Filterkriterium aktiviert ist, erscheint im Filtersymbol ein grünes Häkchen. Ist für eine Spalte ein Filterkriterium definiert, ist das entsprechende Feld orangefarben hinterlegt.

Vorgehen

Um Meldungen zu filtern, gehen Sie folgendermaßen vor:

1. Geben Sie ein Filterkriterium in das Feld ein.
2. Bestätigen Sie die Eingabe mit der Eingabetaste oder klicken Sie in einen anderen Bereich, um den Filter zu aktivieren.
   Es werden bis zu 5 eingegebene Filterkriterien pro Spalte in einer Klappliste gespeichert, die Sie dann bei Bedarf auswählen können.
3. Wiederholen Sie den Vorgang ggf. für alle weiteren gewünschten Kriterien.

Ergebnis

Es werden nur die Meldungen angezeigt, die dem eingestellten Kriterium (bzw. den eingestellten Kriterien) entsprechen.

Sind beim Schließen des Projekts Filterkriterien in der Meldungsanzeige gesetzt, so werden diese beim erneuten Öffnen des Projekts (oder eines anderen Projekts) wieder angezeigt. Beim Schließen des TIA-Portals gehen diese Einstellungen jedoch verloren.

1.5.14 Tastaturbedienung in der Meldungsanzeige

Meldungsanzeige

Funktion	Tastenkombination (Shortcut)
Alle Meldungen auswählen	Ctrl+A
Alle ausgewählten Meldungen quittieren	Ctrl+Q

1.6 Zusatzinformationen zu Konfigurationen

1.6.1 Kommunikationsmodule und Netzkomponenten

1.6.1.1 Industrial Ethernet / PROFINET

Einstellungen

Baugruppenzugriffsschutz / UDP-Pufferung / Dateisystem

Bezug

Parametrierung besonderer Einstellungen in der Parametergruppe "Eigenschaften > Allgemein > Einstellungen"

Abhängig vom CP-Typ finden Sie folgende Einstellbereiche, die nachfolgend beschrieben werden:

- Baugruppenzugriffsschutz
- UDP-Pufferung
- Dateisystem

Baugruppenzugriffsschutz - Option "Schutzstufe"

Mit dieser Option können Sie den CP vor unbeabsichtigten oder nicht autorisierten Eingriffen schützen. Folgende Optionen sind in der Klappliste wählbar:

- Nicht gesperrt
- Zustandsabhängig
  In dieser Einstellung kann nur auf den CP zugegriffen werden, wenn sich die CPU im Betriebszustand STOP befindet.

UDP Telegramm-Pufferung ausschalten

UDP-Telegramme werden je nach Kommunikationslast zwischen CP und CPU vom CP zwischengespeichert.

Mit der Option UDP-Pufferung (UDP Telegramm Pufferung ausschalten) können Sie zwischen folgendem Verhalten wählen:

- deaktiviert (Default-Einstellung)
  Die UDP Telegramm-Pufferung ist eingeschaltet. Vom CP empfangene UDP-Telegramme werden so lange zwischengespeichert, bis diese an die CPU übertragen wurden.
- aktiviert
  Der CP überträgt immer nur das zuletzt empfangene, also aktuelle Telegramm an die CPU. Solange zwischen CP und CPU aufgrund der aktuellen Kommunikationslast kein neues UDP-Telegramm übertragen werden kann, wird im CP immer nur das zuletzt eingetroffene Telegramm zwischengespeichert.

Durch das Ausschalten der UDP Telegramm-Pufferung wird eine kürzest mögliche Reaktionszeit zwischen Eintreffen des UDP-Telegrammes und dessen Auswertung in der CPU erreicht. Das mit der Aktivierung der Option verbundene Verhalten kann bei einem hohen Telegrammaufkommen in bestimmten Anwendungsfällen kritisch sein. Durch die Zwischenpufferung von evtl. vielen Telegrammen kann es zu einem unerwünschten Zeitversatz zwischen den in der CPU erfassten Telegrammen gegenüber den aktuell an der Ethernet-Schnittstelle erfassten Telegrammen kommen.

Dateisystem (Groß-/Kleinschreibung beachten) – Option bei CPs mit IT-Funktionen

aktiviert: Der CP unterscheidet bei der Namensgebung von Dateien für den RAM-Bereich zwischen Großschreibung und Kleinschreibung im Dateinamen.

Datentransfer > 240 Byte

Bezug

Parametrierung besonderer Einstellungen in der Parametergruppe "Eigenschaften > Allgemein > Einstellungen"

Abhängig vom CP-Typ finden Sie die Einstellmöglichkeit "Datentransfer > 240 Byte"

Hinweise zum Datentransfer > 240 Byte

- Der Datentransfer > 240 Byte wird von CPs mit neuerem Ausgabestand unterstützt. Beachten Sie hierzu bitte die Angaben in der Gerätedokumentation Ihres Ethernet-CP.
- Beachten Sie, dass Sie mit dieser Projektierung bei S7-300 eine Verbindungsressource (freie Verbindung für S7-Funktionen) der S7-300 CPU beanspruchen! CPU-Verbindungsressourcen werden z. B. auch von S7-300 CPs bei FMS-Betrieb oder von PGs bzw. OPs beansprucht.

Ethernet- / PROFINET-Schnittstelle

Erweiterte Optionen

Keep-Alive-Telegramme für TCP- und ISO-on-TCP Verbindungen

Bezug

Parametrierung besonderer Einstellungen für die PROFINET-Schnittstelle in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Erweiterte Optionen > Schnittstellen-Optionen"

Keep-Alive-Telegramme (nur bei TCP- und ISO-on-TCP Verbindungen)

Sie können hier die Intervallzeit einstellen, mit der Lebenszeichentelegramme (Keep Alive) an den Partner einer Kommunikationsverbindung gesendet werden.

Der Ethernet-CP ist für alle verbindungsorientierten Dienste so konfiguriert, dass Keep-Alive-Telegramme gesendet werden. Dadurch ist gewährleistet, dass Verbindungen nach dem Ausfall eines Kommunikationspartners beendet, und die Verbindungsressourcen freigegeben werden. Die hier vorgenommene Einstellung gilt für alle über den CP betriebenen TCP- beziehungsweise ISO-on-TCP-Verbindungen. Eine verbindungsorientierte Einstellung ist nicht möglich.

Wertebereich

- Voreinstellung: 30 Sekunden
- Maximalwert: siehe aufgeblendete Angabe im Eingabefeld
- Funktion ausgeschaltet: 0 Sekunden

Hinweis

Beachten Sie, dass der Keep-Alive-Mechanismus dazu führen kann, dass unterlagerte Verbindungen (z. B. eine ISDN-Telefonverbindung) aufrecht erhalten werden, obwohl keine eigentlichen Nutzdaten übertragen werden. Wenn dies unerwünscht ist, dann verfahren Sie nach einer der folgenden Alternativen:

- Schalten Sie den Keep-Alive-Mechanismus aus (Intervallzeit = 0).
- Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung nach dem Ausbleiben von Nutzdaten beendet wird, bevor ein Keep-Alive-Telegramm gesendet wird.

IP-Konfiguration

Bezug

Parametrierung für das IP-Protokoll in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"

Bedeutung

Sie können festlegen, über welchen Weg bzw. über welches Verfahren die IP-Adresse der lokalen Schnittstelle ermittelt und dieser zugewiesen werden soll.

Über die angebotenen Varianten wird es möglich, IP-Adressen außerhalb der Projektierung "dynamisch" zuzuweisen.

Sie legen durch Ihre Auswahl auch fest, ob Kommunikationsverbindungen über die Projektierung oder über die Schnittstelle im Anwenderprogramm (Anweisung IP_CONFIG) eingerichtet werden.

Zur Auswahl stehen die folgenden Optionen:

- IP-Adresse im Projekt einstellen
  Dies ist die Standard-Einstellung für PLCs. Sie legen die IP-Adresse während der Vernetzung des Gerätes fest. Die IP-Adresse ist damit fest projektiert. Kommunikationsverbindungen müssen Sie bei dieser Option projektieren.
- IP-Adresse von einem DHCP-Server beziehen
  Sie legen hiermit fest, dass die IP-Adresse beim Hochlauf des Gerätes von einem DHCP-Server bezogen wird.
  Dem DHCP-Server wird hierzu die MAC-Adresse der Schnittstelle oder die in der Projektierung eingebbare Client-ID übermittelt.
  Die Client-ID ist eine Zeichenfolge mit maximal 63 Zeichen.
  Voraussetzungen / Einschränkungen: Wenn Sie diese Option wählen, kann zunächst keine voll spezifizierte Verbindung im Projekt angelegt werden, da ja die lokale IP-Adresse nicht bekannt ist. Sie müssen daher als Verbindungstyp "unspezifiziert" mit passivem Verbindungsaufbau wählen.
- IP-Adresse im Anwenderprogramm einstellen
  Sie legen hiermit fest, dass die IP-Adresse über die Schnittstelle des Anwenderprogrammes (Funktionsbaustein IP_CONFIG) festgelegt wird. Hiermit kann die IP-Adresse im laufenden Betrieb dynamisch versorgt werden.
  Bei diesem Anwendungsfall werden Kommunikationsverbindungen ausschließlich über die Schnittstelle des Anwenderprogrammes angelegt; eine Verbindungsprojektierung ist ausgeschlossen (betrifft Verbindungen über: TCP, ISO-on-TCP, UDP, ISO-Transport).
- IP-Adresse am Gerät einstellen
  Sie legen hiermit fest, dass die IP-Adresse durch andere Dienste außerhalb der Projektierung festgelegt wird.
  Bei diesem Anwendungsfall ist eine Verbindungsprojektierung ausgeschlossen (betrifft Verbindungen über: TCP, ISO-on-TCP, UDP, ISO-Transport).

Hinweis

Wenn Sie bereits Kommunikationsverbindungen über die hier konfigurierte Schnittstelle projektiert haben und eine andere Einstellung wählen als "IP-Adresse im Projekt einstellen", so wird eine Meldung ausgegeben, dass die projektierten Verbindungen nicht mehr funktionieren.

Uhrzeitsynchronisation

Uhrzeitsynchronisation - Verfahren

Bezug

Projektierung des Uhrzeit-Synchronisationsverfahrens des Moduls in der Parametergruppe "Uhrzeitsynchronisation" unter der PROFINET-Schnittstelle oder in den Security-Einstellungen.

Zur Uhrzeitsynchronisation der S7-1200 siehe Kapitel Uhrzeitsynchronisation bei S7-1200 (Seite 3218).

Zur Projektierung der Uhrzeitsynchronisation des IE/PB Link PN IO (V3) siehe Kapitel Uhrzeitsynchronisation - IE/PB LINK PN IO (V3) (Seite 3220).

Arbeitsweise der Verfahren zur Uhrzeitsynchronisation

- **SIMATIC-Verfahren**
  Wenn das SIMATIC-Verfahren aktiviert ist, dann stellt das Modul seine Uhrzeit nach MMS-Uhrzeitnachrichten (MMS - Manufacturing Message Specifaction).
  Die MMS-Uhrzeitnachrichten stammen von einem SIMATIC-Uhrzeitsender oder einer CPU, welche als Uhrzeit-Master projektiert ist. Die Uhrzeitnachrichten können vom Modul (z. B. CP) entweder von der lokalen CPU oder über LAN empfangen werden. Es ist einstellbar, ob das Modul die empfangene Uhrzeit weiterleiten soll.
  Der Vorteil des SIMATIC-Verfahrens liegt in der meist höheren Genauigkeit als beim NTP-Verfahren.
  Siehe auch SIMATIC-Verfahren - Projektierung (Seite 3215).
- **NTP-Verfahren**
  (NTP - Network Time Protocol)
  Beim NTP-Verfahren sendet das Modul Uhrzeitanfragen an einen oder mehrere NTP-Server im Subnetz (LAN). Anhand der Antworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und die Uhrzeit des anfragenden Moduls synchronisiert. Eventuell empfangene MMS-Uhrzeitnachrichten werden in diesem Fall ignoriert.
  Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time).
  Der Vorteil des NTP-Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.
  Siehe auch NTP-Verfahren - Projektierung (Seite 3217).

SIMATIC-Verfahren - Projektierung

Uhrzeitsynchronisation im SIMATIC-Verfahren

Mit der Methode "SIMATIC" wird der CP veranlasst, MMS-Uhrzeitnachrichten weiterzuleiten, wenn er diese von einem SIMATIC-Uhrzeitsender empfängt. Die Uhrzeitnachrichten können entweder über den Rückwandbus bzw. über den K-Bus (S7-400-CPU) oder über LAN empfangen werden.

"Uhrzeitübernahme im CP"

Die Uhrzeit kann vom CP entweder nur empfangen oder empfangen und weitergeleitet werden.

Wenn die Uhrzeit weitergeleitet wird, dann müssen Sie die Richtung der Weiterleitung festlegen.

"Richtung"

Je nachdem, wo der Uhrzeit-Master angeordnet ist, ist die Richtung der Uhrzeitweiterleitung zu beachten.

Wenn Sie in Ihrer Station mehrere CPs betreiben, sollten Sie den Fluss der Uhrzeitnachrichten abhängig vom Uhrzeitmaster bedenken, da Sie über die hier beschriebene Funktion Uhrzeitnachrichten von einem Netz in ein anderes Netz einkoppeln können. In einer Station darf es nur einen einzigen Uhrzeitmaster geben.

Wenn in einer Station mehrere CPs vorhanden sind, die an das selbe Subnetz angeschlossen sind und wenn für diese die Uhrzeit-Weiterleitung eingeschaltet ist, dann gilt: Die Weiterleitung übernimmt der CP mit der jeweils niedrigsten Steckplatznummer. Alle weiteren CPs, die für Uhrzeit-Weiterleitung projektiert sind, stellen die Weiterleitung automatisch ein.

Sie haben folgende Einstellmöglichkeiten für die Richtung der Uhrzeitweiterleitung:

- Automatisch
  Der CP nimmt die Uhrzeitnachricht vom LAN oder aus der Station entgegen und leitet diese in die Station bzw. zum LAN weiter.
  Wenn mehrere CPs in der Station betrieben werden, kann diese automatische Einstellung zu Kollisionen führen. Um dies zu vermeiden, können Sie die Richtung der Weiterleitung mit den folgenden Optionen gezielt festlegen:
- Von Station
  Der CP nimmt die Uhrzeitnachricht von der Station entgegen und leitet sie an das LAN weiter.
  Dazu muss eine lokale CPU als Uhrzeit-Master projektiert werden oder ein anderer CP 400 leitet Uhrzeitnachrichten auf den K-Bus weiter.
- Von LAN
  Der CP nimmt die Uhrzeitnachricht vom LAN entgegen und leitet sie an die Station weiter.
  Dazu muss die lokale CPU als Uhrzeit-Slave projektiert werden.
  Ausnahme: CPU 318

"Korrigierte Uhrzeit verwenden"

Mit dieser Option können Sie entscheiden, ob ein im Uhrzeittelegramm gegebenenfalls übertragener Korrekturfaktor berücksichtigt werden soll.

Der Korrekturfaktor, der in einer Zentraluhr SICLOCK in Einheiten von ½ Stunde gesetzt werden kann, legt einen Zeitversatz bspw. zur Korrektur einer Zeitzonen-Einstellung oder der Sommer-/Winterzeit fest.

Bei aktivierter Option rechnet der CP den Korrekturfaktor in die weiterzuleitende Uhrzeit ein, so dass nachgeschaltete Systeme diesen Faktor nicht mehr berücksichtigen müssen.

Beispiel:

Im Uhrzeittelegramm wird als Uhrzeit 9:21 übertragen. Zusätzlich enthält das Uhrzeittelegramm den Korrekturfaktor +2 x 0,5 (= 1h). Die vom CP weitergeleitete Zeit ist demnach 10:21.

NTP-Verfahren - Projektierung

Erstellen Sie möglichst vor der Projektierung ein Konzept für die Uhrzeitsynchronisation. Achten Sie darauf, dass Sie für jede Station nur eine Uhrzeitquelle verwenden. Sinnvoll ist nur eine Uhrzeitquelle pro Subnetz.

Beachten Sie die Mechanismen der Uhrzeitsynchronisation der S7-1200 im Kapitel Uhrzeitsynchronisation bei S7-1200 (Seite 3218).

NTP - Network Time Protocol

Beim NTP-Verfahren sendet der CP oder die CPU Uhrzeitanfragen an einen oder mehrere NTP-Server im Subnetz (LAN). Anhand der Antworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und die Uhrzeit des anfragenden Moduls synchronisiert. Eventuell empfangene MMS-Uhrzeitnachrichten werden in diesem Fall ignoriert.

Der Vorteil des NTP-Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time).

Die nachfolgend beschriebenen Parameter stehen nicht für alle Module zur Verfügung.

NTP-Server

Sie können die IP-Adressen von bis zu vier NTP-Servern projektieren.

Hinweis

S7-1500-CPUs ab Version V1.8: Änderung der NTP-Server aus dem Anwenderprogramm

Für S7-1500-CPUs ab Firmware-Version V1.8 können Sie die Adressen von 4 NTP-Servern über die Anweisung T_CONFIG (SFB 105) zur Laufzeit ändern. Voraussetzung hierfür ist, dass Sie mindestens einen NTP-Server in der Parametergruppe "Uhrzeitsynchronisation" projektiert haben.

Die projektierten Adressen der NTP-Server werden durch die IP-Adressen der NTP-Server in der Anweisung T_CONFIG überschrieben. Auch wenn nur 1 NTP-Server projektiert wurde, können bis zu 4 NTP-Server über den SFB programmiert werden. Die Adressen der NTP-Server können bei Bedarf mehrmals mit T_CONFIG geändert werden.

Die Firmware-Version V1.8 für S7-1500-CPUs steht mit STEP 7 V13.0 SP1 Update 3 zur Verfügung.

"Zeitzone"

Durch die Projektierung der lokalen Zeitzone kann der Zeitoffset gegenüber UTC eingestellt werden.

"Synchronisationsintervall"

Das Intervall legt den zeitlichen Zyklus der Uhrzeitanfragen (in Sekunden) fest. Der Wertebereich des Intervalls liegt zwischen 10 Sekunden und einem Tag.

Bei aktivierter Option "Uhrzeitsynchronisation zur vollen Minute" wird der Zyklus automatisch auf 60 Sekunden gesetzt.

"Uhrzeitsynchronisation zur vollen Minute"

Bei aktivierter Option wird die synchronisierte Uhrzeit, die der CP an die CPU weiterleitet, genau zur vollen Minute von der CPU übernommen.

Bei aktiverter Option "Uhrzeitsynchronisation zur vollen Minute" wird der Parameter "Aktualisierungsintervall" automatisch auf 60 Sekunden gesetzt.

"Uhrzeit von nicht synchronisierten NTP-Servern annehmen"

Bei aktivierter Option nimmt der CP die Uhrzeit auch von nicht synchronisierten NTP-Servern mit Stratum ≥ 15 an.

Bei deaktivierter Option nimmt der CP keine Uhrzeit von NTP-Servern mit Stratum ≥ 15 an. Wenn der CP ein Uhrzeit-Telegramm von einem NTP-Server mit Stratum ≥ 15 empfängt, wird die Uhrzeit des CP nicht danach gestellt.

"Uhrzeit an Station weiterleiten"

Beim CP 300/400 ist diese Option in der Voreinstellung aktiviert.

Einige CPUs bieten die Möglichkeit, die Uhrzeit eigenständig von einem NTP-Server anzufordern. Wenn die Uhrzeitsynchronisation durch die CPU genutzt wird, sollten Sie die hier beschriebene Option im CP deaktivieren. Sie vermeiden damit, dass die in der CPU vom NTP-Server direkt erfasste Uhrzeit von der im CP erfassten Uhrzeit wieder überschrieben wird. Durch die Weiterleitung der Uhrzeit vom CP an die CPU könnte sich in diesem Fall eine geringere Genauigkeit ergeben.

Uhrzeitsynchronisation bei S7-1200

Module und Synchronisationsverfahren

Bei Verwendung einer externen Uhrzeitquelle kann die S7-1200-Station die aktuelle Uhrzeit sowohl über die CPU als auch über einen CP beziehen.

Eine Weiterleitung der Uhrzeit von der Station an das Subnetz findet bei S7-1200 nicht statt.

Uhrzeitsynchronisation über die CPU

Als Synchronisationsverfahren kann bei der CPU nur NTP ausgewählt werden.

Über die Option "CPU synchronisiert die Module des Gerätes" können Sie veranlassen, dass alle intelligenten Module der Station (CMs, CPs) in einem einstellbaren

Synchronisationszyklus von 10 bis 86400 Sekunden (24 Stunden) mit der CPU-Zeit synchronisiert werden.

Hinweis

Uhrzeitsynchronisation bei Nutzung von Secure Open User Communication (Secure OUC)

Wenn die S7-1200 Secure OUC-Bausteine verwendet und für die Kommunikation die PROFINET-Schnittstelle des CP nutzt, muss die Option "CPU synchronisiert die Module des Gerätes" aktiviert sein.

Nur wenn Datum und Uhrzeit aktuell sind, können diese Daten gegen die Gültigkeitsdauer eines Zertifikats validiert werden. Wenn sich Datum und Uhrzeit außerhalb der Gültigkeitsdauer befinden, kommt keine Kommunikation über Secure OUC zustande.

Uhrzeitsynchronisation über einen Telecontrol-CP

Folgende Synchronisationsverfahren stehen zur Auswahl:

- **NTP**
  NTP kann nur verwendet werden, wenn die Telecontrol-Kommunikation deaktiviert ist. In diesem Fall wird der Telecontrol-CP als erweiterte PROFINET-Schnittstelle der CPU eingesetzt.
- **Uhrzeit vom Partner**
  Wenn die Telecontrol-Kommunikation aktiviert ist, übernimmt der CP die Uhrzeit automatisch vom Kommunikationspartner.

Hinweis

Uhrzeitweiterleitung an die CPU

Abhängig von der Firmware-Version der beteiligten Module wird die Uhrzeit des CP unterschiedlich an die CPU weitergeleitet:

- Optionale Weiterleitung der CP-Zeit an die CPU über eine PLC-Variable
- Obligatorische Weiterleitung der CP-Zeit über den Rückwandbus an die CPU

Uhrzeitweiterleitung vom CP an die CPU

Hinweis

Empfehlung: Uhrzeitsynchronisation nur durch 1 Modul

Lassen Sie die Uhrzeit der Station von einer externen Uhrzeitquelle nur durch ein einziges Modul der Station synchronisieren, um innerhalb der Station eine konsistente Uhrzeit vorzuhalten.

Wenn die CPU die Uhrzeit vom CP übernimmt, dann deaktivieren Sie die Uhrzeitsynchronisation der CPU.

Die Weiterleitung der CP-Zeit an die CPU ist abhängig von der Firmware-Version des CP und der CPU. Beachten Sie das nachfolgende Verhalten.

- **CP-Firmware ≤ V2.1.5**
  Mit dieser Firmware-Version kann die CP-Uhrzeit optional der CPU über eine PLC-Variable zur Verfügung gestellt werden. Wenn diese PLC-Variable zyklisch von der CPU gelesen wird, übernimmt die CPU die CP-Zeit.
- **CP-Firmware ≥ V2.1.7 und CPU-Firmware ≥ V4.2**
  Wenn beide Module in einer Station eine der genannten Firmware-Versionen aufweisen, wird die Uhrzeit des CP automatisch an die CPU weitergleitet. Da die CPU automatisch die CP-Zeit übernimmt, benötigen Sie die Weiterleitungsoption über die PLC-Variable nicht mehr.

Wenn bei der CPU die Option "CPU synchronisiert die Module des Geräts" aktiviert ist, werden alle intelligenten Module der Station mit der CPU-Zeit synchronisiert.

Uhrzeitsynchronisation - IE/PB LINK PN IO (V3)

Bezug

Projektierung der Uhrzeitsynchronisation des IE/PB LINK PN IO V3

Voraussetzung ist, das die Uhrzeit am entsprechenden Netz von einem Teilnehmer zur Verfügung gestellt wird.

Verfahren der Uhrzeitsynchronisation

Folgende Verfahren stehen mit verschiedenen Optionen zur Auswahl:

- **SIMATIC-Verfahren**
  Wenn das SIMATIC-Verfahren aktiviert ist, dann stellt das LINK seine Uhrzeit nach MMS-Uhrzeit-Telegrammen (MMS - Manufacturing Message Specifaction).
  Die MMS-Uhrzeitnachrichten stammen von einem SIMATIC-Uhrzeitsender oder einer CPU, welche als Uhrzeit-Master projektiert ist. Die Uhrzeitnachrichten können vom CP entweder von der lokalen CPU oder über LAN empfangen werden. Es ist einstellbar, ob und in welche Richtung das LINK die Uhrzeit-Telegramme weiterleiten soll.
  Der Vorteil des SIMATIC-Verfahrens liegt in der meist höheren Genauigkeit gegenüber dem NTP-Verfahren.
- **NTP-Verfahren**
  (NTP - Network Time Protocol)
  Beim NTP-Verfahren stellt das LINK seine Uhrzeit nach Uhrzeitnachrichten, die von einem NTP-Server stammen und von einem Teilnehmer am Ethernet-Netz zur Verfügung gestellt werden. Es ist einstellbar, ob das LINK die Uhrzeit-Telegramme vom LAN nur übernehmen oder übernehmen und weiterleiten soll.
  Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time).
  Der Vorteil des NTP-Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

Projektierung der Uhrzeitsynchronisation

Bedeutung der Optionen:

- **SIMATIC-Verfahren anwenden - von Ethernet an PROFIBUS weiterleiten**
  Das LINK übernimmt die Uhrzeit vom LAN und leitet sie weiter an das PROFIBUS-Netz.
- **SIMATIC-Verfahren anwenden - von PROFIBUS an Ethernet weiterleiten**
  Das LINK übernimmt die Uhrzeit vom PROFIBUS-Netz und leitet sie weiter an das LAN.
- **SIMATIC-Verfahren - Uhrzeit vom Ethernet-Netz übernehmen**
  Das LINK übernimmt die Uhrzeit vom LAN, leitet sie aber nicht weiter.
- **SIMATIC-Verfahren - Uhrzeit vom PROFIBUS-Netz übernehmen**
  Das LINK übernimmt die Uhrzeit vom PROFIBUS-Netz, leitet sie aber nicht weiter.
- **NTP-Verfahren anwenden - von Ethernet an PROFIBUS weiterleiten**
  Das LINK übernimmt die Uhrzeit vom LAN und leitet sie weiter an das PROFIBUS-Netz.
- **NTP-Verfahren - Uhrzeit vom Ethernet-Netz übernehmen**
  Das LINK übernimmt die Uhrzeit vom LAN, leitet sie aber nicht weiter.

Zusatzinformationen

Individuelle Netzwerkeinstellungen

Individuelle Netzwerkeinstellungen

Sie können hier bei Bedarf feste Netzwerkeinstellungen vornehmen. Standardmäßig ist die "Automatische Einstellung" gewählt, die im Normalfall eine problemlose Kommunikation gewährleistet.

Falls Probleme bei der Kommunikation auftreten (beispielsweise wenn Verbindungen nicht aufgebaut werden, häufige Netzstörungen auftreten) kann dies daran liegen, dass die gewählte oder die automatische Netzwerkeinstellung nicht passend sind. Wählen Sie dann hier eine auf Ihre Netzkonfiguration abgestimmte Netzwerkeinstellung.

Wertebereich und Beispiel für IP-Adresse

Wertebereich für IP-Adresse

Die IP-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind.

Schreibweise: xxx.yyy.zzz.www

Beispiel:

141.80.0.16

- Folgende Adressen haben Sonderbedeutungen (* steht für eine beliebige Zahl zwischen 0 und 255):
  - 127.*.*.* Loopback
  - 0.0.0.0 unspezifiziertes Netz
  - 255.255.255.255 Broadcast an alle Teilnehmer (nur zulässig, wenn als Verbindungspartner "Alle Broadcast-Teilnehmer" gewählt wurde)
  - Host-Adresse 255 Broadcast-Adresse für alle Teilnehmer eines Netzes:
    bei Klasse A: xxx.255.255.255
    bei Klasse B: xxx.yyy.255.255
    bei Klasse C: xxx.yyy.zzz.255
- Nicht erlaubt sind:
  - Host-Adresse 0
    bei Klasse A: xxx.0.0.0
    bei Klasse B: xxx.yyy.0.0
    bei Klasse C: xxx.yyy.zzz.0

Wertebereich und Beispiel für Subnetzmaske

Wertebereich für Subnetzmaske

Um in der IP-Adresse die Netz-ID von der Host-Adresse zu trennen, wurde die Subnetzmaske eingeführt.

Die Subnetzmaske besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z. B. 255.255.0.0

Die 4 Dezimalzahlen der Subnetzmaske müssen in ihrer binären Darstellung von links eine Folge von lückenlosen Werten "1" und von rechts eine Folge von lückenlosen Werten "0" enthalten.

Die Werte "1" bestimmen die Netznummer innerhalb der IP-Adresse. Die Werte "0" die Host-Adresse innerhalb der IP-Adresse.

Beispiel:

richtige Werte: 255.255.0.0 D = 1111 1111.1111 1111.0000 0000.0000 0000 B

255.255.128.0 D = 1111 1111.1111 1111.1000 0000.0000 0000 B

255.254.0.0 D = 1111 1111.1111 1110.0000 0000.0000.0000 B

falscher Wert: 255.255.1.0 D = 1111 1111.1111 1111.0000 0001.0000 0000 B

Zusammenhang IP-Adresse und Subnetzmaske

Die erste Dezimalzahl der IP-Adresse (von links) bestimmt den Aufbau der Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt (für "x" steht die Host-Adresse bzw. die "0" in der Subnetzmaske):

- Erste Dezimalzahl der IP-Adresse Subnetzmaske
  Klasse A: 0 bis 126 255.x.x.x
  Klasse B: 128 bis 191 255.255.x.x
  Klasse C: 192 bis 223 255.255.255.x

Hinweis

Für die erste Dezimalzahl der IP-Adresse können Sie auch einen Wert zwischen 224 und 255 eintragen. Dies ist jedoch nicht empfehlenswert, da für diese Werte keine Adreßprüfung durch STEP 7 erfolgt.

Zusammenhang IP-Adresse und Adresse des Netzüberganges

Zusammenhang IP-Adresse und Adresse des Netzübergangs

Die IP-Adresse und die Adresse des Netzübergangs dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

Beispiel:

Sie haben eingegeben: für Subnetzmaske 255.255.255.0; für IP-Adresse 141.30.0.5 und für die Adresse des Netzübergangs 141.30.128.1.

	Übliche dezimale Darstellung	Binäre Darstellung
Subnetzmaske	255.255.255.0	1111 1111.1111 1111.1111 1111.0000 0000
IP-Adresse	141.30.0.5	1000 1101.0001 1110.0000 0000.0000 0101
Netzübergang	141.30.128.1	1000 1101.0001 1110.1000 0000.0000 0001

Die IP-Adresse und die Adresse des Netzübergangs dürfen nur in der 4. Dezimalzahl einen unterschiedlichen Wert haben. Im Beispiel ist aber die 3. Stelle schon unterschiedlich.

Im Beispiel müssen Sie also alternativ ändern:

die Subnetzmaske auf: 255.255.**0**.0 oder

die IP-Adresse auf: 141.30.**128**.5 oder

die Adresse des Netzübergangs auf: 141.30.**0**.1

Port Parameter

Individuelle Netzwerkeinstellungen für jeden Port

Sie können bei Bedarf für jede verfügbare Schnittstelle (Port) feste Netzwerkeinstellungen für die Übertragungseigenschaften vornehmen. Standardmäßig ist die "Automatische Einstellung" gewählt, die im Normalfall eine problemlose Kommunikation gewährleistet.

Falls Probleme bei der Kommunikation auftreten (beispielsweise wenn Verbindungen nicht aufgebaut werden, häufige Netzstörungen auftreten) kann dies daran liegen, dass die gewählte oder die automatische Netzwerkeinstellung nicht passend sind. Wählen Sie dann hier eine auf Ihre Netzkonfiguration abgestimmte Netzwerkeinstellung.

Arbeitsweise bei automatischer Umschaltung

Der CP bietet auf allen Ports einen 10/100 MBit/s Full Duplex-Anschluss mit automatischem Erkennen (Autosensing) und automatischem Aushandeln (Autonegotiation) der Netzwerkeinstellungen. Diese Funktionen laufen nach dem Einschalten des CP wie folgt ab:

- Der CP versucht, die vom Partner verwendete Übertragungsgeschwindigkeit zu erkennen.
- Ist die Erkennung möglich, versucht der CP mit dem Partner eine optimale Duplex-Betriebsart auszuhandeln.
- Ist kein Aushandeln möglich, verwendet der CP die zuvor erkannte Übertragungsgeschwindigkeit und Half Duplex.

Die Dauer der beschriebenen Schritte beträgt ca. 2 Sekunden.

Zusätzlich beinhaltet die "Automatische Einstellung" einen Autocrossing-Mechanismus; unabhängig von den Eigenschaften des Partnergerätes können Sie dadurch ein gekreuztes oder kein gekreuztes Kabel zum Anschluss verwenden.

Auf individuelle Netzwerkeinstellungen umschalten

Sobald Sie eine manuelle Konfiguration projektieren, ist die automatische Umschaltung nicht mehr wirksam. Dies gilt auch für den im CP integrierten Autocrossing-Mechanismus. Wenn das angeschlossene Partnergerät keinen Autocrossing-Mechanismus unterstützt, müssen Sie je nach Partnergerät ein gekreuztes (bei einem Switch) oder kein gekreuztes Kabel (bei einem Endgerät) verwenden.

Auswahlmöglichkeit Port "deaktivieren":

Abhängig vom Baugruppentyp wird in der Klappliste die Option "– deaktivieren -" angeboten. Sie haben damit beispielsweise die Möglichkeit, den Zugang zu einem unbenutzten Port aus Sicherheitsgründen zu unterbinden.

Siehe auch

Weitere Informationen

Beachten Sie bitte die weiteren Angaben im Gerätehandbuch des jeweiligen CPs.

Maximale Telegrammgröße / Datenmenge für PROFINET IO-Devices

Größe der Nutzdaten beim CP 1616

Die Größe der Nutzdaten (Prozessdaten) berechnet sich aus der maximalen Telegrammgröße (Bruttodatenmenge) abzüglich der Werte für die in diesem IO-Device gesteckten Module. Daraus ergibt sich die Nettodatenmenge.

Tabelle 1-155 Beispiel für die Berechnung der Nutzdaten (Byte) des CP 1616 als IO-Device mit homogenen Modulen (z. B. nur DI-Module)

Bytes	Erläuterung
1434	Bruttogröße des Telegramms
-16	für die maximale Anzahl der steckbaren Module
= 1418	Nettogröße der Nutzdaten

Tabelle 1-156 Beispiel für die Berechnung der Nutzdaten (Byte) des CP 1616 als IO-Device mit heterogenen Modulen (z. B. DI/DO)

Bytes	Erläuterung
1434	Bruttogröße des Telegramms
-32	für die doppelte Anzahl der maximal steckbaren Module
= 1370	Nettogröße der Nutzdaten

Größe der Nutzdaten beim CP 343-1 (Lean/Standard/Advanced)

Die Größe der Nutzdaten (Prozessdaten) berechnet sich aus der maximalen Telegrammgröße (Bruttodatenmenge) abzüglich der Werte für den CP und die Schnittstelle selbst, für die Anzahl der Ports und die in diesem IO-Device gesteckten Module. Daraus ergibt sich die Nettodatenmenge.

Tabelle 1-157 Beispiel für die Berechnung der Nutzdaten (Byte) des CP 343-1 als IO-Device

Bytes	Erläuterung
548	Bruttogröße des Telegramms
-1	für den CP selbst
-1	für die Schnittstelle
-2	für 2 Ports
-32	für die maximale Anzahl der steckbaren Module
= 512	Nettogröße der Nutzdaten *

* Bei Verwendung von weniger aber größeren Modulen kann sich die Nutzdatenmenge erhöhen.

DNS-Konfiguration

Bezug

Projektierung der DNS-Server-Adresse in der Parametergruppe "DNS-Konfiguration"

Ein DNS-Server kann erforderlich sein, wenn die Baugruppe selbst, ein Kommunikationspartner oder bspw. ein E-Mail-Server über den Host-Namen (FQDN) erreichbar sein soll.

DNS-Server für E-Mail-Verbindungen

Bei der Projektierung einer E-Mail Verbindung ist die Adresse des Mail-Servers anzugeben, über den die E-Mails geschickt werden sollen. Die Adresse des Mail-Servers kann als IP-Adresse oder als FQDN angegeben werden.

Bei der Angabe der Server-Adresse als FQDN müssen Sie einen DNS-Server projektieren. In diesem Fall wird die IP-Adresse des Mail-Servers über den projektierten DNS-Server ermittelt.

IP-Zugriffsschutz

IP-Zugriffsschutz

Bezug

Parametrierung der Schnittstellen-Eigenschaften in der Parametergruppe "Eigenschaften > Allgemein > IP-Zugriffsschutz".

Übersicht

Für den IP-Zugriffschutz sind abhängig vom CP-Typ folgende Optionen parametrierbar:

- Option "Webserver aktivieren"
  Der CP stellt Ihnen für den Zugriff mittels Webbrowser die Funktion eines Webservers zur Verfügung. Hierzu werden bestimmte HTML-Seiten mit CP-Informationen und Diagnosefunktionen in einen Speicherbereich des CP abgelegt.
  Aktivieren Sie die Option, um Zugriff auf diese HTML-Seiten zu erhalten. Hierdurch wird Port 80 des CP freigegeben.
  Standardmäßig ist der Webserver Zugriff aktiviert.
- Option "FTP-Server aktivieren"
  Wählen sie diese Option, wenn sie FTP-Zugriffe auf die S7-Station über Port 20/21 des CP zulassen möchten.
  Wenn Sie im Register "FTP" den FTP-Zugriff auf File-DBs in der CPU projektieren, ist der Zugriff nur möglich, wenn Sie hier die Option aktivieren.
  Standardmäßig ist der FTP-Serverzugriff über Port 20/21 aktiviert.
- Option "Zugriffschutz für IP-Kommunikation aktivieren"
  Die Ethernet-Schnittstelle bietet Ihnen die Möglichkeit, den Zugriff auf das lokale Gerät auf Partner mit ganz bestimmten IP-Adressen einzuschränken. Von Ihnen nicht autorisierte Partner haben damit über diese Schnittstelle keinen Zugang zu Daten des lokalen Gerätes. Hierzu können Sie den IP-Zugriffsschutz aktivieren oder deaktivieren sowie bestimmte IP-Adressen in einer IP Access Control-Liste (IP-ACL) eingeben.
  Standardmäßig ist der IP-Zugriffsschutz deaktiviert.
  Geblockte Zugriffsversuche werden im CP registriert und können über die Spezialdiagnose im Diagnoseobjekt "IP-Zugriffsschutz" eingesehen werden. Bei CPs mit IT-Funktion wird im Dateisystem des CPs zusätzlich eine LOG-Datei angelegt, die Sie über WEB-Browser einsehen können.
  Beachten Sie bitte die Anleitung zur Handhabung der LOG-Datei und den Hinweis zum Laden der Projektierdaten weiter unten.

IP-Zugriffsschutz für projektierte Kommunikations-Partner

Wenn Sie den Zugriff genau auf die Kommunikations-Partner einschränken möchten, die Sie bei der Projektierung angegeben haben, genügt es, den Zugriffsschutz zu aktivieren. Sie müssen in diesem Fall keine IP-Adressen in die Liste eintragen.

Berücksichtigte Kommunikationspartner sind:

- Stationen, zu denen Kommunikationsverbindungen projektiert sind;
  Dies gilt (außer bei S7-Verbindungen) auch für Verbindungen, bei denen der Verbindungspartner in einem anderen Subnetz liegt.
  Beachten Sie aber, dass auf unspezifizierten Verbindungen, deren Partner-Adressen unbestimmt bleiben, alle nicht eingetragenen IP-Adressen dann nicht autorisiert sind und abgewiesen werden. Auch Verbindungen unter PROFINET CBA werden als unspezifizierte Verbindungen behandelt. Für diese Verbindungen müssen Sie explizit IP-Adressen in die IP Access Control-Liste eintragen.
- PROFINET IO-Devices, wenn der Ethernet-CP als PROFINET IO-Controller verwendet wird;
  Die IP-Adressen von PROFINET IO-Devices werden im RUN-Zustand des CP dynamisch in die IP Access Control-Liste eingetragen.
  Geht der CP in den STOP-Zustand über, werden die IP-Adressen der PROFINET IO-Devices wieder aus der Accessliste gelöscht.
- NTP-Server, SMTP-Server, DNS-Server und DHCP-Server.
  Die IP-Adressen von NTP-Servern, SMTP-Servern, DNS-Servern und DHCP-Servern werden ebenfalls bei entsprechenden Anfragen an die Server dynamisch ein- und ausgetragen.

Der IP-Zugriffsschutz bezieht sich auf alle Verbindungstypen, die über IP-Protokoll abgewickelt werden (TCP, ISO-on-TCP, UDP, S7)

Anmerkung zu dynamisch vergebenen IP-Adressen:
Da jeder Dienst seinen dynamischen Eintrag in die ACL selbst verwaltet, kann die gleiche IP-Adresse durchaus mehrmals in der Baugruppendiagnose erscheinen.

IP-Zugriffsschutz für Partner mit ganz bestimmten IP-Adressen

Um den Zugriff auf ganz bestimmte IP-Adressen einzuschränken, geben Sie diese IP-Adressen in der IP Access Control-Liste an. Dies können beispielsweise die IP-Adressen von in der Verbindungsprojektierung unspezifizierten Verbindungspartnern, von einzelnen Programmiergeräten oder von Verbindungspartnern bei PROFINET CBA sein.

Die IP-Adressen, die Sie in der Verbindungsprojektierung angegeben haben, gehören immer zu den zugelassenen IP-Adressen und müssen daher nicht explizit in der IP-ACL eingetragen werden.

LOG-Datei mittels Web-Browser einsehen

Bei CPs mit IT-Funktion wird im Dateisystem des CP eine LOG-Datei angelegt, die Sie über Web-Browser einsehen können. Gegenüber der Aufzeichnung in der Spezialdiagnose bietet die LOG-Datei Speicherplatz für bis zu 512 Einträge.

Die LOG-Datei ist erst ab dem Zeitpunkt vorhanden, ab dem der IP-Zugriffsschutz erstmalig aktiviert wurde.

Sie finden die LOG-Datei als HTML-Datei im Dateisystem des CPs unter folgendem Verzeichnis:

·ram/security/IPLogFile.htm

Weitere Eigenschaften:

Die LOG-Datei wird als Umlaufpuffer angelegt; wenn mehr als 512 Einträge erfasst sind, werden umlaufend die jeweils ältesten Einträge überschrieben.

Die Einträge erfolgen ausschließlich chronologisch; es gibt keine weiteren Kriterien für eine Sortierung.

Beachten Sie, dass diese Funktion nur bei neueren Firmware-Versionen der CPs unterstützt wird; weitere Informationen hierzu finden Sie in der CP-Gerätedokumentation.

Bei Advanced-CPs ab CP 343-1 Advanced (GX30) und CP 443-1 Advanced (GX20) wird die LOG-Datei nicht angelegt. Dort können Sie geblockte Zugriffsversuche direkt über die Web-Diagnose einsehen.

Eingabe der Zugriffsberechtigungen in der IP Access Control-Liste (IP-ACL)

Bezug

Parametrierung der Schnittstellen-Eigenschaften in der Parametergruppe "Eigenschaften > Allgemein > IP-Zugriffsschutz".

Regeln zur Eingabe der IP-Adressen:

Beachten Sie bei der Eingabe der IP-Adressen die nachfolgenden Regeln:

- In jeder Zeile können Sie die IP-Adresse einzeln oder als Bereich eingeben.
- Zusätzlich können Kommentare eingegeben werden.
- Die maximal eingebbare Anzahl von Adressen beträgt: 32
- Die maximal eingebbare Anzahl von Adressbereichen beträgt: 10
- Es wird nicht geprüft, ob Adressen, die als Einzeladresse bereits eingegeben wurden, in einem Adressbereich ebenfalls enthalten sind.
- Es wird nicht geprüft, ob in einem Bereich ungültige Adressen enthalten sind (beispielsweise können Broadcast-Adressen hier angegeben werden, obwohl sie nicht als IP-Adresse eines Absenders auftreten können).
- Kommentare beginnen mit dem Zeichen "#"

Beispiele:

- einzelne IP-Adresse mit Kommentar: 141.80.0.16 #Adresse von Partner x
- Bereich von IP-Adressen mit Kommentar: 141.80.0.16 - 141.80.0.25 #Adressbereich für Leitebene

Rechte

Die hier beschriebenen Zugriffsberechtigungen sind in der IP Access Control-Liste mit den jeweiligen Kurzbezeichnungen wählbar.

Tabelle 1-158 Zugriffsrechte und Kurzbezeichnungen

Zugriffsrecht	Kurzbezeichnung	Bedeutung / Anmerkung
Zugriff auf Station	A	Access Kommunikationspartner mit Adressen im angegebenen Bereich haben Zugriff auf die dem CP zugehörende Station (CP / CPU). Diese Zugriffsberechtigung ist für IP-Adressen, die Sie in der Verbindungsprojektierung angegeben haben, implizit gesetzt.
Ändern der IP Access Control-Liste	M	Modify Kommunikationspartner mit Adressen im angegebenen Bereich haben ein Zugriffsrecht auf die IP Access Control-Liste. Diese Kommunikationspartner können Einträge für die IP Access Control-Liste per HTTP an den CP senden. Wenn Sie diese Berechtigung wählen, wird die Berechtigung "A" automatisch zusätzlich gesetzt. Hinweise zu den per HTTP gesendeten Einträgen: • Mit jeder per HTTP übertragenen Liste wird eine zuvor per HTTP übertragene Liste ungültig. • Mit den per HTTP übertragenen Zugriffsberechtigungen können über STEP 7 projektierte Einträge ergänzt, nicht jedoch gelöscht werden. • Eine per HTTP übertragene Liste wird mit einem Ausfall der Versorgungsspannung am CP (Spannung AUS) gelöscht. Informationen zur Syntax einer per HTTP übertragbaren Liste entnehmen sie bitte der Dokumentation zum CP.
IP-Routing-Zugriff auf anderes IP-Subnetz	R	Routing Kommunikationspartner mit Adressen im angegebenen Bereich haben Zugriff auf weitere am CP angeschlossene Subnetze. Diese Zugriffsberechtigung ist für IP-Adressen, die Sie in der Verbindungsprojektierung angegeben haben, nicht automatisch gesetzt. Bei Bedarf muss dieses Zugriffsrecht hier explizit gesetzt werden.

Hinweis

- **Projektierdaten über TCP/IP laden**
  Beachten Sie, dass Sie beim Laden über TCP/IP und aktiviertem Zugriffsschutz die IP-Adresse des PG/PC in die IP-ACL aufnehmen müssen, von dem aus die Konfiguration geladen wird! Das PG/PC wird sonst während des Ladevorganges als nicht-autorisiert erkannt und der Ladevorgang wird abgebrochen.
- **S7-Verbindungen nachladen**
  Wenn Sie einzelne S7-Verbindungen in die S7-Station nachladen, werden diese nicht automatisch in der IP-ACL berücksichtigt. Sie müssen in diesem Fall die gesamte Stationsprojektierung erneut laden.
- **Projektierung mittels Hochladen in das PG/PC übernehmen und bearbeiten**
  Wenn Sie beabsichtigen, die Verbindungsprojektierung zu bearbeiten und den IP-Zugriffschutz aktiviert haben, werden auf dem PG/PC die vollständigen Verbindungsinformationen benötigt. Deshalb müssen Sie ein Hochladen der aktuellen Verbindungsdaten durchführen. Das anschließende Herunterladen in die S7-Station erfordert dann ein vollständiges Laden der Verbindungsdaten und der Stationsdaten.

Hinweis

Mittels der Spezialdiagnose können Sie die in der S7-Station aktuelle IP-ACL einsehen und ggf. mit der projektierten IP-ACL vergleichen.

Zugriffsrechte in der Benutzerverwaltung bearbeiten

Bezug

Parametrierung der Benutzerrechte in der Parametergruppe "Eigenschaften > Allgemein > Benutzerverwaltung".

Bedeutung der Benutzerverwaltung

Legen Sie in der Benutzerverwaltung fest, welchen Benutzern welche Rechte beim Zugriff auf Ihre S7-Station über einen Web-Browser bzw. FTP-Client zugewiesen werden sollen.

In einer alphabetisch geordneten Liste finden Sie unter Benutzername die bisher eingegebenen Benutzer, zu denen Passwörter hinterlegt sind.

Der Eintrag "Jeder" ist standardmäßig vorhanden. Er kann nicht verändert werden. Ihm kann auch kein Passwort zugewiesen werden.

Benutzereinträge sind immer mit einem Passwort verknüpft. Ausnahme: der Eintrag "Jeder" ist nicht durch ein Passwort geschützt.

Hinweis

Unter dem Eintrag "Jeder" sind standardmäßig keinerlei Rechte zugewiesen. Für Servicezwecke ist jedoch eine Rechtevergabe möglich.

Es ist aber zu beachten, dass die dem Benutzer "Jeder" zugeteilten Rechte jedem anderen Benutzer zur Verfügung stehen. Daher ist es erforderlich, die Rechte nach dem Servicefall gegebenenfalls wieder zurückzunehmen! Sie gewähren sonst jedem Zugriff das Recht ein, ohne Autorisierung entsprechende Dienste auszuführen.

Falls unter dem Benutzer "Jeder" ein Zugriffsrecht gesetzt ist, ist dies bei der Rechtevergabe für einen anderen Benutzer an dem gefüllten Optionskästchen erkennbar.

Neuen Benutzer eintragen

1. Selektieren Sie die erste freie Zeile in der Anwenderliste.
   Es wird ein neuer Anwendereintrag mit der Zugriffstufe "minimal" angelegt.
2. Passen Sie in der Spalte "Name" den vorgeschlagenen Anwendernamen an.
3. Doppelklicken Sie innerhalb der neu angelegten Zeile in die Spalte "Zugriffstufe".
   Das Symbol für die Klappliste wird sichtbar.
4. Klicken Sie auf das Symbol der Klappliste.
5. Legen Sie im anschließend aufgeblendeten Dialog spezifische Benutzerrechte fest.
6. Bestätigen Sie die Eingabe mit "OK".
7. Vergeben Sie in aufgeblendeten Passwort-Dialog ein neues Passwort.

Benutzerrechte ändern oder ein Passwort ändern:

1. Doppelklicken Sie in der Zeile des gewünschten Anwendereintrages in die Spalte "Zugriffstufe".
   Das Symbol für die Klappliste wird sichtbar.
2. Klicken Sie auf das Symbol der Klappliste.
3. Ändern Sie im anschließend aufgeblendeten Dialog die spezifischen Benutzerrechte oder klicken Sie auf die Schaltfläche "Passwort" um ein neues Passwort zu vergeben.
4. Bestätigen Sie die Eingabe mit "OK".
5. Geben Sie im aufgeblendeten Passwort-Dialog das Passwort ein.

Benutzereinträge löschen

1. Wählen Sie über das Kontextmenü in der Zeile des gewünschten Anwendereintrages den Menübefehl "Löschen".

Variablendeklaration - Variablenliste bearbeiten

Bezug

Parametrierung der in der CPU konfigurierten Variablen in der Parametergruppe "Eigenschaften > Allgemein > Variablendeklaration".

Bedeutung der Variablendeklaration

Um über einen Web-Browser per Java Applet auf Variablen der CPUs zugreifen zu können, müssen dem CP die Namen, Adressen und Zugriffsrechte dieser Variablen bekannt gemacht werden.

Sie sehen in der Symbolliste in der Variablendeklaration die aktuell bekannten Variablen mit ihren symbolischen Namen, den zugehörenden Operanden und den zugewiesenen Zugriffsrechten.

Neuen Variableneintrag hinzufügen

1. Doppelklicken Sie in die erste freie Tabellenzeile.
   Das Symbol für die Auswahl über Klappliste wir angezeigt.
2. Selektieren Sie das Symbol und öffnen Sie damit die Strukturanzeige für die Variablen im aktuellen Gerät.
3. Wählen Sie die Symbole der Variablen oder der Strukturelemente, auf die ein Zugriff per Web-Browser möglich sein soll.
   Anmerkung: Die Symbole müssen zuvor im Symbol Editor hinterlegt worden sein!

Variableneintrag Löschen

1. Wählen Sie über das Kontextmenü in der Zeile der zu löschenden Variable den Menübefehl "Löschen".
   Achtung:

   Hinweis

   Es erfolgt keine Rückfrage; der Eintrag wird sofort gelöscht. Eine Mehrfachauswahl ist nicht möglich.

Zugriffsrechte einstellen - Option "schreibbar"

Standardmäßig werden für die Variablen nur Leserechte.

Um die Schreib-/Leserechte für die hier eingetragenen Variablen zu vergeben, gehen Sie so vor:

1. Wählen Sie in der Symbolliste ein oder mehrere Variablen an.
2. Stellen Sie die Option "nur Lesen erlaubt" entsprechend für die ausgewählten Variablen ein.

Variablendeklaration - Variablenliste bearbeiten (ERPC)

Bezug

Projektierung der in der CPU angelegten Variablen für die ERPC-Applikation

Sie wählen die Variablen aus, um diese für die ERPC-Applikation zugänglich zu machen.

Tabelle der ERPC-Variablen

In der Tabelle werden die im STEP 7-Projekt projektierten PLC Variablen angezeigt, die für die ERPC-Applikation zur Verfügung stehen. Die eventuell zu einem früheren Zeitpunkt in die Station geladenen ERPC-Variablen werden nicht angezeigt.

Zu den für ERPC projektierten Variablen werden in der Tabelle folgende Informationen angezeigt:

- Info
  Zusatzinformation aus der Änderungsprüfung
  ERPC-Variablen, die Sie im STEP 7-Projekt nach dem letzten Speichern geändert haben, erhalten durch die Änderungsprüfung in der Spalte "Info" eine entsprechende Zusatzinformation (siehe unten).
- Variable
  Name der Variablen
- Schreibbar
  Die für die ERPC-Kommunikation projektierten Rechte (lesen aus der CPU, schreiben in die CPU); standardmäßig sind nur Leserechte vergeben.
- Kommentar
  Für die ERPC-Kommunikation spezifische Kommentare der ERPC-Variablen.

Unterhalb der Tabelle:

- Schreibbar
  Durch Aktivierung der Option "Schreibbar" heben Sie den Schreibschutz der selektierten ERPC-Variablen auf und geben sie für Schreibvorgänge der ERPC-Applikation frei.
- Anzahl Variablen:
  Die Gesamtzahl der für ERPC verwendeten Variablen.

Mehrfach-Selektion von Variablen

Über die Tasten <Shift> und <Strg> ist eine Mehrfachauswahl von Variablen in der Tabelle möglich.

Änderungsprüfung

Beim Öffnen der Variablendeklaration findet eine Änderungsprüfung statt. Die für die ERPC-Applikation ausgewählten Variablen werden mit denen der Variablentabelle der zugeordneten CPU im STEP 7-Projekt verglichen (Offline-Variablentabelle).

Hinweis

Ein Vergleich mit ERPC-Variablen, die eventuell zu einem früheren Zeitpunkt aus dem STEP 7-Projekt in die Station geladen wurden, findet nicht statt.

In der Spalte "Info" werden die relevanten Informationen zu Differenzen zwischen den ERPC-Variablen und denen der Variablentabelle der CPU angezeigt.

Anzeige in Spalte "Info"	Zustand / Information
	Variable unverändert
ACC	Die Variable kann derzeit nicht bearbeitet werden. Begründung: Der Datenbaustein, in dem die Variable verwendet wird, ist gerade geöffnet.
DEL	In der Variablentabelle der CPU gelöschte Variable.
ADR	In der Variablentabelle der CPU geänderte Adresse
TYP	In der Variablentabelle der CPU geänderter Datentyp Anmerkung: Wenn für eine Variable in der Variablentabelle die Adresse und der Datentyp geändert werden, dann wird die Info "ADR" von der Info "TYP" überschrieben.
DEF	Nicht unterstützter Variablentyp
UPL	Von der Station hochgeladene Variable. (nach Ausführen der STEP 7-Funktion "Laden in PG")

Wenn Einträge vom Typ "DEL", "ADR" und "TYP" vorhanden sind, müssen Sie die entsprechenden Korrekturen vornehmen, bevor Sie das STEP 7-Projekt speichern.

FTP-Konfiguration für FTP-Server-Betrieb

Ethernet-CP als FTP-Server für die CPU-Daten

Bezug

Datenerfassung zum automatischen Anlegen einer File-Zuordnungstabelle in der Parametergruppe "Eigenschaften > Allgemein > FTP-Konfiguration".

Über die Angaben in der File-Zuordnungstabelle ist es möglich, Datenbausteine in einer oder mehreren CPUs (bis zu 4) in einer S7-Station anzusprechen.

Option "FTP-Server aktivieren" (nur bei bestimmten CP-Typen)

Wählen Sie diese Option, um FTP-Zugriffe auf die S7 Station über Port 20/21 des CP zuzulassen.

Das Aktivieren der Option ist Voraussetzung, um den FTP-Zugriff auf File-DBs in der CPU mit der nachfolgend beschriebenen Option zu aktivieren.

Standardmäßig ist der FTP-Serverzugriff über Port 20/21 aktiviert.

Option "FTP-Server für S7 CPU-Daten verwenden"

Indem Sie diese Option wählen, wird die hier projektierte File-Zuordnungstabelle beim Laden der Projektierdaten in den CP erzeugt und im Verzeichnis /config des Dateisystems des CP hinterlegt.

Eine bereits bestehende Datei file_db.txt wird überschrieben.

Hinweis

Bei aktivierter Security und aktivierter Option "Zugriff nur über FTPS zulassen" gilt:

- Die Dateien werden verschlüsselt übertragen.
- Für den Benutzer müssen die Rechte "FTP: Dateien (DBs) von der S7-CPU lesen" bzw. "FTP: Dateien (DBs) von der S7-CPU schrieben" aktiviert sein.
- Bei aktivierter Firewall müssen die Protokolle FTP/FTPS freigegeben sein.

Ethernet-CP als FTP-Server für die S7 CPU-Daten

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine an; wegen ihrer speziellen Struktur werden diese hier als File-DBs bezeichnet.

Der Ethernet-CP als FTP-Server ermittelt bei einem FTP-Kommando aus einer File-Zuordnungstabelle (Datei file_db.txt), wie die in der S7-Station für den Filetransfer genutzten Datenbausteine auf Dateien (Files) abgebildet werden sollen.

Sie können die File-Zuordnungstabelle wie folgt erstellen und in den CP übertragen:

- Durch Eingabe in der hier beschriebenen Parametergruppe "Eigenschaften > FTP-Konfiguration";
  Die File-Zuordnungstabelle wird dann aus den Angaben in der FTP-Konfiguration erzeugt und zusammen mit den Projektierdaten automatisch in den CP geladen.
- Durch direktes Anlegen einer Datei file_db.txt.
  Sie müssen die so erstellte File-Zuordnungstabelle mittels FTP-Kommando in den CP laden.

Die File-Zuordnungstabelle file_db.txt wird im Dateisystem des Ethernet-CP im Verzeichnis /config hinterlegt.

Siehe auch

Aufbau und Struktur der FTP-Konfigurationsliste (Seite 3237)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb (Seite 3238)

Aufbau und Struktur der FTP-Konfigurationsliste

Aufbau und Struktur der FTP-Konfigurationsliste zum Erzeugen der File-Zuordnungstabelle

Die Felder der Tabelle in der FTP-Konfiguration haben folgende Bedeutung und Syntax:

Betitelung in der Tabelle zur FTP-Konfiguration	"CPU"	"DB"	Dateiname	Kommentar
Bedeutung	Rack-/Slot-Zuordnung der CPU Auswählbar über Klappliste; Eintrag nicht eingebbar.	Nr. des Datenbausteins Eintrag eingebbar	Dem File-DB zugewiesener File Name Automatischer Namensvorschlag; Eintrag eingebbar / änderbar.	Informeller Kommentar
Beispiel 1	cpu1 [PLC_1]	20	cpu1_db20.dat	Messwerte im Bereich Anlage1
Beispiel 2	cpu2 [PLC_2]	35	Cpu2_db35.dat	Messwerte im Bereich Anlage2

Hinweise zur Syntax

- Gültige Trennzeichen für die Einträge sind "Leerzeichen".
- Für den Dateinamen (File Name) eines File-DB gilt:
  - Der Dateiname beginnt mit "cpuX" (mit X=1,2,3 oder 4);
  - "cpuX" muss zuvor in der Rack-/Slot-Zuordnung der CPU definiert sein;
  - Länge: maximal 64 Zeichen (einschließlich der Angabe "cpuX");
- Maximal 100 Einträge;
- Zulässige Zeichen: Buchstaben "A-Z,a-z"; Ziffern "0-9", "_", "."

Hinweis

Beachten Sie die Schreibweise (Kleinbuchstaben für "cpu" und keine führenden Leerzeichen am Zeilenbeginn). Die Dateien werden sonst nicht erkannt.

Der Tabulator ist als Trennzeichen **nicht** zugelassen.

Siehe auch

Ethernet-CP als FTP-Server für die CPU-Daten (Seite 3235)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten

- Abschnitt 1: File-DB-Header (besitzt feste Länge (20 Byte) und Struktur)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)Datentyp "FILE_DB_HEADER

Datentyp "FILE_DB_HEADER

Unter der Funktion "Neuen Baustein hinzufügen" weisen Sie einem Datenbaustein den Typ "FILE_DB_HEADER" direkt zu.

File-DB-Header für FTP-Server-Betrieb

Anmerkung: Der hier beschriebene File-DB-Header ist weitgehend identisch zu dem für den Client-Betrieb beschriebenen File-DB-Header. Die Unterschiede beziehen sich auf die Parameter

- WRITE_ACCESS
- FTP_REPLY_CODE

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten ("Datei existiert nicht"). • 1: Der File-DB enthält gültige Nutzdaten ("Datei existiert").	Das FTP-Kommando dele setzt EXIST=0; Das FTP-Kommando store setzt EXIST=1;
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos store und retr setzen während der Bearbeitung LOCKED=1. Für einen Schreibvorgang aus dem Anwenderprogramm gilt: Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes LOCKED setzen bzw. zurücksetzen. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen; wenn =0 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen; wenn =0 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen

Parameter	Typ	Wert / Bedeutung	Versorgung
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muss nach dem Lesen der Daten NEW=0 setzen, um ein erneutes store zu ermöglichen oder um das File mittels FTP-Kommando dele löschen zu können.
WRITE_ACCESS	BOOL	0: Der FTP-Client auf dem PG/PC hat kein Schreibrecht für die File-DBs in der S7-CPU. 1: Der FTP-Client auf dem PG/PC hat Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungswert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereiches (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Der Parameter ist im FTP-Serverbetrieb ohne Bedeutung.	Wird vom FTP-Server auf "0" gesetzt.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung des Files. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wird die Funktion "Uhrzeitweiterleitung" genutzt, entspricht der Eintrag der weitergeleiteten Zeit. Wird die Funktion "Uhrzeitweiterleitung" nicht genutzt, wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des IT-CP (Initialisierungswert ist 1.1.1994 0.0 Uhr).

Siehe auch

Ethernet-CP als FTP-Server für die CPU-Daten (Seite 3235)

E/A-Adressen

Bezug

Parametrierung für die Ein-/Ausgangsadressen in der Parametergruppe "Eigenschaften > Allgemein > E/A-Adressen"

Bedeutung

- Eingänge, Ausgänge
  Weisen Sie der Baugruppe eine Anfangsadresse zu. (Ausgänge nur, wenn die Option "Adresseinstellung für LOCK/UNLOCK bei FETCH/WRITE" gewählt wird)
- Option "Adresseinstellung für LOCK/UNLOCK bei FETCH/WRITE"
  Wählen Sie diese Option, wenn Sie die Funktion Zugriffskoordinierung mit den FCs LOCK/UNLOCK bei FETCH/WRITE-Betrieb nutzen möchten.
  Diese Funktion verwendet die Prozessausgabe über den Rückwand-/P-Bus. Daher werden auch die Ausgangsadressen einstellbar, sobald Sie diese Option gewählt haben.

Web

Webserver des Moduls

Option "Webserver aktivieren"

Der CP stellt Ihnen für den Zugriff mittels Webbrowser die Funktion eines Webservers zur Verfügung. Hierzu werden HTML-Seiten mit Informationen und Diagnosefunktionen im Speicherbereich des CP abgelegt.

Aktivieren Sie die Option, um Zugriff auf diese HTML-Seiten im Webserver der CPU zu erhalten. Mit der Aktivierung der Funktion wird Port 80 des CP freigegeben.

In der Voreinstellung ist der Webserver-Zugriff aktiviert.

Optionen der Webdiagnose (Verfügbarkeit abhängig vom CP-Typ)

- Option "Topologieanzeige"
  Die Option ermöglicht die Anzeige von Topologieinformationen für die vernetzten PROFINET IO-Schnittstellen des CP im Webserver.
  Zur Ermittlung und Ablage der Topologieinformation wird zusätzlicher Speicherplatz auf dem CP belegt. Bei komplexen Projekt-Konfigurationen mit hohen Anforderungen an die Speicher-Ressourcen kann es daher sinnvoll sein, die Option nur vorübergehend für Servicezwecke zu aktivieren.
- Option "Firmware-Download über Web"
  Indem Sie die Option aktivieren, ist die Funktion zum Laden der Firmware des CP über das Download Center im Webserver freigeschaltet.
  Diese Option unterliegt bei deaktivierter Security nicht der Benutzerverwaltung. Es wird daher empfohlen, die Option nur bei Bedarf freizuschalten.
- Option "Nachladen der Sprachdateien für Diagnoseanzeigen über Web"
  Diagnoseanzeigen der CPs werden im Web-Diagnosepuffer mit Klartext angezeigt. Diese Anzeigen erfolgen sprachspezifisch.
  Indem Sie die Option aktivieren, ist die Funktion zum Nachladen fehlender Sprachdateien über das Download Center im Webserver freigeschaltet.

Automatische Aktualisierung / "Aktualisierungsintervall"

Indem Sie die Option "Aktivieren" wählen, aktualisiert der CP die angezeigten Webseiten in der eingestellten Intervallzeit.

Wertebereich für das Aktualisierungsintervall 1..999 s

Webdiagnose im Modul

Voraussetzung:

Stellen Sie eine physikalische Verbindung zwischen dem PG und der SIMATIC S7-Station her und stellen Sie die PG/PC-Schnittstelle so ein, dass das Modul erreichbar ist. Weitere Hilfe erhalten Sie unter der Funktion "PG/PC-Schnittstelle einstellen...".

"Start der Webdiagnose"

Klicken Sie auf die Schaltfläche "Webdiagnose", um das Ergebnis der Baugruppendiagnose im Webbrowser anzuzeigen. Die Inhalte werden vom integrierten HTTP-Server des Moduls geliefert.

Starten Sie die Diagnose im Webbrowser in der Parametergruppe "Eigenschaften > Allgemein > Webdiagnose".

- Option "Zugang über"
  Wählen Sie in der Liste "Zugang über" die Schnittstelle aus, über die der CP angesprochen wird.
  - Verfügbare Schnittstellen sind in der Klappliste wählbar.
  - Die direkte Eingabe einer IP-Adresse ist möglich.

Hinweis

Bei aktivierten Security-Funktionen (falls verfügbar)

Wenn die Option "Zugriff nur über HTTPS zulassen" aktiviert ist, beachten Sie folgendes:

- Die Daten werden verschlüsselt übertragen.
- Für den Benutzer muss das Recht "Auf Web-Diagnose zugreifen" aktiviert sein.
- Bei aktivierter Firewall müssen die Protokolle HTTP/HTTPS freigegeben sein.

SNMP

Option "SNMP aktivieren"

Bei aktivierter Option "SNMP aktivieren" wird die Kommunikation über SNMP freigegeben und damit lesender und schreibender Zugriff auf die Baugruppe ermöglicht.

Nur bei Baugruppen, welche Security-Funktionen unterstützen, können Sie die Kommunikation nach der Spezifikation SNMPv1 oder SNMPv3 auswählen:

- SNMPv1
  In der Baugruppe wird der SNMPv1-Agent freigegeben. Bei Baugruppen, welche Security-Funktionen unterstützen, können Sie für SNMPv1 die Community Strings für lesenden und schreibenden Zugriff ändern.
- SNMPv3
  In der Baugruppe wird der SNMPv3-Agent freigegeben. Weitere Konfigurationsmöglichkeiten zur Verschlüsselung, Authentifizierung und Vergabe von Rechten für STEP 7-Benutzer stehen Ihnen zur Verfügung.

CP - OPC UA

Security-Profile und Zertifikatsvalidierung

Security-Profile des OPC UA-Servers

Folgende Security-Profile werden von der Server-Applikation des CP unterstützt:

- **Unterstützte Security-Profile**
  - Kein Security-Profil
    Der CP verwendet kein Security-Profil.
  - Basic128Rsa15
    Dies entspricht dem Security-Profil "Basic128Rsa15" der OPC UA-Spezifikation.
    Der CP verwendet Signierung und, falls projektiert, 128-Bit-Verschlüsselung.
  - Basic256
    Dies entspricht dem Security-Profil "Basic256" der OPC UA-Spezifikation.
    Der CP verwendet Signierung und, falls projektiert, 256-Bit-Verschlüsselung.
  - Basic256Sha256
    Dies entspricht dem Security-Profil "Basic256Sha256" der OPC UA-Spezifikation.
    Der CP verwendet Signierung und, falls projektiert, 256-Bit-Verschlüsselung unter Nutzung des Hash-Algorithmus SHA-256.

Wenn Sie mehrere Optionen aktivieren, dann wählt der CP in Abhängigkeit der Einstellungen beim Kommunikationspartner (Client) das Profil mit der höchstmöglichen Sicherheit aus.

- **Signieren und verschlüsseln**
  Die oben aufgeführten Optionen haben folgende Auswirkungen:
  - Signieren
    Der CP erlaubt nur Kommunikation mit signierten Telegrammen.
  - Signieren und verschlüsseln
    Der CP erlaubt nur Kommunikation mit signierten und verschlüsselten Telegrammen.

Zertifikatsvalidierung

In dieser Parametergruppe stellen Sie für die Server- bzw. für die Client-Applikation die Optionen für die Prüfung der Zertifikate der Kommunikationspartner ein.

- **Zertifikate prüfen**
  Der CP prüft generell das Zertifikat des Kommunikationspartners.
  Falls das Partnerzertifikat ungültig oder nicht vertrauenswürdig ist, wird die Kommunikation abgebrochen.
  Bei aktivierter Option können die beiden folgenden Optionen zur Einschränkung der Prüfungsroutinen aktiviert werden.
- **Keine strikte Zertifikatsvalidierung**
  Bei aktivierter Option lässt der CP Kommunikation auch in den folgenden Fällen zu:
  - Die IP-Adresse des Kommunikationspartners ist nicht identisch mit der IP-Adresse in dessen Zertifikat.
  - Die im Zertifikat hinterlegte Verwendung (OPC UA-Client/Server) unterscheidet sich von der Funktion (OPC UA-Client/Server) des Kommunikationspartners.
  - Der aktuelle Zeitpunkt im CP liegt außerhalb der Gültigkeitsdauer des Partnerzertifikats.

  Unabhängig von diesen Ausnahmen müssen für den Aufbau einer Verbindung mindestens folgende Voraussetzungen erfüllt sein:
  - Die vom anfragenden Client mitgeschickte Applikations-URI muss mit der URI der Server-Applikation des CP übereinstimmen.
  - Wenn das Partnerzertifikat nicht vertrauenswürdig ist, muss der CP zumindest ein selbstsigniertes Zertifikat des Partners gespeichert haben.
  - Wenn das Partnerzertifikat von mehreren CAs ausgestellt wurde, müssen alle CAs im Zertifikatspeicher des CP gespeichert sein.
- **Keine Gültigkeitsdauer prüfen**
  Bei aktivierter Option prüft der CP das Zertifikat des Kommunikationspartners. Der CP lässt Kommunikation auch in folgendem Fall zu:
  - Der aktuelle Zeitpunkt im CP liegt außerhalb der Gültigkeitsdauer des Partnerzertifikats.

Wenn keine der drei Optionen aktiviert ist, werden keine Zertifikate geprüft.

Beachten Sie insbesondere für das Zustandekommen der Kommunikation mit Fremd-Applikationen die Hinweise im Zertifikats-Management zum Import von Zertifikaten.

Besonderheiten für die Client-Applikation des CP 443-1 OPC UA

Wenn Sie die Client-Funktion des CP verwenden, dann beachten Sie folgendes:

Der von Ihnen programmierte Wert des Parameters "CheckServerCertificate" in den Verbindungsinformationen (UASessionConnectInfo) für den Client-Programmbaustein "UA_Connect" wird von den hier, in der Parametergruppe "Zertifikatsvalidierung", projektierten Einstellungen überschrieben. Wenn der Client die Zertifikate eines Kommunikationspartners (Server) prüfen soll, dann können Sie den Parameter im UDT "UASessionConnectInfo" unberücksichtigt lassen. Für die Zertifikatsprüfung sind nur die projektierten Werte der Parametergruppe "Zertifikatsvalidierung" relevant.

OPC UA PubSub - TSN

1.6.1.2 Industrial Ethernet

FTP-Konfiguration

CP als FTP-Server für die S7-CPU-Daten

Ethernet-CP als FTP-Server für die CPU-Daten

Bezug

Projektierung der Betriebsart FTP-Server in der Parametergruppe des Ethernet-CP

- Bei deaktivierter Security unter "Eigenschaften > Allgemein > FTP-Konfiguration"
- Bei aktivierter Security-Funktion unter "Eigenschaften > Security > Allgemein > FTP-Konfiguration"

S7-1500 CP als FTP-Server

Die hier beschriebene Funktion ermöglicht Ihnen, Daten in Form von Dateien (Files) über FTP-Kommandos in Datenbausteine oder aus Datenbausteinen einer S7-Station zu übertragen. Dabei können die üblichen FTP-Kommandos genutzt werden, um Dateien zu lesen, zu schreiben und zu verwalten.

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine an. Wegen ihrer speziellen Struktur werden diese hier als File-DBs bezeichnet.

Der CP 1543-1 als FTP-Server ermittelt bei einem FTP-Kommando aus einer File-Zuordnungstabelle, wie die in der S7-Station für den File-Transfer genutzten Datenbausteine auf Dateien abgebildet werden sollen.

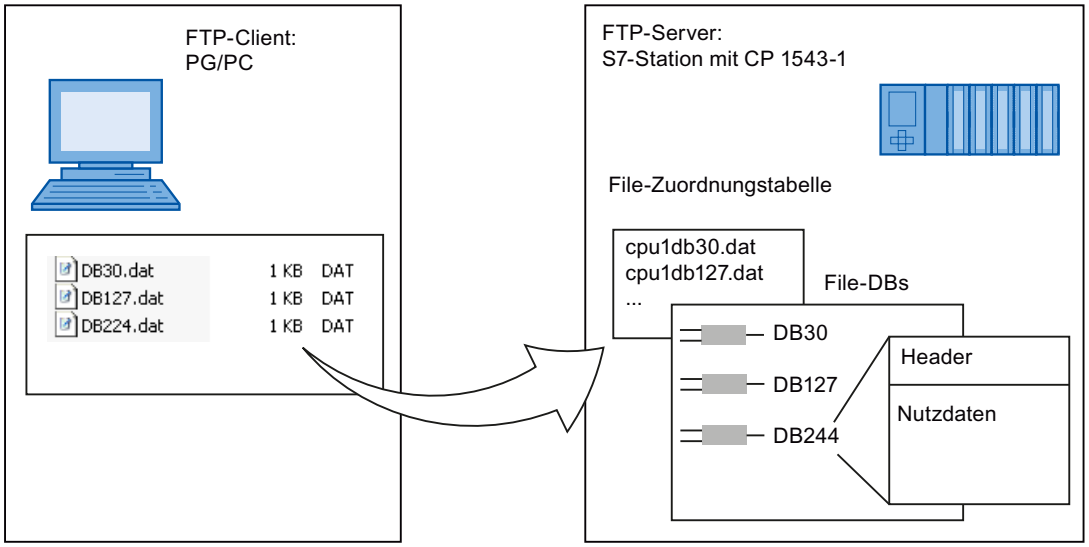

Bild 1-99 S7-CPU mit CP 1543-1 als FTP-Server für die S7 CPU-Daten

File-Zuordnungstabelle

Der CP 1543-1 als FTP-Server benötigt eine Information darüber, wie die in der S7-Station für den File-Transfer genutzten Datenbausteine auf Dateien abgebildet werden sollen.

Die File-Zuordnungstabelle enthält folgende Einträge:

- Zuordnung der CPU
- DB-Zuordnung
- Dateiname

Voraussetzungen in der CPU-Projektierung

Legen Sie als File-DBs Datenbausteine vom Typ "Array of Byte" oder "Array of Char" an.

Verwenden Sie für den CP mit Firmware V1.x folgende Einstellungen um den FTP-Zugriff zu ermöglichen:

- In den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Deaktivieren Sie bei allen als File-DB verwendeten DBs das Attribut "Optimierter Bausteinzugriff".

Diese beiden Einstellungen sind für den CP 1543-1 ab Firmware-Version V2 nicht mehr erforderlich.

Siehe auch

Aufbau und Struktur der FTP-Konfigurationsliste (Seite 3248)

FTP-Kommandos im FTP-Client (Seite 3249)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb (Seite 3251)

Aufbau und Struktur der FTP-Konfigurationsliste

Aufbau und Struktur der FTP-Konfigurationsliste zum Erzeugen der File-Zuordnungstabelle

Die Felder der Tabelle in der FTP-Konfiguration haben folgende Bedeutung und Syntax:

Betitelung in der Tabelle zur FTP-Konfiguration	"CPU"	"DB"	Dateiname	Kommentar
Bedeutung	Zuordnung der CPU Auswählbar über Klappliste; Eintrag nicht eingebbar.	Nr. des Datenbausteins Eintrag eingebbar	Dem File-DB zugewiesener File Name Automatischer Namensvorschlag; Eintrag eingebbar / änderbar.	Informeller Kommentar
Beispiel	cpu1 [PLC_1]	20	cpu1_db20.dat	Messwerte im Bereich Anlage1

Hinweise zur Syntax

- Für den Dateinamen (File Name) eines File-DB gilt:
  - Der Dateiname beginnt mit "cpuX" (mit X=1 bei S7-1500);
  - Länge: maximal 64 Zeichen (einschließlich der Angabe "cpuX");

Hinweis

Beachten Sie die Schreibweise (Kleinbuchstaben für "cpu" und keine führenden Leerzeichen am Zeilenbeginn). Die Dateien werden sonst nicht erkannt.

Siehe auch

Ethernet-CP als FTP-Server für die CPU-Daten (Seite 3246)

FTP-Kommandos im FTP-Client

Ein Zugriffsbeispiel

In der folgenden Darstellung sehen Sie beispielhaft im MS-DOS-Fenster eine typische Zugriffssequenz.

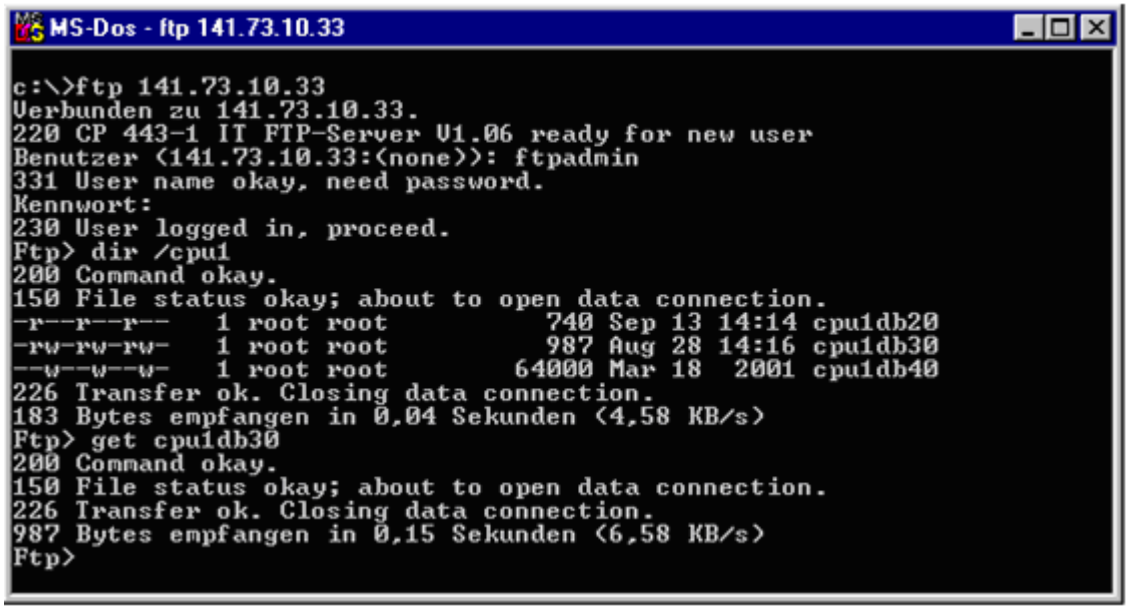

Wie typische FTP-Befehle die zulässigen FTP-Kommandos nutzen

Entnehmen Sie der folgenden Tabelle, welche FTP-Kommandos für den Zugriff auf die File-DBs in der CPU ausgeführt werden können. Die Tabelle zeigt auch, welche FTP-Befehle in typischen Eingabekonsolen, wie beispielsweise die MS-DOS-Eingabeaufforderung, für diese FTP-Kommandos genutzt werden.

Tabelle 1-159 FTP-Kommandos

typische FTP–Befehle						FTP–Komman-do	Bedeutung
open	dir	put	get	close	del		
x						User	Anmelden
x						Pass	Autorisierung durch Passwort
	x	x	x			Port	Angabe des Ports, über den ein Client Daten übertragen will.
	x					List	Listet die File-DBs in der angesprochenen CPU auf.

typische FTP−Befehle						FTP−Kommando	Bedeutung
					x	Dele	Löscht einen File-DB, indem das EXIST-Bit im File-DB-Header auf "0" gesetzt wird.
			x			Retr	Liest die Nutzdaten im angegeben File-DB in die angegebene Datei auf dem FTP-Client.
		x				Stor	Überträgt die angegebene Datei vom FTP-Client in den Nutzdatenbereich im angegeben File-DB.
				x		quit	Beendet die aktuelle FTP-Verbindung.

Hinweis

Die FTP-Kommandos "rename", "append", "rnfr" und "rnto" können Sie nicht auf die File-DBs anwenden.

So werden FTP-Kommandos im CP bearbeitet

Zum besseren Verständnis der FTP-Schnittstelle zu den File-DBs auf der CPU folgt hier beispielhaft am stor-Kommando die Erläuterung des Bearbeitungsablaufes.

Der FTP-Server im CP geht wie folgt vor:

1. Identifizieren des angesprochenen File-DB anhand des Eintrages in der File-Zuordnungstabelle.
2. Prüfen der Bits im File-DB-Header.
   Der Schreibvorgang wird nur ausgeführt, wenn festgestellt wird:
   LOCKED-Bit = 0
   NEW-Bit = 0
   WRITEACCESS-Bit = 1
3. Schreiben des Datei-Inhalts in den Nutzdatenbereich des File-DB in der CPU. Zum Beginn des Schreibvorganges wird das LOCKED-Bit gesetzt und nach abgeschlossenem Schreibvorgang zurückgesetzt.
4. Nach Abschluss des Schreibvorgangs wird zusätzlich im File-DB-Header das NEW-Bit gesetzt und das aktuelle Datum in das Feld DATE_TIME eingetragen.
5. Der FTP-Server sendet eine Meldung über das Ergebnis des Filetransfers an den FTP-Client.

Übertragungsmodus beim Filetransfer

Der Filetransfer erfolgt ausschließlich im Binärmodus. Geben Sie hierzu in der Eingabekonsole nach der Anmeldung den Befehl "binary" ein.

Siehe auch

Ethernet-CP als FTP-Server für die CPU-Daten (Seite 3246)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten

- Abschnitt 1: File-DB-Header (besitzt feste Länge (20 Byte) und Struktur)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

DB mit Variable "FILE_DB_HEADER"

Legen Sie in STEP 7 unter "Programmbausteine" > "Neuen Baustein hinzufügen" einen Datenbaustein an.

Zu den Attributen des DB siehe nachfolgender Abschnitt.

Wählen Sie als Datentyp der ersten Variablen im DB den Datentyp "FILE_DB_HEADER". Die Parameterstruktur des SDT wird automatisch angelegt.

Voraussetzungen in den DB-Attributen

Verwenden Sie folgende Einstellungen, um den FTP-Zugriff zu ermöglichen:

Deaktivieren Sie bei allen als File-DB verwendeten Datenbausteinen das Attribut "Optimierter Bausteinzugriff" aktivieren.

Hinweis

Optimierter Zugriff ab CP-Firmware V2.0

Wenn Sie in STEP 7 ab V14 den CP 1543-1 ab Firmware-Version V2.0 verwenden, dann können Sie die Option "Optimierte Bausteinzugriff verwenden.

File-DB-Header für FTP-Server-Betrieb

Anmerkung: Der hier beschriebene File-DB-Header ist weitgehend identisch zu dem für den Client-Betrieb beschriebenen File-DB-Header. Die Unterschiede beziehen sich auf die Parameter

- WRITE_ACCESS
- FTP_REPLY_CODE

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten ("Datei existiert nicht"). • 1: Der File-DB enthält gültige Nutzdaten ("Datei existiert").	Das FTP-Kommando dele setzt EXIST=0; Das FTP-Kommando store setzt EXIST=1;
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos store und retr setzen während der Bearbeitung LOCKED=1. Für einen Schreibvorgang aus dem Anwenderprogramm gilt: Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes LOCKED setzen bzw. zurücksetzen. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen; wenn =0 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen; wenn =0 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen

Parameter	Typ	Wert / Bedeutung	Versorgung
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muss nach dem Lesen der Daten NEW=0 setzen, um ein erneutes store zu ermöglichen oder um das File mittels FTP-Kommando dele löschen zu können.
WRITE_ACCESS	BOOL	0: Der FTP-Client auf dem PG/PC hat kein Schreibrecht für die File-DBs in der S7-CPU. 1: Der FTP-Client auf dem PG/PC hat Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungswert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereiches (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Der Parameter ist im FTP-Serverbetrieb ohne Bedeutung.	Wird vom FTP-Server auf "0" gesetzt.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung der Datei. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wird die Funktion "Uhrzeitweiterleitung" genutzt, entspricht der Eintrag der weitergeleiteten Zeit. Wird die Funktion "Uhrzeitweiterleitung" nicht genutzt, wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des IT-CP (Initialisierungswert ist 1.1.1994 0.0 Uhr).

Siehe auch

Ethernet-CP als FTP-Server für die CPU-Daten (Seite 3246)

CP als FTP-Client für die S7-CPU-Daten

Funktionsweise - FTP-Client

FTP-Client

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an.

Das Anwenderprogramm setzt FTP-Aufträge ab, die vom CP 1543-1 als FTP-Client ausgeführt werden. Im Anwenderprogramm verwenden Sie die Anweisung FTP_CMD für den Verbindungsaufbau und die Datenübertragung.

Im Auftrag geben Sie unter anderem an:

- IP-Adresse des FTP-Servers,
- Ablageort der Datei auf dem FTP-Server
- Dateinamen
- Zugangsinformationen

Die folgenden Abbildungen zeigen die Funktionsweise bei Einsatz der Anweisung FTP_CMD.

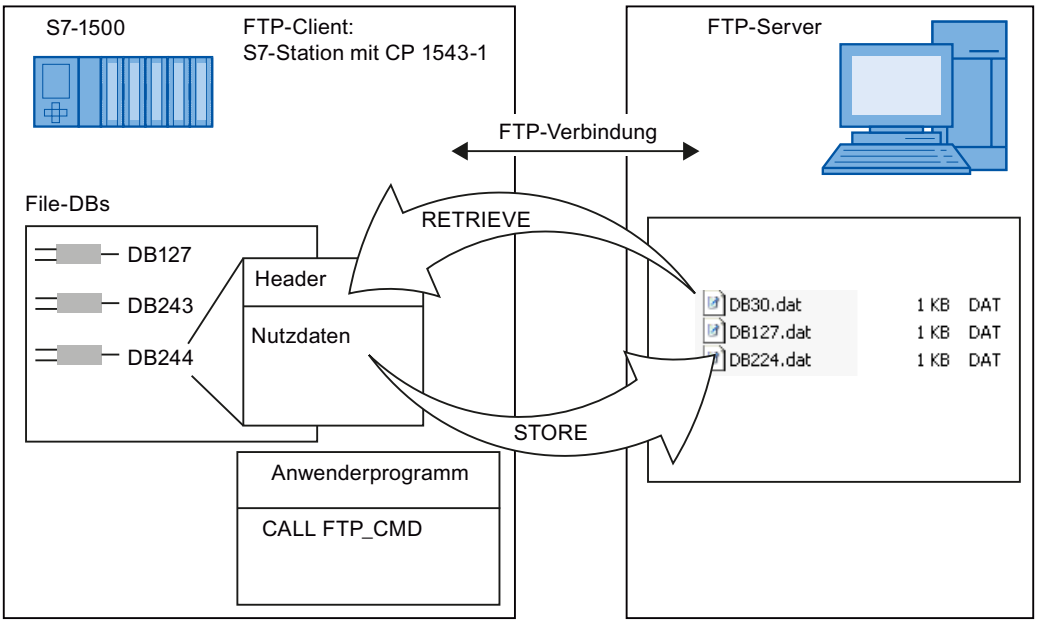

Bild 1-100 CPU mit CP 1543-1 als FTP-Client für die S7 CPU-Daten

Ab Firmware-Version V2 des CP können im DB für die Nutzdaten auch die Datentypen "Array of Byte" und "Array of Char" verwendet werden.

Voraussetzungen in der CPU-Projektierung

Wenn Sie den CP 1543-1 mit Firmware-Version V1.x verwenden, dann nehmen Sie folgende Einstellungen vor, um den FTP-Zugriff zu ermöglichen:

- In den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Bei allen als File-DB verwendeten Datenbausteinen: Deaktivieren Sie das Attribut "Optimierter Bausteinzugriff".

Diese Einstellungen sind für den CP 1543-1 ab Firmware-Version V2 nicht mehr erforderlich.

Betriebsart FETCH/WRITE

Anwendung von FETCH/WRITE

FETCH/WRITE

Mit den FETCH/WRITE-Diensten greifen Sie auf Systemspeicherbereiche der S7-CPU zu.

- FETCH: Daten direkt lesen
- WRITE: Daten direkt schreiben

Der Zugriff ist von folgenden Geräten aus möglich:

- SIMATIC S5
- SIMATIC PC-Stationen
- Fremdgeräte

Der S7-1500-CP ist Server für die FETCH/WRITE-Dienste.

Die FETCH/WRITE-Dienste werden von folgenden Protokollen unterstützt:

- TCP
- ISO-on-TCP
- ISO-Transport

Für die Nutzung der FETCH/WRITE-Dienste wird keine projektierte Verbindung benötigt.

Systemspeicher

Über FETCH oder WRITE können Sie auf folgende Operandenbereiche im Systemspeicher der S7-1500-CPU zugreifen:

- Datenbausteine (DB)
  Beachten Sie für den DB-Zugriff folgende Beschränkung: die höchste DB-Nr ist 255.
  Beachten Sie in der die Projektierung der TSAP-Tabelle des CP den Parameter "S7-Adressierung" für den Zugriff von S5- oder S7-Partnern.
- Merker (M)
- Prozessabbild der Eingänge (PAE)

- Prozessabbild der Ausgänge (PAA)
- Peripheriebereich Eingänge (PEW, PED, PEB)
- Peripheriebereich Ausgänge (PAW, PAD, PAB)
- Zähler (Z)
- Zeiten (T)

Voraussetzungen in der CPU-Projektierung

Verwenden Sie in den Projektierungsdaten der CPU folgende Einstellung, um den FETCH/WRITE-Zugriff zu ermöglichen:

- Parametergruppe "Schutz & Security > Verbindungsmechanismen":
  Deaktivieren Sie die Option "Zugriff über PUT/GET-Kommunikation durch entfernten Partner erlauben".

Zur Projektierung des CP siehe Kapitel FETCH/WRITE-Zugriff konfigurieren (Seite 3256).

Kopplung zu Fremdsystemen

Die bei ISO-Transportverbindungen und ISO-on-TCP- bzw. TCP-Verbindungen unterstützte Betriebsart für FETCH und WRITE kann prinzipiell von beliebigen Fremdgeräten aus für den Zugriff auf S7-Systemspeicherbereiche genutzt werden.

Damit Sie diesen Zugriff z. B. auch für PC-Anwendungen implementieren können, müssen Sie den PDU-Aufbau für die Aufträge kennen. Beachten Sie hierzu das Kapitel Kopplung zu Fremdsystemen mit FETCH/WRITE (Seite 3258).

FETCH/WRITE-Zugriff konfigurieren

Zu den Mechanismen von FETCH/WRITE siehe Kapitel Anwendung von FETCH/WRITE (Seite 3255).

FETCH/WRITE

Aktivieren Sie die Option, um über den CP den FETCH/WRITE-Zugriff auf die S7-1500-CPU freizuschalten.

Legen Sie in der Tabelle für jeden Partner für den FETCH/WRITE-Zugriff eine Zeile an und tragen Sie die Zugriffsidentifikationen in eine Tabelle ein. Nur FETCH/WRITE-Zugriffe, deren Identifikation mit einem Eintrag in der Liste übereinstimmen, werden zugelassen.

Abhängig vom genutzten Protokoll für den FETCH/WRITE-Zugriff sind folgende Eingaben erforderlich:

- ISO-on-TCP / ISO-Transport
  TSAP des Partners
- TCP
  Portnummer des Partners
  Zu den nutzbaren Portnummern siehe Kapitel Port-Einstellungen (Seite 3257).

TSAP-Liste / TCP-Port-Liste

- TSAP
  Geben Sie den TSAP des Partners als ASCII-String ein oder nutzen Sie alternativ die Zelle der Spalte "TSAP als hex-Format".
  Bei Eingabe in die Spalte "TSAP als hex-Format" wird der korrespondierende Wert in dieser Spalte automatisch ergänzt.
- TSAP als hex-Format
  Geben Sie den TSAP des Partners als Hexadezimalwert ein oder nutzen Sie alternativ die Zelle der Spalte "TSAP".
  Bei Eingabe als Hexadezimalwert wird jedes Zeichen des TSAP als Hexadezimalzahl eingegeben, getrennt durch Punkte.
  Bei Eingabe in die Spalte "TSAP" wird der korrespondierende Wert in dieser Spalte automatisch ergänzt.
- TCP-Port
  Geben Sie die Portnummer des Partners ein.
- S7-Adressierung
  Legen Sie mit dieser Option für den FETCH/WRITE-Aufruf in der SIMATIC S7-Station den Adressierungsmodus beim Zugriff auf DBs fest:
  - Option aktiviert
    S7-Adressierung: Byte-Adresse
    Dies ist die Standardeinstellung für den Zugriff von SIMATIC S7-Stationen.
  - Option deaktiviert
    S5-Adressierung: Wort-Adresse
    Durch diese Adressierung wird ermöglicht, dass Applikationen ohne Anpassung der Adressen auf S5- oder S7-Stationen zugreifen. Dies ist beispielsweise für bestehende S5-Applikationen nützlich, die unverändert für den Zugriff auf S7-Stationen verwendet werden sollen.

Port-Einstellungen

Port-Einstellungen

Die folgende Tabelle gibt Aufschluss über den Werte- und Nutzungsbereich von Port-Adressen, die in den Verbindungsbeschreibungen der OUC-Programmbausteine referenziert werden.

Port-Adressen	Anwendung/Hinweis
0	Fest vergeben; darf nicht verwendet werden!
1 .. 1023	Ports aus diesem Band sollten nicht verwendet werden (well known ports).

Port-Adressen		Anwendung/Hinweis
1024 .. 49151	Ports für anwendungsspezifische Protokolle	
	2000 .. 5000	Bereich, in dem von STEP 7 freie Port-Adressen gesucht und zugeordnet werden. Sie können die Port-Adresse in diesem Bereich individuell einstellen.
	5001 .. 49151	Die Port-Adressen ab 5000 werden vom System genutzt! Anmerkung: Wenn Sie diese Port-Adressen nutzen möchten, wenden Sie sich zur Prüfung der Verfügbarkeit ggf. an Ihren Systemadministrator.
49152 .. 65535		Dynamisch zugewiesene Ports Es wird empfohlen, diese Ports nicht zu verwenden.

Kopplung zu Fremdsystemen mit FETCH/WRITE

Die Betriebsart für FETCH und WRITE kann prinzipiell von beliebigen Fremdgeräten aus für den Zugriff auf S7-Systemspeicherbereiche genutzt werden.

Damit Sie diesen Zugriff z. B. auch für PC-Anwendungen implementieren können, müssen Sie den PDU-Aufbau für die Aufträge kennen. Die benötigten S7- bzw. S5-spezifische Header für Anforderungs- und Quittungstelegramme sind standardmäßig 16 Byte lang. Zur Struktur siehe nachfolgende Tabellen.

Struktur für WRITE-Telegramm (PDU)

Die Bedeutung und Werte der Parameter, die nicht mit festen Werten belegt sind, entnehmen Sie dem Abschnitt "Parameterversorgung" unten.

WRITE-Anforderungstelegramm			WRITE-Quittungstelegramm		
0	Systemkennung	="S"	0	Systemkennung	="S"
1		="5"	1		="5"
2	Länge Header (Byte)	=16 (dez.)	2	Länge Header	=16 (dez.)
3	Kennung OP-Code	=01	3	Kennung OP-Code	=01
4	Länge OP-Code	=03	4	Länge OP-Code	=03
5	OP-Code	=03	5	OP-Code	=04
6	Kennung ORG-Block	=03	6	Quittungsblock	=0Fh
7	Länge ORG-Block	=08	7	Länge Quittungsblock	=03
8	ORG-Kennung		8	Fehlerblock	=Nr.
9	DBNR		9	Kennung Leerblock	=FFh
A	Anfangsadresse	High Byte	A	Länge Leerblock	=07
B		Low Byte	B		
C	Länge	High Byte	C		
D		Low Byte	D	frei	
E	Kennung Leerblock	=FFh.	E		
F	Länge Leerblock	=02	F		
	Datenblock mit bis zu 64 kB				

Struktur für FETCH-Telegramm (PDU)

Die Bedeutung und Werte der Parameter, die nicht mit festen Werten belegt sind, entnehmen Sie dem Abschnitt "Parameterversorgung" unten.

FETCH-Anforderungstelegramm			FETCH-Quittungstelegramm		
0	Systemkennung	="S"	0	Systemkennung	="S"
1		="5"	1		="5"
2	Länge Header	=0x10	2	Länge Header	=0x10
3	Kennung OP-Code	=0x01	3	Kennung OP-Code	=0x01
4	Länge OP-Code	=0x03	4	Länge OP-Code	=0x03
5	OP-Code	=0x05	5	OP-Code	=0x06
6	Kennung ORG-Block	=0x03	6	Quittungsblock	=0x0F
7	Länge ORG-Block	=0x08	7	Länge Quittungsblock	=0x03
8	ORG-Kennung		8	Fehlerblock	=Nr.
9	DBNR		9	Kennung Leerblock	=0xFF
A	Anfangsadresse	High Byte	A	Länge Leerblock	=0x07
B		Low Byte	B	frei	
C	Länge	High Byte	C		
D		Low Byte	D		
E	Leerblock	=0xFF	E		
F	Länge Leerblock	=0x02	F		
				Daten bis zu 64 kB, jedoch nur, wenn Fehler-Nr. = 0.	

Parameterversorgung

S7-Operandenbereich	DB	M	E	A
ORG-Kennung	01_H	02_H	03_H	04_H
	Quell-/Zieldaten aus/ in Datenbaustein im Hauptspeicher	Quell-/Zieldaten aus/ in Merkerbereich	Quell-/Zieldaten aus/ in Prozessabbild der Eingänge (PAE)	Quell-/Zieldaten aus/ in Prozessabbild der Ausgänge (PAA)
DBNR	DB, aus dem die Quelldaten entnommen werden bzw. in den die Zieldaten transferiert werden.	irrelevant	irrelevant	irrelevant
Erlaubter Bereich	1...255			
Anfangsadresse	DW-Nummer, ab der die Daten entnommen bzw. eingeschrieben werden.	Merker-Byte-Nr., ab der die Daten entnommen bzw. eingeschrieben werden.	Eingangs-Byte-Nr., ab der die Daten entnommen bzw. eingeschrieben werden.	Ausgangs-Byte-Nr., ab der die Daten entnommen bzw. eingeschrieben werden.
Erlaubter Bereich	0...2047	Alle Merker-Bytes, die eine CPU zur Verfügung stellt.	0...127	0...127

Länge	Länge des Quell-/ Ziel-datenblocks in Worten	Länge des Quell-/ Ziel-datenblocks in Byte	Länge des Quell-/ Zieldatenblocks in Byte	Länge des Quell-/ Zieldatenblocks in Byte
Erlaubter Bereich	Bis zu 8192 Byte	Bis zu 8192 Byte	1...128	1...128

S7-Operandenbereich	PE/PA	Z	T
ORG-Kennung	05_H	06_H	07_H
	Quell-/Ziel-Daten aus/in Peripheriebaugruppen Bei Quelldaten Eingabe-baugruppen Bei Zieldaten Ausgabe-baugruppen	Quell-/Zieldaten aus/in Zählerzellen	Quell-/Zieldaten aus/in Zeitenzellen
DBNR	irrelevant	irrelevant	irrelevant
Anfangsadresse	Peripherie-Byte-Nr., ab der die Daten entnommen bzw. eingeschrieben werden.	Nummer der Zählerzelle, ab der die Daten entnommen bzw. eingeschrieben werden	Nummer der Zeitenzelle, ab der die Daten entnommen bzw. eingeschrieben werden.
Erlaubter Bereich	0...127: Digitale Peripherie 128...255: Aanaloge Peripherie	0...255	0...255
Anzahl	Länge des Quell-/Zieldatenblocks in Byte	Länge des Quell-/Zieldatenblocks in Worten (Zählerzelle = 1 Wort)	Länge des Quell-/Zieldatenblocks in Worten (Zählerzelle = 1 Wort)
Erlaubter Bereich	1...256	1	1

Siehe auch

Anwendung von FETCH/WRITE (Seite 3255)

1.6.1.3 CP PROFIBUS

Uhrzeitsynchronisation

Bezug

Projektierung der Uhrzeitsynchronisation von PROFIBUS-CPs.

Uhrzeitsynchronisation

Stellen Sie ein, ob der CP Uhrzeittelegramme weiterleiten soll oder nicht.

Je nachdem, wo der Uhrzeit-Master angeordnet ist, sind die beiden folgenden Fälle zu unterscheiden:

- Uhrzeittelegramme kommen vom Subnetz und werden in die Station weitergleitet.
- Uhrzeittelegramme kommen aus der Station und werden in das Subnetz weitergleitet.

Bei einigen Baugruppen geben Sie die Richtung, in welche die Uhrzeittelegramme weitergeleitet werden, gezielt an. Bei Baugruppen, bei denen das Weiterleiten nur global ein- oder ausschaltbar ist, kann der CP die Uhrzeittelegramme in beliebige Richtungen weiterleiten.

Wenn Sie in Ihrer Station mehrere CPs betreiben, dann bedenken Sie den Fluss der Uhrzeittelegramme abhängig vom Uhrzeitmaster. Da Sie über die hier beschriebene Funktion Uhrzeittelegramme von einem Netz in ein anderes Netz einkoppeln können, sind Konfliktsituationen möglich.

Wenn in einer Station mehrere CPs vorhanden sind, die an das selbe Netz angeschlossen sind, darf nur ein einziger CP die Uhrzeitsynchronisations-Nachrichten weiterleiten.

Parametrieren von Feldgeräten (Datensatz-Routing)

Bezug

Parametrierung der Eigenschaften für PROFIBUS-CPs in der Parametergruppe "Eigenschaften > Einstellungen > Parametrieren von Feldgeräten".

Parametrieren von Feldgeräten (Datensatz-Routing)

Indem Sie diese Option wählen, können Sie den CP als Router für Datensätze verwenden, die an Feldgeräte (DP-Slaves) gerichtet sind. Der CP leitet dann Datensätze, die von Geräten übertragen werden, die nicht direkt am PROFIBUS angeschlossen und somit keinen direkten Zugriff auf die Feldgeräte (DP-Slaves) haben, an diese weiter.

Ein Werkzeug, das solche Datensätze zur Parametrierung von Feldgeräten erzeugt, ist SIMATIC PDM (Process Device Manager).

Die Funktion ist standardmäßig eingeschaltet. Da die Funktion zusätzliche Speicher-Ressourcen beansprucht, können Sie die Option dann ausschalten, wenn Sie die Speicher-Ressourcen des CPs insgesamt hoch auslasten (Verbindungen etc.) und die Funktion "Datensatz-Routing" nicht benötigen.

CPs mit DP-Funktionalität - Betriebsart wählen

Bezug

Parametrierung der CP-Betriebsart für PROFIBUS DP in der Parametergruppe "Eigenschaften > Allgemein > Betriebsart".

Zusammenhang zwischen CP-Betriebsart und CP-Funktion

Projektierbare CP-Betriebsarten und dadurch mögliche CP-Funktionen entnehmen Sie dieser Tabelle:

CP-Betriebsart	mögliche CP-Funktionen				
	PG/BuB über PROFIBUS	FDL über PROFIBUS	DP-Master	DP-Slave	projektierte S7-Verbindungen
Kein DP-Betrieb	X	X	-	-	X
DP-Master-betrieb	X	X	X	-	X
DP-Slave aktiv (siehe unten)	X	X	-	X	-
DP-Slave passiv (siehe unten)	-	-	-	X	-

Legende: "X" = Funktion wird unterstützt; "-" = Funktion wird nicht unterstützt

DP Masterbetrieb / Parameter "DP-Verzögerungszeit"

Hier können Sie eine zusätzliche Verzögerung für die Datenaktualisierung der DP E/A-Daten einstellen. Diese Verzögerungszeit wird vom PROFIBUS-CP für die Bearbeitung anderer Protokolle wie z.B. FDL-Verbindungen genutzt.

Die angezeigten DP-Reaktionszeiten werden erst dann aktualisiert, wenn Sie eine der folgenden Aktionen ausführen:

- die Schaltfläche "Neu berechnen..." betätigen oder
- eine andere Parametergruppe wählen oder

Wenn viele DP-Slaves projektiert sind, kann diese Neuberechnung einige Sekunden beanspruchen.

DP-Master Klasse 2 (Diagnosefunktionen - nur bei bestimmten CP-Typen wählbar))

In dieser Betriebsart kann das Gerät über den PROFIBUS-CP Ein- und Ausgangsdaten von beliebigen, ihm nicht zugeordneten DP-Slaves lesen.

Damit kann beispielsweise ein Prozesssignal von mehreren DP-Mastern erfasst werden, was entsprechend zu Einsparungen von Sensoren im Feld führen kann (shared input / shared output). Typischerweise arbeiten Programmier-, Diagnose- oder Management-Geräte in der Eigenschaft als DP-Master Klasse 2.

DP-Slavebetrieb / Option "Test, Inbetriebnahme, Routing" (Funktionsangabe variiert je nach CP-Typ)

- Option aktiviert: DP-Slave aktiv
  Aktivieren Sie das Optionskästchen, wenn Sie z. B. PG-Funktionen, die bei der Inbetriebnahme und beim Testen benötigt werden, häufig über diese Schnittstelle durchführen wollen. Sie wählen damit die CP-Betriebsart "DP-Slave aktiv". (d. h. die Schnittstelle ist am Token-Umlauf des PROFIBUS beteiligt).
  Folgende Funktionen sind dann möglich (Funktionsangabe variiert je nach CP-Typ):
  - Programmieren (z. B. laden)
  - Testen (Status/Steuern)
  - S7-Routing (I-Slave als Netzübergang)
  - S5-kompatible Kommunikation (SEND/RECEIVE-Schnittstelle)
  - S7-Kommunikation (nur im Server-Betrieb auf einseitig projektierten Verbindungen ohne Kommunikations-Bausteine; Projektierung beim Partner)

  Hinweis

  Besondere Auswirkungen der Betriebsart "DP-Slave aktiv"
  - S7-Verbindungen für den Datenaustausch über Kommunikations-Bausteine sind in dieser Betriebsart nicht projektierbar.
  - Die Busumlaufzeit kann sich verlängern. Bei zeitkritischen Anwendungen, und wenn S7-Routing sowie die Client-Funktionalität für die Kommunikation nicht erforderlich sind, sollte diese Option deshalb nicht aktiviert werden.

- Option deaktiviert: DP-Slave passiv
  Wenn das Optionskästchen "Test, Inbetriebnahme, Routing" nicht aktiviert ist, wählen Sie damit die CP-Betriebsart "DP-Slave passiv"; dann arbeitet die Schnittstelle nur als Server für Kommunikationsdienste und ist am Token-Umlauf des PROFIBUS nicht beteiligt. Server-Funktionalität bedeutet, dass nur der Partner (z. B. PG, OP oder Automatisierungssystem) die Kommunikation initiieren kann. Folgende Besonderheiten sind bei nicht aktivierter Option zu beachten:
  - S7-Routing ist nicht möglich.
  - Programmieren und Testen ist möglich, aber langsam (keine Verlängerung der Busumlaufzeit).

CP unterstützt mehrere DP-Slavetypen

Bezug

Parametrierung des DP-Slavetyps für PROFIBUS DP in der Parametergruppe "Eigenschaften > Allgemein > Betriebsart".

DP-Mode

Bei CPs, die unterschiedliche DP-Slavetypen unterstützen, können Sie je nach CP-Typ aus den folgenden DP-Modi auswählen:

- DPV1-Funktionalität (voreingestellte Betriebsart)
  Hierunter fallen DP-Slaves nach PROFIBUS DPV0 und DPV1 Norm sowie Siemens DP-Slaves.
  Diese Betriebsart setzt bestimmte CPU-Typen voraus. Beachten Sie für Ihren PROFIBUS-CP die Angaben im Gerätehandbuch für S7-CPs.
- DPV0 kompatibel
  Hierunter fallen DP-Slaves nach PROFIBUS DPV0 Norm; DP-Slaves nach DPV1 Norm sind nur mit eingeschränkter Funktionalität verwendbar.
- S7-kompatibel
  Hierunter fallen DP-Slaves nach PROFIBUS DPV0 Norm sowie Siemens DP-Slaves; DP-Slaves nach DPV1 Norm sind nur mit eingeschränkter Funktionalität verwendbar.

1.6.1.4 CM/CP PROFIBUS

Projektierung der Uhrzeitsynchronisation

Uhrzeitsynchronisationsrollen des Kommunikationsmoduls

Über die Parameter der Uhrzeitsynchronisation legen Sie fest, welche Rolle das Kommunikationsmodul für die Weitergabe von Uhrzeittelegrammen spielt.

Abhängig von der Betriebsart des Moduls am PROFIBUS sind folgende Synchronisationsrollen einstellbar:

- Das Kommunikationsmodul ist passiver Teilnehmer am PROFIBUS (z. B. DP-Slave):
  - Slave
  - Keine
- Das Kommunikationsmodul ist aktiver Teilnehmer am PROFIBUS:
  - Master
  - Slave
  - Keine

Bedeutung der Parameter

Die Parameter der Uhrzeitsynchronisation haben folgende Bedeutung:

- Synchronisationsart
  Der Parameter legt Synchronisationsrolle fest:
  - Slave
    Die CM-Uhr wird von einer anderen Uhr synchronisiert. Die Synchronisationstelegramme kommen vom PROFIBUS. Das CM gibt in dieser Synchronisationsart die Uhrzeittelegramme in einem festgelegten Zeitintervall an die Station weiter.
    Innerhalb der S7-Station darf es nur einen Uhrzeit-Slave geben.
  - Master
    Die Uhr des Kommunikationsmoduls sendet als Master Uhrzeittelegramme an das PROFIBUS-Netz.
    Wenn die Synchronisationsart "Master" eingestellt ist, dann ist das Zeitintervall für die Ausgabe der Uhrzeittelegramme wählbar.
  - Keine
    Es findet keine Synchronisation statt.

  Wenn in einem Netz mehr als eine Baugruppe mit Uhr vorhanden ist, müssen Sie einstellen, welche Baugruppe bei der Uhrzeitsynchronisation als Uhrzeit-Master und welche als Uhrzeit-Slave fungieren soll. Es darf nur einen Uhrzeit-Master geben.
- Aktualisierungszyklus
  Der Parameter definiert in der Synchronisationsrolle "Master" das Aktualisierungsintervall, in dem Uhrzeittelegramme an das Netz ausgegeben werden.

1.6.1.5 Protokollunabhängige / Netzunabhängige CP-Einstellungen

Baugruppentausch ohne PG

Bezug

Parametrierung besonderer Einstellungen in der Parametergruppe "Eigenschaften > Allgemein > Einstellungen"

Abhängig vom CP-Typ finden Sie die Einstellmöglichkeit "Baugruppentausch ohne PG"

Auswirkung - ausfallsichere Langzeitspeicherung in der CPU

Wenn Sie diese Option gewählt haben, erfolgt die ausfallsichere Langzeitspeicherung in der CPU anstatt im EEPROM des CP. Beachten Sie jedoch, dass auch auf der CPU nur dann eine ausfallsichere Langzeitspeicherung gegeben ist, sofern diese durch Batteriepufferung oder durch SIMATIC Memory Card gegen Spannungsausfall abgesichert ist.

Ressourcenbedarf auf der CPU

Wenn Sie diese Option wählen, belegen Sie zusätzliche Ressourcen auf Ihrer CPU. Beim Laden der Anwenderprogramme und der Projektierdaten werden Sie informiert, wenn nicht genügend Speicherplatz zur Verfügung steht. Ressourcenengpässe können durch die Verwendung einer SIMATIC Memory Card umgangen werden

Hinweis

Wenn Sie Ressourcenengpässe annehmen müssen und zunächst keine SIMATIC Memory Card verwenden wollen, können Sie auch zunächst auf die Option verzichten und die Projektierdaten im CP sichern. Zu einem späteren Zeitpunkt können Sie eine SIMATIC Memory Card mit den Projektierdaten dann so erstellen, dass die Option "Baugruppentausch ohne PG" darin eingeschaltet ist. Wenn Sie daraufhin die SIMATIC Memory Card in die CPU stecken, können Sie den CP anschließend jederzeit tauschen. Die Projektierdaten werden dann automatisch beim CP-Anlauf aus der CPU bzw. aus der SIMATIC Memory Card geladen.

OP-Verbindungen multiplexen / interne CPU Verbindungs-Ressource belegen

Bezug

Parametrierung besonderer Einstellungen in der Parametergruppe "Eigenschaften > Allgemein > Einstellungen"

Abhängig vom CP-Typ finden Sie die Einstellmöglichkeit "Multiplex-OP-Verbindungen"

Multiplexbetrieb

Für den Anschluss von TD/OPs beziehungsweise HMI-Geräten können Sie die Verbindungs-Ressourcen in der S7-300 CPU optimieren, indem Sie bis zu 16 dieser Geräte auf einer einzigen CPU Verbindungs-Ressource kommunizieren lassen können (Multiplexbetrieb).

Wenn Sie den Multiplexbetrieb nicht nutzen, ist die Anzahl der betreibbaren TD/OPs beziehungsweise HMI-Geräten von der Anzahl der verfügbaren Verbindungs-Ressourcen der verwendeten CPU abhängig.

Interne CPU Verbindungs-Ressource belegen

Projektierte S7-Verbindungen über den CP benutzen den selben Multiplexkanal, den Sie beim Multiplexbetrieb für die HMI-Verbindungen belegen. Wenn Sie also S7-Verbindungen projektieren, wird dadurch bereits eine CPU-Verbindungs-Ressource belegt.

Hinweis

PG-Verbindungen werden nicht über den Multiplexer betrieben; für den Betrieb eines PGs wird immer eine Verbindungs-Ressource belegt.

Hinweis

Adressierung im Multiplexbetrieb

Im Multiplexbetrieb muss bei der Adressierung für die TD/OP/HMI-Verbindungen die Baugruppenträger/Steckplatz-Zuordnung des CPs anstelle der Baugruppenträger/Steckplatz-Zuordnung der CPU angegeben werden!

Anwendungen (beispielsweise ProAgent), welche bausteinbezogene Meldungen (Alarm_S: SFC 17 , SFC 18 , SFC 19) erfordern, werden im Multiplexbetrieb nicht unterstützt.

1.6.1.6 Hintergrund zu Verbindungen

Window Time und Inactivity Time - Zusammenhang

Bezug

Einstellungen zu ISO-Transportverbindungen / Parametergruppe "Eigenschaften > Allgemein > Dynamik".

Verbindungsabbrüche durch geeigneten Abgleich vermeiden

Lebenszeichen-Telegramme werden von der Partnerstation mit einem Telegramm beantwortet. Sie werden deshalb im Zeitabstand der Window Time zur Partnerstation gesendet.

Damit es nicht zu unerwünschten Verbindungsabbrüchen kommt, sollte die Inactivity Time mindestens dreimal so groß sein wie die Window Time.

Option "Sofortige Rückmeldung bei erkannter Verbindungsunterbrechung"

Bedeutung

Indem sie die Option "Sofortige Rückmeldung bei erkannter Verbindungsunterbrechung" wählen legen Sie fest, dass die Applikation bei nicht verfügbarer Verbindung sofort eine Anzeige über einen nicht ausführbaren Kommunikationsauftrag (FETCH oder WRITE) erhält. Die Anzeige erfolgt unabhängig davon, ob ein gerade laufender Versuch, die Verbindung erneut aufzubauen, gelingt oder nicht.

Wenn sie diese Option nicht wählen, erhält die Applikation im Falle einer nicht verfügbaren Verbindung erst dann eine Anzeige, wenn der Verbindungs-Timeout abgelaufen ist.

Die hier beschriebene Option ist nur dann relevant und verfügbar, wenn die beiden nachfolgend genannten Betriebsarten verwendet werden:

- Die Verbindung ist als permanente Verbindung projektiert.
  Verbindungen können als permanent aufgebaute Verbindung projektiert sein (Option im Register "Details zum Verbindungsaufbau"). Dadurch können Aufträge mit kürzest möglicher Reaktionszeit zum Partner weitergeleitet werden, ohne dass zunächst ein Verbindungsaufbau erforderlich ist.
  Ist die Verbindung gestört, wird automatisch versucht, die Verbindung neu aufzubauen. In diesem Zustand kann es zu Verzögerungen kommen, bis eine entsprechende Meldung an die Applikation weitergeleitet wird.
- Sie verwenden die Auftragsarten FETCH oder WRITE (Auswahlmöglichkeit im Register "Optionen".

Nur bei diesen Auftragsarten ist in bestimmten Anwendungsfällen eine unmittelbare Reaktion bei fehlender Verbindung erforderlich.

Adressangaben - E-Mail Verbindung

Parameter für E-Mail Zustellung

Unter "Eigenschaften > Adressdetails" geben Sie die Parameter ein, die für die Zustellung der E-Mail benötigt werden. Folgende Parameter sind eingebbar:

Parameter	Beschreibung	Beispiel
E-Mail Server (SMTP)	Adresse des Mail-Servers, über den die E-Mails gesendet werden. Die IP-Adresse kann absolut oder symbolisch angegeben werden. Für die symbolische Eingabe gilt: Ein gültiger Name hat 1 bis 64 Zeichen; von diesen Zeichen muss mindestens 1 Zeichen ein Buchstabe sein. Werden in der Eingabe ein oder mehrere Zeichen "." verwendet, gilt zusätzlich: Ein gültiger Name muss hinter dem letzten "." 1 bis 63 weitere Zeichen haben, wobei nur die Zeichen [a-z], [A-Z], [0-9] or [-] erlaubt sind und mindestens eines dieser Zeichen ein Buchstabe sein muss. Die symbolische Angabe setzt voraus, dass dem Internet-CP die Adresse des Domain Name System (DNS) bekannt ist. Ein entsprechender Eintrag ist bei der Projektierung des Internet-CP in HWKonfig vorzunehmen; nähere Angaben hierzu finden Sie dort in der Online-Hilfe. Maximal eingebbar sind 64 Zeichen.	absolut:140.80.0.4 symbolisch:t-online.de
Default Absendername	Angabe einer Adresse, die immer dann in die E-Mail als Absenderadresse eingefügt wird, wenn im Header der E-Mail die Absenderangabe (Parameter FROM) leer ist. Die Angabe ist in der Regel ohne Bedeutung für die Mail-Zustellung. Sie dient zur Information für den Mail-Empfänger. Da einige Mail-Server Aufträge nicht weiterleiten, wenn die Absenderangabe leer ist, **wird empfohlen, einen Eintrag vorzunehmen!** Maximal eingebbar sind 126 Zeichen. **Hinweis:** Beachten Sie, dass ein Default Absendername angegeben sein muss, wenn von der Diagnose aus eine Test-Mail initiiert werden soll.	Station2.CPU412@xy.werk2.de

Symbole im Offline Status

Symbole im Offline-Status

Der Offline-Status einer Verbindung wird wie folgt dargestellt:

Symbol	Bedeutung / Verbindungsstatus
	Konsistent - Verbindungsprojektierung ist ladbar.
	Inkonsistent - Verbindungsprojektierung ist nicht ladbar.

ISO-on-TCP Verbindung: TSAP

TSAP-Format

ISO-on-TCP-Verbindungen haben eine TSAP-Länge von 1 bis 16 Byte. Bei der Eingabe wird die aktuelle Länge automatisch angezeigt. Lokale TSAPs und Partner-TSAPs können als Hexadezimal-Wert oder als ASCII-String eingegeben werden. Bei ASCII-Eingabe werden die eingegebenen Zeichen auch hexadezimal angezeigt. Bei Hexadezimal-Eingabe werden druckbare Zeichen als ASCII-Wert (sichtbar sind 8 Hexadezimalzeichen) dargestellt. Wenn Sie nicht druckbare Zeichen eingegeben, wird die ASCII Anzeige gegraut (keine ASCII-Eingabe mehr möglich) und die nicht druckbaren Zeichen werden als Punkt dargestellt.

Hinweis

TSAP-Länge

TSAPs der Länge 1 oder 2 sollten nicht verwendet werden, da solche TSAPs bereits für interne Zwecke verwendet werden.

Lokale TSAPs und Partner-TSAPs

Partner- und lokale TSAPs können identisch sein, da die Verbindung durch die unterschiedlichen IP-Adressen eindeutig ist. Wenn zwischen zwei Geräten mehr als eine Verbindung eingerichtet werden soll, müssen sich auch die TSAPs unterscheiden.

Default TSAPs

Für die Projektierung der lokalen und Partner-TSAPs gibt es einen Defaultwert ISO-on-TCP-1" für die erste Verbindung zwischen beiden Partnern (änderbar). Bei einer neuen Verbindung zwischen den selben Partnern wird der Defaultwert "ISO-on-TCP-2" vorgeschlagen. Bei einer neuen Verbindung zu einem neuen Partner wird erneut ISO-on-TCP-1 verwendet.

Verbindungsaufbau für FETCH/WRITE

Bezug

Einstellungen zu ISO-Transportverbindungen / ISO-on-TCP Verbindungen / TCP-Verbindungen - Parametergruppe "Eigenschaften > Allgemein > Optionen".

Auswirkungen des Verbindungsaufbaus auf FETCH/WRITE

Beachten Sie für die Betriebsart FETCH/WRITE folgende Hinweise:

- Für S7-Stationen gilt:
  Wenn die Betriebsart FETCH/WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" deaktivieren.
- Für PC-Stationen / OPC-Server gilt:
  Wenn die Betriebsart FETCH/WRITE (siehe Parametergruppe "Optionen") genutzt werden soll, müssen Sie die Option "Aktiver Verbindungsaufbau" aktivieren.

S7-Verbindung

OPC Verbindungsparameter für Zeitüberwachung

Verbindungsaufbau-Timeout

Wird die Anforderung zum Aufbau einer Verbindung vom Kommunikationspartner nicht innerhalb der hier eingestellten Zeit beantwortet, so werden OPC-Aufträge mit einer Fehlermeldung abgebrochen. Es sind Werte von 2.000 bis 100.000 ms einstellbar.

Da je nach Aufbau des Netzes der erfolgreiche Verbindungsaufbau längere Zeit in Anspruch nehmen kann, sollte der Verbindungsaufbau-Timeout größer sein als der Auftragstimeout.

Der Verbindungsaufbau-Timeout startet mit dem Beginn des Verbindungsaufbauwunsches. Dieser Zeitpunkt hängt von der Projektierung des Verbindungsaufbau-Verhaltens ab (s.o.). Es sind zwei Fälle möglich:

- **Verbindung permanent aufrecht erhalten:** Der Verbindungsaufbauwunsch beginnt in diesem Fall mit dem Start des OPC-Servers für das S7-Protokoll. Dieser Start wird ausgelöst durch Verbinden eines OPC-Client auf den OPC-Server und das Durchsuchen des S7-Namensraumes und/oder durch Hinzufügen und aktives Beobachten eines S7-Items.
- **Verbindung bei Bedarf:** Der Verbindungsaufbauwunsch beginnt mit dem ersten Zugriff auf Variablen dieser S7-Verbindung.

Wird die S7-Verbindung im laufenden Betrieb unterbrochen, wird versucht, die Verbindung wieder aufzubauen. Der Verbindungsaufbau-Timeout wird hierbei zurückgesetzt und neu gestartet.

Auftrags-Timeout

Mit dem Auftrags-Timeout werden an das Partnergerät gerichtete Netzaufträge überwacht. Wird ein Netzauftrag innerhalb des Auftrags-Timeout nicht beantwortet, so werden die zugehörenden OPC-Aufträge mit Fehlermeldung abgebrochen. Es sind Werte von 1 bis 100.000 ms einstellbar.

Der Auftrags-Timeout startet zum Zeitpunkt des Absendens eines Netzauftrages im OPC-Server. Es besteht im Allgemeinen nur ein indirekter Zusammenhang mit Schreib-/Lese-/Beobachten-Aufträgen an der OPC-Schnittstelle. Pro Netzauftrag können mehrere OPC-Items behandelt werden, es können ausserdem pro Netzauftrag mehrere OPC-Clients gleiche OPC-Items beauftragen.

Weiterhin spielen die Parameter "Schreibzugriffe optimieren", "Lesezugriffe optimieren" eine Rolle, da dort auf Netzebene mehrere OPC-Aufträge zusammengefasst werden können.

OPC Verbindungsparameter für Verbindungszustand

Sofortige Rückmeldung bei erkannter Verbindungsunterbrechung

Nur aktivierbar, wenn die Option "Die Verbindung permanent aufrecht erhalten" gewählt wurde. Durch die sofortige Rückmeldung über nicht aufgebaute S7-Verbindungen bei OPC Aufträgen vermeiden Sie wiederholte Auftrags-Timeouts.

Funktionsweise: Eine Verbindungsunterbrechung wird erkannt und der Verbindungszustand wird abgespeichert. Falls die Verbindung noch nicht aufgebaut ist oder versucht wird, eine unterbrochene Verbindung wieder aufzubauen, werden OPC Aufträge sofort mit einem entsprechenden Fehlerhinweis beantwortet. Der Fehlerhinweis kommt nicht erst nach Ablauf der Wartezeit für den Auftrags- oder Verbindungs-Timeout.

Schreibzugriffe optimieren, Lesezugriffe optimieren

Prinzip

Für Variablenzugriffe über S7-Verbindungen bietet der OPC-Server für SIMATIC NET einen Optimierungsalgorithmus zur Erhöhung der Geschwindigkeit der Datenübertragung.

Mehrere Zugriffsaufträge auf einzelne Variablen werden intern in einen einzigen Zugriff auf den Kommunikationspartner umgewandelt. So wird die Anzahl der über das Netz transportierten Datenpakete reduziert. Die Auslastung der einzelnen Pakete wird verbessert und der Anteil der Nutzdaten eines Pakets erhöht.

Diese Optimierung gilt sowohl für Lese- als auch für Schreibzugriffe und ist standardmäßig aktiviert. Für den OPC-Client, z. B. ein OPC-fähiges Bedien- oder –Beobachtungssystem, ist dieser Optimierungsalgorithmus unsichtbar.

Weitere Informationen

Weitere Details entnehmen Sie der Dokumentation für den OPC-Server:

- Industrielle Kommunikation mit dem PG/PC Band 1 unter der Beitrags-ID: 42783968 (http://support.automation.siemens.com/WW/view/de/42783968)
- Industrielle Kommunikation mit dem PG/PC Band 2 unter der Beitrags-ID: 42783660 (http://support.automation.siemens.com/WW/view/de/42783660)

Option einseitige S7-Verbindung

Bedeutung

Wenn das Optionskästchen "Einseitig" aktiviert ist, dann sind nur einseitige Kommunikationsbausteine im Anwenderprogramm einer S7-CPU zur Nutzung dieser Verbindung möglich (Anweisungen "PUT" und "GET"). Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.

Zweiseitige Kommunikationsbausteine ("Bausteinpärchen") sind bei einseitigen Verbindungen nicht einsetzbar; wie "USEND" und "URCV", "BSEND" und "BRCV" oder "USTATUS".

Da i. d. R. die Kommunikationsmöglichkeiten anhand der Endpunkte eindeutig identifiziert werden können, ist das Optionskästchen vorbelegt und ggf. nicht änderbar.

Beispiel 1:

Das Optionskästchen ist gegraut (d. h. nicht änderbar) und aktiviert bei folgender Konfiguration:

- der lokale Verbindungsendpunkt ist eine S7-400-CPU oder eine S7-300-CPU mit ladbarer S7-Kommunikation (über CP) und der Partner-Verbindungsendpunkt ist eine S7-300-CPU ohne ladbare S7-Kommunikation.

Beispiel 2:

Das Optionskästchen ist gegraut (d. h. nicht änderbar) und nicht aktiviert bei folgender Konfiguration:

- der lokale Verbindungsendpunkt und der Partner-Verbindungsendpunkt sind eine S7-400-CPU
  In diesem Fall legt STEP 7 automatisch eine zweiseitige Verbindung an. Es wird sowohl für den lokalen Endpunkt als auch für den Endpunkt des Verbindungspartners eine ID eingetragen (Lokale ID und Partner ID).

OPC-Server - Meldungen

Bausteinbezogene Meldungen und Diagnosemeldungen

Die Optionen wirken als Filter, so dass Sie gezielt die Art der Meldungen bestimmen können, die über die S7-Verbindungen kommen darf.

Bausteinbezogene Meldungen (Alarme) erzeugt der Kommunikationspartner über die Bausteine Alarm_8, Alarm_S oder Alarm_SQ.

Bausteinbezogene Meldungen (Alarme) können gemeldet werden

- als Conditional Events (bedingte Ereignisse)
- als Simple Events (einfache Ereignisse für die Kompatibilität mit Altanwendungen, die i. d. R. keine Conditional Events erwarten).

Falls Sie "Bausteinbezogene Meldungen" anwählen, veranlassen Sie den OPC-Server zur Anmeldung für Alarmmeldungen auf dieser Verbindung. Die Übergabe empfangener Alarme erfolgt über die OPC-Schnittstelle Alarms&Events.

Diagnosemeldungen werden aus der Systemzustandsliste (SZL) des S7-Geräts ermittelt. Noch nicht an den OPC-Client gemeldet Diagnoseereignisse werden ihm gemeldet, wenn er sich für Diagnosemeldungen angemeldet hat. Diagnoseereignisse haben keinen Zustand und sind nicht zu quittieren.

Zu den Funktionen zum Einstellen der Prioritäten für einzelne Alarmmeldungen und Diagnosemeldungen gelangen Sie über die Bereichsnavigation.

Weitere Informationen

Weitere Details entnehmen Sie der Dokumentation für den OPC-Server:

- Industrielle Kommunikation mit dem PG/PC Band 1 unter der Beitrags-ID: 42783968 (http://support.automation.siemens.com/WW/view/de/42783968)
- Industrielle Kommunikation mit dem PG/PC Band 2 unter der Beitrags-ID: 42783660 (http://support.automation.siemens.com/WW/view/de/42783660)

1.6.1.7 IE/PB Link PN IO

PROFINET-Gerätenummer

Bezug

Anzeige und Parametrierung der Eigenschaften für PROFINET-IO-Devices am DP-Mastersystem eines IE/PB LINK PN IO in der Parametergruppe "Eigenschaften > Allgemein > PROFINET-Gerätenummer".

Parameter

Um eine möglichst einfache Zuordenbarkeit zu erreichen, versucht STEP 7 beim Plazieren des DP-Slave am DP-Mastersystem, dieselben Ziffern für die Gerätenummer und die PROFIBUS-Adresse zu vergeben. In der Voreinstellung ist daher die Option "PROFIBUS-Adresse als Gerätenummer verwenden" aktiv, sofern noch keine Gerätenummer vergeben ist, die der aktuellen PROFIBUS-Adresse entspricht.

Hinweis

Eindeutige Gerätenummern

Beachten Sie, dass die am PROFINET IO-System vergebenen Gerätenummern eindeutig sein müssen. Dabei sind die am DP-Mastersystem des IE/PB LINK PN IO angeschlossenen IO-Devices (DP-Slaves) mit zu berücksichtigen.

- **PROFIBUS-Adresse**
  Anzeige der PROFIBUS-Adressen der am DP-Mastersystem angeschlossenen DP-Slaves.
- **Name**
  Anzeige der Gerätenamen der am DP-Mastersystem angeschlossenen DP-Slaves. Hier nicht änderbar.
- **PROFINET-Gerätenummer**
  Anzeige bzw. Auswahl der PROFINET-Gerätenummer der am DP-Mastersystem angeschlossenen DP-Slaves. In der Voreinstellung wird die PROFINET-Gerätenummer aus der PROFIBUS-Adresse übernommen.
  Bei Deaktivierung der Option "PB-Adresse als Gerätenummer verwenden" wird die PROFINET-Gerätenummer änderbar. Der Wertebereich ist abhängig vom verwendeten IO-Controller.
  Wenn Sie nach Festlegung der Gerätenummer erneut die Option "PROFIBUS-Adresse als Gerätenummer verwenden" aktivieren, wird der Wert wieder entsprechend zurückgesetzt.
- **PB-Adresse als Gerätenummer verwenden**
  Bei Deaktivierung der Option wird die PROFINET-Gerätenummer änderbar.
- **Priorisierter Hochlauf**
  Bewirkt eine Beschleunigung des Hochlaufs des PROFINET IO-Device (hier des DP-Slave als PROFINET IO-Device). Die Option ist nur wählbar, wenn der Controller Priorisierten Hochlauf unterstützt.
  Sie können die Option bei max. acht am DP-Mastersystem angeschlossen DP-Slaves wählen.

Betriebsarten des IE/PB LINK PN IO

Projektierung und Anzeige der Betriebsart

Projektierung der Betriebsart des Netzübergangs des IE/PB LINK PN IO und Anzeige der resultierenden Betriebsart für PROFIBUS DP.

- Einstellen der Betriebsart:
  Parametergruppe "Eigenschaften > Allgemein > Betriebsart"
- Anzeige der Betriebsart für PROFIBUS DP:
  Parametergruppe "Eigenschaften > Allgemein > DP/MPI-Schnittstelle > Betriebsart"

Betriebsarten des Netzübergangs

Sie können den Netzübergang IE/PB LINK PN IO in folgenden Betriebsarten einsetzen:

- **Netzübergang als PROFINET IO-Proxy**
  Aus Sicht des PROFINET IO-Controllers an Industrial Ethernet ergibt sich kein Unterschied beim Zugriff auf PROFINET IO-Devices, die an Industrial Ethernet angeschlossen sind und auf PROFIBUS DP-Slaves, die an PROFIBUS DP angeschlossen sind.
  IE/PB LINK PN IO übernimmt hierbei für die an PROFIBUS DP angeschlossenen DP-Slaves die Rolle eines Proxy (DP-Betriebsart "DP-Master" ist aktiv).
- **Netzübergang im Standardbetrieb**
  In dieser Betriebsart bestehen folgende Verwendungsmöglichkeiten
  - PG/OP-Kommunikation
    Die PG/OP-Kommunikation dient zum Laden von Programmen und Konfigurationsdaten, zum Durchführen von Test- und Diagnosefunktionen sowie zum Bedienen und Beobachten (HMI-Systeme) einer Anlage.
  - Parametrieren von Feldgeräten (Datensatz-Routing)
    Sie können das IE/PB LINK PN IO als Router für Datensätze verwenden, die an Feldgeräte (DP-Slaves) gerichtet sind. Damit können Geräte, die nicht direkt am PROFIBUS angeschlossen sind und somit keinen direkten Zugriff auf die Feldgeräte (DP-Slaves) haben, über IE/PB LINK PN IO Datensätze an die Feldgeräte übertragen. Ein Werkzeug, das solche Datensätze zur Parametrierung von Feldgeräten erzeugt, ist beispielsweise SIMATIC PDM (Process Device Manager).
  - Netzübergang zu einem DP-Mastersystem mit Äquidistanz
    Das IE/PB LINK PN IO dient als Netzübergang zwischen Ethernet und den Feldgeräten an einem DP-Mastersystem. Das IE/PB LINK PN IO wird hierbei als aktiver Teilnehmer zusammen mit einem DP-Master an einem äquidistant parametrierten PROFIBUS betrieben.
  - Subnetzübergreifende S7-Verbindungen für HMI-Betrieb
    Das IE/PB LINK PN IO leitet die Kommunikation über S7-Verbindungen weiter. Dieser Dienst wird beispielsweise bei HMI-Anwendungen (PC-Stationen) verwendet.

S7-Routing über IE/PB LINK PN IO V2.1 (6GK1411-5AB00)

In den folgenden Fällen ist kein Routing über das IE/PB LINK PN IO möglich:

- S7-Routing zwischen zwei CPUs der S7-1500
- S7-Routing von PG-Verbindungen zu CPUs der S7-1200/1500
- S7-Routing von HMI-Verbindungen zu CPUs der S7-1200/1500

Dieses Verhalten betrifft das IE/PB LINK mit Firmware-Version V2.1 (6GK1 411-5AB00).

1.6.2 Identifikationssysteme

1.6.2.1 RFID-Systeme

Bausteine

SIMATIC Ident

Ausführliche Informationen zum Ident-Profil und den Ident-Bausteinen finden Sie im Handbuch "Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme" auf den Seiten des "Industry Online Support (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)".

Ident-Bausteine programmieren

Basis-Bausteine

Read

Der Baustein "Read" liest die Anwenderdaten vom Transponder aus und stellt diese im Puffer "IDENT_DATA" bereit. Die physikalische Adresse und die Länge der Daten, wird über die Parameter "ADDR_TAG" und "LEN_DATA" übergeben. Bei den Readern RF61xR/RF68xR liest der Baustein die Daten aus der Memory-Bank 3 (USER-Bereich) aus. Der eindeutige Zugriff auf einen bestimmten Transponder erfolgt über die optionalen Parameter "EPCID_UID" und "LEN_ID".

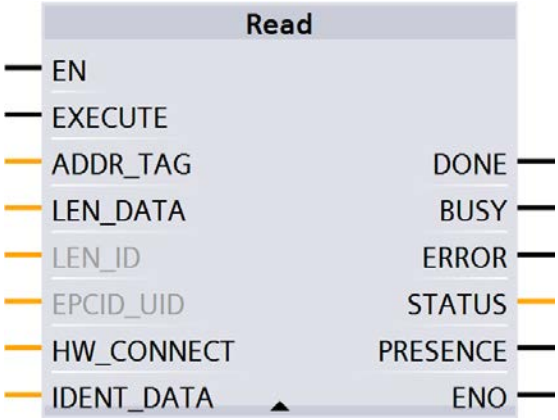

Bild 1-101 Baustein "Read"

Tabelle 1-160 Erläuterung zum Baustein "Read"

Parameter	Datentyp	Default-Werte	Beschreibung
ADDR_TAG	DWord	DW#16#0	Physikalische Adresse auf dem Transponder ab der gelesen wird. Weitere Informationen zur Adressierung finden Sie im Kapitel "Transponder-Adressierung des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)". Bei MV: Ab Adresse "0" steht die Länge des gelesenen Codes (2 Byte). Ab Adresse "2" steht der gelesene Code. [1]
LEN_DATA	Word	W#16#0	Länge der zu lesenden Daten
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF200, RF300, RF61xR, RF68xR)
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
IDENT_DATA	Any / Variant	0	Datenpuffer in dem die gelesenen Daten abgelegt werden. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

[1] Weitere Informationen zur Arbeit mit optischen Lesesystemen, finden Sie in den Betriebsanleitungen "SIMATIC MV420 / SIMATIC MV440" und "SIMATIC MV500".

Read_MV

Der Baustein "Read_MV" liest das Leseergebnis eines optischen Lesegeräts aus. Zum Auslesen der Parametrierung muss der "Read"-Baustein verwendet werden. Die Länge der zu lesenden Daten ermittelt der Baustein automatisch aus der Länge des angelegten Empfangspuffers. Die tatsächliche Länge des Leseergebnisses wird in dem Ausgangsparameter "LEN_DATA" ausgegeben. Die Daten werden im Datenpuffer "IDENT_DATA" abgelegt. Ist der Puffer zu klein, erscheint die Fehlermeldung "0xE7FE0400" und die zu erwartende Länge wird an "LEN_DATA" ausgegeben.

Um eine optimale Geschwindigkeit zu erreichen, empfehlen wir Ihnen, die Länge des Datentyps "IDENT_DATA" so anzupassen, dass diese möglichst nah der maximal erwarteten Länge des Leseergebnisses (2 Byte Codelänge + gelesener Code) entspricht.

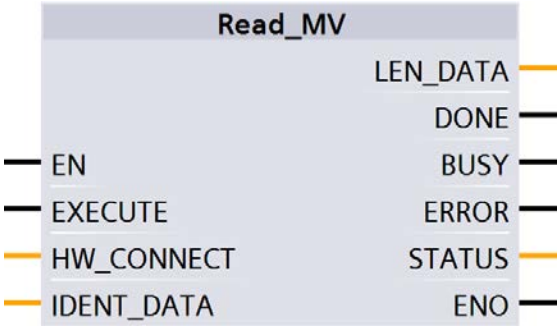

Bild 1-102 Baustein "Read_MV"

Tabelle 1-161 Erläuterung zum Baustein "Read_MV"

Parameter	Datentyp	Default-Werte	Beschreibung
IDENT_DATA	Any / Variant	0	Leseergebnis In Byte 0 und 1 steht die Länge des gelesenen Codes.
LEN_DATA	Word	W#16#0	Länge des Leseergebnis ≙ 2 Byte Codelänge + gelesener Code Hinweis: Bei Variant ist ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Reset_Reader

Mit Hilfe des Bausteins "Reset_Reader" können Sie alle Reader-Typen der Siemens RFID-Systeme zurücksetzen. Dabei werden alle Reader auf die Einstellungen zurückgesetzt, die in der Gerätekonfiguration des RF120C hinterlegt sind bzw. im Reader RF61xR/RF68xR via WBM konfiguriert wurden. Der Baustein "Reset_Reader" besitzt keine gerätespezifischen Parameter und wird über den Parameter "EXECUTE" ausgeführt.

Beschreibungen weiterer Reset-Bausteine für den Betrieb mit den Kommunikationsmodulen RF180C und ASM 456 oder optischen Lesesystemen finden sie im Kapitel "AUTOHOTSPOT".

Mit dem Baustein "Reset_Reader" und den anderen Reset-Bausteinen können Sie jederzeit jeden laufenden Ident-Baustein unterbrechen. Diese Bausteine werden dann mit "DONE = true" und "ERROR = false" beendet.

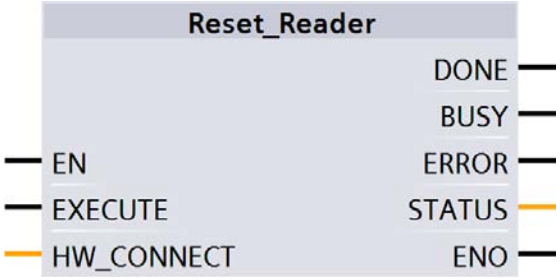

Bild 1-103 Baustein "Reset_Reader"

Set_MV_Program

Mit Hilfe des Bausteins "Set_MV_Program" können Sie das Programm in einer Kamera wechseln. Über den Parameter "PROGRAM" wird die gewünschte Programmnummer übergeben.

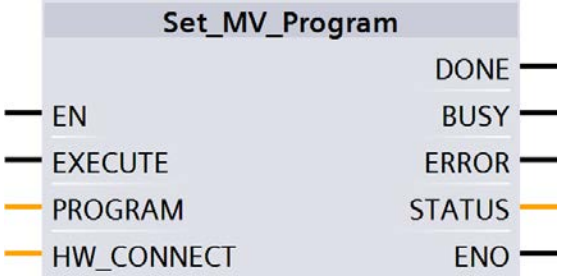

Bild 1-104 Baustein "Set_MV_Program"

Tabelle 1-162 Erläuterung zum Baustein "Set_MV_Program"

Parameter	Datentyp	Default-Werte	Beschreibung
PROGRAM	Byte	B#16#1	Programmnummer Wertebereich: 0x01 ... 0x0F

Write

Der Baustein "Write" schreibt die Anwenderdaten aus dem Puffer "IDENT_DATA" auf den Transponder. Die physikalische Adresse und die Länge der Daten, wird über die Parameter "ADDR_TAG" und "LEN_DATA" übergeben. Bei den Readern RF61xR/RF68xR schreibt der Baustein die Daten in die Memory-Bank 3 (USER-Bereich). Der eindeutige Zugriff auf einen bestimmten Transponder erfolgt über die optionalen Parameter "EPCID_UID" und "LEN_ID".

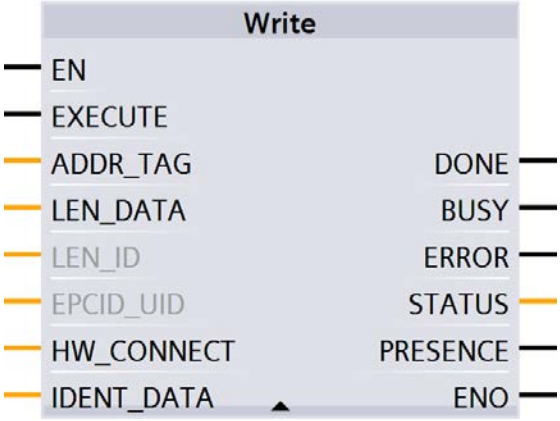

Bild 1-105 Baustein "Write"

Tabelle 1-163 Erläuterung zum Baustein "Write"

Parameter	Datentyp	Default-Werte	Beschreibung
ADDR_TAG	DWord	DW#16#0	Physikalische Adresse auf dem Transponder ab der geschrieben wird. Weitere Informationen zur Adressierung finden Sie im Kapitel "Transponder-Adressierung des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)". Bei MV: Adresse ist immer 0. [1)]
LEN_DATA	Word	W#16#0	Länge der zu schreibenden Daten
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF200, RF300, RF61xR, RF68xR)

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8) Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
IDENT_DATA	Any / Variant	0	Datenpuffer mit den zu schreibenden Daten. Bei MV: Erstes Byte codiert den entsprechenden MV-Befehl.[1)] Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

[1)] Weitere Informationen zur Arbeit mit optischen Lesesystemen, finden Sie in den Betriebsanleitungen "SIMATIC MV420 / SIMATIC MV440" und "SIMATIC MV500".

Erweiterte Bausteine

Config_Upload/-_Download

Mithilfe der Bausteine "Config_Upload" und "Config_Download" können Sie über das Steuerungsprogramm die Konfiguration der Reader RF61xR/RF68xR auslesen ("Config_Upload") oder schreiben ("Config_Download"). Dieser Bausteine ermöglicht es, die Konfigurationen der an dieser Steuerung angeschlossenen Reader auszulesen und zu schreiben.

Die Konfigurationsdaten sind nicht interpretierbare Daten. Speichern Sie die Daten in der Steuerung, um im Falle eines Gerätetauschs diese wieder auf den Reader zu schreiben. Bei gleicher Konfiguration mehrerer Reader müssen Sie die Konfigurationsdaten nur einmal pro Reader-Typ sichern. Eine einzelne Sicherung kann beim Download auf alle gleich konfigurierten Reader kopiert werden.

Die Bytes 6-9 (siehe nachfolgende Tabelle) beinhalten eine Konfigurations-ID (CONFIG_ID) mit einer eindeutigen Versionskennung. Mit der Konfigurations-ID können Sie überprüfen, ob bei einem "Config_Upload" die gelesenen Konfigurationsdaten mit den in der Steuerung gespeicherten Konfigurationsdaten übereinstimmen. Außerdem können Sie mit Hilfe des Geräte-Status die Seriennummer überprüfen, um festzustellen, ob das Gerät getauscht wurde.

Die Konfigurationsdaten sind wie folgt aufgebaut:

Tabelle 1-164 Aufbau der Konfigurationsdaten

Byte	Name
0	Strukturkennung (2 Byte)
2	Längenangabe (4 Byte) Länge von Versionskennung und Parameterblock
6	Versionskennung (≙ CONFIG_ID; 4 Byte) Mit Hilfe der Kennung können Sie die Konfiguration eindeutig identifizieren. Es handelt sich dabei um einen Zeitstempel im Linux-Format. Der Zeitstempel gibt an, wie viele Sekunden seit dem 1. Januar 1979, 00:00 Uhr vergangen sind. Die Kennung wird beim Erzeugen einer Konfiguration vergeben.
10 ... Ende "DATA"	Parameterblock

Der "Config_Upload/Config_Download" kann an jedem Kanal des RF61xR/RF68xR durchgeführt werden. Es werden immer dieselben Konfigurationsdaten übertragen.

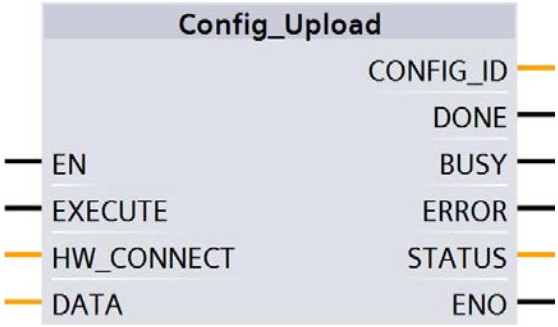

Bild 1-106 Baustein "Config_Upload"

Tabelle 1-165 Erläuterung zum Baustein "Config_Upload"

Parameter	Datentyp	Beschreibung
DATA	Any / Variant	Datenpuffer für Konfigurationsdaten. Die reale Länge der Daten hängt von der Komplexität der Projektierung und dem Firmware-Ausgabestand des Readers ab. Bei einer Standardprojektierung des Readers RF61xR/RF68xR empfehlen wir eine Speichergröße von 4 KB. Verwenden Sie erweitere Reader-Projektierungen (Filterungen) oder möchten Sie in Zukunft die Projektierung ändern ohne die Speichergröße "DATA" anpassen zu müssen, empfehlen wir Ihnen eine Speichergröße von 8-16 KB. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.
CONFIG_ID	DWord	Versionskennung (4 Byte) Mit Hilfe der Kennung können Sie die Konfiguration eindeutig identifizieren. Es handelt sich dabei um einen Zeitstempel im Linux-Format. Der Zeitstempel gibt an, wie viele Sekunden seit dem 1. Januar 1979, 00:00 Uhr vergangen sind. Die Kennung wird beim Erzeugen einer Konfiguration vergeben.

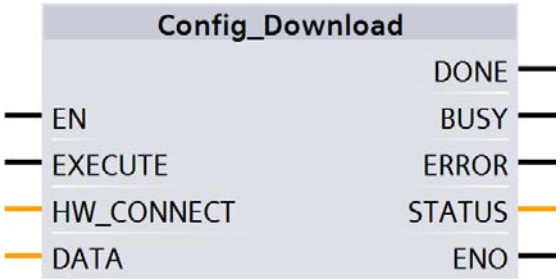

Bild 1-107 Baustein "Config_Download"

Tabelle 1-166 Erläuterung zum Baustein "Config_Download"

Parameter	Datentyp	Beschreibung
DATA	Any / Variant	Datenpuffer für Konfigurationsdaten. Die reale Länge der Daten hängt von der Komplexität der Projektierung und dem Firmware-Ausgabestand des Readers ab. Bei einer Standardprojektierung des Readers RF61xR/RF68xR empfehlen wir eine Speichergröße von 4 KB. Verwenden Sie erweitere Reader-Projektierungen (Filterungen) oder möchten Sie in Zukunft die Projektierung ändern ohne die Speichergröße "DATA" anpassen zu müssen, empfehlen wir Ihnen eine Speichergröße von 8-16 KB. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Inventory

Der Baustein "Inventory" aktiviert die Ausführung von Inventories. Bei einem einzelnen Inventory werden Erfassungszyklen für alle Antennen mit jeweils allen Polarisationen abgefragt. Sind an einem Reader z. B. 2 Antennen mit jeweils 3 Polarisationen angeschlossen, dann beinhaltet ein Inventory 6 Erfassungszyklen.

Bei den Readern RF620R und RF630R werden immer Inventories durchgeführt, sobald die Antenne eingeschaltet ist.

Besonderheit der Reader RF610R, RF615R, RF680R, RF685R

Beachten Sie, dass die Länge des Datenpuffers ("IDENT_DATA") mindestens der Länge der maximal zu erwartenden Daten entsprechen muss. Werden mehr Transponder erkannt und Daten ausgelesen, als in der zugewiesenen Pufferlänge von "IDENT_DATA" Platz haben, gehen die Daten dieser Transponder verloren. Dieses Verhalten wird durch den Fehler "0xE7FE0400" (Pufferüberlauf) angezeigt.

Für die Reader RF61xR/RF68xR stehen zusätzlich die Parameter "DURATION" und "DUR_UNIT" zur Verfügung. Mit Hilfe der Paramater können Sie die Dauer der Inventories festlegen.

Es gibt bei den Readern vier verschiedene Modi, welche Sie über den Parameter "ATTRIBUTE" auswählen können.

- Beim Start wird eine bestimmte Dauer/Anzahl (Zeitdauer, Anzahl Inventories, Anzahl "Observed"-Events bzw. erkannte Transponder) angegeben. Dabei wird zwischen folgenden drei Optionen unterschieden:
  - Dauer
    Inventories über eine festgelegte Zeitdauer durchführen
  - Anzahl Inventories
    Eine festgelegte Anzahl an Inventories durchführen
  - Anzahl "Observed"-Events
    Solange Inventories durchführen, bis eine festgelegte Anzahl Transponder gleichzeitig erkannt wurde.

  Für diese Dauer/Anzahl werden dann Inventories vom Reader durchgeführt. Ist die angegebene Zeit/Anzahl erreicht, wird der Baustein beendet und er liefert alle erkannten Transponder in "IDENT_DATA" zurück. D. h. weitere Befehle können erst dann ausgeführt werden, wenn alle Inventories vollständig durchgeführt wurden. Die Einheit (Zeit oder Anzahl) wird über "DUR_UNIT" und der Wert (Zeitwert oder Anzahl) wird über "DURATION" angegeben. Dieser Modus kann über die Attribute "0x80" und "0x81" ausgeführt werden. Bei dem jeweiligen Attribut werden mehr oder weniger Daten über die erkannten Transponder geliefert.
- Mit den Attributen "0x86" ("Presence_Mode" starten) und "0x87" ("Presence_Mode" beenden) können Inventories dauerhaft ausgeführt werden. Die Anwesenheit eines Transponders kann dann immer über "PRESENCE" abgefragt werden, ohne dass der Baustein mit "EXECUTE" gestartet werden muss. Es werden keine Informationen über die erkannten Transponder bei Ausführung des Befehls zurückgeliefert!
  Um Informationen über die erkannten Transponder zu erhalten, führen Sie einen der ersten beiden oben genannten Aufrufe durch (mit Zeit / Anzahl Inventories = 0).
  Wenn dieser Modus aktiv ist, werden Transponder betreffende Befehle nicht sofort ausgeführt, sondern erst, wenn ein Transponder erfasst wird. Dadurch werden kürzere Reaktionszeiten ermöglicht, da der Befehl bereits ansteht wenn der Transponder ins Antennenfeld kommt.
  Der "Presence_Mode" ist sinnvoll im Rahmen der "Repeat Command"-Funktion.

An dem Ausgangsparameter "NUMBER_TAGS" wird die Anzahl der erkannten Transponder ausgegeben. Bei den Attributen "0x80" und "0x81" wird bei Abschluss des Lesevorgangs die Summe aller erkannten Transponder angezeigt. Bei dem Attribut "0x86" wird an dem Ausgangsparameter "NUMBER_TAGS" immer die Anzahl der aktuell erkannten Transponder angezeigt (max. 15), ohne dass der Baustein mit "EXECUTE" gestartet werden muss.

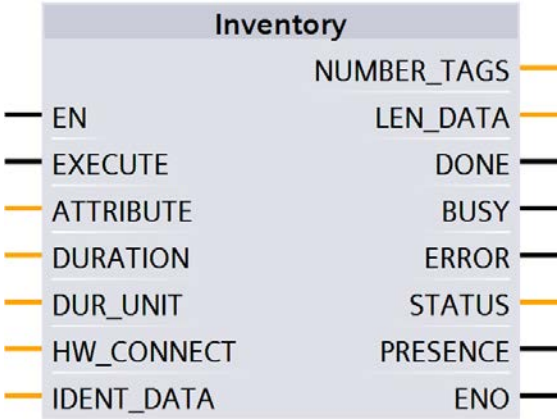

Bild 1-108 Baustein "Inventory"

Tabelle 1-167 Erläuterung zum Baustein "Inventory"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#0	Auswahl des Status-Modus: • MOBY U: 0x00 • RF620R, RF630R: 0x82 (Folgedatensatz auslesen), 0x83, 0x85, 0x90, 0x91, 0x92 • RF610R, RF615R, RF680R, RF685R: 0x80, 0x81, 0x86, 0x87
DURATION	Word	W#16#0	RF610R, RF615R, RF680R, RF685R: Zeitdauer abhängig von "DUR_UNIT" Zeitdauer oder Anzahl Inventories oder Anzahl der "Observed"-Events Bsp.: • 0x00 ≙ kein Inventory • 0x01 ≙ ein Inventory
DUR_UNIT	Word	W#16#0	RF610R, RF615R, RF680R, RF685R: Einheit für "DURATION" • 0x00 ≙ Zeit [ms] • 0x01 ≙ Inventories • 0x02 ≙ Anzahl der Transponder die den Zustand "Observed" erreichen
IDENT_DATA	Any / Variant	0	Datenpuffer für Inventory-Daten Hinweis: Bei Variant ist ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.
NUMBER_TAGS	Int	0	Anzahl der Transponder im Antennenfeld
LEN_DATA	Word	W#16#0	Länge der gültigen Daten Während der Befehlsbearbeitung = 0

Ergebniss-Anzeige

Die nachfolgend genannten Datentypen werden im TIA Portal bei einem neu erstellten Projekt nicht in der Auswahlliste der Datentypen angezeigt. Um diese Datentypen verwenden zu können, müssen Sie den Name des Datentyps (z. B. "IID_IN_I_81") per Hand eintragen. Nach der ersten Verwendung wird der Datentyp automatisch im Projekt und auch in der Auswahlliste aufgeführt.

Ergebnisse für MOBY U

Tabelle 1-168 ATTRIBUTE "0x00" (Datentyp "IID_INVENT_00_MOBY_U")

Name	Typ	Kommentar
number_MDS	WORD	Number of MDS
UID_length	WORD	length of UID
UID	ARRAY[1..12] of IID_IN_I_8BYTE	
UID[1]	IID_IN_I_8BYTE	
UID	ARRAY[1..8] of BYTE	
UID[1]	BYTE	
UID[2]	BYTE	
UID[3]	BYTE	
UID[4]	BYTE	
UID[5]	BYTE	
UID[6]	BYTE	
UID[7]	BYTE	
UID[8]	BYTE	
UID[2]	"IID_IN_I_8BYTE"	
UID[3]	"IID_IN_I_8BYTE"	
UID[4]	"IID_IN_I_8BYTE"	
UID[5]	"IID_IN_I_8BYTE"	
UID[6]	"IID_IN_I_8BYTE"	
UID[7]	"IID_IN_I_8BYTE"	
UID[8]	"IID_IN_I_8BYTE"	
UID[9]	"IID_IN_I_8BYTE"	
UID[10]	"IID_IN_I_8BYTE"	
UID[11]	"IID_IN_I_8BYTE"	
UID[12]	"IID_IN_I_8BYTE"	

Ergebnisse für RF620R, RF630R

Tabelle 1-169 ATTRIBUTE "0x83" (Datentyp "IID_INVENT_82_83_RF600") für RF620R, RF630R mit EPC-ID/UID

Name	Typ	Kommentar
reserved0	BYTE	
number_MDS	BYTE	Number of MDS
EPC	ARRAY[1..19] of "IID_IN_I_12BYTE"	
EPC[1]	"IID_IN_I_12BYTE"	
ID	ARRAY[1..12] of BYTE	
ID[1]	BYTE	
ID[2]	BYTE	
ID[3]	BYTE	
ID[4]	BYTE	
ID[5]	BYTE	
ID[6]	BYTE	
ID[7]	BYTE	
ID[8]	BYTE	
ID[9]	BYTE	
ID[10]	BYT	
ID[11]	BYTE	
ID[12]	BYTE	
EPC[2]	"IID_IN_I_12BYTE"	
EPC[3]	"IID_IN_I_12BYTE"	
EPC[4]	"IID_IN_I_12BYTE"	
EPC[5]	"IID_IN_I_12BYTE"	
EPC[6]	"IID_IN_I_12BYTE"	
EPC[7]	"IID_IN_I_12BYTE"	
EPC[8]	"IID_IN_I_12BYTE"	
EPC[9]	"IID_IN_I_12BYTE"	
EPC[10]	"IID_IN_I_12BYTE"	
EPC[11]	"IID_IN_I_12BYTE"	
EPC[12]	"IID_IN_I_12BYTE"	
EPC[13]	"IID_IN_I_12BYTE"	
EPC[14]	"IID_IN_I_12BYTE"	
EPC[15]	"IID_IN_I_12BYTE"	
EPC[16]	"IID_IN_I_12BYTE"	
EPC[17]	"IID_IN_I_12BYTE"	
EPC[18]	"IID_IN_I_12BYTE"	
EPC[19]	"IID_IN_I_12BYTE"	

Hinweis

Anzahl der EPC-IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen EPC-IDs (1 bis 19) an. Um die Handle-IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

Tabelle 1-170 ATTRIBUTE "0x83", "0x90", "0x91" und "0x92" (Datentyp "IID_INVENT_8x_9x_RF6_D") für RF620R, RF630R mit Handle-ID

Name	Typ	Kommentar
reserved	BYTE	
number_MDS	BYTE	Number of MDS
UID	ARRAY[1..29] of "IID_IN_I_8BYTE"	
UID[1]	"IID_IN_I_8BYTE"	
UID	ARRAY[1..8] of BYTE	
UID[1]	BYTE	
UID[2]	BYTE	
UID[3]	BYTE	
UID[4]	BYTE	
UID[5]	BYTE	
UID[6]	BYTE	
UID[7]	BYTE	
UID[8]	BYTE	
UID[2]	"IID_IN_I_8BYTE"	
UID[3]	"IID_IN_I_8BYTE"	
UID[4]	"IID_IN_I_8BYTE"	
UID[5]	"IID_IN_I_8BYTE"	
UID[6]	"IID_IN_I_8BYTE"	
UID[7]	"IID_IN_I_8BYTE"	
UID[8]	"IID_IN_I_8BYTE"	
UID[9]	"IID_IN_I_8BYTE"	
UID[10]	"IID_IN_I_8BYTE"	
UID[11]	"IID_IN_I_8BYTE"	
UID[12]	"IID_IN_I_8BYTE"	
UID[13]	"IID_IN_I_8BYTE"	
UID[14]	"IID_IN_I_8BYTE"	
UID[15]	"IID_IN_I_8BYTE"	
UID[16]	"IID_IN_I_8BYTE"	
UID[17]	"IID_IN_I_8BYTE"	

Name	Typ	Kommentar
UID[18]	"IID_IN_I_8BYTE"	
UID[19]	"IID_IN_I_8BYTE"	
UID[20]	"IID_IN_I_8BYTE"	
UID[21]	"IID_IN_I_8BYTE"	
UID[22]	"IID_IN_I_8BYTE"	
UID[23]	"IID_IN_I_8BYTE"	
UID[24]	"IID_IN_I_8BYTE"	
UID[25]	"IID_IN_I_8BYTE"	
UID[26]	"IID_IN_I_8BYTE"	
UID[27]	"IID_IN_I_8BYTE"	
UID[28]	"IID_IN_I_8BYTE"	
UID[29]	"IID_IN_I_8BYTE"	
reserved1	DWORD	
Data	ARRAY[1..222] of BYTE	

Hinweis

Anzahl der Handle-IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen Handle-IDs (1 bis 29) an. Um die Handle-IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

Tabelle 1-171 ATTRIBUTE "0x85" (Datentyp "IID_INVENT_85_RF600")

Name	Typ	Kommentar
reserved	BYTE	
number_MDS	STRUCT	Number of MDS
ID	BYTE	
ID[1]	BYTE	
Handle	ARRAY[1..8] of BYTE	
Handle[1]	BYTE	
Handle[2]	BYTE	
Handle[3]	BYTE	
Handle[4]	BYTE	
Handle[5]	BYTE	
Handle[6]	BYTE	
Handle[7]	BYTE	
Handle[8]	BYTE	
EPC	ARRAY[1..12] of BYTE	

Name	Typ	Kommentar
EPC[1]	BYTE	
EPC[2]	BYTE	
EPC[3]	BYTE	
EPC[4]	BYTE	
EPC[5]	BYTE	
EPC[6]	BYTE	
EPC[7]	BYTE	
EPC[8]	BYTE	
EPC[9]	BYTE	
EPC[10]	BYTE	
EPC[11]	BYTE	
EPC[12]	BYTE	
ID[2]	"IID_IN_I_20Byte"	
ID[3]	"IID_IN_I_20Byte"	
ID[4]	"IID_IN_I_20Byte"	
ID[5]	"IID_IN_I_20Byte"	
ID[6]	"IID_IN_I_20Byte"	
ID[7]	"IID_IN_I_20Byte"	
ID[8]	"IID_IN_I_20Byte"	
ID[9]	"IID_IN_I_20Byte"	
ID[10]	"IID_IN_I_20Byte"	
ID[11]	"IID_IN_I_20Byte"	

Hinweis

Anzahl der übertragenen IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen IDs (1 bis 11; Handle-IDs und EPC-IDs) an. Um die IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x82 ≙ 0x02
0x83 ≙ 0x03
0x85 ≙ 0x05
0x90 ≙ 0x10
0x91 ≙ 0x11
0x92 ≙ 0x12

Programmierung des ATTRIBUTE "0x82"

Wenn die Anzahl der im Antennenfeld befindlichen Transponder unbekannt ist, wiederholen Sie den "INVENTORY"-Baustein mit Hilfe des ATTRIBUTE = "0x82".

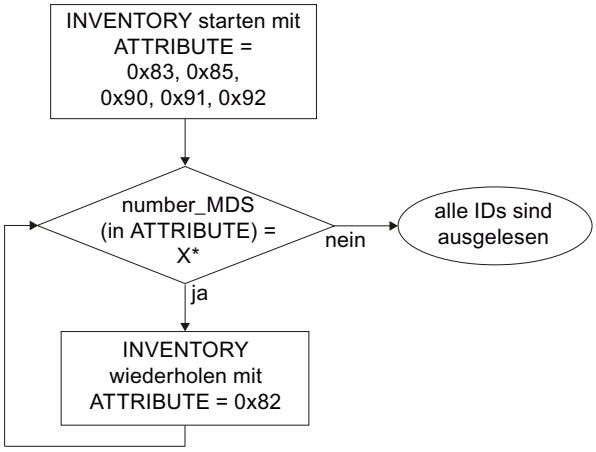

* Die Anzahl der zurückgelieferten IDs "X" ist abhängig vom verwendeten "ATTRIBUTE".

Bild 1-109 Programmablauf des ATTRIBUTE "0x82" bei unbekannten Transponder-Populationen

Ergebnisse für RF610R, RF615R, RF680R, RF685R

Die Anzahl der Elemente "TAG_DATA[x]" der Datentypen der ATTRIBUTE "0x80" und "0x81" ist abhängig von der Anzahl der zu erwartenden Transponder. Aus diesem Grund müssen Sie sich den Empfangspuffer selbst zusammenstellen. Beachten Sie bei der Erstellung des Empfangspuffer "IDENT_DATA"/Datentyps folgenden Aufbau:

- Das erste Element "NUM_MDS" ist immer vom Typ "WORD".
- Das darauf folgende Element "TAG_DATA" ist immer vom Typ "ARRAY". In dem "ARRAY" müssen Sie die Anzahl der zu erwartenden Transponder eintragen ("n").

Die nachfolgenden Tabellen zeigen beispielhaft den Aufbau des Empfangspuffer "IDENT_DATA"/Datentyp für die ATTRIBUTE "0x80" und "0x81".

Tabelle 1-172 ATTRIBUTE "0x80"

Name	Typ	Kommentar
NUM_MDS	WORD	Number of MDS
TAG_DATA	ARRAY[1..n]of IID_IN_I_80	Length of EPC ID
TAG_DATA[1]	IID_IN_I_80	
Reserved	BYTE	
ID_Len	BYTE	Length of EPC ID
EPC_ID	ARRAY[1..62] of BYTE	EPC-ID
tagPC	WORD	
TAG_DATA[2]	IID_IN_I_80	

Name	Typ	Kommentar
...	...	
TAG_DATA[n]	IID_IN_I_80	

Tabelle 1-173 ATTRIBUTE "0x81"

Name	Typ	Kommentar
NUM_MDS	WORD	Number of MDS
TAG_DATA	ARRAY[1..n] of IID_IN_I_81	
TAG_DATA[1]	IID_IN_1_81	
reserved	BYTE	
ID_LEN	BYTE	EPC length
EPC_ID	ARRAY[1..62] of BYTE	EPC-ID
tagPC	WORD	
RSSI	BYTE	RSSI value
MaxRSSI	BYTE	highest RSSI value
MinRSSI	BYTE	lowest RSSI value
channel	BYTE	channel; 1..15_ESTI; 1..53:FCC
antenna	BYTE	antenna; bit coded; Bit 0 = antenna 1; Bit 1 = antenna 2; Bit 2 = antenna 3; Bit 3 = antenna 4
polarization	BYTE	polarizatuin of antenna; 0=undefined; 1=circular; 2=vertical linear; 3=horizontal
time	Time_OF_Day	S7 timestamp
power	BYTE	power in dBm
filterDataAvailable	BYTE	0=false; 1=true
Inventoried	WORD	1)
TAG_DATA[2]	IID_IN_1_81	
...	...	
TAG_DATA[n]	IID_IN_1_81	

1) Angabe, wie oft der Transponder über die Luftschnittstelle erkannt wurde, bis er in den Zustand "Observed" wechselt.

Read_EPC_Mem

Der Baustein "Read_EPC_Mem" liest Daten ab Adresse 4 aus dem EPC-Speicher des RF600-Transponders aus. Der Zugriff erfolgt auf Bank 1 ab Startadresse 4. Über den Parameter "LEN_DATA" wird die auszulesende Länge des EPC-Speichers angegeben.

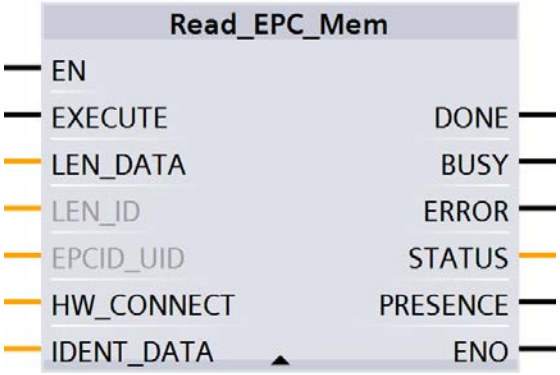

Bild 1-110 Baustein "Read_EPC_Mem"

Tabelle 1-174 Erläuterung zum Baustein "Read_EPC_Mem"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_DATA	Word	W#16#0	Länge des auszulesenden EPC-Speichers (1 ... 62 Byte)
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF61xR, RF68xR)
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
IDENT_DATA	Any / Variant	0	Datenpuffer in den die gelesenen EPC-Speicherdaten abgelegt werden. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Read_TID

Der Baustein "Read_TID" liest Daten aus dem TID-Speicherbereich (Tag Identification Memory Bank) des RF600-Transponders aus. Über den Parameter "LEN_DATA" wird die zu lesende Länge der TID angegeben. Die Länge der TID variiert abhängig vom Transponder und ist dem dazugehörigen Datenblatt zu entnehmen.

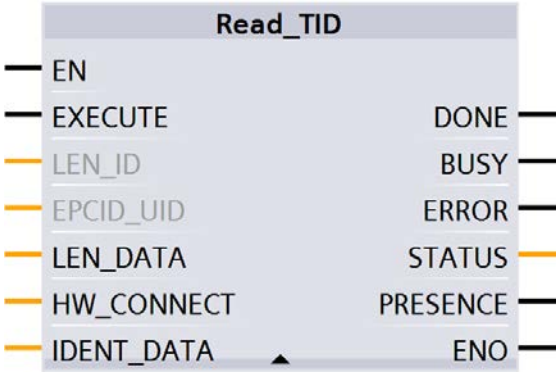

Bild 1-111 Baustein "Read_TID"

Tabelle 1-175 Erläuterung zum Baustein "Read_TID"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF61xR, RF68xR)
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
LEN_DATA	Word	W#16#4	Länge des auszulesenden EPC-Speichers (1 ... 62 Byte)
IDENT_DATA	Any / Variant	0	Gelesene TID Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Read_UID

Der Baustein "Read_UID" liest die UID eines HF-Transponders aus. Die UID hat immer eine feste Länge von 8 Byte.

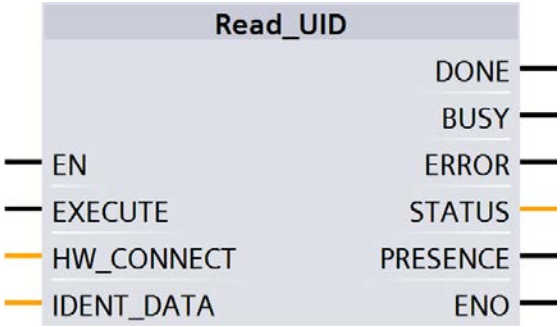

Bild 1-112 Baustein "Read_UID"

Tabelle 1-176 Erläuterung zum Baustein "Read_UID"

Parameter	Datentyp	Beschreibung
IDENT_DATA	Any / Variant	UID Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Set_Ant

Mit Hilfe des Bausteins "Set_Ant" können Sie Antennen aus- oder einschalten. Es gibt unterschiedliche Bausteine für RF300 und RF600. Der Baustein "Set_Ant_RF300" kann auch für RF200, MOBY D und MOBY U verwendet werden. Der Baustein "Set_Ant_RF600" bezieht sich ausschließlich auf die Reader RF620R und RF630R.

Set_Ant_RF300

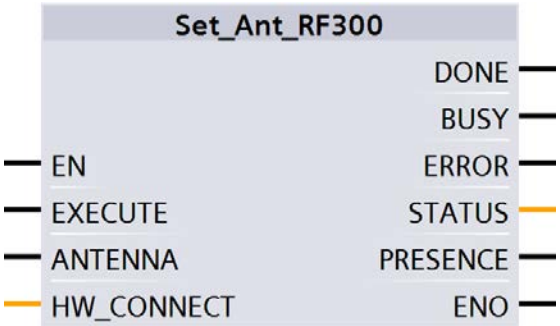

Bild 1-113 Baustein "Set_Ant_RF300"

Tabelle 1-177 Erläuterung zum Baustein "Set_Ant_RF300"

Parameter	Datentyp	Beschreibung
ANTENNA	Bool	0 = Antenne ausschalten 1 = Antenne einschalten

Set_Ant_RF600

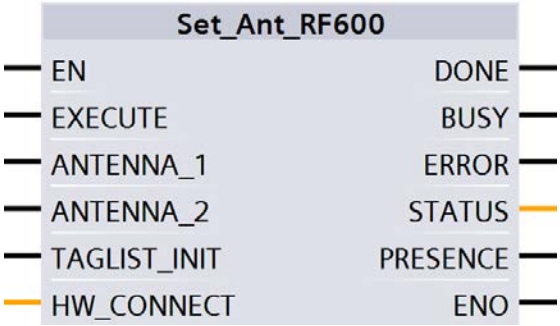

Bild 1-114 Baustein "Set_Ant_RF600"

Tabelle 1-178 Erläuterung zum Baustein "Set_Ant_RF600" (RF620R/RF630R)

Parameter	Datentyp	Beschreibung
ANTENNA_1	Bool	0 = Antenne 1 ausschalten 1 = Antenne 1 einschalten
ANTENNA_2	Bool	0 = Antenne 2 ausschalten 1 = Antenne 2 einschalten
TAGLIST_INIT	Bool	0 = TagList wird zurückgesetzt 1 = Mit vorhandener TagList wird weitergearbeitet

Set_Param

Mit Hilfe des Bausteins "Set_Param" können Sie UHF-Parameter an einem RF61xR/RF68xR während der Laufzeit ändern (z. B. die Antennenleistung).

Hinweis

Einstellungen nur flüchtig gespeichert

Beachten Sie, dass die im Baustein "Set_Param" hinterlegten Parameter nur flüchtig gespeichert werden. Wird die Spannung des Readers unterbrochen, gehen die hinterlegten Werte verloren und müssen erneut gesetzt werden.

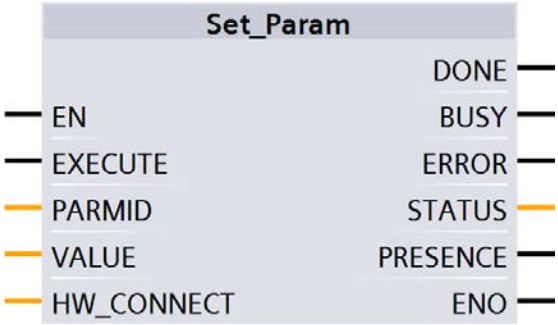

Bild 1-115 Baustein "Set_Param"

Tabelle 1-179 Erläuterung zum Baustein "Set_Param"

Parameter	Datentyp	Default-Werte	Beschreibung
PARMID	DWORD	0x00	Parameterkennung
VALUE	DWORD	0x00	Parameterwert

Tabelle 1-180 Parameterwerte

PARMID (hex)	PARMID (ASCII)	Parameter	VALUE
0x41315057	A1PW	Antenne 01: Strahlungsleistung	Wertebereich: 0,5 ... 33 Schrittweite: 0,25 Strahlungsleistung der Antenne in [dBm]. Byte 1 und 2 sind unbelegt, Byte 3 stellt die Ganzzahl dar und Byte 4 die Nachkommastelle. Beispiel: Eine Strahlungsleistung von 10,25 dBm entspricht einem "VALUE" von "0x0A19".
0x41325057	A2PW	Antenne 02: Strahlungsleistung	
0x41335057	A3PW	Antenne 03: Strahlungsleistung	
0x41345057	A4PW	Antenne 04: Strahlungsleistung	
0x41315452	A1TR	Antenne 01: RSSI-Schwellwert	Wertebereich: 0 ... 255 Schwellwert für RSSI. Transponder mit niedrigeren Werten werden verworfen. Einheitsloser Wert, ohne direkten Bezug zur Strahlungsleistung.
0x41325452	A2TR	Antenne 02: RSSI-Schwellwert	
0x41335452	A3TR	Antenne 03: RSSI-Schwellwert	
0x41345452	A4TR	Antenne 04: RSSI-Schwellwert	
0x5331444C	S1DL	Lesestelle 01: RSSI Delta	Wertebereich: 0 ... 255 Differenz für RSSI-Werte. Transponder mit niedrigeren Werten bezogen auf den Transponder mit dem höchsten RSSI-Wert werden verworfen. Einheitsloser Wert, ohne direkten Bezug zur Strahlungsleistung.
0x5332444C	S2DL	Lesestelle 02: RSSI Delta	
0x5333444C	S3DL	Lesestelle 03: RSSI Delta	
0x5334444C	S4DL	Lesestelle 04: RSSI Delta	

PARMID (hex)	PARMID (ASCII)	Parameter	VALUE
0x4131504F	A1PO	Antenne 01: Polarisation	Wertebereich: 0, 1, 2, 4 Polarisation der Antenne (für intelligente Antennen, z. B. interne Antenne RF685R) • 0: default, undefiniert • 1: zirkular • 2: vertikal linear • 4: horizontal linear Eingabe ist bitkodiert. Kombinationen sind möglich (Werte addieren).
0x4132504F	A2PO	Antenne 02: Polarisation	
0x4133504F	A3PO	Antenne 03: Polarisation	
0x4134504F	A4PO	Antenne 04: Polarisation	
0x52364353	R6CS	Modulationsschema	Wertebereiche: 32, 33, 34, 35, 37, 65 Modulationsschema der Lesestelle Festlegung welche Transponder-Typen erkannt werden (ISO 18000-63/-62). • 32: Tx: 40 kbps / Rx: 40 kbps / FM0 • 33: Tx: 40 kbps / Rx: 62,5 kbps / Miller4 • 34: Tx: 40 kbps / Rx: 75 kbps / Miller4 • 35: Tx: 80 kbps / Rx: 62,5 kbps / Miller4 • 37: Tx: 80 kbps / Rx: 400 kbps / FM0 • 65: Tx: 40 kbps / Rx: 40 kbps / ISO 18000-62
0x57544348	WTCH	Datum und Uhrzeit	Wertebereich: 01.01.2000 00:00 Uhr ... 19.01.2038 3:14 Uhr 01.01.2000 01:00 Uhr ≙ 946684800 Datum und Uhrzeit (UTC) Zeit in Sekunden seit 01.01.1970; Setzen der internen Reader-Uhr, wobei Datum und Uhrzeit gesetzt wird.

PARMID (hex)	PARMID (ASCII)	Parameter	VALUE
0x57544F44	WTOD	Uhrzeit	Wertebereich: 0:00 ... 23:59 Uhr S7-Uhrzeit (TOD, UTC) Millisekunden seit Mitternacht; Setzen der internen Reader-Uhr, wobei nur die Uhrzeit verändert wird und nicht das Datum.
0x57444154	WDAT	Datum	Wertebereich: 01.01.2000 ... 18.01.2038 S7-Datum Tage seit 01.01.1990; Setzen der internen Reader-Uhr, wobei nur das Datum verändert wird und nicht die Uhrzeit.

Write_EPC_ID

Der Baustein "Write_EPC_ID" überschreibt die EPC-ID des RF600-Transponders und passt die Länge der EPC-ID im Speicher des Transponders an. Über den Parameter "LEN_ID_NEW" wird die zu schreibende neue EPC-ID-Länge und über die Parameter "LEN_ID" und "EPCID_UID" wird die bisherige EPC-ID angegeben.

Beachten Sie, dass sich beim Ausführen des Bausteins nur ein Transponder im Antennenfeld befindet. Dadurch wird sichergestellt, dass die Identifizierung beim Schreiben der ID eindeutig ist. Befinden sich mehrere Transponder im Antennenfeld, wird eine negative Antwort zurückgegeben.

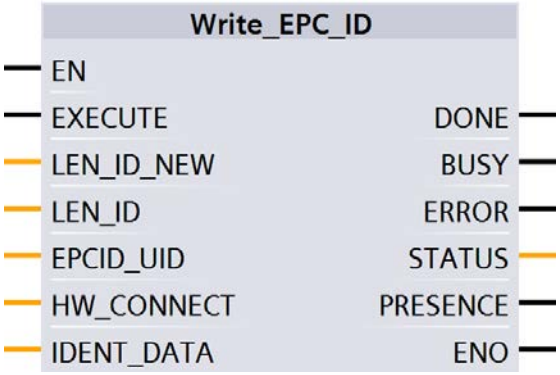

Bild 1-116 Baustein "Write_EPC_ID"

Tabelle 1-181 Erläuterung zum Baustein "Write_EPC_ID"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_ID_NEW	Byte	W#16#0C	Länge der aktuellen EPC-ID
LEN_ID	Byte	B#16#0	Länge der bisherigen EPC-ID

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[1...62] of Byte	0	Bisherige EPC-ID
IDENT_DATA	Any / Variant	0	Aktuelle EPC-ID Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Write_EPC_Mem

Der Baustein "Write_EPC_Mem" überschreibt den EPC-Speicher ab Adresse 4 des RF600-Transponders. Über den Parameter "LEN_DATA" wird die zu überschreibende Länge des EPC-Speichers angegeben.

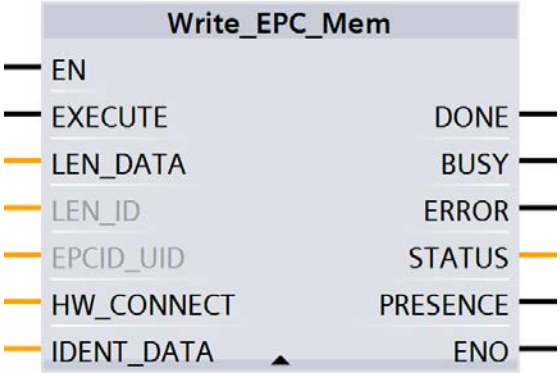

Bild 1-117 Baustein "Write_EPC_Mem"

Tabelle 1-182 Erläuterung zum Baustein "Write_EPC_Mem"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_DATA	Word	W#16#0	Länge des zu überschreibenden EPC-Speichers (1 ... 62 Byte)
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF61xR, RF68xR)

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
IDENT_DATA	Any / Variant	0	Datenpuffer mit den zu überschreibenden EPC-Speicherdaten. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

AdvancedCMD

Mit dem Baustein "AdvancedCmd" kann jeder Befehl ausgeführt werden, auch Befehle die durch die anderen Bausteine nicht dargestellt werden. Dieser allgemeine Aufbau kann für alle Befehle verwendet werden und ist ausschließlich für geschulte Anwender konzipiert.

Dieser Baustein ermöglicht es Ihnen optional den Befehl als verketteter Befehl zu versenden. Dafür stellt der Baustein einen CMD-Puffer für 100 Befehle zur Verfügung. Alle verketteten Befehle müssen ab dem ersten Platz im Puffer eingetragen werden. Zusätzlich muss für jeden verketteten Befehl das "Chained-Bit" in der CMD-Struktur gesetzt werden. Ausschließlich bei dem letzten Befehl in der Kette wird das "Chained-Bit" nicht gesetzt. Weitere Informationen zum "Chained-Bit" finden Sie im Kapitel "Verkettung des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)".

Im Eingangsparameter "CMD" muss die komplette Befehlsstruktur angegeben werden. Die Struktur für den Parameter "CMD" müssen Sie in einem Datenbaustein anlegen.

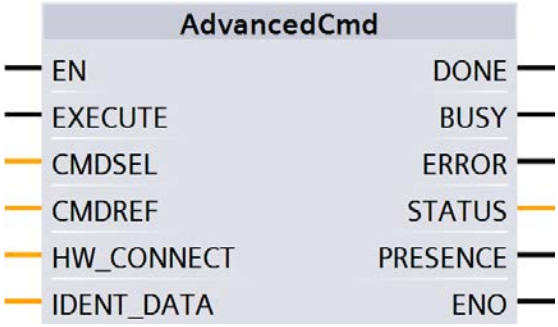

Bild 1-118 Baustein "AdvancedCmd"

Tabelle 1-183 Erläuterung zum Baustein "AdvancedCMD"

Parameter	Datentyp	Default-Werte	Beschreibung
CMDSEL	Int	1	Auswahl des auszuführenden Befehls "CMDREF"; 1 ⇒ 1. Befehl, ... Der Wert des Parameters "CMDSEL" kann nie > 100 sein.
CMDREF	Any / Variant	--	Eine ausführliche Beschreibung des Parameters finden Sie in den Kapiteln: • "Befehlsübersicht des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)" • "Befehlsstruktur des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)"
IDENT_DATA	Any / Variant	0	Puffer für zu schreibende bzw. gelesene Daten. Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Reset-Bausteine

Die in diesem Kapitel beschriebenen Reset-Bausteine benötigen Sie, wenn Sie die Optischen Lesesysteme MV420, MV440 oder die Kommunikationsmodule RF180C, ASM 456 an einer Steuerung SIMATIC S7-1200/-1500 betreiben möchten. Alternativ können Sie diese Bausteine auch für die RF120C verwenden, wenn Sie in der Gerätekonfiguration die entsprechende Einstellung parametriert haben.

Diese Reset-Bausteine haben im System die gleiche Funktion wie der weiter vorne beschriebene Baustein "Reset_Reader". Allerdings müssen Sie bei den hier beschriebenen Bausteinen Reader-abhängige Parameter einstellen.

Beachten Sie, dass bei den Parametern automatisch der angegebene Default-Wert verwendet wird, wenn Sie keinen Wert manuell auswählen.

Reset_MOBY_D

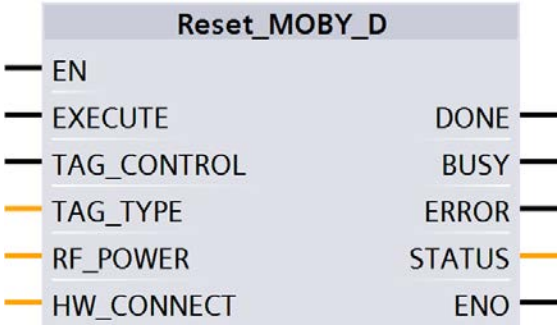

Bild 1-119 Baustein "Reset_MOBY_D"

Tabelle 1-184 Erläuterung zum Baustein "Reset_MOBY_D"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
TAG_TYPE	Byte	1	Transponder-Typ: • 1 = jeder ISO-Transponder
RF_POWER	Byte	0	Ausgangsleistung HF-Leistung von 0,5 W bis 10 W in 0,25 W-Schritten (Wertebereich: 0x02 ... 0x28)

Reset_MOBY_U

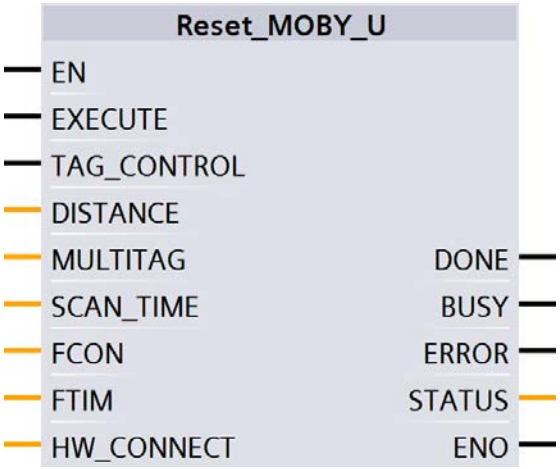

Bild 1-120 Baustein "Reset_MOBY_U"

Tabelle 1-185 Erläuterung zum Baustein "Reset_MOBY_U"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
DISTANCE	Byte	23h	Reichweitenbegrenzung (Wertebereich: 0x02 ... 0x23 oder 0x82 ... 0xA3 für verminderte Sendeleistung)
MULTITAG	Byte	1	Maximale Anzahl der parallel im Antennenfeld bearbeitbaren Transponder. (Wertebereich: 0x01 ... 0x12)
SCAN_TIME	Byte	0	Scanning_Time: Stand-by-Zeit des Transponders (Wertebereich: 0x00 ... 0xC8)
FCON	Byte	0	field_ON_control: Bero-Betriebsart (Wertebereich: 0x00 ... 0x03)
FTIM	Byte	0	field_ON_time: Zeit für Bero-Betriebsart (Wertebereich: 0x00 ... 0xFF)

Reset_MV

Um Kameras der Optischen Lesesysteme zurückzusetzen, rufen Sie den Baustein auf und führen Sie den Parameter "EXECUTE" aus.

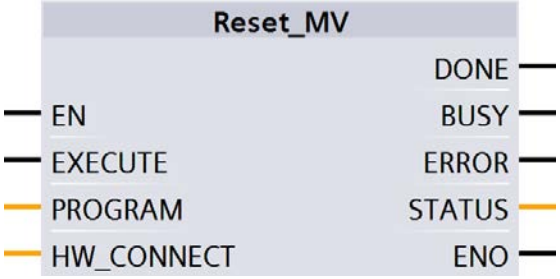

Bild 1-121 Baustein "Reset_MV"

Tabelle 1-186 Erläuterung zum Baustein "Reset_MV"

Parameter	Datentyp	Beschreibung
PROGRAM	Byte	Programmanwahl • B#16#0: Reset ohne Programmanwahl bzw. im Falle einer Diagnose wird der Fehlercode für "IN_OP = 0" am Ausgangsparameter "STATUS" angezeigt. • B#16#1 ... B#16#0F: Nummer des zu startenden Programms ⇒ Reset mit Programmanwahl (ab FW V5.1 der MV4x0)

Reset_RF200

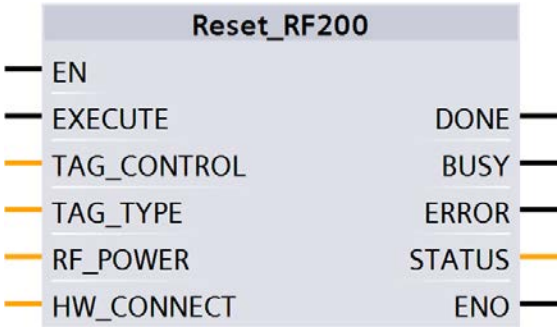

Bild 1-122 Baustein "Reset_RF200"

Tabelle 1-187 Erläuterung zum Baustein "Reset_RF200"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Byte	1	Anwesenheitskontrolle
TAG_TYPE	Byte	1	Transponder-Typ: • 1 = jeder ISO-Transponder • 3 = MDS D3xx - Optimierung
RF_POWER	Byte	4	Ausgangsleistung; nur für RF290R relevant HF-Leistung von 0,5 W bis 5 W in 0,25 W-Schritten (Wertebereich: 0x02 ... 0x14). Default-Wert 0x04 ≙ 1 W.

Reset_RF300

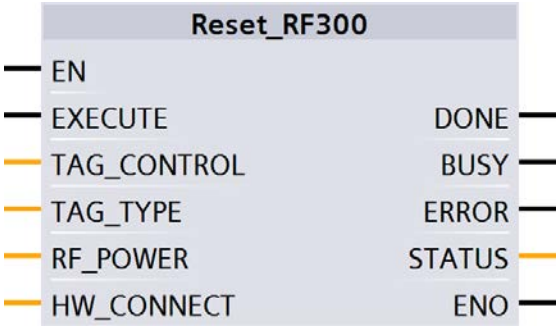

Bild 1-123 Baustein "Reset_RF300"

Tabelle 1-188 Erläuterung zum Baustein "Reset_RF300"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Byte	1	Anwesenheitskontrolle • 0 = Aus • 1 = Ein • 4 = Anwesenheit (Antenne ist aus. Antenne wird nur eingeschaltet, wenn ein Read- oder Write-Befehl geschickt wird.)
TAG_TYPE	Byte	0	Transponder-Typ: • 1 = jeder ISO-Transponder • 0 = RF300-Transponder
RF_POWER	Byte	0	Ausgangsleistung; nur für RF380R relevant HF-Leistung von 0,5 W bis 2 W in 0,25 W-Schritten (Wertebereich: 0x02 ... 0x08). Default-Wert 0x00 ≙ 1,25 W.

Reset_RF600

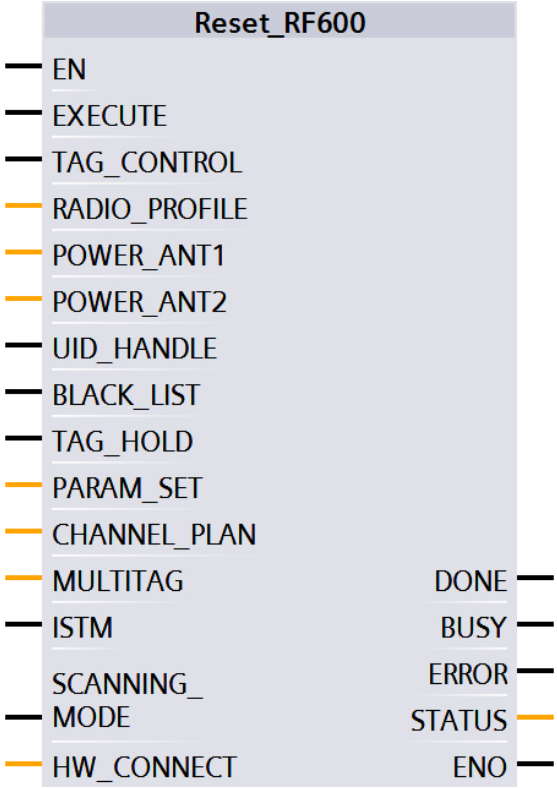

Bild 1-124 Baustein "Reset_RF600"

Tabelle 1-189 Erläuterung zum Baustein "Reset_RF600" (RF620R/RF630R)

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
RADIO_PROFILE	Byte	1	Scanning_Time: Funkprofil nach EPC-Global (Wertebereich: 0x01 ... 0x09, abhängig von der Reader-Variante)
POWER_ANT1	Byte	0	Sendeleistung für Antenne 1 bzw. interne Antenne (Wertebereich: 0x00 ... 0x0F)
POWER_ANT2	Byte	0	Sendeleistung für Antenne 2 bzw. externe Antenne (Wertebereich: 0x00 ... 0x0F)
UID_HANDLE	Bool	False	Bedeutung der UID im Befehl: True = Handle-ID, nur die niederwertigsten 4 Bytes der UID werden ausgewertet; False = UID-/EPC-ID mit 8 Byte Länge
BLACK_LIST	Bool	False	True = Black List aktivieren
TAG_HOLD	Bool	False	True = Tag Hold aktivieren
PARAM_SET	Byte	0	Field_ON_Control (0 = fast; Wertebereich: 0x00, 0x02)
CHANNEL_PLAN	Byte	0F	Field_ON_Time (Wertebereich: 0x00 ... 0x0F; nur bei ETSI)
MULTITAG	Byte	1	Maximale Anzahl der parallel im Antennenfeld bearbeitbaren Transponder. (Wertebereich: 0x01 ... 0x50)
ISTM	Bool	False	True = Intelligenten Singletag-Modus aktivieren
SCANNING_MODE	Bool	False	True = Scanning Mode aktivieren [1)]

[1)] Ist aktuell mit den Ident-Bausteinen nicht möglich.

Reset_Univ

Der Baustein "Reset_Univ" ist ein universeller Reset-Baustein, mit dem Identifikationssysteme zurückgesetzt werden können. Verwenden Sie diesen Baustein nur in Absprache mit dem Support.

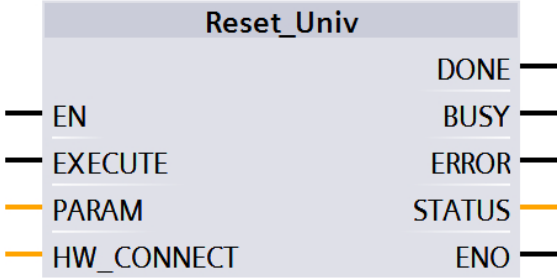

Bild 1-125 Baustein "Reset_Univ"

Tabelle 1-190 Erläuterung zum Baustein "Reset_Univ"

Parameter	Datentyp	Beschreibung
PARAM	Array [1...16] of Byte	Daten für Reset-Telegramm Die hier einzustellenden Daten können für Spezialeinstellungen bei Bedarf vom Support bereitgestellt werden.

Tabelle 1-191 Aufbau des Parameters "PARAM"

Byte	1	2...5	6	7...8	9	10	11	12	13...14	15	16
Wert	04h	0	0Ah	0	scan-ning_ time	param	opti-on_ 1	dis-tance_ limiting	Anzahl der Trans-ponder	field_ on_ control	field_ on_ time

Status-Bausteine

Reader_Status

Der Baustein "Reader_Status" liest Statusinformationen aus dem Reader aus. Für die verschiedenen Reader-Familien gibt es unterschiedliche Status-Modi, die Sie über den Parameter "ATTRIBUTE" auswählen können.

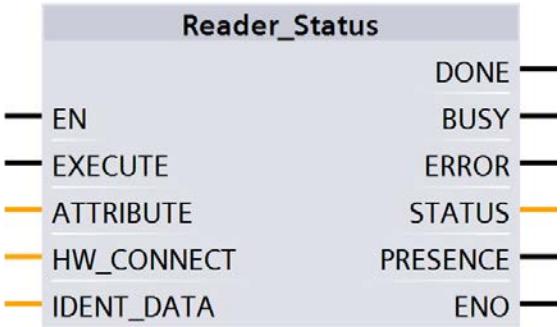

Bild 1-126 Baustein "Reader_Status"

Tabelle 1-192 Erläuterung zum Baustein "Reader_Status"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#81	Kennung der Status-Modi / Mögliche Eingaben: • RF200: 0x81 • RF300: 0x81, 0x86 • RF620R, RF630R: 0x87, 0x88, 0xA0, 0xA1 • RF61xR, RF68xR: 0x89 • RF185C, RF186C, RF188C: 0xA2 • MOBY U: 0x81, 0x84, 0x85 • MOBY D: 0x81
IDENT_DATA	Any / Variant	0	Ergebniswerte je nach Attribute Hinweis: Sie können entweder einen "Array_of_Byte" oder eine Variable mit einem der nachfolgenden Datentypen anlegen. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden

Ergebnisse

Legen Sie den jeweils richtigen Datentyp, der dem ATTRIBUTE-Wert zugeordnet ist, am Eingang "IDENT_DATA" des Bausteins an, damit die Daten richtig interpretiert werden können.

Beachten Sie, dass die UDTs erst dann verwendet werden können, wenn die Bausteine "Reader_Status" oder "Tag_Status" verwendet werden.

Tabelle 1-193 ATTRIBUTE "0x81" (Datentyp "IID_READSTAT_81_RF2_3_U")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode
hardware	CHAR	Type of hardware
hardware_version	WORD	Version of hardware
loader_version	WORD	Version of loader
firmware	CHAR	Type of firmware
firmware_version_HB	BYTE	Version of firmware
firmware_version_LB	BYTE	
driver	CHAR	Type of driver
driver_version	WORD	Version of driver
interface	BYTE	Type of interface (RS 232/RS 422)
baud	BYTE	Baudrate
reserved1	BYTE	Reserved
reserved2	BYTE	Reserved
reserved3	BYTE	Reserved
distance_limiting_SLG	BYTE	Distance limiting of SLG

Name	Typ	Kommentar
multitag_SLG	BYTE	Multitag SLG
field_ON_control_SLG	BYTE	Field ON control
field_ON_time_SLG	BYTE	Field On time
sync_SLG	BYTE	Synchronization with SLG
status_ant	BYTE	Status of antenne
stand_by	BYTE	Time of standby after command
MDS_control	BYTE	Presence mode

Tabelle 1-194 ATTRIBUTE "0x84" (Datenytp "IID_READSTAT_84_MOBY_U")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode
number_MDS	BYTE	Range 1..24
UID	ARRAY [1..24] of DWord	

Tabelle 1-195 ATTRIBUTE "0x86" (Datentyp "IID_READSTAT_86_RF300")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode
FZP	BYTE	Error counter passive: distortion without communication
ABZ	BYTE	Dropout counter
CFZ	BYTE	Code error counter
SFZ	BYTE	Signature error counter
CRCFZ	BYTE	CRC-error counter
BSTAT	BYTE	Status of last command
ASMFZ	BYTE	Error counter for host interface (ASM)
reserved0	ARRAY [1..17]	

Tabelle 1-196 ATTRIBUTE "0x87" (Datentyp "IID_READSTAT_87_RF600")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode
hardware	CHAR	Type of hardware
hardware_version	WORD	Version of hardware
reserved0	WORD	
firmware	CHAR	Type of firmware
firmware_version_HB	BYTE	Version of firmware highbyte
firmware_version_LB	BYTE	Version of firmware lowbyte

Name	Typ	Kommentar
driver	CHAR	Type of driver
current_time_hour	BYTE	Hours [1]
current_time_min	BYTE	Minutes
current_time_sec	BYTE	Seconds
reserved1	BYTE	
SLG_version	BYTE	SLG version
baud	BYTE	Baudrate
reserved2	BYTE	
distance_limiting_SLG	BYTE	Selected transmit power
multitag_SLG	BYTE	Multitag SLG
field_ON_control_SLG	BYTE	Selected comunication typ
field_ON_time_SLG	BYTE	Selected channel
expert_mode	BYTE	Expert mode
status_ant	BYTE	Status of antenna [2]
scanning_time_SLG	BYTE	Radio communication profile (country specific radio standart)
MDS_control	BYTE	Presence mode

[1] Es wird der readerinterne Zeitstempel ausgegeben, welcher sich auf dieses Ereignis bezieht. Der readerinterne Zeitstempel wird nicht mit UTC synchronisiert.

[2] Der Antennen-Status bezieht sich auf den "ATTRIBUTE" (Bit 0 und 1) des zuletzt ausgeführten "SET-ANT" oder auf den durch "INIT"/"WRITE-CONFIG" voreingestellten Wert. Beim "INIT"/"WRITE-CONFIG" des RF620R ist der voreingestellte Wert "1" (int. Antenne an), beim RF630R "3" (Antennen 1 und 2 an).

Tabelle 1-197 ATTRIBUTE "0x88" (Datentyp "IID_READSTAT_88_RF600")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode(Subcommand)
hardware	CHAR	Type of hardware
hardware_version	WORD	Version of hardware
reserved_word1	WORD	Reserved
firmware	CHAR	Type of firmware
firmware_version_HB	BYTE	Version of firmware (High-Byte)
firmware_version_LB	BYTE	Version of firmware (Low-Byte)
driver	CHAR	Type of driver
current_time_hour	BYTE	Hours [1]
current_time_minute	BYTE	Minutes [1]
current_time_sec	BYTE	Seconds [1]
current_time_reservByte	BYTE	
SLG_version	BYTE	SLG-Version
baud	BYTE	Baudrate
reserved_byte1	BYTE	Reserved
distance_limiting_SLG	BYTE	Selected transmit power
multitag_SLG	BYTE	Multitag SLG

Name	Typ	Kommentar
field_ON_control_SLG	BYTE	Selected communication type
field_ON_time_SLG	BYTE	Selected channel
expert_mode	BYTE	Expert mode
status_ant	BYTE	Status of antenna [2)]
scanning_time_SLG	BYTE	Radio communication profile (country specific radio standart)
MDS_control	BYTE	Presence mode
blink_pattern	BYTE	Blink Pattern
act_algor_Single_Tag	Bool	Single_Tag [1]
act_algor_ITF_Phase2	Bool	ITF_Phase2 [2]
act_algor_ITF_Phase1	Bool	ITF_Phase1 [3]
act_algor_Smoothing	Bool	Smoothing [4]
act_algor_Blacklist	Bool	Blacklist [5]
act_algor_RSSI_Threshold	Bool	RSSI_Threshold [6]
act_algor_Power_Ramp	Bool	Power_Ramp [7]
act_algor_Power_Gap	Bool	Power_Gap [8]
Reserved1	Bool	Reserved1 [1]
Reserved2	Bool	Reserved2 [2]
Reserved3	Bool	Reserved3 [3]
Reserved4	Bool	Reserved4 [4]
act_algor_EPC_MemBankFilter	Bool	EPC_MemBankFilteres [5]
act_algor_Tag_Holg	Bool	Tag_Hold [6]
act_algor_Multi_Tag	Bool	Multi_Tag [7]
act_algor_ISTM	Bool	ISTM [8]
reserved_word2	WORD	Reserved
reserved_word3	WORD	Reserved
reserved_word4	WORD	Reserved
filtered_max_rssi	BYTE	Maximum RSSI value of a tag, of all filtered tags
reserved_byte2	BYTE	Reserved
filtered_tags_rssi	BYTE	Number of tags, filtered out by the RSSI threshold
reserved_byte3	BYTE	Reserved
filtered_tags_black_list	WORD	Number of tags, filtered out via Black-List
filtered_tags_epc_data	WORD	Number of tags, filtered out via EPC Data Filter
filtered_tags_smoothing	WORD	Number of tags in Tag List of status Not-Observed
itf_ph1_max_detect	WORD	Number of reads of a Tag, filtered out via ITF-phase 1
itf_ph1_tags_detect	WORD	Number of tags, filtered out via ITF-phase 1
itf_ph2_max_detect	WORD	Number of reads of a Tag, filtered out via ITF-phase 2
itf_ph2_tags_detect	WORD	Number of tags, filtered out via ITF-phase 2
filtered_istm_min_dist	WORD	Minimum distance of tags according to sorting criterion of ISTM
filtered_istm_tags	WORD	Number of tags, filtered out via ISTM algorithm

Name	Typ	Kommentar
last_error	BYTE	error code of the last occuring error (last_command)
reserved_byte4	BYTE	Reserved
error_command1	WORD	Last command (has lead to error code) "last_error"
error_command2	WORD	Last command (has lead to error code) "last_error"
error_command3	WORD	Last command (has lead to error code) "last_error"
reserved_word5	WORD	Reserved
reserved_array_byte	ARRAY[1..30] of Byte	

Tabelle 1-198 ATTRIBUTE "0x89" (Datentyp "IID_READSTAT_89_RF68xR")

Name	Typ	Kommentar
status_info	BYTE	SLG status mode(Subcommand)
hardware_version	BYTE	Version of hardware
firmware_version	ARRAY[1..4] of CHAR	Version of firmware
config ID	DWORD	Unix timestamp
inventory_status	WORD	0=inventory not active; 1=inventory active; 2=presence mode active
sum_of_filtered_tags	WORD	All filtered Tags
filtered_smoothing	WORD	Filtered Tags trough Smoothing
filtered_blacklist	WORD	Filtered Tags trough Blacklist
filtered_data-filter	WORD	Filtered Tags trough Data-Filter
filtered_RSSI_threshold	WORD	Filtered Tags trough RSSI Threshold
filtered_RSSI_delta	WORD	Filtered Tags trough RSSI Delta

Tabelle 1-199 ATTRIBUTE "0xA0" und "0xA1" (Datenytp "IID_READSTAT_A0_A1_RF600")

Name	Typ	Kommentar
reserved	BYTE	
Status_info	BYTE	Status-Info, SLG-Status SubCommand 20/21
number_tags_frame	BYTE	Number of Tags in this frame
number_tags_next_frames	BYTE	Number of Tags in the next frames
reserved_byte1	BYTE	Reserved
reserved_byte2	BYTE	Reserved
reserved_byte3	BYTE	Reserved
reserved_byte4	BYTE	Reserved
reserved_byte5	BYTE	Reserved
reserved_byte6	BYTE	Reserved
Black_List_ID	ARRAY[1..13] of "IID_IN_Blacklist"	EPC-ID Length

Name	Typ	Kommentar
Black_List_ID[1]	"IID_IN_Blacklist"	
EPC_Length	BYTE	EPC-ID Length
Antenna	BYTE	Antenna = Default 3
Filtered_Tag	WORD	Number of times - EPC-ID filtered out via BlackList
EPC	ARRAY[1..12] of Byte	EPC-ID
Black_List_ID[2]	"IID_IN_Blacklist"	
Black_List_ID[3]	"IID_IN_Blacklist"	
Black_List_ID[4]	"IID_IN_Blacklist"	
Black_List_ID[5]	"IID_IN_Blacklist"	
Black_List_ID[6]	"IID_IN_Blacklist"	
Black_List_ID[7]	"IID_IN_Blacklist"	
Black_List_ID[8]	"IID_IN_Blacklist"	
Black_List_ID[9]	"IID_IN_Blacklist"	
Black_List_ID[10]	"IID_IN_Blacklist"	
Black_List_ID[11]	"IID_IN_Blacklist"	
Black_List_ID[12]	"IID_IN_Blacklist"	
Black_List_ID[13]	"IID_IN_Blacklist"	

Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x81 ≙ 0x01
0x82 ≙ 0x02
0x83 ≙ 0x03
0x85 ≙ 0x05
0x87 ≙ 0x07
0x88 ≙ 0x08
0x90 ≙ 0x10
0x91 ≙ 0x11
0x92 ≙ 0x12

0xA0 ≙ 0x20
0xA1 ≙ 0x21

Tag_Status

Der Baustein "Tag_Status" liest die Statusinformationen des Transponders aus. Für die verschiedenen Transponder-Typen und Reader-Familien gibt es unterschiedliche Status-Modi, die Sie über den Parameter "ATTRIBUTE" auswählen können.

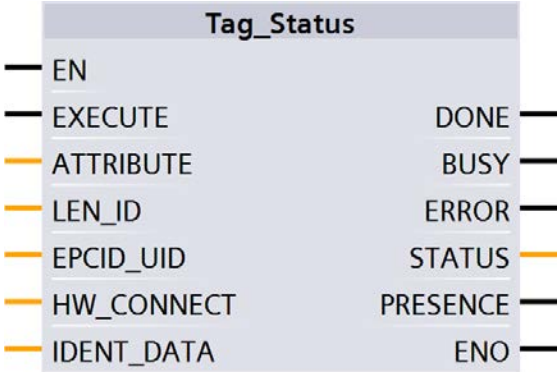

Bild 1-127 Baustein "Tag_Status"

Tabelle 1-200 Erläuterung zum Baustein "Tag_Status"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#0	Kennung der Status-Modi / Mögliche Eingaben: • RF200: 0x83 • RF300: 0x04, 0x82, 0x83 [1)] • RF620R, RF630R: 0x84, 0x85 • MOBY D: 0x83 [2)] • MOBY U: 0x80
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF61xR, RF68xR)

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[1...62] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R)
IDENT_DATA	Any / Variant	0	Ergebniswerte je nach Attribute Hinweis: Sie können entweder einen "Array_of_Byte" oder eine Variable mit einem der nachfolgenden Datentypen anlegen. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

1) Bei RF300 - erste Generation können mit dem Attribut "0x83" ausschließlich bei ISO-Transpondern die Werte abgefragt werden. Bei RF300 - zweite Generation können mit diesem Attribut bei allen Transponder-Typen die Werte abgefragt werden. Im "General Mode" funktioniert ausschließlich dieses Attribut.

2) nur SLG D10S

Ergebnisse

Beachten Sie, dass die UDTs erst dann angezeigt werden, wenn die Bausteine "Reader_Status" oder "Tag_Status" verwendet werden.

Tabelle 1-201 ATTRIBUTE "0x04" (Datentyp "IID_TAG_STATUS_04_RF300")

Name	Typ	Kommentar
reserved	BYTE	
status_info	BYTE	MDS status mode
UID	ARRAY [1..8] of BYTE	
MDS_type	BYTE	Type of MDS
Lock_state	BYTE	Write Protection Status EEPROM
Reserved1	ARRAY[1..6] of BYTE	

Tabelle 1-202 ATTRIBUTE "0x80" (Datentyp "IID_TAG_STATUS_80_MOBY_U")

Name	Typ	Kommentar
UID	ARRAY [1..4] of BYTE	Unique indentifier (MDS-Number)
MDS_type	BYTE	Type of MDS
sum_subframe_access_1	BYTE	Sum of subframe access Byte 1
sum_subframe_access_2	BYTE	Sum of subframe access Byte 2
sum_subframe_access_3	BYTE	Sum of subframe access Byte 3
sum_subframe_access_4	BYTE	Sum of subframe access Byte 4
sum_searchmode_access_1	BYTE	Sum of search mode access Byte 1
sum_searchmode_access_2	BYTE	Sum of search mode access Byte 2
ST_date_Week	BYTE	Date of last sleep-time change (week of year)
ST_date_Year	BYTE	Date of last sleep-time change (year)
battery_left_1	BYTE	Battery power left (percent) Byte 1
battery_left_2	BYTE	Battery power left (percent) Byte 2
ST	BYTE	Actual sleep-time on MDS

Tabelle 1-203 ATTRIBUTE "0x82" (Datentyp "IID_TAG_STATUS_82_RF300")

Name	Typ	Kommentar
reserved	BYTE	
status_info	BYTE	MDS status mode
UID	ARRAY [1..8] of BYTE	
LFD	BYTE	Magnetic flux density: correlation between limit-value
FZP	BYTE	Error counter passive: distortion without communication
FZA	BYTE	Error counter active: distortion during communication
ANWZ	BYTE	Presence counter: measure value for presence time
reserved1	ARRAY [1..3] of BYTE	

Tabelle 1-204 ATTRIBUTE "0x83" (Datentyp "IID_TAG_STATUS_83_ISO")

Name	Typ	Kommentar
reserved	BYTE	
status_info	BYTE	MDS status mode
UID	ARRAY [1..8] of BYTE	

Name	Typ	Kommentar
MDS_Type	BYTE	Type of MDS
IC_version	BYTE	Chip version
size_HB	BYTE	Size of Memory (high Byte)
size_LB	BYTE	Size of memory (low Byte)
lock_state	BYTE	Write protection status EEPROM
block_size	BYTE	Size of a block in addressable memory
number_of_block	BYTE	Number of blocks in addressable memory

Tabelle 1-205 ATTRIBUTE "0x84" (Datentyp "IID_TAG_STATUS_84_RF600")

Name	Typ	Kommentar
reserved	BYTE	
status_info	BYTE	MDS status mode
UID	ARRAY [1..8] of BYTE	
antenna	BYTE	Antenna which has observed the MDS
RSSI	BYTE	RSSI value
last_observed_hour	BYTE	Last observed time hour
last_observed_min	BYTE	Last observed time minute
last_observed_sec	BYTE	Last observed time seconds
last_observed_channel	BYTE	Last observed time channel
EPC_length	BYTE	EPC-Length
reserved1	BYTE	

1) Es wird der interne Zeitstempel der readerinternen Uhr ausgegeben, welche sich auf dieses Ereignis bezieht. Die readerinterne Uhr wird nicht mit UTC synchronisiert.

Tabelle 1-206 ATTRIBUTE "0x85" (Datentyp "IID_TAG_STATUS_85_RF600")

Name	Typ	Kommentar
status_info	BYTE	MDS status mode
antenna	BYTE	Antenna which has observed the MDS
channel	BYTE	Channel
UID	ARRAY [1..8] of BYTE	
DT_glimpsed_1	BYTE	Time elasped between acknowledgement and first read in [ms]1 Highbyte
DT_glimpsed_2	BYTE	Time elasped between acknowledgement and first read in [ms]2
DT_glimpsed_3	BYTE	Time elasped between acknowledgement and first read in [ms]3
DT_glimpsed_4	BYTE	Time elasped between acknowledgement and first read in [ms]4 Low-Byte

Name	Typ	Kommentar
reserved1	BYTE	
reserved2	BYTE	
reserved3	BYTE	
reserved4	BYTE	
last_observed_hour	BYTE	Last observed time hour
last_observed_min	BYTE	Last observed time minutes [1]
last_observed_sec	BYTE	Last observed time seconds [1]
last_observed_EPC_length	BYTE	Last observed time EPC length
EPC_ID_Byte	ARRAY [1..62] of BYTE	EPC-ID
reads_HB	BYTE	Number of Reads of MDS in Inventory (1 - 65535)
reads_LB	BYTE	Number of Reads of MDS in Inventory (1 - 65535)
RSSI	BYTE	Current RSSI value of MDS [2]
mean_RSSI	BYTE	Mean RSSI value of MDS
max_RSSI	BYTE	Max RSSI value of MDS
min_RSSI	BYTE	Min RSSI value of MDS
min_POWER	BYTE	Min Power value of MDS
current_POWER	BYTE	Current Power value of MDS [3]
reserved5	ARRAY[1..137] of BYTE	

[1] Es wird der readerinterne Zeitstempel ausgegeben, welcher sich auf dieses Ereignis bezieht. Der readerinterne Zeitstempel wird nicht mit UTC synchronisiert.

[2] Der Wert "Reads" gibt unabhängig von den eingestellten Smoothing Parametern die Summe der Transpondererkennungen (Inventories) an. Im Grenzfall kann der Zähler "Reads" auf diese Weise recht hohe Werte annehmen ohne dass der Transponder jemals den Zustand "Observed" erzielt hat.

[3] Der "current_Power"-Wert wird als Sendeleistung in 0,25 dBm-Schritten (ERP/EIRP) angegeben. Ein "current_Power"-Wert von "72" (0x48) entspricht demnach 18 dBm (ERP/EIRP).

Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x04 ≙ 0x01
0x82 ≙ 0x02
0x83 ≙ 0x03
0x84 ≙ 0x04
0x85 ≙ 0x05

Ident-Profil programmieren

Aufbau des Ident-Profils

Hinweis

Parallelbetrieb von Ident-Bausteinen und Ident-Profil ist nicht möglich

Beachten Sie, dass das CM bzw. der Reader nicht zeitgleich über die Ident-Bausteine und über das Ident-Profil betrieben werden kann.

Die in dem Kapitel "AUTOHOTSPOT" beschriebenen Bausteine stellen eine vereinfachte Schnittstelle des Ident-Profils dar. Sollten die Funktionalitäten der Bausteine für Ihre Anwendung nicht ausreichen, können Sie alternativ das Ident-Profil verwenden. Mithilfe des Ident-Profil können Sie komplexe Befehlsstrukturen programmieren und mit Befehlswiederholung arbeiten. Die nachfolgende Grafik zeigt das Ident-Profil, inklusive der damit umsetzbaren Befehle.

Hinweis

Ident-Profil für geschulte Anwender

Das Ident-Profil ist ein komplexer Baustein und beinhaltet alle Funktionalitäten der Ident-Bausteine. Das Ident-Profil wurde speziell für geschulte Baustein-Anwender konzipiert, die komplexe Funktionen mit Hilfe eines einzigen Bausteins projektieren wollen. Ungeschulten Anwendern, empfehlen wir die Verwendung der Ident-Bausteine.

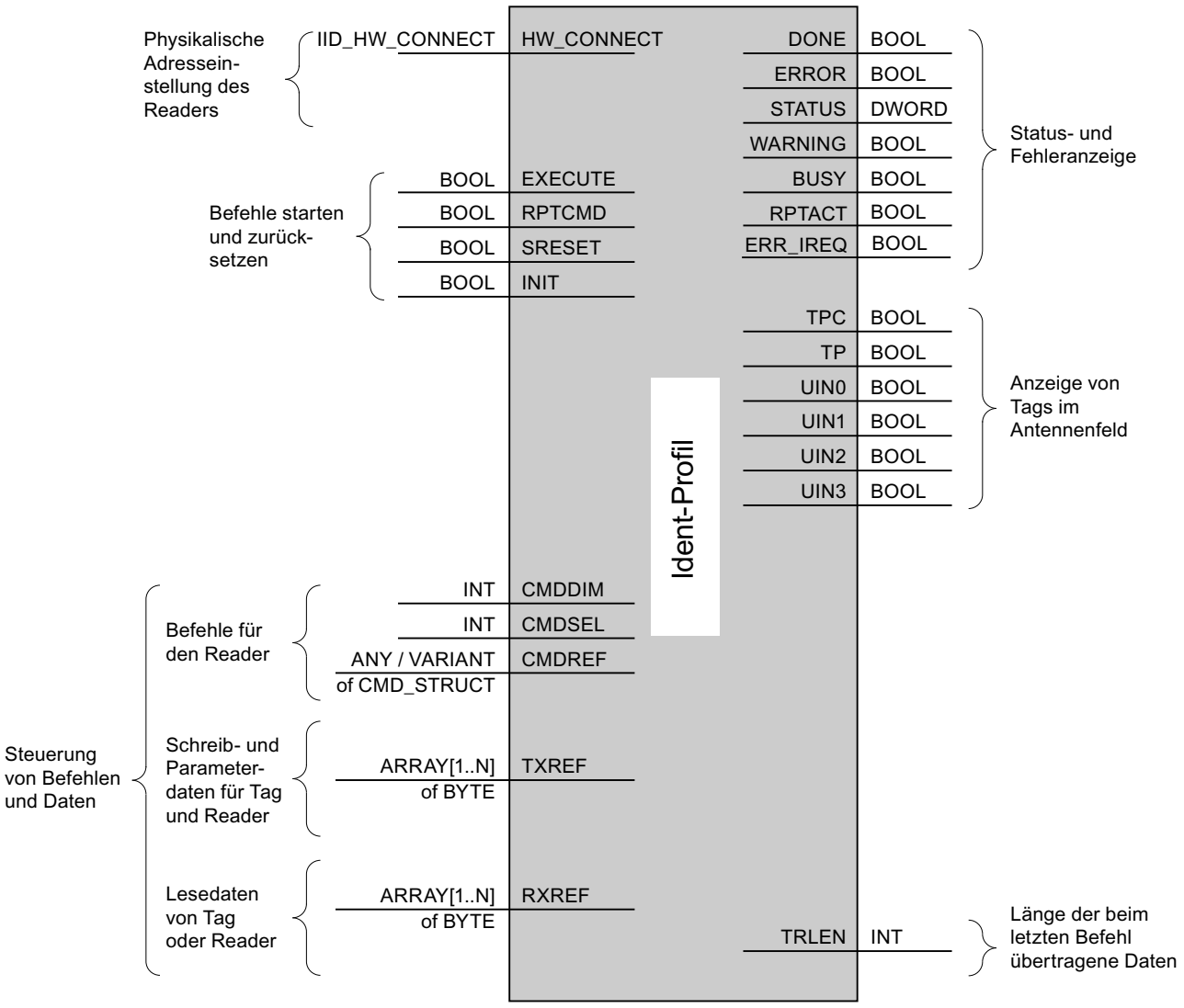

Bild 1-128 Die Input-Parameter des Ident-Profils

Hinweis

Arbeit mit mehreren Kanälen

Wenn Sie mit mehreren Kanälen arbeiten, müssen sie darauf achten, dass für jeden Kanal der Baustein mit einem eigenen Instanz-DB aufgerufen wird.

Schnittstellenbeschreibung

Tabelle 1-207 Eingangsparamater

Eingangsparamater	Datentyp	Default-Wert	Bedeutung
HW_CONNECT	HW_CONNECT	--	Eigener Datentyp zur physikalischen Adressierung von Kommunikationsmodul und Reader und zum Synchronisieren der Bausteine die pro Reader verwendet werden. Die Adressierung erfolgt wie in Kapitel "Datentyp 'I-ID_HW_CONNECT' parametrieren des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)" beschrieben.
EXECUTE	BOOL	FALSE	TRUE = Anstoß eines neuen Befehls Vor dem Starten müssen Sie den Befehl und die entsprechenden Parameter, in den mit "CMDREF" verknüpften Speicher, setzen.
RPTCMD	BOOL	FALSE	TRUE = Wiederholung des aktuell ausgeführten oder als nächstes auszuführenden Befehls durch das Kommunikationsmodul
SRESET	BOOL	FALSE	TRUE = Abbruch des aktuell im Kommunikationsmodul verarbeiteten Befehls
INIT	BOOL	FALSE	TRUE = Kommunikationsmodul führt einen Neustart aus und wird neu parametriert
CMDDIM	INT	10	Anzahl der Befehle im Parameter "CMDREF"
CMDSEL	INT	0	Auswahl des auszuführenden Befehls "CMDREF"; 1 ⇒ 1. Befehl, ... Der Wert des Parameters "CMDSEL" kann nie größer als der Wert des Parameters "CMDDIM" sein.
CMDREF	ARRAY[1...n] of CMD_STRUCT	--	Befehlsfeld Das Feld kann bis zu 100 Befehle aufnehmen. Die Befehle sind komplexe Variablen des Typs "CMD_STRUCT". Weitere Informationen zu "CMDREF" finden Sie im Kapitel "Befehlsübersicht des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)".
TXREF	ARRAY[1...n] of BYTE	--	Verweis auf globalen Speicherbereich für Sendedaten. Der Speicherbereich kann mit anderen Baustein-Instanzen geteilt werden. Der Wert "n" der einzelnen Bausteine ist variabel und kann bis zu 32 KB groß sein.
RXREF	ARRAY[1...n] of BYTE	--	Verweis auf globalen Speicherbereich für Empfangsdaten. Der Speicherbereich kann mit anderen Baustein-Instanzen geteilt werden. Der Wert "n" der einzelnen Bausteine ist variabel und kann bis zu 32 KB groß sein.

Tabelle 1-208 Ausgangsparamater

Ausgangsparamater	Datentyp	Default-Wert	Bedeutung
DONE	BOOL	FALSE	TRUE = Befehl wurde fehlerfrei ausgeführt.
ERROR	BOOL	FALSE	TRUE = Fehler wurde erkannt. Der Fehler wird im Parameter "STATUS" ausgegeben. Das Bit wird mit dem Start eines neuen Befehls automatisch zurückgesetzt.
STATUS	DWORD	FALSE	Warnung- und Fehler Bei "ERROR = TRUE" bzw. "WARNING = TRUE" steht im Parameter "STATUS" die Fehler- bzw. Warninformation. Weitere Informationen finden Sie im Kapitel "Fehlermeldungen des Handbuchs Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)".
WARNING	BOOL	FALSE	TRUE = Warnung wurde erkannt. Die Warnung wird im Parameter "STATUS" ausgegeben. Wenn der Parameter "ERROR" nicht zeitgleich gesetzt ist, wurden die Daten korrekt bearbeitet. Das Bit wird mit dem Start eines neuen Befehls automatisch zurückgesetzt.
BUSY	BOOL	FALSE	TRUE = Der Baustein bearbeitet einen Befehl. Andere Befehle, außer "INIT" und "SRESET", können nicht gestartet werden.
RPTACT	BOOL	FALSE	TRUE = "RPTCMD" ist aktiv Das Quittierungs-Bit zeigt an, dass der "Repeat-Mode" des CMs/Readers aktiv ist.
ERR_IREQ	BOOL	FALSE	TRUE = am Kommunikationsmodul oder Reader ist ein Fehler aufgetreten (z. B. beim Hochlauf des Readers, Verbindungsunterbrechung oder bei RF61xR, RF68xR bei Antennenfehlern oder Konfigurationsänderungen)
TPC	BOOL	FALSE	Transponder Presence Changed (nur bei aktivem Presence_Mode) TRUE = Ein neuer Transponder kommt ins Antennenfeld des Readers oder ein Transponder hat das Antennenfeld verlassen. Der Parameter wird nach erfolgreicher Ausführung des nächsten "INVENTORY"- (0x80, 0x81, 0x87) oder "INIT"-Befehls auf "FALSE" gesetzt.
TP	BOOL	FALSE	Transponder Presence TRUE = ein Transponder befindet sich im Antennenfeld des Readers.

Ausgangsparamater	Datentyp	Default-Wert	Bedeutung
UIN0	BOOL	FALSE	Bei RFID-Readern wird die Anzahl der im Antennenfeld befindlichen Transponder angezeigt. Bei optischen Lesegeräten werden die verschiedenen Zustände des Lesegeräts angezeigt. UIN0: Entspricht "IN_OP"-Bit des Lesegeräts UIN1: Entspricht "RDY"-Bit des Lesegeräts UIN2 + UIN3: Diese zwei Bits werden als unsigned Wert interpretiert (Bit 2 ist das niederwertigere Bit) welcher die Anzahl verfügbarer decodierter Codes darstellt. Wenn der Wert = 3 ist, dann sind drei oder mehr decodierte Codes verfügbar.
UIN1	BOOL	FALSE	
UIN2	BOOL	FALSE	
UIN3	BOOL	FALSE	
TRLEN	INT	0	Anzahl der Daten die nach erfolgreicher Ausführung des Befehls empfangen wurden.

Datenstruktur des Ident-Profils

Bei jedem Aufruf des Ident-Profils müssen Sie die Parameter ("HW_CONNECT", "CMDREF", "TXREF" und "RXREF"), wie im Kapitel "AUTOHOTSPOT" beschrieben, versorgen.

Der Aufruf des Ident-Profils erfolgt immer über den Eingangsparameter "HW_CONNECT" und die "IN/OUT"-Parameter "CMDREF", "TXREF" und "RXREF". Alle drei Parameter müssen in einem Datenbaustein angelegt werden. Der Zusammenhang zwischen den drei "IN/OUT"-Parametern wird im Folgenden genauer beschrieben:

- CMDREF (Befehlspuffer):
  - Array[1...n] of CMD_STRUCT
- TXREF (Sendepuffer):
  - Array[1...n] of Byte
  - Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.
- RXREF (Empfangspuffer):
  - Array[1...n] of Byte
  - Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

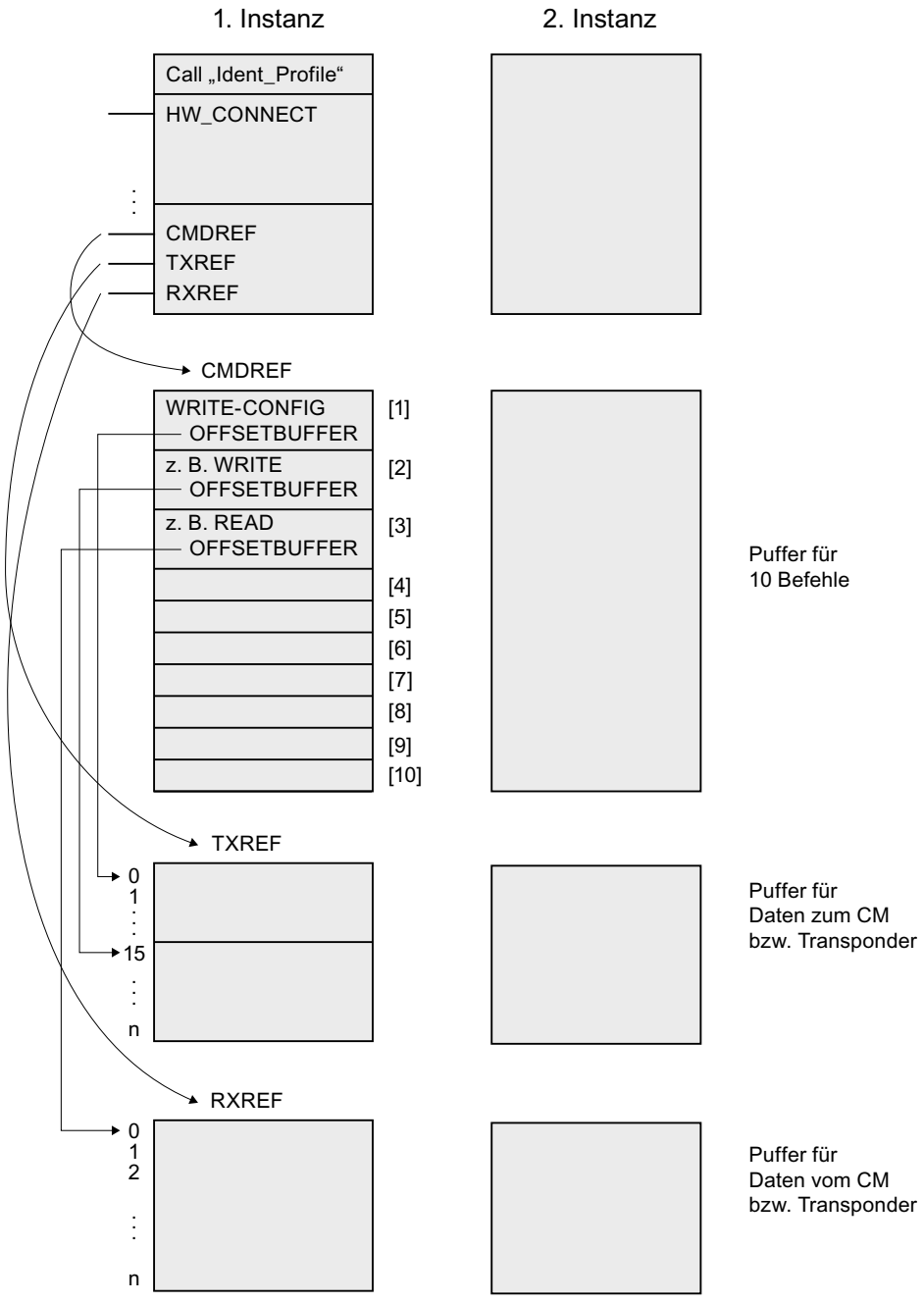

Bild 1-129 Datenstruktur-Beispiel des Ident-Profils

Erläuterung zu dem Datenstruktur-Beispiel

- CMDREF[1]:
  Befehl "WRITE-CONFIG", OFFSETBUFFER = 0
  An der Stelle "CMDREF[1]" müssen Sie den Befehl "WRITE-CONFIG" parametrieren, damit der "INIT/Reset" korrekt ausgeführt wird.
- CMDREF[2]:
  Befehl "WRITE", OFFSETBUFFER = 15
- CMDREF[3]:
  Befehl "READ", OFFSETBUFFER = 0

Wird der Befehl "CMDREF[2]" ausgewählt, wird ein Schreibbefehl gestartet und die zu schreibenden Daten werden aus dem Parameter "TXREF" ab dem Byte 15 geholt. Wird der Befehl "CMDREF[3]" ausgewählt, werden die gelesenen Daten ab Byte 0 im Parameter "RXREF" abgelegt.

Kommunikationsmodule

Kommunikationsmodul RF120C

Parametergruppe "Reader"

In dieser Parametergruppe können Sie alle reader-spezifischen Parameter des RF120C projektieren.

Tabelle 1-209 Parameter der Parametergruppe "Reader"

Parameter	Parameterwert	Default-Wert	Beschreibung
Diagnose-meldungen	Keine Hard Errors	Hard Errors	Mit diesem Parameter stellen Sie ein, ob Hardware-Diagnosemeldungen gemeldet werden sollen. • Keine: Neben der Standarddiagnose werden keine weiteren Alarme generiert. • Hard Errors: Schwerwiegende Hardware-Fehler werden über die S7-Diagnose gemeldet.
User Mode	Ident-Profil	Ident-Profil	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter wählen Sie den Baustein aus: • Ident-Profil: In der Steuerung kommt der Programmbaustein für das Ident-Profil zum Einsatz.
Ident-Gerät/-System	RF200 allgemein RF290R RF300 allgemein RF380R RF600 SLG D10S SLG D11S/D12S MOBY U Allgem. Reader Parameter über FB / Optische Lesegeräte	RF300 allgemein	Auswahl des angeschlossenen Ident-Geräts/-Systems. Abhängig von der getroffenen Auswahl, wird die Parametergruppe "Ident-System" entsprechend angepasst.

Parametergruppe "Ident-Gerät/-System"

In dieser Parametergruppe können Sie abhängig von den eingestellten Grundparametern, nachfolgende Reader-Parameter konfigurieren.

Tabelle 1-210 Übersicht über die "Reader-Parameter"

Reader-Parametrierung bei "Grundparameter"	Übertragungsgeschwindigkeit	MOBY Mode	Stand by-Zeit	Anwesenheitskontrolle	ERR-LED zurücksetzen	HF-Leistung	Reichweitenbegrenzung	Max. Transponder-Anzahl	Transponder-Typ	Reader-Typ	BERO-Betriebsart	BERO-Zeit	Byte-Sequenz des Reset-Parameters [hex]
RF200 allgemein	✓	-	-	✓	✓	-	-	✓	✓	-	-	-	-
RF290 R	✓	-	-	✓	✓	✓	-	✓	✓	-	-	-	-
RF300 allgemein	✓	-	-	✓	✓	-	-	✓	✓	-	-	-	-
RF380 R	✓	-	-	✓	✓	✓	-	✓	✓	-	-	-	-
RF600	✓	-	-	✓	✓	-	-	✓	-	✓	-	-	-
SLG D10S	✓	-	-	✓	✓	✓	-	✓	✓	-	-	-	-
SLG D11S/ D12S	✓	-	-	✓	✓	✓	-	✓	✓	-	-	-	-
MOBY U	✓	-	✓	✓	✓	-	✓	-	-	-	✓	✓	-
Allgem. Reader	✓	-	-	-	-	-	-	-	-	-	-	-	✓
Parameter über FB / Optische Lesegeräte	✓	✓	-	-	-	-	-	-	-	-	-	-	-

Tabelle 1-211 Parameter der Parametergruppe "Ident-Gerät/-System"

Parameter	Parameterwert	Default-Wert	Beschreibung
Übertragungs-geschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Beachten Sie, dass der hier angegebene Wert automatisch aus der Gerätekonfiguration der angeschlossenen Geräte übernommen wird. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem RFID-Reader: Nach der Umstellung der Übertragungsgeschwindigkeit muss der Reader aus- und wieder eingeschaltet werden (Spannung aus/ein). Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der in der Firmware des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.
MOBY Mode	RF200/RF300/RF600; MV400/MV500; MOBY U/D	RF200/RF300/ RF600; MV400/ MV500; MOBY U/D	Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein.
Standby-Zeit	0 - 1400 ms	0 ms	Standby-Zeit (scanning_time) für den Transponder. Erhält der Transponder vor Ablauf der Standby-Zeit einen weiteren Befehl, so kann dieser sofort bearbeitet werden. Erhält der Transponder einen Befehl nach Ablauf der Standby-Zeit, so wird die Befehlsbearbeitung um die "sleep_time" des Transponders verzögert.
Anwesenheitskontrolle	An Aus (HF-Feld an) Aus (HF-Feld aus)	An	An = Sobald sich ein Transponder im Antennenfeld des Readers befindet, wird dessen Anwesenheit gemeldet. Aus (HF-Feld an) = Die Anwesenheitsanzeige am FB wird unterdrückt. Die Antenne am Reader ist jedoch eingeschaltet, solange diese nicht durch einen Befehl abgeschaltet wurde. Aus (HF-Feld aus) = Die Antenne wird nur eingeschaltet, wenn ein Befehl abgeschickt wird und schaltet sich danach wieder ab.
ERR-LED zurücksetzen	An Aus	An	An = Das Blinken der Error-LED am Kommunikationsmodul wird durch jeden Reset des FBs zurückgesetzt. Aus = An der Error-LED wird immer der letzte Fehler angezeigt. Ein Rücksetzen der Anzeige ist nur durch Ausschalten des Kommunikationsmoduls möglich.
HF-Leistung [1)]	0,50 ... 5,00	1,00 1,25	Einstellung für die Ausgangsleistung des Readers. Die auswählbaren Werte sind abhängig vom angeschlossenen Gerät.

Parameter	Parameterwert	Default-Wert	Beschreibung
Reichweitenbegrenzung	0,2 m 0,5 m 1,0 m 1,5 m 2,0 m 2,5 m 3,0 m 3,5 m	1,5 m	Begrenzung der Strahlungsreichweite
Max. Transponder-Anzahl	1 ... 40	1	Anzahl der erwarteten Transponder im Antennenfeld. Die Auswahl ist abhängig vom angeschlossenen Gerät.
Transponder-Typ	1)	1)	Auswahl der verwendeten Transponder-Typen. Die Auswahl ist abhängig vom angeschlossenen Gerät.
Reader-Typ 1)	RF620R ETSI RF620R FCC RF620R CMIIT RF630R ETSI RF630R FCC RF630R CMIIT	RF620R ETSI	Auswahl des verwendeten Readers. Durch Auswahl eines Readers öffnet sich das die Parametergruppe "Reader-Typ". Die Parameter sind in der folgenden Tabelle beschrieben.
BERO-Betriebsart	Ohne BEROs 1 oder 2 BEROs 1er BERO an, 2er BERO aus Synchronisation durch Kabelverbindung	Ohne BEROs	• Ohne BEROs keine Reader-Synchronisation • 1 oder 2 BEROs Die BEROs sind logisch ODER verknüpft. Während der Zeit der Betätigung eines BEROs ist das Antennenfeld eingeschaltet. • 1er BERO an, 2er BERO aus Der 1. BERO schaltet das Antennenfeld ein und der 2. BERO schaltet das Antennenfeld aus. Wenn zwei BEROs vorhanden sind und "BERO-Zeit in s" parametriert ist, wird das Antennenfeld automatisch ausgeschaltet, wenn der 2. BERO nicht innerhalb dieser BERO-Zeit schaltet. Ist keine "BERO-Zeit in s" parametriert, so bleibt das Antennenfeld bis zur Betätigung des 2. BERO eingeschaltet. • Synchronisation durch Kabelverbindung Reader-Synchronisation über Kabelverbindung aktivieren (siehe Handbuch für Projektierung, Montage und Service für MOBY U).

Parameter	Parameterwert	Default-Wert	Beschreibung
BERO-Zeit	0 ... 255 s	0 s	Ist nur einstellbar, wenn die BERO Betriebsart auf "1er BERO an, 2er BERO aus" eingestellt ist. • 0 Die Zeitüberwachung ist abgeschaltet. Für die Feldabschaltung wird der 2. BERO benötigt. • 1 ... 255 s Einschaltzeit für das Reader-Feld
Byte-Sequenz des Reset-Parameters [hex]	1)	1)	Hexadezimale Darstellung der Byte-Sequenz des Reset-Parameters des Readers.

1) Eine ausführliche Beschreibung zu diesen Parametern finden Sie in den nachfolgenden Absätzen.

Der Parameter "HF-Leistung"

Die auswählbaren Werte sind abhängig von dem angegebenen Wert in dem Parameter "Ident-Gerät/-System".

- RF290R
  - Wertebereich: 0,50 ... 5,00 W
  - Default-Wert: 1,00 W
- RF380R
  - Wertebereich: 0,50 ... 2,00 W
  - Default-Wert: 1,25 W

Der Parameter "Transponder-Typ"

Auswahl der verwendeten Transponder. Die Auswahl ist abhängig von dem angegebenen Wert in dem Parameter "Ident-Gerät/-System". Es können folgende Transponder-Typen ausgewählt werden:

Tabelle 1-212 Auswahloptionen "Transponder-Typ"

Parameter "Ident-Gerät/-System"	Mögl. Werte
RF200 allgemein RF290R	ISO 15693
RF300 allgemein RF380R	RF300 ISO 15693
SLG D10S SLG D11S/D12S	I-Code ISO 15693 ISO-FRAM

Der Parameter "Byte-Sequenz des Reset-Parameters [hex]"

Über diese Funktion können Sie die Reset-Parameter in hexadezimaler Darstellung angeben. Diese Einstellung richten sich ausschließlich an geschulte Anwender.

Folgende Reset-Parameter stehen zur Verfügung:

Tabelle 1-213 Reset-Parameter

Byte	1	2...5 [1]	6	7...8	9	10	11	12	13...14	15	16
Wert	0x04	0x00	0x0A	0x00	scan-ning_ time	param	opti-on_1	distance_ limiting	multitag	field_on_ control	field_on_ time

[1] Bei dem Kommunikationsmodul RF180C ab V2.2 in Verbindung mit MOBY U sind Byte 4 mit Kalenderwoche und 5 mit Jahr vorbelegt.

Parametergruppe "Reader-Typ" (nur bei RF600)

In der Parametergruppe "Reader-Typ" werden, für den in der Parametergruppe "Ident-System" ausgewählten "Reader-Typ", weitere spezifische Parameter eingestellt.

Tabelle 1-214 Parameter der Parametergruppe "Reader-Typ"

Parameter	Parameterwert	Default-Wert	Beschreibung
Funkprofil	-	-	Auswahl des jeweiligen Funkprofils für ETSI, FCC oder CMIIT.
Multitag-Betriebsart	UID = EPC-ID (8 Byte) UID = Handle-ID (4 Byte) UID = 0 (Singletag)	UID = EPC-ID (8 Byte)	• UID = EPC-ID (8 Byte) 8 Byte UID der Bytes 5-12 der 12 Bytes langen EPC-ID • UID = Handle-ID (4 Byte) 4 Byte UID als Handle-ID für Zugriffe auf Transponder mit beliebig langer EPC-ID
Intelligent Singletag Mode (ISTM)	An Aus	Aus	Aktivieren/Deaktivieren des Algorithmus "Intelligent Singletag Mode ISTM" [1]
Black List	An Aus	Aus	Aktivieren/Deaktivieren der "Blacklist" [1]
Strahlungsleistung interne Antenne (RF620R)	0 ... B	4	Einstellen der Strahlungsleistung für die interne Antenne [1] [2]
Interne Antenne (RF620R)	-	-	Aktivieren/Deaktivieren die interne Antenne. Bei dem RF620R ist entweder nur die interne oder externe Antenne einstellbar.
Sendeleistung externer Antenneanschluss (RF620R)	0 ... 9	4	Einstellen der Sendeleistung für die externe Antenne [1] [2]
Externe Antenne (SetAnt erforderlich) (RF620R)	-	-	Aktivieren/Deaktivieren die externe Antenne. Bei dem RF620R ist entweder nur die interne oder externe Antenne einstellbar.
Sendeleistung ANT 1 (RF630R)	0 ... 9	4	Einstellen der Sendeleistung für die Antenne 1[1] [2]
Sendeleistung ANT 2 (RF630R)	0 ... 9	4	Einstellen der Sendeleistung für die Antenne 2[1] [2]
Kommunikationsgeschwindigkeit	Sichere Erkennung Schnelle Erkennung	Sichere Erkennung	[1]

Parameter	Parameterwert	Default-Wert	Beschreibung
Tag Hold	An Aus	Aus	Aktivieren/Deaktivieren des "Tag Hold" [1]
Scanning Mode	An Aus	Aus	Aktivieren/Deaktivieren des "Scanning Mode" [1]
Kanalzuordnung (nur bei Funkprofil ETSI)	-	-	Auswahl der zu verwendenden Funkkanäle [1]

[1] Weitere Informationen finden Sie im "Parametrierhandbuch RF620R/RF630R". *

[2] Die Werte für die Sende-/Strahlungsleistung der Antennen finden Sie in der nachfolgenden Tabelle.

* Das Parametrierhandbuch RF620R/RF630R finden Sie hier.

Tabelle 1-215 Sende-/Strahlungsleistung der Antennen

Hex-Wert	RF630R Sendeleistung	RF620R Strahlungsleistung (interne Antenne)			RF620R Sendeleistung
		ETSI	FCC	CMIIT	
	dBm / (mW)	dBm / (mW) ERP	dBm / (mW) EIRP	dBm / (mW) ERP	dBm / (mW)
0	18 / (63)	18 / (63)	20 / (100)	18 / (65)	18 / (63)
1	19 / (79)	19 / (79)	21 / (126)	19 / (79)	19 / (79)
...	...	...	...	...	...
4	22 / (158)	22 / (158)	24 / (251)	22 / (158)	22 / (158)
...	...	...	...	...	...
9	27 / (501)	27 / (501)	29 / (794)	27 / (501)	27 / (501)
A	27 / (501)	28 / (631)	30 / (1000)	28 / (631)	27 / (501)
B (...F)	27 / (501)	29 / (794)	31 / (1259)	29 / (794)	27 / (501)

Kommunikationsmodul RF170C

Kommunikationsmodul für Ident-Systeme - RF170C, RS422/RS232

Die Baugruppe RF170C RS422/RS232 ist ein Kommunikationsmodul für Ident-Systeme für das Peripheriesystem ET 200pro. An einer IM 154-8 CPU sowie an einem Interfacemodul IM 154-1 DP, IM 154-2 DP High Feature oder IM 154-4 PN High Feature der ET 200pro können mehrere Kommunikationsmodule RF170C RS422/RS232 montiert und betrieben werden.

An das RF170C RS422/RS232 können max. zwei Reader angeschlossen werden. Die Bearbeitung der angeschlossenen Reader erfolgt parallel.

Die Funktionsbausteine FB 45, FC 45 und der Programmbaustein für das Ident-Profil/RFID-Normprofil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge. Die Funktionsbausteine FB 45, FC 45 und der Programmbaustein für das Ident-Profil/RFID-Normprofil können sowohl in der S7-300 als auch in der S7-400 betrieben werden. Für die Steuerungen S7-1200 und S7-1500 muss das Ident-Profil verwendet werden.

Der Zugriff auf die Transponder-Daten erfolgt über Normaladressierung oder Filehandler-Adressierung.

Physikalische Adressierung des Transponder

Die physikalische Adressierung eines Transponder wird auch als Normaladressierung bezeichnet. Bei der Normaladressierung können Sie die Datenstruktur seiner Transponder selbst aufbauen. Dabei bestimmen Sie, auf welcher physikalischen Adresse welche Daten geschrieben werden. Die Adressierung des Transponder-Speichers beginnt in der Regel bei Adresse 0x0000 und endet auf einer Endadresse, die der Größe des Transponder-Speichers entspricht.

Filehandler-Adressierung des Transponder

Bei der Filehandler-Adressierung geben Sie für den Zugriff auf die Daten einen File-Namen an. Der File-Name besteht aus acht ASCII-Zeichen. Der Filehandler verwaltet die Anwenderdaten auf dem Transponder selbständig. Sie brauchen keine Datenstrukturen auf dem Transponder zu projektieren.

Parametergruppe "Baugruppenparameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter des RF170C RS422/RS232 projektieren.

Tabelle 1-216 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
User Mode	Ident-Profil/RFID-Normprofil FB 45 / FC 45 FB 55 / FC 55	Ident-Profil/RFID-Normprofil	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter wählen Sie den Baustein aus: • Ident-Profil/RFID-Normprofil: In der Steuerung kommt der Programmbaustein für das Ident-Profil/RFID-Normprofil zum Einsatz. • FB 45 / FC 45: Singletag-Betrieb. In der Steuerung wird der FB 45 (PROFIBUS/PROFINET) oder der FC 45 (PROFIBUS) eingesetzt. • FB 55 / FC 55: Multitag-Betrieb. In der Steuerung wird der FB 55 (PROFIBUS/PROFINET) oder der FC 55 (PROFIBUS) eingesetzt.
MOBY Mode	RF200/RF300/RF600; MV400/MV500; MOBY U/D MOBY I/E MV300 Freeport-Protokoll RF300 Filehandler	RF200/RF300/RF600; MV400/MV500; MOBY U/D	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein. • RF200/RF300/RF600; MV400/MV500; MOBY U/D • MOBY I/E • MV300 • Freeport-Protokoll • RF300 Filehandler Normaladressierung: Der Transponder wird mit physikalischen Adressen angesprochen. Filehandler: Der Transponder muss vor der Benutzung formatiert werden.
Übertragungsgeschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem RFID-Reader: Nach der Umstellung der Übertragungsgeschwindigkeit muss der Reader aus- und wieder eingeschaltet werden (Spannung aus/ein). Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der in der Firmware des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.

Parameter	Parameterwert	Default-Wert	Beschreibung
Diagnose-meldungen	Keine Hard Errors	Keine	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie ein, ob Hardware-Diagnosemeldungen gemeldet werden sollen. • Keine: Neben der Standarddiagnose werden keine weiteren Alarme generiert. • Hard Errors: Schwerwiegende Hardware-Fehler werden über die S7-Diagnose gemeldet.
Unterdrückung Error-LED	Keine Kanal 1 Kanal 2	Keine	Deaktivieren der Error-LED (ERR) eines Kanals. Das Kommunikationsmodul verfügt über zwei Kanäle, an die Reader / optischen Lesegeräte angeschlossen werden können. Wird nur einer der Kanäle verwendet, blinkt die Error-LED des anderen Kanals permanent. Mithilfe der Unter-drückung können Sie die Error-LED des nicht verwendeten Kanals deaktivieren.
Schnittstelle	RS232 RS422	--	Auswahl des Schnittstellen-Typs, den die ange-schlossene Hardware (Reader / optische Lese-geräte) verwendet.

Parametergruppe "Telegramm"

Die Parametergruppe "Telegramm" wird nur angezeigt, wenn Sie im Parameter "MOBY Mode" der Baugruppenparameter den Parameterwert "Freeport-Protokoll" ausgewählt haben. In dieser Parametergruppe können Sie alle spezifischen Parameter des "Freeport-Protokoll" projektieren.

Tabelle 1-217 Parameter der Parametergruppe "Telegramm"

Parameter	Parameterwert	Default-Wert	Beschreibung
Datenbits	7 8	8	Auswahl der Anzahl Bits, auf die ein Zeichen abgebildet wird.
Parität	Keine Ungerade Gerade Fixwert 1 Fixwert 0	Keine	Auswahl der Parität Eine Folge von Datenbits kann um ein Paritätsbit erweitert werden. Das Paritätsbit ergänzt durch seinen Wert "0" oder "1" die Summe aller Bits (Datenbits und Paritätsbits) auf einen definierten Zustand. Die Datensicherheit wird dadurch erhöht. • Keine: Daten werden ohne Paritätsbit gesendet. • Ungerade: Paritätsbit wird so gesetzt, dass die Summe der Datenbits (inkl. Paritätsbit) mit Signalzustand "1" ungerade ist. • Gerade: Paritätsbit wird so gesetzt, dass die Summe der Datenbits (inkl. Paritätsbit) mit Signalzustand "1" gerade ist. • Fixwert 1: Paritätsbit wird fest auf den Wert "1" gesetzt. • Fixwert 0: Paritätsbit wird fest auf den Wert "0" gesetzt.
Stopbits	1 2	1	Auswahl der Anzahl Stopbits, die das Ende eines Zeichens kennzeichnen. Die Stopbits werden bei der Übertragung an jedes übertragene Zeichen angehängt.

Parameter	Parameterwert	Default-Wert	Beschreibung
Festlegung der Endeerkennung	Nach Ablauf der Zeichenverzugszeit Nach Empfang einer festen Zeichenanzahl Nach Empfang der/des Endezeichen(s)	Nach Ablauf der Zeichenverzugszeit	Festlegung der Endeerkennung eines Empfangstelegramms auf: • Nach Ablauf der Zeichenverzugszeit: Das Telegramm hat weder eine feste Läge noch definierte Endezeichen. Das Ende eines Telegramms ist durch eine Lücke in der Zeichenfolge bestimmt. Die Größe dieser Lücke wird durch die Zeichenverzugszeit festgelegt. • Nach Empfang einer festen Zeichenanzahl: Die Läge der Empfangstelegramme ist immer gleich. Beim Empfang von Daten wird das Telegrammende erkannt, wenn die parametrierte Anzahl von Zeichen empfangen wird. • Nach Empfang der/des Endezeichen(s): Am Ende des Telegramms stehen ein oder zwei definierte Endezeichen. Beim Empfang von Daten wird das Telegrammende erkannt, wenn das parametrierte Endezeichen empfangen wird
Anzahl der Endezeichen	1 2	1	Auswahl der Anzahl der Endezeichen. Es können maximal 2 Endezeichen projektiert werden. Beim Empfang von Daten wird das Telegrammende erkannt, wenn die gewählte Endezeichenkombination empfangen wird.
1. Endezeichen	0...7F / 0...FF	3	Eingabe des 1. Endezeichens von maximal zwei Endezeichen für Endekriterium "Nach Empfang der/des Endezeichen(s)". Das gewählte Endezeichen bzw. die gewählteEndezeichenkombination begrenzt die Länge des jeweiligen Telegramms. Parameterwert abhängig vom Parameter "Datenbits".
2. Endezeichen	0...7F / 0...FF	0	Eingabe des 2. Endezeichens von maximal zwei Endezeichen für Endekriterium "Nach Empfang der/des Endezeichen(s)". Das gewählte Endezeichen bzw. die gewählte Endezeichenkombination begrenzt die Länge des jeweiligen Telegramms. Parameterwert abhängig vom Parameter "Datenbits".

Parameter	Parameterwert	Default-Wert	Beschreibung
Telegrammlänge	1...233 / 1...229	233	Eingabe der Telegrammlänge in Byte für das Endekriterium "Nach Empfang einer festen Zeichenanzahl".
Zeichenverzugszeit	0...65535	15	Eingabe der Zeit [ms], die verstreichen darf, bis ein Telegrammende erkannt wird. Wählen Sie die Zeichenverzugszeit in Abhängigkeit vom Sendeverhalten Ihres Kommunikationspartners. Die Zeichenverzugszeit ist in Abhängigkeit von der Datenübertragungsgeschwindigkeit auf einen Mindestwert begrenzt. Beachten Sie, dass der ASCII-Treiber auch beim Senden eine Pause zwischen zwei Telegrammen einhält.

Parametergruppe "E/A-Adressen"

Eingänge

1. Weisen Sie dem Modul eine Anfangsadresse zu.
2. Falls Sie den Adressbereich einem Teilprozessabbild zuweisen wolle, wählen Sie in der Klappliste "Prozessabbild" das von Ihnen gewünschte Teilprozessabbild aus.

Ausgänge

1. Weisen Sie dem Modul eine Anfangsadresse zu.
2. Falls Sie den Adressbereich einem Teilprozessabbild zuweisen wollen, wählen Sie in der Klappliste "Prozessabbild" das von Ihnen gewünschte Teilprozessabbild aus.

Hinweis

Eingangs- und Ausgangsadresse identisch sein müssen.

Kommunikationsmodul RF180C

Kommunikationsmodul für Ident-Systeme - RF180C

Die Baugruppe RF180C ist ein Kommunikationsmodul für Ident-Systeme für PROFINET IO.

An das RF180C können max. zwei Reader angeschlossen werden. Die Bearbeitung der angeschlossenen Reader erfolgt parallel.

Die Funktionsbausteine FB 45, FB 55 und der Programmbaustein für das RFID-Normprofil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge. Die Funktionsbausteine FB 45, FB 55 und der Programmbaustein für das RFID-Normprofil können sowohl in der S7-300 als auch in der S7-400 betrieben werden.

Der Zugriff auf die Transponder-Daten erfolgt über Normaladressierung oder Filehandler-Adressierung.

Physikalische Adressierung des Transponder

Die physikalische Adressierung eines Transponder wird auch als Normaladressierung bezeichnet. Bei der Normaladressierung können Sie die Datenstruktur seiner Transponder selbst aufbauen. Dabei bestimmen Sie, auf welcher physikalischen Adresse welche Daten geschrieben werden. Die Adressierung des Transponder-Speichers beginnt in der Regel bei Adresse 0x0000 und endet auf einer Endadresse, die der Größe des Transponder-Speichers entspricht.

Filehandler-Adressierung des Transponder

Bei der Filehandler-Adressierung geben Sie für den Zugriff auf die Daten einen File-Namen an. Der File-Name besteht aus acht ASCII-Zeichen. Der Filehandler verwaltet die Anwenderdaten auf dem Transponder selbständig. Sie brauchen keine Datenstrukturen auf dem Transponder zu projektieren.

Parametergruppe "Baugruppenparameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter des RF180C projektieren.

Tabelle 1-218 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
User Mode	FB 45 FB 55 FB 56 RFID-Normprofil	FB 45	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter wählen Sie den Baustein aus: • FB 45: Singletag-Betrieb. In der Steuerung wird der FB 45 (PROFIBUS/PROFINET) eingesetzt. • FB 55: Multitag-Betrieb. In der Steuerung wird der FB 55 (PROFIBUS/PROFINET) eingesetzt. • FB 56: Multitag-Betrieb. In der Steuerung wird der FB 56 (PROFIBUS/PROFINET) eingesetzt. • RFID-Normprofil: In der Steuerung kommt der Programmbaustein für das RFID-Normprofil zum Einsatz.
MOBY Mode	MOBY I/E-Normaladressierung MOBY I-Filehandler RF200/RF300/RF600; MOBY U/D-Normaladr. MOBY U-Filehandler RF300 Filehandler	RF200/RF300/ RF600; MOBY U/D-Normaladr.	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein. • MOBY I/E-Normaladressierung • MOBY I-Filehandler • RF200/RF300/RF600; MOBY U/D-Normaladr. • MOBY U-Filehandler • RF300 Filehandler Normaladressierung: Der Transponder wird mit physikalischen Adressen angesprochen. Filehandler: Der Transponder muss vor der Benutzung formatiert werden.

Parameter	Parameterwert	Default-Wert	Beschreibung
Übertragungs-geschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem RFID-Reader: Nach der Umstellung der Übertragungsgeschwindigkeit muss der Reader aus- und wieder eingeschaltet werden (Spannung aus/ein). Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der in der Firmware des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.
Diagnose-meldungen	Keine Hard Errors Hard/Soft Errors	Keine	Mit diesem Parameter stellen Sie ein, ob Hardware-Diagnosemeldungen gemeldet werden sollen. • Keine: Neben der Standarddiagnose werden keine weiteren Alarme generiert. • Hard Errors: Schwerwiegende Hardware-Fehler werden über die S7-Diagnose gemeldet. • Hard/Soft Errors: Schwerwiegende Hardware-Fehler, sowie Fehler, die während der Befehlsbearbeitung auftreten, werden über die S7-Diagnose gemeldet.
Unterdrückung Error-LED	Keine Kanal 1 Kanal 2	Keine	Deaktivieren der Error-LED (ERR) eines Kanals. Das Kommunikationsmodul verfügt über zwei Kanäle, an die Reader / optischen Lesegeräte angeschlossen werden können. Wird nur einer der Kanäle verwendet, blinkt die Error-LED des anderen Kanals permanent. Mithilfe der Unterdrückung können Sie die Error-LED des nicht verwendeten Kanals deaktivieren.

Kommunikationsmodul ASM 456

Kommunikationsmodul für Ident-Systeme - ASM 456

Die Baugruppe ASM 456 ist ein Kommunikationsmodul für Ident-Systeme für PROFIBUS DP V1.

An das ASM 456 können max. zwei Reader angeschlossen werden. Die Bearbeitung der angeschlossenen Reader erfolgt parallel.

Die Funktionsbausteine FB 45, FC 45, FB 55, FC 55, FC 56 und der Programmbaustein für das Ident-Profil/RFID-Normprofil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge. Die Funktionsbausteine FB 45, FC 45, FB 55, FC 55, FC 56 und der

Programmbaustein für das Ident-Profil/RFID-Normprofil können sowohl in der S7-300 als auch in der S7-400 betrieben werden. Für die Steuerungen S7-1200 und S7-1500 muss das Ident-Profil verwendet werden.

Der Zugriff auf die Transponder-Daten erfolgt über Normaladressierung oder Filehandler-Adressierung.

Physikalische Adressierung des Transponder

Die physikalische Adressierung eines Transponder wird auch als Normaladressierung bezeichnet. Bei der Normaladressierung können Sie die Datenstruktur seiner Transponder selbst aufbauen. Dabei bestimmen Sie, auf welcher physikalischen Adresse welche Daten geschrieben werden. Die Adressierung des Transponder-Speichers beginnt in der Regel bei Adresse 0x0000 und endet auf einer Endadresse, die der Größe des Transponder-Speichers entspricht.

Filehandler-Adressierung des Transponder

Bei der Filehandler-Adressierung geben Sie für den Zugriff auf die Daten einen File-Namen an. Der File-Name besteht aus acht ASCII-Zeichen. Der Filehandler verwaltet die Anwenderdaten auf dem Transponder selbständig. Sie brauchen keine Datenstrukturen auf dem Transponder zu projektieren.

Parametergruppe "Gerätespezifische Parameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter des ASM 456 projektieren.

Tabelle 1-219 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
User Mode	Ident-Profil/RFID-Normprofil FB 45 / FC 45 FB 55 / FC 55 FC 56	Ident-Profil/RFID-Normprofil	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter wählen Sie den Baustein aus: • Ident-Profil/RFID-Normprofil: In der Steuerung kommt der Programmbaustein für das Ident-Profil/RFID-Normprofil zum Einsatz. • FB 45 / FC 45: Singletag-Betrieb. In der Steuerung wird der FB 45 (PROFIBUS/PROFINET) oder der FC 45 (PROFIBUS) eingesetzt. • FB 55 / FC 55: Multitag-Betrieb. In der Steuerung wird der FB 55 (PROFIBUS/PROFINET) oder der FC 55 (PROFIBUS) eingesetzt. • FC 56 Filehandler für S7-300 und S7-400
MOBY Mode	RF200/RF300/RF600; MV400/MV500; MOBY U/D RF680R/RF685R MOBY I/E Normaladressierung RF300 Filehandler MOBY U Filehandler MOBY I Filehandler	RF200/RF300/RF600; MV400/MV500; MOBY U/D	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein. • RF200/RF300/RF600; MV400/MV500; MOBY U/D • RF680R/RF685R • MOBY I/E Normaladressierung • RF300 Filehandler • MOBY U Filehandler • MOBY I Filehandler Normaladressierung: Der Transponder wird mit physikalischen Adressen angesprochen. Filehandler: Der Transponder muss vor der Benutzung formatiert werden.

Parameter	Parameterwert	Default-Wert	Beschreibung
Übertragungs-geschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem RFID-Reader: Nach der Umstellung der Übertragungsgeschwindigkeit muss der Reader aus- und wieder eingeschaltet werden (Spannung aus/ein). Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der in der Firmware des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.
Diagnose-meldungen	Keine Hard Errors Hard/Soft Errors nieder-prior Hard/Soft Errors hochpri-or	Keine	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Mit diesem Parameter stellen Sie ein, ob Hardware-Diagnosemeldungen gemeldet werden sollen. • Keine: Neben der Standarddiagnose werden keine weiteren Alarme generiert. • Hard Errors: Schwerwiegende Hardware-Fehler werden über die S7-Diagnose gemeldet. • Hard/Soft Errors niederprior Schwerwiegende Hardware-Fehler und Fehler die während der Befehlsbearbeitung auftreten, werden über die S7-Diagnose gemeldet. Das "Ext_Diag"-Bit wird nicht gesetzt. • Hard/Soft Errors hochprior Schwerwiegende Hardware-Fehler und Fehler die während der Befehlsbearbeitung auftreten, werden über die S7-Diagnose gemeldet. Das "Ext_Diag"-Bit wird gesetzt.
Unterdrückung Error-LED	Keine Kanal 1 Kanal 2	Keine	Deaktivieren der Error-LED (ERR) eines Kanals. Das Kommunikationsmodul verfügt über zwei Kanäle, an die Reader / optischen Lesegeräte angeschlossen werden können. Wird nur einer der Kanäle verwendet, blinkt die Error-LED des anderen Kanals permanent. Mithilfe der Unterdrückung können Sie die Error-LED des nicht verwendeten Kanals deaktivieren.

Kommunikationsmodul ASM 475

Wissenswertes zu ASM 475

Einführung

Bis zu acht Anschaltungen ASM 475 können in einen Baugruppenträger der SIMATIC S7-300 gesteckt und betrieben werden. Bei einem Aufbau mit mehreren Baugruppenträgern (max. vier Baugruppenträger) lässt sich die ASM 475 in jedem Baugruppenträger stecken und betreiben. Im Maximalausbau einer SIMATIC S7-300 können somit bis zu 32 ASM 475 betrieben werden.

An das ASM 475 können max. zwei SLG (Schreib-/Lesegeräte) angeschlossen werden. Die Bearbeitung der angeschlossenen SLG erfolgt parallel. Der FC 45 ermöglicht die einfache Programmierung über die SIMATIC S7 Werkzeuge.

Der FC 45 kann sowohl in der S7-300 als auch in der S7-400 betrieben werden.

Bei der S7-400 erfolgt der Anschluss der ASM 475 über ET 200M.

Bitte beachten Sie, dass das

- IM 153-1 mindestens MLFB Nr. 6ES7 153-1AA03-0XB0 bzw. 6ES7 153-1AA83-0XB0 und
- IM 153-2 mindestens MLFB Nr. 6ES7 153-2AA02-0XB0 bzw. 6ES7 153-2AB01-0XB0

besitzen muss!

Der Zugriff auf die MDS Daten erfolgt über Normaladressierung.

Physikalische Adressierung des MDS (Mobile Datenspeicher)

Die physikalische Adressierung eines MDS wird auch als **Normaladressierung** bezeichnet. Der Anwender baut die Struktur seiner MDS selbst auf. Er weiß, auf welcher physikalischen MDS Adresse welche Daten geschrieben sind. Die Adressierung des MDS Speichers beginnt in der Regel bei Adresse 0000 hex und endet auf einer Endadresse die der Größe des MDS Speichers entspricht.

Filehandler Adressierung des MDS

Bei der Filehandler Adressierung gibt der Anwender für den Zugriff auf seine Daten einen Filenamen an. Der Filename besteht aus acht ASCII Zeichen. Der Filehandler verwaltet die Anwenderdaten auf dem MDS selbständig. Der Anwender braucht keine Datenstrukturen auf dem MDS zu projektieren.

Der MDS muss vor der Benutzung formatiert werden.

Kommunikationsmodul RF18xC

Kommunikationsmodul für Ident-Systeme - RF18xC

Die Kommunikationsmodule SIMATIC RF185C, RF186C und RF188C sind für den Einsatz in allen Bereichen der Automatisierung konzipiert. Die Baugruppe RF18xC ist ein Kommunikationsmodul für Ident-Systeme für PROFINET IO und OPC UA. Je nach Geräte-Typ können 1, 2 oder 4 Reader an das Kommunikationsmodul angeschlossen werden. Die Bearbeitung der angeschlossenen Reader erfolgt parallel.

Die Funktionsbausteine FB 45, FB 55 und der Programmbaustein für das Ident-Profil/RFID-Normprofil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge. Die Funktionsbausteine FB 45, FB 55 und der Programmbaustein für das Ident-Profil/RFID-Normprofil können sowohl in der S7-300 als auch in der S7-400 betrieben werden. Für die Steuerungen S7-1200 und S7-1500 muss das Ident-Profil verwendet werden.

Der Zugriff auf die Transponder-Daten erfolgt über Normaladressierung oder Filehandler-Adressierung.

Physikalische Adressierung des Transponder

Die physikalische Adressierung eines Transponder wird auch als Normaladressierung bezeichnet. Bei der Normaladressierung können Sie die Datenstruktur seiner Transponder selbst aufbauen. Dabei bestimmen Sie, auf welcher physikalischen Adresse welche Daten geschrieben werden. Die Adressierung des Transponder-Speichers beginnt in der Regel bei Adresse "0x0000" und endet auf einer Endadresse, die der Größe des Transponder-Speichers entspricht.

Parametergruppe "Web Based Management"

In dieser Parametergruppe können Sie das Web Based Management starten.

Tabelle 1-220 Parameter der Parametergruppe "Web Based Management"

Parameter	Beschreibung
Web Based Management	Web Based Management des Kommunikationsmoduls starten. Das Web Based Managment (WBM) bietet umfangreiche Funktionen, um das Kommunikationsmodul zu konfigurieren. Hinweis: Das WBM kann erst gestartet werden, wenn entweder die PROFINET-Verbindung zwischen CPU und Kommunikationsmodul aufgebaut ist oder dem Kommunikationsmodul die im Projekt hinterlegte IP-Adresse zugewiesen wurde. D. h. der Gerätename muss vergeben sein und die TIA-Projektierung muss in die SIMATIC-Steuerung geladen sein.

Parametergruppe "Konfigurationsmanagment"

In dieser Parametergruppe können Sie Konfigurationsdaten laden oder speichern.

Tabelle 1-221 Parameter der Parametergruppe "Konfigurationsmanagement"

Parameter	Beschreibung
Benutzername	Benutzername eines auf dem Kommunikationsmodul angelegten Benutzers Beachten Sie, dass der Benutzer über die benötigten Rechte verfügen muss.
Passwort	Eingabe des Passworts für den Benutzer
Konfiguration in Gerät laden	Konfigurationsdaten vom STEP 7-Projekt in das Kommunikationsmodul laden.
Konfiguration im Projekt speichern	Konfigurationsdaten des Kommunikationsmoduls im aktuellen STEP 7-Projekt speichern.

Voraussetzung

Folgende Voraussetzungen müssen erfüllt sein, damit Konfigurationsdaten geladen oder gespeichert werden können:

- In dem Parameter "PROFINET-Schnittstelle [X1]" ist die korrekte IP-Adresse des Kommunikationsmoduls eingetragen.
- Der Benutzername und das dazugehöriges Passwort sind im WBM angelegt.
- Der eingetragene Benutzer hat die nötigen Rechte, um den Down-/Upload durchzuführen.

Hinweis

Benutzername und Passwort nur bei aktiver Benutzerverwaltung nötig

Die Textfelder "Benutzername" und "Passwort" müssen nur ausgefüllt werden, wenn die Benutzerverwaltung des WBM aktiviert ist.

Parametergruppe "Baugruppenparameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter des Kommunikationsmoduls projektieren.

Tabelle 1-222 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
Diagnosealarm	Aus An	Aus	Diagnosemeldungen des Kommunikationsmoduls ein-/ausschalten.

Parametergruppe "Baugruppenparameter" der angeschlossenen Reader

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter der angeschlossenen Reader projektieren.

Tabelle 1-223 Parameter der Parametergruppe "Baugruppenparameter" der angeschlossenen Reader

Parameter	Parameterwert	Default-Wert	Beschreibung
User Mode	RFID-Normprofil FB 45	RFID-Normprofil	Mit diesem Parameter wählen Sie den Baustein aus: • RFID-Normprofil: In der Steuerung kommt der Programmbaustein für das Ident-Profil zum Einsatz. • FB 45: Singletag-Betrieb. In der Steuerung wird der FB 45 (PROFIBUS/PROFINET) eingesetzt.
MOBY Mode	RF200/RF300/RF600; MOBY U/D-Normaladr.	RF200/RF300/ RF600; MOBY U/D-Normaladr.	Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein. • RF200/RF300/RF600; MOBY U/D-Normaladr. Normaladressierung: Der Transponder wird mit physikalischen Adressen angesprochen.
Übertragungs-geschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Ident-System. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.
Diagnose-meldungen	Keine Hard Errors Hard/Soft Errors	Keine	Mit diesem Parameter stellen Sie ein, in welchem Umfang readerbezogene Diagnosealarmmeldungen gemeldet werden sollen. • Keine: Es werden keine weiteren Alarme generiert. • Hard Errors: Schwerwiegende Hardware-Fehler werden über die S7-Diagnose gemeldet. • Hard/Soft Errors: Schwerwiegende Hardware-Fehler, sowie Fehler, die während der Befehlsbearbeitung auftreten, werden über die S7-Diagnose gemeldet.

Reader

Reader RF680R/RF685R

Reader RF680R/RF685R für den UHF-Bereich

Die UHF-Reader SIMATIC RF680R und RF685R sind für den Einsatz in der Logistik und in der Automatisierung konzipiert z. B. in einer Produktionslinie vorgesehen, aber genauso für die Anwendungen im Logistikbereich geeignet. Um diesen Anforderungen gerecht zu werden, wurden die Reader mit einer hohen Sendeleistung, sowie Schutzart (IP65) ausgestattet. Die Reader sind mit umfangreichen Diagnosemöglichkeiten ausgestattet und können ISO 18000-6C- und ISO 18000-6B-Transponder bearbeiten.

Eine Besonderheit bietet der RF685R mit der internen, adaptiven Antenne. Sie erhöht signifikant die Zuverlässigkeit von Lese- und Schreibvorgängen auch unter schwierigen Funkbedingungen.

Die Reader RF680R und RF685R werden entweder über einen integrierten PROFINET-Anschluss oder über die RS422-Schnittstelle und das Kommunikationsmodul ASM 456 über PROFIBUS problemlos in SIMATIC S7-Automatisierungssysteme integriert. Entsprechende Programmierbausteine stehen zur Verfügung. Eine zweite Ethernet-Schnittstelle (beide M12) kann zur Diagnose während des Betriebs genutzt werden, sodass zum überlagerten System keine Unterbrechung notwendig ist.

Das WBM (Web Based Management) ermöglicht die Inbetriebnahme, Konfiguration und Diagnose der Geräte per Internet-Browser. Zusätzliche Updates und Installationen von Konfigurations- und Diagnosesoftware entfallen.

Parametergruppe "Web Based Management"

In dieser Parametergruppe können Sie das Web Based Management starten.

Tabelle 1-224 Parameter der Parametergruppe "Web Based Management"

Parameter	Beschreibung
Web Based Management (WBM)	Web Based Management des Readers starten. Das Web Based Managment bietet umfangreiche Funktionen, um die Reader zu konfigurieren. Hinweis: Das WBM kann erst gestartet werden, wenn entweder die PROFINET-Verbindung zwischen CPU und Reader aufgebaut ist oder dem Reader die im Projekt hinterlegte IP-Adresse zugewiesen wurde. D. h. der Gerätename muss vergeben sein und die TIA-Projektierung muss in die SIMATIC-Steuerung geladen sein.

Parametergruppe "Baugruppenparameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter der Reader RF680R/RF685R projektieren.

Tabelle 1-225 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
RFID Lesestellen-Alarm	Aus An	Aus	Lesestellebezogene Diagnosemeldungen ein-/ausschalten.

Parametergruppe "Konfigurationsmanagment"

In dieser Parametergruppe können Sie Konfigurationsdaten laden oder speichern.

Tabelle 1-226 Parameter der Parametergruppe "Konfigurationsmanagement"

Parameter	Beschreibung
Benutzername	Benutzername eines auf dem Reader angelegten Benutzers Beachten Sie, dass der Benutzer über die benötigten Rechte verfügen muss.
Passwort	Eingabe des Passworts für den Benutzer
Konfiguration in Gerät laden	Konfigurationsdaten vom STEP 7-Projekt in den Reader laden.
Konfiguration im Projekt speichern	Konfigurationsdaten des Readers im aktuellen STEP 7-Projekt speichern.

Voraussetzung

Folgende Voraussetzungen müssen erfüllt sein, damit Konfigurationsdaten geladen oder gespeichert werden können:

- In dem Parameter "PROFINET-Schnittstelle [X1]" ist die korrekte IP-Adresse des Readers eingetragen.
- Der Benutzername und das dazugehöriges Passwort sind im WBM angelegt.
- Der eingetragene Benutzer hat die nötigen Rechte, um den Down-/Upload durchzuführen.

Hinweis

Benutzername und Passwort nur bei aktiver Benutzerverwaltung nötig

Die Textfelder "Benutzername" und "Passwort" müssen nur ausgefüllt werden, wenn die Benutzerverwaltung des WBM aktiviert ist.

Parametergruppe "RFID-Kommunikation"

In dieser Parametergruppe können Sie den Adressbereich der RFID-Kommunikation projektieren.

Tabelle 1-227 Parameter der Parametergruppe "RFID-Kommunikation"

Parameter	Beschreibung
Allgemein	Allgemeine Einstellungen Eingabefelder für den Namen der Kommunikation und für einen Kommentar
E/A-Adressen	Festlegung der E/A-Adresse des Readers ("LADDR") Dieser Parameter wird in der Variablen "IID_HW_CONNECT" verwendet.
HW-Kennung	Festlegung der HW-Kennung des Readers ("HW-ID") Dieser Parameter wird in der Variablen "IID_HW_CONNECT" verwendet.

Parametergruppe "Digitaleingänge/-ausgänge"

In dieser Parametergruppe können Sie den Adressbereich der Digitaleingänge/-ausgänge projektieren.

Tabelle 1-228 Parameter der Parametergruppe "Digitaleingänge/-ausgänge"

Parameter	Beschreibung
Allgemein	Allgemeine Einstellungen Eingabefelder für den Namen der Digitaleingänge/-ausgänge und für einen Kommentar
E/A-Adressen	Festlegung der E/A-Adresse der digitalen Ein-/Ausgänge Über den eingestellten Adressbereich (E/A-Adresse) können die Digitaleingänge/-ausgänge, welche im WBM des Readers konfiguriert wurden, angesprochen werden.
HW-Kennung	Festlegung der HW-Kennung der Digitaleingänge/-ausgänge

Reader RF615R

Reader RF615R für den UHF-Bereich

Die UHF-Reader SIMATIC RF615R sind für den Einsatz in der Logistik und in der Automatisierung konzipiert z. B. in einer Produktionslinie vorgesehen, aber genauso für die Anwendungen im Logistikbereich geeignet. Um diesen Anforderungen gerecht zu werden, wurden die Reader mit einer hohen Sendeleistung, sowie Schutzart (IP65) ausgestattet. Die Reader sind mit umfangreichen Diagnosemöglichkeiten ausgestattet und können ISO 18000-6C- und ISO 18000-6B-Transponder bearbeiten.

Der Reader RF615R wird entweder über einen integrierten PROFINET-Anschluss oder über die RS422-Schnittstelle und das Kommunikationsmodul ASM 456 über PROFIBUS problemlos in SIMATIC S7-Automatisierungssysteme integriert. Entsprechende Programmierbausteine stehen zur Verfügung.

Das WBM (Web Based Management) ermöglicht die Inbetriebnahme, Konfiguration und Diagnose der Geräte per Internet-Browser. Zusätzliche Updates und Installationen von Konfigurations- und Diagnosesoftware entfallen.

Parametergruppe "Baugruppenparameter"

In dieser Parametergruppe können Sie alle baugruppen-spezifischen Parameter der Readers RF615R projektieren.

Tabelle 1-229 Parameter der Parametergruppe "Baugruppenparameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
RFID Lesestellen-Alarm	Aus An	Aus	Lesestellebezogene Diagnosemeldungen ein-/ausschalten.

1.6.2.2 Optische Lesesysteme

Optische Lesegeräte MV400

Optische Lesegeräte MV400

Die optischen Lesegeräte MV400 wurden speziell zur Erkennung und Auswertung einer Vielzahl von maschinenlesbaren Codes, Klarschrift sowie von Objekten in der industriellen Produktion konzipiert.

Die Lesegeräte können sowohl in den Steuerungen S7-300, S7-400, S7-1200 und S7-1500 sowie mit verschiedenen Fremdsteuerungen betrieben werden. Die Anbindung an die Steuerung erfolgt via PROFINET IO (FB79/Ident-Profil), TCP, RS232 oder über Kommunikationsmodule.

Die Funktionsbausteine FB79 und Ident-Profil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge.

Parametergruppe "Betriebsart"

In dieser Parametergruppe können Sie die Betriebsart der optischen Lesegeräte MV400 projektieren.

Tabelle 1-230 Parameter der Parametergruppe "Betriebsart"

Parameter	Parameterwert	Default-Wert	Beschreibung
Funktionsbaustein	FB79 Ident-Profil	FB79	Mit diesem Parameter wählen Sie den Baustein aus: • FB79 Projektierung der optischen Lesegeräte mit Hilfe des Bausteins "FB79". Kompatibel zu dem Lesegerät "VS130-2"; Kompatible Steuerungen: S7-300 und S7-400 • Ident-Profil: Projektierung der optischen Lesegeräte mit Hilfe der Ident-Bausteine. Kompatibel zu allen Ident-Geräten; Kompatible Steuerungen: S7-300, S7-400, S7-1200 und S7-1500

Optische Lesegeräte MV500

Optische Lesegeräte MV500

Die optischen Lesegeräte MV500 wurden speziell zur Erkennung und Auswertung einer Vielzahl von maschinenlesbaren Codes, Klarschrift sowie von Objekten in der industriellen Produktion konzipiert.

Die Lesegeräte können sowohl in den Steuerungen S7-300, S7-400, S7-1200 und S7-1500 sowie mit verschiedenen Fremdsteuerungen betrieben werden. Die Anbindung an die Steuerung erfolgt via PROFINET IO (FB79/Ident-Profil), TCP, RS232 oder über Kommunikationsmodule.

Die Funktionsbausteine FB79 und Ident-Profil ermöglichen die einfache Programmierung über die SIMATIC S7-Werkzeuge.

Parametergruppe "Betriebsart"

In dieser Parametergruppe können Sie die Betriebsart der optischen Lesegeräte MV500 projektieren.

Tabelle 1-231 Parameter der Parametergruppe "Betriebsart"

Parameter	Parameterwert	Default-Wert	Beschreibung
Funktionsbaustein	FB79 Ident-Profil	FB79	Mit diesem Parameter wählen Sie den Baustein aus: • FB79 Projektierung der optischen Lesegeräte mit Hilfe des Bausteins "FB79". Kompatibel zu den Lesegeräten "VS130-2" und "MV400"; Kompatible Steuerungen: S7-300 und S7-400 • Ident-Profil: Projektierung der optischen Lesegeräte mit Hilfe der Ident-Bausteine. Kompatibel zu allen Ident-Geräten; Kompatible Steuerungen: S7-300, S7-400, S7-1200 und S7-1500

Parametergruppe "Web Based Management"

In dieser Parametergruppe können Sie das Web Based Management starten.

Tabelle 1-232 Parameter der Parametergruppe "Web Based Management"

Parameter	Beschreibung
Web Based Management	Web Based Management des Lesegeräts starten. Das Web Based Managment (WBM) bietet umfangreiche Funktionen, um das Lesegerät zu konfigurieren. Hinweis: Das WBM kann erst gestartet werden, wenn entweder die PROFINET-Verbindung zwischen CPU und Lesegerät aufgebaut ist oder dem Lesegerät die im Projekt hinterlegte IP-Adresse zugewiesen wurde. D. h. der Gerätename muss vergeben sein und die TIA-Projektierung muss in die SIMATIC-Steuerung geladen sein.

1.6.3 Dezentrale Peripherie

1.6.3.1 Dezentrale Peripheriesysteme

SIMATIC ET 200 - Für jede Anwendung die richtige Lösung

Mit SIMATIC ET 200 stehen unterschiedlichste dezentrale Peripheriesysteme zur Auswahl.

- Lösungen zum Einsatz im Schaltschrank
- Lösungen ohne Schaltschrank direkt an der Maschine

Weiterhin gibt es Komponenten für den Einsatz im explosionsgefährdeten Bereich. SIMATIC ET 200-Systeme für den schaltschranklosen Aufbau sind in einem robusten, glasfaserverstärkten Kunststoffgehäuse untergebracht und daher schlagfest, schmutzresistent und wasserdicht.

Der modulare Aufbau erlaubt es, die ET 200-Systeme einfach und in kleinen Schritten zu skalieren und zu erweitern. Fertig integrierte Zusatzmodule senken die Kosten und bieten gleichzeitig breit gefächerte Anwendungsmöglichkeiten. Dabei stehen unterschiedlichste Kombinationsmöglichkeiten zur Auswahl:

- digitale und analoge Ein-/Ausgänge
- intelligente Module mit CPU-Funktionalität,
- Sicherheitstechnik,
- Pneumatik,
- Frequenzumrichter
- sowie verschiedene Technologiemodule.

Die Kommunikation über PROFIBUS und PROFINET, das einheitliche Engineering, transparente Diagnosemöglichkeiten sowie die optimale Anbindung an SIMATIC Controller und HMI-Geräte belegen die einzigartige Durchgängigkeit von Totally Integrated Automation.

Die folgende Tabelle zeigt eine Übersicht der Peripheriegeräte für den Einsatz im Schaltschrank:

Peripheriegerät	Eigenschaften
ET 200L	• Kostengünstige digitale Blockperipherie • Digitale Elektronikblöcke bis 32 Kanäle
ET 200M	• Modularer Aufbau mit Standard-Baugruppen der SIMATIC-S7-300 • Fehlersichere E/A-Baugruppen • Einsatz in explosionsgefährdeten Bereich bis in Zone 2, Sensoren und Aktoren bis in Zone 1 • Hohe Anlagenverfügbarkeit, z. B. durch Ziehen und Stecken im laufenden Betrieb

Peripheriegerät	Eigenschaften
ET 200MP	• Modularer Aufbau mit Standard-Baugruppen der SIMATIC-S7-1500 • Taktsynchroner PROFINET IO mit Sendetakten bis zu 250 µs • IO-Device mit PROFINET IO-Schnittstelle • DP-Slave mit PROFIBUS DP-Schnittstelle • Technologiemodule für Zählen, Positionieren, Time-based IO
ET 200S	• Feinmodularer Aufbau mit Mehrleiteranschluss • Multifunktional durch breites Modulspektrum • I-Device • Einsatz im explosionsgefährdeten Bereich (Zone 2)
ET 200S COMPACT	• Feinmodularer Aufbau mit Mehrleiteranschluss • Multifunktional durch breites Modulspektrum • Einsatz im explosionsgefährdeten Bereich (Zone 2) • Integrierte DE/DA
ET 200SP	• Kompakte Module • Stehende Verdrahtung mit Ein- oder Mehrleiteranschluss • PROFINET-Schnittstelle mit 3 Ports • Taktsynchroner PROFINET IO mit den Profilen PROFIsafe und PROFIenergy • I-Device • Module für Zählen, Positionieren, Wiegen und Messen • Fehlersichere CPUs und Module
ET 200iSP	• Modularer Aufbau, auch redundant möglich • Robust und eigensichere Bauweise • Einsatz im explosionsgefährdeten Bereich bis in Zone 1/21, Sensoren und Aktoren dürfen sogar in Zone 0/20 liegen • Hohe Anlagenverfügbarkeit, z. B. durch Ziehen und Stecken im laufenden Betrieb

Die folgende Tabelle zeigt eine Übersicht der Peripheriegeräte für den Einsatz ohne Schaltschrank:

Peripheriegerät	Eigenschaften
ET 200AL	• Einfache Montage in allen Einbaulagen, auf engstem Raum • Anbindung an PROFINET IO, PROFIBUS DP oder ET 200SP • Hohe Schutzart IP65/IP67 Für Temperaturen von -25 °C bis +55 °C, Beschleunigungen bis 5 g • Anbindung von Sensoren und Aktoren über M8- und M12-Anschlusstechnik
ET 200eco PN	• Kostengünstige, Platz sparende Blockperipherie • Digitale Module mit bis 16 Kanälen (auch parametrierbar) • Analoge Module, IO-Link Master und Lastspannungsverteiler • PROFINET-Anschluss mit 2-Port-Switch in jedem Modul • Über PROFINET flexibel in Linien- und/oder Sternstruktur direkt in der Anlage verteilbar
ET 200eco	• Kostengünstige digitale Blockperipherie • Flexible Anschlussmöglichkeiten • Fehlersichere Module • Hohe Anlagenverfügbarkeit
ET 200pro	• Modularer Aufbau mit kompakten Gehäuse • Einfache Montage • Multifunktional durch breites Modulspektrum • I-Device • Hohe Anlagenverfügbarkeit durch Ziehen und Stecken im laufenden Betrieb und stehende Verdrahtung • Umfangreiche Diagnose
ET 200R	• Speziell für den Einsatz an Robotern • Montage direkt am Chassis • Resistent gegen Schweißspritzer durch widerstandsfähiges Metallgehäuse

Siehe auch

Dokumentation zu ET 200L (http://support.automation.siemens.com/WW/view/de/1142908)

Dokumentation zu ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)

Dokumentation zu ET 200M (http://support.automation.siemens.com/WW/view/de/1142798)

Dokumentation zu ET 200pro (http://support.automation.siemens.com/WW/view/de/21210852)

Dokumentation zu ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789)

Dokumentation zu ET 200R (http://support.automation.siemens.com/WW/view/de/11966255)

Dokumentation zu ET 200eco PN (http://support.automation.siemens.com/WW/view/de/29999018)

Dokumentation zu ET 200eco (http://support.automation.siemens.com/WW/view/de/12403834)

Dokumentation zu ET 200SP (http://support.automation.siemens.com/WW/view/de/84133942)

Dokumentation zu ET 200MP (http://support.automation.siemens.com/WW/view/de/86140384)

Dokumentation zu ET 200AL (http://support.automation.siemens.com/WW/view/de/95242965)

1.6.3.2 Projektieren von HART-Variablen

Einleitung

Zahlreiche HART-Feldgeräte stellen zusätzliche Messgrößen (z. B. Sensortemperatur) zur Verfügung. Diese können ausgelesen werden, wenn sie in der Feldgeräte-Projektierung entsprechend eingestellt sind. Über die HART-Variablen ist es möglich, die eingestellten Messwerte direkt vom Feldgerät in den E/A-Bereich ihres Automatisierungsystems zu übernehmen.

Bei den HART-Baugruppen können unabhängig von der Anzahl der projektierten Kanäle max. 8 HART-Variablen parametriert werden, pro Kanal max. 4 HART-Variablen. Die HART-Variablen ordnen Sie in den Eigenschaften der Baugruppe einem Kanal zu (Bereich "HART-Variableneinstellung). Beachten Sie dazu das Handbuch der entsprechenden Baugruppe.

Adressbelegung

Die HART-Baugruppen belegen standardmäßig 16 Ein-/Ausgangsbyte (Nutzdaten). Wenn Sie HART-Variablen projektieren, belegen die Baugruppen für jede HART-Variable zusätzlich 5 Byte Eingangsdaten.

Wenn Sie alle 8 HART-Variablen nutzen, belegt die HART-Eingabebaugruppe insgesamt 56 Ein-/Ausgangsbytes (16 Byte + 8 x 5 Byte = 56 Byte).

Die Projektierung "keine" belegt keine zusätzlichen Eingangsbyte.

Projektierung von HART-Variablen

Für einen Kanal können Sie bis zu 4 HART-Variablen projektieren

- PV (Primary Variable)
- SV (Secondary Variable)
- TV (Tertiary Variable)
- QV (Quaternary Variable)

CiR ist ein Platzhalter und reserviert den Adressraum für eine HART-Variable. HART-Variablen die Sie nicht nutzen, müssen Sie mit dem Parameter "keine" projektieren.

Aufbau der HART-Variablen

Jede HART-Variable belegt 5 Byte Eingangsdaten und ist wie folgt aufgebaut:

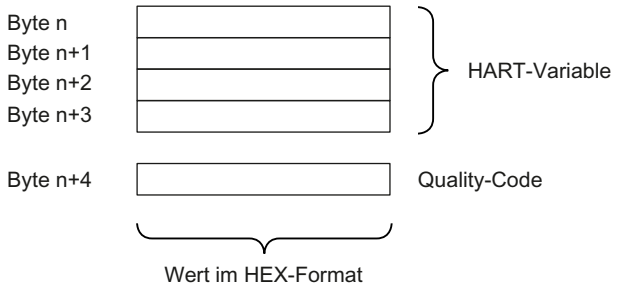

Je nach Baugruppe kann der Quality-Code unterschiedliche Werte annehmen. Beachten Sie dazu das Handbuch der entsprechenden Baugruppe.

Siehe auch

Dokumentation zu HART-Analogbaugruppen (http://support.automation.siemens.com/WW/view/de/22063748)

1.6.3.3 ET 200iSP

Dezentrales Peripheriegerät ET 200iSP

Definition

Das Dezentrale Peripheriegerät ET 200iSP ist ein feinmodularer und eigensicherer DP-Slave mit Schutzart IP 30.

Einsatzgebiet

Das Dezentrale Peripheriegerät ET 200iSP ist einsetzbar in explosionsgefährdeten Bereichen mit einer Gas- und Staubatmosphäre:

Zulassung	ET 200iSP Station*	Ein- und Ausgänge
ATEX	Zone 1, Zone 21	bis in Zone 0, Zone 20 **
IECEx	Zone 2, Zone 22	bis in Zone 0, Zone 20 **
* in Verbindung mit einem entsprechenden Gehäuse ** beim Elektronikmodul 2 DO Relay UC60V/2A: bis in Zone 1; Zone 21		

Selbstverständlich ist das Dezentrale Peripheriegerät ET 200iSP auch im sicheren Bereich einsetzbar.

Direkt neben dem Interfacemodul, das die Daten an den DP-Master überträgt, können Sie Peripheriemodule der ET 200iSP in nahezu beliebiger Kombination stecken. Somit können Sie den Ausbau auf den jeweiligen Bedarf vor Ort ausrichten.

Jede ET 200iSP besteht aus einem Power Supply, einem Interfacemodul und maximal 32 Elektronikmodulen (z.B. Digitale Elektronikmodule). Beachten Sie die maximale Stromaufnahme.

Terminal- und Elektronikmodule

Das Peripheriegerät ET 200iSP besteht prinzipiell aus verschiedenen passiven Terminalmodulen, auf denen Sie die das Power Supply und die Elektronikmodule stecken.

Die ET 200iSP wird über einen Anschlussstecker am Terminalmodul TM-IM/EM an den PROFIBUS RS 485-IS angeschlossen. Jede ET 200iSP ist ein DP-Slave am PROFIBUS RS 485-IS.

DP-Master

Alle Module der ET 200iSP können mit allen DP-Mastern kommunizieren, die sich nach der Norm *IEC 61784-1:2002 Ed1 CP 3/1* mit dem Übertragungsprotokoll "DP" (DP steht für Dezentrale Peripherie) verhalten.

Siehe auch

Dokumentation zur ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789)

Kanal und IEEE-Variable zuordnen

Eigenschaften

Die Analogen Elektronikmodule 4 AI I 2WIRE/HART, 4 AI I 4WIRE/ HART und 4 AO I HART unterstützen bis zu 4 IEEE-Variablen.

Im Prozessabbild der Eingänge (PAE) stehen pro Modul maximal 20 Byte für die IEEE-Variablen zur Verfügung. Für die 4 IEEE-Variablen sind somit 4 Blöcke mit jeweils 5 Byte innerhalb des PAE vorhanden.

Voraussetzungen

Das HART-Feldgerät muss die parametrierte Anzahl der IEEE-Variablen unterstützen.

IEEE-Variablen zuordnen

Sie ordnen die IEEE-Variablen der Feldgeräte beliebig den 4 Blöcken im PAE zu.

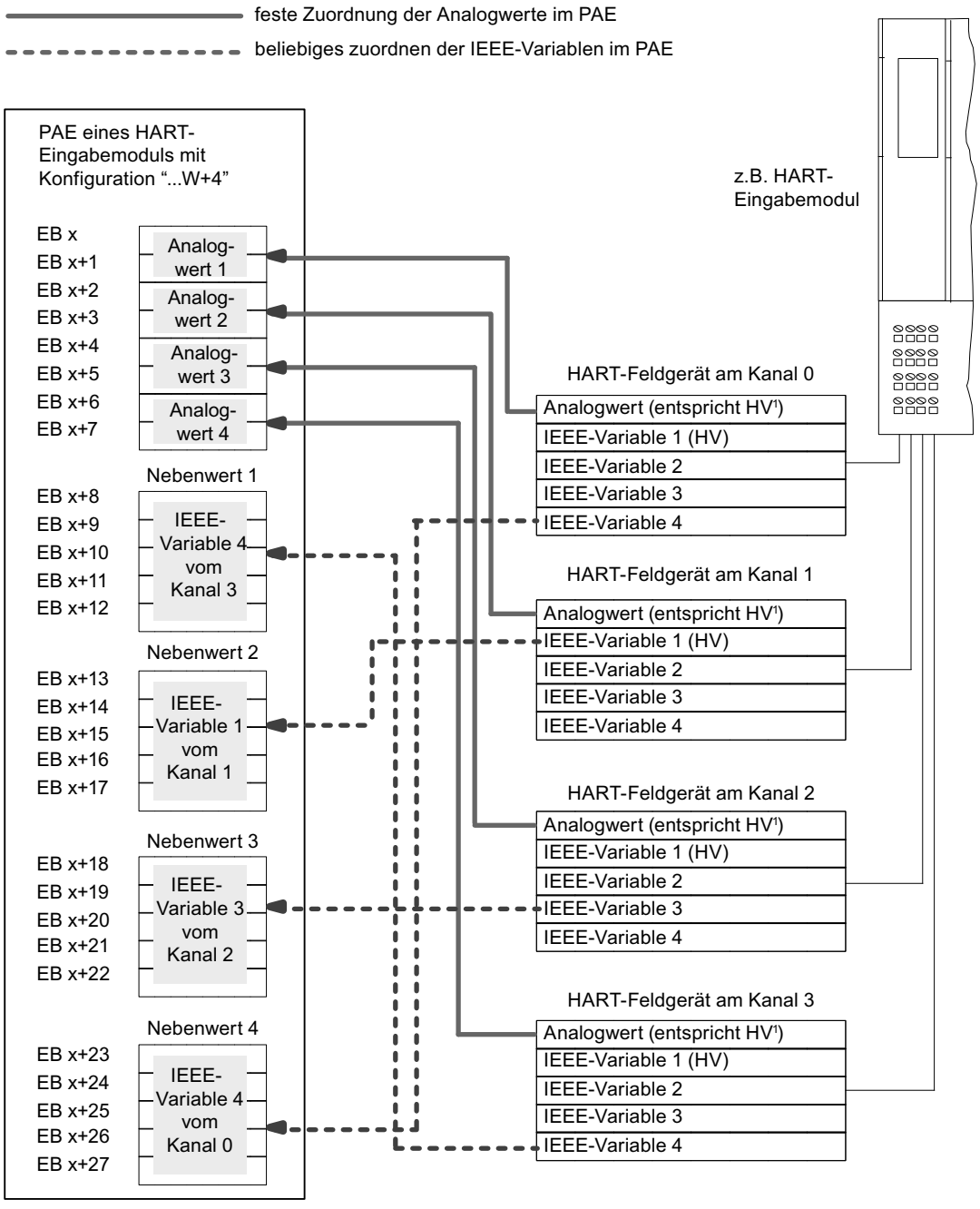

[1] HV=Hauptvariable

Siehe auch

Dokumentation zur ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789)

Vergleichsstellen parametrieren für Thermoelemente

Kompensation der Vergleichsstellentemperatur

Sie haben verschiedene Möglichkeiten, die Vergleichsstellentemperatur zu erfassen, um aus der Temperaturdifferenz zwischen Vergleichsstelle und Messstelle einen absoluten Temperaturwert zu erhalten.

Tabelle 1-233 Kompensation der Vergleichsstellentemperatur

Möglichkeit	Erläuterung	Parameter Vergleichsstelle
Keine Kompensation	Sie erfassen nicht nur die Temperatur der Messstelle. Die Temperatur der Vergleichsstelle (Übergang von Cu-Leitung auf Thermoelementeleitung) beeinflusst zusätzlich die Thermospannung. Der Messwert ist somit fehlerbehaftet.	keine
Verwendung eines Widerstandsthermometers Pt100 Klimabereich zum Erfassen der Vergleichsstellentemperatur (günstigste Methode)	Sie können die Vergleichsstellentemperatur mittels eines Widerstandsthermometers (Pt100 Klimabereich) erfassen. Dieser Temperaturwert wird in der ET 200iSP bei entsprechender Parametrierung an die 4 AI TC-Module verteilt und in den Modulen mit den ermittelten Temperaturwert der Messstelle verrechnet. Anzahl der Vergleichsstellen: 2	Die Parametrierung der IM152 und des 4 AI TC muss abgestimmt sein: • 4 AI RTD parametriert auf Pt100 Klimabereich am richtigen Steckplatz; • 4 AI TC: Vergleichsstelle : "ja"; Vergleichsstellennummer "1" oder "2" auswählen • IM 152-1:Zuweisung der Vergleichsstelle auf einen Steckplatz mit 4 AI RTD; Auswahl eines Kanals;
Interne Kompensation 4 AI TC	Auf die Klemmen des Terminalmoduls des EM 4 AI TC wird das TC-Sensormodul (Temperatursensor) montiert. Der Temperatursensor meldet die Klemmentemperatur an das 4 AI TC. Dieser Wert wird mit dem gemessenen Wert aus dem Kanal des Elektronikmoduls verrechnet.	• 4 AI TC: Vergleichsstellennummer "Intern"

Kompensation durch Widerstandsthermometer am 4 AI RTD

Wenn Thermoelemente, die an die Eingänge der 4 AI TC angeschlossen sind, dieselbe Vergleichsstelle haben, kompensieren Sie durch ein 4 AI RTD.

Für die Kanäle des Moduls 4 AI TC können Sie als Vergleichsstellennummer "1", "2" oder "Intern" wählen. Wenn Sie "1" oder "2" wählen, dann wird für die 4 Kanäle immer dieselbe Vergleichsstelle (RTD-Kanal) verwendet.

Parametrierung der Vergleichsstelle

Die Vergleichsstellen für die Elektronikmodule 4 AI TC stellen Sie über folgende Parameter ein:

Tabelle 1-234 Parameter der Vergleichsstelle

Parameter	Modul	Wertebereich	Erläuterung
Steckplatz Vergleichsstelle 1 bis Steckplatz 2	IM 152	keinen, 4 bis 35	Mit diesem Parameter können Sie bis zu 2 Steckplätze (keinen, 4 bis 35) zuordnen, auf denen sich die Kanäle zur Vergleichstemperaturmessung (Ermittlung des Kompensationswertes) befinden.
Eingang Vergleichsstelle Vergleichsstelle 1 bis 4 Eingang Vergleichsstelle	IM 152	RTD an Kanal 0 RTD an Kanal 1 RTD an Kanal 2 RTD an Kanal 3	Mit diesem Parameter legen Sie den Kanal (0/1/2/3) zur Vergleichstemperaturmessung (Ermittlung des Kompensationswertes) für den zugeordneten Steckplatz fest.
Vergleichsstelle E0 bis Vergleichsstelle E3	4 AI TC	keine ja	Über diesen Parameter können Sie die Verwendung der Vergleichsstelle freigeben.
Vergleichsstellennummer	4 AI TC	1 2 Intern	Mit diesem Parameter weisen Sie die Vergleichsstelle (1, 2) zu, die die Vergleichstemperatur (Kompensationswert) enthält.

Siehe auch

Dokumentation zur ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789)

Grundlagen der Zeitstempelung

Eigenschaften

Zeitstempelung mit der IM 152 ist möglich in Kundenapplikationen mit Nutzung des FB 62 (FB TIMESTMP).

Funktionsweise

Ein geändertes Eingangssignal wird mit einem Zeitstempel versehen und in einen Puffer (Datensatz) abgelegt. Wenn zeitgestempelte Signale vorliegen oder ein Datensatz voll ist, wird ein Prozessalarm zum DP-Master generiert. Der Puffer wird mit "Datensatz lesen" ausgewertet. Bei Ereignissen, die die Zeitstempelung beeinflussen (Kommunikation mit DP-Master unterbrochen, Telegrammausfall vom Uhrzeitmaster, ...) werden Sondermeldungen generiert.

Parametrieren

Mit der Parametrierung legen Sie fest, welche Nutzdaten der IM 152 überwacht werden. Für die Zeitstempelung sind das Digitaleingänge, die auf Signaländerungen überwacht werden.

Parameter	Einstellung	Beschreibung
Zeitstempelung	• gesperrt • freigegeben	Aktivieren Sie die Zeitstempelung für die Kanäle des Elektronikmoduls 8 DI NAMUR.
Flankenauswertung kommendes Ereignis	• steigende Flanke • fallende Flanke	Legen Sie die Art der Signaländerung fest, die zeitgestempelt wird.

Zählen

Eigenschaften Zählen

Zählfunktionen

Das Elektronikmodul 8 DI NAMUR ist mit parametrierbaren Zählfunktionen ausgestattet:

- 2 16-Bit Vorwärtszähler (Normale Zählfunktion) oder
- 2 16-Bit Rückwärtszähler (Periodische Zählfunktion) oder
- 1 32-Bit Rückwärtszähler (Kaskadierzählfunktion)
- Vorgabe eines Sollwertes durch das PAA
- TOR-Funktion
- Die Funktion der Steuersignale der Zähler können Sie konfigurieren:
  - Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "DI": Es werden 2 Zähler konfiguriert. Die Steuersignale der Zähler sind im PAA (Prozessabbild der Ausgänge) hinterlegt.
  - Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "Control": Es werden 2 Zähler konfiguriert. Die Steuersignale der Zähler sind im PAA hinterlegt. Zusätzlich werden Sie von den Digitaleingängen des 8 DI NAMUR gesteuert.

Siehe auch

Funktionsweise (Seite 3367)

Zähler konfigurieren (Seite 3370)

Zähler parametrieren (Seite 3372)

Funktionsweise

16-Bit Vorwärtszähler (Normale Zählfunktion)

Der Zählbereich beträgt 0 bis 65535.

Mit jedem Zählimpuls am Digitaleingang erhöht sich der Zählerstand um den Wert 1. Mit Erreichen der Zählgrenze wird der Zählerstand auf den Wert 0 gesetzt und von diesem Wert wieder hochgezählt.

Bei einem Überlauf des Zählers wird der zugehörige Ausgang im PAE gesetzt.

Durch eine positive Flanke des Steuersignals *Rücksetze Ausgang* wird der Ausgang im PAE zurückgesetzt. Der aktuelle Zählwert wird dadurch nicht beeinflusst.

Beim 16-Bit Vorwärtszählen werden keine Ausgänge im PAA gesetzt. Diese sind generell zurückgesetzt.

Die positive Flanke des Steuersignals *Rücksetze Zähler* setzt den Zählerstand auf den Wert 0 und den gesetzten Zählerausgang zurück.

Das Steuersignal *TOR* hält den Zählvorgang bei einer positiven Flanke an. Erst bei einer negativen Flanke werden wieder Zählimpulse am Digitaleingang verarbeitet. Das Steuersignal *Rücksetze Zähler* ist auch bei aktiviertem *TOR* wirksam.

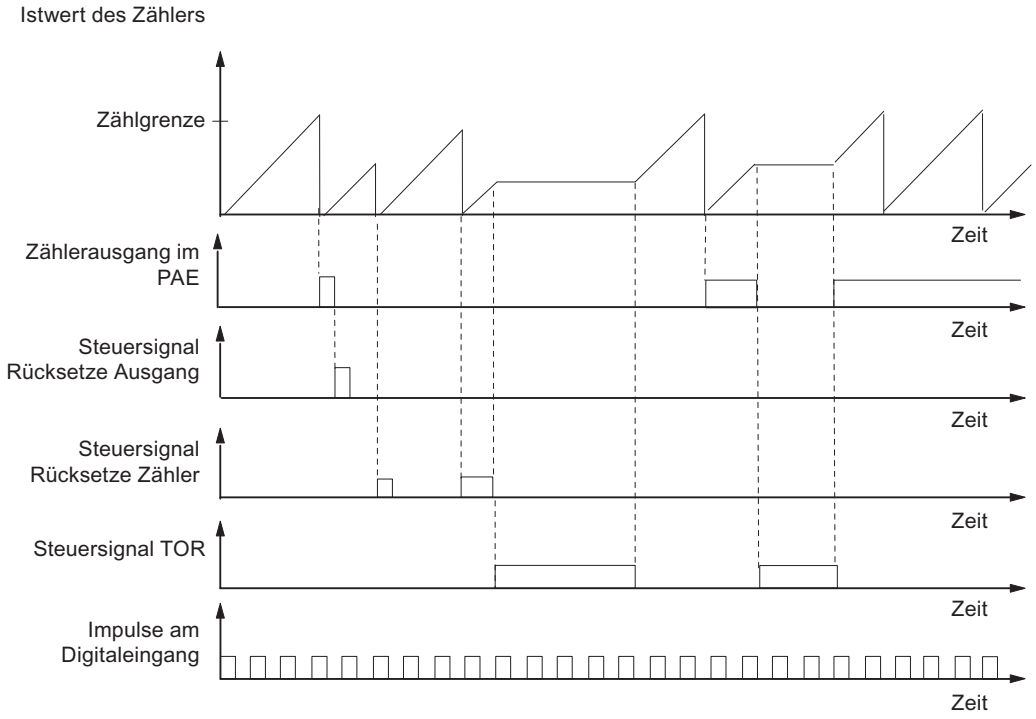

16-Bit Rückwärtszähler (Periodische Zählfunktion)

Der maximale Zählbereich beträgt 65535 bis 0.

Zum Startzeitpunkt des Zählers wird der Istwert auf den vorgegebenen Sollwert gesetzt. Mit jedem Zählimpuls reduziert sich der Istwert um den Wert 1. Erreicht der Istwert den Wert 0, dann wird der zugehörige Ausgang im PAE eingeschaltet und der Istwert wieder auf den vorgegebenen Sollwert gesetzt. Von diesem Wert wird wieder zurückgezählt.

Die positive Flanke des Steuersignals *Rücksetze Zähler* setzt den Istwert auf den vorgegebenen Sollwert und den zugehörigen Ausgang im PAE zurück.

Durch eine positive Flanke des Steuersignals *Rücksetze Ausgang* wird der Ausgang im PAE zurückgesetzt. Der aktuelle Zählwert wird dadurch nicht beeinflußt.

Das Steuersignal *TOR* hält den Zählvorgang bei einer positiven Flanke an. Gleichzeitig wird der zugeordnete Ausgang im PAE zurückgesetzt. Erst bei einer negativen Flanke werden wieder Zählimpulse am Digitaleingang verarbeitet. Die Steuersignale *Rücksetze Ausgang* und *Rücksetze Zähler* sind auch bei aktiviertem *TOR* wirksam.

Der Sollwert des Zählers wird über das PAA vorgegeben und verändert. Durch die positive Flanke des Steuersignals *Rücksetze Zähler* oder einen Nulldurchgang des Zählers wird der Sollwert übernommen.

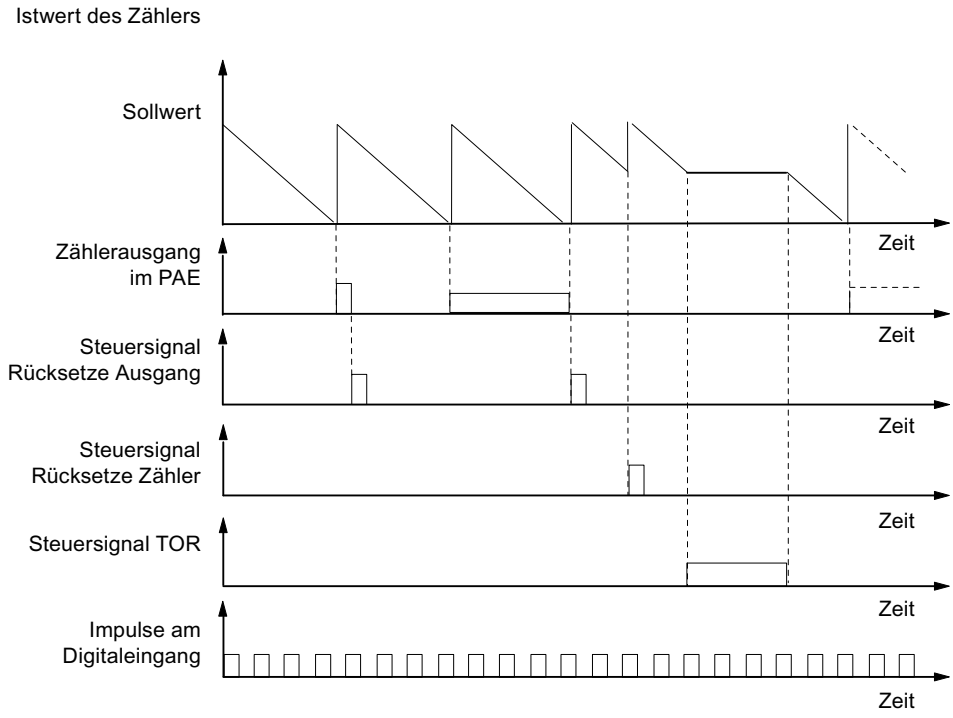

32-Bit Rückwärtszähler (Kaskadierzählfunktion)

Der maximale Zählbereich beträgt 4294967295 bis 0.

Die Funktionsweise ist identisch zum 16-Bit Rückwärtszähler. Der Kanal 1 hat keine Funktion.

Siehe auch

Eigenschaften Zählen (Seite 3367)

Zähler konfigurieren

Vorgehensweise

1. Ziehen Sie mit der Maus das Modul 8 DI Namur aus dem Hardware-Katalog in das dezentral Peripheriegerät ET 200iSP.
2. Wählen Sie die gewünschte Konfiguration (Kanal 0..1: "Zähler", Kanal 2..7: "DI" bzw. "Control"). In den Eigenschaften des Moduls (Inspektorfenster) finden Sie diese Einstellung unter "Parameter > Eingänge > Konfiguration".

Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "DI"

- Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR

Tabelle 1-235 Belegung der Digitaleingänge bei Kanal 0..1: "Zähler", Kanal 2..7: "DI":

Digitaleingang	Klemme	Belegung
Kanal 0	1, 2	Zähler 1
Kanal 1	5, 6	Zähler 2 (bei 32-Bit Rückwärtszähler nicht relevant)
Kanal 2	9, 10	Digitaleingang 2
Kanal 3	13, 14	Digitaleingang 3
Kanal 4	3, 4	Digitaleingang 4
Kanal 5	7, 8	Digitaleingang 5
Kanal 6	11, 12	Digitaleingang 6
Kanal 7	15, 16	Digitaleingang 7

- Belegung des Prozessabbildes der Eingänge (PAE)

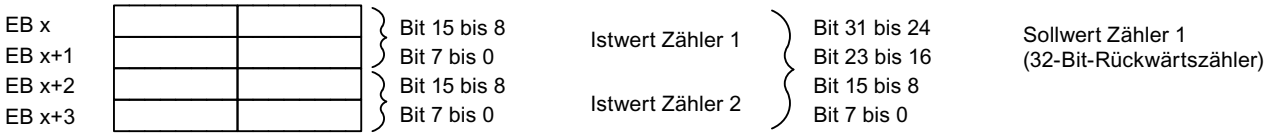

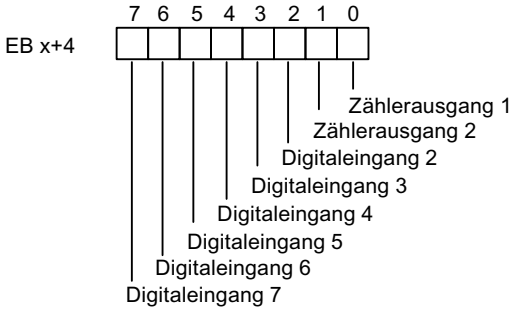

S7-Format

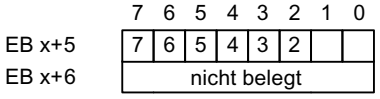

Wertstatus für Kanal 2 bis 7:
1_B: Eingangssignal ist gültig
0_B: Eingangssignal ist ungültig

- Belegung des Prozessabbildes der Ausgänge (PAA)

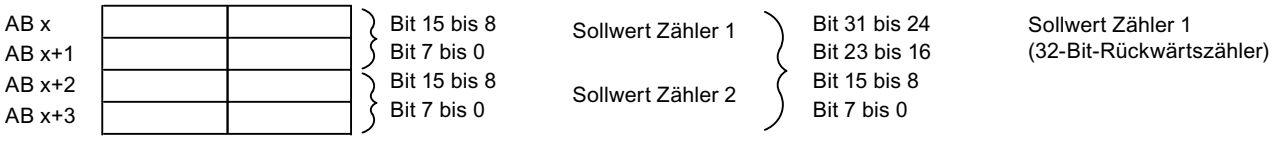

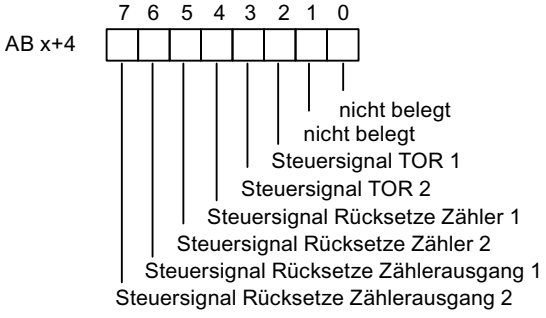

Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "CONTROL"

Bei dieser Konfiguration können Sie die Zähler zusätzlich über die Digitaleingänge steuern.

- Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR
  Weitere Hinweise zur Anschlussbelegung finden Sie in den Technischen Daten des Elektronikmoduls 8 DI NAMUR.

Tabelle 1-236 Belegung der Digitaleingänge bei 2 Count/ 6 Control

Digitaleingang	Klemme	Belegung
Kanal 0	1, 2	Zähler 1
Kanal 1	5, 6	Zähler 2 (bei 32-Bit Rückwärtszähler nicht relevant)
Kanal 2	9, 10	Steuersignal *TOR 1*
Kanal 3	13, 14	Steuersignal *TOR 2*
Kanal 4	3, 4	Steuersignal *Rücksetze Zähler 1*
Kanal 5	7, 8	Steuersignal *Rücksetze Zähler 2*
Kanal 6	11, 12	Steuersignal *Rücksetze Zählerausgang 1*
Kanal 7	15, 16	Steuersignal *Rücksetze Zählerausgang 2*

- Belegung des Prozessabbildes der Eingänge (PAE)
  Die Belegung ist identisch zur Konfiguration 0..1: "Zähler", Kanal 2..7: "DI".
- Belegung des Prozessabbildes der Ausgänge (PAA)
  Die Belegung ist identisch zur Konfiguration 0..1: "Zähler", Kanal 2..7: "DI".

Siehe auch

Eigenschaften Zählen (Seite 3367)

Zähler parametrieren

Parameter für die Zählfunktion

Nachfolgend werden nur die Parameter erläutert, die für die Zähler relevant sind. Diese sind Bestandteil der Parameter des Elektronikmoduls 8 DI NAMUR und abhängig von der gewählten Konfiguration:

Tabelle 1-237 Parameter für die Zähler

Parameter	Einstellung	Beschreibung
Gebertyp Zählereingänge	• Kanal gesperrt • NAMUR-Geber • Einzelkontakt unbeschaltet	Wählen Sie für den Kanal 0 oder 1 den Geber für den jeweiligen Zähler aus.
Betriebsart Zähler 1	• Normale Zählfunktion • Periodische Zählfunktion • Kaskadierzählfunktion	Wählen Sie die Betriebsart des Zählers 1.
Betriebsart Zähler 2	• Normale Zählfunktion • Periodische Zählfunktion • Kaskadierzählfunktion	Wählen Sie die Betriebsart des Zählers 2. Dieser Parameter ist nicht relevant, wenn Sie den Parameter "Betriebsart Zähler 1" auf "Kaskadierzählfunktion" gestellt haben.

Siehe auch

Eigenschaften Zählen (Seite 3367)

Frequenzmessen

Eigenschaften Frequenzmessen

Eigenschaften

Das Elektronikmodul 8 DI NAMUR ermöglicht auf den Kanälen 0 und 1 das Messen von Frequenzen:

- 2 Frequenzmesser von 1 Hz bis 5 kHz
- Parametrierbares Messfenster (TOR)
- Die Signale der Frequenzmesser werden über die Digitaleingänge des Elektronikmoduls eingelesen.

Siehe auch

Funktionsweise (Seite 3373)

Frequenzmesser konfigurieren (Seite 3374)

Frequenzmesser parametrieren (Seite 3376)

Funktionsweise

Frequenzmessung

Die Signalfrequenzen werden aus den Eingangssignalen des Kanal 0 bzw. 1 des Elektronikmoduls ermittelt. Zur Berechnung der Frequenz werden die Signale innerhalb eines parametrierbaren Messfensters gemessen.

Die Frequenz wird als 16-Bit Wert im Festpunktformat dargestellt und in das PAE übertragen.

Die Frequenzmesser berechnen die Frequenz nach folgender Formel:

$$\text{Frequenz [Hz]} = \frac{\text{Anzahl der steigenden Flanken am Digitaleingang}}{\text{Messfenster [s]}}$$

Überschreitung der Eingangsfrequenz

Wenn die Eingangsfrequenz 5kHz übersteigt, dann wird als Istwert $7FFF_H$ gemeldet. Bei einer Eingangsfrequenz über ca. 8kHz können keine korrekten Istwerte mehr ausgegeben werden.

Siehe auch

Eigenschaften Frequenzmessen (Seite 3373)

Frequenzmesser konfigurieren

Vorgehensweise

1. Ziehen Sie mit der Maus das Modul 8 DI Namur aus dem Hardware-Katalog in das dezentral Peripheriegerät ET 200iSP.
2. Wählen Sie die gewünschte Konfiguration (Kanal 0..1: "Trace", Kanal 2..7: "DI"). In den Eigenschaften des Moduls (Inspektorfenster) finden Sie diese Einstellung unter "Parameter > Eingänge > Konfiguration".

Konfiguration 0..1: "Trace", Kanal 2..7: "DI"

Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR

Digitaleingang	Klemme	Belegung
Kanal 0	1, 2	Frequenzmesser 1
Kanal 1	5, 6	Frequenzmesser 2
Kanal 2	9, 10	Digitaleingang 2
Kanal 3	13, 14	Digitaleingang 3
Kanal 4	3, 4	Digitaleingang 4
Kanal 5	7, 8	Digitaleingang 5
Kanal 6	11, 12	Digitaleingang 6
Kanal 7	15, 16	Digitaleingang 7

Belegung des Prozessabbildes der Eingänge (PAE) bei Konfiguration Kanal 0..1: "Trace", Kanal 2..7: "DI"

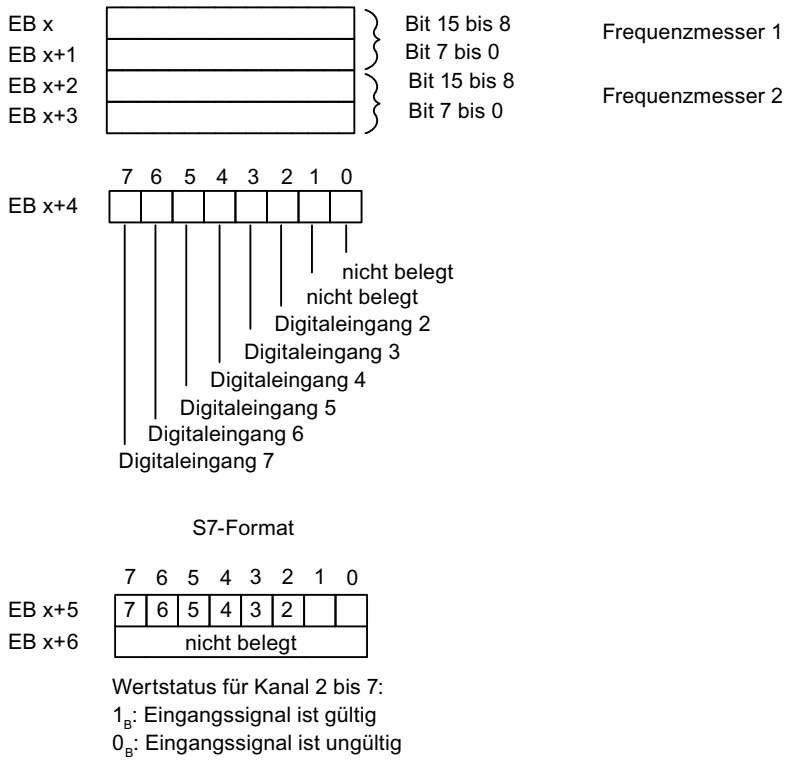

Belegung des Prozessabbildes der Ausgänge (PAA): Das PAA ist nicht belegt.

Siehe auch

Eigenschaften Frequenzmessen (Seite 3373)

Frequenzmesser parametrieren

Parameter für Frequenzmesser

Nachfolgend werden nur die Parameter erläutert, die für die Frequenzmesser relevant sind. Diese sind Bestandteil der Parameter des Elektronikmoduls 8 DI NAMUR:

Tabelle 1-238 Parameter für die Frequenzmesser

Parameter	Einstellung	Beschreibung
Gebertyp Frequenzeingänge	• Kanal gesperrt • NAMUR-Geber • Einzelkontakt unbeschaltet	Wählen Sie für den Kanal 0 oder 1 den Geber für den jeweiligen Frequenzmesser aus.
Messfenster (TOR)	• 50 ms • 200 ms • 1 s	Wählen Sie für den Kanal 0 bzw. 1 das erforderliche Messfenster aus. Damit Sie eine möglichst hohe Genauigkeit bei der Frequenzmessung erreichen, beachten Sie folgende Regeln: • hohe Frequenzen (>4kHz): niedriges Messfenster (50ms) einstellen • variable/ mittlere Frequenzen: mittleres Messfenster (200ms) einstellen • niedrige Frequenzen (<1kHz): hohes Messfenster (1s) einstellen

Siehe auch

Eigenschaften Frequenzmessen (Seite 3373)

1.6.3.4 ET 200eco PN

Dezentrales Peripheriegerät ET 200eco PN

Definition

Das Dezentrale Peripheriegerät ET 200eco PN ist ein kompaktes PROFINET IO-Device in der Schutzart IP 65/66 bzw. IP 67 und UL Enclosure Type 4x, Indoor use only.

Einsatzgebiet

Die Einsatzgebiete der ET 200eco PN ergeben sich aus deren besonderen Eigenschaften.

- Durch die robuste Bauweise und die Schutzart IP 65/66 bzw. IP 67 ist das Dezentrale Peripheriegerät ET 200eco PN vor allem für den Einsatz in rauher Industrieumgebung geeignet.
- Der kompakte Aufbau der ET 200eco PN ermöglicht den Einsatz in platzsensitiven Bereichen.
- Das einfache Handling der ET 200eco PN gewährleistet eine schnelle Inbetriebnahme und Wartung.

Eigenschaften

Die ET 200eco PN verfügt über die folgenden Eigenschaften:

- Integrierter Switch mit 2 Ports
- Unterstützte Ethernet-Dienste:
  - ping
  - arp
  - Netzdiagnose (SNMP)
  - LLDP
- Alarme
  - Diagnosealarme
  - Maintenance Alarme
- Port-Diagnose
- Isochronous Real-Time-Kommunikation
- Priorisierter Hochlauf
- Gerätetausch ohne PG
- Medienredundanz
- Über IO-Link Master Interfacemodul Anbindung an intelligente Sensoren/Aktoren.

IO-Controller

Die ET 200eco PN kann mit allen IO-Controllern kommunizieren, die sich nach der Norm IEC 61158 verhalten.

Die ET 200eco PN ist projektierbar an einer CPU mit erweiterter Diagnose.

Siehe auch

Dokumentation zu ET 200eco PN (http://support.automation.siemens.com/WW/view/de/29999018)

Parameterbeschreibung Analogeingabe

Sammeldiagnose

Mit diesem Parameter können Sie generell die Diagnose des Gerätes freigeben und sperren.

Die Diagnosen "Fehler" und "Parametrierfehler" sind immer unabhängig von der Sammeldiagnose freigegeben.

Diagnose fehlende 1L+

Wenn Sie diesen Parameter freigeben, dann wird die Prüfung auf fehlende Versorgungsspannung freigegeben.

Diagnose Kurzschluss Geberversorgung

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Geberversorgung nach Masse bei aktiviertem Kanal eine Diagnose generiert. Die Geberversorgung wird für die Stecker X1, X3, X5 und X7 überwacht. Es kann nicht differenziert werden, bei welchem Stecker der Geberkurzschluss auftritt.

Störfrequenzunterdrückung

Mit diesem Parameter stellen Sie abhängig von der Auswahl der Störfrequenz die Integrationszeit des Gerätes ein. Wählen Sie hier die Frequenz der verwendeten Netzspannung. Störfrequenzunterdrückung **Aus** bedeutet 500 Hz, was bei einem Messkanal einer Integrationszeit von 2 ms entspricht.

Temperatureinheit

Stellen Sie hier die Einheit ein, in der Sie die Temperatur erfassen möchten.

Messart (kanalweise)

Mit diesem Parameter stellen Sie die Messart ein, z. B. Spannung. Wenn Sie einen Kanal nicht verwenden, dann müssen Sie als Einstellung **Deaktiviert** wählen. Bei einem deaktivierten Kanal ist die Wandlungs- und Integrationszeit des Kanals = 0 s und die Zykluszeit optimiert.

Messbereich

Mit diesem Parameter stellen Sie den Messbereich der gewählten Messart ein.

Temperaturkoeffizient (bei RTD, Thermowiderstand)

Der Korrekturfaktor für den Temperaturkoeffizienten (α-Wert) gibt an, um wieviel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Die α-Werte entsprechen den Normen EN 60751, GOST 6651, JIS C 1604 und ASTM E-1137.

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials.

Glättung

Durch die Glättung von Analogwerten wird ein stabiles Analogsignal für die Weiterverarbeitung erzeugt. Die Glättung der Analogwerte ist sinnvoll bei langsamen Nutzsignaländerungen (Messwertänderungen), z. B. Temperaturmessungen.

Die Messwerte werden mittels digitaler Filterung geglättet. Die Glättung wird erreicht, indem das Gerät Mittelwerte aus einer festgelegten Anzahl von gewandelten (digitalisierten) Analogwerten bildet.

Sie parametrieren die Glättung in maximal 4 Stufen (keine, schwach, mittel, stark). Die Stufe bestimmt die Anzahl der Baugruppenzyklen, die zur Mittelwertbildung herangezogen werden.

Je stärker die Glättung durchgeführt wird, umso stabiler ist der geglättete Analogwert und umso länger dauert es, bis der geglättete Analogwert nach einem Signalwechsel anliegt (siehe folgendes Beispiel).

Das folgende Bild zeigt, nach wie vielen Baugruppenzyklen bei einer Sprungantwort der geglättete Analogwert zu annähernd 100 % anliegt, in Abhängigkeit von der eingestellten Glättung. Das Bild gilt für jeden Signalwechsel am Analogeingang. Der Glättungswert gibt die Anzahl der Zyklen an, die die Baugruppe braucht, bis 63 % vom Endwert der Signalwechsel erreicht ist.

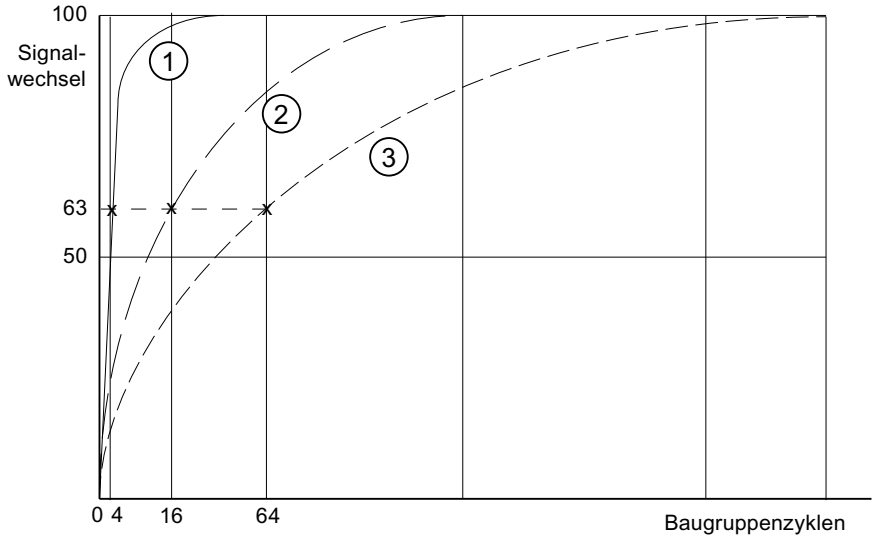

① Glättung schwach
② Glättung mittel
③ Glättung stark

Diagnose Drahtbruch

Wenn Sie diesen Parameter freigeben, dann wird bei einem Drahtbruch die Diagnose **Leitungsbruch** generiert.

Beachten Sie bei einem Drahtbruch in den Messbereichen 1 bis 5 V und 4 bis 20 mA folgende Regeln:

Parameter	Ereignis	Messwert	Erläuterung
Drahtbruch freigegeben[1]	Drahtbruch	$7FFF_H$	Diagnose **Leitungsbruch**
Drahtbruch gesperrt[1] Unterlauf freigegeben	Drahtbruch	8000_H	Messwert nach Verlassen des Untersteuerungsbereiches Diagnosemeldung **Unterer Grenzwert unterschritten**
Drahtbruch gesperrt[1] Unterlauf gesperrt	Drahtbruch	8000_H	Messwert nach Verlassen des Untersteuerungsbereiches
[1] Messbereichsgrenzen für die Erkennung des Drahtbruchs und der Messbereichsunterschreitung: • 1 bis 5 V: bei 0,296 V • 4 bis 20 mA: bei 1,185 mA			

Diagnose Unterlauf

Wenn Sie diesen Parameter freigeben, dann wird die Diagnose **Unterlauf** generiert, wenn der Messwert den Bereich Unterlauf erreicht.

Diagnose Überlauf

Wenn Sie diesen Parameter freigeben, dann wird die Diagnose **Überlauf** generiert, wenn der Messwert den Bereich Überlauf erreicht.

Vergleichsstelle für Thermowiderstand (TC)

Wird die Messstelle einer anderen Temperatur ausgesetzt als die freien Enden des Thermopaars (Anschlussstelle), entsteht zwischen den freien Enden eine Spannung, die Thermospannung. Die Höhe der Thermospannung hängt von der Differenz zwischen der Temperatur der Messstelle und der Temperatur an den freien Enden ab, sowie von der Art der Werkstoffkombination des Thermopaars. Da mit einem Thermopaar immer eine Temperaturdifferenz erfasst wird, müssen die freien Enden an der Vergleichsstelle auf bekannter Temperatur gehalten werden, um die Temperatur der Messstelle bestimmen zu können.

Wenn Sie **Interne Kompensation** einstellen, wird die Temperatur der Messstelle im Gehäuse des Peripheriegerätes gemessen. Bei Einstellung **Externe Kompensation** können Sie eine Kompensationsbox vorschalten, um eine größere Genauigkeit bei der Temperaturmessung zu erzielen.

Parameterbeschreibung Analogausgabe

Sammeldiagnose

Mit diesem Parameter können Sie generell die Diagnose des Gerätes freigeben und sperren.

Die Diagnosen "Fehler" und "Parametrierfehler" sind immer unabhängig von der Sammeldiagnose freigegeben.

Diagnose fehlende 1L+

Wenn Sie diesen Parameter freigeben, dann wird die Prüfung auf fehlende Versorgungsspannung freigegeben.

Diagnose Kurzschluss Geberversorgung

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Geberversorgung nach Masse eine Diagnose generiert. Diese Diagnose wird bei Freigabe der Sammeldiagnose freigeschaltet.

Verhalten bei CPU/Master-STOP

Wählen Sie aus, wie die Ausgänge der Baugruppe im Fall eines CPU-STOPs reagieren sollen:

- Abschalten
  Das Peripheriegerät geht in den sicheren Zustand. Das Prozessabbild der Ausgänge wird gelöscht (=0).
- Letzten Wert halten
  Das Peripheriegerät behält den zuletzt vor STOP ausgegebenen Wert bei.
- Ersatzwert
  Das Peripheriegerät gibt den zuvor für den Kanal eingestellten Wert aus.

Hinweis

Vergewissern Sie sich, dass sich die Anlage im Fall "letzten Wert halten" immer in einem sicheren Zustand befindet!

Ausgabeart

Mit diesem Parameter stellen Sie die Ausgabeart ein, z. B. Spannung. Wenn Sie einen Kanal nicht verwenden, dann sollten Sie als Einstellung **Deaktiviert** wählen. Bei einem deaktivierten Kanal ist die Wandlungs- und Integrationszeit des Kanals = 0 s und die Zykluszeit optimiert.

Ausgabebereich

Mit diesem Parameter stellen Sie den Ausgabebereich der gewählten Ausgabeart ein.

Diagnose Drahtbruch (im Strommodus)

Wenn Sie diesen Parameter freigeben, dann wird bei einem Drahtbruch die Diagnose **Leitungsbruch** generiert. Im Nullbereich ist diese Diagnose nicht erkennbar.

Diagnose Kurzschluss (im Spannungsmodus)

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Ausgangsleitung eine Diagnose generiert. Im Nullbereich ist diese Diagnose nicht erkennbar.

Diagnose Überlast

Wenn Sie diesen Parameter freigeben, dann wird bei einer Überlast die Diagnose generiert.

Ersatzwerte

Bei diesem Parameter geben Sie einen Ersatzwert ein, den die Baugruppe bei CPU-STOP ausgeben soll. Der Ersatzwert muss im Nenn-, Übersteuerungs- oder Untersteuerungsbereich liegen.

1.6.3.5 ET 200SP

Dezentrales Peripheriesystem ET 200SP

Definition

Das Dezentrale Peripheriesystem ET 200SP ist ein skalierbares und hochflexibles, dezentrales Peripheriesystem zur Anbindung der Prozesssignale an eine Zentralsteuerung über einen Feldbus.

Einsatzgebiet

Das ET 200SP ist ein multifunktionales dezentrales Peripheriesystem für unterschiedliche Einsatzbereiche. Durch den skalierbaren Aufbau können Sie den Ausbau exakt auf den jeweiligen Bedarf vor Ort ausrichten.

Das ET 200SP ist für die Schutzart IP 20 zugelassen und für den Einbau in einem Schaltschrank vorgesehen.

Aufbau

Das ET 200SP wird auf eine Profilschiene montiert und setzt sich zusammen aus:

- einem Interfacemodul, das mit allen IO-Controllern kommuniziert, die sich nach der PROFINET-Norm IEC 61158 verhalten,
- bis zu 32 Peripheriemodulen, die in beliebiger Kombination auf passive BaseUnits gesteckt werden,
- einem Servermodul, das den Aufbau des ET 200SP abschließt.

ET 200SP mit eigener CPU-Baugruppe

ET 200SP mit eigener CPU

Die ET 200SP kann mit einer eigenen CPU ausgestattet sein, zum Beispiel mit der CPU1510SP-1 PN oder der CPU 1512SP-1 PN.

An diese CPU können Sie bis zu 65 Module des Automatisierungssystems S7-1500 anfügen.

Beachten Sie dabei die Ausbauregeln der ET 200SP (http://support.automation.siemens.com/WW/view/de/58649293).

Parameter Interfacemodul

Statusbytes

Statusbytes

Wenn Sie die Option "Statusbytes" aktivieren, werden 4 Bytes Eingangsdaten reserviert für den Status der Versorgungsspannung jedes Peripheriemoduls.

	7	6	5	4	3	2	1	0
Byte 0	8	7	6	5	4	3	2	1
Byte 1	16	15	14	13	12	11	10	9
Byte 2	24	23	22	21	20	19	18	17
Byte 3	32	31	30	29	28	27	26	25

Steckplätze der Peripheriemodule

Bit = 0: Lastspannung fehlt oder Peripheriemodul nicht vorhanden

Bit = 1: Lastspannung und Peripheriemodul vorhanden

Hinweis

Ein gestecktes oder fehlendes Servermodul meldet immer für den Steckplatz Bit = 0.

Sammeldiagnose fehlende Versorgungsspannung L+

Sammeldiagnose fehlende Versorgungsspannung L+

Diese Diagnose ist eine Sammeldiagnose über den Versorgungsspannungsstatus aller Peripheriemodule einer Potenzialgruppe, die durch BaseUnits mit Spannungseinspeisung (helle BaseUnit BU...D) festgelegt sind.

Die Sammeldiagnose wird aus den Zuständen der Versorgungsspannung der gesteckten Peripheriemodule innerhalb der Potenzialgruppe gebildet.

Die Sammeldiagnose ist unabhängig von der Freigabe des Parameters "Fehlende Versorgungsspannung L+" der Peripheriemodule.

Das Servermodul beeinflusst nicht die Sammeldiagnose fehlende Versorgungsspannung L+.

Voraussetzungen für die korrekte Funktionsweise der Sammeldiagnose fehlende Versorgungsspannung L+:

- Auf den hellen und dunklen BaseUnits müssen Peripheriemodule bzw. BU Cover stecken. Wenn auf einer hellen BaseUnit kein Peripheriemodul steckt, wird der Beginn dieser Potenzialgruppe vom Interfacemodul nicht erkannt; die Peripheriemodule dieser Potenzialgruppe gehören somit zur vorherigen Potenzialgruppe. Ein Sammelfehler fehlende Versorgungsspannung L+ wird dann der falschen Potenzialgruppe zugeordnet. Wenn ein Peripheriemodul auf die helle BaseUnit gesteckt wird, erkennt das Interfacemodul die neue Potenzialgruppe, bewertet den Status neu und meldet im Fehlerfall eine neue Sammeldiagnose.
- Das Servermodul muss gesteckt sein.
  Das Servermodul selbst beeinflusst aber nicht die Sammeldiagnose fehlende Versorgungsspannung L+.

Konfigurationssteuerung bei ET 200SP

Funktionsprinzip

Durch die Konfigurationssteuerung betreiben Sie mit einer einzigen Projektierung des Dezentralen Peripheriegeräts ET 200SP unterschiedliche reale Konfigurationen (Optionen).

Durch die Konfigurationssteuerung haben Sie die Möglichkeit, das Dezentrale Peripheriegerät ET 200SP mit seinem Vollausbau zu projektieren und trotzdem mit fehlenden Modulen zu betreiben. Wenn fehlende Module später nachgerüstet werden, ist keine neue Projektierung erforderlich und damit auch kein erneutes Laden der Hardware-Konfiguration.

Mit Hilfe des Steuerdatensatzes 196, der im Anwenderprogramm an das Interfacemodul übermittelt wird, legen Sie eine aktuelle Sollkonfiguration fest.

- Auf einem Steckplatz ist das projektierte Modul nicht vorhanden.
  - Statt des projektierten Peripheriemoduls darf ein BU-Cover auf diesem Steckplatz gesteckt sein. Da das projektierte Modul auf dem Steckplatz fehlt, wird auch von einer "Konfigurationsteuerung mit Leerplätzen" gesprochen.
  - Statt des projektierten Moduls darf auf diesem Steckplatz das Modul stecken, das rechts neben dem fehlenden Modul projektiert ist. Die Istkonfiguration erscheint durch das fehlende Modul zusammengeschoben. Da das projektierte Modul fehlt, aber keine Lücke in der Konfiguration entsteht, spricht man auch von einer "Konfigurationssteuerung ohne Leerplätze".
- Die Konfiguration wird durch ein bereits projektiertes Modul erweitert.
  - Bei der Konfigurationssteuerung mit Leerplätzen erweitern Sie die Konfiguration, indem Sie das projektierte Modul auf den entsprechenden Leerplatz stecken.
  - Bei der Konfigurationssteuerung ohne Leerplätze stecken Sie das projektierte Modul rechts neben dem letzten Modul der ET 200SP.

Voraussetzung

Der CPU-Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf Anlauf auch bei Unterschieden (Voreinstellung). Diese Einstellung ist auch für den Anlaufparameter der einzelnen Module der ET 200SP gewählt.

Konfigurationssteuerung aktivieren

Aktivieren Sie in den Eigenschaften des Interfacemoduls unter Baugruppenparameter > Allgemein > Konfigurationssteuerung die Option "Umkonfigurieren des Geräts über Anwenderprogramm ermöglichen". Dadurch ist die Konfigurationssteuerung eingeschaltet.

Steuerdatensatz 196 für ET 200SP

Das Bild unten zeigt den Datenbaustein 196 für die Konfigurationssteuerung einer ET 200SP mit vier Modulen.

Im Element "block_length" steht der Wert "12".

Wenn Sie in STEP 7 eine ET 200SP mit mehr Modulen projektieren, dann verlängert sich der Datenbaustein. Bei einem Maximalausbau mit 65 Modulen ist der Datensatz 134 Bytes lang (Projektierung mit PN-Interfacemodul).

Für jedes Modul sind zwei Bytes im Datensatz vorgesehen.

Die Position dieser beiden Bytes im Datensatz kodiert jeweils ein Modul bei der ursprünglichen Projektierung mit STEP 7:

- "slot_1" und "info_slot_1" (Byte 4 und 5 im Datensatz, siehe folgendes Bild) entsprechen dem Modul auf Steckplatz 1 in der Projektierung mit STEP 7.
- "slot_2" und "info_slot_2" (Byte 6 und 7) entsprechen dem Modul auf dem Steckplatz 2 in der Projektierung mit STEP 7.
- "slot_3" und "info_slot_3" (Byte 8 und 9) entsprechen dem Modul auf dem Steckplatz 3 in der Projektierung mit STEP 7.
- usw.

Byte "slot_x"

Der aktuelle Steckplatz ist durch die Zahl kodiert, die "slot_x" zugewiesen wird (durch seinen Wert). Beispiele:

- Der Wert "1" im Byte 4 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, auch in der aktuellen Konfiguration den Platz 1 zu (slot_1 = 1).
- Der Wert "2" im Byte 4 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, in der aktuellen Konfiguration den Platz 2 zu (slot_1 = 2).
- Der Wert "3" im Byte 4 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, in der aktuellen Konfiguration den Platz 3 zu (slot_1 = 3).
- usw.
- Der Wert "1" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Platz 1 zu (slot_2 = 1).
- Der Wert "2" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Platz 2 zu (slot_2 = 2).

- Der Wert "3" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Platz 3 zu (slot_2 = 3).
- usw.

Wenn anstelle eines Moduls auch ein BU-Cover gesteckt sein kann, dann kodieren Sie das, indem Sie zum Steckplatz die Zahl 128 addieren (Bit 7 des Bytes "slot_x" ist gesetzt). Beispiele:

- Der Wert "129" in slot_1 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, auch in der aktuellen Konfiguration den Steckplatz 1 zu. Zudem kann anstelle dieses Moduls ein BU-Cover verwendet werden. Im realen Aufbau in der Anlage ist dann entweder das Modul gesteckt oder ein BU-Cover.
- Der Wert "130" in slot_1 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, in der aktuellen Konfiguration den Platz 2 zu. Zudem kann anstelle dieses Moduls ein BU-Cover verwendet werden. Im realen Aufbau in der Anlage ist dann entweder das Modul gesteckt oder ein BU-Cover.
- Der Wert "131" in slot_1 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 1 steckte, in der aktuellen Konfiguration den Platz 3 zu. Zudem kann anstelle dieses Moduls ein BU-Cover verwendet werden. Im realen Aufbau in der Anlage ist dann entweder das Modul gesteckt oder ein BU-Cover.

Byte "info_slot_x"

Wenn mit dem Modul eine neue Potenzialgruppe geöffnet wird, dann weisen Sie dem Byte "info_slot_x" den Wert 1 zu (Bit 0 des Bytes gesetzt). Beispiele:

- Der Wert "1" im Byte "info_slot_2" bedeutet, dass mit Modul 2 eine neue Potenzialgruppe geöffnet wird.
- Der Wert "1" im Byte "info_slot_3" bedeutet, dass mit Modul 3 eine neue Potenzialgruppe geöffnet wird.
- Der Wert "1" im Byte "info_slot_4" bedeutet, dass mit Modul 4 eine neue Potenzialgruppe geöffnet wird.

Ausnahme: Dem ersten Modul in der ursprünglichen Projektierung in STEP 7 ist automatisch eine neue Potenzialgruppe zugewiesen. Im Datensatz wird dies nicht kodiert. Sie können einen beliebigen Wert in "info_slot_1" eintragen.

Die Namen der Komponenten des Steuerdatensatzes (zum Beispiel "slot_1") können Sie frei wählen.

Beispiel für einen Steuerdatensatz 196 für ET 200SP

Das folgende Bild zeigt den Steuerdatensatz 196 für eine ET 200SP mit vier Modulen.

Das Modul, das bei der Projektierung in STEP 7 auf Steckplatz 2 steckte, kann bei dieser Konfiguration ebenfalls auf Steckplatz 2 stecken. Oder es steckt auf Steckplatz 2 ein BU-Cover. Sonst hat sich gegenüber der ursprünglichen Projektierung nichts geändert.

Record_196_ET_200SP

	Name	Data type	Default value
1	block_length	USInt	12
2	block_ID	USInt	196
3	version	USInt	2
4	subversion	USInt	0
5	slot_1	USInt	1
6	info_slot_1	USInt	0
7	slot_2	USInt	130
8	info_slot_2	USInt	0
9	slot_3	USInt	3
10	info_slot_3	USInt	0
11	slot_4	USInt	4
12	info_slot_4	USInt	0

Adressierung des Interfacemoduls über HW-Kennung

Um den Datensatz 196 mit der Anweisung WRREC zu übertragen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z. B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Rücklesedatensatz 197 für ET 200SP

Der Rücklesedatensatz 197 dient zum Lesen des tatsächlichen Aufbaus einer Station (hier einer ET 200SP).

Durch diesen Rücklesedatensatz kann der reale Ausbau der ET 200SP (Istkonfiguration) überprüft werden. Der Rücklesedatensatz enthält für jedes projektierte Modul die Angabe, ob es tatsächlich vorhanden ist oder nicht.

- Der Wert "1" bedeutet, dass das richtige Modul auf dem vorgesehenen Steckplatz steckt.
- Der Wert "0" kodiert alle anderen Möglichkeiten (falsches Modul, leerer Steckplatz, BU-Cover).

Der Aufbau im Einzelnen:

Der Aufbau des Datenbausteins entspricht der ursprünglichen Projektierung der ET 200SP mit STEP 7.

Für jedes Modul sind zwei Bytes im Datensatz vorgesehen. Die Position dieser beiden Bytes im Datensatz entspricht der Position eines Moduls bei der ursprünglichen Projektierung mit STEP 7.

Reihenfolge der Bytes:

- "status_slot_1" und "reserve_1" (Byte 4 und 5 im Datensatz) entsprechen dem Modul auf Steckplatz 1 der Projektierung,
- "status_slot_2" und "reserve_2" (Byte 6 und 7) entsprechen dem Modul auf Steckplatz 2 der Projektierung

- "status_slot_3" und "reserve_3" (Byte 8 und 9) entsprechen dem Modul auf Steckplatz 3 der Projektierung,
- usw.

Beispiel

Die ursprüngliche Projektierung in STEP 7 wurde durch einen Steuerdatensatz 196 (siehe Beispiel oben) geändert: In der geänderten Konfiguration kann Modul 2 entweder auf Steckplatz 2 stecken oder durch ein BU-Cover ersetzt sein.

Das folgende Bild zeigt den Rücklesedatensatz 197, mit dem die ET 200SP meldet, dass auf Steckplatz 2 das Modul vorhanden ist: Das Byte "status_slot_2" besitzt den Wert "1".

Die anderen Module sind ebenfalls vorhanden und stecken auf den richtigen Steckplätzen.

Record_197_ET_200SP

	Name	Data type	Default value
1	block_length	USInt	12
2	block_ID	USInt	197
3	version	USInt	2
4	subversion	USInt	0
5	status_slot_1	USInt	1
6	reserve_1	USInt	0
7	status_slot_2	USInt	1
8	reserve_2	USInt	0
9	status_slot_3	USInt	1
10	reserve_3	USInt	0
11	status_slot_4	USInt	1
12	reserve_4	USInt	0

Das folgende Bild zeigt den Rücklesedatensatz 197, mit dem die ET 200SP meldet, dass auf Steckplatz 2 ein BU-Cover verwendet wird: Das Byte "status_slot_2" besitzt den Wert "0".

Die anderen Module sind vorhanden und stecken auf den richtigen Steckplätzen.

Record_197_ET_200SP

	Name	Data type	Default value
1	block_length	USInt	12
2	block_ID	USInt	197
3	version	USInt	2
4	subversion	USInt	0
5	status_slot_1	USInt	1
6	reserve_1	USInt	0
7	status_slot_2	USInt	0
8	reserve_2	USInt	0
9	status_slot_3	USInt	1
10	reserve_3	USInt	0
11	status_slot_4	USInt	1
12	reserve_4	USInt	0

Rücklesedatensatz 197 lesen

Mit der Anweisung RDREC können Sie den Rücklesedatensatz 197 von der ET 200SP lesen. RDREC arbeitet asynchron. Wenn Sie RDREC im Anlauf-OB aufrufen, dann müssen Sie die Anweisung über eine Schleife mehrmals aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz gelesen ist.

Um den Datensatz 197 mit der Anweisung RDREC zu lesen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Weiterführende Informationen und Beispiele

Konkrete Beispiele für die Konfigurationssteuerung finden Sie in dieser Applikationsbeschreibung (http://support.automation.siemens.com/WW/view/de/29430270).

Weitere Informationen zur ET 200SP finden Sie in den Gerätehandbüchern IM 155-6 PN (http://support.automation.siemens.com/WW/view/de/73184046) und IM 155-6 DP (http://support.automation.siemens.com/WW/view/de/73098660)

Eine Bibliothek der Datensätze und weitere Anwendungsbeispiele sind hier abgelegt:

Vorlagen für die Datensätze (http://support.automation.siemens.com/WW/view/de/29430270)

Siehe auch

Konfigurationssteuerung bei ET 200AL (Seite 3431)

Konfigurationssteuerung bei ET 200SP mit integierten ET 200AL-Modulen (Seite 3389)

Konfigurationssteuerung bei ET 200SP mit integierten ET 200AL-Modulen

ET 200SP mit integrierten ET 200AL-Modulen

Die Konfigurationssteuerung bei ET 200SP und ET 200AL ist in separaten Hilfenthemen beschrieben. Siehe hierzu die Links unter "Siehe auch".

Die dort gezeigte Vorgehensweise gilt auch für die Konfigurationssteuerung einer ET 200SP mit integrierten AL-Modulen. Unterschiede gibt es beim Steuerdatensatz 196 und dem Rücklesedatensatz 197.

Der folgende Hilfetext beschreibt den Steuerdatensatz 196 und den Rücklesedatensatz 197 für eine ET 200SP, die mit ET 200AL-Modulen erweitert wurde.

Steuerdatensatz 196

Die folgenden zwei Bilder zeigen Ausschnitte aus dem Steuerdatensatz 196 für eine Projektierung einer ET 200SP, die mit ET 200AL-Modulen erweitert wurde.

Diese Projektierung soll als Beispiel dienen:

- Auf dem Steckplatz 1 "slot_1"befindet sich das Modul "BA Send 1xFC". Durch diese Baugruppe ist es möglich, ET 200AL-Module in eine ET 200SP zu integrieren. In unserer Beispiel-Projektierung sind 16 AL-Module an der BA-Send angeschlossen (maximaler Ausbau). Wenn eine BA-Send verwendet wird, dann muss dieses Modul auf den Steckplatz 1 stecken.
- Alle Steckplätze von 2 bis 64 sind mit ET 200SP-Modulen bestückt.
- Auf Steckplatz 65 steckt ein Server-Modul.
- Auf den Steckplätzen 66 bis 81 befinden sich 16 AL-Module.

Die ursprünglich mit STEP 7 projektierte ET 200SP mit integrierten AL-Modulen soll nun aus dem Anwenderprogramm heraus umkonfiguriert werden.

Die neue Konfiguration hat die folgenden Eigenschaften:

- Das Modul "BA Send 1xFC" steckt auf "slot_1_BA-Send" (fest vorgegeben).
- Modul 2 "slot_2" ist in der geänderten Konfiguration nicht vorhanden (Wert "0").
- Modul 3 "slot_3" steckt in der geänderten Konfiguration auf Steckplatz 2 (Wert "2").
- Modul 4 "slot_4" steckt in der geänderten Konfiguration auf Steckplatz 3 (Wert "3").
- Alle Module ab Steckplatz 5 bis Steckplatz 81 werden in der ursprünglichen Projektierung mit STEP 7 betrieben.

Configuration_ET200SP_with_AL-Modules

	Name	Data type	Start value
1	Static		
2	block_length	USInt	166
3	block_ID	USInt	196
4	version	USInt	2
5	subversion	USInt	0
6	slot_1_BA-Send	USInt	1
7	info_slot_1_BA-Send	USInt	0
8	slot_2	USInt	0
9	info_slot_2	USInt	0
10	slot_3	USInt	2
11	info_slot_3	USInt	0
12	slot_4	USInt	3
13	info_slot_4	USInt	0
14	slot_5	USInt	5
15	info_slot_5	USInt	0
16	slot_6	USInt	6
17	info_slot_6	USInt	0
18	slot_7	USInt	7
19	info_slot_7	USInt	0

Die Komponenten des Steuerdatensatzes 196 (Bild oben):

- block_length: Notieren Sie hier die Länge des Steuerdatensatzes, im Beispiel 166 (Byte). Die Länge des Steuerdatenblocks errechnet sich nach der Formel "2 x Anzahl der Steckplätze + 4".
- block_ID: Tragen Sie hier die Zahl 196 ein.

- version: Die ET 200SP verwendet die Version 2 des Steuerdatensatzes 196.
- subversion: Die ET 200SP verwendet die Subversion 0 des Steuerdatensatzes 196.

127		slot_62_SP	USInt	62
128		info_slot_62_SP	USInt	0
129		slot_63_SP	USInt	63
130		info_slot_63_SP	USInt	0
131		slot_64_SP	USInt	64
132		info_slot_64_SP	USInt	0
133		slot_65_SP	USInt	65
134		info_slot_65_SP	USInt	0
135		slot_66_AL	USInt	66
136		info_slot_66_AL	USInt	0
137		slot_67_AL	USInt	67
138		info_slot_67_AL	USInt	0
139		slot_68_AL	USInt	68
140		info_slot_68_AL	USInt	0
141		slot_69_AL	USInt	69
142		info_slot_69_AL	USInt	0

Die Komponenten des Steuerdatensatzes 196 (Bild oben):

- slot_65_SP: Dieses Byte bezieht sich auf das Server-Modul im ET 200SP-Rack. Es schließt den Rückwandbus der ET 200SP ab.
- Ab "slot_66_AL" folgen die 16 projektierten ET 200AL-Module: Unsere Beispiel-Konfiguration verändert die Projektierung durch STEP 7 nicht. Das Byte "slot_66_AL" hat den Wert "66", das Byte "slot_67_AL" hat den Wert "67", das Byte "slot_68_AL" hat den Wert "68", usw.

Definition des Steuerdatensatzes 196

Für die Konfigurationssteuerung wird ein Steuerdatensatz 196 definiert, der eine Steckplatzzuordnung enthält.

Byte	Komponente	Wert	Erläuterungen	
0	block_length	z.B. 166 bei Maximalausbau mit 65 ET-200SP-Modulen und 16 ET 200AL-Modulen (bei DP-Interfacemodul maximal 33 ET 200SP-Module und 16 ET 200AL-Module)	Die Länge des Datensatzes errechnet sich nach der Formel: 4 + "Anzahl der Module" x 2	Header
1	block-ID	196	ID für den Steuerdatensatz 196	
2	version	2	Version 2 des Steuerdatensatzes 196	
3	subversion	0	Subversion 0 des Steuerdatensatzes 196	
4	slot_1_BA-Send	Realer Steckplatz für SP-Modul 1 Möglicher Wert: 1	Bei Integration von AL-Modulen in ET 200SP muss auf Steckplatz 1 immer das Modul "BA Send 1xFC" stecken.	1. Steckplatz für SP-Module Zuordnung für das projektierte SP-Modul 1 zu einem realen Steckplatz
5	info_slot_1_BA-Send	0 oder 1	Der Wert "1" bedeutet, dass mit diesem Modul eine neue Potenzialgruppe geöffnet wird. (In diesem Byte nicht ausgewertet)	
6	slot_2	Realer Steckplatz für SP-Modul 2 Mögliche Werte: 2 bis 65 (nicht 66 bis 81, reserviert für AL-Module) 0 (falls das projektierte Modul 2 fehlt)	Das projektierte SP-Modul 2 kann real in einem der Steckplätze 2 bis 65 gesteckt sein (2 bis 33 bei DP-Interfacemodul). Für AL-Module sind die Steckplätze 66 bis 81 vorgesehen (34 bis 49 bei DP-Interfacemodul).	2. Steckplatz für SP-Module Zuordnung für das projektierte SP-Modul 2 zu einem realen Steckplatz
7	info_slot_2	1	Der Wert "1" bedeutet, dass mit diesem Modul eine neue Potenzialgruppe geöffnet wird. In diesem Byte muss immer eine neue Potenzialgruppe geöffnet werden, da BA-Send keine neue Potenzialgruppe eröffnen kann.	

8	slot_3	Realer Steckplatz für SP-Modul 3 Mögliche Werte: 2 bis 65 (nicht 66 bis 81, reserviert für AL-Module) 0 (falls das projektierte Modul 3 fehlt)	Das projektierte SP-Modul 3 kann real in einem der Steckplätze 2 bis 65 gesteckt sein (2 bis 33 bei DP-Interfacemodul). Für AL-Module sind die Steckplätze 66 bis 81 vorgesehen (34 bis 49 bei DP-Interfacemodul).	3. Steckplatz für SP-Module Zuordnung für das projektierte SP-Modul 3 zu einem realen Steckplatz
9	info_slot_3	1	Der Wert "1" bedeutet, dass mit diesem Modul eine neue Potenzialgruppe geöffnet wird. .	
:	:	:	:	:
132	slot_65	Realer Steckplatz für SP-Modul 65 Mögliche Werte: 2 bis 65 (nicht 66 bis 81, reserviert für AL-Module) 0 (falls das projektierte Modul 65 fehlt)	Das projektierte SP-Modul 65 kann real in einem der Steckplätze 2 bis 65 gesteckt sein (2 bis 33 bei DP-Interfacemodul). Für AL-Module sind die Steckplätze 66 bis 81 vorgesehen (34 bis 49 bei DP-Interfacemodul)	65. Steckplatz für SP-Module Zuordnung für das projektierte SP-Modul 65 zu einem realen Steckplatz
133	info_slot_65	0 oder 1	Der Wert "1" bedeutet, dass mit diesem Modul eine neue Potenzialgruppe geöffnet wird (Der Wert wird auf diesem Steckplatz nicht ausgewertet)	
134	slot_66	Realer Steckplatz für AL-Modul 1 Mögliche Werte: 66 bis 81 (nicht 1 bis 65, reserviert für SP-Module) 0 (falls das projektierte AL-Modul 1 fehlt)	Das projektierte AL-Modul 1 kann real in einem der Steckplätze 66 bis 81 gesteckt sein (34 bis 49 bei PROFIBUS).	1. Steckplatz für AL-Module Zuordnung für das projektierte AL-Modul 1 zu einem realen Steckplatz
135	info_slot_66	-	Reserve	
:	:	:	:	:

164	slot_81	Realer Steckplatz für AL-Modul 16 Mögliche Werte: 66 bis 81 (nicht 1 bis 65, reserviert für SP-Module) 0 (falls das projektierte AL-Modul 16 fehlt)	Das projektierte AL-Modul 16 kann real in einem der Steckplätze 66 bis 81 gesteckt sein (34 bis 49 bei DP-Interfacemodul).	16. Steckplatz für AL-Module Zuordnung für das projektierte AL-Modul 16 zu einem realen Steckplatz
165	info_slot_81	-	Reserve	

Regeln

- Wenn das Modul "BA Send 1xFC" verwendet wird, dann muss es auf Steckplatz 1 stecken.
- Auf den Steckplätzen 2 bis 65 stecken ET 200SP-Module (auf den Steckplätzen 2 bis 33 bei DP-Interfacemodul).
- Auf den Steckplätzen 66 bis 81 stecken AL-Module (auf den Steckplätzen 34 bis 49 bei DP-Interfacemodul).
- Wenn Sie eine ET 200SP mit ET 200 AL-Modulen erweitern, dann ist das 1. AL-Modul immer in den Bytes 134 und 135 des Steuerdatensatzes kodiert, das 2. AL-Modul immer in den Bytes 136 und 137, usw., auch wenn nicht alle SP-Steckplätze mit SP-Modulen belegt sein sollten. Nicht-belegte SP-Steckplätze werden mit dem Wert "0" kodiert.

Fehlermeldungen

Beim Schreiben des Steuerdatensatzes 196 werden Ihnen im Fehlerfall folgende Fehlermeldungen zurückgegeben:

Tabelle 1-239 Fehlermeldungen

Fehlercode	Bedeutung
16#80A2	DP-Protokollfehler auf Layer 2. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.
16#80B1	Unzulässige Länge; Die Längenangabe im Datensatz 196 ist nicht korrekt.
16#80B5	Konfigurationssteuerung nicht parametriert.
16#80B2	invalid slot: Der projektierte Steckplatz ist nicht belegt.
16#80B8	Parameterfehler; Modul meldet ungültige Parameter.
16#80C5	DP-Slave oder Modul nicht verfügbar. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.

Rücklesedatensatz 197 für ET 200SP mit AL-Modulen

Der tatsächliche Aufbau einer ET 200SP mit AL-Modulen kann mithilfe des Rücklesedatensatzes 197 überprüft werden.

Dabei entspricht der Datensatz 197 weitgehend dem Rücklesedatensatz 197 für ET 200SP ohne AL-Module, ist allerdings länger, weil die zusätzlichen AL-Module ebenfalls kodiert werden müssen.

Für jedes Modul sind zwei Bytes im Datensatz vorgesehen. Die Position dieser beiden Bytes im Datensatz kodiert ein Modul bei der ursprünglichen Projektierung mit STEP 7.

Im folgenden Bild entsprechen

- Die Komponenten "status_slot_1" und "reserve_slot_1" (Byte 4 und 5 im Datensatz) dem Modul auf Steckplatz 1 in der Projektierung mit STEP 7,
- "status_slot_2" und "reserve_slot_2" (Byte 6 und 7) dem Modul auf Steckplatz 2,
- "status_slot_3" und "reserve_slot_3" (Byte 8 und 9) dem Modul auf Steckplatz 3,
- usw.

Der folgende Datensatz ist für eine Projektierung mit 65 SP-Modulen und 16 AL-Modulen aufgebaut. Deshalb steht im Element "block_length" des Datensatzes der Wert "166".

Wenn Sie in STEP 7 eine ET 200SP mit weniger AL-Modulen projektieren, dann verkürzt sich der Datenbaustein.

Wenn Sie in einer Projektierung weniger SP-Module verwenden, dann hat dies keinen Einfluss auf die Länge des Datensatzes 197 (bei einer Erweiterung der ET 200SP mit ET 200AL-Modulen).

Die Komponente "reserve_x" des Rücklesedatensatzes 197 ist für zukünftige Anwendungen reserviert.

Die Namen der Komponenten des Rücklesedatensatzes (zum Beispiel "status_slot_1") können Sie frei wählen.

Das folgende Bild zeigt den Anfang des Rücklesedatensatzes 197 für das Lesen des tatsächlichen Aufbaus einer ET 200SP mit AL-Modulen.

DB_read_back_ET_200SP_with_AL

	Name	Data type	Start value
1	Static		
2	block_length	USInt	166
3	block_ID	USInt	197
4	version	USInt	2
5	subversion	USInt	0
6	status_slot_1	USInt	1
7	reserve_slot_1	USInt	0
8	status_slot_2	USInt	1
9	reserve_slot_2	USInt	0
10	status_slot_3	USInt	1
11	reserve_slot_3	USInt	0
12	status_slot_4	USInt	1
13	reserve_slot_4	USInt	0
14	status_slot_5	USInt	1
15	reserve_slot_5	USInt	0
16	status_slot_6	USInt	1
17	reserve_slot_6	USInt	0

Nicht dargestellt sind die Komponenten "status_slot_7" bis "reserve_81" (Maximalausbau einer ET 200SP mit PN-Interfacemodul) bzw. "status_slot_7" bis "reserve_slot_49" (Maximalausbau einer ET 200SP mit DP-Interfacemodul).

Die Bedeutung von "status_slot_x":

- Der Wert "1" in satus_slot_x bedeutet, dass das Modul x auf dem vorgesehenen Steckplatz steckt
- Der Wert "0" kodiert alle anderen Möglichkeiten (falsches Modul, leerer Steckplatz, BU-Cover).

Das folgende Bild zeigt einen Ausschnitt aus dem Rückmeldedatensatzes 197 für das Lesen des tatsächlichen Aufbaus einer ET 200SP mit AL-Modulen (und PN-Interfacemodul). Bis zum Steckplatz 65 werden ET 200SP-Module gesteckt, dann kommen die AL-Module. Beispielsweise bedeutet der Wert "1" im Byte "status_slot_66_AL", dass das 1. AL-Modul tatsächlich in der Anlage vorhanden ist und auf dem richtigen Steckplatz steckt.

128	status_slot_62_SP	USInt	1
129	reserve_slot_62_SP	USInt	0
130	status_slot_63_SP	USInt	1
131	reserve_slot_63_SP	USInt	0
132	status_slot_64_SP	USInt	1
133	reserve_slot_64_SP	USInt	0
134	status_slot_65_SP	USInt	1
135	reserve_slot_65_SP	USInt	0
136	status_slot_66_AL	USInt	1
137	reserve_slot_66_AL	USInt	0
138	status_slot_67_AL	USInt	1
139	reserve_slot_67_AL	USInt	0
140	status_slot_68_AL	USInt	1
141	reserve_slot_68_AL	USInt	0
142	status_slot_69_AL	USInt	1
143	reserve_slot_69_AL	USInt	0

Rückmeldedatensatz 197 lesen

Mit der Anweisung RDREC können Sie den Rückmeldedatensatz 197 von der ET 200SP lesen. RDREC arbeitet asynchron. Wenn Sie RDREC im Anlauf-OB aufrufen, dann müssen Sie die Anweisung über eine Schleife mehrmals aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz gelesen ist.

Weiterführende Informationen und Beispiele

Konkrete Beispiele für die Konfigurationssteuerung finden Sie in dieser Applikationsbeschreibung (http://support.automation.siemens.com/WW/view/de/29430270).

Weitere Informationen zur ET 200SP finden Sie in den Geätehandbüchern zur IM 155-6 PN (http://support.automation.siemens.com/WW/view/de/73184046) und IM 155-6 DP (http://support.automation.siemens.com/WW/view/de/73098660)

Weitere Informationen zur ET 200AL finden Sie hier. (http://support.automation.siemens.com/WW/view/de/89254863)

Siehe auch

Konfigurationssteuerung bei ET 200SP (Seite 3384)

Konfigurationssteuerung bei ET 200AL (Seite 3431)

ET 200SP mit ET 200AL-Modulen erweitern

Einleitung

Die ET 200SP ist ein dezentrales Peripheriesystem für den Einbau in einem Schaltschrank.

Das System lässt sich mit Modulen der Baureihe ET 200AL erweitern, die in Schutzart IP65/67 ausgeführt sind. ET 200AL-Module können vor Ort montiert werden, zum Beispiel an einer Maschine.

Steckplatzregeln

Für den Einsatz von Modulen der ET 200AL in einer ET 200SP mit Interfacemodul gelten folgende Regeln:

- Auf dem Steckplatz 1 der ET 200SP muss das Modul "BA-Send 1xFC" stecken, wenn die ET 200SP mit ET 200AL-Modulen erweitert werden soll.
- Die ET 200AL-Module müssen lückenlos projektiert sein.

In dem weiter unten beschriebenen Vorgehen wird auf den Einsatz von Modulen der ET 200AL in einer ET 200SP mit Interfacemodul eingegangen.

Für den Einsatz von Modulen der ET 200AL in einer ET 200SP mit CPU gelten folgende abweichende Regeln:

- Das Modul "BA-Send 1xFC" darf auf den Steckplätzen 2 bis 4 der ET 200SP CPU stecken
- Ein Kommunikationsmodul CM DP darf als Submodul der CPU gesteckt sein, aber belegt dabei Steckplatz 2.

Vorgehen

Um eine ET 200SP mit Interfacemodul mit ET 200AL-Modulen zu projektieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie ein Interfacemodul (PROFINET oder PROFIBUS) der Baureihe ET 200SP mit Drag & Drop in die Netzsicht.
2. Wechseln Sie in die Gerätesicht. Klicken Sie dazu zweimal auf das Symbol des gerade eingefügten Moduls.

3. Fügen Sie auf Steckplatz 1 der ET 200SP das Modul "BA Send 1xFC" ein.
   STEP 7 erzeugt nun ein ET-Connection-Rack mit 16 Steckplätzen für ET 200AL-Module (folgendes Bild).
   Ein ET-Connnection-Rack ist ein virtuelles Rack, aus dem die Reihenfolge der angeschlossenen ET 200AL-Module hervorgeht.
   Über den Steckplätzen werden Fragezeichen angezeigt, weil noch kein ET 200AL-Modul an BA-Send angeschlossen ist.

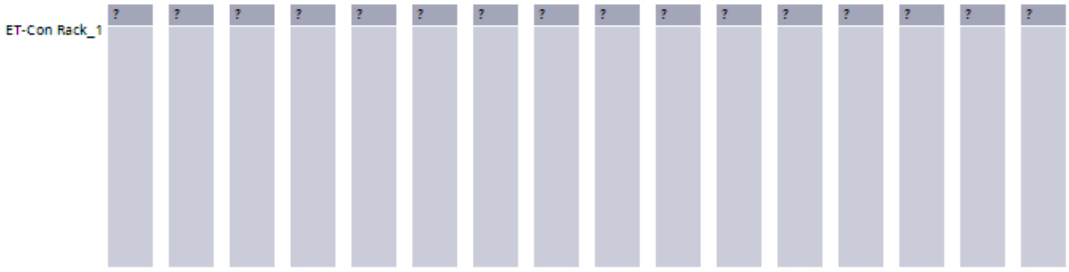

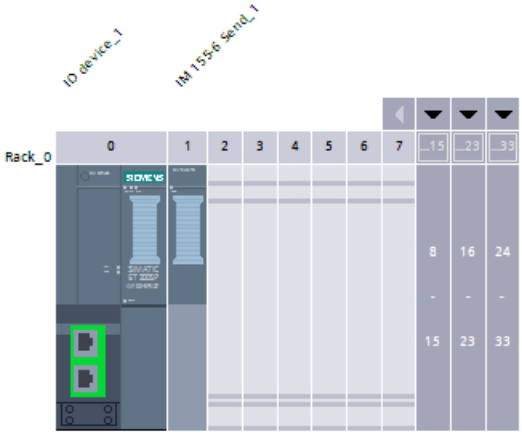

4. Wählen Sie aus dem Hardware-Katalog (Unterordner ET 200AL im Ordner ET 200SP) das erste ET 200AL-Modul, das an die ET 200SP angeschlossen sein soll: Ziehen Sie dieses Modul per Drag & Drop auf den 1. Steckplatz des ET-Connection-Racks.
   Von diesem Modul ausgehend erzeugt STEP 7 eine Linie zum Modul "BA-Send 1xFC" und vergibt die Steckplatznummern 66 bis 81 (folgendes Bild).
   Wenn Sie die ET 200SP mit einem DP-Interfacemodul projektieren, dann vergibt STEP 7 die Steckplatznummern 34 bis 49 für die ET 200AL-Module.

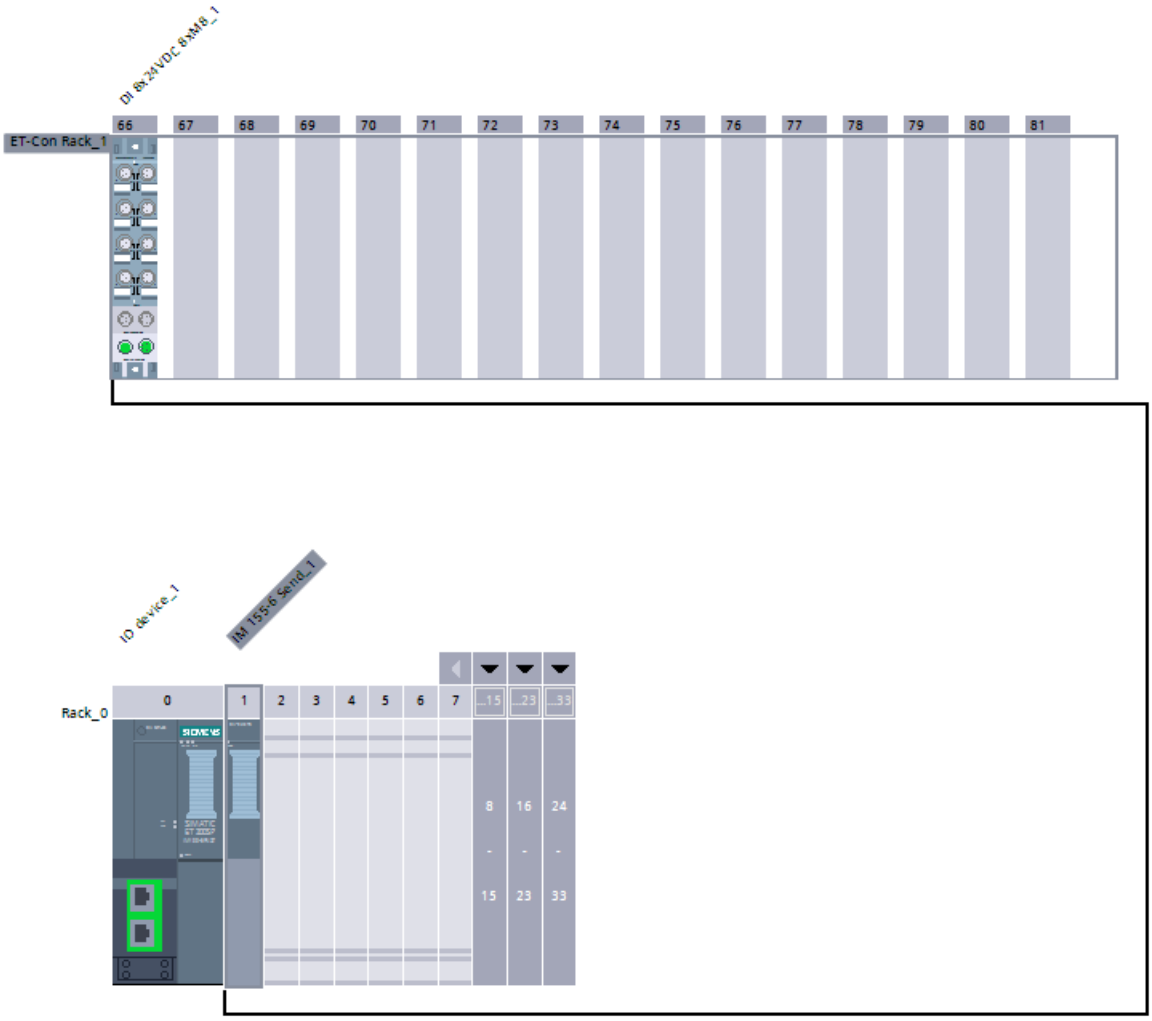

5. Ziehen Sie nun alle weiteren ET 200AL-Module per Drag & Drop auf die freien Steckplätze im ET-Connection-Rack.
   STEP 7 fügt automatisch die ET-Connection-Verbindung zwischen den einzelnen ET 200AL-Modulen ein (grüne Schleifen).
   In der folgenden Projektierung sind fünf ET 200AL-Module hintereinander angeschlossen.

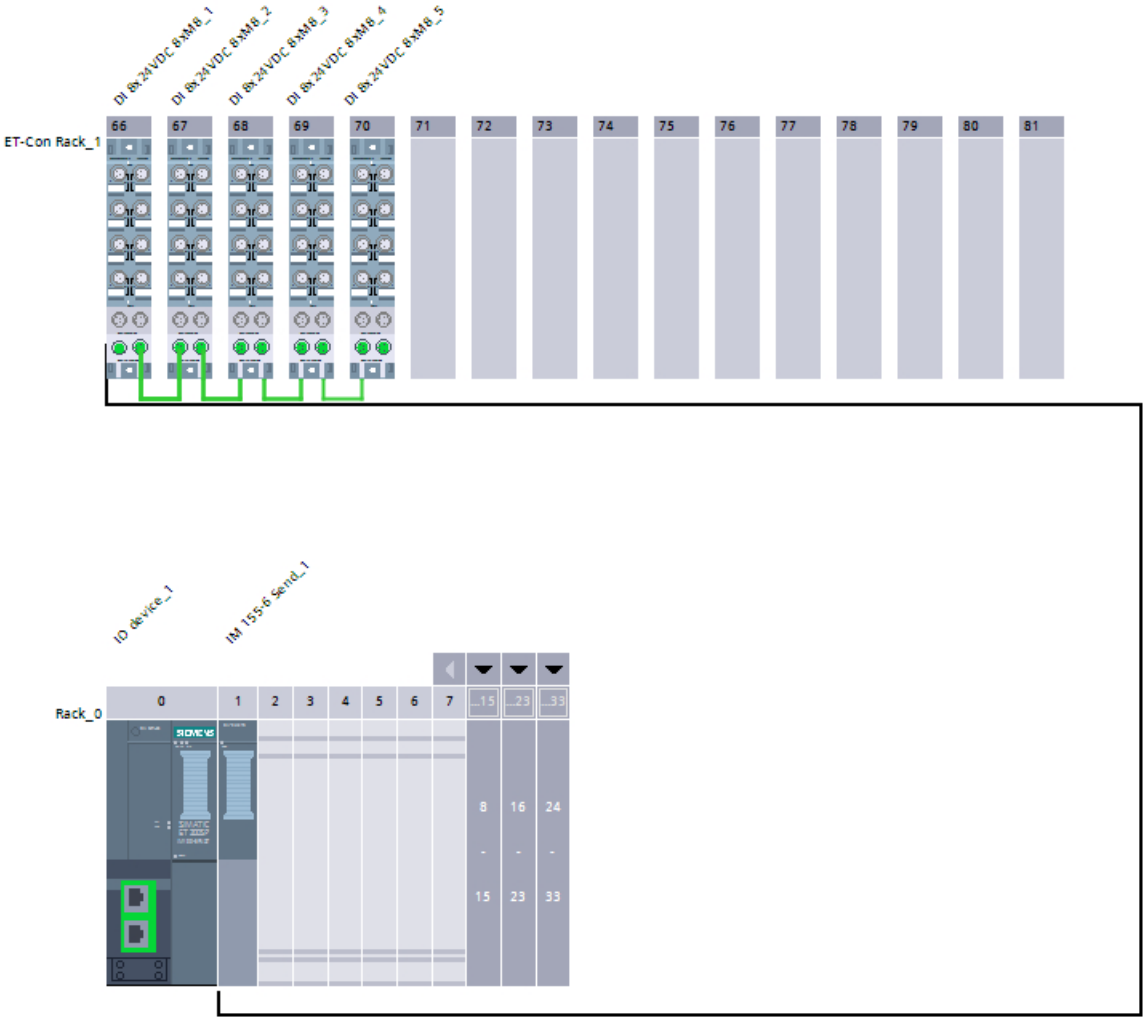

6. Vervollständigen Sie die Projektierung der ET 200SP: Ziehen Sie alle ET 200SP-Module vom Hardware-Katalog zu den Steckplätzen in der ET 200SP.
   In der folgenden Beispiel-Projektierung sind fünf ET 200SP-Module gesteckt: Auf Steckplatz 0 das PN-Interfacemodul, auf Steckplatz 1 das Modul "BA-Send 1xFC", auf den Steckplätzen 2 und 3 jeweils ein Eingabemodul und auf Steckplatz 4 ein Servermodul:

Siehe auch

Konfigurationssteuerung bei ET 200AL (Seite 3431)

Dezentrales Peripheriesystem ET 200AL (Seite 3424)

Parameter Ausgabemodule

Ersatzwertverhalten

Ersatzwertverhalten

Im ET 200SP wird das Ersatzwertverhalten durch den IO-Controller je Steckplatz durchgeführt.

Der jeweilige Ausgang verhält sich nach seinem parametrierten Ersatzwertverhalten:

- "Abschalten"
- "Ersatzwert ausgeben"
- "Letzten Wert halten"

Das Ersatzwertverhalten wird in folgenden Fällen ausgelöst:

- IO-Controller in STOP
- Controller-Ausfall (Verbindungsunterbrechung)
- Firmware-Update
- Rücksetzen auf Werkseinstellungen
- Ziehen von mehr als einem Peripheriemodul gleichzeitig
- Deaktivieren des IO-Devices
- Stationsstopp
  - fehlendes Servermodul
  - Ziehen von mehr als einem Peripheriemodul gleichzeitig
  - mindestens ein Peripheriemodul steckt auf einem falschen BaseUnit

Hinweis

Verkleinern einer Konfiguration

Wenn Sie die Konfiguration des ET 200SP verkleinern und die Projektierung in die CPU laden, dann behalten die nicht mehr projektierten, aber noch vorhandenen Module ihr ursprüngliches Ersatzwertverhalten bei. Dies gilt bis zum Abschalten der Versorgungsspannung am BaseUnit BU...D oder am Interfacemodul.

Parameter Eingabemodule

Parameter der Digitaleingabemodule

Diagnose fehlende Versorgungsspannung L+

Freigabe der Diagnosemeldung bei fehlender oder zu geringer Versorgungsspannung L+.

Diagnose Kurzschluss nach M

Freigabe der Diagnose, wenn ein Kurzschluss der Geberversorgung nach M auftritt.

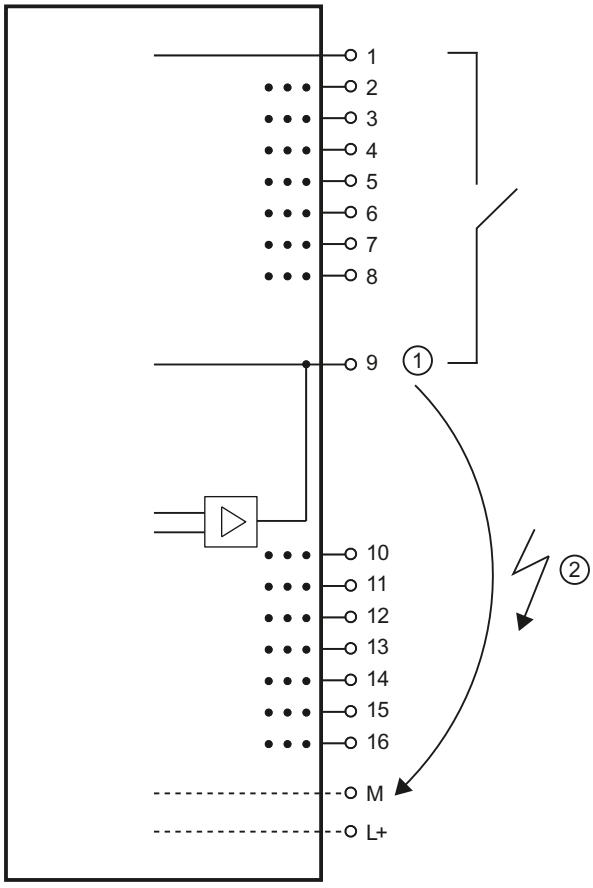

① Geberversorgung
② Kurzschluss

Diagnose Kurzschluss nach L+

Freigabe der Diagnose, wenn ein Kurzschluss der Geberversorgung nach L+ auftritt.

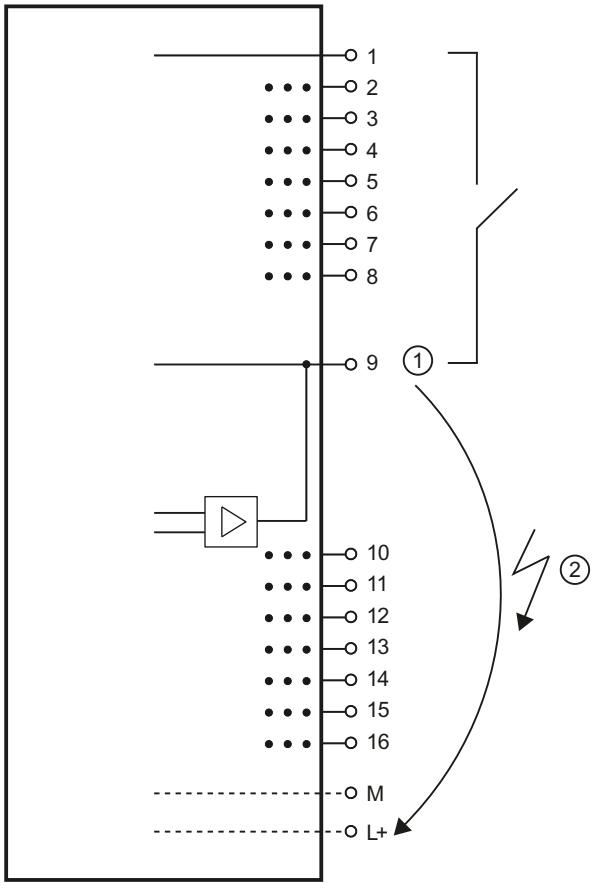

① Geberversorgung
② Kurzschluss

Diagnose Drahtbruch

Freigabe der Diagnose, wenn die Leitung zum Geber unterbrochen ist.

Betriebsart

Legt fest, ob ein Kanal aktiviert oder deaktiviert ist.

Impulsverlängerung (nur High Feature Module)

Die Impulsverlängerung ist eine Funktion zur Veränderung eines digitalen Eingangssignals. Ein Impuls an einem Digitaleingang wird mindestens auf die parametrierte Länge verlängert. Ist der Eingangsimpuls bereits länger als die parametrierte Länge, dann wird der Impuls nicht verändert.

Die Impulsverlängerung wird immer dann gestartet, wenn sich der Zustand des Eingangssignals ändert und keine Impulsverlängerung für diesen Kanal aktiv ist.

Potentialgruppe des linken Moduls/Neue Potentialgruppe

Legt fest, ob sich das Peripheriemodul auf einem BaseUnit mit Einspeisung der Versorgungsspannung befindet (Neue Potentialgruppe) oder ob es sich auf einem BaseUnit ohne Einspeisung der Versorgungsspannung befindet (es gehört dann zur Potentialgruppe des linken Moduls).

Parameter der Analogeingabemodule

Fehlende Versorgungsspannung L+

Freigabe der Diagnose, bei fehlender oder zu geringer Versorgungsspannung L+.

Vergleichsstelle (AI 4xRTD/TC 2-/3-/4-wire HF)

Als Vergleichsstelle kann ein BaseUnit mit internen Temperatursensor (BU..T) bzw. der Kanal 0 des Peripheriemoduls verwendet werden, wenn dieser als "Thermowiderstand Pt100 Klimabereich Celsius" parametriert wurde.

Nachfolgend ist eine mögliche Parametrierung dargestellt (siehe auch Wissenswertes zum Referenzkanalbetrieb (Seite 3408)):

Tabelle 1-240 RTD-Kanal

Einstellung	Beschreibung
Kein Referenzkanalbetrieb	Anliegender Temperaturwert am Kanal 0 kann bei entsprechender Parametrierung der anderen Kanäle als modulweiter Referenzwert verwendet werden.
Referenzkanal der Gruppe x	Der Kanal wirkt als Sender für die Vergleichsstellentemperatur von Gruppe x. Die Verteilung erfolgt über das Interfacemodul.

Tabelle 1-241 TC-Kanal

Einstellung	Beschreibung
Referenzkanal des Moduls	Der entsprechende TC-Kanal nutzt den Kanal 0 desselben Moduls als Vergleichsstellentemperatur. Dieser muss als "Thermowiderstand Pt 100 Klimabereich Celsius" und "Kein Referenzkanalbetrieb" parametriert sein, sonst wird Diagnose Vergleichsstelle ausgelöst.
Interne Vergleichsstelle	Die Vergleichsstellentemperatur wird von einem internen Temperatursensor auf der BaseUnit gelesen. Bei falschem BaseUnit-Typ wird Diagnose Vergleichsstelle ausgelöst.
Referenzkanal der Gruppe x	Bei der Einstellung "TC" (Thermoelement…) wirkt der Kanal als Empfänger für die Vergleichsstellentemperatur von Gruppe x.
Feste Referenztemperatur	Es erfolgt keine Temperaturkompensation. Die Linearisierung wird mit einer angenommenen Vergleichsstellentemperatur von 0 °C durchgeführt.

Überlauf

Freigabe der Diagnose, wenn der Messwert den Überlaufbereich überschreitet.

Unterlauf

Freigabe der Diagnose, wenn der Messwert den Unterlaufbereich unterschreitet.

Drahtbruch

Freigabe der Diagnose, wenn das Modul am entsprechend parametrierten Eingang keinen Stromfluss bzw. zu geringen Strom für die Messung hat.

Glättung

Die einzelnen Messwerte werden mittels digitaler Filterung geglättet. Die Glättung ist in 4 Stufen einstellbar, wobei der Glättungsfaktor k multipliziert mit der Zykluszeit des Peripheriemoduls der Zeitkonstante des Glättungsfilters entspricht. Je größer die Glättung, umso größer ist die Zeitkonstante des Filters.

Nachfolgendes Bild zeigt die Sprungantwort bei den verschiedenen Glättungsfaktoren in Abhängigkeit von der Anzahl der Baugruppenzyklen.

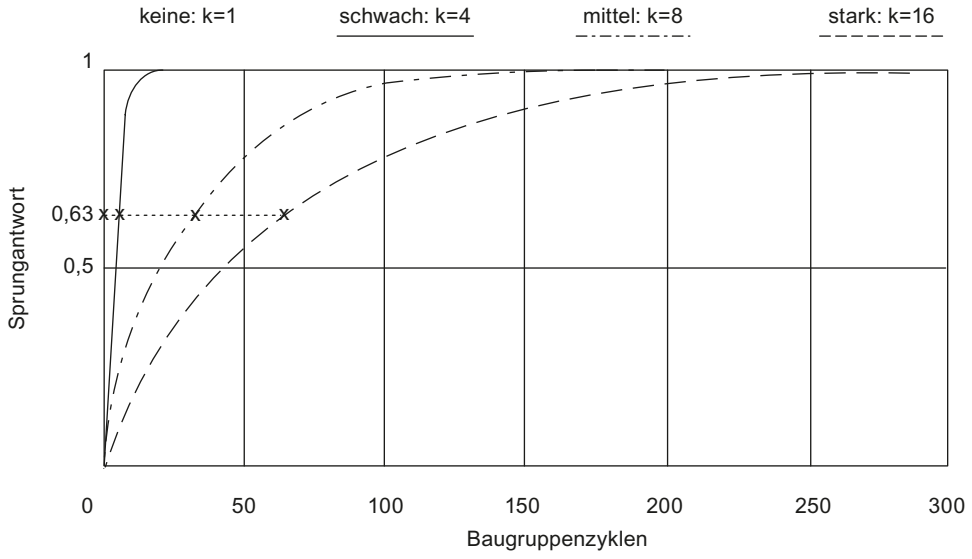

Bild 1-130 Glättung bei AI 4×RTD/TC 2-/3-/4-wire HF

Störfrequenzunterdrückung

Unterdrückt bei Analogeingabebaugruppen die Störungen, die durch die Frequenz des verwendeten Wechselspannungsnetzes hervorgerufen werden.

Die Frequenz des Wechselspannungsnetzes kann sich besonders bei der Messung in kleinen Spannungsbereichen und bei Thermoelementen störend auf den Messwert auswirken. Mit diesem Parameter gibt der Anwender die Netzfrequenz an, die in seiner Anlage vorherrscht.

Grenzwerte Prozessalarm

Wenn die obere Grenze 1/2 überschritten oder die untere Grenze 1/2 unterschritten wird, löst das Modul einen Prozessalarm aus.

Nachfolgend einige Beispiele für die Wahl der Grenzwerte1 und 2.

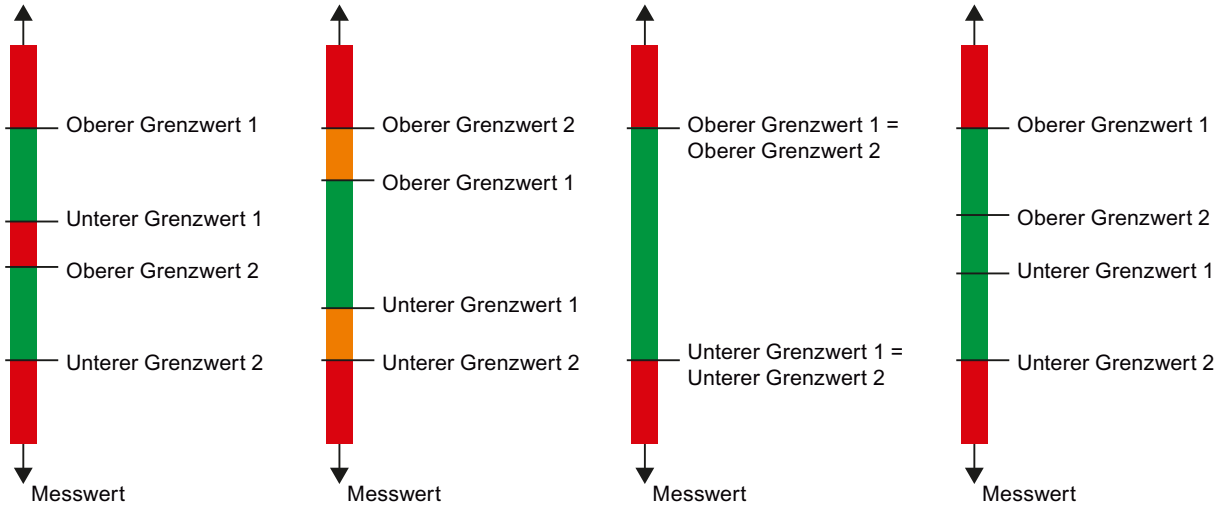

Unterer Grenzwert 1/2

Legen Sie eine Schwelle fest, bei deren Unterschreitung ein Prozessalarm ausgelöst wird.

Oberer Grenzwert 1/2

Legen Sie eine Schwelle fest, bei deren Überschreitung ein Prozessalarm ausgelöst wird.

Potenzialgruppe des linken Moduls/Neue Potenzialgruppe

Legt fest, ob sich das Peripheriemodul auf einem BaseUnit mit Einspeisung der Versorgungsspannung befindet (Neue Potenzialgruppe) oder ob es sich auf einem BaseUnit ohne Einspeisung der Versorgungsspannung befindet (es gehört dann zur Potenzialgruppe des linken Moduls).

Temperaturkoeffizient (Messart Thermowiderstand)

Der Korrekturfaktor für den Temperaturkoeffizienten (α-Wert) gibt an, um wieviel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials. In Europa wird pro Sensorart nur ein Wert verwendet (voreingestellter Wert).

Die weiteren Werte ermöglichen eine sensorspezifische Einstellung des Temperaturkoeffizienten und somit eine noch höhere Genauigkeit.

Siehe auch

Besonderheiten bei AI 4xRTD/TC 2-/3-/4-wire HF (Seite 3415)

Wissenswertes zum Referenzkanalbetrieb

Ein RTD/TC-Modul der ET 200SP arbeitet dann im Referenzkanalbetrieb, wenn ein Kanal die Referenztemperatur an andere Kanäle der Station sendet. Die empfangenden Kanäle verwenden die Referenztemperatur für die Temperaturkompensation bei der Messung mit Thermoelementen.

Aufbau und Verwendung von Thermoelementen

Ein Thermoelement besteht aus zwei Drähten mit unterschiedlichen Metallen oder Legierungen, die an einem Ende miteinander verschweißt sind. Die Schweißstelle wird Messstelle genannt.

Das andere Ende der beiden Drähte ist offen. Dieses Ende wird Vergleichsstelle genannt.

An der Messstelle entsteht zwischen den beiden Metallen/Legierungen eine Thermospannung, die abhängig von der Temperatur an der Messstelle ist. Weiterhin treten an der Vergleichsstelle - beim Übergang vom Thermoelement auf z. B. Kupferleitungen - erneut Thermospannungen auf, die den eigentlichen Messwert verfälschen und kompensiert werden müssen. Bei einer Vergleichsstellentemperatur von 0 °C ist keine Kompensation notwendig.

Für die Kompensation der Vergleichsstellentemperatur gibt es unterschiedliche Techniken:

- Feste Referenztemperatur: Die Vergleichsstelle ist fest auf eine bestimmte Temperatur eingestellt, zum Beispiel durch ein Eiswasserbad auf 0 °C (keine Kompensation erforderlich).
- Interne Vergleichsstelle: Die Vergleichsstelle ist die Klemme der BaseUnit, auf der das Analogmodul steckt. Wenn Sie die Kompensationsart "Interne Vergleichsstelle" wählen, müssen Sie BaseUnits mit integrierter Temperaturmessung zur Kompensation der Vergleichsstellentemperatur verwenden. Diese BaseUnits haben die Bezeichnung "BU..T". Das Modul erfasst die Temperatur an der Vergleichsstelle und bestimmt damit die tatsächliche Temperatur an der Messstelle.
- Referenzkanal der Gruppe x: Ein externer Thermowiderstand erfasst die Temperatur an der Vergleichsstelle für die Gruppe x (Gruppe von Kanälen innerhalb einer Station). Dadurch ist die tatsächliche Temperatur an der jeweiligen Messstelle bestimmbar. Pro Gruppe ist ein externer Thermowiderstand erforderlich. Die Thermowiderstände werden jeweils an einen Kanal eines Analogmoduls angeschlossen. Diese Kanäle werden als Sender (für die Temperatur an der Vergleichsstelle) bezeichnet.
  Funktionsweise und Einstellungen sind im Abschnitt "Stationsweite Verteilung der Referenztemperatur" beschrieben.
- Referenzkanal des Moduls: Die Funktionsweise ist vergleichbar mit "Referenzkanal der Gruppe x". Sie schließen einen externen Thermowiderstand an Kanal 0 des Moduls an zur Messung der Temperatur an der Vergleichsstelle. Andere Kanäle desselben Moduls nutzen diese Referenztemperatur zur Temperaturkompensation.
  Funktionsweise und Einstellungen sind im Abschnitt "Modulweite Verteilung der Referenztemperatur" beschrieben.

Informationen zum Aufbau und zur Arbeitsweise eines Thermoelements finden Sie im Handbuch Analogwertverarbeitung (http://support.automation.siemens.com/WW/view/de/67989094).

Stationsweite Verteilung der Referenztemperatur

Sie können die Temperatur an der Vergleichsstelle x mithilfe eines Thermowiderstands an einem Kanal (Sender der Referenztemperatur) erfassen und an andere Kanäle innerhalb einer Station (Empfänger) senden. Alle Kanäle, welche die Temperatur an einer Vergleichsstelle x empfangen, bilden die Gruppe x.

Pro Gruppe parametrieren Sie genau einen Kanal als Sender der Referenztemperatur.

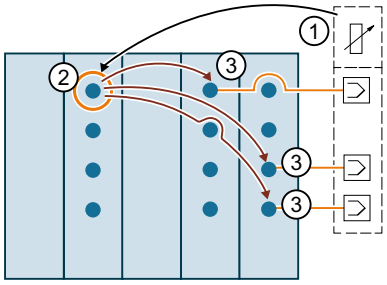

① Thermowiderstand an der Vergleichsstelle

② Kanal erfasst Temperatur an der Vergleichsstelle und sendet sie an andere Kanäle innerhalb einer Station (Sender der Gruppe x). Der Temperaturwert dient zur Kompensation der Vergleichsstellentemperatur.

③ Kanäle der Gruppe x erhalten die Temperatur an der Vergleichsstelle (Empfänger)

Parametrierung eines Kanals als Referenzkanal (Sender für Gruppe 1)

Im Folgenden ist die Parametrierung am Beispiel der Gruppe 1 dargestellt:

1. Öffnen Sie das Projekt in STEP 7
2. Markieren Sie in der Gerätesicht das gewünschte Analogmodul (RTD/TC).
3. Wählen Sie anschließend einen Kanal, der als Sender der Vergleichsstellentemperatur arbeiten soll.
   Die folgenden Einstellungen sind erforderlich:
   "Messart": "Thermowiderstand“, zum Beispiel "Thermowiderstand (4-Leiteranschluss)"
   "Messbereich": "Pt 100 Klimabereich"
   "Temperatureinheit": "Grad Celsius"
   "Vergleichsstelle": "Referenzkanal der Gruppe 1"

Das folgende Bild zeigt die Parametrierung.

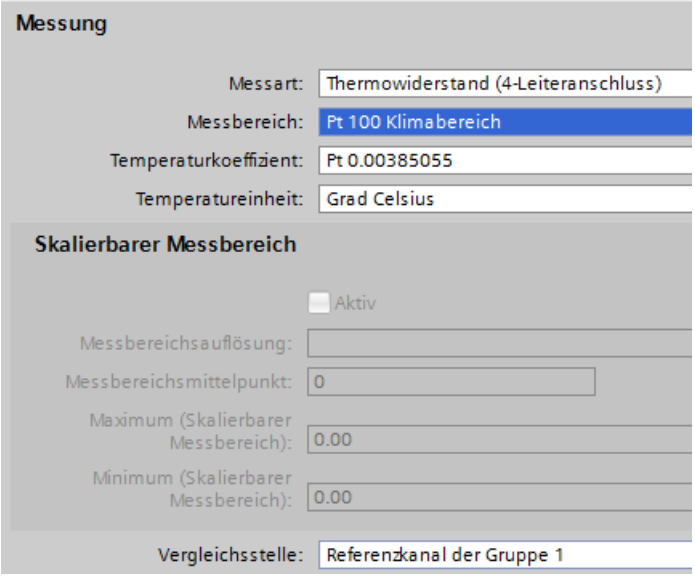

Der so konfigurierte Kanal (Messart Thermowiderstand) arbeitet als Referenzkanal der Gruppe 1 und sendet die gemessene Temperatur zu allen Kanälen (Messart Thermoelement), die als Empfänger der Gruppe 1 parametriert sind.

Wie Kanäle zu parametrieren sind, die Empfänger der Gruppe 1 sind, erfahren Sie im nächsten Abschnitt.

Parametrierung eines Kanals als Empfänger der Gruppe 1

Das folgende Bild zeigt die Parametrierung eines Kanals, der die Temperatur an der Vergleichsstelle der Gruppe 1 empfängt.

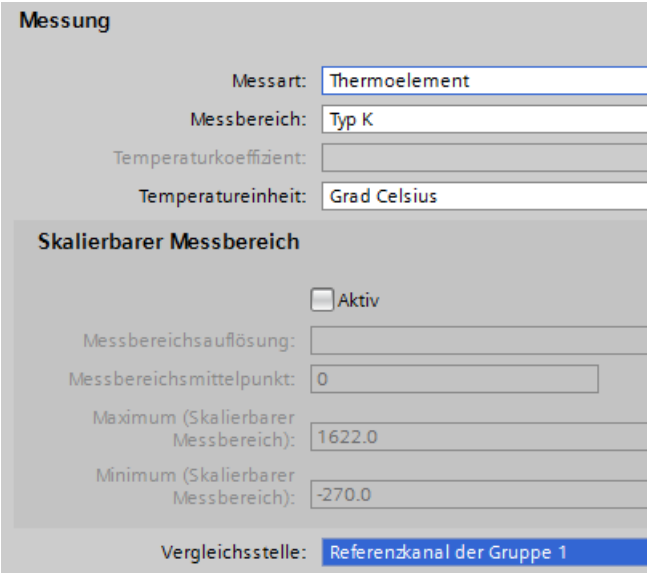

Die folgenden Einstellungen sind erforderlich:

- "Messart“: "Thermoelement"
- "Vergleichsstelle": "Referenzkanal der Gruppe 1"

Modulweite Verteilung der Referenztemperatur

Sie können mit dem Kanal 0 eines Moduls die Temperatur an einer Vergleichsstelle erfassen und den Temperaturwert für die Kanäle 1, 2, 3 … dieses Moduls verwenden. Dabei wird der erfasste Temperaturwert nicht an Kanäle anderer Module innerhalb der Station gesendet (kein Referenzkanalbetrieb).

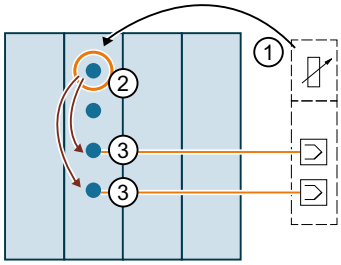

① Thermowiderstand an der Vergleichsstelle

② Kanal erfasst Temperatur an der Vergleichsstelle, Parameter "Vergleichsstelle": "Kein Referenzkanalbetrieb"

③ Kanäle desselben Moduls nutzen den Temperaturwert zur Kompensation der Vergleichsstellentemperatur für die Messung mit Thermoelementen, Parameter "Vergleichsstelle": "Referenzkanal des Moduls"

Parametrierung von Kanal 0 als Referenzkanal des Moduls

Das folgende Bild zeigt die Parametrierung des Kanals 0 eines Moduls, der für die Erfassung der Temperatur an der Vergleichsstelle genutzt werden soll:

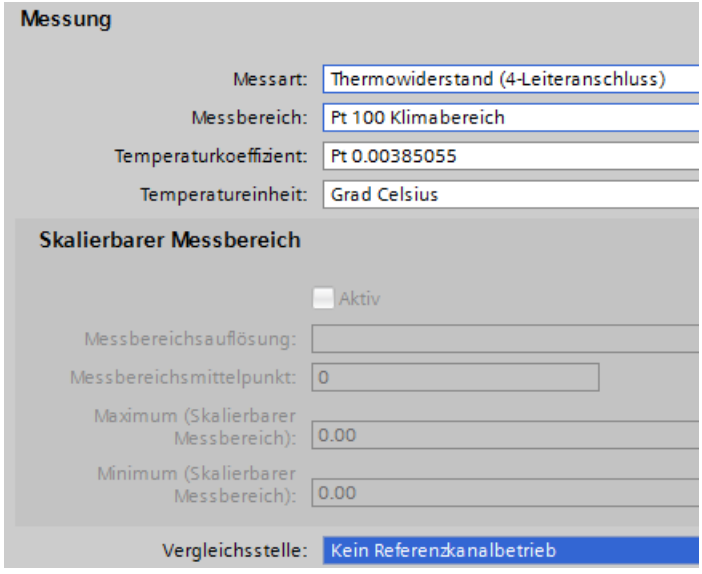

Die folgenden Einstellungen sind erforderlich:

- "Messart": "Thermowiderstand“, zum Beispiel "Thermowiderstand (4-Leiteranschluss)"
- "Messbereich": "Pt 100 Klimabereich"
- "Temperatureinheit": "Grad Celsius"
- "Vergleichsstelle": "Kein Referenzkanalbetrieb"

Parametrierung eines Kanals, der Kanal 0 als Referenzkanal nutzt

Das folgende Bild zeigt, wie ein Kanal parametriert werden muss, der den Kanal 0 dieses Moduls als Referenzkanal zur Temperaturkompensation nutzt.

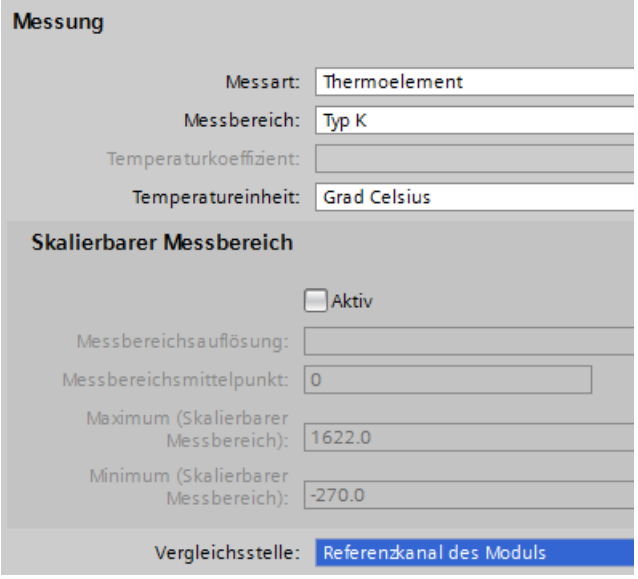

Die folgenden Einstellungen sind für die Kanäle des Moduls, die mithilfe des Kanals 0 die Temperatur der Vergleichsstelle kompensieren, erforderlich:

- "Messart": "Thermoelement"
- "Vergleichsstelle": "Referenzkanal des Moduls"

Siehe auch

Parameter der Analogeingabemodule (Seite 3405)

Wissenswertes zur Funktion Oversampling

Für hohe Anforderungen bezüglich Leistung und Geschwindigkeit stehen High-Speed-Analogmodule (HS) zur Verfügung. Hauptmerkmal dieser HS-Analogmodule gegenüber Standard-Analogmodulen (ST) sind kürzere Zykluszeiten. Um das zu erreichen, verfügen die Ein- und Ausgabemodule über Bauelemente mit extrem kurzen Durchlauf- und Wandlungszeiten. Zudem ist die gesamte Architektur der Module auf eine schnelle Signalverarbeitung ausgelegt.

HS-Analogmodule wandeln die Mess- und Ausgabewerte parallel. Jeder Kanal innerhalb des Moduls besitzt einen eigenen AD- bzw. DA-Umsetzer. So entspricht die Zykluszeit im Wesentlichen der Wandlungszeit und ist unabhängig von der Anzahl der aktivierten Kanäle. Das gilt sowohl für die Analogein- als auch für die Analogausgabe. Somit ist es möglich, HS-Module im schnellen taktsynchronen Betrieb einzusetzen.

Neben der Taktsynchronität bieten HS-Analogmodule auch im nicht-taktsynchronen (freilaufenden) Betrieb Vorteile. Durch die schnelle Verarbeitung der Prozesssignale sind HS-Analogmodule in der Lage, Änderungen in den Prozesswerten schneller zu erfassen und auf diese Ereignisse mit den geeigneten Programmbausteinen (z. B. Prozessalarm- oder Weckalarm-Organisationsbausteine) zu reagieren.

Taktsynchronität

Unter Taktsynchronität versteht man die synchrone Kopplung

- der Signalerfassung und -ausgabe durch die dezentrale Peripherie
- der Signalübertragung via PROFIBUS bzw. PROFINET
- der Programmbearbeitung an den Takt des äquidistanten PROFIBUS bzw. PROFINET.

Somit entsteht ein System, das in konstanten Zeitabständen seine Eingangssignale erfasst, bearbeitet und die Ausgangssignale ausgibt. Taktsynchronität gewährleistet genau reproduzierbare und definierte Prozessreaktionszeiten sowie äquidistante und synchrone Signalverarbeitung bei dezentraler Peripherie.

Bei projektierter Taktsynchronität arbeiten das Bussystem und die Peripheriemodule synchron. Die übertragenen Ein- und Ausgabedaten sind an eine "taktsynchrone Task" in der CPU gekoppelt. Dadurch sind die Daten eines Zyklus immer konsistent. Alle Daten des Prozessabbildes gehören logisch und zeitlich zusammen. Jitter im Anwenderprogramm, die sich durch eine Erfassung unterschiedlich alter Werte ergeben, sind somit nahezu ausgeschlossen.

Durch die genaue zeitliche Reproduzierbarkeit aller Abläufe lassen sich auch schnelle Vorgänge sicher beherrschen. Taktsynchronität trägt zu einer hohen Regelungsgüte und damit zu einer größeren Fertigungsgenauigkeit bei. Gleichzeitig werden mögliche Schwankungen der Prozessreaktionszeiten drastisch reduziert. Die zeitlich gesicherte Bearbeitung kann für einen höheren Maschinentakt genutzt werden. Kürzere Taktzeiten erhöhen die Verarbeitungsgeschwindigkeit und tragen somit auch zu einer Reduzierung der Produktkosten bei.

Oversampling

Die Verwendung der Oversampling-Funktion in Analogeingabe- bzw. Analogausgabemodulen setzt eine taktsynchrone Projektierung voraus.

Bei Analogeingabemodulen wird der eingestellte Sendetakt in zeitäquidistante Subtakte unterteilt. Die Sendetakte lassen sich in 2 bis 16 Subtakte unterteilen. Jeder Subtakt liest einen Messwert ein. Die eingelesenen Messwerte eines Datenzyklus werden im darauf folgenden Sendetakt in das Interfacemodul (IM) kopiert und stehen einen Takt später der verarbeitenden CPU zur Verfügung.

Bei Analogausgabemodulen wird der eingestellte Sendetakt ebenfalls in zeitäquidistante Subtakte unterteilt. Die Sendetakte lassen sich in 2 bis 16 Subtakte unterteilen. Jeder Subtakt gibt einen Ausgabewert aus. Innerhalb eines Sendetakts werden die Ausgabewerte von der

CPU in das Interfacemodul kopiert und einen weiteren Sendetakt später zum Prozess herausgeschrieben.

Die eingelesenen bzw. ausgegebenen Werte werden in den Nutzdaten des Analogmoduls übertragen. Dadurch vergrößert sich der Adressraum des Moduls von 2 Byte Nutzdaten pro Kanal auf 16 x 2 Byte Nutzdaten pro Kanal (bei 16 Subtakten). Wenn Sie den Sendetakt in weniger als 16 Subtakte unterteilen, werden bei der Eingabe die nicht genutzten Adressen mit dem Fehlerwert 0x7FFF befüllt. Bei der Ausgabe werden die Werte der nicht genutzten Adressen ignoriert.

Da die Subtakte prinzipiell innerhalb eines Sendetakts liegen müssen, benötigt Oversampling im Gegensatz zum 3-Zyklen-Modell der Taktsynchronität jeweils einen zusätzlichen Takt zum Kopieren der Daten in das Interfacemodul. Dadurch ergibt sich ein 5-Zyklen-Modell.

Taktsynchronität

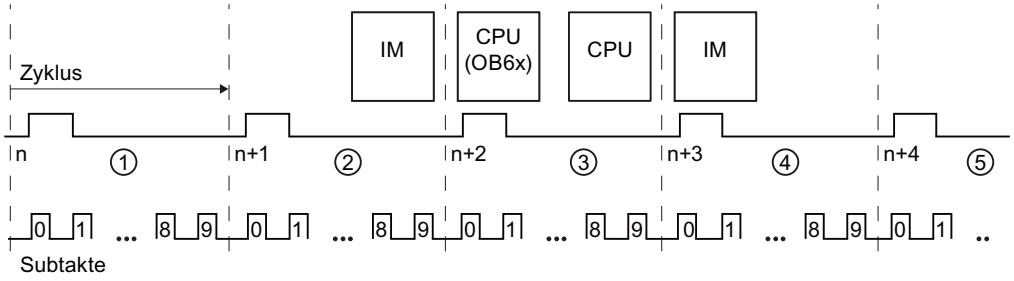

① Sendetakt wird in Subtakte unterteilt, die jeweils den Messwert erfassen (hier: 10 Subtakte)

② Messwerte werden ins IM kopiert

③ Messwerte werden verarbeitet und Ausgabewerte ermittelt

④ Ausgabewerte werden von der CPU ins IM kopiert

⑤ Ausgabewerte werden zum Prozess herausgeschrieben

Höhere Abtastraten durch Oversampling

Ein IO-Device besitzt aufgrund der projektierten Module eine minimal mögliche Aktualisierungszeit. Innerhalb dieser Aktualisierungszeit wird das IO-Device/der IO-Controller des PROFINET IO-Systems einmalig mit neuen Daten versorgt.

Bezogen auf die Kanäle eines einzelnen Standard-Moduls im IO-Device gilt: Die kürzeste Aktualisierungszeit ("Abtastrate") ist genau ein Sendetakt.

Wenn Sie die Aktualisierungszeit für die Kanäle eines Standard-Moduls verkürzen wollen, müssen Sie den Sendetakt verkürzen. Das ist aufgrund der Eigenschaften der beteiligten Komponenten und des Aufbaus des IO-Systems nur bis zu einem gewissen Grad möglich (z. B. bis hinunter zu 0,25 ms).

Module mit Oversampling bieten jedoch die Möglichkeit, die Aktualisierungszeit ("Abtastrate") für deren Kanäle noch weiter zu minimieren, ohne dass dabei der Sendetakt für das gesamte IO-Device verkürzt werden muss.

Die Unterteilung des Sendetakts in zeitäquidistante Subtakte ermöglicht die Bearbeitung schneller Prozesse durch höhere Abtastraten.

Beispiel

In der Praxis bietet sich die Verwendung von Oversampling dann an, wenn die taktsynchrone Anlage aufgrund der verwendeten Module nur mit einem bestimmten Sendetakt arbeitet (z. B.

1 ms), die Prozesswerte jedoch schneller abgetastet werden müssen. Mit Hilfe von Oversampling und einer Unterteilung des Sendetakts in z. B. 4 Subtakte, können Sie die Prozesswerte alle 250 µs abtasten.

Oversampling projektieren

Aktivieren Sie in dem von Ihnen verwendeten IO-Device und den taktsynchron betriebenen Modulen die Option "Taktsynchroner Betrieb" und setzen Sie die dazugehörigen Parameter ("Sendetakt", etc.).

Bei den dezentralen Analogeingabemodulen (z. B. AI 2xU/I 2,4-wire HS) legen Sie die Anzahl der Subtakte mit dem Parameter "Abtastrate" fest.

Bei dezentralen Analogausgabemodulen (z. B. AQ 2xU/I HS) legen Sie die Anzahl der Subtakte mit dem Parameter "Ausgaberate" fest.

Wenn Sie z. B. eine "Abtastrate" von 4 "Werte/Zyklus" bei einem Sendetakt von 1 ms projektieren, wird der Sendetakt in 4 Subtakte unterteilt und die Prozesswerte alle 250 µs abgetastet.

Verweis

Weitere Informationen finden Sie in den Gerätehandbüchern zu den High-Speed-Analogmodulen sowie im Funktionshandbuch Analogwertverarbeitung (http://support.automation.siemens.com/WW/view/de/67989094).

Besonderheiten bei AI 4xRTD/TC 2-/3-/4-wire HF

Einsatz von Cu10-Sensoren

- Wählen Sie in der Parametrierung "Thermowiderstand 3-Leiter" und "Cu10".
- Verdrahten Sie den Cu10-Sensor in 3-Leiter-Anschlusstechnik.
- Während des Betriebes findet eine automatische, interne Kompensation des Leitungswiderstandes der fehlenden Messleitung statt.

Hinweis

Zur Gewährleistung einer optimalen Leitungskompensation bei Cu10 beachten Sie bitte Folgendes:

- Die Summe aus Kabelwiderstand und Messwiderstand darf 31 Ω nicht überschreiten.
- Das Kabel darf maximal einen Widerstand von 8 Ω aufweisen, wenn Sie den Temperaturbereich bis über 312 °C nutzen möchten.
  Beispiel: Ein 200 m langes Cu-Kabel mit 0,5 mm^2 Adernquerschnitt hat etwa 7 Ω. Ein geringerer Querschnitt verkürzt die zulässige Kabellänge entsprechend.

Einsatz von PTC-Widerständen

PTCs eignen sich für die Temperaturüberwachung bzw. als thermische Schutzeinrichtung von komplexen Antrieben oder Transformatorwicklungen.

- Wählen Sie in der Parametrierung "Widerstand 2-Leiter" und "PTC":
- Schließen Sie den PTC in 2-Leiter-Anschlusstechnik an.
- Verwenden Sie PTC-Widerstände vom Typ A (Kaltleiter) nach DIN/VDE 0660, Teil 302.
- Wenn die Diagnose "Über-/Unterlauf" freigegeben ist, wird bei Widerstandswerten < 18 Ω eine Diagnose "unterer Grenzwert unterschritten" erzeugt, die einen Kurzschluss anzeigt.
- Sensordaten zum PTC-Widerstand:

Tabelle 1-242 Einsatz von PTC-Widerständen

Eigenschaft	Technische Daten	Bemerkung
Schaltpunkte	**Verhalten bei steigender Temperatur**	
	< 550 Ω	**Normalbereich:** • SIMATIC S7: Bit 0 = "0", Bit 2 = "0" (im PAE)
	550 Ω bis 1650 Ω	**Vorwarnbereich:** • SIMATIC S7: Bit 0 = "0", Bit 2 = "1" (im PAE)
	> 1650 Ω	**Ansprechbereich:** • SIMATIC S7: Bit 0 = "1", Bit 2 = "0" (im PAE)
	Verhalten bei fallender Temperatur	
	> 750 Ω	**Ansprechbereich:** • SIMATIC S7: Bit 0 = "1", Bit 2 = "0" (im PAE)
	750 Ω bis 540 Ω	**Vorwarnbereich:** • SIMATIC S7: Bit 0 = "0", Bit 2 = "1" (im PAE)
	< 540 Ω	**Normalbereich:** • SIMATIC S7: Bit 0 = "0", Bit 2 = "0" (im PAE)
(TNF-5) °C (TNF+5) °C (TNF+15) °C Messspannung Spannung am PTC	max. 550 Ω min. 1330 Ω min. 4000 Ω max. 7,5V	TNF= Nennansprechtemperatur

- Belegung im Prozessabbild der Eingänge (PAE) bei SIMATIC S7

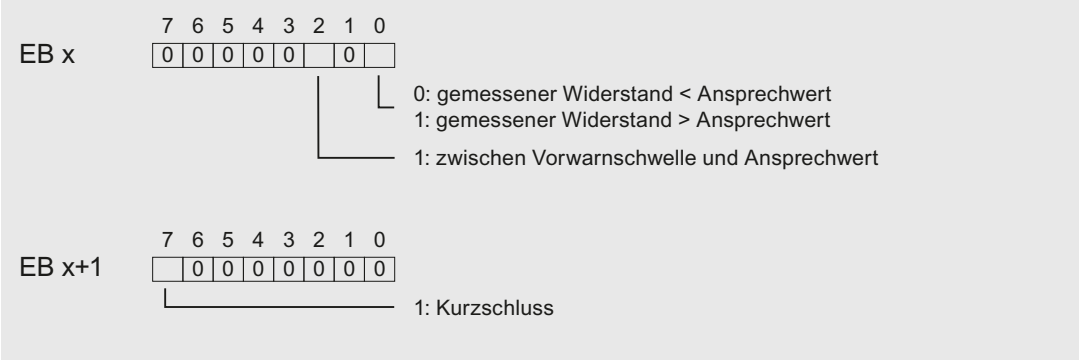

Bild 1-131 Belegung im Prozessabbild der Eingänge (PAE)

- Hinweise zur Programmierung

Hinweis

Im Prozessabbild der Eingänge sind lediglich die Bits 0+2 für die Auswertung relevant. Über die Bits 0+2 können Sie die Temperatur z. B. eines Motors überwachen.

Die Bits 0+2 im Prozessabbild der Eingänge haben kein speicherndes Verhalten. Berücksichtigen Sie bei der Parametrierung, dass z. B. ein Motor kontrolliert (über eine Quittierung) anläuft.

Die Bits 0+2 können niemals gleichzeitig gesetzt sein, sondern werden nacheinander gesetzt.

Werten Sie aus Sicherheitsgründen immer die Diagnoseeinträge des AI 4×RTD/TC 2-/3-/4-wire HF aus, da bei gezogenen Peripheriemodulen, ausgefallener Versorgungsspannung des Peripheriemoduls, Drahtbruch oder Kurzschluss der Messleitungen keine Messung möglich ist.

Beispiel

Das unten stehende Diagramm zeigt den Temperaturverlauf und die dazugehörenden Schaltpunkte.

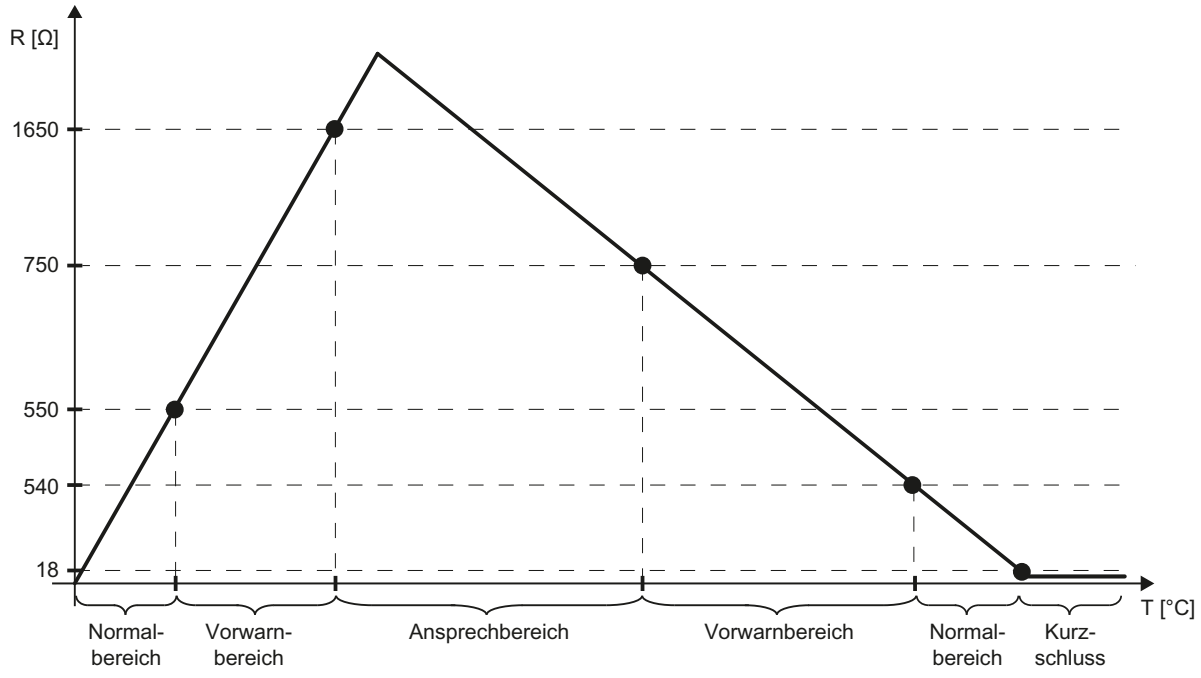

Siehe auch

Parameter der Analogeingabemodule (Seite 3405)

Wissenswertes zum skalierbaren Messbereich (Seite 3418)

Wissenswertes zum skalierbaren Messbereich

Der skalierbare Messbereich

Der skalierbare Messbereich ist ein Ausschnitt aus dem Temperatur-Messbereich eines Analogeingabemoduls (zum Beispiel des ET 200SP-Moduls "AI 8xRTD/TC 2-wire HF").

In diesem Ausschnitt ist eine höhere Auslösung der Messwerte möglich, vergleichbar einer Lupe, die einen Teilbereich genauer betrachtet.

Der skalierbare Messbereich wird für die folgenden Messarten unterstützt:

- Thermowiderstand (RTD) Standard
- Thermoelement

Nicht zur Verfügung steht der skalierbare Messbereich bei den folgenden Messarten:

- Spannung
- Widerstand
- Thermowiderstand Klima

Lage und Auflösung des skalierbaren Messbereichs

Die Lage und die Auflösung des skalierbarern Messbereichs sind parametrierbar (skalierbar):

- **Lage:** Der skalierbare Messbereich lässt sich über den gesamten Standardmessbereich verschieben. Damit können Sie den Temperaturbereich bestimmen, für den Ihre Anwendung eine höhere Auslösung erfordert.
  Ausnahme: Der skalierbare Messbereich kann nicht so weit verschoben werden, dass er in den Über- oder Unterlauf des Standard-Messbereichs hineinreicht (Clipping).
  Die Lage des skalierbaren Messbereichs bestimmen Sie mit dem Parameter "Messbereichsmittelpunkt" (Bild unten).
- **Auflösung:** Es sind die folgenden Werte einstellbar:
  - 2 Nachkommastellen (0,01 °C)
  - 3 Nachkommastellen (0,001 °C)

  Die Auslösung bestimmen Sie mit dem Parameter "Messbereichsauflösung" (Bild unten).

Beispiel für eine Parametrierung

Das folgende Bild zeigt eine Parametrierung für das ET 200SP-Modul "AI 8xRTD/TC 2-,3-,4-wire HF".

In STEP 7 gelangen Sie zu diesen Parametern im Eigensschaftsfeld über Allgemein > AI 4 > Eingänge > Kanal 0 bis Kanal 3.

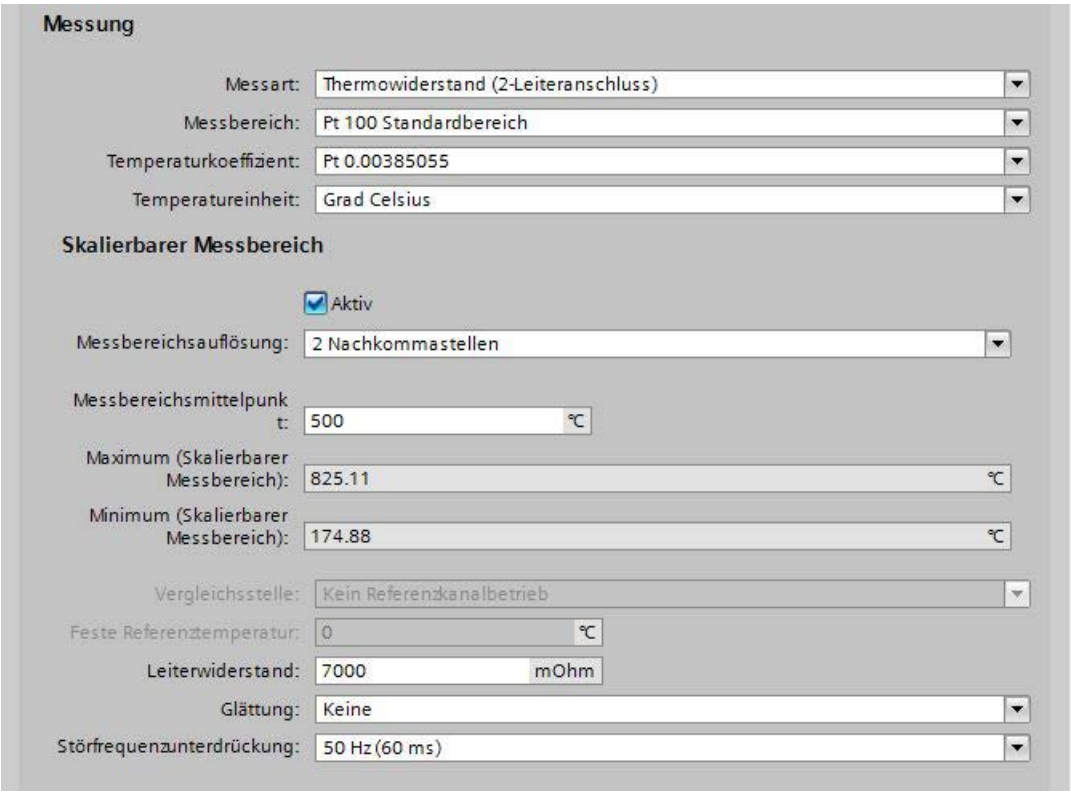

Leiterwiderstand:

Der Parameter "Leiterwiderstand" in der oberen Parametrierung ist nur aktiv, wenn die Messart "Thermowiderstand (2-Leiteranschluss)" gewählt wurde.

Tragen Sie hier den Widerstandswert der Anschlussleitung des Thermowiderstands ein: Eine 200 Meter lange Kupferleitung mit 0,5 mm^2 Adernquerschnitt besitzt zum Beispiel einen Widerstandswert von sieben Ohm.

Messbereichsauflösung:

In der oberen Parametrierung wurde eine Auslösung von 0,01 °C bewählt (Messbereichsauflösung "2 Nachkommastellen").

Messbereichsmittelpunkt:

Der Messbereichsmittelpunkt steht auf 500 °C.

Daraus ergibt sich bei einer Auslösung von 0,01 °C ein skalierbarer Messbereich von 174,88 °C bis 825,11 °C.

Bei einer Auflösung von 0,01 °C umfasst der skalierbare Messbereich 650,23 °C.

Maximum (Skalierbarer Messbereich):

Dieser Wert stellt die obere Grenze des skalierbaren Messbereichs dar. Im Beispiel oben 825,11°C.

Der Wert wird von STEP 7 berechnet (bei einer Auflösung auf 0,001 °C liegt die obere Grenze bei 532.511 °C, siehe Bild unten).

Minimum (Skalierbarer Messbereich):

Dieser Wert stellt die untere Grenze des skalierbaren Messbereichs dar. Im Beispiel oben 174,88 °C.

Der Wert wird von STEP 7 berechnet (bei einer Auflösung auf 0,001 °C liegt die untere Grenze bei 467.488 °C, siehe Bild unten).

Höhere Auflösung:

Das folgende Bild zeigt eine Parametrierung mit einer Auflösung von 0,001 °C (sonst gleiches Beispiel wie im Bild oben):

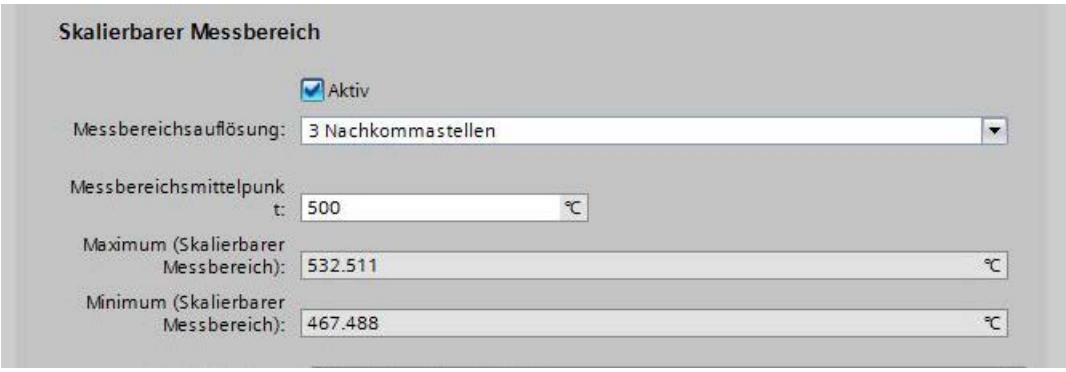

Bei einer Auflösung von 0,001 °C liegt der skalierbare Messbereich zwischen 467,488 und 532,511 °C und umfasst 65,023 °C (ein Zehntel des Messbereichs bei 0,01 °C Auflösung).

Standardmessbereich mit 0,1 °C Auflösung

Die folgende Tabelle zeigt den Standardmessbereich für Thermowiderstände des Typs "Pt 100", Werte in Grad Celsius.

Pt 100 Standard in °C (1 Digit = 0,1 °C)	Dezimalwerte	Hexadezimalwerte	Bereiche
> 1000,0	32767	7FFF	Überlauf
1000,0 : 850,1	10000 : 8501	2710 : 2135	Übersteuerungs-bereich
850,0 : -200,0	8500 : -2000	2134 : F830	Nennbereich
-200,1 : -243,0	-2001 : -2430	F82F : F682	Untersteuerungs-bereich
< -243,0	-32768	8000	Unterlauf

Der Standardmessbereich ist die Grundlage für den skalierbaren Messbereich.

Sie können den Messbereichsmittelpunkt innerhalb des Nennbereichs (-200 °C bis 850 °C, Tabelle oben) einstellen.

Für Temperaturen unterhalb und oberhalb des eingestellten Messbereichsmittelpunkts erhalten Sie dann Messwerte in hoher Auflösung.

Die Breite dieses Bereichs um den Messwertmittelpunkt ist abhängig von der gewählten Auflösung.

Skalierbarer Messbereich mit 0,01 °C und 0,001 °C Auflösung

Der skalierbare Messbereich ist durch die folgenden Wertebereiche gekennzeichnet:.

Skalierbarer Messbereich	Messbereichsauflösung (Werte in °C)		Hexadezimalwerte
	2 Nachkommastellen	3 Nachkommastellen	
Überlauf	> 325,11	> 32,511	7FFF
Obere Grenze	325,11	32,511	7EFF
Messbereichsmittelpunkt	0	0	0
Untere Grenze	-325,11	-32,511	8100
Unterlauf	< -325,11	< -32,511	8000

Maximum und Minimum des skalierbaren Messbereichs sind abhängig von der gewählten Auflösung:

- 2 Nachkommastellen, Auflösung von 0,01 °C:
  Die obere Grenze liegt 325,11 °C oberhalb des von Ihnen eingestellten Messbereichsmittelpunkts.
  Die untere Grenze liegt 325,11 °C unterhalb des von Ihnen eingestellten Messbereichsmittelpunkts.
  Somit beträgt der skalierbare Messbereich um den Messbereichsmittelpunkt 650,22 °C.
- 3 Nachkommastellen, Auflösung von 0,001 °C:
  Die obere Grenze liegt 32,511 °C oberhalb des von Ihnen eingestellten Messbereichsmittelpunkts.
  Die untere Grenze liegt 32,511 °C unterhalb des von Ihnen eingestellten Messbereichsmittelpunkts.
  Somit beträgt der skalierbare Messbereich um den Messbereichsmittelpunkt 65,022 °C.

Berechnung der Temperatur

Den Temperaturwert berechnen Sie, indem Sie zum Messbereichsmittelpunkt den Wert addieren, den Sie vom Modul erhalten.

Beispiel:

- Sie haben den Messbereichsmittelpunkt 500 °C eingestellt (siehe Beispiel im Abschnitt "Beispiel für eine Parametrierung"). Für die Auflösung wählten Sie "2 Nachkommastellen".
- Vom Modul erhalten Sie den hexadezimalen Wert "0100" im S7-Format:

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

- Der hexadezimale Wert "0100" entspricht dem Dezimalwert 256.

- Da Sie eine Auflösung von 0,**01** °C gewählt haben, entspricht die Zahl 256 dem Temperaturwert "2,**56** °C".
- Sie addieren nun 500 °C und 2,56 °C und erhalten den Messwert 502,56 °C.

Skalierbarer Messbereich im Standardmessbereich

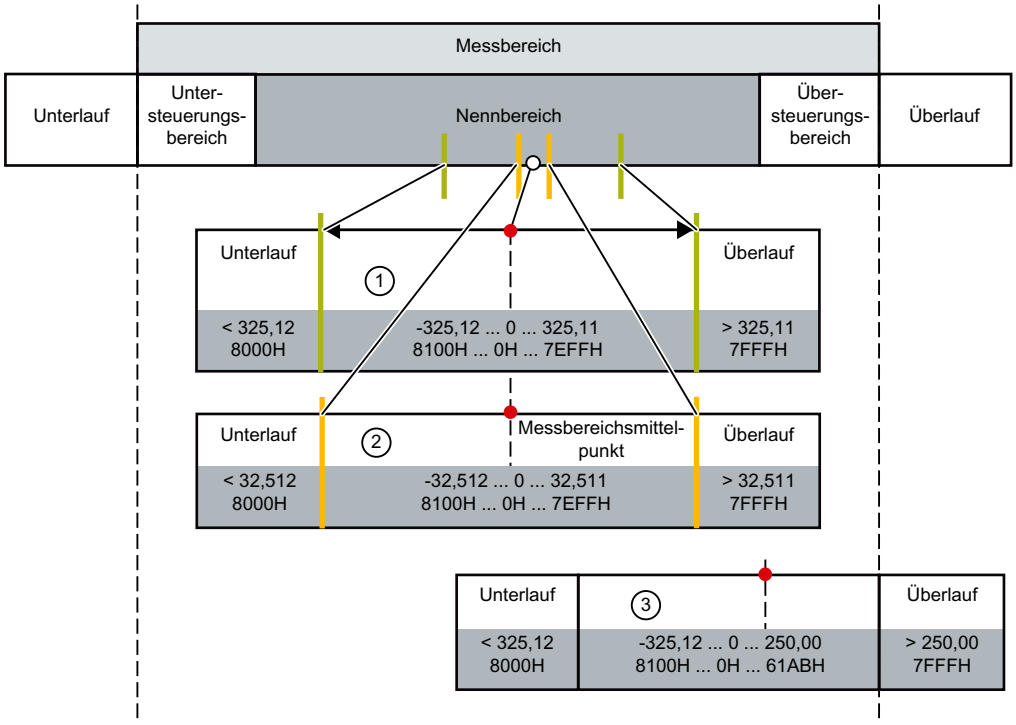

① Skalierbarer Messbereich mit 2 Nachkommastellen, Temperaturwerte im S7-Format.

② Skalierbarer Messbereich mit 3 Nachkommastellen, Temperaturwerte im S7-Format.

③ Skalierbarer Messbereich, der am Überlauf des Standardmessbereichs abgeschnitten ist ("Clipping").
Die Summe aus dem Messbereichsmittelpunkt (zum Beispiel 750 °C) und dem Messwert, den das Modul liefert, darf nicht in den Überlauf des Standardmessbereichs hineinreichen. Deshalb ist im oberen Beispiel der maximale Wert, den das Modul liefern kann, auf 250 °C begrenzt

Clipping

STEP 7 begrenzt das Maximum des skalierbaren Messbereichs, sodass die Summe aus Messwertmittelpunkt und dem maximalen Messwert, den das Modul liefern kann, nicht im Überlauf des Standardmessbereichs liegt. Gleichermaßen begrenzt STEP 7 das Minimum des skalierbaren Messbereichs.

Siehe auch

Besonderheiten bei AI 4xRTD/TC 2-/3-/4-wire HF (Seite 3415)

1.6.3.6 ET 200AL

Dezentrales Peripheriesystem ET 200AL

SIMATIC ET 200AL

Das Dezentrale Peripheriesystem SIMATIC ET 200AL ist ein skalierbares und hochflexibles, dezentrales Peripheriesystem zur Anbindung der Prozess-Signale an eine übergeordnete Steuerung mit einem Feldbus.

Eigenschaften

- Anbindung an PROFINET, PROFIBUS oder Integration in ET 200SP / ET 200SP CPU
- Bis zu 32 Module an einer ET 200AL
- Integration in ET 200SP / ET 200SP CPU: Bis zu 16 AT-Module an einer ET 200SP anschließbar
- Verbindung der Module über ET-Connection
- Räumlich getrennte Montage möglich
- Modulbreiten von 30 und 45 Millimetern
- Schutzart IP65/IP67
- Geeignet für Temperaturen von -25 bis +55 °C und Beschleunigungen bis 5 g.
- Montage in allen Einbaulagen
- Farbliche Kennzeichnung der Leitungen und Anschlüsse
- CA-gerechte Beschriftung der Schnittstellen
- PROFIenergy integriert
- Konfigurationssteuerung
- Anschluss von Sensoren und Aktoren über M8- und M12-Anchlusstechnik

Einsatzgebiet

Das Dezentrale Peripheriesystem SIMATIC ET 200AL ist für den Einsatz bei geringem Platz, bewegte Anwendungen und für die Montage- und Handhabungstechnik besonders geeignet. Durch den skalierbaren Aufbau haben Sie die Möglichkeit, den Ausbau exakt auf den Bedarf vor Ort auszurichten.

Das Dezentrale Peripheriesystem SIMATIC ET 200AL ist in der Schutzart IP65/IP67 ausgeführt und für den dezentralen Einsatz an einer Maschine oder Montagelinie geeignet.

Aufbau

Das Dezentrale Peripheriesystem SIMATIC ET 200AL setzt sich aus folgenden Komponenten zusammen:

- Interfacemodule (PROFINET/PROFIBUS)
- Digitale und analoge Peripheriemodule
- Kommunikationsmodul

Nach einem Interfacemodul können Sie 2 Linien (ET-Connection) mit je 16 Modulen aufbauen.

Alternativ können Sie eine Linie mit 16 Peripheriemodulen am Dezentralen Peripheriesystem SIMATIC ET 200SP mit dem BusAdapter BA-Send 1xFC aufbauen.

Der Rückwandbus ET-Connection ist als Leitung ausgeführt. Dadurch realisieren Sie räumliche Entfernungen von bis zu 10 m zwischen den Modulen.

Beispielkonfiguration

Das folgende Bild zeigt eine Beispielkonfiguration des Dezentralen Peripheriesystems SIMATIC ET 200AL mit einem PROFINET-Interfacemodul.

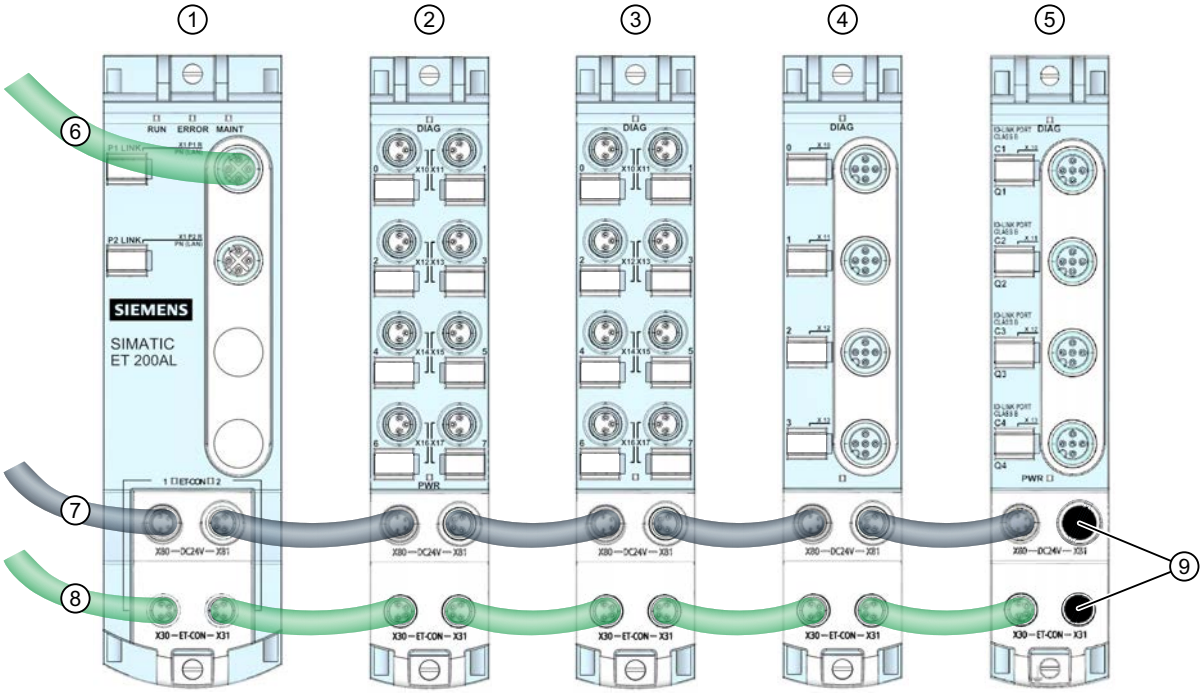

① Interfacemodul (PROFINET)
② Digitalein-/Digitalausgabemodul
③ Digitaleingabemodul
④ Analogeingabemodul
⑤ Kommunikationsmodul
⑥ PROFINET-Leitung
⑦ Stromversorgungsleitung
⑧ ET-Connection Leitung
⑨ Verschlusskappen

Bild 1-132 Beispielkonfiguration des ET 200AL

Siehe auch

ET 200SP mit ET 200AL-Modulen erweitern (Seite 3397)

ET 200AL projektieren

Einleitung

Die ET 200AL ist ein dezentrales Peripheriesystem in der Schutzart IP65/67. Es ist somit für den Einsatz vor Ort ausgelegt, zum Beispiel direkt an einer Maschine (kein Schaltschrank erforderlich).

Das System umfasst Interface- und Kommunikationsmodule sowie Eingabe- und Ausgabebaugruppen.

Die ET 200AL bietet zwei Einsatzfälle:

1. Als IO-Device oder DP-Slave: Das Interfacemodul des Systems ist an einem Feldbus (PROFINET oder PROFIBUS) angeschlossen und mit der PN- oder DP-Schnittstelle einer CPU verbunden.
2. Als Erweiterung einer ET 200SP: Die ET 200AL-Module sind über das Modul "BA Send 1xFC" an die ET 200SP angeschlossen ("Mischbetrieb").

Im Folgendem ist beschrieben wie Sie eine ET 200AL als IO-Device oder DP-Slave projektieren (Einsatzfall 1).

Für Einsatzfall 2 siehe Link unter "Siehe auch".

Vorgehen

Um eine ET 200AL in STEP 7 zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie ein Interfacemodul (PROFINET oder PROFIBUS) der Baureihe ET 200AL in der Netzsicht ein (per Drag & Drop aus dem Hardware-Katalog).
2. Wechseln Sie in die Gerätesicht. Dazu klicken Sie zweimal auf das gerade eingefügte Interfacemodul.
   In der Gerätesicht sind das Interfacemodul und zwei ET-Connection-Racks dargestellt (folgendes Bild).
   Es sind noch keine Steckplatznummern zugeordnet, deshalb werden über den Steckplätzen Fragezeichen angezeigt.

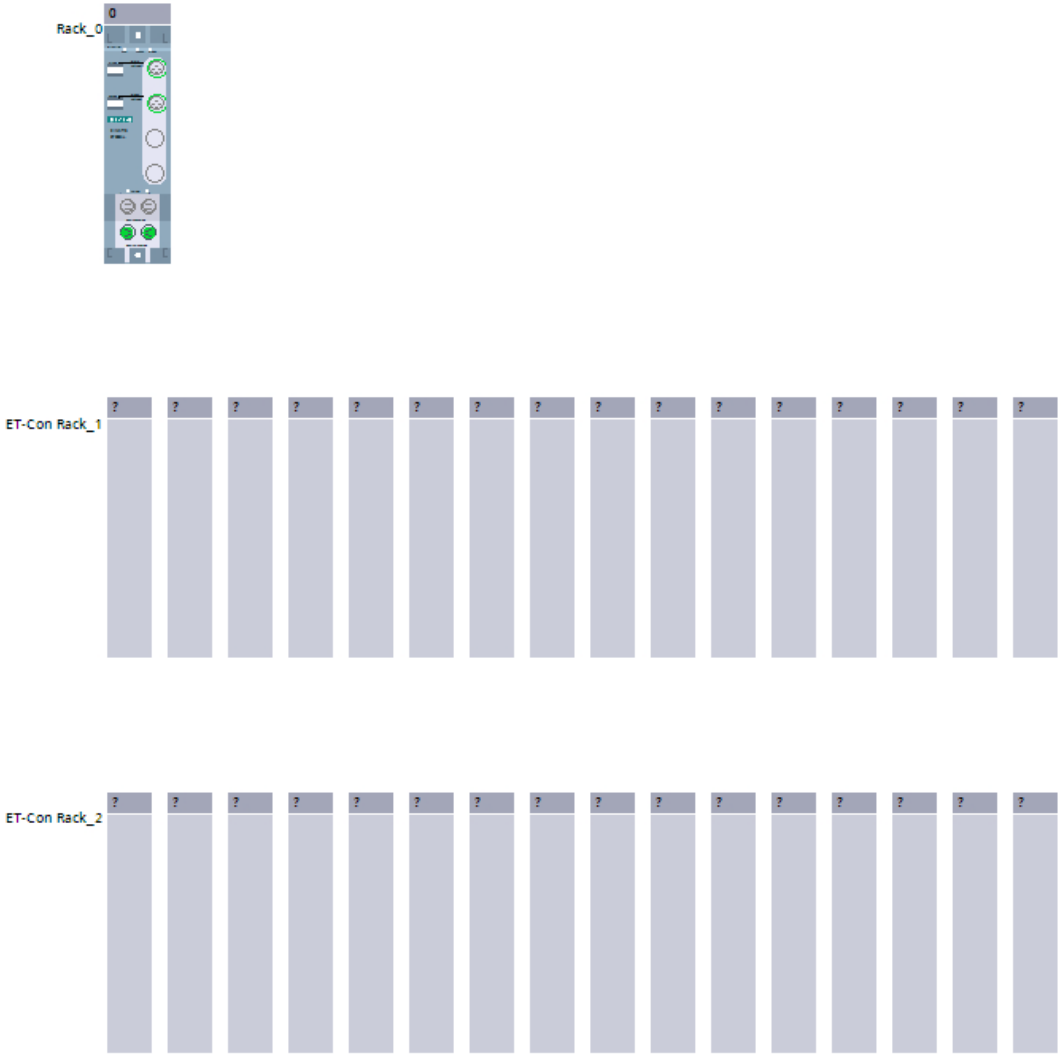

3. Wählen Sie nun aus dem Hardware-Katalog (Ordner ET 200AL) die Module (Eingabe-, Ausgabe- und Kommunikationsmodule) aus und ziehen Sie die Module auf die freien Steckplätze (blau umrandet, im Bild nicht dargestellt).
   In jedem ET-Connection-Rack können Sie bis zu 16 ET 200AL-Module platzieren (folgendes Bild).
   Ein ET-Connnection-Rack ist ein virtuelles Rack, aus dem die Reihenfolge der angeschlossenen ET 200AL-Module hervorgeht.

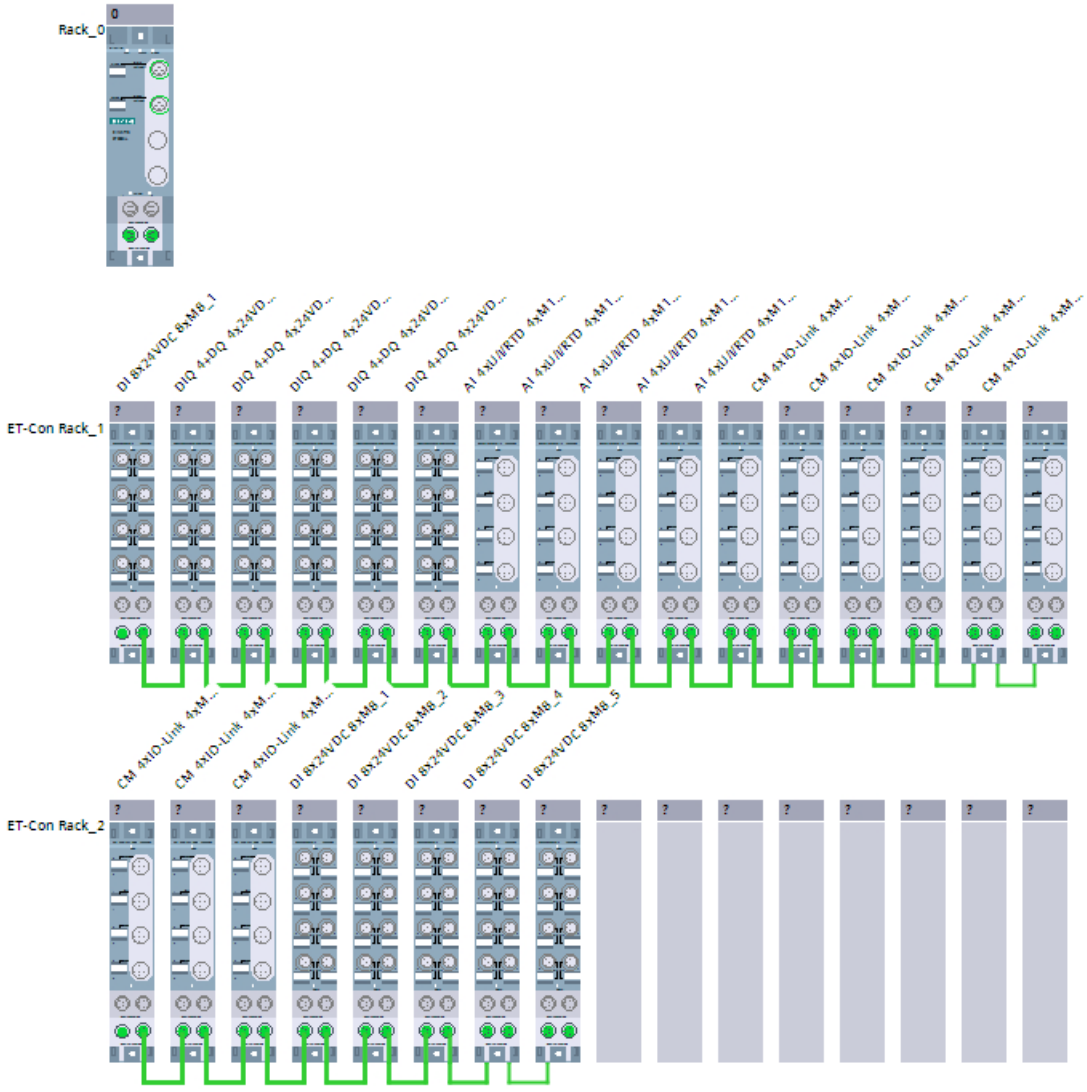

4. Verbinden Sie nun das Interfacemodul mit den beiden ET-Connection-Racks.
   Dazu klicken Sie zunächst auf eine ET-Connection-Schnittstelle des interfacemoduls und ziehen bei gedrückter linker Maustaste eine Linie zum linken ET-Anschluss des ersten Moduls eines der beiden ET-Connection-Racks.
   Wiederholen Sie den Schritt für die zweite ET-Connection-Schnittstelle des interfacemoduls und das zweite ET-Connection-Rack (falls verwendet).

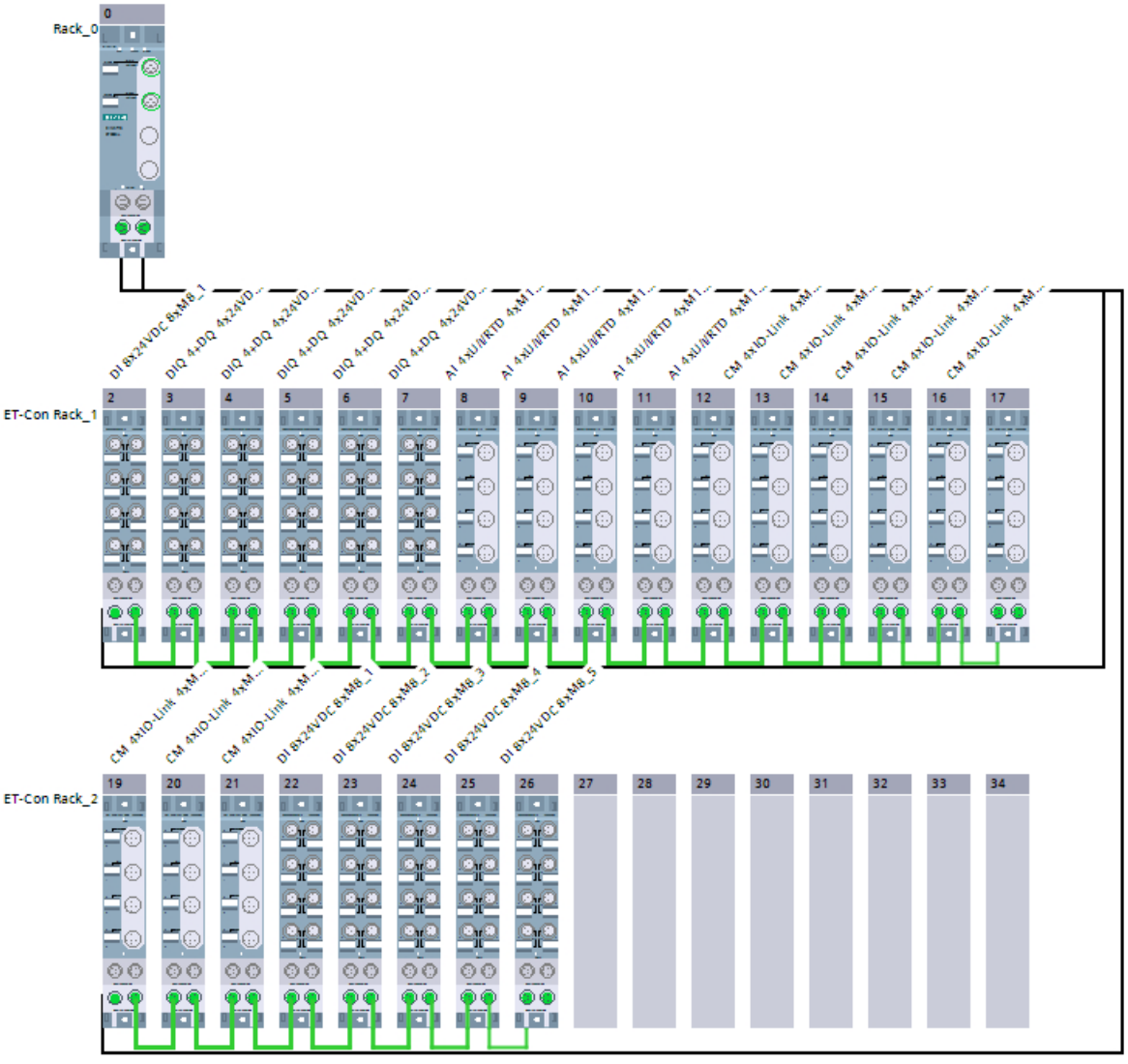

5. Mit einem Doppelklick auf ein Modul gelangen Sie zu den Eigenschaften des Moduls, um die Parameter des Moduls einzustellen.

Regeln

- Die ET 200AL-Module müssen lückenlos projektiert sein.
- Das erste Modul in einem ET-Connection-Rack muss mit dem Interfacemodul verbunden sein.

Siehe auch

Dezentrales Peripheriesystem ET 200AL (Seite 3424)

Konfigurationssteuerung bei ET 200AL

Funktionsprinzip

Durch die Konfigurationssteuerung ist es möglich, die ursprüngliche Konfiguration einer ET 200AL (erfolgte durch die Projektierung mit STEP 7) im Nachhinein durch ein Anwenderprogramm zu ändern und die ET 200AL in dieser geänderten Konfiguration zu betreiben. Für diese Konfiguration ist STEP 7 nicht mehr erforderlich: Sie teilen der ET 200AL über Ihr Anwenderprogramm mit, auf welchem Steckplatz ein projektiertes Modul tatsächlich gesteckt ist.

Dazu verwenden Sie den Steuerdatensatz 196. In diesem Datensatz kodieren Sie, welche Module abweichend von der Projektierung mit STEP 7 im realen Aufbau fehlen oder welche Module sich abweichend von der Projektierung auf einem anderen Steckplatz befinden. Auf die Parametrierung der Module (zum Beispiel das Freigeben von Diagnosemeldungen) hat die Konfigurationssteuerung keinen Einfluss.

Anschließend rufen Sie die Anweisung "WRREC" auf und schreiben damit den Datensatz an das Interfacemodul der ET 200AL.

Durch diese Konfigurationssteuerung können Sie flexibel den Aufbau einer ET 200AL variieren, solange sich die reale Konfiguration aus der projektierten Konfiguration eines Maximalausbaus (ursprünglich erstellt durch STEP 7) ableiten lässt.

Im Folgenden ist beschrieben, wie Sie die Konfigurationssteuerung aktivieren und wie der erforderliche Datensatz 196 für die ET 200AL aufgebaut sein muss.

Voraussetzung

Der CPU-Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf Anlauf auch bei Unterschieden (Voreinstellung). Diese Einstellung ist auch für den Anlaufparameter der

Konfigurationsste

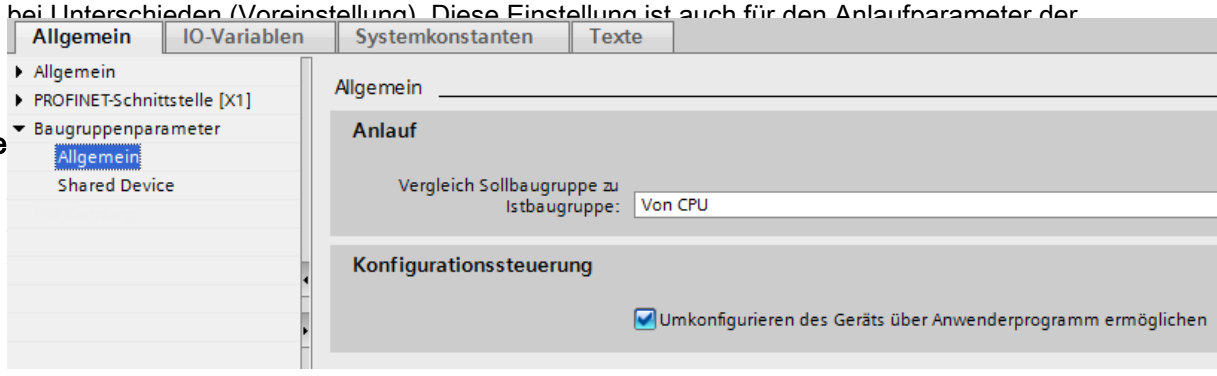

Aufbau von Steuerdatensatz 196

Der Aufbau des Datenbausteins entspricht der ursprünglichen Projektierung der ET 200AL mit STEP 7.

Für jedes Modul sind zwei Bytes im Datensatz vorgesehen. Die Position dieser beiden Bytes im Datensatz kodiert ein Modul bei der ursprünglichen Projektierung mit STEP 7.

- Byte 4 und 5 im Datensatz entsprechen dem Modul auf Steckplatz 1 in der ursprünglichen Projektierung.
- Byte 6 und 7 im Datensatz entsprechen dem Modul auf Steckplatz 2 in der ursprünglichen Projektierung.
- Byte 8 und 9 im Datensatz entsprechen dem Modul auf Steckplatz 3 in der ursprünglichen Projektierung.
- usw.

Der aktuelle (reale) Steckplatz ist durch die Zahl kodiert, die dem Byte "slot_x" zugewiesen wird (durch seinen Wert): Beispiele:

- Der Wert "2" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, auch in der aktuellen Konfiguration den Steckplatz 2 zu.
- Der Wert "3" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Steckplatz 3 zu.
- Der Wert "4" im Byte 6 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Steckplatz 4 zu.
- usw.

Steuerdatensatz 196 erstellen

Das folgende Bild zeigt einen Ausschnitt aus dem Steuerdatensatz 196 für eine Projektierung einer ET 200AL.

Diese Projektierung soll als Beispiel dienen:

- ET-Con1 steckt auf "slot_1" (fest vorgegeben). Die beiden ET-Connection-Submodule "ET-Con1" und "ET-Con2" sind Submodule des Interface-Moduls der ET 200AL. Diese Module sind fest im IM-Modul integriert. Sie können nicht einzeln gesteckt werden.
- ET-Con2 steckt auf "slot_18" (fest vorgegeben).
- An ET-Con1 sind in dieser Projektierung 16 AL-Module angeschlossen (im folgenden Datensatz "slot_2" bis "slot_17"). Dies ist der maximale Ausbau.
- An ET-Con2 ist in dieser Projektierung ein AL-Modul angeschlossen ("slot_19"). An ET-Con2 könnten aber (wie an ET-Con1) insgesamt 16 AL-Elektronikmodule angeschlossen sein.

Die ursprünglich mit STEP 7 projektierte ET 200AL soll nun aus dem Anwenderprogramm heraus umkonfiguriert werden.

Die neue Konfiguration hat die folgenden Eigenschaften:

- ET-Con1 steckt auf "slot_1" (fest vorgegeben).
- Modul 2 wird auch in der geänderten Konfiguration auf Steckplatz 2 betrieben.

- Modul 3 wird nicht genützt.
- Modul 4 steckt jetzt auf Steckplatz 3.
- Modul 5 steckt jetzt auf Steckplatz 4.
- Alle anderen Module an ET-Con1 nicht verwendet.
- ET-Con2 steckt auf "slot_18" (fest vorgegeben)
- Das Modul an ET-Con2 wird verwendet.

Record_196_ET_200AL

	Name	Data type	Default value
1	block_length	USInt	42
2	block_ID	USInt	196
3	version	USInt	2
4	subversion	USInt	1
5	slot_1_ET_Con1	USInt	1
6	reserve_1_ET_Con1	USInt	0
7	slot_2	USInt	2
8	reserve_2	USInt	0
9	slot_3	USInt	0
10	reserve_3	USInt	0
11	slot_4	USInt	3
12	reserve_4	USInt	0
13	slot_5	USInt	4
14	reserve_5	USInt	0
15	slot_6	USInt	0
16	reserve_6	USInt	0
17	slot_7	USInt	0
18	reserve_7	USInt	0
19	slot_8	USInt	0
20	reserve_8	USInt	0
21	slot_9	USInt	0
22	reserve_9	USInt	0
23	slot_10	USInt	0
24	reserve_10	USInt	0
25	slot_11	USInt	0
26	reserve_11	USInt	0
27	slot_12	USInt	0
28	reserve_12	USInt	0
29	slot_13	USInt	0
30	reserve_13	USInt	0
31	slot_14	USInt	0
32	reserve_14	USInt	0
33	slot_15	USInt	0
34	reserve_15	USInt	0
35	slot_16	USInt	0
36	reserve_16	USInt	0
37	slot_17	USInt	0
38	reserve_17	USInt	0
39	slot_18 ET_Con2	USInt	1
40	reserve_18_ET_Con2	USInt	0
41	slot_19	USInt	1
42	reserve_19	USInt	0

Die Komponenten des Steuerdatensatzes 196 (Definition im Abschnitt "Steuerdatensatz 196" unten):

- block_length: Notieren Sie hier die Länge des Steuerdatensatzes, im Beispiel 42 (Byte). Die Länge des Steuerdatenblocks errechnet sich nach der Formel: 2 x "Anzahl der Module" + 4.
- block_ID: Tragen Sie hier die Zahl 196 ein. Durch diese Zahl wird der Datensatz als Datensatz für die Konfigurationssteuerung identifiziert.
- version: Die ET200AL verwendet die Version 2 des Steuerdatensatzes 196.
- subversion: Die ET200AL verwendet die Subversion 1 des Steuerdatensatzes 196.
- slot_1: Auf Steckplatz 1 befindet sich bei der ET 200AL immer das Submodul ET-Connection 1.
- reserve_1: Dieses Byte wird nicht genützt (Wert "0").
- slot_2: Das projektierte Modul 2 steckt im Steckplatz 2 (Wert "2").
- reserve_2: Dieses Byte wird nicht genützt (Wert "0").
- slot_3: Das projektierte Modul 3 fehlt in der aktuellen Konfiguration (Wert "0").
- reserve_3: Dieses Byte wird nicht genützt (Wert "0").
- slot_4: Das projektierte Modul 4 steckt bei der aktuellen Konfiguration im Steckplatz 3 (Wert "3").
- reserve_4: Dieses Byte wird nicht genützt (Wert "0").
- slot_5: Das projektierte Modul 5 steckt bei der aktuellen Konfiguration im Steckplatz 4 (Wert "4").
- reserve_5: Dieses Byte wird nicht genützt (Wert "0").
- slot_6: Das projektierte Modul 6 fehlt in der aktuellen Konfiguration (Wert "0").
- reserve_6: Dieses Byte wird nicht genützt (Wert "0").
- slot_7: Das projektierte Modul 7 fehlt in der aktuellen Konfiguration (Wert "0").
- reserve_7: Dieses Byte wird nicht genützt (Wert "0").
- usw.
- slot_18: Auf Steckplatz 18 befindet sich bei der ET 200AL immer das Submodul ET-Connection 2 (Wert "18").
- reserve_18: Dieses Byte wird nicht genützt (Wert "0").
- slot_19: Das projektierte Modul 19 steckt bei der aktuellen Konfiguration im Steckplatz 19 (Wert "19").
- reserve_19: Dieses Byte wird nicht genützt. (Wert "0")

Steuerdatensatz 196 für ET 200AL

Für die Konfigurationssteuerung wird ein Steuerdatensatz 196 definiert, der eine Steckplatzzuordnung enthält.

Byte	Element	Wert	Erläuterungen	
0	Blocklänge in Byte	z.B. 14 für ET 200AL mit 5 Modulen	Die Länge des Datensatzes errechnet sich nach der Formel: 4 + (Anzahl der Module x 2) Byte	Header
1	Block-ID	196	ID für den Steuerdatensatz 196	
2	Version	2	Version 2 des Steuerdatensatzes 196	
3	Subversion	1	Subversion 1 des Steuerdatensatzes 196	
4	Projektiertes Modul 1 (ET-Connection 1)	1	ET-Connection 1 ist dem Steckplatz 1 fest zugeordnet. Deshalb muss im Byte 4 immer der Wert "1" eingetragen sein.	Zuordnung für das projektierte Modul 1 (ET-Connection 1) zum realen Steckplatz 1
5	Reserve für projektiertes Modul 1	0	Nicht verwendet	
6	Projektiertes Modul 2	Realer Steckplatz des Moduls 2 Mögliche Werte: 2 bis Anzahl der Module (außer 18) 0 (falls das projektierte Modul 2 fehlt)	Das projektierte Modul 2 kann real in einem der Steckplätze 2 bis 34 gesteckt sein. Steckplatz 18 ist für ET-Con 2 reserviert. Wird das projektierte Modul nicht verwendet, so steht in diesem Byte der Wert "0"	Zuordnung für das projektierte Modul 2 zu einem realen Steckplatz
7	Reserve für projektiertes Modul 2	0	Nicht verwendet	
8	Projektiertes Modul 3	Realer Steckplatz des Moduls 3 Mögliche Werte: 2 bis Anzahl der Module (außer 18) 0 (falls das projektierte Modul 2 fehlt)	Das projektierte Modul 3 kann real in einem der Steckplätze 2 bis 34 gesteckt sein. Steckplatz 18 ist für ET-Connection 2 reserviert. Wird das projektierte Modul nicht verwendet, so steht in diesem Byte der Wert "0".	Zuordnung für das projektierte Modul 3 zu einem realen Steckplatz
9	Reserve für projektiertes Modul 3	0	Nicht verwendet	
:	:	:	:	:
39	Projektiertes Modul 18 (ET-Connection 2)	18	ET-Connection 2 nimmt immer den Steckplatze 18 ein, wenn an diesem Sub-Modul AT-Module angeschlossen sind.	Zuordnung für das projektierte Modul 18 (ET-Connection 2) zum realen Steckplatz 18
40	Reserve für projektiertes Modul 18	0	Nicht verwendet	

:	:	:	:	:
(Byte 4 bis 70, nicht Byte 39)	Projektiertes Modul x	Realer Steckplatz des Moduls x Mögliche Werte: 2 bis Anzahl der Module (außer 18) 0 (falls das projektierte Modul x fehlt)	Das projektierte Modul x kann real in einem der Steckplätze 2 bis 34 gesteckt sein. Steckplatz 18 ist für ET-Con 2 reserviert (Bytes 39 und 40 im Steuerdatensatz)	Zuordnung für das projektierte Modul x zu einem realen Steckplatz y
(Byte 5 bis 71, nicht Byte 40)	Reserve für projektiertes Modul x	0	Nicht verwendet	

Regeln

- Die beiden Submodule ET-Connection 1 und ET-Connection 2 sind bei der Konfigurationssteuerung wie reale Module zu behandeln. Einschränkung: Platzierung von ET-Connetion 1 immer fest auf Steckplatz 1 und ET-Connection 2 immer fest auf Steckplatz 18.
- Bei der ET 200AL gibt es keine Reservemodule (wie bei der ET 200S oder BU-Cover-Module bei der ET 200SP), deshalb darf das Bit 7 von "slot_x" nicht gesetzt sein (d. h., es dürfen nur die Werte 0 bis 127 verwendet werden).
- Der Wert "0" von "slot_x" drückt aus, dass dieses Modul in der aktuellen Konfiguration nicht gesteckt ist.
- Bei der Projektierung mit STEP 7 dürfen keine Lücken zwischen den AL-Modulen gelassen werden.
- Werden bei der Projektierung mit STEP 7 keine Module an ET-Con2 angeschlossen, so ist ET-Con2 nicht projektiert: Dadurch verkürzt sich der Datensatz 196.
- Werden bei der Projektierung mit STEP 7 weniger als 16 Module an ET-Con1 angeschlossen und sind zudem auch an ET-Con2 Module angeschlossen, dann müssen die nicht-belegten Steckplätze an ET Con 1 im Steuerdatensatz 196 enthalten sein. Ihnen wird als Wert für den realen Steckplatz die Null zugewiesen.

Datensatz schreiben

Übertragen Sie den Steuerdatensatz zur Baugruppe ET 200AL.

Rufen Sie dazu die erweiterte Anweisung WRREC (Datensatz schreiben) auf und übertragen Sie den erstellten Steuerdatensatz.

Übertragen Sie keinen Steuerdatensatz, so verwendet das Interface-Modul die ursprüngliche Projektierung mit STEP 7. Hier gilt: Projektiertes Modul x ist im realen Steckplatz x eingesteckt.

Adressierung des Interfacemoduls über HW-Kennung

Um den Datensatz 196 mit der Anweisung WRREC zu übertragen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z. B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Fehlermeldungen

Beim Schreiben des Steuerdatensatzes 196 werden Ihnen im Fehlerfall folgende Fehlermeldungen zurückgegeben:

Tabelle 1-243 Fehlermeldungen

Fehlercode	Bedeutung
16#80A2	DP-Protokollfehler auf Layer 2. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.
16#80B1	Unzulässige Länge; Die Längenangabe im Datensatz 196 ist nicht korrekt.
16#80B5	Konfigurationssteuerung nicht parametriert.
16#80B2	invalid slot: Der projektierte Steckplatz ist nicht belegt.
16#80B8	Parameterfehler; Modul meldet ungültige Parameter.
16#80C5	DP-Slave oder Modul nicht verfügbar. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.

Rückmeldedatensatz 197 für ET 200AL

Der Rückmeldedatensatz 197 dient zum Lesen des tatsächlichen Aufbaus einer Station (hier einer ET 200AL).

Durch diesen Datensatz kann der reale Ausbau der ET 200AL (Istkonfiguration) überprüft werden. Der Rückmeldedatensatz enthält für jedes projektierte Modul die Angabe, ob es tatsächlich vorhanden ist oder nicht.

- Der Wert "1" bedeutet, dass das richtige Modul auf dem vorgesehenen Steckplatz steckt.
- Der Wert "0" kodiert alle anderen Möglichkeiten (falsches Modul, leerer Steckplatz, BU-Cover).

Beispiel:

Ein Modul wurde mit STEP 7 auf Steckplatz 4 projektiert.

Dann wurde dieses Modul mithilfe des Datensatzes 196 in der aktuellen Konfiguration auf Steckplatz 3 verschoben.

Wenn dieses Modul auch tatsächlich auf Steckplatz 3 steckt, dann wird dies durch den Wert "1" kodiert (status_slot_4 = 1).

Der Aufbau im Einzelnen:

Der Aufbau des Datenbausteins entspricht der ursprünglichen Projektierung der ET 200AL mit STEP 7.

Für jedes Modul sind zwei Bytes im Datensatz vorgesehen. Die Position dieser beiden Bytes im Datensatz entspricht der Position eines Moduls bei der ursprünglichen Projektierung mit STEP 7.

Reihenfolge der Bytes:

- "status_slot_1_ET_Con1" und "reserve_slot_1_ET-Con1" (Byte 4 und 5 im Datensatz) entsprechen dem Modul auf Steckplatz 1 der Projektierung,
- "status_slot_2" und "reserve_slot_2" (Byte 6 und 7) entsprechen dem Modul auf Steckplatz 2 der Projektierung
- "status_slot_3" und "reserve_slot_3" (Byte 8 und 9) entsprechen dem Modul auf Steckplatz 3 der Projektierung,
- usw.

Beispiel

Den folgenden Rückmeldedatensatz 197 sendet die ET 200AL zurück, die mit dem Steuerdatensatz 196 aus dem Beispiel oben umkonfiguriert wurde (Abschnitt "Steuerdatensatz 196 erstellen").

Record_197_ET_200AL

	Name	Data type	Default value
1	block_length	USInt	42
2	block_ID	USInt	197
3	version	USInt	2
4	subversion	USInt	1
5	status_slot_1_ET_Con1	USInt	1
6	reserve_1_ET_Con1	USInt	0
7	status_slot_2	USInt	1
8	reserve_2	USInt	0
9	status_slot_3	USInt	0
10	reserve_3	USInt	0
11	status_slot_4	USInt	1
12	reserve_4	USInt	0
13	status_slot_5	USInt	1
14	reserve_5	USInt	0
15	status_slot_6	USInt	0
16	reserve_6	USInt	0
17	status_slot_7	USInt	0
18	reserve_7	USInt	0
19	status_slot_8	USInt	0
20	reserve_8	USInt	0
21	status_slot_9	USInt	0
22	reserve_9	USInt	0
23	status_slot_10	USInt	0
24	reserve_10	USInt	0
25	status_slot_11	USInt	0
26	reserve_11	USInt	0
27	status_slot_12	USInt	0
28	reserve_12	USInt	0
29	status_slot_13	USInt	0
30	reserve_13	USInt	0
31	status_slot_14	USInt	0
32	reserve_14	USInt	0
33	status_slot_15	USInt	0
34	reserve_15	USInt	0
35	status_slot_16	USInt	0
36	reserve_16	USInt	0
37	status_slot_17	USInt	0
38	reserve_17	USInt	0
39	status_slot_18 ET_Con2	USInt	1
40	reserve_18_ET_Con2	USInt	0
41	status_slot_19	USInt	1
42	reserve_19	USInt	0

An ET-Con1 sind die Module 2, 4 und 5 tatsächlich angeschlossen.

Alle anderen Module, die in der Projektierung mit STEP 7 an ET-Con1 angeschlossen waren, fehlen im aktuellen Aufbau (entsprechend der Vorgabe des Steuerdatensatzes 196 aus dem Beispiel von oben).

An ET-Con2 ist tatsächlich ein Modul angeschlossen, wie auch in der ursprünglichen Projektierung mit STEP 7.

Rückmeldedatensatz 197 lesen

Mit der Anweisung RDREC können Sie den Rückmeldedatensatz 197 von der ET 200AL lesen. RDREC arbeitet asynchron. Wenn Sie RDREC im Anlauf-OB aufrufen, dann müssen Sie die Anweisung über eine Schleife mehrmals aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz gelesen ist.

Um den Datensatz 197 mit der Anweisung RDREC zu lesen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Weiterführende Informationen und Beispiele

Weitere Informationen zur ET 200AL finden Sie hier. (http://support.automation.siemens.com/WW/view/de/89254863)

Konkrete Beispiele für die Konfigurationssteuerung finden Sie hier in dieser Applikationsbeschreibung (http://support.automation.siemens.com/WW/view/de/29430270).

Eine Bibliothek der Datensätze und weitere Anwendungsbeispiele sind hier abgelegt: Vorlagen für die Datensätze (http://support.automation.siemens.com/WW/view/de/29430270)

Siehe auch

Dezentrales Peripheriesystem ET 200AL (Seite 3424)

ET 200SP mit ET 200AL-Modulen erweitern (Seite 3397)

Konfigurationssteuerung bei ET 200SP (Seite 3384)

1.6.3.7 ET 200MP

Dezentrales Peripheriesystem ET 200MP

Definition

Das Dezentrale Peripheriesystem ET 200MP ist ein skalierbares und flexibles, dezentrales Peripheriesystem zur Anbindung der Prozesssignale an eine Zentralsteuerung über einen Feldbus.

Einsatzgebiet

Das ET 200MP ist ein multifunktionales dezentrales Peripheriesystem für unterschiedliche Einsatzbereiche. Durch den skalierbaren Aufbau können Sie den Ausbau exakt auf den jeweiligen Bedarf vor Ort ausrichten.

Das ET 200MP ist für die Schutzart IP 20 zugelassen und für den Einbau in einem Schaltschrank vorgesehen.

Aufbau

Das ET 200MP wird auf eine Profilschiene montiert und setzt sich zusammen aus:

- Einem Interfacemodul, das mit allen IO-Controllern kommuniziert, die sich nach der PROFINET-Norm IEC 61158 verhalten,
- Bis zu 30 Module (Stromversorgungsmodule und Peripheriemodule aus dem Peripheriespektrum der S7-1500) rechts neben dem Interfacemodul steckbar.
- Wenn Sie ein Stromversorgungsmodul links vom Interfacemodul einsetzen, ergibt sich ein möglicher Maximalausbau von insgesamt 32 Modulen.
- die Anzahl der steckbarer Peripheriemodule durch deren Strombedarf eingeschränkt sein.

Steckplatzregeln

- Steckplatz 0: Stromversorgungsmodul (opitonal)
- Steckplatz 1: Interfacemodul
- Steckplatz 2 bis 31: Peripheriemodule bzw. Stromversorgungsmodule

Parameter Interfacemodul

Versorgungsspannung L+ angeschlossen

Parameter "Versorgungsspannung L+ angeschlossen"

Dieser Parameter hat Einfluss auf die Diagnose und auf die Prüfung der Leistungsbilanz.

- Diagnose des ET 200MP:
  Wenn die Istkonfiguration nicht der Sollkonfiguration hinsichtlich der Versorgungsspannung des Interfacemoduls entspricht, erzeugt das Interfacemodul eine Diagnosemeldung.
  Beispiel: Sie haben die Option "Versorgungsspannung L+ angeschlossen" deaktiviert, im tatsächlichen Aufbau aber die Versorgungsspannung angeschlossen.
- Prüfung der Leistungsbilanz bei der Projektierung:
  Entsprechend der Einstellung des Parameters verändert sich die Leistungsbilanz:
  Entweder speist das Interfacemodul Leistung in den Rückwandbus ein oder es entnimmt Leistung aus dem Rückwandbus.

Die Voreinstellung (Option "Versorgungsspannung L+ angeschlossen" ist **aktiviert**) bedeutet, dass das Interfacemodul frontseitig mit DC 24 V versorgt wird und Leistung in den Rückwandbus einspeist.

Wenn die Option "Versorgungsspannung L+ angeschlossen" **deaktiviert** ist, dann darf das Interfacemodul nicht frontseitig mit DC 24 V versorgt werden.

In diesem Fall muss links neben dem Interfacemodul eine Systemstromversorgung (PS) stecken, das das Interfacemodul und die Module rechts neben dem Interfacemodul versorgt.

Hinweis

Wir empfehlen, das Interfacemodul immer frontseitig mit DC 24 V zu versorgen. Wenn zusätzlich eine Systemstromversorgung (PS) **vor** bzw. links neben dem Interfacemodul gesteckt und angeschlossen wird, dann steht dem Aufbau sowohl die Leistung der Systemstromversorgung (PS) als auch die Leistung der integrierten Stromversorgung des Interfacemoduls zur Verfügung.

Sie brauchen dann die Voreinstellung des Parameters nicht verändern.

Konfigurationssteuerung (Optionenhandling) bei ET 200MP

Funktionsprinzip

Durch die Konfigurationssteuerung betreiben Sie mit einer einzigen Projektierung des dezentralen Peripheriegeräts ET 200MP unterschiedliche reale Konfigurationen (Optionen).

Durch die Konfigurationssteuerung haben Sie die Möglichkeit, das dezentrales Peripheriegerät ET 200MP mit seinem Vollausbau zu projektieren und trotzdem mit fehlenden Modulen zu betreiben. Wenn fehlende Module später nachgerüstet werden, ist keine neue Projektierung erforderlich und damit auch kein erneutes Laden der Hardware-Konfiguration.

Mit dem Steuerdatensatz 196, der im Anwenderprogramm an das Interfacemodul übermittelt wird, legen Sie eine aktuelle Konfiguration fest. Den Steuerdatensatz übertragen Sie mit der Anweisung WRREC.

Mit dem Rücklesedatensatz 197 lesen Sie den tatsächlichen Aufbau einer ET 200MP.

Voraussetzungen

Der CPU-Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf Anlauf auch bei Unterschieden (Voreinstellung). Diese Einstellung ist auch für den Anlaufparameter der einzelnen Module der ET 200MP gewählt.

Konfigurationssteuerung aktivieren

Aktivieren Sie in den Eigenschaften des Interfacemoduls unter Baugruppenparameter > Allgemein > Konfigurationssteuerung die Option "Umkonfigurieren des Geräts über Anwenderprogramm ermöglichen". Dadurch ist die Konfigurationssteuerung eingeschaltet.

Steuerdatensatz 196 für ET 200MP

Das folgende Bild zeigt den Anfang des Steuerdatensatzes 196 für die Konfigurationssteuerung einer ET 200MP.

Der Datenbaustein ist 36 Byte lang (Maximalausbau mit 32 Modulen). Deshalb steht im Element "Block_length" des Datensatzes der Wert "36".

Wenn Sie in STEP 7 eine ET 200MP mit weniger Modulen projektieren, dann verkürzt sich der Datenbaustein: Bei nur fünf Modulen zum Beispiel verkürzt sich der Datensatz auf 9 Byte (4 Byte für den Header plus jeweils ein Byte für jedes Modul).

Im Datensatz ist für jedes Modul ein Byte vorgesehen. Die Position dieses Bytes im Datensatz kodiert ein Modul bei der ursprünglichen Projektierung mit STEP 7:

- "slot_0 power supply" (Byte 4 im Datensatz unten) entspricht dem Stromversorgungsmodul auf Steckplatz 0 in der Projektierung mit STEP 7.
- "slot_2" (Byte 5 im Datensatz) entspricht dem Modul auf dem Steckplatz 2 in der Projektierung.
- "slot_3" (Byte 6 im Datensatz) entspricht dem Modul auf dem Steckplatz 3 in der Projektierung.
- "slot_4" (Byte 7) entspricht dem Modul auf dem Steckplatz 4 in der Projektierung.
- usw.

Hinweis

Interfacemodul auf Steckplatz 1

Die Konfigurationssteuerung wird über das Interfacemodul (Steckplatz 1/Submodul 1) gesteuert. Das Interfacemodul auf Steckplatz 1 ist daher kein Element der Konfigurationssteuerung und wird nicht im Steuerdatensatz aufgeführt.

Wert in slot_x

Der aktuelle Steckplatz ist durch die Zahl kodiert, die "slot_x" zugewiesen wird (durch seinen Wert). Beispiele:

- Der Wert "2" in slot_2 bedeutet, Sie weisen dem Modul, das ursprünglich auf Platz 2 steckte, auch in der aktuellen Konfiguration den Platz 2 zu (slot_2 = 2).
- Der Wert "3" in slot_2 bedeutet, Sie weisen dem Modul, das ursprünglich auf Platz 2 steckte, in der aktuellen Konfiguration den Platz 3 zu (slot_2 = 3).
- Der Wert "4" in slot_2 bedeutet, Sie weisen dem Modul, das ursprünglich auf Platz 2 steckte, in der aktuellen Konfiguration den Platz 4 zu (slot_2 = 4).
- usw.

Beispiel für Datensatz 196

Der folgende Datensatz wurde für eine Konfiguration erstellt, die die ursprüngliche Projektierung mit STEP 7 ändert.

Die geänderte Konfiguration weist die folgenden Eigenschaften auf:

- Das Modul, das in der Projektierung auf Steckplatz 0 steckte (Stromversorgungsmodul), steckt in der aktuellen Konfiguration ebenfalls auf Steckplatz 0 (Vorgabe).
- Das Modul, das in der Projektierung auf Steckplatz 2 steckte (Modul 2), steckt in der aktuellen Konfiguration ebenfalls auf Steckplatz 2.
- Das Modul, das in der Projektierung auf Steckplatz 3 steckte (Modul 3), ist in der aktuellen Konfiguration nicht vorhanden.

- Das Modul, das in der Projektierung auf Steckplatz 4 steckte (Modul 4), steckt in der aktuellen Konfiguration auf Steckplatz 3.
- Das Modul, das in der Projektierung auf Steckplatz 5 steckte (Modul 5), steckt in der aktuellen Konfiguration auf Steckplatz 4.
- usw.

Im Bild unten sind die Bytes "slot_6" bis "slot_31" nicht dargestellt.

Record_196_ET_200MP

	Name	Datentyp	Defaultwert
1	Block_length	USInt	36
2	Block_ID	USInt	196
3	version	USInt	3
4	subversion	USInt	0
5	slot_0 power supply	USInt	0
6	slot_1 interfae module	USInt	1
7	slot_2	USInt	2
8	slot_3	USInt	127
9	slot_4	USInt	3
10	slot_5	USInt	4

Regeln

- Wenn ein Modul in der aktuellen Konfiguration nicht vorhanden ist, dann wird dies durch den Wert 127 angezeigt: "slot_x" = 127.
- Das Stromversorgungsmodul ist immer auf Steckplatz 0 ("slot_0 power supply" = 0).
- Das Interfacemodul auf Steckplatz 1 wird im Steuerdatensatz nicht berücksichtigt.

Adressierung des Interfacemoduls über HW-Kennung

Um den Datensatz 196 mit der Anweisung WRREC zu übertragen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z. B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Rücklesedatensatz 197 für ET 200MP

Der Rücklesedatensatz 197 dient zum Lesen des tatsächlichen Aufbaus einer Station (hier einer ET 200MP).

Durch diesen Datensatz kann der reale Ausbau der ET 200MP (Istkonfiguration) überprüft werden. Der Rücklesedatensatz enthält für jedes projektierte Modul die Angabe, ob es tatsächlich vorhanden ist oder nicht.

- Der Wert "1" bedeutet, dass das richtige Modul auf dem vorgesehenen Steckplatz steckt.
- Der Wert "0" kodiert alle anderen Möglichkeiten (falsches Modul, leerer Steckplatz, Reservemodul).

Beispiel:

Ein Modul wurde mit STEP 7 auf Steckplatz 4 projektiert.

Dann wurde dieses Modul mithilfe des Datensatzes 196 in der aktuellen Konfiguration auf Steckplatz 3 verschoben.

Wenn dieses Modul auch tatsächlich auf Steckplatz 3 steckt, dann wird dies durch den Wert "1" kodiert (status_slot_4 = 1).

Der Aufbau im Einzelnen:

Der Aufbau des Datenbausteins entspricht der ursprünglichen Projektierung der ET 200MP mit STEP 7.

Für jedes Modul ist ein Byte im Datensatz vorgesehen. Die Position dieses Bytes im Datensatz entspricht der Position eines Moduls bei der ursprünglichen Projektierung mit STEP 7.

Reihenfolge der Bytes:

- "status_slot_0 power supply" (Byte 4 im Datensatz unten) entspricht dem Stromversorgungsmodul auf Steckplatz 0 in der Projektierung mit STEP 7.
- "status_slot_2" (Byte 5) entspricht dem Modul auf Steckplatz 2 der Projektierung.
- "status_slot_3" (Byte 6) entspricht dem Modul auf Steckplatz 3 der Projektierung.
- usw.

Die Namen der Komponenten (zum Beispiel "status_slot_2") können Sie frei wählen.

Die Bedeutung von "status_slot_x":

- Der Wert "1" in status_slot_x bedeutet, dass das Modul x auf dem vorgesehenen Steckplatz steckt.
- Der Wert "0" in status_slot_x kodiert alle anderen Möglichkeiten (falsches Modul, Modul nicht vorhanden).

Beispiel:

Das folgende Bild zeigt den Rücklesedatensatz 197 für einen Aufbau einer ET 200MP, bei dem Modul 3 fehlt (das Modul auf Steckplatz 3 in der Projektierung).

Alle anderen Module sind vorhanden und richtig gesteckt.

Im Bild unten sind die Bytes "status_slot_6" bis "status_slot_31" nicht dargestellt.

Record_197_ET_200MP

	Name	Datentyp	Defaultwert
1	Block_length	USInt	36
2	Block_ID	USInt	197
3	version	USInt	3
4	subversion	USInt	0
5	status_slot_0 power supply	USInt	1
6	status_slot_2	USInt	1
7	status_slot_3	USInt	0
8	status_slot_4	USInt	1
9	status_slot_5	USInt	1

Rücklesedatensatz 197 lesen

Mit der Anweisung RDREC können Sie den Rücklesedatensatz 197 von der ET 200MP lesen. RDREC arbeitet asynchron. Wenn Sie RDREC im Anlauf-OB aufrufen, dann müssen Sie die Anweisung über eine Schleife mehrmals aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz gelesen ist.

Um den Datensatz 197 mit der Anweisung RDREC zu lesen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Weiterführende Informationen und Beispiele

Weitere Informationen zur ET 200MP finden Sie hier im Gerätehandbuch zur IM 155-5 PN (http://support.automation.siemens.com/WW/view/de/89261636).

Konkrete Beispiele für die Konfigurationssteuerung finden Sie in dieser Applikationsbeschreibung (http://support.automation.siemens.com/WW/view/de/29430270).

Eine Bibliothek der Datensätze und weitere Anwendungsbeispiele sind hier abgelegt: Vorlagen für die Datensätze (http://support.automation.siemens.com/WW/view/de/29430270)

Siehe auch

Dokumentation zur Konfigurationssteuerung (http://support.automation.siemens.com/WW/view/de/67295970)

Parameter Eingabemodule

Parameter der Analogeingabemodule

Fehlende Versorgungsspannung L+

Freigabe der Diagnose bei fehlender oder zu geringer Versorgungsspannung L+.

Drahtbruch

Freigabe der Diagnose, wenn das Modul am entsprechend parametrierten Eingang keinen Stromfluss bzw. zu geringen Strom für die Messung hat oder eine zu niedrige Spannung anliegt.

Stromgrenze für Diagnose Drahtbruch

Schwellwert bei dem Drahtbruch gemeldet wird. Je nach verwendetem Sensor kann der Wert auf 1,185 mA oder 3,6 mA eingestellt werden.

Überlauf

Freigabe der Diagnose, wenn der Messwert den Übersteuerungsbereich überschreitet.

Unterlauf

Freigabe der Diagnose, wenn der Messwert den Untersteuerungsbereich unterschreitet.

Gleichtaktfehler

Freigabe der Diagnose, wenn die zulässige Common-Mode-Spannung überschritten wird.

Referenzkanalfehler (nur bei AI 8xU/I/RTD/TC ST)

- Freigabe der Diagnose, wenn ein Fehler am Kanal für die Temperaturkompensation vorliegt, z. B. Drahtbruch.
- Kompensationsart Dynamische Referenztemperatur ist parametriert und an das Modul wurde noch keine Referenztemperatur übertragen.

Temperaturkoeffizient

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials. In Europa wird pro Sensorart nur ein Wert verwendet (voreingestellter Wert).

Der Korrekturfaktor für den Temperaturkoeffizienten (α-Wert) gibt an, um wie viel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Die weiteren Werte ermöglichen eine sensorspezifische Einstellung des Temperaturkoeffizienten und somit eine noch höhere Genauigkeit.

Störfrequenzunterdrückung

Unterdrückt bei Analogeingabemodulen die Störungen, die durch die Frequenz des verwendeten Wechselspannungsnetzes hervorgerufen werden.

Die Frequenz des Wechselspannungsnetzes kann sich besonders bei der Messung in kleinen Spannungsbereichen und bei Thermoelementen störend auf den Messwert auswirken. Mit diesem Parameter geben Sie die Netzfrequenz an, die in Ihrer Anlage vorherrscht.

Glättung

Die einzelnen Messwerte werden mittels Filterung geglättet. Die Glättung ist bei den Analogeingabemodulen AI 8xU/I/RTD/TC ST und AI 8xU/I HS in 4 Stufen einstellbar.

Glättungszeit = Anzahl der Modulzyklen (k) x Zykluszeit des Moduls.

Das folgende Bild zeigt, nach wie vielen Modulzyklen der geglättete Analogwert zu annähernd 100 % anliegt in Abhängigkeit der eingestellten Glättung. Gilt für jeden Signalwechsel am Analogeingang.

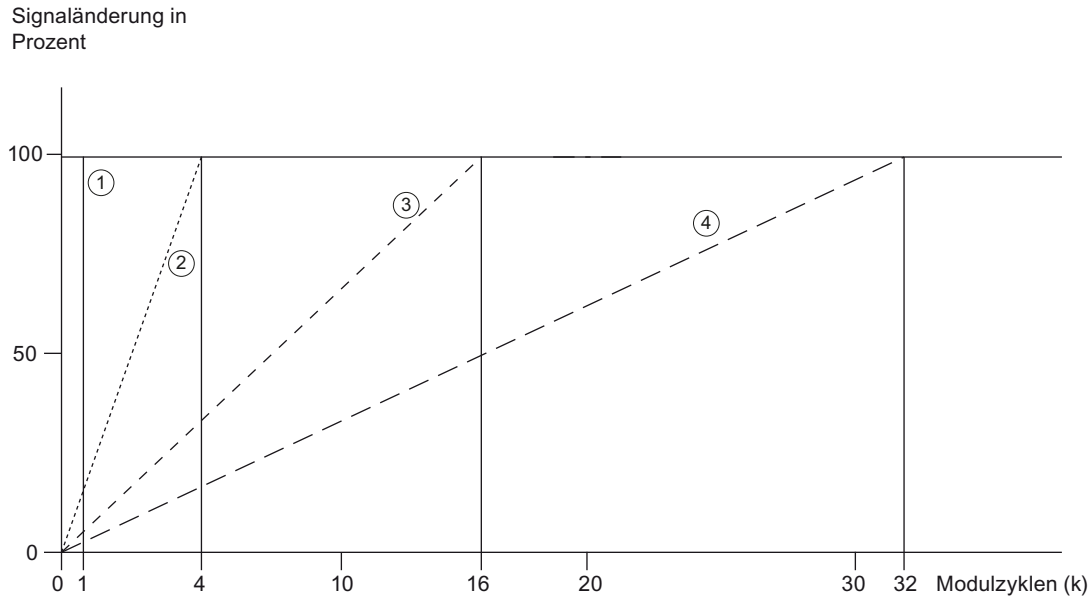

① Keine (k = 1)
② Schwach (k = 4)
③ Mittel (k = 16)
④ Stark (k = 32)

Vergleichsstelle (nur bei AI 8xU/I/RTD/TC ST)

Für den Parameter Vergleichsstelle können folgende Einstellungen parametriert werden:

Tabelle 1-244 Mögliche Parametrierungen für den Parameter Vergleichsstelle

Einstellung	Beschreibung
Feste Referenztemperatur	Die Temperatur der Vergleichsstelle wird parametriert und als fester Wert im Modul hinterlegt.
Dynamische Referenztemperatur	Die Temperatur der Vergleichsstelle wird im Anwenderprogramm mit der Anweisung WRREC (SFB 53) über die Datensätze 192 bis 199 von der CPU an das Modul übertragen.
Interne Vergleichsstelle	Die Temperatur der Vergleichsstelle wird mit einem Sensor ermittelt, der im Modul integriert ist.
Referenzkanal des Moduls	Die Temperatur der Vergleichsstelle wird mit einem externen Widerstandsthermometer (RTD) am Referenzkanal (COMP) des Moduls ermittelt.

Hinweis

Feste Referenztemperatur

Bei Parametrierung eines Thermoelements Typ B ist nur die Einstellung "Feste Referenztemperatur" mit einer Temperatur 0 °C möglich.

Freigabe Prozessalarm 1 bzw. 2

Freigabe eines Prozessalarms, wenn die obere Grenze 1 bzw. 2 überschritten oder die untere Grenze 1 bzw. 2 unterschritten wird.

Untere Grenze 1 bzw. 2

Legen Sie die Schwelle fest, bei deren Unterschreitung der Prozessalarm 1 bzw. 2 ausgelöst wird.

Obere Grenze 1 bzw. 2

Legen Sie eine Schwelle fest, bei deren Überschreitung der Prozessalarm 1 bzw. 2 ausgelöst wird.

Temperaturkompensation bei Thermoelementen

Einleitung

Sie haben verschiedene Möglichkeiten, die Vergleichsstellentemperatur zu erfassen, um aus der Temperaturdifferenz zwischen Vergleichsstelle und Messstelle einen absoluten Temperaturwert zu erhalten.

Je nachdem wo (örtlich) Sie die Vergleichsstelle benötigen, können Sie die verschiedenen Kompensationsmöglichkeiten nutzen.

Hinweis

Bei Parametrierung eines Thermoelements Typ B ist nur die Einstellung "Feste Referenztemperatur" mit einer Temperatur 0 °C möglich.

Möglichkeiten zur Kompensation der Vergleichsstellentemperatur

Kompensatiosmöglichkeiten	Erläuterung	Anwendungsfall
Interne Vergleichsstelle	Bei dieser Kompensation wird die Temperatur der Vergleichsstelle mit einem Sensor ermittelt, der im Modul integriert ist. **Vorgehensweise** Schließen Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an das Peripheriemodul an.	• Sie verwenden für den Anschluss Ausgleichsleitungen, die dem Material des Thermoelements entsprechen. • Wenn in Ihrer Anlage die Temperatur der Vergleichsstelle und die Temperatur des Moduls identisch sind, dann dürfen Sie auch Leitungen aus unterschiedlichem Material verwenden.
Referenzkanal des Moduls	Die Temperatur der Vergleichsstelle wird mit einem externen Widerstandsthermometer (RTD) ermittelt. **Vorgehensweise** Verbinden Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an der Vergleichsstelle mit den Zuleitungen. Die Zuleitungen schließen Sie an die entsprechenden Klemmen des Moduls an. Schließen Sie das Widerstandsthermometer (RTD) am Referenzkanal des Moduls an. Das Widerstandsthermometer (RTD) muss im Bereich der Vergleichsstelle platziert werden.	• Sie möchten die Temperatur direkt an der Vergleichsstelle erfassen. • Die gemessenen Temperaturen aller Kanäle, die Sie für diese Kompensationsart projektiert haben, werden um den Temperaturwert der Vergleichsstelle automatisch korrigiert. • Von der Vergleichsstelle bis zum Modul können Sie preiswerte z. B. Kupferleitungen verwenden.
Dynamische Referenztemperatur	Die Temperatur der Vergleichsstelle wird über ein Modul ermittelt. Diesen Temperaturwert übergeben Sie im Anwenderprogramm über einen Datensatz an weitere Module. **Vorgehensweise** Schließen Sie das Widerstandsthermometer (RTD) für die Vergleichsstelle an einem beliebigen Kanal an. Die Temperatur der Vergleichsstelle wird mit der Anweisung WRREC durch den Datensatz 192 bis 199 von der CPU an das Modul übermittelt.	• Sie setzen mehrere Module an der Vergleichsstelle ein und können deshalb alle Kanäle über einen gemeinsamen Temperaturwert kompensieren. • Für das Erfassen des Temperaturwerts benötigen Sie nur ein Widerstandsthermometer (RTD). • Von der Vergleichsstelle bis zum Modul können Sie preiswerte Leitungen, z. B. Kupferleitungen verwenden.
Feste Referenztemperatur	Die Temperatur der Vergleichsstelle wird als fester Wert im parametriert. **Vorgehensweise** Verbinden Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an der Vergleichsstelle mit den Zuleitungen. Die Zuleitungen schließen Sie an die entsprechenden Klemmen des Moduls an. Geben Sie bei der Projektierung des Moduls einen festen Temperaturwert für die Vergleichsstelle vor (z. B. 20°C).	• Sie halten die Temperatur der Vergleichsstelle auf einer konstanten Temperatur und kennen den Temperaturwert. • Von der Vergleichsstelle bis zum Modul können Sie preiswerte Leitungen, z. B. Kupferleitungen verwenden.

Parameter Ausgabemodule

Parameter der Analogausgabemodule

Fehlende Versorgungsspannung L+

Freigabe der Diagnose, bei fehlender oder zu geringer Versorgungsspannung L+.

Kurzschluss nach M

Freigabe der Diagnose, wenn ein Kurzschluss der Aktorversorgung nach M auftritt.

Drahtbruch

Freigabe der Diagnose, wenn die Leitung zum Geber unterbrochen ist.

Überlauf

Freigabe der Diagnose, wenn der Messwert den Überlaufbereich überschreitet.

Unterlauf

Freigabe der Diagnose, wenn der Messwert den Unterlaufbereich unterschreitet.

Verhalten bei CPU-STOP

Bestimmt das Verhalten des Ausgangs, wenn die CPU in den Betriebszustand STOP geht.

Ersatzwert

Die Ersatzwerte sind Werte, die die Ausgänge (der Ausgang) im Fall eines CPU-STOPs ausgeben.

Pulsweitenmodulation (PWM)

Die Kanäle einiger Module unterstützen die Funktion Pulsweitenmodulation (PWM). Mit der Funktion Pulsweitenmodulation können Sie auf einfache Weise periodische Impulse mit konstanter Nennspannung und variabler Impulsdauer für die o. g. Kanäle erzeugen.

Projektierung

Sie projektieren die Pulsweitenmodulation mit den folgenden Parametern:

- Funktion PWM auf Modul aktivieren
- Betriebsart PWM für entsprechenden Kanal einstellen
- Periodendauer einstellen

Funktionsweise

In der Betriebsart Pulsweitenmodulation stellen die entsprechend konfigurierten Ausgänge ein pulsweitenmoduliertes Ausgangssignal zur Verfügung.

Die Pulsweitenmodulation ist charakterisiert durch ihre Periodendauer (Frequenz) und ihr Tastverhältnis. Das Tastverhältnis kennzeichnet das Verhältnis von Impulsdauer zu Periodendauer.

Die Impulsdauer ergibt sich aus der Periodendauer und dem Tastverhältnis: Impulsdauer = Tastverhältnis x Periodendauer.

Beispiel für Tastverhältnis von 50 % und Periodendauer von 10 ms:

Impulsdauer: 0,5 x 10 ms = 5 ms

Das Tastverhältnis der Kanäle definieren Sie im Anwenderprogramm über den Ausgabewert (0 ... 1000) im Prozessabbild der Ausgänge.

Das Ausgangssignal ist ein Rechtecksignal (Impulsfolge von Ein- und Ausschaltimpulsen).

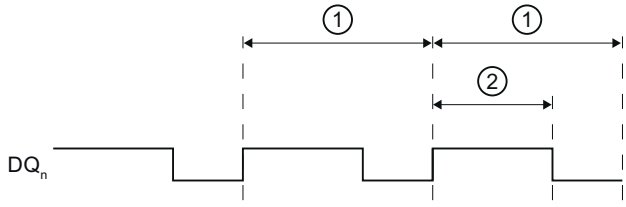

① Periodendauer T (2 bis 100 ms); Frequenz der Pulsweitenmodulation: f = 1/T (10 bis 500 Hz)

② Impulsdauer (Tastverhältnis x Periodendauer)

Bild 1-133 Funktionsweise der Pulsweitenmodulation

Mindestimpulsdauer

Die Mindestimpulsdauer beträgt hardwarebedingt 300 µs. Das Tastverhältnis ist einstellbar zwischen 0,0 bis 100,0 %. Die Periodendauer ist parametrierbar von 2 bis 100 ms.

Beispiel: Wenn Sie eine Periodendauer von 2 ms parametriert haben und für den Ausgang mit einem Tastverhältnis von 0,1 % setzen, dann würde sich rechnerisch eine Impulsdauer von 200 µs ergeben. Tatsächlich arbeitet der Ausgang mit der Mindestimpulsdauer von 300 µs.

Impulsverlauf

Die Impulsdauer des tatsächlichen Signalverlaufs, ist geringfügig länger als die vorgegebene, ideale Impulsdauer.

Das folgende Bild zeigt das Verhalten des Ausgangs bei Ansteuerung durch PWM. Die blaue Linie zeigt den vorgegebenen, idealen Signalverlauf (Rechtecksignal), mit der der Ausgang angesteuert wird. Die rot gestrichelte Linie zeigt den tatsächlichen Signalverlauf an der Ausgangsklemme, bedingt durch die extern angeschlossenen Last.

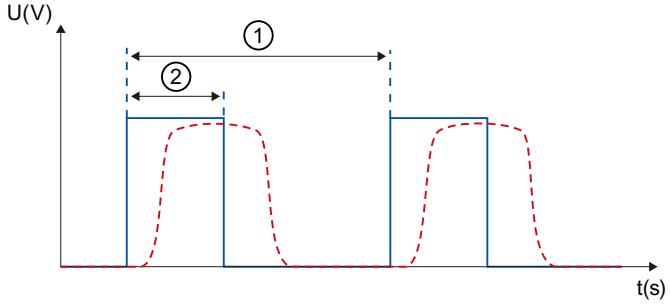

① Periodendauer

② Impulsdauer (Tastverhältnis x Periodendauer)

Bild 1-134 Impulsverlauf an der Ausgangsklemme

1.6.3.8 ET 200M

ET 200M konfigurieren

Einführung

Für die Familie ET 200M finden Sie im Hardware-Katalog unter "Dezentrale Peripherie" ein breites Modulspektrum.

Konfigurieren und Parametrieren

Informationen zum Konfigurieren und Parametrieren finden Sie in den folgenden Kapiteln.

Aufbau ET 200M

Definition

Das Dezentrale Peripheriegerät ET 200M ist ein modulares Peripheriegerät in der Schutzart IP 20.

ET 200M hat die Aufbautechnik des Automatisierungssystems S7-300 und besteht aus IM 153-x und Peripheriebaugruppen der S7-300.

ET 200M kann kommunizieren mit:

- allen DP-Mastern, die sich nach Norm IEC 61784-1:2002 Ed1 CP 3/1 verhalten
- allen IO-Controllern, die sich nach der Norm IEC 61158 verhalten

Aufbau von ET 200M (Beispiel)

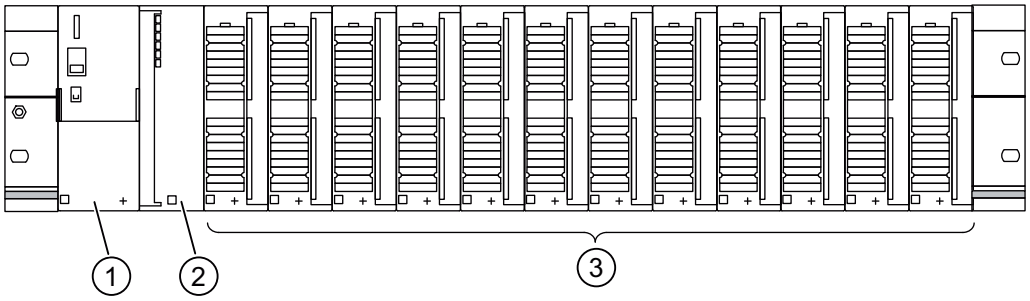

① Stromversorgungsbaugruppe PS 307

② Interfacemodul IM 153-x

③ bis zu 12 Peripheriebaugruppen (SM / FM / CP)

Projektieren der Funktion 'Baugruppenwechsel im Betrieb'

Einleitung

Die ET 200M unterstützt die Funktion "Baugruppenwechsel im Betrieb" und den damit verbundenen Ziehen/Stecken-Alarm.

Die Funktion "Baugruppenwechsel im Betrieb" ermöglicht es Ihnen, Module während des Betriebes aus dem Baugruppenträger der ET 200M zu ziehen oder zu stecken.

Voraussetzung

Sie haben ein Interfacemodul projektiert, dass den Baugruppenwechsel im Betrieb unterstützt. (ab IM 153-1, Artikel-Nr. 153-1AA02-0XB0).

Außerdem muss die projektierte CPU die Funktion ebenfalls unterstützten, z. B. für PROFIBUS eine S7-400 mit DP-Schnittstelle.

Für den Hardware-Aufbau müssen Sie den aktiven Rückwandbus (Busschiene mit Steckplätzen) verwenden. Die herkömmliche Profilschiene mit Busverbindern zwischen den Baugruppen unterstützt diese Funktion nicht.

Projektierung

Wenn die Projektierungsvoraussetzungen erfüllt sind, wird im Inspektorfenster, Bereich "Baugruppenparameter" der Parameter "Baugruppenwechsel im Betrieb" angeboten, den Sie aktivieren können. Für die projektierten Baugruppen wird unterhalb dieses Parameters eine Tabelle angezeigt, die benötigte aktive Busmodule für den Hardware-Aufbau anzeigt.

Für eine PROFIBUS-Konfiguration wird außerdem die Option "Anlauf bei Sollausbau ungleich Istausbau" angezeigt. Diese Option wird automatisch aktiviert, wenn "Baugruppenwechsel im Betrieb" aktiviert ist.

Signalbaugruppen für Prozessautomatisierung

Grundlagen

Einführung

Signalbaugruppen für die Prozessautomatisierung sind Baugruppen der S7-300 wie z. B. die SM 321; DI 16xNAMUR oder SM 322; DO 16xDC24V/0,5A.

Sie werden in einem DP-Slave (IM 153-2) betrieben.

Die bieten gegenüber Standardbaugruppen folgende zusätzliche technologische Funktionen wie z. B. Impulsverlängerung und Flatterüberwachung.

Siehe auch

Wechsler (Seite 3455)

Technologische Parameter (Seite 3456)

Wechsler

Gebertyp "Wechsler"

Wenn die Digitaleingänge einer Kanalgruppe als "Wechsler" parametriert sind, führt die Baugruppe für diese Kanalgruppe eine Diagnose auf den Gebertyp Wechsler durch.

Wechsler

Ein Wechsler ist ein Hilfsschalter mit nur einem beweglichen Schaltstück, der je eine Schließstellung bei geschlossenem und geöffnetem Schaltgerät hat.

Beachten Sie dabei folgende Vorschrift:

- Schließerkontakt immer am "geraden" Kanal anschließen,
- Öffnerkontakt immer am "ungeraden" Kanal anschließen.

Die tolerierte Umschaltzeit zwischen den beiden Kanälen ist fest 300 ms lang.

Ist die Überprüfung negativ, dann

- kennzeichnet die Baugruppe den Wertstatus des Schließerkanals als "ungültig"
- erzeugt die Baugruppe einen Diagnoseeintrag für den Schließerkanal
- löst einen Diagnosealarm aus (sofern Diagnosalarm freigegeben wurde)

Nur für den Schließerkanal werden das Digitaleingangssignal und der Wertstatus aktualisiert. Für den Öffnerkanal ist das Digitaleingangssignal fest auf "Null" und der Wertstatus "ungültig", da dieser Kanal nur zur Überprüfung des Sensors dient.

Die Diagnose ist abhängig vom Parameter "Auswahl" (des Gebers). Beachten Sie auch die Besonderheiten bei der Diagnose mit dem Gebertyp Wechsler im Handbuch "Signalbaugruppen für die Prozessautomatisierung".

Siehe auch

Dokumentation zu Baugruppen für die Prozessautomatisierung (http://support.automation.siemens.com/WW/view/de/7215812)

Technologische Parameter

Impulsverlängerung und Flatterüberwachung

Die Impulsverlängerung ist eine Funktion zur Veränderung eines digitalen Eingangssignals. Ein Impuls an einem Digitaleingang wird mindestens auf die parametrierte Länge verlängert. Ist der Eingangsimpuls bereits länger als die parametrierte Länge, dann wird er nicht verändert.

Falls die Impulse verlängert werden sollen, klicken Sie in das Feld, um die Zeit auszuwählen. Wenn die Impulse nicht verlängert werden sollen, wählen Sie den Eintrag "---".

Die Flatterüberwachung ist eine leittechnische Funktion für digitale Eingangssignale. Sie erkennt und meldet prozesstechnisch ungewöhnliche Signalverläufe, z. B. ein zu häufiges Schwanken des Eingangssignals zwischen "0" und "1".

Voraussetzung für das Aktivieren der Flatterüberwachung ist, dass die Sammeldiagnose für diesen Eingang ebenfalls freigegeben ist.

Überwachungsfenster und Anzahl Signalwechsel

Die Funktionsweise der Flatterüberwachung wird mit Hilfe der beiden Parameter Überwachungsfenster und Anzahl Signalwechsel festgelegt.

Mit dem ersten Signalwechsel des Eingangssignals wird die Zeit gestartet, die als Überwachungsfenster parametriert wurde. Ändert sich das Eingangssignal innerhalb dieser Zeit öfter als die parametrierte Anzahl Signalwechsel, so wird das als Flatterfehler erkannt. Wird innerhalb des Überwachungsfensters kein Flatterfehler erkannt, dann wird beim nächsten Signalwechsel das Überwachungsfenster erneut gestartet.

Hinweis

Wenn Sie für einen Eingangskanal eine Impulsverlängerung parametrieren, hat das auch Auswirkungen auf die für diesen Kanal freigegebene Flatterüberwachung. Das "impulsverlängerte" Signal ist das Eingangssignal für die Flatterüberwachung. Stimmen Sie deshalb die Parametrierungen von Impulsverlängerung und Flatterüberwachung aufeinander ab.

Siehe auch

Dokumentation zu Baugruppen für die Prozessautomatisierung (http://support.automation.siemens.com/WW/view/de/7215812)

IQ-Sense Modul

Eigenschaften von 8 IQ-SENSE

Eigenschaften

Das 8 IQ-SENSE Modul zeichnet sich durch folgende Eigenschaften aus:

- Anschluss von Sensoren mit IQ-SENSE®, Photoelektrische Näherungsschalter: z. B. Reflexionslichtschranken, Reflexionslicht- und Lasertaster.
- Zentral in einer S7-300 oder dezentral in einer ET 200M einsetzbar.
- An jedes Modul können bis zu 8 Sensoren angeschlossen werden. Je Sensor ist eine Zweidrahtleitung erforderlich.
- Funktionsreserve parametrierbar.
- Parametrierbare Zeitfunktionen, Schalthysterese, Synchronbetrieb
- Vorgabe von Empfindlichkeits- und Abstandswerten (*IntelliTeach* über FB "IQ-SENSE Opto")
- Teach in
- Ziehen und Stecken der Sensoren im laufenden Betrieb (Automatische Nachparametrierung)

Antiinterferenzgruppe

Nur für optische IQ-Sense-Geräte (IQ-Profil-ID 1).

Für IQ-Sense-Geräte mit IQ-Profil-ID 128 (Ultraschall) siehe bei den kanalgranularen Parametern "Multiplex-/Synchronbetrieb".

Verhinderung einer Störbeeinflussung (Interferenz, z. B. Streulicht) durch Zuweisen einer Antiinterferenzgruppe. Das bedeutet:

- Antiinterferenzgruppe: Keine (= Voreinstellung)
  Optische Sensoren an einer oder mehreren Baugruppen können sich bei ungünstiger Anordnung gegenseitig beeinflussen.
- Antiinterferenzgruppe: 3 oder 4
  Optische Sensoren an der selben Baugruppe mit Antiinterferenzgruppe 3 oder 4 können sich gegenseitig nicht beeinflussen. Optische Sensoren an unterschiedlichen Baugruppe mit Antiinterferenzgruppe 3 oder 4 können sich ebenfalls gegenseitig nicht beeinflussen. Sie müssen keinen Mindestabstand zwischen den optischen IQ-Sense-Geräten einhalten und können z. B. zwei Reflexionsschranken auf einen gemeinsamen Reflektor ausrichten.

Funktionsprinzip

Nachfolgendes Bild verdeutlicht die Funktionsweise des Parameters Antiinterferenzgruppe:

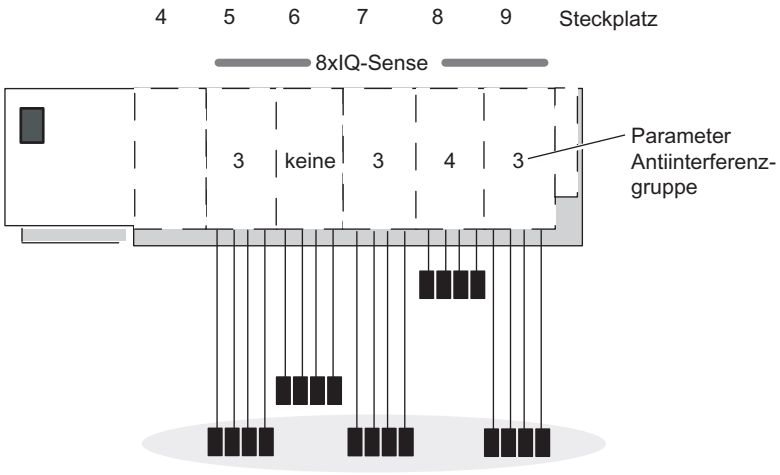

Eine gegenseitige Störbeeinflussung ist nur zwischen den optischen Sensoren der Baugruppen am Steckplatz 5, 6, 7 und 9 möglich, da sich diese in der gleichen Antiinterferenzgruppe 3 befinden bzw. "Keine" eingestellt ist.

Hinweis

Sensoren in gleichen Antiinterferenzgruppen müssen so montiert werden, dass der Mindestabstand (siehe Beipackzettel Sensor) eingehalten wird und somit eine gegenseitige Beeinflussung ausgeschlossen ist.

Geberart

Mit diesem Parameter stellen Sie die Sensorart je Kanal ein:

- Reflexionslichtschranke oder
- Reflexionslichttaster oder
- deaktiviert

Reflexionslichttaster

Tabelle 1-245 Reflexionslichttaster

Reflexionslichttaster	Objekt	
Sender Empfänger		Schaltzustand 0: Kein Objekt erkannt, d. h. der Gegenstand befindet sich nicht im Strahlengang. Der Empfänger sieht kein Licht
Sender Empfänger		Schaltzustand 1: Objekt erkannt, d. h. der Gegenstand befindet sich im Strahlengang. Der Empfänger sieht kein Licht.

Reflexionslichtschranke

Tabelle 1-246 Reflexionslichtschranke

Reflexionslichtschranke	Objekt	
Sender Empfänger		Schaltzustand 0: Kein Objekt erkannt, d. h. der Gegenstand befindet sich nicht im Strahlengang. Der Empfänger sieht Licht.
Sender Empfänger		Schaltzustand 1: Objekt erkannt, d. h. der Gegenstand befindet sich im Strahlengang. Der Empfänger sieht kein Licht.

Schalthysterese

Durch Störungen am Reflexionslichttaster oder im Fertigungsprozess kann ein sogenanntes "Signalflattern" auftreten. Dabei pendelt der Messwert um die Schaltschwelle von 100 % (Objekt erkannt – Objekt nicht erkannt). Dieses Flattern der Schaltschwelle können Sie durch den Parameter Schalthysterese verhindern. Dadurch erhalten Sie ein stabiles Ausgangssignal am Sensor.

Sie können die Bereiche 5 / 10 / 20 / 50 % als Schalthysterese parametrieren.

Voraussetzungen

Der Parameter Schalthysterese ist nur einstellbar bei Reflexionslichttastern mit Hintergrundausblendung.

Funktionsprinzip

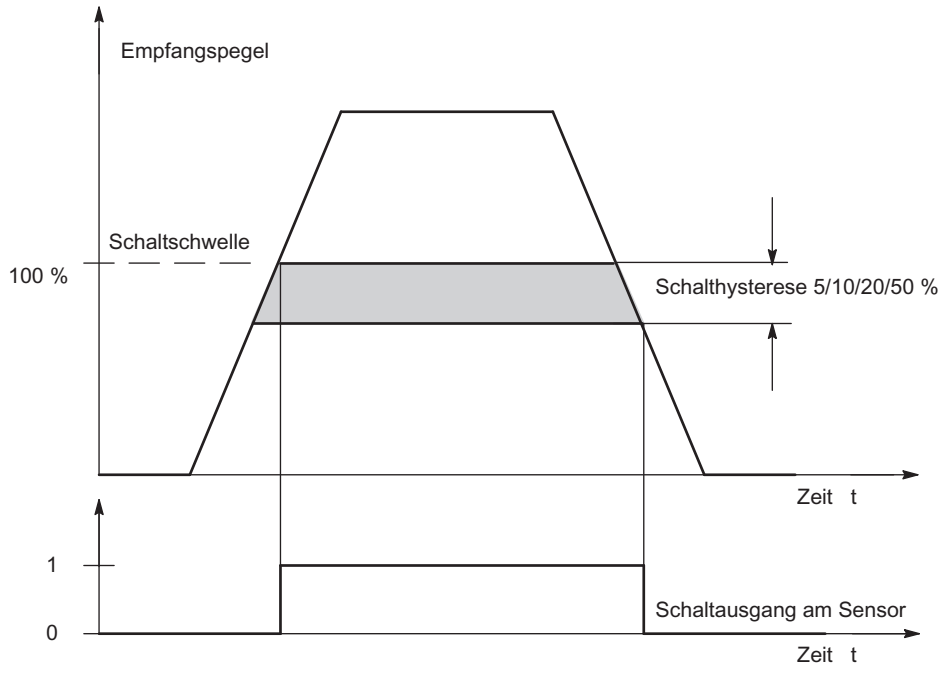

Bild 1-135 Parameter Schalthysterese

Zeitfunktionen, Zeitwert

Mit diesen Parametern können Sie das Elektronikmodul an Ihre jeweilige Applikation anpassen.

Funktionsprinzip

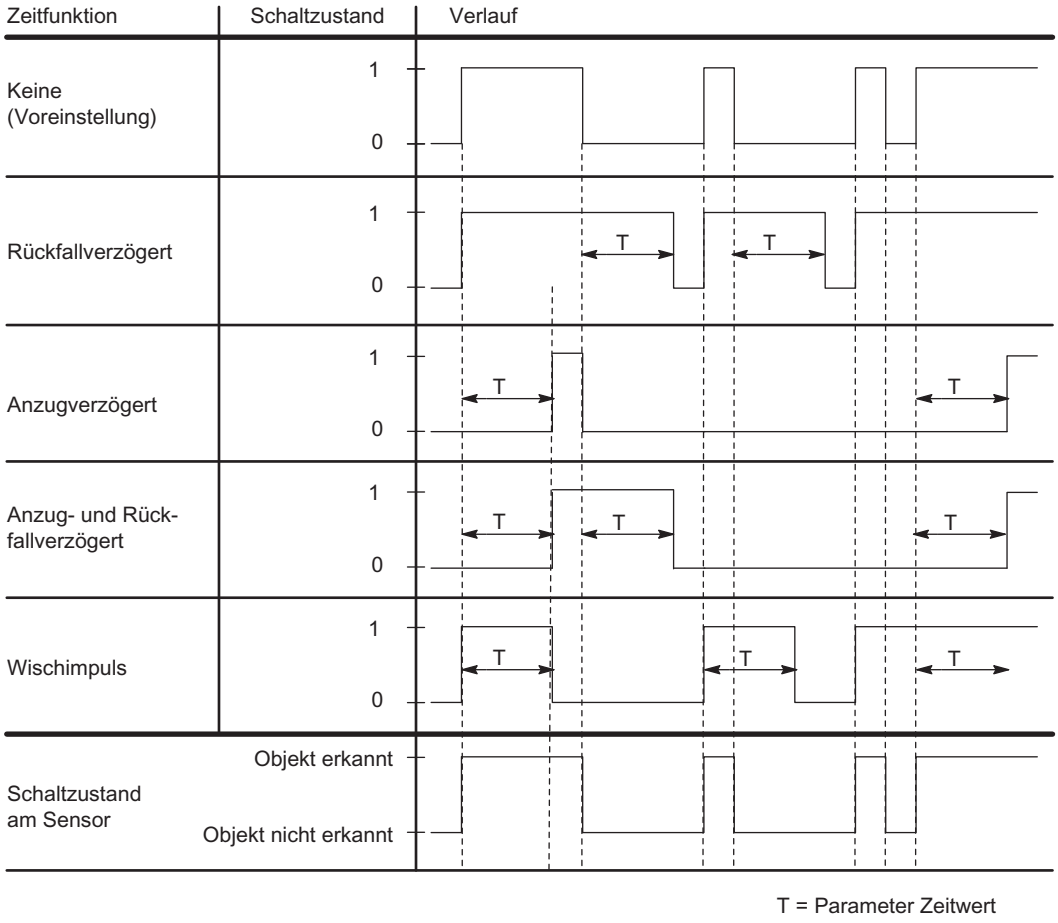

Bild 1-136 Parameter Zeitwert, Zeitfunktionen

Multiplex-/Synchronbetrieb

Zur Vermeidung gegenseitiger Beeinflussung räumlich benachbarter IQ-Sense-Ultraschallgeräte (Geräte mit IQ-Profil-ID 128) verwenden Sie den Parameter "Multiplex-/Synchronbetrieb".

Einstellungen für den Parameter Multiplex-/Synchronbetrieb

Deaktiviert: Eine gegenseitige Beeinflussung räumlich benachbarter IQ-Sense-Ultraschallsensoren ist möglich (Voreinstellung). Die Zykluszeit wird durch den IQ-Sense-Ultraschallsensor bestimmt.

Multiplex: Die IQ-Sense-Ultraschallsensoren ermitteln nacheinander den Prozesswert (Abstand), sodass eine gegenseitige Beeinflussung ausgeschlossen ist. Die Zykluszeit ist hierbei die Summe der parametrierten Synchronzykluszeiten der zu multiplexenden IQ-Sense-Ultraschallsensoren.

Synchronisation: Die IQ-Sense-Ultraschallsensoren ermitteln zum exakt gleichen Zeitpunkt den Prozesswert (Abstand), sodass eine gegenseitige Beeinflussung ausgeschlossen ist. Die Zykluszeit entspricht hierbei der größten parametrierten Synchronzykluszeit der zu synchronisierenden IQ-Sense-Ultraschallsensoren.

Den Synchronbetrieb können Sie z. B. für eine Vorhangfunktion verwenden, bei der sich mehrere parallel angeordnete IQ-Sense-Ultraschallsensoren einen ausgedehnten Erfassungsbereich teilen. Die Sensoren senden gleichzeitig einen Ultraschallimpuls aus. Wenn ein Objekt in den Erfassungsbereich eintritt, so empfängt derjenige Sensor das Echo am schnellsten, der dem Objekt am nächsten liegt. So lässt sich das Objekt nicht nur detektieren, sondern auch lokalisieren.

AFI-Wert

Mit Hilfe des AFI-Wertes (Application Family Identifier, definiert in der internationalen Norm ISO 15693-3) können Transponder für unterschiedliche Applikationen ausgewählt werden. Es werden nur Transponder bearbeitet, deren AFI-Wert mit dem am Sensor eingestellten Wert übereinstimmt. Hat ein Transponder den AFI-Wert "0", so kann er unabhängig vom AFI-Wert des Sensors identifiziert und bearbeitet werden.

Dieser Parameter hat nur Bedeutung, wenn er vom Identsystem unterstützt wird, ansonsten kann er einen beliebigen Wert (im Normalfall "0" erhalten.

Transpondertyp

Je nach Typ des Transponders müssen Sie einstellen, ob es sich um einen ISO-Transponder oder um einen herstellerspezifischen Typ handelt.

Bei Transpondern gemäß internationalem Standard nach ISO 15693 ist der Wert "1" auszuwählen, bei allen anderen Typen wird "0" eingestellt. Aufgrund dieser Einstellung wird im Sensor einer von zwei möglichen Luftschnittstellentreibern ausgewählt.

Dieser Parameter hat nur Bedeutung, wenn er vom Identsystem unterstützt wird, ansonsten kann er einen beliebigen Wert (im Normalfall "0") erhalten.

1.6.3.9 ET 200S

ET 200S konfigurieren

Einführung

Für die Familie ET 200S finden Sie im Hardware-Katalog unter "Dezentrale Peripherie" ein breites Modulspektrum.

Parametrierung

Informationen zum Konfigurieren und Parametrieren finden Sie unter "Siehe auch".

Frequenzumrichter

Einsatz des Frequenzumrichters

Frequenzumrichter

Die Frequenzumrichter ICU24 und ICU24F (als fehlersichere Ausführung) sind modular aufgebaute Frequenzumrichter, die vollständig in das dezentrale Peripheriesystem ET 200S eingebettet sind. Im Folgenden erfahren Sie, was Sie bei der Parametrierung beider Baugruppen beachten müssen.

Telegramm

Die Telegrammnummer und Betriebsart der Baugruppe werden nur angezeigt und sind nicht änderbar.

Anwendungs-ID

Mit der Anwendungs-ID kennzeichnen Sie die Gesamtheit der im Frequenzumrichter gespeicherten Parameter. Tragen Sie eine Anwendungs-ID aus dem Wertebereich 0 bis 65535 ein. Beim Anlauf (bzw. Ziehen/Stecken) wird diese Kennung mit der auf dem Umrichter gespeicherten Anwendungs-ID verglichen.

Umrichter, die identische Applikationen bearbeiten, sind normalerweise auch identisch parametriert und sollten mit derselben Anwendungs-ID gekennzeichnet werden. Umrichter mit derselben Anwendungs-ID dürfen untereinander getauscht werden .Ebenso wird das Kopieren der kompletten Parametrierung von einem Umrichter auf einen anderen, beispielsweise über eine MMC, nur dann akzeptiert, wenn beide dieselbe Anwendungs-ID haben.

Umrichter, die unterschiedliche Anwendungen bearbeiten und unterschiedlich parametriert sind, müssen durch verschiedene Anwendungs-IDs gekennzeichnet werden. Damit wird verhindert, dass ein Umrichter mit unpassender Parametrierung auf einem falschen Steckplatz, d. h. an der falschen Anwendung, anläuft. Ebenso wird verhindert, dass die im Umrichter gespeicherte Parametrierung zufälligerweise mit einer beliebigen Parametrierung, die auf einer MMC gespeichert ist, überschrieben werden kann.

Diagnosealarm freigeben

Sie können für die Frequenzumrichter den Diagnosealarm freigeben. Bei freigegebenem Diagnosealarm muss in der CPU ein OB 82 vorhanden sein, der die Diagnoseereignisse verarbeitet.

Siehe auch

Dokumentation zum Frequenzumrichter (http://support.automation.siemens.com/WW/view/de/26291825)

1.6.3.10 ET 200pro

ET 200pro mit eigener CPU-Baugruppe

ET 200pro mit eigener CPU

Die ET 200pro kann mit einer eigenen CPU ausgestattet sein, zum Beispiel mit der CPU 1516pro.

Beachten Sie dabei die folgenden Regeln.

Ausbauregeln bei der CPU 1516pro:

- Sie können die CPU 1516pro mit bis zu 16 Modulen erweitern.
- Die Gesamtbreite darf maximal einen Meter betragen.

Ausbauten mit breiten Modulen

Wenn bei einer Projektierung mit großen Baugruppen (zum Beispiel Frequenzumrichter oder Motorstarter) die maximale Breite von einem Meter überschritten wird, dann lässt sich die Projektierung nicht übersetzen. Reduzieren Sie dann den Ausbau.

Konfigurationssteuerung bei ET 200pro

Funktionsprinzip

Durch die Konfigurationssteuerung ist es möglich, die ursprüngliche Konfiguration einer ET 200pro (erfolgte durch die Projektierung mit STEP 7) im Nachhinein durch ein Anwenderprogramm zu ändern und die ET 200pro in dieser geänderten Konfiguration zu betreiben. Für diese Konfiguration ist STEP 7 nicht mehr erforderlich: Sie teilen der ET 200pro über Ihr Anwenderprogramm mit, auf welchem Steckplatz ein projektiertes Modul tatsächlich gesteckt ist.

Dazu verwenden Sie den Steuerdatensatz 196. In diesem Datensatz kodieren Sie, welche Module abweichend von der Projektierung im realen Aufbau fehlen oder welche Module sich abweichend von der Projektierung auf einem anderen Steckplatz befinden. Auf die Parametrierung der Module (zum Beispiel das Freigeben von Diagnosemeldungen) hat die Konfigurationssteuerung keinen Einfluss.

Anschließend rufen Sie die Anweisung "WRREC" auf und schreiben damit den Datensatz an das Interface-Modul der ET 200pro.

Durch diese Konfigurationssteuerung können Sie flexibel den Aufbau einer ET 200pro variieren, solange sich die reale Konfiguration aus der projektierten Konfiguration eines Maximalausbaus (ursprünglich erstellt durch STEP 7) ableiten lässt.

Im Folgenden ist beschrieben, wie Sie die Konfigurationssteuerung aktivieren. Zudem wird gezeigt, wie der Steuerdatensatz 196 und der Rücklesedatensatz 197 aufgebaut sind.

Voraussetzungen

Der CPU-Anlaufparameter "Vergleich Sollausbau zu Istausbau" ist eingestellt auf Anlauf auch bei Unterschieden (Voreinstellung). Diese Einstellung ist auch für den Anlaufparameter der einzelnen Module der ET 200pro gewählt.

Konfigurationsste

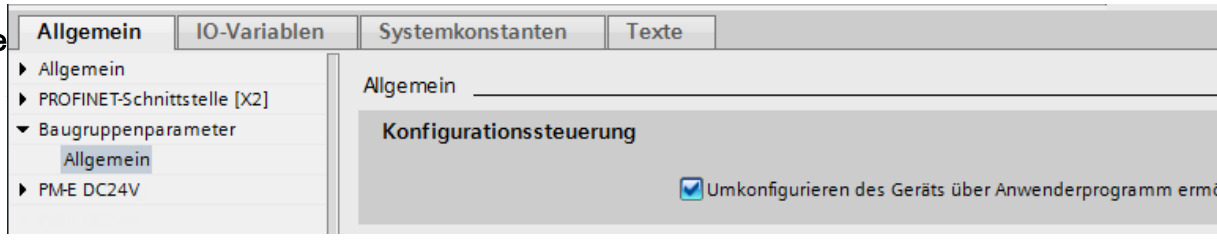

Aufbau des Steuerdatensatzes 196 für die ET 200pro

Für jedes Modul ist Steuerdatensatz 196 ein Byte vorgesehen.

Die Position dieses Bytes im Datensatz kodiert jeweils ein Modul bei der ursprünglichen Projektierung mit STEP 7:

- "slot_IM" (Byte 4 im Datensatz, Bild unten) entspricht dem Modul auf Steckplatz 1 in der Projektierung
- "slot_2" (Byte 5) entspricht dem Modul auf Steckplatz 2 in der Projektierung.
- "slot_3" (Byte 6) entspricht dem Modul auf Steckplatz 3 in der Projektierung.
- usw.

Byte "slot_x"

Der aktuelle Steckplatz ist durch die Zahl kodiert, die "slot_x" zugewiesen wird (durch seinen Wert). Beispiele:

- Der Wert "2" im Byte 5 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, auch in der aktuellen Konfiguration den Platz 2 zu (slot_2 = 2).
- Der Wert "3" im Byte 5 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Platz 3 zu (slot_2 = 3).
- Der Wert "4" im Byte 5 bedeutet, Sie weisen dem Modul, das ursprünglich auf Steckplatz 2 steckte, in der aktuellen Konfiguration den Platz 4 zu (slot_2 = 4).
- usw.

Bei der ET 200pro gibt es keine Reservemodule (wie bei der ET 200S oder BU-Cover-Module bei der ET 200SP), deshalb darf das Bit 7 von "slot_x" nicht gesetzt sein.

Der Wert "0" in "slot_x" drückt aus, dass dieses Modul in der aktuellen Konfiguration nicht gesteckt ist.

Beispiel für einen Steuerdatensatz 196

Das folgende Bild zeigt den Steuerdatensatz 196 für eine Projektierung einer ET 200pro mit vier Modulen.

Diese Projektierung soll als Beispiel dienen:

- Das ursprünglich mit STEP 7 projektierte Modul auf Steckplatz 1 steckt in der aktuellen Konfiguration auch auf Steckplatz 1.
- Das Modul auf Steckplatz 2 steckt in der aktuellen Konfiguration auf Steckplatz 2.
- Das Modul auf Steckplatz 3 fehlt in der aktuellen Konfiguration.
- Das Modul auf Steckplatz 4 steckt in der aktuellen Konfiguration tatsächlich auf Steckplatz 3.

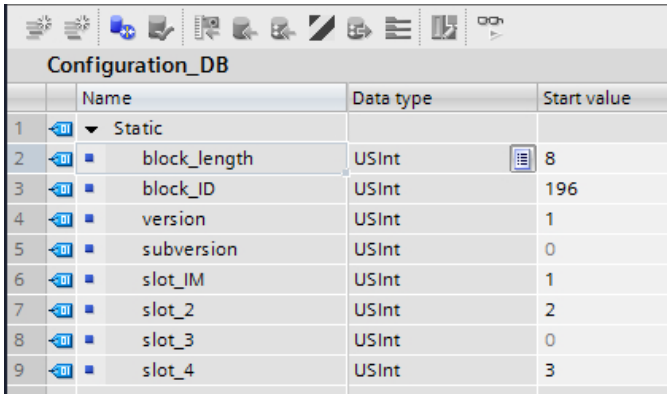

Configuration_DB

	Name	Data type	Start value
1	Static		
2	block_length	USInt	8
3	block_ID	USInt	196
4	version	USInt	1
5	subversion	USInt	0
6	slot_IM	USInt	1
7	slot_2	USInt	2
8	slot_3	USInt	0
9	slot_4	USInt	3

Die Komponenten des Steuerdatensatzes 196 (Bild oben):

- block_length: Notieren Sie hier die Länge des Steuerdatensatzes, im Beispiel 8 (Byte). Die Länge errechnet sich nach der Formel: "Anzahl der belegten Steckplätze" + 4.
- block_ID: Tragen Sie hier die Zahl 196 ein.
- version: Die ET200pro verwendet die Version 1 des Steuerdatensatzes 196.
- subversion: Die ET200pro verwendet die Subversion 0 des Steuerdatensatzes 196.
- slot_IM: Auf Steckplatz 0 befindet sich bei der ET 200pro immer das IM-Modul. Auf Steckplatz 1 befindet sich immer das virtuelle Powermodul, das fest im IM-Modul integriert ist. Im "slot_IM" (Name änderbar) können beliebige Werte stehen. Dieses Byte wird bei der Konfigurationsteuerung der ET 200pro nicht interpretiert.
- slot_2: Das projektierte Modul 2 steckt im Steckplatz 2 (Wert "2").
- slot_3: Das projektierte Modul 3 fehlt in der aktuellen Konfiguration (Wert "0").
- slot_4: Das projektierte Modul 4 steckt bei der aktuellen Konfiguration im Steckplatz 3 (Wert "3").

Definition des Steuerdatensatzes 196

Für die Konfigurationssteuerung wird ein Steuerdatensatz 196 definiert, der eine Steckplatzzuordnung enthält.

Byte	Element	Wert	Erläuterungen
0	Blocklänge	z.B. 8 für ET 200pro mit vier Modulen	Die Länge des Datensatzes errechnet sich nach der Formel: 4 + Anzahl der Module in Byte
1	Block-ID	196	ID für den Steuerdatensatz 196
2	Version	1	Version 1 des Steuerdatensatzes 196
3	Subversion	0	Subversion 0 des Steuerdatensatzes 196
4	Steckplatz_1	Beliebige Werte möglich: zum Beispiel "1"	Bei der ET 200pro wird dieses Byte nicht interpretiert, da sich auf Steckplatz 1 immer das Powermodul befindet, das fest in der IM-Baugruppe der ET 200pro integriert ist.
5	Steckplatz_2	Kodierung des tatsächlichen Steckplatzes: 2 = Steckplatz 2 3 = Steckplatz 3 4 = Steckplatz 4 usw. Kodierung für fehlendes Modul: 0 = kein Steckplatz, Modul fehlt	Byte 5 enthält die Angabe, wo das ursprünglich mit STEP 7 auf Platz 2 projektierte Modul in der aktuellen Konfiguration tatsächlich steckt. Beispiel: 2 Das ursprünglich mit STEP 7 projektierte Modul auf Steckplatz 2 befindet sich in der aktuellen Konfiguration auch tatsächlich auf Steckplatz 2 (Wert "2").
6	Steckplatz_3	Kodierung des tatsächlichen Steckplatzes: 2 = Steckplatz 2 3 = Steckplatz 3 4 = Steckplatz 4 usw. Kodierung für fehlendes Modul: 0 = kein Steckplatz, Modul fehlt	Byte 6 enthält die Angabe, wo das ursprünglich mit STEP 7 auf Platz 3 projektierte Modul in der aktuellen Konfiguration tatsächlich steckt. Beispiel: 0 Das ursprünglich mit STEP 7 projektierte Modul auf Steckplatz 3 fehlt in der aktuellen Konfiguration (Wert "0")

7	Steckplatz_4	Kodierung des tatsächlichen Steckplatzes: 2 = Steckplatz 2 3 = Steckplatz 3 4 = Steckplatz 4 usw. Kodierung für fehlendes Modul: 0 = kein Steckplatz, Modul fehlt	Byte 7 enthält die Angabe, wo das ursprünglich mit STEP 7 auf Platz 4 projektierte Modul in der aktuellen Konfiguration tatsächlich steckt. Beispiel: 3 Das ursprünglich mit STEP 7 projektierte Modul auf Steckplatz 4 befindet sich in der aktuellen Konfiguration tatsächlich auf Steckplatz 3 (Wert "3").
:	:	:	:

Datensatz schreiben

Übertragen Sie den Steuerdatensatz zur Baugruppe ET 200pro.

Rufen Sie dazu die erweiterte Anweisung WRREC (Datensatz schreiben) auf und übertragen Sie den erstellten Steuerdatensatz.

Übertragen Sie keinen Steuerdatensatz, so verwendet das Interface-Modul die ursprüngliche Projektierung mit STEP 7. Hier gilt: Projektiertes Modul x ist im realen Steckplatz x eingesteckt.

Adressierung des Interfacemoduls über HW-Kennung

Um den Datensatz 196 mit der Anweisung WRREC zu übertragen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z. B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Fehlermeldungen

Beim Schreiben des Steuerdatensatzes 196 werden Ihnen im Fehlerfall folgende Fehlermeldungen zurückgegeben:

Tabelle 1-247 Fehlermeldungen

Fehlercode	Bedeutung
16#80A2	DP-Protokollfehler auf Layer 2. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.
16#80B1	Unzulässige Länge; Die Längenangabe im Datensatz 196 ist nicht korrekt.
16#80B5	Konfigurationssteuerung nicht parametriert.
16#80B2	invalid slot: Der projektierte Steckplatz ist nicht belegt.

Fehlercode	Bedeutung
16#80B8	Parameterfehler; Modul meldet ungültige Parameter.
16#80C5	DP-Slave oder Modul nicht verfügbar. Deutet auf systembedingt ausgebliebene Datensatzquittung hin.

Rückmeldedatensatz 197 für ET 200pro

Der Rückmeldedatensatz 197 dient zum Lesen des tatsächlichen Aufbaus einer Station (hier einer ET 200pro).

Durch diesen Datensatz kann der reale Ausbau der ET 200pro (Istkonfiguration) überprüft werden. Der Rückmeldedatensatz enthält für jedes projektierte Modul die Angabe, ob es tatsächlich vorhanden ist oder nicht.

- Der Wert "1" bedeutet, dass das richtige Modul auf dem vorgesehenen Steckplatz steckt.
- Der Wert "0" kodiert alle anderen Möglichkeiten (falsches Modul, leerer Steckplatz).

Beispiel:

Ein Modul wurde mit STEP 7 auf Steckplatz 4 projektiert.

Dann wurde dieses Modul mithilfe des Datensatzes 196 in der aktuellen Konfiguration auf Steckplatz 3 verschoben.

Wenn dieses Modul auch tatsächlich auf Steckplatz 3 steckt, dann wird dies durch den Wert "1" kodiert (status_slot_4 = 1).

Der Aufbau im Einzelnen:

Der Aufbau des Datenbausteins entspricht der ursprünglichen Projektierung der ET 200pro mit STEP 7.

Für jedes Modul ist ein Byte im Datensatz vorgesehen. Die Position dieses Bytes im Datensatz entspricht der Position eines Moduls bei der ursprünglichen Projektierung mit STEP 7.

Reihenfolge der Bytes:

- "status_slot_IM" (Byte 4 im Datensatz) entspricht dem Modul auf Steckplatz 1 der Projektierung,
- "status_slot_2" (Byte 5) entspricht dem Modul auf Steckplatz 2 der Projektierung
- "status_slot_3" (Byte 6) entspricht dem Modul auf Steckplatz 3 der Projektierung,
- usw.

Das folgende Beispiel ist für eine Projektierung mit 4 Modulen aufgebaut. Deshalb steht im Element "block_length" des Datensatzes der Wert "8".

Wenn Sie in STEP 7 eine ET 200pro mit weniger Modulen projektieren, dann verkürzt sich der Datenbaustein.

Die Namen der Komponenten des Steuerdatensatzes (zum Beispiel "status_slot_2") können Sie frei wählen.

Die Bedeutung von "status_slot_x":

- Der Wert "1" in satus_slot_x bedeutet, dass das Modul x auf dem vorgesehenen Steckplatz steckt
- Der Wert "0" in satus_slot_x kodiert alle anderen Möglichkeiten (falsches Modul, Modul nicht vorhanden).

Beispiel:

Das folgende Bild zeigt den Rücklesedatensatz 197 für eine ET 200pro mit vier Modulen.

- Das Modul 3 fehlt (dies wurde im Steuerdatensatz 196 angewiesen, siehe den Abschnitt "Beispiel für einen Steuerdatensatz 196" oben)
- Die drei anderen Module sind in der ET 200pro tatsächlich gesteckt.

Record_197_ET_200pro

	Name	Datentyp	Defaultwert
1	block_length	USInt	8
2	block_ID	USInt	197
3	version	USInt	1
4	subversion	USInt	0
5	status_slot_IM	USInt	1
6	status_slot_2	USInt	1
7	status_slot_3	USInt	0
8	status_slot_4	USInt	1

Rücklesedatensatz 197 lesen

Mit der Anweisung RDREC können Sie den Rücklesedatensatz 197 von der ET 200pro lesen. RDREC arbeitet asynchron. Wenn Sie RDREC im Anlauf-OB aufrufen, dann müssen Sie die Anweisung über eine Schleife mehrmals aufrufen, bis die Ausgangsparameter "BUSY" oder "DONE" anzeigen, dass der Datensatz gelesen ist.

Um den Datensatz 197 mit der Anweisung RDREC zu lesen, müssen Sie die HW-Kennung des IM-Submoduls mit der Endung "~Head" als Eingangsparameter für die Anweisung eintragen. Die Systemkonstante dieser HW-Kennung heißt z B. "IO-Device_2~Head". Die Systemkonstanten eines markierten Gerätes werden z. B. in der Netzsicht im Register "Systemkonstanten" angezeigt. Verwenden Sie den zugehörigen Wert für die Adressierung.

Weiterführende Informationen und Beispiele

Weitere Informationen zum PN-Interfacemodul der ET 200pro finden Sie hier (http://support.automation.siemens.com/WW/view/de/98099372).

Konkrete Beispiele für die Konfigurationssteuerung finden Sie hier in dieser Applikationsbeschreibung (http://support.automation.siemens.com/WW/view/de/29430270).

Eine Bibliothek der Datensätze und weitere Anwendungsbeispiele sind hier abgelegt:

Vorlagen für die Datensätze (http://support.automation.siemens.com/WW/view/de/29430270)

Siehe auch

Konfigurationssteuerung bei ET 200SP (Seite 3384)

Einsatz des Frequenzumrichters

Frequenzumrichter

Die Frequenzumrichter ET 200pro FC und ET 200pro F-FC (als fehlersichere Ausführung) sind modular aufgebaute Frequenzumrichter, die vollständig in das dezentrale Peripheriesystem ET 200pro eingebettet sind. Im Folgenden erfahren Sie, was Sie bei der Parametrierung beider Baugruppen beachten müssen.

Telegramm

Die Telegrammnummer und Betriebsart der Baugruppe werden nur angezeigt und sind nicht änderbar.

Anwendungs-ID

Mit der Anwendungs-ID kennzeichnen Sie die Gesamtheit der im Frequenzumrichter gespeicherten Parameter. Tragen Sie eine Anwendungs-ID aus dem Wertebereich 0 bis 65535 ein. Beim Anlauf (bzw. Ziehen/Stecken) wird diese Kennung mit der auf dem Umrichter gespeicherten Anwendungs-ID verglichen.

Umrichter, die identische Applikationen bearbeiten, sind normalerweise auch identisch parametriert und sollten mit derselben Anwendungs-ID gekennzeichnet werden. Umrichter mit derselben Anwendungs-ID dürfen untereinander getauscht werden .Ebenso wird das Kopieren der kompletten Parametrierung von einem Umrichter auf einen anderen, beispielsweise über eine MMC, nur dann akzeptiert, wenn beide dieselbe Anwendungs-ID haben.

Umrichter, die unterschiedliche Anwendungen bearbeiten und unterschiedlich parametriert sind, müssen durch verschiedene Anwendungs-IDs gekennzeichnet werden. Damit wird verhindert, dass ein Umrichter mit unpassender Parametrierung auf einem falschen Steckplatz, d. h. an der falschen Anwendung, anläuft. Ebenso wird verhindert, dass die im Umrichter gespeicherte Parametrierung zufälligerweise mit einer beliebigen Parametrierung, die auf einer MMC gespeichert ist, überschrieben werden kann.

Diagnosealarm freigeben

Sie können für die Frequenzumrichter den Diagnosealarm freigeben. Bei freigegebenem Diagnosealarm muss in der CPU ein OB 82 vorhanden sein, der die Diagnoseereignisse verarbeitet.

1.6.4 IPv4-Routing beim CP 1543-1 / CM 1542-1

IPv4-Routing beim CP 1543-1 / CM 1542-1

Für die Funktion "IPv4-Routing über den Rückwandbus" des CM/CP gelten die folgenden Einschränkungen.

- Das IPv4-Routing über das CM / den CP in die CPU der Station ist nicht möglich.
- IP-Konfiguration beim CM 1542-1 als PROFINET IO-Controller
  Wenn das CM mit aktiviertem IPv4-Routing über den Rückwandbus als PROFINET IO-Controller eingesetzt wird, dann beachten Sie Folgendes für die IP-Konfiguration der unterlagerten IO-Devices:
  Wenn die IP-Adresse der unterlagerten IO-Devices fest projektiert wurde (Option "IP-Adresse im Projekt einstellen"), dann ist kein IPv4-Routing über das CM zu den IO-Devices möglich.
  Um IPv4-Routing über das CM zu den unterlagerten IO-Devices zu ermöglichen, muss in der IP-Konfiguration der IO-Devices folgendes eingestellt werden:
  - Aktivieren Sie beim IO-Device zunächst die Option "Router verwenden".
    Kopieren Sie die IP-Adresse des IO-Controllers (CM) in das Feld der Router-Adresse.
  - Aktivieren Sie beim IO-Device dann die Option "Anpassen der IP-Adresse direkt am Gerät erlauben".

1.6.5 IPv6-Konfiguration

1.6.5.1 IPv6-Protokoll

Das Internet-Protokoll Version 6 (IPv6) erweitert den Adressbereich des Internet-Protokolls Version 4 (IPv4).

Adressformat IPv6: Notation

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vierstelligen Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt.

Beispiel: fd00:0000:0000:0000:0db8:002f:0000:0000

Regeln / Vereinfachungen:

- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.
  Beispiel: fd00:0000:0000:0000:0db8:002f:0000:0000 abgekürzt fd00:0:0:0:db8:2f:0:0
- Wenn ein oder mehr Blöcke den Wert 0 haben, ist eine verkürzte Schreibweise mit zwei aufeinanderfolgenden Doppelpunkten möglich. Damit Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden. Bei zwei oder mehr solcher Verkürzungen kann unter Umständen nicht ermittelt werden, wie viele Blöcke gekürzt wurden.
  Gültiges Beispiel: fd00::db8:2f:0:0 oder fd00:0:0:0:db8:2f:: (beide eindeutig erkennbar als fd00:0:0:0:db8:2f:0:0)
  Ungültiges Beispiel: fd00::db8:2f:: (nicht eindeutig, da interpretierbar als fd00:0:0:0:db8:2f:0:0 oder fd00:0:0:db8:2f:0:0:0 usw.)
- Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.
  Beispiel: Die IPv6-Adresse fd00::db8:fd01:2f ist äquivalent zu fd00::db8.253.1.0.47

Die Eingabe von IPv6-Adressen ist in den zuvor beschriebenen Notationen möglich. IPv6-Adressen werden immer in der Notation der Eingabe angezeigt.

Unterschiedliche IPv6-Adressen einer Schnittstelle

Wenn Sie für eine Schnittstelle eines Kommunikationsmoduls mehrere IPv6-Adressen projektieren, dann achten Sie darauf, dass die beiden IPv6-Adressen unterschiedliche Subnetzbereiche abdecken.

Siehe auch

IPv6 beim CP 1543-1 (Seite 3473)

1.6.5.2 IPv6 beim CP 1543-1

IPv6-Verwendung beim CP 1543-1

Der CP unterstützt für sämtliche IP-Dienste das Internet Protocol Version 4 (IPv4)

Die zusätzliche Angabe von Adressen im IPv6-Format ist beim CP für folgende Dienste und Anwendungsfälle nutzbar:

- FETCH/WRITE: Direkte Schreib-/Lesezugriffe von PC-Stationen, SIMATIC S5 oder Fremdgeräten
- FTP-Client: FTP-Zugriffe von der S7-1500 CPU aus auf einen FTP-Server mit Programmbaustein FTP_CMD
- FTP-Server: FTP-Zugriffe von einem FTP-Client aus auf Datenbereiche der S7-1500 CPU
- SNMP: Datenabfrage über MIB-Objekte gemäß SNMP
- E-Mail: Datenübertragung von der S7-1500 CPU aus mit Programmbaustein T_Mail

Geräte und Netze diagnostizieren 2

2.1 Hardware diagnostizieren

2.1.1 Überblick über die Hardware-Diagnose

2.1.1.1 Prinzipielle Möglichkeiten, Hardware zu diagnostizieren

Prinzipielle Möglichkeiten der Hardware-Diagnose

Sie können die Hardware wie folgt diagnostizieren:

- Über die Online- und Diagnosesicht
- Über die Task Card "Online-Tools"
- Über den Bereich "Diagnose > Geräte-Information" des Inspektorfensters
- Über Diagnosesymbole z. B. in der Gerätesicht und in der Projektnavigation

Aufbau der Online- und Diagnosesicht

Die Online- und Diagnosesicht besteht aus zwei nebeneinander liegenden Fenstern:

- Das linke Fenster zeigt eine Baumstruktur mit Ordnern und - falls Sie die Ordner aufgeklappt haben - Gruppen.
- Das rechte Fenster enthält Detailinformationen zum selektierten Ordner bzw. zur selektierten Gruppe.

Es gibt die Gruppe "Online-Zugänge" und die Ordner "Diagnose" und "Funktionen":

- Gruppe "Online-Zugänge": Sie enthält die Anzeige, ob momentan eine Online-Verbindung zum zugehörigen Ziel besteht oder nicht. Sie können darüber hinaus die Online-Verbindung herstellen bzw. trennen.
- "Diagnose": enthält mehrere Diagnosegruppen zur selektierten Baugruppe.
- "Funktionen": enthält mehrere Gruppen, in denen Sie Einstellungen auf der selektierten Baugruppe vornehmen bzw. Kommandos an die Baugruppe ausgeben.

Funktion und Aufbau der Task Card "Online-Tools"

In der Task Card "Online-Tools" können Sie bei Baugruppen mit einem eigenen Betriebszustand (z. B. bei CPUs) aktuelle Diagnoseinformationen auslesen und Kommandos an die Baugruppe übertragen.

Falls Sie vor der Aktivierung der Task Card "Online-Tools" eine Baugruppe ohne eigenen Betriebszustand oder mehrere Baugruppen selektiert haben, bezieht sich die Task Card auf die zugehörige CPU.

Die Task Card "Online-Tools" besteht aus folgenden Paletten:

- CPU-Bedienpanel
- Zykluszeit
- Speicher

Hinweis

Eine Palette ist nur dann mit Inhalt gefüllt, wenn die Baugruppe die zugehörigen Funktionen beherrscht sowie eine Online-Verbindung besteht.

Falls keine Online-Verbindung zur zugehörigen Baugruppe besteht, erfolgt in jeder Palette die Anzeige "Keine Online-Verbindung" in blauer Schrift. Wenn eine bestehende Online-Verbindung getrennt wurde, wird "Dieses Ziel ist nicht erreichbar" angezeigt.

Aufbau des Registers "Diagnose" des Inspektorfensters

Das Register "Diagnose" des Inspektorfensters besteht selbst wiederum aus mehreren Registern. Davon ist das folgende für die Hardware-Diagnose relevant:

- Geräte-Information
  Dieses Register bezieht sich auf alle Online Devices (z. B. CPUs), zu denen eine Online-Verbindung besteht, und auf die Geräte, die diesen Online Devices zugeordnet sind (z. B. auf PROFINET-Devices und PROFIBUS-Slaves). Hier werden Meldungen zu den gestörten Online Devices und den gestörten Geräten ausgegeben.

Hinweis

Was wird angezeigt, wenn ein Modul gestört ist?

Wenn ein Modul innerhalb eines Geräts gestört ist, wird nur das zugehörige Gerät bzw. dessen Proxy (z. B. Kopfmodul) angezeigt, nicht aber das Modul selbst.

Die fehlerhaften Geräte werden "auf der obersten Stufe" angezeigt und nicht in einer hierarchischen Sicht unter ihrem Online Device (wie es in der Projektnavigation der Fall ist).

2.1.1.2 Unter den online verbundenen Geräten diejenigen ermitteln, die gestört sind

Übersicht über die gestörten Geräte

Im Bereich "Diagnose > Geräte-Information" des Inspektorfensters erhalten Sie eine Übersicht über die gestörten Geräte, zu denen eine Online-Verbindung besteht oder bestanden hat.

Der Bereich "Diagnose > Geräte-Information" des Inspektorfensters besteht aus folgenden Elementen:

- Kopfzeile mit der Anzahl der gestörten Geräte
- Tabelle mit Detailinformationen zu jedem gestörten Gerät

Wenn Sie den Aufbau der Online-Verbindung zu einem Gerät angestoßen haben, gilt es als gestört, wenn es nicht erreichbar ist oder mindestens einen Fehler meldet oder nicht im Betriebszustand "RUN" ist.

Aufbau der Tabelle mit Detailinformationen zu den gestörten Geräten

Die Tabelle besteht aus den folgenden Spalten:

- Online-Status: enthält den Online-Status als Diagnosesymbol und in Worten
- Betriebszustand: enthält den Betriebszustand als Symbol und in Worten
- Gerät / Baugruppe: Name des betroffenen Geräts bzw. der betroffenen Baugruppe
- Meldung: erläutert den Eintrag der vorhergehenden Spalte
- Details: Der Link öffnet die zum Gerät gehörende Online- und Diagnosesicht bzw. legt sie in den Vordergrund. Falls keine Online-Verbindung mehr besteht, öffnet der Link den Dialog zum Aufbau der Verbindung.
- Hilfe: Der Link liefert weitere Informationen zur aufgetretenen Störung.

Siehe auch

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 3477)

2.1.1.3 Diagnose- und Vergleichsstatus mittels Symbolen anzeigen

Diagnosestatus online ermitteln und mittels Symbolen anzeigen

Beim Aufbau der Online-Verbindung zu einem Gerät wird auch sein Diagnosestatus und ggf. der seiner unterlagerten Komponenten und ggf. sein Betriebszustand ermittelt.

Im Folgenden wird beschrieben, in welcher Sicht welche Symbole angezeigt werden.

- Gerätesicht
  - Bei jeder Hardware-Komponente (mit Ausnahme des Signalboards auf der CPU) wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Bei Hardware-Komponenten mit eigenem Betriebszustand wird zusätzlich das Betriebszustandssymbol links oder oberhalb vom Diagnosesymbol angezeigt.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es werden keine Diagnosesymbole angezeigt (wegen der Projektierung als GSDML-Gerät).

Hinweis

Diagnosestatus von SCALANCE-XM-400- und SCALANCE-XR-500-managed-Modulen

Bei einer bestehenden Online-Verbindung zu den SCALANCE-XM-400- und SCALANCE-XR-500-managed-Modulen werden falsche oder keine Diagnosesymbole angezeigt.

Sie erhalten die korrekten Diagnosesymbole, wenn Sie die Online-Verbindung zu einer zugehörigen CPU aufbauen.

- Geräteübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Für die der CPU zugeordneten Module wird das zugehörige Diagnosesymbol angezeigt (Die nicht zugeordneten Module erhalten kein Diagnosesymbol.). Für die steckbaren Submodule eines zugeordneten Moduls wird das zugehörige Diagnosesymbol angezeigt (Die nicht steckbaren Submodule sind nicht sichtbar und erhalten folglich auch kein Diagnosesymbol.).
- Netzsicht
  - Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).

- Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
- Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.

- Netzübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.
- Topologiesicht
  - Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für jeden Port wird das zugehörige Diagnosesymbol angezeigt. Die Bedeutung der einzelnen Farben ist weiter unten beschrieben.
  - Jede Leitung zwischen zwei Ports, die online sind, erhält die ihrem Diagnosezustand entsprechende Farbe.
    Die Farbe der Leitung zwischen zwei Ports ist abhängig vom Status der einzelnen Ports:

Farbe des ersten Ports	Farbe des zweiten Ports	Farbe der Verbindungsleitung
hellgrün	hellgrün	hellgrün
hellgrün	dunkelgrün	dunkelgrün
grün	grau	grau
grün	rot	rot
grau	rot	rot

  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.
- Topologieübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
    Ein Doppelklick auf das Diagnosesymbol startet die Online- und Diagnosesicht (sofern vorhanden).
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.

  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.

- Projektnavigation
  - Hinter jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten (z. B. Dezentrale Peripherie, Slave_1) in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Bei Hardware-Komponenten mit eigenem Betriebszustand wird zusätzlich das Betriebszustandssymbol in die rechte obere Ecke des Diagnosesymbols eingeblendet.
  - Falls auf einer CPU Forcen aktiv ist, wird am linken Rand des Diagnosesymbols ein rotes F eingeblendet.
  - Hinter dem Ordner "Lokale Baugruppen" wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einer der zugehörigen Baugruppen ein Hardware-Fehler vorliegt.
  - Hinter dem Ordner "Dezentrale Peripherie" wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einer der zugehörigen Komponenten ein Hardware-Fehler vorliegt.
  - Hinter dem Projektordner wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn hinter mindestens einem der Ordner "Lokale Baugruppen" und "Dezentrale Peripherie" das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt wird.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Für die der CPU zugeordneten Module wird das zugehörige Diagnosesymbol angezeigt (Die nicht zugeordneten Module sind gegraut und erhalten kein Diagnosesymbol.). Für die steckbaren Submodule eines zugeordneten Moduls wird das zugehörige Diagnosesymbol angezeigt (Die nicht steckbaren Submodule sind nicht sichtbar und erhalten folglich auch kein Diagnosesymbol.).

Hinweis

Wenn bei einer Hardware-Komponente die Diagnose "ist von der CPU aus nicht erreichbar" gestellt wird, wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" nicht zusätzlich eingeblendet.

Diagnosesymbole für Baugruppen und Geräte

Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

Symbol	Bedeutung
	Die Verbindung zu einer CPU wird gerade aufgebaut.
	Unter der eingestellten Adresse ist die CPU nicht erreichbar.
	Die projektierte CPU und die tatsächlich vorhandene CPU sind vom Typ her inkompatibel.
	Beim Aufbau der Online-Verbindung zu einer geschützten CPU wurde der Passwort-Dialog ohne Eingabe des korrekten Passworts abgebrochen.
	Keine Störung
	Wartungsbedarf
	Wartungsanforderung
	Fehler
	Die Baugruppe bzw. das Gerät ist deaktiviert.
	Die Baugruppe bzw. das Modul bzw. das Gerät ist von der CPU aus nicht erreichbar (gültig für Baugruppen und Module und Geräte unterhalb einer CPU).
	Die Funktionalität des Moduls bzw. Submoduls steht nicht zur Verfügung (z. B. Eingangs- und Ausgangsdaten). Mögliche Ursachen: • Soll-/Ist-Unterschied • Zugriffsfehler beim Aktualisieren der Prozessabbilder
	Eine Diagnose ist nicht oder nur eingeschränkt möglich. Mögliche Ursachen: • Unterschiede zwischen aktuellen Online- und Offline-Konfigurationsdaten • Sie haben für die Hardware den Befehl "Übersetzen" nicht durchgeführt. • Sie haben für die Hardware bzw. die Hardwarekonfiguration den Befehl "Laden in Gerät" nicht durchgeführt. • Das Objekt unterstützt keine Diagnose.
	Die Verbindung ist aufgebaut, aber der Zustand der Baugruppe wird momentan noch ermittelt bzw. ist unbekannt..
	Die projektierte Baugruppe unterstützt die Anzeige des Diagnosezustands nicht.
	Hardware-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Hardware-Komponente liegt ein Hardware-Fehler vor. (kommt als eigenständiges Symbol nur in der Projektnavigation vor)

Hinweis

Manche Baugruppen wie z. B. FM 450-1 werden beim Vorliegen eines Fehlers nur dann als fehlerhaft gekennzeichnet, wenn Sie bei der Parametrierung der Baugruppeneigenschaften den Diagnosealarm freigegeben haben.

Symbole für den Vergleichsstatus

Die Diagnose-Symbole können rechts unten mit kleineren Zusatzsymbolen kombiniert werden, die das Ergebnis des Online-/Offline-Vergleichs anzeigen. Die folgende Tabelle zeigt die möglichen Vergleichssymbole und ihre Bedeutung.

Symbol	Bedeutung
	Hardware-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Hardware-Komponente sind Online- und Offline-Version verschieden (nur in der Projektnavigation).
	Software-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Software-Komponente sind Online- und Offline-Version verschieden (nur in der Projektnavigation).
	Online- und Offline-Version des Objekts sind verschieden
	Objekt nur online vorhanden
	Objekt nur offline vorhanden
	Online- und Offline-Version des Objekts sind gleich

Hinweis

Wenn in der Gerätesicht sowohl ein Vergleichssymbol als auch das Diagnosesymbol " Fehler in unterlagerter Komponente " rechts unten eingeblendet werden sollen, gilt die folgende Regel: Das Diagnosesymbol für die unterlagerte Hardware-Komponente hat eine höhere Priorität als das Vergleichssymbol. Das hat zur Folge, dass ein Vergleichssymbol nur dann angezeigt wird, wenn in den unterlagerten Hardware-Komponenten keine Störung vorliegt.

Anzeige von Software-Fehlern in der Projektnavigation

- Hinter jedem Baustein wird das zugehörige Vergleichssymbol eingeblendet.
- Hinter jedem Ordner, unter dem sich ausschließlich Bausteine verbergen, wird das Diagnosesymbol "Software-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einem der zugehörigen Bausteine ein Software-Fehler vorliegt.
- Wenn bei einer Hardware-Komponente mit unterlagerten Software-Komponenten kein Hardware-Fehler vorliegt und in mindestens einer unterlagerten Software-Komponente ein Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Software-Fehler in unterlagerter Komponente" eingeblendet.

Kombinierte Diagnose- und Vergleichssymbole

Die folgende Tabelle zeigt Beispiele für Symbole, die in das Diagnosesymbol eingeblendet werden.

Symbol	Bedeutung
	Ordner enthält Objekte, deren Online- und Offline-Version verschieden sind (nur in der Projektnavigation)
	Objekt nur online vorhanden

Betriebszustandssymbole für CPUs und CPs

Die folgende Tabelle zeigt die möglichen Symbole und die zugehörigen Betriebszustände.

Symbol	Betriebszustand
	RUN
	STOP
	ANLAUF
	HALT
	DEFEKT
	Unbekannter Betriebszustand
	Die projektierte Baugruppe unterstützt die Anzeige des Betriebszustands nicht.

Hinweis

Falls auf einer CPU Forcen aktiv ist, wird in das Betriebszustandssymbol rechts unten ein rotes F auf rosafarbenem Grund eingeblendet.

Farbliche Kennzeichnung von Ports und Ethernet-Leitungen

Die folgende Tabelle zeigt die möglichen Farben und ihre jeweilige Bedeutung.

Farbe	Bedeutung
	Keine Störung oder Wartungsbedarf
	Offline
	Wartungsanforderung
	Kommunikation gestört oder topologischer Fehler
	nicht diagnosefähig

2.1.1.4 Online- und Diagnosesicht starten

Übersicht über die Möglichkeiten, die Online- und Diagnosesicht zu starten

Sie können die Online- und Diagnosesicht einer zu diagnostizierenden Baugruppe an folgenden Stellen starten:

- Übersicht
- Projektnavigation
- Gerätesicht
- Geräteübersicht
- Netzsicht
- Netzübersicht
- Topologiesicht
- Topologieübersicht

Im Folgenden wird anhand von Beispielen dargestellt, wie dabei vorzugehen ist.

Voraussetzung

Das Projekt mit der zu diagnostizierenden Baugruppe ist geöffnet.

Hinweis

Diese Voraussetzung ist nicht erforderlich, wenn Sie die Online- und Diagnosesicht aus der Projektnavigation heraus aufrufen, nachdem Sie die erreichbaren Teilnehmer ermittelt haben.

Vorgehen

Um die Online- und Diagnosesicht einer Baugruppe zu starten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den betroffenen Geräteordner.
2. Doppelklicken Sie "Online & Diagnose".

Oder:

1. Selektieren Sie in der Projektnavigation den betroffenen Geräteordner.
2. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie in der Projektnavigation den Ordner "Online-Zugänge".
2. Öffnen Sie den Ordner der Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
3. Doppelklicken Sie auf "Erreichbare Teilnehmer anzeigen / aktualisieren".

4. Selektieren Sie die zu diagnostizierende Baugruppe.
5. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie in der Projektnavigation den Ordner "Lokale Baugruppen".
2. Selektieren Sie das betroffene Gerät bzw. die zu diagnostizierende Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie die Gerätesicht der Gerätekonfiguration.
2. Selektieren Sie die zu diagnostizierende Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie die Gerätesicht der Gerätekonfiguration.
2. Stellen Sie eine Online-Verbindung mit der zu diagnostizierenden Baugruppe her.
3. Doppelklicken Sie auf das Diagnosesymbol oberhalb der Baugruppe.

Oder:

1. Öffnen Sie die Netzsicht der Gerätekonfiguration.
2. Selektieren Sie die Station mit der zu diagnostizierenden Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie die Topologiesicht der Gerätekonfiguration.
2. Stellen Sie eine Online-Verbindung mit der zu diagnostizierenden Baugruppe her.
3. Doppelklicken Sie in der Topologieübersicht auf das zur Baugruppe gehörende Diagnosesymbol.

Ergebnis

Die Online- und Diagnosesicht der zu diagnostizierenden Baugruppe wird gestartet. Falls vorher eine Online-Verbindung zur zugehörigen CPU bestand, wird die Titelzeile der Online- und Diagnosesicht mit der Online-Farbe orange hinterlegt.

Hinweis

Falls beim Start der Online- und Diagnosesicht keine Online-Verbindung besteht, können keine Online-Informationen angezeigt werden und die entsprechenden Anzeigefelder bleiben leer.

2.1.1.5 Aktivieren der Task Card "Online-Tools"

Aktivieren der Task Card "Online-Tools"

Sie können diese Task Card wie folgt aktivieren:

1. Starten Sie die Online- und Diagnosesicht.
2. Klicken sie auf die Task Card "Online-Tools".

Oder:

1. Starten Sie die Gerätesicht.
2. Klicken sie auf die Task Card "Online-Tools".

Oder:

1. Starten Sie die Netzsicht.
2. Klicken sie auf die Task Card "Online-Tools".

2.1.2 Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen

2.1.2.1 Allgemeine Eigenschaften und anlagenrelevante Informationen einer Baugruppe anzeigen

Wo befinden sich die gesuchten Informationen?

Die allgemeinen Eigenschaften und anlagenrelevanten Informationen einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Allgemein".

Aufbau der Gruppe "Allgemein"

Die Gruppe "Allgemein" besteht aus folgenden Bereichen:

- Modul
- Baugruppeninformation
- Herstellerinformation

Bereich "Modul"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Kurzbezeichnung, z. B. CPU 1214C DC/DC/DC
- Artikel-Nr.

- Hardware
- Firmware
- TIA Portal-Version

Hinweis

TIA Portal-Version

Die angezeigte TIA Portal-Version ist diejenige Version, die mindestens erforderlich ist, um die in das Gerät geladenen Bausteine öffnen oder in das ES laden zu können. Sie muss also nicht unbedingt mit derjenigen Version übereinstimmen, mit der Sie die Bausteine in das Gerät geladen haben.

Fiktives Beispiel: Sie haben auf Ihrem ES das TIA Portal in der Version V14.0 SP1 installiert und legen ein neues Projekt an. Dieses Projekt erhält die Projektversion V14.0. Sie laden dieses Projekt (und mit ihm seine Projektversion) in die CPU. Angezeigt wird die Projektversion, also V14.0.

- Baugruppenträger
- Steckplatz

Bereich "Baugruppeninformation"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an, die Sie bei der Konfiguration der Hardware projektiert haben:

- Baugruppenname
- Einbaudatum (wird nicht bei allen Baugruppen angezeigt)
- Zusatzinformation (wird nicht bei allen Baugruppen angezeigt)

Bereich "Herstellerinformation"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Hersteller
- Seriennummer
- Profil: Profil-ID als Hexadezimalzahl

Hinweis

Den zugehörigen Profilnamen entnehmen Sie der Profile ID Table bei PROFIBUS International (siehe "www.profibus.com".).

- Profildetails: Profilspezifischer Typ als Hexadezimalzahl

Hinweis

Den zugehörigen profilspezifischen Typnamen entnehmen Sie der Profile Specific Type Table bei PROFIBUS International (siehe "www.profibus.com".).

2.1.2.2 Parametrierte Zykluszeiten anzeigen

Wo befinden sich die gesuchten Informationen?

Die gesuchten Informationen finden Sie an den folgenden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Zykluszeit"
- In der Task Card "Online-Tools" in der Palette "Zykluszeit"

Aufbau der Gruppe "Zykluszeit" im Ordner "Diagnose" der Online- und Diagnosesicht

Die Gruppe "Zykluszeit" besteht aus folgenden Bereichen:

- Zykluszeit-Diagramm (Grafische Anzeige der parametrierten und der gemessenen Zykluszeiten)
- Zykluszeit parametriert (Anzeige der parametrierten Zykluszeiten als Absolutwerte)
- Zykluszeiten gemessen (Anzeige der gemessenen Zykluszeiten als Absolutwerte)

Aufbau der Palette "Zykluszeit" der Task Card "Online-Tools"

Die Palette "Zykluszeit" zeigt das Zykluszeit-Diagramm und darunter die gemessenen Zykluszeiten als Absolutwerte.

Parametrierte Zykluszeiten

Folgende parametrierten Zykluszeiten werden im Zykluszeit-Diagramm und im Bereich "Zykluszeit parametriert" dargestellt:

- Mindestzykluszeit
- Zyklusüberwachungszeit

Im Zykluszeit-Diagramm entsprechen die Mindestzykluszeit und die Zyklusüberwachungszeit den beiden Markierungen auf der Zeitachse.

Im Bereich "Zykluszeit parametriert" werden die parametrierten Zykluszeiten als Absolutwerte dargestellt.

2.1.2.3 Schnittstellen und Schnittstelleneigenschaften einer Baugruppe anzeigen

Wo befinden sich die gesuchten Informationen?

Die Schnittstellen und Schnittstelleneigenschaften einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der folgenden Gruppe:

- PROFINET-Schnittstelle

Gruppe "PROFINET-Schnittstelle"

Diese Gruppe gliedert sich in folgende Bereiche:

- "Ethernet-Adresse" mit den Unterbereichen "Netzwerkverbindung" und "IP-Parameter"
- "Ports"

Unterbereich "Netzwerkverbindung" des Bereichs "Ethernet-Adresse"

Dieser Unterbereich zeigt die folgenden Daten der Baugruppe an:

- MAC-Adresse:
  MAC-Adresse der Schnittstelle.
  Die MAC-Adresse besteht aus zwei Teilen. Der erste Teil ("Basis-MAC-Adresse") kennzeichnet den Hersteller (Siemens, 3COM, ...). Der zweite Teil der MAC-Adresse unterscheidet die verschiedenen Ethernet-Teilnehmer. Jeder Ethernet-Baugruppe ist eine eindeutige MAC-Adresse zugeordnet.

Unterbereich "IP-Parameter" des Bereichs "Ethernet-Adresse"

Dieser Unterbereich zeigt die folgenden Daten der Baugruppe an:

- IP-Adresse:
  Internet-Protokolladresse des Teilnehmers am Bus (TCP/IP)
- Subnetzmaske:
  Die Subnetzmaske gibt an, welcher Teil der IP-Adresse die Zugehörigkeit zu einem bestimmten Subnetz bestimmt.
- Default-Router:
  Falls das Subnetz über Router an andere Subnetze angeschlossen ist, muss die IP-Adresse des Standard-Routers bekannt sein. Nur so kann ein Datagramm mit einer nicht übereinstimmenden Subnetzadresse weitergeleitet werden.
- IP-Einstellungen:
  Kennung für den Weg, über den der Teilnehmer seine IP-Einstellungen (IP-Adresse, Subnetzmaske, Default-Router) bekommen hat.

Kennung	Bedeutung
0	IP-Adresse ist nicht initialisiert
1	Über Projektierung (d. h. über die Konfiguration, die aus der Geräte- oder Netzsicht in den Teilnehmer geladen wurde)
2	Über die Gruppe "IP-Adresse zuweisen" der Online- und Diagnosesicht
3	Über DHCP-Server (d. h. die IP-Parameter wurden über einen speziellen Dienst von einem DHCP-Server (Dynamic Host Configuration Protocol) bezogen und für begrenzte Zeit zugewiesen)
4	IP-Adresse von einem Benutzerprogramm eingestellt
5	Quelle der IP-Adresse unbekannt

- IP-Einstellungszeit:
  Zeitstempel der letzten Änderung der IP-Adresse direkt über den Ethernetanschluss der Baugruppe

Bereich "Ports"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Ethernet-Ports
  Physikalische Eigenschaften der PROFINET-Schnittstelle

Eigenschaft der PROFI-NET-Schnittstelle	Bedeutung
Port-Nr.	Port-Nummer. In Klammern ist die Kurzbezeichnung von Schnittstelle (X + Nr. der Schnittstelle) und Port (P + Nr. des Ports) angegeben. Ein "R" in der Kurzbezeichnung eines Ports bedeutet, dass es sich um einen Ringport handelt.
Status	Anzeige des Zustands der zum Port gehörigen LINK-LED: • Bei "OK" ist ein anderes Gerät (z. B. ein Switch) an den Port angeschlossen, und die physikalische Verbindung steht. • Bei "Getrennt" ist kein weiteres Gerät an den Port angeschlossen. • Bei "Abgeschaltet" ist der Zugang zum Port unterbunden.
Einstellungen	• "automatisch" bei automatischen Netzwerkeinstellungen des Teilnehmers • Netzwerkeinstellungen hinsichtlich der Geschwindigkeit und des Übertragungsverfahrens bei manuellen Netzwerkeinstellungen des Teilnehmers
Betriebsart	Netzwerkeinstellungen hinsichtlich der Geschwindigkeit und des Übertragungsverfahrens

Wenn Sie in der Port-Tabelle eine Zeile markieren, dann werden zum zugehörigen Port zusätzliche Hilfeinformationen bereitgestellt.

2.1.2.4 IO-Controller anzeigen, die auf Module eines Shared Device zugreifen

Wo befinden sich die gesuchten Informationen?

Die Anzeige derjenigen IO-Controller, die auf Module eines Shared Device zugreifen, finden Sie in der Online- und Diagnosesicht des Interfacemoduls des Shared Device im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- IO-Controller

Siehe auch

Wissenswertes zu Shared Devices (Seite 3116)

2.1.2.5 Sync-Domain-Eigenschaften eines PROFINET-Geräts anzeigen

Wo befinden sich die gesuchten Informationen?

Die Sync-Domain-Eigenschaften eines PROFINET-Geräts finden Sie in der Online- und Diagnosesicht des zu diagnostizierenden Geräts im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- Domain

Bereich "Domain"

Dieser Bereich gliedert sich in die folgenden Unterbereiche:

- Sync-Domain
- MRP-Domain

Was ist eine Sync-Domain?

Eine Sync-Domain ist eine Gruppe von PROFINET-Geräten, die auf einen gemeinsamen Takt synchronisiert sind. Genau ein Gerät hat die Rolle des Sync-Masters (Taktgeber), alle anderen Geräte nehmen die Rolle eines Sync-Slaves ein. Der Sync-Master ist meist ein IO-Controller oder ein Switch.

Nicht synchronisierte PROFINET-Geräte sind nicht Teil einer Sync-Domain.

Unterbereich "Sync-Domain" des Bereichs "Domain"

Dieser Unterbereich zeigt die folgenden Eigenschaften der Sync-Domain an:

- Name:
  Name der Sync-Domain
- Rolle:
  Rolle des PROFINET-Geräts in der Sync-Domain. Folgende Rollen sind möglich:
  - Sync-Master
  - Sync-Slave
- Synchronisationsintervall:
  Intervall, in dem die Synchronisation durchgeführt wird
- Sendetakt
  kleinstmögliches Sende-Intervall für den Datenaustausch
- Jittergenauigkeit des Sendetakts
- Reservierte Bandbreite für zyklische Kommunikation

2.1.2.6 MRP-Domain-Eigenschaften eines PROFINET-Geräts anzeigen

Wo befinden sich die gesuchten Informationen?

Die MRP-Domain-Eigenschaften eines PROFINET-Geräts finden Sie in der Online- und Diagnosesicht des zu diagnostizierenden Geräts im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- Domain

Bereich "Domain"

Dieser Bereich gliedert sich in die folgenden Unterbereiche:

- Sync-Domain
- MRP-Domain

Was ist eine MRP-Domain?

Über das sogenannnte Media Redundancy Protocol (MRP) ist es möglich, redundante Netze aufzubauen. Redundante Übertragungsstrecken (Ringtopologie) sorgen dafür, dass bei Ausfall einer Übertragungsstrecke ein alternativer Kommunikationsweg zur Verfügung steht. Diejenigen PROFINET-Geräte, die Teil dieses redundanten Netzes sind, bilden eine MRP-Domain.

Unterbereich "MRP-Domain" des Bereichs "Domain"

Dieser Unterbereich zeigt die folgenden Eigenschaften der MRP-Domain an:

- Name:
  Name der MRP-Domain
- Rolle:
  Rolle des PROFINET-Geräts in der MRP-Domain. Folgende Rollen sind möglich:
  - Manager
  - Manager (Auto)
  - Client
  - Nicht Teilnehmer des Rings
- Ringport 1:
  derjenige Port des PROFINET-Geräts, der die Eigenschaft "Ringport 1" hat
- Ringport 2:
  derjenige Port des PROFINET-Geräts, der die Eigenschaft "Ringport 2" hat
- Zustand des MRP-Rings:
  Hier wird angezeigt, ob der Ring unterbrochen ist (Zustand "offen") oder nicht (Zustand "geschlossen").

2.1.2.7 Aktuelle Firmware einer Baugruppe anzeigen

Firmware anzeigen

Sie können sich die aktuell installierte Firmware einer Baugruppe anzeigen lassen.

Voraussetzungen

- Die Baugruppe unterstützt ein Firmware-Update.
- Die Baugruppe ist online verbunden.

Vorgehen

Um die aktuelle Firmware anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Baugruppe in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".
3. Die aktuelle Firmware lesen Sie im Bereich "Online-Daten" unter "Firmware" ab.

2.1.3 Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen

2.1.3.1 Gemessene Zykluszeiten anzeigen

Wo befinden sich die gesuchten Informationen?

Die gemessenen Zykluszeiten finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Zykluszeit"
- In der Task Card "Online-Tools" in der Palette "Zykluszeit"

Aufbau der Gruppe "Zykluszeit" im Ordner "Diagnose" der Online- und Diagnosesicht

Die Gruppe "Zykluszeit" besteht aus folgenden Bereichen:

- Zykluszeit-Diagramm (Grafische Anzeige der parametrierten und der gemessenen Zykluszeiten)
- Zykluszeit parametriert (Anzeige der parametrierten Zykluszeiten als Absolutwerte)
- Zykluszeiten gemessen (Anzeige der gemessenen Zykluszeiten als Absolutwerte)

Aufbau der Palette "Zykluszeit" der Task Card "Online-Tools"

Die Palette "Zykluszeit" zeigt das Zykluszeit-Diagramm und darunter die gemessenen Zykluszeiten als Absolutwerte.

Grafische Darstellung der gemessenen Zykluszeiten

Folgende gemessenen Zykluszeiten werden im Zykluszeit-Diagramm dargestellt:

- Kürzeste Zykluszeit: Dauer des kürzesten Zyklus seit dem letzten Übergang von STOP nach RUN
  Im Diagramm entspricht das dem linken gestrichelten grauen Pfeil.
- Aktuelle / letzte Zykluszeit: Dauer des letzten Zyklus
  Im Diagramm entspricht das dem grünen Pfeil. Wenn die aktuelle / letzte Zykluszeit die Zyklusüberwachungszeit überschreitet, wird der Pfeil rot.

  Hinweis

  Wenn die Dauer des letzten Zyklus nahe an die Zyklusüberwachungszeit heranreicht, ist es möglich, dass diese überschritten wird. Abhängig vom CPU-Typ, Ihrer Parametrierung und Ihrem Anwenderprogramm kann die CPU in den Betriebszustand STOP gehen. Wenn Sie z. B. Variablen Ihres Programms beobachten, verlängern Sie damit die Zykluszeit.

  Wenn der Zyklus länger als die doppelte Zyklusüberwachungszeit dauert und Sie im Anwenderprogramm die Zyklusüberwachungszeit nicht neu starten (durch Aufruf der erweiterten Anweisung RE_TRIGR), geht die CPU in STOP.

- Längste Zykluszeit: Dauer des längsten Zyklus seit dem letzten Übergang von STOP nach RUN
  Im Diagramm entspricht das dem rechten gestrichelten blauen Pfeil.

Zwischen den beiden gestrichelten Pfeilen erstreckt sich ein blauer Streifen, der dem gesamten Bereich der gemessenen Zykluszeiten entspricht. Wenn eine gemessene Zykluszeit größer als die Zyklusüberwachungszeit ist, wird derjenige Teil des Streifens, der außerhalb der parametrierten Grenzen liegt, rot eingefärbt.

Darstellung der gemessenen Zykluszeiten als Absolutwerte

Folgende gemessenen Zeiten werden im Bereich "Zykluszeiten gemessen" und in der Palette "Zykluszeit" angezeigt:

- Kürzeste Zykluszeit seit dem letzten Übergang von STOP nach RUN
- Aktuelle/letzte Zykluszeit
- Längste Zykluszeit seit dem letzten Übergang von STOP nach RUN

2.1.3.2 Aktuellen Zustand der LEDs einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Den aktuellen Zustand der LEDs einer CPU finden Sie in der Task Card "Online-Tools" im Anzeigebereich der Palette "CPU-Bedienpanel".

Anzeigebereich der Palette "CPU-Bedienpanel" in der Task Card "Online-Tools"

Dieser Bereich enthält die folgenden Anzeigen:

- Stationsname und CPU-Typ (Kurzbezeichnung)
- RUN / STOP (entspricht der LED "RUN / STOP" der CPU)
- ERROR (entspricht der LED "ERROR" der CPU)
- MAINT (entspricht der LED "MAINT" der CPU)

2.1.3.3 Füllstand aller Speicherarten einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Den Füllstand aller Speicherarten einer CPU finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" im Anzeigebereich der Gruppe "Speicher"
- In der Task Card "Online-Tools" im Anzeigebereich der Palette "Speicher"

Anzeigebereich der Gruppe "Speicher" im Ordner "Diagnose" der Online- und Diagnosesicht

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe und Details der einzelnen Speicherbereiche.

Die Speicherauslastung wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher
  Falls keine Memory Card steckt, wird der interne Ladespeicher angezeigt.
  Falls eine Memory Card steckt, benutzt das Betriebssystem als Ladespeicher ausschließlich den gesteckten Ladespeicher. Dieser wird hier angezeigt.
- Arbeitsspeicher
- Remanenzspeicher

Anzeigebereich der Palette "Speicher" in der Task Card "Online-Tools"

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe. Der freie Speicher wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt. Der Zahlenwert ist auf ganzzahlige Werte gerundet.

Hinweis

Wenn ein Speicherbereich weniger als 1 % belegt ist, wird beim freien Teil dieses Speicherbereichs "99 %" angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher
  Falls keine Memory Card steckt, wird der interne Ladespeicher angezeigt.
  Falls eine Memory Card steckt, benutzt das Betriebssystem als Ladespeicher ausschließlich den gesteckten Ladespeicher. Dieser wird hier angezeigt.
- Arbeitsspeicher
- Remanenzspeicher

Siehe auch

Ladespeicher (Seite 1373)

Arbeitsspeicher (Seite 1374)

Remanente Speicherbereiche (Seite 1376)

2.1.3.4 Füllstand aller Speicherarten einer S7-1500-CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Den Füllstand aller Speicherarten einer S7-1500-CPU finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" im Anzeigebereich der Gruppe "Speicher"
- In der Task Card "Online-Tools" im Anzeigebereich der Palette "Speicher"

Anzeigebereich der Gruppe "Speicher" im Ordner "Diagnose" der Online- und Diagnosesicht

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe und Details der einzelnen Speicherbereiche.

Die Speicherauslastung wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher

  Hinweis

  Der Ladespeicher befindet sich auf der SIMATIC Memory Card.

- Code-Arbeitsspeicher: Arbeitsspeicher für Programmcode
- Daten-Arbeitsspeicher: Arbeitsspeicher für Datenbausteine
- Remanenzspeicher

Anzeigebereich der Palette "Speicher" in der Task Card "Online-Tools"

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe. Der freie Speicher wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt. Der Zahlenwert ist auf ganzzahlige Werte gerundet.

Hinweis

Wenn ein Speicherbereich weniger als 1 % belegt ist, wird beim freien Teil dieses Speicherbereichs "99 %" angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher

  Hinweis

  Der Ladespeicher befindet sich auf der SIMATIC Memory Card.

- Code-Arbeitsspeicher: Arbeitsspeicher für Programmcode
- Daten-Arbeitsspeicher: Arbeitsspeicher für Datenbausteine
- Remanenzspeicher

2.1.4 Eine Baugruppe auf Störungen überprüfen

2.1.4.1 Diagnosestatus einer Baugruppe ermitteln

Wo wird der Diagnosestatus einer Baugruppe angezeigt?

Der Diagnosestatus einer Baugruppe wird in der Online- und Diagnosesicht der Baugruppe im Ordner "Diagnose" in der Gruppe "Diagnosestatus" angezeigt.

Die Gruppe "Diagnosestatus" besteht aus folgenden Bereichen:

- Status
- Standarddiagnose (bei S7-300 und S7-400 nur bei Nicht-CPU-Baugruppen)

Bereich "Status"

In diesem Bereich werden die folgenden Statusinformationen angezeigt:

- Status der Baugruppe aus Sicht der CPU, z. B.:
  - Baugruppe vorhanden und OK.
  - Baugruppe gestört.
    Falls die Baugruppe eine Störung hat und Sie bei der Projektierung den Diagnosealarm freigegeben haben, wird der Status "Baugruppe gestört" angezeigt.
  - Baugruppe projektiert, aber nicht vorhanden.
    Bsp.: Es sind keine Diagnosedaten verfügbar, weil die aktuelle Online-Konfiguration sich von der Offline-Konfiguration unterscheidet.
- Erkannte Unterschiede zwischen projektierter und gesteckter Baugruppe. Soweit ermittelbar wird für Soll- und Isttyp die Artikelnummer angezeigt.

Der Umfang der angezeigten Informationen ist abhängig von der selektierten Baugruppe.

Bereich "Standarddiagnose"

In diesem Bereich werden die folgenden Diagnoseinformationen von Nicht-CPU-Baugruppen angezeigt:

- Interne und externe Störungen, die sich auf die gesamte Baugruppe beziehen
- Zugehörige Diagnoseereignisse

Beispiele für solche Diagnoseinformationen sind:

- Gesamte Pufferung ausgefallen
- Baugruppe gestört

Hinweis

Diagnosealarme

Ein Diagnosealarm kann nur dann zur CPU gemeldet werden, wenn die Baugruppe diagnosealarmfähig ist und Sie den Diagnosealarm freigegeben haben.

Die Anzeige des Diagnosealarms ist eine Momentaufnahme. Sporadische Störungen einer Baugruppe erkennen Sie im Diagnosepuffer der zugehörigen CPU.

2.1.4.2 Kommunikationsdiagnose von PROFINET-Ports durchführen

Wo werden portspezifische Fehler von PROFINET-Ports angezeigt?

Die Anzeige von portspezifischen Fehlern von PROFINET-Ports erfolgt in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" unter "PROFINET-Schnittstelle" in der Gruppe "Kommunikationsdiagnose".

Aufbau der Gruppe "Kommunikationsdiagnose"

Die Gruppe "Kommunikationsdiagnose" besteht aus folgenden Bereichen:

- Tabelle mit den portspezifischen Fehlern
- Textfeld "Details zum Ereignis"
- Textfeld "Hilfe zum Ereignis"

Aufbau der Tabelle mit den portspezifischen Fehlern

Jede Zeile der Fehlertabelle entspricht einem Fehler.

Die Tabelle besteht aus den folgenden Spalten:

- Name: Port-Nr. und Portbezeichnung sowie Diagnosesymbol
- Fehler: Beschreibung des bei diesem Port aufgetretenen Fehlers

Textfeld "Details"

Wenn Sie in der Fehlertabelle eine Zeile markieren, enthält das Textfeld "Details" Detailinformationen zum zugehörigen Fehler (sofern vorhanden).

Textfeld "Hilfe zur markierten Diagnosezeile"

Wenn Sie in der Fehlertabelle eine Zeile markieren, enthält das Textfeld "Hilfe zur markierten Diagnosezeile" Hilfeinformationen zum zugehörigen Fehler (sofern vorhanden).

2.1.4.3 PROFINET IO-Komponenten diagnostizieren

Wo diagnostizieren Sie PROFINET IO-Komponenten?

Die Diagnose von PROFINET IO-Komponenten erfolgt in deren Online- und Diagnosesicht im Ordner "Diagnose" unter "PROFINET-Schnittstelle" in der Gruppe "PROFINET IO-Diagnose".

Gruppe "PROFINET IO-Diagnose"

In diesem Bereich werden herstellerbezogene Diagnosetexte für das IO-Device bzw. für das selektierte Modul des IO-Devices angezeigt.

Wenn vom IO-Controller die Diagnose gelesen werden kann, werden hier Kommunikationsfehler und Projektierungsfehler (Unterschiede online/offline) angezeigt.

2.1.4.4 Den Diagnosepuffer einer CPU auslesen

Wo lesen Sie den Diagnosepuffer einer CPU aus?

Sie lesen den Diagnosepuffer einer CPU in deren Online- und Diagnosesicht im Ordner "Diagnose" in der Gruppe "Diagnosepuffer" aus.

Aufbau der Gruppe "Diagnosepuffer"

Die Gruppe "Diagnosepuffer" besteht aus folgenden Bereichen:

- "Ereignisse"
- "Einstellungen"

Diagnosepuffer

Der Diagnosepuffer dient als Log-Datei für die auf der CPU und den ihr zugeordneten Baugruppen aufgetretenen Diagnoseereignisse. Diese werden in der Reihenfolge ihres Auftretens eingetragen, wobei das jüngste Ereignis an oberster Stelle dargestellt wird.

Bereich "Ereignisse"

Der Bereich "Ereignisse" besteht aus folgenden Elementen:

- Optionskästchen "CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit"
- Ereignistabelle
- Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"
- Details zum Ereignis: Ereignis-Nr., Ereignis-ID, Modul und ggf. Stations- bzw. Gerätename, Rack/Steckplatz, Beschreibung, Hilfe zum Ereignis, Anlagenkennzeichen, Ortskennzeichen, Kommend/Gehend-Information, Ereignistyp
- Schaltflächen "Im Editor öffnen", "Speichern unter ..."

Optionskästchen " CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit "

Wenn Sie das Optionskästchen nicht aktiviert haben, werden die Diagnosepuffereinträge mit der Baugruppenzeit angezeigt.

Wenn Sie das Optionskästchen aktiviert haben, werden die Diagnosepuffereinträge mit derjenigen Uhrzeit angezeigt, die sich aus folgender Formel ergibt:

Angezeigte Uhrzeit = Baugruppenzeit + Zeitzonen-Offset Ihres PG / PC

Dabei wird vorausgesetzt, dass die Baugruppenzeit identisch mit UTC ist.

Sie verwenden diese Einstellung, wenn sie bei den Diagnosepuffereinträgen der Baugruppe als Uhrzeit die Lokalzeit Ihres PG / PC sehen wollen.

Wenn Sie das Optionskästchen aktivieren bzw. deaktivieren, werden die Zeitangaben der Diagnosepuffereinträge sofort angepasst.

Hinweis

Wenn Sie in Ihrem Programm die Anweisung "WR_SYS_T" einsetzen oder die Echtzeituhr der CPU über ein Bediengerät stellen und dabei nicht UTC verwenden, wird empfohlen, das Optionskästchen " CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit " zu deaktivieren. In diesem Fall orientieren Sie sich nur an der Baugruppenzeit.

Ereignistabelle

In der Tabelle werden zu jedem Diagnoseereignis folgende Informationen angezeigt:

- Laufende Nummer des Eintrags
  Der erste Eintrag enthält das neueste Ereignis.
- Datum und Uhrzeit des Diagnoseereignisses
  Werden kein Datum und keine Uhrzeit ausgegeben, besitzt die Baugruppe keine integrierte Uhr.
- Kurzbezeichnung des Ereignisses und ggf. die Reaktion der CPU

  Hinweis

  Wenn ein einzelner Parameter eines Textes nicht ermittelt werden kann, wird an dieser Stelle die Zeichenfolge "###" ausgegeben.

  Wenn für neue Baugruppen oder neue Ereignisse noch kein Anzeigetext vorhanden ist, werden die Nummern der Ereignisse und die einzelnen Parameter als Hexadezimalwerte ausgegeben.

- Nur bei S7-1200- und S7-1500-CPUs: Symbol für den Ereignistyp
  Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

Symbol	Bedeutung
	OK (keine Wartung bzw. keine Störung)
	Wartungsbedarf
	Wartungsanforderung
	Fehler

- Symbol für die Kommend-/Gehend-Information
  Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

Symbol	Bedeutung
	Kommendes Ereignis
	Gehendes Ereignis

Symbol	Bedeutung
	Kommendes Ereignis, zu dem es kein eigenständiges gehendes Ereignis gibt
	Anwenderdefiniertes Diagnoseereignis

Sie können in der Ereignistabelle die Spaltenreihenfolge ändern, die Spaltenbreiten ändern und einzelne Spalten entfernen und wieder hinzufügen. Darüber hinaus können Sie wie folgt sortieren: nach der laufenden Nummer, nach "Datum und Uhrzeit" und nach "Ereignis".

Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

Die Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben" ist nur aktiv, wenn eine Online-Verbindung zur CPU besteht.

Die Voreinstellung ist "Anzeige einfrieren".

Wenn Sie auf die Schaltfläche "Anzeige einfrieren" klicken, geschieht Folgendes:

- Die aktuelle Anzeige der Diagnosepuffereinträge wird eingefroren.
- Die Beschriftung der Schaltfläche wechselt zu " Einfrieren aufheben".

Wenn ein Fehler in Ihrer Anlage aufgetreten ist, kann es zu schnell aufeinanderfolgenden Diagnoseereignissen kommen. Daraus resultiert eine hohe Aktualisierungsrate der Anzeige. Das Einfrieren der Anzeige bietet Ihnen in diesem Fall die Möglichkeit, den Sachverhalt in Ruhe näher zu untersuchen.

Wenn die Anzeige eingefroren ist und Sie auf die Schaltfläche "Einfrieren aufheben" klicken, geschieht Folgendes:

- Die Anzeige der Diagnosepuffereinträge wird wieder aktualisiert.
- Die Beschriftung der Schaltfläche wechselt zu "Anzeige einfrieren".

Hinweis

Beim Einfrieren der Diagnosepufferanzeige trägt die CPU weiterhin Ereignisse in den Diagnosepuffer ein.

Details zum Ereignis

Wenn Sie in der Liste der Ereignisse eine Zeile selektieren, erhalten Sie hier Detailinformationen zum zugehörigen Ereignis:

- Laufende Nummer des Ereignisses im Diagnosepuffer
- Ereignis-ID
- Modul und ggf. Stations- bzw. Gerätename
- Rack / Steckplatz

- Beschreibung des Ereignisses mit ereignisabhängigen Zusatzinformationen. Beispiele für diese Zusatzinformationen sind:
  - Befehl, der das Ereignis verursacht hat
  - Betriebszustandsübergang, der durch das Diagnoseereignis verursacht wurde
- Hilfe zum Ereignis: Das selektierte Ereignis wird näher erläutert und es werden ggf. Abhilfen genannt. Bei gehenden Ereignissen wird der Text "Gehendes Ereignis: Keine Benutzeraktion erforderlich" angezeigt.
- Nur bei S7-1200- und S7-1500-CPUs: Anlagenkennzeichen, Ortskennzeichen
- Information, ob es sich um ein kommendes oder ein gehendes Ereignis handelt
- Ereignistyp. Folgende Ereignistypen sind möglich:
  - OK (keine Wartung bzw. keine Störung)
  - Wartungsbedarf
  - Wartungsanforderung
  - Fehler

Schaltfläche "Im Editor öffnen"

Die folgende Tabelle zeigt, wann die Schaltfläche "Öffne Baustein" aktiv ist und welche Funktion sich dahinter verbirgt.

Wann ist die Schaltfläche "Im Editor öffnen" aktiv?	Was bewirkt ein Klicken auf diese Schaltfläche?
Wenn im Diagnoseereignis auf eine Relativadresse eines Bausteins verwiesen wird. Dies ist die Adresse des Ereignis verursachenden Befehls.	Die Funktion "Im Editor öffnen" öffnet den referenzierten Baustein in der Offline-Ansicht an der Fehler verursachenden Programmieranweisung. Dadurch können Sie den Quellcode des Bausteins an der angegebenen Stelle überprüfen, ggf. ändern und anschließend wieder in die CPU laden.
Wenn das Diagnoseereignis von einem Modul ausgelöst wurde.	Die Funktion "Im Editor öffnen" öffnet die Gerätesicht des betroffenen Moduls.

Schaltfläche "Speichern unter ..."

Wenn Sie auf diese Schaltfläche klicken, wird der Inhalt des Diagnosepuffers in einer Textdatei gespeichert. Als Dateiname wird sprachabhängig "Diagnose" mit der Erweiterung ".txt" vorgeschlagen. Sie können diesen Namen jedoch ändern.

Bereich "Einstellungen"

Der Bereich "Einstellungen" besteht aus folgenden Elementen:

- Liste "Ereignisse anzeigen"
- Schaltfläche "Einstelllungen als Standard übernehmen"
- Optionskästchen "Ereignisinformation hexadezimal ausgeben"

Liste "Ereignisse anzeigen:"

In dieser Liste gibt es zu jeder Ereignisklasse ein Optionskästchen (Voreinstellung: Alle Optionskästchen sind aktiviert.). Wenn Sie ein Optionskästchen deaktivieren, werden die Ereignisse der zugehörigen Ereignisklasse im Bereich "Ereignisse" nicht mehr angezeigt. Wenn Sie das Optionskästchen wieder aktivieren, werden die zugehörigen Ereignisse wieder angezeigt.

Schaltfläche "Einstellungen als Standard übernehmen"

Wenn Sie auf diese Schaltfläche klicken, gelten die Einstellungen auch bei künftigen Aufrufen des Registers "Ereignisse".

Optionskästchen "Ereignisinformation hexadezimal ausgeben"

Wenn Sie das Optionskästchen aktivieren, wird in der Ereignisliste des Bereichs "Ereignisse" die Ereignis-ID hexadezimal angegeben. Wenn Sie das Optionskästchen deaktivieren, wird die Ereignisinformation in Textform angegeben.

Siehe auch

Grundlagen zum Diagnosepuffer (Seite 3527)

2.1.4.5 Servicedaten speichern

Zweck

Im Servicefall kann es notwendig sein, dass der Customer Support von SIEMENS zu Diagnosezwecken sehr spezielle Informationen über den Zustand einer Baugruppe Ihrer Anlage benötigt.

Wenn ein solcher Fall in Ihrer Anlage auftritt, werden Sie vom Customer Support gebeten, die Servicedaten der Baugruppe zu speichern und ihm die erstellte Datei zukommen zu lassen.

Wo führen Sie das Speichern von Servicedaten einer Baugruppe durch?

Das Speichern von Servicedaten einer Baugruppe führen Sie in deren Online- und Diagnosesicht an der folgenden Stelle durch: im Ordner "Funktionen" in der Gruppe "Servicedaten speichern"

Die Gruppe "Servicedaten speichern" besteht aus folgenden Bereichen:

- Online-Daten
- Servicedaten speichern

Bereich "Online-Daten"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Artikelnummer
- Firmware-Stand
- Baugruppenname (Diesen haben Sie bei der Konfiguration der Hardware projektiert.)
- Baugruppenträger
- Steckplatz

Bereich "Servicedaten speichern"

Um eine Datei mit speziellen Servicedaten zu erstellen und zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie die Stelle im Dateisystem, an der Sie die Datei speichern wollen:
   - Sie benutzen den im Feld "Pfad" voreingestellten Pfad.
   - Sie klicken auf die Schaltfläche mit den drei Punkten. Im sich öffnenden Dialog geben Sie den gewünschten Pfad vor und legen den Dateinamen fest.
2. Klicken Sie auf die Schaltfläche "Daten speichern".

2.1.5 Eigenschaften einer Baugruppe bzw. des PG/PC ändern

2.1.5.1 Den Betriebszustand einer CPU umschalten

Voraussetzung

Es besteht eine Online-Verbindung zur CPU, deren Betriebszustand Sie umschalten wollen.

Vorgehen

Um den Betriebszustand einer CPU umzuschalten, gehen Sie wie folgt vor:

1. Aktivieren Sie die Task Card "Online Tools" der CPU.
2. Klicken Sie in der Palette "CPU-Bedienpanel" auf die Schaltfläche "RUN", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. auf die Schaltfläche "STOP", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

   Hinweis

   Es ist nur diejenige Schaltfläche aktiv, die Sie im aktuellen Betriebszustand der CPU anwählen können.

3. Beantworten Sie die Sicherheitsabfrage mit "OK".

Oder:

1. Öffnen Sie das Menü "Online".
2. Wählen Sie den Menübefehl "CPU starten", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. "CPU stoppen", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

   Hinweis

   Es ist nur derjenige Menübefehl aktiv, den Sie im aktuellen Betriebszustand der CPU wählen können.

3. Beantworten Sie die Sicherheitsabfrage mit "OK".

Oder:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "CPU starten", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. "CPU stoppen", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

   Hinweis

   Es ist nur diejenige Schaltfläche aktiv, die Sie im aktuellen Betriebszustand der CPU wählen können.

2. Beantworten Sie die Sicherheitsabfrage mit "OK".

Ergebnis

Die CPU wird in den gewünschten Betriebszustand umgeschaltet.

2.1.5.2 Urlöschen durchführen

Voraussetzung

- Es besteht eine Online-Verbindung zur CPU, auf der Urlöschen durchgeführt werden soll.
- Diese CPU befindet sich im Betriebszustand STOP.

  Hinweis

  Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Urlöschens nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

Vorgehen

Um auf einer CPU Urlöschen durchzuführen, gehen Sie wie folgt vor:

1. Aktivieren Sie die Task Card "Online Tools" der CPU.
2. Klicken Sie in der Palette "CPU-Bedienpanel" auf die Schaltfläche "MRES".
3. Beantworten Sie die Sicherheitsabfrage mit "OK".

Ergebnis

Die CPU wird ggf. in den Betriebszustand STOP gesetzt, und auf der CPU wird Urlöschen durchgeführt.

Siehe auch

Grundlagen zum Urlöschen (Seite 1370)

Grundlagen zum Urlöschen (Seite 766)

2.1.5.3 Die Uhrzeit einer CPU ermitteln und einstellen

Wo befinden sich die gesuchten Funktionen?

Die Ermittlung und die Änderung der Uhrzeit einer CPU führen Sie in deren Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Uhrzeit einstellen" durch. Dies ist nur bei einer bestehenden Online-Verbindung möglich.

Aufbau der Gruppe "Uhrzeit einstellen"

Die Gruppe "Uhrzeit einstellen" besteht aus folgenden Bereichen:

- Bereich zum Auslesen und Einstellen der Uhrzeit
- Zeitsystem (Diesen Bereich gibt es nicht bei S7-1200. Er wird hier nicht betrachtet.)

Aufbau des Bereichs zum Auslesen und Einstellen der Uhrzeit

Dieser Bereich besteht aus den folgenden Teilen:

- PG / PC-Zeit
  Hier werden die eingestellte Zeitzone, das aktuelle Datum und die aktuelle Uhrzeit Ihres PG / PC angezeigt.
- Baugruppenzeit
  Hier werden die aktuell von der Baugruppe (z. B. CPU) gelesenen und auf Lokalzeit umgerechneten Werte von Datum und Uhrzeit angezeigt.
  Bei aktiviertem Optionskästchen "Von PG / PC übernehmen" bewirkt das Klicken auf die Schaltfläche "Übernehmen" die Übertragung des Datums und der auf UTC-Zeit umgerechneten PG / PC-Zeit auf die Baugruppe.
  Bei deaktiviertem Optionskästchen "Von PG / PC übernehmen" können Sie das Datum und die Uhrzeit für die integrierte Uhr der Baugruppe vorgeben. Nach Klicken auf die Schaltfläche "Übernehmen" werden das Datum und die auf UTC-Zeit umgerechnete Uhrzeit auf die Baugruppe übertragen.

2.1.5.4 Eine S7-1200-CPU auf Werkseinstellungen zurücksetzen

Voraussetzung

- Es steckt keine Memory Card in der CPU.
- Es besteht eine Online-Verbindung zur CPU, die auf Werkseinstellungen zurückgesetzt werden soll.
- Diese CPU befindet sich im Betriebszustand STOP.

Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Rücksetzvorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

Vorgehen

Um eine S7-1200-CPU auf Werkseinstellungen zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen auf Werkseinstellungen".
3. Aktivieren Sie das Optionsfeld "IP-Adresse beibehalten", falls Sie die IP-Adresse beibehalten wollen, bzw. das Optionsfeld "IP-Adresse löschen", wenn Sie die IP-Adresse löschen wollen.

Hinweis

Die beiden genannten Optionsfelder sind nur dann vorhanden, wenn die zurückzusetzende Baugruppe die Wahlmöglichkeit beherrscht, die IP-Adresse beizubehalten oder zu löschen.

4. Klicken Sie auf die Schaltfläche "Rücksetzen".
5. Beantworten Sie die Sicherheitsabfrage mit "OK".

Ergebnis

Die Baugruppe wird ggf. in den Betriebszustand STOP gesetzt und auf Werkseinstellungen zurückgesetzt. Das bedeutet:

- Der Arbeitsspeicher und der interne Ladespeicher und alle Operandenbereiche werden gelöscht.
- Alle Parameter werden auf ihre Voreinstellung zurückgesetzt.
- Der Diagnosepuffer wird gelöscht.
- Die Uhrzeit wird zurückgesetzt.
- Die IP-Adresse wird beibehalten oder gelöscht, je nachdem, welche Einstellung Sie getroffen haben.

2.1.5.5 Eine S7-1500-CPU auf Werkseinstellungen zurücksetzen

Voraussetzung

- Wenn Sie das Rücksetzen auf Werkseinstellungen aus dem Projektkontext aufrufen, muss eine Online-Verbindung zur betroffenen CPU vorhanden sein.
- Die betroffene CPU befindet sich im Betriebszustand STOP.

Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Rücksetzvorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

Vorgehen

Um eine S7-1500-CPU auf Werkseinstellungen zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU (entweder aus dem Projektkontext oder über "Erreichbare Teilnehmer").
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen auf Werkseinstellungen".

3. Aktivieren Sie das Optionsfeld "IP-Adresse beibehalten", falls Sie die IP-Adresse beibehalten wollen, bzw. das Optionsfeld "IP-Adresse löschen", wenn Sie die IP-Adresse löschen wollen.

   Hinweis

   Bei "IP-Adresse löschen" werden alle IP-Adressen gelöscht. Dies gilt unabhängig davon, wie Sie die Online-Verbindung hergestelllt haben.

   Wenn eine Memory Card steckt, bewirkt die Aktivierung des Optionsfelds "IP-Adresse löschen" das Folgende: Die IP-Adressen werden gelöscht, und die CPU wird auf Werkseinstellungen zurückgesetzt. Anschließend wird die auf der Memory Card gespeicherte Konfiguration (einschließlich IP-Adresse) in die CPU übertragen (siehe unten). Wurde die Memory Card vor dem Rücksetzen auf Werkseinstellungen formatiert oder ist sie leer, wird keine IP-Adresse in die CPU übertragen.

4. Klicken Sie auf die Schaltfläche "Rücksetzen".
5. Beantworten Sie die Sicherheitsabfrage mit "OK".

Ergebnis

Die Baugruppe wird ggf. in den Betriebszustand STOP gesetzt und auf Werkseinstellungen zurückgesetzt. Das bedeutet:

- Der Arbeitsspeicher und der interne remanente Systemspeicher und alle Operandenbereiche werden gelöscht.
- Alle Parameter werden auf ihre Voreinstellung zurückgesetzt.
- Der Diagnosepuffer wird gelöscht.
- Die Uhrzeit wird zurückgesetzt.
- Die I&M-Daten werden mit Ausnahme der I&M0-Daten gelöscht.
- Die Betriebsstundenzähler werden zurückgesetzt.
- Die IP-Adresse wird beibehalten oder gelöscht, je nachdem, welche Einstellung Sie getroffen haben.
- Wenn vor dem Rücksetzen auf Werkseinstellungen eine Memory Card gesteckt war, wird die auf der Memory Card enthaltene Konfiguration (Hardware und Software) in die CPU geladen.

2.1.5.6 Ein PROFINET IO-Gerät auf Werkseinstellungen zurücksetzen

Voraussetzung

- Es muss eine Online-Verbindung zum zurückzusetzenden PROFINET IO-Gerät vorhanden sein.

Vorgehen

Um ein PROFINET IO-Gerät auf Werkseinstellungen zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht des Geräts (entweder aus dem Projektkontext oder über "Erreichbare Teilnehmer").
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen auf Werkseinstellungen".
3. Aktivieren Sie das Optionsfeld "Identifikations- und Maintenance-Daten beibehalten" bzw. das Optionsfeld " Identifikations- und Maintenance-Daten löschen", je nachdem, ob Sie die Identifikations- und Maintenance-Daten IM0 bis IM3 auf dem Gerät beibehalten oder löschen wollen.
4. Klicken Sie auf die Schaltfläche "Rücksetzen".
5. Beantworten Sie die Sicherheitsabfrage mit "OK".
   Wenn das zurückzusetzende PROFINET IO-Gerät den Reset-to-Factory-Befehl gemäß PROFINET V2.3 unterstützt, dann werden sein PROFINET-Gerätename, seine IP-Adresse und die SNMP-Parameter (SNMP: Simple Network Management Protocol) zurückgesetzt. Ob die Identifikations- und Maintenance-Daten beibehalten oder gelöscht werden, hängt von Ihrer Auswahl in Schritt 3 ab.
   Der Rücksetzvorgang ist jetzt abgeschlossen.
6. Wenn das zurückzusetzende PROFINET IO-Gerät hingegen den Reset-to-Factory-Befehl gemäß PROFINET V2.3 nicht unterstützt, dann erscheint eine Meldung, dass bei einem Rücksetzen auf Werkseinstellungen sämtliche Daten auf dem Modul verlorengehen können.
   Bestätigen Sie die Sicherheitsabfrage. Der Rücksetzvorgang wird dann durchgeführt. Wenn Sie in Schritt 3 "Identifikations- und Maintenance-Daten löschen" gewählt haben, werden die Identifikations- und Maintenance-Daten IM0 bis IM3 beim Rücksetzen gelöscht. Wenn Sie in Schritt 3 hingegen "Identifikations- und Maintenance-Daten beibehalten" gewählt haben, kann es dennoch passieren, dass die Identifikations- und Maintenance-Daten IM0 bis IM3 beim Rücksetzen gelöscht werden: Das Rücksetz-Verhalten ist in diesem Fall baugruppenabhängig.

Ergebnis

Das PROFINET IO-Gerät wird zurückgesetzt. Welche Daten dabei gelöscht werden, hängt von den oben beschriebenen Faktoren ab.

2.1.5.7 Eine S7-1500-Memory Card formatieren

Voraussetzung

- Wenn Sie die Formatierung der Memory Card aus dem Projektkontext aufrufen, muss eine Online-Verbindung zur betroffenen CPU vorhanden sein.
- Die betroffene CPU befindet sich im Betriebszustand STOP.

Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Formatiervorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

Vorgehen

Um eine S7-1500-Memory Card zu formatieren, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU (entweder aus dem Projektkontext oder über "Erreichbare Teilnehmer").
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Memory Card formatieren".
3. Klicken Sie auf die Schaltfläche "Formatieren".
4. Beantworten Sie die Sicherheitsabfrage mit "Ja".

Ergebnis

- Die Memory Card wird formatiert.
- Die CPU ist vorübergehend nicht erreichbar.
- Die Projektdaten auf der CPU werden mit Ausnahme der IP-Adresse gelöscht.
- Wenn Sie die Formatierung der Memory Card aus dem Projektkontext aufrufen, bleibt die Online- und Diagnosesicht geöffnet, beim Aufruf über "Erreichbare Teilnehmer" wird sie geschlossen.

2.1.5.8 Die Firmware einer Baugruppe aktualisieren

Grundsätzliches zur Firmware-Aktualisierung einer Baugruppe

Firmware-Update durchführen

Mit Hilfe von Firmware-Dateien können Sie die Firmware einer Baugruppe aktualisieren.

Voraussetzungen

- Die Baugruppe ist online verbunden.
- Die Baugruppe unterstützt ein Firmware-Update.
- Für diejenigen Module, die zur korrekten Durchführung des Firmware-Updates eine anliegende Versorgungsspannung benötigen: Die Versorgungsspannung des Moduls ist sichergestellt. Genaueres entnehmen Sie bitte der Dokumentation des Moduls.

Vorgehen

Sie können die Firmware-Aktualisierung einer Baugruppe aus der Projektnavigation oder aus dem Projektkontext aufrufen.

⚠ WARNUNG

Unzulässige Anlagenzustände möglich

Eine S7-1500-CPU geht beim Start der Firmware-Aktualisierung sofort in den Betriebszustand STOP, was sich auf den Betrieb eines Online-Prozesses oder einer Maschine auswirken kann.
Unerwarteter Betrieb eines Prozesses oder einer Maschine kann zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen.

Hinweis

Nach der Durchführung eines Firmware-Updates müssen Sie in der Hardware-Konfiguration Ihres Projekts die betroffene Baugruppe durch die gleiche Baugruppe mit dem aktuellem Firmware-Stand ersetzen. Dann entspricht die projektierte Konfiguration wieder der tatsächlich vorhandenen Konfiguration.

Optionskästchen "Firmware nach Aktualisierung aktivieren"

Wenn Sie das Optionskästchen "Aktiviere Firmware nach Aktualisierung" nicht aktiviert haben, bleibt die bisherige Firmware so lange aktiv, bis die Baugruppe zurückgesetzt wird (z. B. durch einen Übergang NETZ AUS - NETZ EIN). Erst nach dem Rücksetzen der Baugruppe wird die neue Firmware aktiv.

Wenn Sie das Optionskästchen aktiviert haben, wird die Baugruppe nach erfolgreichem Ladevorgang automatisch zurückgesetzt und läuft anschließend mit der neuen Firmware.

Die Aktivierung der Firmware nach der Aktualisierung hat die folgenden Konsequenzen:

- Eine Station führt einen Neustart durch. Dieser hat den Ausfall sämtlicher Module der Station zur Folge.
- Wenn die zugehörige CPU im Betriebszustand RUN ist, kann die Aktivierung der Firmware zu Zugriffsfehlern oder anderen Beeinträchtigungen des Anwenderprogramms führen bis hin zum dauerhaften STOP der CPU.

Hinweis

Bei einigen CPUs ist das Optionskästchen "Aktiviere Firmware nach Aktualisierung" gegraut dargestellt und deaktiviert. In diesem Fall müssen Sie die CPU manuell neu starten.

Bei S7-1500-CPUs ist das Optionskästchen "Aktiviere Firmware nach Aktualisierung" gegraut dargestellt und aktiviert. In diesem Fall wird die neue Firmware unmittelbar nach dem Ladevorgang aktiviert.

Die Firmware-Aktualisierung aus der Projektnavigation aufrufen

Voraussetzung

- Die Baugruppe befindet sich in der ersten Hierarchiestufe hinter einer Kopfbaugruppe (IM oder CPU der Baugruppenfamilie S7-1500 oder ET200SP).
- Die Kopfbaugruppe ist über PROFINET mit dem Engineering System verbunden.

Vorgehen

Gehen SIe folgendermaßen vor:

1. Markieren Sie in der Projektnavigation die zugehörige Kopfbaugruppe.
2. Öffnen Sie über Kontextmenü die Online- und Diagnosesicht der Kopfbaugruppe.
3. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".
4. Wählen Sie "Lokale Module".
   Im Bereich "Online-Daten" werden in einer Tabelle Baugruppen der ersten Hierarchiestufe hinter der Kopfbaugruppe aufgelistet. Bei Siemens-Baugruppen sind das diejenigen Baugruppen, die eine Firmware-Aktualisierung unterstützen. Bei GSDML-basierten Geräten sind das alle Baugruppen.
5. Markieren SIe diejenige Baugruppe, deren Firmware Sie aktualisieren wollen.
6. Klicken Sie im Bereich "Firmware-Lader" auf die Schaltfläche "Durchsuchen", um den Pfad zu den Firmware-Update-Dateien zu wählen.
7. Wählen Sie eine dieser Dateien aus. In der Tabelle werden dann alle Baugruppen aufgelistet, für die mit der gewählten Firmware-Datei ein Update möglich ist.

8. Optional: Aktivieren Sie das Optionskästchen "Aktiviere Firmware nach Aktualisierung", um die Baugruppe nach dem Ladevorgang automatisch zurückzusetzen und die neue Firmware zu starten.
9. Klicken Sie auf die Schaltfläche "Starte Aktualisierung". Wenn die ausgewählte Datei von der Baugruppe interpretiert werden kann, wird sie in die Baugruppe geladen. Falls dazu der Betriebszustand der CPU geändert werden muss, werden Sie über Dialoge dazu aufgefordert.

Die Firmwware-Aktualisierung aus dem Projektkontext aufrufen

Vorgehen

Um ein Firmware-Update durchzuführen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Baugruppe in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".

   Hinweis

   Bei S7-1500-CPUs ist diese Gruppe unterteilt in "PLC" und "Display".

3. Klicken Sie im Bereich "Firmware-Lader" auf die Schaltfläche "Durchsuchen", um den Pfad zu den Firmware-Update-Dateien zu wählen.
4. Wählen Sie eine dieser Dateien aus. In der Tabelle werden dann alle Baugruppen aufgelistet, für die mit der gewählten Firmware-Datei ein Update möglich ist.
5. Optional: Aktivieren Sie das Optionskästchen "Aktiviere Firmware nach Aktualisierung", um die Baugruppe nach dem Ladevorgang automatisch zurückzusetzen und die neue Firmware zu starten.
6. Klicken Sie auf die Schaltfläche "Starte Aktualisierung". Wenn die ausgewählte Datei von der Baugruppe interpretiert werden kann, wird sie in die Baugruppe geladen. Falls dazu der Betriebszustand der CPU geändert werden muss, werden Sie über Dialoge dazu aufgefordert.

2.1.5.9 Einem PROFINET IO-Device eine IP-Adresse zuweisen

Grundsätzliches zur Vergabe einer IP-Adresse an ein PROFINET IO-Device

Übersicht

Alle PROFINET IO-Devices beherrschen das TCP/IP-Protokoll und benötigen daher für den Betrieb am Industrial Ethernet eine IP-Adresse. Nachdem ein IO-Device eine IP-Adresse erhalten hat, ist es über diese Adresse erreichbar. Sie können dann beispielsweise Projektierdaten laden oder eine Diagnose durchführen.

Voraussetzung

- Der Anschluss zum Ethernet LAN muss hergestellt sein.
- Die Ethernet-Schnittstelle ihres PG/PC muss erreichbar sein.
- Das IO-Device, das eine IP-Adresse erhalten soll, muss sich in demselben IP-Band befinden wie das PG/PC.

Die Adresszuweisung über "Erreichbare Teilnehmer" aufrufen

Voraussetzung

- Die über die zugehörige Schnittstelle des PG / PC erreichbaren Teilnehmer werden in der Projektnavigation angezeigt (Dies erreichen Sie, indem Sie entweder in der Projektnavigation auf "Erreichbare Teilnehmer aktualisieren" doppelklicken oder indem Sie im Menü "Online" den Befehl "Erreichbare Teilnehmer..." wählen.).
- Sie haben in der Projektnavigation unter "Online-Zugänge" -> <gewählte Schnittstelle> -> <PROFINET IO-Device> -> "Online & Diagnose" doppelgeklickt, so dass die Online- und Diagnosesicht geöffnet wurde.

Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "IP-Adresse zuweisen". Im Feld "MAC-Adresse" wird die MAC-Adresse des PROFINET IO-Devices angezeigt. Die Schaltfläche "Erreichbare Teilnehmer" ist gegraut.
2. Tragen Sie die gewünschte IP-Adresse ein.
3. Tragen Sie die Subnetzmaske ein.
4. Falls ein Router verwendet werden soll, aktivieren Sie das Optionskästchen "Router verwenden" und tragen anschließend dessen IP-Adresse ein.
5. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

Ergebnis

Dem IO-Device bzw. der betroffenen PROFINET-Schnittstelle des IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder über einen Spannungsausfall hinaus erhalten.

Hinweis

Bei einer S7-1500-CPU können Sie über den oben beschriebenen Weg die IP-Adresse einer PROFINET-Schnittstelle auch dann ändern, wenn über diese Schnittstelle auf die CPU bereits ein Projekt geladen wurde. Die über das Projekt geladene IP-Adresse wird dadurch überschrieben.

Die Adresszuweisung aus dem Projektkontext aufrufen

Voraussetzung

- Es besteht eine Online-Verbindung zum PROFINET IO-Device.
- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device aus dem Projektkontext aufgerufen.
- Das PROFINET IO-Device ist keinem IO-Controller zugeordnet.

Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "IP-Adresse zuweisen".
2. Klicken Sie auf die Schaltfläche "Erreichbare Teilnehmer", um die erreichbaren Geräte zu ermitteln.
   Hinweis: Bei einer S7-1500-CPU gibt es hier zwei Einträge, da sie zwei PROFINET-Schnittstellen besitzt.
3. Wählen Sie das IO-Device aus. Die Felder "IP-Adresse", "Subnetzmaske", das Optionskästchen "Router verwenden" und das Feld "Routeradresse" sind gegraut und enthalten die Knoteneigenschaften, über die Sie den aktuellen Online-Zugang hergestellt haben.
4. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

Ergebnis

Dem IO-Device bzw. der betroffenen PROFINET-Schnittstelle des IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder über einen Spannungsausfall hinaus erhalten.

2.1.5.10 PROFINET-Gerätename vergeben

Grundsätzliches zur Vergabe eines Namens an ein PROFINET IO-Device

Gerätename

Bevor ein IO-Device von einem IO-Controller angesprochen werden kann, muss es einen Gerätenamen haben. Bei PROFINET hat man diese Vorgehensweise gewählt, weil Namen einfacher zu handhaben sind als komplexe IP-Adressen.

Das Zuweisen eines Gerätenamens an ein PROFINET IO-Device ist mit dem Einstellen der PROFIBUS-Adresse bei einem DP-Slave vergleichbar.

Im Auslieferungszustand hat ein IO-Device keinen Gerätenamen. Erst nach der Zuweisung eines Gerätenamens mit dem PG/PC ist ein IO-Device für einen IO-Controller adressierbar, z. B. für die Übertragung der Projektierungsdaten (u. a. die IP-Adresse) im Anlauf oder für den Nutzdatenaustausch im zyklischen Betrieb.

Regeln für den Gerätenamen

Der Gerätename unterliegt folgenden Einschränkungen:

- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein.
- Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Der Bindestrich ist das einzige erlaubte Sonderzeichen.
- Der Gerätename darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Der Gerätename darf nicht mit Ziffern beginnen.
- Der Gerätename darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Gerätename darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).

Wo befindet sich die gesuchte Funktion?

Die Namensvergabe an ein PROFINET IO-Device können Sie an den folgenden Stellen durchführen:

- in dessen Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Name zuweisen". Die Benutzeroberfläche dieser Gruppe hängt davon ab, wie Sie die Online- und Diagnosesicht aufrufen:
  - Aufruf über "Erreichbare Teilnehmer"
  - Aufruf aus dem Projektkontext
- im Dialog "PROFINET-Gerätename vergeben"

Siehe auch

Einen Namen in der Online- und Diagnosesicht vergeben, die über "Erreichbare Teilnehmer" aufgerufen wurde (Seite 3519)

Einen Namen in der Online- und Diagnosesicht vergeben, die aus dem Projektkontext aufgerufen wurde (Seite 3520)

Einen Namen im Dialog "PROFINET-Gerätename vergeben" vergeben (Seite 3521)

Einen Namen in der Online- und Diagnosesicht vergeben, die über "Erreichbare Teilnehmer" aufgerufen wurde

Voraussetzung

- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device über "Erreichbare Teilnehmer aktualisieren" (in der Projektnavigation") oder über "Erreichbare Teilnehmer..." (Menü "Online") aufgerufen.

Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "Name zuweisen". Im Feld "Typ" wird der Baugruppentyp des PROFINET IO-Devices angezeigt.
2. Geben Sie im Eingabefeld "PROFINET-Gerätename" den gewünschten Gerätenamen ein.
3. Optional: Aktivieren Sie das Optionskästchen "LED blinken", um einen LED-Blinktest auf dem PROFINET IO-Device durchzuführen. Damit überprüfen Sie, ob Sie sich auf dem gewünschten IO-Device befinden.

   Hinweis

   Der LED-Blinktest wird nicht von allen PROFINET IO-Devices unterstützt.

   Der LED-Blinktest läuft so lange, bis sie ihn abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED blinken" oder durch Schließen der Online- und Diagnosesicht.
4. Klicken Sie auf die Schaltfläche "Name zuweisen".

Ergebnis

Dem PROFINET IO-Device wird der vorgegebene Name zugewiesen.

Einen Namen in der Online- und Diagnosesicht vergeben, die aus dem Projektkontext aufgerufen wurde

Voraussetzung

- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device aus dem Projektkontext aufgerufen.
- Das PROFINET IO-Device kann über mindestens eine PG/PC-Schnittstelle erreicht werden.

Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "Name zuweisen". In der Klappliste "PROFINET-Gerätename" wird der aktuell im Offline-Projekt vorhandene Name und im Feld "Typ" der Baugruppentyp des PROFINET IO-Device angezeigt.

   Hinweis

   Bei CPUs mit mehreren PROFINET-Schnittstellen werden die im Offline-Projekt vorhandenen Namen aller PROFINET-Schnittstellen angezeigt.

2. Wählen Sie aus der Klappliste ggf. einen anderen Namen.

   Hinweis

   In den Schritten 3 bis 5 ermitteln Sie die am PROFINET-Subnetz vorhandenen IO-Devices.

3. Wählen Sie in der Klappliste "PG/PC-Schnittstelle für die Zuweisung" diejenige PG/PC-Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
4. Optional: Treffen Sie über die drei Optionskästchen eine Auswahl unter allen online vorhandenen IO-Devices.
5. Klicken Sie auf das Symbol zur Ermittlung der am PROFINET-Subnetz vorhandenen IO-Devices. Anschließend wird die Tabelle aktualisiert.
6. Markieren Sie in der Tabelle das gewünschte IO-Device.
7. Optional: Aktivieren Sie das Optionskästchen "LED blinken", um einen LED-Blinktest auf dem PROFINET IO-Device durchzuführen. Damit überprüfen Sie, ob Sie sich auf dem gewünschten IO-Device befinden.

   Hinweis

   Der LED-Blinktest wird nicht von allen PROFINET IO-Devices unterstützt.

   Der LED-Blinktest läuft so lange, bis sie ihn abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED blinken", durch Selektieren eines anderen IO-Devices in der Tabelle, durch Schließen der Online- und Diagnosesicht.
8. Klicken Sie auf die Schaltfläche "Name zuweisen".

Ergebnis

Dem PROFINET IO-Device bzw. einer seiner PROFINET-Schnittstellen wird der ausgewählte Name zugewiesen.

Einen Namen im Dialog "PROFINET-Gerätename vergeben" vergeben

Voraussetzung

- Sie haben den Dialog "PROFINET-Gerätename vergeben" aus der Netzsicht aufgerufen (aus dem Kontextmenü einer PN/IE-Verbindung).
- Das PROFINET IO-Device kann über mindestens eine PG/PC-Schnittstelle erreicht werden.

Vorgehen

1. In der Klappliste "PROFINET-Gerätename" wird das Folgende angezeigt:
   - der aktuell im Offline-Projekt vorhandene Name derjenigen Schnittstelle, über die man den Dialog geöffnet hat
   - die Namen derjenigen IO-Devices, die über diese Schnittstelle verbunden sind

   im Feld "Typ" wird der Baugruppentyp des PROFINET IO-Device angezeigt.
   Wählen Sie aus der Klappliste ggf. einen anderen Namen.

   Hinweis

   In den Schritten 2 bis 4 ermitteln Sie die am PROFINET-Subnetz vorhandenen IO-Devices.

2. Wählen Sie in der Klappliste "PG/PC-Schnittstelle für die Zuweisung" diejenige PG/PC-Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
3. Optional: Treffen Sie über die drei Optionskästchen eine Auswahl unter allen online vorhandenen IO-Devices.
4. Klicken Sie auf das Symbol zur Ermittlung der am PROFINET-Subnetz vorhandenen IO-Devices. Anschließend wird die Tabelle aktualisiert.
5. Markieren Sie in der Tabelle das gewünschte IO-Device.
6. Optional: Aktivieren Sie das Optionskästchen "LED blinken", um einen LED-Blinktest auf dem PROFINET IO-Device durchzuführen. Damit überprüfen Sie, ob Sie sich auf dem gewünschten IO-Device befinden.

   Hinweis

   Der LED-Blinktest wird nicht von allen PROFINET IO-Devices unterstützt.

   Der LED-Blinktest läuft so lange, bis sie ihn abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED blinken", durch Selektieren eines anderen IO-Devices in der Tabelle.
7. Klicken Sie auf die Schaltfläche "Name zuweisen".

Ergebnis

Dem PROFINET IO-Device bzw. der Schnittstelle, über die man den Dialog geöffnet hat, wird der ausgewählte Name zugewiesen.

2.1.5.11 Ein S7-1500-Analogmodul kalibrieren

Ein S7-1500-Analogmodul kalibrieren - Überblick

Wo kalibrieren Sie ein S7-1500-Analogmodul?

Die Kalibrierung eines S7-1500-Analogmoduls führen Sie in dessen Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Kalibrieren" durch.

Übersicht über den Funktionsumfang der Kalibrierfunktion

In der Gruppe "Kalibrieren" können Sie die folgenden Funktionen bei einem S7-1500-Analogmodul durchführen:

- Feststellen der aktuellen Kalibrierung aller Kanäle
- Kalibrieren eines Kanals
- Abbrechen eines laufenden Kalibriervorgangs
- Rücksetzen der Kalibrierung eines Kanals auf die Werkseinstellungen

Voraussetzung für die im Folgenden beschriebene Kalibrierfunktion

Für die im Folgenden beschriebene Kalibrierfunktion wird das Folgende vorausgesetzt:

- Sie haben die Online- und Diagnosesicht aus dem Projektkontext aufgerufen (also nicht aus der Projektnavigation oder über das Menü "Online").
- Es besteht eine Online-Verbindung zum Analogmodul, das kalibriert werden soll.
- Offline- und Online-Projektierung sind identisch.

Ein S7-1500-Analogmodul kalibrieren

Übersicht über die Kalibrierung eines Kanals eines S7-1500-Analogmoduls

Die Kalibrierung eines Kanals eines S7-1500-Analogmoduls besteht aus folgenden Schritten:

1. Kalibriervorgang starten
2. Zweiten bis vorletzten Schritt des Kalibriervorgangs durchführen
3. Kalibriervorgang abschließen

Darauf wird im Folgenden näher eingegangen.

Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.
- Aktuell läuft kein Kalibriervorgang auf dem Analogmodul (wenn Sie den Kalibriervorgang starten wollen) bzw. der zuletzt angestoßene Schritt wurde erfolgreich durchgeführt (wenn Sie den Kalibriervorgang fortsetzen oder abschließen wollen).

Vorgehen für den Start des Kalibriervorgangs

Um den Kalibriervorgang zu starten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Übersichtstabelle diejenige Zeile, die zum zu kalibrierenden Kanal gehört.
2. Klicken Sie auf die Schaltfläche "Manuelle Kalibrierung starten".

Anschließend ändert sich die Bedienoberfläche wie folgt:

- Die Übersichtstabelle und die Schaltflächen "Manuelle Kalibrierung starten" und "Auf Werkseinstellungen setzen" werden inaktiv.
- Die Schrittanzeige wird aktiviert und zeigt die Nummer des aktuellen und des letzten Schritts an.
- Das Feld "Befehl" wird aktiv und zeigt an, was der Anwender im nächsten Kalibrierungsschritt zu tun hat.
- Das Feld "Status" wird aktiv und zeigt den aktuellen Zustand im Kalibrierungsvorgang an, z. B. "Kalibrierung erfolgreich gestartet".
- Das Feld "Messwert" wird aktiv. Bei einem Eingabemodul wird hier ein Wert angezeigt, bei einem Ausgabemodul müssen Sie hier einen Wert eingeben.
- Die Schaltfläche "Abbrechen" wird aktiv.
- Bei bestimmten Analogmodulen (AI Energy Meter) wird die Schaltfläche "Überspringen" aktiv. Damit kann der nächste Schritt im Kalibriervorgang übersprungen werden.
- Die Schaltfläche "Weiter" wird aktiv. Damit kann der nächste Schritt im Kalibriervorgang durchgeführt werden.

Vorgehen für den zweiten bis vorletzten Schritt des Kalibriervorgangs

Sie gehen folgendermaßen vor:

1. Klicken sie auf die Schaltfläche "Weiter".

Anschließend werden die oben beschriebenen Felder der Bedienoberfläche aktualisiert.

Bei bestimmten Analogmodulen (AI Energy Meter) können Sie den nächsten Kalibrierungsschritt überspringen, sofern der nächste Schritt nicht der letzte Schritt des Kalibriervorgangs ist. Dazu gehen Sie folgendermaßen vor:

1. Klicken sie auf die Schaltfläche "Überspringen".

Anschließend werden die oben beschriebenen Felder der Bedienoberfläche aktualisiert.

Vorgehen für den letzten Schritt des Kalibriervorgangs

Sie gehen folgendermaßen vor:

1. Klicken sie auf die Schaltfläche "Weiter".

Anschließend ändert sich die Bedienoberfläche wie folgt:

- Die Übersichtstabelle wird aktiv.
- Die Kalibrierungsanzeige des kalibrierten Kanals wird aktualisiert.

- Die Schaltflächen "Manuelle Kalibrierung starten" und "Auf Werkseinstellungen setzen" werden aktiv.
- Die Schrittanzeige wird deaktiviert und die Nummern des aktuellen und des letzten Schritts sind leer.
- Das Feld "Befehl" wird inaktiv und ist leer.
- Das Feld "Status" wird inaktiv und zeigt den letzten Zustand im Kalibrierungsvorgang an, z. B. "Kalibrierung erfolgreich beendet".
- Das Feld "Messwert" wird inaktiv und ist leer.
- Die Schaltfläche "Abbrechen" wird inaktiv.
- Die Schaltfläche "Überspringen" wird inaktiv.
- Die Schaltfläche "Weiter" wird inaktiv.

Auftreten eines Fehlers

Wenn während des Kalibriervorgangs ein Fehler auftritt, bricht das Modul den Kalibriervorgang ab. Der Kanal, der kalibriert werden sollte, hat danach dieselben Einstellungen wie vor Beginn des Kalibriervorgangs.

Die Bedienoberfläche sieht nach dem Auftreten eines Fehlers mit Ausnahme des Felds "Status" aus wie vor Beginn des Kalibriervorgangs. Das Feld "Status" zeigt den Fehler an, den das Modul beim Kalibriervorgang erkannt hat.

Einen laufenden Kalibriervorgang eines S7-1500-Analogmoduls abbrechen

Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.
- Aktuell läuft ein Kalibriervorgang auf dem Analogmodul.

Vorgehen

Um einen laufenden Kalibriervorgang abzubrechen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Abbrechen".

Ergebnis

Der laufende Kalibriervorgang wird abgebrochen, und der zu kalibrierende Kanal hat danach dieselben Einstellungen wie vor Beginn des Kalibriervorgangs.

In der Bedienoberfläche werden alle Bedienelemente deaktiviert, bis der Abbruch abgeschlossen ist. Die Bedienoberfläche sieht danach mit Ausnahme des Felds "Status" aus wie vor Beginn des Kalibriervorgangs. Das Feld "Status" zeigt das Ergebnis des Abbruchs an.

Ein S7-1500-Analogmodul auf Werkseinstellungen zurücksetzen

Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.

Vorgehen

Um einen Kanal eines S7-1500-Analogmoduls auf Werkseinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Übersichtstabelle die zum rückzusetzenden Kanal gehörige Zeile.
2. Klicken Sie auf die Schaltfläche "Auf Werkseinstellungen setzen".

Ergebnis

In der Bedienoberfläche werden alle Bedienelemente deaktiviert, bis der Rücksetzvorgang abgeschlossen ist. Die Bedienoberfläche sieht danach mit Ausnahme des Felds "Status" aus wie vor Beginn des Rücksetzvorgangs. Das Feld "Status" zeigt das Ergebnis des Rücksetzvorgangs an.

2.1.5.12 I&M-Daten in PROFINET-IO-Geräte und ihre Module laden

Welche I&M-Daten können Sie in PROFINET-IO-Geräte und ihre Module laden?

Sie können die I&M 1-Daten (Anlagenkennzeichen und Ortskennzeichen) und / oder die I&M 2-Daten (Einbaudatum) und / oder die I&M 3-Daten (Zusatzinformation) in die reale Hardware laden.

Voraussetzung

- In den Projekteinstellungen (Extras > Einstellungen, Hardware-Konfiguration > Übersetzen und herunterladen) ist die Option "I&M-Daten herunterladen" aktiviert.
- Es besteht eine Online-Verbindung zu den PROFINET-IO-Geräten und ihren Modulen, in die Sie I&M-Daten laden wollen.
- Sie haben in den Eigenschaften der betroffenen PROFINET-IO-Geräte und ihren Modulen (Inspektorfenster: Register „Eigenschaften“ > Register „Allgemein“, Einstellungen > Identification & Maintenance) die zu ladenden I&M-Daten eingetragen.

Wo legen Sie fest, welche I&M-Daten in welche PROFINET-IO-Geräte geladen werden?

Welche I&M-Daten Sie in welche PROFINET-IO-Geräte laden wollen, legen Sie im Dialog "Vorschau laden" fest. Dabei haben Sie in der Klappliste der Zeile „Identifikations- und Maintenance-Daten (I&M)“ die folgenden Alternativen:

- Nichts laden
  Die Optionskästchen zu allen PROFINET-IO-Geräten sind deaktiviert, ebenso die Optionskästchen zu den ladbaren I&M-Daten.
  Diese Einstellung bewirkt, dass beim Laden keine I&M-Daten auf die reale Hardware übertragen werden.
- Daten laden
  Die Optionskästchen zu allen PROFINET-IO-Geräten sind aktiviert, ebenso die Optionskästchen zu den ladbaren I&M-Daten.
  Diese Einstellung bewirkt, dass beim Laden die jeweiligen I&M 1-, I&M 2- und I&M 3-Daten auf sämtliche PROFINET-IO-Geräte übertragen werden.
- Ausgewählte laden
  Sie aktivieren die Optionskästchen derjenigen PROFINET-IO-Geräte, in die Sie I&M-Daten laden wollen. Sie aktivieren außerdem die Optionskästchen derjenigen Identifikationsdaten, die Sie laden wollen.
  Diese Einstellung bewirkt, dass beim Laden die ausgewählten I&M-Daten auf die ausgewählten PROFINET-IO-Geräte übertragen werden.

Hinweis

Sprachabhängigkeit der zu ladenden I&M-Daten

Die I&M-Daten werden in der Form, in der Sie sie in den Eigenschaften der betroffenen PROFINET-IO-Geräte und ihren Modulen vorgegeben haben, in die reale Hardware geladen. Es besteht keine Sprachabhängigkeit.

2.1.6 Diagnose im Betriebszustand STOP

2.1.6.1 Grundlagen zum Diagnosepuffer

Funktion

Das Betriebssystem der CPU trägt die von der CPU und diagnosefähigen Baugruppen erkannten Fehler in der Reihenfolge ihres Auftretens in den Diagnosepuffer ein. Dazu gehören u. a. die folgenden Ereignisse:

- Jeder Betriebszustandswechsel der CPU (z. B. NETZEIN, Übergang in den Betriebszustand STOP, Übergang in den Betriebszustand RUN)
- Jeder Prozess- und jeder Diagnosealarm

Der Eintrag an oberster Stelle enthält das neueste Ereignis. Die Einträge im Diagnosepuffer werden dauerhaft gespeichert: Sie bleiben auch beim Ausfall der Spannungsversorgung erhalten und können nur durch Rücksetzen der CPU auf Werkseinstellungen gelöscht werden.

Ein Diagnosepuffereintrag enthält die folgenden Elemente:

- Zeitstempel
- Fehler-ID
- Zusätzliche Fehler-ID-spezifische Information

Vorteile des Diagnosepuffers

Der Diagnosepuffer bietet folgende Vorteile:

- Nachdem die CPU in den Betriebszustand STOP gegangen ist, können Sie die letzten Ereignisse vor STOP auswerten und so die STOP-Ursache einkreisen bzw. ermitteln.
- Sie können Fehlerursachen schneller erkennen und beheben und dadurch die Verfügbarkeit der Anlage erhöhen.
- Sie können das dynamische Anlagenverhalten auswerten und anschließend optimieren.

Organisation des Diagnosepuffers

Der Diagnosepuffer ist ein Ringpuffer. Die maximale Anzahl der Einträge ist bei den S7-1200-CPUs 50. Wenn der Diagnosepuffer voll ist und ein weiterer Eintrag erfolgen soll, werden alle vorhandenen Einträge um eine Stelle weitergerückt (mit der Folge, dass der älteste Eintrag gelöscht wird), und der neue Eintrag gelangt an die frei gewordene erste Stelle (FIFO-Prinzip: first in, first out).

Auswertung des Diagnosepuffers

Sie haben folgende Möglichkeit, auf Inhalte des Diagnosepuffers zuzugreifen:

- Über die Online- und Diagnosesicht

Durch die Auswertung der Ereignisse vor dem Fehlerereignis (z. B. Betriebszustandsübergang nach STOP) verschaffen Sie sich ein Bild über die mögliche Ursache bzw. kreisen sie näher ein oder spezifizieren sie genauer (abhängig vom Fehlertyp).

Lesen Sie die Detailangaben zu den Ereignissen sorgfältig und nutzen Sie das Textfeld "Hilfe zum Ereignis", um weitere Informationen und mögliche Ursachen einzelner Einträge zu erhalten.

Hinweis

Um die Zeitangaben der Diagnosepuffer-Einträge bei zeitkritischen Anlagen sinnvoll nutzen zu können, wird empfohlen, die Uhrzeit und das Datum auf der CPU gelegentlich zu überprüfen und gegebenenfalls zu korrigieren.

Alternativ besteht die Möglichkeit der Uhrzeitsynchronisation über einen NTP-Zeitserver.

Siehe auch

Eine S7-1200-CPU auf Werkseinstellungen zurücksetzen (Seite 3509)

Die STOP-Ursache einer CPU ermitteln (Seite 3529)

Die Uhrzeit einer CPU ermitteln und einstellen (Seite 3508)

Parametrieren der Uhr (Seite 1400)

2.1.6.2 Die STOP-Ursache einer CPU ermitteln

Voraussetzung

Die zu analysierende CPU befindet sich im Betriebszustand STOP.

Vorgehen

Um die STOP-Ursache einer CPU zu ermitteln, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU.
2. Wählen Sie im Ordner "Diagnose" die Gruppe "Diagnosepuffer".
3. Werten Sie die Ereignisse vor dem Übergang in den Betriebszustand STOP aus.
   Verschaffen Sie sich damit ein Bild über die mögliche Ursache bzw. kreisen Sie sie näher ein oder spezifizieren Sie sie genauer (abhängig vom Fehlertyp).
   Lesen Sie die Detailangaben zu den Ereignissen sorgfältig und nutzen Sie die Schaltfläche "Hilfe zum Ereignis", um weitere Informationen und mögliche Ursachen einzelner Einträge zu erhalten.

Ergebnis

Sie konnten die STOP-Ursache der CPU näher einkreisen bzw. genau ermitteln.

Hinweis

Wenn Sie bei der Analyse nicht weiterkommen, wenden Sie sich an den Customer Support. Sichern Sie in diesem Fall den Inhalt des Diagnosepuffers mit der Schaltfläche "Speichern unter" in eine Textdatei und stellen Sie diese dem Customer Support zur Verfügung.

Siehe auch

Den Diagnosepuffer einer CPU auslesen (Seite 3501)

2.1.7 Online-Zugänge in der Online- und Diagnosesicht

2.1.7.1 Status der Online-Verbindung anzeigen

Voraussetzung

- Das zugehörige Gerät kann über mindestens eine PG/PC-Schnittstelle erreicht werden.

Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, dessen Status der Online-Verbindung angezeigt werden soll.
2. Wählen Sie die Gruppe "Online-Zugänge".

Hinweis

Die Gruppe "Online-Zugänge" existiert nur bei CPUs und einigen CPs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nícht angezeigt.

Ergebnis

Der Status der Online-Verbindung wird im Bereich "Status" sowohl grafisch als auch in Worten angezeigt.

2.1.7.2 PG/PC-Schnittstelle festlegen, Online verbinden

Voraussetzung

- Das zugehörige Gerät kann über mindestens eine PG/PC-Schnittstelle erreicht werden.
- Zum zugehörigen Gerät besteht momentan keine Online-Verbindung.

Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, zu dem Sie eine Online-Verbindung herstellen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Online-Zugänge".

Hinweis

Die Gruppe "Online-Zugänge" existiert bei CPUs und einigen CPs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

3. Wenn zum Gerät bereits zuvor eine Online-Verbindung bestanden hat, sind in den Klapplisten die zugehörigen Werte dieser Online-Verbindung voreingestellt. In diesem Fall können Sie sofort mit dem letzten Schritt dieser Handlungsanweisung fortfahren, sofern Sie zwischenzeitlich die IP-Adresse nicht über die Online- und Diagnosesicht geändert haben.
4. Wählen Sie in der Klappliste "Typ der PG/PC-Schnittstelle" den Schnittstellentyp. Abhängig von dieser Auswahl werden in der Klappliste "PG/PC-Schnittstelle für den Online-Zugang" nur noch diejenigen Schnittstellen des PG/PC angezeigt, die dem gewählten Schnittstellentyp entsprechen.
5. Wählen Sie in der Klappliste "PG/PC-Schnittstelle für den Online-Zugang" diejenige PG/PC-Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.

6. Optional: Klicken Sie auf die Schaltfläche "Eigenschaften", um die Eigenschaften des zugehörigen CP zu verändern.
7. Wählen Sie in der Klappliste "Verbindung mit Subnetz" dasjenige Subnetz aus, über welches das Gerät mit der PG/PC-Schnittstelle verbunden ist.

   Hinweis

   Die PG/PC-Schnittstelle sei mit einer Schnittstelle eines Geräts verbunden.

   Wenn Sie ausschließlich auf dieses Gerät zugreifen wollen, dann wählen Sie in der Klappliste die Einstellung "Direkt am Steckplatz <Schnittstellenbezeichnung>".

   Wenn Sie hingegen mittels Routing auf ein anderes Gerät zugreifen wollen, dann legen Sie in der Gerätekonfiguration an dieser Schnittstelle ein Subnetz an und wählen Sie anschließend in der Klapppliste dieses Subnetz aus.

8. Falls das Gerät über ein Gateway erreichbar ist, wählen Sie aus der Klappliste "1. Gateway" dasjenige Gateway aus, das die betroffenen Subnetze miteinander verbindet.
9. Geben Sie im Eingabefeld "Geräteadresse" ggf. die IP-Adresse des Geräts ein, zu dem Sie eine Online-Verbindung herstellen wollen.

   Hinweis

   Bei CPUs mit mehreren IP-Adressen wählen Sie über die Klappliste "Geräteadresse" die IP-Adresse derjenigen PROFINET-Schnittstelle, über die Sie eine Online-Verbindung herstellen wollen.

10. Alternativ: Klicken Sie auf die Schaltfläche "Erreichbare Teilnehmer anzeigen" und wählen Sie das Gerät, zu dem Sie eine Online-Verbindung herstellen wollen, aus der Liste der erreichbaren Teilnehmer.
11. Klicken Sie auf die Schaltfläche "Online verbinden".

Ergebnis

Die Online-Verbindung zum gewünschten Gerät wird aufgebaut.

2.1.7.3 Online-Verbindung trennen

Voraussetzung

- Zum zugehörigen Gerät besteht momentan eine Online-Verbindung.

Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, dessen Online-Verbindung Sie trennen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Online-Zugänge".

   Hinweis

   Die Gruppe "Online-Zugänge" existiert nur bei CPUs und einigen CPs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

3. Klicken Sie auf die Schaltfläche "Online-Verbindung trennen".

Ergebnis

Die Online-Verbindung zum gewünschten Gerät wird getrennt.

2.1.7.4 Den Blinktest bei einem online verbundenen Gerät durchführen

Voraussetzung

- Zum zugehörigen Gerät besteht momentan eine Online-Verbindung.
- Die Funktion FORCEN ist nicht aktiv.

Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, bei dem Sie einen Blinktest durchführen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Status".

   Hinweis

   Die Gruppe "Online-Zugänge" existiert nur bei CPUs und einigen CPs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

3. Aktivieren Sie das Optionskästchen "LED-Blinktest".

Ergebnis

- Bei einer S7-1200-CPU blinken die LEDs RUN/STOP, ERROR und MAINT.
- Bei einer S7-1500-CPU blinken die LEDs RUN/STOP, ERROR und MAINT.
- Bei einer S7-300- oder S7-400-CPU blinkt die FRCE-LED.

Die LEDs blinken so lange, bis Sie den Blinktest abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED-Blinktest", durch Wechsel zu einer anderen Gruppe der Online- und Diagnosesicht, durch Änderung von Einstellungen im Bereich "Online-Zugänge".

2.1.8 PROFIBUS DP-Subnetze auf Störungen überprüfen

2.1.8.1 Grundlagen zum Diagnose-Repeater

Was ist der Diagnose-Repeater?

Der Diagnose-Repeater ist ein Repeater mit der Fähigkeit, ein Segment eines RS 485-PROFIBUS-Subnetzes (Kupferleitung) im laufenden Betrieb zu überwachen und Leitungsfehler per Diagnosetelegramm an den DP-Master zu melden.

Der Diagnose-Repeater ermöglicht durch seine Leitungsdiagnose im laufenden Betrieb, Leitungsfehler frühzeitig zu erkennen, zu lokalisieren und zu visualisieren. Damit werden Anlagenstörungen frühzeitig erkannt und Anlagenstillstände minimiert.

Funktionsweise des Diagnose-Repeaters

Der Diagnose-Repeater kann an den Segmenten DP2 und DP3 eine Leitungsdiagnose durchführen, weil er für diese Segmente eine Messschaltung besitzt.

Die Leitungsdiagnose läuft in zwei Schritten ab:

- 1. Schritt: Topologieermittlung
  Sie starten die Topologieermittlung, indem Sie in Ihrem Programm die erweiterte Anweisung "DP_TOPOL" aufrufen.
  Der Diagnose-Repeater ermittelt dabei die PROFIBUS-Adressen und die Entfernung der Teilnehmer und erstellt eine Topologietabelle.
- 2. Schritt: Störstellenermittlung
  Der Diagnose-Repeater prüft im laufenden Busbetrieb die Leitungen. Er ermittelt die Entfernung der Störstelle, bestimmt die Fehlerursache und setzt eine Diagnosemeldung mit relativer Angabe des Fehlerorts ab.

Anzeige detaillierter Informationen zur ermittelten Störstelle

In der Online- und Diagnosesicht des Diagnose-Repeaters erhalten Sie detaillierte Informationen zur ermittelten Störstelle:

- Mittels Symbolen
- Mittels einer grafischen und einer textuellen Anzeige

Siehe auch

Status der Segmentdiagnose mittels Symbolen anzeigen (Seite 3534)

Status der Segmentdiagnose grafisch und textuell anzeigen (Seite 3534)

2.1.8.2 Status der Segmentdiagnose mittels Symbolen anzeigen

Wo befinden sich die gesuchten Informationen?

Die Symbole zum Status der Segmentdiagnose finden Sie an der folgenden Stelle:

- Im Navigationsfenster der Online- und Diagnosesicht des zugehörigen Diagnose-Repeaters im aufgeklappten Ordner "Segmentdiagnose"

Das zum Segment gehörige Diagnosesymbol wird hinter der Segmentbezeichnung angezeigt. Dabei ist zu beachten, dass Leitungsfehler nur bei den Segmenten DP2 und DP3 angezeigt werden. Die Segmente DP1 und PG zeigen keine Fehler in Form eines Diagnosesymbols an, sondern melden nur wenige Busfehler.

Diagnosesymbole

Die folgende Tabelle zeigt die möglichen Symbole und ihre Bedeutung.

Symbol	Bedeutung
	Segment ist fehlerfrei
	Segment enthält Fehler
	Segment ist abgeschaltet

2.1.8.3 Status der Segmentdiagnose grafisch und textuell anzeigen

Wo wird der Status der Segmentdiagnose grafisch und textuell angezeigt?

Die grafische und textuelle Anzeige des Status der Segmentdiagnose erfolgt in der Online- und Diagnosesicht des zugehörigen Diagnose-Repeaters im Ordner "Segmentdiagnose" in den Gruppen "DP1", "DP2", "DP3" und "PG".

Aufbau der Gruppen "DP1", "DP2" "DP3" und "PG"

Die Gruppen "DP1", "DP2", "DP3" und "PG" bestehen aus folgenden Elementen:

- Feld "Fehlerort"
- Feld "Fehler"
- Feld "Behebung"
- Schaltfläche "Hilfe zum Ereignis"
- Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

Feld "Fehlerort"

Hier wird der Fehlerort grafisch dargestellt, sofern der Diagnose-Repeater den Ort ermitteln kann.

Das folgende Bild zeigt ein Beispiel für einen aufgetretenen Leitungsfehler im Segment DP2.

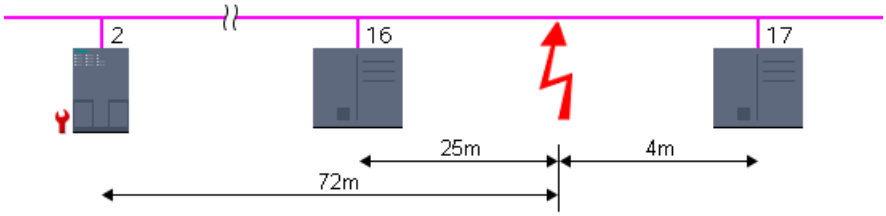

In diesem Beispiel hat der Diagnose-Repeater die PROFIBUS-Adresse 2, und zwischen den Teilnehmern mit den PROFIBUS-Adressen 16 und 17 ist ein Leitungsfehler aufgetreten. Dieser Leitungsfehler befindet sich 25 m vom Teilnehmer 16 entfernt, 4 m vom Teilnehmer 17 entfernt und 72 m vom Diagnose-Repeater entfernt.

Feld "Fehler"

Hier wird der Fehler im Klartext erläutert.

Feld "Behebung"

Hier finden Sie Maßnahmen zur Behebung des Fehlers.

Schaltfläche "Hilfe zum Ereignis"

Wenn Sie auf diese Schaltfläche klicken, wird der aufgetretene Fehler näher erläutert und es werden ggf. weitere Einzelheiten zu seiner Behebung gegeben.

Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

Die Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben" ist nur aktiv, wenn eine Online-Verbindung zum Diagnose-Repeater besteht.

Die Voreinstellung ist "Anzeige einfrieren".

Wenn Sie auf die Schaltfläche "Anzeige einfrieren" klicken, geschieht Folgendes:

- Die aktuelle Anzeige der Segmentdiagnose wird eingefroren.
- Die Beschriftung der Schaltfläche wechselt zu " Einfrieren aufheben".

Wenn die Anzeige eingefroren ist und Sie auf die Schaltfläche "Einfrieren aufheben" klicken, geschieht Folgendes:

- Die Anzeige der Segmentdiagnose wird wieder aktualisiert.
- Die Beschriftung der Schaltfläche wechselt zu "Anzeige einfrieren".

2.2 Verbindungen diagnostizieren

2.2.1 Überblick über die Verbindungsdiagnose

Grundlagen

Unter Verbindungsdiagnose wird im Folgenden die Diagnose von Kommunikationsverbindungen verstanden.

Die Verbindungsdiagnose wird bei jedem Aufbau der Online-Verbindung zu einer Baugruppe (CPU oder CP), die an einem oder mehreren Kommunikationsdiensten beteiligt ist, gestartet. Aktualisierungen des Verbindungsstatus erfolgen automatisch im Hintergrund.

Bei einseitigen Verbindungen muss zu demjenigen Kommunikationspartner eine Online-Verbindung bestehen, der die Kommunikationsverbindung aufgebaut hat.

Bei zweiseitigen Verbindungen sind die folgenden beiden Fälle zu unterscheiden:

- Wenn zu genau einem Verbindungsendpunkt eine Online-Verbindung besteht, kann nur der zu diesem Verbindungsendpunkt gehörende Verbindungsteil diagnostiziert werden.
- Wenn zu beiden Verbindungsendpunkten eine Online-Verbindung besteht, können beide Verbindungsteile (und damit die gesamte Verbindung) diagnostiziert werden.

Prinzipielle Möglichkeiten der Verbindungsdiagnose

Sie können Verbindungen wie folgt diagnostizieren:

- Über die Anzeige des Verbindungsstatus mittels Symbolen
  Diese Anzeige erfolgt in der Verbindungstabelle.
- Durch eine detaillierte Verbindungsdiagnose
  Diese erfolgt im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

Voraussetzung für die im Folgenden beschriebene Verbindungsdiagnose

Die Verbindungsdiagnose ist nur möglich, wenn die an der Verbindung beteiligten Geräte die entsprechenden Diagnosedaten zur Verfügung stellen.

In der Verbindungstabelle können Sie sich die Details entweder aller im Projekt angelegten Kommunikationsverbindungen (Voreinstellung) oder auswahlbezogener Kommunikationsverbindungen anzeigen lassen.

Für die im Folgenden beschriebene Verbindungsdiagnose wird vorausgesetzt, dass Sie sich die Details auswahlbezogener Kommunikationsverbindungen anzeigen lassen. Deaktivieren Sie hierzu im Kontextmenü die Option "Alle Verbindungen anzeigen".

Siehe auch

Arbeiten mit der Verbindungstabelle (Seite 237)

2.2.2 Verbindungsstatus mittels Symbolen anzeigen

Inhalt der Verbindungstabelle bei nicht aufgebauter Online-Verbindung

- Im Kontext einer CPU oder eines CP werden in der Verbindungstabelle bei nicht aufgebauter Online-Verbindung die offline projektierten Kommunikationsverbindungen (inkl. Eigenschaften) aufgelistet.

Inhalt der Verbindungstabelle bei aufgebauter Online-Verbindung

Nach dem Aufbau der Online-Verbindung werden die Eigenschaften der offline aufgelisteten Kommunikationsverbindungen um Diagnosesymbole für den Verbindungsstatus (Spalte "Online-Status") ergänzt.

Zusätzlich enthält die Verbindungstabelle jetzt auch Einträge zu allen Kommunikationsverbindungen, die nur online vorhanden sind (z. B. Verbindungen bei den Anweisungen zur Open User Communication, PG- und OP-Verbindungen, Verbindungen für den Webserver-Zugriff).

Bei Verbindungen, die ausschließlich online oder offline vorhanden sind, wird das Diagnosesymbol rechts unten mit einem kleineren Zusatzsymbol für den Vergleichsstatus kombiniert.

Diagnosesymbole für Kommunikationsverbindungen

Die folgende Tabelle zeigt die Diagnosesymbole für Kommunikationsverbindungen.

Symbol	Bedeutung
	Verbindung aufgebaut
	Verbindung nicht aufgebaut / wird aufgebaut
	Verbindung nicht verfügbar

Diagnosesymbole für den Vergleichsstatus

Die Diagnosesymbole für Kommunikationsverbindungen können rechts unten mit kleineren Zusatzsymbolen kombiniert werden, die das Ergebnis des Online-/Offline-Vergleichs anzeigen. Die folgende Tabelle zeigt die möglichen Vergleichssymbole und ihre Bedeutung.

Symbol	Bedeutung
	Verbindung nur online vorhanden
	Verbindung nur offline vorhanden

2.2.3 Verbindungen im Detail diagnostizieren

2.2.3.1 Verbindungen im Detail diagnostizieren - Überblick

Wo diagnostizieren Sie Verbindungen im Detail?

Die detaillierte Diagnose von Verbindungen führen Sie im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters durch.

Wie öffnen Sie den Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters?

Sie haben die folgenden Möglichkeiten, um das Register "Verbindungsinformation" des Inspektorfensters zu öffnen:

- Markieren Sie in der Verbindungstabelle diejenige Zeile, die zu der betrachteten Verbindung gehört. Klicken Sie dann im Inspektorfenster nacheinander auf die Register "Diagnose" und "Verbindungsinformation".
- Doppelklicken Sie in der Verbindungstabelle auf das Diagnosesymbol der betrachteten Verbindung.
- Sie befinden sich im Programmiereditor bei einer Anweisung der S7-Kommunikation oder der Open User Communication. Doppelklicken Sie auf das Diagnosesymbol der Anweisung (Stethoskop).

Aufbau des Bereichs "Diagnose > Verbindungsinformation" des Inspektorfensters

Voraussetzung dafür, dass der Inhalt des Registers "Verbindungsinformation" befüllt ist, ist eine bestehende Online-Verbindung zu mindestens einem Endpunkt der betrachteten Verbindung.

Bei selektierter Baugruppe (Netzsicht) enthält das Register die folgende Gruppe:

- Verbindungsressourcen (bei S7-1200 und S7-1500)

Bei selektierter Verbindung (Verbindungstabelle) enthält es die folgenden Gruppen:

- Verbindungsdetails
- Adressdetails der Verbindung (bei S7-1200 und S7-1500)

2.2.3.2 Online-Verbindungsressourcen bei S7-1200 ermitteln

Wo ermitteln Sie die Online-Verbindungsressourcen?

Die Online-Verbindungsressourcen entnehmen Sie der Gruppe "Verbindungsressourcen". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters. Sie wird nur dann angezeigt, wenn Sie in der Netzsicht eine Baugruppe ausgewählt haben, zu der eine Online-Verbindung besteht.

Anzahl von Verbindungsressourcen

- Maximale Anzahl: Gibt die maximal mögliche Anzahl verfügbarer Verbindungsressourcen der Baugruppe an.
- Nicht belegt: Gibt an, wie viele Verbindungsressourcen noch nicht belegt sind. Wenn für bestimmte Kommunikationsarten bereits Verbindungsressourcen reserviert sind, können die nicht belegten Verbindungsressourcen nicht immer für beliebige Verbindungstypen verwendet werden.

Reservierte und aktuell belegte Verbindungsressourcen

Für die im Folgenden genannten Kommunikationsarten werden die von der Baugruppe reservierten und aktuell belegten Verbindungsressourcen angezeigt.

Kommunikationsart	Bedeutung
PG-Kommunikation	Ressourcen für Verbindungen zwischen der Baugruppe und PGs (z. B. für den Verbindungsaufbau aus der Projektnavigation, für Online-Diagnose etc.)
HMI-Kommunikation	Ressourcen für Verbindungen zwischen der Baugruppe und HMI-Geräten
Open User Communication	Ressourcen für Verbindungen von Anweisungen der Open User Communication
S7-Kommunikation	Ressourcen für projektierte S7-Verbindungen, über die durch Aufruf von Anweisungen im Anwenderprogramm Daten ausgetauscht werden können
Sonstige Kommunikation	Gibt weitere belegte Verbindungsressourcen an, für die keine Verbindungsressourcen reserviert sind

2.2.3.3 Online-Verbindungsressourcen bei S7-1500 ermitteln

Wo ermitteln Sie die Online-Verbindungsressourcen?

Die Online-Verbindungsressourcen entnehmen Sie der Gruppe "Verbindungsressourcen". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters. Sie wird nur dann angezeigt, wenn Sie in der Netzsicht eine Baugruppe ausgewählt haben, zu der eine Online-Verbindung besteht.

Beschreibung der detaillierten Anzeige der Verbindungsressourcen

Zur detaillierten Anzeige der Verbindungsressourcen gehören u. a.:

- die Anzahl der zur Verfügung stehenden Verbindungsressourcen
- die Anzahl der projektierten Verbindungsressourcen
- die Anzahl der noch zur Verfügung stehenden Verbindungsressourcen.

Deren Beschreibung finden Sie hier .

2.2.3.4 Verbindungsdetails ermitteln

Wo ermitteln Sie die Verbindungsdetails?

Die Verbindungsdetails entnehmen Sie der Gruppe "Verbindungsdetails". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

Wann ist die Gruppe "Verbindungsdetails" befüllt?

Damit die Gruppe "Verbindungsdetails" des Registers "Verbindungsinformation" befüllt ist, müssen folgende Voraussetzungen erfüllt sein:

- Es besteht eine Online-Verbindung zum Endpunkt der betrachteten Verbindung.
- Sie haben in der Verbindungstabelle eine Zeile markiert.

Aufbau der Gruppe "Verbindungsdetails"

Die Gruppe "Verbindungsdetails" besteht aus folgenden Elementen:

- Lokale ID (hex)
- Verbindungstyp (bei S7-1200 und S7-1500)
- Protokoll
- Verbindungsstatus: Symbol und Beschreibung
- Details
- Letzte Zustandsänderung (nur bei S7-300 und S7-400)

2.2.3.5 Adressdetails einer Verbindung ermitteln

Wo ermitteln Sie die Adressdetails einer Verbindung?

Die Adressdetails einer Verbindung entnehmen Sie der Gruppe "Adressdetails der Verbindung". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

Bei welchen CPUs ist die Gruppe "Adressdetails der Verbindung" vorhanden?

Die Gruppe "Adressdetails der Verbindung" des Registers "Verbindungsinformation" gibt es bei S7-1200- und S7-1500-CPUs.

Wann ist die Gruppe "Adressdetails der Verbindung" befüllt?

Damit die Gruppe "Adressdetails der Verbindung" des Registers "Verbindungsinformation" befüllt ist, müssen folgende Voraussetzungen erfüllt sein:

- Es besteht eine Online-Verbindung zu den Endpunkten der betrachteten Verbindung.
- Sie haben in der Verbindungstabelle eine Zeile markiert.

Aufbau der Gruppe "Adressdetails der Verbindung"

Für die beiden Kommunikationspartner werden die für den Verbindungstyp relevanten Adressdetails angegeben.

2.3 Systemdiagnose mit 'Report System Errors'

2.3.1 Einführung zur Systemdiagnose mit "Report System Errors"

Einführung

Hardware-Komponenten und Fremdgeräte (Slaves, deren Eigenschaften durch ihre GSD-Datei bestimmt werden) können bei Eintritt eines Systemfehlers Aufrufe von Organisationsbausteinen auslösen.

Beispiel: Bei Drahtbruch kann durch eine diagnosefähige Baugruppe der OB 82 aufgerufen werden.

Die Hardware-Komponenten stellen Informationen zum aufgetretenen Systemfehler zur Verfügung. Die Startereignisinformationen, d. h. die Lokaldaten des zugeordneten OB (enthalten u. a. den Datensatz 0) geben allgemeine Auskunft über Ort (z. B. logische Adresse der Baugruppe) und Art (z. B. Kanalfehler oder Baugruppenfehler) des Fehlers.

Darüber hinaus kann über zusätzliche Diagnoseinformationen (Lesen von Datensatz 1 mit der Anweisung "RDSYSST" bzw. Lesen des Diagnosetelegramms von DP-Normslaves mit der Anweisung "DPNRM_DG") der Fehler genauer spezifiziert werden: z. B. Kanal 0 oder 1, Drahtbruch oder Messbereichsüberschreitung.

Die Systemdiagnose mit "Report System Errors" (RSE) bietet Ihnen für S7-300/400 PLCs, ET200S, ET200Pro und Software-SPS eine komfortable Möglichkeit, diese Diagnoseinformationen auszuwerten und in Form von Meldungen anzeigen zu lassen. Die hierfür notwendigen Bausteine und Meldetexte werden in den Eigenschaften der jeweiligen PLC angelegt. Sie müssen die erzeugten Bausteine nur in die CPU laden und, falls gewünscht, die Texte in angeschlossene HMI-Geräte transferieren.

Um Diagnoseereignisse beispielsweise auf einem HMI-Gerät oder über einen Webserver grafisch auszugegeben, können Sie einen oder mehrere Status-DBs generieren lassen. Diese Status-DBs werden von den Systemdiagnosebausteinen aktualisiert und enthalten dann Informationen zum aktuellen Zustand des Systems.

2.3.2 Grundlagen zur Systemdiagnose

Über die Systemdiagnose mit "Report System Errors" können Sie Bausteine erzeugen, die Fehler im System analysieren und Meldungen mit einer textuellen Fehlerbeschreibung und dem Fehlerort erzeugen.

Diese Meldungen werden pro meldefähiger Komponente (z. B. Kanalfehler, Baugruppenträgerfehler) definiert und sind auf 255 Meldungen pro meldefähiger Komponente begrenzt. Wenn diese Grenze überschritten wird, erhalten Sie eine entsprechende Meldung.

Allgemeine Einstellungen, die Sie in der Systemdiagnose vornehemen, bzw. ändern, werden zusammen mit dem Projekt gespeichert. Der Einfluss auf die Systemdiagnose wirkt sich erst nach dem Generieren der Bausteine, dem Übersetzen der Hardware-Konfiguration bzw. dem Laden der Konfiguration in die beteiligten Komponenten aus.

Empfohlene Vorgehensweise

Nehmen Sie die Einstellungen für das Melden von Systemfehlern und die Struktur der Meldungen vor. Legen Sie fest, welche Diagnosebausteine angelegt werden sollen und konfigurieren Sie die Status-DBs. Generieren Sie die Bausteine und laden Sie die Projektierung, um die Änderungen zu übernehmen.

Detaillierte Informationen erhalten Sie in den folgenden Kapiteln.

Hinweis

Bei der Verwendung der Systemdiagnose kann sich das Systemverhalten der Anlage im Fehlerfall ändern. Beispielsweise geht die CPU unter Umständen nicht in den Betriebszustand "STOP", wie es ohne die Systemdiagnose der Fall ist.

Stellen Sie daher sicher, dass alle Schutzmechanismen der Anlage funktionsfähig sind.

2.3.3 Baugruppen mit eingeschränkten Diagnosemöglichkeiten

Einleitung

Die Komponenten der S7-300-Stationen, S7-400-Stationen, PROFINET IO-Devices, DP-Slaves und WinAC werden von "Report System Errors" unterstützt, sofern sie Funktionen wie Diagnosealarm, Ziehen/Stecken-Alarm und kanalspezifische Diagnose unterstützen.

Nicht unterstützte Komponenten

Folgende Komponenten werden von "Report System Errors" nicht unterstützt:

- M7-Konfigurationen
- PROFIBUS-DP-Konfigurationen über DP-Masteranschaltung (CP 342-5 DP) in S7-300 Stationen
- PROFINET IO-Devices über einen externen Controller (CP 343-1 Advanced) in S7-300-Stationen

Im Falle eines Wiederanlaufes ist noch zusätzlich zu beachten, dass es zu fehlenden Alarmmeldungen kommen kann. Dieses Verhalten beruht darauf, dass der Meldequittierspeicher der CPU bei einem Wiederanlauf nicht gelöscht wird, aber "Report System Errors" die internen Daten zurücksetzt. Die Modul- oder Kanalfehler, die vor dem Anlauf oder während eines Ausfalls auftreten, werden nicht alle gemeldet.

Pro Modul werden maximal 8 Kanalfehler gemeldet.

Hinweis

Falls Sie einen CP 443-5 einsetzen und dieser im Betriebszustand STOP ist, wird beim Anlauf kein Mastersystemausfall gemeldet.

PROFIBUS-DP

Für alle PROFIBUS DP-Slaves werden die Stationsfehler (Ausfall/Wiederkehr) mit einer Klartextmeldung angezeigt.

Für alle PROFIBUS DP-Slaves wird die herstellerspezifische Diagnose mit folgenden Einschränkungen unterstützt:

- Es werden nur V1-Slaves unterstützt. Dies sind Slaves, in deren GSD-Datei der Eintrag "DPV1_Slave=1" gesetzt ist.
- Der DP-Alarmmodus für diesen Slave muss auf "DPV0" eingestellt sein.

Diagnose-Repeater

Die Meldungen des Diagnose-Repeaters werden im DPV0-Modus als Klartext ausgegeben. Die Texte werden aus der GSD-Datei ausgelesen.

PROFINET-IO

Nachfolgend finden Sie die Diagnose von den verschiedenen PROFINET IO-Devices, die von "Report System Errors" unterstützt werden.

Fehler, die während des Betriebs (CPU in RUN) auftreten:

- Devicefehler (Ausfall, Wiederkehr) werden unterstützt
- Modulfehler und Submodulfehler (Modul/Submodul gezogen, falsches Modul/Submodul, kompatibles Modul/Submodul) werden unterstützt
- Kanalfehler werden unterstützt und erweiterte Kanalfehlerinformationen werden ausgewertet

AS-Interface

Für AS-Interface-Slaves wird eine Meldung verschickt, falls der Istausbau ungleich dem Sollausbau ist.

Shared Devices

"Report System Errors" wertet die Information, ob ein (Sub-)Modul als Shared Device projektiert wurde, aus. Da "Report System Errors" immer die Sicht auf eine CPU hat, werden nur die (Sub-)Module, die für diese CPU Vollzugriff gesetzt haben für die Diagnose berücksichtigt. Die als "nicht zugeordnet" projektieren (Sub-)Module werden nicht betrachtet.

2.3.4 Diagnosebausteine für das Melden von Systemfehlern

Nachdem Sie die Einstellungen für das Melden von Systemfehlern vorgenommen haben, werden die notwendigen Bausteine (FB mit zugeordnetem Instanz-DB und ein oder mehrere Global-DB(s) und einem FC, je nach Einstellung auch noch nicht vorhandenen OBs) und Status-DBs beim nächsten Übersetzen der Hardware-Konfiguration angelegt.

Folgende Bausteine werden angelegt:

- Diagnose-FB (voreingestellt: FB 49),
- Instanz-DB für den Diagnose-FB (voreingestellt: DB 49),
- Global-DB (voreingestellt: DB 50),
- Diagnose-FC (voreingestellt: FC 49),
- Fehler-OBs (werden automatisch angelegt, wenn die CPU diese Funktion unterstützt),
- Zyklische OBs (falls Sie diese Option in der Gruppe "OB-Konfiguration" angewählt haben),
- Optionale Status-DBs

Die angelegten Bausteine werden in der Projektnavigation unter "Programmbausteine > Systembausteine > Report System Errors" abgelegt (außer OBs).

Hinweis

Die eingestellten Bausteinnummern der erzeugten Systemdiagnosebausteine dürfen sich nicht mit vorhandenen Bausteinnummern überschneiden. Es darf auch keine Überschneidung mit Nummern von Bausteinen geben, die in Bibliotheken liegen.

Prüfen Sie bei Problemen mit der Generierung der Bausteine ob die voreingestellten Nummern (FB 49, DB 49, FC 49, DB 50) geändert wurden. Falls dies der Fall ist, versuchen sie die Generierung mit den voreingestellten Bausteinnummern zu wiederholen oder verwenden Sie Nummern, bei denen Sie sicher sind, dass diese nicht anderweitig verwendet werden.

Siehe auch

Eigenschaften der Bausteine (Seite 3545)

2.3.5 Eigenschaften der Bausteine

Diagnosebausteine

Die angelegten Diagnosebausteine (FB mit zugeordnetem Instanz-DB und ein oder mehrere Global-DB(s) und einem FC) werten die Lokaldaten des Fehler-OBs aus und lesen eventuell zusätzliche Diagnoseinformationen der Fehler auslösenden Hardware-Komponente.

Sie haben folgende Eigenschaften:

- Erstellsprache RSE (Report System Errors) (gilt ebenfalls für die oben genannten Bausteine)
- Know-how-protected (gilt ebenfalls für die oben genannten Bausteine)
- Verzögern während der Laufzeit ankommende Alarme

Status-DBs

Die Status-DBs dienen als Schnittstelle für die Diagnosebausteine und ermöglichen es, auf einem HMI-Gerät oder einem Webserver Diagnoseereignisse grafisch auszugeben.

Siehe auch

Einstellungen für Systemdiagnosebausteine (Seite 3552)

2.3.6 Unterstützte Fehler-OBs

Die folgenden Fehler-OBs werden, soweit sie auch von der projektierten CPU unterstützt werden, automatisch angelegt:

OB	Aufruf des Diagnose-FBs	Beschreibung
OB 70 (Peripherie-Redundanzfehler)	möglich	Diesen OB gibt es nur bei H-CPUs.
OB 72 (CPU-Redundanzfehler)	möglich	Diesen OB gibt es nur bei H-CPUs.
OB 73 (Kommunikations-Redundanzfehler)	möglich	Diesen OB gibt es nur bei wenigen H-CPUs.
OB 81 (Stromversorgungsfehler)	möglich	
OB 82 (Diagnosealarm-OB)	möglich	
OB 83 (Ziehen/Stecken-Alarm)	möglich	
OB 85 (Programmablauffehler)	nicht möglich	Falls 'RSE' diesen OB beim Generieren der Diagnose-Bausteine anlegt, werden zusätzlich Netzwerke eingefügt, die folgenden Programmablauf realisieren: • Bei Fehlern der Prozessabbildaktualisierung (z. B. Ziehen der Baugruppe) wird der STOP der CPU verhindert, damit Diagnose-FB in OB 83 bearbeitet werden kann. Eine eventuelle Einstellung "CPU-STOP" nach Meldung von "RSE" wird im OB 83 wirksam. • Bei allen anderen Fehlerereignissen von OB 85 geht die CPU in STOP.
OB 86 (Ausfall eines Erweiterungsbaugruppenträgers, eines DP-Mastersystems oder eines dezentralen Peripheriegerätes)	möglich	

Wenn die Fehler-OBs bereits existieren...

Vorhandene Fehler-OBs werden nicht überschrieben. Gegebenenfalls wird der Aufruf des Diagnose-FBs in den vorhandenen OB angehängt.

Wenn die Konfiguration Dezentrale Peripheriegeräte umfasst...

Für die Auswertung von Fehlern in der dezentralen Peripherie ruft der angelegte FB automatisch die Anweisung "DPNRM_DG" auf (Diagnosedaten der DP-Slaves lesen). Um diese Funktion sicherzustellen, muss der angelegte FB entweder nur im OB 1 oder in einem Weckalarm-OB mit kurzem Zeittakt und in den Anlauf-OBs aufgerufen werden.

Hinweis

Bitte beachten Sie Folgendes:

- Durch den von der Systemdiagnose angelegten OB 85 bei dem Fehlerereignis "Fehler beim Aktualisieren des Prozessabbilds" erfolgt kein STOP der CPU.
- OB 85 wird von der CPU zusätzlich auch in den folgenden Fehlerfällen aufgerufen.
  - "Fehlerereignis für einen nicht geladenen OB"
  - "Fehler beim Aufruf bzw. Zugriff auf einen nicht geladenen Baustein"
    In diesen Fehlerfällen erfolgt mit dem von der Systemdiagnose erzeugten OB 85 weiterhin ein STOP der CPU wie vor der Verwendung der Systemdiagnose.

2.3.7 Diagnosestatus-DB

Innerhalb der Systemdiagnose besteht die Möglichkeit zum Anlegen eines Status-DBs. Dieser Status-DB wird von den Systemdiagnosebausteinen aktualisiert und enthält dann Informationen zum aktuellen Zustand des Systems.

Schnittstelle für den Diagnosestatus-DB

Der erzeugte Datenbaustein (Default: DB 127) bietet die Möglichkeit, den Systemzustand einer projektierten Komponente sowie gegebenenfalls aller ihr unterlagerten Komponenten abzufragen.

Dieser Datenbaustein wird für die Unterstützung der Systemdiagnose über den Webserver oder auch den Diagnose Viewer benötigt. Standardmäßig ist er aktiviert.

Hinweis

Beachten Sie bitte, das die Systemdiagnose sowohl im Webserver oder Diagnose Viewer als auch im Inspektorfenster der PLC aktiviert sein muss, um die Diagnosedaten des Status-DBs darstellen zu können.

Nach dem Neustart einer CPU mit Webserver wird der Baugruppenzustand mit Verzögerung angezeigt. Um die Wartezeit zu verkürzen, können Sie den RSE-Diagnosebaustein in einem Weckalarm-OB mit kürzerem Zeitintervall aufrufen.

Aufbau des Diagnosestatus-DBs

Adresse	Name	Datentyp	Beschreibung
+0	Directory		
0	D_Version	WORD	Version, die die Systemdiagnose unterstützt
2	D_pGlobalState	WORD	Byte-Offset bis zum Anfang des Teils "GlobalState"
4	D_pQuery	WORD	Byte-Offset bis zum Anfang des Teils "Query"
6	D_pComponent	WORD	Byte-Offset bis zum Anfang des Teils "Component"
8	D_pError	WORD	Byte-Offset bis zum Anfang des Teils "Error"
10	D_pState	WORD	Byte-Offset bis zum Anfang des Teils "State"
12	D_pAlarm	WORD	Byte-Offset bis zum Anfang des Teils "Alarm"
14	D_pSubComponent	WORD	Byte-Offset bis zum Anfang des Teils "Subcomponent"
+16	GlobalState		
0	G_EventCount	WORD	ID des letzten Ereignisses (Zähler)
2.0	G_StartReporting	BOOL	Anlaufauswertung läuft
+20	Query		
0	Q_ClientID_User	DWORD	ID des Clients; bitte verwenden Sie hier einen Wert zwischen 1 und 255. Stellen Sie sicher, dass unterschiedliche Clients unterschiedliche IDs verwenden.
4	Q_ClientID_Intern	DWORD	ID des Clients (intern)
8.0	Q_WithSubComponent	BOOL	Mit/Ohne Status der untergeordneten Komponenten (langsamer)
8.1	Q_SubComponentAlarm	BOOL	AS-iMaster gibt AS-iSlave Alarme zurück
8.2	Q_Reserved2	BOOL	Reserviert
8.3	Q_Reserved3	BOOL	Reserviert
8.4	Q_Reserved4	BOOL	Reserviert
8.5	Q_Reserved5	BOOL	Reserviert
8.6	Q_Reserved6	BOOL	Reserviert
8.7	Q_Reserved7	BOOL	Reserviert
9.0	Q_Start	BOOL	Abfrage starten
10	Q_Error	BYTE	Interner Fehler bei der Abfrage
11	Q_Reserved8	BYTE	Reserviert
+32	Component		
0	C_AddressMode	BYTE	Adressierungsmodus der Baugruppe
1	C_Reserved1	BYTE	Reserviert
2	C_ComponentID	WORD	Hardware-ID der Komponente (intern)
+36	Error		

Adresse	Name	Datentyp	Beschreibung
0	E_ErrorNo	WORD	Index des angeforderten/tatsächlichen Fehlers
2.0	E_LastError	BOOL	Wird gesetzt, wenn E_ErrorNo ungleich 0 ist. Wert TRUE, falls E_ErrorNo der Index des letzten Fehlers ist, ansonsten FALSE
+40	State		
0	S_Hierarchy	BYTE	Reserviert
1	S_Periphery	BYTE	Reserviert
2.0	S_SupFault	BOOL	Die Komponente ist nicht erreichbar
2.1	S_NotAvailable	BOOL	Die Komponente ist nicht vorhanden
2.2	S_Faulty	BOOL	Die Komponente ist gestört; der Teil "Alarm" ist nicht leer
2.3	S_MoreErrors	BOOL	Es liegen mehr Fehler an, als die Systemdiagnose speichern kann
2.4	S_Maintenance1	BOOL	Ein Wartungsbedarf liegt an
2.5	S_Maintenance2	BOOL	Eine Wartungsanforderung liegt an
2.6	S_Deactivated	BOOL	Die Komponente wurde deaktiviert *)
2.7	S_Reserved2	BOOL	Reserviert
3.0	S_SubFault	BOOL	Eine untergeordnete Komponente ist gestört
3.1	S_SubMaintenance1	BOOL	Ein Wartungsbedarf liegt an einer untergordneten Komponente an
3.2	S_SubMaintenance2	BOOL	Eine Wartungsanforderung liegt an einer untergordneten Komponente an
3.3	S_SubDeactivated	BOOL	Eine untergeordnete Komponente ist deaktiviert
3.4	S_Reserved4	BOOL	Reserviert
3.5	S_Reserved5	BOOL	Reserviert
3.6	S_Reserved6	BOOL	Reserviert
3.7	S_Reserved7	BOOL	Reserviert
4	S_TIAMS	DWORD	Maintenance State der Komponente
8	S_TIAMSChannelExist	DWORD	Maintenance State: Projektierte Kanäle
12	S_TIAMSChannelOK	DWORD	Maintenance State: Gestörte Kanäle
16	S_ChannelCount	WORD	Anzahl Kanäle; nur gültig, wenn Q_WithSubComponent gesetzt ist
18.0	S_ChannelVector	ARRAY [0..255] BOOL	Liste der betroffenen Kanäle; nur gültig, wenn Q_WithSubComponent gesetzt ist
+90	Alarm		
0	A_ComponentID	WORD	Hardware-ID der Komponente (intern)
2	A_TextID1	WORD	ID des ersten Fehlertexts
4	A_TextLexikonID1	WORD	ID des ersten Fehlertextlexikons
6	A_HelpTextLexikonID1	WORD	ID des ersten Hilfetextlexikons
8	A_MapTextID	WORD	ID des ersten Fehlertexts (HMI)
10	A_MapHelpTextID	WORD	ID des ersten Hilfetexts (HMI)
12	A_TextID2	WORD	ID des zweiten Fehlertexts

Adresse	Name	Datentyp	Beschreibung
14	A_TextLexikonID2	WORD	ID des zweiten Fehlertextlexikons
16	A_HelpTextLexikonID2	WORD	ID des zweiten Hilfetextlexikons
18	A_MapTextID2	WORD	ID des zweiten Fehlertexts (HMI)
20	A_MapHelpTextID2	WORD	ID des zweiten Hilfetexts (HMI)
22	A_AlarmID	DWORD	Meldenummer
26	A_ValueCount	WORD	Anzahl der belegten weiteren Bytes (12)
28	A_AssociatedValue	ARRAY [1..n] WORD	Begleitwerte der Meldung n = A_ValueCount / 2 (= 6)
+130	SubComponent		
0	U_SubComponentCount	WORD	Anzahl der untergeordneten Komponenten
2	U_SubComponentFault	ARRAY [1..n] BYTE	Liste der untergordneten Komponenten "n" hängt von der Konfiguration ab **)

*) Wenn die Komponente dekativiert wurde, wird der Index des angeforderten/tatsächlichen Fehlers nicht verändert und "E_LastError" wird auf "true" gesetzt. Außerdem wird der Variablenbereich des Alarms nicht ausgefüllt.

**) Die Liste der untergeordneten Komponenten ist nur gültig, wenn Q_WithSubComponent gesetzt ist. Pro projektierter Komponente enthält das ARRAY ein Statusbyte. Für einen Master enthält das ARRAY den Status der projektierten Stationen, aufsteigend sortiert nach Stations-ID. Für eine Station enthält das ARRAY den Status der projektierten Steckplätze, aufsteigend sortiert nach Steckplatznummer. Dieses Feld kann maximal 4096 Einträge enthalten (für ein IO-System); nur die tatsächliche maximale Größe wird angezeigt.
Das Statusbyte pro untergeordnete Komponente ist folgendermaßen definiert:
Bit 0 = SubFault: die Komponente ist nicht erreichbar
Bit 1 = Fault: die Komponente ist nicht verfügbar oder fehlerhaft
Bit 2 = Maintenance1: die Komponente hat eine Wartung gemeldet
Bit 3 = Maintenance2: die Komponente hat eine Wartung gemeldet
Bit 4 = Deactivated: Die Komponente wurde deaktiviert
Bit 5 = SubFault: eine untergeordnete Komponente ist gestört
Bit 6 = SubMaintenance1: eine untergeordnete Komponente hat eine Wartung gemeldet
Bit 7 = SubMaintenance2: eine untergeordnete Komponente hat eine Wartung gemeldet

2.3.8 Einstellungen der Systemdiagnose anzeigen

Voraussetzung

Sie befinden sich im Ordner "Gemeinsame Daten" in der Projektnavigation.

Vorgehen

Um die Einstellungen der Systemdiagnose anzuzeigen und zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Eintrag "Systemdiagnoseeinstellungen".
2. In der Systemdiagnose können Sie sich die Einstellungen anzeigen lassen und bearbeiten.

2.3.9 Einstellungen

2.3.9.1 Allgemeine Einstellungen

Systemdiagnose für diese PLC aktivieren

Diese Option ist standardmäßig deaktiviert.

Hinweis

Wenn diese Option deaktiviert ist, werden auch im Webserver und Diagnose Viewer keine Diagnosedaten ausgelesen, auch wenn die Option dort aktiviert ist.

Wenn Sie diese Option in einer PLC deaktivieren, in der sie zuvor aktiviert war, werden alle Diagnosedaten beim nächsten Übersetzen der Hardware-Konfiguration gelöscht.

2.3.9.2 Meldungseinstellungen

Für die nachfolgend aufgelisteten Meldungskategorien können Sie definieren, ob eine Meldung ausgegeben werden soll und ob diese quittierpflichtig ist oder nicht:

- Fehler
- Wartungsanforderung
- Wartungsbedarf
- Info

Aktivieren Sie hierzu die gewünschten Optionsfelder in den Spalten "Meldung" und "Quittierung".

In der Spalte "Meldeklasse" wird die eingestellte Meldeklasse angezeigt.

2.3.9.3 Einstellungen für die Diagnoseunterstützung

Mit Hilfe der Diagnoseunterstützung werden Diagnosedaten in einen speziellen Status-DB durch den Diagnose-FB zur Verfügung gestellt. Dieser kann dann grafisch auf beispielsweise einem HMI-Gerät dargestellt werden.

Status-DB

Aktivieren Sie den Status-DB, um den aktuellen Systemstatus projektierter Komponenten über diesen Datenbaustein auszulesen.

Der folgende Status-DB ist standardmäßig vorgegeben:

Baustein	Name	Nummer
Diagnosestatus-DB	RSE_DIAGNOSTIC_STATUS_DB	127

Sie können für diesen Baustein jedoch einen anderen Namen und/oder eine andere Nummer vergeben, soweit diese nicht bereits verwendet werden.

Falls die Funktionalität "Webserver auf dieser Baugruppe aktivieren" in den Eigenschaften der CPU aktiviert ist, muss der Diagnosestatus-DB zwingend aktiviert sein. Standardmäßig wird der DB 127 erzeugt, wenn die CPU den Webserver unterstützt und dieser auch in den Eigenschaften der CPU aktiviert ist.

Erweiterte Diagnoseeinstellungen

Wichtige Voraussetzung für die Abfrage ist, dass die PLC die Anweisung "D_ACT_DP" enthält!

Aktivieren Sie die Option "Abfrage auf Status "aktiviert/deaktiviert" nach PLC-Anlauf", wenn Sie möchten, dass der Status von Slaves abgefragt werden soll.

Aktivieren Sie die Option "Meldung verschicken, wenn der Status von/nach aktiviert oder deaktiviert wechselt", wenn Sie möchten, dass eine Meldung ausgegeben werden soll, wenn ein Statuswechsel eingetreten ist. Um diese Option zu nutzen, müssen Sie die Anweisung "D_ACT_DP" im Modus 3/4 betreiben und den Aufruf des Diagnose-FB im OB 86 in der Gruppe "OB-Konfiguration" ausgewählt haben.

2.3.9.4 Einstellungen für Systemdiagnosebausteine

Für die Systemdiagnosebausteine sind folgende Werte standardmäßig vorgegeben:

Baustein	Name	Nummer
Diagnose-FB	RSE_FB	49
Diagnose-DB	RSE_DB	49
erster Global-DB	RSE_GLOBAL_DB	50
Diagnose-FC	RSE_FC	49

Sie können für diese Bausteine jedoch andere Nummern und/oder Namen vergeben, soweit diese nicht bereits verwendet werden.

2.3.10 Systemdiagnoseeinstellungen exportieren und importieren

2.4 Systemdiagnose für S7-1500 PLCs

2.4.1 Einführung

Einführung

Hardware-Komponenten und Fremdgeräte (Slaves, deren Eigenschaften durch ihre GSD-Datei bestimmt werden) können bei Eintritt eines Systemfehlers eine Systemreaktion auslösen. Die Hardware-Komponenten stellen Informationen zum aufgetretenen Systemfehler zur Verfügung.

Die Systemdiagnose bietet Ihnen für S7-1500 PLCs eine komfortable Möglichkeit, diese Diagnoseinformationen auszuwerten und in Form von Meldungen anzeigen zu lassen.

Darüberhinaus kann über zusätzliche Diagnoseinformationen der Fehler genauer spezifiziert werden, z. B. Kanal "Schalter_Oben", Drahtbruch oder Messbereichsüberschreitung.

2.4.2 Grundlagen zur Systemdiagnose

Über die Systemdiagnose können Sie Fehler im System analysieren und Meldungen mit einer textuellen Fehlerbeschreibung und dem Fehlerort erzeugen.

Diese Meldungen werden pro meldefähiger Komponente (z. B. Kanalfehler, Baugruppenträgerfehler) definiert.

Empfohlene Vorgehensweise

Nehmen Sie die Einstellungen für die Systemdiagnose und die Struktur der Meldungen vor.

Detaillierte Informationen erhalten Sie in den folgenden Kapiteln.

2.4.3 Einstellungen der Systemdiagnose anzeigen

Voraussetzung

Sie befinden sich im Ordner "Gemeinsame Daten" in der Projektnavigation.

Vorgehen

Um die Einstellungen der Systemdiagnose anzuzeigen und zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Eintrag "Systemdiagnoseeinstellungen".
2. In der Systemdiagnose können Sie sich die Einstellungen anzeigen lassen und bearbeiten.

2.4.4 Einstellungen

2.4.4.1 Allgemeine Einstellungen

Systemdiagnose für diese PLC aktivieren

Diese Option ist standardmäßig aktiviert und kann nicht deaktiviert werden.

Hinweis

Wenn diese Option deaktiviert ist, werden auch im Webserver und Diagnose Viewer keine Diagnosedaten ausgelesen, auch wenn die Option dort aktiviert ist.

Wenn Sie diese Option in einer PLC deaktivieren, in der sie zuvor aktiviert war, werden alle Diagnosedaten beim nächsten Übersetzen der Hardware-Konfiguration gelöscht.

Netzwerkfehler nicht als Fehler sondern als Wartungsanforderung melden

Wenn Sie diese Option aktivieren, werden Netzwerkfehler als 'Wartungsanforderung' gemeldet.

2.4.4.2 Meldungseinstellungen

Für die nachfolgend aufgelisteten Meldungskategorien können Sie definieren, ob eine Meldung ausgegeben werden soll und ob diese quittierpflichtig ist oder nicht:

- Fehler
- Wartungsanforderung
- Wartungsbedarf
- Info

Aktivieren Sie hierzu die gewünschten Optionsfelder in den Spalten "Meldung" und "Quittierung".

In der Spalte "Meldeklasse" wird die eingestellte Meldeklasse angezeigt.

2.4.5 Systemdiagnoseeinstellungen exportieren und importieren

2.4.5.1 Grundlagen zum Exportieren und Importieren von Systemdiagnoseeinstellungen

Beschreibung

Es ist möglich, die Systemdiagnoseeinstellungen eines Projekts zu exportieren und in ein anderes Projekt zu importieren. Dies hat den Vorteil, dass diese Einstellungen nicht für jedes Projekt separat angelegt werden müssen.

Die Systemdiagnoseeinstellungen werden in eine Datendatei (*.dat) gespeichert.

2.4.5.2 Systemdiagnoseeinstellungen exportieren

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um die Systemdiagnoseeinstellungen zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Ordner "Gemeinsame Daten" und doppelklicken Sie auf den Eintrag "Systemdiagnoseeinstellungen".
2. Klicken Sie in der Systemdiagnose auf das Symbol "Systemdiagnoseeinstellungen exportieren" in der Funktionsleiste. Der Dialog "Speichern unter" wird geöffnet.
3. Wählen Sie den Pfad aus, in dem die Exportdatei abgelegt werden soll.
4. Klicken Sie auf die Schaltfläche "Speichern".

Ergebnis

Am ausgewählten Speicherort wird eine .dat-Datei angelegt.

2.4.5.3 Systemdiagnoseeinstellungen importieren

Voraussetzung

Das Projekt, in das die Systemdiagnoseeinstellungen importiert werden sollen, ist geöffnet.

Vorgehen

Um die Systemdiagnoseeinstellungen zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Ordner "Gemeinsame Daten" und doppelklicken Sie auf den Eintrag "Systemdiagnoseeinstellungen".
2. Klicken Sie in der Systemdiagnose auf das Symbol "Systemdiagnoseeinstellungen importieren" in der Funktionsleiste. Der Dialog "Öffnen" wird geöffnet.
3. Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist.
4. Wählen Sie die gewünschte .dat-Datei aus.
5. Klicken Sie auf die Schaltfläche "Öffnen".

Hinweis

Das Symbol "Importieren" ist gegraut, wenn das geöffnete Projekt schreibgeschützt ist (z. B. Referenzprojekt).

Wenn es während des Imports zu Fehlern kommt, wird eine Meldung angezeigt und der Vorgang wird abgebrochen.

Ergebnis

Die Systemdiagnoseeinstellungen werden angezeigt.

2.5 Zusätzliche Diagnosemöglichkeiten nutzen

2.5.1 Hardware diagnostizieren

2.5.1.1 Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen

Allgemeine Eigenschaften und anlagenrelevante Informationen einer Baugruppe anzeigen

Wo befinden sich die gesuchten Informationen?

Die allgemeinen Eigenschaften und anlagenrelevanten Informationen einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Allgemein".

Aufbau der Gruppe "Allgemein"

Die Gruppe "Allgemein" besteht aus folgenden Bereichen:

- Baugruppe
- Baugruppeninformation
- Herstellerinformation

Bereich "Baugruppe"

Dieser Bereich zeigt bei S7-300 und S7-400 zusätzlich das folgende Datum der Baugruppe an:

- Version des Bootloaders

Bereich "Baugruppeninformation"

Dieser Bereich zeigt bei S7-300 und S7-400 zusätzlich die folgenden Daten der Baugruppe an, die Sie bei der Konfiguration der Hardware projektiert haben:

- Stationsname (Diesen haben Sie in der Netzsicht vergeben.)
- Anlagenkennzeichen
  Wenn eine CPU, ein Slave, ein IO-Device bzw. eine Baugruppe in der Station die Parametrierung eines anlagenweit eindeutigen Anlagenkennzeichens unterstützt, wird es hier angezeigt (Mit dem Anlagenkennzeichen werden Teile der Anlage eindeutig nach funktionalen Gesichtspunkten gekennzeichnet. Es ist gemäß IEC 1346-1 hierarchisch aufgebaut.). Damit ist z. B. das Identifizieren einer CPU aus dem Anwenderprogramm heraus möglich. Das Anlagenkennzeichen können Sie mit der Anweisung "RDSYSST" auslesen.
- Ortskennzeichen
  Wenn eine CPU, ein Slave, ein IO-Device bzw. eine Baugruppe in der Station die Parametrierung eines Ortskennzeichens unterstützt, wird es hier angezeigt (Das Ortskennzeichen ist Teil des Betriebsmittelkennzeichens, das die genaue Lage z. B. einer Messstelle innerhalb einer prozesstechnischen Anlage kennzeichnet, ähnlich einer Straße in einem Stadtplan.).

Bereich "Herstellerinformation"

Dieser Bereich zeigt bei S7-300 und S7-400 zusätzlich das folgende Datum der Baugruppe an:

- Urheberrechtseintrag

Kommunikationseigenschaften einer Baugruppe anzeigen

Wo befinden sich die gesuchten Informationen?

Die Kommunikationseigenschaften einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Kommunikation".

Es handelt sich um die folgenden Eigenschaften:

- Zyklusbelastung durch Kommunikation, konfiguriert:
  Eingestellter Wert des Parameters "Zyklusbelastung durch Kommunikation", d. h. derjenige Anteil an der CPU-Verarbeitungsleistung, der im Zyklus für Kommunikation zur Verfügung steht
- Verbindungsressourcen, maximale Anzahl:
  Gibt die maximal mögliche Anzahl verfügbarer Verbindungs-Ressourcen der Baugruppe an. Jeweils eine bestimmte Anzahl der verfügbaren Verbindungs-Ressourcen ist reserviert für PG-, OP-Kommunikation und bei S7-300 (mit Ausnahme der CPU 318-2 DP) für S7-Basis-Kommunikation. Sie können jedoch darüber hinaus weitere PG- bzw. OP-Verbindungen herstellen; die Anzahl der nicht belegten Verbindungs-Ressourcen verringert sich dann entsprechend.

Schnittstellen und Schnittstelleneigenschaften einer Baugruppe anzeigen

Wo befinden sich die gesuchten Informationen?

Die Schnittstellen und Schnittstelleneigenschaften einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in den folgenden Gruppen:

- MPI-Schnittstelle
- MPI/DP-Schnittstelle
- DP-Schnittstelle
- PROFINET-Schnittstelle

Gruppe "MPI-Schnittstelle"

Diese Gruppe zeigt das folgende Datum an:

- Übertragungsgeschwindigkeit:
  Aktuell eingestellte Übertragungsrate an der MPI-Schnittstelle des Automatisierungssystems

Gruppe "MPI/DP-Schnittstelle"

Diese Gruppe zeigt die folgenden Daten an:

- Übertragungsgeschwindigkeit:
  Aktuell eingestellte Übertragungsrate an der MPI/DP-Schnittstelle des Automatisierungssystems
- DP-Betriebsart:
  Kompatibilitätsmode der auf den Schnittstellentyp "PROFIBUS" eingestellten MPI/DP-Schnittstelle. Bei Baugruppen, die die Betriebsart "DPV1" unterstützen, kann dieser die Werte "S7-kompatibel" und "DPV1" annehmen.

Hinweis

Falls Sie die MPI / DP-Schnittstelle auf den Schnittstellentyp "MPI" eingestellt haben, enthält "DP-Betriebsart" keinen gültigen Wert.

Gruppe "DP-Schnittstelle"

Diese Gruppe zeigt die folgenden Daten an:

- DP-Betriebsart:
  Kompatibilitätsmode der DP-Schnittstelle. Bei Baugruppen, die die Betriebsart "DPV1" unterstützen, kann dieser die Werte "S7-kompatibel" und "DPV1" annehmen.

Gruppe "PROFINET-Schnittstelle"

Diese Gruppe gliedert sich in folgende Bereiche:

- "Ethernet-Adresse" mit den Unterbereichen "Netzwerkverbindung" und "IP-Parameter"
- "Ports"
- "Statistik" (Dieser Bereich wird hier nicht betrachtet, da es sich hierbei nicht um aktuelle Werte von projektierbaren Baugruppeneigenschaften handelt.)

Hinweis

Die Bereiche "Ethernet-Adresse" und "Ports" wurden bereits außerhalb des Kapitels "Zusätzliche Diagnosemöglichkeiten nutzen" beschrieben, da sie für S7-1200 und S7-300/ S7-400 gelten.

Operandenbereiche, Organisationsbausteine und Systembausteine einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Die Operandenbereiche, Organisationsbausteine und Systembausteine einer CPU finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Leistungsdaten".

Gruppe "Leistungsdaten"

In der Gruppe "Leistungsdaten" werden die folgenden Informationen der zugehörigen CPU angezeigt:

- Operandenbereiche:
  Diese Tabelle gibt Auskunft über
  - den Operandentyp, z. B. Prozessabbild der Eingänge, Zähler
  - die Anzahl der Operanden oder deren Bereichsgröße
  - den verfügbaren Adressbereich der Operanden, z. B. E0.0 bis E127.7
- Organisationsbausteine:
  Alle Organisationsbausteine (OBs) samt Funktion
  Die Tabellenzeilen können Sie wie folgt sortieren: Ein Mausklick auf eine Spaltenüberschrift ordnet die Tabellenzeilen so an, dass die Inhalte dieser Spalte von oben nach unten alphabetisch geordnet sind.
- Systembausteine:
  Alle Systemfunktionen (SFCs) und Systemfunktionsbausteine (SFBs)
  Die Tabellenzeilen können Sie wie folgt sortieren: Ein Mausklick auf eine Spaltenüberschrift ordnet die Tabellenzeilen so an, dass die Inhalte dieser Spalte von oben nach unten alphabetisch geordnet sind.

Eigenschaften des Zeitsystems einer Baugruppe anzeigen

Wo befinden sich die gesuchten Funktionen?

Die Eigenschaften des Zeitsystems einer Baugruppe finden Sie in deren Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Uhrzeit einstellen".

Aufbau der Gruppe "Uhrzeit einstellen"

Die Gruppe "Uhrzeit einstellen" besteht aus folgenden Bereichen:

- Bereich zum Auslesen und Einstellen der Uhrzeit (Dieser Bereich wird hier nicht betrachtet)
- Zeitsystem

Aufbau des Bereichs "Zeitsystem"

Der Bereich "Zeitsystem" besteht aus folgenden Unterbereichen:

- "Uhr"
- "Synchronisation"

Unterbereich "Uhr"

Der Unterbereich "Uhr" besteht aus folgenden Elementen:

- Auflösung:
  Angabe der Genauigkeit, mit der die Berechnung und Ausgabe der Uhrzeit erfolgt
- Echtzeituhr:
  Anzeige, ob die CPU eine Echtzeituhr hat: "vorhanden", falls ja; "nicht vorhanden", falls nein
- Korrekturfaktor:
  Parametrierter Korrekturfaktor für die Echtzeituhr der Baugruppe. Mit dem Korrekturfaktor wird die tägliche Abweichung der Uhr ausgeglichen.

Unterbereich "Synchronisation"

Durch Synchronisationsmechanismen kann zwischen den Uhren verschiedener Teilsysteme Gleichlauf hergestellt werden. Die Einstellungen für die Uhrzeitsynchronisation werden beim Parametrieren der Baugruppe vorgenommen.

Die integrierte Echtzeituhr - sofern vorhanden - kann dabei die Uhren anderer Baugruppen synchronisieren (Master) oder sie kann selbst von den Uhren anderer Teilnehmer synchronisiert werden (Slave) oder sie nimmt nicht an der Synchronisation teil. Die Synchronisation kann im AS oder auf MPI erfolgen.

2.5.1.2 Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen

Aktuellen Zustand der LEDs und des Betriebsartenschalters einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Den aktuellen Zustand der LEDs und des Betriebsartenschalters einer CPU finden Sie in der Task Card "Online-Tools" im Anzeigebereich der Palette "CPU-Bedienpanel".

Anzeigebereich der Palette "CPU-Bedienpanel" in der Task Card "Online-Tools"

Dieser Bereich enthält die folgenden Anzeigen:

- Stationsname und CPU-Typ
- Error (entspricht der LED "SF" bei S7-300-CPUs)
- RUN (entspricht dem Betriebszustand "RUN" der CPU)
- STOP (entspricht dem Betriebszustand "STOP" der CPU)
- FORCE (entspricht der LED "FRCE" der CPU)
- Aktuelle Stellung des Betriebsartenschalters

Hinweis

Es kann vorkommen, dass bei S7-300-CPUs mit einem Betriebsartenschalter fälschlicherweise "RUN-P" angezeigt wird. In diesem Fall steht der Betriebsartenschalter auf "RUN".

Noch nicht belegte Verbindungsressourcen und belegte und reservierte Verbindungen einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Die noch nicht belegten Verbindungsressourcen und die belegten und reservierten Verbindungen einer CPU finden Sie in der Online- und Diagnosesicht im Ordner "Diagnose" in der Gruppe Kommunikation.

Der Bereich "Verbindungsressourcen" enthält die aktuellen Werte der folgenden dynamischen Baugruppeneigenschaften:

- Anzahl der noch nicht belegten Verbindungs-Ressourcen
  Diese finden Sie unter "Nicht belegt".

  Hinweis

  Wenn für bestimmte Kommunikationsarten bereits Verbindungs-Ressourcen reserviert sind, können die nicht belegten nicht immer für beliebige Verbindungstypen verwendet werden.

- PG-Kommunikation "reserviert" und "belegt":
  Anzahl der Verbindungs-Ressourcen, die für Verbindungen zwischen der Baugruppe und PGs reserviert / belegt sind.

  Hinweis

  Die Zahl der belegten Verbindungs-Ressourcen kann größer als die Zahl der reservierten sein, da - innerhalb der maximalen Anzahl - noch freie Verbindungs-Ressourcen für die PG-Kommunikation verwendet werden können. Bei S7-300-CPUs mit parametrierbaren Verbindungs-Ressourcen können Sie die Anzahl Verbindungs-Ressourcen für PG-Kommunikation zu Lasten der S7-Basis-Kommunikation erhöhen.

- OP-Kommunikation "reserviert" und "belegt":
  Anzahl der Verbindungs-Ressourcen, die für Verbindungen zwischen der Baugruppe und HMI-Geräten (TDs, OPs) reserviert / belegt sind.

  Hinweis

  Die Zahl der belegten Verbindungs-Ressourcen kann größer als die Zahl der reservierten sein, da - innerhalb der maximalen Anzahl - noch freie Verbindungs-Ressourcen für die OP-Kommunikation verwendet werden können. Bei S7-300-CPUs mit parametrierbaren Verbindungs-Ressourcen können Sie die Anzahl Verbindungs-Ressourcen für OP-Kommunikation zu Lasten der S7-Basis-Kommunikation erhöhen.

- S7-Kommunikation "reserviert" und "belegt":
  Wird angezeigt bei S7-400 CPUs sowie bei der CPU 318-2 und bei der CPU 317-2 PN/DP. Reservierte / Belegte Verbindungs-Ressourcen für projektierte S7-Verbindungen, über die durch Aufruf von Kommunikations-Funktionsbausteinen (SFBs) im Anwenderprogramm Daten ausgetauscht werden können. Diese Verbindungen können wie fest verdrahtet betrachtet werden (z.B. CPU-CPU-Verbindungen). Sie können im Anlauf der Kommunikation automatisch aufgebaut werden oder programmgesteuert auf- und wieder abgebaut werden. S7-Kommunikation ist über MPI, PROFIBUS und Industrial Ethernet möglich.

- S7-Basiskommunikation "reserviert" und "belegt":
  Wird angezeigt bei S7-300-CPUs (nicht bei CPU 318-2 DP und nicht bei CPU 317-2 PN/DP).
  Reservierte / belegte Verbindungs-Ressourcen für nichtprojektierte Verbindungen, über die durch Aufruf von Kommunikations-Funktionen (SFCs) im Anwenderprogramm Daten ausgetauscht werden können. S7-Basis-Kommunikation ist nur über MPI bzw. innerhalb des Automatisierungssystems möglich.
- Sonstige Kommunikation "belegt":
  Weitere belegte Verbindungs-Ressourcen, für die keine Verbindungs-Ressourcen reserviert sind.
  Beispiele:
  - Kommunikation einer S7-300-CPU über einen CP (nur bei CPUs mit parametrierbaren Verbindungs-Ressourcen)
  - Netzübergang (S7-Routing) bei S7-400-CPUs

Hinweis

Durch eine Server-Funktion bei der S7-Kommunikation belegte Verbindungs-Ressourcen (d. h. wenn eine S7-400-CPU über die SFCs "PUT" oder "GET" mit einer S7-300-CPU kommuniziert) werden bei S7-300-CPUs unter folgenden Ressourcen angezeigt:

- Wenn die CPU keine parametrierbaren Verbindungs-Ressourcen hat: unter "S7-Basiskommunikation"
- Wenn die CPU parametrierbare Verbindungs-Ressourcen hat: bei den CPUs 317-2 PN / DP und 318-2 unter "S7-Kommunikation", bei allen anderen unter "Sonstige Kommunikation"

Statistikdaten der über die Ethernet-Schnittstelle stattgefundenen Datenübertragungen anzeigen

Wo befinden sich die gesuchten Informationen?

Die Statistikdaten der über die Ethernet-Schnittstelle stattgefundenen Datenübertragungen finden Sie in der Online- und Diagnosesicht im Ordner "Diagnose" in der Gruppe "PROFINET-Schnittstelle" im Bereich "Statistik".

Bereich "Statistik"

- Anzeige
  Angezeigt werden Statistikdaten für gesendete und empfangene Datenpakete. Mit Hilfe dieser Daten kann die Qualität der Datenübertragung bewertet werden.
  Der Beginn der Aufzeichnungen der Statistikdaten wird in der ersten Zeile angezeigt ("Datenpakete seit:"). Jedes Rücksetzen der Statistikdaten setzt diese Zeit neu (Schaltfläche "Rücksetzen").
  Die Aktualisierung der Anzeige der Statistikdaten erfolgt fortlaufend.
- Schaltfläche "Rücksetzen"
  Die Zeit "Datenpakete seit" wird neu gesetzt und die Statistikdaten werden auf "0" zurückgesetzt.

Informationen über die Betriebsstundenzähler einer CPU anzeigen

Wo befinden sich die gesuchten Informationen?

Informationen über die Betriebsstundenzähler finden Sie in der Online- und Diagnosesicht im Ordner "Diagnose" in der Gruppe "Betriebsstundenzähler".

Gruppe "Betriebsstundenzähler"

In der Gruppe "Betriebsstundenzähler" werden für die aktiven Betriebsstundenzähler der CPU die folgenden Informationen angezeigt:

- Nr. des Betriebsstundenzählers samt Information, ob es sich um einen 16- oder 32-Bit-Betriebsstundenzähler handelt
- Aufgelaufene Betriebsstunden
- Status
- Information, ob ein Überlauf stattgefunden hat

Falls die CPU einen oder mehrere 32-Bit-Betriebsstundenzähler hat, erhält das zugehörige Icon beim Überschreiten von 32767 (Maximalwert eines 16-Bit-Betriebsstundenzählers) eine rote Umrandung. In der Spalte "Überlauf" wird "nein" angezeigt, wenn der Betriebsstundenzähler den Maximalwert eines 32-Bit-Betriebsstundenzählers noch nicht überschritten hat.

Der 32-Bit-Betriebsstundenzähler wird durch die Anweisung "RTM" gesetzt, gestartet, gestoppt und ausgelesen.

2.5.1.3 Eine Baugruppe auf Störungen überprüfen

Kanalspezifische Diagnose von Nicht-CPU-Baugruppen durchführen

Wo werden kanalspezifische Diagnoseereignisse von Nicht-CPU-Baugruppen angezeigt?

Die Anzeige von kanalspezifischen Diagnoseereignissen von Nicht-CPU-Baugruppen erfolgt in deren Online- und Diagnosesicht in der Gruppe "Kanaldiagnose".

Gruppe "Kanaldiagnose"

Die Anzeige besteht für jedes Ereignis aus den folgenden Elementen:

- Steckplatz der gestörten Baugruppe
- Nummer des gestörten Kanals
  Beachten Sie, dass die Zählweise der Kanalnummern in dieser Darstellung bei 0 beginnt.
- Fehlerart

Hinweis

Diagnosealarme

Ein Diagnosealarm kann nur dann zur CPU gemeldet werden, wenn die Baugruppe diagnosealarmfähig ist und Sie den Diagnosealarm freigegeben haben.

Die Anzeige des Diagnosealarms ist eine Momentaufnahme. Sporadische Störungen einer Baugruppe erkennen Sie im Diagnosepuffer der zugehörigen CPU.

Wenn Sie in der Tabelle mit den Kanalfehlern eine Zeile markieren, dann werden zu diesem Fehler zusätzliche Informationen bereitgestellt (sofern vorhanden).

Kommunikationsdiagnose von PROFINET-Ports durchführen

Wo werden portspezifische Fehler von PROFINET-Ports angezeigt?

Die Anzeige von portspezifischen Fehlern von PROFINET-Ports erfolgt in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" unter "PROFINET-Schnittstelle" in der Gruppe "Kommunikationsdiagnose".

Aufbau der Gruppe "Kommunikationsdiagnose"

Die Gruppe "Kommunikationsdiagnose" besteht aus folgenden Bereichen:

- Tabelle mit den portspezifischen Fehlern
- Textfeld "Details zum Ereignis"
- Textfeld "Hilfe zum Ereignis"

Aufbau der Tabelle mit den portspezifischen Fehlern

Jede Zeile der Fehlertabelle entspricht einem Fehler.

Die Tabelle besteht aus den folgenden Spalten:

- Name: Port-Nr. und Portbezeichnung sowie Diagnosesymbol
- Fehler: Beschreibung des bei diesem Port aufgetretenen Fehlers

Textfeld "Details"

Wenn Sie in der Fehlertabelle eine Zeile markieren, enthält das Textfeld "Details" Detailinformationen zum zugehörigen Fehler (sofern vorhanden).

Textfeld "Hilfe zur markierten Diagnosezeile"

Wenn Sie in der Fehlertabelle eine Zeile markieren, enthält das Textfeld "Hilfe zur markierten Diagnosezeile" Hilfeinformationen zum zugehörigen Fehler (sofern vorhanden).

PROFINET IO-Komponenten diagnostizieren

Wo diagnostizieren Sie PROFINET IO-Komponenten?

Die Diagnose von PROFINET IO-Komponenten erfolgt in deren Online- und Diagnosesicht im Ordner "Diagnose" unter "PROFINET-Schnittstelle" in der Gruppe "PROFINET IO-Diagnose".

Gruppe "PROFINET IO-Diagnose"

In diesem Bereich werden herstellerbezogene Diagnosetexte für das IO-Device bzw. für das selektierte Modul des IO-Devices angezeigt.

Wenn vom IO-Controller die Diagnose gelesen werden kann, werden hier Kommunikationsfehler und Projektierungsfehler (Unterschiede online/offline) angezeigt.

NCM S7-Diagnose starten

Voraussetzungen

- Eine Online-Verbindung zur Baugruppe braucht nicht zu bestehen.

Vorgehen

Um die NCM S7-Diagnose zu starten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Baugruppe in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Spezialdiagnose".
3. Klicken Sie im Bereich "Spezialdiagnose" auf die Schaltfläche "Start Spezialdiagnose".

Ergebnis

Die NCM S7-Diagnose wird gestartet.

Online-Verbindungen für Spezialdiagnose und Firmware-Lader

Spezialdiagnose: Online-Verbindung herstellen

Einige Netzkomponenten (S7-300/400-CPs, PC-CPs, Netzübergänge) stellen erweiterte Diagnosedaten in der Spezialdiagnose zur Verfügung.

Voraussetzung:

Eine PG/PC-Schnittstelle ist physikalisch mit dem Ziel-Gerät verbunden, z. B. über ein Ethernet-Kabel.

Vorgehen:

Gehen Sie zum Start der Spezialdiagnose folgendermaßen vor:

1. Öffnen Sie die Online- und Diagnosesicht der Baugruppe.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Spezialdiagnose".
3. Klicken Sie im Bereich "Spezialdiagnose" auf die Schaltfläche "Starte Spezialdiagnose".

Ab STEP 7 Professional V14 kann von der Engineering-Station für einige CPs nur eine einzige Online-Verbindung über den CP aufgebaut werden. Diese Verbindung kann entweder mit STEP 7 oder der Spezialdiagnose aufgebaut werden.

Wenn die Spezialdiagnose über die oben genannten Schritte nicht gestartet und eine Fehlermeldung ausgegeben wird, dann gehen Sie folgendermaßen vor:

1. Trennen Sie die bestehende Online-Verbindung zur Station.
2. Öffnen Sie die Online- und Diagnosesicht der Baugruppe erneut wie oben beschrieben und starten Sie die Spezialdiagnose.

32-Bit-Applikationen: NCM S7-Diagnose, Firmware-Lader

Wenn Sie von einer 32-Bit-Applikation aus eine Verbindung mit einem PC-CP aufgebaut haben und eine weitere Online-Verbindung über dieselbe Schnittstelle des PC-CP aufbauen möchten, dann müssen Sie vorher die Verbindung der 32-Bit-Applikation beenden.

Spezialdiagnose und Firmware-Lader: Chinesische GUI

Wenn Sie die Spezialdiagnose oder den Firmware-Lader unter chinesischer GUI öffnen, dann müssen Sie die Sprache für nicht-Unicode-Applikationen im Betriebssystem auf Chinesisch stellen.

2.5.1.4 Eigenschaften einer Baugruppe bzw. des PG/PC ändern

Besonderheit beim Umschalten des Betriebszustands einer S7-400-CPU

Vorgeben der Anlaufart einer S7-400-CPU beim Umschalten in den Betriebszustand "RUN"

Wenn Sie eine S7-400-CPU in den Betriebszustand "RUN" umschalten wollen, wählen Sie eine der folgenden Anlaufarten:

- Neustart (Warmstart)
- Kaltstart
- Wiederanlauf

Sich bei der CPU für Meldungen und Systemdiagnoseereignisse anmelden

Voraussetzung

- Über die PG / PC-Schnittstelle können ein oder mehrere Teilnehmer eines Subnetzes erreicht werden.

Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht der zugehörigen CPU.
2. Wählen Sie die Gruppe "Online-Zugänge".
3. Aktivieren Sie im Bereich "Meldungen" das Optionskästchen "Meldungen empfangen".

Ergebnis

Ihr PG / PC wird beim nächsten Online Gehen bei der betroffenen CPU für Meldungen und Systemdiagnose-Ereignisse angemeldet.

Hinweis

Die Meldungsanzeige benötigt für jede CPU, von der Meldungen empfangen werden, eine ständige Online-Verbindung.

Anmeldung bei bereits bestehender Online-Verbindung vornehmen

Wenn die Online-Verbindung zu einer CPU bereits aufgebaut ist, können Sie sich nachträglich folgendermaßen für Meldungen an- bzw. abmelden:

- Im Menü "Online" über den Befehl "Meldungen empfangen"
- In der Projektnavigation über das Kontextmenü der CPU
- In der Gerätesicht der Hardware-Konfiguration über das Kontextmenü der CPU
- In der Netzsicht der Hardware-Konfiguration über das Kontextmenü der CPU

Siehe auch

Meldungen empfangen (Seite 3206)

Speicher komprimieren und von RAM nach ROM kopieren

Wo befinden sich die gesuchten Informationen?

Die Reorganisation des Speichers und das Kopieren von RAM nach ROM finden Sie in der Task Card "Online-Tools" im Kommandobereich der Palette "Speicher".

Die Reorganisation des Speichers finden Sie außerdem in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" im Kommandobereich der Gruppe "Speicher".

Kommandobereich der Palette "Speicher" in der Task Card "Online Tools"

Dieser Bereich kann die folgenden Schaltflächen enthalten:

- Komprimieren
  Mit dieser Schaltfläche reorganisieren Sie den Speicher, um die Größe des größten zusammenhängenden freien Speicherbereichs zu erhöhen.
- Kopieren RAM nach ROM
  Mit dieser Schaltfläche kopieren Sie den Inhalt des RAM-Ladespeichers (Anwenderprogramm) auf ein FEPROM:
  - auf die in der CPU steckende Memory Card
  - auf das integrierte FEPROM

Hinweis

Eine Schaltfläche wird genau dann angezeigt, wenn die Baugruppe die zugehörige Funktion beherrscht.

Kommandobereich der Gruppe "Speicher" im Ordner "Diagnose" in der Online- und Diagnosesicht

Dieser Bereich enthält die folgenden Schaltflächen:

- Komprimieren
  Mit dieser Schaltfläche reorganisieren Sie den Speicher, um die Größe des größten zusammenhängenden freien Speicherbereichs zu erhöhen.

Die PROFINET-Schnittstellenparameter einer S7-300- oder S7-400-CPU zurücksetzen

Voraussetzung

- Sie benutzen eine S7-300-CPU beliebigen Firmware-Stands oder eine S7-400-CPU mit Firmware-Stand >= V6.0.
- Es steckt keine Memory Card in der CPU.
- Es besteht eine Online-Verbindung zur CPU, deren PROFINET-Schnittstellenparameter zurückgesetzt werden sollen.
- Diese CPU befindet sich im Betriebszustand STOP.

Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Rücksetzvorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

Vorgehen

Um die PROFINET-Schnittstellenparameter einer S7-300- oder S7-400-CPU zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen der PROFINET-Schnittstellenparameter".

   Hinweis

   Die Benutzeroberfläche der Gruppe "Rücksetzen der PROFINET-Schnittstellenparameter" der Online- und Diagnosesicht hängt davon ab, wie Sie die Online- und Diagnosesicht aufrufen:
   - Beim Aufruf aus der Projektnavigation über "Erreichbare Teilnehmer aktualisieren" ist die MAC-Adresse der zugehörigen Baugruppe bekannt und daher gegraut.
   - Beim Aufruf aus dem Projektkontext müssen Sie die zugehörige MAC-Adresse eingeben. Andernfalls bleibt die Schaltfläche "Rücksetzen" gegraut.

3. Klicken Sie auf die Schaltfläche "Rücksetzen".
4. Beantworten Sie die Sicherheitsabfrage mit "OK".

Ergebnis

Die Baugruppe wird ggf. in den Betriebszustand STOP gesetzt und die PROFINET-Schnittstellenparameter werden zurückgesetzt. Das bedeutet:

- Die Einstellungen der Ethernet-Schnittstelle werden zurückgesetzt.
- Die IP-Adresse wird gelöscht.

Firmware-Lader für einen S7-300- bzw. S7-400-CP starten

Voraussetzungen

- Der zu aktualisierende CP unterstützt ein Firmware-Update.
- Eine Online-Verbindung zum CP braucht nicht zu bestehen.

Vorgehen

Um den Firmware-Lader für einen S7-300- oder S7-400-CP zu starten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den CP in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".
3. Klicken Sie im Bereich "Firmware-Update" auf die Schaltfläche "Starte Firmware-Lader".

Ergebnis

Der Firmware-Lader wird gestartet.

Hinweis

Das Aktualisieren der Firmware bei S7-300- und S7-400-CPs über den Firmware-Lader weicht von den Standardmechanismen zum Firmware-Update ab.

Firmware einer TIM 3V / TIM 4R aktualisieren

Gültigkeit

Die nachfolgenden Ausführungen gelten für folgende TIM-Baugruppen für die SIMATIC S7-300 / 400:

- ST7-TIM
  - TIM 3V-IE
  - TIM 3V-IE Advanced
  - TIM 4R-IE
  - TIM 4R-IE Standalone
- DNP3-TIM
  - TIM 3V-IE DNP3
  - TIM 4R-IE DNP3

Reihenfolge der Aktualisierung

Für die Firmware-Aktualisierung diese TIM-Baugruppen stehen verschiedene Dateien zur Verfügung:

- Firmware-Dateien
- Treiberpakete

Hinweis

Reihenfolge der Aktualisierung

Beachten Sie bei einem Firmware-Update die Reihenfolge der Aktualisierung:

1. **Aktualisieren Sie zuerst die Treiberpakete (*.Z).**
2. **Aktualisieren Sie erst anschließend die Firmware-Dateien (*.FWL).**

Wenn Sie zuerst die Firmware-Dateien aktualisieren und versehentlich die falsche Datei auswählen, ist die TIM-Baugruppe unter Umständen nicht mehr erreichbar.

Dateien für die Firmware-Aktualisierung

Die Dateien für die Firmware-Aktualisierung werden als gepackte Datei-Archive zur Verfügung gestellt. Die Archive enthalten die Dateien für die Treiberpakete und die Firmware sowie Liesmich-Dateien, welche Angaben zur Firmware-Version enthalten.

Für folgende TIM-Baugruppen wird jeweils ein Archiv zur Verfügung gestellt:

- Archiv für ST7-TIM 3V..
  - TIM 3V-IE
  - TIM 3V-IE Advanced
- Archiv für ST7-TIM 4R..
  - TIM 4R-IE
  - TIM 4R-IE Standalone
- Archiv für TIM 3V-IE DNP3
- Archiv für TIM 4R-IE DNP3

Gehen Sie zum Speichern eines Datei-Archivs folgendermaßen vor:

1. Kopieren Sie das Archiv in das Dateisystem Ihrer Engineering-Station.
   Das Verzeichnis geben Sie beim Aktualisierungsvorgang im Online-Dialog an.
   Empfehlung:
   Kopieren Sie Archive für Firmware-Versionen verschiedener Baugruppentypen in separate Verzeichnisse.
2. Entpacken Sie das Archiv, nachdem Sie es in das Dateisystem Ihrer Engineering-Station kopiert haben.

Siehe auch

Online-Verbindungen für Spezialdiagnose und Firmware-Lader (Seite 3567)

Einem PROFINET IO-Device eine IP-Adresse zuweisen

Grundsätzliches zur Vergabe einer IP-Adresse an ein PROFINET IO-Device

Übersicht

Für die Wahl einer IP-Adresse haben Sie grundsätzlich zwei Möglichkeiten:

- Sie geben die IP-Adresse selbst vor.
- Sie beziehen die IP-Adresse von einem DHCP-Server. (Dies ist nicht bei allen IO-Devices möglich.). In diesem Fall wird dem DHCP-Server je nach gewählter Option entweder die Client-ID oder die MAC-Adresse oder der Gerätename des IO-Device übermittelt.

Voraussetzung

Für das Folgende wird vorausgesetzt, dass die IP-Adresse des IO-Device auch von einem DHCP-Server bezogen werden kann. Dies ist z. B. bei einigen S7-300-CPs der Fall.

Die Adresszuweisung über "Erreichbare Teilnehmer" aufrufen

Voraussetzung

- Die über die zugehörige Schnittstelle des PG / PC erreichbaren Teilnehmer werden in der Projektnavigation angezeigt (Dies erreichen Sie, indem Sie entweder in der Projektnavigation auf "Erreichbare Teilnehmer aktualisieren" doppelklicken oder indem Sie im Menü "Online" den Befehl "Erreichbare Teilnehmer..." wählen.).
- Sie haben in der Projektnavigation unter "Online-Zugänge" -> <gewählte Schnittstelle> -> <PROFINET IO-Device> -> "Online & Diagnose" doppelgeklickt, so dass die Online- und Diagnosesicht geöffnet wurde.

Vorgehen bei eigener Vorgabe der IP-Adresse

Um dem IO-Device eine von Ihnen vorgegebene IP-Adresse zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Online- und Diagnosesicht des IO-Device.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "IP-Adresse zuweisen".
3. Wählen Sie die Option "IP-Parameter verwenden".
4. Tragen Sie die gewünschte IP-Adresse ein.
5. Tragen Sie die Subnetzmaske ein.
6. Falls ein Router verwendet werden soll, aktivieren Sie das Optionskästchen "Router verwenden" und tragen anschließend dessen IP-Adresse ein.
7. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

Vorgehen bei Bezug der IP-Adresse von einem DHCP-Server

Um dem IO-Device eine von einem DHCP-Server vorgegebene IP-Adresse zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Online- und Diagnosesicht des IO-Device.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "IP-Adresse zuweisen".
3. Wählen Sie die Option "IP-Adresse von einem DHCP-Server beziehen".
4. Wenn der DHCP-Server das Ziel-Device über dessen Client-ID identifizieren soll, wählen Sie die Option "Client-ID". Tragen Sie die Client-ID in das gleichnamige Eingabefeld ein.

Hinweis

Die Client-ID ist eine Zeichenfolge mit max. 63 Zeichen. Es dürfen nur die folgenden Zeichen verwendet werden: a - z, A - Z, 0 - 9, - (Bindestrich)

5. Alternativ: Wenn der DHCP-Server das Ziel-Device über dessen MAC-Adresse identifizieren soll, wählen Sie die Option "MAC-Adresse
6. Alternativ: Wenn der DHCP-Server das Ziel-Device über dessen Gerätenamen identifizieren soll, wählen Sie die Option "Gerätename".

   Hinweis

   Wenn Sie hier festlegen, dass der DHCP-Server das Ziel-Device über dessen Gerätenamen identifizieren soll, müssen Sie dem Ziel-Device zuvor einen Gerätenamen zugewiesen haben.

7. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

Ergebnis

Dem IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder einen Spannungsausfall hinaus erhalten.

Die Adresszuweisung aus dem Projektkontext aufrufen

Voraussetzung

- Es besteht eine Online-Verbindung zum PROFINET IO-Device.
- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device aus dem Projektkontext aufgerufen.

Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "IP-Adresse zuweisen". Die Option "IP-Parameter verwenden" ist gewählt.
2. Klicken Sie auf die Schaltfläche "Erreichbare Teilnehmer", um die über MAC-Adressen erreichbaren Geräte zu ermitteln. Wählen Sie nach der durchgeführten Suche das IO-Device mit der Ihnen bekannten MAC-Adresse aus.
3. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

Ergebnis

Dem IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder einen Spannungsausfall hinaus erhalten.

2.5.2 Verbindungen diagnostizieren

2.5.2.1 Verbindungen im Detail diagnostizieren

Verbindungsdetails einer NCM-Verbindung ermitteln

Wo ermitteln Sie die Verbindungsdetails einer NCM-Verbindung?

Die Verbindungsdetails einer NCM-Verbindung entnehmen Sie der Gruppe "Verbindungsdetails". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

Wann ist die Gruppe "Verbindungsdetails" befüllt?

Damit die Gruppe "Verbindungsdetails" des Registers "Verbindungsinformation" befüllt ist, müssen folgende Voraussetzungen erfüllt sein:

- Es besteht eine Online-Verbindung zur betrachteten Baugruppe.
- Sie haben in der Verbindungstabelle eine Zeile markiert.

Aufbau der Gruppe "Verbindungsdetails" bei einer NCM-Verbindung

Die Gruppe "Verbindungsdetails" besteht bei einer NCM-Verbindung aus folgenden Elementen:

- Lokale Verbindungs-ID (hex)
- Verbindungsname
- Verbindungszustand
- Sendestatus
- Empfangsstatus
- Anzahl der gesendeten Bytes
- Anzahl der empfangenen Bytes
- Schaltfläche "Spezialdiagnose"
  Durch Klicken auf diese Schaltfläche starten Sie die NCM-Verbindungsdiagnose.

Siehe auch

NCM S7-Diagnose starten (Seite 3567)

Index

*

*.cer, 500
*.dat, 500
*.p12, 500

:

:P, 971, 1379

3

32-Bit Rückwärtszähler, 3369
3964(R)
    Daten empfangen, 1651
    Daten senden, 1650
3964(R)-Protokoll (CPU 31xC), 852

A

A-Adresse, 1161, 1458
Ablage nicht gesteckter Baugruppen, 20
Abmeldung
    automatisch, 664, 2577, 2777
Abtastrate (High-Speed-Analogmodule), 3415
Access Point
    Überlappende Kanäle, 2514
    Übersicht, 2509
ACL, 2325
ACT_TINT, 798
Adressband, 426
Adressbelegung (Wertstatus), 974
Adressbereich
    verändern, 1162, 1459
Adressbildung bei gepackten Adressen, 2893, 2973
Adresse, (Siehe Baugruppen-Anfangsadresse)
Adresse des Netzübergangs, 590, 2696
Adressen, 3269
    Alarm bei gepackten, 2900, 2979
    Alarme im I-Slave, 2905
    DP-Schnittstellen, 2933
    DP-Slave, 2935
    entpacken, 2893, 2973
    Nutzdaten mit DPV1, 2933
    packen, 2893, 2972
    Systeminformationen, 2935
    zuordnen, 1175, 1461
Adressierung, 971, 1379
    allgemein, 1160, 1161, 1458
    verändern, 1162, 1459
Adressparameter
    Broadcast, 276
Adressübersicht, 66, 1170
Adressvergabe, 1160
Adresszuordnung
    Nutzdaten mit DPV1, 2934
Advanced Encryption Algorithm, 326
AES, 326
AES-128, 459
Aging Time, 2329, 2333
Akkumulatoren, 863
Aktive Teilnehmer, 479
Aktives Busmodul (ET 200M), 3454
Aktivieren und Deaktivieren von IO-Devices, 3090
Aktualisierungszeit, 3017
Alarm
    senden im I-Slave, 2905
Alarme
    bei gepackten Adressen, 2900, 2979
    Diagnosealarm (S7-300, S7-400), 894
    OBs für DPV1, 2928
    Prozessalarm (S7-300, S7-400), 878
    Uhrzeitalarm (S7-300, S7-400), 871
    Verzögerung (S7-300, S7-400), 874
    Weckalarm (S7-300, S7-400), 876
    Ziehen/Stecken-Alarm (S7-300, S7-400), 897
Alarmereignisse, 650, 2765
Alarminfos auswerten mit DPV1, 2931
Alternative Partner (Im Betrieb wechselnde IO-Devices, 3092
Analogmodul
    auf Werkseinstellungen zurücksetzen, 3526
Anfangsadresse, 971, 1160, 1161, 1378, 1458
Anfangssequenzen, 1642
Anlagenkennzeichen, 789
Anlagenrelevante Informationen
    Angaben zur Baugruppe, 3557
    Baugruppeninformation, 3487
    Herstellerangaben, 3557
    Herstellerinformation, 3487
ANLAUF, 756
    Abbruch, 761
    Anlauftätigkeiten, 760, 955, 1367
    Automatischer Neustart (Warmstart), 757

Funktion, 953, 1366
Gesamtübersicht,
Manueller Neustart (Warmstart), 757
Neustart (Warmstart), 757, 954, 1367
Organisationsbausteine,
Remanenzverhalten, 759
Ungepufferter automatischer Neustart
(Warmstart), 757
Anlauf beschleunigen (PROFINET), 3094
Anlaufart, 988, 1390
Anlaufart der CPU, 1550
Anlauf-OB
Beschreibung, 1432
Anlauf-OBs (S7-1500), 1056
Anlauf-OBs OB 100, OB 101, OB 102 (S7-300, S7-400), 919
Anlaufparameter, 790, 956, 1369
Anlaufprogramm, 1432
Anlaufprogramm (S7-1500), 1056
Anlaufverhalten, 1796
Anlegen
Direkter Datenaustausch, 2916
Anmelden, 1323
Anmelden für Webserver, 1465
Anschaltungsbaugruppe, 1156
Ansicht vergrößern/verkleinern, (Siehe Zoom)
Ansicht verschieben
Tastaturbedienung, 39
Übersichtsnavigation, 13, 16, 23
Ansprechüberwachungszeit, 3019
Antragsteller, 328
Antwort-Zeitüberschreitung, 1643
Anweisung "WWW", 1352, 1487
Anwenderdefinierte Webseiten, 1330, 1334, 1345, 1346, 1348, 1349, 1351, 1352, 1362, 1469, 1473, 1482, 1483, 1485, 1486, 1487
Anwenderkonstante, 1174
Anwenderprogramm, 1111
Anwenderseiten, (siehe Anwenderdefinierte Seiten), (siehe Anwenderdefinierte Webseiten)
Anwenderseite als Startseite, 1335
Anwendungsgebiet, 3187
Anzeigeklasse, 806
Anzeigeklasse für Meldungen, 1330
Applikationszyklus, 3064
Äquidistanz, 2939
Aktivierung im PROFIBUS-Netz, 2948
Definition, 2937
Voraussetzungen, 2938
Zeit für äquidistanten DP-Zyklus, 2938
Zusammensetzung der Buszyklen, 2937
Arbeitsspeicher, 771, 775, 964, 1374
ARP, 552
Diagnose, 552
Tabelle, 552
Array
Webserver, 1346, 1349
Artikelnummer, 30, 618, 2131, 2488, 2724
ASCII-Codetabelle, 134
ASCII-Protokoll, 1639
ASCII-Protokoll (CPU 31xC), 845
ASCII-Treiber, 1682
Daten empfangen, 1684, 1719
Daten senden, 1682
Empfangspuffer, 1689
RS232C-Begleitsignale, 1690, 1715
ASCII-TSAP, 134
AS-Interface, 2985
Assetmanagement, 3173
Asymmetrische Verschlüsselung, 327
Asynchrone Datenübertragung, 1620
Asynchronfehler
Einstellungen, 797
Asynchronfehlerereignis
Ereignis-ID, 931
ATTACH, 1386
Attacke, 1035, 1423
Audit-Log, 496
Aufbau, 3382, 3425, 3441
Aufrufstruktur, 964, 965
Auftragsheader, 288, 289, 290
Auftragspuffer, 288, 289, 290
Ausgaberate (High-Speed-Analogmodule), 3415
Ausgang (A), 1374
Ausgangsbyte (AB), 1374
Ausgangsdoppelwort (AD), 1374
Ausgangswort (AW), 1374
Auswerten
Alarminfos mit DPV1, 2931
Diagnoseinfos mit DPV1, 2931
Authentifizierung, 655, 2223, 2420, 2567, 2769
Authentifizierungsmethoden, 464, 467
Automatikbetrieb des PROFINET IO-Controllers, 1567
Automatisch aktualisieren, 1172
Automatische Aktualisierung, 1172
Automatische Bedienung der Begleitsignale, 1636
Automatische Inbetriebnahme, 3026
Autonegation aktivieren, 3023
Autonegotiation, 3025
AWP_In_Variable, 1477
AWP_Out_Variable, 1338, 1476
AWP-Kommando, 1334, 1337, 1473, 1474

B

Backup, 2264
Baisis-Submodul, 974
Bandbreite (IRT-Kommunikation), 3053
BA-Send, 3397
BA-Send 1xFC, 3397
Baugruppe
  adressieren, 1160, 1161, 1458
  Diagnosestatus ermitteln, 3498
  einfügen, 52
  Kommunikationseigenschaften, 3558
  kopieren, 57
  löschen, 63
  Mehrfachauswahl, 54
  selektieren, 49
  stecken, 51
  tauschen, 60
  Uhrzeit einer Baugruppe, 3508
  Umbennen, 61
  verschieben, 59
Baugruppenadressierung, 786, 1378
Baugruppen-Anfangsadresse, 971
Baugruppenanordnung, 50
Baugruppenbeschriftung, 48
Baugruppenträger, 48, 1155
  Baugruppe stecken, 51
  Bestückungsregeln S7-300, 1140
  Bestückungsregeln S7-400, 1141
  Maximale Anzahl, 1140, 1141
Baugruppenträgerausfall-OB (OB 86) (S7-300, S7-400), 908
Baugruppenträgerfehler-OB, 1446
Baugruppenträgerfehler-OB (S7-1500), 1076
Baugruppenwechsel im Betrieb, 3454
Bausteinparameter
  Grundlagen, 1451
Baustein-Stack, 774, 781
BCC, 1649
Bearbeitungsabbruch-OB (OB88) (S7-300, S7-400), 916
Bearbeitungszyklus, 2939
Befehlstelegramm, 1703
Begleitsignale, 1617
Beispiel
  Konfiguration ET 200AL, 3426
  Vorgabe eines Ersatzwerts, 928
Beispielprogramm
  OB 122, 928
  OB 81 (S7-300, S7-400), 893
Belegung von Verbindungsressourcen, 187
Belegungsplan, 964, 965
Benutzergruppen, 728, 2415, 2650, 2846
Benutzerrechte, 3231
Benutzerverwaltung, 806, 3231
  Webserver, 1326
Beobachtungstabelle
  Webserver, 1323
Bestellnummer, (siehe Artikelnummer)
Bestellnummern, 1615
Betrieb bei Sollausbau ungleich Istausbau, 2890
Betriebsartenschalter, 3562
Betriebsstundenzähler, 804
  Informationen anzeigen, 3565
Betriebsverhalten
  Baugruppenparameter, 788
  Parameter einstellen und laden, 788
  Standardeinstellungen, 788
  Systemmerker verwenden, 1025, 1416
  Taktmerker verwenden, 804, 1026, 1417
  Uhrzeitfunktionen nutzen, 802, 1023, 1399
  Zeiten und Zähler verwenden, 784
Betriebszustand, 557
  Ablaufereignisse, 938
  ANLAUF, 953, 1366
  Einführung, 951, 1364
  HALT, 765
  Priorität, 755
  RUN, 765, 957, 1369
  STOP, 755, 957, 958, 1370
  Übergänge, 754, 934, 952, 1365
  Übersicht, 753
Betriebszustand ANLAUF, 1107
Betriebszustand RUN, 1109
Betriebszustand STOP, 1108
Betriebszustand umschalten, 3506
Betriebszustandsübergänge, 1668, 1796
Bidirektionaler Datenverkehr, 1620
Bidirektionaler Multicast, 2116
Blinktest, 3532
Blockprüfzeichen, 1649, 1695
Bootstrap Router, 2117
Boundaries, 3026
Bridge, 629, 704, 2742, 2823
  Bridge Priority, 629, 704, 2742, 2823
  Root Bridge, 629, 704, 2742, 2823
Bridge Max Age, 629, 2742
Bridge Max Hop Count, 630, 2743
Bridge Priority, 2479
Broadcast, 243, 289, 427, 435, 1654, 1731, 2334
  bei UDP, 275
Brute Force, 1035, 1423
Brute-Force-Angriff, 1035, 1423

Brute-Force-Attacke, 1035, 1423
Bruttodatenmenge, 3225
BSR-Border, 2117
BUSY-Signal, 1718
Buszyklen, 2937

C

CA-Gruppenzertifikat, 381
CA-Gruppenzertifikat erneuern, 479
CA-Zertifikat, 375, 381, 605, 2713
CER, 347
Certificate Authority, 375
Certificates, 732, 2850
Class of Service, 2269
Client
    Übersicht, 2515
Client Supplicant, 2662
Clipping, 3418
Codebaustein, 964
Codetransparenz, 1646
Configuration Mode, 640, 2753
Cookie, 1342, 1478
CoS, 2269
    Warteschlange, 2269
CoS (Class of Service), 594, 2082, 2461, 2698
CoS-Priorisierung, 594, 2082, 2461, 2698
CP
    als I-Slave, 2908
    mit eigener MPI-Adresse, 1160
    Punkt-zu-Punkt-Kopplung, 1670
C-PLUG, 675, 2584, 2782
    Formatieren, 676, 2253, 2586, 2784
    Konfiguration speichern, 676, 2586, 2784
CPU
    aktuellen LED-Zustand anzeigen, 3495
    als I-Slave, 2908
    Anzahl der Kommunikationsverbindungen, 178
    Auswahl aus Hardware-Katalog, 45
    belegte und reservierte Verbindungen, 3562
    Betriebszustand umschalten, 3506
    Diagnosepuffer auslesen, 3501
    Eigenschaften, 1389
    Füllstand aller Speicherarten, 3496, 3497
    nicht belegte Verbindungsressourcen, 3562
    Parameter, 789
    sich für Meldungen anmelden, 3569
    Signalboard stecken, 1463
CPU 1516-3 PN/DP, 1010
CPU 1518-4 PN/DP, 1010
CPU konfigurieren
    Projekt laden, 1523
CPU urlöschen, 766
CPU-Bedienpanel
    Anzeigebereich, 3496, 3562
CPU-Display, (Siehe Display)
CPU-Eigenschaften, 988, 1390
CPU-Firmware, 1456
CPU-Hardwarefehler-OB (OB 84) (S7-300, S7-400), 902
CPU-Meldungen - OPC-Server, 1608
CPU-RUN, 1668, 1796
CPU-STOP, 1668, 1796
CRC, 1655, 1731
CRL, 347
CRT, 347
CTS, 1625
Cu10-Sensor, 3415
Cyclic interrupt, 1440
Cyclic interrupt (S7-1500), 1066

D

D_ACT_DP, 1166
D_ACT_DP (Aktivieren und Deaktivieren von IO-Devices), 3090
Data Log, 1468
Dateiformat
    CER, 347
    CRL, 347
    CRT, 347
    DER, 347
    P12, 347
    PEM, 347
Daten empfangen, 1663
    ASCII-Treiber, 1684, 1719
    Prozedur 3964(R), 1699
Daten holen
    RK512, 1709
Daten senden, 1663
    ASCII-Treiber, 1682, 1717
    Prozedur 3964(R), 1696
    RK512, 1706
Datenaustausch
    DP-Slave, 2874, 2957
    I-Slave - DP-Master, 2876
    I-Slave - DP-Master, 2958
Datenbaustein, 964, 970
Datenbaustein 196, 3384, 3442, 3464
Datenbaustein 197, 3384, 3442, 3464
Datenbit (DBX), 1374
Datenbyte (DBB), 1374
Datendoppelwort (DBD), 1374

Datenflusskontrolle, 1409, 1617, 1624, 1632
Hardware, 1633
Software, 1633
Datenkonsistenz, 197
Datentyp
Pip, 1174
Datenübertragung beim Modbus-Master ET 200S, 1741
Datenübertragungsraten, 1616
Datenwort (DBW), 1374
Datenzyklus, 1054
DCD, 1625
DCP Server, 2184, 2531
DCP-Server, 638, 2751
Dead Peer Detection, 609, 2717
Dead-Peer-Detection (DPD), 483
Default-Domain, 3053
Default-Routen
IPv6-Routen, 2537
DER, 347
DES, 459
Designated Forwarder, 2116
Designated Router, 2116
Details zum Verbindungsaufbau, 254, 260, 266
Device Tool, 3188, 3190
Dezentrale Peripherie, 2871, 2956, 3376, 3428
konfigurieren, 2871
Steckplatznummern, 2872
Dezentrales Peripheriegerät ET 200iSP
Definition, 3362
Dezentrales Peripheriegerät ET 200M, 3453
DHCP
Client, 683, 2208, 2558, 2795
Server-Konfiguration, 538
DHCP-Server, 539
DHCPv6
Client, 2218
PD-Sub-Client, 2219
Präfix-Delegation, 2218, 2219
DHCPv6 Relay Agent
A,llgemeine Einstellungen, 2392
Diagnose, 487, 1669, 1797, 3378
I-Slave, 2912
kanalspezifisch, 3565
Online & Diagnose im TIA Portal, 1522
PC-System V2.0, 1522
PROFINET, 3047
PROFINET IO-Komponenten, 3500, 3567
Diagnose I-Device, 3084
Diagnoseadresse, 1160, 1162
I-Device, 3074
I-Slave, 2902
konfigurieren, 2904, 3076
zuordnen, 2928
Diagnosealarm-OB, 1444
Diagnosealarm-OB (OB 82) (S7-300, S7-400), 894
Diagnosealarm-OB (S7-1500), 1073
Diagnosedaten, 785
Diagnoseereignis
für Baugruppen, 940
Diagnosefunktionen, 1669, 1797
Diagnoseinfos
Auswerten mit DPV1, 2931
Diagnosepuffer, 1035, 1317, 1423
Grundlagen, 782, 1378, 3527
Organisation, 782, 3527
Diagnosepuffer auslesen, 3501
Diagnose-Repeater, 3533
Diagnosestatus
online ermitteln und anzeigen, 3477
Diagnosestatus ermitteln, 3498
Diagnostic error interrupt, 1444
Diagnostic error interrupt (S7-1500), 1073
Dialog ", 1582
Dialog "Konfiguration Zielrechner", 1582
Dienstgruppen, 412, 415
Digitale Zertifikate, 328
DIRECT_OPERATE, 1906
Direkte Kommunikation, 1905
Direkter Datenaustausch
anlegen, 2916
DP-Slave - DP-Master, 2875
DP-Slave - I-Slave, 2875, 2877
Funktionsweise, 2914
konfigurieren, 2917
PROFIBUS, 2915
Direkter Zugriff auf die Peripherie, 864, 971, 1379
Display, 998
Display (CPU), 1038
Distributed I/O, 3397
DLE, 1649
DM 370 Dummy, 1140
DNS-Client, 2190
DNS-Domain, 2190
Docking Station, 3092
Docking System, 3092
Docking Unit, 3090, 3092
Download, (Siehe Laden in Gerät)
DP/DP-Koppler, 2873, 2957
DP-Master, 3363
Anzeige im DP-Slave, 2880, 2961
DP-Mastersystem hinzufügen, 2882, 2963
DP-Schnittstelle, 2878, 2960
DPV1-Funktionalität, 2927

Geräte und Baugruppen, 2878, 2959
Trennung vom DP-Mastersystem, 2882, 2963
DP-Mastersystem
anlegen, 2872, 2878, 2918, 2956, 2959
DP-Slave einfügen, 2884, 2964
DP-Slave trennen, 2884, 2965
Eigenschaften bearbeiten, 2883, 2963
erstellen, 2886, 2966
hervorheben, 2881, 2962
mehrere, 2875
Trennung der Teilnehmer, 2881, 2962
Trennung vom Subnetz, 2881, 2962
DP-Normslave, 2926, 2983
DPRD_DAT, 1166
DP-Schnittstelle, 2878, 2960, 3559
Adressen, 2933
umschalten, 2882
DP-Slave
Adressen, 2935
als I-Slave, 2909
am DP-Mastersystem einfügen, 2884, 2964
Anzeige des DP-Masters, 2880, 2961
Datenaustausch, 2874, 2957
DP-Mastersystem zuweisen, 2885, 2965
Hardware-Katalog, 2873, 2956
intelligenter, (Siehe I-Slave)
konfigurieren, 2872, 2874, 2885, 2957, 2966
mit Vorverarbeitung, (Siehe I-Slave), (Siehe I-Slave)
Steckplatzregeln, 2872, 2884, 2964
SYNC/FREEZE-Gruppen, 2888
Taktsynchronität einstellen, 2951
Typen, 2883, 2964
vernetzen, 2886, 2966
vom DP-Mastersystem trennen, 2884, 2965
DPSYC_FR, 1166
DPV0/DPV1, 2926
DPV1, 800
Auswerten von Alarminfos, 2931
Auswerten von Diagnoseinfos, 2931
ET 200S konfigurieren, 2899, 2978
OBs für Alarme, 2928
Steckplatzmodell, 2933
DPWR_DAT, 1166
DP-Zyklus
Zeit für äquidistanten DP-Zyklus, 2938
Drahtbruch, 3379, 3446
Drucken
Gerätesicht drucken, 35
Netzsicht drucken, 35
Druckertreiber
Beispiele, 1716
BUSY-Signal, 1718
Formatstring, 1716
Handshakeverfahren, 1717
Softwarehandshake, 1717
Variablen, 1715
Drucker-Treiber
Meldetext ausgeben, 1717
Meldetexte, 1715
DSCP, 2270
DSL
Übersicht, 2737
DSR, 1625
DST
Sommerzeit, 2226, 2227, 2570, 2571
DTR, 1625
Dynamische Referenztemperatur, 3449

E

E/A-Kommunikation, 2998
E-Adresse, 1161, 1458
EC31-HMI/RTX
Visualisierung, 1562
Echtzeituhr, 3561
Eigenschaften (CPU, 988, 1390
Eigenschaften der VPN-Gruppe, 466
Eigenschaftsfenster, (siehe Inspektorfenster)
Eingang (E), 1374
Eingangsbyte (EB), 1374
Eingangswort (EW), 1374
Einsatzmöglichkeiten, 1720
Einsatzmöglichkeiten der Kommunikationsbaugruppe, 1721
E-Mail
versenden, 296
E-Mail DB, 297
E-Mail Server, 296
E-Mail Verbindung
projektieren, 294
E-Mail Verbindung - DNS-Konfiguration, 3226
E-Mail-Funktion, 650, 2765
Alarmereignisse, 650, 2765
Netzüberwachung, 650, 2765
Embedded Controller, 1563
laden:Download, 1563
Empfängerbaugruppe, 1156
Empfangspuffer, 1646, 1689, 1728
Empfangspuffergrößen, 1616
Ende der Erfassung erreichbarer Teilnehmer, 3026
Ende der Sync-Domain, 3026
Ende der Topologieerkennung, 3026
Endeerkennung eines Empfangstelegramms, 1719

Endekriterium, 1685
Ablauf der Zeichenverzugszeit, 1685
Endezeichen, 1686
feste Telegrammlänge, 1688
End-Entity-Zertifikat, 331
Endesequenz, 1643
Endezeichen, 1719
Enumerationen
Webserver, 1345, 1482
Enumerationstypen, 1344, 1481
Ereignis, 929
Ereignis-ID, 929
Ereignisklasse, 929
Ereignisklasse 1 - Standard-OB-Ereignisse, 930
Ereignisklasse 2 - Synchrone Fehlerereignisse, 930
Ereignisklasse 3 - Asynchrone Fehlerereignisse, 931
Ereignisklasse 4 - Stopereignisse und andere Betriebszustandsübergänge, 934
Ereignisklasse 5 - Betriebszustands-Ablaufereignisse, 938
Ereignisklasse 6 - Kommunikationsereignisse, 939
Ereignisklasse 8 - Diagnoseereignisse für Baugruppen, 940
Ereignisklasse 9 - Standard-Anwenderereignisse, 942
Ereignisklasse A und B - freie Anwenderereignisse, 943
Ereignisname, 1454
Ereignisse
Konfiguration, 2203
Log-Tabelle, 619, 2133, 2490, 2725
Severity-Filter, 2206, 2556
Ereignisse und Ereignis-ID, 929
ERPC-Variable, 3234
Erreichbare Teilnehmer, 3026
Ersatzwert
Reservemodul, 2896, 2975
Erweiterbare Baugruppenträger, 1157
Erweiterter Modus
Globale Firewall-Regeln, 394
Erweiterter Offline-/Online-Vergleich
automatische Gerätezuordnung, 323
erweiterter Schreibfilter (EWF), 1563
Erweiterungsbaugruppenträger
ER1 (S7-400), 1157
IM 46x, 1157
Regeln, 1158
ET 200AL, 3360, 3389
ET 200eco, 3360
ET 200eco PN, 3360, 3376
ET 200iSP, 3358
ET 200L, 3358
ET 200M, 2952, 3358, 3454
Definition, 3453
Steckplatzregeln, 2872
ET 200MP, 3358, 3440
Pulsweitenmodulation, 3452
ET 200pro, 3360
ET 200R, 3360
ET 200S, 2952, 3358
DPV1-Modus, 2899, 2978
Optionenhandling, 2895, 2898, 2974, 2977
Steckplatzregeln, 2889, 2970
Vergleichsstellen, 2890, 2970
ET 200S 1SI, 1670
ET 200S COMPACT, 3358
ET 200SP, 3358, 3382, 3389
Einsatzgebiet, 3382
ET 200AL, 3397, 3428
Beispielkonfiguration, 3426
Einsatzgebiet, 3425
ET 200SP, 3397
ET 200SP erweitern, 3397
ET 200SP mit ET 200AL-Modulen erweitern, 3397
ET-Connection, 3397
ET-Connection-Rack, 3397
Ethernet, 92, 94
Anzahl der Kommunikationsverbindungen, 178
Ethernet-Non-IP-Telegramme, 392
Ethernet-PC-CPs ziehen/stecken, 1567
Ethernet-Schnittstellen
EC31-HMI/RTX, 1562
EC31-RTX, 1562
EC31-RTX F, 1562
Ethernet-Statistiken
Schnittstellenstatistik, 2147, 2497
Telegrammfehler, 2149, 2150, 2500
Telegrammlänge, 2148, 2498
Telegrammtyp, 2149, 2499
ETX, 1649
Exportdatei für OPC UA, 1226
ExtendedKeyUsage, 1042

F

Facility, 433
FAQs
Anwenderseite als Startseite, 1336
Zertifikat herunterladen, 1329
FAQs zu einem Modul aufrufen, 35
Farbe
Diagnose von Ethernet-Leitungen, 3484
Diagnose von Ports, 3484

Fast Start-Up (PROFINET IO), (siehe Priorisierter Hochlauf)
F-CPU, 47
FDL-Verbindung
    Freier Layer 2 Zugang, 288
    mit Broadcast, 289
    mit Multicast, 290
    unspezifizierte, 288
Fehlende Versorgungsspannung, 3405, 3446
Fehlersichere Baugruppen
    Auswahl aus Hardware-Katalog, 47, 1153
Fehlerstatus, 623, 2135, 2493, 2729
Fehlerüberwachung
    Redundanz, 2248
    SIM-Karte, 2780
    Spannungsversorgung, 2246, 2578
    Verbindungszustandsänderung, 673, 2247, 2579, 2779
Feldgerät, 2926, 2983, 3036, 3037, 3078
Feste Referenztemperatur, 3449
Feste Telegrammlänge, 1643
FETCH/WRITE, 1034, 1422
FETCH/WRITE-Dienste
    Projektierung für ISO-Transport, 281
FETCH_RK, 1672
FETCH-Telegramm, 1703
File Browser, 1468
Filebrowser, 1317
File-Zuordnungstabelle, 3235, 3247
Filter
    Hardware-Katalog, 32
Firewall, 1321
    Dienstgruppen anlegen, 412, 415
    Dienstgruppen verwalten, 413, 416
    Firewall-Regeln, 393
    ICMP-Dienste definieren, 411
Firewall aktivieren
    SCALANCE S, 524, 526
Firmware, 1456
Firmware Card, 1371
Firmware-Aktualisierung
    aus dem Projektkontext, 3516
    aus der Projektnavigation, 3515
    Grundsätzliches, 3513
Firmware-Download über Web, 3242
Firmware-Laden bei S7-300/400-PROFIBUS-CPs über CP 5612 / CP 5622, 1576
Firmware-Lader - chinesische GUI, 3568
Firmware-Lader starten, 3571
Firmware-Update, 988
    aus dem Projektkontext, 3516
    aus der Projektnavigation, 3515
    Grundsätzliches, 3513
Firmware-Update Memory Card, 960, 1371
Flatterüberwachung, 3456
Flexible Speichergröße, 775
FM
    mit eigener MPI-Adresse, 1160
Folge-FETCH-Telegramm, 1711
Folge-SEND-Telegramm, 1707
Folge-SEND-Telegramme, 1707
Folgetelegramm, 1703
Forceauftrag, 959
Forward Delay, 629, 2742
Fragment, 1349, 1355, 1483, 1490
Freeport, 1663
    Codetransparenz, 1646
    Empfangspuffer, 1646
    Endekriterien, 1642
    Nachrichtenanfang, 1640
    Nachrichtenende, 1640
    Startkriterien, 1641
Freeport-Protokoll, 1639
FREEZE, 2889
Freie Anwenderereignisse, 943
Freier Layer 2 Zugang, 288
Frequenzmessen (CPU 31xC), 818
Frequenzmesser, 3373, 3376
Frequenzmesser Funktionsweise, 3373
FTP, 383, 384, 1034, 1422
FTP-Client
    S7-1500, 3254
FTP-Konfiguration, 3235
    File-Zuordnungstabelle, 3235
FTP−Server
    S7-1500, 3246
FTPS-Zertifikate, 376
Funktionalität
    PROFIBUS DPV0/DPV1, 2927
Funktionsweise
    Alarmerzeugung im I-Slave, 2905
    Direkter Datenaustausch, 2914

G

GBIT-Schnittstelle (CPU 1518-4 PN/DP), 3024
Gebertyp Wechsler, 3455
geografische Koordinaten, 642, 2187, 2534, 2755
Gerät
    einer Hardware-Konfiguration hinzufügen, 42
    gestörte Geräte, 3476
    kopieren, 57
    löschen, 63

Mehrfachauswahl, 54
umbenennen, 319
Umbennen, 61
verschieben, 59
Gerät tauschen, 60
Geräte vernetzen
Grundlagen zum Projektieren von S7-Netzwerken, 74
Kommunikationsarten, 74
Netze innerhalb eines Projekts, 75
Netzparameter bearbeiten, 81
Schnittstellenadressen bearbeiten, 82
Schnittstellenparameter bearbeiten, 81
Vernetzen bei nicht vorhandenem Subnetz, 76
Vernetzen bei vorhandenem Subnetz, 78
Vernetzen mehrerer Schnittstellen gleichzeitig, 76
Voraussetzungen, 81
Geräte-Information, 3476
Gerätename, 2989, 2996, 3147
Gerätename überschreiben ermöglichen, 3026
Gerätename, Automatische Inbetriebnahme (PROFINET), 2992
Gerätenamen
PROFINET, 3040
Gerätenummer, 2990
PROFINET, 3040
Gerätesicht
Ablage nicht gesteckter Baugruppen, 20
Baugruppe einfügen, 51
Baugruppenträger, 48, 1155
Eigenschaften bearbeiten, 65
Hardware- und Netzwerkeditor, 15
Parameter bearbeiten, 65
Signalboard einfügen, 1462
Gerätetausch, 60
Geräteübersicht
Adressbereich, 1160, 1161, 1458
Gerätevergleich, 72
Gerätezertifikat, 605, 2713
Gerätezertifikate, 1040
Gestörte Geräte, 3476
Get_Features, 1619, 1663
Getunnelte Kommunikation aktivieren
CP 343-1 Adv. / 443-1 Adv., 558
SCALANCE S, 525, 526
Gigabit-PROFINET Schnittstelle, 3024
Glättung, 3379, 3447
Globale Firewall-Regeln, 394
zuweisen, 396
Globale Paketfilter-Regeln, 396
GMRP, 2331
Grenzwert, 3405
Gruppen, 728, 2415, 2650, 2846
Gruppeneigenschaften, 466
Gruppennamen, 410, 413
GSD-Dateien
für I-Devices erzeugen, 3124
Geräte konfigurieren (PROFIBUS), 2926, 2983
GSD-Revisionen (PROFIBUS), 2922, 2979
Installieren, 2923, 2980
Löschen, 2924, 2981
Ungenutzte GSD-Daten löschen, 2925, 2982, 3038
GSD-Dateien (PROFINET), 3035
Ausgabestand ändern, 3039
Exportieren, 3078
Installieren, 3036
Löschen, 3037
GSDML, (Siehe GSD-Dateien (PROFINET))

H

Halbduplexbetrieb, 1620, 1676
HALT, 765
Handbücher zu einem Modul aufrufen, 35
Handshake Protocol, 328
Handshakeverfahren, 1632
Handshaking, 1410
Hardware
Eigenschaften bearbeiten, 65
Konfigurieren und Parametrieren, 41
Parameter bearbeiten, 65
Hardware interrupt, 1442
Hardware interrupt (S7-1500), 1068
Hardware Katalog
Auswahl der Hardware Komponente, 1519
Hardware RTS immer geschaltet, 1636
Hardware RTS immer ON, 1635
Hardware RTS immer ON, DTR/DSR ignorieren, 1635
Hardware- und Netzwerkeditor
Gerätesicht, 15
Netzsicht, 12
Topologiesicht, 22
Hardwareausgabestand, 618, 2131, 2488, 2724
Hardware-Datenflusskontrolle, 1633
Hardware-Diagnose, 3475
Hardwareeditor
Funktion, 9
Hardware-Katalog, 30
Inspektorfenster, 24
Komponenten, 10
Hardware-Erkennung, 44

Hardwaregesteuerte Datenflusskontrolle, 1409
Hardware-Katalog
Auswahl der HW-Komponente, 45
Auswahl fehlersicherer Baugruppen, 47, 1153
Auswahl zentraler Baugruppen, 1151
DP/DP-Koppler, 2873, 2957
DP-Slave, 2873, 2956
durchsuchen, 32
Gerät hinzufügen, 43
I-Slave, 2873, 2957
Task Card, 30
Hardware-Kennung, 1458, (siehe HW-Kennung)
Hardwarekompatibilität, 988
Hardware-Konfiguration, 964
Baugruppe hinzufügen, 51
Gerät hinzufügen, 42
Übersicht der Einstellungen, 11
HART, 3364
HART-Variablen
Aufbau, 3362
projektieren, 3361
Haupteintrag, 1663
Hello Time, 629, 2742
Herstellerkennung, 618, 2131, 2488, 2724
High-Speed-Analogmodule
Taktsynchronität, 3413
Hintergrund-OB (OB 90) (S7-300, S7-400), 918
HMI-Verbindung, 137, 1034, 1422
Hochlauf beschleunigen (PROFINET), 3094
Hochlauf von IO-Devices, 3090
HSC, 1401
HTML5, 1362
HTTP, 411
Laden/Speichern, 2195, 2545
HTTPS, 525, 805, 1328, 1467
HW-ID, (siehe HW-Kennung)
siehe HW-Kennung, 1458
HW-Kennung, 1161, 1163, 1167, 3387, 3437, 3444, 3468

I

I&M-Daten
in PROFINET-IO-Geräte laden, 3526
ICMP, 417, 591, 2703
I-Device, 2985, 3001, 3070, 3116
als GSD-Datei exportieren, 3124
Diagnoseadressen, 3074
Proxy-Gerät, 3199
Idle Line, 1642
IE/AS-i Link PN IO, 2985
IE/PB Link, 3028
IEC V2.2, 1015
IEC V2.3, 1015
IEEE 802.11n, 2464, 2604
Channel Bonding, 2466
Frame Aggregation, 2466
Guard-Intervall, 2466
Maximum Ratio Combining, 2465
MIMO, 2465
Spatial Multiplexing, 2465
IEEE 802.3, 392
IEEE-konformes LLDP, 1015
IEEE-Variable, 3363
iFeatures
iREF, 2473
IGMP, 2329
IGMP-Join, 2116
IGMP-Querier, 2329
IIN1.6, 1910
IKE-Einstellungen, 467
Im Betrieb wechselnde Partner-Ports, 3090
Impulsverlängerung, 3404, 3456
Index, 1553
Industrial Security, 1362
Industrie-PC, (siehe IPC)
Information
ARP-Tabelle, 618, 2132, 2489, 2725
DSL, 2737
Ereignis-Log, 619, 2133, 2490, 2725
Ethernet-Statistiken, 2148, 2498
Gruppen, 637, 2183, 2507, 2750
Hardware, 616, 2723
IPsec VPN, 626, 2739
IPv6-Nachbarschaftstabelle, 2133, 2489
LLDP, 624, 2155, 2731
Log Table, 622, 2728
Mobile, 2733
OpenVPN Client, 628, 2741
Rolle, 637, 2182, 2508, 2750
Security, 634, 636, 2180, 2182, 2505, 2507, 2747, 2749
Security Log, 621, 2727
SHDSL, 2735
SINEMA RC, 627, 2740
SNMP, 624, 2179, 2504, 2731
Software, 616, 2723
Spanning Tree, 630, 2136, 2494, 2743
Telegrammfehler, 2149, 2150, 2500
Telegrammtyp, 2149, 2499
Versionen, 2487
Versions, 617, 2130, 2723
WLAN Authentifizierungs-Log, 2492

Informationen
Inter AP Blocking, 2509
Initialisierung, 1796
Initialisierung (Anwenderdefinierte Webseiten), 1332, 1471
Inspektorfenster, 65
Hardware- und Netzwerkeditor, 24
Register Diagnose, 3476
Installationsvoraussetzung
PC-Station, 1520
Intelligenter DP-Slave, 2910, 2911, (Siehe I-Slave)
Intelligentes IO-Device, 3070
Inter AP Blocking
Basic, 2675
Erlaubte Adressen, 2676
Informationen, 2509
Konfigurieren, 2675
Interfaces (PROFINET), 1010
Interne Netzknoten
Diagnose, 494
konfigurieren, 480
Interne Subnetze manuell konfigurieren, 544, 582
Interne Vergleichsstelle, 3449
Interpolator-OB, 1084, 1450
IO access error (S7-1500), 1080
IO-Controller
Anzeige im IO-Device, 3009
Geräte und Baugruppen, 3007
PN-Schnittstelle, 3007
IO-Device
Aktualisierungszeit, 3017
Ansprechüberwachungszeit, 3019
Anzeige des IO-Controllers, 3009
vernetzen, 3013
IO-Link, 3189
IO-Routing, 3046
IO-System, 3010, 3138, 3141, 3144, 3148, 3151, 3152, 3159, 3160
erstellen, 3013
IP Access Control-Liste, 383
IP-ACL (IP Access Control-Liste), 3227
iPad, 1362
IP-Adresse, 92, 94, 2989, (Siehe Display)
Konfiguration, 712, 2830
IP-Adresse zuweisen
aus dem Projektkontext, 3518, 3575
Grundsätzliches, 3516, 3573
über "Erreichbare Teilnehmer", 3517, 3574
IP-Adressen
PROFINET, 3040
IP-Adressierung über DHCP, 218
IP-Adressparameter, 2990, 3001
IPC, 1517
iPCF
Einschränkung, 2471
Funktionsprinzip, 2469
Konfigurieren, 2526, 2527, 2677
PROFINET-Kommunikation, 2469
iPCF-HT
Funktionsprinzip, 2469
Konfigurieren, 2680
iPCF-MC
PROFINET-Kommunikation, 2469
IP-Dienste, 410
IP-Mapping, 2519
IP-Netzknoten manuell konfigurieren, 543
IP-Paketfilter-Regeln, 417
IP-Parameter, 3490
IP-Regelsätze, 394
iPRP
Information, 2529
Konfigurieren, 2685
IPsec VPN
NETMAP, 604, 2710
Source-NAT, 604, 2710
IPsec-Einstellungen, 467
IPsec-Verfahren, 606, 2714
IP-Subnetz, 94
IP-Subnetzadresse, 1010
IPv4
Notation, 588, 2694
OSPf, 2109
VRRP, 2108
VRRPv3, 2108
IPv4 Routing
Routing-Tabelle, 625, 2732
IPv4-Adresse, 588, 2694
IPv4-Adressen, 426
IPv4-Routing
MSDP-Cache, 2169
NAT-Übersetzungen, 2165
PIM BSRs, 2168
PIM-Nachbarn, 2166
PIM-Routen, 2167
PIM-RPs, 2168
PIM-Schnittstellen, 2166
IPv4-Routing über Rückwandbus (CM/CP 154x-1), 3472
IPv6
E-Mail, 3473
FETCH/WRITE, 3473
FTP-Client, 3473
FTP-Server, 3473
Notation, 2077, 2458, 3472

OSPFv3, 2113
SNMP, 3473
Verwendung beim CP 1543-1, 3473
VRRPv3, 2108
IPv6 Routing
OSPFv3 AS-Scope LSDB, 2174, 2175
OSPFv3 Link-Scope LSDB, 2177
OSPFv3 Neighbors, 2171
OSPFv3 Virtual Neighbors, 2173
OSPFv3-Schnittstellen, 2170
Routing-Tabelle, 2177
IPv6- Routing
Routing-Tabelle, 2169, 2503
IPv6-Adressen, 427
IPv6-Routing
Default-Routen, 2537
statische Routen, 2387
IP-Zugriffschutz, 220
IP-Zugriffsschutz, 3226, 3229
IP Access Control-Liste, 3228
IQ-Sense, 3457
iREF, 2473
Konfigurieren, 2686
IRT, 3053
Aufbauempfehlungen, 3031
IRT (Isochronous Realtime), 3048, 3053, 3057
ISAKMP, 484
I-Slave, 2876, 2901, 2910, 2911
Alarm senden mit SALRM, 2905
CP konfigurieren, 2908
CPU konfigurieren, 2908
Datenaustausch, 2876
Datenzugriff, 2969
Diagnoseadressen, 2902
Direkter Datenaustausch, 2877
DP-Master, 2876
DP-Slave konfigurieren, 2909
Geräte, 2873
Hardware-Katalog, 2873, 2957
konfigurieren, 2876, 2902
Proxy-Gerät, 3197
Prozessalarm auslösen, 2936
Steckplatzmodell der DP-Slaves, 2933
I-Slave, 2958, 2968
Datenaustausch, 2958
konfigurieren, 2969
I-Slave (Intelligentes DP-Slave)
Alarmverhalten, 2912
Diagnose, 2912
isochroner PROFIBUS, (Siehe Taktsynchronität)
Isochronous Real-Time-Kommunikation (IRT), 2985
ISO-on-TCP
Merkmale, 113
TSAP, 132
ISO-Protokoll, 542

J

jQuery, 1362

K

kalibrieren, 3523
Überblick, 3522
Kalibriervorgang abbrechen, 3525
Kaltstart, 757
Kaltstart (S7-300, S7-400), 919
Kanaladresse, (Siehe Baugruppen-Anfangsadresse)
Kanaldiagnose, 3566
kanalspezifische Diagnose, 3565
Kartentyp, (siehe Memory Card)
Kaskadierzählfunktion, 3369
Katalog, (siehe Hardware-Katalog)
Keep-Alive-Telegramme, 3213
Kennungen von Elementen und Bausteinen, 866
KEY-PLUG, 2251, 2254
Formatieren, 676, 2253, 2586, 2784
iFeatures, 677, 2587, 2785
KeyUsage, 1042
Kommunikation, 1034, 1422, 1559
Anzahl der Verbindungen (PROFINET/ PROFIBUS), 178
Open User Communication, 220
Schnittstellen, 1559
Verbindungsressourcen, 807
Zyklusbelastung, 1000, 1398
Kommunikation über PUT/GET-Anweisung
Grundlagen zur PUT/GET-Anweisung, 160
Verbindung anlegen und parametrieren, 165
Verbindung löschen, 166
Verbindungsparametrierung starten, 164
Verbindungsparametrierung Übersicht, 161
Voraussetzungen, 161
Kommunikationsbereitschaft erklären
- passiver Verbindungsaufbau, 217
Kommunikationsdiagnose, 3499, 3566
Kommunikationseigenschaften einer Baugruppe, 3558
Kommunikationsereignisse, 939
Kommunikationsfehler-OB (OB87) (S7-300, S7-400), 915
Kommunikationslast, 988, 1000, 1390, 1398

Kommunikationslast einstellen, 91
Kommunikationsmodule, 1407
    Eigenschaften, 1407
Kommunikationsmöglichkeiten, 2661
Kommunikations-Port
    konfigurieren, 1408
Kommunikationsprotokoll
    festlegen, 1411
    Überblick, 1410
Kommunikationsprozessor
    SIMATIC NET PC Software, 1564
    STEP 7 V5 + STEP 7 Professional, 1575
Kommunikationsverbindung, 200
Kompatibilität
    S7-1200-CPU-Firmwareversionen, 1456
Komponente stecken
    PC-Station, 1565
Komprimieren
    Speicher in Online- und Diagnosesicht, 3569
Konfigruationssteuerung für IO-Systeme, 3163
Konfiguration
    Dezentrale Peripherie, 2871
    DP-Slave, 2872, 2885, 2966
    DP-Slave, einfach, 2874, 2957
    Hardware, 41
    in Gerät laden, 3191
    in PG/PC laden, 3192
    I-Slave, 2876
    PC-Kommunikationsschnittstelle, 1552
    PPP, 697
    Software Controller, 1552
    Zentrale Baugruppen, 1138
Konfigurationsdatei, 1527, 1528
Konfigurationsmodus, 2184, 2531
Konfigurationssteuerung, 1166, 1177, 1186, 1499, 1508, 3137, 3384, 3389, 3431, 3442, 3464
Konfigurationssteuerung bei ET 200AL, 3431
Konfigurationssteuerung bei ET 200MP, 3442
Konfigurationssteuerung bei ET 200pro, 3464
Konfigurationssteuerung bei ET 200SP, 3384
Konfigurationssteuerung für IO-Systeme, 1190, 3148
Konfiguratitionssteuerung für IO-Systeme, 3137
Konfigurieren
    CP als I-Slave, 2908
    CPU als I-Slave, 2908
    Diagnoseadressen im I-Device, 3076
    Diagnoseadressen im I-Slave, 2904
    Direkter Datenaustausch, 2917
    DP-Slave als I-Slave, 2909
    I-Slave, 2902
    I-Slave, 2969
    PC-Station, offline, 1528
Konfigurieren des Netzes über Ethernet, 92
    Einstellen der IP-Adresse, 92
    Einstellen der Subnetzmaske, 93
    Netze verbinden, 94
    Private Subnetze bilden, 93
    Zusammenhang IP-Adresse und Subnetzmaske, 93
Konsistenz
    Steckplatzregeln, 50
Konsistenz von Daten, 197
Konsistenzprüfung, 540
    lokal, 374
    projektweit, 374
Kontextfilter, 32
konvertierter Name (PROFINET), 2996
Kopieren
    Hardware-Komponente, 57
Kopplung von Baugruppenträgern, 1156, 1157
Kurzschluss nach L+, 3404
Kurzschluss nach M, 3403

L

LACP, 2317
Laden
    Dezentrale Peripherie, 3194
    Gerätekonfigurationen, 3193
    GSDML, 3198
    HMI-Geräte, 3201
    I-Device, 3199
    IE/PB-Link, 3200
    in Gerät, 3191
    in PG/PC, 3192
    I-Slave, 3197
    Konfigurationen mit PROFIBUS, 3196
    Konfigurationen mit PROFINET, 3198
    Konfigurationen mit Webserver, 3196
    Online verbinden, 3202
    Proxy-Gerät, 3197, 3199
    Shared Devices, 3201
    Subnetze, 3195
    taktsynchrone Applikationen, 3191, 3197
Ladespeicher, 771, 775, 964, 1373
Landessprache für Webserver, 1003, 1326
Layer 2, 392, 698, 2817
    MLD (IPv6), 2333
Layer 3, 392, 2254
Layer 3 (IPv6), 2385
    Konfiguration, 2385
LDB-Konfiguration, 1579

Lebensdauer von Zertifikaten, 463
Lebenszeichentelegramm, 3213
LEDs, 1555
LEDs der Hardware-Plattform, 1555
LED-Zustand, 3562
Leistungsbilanz, 3441
Leistungsdaten, 3560
Leitungsbruch, 3379
Lernfunktionalität, 480
Lernmodus, 541
Line Break, 1642
Linearachse (CPU 31xC), 834, 842
Link Check, 2145
Link Check Status, 2145
Link Layer Discovery Protocol (LLDP), 1015
LLDP, 383, 384, 624, 2155, 2731, 3026
LLDP (Link Layer Discovery Protocol), 3026
LLDP (siehe Link Layer Discovery Protocol, 1015
Log Table
    Firewall Log, 622, 2728
    Security Log, 621, 2727
Log-Datei öffnen
    Paketfilter-Ereignisse, 497
    Sicherheits-Ereignisse, 496
    System-Ereignisse, 495
Logging, 487
    CP 343-1 Adv. / 443-1 Adv., 558
    SCALANCE S, 525, 526
Login Webserver, 1465
Log-Tabelle
    Ereignis-Log, 619, 2133, 2490, 2725
    WLAN Authentifizierungs-Log, 2492
Lokaldatenbit (L), 1374
Lokaldatenbyte (LB), 1374
Lokaldatendoppelwort (LD), 1374
Lokaldaten-Stack, 779
Lokaldatenwort (LW), 1374
Lokale Benutzer, 724, 2411, 2646, 2842
Lokaler Zugriffsschutz, 1038
Lokales Logging, 488
Löschen
    Hardware-Komponente, 63
LSAP-Konfiguration, 1576

M

MAC-Dienste, 413
Machine Tailoring, (siehe Konfigurationssteuerung für IO-Systeme)
    siehe Konfigurationssteuerung für IO-Systeme, 3148
MAC-Netzknoten manuell konfigurieren, 544
MAC-Paketfilter-Regeln, 423
MAC-Regelsätze, 394
Main, 1433
Main (S7-1500), 1057
Man-In-The-Middle Attack, 328
Manueller Wiederanlauf, 791
Manuelles Fragment, 1355, 1490
Master, 1725
Maximalausbau einer S7-1500, 1144
Maximale Zeichenanzahl, 1643
Maximale Zykluszeit, 999, 1396
MC-Interpolator-OB, 1084, 1450
MC-PostServo-OB, 1086, 1449
MC-PreServo-OB, 1085, 1449
MC-Servo
    parametrieren, 1083, 1455
MC-Servo-OB, 1082, 1448
MD5, 459
Media Redundancy Protocol (MRP), 3100
Medienkonverter, 3016
Medienredundanz
    projektieren, 3102
Medienredundanz (MRP), 2985, 3098, 3107
mehrfach einsetzbare IO-Systeme, 3137
Mehrfach einsetzbares IO-System, 3138, 3141, 3144, 3147
Mehrfachauswahl
    Hardware-Komponente, 54
Mehrfachselektion, 49
Mehrpunkt, 1725
Mehrsprachigkeit, 1003
Meldetexte, 1715
    Seitenlayout, 1715
    Steuerzeichen, 1715
    Steuerzeichentabelle, 1715
    Zeichenwandeltabelle, 1715
Meldetexte speichern, 1715
Meldungen
    sich bei CPU dafür anmelden, 3569
Meldungen anzeigen, 3203
Meldungsanzeige, 3203
    Ansicht "Aktive Meldungen", 3204
    Archiv exportieren, 3206
    Archivansicht, 3204
    Aufbau der Meldungen in der Ansicht "Aktive Meldungen", 3204
    Aufbau der Meldungen in der Archivansicht, 3205
    Meldung quittieren, 3208
    Meldungen empfangen, 3206
    Meldungsarchiv vollständig leeren, 3205
    Tastaturbedienung, 3210
Meldungseinstellungen, 988

Memory Card, 769, 959, 964, 1371
formatieren, 3513
Merker, 966
Merkerbit (M), 1374
Merkerbyte (MB), 1374
Merkerdoppelwort (MD), 1374
Merkerwort (MW), 1374
Messbereich, 3418
Messfenster (TOR), 3376
MIB, 383
Micro Memory Card, 767
Mindestzykluszeit, 957, 999, 1369, 1396
Mirroring, 2288, 2292
Mischbetrieb, 478
Mischbetrieb ET 200SP mit ET 200AL, 3397, 3428
MLD, 2333
Mobile
Information, 2733
Mobilfunkanbieter
Schnittstellen, 2809
MODB_341, 1674
MODB_441, 1674
Modbus
Exception Code, 1655
RS232-Signale, 1636
Telegrammende, 1655
Modbus_Comm_Load, 1619, 1665
Modbus_Master, 1619, 1665
Modbus_Slave, 1619, 1665
Modbus-Anweisungen, 1665
Modbus-Kommunikation, 1653
Modul-Anfangsadresse, (Siehe Baugruppen-Anfangsadresse)
Module
Kommunikationszustand, 493
Modulinternes Shared Input (MSI), 3133
Modulinternes Shared Output (MSO), 3133
Modulversion
aktualisieren, 63
Mono-Master-System, 2874, 2958
MPI
Adressvergabe, 1160
MPI/DP-Schnittstelle, 3559
MPI-Schnittstelle, 3559
MRP, 3100
MRP (Media Redundancy Protocol), 3098
MRP-Domain, 3107, 3493
MRP-Domain-Eigenschaften, 3493
MRP-Interconnection, 2145
Funktionsprinzip, 2105
Topologie, 2105
MSDP, 2117, 2382
Peer, 2383
MSDP-Cache, 2169
MSI, 973
MSI (Modulinternes Shared Input), 974
MSI/MSO, 3133
MSI-Submodul, 974
MSO, 973
MSTP, 2311, 2637
Multicast, 243, 290, 435, 2327
bei UDP, 275, 276
Multi-Master-Systeme, 2875
Multiple Spanning Tree, 2311, 2637
Multiplex-/Synchronbetrieb, 3462

N

Nachricht
Beginn festlegen, 1414
Ende festlegen, 1415
senden, 1413
Nachrichtenlänge aus Nachricht, 1643
Nachrichten-Zeitüberschreitung, 1643
NAK, 1649
NAPT, 2468
konfigurieren, 714, 2348, 2643, 2832
NAT, 2468
1-to-1 NAT, 718, 2836
konfigurieren, 714, 2345, 2347, 2641, 2832
Masquerading, 603, 2709
NAPT, 603, 2709
NAT-Traversal, 609, 2716
NETMAP, 604, 2710
Source-NAT, 604, 2710
NAT/NAPT
Routing, 434
NAT-Traversal, 609, 2716
NAT-Übersetzungen, 2165
NCM S7-Diagnose starten, 3567
NCM-Verbindung
Verbindungsdetails, 3576
Negotiation, 694, 2239, 2590, 2806
Nettodatenmenge, 3225
Netz-ID, 534
Netzsicht
Bedeutung, 234
E/A-Kommunikation, 2998
Gerät hinzufügen, 43
Hardware- und Netzwerkeditor, 12
Netzübergang, 92

Netzübersicht
Grundfunktionen, 79
Grundfunktionen zur Bearbeitung der Netzübersichtstabelle, 79
Netzüberwachung, 650, 2765
Netzwerkeditor
Funktion, 9
Hardware-Katalog, 30
Inspektorfenster, 24
Komponenten, 10
Netzwerklastenausgleich-Manager (NLB), 1606
Netzwerk-Syslog, 488
Neustart, 791, 2192, 2540, 2756
Neustart (Warmstart), 757, 954, 1367
Neustart (Warmstart) (S7-300, S7-400), 919
Neustart der CPU, 1796
Nicht gesteckte Baugruppe, 20
Nicht spezifizierte CPU, 44
Normdevice, 3035
Normslave, 2926, 2983
NTP, 2327
Server, 663, 2235, 2776
NTP (secure), 453
NTP-Server, 453
NTP-Server exportieren, 457
NTP-Server importieren, 457
Nur HTTPS-Server, 2183
Nutzdaten, 784
Adressen, 2933
Bereich, 787, 971, 1379
Steckplätze, 2933
Nutzdatengröße IO-Device, 3225

O

OB
Ereignisquelle, 981
Ereignisse und OBs, 980, 1380
Übersicht, 980, 1380
OB 1, 870, 1433
OB 10 bis OB 17 (S7-300, S7-400), 871
OB 100
OB 101 und OB 102 (S7-300, S7-400), 919
OB 121 (S7-300, S7-400), 924
OB 122 (S7-300, S7-400), 927
OB 1-PA (OB 1-Prozessabbild), 863
OB 20 bis OB 23 (S7-300, S7-400), 874
OB 30 bis OB 38 (S7-300, S7-400), 876
OB 40 bis OB 47 (S7-300, S7-400), 878
OB 61 (S7-300, S7-400), 887
OB 61...64, 2942
OB 62 (S7-300, S7-400), 887
OB 63 (S7-300, S7-400), 887
OB 64 (S7-300, S7-400), 887
OB 80, 1443
OB 80 (S7-300, S7-400), 888
OB 81 (S7-300, S7-400), 891
OB 82, 1444
OB 82 (S7-300, S7-400), 894
OB 83, 1445
OB 83 (S7-300, S7-400), 897
OB 84 (S7-300, S7-400), 902
OB 85, 778
OB 85 (S7-300, S7-400), 904
OB 86 (S7-300, S7-400), 908
OB 87 (S7-300, S7-400), 915
OB 88 (S7-300, S7-400), 916
OB für herstellerspezifische Alarme (OB 57) (S7-300, S7-400), 885
OB für herstellerspezifischen Alarm, 1438
OB für herstellerspezifischen Alarm (S7-1500), 1064
OB für profilspezifischen Alarm, 1438
OB für profilspezifischen Alarm (S7-1500), 1064
Oberflächensprache für Webserver, 1003
OB-Priorität, 981
OB-Startereignis, 980
OB-Startinformation, 1056
Offline, 368
Offline-/Online-Vergleich
automatische Gerätezuordnung, 320
Online, 368
Hardware-Erkennung, 44
Online Device, 3476
Online- und Diagnosesicht, 3475
Online- und Diagnosesicht starten, 3485
Online verbinden, 1010, 3530
Online-Anzeige
Farbe orange, 3485
Online-Diagnose, 490
Online-Tools, 3475, 3487
Online-Verbindung trennen, 3531
Online-Zuweisung (PROFINET-Gerätenamen), 2998
OPC Item spezifische Zugriffsrechte vergeben, 1588, 1596
OPC UA
DB-Variablen, 1222
Einführung, 1192, 1193, 1194
Endpunkte, 1217
Identifier, 1195
Namespace, 1195
OpenSSL, 1209
PLC-Variablen, 1222
Schichtmodell, 1213
Secure Channel, 1213

Security-Einstellungen, 1217
Sichere Verbindung, 1213
Sicherheitsmechanismen, 1203
Signieren und Verschlüsseln, 1206
X.509-Zertifikate, 1208
Zertifikatsgenerator, 1208
OPC UA-Client
Authentifizierung, 1309
Grundlagen, 1200
Zertifikat, 1240, 1307
OPC UA-Redundanz, 1605
OPC UA-Server
Abtastintervall, 1233
Adressierung, 1229
Adressraum, 1196
Applikationsname, 1228
Authentifizierung, 1248
Grundlagen, 1215
Inbetriebnahme, 1227
Leistungssteigerung, 1225
Performance, 1225
Runtime-Lizenzen, 1251
Schreib- und Leserechte, 1222
Security-Einstellungen, 1238, 1245
Sendeintervall, 1233
Server-Zertifikat anpassen, 1242
Server-Zertifikat erzeugen, 1234
Subscription, 1231
TCP-Port, 1231, 1232
XML-Exportdatei, 1226
OPC UA-Client, 1605
OPC UA-Server, 1605
OPC-Eigenschaften", 258, 264, 270
OPC-Konfiguration, 1577
OPC-Server
als SMTP-Server nutzen, 295
Steckplatzregeln, 1565
Zentraler Kommunikationsdienstleister, 1514
OPC-Server - PLC-Meldungen, 1608
OPC-Serverfunktionen
Datensatzfunktionen, 1578
für DP Master Klasse 2, 1578
Open User Communication, 220
Allgemein, 98
Anweisungen, 99
Parameterwerte ändern, 131
Port-Nummern, 130
Rücklesbarkeit, 131
TCON_FDL, 126
TCON_IP_RFC, 124
TCON_IP_v4, 118
TCON_IP_V4_SEC, 119
TCON_Param, 115
TCON_QDN, 121
TCON_QDN_SEC, 122
TSAP, 132
über Industrial Ethernet, 220
Verbindung anlegen, 107
Verbindung löschen, 111
Verbindungsaufbau, 99
Verbindungsbeschreibung, 115, 117, 119, 120, 122, 123
Verbindungsparameter, 103
Verbindungsparametrierung, 99, 101
Verbindungsparametrierung starten, 106
Verwendete Protokolle, 112
OpenSSL, 1209
Operandenbereich, 775, 1374
anzeigen, 3560
Optimierter Bausteinzugriff (FTP - S7-1500), 3251
Optionales IO-Device, 3151
Optionenhandling, 2895, 2898, 2974, 2977, (Siehe Konfigurationssteuerung (ET 200MP)), (Siehe Konfigurationssteuerung (ET 200SP))
Optionenhandling (siehe Konfigurationssteuerung), 1177, 1499, 3389, 3431, 3464
Optionenhandling für Netzwerke, (siehe Konfigurationssteuerung für IO-Systeme)
Organisationsbaustein, 1172
Organisationsbaustein (OB)
alle OBs im Überblick, 867
anzeigen, 3560
Baugruppenträgerausfall-OB (OB 86) (S7-300, S7-400), 908
Bearbeitungsabbruch-OB (OB 88) (S7-300, S7-400), 916
CPU-Hardwarefehler-OB (OB84) (S7-300, S7-400), 902
Diagnosealarm-OB (OB 82) (S7-300, S7-400), 894
Hintergrund-OB (OB 90) (S7-300, S7-400), 918
Kaltstart-OB (OB 102) (S7-300, S7-400), 919
Kommunikationsfehler-OB (OB 87) (S7-300, S7-400), 915
Neustart (Warmstart)-OB (OB 100) (S7-300, S7-400), 919
Peripheriezugriffsfehler-OB (OB 122) (S7-300, S7-400), 927
Programmablauffehler-OB (OB85) (S7-300, S7-400), 904
Programmierfehler-OB (OB 121) (S7-300, S7-400), 924

Prozessalarm-OBs (OB 40 bis OB 47) (S7-300, S7-400), 878
Statusalarm-OB (OB 55) (S7-300, S7-400), 880
Stromversorgungsfehler-OB (OB 81) (S7-300, S7-400), 891
Taktsynchronalarm-OB (OB 61 bis OB 64) (S7-300, S7-400), 887
Uhrzeitalarm-OBs (OB 10 bis OB 17) (S7-300, S7-400), 871
Update-Alarm-OB (OB 56) (S7-300, S7-400), 882
Verzögerungsalarm-OBs (OB 20 bis OB 23) (S7-300, S7-400), 874
Weckalarm-OBs (OB 30 bis OB 38) (S7-300, S7-400), 876
Wiederanlauf-OB (OB 101) (S7-300, S7-400), 919
Zeitfehler-OB (OB 80) (S7-300, S7-400), 888
Ziehen/Stecken-OB (OB83) (S7-300, S7-400), 897
Zyklisches Programm (OB1) (S7-300, S7-400), 870
Organsisationsbaustein (OB)
OB für herstellerspezifische Alarme (OB 57) (S7-300, S7-400), 885
Ortskennzeichen, 789
Ortszeit, 988, 1390
OSPF
Area Range, 2370
Areas, 2369
Bereiche, 2109
Link State Advertisment, 2109
OSPFv2 LSDB (Information), 2163
OSPFv2 Nachbarn, 2160
OSPFv2 Virtual Neighbors, 2162
OSPFv2 Virtuelle Nachbarn, 2162
OSPFv2-Schnittstellen, 2159
Router, 2109
Router-Status, 2109
Schnittstellen, 2371
Virtual LInks, 2374
OSPF (IPv6)
Area Range, 2402
Areas, 2401
Bereiche, 2401
OSPFv3 AS-Scope LSDB, 2174, 2175
OSPFv3 Interfaces, 2403
OSPFv3 Link-Scope LSDB, 2177
OSPFv3 Neighbors, 2171
OSPFv3 Virtual Neighbors, 2173
OSPFv3-Schnittstellen, 2170
Virtual LInks, 2405
OSPFv2
Konfiguration, 2367
OSPFv3
Konfiguration, 2400
OUC, (Siehe Open User Communication)
Overlay-Icon, 3484
Oversampling, 3413

P

P_PRINT, 1671
P_PRINT341, 1672
P_RCV, 1671
P_RCV_RK, 1672
P_RESET, 1671
P_SEND, 1671
P_SND_RK, 1672
P12, 347
P3964_Config, 1619
Packen von Adressen, 2893, 2972
Paketfilter-Log, 497
Panning, (Siehe Ansicht verschieben)
Parameter
für CPU, 1390
Parameter (CPU), 988
Parameter der Uhr, 804, 1024, 1025
Parameterdaten, 785
Parametrierung
Hardware, 24, 41
Partnerport
Information über Überwachung, 3016
Passwort, 1038
Empfehlungen zur Passwortvergabe, 1327
für CPU-Zugriffsschutz, 988, 1390
Optionen, 2417, 2645
Passwortschutz, 809, 988, 1026, 1031, 1390, 1418, 1419, 1465
PC-CP, 367
PC-Station
Gemeinsamkeiten der PC-Station V2.0 und V1.0, 1516
Installationsvoraussetzung, 1520
Komponente stecken, 1565
offline konfigurieren, 1527, 1528
OPC-Konfiguration, 1567, 1576, 1577
Steckplatzregeln, 1565
Submodule stecken, 1566
Verbesserungen von PC-Station V2.0 auf V1.0, 1516
Verbesserungen von PC-Station V2.0 auf V2.1, 1516

Verbesserungen von PC-Station V2.1 auf V2.2, 1516
Verbindungen, 216
Wahl der Version, 1510
XDB-Konfiguration, 1585
PC-System
Auswahl aus Hardware Katalog, 1519
konfigurieren, 1519
PEM, 347
Periodendauer (CPU 31xC), 823
Peripherie
direkter Zugriff auf, 971, 1379
direkter Zugriff auf ~, 864
Peripherieadresse, 971, 1160, 1161, 1379, 1458
Peripherieadressierung, 787
Peripherieausgang, 1374
Peripheriedatenbereich
Datensätze der Baugruppe, 785
Diagnosedaten, 785
Nutzdaten, 784
Parameterdaten, 785
Zugriff, 784
Peripherieeingang, 1374
Peripheriezugriffsfehler, 778, 1377
Peripheriezugriffsfehler (PZF), 778
Peripheriezugriffsfehler-OB (S7-1500), 1080
Peripheriezugriffsfehler-OB (S7-300, S7-400), 927
Phasenverschiebung (S7-1500), 1066
PIM
Protokoll, 2379
RP-Kandidat, 2381
RP-Statisch, 2380
Schnittstellen, 2379
PIM BSRs, 2168
PIM-Nachbarn, 2166
PIM-Schnittstellen, 2166
Ping, 2255, 2589
Pip (Datentyp), 1174
Platzhalterbaugruppe, 1140
PLC-Variablen, 1337, 1338, 1341, 1475, 1476, 1477
PLC-Variablen lesen
String- oder Character-Variablen in Ausdrücken, 1340
Variablen vom Typ String und Character, 1339
PLUG, 2254
C-PLUG, 675, 2584, 2782, (C-PLUG)
KEY-PLUG, 677, 2587, 2785, (KEY-PLUG)
PN/PN Coupler
Ethernet-Subnetze koppeln, 3187
Gruppieren, 3187
Port, 94
102 (S7-Protokoll - TCP), 411
123 (NTP), 436, 452
161 (SNMP), 436
20/21 (FTP), 411
443 (HTTPS), 436
4500 (IPsec), 436
500 (IPsec), 436
500 (ISAKMP - UDP), 484
514 (Syslog), 436
80 (HTTP), 411
8448 (Security-Diagnose), 572, 576
Konfiguration, 695, 2241, 2591, 2806
Link Check, 2145
Portkonfiguration, 666, 672, 693, 2238, 2246, 2590, 2592, 2805
umbenennen, 319
verschalten, 3015
Port_Config, 1619, 1663
Port-Diagnose
SFP-Diagnose, 679, 2263
Portkonfiguration, 672, 2246, 2592
Port-Nummern, 130
Port-Optionen, 3025
Autonegation aktivieren, 3023
Übertragungsrate/Duplex, 3021
Überwachen, 3021
Ports verschalten
grafische Ansicht, 316, 318
tabellarische Ansicht, 317, 319
Portverschaltung ändern
grafische Ansicht, 316
Positionieren (CPU 31xC), 828
PostServo-OB, 1086, 1449
Powersegment, 1143
PPP
Konfiguration, 697
Overview, 696
PreServo-OB, 1085, 1449
Pre-shared keys, 465
Priorisierter Hochlauf, 2985, 3090
Priorisierter Hochlauf (PROFINET), 3094
Priorisierung, 2271
Priorität, 2271
Prioritätsklasse, 873, 934
Private Key, 324
ProDiag-OB, 1081
ProDiag-Organisationsbaustein, 1081
PROFIBUS, 88, 2939
Anzahl der Kommunikationsverbindungen, 178
Direkter Datenaustausch, 2915
DP-Mastersystem anlegen, 2918
DP-Schnittstelle umschalten, 2882
DPV0/DPV1, 2926

ET 200S im DPV1-Modus, 2899, 2978
Kopplung mit PROFINET, 2985
unterschiedliche Funktionalität, 2928
PROFIBUS DP, 2878, 2886, 2960, 2966
PROFIBUS DPV1, 800
PROFIBUS Leitungskonfiguration
Optischer Ring, 91
PROFIBUS Profile, 88
Auswirkungen auf die Übertragungsgeschwindigkeit, 89
Bedeutung der Profile, 89
Unterschiedliche Profile am selben Subnetz, 88
PROFIBUS Subnetz konfigurieren
Bedeutung der Busparameter, 87
Busparameter einstellen, 86
Parameter aufeinander abstimmen, 86
PROFIdrive Modus, 1578
PROFIenergy, 2985, 3115, 3116
Profil, (Siehe Busprofil)
Profile, 1438
Profile (S7-1500), 1064
PROFINET, 94, 542, 1015, 2079, 2467, 2581, 3043, 3048, 3053, 3084, 3107
Anzahl der Kommunikationsverbindungen, 178
Diagnose, 3047
Gerätenamen, 3040
Gerätenamen zuweisen, 2998
Gerätenummer, 3040
Gerätetausch ohne Wechselmedium ermöglichen, 3026
IP-Adressen, 3040
Kopplung mit PROFIBUS, 2985
Medienredundanz, 3098
optimieren mit IRT, 3031
optimieren mit RT, 3029
PROFINET IO, 2079, 2467, 2983, 2985, 2989, 2990, 3007, 3013, 3070
Port-Optionen, 3021
PROFINET IO-Device
Namensvergabe im Dialog, 3521
Namensvergabe in der Online- und Diagnosesicht, 3519, 3520
PROFINET IO-System, 3010
anlegen, 3006
hervorheben, 3009
PROFINET IRT, 3057
PROFINET RT-Klasse, 2984
PROFINET-Gerätename, 2996, 3147, (Siehe Display)
PROFINET-IO-Device
Namen vergeben, 3518
PROFINET-Schnittstelle, 988, 1010, 1390, 3001, 3490, 3560
PROFINET-Schnittstellenparameter zurücksetzen, 3571
Programm
zyklisches (S7-300, S7-400), 870
Programmablauffehler-OB (OB 85) (S7-300, S7-400), 904
Programmbearbeitung, 1112
zyklisch, 1433
Programmierfehler-OB (S7-1500), 1077
Programmierfehler-OB (S7-300, S7-400), 924
Programmierung
Modbus, 1664
PtP, 1661
USS, 1666
Programminformation, 965
Programming error (S7-1500), 1077
Programmkarte, 1371
Projekt
laden, 1523
Projektieren
HART-Variable, 3361
Projektnavigation
Gerät hinzufügen, 42
Protokoll, 410
Protokolle der Kommunikationsmodule, 1617
Protokollparameter, 1719
Proxy-ARP, 552
Proxy-Gerät, 3197, 3199
Prozedur 3964(R), 1648, 1694
Blockprüfzeichen, 1695
Daten empfangen, 1699
Daten senden, 1696
Priorität, 1649, 1695
Steuerzeichen, 1649, 1694
Prozedur 3964R
Blockprüfzeichen, 1649
Prozessabbild, 971, 1170, 1172, 1174, 1379
Aktualisieren, 778, 1377
der Ein- und Ausgänge, 863
Grundlagen, 778, 1376
Peripheriezugriffsfehler, 778, 1377
Prozessalarm
Beispiel, 983
Einstellungen, 796
Funktionsweise, 1386
I-Slave, 2936
OB 40 bis OB 47 (S7-300, S7-400), 878
Prozessalarm (S7-1500), 1068

Prozessalarm-OB
Beschreibung, 1442
parametrieren, 1091, 1454
Prozessalarm-OBs (S7-1500), 1068
Prozessbetrieb, 811
Prozessdatenbereich
Baugruppenadressierung, 786, 1378
Baugruppen-Anfangsadresse, 1378
Peripherieadressierung, 787
Prozessreaktionszeit verkürzen, 2952
PTC-Widerstand, 3415
PtP-Anweisungen, 1663
Public Key, 324
Pufferung, 775
Pull or plug of modules (S7-1500), 1075
Pulsweitenmodulation
ET 200MP, 3452
Pulsweitenmodulation (CPU 31xC), 822
Punkt zu Punkt, 2090, 2480
Punkt-zu-Punkt, 1725
Punkt-zu-Punkt-Kommunikation (PtP), 1406
Freeport-Protokoll, 1407
Punkt-zu-Punkt-Kopplung, 1617
Punkt-zu-Punkt-Kopplung ((3964(R), CPU 31xC), 852
Punkt-zu-Punkt-Kopplung ((ASCII), CPU 31xC), 845
Punkt-zu-Punkt-Kopplung ((RK 512), CPU 31xC), 857
Punkt-zu-Punkt-Kopplung (CPs, ET 200S 1SI), 1670
PUT/GET, 1034, 1422
PUT-Telegramm, 1703
PWM
ET 200MP, 3452
PZF (Peripheriezugriffsfehler), 778

Q

QI (Quality Information), (Siehe Wertstatus)
QoS, 2271
Querverkehr, (Siehe Direkter Datenaustausch)
QV, (Siehe Direkter Datenaustausch)

R

R, 88
Rack or station failure, 1446
Rack or station failure (S7-1500), 1076
RADIUS, 729, 2418, 2653, 2847
RCV_PTP, 1672
RDREC, 1166
Reaktionstelegramm, 1703
Aufbau und Inhalt, 1705
Receive_Config, 1619, 1663
Receive_P2P, 1619, 1663
Receive_Reset, 1619, 1663
Rechnerkopplung RK512, 1703
Befehlstelegramm, 1703, 1704
Daten holen, 1709
Daten senden, 1706
Reaktionstelegramm, 1703
Rechteabhängigkeiten, 386
Recipe, 1468
Record Protocol, 328
RECV_440, 1673
redundante Netzwerke, 704, 2823
Redundanter Sync-Master, 3056
Redundanz
Redundanz-Domäne, 3100
Redundanzgruppe, 1605
Redundanz-Port, 1606
Redundanz-Verfahren
HRP, 2091
Redundierbare Stromversorgungsbaugruppen, 1141
Referenzkanal, 3405
Referenzkanal des Moduls, 3449
Referenzkanalfehler, 3447
Referenztemperatur, 3405, 3408, 3448, 3449
Regeln für die Netzkonfiguration von MPI-Netzen
Regeln für die MPI-Adressvergabe, 84
Regeln für die Netzkonfiguration von PROFIBUS-Netzen
Teilnehmeradressen vergeben, 84, 85
Regeln zur Adressvergabe, 1160
Register "IP-Konfiguration", 3214
Register "IP-Zugriffsschutz", 3227
Register "OPC-Eigenschaften", 287
Register "Port Parameter", 3224
Register der CPU, 862
Remanenter Speicher, 772
Remanenz, 968, 1376
Ladespeicher, Arbeitsspeicher, Systemspeicher, 772
Remanenz von IP-Adressparametern, 3001
Remanenzspeicher, 964
Remanenzverhalten, 794
Remanenzverhalten nach Netzausfall, 759
Rendezvous Punkt, 2116
Report System Errors, 1325, (Siehe Systemdiagnose)
angelegte Fehler-OBs, 3546
RES_RCVB, 1672
RES_RECV, 1673

Reservemodul
Ersatzwert, 2896, 2975
Reservierte Ereignisklassen, 944
RFC 5280, 324
RI, 1625
Ringport, 3107
Ringredundanz, 2299
HRP, 2266
MRP, 2266
RIP
RIPv2-Statistik, 2165
RIPng
Konfiguration, 2406
RIPng Statistics, 2177
Schnittstellen, 2407
RIPv2
Konfiguration, 2376
Schnittstellen, 2377
RK 512-Protokoll (CPU 31xC), 857
RMON
History, 2338
Statistik, 2337
Rollen, 726, 2413, 2648, 2844
Root Max Age, 629, 2742
Root-Bridge, 2479
Route anlegen, 534
Router, 92, 3004
Router-IP-Adresse, 534
Routing, 2107
ICMP, 591, 2703
IPv4 Routing-Tabelle, 625, 2732
IPv6 Routing-Tabelle, 2169, 2503
Routing-Tabelle, 2158
statische Routen, 2107
VRRP, 2107
Routing-Modus, 433
RS232/PPI-Multi-Master-Kabel, 1411
RS232-Begleitsignale
Automatische Bedienung, 1636
RS232-Betrieb, 1623
RS232C-Begleitsignale, 1690
Automatische Bedienung, 1691
RS232-Signale, 1623
RS422-Betrieb, 1627
RS422-Signale, 1627, 1630
RS485-Betrieb, 1630
RSTP, 703, 2822
RT-Klasse, 2984
RTS, 1625
Rücklesbarkeit von Verbindungsparametern, 131
Rücklesedatensatz, 3464, (siehe Rückmeldedatensatz)
Rücklesedatensatz 197, 3384, 3442, 3464
Rückmeldedatensatz, 3389, 3431
Rücksetzen, 2192, 2540, 2756
Rücksetzen auf Werkseinstellungen, 968
RUN, 763, 765, 957, 1369
Rundachse (CPU 31xC), 835, 843

S

S_MODB, 1673
S_RCV, 1673
S_RTS, 1673
S_SEND, 1673, 1742
S_USSI, 1674
S_USSR, 1674
S_USST, 1673
S_V24, 1673
S_VSET, 1673
S_VSTAT, 1673
S_XON, 1673
S7-1500
Definition, 1142
Eigenschaften, 1142
Einsatzgebiet, 1142
Powersegment, 1143
S7-300
Steckplatzregeln, 1140
S7-400
Steckplatzregeln, 1141
S7-400-CPUs
Speichertypen, 773
S7-CP, 367
S7-CPU
Arbeitsspeicher, 1374
Ladespeicher, 1373
Operandenbereich, 775, 1374
Speicherbereiche, 770
S7-Kommunikation, 186
S7-modular Embedded Controller, 1561
Projektieren, 1562
S7-PCT, 3190
S7-Routing
Verbindungsressourcen, 186
S7-Verbindung, 147
TSAP, 158, 251
SALRM, 2905
Sammelmeldung, 1035, 1423
SCALANCE S, 367
Schneller Zähler
Allgemeines, 1401
Funktionsweise, 1401
Konfigurieren, 1404

Schnittstelle
anzeigen, 3489, 3559
umbenennen, 319
X27 (RS 485), 1631
X27 (RS422), 1628
Schnittstellen, 1615
Schnittstellen zuweisen, 1557, 1559
Bezugslinien, 1557
Schnittstelleneigenschaften, 3489, 3559
Schreibschutz, 1418
Schutzkonzept, 1027, 1034, 1037, 1420, 1422
Schutzstufe, 809, 1026, 1031, 1418, 1419
SD-Karte, (siehe Memory Card)
Secure Communication, 324
Secure Socket Layer, 328
Security Configuration Tool
Projektierungssichten, 368
Security-Einstellungen, 1038
Rollen und Rechte, 345
Security-Ereignis, 1035, 1423
Security-Event, 1035, 1423
Security-Modul, 367
Segmentdiagnose
grafische Anzeige, 3534
Symbole, 3534
textuelle Anzeige, 3534
Seitenlayout
in Meldetexten, 1715
Sekundärer Sync-Master, (siehe Redundanter Sync-Master)
Selbstsignierte Zertifikate, 329, 1042
select before operate, 1906
SEND/PUT-Telegramm, 1703, 1704, 1707
SEND_440, 1673
Send_Config, 1619, 1663
Send_P2P, 1619, 1663
SEND_PTP, 1672
SEND_RK, 1672
Senderbaugruppe, 1156
Sendetakt, 1054, 3017, 3043, 3053
Sendetakt-Adaption, 3043
SEND-Telegramm, 1703, 1707
Serielle Datenübertragung, 1620
Serienmaschinen-Projekt, 1190, 3137, 3138, 3141, 3144, 3147, 3151, 3152, 3159, 3160
Seriennummer, 618, 2131, 2488, 2724
SERVE_RK, 1672
Serverzertifikat, 605, 2713
Server-Zertifikat, 1242
Servicedaten
speichern, 3505
Servicedaten auslesen, 1322
Servo-OB, 1082, 1085, 1086, 1448, 1449
Set_Features, 1619, 1663
SET-Taster, 664, 2778
Severity, 433
SFP-Diagnose, 679, 2263
SFTP
Laden/Speichern, 644, 2199, 2550, 2759
SHA1, 459
SHA-Algorithmus, 654, 2222, 2566, 2768
Shared Device, 974, 3116, 3133
zugehörige IO-Controller, 3491
Shared I-Device, 3124
Shared-Device, 2985
SHDSL
Schnittstelle, 2814
Übersicht, 2735, 2813
Sicherheit (Webserver), 1321
Sicherheitseinstellungen, 654, 2222, 2566, 2768
Sicherheits-Ereignisse, 496
SiClock, 415
Siemens Industry Online Support öffnen, 35
Signal Recorder, 2735
Signal_Get, 1619, 1663
Signal_Set, 1619, 1663
Signalboard, 1462
stecken, 1463
Signalrekorder, 2616
Signatur, 330
SIM
Schnittstellen, 2807
SIMATIC ET 200AL, 3397, 3424, 3428
SIMATIC Memory Card, 964
formatieren, 3513
SIMATIC NET PC Software für Kommunikationsprozessor, 1564
SIMATIC NET Version
Onboard-Schnittstelle, 1560
PC-Station, 1561
SIMATIC-ACC, 158, 251
Skalierbarer Messbereich, 3418
Slave, 1725
SMS
empfangen, 2791
senden, 2789
SMS-Text (TCON_Phone), 125
SMTP
Client, 638, 2751
SMTP-Client, 2207, 2557
SNAT
konfigurieren, 716, 2834

SNMP, 383, 596, 639, 654, 2086, 2185, 2222, 2477, 2531, 2566, 2699, 2752, 2768, 3243
Benutzer, 655, 2223, 2567, 2769
Gruppen, 654, 2222, 2566, 2768
SNMP-Trap, 653, 2221, 2565, 2767
SNMPv1, 596, 2086, 2477, 2699
SNMPv2c, 596, 2086, 2477, 2699
SNMPv3, 596, 2086, 2477, 2699
Übersicht, 624, 2179, 2504, 2731
SNMPv1, 459
SNMPv3, 459
SOFTNET Security Client
Datenbasis, 500
Einsatz des, 498
im Projekt konfigurieren, 499
Konfigurationsdatei erstellen, 499
SOFTNET Security Client, 367
Softwareänderung, 969
Softwareausgabestand, 618, 2131, 2488, 2724
Software-Datenflusskontrolle, 1633
Softwaregesteuerte Datenflusskontrolle, 1410
Softwarehandshake, 1717
Sonderzeichen
in Variablen für Webserver, 1337, 1475
Source specific-Multicast, 2117
Source-NAT
Masquerading, 603, 2709
Spanning Tree, 703, 2822
Information, 630, 2136, 2494, 2743
Rapid Spanning Tree, 2090, 2480
Spannungsversorgung
Überwachung, 672
Sparse-Mode, 2116
Speedy Splitter, 12, 15, 22, 38
Speicher, 964, 968
Kommandobereich, 3569
komprimieren, 3569
von RAM nach ROM kopieren, 3569
Speicherarten einer CPU anzeigen, 3496, 3497
Speicherauslastung, 965
Speicherbereich
Arbeitsspeicher, 1374
Aufteilung, 770
der Ein und Ausgänge, 863
Ladespeicher, 1373
PE und PA, 864
Remanente Speicherbereiche, 1376
Speicherbereiche
Arbeitsspeicher, 771
Berechnungsgrundlage, 774
Ladespeicher, 771
Systemspeicher, 771
Speichergröße, 775
Speicherkarte, (Siehe Memory Card)
Speicherkonzept S7-400, 773
Speichern
Speicherort für remanente Daten, 1554
Speicherreserve, 970
Speichertest, 1550
Spektrumanalysator, 2621
Spezialdiagnose, 3567
Spezialdiagnose - chinesische GUI, 3568
Spezialvariable (Webserver), 1342, 1478
Spezifische Bausteine, 1112
Sprachdateien
für Diagnoseanzeige, 3242
Sprache (Webserver und CPU-Display), 1003
Sprache (Webserver), 1326
Sprachkürzel, 1494
Sprachumschaltung, 1359, 1360, 1495
SSH
Server, 638, 2751
SSH-Server, 2183
SSL, 328
Stammzertifikat, 331
Stammzertifizierungsstellen, 379
Standard-Anwenderereignisse, 942
Standard-Applikation, 1513
Standard-Modus, 606, 2714
Standard-OB-Ereignisse, 930
Standard-Router, 534
Standby-Redundanz, 2093
Startinformation, 1056
optimiert, 1056, 1057, 1058, 1061, 1062, 1063, 1064, 1066, 1067, 1069, 1071, 1072, 1074, 1075, 1076, 1077, 1080
wie bei S7-300 und S7-400, 1056, 1057, 1058, 1061, 1062, 1063, 1064, 1066, 1067, 1069, 1071, 1072, 1074, 1075, 1076, 1077, 1080
Startup, 1432
Startup (S7-1500), 1056
Startwert, 964
Startzeichen, 1642
Stateful Inspection Firewall, 601, 2707
Stateful Packet Inspection, 392
Station
kopieren, 57
löschen, 63
Mehrfachauswahl, 54
umbenennen, 319
Umbennen, 61
verschieben, 59
Station_Info, 1166
Stationsausfall, (Baugruppenträgerfehler)

Stationsname, (Siehe Gerätename)
Statische Routen
IPv6-Routen, 2387
Statistikdaten, 3564
Status, 1437
Baugruppe gestört, 3499
Status der Online-Verbindung, 3529
Statusalarm, 1437
Statusalarm (S7-1500), 1062
Statusalarm-OB, 1437
Statusalarm-OB (OB 55) (S7-300, S7-400), 880
Statusalarm-OB (S7-1500), 1062
Statussymbole, 1584
Steckleitungen, 1624, 1628, 1631
Steckplatz
Baugruppenträger, 48
Nutzdaten mit DPV1, 2933
selektieren, 49
Steckplatzmodell, 2933
Steckplatzzuordnung bei DPV1, 2933
Steckplatzmodell bei DPV1, 2933
Steckplatzregeln, 49
DP-Slave, 2872, 2884, 2964
DPV1, 2933
ET 200M, 2872
ET 200S, 2889, 2970
OPC-Server, 1565
PC-Station, 1565
S7-1500, 1143
S7-300, 1140
S7-400, 1141
Steuerdatensatz, 1177, 1499, 3389, 3431, 3464
S7-1500, 1188
Steuerdatensatz 196, 3384, 3389, 3442, 3464
Steuerkommando SYNC/FREEZE, 2888
Steuerzeichen, 1695
in Meldetexten, 1715
STOP, 755, 763, 957, 958, 1370
Stopereignisse, 934
Störfrequenzunterdrückung, 2890, 3447
Stromversorgungsbaugruppen, redundierbare, 1141
Stromversorgungsfehler-OB (OB 81) (S7-300, S7-400), 891
Strukturen
Webserver, 1348, 1349
STX, 1649
Submodul, 3133
Submodule stecken
PC-Station, 1566
Submodul-Konfiguration, 974
Subnetz, 94
Konfiguration, 712, 2830
Übersicht, 710, 2829
Subnetze
Default-Gateway, 2345
Konfiguration (IPv6), 2385
Verbundene Subnetze (IPv6), 2385
Subnetzmaske, 92, 589, 2075, 2456, 2695, 2990
Subtakt, 3413
Suche
Hardware-Katalog, 32
Support zu einem Modul anzeigen, 35
Switchport, 3025
Symbol
für Betriebszustand, 3484
für Hardware-Diagnose, 3482
für Software-Diagnose, 3483
für Verbindungsstatus, 3537
für Vergleich, 3483
für Vergleichsstatus, 3537
Overlay-Icon, 3484
Symbol zuordnen, 1175, 1461
Symmetrische Verschlüsselung, 326
SYNC, 2888
SYNC_PI, 800
SYNC_PO, 800
Sync-Domain, 3026, 3053, 3492
Sync-Domain-Eigenschaften, 3492
Synchrone Fehler
OB 121 (S7-300, S7-400), 924
OB 122 (S7-300, S7-400), 927
Synchronfehlereignisse, 930
Synchronisation der Uhrzeit, 803, 1024
Synchronisierte Bearbeitungszyklen, 2939
Synchronisierung (Anwenderdefinierte Webseiten), 1332, 1471
Synchronous cycle (S7-1500), 1070
Sync-Master, 3056
Syntax für AWP-Kommandos, 1337, 1474
Syntax für Konstanten, 865
Syslog
Client, 638, 2751
Syslog-Client, 666, 2238, 2578, 2779
Syslog-Server, 488
System
Allgemeine Informationen, 640, 2186, 2533, 2753
Configuration, 638, 2530, 2751
Systemkonfiguration, 2183
Systembaustein
anzeigen, 3560
Systemdiagnose
angelegte Bausteine, 3545

Diagnosestatus-DB, 3547
Einführung, 3542, 3553
Systemereignisse
Konfiguration, 646, 2553, 2762
Severity Filter, 650, 2764
System-Ereignisse, 495
Systemfehler melden, (Siehe Report System Errors), (Siehe Systemdiagnose)
Systemfunktionen
Parameter im Anwenderprogramm, 1155
Systeminformationen
Adressen, 2935
System-Log, 495
Systemmerker, 988, 1025, 1390, 1416
Systemspeicher, 771
Baustein-Stack, 781
Diagnosepuffer, 782, 1378, 3527
Lokaldaten-Stack, 779
Operandenbereiche, 775, 1374
Prozessabbild der Ein- und Ausgänge, 778, 1376
Unterbrechungs-Stack, 781
Zugriff über FETCH/WRITE, 282
Systemstromversorgung, 3441
Systemtakt, 3059
Systemtakt verkürzen, 2952
Systemzeit
NTP-Client, 660, 2231, 2574, 2774
PTP-Client, 2234
SNTP-Client, 658, 2229, 2572, 2772

T

T_DC, 1054
Tablet, 1362
Taktmerker, 804, 988, 1026, 1390, 1417
Taktsynchronalarm, 800
Taktsynchronalarm-OB, 1054, 2942
OB 61 bis OB 64 (S7-300, S7-400), 887
parametrieren, 1092
Taktsynchronalarm-OBs (S7-1500), 1070
Taktsynchroner Betrieb, 2985
Taktsynchronität, 2939, 3064, 3191, 3197
Ablauf, 3059
am Subnetz einstellen, 3067
Anwenderprogramm erstellen, 2951
Beispiel: Erfassung mehrerer Messstellen, 2943
CPU-Eigenschaften, 2947
DP-Zykluszeit und Ti/To-Werte, 2949
Eigenschaften am PROFIBUS, 2949
Einstellungen am DP-Slave, 2951
gleiche Ti/To-Zeiten für alle Slaves, 2951
Kombination von DP mit und ohne Taktsynchronität, 2945
konfigurieren, 2946
nicht wählbar, 2949
Oversampling, 3413
projektieren, 3065, 3066
Regeln, Voraussetzungen und Randbedingungen, 2944, 3068
Reserven, 2955
Teilprozessabbilder, 2947
Überlappung von Ti und To, 2954
Verzögerungszeit, 2948
Warum?, 3058
Wirkung von Ti, 3061
Wirkung von To, 3062
zeitliche Synchronisation, 1054
Task Card
Hardware-Katalog, 30
Online-Tools, 3487
Tastaturbedienung, 38
Taster, 664, 2778
Tauschen
Baugruppe, 60
TCI (Tool Calling Interface), 3188
TCON_FDL, 126
TCON_IP_RFC, 124
TCON_IP_v4, 118
TCON_IP_V4_SEC, 119
TCON_Param, 115
TCON_PHONE für Telecontrol-Verbindungen, 125
TCON_QDN, 121
TCON_QDN_SEC, 122
TCP, 410
Merkmale, 113
Port-Nummern, 130
TDP (Systemtakt), 2940
Teilnehmeradresse, 1160, 1161, 1458
Teilprozessabbild, 1171, 1173, 1174, 2947
Telegrammaufbau, 1654, 1730
Telegrammgröße IO-Device, 3225
Telegrammkopf
Aufbau beim RK512-Befehlstelegramm, 1704
Telnet
Server, 638, 2751
Temperaturkoeffizient, 3378, 3447
Temperaturkompensation, 3365, 3408
Temporäre Variablen (TEMP)
erforderlich für OBs, 927
im zyklischen Programm (S7-300, S7-400), 870
Terminal- und Elektronikmodule, 3363
Testbetrieb, 811

TFTP
Laden/Speichern, 642, 2197, 2548, 2757
Thermoelement, 3408, 3418
Thermowiderstand, 3408, 3418
Ti, 1054
Ti (Zeitpunkt Eingangsdaten einlesen), 2940, 2952, 3067
Time delay interrupt, 1439
Time delay interrupt (S7-1500), 1064
Time error interrupt, 1443
Time error interrupt (S7-1500), 1072
Time of day (S7-1500), 1059
TLS, 328
TM, 3046
To, 1054
To (Zeitpunkt Augangsdaten ausgeben), 2940
To (Zeitpunkt Ausgangsdaten ausgeben), 2952, 3067
Tool Calling Interface (TCI), 3188
Topologie
Ring, 3100
Topologieanzeige, 3242
Topologieeditor, 94
Topologieerkennung, 3026
Topologien, 1725
Topologiesicht
Diagnosestatus in der grafischen Ansicht, 310
Diagnosestatus in der tabellarischen Ansicht, 310
Erweiterter Offline-/Online-Vergleich, 314
Funktionen, 304
Gerät hinzufügen, 43
Hardware- und Netzwerkeditor, 22
Offline-/Online-Vergleich, 311
Online ermittelte Geräte übernehmen, 322
Online ermittelte Port-Verschaltungen übernehmen, 321
Ports verschalten, 314
projektierte Topologie, 307, 308
Unterschiede zur Netzsicht, 305
Topologiesicht starten, 307
Topologieübersicht, 22
TPA, (Siehe Teilprozessabbild)
Transfer Card; siehe Übertragungskarte, 1371
Transferbereich, 2911, 2912, 2933, 3116
Transfermodul-Mapping, 3046
Transport Layer Security, 328
Trust Mode, 2271
TSAP
ASCII-Codetabelle, 134
Aufbau, 132, 158, 251
TSAP-Vergabe
Beispiele, 135
Typen von DP-Slaves, 2883, 2964

U

UA-Redundanzserver, 1603
UART-Datenübertragung, 1406
Überlappung von Ti und To
Einstellung, 2954
Funktionsprinzip, 2953
Regeln, 2955
Voraussetzung, 2952
Überlastverhalten, 982
Überlauf, 3380, 3405
Überschreiben des PROFINET-Gerätenamens, 3147
Übersicht
Access Point, 2509
Clients, 2515
Overlap APs, 2514
Überlappende Kanäle, 2514
Übersichtsnavigation, 13, 16, 23
Übertragungskarte, 1371, (Siehe Memory Card)
Übertragungsmedium/Duplex, 3021
Übertragungssicherheit, 1620, 1679
bei 3964(R), 1622
bei Freeport, 1621
bei Modbus und USS, 1622
bei RK512, 1681
beim ASCII-Treiber, 1680
Überwachen, 3021
Überwachung des Partnerports..., 3016
Überwachungsfenster, 3456
UDP, 410, 417, 452
Merkmale, 114
Port-Nummern, 130
UDP-Verbindung / FDL-Verbindung
unspezifiziert, 244
Uhrzeit, 988, 1390
Precisition Time Protocol, 2234
SNTP (Simple Network Time Protocol), 658, 2229, 2572, 2772
Synchronisation, 3561
Systemzeit, 657, 2225, 2569, 2771
Uhrzeit einstellen in Online- und Diagnosesicht, 3508
Uhrzeitsynchronisation, 658, 2229, 2572, 2772
Uhrzeit ermitteln, 3508
Uhrzeit synchronisieren, 803, 1024
Uhrzeitalarm, 797
Funktion, 1435
OB 10 bis OB 17 (S7-300, S7-400), 871
Regeln, 1435

Status abfragen, 1435
stellen und aktivieren, 1435
stornieren, 1435
Uhrzeitalarm (S7-1500), 1059
Uhrzeitalarm-OB
parametrieren, 1089, 1451
Uhrzeitalarm-OB (S7-1500), 1059
Uhrzeitfunktion, 804, 1024, 1025
Betriebsstundenzähler, 804
Grundlagen, 802, 1023, 1399
Lesen der Uhrzeit, 802, 1023, 1400
Parameter der Uhr, 803, 1400
Stellen der Uhrzeit, 802, 1023, 1400
Uhrzeitformat, 802, 1023, 1399
Uhrzeitsynchronisation, 1390, 3215, 3264
PROFIBUS-CP, 3260
Uhrzeitsynchronisierung, 988
Umbennen, 61
Umkonfigurieren über Anwenderprogramm, 1177, 1499, 3384, 3389, 3431, 3442, 3464
Umschaltzeit, 1413
Umstellung von DPV0 nach DPV1, 2930
Ungerader Sendetakt, 3043
Uni/-Bidirektioneller Datenverkehr, 1624, 1675, 1691
Unspezifizierte Verbindungen
bei Industrial Ethernet, 217
bei PROFIBUS, 217
Unspezifizierte Verbindung
dialoggeführt projektieren, 243
grafisch projektieren, 242
Unspezifizierten Verbindung
Merkmale, 245
Unterbrechungs-Stack, 781
Unterlauf, 3380, 3405
Unterstützte Hardwarekompatibilität, 988
Update, 1438
Update-Alarm, 1438
Update-Alarm (S7-1500), 1063
Update-Alarm-OB, 1438
Update-Alarm-OB (OB 56) (S7-300, S7-400), 882
Update-Alarm-OB (S7-1500), 1063
Upload, 3192
Urlöschen, 766, 958, 968, 1370
durchführen, 3507
USB-Netzwerkkarte ziehen/stecken, 1567
USS_Drive_Control, 1619, 1667
USS_Port_Scan, 1619, 1667
USS_Read_Param, 1619, 1667
USS_Write_Param, 1619, 1667
USS-Anweisungen, 1667
USS-Kommunikation, 1659
USS-Master
Funktionsübersicht, 1661, 1793
Reihenfolge Funktionsaufrufe, 1794
USS-Protokoll, 1659, 1792
USS-Protokoll: Datenfeld, 1660
USS-Protokoll: Netzdatenbaustein, 1793
USS-Protokoll:Datenübertragungsprozedur, 1660, 1792
USS-Protokoll:Datenverschlüsselung, 1660, 1792
USS-Protokoll:Telegrammaufbau, 1660, 1792
USS-Protokoll
Allgemeiner Aufbau des Datenbausteins:Parameterbereich (PKW), 1661
Allgemeiner Aufbau des Datenbausteins:Prozessdatenbereich (PZD), 1661
Allgemeiner Aufbau des Netzdatenbausteins:Parameterbereich (PKW), 1793
Allgemeiner Aufbau des Netzdatenbausteins:Prozessdatenbereich (PZD), 1793

V

V24_SET, 1672
V24_SET_340, 1671
V24_SET_441, 1673
V24_STAT, 1672
V24_STAT_340, 1671
V24_STAT_441, 1673
Varablenname
Webserver (PLC), 1337, 1475
Variable lesen, 1338, 1476
Variable schreiben, 1477
Variable zuordnen, 1175, 1461
Variablen, 3234
Variablentabelle
Webserver, 1323
Verbindung, 137, 147, 222
Adressdetails, 3540
anlegen, 107
bei fehlender oder nicht eindeutiger Netzzuordnung projektieren, 152, 229
bei fehlender Vernetzung projektieren, 233
belegte und reservierte anzeigen, 3562
Einfluss der Vernetzung - Verhalten bei fehlender Vernetzung, 223
E-Mail, 201, 205, 207, 209
FDL, 201, 205, 207, 209
ISO-on-TCP, 201, 205, 207, 209

ISO-Transport, 201, 205, 207, 209
löschen, 111
ohne Port-Spezifizierung, 218
programmierte, 219, 220
projektieren, 201
TCP, 201, 205, 207, 209
UDP, 201, 205, 207, 209
unspezifiziert, 222
zu einer bestimmten Station in einem beliebigen Projekt, 217
Verbindungen
Anzahl der Verbindungen (PROFINET/ PROFIBUS), 178
grafisch projektieren, 226
Verbindungen dialoggeführt ändern, 239
Verbindungen projektieren
dialoggeführt, 230
Verbindungsanzeige in der Netzsicht, 225
Verbindungsbeschreibung
Aufbau, 115, 118, 119, 121, 122, 124
Datenbaustein, 115, 117, 119, 120, 122, 123
Parameterwerte ändern, 131
Verbindungsdetails, 3540
Verbindungsdiagnose
detailliert, 3538
Überblick, 3536
Verbindungsinformation, 3538
Verbindungsmechanismen, 1034, 1422
Verbindungsmodus, 227
Verbindungsparameter PUT/GET-Anweisung, 163
Verbindungsparametrierung
Allgemein, 99
starten, 106
Übersicht, 101
Verbindungsparameter, 103
Verbindungsparametrierung PUT/GET-Anweisung, 164
Verbindungspartner ändern, 238
Verbindungsprojektierung, 201
dialoggeführt, 221
grafisch, 221
Verbindungsprojektierung mit OPC-Server, 216
Verbindungsregeln, 566, 580
Verbindungsressource, 137, 148
Verbindungsressourcen, 172, 3562
Anweige im Webserver, 192
Anzeige in STEP 7, 189
belegen, 187
Datensatz-Routing, 186
HMI-Kommunikation, 186
modulspezifisch, 179, 190
online, 3538, 3539
S7-Routing, 186
stationsspezifisch, 179, 189
Überblick, 182
Verbindungsstatus
mittels Symbolen anzeigen, 3537
Verbindungstabelle, 225, 237
Verbindungsunterbrechung, 1905
Verbindungsweg
in der Netzsicht hervorheben, 235
Verbindungsweg automatisch ermitteln, 227
Verbindungsweg gezielt wählen, 229
Verbindungsweg im Inspektorfenster ändern, 241
Verbindungsweg zuweisen, 221
Verfügbare Systemfunktionen, 586, 2064, 2436, 2442, 2447, 2689
Vergleichseditor
Gerätevergleich, 72
Vergleichsstelle, 2891, 2971, 3365, 3380, 3405, 3408, 3448, 3449
Verkablelungregeln (PROFINET), 3025
Vernetzen in der Gerätesicht
Möglichkeiten zur Vernetzung, 80
Vorgehen, 80
Vernetzung bearbeiten
Subnetz kopieren, 83
Subnetze und Geräte kopieren, 84
Vernetzungen bearbeiten
Netzanschluss aufheben, 82
Vernetzungsmodus, 77
Verschalten, 94
Verschalten von Ports, 3016
Verschieben
Hardware-Komponenten, 59
Verschlüsselung, 369
Versionsübersicht, 1456
Versorgungsspannung, 3405
Verzögerungsalarm
Einstellungen, 796
OB 20 bis OB 23 (S7-300, S7-400), 874
Verzögerungsalarm (S7-1500), 1064
Verzögerungsalarm-OB, 1439
Verzögerungsalarm-OBs (S7-1500), 1064
Verzögerungszeit
Taktsynchronität, 2948
Visualisierung, 1562
VLAN, 592, 2080, 2462, 2696
Port VID, 702, 2281, 2629, 2821
Priorität, 701, 2281, 2629, 2821
Tag, 701, 2281, 2629, 2821
VLAN-ID, 595, 2083, 2462, 2699
VLAN-Tag, 594, 2081, 2460, 2697
VLAN-Betrieb, 465

VLAN-Tagging, 465
Vollduplexbetrieb, 1620, 1676
Vorbelegung, 1726
Vorbelegung der Empfangsleitung, 1627, 1724
Vordefinierte Firewall-Regeln
    CP 1543-1, 571, 575
    CP 1628, 561
    CP 343-1 Adv. / 443-1 Adv., 559
    SCALANCE S, 524, 526
Vorlaufzeit, 1054
Vorlaufzeit Ti, 2942
VPN-Gruppe, 461
VPN-Verbindung
    Status, 626, 2739
    Status OpenVPN-Client, 628, 2741
VRRP
    Adressübersicht, 2360
    Backup-Router, 2108
    Konfiguration, 2359
    Master-Router, 2108
    Router, 2357
    Schnittstellenüberwachung, 2361, 2366, 2399
    Virtueller Router, 2108
    VRRP Statistics, 2138
    VRRP-Router, 2108
    VRRPv3 Adressenkonfiguration (IPv4), 723, 2366, 2841
    VRRPv3 Adressenkonfiguration (IPv6), 2398
    VRRPv3 Adressenübersicht (IPv4), 723, 2365, 2841
    VRRPv3 Adressenübersicht (IPv6), 2398
    VRRPv3 Konfiguration (IPv4), 721, 2364, 2839
    VRRPv3 Konfiguration (IPv6), 2396
    VRRPv3 Router (IPv4), 719, 2362, 2394, 2837
VRRPv3
    Schnittstellenüberwachung, 724, 2842
    VRRPv3-Statistik, 632, 2140, 2745

W

WAN-IP-Adresse
    festlegen, 484
WDS, 2607
Web Server, (siehe Webserver)
Webapplikation, 1330, 1469
Web-Control-DB, 1353, 1488
Web-Datenbausteine, 1351, 1485, 1486
Webdiagnose, 3242
Webseiten
    Aktualisierungsintervall, 3242
Webseiten im Browser, 1359, 1493
Webserver, 805, 1003, 1317, 1326, 1330, 1334, 1351, 1359, 1362, 1463, 1465, 1469, 1473, 1485, 1486
    aktivieren, 1324, 1325, 1466
    Anwenderdefinierte Webseiten, 1330, 1469
    Anzeigeklassen für Meldungen, 1330
    Array, 1346, 1349
    Benutzerverwaltung, 1326
    Enumerationen, 1345, 1482
    Fragment, 1349, 1483
    HTTPS, 1328, 1467
    Regeln PLC-Variablennamen, 1337, 1474
    Strukturen, 1348, 1349
    Zertifikat, 1328
    Zugriff verwalten, 1324
Web-Zugriff, 1323, 1324
Wechselmedium, 3026
Wechselnder Partner-Port, 3090
Wechsler, 3455
Weckalarm
    Einstellungen, 799
    OB 30 bis OB 38 (S7-300, S7-400), 876
Weckalarm-OB
    Beschreibung, 1440
    parametrieren, 1090, 1453
Weckalarm-OBs (S7-1500), 1066
Weg einer projektierten Verbindung ändern, 239
Wegerfassung (CPU 31xC), 830, 838
Weitere Feldgeräte (PROFIBUS und PROFINET), 2926, 2983, 3036, 3037, 3078
Werkseinstellungen
    zurücksetzen auf, 3509, 3510, 3512
Werkzeug, (Siehe Docking Unit)
Werkzeugwechsler, (Siehe Docking Station)
Wertebereich für IPv4-Adresse, 588, 2694
Wertebereiche IP-Adresse Subnetzmaske und Adresse des Netzübergangs,
Wertstatus, 973, 974, 3133
Widerstandsthermometer, (siehe Thermowiderstand)
Wiederanlauf, 758, 791
Wiederanlauf (S7-300, S7-400), 919
Wiederanlauf der CPU, 1796
WinCC RT), 1512
Wireless-LAN/PB-Link, 2985
Wirkung von Ti
    Taktsynchronität, 3061
Wirkung von To
    Taktsynchronität, 3062
WLAN
    WLAN Authentifizierungs-Log, 2492
WLAN-Statistik
    Fehler, 2521

WLAN-Statistiken
Empfangene Datentelegramme, 2525
Gesendete Datentelegramme, 2525
WRREC, 1166
WWW (Anweisung), 1352, 1487

X

X.509, 324
X27 (RS 485)-Schnittstelle, 1631
X27 (RS422)-Schnittstelle, 1628
X27 (RS422/485)- Schnittstelle, 1724
XDB-Konfiguration, 1527, 1528, 1585
XON/XOFF, 1633

Z

Zählen (8 DI NAMUR), 3367
Zählen (CPU 31xC), 811
Zähler, schneller, 1401
Zählerart, 1402
Zählereingang, 1402
Zeichensatz
in Meldetexten, 1715
Zeichenverzugszeit, 1643
Zeichenverzugszeit ZVZ, 1655, 1731
Zeiteinstellung, 639, 2184, 2531, 2752
Zeitfehler-OB, 1443
Schwellwert-Mechanismus, 983
Zeitfehler-OB (OB 80) (S7-300, S7-400), 888
Zeitfehler-OB (S7-1500), 1072
Zeitstempelung, 3366
Zeitsynchronisierung, 453
Zentrale Baugruppen
Auswahl aus Hardware-Katalog, 1151
Eigenschaften der CPUs, 1154
Parameter im Anwenderprogramm, 1155
Übersicht Konfiguration, 1138
Zertifikat, 376, 465
austauschen, 381
durch Zertifizierungsstelle signiert, 380
erneuern, 380
ersetzen, 381
exportieren, 376
importieren, 376
selbst-signiert, 380
zuweisen, 381
Zertifikate
anzeigen, 350
erneuern, 349
ersetzen, 350
erstellen, 1042
exportieren (global), 347
exportieren (lokal), 1044
für OPC UA-Server, 1047
für Webserver, 1046
importieren, 346
löschen (global), 351
löschen (lokal), 1045
selbstsigniert, 1042
zuweisen, 1043
Zertifikatsinhaber, 328
Zertifikatsmanager, 376, 1038
Zertifikatsmanager (global), 344
Zertifikatsmanager (lokal), 1040
Zertifizierungsstelle, 377
Zertifizierungsstellen, 328
Ziehen/Stecken-Alarm (ET 200M), 3454
Ziehen/Stecken-OB, 1445
Ziehen/Stecken-OB (S7-1500), 1075
Ziehen/Stecken-OB (S7-300, S7-400), 897
Zoom
Einstellen der Zoom-Stufe, 12, 16, 22
Tastaturbedienung, 39
Zugriff
direkter ~ auf die Peripherie, 864
Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben, 1034, 1422
Zugriffsschutz, 1038
parametrieren, 1551
Zugriffsstufe, 1026, 1031
Zugriffsteuerung, 2325
automatisches Lernen, 2325
Zurücksetzen
auf Werkseinstellungen, 3509, 3510
der PROFINET-Schnittstellenparameter, 3571
Zurücksetzen auf Werkseinstellungen, 968, 3512
Zusammenfassung der Bit-, Byte-, Wort- und Doppelwortkennungen, 865
Zyklische Programmbearbeitung
Programmieren, 1433
Unterbrechungsmöglichkeiten, 1433
Zyklisches Programm (OB 1) (S7-300, S7-400), 870
Zyklisches Programm (S7-1500), 1057
Zyklusbelastung, 1000, 1398
Zyklusbelastung durch Kommunikation, 793
Zyklus-OB
Beschreibung, 1433
Zyklus-OB (S7-1500), 1057
Zyklusüberwachungszeit, 957, 999, 1369, 1397
Zyklusverhalten, 792

Zykluszeit, 988, 999, 1390, 1396
gemessene anzeigen, 3494
parametrierte anzeigen, 3489

SIEMENS

PLC programmieren

Programmier- und Bedienhandbuch

Grundlagen zur Programmierung 1

Programmierempfehlungen 2

Datentypen 3

Anweisungen 4

Programmiereditor 5

Software Units verwenden 6

Bausteine anlegen und verwalten 7

Bausteine schützen 8

Bausteinschnittstelle deklarieren 9

Datenbausteine programmieren 10

PLC-Variablen deklarieren 11

PLC-Datentypen (UDT) deklarieren 12

KOP-Programme erstellen 13

FUP-Programme erstellen 14

AWL-Programme erstellen 15

SCL-Programme erstellen 16

10/2018

Fortsetzung nächste Seite

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

Ⓟ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

PLC programmieren

Programmier- und Bedienhandbuch

Fortsetzung

GRAPH-Programme erstellen 17

Meldungen projektieren 18

PLC-Programme übersetzen und laden 19

PLC-Programme vergleichen 20

Programminformationen anzeigen 21

Querverweise anzeigen 22

Anwenderprogramm testen 23

Maschinen und Anlagen mit ProDiag überwachen 24

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Grundlagen zur Programmierung....35

1.1 Betriebssystem und Anwenderprogramm....35

1.2 Bausteine im Anwenderprogramm....37
1.2.1 Lineare und strukturierte Programmierung....37
1.2.2 Übersicht über die Bausteinarten....39
1.2.3 Organisationsbausteine (OB)....39
1.2.4 Funktionen (FC)....40
1.2.5 Funktionsbausteine (FB)....41
1.2.6 Globale Datenbausteine (DB)....42
1.2.7 Globale ARRAY-Datenbausteine (DB)....43
1.2.8 Beispiel zur Verwendung von ARRAY-Datenbausteinen....44
1.2.9 Instanz-Datenbausteine....53
1.2.10 CPU-Datenbausteine....54
1.2.11 Bausteine mit optimiertem Zugriff....55
1.2.11.1 Grundlagen zum Bausteinzugriff....55
1.2.11.2 Bausteinzugriff einstellen....58

1.3 Bausteinaufrufe....62
1.3.1 Grundlagen zu Bausteinaufrufen....62
1.3.2 Aufrufhierarchie....63
1.3.3 Instanzen....64
1.3.3.1 Grundlagen zu Instanzen....64
1.3.3.2 Einzelinstanzen....65
1.3.3.3 Multiinstanzen....67
1.3.3.4 Parameterinstanzen....70
1.3.3.5 Beispiel zur Verwendung von Parameterinstanzen....73
1.3.4 Parameterübergabe beim Bausteinaufruf....78
1.3.4.1 Regeln zur Versorgung von Bausteinparametern....78
1.3.4.2 Parameterversorgung von Funktionen....80
1.3.4.3 Parameterversorgung von Funktionsbausteinen....83
1.3.4.4 Parameter als Kopie oder als Pointer übergeben....84
1.3.4.5 Weiterreichen von Bausteinparametern....87

1.4 Operanden verwenden und adressieren....93
1.4.1 Grundlagen zu Operanden....93
1.4.2 Schlüsselwörter....93
1.4.3 Variablen....100
1.4.4 Konstanten....102
1.4.4.1 Grundlagen zu Konstanten....102
1.4.4.2 Deklaration symbolischer Namen für Konstanten....103
1.4.4.3 Datentypen von Konstanten....105
1.4.4.4 Beispiele zur Verwendung von Konstanten....106
1.4.5 Operanden adressieren....108
1.4.5.1 Globale Variablen adressieren....108
1.4.5.2 Variablen in Datenbausteinen adressieren....111
1.4.5.3 Operanden indirekt adressieren....119

1.5 Programmablauffehler behandeln....143
1.5.1 Fehlerursachen....143
1.5.2 Mechanismen zur Fehlerbehandlung im Überblick....146
1.5.3 EN-/ENO-Mechanismus....150
1.5.3.1 Grundlagen zum EN-/ENO-Mechanismus....150
1.5.3.2 EN-/ENO-Mechanismus in KOP....151
1.5.3.3 EN-/ENO-Mechanismus in FUP....156
1.5.3.4 EN-/ENO-Mechanismus in AWL....161
1.5.3.5 EN-/ENO-Mechanismus in SCL....163
1.5.3.6 EN-/ENO-Mechanismus in GRAPH....168
1.5.3.7 EN-/ENO-Mechanismus in Bausteinen mit unterschiedlichen Netzwerksprachen....170
1.5.4 Fehlerauswertung mit dem Ausgangsparameter RET_VAL....171
1.5.5 Verwendung der Anweisungen GET_ERROR und GET_ERR_ID....176
1.5.6 Beispiel für die Behandlung von Programmablauffehlern....177

1.6 Programmflusssteuerung....183
1.6.1 Statuswort bei S7-300, S7-400....183
1.6.1.1 Grundlagen zum Statuswort....183
1.6.1.2 Setzen der Statusbits....185
1.6.1.3 Abfragen der Statusbits in AWL....193
1.6.2 Statuswort bei S7-1500....196
1.6.2.1 Grundlagen zum Statuswort....196
1.6.2.2 Abfragen und Setzen der Statusbits in AWL....197
1.6.3 Master Control Relay....201
1.6.3.1 Das Master Control Relay....201
1.6.3.2 Wichtige Hinweise zur Benutzung der MCR-Funktionalität....202

2 Programmierempfehlungen....205

2.1 Die neuen Funktionen der S7-1200/1500-CPU und Programmierempfehlungen im Überblick....205

2.2 Symbolisch adressieren....209

2.3 ARRAY-Komponenten indirekt adressieren....211

2.4 Datentyp DB_ANY verwenden....215

2.5 PLC-Datentypen (UDT) verwenden....231

2.6 MOVE-Anweisungen in AWL verwenden....235

2.7 IEC-Zeiten und -Zähler verwenden....238

3 Datentypen....243

3.1 Übersicht über die gültigen Datentypen....243

3.2 Binärzahlen....248
3.2.1 BOOL (Bit)....248
3.2.2 Bitfolgen....249
3.2.2.1 BYTE....249
3.2.2.2 WORD....250
3.2.2.3 DWORD....251
3.2.2.4 LWORD....252

3.3 Ganzzahlen....254
3.3.1 SINT (8-Bit-Ganzzahlen)....254

3.3.2 USINT (8-Bit-Ganzzahlen)....255
3.3.3 INT (16-Bit-Ganzzahlen)....256
3.3.4 UINT (16-Bit-Ganzzahlen)....257
3.3.5 DINT (32-Bit-Ganzzahlen)....258
3.3.6 UDINT (32-Bit-Ganzzahlen)....259
3.3.7 LINT (64-Bit-Ganzzahlen)....260
3.3.8 ULINT (64-Bit-Ganzzahlen)....261

3.4 Gleitpunktzahlen....264
3.4.1 REAL....264
3.4.2 LREAL....265
3.4.3 Ungültige Gleitpunktzahlen....266

3.5 Zeiten....268
3.5.1 S5TIME (Zeitdauer)....268
3.5.2 TIME (IEC-Zeit)....269
3.5.3 LTIME (IEC-Zeit)....270

3.6 Datum und Uhrzeit....271
3.6.1 DATE....271
3.6.2 TOD (TIME_OF_DAY)....271
3.6.3 LTOD (LTIME_OF_DAY)....272
3.6.4 DT (DATE_AND_TIME)....272
3.6.5 LDT (DATE_AND_LTIME)....274
3.6.6 DTL....274

3.7 Zeichenfolgen....276
3.7.1 Zeichen....276
3.7.1.1 CHAR....276
3.7.1.2 WCHAR....276
3.7.2 Zeichenketten....277
3.7.2.1 STRING....277
3.7.2.2 Aufbau einer STRING-Variablen....279
3.7.2.3 WSTRING....280
3.7.2.4 Einzelne Zeichen eines STRINGs oder WSTRINGs adressieren....283
3.7.2.5 Variable vom Datentyp STRING oder WSTRING übergeben....285

3.8 PLC-Datentypen (UDT)....286
3.8.1 Grundlagen zu PLC-Datentypen (UDT)....286
3.8.2 Variable vom Datentyp PLC-Datentyp (UDT) übergeben....289
3.8.3 PLC-Datentypen (UDT) adressieren....290

3.9 Datenstruktur STRUCT (Anonyme Strukturen)....291
3.9.1 Grundlagen zu STRUCT....291
3.9.2 Aufbau einer STRUCT-Variablen....293
3.9.3 STRUCT-Komponenten adressieren....294
3.9.4 Variablen vom Datentyp STRUCT übergeben....295

3.10 ARRAY....296
3.10.1 Grundlagen zum ARRAY....296
3.10.2 Aufbau einer ARRAY-Variablen....302
3.10.3 ARRAY-Komponenten adressieren....303
3.10.4 Variablen vom Datentyp ARRAY / ARRAY[*] übergeben....305
3.10.5 Beispiele für die Verwendung von ARRAYs....310
3.10.5.1 Beispiel für ein mehrdimensionales ARRAY....310
3.10.5.2 Beispiel für die Berechnung des Skalarprodukts zweier Vektoren mit ARRAY[*]....311

3.11 Zeiger....315
3.11.1 Referenzen....315
3.11.1.1 Grundlagen zu Referenzen....315
3.11.1.2 Referenzen deklarieren....318
3.11.1.3 Referenzieren....320
3.11.1.4 Dereferenzieren....324
3.11.1.5 Standardanweisungen mit Referenzen....326
3.11.1.6 Zuweisungsversuch von VARIANT auf eine Referenz....329
3.11.1.7 Referenzen als Bausteinparameter übergeben....331
3.11.1.8 Beispiel: Übergabe von Variablen unterschiedlichen Datentyps mithilfe von Referenzen....334
3.11.1.9 Beispiel: Daten per Referenz an den aufrufenden Baustein zurückgeben....336
3.11.2 VARIANT....339
3.11.2.1 Grundlagen zum VARIANT....339
3.11.2.2 Anwendungsfälle für Zeiger im Vergleich....341
3.11.2.3 VARIANT-Anweisungen....342
3.11.2.4 Anwendungsmöglichkeiten für VARIANT-Anweisungen....343
3.11.2.5 VARIANT initialisieren....346
3.11.2.6 Verschiedene Datentypen mit VARIANT übergeben und auslesen....347
3.11.2.7 Beispiele für die Verwendung des Datentyps VARIANT....348
3.11.3 POINTER....356
3.11.4 ANY....359

3.12 Parametertypen....363
3.12.1 Parametertypen....363

3.13 Systemdatentypen....364
3.13.1 Systemdatentypen....364

3.14 Hardware-Datentypen....366
3.14.1 Hardware-Datentypen....366

3.15 Datentypkonvertierungen bei S7-1500....369
3.15.1 Übersicht über die Datentypkonvertierung....369
3.15.2 Implizite Konvertierungen....371
3.15.2.1 IEC-Prüfung aktivieren oder deaktivieren....371
3.15.2.2 Binärzahlen....372
3.15.2.3 Ganzzahlen....380
3.15.2.4 Gleitpunktzahlen....394
3.15.2.5 Zeiten....398
3.15.2.6 Datum und Uhrzeit....404
3.15.2.7 Zeichenfolgen....416
3.15.3 Explizite Konvertierungen....423
3.15.3.1 Binärzahlen....423
3.15.3.2 Ganzzahlen....439
3.15.3.3 Gleitpunktzahlen....468
3.15.3.4 Zeiten....474
3.15.3.5 Datum und Uhrzeit....482
3.15.3.6 Zeichenfolgen....497

3.16 Datentypkonvertierungen bei S7-1200....507
3.16.1 Übersicht über die Datentypkonvertierung....507
3.16.2 Implizite Konvertierungen....509
3.16.2.1 IEC-Prüfung aktivieren oder deaktivieren....509
3.16.2.2 Binärzahlen....510

3.16.2.3 Ganzzahlen....514
3.16.2.4 Gleitpunktzahlen....520
3.16.2.5 Zeiten....522
3.16.2.6 Datum und Uhrzeit....523
3.16.2.7 Zeichenfolgen....526
3.16.3 Explizite Konvertierungen....530
3.16.3.1 Binärzahlen....530
3.16.3.2 Ganzzahlen....540
3.16.3.3 Gleitpunktzahlen....560
3.16.3.4 Zeiten....566
3.16.3.5 Datum und Uhrzeit....569
3.16.3.6 Zeichenfolgen....573

3.17 Datentypkonvertierungen bei S7-300, S7-400....578
3.17.1 Übersicht über die Datentypkonvertierungen....578
3.17.2 Implizite Konvertierungen....580
3.17.2.1 IEC-Prüfung aktivieren oder deaktivieren....580
3.17.2.2 Binärzahlen....581
3.17.2.3 Ganzzahlen....585
3.17.2.4 Gleitpunktzahlen....586
3.17.2.5 Zeiten....587
3.17.2.6 Datum und Uhrzeit....589
3.17.2.7 Zeichenfolgen....591
3.17.3 Explizite Konvertierungen....592
3.17.3.1 Binärzahlen....592
3.17.3.2 Ganzzahlen....600
3.17.3.3 Gleitpunktzahlen....606
3.17.3.4 Zeiten....607
3.17.3.5 Datum und Uhrzeit....609
3.17.3.6 Zeichenfolgen....612
3.17.3.7 Weitere Konvertierungsfunktionen....614

4 Anweisungen....615

4.1 Anweisungen....615
4.1.1 Beispielbibliotheken....615
4.1.1.1 Sample Library for Instructions....615
4.1.1.2 Library of General Functions....617
4.1.2 Asynchron arbeitende Anweisungen....618
4.1.2.1 Unterschied zwischen synchron und asynchron arbeitenden Anweisungen....618
4.1.3 Einfache Anweisungen....621
4.1.3.1 KOP....621
4.1.3.2 FUP....976
4.1.3.3 AWL....1336
4.1.3.4 SCL....1726
4.1.3.5 GRAPH....1983
4.1.4 Erweiterte Anweisungen....2178
4.1.4.1 Datum und Uhrzeit....2178
4.1.4.2 String + Char....2215
4.1.4.3 Prozessabbild....2289
4.1.4.4 Dezentrale Peripherie....2298
4.1.4.5 PROFIenergy....2464
4.1.4.6 Baugruppenparametrierung....2521
4.1.4.7 Alarme....2534

4.1.4.8 Meldungen....2624
4.1.4.9 Diagnose....2676
4.1.4.10 Puls....2752
4.1.4.11 Rezepturen und Data Logging....2757
4.1.4.12 Datenbausteinfunktionen....2821
4.1.4.13 Adressierung....2849
4.1.4.14 File Handling....2870
4.1.4.15 R/H-System....2875
4.1.5 Technologie....2877
4.1.5.1 Schnelle Zähler....2877
4.1.5.2 Zählen und Messen (S7-1500)....2883
4.1.5.3 PID Control....2900
4.1.5.4 Motion Control....3135
4.1.5.5 Time-based IO....3568
4.1.6 Kommunikation....3589
4.1.6.1 S7-Kommunikation....3589
4.1.6.2 Open User Communication....3625
4.1.6.3 OPC UA....3797
4.1.6.4 Webserver....4002
4.1.6.5 Kommunikationsprozessor....4013
4.1.6.6 TeleService....4362

4.2 Anweisungen....4370
4.2.1 Asynchron arbeitende Anweisungen....4370
4.2.1.1 Unterschied zwischen synchron und asynchron arbeitenden Anweisungen....4370
4.2.2 Einfache Anweisungen....4373
4.2.2.1 KOP....4373
4.2.2.2 FUP....4591
4.2.2.3 AWL....4814
4.2.2.4 SCL....5115
4.2.2.5 GRAPH....5267
4.2.3 Erweiterte Anweisungen....5403
4.2.3.1 Datum und Uhrzeit....5403
4.2.3.2 String + Char....5436
4.2.3.3 Prozessabbild....5449
4.2.3.4 Dezentrale Peripherie....5456
4.2.3.5 PROFIenergy....5541
4.2.3.6 Baugruppenparametrierung....5593
4.2.3.7 Alarme....5604
4.2.3.8 Meldungen....5632
4.2.3.9 Diagnose....5681
4.2.3.10 Datenbausteinfunktionen....5753
4.2.3.11 Tabellenfunktionen....5766
4.2.3.12 Adressierung....5789
4.2.3.13 Weitere Funktionen....5798
4.2.4 Technologie....5800
4.2.4.1 PID Control....5800
4.2.4.2 Motion Control....6050
4.2.4.3 Funktionsmodule....6112
4.2.4.4 300C Funktionen....6346
4.2.5 Kommunikation....6376
4.2.5.1 Gemeinsame Parameter- und Fehlerinformationen....6376
4.2.5.2 S7-Kommunikation....6381

4.2.5.3 Open User Communication....6440
4.2.5.4 Webserver....6497
4.2.5.5 Weitere Kommunikation....6499
4.2.5.6 Kommunikationsprozessor....6499
4.2.5.7 300C Funktionen....7047
4.2.5.8 Kommunikation mit iSlave/iDevice....7066
4.2.5.9 PROFINET CBA....7072
4.2.5.10 MPI-Kommunikation....7077
4.2.5.11 TeleService....7088

5 Programmiereditor....7117

5.1 Übersicht über den Programmiereditor....7117

5.2 Tastaturbedienung im Programmiereditor....7122

5.3 Fläche des Programmierfensters vergrößern....7127

5.4 Einstellen der Mnemonik....7130

5.5 Symbolische und absolute Operanden anzeigen....7131

5.6 Nach Anweisungen suchen....7133

5.7 Mehrsprachige Projekttexte in Bausteinen bearbeiten....7134

5.8 Anweisungsversionen verwenden....7137
5.8.1 Grundlagen zu Anweisungsversionen....7137
5.8.2 Projekt und Anweisungen hochrüsten....7138
5.8.3 Programm der aktuellen CPU aktualisieren....7139

5.9 Anweisungsprofile verwenden....7141
5.9.1 Grundlagen zu Anweisungsprofilen....7141
5.9.2 Neues Anweisungsprofil erzeugen....7141
5.9.3 Anweisungsprofil öffnen und bearbeiten....7143
5.9.4 Anweisungsprofil aktivieren und deaktivieren....7144
5.9.5 Anweisungsprofil löschen....7145

5.10 Autovervollständigung verwenden....7147
5.10.1 Grundlagen zur Autovervollständigung....7147
5.10.2 Autovervollständigung in grafischen Programmiersprachen verwenden....7147
5.10.3 Autovervollständigung in textuellen Programmiersprachen verwenden....7149

5.11 Allgemeine Einstellungen für die PLC-Programmierung....7150
5.11.1 Übersicht über die allgemeinen Einstellungen....7150
5.11.2 Einstellungen ändern....7153

5.12 Syntaxfehler im Programm beheben....7154
5.12.1 Grundlagen zu Syntaxfehlern....7154
5.12.2 Syntaxfehler im Programm finden....7154

5.13 Programmiersprache umschalten....7156
5.13.1 Regeln zum Umschalten der Programmiersprache....7156
5.13.2 Programmiersprache umschalten....7156

6 Software Units verwenden....7159

6.1 Grundlagen zu Software Units....7159
6.1.1 Einführung zu Software Units....7159
6.1.2 PLC-Programm in Software Units erstellen....7163

6.1.3 Mögliche Programmstrukturen mit Software Units....7165

6.2 Software Units anlegen und verwalten....7169
6.2.1 Software Units anlegen....7169
6.2.2 Software Units umbenennen....7169
6.2.3 Software Units kopieren und einfügen....7170
6.2.4 Software Units löschen....7171
6.2.5 Eigenschaften einer Software Unit anzeigen und ändern....7172
6.2.6 Software Unit-Elemente in Bibliotheken verwalten....7173

6.3 Programmelemente einfügen....7175
6.3.1 Bausteine anlegen....7175
6.3.2 PLC-Variablen deklarieren....7176
6.3.3 PLC-Datentypen anlegen....7178
6.3.4 Bausteinaufrufe einfügen....7179
6.3.5 Programmelemente kopieren oder verschieben....7181

6.4 Relationen erstellen....7185
6.4.1 Grundlagen zu Relationen....7185
6.4.2 Programmelemente veröffentlichen....7187
6.4.3 Relationen einrichten....7188

6.5 Software Units übersetzen und laden....7190
6.5.1 Grundlagen zum Übersetzen und Laden von Software Units....7190
6.5.2 Software Units übersetzen....7192
6.5.3 Software Units laden....7193
6.5.3.1 Software Units in ein Gerät laden....7193
6.5.3.2 Software Units von einem Gerät laden....7195
6.5.3.3 Software Units auf eine Memory Card laden....7196
6.5.3.4 Software Units von einer Memory Card laden....7198

6.6 Software Units vergleichen....7200
6.6.1 Grundlagen zum Software Unit-Vergleich....7200

7 Bausteine anlegen und verwalten....7201

7.1 Bausteine anlegen....7201
7.1.1 Bausteinordner....7201
7.1.2 Organisationsbausteine anlegen....7202
7.1.3 Funktionen und Funktionsbausteine anlegen....7203
7.1.4 Datenbausteine anlegen....7204
7.1.5 Bausteine aus Bibliotheken verwenden....7205
7.1.6 Bausteine kopieren und einfügen....7207
7.1.6.1 Grundlagen zum Kopieren und Einfügen von Bausteinen....7207
7.1.6.2 Bausteine kopieren....7210
7.1.6.3 Bausteine einfügen....7211
7.1.7 Bausteintitel eingeben....7211
7.1.8 Bausteinkommentar eingeben....7212

7.2 Bausteineigenschaften festlegen....7214
7.2.1 Grundlagen zu Bausteineigenschaften....7214
7.2.2 Übersicht über die Bausteineigenschaften....7215
7.2.3 Zeitstempel von Bausteinen....7219
7.2.4 Bausteineigenschaften anzeigen und bearbeiten....7222

7.3 Bausteine verwalten....7224
7.3.1 Bausteine öffnen....7224

7.3.2 Bausteine speichern....7226
7.3.3 Bausteine schließen....7226
7.3.4 Bausteine umbenennen....7227
7.3.5 Bausteine offline löschen....7227
7.3.6 Bausteine online löschen....7228
7.3.7 CPU-Datenbausteine löschen....7229

7.4 Externe Quelldateien verwenden....7230
7.4.1 Grundlagen zur Verwendung von externen Quelldateien....7230
7.4.2 Regeln zur Programmierung von externen Quelldateien....7231
7.4.3 Baustein als externe Quelldatei speichern....7233
7.4.4 Externe Quelldateien einbinden....7235
7.4.5 Externe Quelldateien öffnen und bearbeiten....7235
7.4.6 Bausteine aus externen Quelldateien generieren....7236

8 Bausteine schützen....7239

8.1 Schützen von Bausteinen....7239

8.2 Passwort-Provider anbinden....7242

8.3 Verfügbare Passwort-Provider anzeigen....7245

8.4 Aktivitäten der Passwort-Provider aufzeichnen....7246

8.5 Art der Passwortvergabe festlegen....7248

8.6 Kopierschutz für Bausteine einrichten und entfernen....7250

8.7 Schreibschutz für Bausteine einrichten und entfernen....7252

8.8 Know-how-Schutz für Bausteine einrichten....7255

8.9 Know-how-geschützte Bausteine öffnen....7256

8.10 Know-how-geschützte Bausteine drucken....7258

8.11 Passwort ändern....7260

8.12 Know-how-Schutz für Bausteine entfernen....7261

9 Bausteinschnittstelle deklarieren....7263

9.1 Übersicht über die Bausteinschnittstelle....7263

9.2 Regeln zur Deklaration der Bausteinschnittstelle....7265
9.2.1 Allgemeine Regeln zur Deklaration der Bausteinschnittstelle....7265
9.2.2 Gültige Datentypen in der Bausteinschnittstelle....7266
9.2.3 Gültige Datentypen in der Bausteinschnittstelle....7267
9.2.4 Remanenz von lokalen Variablen....7269

9.3 Tabellarische oder textuelle Darstellung der Bausteinschnittstelle wählen....7271

9.4 Bausteinschnittstelle deklarieren....7272
9.4.1 Aufbau der Bausteinschnittstelle....7272
9.4.2 Lokale Variablen und Konstanten in der Bausteinschnittstelle deklarieren....7274
9.4.3 Lokale Variable im Programmiereditor deklarieren....7275
9.4.4 Variablen vom Datentyp ARRAY deklarieren....7277
9.4.5 Variablen vom Datentyp STRUCT deklarieren....7279
9.4.6 Variablen vom Datentyp STRING und WSTRING deklarieren....7280
9.4.7 Variablen basierend auf einem PLC-Datentyp deklarieren....7281
9.4.8 Überlagerte Variablen deklarieren....7281

9.4.9 Multiinstanzen deklarieren....7282
9.4.10 Parameterinstanzen deklarieren....7283
9.4.11 Parameter beim Bausteinaufruf verstecken....7284
9.4.12 Vordefinierte Aktualparameter deklarieren....7285
9.4.13 Tabelle der Bausteinschnittstelle bearbeiten....7287
9.4.13.1 Tabellenzeilen einfügen....7287
9.4.13.2 Tabellenzeilen hinzufügen....7287
9.4.13.3 Variablen löschen....7287
9.4.13.4 Aufeinander folgende Zellen automatisch ausfüllen....7287
9.4.13.5 Tabellenspalten ein- und ausblenden....7288
9.4.13.6 Variablen mit externen Editoren bearbeiten....7289

9.5 Bausteinschnittstelle in der textuellen Ansicht deklarieren....7290
9.5.1 Aufbau der textuellen Bausteinschnittstelle....7290
9.5.2 Bausteindeklaration und Rückgabewert....7291
9.5.3 Deklarationsabschnitte....7292
9.5.4 Variablendeklaration und Initialisierung....7293
9.5.5 Deklaration von STRUCT....7294
9.5.6 Deklaration von ARRAY....7297
9.5.7 Deklaration von STRING und WSTRING....7300
9.5.8 Deklaration von Instanzen....7301
9.5.9 Attribute für Variablen....7302
9.5.10 Kommentare....7303
9.5.11 Textuelle Bausteinschnittstelle bearbeiten....7303
9.5.12 Standard-Schnittstelle eines Organisationsbausteins wiederherstellen....7305
9.5.13 Beispiel einer textuellen Bausteinschnittstelle....7306

9.6 Eigenschaften von lokalen Variablen und Konstanten bearbeiten....7308
9.6.1 Eigenschaften von lokalen Variablen und Konstanten....7308
9.6.2 Eigenschaften von lokalen Variablen und Konstanten ändern....7310
9.6.3 Mehrsprachige Kommentare in der Bausteinschnittstelle bearbeiten....7312

9.7 Bausteinschnittstelle aktualisieren....7314

9.8 Bausteinschnittstelle erweitern....7316

10 Datenbausteine programmieren....7319

10.1 Grundlagen zur Programmierung von Datenbausteinen....7319

10.2 Aufbau der Deklarationstabelle für Datenbausteine....7322

10.3 Datenbausteine anlegen....7324

10.4 Datenbausteine aktualisieren....7326

10.5 Datenbausteine erweitern....7328

10.6 Datenstruktur von globalen Datenbausteinen erstellen....7330
10.6.1 Variablen von elementarem Datentyp deklarieren....7330
10.6.2 Variablen vom Datentyp ARRAY deklarieren....7331
10.6.3 Variablen vom Datentyp STRUCT deklarieren....7332
10.6.4 Variablen vom Datentyp STRING deklarieren....7334
10.6.5 Variablen basierend auf einem PLC-Datentyp deklarieren....7335

10.7 Startwerte definieren....7336
10.7.1 Grundlagen zu Startwerten....7336
10.7.2 Startwerte definieren....7337

10.8 Remanenz einstellen....7338
10.8.1 Remanenz von Variablen in Datenbausteinen....7338
10.8.2 Remanenz in einem Instanz-Datenbaustein einstellen....7338
10.8.3 Remanenz in einem globalen Datenbaustein einstellen....7339

10.9 Eigenschaften von Variablen in Datenbausteinen bearbeiten....7341
10.9.1 Eigenschaften von Variablen in Datenbausteinen....7341
10.9.2 Eigenschaften von Variablen in Instanz-Datenbausteinen ändern....7344
10.9.3 Eigenschaften von Variablen in globalen Datenbausteinen ändern....7345
10.9.4 Mehrsprachige Kommentare im Datenbaustein bearbeiten....7346

10.10 Deklarationstabelle für Datenbausteine bearbeiten....7348
10.10.1 Tabellenzeilen einfügen....7348
10.10.2 Tabellenzeilen hinzufügen....7348
10.10.3 Variablen löschen....7348
10.10.4 Aufeinander folgende Zellen automatisch ausfüllen....7349
10.10.5 Tabellenspalten ein- und ausblenden....7350
10.10.6 Variablen mit externen Editoren bearbeiten....7350

10.11 Online- und Diagnosefunktionen für Datenbausteine....7352
10.11.1 Datenbausteine im Online-Betrieb bearbeiten....7352
10.11.2 Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen....7354
10.11.3 Variablen beobachten....7355
10.11.4 Variablen steuern....7356
10.11.5 Startwerte als Aktualwerte laden....7357
10.11.6 Momentaufnahme der Aktualwerte anlegen....7359
10.11.7 Momentaufnahmen als Aktualwerte laden....7361
10.11.8 Momentaufnahme in die Startwerte kopieren....7363
10.11.9 Einstellwerte während der Inbetriebnahme nutzen....7365
10.11.9.1 Grundlagen zum Anpassen von Einstellwerten während der Inbetriebnahme....7365
10.11.9.2 Daten als einstellbare Werte kennzeichnen....7365
10.11.9.3 Einstellwerte im Online-Programm initialisieren....7367
10.11.9.4 Werte aus dem Online-Programm als Startwerte übernehmen....7368

11 PLC-Variablen deklarieren....7371

11.1 Übersicht über PLC-Variablentabellen....7371

11.2 Aufbau der PLC-Variablentabellen....7373

11.3 Regeln für PLC-Variablen....7375
11.3.1 Zulässige Namen von PLC-Variablen....7375
11.3.2 Zulässige Adressen und Datentypen von PLC-Variablen....7376

11.4 PLC-Variablentabellen anlegen und verwalten....7379
11.4.1 PLC-Variablentabelle anlegen....7379
11.4.2 PLC-Variablentabellen gruppieren....7379
11.4.3 PLC-Variablentabelle öffnen....7380

11.5 PLC-Variablen deklarieren....7382
11.5.1 PLC-Variablendeklaration eingeben....7382
11.5.1.1 Variablen in der PLC-Variablentabelle deklarieren....7382
11.5.1.2 PLC-Variablen im Programmiereditor deklarieren....7383
11.5.1.3 Zur Gerätesicht navigieren....7384
11.5.2 Remanenz von PLC-Variablen einstellen....7384
11.5.2.1 Remanenzverhalten von PLC-Variablen....7384

11.5.2.2 Remanenzverhalten von PLC-Variablen einstellen....7385
11.5.2.3 Remanenzverhalten von PLC-Variablen einstellen....7386

11.6 PLC-Variablen für Ein- und Ausgänge in Strukturen zusammenfassen....7387
11.6.1 Wissenswertes zu strukturierten PLC-Variablen....7387
11.6.2 Strukturierte PLC-Variablen anlegen....7389

11.7 Globale Konstanten deklarieren....7391
11.7.1 Regeln für globale Anwenderkonstanten....7391
11.7.2 Regeln für globale Systemkonstanten....7392
11.7.3 Globale Konstanten deklarieren....7392

11.8 Eigenschaften bearbeiten....7394
11.8.1 Eigenschaften von PLC-Variablen bearbeiten....7394
11.8.1.1 Eigenschaften von PLC-Variablen....7394
11.8.1.2 Eigenschaften von PLC-Variablen bearbeiten....7394
11.8.2 Eigenschaften von globalen Konstanten bearbeiten....7396
11.8.2.1 Eigenschaften von globalen Konstanten....7396
11.8.2.2 Eigenschaften von globalen Konstanten bearbeiten....7396
11.8.3 Mehrsprachige Kommentare in der PLC-Variablentabelle bearbeiten....7396

11.9 PLC-Variablen beobachten....7398
11.9.1 PLC-Variablen beobachten....7398

11.10 PLC-Variablentabellen bearbeiten....7400
11.10.1 Tabellenzeilen in der PLC-Variablentabelle einfügen....7400
11.10.2 Einträge in der PLC-Variablentabelle kopieren....7400
11.10.3 Einträge in der PLC-Variablentabelle löschen....7401
11.10.4 Zeilen in PLC-Variablentabellen sortieren....7401
11.10.5 Zellen in der PLC-Variablentabelle automatisch ausfüllen....7401
11.10.6 Tabellenspalten ein- und ausblenden....7402
11.10.7 PLC-Variablen exportieren und importieren....7402
11.10.7.1 Grundlagen zum Exportieren und Importieren von PLC-Variablen....7402
11.10.7.2 Format der Exportdatei (*.xlsx)....7403
11.10.7.3 Format der Exportdatei (*.xml)....7404
11.10.7.4 Format der Exportdatei (*.sdf)....7404
11.10.7.5 Exportieren von PLC-Variablen....7405
11.10.7.6 Importieren von PLC-Variablen....7406
11.10.7.7 Einzelne PLC-Variablen mit externen Editoren bearbeiten....7406

12 PLC-Datentypen (UDT) deklarieren 7409

12.1 Aufbau der Deklarationstabelle für PLC-Datentypen....7409

12.2 PLC-Datentypen in der Projektnavigation....7411

12.3 PLC-Datentypen anlegen....7412

12.4 PLC-Datentypen löschen....7413

12.5 PLC-Datentypen umnummerieren....7414

12.6 Struktur von PLC-Datentypen programmieren....7415
12.6.1 Variablen von elementarem Datentyp deklarieren....7415
12.6.2 Variablen vom Datentyp ARRAY deklarieren....7415
12.6.3 Variablen vom Datentyp STRUCT deklarieren 7417
12.6.4 Variablen basierend auf einem anderen PLC-Datentyp deklarieren....7418

12.7 Eigenschaften von Variablen in PLC-Datentypen bearbeiten....7419

12.7.1 Eigenschaften von Variablen in PLC-Datentypen....7419
12.7.2 Eigenschaften von Variablen in PLC-Datentypen ändern....7420
12.7.3 Mehrsprachige Kommentare im PLC-Datentyp bearbeiten....7420

12.8 Deklarationstabelle für PLC-Datentypen bearbeiten....7422
12.8.1 Tabellenzeilen einfügen....7422
12.8.2 Tabellenzeilen hinzufügen....7422
12.8.3 Variablen löschen....7422
12.8.4 Aufeinander folgende Zellen automatisch ausfüllen....7423
12.8.5 Tabellenspalten ein- und ausblenden....7423

13 KOP-Programme erstellen....7425

13.1 Grundlagen zu KOP....7425
13.1.1 Programmiersprache KOP....7425
13.1.2 Übersicht über die KOP-Elemente....7425

13.2 Einstellungen zu KOP....7428
13.2.1 Übersicht über die Einstellungen zu KOP....7428
13.2.2 Einstellungen ändern....7428

13.3 Arbeiten mit Netzwerken....7430
13.3.1 Netzwerke verwenden....7430
13.3.2 Netzwerke einfügen....7431
13.3.3 Netzwerke selektieren....7432
13.3.4 Netzwerke kopieren und einfügen....7433
13.3.5 Netzwerke löschen....7433
13.3.6 Netzwerke reduzieren und erweitern....7434
13.3.7 Netzwerktitel einfügen....7435
13.3.8 Netzwerkkommentar eingeben....7436
13.3.9 In Netzwerken navigieren....7437

13.4 KOP-Elemente einfügen....7439
13.4.1 Regeln für die Verwendung von KOP-Elementen....7439
13.4.2 Unzulässige Verschaltungen in KOP....7441
13.4.3 KOP-Elemente über Task Card "Anweisungen" einfügen....7441
13.4.4 KOP-Elemente mithilfe einer Leerbox einfügen....7442
13.4.5 Datentyp eines KOP-Elements auswählen....7444
13.4.5.1 Auswahl eines Datentyps....7444
13.4.5.2 Datentyp einer Anweisung festlegen....7445
13.4.6 Favoriten in KOP verwenden....7446
13.4.6.1 KOP-Elemente zu den Favoriten hinzufügen....7446
13.4.6.2 KOP-Elemente mithilfe von Favoriten einfügen....7447
13.4.6.3 KOP-Elemente aus den Favoriten entfernen....7448
13.4.7 Bausteinaufrufe in KOP einfügen....7449
13.4.7.1 Bausteinaufrufe per Drag & Drop einfügen....7449
13.4.7.2 Bausteinaufrufe in KOP aktualisieren....7450
13.4.7.3 Bausteinaufruf ändern....7452
13.4.7.4 Instanztyp ändern....7453
13.4.8 Komplexe KOP-Anweisungen einfügen....7454
13.4.8.1 Anweisung "Berechnen" verwenden....7454
13.4.9 Freie Kommentare verwenden....7455
13.4.9.1 Grundlagen zur Verwendung von freien Kommentaren in KOP....7455
13.4.9.2 Freie Kommentare einfügen....7456
13.4.9.3 Freie Kommentare bearbeiten....7456

13.4.9.4 Freie Kommentare löschen....7458

13.5 KOP-Elemente bearbeiten....7459
13.5.1 KOP-Elemente selektieren....7459
13.5.2 KOP-Elemente kopieren....7460
13.5.3 KOP-Elemente ausschneiden....7460
13.5.4 KOP-Elemente aus der Zwischenablage einfügen....7461
13.5.5 KOP-Elemente ersetzen....7462
13.5.6 Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen....7462
13.5.7 Eingänge und Ausgänge entfernen....7464
13.5.8 KOP-Elemente löschen....7465

13.6 Operanden in KOP-Anweisungen beschalten....7466
13.6.1 Operanden beschalten....7466
13.6.2 Versteckte Parameter beschalten....7467
13.6.3 Variableninformationen ein- oder ausblenden....7468

13.7 Zweige in KOP....7471
13.7.1 Grundlagen zu Zweigen in KOP....7471
13.7.2 Regeln für Zweige in KOP....7471
13.7.3 Zweige in KOP-Netzwerke einfügen....7472
13.7.4 Zweige in KOP-Netzwerke schließen....7473
13.7.5 Zweige in KOP-Netzwerke löschen....7473

13.8 Kreuzungen in KOP....7474
13.8.1 Grundlagen zu Kreuzungen in KOP....7474
13.8.2 Kreuzungen einfügen....7474
13.8.3 Kreuzungen umhängen....7475
13.8.4 Kreuzungen löschen....7476

13.9 Strompfade in KOP....7477
13.9.1 Grundlagen zu Strompfaden in KOP....7477
13.9.2 Strompfad einfügen....7478
13.9.3 Strompfad löschen....7478

13.10 Programmierbeispiele KOP....7480
13.10.1 Beispiel für das Steuern eines Förderbands....7480
13.10.2 Beispiel für das Erfassen der Richtung eines Förderbands....7482
13.10.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs....7484
13.10.4 Beispiel für das Berechnen einer Gleichung....7487
13.10.5 Beispiel für das Kontrollieren der Raumtemperatur....7490

14 FUP-Programme erstellen....7493

14.1 Grundlagen zu FUP....7493
14.1.1 Programmiersprache FUP....7493
14.1.2 Übersicht über die FUP-Elemente....7493

14.2 Einstellungen zu FUP....7496
14.2.1 Übersicht über die Einstellungen zu FUP....7496
14.2.2 Einstellungen ändern....7496

14.3 Arbeiten mit Netzwerken....7498
14.3.1 Netzwerke verwenden....7498
14.3.2 Netzwerke einfügen....7499
14.3.3 Netzwerke selektieren....7500
14.3.4 Netzwerke kopieren und einfügen....7500

14.3.5 Netzwerke löschen....7501
14.3.6 Netzwerke reduzieren und erweitern....7502
14.3.7 Netzwerktitel einfügen....7502
14.3.8 Netzwerkkommentar eingeben....7504
14.3.9 In Netzwerken navigieren....7505

14.4 FUP-Elemente einfügen....7507
14.4.1 Regeln für die Verwendung von FUP-Elementen....7507
14.4.2 FUP-Elemente über Task Card "Anweisungen" einfügen....7508
14.4.3 FUP-Elemente mithilfe einer Leerbox einfügen....7509
14.4.4 Datentyp eines FUP-Elements auswählen....7511
14.4.4.1 Auswahl eines Datentyps....7511
14.4.4.2 Datentyp einer Anweisung festlegen....7512
14.4.5 Favoriten in FUP verwenden....7513
14.4.5.1 FUP-Elemente zu den Favoriten hinzufügen....7513
14.4.5.2 FUP-Elemente mithilfe von Favoriten einfügen....7514
14.4.5.3 FUP-Elemente aus den Favoriten entfernen....7515
14.4.6 Bausteinaufrufe in FUP einfügen....7516
14.4.6.1 Bausteinaufrufe per Drag & Drop einfügen....7516
14.4.6.2 Bausteinaufrufe in FUP aktualisieren....7517
14.4.6.3 Bausteinaufruf ändern....7519
14.4.6.4 Instanztyp ändern....7520
14.4.7 Komplexe FUP-Anweisungen einfügen....7521
14.4.7.1 Anweisung "Berechnen" verwenden....7521
14.4.8 Freie Kommentare verwenden....7522
14.4.8.1 Grundlagen zur Verwendung von freien Kommentaren in FUP 7522
14.4.8.2 Freie Kommentare einfügen....7523
14.4.8.3 Freie Kommentare bearbeiten....7523
14.4.8.4 Freie Kommentare löschen....7525

14.5 FUP-Elemente bearbeiten....7526
14.5.1 FUP-Elemente selektieren....7526
14.5.2 FUP-Elemente kopieren....7526
14.5.3 FUP-Elemente ausschneiden....7527
14.5.4 FUP-Elemente aus der Zwischenablage einfügen....7528
14.5.5 FUP-Elemente ersetzen....7529
14.5.6 Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen....7529
14.5.7 Eingänge und Ausgänge einer Anweisung entfernen....7531
14.5.8 FUP-Elemente löschen....7532

14.6 Operanden in FUP-Anweisungen beschalten....7533
14.6.1 Operanden beschalten....7533
14.6.2 Versteckte Parameter beschalten....7534
14.6.3 Variableninformationen ein- oder ausblenden....7535

14.7 Zweige in FUP....7537
14.7.1 Grundlagen zu Zweigen in FUP....7537
14.7.2 Regeln für Zweige in FUP....7537
14.7.3 Zweige in FUP-Netzwerke einfügen....7538
14.7.4 Zweige in FUP-Netzwerke löschen....7538

14.8 Verknüpfungspfade in FUP....7540
14.8.1 Grundlagen zu Verknüpfungspfaden in FUP....7540
14.8.2 Verknüpfungspfad einfügen....7541
14.8.3 Verknüpfungspfad löschen....7541

14.9 Programmierbeispiele FUP....7543
14.9.1 Beispiel für das Steuern eines Förderbands....7543
14.9.2 Beispiel für das Erfassen der Richtung eines Förderbands....7545
14.9.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs....7547
14.9.4 Beispiel für das Berechnen einer Gleichung....7551
14.9.5 Beispiel für das Kontrollieren der Raumtemperatur....7553

15 AWL-Programme erstellen....7557

15.1 Grundlagen zu AWL....7557
15.1.1 Programmiersprache AWL....7557
15.1.2 Datenaustausch in AWL....7557
15.1.3 Parameterversorgung über Register....7559
15.1.3.1 Parameterversorgung über Register beim Bausteinaufruf in AWL....7559
15.1.3.2 Parameterversorgung über Register aktivieren....7561

15.2 Einstellungen zu AWL....7563
15.2.1 Übersicht über die Einstellungen zu AWL....7563
15.2.2 Einstellungen ändern....7563

15.3 Arbeiten mit Netzwerken....7564
15.3.1 Netzwerke verwenden....7564
15.3.2 Netzwerke einfügen....7564
15.3.3 Netzwerke selektieren....7565
15.3.4 Netzwerke kopieren und einfügen....7566
15.3.5 Netzwerke löschen....7566
15.3.6 Netzwerke reduzieren und erweitern....7567
15.3.7 Netzwerktitel einfügen....7568
15.3.8 Netzwerkkommentar eingeben....7569
15.3.9 In Netzwerken navigieren....7570

15.4 AWL-Anweisungen einfügen....7572
15.4.1 Regeln zu AWL....7572
15.4.2 Anweisungen der globalen Bibliothek "Long Functions"....7573
15.4.3 AWL-Anweisungen manuell eingeben....7575
15.4.4 AWL-Anweisungen über Task Card "Anweisungen" einfügen....7576
15.4.5 Datentyp einer AWL-Anweisung auswählen....7578
15.4.5.1 Auswahl eines Datentyps....7578
15.4.5.2 Datentyp einer Anweisung festlegen....7578
15.4.6 Operanden einfügen....7579
15.4.7 Variableninformationen ein- oder ausblenden....7580
15.4.8 Favoriten in AWL verwenden....7581
15.4.8.1 AWL-Elemente zu den Favoriten hinzufügen....7581
15.4.8.2 AWL-Anweisungen mithilfe von Favoriten einfügen....7582
15.4.8.3 AWL-Anweisungen aus den Favoriten entfernen....7583
15.4.9 Bausteinaufrufe in AWL einfügen....7583
15.4.9.1 Bausteinaufrufe manuell einfügen....7583
15.4.9.2 Bausteinaufrufe per Drag & Drop einfügen....7585
15.4.9.3 Bausteinaufrufe in AWL aktualisieren....7586
15.4.9.4 Instanztyp ändern....7588
15.4.10 Kommentare einfügen....7590

15.5 AWL-Anweisungen bearbeiten....7592
15.5.1 AWL-Anweisungen selektieren....7592
15.5.2 AWL-Anweisungen kopieren....7592

15.5.3 AWL-Anweisungen ausschneiden....7593
15.5.4 AWL-Anweisungen aus der Zwischenablage einfügen....7594
15.5.5 AWL-Anweisungen löschen....7594

15.6 Programmierbeispiele AWL....7595
15.6.1 Beispiel für das Steuern eines Förderbands....7595
15.6.2 Beispiel für das Erfassen der Richtung eines Förderbands....7596
15.6.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs....7598
15.6.4 Beispiel für das Berechnen einer Gleichung....7600
15.6.5 Beispiel für das Heizen eines Ofens....7602
15.6.6 Beispiel für eine Schrittkette....7604

16 SCL-Programme erstellen....7607

16.1 Grundlagen zu SCL....7607
16.1.1 Programmiersprache SCL....7607
16.1.2 Ausdrücke....7608
16.1.3 Arithmetische Ausdrücke....7609
16.1.4 Vergleichsausdrücke....7612
16.1.5 Logische Ausdrücke....7617
16.1.6 Operatoren und Operatorenrangfolge....7618
16.1.7 Wertzuweisungen....7620
16.1.8 Mit Gleitpunktzahlen (REAL und LREAL) in SCL rechnen....7623
16.1.9 Mit Konstanten in SCL rechnen....7627

16.2 Einstellungen zu SCL....7631
16.2.1 Übersicht über die Einstellungen zu SCL....7631
16.2.2 Einstellungen ändern....7632

16.3 Das Programmierfenster von SCL....7633
16.3.1 Übersicht über das Programmierfenster....7633
16.3.2 Anpassen des Programmierfensters....7634
16.3.3 SCL-Code formatieren....7635
16.3.4 Codeabschnitte auf- und zuklappen....7637
16.3.5 Zu Definitionen navigieren....7637
16.3.6 Lesezeichen verwenden....7638
16.3.6.1 Grundlagen zu Lesezeichen....7638
16.3.6.2 Lesezeichen setzen....7639
16.3.6.3 Zwischen Lesezeichen navigieren....7639
16.3.6.4 Lesezeichen löschen....7640

16.4 SCL-Anweisungen eingeben....7642
16.4.1 Regeln zu SCL-Anweisungen....7642
16.4.2 SCL-Anweisungen manuell eingeben....7643
16.4.3 SCL-Anweisungen über Task Card "Anweisungen" einfügen....7644
16.4.4 Programmcode mit Strukturelementen umschließen....7645
16.4.5 Arbeiten mit Bereichen....7646
16.4.5.1 Bereiche verwenden....7646
16.4.5.2 Bereiche einfügen....7648
16.4.5.3 Bereiche kopieren und einfügen....7650
16.4.5.4 In Bereichen navigieren....7651
16.4.5.5 Bereiche löschen....7653
16.4.6 Datentyp einer SCL-Anweisung festlegen....7654
16.4.6.1 Grundlagen zu den Datentypen von SCL-Anweisungen....7654
16.4.6.2 Datentyp einer SCL-Anweisung ändern....7655

16.4.6.3 Datentyp von IEC-Zeiten und IEC-Zählern ändern....7655
16.4.6.4 Beispiel zum Ändern des Datentyps einer SCL-Anweisung....7656
16.4.7 Variableninformationen ein- oder ausblenden....7656
16.4.8 Favoriten in SCL verwenden....7657
16.4.8.1 SCL-Anweisungen zu den Favoriten hinzufügen....7657
16.4.8.2 SCL-Anweisungen mithilfe von Favoriten einfügen....7658
16.4.8.3 SCL-Anweisungen aus den Favoriten entfernen....7659
16.4.9 Bausteinaufrufe in SCL einfügen....7660
16.4.9.1 Grundlagen zum Bausteinaufruf in SCL....7660
16.4.9.2 Bausteinaufrufe manuell einfügen....7665
16.4.9.3 Bausteinaufrufe per Drag & Drop einfügen....7667
16.4.9.4 Bausteinaufrufe aktualisieren....7669
16.4.10 Parameterliste erweitern und reduzieren....7670
16.4.11 Pragmas einfügen....7671
16.4.12 Kommentare einfügen....7672

16.5 SCL-Anweisungen bearbeiten....7674
16.5.1 Anweisungen selektieren....7674
16.5.2 Anweisungen kopieren, ausschneiden und einfügen....7674
16.5.3 Anweisungen löschen....7675

16.6 Programmierbeispiele SCL....7676
16.6.1 Beispiel für das Steuern eines Förderbands....7676
16.6.2 Beispiel für das Erfassen der Richtung eines Förderbands....7678
16.6.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs 7680

17 GRAPH-Programme erstellen....7685

17.1 Grundlagen zu GRAPH....7685
17.1.1 Programmiersprache GRAPH....7685
17.1.2 Ablaufketten....7687
17.1.2.1 Grundlagen zu Ablaufketten....7687
17.1.2.2 Elemente einer Ablaufkette....7688
17.1.3 Schritte und Transitionen....7692
17.1.3.1 Grundlagen zu Schritten....7692
17.1.3.2 Elemente eines Schritts....7693
17.1.3.3 Grundlagen zu Transitionen....7694
17.1.4 Aktionen....7695
17.1.4.1 Grundlagen zu Aktionen....7695
17.1.4.2 Standardaktionen....7696
17.1.4.3 Ereignisabhängige Aktionen....7698
17.1.4.4 Bausteinaufrufe....7701
17.1.4.5 Zeiten....7702
17.1.4.6 Zähler....7704
17.1.5 Permanente Anweisungen....7705
17.1.6 Bedingungen....7706
17.1.7 Interlock- und Supervisionsmeldungen....7706
17.1.8 Bausteinschnittstelle eines GRAPH-Funktionsbausteins....7707
17.1.8.1 Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins....7707
17.1.8.2 Eingangsparameter....7710
17.1.8.3 Ausgangsparameter....7714
17.1.8.4 Statische Parameter....7718
17.1.8.5 Standardverhalten von GRAPH erweitern....7726

17.2 Einstellungen zu GRAPH....7729

17.2.1 Übersicht über die Einstellungen zu GRAPH....7729
17.2.2 Einstellungen ändern....7731
17.2.3 GRAPH-Versionen....7732
17.2.3.1 Übersicht über die GRAPH-Versionen....7732
17.2.3.2 Version des GRAPH-Funktionsbausteins festlegen....7732
17.2.4 Programmiersprache in Netzwerken festlegen....7733
17.2.5 Schnittstellenparametersatz festlegen....7734
17.2.6 Speicherplatzmodell des Datenbausteins festlegen....7736
17.2.7 Einstellungen für interne Parameter ändern....7737

17.3 Das Programmierfenster von GRAPH....7739
17.3.1 Übersicht über das Programmierfenster....7739
17.3.2 Ansicht "Vorausgeschaltete permanente Anweisungen"....7741
17.3.3 Kettenansicht....7742
17.3.4 Einzelschrittansicht....7744
17.3.5 Ansicht "Nachgeschaltete permanente Anweisungen"....7746
17.3.6 Meldungsansicht....7747

17.4 Favoriten in GRAPH verwenden....7749
17.4.1 GRAPH-Elemente zu den Favoriten hinzufügen....7749
17.4.2 GRAPH-Elemente mithilfe von Favoriten einfügen....7750
17.4.3 GRAPH-Elemente aus den Favoriten entfernen....7751

17.5 Struktur einer Ablaufkette programmieren....7752
17.5.1 Ablaufketten einfügen....7752
17.5.2 Ablaufketten umbenennen....7752
17.5.3 Ablaufketten kommentieren....7753
17.5.4 Elemente in eine Ablaufkette einfügen....7754
17.5.5 Ablaufketten beenden....7755
17.5.6 Ablaufketten löschen....7756
17.5.7 Schritte und Transition verwenden....7756
17.5.7.1 Schritte und Transitionen einfügen....7756
17.5.7.2 Initialschritt festlegen und abwählen....7758
17.5.7.3 Schritte und Transitionen neu nummerieren....7759
17.5.7.4 Schritte und Transitionen umbenennen....7761
17.5.7.5 Schritte kommentieren....7762
17.5.7.6 Schritte und Transitionen kopieren und einfügen....7763
17.5.7.7 Schritte überspringen....7764
17.5.7.8 Schritte und Transitionen löschen....7765
17.5.7.9 Übersetzungen der Schritt- und Transitionsnamen anzeigen und editieren....7765
17.5.8 Verzweigungen verwenden....7767
17.5.8.1 Alternativzweige erstellen....7767
17.5.8.2 Simultanzweig erstellen....7767
17.5.8.3 Verzweigungen schließen....7768
17.5.8.4 Verzweigungen löschen....7769
17.5.9 Sprünge verwenden....7770
17.5.9.1 Sprünge einfügen....7770
17.5.9.2 Sprungziel ändern....7770
17.5.9.3 Zum Sprungziel navigieren....7771
17.5.9.4 Sprünge löschen....7772

17.6 Aktionen programmieren....7773
17.6.1 Aktionen einfügen....7773
17.6.2 Datentyp einer GRAPH-Anweisung festlegen....7776

17.6.2.1 Grundlagen zu den Datentypen von GRAPH-Anweisungen....7776
17.6.2.2 Datentyp einer GRAPH-Anweisung ändern....7777
17.6.3 Aktionen ändern....7778
17.6.4 Aktionen kopieren, ausschneiden und einfügen....7779
17.6.5 Aktionen verschieben....7780
17.6.6 Aktionen löschen....7781

17.7 Bedingungen programmieren....7782
17.7.1 Interlocks programmieren....7782
17.7.2 Supervisionen programmieren....7783
17.7.2.1 Supervisionsbedingungen einfügen....7783
17.7.2.2 Schrittzeit-Überwachung verwenden....7784
17.7.3 Transitionen programmieren....7789
17.7.4 Permanente Anweisungen programmieren....7790
17.7.4.1 Arbeiten mit Netzwerken....7790
17.7.4.2 Permanente Anweisungen einfügen....7796
17.7.5 Variableninformationen ein- oder ausblenden....7798

17.8 Bausteinaufrufe in GRAPH einfügen....7800
17.8.1 Bausteinaufrufe manuell einfügen....7800
17.8.2 Bausteinaufrufe per Drag & Drop einfügen....7801
17.8.3 Bausteinaufrufe in GRAPH aktualisieren....7803
17.8.4 Bausteinaufruf ändern....7804

17.9 Meldungen festlegen....7806
17.9.1 Meldungsanzeige aktivieren und deaktivieren....7806
17.9.2 Meldungen bearbeiten....7806

17.10 Erstwerte erfassen....7809
17.10.1 Übersicht über die Erstwerterfassung....7809
17.10.2 Erstwerterfassung aktivieren....7810
17.10.3 Erstwerte zurücksetzen....7811
17.10.4 Beispiel für die Reihenfolge der Auflistung....7812

18 Meldungen projektieren....7815

18.1 Einführung in die Meldungsprojektierung....7815

18.2 Vergabe der Meldungsnummern....7816

18.3 Bestandteile einer Meldung....7817

18.4 Verfügbare Meldebausteine....7818

18.5 Meldungstyp und Meldungen....7820

18.6 Formalparameter, Meldungsdatentypen und Meldebausteine....7822

18.7 Aufbau des Meldungseditors....7823

18.8 Arbeiten mit dem Meldungseditor....7825

18.9 Meldungen anlegen und bearbeiten....7826
18.9.1 Programmmeldungen anlegen....7826
18.9.2 Programmmeldungen im Meldungseditor bearbeiten....7826
18.9.3 Programmmeldungen im Programmiereditor bearbeiten....7827
18.9.4 Programmmeldungen löschen....7827
18.9.5 Meldungstyp bearbeiten....7828
18.9.6 Instanz-DB anlegen und Meldungsinstanzen bearbeiten....7828

18.9.7 Anwenderdiagnosemeldungen anlegen....7829
18.9.8 Anwenderdiagnosemeldungen bearbeiten....7830
18.9.9 Anwenderdiagnosemeldungen löschen....7830
18.9.10 Begleitwerte in Meldungen einfügen....7831
18.9.11 Aufbau von Begleitwerten....7831
18.9.12 Beispiele für Begleitwerte....7833
18.9.13 Begleitwerte löschen....7834

18.10 Meldungen anlegen und bearbeiten....7835
18.10.1 Programmmeldungen anlegen....7835
18.10.2 Programmmeldungen im Meldungseditor bearbeiten....7836
18.10.3 Programmmeldungen im Programmiereditor bearbeiten....7836
18.10.4 Programmmeldungen löschen....7837
18.10.5 Meldungstyp bearbeiten....7837
18.10.6 Instanz-DB anlegen und Meldungsinstanzen bearbeiten....7838
18.10.7 Begleitwerte in Meldungen einfügen....7838
18.10.8 Einstellungen beim Eingeben von Begleitwerten....7840
18.10.9 Besonderheiten bei Begleitwerten....7841
18.10.10 Beispiele für Begleitwerte....7842
18.10.11 Begleitwerte löschen....7843

18.11 Texte und Attribute....7844
18.11.1 Schlüsselwörter für Meldetexte....7844
18.11.2 Texte eingeben....7845
18.11.3 Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen....7845
18.11.4 Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden....7846
18.11.5 Texte verriegeln....7847
18.11.6 Attribute verriegeln....7848

18.12 Textlisten für Meldungen....7850
18.12.1 Grundlagen zu Textlisten für Meldungen....7850
18.12.2 Textlisten für Meldungen bearbeiten....7850
18.12.3 Texte aus Textlisten in Meldungen integrieren....7851
18.12.4 Beispiel für das Integrieren von Texten aus Textlisten in Meldungen....7852

18.13 Meldeklassen....7853
18.13.1 Meldeklassen anlegen....7853
18.13.2 Meldeklassen bearbeiten....7854
18.13.3 Meldeklassen löschen....7854
18.13.4 Priorität der Meldeklasse....7855
18.13.5 Meldeklassen exportieren und importieren....7856
18.13.5.1 Grundlagen zum Exportieren und Importieren von Meldeklassen....7856
18.13.5.2 Meldeklassen exportieren....7856
18.13.5.3 Meldeklassen importieren....7857

18.14 Meldetexte exportieren und importieren....7858
18.14.1 Grundlagen zum Exportieren und Importieren von Meldetexten....7858
18.14.2 Meldetexte exportieren....7859
18.14.3 Meldetexte importieren....7860

19 PLC-Programme übersetzen und laden....7863

19.1 Grundlagen zum Übersetzen und Laden von PLC-Programmen....7863

19.2 Bausteine übersetzen....7865
19.2.1 Grundlagen zum Übersetzen von Bausteinen....7865

19.2.2 Bausteine in der Projektnavigation übersetzen....7867
19.2.3 Bausteine im Programmiereditor übersetzen....7869
19.2.4 Übersetzungsfehler beheben....7870

19.3 Bausteine laden für S7-1200/1500....7871
19.3.1 Einführung zum Laden von Bausteinen....7871
19.3.2 Bausteine im Betriebszustand "RUN" in Gerät laden....7875
19.3.3 Bausteine vom Programmiereditor in Gerät laden....7879
19.3.4 Bausteine aus der Projektnavigation in Gerät laden....7881
19.3.5 Bausteine von Gerät laden....7884
19.3.6 Bausteine auf eine Memory Card laden....7886
19.3.7 Bausteine von einer Memory Card laden....7887
19.3.8 Ablaufkette vor dem Laden eines GRAPH-DB ausschalten....7888
19.3.9 Bausteinerweiterungen ohne Reinitialisierung laden....7889
19.3.9.1 Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung....7889
19.3.9.2 Speicherreserve für Bausteinerweiterungen einstellen....7891
19.3.9.3 Speicherreserve aktivieren....7892
19.3.9.4 Speicherreserve deaktivieren....7893
19.3.9.5 Bausteinschnittstelle oder Datenbaustein erweitern....7894
19.3.9.6 Speicherreserve zurücksetzen....7895

19.4 Bausteine laden für S7-300/400....7897
19.4.1 Einführung zum Laden von Bausteinen....7897
19.4.2 Bausteine im Betriebszustand "RUN" in Gerät laden....7900
19.4.3 Bausteine vom Programmiereditor in Gerät laden....7904
19.4.4 Bausteine aus der Projektnavigation in Gerät laden....7906
19.4.5 Bausteine von Gerät laden....7909
19.4.6 Bausteine auf eine Memory Card laden....7910
19.4.7 Bausteine von einer Memory Card laden....7911
19.4.8 Ablaufkette vor dem Laden eines GRAPH-DB ausschalten....7912

19.5 PLC-Variablen laden....7914
19.5.1 Einführung zum Laden von PLC-Variablen....7914
19.5.2 Online/Offline-Sichtbarkeit von PLC-Variablen....7916

20 PLC-Programme vergleichen....7917

20.1 Grundlagen zum Vergleichen von PLC-Programmen....7917
20.1.1 Einführung zum Vergleichen von PLC-Programmen....7917
20.1.2 Vergleich von Bausteinen und PLC-Datentypen....7919
20.1.3 Vergleich von PLC-Variablen....7920
20.1.4 Vergleich von PLC-Programmen anhand von Prüfsummen....7921

20.2 Bausteine vergleichen....7923
20.2.1 Bausteine im Vergleichseditor vergleichen....7923
20.2.2 Detailvergleich von Bausteinen durchführen....7925
20.2.2.1 Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten....7925
20.2.2.2 Detailvergleich für GRAPH-Bausteine starten....7927
20.2.2.3 Darstellung des Vergleichsergebnisses....7930
20.2.2.4 Im Detailvergleich navigieren....7946
20.2.2.5 Bausteine während des Detailvergleichs ändern....7948
20.2.2.6 Vergleichsergebnisse aktualisieren....7949

20.3 PLC-Variablen vergleichen....7951

20.4 PLC-Datentypen vergleichen....7955

21 Programminformationen anzeigen....7959

21.1 Übersicht der möglichen Programminformationen....7959

21.2 Belegungsplan anzeigen....7960
21.2.1 Einführung zum Belegungsplan....7960
21.2.2 Aufbau des Belegungsplans....7961
21.2.3 Symbole im Belegungsplan....7962
21.2.4 Belegungsplan anzeigen....7963
21.2.5 Ansichtsoptionen für Belegungsplan einstellen....7964
21.2.6 Filtermöglichkeiten im Belegungsplan....7964
21.2.7 Filter für Belegungsplan definieren....7965
21.2.8 Belegungsplan filtern....7966
21.2.9 Remanente Speicherbereiche für Merker definieren....7967
21.2.10 Anzeige der Remanenz für Merker aktivieren....7968

21.3 Aufrufstruktur anzeigen....7969
21.3.1 Einführung zur Aufrufstruktur....7969
21.3.2 Symbole in der Aufrufstruktur....7971
21.3.3 Aufbau der Aufrufstruktur....7972
21.3.4 Aufrufstruktur anzeigen....7973
21.3.5 Ansichtsoptionen für die Aufrufstruktur einstellen....7974
21.3.6 Einführung zur Konsistenzprüfung in der Aufrufstruktur....7975
21.3.7 Bausteinkonsistenz in der Aufrufstruktur prüfen....7975

21.4 Abhängigkeitsstruktur anzeigen....7977
21.4.1 Einführung zur Abhängigkeitsstruktur....7977
21.4.2 Aufbau der Abhängigkeitsstruktur....7978
21.4.3 Symbole in der Abhängigkeitsstruktur....7979
21.4.4 Abhängigkeitsstruktur anzeigen....7979
21.4.5 Ansichtsoptionen für die Abhängigkeitsstruktur einstellen....7980
21.4.6 Einführung zur Konsistenzprüfung in der Abhängigkeitsstruktur....7981
21.4.7 Bausteinkonsistenz in der Abhängigkeitsstruktur prüfen....7982

21.5 Speicherauslastung der CPU anzeigen....7983
21.5.1 Einführung zur Speicherauslastung....7983
21.5.2 Aufbau des Registers "Speicherauslastung"....7985
21.5.3 Speicherauslastung anzeigen....7986
21.5.4 Maximal verfügbaren Ladespeicher auswählen....7987

22 Querverweise anzeigen....7989

22.1 Allgemeine Informationen zu Querverweisen....7989

22.2 Aufbau der Querverweisliste....7991

22.3 Einstellungen in der Querverweisliste....7994

22.4 Querverweisliste anzeigen....7997

22.5 Querverweisliste sortieren....7999

22.6 Filtermöglichlichkeiten für Querverweise....8001
22.6.1 Einführung zum Filtern von Querverweisen....8001
22.6.2 Aufbau des Filterdialogs....8003
22.6.3 Attribute, Operatoren und Werte im Filterdialog....8004
22.6.4 Anwenderdefinierte Filter für Querverweise erstellen....8006
22.6.5 Anwenderdefinierte Filter bearbeiten....8008

22.6.6 Anwenderdefinierte Filter löschen....8010
22.6.7 Anwenderdefinierte Filter exportieren und importieren....8011
22.6.8 Filter als Voreinstellung setzen....8012
22.6.9 Filter duplizieren....8014
22.6.10 Querverweisliste filtern....8014

22.7 Querverweisliste drucken....8016

22.8 Neues Objekt in der Querverweisliste hinzufügen....8017

22.9 Überlappende Zugriffe in der Querverweisliste anzeigen....8018

22.10 Querverweise nach Projektupgrade wieder herstellen....8021

22.11 Querverweise im Inspektorfenster anzeigen und bearbeiten....8022

23 Anwenderprogramm testen....8025

23.1 Grundlagen zum Testen des Anwenderprogramms....8025

23.2 Testen mit Programmstatus....8027
23.2.1 Einführung zum Testen mit Programmstatus....8027
23.2.2 Aufrufumgebung einstellen....8029
23.2.2.1 Grundlagen zur Aufrufumgebung....8029
23.2.2.2 Aufrufumgebung zum Beobachten von Bausteinen einstellen....8030
23.2.3 Test mit Programmstatus ein-/ausschalten....8031
23.2.4 Schleifen beobachten....8034
23.2.5 Strukturen beobachten....8036
23.2.6 Bausteine während des Programmtests bearbeiten....8038
23.2.7 Variablen während des Beobachtens steuern....8039
23.2.8 Anzeigeformate im Programmstatus umstellen....8040
23.2.9 Besonderheiten beim Testen von GRAPH-Programmen....8041
23.2.9.1 Grundlagen zum Testen von GRAPH-Programmen....8041
23.2.9.2 Testeinstellungen festlegen....8043
23.2.9.3 Ablaufkette beim Testen steuern....8045
23.2.10 Beispiele für die Programmstatusanzeige....8050
23.2.10.1 Programmstatusanzeige für KOP-Programme....8050
23.2.10.2 Programmstatusanzeige für FUP-Programme....8051
23.2.10.3 Programmstatusanzeige für AWL-Programme....8052
23.2.10.4 Programmstatusanzeige für SCL-Programme....8053
23.2.10.5 Programmstatusanzeige für GRAPH-Programme....8054

23.3 Testen mit Haltepunkten....8058
23.3.1 Einführung zum Testen mit Haltepunkten....8058
23.3.2 Security-Hinweise zum Testen mit Haltepunkten....8060
23.3.3 Sicherheitshinweise beim Testen mit Haltepunkten....8060
23.3.4 Regeln zum Testen mit Haltepunkten....8063
23.3.5 Statusanzeige für Haltepunkte....8064
23.3.6 Aufrufumgebung für Haltepunkte einstellen....8065
23.3.7 Haltepunkte setzen....8067
23.3.8 Haltepunkte aktivieren....8068
23.3.9 Haltepunkte deaktivieren....8071
23.3.10 Zu den Haltepunkten navigieren....8073
23.3.11 Haltepunkte löschen....8074
23.3.12 Testen mit Haltepunkten im Einzelschrittmodus durchführen....8076
23.3.13 Haltepunkte "Ausführen bis Markierung"....8078
23.3.14 Variablen bei erreichtem Haltepunkt beobachten....8079

23.3.15 Variablen bei erreichtem Haltepunkt steuern....8081
23.3.16 Programmausführung fortsetzen....8083

23.4 Testen mit der Beobachtungstabelle....8084
23.4.1 Einführung zum Testen mit der Beobachtungstabelle....8084
23.4.2 Aufbau der Beobachtungstabelle....8085
23.4.3 Basismodus und Erweiterter Modus in der Beobachtungstabelle....8086
23.4.4 Symbole in der Beobachtungstabelle....8087
23.4.5 Beobachtungstabellen erstellen und bearbeiten....8089
23.4.5.1 Beobachtungstabelle erstellen....8089
23.4.5.2 Beobachtungstabelle öffnen....8090
23.4.5.3 Beobachtungstabelle kopieren und einfügen....8090
23.4.5.4 Beobachtungstabelle speichern....8091
23.4.6 Variablen in die Beobachtungstabelle eingeben....8091
23.4.6.1 Grundlagen für die Eingabe von Variablen in die Beobachtungstabelle....8091
23.4.6.2 Erlaubte Operanden für die Beobachtungstabelle....8092
23.4.6.3 Erlaubte Steuerwerte für die Beobachtungstabelle....8094
23.4.6.4 Übersicht über die Anzeigeformate....8096
23.4.6.5 Anzeigeformat für Variablen auswählen....8098
23.4.6.6 Kommentarzeilen erstellen und bearbeiten....8099
23.4.7 Variablen in der Beobachtungstabelle beobachten....8100
23.4.7.1 Einführung zum Beobachten von Variablen in der Beobachtungstabelle....8100
23.4.7.2 Festlegen des Beobachtungs- und Steuermodus....8101
23.4.7.3 Variablen "Alle beobachten"....8104
23.4.7.4 Variablen "Sofort beobachten"....8105
23.4.8 Variablen in der Beobachtungstabelle steuern....8106
23.4.8.1 Einführung zum Steuern von Variablen....8106
23.4.8.2 Variablen steuern auf "0"....8107
23.4.8.3 Variablen steuern auf "1"....8109
23.4.8.4 Variablen "Sofort steuern"....8110
23.4.8.5 Variablen "Steuern mit Trigger"....8111
23.4.8.6 Peripherieausgänge freischalten....8113

23.5 Testen mit der Forcetabelle....8115
23.5.1 Einführung zum Testen mit der Forcetabelle....8115
23.5.2 Sicherheitsvorkehrungen beim Forcen von Variablen....8116
23.5.3 Aufbau der Forcetabelle....8117
23.5.4 Basismodus und Erweiterter Modus in der Forcetabelle....8118
23.5.5 Symbole in der Forcetabelle....8118
23.5.6 Forcetabelle öffnen und bearbeiten....8120
23.5.6.1 Forcetabelle anzeigen....8120
23.5.6.2 Forcetabelle öffnen....8120
23.5.6.3 Forcetabelle speichern....8120
23.5.7 Variablen in die Forcetabelle eingeben....8121
23.5.7.1 Grundlagen für die Eingabe von Variablen in die Forcetabelle....8121
23.5.7.2 Erlaubte Operanden für die Forcetabelle....8122
23.5.7.3 Erlaubte Forcewerte für die Forcetabelle....8123
23.5.7.4 Übersicht über die Anzeigeformate....8124
23.5.7.5 Anzeigeformat für Variablen auswählen....8127
23.5.7.6 Kommentarzeilen erstellen und bearbeiten....8127
23.5.8 Variablen in der Forcetabelle beobachten....8129
23.5.8.1 Einführung zum Beobachten von Variablen in der Forcetabelle....8129
23.5.8.2 Festlegen des Beobachtungsmodus in der Forcetabelle....8130

23.5.8.3 Variablen "Alle beobachten"....8131
23.5.8.4 Variablen "Sofort beobachten"....8132
23.5.9 Variablen in der Forcetabelle forcen....8132
23.5.9.1 Einführung zum Forcen von Variablen....8132
23.5.9.2 Sicherheitsvorkehrungen beim Forcen von Variablen....8134
23.5.9.3 Geforcte Operanden aktualisieren....8135
23.5.9.4 Variablen forcen auf "0"....8137
23.5.9.5 Variablen forcen auf "1"....8138
23.5.9.6 Variablen forcen "Alle forcen"....8140
23.5.10 Forcen von Variablen beenden....8141
23.5.10.1 Forcen aller Variablen beenden....8141
23.5.10.2 Forcen einzelner Variablen beenden....8143

24 Maschinen und Anlagen mit ProDiag überwachen....8145

24.1 Systemvoraussetzungen für ProDiag....8145
24.1.1 Hinweise zu Lizenzen....8145
24.1.2 ProDiag-Lizenzen einstellen....8145
24.1.3 Software- und Hardwarevoraussetzungen....8146

24.2 Technische Daten für ProDiag....8147
24.2.1 Technische Daten für ProDiag....8147

24.3 Grundlagen der Überwachung mit ProDiag....8148
24.3.1 Einführung in ProDiag....8148
24.3.2 Beispiel für die Erstellung einer Überwachung anhand einer Operandenüberwachung....8151
24.3.3 Überwachungsarten....8154
24.3.3.1 Übersicht über die Überwachungsarten....8154
24.3.3.2 Operandenüberwachung....8155
24.3.3.3 Verriegelungsüberwachung....8157
24.3.3.4 Aktionsüberwachung....8159
24.3.3.5 Reaktionsüberwachung....8161
24.3.3.6 Positionsüberwachung....8163
24.3.3.7 Textmeldung....8165
24.3.3.8 Fehlermeldung....8167
24.3.4 Beispiele für die einzelnen Überwachungsarten....8169
24.3.4.1 Einführung....8169
24.3.4.2 Beispiel für eine Operandenüberwachung....8170
24.3.4.3 Beispiel für eine Verriegelungsüberwachung....8172
24.3.4.4 Beispiel für eine Aktionsüberwachung....8176
24.3.4.5 Beispiel für eine Reaktionsüberwachung....8179
24.3.4.6 Beispiel für eine Positionsüberwachung....8182
24.3.4.7 Beispiel für eine Textmeldung....8185
24.3.4.8 Beispiel für eine Fehlermeldung....8187
24.3.5 Anwendungsbeispiel für die Überwachung mit ProDiag....8189

24.4 ProDiag-Funktionsbausteine anlegen....8190
24.4.1 Überblick über die ProDiag-Funktionsbausteine....8190
24.4.2 Aufbau eines ProDiag-Funktionsbausteins....8191
24.4.3 Anordnung der Spalten im ProDiag-FB sichern....8191
24.4.4 ProDiag-Funktionsbausteine anlegen....8192
24.4.4.1 ProDiag-FBs beim Erstellen globaler Überwachungen anlegen....8192
24.4.4.2 ProDiag-FBs beim Bausteinaufruf anlegen....8193
24.4.4.3 ProDiag-FBs am Instanzdatenbaustein eines FBs anlegen....8194
24.4.4.4 ProDiag-FBs mithilfe des Dialogs Neuen Baustein hinzufügen anlegen....8195

24.4.4.5 ProDiag-FBs in den ProDiag-Übersichtstabellen anlegen....8196

24.5 Zentralen Zeitstempel verwenden....8197
24.5.1 Zentralen Zeitstempel verwenden....8197
24.5.2 Mehrere Überwachungsmeldungen eines ProDiag-FBs innerhalb einer Zykluszeit....8199

24.6 ProDiag-Instanzdatenbausteine verwenden....8200
24.6.1 Aufbau eines ProDiag-Instanzdatenbausteins der Version 1.0....8200
24.6.2 Inhalt eines ProDiag-Instanzdatenbausteins der Version 1.0....8201
24.6.3 Aufbau eines ProDiag-Instanzdatenbausteins der Version 2.0....8207
24.6.4 Inhalt eines ProDiag-Instanzdatenbausteins der Version 2.0....8208

24.7 Eigenschaften eines ProDiag-Funktionsbausteins definieren....8215
24.7.1 Aufbau der Überwachungseinstellungen am ProDiag-FB....8215
24.7.2 Freigabekriterien für Überwachungsmeldungen anlegen....8216
24.7.2.1 Überblick über die Freigabekriterien....8216
24.7.2.2 Globales Freigabekriterium verwenden....8218
24.7.2.3 Freigabekriterium der Kategorie verwenden....8218
24.7.3 Quittiervariable für einzelne Kategorien definieren....8219
24.7.3.1 Überblick über die Quittiervariablen....8219
24.7.3.2 Quittiervariable verwenden....8220
24.7.4 Anzeigeklassen einstellen....8220
24.7.4.1 Überblick über die Anzeigeklassen....8220
24.7.4.2 Anzeigeklasse einstellen....8220
24.7.5 Anzahl der Überwachungen eines ProDiag-FBs anzeigen....8221

24.8 ProDiag-Überwachungseinstellungen festlegen....8222
24.8.1 Überblick über die ProDiag-Überwachungseinstellungen....8222
24.8.2 Allgemeine ProDiag-Überwachungseinstellungen festlegen....8223
24.8.2.1 Aufbau der ProDiag-Überwachungseinstellungen....8223
24.8.2.2 Kategorien....8224
24.8.2.3 Unterkategorien....8227
24.8.2.4 Überwachungsarten....8230
24.8.3 Zentrale ProDiag-Meldetexte festlegen....8233
24.8.3.1 Aufbau des Meldetexteditors....8233
24.8.3.2 Aufbau einer GRAPH-Überwachungsmeldung....8236
24.8.3.3 Aufbau einer globalen Überwachungsmeldung (Variable)....8237
24.8.3.4 Aufbau einer lokalen Überwachungsmeldung (Parameter)....8238
24.8.3.5 Meldetexte anlegen....8240
24.8.3.6 Export von ProDiag- und GRAPH-Texten....8241
24.8.4 ProDiag-Überwachungseinstellungen exportieren....8242
24.8.5 ProDiag-Überwachungseinstellungen importieren....8243

24.9 ProDiag-Überwachungen anlegen....8245
24.9.1 Überblick über die Überwachungen....8245
24.9.2 Verwendungsmöglichkeiten der Überwachungen....8249
24.9.3 Globale Überwachung anlegen....8250
24.9.4 Lokale Überwachung anlegen....8252
24.9.5 Spezifisches Textfeld anlegen....8255
24.9.5.1 Verwendung der Begleitwerte....8255
24.9.5.2 Spezifisches Textfeld anlegen....8257
24.9.5.3 Übersetzungen des spezifischen Textfelds anzeigen und editieren....8258
24.9.6 Überwachung löschen....8260

24.10 ProDiag-Funktionsbausteine kopieren....8261

24.10.1 Überblick über das Kopieren von ProDiag-Funktionsbausteinen....8261
24.10.2 ProDiag-Funktionsbaustein kopieren....8261
24.10.3 ProDiag-Funktionsbaustein einfügen....8262

24.11 ProDiag-Funktionsbausteine zuweisen....8263
24.11.1 Überblick über die Zuweisung zu einem ProDiag-FB....8263
24.11.2 Globale Überwachungen einem ProDiag-FB zuweisen....8263
24.11.3 Lokale Überwachungen einem ProDiag-FB zuweisen....8264
24.11.4 Instanz-Datenbaustein einem ProDiag-FB zuweisen (lokale Überwachungen)....8266
24.11.5 Bestehende Zuweisung eines ProDiag-FBs ändern....8267
24.11.6 Bestehende Zuweisung eines ProDiag-FBs entfernen....8269

24.12 ProDiag-Funktionsbausteine bearbeiten....8271
24.12.1 Neue Überwachungen im ProDiag-FB einfügen....8271
24.12.2 Überwachungen im ProDiag-FB kopieren....8272
24.12.3 Überwachungen im ProDiag-FB löschen....8272
24.12.4 Überwachungen im ProDiag-FB sortieren....8273

24.13 ProDiag-Funktionsbausteine aufrufen....8274
24.13.1 Überblick über den Aufruf der ProDiag-Funktionsbausteine....8274

24.14 ProDiag-Bausteine laden....8275
24.14.1 ProDiag-Bausteine laden....8275

24.15 ProDiag-Überwachungen anzeigen....8276
24.15.1 Überblick über die Anzeigemöglichkeiten....8276
24.15.2 Überwachungen im ProDiag-Übersichtseditor anzeigen....8277
24.15.2.1 Aufbau des ProDiag-Übersichtseditors....8277
24.15.2.2 Überwachungen aller globalen Variablen anzeigen....8277
24.15.2.3 Überwachungen aller Funktionsbausteine anzeigen....8278
24.15.2.4 Überwachungen aller aufgerufenen Funktionsbausteine anzeigen....8279
24.15.3 ProDiag-Übersichtseditor bearbeiten....8280
24.15.3.1 Neue Variablenüberwachungen im ProDiag-Übersichtseditor einfügen....8280
24.15.3.2 Variablenüberwachungen im ProDiag-Übersichtseditor kopieren....8280
24.15.3.3 Variablenüberwachungen im ProDiag-Übersichtseditor löschen....8281
24.15.3.4 Variablenüberwachungen im ProDiag-Übersichtseditor sortieren....8282
24.15.3.5 Überwachungsdefinitionen am Funktionsbaustein im ProDiag-Übersichtseditor editieren..8282
24.15.3.6 Überwachungsdefinitionen am Funktionsbaustein im ProDiag-Übersichtseditor löschen..8283
24.15.4 Überwachungen eines Funktionsbausteins anzeigen....8283
24.15.5 Überwachungen eines ProDiag-Funktionsbausteins anzeigen....8284

24.16 Überwachte anwenderdefinierte Bausteine anzeigen....8286
24.16.1 Überwachte anwenderdefinierte Bausteine anzeigen....8286

24.17 ProDiag-Überwachungen und Eigenschaften eines ProDiag-FB exportieren und importieren....8287
24.17.1 Überblick über das Exportieren und Importieren von Überwachungen und Eigenschaften eines ProDiag-FBs....8287
24.17.2 Überwachungen und Eigenschaften eines ProDiag-FB exportieren....8289
24.17.3 Überwachungen und Eigenschaften eines ProDiag-FB importieren....8291

24.18 Zuweisungen von ProDiag-FBs zu Instanz-DBs exportieren und importieren....8294
24.18.1 Überblick über das Exportieren und Importieren von Zuweisungen....8294
24.18.2 Zuweisungen exportieren....8295
24.18.3 Zuweisungen importieren....8295

24.19 ProDiag-Querverweise anzeigen....8297

24.19.1 Übersicht zu Querverweisen....8297
24.19.2 Querverweise anzeigen....8297

24.20 Erstwerte erfassen....8299
24.20.1 Übersicht über die Erstwerterfassung....8299
24.20.2 Erstwerterfassung aktivieren....8302
24.20.3 Beispiel für die Reihenfolge der Auflistung....8302

Index....8305

Grundlagen zur Programmierung

1

1.1 Betriebssystem und Anwenderprogramm

Das folgende Bild zeigt das Zusammenspiel zwischen Betriebssystem und Anwenderprogramm:

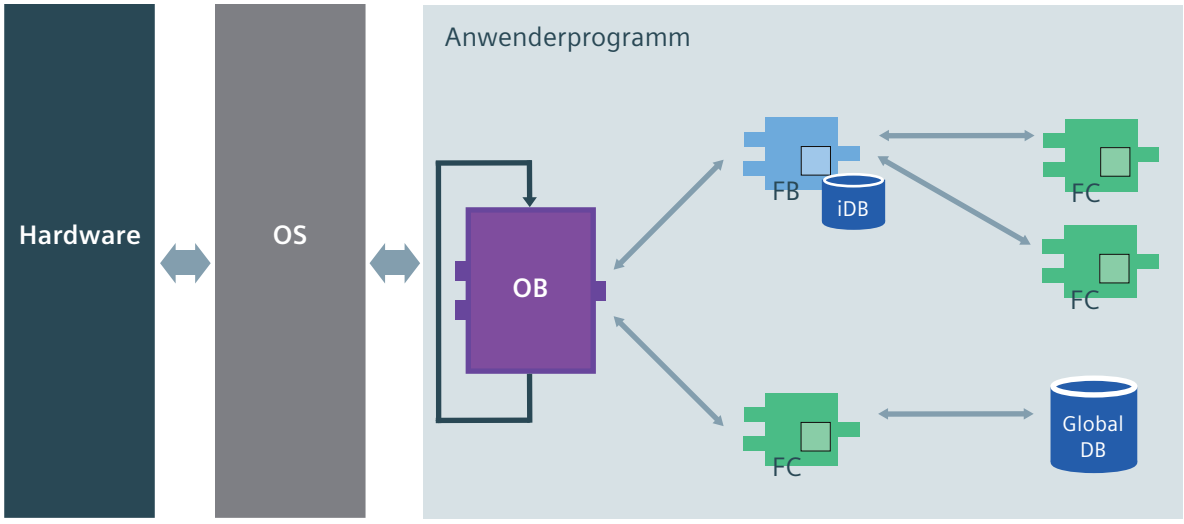

Betriebssystem (OS)

Das Betriebssystem ist in jeder CPU enthalten und organisiert alle Funktionen und Abläufe der CPU, die nicht mit einer spezifischen Steuerungsaufgabe verbunden sind.

Zu den Aufgaben des Betriebssystems gehören z. B.:

- Abwickeln von Neustart (Warmstart)
- Aktualisieren des Prozessabbilds der Eingänge und des Prozessabbilds der Ausgänge
- Aufrufen des Anwenderprogramms
- Erfassen von Alarmen und Aufrufen der Alarm-OBs
- Erkennen und Behandeln von Fehlern
- Verwalten von Speicherbereichen

Das Betriebssystem ist Bestandteil der CPU und ist bei der Auslieferung bereits auf dieser enthalten.

Anwenderprogramm

Das Anwenderprogramm enthält alle Funktionen, die zur Bearbeitung Ihrer spezifischen Automatisierungsaufgabe erforderlich sind.

Zu den Aufgaben des Anwenderprogramms gehören:

- Prüfung der Vorbedingungen für einen Neustart (Warmstart) mithilfe von Anlauf-OBs, z. B. Endschalter in richtiger Stellung oder Sicherheitseinrichtung aktiv
- Bearbeiten von Prozessdaten, z. B. Binärsignale verknüpfen, Analogwerte einlesen und auswerten, Binärsignale für die Ausgabe festlegen, Analogwerte ausgeben
- Reaktion auf Alarme, z. B. Diagnosealarm bei Unterschreitung des Grenzwerts einer analogen Erweiterungsbaugruppe
- Bearbeiten von Störungen im normalen Programmablauf

Das Anwenderprogramm wird von Ihnen erstellt und in die CPU geladen.

1.2 Bausteine im Anwenderprogramm

1.2.1 Lineare und strukturierte Programmierung

Lineare Programmierung

Kleine Automatisierungsaufgaben können Sie lösen, indem Sie das gesamte Anwenderprogramm linear in einen Zyklus-OB schreiben. Dies ist nur empfehlenswert bei einfachen Programmen.

Das folgende Bild zeigt schematisch ein lineares Programm: Der Zyklus-OB "Main" enthält das gesamte Anwenderprogramm.

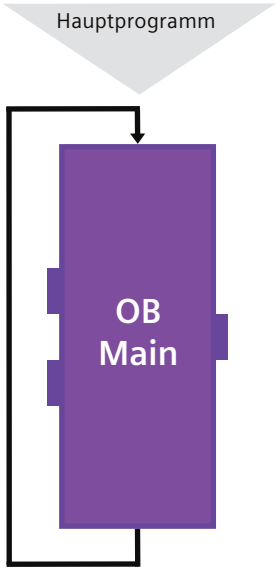

Strukturierte Programmierung

Komplexe Automatisierungsaufgaben können einfacher realisiert und gewartet werden, wenn sie in kleinere Teilaufgaben untergliedert werden, die den technologischen Funktionen des Automatisierungsprozesses entsprechen oder mehrfach verwendet werden können. Im Anwenderprogramm werden diese Teilaufgaben durch Bausteine repräsentiert. Dabei ist jeder Baustein ein unabhängiger Abschnitt des Anwenderprogramms.

Die Strukturierung des Programms bietet folgende Vorteile:

- Umfangreiche Programme lassen sich übersichtlich programmieren.
- Einzelne Programmteile können standardisiert und mehrfach mit wechselnden Parametern verwendet werden.
- Die Programmorganisation wird vereinfacht.
- Änderungen des Programms lassen sich leichter durchführen.

- Der Programmtest wird vereinfacht, weil er abschnittsweise erfolgen kann.
- Die Inbetriebnahme wird erleichtert.

Das folgende Bild zeigt schematisch ein strukturiertes Programm: Der Zyklus-OB "Main" ruft nacheinander Unterprogramme auf, die definierte Teilaufgaben ausführen.

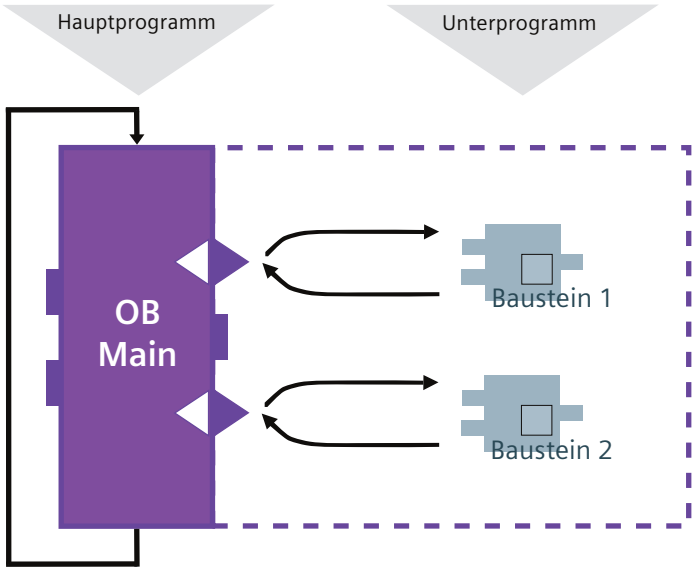

Schachtelungstiefe für Bausteine

Die zulässige Schachtelungstiefe für Bausteine ist von der eingesetzten CPU abhängig.

Die folgende Tabelle zeigt die Richtwerte für die maximale Schachtelungstiefe. Spezifische Angaben für die von Ihnen eingesetzte CPU finden Sie in den technischen Daten in der Hardware-Dokumentation. Um die Hardware-Dokumentation im Internet aufzurufen, klicken Sie auf die Links in der Tabelle.

CPU-Familie	Schachtelungstiefe (Richtwert)	Link auf Hardware-Dokumentation
S7-1500	24 Bausteine pro Prioritätsklasse	SIMATIC S7-1500 / ET 200MP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/86140384)
S7-1200	16 Bausteine aus Zyklus- oder Anlauf-OB 6 zusätzliche Bausteine innerhalb eines beliebigen Alarmereignis-OB	SIMATIC S7-1200 Automatisierungssystem (https://support.industry.siemens.com/cs/ww/de/view/91696622)
S7-400	24 Bausteine pro Prioritätsklasse 1-2 zusätzliche Bausteine innerhalb eines Fehler-OB	SIMATIC S7-400 Automatisierungssystem S7-400 CPU-Daten (https://support.industry.siemens.com/cs/de/de/view/53385241)
S7-300	16 Bausteine pro Prioritätsklasse 4 zusätzliche Bausteine innerhalb eines Fehler-OB	SIMATIC S7-300 CPU 31xC und CPU 31x: Technische Daten (https://support.industry.siemens.com/cs/de/de/view/12996906)
ET 200SP	24 Bausteine	SIMATIC ET 200SP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/84133942)

Schachtelungstiefe für Strukturen

Strukturen (STRUCT) und PLC-Datentypen (UDT) können bis zu einer Tiefe von 8 geschachtelt werden. Diese Schachtelungstiefe ist unabhängig von der eingesetzten CPU.

1.2.2 Übersicht über die Bausteinarten

Bausteinarten

Um die Aufgaben innerhalb eines Automatisierungssystems zu erledigen, gibt es unterschiedliche Bausteinarten. Die folgende Tabelle zeigt die Bausteinarten, die zur Verfügung stehen:

Bausteinart	Kurzbeschreibung
Organisationsbausteine (Seite 39) (OB)	Organisationsbausteine legen die Struktur des Anwenderprogramms fest.
Funktionen (Seite 40) (FC)	Funktionen enthalten Programmroutinen für wiederkehrende Aufgaben. Sie haben kein "Gedächtnis".
Funktionsbausteine (Seite 41) (FB)	Funktionsbausteine sind Codebausteine, die ihre Werte dauerhaft in Instanz-Datenbausteinen ablegen, so dass sie auch nach der Bausteinbearbeitung zur Verfügung stehen.
Instanz-Datenbausteine (Seite 53)	Instanz-Datenbausteine werden beim Aufruf eines Funktionsbausteins dem Baustein zugeordnet und dienen zur Speicherung der Programmdaten.
Globale Datenbausteine (Seite 42)	Globale Datenbausteine sind Datenbereiche zur Speicherung von Daten, die von beliebigen Bausteinen genutzt werden können.

1.2.3 Organisationsbausteine (OB)

Definition

Organisationsbausteine (OBs) bilden die Schnittstelle zwischen dem Betriebssystem und dem Anwenderprogramm. Sie werden vom Betriebssystem aufgerufen und steuern z. B. folgende Vorgänge:

- Anlaufverhalten des Automatisierungssystems
- Zyklische Programmbearbeitung
- Alarmgesteuerte Programmbearbeitung
- Behandlung von Fehlern

Sie können die Organisationsbausteine programmieren und so das Verhalten der CPU bestimmen. Abhängig von der verwendeten CPU stehen Ihnen unterschiedliche Organisationsbausteine zur Verfügung.

Weitere Informationen zu den Organisationsbausteinen finden Sie in den Beschreibungen der Arbeitsweisen der CPUs im Kapitel "Zusatzinformationen zu Konfigurationen" von "Geräte und Netze konfigurieren".

Startinformation von Organisationsbausteinen

Beim Start einiger Organisationsbausteine stellt das Betriebssystem Informationen bereit, die im Anwenderprogramm ausgewertet werden können. Ob und welche Informationen bereitgestellt werden, finden Sie in den Beschreibungen der Organisationsbausteine.

Siehe auch

Organisationsbausteine anlegen (Seite 7202)

1.2.4 Funktionen (FC)

Definition

Funktionen (FCs) sind Codebausteine ohne Gedächtnis. Sie haben keinen Datenspeicher, in denen Werte von Bausteinparametern gespeichert werden könnten. Deshalb müssen beim Aufruf einer Funktion alle Formalparameter mit Aktualparametern versorgt werden.
Um Daten dauerhaft zu speichern, stehen den Funktionen globale Datenbausteine zur Verfügung.

Anwendungsbereich

Eine Funktion enthält ein Programm, das immer dann ausgeführt wird, wenn die Funktion von einem anderen Codebaustein aufgerufen wird. Funktionen können z. B. zu folgenden Zwecken eingesetzt werden:

- Funktionswerte an den aufrufenden Baustein zurückgeben, z. B. bei mathematischen Funktionen
- Technologische Funktionen ausführen, z. B. Einzelsteuerungen mit Binärverknüpfungen

Eine Funktion kann auch mehrmals an verschiedenen Stellen innerhalb eines Programms aufgerufen werden. Sie erleichtern so die Programmierung häufig wiederkehrender Funktionen.

Hinweis

Parameterübergabe beim Aufruf von Funktionen

Um Fehler beim Arbeiten mit Funktionen zu vermeiden, beachten Sie das Kapitel "Parameterübergabe beim Bausteinaufruf (Seite 78)".

Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 7203)

1.2.5 Funktionsbausteine (FB)

Definition

Funktionsbausteine sind Codebausteine, die ihre Ein-, Aus- und Durchgangsparameter dauerhaft in Instanz-Datenbausteinen ablegen, sodass sie auch nach der Bausteinbearbeitung zur Verfügung stehen. Deshalb werden sie auch als "Bausteine mit Gedächtnis" bezeichnet.

Funktionsbausteine können auch mit temporären Variablen arbeiten. Temporäre Variablen werden jedoch nicht im Instanz-DB abgespeichert, sondern stehen nur einen Zyklus lang zur Verfügung.

Anwendungsbereich

Funktionsbausteine enthalten Unterprogramme, die immer dann ausgeführt werden, wenn ein Funktionsbaustein von einem anderen Codebaustein aufgerufen wird. Ein Funktionsbaustein kann auch mehrmals an verschiedenen Stellen innerhalb eines Programms aufgerufen werden. Sie erleichtern so die Programmierung häufig wiederkehrender Funktionen.

Instanzen von Funktionsbausteinen

Ein Aufruf eines Funktionsbausteins wird als Instanz bezeichnet. Für jede Instanz eines Funktionsbausteins wird ein Instanz-Datenbaustein benötigt, in dem für die im FB deklarierten Formalparameter instanzspezifische Werte gespeichert werden.

Der Funktionsbaustein kann seine instanzspezifischen Daten in einem eigenen Instanz-Datenbaustein ablegen oder im Instanz-Datenbaustein des aufrufenden Bausteins.

Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für die Instanz-Datenbausteine, die einem Funktionsbaustein beim Aufruf zugeordnet werden:

- Datenbausteine mit optimiertem Zugriff
  Datenbausteine mit optimiertem Zugriff haben keine fest definierte Speicherstruktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)
  Datenbausteine mit Standardzugriff haben eine feste Speicherstruktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

Hinweis

Um Fehler beim Arbeiten mit Funktionsbausteinen zu vermeiden, beachten Sie das Kapitel "Parameterübergabe beim Bausteinaufruf (Seite 78)".

Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 7203)

Multiinstanzen (Seite 67)

Instanz-Datenbausteine (Seite 53)

Grundlagen zum Bausteinzugriff (Seite 55)

1.2.6 Globale Datenbausteine (DB)

Definition

Datenbausteine dienen der Speicherung von Programmdaten. In Datenbausteinen stehen also variable Daten, mit denen das Anwenderprogramm arbeitet. Globale Datenbausteine nehmen Daten auf, die von allen anderen Bausteinen aus verwendet werden können.

Die maximale Größe von Datenbausteinen variiert abhängig von der CPU. Die Struktur globaler Datenbausteine können Sie beliebig festlegen.

Sie haben auch die Möglichkeit, PLC-Datentypen (UDT) als Vorlage für die Erstellung von globalen Datenbausteinen zu verwenden.

Globale Datenbausteine im Anwenderprogramm

Jeder Funktionsbaustein, jede Funktion oder jeder Organisationsbaustein kann die Daten aus einem globalen Datenbaustein lesen oder selbst Daten in einen globalen Datenbaustein schreiben. Diese Daten bleiben im Datenbaustein auch dann erhalten, wenn der Datenbaustein verlassen wird. Ein globaler Datenbaustein und ein Instanz-Datenbaustein können gleichzeitig geöffnet sein.

Das folgende Bild zeigt die verschiedenen Zugriffe auf Datenbausteine:

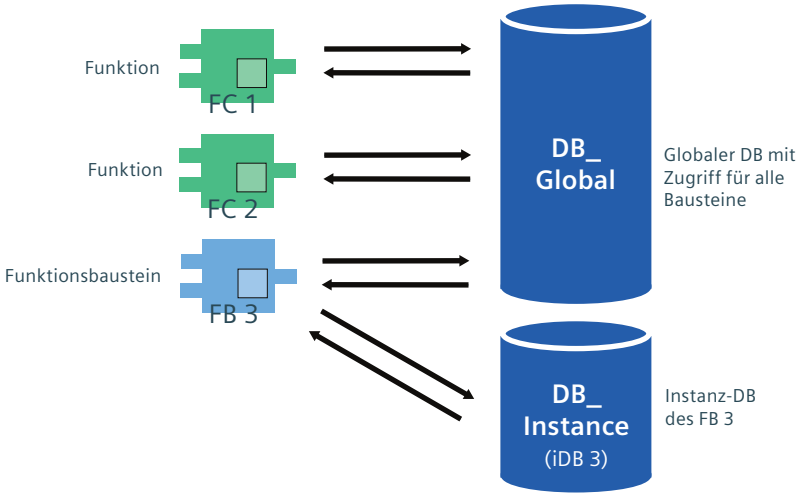

Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für globale Datenbausteine:

- Datenbausteine mit optimiertem Zugriff
  Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)
  Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

Siehe auch

Datenbausteine anlegen (Seite 7324)

Grundlagen zum Bausteinzugriff (Seite 55)

1.2.7 Globale ARRAY-Datenbausteine (DB)

ARRAY-Datenbausteine (S7-1500)

Eine besondere Art von globalen Datenbausteinen sind ARRAY-Datenbausteine. Diese bestehen aus einem ARRAY eines beliebigen Datentyps. Es kann z. B. ein ARRAY eines PLC-Datentyps (UDT) sein. Neben dem ARRAY enthält der DB keine weiteren Elemente. ARRAY-Datenbausteine erleichtern durch ihre flache Struktur den Zugriff auf die ARRAY-Elemente sowie deren Übergabe an aufgerufene Bausteine.

Für ARRAY-Datenbausteine ist das Attribut "Optimierter Bausteinzugriff" immer aktiviert. ARRAY-Datenbausteine mit Standardzugriff sind nicht möglich.

Erweiterte Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen > Einfache Anweisungen" im Abschnitt "Verschieben > ARRAY-DB". Diese Anweisungen bieten z. B. die Möglichkeit, den DB-Namen indirekt zu adressieren:

- ReadFromArrayDB: Aus Array-Datenbaustein lesen
- WriteToArrayDB: In Array-Datenbaustein schreiben
- ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen
- WriteToArrayDBL: In Array-Datenbaustein im Ladespeichern schreiben

Siehe auch

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

ARRAY-DBs adressieren (Seite 114)

Datenbausteine anlegen (Seite 7324)

1.2.8 Beispiel zur Verwendung von ARRAY-Datenbausteinen

Verwendung eines ARRAY-Datenbausteins (S7-1500)

ARRAY-Datenbausteine sind globale Datenbausteine, die ausschließlich aus einem ARRAY bestehen. Für die meisten Anwendungsfälle ist ein Datenbaustein mit einer Variable vom Datentyp ARRAY ausreichend, denn die Zugriffe lassen sich mit einer Variable vom Datentyp ARRAY intuitiv programmieren (z. B. #myArray[#index]) und bieten eine bessere Laufzeitperformance als ARRAY-Datenbausteine. In bestimmten Szenarien ist es jedoch erforderlich, ARRAYs mit unterschiedlichen Längen zu verarbeiten. Genau für diese Szenarien ist der ARRAY-Datenbaustein geeignet.

Das folgende Beispiel zeigt, wie Sie einen ARRAY-Datenbaustein einsetzen können.

Programmierbeispiel

Auf einem Förderband laufen einzelne Materialstücke. Diese Materialstücke passieren einen Scanner, der die Informationen auslesen kann, die ein Materialstück mit sich führt. Die Informationen werden ausgelesen und an ein Panel weitergegeben. Da der Scanner und das Panel verschiedene Taktzyklen/Geschwindigkeiten haben, muss die Materialinformation jeweils zwischengespeichert werden.

Im folgenden Programmierbeispiel zeigen wir Ihnen, wie Sie den Programmcode zur Weitergabe einer Materialinformation programmieren. Dazu verwenden Sie einen ARRAY-Datenbaustein.

Das Beispiel ist so flexibel aufgebaut, dass Sie zum Zeitpunkt der Programmcodeerstellung noch nicht wissen müssen, welcher ARRAY-Datenbaustein gelesen, bzw. geschrieben wird,

oder wie groß er ist. Dadurch können Sie ARRAYs unterschiedlicher Länge verwenden. Diese Flexibilität erreichen Sie durch Verwendung des Datentyps DB_ANY.

Damit Sie auch bei der Angabe des Werts, der gelesen oder geschrieben werden soll, flexibel sind, wird der Datentyp VARIANT verwendet.

Erst zur Laufzeit wird der ARRAY-Datenbaustein übergeben, um dann im Programmbaustein auf die Werte zuzugreifen. Der Datentyp der ARRAY-Elemente und der Datentyp des Werts, der gelesen oder geschrieben werden soll, wird ermittelt.

Die Anzahl der Objekte bzw. den Füllgrad des ARRAY-Datenbausteins können Sie mit Hilfe spezieller Anweisungen ermitteln.

Vorgehensweise

Legen Sie den PLC-Datentyp "UDT_Queue" an. Dieser PLC-Datentyp wird von beiden Funktionen ("FC_Enqueue" und "FC_Dequeue") verwendet. Das ist z. B. für den Zugriff auf die Variable #Queue.Used wichtig, da die Funktion "FC_Enqueue" die Variable um eins hochzählt und die Funktion "FC_Dequeue" die Variable um eins verringert.

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   DB > Datentyp: DB_ANY
   Size > Datentyp: DINT
   Used > Datentyp: DINT
   ReadPos > Datentyp: DINT
   WritePos > Datentyp: DINT

UDT_Queue

		Name	Data type	Default value	Comment
1		DB	DB_ANY	0	
2		Size	DInt	0	
3		Used	DInt	0	
4		ReadPos	DInt	0	Pointer Reading Position
5		WritePos	DInt	0	Pointer Writing Position

Programmieren Sie die Funktion "FC_Enqueue", die die Werte der Materialinformation in einen ARRAY-Datenbaustein schreibt. Der konkrete ARRAY-Datenbaustein und der Datentyp des

Werts müssen zu diesem Zeitpunkt noch nicht bekannt sein, da die Schnittstellen mit den Datentypen VARIANT und DB_ANY programmiert werden:

1. Legen Sie eine SCL-Funktion an und benennen sie "FC_Enqueue".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_Enqueue

	Name	Data type	Comment
1	Input		
2	Value	Variant	
3	Output		
4	InOut		
5	Queue	"UDT_Queue"	
6	DB	DB_ANY	
7	Size	DInt	
8	Used	DInt	
9	ReadPos	DInt	Pointer Reading Position
10	WritePos	DInt	Pointer Writing Position
11	Temp		
12	Error	Int	
13	<Add new>		
14	Constant		
15	Return		
16	FC_Enqueue	Int	

3. Schreiben Sie folgenden Programmcode:

```
1  IF #Queue.Used < #Queue.Size THEN // As long as there is enough space in the DB
2                                    // the function keeps on writing.
3
4      #Error := WriteToArrayDB(db := #Queue.DB, index := #Queue.WritePos, value := #Value);
5
6      IF #Error = 0 THEN
7
8          #Queue.Used := #Queue.Used + 1;
9          #Queue.WritePos := #Queue.WritePos + 1;
10
11         IF #Queue.WritePos >= #Queue.Size THEN // If this condition is fullfilled
12                                                // the end of the DB is reached.
13                                                // End of writing pointer.
14
15             #Queue.WritePos := 0;
16
17         END_IF;
18
19     END_IF;
20
21 ELSE
22
23     #Error := 4711; // The queue is full.
24
25 END_IF;
```

Mit dieser Funktion prüfen Sie, ob im Datenbaustein noch Platz frei ist. Wenn ja, dann schreiben Sie den Wert, der am Parameter value angegeben wird, in den Datenbaustein am Parameter db. Mit jeder neuen Materialinformation, die geschrieben wird, wird die Variable #Queue.Used und die Zeigervariable #Queue.WritePos um eins nach oben gezählt. Sobald der Schreibzeiger am Ende des Datenbausteins angekommen ist, wird er wieder auf 0 gesetzt. Wenn der Datenbaustein voll ist, wird der Fehlercode #4711 ausgegeben.

Programmieren Sie die Funktion "FC_Dequeue", die die Materialinformation aus einem ARRAY-Datenbaustein ausliest und an das Panel schreibt. Der konkrete ARRAY-Datenbaustein und der Datentyp des Werts müssen zu diesem Zeitpunkt noch nicht bekannt

sein, da die Schnittstellen mit den Datentypen VARIANT und DB_ANY programmiert werden. Anschließend kann die Materialinformation z. B. an einem Panel angezeigt werden:

1. Legen Sie eine SCL-Funktion an und benennen sie "FC_Dequeue".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_Dequeue

	Name	Data type	Comment
1	Input		
2	Output		
3	InOut		
4	Value	Variant	
5	Queue	"UDT_Queue"	
6	DB	DB_ANY	
7	Size	DInt	
8	Used	DInt	
9	ReadPos	DInt	Pointer Reading Position
10	WritePos	DInt	Pointer Writing Position
11	Temp		
12	Error	Int	
13	Constant		
14	Return		
15	FC_Dequeue	Int	

3. Schreiben Sie folgenden Programmcode:

```
1  IF #Queue.Used > 0 THEN // As long as there is some information in the DB
2                          // the function keeps on reading.
3
4      #Error := ReadFromArrayDB(db := #Queue.DB, index := #Queue.ReadPos, value => #Value);
5
6      IF #Error = 0 THEN
7
8          #Queue.Used := #Queue.Used - 1;
9          #Queue.ReadPos := #Queue.ReadPos + 1;
10
11         IF #Queue.ReadPos >= #Queue.Size THEN // If this condition is fullfilled
12                                               // the end of the DB is reached.
13                                               // End of reading pointer.
14
15             #Queue.ReadPos := 0;
16
17         END_IF;
18
19     END_IF;
20
21 ELSE
22
23     #Error := 4712; // The queue is empty.
24
25 END_IF;
```

Mit dieser Funktion prüfen Sie, ob im Datenbaustein Materialinformationen stehen. Wenn ja, dann lesen Sie den Wert, auf den der Zeiger #Queue.ReadPos zeigt aus, und schreiben ihn in die Variable #Value. Mit jeder Materialinformation, die ausgelesen wird, wird die Variable #Queue.Used um eins verringert und die Zeigervariable #Queue.ReadPos um eins nach oben gezählt. Sobald der Lesezeiger am Ende des Datenbausteins angekommen ist, wird er wieder auf 0 gesetzt. Wenn der Datenbaustein leer ist, wird der Fehlercode #4712 ausgegeben.

Um die Materialdaten zu speichern, legen Sie einen ARRAY-Datenbaustein an. Als Datentyp des ARRAY-Datenbausteins verwenden Sie den PLC-Datentyp "UDT_Material".

Legen Sie zunächst den PLC-Datentyp "UDT_Material" an. Dieser PLC-Datentyp enthält eine Struktur für die vom Scanner gelieferten Materialinformationen.

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   ArticleNumber > Datentyp: DINT
   ArticleName > Datentyp: STRING
   Amount > Datentyp: REAL
   Unit > Datentyp: STRING

UDT_Material

		Name	Data type
1		ArticleNumber	DInt
2		ArticleName	String
3		Amount	Real
4		Unit	String

Legen Sie den ARRAY-Datenbaustein "DB_MaterialBuffer" an. Der ARRAY-Datenbaustein soll Datensätze mit Materialinformationen vom Datentyp "UDT_Material" enthalten. Die Materialinformationen werden mit Hilfe der Funktion "FC_Enqueue" in den ARRAY-Datenbaustein gschrieben.

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_MaterialBuffer" an.
4. Wählen Sie als Typ des Datenbausteins "ARRAY-DB" aus.
5. Wählen Sie als ARRAY-Datentyp den PLC-Datentyp "UDT_Material" aus.

6. Geben Sie als obere ARRAY-Grenze "1000" an.
7. Klicken Sie auf "OK".

DB_MaterialBuffer

	Name	Data type
1	DB_MaterialBuffer	Array[0..1000] of "UDT_Material"
2	DB_MaterialBuffer[0]	"UDT_Material"
3	ArticleNumber	DInt
4	ArticleName	String
5	Amount	Real
6	Unit	String
7	DB_MaterialBuffer[1]	"UDT_Material"
8	DB_MaterialBuffer[2]	"UDT_Material"
9	DB_MaterialBuffer[3]	"UDT_Material"
10	DB_MaterialBuffer[4]	"UDT_Material"
11	DB_MaterialBuffer[5]	"UDT_Material"

Legen Sie den Anlauf-Organisationsbaustein (OB) "OB_MaterialQueue" an. In diesem Organisationsbaustein initialisieren Sie die Variablen DB und Size.

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Organisationsbaustein (OB)".
3. Geben Sie den Namen "OB_MaterialQueue" an.
4. Wählen Sie den Typ "Startup" aus.
5. Wählen Sie als Sprache des Organisationsbausteins SCL aus.
6. Klicken Sie auf "OK".
7. Schreiben Sie folgenden Programmcode:

```
1  "DB_MaterialQueue".DB := "DB_MaterialBuffer";
2  "DB_MaterialQueue".Size := 1000;
3  "DB_MaterialQueue".Used := 0;
4  "DB_MaterialQueue".ReadPos := 0;
5  "DB_MaterialQueue".WritePos := 0;
```

Durch das Zuweisen des Datenbausteins verbinden Sie den ARRAY-Datenbaustein "DB_MaterialBuffer" mit den SCL-Funktionen "FC_Enqueue" und "FC_Dequeue". Am Parameter Size geben Sie die Größe des ARRAY-Datenbausteins an. Der Startwert des Parameters Used ist "0", damit wird die erste Materialinformation in das ARRAY-Element "0" geschrieben.

1. Deklarieren Sie in der "Standard-Variablentabelle" die folgenden Variablen:

Name	Data type	Address
MaterialReceived	Bool	%I0.0
MaterialReceivedEdge	Bool	%M0.0
MaterialDone	Bool	%I1.0
MaterialDoneEdge	Bool	%M0.1

2. Rufen Sie die SCL-Funktion "FC_Enqueue" innerhalb des Funktionsbausteins auf, in dem der Scanner die Materialinformation ausliest.
3. Deklarieren Sie in der Bausteinschnittstelle die Variable "ConnectionToUDT" im Abschnitt "Temp" und verknüpfen Sie diese mit dem PLC-Datentyp "UDT_Material":

Temp	
ConnectionToUDT	"UDT_Material"
ArticleNumber	DInt
ArticleName	String
Amount	Real
Unit	String

4. Verknüpfen Sie den Funktionsaufruf mit den folgenden Variablen und legen Sie am Freigabeeingang EN die Signalflanke "P: Operand auf positive Signalflanke abfragen" an. Verknüpfen Sie die Signalflanke mit den globalen Variablen aus der Standard-Variablentabelle:

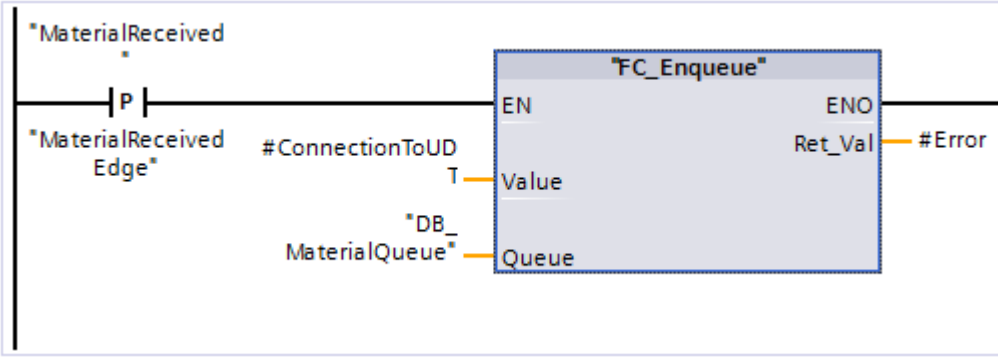

5. Rufen Sie die SCL-Funktion "FC_Dequeue" auf
6. Verknüpfen Sie den Funktionsaufruf mit den folgenden Variablen und legen Sie am Freigabeeingang EN die Signalflanke "P: Operand auf positive Signalflanke abfragen" an. Verknüpfen Sie die Signalflanke mit den globalen Variablen aus der Standard-Variablentabelle:

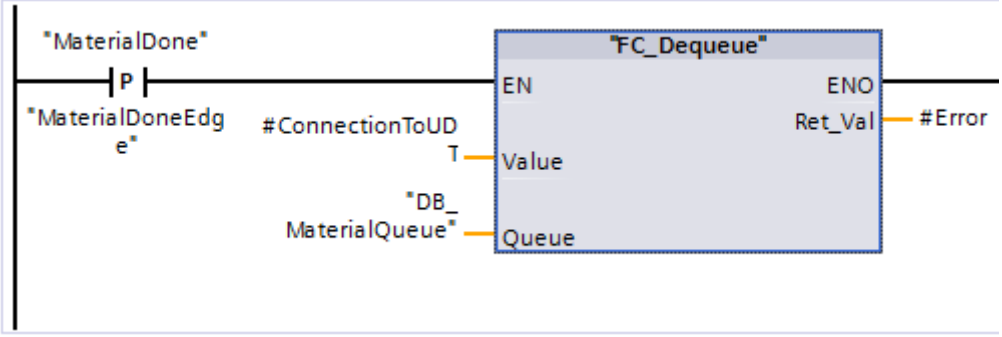

Ergebnis

Sobald eine positive Signalflanke anliegt, wird eine Materialinformation mithilfe der Anweisung "WriteToArrayDB" in einen ARRAY-Datenbaustein geschrieben und mithilfe der Anweisung "ReadFromArrayDB" an das Panel weitergegeben.

Siehe auch

Globale ARRAY-Datenbausteine (DB) (Seite 43)

ARRAY-DBs adressieren (Seite 114)

1.2.9 Instanz-Datenbausteine

Definition

Der Aufruf eines Funktionsbausteins wird als Instanz bezeichnet. Die Daten, mit denen die Instanz arbeitet, werden in einem Instanz-Datenbaustein gespeichert.

Die maximale Größe von Instanz-Datenbausteinen variiert abhängig von der CPU. Die im Funktionsbaustein deklarierten Variablen bestimmen die Struktur des Instanz-Datenbausteins.

Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für die Instanz-Datenbausteine, die einem Funktionsbaustein beim Aufruf zugeordnet werden:

- Datenbausteine mit optimiertem Zugriff
  Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)
  Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

Siehe auch

Datenbausteine anlegen (Seite 7204)

Grundlagen zum Bausteinzugriff (Seite 55)

Instanzen (Seite 64)

Bausteinaufrufe in KOP einfügen (Seite 7449)

Bausteinaufrufe in FUP einfügen (Seite 7516)

1.2.10 CPU-Datenbausteine

Definition

CPU-Datenbausteine sind Datenbausteine, die von der CPU zur Laufzeit erzeugt werden. Dazu fügen Sie in Ihr Anwenderprogramm die Anweisung "CREATE_DB" ein. Nach dem Erzeugen des Datenbausteins zur Laufzeit können Sie den Datenbaustein zum Speichern Ihrer Daten verwenden.

CPU-Datenbausteine werden innerhalb eines erreichbaren Teilnehmers im Ordner "Programmbausteine" mit einem kleinen CPU-Symbol angezeigt. Sie können die aktuellen Werte der Variablen eines CPU-Datenbausteins wie die eines anderen Datenbausteintyps online beobachten.

CPU-Datenbausteine können Sie nicht offline in Ihrem Projekt anlegen.

Laden von CPU-Datenbausteinen

Nach dem das Anwenderprogramm über die Anweisung "CREATE_DB" den CPU-Datenbaustein erzeugt hat, steht er zunächst nur online auf dem Gerät zur Verfügung. Beim nächsten vollständigen Ladevorgang, bei dem Sie die Bausteine vom Gerät in Ihr Projekt laden, werden auch alle CPU-Datenbausteine mit in das Projekt geladen. Dabei werden die CPU-Datenbausteine mit einem kleinen CPU-Symbol gekennzeichnet. Sie können die CPU-Datenbausteine jedoch nicht mehr zurück in Ihr Gerät laden.

Beschränkungen von CPU-Datenbausteinen im Projekt

Wenn die CPU-Datenbausteine durch einen Ladevorgang in Ihr offline Projekt geladen wurden, können Sie sie öffnen und sich ihren Inhalt ansehen. Beachten Sie jedoch, dass CPU-Datenbausteine im Projekt schreibgeschützt sind. CPU-Datenbausteine unterliegen im Projekt daher folgenden Beschränkungen:

- Sie können CPU-Datenbausteine weder bearbeiten noch in einen anderen Datenbausteintyp konvertieren.
- Sie können CPU-Datenbausteine nicht mit einem Know-how-Schutz versehen.
- Sie können die Programmiersprache eines CPU-Datenbausteins nicht ändern.
- CPU-Datenbausteine können nicht übersetzt und auch nicht in ein Gerät geladen werden.

Vergleichen von CPU-Datenbausteinen

Wenn die CPU-Datenbausteine durch einen Ladevorgang in Ihr offline Projekt geladen wurden, können Sie für die geladenen CPU-Datenbausteine einen Offline/Online-Vergleich durchführen. Im Vergleichseditor erhalten Sie eine Übersicht über die Unterschiede. Sie können CPU-Datenbausteine, die sich in ihrer Online- und Offline-Version unterscheiden, jedoch nicht durch Laden der Offline-Version in das Gerät synchronisieren.

Löschen von CPU-Datenbausteinen

Sie können CPU-Datenbausteine sowohl aus Ihrem Projekt als auch in der CPU löschen.

Siehe auch

CPU-Datenbausteine löschen (Seite 7229)

1.2.11 Bausteine mit optimiertem Zugriff

1.2.11.1 Grundlagen zum Bausteinzugriff

Einführung

STEP 7 bietet Datenbausteine mit unterschiedlichen Zugriffsmöglichkeiten an:

- Datenbausteine mit optimiertem Zugriff (S7-1200 / S7-1500)
- Datenbausteine mit Standardzugriff (S7-300 / S7-400 / S7-1200 / S7-1500)

Innerhalb eines Programms können Sie beide Arten von Bausteinen kombinieren.

Datenbausteine mit optimiertem Zugriff

Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins. Die Elemente werden im verfügbaren Speicherbereich des Bausteins automatisch so angeordnet, dass keine Lücken im Speicher entstehen. Somit wird die Speicherkapazität optimal ausgeschöpft.

Variablen in diesen Datenbausteinen werden durch ihren symbolischen Namen identifiziert. Um die Variablen zu adressieren, geben Sie den symbolischen Namen der Variablen an. Zum Beispiel greifen Sie auf die Variable "Füllstand" im DB "Daten" folgendermaßen zu:
`"Daten".Füllstand`

Bausteine mit optimiertem Zugriff bieten folgende Vorteile:

- Sie können Datenbausteine mit beliebiger Struktur aufbauen, ohne auf die physikalische Anordnung der einzelnen Datenelemente zu achten.
- Der Zugriff auf optimierte Daten erfolgt immer schnellstmöglich, da die Datenablage vom System optimiert und verwaltet wird.
- Zugriffsfehler, z. B. bei indirekter Adressierung oder aus HMI heraus, sind nicht möglich.
- Sie können gezielt einzelne Variablen als remanent definieren.
- Optimierte Bausteine verfügen standardmäßig über eine Speicherreserve, die es Ihnen ermöglicht, die Schnittstellen von Funktions- oder Datenbausteinen im laufenden Betrieb zu erweitern. Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Hinweis

Das Attribut "optimierter Bausteinzugriff" ist für folgende Bausteine immer aktiviert und kann nicht abgewählt werden:

- GRAPH-Bausteine
- ARRAY-Datenbausteine

Datenbausteine mit Standardzugriff

Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins. Die Adresse wird in der Spalte "Offset" angezeigt.

Variablen in diesen Datenbausteinen können Sie sowohl symbolisch als auch absolut adressieren:
`"Daten".Füllstand`
`DB1.DBW2`

Remanenz bei optimiertem Zugriff oder Standardzugriff

Wenn Sie Daten als remanent definieren, bleiben deren Werte nach einem Spannungsausfall oder Netz-Aus erhalten. Eine remanente Variable wird nach dem Wiederanlauf nicht initialisiert, sondern behält den Wert, den sie vor dem Spannungsausfall hatte. Wenn eine DB-Variable als remanent definiert wird, wird sie im remanenten Speicherbereich des Datenbausteins abgelegt.

Die Möglichkeiten zur Einstellung der Remanenz hängen von der Zugriffsart des Bausteins ab:

- In Datenbausteinen mit Standardzugriff können Sie das Remanenzverhalten einzelner Variablen nicht einstellen. Die Remanenzeinstellung gilt für alle Variablen des Datenbausteins.
- In Datenbausteinen mit optimiertem Zugriff können Sie das Remanenzverhalten einzelner Variablen bestimmen.
  Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb des Datentyps können Sie keine individuelle Remanenzeinstellung vornehmen.

Adressierungsmöglichkeiten bei optimiertem Zugriff und Standardzugriff

Bausteine mit optimiertem Zugriff lassen nur sogenannte "typsichere" Zugriffe zu. Typsichere Zugriffe adressieren Variablen nur über den symbolischen Namen. So kann es auch bei Änderungen am Baustein oder an der Bausteinschnittstelle nicht zu Inkonsistenzen im Programm oder Zugriffsfehlern kommen.

Die folgende Tabelle zeigt die zulässigen Adressierungsmöglichkeiten für optimierte Daten:

Adressierung	Baustein mit Standardzugriff	Baustein mit optimiertem Zugriff
Symbolische Adressierung	x	x
Indizierte Adressierung von ARRAYs		x
Slice-Zugriffe	x	x
Überlagerung mit AT	x	-
Absolute Adressierung	x	-
Indirekte Adressierung über ANY	x	-
Indirekte Adressierung über POINTER und VARIANT	x	nur bei symbolischer Notation

Siehe auch

Bausteinzugriff einstellen (Seite 58)

Variablen in Datenbausteinen adressieren (Seite 111)

Bausteine anlegen (Seite 7201)

1.2.11.2 Bausteinzugriff einstellen

Einführung

Der Bausteinzugriff wird beim Anlegen eines Bausteins automatisch eingestellt:

- Auf CPUs der Baureihe S7-1200/1500 haben neu erzeugte Bausteine per Voreinstellung optimierten Zugriff.
- Auf CPUs der Baureihe S7-300/S7-400 haben neu erzeugte Bausteine per Voreinstellung Standardzugriff.

Wenn Sie einen Baustein in eine CPU einer anderen Baureihe kopieren oder migrieren, wird der Bausteinzugriff nicht automatisch umgestellt. Es kann jedoch sinnvoll sein, den Bausteinzugriff manuell zu ändern, z. B. um den vollen Funktionsumfang der CPU nutzen zu können.

Nach der Umstellung des Bausteinzugriffs müssen Sie das Programm in den meisten Fällen neu übersetzen und laden.

ACHTUNG
Optimierter Bausteinzugriff bei GRAPH-Bausteinen Das Attibut "optimierter Bausteinzugriff" ist für GRAPH-Bausteine in S7-1500 immer aktiviert und kann nicht abgewählt werden.

Vorgehensweise

Um den Bausteinzugriff einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Baustein, dessen Bausteinzugriff Sie ändern möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf "Attribute".
5. Aktivieren oder deaktivieren Sie die Option "Optimierter Bausteinzugriff".
6. Bestätigen Sie Ihre Eingaben mit "OK".

Einschränkungen und Besonderheiten

Grundsätzlich können Sie den Bausteinzugriff nur bei CPUs der Baureihe S7-1200/1500 umstellen, da nur diese die Zugriffsart "optimiert" unterstützen.

Dabei gibt es folgende Einschränkungen oder Besonderheiten:

- Instanzdatenbausteine
  Der Bausteinzugriff von Instanzdatenbausteinen richtet sich immer nach dem zugeordneten Funktionsbaustein und kann nicht manuell geändert werden. Wenn Sie die Zugriffsart eines Funktionsbausteins ändern, müssen Sie die zugeordneten Instanzdatenbausteine aktualisieren. Bei der Aktualisierung wird die Zugriffsart des Instanzdatenbausteins angepasst.
- Systembausteine und know-how-geschützte Bausteine
  Bei Systembausteinen und know-how-geschützten Bausteinen können Sie den Bausteinzugriff nicht manuell ändern.
- Organisationsbausteine
  Die Startinformationen eines OB mit Standardzugriff liegen in der Bausteinschnittstelle immer in den ersten 20 Byte des Abschnitts "Temp". Die Startinformationen eines OB mit optimiertem Zugriff hingegen werden im Abschnitt "Input" abgelegt. Deshalb ändert sich bei OBs durch die Umstellung des Bausteinzugriffs auch die Bausteinschnittstelle. Details sind in den nachfolgenden Abschnitten beschrieben.

Umstellen des Bausteinzugriffs von "standard" nach "optimiert"

Wenn Sie einen Baustein von einer CPU der Baureihe S7-300/400 auf eine CPU der Baureihe S7-1200/1500 kopieren oder migrieren, behält er zunächst die Zugriffsart "standard". Die Programmbearbeitung wird jedoch durch die Verwendung von Bausteinen mit optimiertem Zugriff deutlich performanter. Aus diesem Grund kann es sinnvoll sein, die Zugriffsart manuell zu ändern.

Bei der Umstellung werden die Bausteine wie folgt angepasst:

- Funktionsbausteine
  Alle Schnittstellenparameter erhalten die Remanenzeinstellung "nicht remanent".
- Globale Datenbausteine
  Die Remanenzeinstellung, die zuvor zentral für den gesamten Datenbaustein eingestellt war, wird auf die einzelnen Schnittstellenparameter übertragen. Anschließend können Sie die Remanenzeinstellung der einzelnen Parameter manuell einstellen.
  Es gilt jedoch weiterhin folgende Regel: Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb eines strukturierten Datentyps können Sie keine individuelle Remanenzeinstellung vornehmen. Daraus folgt, dass auch in Datenbausteinen, die auf PLC-Datentypen basieren, keine individuellen Remanenzeinstellungen für die einzelnen Variablen möglich sind.
- Organisationsbausteine
  Alle Schnittstellenparameter, die in den ersten 20 Byte des Abschnitts "Temp" liegen, werden gelöscht. Im Abschnitt "Input" werden neue CPU-spezifische Startinformationen angelegt. Wenn dabei Namenskonflikte mit anwenderdefinierten Schnittstellenparametern auftreten, werden die anwenderdefinierten Schnittstellenparameter umbenannt.

⚠ VORSICHT

Die Umstellung des Bausteinzugriffs hat folgende Konsequenzen:

- Nach der Umstellung des Bausteinzugriffs auf "optimiert" können Sie die Schnittstellenparameter des Bausteins nicht mehr absolut adressieren.
  Beispiel: `#L0.1` ist nicht mehr zulässig.
- Durch die Umstellung des Bausteinzugriffs von Organisationsbausteinen auf "optimiert" ändert sich die Schnittstelle des OB.

Sie müssen das Programm aufgrund dieser Änderungen möglicherweise anpassen, neu übersetzen und laden.

Umstellen des Bausteinzugriffs von "optimiert" nach "standard"

Wenn Sie einen Baustein von einer CPU der Baureihe S7-1200/1500 auf eine CPU der Baureihe S7-300/400 kopieren oder verschieben möchten, müssen Sie zunächst den Bausteinzugriff auf "standard" setzen.

Bei der Umstellung werden die Bausteine wie folgt angepasst:

- Funktionsbausteine und globale Datenbausteine:
  Im Funktionsbaustein kann keine Remanenzeinstellung mehr vorgenommen werden. Die Einstellung erfolgt am Instanz-Datenbaustein.
  Alle Schnittstellenparameter im Instanzdatenbaustein oder globalen Datenbaustein erhalten dieselbe Remanenzeinstellung. Die Umstellung erfolgt nach folgender Regel:
  - Wenn alle Schnittstellenparameter im Ausgangsbaustein remanent waren, wird der gesamte Baustein nach der Umstellung remanent.
  - Wenn alle Schnittstellenparameter im Ausgangsbaustein nicht-remanent waren, wird der gesamte Baustein nach der Umstellung nicht-remanent.
  - Wenn die Schnittstellenparameter im Ausgangsbaustein unterschiedliche Remanenzeinstellungen hatten, wird der gesamte Baustein nach der Umstellung nicht-remanent.
- Organisationsbausteine
  Alle Schnittstellenparameter, die im Abschnitt "Input" liegen, werden gelöscht. Im Abschnitt "Temp" werden neue CPU-spezifische Startinformationen angelegt. Dabei werden die ersten 20 Byte belegt. Wenn dabei Namenskonflikte mit anwenderdefinierten Schnittstellenparametern auftreten, werden die anwenderdefinierten Schnittstellenparameter umbenannt.

⚠ VORSICHT

Die Umstellung des Bausteinzugriffs hat folgende Konsequenzen:

Bei der Umstellung des Bausteinzugriffs auf "standard" können sich die Remanenzeinstellungen der Schnittstellenparameter ändern. Sie müssen das Programm aufgrund dieser Änderungen möglicherweise anpassen, neu übersetzen und laden.

Siehe auch

Grundlagen zum Bausteinzugriff (Seite 55)
Bausteineigenschaften festlegen (Seite 7214)

1.3 Bausteinaufrufe

1.3.1 Grundlagen zu Bausteinaufrufen

Funktion von Bausteinaufrufen

Damit Ihre Bausteine im Anwenderprogramm ausgeführt werden, müssen sie von einem anderen Baustein aus aufgerufen werden.

Ruft ein Baustein einen weiteren Baustein auf, werden die Anweisungen des aufgerufenen Bausteins bearbeitet. Erst wenn die Bearbeitung des aufgerufenen Bausteins beendet ist, wird die Bearbeitung des aufrufenden Bausteins wieder aufgenommen. Die Bearbeitung wird mit der Anweisung fortgesetzt, die auf den Bausteinaufruf folgt.

Das folgende Bild zeigt den Ablauf eines Bausteinaufrufs innerhalb eines Anwenderprogramms:

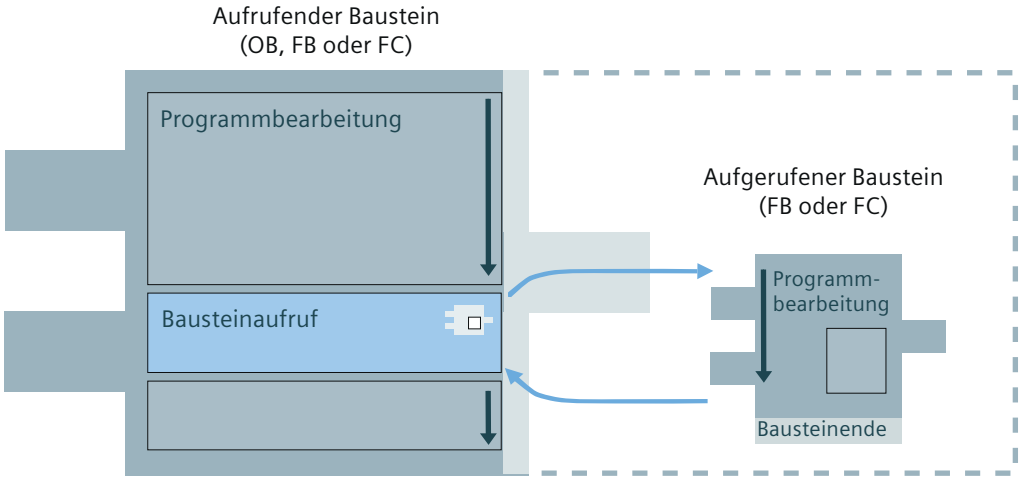

Parameterübergabe

Beim Bausteinaufruf versorgen Sie die Parameter in der Schnittstelle des Bausteins mit Werten. Durch die Versorgung der Eingangsparameter legen Sie fest, mit welchen Daten die Bearbeitung des Bausteins erfolgt. Durch die Versorgung der Ausgangsparameter legen Sie fest, wo die Bearbeitungsergebnisse abgelegt werden.

Siehe auch

Aufrufhierarchie (Seite 63)

Grundlagen zu Instanzen (Seite 64)

Parameterübergabe beim Bausteinaufruf (Seite 78)

1.3.2 Aufrufhierarchie

Definition

Als Aufrufhierarchie wird die Reihenfolge und Schachtelung der Bausteinaufrufe bezeichnet.

Das folgende Bild zeigt ein Beispiel für die Reihenfolge und Schachtelung der Bausteinaufrufe innerhalb eines Bearbeitungszyklus:

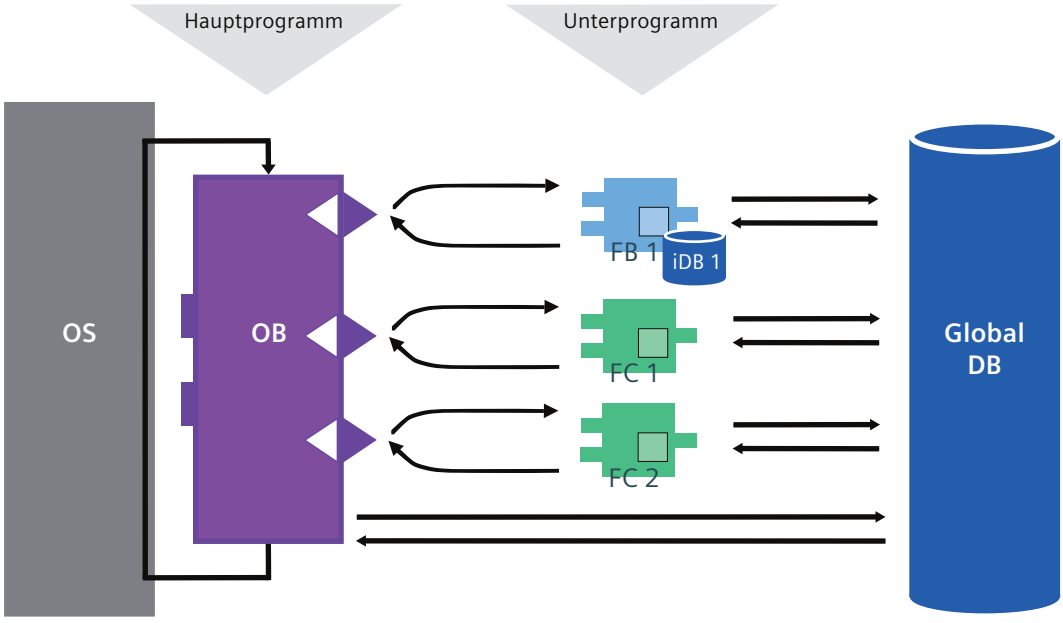

Schachtelungstiefe für Bausteine

Die zulässige Schachtelungstiefe für Bausteine ist von der eingesetzten CPU abhängig.

Die folgende Tabelle zeigt die Richtwerte für die maximale Schachtelungstiefe. Spezifische Angaben für die von Ihnen eingesetzte CPU finden Sie in den technischen Daten in der Hardware-Dokumentation. Um die Hardware-Dokumentation im Internet aufzurufen, klicken Sie auf die Links in der Tabelle.

CPU-Familie	Schachtelungstiefe (Richtwert)	Link auf Hardware-Dokumentation
S7-1500	24 Bausteine pro Prioritätsklasse	SIMATIC S7-1500 / ET 200MP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/86140384)
S7-1200	16 Bausteine aus Zyklus- oder Anlauf-OB 6 zusätzliche Bausteine innerhalb eines beliebigen Alarmereignis-OB	SIMATIC S7-1200 Automatisierungssystem (https://support.industry.siemens.com/cs/ww/de/view/91696622)
S7-400	24 Bausteine pro Prioritätsklasse 1-2 zusätzliche Bausteine innerhalb eines Fehler-OB	SIMATIC S7-400 Automatisierungssystem S7-400 CPU-Daten (https://support.industry.siemens.com/cs/de/de/view/53385241)

CPU-Familie	Schachtelungstiefe (Richtwert)	Link auf Hardware-Dokumentation
S7-300	16 Bausteine pro Prioritätsklasse 4 zusätzliche Bausteine innerhalb eines Fehler-OB	SIMATIC S7-300 CPU 31xC und CPU 31x: Technische Daten (https://support.industry.siemens.com/cs/de/de/view/12996906)
ET 200SP	24 Bausteine	SIMATIC ET 200SP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/84133942)

Schachtelungstiefe für Strukturen

Strukturen (STRUCT) und PLC-Datentypen (UDT) können bis zu einer Tiefe von 8 geschachtelt werden. Diese Schachtelungstiefe ist unabhängig von der eingesetzten CPU.

Siehe auch

Grundlagen zu Instanzen (Seite 64)

Grundlagen zu Bausteinaufrufen (Seite 62)

1.3.3 Instanzen

1.3.3.1 Grundlagen zu Instanzen

Definition

Ein Funktionsbaustein benötigt nach seinem Aufruf Speicher für seine Arbeitsdaten. Diese Daten werden als Instanz bezeichnet.

Instanzen haben folgende Eigenschaften:

- Instanzen sind immer einem FB zugeordnet.
- Die Struktur einer Instanz wird von der Schnittstelle des zugehörigen FB abgeleitet und kann nur dort geändert werden.
- Instanzen werden automatisch beim Aufruf eines Funktionsbausteins angelegt.

Ablage von Instanzen

Für die Ablage von Instanzen haben Sie folgende Möglichkeiten:

- Einzelinstanz:
  Der aufgerufene Funktionsbaustein speichert seine Daten in einem eigenen Instanz-Datenbaustein.
  Siehe auch: Einzelinstanzen (Seite 65)
- Multiinstanz:
  Der aufgerufene Funktionsbaustein speichert seine Daten nicht in einem eigenen Instanz-Datenbaustein, sondern in der Instanz eines anderen Funktionsbausteins.
  Siehe auch: Multiinstanzen (Seite 67)
- Parameterinstanz:
  Sie übergeben die Instanz eines Funktionsbausteins, als Durchgangsparameter (InOut) einen anderen Baustein. Dieser Baustein kann auf die Daten in der übergebenen Instanz zugreifen oder den zugehörigen FB aufrufen.
  Siehe auch: Parameterinstanzen (Seite 70)

Siehe auch

Übersicht über die Bausteinschnittstelle (Seite 7263)

Instanzdaten adressieren (Seite 113)

Variablen in Datenbausteinen adressieren (Seite 111)

Grundlagen zu Bausteinaufrufen (Seite 62)

Aufrufhierarchie (Seite 63)

1.3.3.2 Einzelinstanzen

Definition

Eine Instanz eines Funktionsbausteins (FB), die in einem eigenen Instanz-DB liegt, wird als Einzelinstanz bezeichnet. Der Instanz-DB enthält also die Arbeitsdaten für einen einzelnen Bausteinaufruf.

Vorteile

Die Verwendung von Einzelinstanzen bietet folgende Vorteile:

- Wiederverwendbarkeit der Funktionsbausteine
- Gute Strukturierungsmöglichkeit für einfache Programme

Funktionsweise von Einzelinstanzen

Die folgende Abbildung zeigt einen FB, der einen weiteren FB („FB_Valve“) nutzt. "FB_Valve" wird als Einzelinstanz aufgerufen, d. h. er speichert seine Daten in seinem eigenen Instanz-DB.

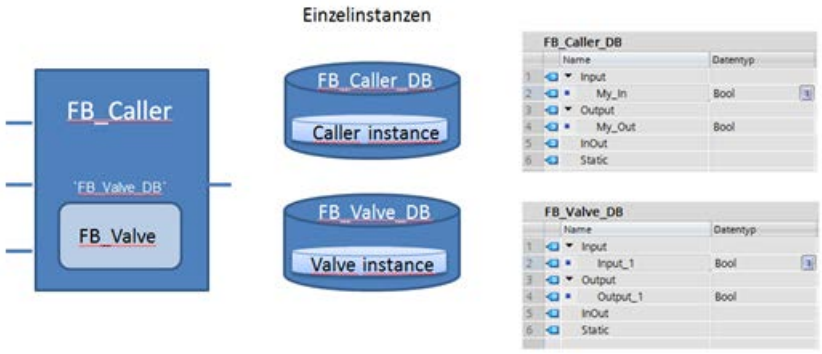

Die Struktur des Instanz-Datenbausteins wird von der Schnittstelle des zugehörigen FB definiert und kann nur dort geändert werden. Der Instanz-DB enthält die folgenden Daten:

- Bausteinparameter
  Die Bausteinparameter in den Abschnitten "Input", "Output" und "InOut" bilden die Schnittstelle des Bausteins für den Aufruf im Programm.
- Statische Lokaldaten
  Statische Lokaldaten im Abschnitt "Static" dienen zum dauerhaften Speichern von Zwischenergebnissen über den aktuellen Programmzyklus hinaus, z. B. zur Speicherung des Signalzustands bei einer Flankenauswertung.

Einzelinstanzen anlegen

Sie haben folgende Möglichkeiten, eine Einzelinstanz anzulegen:

- Wenn Sie einen FB im Programm aufrufen, erscheint der Dialog "Aufrufoptionen". Hier können Sie angeben, ob Sie den FB als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen wollen.
- Wenn Sie einen neuen Datenbaustein anlegen, erscheint der Dialog "Neuen Baustein hinzufügen". Hier können Sie unter "Typ" einen Funktionsbaustein auswählen, für den Sie einen Instanz-Datenbaustein erstellen wollen.

Einzelinstanzen aufrufen

Beim Aufruf einer Einzelinstanz wird der zugeordnete Instanz-Datenbaustein angegeben.

Das folgende Bild zeigt den Aufruf des Bausteins "Called_Block" als Einzelinstanz in KOP. Der Instanz-Datenbaustein "DB_Called_Block" steht über dem Aufruf.

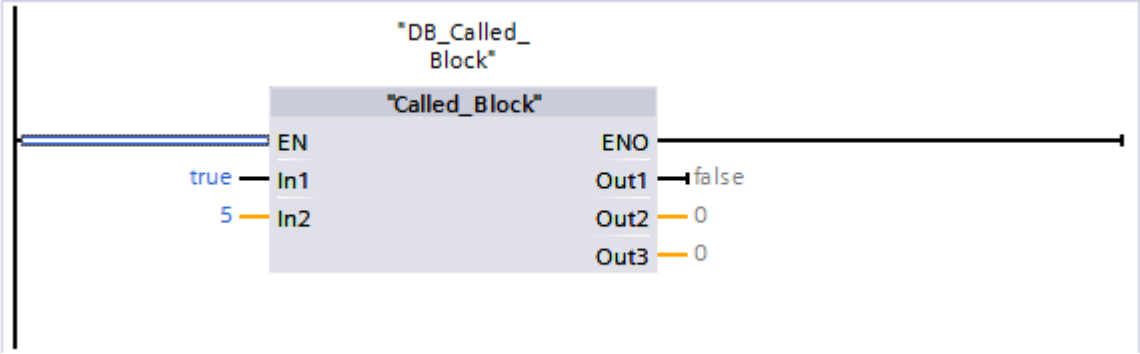

Siehe auch

Grundlagen zu Instanzen (Seite 64)

Multiinstanzen (Seite 67)

1.3.3.3 Multiinstanzen

Definition

Wenn ein Funktionsbaustein (FB) einen weiteren FB aufruft, können seine Instanzdaten auch im Instanz-DB des aufrufenden FB gespeichert werden. Diese Art des Bausteinaufrufs wird als Multiinstanz bezeichnet.

Vorteile

Die Verwendung von Multiinstanzen bietet folgende Vorteile:

- Gute Strukturierungsmöglichkeit für komplexe Bausteine
- Geringere Zahl von Instanz-DBs
- Einfaches Programmieren lokaler Unterprogramme, z. B. für lokale Timer oder Flankenauswertungen.

Funktionsweise von Multiinstanzen

Die folgende Abbildung zeigt einen FB, der einen weiteren FB („FB_Valve“) nutzt. "FB_Valve" wird als Multiinstanz aufgerufen, d. h. er speichert seine Daten im Instanz-DB des aufrufenden FB. Multiinstanzdaten liegen im Abschnitt "Static" des aufrufenden Bausteins. In CPUs der Baureihen S7-1200/S7-1500 kann die Instanz auch im Instanz-DB eines anderen Funktionsbausteins liegen.

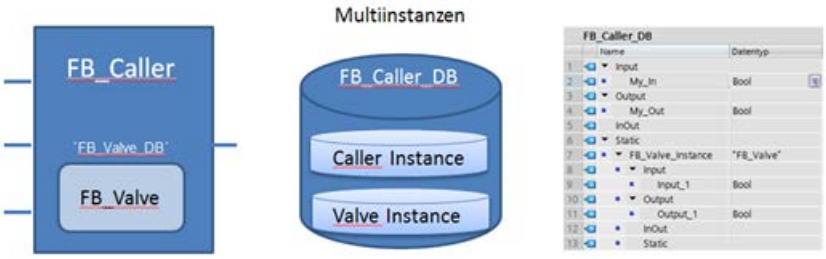

Multiinstanzen anlegen

Sie haben folgende Möglichkeiten, eine Multiinstanz anzulegen:

- Wenn Sie einen FB im Programm aufrufen, erscheint der Dialog "Aufrufoptionen". Hier können Sie angeben, ob Sie den FB als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen wollen.
- Sie deklarieren die Multiinstanz direkt in der Schnittstelle des aufrufenden Bausteins.
- Sie deklarieren die Multiinstanz in der Schnittstelle eines anderen Funktionsbausteins (S7-1200/S7-1500).

Siehe auch: Multiinstanzen deklarieren

ARRAYs von Multiinstanzen

Multiinstanzen können auch als ARRAY angelegt werden. Die einzelnen ARRAY-Elemente können Sie über einen variablen Index adressieren, z. B. bei der Bearbeitung von Programmschleifen.

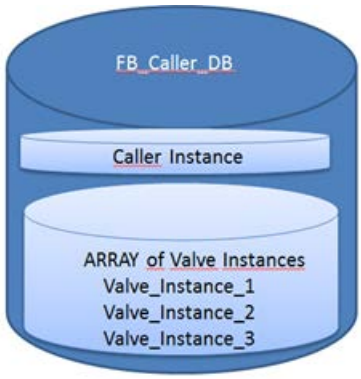

ARRAYs von Multiinstanzen deklarieren Sie direkt in der Schnittstelle eines Funktionsbausteins.

Multiinstanzen aufrufen

Beim Aufruf einer Multiinstanz wird die zugeordnete Instanz angegeben. Die folgenden Bilder zeigen den Aufruf des Bausteins "Called_Block" als Multiinstanz in KOP.

Im folgenden Beispiel liegt die Instanz lokal im Instanz-DB des aufrufenden Bausteins:

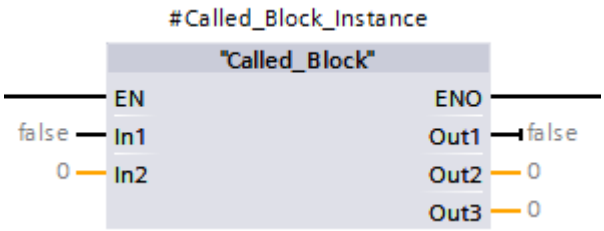

Im folgenden Beispiel liegt die Instanz im Instanz-DB "MyOtherFB" (S7-1200/S7-1500):

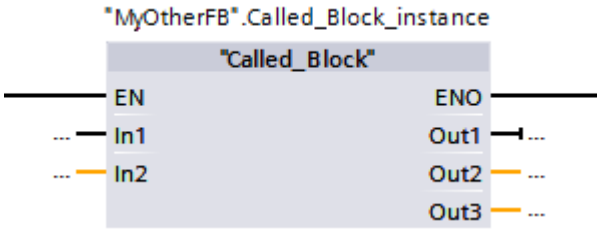

Im folgenden Beispiel liegt die Instanz in einem ARRAY von Multiinstanzen im Instanz-DB "MyOtherFB" (S7-1200/S7-1500):

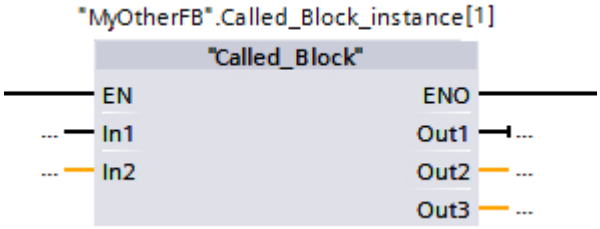

Siehe auch

Grundlagen zu Instanzen (Seite 64)

Einzelinstanzen (Seite 65)

Beispiel zur Verwendung von Parameterinstanzen (Seite 73)

1.3.3.4 Parameterinstanzen

Definition

Eine besondere Möglichkeit der Instanziierung bietet die Parameterinstanz:

Hier übergeben Sie zur Laufzeit die Instanz des Bausteins, die genutzt werden soll, als Durchgangsparameter (InOut) an den aufrufenden Baustein.

Vorteile

Die Verwendung von Parameterinstanzen bietet folgende Vorteile:

- Sie können zur Laufzeit definieren, welche Instanz aktuell verwendet wird.
- Sie können verschiedene Instanzen iterierend in Programmschleifen bearbeiten.

Hinweis

ARRAYs von Multiinstanzen

Um die iterierende Bearbeitung von Instanzen in Programmschleifen zu vereinfachen, definieren Sie ein ARRAY von Multiinstanzen.

Siehe auch: Multiinstanzen (Seite 67)

Funktionsweise von Parameterinstanzen

Das folgende Bild zeigt den "FB_Caller", der einen weiteren FB ("Valve") nutzt. Eine Instanz von "Valve" wird als Parameterinstanz übergeben. Dazu definieren Sie einen Durchgangsparameter ("#Valve_Instance"), über den zur Laufzeit die konkrete Instanz übergeben wird.

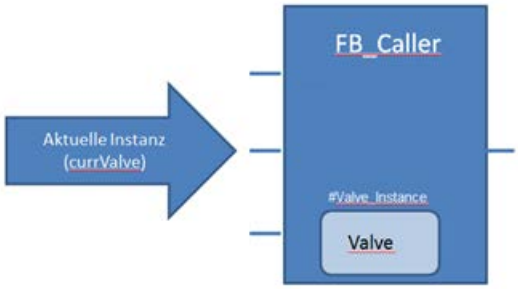

Parameterinstanzen anlegen

Parameterinstanzen werden beim Aufruf eines Funktionsbausteins definiert: Wenn Sie einen FB aufrufen, erscheint ein Dialog, in dem Sie angeben können, ob Sie den FB als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen wollen. Alternativ dazu haben Sie auch die Möglichkeit, Parameterinstanzen direkt in der Bausteinschnittstelle manuell einzugeben.

Siehe auch: Parameterinstanzen deklarieren (Seite 7283)

Das folgende Bild zeigt den Aufruf einer Parameterinstanz für den FB "Valve":

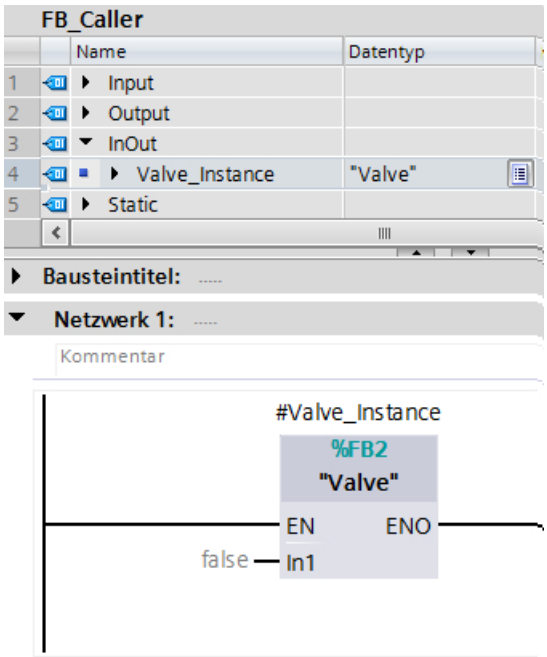

Instanz als Parameter übergeben

Bei jedem Aufruf des übergeordneten FB (im Beispiel der "FB_Caller") übergeben Sie eine Instanz für den aufgerufenen FB (im Beispiel "Valve"). Sie können folgende Arten von Instanzen übergeben:

- Einzelinstanz
  Übergeben Sie einen bestehenden Instanz-DB des aufgerufenen FB.
- Multiinstanz
  Übergeben Sie eine bestehende Multiinstanz des aufgerufenen FB.
- Einzelnes Element eines ARRAYs von Multiinstanzen
  Übergeben Sie ein Element eines bestehenden ARRAYs von Multiinstanzen.

Das folgende Bild zeigt den Aufruf des "FB _Caller". Am Parameter "Valve_Instance" wird die Instanz "#currValve" übergeben. Diese Instanz, die zur Laufzeit am Parameter #currValve ansteht, wird für den Aufruf von "Valve" genutzt.

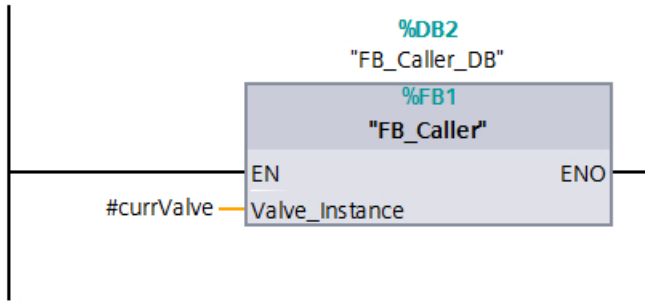

In CPUs der Baureihen S7-1200/S7-1500 können Sie auch Instanzen als Parameter übergeben, die im Instanz-DB eines anderen Bausteins liegen.

Im folgenden Beispiel liegt die übergebene Instanz im Instanz-DB "MyOtherIDB" (S7-1200/S7-1500):

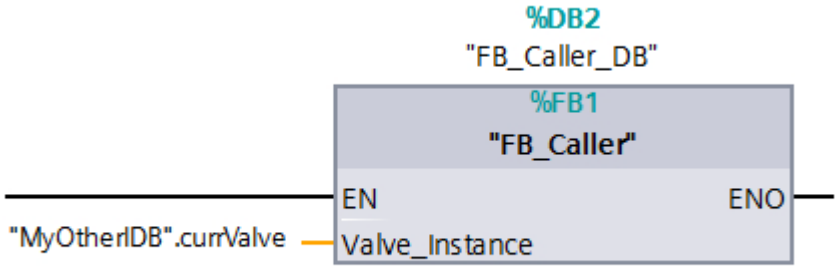

Im folgenden Beispiel liegt die übergebene Instanz in einem ARRAY von Multiinstanzen im Instanz-DB "MyOtherIDB" (S7-1200/S7-1500):

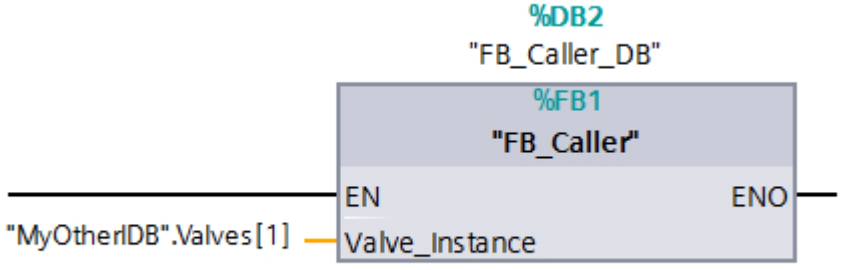

Siehe auch

Beispiel zur Verwendung von Parameterinstanzen (Seite 73)

1.3.3.5 Beispiel zur Verwendung von Parameterinstanzen

Beschreibung

Die Verwendung von Parameterinstanzen ermöglicht es Ihnen, die Instanz eines Funktionsbausteins an einen anderen Baustein (FB oder FC) für eine weitere Bearbeitung, wie z. B. eine Datenabfrage oder eine Fehleranalyse, zu übergeben oder den Funktionsbaustein selbst mit der übergebenen Instanz auszuführen.

Das ARRAY von Multiinstanzen kann dafür verwendet werden, gleichartige Objekte zusammenzufassen und deren Instanzen indiziert in einer Programmschleife zu bearbeiten. Indizierte ARRAY-Elemente können auch als Parameterinstanz an einen anderen Baustein übergeben werden.

Zur Veranschaulichung hierfür wurde im folgenden Beispiel das Objekt "Ventil" (FB_Valve) gewählt. Auf dieser Basis stehen Ihnen alle notwendigen Informationen rund um die Ventilbearbeitung in einem Programmbaustein, hier "FB_ValveControl", zur Verfügung.

Das folgende Bild veranschaulicht Ihnen, welche Funktionen benötigt werden und wie sie eingesetzt werden können:

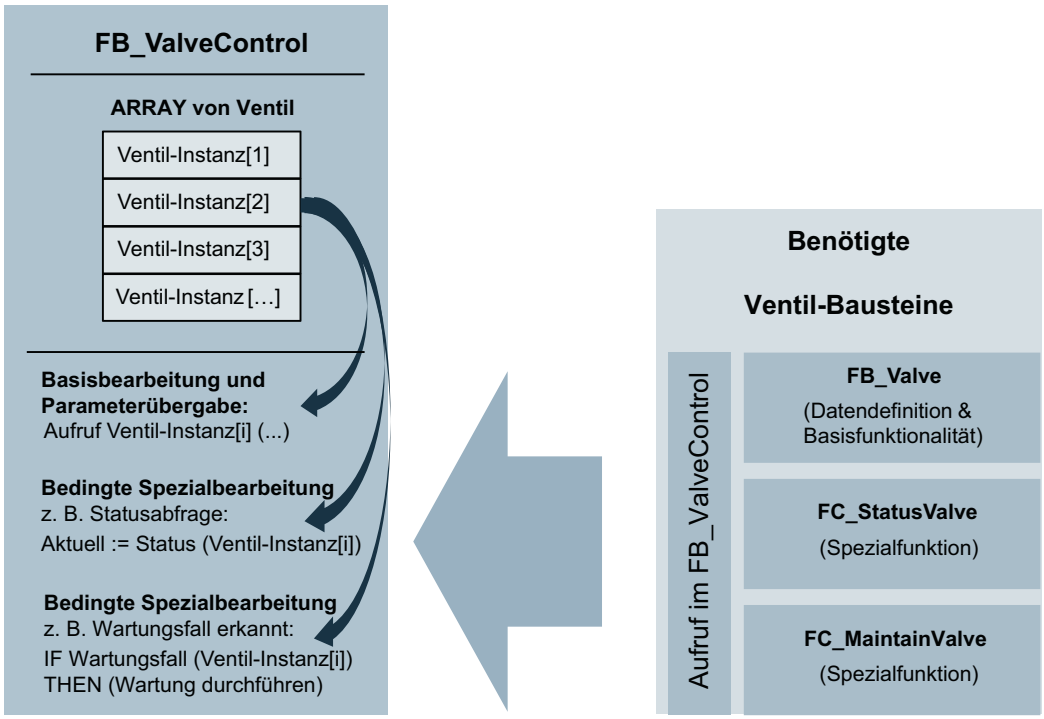

- Die beiden Funktionen "FC_StatusValve" und "FC_MaintainValve" bearbeiten eine Parameterinstanz des Programmbausteins "FB_Valve", die sie beim Aufruf übergeben bekommen haben.
- Im Programmbaustein "FB_ValveControl" nutzen Sie das ARRAY von Multiinstanzen, um zum einen die Anzahl der vorhandenen Ventile zu deklarieren und zum anderen, um alle Ventile in einer Programmschleife mittels verschiedener Funktionen zu bearbeiten.

Hinweis

Vollständigkeit

Diese Lösung dient nur als Beispiel, um die Herangehensweise an diese Aufgabenstellung zu veranschaulichen. Bitte bedenken Sie, dass Sie den Programmcode jeweils an Ihre tatsächliche Aufgabe anpassen müssen.

Für die Umsetzung dieses Beispiels werden die folgenden Programmbausteine benötigt:

Baustein	Beschreibung	Programmiersprache
FB_ValveControl	Dieser Funktionsbaustein dient als Steuerungsbaustein, in dem Sie alle Ventile mithilfe eines ARRAY von Multiinstanzen bearbeiten.	SCL
FB_Valve	Dieser Funktionsbaustein enthält die Definition der Ventildaten und den Programmcode zur Bearbeitung eines Ventils.	SCL (Die drei Programmbausteine werden vom Programmbaustein "FB_ValveControl" verwendet.)
FC_StatusValve	Die Funktion gibt den Status des aktuell bearbeiteten Ventils aus.	
FC_MaintainValve	Diese Funktion prüft einen Wartungsbedarf am Ventil. Bei Bedarf wird die Wartung ausgeführt und der Funktionswert TRUE geliefert, wenn alles fertiggestellt wurde.	

Vorgehen: "FB_Valve" anlegen

Um den SCL-Funktionsbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktionsbaustein (FB)".
3. Geben Sie den Namen "FB_Valve" an.
4. Wählen Sie als Sprache "SCL" aus.
5. Klicken Sie auf "OK".

6. Die Deklaration der Bausteinschnittstelle orientiert sich an einem Ventil, das Sie öffnen und schließen können, und könnte z. B. folgendermaßen aussehen:

FB_Valve

	Name	Data type	Comment
1	Input		
2	R_max_A_Valve	Real	Maximum valve performance in mA
3	period_duration_factor	Int	Duration = "period_duration_factor" x 0.1 x configured duration
4	MANUAL_DO	Bool	Check output manually
5	SET_DO	Bool	Set output
6	Output		
7	I_actual_A	Real	Actual valve performance in mA
8	Current_Limit	Bool	Maximum valve performance exceeded (< 4A)
9	InOut		
10	SW_ENABLE	Bool	When this tag has the signal state TRUE, maintenance is required.
11	Static		
12	I_actual_mil	Int	The actual value of the program block "ValveControl", that is queried every millisecond.
13	control_byte	Byte	Control Byte: Bit0=SW_ENABLE; Bit1=MANUAL_DO; Bit2=SET_DO

7. Der Programmcode für die Steuerung der Ventile könnte z. B. folgendermaßen aussehen:

```
1  #I_actual_mil := #period_duration_factor;
2  #SW_ENABLE := #control_byte.%X4;
```

Vorgehen: "FC_StatusValve" anlegen

Um die SCL-Funktion "FC_StatusValve" anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie eauf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktion (FC)".
3. Geben Sie den Namen "FC_StatusValve" an.
4. Wählen Sie als Sprache "SCL" aus.
5. Klicken Sie auf "OK".
6. Die Deklaration der Bausteinschnittstelle mit Einbindung der Parameterinstanz "FB_Valve" könnte z. B. folgendermaßen aussehen:

FC_StatusValve

	Name	Data type	Comment
1	Input		
2	Output		
3	InOut		
4	CurrentValve	"FB_Valve"	The function block "FB_Valve" is used as a parameter instance to exchange its data with the "FB_Valve".
5	<Add new>		
6	Temp		
7	Constant		
8	Return		
9	FC_StatusValve	Byte	Return value of the function

7. Schreiben Sie z. B. folgenden Programmcode:

```
1  #FC_StatusValve := #CurrentValve.control_byte;
```

Vorgehen: "FC_MaintainValve" anlegen

Um die SCL-Funktion "FC_MaintainValve" anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie eauf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktion (FC)".
3. Geben Sie den Namen "FC_MaintainValve" an.
4. Wählen Sie als Sprache "SCL" aus.
5. Klicken Sie auf "OK".
6. Die Deklaration der Bausteinschnittstelle mit Einbindung der Parameterinstanz "FB_Valve" könnte z. B. folgendermaßen aussehen:

FC_MaintainValve

	Name	Data type	Comment
1	Input		
2	NewValue	Int	Specifies a new value for the duration of the current valve.
3	Output		
4	InOut		
5	CurrentValve	"FB_Valve"	The function block "FB_Valve" is used as a parameter instance to exchange its data with the "FB_Valve".
6	Temp		
7	Constant		
8	Return		
9	FC_MaintainValve	Bool	Return value of the function

7. Schreiben Sie z. B. folgenden Programmcode:

```
1  IF #CurrentValve.SW_ENABLE THEN // Valve control enabled
2      // Do some tuning and maintenance
3      #CurrentValve.period_duration_factor := #NewValue;
4      #CurrentValve.control_byte.%X0 := true;
5      #FC_MaintainValve := true;
6  ELSE
7      // Tuning currently not allowed
8      #FC_MaintainValve := false;
9      ;
10 END_IF;
```

Vorgehen: "FB_ValveControl" anlegen

Um den SCL-Funktionsbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktionsbaustein (FB)".
3. Geben Sie den Namen "FB_ValveControl" an.
4. Wählen Sie als Sprache "SCL" aus.
5. Klicken Sie auf "OK".

6. Definieren Sie die Anwenderkonstante:

		Name	Data type	Value	Comment
1		vmax	Int	99	Number of the valves

Die Anwenderkonstante "vmax" wird in diesem Beispiel verwendet, um das Programm einfach an eine unterschiedliche Anzahl von Ventilen anpassen zu können.

7. Die Deklaration der Bausteinschnittstelle mit Einbindung der Ventil-Instanzen von "FB_Valve" könnte z. B. folgendermaßen aussehen:

FB_ValveControl

		Name	Data type	Comment
1		Input		
2		ValueMaintain	Int	Maintenance
3		Duration	Int	Duration of the valve control
4		Output		
5		Bypass	Bool	True or false
6		InOut		
7		Index	Int	Number index
8		Static		
9		ValveInstances	Array[0.."vmax"] of "FB_Valve"	Multiple instances of the valve
10		ValveStatus	Array[0.."vmax"] of Byte	Status of the valve
11		CurrentAmpere	Real	Current number of amperes

8. Schreiben Sie z. B. folgenden Programmcode:

```
1  // Set #Index, given from outside, to a value that adresses an existing valve.
2  #Index := (#Index + 1) MOD ("vmax" + 1);
3
4  // Indirect call of a valve that is addressed via an ARRAY index.
5  #ValveInstances[#Index](R_max_A_Valve := 1.0,
6                          period_duration_factor := #Duration,
7                          MANUAL_DO := 0,
8                          SET_DO := 0,
9                          I_actual_A => #CurrentAmpere,
10                         Current_Limit => #Bypass);
11
12 // Do some checks and tuning on the valve.
13 // FC_MaintainValve sets some values of the passed valve.
14 // FC_StatusValve returns the current status that is stored as internal value in the Instance.
15 IF "FC_MaintainValve"(CurrentValve := #ValveInstances[#Index],
16                       NewValue := #ValueMaintain) THEN
17     #ValveStatus[#Index] := #ValveStatus[#Index] XOR "FC_StatusValve"(CurrentValve := #ValveInstances[#Index]);
18 END_IF;
```

Vorgehen: Aufruf des "FB_ValveControl" im OB 1

Um den Funktionsbaustein "FB_ValveControl" im OB 1 aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein "Main (OB1)" mit einem Doppelklick.
2. Fügen Sie den Funktionsbaustein "FB_ValveControl" per Drag & Drop in den OB 1 ein.

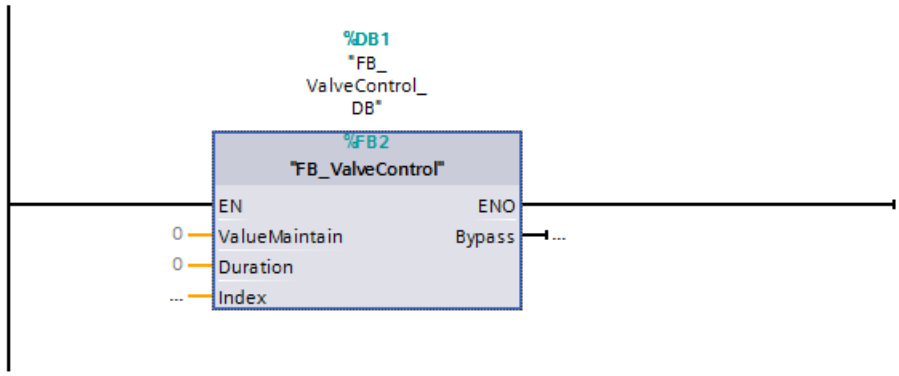

Ergebnis

Pro Programmzyklus wird jeweils ein Ventil bearbeitet. Das jeweils mittels Index angewählte Ventil wird bearbeitet, der Status abgefragt und bei Bedarf eine Wartung durchgeführt.

In diesem Beispiel wurde ein Ventil pro Zyklus bearbeitet. Es ist natürlich auch möglich, alle Ventile in einer Programmschleife zu bearbeiten. Dann steigt aber die Zykluszeit entsprechend. Wenn Sie einen kurzen Programmzyklus benötigen, dann ist es empfehlenswert, nicht mit einer Programmschleife zu arbeiten.

Siehe auch

Parameterinstanzen (Seite 70)

Multiinstanzen (Seite 67)

1.3.4 Parameterübergabe beim Bausteinaufruf

1.3.4.1 Regeln zur Versorgung von Bausteinparametern

Bausteinparameter

Der aufrufende Baustein gibt dem aufgerufenen Baustein Werte mit, mit denen er arbeiten soll. Diese Werte werden als Bausteinparameter bezeichnet. Über die Eingangsparameter erhält der aufgerufene Baustein die Werte, die er verarbeiten soll. Die Ergebnisse liefert der Baustein über die Ausgangsparameter zurück.

Bausteinparameter stellen also die Schnittstelle zwischen dem aufrufenden und dem aufgerufenen Baustein dar.

Sie verwenden Eingangsparameter, wenn Sie Werte nur abfragen oder lesen wollen, und Ausgangsparameter, wenn Sie diese nur setzen oder schreiben wollen. Werden Bausteinparameter gelesen und geschrieben, müssen Sie sie als Durchgangsparameter anlegen.

Für die Verwendung der Bausteinparameter innerhalb des Bausteins gelten die folgenden Regeln:

- Eingangsparameter dürfen nur gelesen werden.
- Ausgangsparameter dürfen nur geschrieben werden.
- Durchgangsparameter dürfen gelesen und geschrieben werden.

Formal- und Aktualparameter

Die Bausteinparameter sind in der Schnittstelle des aufgerufenen Bausteins definiert. Diese Parameter werden als Formalparameter bezeichnet. Sie sind Platzhalter für die Werte, die dem Baustein beim Aufruf übergeben werden. Die beim Aufruf übergebenen Werte werden als Aktualparameter bezeichnet.

Die Datentypen von Aktual- und Formalparameter müssen gleich oder nach den Regeln der Datentypkonvertierung konvertierbar sein.

Hinweis

S7-1200/1500: Peripherieeingänge oder -ausgänge als Bausteinparameter übergeben

Wenn Sie einen Eingangsparameter mit einem Peripherieeingang oder -ausgang versorgen, besteht das Risiko, dass zur Laufzeit Peripheriezugriffsfehler beim Bausteinaufruf auftreten, z. B. ein Lesefehler beim direkten Zugriff auf Daten einer Eingabebaugruppe.

Das Systemverhalten bei CPUs der Baureihen S7-1500 mit FW-Version ab V2.1 ist wie folgt:

Der Baustein wird aufgerufen und mit dem Ersatzwert des Signals bearbeitet.

Das Systemverhalten bei CPUs der Baureihen S7-1200 und S7-1500 mit FW-Version kleiner V2.1 ist wie folgt:

Der Baustein wird aufgrund des Peripheriezugriffsfehlers nicht aufgerufen. Die Programmausführung wird nach dem Bausteinaufruf fortgesetzt. Wenn ein OB 122 vorhanden ist, oder die lokale Fehlerbehandlung aktiviert ist, werden diese ausgeführt.

Um zu verhindern, dass ein Bausteinaufruf bei einem Peripheriezugriffsfehler nicht ausgeführt wird, kopieren Sie den Peripherieeingang oder -ausgang zunächst in eine lokale Variable (Temp) und übergeben diese als Bausteinparameter an den aufgerufenen Baustein.

Der folgende FAQ gibt weitere Informationen zu diesem Thema:

FAQ 89377245: Warum werden Bausteine in der S7-1200/S7-1500 nicht bearbeitet, wenn Sie sie mit nicht vorhandenen PROFINET-Komponenten adressieren? (https://support.industry.siemens.com/cs/ww/de/view/89377245)

Siehe auch

Variablen (Seite 100)

Parameterversorgung von Funktionsbausteinen (Seite 83)

Parameterversorgung von Funktionen (Seite 80)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

PLC-Datentypen (UDT) (Seite 286)

Grundlagen zu AWL (Seite 7557)

1.3.4.2 Parameterversorgung von Funktionen

Parameter von Funktionen (FC)

Funktionen haben keinen Datenspeicher, in denen Werte von Bausteinparametern gespeichert werden könnten. Deshalb müssen beim Aufruf einer Funktion alle Formalparameter mit Aktualparametern versorgt werden.

Eingangsparameter (Input)

Eingangsparameter werden nur einmalig vor jedem Bausteinaufruf gelesen. Deshalb gilt die Regel, dass das Schreiben eines Eingangsparameters innerhalb des Bausteins sich nicht auf den Aktualparameter auswirkt. Es wird nur der Formalparameter geschrieben.

Ausgangsparameter (Output)

Ausgangsparameter werden nur einmalig nach jedem Bausteinaufruf geschrieben. Deshalb gilt die Regel, dass Ausgangsparametern innerhalb des Bausteins nicht gelesen werden sollten. Wenn Sie dennoch einen Ausgangsparameter lesen, beachten Sie, dass dabei nur der Wert des Formalparameters gelesen wird. Der Wert des Aktualparameters kann innerhalb des Bausteins nicht gelesen werden.

Wenn ein Ausgangsparameter einer Funktion in dieser Funktion nicht geschrieben wird, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Strukturierte Ausgangsparameter werden jedoch nicht mit einem Wert vorbelegt.

Um nicht ungewollt den vordefinierten Wert oder einen undefinierten Wert weiterzuverarbeiten, beachten Sie bei der Programmierung des Bausteins Folgendes:

- Achten Sie darauf, dass die Ausgangsparameter für jeden möglichen Programmpfad innerhalb des Bausteins mit Werten beschrieben werden. Beachten Sie hierbei z. B., dass Sprungbefehle möglicherweise über Anweisungssequenzen springen, in denen Ausgänge gesetzt werden.
- Beachten Sie, dass Setze- und Rücksetze-Befehle vom Verknüpfungsergebnis abhängig sind. Wird der Wert eines Ausgangsparameters mit diesen Befehlen ermittelt, wird bei VKE = 0 kein Wert gebildet.
- Vergeben Sie für Ausgangsparameter von Funktionen wenn möglich einen Defaultwert.

Durchgangsparameter (InOut)

Durchgangsparameter werden vor dem Bausteinaufruf gelesen und nach dem Bausteinaufruf geschrieben. Wenn Sie den Parameter innerhalb des Bausteins lesen oder schreiben, greifen Sie nur auf dessen Formalparameter zu.

Eine Ausnahme bilden die Durchgangsparameter mit strukturiertem Datentyp. Strukturierte Datentypen sind die Datentypen, die aus mehreren Datenelementen bestehen, z. B. ARRAY oder STRUCT. Diese werden durch einen Zeiger an den aufgerufenen Baustein übergeben. Sie greifen also immer auf den Aktualparameter zu, wenn Sie einen strukturierten Durchgangsparameter innerhalb eines Bausteins lesen oder schreiben.

Wenn ein Durchgangsparameter einer Funktion in dieser Funktion nicht geschrieben wird, wird der alte Ausgangswert bzw. der Eingangswert als Wert verwendet. Dennoch sollten Sie auch hier die oben für Ausgangsparameter gegebenen Hinweise beachten, um nicht unbeabsichtigt alte Werte weiter zu verarbeiten.

Temporäre Lokaldaten (Temp)

Temporäre Lokaldaten stehen nur während der Bausteinbearbeitung zur Verfügung. Sie werden abhängig von der Optimierungseinstellung des Bausteins unterschiedlich behandelt:

- Standardzugriff
  In Codebausteinen mit Standardzugriff sowie für alle Variablen mit der Remanenzeinstellung "Im IDB setzen" gilt die folgende Regel:
  Wenn Sie temporäre Lokaldaten verwenden, müssen Sie sicherstellen, dass die Werte vor der Verwendung initialisiert werden. Ansonsten sind die Werte zufällig. Eine Ausnahme sind temporäre Daten vom Datentyp STRING oder WSTRING: Sie werden automatisch mit der tatsächlichen Länge 0 vorbelegt.
- Optimierter Zugriff
  In Codebausteinen mit optimiertem Zugriff gelten folgende Regeln:
  - Wenn eine temporäre Variable innerhalb einer Funktion nicht geschrieben wird, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist.
    Die folgende Tabelle zeigt einige Beispiele für vordefinierte Werte:

Datentyp	vordefinierter Wert
Bool	false
Int	0
REAL	0.0
Char	' '
Wchar	WCHAR#' '
DTL	DTL#1970-01-01-00:00:00
Date_And_Time	DT#1990-01-01-00:00:00
Date	D#1990-01-01

  - Elemente von PLC-Datentypen werden mit dem Defaultwert vorbelegt, der in der Deklaration des PLC-Datentyps (UDT) angegeben ist.
  - STRINGs und WSTRINGs werden mit der tatsächlichen Länge "0" vorbelegt, auch wenn Sie innerhalb eines PLC-Datentyps verwendet werden.
  - Elemente vom Datentyp ARRAY werden mit dem Wert "0" vorbelegt, auch wenn sie innerhalb eines PLC-Datentyps verwendet werden.

Funktionswert (Return)

Funktionen berechnen üblicherweise einen Funktionswert. Dieser Funktionswert kann über den Ausgangsparameter RET_VAL an den aufrufenden Baustein zurückgeliefert werden. Dazu muss der Ausgangsparameter RET_VAL in der Schnittstelle der Funktion deklariert sein. RET_VAL ist immer der erste Ausgangsparameter einer Funktion. Für den Parameter RET_VAL sind alle Datentypen zulässig mit Ausnahme von ARRAY und STRUCT sowie der Parametertypen TIMER und COUNTER.

In der Programmiersprache SCL können Funktionen direkt in einem Ausdruck aufgerufen werden. Das Ergebnis des Ausdrucks wird dann mit dem berechneten Funktionswert gebildet. In SCL ist deshalb der Datentyp ANY für den Funktionswert nicht zulässig.

Siehe auch

Parameterversorgung von Funktionsbausteinen (Seite 83)

Regeln zur Versorgung von Bausteinparametern (Seite 78)

Aufruf von Funktionen (Seite 7661)

Beispiele zum Aufruf von Funktionen in SCL (Seite 7664)

1.3.4.3 Parameterversorgung von Funktionsbausteinen

Versorgung der Parameter von Funktionsbausteinen (FB)

Bei Funktionsbausteinen werden die Parameterwerte in den Instanzdaten gespeichert.

Sind Ein-, Aus- oder Durchgangsparameter eines Funktionsbausteins nicht mit aktuellen Werten versorgt, werden die gespeicherten Werte verwendet, wenn der Parameter bereits in einem vorherigen Zyklus einmalig initialisiert wurde. Ist dies nicht der Fall, geht die CPU mit einem Laufzeitfehler in STOP.

In einigen Fällen ist die Angabe eines Aktualparameters zwingend erforderlich.

Die folgende Tabelle zeigt, welchen Parametern eines Funktionsbausteins Aktualparameter zugeordnet werden müssen:

Parameter	Elementarer Datentyp	Strukturierter Datentyp	Parametertyp
Eingang (Input)	optional	optional	optional
Ausgang (Output)	optional	optional	nicht zulässig
Durchgang (InOut)	optional	• Aufruf eines nicht-optimierten-Bausteins: nicht unbedingt erforderlich, aber der Wert muss durch einen einmaligen Aufruf initialisiert werden, sonst wird ein Laufzeitfehler ausgegeben. • Aufruf von optimierten Bausteinen: erforderlich	nicht zulässig

Siehe auch

Regeln zur Versorgung von Bausteinparametern (Seite 78)

Parameterversorgung von Funktionen (Seite 80)

Parametertypen (Seite 363)

1.3.4.4 Parameter als Kopie oder als Pointer übergeben

Einführung

Beim Bausteinaufruf übergeben Sie den Parametern in der Schnittstelle des Bausteins Daten. An den Eingangsparametern (Input) übergeben Sie die Daten, mit denen der Baustein arbeiten soll. An den Ausgangsparametern (Output) legen Sie fest, wo die Bearbeitungsergebnisse abgelegt werden. Durchgangsparameter (InOut) dienen sowohl der Übergabe von Daten an den aufgerufenen Baustein als auch der Rückgabe von Ergebnissen.

Intern kennt STEP 7 zwei unterschiedliche Methoden der Parameterübergabe: Abhängig von Übergabebereich und vom Datentyp des Parameters werden die Daten entweder als Pointer oder als Kopie übergeben.

Übergabe als Kopie (Call by value)

Beim Bausteinaufruf wird der Wert des Operanden auf den Eingangsparameter des aufgerufenen Bausteins kopiert. Bei Funktionsbausteinen wird die Kopie im Instanz-DB abgelegt, bei Funktionen im Baustein-Stack. Für die Kopie wird zusätzlicher Speicherplatz benötigt.

Das bedeutet, dass der aufgerufene Baustein immer mit dem Wert arbeitet, den der angegebene Operand zum Zeitpunkt des Bausteinaufrufs hatte. Er kann nicht direkt auf den Operanden zugreifen. Schreibende Zugriffe verändern nur die Kopie, nicht aber den tatsächlichen Wert des angegebenen Operanden. Lesende Zugriffe lesen nur die Kopie, die zum Zeitpunkt des Bausteinaufrufs angelegt wurde.

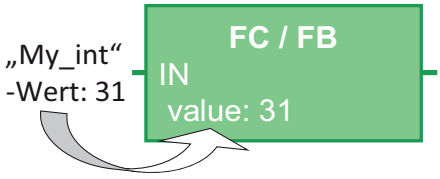

Übergabe als Pointer (Call by reference)

Die Parameter werden beim Bausteinaufruf über einen Pointer referenziert.

Das bedeutet, dass der aufgerufene Baustein unmittelbar auf die Speicheradresse des Operanden zugreift, der als Parameter angegeben ist: Schreibende Zugriffe führen direkt zur Veränderung des angegebenen Operanden. Lesende Zugriffe lesen den Wert des Operanden unmittelbar zum Zeitpunkt des Zugriffs. Da keine Kopien angelegt werden, wird kein zusätzlicher Speicher benötigt.

Hinweis

Deklarieren Sie strukturierte Datentypen im Bereich "InOut"

Nutzen Sie für strukturierte Variablen (z. B. vom Datentyp ARRAY, STRUCT, STRING, …) wenn möglich den Bereich "InOut" in der Bausteinschnittstelle. Da strukturierte Durchgangsparameter (InOut) standardmäßig als Pointer übergeben werden, wird der benötigte Datenspeicher so nicht unnötig vergrößert.

Parameterübergabe bei S7-1200/1500

Die folgende Tabelle zeigt, wie in S7-1200/1500 Bausteinparameter mit elementarem bzw. strukturiertem Datentyp übergeben werden. Elementare Datentypen sind z. B. BOOL, INT oder BYTE. Strukturierte Datentypen sind z. B. ARRAY, STRUCT oder STRING.

		Elementare Datentypen	Strukturierte Datentypen
FC	Input	Kopie	Pointer
	Output	Kopie	Pointer
	InOut	Kopie	Pointer
FB	Input	Kopie	Kopie
	Output	Kopie	Kopie
	InOut	Kopie	Pointer

Hinweis

Parameterübergabe zwischen Bausteinen mit optimiertem Zugriff und Bausteinen mit Standardzugriff

Wenn bei einem Bausteinaufruf Strukturen als Durchgangsparameter (InOut) an den aufgerufenen Baustein übergeben werden, werden diese standardmäßig als Pointer übergeben (Call by reference).

Dies gilt jedoch nicht, wenn beide Bausteine unterschiedliche Optimierungseinstellungen haben: Wenn einer der Bausteine die Eigenschaft "Optimierter Zugriff" und der andere Baustein die Eigenschaft "Standardzugriff" hat, werden grundsätzlich alle Parameter als Kopie übergeben (Call by value).

Wenn der Baustein viele strukturierte Parameter enthält, kann das schnell dazu führen, dass der temporäre Speicherbereich (Lokaldaten-Stack) des Bausteins überläuft.

Außerdem können Probleme entstehen, wenn die ursprünglichen Operanden durch asynchrone Prozesse verändert werden, z. B. durch HMI-Zugriffe oder durch Alarm-OBs. Wenn nach der Bausteinbearbeitung die Kopien wieder auf die ursprünglichen Operanden zurückkopiert werden, werden dabei die seit dem Anlegen der Kopie beim Bausteinaufruf asynchron durchgeführten Änderungen an den ursprünglichen Operanden überschrieben.

Das können Sie vermeiden, indem Sie für beide Bausteine dieselbe Zugriffsart einstellen oder die asynchronen Zugriffe zuerst in einen separaten Speicherbereich schreiben lassen und diesen Bereich dann zu einem geeigneten Zeitpunkt synchron kopieren.

Siehe auch:

Bausteine mit optimiertem Zugriff (Seite 55)

FAQ 109478253: Warum kann es zum Überschreiben von Daten des HMI-Systems oder des Webservers in der S7-1500 kommen? (https://support.industry.siemens.com/cs/de/de/view/109478253)

Parameterübergabe bei S7-300/400

Die folgende Tabelle zeigt, wie in S7-300/400 Bausteinparameter mit elementarem bzw. strukturiertem Datentyp übergeben werden.

		Elementare Datentypen	Strukturierte Datentypen
FC	Input	Kopie*	Pointer
	Output	Kopie*	Pointer
	InOut	Kopie*	Pointer

		Elementare Datentypen	Strukturierte Datentypen
FB	Input	Kopie	Kopie
	Output	Kopie	Kopie
	InOut	Kopie	Pointer

* Ausnahme: Operanden aus den Speicherbereichen E, A, M, P, L und teilqualifizierte DB-Adressen (z. B. "DW 2") werden als Pointer übergeben.

Hinweis

Besonderheiten bei der Übergabe als Pointer in S7-300/400

In den Fällen, in denen die Parameter über einen Pointer übergeben werden, ist es nicht möglich, Ausgangs- oder Durchgangsparameter vom aufrufenden Baustein an die Eingangsparameter des aufgerufenen Bausteins weiterzureichen.

1.3.4.5 Weiterreichen von Bausteinparametern

Grundlagen zum Weiterreichen von Bausteinparametern

Definition

Eine besondere Art der Parameterverwendung ist das "Weiterreichen" von Bausteinparametern. Dabei werden die Bausteinparameter des aufrufenden Bausteins an die Parameter des aufgerufenen Bausteins weitergereicht. Der aufgerufene Baustein verwendet als Aktualparameter die Werte, die aktuell an den Bausteinparametern des aufrufenden Bausteins anstehen.

Das folgende Bild zeigt, wie die Parameter der Funktion FC 10, an die Funktion FC 12 weitergereicht werden:

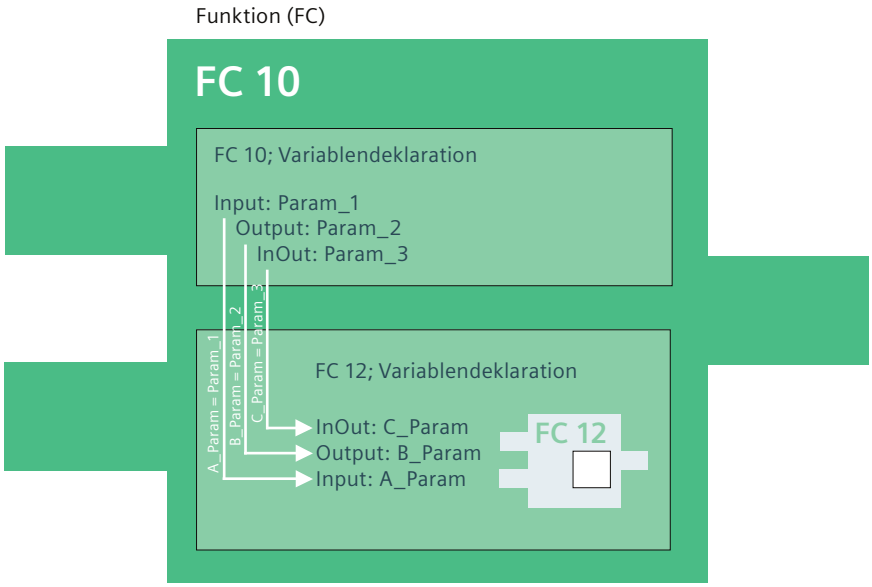

Regeln für KOP/FUP

In KOP und FUP gelten folgende allgemeine Regeln:

- Eingangsparameter können nur an Eingangsparameter weitergereicht werden.
- Ausgangsparameter können nur an Ausgangsparameter weitergereicht werden.
- Durchgangsparameter können an alle Parameterarten weitergereicht werden.
- In S7-300/400 müssen beide Bauteinparameter vom gleichen Datentyp sein.
- In S7-1200/1500 können die Parameter auch nach den Regeln der impliziten Konvertierung konvertiert werden.

Regeln für AWL

In AWL gelten folgende allgemeine Regeln:

- Eingangsparameter können nur an Eingangsparameter weitergereicht werden.
- Ausgangsparameter können nur an Ausgangsparameter weitergereicht werden.
- Durchgangsparameter können an alle Parameterarten weitergereicht werden.
- Beide Bauteinparamter müssen vom gleichen Datentyp sein. Diese Regel gilt bei AWL in allen CPU-Familien.

Regeln für SCL

Für SCL gelten weniger strenge Regeln. Damit Programme aus früheren SCL-Versionen leichter übernommen werden können, sind weitere Möglichkeiten der Parameterübergabe zulässig. Sie können beispielsweise einen strukturierten Durchgangsparameter an einen Eingangsparameter weiterreichen. Stellen Sie in diesem Fall jedoch sicher, dass der Aktualparameter nicht in den temporären Lokaldaten oder einem globalen Datenbaustein liegt.

Weitere Regeln sind detailliert in den nachfolgenden Kapiteln beschrieben.

Siehe auch

Aufruf einer Funktion durch eine andere Funktion (Seite 89)

Aufruf einer Funktion durch einen Funktionsbaustein (Seite 90)

Aufruf eines Funktionsbausteins durch eine Funktion (Seite 91)

Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein (Seite 92)

Aufruf einer Funktion durch eine andere Funktion

Zulässige Datentypen beim Aufruf einer Funktion durch eine andere Funktion

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

FC ruft FC		Datentypen						
Aktualparameter (aufrufender Baustein)	Formalparameter (aufgerufener Baustein)	Standard-Datentypen	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Parameterty-pen (TIMER, COUNTER, BLOCK_XX)	DB_Any	REF_TO
Input	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 ab V2 S7-1500	S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-	S7-1500
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-	-

Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 87)

Aufruf einer Funktion durch einen Funktionsbaustein

Zulässige Datentypen beim Aufruf einer Funktion durch einen Funktionsbaustein

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

FB ruft FC		Datentypen						
Aktualparameter (aufrufender Baustein)	Formalparameter (aufgerufener Baustein)	Standard-Datentypen	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Parametertypen (TIMER, COUNTER, BLOCK_XX)	DB_Any	REF_TO
Input	Input	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 ab V2 S7-1500	S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	-	S7-1200 S7-1500	-	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-	S7-1500
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-	-

Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 87)

Aufruf eines Funktionsbausteins durch eine Funktion

Zulässige Datentypen beim Aufruf eines Funktionsbausteins durch eine Funktion

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

FC ruft FB		Datentypen					
Aktualparameter (aufrufender Baustein)	Formalparameter (aufgerufener Baustein)	Standard-Datentypen	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Parametertypen (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-300/400 S7-1500	S7-1200 ab V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 87)

Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein

Zulässige Datentypen beim Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

FB ruft FB		Datentypen					
Aktualparameter (aufrufender Baustein)	Formalparameter (aufgerufener Baustein)	Standard-Datentypen	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Parametertypen (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-300/400 S7-1500	S7-1200 ab V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 87)

1.4 Operanden verwenden und adressieren

1.4.1 Grundlagen zu Operanden

Einführung

Bei der Programmierung von Anweisungen müssen Sie angeben, welche Datenwerte die Anweisung verarbeiten soll. Diese Werte werden als Operanden bezeichnet. Sie können z. B. folgende Elemente als Operanden verwenden:

- PLC-Variablen
- Konstanten
- Variablen in Instanzdatenbausteinen
- Variablen in globalen Datenbausteinen

Absolute Adresse und symbolischer Name

Operanden werden über eine absolute Adresse und einen symbolischen Namen identifiziert. Die Namen und Adressen definieren Sie in der PLC-Variablentabelle oder in der Variablendeklaration der Bausteine.

Datenbausteine mit optimiertem Zugriff (S7-1200, S7-1500)

Datenelemente in Datenbausteinen mit optimiertem Zugriff erhalten in der Deklaration nur einen symbolischen Namen, keine absolute Adresse.

Siehe auch

Symbolische und absolute Operanden anzeigen (Seite 7131)

Grundlagen zum Bausteinzugriff (Seite 55)

1.4.2 Schlüsselwörter

SIMATIC kennt eine Reihe fest definierter Schlüsselwörter, die im Programm eine bestimmte Bedeutung haben. Sie sollten diese Schlüsselwörter nicht als Namen für Variablen oder Konstanten verwenden.

Tabelle der Schlüsselwörter

Die folgende Tabelle zeigt alle Schlüsselwörter.

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
&	&	Und-Verknüpfung logischer Ausdrücke
^	^	Operator zum Lesen oder Schreiben des Wertes einer Referenz
A	Q	Ausgang, Bit
A1	CC1	Anzeigebit
A0	CC0	Anzeigebit
AB	QB	Ausgang, Byte
AD	QD	Ausgang, Doppelwort
AND	AND	Und-Verknüpfung logischer Ausdrücke
ANY	ANY	Datentyp, Zeiger
AR1	AR1	Adressregister 1
AR2	AR2	Adressregister 2
ARRAY	ARRAY	Einleitung der Spezifikation eines Arrays, danach folgt zwischen "[" und "]" die Indexliste
AT	AT	Überlagernde Variablendeklaration
AUTHOR	AUTHOR	Name des Autors, Firmenname, Abteilungsname oder andere Namen (max. 8 Zeichen, ohne Leerzeichen)
AW	QW	Ausgang, Wort
B	B	Byte
BEGIN	BEGIN	Einleitung Anweisungsteil bei Codebausteinen oder Initialisierungsteil bei Datenbaustein
BIE	BR	Binärergebnis
BLOCK_FB	BLOCK_FB	Parametertyp zur Angabe eines FB
BLOCK_FC	BLOCK_FC	Parametertyp zur Angabe einer FC
BLOCK_SDB	BLOCK_SDB	Parametertyp zur Angabe eines SDB
BOOL	BOOL	Datentyp
BY	BY	Schrittweite der FOR-Schleife
BYTE	BYTE	Datentyp
CALL	CALL	Aufruf
CASE	CASE	Einleitung der CASE-Anweisung
CHAR	CHAR	Elementarer Datentyp

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
CODE_VERSION1	CODE_VERSION1	Kennung, ob ein FB multiinstanzfähig ist oder nicht. Wenn Sie Multiinstanzen deklarieren möchten, darf der FB diese Eigenschaft nicht haben.
CONST	CONST	Beginn der Kostantendeklaration
CONTINUE	CONTINUE	Anweisung zum Verlassen einer Schleife in SCL
COUNTER	COUNTER	Parametertyp zur Angabe eines Zählers
DATA_BLOCK	DATA_BLOCK	Einleitung des Datenbausteins
DATE	DATE	Datentyp
DATE_AND_TIME	DATE_AND_TIME	Datentyp
DB	DB	Datenbaustein
DB_ANY	DB_ANY	Datentyp
DBB	DBB	Datenbaustein, Datenbyte
DBD	DBD	Datenbaustein, Datendoppelwort
DBLG	DBLG	Datenbausteinlänge
DBNO	DBNO	Datenbausteinnummer
DBW	DBW	Datenbaustein, Datenwort
DBX	DBX	Datenbaustein, Datenbit
DI	DI	Instanz-Datenbaustein
DIB	DIB	Instanz-Datenbaustein, Datenbyte
DID	DID	Instanz-Datenbaustein, Datendoppelwort
DILG	DILG	Instanzdatenbausteinlänge
DINO	DINO	Instanzdatenbausteinnummer
DINT	DINT	Datentyp
DIW	DIW	Instanz-Datenbaustein, Datenwort
DIX	DIX	Instanz-Datenbaustein, Datenbit
DO	DO	Einleitung des Anweisungsteils in FOR- und WHILE-Anweisung
DT	DT	Datentyp
DTL	DTL	Datentyp
DWORD	DWORD	Datentyp
E	I	Eingang (über Prozessabbild), Bit
EB	IB	Eingang (über Prozessabbild), Byte
ED	ID	Eingang (über Prozessabbild), Doppelwort

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
ELSE	ELSE	Alternativer Zweig in IF- und CASE-Anweisung
ELSIF	ELSIF	Alternative Bedingung der IF-Anweisung
EN	EN	Systemoperand des EN/ENO-Mechanismus
ENO	ENO	Systemoperand des EN/ENO-Mechanismus
END_CASE	END_CASE	Abschluss der CASE-Anweisung
END_DATA_BLOCK	END_DATA_BLOCK	Abschluss des Datenbausteins
END_FOR	END_FOR	Abschluss der FOR-Anweisung
END_FUNCTION	END_FUNCTION	Abschluss der Funktion
END_FUNCTION_BLOCK	END_FUNCTION_BLOCK	Abschluss des Funktionsbausteins
END_IF	END_IF	Abschluss der IF-Anweisung
END_ORGANIZATION_BLOCK	END_ORGANIZATION_BLOCK	Abschluss des Organisationsbausteins
END_REGION	END_REGION	Abschluss eines Bereichs
END_REPEAT	END_REPEAT	Abschluss der REPEAT-Anweisung
END_STRUCT	END_STRUCT	Abschluss der Spezifikation einer Struktur
END_SYSTEM_FUNCTION	END_SYSTEM_FUNCTION	Abschluss der Systemfunktion
END_SYSTEM_FUNCTI-ON_BLOCK	END_SYSTEM_FUNCTI-ON_BLOCK	Abschluss des Systemfunktionsbausteins
END_TYPE	END_TYPE	Abschluss des PLC-Datentyps
END_VAR	END_VAR	Abschluss eines Deklarationsblocks
END_WHILE	END_WHILE	Abschluss der WHILE-Anweisung
EW	IW	Eingang (über Prozessabbild), Wort
EXIT	EXIT	Anweisung zum Verlassen einer Schleife in SCL
FALSE	FALSE	Vordefinierte boolesche Konstante: Logische Bedingung nicht erfüllt, Wert gleich 0
FAMILY	FAMILY	Name der Bausteinfamilie: z. B. Regler
FB	FB	Funktionsbaustein
FC	FC	Funktion
FOR	FOR	Einleitung der FOR-Anweisung
FUNCTION	FUNCTION	Einleitung der Funktion
FUNCTION_BLOCK	FUNCTION_BLOCK	Einleitung des Funktionsbausteins
GOTO	GOTO	Einleitung der GOTO-Anweisung

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
IF	IF	Einleitung der IF-Anweisung
INSTANCE	INSTANCE	Datentyp
INT	INT	Datentyp
KNOW_HOW_PROTECT	KNOW_HOW_PROTECT	Bausteinschutz
L	L	Lokaldatenbit
LB	LB	Lokaldatenbyte
LD	LD	Lokaldatendoppelwort
LDT	LDT	Datentyp
LINT	LINT	Datentyp
LTIME	LTIME	Datentyp
LTOD	LTOD	Datentyp
LW	LW	Lokaldatenwort
LWORD	LWORD	Datentyp
M	M	Merkerbit
MB	MB	Merkerbyte
MD	MD	Merkerdoppelwort
MOD	MOD	Modulo-Operator
MW	MW	Merkerwort
NAME	NAME	Bausteinname
NETWORK	NETWORK	Netzwerk
NOT	NOT	Logische Inversion
NULL	NULL	Null-Pointer
OB	OB	Organisationsbaustein
OF	OF	Einleitung der Datentypspezifikation/ Einleitung des Anweisungsteils der CASE-Anweisung
OR	OR	Oder-Verknüpfung logischer Ausdrücke
ORGANIZATION_BLOCK	ORGANIZATION_BLOCK	Einleitung des Organisationsbausteins
OS	OS	Überlauf, speichernd
OV	OV	Überlauf
PA	PQ	Ausgang (Peripherie direkt), Bit
PAB	PQB	Ausgang (Peripherie direkt), Byte
PAD	PQD	Ausgang (Peripherie direkt), Doppelwort
PAW	PQW	Ausgang (Peripherie direkt), Wort
PE	PI	Eingang (Peripherie direkt), Bit
PEB	PIB	Eingang (Peripherie direkt), Byte
PED	PID	Eingang (Peripherie direkt), Doppelwort
PEW	PIW	Eingang (Peripherie direkt), Wort

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
POINTER	POINTER	Datentyp
PRAGMA_BEGIN	PRAGMA_BEGIN	Einleitung eines Pragmas
PRAGMA_END	PRAGMA_END	Abschluss eines Pragmas
READ_ONLY	READ_ONLY	Schreibschutz für Datenbausteine
REAL	REAL	Datentyp
REF	REF	Anweisung zur Bildung einer Referenz
REF_TO	REF_TO	Einleitung für die Deklaration einer Referenz
REGION	REGION	Einleitung eines Bereichs
REPEAT	REPEAT	Einleitung der REPEAT-Anweisung
RET_VAL	RET_VAL	Rücksprungwert
RETURN	RETURN	RETURN-Anweisung in SCL
S5T	S5T	Syntax für Datentyp S5TIME
S5TIME	S5TIME	Datentyp
S7_	S7_	Schlüsselwörter für Systemattribute
SDB	SDB	Systemdatenbaustein
SFB	SFB	Systemfunktionsbaustein
SFC	SFC	Systemfunktion
SINT	SINT	Datentyp
STRING	STRING	Datentyp
STRUCT	STRUCT	Einleitung der Spezifikation einer Struktur, danach folgt Liste der Komponenten
STW	STW	Statuswort
SYSTEM_FUNCTION	SYSTEM_FUNCTION	Systemfunktion
SYSTEM_FUNCTION_BLOCK	SYSTEM_FUNCTION_BLOCK	Systemfunktionsbaustein
T	T	Zeitglied (Timer)
THEN	THEN	Einleitung des Anweisungsteils einer IF-Anweisung
THIS	THIS	Syntax für den Zugriff auf einen ARRAY-Datenbaustein
TIME	TIME	Elementarer Datentyp für Zeitangaben
TIME_OF_DAY	TIME_OF_DAY	Datentyp
TIMER	TIMER	Parametertyp zur Angabe einer Zeit
TITLE	TITLE	Optionaler Baustein- oder Netzwerktitel
TO	TO	Definition des Endwerts einer FOR-Anweisung
TOD	TOD	Datentyp

Schlüsselwörter Deutsche Mnemonik	Schlüsselwörter Englische Mnemonik	Beschreibung
TRUE	TRUE	Vordefinierte boolesche Konstante: Logische Bedingung erfüllt, Wert ungleich 0
TYPE	TYPE	Einleitung des PLC-Datentyps
UDT	UDT	globaler bzw. PLC-Datentyp
UDINT	UDINT	Datentyp
UINT	UINT	Datentyp
ULINT	ULINT	Datentyp
UNLINKED	UNLINKED	Kennzeichnung 'nicht ablaufrelevant'
UNTIL	UNTIL	Abschluss des Anweisungsteils einer REPEAT-Anweisung
USINT	USINT	Datentyp
UO	UO	Abfrage nach (A1=1) UND (A0=1)
VAR	VAR	Einleitung eines Deklarationsblocks
VAR_IN_OUT	VAR_IN_OUT	Einleitung eines Deklarationsblocks
VAR_INPUT	VAR_INPUT	Einleitung eines Deklarationsblocks
VAR_OUTPUT	VAR_OUTPUT	Einleitung eines Deklarationsblocks
VAR_TEMP	VAR_TEMP	Einleitung eines Deklarationsblocks
VARIANT	VARIANT	Datentyp
VERSION	VERSION	Versionsnummer des Bausteins
VOID	VOID	Funktion hat keinen Rückgabewert
WCHAR	WCHAR	Datentyp
WSTRING	WSTRING	Datentyp
WHILE	WHILE	Einleitung einer WHILE-Anweisung
WORD	WORD	Datentyp
XOR	XOR	Logische Operation
Z	C	Zähler (Counter)

1.4.3 Variablen

Definition

Eine Variable ist ein Platzhalter für einen im Programm veränderbaren Datenwert. Das Format des Datenwerts ist festgelegt. Durch die Verwendung von Variablen wird Ihr Programm flexibler. Sie können z. B. Variablen, die Sie in der Schnittstelle von Bausteinen deklariert haben, bei jedem Aufruf des Bausteins mit anderen Werten versorgen. So können Sie einen einmal programmierten Baustein für verschiedene Zwecke wiederverwenden.

Eine Variable besteht aus folgenden Elementen:

- Name
- Datentyp
- Absolute Adresse
  - PLC-Variablen und DB-Variablen in Bausteinen mit Standardzugriff haben eine absolute Adresse.
  - DB-Variablen in Bausteinen mit optimiertem Zugriff haben keine absolute Adresse.
- Wert (optional)

Deklaration von Variablen

Sie können für Ihr Programm Variablen mit unterschiedlichem Geltungsbereich definieren:

- PLC-Variablen, die CPU-weit gelten
- DB-Variablen in globalen Datenbausteinen, die CPU-weit von allen Bausteinen genutzt werden können.
- DB-Variablen in Instanzdatenbausteinen, die vorrangig in dem Baustein genutzt werden, in dem sie deklariert sind.

Die folgende Tabelle zeigt den Unterschied zwischen den Variablenarten:

	PLC-Variablen	Variablen in Instanz-DBs	Variablen in globalen DBs
Gültigkeitsbereich	• Gelten in der gesamten CPU. • Können von allen Bausteinen in der CPU benutzt werden. • Der Name ist innerhalb der CPU eindeutig.	• Werden vorrangig in dem Baustein genutzt, in dem sie definiert sind. • Der Name ist innerhalb des Instanz-DB eindeutig.	• Können von allen Bausteinen in der CPU benutzt werden. • Der Name ist innerhalb des globalen DB eindeutig.
Zulässige Zeichen	• Buchstaben, Ziffern, Sonderzeichen • Anführungszeichen sind nicht zulässig. • Reservierte Schlüsselwörter sind nicht zulässig.	• Buchstaben, Ziffern, Sonderzeichen • Reservierte Schlüsselwörter sind nicht zulässig.	• Buchstaben, Ziffern, Sonderzeichen • Reservierte Schlüsselwörter sind nicht zulässig.

	PLC-Variablen	Variablen in Instanz-DBs	Variablen in globalen DBs
Einsatz	• E/A-Signale (E, EB, EW, ED, A, AB, AW, AD) • Merker (M, MB, MW, MD)	• Bausteinparameter (Eingangs-, Ausgangs- und Durchgangsparameter), • Statische Daten eines Bausteins	• Statische Daten
Ort der Definition	PLC-Variablentabelle	Schnittstelle des Bausteins	Deklarationstabelle des globalen DB

Hinweis

Weitere Informationen zu zulässigen Zeichen in Variablennamen finden Sie im Siemens Industry Online Support in folgenden Beiträgen:

Wann müssen Bezeichner bzw. Operanden in STEP 7 (TIA Portal) in "Anführungszeichen" verwendet werden?

https://support.industry.siemens.com/cs/ww/de/view/10947785 (https://support.industry.siemens.com/cs/ww/de/view/109477857)

Empfehlungen für eine projektweit einheitliche Benennung von Variablen finden Sie im Programmierstyleguide:

https://support.industry.siemens.com/cs/ww/de/view/109478084 (https://support.industry.siemens.com/cs/ww/de/view/81318674)

Siehe auch

Schlüsselwörter (Seite 93)

Grundlagen zu Operanden (Seite 93)

Symbolische und absolute Operanden anzeigen (Seite 7131)

Zulässige Namen von PLC-Variablen (Seite 7375)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

1.4.4 Konstanten

1.4.4.1 Grundlagen zu Konstanten

Definition

Konstanten sind Daten, die einen festen Wert besitzen, der sich zur Programmlaufzeit nicht ändern kann. Konstanten können während der Programmbearbeitung von verschiedenen Programmelementen ausgelesen, aber nicht überschrieben werden. Für den Wert einer Konstante gibt es je nach Datentyp und Datenformat definierte Schreibweisen. Grundsätzlich wird zwischen der typisierten und der untypisierten Schreibweise unterschieden.

Untypisierte Konstanten

Bei untypisierter Schreibweise geben Sie nur den Wert der Konstanten ein, jedoch keinen Datentyp. Untypisierte Konstanten erhalten ihren Datentyp immer erst mit der arithmetischen oder logischen Verknüpfung, in der sie verwendet werden.

Das folgende Beispiel zeigt die untypisierte Konstantenschreibweise:

SCL

```
#My_Int1 := #My_Int2 + 12345        (*Der Datentyp der Konstanten "12345"
                                     ergibt sich aus der Addition mit My_Int 2.
                                     "12345" erhält den Datentyp INT.*)
#My_Real1 := #My_Real2 + 12345      (*Der Datentyp der Konstanten "12345" ergibt
                                     sich aus der Addition mit My_Real2. "12345" er-
                                     hält den Datentyp REAL.*)
```

Typisierte Konstanten

Bei typisierter Schreibweise geben Sie neben dem Wert der Konstanten explizit einen Datentyp an.

Das folgende Beispiel zeigt die typisierte Konstantenschreibweise:

SCL

```
#My_Int1 := INT#12345                    (*Der Datentyp der Konstante ist immer
                                         INT.*)
```

Hinweis

Konstanten vom Typ BOOL in KOP/FUP

Die Verwendung von Konstanten vom Typ BOOL an Eingängen von Anweisungen ist in KOP/FUP nur in CPUs der Baureihen S7-1200 ab FW V4.0 und S7-1500 ab FW V1.8 möglich.

Bei älterern Firmwareständen ist die Verwendung nur für Anweisungen zulässig, die systemintern ein Funktionsbaustein (FB) sind. Diese Anweisungen erkennen Sie daran, dass sich der Dialog "Aufrufoptionen" öffnet, wenn Sie die Anweisung in ein Netzwerk einfügen. Bei allen anderen Anweisungen dürfen boolesche Konstanten nicht als Eingänge verwendet werden.

In S7-300/400 ist die Verwendung nicht zulässig.

Siehe auch

Bausteinschnittstelle deklarieren (Seite 7272)

Globale Konstanten deklarieren (Seite 7391)

Mit Konstanten in SCL rechnen (Seite 7627)

1.4.4.2 Deklaration symbolischer Namen für Konstanten

Symbolische Konstanten

Sie haben die Möglichkeit, symbolische Namen für Konstanten zu deklarieren, und so konstante Werte unter einem Namen im Programm verfügbar zu machen. Dadurch erreichen Sie eine bessere Lesbarkeit und eine einfachere Pflege des Programms bei Änderungen von konstanten Werten.

Eine symbolische Konstante besteht aus folgenden Elementen:

- Name
- Datentyp
  Symbolische Konstanten haben immer einen Datentyp, die untypisierte Schreibweise ist bei symbolischen Konstanten nicht möglich.
- Konstantenwert
  Als Konstantenwert können Sie einen beliebigen Wert aus dem Wertebereich des angegebenen Datentyps wählen. Angaben zu den Wertebereichen finden Sie im Kapitel "Datentypen".

Deklaration von Konstanten

Sie können Konstanten mit unterschiedlichem Geltungsbereich definieren:

- Globale Konstanten, die CPU-weit gelten
- Lokale Konstanten, die nur innerhalb eines Bausteins gelten

Die folgende Tabelle zeigt den Unterschied zwischen den Konstantenarten:

	Globale Konstanten	Lokale Konstanten
Gültigkeitsbereich	• Gelten in der gesamten CPU • Der Name ist innerhalb der CPU eindeutig.	• Gelten in dem Baustein, in dem sie deklariert wurden. • Der Name ist innerhalb des Bausteins eindeutig.
Zulässige Zeichen	• Zulässige Zeichen im Konstantennamen sind Buchstaben, Ziffern und Sonderzeichen.	• Zulässige Zeichen im Konstantennamen sind Buchstaben, Ziffern und Sonderzeichen.
Ort der Definition	Register "Konstanten" der PLC-Variablentabelle	Schnittstelle des Bausteins
Darstellung	In Anführungszeichen, z. B.: `"Glob_Const"`	Mit vorangestelltem Nummernzeichen, z. B.: `#Loc_Const`

Namenskonflikte können auftreten, wenn Sie eine lokale und eine globale Konstante mit gleichem symbolischem Namen deklariert haben, und den doppelt vergebenen Namen als Defaultwert einer Variablen verwenden. In diesem Fall wird automatisch die lokale Konstante verwendet.

Hinweis

Laden der Konstantendeklaration (S7-300/400)

Lokale und globale Konstantendeklarationen werden nicht in die CPU geladen. Wenn Sie ein Programm von einem Gerät laden, stehen die Konstantendeklarationen möglicherweise nicht mehr zur Verfügung.

Hinweis

Weitere Informationen zu zulässigen Zeichen in Konstantennamen finden Sie im Siemens Industry Online Support in folgenden Beiträgen:

Wann müssen Bezeichner bzw. Operanden in STEP 7 (TIA Portal) in "Anführungszeichen" verwendet werden?

https://support.industry.siemens.com/cs/ww/de/view/10947785 (https://support.industry.siemens.com/cs/ww/de/view/109477857)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Mit Konstanten in SCL rechnen (Seite 7627)

Globale Konstanten deklarieren (Seite 7391)

Bausteinschnittstelle deklarieren (Seite 7272)

1.4.4.3 Datentypen von Konstanten

Zulässige Datentypen

Für Konstanten sind alle Basisdatentypen sowie davon abgeleitete Datentypen zulässig:

- Binärzahlen
- Bitfolgen
- Ganzzahlen
- Gleitpunktzahlen
- Zeiten
- Datum und Uhrzeit
- Zeichenfolgen

Es gelten die allgemeinen Regeln zur expliziten und impliziten Datentypkonvertierung.

Datentypen untypisierter Konstanten

Untypisierte Konstanten enthalten keine explizite Datentypangabe. Sie erhalten ihren Datentyp immer erst mit der arithmetischen oder logischen Verknüpfung, in der sie verwendet werden.

Das folgende Beispiel zeigt die Behandlung untypisierter Konstanten:

SCL

```
#My_Int1 := #My_Int2 + 12345        (*Der Datentyp der Konstanten "12345"
                                     ergibt sich aus der Addition mit My_Int 2.
                                     "12345" erhält den Datentyp INT.*)
#My_Real1 := #My_Real2 + 12345      (*Der Datentyp der Konstanten "12345" ergibt
                                     sich aus der Addition mit My_Real2. "12345" er-
                                     hält den Datentyp REAL.*)
```

Hinweis

STEP 7 verwendet immer den Datentyp mit der höchstmöglichen Genauigkeit

Solange innerhalb eines Ausdrucks der Datentyp einer Konstanten nicht eindeutig definierbar ist, wird immer der Datentyp mit der höchstmöglichen Genauigkeit verwendet, der auf der aktuellen CPU verfügbar ist.

Beispiel:

```
#My_Real := #My_Int / 3.5
```

In diesem Ausdruck wird eine Ganzzahlvariable mit einer untypisierten Gleitpunktkonstanten kombiniert. In S7-300/400 wird hier der rechte Teil der Zuweisung im Format REAL gerechnet. In S7-1200/1500 wird automatisch mit der größtmöglichen Genauigkeit gerechnet, d. h. in diesem Fall in LREAL. Als Folge davon ist die Zuweisung an eine REAL-Variable ungültig oder erzeugt eine Warnung.

Um den Datentyp einer Konstanten genau zu definieren, verwenden Sie die typisierte Schreibweise:

Beispiel:

```
#My_Real := #My_Int / REAL#3.5
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Mit Konstanten in SCL rechnen (Seite 7627)

Globale Konstanten deklarieren (Seite 7391)

Bausteinschnittstelle deklarieren (Seite 7272)

1.4.4.4 Beispiele zur Verwendung von Konstanten

Verwendung in Anweisungen, Zuweisungen und Ausdrücken

Konstanten können in Anweisungen oder Zuweisungen anstelle von Variablen verwendet werden. In SCL können Sie Konstanten auch in Ausdrücken verwenden. Da Konstanten nicht geschrieben werden können, dürfen sie jedoch nur als Eingänge verwendet werden.

Das folgende Beispiel zeigt mögliche Verwendungen von Konstanten:

SCL

```
#My_Int :=   3;
#My_Real1 := #My_Real2 * 3;
#My_Real1 := #My_Real2 * #My_local_const;
#My_Real1 := #My_Real2 * "My_global_const";
```

Verwendung als Defaultwert

Sie können Konstanten als Defaultwert einer Variablen verwenden. Dazu tragen Sie entweder den Wert oder den symbolischen Namen der Konstanten in der Bausteinschnittstelle in die Spalte "Defaultwert" ein. Der Datentyp der Konstanten muss dem Datentyp der Variablen entsprechen oder nach den Regeln der impliziten Datentypkonvertierung mit IEC-Prüfung konvertierbar sein.

Verwendung als maximale STRING-Länge

Sie können lokale oder globale Konstanten vom Datentyp UINT, UDINT, ULINT, SINT, INT, DINT, LINT als maximale STRING-Länge verwenden.

Das folgende Beispiel zeigt die Verwendung von Konstanten als maximale STRING-Länge:

SCL

```
STRING[#My_local_const1]
STRING["My_global_const1"]
```

Verwendung als ARRAY-Grenze

Sie können lokale oder globale Konstanten vom Datentyp UINT, UDINT, ULINT, SINT, INT, DINT, LINT als ARRAY-Grenzen verwenden.

Das folgende Beispiel zeigt die Verwendung von Konstanten als ARRAY-Grenzen:

SCL

```
Array[#My_local_const1..#My_local_const2] of REAL
Array["My_global_const1".."My_global_const1"] of REAL
```

Hinweis

Konstanten als ARRAY-Grenzen oder als maximale STRING-Länge

- Konstanten, die als ARRAY-Grenzen oder maximale STRING-Länge verwendet werden, lassen sich nicht ändern, wenn die Speicherreserve des Bausteins aktiviert ist. Dies gilt sowohl für lokale als auch für globale Konstanten. Um diese Konstanten zu ändern, müssen Sie zunächst die Speicherreserve deaktivieren.
- Änderungen an globalen Konstanten führen zu Inkonsistenzen in den Bausteinen, die diese verwenden. Die Inkonsistenzen werden im verwendenden Baustein rot markiert. Um die Inkonsistenzen zu beheben, müssen Sie die Bausteine aktualisieren.
  Siehe auch: Datenbausteine aktualisieren (Seite 7326)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Mit Konstanten in SCL rechnen (Seite 7627)

Grundlagen zum ARRAY (Seite 296)

Globale Konstanten deklarieren (Seite 7391)

Bausteinschnittstelle deklarieren (Seite 7272)

1.4.5 Operanden adressieren

1.4.5.1 Globale Variablen adressieren

Globale Variablen adressieren

Zur Adressierung einer globalen PLC-Variablen können Sie die absolute Adresse oder den symbolischen Namen verwenden.

Hinweis

Die Datentypen LWORD, LINT, ULINT, LREAL, LTIME, LTOD und LDT können nur mit ihrem symbolischen Namen adressiert werden.

Globale Variablen symbolisch adressieren

Bei der symbolischen Adressierung geben Sie den Variablennamen aus der PLC-Variablentabelle ein. Der symbolische Name von globalen Variablen wird automatisch in Anführungszeichen eingeschlossen.

Strukturierte Variablen, die auf einem PLC-Datentyp basieren, adressieren Sie mit dem symbolischen Namen der PLC-Variablen. Getrennt durch einen Punkt können Sie auch die Namen der einzelnen Komponenten angeben.

Globale Variablen absolut adressieren

Bei der absoluten Adressierung geben Sie die Adresse der Variablen aus der PLC-Variablentabelle ein. Die absolute Adresse verwendet numerische Adressen beginnend bei Null für jeden Operandenbereich. Der absoluten Adresse von globalen Variablen wird automatisch das Adresskennzeichen % vorangestellt.

Beispiele

Die folgenden Beispiele zeigen Anwendungen der symbolischen und absoluten Adressierung:

Adressierung	Erläuterung
%A1.0	Absolute Adresse: Ausgang 1.0
%E16.4	Absolute Adresse: Eingang 16.4
%EW4	Absolute Adresse: Eingangswort 4
"Motor"	Symbolische Adresse "Motor"
"Value"	Symbolische Adresse "Value"

Adressierung	Erläuterung
"Strukturierte_Variable"	Symbolische Adresse einer Variable, die auf einem PLC-Datentyp basiert
"Strukturierte_Variable".Komponente	Symbolische Adresse der Komponente einer strukturierten Variable.

Siehe auch

Symbolische und absolute Operanden anzeigen (Seite 7131)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

Peripherie adressieren (Seite 109)

Peripherie adressieren

Beschreibung

In einem Zyklus wird das Prozessabbild der CPU einmal aktualisiert. In zeitkritischen Anwendungen kann es jedoch sein, dass Sie den aktuellen Zustand eines digitalen Eingangs oder Ausgangs häufiger als einmal pro Zyklus lesen oder übertragen müssen. Zu diesem Zweck können Sie mithilfe einer Peripheriezugriffskennung am Operanden direkt auf die Peripherie zugreifen.

Wenn Sie den Eingang direkt aus der Peripherie lesen möchten, verwenden Sie den Speicherbereich Peripherieeingänge (PE) im Gegensatz zum Prozessabbild der Eingänge (E). Der Speicherbereich der Peripherie kann als Bit, Byte, Wort oder Doppelwort gelesen werden.

Wenn Sie den Ausgang direkt schreiben möchten, verwenden Sie den Speicherbereich der Peripherieausgänge (PA) im Gegensatz zum Prozessabbild der Ausgänge (A). Der Speicherbereich der Peripherieausgänge kann als Bit, Byte, Wort und Doppelwort beschrieben werden.

Um ein Signal direkt von einem Peripherieeingang zu lesen oder zu schreiben, kann ein Operand um die Peripheriezugriffskennung ":P" erweitert werden.

Komponenten von strukturierten PLC-Variablen können ebenfalls mit ":P" adressiert werden. Ein Zugriff auf die übergeordnete Variable mit ":P" ist jedoch nicht möglich.

⚠ WARNUNG

Direktes Schreiben der Peripherie

Das direkte Schreiben der Peripherie kann gefährliche Zustände hervorrufen, z. B. wenn ein Peripherieausgang in einem Programmzyklus mehrfach schreibend adressiert wird.

Hinweis

S7-1200/1500: Peripherieeingänge oder -ausgänge als Bausteinparameter übergeben

Wenn Sie einen Eingangsparameter mit einem Peripherieeingang oder -ausgang versorgen, besteht das Risiko, dass zur Laufzeit Peripheriezugriffsfehler beim Bausteinaufruf auftreten, z. B. ein Lesefehler beim direkten Zugriff auf Daten einer Eingabebaugruppe.

Das Systemverhalten bei CPUs der Baureihen S7-1500 mit FW-Version ab V2.1 ist wie folgt:

Der Baustein wird aufgerufen und mit dem Ersatzwert des Signals bearbeitet.

Das Systemverhalten bei CPUs der Baureihen S7-1200 und S7-1500 mit FW-Version kleiner V2.1 ist wie folgt:

Der Baustein wird aufgrund des Peripheriezugriffsfehlers nicht aufgerufen. Die Programmausführung wird nach dem Bausteinaufruf fortgesetzt. Wenn ein OB 122 vorhanden ist, oder die lokale Fehlerbehandlung aktiviert ist, werden diese ausgeführt.

Um zu verhindern, dass ein Bausteinaufruf bei einem Peripheriezugriffsfehler nicht ausgeführt wird, kopieren Sie den Peripherieeingang oder -ausgang zunächst in eine lokale Variable (Temp) und übergeben diese als Bausteinparameter an den aufgerufenen Baustein.

Der folgende FAQ gibt weitere Informationen zu diesem Thema:

FAQ 89377245: Warum werden Bausteine in der S7-1200/S7-1500 nicht bearbeitet, wenn Sie sie mit nicht vorhandenen PROFINET-Komponenten adressieren? (https://support.industry.siemens.com/cs/ww/de/view/89377245)

Syntax

```
<Operand>:P
```

Beispiel

Das folgende Beispiel zeigt Anwendungen der Peripheriezugriffskennung:

Adressierung	Erläuterung
"Motor"	Adressiert die Variable "Motor" im Prozessabbild.
"Motor":P	Adressiert die Variable "Motor" im Speicherbereich Peripherie (PE oder PA).
"Strukturierte_Variable".Komponente	Adressiert die Komponente einer strukturierten PLC-Variable im Prozessabbild.
"Strukturierte_Variable".Komponente:P	Adressiert die Komponente einer strukturierten PLC-Variable im Speicherbereich Peripherie (PE oder PA).

Siehe auch

Globale Variablen adressieren (Seite 108)

1.4.5.2 Variablen in Datenbausteinen adressieren

Variablen in globalen Datenbausteinen adressieren

Beschreibung

Variablen in globalen Datenbausteinen können symbolisch oder absolut adressiert werden. Bei symbolischer Adressierung verwenden Sie den Namen des Datenbausteins und den Namen der Variablen, getrennt durch einen Punkt. Der Name des Datenbausteins wird in Anführungszeichen gesetzt.

Bei absoluter Adressierung verwenden Sie die Nummer des Datenbausteins und die absolute Adresse der Variablen im Datenbaustein, getrennt durch einen Punkt. Der absoluten Adresse wird automatisch das Adresskennzeichen % vorangestellt.

In S7-1200/1500 haben Sie die Möglichkeit, auf einen Datenbaustein zuzugreifen, der bei der Programmierung noch nicht bekannt ist. Dazu erstellen Sie in der Bausteinschnittstelle des zugreifenden Bausteins einen Bausteinparameter vom Datentyp DB_ANY. An diesen Parameter wird zur Laufzeit der Datenbausteinname oder die Datenbausteinnummer übergeben. Um auf die Variablen innerhalb des Datenbausteins zuzugreifen, verwenden Sie den Namen des Bausteinparameters vom Datentyp DB_ANY und die absolute Adresse der Variablen, getrennt durch einen Punkt.

Hinweis

DB mit Speicherreserve an den Parameter DB_ANY übergeben.

Es ist nicht möglich, einen DB mit Speicherreserve an einem Bausteinparameter vom Datentyp "DB_ANY" zu übergeben.

Hinweis

DB-Variablen absolut adressieren

Folgende Variablen können nicht absolut adressiert werden:

- Variablen in Bausteinen mit optimiertem Zugriff.
- Variablen der Datentypen LWORD, LINT, ULINT, LREAL, LTIME, LTOD und LDT.

Verwenden Sie für diese Variablen die komfortablere symbolische Adressierung.

ARRAY-Datenbausteine

Eine besondere Art von globalen Datenbausteinen sind ARRAY-Datenbausteine. Diese bestehen aus einem ARRAY eines beliebigen Datentyps. Es kann z. B. ein ARRAY eines PLC-Datentyps (UDT) sein.

Elemente in ARRAY-Datenbausteinen adressieren Sie mithilfe des Schlüsselworts "THIS". Anschließend wird der Index in eckigen Klammern angegeben. Der Index kann sowohl eine Konstante als auch eine Variable sein. Als Variablen für den Index sind Ganzzahlen von einer Breite bis 32 Bit zulässig.
Erweiterte Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben". Diese Anweisungen bieten z. B. die Möglichkeit, auch den DB-Namen indirekt zu adressieren.

Syntax

```
"<DBname>".Variablenname
%<DBnummer>.absoluteAdresse
#<DBAny_name>.%absoluteAdresse
"<ArrayDBname".THIS[#i].<Component>.<ComponentElement>
```

SCL:
```
"<ArrayDBname"."THIS"[#i].<Component>.<ComponentElement>
```

Die folgende Tabelle zeigt die möglichen absoluten Adressen von Variablen in Datenbausteinen:

Datentyp	Absolute Adresse	Beispiel	Erläuterung
BOOL	%DBn.DBXx.y	%DB1.DBX1.0	Datenbit 1.0 im DB1
BYTE, CHAR, SINT, USINT	%DBn.DBBy	%DB1.DBB1	Datenbyte 1 im DB1
WORD, INT, UINT	%DBn. DBWy	%DB1.DBW1	Datenwort 1 im DB1
DWORD, DINT, UDINT, REAL, TIME	%DBn.DBDy	%DB1.DBD1	Datendoppelwort 1 im DB1

Beispiel

Die folgenden Beispiele zeigen die Adressierung von Variablen in globalen Datenbausteinen:

Adressierung	Erläuterung
"Motor".Value	Symbolische Adressierung der Variablen "Value" im globalen Datenbaustein "Motor".
%DB1.DBX1.0	Absolute Adressierung der Variablen "DBX1.0" im globalen Datenbaustein "DB1".
#MyDBAny.%DBX30.0	Absolute Adressierung der Variablen "DBX30.0" im dem globalen Datenbaustein, der zur Laufzeit am Parameter "MyDBany" übergeben wird.
"MyARRAY_DB".THIS[#MyIndex].My-Component.MyComponentElement	Adressierung eines ARRAY-Datenbausteins. Der ARRAY-Index wird über die Variable "MyIndex" angegeben. Das ARRAY-Element enthält noch zwei weitere Unterstrukturen: "MyComponent" und "MyComponentElement".

Siehe auch

Datentyp DB_ANY verwenden (Seite 215)

Einzelne Zeichen eines STRINGs oder WSTRINGs adressieren (Seite 283)

PLC-Datentypen (UDT) adressieren (Seite 290)

STRUCT-Komponenten adressieren (Seite 294)

ARRAY-Komponenten adressieren (Seite 303)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zur indirekten Adressierung (Seite 119)

Instanzdaten adressieren (Seite 113)

Instanzdaten adressieren

Beschreibung

Sie können Datenelemente aus der Schnittstelle des aktuellen Bausteins adressieren. Diese Variablen werden im Instanz-Datenbaustein gespeichert.

Hinweis

Variablen in Bausteinen mit optimiertem Zugriff können nur symbolisch adressiert werden.

Um eine Variable aus der Schnittstelle des aktuellen Bausteins zu adressieren, geben Sie das Zeichen # gefolgt von dem symbolischen Variablennamen ein.

Sie können auch auf die Variablen eines Multiinstanzbausteins zugreifen. Innerhalb des Multiinstanzbausteins adressieren Sie die Daten ebenfalls durch das Zeichen # gefolgt von dem Variablennamen. Vom aufrufenden Baustein greifen Sie auf die Daten des Multiinstanzbausteins über #<Multiinstanzname.Variablenname> zu.

Syntax

Für die Adressierung von Variablen in Instanz-Datenbausteinen wird die folgende Syntax verwendet:

```
#<Variablenname>
#<Multiinstanzname>.<Variablenname>
```

Beispiele

Die folgenden Beispiele zeigen die Adressierung von Variablen in Instanzdatenbausteinen:

Addressierung	Erläuterung
#Value	Adressierung der Variablen "Value" im Instanz-Datenbaustein
#On	Adressierung der Variablen "On" innerhalb des Multiinstanzbausteins
#Multi.On	Adressierung der Variablen "On" des Multiinstanzbausteins vom aufrufenden Baustein aus

Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 111)

Instanzen (Seite 64)

ARRAY-Komponenten adressieren (Seite 303)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zur indirekten Adressierung (Seite 119)

ARRAY-DBs adressieren

ARRAY-Datenbausteine adressieren

Eine besondere Form des ARRAYs sind ARRAY-Datenbausteine. ARRAY-Datenbausteine sind globale Datenbausteine, die aus genau einem ARRAY bestehen. Elemente in ARRAY-Datenbausteinen adressieren Sie über die folgende Syntax:

```
"<GlobArrayDBname>".THIS[#i].<elementname>
```

SCL:

```
"<GlobArrayDBname>"."THIS"[#i].<elementname>
```

Beispiele

Die folgenden Beispiele zeigen die Adressierung von Elementen in einem ARRAY-DB:

Adressierung in SCL	Erläuterung
"My_ARRAY_DB"."THIS"[0]	"My_ARRAY_DB" ist ein ARRAY-DB, der Komponenten vom Datentyp REAL enthält. Hier wird die erste Komponente innerhalb von "My_ARRAY_DB" adressiert.
"MY_UDT_ARRAY_DB"."THIS"[3].MyREAL	"MY_UDT_ARRAY_DB" ist ein ARRAY-DB, der Komponenten eines PLC-Datentyps (UDT) enthält. Der PLC-Datentyp enhtält Elemente vom Datentyp BYTE. Hier wird im "MY_UDT_ARRAY_DB" die vierte ARRAY-Komponente und darin das Element "MyREAL" adressiert.

Anweisungen zur Adressierung von ARRAY-DBs

Erweiterte Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen > Einfache Anweisungen" im Abschnitt "Verschieben > ARRAY-DB". Diese Anweisungen bieten z. B. die Möglichkeit, den DB-Namen indirekt zu adressieren:

- ReadFromArrayDB: Aus Array-Datenbaustein lesen
- WriteToArrayDB: In Array-Datenbaustein schreiben
- ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen
- WriteToArrayDBL: In Array-Datenbaustein im Ladespeichern schreiben

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Indirektes Indizieren von ARRAY-Komponenten (Seite 127)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Globale ARRAY-Datenbausteine (DB) (Seite 43)

ARRAY (Seite 296)

Bereiche einer Variablen mit Slice-Zugriffen adressieren

Beschreibung

Sie haben die Möglichkeit, gezielt Bereiche innerhalb deklarierter Variablen zu adressieren. Dabei können Sie auf Bereiche der Breiten 1 Bit, 8 Bit, 16 Bit oder 32 Bit zugreifen. Die Aufteilung eines Speicherbereiches (z. B. BYTE oder WORD) in einen kleineren Speicherbereich (z. B. BOOL) wird auch "Slice" genannt.

Strukturen, Konstanten und mit AT überlagernde Variablen sind nicht durch Slice-Zugriffe adressierbar.

Syntax

Für die Adressierung wird die folgende Syntax verwendet:

```
<Variable>.X<BIT-Nummer>
<Variable>.B<BYTE-Nummer>
<Variable>.W<WORD-Nummer>
<Variable>.D<DWORD-Nummer>
```

Die Syntax besteht aus folgenden Teilen:

Teil	Beschreibung
<Variable>	Variable, auf die Sie zugreifen. Die Variable muss vom Datentyp "Bitfolge" oder „Ganzzahl“ sein. Mit SCL können Sie nur bei deaktivierter IEC-Prüfung Slice-Zugriffe auf Variablen des Datentyps "Ganzzahl" programmieren.
X B W D	Kennung für die Zugriffsbreite "Bit (1Bit)" Kennung für die Zugriffsbreite "Byte (8 Bit)" Kennung für die Zugriffsbreite "Word (16 Bit)" Kennung für die Zugriffsbreite "DWord (32 Bit)"
<BIT-Nummer>	Nummer des Bits innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste BIT zu.
<BYTE-Nummer>	Nummer des Bytes innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste BYTE zu.

Teil	Beschreibung
<WORD-Nummer>	Nummer des Words innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste WORD zu.
<DWORD-Nummer>	Nummer des DWords innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste DWORD zu.

Beispiele zur Slice-Zugriffe

Die folgenden Beispiele zeigen die bit-, byte-, wort- und doppelwortweise Adressierung von Slices:

Adressierung	Erläuterung
"Engine".Motor.X0 "Engine".Motor.X7	"Motor" ist eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD im globalen Datenbaustein "Engine". X0 adressiert die Bitadresse 0, X7 die Bitadresse 7 innerhalb von "Motor".
"Engine".Speed.B0 "Engine".Speed.B1	"Speed" ist eine Variable vom Datentyp WORD, DWORD oder LWORD im globalen Datenbaustein "Engine". B0 adressiert die Byteadresse 0, B1 die Byteadresse 1 innerhalb von "Speed".
"Engine".Fuel.W0 "Engine".Fuel.W1	"FUEL" ist eine Variable vom Datentyp DWORD oder LWORD im globalen Datenbaustein "Engine". W0 adressiert die Wortadresse 0, W1 die Wortadresse 1 innerhalb von "Fuel".
"Engine".Data.D0 "Engine".Data.D1	"Data" ist eine Variable vom Datentyp LWORD im globalen Datenbaustein "Engine". D0 adressiert die Doppelwortadresse 0, D1 die Doppelwortadresse 1 innerhalb von "Data".

Programmierbeispiel

Ein ausführliches Beispiel finden Sie im Siemens Industry Online Support:

https://support.industry.siemens.com/cs/ww/de/view/57374718 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=57374718&Datakey=47071380&caller=nl)

Variablen mit AT überlagern

Beschreibung

Um auf Datenbereiche innerhalb einer deklarierten Variablen zuzugreifen, können Sie die deklarierten Variablen mit einer weiteren Deklaration überlagern. Sie haben so die Möglichkeit, eine bereits deklarierte Variable mit einem anderen Datentyp anzusprechen. Sie können z. B. die einzelnen Bits einer Variablen vom Datentyp WORD mit einem ARRAY of BOOL ansprechen.

Alternativ können Sie auch die Anweisungen "SCATTER" und "GATHER" verwenden, um Bitfolgen in ein ARRAY of BOOL zu zerlegen oder einzelne Bits zu einer Bitfolge zusammenzufassen. Diese Anweisungen finden Sie in der Task Card "Einfache Anweisungen" im Bereich "Verschieben".

Regeln

Folgende allgemeine Regeln gelten für das Überlagern von Variablen:

- In AWL, KOP, FUP und GRAPH ist das Überlagern in S7-1200 und S7-1500 möglich.
- In SCL ist das Überlagern in allen CPU-Familien möglich.
- Das Überlagern von Variablen ist in folgenden Bausteinen möglich:
  - In Codebausteinen mit Standardzugriff
  - In Codebausteinen mit optimiertem Zugriff für Variablen mit der Remanenzeinstellung "Im IDB setzen"
- Die Datenbreite der überlagernden Variablen muss gleich oder kleiner als die der überlagerten Variablen sein.
- Die Datenypen VARIANT und INSTANCE können nicht überlagert werden.
- Wenn die überlagerte Variable vom Datentyp STRING, ARRAY of BYTE, ARRAY of CHAR etc. ist, sollte ihre Länge einer geraden Anzahl von Byte entsprechen.
- Bausteine aus Bibliotheken, die als Parameter in der Schnittstelle deklariert sind, können nicht überlagert werden.
- Strukturierte PLC-Variablen, die als Parameter in der Schnittstelle deklariert sind, können nicht überlagert werden.
- Überlagernde Variablen können nicht durch Slice-Zugriffe adressiert werden.

Hinweis

S7-1200/1500: AT in FCs verwenden

Bei FCs in S7-1200/1500 müssen die Datenbreiten der überlagernden Variablen und der überlagerten Variablen gleich sein. Falls dies in Ihrem Programm nicht möglich ist, prüfen Sie, ob Sie anstelle des AT-Konstrukts einen Slice-Zugriff verwenden können, oder verwenden Sie die Anweisungen "SCATTER" bzw. "GATHER".

Siehe auch: Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Darüber hinaus gelten folgende Kombinationsregeln:

		Überlagernde Variable	Überlagerte Variable			
			Elementar	Strukturiert *	Any/Pointer	DB_ANY
FB	Input	Elementar	x	x		x
		Strukturiert *	x	x	x	x
		Any/Pointer		x		
	Temp	Elementar	x	x		
		Strukturiert	x	x	x	
		Any/Pointer		x		
	Static, Output	Elementar	x	x		x
		Strukturiert	x	x		x
		Any/Pointer				
	InOut	Elementar	x			x
		Strukturiert		x		
		Any/Pointer				
FC	Temp	Elementar	x	x		
		Strukturiert	x	x	x	
		Any/Pointer		x		
	Input, Output, InOut	Elementar (beide Variablen müssen die gleiche Bitbreite haben)	x			x
		Strukturiert		x		
		Any/Pointer				
OB	Temp	Elementar	x	x		
		Strukturiert	x	x	x	
		Any/Pointer		x		

* Strukturierte Datentypen sind Datentypen, die aus mehreren Datenelementen bestehen, z. B. ARRAY oder STRUCT.

Deklaration

Um eine Variable zu überlagern, deklarieren Sie eine weitere Variable direkt nach der zu überlagernden Variable und kennzeichnen sie mit dem Schlüsselwort "AT".

Beispiel

Das folgende Bild zeigt die Deklaration einer überlagerten Variable in der Schnittstelle eines FB:

Input		
MyByte		Byte
AT	AT "MyByte"	Array [0..7] of Bool
AT[0]		Bool
AT[1]		Bool
AT[2]		Bool
AT[3]		Bool
AT[4]		Bool
AT[5]		Bool
AT[6]		Bool
AT[7]		Bool

Beim Aufruf eines Bausteins mit der gezeigten Variablendeklaration wird die Variable "MyByte" versorgt. Innerhalb des Bausteins stehen nun zwei Möglichkeiten zur Verfügung, die Daten zu interpretieren:

- als Byte
- als eindimensionales ARRAY of BOOL

Siehe auch

Remanenz einstellen (Seite 7338)

1.4.5.3 Operanden indirekt adressieren

Indirekte Adressierung

Grundlagen zur indirekten Adressierung

Einführung

Die indirekte Adressierung bietet die Möglichkeit, Operanden zu adressieren, deren Adresse erst zur Laufzeit berechnet wird. Sie können mit indirekter Adressierung auch Programmteile mehrfach bearbeiten lassen und bei jedem Durchlauf einen anderen Operanden verwenden. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

Dadurch haben Sie die folgenden Vorteile:

- Sie erreichen mehr Flexibilität innerhalb Ihres Programms.
- Die indirekte Adressierung steht Ihnen in allen STEP 7-Programmiersprachen zur Verfügung.
- Der Programmcode wir übersichtlicher.
- Es werden die vorhandenen Namen der Datenbausteine und Variablen verwendet (Symbolische Adressierung). Dadurch erhöht sich die Lesbarkeit des Programmcodes.

⚠ WARNUNG
Gefahr von Zugriffsfehlern Da bei der indirekten Adressierung die Operanden erst zur Laufzeit berechnet werden, besteht die Gefahr, dass Zugriffsfehler auftreten und das Programm mit falschen Werten arbeitet. Außerdem können ungewollt Speicherbereiche mit falschen Werten überschrieben werden. Das Automatisierungssystem kann dann unerwartet reagieren. Setzen Sie deshalb die indirekte Adressierung nur mit Vorsicht ein.

Allgemeine Möglichkeiten der indirekten Adressierung in S7-1200 und S7-1500

Die folgenden Möglichkeiten der indirekten Adressierung stehen in allen Programmiersprachen zur Verfügung:

- Indirekte Adressierung über Pointer
- Indirektes Indizieren von ARRAY-Komponenten
- Indirekte Adressierung eines Datenbausteins über den Datentyp DB_ANY.

Sprachspezifische Möglichkeiten der indirekten Adressierung

Darüberhinaus gibt es in den einzelnen Programmiersprachen folgende spezifische Adressierungsmöglichkeiten:

- In AWL können Sie über die Adressregister Operanden indirekt adressieren.
- In SCL können Sie einen variablen Speicherbereich mit den folgenden Anweisungen lesen oder schreiben:
  - POKE - Speicheradresse schreiben
  - POKE_BOOL - Speicherbit schreiben
  - PEEK - Speicheradresse lesen
  - PEEK_BOOL - Speicherbit lesen
  - POKE_BLK - Speicherbereich schreiben

Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 111)

POKE: Speicheradresse schreiben (Seite 1854)

POKE_BOOL: Speicherbit schreiben (Seite 1855)

PEEK: Speicheradresse lesen (Seite 1850)

PEEK_BOOL: Speicherbit lesen (Seite 1852)

POKE_BLK: Speicherbereich schreiben (Seite 1857)

Indirektes Adressieren über Zeiger (Seite 126)

Indirektes Indizieren von ARRAY-Komponenten (Seite 127)

Indirektes Adressieren in AWL (Seite 133)

Beispiele zur indirekten Adressierung

1. Programmierbeispiel

Im folgenden Beispiel greifen Sie per Index auf drei Variablen aus unterschiedlichen Speicherbereichen zu.

Übersicht der drei Variablen, die je einem Index zugeordnet sind:

Index	Zugriff auf Variable	Speicherbereich
1	Input_WORD_0	EW 0
2	"Processdata".Temperature	DB 1
3	Output_WORD_4	AW 4

Deklarieren Sie in der "Standard-Variablentabelle" die beiden folgenden Variablen:

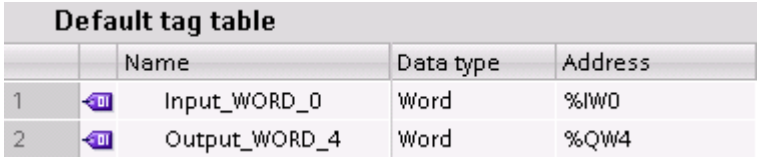

Default tag table

		Name	Data type	Address
1		Input_WORD_0	Word	%IW0
2		Output_WORD_4	Word	%QW4

Legen Sie einen globalen Datenbaustein an:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_Processdata" an.
4. Wählen Sie als Typ des Datenbausteins "Global DB" aus.

5. Klicken Sie auf "OK".
6. Deklarieren Sie das Datenbaustein-Element "Temperature":

DB_Processdata			
	Name	Data type	Start value
1	Static		
2	Temperature	DInt	50

Deklarieren Sie innerhalb einer Funktion die indirekten Zugriffe mithilfe eines Index.

1. Legen Sie eine SCL-Funktion an und benennen sie "FB_AccessGroupInt".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_AccessGroupInt		
	Name	Data type
1	Input	
2	Index	Int
3	Output	
4	InOut	
5	Temp	
6	Constant	
7	Return	
8	FC_AccessGroupInt	DInt

3. Schreiben Sie folgenden Programmcode:

```
1
2  CASE #Index OF
3
4      1: // First case: "Input_WORD_0"
5          #FC_AccessGroupInt := "Input_WORD_0";
6      2: // Second case: "DB_Processdata".Temperature
7          #FC_AccessGroupInt := "DB_Processdata".Temperature;
8      3: // Third case: "Output_WORD_4"
9          #FC_AccessGroupInt := "Output_WORD_4";
10
11 END_CASE;
```

4. Rufen Sie die Funktion "FC_AccessGroupInt" im OB1 auf:

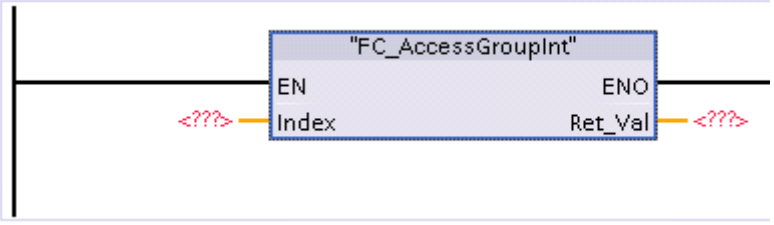

Je nachdem, welche Zahl Sie am Parameter Index angeben (1, 2 oder 3) wird der erste, zweite oder dritte Fall der Anweisung "FC_AccessGroupInt" ausgeführt.

2. Programmierbeispiel

Im folgenden Beispiel greifen Sie per Index auf drei verschiedene optimierte Datenbausteine zu.

Da alle Datenbausteine die gleichen Variablen enthalten sollen, können Sie hier mit einem PLC-Datentyp (UDT) arbeiten.

1. Legen Sie einen PLC-Datentyp an, indem Sie in der Projektnavigation im Ordner "PLC-Datentypen" den Befehl "Neuen Datentyp hinzufügen" doppelklicken.
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Benennen Sie den PLC-Datentyp um in "UDT_SiloContents".
3. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   MyBool > Datentyp: BOOL
   MyInt > Datentyp: INT
   MyWord > Datentyp: WORD

UDT_SiloContents

		Name	Data type
1	...	MyBool	Bool
2	...	MyInt	Int
3	...	MyWord	Word

Legen Sie drei globale Datenbausteine an.

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_SiloWater", "DB_SiloSugar" und "DB_SiloMilk" an.
4. Wählen Sie als Typ der Datenbausteine "UDT_SiloContents" aus.
5. Klicken Sie auf "OK".

DB_SiloWater

		Name	Data type
1		Static	
2		MyBool	Bool
3		MyInt	Int
4		MyWord	Word

DB_SiloSugar

		Name	Data type
1		Static	
2		MyBool	Bool
3		MyInt	Int
4		MyWord	Word

DB_SiloMilk

		Name	Data type
1		Static	
2		MyBool	Bool
3		MyInt	Int
4		MyWord	Word

Erstellen Sie eine Funktion, um die Werte der Datenbaustein-Variablen auszulesen und in einen PLC-Datentyp zu schreiben.

1. Legen Sie eine SCL-Funktion an und benennen Sie sie "FC_AccessGroupSiloRead".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_AccessGroupSiloRead

	Name	Data type
1	Input	
2	Index	DInt
3	Output	
4	InOut	
5	Temp	
6	Constant	
7	Return	
8	FC_AccessGroupSiloRead	"UDT_SiloContents"

3. Schreiben Sie folgenden Programmcode:

```
1
2  CASE #Index OF
3
4      1: // First case: "DB_SiloWater"
5          #FC_AccessGroupSiloRead := "DB_SiloWater";
6      2: // Second case: "DB_SiloSugar"
7          #FC_AccessGroupSiloRead := "DB_SiloSugar";
8      3: // Third case: "DB_SiloMilk"
9          #FC_AccessGroupSiloRead := "DB_SiloMilk";
10
11     ELSE // Optional case
12         #FC_AccessGroupSiloRead := "DB_SiloWater";
13
14 END_CASE;
```

4. Rufen Sie die Funktion "FC_AccessGroupSiloRead" im OB1 auf:

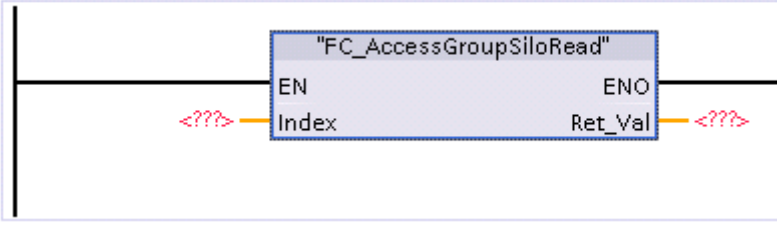

Je nachdem, welche Zahl Sie am Parameter
Index angeben (1, 2 oder 3) wird der erste, zweite oder dritte Fall der Anweisung "FC_AccessGroupSiloRead" ausgeführt.

Siehe auch

Indirekte Adressierung in AWL (Seite 139)

Indirektes Adressieren in SCL (Seite 136)

Operanden indirekt adressieren (Seite 119)

Indirektes Adressieren über Zeiger

Beschreibung

Für die indirekte Adressierung wird ein besonderes Datenformat benötigt, das die Adresse und gegebenenfalls auch den Bereich und den Datentyp eines Operanden enthält. Dieses Datenformat wird als Zeiger bezeichnet. Folgende Typen von Zeigern stehen Ihnen zur Verfügung:

- Referenzen (S7-1500)
- VARIANT (S7-1200/1500)
- POINTER (S7-1500)
- ANY (S7-1500, nur Bausteine mit Standardzugriff)

Weitere Informationen zu den Zeiger-Datentypen finden Sie unter "Siehe auch".

Hinweis

POINTER in SCL

In SCL ist die Verwendung von POINTER eingeschränkt. Es besteht lediglich die Möglichkeit, ihn an aufgerufene Bausteine weiterzugeben.

Beispiel

Das folgende Beispiel zeigt eine indirekte Adressierung mit einem bereichsinternen Pointer:

Adressierung in AWL	Erläuterung
L P#10.0	// Pointer (P#10.0) in Akkumulator 1 laden
T MD20	// Pointer in den Operanden MD20 übertragen
L MW [MD20]	// MW10 in Akkumulator 1 laden
....	// Beliebiges Programm
L MD [MD20]	// MD10 in Akkumulator 1 laden
....	// Beliebiges Programm
= M [MD20]	// Bei VKE=1 das Merkerbit M10.0 setzen

Der Pointer P#10.0 wird in den Operanden MD20 übertragen. Wenn der Operand MD20 in eckigen Klammern programmiert ist, wird dieser zur Laufzeit durch die Adresse ersetzt, die im Pointer enthalten ist.

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Zeiger (Seite 315)

Indirektes Indizieren von ARRAY-Komponenten

ARRAY-Zugriffe mit variablem Index

Zur Adressierung von ARRAY-Elementen können Sie als Index entweder Konstanten oder Variablen vom Datentyp Ganzzahl angeben. Dabei sind Ganzzahlen mit einer Länge bis 32 Bit zulässig.

Bei der indirekten Adressierung mithilfe einer Variable wird der Index erst zur Programmlaufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

Syntax

Für das indirekte Indizieren eines ARRAYs wird die folgende Syntax verwendet:

```
"MyDB".MyArray[#i]   // eindimensionales ARRAY
"MyDB".MyArray[#i].a // eindimensionales ARRAY of STRUCT
"MyDB".MyArray[#i,#j]// mehrdimensionales ARRAY
"MyDB".MyArray[#i].a // mehrdimensionales ARRAY of STRUCT
```

Die Syntax besteht aus folgenden Teilen:

Teil	Beschreibung
MyDB	Name des Datenbausteins, in dem sich das ARRAY befindet
MyArray	Variable vom Datentyp ARRAY
i, j	PLC-Variablen vom Datentyp Ganzzahl, die als Zeiger verwendet werden
a	Weitere Teilvariable der Struktur

Hinweis

Wenn Sie einen Baustein aufrufen und diesem eine indirekt indizierte ARRAY-Komponente ("MyDB".MyArray[#i]) als Durchgangsparameter (InOut) übergeben, können Sie während der Bausteinbearbeitung den Wert der Indexvariable [i] nicht ändern. Der Wert wird also immer in dieselbe ARRAY-Komponente zurückgeschrieben, aus der er gelesen wurde.

ARRAY-Komponenten mit den Anweisungen "FieldRead" und "FieldWrite" indizieren

Zur indirekten Indizierung von ARRAY-Komponenten in KOP und FUP können Sie auch die folgenden Anweisungen verwenden:

- FieldWrite - Feld schreiben
- FieldRead - Feld lesen

Siehe auch

Beispiele zur indirekten Indizierung von ARRAYs (Seite 128)

ARRAY-Komponenten indirekt adressieren (Seite 211)

Grundlagen zur indirekten Adressierung (Seite 119)

ARRAY-Komponenten adressieren (Seite 303)

ARRAY (Seite 296)

Beispiele zur indirekten Indizierung von ARRAYs

Beispiel: ARRAY-Zugriff mit variablem Index in KOP

Das folgende Beispiel zeigt das indirekte Indizieren einer ARRAY-Komponente am Beispiel von KOP. "MyArray" ist ein dreidimensionales ARRAY. #Tag_1, #Tag_2 und #Tag_3 sind Eingangsparameter vom Datentyp "Integer". Abhängig von deren Werten wird eine der Komponenten von MyArray in die Variable "MyTarget" kopiert.

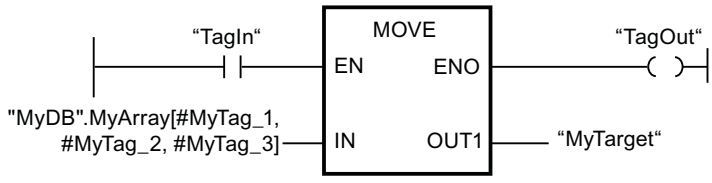

Beispiel: ARRAY-Zugriff mit variablem Index in SCL

Das folgende Beispiel zeigt, wie Sie mehrere Datenbausteine über einen variablen Index adressieren. Im Beispiel werden fünf Drehzahlachsen verwendet. Diese Achsen sollen mit SCL iterierend in einer FOR-Schleife bearbeitet werden.

1. Legen Sie zunächst fünf Drehzahlachsen an. Alle fünf Achsen müssen vom gleichen Typ sein.
   Für jede Achse wird in der Projektnavigation ein Datenbaustein angelegt:

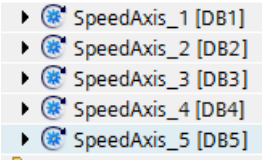

2. Erzeugen Sie anschließend einen globalen Datenbaustein und deklarieren Sie ein ARRAY vom Datentyp ANY mit fünf Komponenten.

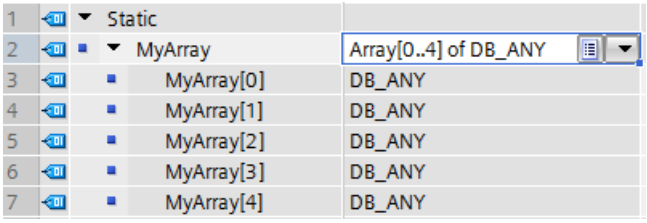

1	Static	
2	MyArray	Array[0..4] of DB_ANY
3	MyArray[0]	DB_ANY
4	MyArray[1]	DB_ANY
5	MyArray[2]	DB_ANY
6	MyArray[3]	DB_ANY
7	MyArray[4]	DB_ANY

3. Zur Initialisierung weisen Sie den einzelnen ARRAY-Elementen im Anlauf-OB die symbolischen Namen der fünf Achsen zu.

```
1  "MyGlobalDB".MyArray[0] := "SpeedAxis_1";
2  "MyGlobalDB".MyArray[1] := "SpeedAxis_2";
3  "MyGlobalDB".MyArray[2] := "SpeedAxis_3";
4  "MyGlobalDB".MyArray[3] := "SpeedAxis_4";
5  "MyGlobalDB".MyArray[4] := "SpeedAxis_5";
6
```

4. In SCL wird mit Hilfe einer FOR-Schleife iterierend auf die einzelnen Achsen zugegriffen. Die Achsen werden nacheinander an die Funktion "MaximumVelocity" zur Bearbeitung übergeben.

1	Input							
2	Output							
3	InOut							
4	CurrentMaximum	LReal	0.0	☑	☐	☑	☐	
5	Static							
6	Temp							
7	Index	DInt						
8	Constant							

IF... CASE... OF... FOR... TO DO... WHILE... DO... (*...*) REGION

```
1 FOR #Index := 0 TO 9 BY 1 DO
2     #CurrentMaximum := "MaximumVelocity" (CurrentAxis:="MyGlobalDB".MyArray[#Index]);
3 END_FOR;
4
```

Beispiel: ARRAY-Zugriff mit variablem Index in AWL

Ein Anwendungsbeispiel für einen ARRAY-Zugriff mit variablem Index in AWL finden Sie unter "ARRAY-Elemente indirekt adressieren". (Seite 211)

Siehe auch

Indirektes Indizieren von ARRAY-Komponenten (Seite 127)

Grundlagen zur indirekten Adressierung (Seite 119)

ARRAY-Komponenten adressieren (Seite 303)

ARRAY (Seite 296)

Indirektes Adressieren einzelner Zeichen eines STRINGs

Beschreibung

Zur Adressierung der einzelnen Zeichen eines STRINGs oder WSTRINGs können Sie als Index nicht nur Konstanten sondern auch Variablen angeben. Die Variablen müssen vom Datentyp Ganzzahl sein. Bei Verwendung von Variablen wird der Index zur Laufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

Wenn Sie beim Bausteinaufruf einen variabel indizierten STRING oder WSTRING an einen Durchgangsparameter (InOut) übergeben, beachten Sie bitte: Die Indexvariable [i] wird einmal zu Beginn des Bausteinaufrufs gelesen und kann während der Bearbeitung des aufgerufenen Bausteins von diesem nicht geändert werden.

Hinweis

Überwachung von STRING-Zugriffen zur Laufzeit

Wenn zur Laufzeit ein STRING oder WSTRING geschrieben wird, der die definierte Länge überschreitet, kann es zu unerwünschten Reaktionen im Programm kommen. In S7-1200/1500 wird das Überschreiten der STRING- oder WSTRING-Länge überwacht. Bei lesendem Zugriff auf den STRING erhalten Sie das Zeichen '$00' oder '$0000', ein schreibender Zugriff in den STRING wird nicht ausgeführt. Wenn die Anweisung über den Freigabeausgang ENO verfügt, wird ENO auf den Signalzustand FALSE gesetzt. Die CPU geht nicht in STOP.

Syntax

Für das indirekte Indizieren eines STRINGs oder WSTRINGs wird die folgende Syntax verwendet:

```
"<Datenbaustein>".<STRING>["i"]
"<Datenbaustein>".<WSTRING>["i"]
```

Beispiel

Das folgenden Beispiel zeigt das indirekte Indizieren eines STRINGs am Beispiel von SCL. "STRING", "WSTRING", "CHAR" und "WCHAR" sind Variablen. "Tag_1" ist eine PLC-Variable vom Datentyp "Integer".

Adressierung in SCL	Erläuterung
`STRING["Tag_1"] := CHAR;`	`(*Indirekte Adressierung: Zuweisung von "CHAR" an das durch "Tag_1" spezifizierten Zeichen des STRINGs*)`
`WSTRING["Tag_1"] := WCHAR;`	`(*Indirekte Adressierung: Zuweisung von "WCHAR" an das durch "Tag_1" spezifizierten Zeichen des WSTRINGs*)`
`WCHAR := WSTRING["Tag_1"];`	`(*Indirekte Adressierung: Zuweisung des durch "Tag_1" spezifizierten Zeichen des WSTRINGs an WCHAR*)`

Siehe auch

STRING (Seite 277)

WSTRING (Seite 280)

Indirekte Adressierung in AWL

Grundlagen zu Adressregistern

Einführung

Zur indirekten Adressierung von Operanden stehen Ihnen zwei Adressregister zur Verfügung, das Adressregister 1 (AR1) und das Adressregister 2 (AR2). Die Adressregister sind gleichwertig und haben eine Länge von 32 Bits. In den Adressregistern können Sie bereichsinterne und bereichsübergreifende Zeiger speichern. Die gespeicherten Daten können Sie im Programm aufrufen, um die Adresse eines Operanden zu bestimmen.

Der Datenaustausch zwischen den Registern und den anderen verfügbaren Speicherbereichen erfolgt mithilfe von Lade- und Transferanweisungen.

Hinweis

In S7-1500 gelten besondere Regeln zum Datenaustausch über Adress- und Datenbausteinregister:

- Die Werte in den Registern bleiben nicht über Bausteingrenzen hinweg bestehen.
- Bei Sprachwechsel innerhalb eines Bausteins werden die Register rückgesetzt.
- Auf Daten in Bausteinen mit optimiertem Zugriff können Sie nur verweisen, wenn diese die Remanenzeinstellung "Im IDB setzen" haben.
- Es ist nicht möglich, mit Hilfe der Adressregister (bereichsübergreifend) auf Lokaldaten in Bausteinen mit optimiertem Zugriff zu verweisen.

Siehe auch

LAR1: AR1 mit Inhalt von Akkumulator 1 laden (Seite 5003)

LAR1 <D>: AR1 mit Doppelwort oder Bereichszeiger laden (Seite 5004)

LAR1 AR2: AR1 mit Inhalt von AR2 laden (Seite 5005)

LAR2: AR2 mit Inhalt von Akkumulator 1 laden (Seite 5006)

TAR: AR1 und AR2 tauschen (Seite 5011)

TAR1: AR1 in Akkumulator 1 transferieren (Seite 5012)

TAR1 <D>: AR1 in Doppelwort transferieren (Seite 5013)

TAR1 AR2: AR1 in AR2 transferieren (Seite 5014)

TAR2: AR2 in Akkumulator 1 transferieren (Seite 5014)

+AR1: Akkumulator 1 zu AR1 addieren (Seite 5104)

+AR2: Akkumulator 1 zu AR2 addieren (Seite 5105)

Indirektes Adressieren in AWL (Seite 133)

Datenaustausch in AWL (Seite 7557)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Laden und Transferieren (Seite 1618)

Remanenz einstellen (Seite 7338)

Indirektes Adressieren in AWL

In AWL stehen Ihnen folgende Möglichkeiten der indirekten Adressierung zur Verfügung:

- Speicherindirekte Adressierung
- Registerindirekte bereichsinterne Adressierung
- Registerindirekte bereichsübergreifende Adressierung

Speicherindirekte Adressierung

Bei speicherindirekter Adressierung legen Sie die Adresse in einer Variablen ab. Diese Variable kann vom Datentyp WORD oder DWORD sein. Die Variable kann in den Speicherbereichen "Daten" (DB oder DI), "Merker" (M) oder "temporäre Lokaldaten" (L) liegen. In S7-1500 können auch FB-Parameter zum Ablegen der Adresse verwendet werden. Wenn die Variable in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein.

Das folgende Beispiel zeigt Anwendungen der speicherindirekten Adressierung:

Adressierung in AWL	Erläuterung
`U E [MD 2]`	`// Führe eine UND-Verknüpfung mit einem variablen Eingangsbit aus. Die Adresse des Eingangsbits befindet sich in Merkerdoppelwort MD2.`
`= DIX [DBD 2]`	`// Weise das VKE einem variablen Datenbit zu. Die Adresse des Datenbits befindet sich im Datendoppelwort DBD2.`
`L EB [DID 4]`	`// Lade ein variables Eingangsbyte in AKKU 1. Die Adresse des Eingangsbytes befindet sich im Instanzdoppelwort DID4.`
`AUF DB [LW 2]`	`// Öffne einen variablen Datenbaustein. Die Nummer des Datenbausteins befindet sich im Lokaldatenwort LW2.`

Registerindirekte bereichsinterne Adressierung

Die registerindirekte Adressierung verwendet eines der Adressregister (AR1 oder AR2), um die Adresse des Operanden aufzunehmen.

Bei der registerindirekten bereichsinternen Adressierung indizieren Sie lediglich die Bitadresse und die Byteadresse über das Adressregister (z. B. `P#10.0`). Den Speicherbereich, für den die Adresse im Adressregister gelten soll, geben Sie erst bei der Programmierung der Anweisung an. Die Adresse im Adressregister bewegt sich dann in dem Speicherbereich, der in der Anweisung angegeben ist.

Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (PE oder PA), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI). Wenn der Operand in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein.

Bei der Eingabe der registerindirekten bereichsinternen Adressierung geben Sie nach der Angabe des Adressregisters einen Versatz an, der zum Inhalt des Adressregisters addiert wird, ohne das Adressregister zu verändern. Dieser Versatz hat ebenfalls das Format eines Zeigers. Die Angabe des Zeigers ist zwingend und muss als Konstante erfolgen (z. B. `P#0.0` oder `P#2.0`).

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsinternen Adressierung:

AWL	**Erläuterung**
`LAR1 P#10.0`	`// Zeiger (P#10.0) in Adressregister 1 laden`
`L EW [AR1, P#2.0]`	`// Inhalt des Adressregisters 1 (P#10.0) um Versatz P#2.0 erhöhen.`
	`// Inhalt des Eingangswort EW12 in Akkumulator 1 laden`
`L EW [AR1, P#0.0]`	`// Inhalt des Adressregisters 1 (P#10.0) um Versatz P#0.0 erhöhen.`
	`// Inhalt des Eingangswort EW10 in Akkumulator 1 laden`

Registerindirekte bereichsübergreifende Adressierung

Bei der registerindirekten bereichsübergreifenden Adressierung indizieren Sie die gesamte Adresse des Operanden, d. h. die Bit- und Byteadresse sowie den Speicherbereich, über das Adressregister. Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (P), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI). Wenn der Operand in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein oder der Operand muss die Remanenzeinstellung "Im IDB setzen" haben.

In der Anweisung programmieren Sie nur die Operandenbreite. Mögliche Operandenbreiten sind Bit, Byte, Wort und Doppelwort.

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsübergreifenden Adressierung:

`LAR1 P#M10.0`	`// Bereichsübergreifenden Zeiger (P#M10.0)in Adressregister 1 laden`
`L W [AR1, P#2.0]`	`// Inhalt des Adressregisters 1 (P#M10.0) um Versatz P#2.0 erhöhen.`
	`// Inhalt des Merkerwort MW12 in Akkumulator 1 laden`
`LAR1 P#A10.0`	`// Bereichsübergreifenden Zeiger (P#A10.0)in Adressregister 1 laden`
`L W [AR1, P#2.0]`	`// Inhalt des Adressregisters 1 (P#A10.0) um Versatz P#2.0 addieren`
	`// Inhalt des Ausgangswort AW12.0 in Akkumulator 1 laden`

Hinweis

Besonderheiten in S7-1500

In S7-1500 gelten besondere Regeln zum Datenaustausch über Adress- und Datenbausteinregister:

- Die Werte in den Registern bleiben nicht über Bausteingrenzen hinweg bestehen. Auch bei Sprachwechsel innerhalb eines Bausteins werden die Register rückgesetzt.
- Wenn Sie über registerindirekte Adressierung auf einen Operanden vom Typ BYTE, WORD oder DWORD zugreifen, muss die Adresse auf einer Byte-Grenze beginnen.
  Beispiele:
  LAR1 P#0.0
  L MW [AR1, P#0.0] // P#0.0 + P#0.0 = P#0.0 - Die Adressierung ist zulässig, da der Zeiger P#0.0 auf eine Bytegrenze zeigt.
  L MW [AR1, P#2.1] // P#0.0 + P#2.1 = P#2.1 - Die Adressierung ist nicht zulässig, da der Zeiger P#2.1 nicht auf eine Bytegrenze zeigt.

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Indirektes Adressieren von Datenbausteinen (Seite 136)

ARRAY-Komponenten adressieren (Seite 303)

Grundlagen zu Adressregistern (Seite 132)

Datenaustausch in AWL (Seite 7557)

Indirekte Adressierung

Indirektes Adressieren über Pointer

Beschreibung

Für die indirekte Adressierung wird ein besonderes Datenformat benötigt, das die Adresse und gegebenenfalls auch den Bereich und den Datentyp eines Operanden enthält. Dieses Datenformat wird als Pointer (Zeiger) bezeichnet. Folgende Typen von Pointern stehen Ihnen zur Verfügung:

- POINTER (S7-300/400)
- ANY (S7-300/400)
- VARIANT (S7-1200/1500)

Weitere Informationen zu den Pointer-Datentypen finden Sie unter "Siehe auch".

Hinweis

In SCL ist die Verwendung des Datentyps Pointer eingeschränkt. Es besteht lediglich die Möglichkeit, ihn an aufgerufene Bausteine weiterzugeben.

Beispiel

Das folgende Beispiel zeigt eine indirekte Adressierung mit einem bereichsinternen Pointer:

Adressierung in AWL	Erläuterung
`L P#10.0`	`// Pointer (P#10.0) in Akkumulator 1 laden`
`T MD20`	`// Pointer in den Operanden MD20 übertragen`
`L MW [MD20]`	`// MW10 in Akkumulator 1 laden`
`....`	`// Beliebiges Programm`
`L MD [MD20]`	`// MD10 in Akkumulator 1 laden`
`....`	`// Beliebiges Programm`
`= M [MD20]`	`// Bei VKE=1 das Merkerbit M10.0 setzen`

Der Pointer P#10.0 wird in den Operanden MD20 übertragen. Wenn der Operand MD20 in eckigen Klammern programmiert ist, wird dieser zur Laufzeit durch die Adresse ersetzt, die im Pointer enthalten ist.

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Zeiger (Seite 315)

Indirektes Adressieren in SCL

Indirektes Adressieren von Datenbausteinen

Indirekte Adressierung von Datenbausteinen in SCL

Um Datenbausteine indirekt zu adressieren, können Sie die Konvertierungsfunktion WORD_TO_BLOCK_DB verwenden. Dabei wird die DB-Nummer als Variable oder Ausdruck mit dem Datentyp WORD angegeben.

Syntax

Für die indirekte Adressierung eines Datenbausteins wird die folgende Syntax verwendet:
`WORD_TO_BLOCK_DB(Index).Operandenkennzeichen(Adresse)`

Beispiele

Beispiel 1: Die globale Variable "Adressindex" vom Datentyp WORD wird als Nummer des DBs verwendet.

Adressierung in SCL

```
%M0.0:=WORD_TO_BLOCK_DB("Adressindex").DX(0,0);
%MW0:=WORD_TO_BLOCK_DB("Adressindex").DW(4);
```

Beispiel 2: Die globale Variable "Adressindex" vom Datentyp WORD wird als Nummer des DBs verwendet. Das Datenelement innerhalb des DBs wird ebenfalls über einen Index spezifiziert.

Adressierung in SCL

```
%M0.0:=WORD_TO_BLOCK_DB("Adressindex").DX(#i,#y);
%MW0:=WORD_TO_BLOCK_DB("Adressindex").DW(#y);
```

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Indirektes Adressieren in AWL (Seite 133)

ARRAY-Komponenten adressieren (Seite 303)

Indirektes Adressieren von Variablen

Indirekte Adressierung von Variablen in SCL

Die indirekte Adressierung geschieht ähnlich wie die absolute Adressierung. Anstelle der Adresse wird ein Offset in runder Klammer spezifiziert. Das Offset besteht aus einer Bytevariablen, bei booleschen Operanden aus einer Byte- und einer Bitvariablen. Byte- und Bitvariablen müssen vom Datentyp INT sein.

Zeiten und Zähler aus der PLC-Variablentabelle können auf diese Weise nicht indirekt adressiert werden.

Syntax

Für das indirekte Adressieren einer globalen Variablen wird die folgende Syntax verwendet:
`Operandenkennzeichen(Bytevariable)`
`Operandenkennzeichen(Bytevariable.Bitvariable)`

Für das indirekte Adressieren einer DB-Variablen wird die folgende Syntax verwendet:
`MyDB.Operandenkennzeichen(Bytevariable)`
`MyDB.Operandenkennzeichen(Bytevariable.Bitvariable)`

Beispiele

Beispiel 1: Abhängig vom Wert der Laufvariable i wird eines der Eingangsworte 2 bis 8 auf 0 gesetzt.

Adressierung in SCL

```
#i:=2;
FOR #i := 2 TO 8 DO
    %EW(#i) := 0 ;
END_FOR;
```

Beispiel 2: Abhängig vom Wert der Laufvariable i wird eines der Worte 2 bis 8 in DB10 auf 0 gesetzt.

Adressierung in SCL

```
#i:=2;
FOR #i := 2 TO 8 DO
    %DB10.DW(#i) := 0 ;
END_FOR;
```

Indirektes Adressieren von ARRAY-Komponenten

Beschreibung

Zur Adressierung der Komponenten eines ARRAYs können Sie als Index neben Konstanten auch Variablen vom Datentyp Ganzzahl angeben. Bei Verwendung von Variablen wird der Index zur Laufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

Hinweis

Die Indexvariable [i] wird einmal zu Beginn des Bausteinaufrufes gelesen und kann während der Bearbeitung des aufgerufenen Bausteins von diesem nicht geändert werden.

Wenn Sie einen Baustein aufrufen und diesem eine indirekt indizierte ARRAY-Komponente ("<Datenbaustein>".<ARRAY>["i"]) als Durchgangsparameter (InOut) übergeben, können Sie während der Bausteinbearbeitung den Wert der Indexvariable nicht ändern. Der Wert wird also immer in dieselbe ARRAY-Komponente zurückgeschrieben, aus der er gelesen wurde.

Syntax

Für das indirekte Indizieren eines ARRAYs wird die folgende Syntax verwendet:

```
"<Datenbaustein>".<ARRAY>["i"] // eindimensionales ARRAY
"<Datenbaustein>".<ARRAY>["i"].a // eindimensionales ARRAY of STRUCT
"<Datenbaustein>".<ARRAY>["i", "j"] // mehrdimensionales ARRAY
"<Datenbaustein>".<ARRAY>["i", "j"].a // mehrdimensionales ARRAY of
STRUCT
```

Die Syntax besteht aus folgenden Teilen:

Teil	Beschreibung
Datenbaustein	Name des Datenbausteins, in dem sich das ARRAY befindet
ARRAY	Variable vom Datentyp ARRAY
i, j	PLC-Variablen vom Datentyp Ganzzahl, die als Zeiger verwendet werden
a	Weitere Teilvariable der Struktur

Beispiele

Die folgenden Beispiele zeigen das indirekte Indizieren einer ARRAY-Komponente am Beispiel von SCL. MOTOR ist ein eindimensionales ARRAY_of_INT mit drei Zeilen. VALUES ist eine PLC-Variable vom Datentyp "Integer".

Adressierung in SCL	Erläuterung
`MOTOR[2] := VALUES;`	`(*Direkte Adressierung: Zuweisung von VALUES zur zweiten Zeile des ARRAYs MOTOR*)`
`MOTOR["Tag_1"] := VALUES;`	`(*Indirekte Adressierung: Zuweisung von VALUES zur durch "Tag_1" spezifizierten Zeile des ARRAYs MOTOR*)`
`#MOTOR["Tag_2"+"Tag_3"] := #Values;`	`(*Indirekte Adressierung: Zuweisung von VALUES zur durch den Ausdruck "Tag_2"+"Tag_3" spezifizierten Zeile des ARRAYs MOTOR*)`

Siehe auch

ARRAY-Komponenten adressieren (Seite 303)

Indirekte Adressierung in AWL

Grundlagen zu Adressregistern

Einführung

Zur indirekten Adressierung von Operanden stehen Ihnen zwei Adressregister zur Verfügung, das Adressregister 1 (AR1) und das Adressregister 2 (AR2). Die Adressregister sind gleichwertig und haben eine Länge von 32 Bits. In den Adressregistern können Sie bereichsinterne und bereichsübergreifende Zeiger speichern. Die gespeicherten Daten können Sie im Programm aufrufen, um die Adresse eines Operanden zu bestimmen.

Der Datenaustausch zwischen den Registern und den anderen verfügbaren Speicherbereichen erfolgt mithilfe von Lade- und Transferanweisungen.

Siehe auch

LAR1: AR1 mit Inhalt von Akkumulator 1 laden (Seite 5003)

LAR1 <D>: AR1 mit Doppelwort oder Bereichszeiger laden (Seite 5004)

LAR1 AR2: AR1 mit Inhalt von AR2 laden (Seite 5005)

LAR2: AR2 mit Inhalt von Akkumulator 1 laden (Seite 5006)

TAR: AR1 und AR2 tauschen (Seite 5011)

TAR1: AR1 in Akkumulator 1 transferieren (Seite 5012)

TAR1 <D>: AR1 in Doppelwort transferieren (Seite 5013)

TAR1 AR2: AR1 in AR2 transferieren (Seite 5014)

TAR2: AR2 in Akkumulator 1 transferieren (Seite 5014)

+AR1: Akkumulator 1 zu AR1 addieren (Seite 5104)

+AR2: Akkumulator 1 zu AR2 addieren (Seite 5105)

Indirektes Adressieren in AWL (Seite 140)

Datenaustausch in AWL (Seite 7557)

Laden und Transferieren (Seite 1618)

Indirektes Adressieren in AWL

In AWL stehen Ihnen folgende Möglichkeiten der indirekten Adressierung zur Verfügung:

- Speicherindirekte Adressierung
- Registerindirekte bereichsinterne Adressierung
- Registerindirekte bereichsübergreifende Adressierung

Speicherindirekte Adressierung

Bei speicherindirekter Adressierung legen Sie die Adresse in einer Variablen ab. Diese Variable kann vom Datentyp WORD oder DWORD sein. Die Variable kann in den Speicherbereichen "Daten" (DB oder DI), "Merker" (M) oder "temporäre Lokaldaten" (L) liegen.

Das folgende Beispiel zeigt Anwendungen der speicherindirekten Adressierung:

Adressierung in AWL	Erläuterung
U E [MD 2]	// Führe eine UND-Verknüpfung mit einem variablen Eingangsbit aus. Die Adresse des Eingangsbits befindet sich in Merkerdoppelwort MD2.
= DIX [DBD 2]	// Weise das VKE einem variablen Datenbit zu. Die Adresse des Datenbits befindet sich im Datendoppelwort DBD2.
L EB [DID 4]	// Lade ein variables Eingangsbyte in AKKU 1. Die Adresse des Eingangsbytes befindet sich im Instanzdoppelwort DID4.
AUF DB [LW 2]	// Öffne einen variablen Datenbaustein. Die Nummer des Datenbausteins befindet sich im Lokaldatenwort LW2.

Registerindirekte bereichsinterne Adressierung

Die registerindirekte Adressierung verwendet eines der Adressregister (AR1 oder AR2), um die Adresse des Operanden aufzunehmen.

Bei der registerindirekten bereichsinternen Adressierung indizieren Sie lediglich die Bitadresse und die Byteadresse über das Adressregister (z. B. `P#10.0`). Den Speicherbereich, für den die Adresse im Adressregister gelten soll, geben Sie erst bei der Programmierung der Anweisung an. Die Adresse im Adressregister bewegt sich dann in dem Speicherbereich, der in der Anweisung angegeben ist.

Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (PE oder PA), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI).

Bei der Eingabe der registerindirekten bereichsinternen Adressierung geben Sie nach der Angabe des Adressregisters einen Versatz an, der zum Inhalt des Adressregisters addiert wird, ohne das Adressregister zu verändern. Dieser Versatz hat ebenfalls das Format eines Zeigers. Die Angabe des Zeigers ist zwingend und muss als Konstante erfolgen (z. B. `P#0.0` oder `P#2.0`).

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsinternen Adressierung:

AWL	**Erläuterung**
`LAR1 P#10.0`	`// Zeiger (P#10.0) in Adressregister 1 laden`
`L EW [AR1, P#2.0]`	`// Inhalt des Adressregisters 1 (P#10.0) um Versatz P#2.0 erhöhen.`
	`// Inhalt des Eingangswort EW12 in Akkumulator 1 laden`
`L EW [AR1, P#0.0]`	`// Inhalt des Adressregisters 1 (P#10.0) um Versatz P#0.0 erhöhen.`
	`// Inhalt des Eingangswort EW10 in Akkumulator 1 laden`

Registerindirekte bereichsübergreifende Adressierung

Bei der registerindirekten bereichsübergreifenden Adressierung indizieren Sie die gesamte Adresse des Operanden, d.h. die Bit- und Byteadresse sowie den Speicherbereich, über das Adressregister. Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (P), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI). In der Anweisung programmieren Sie nur die Operandenbreite. Mögliche Operandenbreiten sind Bit, Byte, Wort und Doppelwort.

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsübergreifenden Adressierung:

`LAR1 P#M10.0`	`// Bereichsübergreifenden Zeiger (P#M10.0)in Adressregister 1 laden`
`L W [AR1, P#2.0]`	`// Inhalt des Adressregisters 1 (P#M10.0) um Versatz P#2.0 erhöhen.`
	`// Inhalt des Merkerwort MW12 in Akkumulator 1 laden`
`LAR1 P#A10.0`	`// Bereichsübergreifenden Zeiger (P#A10.0)in Adressregister 1 laden`

```
L W [AR1, P#2.0]        // Inhalt des Adressregisters 1 (P#A10.0) um Versatz P#2.0 ad-
                        dieren
                        // Inhalt des Ausgangswort AW12.0 in Akkumulator 1 laden
```

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)
Indirektes Adressieren von Datenbausteinen (Seite 136)
ARRAY-Komponenten adressieren (Seite 303)
Grundlagen zu Adressregistern (Seite 139)
Datenaustausch in AWL (Seite 7557)

1.5 Programmablauffehler behandeln

1.5.1 Fehlerursachen

Einführung

Im TIA Portal gibt es unterschiedliche Ursachen für verschiedene Fehlerqualitäten, auf die Sie mit unterschiedlichen Mechanismen reagieren können. Die Mechanismen, wie Sie auf einen Fehler reagieren können, sind Programmiersprachen unabhängig. Nur die Darstellung der einzelnen Mechanismen ist von der jeweiligen Programmiersprache abhängig.

Was als Fehler gilt, hängt von der Situation ab. Der Überlauf bei einer Addition z. B. kann einen Fehler darstellen, weil nicht das gewünschte Ergebnis geliefert wird. In manchen Fällen ist der Additionsüberlauf aber ein akzeptiertes und definiertes Verhalten und damit kein Fehler.

Bei der Erstellung Ihres Programmcodes, muss Ihnen auch bewusst sein, dass bestimmte Situationen auftreten können. Wenn Sie z. B. eine Kommunikation programmieren, kann die Verbindung jederzeit abbrechen. Damit Sie auf diese Situation vorbereitet sind, müssen Sie eine entsprechende Fehlerreaktion einbauen, denn ein Verbindungsabbruch hindert den Programmbaustein T_SEND daran, eine Nachricht zu übermitteln. Daher ist es wichtig, dass der Verbindungsabbruch gemeldet und dadurch bekannt wird, dass die Übermittlung einer Nachricht nicht mehr funktioniert. Da die Anweisung T_SEND den Verbindungsabbruch nicht verhindern kann, ist für T_SEND die Fehlerausgabe ein korrektes Verhalten. Diese Ausgabe nicht zu beachten kann als Fehler des Programmerstellers betrachtet werden.

Im Folgenden wird immer von einem Fehler gesprochen, auch wenn es sich dabei um das definierte Verhalten handelt.

Unterschiedliche Ursachen für einen Fehler

Die Ursachen eines Fehlers können folgendermaßen unterschieden werden:

	Ungünstige Parameterwerte	Ungünstige Programmierung	Ausfall von Ressourcen
Beschreibung	Fehler, die direkt an der Anweisung behandelt werden.	Programmier- oder Zugriffsfehler, die zur Laufzeit zum Abbruch der Ausführung einer Anweisung führen.	Fehler, die durch das Betriebssystem behandelt werden, und auf die Sie im Programmcode eine Reaktion programmieren können.
Fehlerqualität	Behandelter Fehler	Synchroner Fehler	Asynchroner Fehler

	Ungünstige Parameterwerte	Ungünstige Programmierung	Ausfall von Ressourcen
Beispiel	Überlauf bei einer arithmetischen Anweisung	Ungünstige Programmierung: • Abfragen eines nicht vorhandenen Peripherie-Eingangs • Beim Zugriff auf ein ARRAY mittels variablem Index liegt der Wert des Index außerhalb der gültigen ARRAY-Grenzen	Auftreten eines bestimmten Ereignisses außerhalb Ihres Programmcodes.
Reaktion des Programms bzw. des Betriebssystems	Anweisungen, die mit dem Freigabeausgang ENO verknüpft sind, werden nicht ausgeführt.	Wenn Sie keinen Fehler-OB programmiert haben, reagiert das Betriebssystem CPU-abhängig.	1. Ist dem Ereignis kein Organisationsbaustein (OB) zugeordnet, dann führt das Betriebssystem beim Auftreten des Ereignisses die voreingestellte Systemreaktion durch. 2. Ist dem Ereignis ein Organisationsbaustein (OB) zugeordnet, dann wird dieser aufgerufen.

	Ungünstige Parameterwerte	Ungünstige Programmierung	Ausfall von Ressourcen
Mechanismen zur Fehlerbehandlung im Programm	Hierfür stehen Ihnen je nach Anweisung unterschiedliche lokale Fehlerbehandlungsmöglichkeiten [2] zur Verfügung: • EN-/ENO-Mechanismus • Die Ausgangsparameter: – RET_VAL – STATUS – ERROR	Globale Fehlerbehandlung [1]: • Programmablauffehler-OB • Programmierfehler-OB • Peripheriezugriffsfehler-OB Lokale Fehlerbehandlung [2] mit den Anweisungen: • GET_ERROR • GET_ERR_ID	Fehler-Organisationsbausteine (OB): • Zeitfehler (OB 80) • Diagnose-Alarm (OB 82) • Ziehen/Stecken-Alarm (OB 83) • Baugruppenträgerfehler (OB 86) Mögliche Systemreaktionen ohne zugeordnetem Fehler-OB: • Das Ereignis wird vom Betriebssystem ignoriert. • Die CPU wechselt in den Betriebszustand STOP. • Es wird wenn möglich eine lokale Fehlerbehandlung durchgeführt. Bei zugeordnetem Fehler-OB wird dieser bei einem entsprechenden Ereignis aufgerufen. Ein Applikationsbeispiel zur Diagnose im Anwenderprogramm mit Auswertung der Fehler in den Fehler-OBs (https://support.industry.siemens.com/cs/ww/de/view/98210758)

[1] Die globale Fehlerbehandlung realisieren Sie mithilfe von Organisationsbausteinen.

[2] Die lokale Fehlerbehandlung programmieren Sie innerhalb Ihres Programmcodes.

Hinweis

Asynchrone Fehlerbehandlung

Bei einer CPU der Baureihe S7-1500 werden die Fehler-OBs asynchron aufgerufen. Daher werden die Peripheriezugriffsfehler- oder Programmierfehler-OBs möglicherweise nicht sofort beim Auftreten des Fehlers, sondern abhängig von der eingestellten Priorität, erst verzögert bearbeitet. Wenn weitere Fehler auftreten, bevor der Peripheriezugriffsfehler- oder Programmierfehler-OB fertig bearbeitet wurde, dann wird kein weiterer Peripheriezugriffsfehler- oder Programmierfehler-OB aufgerufen. Möchten Sie verhindern, dass Peripheriezugriffsfehler- oder Programmierfehler-OBs verworfen werden, stellen Sie die Priorität entsprechend hoch ein.

Siehe auch

Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

1.5.2 Mechanismen zur Fehlerbehandlung im Überblick

Überblick

Es gibt verschiedene Mechanismen, wie Sie mögliche Parameter-, Programmier- oder Zugriffsfehler abfangen können:

Mechanismus	Aufgabe	Fehlerbehandlung
Freigabeeingang EN oder IF-Anweisung	Verhinderung der Ausführung eines Programmcodes	lokal
Freigabeausgang ENO oder BIE-Bit	Hinweis auf einen Fehler	
Parameterausgänge RET_VAL, STATUS und ERROR		
Anweisungen GET_ERROR und GET_ERR_ID	Reaktion auf einen Fehler	
Organisationsbausteine		global

Lokale Fehlerbehandlung bei ungünstigen Parameterwerten

Die lokale Fehlerbehandlung ermöglicht es Ihnen nicht nur direkt nach dem Auftreten eines Fehlers zu reagieren, sondern auch innerhalb Ihres Programmcodes eine bestimmte Reaktion zu erzielen. Die lokale Fehlerbehandlung programmieren Sie direkt in einem Programmbaustein (OB, FB oder FC). Sie behandelt nur Fehler, die innerhalb dieses einzelnen Programmbausteins auftreten.

Vorteile der lokalen Fehlerbehandlung:

- Mithilfe der Fehlerinformation können Sie im Programmbaustein eine Reaktion auf den aufgetretenen Fehler programmieren.
- Programmierte Fehlerauswertungen und -reaktionen unterbrechen den Programmzyklus nicht.
- Die System-Performance wird durch die lokale Fehlerbehandlung nicht unnötig belastet. Wenn keine Fehler auftreten, werden programmierte Fehlerauswertungen und -reaktionen nicht ausgeführt.

Es stehen Ihnen die folgenden lokalen Fehlerbehandlungsmöglichkeiten zur Verfügung:

Art der Fehlerbehandlung	Gültigkeit	Erklärung
EN-/ENO-Mechanismus [1]	S7-300 / S7-400 / S7-1200 / S7-1500	Mithilfe des Freigabeausgangs ENO können Sie bestimmte Laufzeitfehler erkennen und behandeln. Die Ausführung nachfolgender Anweisungen ist vom Signalzustand des Freigabeausgangs abhängig. Durch die Nutzung des EN-/ENO-Mechanismus vermeiden Sie, dass es zu Programmabbrüchen kommt. Der Bausteinstatus wird in Form einer booleschen Variable weitergegeben. Weitere Informationen zum EN-/ENO-Mechanismus finden Sie hier: Grundlagen zum EN-/ENO-Mechanismus (Seite 150)
Ausgangsparameter STATUS und ERROR	S7-300 / S7-400 / S7-1200 / S7-1500	Mithilfe der Parameter STATUS und ERROR als Rückgabewerte von Systemfunktionsbausteinen (SFBs) können Sie bausteinspezifische Fehlerinformationen abfragen. Die Fehlerinformationen werden in einer vorgebenen Struktur ausgegeben. Weitere Informationen zu den Ausgangsparametern finden Sie in den Beschreibungen der einzelnen Anweisungen im Informationssystem.
Ausgangsparameter RET_VAL	S7-300 / S7-400 / S7-1200 / S7-1500	Mithilfe des Ausgangsparameters RET_VAL als Rückgabewert von Systemfunktionen (SFCs) können Sie sich allgemeine Fehlercodes oder spezifische Fehlercodes anzeigen lassen. Die allgemeinen Fehlercodes beziehen sich auf jede beliebige Anweisung und die spezifischen Fehlercodes beziehen sich nur auf eine konkrete Anweisung. Es kann maximal eine Variable vom Datentyp INT oder WORD ausgegeben werden. Weitere Informationen zum Ausgangsparameter RET_VAL finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
[1] Wenn durch die Parameter einer Anweisung keine Speicherzugriffsfehler aufgetreten sind, liefert der zugehörige Freigabeausgang ENO den Signalzustand "1" und die Ausgänge liefern gültige Werte, die Sie abfragen können.		

Globale und lokale Fehlerbehandlung bei ungünstiger Programmierung

Mithilfe der globalen und lokalen Fehlerbehandlung können Sie direkt auf aufgetretene Fehler reagieren, ohne dass die CPU in den Betriebszustand STOP wechselt. Für die Behandlung von Programmier- und Zugriffsfehlern stehen Ihnen folgende Möglichkeiten zur Verfügung:

Art der globalen Fehlerbehandlung	Gültigkeit	Erklärung
Programmablauffehler-OB (OB 85)	S7-300 / S7-400	Wenn Sie den OB 85 nicht verwenden, dann wechselt die CPU bei einem Programmablauffehler vom Betriebszustand RUN nach STOP und schreibt einen Eintrag in den Diagnosepuffer. Weitere Informationen zum OB 85 finden Sie hier: AUTOHOTSPOT
CPU interne Fehlerbehandlung bei Programmier- und Zugriffsfehlern	S7-1200	Ohne zusätzlichen Programmieraufwand erzeugt die CPU im Fehlerfall einen Eintrag im Diagnosepuffer und verbleibt im Betriebszustand RUN.
Programmierfehler-OB (OB 121)	S7-300/ S7-400 / S7-1500	Wenn Sie den OB 121 nicht verwenden, dann wechselt die CPU bei einem Programmierfehler vom Betriebszustand RUN nach STOP und schreibt einen Eintrag in den Diagnosepuffer. Weitere Informationen zum OB 121 finden Sie hier: S7-300 / S7-400: AUTOHOTSPOT S7-1500: AUTOHOTSPOT
Peripheriezugriffsfehler-OB (OB 122)	S7-300/ S7-400 / S7-1500	S7-300 / S7-400: Wenn Sie den OB 122 nicht verwenden, dann wechselt die CPU bei einem Zugriffsfehler vom Betriebszustand RUN nach STOP. AUTOHOTSPOT S7-1500: Bei einem Peripheriezugriffsfehler bleibt die CPU immer im Betriebszustand RUN und schreibt einen Eintrag in den Diagnosepuffer, auch wenn Sie den OB 122 nicht verwendet haben. Weitere Informationen zum OB 122 finden Sie hier: AUTOHOTSPOT
Weitere Informationen zum Auslesen des Diagnosepuffers finden Sie hier: AUTOHOTSPOT		

Die lokale Fehlerbehandlung mithilfe der Anweisungen GET_ERROR und GET_ERR_ID können Sie direkt in Ihrem Programmcode integrieren. Zusätzlich haben Sie die Möglichkeit, sich detaillierte Informationen zum Fehler geben zu lassen und in Ihrem Programm nahe der Fehlerstelle auszuwerten. Die lokale Fehlerbehandlung programmieren Sie direkt in einem

Programmbaustein (OB, FB oder FC). Sie behandelt nur Fehler, die innerhalb dieses einzelnen Programmbausteins auftreten.

Art der lokalen Fehlerbehandlung	Gültigkeit	Erklärung
Anweisungen GET_ERROR und GET_ERR_ID	S7-1200 / S7-1500	Die Anweisungen bieten Ihnen die Möglichkeit, eine Fehler-ID oder eine ausführliche Fehlerinformation zu erhalten, und eine direkte Reaktion im Programmcode zu programmieren. Wenn die Information zum ersten aufgetretenen Fehler abgefragt wird, wird der Speicherplatz des Fehlers im Systemspeicher wieder freigegeben. Wenn dann weitere Fehler auftreten, wird die Information zum nächsten aufgetretenen Fehler ausgegeben. Weitere Informationen zu den Anweisungen GET_ERROR und GET_ERR_ID und eine Übersicht zur Priorisierung der auftretenden Fehler finden Sie hier: Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Bei Verwendung der lokalen Fehlerbehandlung, Abfrage über die Anweisung GET_ERROR, gibt es die folgenden voreingestellten Reaktionen:

- bei einem Schreibfehler: Der Fehler wird ignoriert und die Bearbeitung des Programms fortgesetzt.
- bei einem Lesefehler: Die Bearbeitung des Programms wird bei arithmetischen Anweisungen mit dem Ersatzwert "0" fortgesetzt.
- bei einem Ausführungsfehler: Die Bearbeitung der Anweisung wird abgebrochen und die Programmbearbeitung mit der nächsten Anweisung fortgesetzt.

Vorteile der lokalen Fehlerbehandlung:

- Im Systemspeicher wird eine Fehlerinformation gespeichert, die Sie abfragen und auswerten können (z. B. mithilfe der Anweisungen GET_ERROR und GET_ERR_ID).
- Mithilfe der Fehlerinformation können Sie im Programmbaustein eine Reaktion auf den aufgetretenen Fehler programmieren.
- Programmierte Fehlerauswertungen und -reaktionen unterbrechen den Programmzyklus nicht.
- Die System-Performance wird durch die lokale Fehlerbehandlung weniger als bei der globalen Fehlerbehandlung belastet. Wenn keine Fehler auftreten, werden programmierte Fehlerauswertungen und -reaktionen nicht ausgeführt.
- Wenn für einen Programmbaustein eine lokale Fehlerbehandlung programmiert wurde, wird im Fehlerfall keine globale Fehlerbehandlung durchgeführt.

Hinweis

Um zu vermeiden, dass die CPU bei einem Fehler in den Betriebszustand STOP wechselt, müssen alle Programmier- und Peripheriezugriffsfehler entweder durch eine globale oder eine lokale Fehlerbehandlung abgefangen werden.

Beispiel

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere der oben genannten Möglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Siehe auch

EN-/ENO-Mechanismus (Seite 150)

1.5.3 EN-/ENO-Mechanismus

1.5.3.1 Grundlagen zum EN-/ENO-Mechanismus

Einführung

Mithilfe des Freigabeausgangs ENO können Sie bestimmte Laufzeitfehler erkennen und behandeln. Die Ausführung nachfolgender Anweisungen ist vom Signalzustand des Freigabeausgangs abhängig. Durch die Nutzung des EN-/ENO-Mechanismus vermeiden Sie, dass es zu Programmabbrüchen kommt. Der Bausteinstatus wird in Form einer booleschen Variable weitergegeben.

Der EN-/ENO-Mechanismus kann auf zwei Ebenen eingesetzt werden:

- Für einzelne Anweisungen (Anweisungs-ENO)

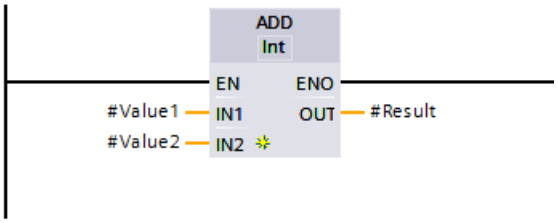

- Bei Programmbausteinaufrufen (Baustein-ENO)

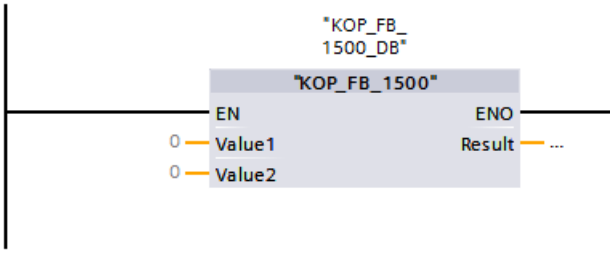

Der EN-/ENO-Mechanismus steht in KOP und FUP bereits beim Aufruf der einfachen und erweiterten Anweisungen in den Programmcodeboxen zur Verfügung.

Mit dem EN-/ENO-Mechanismus können Sie den Aufruf nachfolgender Anweisungen beeinflussen und das Herausspringen aus einem Programmbaustein mithilfe der Anweisung "RET: Zurück springen" individuell gestalten. Dabei kann auch dem Freigabeausgang ENO

des Programmbausteins ein individueller Wert (0 oder 1) zugewiesen werden. Dieses Verhalten wird in der Regel bei KOP- und FUP-Programmbausteinen eingesetzt. Bei SCL-Programmbausteinen können Sie auch so vorgehen. Sie müssen dabei nicht unbedingt die Anweisung "RET" verwenden um den Freigabeausgang ENO des SCL-Programmbausteins zu beeinflussen.

An den Netzwerkgrenzen wird der Freigabeausgang ENO immer wieder auf den Signalzustand "1" gesetzt. In einem KOP-Programmbaustein kann man das z. B. daran erkennen, dass die linke Stromschiene immer Spannung liefert, auch wenn der Freigabeausgang ENO der letzten Anweiung des vorherigen Netzwerks den Signalzustand "0" liefert.

Funktion der Anweisung "RET: Zurück springen" (KOP/FUP)

Bei VKE = 0 wird die Anweisung nicht ausgeführt und das nächste Netzwerk wird bearbeitet.

Bei VKE = 1 wird die Anweisung ausgeführt und es erfolgt ein Rücksprung zum aufrufenden Programmbaustein.

Der Signalzustand des Baustein-ENO kann mit vier Möglichkeiten bestimmt werden:

- RLO: Das VKE = 1 wird übernommen, d. h. das Baustein-ENO wird auf TRUE gesetzt.
- TRUE: Das Baustein-ENO wird auf TRUE gesetzt.
- FALSE: Das Baustein-ENO wird auf FALSE gesetzt.
- Operand: Der Signalzustand des angegebenen Operanden bestimmt über das Baustein-ENO.

Programmier- und Peripheriezugriffsfehlern

Programmier- und Peripheriezugriffsfehler können Sie nicht über den EN-/ENO-Mechanismus auswerten. Nutzen Sie hierzu entweder die globale Fehlerbehandlung über OBs oder die lokale Fehlerbehandlung über die Anweisungen "GET_ERROR" oder "GET_ERR_ID" (nur S7-1200/1500 CPUs). Falls für eine Anweisung keiner der Fehler aufgetreten ist, können Sie den zugehörigen Freigabeausgang ENO auswerten.

Programmbausteinaufrufe in allen Programmiersprachen (S7-300/400)

Für eine CPU der Baureihen S7-300/400 gilt: Wenn Sie Programmbausteine aufrufen, die keine Anweisungen enthalten, dann wird weder das BIE-Bit noch der Freigabeausgang ENO beeinflusst. Dadurch bleibt der Signalzustand des BIE-Bit konstant. Es kann keine Aussage über den Erfolg des Programmbausteinaufrufs erfolgen.

1.5.3.2 EN-/ENO-Mechanismus in KOP

Überblick über den EN-/ENO-Mechanismus in KOP

Um die Performance zu erhöhen, ist der EN-/ENO-Mechanismus bei den Anweisungen per Voreinstellung deaktiviert. Er kann jederzeit auch für einzelne Anweisungen wieder aktiviert werden. Weitere Informationen zur Aktivierung/Deaktivierung des EN-/ENO-Mechanismus finden Sie hier: EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

Bei KOP-/FUP-Programmbausteinen funkioniert der EN-/ENO-Mechanismus nur mit einem aktivierten Freigabeausgang ENO an den jeweiligen Anweisungen.

EN-/ENO-Mechanismus innerhalb einer Anweisung

Mit dem Freigabeeingang EN können Sie die Bearbeitung der Anweisung von Bedingungen abhängig machen. Die Anweisung wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie Laufzeitfehler in einer Anweisung abfragen und darauf reagieren:

- Der Freigabeausgang ENO liefert den Signalzustand "1", wenn während der Bearbeitung kein Fehler aufgetreten ist.
- Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:
  - Der Freigabeeingang EN liefert den Signalzustand "0".
  - Während der Bearbeitung ist ein Fehler aufgetreten.

EN-/ENO-Mechanismus bei Programmbausteinaufrufen

Alle Programmbausteinaufrufe werden mit einem Freigabeeingang EN und einem Freigabeausgang ENO versehen. Das gilt für alle aufrufenden Programmbausteine, unabhängig von der Programmiersprache, in der der aufgerufene Programmbaustein erstellt wurde. Das bedeutet, wenn Sie einen AWL- oder SCL-Programmbaustein aufrufen, in dem es den EN-/ENO-Mechanismus nicht standardmäßig und vorgefertigt gibt, hat der KOP- oder FUP-Programmbausteinaufruf trotzdem einen EN-/ENO-Mechanismus.

Den Freigabeeingang EN können Sie nutzen, um einen Programmbaustein, abhängig von Bedingungen, aufzurufen. Der Programmbaustein wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie den Fehlerstatus des Programmbausteins abfragen:

- Der Freigabeausgang ENO liefert den Signalzustand "1", wenn die Programmbausteinbearbeitung erfolgte.
- Wenn Sie den Freigabeausgang ENO im Fehlerfall innerhalb des aufgerufenen Programmbausteins nicht explizit auf den Signalzustand "0" setzen, behält er den Signalzustand "1". Mithilfe der Anweisung "RET: Zurück springen" können Sie das Baustein-ENO auf den Signalzustand "0" setzen.
  Weitere Informationen zur Verwendung der Anweisung RET finden Sie hier: Baustein-ENO eines KOP-/FUP-Programmbausteins beeinflussen (Seite 153)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren

In KOP und FUP haben bestimmte Anweisungen einen Freigabeausgang ENO (enable output) und nutzen somit den EN-/ENO-Mechanismus. Damit können Sie Laufzeitfehler in Anweisungen abfragen und darauf reagieren. Um die Performance der CPU zu steigern, ist der EN-/ENO-Mechanismus in der Voreinstellung deaktiviert. Dadurch haben Sie zunächst nicht mehr die Möglichkeit auf Laufzeitfehler der Anweisung über den ENO-Wert zu reagieren. Sie können den EN-/ENO-Mechanismus jedoch bei Bedarf jederzeit wieder aktivieren.

Sie können für jede Anweisung den EN-/ENO-Mechanismus einzeln aktivieren, um den ENO zu generieren. Wenn Sie für eine Anweisung den EN-/ENO-Mechanismus aktivieren, werden weitere Anweisungen, die Sie danach in Ihr Programm ziehen, ebenfalls mit aktiviertem EN-/ENO-Mechanismus eingefügt. Falls Sie die Auswertung des ENO für eine Anweisung nicht nutzen möchten, können Sie den EN-/ENO-Mechanismus jederzeit wieder deaktivieren. Weitere Anweisungen, die Sie danach in Ihr Programm ziehen, werden dann ohne EN-/ENO-Mechanismus eingefügt.

Bei aktiviertem Freigabeausgang ENO führen Laufzeitfehler nicht zum STOP der CPU.

EN-/ENO-Mechanismus aktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".
   Für die Anweisung wird der ENO-Wert wieder generiert. Auch weitere Anweisungen werden dann mit dem aktivierten Freigabeausgang eingefügt.

EN-/ENO-Mechanismus deaktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO nicht generieren".
   Für die Anweisung wird der ENO-Wert nicht mehr generiert. Auch weitere Anweisungen werden dann ohne den aktivierten Freigabeausgang eingefügt.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Baustein-ENO eines KOP-/FUP-Programmbausteins beeinflussen

Beschreibung

Mithilfe der Anweisung "RET: Zurück springen" springen Sie aus einem Programmbaustein heraus und beeinflussen damit den Signalzustand des Baustein-ENO.

Vorgehen

Um den Signalzustand des Baustein-ENO zu beeinflussen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie den EN-/ENO-Mechanismus einer Anweisung (z. B. der Addition "ADD").
2. Programmieren Sie anschließend am Freigabeausgang ENO über eine Negation die Anweisung "RET: Zurück springen" mit Signalzustand FALSE.

Ergebnis

Damit wird im Fehlerfall (z. B. einem Überlauf im Ergebnis) am Freigabeausgang ENO zunächst der Signalzustand "0" geliefert. Dieser Signalzustand "0" wird mithilfe der Negation zum Signalzustand "1" und bildet damit das VKE = 1 zur Ausführung der Anweisung "RET", die den Wert FALSE zurückliefert. Damit ist das Baustein-ENO des Programmbausteins FALSE und es erfolgt ein Rücksprung aus dem Programmbaustein zur nächsten Anweisung nach dem vorherigen Programmbausteinaufruf. Diese Vorgehensweise kann in jedem Netzwerk programmiert werden (z. B. bei mehreren Mathematischen Funktionen, etc.).

Es muss aber nicht zwingend ein Sprung aus dem Programmbaustein programmiert werden. Sie können auch innerhalb eines Netzwerks den Signalzustand "0" des Freigabeausgangs ENO einer einzelnen Anweisung dazu nutzen, dass die anschließende Anweisung nicht ausgeführt wird. Dabei wird das Baustein-ENO nicht beeinflusst.

Hinweis

Beeinflussung des Baustein-ENO

Nur durch das Herausspringen aus dem Programmbaustein können Sie den Signalzustand des Baustein-ENO beeinflussen.

Auch wenn der Freigabeausgang ENO der letzten Anweisung im letzten Netzwerk innerhalb eines Programmbausteins den Signalzustand "0" hat, wird das Baustein-ENO nicht beeinflusst.

Beispiel für den EN-/ENO-Mechanismus in KOP

Beispiel für eine Anweisung mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt die Anweisung "Addieren" mit EN-/ENO-Beschaltung und einer RET-Spule (Ret False):

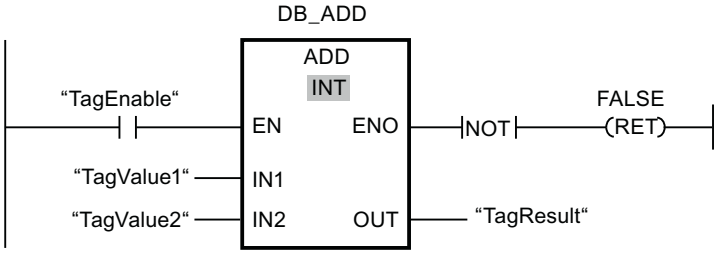

Nach dem Schließerkontakt "TagEnable" enthält der Freigabeeingang EN das Ergebnis der Vorverknüpfung:

- Liefert der Operand "TagEnable" den Signalzustand "0", dann wird die Anweisung "Addieren" nicht ausgeführt. Der Freigabeausgang ENO wird auf den Signalzustand "0" gesetzt und der aufgerufene Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins führt dann ebenfalls den Signalzustand "0".
- Liefert der Operand "TagEnable" den Signalzustand "1", dann führt der Freigabeeingang EN den Signalzustand "1" und die Anweisung "Addieren" wird ausgeführt. Die Anweisung addiert zwei Werte vom Datentyp INT. Liegt das erwartete Ergebnis außerhalb des Wertebereichs von INT (16 Bit: -32768 bis +32767), dann liefert die Anweisung trotzdem ein Ergebnis, aber dieses liegt innerhalb des Wertebereichs von INT. Das kommt dadurch zustande, dass das 16. Bit bei INT das Vorzeichenbit ist. Diesem Ergebnis ist der Überlauf nicht anzusehen. Deswegen gilt bei einer Addition => ENO := NOT(OV). Wenn es während der Bearbeitung des Programmbausteins zu einem Fehler kommt, dann wird der Freigabeausgang ENO auf den Signalzustand "0" gesetzt und der aufgerufene Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins führt dann ebenfalls den Signalzustand "0".
- Liefert der Operand "TagEnable" den Signalzustand "1", dann führt der Freigabeeingang EN den Signalzustand "1" und die Anweisung "Addieren" wird ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO ebenfalls den Signalzustand "1" und der Ausgang "TagResult" das Ergebnis.

Eine detaillierte Beschreibung zur Anweisung "RET: Zurück springen" finden Sie hier: --(RET): Zurück springen (Seite 902)

Beispiel für die Auswirkung des Freigabeausgangs ENO

Das folgende Beispiel zeigt, wie Sie Anweisungen mit aktiviertem und deaktiviertem Freigabeausgang ENO verwenden können:

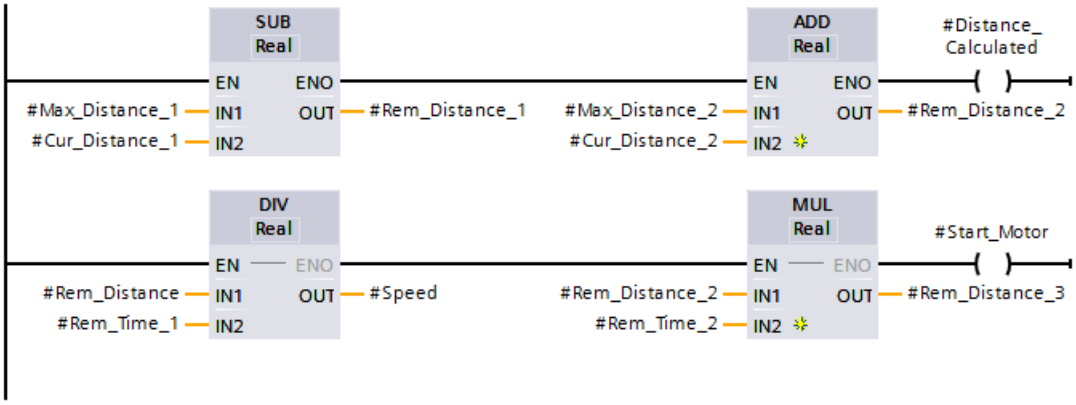

Wenn Sie den Freigabeausgang ENO, wie bei der Anweisung SUB, aktiviert haben, werden alle nachfolgenden Anweisungen ebenfalls mit aktiviertem Freigabeausgang ENO angelegt. Wenn es dann zu einem Arithmetikfehler während der Bearbeitung der Anweisung SUB kommt, wird die Anweisung ADD nicht ausgeführt.

Bei der Anweisung DIV im zweiten Zweig ist der Freigabeausgang ENO deaktiviert. Wenn es während der Bearbeitung zu einem Laufzeitfehler kommt, dann wird die Anweisung MUL trotzdem ausgeführt.

Beispiel für einen Programmbausteinaufruf mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt den Aufruf des Programmbausteins mit EN-/ENO-Beschaltung:

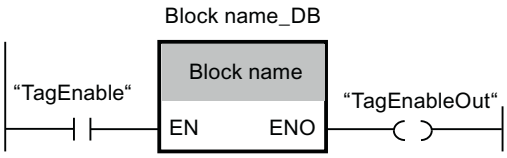

Der Programmbaustein wird nur ausgeführt, wenn der Operand "TagEnable" den Signalzustand "1" führt:

- Der Freigabeausgang ENO liefert einen Signalzustand, der davon abhängig ist, was innerhalb des Programmbausteins programmiert wurde.
- Liefert der Operand "TagEnable" den Signalzustand "0", dann wird der Programmbausteinaufruf nicht ausgeführt. Der Freigabeeingang EN und der Freigabeausgang ENO führen beide den Signalzustand "0".

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.3.3 EN-/ENO-Mechanismus in FUP

Überblick über den EN-/ENO-Mechanismus in FUP

Um die Performance zu erhöhen, ist der EN-/ENO-Mechanismus bei den Anweisungen per Voreinstellung deaktiviert. Er kann jederzeit auch für einzelne Anweisungen wieder aktiviert werden. Weitere Informationen zur Aktivierung/Deaktivierung des EN-/ENO-Mechanismus finden Sie hier: EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

Bei KOP-/FUP-Programmbausteinen funkioniert der EN-/ENO-Mechanismus nur mit einem aktivierten Freigabeausgang ENO an den jeweiligen Anweisungen.

EN-/ENO-Mechanismus innerhalb einer Anweisung

Mit dem Freigabeeingang EN können Sie die Bearbeitung der Anweisung von Bedingungen abhängig machen. Die Anweisung wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie Laufzeitfehler in einer Anweisung abfragen und darauf reagieren:

- Der Freigabeausgang ENO liefert den Signalzustand "1", wenn während der Bearbeitung kein Fehler aufgetreten ist.
- Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:
  - Der Freigabeeingang EN liefert den Signalzustand "0".
  - Während der Bearbeitung ist ein Fehler aufgetreten.

EN-/ENO-Mechanismus bei Programmbausteinaufrufen

Alle Programmbausteinaufrufe werden mit einem Freigabeeingang EN und einem Freigabeausgang ENO versehen. Das gilt für alle aufrufenden Programmbausteine, unabhängig von der Programmiersprache, in der der aufgerufene Programmbaustein erstellt wurde. Das bedeutet, wenn Sie einen AWL- oder SCL-Programmbaustein aufrufen, in dem es den EN-/ENO-Mechanismus nicht standardmäßig und vorgefertigt gibt, hat der KOP- oder FUP-Programmbausteinaufruf trotzdem einen EN-/ENO-Mechanismus.

Den Freigabeeingang EN können Sie nutzen, um einen Programmbaustein, abhängig von Bedingungen, aufzurufen. Der Programmbaustein wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie den Fehlerstatus des Programmbausteins abfragen:

- Der Freigabeausgang ENO liefert den Signalzustand "1", wenn die Programmbausteinbearbeitung erfolgte.
- Wenn Sie den Freigabeausgang ENO im Fehlerfall innerhalb des aufgerufenen Programmbausteins nicht explizit auf den Signalzustand "0" setzen, behält er den Signalzustand "1". Mithilfe der Anweisung "RET: Zurück springen" können Sie das Baustein-ENO auf den Signalzustand "0" setzen.
  Weitere Informationen zur Verwendung der Anweisung RET finden Sie hier: Baustein-ENO eines KOP-/FUP-Programmbausteins beeinflussen (Seite 158)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren

In KOP und FUP haben bestimmte Anweisungen einen Freigabeausgang ENO (enable output) und nutzen somit den EN-/ENO-Mechanismus. Damit können Sie Laufzeitfehler in Anweisungen abfragen und darauf reagieren. Um die Performance der CPU zu steigern, ist der EN-/ENO-Mechanismus in der Voreinstellung deaktiviert. Dadurch haben Sie zunächst nicht mehr die Möglichkeit auf Laufzeitfehler der Anweisung über den ENO-Wert zu reagieren. Sie können den EN-/ENO-Mechanismus jedoch bei Bedarf jederzeit wieder aktivieren.

Sie können für jede Anweisung den EN-/ENO-Mechanismus einzeln aktivieren, um den ENO zu generieren. Wenn Sie für eine Anweisung den EN-/ENO-Mechanismus aktivieren, werden weitere Anweisungen, die Sie danach in Ihr Programm ziehen, ebenfalls mit aktiviertem EN-/ENO-Mechanismus eingefügt. Falls Sie die Auswertung des ENO für eine Anweisung nicht nutzen möchten, können Sie den EN-/ENO-Mechanismus jederzeit wieder deaktivieren. Weitere Anweisungen, die Sie danach in Ihr Programm ziehen, werden dann ohne EN-/ENO-Mechanismus eingefügt.

Bei aktiviertem Freigabeausgang ENO führen Laufzeitfehler nicht zum STOP der CPU.

EN-/ENO-Mechanismus aktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".
   Für die Anweisung wird der ENO-Wert wieder generiert. Auch weitere Anweisungen werden dann mit dem aktivierten Freigabeausgang eingefügt.

EN-/ENO-Mechanismus deaktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO nicht generieren".
   Für die Anweisung wird der ENO-Wert nicht mehr generiert. Auch weitere Anweisungen werden dann ohne den aktivierten Freigabeausgang eingefügt.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Baustein-ENO eines KOP-/FUP-Programmbausteins beeinflussen

Beschreibung

Mithilfe der Anweisung "RET: Zurück springen" springen Sie aus einem Programmbaustein heraus und beeinflussen damit den Signalzustand des Baustein-ENO.

Vorgehen

Um den Signalzustand des Baustein-ENO zu beeinflussen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie den EN-/ENO-Mechanismus einer Anweisung (z. B. der Addition "ADD").
2. Programmieren Sie anschließend am Freigabeausgang ENO über eine Negation die Anweisung "RET: Zurück springen" mit Signalzustand FALSE.

Ergebnis

Damit wird im Fehlerfall (z. B. einem Überlauf im Ergebnis) am Freigabeausgang ENO zunächst der Signalzustand "0" geliefert. Dieser Signalzustand "0" wird mithilfe der Negation zum Signalzustand "1" und bildet damit das VKE = 1 zur Ausführung der Anweisung "RET", die den Wert FALSE zurückliefert. Damit ist das Baustein-ENO des Programmbausteins FALSE und es erfolgt ein Rücksprung aus dem Programmbaustein zur nächsten Anweisung nach dem vorherigen Programmbausteinaufruf. Diese Vorgehensweise kann in jedem Netzwerk programmiert werden (z. B. bei mehreren Mathematischen Funktionen, etc.).

Es muss aber nicht zwingend ein Sprung aus dem Programmbaustein programmiert werden. Sie können auch innerhalb eines Netzwerks den Signalzustand "0" des Freigabeausgangs

ENO einer einzelnen Anweisung dazu nutzen, dass die anschließende Anweisung nicht ausgeführt wird. Dabei wird das Baustein-ENO nicht beeinflusst.

Hinweis

Beeinflussung des Baustein-ENO

Nur durch das Herausspringen aus dem Programmbaustein können Sie den Signalzustand des Baustein-ENO beeinflussen.

Auch wenn der Freigabeausgang ENO der letzten Anweisung im letzten Netzwerk innerhalb eines Programmbausteins den Signalzustand "0" hat, wird das Baustein-ENO nicht beeinflusst.

Beispiel für den EN-/ENO-Mechanismus in FUP

Beispiel für eine Anweisung mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt die Anweisung "Addieren" mit EN-/ENO-Beschaltung und einer RET-Spule (Ret False):

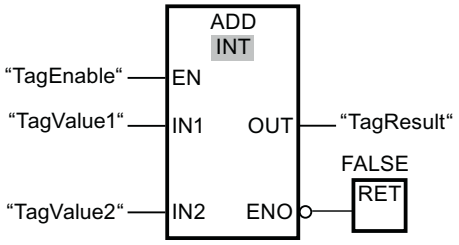

Nach dem Schließerkontakt "TagEnable" enthält der Freigabeeingang EN das Ergebnis der Vorverknüpfung:

- Liefert der Operand "TagEnable" den Signalzustand "0", dann wird die Anweisung "Addieren" nicht ausgeführt. Der Freigabeausgang ENO wird auf den Signalzustand "0" gesetzt und der aufgerufene Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins führt dann ebenfalls den Signalzustand "0".
- Liefert der Operand "TagEnable" den Signalzustand "1", dann führt der Freigabeeingang EN den Signalzustand "1" und die Anweisung "Addieren" wird ausgeführt. Die Anweisung addiert zwei Werte vom Datentyp INT. Liegt das erwartete Ergebnis außerhalb des Wertebereichs von INT (16 Bit: -32768 bis +32767), dann liefert die Anweisung trotzdem ein Ergebnis, aber dieses liegt innerhalb des Wertebereichs von INT. Das kommt dadurch zustande, dass das 16. Bit bei INT das Vorzeichenbit ist. Diesem Ergebnis ist der Überlauf nicht anzusehen. Deswegen gilt bei einer Addition => ENO := NOT(OV). Wenn es während der Bearbeitung des Programmbausteins zu einem Fehler kommt, dann wird der Freigabeausgang ENO auf den Signalzustand "0" gesetzt und der aufgerufene Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins führt dann ebenfalls den Signalzustand "0".
- Liefert der Operand "TagEnable" den Signalzustand "1", dann führt der Freigabeeingang EN den Signalzustand "1" und die Anweisung "Addieren" wird ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO ebenfalls den Signalzustand "1" und der Ausgang "TagResult" das Ergebnis.

Eine detaillierte Beschreibung zur Anweisung "RET: Zurück springen" finden Sie hier: RET: Zurück springen (Seite 1262)

Beispiel für die Auswirkung des Freigabeausgangs ENO

Das folgende Beispiel zeigt, wie Sie Anweisungen mit aktiviertem und deaktiviertem Freigabeausgang ENO verwenden können:

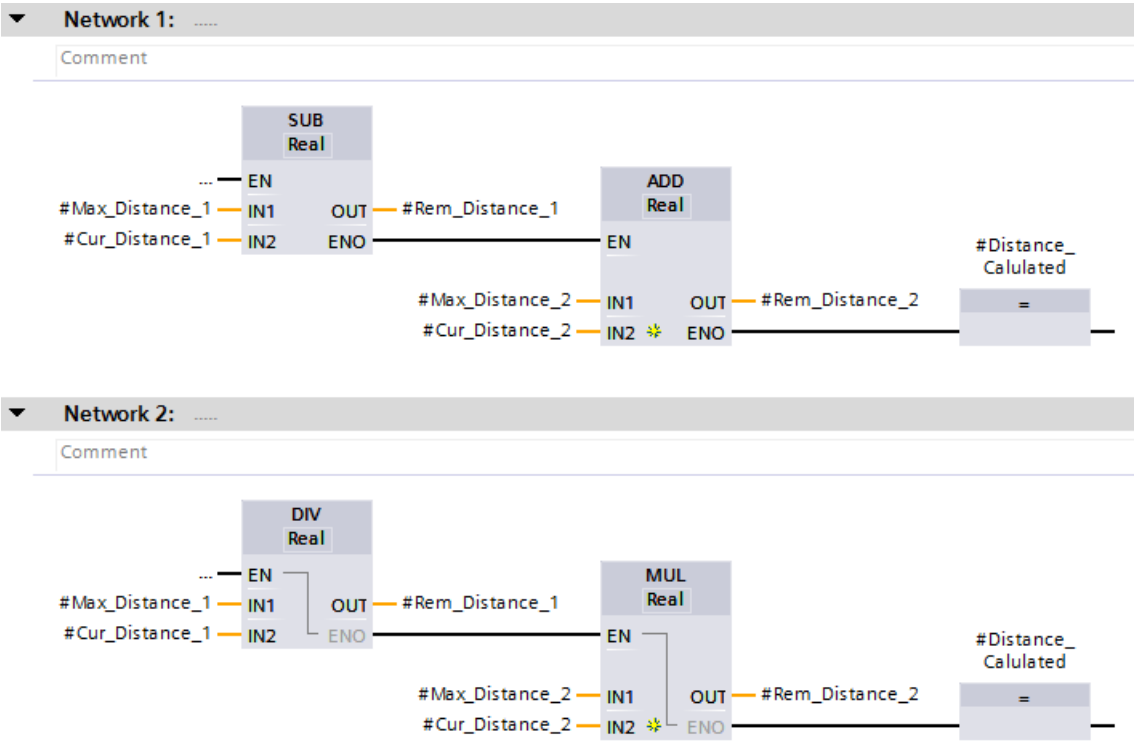

Wenn Sie den Freigabeausgang ENO, wie bei der Anweisung SUB, aktiviert haben, werden alle nachfolgenden Anweisungen ebenfalls mit aktiviertem Freigabeausgang ENO angelegt. Wenn es dann zu einem Arithmetikfehler während der Bearbeitung der Anweisung SUB kommt, wird die Anweisung ADD nicht ausgeführt.

Bei der Anweisung DIV im zweiten Netzwerk ist der Freigabeausgang ENO deaktiviert. Wenn es während der Bearbeitung zu einem Laufzeitfehler kommt, dann wird die Anweisung MUL trotzdem ausgeführt.

Beispiel für einen Programmbausteinaufruf mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt den Aufruf des Programmbausteins mit EN-/ENO-Beschaltung:

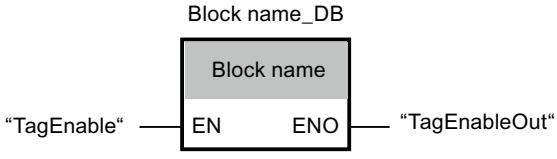

Der Programmbaustein wird nur ausgeführt, wenn der Operand "TagEnable" den Signalzustand "1" führt:

- Der Freigabeausgang ENO liefert einen Signalzustand, der davon abhängig ist, was innerhalb des Programmbausteins programmiert wurde.
- Liefert der Operand "TagEnable" den Signalzustand "0", dann wird der Programmbausteinaufruf nicht ausgeführt. Der Freigabeeingang EN und der Freigabeausgang ENO führen beide den Signalzustand "0".

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.3.4 EN-/ENO-Mechanismus in AWL

Überblick über den EN-/ENO-Mechanismus in AWL

EN-/ENO-Mechanismus innerhalb einer Anweisung

Der EN-/ENO-Mechanismus ist für einzelne Anweisungen nicht vorhanden. Er wird durch sprachspezifische Anweisungsfolgen und der Verwendung des Statuswortes (BIE-Bit) abgebildet.

Weitere Informationen zum Statuswort finden Sie hier: Statuswort bei S7-1500 (Seite 196)

Weitere Informationen finden Sie hier: Beispiel für den Nachbau des EN-/ENO-Mechanismus in AWL (Seite 162)

EN-/ENO-Mechanismus bei Programmbausteinaufrufen

Ein Programmbaustein, den Sie aus einem AWL-Programmbaustein heraus aufrufen, wird nicht mit den Parametern EN und ENO versehen. Unabhängig von der Programmiersprache, in der der Programmbaustein erstellt wurde, können sie jedoch eine Fehleraussage an den aufrufenden AWL-Programmbaustein aus dem BIE-Bit des Statusworts holen.

Sie können den Fehlerstatus des aufgerufenen Programmbausteins auswerten, indem Sie das BIE-Bit des Statusworts mit dem Verknüpfungsergebins (VKE) verknüpfen. Das VKE führt den Signalzustand "1", sobald die Programmbausteinbearbeitung des aufgerufenen Programmbausteins beginnt. Wenn Sie das VKE nach der Bearbeitung der Aktionen nicht explizit wieder auf "0" zurücksetzen, behält es den Signalzustand "1". Um eine Fehleraussage an den aufrufenden Programmbaustein zurückzumelden, müssen Sie das VKE explizit auf den Signalzustand "0" zurücksetzen. Die Fehleraussage wird mit den Anweisungen "SAVE: VKE im BIE-Bit speichern" und "SPBNB: Springen bei VKE = 0 und VKE speichern" gesetzt.

In AWL steht Ihnen zur Fehlerauswertung das BIE-Bit zur Verfügung. In den Programmiersprachen KOP, FUP und SCL wird das BIE-Bit als Grundlage für die Generierung des Freigabeausgangs ENO verwendet.

Der Wert "0" im BIE-Bit des Statusworts zeigt an, dass bei einem AWL-Programmbausteinaufruf (aufrufender Programmbaustein ist ein AWL Programmbaustein) ein Fehler aufgetreten ist.

Bearbeitung der Anweisung durch die CPU	BIE-Bit	Rückgabewert	Vorzeichen der Ganzzahl
fehlerhaft	0	kleiner als "0"	negativ (Vorzeichenbit ist "1")
fehlerfrei	1	größer als oder gleich "0"	positiv (Vorzeichenbit ist "0")

Weitere Informationen zum BIE-Bit finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel für den Nachbau des EN-/ENO-Mechanismus in AWL

Beispiel für eine Programmabfolge mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt einen Programmabschnitt zum Addieren von Werten mit EN-/ENO-Mechanismus:

AWL	Erläuterung
U "TagEnable"	// Signalzustand des Operanden "TagEnable" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
SPBNB MyLABEL	// Auswertung des Freigabeeingangs EN
	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und das aktuelle VKE im BIE-Bit speichern. Die folgenden Aktionen werden nicht ausgeführt.
	// Bei VKE = "1" die folgenden Aktionen bearbeiten.
L "Tag_Input_1"	// Ersten Wert der Addition laden.
L "Tag_Input_2"	// Zweiten Wert der Addition laden.
+I	// Werte addieren
T "Tag_Result"	// Summe in den Operanden "Tag_Result" übertragen.
UN OV	// Abfragen, ob ein Überlauf aufgetreten ist.
SAVE	// Signalzustand des VKE in das BIE-Bit übertragen.
CLR	/// VKE auf "0" zurücksetzen und Verknüpfungskette beenden.
MyLABEL: U BIE	// Sprungmarke "MyLABEL"
	// BIE-Bit abfragen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden "Tag_Output" zuweisen.

Die Abfrage des Operanden "TagEnable" ergibt das Ergebnis der Vorverknüpfung (VKE). Die Anweisung "SPBNP: Springen bei VKE = 0 und VKE speichern" speichert das VKE im BIE-

Bit. Zusätzlich wertet die Anweisung das VKE aus und führt, abhängig vom Signalzustand, die folgenden Aktionen aus:

- Ist das VKE "0", wird die Programmbearbeitung bei der Sprungmarke "MyLABEL" mit der Abfrage des BIE-Bit fortgesetzt. Die Addition wird nicht ausgeführt. Das aktuelle VKE wird dem Operanden "Tag_Output" zugewiesen.
- Ist das VKE "1", wird die Addition ausgeführt. Mit der Abfrage des Überlaufbits (OV) wird ermittelt, ob bei der Addition ein Fehler auftrat. Das Abfrageergebnis wird im BIE-Bit gespeichert. Die Anweisung "CLR: VKE auf 0 rücksetzen" setzt das VKE auf "0" zurück und beendet die Verknüpfungskette. Anschließend wird das BIE-Bit abgefragt und dem Operanden "Tag_Output" zugewiesen. Der Signalzustand des BIE-Bit, sowie des Operanden "Tag_Output" zeigt, ob die Addition fehlerfrei durchgeführt wurde.

Beispiel für einen Programmbausteinaufruf mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt den Aufruf des Programmbausteins mit EN-/ENO-Beschaltung:

AWL	Erläuterung
U "TagEnable"	// Signalzustand des Operanden "TagEnable" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
SPBNB MyLABEL	// Auswertung des Freigabeeingangs EN
	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und das aktuelle VKE im BIE-Bit speichern. Die folgenden Aktionen werden nicht ausgeführt.
	// Bei VKE = "1" die folgenden Aktionen bearbeiten.
CALL "Block name", "Block name_DB"	// Der Programmbaustein wird aufgerufen.
MyLABEL: U BIE	// Sprungmarke "MyLABEL"
	// BIE-Bit abfragen und mit dem VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden "Tag_Output" zuweisen.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.3.5 EN-/ENO-Mechanismus in SCL

Überblick über den EN-/ENO-Mechanismus in SCL

Um die Performance zu erhöhen, ist der EN-/ENO-Mechanismus per Voreinstellung deaktiviert. Er kann aber jederzeit mithilfe des Bausteinattributs "ENO automatisch setzen" aktiviert werden. Weitere Informationen zur Aktivierung/Deaktivierung des EN-/ENO-Mechanismus finden Sie hier: EN-/ENO-Mechanismus in SCL aktivieren und deaktivieren (Seite 164)

Neben der Aktivierung des Bausteinattributs "ENO automatisch setzen" können Sie den Freigabeausgang ENO auch mithilfe einer IF-Anweisung beeinflussen. Weitere Informationen finden Sie hier: Freigabeausgang ENO mithilfe einer IF-Anweisung beeinflussen (Seite 166)

EN-/ENO-Mechanismus innerhalb einer Anweisung

Der EN-/ENO-Mechanismus ist für Anweisungen optional.

Weitere Informationen finden Sie hier: Freigabeausgang ENO bei SCL-Anweisungen verwenden (Seite 165)

EN-/ENO-Mechanismus bei Programmbausteinaufrufen

Der Aufruf eines SCL-Programmbausteins wird immer bedingungslos ausgeführt, unabhängig davon, ob der EN-/ENO-Mechanismus aktiviert ist oder nicht. Sie können aber den Aufruf mit einer IF-Anweisung verknüpfen und damit den Aufruf von verschiedenen Bedingungen abhängig machen. Eine mögliche Bedingung kann z. B. das Baustein-ENO des SCL-Programmbausteins sein. Dafür muss dann aber der EN-/ENO-Mechanismus aktiviert sein.

Wenn das Baustein-ENO aufgrund eines Laufzeitfehlers in einer vorangegangenen Anweisung den Signalzustand "0" liefert, dann wird der Bausteinaufruf nicht ausgeführt.

Weitere Informationen finden Sie hier: Freigabeausgang ENO eines SCL-Programmbausteins beeinflussen (Seite 166)

EN-/ENO-Mechanismus in SCL aktivieren und deaktivieren

Beschreibung

Der EN-/ENO-Mechanismus prüft zur Laufzeit, ob Fehler bei der Bearbeitung bestimmter Anweisungen auftreten. Wenn ein Laufzeitfehler auftritt, wird das Baustein-ENO auf "0" gesetzt.

EN-/ENO-Mechanismus in den Bausteineigenschaften aktivieren

Um den EN-/ENO-Mechanismus in den Bausteineigenschaften zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den SCL-Baustein, dessen Eigenschaften Sie anzeigen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf die Gruppe "Attribute".
5. Aktivieren Sie die Eigenschaft "ENO automatisch setzen".
6. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis: Der EN-/ENO-Mechanismus ist nur für den angewählten SCL-Programmbaustein aktiviert.

EN-/ENO-Mechanismus in den Programmeigenschaften aktivieren

Um den EN-/ENO-Mechanismus in den Programmeigenschaften zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wählen Sie die Gruppe "SCL (Structured Control Language)" aus.
4. Aktivieren Sie die Eigenschaft "ENO automatisch setzen".

Ergebnis: Der EN-/ENO-Mechanismus ist für alle neuen SCL-Programmbausteine als Voreinstellung aktiviert.

Freigabeausgang ENO bei SCL-Anweisungen verwenden

Beschreibung

Bei jeder SCL-Anweisung, z. B. einer mathematischen Funktion, müssen Sie den Freigabeausgang ENO abfragen, um mit ihm arbeiten zu können.

Vorgehen

Um den Freigabeausgang ENO abzufragen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie den EN-/ENO-Mechanismus.
2. Fragen Sie nach der SCL-Anweisung den Freigabeausgang ENO ab (z. B. #MyOutputBool := ENO;)
   Wenn der Fehlerfall eintritt, z. B. ein Überlauf im Ergebnis, liefert der Freigabeausgang ENO den Signalzustand "0". Mit diesem Signalzustand können Sie jetzt weiterarbeiten und z. B. die Anweisung "RETURN", eine Fehleranzeige oder einen Ersatzwert programmieren.
3. Setzen Sie den Signalzustand des Freigabeausgangs ENO vor der Bearbeitung der nächsten Anweisung wieder auf "1" (z. B. ENO := 1;).
   Damit kann der Signalzustand beim nächsten Fehlerfall wieder auf "0" gesetzt werden. Wenn Sie das nicht tun, bleibt der Signalzustand des Freigabeausgangs ENO "0".

Hinweis

Sprung in einen SCL-Baustein

Bei einem Sprung in einen SCL-Baustein wird der Signalzustand des Freigabeausgangs ENO automatisch auf "1" gesetzt.

Freigabeausgang ENO eines SCL-Programmbausteins beeinflussen

Beschreibung

Das letzte Anweisungs-ENO innerhalb eines SCL-Programmbausteins beeinflusst immer das Baustein-ENO.

Darüber hinaus können Sie auch mithilfe der Anweisung "RETURN" aus dem Programmbaustein herausspringen und dabei dem Baustein-ENO den Wert "1" oder "0" zuweisen.

Vorgehen

Um den Signalzustand des Baustein-ENO zu beeinflussen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie den EN-/ENO-Mechanismus.
2. Programmieren Sie anschließend:
```
IF ENO := 0 THEN
RETURN;
END_IF;
```

Ergebnis

Damit wird im Fehlerfall (z. B. einem Überlauf im Ergebnis) am Freigabeausgang ENO, z. B. einer Additionsanweisung, zunächst der Signalzustand "0" geliefert. Damit ist das Baustein-ENO FALSE und es erfolgt ein Sprung aus dem Programmbaustein zur nächsten Anweisung nach dem vorherigen Programmbausteinaufruf. Diese Vorgehensweise kann mehrfach in einem SCL-Programmbaustein programmiert werden (z. B. bei mehreren Mathematischen Funktionen, etc.).

Es muss aber nicht zwingend ein Sprung aus dem Programmbaustein programmiert werden. Sie können auch innerhalb des SCL-Programmbausteins den Signalzustand "0" des Freigabeausgangs ENO einer einzelnen Anweisung dazu nutzen, dass die anschließende Anweisung nicht ausgeführt wird. Dazu wird das Anweisungs-ENO in einer nachfolgenden IF-Anweisung mit der folgenden Anweisung verknüpft.

Hinweis

Beeinflussung des Baustein-ENO

Bei einem SCL-Programmbaustein wird der Signalzustand des letzten Anweisungs-ENO am Ende des Bausteins an das Baustein-ENO weitergegeben.

Damit beeinflusst das Anweisungs-ENO direkt das Baustein-ENO.

Freigabeausgang ENO mithilfe einer IF-Anweisung beeinflussen

Beschreibung

Wenn Sie den Freigabeausgang ENO mithilfe einer IF-Anweisung beeinflussen, dann müssen Sie das Bausteinattribut "ENO automatisch setzen" nicht aktivieren.

Vorgehen

Um den Freigabeausgang ENO mithilfe einer IF-Anweisung zu beeinflussen, gehen Sie folgendermaßen vor:

1. Programmieren Sie eine IF-Anweisung.
2. Verwenden Sie den Freigabeausgang ENO entweder innerhalb der IF-Anweisung oder bei ELSE.

Ergebnis

Abhängig von einem booleschen Wert oder einem Wertevergleich wird der Freigabeausgang ENO innerhalb der IF-Anweisung auf den Signalzustand "1" (ENO := 1;) oder bei ELSE auf den Signalzustand "0" (ENO := 0;) gesetzt.

Beispiel für den EN-/ENO-Mechanismus in SCL

Beispiel für eine Anweisung mit EN-/ENO-Mechanismus

Um den EN-/ENO-Mechanismus für Anweisungen nutzen zu können, müssen Sie die Bausteineigenschaft "ENO automatisch setzen" aktivieren. Das folgende Beispiel zeigt die Nutzung des Freigabeausgangs ENO für die Operation "a + b":

```
"MyOutputREAL" := #a + #b;
"MyOutputBOOL" := ENO;
```

Wenn die Operation "a + b" fehlerfrei ausgeführt wird, führt der Operand "MyOutputBool" den Signalzustand "1". im Fehlerfall liefert der Freigabeausgang ENO den Signalzustand "0".

Beispiel für einen Programmbausteinaufruf mit EN-/ENO-Mechanismus

Das folgende Beispiel zeigt den Aufruf des Programmbausteins. Dazu muss der EN-/ENO-Mechanismus aktiviert und eine Anweisung davor programmiert sein, die das Baustein-ENO beeinflusst:

```
IF ENO := 1 THEN
"Block name_DB"();
END_IF;
```

Der Programmbaustein wird nur aufgerufen und ausgeführt, wenn der Freigabeausgang ENO den Signalzustand "1" führt. Der aufgerufene Programmbaustein liefert auch ein Baustein-ENO zurück, wenn dessen EN-/ENO-Mechanismus aktiviert ist. Dieses Baustein-ENO wird dann im aufrufenden Programmbaustein weiterverwendet.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.3.6 EN-/ENO-Mechanismus in GRAPH

Überblick über den EN-/ENO-Mechanismus in GRAPH

EN-/ENO-Mechanismus innerhalb einer Anweisung

Sie haben keinen Zugriff auf den Freigabeausgang ENO der Anweisungen, d. h. Sie können den Zustand des Freigabeausgangs ENO am GRAPH-Funkionsbaustein nicht beeinflussen. Sie können sich aber z. B. bei Konvertierungen, mathematischen Funktionen oder KOP/FUP-Anweisungen den Freigabeausgang ENO im Programmstatus anzeigen lassen.

EN-/ENO-Mechanismus bei Programmbausteinaufrufen

Den Freigabeeingang EN können Sie nutzen, um einen Programmbaustein, abhängig von Bedingungen, aufzurufen. Der Programmbaustein wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie den Fehlerstatus des Programmbausteins abfragen:

- Der Freigabeausgang ENO liefert den Signalzustand "1", sobald der aufgerufene Programmbaustein fehlerfrei bearbeitet wurde.
- Der Freigabeausgang ENO liefert den Signalzustand "0", wenn während der Bearbeitung des aufgerufenen Programmbausteins ein Fehler auftrat.
- Der Freigabeausgang ENO kann nicht explizit gesetzt oder zurückgesetzt werden.
- Der Freigabeausgang ENO wird nicht durch das ENO der Anweisungen beeinflusst.

EN-/ENO-Mechanismus in GRAPH aktivieren und deaktivieren

Beschreibung

Beim Testen mit Programmstatus wird der Status des Freigabeausgangs ENO angezeigt. Dieser hat den Wert TRUE, wenn die Aktion erfolgreich war, oder FALSE bei einer fehlgeschlagenen Aktion.

Die Anzeigemöglichkeit des Status des Freigabeausgangs ENO steht Ihnen an folgenden Stellen im GRAPH-Programmbaustein zur Verfügung:

- Vorausgeschaltete permanente Anweisungen
- Kettenansicht > Aktionen
- Nachgeschaltete permanente Anweisungen

EN-/ENO-Mechanismus in den Bausteineigenschaften aktivieren

Um den EN-/ENO-Mechanismus in den Bausteineigenschaften zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den GRAPH-Baustein, dessen Eigenschaften Sie anzeigen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf die Gruppe "Attribute".
5. Aktivieren Sie die Eigenschaft "ENO automatisch setzen".
6. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis: Der EN-/ENO-Mechanismus ist nur für den angewählten GRAPH-Baustein aktiviert.

Beispiel für den EN-/ENO-Mechanismus in GRAPH

Beispiel für einen Programmstatus mit Freigabeausgang ENO

Das folgende Beispiel zeigt den Programmstatus des Freigabeausgangs ENO in der Kettenansicht unter Aktionen:

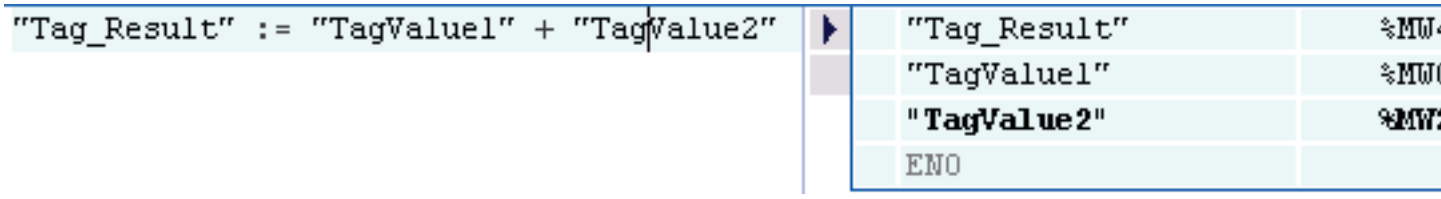

Wenn es während der Bearbeitung zu einem Fehler kommt, dann liefert der Freigabeausgang ENO den Signalzustand FALSE.

Beispiel für einen Programmbausteinaufruf

Das folgende Beispiel zeigt den Aufruf des GRAPH-Programmbausteins in einem KOP-Programmbaustein mit EN-/ENO-Beschaltung :

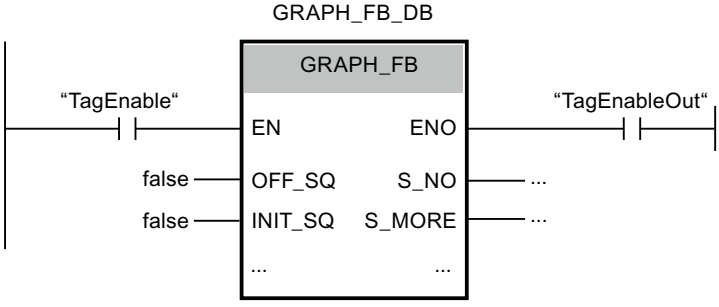

Der Programmbaustein wird nur ausgeführt, wenn der Operand "TagEnable" den Signalzustand "1" führt:

- Liefert der Operand "TagEnable" den Signalzustand "0", wird der aufgerufene Programmbaustein nicht bearbeitet. Sowohl der Freigabeeingang EN als auch der Freigabeausgang ENO liefern den Signalzustand "0".
- Liefert der Operand "TagEnable" den Signalzustand "1", dann führt der Freigabeeingang EN den Signalzustand "1" und der aufgerufene Programmbaustein wird ausgeführt. Der Freigabeausgang ENO liefert einen Signalzustand, der davon abhängig ist, was innerhalb des Programmbausteins programmiert wurde.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.3.7 EN-/ENO-Mechanismus in Bausteinen mit unterschiedlichen Netzwerksprachen

Beschreibung

Sie können auch in Bausteinen mit unterschiedlichen Netzwerksprachen den EN-/ENO-Mechanismus einsetzen. Jede Programmiersprache bildet den Fehlerstatus unterschiedlich ab:

- SCL besitzt eine Variable ENO, die den Fehlerstatus speichert und die Sie abfragen können. Der direkte Zugriff auf diese Variable ist nur mit SCL möglich.
- KOP/FUP/AWL haben keine spezielle Variable für ENO. Sie können den Fehlerstatus aber für AWL aus dem BIE-Bit auslesen und für KOP/FUP über die RET-Spule abfragen.

Für das Auslesen des Fehlerstatus für den gesamten Baustein gelten die folgenden Regeln:

- Das letzte Netzwerk im Baustein ist ein KOP-/FUP-Netzwerk:
  Falls Sie keine RET-Spule verwenden, ist der Fehlerstatus standardmäßig TRUE.
- Das letzte Netzwerk im Baustein ist ein AWL-Netzwerk:
  Das BIE-Bit bestimmt den Fehlerstatus. Das BIE-Bit kann in AWL-Netzwerken durch das BIE-Register bearbeitet werden.
- Das letzte Netzwerk im Baustein ist ein SCL-Netzwerk:
  Die Variable ENO bestimmt den Fehlerstatus des Bausteins.

Beeinflussung des Baustein-ENO

Bei KOP-/FUP-Programmbausteinen mit mehreren Netzwerken, die aus unterschiedlichen Programmiersprachen (z. B. KOP, FUP, AWL oder SCL) bestehen können, entscheidet das zuletzt bearbeitete Netzwerk über den Signalzustand des Baustein-ENO.

Ist das zuletzt bearbeitete Netzwerk eines solchen Programmbausteins ein KOP- oder FUP-Netzwerk, dann wird der Signalzustand des Baustein-ENO nicht zwingend beeinflusst. Es wird nur dann beeinflusst, wenn bei einer Anweisung, bei der der EN-/ENO-Mechanismus aktiviert ist, der Freigabeausgang ENO den Signalzustand "0" liefert und mithilfe der Anweisung "RET: Zurück springen" aus dem Programmbaustein gesprungen wird. Wenn diese Anweisung nicht programmiert wurde, dann liefert das Baustein-ENO immer den Signalzustand "1". Der

möglicherweise vorhandene Signalzustand "0" eines Anweisungs-ENO hat keinen Einfluss auf das Baustein-ENO, da es am Anfang eines Netzwerks immer auf den Signalzustand "1" gesetzt wird.

Das gilt auch für ein KOP- oder FUP-Netzwerk, das vor einem anderen Netzwerk (KOP, FUP oder SCL) steht. Der Signalzustand des Baustein-ENO wird nur mithilfe der Anweisung "RET: Zurück springen" beeinflusst. Ansonsten wird mit dem Übergang zum nächsten Netzwerk der Signalzustand des Baustein-ENO wieder auf "1" gesetzt.

Wenn das letzte Netzwerk eines gemischten Programmbausteins ein SCL-Netzwerk ist, dann wird das Baustein-ENO immer davon beeinflusst.

Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

1.5.4 Fehlerauswertung mit dem Ausgangsparameter RET_VAL

Grundlagen zur Fehlerauswertung von Bibliotheksbausteinen (SFB und SFC)

Neben dem Ausgangsparameter RET_VAL stehen Ihnen noch zwei weitere Fehlerauswertungsmöglichkeiten zur Verfügung:

- Mit dem EN-/ENO-Mechanismus (KOP, FUP und SCL)
  Weitere Informationen zum EN-/ENO-Mechanismus finden Sie hier: Grundlagen zum EN-/ENO-Mechanismus (Seite 150)
- Mit dem BIE-Bit (Binärergebnisbit) des Statusworts (AWL)
- Mit dem Ausgangsparameter RET_VAL (return value)

Der Freigabeausgang ENO gibt nur die Information aus, dass ein Fehler aufgetreten ist. Wenn Sie wissen möchten, welcher Fehler genau aufgetreten ist, hilft Ihnen der Ausgangsparameter RET_VAL weiter. Mit dessen Hilfe können Sie feststellen, ob die CPU die Anweisung erfolgreich ausführen konnte oder nicht. Im Fehlerfall erfahren Sie auch, warum die Anweisung nicht erfolgreich ausgeführt werden konnte.

Im nachfolgend beschriebenen Kapitel erhalten Sie eine detaillierte Information zu den beiden Fehlerauswertungsmöglichkeiten RET_VAL und BIE-Bit.

Empfehlung zur Reihenfolge der Fehlerauswertungen

Vor der Auswertung der anweisungsspezifischen Ausgangsparameter (z. B. OUT) sollten Sie wie folgt vorgehen:

1. Werten Sie als Erstes den Freigabeausgang ENO, bzw. das BIE-Bit des Statusworts in AWL, aus.
2. Überprüfen Sie anschließend den Ausgangsparameter RET_VAL.

Falls durch den Freigabeausgang ENO oder das BIE-Bit eine fehlerhafte Bearbeitung der Anweisung signalisiert wird oder im Ausgangsparameter RET_VAL ein allgemeiner

Fehlercode steht, liefern die anweisungsspezifischen Ausgangsparameter einen ungültigen Wert.

Falls beim Ausgangsparameter RET_VAL ein allgemeiner Fehler auftritt, so wird dies nur durch den Wert "0" im BIE-Bit des Statusworts angezeigt. Der Rückgabewert ist vom Datentyp Ganzzahl (INT). Die Relation des Rückgabewerts zu dem Wert "0" zeigt an, ob während der Bearbeitung der Anweisung ein Fehler aufgetreten ist.

Allgemeine und spezifische Fehlercodes (RET_VAL)

Bei den Fehlercodes des Ausgangsparameters RET_VAL wird unterschieden zwischen:

- einem allgemeinen Fehlercode, den alle Anweisungen ausgeben können
- einem spezifischen Fehlercode, den eine Anweisung, abhängig von ihren spezifischen Funktionen, ausgeben kann.

Es handelt sich bei dem Datentyp des Ausgangsparameters RET_VAL um eine Ganzzahl (INT). Die Fehlercodes der Anweisung werden nach hexadezimalen Werten gegliedert. Wenn Sie einen Rückgabewert auswerten und den Wert mit den Fehlercodes vergleichen, die in dieser Dokumentation aufgeführt sind, dann lassen Sie sich den Fehlercode im Hexadezimalformat anzeigen.

Sie können Ihr Programm so schreiben, dass es auf mögliche Fehler in der Bearbeitung der Anweisung reagiert. So können Sie Folgefehler vermeiden.

Hinweis

Fehler bei der Versorgung der Eingangsparameter

Wenn es bei der Bearbeitung der Anweisung, die den Parameter RET_VAL beinhaltet, zu Fehlern bei der Versorgung der Eingangsparameter kommt, dann wird am Parameter RET_VAL ein ungültiger Fehlercode ausgegeben und die Ausgangsparameter der Anweisung dürfen nicht ausgewertet werden.

Das folgende Bild erläutert den Aufbau eines Fehlercodes einer Systemfunktion im Hexadezimalformat:

Fehlercode, z.B. W#16#8081

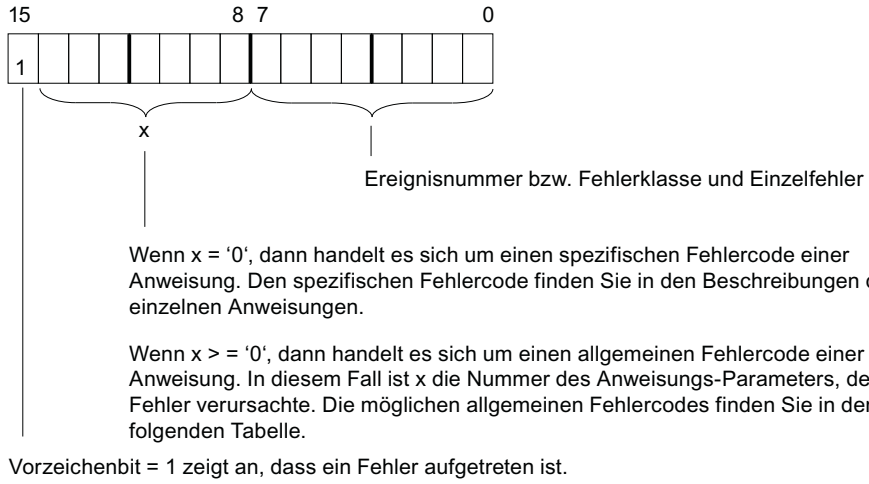

Da der Ausgangsparameter RET_VAL vom Typ INT ist, können Sie leicht unterscheiden, ob ein Fehler aufgetreten ist:

- Wenn der Wert <0 ist, dann handelt es sich um einen Fehler.
- Wenn der Wert =0 ist, dann ist kein Fehler aufgetreten.
- Wenn der Wert >0 ist, dann ist zwar kein Fehler aufgetreten, aber die Anweisung wurde noch nicht erfolgreich ausgeführt. Dies wird z. B. bei asynchronen Anweisungen verwendet um anzuzeigen, dass die Bearbeitung der Anweisung begonnen wurde, aber noch nicht abgeschlossen ist. Siehe dazu die Rückgabewerte von T_SEND oder WRIT_DBL.

Allgemeine Fehlercodes

Der allgemeine Fehlercode zeigt Fehler an, die bei allen Anweisungen auftreten können. Ein allgemeiner Fehlercode besteht aus den beiden folgenden Nummern:

- Einer Parameternummer zwischen 1 und 111, wobei 1 den ersten Parameter, 2 den zweiten Parameter usw. der aufgerufenen Anweisung anzeigt.
- Eine Ereignisnummer zwischen 0 und 127. Die Ereignisnummer zeigt einen Fehler an.

Das folgende Bild zeigt den Aufbau eines allgemeinen Fehlercodes:

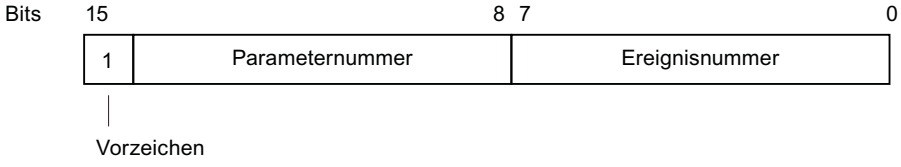

Hinweis

Reaktion auf einen allgemeinen Fehlercode

Wenn im Ausgangsparameter RET_VAL ein allgemeiner Fehlercode eingetragen wurde, könnte folgendes passiert sein:

- Die zur Anweisung gehörige Aktion wurde angestoßen oder bereits vollständig durchgeführt.
- Bei der Aktion ist ein anweisungsspezifischer Fehler aufgetreten. Aufgrund eines hinterher zusätzlich aufgetretenen allgemeinen Fehlers konnte der spezifische Fehler jedoch nicht mehr angezeigt werden.

Die folgende Tabelle erläutert die allgemeinen Fehlercodes eines Rückgabewerts. Der Fehlercode wird im Hexadezimalformat gezeigt. Der Buchstabe x in jeder Codenummer dient nur als Platzhalter und stellt die Nummer des Parameters der Systemfunktion dar, die den Fehler verursacht hat:

Fehlercode (W#16#...)	Erläuterung
8x01	Unzulässige Syntaxkennung bei einem VARIANT-Parameter
8x22	Bereichslängenfehler beim Lesen eines Parameters.
8x23	Bereichslängenfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem VARIANT-Parameter nicht durch 8 teilbar ist.
8x24	Bereichsfehler beim Lesen eines Parameters.
8x25	Bereichsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x in einem Bereich befindet, der für die Systemfunktion unzulässig ist. Die Beschreibung der jeweiligen Funktion gibt die Bereiche an, die für die Funktion unzulässig sind.
8x26	Der Parameter enthält eine zu große Nummer einer Zeitzelle. Dieser Fehlercode zeigt an, dass die Zeitzelle, die in Parameter x angegeben wird, nicht vorhanden ist.
8x27	Der Parameter enthält eine zu große Nummer einer Zählerzelle (Nummernfehler des Zählers). Dieser Fehlercode zeigt an, dass die Zählerzelle, die in Parameter x angegeben wird, nicht vorhanden ist.
8x28	Ausrichtungsfehler beim Lesen eines Parameters.
8x29	Ausrichtungsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass der Verweis auf den Parameter x ein Operand ist, dessen Bitadresse ungleich 0 ist.
8x30	Der Parameter befindet sich in dem schreibgeschützten Global-DB.
8x31	Der Parameter befindet sich in dem schreibgeschützten Instanz-DB. Dieser Fehlercode zeigt an, dass der Parameter x sich in einem schreibgeschützten Datenbaustein befindet. Wenn der Datenbaustein von der Systemfunktion selbst geöffnet wurde, gibt die Systemfunktion immer den Wert W#16#8x30 aus.
8x32	Der Parameter enthält eine zu große DB-Nummer (Nummernfehler des DB).
8x34	Der Parameter enthält eine zu große FC-Nummer (Nummernfehler der FC).

Fehlercode (W#16#...)	Erläuterung
8x35	Der Parameter enthält eine zu große FB-Nummer (Nummernfehler des FB). Dieser Fehlercode zeigt an, dass der Parameter x eine Bausteinnummer enthält, die größer ist als die maximal zulässige Bausteinnummer.
8x3A	Der Parameter enthält die Nummer eines DB, der nicht geladen ist.
8x3C	Der Parameter enthält die Nummer einer FC, die nicht geladen ist.
8x3E	Der Parameter enthält die Nummer eines FB, der nicht geladen ist.
8x42	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.
8x43	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.
8x44	Fehler beim n-ten (n > 1) Lesezugriff nach Auftreten eines Fehlers.
8x45	Fehler beim n-ten (n > 1) Schreibzugriff nach Auftreten eines Fehlers. Dieser Fehlercode zeigt an, dass der Zugriff auf den gewünschten Parameter verweigert wird.
8x7F	Interner Fehler Dieser Fehlercode zeigt einen internen Fehler am Parameter x an.

Spezifische Fehlercodes

Einige Anweisungen besitzen einen Rückgabewert, der einen für die Anweisung spezifischen Fehlercode zur Verfügung stellt. Ein spezifischer Fehlercode zeigt Fehler an, die nur bei einzelnen Anweisungen auftreten können.

Ein spezifischer Fehlercode besteht aus den beiden folgenden Nummern:

- Eine Fehlerklasse zwischen 0 und 7.
- Ein Einzelfehler zwischen 0 und 15.

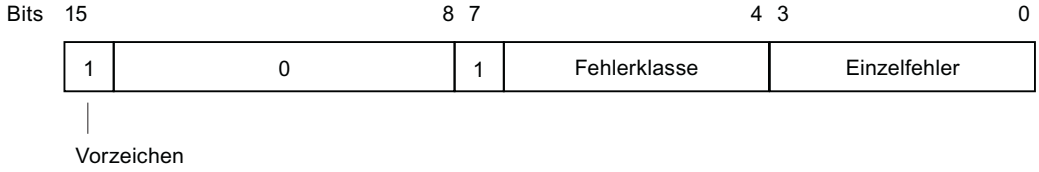

Informationen zu den spezifischen Fehlercodes finden Sie in den Beschreibungen der einzelnen Anweisungen im Informationssystem.

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

1.5.5 Verwendung der Anweisungen GET_ERROR und GET_ERR_ID

Einführung

Die lokale Fehlerbehandlung bietet die Möglichkeit, das Auftreten von Fehlern innerhalb eines Programmbausteins abzufragen und die dazugehörigen Fehlerinformationen auszuwerten. Die lokale Fehlerbehandlung können Sie für Organisationsbausteine (OB), Funktionsbausteine (FB) und Funktionen (FC) einstellen. Wenn die lokale Fehlerbehandlung aktiviert ist, wird die Systemreaktion ignoriert.

Aus der Fehlerinformation der Anweisung GET_ERR_ID können Sie die Fehlernummer auslesen. Aus der Fehlerinformation der Anweisung GET_ERROR können Sie z. B. bei einem Zugriffsfehler auslesen, welcher Parameter den Zugriffsfehler verursacht hat. Damit die Anweisungen die benötigten Fehlerinformationen ausgeben können, müssen diese im Anwenderprogramm für jeden einzelnen Programmbaustein, aus dem mögliche Fehler ausgewertet werden sollen, programmiert werden. Wenn Sie mit den Anweisungen arbeiten, wird kein Fehler-OB aufgerufen und es entsteht kein Eintrag im Diagnosepuffer. Mit dieser Art der Fehlerbehandlung greifen Sie beim Auftreten eines Fehlers aktiv in den Programmablauf ein, indem Sie eine Reaktion auf den aufgetretenen Fehler programmieren. Da die Fehler an jeder beliebigen Stelle innerhalb des Programmbausteins auftreten können, empfiehlt es sich, die Anweisung am Ende des Programmbausteins zu integrieren.

Die Anweisungen GET_ERROR und GET_ERR_ID unterscheiden sich in der Tiefe der ausgegebenen Fehlerinformationen.

Sobald Sie eine der beiden Anweisungen in Ihren Programmcode integriert haben, wird im Inspektorfenster unter "Eigenschaften > Attribute" das Optionskästchen "Lokale Fehlerbehandlung im Baustein" aktiviert. Die Einstellung kann im Inspektorfenster nicht editiert werden. Die lokale Fehlerbehandlung kann durch das Löschen der eingefügten Anweisungen zur lokalen Fehlerbehandlung wieder deaktiviert werden.

Hinweis

Bausteineigenschaft "Lokale Fehlerbehandlung im Baustein"

Diese Einstellung wird von einem aufrufenden Baustein nicht übernommen und auch nicht auf aufgerufene Programmbausteine übertragen. Für übergeordnete und untergeordnete Programmbausteine gelten weiterhin die Systemeinstellungen, sofern für diese keine eigene lokale Fehlerbehandlung programmiert wurde.

Prioritäten bei der Fehlerausgabe

Bei der lokalen Fehlerbehandlung mithilfe der Anweisungen GET_ERROR oder GET_ERR_ID wird die Information zum ersten aufgetretenen Fehler ausgegeben. Wenn während der Bearbeitung einer Anweisung mehrere Fehler gleichzeitig auftreten, werden die Fehler nach ihrer Priorität ausgegeben. Die folgende Tabelle zeigt, welche Priorität die verschiedenen Fehlerarten haben:

Priorität	Fehlerart
1	Fehler im Programmcode
2	Fehlende Referenz
3	Ungültiger Bereich

Priorität	Fehlerart
4	DB existiert nicht
5	Operanden nicht kompatibel
6	Breite des angegebenen Bereichs reicht nicht aus
7	Zeiten oder Zähler existieren nicht
8	Kein Schreibzugriff auf einen DB
9	Peripheriefehler
10	Anweisung existiert nicht
11	Baustein existiert nicht
12	Ungültige Schachtelungstiefe

Die höchste Priorität ist 1 und die niedrigste Priorität ist 12.

Weitere Informationen

Weitere Informationen zu den Anweisungen finden Sie im Informationssystem unter PLC programmieren > Anweisungen > Anweisungen (S7-1200, S7-1500) > Einfache Anweisungen. Die Anweisungen stehen Ihnen in den Programmiersprachen KOP/FUP/AWL/SCL und GRAPH zur Verfügung.

Siehe auch

GET_ERROR: Fehler lokal abfragen (Seite 1272)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

GET_ERROR: Fehler lokal abfragen (Seite 912)

1.5.6 Beispiel für die Behandlung von Programmablauffehlern

Einführung

Die lokalen Fehlerbehandlungsmöglichkeiten können sowohl einzeln als auch in Kombination miteinander programmiert werden. Um sicherzustellen, dass jedes Fehlerbild, dass innerhalb Ihres Programms auftreten kann, erkannt wird, empfehlen wir Ihnen eine Kombination der lokalen Fehlerbehandlungsmöglichkeiten, wie wir Sie Ihnen im unten stehenden Beispiel zeigen.

Zur genaueren Fehleranalyse können Sie neben dem Ausgangsparameter RET_VAL auch die Anweisungen "GET_ERROR" oder "GET_ERR_ID" verwenden. Diese Möglichkeiten liefern Ihnen Fehlercodes, deren genauere Erklärung Sie den Beschreibungen der einzelnen Anweisungen entnehmen können.

Es gibt aber auch Fehlerbilder, bei denen der Ausgangsparameter RET_VAL keinen gültigen Fehlercode ausgibt. Kommt es z. B. beim Lesen eines Eingangsparameters zu einem Zugriffsfehler, dann werden die Ausgänge der Anweisung nicht mehr geschrieben, da die Ausführung der Anweisung unterbrochen wird. In diesem Fall empfehlen wir Ihnen, die beiden

Anweisungen "GET_ERROR" oder "GET_ERR_ID" in Ihr Programm zu integrieren, da sie auch bei diesem Fehlerfall zuverlässige Fehlerinformationen liefern.

⚠ WARNUNG
Zugriffsfehler beim Lesen eines Eingangsparameters Der Parameter RET_VAL liefert keinen gültigen Fehlercode und im Diagnosepuffer wird keine genaue Fehlerinformation ausgegeben.

Ein erster Indikator für einen Fehler kann entweder das BIE-Bit des Statusworts oder der Freigabeausgang ENO sein. Wenn diese den Signalzustand "0" liefern, dann liegt ein Fehler bei der Ausführung der Anweisung vor. Beim Signalzustand "1" liegt kein Fehler vor und es ist keine weitere Fehlerauswertung notwendig. Außer, es handelt sich um einen Speicherzugriffsfehler. Dann kann auch der Signalzustand "1" einen Fehler bedeuten.

Vorgehen

Das folgende Beispiel zeigt Ihnen, wie Sie einen Zugriffsfehler beim Lesen eines Eingangsparameters erkennen können.

1. Deklarieren Sie die Bausteinschnittstelle Ihres Programmbausteins wie folgt:

Interface			
	Name	Data type	Default value
1	Input		
2	SrcField	Array [1..100] of Real	
3	index	DInt	0
4	<Add new>		
5	Output		
6	TagOutput	Real	0.0
7	Test2	Bool	false
8	TagRet_Val	Int	0
9	DestField	Array [0..20] of Real	
10	Test3	Bool	false
11	<Add new>		
12	InOut		
13	Static		
14	Temp		
15	TagError_ID	Word	

2. Schreiben Sie folgenden Programmcode:

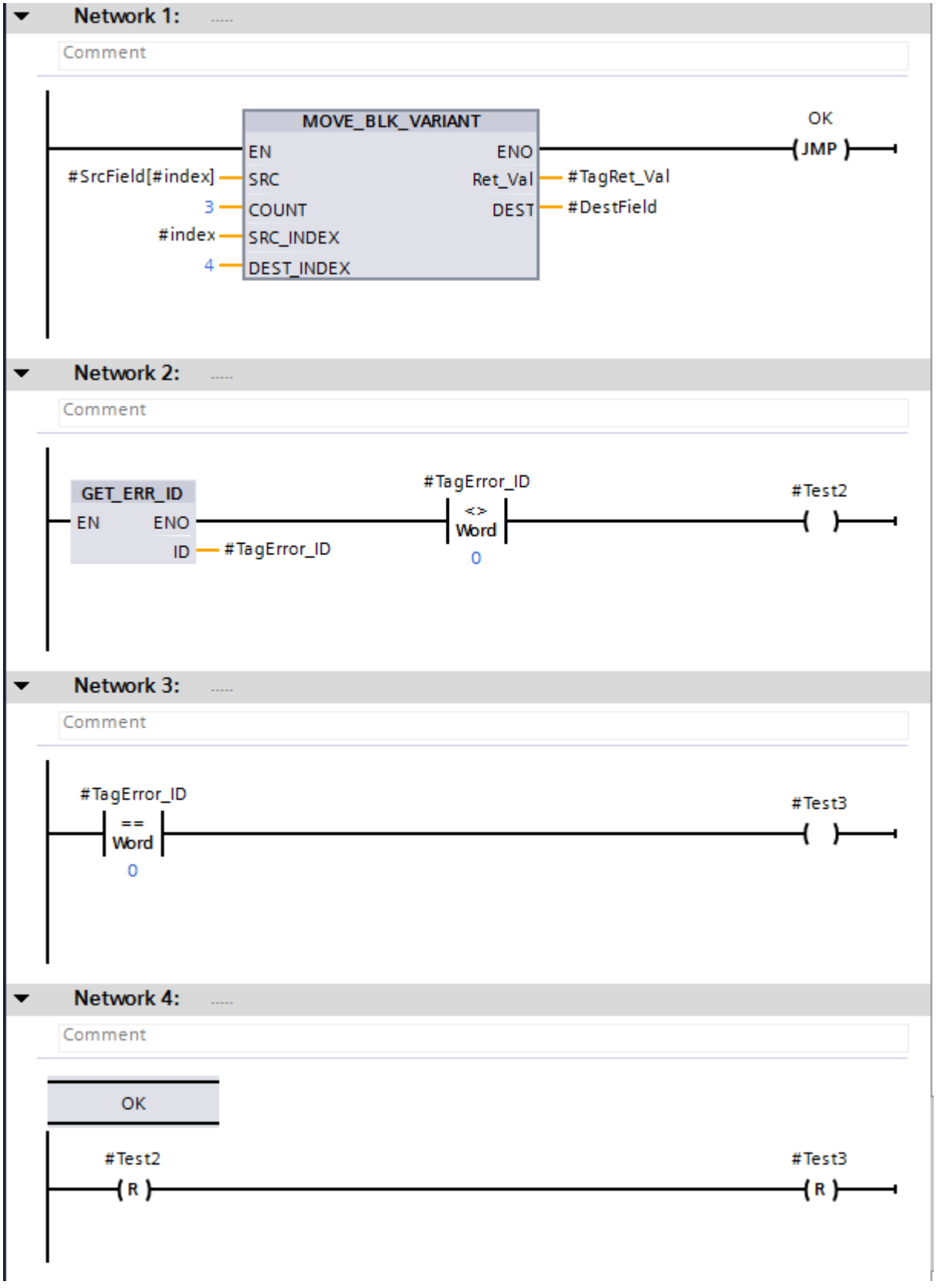
Network 1:
Comment
MOVE_BLK_VARIANT
EN
ENO
#SrcField[#index]
SRC
Ret_Val
#TagRet_Val
3
COUNT
DEST
#DestField
#index
SRC_INDEX
4
DEST_INDEX
OK
JMP
Network 2:
Comment
GET_ERR_ID
EN
ENO
ID
#TagError_ID
#TagError_ID
<>
Word
0
#Test2
Network 3:
Comment
#TagError_ID
==
Word
0
#Test3
Network 4:
Comment
OK
#Test2
R
#Test3
R

Im Netzwerk 1 wird die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" aufgerufen. Am Parameter SRC wird mithilfe eines variablen Index auf den Quellbereich "SrcField" zugegriffen. Wenn während der Bearbeitung der Anweisung keine Fehler auftreten, dann liefert der Freigabeausgang ENO den Signalzustand "1" und die Programmbearbeitung springt in das Netzwerk 4 und wird dort fortgeführt.

Wenn es während der Bearbeitung der Anweisung, z. B. aufgrund des variablen Index zu einem Zugriffsfehler kommt, dann liefert die Anweisung "GET_ERR_ID: Fehler-ID lokal abfragen" im Netzwerk 2 eine Fehler-ID. Der Vergleich der Fehler-ID auf "UNGLEICH" mit dem Wert "0" im Netzwerk 2 liefert das Ergebnis #Test2 = TRUE. Der Vergleich der Fehler-ID auf "GLEICH" mit dem Wert "0" im Netzwerk 3 liefert das Ergebnis #Test3 = TRUE.

Der Operand #TagRet_Val am Ausgangsparameter RET_VAL liefert in diesem Fall keinen gültigen Fehlercode.

Ausnahmen

Es gibt allerdings ein paar Anweisungen, bei denen Sie die Fehlerbehandlung nicht wie im oben genannten Beispiel programmieren können. Dabei handelt es sich um die folgenden Anweisungen:

- Anweisungen, die generell keinen EN-/ENO-Mechanismus haben
- Anweisungen, bei denen das ENO deaktiviert wurde
- S_COMP
- PEEK, PEEK_BOOL, POKE, POKE_BOOL und POKE_BLK

Das BIE-Bit bzw. der Freigabeausgang ENO wird bei diesen Anweisungen auf TRUE gesetzt, auch wenn es zu einem Zugriffsfehler kommt.

Das folgende Beispiel zeigt Ihnen, wie Sie in der Programmiersprache AWL eine zuverlässige Fehlerbehandlung programmieren können:

		Name	Data type
1		Input	
2		StringArray	Array[0..10] of String
3		index	DInt
4		Output	
5		TagResult	Bool
6		InOut	
7		Static	
8		Temp	
9		TagError_ID	Word
10		Constant	

AWL	Erklärung
SET L 0 T #Tag_ErrorID	// Der Operand #Tag_ErrorID wird mit "0" initialisiert.

AWL	Erklärung
CALL S_COMP	// Die Anweisung wird aufgerufen.
src_type := String	// Datentyp der Parameter IN1 und IN2
relation := EQ	// Der Vergleichstyp der Anweisung
IN1 := #StringArray.THIS[#index]	// Auf die ARRAY-Komponente wird variabel zugegriffen.
IN2 := 'STRING'	// Die beiden Werte werden miteinander verglichen.
OUT := #TagResult	// Wenn die beiden Werte gleich sind, dann erhält der Operand #TagResult den Signalzustand "1".
U BIE	// Das BIE-Bit wird abgefragt.
CALL GET_ERR_ID	// Die Anweisung wird aufgerufen.
RET_VAL := #Tag_ErrorID	// Bei einem Zugriffsfehler gibt die Anweisung einen Fehlercode aus.

Auch wenn das BIE-Bit den Signalzustand "1" hat, wird der Zugriffsfehler erkannt. Über die Auswertung des Operanden #Tag_ErrorID der Anweisung "GET_ERR_ID: Fehler-ID lokal abfragen" können Sie den Fehlercode abfragen.

1.6 Programmflusssteuerung

1.6.1 Statuswort bei S7-300, S7-400

1.6.1.1 Grundlagen zum Statuswort

Beschreibung

Das Statuswort fasst Statusbits zusammen, die die CPU zur Steuerung der binären Verknüpfungen verwendet und bei der digitalen Bearbeitung setzt. Sie können die Statusbits abfragen und gezielt beeinflussen.

Die folgende Tabelle zeigt die Anordnung der Statusbits im Statuswort:

	Statuswort									
Bitnummer	15-9	8	7	6	5	4	3	2	1	0
Inhalt	0	BIE	A1	A0	OV	OS	OR	STA	VKE	/ER

Die Statusbit /ER, VKE, STA, OR und BIE sind Binäranzeigen, die die CPU bei Binäranweisungen verwendet. Die Statusbits OS, OV, A0 und A1 sind Digitalanzeigen, die hauptsächlich Ergebnisse von mathematischen Funktionen anzeigen.

/ER (Erstabfrage)

Der Signalzustand des /ER-Bits steuert den Ablauf einer Verknüpfungskette.

Eine Verknüpfungskette beginnt mit einem Signalzustand "0" des /ER-Bits und einer binären Abfrageanweisung (Erstabfrage). Die Erstabfrage entspricht der ersten binären Bitverknüpfung bzw. Abfrage in einem Netzwerk. Diese setzt das /ER-Bit auf "1". Die Verknüpfungskette endet mit einer binären Wertzuweisung (z. B. mit der Anweisung "Setzen"), mit einem bedingten Sprung oder einem Bausteinwechsel. Diese setzen das /ER-Bit auf "0".

VKE (Verknüpfungsergebnis)

Das Statusbit VKE ist der Zwischenspeicher bei binären Verknüpfungen.

Die CPU überträgt bei einer Erstabfrage das Abfrageergebnis ins VKE. Bei jeder folgenden Abfrage wird das Abfrageergebnis mit dem gespeicherten VKE verknüpft und das Ergebnis wiederum im VKE abgelegt.

Sie können das VKE mithilfe der entsprechenden Anweisungen setzen, rücksetzen, negieren oder im Binärergebnis (BIE) speichern.

Mit dem VKE werden Speicher-, Zeit-, und Zählanweisungen gesteuert und bestimmte Sprunganweisungen ausgeführt.

STA (Status)

Das Statusbit STA entspricht dem Signalzustand des abgefragten Binäroperanden.

Der Status einer Verknüpfungsoperation, die Lesezugriff auf den Speicher hat, ist immer gleich dem Wert des adressierten Bits. Der Status einer Verknüpfungsoperation, die Schreibzugriff auf den Speicher hat (z. B. die Anweisungen "Setzen" oder "Rücksetzen"), ist gleich dem Wert des Bits, in das die Anweisung schreibt. Falls nicht geschrieben wird, ist der Status gleich dem Wert des adressierten Bits. Das Statusbit hat keine Bedeutung für Verknüpfungsanweisungen, die nicht auf den Speicher zugreifen. Diese Anweisungen setzen das Statusbit auf "1". Das Statusbit wird von Anweisungen nicht abgefragt. Es wird lediglich beim Anzeigen des Online-Status von Programmvariablen ausgewertet.

OR (ODER-Bit)

Das Statusbit OR wird benötigt, wenn eine UND-Verknüpfung vor einer ODER-Verknüpfung durchgeführt wird.

Das OR-Bit wird gesetzt, wenn das VKE der UND-Verknüpfung "1" ist. Damit wird das Ergebnis der ODER-Verknüpfung vorweggenommen. Jede andere binäre Anweisung setzt das OR-Bit zurück.

OS (Überlauf speichernd)

Das Statusbit OS speichert ein Setzen des Statusbits OV.

Wenn die CPU das Statusbit OV setzt, setzt sie auch das Statusbit OS. Während jedoch die nächste ordnungsgemäß ausgeführte Anweisung das Bit OV wieder zurücksetzt, bleibt das Statusbit OS gesetzt. Somit haben Sie die Möglichkeit, an einer späteren Stelle im aktuellen Zyklus der CPU einen Zahlenbereichsüberlauf oder z. B. die Verwendung ungültiger Gleitpunktzahlen abzufragen.

OV (Überlauf)

Das Statusbit OV zeigt einen Zahlenbereichsüberlauf oder die Verwendung ungültiger Gleitpunktzahlen.

Das Statusbit OV kann durch mathematische Funktionen, Umwandlungsanweisungen und Vergleiche von Gleitpunktzahlen beeinflusst werden.

A0 und A1 (Anzeigebits)

Die Statusbits A0 und A1 geben Auskunft über das Ergebnis der folgenden Anweisungen:

- Vergleichsanweisungen
- Mathematische Funktionen
- Wortverknüpfungen
- Schiebe- und Rotieranweisungen

BIE (Binärergebnis)

Das Statusbit BIE dient zur Realisierung des EN-/ENO-Mechanismus für Boxen oder als Bedingung bei einigen Sprunganweisungen (AWL). Sie können das Statusbit BIE durch bestimmte Anweisungen (z. B. SAVE) beeinflussen.

1.6.1.2 Setzen der Statusbits

Setzen der Statusbits bei Bitverknüpfungen

Beschreibung

Binäre Anweisungen beeinflussen die Statusbits OR, STA, VKE und /ER.

Setzen der Statusbits in KOP

Die folgende Tabelle zeigt das Setzen der Statusbits in der Programmiersprache KOP:

Anweisung		Statusbit			
Mnemonik	Titel	OR	STA	VKE	/ER
-| |-	Schließerkontakt	x	x	x	1
-| / |-	Öffnerkontakt	x	x	x	1
-|NOT|-	VKE invertieren	-	1	x	-
-( )-	Zuweisung	0	x	-	0
-(R)-	Ausgang rücksetzen	0	x	-	0
-(S)-	Ausgang setzen	0	x	-	0
SR	Flipflop setzen/rücksetzen	x	x	x	1
RS	Flipflop rücksetzen/setzen	x	x	x	1
-|P|-	Operand auf positive Signalflanke abfragen	0	1	x	1
-|N|-	Operand auf negative Signalflanke abfragen	0	1	x	1
P_TRIG	VKE auf positive Signalflanke abfragen	0	x	x	1
N_TRIG	VKE auf negative Signalflanke abfragen	0	x	x	1
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.					

Setzen der Statusbits in FUP

Die folgende Tabelle zeigt das Setzen der Statusbits in der Programmiersprache FUP:

Anweisung		Statusbit			
Mnemonik	Titel	OR	STA	VKE	/ER
&	UND-Verknüpfung	x	x	x	1
<=1	ODER-Verknüpfung	x	x	x	1
X	EXKLUSIV ODER-Verknüpfung	x	x	x	1
-|	Binären Eingang einfügen	-	1	x	-
-o|	VKE invertieren	-	1	x	-
-( )-	Zuweisung	0	x	-	0
R	Ausgang rücksetzen	0	x	-	0

Anweisung		Statusbit			
Mnemonik	Titel	OR	STA	VKE	/ER
S	Ausgang setzen	0	x	-	0
SR	Flipflop setzen/rücksetzen	x	x	x	1
RS	Flipflop rücksetzen/setzen	x	x	x	1
P	Operand auf positive Signalflanke abfragen	0	1	x	1
N	Operand auf negative Signalflanke abfragen	0	1	x	1
P_TRIG	VKE auf positive Signalflanke abfragen	0	x	x	1
N_TRIG	VKE auf negative Signalflanke abfragen	0	x	x	1
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.					

Setzen der Statusbits in AWL

Die folgende Tabelle zeigt das Setzen der Statusbits in der Programmiersprache AWL:

Anweisung		Statusbit			
Mnemonik	Titel	OR	STA	VKE	/ER
U	UND-Verknüpfung	0	x	x	1
UN	Negierte UND-Verknüpfung	0	x	x	1
O	ODER-Verknüpfung	0	x	x	1
ON	Negierte ODER- Verknüpfung	0	x	x	1
X	EXKLUSIV ODER- Verknüpfung	0	x	x	1
XN	Negierte EXKLUSIV ODER- Verknüpfung	0	x	x	1
O	ODER- Verknüpfung von UND-Funktionen	0	1	-	0
U(	UND- Verknüpfung mit Verzweigung	0	1	-	0
UN(	Negierte UND- Verknüpfung mit Verzweigung	0	1	-	0
O(	ODER- Verknüpfung mit Verzweigung	0	1	-	0
ON(	Negierte ODER- Verknüpfung mit Verzweigung	0	1	-	0
X(	EXKLUSIV ODER- Verknüpfung mit Verzweigung	0	1	-	0
XN(	Negierte EXKLUSIV ODER- Verknüpfung mit Verzweigung	0	1	-	0
=	Zuweisung	0	x	-	0
R	Ausgang rücksetzen	0	x	-	0
S	Ausgang setzen	0	x	-	0
NOT	VKE invertieren	-	1	x	-
SET	VKE auf 1 setzen	0	1	1	0
CLR	VKE auf 0 rücksetzen	0	0	0	0
SAVE	VKE im BIE-Bit speichern	-	-	-	-
FN	VKE auf negative Signalflanke abfragen	0	x	x	1

Anweisung		Statusbit			
Mnemonik	Titel	OR	STA	VKE	/ER
FP	VKE auf positive Signalflanke abfragen	0	x	x	1
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.					

Setzen der Statusbits bei Zeit- und Zählanweisungen

Beschreibung

Die Zeit- und Zählanweisungen beeinflussen die Statusbits OR, STA, VKE und /ER.

Setzen der Statusbits in KOP und FUP

Die folgende Tabelle zeigt das Setzen der Statusbits in den Programmiersprachen KOP und FUP:

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
S_IMPULS	Zeit als Impuls parametrieren und starten	-	x	x	x	1
S_VIMP	Zeit als verlängerten Impuls parametrieren und starten	-	x	x	x	1
S_EVERZ	Zeit als Einschaltverzögerung parametrieren und starten	-	x	x	x	1
S_SEVERZ	Zeit als speichernde Einschaltverzögerung parametrieren und starten	-	x	x	x	1
S_AVERZ	Zeit als Ausschaltverzögerung parametrieren und starten	-	x	x	x	1
SI	Zeit als Impuls starten	-	0	-	-	0
SV	Zeit als verlängerten Impuls starten	-	0	-	-	0
SE	Zeit als Einschaltverzögerung starten	-	0	-	-	0
SS	Zeit als speichernde Einschaltverzögerung starten	-	0	-	-	0
SA	Zeit als Ausschaltverzögerung starten	-	0	-	-	0
ZÄHLER	Parametrieren und vorwärts/rückwärts zählen	-	x	x	x	1
Z_VORW	Parametrieren und vorwärts zählen	-	x	x	x	1
Z_RUECK	Parametrieren und rückwärts zählen	-	x	x	x	1
SZ	Zähleranfangswert setzen	-	0	x	-	0
ZV	Vorwärts zählen	-	0	x	-	0

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
ZR	Rückwärts zählen	-	0	-	-	0
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.						

Setzen der Statusbits in AWL

Die folgende Tabelle zeigt das Setzen der Statusbits in der Programmiersprache AWL:

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
FR	Zeit freigeben	-	0	-	-	0
R	Zeit rücksetzen	-	0	-	-	0
SI	Zeit als Impuls starten	-	0	-	-	0
SV	Zeit als verlängerten Impuls starten	-	0	-	-	0
SE	Zeit als Einschaltverzögerung starten	-	0	-	-	0
SS	Zeit als speichernde Einschaltverzögerung starten	-	0	-	-	0
SA	Zeit als Ausschaltverzögerung starten	-	0	-	-	0
FR	Zähler freigeben	-	0	-	-	0
R	Zähler rücksetzen	-	0	-	-	0
S	Zähler setzen	-	0	-	-	0
ZV	Vorwärts zählen	-	0	-	-	0
ZR	Rückwärts zählen	-	0	-	-	0
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.						

Setzen der Statusbits bei Anweisungen mit Ganzzahlen

Beschreibung

Anweisungen mit Ganzzahlen beeinflussen die Statusbits A1, A0, OV und OS im Statuswort.

Bei einem negativen Ergebnis werden die Statusbits A1 und A0 auf "0" gesetzt. Ein positives Ergebnis setzt das Statusbit A1 auf "1" und das Statusbit A0 auf "0". Das Statusbit A1 wird auf "0" und das Statusbit A0 auf "1" gesetzt, wenn das Ergebnis negativ ist.

Bei einem Zahlenbereichsüberlauf werden die Statusbits OV und OS auf "1" gesetzt. Eine Division durch Null wird mit dem Signalzustand "0" der Statusbits A1, A0, OV und OS angezeigt.

Die folgenden Tabellen zeigen das Setzen der Statusbits bei der Verwendung von Ganzzahlen:

Gültiger Bereich	A1	A0	OV	OS
0 (Null)	0	0	0	*
16 Bit: -32768 <= Ergebnis < 0 (negative Zahl) 32 Bit: -2147483648 <=Ergebnis < 0 (negative Zahl)	0	1	0	*
16 Bit: 32767 >= Ergebnis > 0 (positive Zahl) 32 Bit: 2147483647 >= Ergebnis > 0 (positive Zahl)	1	0	0	*
* Das OS-Bit wird vom Ergebnis der Operation nicht beeinflusst.				

Ungültiger Bereich	A1	A0	OV	OS
Unterschreitung bei Addition 16 Bit: Ergebnis = - 65536 32 Bit: Ergebnis = - 4294967296	0	0	1	1
Unterschreitung bei Multiplikation 16 Bit: Ergebnis < -32768 (negative Zahl) 32 Bit: Ergebnis < -2147483648 (negative Zahl)	0	1	1	1
Überlauf bei Addition, Subtraktion 16 Bit: Ergebnis > 32767 (positive Zahl) 32 Bit: Ergebnis > 2147483647 (positive Zahl)	0	1	1	1
Überlauf bei Multiplikation, Division 16 Bit: Ergebnis > 32767 (positive Zahl) 32 Bit: Ergebnis > 2147483647 (positive Zahl)	1	0	1	1
Unterschreitung bei Addition, Subtraktion 16 Bit: Ergebnis < -32768 (negative Zahl) 32 Bit: Ergebnis < -2147483648 (negative Zahl)	1	0	1	1
Division durch 0	1	1	1	1

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen

Beschreibung

Anweisungen mit Gleitpunktzahlen beeinflussen die Statusbits A1, A0, OV und OS im Statuswort.

Bei einem negativen Ergebnis werden die Statusbits A1 und A0 auf "0" gesetzt. Ein positives Ergebnis setzt das Statusbit A1 auf "1" und das Statusbit A0 auf "0". Das Statusbit A1 wird auf "0" und das Statusbit A0 auf "1" gesetzt, wenn das Ergebnis negativ ist.

Bei einem Zahlenbereichsüberlauf werden die Statusbits OV und OS auf "1" gesetzt. Eine ungültige Gleitpunktzahl wird mit dem Signalzustand "0" der Statusbits A1, A0, OV und OS angezeigt.

Die folgenden Tabellen zeigen das Setzen der Statusbits bei der Verwendung von Gleitpunktzahlen:

Gültiger Bereich	A1	A0	OV	OS
+0, -0 (Null)	0	0	0	*
-3,402823E+38 < Ergebnis < -1,175494E-38 (negative Zahl)	0	1	0	*
+1,175494E-38 < Ergebnis < 3,402824E +38 (positive Zahl)	1	0	0	*
* Das OS-Bit wird vom Ergebnis der Operation nicht beeinflusst.				

Ungültiger Bereich	A1	A0	OV	OS
Unterschreitung -1,175494E-38 < Ergebnis < -1,401298E-45 (negative Zahl)	0	0	1	1
Unterschreitung +1,401298E-45 < Ergebnis < +1,175494E-38 (positive Zahl)	0	0	1	1
Überlauf Ergebnis < -3,402823E+38 (negative Zahl)	0	1	1	1
Überlauf Ergebnis > 3,402823E+38 (positive Zahl)	1	0	1	1
Keine gültige Gleitpunktzahl oder unzulässige Operation (Eingangswert außerhalb des gültigen Wertebereichs)	1	1	1	1

Setzen der Statusbits bei Vergleichsanweisungen

Beschreibung

Vergleichsanweisungen beeinflussen die Statusbits A1, A0, OV, OS im Statuswort.

Vergleichsanweisungen setzen die Statusbits A1 und A0 unabhängig von der Vergleichsrelation.

Bei der Verwendung ungültiger Gleitpunktzahlen werden die Statusbits A1, A0, OV und OS auf "1" gesetzt.

Die folgende Tabelle zeigt das Setzen der Statusbits bei Vergleichsanweisungen:

Ergebnis	A1	A0	OV	OS
Gleich (AKKU2 = AKKU1)	0	0	0	*
Größer (AKKU2 > AKKU1)	1	0	0	*
Kleiner (AKKU2 < AKKU1)	0	1	0	*
Ungültige Gleitpunktzahl	1	1	1	1
* Das OS-Bit wird vom Ergebnis der Operation nicht beeinflusst.				

Setzen der Statusbits bei Anweisungen zur Programmsteuerung

Beschreibung

Die Anweisungen zur Programmsteuerung beeinflussen die Statusbits OS, OR, STA, VKE und /ER.

Setzen der Statusbits in KOP und FUP

Die folgende Tabelle zeigt das Setzen der Statusbits in den Programmiersprachen KOP und FUP:

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
JMP	Springen bei VKE = 1	-	0	1	1	0
JMPN	Springen bei VKE = 0	-	0	1	1	0
RET	Zurück springen	0	0	1	1	0
CALL	Baustein aufrufen	0	0	1	-	0
OPN	Globalen Datenbaustein öffnen	-	-	-	-	-
OPNI	Instanz-Datenbaustein öffnen	-	-	-	-	-
MCR<	MCR-Bereiche öffnen	-	0	1	-	0
MCR>	MCR-Bereiche schließen	-	0	1	-	0
MCRA	MCR-Bereich aktivieren	-	-	-	-	-
MCRD	MCR-Bereich deaktivieren	-	-	-	-	-
x: Die Anweisung kann "1" oder "0" schreiben 1: Die Anweisung setzt das Statusbit auf "1". 0: Die Anweisung setzt das Statusbit auf "0" zurück. -: Das Statusbit wird nicht beeinflusst.						

Setzen der Statusbits in AWL

Die folgende Tabelle zeigt das Setzen der Statusbits in der Programmiersprache AWL:

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
SPA	Absolut springen	-	-	-	-	-
SPB	Springen bei VKE = 1	-	0	1	1	0
SPBN	Springen bei VKE = 0	-	0	1	1	0
SPBB	Springen bei VKE = 1 und VKE speichern	-	0	1	1	0
SPBNB	Springen bei VKE = 0 und VKE speichern	-	0	1	1	0
SPBI	Springen bei BIE = 1	-	0	1	-	0
SPBIN	Springen bei BIE = 1	-	0	1	-	0
SPO	Springen bei OV = 1	-	-	-	-	-
SPS	Springen bei OS = 1	0	-	-	-	-
SPZ	Springen bei Ergebnis Null	-	-	-	-	-

Anweisung		Statusbit				
Mnemonik	Titel	OS	OR	STA	VKE	/ER
SPN	Springen bei Ergebnis nicht Null	-	-	-	-	-
SPP	Springen bei Ergebnis größer Null	-	-	-	-	-
SPM	Springen bei Ergebnis kleiner Null	-	-	-	-	-
SPPZ	Springen bei Ergebnis größer oder gleich Null	-	-	-	-	-
SPMZ	Springen bei Ergebnis kleiner oder gleich Null	-	-	-	-	-
SPU	Springen bei Ergebnis ungültig	-	-	-	-	-
SPL	Sprungliste definieren	-	-	-	-	-
LOOP	Schleifensprung	-	-	-	-	-
AUF	Globalen Datenbaustein öffnen	-	-	-	-	-
AUFDI	Instanz-Datenbaustein öffnen	-	-	-	-	-
CALL	Baustein aufrufen	0	0	1	-	0
CC	Baustein bedingt aufrufen	0	0	1	1	0
UC	Baustein unbedingt aufrufen	0	0	1	-	0
BE	Bausteinende	0	0	1	-	0
BEB	Bedingtes Bausteinende	x	0	1	1	0
BEA	Absolutes Bausteinende	0	0	1	-	0
MCR(	MCR-Bereiche öffnen	-	0	1	-	0
)MCR	MCR-Bereiche schließen	-	0	1	-	0
MCRA	MCR-Bereich aktivieren	-	-	-	-	-
MCRD	MCR-Bereich deaktivieren	-	-	-	-	-

x: Die Anweisung kann "1" oder "0" schreiben
1: Die Anweisung setzt das Statusbit auf "1".
0: Die Anweisung setzt das Statusbit auf "0" zurück.
-: Das Statusbit wird nicht beeinflusst.

Setzen der Statusbits bei Wortverknüpfungen

Beschreibung

Wortverknüpfungen beeinflussen die Statusbits A1, A0 und OV im Statuswort.

Wenn im Ergebnis einer Wortverknüpfung alle Bits "0" sind, wird das Statusbit A1 auf "0" zurückgesetzt. Wenn mindestens ein Bit im Ergebnis "1" ist, wird das Statusbit A1 auf "1" gesetzt.

Die folgende Tabelle zeigt das Setzen der Statusbits bei Wortverknüpfungen:

Ergebnis	A1	A0	OV
Null	0	0	0
Nicht Null	1	0	0

Setzen der Statusbits bei Schiebe- und Rotieranweisungen

Beschreibung

Schiebe- und Rotieranweisungen beeinflussen die Statusbits A1, A0 und OV im Statuswort.

Bei Schiebe- und Rotieranweisungen wird der Signalzustand des zuletzt hinausgeschobenen bzw. rotierten Bit in das Statusbit A1 übertragen. Die Statusbits A0 und OV werden zurückgesetzt.

Die folgende Tabelle zeigt das Setzen der Statusbits bei Schiebe- und Rotieranweisungen:

Hinausgeschobenes/Rotiertes Bit	A1	A0	OV
"0"	0	0	0
"1"	1	0	0
Bei Schiebe- oder Rotierzahl 0	*	*	*
* Das Bit wird vom Ergebnis der Operation nicht beeinflusst.			

1.6.1.3 Abfragen der Statusbits in AWL

Abfragen der Statusbits mit Bitverknüpfungsanweisungen

Beschreibung

Mit den folgenden Bitverknüpfungsanweisungen kann der Signalzustand der Bits im Statuswort abgefragt werden:

- U: UND-Verknüpfung
- UN: Negierte UND-Verknüpfung
- O: ODER-Verknüpfung
- ON: Negierte ODER-Verknüpfung
- X: EXKLUSIV ODER-Verknüpfung
- XN: Negierte EXKLUSIV ODER-Verknüpfung

Die folgende Tabelle zeigt die Operanden, die zur Abfrage der Signalzustände der Bits im Statuswort verwendet werden können:

Operand	Abfrage des Statusworts
== 0	Abfrage nach ((A0 = 0) UND (A1 = 0))
<> 0	Abfrage nach A0 <>A1
> 0	Abfrage nach ((A0 = 0) UND (A1 = 1))
< 0	Abfrage nach ((A0 = 1) UND (A1 = 0))
>= 0	Abfrage nach A0 = 0
<= 0	Abfrage nach A1 = 0
UO	Abfrage nach ((A1 = 1) UND (A0 = 1))
OV	Abfrage nach OV = 1

Operand	Abfrage des Statusworts
OS	Abfrage nach OS = 1
BIE	Abfrage nach BIE = 1

Siehe auch

U: UND-Verknüpfung (Seite 4917)
UN: Negierte UND-Verknüpfung (Seite 4919)
O: ODER-Verknüpfung (Seite 4920)
ON: Negierte ODER-Verknüpfung (Seite 4921)
X: EXKLUSIV ODER-Verknüpfung (Seite 4923)
XN: Negierte EXKLUSIV ODER-Verknüpfung (Seite 4924)
O: ODER-Verknüpfung von UND-Funktionen (Seite 4925)
U(: UND-Verknüpfung mit Verzweigung (Seite 4926)
UN(: Negierte UND-Verknüpfung mit Verzweigung (Seite 4927)
O(: ODER-Verknüpfung mit Verzweigung (Seite 4928)
ON(: Negierte ODER-Verknüpfung mit Verzweigung (Seite 4929)
X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung (Seite 4930)
XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung (Seite 4931)
): Verzweigung schließen (Seite 4932)
=: Zuweisung (Seite 4933)
R: Rücksetzen (Seite 4934)
S: Setzen (Seite 4935)
NOT: VKE invertieren (Seite 4936)
SET: VKE auf 1 setzen (Seite 4937)
CLR: VKE auf 0 rücksetzen (Seite 4938)
SAVE: VKE im BIE-Bit speichern (Seite 4939)
FN: VKE auf negative Signalflanke abfragen (Seite 4940)
FP: VKE auf positive Signalflanke abfragen (Seite 4942)

Abfragen der Statusbits mit Sprunganweisungen

Beschreibung

Die folgende Tabelle zeigt die Auswertung der Statusbits durch die Sprunganweisungen:

VKE	BIE	A1	A0	OV	OS	Mnemonik der ausgeführten Anweisung
1	-	-	-	-	-	SPB, SPBB
0	-	-	-	-	-	SPBN, SPBNB
-	1	-	-	-	-	SPBI
-	0	-	-	-	-	SPBIN
-	-	0	0	-	-	SPZ, SPMZ, SPPZ
-	-	1	0	-	-	SPN, SPP, SPPZ
-	-	0	1	-	-	SPN, SPM, SPMZ
-	-	1	1	-	-	SPU
-	-	-	-	1	-	SPO
-	-	-	-	-	1	SPS

Die folgende Tabelle zeigt die Abhängigkeit zwischen den Statusbits A1 und A0 und den bedingten Sprunganweisungen:

A1	A0	Ergebnis	Mnemonik der ausgeführten Anweisung
0	0	= 0	SPZ
1 oder 0	0 oder 1	<> 0	SPN
1	0	> 0	SPP
0	1	< 0	SPM
0 oder 1	0	>= 0	SPPZ
0	0 oder 1	<= 0	SPMZ
1	1	UO (ungültig)	SPU

Siehe auch

SPA: Absolut springen (Seite 5034)

SPB: Springen bei VKE = 1 (Seite 5035)

SPBN: Springen bei VKE = 0 (Seite 5036)

SPBB: Springen bei VKE = 1 und VKE speichern (Seite 5037)

SPBNB: Springen bei VKE = 0 und VKE speichern (Seite 5039)

SPBI: Springen bei BIE = 1 (Seite 5040)

SPBIN: Springen bei BIE = 0 (Seite 5041)

SPO: Springen bei OV = 1 (Seite 5042)

SPS: Springen bei OS = 1 (Seite 5043)

SPZ: Springen bei Ergebnis Null (Seite 5045)

SPN: Springen bei Ergebnis nicht Null (Seite 5047)

SPP: Springen bei Ergebnis größer Null (Seite 5048)

SPM: Springen bei Ergebnis kleiner Null (Seite 5050)

SPPZ: Springen bei Ergebnis größer oder gleich Null (Seite 5052)

SPMZ: Springen bei Ergebnis kleiner oder gleich Null (Seite 5053)

SPU: Springen bei Ergebnis ungültig (Seite 5055)

SPL: Sprungliste definieren (Seite 5057)

LOOP: Schleifensprung (Seite 5058)

1.6.2 Statuswort bei S7-1500

1.6.2.1 Grundlagen zum Statuswort

Beschreibung

Das Statuswort fasst Statusbits zusammen, die die CPU zur Steuerung der binären Verknüpfungen verwendet und bei der digitalen Bearbeitung setzt. Sie können die Statusbits abfragen und gezielt beeinflussen.

Die folgende Tabelle zeigt die Anordnung der Statusbits im Statuswort:

	Statuswort									
Bitnummer	15-9	8	7	6	5	4	3	2	1	0
Inhalt	0	BIE	A1	A0	OV	OS	0	0	0	0

Die Statusbits OS, OV, A0 und A1 sind Digitalanzeigen, die hauptsächlich Ergebnisse von mathematischen Funktionen anzeigen.

OS (Überlauf speichernd)

Das Statusbit OS speichert ein Setzen des Statusbits OV.

Wenn die CPU das Statusbit OV setzt, setzt sie auch das Statusbit OS. Während jedoch die nächste ordnungsgemäß ausgeführte Anweisung das Bit OV wieder zurücksetzt, bleibt das Statusbit OS gesetzt. Somit haben Sie die Möglichkeit, an einer späteren Stelle im aktuellen Baustein der CPU einen Zahlenbereichsüberlauf oder z. B. die Verwendung ungültiger Gleitpunktzahlen abzufragen.

OV (Überlauf)

Das Statusbit OV zeigt einen Zahlenbereichsüberlauf oder die Verwendung ungültiger Gleitpunktzahlen.

Das Statusbit OV kann durch mathematische Funktionen, Umwandlungsanweisungen und Vergleiche von Gleitpunktzahlen beeinflusst werden.

A0 und A1 (Anzeigebits)

Die Statusbits A0 und A1 geben Auskunft über das Ergebnis der folgenden Anweisungen:

- Vergleichsanweisungen
- Mathematische Funktionen
- Wortverknüpfungen
- Schiebe- und Rotieranweisungen

BIE (Binärergebnis)

Das Statusbit BIE dient zur Realisierung des EN-/ENO-Mechanismus für Boxen oder als Bedingung bei einigen Sprunganweisungen (AWL). Sie können das Statusbit BIE durch bestimmte Anweisungen (z. B. SAVE) beeinflussen.

VKE (Verknüpfungsergebnis)

Das Statusbit VKE ist der Zwischenspeicher bei binären Verknüpfungen und kein Bestandteil des Statusworts.

Die CPU überträgt bei einer Erstabfrage das Abfrageergebnis ins VKE. Bei jeder folgenden Abfrage wird das Abfrageergebnis mit dem gespeicherten VKE verknüpft und das Ergebnis wiederum im VKE abgelegt.

Sie können das VKE mithilfe der entsprechenden Anweisungen setzen oder zurücksetzen.

Mit dem VKE werden Speicher-, Zeit-, und Zählanweisungen gesteuert und bestimmte Sprunganweisungen ausgeführt.

Hinweis

Da die Statusbits teilweise nur zur internen Verarbeitung benötigt werden, werden sie nur dann angezeigt, wenn sie zur weiteren Programmbearbeitung zur Verfügung stehen.

1.6.2.2 Abfragen und Setzen der Statusbits in AWL

Abfragen der Statusbits mit Bitverknüpfungsanweisungen

Beschreibung

Mit den folgenden Bitverknüpfungsanweisungen kann der Signalzustand der Bits im Statuswort abgefragt werden:

- U: UND-Verknüpfung
- UN: Negierte UND-Verknüpfung
- O: ODER-Verknüpfung

- ON: Negierte ODER-Verknüpfung
- X: EXKLUSIV ODER-Verknüpfung
- XN: Negierte EXKLUSIV ODER-Verknüpfung

Die folgende Tabelle zeigt die Operanden, die zur Abfrage der Signalzustände der Bits im Statuswort verwendet werden können:

Operand	Abfrage des Statusworts
== 0	Abfrage nach ((A0 = 0) UND (A1 = 0))
<> 0	Abfrage nach A0 <>A1
> 0	Abfrage nach ((A0 = 0) UND (A1 = 1))
< 0	Abfrage nach ((A0 = 1) UND (A1 = 0))
>= 0	Abfrage nach A0 = 0
<= 0	Abfrage nach A1 = 0
UO	Abfrage nach ((A1 = 1) UND (A0 = 1))
OV	Abfrage nach OV = 1
OS	Abfrage nach OS = 1
BIE	Abfrage nach BIE = 1

Siehe auch

U: UND-Verknüpfung (Seite 1542)

UN: Negierte UND-Verknüpfung (Seite 1543)

O: ODER-Verknüpfung (Seite 1544)

ON: Negierte ODER-Verknüpfung (Seite 1546)

X: EXKLUSIV ODER-Verknüpfung (Seite 1547)

XN: Negierte EXKLUSIV ODER-Verknüpfung (Seite 1548)

O: ODER-Verknüpfung von UND-Funktionen (Seite 1549)

U(: UND-Verknüpfung mit Verzweigung (Seite 1550)

UN(: Negierte UND-Verknüpfung mit Verzweigung (Seite 1551)

O(: ODER-Verknüpfung mit Verzweigung (Seite 1552)

ON(: Negierte ODER-Verknüpfung mit Verzweigung (Seite 1553)

X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung (Seite 1554)

XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung (Seite 1555)

): Verzweigung schließen (Seite 1556)

=: Zuweisung (Seite 1557)

R: Rücksetzen (Seite 1558)

S: Setzen (Seite 1558)

NOT: VKE invertieren (Seite 1559)

SET: VKE auf 1 setzen (Seite 1560)

CLR: VKE auf 0 rücksetzen (Seite 1561)

SAVE: VKE im BIE-Bit speichern (Seite 1562)

FN: VKE auf negative Signalflanke abfragen (Seite 1563)

FP: VKE auf positive Signalflanke abfragen (Seite 1564)

Abfragen der Statusbits mit Sprunganweisungen

Beschreibung

Die folgende Tabelle zeigt die Auswertung der Statusbits durch die Sprunganweisungen:

VKE	BIE	A1	A0	OV	OS	Mnemonik der ausgeführten Anweisung
1	-	-	-	-	-	SPB, SPBB
0	-	-	-	-	-	SPBN, SPBNB
-	1	-	-	-	-	SPBI
-	0	-	-	-	-	SPBIN
-	-	0	0	-	-	SPZ, SPMZ, SPPZ
-	-	1	0	-	-	SPN, SPP, SPPZ
-	-	0	1	-	-	SPN, SPM, SPMZ
-	-	1	1	-	-	SPU
-	-	-	-	1	-	SPO
-	-	-	-	-	1	SPS

Die folgende Tabelle zeigt die Abhängigkeit zwischen den Statusbits A0 und A1 und den bedingten Sprunganweisungen:

A1	A0	Ergebnis	Mnemonik der ausgeführten Anweisung
0	0	= 0	SPZ
1 oder 0	0 oder 1	<> 0	SPN
1	0	> 0	SPP
0	1	< 0	SPM
0 oder 1	0	>= 0	SPPZ
0	0 oder 1	<= 0	SPMZ
1	1	UO (ungültig)	SPU

Siehe auch

SPA: Absolut springen (Seite 5034)

SPB: Springen bei VKE = 1 (Seite 5035)

SPBN: Springen bei VKE = 0 (Seite 5036)

SPBB: Springen bei VKE = 1 und VKE speichern (Seite 5037)

SPBNB: Springen bei VKE = 0 und VKE speichern (Seite 5039)

SPBI: Springen bei BIE = 1 (Seite 5040)

SPBIN: Springen bei BIE = 0 (Seite 5041)

SPO: Springen bei OV = 1 (Seite 5042)

SPS: Springen bei OS = 1 (Seite 5043)

SPZ: Springen bei Ergebnis Null (Seite 5045)

SPN: Springen bei Ergebnis nicht Null (Seite 5047)

SPP: Springen bei Ergebnis größer Null (Seite 5048)

SPM: Springen bei Ergebnis kleiner Null (Seite 5050)

SPPZ: Springen bei Ergebnis größer oder gleich Null (Seite 5052)

SPMZ: Springen bei Ergebnis kleiner oder gleich Null (Seite 5053)

SPU: Springen bei Ergebnis ungültig (Seite 5055)

SPL: Sprungliste definieren (Seite 5057)

LOOP: Schleifensprung (Seite 5058)

Sprungmarke (Seite 1651)

SPA: Absolut springen (Seite 1652)

SPB: Springen bei VKE = 1 (Seite 1653)

SPBN: Springen bei VKE = 0 (Seite 1654)

SPBB: Springen bei VKE = 1 und VKE speichern (Seite 1655)

SPBNB: Springen bei VKE = 0 und VKE speichern (Seite 1657)

SPBI: Springen bei BIE = 1 (Seite 1658)

SPBIN: Springen bei BIE = 0 (Seite 1659)

SPO: Springen bei OV = 1 (Seite 1660)

SPS: Springen bei OS = 1 (Seite 1661)

SPZ: Springen bei Ergebnis Null (Seite 1663)

SPN: Springen bei Ergebnis nicht Null (Seite 1665)

SPP: Springen bei Ergebnis größer Null (Seite 1666)

SPM: Springen bei Ergebnis kleiner Null (Seite 1668)

SPPZ: Springen bei Ergebnis größer oder gleich Null (Seite 1670)

SPMZ: Springen bei Ergebnis kleiner oder gleich Null (Seite 1671)

SPU: Springen bei Ergebnis ungültig (Seite 1673)

SPL: Sprungliste definieren (Seite 1675)

LOOP: Schleifensprung (Seite 1676)

Setzen der Statusbits bei Wortverknüpfungen

Beschreibung

Wortverknüpfungen beeinflussen die Statusbits A0, A1 und OV im Statuswort. Die Statusbits werden nur dann gesetzt, wenn sie im Programmverlauf verwendet werden.

Wenn im Ergebnis einer Wortverknüpfung alle Bits "0" sind, wird das Statusbit A1 auf "0" zurückgesetzt. Wenn mindestens ein Bit im Ergebnis "1" ist, wird das Statusbit A1 auf "1" gesetzt.

Die folgende Tabelle zeigt das Setzen der Statusbits bei Wortverknüpfungen:

Ergebnis	A1	A0	OV
Null	0	0	0
Nicht Null	1	0	0

Siehe auch

UW: Wortweise nach UND verknüpfen (Seite 1691)

OW: Wortweise nach ODER verknüpfen (Seite 1693)

XOW: Wortweise nach EXKLUSIV ODER verknüpfen (Seite 1695)

UD: Doppelwortweise nach UND verknüpfen (Seite 1697)

OD: Doppelwortweise nach ODER verknüpfen (Seite 1699)

XOD: Doppelwortweise nach EXKLUSIV ODER verknüpfen (Seite 1701)

1.6.3 Master Control Relay

1.6.3.1 Das Master Control Relay

Beschreibung der MCR-Funktionalität

Das Master Control Relay (MCR) aktiviert oder deaktiviert bei kontaktbehafteten Steuerungen den Signalfluss in einem Teil der Steuerung.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)"

⚠ WARNUNG
Um das Risiko möglicher Personen- oder Sachschäden auszuschließen, verwenden Sie die MCR-Funktionalität niemals als Ersatz für ein festverdrahtetes, mechanisches Master Control Relay, das als NOT-AUS- oder Sicherheitseinrichtung dient.

Das MCR wirkt auf alle Anweisungen, die einen Wert in den Speicher zurückschreiben. Diese Anweisungen reagieren bei eingeschalteter MCR-Abhängigkeit unabhängig von einer vorangegangenen binären oder digitalen Verknüpfung.

Für die Realisierung des MCR stehen Ihnen die folgenden Anweisungen zur Verfügung:

- MCRA: MCR-Bereich aktivieren
- MCR: MCR-Bereiche öffnen
- MCR: MCR-Bereiche schließen
- MCRD: MCR-Bereich deaktivieren

Mit einem Verknüpfungsergebnis "1" direkt vor dem Öffnen eines MCR-Bereichs wird die MCR-Abhängigkeit für diesen Bereich aktiviert. Wenn das VKE beim Öffnen des Bereichs "0" ist, erfolgt die Bearbeitung in diesem Bereich ohne MCR-Abhängigkeit.

1.6.3.2 Wichtige Hinweise zur Benutzung der MCR-Funktionalität

⚠ VORSICHT
Bei Bausteinen, in denen mit der Anweisung "MCR-Bereich aktivieren" (MCRA) das Master Control Relay aktiviert wurde, ist folgendes zu beachten: • Wenn das MCR abgeschaltet ist, wird in Programmabschnitten zwischen den Anweisungen "MCR-Bereich öffnen" (MCR( ) und "MCR-Bereich schließen" ( )MCR) durch alle Zuweisungen der Wert "0" geschrieben! • Das MCR ist genau dann abgeschaltet, wenn vor einer Anweisung "MCR-Bereich öffnen" (MCR( ) das Verknüpfungsergebnis (VKE) "0" war.

GEFAHR

STOP der CPU oder undefiniertes Laufzeitverhalten!

Der Compiler greift für Adressberechnungen auch schreibend auf Lokaldaten hinter den in VAR_TEMP definierten temporären Variablen zu. Daher setzen folgende Befehlssequenzen die AS in STOP oder führen zu undefiniertem Laufzeitverhalten (siehe nachfolgende Erläuterungen).

Formalparameterzugriffe

- Zugriffe auf Komponenten komplexer FC-Parameter vom Typ STRUCT, PLC-Datentyp (UDT), ARRAY, STRING
- Zugriffe auf Komponenten komplexer FB-Parameter vom Typ STRUCT, PLC-Datentyp (UDT), ARRAY, STRING aus dem Bereich InOut in einem multiinstanzfähigen Baustein.
- Zugriffe auf Parameter eines multiinstanzfähigen FB, wenn ihre Adresse größer als 8180.0 ist.
- Zugriff im multiinstanzfähigen FB auf einen Parameter vom Typ BLOCK_DB schlägt den DB 0 auf. Nachfolgende Datenzugriffe bringen die CPU in STOP. Bei TIMER, COUNTER, BLOCK_FC, BLOCK_FB wird auch immer T 0, Z 0, FC 0 bzw. FB 0 verwendet.

Parameterübergabe

Erfolgt bei Bausteinaufrufen, bei denen Parameter übergeben werden.

Abhilfe

Lösen Sie die genannten Befehle aus der MCR-Abhängigkeit:

1. Deaktivieren Sie das Master Control Relay mit der Anweisung "MCR-Bereich deaktivieren" (MCRD) vor der betreffenden Anweisung bzw. vor dem betreffenden Netzwerk.
2. Aktivieren Sie das Master Control Relay mit der Anweisung "MCR-Bereich aktivieren" (MCRA) nach der betreffenden Anweisung bzw. nach dem betreffenden Netzwerk.

Programmierempfehlungen 2

2.1 Die neuen Funktionen der S7-1200/1500-CPU und Programmierempfehlungen im Überblick

Höhere Performance

Die CPUs der Baureihen S7-1200/1500 bieten Ihnen eine wesentlich höhere Performance, als die CPUs der Baureihen S7-300/400. Bei der Programmierung mit STEP 7 V5.x waren Sie es wahrscheinlich gewohnt, z. B. absolut zu adressieren, um eine höhere Performance der CPU und einen schlankeren Programmcode zu erreichen. Aufgrund der hohen Performance, die die CPUs der Baureihen S7-1200/1500 mit sich bringen, ist das jetzt nicht mehr notwendig. Wenn Sie den Programmcode, den Sie bereits auf einer S7-300/400-CPU erstellt haben, auf einer S7-1200/1500-CPU wieder verwenden möchten, dann ist es wahrscheinlich an der ein oder anderen Stelle notwendig, den Programmcode zu überarbeiten, um die hohe Performance voll ausschöpfen zu können.

In den folgenden Absätzen stellen wir Ihnen einige der neuen Programmiermöglichkeiten der S7-1200/1500-CPUs vor.

Weitere Tipps, wie Sie die Performance steigern können, finden Sie hier: Performance in STEP 7 (TIA Portal) steigern (https://support.industry.siemens.com/cs/ww/de/view/37571372)

Durchgängige Symbolik

Die S7-1500 bietet Ihnen die Möglichkeit, die Symbolik durchgängig im gesamten Projekt zu verwenden. Mit Hilfe der Autovervollständigung werden Sie beim Programmieren mit Symbolen innerhalb der Programmiereditoren kontextabhängig unterstützt. Die Datenelemente, z. B. innerhalb eines Datenbausteins, erhalten in der Deklaration einen symbolischen Namen, aber keine feste Adresse innerhalb des Datenbausteins. Dadurch können Sie beim Zugriff auf diese Datenelemente die hohe Perfomance der S7-1500 voll ausschöpfen. Die absoluten Adressen der Operanden müssen nicht mehr bekannt sein und Zugriffsfehler werden vermieden.

Ihr Programmcode wird durch die Symbolik verständlicher und Sie müssen weniger kommentieren. Bei einer Korrektur der Symbolik werden automatisch alle Verwendungsstellen aktualisiert.

Hinweis

Weitere Informationen zur Verwendung von durchgängiger Symbolik finden Sie:

Wie können Sie symbolisch adressieren?

Symbolisch adressieren (Seite 209)

Optimierter Bausteinzugriff

Bei optimiertem Bausteinzugriff werden die deklarierten Datenelemente im verfügbaren Speicherbereich des Bausteins automatisch so angeordnet, dass seine Kapazität optimal ausgeschöpft wird. Die Daten werden auf eine Weise strukturiert und abgelegt, die für die verwendete CPU optimal ist. Die Ablage bleibt dem System überlassen. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, über den die Variable innerhalb des Bausteins angesprochen werden kann. Dadurch steigern Sie die Performance der CPU. Zugriffsfehler, z. B. aus HMI heraus, sind nicht möglich.

Bausteinzugriffe im Vergleich
Standard < > Optimiert

SIEMENS

	Standard Bausteinzugriff (S7-1200/1500 kompatibel mit S7-300/400)	Optimierter Bausteinzugriff (Nur S7-1200/1500)
Datenverwaltung	Sie können die Variablen sowohl symbolisch (speicheroptimiert) als auch absolut (anwenderdefiniert) adressieren.	Das System übernimmt die Verwaltung und Optimierung der Datenablage. Somit wird die Speicherkapazität optimal ausgeschöpft.
Performance	Der Zugriff auf eine CPU der Baureihen S7-1200/1500 erfolgt nicht immer schnellstmöglich, da die Datenablage aufgrund der absoluten Adressierung ineffektiv sein kann.	Der Zugriff erfolgt immer schnellstmöglich, da die Datenablage vom System optimiert wird und keine festen Adressen vergeben werden.
Fehleranfälligkeit	Absolute Adressierung (z.B. aus HMI oder bei indirekter Adressierung) kann nach einer Änderung der festen Adresse zu Inkonsistenzen führen.	Zugriffsfehler, z.B. bei indirekter Adressierung oder aus HMI heraus, sind nicht möglich, da symbolisch zugegriffen wird.
Remanenz	Gültig für alle Variablen eines Datenbausteins	Gültig für einzelne Variablen

Empfehlung:
Um eine bestmögliche Performance zu erzielen, ist es empfehlenswert, die Bausteinzugriffsarten innerhalb Ihres Programms nicht zu mischen.

Hinweis

Weitere Informationen zu Bausteinen mit optimiertem Zugriff finden Sie hier:

Bausteine mit optimiertem Zugriff (Seite 55)

Neue Datentypen

Die neuen Datentypen LWORD, LINT, ULINT, LTIME, LTOD, LDT und das ARRAY (32-Bit-Grenze) bieten Ihnen eine wesentlich höhere Berechnungsgenauigkeit bei der Verwendung von mathematischen Funktionen. Im Bereich der impliziten und expliziten Datentypkonvertierung stehen Ihnen im Vergleich zu den CPUs der Baureihen S7-300/400 mehr Möglichkeiten zur Verfügung.

Hinweis

Weitere Informationen zu den neuen Datentypen finden Sie hier:

Übersicht über die gültigen Datentypen (Seite 243)

PLC-Datentypen

PLC-Datentypen (UDT) sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen, die verschiedene Datentypen aufweisen können. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest.

Den PLC-Datentyp können Sie als Basis-Datentyp für die Definition von Variablen und als Vorlage für die Erstellung von globalen Datenbausteinen verwenden. Wenn Sie nachträglich Änderungen am PLC-Datentyp vornehmen, werden diese Änderungen automatisch an allen Verwendungsstellen nachgezogen.

Sie können auch symbolisch auf einzelne Elemente eines ARRAYs innerhalb eines PLC-Datentyps zugreifen.

Hinweis

Weitere Informationen zu den PLC-Datentypen finden Sie hier:

Wie sieht ein PLC-Datentyp aus und wie können Sie ihn in Ihrem Programm deklarieren?

- PLC-Datentypen (UDT) (Seite 286)

Programmierempfehlungen zum PLC-Datentyp finden Sie hier:

- PLC-Datentypen (UDT) verwenden (Seite 231)
- Datentyp DB_ANY verwenden (Seite 215)

Einheitliche Anweisungen in allen Programmiersprachen

Es steht Ihnen ein einheitlicher Satz von Anweisungen in allen Programmiersprachen (KOP, FUP, AWL, SCL und GRAPH) zur Verfügung.

Slice-Zugriffe

Mit Hilfe der Slice-Zugriffe haben Sie die Möglichkeit, gezielt Bereiche innerhalb deklarierter Variablen zu adressieren. Den Zugriff auf ein einzelnes Bit können Sie bis zur Ebene der Variable symbolisch realisieren. Das einzelne Bit wird dann absolut angesprochen.

Hinweis

Weitere Informationen zu den Slice-Zugriffen finden Sie:

Wie können Sie mit Slice-Zugriffen adressieren?

- Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)
- Grundlagen zum Bausteinzugriff (Seite 55)

Indirekte Adressierung

Die indirekte Adressierung bietet die Möglichkeit, Operanden zu adressieren, deren Adresse erst zur Laufzeit berechnet wird. Innerhalb aller Programmiersprachen stehen Ihnen allgemeine Möglichkeiten, wie z. B. die indirekte Adressierung über POINTER, zur Verfügung. In den Programmiersprachen AWL und SCL können Sie dazu auch die Anweisungen PEEK und POKE verwenden.

Hinweis

Weitere Informationen zur indirekten Adressierung finden Sie hier:

Indirekte Adressierung (Seite 119)

Bibliothek "Sample Library for Instructions"

Als Unterstützung bei der Programmierung können Sie die "Sample Library for Instructions" nutzen. Hier finden Sie praxisbezogene und leicht verständliche Programmierbeispiele für SIMATIC S7-1500 und S7-1200 in der Programmiersprache KOP. Sie können die Beispiele einfach in Ihr Programm einfügen und dort weiterverwenden.

Hinweis

Weitere Informationen zu den Programmbeispielen finden Sie hier:

Sample Library for Instructions (Seite 615)

Weitere Programmierempfehlungen

Weitere Informationen zu Programmierempfehlungen und einem Programmierleitfaden und -styleguide finden Sie im Siemens Industry Online Support unter:

- FAQs: Programmierempfehlungen (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=67582299&Datakey=47071380&caller=nl)
- Hintergrund und Systembeschreibungen im Programmierleitfaden und -styleguide (https://support.industry.siemens.com/cs/document/81318674/programmierleitfaden-und-programmierstyleguide-f%C3%BCr-s7-1200-und-s7-1500?dti=0&lc=de-WW)

2.2 Symbolisch adressieren

Vorteile der symbolischen Adressierung

Mit Hilfe der durchgängigen Verwendung von aussagekräftigen Symbolen im gesamten Projekt wird der Programmcode verständlicher und leichter lesbar.

Dadurch haben Sie die folgenden Vorteile:

- Sie müssen keine umfassenden Kommentare mehr schreiben.
- Der Datenzugriff ist performanter.
- Es treten keine Fehler beim Datenzugriff auf.
- Sie müssen nicht mehr mit den absoluten Adressen arbeiten.
- Die Zuordnung des Symbols zur Speicheradresse wird von STEP 7 überwacht, d. h. wenn sich der Name oder die Adresse einer Variablen ändert, werden automatisch alle Verwendungsstellen aktualisiert.

Programmierung in STEP 7 V5.x

In STEP 7 V5.x hatten Sie bereits die Möglichkeit, mit Hilfe von sprechenden Namen für Operanden und Bausteine, Ihr Programm aussagekräftiger zu schreiben. Dazu ordneten Sie die symbolischen Operanden den Speicheradressen und Bausteinen in der Symboltabelle zu. Damit sich eine Änderung in der Symbolik auch auf den Programmcode im Programmiereditor ausgewirkt hat, mussten Sie über die Eigenschaft "Operandenvorrang" festlegen, ob das Symbol oder der Absolutwert maßgeblich sein sollte.

Durch die symbolische Adressierung konnte die Übersichtlichkeit des Programms erhöht werden. In einigen Fällen, z. B. bei der Programmierung mit anwenderdefinierten Datentypen (UDT) konnte dies jedoch eine Beeinträchtigung der Performance nach sich ziehen.

Eine Steigerung der Performance erreichten Sie, indem Sie im UDT die Symbolik ignorierten und absolut adressierten. Dazu war es allerdings nötig, die Datenablage zu kennen. Änderungen am UDT wurden nicht automatisch nachgezogen. Mit Hilfe der absoluten Adressierung konnten Sie auch auf Teile einer Variablen zugreifen und diese bearbeiten. Der Nachteil bei einer ausschließlich absoluten Adressierung war allerdings, dass der Programmcode ab einem gewissen Umfang unübersichtlich wurde und Sie zur besseren Orientierung zusätzlich Kommentare einfügen mussten.

Vorgehensweise in STEP 7 TIA Portal

Die S7-1500-CPU bietet Ihnen eine wesentlich höhere Performance als die S7-300/400-CPUs. Um diese hohe Performance voll nutzen zu können, empfehlen wir Ihnen, für alle Bausteine den optimierten Bausteinzugriff zu aktivieren und im Programmcode symbolisch zu adressieren.

Der Programmiereditor unterstützt Sie beim Arbeiten mit Symbolen durch kontextabhängige Eingabehilfen, wie z. B. der Autovervollständigung. Mit ihrer Hilfe können Sie während des Programmierens bequem auf vorhandene Variablen oder Anweisungen zugreifen.

Programmierbeispiel

Das folgende Beispiel zeigt, wie Sie symbolisch auf einzelne Elemente zugreifen können:

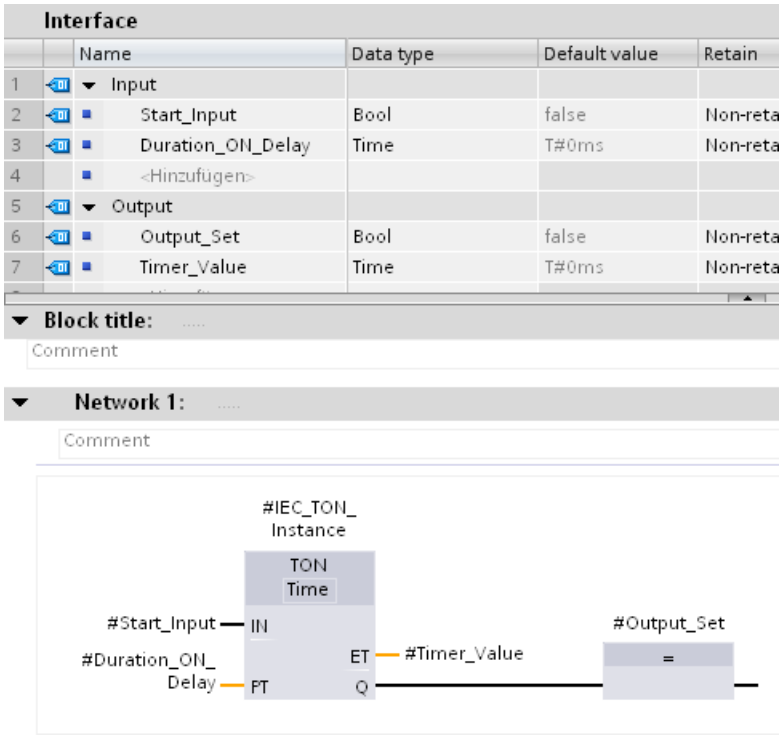

Die Variablennamen, die Sie in der Bausteinschnittstelle definiert haben, können Sie direkt an den Parametern der Anweisung TON verwenden, ohne die absolute Adresse der Variablen zu kennen.

Siehe auch

Bausteinzugriff einstellen (Seite 58)

Grundlagen zum Bausteinzugriff (Seite 55)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Globale Variablen adressieren (Seite 108)

Variablen in globalen Datenbausteinen adressieren (Seite 111)

Autovervollständigung verwenden (Seite 7147)

2.3 ARRAY-Komponenten indirekt adressieren

ARRAY-Zugriffe im TIA Portal mit variablem Index realisieren

Die Verwendung eines ARRAYs empfiehlt sich, wenn Sie zusammenliegende Daten des gleichen Datentyps verarbeiten möchten. Zur Adressierung von ARRAY-Elementen können Sie als Index entweder Konstanten oder Variablen vom Datentyp Ganzzahl angeben. Dabei sind Ganzzahlen mit einer Länge bis 32 Bit zulässig.

Bei der indirekten Adressierung mithilfe einer Variable wird der Index erst zur Programmlaufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden. Sie können auch auf ein ARRAY innerhalb eines PLC-Datentyps (UDT) zugreifen.

Dadurch haben Sie die folgenden Vorteile:

- Es ist keine Adressierung mit Adressregistern bzw. mit selbst aufgebauten Zeigern, wie z. B. einem ANY-Zeiger, notwendig.
- Mehr Flexibilität innerhalb Ihres Programms.
- Der variable Index steht Ihnen in allen STEP 7-Programmiersprachen zur Verfügung.
- Es werden die vorhandenen Namen der Datenbausteine und ARRAY-Variablen verwendet (Symbolische Adressierung). Dadurch erhöht sich die Lesbarkeit des Programmcodes.
- Die Basisadresse des ARRAYs muss nicht bekannt sein.
- Der Programmcode ist einfacher zu erstellen und der Compiler erzeugt optimierten Programmcode.

Vorgehensweise in STEP 7 V5.x

In STEP 7 V5.x mussten Sie zur indirekten Adressierung von ARRAY-Elementen Adressregister mittels selbst aufgebauter POINTER verwenden. Dabei war Folgendes zu berücksichtigen:

- Der Name des ARRAYs wurde nicht verwendet. Dadurch sank die Lesbarkeit des Programmcodes und machte eine zusätzliche Kommentierung erforderlich.
- Die Basisadresse des ARRAYs musste zur Adressierung bekannt sein.

Die Programmiersprache SCL unterstützte bereits die indirekte Adressierung mittels variablem Index.

Programmierbeispiel in STEP 7 V5.x

Für das folgende AWL-Beispiel wird als Voraussetzung der Datenbaustein "Data_classic" benötigt. Um ein Element des ARRAYs "Quantities" anzusprechen, müssen folgende Befehle verwendet werden:

AWL	Erläuterung
`AUF "Data_classic"`	`// Der Datenbaustein "Data_classic" wird aufgerufen.`
`L #index`	`// Der Wert der lokalen Variable #index wird in den Akkumulator 1 geladen.`

AWL	Erläuterung
SLD 3	// Bits 0 bis 31 des Akkumulators 1 um 3 Stellen nach links schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
LAR1	// Adressregister 1 mit dem Inhalt des Akkumulators 1 laden.
L DBW [AR1, P#10.0]	// Das mittels #index adressierte ARRAY-Element in den Akkumulator 1 laden.
	// P#10.0 = Basisadresse des Feldes

Programmierbeispiel in STEP 7 TIA Portal

Im folgenden Beispiel sehen Sie das indirekte Adressieren eines ARRAY-Elements in AWL im TIA Portal.

Legen Sie dazu einen globalen Datenbaustein an:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_Quantities" an.
4. Wählen Sie als Typ des Datenbausteins "ARRAY-DB" aus.
5. Wählen Sie als ARRAY-Datentyp den Datentyp "DINT" aus.
6. Geben Sie als obere ARRAY-Grenze "10" an.
7. Klicken Sie auf "OK".

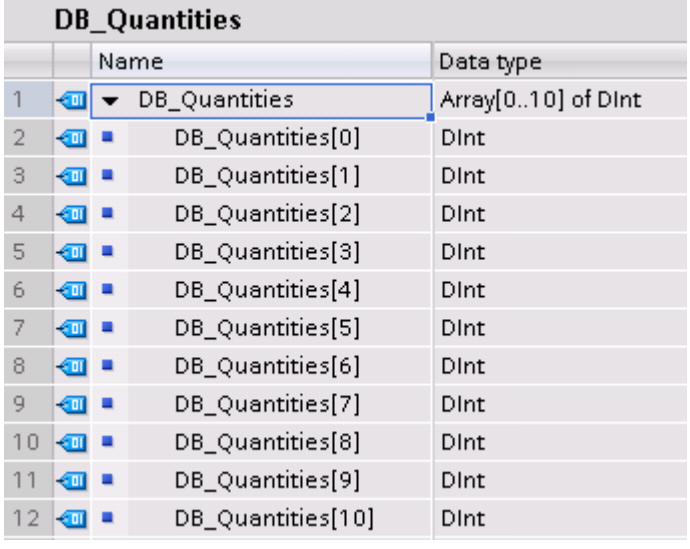

DB_Quantities

	Name	Data type
1	DB_Quantities	Array[0..10] of DInt
2	DB_Quantities[0]	DInt
3	DB_Quantities[1]	DInt
4	DB_Quantities[2]	DInt
5	DB_Quantities[3]	DInt
6	DB_Quantities[4]	DInt
7	DB_Quantities[5]	DInt
8	DB_Quantities[6]	DInt
9	DB_Quantities[7]	DInt
10	DB_Quantities[8]	DInt
11	DB_Quantities[9]	DInt
12	DB_Quantities[10]	DInt

1. Legen Sie einen Funktionsbaustein an und benennen ihn "FB_Quantities".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FB_Quantities

		Name	Data type
1		Input	
2		Output	
3		InOut	
4		Static	
5		Temp	
6		Index	Int
7		Constant	

3. Schreiben Sie folgenden Programmcode:

Network 1: .....

Comment

```
1
2        L     "DB_Quantities".THIS[#index]
3
```

Für die Adressierung eines ARRAY-Elements benötigen Sie im TIA Portal nur noch eine Programmzeile. Der Wert des ARRAY-Elements #index wird direkt aus dem Datenbaustein in den Akkumulator 1 geladen.

4. Rufen Sie den Funktionsbaustein "FB_Quantities" im OB1 auf und vergeben Sie einen Index zwischen 0 - 10:

Network 1: .....

Comment

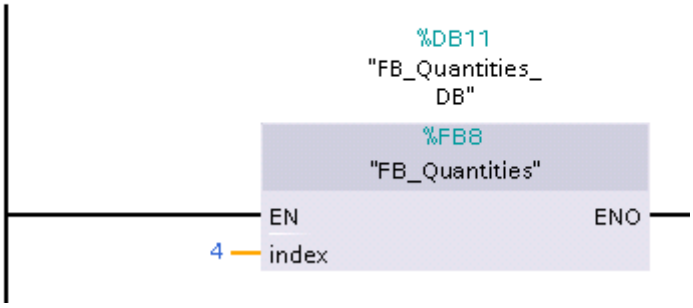

Um eine bestmögliche Performance zu erzielen, beachten Sie folgende Hinweise:

- Variablen, die als ARRAY-Index genutzt werden, deklarieren Sie als Ganzzahl kleiner gleich 32 Bit.
- Zwischenergebnisse und ARRAY-Indizes legen Sie im temporären Lokaldatenbereich an.

Siehe auch

Indirektes Adressieren von ARRAY-Komponenten (Seite 138)
Grundlagen zum ARRAY (Seite 296)
ARRAY (Seite 296)
Indirekte Adressierung in AWL (Seite 139)
Indirektes Adressieren in SCL (Seite 136)

2.4 Datentyp DB_ANY verwenden

Verwendung des Datentyps DB_ANY (S7-1200/1500)

Der Datentyp DB_ANY dient zur Identifikation eines beliebigen Datenbausteins. Bei den CPUs der Baureihen S7-1200/1500 haben Sie die Möglichkeit, auf einen Datenbaustein zuzugreifen, der bei der Programmierung noch nicht vorliegt. Dazu erstellen Sie in der Bausteinschnittstelle des zugreifenden Bausteins einen Bausteinparameter vom Datentyp DB_ANY. An diesen Parameter wird zur Laufzeit der Datenbausteinname oder eine Variable vom Datentyp DB_ANY, der vorher der Datenbausteinname zugewiesen wurde, übergeben. Mittels folgender Anweisungen können Sie den Inhalt eines Datenbausteins symbolisch verarbeiten:

- VARIANT_TO_DB_ANY: VARIANT in DB_ANY konvertieren
- DB_ANY_TO_VARIANT: DB_ANY in VARIANT konvertieren

Weitere Informationen zu den Anweisungen finden Sie unter "Einfache Anweisungen > AWL/SCL > Umwandler > VARIANT".

Der Vorteil dieser Vorgehensweise: Sie können den Programmcode bereits erstellen, bevor Sie wissen, welcher Datenbaustein verarbeitet wird.

Für dieses Beispiel werden folgende Objekte benötigt:

Call structure of PLC S7-1500

	Call structure	!	Address	Details
1	Main		OB1	
2	FB_PrepareNextOrder, DB_FB_PrepareNextOrder		FB1, DB4	@Main ▸ NW1
3	DB_OrderList		DB2	@FB_PrepareNextOrder ▸ Program code
4	FC_PrepareOrder		FC3	@FB_PrepareNextOrder ▸ Program code
5	FC_PrepareHole		FC1	@FC_PrepareOrder ▸ Program code
6	DB_PunchList		DB3	@FC_PrepareHole ▸ Program code
7	DB_PunchList		DB3	@FC_PrepareHole ▸ Program code
8	DB_PunchList		DB3	@FC_PrepareHole ▸ Program code
9	DB_PunchList		DB3	@FC_PrepareHole ▸ Program code
10	FC_PrepareWindowFlexible		FC5	@FC_PrepareOrder ▸ Program code
11	DB_PunchList		DB3	@FC_PrepareWindowFlexible ▸ Program code
12	DB_PunchList		DB3	@FC_PrepareWindowFlexible ▸ Program code
13	DB_PunchList		DB3	@FC_PrepareWindowFlexible ▸ Program code
14	DB_PunchList		DB3	@FC_PrepareWindowFlexible ▸ Program code
15	FC_PrepareWindowStatic		FC2	@FC_PrepareOrder ▸ Program code
16	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
17	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
18	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
19	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
20	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
21	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
22	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
23	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
24	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
25	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
26	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
27	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
28	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
29	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
30	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
31	DB_PunchList		DB3	@FC_PrepareWindowStatic ▸ Program code
32	UDT_Hole		UDT1	@FC_PrepareOrder ▸ Program code
33	UDT_RectangleWindowFlexible		UDT3	@FC_PrepareOrder ▸ Program code
34	UDT_RectangleWindowStatic		UDT2	@FC_PrepareOrder ▸ Program code
35	FC_SwapOrderList		FC4	@Main ▸ NW2
36	DB_OrderList		DB2	@FC_SwapOrderList ▸ Program code
37	FB_PrepareNextOrder, #OrderList		FB1	Block interface
38	DB_OrderList		DB2	@FB_PrepareNextOrder ▸ Program code
39	FC_PrepareOrder		FC3	@FB_PrepareNextOrder ▸ Program code
40	MOVE_BLK_VARIANT		FC900	@FC_SwapOrderList ▸ Program code
41	OrderRepository		DB6	@Main ▸ NW2
42	OrderRepository		DB6	@Main ▸ NW2
43	DB_OrderHole (Data block derived from UDT_Hole)		DB1	

Programmierbeispiel

Das folgende Beispiel zeigt, wie Sie den Datentyp DB_ANY einsetzen können:

Eine Stanze kann unterschiedliche geometrische Figuren ausstanzen. Die Stanzaufträge werden an die Maschine übergeben und zu jedem einzelnen Auftrag gibt es spezifische Auftragsdaten. Die Auftragsdaten unterscheiden sich sowohl in ihrem Auftragstyp als auch in ihren Werten.

Vorgehensweise - Erstellung der PLC-Datentypen

Im ersten Auftrag soll ein rundes Loch aus einem Blech gestanzt werden. Damit die Stanze diesen Auftrag ausführen kann, werden die Mittelpunkts-Koordinate und der Radius des Loches benötigt. Diese Auftragsdaten können Sie gesammelt in einem PLC-Datentyp (UDT) an die Stanze übergeben.

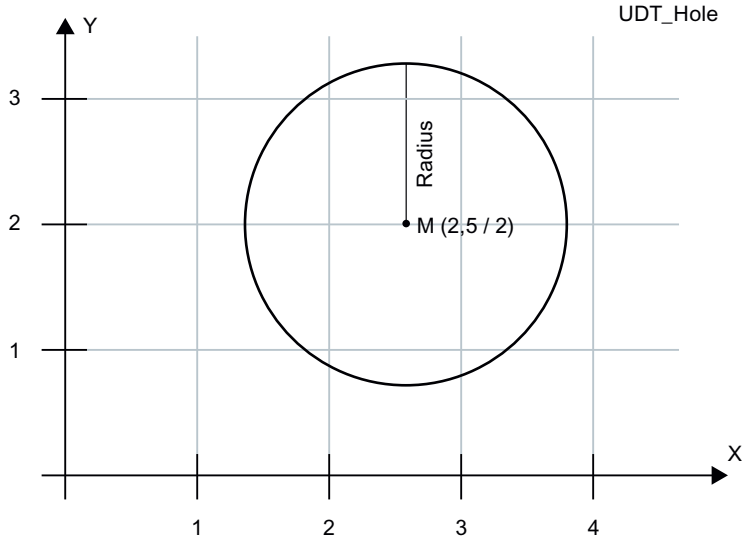

Legen Sie den PLC-Datentyp "UDT_Hole" an, um die Auftragsdaten zu übermitteln:

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   X-coordinate > REAL
   Y-coordinate > REAL
   Diameter > REAL

UDT_Hole

	Name	Data type
1	X-coordinate	Real
2	Y-coordinate	Real
3	Diameter	Real

Im zweiten Auftrag soll ein Rechteck aus einem Blech gestanzt werden. Für diesen Auftrag werden zwei Koordinaten benötigt, die jeweils den linken oberen Punkt und den rechten unteren Punkt des Rechtecks markieren. Diese Auftragsdaten können Sie gesammelt im PLC-Datentyp "UDT_RectangleWindowStatic" an die Stanze übergeben.

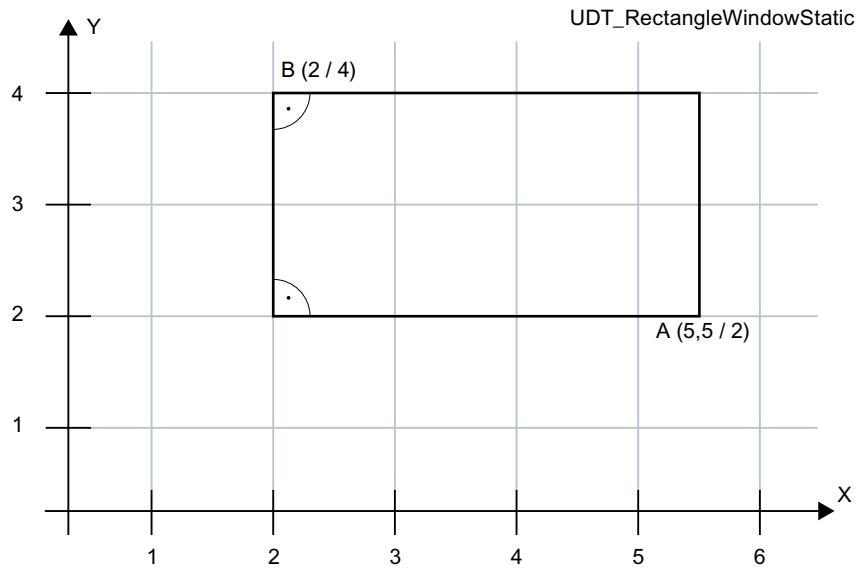

Legen Sie den PLC-Datentyp "UDT_RectangleWindowStatic" an:

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   X1-coordinate > REAL
   Y1-coordinate > REAL
   X2-coordinate > REAL
   Y2-coordinate > REAL

UDT_RectangleWindowStatic			
		Name	Data type
1		X1-coordinate	Real
2		Y1-coordinate	Real
3		X2-coordinate	Real
4		Y2-coordinate	Real

Mit den Auftragsdaten des "UDT_RectangleWindowStatic" können Sie nur Rechtecke ausstanzen, deren Kanten parallel zur x- und y-Achse verlaufen.

Wenn Sie ein Rechteck mit beliebiger Ausrichtung, d. h. nicht parallel zur x- und y-Achse, ausstanzen möchten, dann brauchen Sie einen weiteren PLC-Datentyp. In diesem können Sie z. B. die Höhe und Breite, sowie die Ausrichtung des Rechtecks zur x-Achse durch einen Winkel angeben.

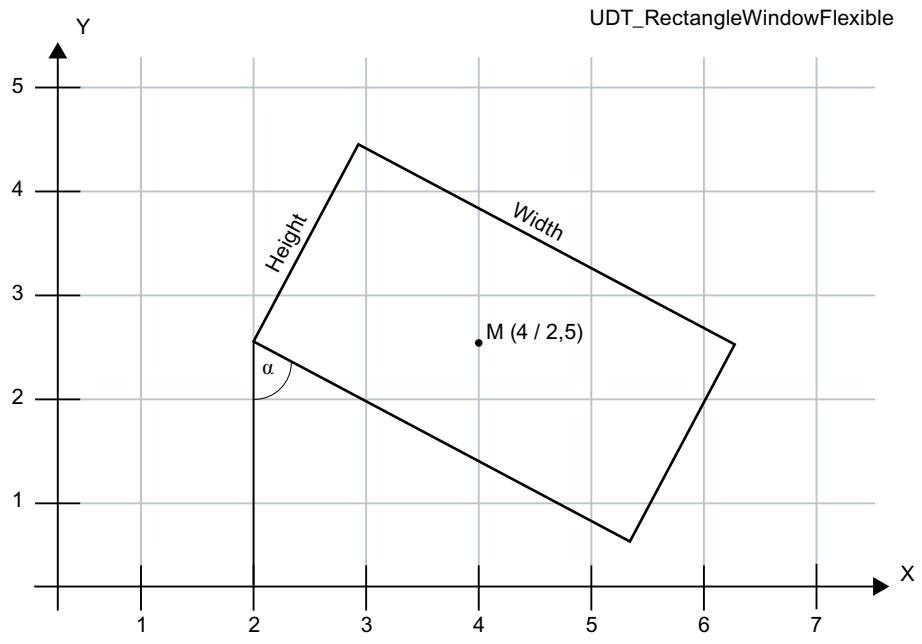

Legen Sie den PLC-Datentyp "UDT_RectangleWindowFlexible" an:

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   X-coordinate > REAL
   Y-coordinate > REAL
   Height > REAL
   Width > REAL
   Angle > REAL

UDT_RectangleWindowFlexible

		Name	Data type
1		X-coordinate	Real
2		Y-coordinate	Real
3		Height	Real
4		Width	Real
5		Angle	Real

Die x-y-Koordinaten geben den Mittelpunkt des Rechtecks an.

Vorgehensweise - Erstellung der Datenbausteine

Im nächsten Abschnitt sehen Sie, wie Sie die einfachen, geometrischen Figuren, deren Auftragsdaten Sie in den PLC-Datentypen erfasst haben, an die Stanze übertragen können. Die Stanzaufträge werden im Programmcode in einzelne Stanzschläge zerlegt, die dann der Reihe nach durch die Stanze ausgeführt werden. Die Stanze hat einen Kreuztisch, auf dem ein Blech fest eingespannt ist. Einen Kreuztisch können Sie wie in einem Koordinatensystem entlang der x- und/oder y-Achse verschieben. Der Kreuztisch kann über zwei Motoren bewegt werden. Das Werkzeug hat verschiedene Einsätze um aus dem Blech etwas heraus stanzen zu können, wie z. B. Kreise und Rechtecke verschiedener Größen. Das Werkzeug kann zusätzlich um bis zu 90° gedreht werden, damit nicht nur Rechtecke einer Ausrichtung möglich sind.

Die PLC-Datentypen werden nun genutzt um mehrere Instanz-Datenbausteine zu erzeugen. Der Instanz-Datenbaustein enthält dann die konkreten Werte, z. B. für ein Loch.

Legen Sie den Instanz-Datenbaustein "DB_OrderHole" an:

1. Doppelklicken Sie im Ordner "Programmbausteine" auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
1. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
2. Geben Sie den Namen "DB_OrderHole" an.
3. Wählen Sie als Typ des Datenbausteins "UDT_Hole" aus.
4. Klicken Sie auf "OK".

Geben Sie die entsprechenden Startwerte an:

DB_OrderHole

		Name	Data type	Start value
1		Static		
2		X-coordinate	Real	10.0
3		Y-coordinate	Real	900.0
4		Diameter	Real	4.0

Um nun ein bestimmtes Blechteil, z. B. eine Schaltschrankseitenwand, zu fertigen, werden die dafür nötigen geometrischen Figuren in die Stanze geladen. Dazu legen Sie einen weiteren Datenbaustein an, in dem sich eine Liste von Datenbausteinen befindet.

Legen Sie den Datenbaustein "DB_OrderList" an:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_OrderList" an.
4. Wählen Sie als Typ des Datenbausteins "Global-DB" aus.
5. Klicken Sie auf "OK".

Legen Sie innerhalb des Datenbausteins die folgende Auftragsliste an:

DB_OrderList

	Name	Data type	Start value	Comment
1	Static			
2	Order	Array[1..100] of DB_ANY		
3	Order[1]	DB_ANY	0	Supply
4	Order[2]	DB_ANY	0	Display
5	Order[3]	DB_ANY	0	Anchoring1
6	Order[4]	DB_ANY	0	Anchoring2

Vorgehensweise - Erstellung des Programmcodes

Für jeden Auftragstyp wird jeweils eine eigene Funktion erstellt. Die Stanzaufträge werden hier in einzelne Schläge zerlegt und in einem ARRAY gesammelt.

1. Legen Sie den PLC-Datentyp "UDT_Punch" an.
2. Doppelklicken Sie auf den Befehl "Neuen Datentyp hinzufügen" unter "PLC-Datentypen". Es wird ein neuer PLC-Datentyp mit dem Namen "Anwenderdatentyp_x" angelegt.
3. Benennen Sie den PLC-Datentyp um in "UDT_Punch".
4. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   Tool > DINT
   x > REAL
   y > REAL
   w > REAL

UDT_Punch

	Name	Data type
1	Tool	DInt
2	x	Real
3	y	Real
4	w	Real

5. Legen Sie den ARRAY-Datenbaustein "DB_PunchList" an:

DB_PunchList

	Name	Data type
1	DB_PunchList	Array[0..100] of "UDT_Punch"
2	DB_PunchList[0]	"UDT_Punch"
3	DB_PunchList[1]	"UDT_Punch"
4	DB_PunchList[2]	"UDT_Punch"
5	DB_PunchList[3]	"UDT_Punch"
6	DB_PunchList[4]	"UDT_Punch"
7	DB_PunchList[5]	"UDT_Punch"
8	DB_PunchList[6]	"UDT_Punch"
9	DB_PunchList[7]	"UDT_Punch"
10	DB_PunchList[8]	"UDT_Punch"

Um den Stanzauftrag für ein Loch vorzubereiten und in einzelne Schläge aufzuteilen, legen Sie eine SCL-Funktion an und nennen Sie sie "FC_PrepareHole".

1. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_PrepareHole		
	Name	Data type
1	Input	
2	a	"UDT_Hole"
3	Output	
4	InOut	
5	Temp	
6	Constant	
7	Return	
8	FC_PrepareHole	Void

2. Schreiben Sie folgenden Programmcode:

```
1  IF #a.Diameter >= 4.0 THEN
2      "DB_PunchList"."THIS"[0].Tool := 1;
3  END_IF;
4
5  IF #a.Diameter >= 6.0 THEN
6      "DB_PunchList"."THIS"[0].Tool := 2;
7  END_IF;
8
9   "DB_PunchList"."THIS"[0].x := #a."X-coordinate";
10  "DB_PunchList"."THIS"[0].y := #a."Y-coordinate";
```

Um den Stanzauftrag für ein Fenster vorzubereiten, benötigen Sie eine Funktion, die den Stanzauftrag aus vier Folgen von Schlägen zusammensetzt. Legen Sie eine SCL-Funktion an und benennen Sie sie "FC_PrepareWindowStatic".

1. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_PrepareWindowStatic		
	Name	Data type
1	Input	
2	a	"UDT_RectangleWindowStatic"
3	Output	
4	InOut	
5	Temp	
6	delta	Real
7	x	Real
8	y	Real
9	i	DInt
10	start	DInt
11	punches_x	DInt
12	punches_y	DInt
13	Constant	
14	Return	
15	FC_PrepareWindowStatic	Void

2. Schreiben Sie folgenden Programmcode:

```
1  #delta := 6.0; // defined by tool 7
2  #i := 0;
3
4  #punches_x := REAL_TO_DINT(( #a.X2coordinate - #a.X1coordinate ) / #delta );
5  #punches_y := REAL_TO_DINT(( #a.Y2coordinate - #a.Y1coordinate ) / #delta );
6  #x := #a.X1coordinate + #delta / 2.0;
7  #y := #a.Y1coordinate + #delta / 2.0;
8
9  #start := 0;
10 FOR #i := 0 TO #punches_x-1 DO
11     "DB_PunchList"."THIS"[ #i + #start ].Tool := 7;
12     "DB_PunchList"."THIS"[ #i + #start ].x := #x + DINT_TO_REAL( #i )* #delta;
13     "DB_PunchList"."THIS"[ #i + #start ].y := #y;
14     "DB_PunchList"."THIS"[ #i + #start ].a := 0; // don't rotate
15 END_FOR;
16
17 #start += #punches_x;
18 FOR #i := 0 TO #punches_y-1 DO
19     "DB_PunchList"."THIS"[ #i + #start ].Tool := 7;
20     "DB_PunchList"."THIS"[ #i + #start ].x := #x + DINT_TO_REAL( #punches_x )* #delta;
21     "DB_PunchList"."THIS"[ #i + #start ].y := #y + DINT_TO_REAL( #i ) * #delta;
22     "DB_PunchList"."THIS"[ #i + #start ].a := 0; // don't rotate
23 END_FOR;
24
25 #start += #punches_y;
26 FOR #i := 0 TO #punches_x-1 DO
27     "DB_PunchList"."THIS"[ #i + #start ].Tool := 7;
28     "DB_PunchList"."THIS"[ #i + #start ].x := #x + DINT_TO_REAL( #punches_x - #i ) * #delta;
29     "DB_PunchList"."THIS"[ #i + #start ].y := #y + DINT_TO_REAL( #punches_y ) * #delta;
30     "DB_PunchList"."THIS"[ #i + #start ].a := 0; // don't rotate
31 END_FOR;
32
33 #start += #punches_x;
34 FOR #i := 0 TO #punches_y-1 DO
35     "DB_PunchList"."THIS"[ #i + #start ].Tool := 7;
36     "DB_PunchList"."THIS"[ #i + #start ].x := #x;
37     "DB_PunchList"."THIS"[ #i + #start ].y := #y + DINT_TO_REAL( #punches_y - #i) * #delta;
38     "DB_PunchList"."THIS"[ #i + #start ].a := 0; // don't rotate
39 END_FOR;
```

Um den Stanzauftrag für ein variables Rechteck vorzubereiten, benötigen Sie eine Funktion, die den Stanzauftrag aus vier Folgen von Schlägen zusammensetzt. Legen Sie eine SCL-Funktion an und benennen Sie sie "FC_PrepareWindowFlexible".

1. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_PrepareWindowFlexible		
	Name	Data type
1	Input	
2	a	"UDT_RectangleWindowFlexible"
3	Output	
4	InOut	
5	Temp	
6	delta	Real
7	delta_x	Real
8	delta_y	Real
9	x	Real
10	y	Real
11	i	DInt
12	start	DInt
13	punches_x	DInt
14	punches_y	DInt
15	Constant	
16	Return	
17	FC_PrepareWindowFlexible	Void

2. Schreiben Sie folgenden Programmcode:

```
1  #delta := 6.0; // defined by tool 7
2  #i := 0;
3
4  #punches_x := REAL_TO_DINT(( #a.Width ) / #delta );
5  #punches_y := REAL_TO_DINT(( #a.Height ) / #delta );
6
7  // delta_x := ...
8  // delta_y := ...
9  // #x :=
10 // #y :=
11
12 #start := 0;
13 FOR #i := 0 TO #punches_x-1 DO
14     "DB_PunchList"."THIS"[ #i + #start ].Tool := 7;
15     "DB_PunchList"."THIS"[ #i + #start ].x := #x + DINT_TO_REAL( #i )* #delta_x;
16     "DB_PunchList"."THIS"[ #i + #start ].y := #y + DINT_TO_REAL( #i )* #delta_y;
17     "DB_PunchList"."THIS"[#i + #start].a := #a.Angle;
18 END_FOR;
19
20 // ....
```

Nun soll die Stanze beginnen, die Aufträge abzuarbeiten. Wenn sie sich bereits in der Abarbeitung der Aufträge befindet, soll sie sich den nächsten Auftrag aus der Auftragsliste holen und diesen vorbereiten.

1. Legen Sie einen SCL-Funktionsbaustein an.
2. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "Funktionsbaustein (FB)".
4. Geben Sie den Namen "FB_PrepareNextOrder" an.
5. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FB_PrepareNextOrder

	Name	Data type
1	Input	
2	Output	
3	InOut	
4	Static	
5	OrderReady	Bool
6	OrderNo	UDInt
7	Order	DB_ANY
8	OrderCount	UDInt
9	Temp	
10	Error	Int

6. Deklarieren Sie die globale Variable "AllOrdersDone":

AllOrdersDone	Bool	%M0.0

7. Schreiben Sie folgenden Programmcode:

```
1  IF #OrderReady AND NOT "AllOrdersDone" THEN
2      IF #OrderNo < #OrderCount THEN
3          #Order := "DB_OrderList".Order[#OrderNo];
4          #OrderNo := #OrderNo + 1;
5          #OrderReady := FALSE;
6          "FC_PrepareOrder"( Order := #Order , Error => #Error );
7      ELSE;
8          "AllOrdersDone" := TRUE;
9      END_IF;
10 END_IF;
```

Der nächste Auftrag der Liste wird vorbereitet, indem der aktuelle Stanzauftrag in einzelne Stanzschläge zerlegt wird. Dazu muss die Stanze erkennen können, welcher Stanzauftrag vorliegt.

1. Legen Sie eine SCL-Funktion an.
2. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "Funktion (FC)".
4. Geben Sie den Namen "FC_PrepareOrder" an.

5. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_PrepareOrder		
	Name	Data type
1	Input	
2	Order	DB_ANY
3	Output	
4	Error	Int
5	InOut	
6	Temp	
7	WindowFlexible	"UDT_RectangleWindowFlexible"
8	WindowStatic	"UDT_RectangleWindowStatic"
9	Hole	"UDT_Hole"
10	a	Variant
11	Constant	
12	Return	
13	FC_PrepareOrder	Void

6. Schreiben Sie folgenden Programmcode:

```
1  #a := DB_ANY_TO_VARIANT(in := #Order, err => #Error);
2
3  IF TypeOf(#a) = UDT_Hole THEN
4      "FC_PrepareHole"(#Hole);
5  END_IF;
6
7  IF TypeOf(#a) = UDT_RectangleWindowStatic THEN
8      "FC_PrepareWindowStatic"(#WindowStatic);
9  END_IF;
10
11 IF TypeOf(#a) = UDT_RectangleWindowFlexible THEN
12     "FC_PrepareWindowFlexible"(#WindowFlexible);
13 END_IF;
```

Rufen Sie im SCL-Funktionsbaustein "FB_PrepareNextOrder" die SCL-Funktion "FC_PrepareOrder" auf:

```
1  IF #OrderReady AND NOT "AllOrdersDone" THEN
2      IF #OrderNo < #OrderCount THEN
3          #Order := "DB_OrderList".Order[#OrderNo];
4          #OrderNo := #OrderNo + 1;
5          #OrderReady := FALSE;
6          "FC_PrepareOrder"( Order := #Order , Error => #Error );
7      ELSE;
8          "AllOrdersDone" := TRUE;
9      END_IF;
10 END_IF;
```

Rufen Sie anschließend den "FB_PrepareNextOrder" im OB1 auf:

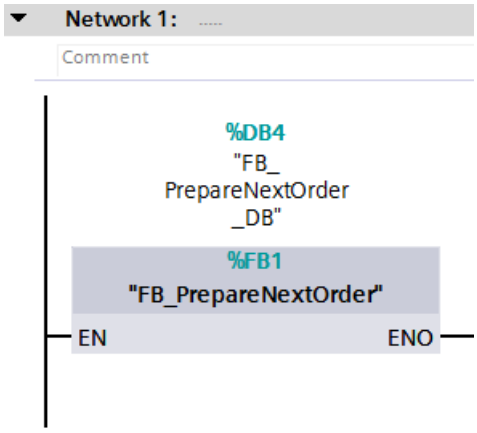

Neben der Möglichkeit wie oben beschrieben, die Aufträge mit Hilfe des Datenbausteins "DB_OrderList" in einer fest vorgegebenen Reihenfolge in die Stanze zu laden, könnte die Auftragsliste zum Beispiel auch dynamisch erzeugt werden. Oder, dass es zum Beispiel eine Auswahl an unterschiedlichen Auftragslisten gibt. Dann haben Sie die Möglichkeit, nachdem eine Auftragsliste abgearbeitet wurde, das eine neue Auftragsliste in die Stanze geladen wird. Für diese Funktionsweise brauchen Sie folgenden zusätzlichen Programmcode.

Legen Sie einen Instanz-Datenbaustein des Funktionsbausteins "FB_PrepareNextOrder" an:

1. Doppelklicken Sie im Ordner "Programmbausteine" auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_FB_PrepareNextOrder" an.
4. Wählen Sie als Typ des Datenbausteins "FB_PrepareNextOrder" aus.
5. Klicken Sie auf "OK".

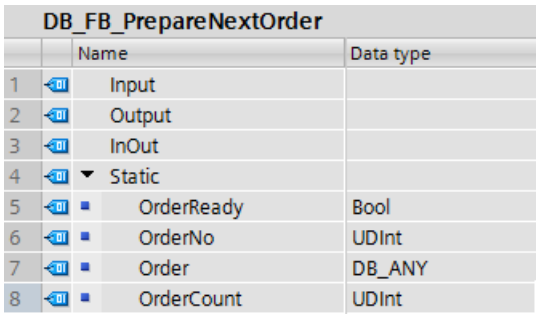

DB_FB_PrepareNextOrder

	Name	Data type
1	Input	
2	Output	
3	InOut	
4	Static	
5	OrderReady	Bool
6	OrderNo	UDInt
7	Order	DB_ANY
8	OrderCount	UDInt

Um die Auftragsliste tauschen zu können, erstellen Sie eine weitere SCL-Funktion:

1. Legen Sie die SCL-Funktion "FC_SwapOrderList" an.
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_SwapOrderList		
	Name	Data type
1	Input	
2	List	Array[*] of DB_ANY
3	ListLength	UDInt
4	Output	
5	InOut	
6	OrderList	"FB_PrepareNextOrder"
7	Temp	
8	Error	Int
9	Constant	
10	Return	
11	FC_SwapOrderList	Void

3. Deklarieren Sie die globale Variable "NewModelNr":

NewModelNr	Int	%MW2

4. Schreiben Sie folgenden Programmcode:

```
1  IF "AllOrdersDone" = TRUE THEN
2      #Error := MOVE_BLK_VARIANT(
3          SRC := #List,
4          COUNT := #ListLength,
5          SRC_INDEX := 0,
6          DEST_INDEX := 0,
7          DEST => "DB_OrderList".Order);
8      IF #Error = 0 THEN
9          #OrderList.OrderNo := 0;
10         #OrderList.OrderReady := FALSE;
11         #OrderList.OrderCount := #ListLength;
12         "AllOrdersDone" := FALSE;
13         "NewModelNr" := 0;
14     END_IF;
15 END_IF;
```

Legen Sie den Datenbaustein "DB_OrderRepository" an:

1. Doppelklicken Sie im Ordner "Programmbausteine" auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_OrderRepository" an.
4. Wählen Sie als Typ des Datenbausteins "Global-DB" aus.
5. Klicken Sie auf "OK".

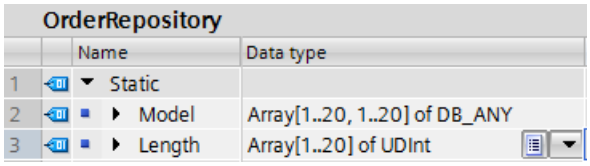

OrderRepository

		Name	Data type
1		Static	
2		Model	Array[1..20, 1..20] of DB_ANY
3		Length	Array[1..20] of UDInt

Rufen Sie anschließend den "FC_SwapOrderList" im OB1 auf:

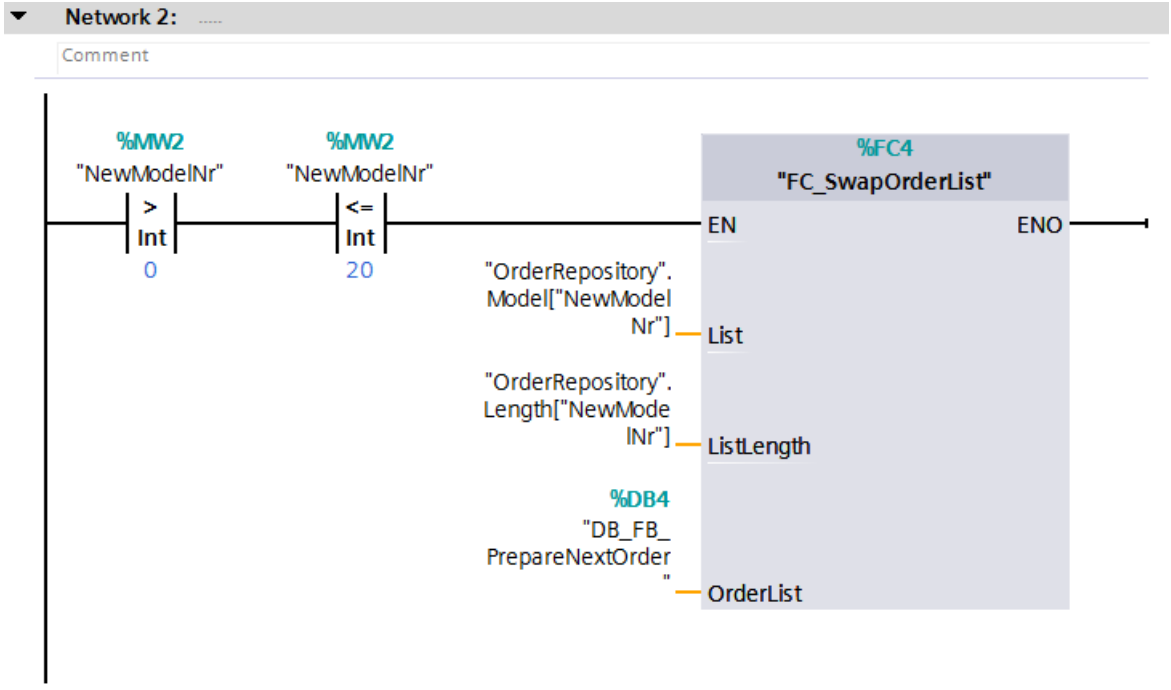

Ergebnis

Dieses Beispiel zeigt Ihnen, wie Sie mit Hilfe der Anweisung "DB_ANY_TO_VARIANT: DB_ANY in VARIANT konvertieren" den PLC-Datentyp eines Datenbausteins bestimmen und wie Sie auf dieser Basis eine geeignete Funktion auswählen und ausführen.

Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 111)

2.5 PLC-Datentypen (UDT) verwenden

Verwendung eines PLC-Datentyps

PLC-Datentypen (UDT) sind Datenstrukturen, die Sie selbst definieren und im Programm mehrmals verwenden können. Die Struktur kann sich aus mehreren Elementen unterschiedlicher Datentypen zusammen setzen. Die Datentypen der einzelnen Elemente legen Sie bei der Deklaration eines PLC-Datentyps fest.

PLC-Datentypen werden häufig verwendet, wenn ein zusammenhängender Datensatz mit unterschiedlichen Datentypen benötigt wird und dieser von verschiedenen Programmstellen aus bearbeitet werden soll. Das können z. B. sein:

- Datensätze zur Materialverfolgung
- Parametersätze für eine Motoreinstellung
- Rezepte

Die Verwendung von PLC-Datentypen bringt Ihnen die folgenden Vorteile:

- Die Elemente eines PLC-Datentyps können auch indirekt adressiert werden, d. h. die Adresse ist variabel und wird erst zur Laufzeit berechnet.
- Variablen, die auf einem PLC-Datentyp basieren, erben alle Eigenschaften des PLC-Datentyps. Damit werden bei einer Änderung des PLC-Datentyps automatisch alle Variablen angepasst, die diesen PLC-Datentyp zur Basis haben.
- Mithilfe der durchgängigen Symbolik wird das Programm leichter lesbar, da die Namen der einzelnen Elemente eines PLC-Datentyps im Programm angezeigt werden.
- Eine optimale Ausnutzung der hohen Performance einer S7-1500-CPU.
- Beim Bausteinaufruf kann der PLC-Datentyp als ganze Struktur übergeben werden.
- Eine vereinfachte Aufrufschnittstelle durch eine geringere Anzahl zu versorgender Parameter.

Vorgehensweise in STEP 7 V5.x

In STEP 7 V5.x hatten Sie bereits die Möglichkeit, mit Hilfe des Datentyps STRUCT oder eines PLC-Datentyps (UDT) einen Datensatz als strukturierte Variable anzulegen. Allerdings wurde die Performance durch die Nutzung der symbolischen Adressierung beeinträchtigt.

Die Deklaration in den Datenbausteinen wurde meistens als anonyme Struktur realisiert. Die Bausteine selbst wurden dann so programmiert, dass als Aktualparameter die Werte der Struktur übergeben und die berechneten Werte wieder in die Struktur kopiert wurden. So konnten Sie auch die Datenbausteinnummer übergeben und im Baustein absolut adressieren. Dabei war die Anzahl der Parameter, die Sie versorgen mussten, häufig sehr groß. Die Aktualdaten wurden in den Datenbausteinen abgelegt und die berechneten Werte an weitere Bausteine übergeben. Bei der Übergabe der Datenbausteinvariablen war allerdings keine Symbolik mehr verfügbar.

Programmierbeispiel in STEP 7 TIA Portal

Sie können einem PLC-Datentyp sowohl einen Formal- als auch einen Aktualparameter zuweisen. Dadurch müssen Sie keine einzelnen Parameter mehr deklarieren. Wenn ein Baustein einen Eingangsparameter hat, der auf einem PLC-Datentyp basiert, müssen Sie eine Variable vom gleichen PLC-Datentyp als Aktualparameter übergeben.

Das folgende Beispiel zeigt den Aufruf und die Parametrierung eines Funktionsbausteins (FB) mit zwei Formalparametern:

1. Legen Sie einen PLC-Datentyp an, indem Sie in der Projektnavigation im Ordner "PLC-Datentypen" den Befehl "Neuen Datentyp hinzufügen" doppelklicken.
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Benennen Sie den PLC-Datentyp um in "UDT_Material".
3. Deklarieren Sie die folgenden Zeilen innerhalb des PLC-Datentyps:
   ArticleNumber > Datentyp: DINT
   ArticleName > Datentyp: STRING
   Amount > Datentyp: REAL
   Unit > Datentyp: STRING

UDT_Material

		Name	Data type
1		ArticleNumber	DInt
2		ArticleName	String
3		Amount	Real
4		Unit	String

Verwenden Sie den PLC-Datentyp innerhalb eines globalen Datenbausteins. Sie können den PLC-Datentyp entweder direkt als Datentyp des Datenbausteins angeben oder innerhalb des Datenbausteins als Datentyp einer Variable.

Legen Sie dazu einen globalen Datenbaustein an:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_MaterialBuffer" an.
4. Wählen Sie als Typ des Datenbausteins "ARRAY-DB" aus.
5. Wählen Sie als ARRAY-Datentyp den PLC-Datentyp "UDT_Material" aus.

6. Geben Sie als obere ARRAY-Grenze "1000" an.
7. Klicken Sie auf "OK".

DB_MaterialBuffer

	Name	Data type
1	DB_MaterialBuffer	Array[0..1000] of "UDT_Material"
2	DB_MaterialBuffer[0]	"UDT_Material"
3	ArticleNumber	DInt
4	ArticleName	String
5	Amount	Real
6	Unit	String
7	DB_MaterialBuffer[1]	"UDT_Material"
8	DB_MaterialBuffer[2]	"UDT_Material"
9	DB_MaterialBuffer[3]	"UDT_Material"
10	DB_MaterialBuffer[4]	"UDT_Material"
11	DB_MaterialBuffer[5]	"UDT_Material"

Verschalten Sie beim Funktionsbausteinaufruf die Formalparameter mit Variablen aus dem globalen Datenbaustein "DB_MaterialBuffer".

1. Legen Sie einen SCL-Funktionsbaustein an und benennen ihn "FB_Material".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FB_Material

	Name	Data type
1	Input	
2	Source	Variant
3	SourceIndex	DInt
4	DestinationIndex	DInt
5	Output	
6	InOut	
7	Destination	Variant
8	Static	
9	Temp	
10	Result	Int
11	Constant	

3. Schreiben Sie folgenden Programmcode:

```
1
2 #Result := MOVE_BLK_VARIANT(SRC:=#Source,
3                             COUNT:=3,
4                             SRC_INDEX:=#SourceIndex,
5                             DEST_INDEX:=#DestinationIndex,
6                             DEST=>#Destination);
```

4. Rufen Sie den Funktionsbaustein "FB_Material" im OB1 auf und verschalten Sie die Formalparameter mit Variablen des globalen Datenbausteins "DB_MaterialBuffer":

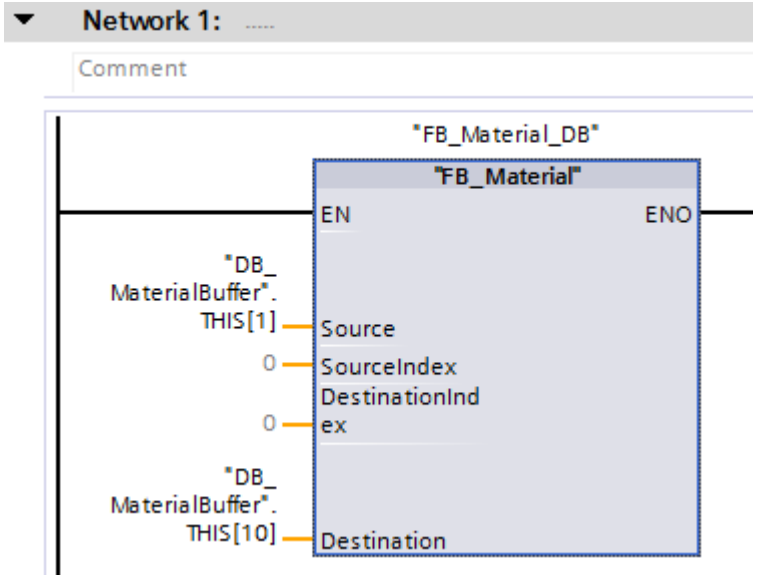

Die Materialdaten werden innerhalb des globalen Datenbausteins "DB_MaterialBuffer" verschoben.

Siehe auch

Wissenswertes zu strukturierten PLC-Variablen (Seite 7387)

Strukturierte PLC-Variablen anlegen (Seite 7389)

Regeln zur Versorgung von Bausteinparametern (Seite 78)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

PLC-Datentypen (UDT) (Seite 286)

2.6 MOVE-Anweisungen in AWL verwenden

Verwendungsmöglichkeiten

Auf einer S7-1500-CPU haben Sie jetzt auch in AWL die Möglichkeit, mit MOVE-Anweisungen zu programmieren.

Dadurch haben Sie die folgenden Vorteile:

- Es ist ein einfacher Programmaufbau möglich.
- Die Performance der CPU steigt.

Programmierung in STEP 7 V5.x

In STEP 7 V5.x haben Sie zur Realisierung der MOVE-Funktionalitäten die Systemfunktionen "BLKMOV: Bereich kopieren" und "UBLKMOV: Bereich ununterbrechbar kopieren" verwendet.

Vorgehensweise in STEP 7 TIA Portal

In STEP 7 TIA Portal stehen Ihnen die folgenden neuen MOVE-Anweisungen zur Verfügung:

- MOVE: Wert kopieren
- MOVE_BLK: Bereich kopieren
- MOVE_BLK_VARIANT: Bereich kopieren
- UMOVE_BLK: Bereich ununterbrechbar kopieren

Programmierbeispiel

Das folgende Beispiel zeigt die Verwendung der Anweisung "MOVE_BLK: Bereich kopieren". Dazu wird ein ARRAY-Bereich in einen anderen ARRAY-Bereich kopiert:

Data_DB

	Name	Data type
1	Static	
2	Array_1	Array [0..10] of Int
3	Array_1[0]	Int
4	Array_1[1]	Int
5	Array_1[2]	Int
6	Array_1[3]	Int
7	Array_1[4]	Int
8	Array_1[5]	Int
9	Array_1[6]	Int
10	Array_1[7]	Int
11	Array_1[8]	Int
12	Array_1[9]	Int
13	Array_1[10]	Int
14	Array_2	Array [0..10] of Int
15	Array_2[0]	Int
16	Array_2[1]	Int
17	Array_2[2]	Int
18	Array_2[3]	Int
19	Array_2[4]	Int
20	Array_2[5]	Int
21	Array_2[6]	Int
22	Array_2[7]	Int
23	Array_2[8]	Int
24	Array_2[9]	Int
25	Array_2[10]	Int

Network 1:

Comment

```
1    CALL  MOVE_BLK
2        Int  UInt
3        IN    :="Data_DB".Array_1[0]
4        COUNT :=10
5        OUT   :="Data_DB".Array_2[0]
6
7
```

Mit Hilfe der Anweisung MOVE_BLK werden 10 Elemente aus dem "Array_1" des Datenbausteins "Data_DB" in das "Array_2" des gleichen Datenbausteins kopiert.

Siehe auch

MOVE: Wert kopieren (Seite 1383)

MOVE_BLK: Bereich kopieren (Seite 1396)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1398)
UMOVE_BLK: Bereich ununterbrechbar kopieren (Seite 1402)

2.7 IEC-Zeiten und -Zähler verwenden

Vorteile der IEC-Zeiten und -Zähler

Mit Hilfe der durchgängigen Verwendung von IEC-Zeiten und -Zählern wird Ihr Programmcode effektiver.

Dadurch haben Sie die folgenden Vorteile:

- Die Bausteine können mehrfach mit neu erzeugten Instanzdatenbausteinen aufgerufen werden.
- Die IEC-Zähler verfügen über einen großen Zählbereich.
- Die IEC-Zeiten verfügen gegenüber den S5-Zeiten über eine bessere Performance und die Zeitgenauigkeit ist höher.

Programmierung in STEP 7 V5.x

Die S5-Zeiten und -Zähler in STEP 7 V5.x wurden über eine Nummer absolut adressiert. Durch diese Nummernabhängigkeit waren Programmbausteine mit S5-Zeiten und -Zählern nicht mehrfach verwendbar.

Der Wertebereich einer Zeit war auf maximal 9990s und der eines Zählers auf einen maximalen Grenzwert von 999 begrenzt.

Vorgehensweise in STEP 7 TIA Portal

Deklarieren Sie die IEC-Zeiten und -Zähler in dem Programmbaustein, in dem sie aufgerufen bzw. gebraucht werden. Die IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, oder z. B. TON_TIME und TON_LTIME, die Sie auch als lokale Variable in einem Baustein deklarieren können. Der IEC-Zähler ist eine Struktur vom Datentyp IEC_SCOUNTER, IEC_USCOUNTER, etc.

Programmbeispiele im TIA Portal

Das folgende Beispiel zeigt, wie Sie eine IEC-Zeit und einen IEC-Zähler als lokale Variable deklarieren können:

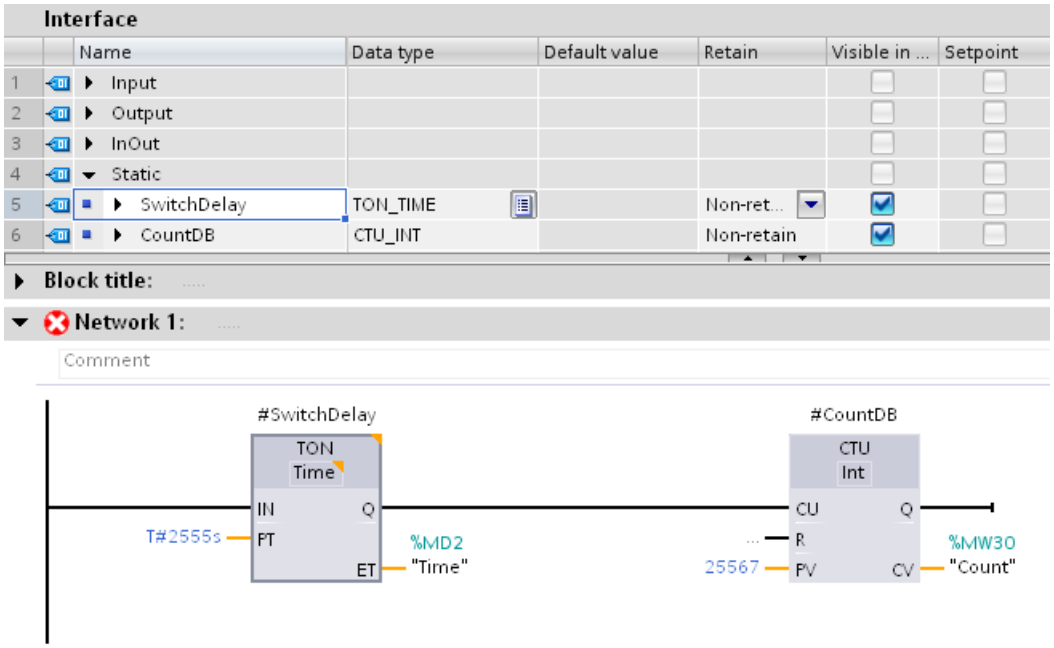

Die Daten der IEC-Zeit TON und des IEC-Zählers CTU werden als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt.

Sie haben auch die Möglichkeit, IEC-Zeiten und IEC-Zähler innerhalb von Strukturen als Multiinstanzen anzulegen und diese in Ihrem Programmcode zu verwenden.

1. Legen Sie dazu z. B. einen globalen Datenbaustein mit einem ARRAY of TON an. Der Datentyp TON erscheint nicht in der Klappliste, er kann aber manuell eingegeben werden:

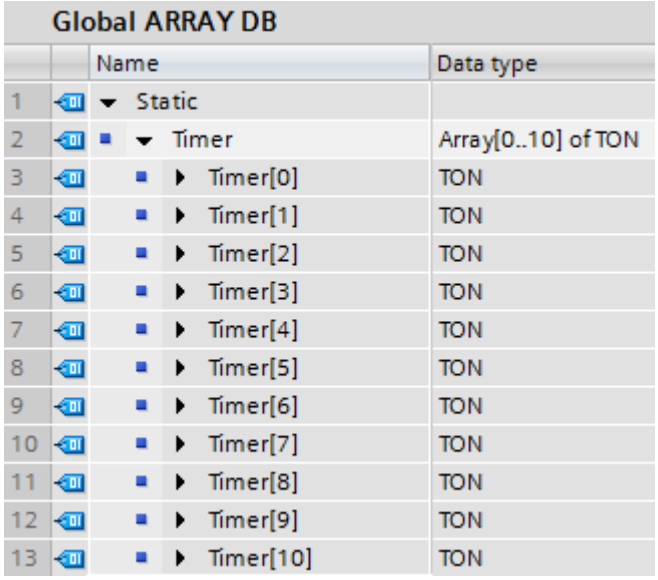

Global ARRAY DB

		Name	Data type
1		Static	
2		Timer	Array[0..10] of TON
3		Timer[0]	TON
4		Timer[1]	TON
5		Timer[2]	TON
6		Timer[3]	TON
7		Timer[4]	TON
8		Timer[5]	TON
9		Timer[6]	TON
10		Timer[7]	TON
11		Timer[8]	TON
12		Timer[9]	TON
13		Timer[10]	TON

2. Legen Sie einen Funktionsbaustein an und ziehen Sie die Anweisung "TON: Einschaltverzögerung erzeugen" in ein Netzwerk. Rufen Sie die Instanz der IEC-Zeit TON wie folgt auf:

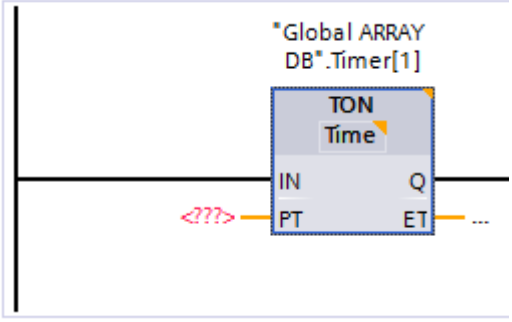

Aufruf einer Zeit als Multiinstanz

Wenn Sie zum Start einer Zeit als Multiinstanz den Parameter IN verwenden möchten, dann dürfen Sie diesen nicht vorher in Ihrem Programmcode initialisieren. In diesem Fall erkennt die anschließend aufgerufene Zeit am Parameter IN keine steigende Signalflanke mehr und die Zeit wird nicht gestartet.

1. Legen Sie die Zeit "Time_1" als Multiinstanz vom Datentyp "TP_TIME" in der Bausteinschnittstelle im Abschnitt "Static" an.

2. Schreiben Sie folgenden Programmcode:

AWL	Erläuterung
U "Tag_Output"	// Wenn der Ausgang "Tag_Output" den Signalzustand 1 erhält, dann
= #Time_1.IN	// wird der Parameter IN der Multiinstanz Zeit #Zeit_1 mit einer positiven Signalflanke initialisiert.

CALL #Time_1
time_type := Time
IN := "Tag_Output"
PT := T#30s
Q := "Tag_4"
ET := "Tag_ElapsedTime"
	// Wenn die Multiinstanz Zeit nun aufgerufen wird und der Parameter IN erneut abgefragt wird, dann wird die Zeit nicht gestartet, denn es liegt keine erneute positive Signalflanke vor.
	
	// Geben Sie als Datentyp der Anweisung TIME an.

Aus diesem Grund müssen Sie die Initialisierung der Multiinstanz Zeit innerhalb des Aufrufs programmieren.

AWL	Erläuterung

CALL #Time_1
time_type := Time
IN := "Tag_Output"
PT := T#30s
Q := "Tag_4"
ET := "Tag_ElapsedTime"
	// Die Zeit wird aufgerufen und gestartet.
	
	// Geben Sie als Datentyp der Anweisung TIME an.

Siehe auch

Aufruf von IEC-Zeiten (Seite 1728)

Aufruf von IEC-Zählern (Seite 1759)

Zeiten (Seite 268)

Datentypen 3

3.1 Übersicht über die gültigen Datentypen

Gültigkeit der Datentypgruppen

Die Datentypen legen die Eigenschaften der Daten fest, z. B. die Darstellung des Inhalts und die zulässigen Speicherbereiche.

Im Anwenderprogramm haben Sie die Möglichkeit vordefinierte Datentypen zu verwenden, die Sie zusätzlich auch zu selbst definierten Datentypen zusammenstellen können. Dafür stehen Ihnen die folgenden Datentypkategorien zur Verfügung:

- Elementare Datentypen (Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR, WCHAR)
- Zusammengesetzte Datentypen (DT, LDT, DTL, STRING, WSTRING, ARRAY, STRUCT)
- Anwenderdefinierte Datentypen (PLC-Datentyp (UDT))
- Zeiger
- Parametertypen
- Systemdatentypen
- Hardwaredatentypen

Die folgenden Tabellen zeigen die Verfügbarkeit der Datentypen in den verschiedenen S7-CPUs:

Tabelle 3-1 Binärzahlen

Binärzahlen	S7-300/400	S7-1200	S7-1500
BOOL (Seite 248)	X	X	X
Bitfolgen			
BYTE (Seite 249)	X	X	X
WORD (Seite 250)	X	X	X
DWORD (Seite 251)	X	X	X
LWORD (Seite 252)	-	-	X

Tabelle 3-2 Ganzzahlen

Ganzzahlen	S7-300/400	S7-1200	S7-1500
SINT (Seite 254)	-	X	X
INT (Seite 256)	X	X	X
DINT (Seite 258)	X	X	X
USINT (Seite 255)	-	X	X
UINT (Seite 257)	-	X	X

Ganzzahlen	S7-300/400	S7-1200	S7-1500
UDINT (Seite 259)	-	X	X
LINT (Seite 260)	-	-	X
ULINT (Seite 261)	-	-	X

Tabelle 3-3 Gleitpunktzahlen

Gleitpunktzahlen	S7-300/400	S7-1200	S7-1500
REAL (Seite 264)	X	X	X
LREAL (Seite 265)	-	X	X

Tabelle 3-4 Zeiten

Zeiten	S7-300/400	S7-1200	S7-1500
S5TIME (Seite 268)	X	-	X
TIME (Seite 269)	X	X	X
LTIME (Seite 270)	-	-	X

Tabelle 3-5 Datum und Uhrzeit

Datum und Uhrzeit	S7-300/400	S7-1200	S7-1500
DATE (Seite 271)	X	X	X
TOD (TIME_OF_DAY) (Seite 271)	X	X	X
LTOD (LTIME_OF_DAY) (Seite 272)	-	-	X
DT (DATE_AND_TIME) (Seite 272)	X	-	X
LDT (Seite 274)	-	-	X
DTL (Seite 274)	-	X	X

Tabelle 3-6 Zeichenfolgen

Zeichenfolgen	S7-300/400	S7-1200	S7-1500
CHAR (Seite 276)	X	X	X
WCHAR (Seite 276)	-	X	X
STRING (Seite 277)	X	X	X
WSTRING (Seite 280)	-	X	X

Tabelle 3-7 PLC-Datentypen (UDT)

PLC-Datentypen (UDT)	S7-300/400	S7-1200	S7-1500
PLC-Datentyp (Seite 286) (UDT)	X	X	X

Tabelle 3-8 Anonyme Strukturen

Anonyme Strukturen	S7-300/400	S7-1200	S7-1500
STRUCT (Seite 291)	X	X	X

Tabelle 3-9 ARRAY

ARRAY	S7-300/400	S7-1200	S7-1500
ARRAY [….] of <Datentyp> (Seite 296)	X	X	X

Tabelle 3-10 Zeiger

Zeiger	S7-300/400	S7-1200	S7-1500
Referenzen (Seite 315)	-	-	X
VARIANT (Seite 339)	-	X	X
POINTER (Seite 356)	X	-	X
ANY (Seite 359)	X	-	X

Tabelle 3-11 Parametertypen

Parametertypen	S7-300/400	S7-1200	S7-1500
TIMER (Seite 363)	X	-	X
COUNTER (Seite 363)	X	-	X
BLOCK_FC (Seite 363)	X	-	X
BLOCK_FB (Seite 363)	X	-	X
BLOCK_DB (Seite 363)	X	-	-
BLOCK_SDB (Seite 363)	X	-	-
VOID (Seite 363)	X	X	X
PARAMETER (Seite 363)	-	X	X

Tabelle 3-12 Systemdatentypen

Systemdatentypen	S7-300/400	S7-1200	S7-1500
IEC_TIMER (Seite 364)	X[1)]	X	X
IEC_LTIMER (Seite 364)	-	-	X
IEC_SCOUNTER (Seite 364)	-	X	X
IEC_USCOUNTER (Seite 364)	-	X	X
IEC_COUNTER (Seite 364)	X[2)]	X	X
IEC_UCOUNTER (Seite 364)	-	X	X
IEC_DCOUNTER (Seite 364)	-	X	X
IEC_UDCOUNTER (Seite 364)	-	X	X
IEC_LCOUNTER (Seite 364)	-	-	X
IEC_ULCOUNTER (Seite 364)	-	-	X

Systemdatentypen	S7-300/400	S7-1200	S7-1500
ERROR_STRUCT (Seite 364)	-	X	X
NREF (Seite 364)	-	X	X
CREF (Seite 364)	-	X	X
VREF (Seite 364)	-	X	X
SSL_HEADER (Seite 364)	X	-	-
CONDITIONS (Seite 364)	-	X	-
TADDR_Param (Seite 364)	-	X	X
TCON_Param (Seite 364)	-	X	X
HSC_Period (Seite 364)	-	X	-
[1] Bei S7-300/400 CPUs wird der Datentyp durch TP, TON und TOF repräsentiert. [2] Bei S7-300/400 CPUs wird der Datentyp durch CTU, CTD und CTUD repräsentiert.			

Tabelle 3-13 Hardwaredatentypen

Hardwaredatentypen	S7-300/400	S7-1200	S7-1500
REMOTE (Seite 366)	-	X	X
HW_ANY (Seite 366)	-	X	X
HW_DEVICE (Seite 366)	-	X	X
HW_DPMASTER (Seite 366)	-	-	X
HW_DPSLAVE (Seite 366)	-	X	X
HW_IO (Seite 366)	-	X	X
HW_IOSYSTEM (Seite 366)	-	X	X
HW_SUBMODULE (Seite 366)	-	X	X
HW_MODULE (Seite 366)	-	-	X
HW_INTERFACE (Seite 366)	-	X	X
HW_IEPORT (Seite 366)	-	X	X
HW_HSC (Seite 366)	-	X	X
HW_PWM (Seite 366)	-	X	X
HW_PTO (Seite 366)	-	X	X
EVENT_ANY (Seite 366)	-	X	X
EVENT_ATT (Seite 366)	-	X	X
EVENT_HWINT (Seite 366)	-	X	X
OB_ANY (Seite 366)	-	X	X
OB_DELAY (Seite 366)	-	X	X
OB_TOD (Seite 366)	-	X	X
OB_CYCLIC (Seite 366)	-	X	X
OB_ATT (Seite 366)	-	X	X
OB_PCYCLE (Seite 366)	-	X	X
OB_HWINT (Seite 366)	-	X	X
OB_DIAG (Seite 366)	-	X	X
OB_TIMEERROR (Seite 366)	-	X	X
OB_STARTUP (Seite 366)	-	X	X

Hardwaredatentypen	S7-300/400	S7-1200	S7-1500
PORT (Seite 366)	-	X	X
RTM (Seite 366)	-	X	X
PIP (Seite 366)	-	-	X
CONN_ANY (Seite 366)	-	X	X
CONN_PRG (Seite 366)	-	X	X
CONN_OUC (Seite 366)	-	X	X
CONN_R_ID (Seite 366)	-	-	X
DB_ANY (Seite 366)	-	X	X
DB_WWW (Seite 366)	-	X	X
DB_DYN (Seite 366)	-	X	X

Hinweis

Abhängig von der CPU-Version können die tatsächlich gültigen Datentypen geringfügig von der Tabelle abweichen.

3.2 Binärzahlen

3.2.1 BOOL (Bit)

Beschreibung

Ein Operand vom Datentyp BOOL stellt einen Bitwert dar und enthält einen der folgenden Werte:

- TRUE
- FALSE

Die folgende Tabelle zeigt die Eigenschaften des Datentyps BOOL:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
1	Boolesch	FALSE oder TRUE BOOL#0 oder BOOL#1 BOOL#FALSE oder BOOL#TRUE	TRUE BOOL#1 BOOL#TRUE
	Ganzzahlen ohne Vorzeichen (Dezimalsystem)	0 oder 1	1
	Dualzahlen	2#0 oder 2#1	2#0
	Oktalzahlen	8#0 oder 8#1	8#1
	Hexadezimalzahlen	16#0 oder 16#1	16#1

Hinweis

Gültig für CPUs der Baureihe S7-1500

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das Bit eine Länge von 1 Byte.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.2.2 Bitfolgen

3.2.2.1 BYTE

Beschreibung

Ein Operand vom Datentyp BYTE ist eine Bitfolge aus 8 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps BYTE:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben	
			Konstanten	Absolute und symbolische Adressen
8	Ganzzahlen[1] (Dezimalsystem)	Ganzzahlen mit Vorzeichen: -128 bis +127 Ganzzahlen ohne Vorzeichen: 0 bis 255	• 15 • BYTE#15 • BYTE#10#15 • B#15	• EB2 • MB10 • DB1.DBB4 • Variable_Name
	Dualzahlen	2#0 bis 2#1111_1111	• 2#0000_1111 • BYTE#2#0000_1111 • B#2#0000_1111	
	Oktalzahlen	8#0 bis 8#377	• 8#17 • BYTE#8#17 • B#8#17	
	Hexadezimalzahlen	16#0 bis 16#FF	• 16#0F • BYTE#16#0F • B#16#0F	
[1] Der Wertebereich hängt von der jeweiligen Interpretation bzw. Konvertierung ab.				

Hinweis

Der Datentyp BYTE kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen SINT und USINT verarbeiten können.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.2.2.2 WORD

Beschreibung

Ein Operand vom Datentyp WORD ist eine Bitfolge aus 16 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps WORD:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe	
			Konstanten	Absolute und symbolische Adressen
16	Ganzzahlen (Dezimalsystem)	Ganzzahlen mit Vorzeichen: -32_768 bis +32_767 Ganzzahlen ohne Vorzeichen: 0 bis 65_535	• 61_680 • WORD#61_680 • WORD#10#61_680 • W#61_680	• MW10 • DB1.DBW2 • Variable_Name
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111	• 2#1111_0000_1111_0000 • WORD#2#1111_0000_1111_0000 • W#2#1111_0000_1111_0000	
	Oktalzahlen	8#0 bis 8#177_777	• 8#170_360 • WORD#8#170_360 • W#8#170_360	
	Hexadezimalzahlen	16#0 bis 16#FFFF	• 16#F0F0 • WORD#16#F0F0 • W#16#F0F0	
	BCD	C#0 bis C#999	C#55	
	Dezimalfolge	B#(0, 0) bis B#(255, 255)	B#(127, 200)	

Hinweis

Der Datentyp WORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen INT und UINT verarbeiten können.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.2.2.3 DWORD

Beschreibung

Ein Operand vom Datentyp DWORD ist eine Bitfolge aus 32 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DWORD:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben	
			Konstanten	Absolute und symbolische Adressen
32	Ganzzahlen (Dezimalsystem)	Ganzzahlen mit Vorzeichen: -2_147_483_647 bis +2_147_483_647 Ganzzahlen ohne Vorzeichen: 0 bis 4_294_967_295	• +15_793_935 • DWORD#+15_793_935 • DWORD#10# +15_793_935 • DW#+15_793_935	• MD10 • DB1.DBD8 • Variable_Name
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111_1111_1111_1111_1111	• 2#0000_0000_1111_0000_1111_1111_0000_1111 • DWORD#2#0000_0000_1111_0000_1111_1111_0000_1111 • DW#2#0000_0000_1111_0000_1111_1111_0000_1111	
	Oktalzahlen	8#0 bis 8#37_777_777_777	• 8#74_177_417 • DWORD#8#74_177_417 • DW#8#74_177_417	
	Hexadezimalzahlen	16#0000_0000 bis 16#FFFF_FFFF	• 16#00F0_FF0F • DWORD#16#00F0_FF0F • DW#16#00F0_FF0F	
	Dezimalfolge	B#(0, 0, 0, 0) bis B#(255, 255, 255, 255)	B#(127, 200, 127, 200)	

Hinweis

Der Datentyp DWORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen DINT und UDINT verarbeiten können.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.2.2.4 LWORD

Beschreibung

Ein Operand vom Datentyp LWORD ist eine Bitfolge aus 64 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LWORD:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
64	Ganzzahlen (Dezimalsystem)	Ganzzahlen mit Vorzeichen: -9_223_372_036_854_775_808 bis +9_223_372_036_854_775_807 Ganzzahlen ohne Vorzeichen: 0 bis 18_446_744_073_709_551_615	• +26_123_590_360_715 • LWORD#+26_123_590_360_715 • LWORD#10#+26_123_590_360_715 • LW#+26_123_590_360_715
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111	• 2#0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011 • LWORD#2#0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011 • LW#2#0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011
	Oktalzahlen	8#0 bis 8#1_777_777_777_777_777_777_777	• 8#13_724_557_213 • LWORD#8#13_724_557_213 • LW#8#13_724_557_213
	Hexadezimalzahlen	16#0000_0000 bis 16#FFFF_FFFF_FFFF_FFFF	• 16#0000_0000_5F52_DE8B • LWORD#16#0000_0000_5F52_DE8B • LW#16#0000_0000_5F52_DE8B
	Dezimalfolge	B#(0, 0, 0, 0, 0, 0, 0, 0) bis B#(255, 255, 255, 255, 255, 255, 255, 255)	B#(127, 200, 127, 200, 127, 200, 127, 200)

Hinweis

Der Datentyp LWORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen LINT und ULINT verarbeiten können.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Übersicht über die Datentypkonvertierung (Seite 369)

Grundlagen zu Konstanten (Seite 102)

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)
Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3 Ganzzahlen

3.3.1 SINT (8-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp SINT (Short INT) hat eine Länge von 8 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 6 stehen für die Größe der Zahl. Der Signalzustand von Bit 7 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

Ein Operand vom Datentyp SINT belegt im Speicher ein BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps SINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
8	Ganzzahlen mit Vorzeichen (Dezimalsystem)	-128 bis +127	• +44 • SINT#+44 • SINT#10#+44 Unter Verwendung der Typisierung SINT# geht der Wertebereich bis maximal SINT#255. Dieser Wert wird als Ganzzahl mit -1 interpretiert.
	Dualzahlen (nur positiv)	2#0 bis 2#0111_1111	• 2#0010_1100 • SINT#2#0010_1100 • SINT#2#10
	Oktalzahlen (nur positiv)	8#0 bis 8#177	• 8#54 • SINT#8#54
	Hexadezimalzahlen (nur positiv)	16#0 bis 16#7F	• 16#2C • SINT#16#2C Unter Verwendung der Typisierung SINT# geht der Wertebereich bis maximal SINT#16#FF. Dieser Wert wird als Ganzzahl mit -1 interpretiert.

Beispiel

Das folgende Bild zeigt die Ganzzahl +44 als Dualzahl:

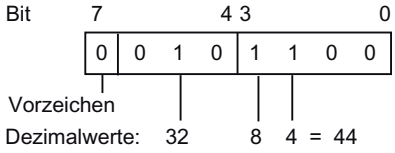

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.2 USINT (8-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp USINT (Unsigned Short INT) hat eine Länge von 8 Bit und enthält Zahlenwerte ohne Vorzeichen.

Ein Operand vom Datentyp USINT belegt im Speicher ein BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps USINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
8	Ganzzahlen ohne Vorzeichen (Dezimalsystem)	0 bis 255	• 78 • USINT#78 • USINT#10#78
	Dualzahlen	2#0 bis 2#1111_1111	• 2#0100_1110 • USINT#2#0100_1110 • USINT#2#10
	Oktalzahlen	8#0 bis 8#377	• 8#116 • USINT#8#116
	Hexadezimalzahlen	16#0 bis 16#FF	• 16#4E • USINT#16#4E

Beispiel

Das folgende Bild zeigt die Ganzzahl 78 als Dualzahl:

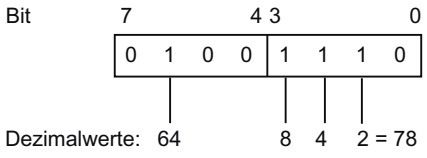

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.3 INT (16-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp INT hat eine Länge von 16 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 14 stehen für die Größe der Zahl. Der Signalzustand von Bit 15 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

Ein Operand vom Datentyp INT belegt im Speicher zwei BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps INT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
16	Ganzzahlen mit Vorzeichen (Dezimalsystem)	-32_768 bis +32_767	• +3_785 • INT#+3_785 • INT#10#+3_785
	Dualzahlen (nur positiv)	2#0 bis 2#0111_1111_1111_1111	• 2#0000_1110_1100_100 1 • INT#2#0000_1110_1100_ 1001 • INT#2#10
	Oktalzahlen (nur positiv)	8#0 bis 8#7_7777	• 8#7311 • INT#8#7311
	Hexadezimalzahlen (nur positiv)	16#0 bis 16#7FFF	• 16#0EC9 • INT#16#0EC9

Beispiel

Das folgende Bild zeigt die Ganzzahl +3785 als Dualzahl:

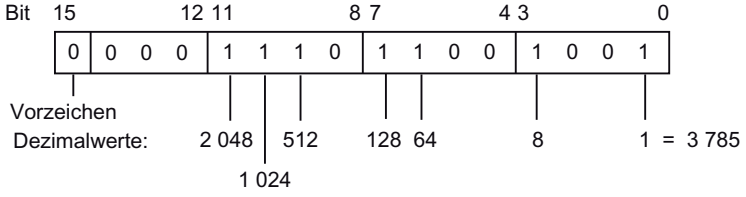

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.4 UINT (16-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp UINT (Unsigned INT) hat eine Länge von 16 Bit und enthält Zahlenwerte ohne Vorzeichen.

Ein Operand vom Datentyp UINT belegt im Speicher zwei BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps UINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
16	Ganzzahlen ohne Vorzeichen (Dezimalsystem)	0 bis 65_535	• 65_295 • UINT#65_295 • UINT#10#65_295
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111	• 2#1111_1111_0000_1111 • UINT#2#1111_1111_0000_1111 • UINT#2#10
	Oktalzahlen	8#0 bis 8#17_7777	• 8#17_7417 • UINT#8#17_7417
	Hexadezimalzahlen	16#0 bis 16#FFFF	• 16#FF0F • UINT#16#FF0F

Beispiel

Das folgende Bild zeigt die Ganzzahl 65295 als Dualzahl:

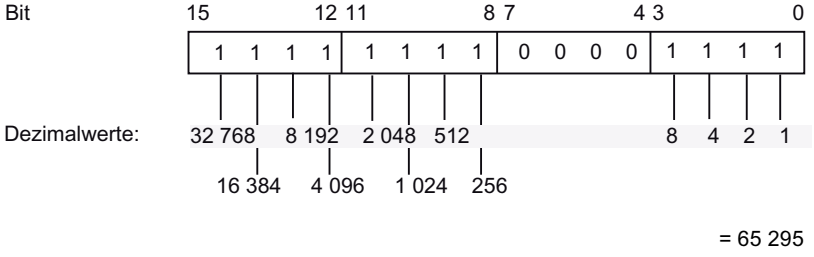

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.5 DINT (32-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp DINT (Double INT) hat eine Länge von 32 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 30 stehen für die Größe der Zahl. Der Signalzustand von Bit 31 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

Ein Operand vom Datentyp DINT belegt im Speicher vier BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
32	Ganzzahlen mit Vorzeichen (Dezimalsystem)	-2_147_483_648 bis +2_147_483_647	• +125_790 • DINT#+125_790 • DINT#10#+125_790 • L#275
	Dualzahlen (nur positiv)	2#0 bis 2#0111_1111_1111_1111_1111_1111_1111_1111	• 2#0000_0000_0000_0001_1110_1011_0101_1110 • DINT#2#0000_0000_0000_0001_1110_1011_0101_1110 • DINT#2#10
	Oktalzahlen (nur positiv)	8#0 bis 8#177_7777_7777	• 8#36_5536 • DINT#8#36_5536
	Hexadezimalzahlen	16#0 bis 16#7FFF_FFFF	• 16#0001_EB5E • DINT#16#0001_EB5E

Beispiel

Das folgende Bild zeigt die Ganzzahl +125790 als Dualzahl:

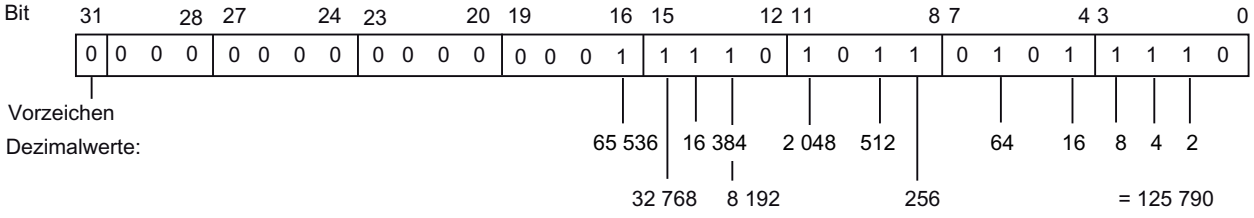

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.6 UDINT (32-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp UDINT (Unsigned Double INT) hat eine Länge von 32 Bit und enthält Zahlenwerte ohne Vorzeichen.

Ein Operand vom Datentyp UDINT belegt im Speicher vier BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps UDINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
32	Ganzzahlen ohne Vorzeichen (Dezimalsystem)	0 bis 4_294_967_295	• 4_042_322_160 • UDINT#4_042_322_160 • UDINT#10#4_042_322_160
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111_1111_1111_1111_1111	• 2#1111_0000_1111_0000_1111_0000_1111_0000 • UDINT#2#1111_0000_1111_0000_1111_0000_1111_0000 • UDINT#2#10
	Oktalzahlen	8#0 bis 8#377_7777_7777	• 8#360_7417_0360 • UDINT#8#360_7417_0360
	Hexadezimalzahlen	16#0 bis 16#FFFF_FFFF	• 16#F0F0_F0F0 • UDINT#16#F0F0_F0F0

Beispiel

Das folgende Bild zeigt die Ganzzahl 4042322160 als Dualzahl:

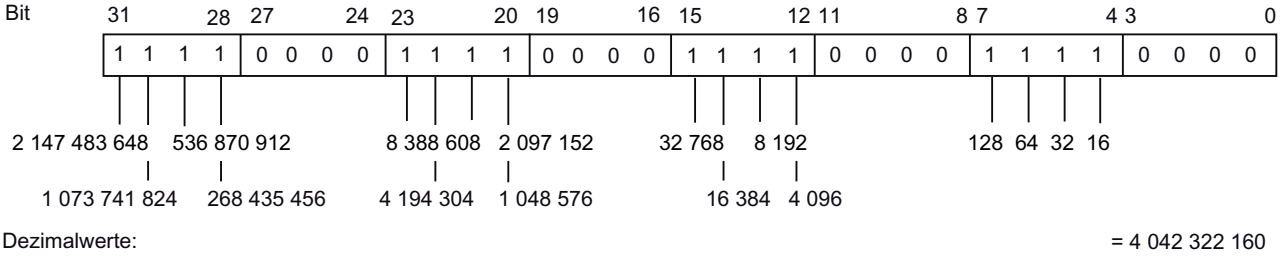

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.7 LINT (64-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp LINT (Long INT) hat eine Länge von 64 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 62 stehen für die Größe der Zahl. Der Signalzustand von Bit 63 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

Ein Operand vom Datentyp LINT belegt im Speicher acht BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
64	Ganzzahlen mit Vorzeichen (Dezimalsystem)	-9_223_372_036_854_775_808 bis +9_223_372_036_854_775_807	• +154_325_790_816_159 • LINT#+154_325_790_816_159 • LINT#10#+154_325_790_816_159
	Dualzahlen (nur positiv)	2#0 bis 2#0111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111	• 2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 • LINT#2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 • LINT#2#10
	Oktalzahlen (nur positiv)	8#0 bis 8#7_7777_7777_7777_7777_7777	• 8#4305_5705_7417_3637 • LINT#8#4305_5705_7417_3637
	Hexadezimalzahlen (nur positiv)	16#0 bis 16#7FFF_FFFF_FFFF_FFFF	• 16#0000_8C5B_C5F0_F79F • LINT#16#0000_8C5B_C5F0_F79F

Beispiel

Das folgende Bild zeigt die Ganzzahl +154325790816159 als Dualzahl:

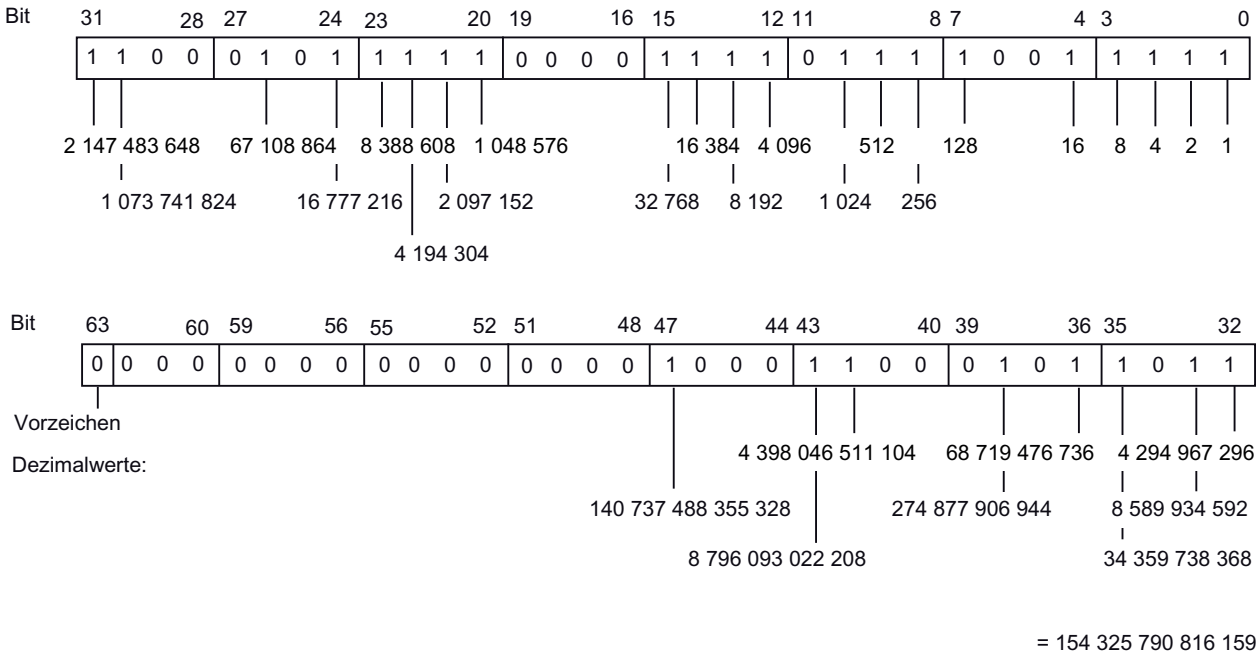

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Übersicht über die Datentypkonvertierung (Seite 369)

Grundlagen zu Konstanten (Seite 102)

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.3.8 ULINT (64-Bit-Ganzzahlen)

Beschreibung

Ein Operand vom Datentyp ULINT (Unsigned Long INT) hat eine Länge von 64 Bit und enthält Zahlenwerte ohne Vorzeichen.

Ein Operand vom Datentyp ULINT belegt im Speicher acht BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps ULINT:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingaben
64	Ganzzahlen ohne Vorzeichen (Dezimalsystem)	0 bis 18_446_744_073_709_551_615	• 154_325_790_816_159 • ULINT#154_325_790_816_159 • ULINT#10#154_325_790_816_159
	Dualzahlen	2#0 bis 2#1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111	• 2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 • ULINT#2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 • ULINT#2#10
	Oktalzahlen	8#0 bis 8#17_7777_7777_7777_7777_7777	• 8#4305_5705_7417_3637 • ULINT#8#4305_5705_7417_3637
	Hexadezimalzahlen	16#0 bis 16#FFFF_FFFF_FFFF_FFFF	• 16#0000_8C5B_C5F0_F79F • ULINT#16#0000_8C5B_C5F0_F79F

Beispiel

Das folgende Bild zeigt die Ganzzahl 154325790816159 als Dualzahl:

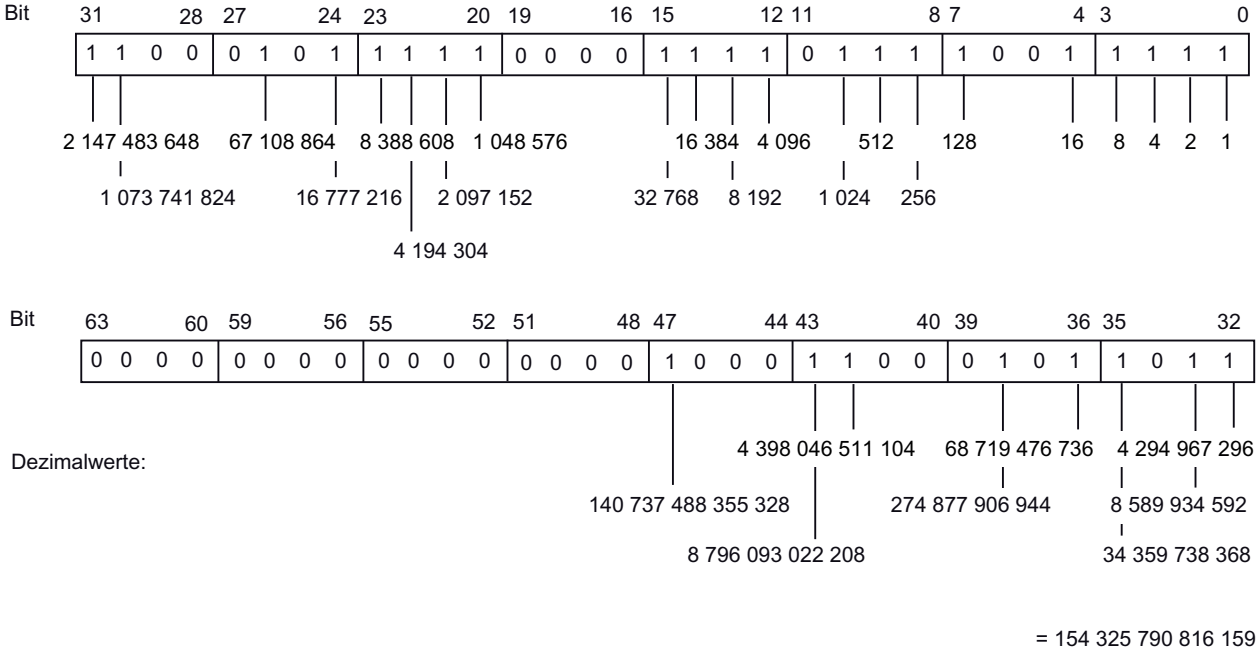

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)
Übersicht über die Datentypkonvertierung (Seite 369)
Grundlagen zu Konstanten (Seite 102)
Implizite Konvertierungen (Seite 371)
Explizite Konvertierungen (Seite 423)
Datentypkonvertierungen bei S7-1200 (Seite 507)

3.4 Gleitpunktzahlen

3.4.1 REAL

Beschreibung

Operanden vom Datentyp REAL haben eine Länge von 32 Bits und werden zur Darstellung von Gleitpunktzahlen verwendet. Ein Operand vom Datentyp REAL besteht aus den folgenden drei Komponenten:

- Vorzeichen: Das Vorzeichen wird durch den Signalzustand von Bit 31 bestimmt. Das Bit 31 kann die Werte "0" (positiv) und "1" (negativ) annehmen.
- 8-Bit-Exponenten zur Basis 2: Der Exponent wird um eine Konstante (Basis, +127) erhöht, so dass er einen Wertebereich von 0 bis 255 aufweist.
- 23-Bit-Mantisse: Nur der gebrochene Anteil der Mantisse wird dargestellt. Der ganzzahlige Anteil der Mantisse ist bei normalisierten Gleitpunktzahlen immer 1 und wird nicht gespeichert.

Der Datentyp REAL wird mit einer Genauigkeit von 6 Stellen verarbeitet.

Das folgende Bild zeigt den Aufbau des Datentyps REAL:

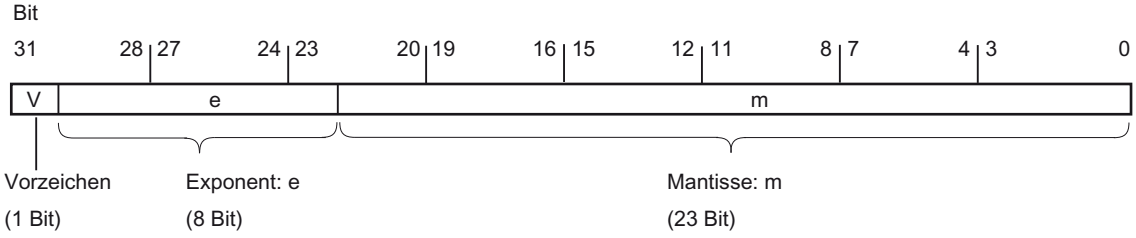

Hinweis

Bei Gleitpunktzahlen werden nur die von der IEEE754 Norm definierten Genauigkeiten gespeichert. Zusätzlich angegebene Dezimalstellen werden nach IEEE754 gerundet.

Bei häufig geschachtelten arithmetischen Berechnungen kann sich die Anzahl der Dezimalstellen verringern.

Werden mehr Dezimalstellen eingegeben, als der Datentyp speichern kann, dann wird die Zahl auf den, der möglichen Genauigkeit in diesem Wertebereich, entsprechenden Wert gerundet.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps REAL:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe
32	Gleitpunktzahlen nach IEEE754	-3.402823e+38 bis -1.175495e-38 ±0,0	1.0e-5; REAL#1.0e-5
	Gleitpunktzahlen	+1.175495e-38 bis +3.402823e+38	1.0; REAL#1.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Mit Gleitpunktzahlen (REAL und LREAL) in SCL rechnen (Seite 7623)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.4.2 LREAL

Beschreibung

Operanden vom Datentyp LREAL haben eine Länge von 64 Bits und werden zur Darstellung von Gleitpunktzahlen verwendet. Ein Operand vom Datentyp LREAL besteht aus den folgenden drei Komponenten:

- Vorzeichen: Das Vorzeichen wird durch den Signalzustand von Bit 63 bestimmt. Das Bit 63 kann die Werte "0" (positiv) und "1" (negativ) annehmen.
- 11-Bit-Exponenten zur Basis 2: Der Exponent wird um eine Konstante (Basis, +1023) erhöht, so dass er einen Wertebereich von 0 bis 2047 aufweist.
- 52-Bit-Mantisse: Nur der gebrochene Anteil der Mantisse wird dargestellt. Der ganzzahlige Anteil der Mantisse ist bei normalisierten Gleitpunktzahlen immer 1 und wird nicht gespeichert.

Der Datentyp LREAL wird mit einer Genauigkeit von 15 Stellen verarbeitet.

Das folgende Bild zeigt den Aufbau des Datentyps LREAL:

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LREAL:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe
64	Gleitpunktzahlen nach IEEE754	-1.7976931348623157e+308 bis -2.2250738585072014e-308	1.0e-5; LREAL#1.0e-5
	Gleitpunktzahlen	±0,0 +2.2250738585072014e-308 bis +1.7976931348623157e+308	1.0; LREAL#1.0

Hinweis

Bei Gleitpunktzahlen werden nur die von der IEEE754 Norm definierten Genauigkeiten gespeichert. Zusätzlich angegebene Dezimalstellen werden nach IEEE754 gerundet.

Bei häufig geschachtelten arithmetischen Berechnungen kann sich die Anzahl der Dezimalstellen verringern.

Werden mehr Dezimalstellen eingegeben als der Datentyp speichern kann, dann wird die Zahl auf den, der möglichen Genauigkeit in diesem Wertebereich, entsprechenden Wert gerundet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Mit Gleitpunktzahlen (REAL und LREAL) in SCL rechnen (Seite 7623)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.4.3 Ungültige Gleitpunktzahlen

Beschreibung

Bei den Datentypen REAL und LREAL unterscheidet man vier Zahlenbereiche:

- normalisierte Zahlen, die mit der vollen Genauigkeit gespeichert werden
- denormalisierte Zahlen, die nicht mit der vollen Genauigkeit gespeichert werden können
- Unendliche Zahlen: +Inf/-Inf (Infinity)
- Ungültige Zahlen: NaN (Not a Number)

Hinweis

Gleitpunktzahlen werden wie in der IEEE754 Norm abgelegt. Ergebnisse von Konvertierungen bzw. arithmetischen Funktionen mit einer denormalisierten, unendlichen oder NaN (Not a Number) Gleitpunktzahl sind CPU abhängig.

Wenn Sie in mathematischen Funktionen nicht mit normalisierten Gleitpunktzahlen arbeiten, dann gibt es im Ergebnis, abhängig davon welche CPU aus welcher Baureihe Sie verwenden, starke Unterschiede.

Eine CPU kann nicht mit denormalisierten Gleitpunktzahlen rechnen, mit Ausnahme von älteren CPU Versionen der Baureihen S7-300 und S7-400. Das Bitmuster einer denormalisierten Zahl wird wie eine Null interpretiert. Fällt ein Rechenergebnis in diesen Bereich, wird mit Null weiter gemacht, wobei die Statusbits OV und OS gesetzt werden (Zahlenbereichsunterschreitung).

Obwohl die Werte von ungültigen Gleitpunktzahlen nur mit einer eingeschränkten Genauigkeit bei mathematischen Funktionen dargestellt werden können, können Zahlen mit einem Exponenten von -39 (z. B. 2,4408e-039) im TIA Portal beobachtet werden und stellen daher nicht unbedingt ein fehlerhaftes Ergebnis dar. Daher kann es vorkommen, dass Gleitpunktzahlenwerte außerhalb der gültigen Zahlenwerte liegen.

Hinweis

Für CPUs der Baureihe S7-1200 V1, V2 und V3 gilt:

Die Vergleichsoperation "Gleich" verwendet das Bitmuster der ungültigen Gleitpunktzahl. Wenn zwei "NaN-Zahlen" mit dem gleichen Bitmuster verglichen werden, dann liefert der Ausgang der Vergleichsoperation "Gleich" das Ergebnis TRUE.

Hinweis

Für CPUs der Baureihe S7-1200 V4 und S7-1500 gilt:

Werden zwei ungültige Zahlen (NaN) miteinander verglichen, so ist das Ergebnis immer FALSE, unabhängig vom Bitmuster der ungültigen Zahl oder der Relation (>, >, ...).

Hinweis

Vergleich von denormalisierten Gleitpunktzahlen

Bei der Vergleichsoperation "Gleich" mit zwei denormalisierten Gleitpunktzahlen wird der Ausgang bei CPUs der Baureihen S7-300/400 auf den Signalzustand "0" und bei CPUs der Baureihen S7-1200/1500 auf den Signalzustand "1" gesetzt.

Wenn die Eingangsvariablen einer mathematischen Funktion eine ungültige Gleitpunktzahl darstellen, dann wird auch eine ungültige Gleitpunktzahl als Ergebnis ausgegeben.

Zur Auswertung möglicher Fehler durch ungültige Gleitpunktzahlen, haben Sie folgende Möglichkeiten:

- In KOP/FUP und SCL können Sie den Freigabeausgang ENO auf FALSE abfragen
- In AWL können Sie das Statusbit OV auswerten

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Mit Gleitpunktzahlen (REAL und LREAL) in SCL rechnen (Seite 7623)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.5 Zeiten

3.5.1 S5TIME (Zeitdauer)

Format

Der Datentyp S5TIME speichert die Zeitdauer im BCD-Format. Die Zeitdauer ist das Produkt aus einem Zeitwert im Bereich von 0 bis 999 und einer Zeitbasis. Die Zeitbasis gibt das Intervall an, in dem eine Zeit den Zeitwert um je eine Einheit vermindert, bis er "0" erreicht. Über die Zeitbasis kann die Auflösung der Zeitwerte gesteuert werden.

Die folgende Tabelle zeigt den Wertebereich für den Datentyp S5TIME:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe
16	S7-Zeit in Schritten von 10 ms (Standardwert)	S5T#0MS bis S5T#2H_46M_30S_0MS	S5T#10s, S5TIME#10s

Die folgende Tabelle zeigt die Codierung Zeitbasis für S5TIME:

Zeitbasis	Binärcode für Zeitbasis
10ms	00
100ms	01
1s	10
10s	11

Wenn Sie den Datentyp S5TIME mit Zeiten verwenden, müssen Sie Grenzwerte für den Bereich und die Auflösung der Zeitwerte beachten. Die folgende Tabelle gibt jeweils den Bereich für die verschiedenen Auflösungen an:

Auflösung	Bereich
0,01 s	10ms bis 9s 990ms
0,1 s	100ms bis 1m 39s 900ms
1 s	1s bis 16m 39s
10 s	10s bis 2h 46m 30s

Werte, die 2h46m30s überschreiten, werden nicht akzeptiert.

Beispiel

Das folgende Bild zeigt den Inhalt des Zeitoperanden bei einem Zeitwert von 127 und einer Zeitbasis von 1s:

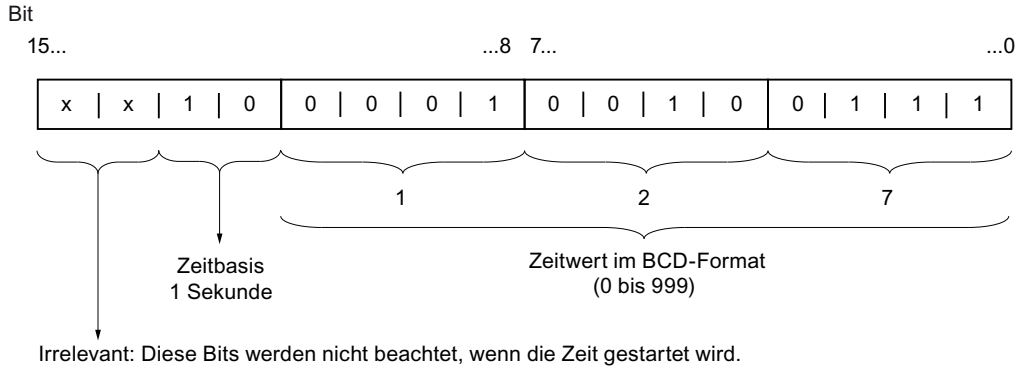

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.5.2 TIME (IEC-Zeit)

Beschreibung

Der Inhalt eines Operanden vom Datentyp TIME wird als Millisekunden interpretiert. Die Darstellung enthält Angaben für Tage (d), Stunden (h), Minuten (m), Sekunden (s) und Millisekunden (ms).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps TIME:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe
32	Zeitdauer mit Vorzeichen	T#-24d_20h_31m_23s_648ms bis T# +24d_20h_31m_23s_647ms	T#10d_20h_30m_20s_630ms, TIME#10d_20h_30m_20s_630ms

Es ist nicht erforderlich, alle Zeiteinheiten anzugeben. So ist z. B. T#5h10s gültig. Wenn nur eine Einheit angegeben wird, darf der absolute Wert an Tagen, Stunden und Minuten die oberen oder unteren Grenzwerte nicht überschreiten. Wenn mehr als eine Zeiteinheit angegeben wird, darf der Wert die Einheit 24 Tage, 23 Stunden, 59 Minuten, 59 Sekunden oder 999 Millisekunden nicht überschreiten.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.5.3 LTIME (IEC-Zeit)

Beschreibung

Der Inhalt eines Operanden vom Datentyp LTIME wird als Nanosekunden interpretiert. Die Darstellung enthält Angaben für Tage (d), Stunden (h), Minuten (m), Sekunden (s), Millisekunden (ms), Mikrosekunden (us) und Nanosekunden (ns).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LTIME:

Länge (Bit)	Format	Wertebereich	Beispiele für Werteingabe
64	Zeitdauer mit Vorzeichen	LT#-106751d_23h_47m_16s_854ms_775us_808ns bis LT#+106751d_23h_47m_16s_854ms_775us_807ns	LT#11350d_20h_25m_14s_830ms_652us_315ns, LTIME#11350d_20h_25m_14s_830ms_652us_315ns

Es ist nicht erforderlich, alle Zeiteinheiten anzugeben. So ist z. B. LT#5h10s gültig. Wenn nur eine Einheit angegeben wird, darf der absolute Wert an Tagen, Stunden und Minuten die oberen oder unteren Grenzwerte nicht überschreiten. Wenn mehr als eine Zeiteinheit angegeben wird, darf der Wert die Einheit 106751 Tage, 23 Stunden, 59 Minuten, 59 Sekunden, 999 Millisekunden, 999 Mikrosekunden oder 999 Nanosekunden nicht überschreiten.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Übersicht über die Datentypkonvertierung (Seite 369)

Grundlagen zu Konstanten (Seite 102)

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6 Datum und Uhrzeit

3.6.1 DATE

Format

Der Datentyp DATE speichert ein Datum als vorzeichenlose Ganzzahl. Die Darstellung enthält das Jahr, den Monat und den Tag.

Der Inhalt eines Operanden vom Datentyp DATE entspricht im Hexadezimalformat der Anzahl der Tage seit 01.01.1990 (16#0000).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DATE:

Länge (Byte)	Format	Wertebereich	Beispiele für Werteingabe
2	IEC-Datum (Jahr-Monat-Tag)	D#1990-01-01 bis D#2169-06-06	D#2009-12-31, DATE#2009-12-31

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6.2 TOD (TIME_OF_DAY)

Format

Der Datentyp TOD (TIME_OF_DAY) belegt ein Doppelwort und speichert die Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr) als vorzeichenlose Ganzzahl.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps TOD:

Länge (Byte)	Format	Wertebereich	Beispiele für Werteingabe
4	Tageszeit (Stunden:Minuten:Sekunden.Millisekunden)	TOD#00:00:00.000 bis TOD#23:59:59.999	TOD#10:20:30.400, TIME_OF_DAY#10:20:30.400

Die Angabe von Stunden, Minuten und Sekunden ist erforderlich. Die Angabe der Millisekunden ist optional.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6.3 LTOD (LTIME_OF_DAY)

Format

Der Datentyp LTOD (LTIME_OF_DAY) belegt zwei Doppelworte und speichert die Anzahl der Nanosekunden seit Tagesbeginn (0:00 Uhr) als vorzeichenlose Ganzzahl.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LTOD:

Länge (Byte)	Format	Wertebereich	Beispiele für Werteingabe
8	Tageszeit (Stunden:Minuten:Sekunden.Nanosekunden)	LTOD#00:00:00.000000000 bis LTOD#23:59:59.999999999	LTOD#10:20:30.400_365_215, LTIME_OF_DAY#10:20:30.400_365_215

Die Angabe von Stunden, Minuten und Sekunden ist erforderlich. Die Angabe der Nanosekunden ist optional.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Übersicht über die Datentypkonvertierung (Seite 369)

Grundlagen zu Konstanten (Seite 102)

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6.4 DT (DATE_AND_TIME)

Format

Der Datentyp DT (DATE_AND_TIME) speichert Angaben zu Datum und Uhrzeit im BCD-Format.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DT:

Länge (Byte)	Format	Wertebereich	Beispiel für Werteingabe
8	Datum und Uhrzeit (Jahr-Monat-Tag-Stunde:Minute:Sekunde:Millisekunde [3])	Min.: DT#1990-01-01-00:00:00.000 Max.: DT#2089-12-31-23:59:59.999	DT#2008-10-25-08:12:34.567, DATE_AND_TIME#2008-10-25-08:12:34.567

Die folgende Tabelle zeigt die Struktur des Datentyps DT:

Byte	Inhalt	Wertebereich
0	Jahr	0 bis 99 (Jahre 1990 bis 2089) BCD#90 = 1990 ... BCD#0 = 2000 ... BCD#89 = 2089
1	Monat	BCD#1 bis BCD#12
2	Tag	BCD#1 bis BCD# 31
3	Stunde	BCD#0 bis BCD#23
4	Minute	BCD#0 bis BCD#59
5	Sekunde	BCD#0 bis BCD#59
6	Die beiden höchstwertigen Ziffern von MSEC	BCD#0 bis BCD#999
7 (4MSB) [1]	Die niederwertige Ziffer von MSEC	BCD#0 bis BCD#9
7 (4LSB) [2]	Wochentag	BCD#1 bis BCD#7 BCD#1 = Sonntag ... BCD#7 = Samstag

[1] MSB: Most Significant Bit
[2] LSB: Least Significant Bit
[3] Festpunktzahl

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)
Grundlagen zu Konstanten (Seite 102)
Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6.5 LDT (DATE_AND_LTIME)

Format

Der Datentyp LDT (DATE_AND_LTIME) speichert Angaben zu Datum und Uhrzeit in Nanosekunden seit dem 01.01.1970 0:0.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LDT:

Länge (Byte)	Format	Wertebereich	Beispiel für Werteingabe
8	Datum und Uhrzeit (Jahr-Monat-Tag-Stunde:Minute:Sekunde.Nanosekunde)	Min.: LDT#1970-01-01-00:00:00.000000000 Max.: LDT#2262-04-11-23:47:16.854775807	LDT#2008-10-25-08:12:34.567

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Übersicht über die Datentypkonvertierung (Seite 369)

Grundlagen zu Konstanten (Seite 102)

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.6.6 DTL

Beschreibung

Ein Operand vom Datentyp DTL hat eine Länge von 12 Byte und speichert Angaben zu Datum und Uhrzeit in einer vordefinierten Struktur.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DTL:

Länge (Byte)	Format	Wertebereich	Beispiel für Werteingabe
12	Datum und Uhrzeit (Jahr-Monat-Tag-Stunde:Minute:Sekunde.Nanosekunde)	Min.: DTL#1970-01-01-00:00:00.0 Max.: DTL#2262-04-11-23:47:16.854775807	DTL#2008-12-16-20:30:20.250

Die Struktur des Datentyps DTL setzt sich aus mehreren Komponenten zusammen, die jeweils einen unterschiedlichen Datentyp und Wertebereich aufweisen können. Der Datentyp eines angegebenen Werts muss dabei zum Datentyp der jeweiligen Komponente passen.

Hinweis

Ungültiger Beobachtungswert der DTL-Variablen im Hexadezimal Format

Wenn der Beobachtungswert der DTL-Variablen im Hexadezimal Format dargestellt wird, dann kann es daran liegen, dass einer der Werte (YEAR, MONTH, DAY, etc.) ungültig ist. Das ist z. B. der Fall, wenn an der Variablen HOUR ein Wert > 24 angegeben wurde.

Die folgende Tabelle zeigt die Strukturkomponenten des Datentyps DTL und deren Eigenschaften:

Byte	Komponente	Datentyp	Wertebereich
0	Jahr	UINT	1970 bis 2262
1			
2	Monat	USINT	1 bis 12
3	Tag	USINT	1 bis 31
4	Wochentag	USINT	1 (Sonntag) bis 7 (Samstag) Der Wochentag wird bei der Werteingabe nicht berücksichtigt.
5	Stunde	USINT	0 bis 23
6	Minute	USINT	0 bis 59
7	Sekunde	USINT	0 bis 59
8	Nanosekunde	UDINT	0 bis 999999999
9			
10			
11			

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.7 Zeichenfolgen

3.7.1 Zeichen

3.7.1.1 CHAR

Beschreibung

Eine Variable vom Datentyp CHAR (Character) hat eine Länge von 8 Bit und belegt ein BYTE im Speicher.

Der Datentyp CHAR speichert ein einziges Zeichen, das in ASCII-Kodierung abgelegt ist. Informationen zur Kodierung von Sonderzeichen finden Sie unter "Siehe auch > STRING".

Die folgende Tabelle zeigt den Wertebereich des Datentyps CHAR:

Länge (Bit)	Format	Wertebereich	Beispiel für Werteingaben
8	ASCII-Zeichen	ASCII-Zeichensatz	'A', CHAR#'A'

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

STRING (Seite 277)

3.7.1.2 WCHAR

Beschreibung

Eine Variable vom Datentyp WCHAR (Wide Characters) hat eine Länge von 16 Bit und belegt zwei BYTE im Speicher.

Der Datentyp WCHAR speichert ein einzelnes Zeichen eines erweiterten Zeichensatzes UFT-16 kodiert. Es wird jedoch nur eine Teilmenge des gesamten Unicodebereichs abgedeckt. Nicht darstellbare Zeichen werden mit Hilfe eines Fluchtsymbols darstellbar gemacht.

Die folgende Tabelle zeigt den Wertebereich des Datentyps WCHAR:

Länge (Bit)	Format	Wertebereich	Beispiel für Werteinga-be
16	Unicode	$0000 - $D7FF	WCHAR#'a'

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

3.7.2 Zeichenketten

3.7.2.1 STRING

Beschreibung

Ein Operand vom Datentyp STRING speichert mehrere Zeichen in einer Zeichenkette, die aus maximal 254 Zeichen bestehen kann. In einer Zeichenkette sind alle Zeichen der auf dem erstellenden System eingestellten Codepage zugelassen. Die Angabe der Zeichen erfolgt dabei in einfachen Anführungszeichen.

Eine Zeichenkette kann auch Sonderzeichen enthalten. Für die Angabe von Steuerungszeichen, Dollarzeichen und einfachen Anführungszeichen wird das Fluchtsymbol $ verwendet.

Hinweis

Unterschiedliche Codepages

Beachten Sie, dass die Sonderzeichen mittels der aktuell in Windows eingestellten Codepage kodiert werden. Das bedeutet, dass ein String, der Sonderzeichen enthält, auf einem anderen Betriebssystem, mit einer anderen Codepage, unterschiedlich dargestellt werden kann.

Die Abhängigkeit der Codepage vom erstellenden System erschwert eine internationale Verwendung des Anwenderprogramms. Nur die Zeichen aus der 7-Bit ASCII-Kodierung sind international gültig.

Die folgende Tabelle zeigt die Eigenschaften einer STRING-Variablen:

Länge (Byte)	Format	Wertebereich	Beispiel für Werteingabe
n + 2 [1]	ASCII-Zeichenkette einschließlich Sonderzeichen	0 bis 254 Zeichen	• 'Name' • STRING#'NAME' • STRING#'Na... (Die aktuelle Länge des Strings ist länger als der am Bildschirm zzur Verfügung stehende Platz.) • STRING#'' (Der String ist leer.)
[1] Ein Operand vom Datentyp STRING belegt im Speicher zwei Bytes mehr als die angegebene Maximallänge.			

Die folgende Tabelle zeigt Beispiele für die Notation von Sonderzeichen:

Zeichen	Hex	Bedeutung	Beispiel
$L oder $l	0A	Zeilenvorschub	'$LText', '$0AText'
$N	0A und 0D	Zeilenumbruch Der Zeilenumbruch belegt 2 Zeichen in der Zeichenkette und wird im Editor in der Anzeige zu RL umgewandelt.	'$NText', '$0A$0DText'
$P oder $p	0C	Seitenvorschub	'$PText', '$0CText'

Zeichen	Hex	Bedeutung	Beispiel
$R oder $r	0D	Wagenrücklauf (CR)	'$RText','$0DText'
$T oder $t	09	Tabulator	'$TText', '$09Text'
$$	24	Dollarzeichen	'100$$', '100$24'
$'	27	Einfaches Anführungszeichen	'$'Text$'','$27Text$27'

Folgt dem Fluchtsymbol $ ein Buchstabe aus der Tabelle, so wird das in der Tabelle angegebene Zeichen im String eingetragen. Folgt dem Fluchtsymbol $ ein Buchstabe, der nicht in der Tabelle aufgeführt ist, dann wird dieser im String eingetragen. Folgen dem Fluchtsymbol $ zwei Sedezimal(Hexadezimal)-Ziffern oder nur eine Ziffer, dann wird diese Kodierung im String eingetragen.

Verwendung in der Beobachtungstabelle

Bei einer CPU der Baureihen S7-300/400 gilt: Wird eine Variable vom Datentyp STRING beobachtet, dann werden nur die ersten 30 Zeichen dargestellt. Ist die tatsächliche Länge größer als 30 Zeichen, dann wird statt des schließenden Apostrophs (') eine Ellipse/ Auslassungszeichen (...) angezeigt. STRING-Werte mit mehr als 30 Zeichen können nicht zum Steuern verwendet werden.

Maximallänge einer Zeichenkette

Die Maximallänge der Zeichenkette kann bei der Deklaration eines Operanden mithilfe von eckigen Klammern nach dem Schlüsselwort STRING angegeben werden (z. B. STRING[4]). Sie können zur Deklaration der Maximallänge auch lokale oder globale Konstanten verwenden (z. B. STRING[#loc_const] oder STRING["glob_const"]. Wenn die Angabe der Maximallänge weggelassen wird, wird für den jeweiligen Operanden die Standardlänge von 254 Zeichen eingestellt.

Weitere Informationen zur Verwendung von lokalen oder globalen Konstanten zur Deklaration der Maximallänge finden Sie hier:

- Bausteinschnittstelle deklarieren: Variablen vom Datentyp STRING und WSTRING deklarieren (Seite 7280)
- Datenbausteine programmieren: Variablen vom Datentyp STRING deklarieren (Seite 7334)
- Beispiele zur Verwendung von Konstanten (Seite 106)

Wenn die tatsächliche Länge einer angegebenen Zeichenkette kürzer als die deklarierte Maximallänge ist, werden die Zeichen linksbündig in die Zeichenkette geschrieben, die restlichen Zeichenplätze bleiben undefiniert. Bei der Wertbearbeitung und allen Anzeigen werden nur belegte Zeichenplätze berücksichtigt, die die tatsächliche Länge des Strings festlegen.

Hinweis

Für S7-300/400 CPUs ist zu beachten: Wurde eine temporäre Variable vom Datentyp STRING definiert, muss das BYTE "Max. Länge des Strings" vor der Verwendung der Variablen im Anwenderprogramm mit der definierten Länge beschrieben werden.

STRING zur Parameterversorgung übergeben

Sie können den Datentyp STRING als Parameter übergeben. Weitere Informationen zur Parameterversorgung mit STRING finden Sie hier:

- Variable vom Datentyp STRING oder WSTRING übergeben (Seite 285)
- Für S7-1200/1500: Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)
- Für S7-300/400: Gültige Datentypen in der Bausteinschnittstelle (Seite 7266)

Beispiel

Das folgende Beispiel zeigt die Reihenfolge der Bytes bei der Angabe des Datentyps STRING[4] mit dem Ausgangswert 'AB':

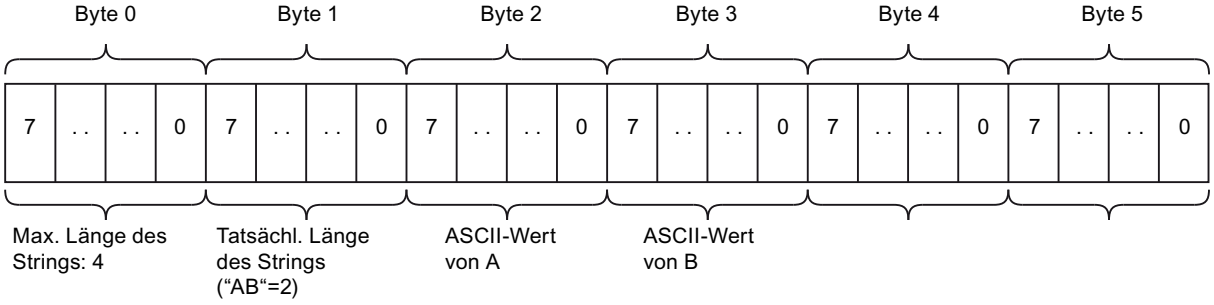

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.7.2.2 Aufbau einer STRING-Variablen

Einführung

Eine Variable mit Datentyp STRING (Zeichenkette) ist maximal 256 Zeichen lang mit 254 Bytes Nettodaten. Sie beginnt in einem nicht optimierten Baustein an einer Wortgrenze (an einem Byte mit gerader Adresse). In einem optimierten Baustein kann sie an jeder Byte-Grenze beginnen.

Beim Anlegen der Variablen wird deren maximale Länge festgelegt. Bei der Vorbelegung bzw. beim Bearbeiten der Zeichenkette wird die aktuelle Länge (die tatsächlich benutzte Länge der Zeichenkette = Anzahl der gültigen Zeichen) eingetragen. Im ersten Byte der Zeichenkette steht die maximale Länge. Im zweiten Byte steht die aktuelle Länge. Danach folgen die Zeichen kodiert nach der im Windows eingestellten Codepage.

Aufbau einer STRING-Variablen:

		Datentyp	Bereich	
Byte n[1]	maximale Länge	USINT	0 bis 254	(k)
Byte n+1	aktuelle Länge	USINT	0 bis 254	(m, m <= k)
Byte n+2	1. Zeichen	CHAR	aktuelle Länge (m)	maximale Länge (k)
Byte n+3	2. Zeichen	CHAR		
Byte ...	...	CHAR		
Byte n+m+1	m. Zeichen	CHAR		
Byte ...	...	CHAR		
Byte n+k+1	...	CHAR		

[1] n = gerade

3.7.2.3 WSTRING

Beschreibung

Ein Operand vom Datentyp WSTRING (Wide Character String) speichert mehrere Unicode-Zeichen vom Datentyp WCHAR in einer Zeichenkette. Wenn Sie keine Länge angeben, hat die Zeichenkette einen voreingestellten Wert von 254 Zeichen. In einer Zeichenkette sind alle vom Betriebssystem unterstützten Zeichen zugelassen. Dadurch ist es möglich, auch chinesische Zeichen innerhalb einer Zeichenkette zu verwenden. Windows unterstützt nur eine, jedoch ausreichende, Teilmenge der bei Unicode definierten Zeichen.

Hinweis

Kodierungen

Durch STEP 7 sind alle Kodierungen von $D000 bis $FFFF untersagt.

Bei der Deklaration eines Operanden vom Datentyp WSTRING können Sie mithilfe von eckigen Klammern seine Länge definieren (z. B. WSTRING[10]). Wenn Sie keine Längenangabe machen, dann wird die Länge des WSTRINGs auf 254 Zeichen eingestellt. Sie können eine Länge von maximal 16382 Zeichen deklarieren (WSTRING[16382]).

Die Angabe der Zeichen erfolgt in einfachen Anführungszeichen und immer mit der Kennung WSTRING#.

Die folgende Tabelle zeigt die Eigenschaften einer WSTRING-Variablen:

Länge (WORD)	Format	Wertebereich	Beispiel für Werteingabe
n + 2 [1]	Unicode-Zeichenkette; n gibt die Länge der Zeichenkette an.	Voreingestellter Wert: 0 bis 254 Zeichen Max. möglicher Wert: 0 bis 16382	• WSTRING#'Hallo Welt' • WSTRING#'Hallo We... (Die aktuelle Länge des Strings ist länger als der am Bildschirm zzur Verfügung stehende Platz.) • WSTRING#'' (Der String ist leer.)
[1] Ein Operand vom Datentyp WSTRING belegt im Speicher zwei WORD mehr als die angegebene Maximallänge.			

Eine Zeichenkette kann auch Sonderzeichen enthalten. Für die Angabe von Steuerungszeichen, Dollarzeichen und einfachen Anführungszeichen wird das Fluchtsymbol $ verwendet.

Die folgende Tabelle zeigt Beispiele für die Notation von Sonderzeichen:

Zeichen	Hex	Bedeutung	Beispiel
$L oder $l	000A	Zeilenvorschub	'$LText', '$000AText'
$N	000A und 000D	Zeilenumbruch Der Zeilenumbruch belegt 2 Zeichen in der Zeichenkette und wird im Editor in der Anzeige zu RL umgewandelt.	'$NText', '$000A $000DText'
$P oder $p	000C	Seitenvorschub	'$PText', '$000CText'
$R oder $r	000D	Wagenrücklauf (CR)	'$RText','$000DText'
$T oder $t	0009	Tabulator	'$TText', '$0009Text'
$$	0024	Dollarzeichen	'100$$', '100$0024'
$'	0027	Einfaches Anführungszeichen	'$'Text$'','$0027Text $0027'

Folgt dem Fluchtsymbol $ ein Buchstabe aus der Tabelle, so wird das in der Tabelle angegebene Zeichen im String eingetragen. Folgt dem Fluchtsymbol $ ein Buchstabe, der

nicht in der Tabelle aufgeführt ist, dann wird dieser im String eingetragen. Folgen dem Fluchtsymbol $ vier Sedezimal-Ziffern, dann wird diese Kodierung im String eingetragen.

Hinweis

Konvertierung von WSTRING-Variablen

Eine implizite Konvertierung des Datentyps WSTRING ist nicht möglich. Eine explizite Konvertierung des Datentyps WSTRING nach STRING ist generell möglich. Allerdings funktionieren in allen Windows Codepages standardmäßig nur die Konvertierungen der Zeichen, die im Coderaum 0 - 127 liegen. Bei allen darüberliegenden Zeichen müssen das Codepage Zeichen und das untere Byte des Unicode Zeichens an exakt der gleichen Stelle stehen, damit die Konvertierung fehlerfrei funktioniert.

Verwendung in der Beobachtungstabelle

Wird eine Variable vom Datentyp WSTRING beobachtet, dann werden nur die ersten 254 Zeichen dargestellt. Ist die tatsächliche Länge größer als 254 Zeichen, dann wird statt des schließenden Apostrophs (') eine Ellipse/Auslassungszeichen (...) angezeigt. WSTRING-Werte mit mehr als 254 Zeichen können nicht zum Steuern verwendet werden.

Maximallänge einer Zeichenkette

Die Maximallänge der Zeichenkette kann bei der Deklaration eines Operanden mithilfe von eckigen Klammern nach dem Schlüsselwort WSTRING angegeben werden (z. B. WSTRING[4]). Sie können zur Deklaration der Maximallänge auch lokale oder globale Konstanten verwenden (z. B. WSTRING[#loc_const] oder WSTRING["glob_const"]. Wenn die Angabe der Maximallänge weggelassen wird, wird für den jeweiligen Operanden die Standardlänge von 254 Zeichen eingestellt.

Weitere Informationen zur Verwendung von lokalen oder globalen Konstanten zur Deklaration der Maximallänge finden Sie hier:

- Bausteinschnittstelle deklarieren: Variablen vom Datentyp STRING und WSTRING deklarieren (Seite 7280)
- Datenbausteine programmieren: Variablen vom Datentyp STRING deklarieren (Seite 7334)
- Beispiele zur Verwendung von Konstanten (Seite 106)

Wenn die tatsächliche Länge einer angegebenen Zeichenkette kürzer als die deklarierte Maximallänge ist, werden die Zeichen linksbündig in die Zeichenkette geschrieben, die restlichen Zeichenplätze bleiben undefiniert. Bei der Wertbearbeitung werden nur belegte Zeichenplätze berücksichtigt.

WSTRING zur Parameterversorgung übergeben

Bei Bausteinen mit der Zugriffsart "optimiert" können Operanden vom Datentyp WSTRING bis zur maximalen Länge als Parameter übergeben werden.

Bei Funktionsbausteinen (FB) mit der Zugriffsart "standard" können Operanden vom Datentyp WSTRING in allen Abschnitten der Bausteinschnittstelle, außer im Abschnitt "InOut", als

Parameter deklariert werden. Bei einer Funktion (FC) mit der Zugriffsart "standard" können nur Operanden vom Datentyp WSTRING als Parameter übergeben werden.

Eine weitere Ausnahme bilden der Funktionswert eines FCs im Abschnitt "Return" der Bausteinschnittstelle und Ausdrücke in der Programmiersprache SCL. In diesen Fällen darf die WSTRING-Variable nicht länger als 1022 Zeichen sein.

Die deklarierten Längen von Formal- und Aktualparameter dürfen unterschiedlich sein. Weitere Informationen zur Parameterversorgung mit WSTRING finden Sie hier:

- Variable vom Datentyp STRING oder WSTRING übergeben (Seite 285)
- Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)

Beispiel

Das folgende Beispiel zeigt die Reihenfolge der Bytes bei der Angabe des Datentyps WSTRING[4] mit dem Ausgangswert 'AB':

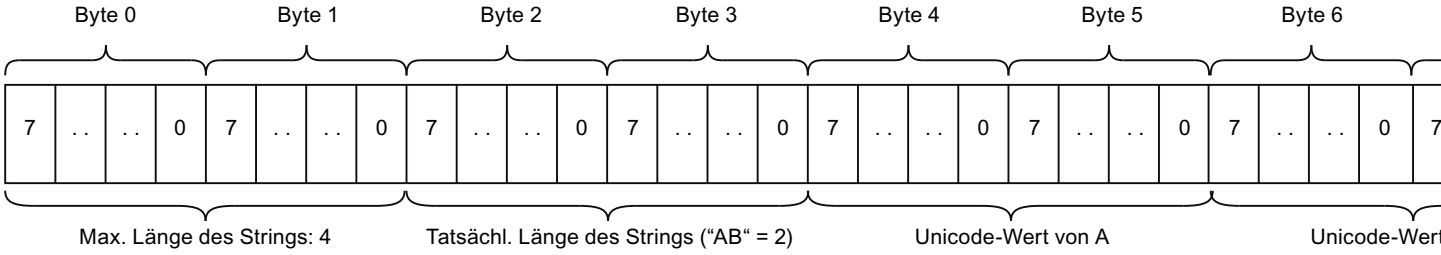

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Konstanten (Seite 102)

3.7.2.4 Einzelne Zeichen eines STRINGs oder WSTRINGs adressieren

Einzelne Zeichen eines STRINGs oder WSTRINGs adressieren

Sie greifen auf ein einzelnes Zeichen einer STRING- oder WSTRING-Variable über die Syntax `StringName[i]` zu. Der Zählindex "i" beginnt mit "1". Sie greifen also mit StringName[1] auf das erste Zeichen des Strings zu.

Auf einzelne Zeichen einer STRING- oder WSTRING-Konstante können Sie nicht zugreifen.

Beispiele

Die folgenden Beispiele zeigen die Adressierung von (W)STRINGs:

Adressierung	Erläuterung
"myDB".mySTRING[3]	Adressiert das dritte Zeichen des STRINGs im Datenbaustein.

Adressierung	Erläuterung
#myWSTRING[3]	Adressiert das dritte Zeichen des WSTRINGs.

Fehlerbehandlung bei (W)STRING-Zugriffen

Zugriffsfehler entstehen, wenn Sie zur Laufzeit auf ein Zeichen zugreifen, das außerhalb der STRING-Länge liegt. Bei lesendem Zugriff auf die Zeichenkette erhalten Sie das Zeichen '$00' oder '$0000', ein schreibender Zugriff in die Zeichenkette wird nicht ausgeführt. Wenn die Anweisung über den Freigabeausgang ENO verfügt, wird ENO auf den Signalzustand FALSE gesetzt. Die CPU geht nicht in STOP.

Eine Ausnahme besteht nur, wenn das Zeichen direkt hinter die tatsächliche Länge der Zeichenkette geschrieben wird.

Das folgende Beispiel zeigt den STRING 'Hello' mit der tatsächlichen Länge 5. Das 27. Zeichen des STRINGs liegt außerhalb der tatsächlichen Länge und kann deshalb nicht geschrieben werden. Der STRING bleibt unverändert, das Ergebnis der Zuweisung ist 'hello'.

SCL

```
MyDB.mystring := 'hello';
MyDB.mystring[27] := CHAR_TO_BYTE('!');
```

Das folgende Beispiel zeigt den genannten Ausnahmefall: Das Zeichen wird direkt hinter den STRING in das 6. Zeichen geschrieben. Das Ergebnis der Zuweisung ist 'hello!'.

SCL

```
MyDB.mystring := 'hello';
MyDB.mystring[6] := CHAR_TO_BYTE('!');
```

Verwenden Sie zur Hantierung von STRINGs wenn möglich die Anweisungen aus der Palette "Erweiterte Anweisungen > String + Char".

```
CONCAT(IN1 := 'hello', IN2 := '!');
```

3.7.2.5 Variable vom Datentyp STRING oder WSTRING übergeben

Beschreibung

Sie können Variablen vom Datentyp STRING oder WSTRING als Parameter übergeben. Die folgende Tabelle zeigt, welche Regeln bei der (W)STRING-Übergabe in den unterschiedlichen CPU-Familien gelten:

CPU-Familie	Datentyp	Regeln zur Übergabe beim Bausteinaufruf
S7-300/400	STRING	Die deklarierten Längen von Formal- und Aktualparameter müssen identisch sein.
S7-1200/1500	STRING WSTRING	Die deklarierten Längen von Formal- und Aktualparameter dürfen unterschiedlich sein. Wenn die deklarierte Länge des Zielparameters zur Laufzeit nicht ausreicht, um den (W)STRING aufzunehmen, wird der (W)STRING abgeschnitten und der Freigabeausgang ENO auf FALSE gesetzt. Im Programmiereditor wird durch ein graues Viereck am Parameter angezeigt, dass der (W)STRING zur Laufzeit möglicherweise abgeschnitten wird. Ausnahme: Beim Aufruf von AWL-Bausteinen müssen die deklarierten Längen von Formal- und Aktualparameter immer identisch sein.

Das folgende Bild zeigt einen Bausteinaufruf, bei dem sich die Längen der deklarierten Formal- und Aktualparameter unterscheiden. Aufgrund der unterschiedlichen deklarierten Längen könnten sowohl "Input_String_20" als auch "Output_String_10" zur Laufzeit abgeschnitten werden.

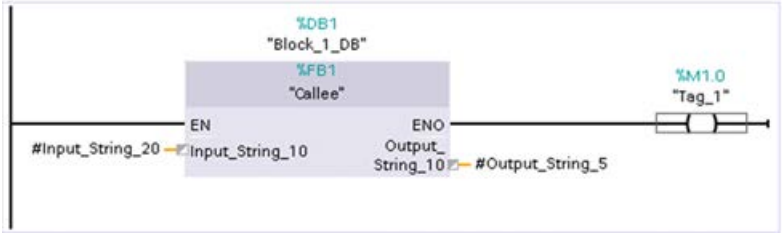

Siehe auch

Regeln zur Versorgung von Bausteinparametern (Seite 78)

3.8 PLC-Datentypen (UDT)

3.8.1 Grundlagen zu PLC-Datentypen (UDT)

Beschreibung

Ein PLC-Datentyp (UDT) ist ein zusammengesetzter anwenderdefinierter Datentyp, der für die Deklaration einer Variablen genutzt werden kann. Er repräsentiert eine Datenstruktur aus mehreren Komponenten unterschiedlicher Datentypen. Die Komponenten können auch von einem anderen PLC-Datentyp, von einem ARRAY oder direkt mit Hilfe des Schlüsselworts STRUCT als Struktur abgeleitet werden. Die Schachtelungstiefe ist dabei auf 8 Ebenen beschränkt.

Einen PLC-Datentyp (UDT) können Sie zentral ändern und im Programmcode mehrmals verwenden. Alle Verwendungsstellen werden automatisch aktualisiert.

Vorteile der PLC-Datentypen:

- Ein einfacher Datenaustausch über Bausteinschnittstellen zwischen mehreren Bausteinen
- Daten entsprechend der Prozesssteuerung gruppieren
- Parameter als eine Dateneinheit übergeben

Verwendung eines PLC-Datentyps

Sie können PLC-Datentypen als Typ für die Erstellung von Datenbausteinen angeben. Basierend auf diesem Typ können Sie eine Vielzahl von Datenbausteinen erzeugen, die die gleiche Datenstruktur haben. Diese Datenbausteine können durch die Eingabe von unterschiedlichen Aktualwerten für die entsprechende Aufgabe angepasst werden.

Erzeugen Sie z. B. einen PLC-Datentyp für ein Rezept zum Mischen von Farben. Sie können dann diesen Datentyp mehreren Datenbausteinen zuordnen, die jeweils andere Mengenangaben beinhalten.

Das folgende Bild zeigt diesen Anwendungsfall:

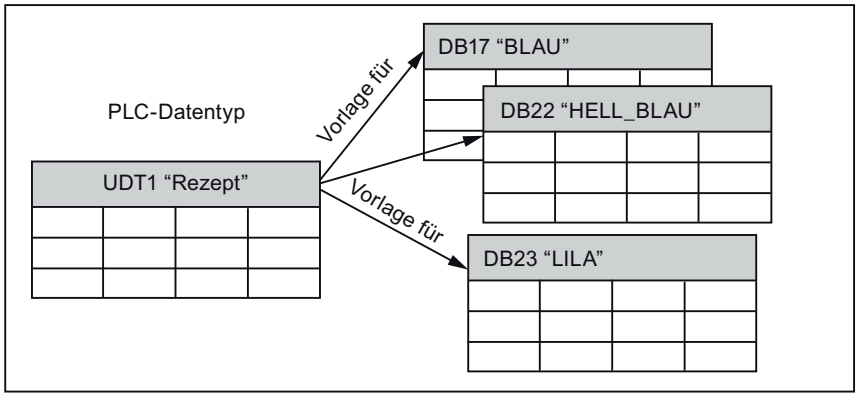

Es gibt folgende Verwendungsmöglichkeiten für PLC-Datentypen:

- Sie können sie als Datentypen für Variablen in der Variablendeklaration von Codebausteinen oder in Datenbausteinen verwenden.
- Sie können als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen.
- Sie können in S7-1200 und S7-1500 als Vorlage für die Erstellung von strukturierten PLC-Variablen genutzt werden.

Schachtelungstiefe und mögliche Anzahl von PLC-Datentypen

Es können max. 64k Objekte von FCs, FBs, DBs und PLC-Datentypen (für die CPUs ist die Gesamtmenge aber entsprechend eingeschränkt) definiert werden.

Für jede Strukturbeschreibung (z. B. ein DB oder ein PLC-Datentyp (UDT)) können maximal 16k Komponenten je Hierarchieebene definiert werden. Diese Komponenten können auch von einem ARRAY oder vom Datentyp STRUCT sein. Wenn hierbei eine strukturierte Komponente von einem explizit deklarierten PLC-Datentyp abgeleitet ist, dann gibt es keine Einschränkung. Es werden also je Hierarchieebene 16k Komponenten vom Datentyp PLC-Datentyp (UDT)/ SDT unterstützt. Von den 16k Komponenten dürfen jedoch maximal 252 "anonyme Strukturen" sein.

PLC-Datentypen können bis zu einer Tiefe von 8 Hierarchieebenen geschachtelt werden. Diese Schachtelungstiefe ist unabhängig von der eingesetzten CPU.

Nachteile von anonymen Strukturen

Die Komponenten strukturierter Variable können identisch adressiert werden. Dies ist unabhängig davon, ob die Deklaration mittels eines PLC-Datentyps oder einer anonymen Struktur erfolgt ist.

Es gibt folgende Nachteile bei der Nutzung anonymer Strukturen:

- Wiederverwendbarkeit derselben Struktur durch Kopieren. Das erschwert Änderungen der Struktur
- Anonyme Strukturen sind nicht kompatibel zu PLC-Datentypen (UDT) mit gleichem Aufbau
- Performance, da zur Laufzeit die Typgleichheit der Strukturkomponenten geprüft wird
- Die Obergrenze wird bei der Verwendung anonymer Strukturen leichter erreicht, da alle Komponenten einzeln ausgewertet werden müssen.

Beispiel

Das folgende Beispiel zeigt die Datentyp Definition des PLC-Datentyps "MyUDT":

myUDT

	Name	Datentyp
1	FirstEntry	Struct
2	Value1	Int
3	Value2	Real
4	Value3	"myOtherUDT"
5	Station1	Int
6	Station2	UInt
7	Station3	DInt

Systemerzeugte PLC-Datentypen

Einige Anweisungen erstellen beim Instanziieren eigene PLC-Datentypen, die im Projektordner "PLC-Datentypen" gespeichert werden. Verwenden Sie diese vom System erzeugten PLC-Datentypen nicht in einer Bibliothek, da dies zu unerwünschtem Systemverhalten führen kann. Bei Bedarf werden diese PLC-Datentypen vom System erzeugt, die Ablage in einer Bibliothek ist somit nicht erforderlich.

Externe Quellen aus Bausteinen generieren

Beim Generieren von externen Quellen aus Bausteinen werden die direkt in der Bausteinschnittstelle durchgeführten Änderungen an voreingestellten Werten von PLC-Datentypen nicht in die Quellen exportiert. Dadurch stehen diese Werte nach einem erneuten Import der Quellen nicht mehr zur Verfügung. Stattdessen werden wieder die voreingestellten Werte übernommen. Um diesen Datenverlust der geänderten voreingestellten Werte zu verhindern, müssen die Änderungen direkt im PLC-Datentyp vorgenommen werden und nicht in der Bausteinschnittstelle. Dann werden die Änderungen beim Generieren von externen Quellen mit exportiert.

Weitere Informationen zu den PLC-Datentypen (UDTs)

Hinweis

PLC-Datentypen (UDTs) deklarieren:

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

Programmierempfehlungen zu den PLC-Datentypen:

PLC-Datentypen (UDT) verwenden (Seite 231)

Datentyp DB_ANY verwenden (Seite 215)

Vergleich von PLC-Datentypen (UDTs) im Programm:

PLC-Datentypen vergleichen (Seite 7955)

Hinweis

Weitere Informationen zu den PLC-Datentypen im Siemens Industry Online Support finden Sie in folgenden Beiträgen:

Wie können Sie in STEP 7 (TIA Portal) Strukturen in optimierten Speicherbereichen bei der S7-1500 initialisieren?

https://support.industry.siemens.com/cs/ww/de/view/78678760 (https://support.industry.siemens.com/cs/ww/de/view/78678760)

Wie legen Sie bei einer S7-1500 Steuerung einen PLC-Datentyp (UDT) an?

https://support.industry.siemens.com/cs/ww/de/view/67599090 (https://support.industry.siemens.com/cs/ww/de/view/67599090)

Wie erfolgt in STEP 7 (TIA Portal) die gezielte Anwendung eigener PLC-Datentypen (UDT)?

https://support.industry.siemens.com/cs/de/de/view/67582844 (https://support.industry.siemens.com/cs/de/de/view/67582844)

Warum sollten für die S7-1500 beim Bausteinaufruf ganze Strukturen übergeben werden, anstelle vieler Einzelkomponenten?

https://support.industry.siemens.com/cs/ww/de/view/67585079 (https://support.industry.siemens.com/cs/ww/de/view/67585079)

Siehe auch

PLC-Datentypen (UDT) adressieren (Seite 290)

Variablen basierend auf einem PLC-Datentyp deklarieren (Seite 7281)

Variablen basierend auf einem PLC-Datentyp deklarieren (Seite 7335)

Vergleich von PLC-Variablen (Seite 7920)

Strukturierte PLC-Variablen anlegen (Seite 7389)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

CMP ==: Gleich (Seite 712)

Variable vom Datentyp PLC-Datentyp (UDT) übergeben (Seite 289)

3.8.2 Variable vom Datentyp PLC-Datentyp (UDT) übergeben

Beschreibung

Sie können auch Variablen, die als PLC-Datentyp (UDT) deklariert sind, als Aktualparameter übergeben. Wenn der Formalparameter in der Variablendeklaration als PLC-Datentyp (UDT) deklariert ist, müssen Sie eine Variable als Aktualparameter übergeben, die den gleichen PLC-Datentyp (UDT) hat.

Auch ein Element einer mittels PLC-Datentyp (UDT) deklarierten Variablen kann beim Bausteinaufruf als Aktualparameter übergeben werden, sofern der Datentyp des Elements der Variablen dem Datentyp des Formalparameters entspricht.

Strukturen und anwenderdefinierte PLC-Datentypen sind einander zuweisbar, wenn sie einen identischen Aufbau haben. Das bedeutet, dass die Datentypen und die Reihenfolge aller Strukturkomponenten identisch sein müssen. In geschachtelten Strukturen müssen auch die Datentypen und die Reihenfolge der unterlagerten Strukturen und ihrer Komponenten übereinstimmen.

Auch zwei unterschiedliche PLC-Datentypen sind einander zuweisbar, wenn die Datentypen und die Reihenfolge aller Strukturkomponenten, einschließlich der unterlagerten Strukturen, identisch sind. Die Namen der PLC-Datentypen müssen nicht übereinstimmen.

Anwenderdefinierte PLC-Datentypen bzw. Strukturen können nicht auf Systemdatentypen zugewiesen werden.

Siehe auch

Regeln zur Versorgung von Bausteinparametern (Seite 78)

3.8.3 PLC-Datentypen (UDT) adressieren

Datenelemente eines PLC-Datentyps adressieren

Für den Zugriff auf Elemente eines PLC-Datentyps wird die folgende Syntax verwendet:

```
<PLCDatentypName>.<Elementname>
```

Beispiele

Die folgenden Beispiele zeigen die Adressierung von Variablen strukturierten Datentyps:

Adressierung	Erläuterung
Values.Temperature	Adressierung des Elements "Temperature" in der Variable "Values", die auf einem PLC-Datentyp basiert.

Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 111)

3.9 Datenstruktur STRUCT (Anonyme Strukturen)

3.9.1 Grundlagen zu STRUCT

Beschreibung

Der Datentyp STRUCT repräsentiert eine Datenstruktur, die sich aus einer festen Anzahl von Komponenten unterschiedlicher Datentypen zusammensetzt. Auch Komponenten vom Datentyp STRUCT oder ARRAY können in einer Struktur geschachtelt werden. Mithilfe von Strukturen können Daten entsprechend der Prozesssteuerung gruppiert und Parameter als eine Dateneinheit übergeben werden.

Struktur-Deklarationen, die direkt an der Variable verwendet werden, werden anonyme Strukturen genannt. Eine anonyme Struktur kann folgendermaßen aussehen:

Global_DB

	Name	Data type
1	Static	
2	Porsche	Struct
3	Motor	Struct
4	Valves	Array[0..5] of Bool
5	Speed_Min	Byte
6	Speed_Max	Byte
7	Temperature	Word
8	Axes	Array[0..4] of Bool
9	Oil	Byte
10	Carriage	Struct
11	Tuning	Struct
12	BMW	Struct
13	Audi	Struct

Alle nachfolgenden Erklärungen beziehen sich auf diesen Strukturaufbau.

Schachtelungstiefe und mögliche Anzahl für Strukturen

Eine geschachtelte Struktur enthält mindestens eine weitere Struktur als Komponente. Es können jeweils 65535 FCs, FBs, DBs und PLC-Datentypen definiert werden. Für jede CPU gilt bezüglich der Bausteintypen und der Summe aller Bausteine eine gesonderte Obergrenze.

Für jede Strukturbeschreibung (z. B. ein DB oder ein PLC-Datentyp (UDT)) können maximal 16k Komponenten je Hierarchieebene definiert werden. Diese Komponenten können auch von einem ARRAY oder vom Datentyp STRUCT sein. Wenn hierbei eine strukturierte Komponen te von einem explizit deklarierten PLC-Datentyp abgeleitet ist, dann gibt es keine Einschränkung. Es werden pro Hierarchieebene 16k Komponenten vom Datentyp PLC-Datentyp (UDT)/SDT unterstützt. Innerhalb eines Bausteins dürfen jedoch maximal 252 "anonyme Strukturen" sein.

Strukturen (STRUCT) können bis zu einer Tiefe von 8 Hierarchieebenen geschachtelt werden. Diese Schachtelungstiefe ist unabhängig von der eingesetzten CPU.

STRUCT zur Parameterversorgung übergeben

Sie können den Datentyp STRUCT als Parameter übergeben. Weitere Informationen zur Parameterversorgung mit STRUCT finden Sie hier:

Variablen vom Datentyp STRUCT übergeben (Seite 295)

Nachteile von anonymen Strukturen

Die Komponenten strukturierter Variable können identisch adressiert werden. Dies ist unabhängig davon, ob die Deklaration mittels eines PLC-Datentyps oder einer anonymen Struktur erfolgt ist.

Es gibt folgende Nachteile bei der Nutzung anonymer Strukturen:

- Erhöhter Pflegeaufwand: Wurde eine anonyme Struktur mehrfach kopiert, dann muss sie bei einer Änderung auch mehrfach geändert werden.
- Anonyme Strukturen sind nicht kompatibel zu PLC-Datentypen (UDT) mit gleichem Aufbau
- Performance, da die Typgleichheit aller Strukturkomponenten geprüft wird
- Erhöhter Speicherbedarf: Jede anonyme Struktur ist ein eigenständiges Objekt, deren Beschreibung auf das AS geladen wird.

Beispiel

Wenn Sie die Variable vom Datentyp STRUCT in einem PLC-Datentyp (UDT) deklarieren, dann stehen Ihnen mehr Verwendungsmöglichkeiten zur Verfügung (siehe linkes Bild). Sie können die Variable aber auch direkt mit dem Datentyp STRUCT deklarieren (siehe rechtes Bild).

Deklaration der strukturierten Variable "Motor" mit bzw. ohne PLC-Datentyp (UDT):

Strukturierte Variable mit PLC-Datentyp (UDT)	Strukturierte Variable ohne PLC-Datentyp (UDT)

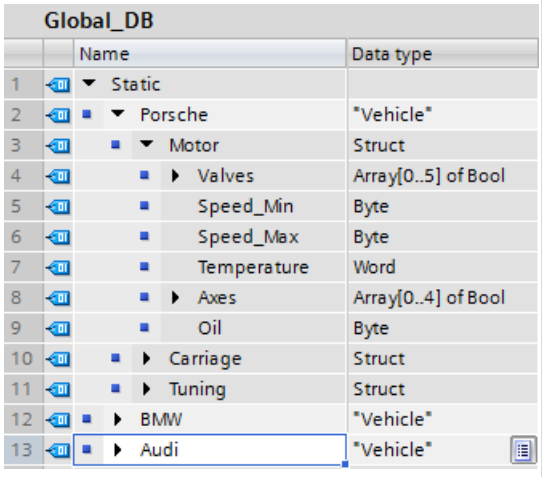

Global_DB

	Name	Data type
1	Static	
2	Porsche	"Vehicle"
3	Motor	Struct
4	Valves	Array[0..5] of Bool
5	Speed_Min	Byte
6	Speed_Max	Byte
7	Temperature	Word
8	Axes	Array[0..4] of Bool
9	Oil	Byte
10	Carriage	Struct
11	Tuning	Struct
12	BMW	"Vehicle"
13	Audi	"Vehicle"

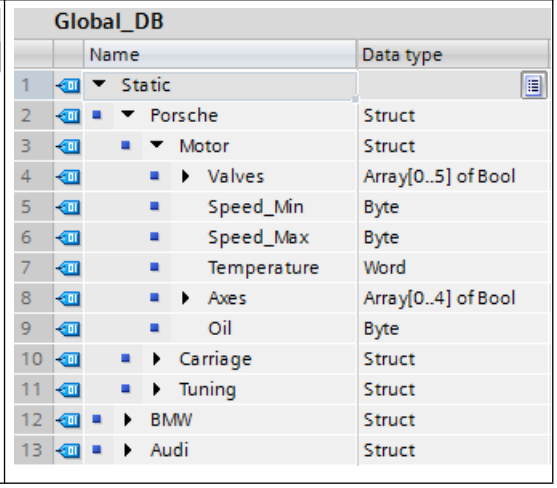

Global_DB

	Name	Data type
1	Static	
2	Porsche	Struct
3	Motor	Struct
4	Valves	Array[0..5] of Bool
5	Speed_Min	Byte
6	Speed_Max	Byte
7	Temperature	Word
8	Axes	Array[0..4] of Bool
9	Oil	Byte
10	Carriage	Struct
11	Tuning	Struct
12	BMW	Struct
13	Audi	Struct

Verarbeitung komplexer Strukturen

Eine Reihe von Anweisungen (z. B. "Serialize: Serialisieren", "Deserialize: Deserialisieren", "CMP" (Vergleicher) und "MOVE: Wert kopieren") können sehr große und komplex strukturierte Variablen verarbeiten. Hierbei analysiert die CPU den Aufbau der Variablenstruktur und führt für jede in der Gesamtstruktur enthaltene Substruktur bzw. für alle enthaltenen elementaren Komponenten die entsprechende Anweisung aus.

Bei sehr komplexem Aufbau kann diese Strukturanalyse unter Umständen zu einer unerwarteten Erhöhung der Laufzeit der entsprechenden Anweisung führen. Neben der Komplexität der strukturierten Variablen, die an der Operation angegeben wird, hat auch die Gesamtanzahl der im Programm deklarierten anonymen Strukturen einen Einfluss auf die Laufzeit. Eine sehr große Anzahl unterschiedlicher anonymer Strukturdefinitionen kann die Laufzeit zusätzlich erhöhen.

Abhilfe:

- Vermeiden Sie anonyme Strukturen. Setzen Sie stattdessen PLC-Datentypen ein.
- Vermeiden Sie die mehrfache Deklaration von sehr ähnlich aufgebauten Datenstrukturen. Versuchen Sie, diese in einer Strukturdeklaration zusammenzufassen.
- Vermeiden Sie die Deklaration von vielen Einzelvariablen in Strukturen und Datenbausteinen, wenn diese den gleichen Datentyp aufweisen. Verwenden Sie in diesem Fall nach Möglichkeit der Datentyp ARRAY.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

3.9.2 Aufbau einer STRUCT-Variablen

Einführung

Im nicht optimierten Baustein beginnt eine STRUCT-Variable immer an einer Wortgrenze, d. h. an einem Byte mit gerader Adresse. Anschließend liegen dann die einzelnen Komponenten in der Reihenfolge ihrer Deklaration im Speicher. STRUCT-Variablen belegen den Speicher bis zur nächsten Wortgrenze.

Komponenten mit Datentyp BOOL beginnen im niederwertigsten Bit des nächsten Bytes. Komponenten mit Datentyp BYTE und CHAR beginnen im nächsten BYTE. Komponenten mit anderen Datentypen beginnen an einer Wortgrenze. In optimierten Bausteinen werden Strukturen so abgelegt, dass alle Komponenten entsprechend ihrer Breite ausgerichtet sind. Die Reihenfolge im Speicher entspricht nicht der Darstellung im Editor.

Eine geschachtelte Struktur ist eine Struktur als Komponente einer anderen Struktur. Es ist eine Schachtelungstiefe von maximal 8 Strukturen möglich. Im Abschnitt "InOut" ist eine Schachtelungstiefe von 9 Strukturen möglich. Die Schachtelungstiefe berechnet sich einschließlich des obersten Strukturelements (Beispiel: "Anna.Berta.Carla" entspricht einer Schachtelungstiefe von 3).

Alle Komponenten können einzeln angesprochen werden. Die einzelnen Namen der Komponenten sind durch einen Punkt getrennt.

Aufbau einer STRUCT-Variablen mit elementaren und zusammengesetzten Datentypen im nicht optimierten Speicherbereich:

- Deklaration der STRUCT-Variable "Motor"

Global_DB

	Name	Data type
1	Static	
2	Porsche	Struct
3	Motor	Struct
4	Valves	Array[0..5] of Bool
5	Speed_Min	Byte
6	Speed_Max	Byte
7	Temperature	Word
8	Axes	Array[0..4] of Bool
9	Oil	Byte
10	Carriage	Struct
11	Tuning	Struct
12	BMW	Struct
13	Audi	Struct

- Speicherbedarf der STRUCT-Variable "Motor"

	7	6	5	4	3	2	1	0	
Byte n[1]	...	...	5	4	3	2	1	0	Valves (Bit-Komponenten und Füll-Bits)
Byte n+1	Byte 1								Speed_Min (Byte-Komponente)
Byte n+2	Byte 2								Speed_Max (Byte-Komponente)
Byte n+3	(Füllbyte)								
Byte n+4	Wort 1								Temperature (Wort-Komponente)
Byte n+5									
Byte n+6	...	...	...	4	3	2	1	0	Axes (Bit-Komponenten und Füll-Bits)
Byte n+7	Byte 3								Oil (Byte-Komponente)

[1] n = gerade

3.9.3 STRUCT-Komponenten adressieren

Komponenten in Strukturen adressieren

Sie greifen auf die einzelnen Komponenten einer Struktur über die folgende Syntax zu:

```
"DBname".name1.name2.name3
```

Beispiel

Das folgende Beispiel zeigt die Adressierung von Variablen strukturierten Datentyps:

Adressierung	Erläuterung
"Global_DB".Motor.Valves[2]	Adressierung des 3. Ventils des Elements "Valves" in der Struktur Motor des globalen Datenbausteins "Global_DB".

3.9.4 Variablen vom Datentyp STRUCT übergeben

Beschreibung

Sie können einzelne Komponenten einer STRUCT-Variablen als Aktualparameter übergeben, wenn die Komponente dem Datentyp des Formalparameters entspricht.

Sie können auch komplette Strukturen als Parameter übergeben. Wenn ein Baustein einen Eingangsparameter vom Datentyp STRUCT hat, müssen Sie als Aktualparameter ein STRUCT mit identischem Aufbau übergeben. Das bedeutet, dass die Namen und Datentypen aller Strukturkomponenten identisch sein müssen.

Strukturen und anwenderdefinierte PLC-Datentypen sind einander zuweisbar, wenn sie einen identischen Aufbau haben. Das bedeutet, dass die Datentypen und die Reihenfolge aller Strukturkomponenten identisch sein müssen. In geschachtelten Strukturen müssen auch die Datentypen und die Reihenfolge der unterlagerten Strukturen und ihrer Komponenten übereinstimmen.

Auch zwei unterschiedliche PLC-Datentypen sind einander zuweisbar, wenn die Datentypen und die Reihenfolge aller Strukturkomponenten, einschließlich der unterlagerten Strukturen, identisch sind. Die Namen der PLC-Datentypen müssen nicht übereinstimmen.

Anwenderdefinierte PLC-Datentypen bzw. Strukturen können nicht auf Systemdatentypen zugewiesen werden.

Hinweis

Wir empfehlen, Strukturen als PLC-Datentypen (UDT) zu programmieren. PLC-Datentypen (UDT) erleichtern die Programmierung, da sie mehrfach verwendet und zentral geändert werden können.

Siehe auch

Regeln zur Versorgung von Bausteinparametern (Seite 78)

3.10 ARRAY

3.10.1 Grundlagen zum ARRAY

Beschreibung

Eine Variable vom Datentyp ARRAY repräsentiert eine Datenstruktur, die aus einer festen Anzahl von Komponenten des gleichen Datentyps besteht. Für die Komponenten sind alle Datentypen außer ARRAY zugelassen.

Beim Anlegen einer ARRAY-Variablen werden die Indexgrenzen in eckigen Klammern und der Datentyp nach dem Schlüsselwort "of" definiert. Die ARRAY-Grenzen können entweder fest mithilfe von Ganzzahlen oder globalen und lokalen Konstanten oder als Formalparameter eines Bausteins auch variabel mithilfe von ARRAY[*] definiert werden. Der untere Grenzwert muss kleiner oder gleich dem oberen Grenzwert sein. Ein ARRAY kann bis zu sechs Dimensionen enthalten, deren Grenzen durch je ein Komma getrennt angegeben werden.

Die Struktur einer eindimensionalen Variable vom Datentyp ARRAY sieht z. B. folgendermaßen aus:

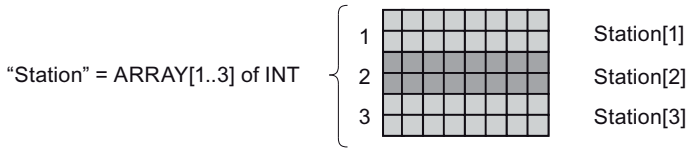

Die maximalen ARRAY-Grenzen hängen von folgenden Faktoren ab:

- Datentyp der ARRAY-Komponente
- Speicherreserve (nur in Bausteinen mit optimiertem Zugriff)
  Weitere Informationen dazu finden Sie unter: Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889).
- Maximale Größe des Datenbausteins
- Maximale Speicherkapazität der CPU (Weitere Informationen finden Sie im jeweiligen Gerätehandbuch)

Die folgende Tabelle zeigt die Eigenschaften des Datentyps ARRAY:

Bausteineigenschaft	Format	ARRAY-Grenzen	Datentyp
Standard	ARRAY[unterer Grenzwert..oberer Grenzwert] of <Datentyp>	[-32 768..32 767] of <Datentyp>	Alle Datentypen außer ARRAY Multiinstanzen
Optimiert		[-2 147 483 648..2 147 483 647] of <Datentyp>	

Je nachdem, wo Sie das ARRAY deklariert haben, gibt es verschiedene Möglichkeiten, die ARRAY-Grenzen zu definieren:

ARRAY-Grenzen	Globaler Datenbaustein / Variablentabelle	Bausteinschnittstelle
Verwendung von Ganzzahlen als feste ARRAY-Grenzen	x	x
Verwendung von globalen Konstanten als feste ARRAY-Grenzen	x	x
Verwendung von lokalen Konstanten als feste ARRAY-Grenzen	-	x
Verwendung von variablen ARRAY-Grenzen ARRAY[*]	-	x
ARRAY von Multiinstanzen	-	x

Weitere Informationen, im welchem Abschnitt der Bausteinschnittstelle Sie ein ARRAY oder ein ARRAY[*] definieren können, finden Sie hier: Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)

Hinweis

Gültig für CPUs der Baureihe S7-1500

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" benötigt ein Element vom Datentyp BOOL einen Speicherplatz von 1 Byte. Das gilt auch, wenn Sie ein ARRAY of <Datentyp> verwenden. So benötigt z. B. ein ARRAY[0..1] of BOOL in einem optimierten Programmbaustein 2 Byte.

Verwendung von Ganzzahlen als feste ARRAY-Grenzen

Sie können Ganzzahlen als feste ARRAY-Grenzen verwenden:

- Beispiel für ein eindimensionales ARRAY

Static	
ValueArray	Array[0..2] of Int
ValueArray[0]	Int
ValueArray[1]	Int
ValueArray[2]	Int

- Beispiel für ein zweidimensionales ARRAY

Static	
ValueArray	Array[0..2, 0..2] of Int
ValueArray[0,0]	Int
ValueArray[0,1]	Int
ValueArray[0,2]	Int
ValueArray[1,0]	Int
ValueArray[1,1]	Int
ValueArray[1,2]	Int
ValueArray[2,0]	Int
ValueArray[2,1]	Int
ValueArray[2,2]	Int

Verwendung von Konstanten als feste ARRAY-Grenzen

Sie können lokale oder globale Konstanten als feste ARRAY-Grenzen verwenden:

- Feste ARRAY-Grenzen für ein eindimensionales ARRAY, bestehend aus zwei globalen Anwenderkonstanten

Static	
ValueArray	Array["a".."n"] of Int
ValueArray[0]	Int
ValueArray[1]	Int
ValueArray[2]	Int

- Feste ARRAY-Grenzen für ein eindimensionales ARRAY, bestehend aus zwei lokalen Anwenderkonstanten

Input		
ValueArray	Array[#b..#d] of Int	
ValueArray[1]	Int	0
ValueArray[2]	Int	0
ValueArray[3]	Int	0
ValueArray[4]	Int	0
ValueArray[5]	Int	0
Output		
InOut		
Static		
Temp		
Constant		
b	Int	1
d	Int	5

- Feste ARRAY-Grenzen für ein zweidimensionales ARRAY, bestehend aus globalen und lokalen Anwenderkonstanten

Input		
ValueArray	Array["a"..#b, "n"..#d] of Int	
Output		
InOut		
Static		
Temp		
Constant		
b	Int	1
d	Int	5

Weitere Informationen zur Erstellung und Verwendung von Konstanten als ARRAY-Grenzen finden Sie hier:

- Lokale Variablen und Konstanten in der Bausteinschnittstelle deklarieren (Seite 7274)
- Beispiele zur Verwendung von Konstanten (Seite 106)

Verwendung von Ganzzahlen und globalen und lokalen Konstanten als feste ARRAY-Grenzen

Zusätzlich können Sie alle drei oben genannten Möglichkeiten auch miteinander kombinieren und als ARRAY-Grenzen auch Ganzzahlen und globale und lokale Konstanten gemischt innerhalb eines ARRAYs verwenden:

Input		
▸ ValueArray	Array["a"..#b, 2..#d, 1.."n"] of Int	
▸ Output		
▸ InOut		
▸ Static		
▸ Temp		
Constant		
b	Int	1
d	Int	5

Verwendung von variablen ARRAY-Grenzen ARRAY[*] (nur in Bausteinen mit optimiertem Zugriff)

Mithilfe von ARRAY[*] können Sie Bausteine programmieren, die ein ARRAY mit flexibler Länge verarbeiten können. Hierzu deklarieren Sie variable ARRAY-Grenzen an den Parametern einer Funktion oder eines Funktionsbausteins. ARRAY[*] steht Ihnen in allen Programmiersprachen zur Verfügung.

Mithilfe der Anweisungen "LOWER_BOUND" und "UPPER_BOUND" können Sie zur Laufzeit jeweils die untere bzw. obere Grenze eines ARRAYs mit variablen Grenzen auslesen.

Hinweis

Verfügbarkeit von ARRAY[*]

ARRAY[*] steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 in optimierten Bausteinen zur Verfügung. In Funktionen (FC) können Sie ARRAY[*] in allen Deklarationsabschnitten nutzen. In Funktionsbausteinen (FB) können Sie ARRAY[*] nur als Durchgangsparameter im Abschnitt "InOut" deklarieren.

- Variable ARRAY-Grenzen für ein eindimensionales ARRAY:

Input	
Output	
InOut	
ValueArrayVariable	Array[*] of Int
ValueArrayVariable[*]	Int
Static	
Temp	
Constant	

- Variable ARRAY-Grenzen für ein dreidimensionales ARRAY:

Input	
Output	
InOut	
ValueArrayVariable	Array[*, *, *] of Int
ValueArrayVariable[*, *, *]	Int
Static	
Temp	
Constant	

Weitere Informationen zur Verwendung von ARRAY[*] finden Sie hier:

- Variablen vom Datentyp ARRAY / ARRAY[*] übergeben (Seite 305)
- Variablen vom Datentyp ARRAY deklarieren (Seite 7277) in der Bausteinschnittstelle

Verwendung von Multiinstanzen (nur in Bausteinen mit optimiertem Zugriff)

Sie können Multiinstanzen auch als ARRAY anlegen, um z. B. bei der Bearbeitung von Programmschleifen mithilfe eines variablen Index die einzelnen Multiinstanzen adressieren zu können:

Static	
MultiinstanzArray	Array[0..1] of "SCL_FB_1500"
MultiinstanzArray[0]	"SCL_FB_1500"
Input	
Tag_In	Bool
PresetTime	Time
Output	
InOut	
Static	
MultiinstanzArray[1]	"SCL_FB_1500"

Weitere Informationen zur Erstellung und Verwendung von Multiinstanzen finden Sie hier:

- Multiinstanzen (Seite 67)
- Multiinstanzen deklarieren (Seite 7282)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

ARRAY-Komponenten indirekt adressieren (Seite 211)

3.10.2 Aufbau einer ARRAY-Variablen

Einführung

Eine ARRAY-Variable beginnt bei nicht optimierten Bausteinen nimmer an einer Wortgrenze, d. h. an einem Byte mit gerader Adresse. ARRAY-Variablen belegen den Speicher bis zur nächsten Wortgrenze.

Bei optimierten Bausteinen benötigt das ARRAY so Platz, wie das breiteste Element der Struktur. Das bedeutet z. B., dass ein ARRAY of BYTE auf einer Byte-Grenze liegt und ein ARRAY of LREAL auf einer 8-Byte-Grenze.

Aufbau einer ARRAY-Variablen im eindimensionalen Feld:

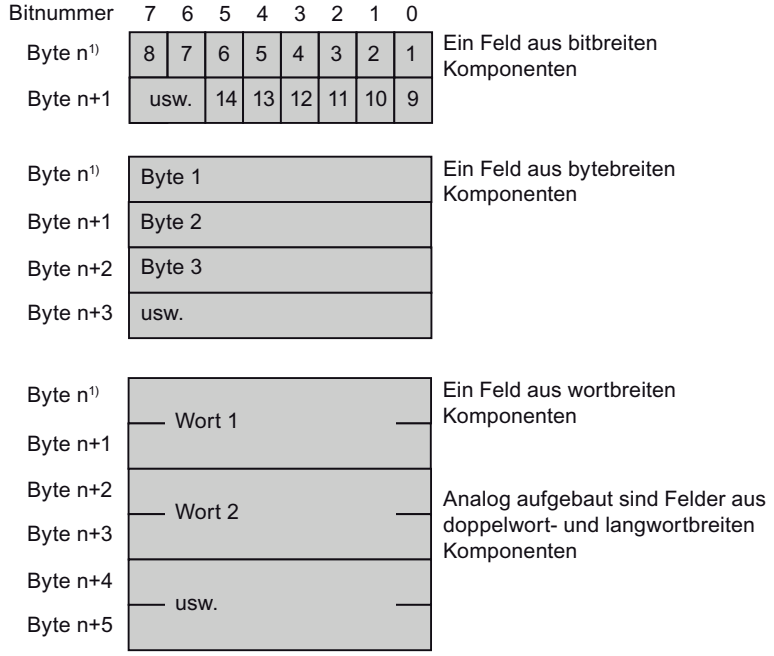

1) n = gerade

Komponenten mit Datentyp BOOL beginnen im niederwertigsten Bit. Komponenten mit Datentyp BYTE und CHAR beginnen im rechten Byte. Die einzelnen Komponenten werden der Reihe nach aufgelistet.

Aufbau einer ARRAY-Variablen im mehrdimensionalen Feld:

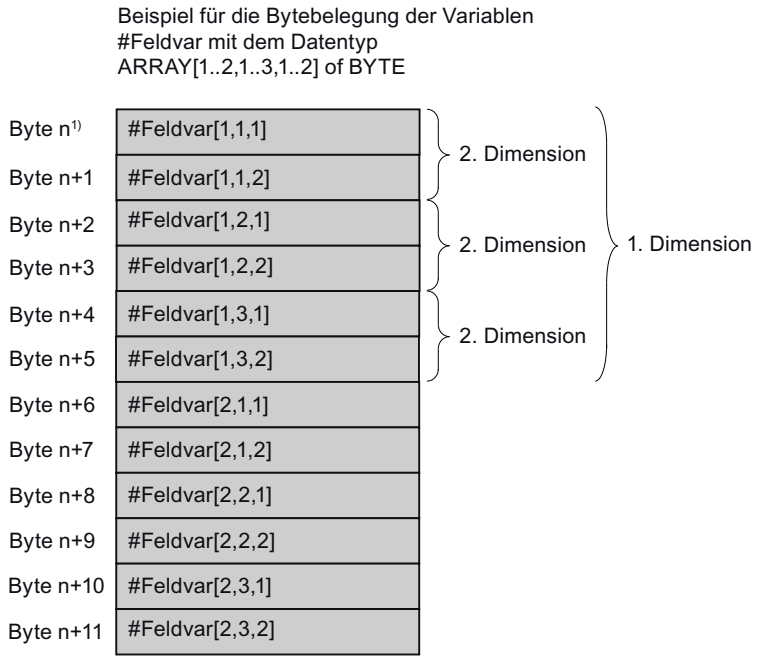

In mehrdimensionalen Feldern werden die Komponenten zeilenweise (dimensionsweise) abgelegt, beginnend mit der ersten Dimension. Eine neue Dimension beginnt bei Bit- und Byte-Komponenten immer im nächsten Byte. Bei Komponenten anderer Datentypen beginnt sie in nicht optimierten Bausteinen immer im nächsten Wort (im nächsten geraden Byte).

Bei optimierten Bausteinen benötigt jede Dimension so viel Platz wie die erste Dimension.

3.10.3 ARRAY-Komponenten adressieren

Einführung

ARRAY-Komponenten werden mittels festem oder variablem Index adressiert. Die Komponenten einer ARRAY-Variablen können wie Variablen des gleichen Datentyps behandelt werden.

In der eckigen Klammer wird der Index der Komponente angegeben. Der Index enthält pro ARRAY-Dimension einen ganzzahligen Wert (fester Index) oder eine Variable (variabler Index).

Adressierung von ARRAY-Komponenten mittels festem Index

Die Adressierung einer ARRAY-Komponente mittels festem Index sieht folgendermaßen aus:

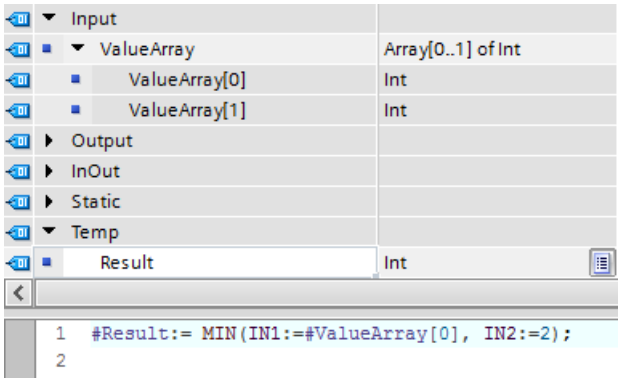

Adressierung von ARRAY-Komponenten mittels variablem Index

Eine ARRAY-Komponente kann auch mit einer Variablen adressiert werden, deren Wert erst zur Laufzeit berechnet wird. Die Variable kann eine absolut oder symbolisch adressierte globale oder lokale Variable vom Datentyp Ganzzahl sein. Diese Adressierung ist auch bei mehrdimensionalen ARRAYs und bei der Adressierung von Teilfeldern möglich. (<ArrayName>[i,j,k...])

Bei einer mittels variablem Index adressierten ARRAY-Komponente, die als Aktualparameter an einem Durchgangsparameter angelegt ist, hat eine Änderung der Variablen im aufgerufenen Baustein keine Wirkung. Der Wert wird in die beim Aufruf übergebene ARRAY-Komponente zurückgeschrieben, aus der er gelesen wurde.

Die Adressierung einer ARRAY-Komponente mithilfe einer Index-Variable sieht folgendermaßen aus:

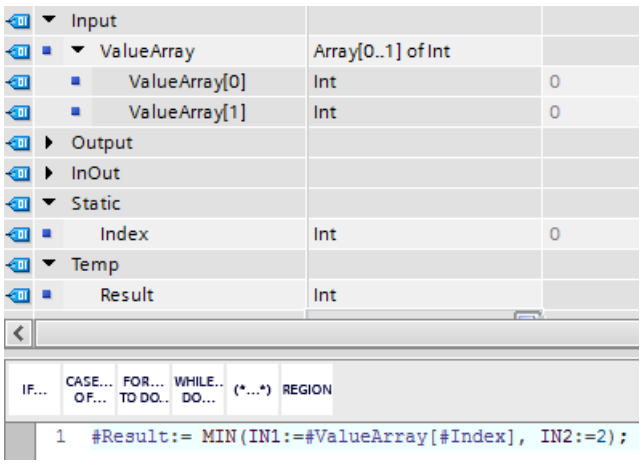

Weitere Informationen zur indirekten Adressierung von ARRAY-Komponenten finden Sie hier:

- Indirektes Indizieren von ARRAY-Komponenten (Seite 127)
- Programmierempfehlungen:ARRAY-Komponenten indirekt adressieren (Seite 211)

Fehlerbehandlung bei ARRAY-Zugriffen

Zugriffsfehler entstehen, wenn Sie zur Laufzeit auf ein Element zugreifen, das außerhalb der deklarierten ARRAY-Grenzen liegt. Die verschiedenen CPU-Familien reagieren unterschiedlich auf Überschreitungen der ARRAY-Grenzen:

- S7-300/400
  - Die CPU wechselt in den Betriebszustand "STOP".
  - Um dies zu vermeiden, können Sie den Programmablauffehler-OB (OB 85) programmieren.
  - In SCL haben Sie zusätzlich die Möglichkeit, in den Bausteineigenschaften das Attribut "ARRAY-Grenzen prüfen" zu aktivieren. Es bewirkt, dass bei ARRAY-Zugriffsfehlern der Freigabeausgang ENO auf FALSE gesetzt wird.
- S7-1200
  - Die CPU erzeugt einen Diagnosepuffereintrag und verbleibt im Betriebszustand "RUN".
- S7-1500
  - Die CPU wechselt in den Betriebszustand "STOP".
  - Um dies zu vermeiden, können Sie den Programmierfehler-OB (OB 121) programmieren.
  - Außerdem haben Sie die Möglichkeit, die lokale Fehlerbehandlung mit den Anweisungen "GET_ERROR: Fehler lokal abfragen" oder "GET_ERROR_ID: Fehler-ID lokal abfragen" zu programmieren.

Hinweis

Überwachung von ARRAY-Zugriffsfehlern durch ENO

Der Freigabeausgang ENO wird nicht auf den Signalzustand FALSE gesetzt, wenn bei der Ausführung einer Anweisung die ARRAY-Grenzen überschritten werden. Eine Ausnahme bilden nur SCL-Bausteine auf CPUs der Baureihen S7-300/400, für die die Bausteineigenschaft "ARRAY-Grenzen prüfen" gesetzt ist.

3.10.4 Variablen vom Datentyp ARRAY / ARRAY[*] übergeben

Variablen vom Datentyp ARRAY übergeben

Sie können auch einzelne Elemente eines ARRAYs als Aktualparameter übergeben, wenn das Element dem Datentyp des Formalparameters entspricht.

Sie können Variablen vom Datentyp ARRAY als Parameter übergeben. Wenn ein Baustein einen Eingangsparameter vom Datentyp ARRAY hat, müssen Sie als Aktualparameter ein ARRAY mit identischem Aufbau übergeben. Das bedeutet, dass der Datentyp, die Anzahl der Dimensionen und die Anzahl der Feldkomponenten identisch sein müssen.

ARRAYs sind einander zuweisbar, wenn sie einen identischen Aufbau haben. Das bedeutet, dass der Datentyp, die Anzahl der Dimensionen und die Anzahl der Feldkomponenten identisch sein müssen. Die Namen der ARRAYs müssen nicht übereinstimmen.

Variablen vom Datentyp ARRAY[*] übergeben

Mithilfe von ARRAY[*] können Sie ARRAYs mit variablen Grenzen an den Parametern einer Funktion oder eines Funktionsbausteins deklarieren. Sie können, wenn Sie den Baustein erstellen, die Grenzen des ARRAYs, das später zur Laufzeit übergeben wird, nach dem Aufruf zur Laufzeit bestimmen.

Das folgende Bild zeigt zwei Aufrufe eines Bausteins mit einem Eingangsparameter des Datentyps ARRAY[*]. Bei jedem Aufruf werden ARRAYs verschiedener Länge übergeben.

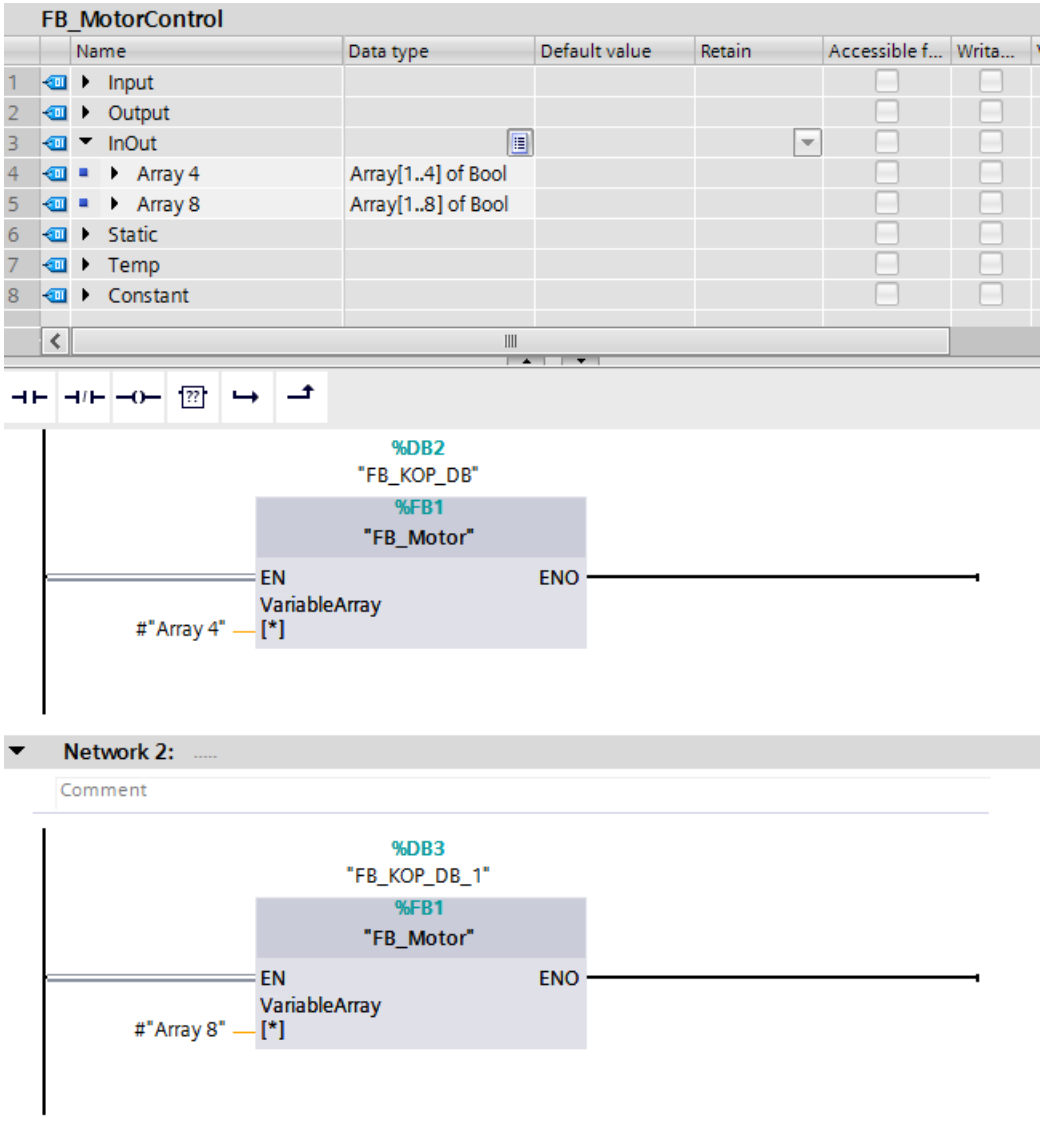

Geltende Regeln bei der Übergabe von ARRAYs mit flexiblen Grenzen

Erstellen Sie für das Beispiel die Funktion "BlockWithArrayStarIn_FC", um sie nachher aufrufen zu können:

BlockWithArrayStarIn_FC

	Name	Data type
1	Input	
2	Array_Star_In	Array[*] of Real
3	Array_Star_In[*]	Real
4	Output	
5	InOut	
6	Temp	
7	Constant	
8	Return	
9	BlockWithArrayStarIn_FC	Void

Die Zuweisung von ARRAY[*] an ARRAY[*] ist zulässig, wenn die Anzahl der Dimensionen und der Datentyp übereinstimmen. Einzelne ARRAY-Komponenten können hier jedoch nicht übergeben werden:

1. Deklarieren Sie den Funktionsbaustein "BlockCaller_FB" und rufen Sie die Funktion "BlockWithArrayStarIn_FC" auf:

BlockCaller_FB

	Name	Data type	Default value	Retain
1	Input			
2	Output			
3	InOut			
4	oneDimArrayOfReal	Array[*] of Real		
5	oneDimArrayOfReal[*]	Real		
6	Static			
7	Temp			
8	Constant			

```
1  "BlockWithArrayStarIn_FC"(Array_Star_In := #oneDimArrayOfReal);
2
```

Die Zuweisung eines ARRAYs mit bekannten Grenzen an ein ARRAY[*] ist zulässig, wenn die Anzahl der Dimensionen und der Datentyp übereinstimmen. Hier ist auch die Zuweisung einzelner ARRAY-Komponenten möglich:

1. Deklarieren Sie den Funktionsbaustein "BlockCallerFixLimits_FB" und rufen Sie die Funktion "BlockWithArrayStarIn_FC" zweimal auf:

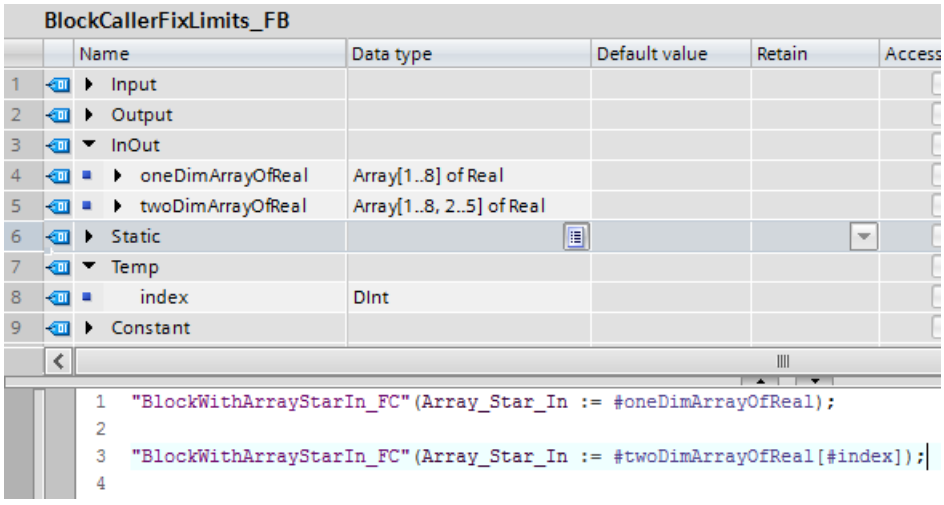

Die Zuweisung von ARRAY[*] an VARIANT ist zulässig. Hier ist auch die Zuweisung einzelner ARRAY-Komponenten möglich:

1. Erstellen Sie die Funktion "BlockWithVariantIn_FC", um sie aufrufen zu können:

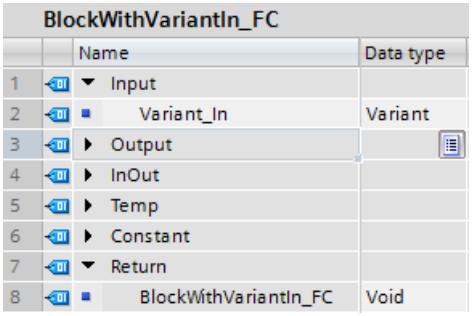

BlockWithVariantIn_FC

	Name	Data type
1	Input	
2	Variant_In	Variant
3	Output	
4	InOut	
5	Temp	
6	Constant	
7	Return	
8	BlockWithVariantIn_FC	Void

2. Deklarieren Sie die Funktion "BlockWithArrayStarInVariant_FC" und rufen Sie die Funktion "BlockWithVariantIn_FC" auf:

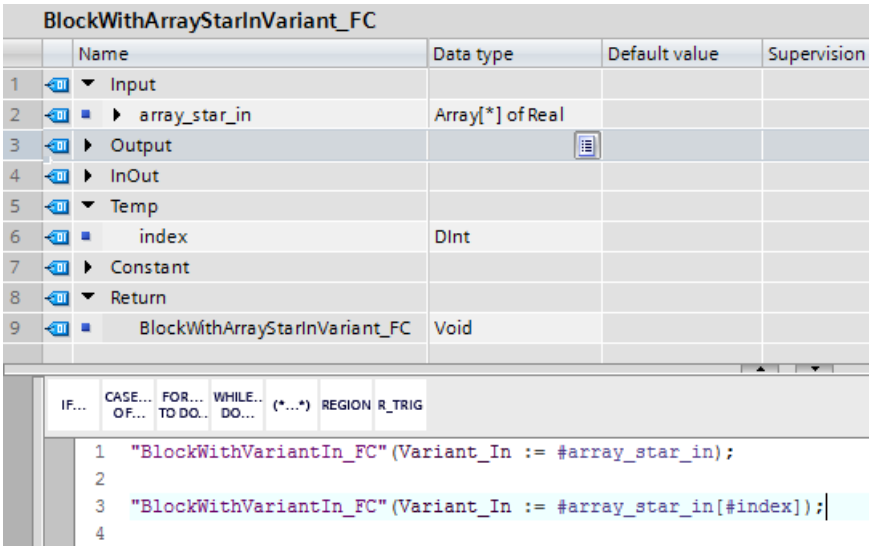

BlockWithArrayStarInVariant_FC

	Name	Data type	Default value	Supervision
1	Input			
2	array_star_in	Array[*] of Real		
3	Output			
4	InOut			
5	Temp			
6	index	DInt		
7	Constant			
8	Return			
9	BlockWithArrayStarInVariant_FC	Void		

IF... CASE... OF... FOR... TO DO... WHILE... DO... (*...*) REGION R_TRIG

```
1  "BlockWithVariantIn_FC"(Variant_In := #array_star_in);
2
3  "BlockWithVariantIn_FC"(Variant_In := #array_star_in[#index]);
4
```

Die Zuweisung von ARRAY[*] an ARRAYs mit festen Grenzen ist nicht zulässig.

Hinweis

Bausteinparameter vom Typ ARRAY[*] müssen zwingend mit einem Aktualparameter versorgt werden.

Siehe auch

Regeln zur Versorgung von Bausteinparametern (Seite 78)

3.10.5 Beispiele für die Verwendung von ARRAYs

3.10.5.1 Beispiel für ein mehrdimensionales ARRAY

Beschreibung

Variablen vom Datentyp ARRAY können bis zu 6 Dimensionen enthalten. Es gelten die gleichen Bestimmungen wie für eindimensionale ARRAYs. Die Bereiche der Dimensionen werden bei der Deklaration in eckigen Klammern, durch je ein Komma getrennt, geschrieben. In mehrdimensionalen ARRAYs werden die Komponenten beginnend mit der ersten Dimension abgelegt.

Die folgende Tabelle zeigt die Deklaration einer zweidimensionalen Variable vom Datentyp ARRAY:

Name	Datentyp	Wert	Kommentar
Betr_Temp	ARRAY[1..2, 1..3] of INT	1,1,4(0)	Zweidimensionale Variable vom Datentyp ARRAY mit 6 Komponenten. Die ersten zwei Komponenten werden mit dem Wert "1" belegt. Die restlichen vier Komponenten werden mit dem Wert "0" belegt.

Das folgende Bild zeigt die Struktur der deklarierten Variablen vom Datentyp ARRAY:

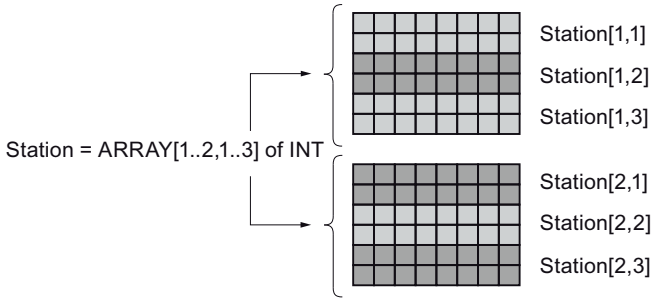

Zugriff auf die Komponenten

Auf die Werte der einzelnen Komponenten greifen Sie über einen Index zu. Für den Index können Sie eine Konstante oder eine Variable verwenden. Der Index der ersten Komponente ist z. B. [1,1] und der Index der vierten Komponente [2,1]. Im Programm ist z. B. die Angabe "Station[2,1]" erforderlich, um auf den Wert der vierten Komponente zuzugreifen.

3.10.5.2 Beispiel für die Berechnung des Skalarprodukts zweier Vektoren mit ARRAY[*]

Beschreibung

Das Skalarprodukt ist eine mathematische Verknüpfung, die zwei Vektoren eine Zahl (Skalar) zuordnet.

Das Skalarprodukt zweier Vektoren ergibt eine skalare Größe und ist definiert durch:

$$\vec{a} \cdot \vec{b} = a \cdot b \cdot \cos\alpha$$
$$= a_1 \cdot b_1 + a_2 \cdot b_2 + a_3 \cdot b_3$$

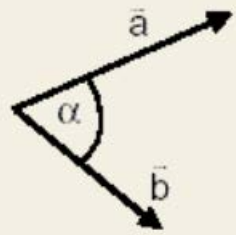

Dabei ist ∝ der Winkel zwischen den beiden Vektoren $\vec{a}$ und $\vec{b}$.

Ein Beispiel für die Berechnung eines Skalarprodukts ist:

$$\vec{a} = \begin{pmatrix} 1 \\ 3 \\ 5 \end{pmatrix}, \vec{b} = \begin{pmatrix} 2 \\ 0 \\ 4 \end{pmatrix}, \vec{a} * \vec{b} = (1*2 + 3*0 + 5*4) = 22$$

In diesem Fall ist das Ergebnis die Zahl 22.

Im folgenden Programmierbeispiel sehen Sie, wie Sie mithilfe des ARRAY[*] das Skalarprodukt zweier Vektoren mit flexiblen ARRAY-Grenzen berechnen können. Dabei dient die Funktion "ScalarProduct_FC" als Vorlage für die Berechnung der einzelnen Skalarprodukte.

Für das Programmierbeispiel benötigen Sie die folgenden Objekte:

- Zwei Datenbausteine und einen PLC-Datentyp (UDT) zur Verwaltung der Vektorendaten
- Funktion, die den Programmcode zur Berechnung eines Skalarprodukts enthält
- Die beiden Anweisungen "LOWER_BOUND" und "UPPER_BOUND" um die ARRAY-Grenzen auszulesen.
- Organisationsbaustein zur Berechnung der Skalarprodukte

Vorgehen

In diesem Beispiel werden Vektoren, also ARRAYs mit 1 Dimension, verwendet. Andere Berechnungen, wie z. B. Matrizenmultiplikationen, können natürlich auch mit mehrdimensionalen ARRAYs arbeiten.

Um die Skalarprodukte zu berechnen, gehen Sie folgendermaßen vor:

1. Legen Sie den PLC-Datentyp (UDT) "VectorArrays_UDT" an:

VectorArrays_UDT

	Name	Data type
1	VectorD5Coordinates	Array[1..5] of Real
2	VectorD5Coordinates[1]	Real
3	VectorD5Coordinates[2]	Real
4	VectorD5Coordinates[3]	Real
5	VectorD5Coordinates[4]	Real
6	VectorD5Coordinates[5]	Real
7	VectorE13Coordinates	Array[1..13] of Real
8	VectorE13Coordinates[1]	Real
9	VectorE13Coordinates[2]	Real
10	VectorE13Coordinates[3]	Real
11	VectorE13Coordinates[4]	Real
12	VectorE13Coordinates[5]	Real
13	VectorE13Coordinates[6]	Real
14	VectorE13Coordinates[7]	Real
15	VectorE13Coordinates[8]	Real
16	VectorE13Coordinates[9]	Real
17	VectorE13Coordinates[10]	Real
18	VectorE13Coordinates[11]	Real
19	VectorE13Coordinates[12]	Real
20	VectorE13Coordinates[13]	Real

Die beiden ARRAYs "VectorD5Coordinates" und "VectorE13Coordinates" liefern eine Datengrundlage, mit der Sie anschließend die Skalarprodukte berechnen können.

2. Legen Sie den Datenbaustein "VectorArrays1_DB" auf Basis des PLC-Datentyps "VectorArrays_UDT" an:

VectorArrays1_DB

	Name	Data type
1	Static	
2	VectorD5Coordinates	Array[1..5] of Real
3	VectorE13Coordinates	Array[1..13] of Real

3. Legen Sie den zweiten Datenbaustein "VectorArrays2_DB" an. Er beinhaltet neben den Vektoren aus dem PLC-Datentyp "VectorArrays_UDT" noch zwei weitere Vektoren:

VectorArrays2_DB

	Name	Data type
1	Static	
2	VectorA10Coordinates	Array[1..10] of Real
3	VectorB13Coordinates	Array[1..13] of Real
4	VectorsC	"VectorArrays_UDT"
5	VectorD5Coordinates	Array[1..5] of Real
6	VectorE13Coordinates	Array[1..13] of Real

Jetzt haben Sie die Datengrundlagen für die Berechnung der Skalarprodukte erstellt.

4. Legen Sie für die Erstellung des Rechenwegs zur Berechnung eines Skalarprodukts die Funktion "ScalarProduct_FC" als Vorlage an:
   Die Bausteinschnittstelle:

ScalarProduct_FC

	Name	Data type	Comment
1	Input		
2	Vector1	Array[*] of Real	Interface for data input for Vector 1
3	Vector1[*]	Real	Interface for data input for Vector 1
4	Vector2	Array[*] of Real	Interface for data input for Vector 1
5	Vector2[*]	Real	Interface for data input for Vector 1
6	Output		
7	InOut		
8	Temp		
9	Sum	Real	
10	i	Int	
11	LowerBound1	DInt	
12	LowerBound2	DInt	
13	Upperbound1	DInt	
14	UpperBound2	DInt	
15	Constant		
16	Return		
17	ScalarProduct_FC	Real	

Der Programmcode:

```
1  #LowerBound1 := LOWER_BOUND(ARR := #Vector1, DIM := 1);
2  #LowerBound2 := LOWER_BOUND(ARR := #Vector2, DIM := 1);
3  #Upperbound1 := UPPER_BOUND(ARR := #Vector1, DIM := 1);
4  #Upperbound1 := UPPER_BOUND(ARR := #Vector1, DIM := 1);
5
6  IF #LowerBound1 <> #LowerBound2 OR #Upperbound1 <> #UpperBound2 THEN
7      #ScalarProduct_FC := -1;
8      RETURN;
9  END_IF;
10 #Sum := 0;
11 FOR #i := #LowerBound1 TO #Upperbound1 DO
12     #Sum += #Vector1[#i] * #Vector2[#i];
13 END_FOR;
14
15 #ScalarProduct_FC := #Sum;
16
```

In den Zeilen 1 - 4 werden die unteren und oberen ARRAY-Grenzen von Vektor 1 und Vektor 2 abgefragt. Damit ist bekannt, wie viele Koordinaten die beiden ARRAYs der Vektoren haben. Da das Skalarprodukt nur dann gebildet werden kann, wenn die Anzahl der Koordinaten der beiden Vektoren, die Sie miteinander multiplizieren wollen, übereinstimmt, werden die Zeilen 6 bis 9 benötigt.
Wenn die beiden Unter- oder Obergrenzen der ARRAYs ungleich sind, dann liefert die Funktion "ScalarProduct_FC" den Funktionswert "-1" und der Programmbaustein wird verlassen (RETURN).

Wenn die beiden Unter- bzw. Obergrenzen der ARRAYs gleich sind, dann wird die Variable #Sum mit dem Wert "0" initialisiert (Zeile 10) und die Berechnung der Skalarprodukte wird ausgeführt (Zeilen 11 - 13).

5. Legen Sie den Organisationsbaustein "Main_OB" mit der Ereignisklasse "Program cycle" an. Die im Schritt 4 erstellte Funktion "ScalarProduct_FC" dient jetzt im "Main_OB" als Vorlage für die Berechnung von Skalarprodukten:
   Die Bausteinschnittstelle:

Main_OB		
	Name	Data type
1	Input	
2	Initial_Call	Bool
3	Remanence	Bool
4	Temp	
5	Result1	Real
6	Result2	Real
7	Result3	Real
8	Result4	Real
9	Result5	Real
10	Result6	Real
11	Result7	Real
12	Constant	

   Der Programmcode:

```
1  #Result1 := "ScalarProduct_FC"(Vector1 := "VectorArrays2_DB".VectorA10Coordinates,
2                                 Vector2 := "VectorArrays1_DB".VectorE13Coordinates);
3  #Result2 := "ScalarProduct_FC"(Vector1 := "VectorArrays2_DB".VectorsC.VectorD5Coordinates,
4                                 Vector2 := "VectorArrays1_DB".VectorD5Coordinates);
5  #Result3 := "ScalarProduct_FC"(Vector1 := "VectorArrays2_DB".VectorB13Coordinates,
6                                 Vector2 := "VectorArrays1_DB".VectorE13Coordinates);
7  #Result4 := "ScalarProduct_FC"(Vector1 := "VectorArrays2_DB".VectorA10Coordinates,
8                                 Vector2 := "VectorArrays2_DB".VectorA10Coordinates);
9  #Result5 := "ScalarProduct_FC"(Vector1 := "VectorArrays2_DB".VectorA10Coordinates,
10                                Vector2 := "VectorArrays1_DB".VectorE13Coordinates);
11
```

Ergebnis

Die Ergebnisse der einzelnen Skalarprodukt-Berechnungen werden in die Variablen #Result1 -5 geschrieben.

3.11 Zeiger

3.11.1 Referenzen

3.11.1.1 Grundlagen zu Referenzen

Beschreibung

Eine Referenz ist eine Variable, die selbst keinen Wert enthält, sondern auf den Speicherort einer anderen Variablen zeigt.

Referenzen ermöglichen die Weitergabe von Variablen über Bausteingrenzen hinweg und damit die direkte Manipulation von Variablenwerten, ohne eine Kopie der Variablen anlegen zu müssen.

Sie geben bereits bei der Deklaration der Referenz an, von welchem Datentyp die referenzierte Variable sein muss. Referenzen sind typsicher. Dies ist insbesondere für Steuerungssysteme, auf denen Laufzeitfehler vermieden werden müssen, ein wesentlicher Aspekt. Die in der IEC getroffene Einschränkung, dass Referenzen auf temporäre Datenelemente nicht gebildet werden dürfen, steigert zusätzlich die Sicherheit. Fehlersituationen im Laufzeitverhalten sind so vermeidbar.

Es ist sichergestellt, dass Referenzen entweder auf einen gültigen Speicherplatz vom richtigen Datentyp verweisen oder mit dem Wert NULL belegt sind.

Referenzen im Programm einsetzen

Voraussetzung für den Einsatz von Referenzen ist eine CPU der Baureihe S7-1500 mit dem Firmwarestand V2.5 oder höher.

Die folgende Grafik gibt einen Überblick über die Einsatzmöglichkeiten von Referenzen.

Anweisung	Anwendung	Weitere Information
Referenzen deklarieren REF_TO <data type> Deklaration der Referenz in der Bausteinschnittstelle	#myRefInt : REF_TO INT; #myRefType : REF_TO myType; #myRefArray : REF_TO Array[0..100] of myType;	Referenzen deklarieren
Referenzieren REF (<tag>) Bildung einer Referenz auf eine konkrete Variable	#myRefInt := REF (#myInt); #myRefType := REF ("myDB".myType); #myRefArray := REF (#myArray); //myInt and myArray defined in Static	Referenzieren
Dereferenzieren <Reference>^ Zugriff auf den Wert der referenzierten Variable	#myInt := #myRefInt^; #myInt := #myRefType^.elementOfInt; #myInt := #myRefArray^[#index].elementOfInt;	Dereferenzieren
Standardanweisungen mit Referenzen Zuweisung Vergleich	#myRefInt := #yourRefInt; #myVARIANT := #myReference; //myVARIANT defined in Temp IF #myRefInt = NULL THEN...; IF #myRefInt <> NULL THEN...;	Standardanweisungen mit Referenzen
Zuweisungsversuch <Reference> ?= <VARIANT> Zuweisungsversuch einer Variable vom Datentyp VARIANT auf eine Referenzvariable	#myReference ?= #myVariant;	Zuweisungsversuch
Zuweisung beim Bausteinaufruf Übergabe von Referenzen beim Aufruf von Funktionen oder Funktionsbausteinen	"Callee"(MyRefIn1:= REF("MyDataBlock".MyInt), MyRefIn2:= NULL, MyRefOut=> #MyRefWord); //MyRefWord defined in Temp	Referenzen als Baustein-parameter übergeben

Unterschiede zwischen Referenzen und VARIANT

Eine Variable vom Datentyp VARIANT ist wie ein Zeiger, der auf andere Variablen beliebigen Datentyps zeigen kann. Bei der Programmerstellung ist deshalb noch nicht definiert, auf welchen Datentyp die VARIANT-Variable zeigen wird. Der Datentyp wird erst zur Laufzeit bestimmt. Eine VARIANT-Variable kann in unterschiedlichen Programmzyklen sogar unterschiedliche Datentypen annehmen. So eignet sich der Datentyp VARIANT zur Erstellung generischer Programme und zur indirekten Adressierung. Wenn Sie eine VARIANT-Variable im Programmcode weiterbearbeiten wollen, müssen Sie jedoch zunächst mithilfe spezieller Anweisungen ermitteln, welcher Datentyp aktuell vorliegt. Sie können VARIANT-Variablen zudem nicht direkt lesen und schreiben, sondern müssen dazu ebenfalls spezielle Anweisungen, z. B. VariantGet und VariantPut verwenden.

Wenn Sie Referenzen verwenden, definieren Sie den Datentyp bereits bei der Programmerstellung. Da der Datentyp nicht zur Laufzeit ermittelt werden muss, wird das Programm performanter und übersichtlicher. Durch Dereferenzieren können Sie direkt schreibend oder lesend auf die referenzierte Variable zugreifen, ohne zusätzliche Anweisungen in Ihr Programm einzubinden.

Im Gegensatz zu VARIANT dürfen Referenzen jedoch nur auf Daten zeigen, die in einem optimierten Speicherbereich liegen.

Beispiele

Das folgende Beispiel zeigt verschiedene Anwendungsmöglichkeiten für Referenzen.

Die Bausteinschnittstelle enthält die Variable "myRefInt", die als Referenz deklariert wurde:

Name	Data type	Default value
Input		
Output		
InOut		
myVARIANT	Variant	
Static		
a	Int	5
b	Int	10
result	Int	0
Temp		
myRefInt	REF_TO Int	

Das folgende Bild zeigt, wie diese Variable in SCL verwendet wird:

```
1   #result := #a + #b;          //#result has value 15
2   #myRefInt := REF(#a);        //#myRefInt points to #a
3   #myRefInt^:= 3;              //#a contains value 3
4   #result := #a + #b;          //#result has value 13
5
6
7
8   #myRefInt ?= #myVARIANT;     //Ok, when tag of data type INT is passed
9                                // to #myVARIANT via block call
10  IF #myRefInt <> NULL THEN    //Check if Assignment Attempt was successful
11      #result := #myRefInt^ + #a;
12  END_IF;
13
```

3.11.1.2 Referenzen deklarieren

Referenzen deklarieren REF_TO <data type> Deklaration der Referenz in der Bausteinschnittstelle	#myRefInt : REF_TO INT; #myRefType : REF_TO myType; #myRefArray : REF_TO Array[0..100] of myType;

Beschreibung

Referenzen können in der Bausteinschnittstelle von Funktionen und Funktionsbausteinen deklariert werden. Die folgenden Deklarationsbereiche sind dafür zulässig:

- FC: Input, Output, Temp, Return
- FB: Temp
- OB: Temp

Zur Deklaration von Referenzen verwenden Sie das Schlüsselwort "REF_TO" und geben an, von welchem Datentyp die referenzierte Variable sein muss. Sie geben jedoch noch nicht an, auf welche konkrete Variable die Referenz zeigt:

1	Input	
2	Output	
3	InOut	
4	Static	
5	Temp	
6	myRefInt	REF_TO Int
7	myRefByte	REF_TO Byte
8	myRefWord	REF_TO Word
9	myRefDWord	REF_TO DWord
10	myRefReal	REF_TO Real
11	myRefLReal	REF_TO LReal
12	myRefType	REF_TO "myUDT"
13	myRefARRAY	REF_TO Array[0..3] of Int
14	myRefARRAY[0]	Int
15	myRefARRAY[1]	Int
16	myRefARRAY[2]	Int
17	myRefARRAY[3]	Int

Referenzen können auf folgende Elemente zeigen:

- Bitfolgen
  - Referenzen auf BOOL sind nicht möglich.
- Ganzzahlen
- Gleitpunktzahlen
- Zeichenfolgen
  - Bei Zeichenfolgen sind keine Längendeklarationen möglich.
- IEC-Zeiten
  - Referenzen auf IEC_TIMER und IEC_LTIMER sind möglich.
  - Referenzen auf abgeleitete Datentypen, z. B. TON, sind nicht möglich.
- IEC-Zähler
  - Referenzen auf IEC_COUNTER, IEC_UCOUNTER, IEC_SCOUNTER, IEC_USCOUNTER, IEC_DCOUNTER, IEC_UDCOUNTER sind möglich.
  - Referenzen auf abgeleitete Datentypen, z. B. CTU, sind nicht möglich.
- PLC-Datentypen (UDT)
- Systemdatentypen (SDT)
- ARRAYs aus den genannten Datentypen
  - Referenzen auf ARRAY[*] sind nicht möglich.
  - In der aktuellen Version des TIA Portals ist es nicht möglich, ein ARRAY aus Referenzen zu deklarieren. Die folgende Deklaration wäre also nicht zulässig:
    `ARRAY of REF_TO<Datentyp>`

Auf einen ganzen Datenbaustein kann nur dann gezeigt werden, wenn dieser ursprünglich von einem PLC-Datentyp (UDT) oder Systemdatentyp (SDT) abgeleitet wurde.

Initialisierung

Wird eine Referenzvariable angelegt, initialisiert das System diese mit dem Wert NULL. Das bedeutet, dass die Referenz selbst zwar existiert, jedoch noch auf keinen gültigen Speicherort verweist. Wird zur Laufzeit tatsächlich auf eine NULL-Referenz zugegriffen, wird ein Programmierfehler ausgegeben. Sie selbst können in der Bausteinschnittstelle keine Initialisierung vornehmen.

Um eine Referenz zu initialisieren, verwenden Sie die Anweisung "REF ()".

Siehe auch: Referenzieren (Seite 320)

Remanenz

Referenzen können nicht remanent sein. Sie können jedoch auf remanente Daten zeigen.

Hinweis

Maximale Anzahl an Referenzparametern pro Baustein

Die zulässige Maximalmenge von Parametern des Datentyps "REF_TO" in einem Baustein hängt von verschiedenen Faktoren ab, u. a. von der Bausteinart und der Anzahl an weiteren deklarierten Parametern mit strukturiertem Datentyp (ARRAYs, PLC-Datentypen, etc.) sowie von der Anzahl der deklarierten Instanzen.

Bei Überschreitung der Maximalmenge erhalten Sie während des Übersetzungslaufs eine Meldung. In diesem Fall können Sie mehrere Parameter in einem PLC-Datentyp (UDT) oder in einem globalen Datenbaustein (DB) zusammenfassen und diesen als Bausteinparameter übergeben.

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)

3.11.1.3 Referenzieren

Referenzieren	
REF (<tag>) Bildung einer Referenz auf eine konkrete Variable	#myRefInt := REF (#myInt); #myRefType := REF ("myDB".myType); #myRefArray := REF (#myArray); //myInt and myArray defined in Static

Beschreibung

Mit dem Schlüsselwort "REF()" geben Sie an, auf welche Variable eine zuvor deklarierte Referenz zeigen soll. Als Parameter geben Sie die zu referenzierende Variable an. Der Datentyp der Variable muss genau dem Datentyp der deklarierten Referenz entsprechen. Eine Referenz mit dem Datentyp "REF_TO Int" kann also nur auf eine Variable vom Datentyp "Int" zeigen. Eine Datentypkonvertierung findet nicht statt.

Hinweis

"REF()" als Aktualparameter beim Bausteinaufruf übergeben

Sie können "REF()" auch als Aktualparameter an einen aufgerufenen Baustein übergeben, in dessen Schnittstelle Referenzen deklariert sind.

Siehe auch: Referenzen als Bausteinparameter übergeben (Seite 331)

Regeln

Für das Referenzieren gelten folgende Regeln:

- Die Daten, auf die eine Referenz zeigt, müssen in einem optimierten Speicherbereich liegen.
- Die Referenz darf nur auf Daten in einem globalen DB oder statische Variablen zeigen.
- Die Referenz darf nicht auf folgende Daten zeigen:
  - temporäre Daten (TEMP)
  - globale Variablen aus der PLC-Variablentabelle
  - Bausteinparameter
  - Konstanten
  - schreibgeschützte Variablen
- Bei Referenzen auf Arrays gilt:
  - Die Arraygrenzen und Dimensionen der Referenz und der referenzierten Variable müssen identisch sein.
  - Referenzen auf Arrays mit dynamischen Grenzen (Array[*]) können nicht gebildet werden.
  - Referenzen auf ARRAY-DBs, die auf einem PLC-Datentyp basieren, müssen wie folgt gebildet werden:
    ```
    REF("my_ArrayDB_UDT"."THIS")
    REF("my_ArrayDB_UDT"."THIS"[i])
    ```

Beispiele

Das folgende Beispiel zeigt die Schnittstelle eines Bausteins. Die Schnittstelle enthält einige Parameter, die mit dem Datentyp "REF_TO" deklariert wurden. Bei der Deklaration der Referenz definieren Sie nur den Datentyp der referenzierten Variable, geben aber noch nicht an, auf welche Variable die Referenz verweisen soll.

Name	Data type
Input	
Output	
InOut	
Static	
a	Int
Temp	
myRefInt	REF_TO Int
myRefType	REF_TO "myUDT"
myRefArrayInt	REF_TO Array[0..1] of Int
Constant	

Im Programmcode geben Sie an, auf welche konkrete Variablen die deklarierten Referenzparameter zeigen sollen.

Beispiele in SCL:

```
1  #myRefInt := REF(#a);
2
3  #myRefType := REF("myDB".myUDT);
4
5  #myRefType := REF("myArrayDB_UDT"."THIS"[1]);
6
7  #myRefArrayInt := REF("myDB".myArray);
```

Beispiele in KOP:

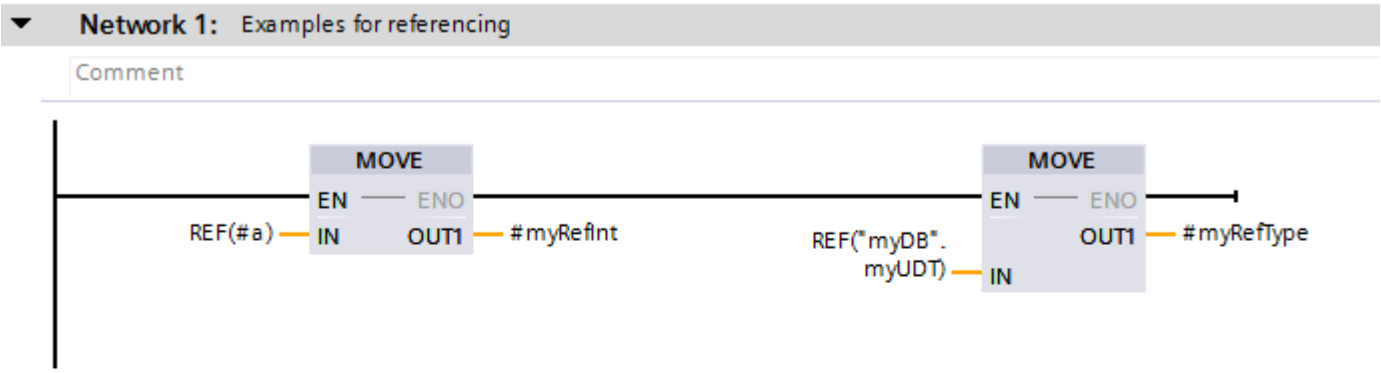

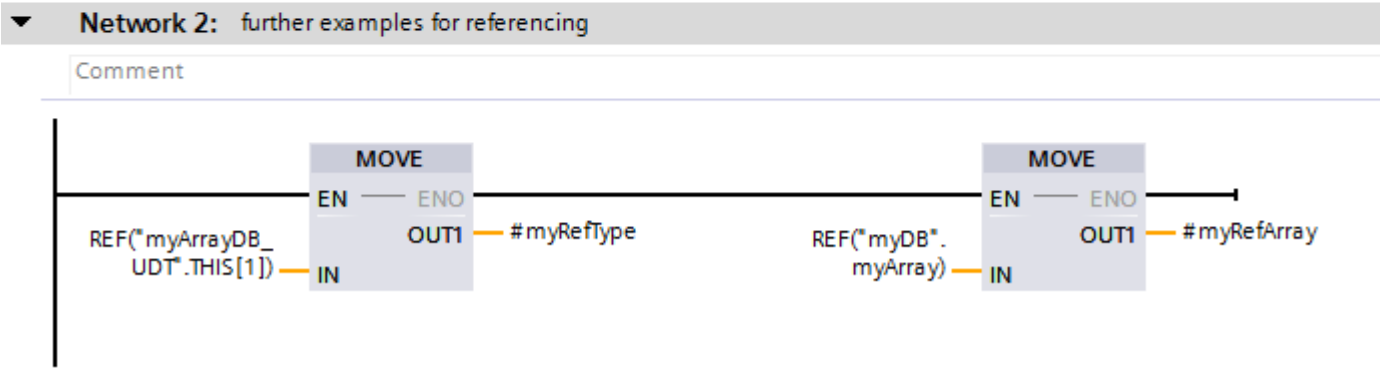

Beispiele in AWL:

Network 1: Examples for referencing

Comment

```
1        CALL  MOVE
2           Variant
3           IN  :=REF(#a)
4           OUT :=#myRefInt
5
6
7        CALL  MOVE
8           Variant
9           IN  :=REF("myDB".myUDT)
10          OUT :=#myRefType
11
12
13       CALL  MOVE
14          Variant
15          IN  :=REF("myArrayDB_UDT".THIS[1])
16          OUT :=#myRefType
17
18       CALL  MOVE
19          Variant
20          IN  :=REF("myDB".myArray)
21          OUT :=#myRefArray
```

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Referenzen deklarieren (Seite 318)

3.11.1.4 Dereferenzieren

Dereferenzieren <Reference>^ Zugriff auf den Wert der referenzierten Variable	#myInt := #myRefInt^; #myInt := #myRefType^.elementOfInt; #myInt := #myRefArray^[#index].elementOfInt;

Beschreibung

Um lesend oder schreibend auf den Wert einer referenzierten Variable zuzugreifen, verwenden Sie das Caret-Zeichen "^". Diese Art des Zugriffs wird auch als "Dereferenzieren" bezeichnet.

Hinweis

Bevor Sie auf den Wert einer Referenz zugreifen können, müssen Sie zunächst mit der Anweisung "REF()" angeben, auf welche Variable die Referenz zeigen soll.

Beispiele

Das folgende Beispiel zeigt die Schnittstelle eines Bausteins. Die Schnittstelle enthält den Parameter "myRefInt", der als Referenz deklariert wurde, sowie einige statische Parameter, die bereits mit einem Wert initialisiert wurden:

		Name	Data type	Default value
1		Input		
2		Output		
3		InOut		
4		Static		
5		a	Int	10
6		b	Int	20
7		result	Int	0
8		Temp		
9		myRefInt	REF_TO Int	
10		Constant		

Das folgende Bild zeigt, wie in SCL zunächst mit der Anweisung "REF()" angegeben wird, dass die Referenz "myRefInt" auf "#a" zeigt, und anschließend mit "#a" gerechnet wird:

```
1  #result := #a + #b;              // #result has the value 30
2
3  #myRefInt := REF (#a);           // #myRef1 points to #a
4  #myRefInt^ := 40;                  // #a has the value 40
5  #result := #a + #b;              // result has the value 60
6  #result := #myRefInt^ + #b;        // result also has the value 60
7
```

Siehe auch

Grundlagen zu Referenzen (Seite 315)

3.11.1.5 Standardanweisungen mit Referenzen

Standardanweisungen mit Referenzen	
Zuweisung	#myRefInt := #yourRefInt; #myVARIANT := #myReference; //myVARIANT defined in Temp
Vergleich	IF #myRefInt = NULL THEN...; IF #myRefInt <> NULL THEN...;

Referenzen können Sie als Ein- oder Ausgangsparameter von Zuweisungs- oder Vergleichsanweisungen verwenden.

Dereferenzierte Werte können Sie als Parameter an allen Anweisungen verschalten, sofern sie den richtigen Datentyp haben.

Zuweisung

Referenzen können wie normale Variablen einander zugewiesen werden. Dabei wird die Adresse der referenzierten Variable zugewiesen, nicht deren Wert. Referenzen sind nur dann einander zuweisbar, wenn Sie auf den gleichen Datentyp verweisen. Eine implizite Datentypkonvertierung findet nicht statt.

Auch Referenzen auf PLC-Datentypen müssen vom gleichen Datentyp sein. Es genügt nicht, dass beide PLC-Datentypen den gleichen Aufbau haben.

Sie können eine Referenz auch einem VARIANT zuweisen. Der VARIANT muss in diesem Fall als temporäre Variable (Temp) deklariert sein.

Vergleich

Referenzen können in Vergleichsanweisungen verwendet werden. So können Sie durch einen Vergleich mit NULL feststellen, ob einer Referenz bereits eine Variable zugewiesen wurde. Dies ist immer dann sinnvoll, wenn durch den Programmfluss nicht eindeutig sichergestellt ist, dass eine Referenz initialisiert wurde.

Weitere Vergleiche sind nicht möglich.

Beispiele

Das folgende Beispiel zeigt die Verwendung von Referenzen in Zuweisungen und Vergleichsoperationen in SCL:

```
1  // Assignment of one reference to another reference
2  #myRefInt1 := #myRefInt2;
3
4  // Assignment of a reference to a variant
5  #myVariant := #myReference;
6
7  // Assignment of a value to the element "SetLimit"
8  // in the referenced PLC data type "Speed"
9  #myRefType^.SetLimit := 8;
10
11 // Assignment of a value to an element in a referenced ARRAY
12 #RefARRAY^[0] := 8;
13
14 // A reference is compared to NULL to determine if
15 // a valid reference for myRefInt has already been created.
16 IF #myRefInt <> NULL THEN
17     #myRefInt^ := 10;
18 END_IF;
19
20
```

Das folgende Beispiel zeigt die Verwendung in KOP:

Network 1:

Assignment of one reference to another reference

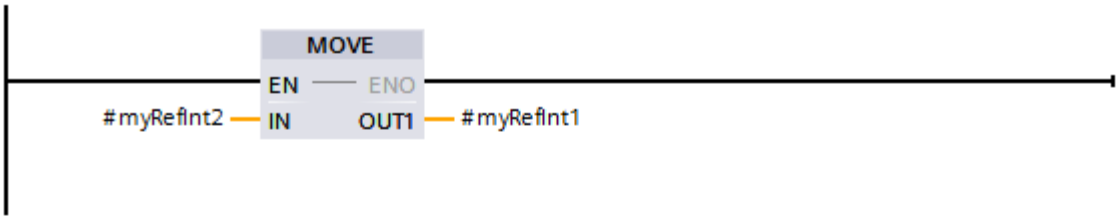

Network 2:

Assignment of a reference to a variant

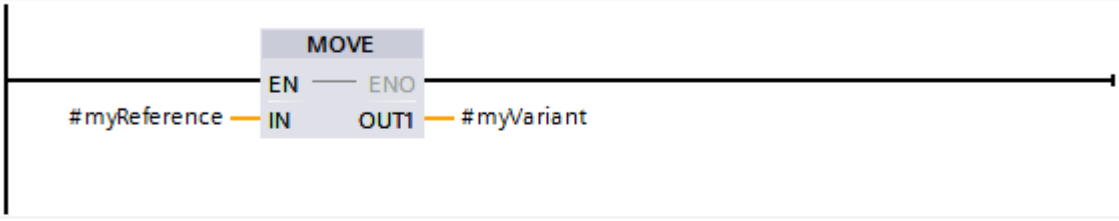

Network 3:

Assignment of a value to the element "SetLimit" in the referenced PLC data type "Speed"

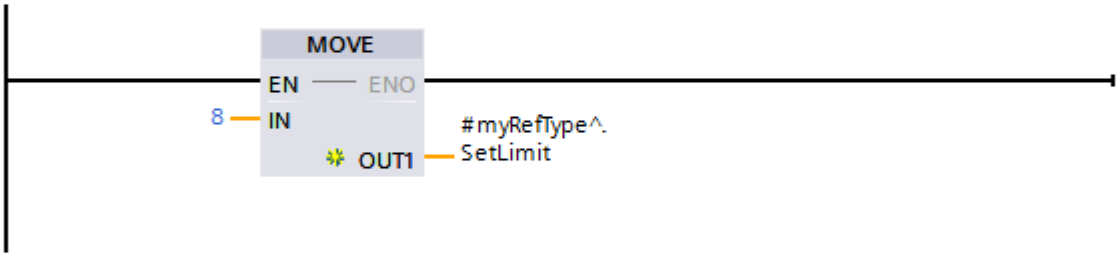

Network 4:

Assignment of a value to an element in a referenced ARRAY

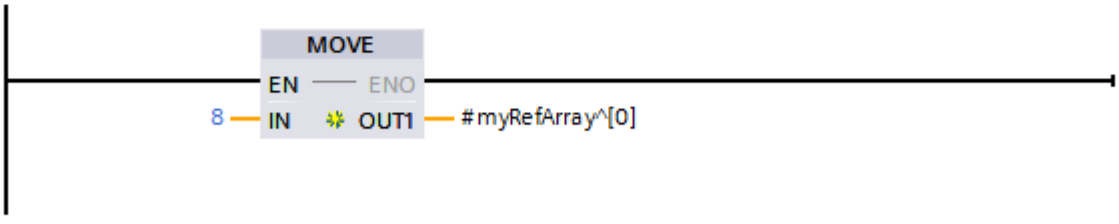

Network 5:

A reference is compared to NULL to determine if a valid reference for "myRefInt" has already been created

Siehe auch

Grundlagen zu Referenzen (Seite 315)

3.11.1.6 Zuweisungsversuch von VARIANT auf eine Referenz

Zuweisungsversuch	
<Reference> ?= <VARIANT> Zuweisungsversuch einer Variable vom Datentyp VARIANT auf eine Referenzvariable	#myReference ?= #myVariant;

Beschreibung

Den Zuweisungsversuch mit "?=" können Sie verwenden, um einer Referenz eine Variable vom Datentyp VARIANT zuzuweisen.

Ist die zur Laufzeit durch VARIANT adressierte Variable vom richtigen Datentyp und liegt sie in einem optimierten Speicherbereich, so steht nach der Ausführung der Zuweisung eine gültige Referenz in der Zielvariablen, andernfalls NULL.

Nach dem Zuweisungsversuch können Sie prüfen, ob der Versuch erfolgreich war und abhängig davon die Programmbearbeitung fortführen. In KOP und FUP können Sie dafür den Freigabeausgang "ENO" nutzen. "ENO" führt den Signalzustand "1", wenn der Zuweisungsversuch erfolgreich war. Nur dann werden die nachfolgenden Anweisungen im Netzwerk ausgeführt.

In AWL und SCL können Sie z. B. die Anweisungen "IS_NULL" oder "NOT_NULL" verwenden, um den Erfolg eines Zuweisungsversuchs zu prüfen. Beispiele hierzu finden Sie unten.

Regeln

Für den Zuweisungsversuch gelten folgende Regeln:

- Der zugewiesene VARIANT darf nicht auf eine temporäre Variable zeigen.
- Der zugewiesene VARIANT muss auf eine Variable im optimierten Speicherbereich zeigen.
- Wenn Sie einer Referenz auf ein ARRAY eine Variable vom Datentyp VARIANT zuweisen wollen, muss VARIANT auf ein ARRAY zeigen, dessen Grenzen exakt mit denen der deklarierten Referenz übereinstimmen. Ein VARIANT, der auf ein ARRAY [0..9] zeigt, passt nicht zu einer Variable REF_TO ARRAY[1..10].
- In SCL können Zuweisungsversuche nicht in Mehrfachzuweisungen (a := b := c;) verwendet werden.

Beispiel

Das folgende Beispiel zeigt, wie Sie durch Nutzung von Referenzen den Wert einer VARIANT-Variable direkt symbolisch lesen oder schreiben können. Das Kopieren des Werts über die Anweisungen "VariantGet" und "VariantPut" wird dadurch überflüssig.

	Name	Data type
	Input	
	<Hinzufügen>	
	Output	
	<Hinzufügen>	
	InOut	
	variantTelegramData	Variant
	Static	
	<Hinzufügen>	
	Temp	
	refTelegram1	REF_TO "Telegram1"
	refTelegram2	REF_TO "Telegram2"

Die VARIANT-Variable "variantTelegramData" kann sowohl Daten vom Typ "Telegram1" als auch Daten vom Typ "Telegram2" übergeben.

Durch einen Zuweisungsversuch wird geprüft, ob die Daten vom Typ "Telegram1" sind. Ist dies der Fall, so werden den Parametern "Info1" und "Info2" die Werte "T" und "W" zugewiesen.

Beispiel in SCL:

```
1  #refTelegram1 ?=  #variantTelegramData: // AssignmentAttempt
2  IF #refTelegram1 <> NULL THEN           //Check if Assignment Attempt was successful
3            #refTelegram1^.Info1 := 'T' ; // Dereference by using a
4            #refTelegram1^.Info2 := 'W' ; // succeeding caret ^
5  END_IF;
6
```

Beispiel in KOP:

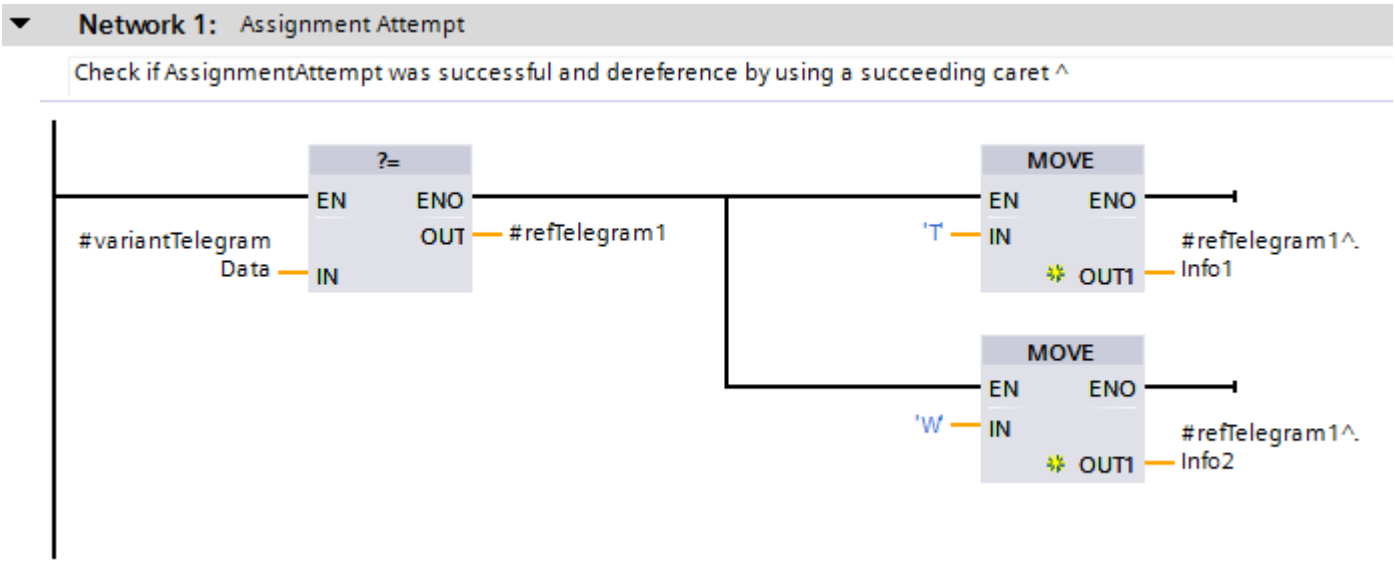

Siehe auch

Grundlagen zu Referenzen (Seite 315)

3.11.1.7 Referenzen als Bausteinparameter übergeben

Beschreibung

Referenzen können auch als Bausteinparameter beim Aufruf von Funktionen oder Funktionsbausteinen übergeben werden.

Wenn Sie einen Baustein aufrufen, in dessen Schnittstelle eine Referenz als Formalparameter deklariert ist, können Sie als Aktualparameter eine Referenz in Form von "REF()" oder "#MyRef" übergeben.

Regeln

Für die Übergabe von Referenzen als Bausteinparameter gelten folgende Regeln:

- "REF()" kann nur als Aktualparameter für einen Eingangsparameter (Input) einer Funktion übergeben werden.
- "REF()" kann nicht als Aktualparameter an einen Bausteinparameter vom Datentyp "VARIANT" übergeben werden.
- Der Datentyp der übergebenen Variable muss genau dem Datentyp der deklarierten Referenz entsprechen. Eine Datentypkonvertierung findet nicht statt.

Beispiel 1

Das folgende Beispiel zeigt die Schnittstelle der Funktion "Callee". Die Schnittstelle enthält einige Parameter, die als Referenz deklariert wurden:

Callee		
	Name	Data type
1	Input	
2	MyRefIn1	REF_TO Int
3	MyRefIn2	REF_TO Int
4	Output	
5	MyRefOut	REF_TO Word
6	InOut	
7	Temp	
8	Constant	
9	Return	
10	Callee	REF_TO Word

Das folgende Bild zeigt den Aufruf dieses Bausteins in SCL und die Parameterversorgung der deklarierten Referenzen:

```
1
2 #ref := "Callee"(MyRefIn1 := REF("MyDataBlock".MyInt),
3                  MyRefIn2 := NULL,
4                  MyRefOut => #MyRefWord);
```

Beispiel 2

Das folgende Beispiel zeigt weitere Möglichkeiten der Übergabe von Referenzen beim Bausteinaufruf:

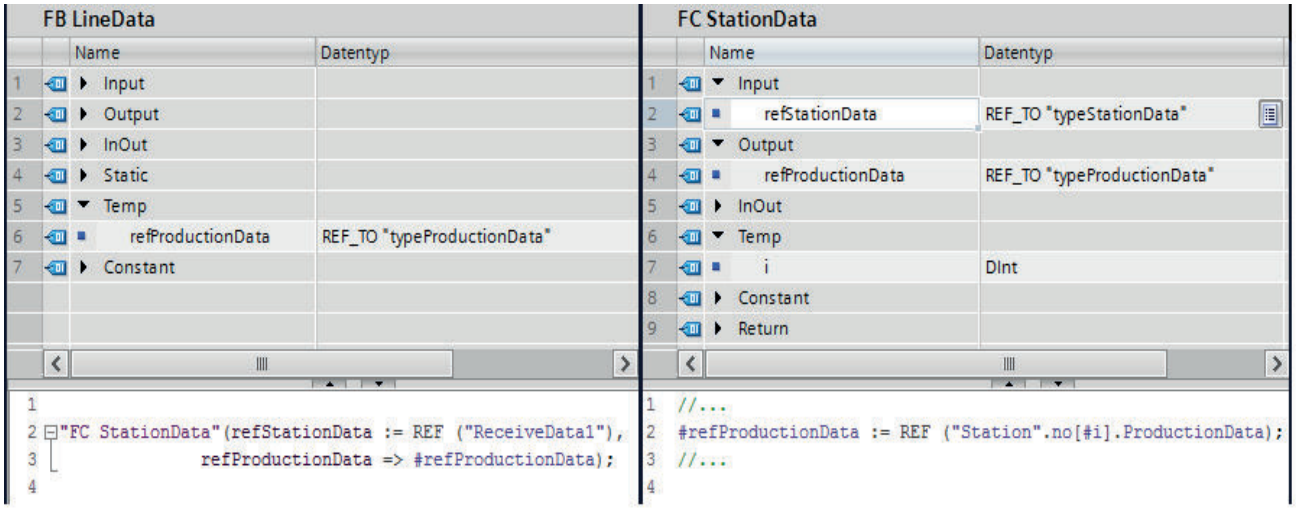

Im Beispiel sehen Sie zwei Möglichkeiten der Übergabe:

- Übergabe vom FB an den aufgerufenen FC:
  Der Eingangsparameter "refStationData" ist im "FC StationData" deklariert.
  Der "FB LineData" übergibt beim Aufruf `REF("ReceiveData1")` als Aktualparameter (Zeile 2 im Programmcode des "FB LineData"). "ReceiveData1" ist ein Datenbaustein, der auf dem PLC-Datentyp "typeStationData" basiert.
- Rückgabe vom FC an den aufrufenden FB:
  In der Schnittstelle beider Bausteine sind Referenzen mit gleichem Datentyp deklariert (im Beispiel: `"refPoductionData"`).
  Beim Bausteinaufruf werden die beiden Referenzen einander zugewiesen (Zeile 3 im Programmcode des "FB LineData").
  Die Initialisierung der Referenz erfolgt im aufgerufenen "FC StationData" (Zeile 2) und wird über den Ausgangsparameter direkt in die temporären Daten des aufrufenden FB geschrieben.

Das Beispiel zeigt einen Ausschnitt aus dem Programmierbeispiel "Daten per Referenz an den aufrufenden Baustein zurückgeben". Den detaillierten Programmcode finden Sie unter "Siehe auch".

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Dereferenzieren (Seite 324)

Beispiel: Daten per Referenz an den aufrufenden Baustein zurückgeben (Seite 336)

3.11.1.8 Beispiel: Übergabe von Variablen unterschiedlichen Datentyps mithilfe von Referenzen

Aufgabe

Im folgenden Anwendungsbeispiel fordert eine Produktionsanlage Daten an. Je nach Art der angeforderten Datenstruktur werden unterschiedliche Daten an die Anlage übergeben.

Die Datenstrukturen werden in zwei verschiedenen PLC-Datentypen abgebildet. Der PLC-Datentyp "typeTelegram1" enthält allgemeine Informationen vom Datentyp "CHAR", der PLC-Datentyp "typeTelegram2" enthält die Part-ID vom Datentyp "STRING". Aus den beiden PLC-Datentypen wird je ein Datenbaustein erzeugt:

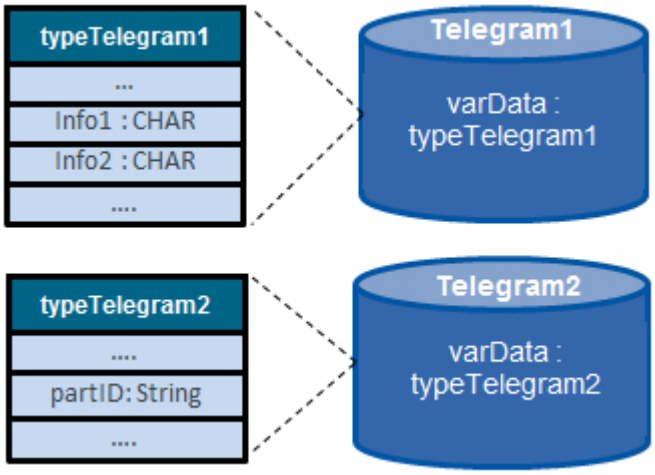

Beispielprogramme

Das folgende Bild zeigt die Umsetzung der Aufgabe mit und ohne Nutzung von Referenzen in SCL:

Example without references

	Name	Data type
1	Input	
2	Output	
3	InOut	
4	telegram	Variant
5	Static	
6	Temp	
7	tempTelegram1	"typeTelegram1"
8	tempTelegram2	"typeTelegram2"
9	Constant	

```
1
2  CASE TypeOf(#telegram) OF
3    typeTelegram1:
4      VariantGet(SRC := #telegram,    // copy data to "Temp"
5                 DST => #tempTelegram1);
6
7      #tempTelegram1.Info1 := 'T';
8      #tempTelegram1.Info2 := 'W';
9
10     VariantPut(SRC := #tempTelegram1,// copy back to "Inout"
11                DST := #telegram);
12
13   typeTelegram2:
14     VariantGet(SRC := #telegram,    // copy data to "Temp"
15                DST => #tempTelegram2);
16
17     #tempTelegram2.partID := 'RE2346';
18
19     VariantPut(SRC := #tempTelegram2,// copy back to "Inout"
20                DST := #telegram);
21 END_CASE;
```

Example with references

	Name	Data type
1	Input	
2	Output	
3	InOut	
4	telegram	Variant
5	Static	
6	Temp	
7	refTelegram1	REF_TO "typeTelegram1"
8	refTelegram2	REF_TO "typeTelegram2"
9	Constant	

```
1
2  CASE TypeOf(#telegram) OF
3    typeTelegram1:
4      #refTelegram1 ?= #telegram;   // AssignmentAttempt
5      #refTelegram1^.Info1 := 'T';  // Dereference by using
6      #refTelegram1^.Info2 := 'W';  // a succeeding caret ^
7
8    typeTelegram2:
9      #refTelegram2 ?= #telegram;  // AssignmentAttempt
10     #refTelegram2^.partID:='RE2346';
11 END_CASE;
12
```

Da beide Bausteine flexibel unterschiedliche Telegrammformate verarbeiten sollen, haben sie einen Durchgangsparameter (InOut) vom Datentyp "VARIANT", an dem die Telegramme als strukturierte Variablen übergeben werden. Da der Datentyp eines VARIANT-Parameters zum Zeitpunkt der Programmerstellung noch nicht bekannt ist, ist es nicht möglich, direkt auf den Parameter zuzugreifen. In beiden Beispielen wird deshalb zunächst mit der Anweisung "TypeOf" ermittelt, welcher Datentyp zum Aufrufzeitpunkt aktuell ansteht.

Im linken Baustein wird die zur Laufzeit anstehende Variable nun mithilfe der Anweisung "VariantGet" in eine temporäre Variable des entsprechenden Datentyps umkopiert und dort mit den passenden Werten beschrieben. Anschließend wird die Struktur mithilfe der Anweisung "VariantPut" wieder auf den Parameter "telegram" zurück kopiert. Das Kopieren wirkt sich ungünstig auf die Programmlaufzeit aus und kostet Speicherplatz.

Im rechten Baustein werden Referenzen genutzt, um die Aufgabe zu lösen: Die Schnittstelle enthält je einen Referenzparameter für die beiden möglichen Datenstrukturen. Abhängig von der aktuell anstehenden Variable wird die passende Referenz mithilfe des

"AssignmentAttempt" initialisiert und zeigt nun auf den Speicherort der Variable im Datenbaustein. Die Werte werden jetzt direkt in den Datenbaustein geschrieben.

Der Programmcode ist bei dieser Lösung übersichtlicher und leichter zu pflegen. Da die strukturierten Variablen nicht kopiert werden müssen, werden die Programmlaufzeit und der Speicherbedarf nicht beeinträchtigt.

3.11.1.9 Beispiel: Daten per Referenz an den aufrufenden Baustein zurückgeben

Aufgabe

Im folgenden Anwendungsbeispiel wird eine Produktionsanlage projektiert, die aus mehreren Stationen besteht. Die Stationen werden von zwei CPUs gesteuert, die Daten an eine Leit-CPU senden. Diese muss auswerten, zu welcher Station neue Daten vorliegen und übergeordnete Kontrollaufgaben durchführen, z. B. die produzierten Stückzahlen addieren.

Das folgende Bild zeigt den Aufbau der Produktionsanlage:

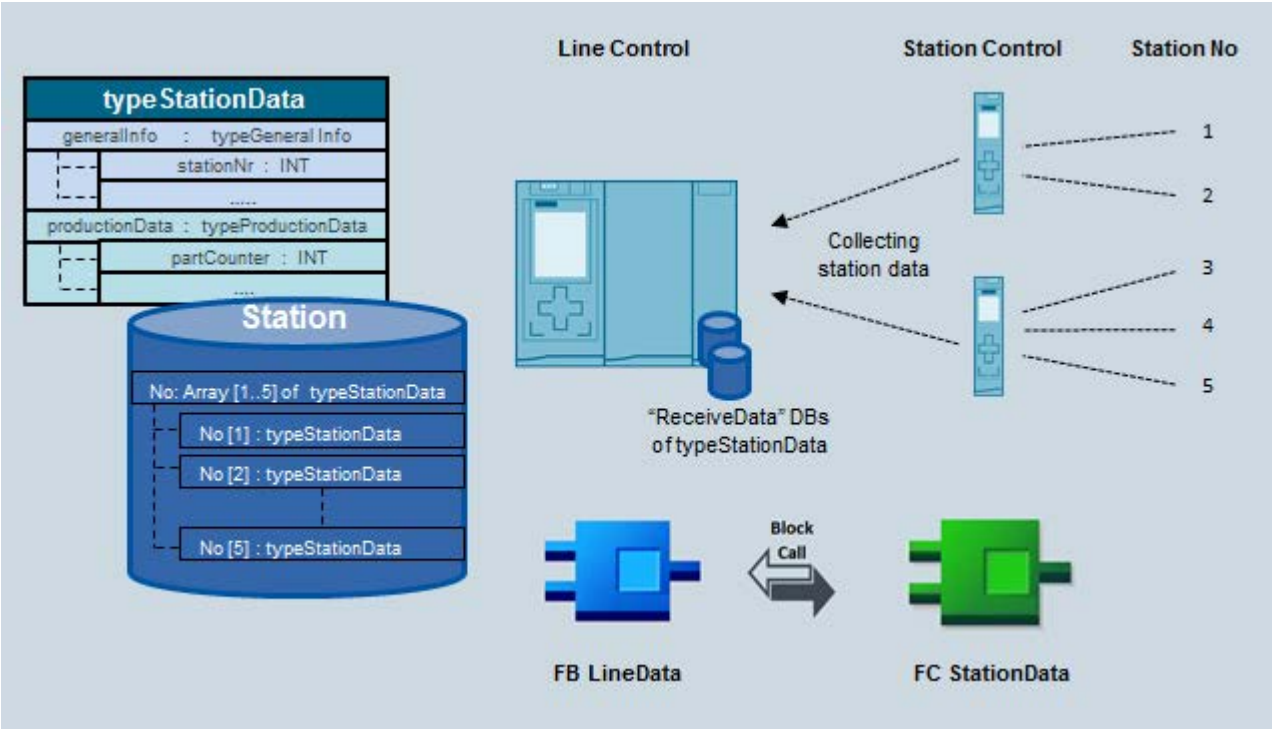

Bausteine im Beispielprogramm

Das Programm auf der Leit-CPU ist wie folgt aufgebaut:

- PLC-Datentyp "typeStationData"
  Die Daten einer Station werden in dem PLC-Datentyp "typeStationData" abgebildet. Der PLC-Datentyp enthält zwei unterlagerte PLC-Datentypen:
  - "typeGeneralInfo"
    Der Datentyp enthält die Stationsnummer.
  - "typeProductionData"
    Der Datentyp enthält die Stückzahl der Station.
- DB "ReceiveData"
  Für die Kommunikation zwischen den CPUs gibt es pro CPU einen Datenbaustein "ReceiveData" vom Typ "typeStationData", in den die Kommunikationsdaten geschrieben werden.
- "FB LineData"
  Der FB führt eine Gesamtauswertung der Produktionslinie durch. Dazu gehört z. B. das Summieren aller Stückzahlenzähler der einzelnen Stationen. Zudem ruft er den "FC StationData" auf, der unter anderem das Umkopieren der Empfangsdaten aus dem "ReceiveData" DB in den Global-DB "Station" übernimmt.
- "FC StationData"
  Der FC überprüft, von welcher Station neue Empfangsdaten vorliegen und kopiert diese in das entsprechende ARRAY-Element im Global-DB "Station". Das ARRAY-Element wird anschließend als Referenz an den aufrufenden "FB LineData" zur Weiterverarbeitung zurückgegeben.
- Global-DB "Station"
  In diesem DB liegen die Stationsdaten der fünf Stationen. Sie sind in einem ARRAY vom Typ "typeStationData" aus fünf Komponenten abgelegt.

Beispielprogramm

Das folgende Bild zeigt die Umsetzung der Aufgabe in SCL:

FB LineData

	Name	Data type	Defa
1	Input		
2	Output		
3	InOut		
4	Static		
5	assemblyLinePartNo	Int	0
6	Temp		
7	refProductionData	REF_TO "typeProductionData"	
8	Constant		

```
1  //block call with passing of references
2  "FC StationData"(refStationData := REF ("ReceiveData"),
3                   refProductionData => #refProductionData);
4
5  //update assembly line data
6  //using data from global DB element
7  #assemblyLinePartNo += #refProductionData^.partCounter;
8
```

FC StationData

	Name	Data type
1	Input	
2	refStationData	REF_TO "typeStationData"
3	Output	
4	refProductionData	REF_TO "typeProductionData"
5	InOut	
6	Temp	
7	statioNo	DInt
8	Constant	
9	Return	

```
1  //...
2  //write station no to #statioNo
3  #statioNo := #refStationData^.GeneralInfo.stationNo;
4
5  // write production data to global DB "station"
6  "Station".no[#statioNo].ProductionData := #refStationData^.ProductionData;
7
8  //write reference from global DB element to ouput
9  #refProductionData := REF ("Station".no[#statioNo].ProductionData);
10 //...
11
12
```

Der "FB LineData" ruft den "FC StationData" aufund übergibt beim Aufruf durch `REF("ReceiveData")` die Referenz auf den Empfangsdatenbaustein.

Durch Dereferenzieren auf den übergebenen Empfangsdatenbaustein liest der FC die Stationsnummer aus und schreibt sie in die temporäre Variable "stationNo".

Die Produktionsdaten aus dem Empfangsdatenbaustein werden anschließend in das entsprechende ARRAY-Element im Global-DB "Station" kopiert.

Dieses ARRAY-Element wird über den Ausgang "refProductionData" als Referenz an den aufrufen "FB LineData" zurückgegeben.

Dieser kann im Anschluss durch Dereferenzierung direkt auf das übergebene ARRAY-Element zugreifen und somit den Stückzahlenzähler für die gesamte Produktionslinie aktualisieren.

Siehe auch

Referenzen als Bausteinparameter übergeben (Seite 331)

3.11.2 VARIANT

3.11.2.1 Grundlagen zum VARIANT

Beschreibung

Ein Parameter vom Datentyp VARIANT ist ein Zeiger bzw. eine Referenz. Er kann auf Variablen verschiedener Datentypen zeigen. Der VARIANT-Zeiger kann auf keine Instanz und damit auch auf keine Multiinstanz oder ein ARRAY of Multiinstanzen zeigen. Der VARIANT-Zeiger kann ein Objekt eines elementaren Datentyps, wie z. B. INT oder REAL, sein. Es kann auch ein STRING, DTL, ARRAY of STRUCT, UDT oder ein ARRAY of UDT sein. Der VARIANT-Zeiger kann Strukturen erkennen und auf einzelne Strukturkomponenten zeigen. Ein Operand vom Datentyp VARIANT belegt keinen Platz im Instanz-Datenbaustein oder dem Arbeitsspeicher. Innerhalb des Speicherbereichs der CPU wird jedoch Speicherplatz belegt.

Eine Variable vom Typ VARIANT ist kein Objekt, sondern ein Verweis auf ein anderes Objekt. Einzelne Elemente vom Typ VARIANT können nur auf Formalparametern innerhalb der Bausteinschnittstelle einer Funktion in den Abschnitten VAR_IN, VAR_IN_OUT und VAR_TEMP deklariert werden. Daher können sie z. B. nicht in einem Datenbaustein oder im Abschnitt Static der Bausteinschnittstelle eines Funktionsbausteins deklariert werden, da ihre Größe nicht bekannt ist. Die Größe der referenzierten Objekte kann sich ändern.

Mithilfe des Datentyps VARIANT können Sie vor allem generische, standardisierte Funktionsbausteine (FB) oder Funktionen (FC) für verschiedene Datentypen erstellen. Dafür stehen Ihnen unterschiedliche Anweisungen in allen Programmiersprachen zur Verfügung. Bei der Programmerstellung können Sie festlegen, welche Datentypen der Baustein verarbeiten können soll. Der Datentyp VARIANT unterstützt Sie hierbei, indem er die Verschaltung beliebiger Variablen zulässt, auf deren Datentyp Sie dann im Baustein entsprechend reagieren können. Beim Bausteinaufruf können Sie die Parameter des Bausteins mit Variablen beliebigen Datentyps verschalten. Beim Bausteinaufruf wird neben einem Zeiger auf die Variable auch die Typinformation der Variable übergeben. Der Code des Bausteins kann dann entsprechend, entsprechend der zur Laufzeit übergebenen Variable, typkonform bearbeitet werden.

Wenn z. B. ein Bausteinparameter einer Funktion den Datentyp VARIANT hat, dann kann ihm an einer Stelle im Programm eine Variable vom Datentyp Ganzzahl und an einer anderen Stelle im Programm eine Variable vom Datentyp PLC-Datentyp übergeben werden. Die Funktion kann dann mit Hilfe der VARIANT-Anweisungen in die Lage versetzt werden, fehlerfrei zu reagieren.

Hinweis

Auf einen ganzen Datenbaustein kann nur dann gezeigt werden, wenn dieser ursprünglich von einem anwenderdefinierten Datentyp (UDT) abgeleitet wurde.

Hinweis

Peripherie adressieren

Das direkte Lesen oder Schreiben eines Signals von einem Peripherieeingang bzw. -ausgang ist nur auf einer CPU der Baugruppe S7-1500 möglich. (<Operand>:P)

Die folgende Tabelle zeigt die Eigenschaften des VARIANT-Zeigers:

Länge (Byte)	Darstellung	Format	Beispiel für Werteingabe
0	Symbolisch	Operand	"TagResult"
		NameDatenbaustein.NameOperand.Komponente	"Daten_TIA_Portal".StructVariable.ErsteKomponente
	Absolut	Operand	%MW10
		Datenbausteinnummer.Operand Typ Länge	P#DB10.DBX10.0 INT 12 [1]
		NULL-Zeiger	NULL
[1] Wenn Sie das Präfix P# verwenden, können Sie nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen.			

Codierung der Datentypen

Wenn Sie die absolute Adressierung mit P# verwenden, dann sind die folgenden Datentypen erlaubt:

- BOOL
- BYTE
- CHAR
- WORD
- INT
- DWORD
- DINT
- REAL
- TIME
- S5TIME
- DATE
- TOD
- DT

Beispiel

Das folgende Beispiel zeigt die Funktionsweise von VARIANT anhand der AWL Anweisung "MOVE: Wert kopieren":

AWL	Erläuterung
CALL MOVE	// Die Anweisung wird aufgerufen.
value_type := VARIANT	// Datentyp der Parameter IN und OUT
IN := "Daten_TIA_Portal".StructVariable.ErsteKomponente	// Die Inhalte des Operanden "ErsteKomponente" aus dem DB "Daten_TIA_Portal" werden kopiert.

AWL	Erläuterung
`OUT := "MotorDB".StructResult.TagResult`	`// Und in den Operanden "TagResult" aus dem DB "MotorDB" übertragen.`

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Grundlagen zum Bausteinzugriff (Seite 55)

Grundlagen zu Bausteinaufrufen (Seite 62)

VARIANT-Anweisungen (Seite 342)

Peripherie adressieren (Seite 109)

Datentypkonvertierungen bei S7-1200 (Seite 507)

3.11.2.2 Anwendungsfälle für Zeiger im Vergleich

Anwendungsfälle für Zeiger bei einer CPU der Baureihen S7-1200/1500 im Vergleich zu S7-300/400

Die folgende Tabelle gibt Ihnen einen Überblick über die verschiedenen Anwendungsfälle für Zeiger auf einer CPU der Baureihen S7-300/400 (ANY-Zeiger) sowie deren Lösung mit einer CPU der Baureihen S7-1200/1500.

In den meisten Anwendungsfällen wird auf einer CPU der Baureihen S7-1200/1500 die Verwendung eines Zeigers nicht mehr benötigt. Stattdessen stehen deutlich einfachere Sprachmittel zur Verfügung.

Nur bei der indirekten Adressierung, wenn die Datentypen erst zur Programmlaufzeit bestimmt werden, ist es sinnvoll, den Datentyp VARIANT zu verwenden.

Wofür wurde der ANY-Zeiger genutzt?	Empfehlungen im TIA Portal (S7-1200/S7-1500)
Kopieren von Daten beliebigen Quell- und Zieldatentyps im Programm mithilfe der Anweisung "BLKMOV: Bereich kopieren".	Definition von Variablen innerhalb eines PLC-Datentyps. Mithilfe der Anweisung "Serialize" und "Deserialize" können Sie die Variablen kopieren.
Initialisieren einer ARRAY-Struktur	Mithilfe der Anweisung "FILL_BLK: Bereich befüllen" initialisieren bzw. befüllen Sie eine ARRAY-Struktur.
Kopieren von ARRAY-Elementen	Mithilfe der Anweisung "MOVE_BLK: Bereich kopieren" kopieren Sie den Inhalt mehrerer Elemente einer ARRAY-Struktur in eine andere ARRAY-Struktur.

Wofür wurde der ANY-Zeiger genutzt?	Empfehlungen im TIA Portal (S7-1200/S7-1500)
Speicher- und Performanceoptimierung mithilfe von strukturierten Daten	Verwenden Sie den InOut-Abschnitt in der Bausteinschnittstelle, um eine Speicher- und Performanceoptimierung zu erreichen. Weitere Informationen finden Sie im "Programmierleitfaden für S7-1200/1500" unter folgendem Link Programmierleitfaden für S7-1200/1500 (http://support.automation.siemens.com/WW/view/de/81318674)
Zugriff auf einzelne Bit/Byte eines WORD	Verwenden Sie den "Slice-Zugriff" Weitere Informationen finden Sie hier: Beispiel für einen Slice-Zugriff (http://support.automation.siemens.com/WW/view/de/57374718)
Ermittlung der Länge von Strukturen oder Datenbausteinen	Verwenden Sie ein ARRAY und lesen Sie seine Länge mithilfe der Anweisung "CountofElements: Anzahl der ARRAY-Elemente abfragen" aus. Die Anweisung funktioniert nur in Verbindung mit dem Datentyp VARIANT.
Indirekte Adressierung	Für die indirekte Adressierung von Datentypen, die erst zur Laufzeit bekannt sind, können Sie den VARIANT-Zeiger verwenden. Beim indirekten Zugriff auf einen Datenbaustein können Sie den Datentyp DB_ANY verwenden.

3.11.2.3 VARIANT-Anweisungen

VARIANT Anweisungen

Im TIA Portal stehen Ihnen die folgenden Anweisungen zum Arbeiten mit VARIANT zur Verfügung:

Einfache Anweisungen		
Kategorie	**Anweisung**	**Beschreibung**
Vergleicher	EQ_Type	Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen
	NE_Type	Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen
	EQ_ElemType	Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen
	NE_ElemType	Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen
	IS_NULL	Auf GLEICH NULL-Pointer abfragen
	NOT_NULL	Auf UNGLEICH NULL-Pointer abfragen
	IS_ARRAY	Auf ARRAY abfragen
	TypeOf	Datentyp einer VARIANT-Variable abfragen
	TypeOfElements	Elementdatentyp einer VARIANT-Variable abfragen

Einfache Anweisungen		
Kategorie	**Anweisung**	**Beschreibung**
Verschieben	MOVE_BLK_VARIANT	Bereich kopieren
	VariantGet	Wert einer VARIANT-Variable auslesen
	VariantPut	Wert in eine VARIANT-Variable schreiben
	CountOfElements	Anzahl der ARRAY-Elemente abfragen
Umwandler	VARIANT_TO_DB_ANY	VARIANT in DB_ANY konvertieren
	DB_ANY_TO_VARIANT	DB_ANY in VARIANT konvertieren

Hinweis

Unterschiede zwischen MOVE, MOVE_BLK und MOVE_BLK_VARIANT

- Um komplette Strukturen zu kopieren, können Sie die Anweisung MOVE verwenden.
- Um Teile von ARRAYs mit bekanntem Datentyp zu kopieren, können Sie die Anweisung MOVE_BLK verwenden.
- Nur wenn Sie Teile von ARRAYs mit erst zur Programmlaufzeit bekanntem Datentyp kopieren wollen, wird die Anweisung MOVE_BLK_VARIANT benötigt.

Weitere Informationen zu den einzelnen Anweisungen finden Sie im Informationssystem unter "Einfache Anweisungen > jeweilige Programmiersprache".

Weitere Anweisungen, die ebenfalls mit dem Datentyp VARIANT arbeiten, finden Sie auch unter den "Erweiterten Anweisungen".

Siehe auch

Grundlagen zum VARIANT (Seite 339)

Indirekte Adressierung mit dem Datentyp VARIANT (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=42603286&caller=view)

3.11.2.4 Anwendungsmöglichkeiten für VARIANT-Anweisungen

Einleitung

Im folgenden Kapitel erfahren Sie, welche Anwendungsmöglichkeiten Sie mit VARIANT Anweisungen haben.

Datentypen der Variablen auswerten, auf die ein VARIANT zeigt

In der folgenden Tabelle sehen Sie, welche Anweisungen Ihnen zur Verfügung stehen, um die Datentypen von Variablen auszuwerten, auf die ein VARIANT zeigt:

Funktion	Anweisung	Beschreibung
Zur Ermittlung des Datentyps	TypeOf(): Datentyp einer VARIANT-Variable abfragen (Diese Anweisung steht Ihnen nur in SCL zur Verfügung und nur in Verbindung mit einer IF-Anweisung.)	Mit dieser Anweisung vergleichen Sie den Datentyp, auf den eine VARIANT-Variable zeigt, mit dem Datentyp einer beliebigen anderen Variable. Sie können den Vergleich auch mit einem PLC-Datentyp durchführen.
	TypeOfElements(): Elementdatentyp einer VARIANT-Variable abfragen (Diese Anweisung steht Ihnen nur in SCL zur Verfügung und nur in Verbindung mit einer IF-Anweisung.)	Mit dieser Anweisung vergleichen Sie den Datentyp, auf den eine VARIANT-Variable zeigt, mit dem Datentyp einer beliebigen anderen Variable. Sie können den Vergleich auch mit einem PLC-Datentyp durchführen. Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.
	EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen	Mit dieser Anweisung vergleichen Sie den Datentyp, auf den eine VARIANT-Variable zeigt, mit dem Datentyp einer beliebigen anderen Variable. Sie können den Vergleich auch mit einem PLC-Datentyp durchführen.
	EQ_ElemType: Elementdatentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen NE_ElemType: Elementdatentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen	Mit dieser Anweisung vergleichen Sie den Datentyp, auf den eine VARIANT-Variable zeigt, mit dem Datentyp einer beliebigen anderen Variable. Sie können den Vergleich auch mit einem PLC-Datentyp durchführen. Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.
Zur Auswertung von ARRAY-Elementen	IS_ARRAY: Auf ARRAY abfragen	Mit dieser Anweisung überprüfen Sie, ob es sich bei dem Datentyp, auf den eine VARIANT-Variable zeigt, um ein ARRAY handelt.
	CountOfElements: Anzahl der ARRAY-Elemente abfragen	Mit dieser Anweisung lesen Sie aus, wie viele ARRAY-Elemente die Variable hat, auf die die VARIANT-Variable zeigt.

Weitere Informationen zu den einzelnen Anweisungen finden Sie im Informationssystem unter "Einfache Anweisungen > jeweilige Programmiersprache".

Daten lesen, auf die ein VARIANT zeigt

Um die Daten verwenden zu können, müssen diese als Zwischenschritt in eine Variable kopiert werden, da sie nicht direkt verarbeitet werden können.

Anweisung	Beschreibung	Beispiel		Ergebnis
		VARIANT zeigt auf	Zieldatentyp	
VariantGet: Wert einer VARIANT-Variable auslesen	Mit dieser Anweisung kopieren Sie den Wert einer einzelnen Variablen in eine andere Variable. Die Datentypen der beiden Variablen müssen übereinstimmen.	UDT_1	UDT_1	Die Anweisung wird ausgeführt
		REAL	REAL	
		DINT	DWORD	Die Anweisung wird nicht ausgeführt.

Daten einer VARIANT-Variable zuweisen

Sie können die Anweisung nicht dazu verwenden, VARIANT-Variablen zu initialisieren. Die VARIANT-Variablen müssen also bereits initialisiert sein, wenn Sie Daten an die Variable zurückgeben. Sie dürfen keine uninitialisierte temporäre VARIANT-Variable verwenden.

Anweisung	Beschreibung	Beispiel		Ergebnis
		Quelldatentyp	VARIANT zeigt auf:	
VariantPut: Wert in eine VARIANT-Variable schreiben	Mit dieser Anweisung kopieren Sie den Wert einer einzelnen Variable in eine andere Variable. Die Datentypen der beiden Variablen müssen übereinstimmen.	UDT_1	UDT_1	Die Anweisung wird ausgeführt
		REAL	REAL	
		DINT	DWORD	Die Anweisung wird nicht ausgeführt, da die Datentypen unterschiedlich sind.

Dynamische ARRAY-Strukturen verarbeiten

Zur Auswertung von ARRAY-Elementen	TypeOfElements(): Datentyp eines ARRAY-Elements einer VARIANT-Variable abfragen (Diese Anweisung steht Ihnen nur in SCL zur Verfügung und nur in Verbindung mit einer IF-Anweisung.)	Mit dieser Anweisung vergleichen Sie den Datentyp, auf den eine VARIANT-Variable zeigt, mit dem Datentyp einer beliebigen anderen Variable. Sie können den Vergleich auch mit einem PLC-Datentyp durchführen. Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.
	IS_ARRAY: Auf ARRAY abfragen	Mit dieser Anweisung überprüfen Sie, ob es sich bei dem Datentyp, auf den eine VARIANT-Variable zeigt, um ein ARRAY handelt.
	CountOfElements: Anzahl der ARRAY-Elemente abfragen	Mit dieser Anweisung lesen Sie aus, wie viele ARRAY-Elemente die Variable hat, auf die die VARIANT-Variable zeigt.
	MOVE_BLK_VARIANT: Bereich kopieren	Mit dieser Anweisung kopieren Sie dynamisch und typsicher (integrierte Typprüfung) ARRAYs. Die Grenzwerte des Quell- und Ziel-ARRAYs können Sie frei wählen. Die Datentypen der ARRAY-Elemente müssen übereinstimmen.

Hinweis

Unterschiede zwischen MOVE, MOVE_BLK und MOVE_BLK_VARIANT

- Um komplette Strukturen zu kopieren, können Sie die Anweisung MOVE verwenden.
- Um Teile von ARRAYs mit bekanntem Datentyp zu kopieren, können Sie die Anweisung MOVE_BLK verwenden.
- Nur wenn Sie Teile von ARRAYs mit erst zur Programmlaufzeit bekanntem Datentyp kopieren wollen, wird die Anweisung MOVE_BLK_VARIANT benötigt.

Weitere Informationen zur Verwendung der Anweisung MOVE_BLK_VARIANT finden Sie im Programmierbeispiel "Daten verschieben".

Siehe auch

Grundlagen zum VARIANT (Seite 339)

Beispiel für das Verschieben von Daten (Seite 348)

3.11.2.5 VARIANT initialisieren

Beschreibung

Initialisieren Sie den Datentyp VARIANT, indem Sie beim Bausteinaufruf dem VARIANT-Bausteinparameter eine konkrete Variable zuweisen. Damit wird eine Referenz auf die Adresse der übergebenen Variable gebildet. Legen Sie dazu in der Bausteinschnittstelle einen Bausteinparameter vom Datentyp VARIANT an. Im folgenden Beispiel sind das die beiden Bausteinparameter SourceArray und DestinationArray im Abschnitt InOut.

FC_PartialArrayCopy

		Name	Data type
1		Input	
2		Count	UDInt
3		SourceIndex	DInt
4		DestinationIndex	DInt
5		Output	
6		InOut	
7		SourceArray	Variant
8		DestinationArray	Variant
9		Temp	
10		Error	Int
11		Constant	
12		<Add new>	
13		Return	
14		FC_PartialArrayCopy	Int

Das Beispiel zeigt einen Ausschnitt aus dem Programmierbeispiel "Beispiel für das Verschieben von Daten". Den detaillierten Programmcode finden Sie unter "Siehe auch".

Hinweis

Eine direkte Übergabe einer Variable an eine VARIANT-Variable ist nicht möglich, wie z. B. myVARIANT := #Variable

Siehe auch

Beispiel für das Verschieben von Daten (Seite 348)

3.11.2.6 Verschiedene Datentypen mit VARIANT übergeben und auslesen

Die Beispiele zeigen einen Ausschnitt aus dem Programmierbeispiel "Beispiel für das Verschieben von Daten". Den detaillierten Programmcode finden Sie unter "Siehe auch".

Hinweis

VARIANT als Formalparameter

Wenn der VARIANT als Formalparameter deklariert ist, dann dürfen keine schreibgeschützten Daten als Aktualparameter übergeben werden.

Übergabe verschiedener Datentypen

Im folgenden Beispiel sehen Sie, wie beim mehrmaligen Aufruf einer generischen, standardisierten Funktion der VARIANT-Bausteinparameter jeweils mit verschiedenen Variablen initialisiert werden kann:

Die Funktion "FC_PartialArrayCopy" wird zweimal aufgerufen. Beim linken Aufruf wird der VARIANT-Parameter SourceArray mit einem ARRAY of "my_struct" verschalten. Beim rechten Aufruf wird der VARIANT-Parameter SourceArray mit einem ARRAY of REAL verschalten.

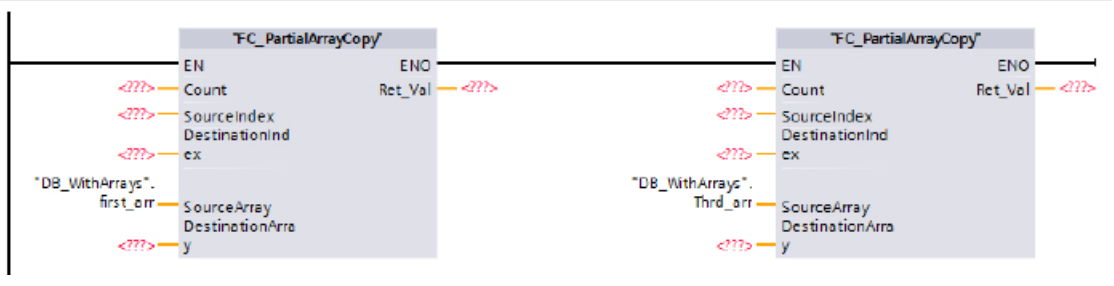

Datentypen auslesen und prüfen

Um den Datentyp einer Variable, bzw. eines Elements auszulesen und mit Datentypen anderer Variablen, bzw. Elemente zu vergleichen, stehen Ihnen verschiedene Vergleichsanweisungen zur Verfügung.

Im folgenden Bild sehen Sie die Verwendung mehrerer Vergleichsanweisungen um zu überprüfen, ob die Elemente der ARRAYs den gleichen Datentyp haben:

```
6  IF IS_ARRAY(#SourceArray) AND IS_ARRAY(#DestinationArray)
7      AND TypeOfElements(#SourceArray) = TypeOfElements(#DestinationArray) THEN
8      #Error := MOVE_BLK_VARIANT(COUNT := #Count, SRC := #SourceArray, SRC_INDEX := #SourceIndex,
9                                 DEST => #DestinationArray, DEST_INDEX := #DestinationIndex);
10 END_IF;
```

Nur, wenn die Datentypen der ARRAY-Elemente gleich sind, wird die Anweisung MOVE_BLK_VARIANT ausgeführt.

Siehe auch

Beispiel für das Verschieben von Daten (Seite 348)

3.11.2.7 Beispiele für die Verwendung des Datentyps VARIANT

Beispiel für das Verschieben von Daten

Programmierbeispiel

Im folgenden Programmierbeispiel werden Datenwerte, die z. B. während einer Produktionsschicht gesammelt wurden, zur Weiterverarbeitung verschoben. Die Daten werden in einem ARRAY gesammelt und mit Hilfe der Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" können Sie entweder das gesamte ARRAY oder nur einzelne ARRAY-Elemente dynamisch und typsicher kopieren. Die ARRAY-Grenzen des jeweiligen Quell- und Ziel-ARRAYs können frei gewählt werden und müssen nicht übereinstimmen. Der Datentyp der Datenwerte, die kopiert werden sollen, muss allerdings übereinstimmen. Diese Anweisung steht Ihnen in allen Programmiersprachen zur Verfügung.

Durch die Verwendung des Datentyps VARIANT können Sie den erstellten Programmcode zum Verschieben von Daten auch für andere Produktionsschichten verwenden, indem Sie am Bausteinaufruf einen anderen Quell- und Zielbereich angeben.

Vorgehensweise

1. Legen Sie eine Funktion mit der Programmiersprache SCL an und geben Sie ihr den Namen "FC_PartialArrayCopy".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_PartialArrayCopy

		Name	Data type
1		Input	
2		Count	UDInt
3		SourceIndex	DInt
4		DestinationIndex	DInt
5		Output	
6		InOut	
7		SourceArray	Variant
8		DestinationArray	Variant
9		Temp	
10		Error	Int
11		Constant	
12		<Add new>	
13		Return	
14		FC_PartialArrayCopy	Int

3. Erstellen Sie den SCL-Programmcode wie folgt:
   Untenstehend finden Sie den Programmcode als Kopiervorlage.

```
6  IF IS_ARRAY(#SourceArray) AND IS_ARRAY(#DestinationArray)
7      AND TypeOfElements(#SourceArray) = TypeOfElements(#DestinationArray) THEN
8      #Error := MOVE_BLK_VARIANT(COUNT := #Count, SRC := #SourceArray, SRC_INDEX := #SourceIndex,
9                                 DEST => #DestinationArray, DEST_INDEX := #DestinationIndex);
10 END_IF;
11 #FC_PartialArrayCopy := #Error;
```

4. Legen Sie den PLC-Datentyp "UDT_MyStruct" an:

UDT_MyStruct

	Name	Data type	Default value
1	a	Bool	1
2	b	Bool	false
3	c	Bool	1
4	d	Bool	false
5	e	Bool	1
6	arr_int	Array[1..100] of Int	
7	f	Int	0
8	g	Real	0.0
9	h	LReal	0.0

5. Legen Sie den globalen Datenbaustein "DB_WithArrays" an:

DB_WithArrays

	Name	Data type
1	Static	
2	first_arr	Array[0..100] of "UDT_MyStruct"
3	Sec_arr	Array[1..200] of "UDT_MyStruct"
4	Thrd_arr	Array[1..200] of Real
5	For_arr	Array[1..500] of Real

6. Rufen Sie die Funktion "FC_PartialArrayCopy" in einem Organisationsbaustein, z. B. OB1, auf und initialisieren Sie die Parameter mit dem Datenbaustein "DB_WithArrays". Geben Sie die genannten Konstanten an:

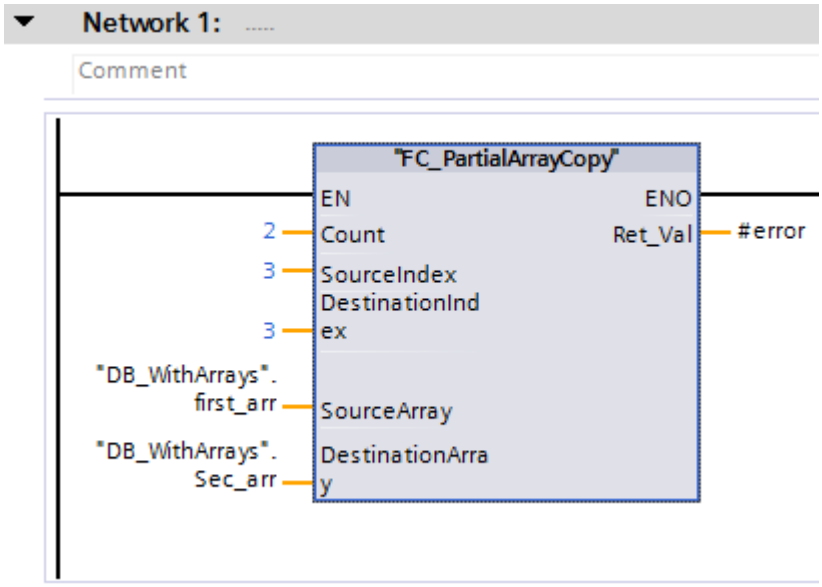

7. Sie können statt der ersten beiden ARRAYs, die den Datentyp UDT_MyStruct haben, auch das dritte und vierte ARRAY verwenden, die den Datentyp REAL haben:

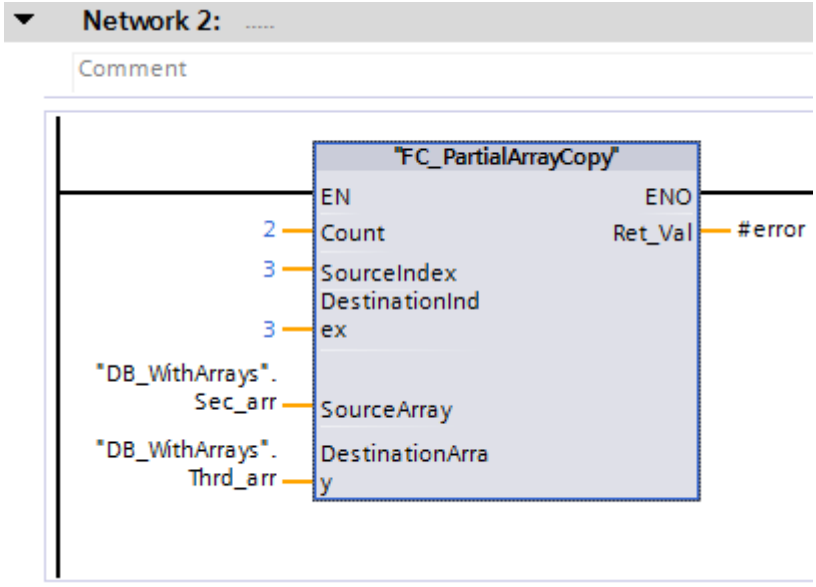

Ergebnis

Sobald der Baustein "FC_PartialArrayCopy" im Programmzyklus aufgerufen wird, werden zwei Datenwerte, beginnend mit dem vierten Element, aus dem ersten ARRAY des globalen Datenbausteins "DB_WithArrays" in das zweite ARRAY des Datenbausteins kopiert. Die kopierten Datenwerte werden im zweiten ARRAY ab dem vierten Element eingefügt.

SCL-Programmcode zum Kopieren:

SCL

```
IF IS_ARRAY(#SourceArray) AND TypeOfElements(#SourceArray) =
TypeOfElements(#DestinationArray) THEN
#Error := MOVE_BLK_VARIANT(COUNT := #Count, SRC := #SourceArray, SRC_INDEX := #SourceIndex,
                           DEST => #DestinationArray, DEST_INDEX := #DestinationIndex);
END_IF;
#FC_PartialArrayCopy := #Error;
```

Siehe auch

Grundlagen zum VARIANT (Seite 339)

Beispiel für das Programmieren einer Warteschlange (FIFO)

Programmierbeispiel

Im folgenden Beispiel programmieren Sie einen Ring-Puffer, der aus einem ARRAY besteht und nach dem FIFO-Prinzip geschrieben und gelesen wird. Der Programmcode hat jeweils einen Lese- und einen Schreib-VARIANT-Zeiger. Mithilfe der VARIANT-Anweisungen können Sie den Programmcode robust programmieren und ein sicheres Kopieren oder Löschen gewährleisten.

Mithilfe des Datentyps VARIANT können Programmteile zur Laufzeit beeinflusst werden. Der VARIANT-Zeiger ist ein typsicherer Zeiger, d. h. zur Laufzeit erfolgt eine Typprüfung. Bei Bausteinen, die mit der Bausteineigenschaft "Optimiert" angelegt worden sind, können Teilfunktionen, die bisher mit einem ANY-Zeiger programmiert wurden, jetzt mit einem VARIANT-Zeiger gelöst werden. Der Datentyp VARIANT wird zur Übergabe von Strukturen an Systemfunktionsbausteine verwendet.

Vorgehensweise

1. Legen Sie einen SCL-Funktionsbaustein an und benennen ihn "FIFOQueue".
2. Deklarieren Sie die Bausteinschnittstelle wie folgt:

Deklaration	Parameter	Datentyp	Kommentar
Input	request	BOOL	Die Anweisung wird ausgeführt, wenn am Parameter "request" eine positive Signalflanke erfasst wird.
	mode	BOOL	0 = Der erste Eintrag aus dem Ring-Puffer wird zurückgegeben. 1 = Ein Eintrag wird an letzter Stelle in den Ring-Puffer geschrieben.
	initialValue	VARIANT	Wert, mit dem das ARRAY des Ring-Puffers initialisiert wird.
Output	error	INT	Fehlerinformation
InOut	item	VARIANT	Der Eintrag, der entweder aus dem Ring-Puffer zurückgegeben oder in den Ring-Puffer geschrieben wird.
	buffer	VARIANT	Ein ARRAY, das als Ring-Puffer verwendet wird.
Static	edgeupm	BOOL	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
	firstItemIndex	INT	Index des ältesten Eintrags im Ring-Puffer
	nextEmptyItemIndex	INT	Index des nächsten freien Elements im Ring-Puffer
Temp	edgeup	BOOL	Ergebnis der Flankenauswertung
	internalError	INT	Fehlerinformation
	newFirstItemIndex	INT	Variabler Index
	newNextEmptyItemIndex	INT	Variabler Index
	bufferSize	UDINT	Anzahl der ARRAY-Elemente im Ring-Puffer

3. Erstellen Sie im Funktionsbaustein "FIFOQueue" den folgenden Programmcode:

```
(* Dieser Programmcodeabschnitt wird nur einmalig nach einer positiven
Signalflanke, ausgeführt. Wenn keine Änderung im Signalzustand des
Verknüpfungsergebnisses vorliegt, dann wird die Programmbearbeitung des FB
"FIFOQueue" beendet. *)
#edgeup := #request & NOT #edgeupm;
```

```
#edgeupm := #request;
IF NOT (#edgeup) THEN
   RETURN;
END_IF;


// ------Validierung, ob alle Parametereingänge gültig sind.----
(* Dieser Programmcodeabschnitt überprüft, ob der Ring-Puffer ein ARRAY
ist. Wenn ja, dann wird die Zahl der ARRAY-Elemente ausgelesen. Wenn es sich
nicht um ein ARRAY handelt, dann wird die Programmbearbeitung an dieser
Stelle beendet und der Fehlercode "-10" ausgegeben. *)
IF NOT (IS_ARRAY(#buffer)) THEN
   #error := -10;
   RETURN;
ELSE
   #bufferSize := CountofElements(#buffer);
END_IF;

(* Dieser Programmcodeabschnitt überprüft, ob der Datentyp der ARRAY-
Elemente mit dem Datentyp des Eintrags (Variable #item) übereinstimmt. Wenn
die Datentypen nicht übereinstimmen, dann wird die Programmbearbeitung an
dieser Stelle beendet und der Fehlercode "-11" ausgegeben. *)
IF NOT (TypeOf(#item) = TypeOfElements(#buffer)) THEN
   #error := -11;
   RETURN;
END_IF;

(* Dieser Programmcodeabschnitt überprüft, ob der Initial-Wert des Ring-
Puffers mit dem Eintrag (Variable #item) übereinstimmt. Wenn die Datentypen
nicht übereinstimmen, dann wird die Programmbearbeitung an dieser Stelle
beendet und der Fehlercode "-12" ausgegeben. *)
IF NOT (TypeOf(#item) = TypeOf(#initialValue)) THEN
   #error := -12;
   RETURN;
END_IF;

(* Dieser Programmcodeabschnitt überprüft, ob die variablen Indices
innerhalb der ARRAY-Grenzen liegen. Wenn das nicht der Fall ist, dann wird
die Programmbearbeitung an dieser Stelle beendet und je nach Index die
Fehlercodes "-20" oder "-21" ausgegeben. *)
IF (#nextEmptyItemIndex >= #bufferSize) THEN
   #error := -20;
   RETURN;
END_IF;
IF (#firstItemIndex >= #bufferSize) THEN
   #error := -21;
   RETURN;
```

```
END_IF;


//-----------Programmcodeausführung, abhängig vom Parameter
Mode-------------
// Die Ausführung der Anweisungen hängt vom Signalzustand des Parameters
Mode ab.
IF #mode = 0 THEN

// Wenn der Parameter Mode den Signalzustand "0" hat, dann wird der erste
Eintrag aus dem übergegebenen Ring-Puffer zurückgegeben.
(* Dieser Programmcodeabschnitt überprüft, ob der Ring-Puffer leer ist.
Wenn das der Fall ist, dann wird die Programmbearbeitung an dieser Stelle
beendet und der Fehlercode "-40" ausgegeben. *)
   IF (#firstItemIndex = -1) THEN
      #error := -40;
      RETURN;
   END_IF;

// Dieser Programmcodeabschnitt gibt den ersten Eintrag des Ring-Puffers
zurück.
   #internalError := MOVE_BLK_VARIANT(SRC := #buffer,
                                      COUNT := 1,
                                      SRC_INDEX := #firstItemIndex,
                                      DEST_INDEX := 0,
                                      DEST => #item);

   IF (#internalError = 0) THEN
(* Dieser Programmcodeabschnitt überprüft, ob der Ring-Puffer ARRAY-
Elemente enthält. Wenn ja, dann wird der erste Eintrag weitergeschoben und
der Index um 1 erhöht. *)
      #internalError := MOVE_BLK_VARIANT(SRC := #initialValue,
                                         COUNT := 1,
                                         SRC_INDEX := 0,
                                         DEST_INDEX := #firstItemIndex,
                                         DEST => #buffer);

// Dieser Programmcodeabschnitt berechnet den neuen Index des ersten
Eintrags.
      #newFirstItemIndex := #firstItemIndex +1;
      #newFirstItemIndex := #newFirstItemIndex MOD
UDINT_TO_INT(#bufferSize);

// Dieser Programmabschnitt überprüft, ob der Ring-Puffer leer ist.
      IF (#nextEmptyItemIndex = #newFirstItemIndex) THEN
// Wenn der Ring-Puffer leer ist, dann wird der Index auf 0 gesetzt.
```

```
         #firstItemIndex := -1;
         #nextEmptyItemIndex := 0;
      ELSE
// Der Index des ersten Eintrags wird geändert.
         #firstItemIndex := #newFirstItemIndex;
      END_IF;
   END_IF;
ELSE


// Wenn der Parameter Mode den Signalzustand "1" hat, dann wird der Eintrag
in den übergegebenen Ring-Puffer geschrieben.
(* Dieser Programmcodeabschnitt überprüft, ob der Ring-Puffer voll ist.
Wenn das der Fall ist, dann wird die Programmbearbeitung an dieser Stelle
beendet und der Fehlercode "-50" ausgegeben. *)
   IF (#nextEmptyItemIndex = #firstItemIndex) THEN
      #error := -50;
      RETURN;
   END_IF;

// Dieser Programmcodeabschnitt schreibt den Eintrag in den Ring-Puffer.
   #internalError := MOVE_BLK_VARIANT(SRC := #item,
                                      COUNT := 1,
                                      SRC_INDEX := 0,
                                      DEST_INDEX := #nextEmptyItemIndex,
                                      DEST => #buffer);

   IF (#internalError = 0) THEN
// Dieser Programmcodeabschnitt erhöht den Index um 1 und berechnet den
neuen leeren Eintrag-Index.
      #newNextEmptyItemIndex := #nextEmptyItemIndex +1;
      #newNextEmptyItemIndex := #newNextEmptyItemIndex MOD #bufferSize;
      #nextEmptyItemIndex := #newNextEmptyItemIndex;

(* Dieser Programmcodeabschnitt prüft, welchen Index die Variable
"#firstItemIndex" hat. Wenn die Zahl = -1 ist, dann wurde der Ring-Puffer
initialisiert und der Eintrag wird in den Ring-Puffer geschrieben. Daher
muss der Variable "0" zugewiesen werden. *)
      IF (#firstItemIndex = -1) THEN
         #firstItemIndex := 0;
      END_IF;
   END_IF;
END_IF;

//-------------------------Lokale
Fehlerbehandlung----------------------------
```

```
(* Dieser Programmcodeabschnitt überprüft, ob ein lokaler Fehler
aufgetreten ist. Wenn das der Fall ist, dann wird das Programm an dieser
Stelle beendet und der Fehlercode "-100" ausgegeben. *)
IF (#internalError > 0) THEN
   #error := -100;
   RETURN;
END_IF;

// Wenn kein Fehler während der Programmbearbeitung aufgetreten ist, dann
wird der Fehlercode "0" ausgegeben.
#error := 0;
```

Ergebnis

Rufen Sie den SCL-Funktionsbaustein an der Stelle in Ihrem Programm auf, an der die FIFO-Warteschlange ablaufen soll.

3.11.3 POINTER

Beschreibung

Ein Parameter vom Typ POINTER ist ein Zeiger, der auf eine bestimmte Variable zeigen kann. Er belegt 6 Byte (48 Bit) im Speicher und kann die folgenden Informationen zu einer Variablen enthalten:

- DB-Nummer bzw. 0, wenn die Daten nicht in einem DB gespeichert werden
- Speicherbereich in der CPU
- Variablenadresse

Das folgende Bild zeigt die Struktur des Parametertyps POINTER:

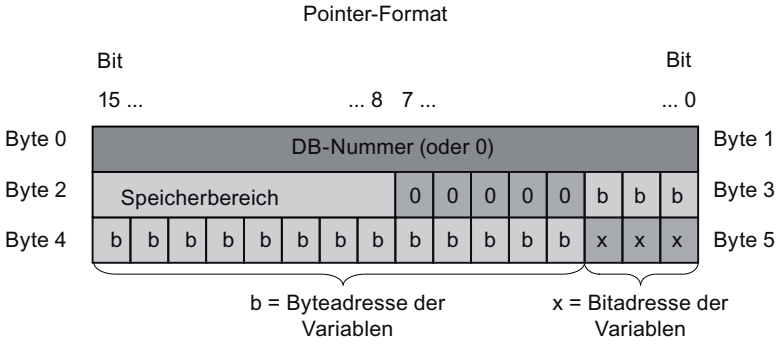

Arten von Zeigern

Abhängig von der Angabe können Sie mit dem Parametertyp POINTER die folgenden vier Arten von Zeigern deklarieren:

- Bereichsinterner Zeiger:
  Ein bereichsinterner Zeiger enthält Angaben zur Adresse einer Variablen.
- Bereichsübergreifender Zeiger:
  Ein bereichsübergreifender Zeiger enthält Angaben zu dem Speicherbereich und der Adresse eines Operanden.
- DB-Zeiger:
  Mit einem DB-Zeiger können Sie auf eine Variable eines Datenbausteins zeigen. Ein DB-Zeiger enthält zusätzlich zu dem Speicherbereich und der Adresse einer Variablen auch eine Datenbausteinnummer.
- Null Zeiger:
  Mit einem Null Zeiger zeigen Sie das Fehlen eines Wertes an. Das Fehlen eines Wertes kann entweder bedeuten, dass kein Wert existiert oder dass der Wert noch nicht bekannt ist. Ein Nullwert steht für die Abwesenheit eines Wertes, ist aber gleichzeitig ein Wert.

Die folgende Tabelle zeigt die Formate zur Deklaration der verschiedenen Zeiger-Arten:

P#ByteDarstellung	Format	Beispiel für Werteingabe	Beschreibung
Symbolisch	P#Byte.Bit	"MyTag"	Bereichsinterner Zeiger
	P#Operandenbereich-Byte.Bit	"MyVariable"	Bereichsübergreifender Zeiger
	P#Datenbaustein.Datenoperand	"MyDB"."MyTag"	DB-Zeiger
	P#Nullwert	-	Null Zeiger
Absolut	P#Byte.Bit	P#20.0	Bereichsinterner Zeiger
	P#Operandenbereich-Byte.Bit	P#M20.0	Bereichsübergreifender Zeiger
	P#Datenbaustein.Datenoperand	P#DB10.DBX20.0	DB-Zeiger
	P#Nullwert	P#0.0, NULL	Null Zeiger

Hinweis

SCL: Zugriff auf Datenbausteine über absolute DB-Pointer

Wenn Sie einen Datenbaustein aus dem Ordner "Systembausteine" in SCL über einen absoluten DB-Pointer (z. B. P#DB1.dbx0.0) adressieren und dies der einzige Zugriff auf den DB ist, wird in seltenen Fällen der Datenbaustein beim nächsten Übersetzungslauf gelöscht. Der Compiler erkennt den DB-Pointer nicht als Zugriff und nimmt deshalb an, dass der DB im Programm nicht verwendet wird.Verwenden Sie statt des Pointers einen lesenden Zugriff auf den DB (z.B. “MyDB.MyFirstMember“). Diese Form des Zugriffs wird in jedem Fall korrekt durchgeführt.

Das Präfix P#

Geben Sie bei der Versorgung eines Formalparameters vom Datentyp POINTER im Bausteinaufruf den Aktualwert ohne Präfix P# ein. Ihr Eintrag wird dann automatisch in das POINTER-Format umgewandelt.

Wenn Sie das Präfix P# verwenden, können Sie nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen. Weitere Informationen und einen direkten Vergleich der beiden Zugriffsmöglichkeiten finden Sie hier:

- Grundlagen zum Bausteinzugriff (Seite 55)
- Die neuen Funktionen der S7-1200/1500-CPU und Programmierempfehlungen im Überblick (Seite 205)

In der Programmiersprache AWL ist für die Verwendung des Präfix P# folgendes zu beachten:

Bausteinart	Zugriffsart "optimiert"	Zugriffsart "standard"
Funktion (FC)	Die Verwendung einer Variablen ist im Programmcode mit dem Präfix P# nicht möglich.	Folgende Variablen können im Programmcode mit dem Präfix P# verwendet werden: • Strukturierte Variablen, die im Abschnitt InOut der Bausteinschnittstelle deklariert wurden. Variablen mit elementaren Datentypen können nicht mit dem Präfix P# verwendet werden.
Funktionsbaustein (FB)	Die Verwendung einer Variablen ist im Programmcode mit dem Präfix P# nicht möglich.	Folgende Variablen können im Programmcode mit dem Präfix P# verwendet werden: • Variablen, die in den Abschnitten Static, Input und Output der Bausteinschnittstelle deklariert wurden. • Variablen mit dem Datentyp PLC-Datentyp (UDT), die im Abschnitt InOut der Bausteinschnittstelle deklariert wurden.

Weitere Informationen zu strukturierten und elementaren Datentypen finden Sie hier: Parameter als Kopie oder als Pointer übergeben (Seite 84)

Speicherbereiche

Die folgende Tabelle zeigt die Hexadezimalcodes der Speicherbereiche für den Parametertyp POINTER:

Hexadezimalcode	Speicherbereich	Beschreibung
B#16#80[1)]	P	Peripherie auf einer CPU S7-300/400
16#1	P	Peripherieeingänge auf einer CPU S7-1500
16#2	P	Peripherieausgänge auf einer CPU S7-1500
B#16#81	E	Speicherbereich der Eingänge
B#16#82	A	Speicherbereich der Ausgänge

Hexadezimalcode	Speicherbereich	Beschreibung
B#16#83	M	Speicherbereich der Merker
B#16#84	DBX	Datenbaustein
B#16#85	DIX	Instanz-Datenbaustein
B#16#86	L	Lokaldaten
B#16#87	V	Vorherige Lokaldaten
[1] Diese Datentypen können für den POINTER-Zeiger nur auf einer CPU S7-300/400 verwendet werden.		

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

3.11.4 ANY

Beschreibung

Ein Parameter vom Typ ANY ist ein Zeiger, der auf den Anfang eines Datenbereichs zeigt und dessen Länge nennt. Ein ANY-Zeiger belegt 10 Byte im Speicher und kann die folgenden Informationen enthalten:

- Datentyp:
  Datentyp der Elemente des Datenbereichs
- Wiederholfaktor:
  Anzahl der Elemente des Datenbereichs
- DB-Nummer:
  Datenbaustein, in dem die Elemente des Datenbereichs deklariert sind.
- Speicherbereich:
  Speicherbereich der CPU, in dem die Elemente des Datenbereichs abgelegt sind.
- Anfangsadresse der Daten in Format "Byte.Bit":
  Anfang des Datenbereichs, auf den der ANY-Zeiger zeigt.
- Null Zeiger:
  Mit einem Null Zeiger zeigen Sie das Fehlen eines Wertes an. Das Fehlen eines Wertes kann entweder bedeuten, dass kein Wert existiert oder dass der Wert noch nicht bekannt ist. Ein Nullwert steht für die Abwesenheit eines Wertes, ist aber gleichzeitig ein Wert.

Hinweis

ANY-Zeiger auf einen STRING

Wenn ein ANY-Zeiger in einem KOP-/FUP-/AWL- oder SCL-Netzwerk durch die Zuweisung eines Strings gebildet wird, entsteht ein ANY-Zeiger vom Datentyp BYTE und der Wiederholfaktor maxstringlen.

Wenn ein ANY-Zeiger in einem SCL-Baustein durch die Zuweisung eines Strings gebildet wird, dann entsteht ein ANY-Zeiger auf einen STRING.

Ein Zeiger dieser Art kann z. B. durch die Übergabe aus einem SCL-Baustein an ein KOP-Netzwerk, oder durch das Überlagern mit einem AT-Konstrukt und manuelles Setzen des Datentyps auf STRING, entstehen.

Hinweis

Zugriffsmöglichkeiten

Der ANY-Zeiger darf auch bei einer CPU der Baureihe S7-1500 nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen. Weitere Informationen und einen direkten Vergleich der beiden Zugriffsmöglichkeiten finden Sie hier:

- Grundlagen zum Bausteinzugriff (Seite 55)
- Die neuen Funktionen der S7-1200/1500-CPU und Programmierempfehlungen im Überblick (Seite 205)

In den Programmiersprachen SCL und AWL können Sie bei einem Bausteinaufruf beliebige Speicher übergeben, wenn Sie an einem Bausteinparameter einen ANY-Zeiger programmiert haben.

Der ANY-Zeiger kann jedoch keine Information über die Struktur des Speichers speichern. Der ANY-Zeiger speichert z. B. nicht, dass er auf eine Variable vom Datentyp PLC-Datentyp zeigt. Der ANY-Zeiger betrachtet diesen als ARRAY of BYTE.

Parameter vom Datentyp ANY können an Systemfunktionsbausteine (SFBs) bzw. Systemfunktionen (SFCs) weitergegeben werden.

Das folgende Bild zeigt die Struktur des ANY-Zeigers:

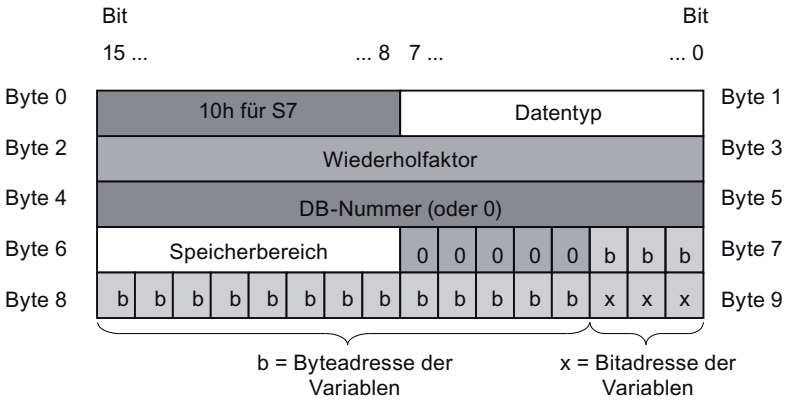

Ein ANY-Zeiger kann keine Strukturen erkennen. Er kann nur lokalen Variablen zugewiesen werden.

Die folgende Tabelle zeigt die Formate zur Deklaration eines ANY-Zeigers:

Darstellung	Format	Beispiel für Werteingabe	Beschreibung
Symbolisch	P#Datenbaustein.Speicherbereich Datenadresse Typ Anzahl	"MyDB".StructVariable.ErsteKomponente	Bereich mit 10 Wörtern im globalen DB11 ab DBB20.0
	P#Speicherbereich Datenadresse Typ Anzahl	"MyMarkerTag"	Bereich mit 4 Bytes ab MB20.0
		"MyTag"	Eingang E1.0
	P#Nullwert	-	Nullwert
Absolut	P#Datenbaustein.Speicherbereich Datenadresse Typ Anzahl	P#DB11.DBX20.0 INT 10	Bereich mit 10 Wörtern im globalen DB11 ab DBB20.0
	P#Speicherbereich Datenadresse Typ Anzahl	P#M20.0 BYTE 10	Bereich mit 10 Bytes ab MB20.0
		P#E1.0 BOOL 8	Bereich mit 8 Bits ab Eingang E1.0 (Die angegebene Länge des Bereichs muss durch 8 teilbar sein.)
	P#Nullwert	P#P0.0 VOID 0, NULL [1]	Nullwert
[1] In den Programmiersprachen KOP und FUP ist als Werteingabe beim Nullwert nur NULL gültig.			

Codierung der Datentypen

Die folgende Tabelle zeigt die Codierung der Datentypen für den ANY-Zeiger:

Hexadezimalcode	Datentyp	Beschreibung
B#16#00	NIL	Nullpointer
B#16#01[1]	BOOL	Bits
B#16#02	BYTE	Bytes, 8 Bits
B#16#03	CHAR	8-Bit-Zeichen
B#16#04	WORD	16-Bit-Wörter
B#16#05	INT	16-Bit-Ganzzahlen
B#16#06	DWORD	32-Bit-Wörter
B#16#07	DINT	32-Bit-Ganzzahlen
B#16#08	REAL	32-Bit-Gleitpunktzahlen
B#16#0B	TIME	Zeitdauer
B#16#0C	S5TIME	Zeitdauer
B#16#09	DATE	Datum
B#16#0A	TOD	Datum und Uhrzeit
B#16#0E	DT	Datum und Uhrzeit
B#16#13	STRING	Zeichenkette
B#16#17[1]	BLOCK_FB	Funktionsbaustein
B#16#18[1]	BLOCK_FC	Funktion
B#16#19[1]	BLOCK_DB	Datenbaustein

Hexadezimalcode	Datentyp	Beschreibung
B#16#1A[1)]	BLOCK_SDB	Systemdatenbaustein
B#16#1C[1)]	COUNTER	Zähler
B#16#1D[1)]	TIMER	Zeit
[1)] Diese Datentypen können für den ANY-Zeiger nur auf einer CPU S7-300/400 verwendet werden.		

Codierung der Speicherbereiche

Die folgende Tabelle zeigt die Codierung der Speicherbereiche für den ANY-Zeiger:

Hexadezimalcode	Bereich	Beschreibung
B#16#80[1)]	P	Peripherie
B#16#81	E	Speicherbereich der Eingänge
B#16#82	A	Speicherbereich der Ausgänge
B#16#83	M	Speicherbereich der Merker
B#16#84	DBX	Datenbaustein
B#16#85[1)]	DIX	Instanz-Datenbaustein
B#16#86	L	Lokaldaten
B#16#87	V	Vorherige Lokaldaten
[1)] Diese Speicherbereiche können für den ANY-Zeiger nur auf einer CPU S7-300/400 verwendet werden.		

Siehe auch

Grundlagen zur indirekten Adressierung (Seite 119)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu Konstanten (Seite 102)

Parameter als Kopie oder als Pointer übergeben (Seite 84)

3.12 Parametertypen

3.12.1 Parametertypen

Beschreibung

Die Parametertypen sind Datentypen für Formalparameter, die an aufgerufene Bausteine übergeben werden. Ein Parametertyp kann auch ein PLC-Datentyp sein.

Die folgende Tabelle zeigt verfügbare Parametertypen und deren Bedeutung:

Parametertyp	Länge (Bits)	Beschreibung
TIMER	16	Dient zur Angabe einer Zeit, die im aufgerufenen Codebaustein verwendet wird. Wenn Sie einen Formalparameter vom Parametertyp TIMER versorgen, muss der zugehörige Aktualparameter eine Zeit sein. Beispiel: T1
COUNTER	16	Dient zur Angabe eines Zählers, der im aufgerufenen Codebaustein verwendet wird. Wenn Sie einen Formalparameter vom Parametertyp COUNTER versorgen, muss der zugehörige Aktualparameter ein Zähler sein. Beispiel: Z10
BLOCK_FC	16	Dient zur Angabe eines Bausteins, der im aufgerufenen Codebaustein als Eingang verwendet wird. Die Deklaration des Parameters bestimmt die Bausteinart (z. B. FB, FC, DB), die verwendet werden soll. Wenn Sie einen Formalparameter vom Parametertyp BLOCK versorgen, geben Sie eine Bausteinadresse als Aktualparameter an. Beispiel: DB3
BLOCK_FB	16	
BLOCK_DB	16	
BLOCK_SDB	16	

Hinweis

Parametertyp “Block_DB” zur Übergabe eines Instanz-DBs

In KOP und FUP können Sie den Instanz-DB eines Funktionsbausteins nicht über einen Eingang des Datentyps „Block_DB“ angeben. Verwenden Sie stattdessen eine Parameterinstanz.

Siehe auch: Parameterinstanzen (Seite 70)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

3.13 Systemdatentypen

3.13.1 Systemdatentypen

Beschreibung

Die Systemdatentypen (SDT) werden vom System zur Verfügung gestellt und haben eine vordefinierte Struktur. Die Struktur eines Systemdatentyps besteht aus einer festen Anzahl von Komponenten, die unterschiedliche Datentypen aufweisen können. Eine Änderung der Struktur eines Systemdatentyps ist nicht möglich.

Systemdatentypen können einander nur zugewiesen werden, wenn sie vom gleichen Typ sind und ihr Name übereinstimmt. Dies gilt auch für vom System generierte PLC-Datentypen, z. B. IEC_Timer, etc.

Die Systemdatentypen können nur für bestimmte Anweisungen verwendet werden. Die folgende Tabelle zeigt verfügbare Systemdatentypen und deren Bedeutung:

Systemdatentyp	Länge (Byte)	Beschreibung
IEC_TIMER	16	Struktur eines Zeitgebers, in der die Parameter PT, ET, IN und Q deklariert sind. Die Zeitwerte sind vom Datentyp TIME. Dieser Datentyp wird z. B. für die Anweisungen "TP", "TOF", "TON", "TONR", "RT" und "PT" verwendet.
IEC_LTIMER	32	Struktur eines Zeitgebers, in der die Parameter PT, ET, IN und Q deklariert sind. Die Zeitwerte sind vom Datentyp LTIME. Dieser Datentyp wird z. B. für die Anweisungen "TP", "TOF", "TON", "TONR", "RT" und "PT" verwendet.
IEC_SCOUNTER	3	Struktur eines Zählers, dessen Zählwerte vom Datentyp SINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_USCOUNTER	3	Struktur eines Zählers, dessen Zählwerte vom Datentyp USINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_COUNTER	6	Struktur eines Zählers, dessen Zählwerte vom Datentyp INT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_UCOUNTER	6	Struktur eines Zählers, dessen Zählwerte vom Datentyp UINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_DCOUNTER	12	Struktur eines Zählers, dessen Zählwerte vom Datentyp DINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.

Systemdatentyp	Länge (Byte)	Beschreibung
IEC_UDCOUNTER	12	Struktur eines Zählers, dessen Zählwerte vom Datentyp UDINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_LCOUNTER	24	Struktur eines Zählers, dessen Zählwerte vom Datentyp LINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
IEC_ULCOUNTER	24	Struktur eines Zählers, dessen Zählwerte vom Datentyp ULINT sind. Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.
ERROR_STRUCT	28	Struktur einer Fehlerinformation zu einem Programmier- oder Peripheriezugriffsfehler. Dieser Datentyp wird z. B. für die Anweisung "GET_ERROR" verwendet.
CREF	8	Komponente des Datentyps ERROR_STRUCT, in der Informationen zur Adresse eines Bausteins gespeichert werden.
NREF	8	Komponente des Datentyps ERROR_STRUCT, in der Informationen zur Adresse eines Operanden gespeichert werden.
VREF	12	Dient zur Ablage eines VARIANT-Zeigers. Dieser Datentyp wird z. B. für Anweisungen von S7-1200/1500 Motion Control verwendet.
SSL_HEADER	4	Gibt die Datenstruktur an, in der Informationen zu den Datensätzen beim Lesen von Systemzustandslisten gespeichert werden. Dieser Datentyp wird z. B. für die Anweisung "RDSYSST" verwendet.
CONDITIONS	52	Definierte Datenstruktur, die Bedingungen für Anfang und Ende eines Datenempfangs definiert. Dieser Datentyp wird z. B. für die Anweisung "RCV_CFG " verwendet.
TADDR_Param	8	Gibt die Struktur eines Datenbausteins vor, in dem Verbindungsbeschreibungen für die Open User Communication über UDP abgelegt werden. Dieser Datentyp wird z. B. für die Anweisungen "TUSEND" und "TURSV" verwendet.
TCON_Param	64	Gibt die Struktur eines Datenbausteins vor, in dem Verbindungsbeschreibungen für die Open User Communication über Industrial Ethernet (PROFINET) abgelegt werden. Dieser Datentyp wird z. B. für die Anweisungen "TSEND" und "TRSV" verwendet.
HSC_Period	12	Gibt die Struktur eines Datenbausteins für die Periodendauermessung mit dem erweiterten Schnellen Zähler vor. Dieser Datentyp wird z. B. für die Anweisung "CTRL_HSC_EXT" verwendet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

3.14 Hardware-Datentypen

3.14.1 Hardware-Datentypen

Beschreibung

Die Hardware-Datentypen werden von der CPU zur Verfügung gestellt. Die Anzahl der verfügbaren Hardware-Datentypen hängt dabei von der CPU ab.

Abhängig von den in der Hardware-Konfiguration parametrierten Baugruppen, werden Konstanten von einem bestimmten Hardware-Datentyp angelegt. Wenn eine Anweisung zur Steuerung oder Aktivierung einer parametrierten Baugruppe ins Anwenderprogramm eingefügt wird, können die verfügbaren Konstanten zur Versorgung der Parameter verwendet werden.

Die folgende Tabelle zeigt verfügbare Hardware-Datentypen und deren Bedeutung:

Datentyp	Basis-Datentyp	Beschreibung
REMOTE	ANY	Dient zur Angabe der Adresse einer remoten CPU. Dieser Datentyp wird z. B. für die Anweisungen "PUT" und "GET" verwendet.
HW_ANY	UINT	Identifikation einer beliebigen Hardware-Komponente, z. B. einer Baugruppe.
HW_DEVICE	HW_ANY	Identifikation eines DP-Slaves/PROFINET IO-Devices
HW_DPMASTER	HW_INTERFACE	Identifikation eines DP-Masters
HW_DPSLAVE	HW_DEVICE	Identifikation eines DP-Slaves
HW_IO	HW_ANY	Identifikationsnummer der CPU oder des Interfaces Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt.
HW_IOSYSTEM	HW_ANY	Identifikation eines PN/IO-Systems oder DP-Mastersystems
HW_SUBMODULE	HW_IO	Identifikation einer zentralen Hardware-Komponente
HW_MODULE	HW_IO	Identifikation eines Moduls
HW_INTERFACE	HW_SUBMODULE	Identifikation einer Interface-Komponente
HW_IEPORT	HW_SUBMODULE	Identifikation eines Ports (PN/IO)
HW_HSC	HW_SUBMODULE	Identifikation eines schnellen Zählers Dieser Datentyp wird z. B. für die Anweisungen "CTRL_HSC" und "CTRL_HSC_EXT" verwendet.
HW_PWM	HW_SUBMODULE	Identifikation einer Pulsbreitenmodulation Dieser Datentyp wird z. B. für die Anweisung "CTRL_PWM" verwendet.
HW_PTO	HW_SUBMODULE	Identifikation eines Impulsgebers Dieser Datentyp wird für Motion Control verwendet.
EVENT_ANY	AOM_IDENT	Dient zur Angabe eines beliebigen Ereignisses

Datentyp	Basis-Datentyp	Beschreibung
EVENT_ATT	EVENT_ANY	Dient zur Angabe eines Ereignisses, das einem OB dynamisch zugewiesen werden kann Dieser Datentyp wird z. B. für die Anweisungen "ATTACH" und "DETACH" verwendet.
EVENT_HWINT	EVENT_ATT	Dient zur Angabe eines Prozessalarm-Ereignisses
OB_ANY	INT	Dient zur Angabe eines beliebigen Organisationsbausteins.
OB_DELAY	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Verzögerungsalarms aufgerufen wird. Dieser Datentyp wird z. B. für die Anweisungen "SRT_DINT" und "CAN_DINT" verwendet.
OB_TOD	OB_ANY	Gibt die Nummer eines Uhrzeitalarm-Organisationsbausteins an. Dieser Datentyp wird z. B. für die Anweisungen "SET_TINT", "CAN_TINT", "ACT_TINT" und "QRY_TINT" verwendet.
OB_CYCLIC	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Weckalarms aufgerufen wird.
OB_ATT	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der einem Ereignis dynamisch zugewiesen werden kann. Dieser Datentyp wird z. B. für die Anweisungen "ATTACH" und "DETACH" verwendet.
OB_PCYCLE	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der einem Ereignis der Ereignisklasse "Zyklisches Programm" zugewiesen werden kann.
OB_HWINT	OB_ATT	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Prozessalarms aufgerufen wird.
OB_DIAG	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Diagnosealarms aufgerufen wird.
OB_TIMEERROR	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten von Zeitfehlern aufgerufen wird.
OB_STARTUP	OB_ANY	Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Anlauf-Ereignisses aufgerufen wird.
PORT	HW_SUBMODULE	Dient zur Angabe eines Kommunikationsports. Dieser Datentyp wird für die Punkt-zu-Punkt-Kommunikation verwendet.
RTM	UINT	Dient zur Angabe der Nummer eines Betriebsstundenzählers. Dieser Datentyp wird z. B. für die Anweisung "RTM" verwendet.
PIP	UINT	Dient zum Anlegen und Verschalten eines OB "Synchronous Cycle". Dieser Datentyp wird bei den SFCs 26, 27, 126 und 127 verwendet.
CONN_ANY	WORD	Dient zur Angabe einer beliebigen Verbindung.
CONN_PRG	CONN_ANY	Dient zur Angabe einer Verbindung für die Offene Kommunikation über UDP.

Datentyp	Basis-Datentyp	Beschreibung
CONN_OUC	CONN_ANY	Dient zur Angabe einer Verbindung für die offene Kommunikation über Industrial Ethernet (PROFINET).
CONN_R_ID	DWORD	Datentyp für den R_ID Parameter an den S7 Kommunikationsbausteinen.
DB_ANY	UINT	Identifikation (Name oder Nummer) eines DBs Der Datentyp "DB_ANY" hat im Abschnitt "Temp" die Länge 0.
DB_WWW	DB_ANY	Nummer des DBs, der die Web Applikation beschreibt (z. B. Anweisung "WWW") Der Datentyp "DB_WWW" hat im Abschnitt "Temp" die Länge 0.
DB_DYN	DB_ANY	Nummer eines durch das Anwenderprogramm generierten DBs
C_ALARM_SD	DWORD	Dient zur Erfassung der Meldungsnummer. Dieser Datentyp ist nur mit der CPU S7-410H nutzbar.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

3.15 Datentypkonvertierungen bei S7-1500

3.15.1 Übersicht über die Datentypkonvertierung

Einführung

Wenn Sie mehrere Operanden in einer Anweisung verknüpfen, müssen Sie die Verträglichkeit ihrer Datentypen beachten. Dies gilt auch bei Zuweisungen oder beim Versorgen von Bausteinparametern. Sind die Operanden ungleichen Datentyps, muss eine Konvertierung durchgeführt werden.

Es gibt zwei Möglichkeiten der Konvertierung:

- Implizite Konvertierung
  Die implizite Konvertierung wird von den Programmiersprachen KOP, FUP, SCL und GRAPH unterstützt. In der Programmiersprache AWL ist die implizite Konvertierung nicht möglich.
- Explizite Konvertierung
  Sie verwenden eine explizite Konvertierungsanweisung, bevor die eigentliche Anweisung ausgeführt wird.

Hinweis

Konvertieren von Bitfolgen in SCL

Alle Bitfolgen (BYTE, WORD, DWORD und LWORD) werden in Ausdrücken wie die entsprechenden Ganzzahlen ohne Vorzeichen (USINT, UINT, UDINT und ULINT) behandelt. Daher wird z. B. die implizite Konvertierung von DWORD nach REAL wie eine Konvertierung von UDINT nach REAL durchgeführt.

Hinweis

Konvertieren von REF()

Eine Datentypkonvertierung von REF() ist nicht möglich. Die übergebene Variable muss genau dem Zieldatentyp entsprechen.

Implizite Konvertierung

Eine implizite Konvertierung wird automatisch durchgeführt, wenn die Datentypen der Operanden kompatibel sind. Die Prüfung der Kompatibilität kann nach mehr oder weniger strengen Kriterien erfolgen:

- Mit IEC-Prüfung (Voreinstellung)
  Bei eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, BYTE, WORD, DWORD, DINT, INT, SINT, UDINT, UINT, USINT, TIME, LDT, DTL, DT, TOD, WCHAR und CHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp BYTE erwartet wird.
- Ohne IEC-Prüfung
  Bei nicht eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, LREAL, BYTE, WORD, DWORD, LWORD, SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, TIME, LTIME, S5TIME, LDT, DTL, TOD, LTOD, DATE, STRING, WSTRING, WCHAR und CHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp DWORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp WORD erwartet wird.
  - An Durchgangsparametern (InOut) muss die Bitlänge eines angegebenen Operanden mit der programmierten Bitlänge des jeweiligen Parameters übereinstimmen.

  Hinweis

  Implizite Konvertierung ohne IEC-Prüfung

  Der Programmiereditor kennzeichnet Operanden, die implizit konvertiert werden, mit einem grauen Rechteck. Ein dunkelgraues Rechteck signalisiert, dass eine implizite Konvertierung ohne Genauigkeitsverlust möglich ist, z. B., wenn Sie den Datentyp INT nach DINT konvertieren. Ein hellgraues Rechteck signalisiert, dass eine implizite Konvertierung möglich ist, aber während der Laufzeit können Fehler auftreten. Wenn Sie z. B. den Datentyp LINT nach DINT konvertieren und es zu einem Überlauf kommt, wird der Freigabeausgang ENO auf "0" gesetzt.

Weitere Informationen zur Einstellung der IEC-Prüfung und der impliziten Konvertierung finden Sie unter "Siehe auch".

Explizite Konvertierung

Wenn die Operanden nicht kompatibel sind und deshalb eine implizite Konvertierung nicht möglich ist, können Sie eine explizite Konvertierung vornehmen. Dazu können Sie entweder die Konvertierungsanweisungen in der Task Card "Anweisungen" nutzen oder Sie können die Konvertierung manuell in das Programm einfügen. Die Schreibweise der expliziten Konvertierungsfunktionen finden Sie unter "Siehe auch".

Ein möglicher Überlauf wird am Freigabeausgang ENO angezeigt. Ein Überlauf entsteht, wenn z. B. der Wert des Quelldatentyps größer als der Wert des Zieldatentyps ist.

Hinweis

Verschieben von Bitmustern

Wenn es sich bei der expliziten Konvertierung um das Verschieben eines Bitmusters handelt, dann wird der Freigabeausgang ENO nicht gesetzt.

Weitere Informationen zur expliziten Konvertierung finden Sie unter "Siehe auch".

Das folgende Bild zeigt ein Beispiel, in dem eine explizite Datentypkonvertierung durchgeführt werden muss:

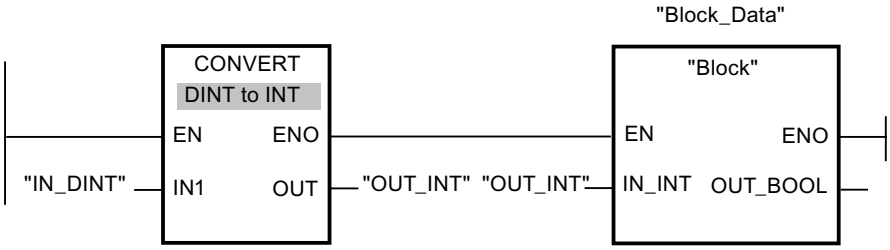

Der Funktionsbaustein "Block" erwartet am Eingangsparameter "IN_INT" eine Variable vom Datentyp INT. Der Wert der Variablen "IN_DINT" muss daher zunächst von DINT nach INT konvertiert werden. Befindet sich der Wert der Variablen "IN_DINT" innerhalb des zulässigen Wertebereichs des Datentyps INT, dann findet eine Konvertierung statt. Andernfalls wird ein Überlauf gemeldet. Doch auch im Falle eines Überlaufs findet trotzdem eine Konvertierung statt, aber die Werte werden abgeschnitten und der Freigabeausgang ENO auf "0" gesetzt.

Siehe auch

Implizite Konvertierungen (Seite 371)

Explizite Konvertierungen (Seite 423)

3.15.2 Implizite Konvertierungen

3.15.2.1 IEC-Prüfung aktivieren oder deaktivieren

Es wird geprüft, ob die Datentypen der verwendeten Operanden kompatibel sind. Diese Kompatibilitätsprüfung kann nach mehr oder weniger strengen Kriterien erfolgen. Bei aktivierter "IEC-Prüfung für Codebausteine" werden strengere Kriterien angewandt.

Sie können die IEC-Prüfung zentral für alle neuen Bausteine des Projekts oder für einzelne Bausteine einstellen.

IEC-Prüfung für neue Bausteine einstellen

Um die IEC-Prüfung für alle neuen Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Aktivieren oder deaktivieren Sie in der Gruppe "Voreinstellung für neue Bausteine" das Optionskästchen "IEC-Prüfung für Codebausteine".
   Die IEC-Prüfung wird für alle neuen Bausteine im Programm aktiviert oder deaktiviert.

IEC-Prüfung für einen Baustein einstellen

Um die IEC-Prüfung für einen Baustein einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "IEC-Prüfung".
   Die IEC-Prüfung wird für diesen Baustein aktiviert oder deaktiviert. Die Einstellung wird zusammen mit dem Projekt gespeichert.

3.15.2.2 Binärzahlen

Implizite Konvertierung von BOOL

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp BOOL ist nicht möglich.

Siehe auch

BOOL (Bit) (Seite 248)

Übersicht über die Datentypkonvertierung (Seite 369)

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Explizite Konvertierung von BOOL (Seite 423)

Bitfolgen

Implizite Konvertierung von BYTE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
BYTE	BOOL	-	-	Keine implizite Konvertierung
	WORD	X	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DWORD	X	X	
	LWORD	X	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

BYTE (Seite 249)

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

Explizite Konvertierung von BYTE (Seite 425)

Implizite Konvertierung von WORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert.
	DWORD	X	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LWORD	X	X	
	SINT	-	X	Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert.
	USINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LDT	-	-	Keine implizite Konvertierung
	DTL	-	-	
	DT	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	TOD	-	-	Keine implizite Konvertierung
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

WORD (Seite 250)
IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
Explizite Konvertierung von WORD (Seite 428)

Implizite Konvertierung von DWORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DWORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Die rechten Byte werden in den Zieldatentyp übertragen, die linken Byte werden ignoriert.
	WORD	-	X	
	LWORD	X	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	-	-	Keine implizite Konvertierung
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTIME	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTOD	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DWORD (Seite 251)
IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
Explizite Konvertierung von DWORD (Seite 431)

Implizite Konvertierung von LWORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LWORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LWORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Die rechten Byte werden in den Zieldatentyp übertragen, die linken Byte werden ignoriert.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	S5TIME	-	-	Keine implizite Konvertierung
	LDT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DTL	-	-	Keine implizite Konvertierung
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
LWORD (Seite 252)
Explizite Konvertierung von LWORD (Seite 435)

3.15.2.3 Ganzzahlen

Implizite Konvertierung von SINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp SINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
SINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	USINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertübertragung von z. B. SINT #-1 -> INT #-1, es wird nicht mit "0" aufgefüllt.)
	INT	X	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

SINT (8-Bit-Ganzzahlen) (Seite 254)
Explizite Konvertierung von SINT (Seite 439)

Implizite Konvertierung von USINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp USINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
USINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. USINT #10 -> DINT #10, oder USINT #128 -> SINT #-128)
	INT	X	X	
	UINT	X	X	
	DINT	X	X	
	UDINT	X	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
USINT (8-Bit-Ganzzahlen) (Seite 255)
Explizite Konvertierung von USINT (Seite 443)

Implizite Konvertierung von INT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
INT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. INT #-1 -> SINT #-1, oder INT #-32 767 -> UINT #32 769)
	USINT	-	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	TOD	-	-	Keine implizite Konvertierung
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

INT (16-Bit-Ganzzahlen) (Seite 256)
Explizite Konvertierung von INT (Seite 446)

Implizite Konvertierung von UINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
UINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. UINT #100 -> DINT #100, oder UINT #60 000 -> INT #-5536)
	USINT	-	X	
	INT	-	X	
	DINT	X	X	
	UDINT	X	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	TOD	-	-	Keine implizite Konvertierung
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
UINT (16-Bit-Ganzzahlen) (Seite 257)
Explizite Konvertierung von UINT (Seite 450)

Implizite Konvertierung von DINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTIME	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTOD	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
DINT (32-Bit-Ganzzahlen) (Seite 258)
Explizite Konvertierung von DINT (Seite 453)

Implizite Konvertierung von UDINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UDINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
UDINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTIME	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	LTOD	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
UDINT (32-Bit-Ganzzahlen) (Seite 259)
Explizite Konvertierung von UDINT (Seite 457)

Implizite Konvertierung von LINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. LINT #-1 -> SINT #-1, oder LINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	ULINT	-	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. LINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. LINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 9 007 199 254 740 992)
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Zeitdauer in Nanosekunden)
	S5TIME	-	-	Keine implizite Konvertierung
	LDT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Nanosekunden seit 1.1.1970)
	DTL	-	-	Keine implizite Konvertierung
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
LINT (64-Bit-Ganzzahlen) (Seite 260)
Explizite Konvertierung von LINT (Seite 461)

Implizite Konvertierung von ULINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp ULINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
ULINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. ULINT #-1 -> SINT #-1, oder ULINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. LINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. LINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 9 007 199 254 740 992)
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Zeitdauer in Nanosekunden)
	S5TIME	-	-	Keine implizite Konvertierung
	LDT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Nanosekunden seit 1.1.1970)
	DTL	-	-	Keine implizite Konvertierung
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
ULINT (64-Bit-Ganzzahlen) (Seite 261)
Explizite Konvertierung von ULINT (Seite 465)

3.15.2.4 Gleitpunktzahlen

Implizite Konvertierung von REAL

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp REAL:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
REAL	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	LWORD	-	-	Keine implizite Konvertierung
	SINT	-	X	Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. REAL #2.5 -> INT #2, oder negative Zahlen REAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	LREAL	X	X	Der Wert wird in den Zieldatentyp übertragen.
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
REAL (Seite 264)
Explizite Konvertierung von REAL (Seite 468)

Implizite Konvertierung von LREAL

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LREAL:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LREAL	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	SINT	-	X	Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. LREAL #2.5 -> INT #2, oder negative Zahlen LREAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt LREAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	X	Der Wert wird in den Zieldatentyp übertragen.
	TIME	-	-	Keine implizite Konvertierung
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

LREAL (Seite 265)
Explizite Konvertierung von LREAL (Seite 471)

3.15.2.5 Zeiten

Implizite Konvertierung von S5TIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp S5TIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
S5TIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	DWORD	-	-	Keine implizite Konvertierung
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	

x: Konvertierung möglich
-: Konvertierung nicht möglich

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
S5TIME (Zeitdauer) (Seite 268)
Explizite Konvertierung von S5TIME (Seite 474)

Implizite Konvertierung von TIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	LWORD	-	-	Keine implizite Konvertierung
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	UDINT	-	X	
	LINT	-	-	Keine implizite Konvertierung
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	S5TIME	-	-	
	LTIME	X	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Nanosekunden. (1 ms = 1 000 000 ns)
	LDT	-	-	Keine implizite Konvertierung
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	Wenn der Quellwert zwischen 0 s und 84599.999 s liegt, dann wird das Bitmuster des Quellwerts ohne Änderung in den Zieldatentyp übertragen. (Darstellung in Nanosekunden) Ansonsten wird der Zielwert nicht verändert. Das Ergebnis der Konvertierung zeigt die Zeit, die seit Mitternacht vergangen ist.
	LTOD	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
TIME (IEC-Zeit) (Seite 269)
Explizite Konvertierung von TIME (Seite 476)

Implizite Konvertierung von LTIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LTIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LTIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Nanosekunden.
	SINT	-	-	Keine implizite Konvertierung
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Nanosekunden.
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	S5TIME	-	-	
	TIME	-	X	Liegt der Quellwert außerhalb des Wertebereichs des Zieldatentyps, dann wird der Zielwert nicht verändert. (Aus 0.123456789 s wird 0.123 s)
	LDT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Nanosekunden seit 1.1.1970.
	DTL	-	-	Keine implizite Konvertierung
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Nanosekunden seit 1.1.1970.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
LTIME (IEC-Zeit) (Seite 270)
Explizite Konvertierung von LTIME (Seite 479)

3.15.2.6 Datum und Uhrzeit

Implizite Konvertierung von DT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	X	X	Der Quellwert wird ohne Änderung werterhaltend in den Zieldatentyp übertragen. (Der 24.12.2012 14:30 bleibt der 24.12.2012 14:30)
	DTL	X	X	
	DATE	-	-	Keine implizite Konvertierung
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
DT (DATE_AND_TIME) (Seite 272)
Explizite Konvertierung von DT (Seite 482)

Implizite Konvertierung von LDT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LDT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LDT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Nanosekunden seit 1.1.1970)
	SINT	-	-	Keine implizite Konvertierung
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Nanosekunden seit 1.1.1970)
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. (Nanosekunden seit 1.1.1970)
	S5TIME	-	-	Keine implizite Konvertierung
	DT	-	X	Das Bitmuster des Quellwerts wird mit einem Genauigkeitsverlust in den Zieldatentyp konvertiert. (Aus 24.12.2012 12:34:56.123456789 wird 24.12.2012 12:34:56.123)
	DTL	X	X	Der Quellwert wird ohne Änderung werterhaltend in den Zieldatentyp übertragen. (Der 24.12.2012 14:30 bleibt der 24.12.2012 14:30)
	DATE	-	-	Keine implizite Konvertierung
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
Explizite Konvertierung von LDT (Seite 485)
LDT (DATE_AND_LTIME) (Seite 274)

Implizite Konvertierung von DTL

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DTL:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DTL	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	X	X	Der Quellwert wird ohne Änderung werterhaltend in den Zieldatentyp übertragen. (Der 24.12.2012 14:30 bleibt der 24.12.2012 14:30)
	DT	-	X	Das Bitmuster des Quellwerts wird mit einem Genauigkeitsverlust in den Zieldatentyp konvertiert. (Aus 24.12.2012 12:34:56.123456789 wird 24.12.2012 12:34:56.123)
	DATE	-	-	Keine implizite Konvertierung
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
DTL (Seite 274)
Explizite Konvertierung von DTL (Seite 488)

Implizite Konvertierung von DATE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DATE	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	DWORD	-	-	Keine implizite Konvertierung
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	UINT	-	X	
	DINT	-	-	Keine implizite Konvertierung
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

DATE (Seite 271)
Explizite Konvertierung von DATE (Seite 490)

Implizite Konvertierung von TOD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TOD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	LWORD	-	-	Keine implizite Konvertierung
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	UDINT	-	X	
	LINT	-	-	Keine implizite Konvertierung
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	LTIME	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	DATE	-	-	
	LTOD	X	X	Der Quellwert wird ohne Änderung werterhaltend in den Zieldatentyp übertragen. (Aus 12:34:56.123 wird 12:34:56.123000000)
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
TOD (TIME_OF_DAY) (Seite 271)
Explizite Konvertierung von TOD (Seite 492)

Implizite Konvertierung von LTOD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LTOD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LTOD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Nanosekunden seit Tagesbeginn (0:00 Uhr).
	SINT	-	-	Keine implizite Konvertierung
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Nanosekunden seit Tagesbeginn (0:00 Uhr).
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Nanosekunden seit Tagesbeginn (0:00 Uhr).
	S5TIME	-	-	Keine implizite Konvertierung
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	DATE	-	-	
	TOD	-	X	Der Quellwert wird ohne Änderung werterhaltend mit einer Rundung in den Zieldatentyp übertragen. (Aus 12:34:56.123456789 wird 12:34:56.123)
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
Explizite Konvertierung von LTOD (Seite 494)
LTOD (LTIME_OF_DAY) (Seite 272)

3.15.2.7 Zeichenfolgen

Implizite Konvertierung von CHAR

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
CHAR	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden von links mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	WCHAR	-	-	
	STRING	X	X	Der STRING wird auf die Länge 1 gekürzt und enthält das Zeichen.
	WSTRING	-	-	Keine implizite Konvertierung
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
CHAR (Seite 276)
Explizite Konvertierung von CHAR (Seite 497)

Implizite Konvertierung von WCHAR

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WCHAR:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WCHAR	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bits werden von links mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	-	
	STRING	-	-	
	WSTRING	X	X	Der WSTRING wird auf die Länge 1 gekürzt und enthält das Zeichen.
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

WCHAR (Seite 276)

Explizite Konvertierung von WCHAR (Seite 499)

Implizite Konvertierung von STRING

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp STRING:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
STRING	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	X	Das erste Zeichen des STRING wird zurückgegeben, wenn der STRING ein oder mehr Zeichen enthält. Sonst wird das Zeichen mit der Codierung $00 ausgegeben.
	WCHAR	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)
Übersicht über die Datentypkonvertierung (Seite 369)
STRING (Seite 277)
Explizite Konvertierung von STRING (Seite 501)

Implizite Konvertierung von WSTRING

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WSTRING:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WSTRING	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	-	
	WCHAR	-	X	Das erste Zeichen des WSTRING wird zurückgegeben, wenn der WSTRING ein oder mehr Zeichen enthält. Sonst wird das Zeichen mit der Codierung $0000 ausgegeben.
	STRING	-	-	Keine implizite Konvertierung
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 371)

Übersicht über die Datentypkonvertierung (Seite 369)

WSTRING (Seite 280)
Explizite Konvertierung von WSTRING (Seite 504)

3.15.3 Explizite Konvertierungen

3.15.3.1 Binärzahlen

Explizite Konvertierung von BOOL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BOOL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BOOL	BYTE	X	Im Zieldatentyp wird nur das LSB (Least Significant Bit) gesetzt. Der Freigabeausgang ENO ist immer "1".	BOOL_TO_BYTE
	WORD	X		BOOL_TO_WORD
	DWORD	X		BOOL_TO_DWORD
	LWORD	X		BOOL_TO_LWORD
	SINT	X		BOOL_TO_SINT
	USINT	X		BOOL_TO_USINT
	INT	X		BOOL_TO_INT
	UINT	X		BOOL_TO_UINT
	DINT	X		BOOL_TO_DINT
	UDINT	X		BOOL_TO_UDINT
	LINT	X		BOOL_TO_LINT
	ULINT	X		BOOL_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	-		-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Übersicht über die Datentypkonvertierung (Seite 369)
Implizite Konvertierung von BOOL (Seite 372)
BOOL (Bit) (Seite 248)

Bitfolgen

Explizite Konvertierung von BYTE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BYTE[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	BYTE_TO_BOOL
	WORD[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_WORD
	DWORD[1)]	X		BYTE_TO_DWORD
	LWORD[1)]	X		BYTE_TO_LWORD
	SINT	X		BYTE_TO_SINT
	USINT	X		BYTE_TO_USINT
	INT	X		BYTE_TO_INT
	UINT	X		BYTE_TO_UINT
	DINT	X		BYTE_TO_DINT
	UDINT	X		BYTE_TO_UDINT
	LINT	X		BYTE_TO_LINT
	ULINT	X		BYTE_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_TIME
	LTIME	X		BYTE_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_TOD
	LTOD	X		BYTE_TO_LTOD
	DATE	X		BYTE_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_CHAR
	WCHAR	X		BYTE_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT, DWORD als UDINT und LWORD als ULINT interpretiert.

Siehe auch

Übersicht über die Datentypkonvertierung (Seite 369)

Implizite Konvertierung von BYTE (Seite 373)

BYTE (Seite 249)

Explizite Konvertierung von WORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WORD[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	WORD_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_BYTE
	DWORD[1)]	X		WORD_TO_DWORD
	LWORD[1)]	X		WORD_TO_LWORD

	SINT	X	ENO = TRUE
			#sint1 := WORD_TO_SINT(16#FFFF); // -1 bis
			#sint1 := WORD_TO_SINT(16#FF80); // -128
			#sint1 := WORD_TO_SINT(16#0); // 0 bis
			#sint1 := WORD_TO_SINT(16#007F); // 127
			
			ENO = FALSE
			#sint1 := WORD_TO_SINT(16#FF7F); // -129 bis
			#sint1 := WORD_TO_SINT(16#8000); // -32768
			#sint1 := WORD_TO_SINT(16#0080); // 128 bis
			#sint1 := WORD_TO_SINT(16#7FFF); // 32767
				WORD_TO_SINT

	USINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_USINT
	INT	X		WORD_TO_INT
	UINT	X		WORD_TO_UINT
	DINT	X		WORD_TO_DINT
	UDINT	X		WORD_TO_UDINT
	LINT	X		WORD_TO_LINT
	ULINT	X		WORD_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_TIME
	LTIME	X		WORD_TO_LTIME
	S5TIME	X		WORD_TO_S5TIME
	LDT	X		WORD_TO_LDT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_TOD
	LTOD	X		WORD_TO_LTOD
	DATE	X		WORD_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_CHAR
	WCHAR	X		WORD_TO_WCHAR
WORD_ BCD16	INT	X	Der zu konvertierende Wert hat den Datentyp WORD und wird als BCD-codierter Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ INT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	WORD_BCD16_TO_INT
BCD16	INT	X		BCD16_TO_INT

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT, DWORD als UDINT und LWORD als ULINT interpretiert.

Siehe auch

Implizite Konvertierung von WORD (Seite 374)

Übersicht über die Datentypkonvertierung (Seite 369)

WORD (Seite 250)

Explizite Konvertierung von DWORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DWORD[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	DWORD_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_BYTE
	WORD[1)]	X		DWORD_TO_WORD
	LWORD[1)]	X		DWORD_TO_LWORD

	SINT	X	ENO = TRUE
			#sint1 := DWORD_TO_SINT(16#FFFF_FFFF); // -1 bis
			#sint1 := DWORD_TO_SINT(16#FFFF_FF80); // -128
			#sint1 := DWORD_TO_SINT(16#0); // 0 bis
			#sint1 := DWORD_TO_SINT(16#0000_007F); // 127
			
			ENO = FALSE
			#sint1 := DWORD_TO_SINT(16#FFFF_FF7F); // -129
			#sint1 := DWORD_TO_SINT(16#8000_0000); // -2147483648
			#sint1 := DWORD_TO_SINT(16#0000_0080); // 128 bis
			#sint1 := DWORD_TO_SINT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_SINT

	USINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_USINT

	INT	X	ENO = TRUE
			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 bis
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
				DWORD_TO_INT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung

			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 bis
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647	

	UINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand von ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand von ENO = 0.	DWORD_TO_UINT
	DINT	X		DWORD_TO_DINT
	UDINT	X		DWORD_TO_UDINT
	LINT	X		DWORD_TO_LINT
	ULINT	X		DWORD_TO_ULINT
	REAL	X		DWORD_TO_REAL
	LREAL	-	Keine explizite Konvertierung	-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_TIME
	LTIME	X		DWORD_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_TOD
	LTOD	X		DWORD_TO_LTOD
	DATE	X		DWORD_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_CHAR
	WCHAR	X		DWORD_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DWORD_BCD32	DINT	X	Der zu konvertierende Wert hat den Datentyp DWORD und wird als BCD-codierter Wert zwischen -9999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ DINT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	DWORD_BCD32_TO_DINT
BCD32	DINT	X		BCD32_TO_DINT
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD, LWORD als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT, DWORD als UDINT und LWORD als ULINT interpretiert.				

Siehe auch

Implizite Konvertierung von DWORD (Seite 376)

Übersicht über die Datentypkonvertierung (Seite 369)

DWORD (Seite 251)

Explizite Konvertierung von LWORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LWORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LWORD[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	LWORD_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_BYTE
	WORD[1)]	X		LWORD_TO_WORD
	DWORD[1)]	X		LWORD_TO_DWORD

	SINT	X	ENO = TRUE
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FFFF); // -1 bis
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FF80); // -128
			#sint1 := LWORD_TO_SINT(16#0); // 0 bis
			#sint1 := LWORD_TO_SINT(16#0000_0000_0000_007F); // 127
			
			ENO = FALSE
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FF7F); // -129
			#sint1 := LWORD_TO_SINT(16#8000_0000_0000_0000); // -9223372036854775808
			#sint1 := LWORD_TO_SINT(16#0000_0000_0000_0080); // 128
			#sint1 := LWORD_TO_SINT(16#7FFF_FFFF_FFFF_FFFF); // 9223372036854775807
				LWORD_TO_SINT

	USINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_USINT

	INT	X	ENO = TRUE
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_FFFF); // -1 bis
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_8000); // -32768
			#int1 := LWORD_TO_INT(16#0); // 0 bis
			#int1 := LWORD_TO_INT(16#0000_0000_0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_7FFF); // -32769
			#int1 := LWORD_TO_INT(16#8000_0000_0000_0000); // -2147483648
				LWORD_TO_INT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			#int1 := LWORD_TO_INT(16#0000_0000_0000_8000); // 32768 bis #int1 := LWORD_TO_INT(16#7FFF_FFFF_FFFF_FFFF); // 2147483647	
	UINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_UINT

	DINT	X	ENO = TRUE
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_FFFF_FFFF); // -1 bis
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_8000_0000); // -2147483648
			#dint1 := LWORD_TO_DINT(16#0); // 0 bis
			#dint1 := LWORD_TO_DINT(16#0000_0000_7FFF_FFFF); // 2147483647
			
			ENO = FALSE
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_7FFF_FFFF); // -2147483649 bis
			#dint1 := LWORD_TO_DINT(16#8000_0000_0000_0000); // -9223372036854775808
			#dint1 := LWORD_TO_DINT(16#0000_0000_8000_0000); // 2147483648 bis
			#dint1 := LWORD_TO_DINT(16#7FFF_FFFF_FFFF_FFFF); // 9223372036854775807
				LWORD_TO_DINT

	UDINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_UDINT
	LINT	X		LWORD_TO_LINT
	ULINT	X		LWORD_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand von ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand von ENO = 0.	LWORD_TO_LREAL
	TIME	X		LWORD_TO_TIME
	LTIME	X		LWORD_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_TOD
	LTOD	X		LWORD_TO_LTOD
	DATE	X		LWORD_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LWORD_TO_CHAR
	WCHAR	X		LWORD_TO_WCHAR
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD, LWORD als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT, DWORD als UDINT und LWORD als ULINT interpretiert.				

Siehe auch

Implizite Konvertierung von LWORD (Seite 378)

Übersicht über die Datentypkonvertierung (Seite 369)

LWORD (Seite 252)

3.15.3.2 Ganzzahlen

Explizite Konvertierung von SINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp SINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
SINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	SINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_BYTE
	WORD[1)]	X		SINT_TO_WORD
	DWORD[1)]	X		SINT_TO_DWORD
	LWORD[1)]	X		SINT_TO_LWORD
	USINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_USINT
	INT	X		SINT_TO_INT
	UINT	X		SINT_TO_UINT
	DINT	X		SINT_TO_DINT
	UDINT	X		SINT_TO_UDINT
	LINT	X		SINT_TO_LINT
	ULINT	X		SINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	SINT_TO_REAL, NORM_X
	LREAL	X		SINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	SINT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	SINT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	SINT_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)	SINT_TO_TOD
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Nanosekunden seit 0:0)	SINT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)	SINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP / AWL: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		SINT_TO_WSTRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_CHAR
	WCHAR[1)]	X		SINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von SINT (Seite 380)

Übersicht über die Datentypkonvertierung (Seite 369)

SINT (8-Bit-Ganzzahlen) (Seite 254)

Explizite Konvertierung von USINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp USINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
USINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	USINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	USINT_TO_BYTE
	WORD[1)]	X		USINT_TO_WORD
	DWORD[1)]	X		USINT_TO_DWORD
	LWORD[1)]	X		USINT_TO_LWORD
	SINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichen wechselt, wird der Freigabeausgang ENO auf "0" gesetzt.	USINT_TO_SINT
	INT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen.	USINT_TO_INT
	UINT	X		USINT_TO_UINT
	DINT	X		USINT_TO_DINT
	UDINT	X		USINT_TO_UDINT
	LINT	X		USINT_TO_LINT
	ULINT	X		USINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	USINT_TO_REAL, NORM_X
	LREAL	X		USINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	USINT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	USINT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	USINT_TO_LDT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Millisekunden seit 0:0)	USINT_TO_TOD
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Nanosekunden seit 0:0)	USINT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Tagen seit 1990-1-1)	USINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	USINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		USINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen.	USINT_TO_CHAR
	WCHAR[1)]	X		USINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von USINT (Seite 381)

Übersicht über die Datentypkonvertierung (Seite 369)

USINT (8-Bit-Ganzzahlen) (Seite 255)

Explizite Konvertierung von INT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
INT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	INT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_BYTE
	WORD[1)]	X		INT_TO_WORD
	DWORD[1)]	X		INT_TO_DWORD
	LWORD[1)]	X		INT_TO_LWORD
	SINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_SINT
	USINT	X		INT_TO_USINT
	UINT	X		INT_TO_UINT
	DINT	X		INT_TO_DINT
	UDINT	X		INT_TO_UDINT
	LINT	X		INT_TO_LINT
	ULINT	X		INT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	INT_TO_REAL, NORM_X
	LREAL	X		INT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	INT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	INT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	INT_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp	INT_TO_TOD

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Nanosekunden seit 0:0; Prüfung auf 24h Grenze)	INT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	INT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP / AWL: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		INT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1"	INT_TO_CHAR
	WCHAR[1)]	X		INT_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			(16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	
	BCD16	X	Der zu konvertierende Wert hat den Typ INT und wird als Ganzzahl mit einem Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ WORD vor. Es findet eine echte Wandlung statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	INT_TO_BCD16
	BCD16_WORD	X		INT_TO_BCD16_WORD

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von INT (Seite 383)

Übersicht über die Datentypkonvertierung (Seite 369)

INT (16-Bit-Ganzzahlen) (Seite 256)

Explizite Konvertierung von UINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
UINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	UINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn dabei Bits verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_BYTE
	WORD[1)]	X		UINT_TO_WORD
	DWORD[1)]	X		UINT_TO_DWORD
	LWORD[1)]	X		UINT_TO_LWORD
	SINT	X		UINT_TO_SINT
	USINT	X		UINT_TO_USINT
	INT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_INT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	UINT_TO_DINT
	UDINT	X		UINT_TO_UDINT
	LINT	X		UINT_TO_LINT
	ULINT	X		UINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	UINT_TO_REAL, NORM_X
	LREAL	X		UINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	UINT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	UINT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	UINT_TO_LDT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	UINT_TO_TOD
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Nanosekunden seit 0:0; Prüfung auf 24h Grenze)	UINT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	UINT_TO_DATE, T_CONV
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		UINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_CHAR
	WCHAR[1)]	X		UINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von UINT (Seite 384)

Übersicht über die Datentypkonvertierung (Seite 369)

UINT (16-Bit-Ganzzahlen) (Seite 257)

Explizite Konvertierung von DINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	DINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_BYTE
	WORD[1)]	X		DINT_TO_WORD
	DWORD[1)]	X		DINT_TO_DWORD
	LWORD[1)]	X		DINT_TO_LWORD
	SINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_SINT
	USINT	X		DINT_TO_USINT
	INT	X		DINT_TO_INT
	UINT	X		DINT_TO_UINT
	UDINT	X		DINT_TO_UDINT
	LINT	X		DINT_TO_LINT
	ULINT	X		DINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	DINT_TO_REAL, NORM_X
	LREAL	X		DINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	DINT_TO_TIME, T_CONV
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	DINT_TO_LTIME, T_CONV
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	DINT_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)	DINT_TO_TOD
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Nanosekunden seit 0:0)	DINT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)	DINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP / AWL: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		DINT_TO_WSTRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	CHAR[1]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_CHAR
	WCHAR[1]	X		DINT_TO_WCHAR
	BCD32	X	Der zu konvertierende Wert hat den Typ DINT und wird als Ganzzahl mit einem Wert zwischen –999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ DWORD vor. Bei einem Überlauf wird der Freigabeausgang auf "0" gesetzt. Es findet eine echte Wandlung statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	DINT_TO_BCD32
	BCD32_DWORD	X		DINT_TO_BCD32_DWORD

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von DINT (Seite 386)

Übersicht über die Datentypkonvertierung (Seite 369)

DINT (32-Bit-Ganzzahlen) (Seite 258)

Explizite Konvertierung von UDINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UDINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
UDINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	UDINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn dabei Bits verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_BYTE
	WORD[1)]	X		UDINT_TO_WORD
	DWORD[1)]	X		UDINT_TO_DWORD
	LWORD[1)]	X		UDINT_TO_LWORD
	SINT	X		UDINT_TO_SINT
	USINT	X		UDINT_TO_USINT
	INT	X		UDINT_TO_INT
	UINT	X		UDINT_TO_UINT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_DINT
	LINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	UDINT_TO_LINT
	ULINT	X		UDINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	UDINT_TO_REAL, NORM_X
	LREAL	X		UDINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	UDINT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	UDINT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	UDINT_TO_LDT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	UDINT_TO_TOD, T_CONV
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Nanosekunden seit 0:0; Prüfung auf 24h Grenze)	UDNT_TO_LTOD, T_CONV
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	UDINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		UDINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_CHAR
	WCHAR[1)]	X		UDINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von UDINT (Seite 388)
Übersicht über die Datentypkonvertierung (Seite 369)
UDINT (32-Bit-Ganzzahlen) (Seite 259)

Explizite Konvertierung von LINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	LINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	LINT_TO_BYTE
	WORD[1)]	X		LINT_TO_WORD
	DWORD[1)]	X		LINT_TO_DWORD
	LWORD[1)]	X		LINT_TO_LWORD
	SINT	X		LINT_TO_SINT
	USINT	X		LINT_TO_USINT
	INT	X		LINT_TO_INT
	UINT	X		LINT_TO_UINT
	DINT	X		LINT_TO_DINT
	UDINT	X		LINT_TO_UDINT
	ULINT	X		LINT_TO_ULINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	LINT_TO_REAL, NORM_X
	LREAL	X		LINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	LINT_TO_TIME, T_CONV
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	LINT_TO_LTIME, T_CONV
	S5TIME	-	Keine explizite Konvertierung	-
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	LINT_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1"	LINT_TO_TOD

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			(16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)	
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Nanosekunden seit 0:0)	LINT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)	LINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP / AWL: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	LINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		LINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1"	LINT_TO_CHAR
	WCHAR[1)]	X		LINT_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			(16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

Implizite Konvertierung von LINT (Seite 390)

Übersicht über die Datentypkonvertierung (Seite 369)

LINT (64-Bit-Ganzzahlen) (Seite 260)

Explizite Konvertierung von ULINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp ULINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
ULINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	ULINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	ULINT_TO_BYTE
	WORD[1)]	X		ULINT_TO_WORD
	DWORD[1)]	X		ULINT_TO_DWORD
	LWORD[1)]	X		ULINT_TO_LWORD
	SINT	X		ULINT_TO_SINT
	USINT	X		ULINT_TO_USINT
	INT	X		ULINT_TO_INT
	UINT	X		ULINT_TO_UINT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit überschrieben wird, wird der Freigabeausgang ENO auf "0" gesetzt.	ULINT_TO_DINT
	UDINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	ULINT_TO_UDINT
	LINT	X		ULINT_TO_LINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	ULINT_TO_REAL, NORM_X
	LREAL	X		ULINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	ULINT_TO_TIME
	LTIME	X	Der Wert wird in den Zieldatentyp übertragen und als Nanosekunden interpretiert.	ULINT_TO_LTIME
	S5TIME	-	Keine explizite Konvertierung	-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	LDT	X	Das Ergebnis wird in Nanosekunden seit 1970-1-1-0:0:0.0 geliefert.	ULINT_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Millisekunden seit 0:0)	ULINT_TO_TOD, T_CONV
	LTOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Nanosekunden seit 0:0)	ULINT_TO_LTOD, T_CONV
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Interpretation in Tagen seit 1990-1-1)	ULINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	ULINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		ULINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen.	ULINT_TO_CHAR
	WCHAR[1)]	X		ULINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von ULINT (Seite 392)

Übersicht über die Datentypkonvertierung (Seite 369)

ULINT (64-Bit-Ganzzahlen) (Seite 261)

3.15.3.3 Gleitpunktzahlen

Explizite Konvertierung von REAL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp REAL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
REAL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	REAL_TO_DWORD
	LWORD	-	Keine explizite Konvertierung	-
	SINT	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich des Zieldatentyps überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	REAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		REAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		REAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		REAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		REAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		REAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LINT	X		REAL_TO_LINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	ULINT	X		REAL_TO_ULINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LREAL	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab, z. B. TRUNC(2.5) = 2.0; CEIL(2.5) = 3.0	REAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Keine explizite Konvertierung	-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 14 Zeichen.	REAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		REAL_TO_WSTRING
	CHAR	-	Keine explizite Konvertierung	-
	WCHAR	-		

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Implizite Konvertierung von REAL (Seite 394)
Übersicht über die Datentypkonvertierung (Seite 369)
REAL (Seite 264)

Explizite Konvertierung von LREAL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LREAL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LREAL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LREAL_TO_LWORD
	SINT	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich des Zieldatentyps überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	LREAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		LREAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		LREAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		LREAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		LREAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		LREAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LINT	X		LREAL_TO_LINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	ULINT	X		LREAL_TO_ULINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	REAL	X	Der Wert wird in den Zieldatentyp konvertiert. Wenn bei der Konvertierung der erlaubte Wertebereich überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Ein Genauigkeitsverlust wird toleriert.	LREAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Keine explizite Konvertierung	-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0"	LREAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		LREAL_TO_WSTRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			gesetzt. Die min. Länge der Zeichenkette beträgt 21 Zeichen.	
	CHAR	-	Keine explizite Konvertierung	-
	WCHAR	-		-
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Implizite Konvertierung von LREAL (Seite 396)

Übersicht über die Datentypkonvertierung (Seite 369)

LREAL (Seite 265)

3.15.3.4 Zeiten

Explizite Konvertierung von S5TIME

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp S5TIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
S5TIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	S5TIME_TO_WORD
	DWORD	-	Keine explizite Konvertierung	-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	TIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. Aus s5t#10ms wird T#10ms)	S5TIME_TO_TIME
	LTIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. Aus s5t#10ms wird LTIME#10ms)	S5TIME_TO_LTIME
	LDT	-	Keine explizite Konvertierung	-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich [1] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

Implizite Konvertierung von S5TIME (Seite 398)

Übersicht über die Datentypkonvertierung (Seite 369)

S5TIME (Zeitdauer) (Seite 268)

Explizite Konvertierung von TIME

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	TIME_TO_BYTE
	WORD[1)]	X		TIME_TO_WORD
	DWORD[1)]	X		TIME_TO_DWORD
	LWORD[1)]	X		TIME_TO_LWORD
	SINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen.	TIME_TO_SINT
	USINT	X		TIME_TO_USINT
	INT	X		TIME_TO_INT
	UINT	X		TIME_TO_UINT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.	TIME_TO_DINT, T_CONV
	UDINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen. Ein Vorzeichenwechsel führt dazu, dass der Freigabeausgang ENO "0" ist.	TIME_TO_UDINT
	LINT	X		TIME_TO_LINT
	ULINT	X		TIME_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus T#10ms wird s5t#10ms)	TIME_TO_TIME
	LTIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus T#10ms wird LTIME#10ms)	TIME_TO_LTIME
	LDT	-	Keine explizite Konvertierung	-
	DT	-		-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn der Quellwert außerhalb des Wertebereichs von TOD liegt, dann wird der Zieldatentyp nicht verändert.	TIME_TO_TOD
	LTOD	-	Keine explizite Konvertierung	-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

Implizite Konvertierung von TIME (Seite 400)

Übersicht über die Datentypkonvertierung (Seite 369)

TIME (IEC-Zeit) (Seite 269)

Explizite Konvertierung von LTIME

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LTIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LTIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LTIME_TO_BYTE
	WORD[1)]	X		LTIME_TO_WORD
	DWORD[1)]	X		LTIME_TO_DWORD
	LWORD[1)]	X		LTIME_TO_LWORD
	SINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Nanosekunden interpretiert, in den Zieldatentyp übertragen. Ein Vorzeichenwechsel führt dazu, dass der Freigabeausgang ENO "0" ist.	LTIME_TO_SINT
	USINT	X		LTIME_TO_USINT
	INT	X		LTIME_TO_INT
	UINT	X		LTIME_TO_UINT
	DINT	X		LTIME_TO_DINT
	UDINT	X		LTIME_TO_UDINT
	LINT	X		LTIME_TO_LINT, T_CONV
	ULINT	X		LTIME_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTIME#10ms wird s5t#10ms)	LTIME_TO_S5TIME
	TIME	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTIMET#10ms wird T#10ms)	LTIME_TO_TIME
	LDT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTIMET#10ms wird LDT#1970-1-1-0:0:0.010000000)	LTIME_TO_LDT
	DT	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	-		-
	LTOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn der Quellwert außerhalb des Wertebereichs von LTOD liegt, dann wird der Zieldatentyp nicht verändert.	LTIME_TO_LTOD
	DATE	-	Keine explizite Konvertierung	-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von LTIME (Seite 402)
Übersicht über die Datentypkonvertierung (Seite 369)
LTIME (IEC-Zeit) (Seite 270)

3.15.3.5 Datum und Uhrzeit

Explizite Konvertierung von DT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DT	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	LDT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DT#1990-1-2-0:0:1.0 wird LDT#1990-1-2-0:0:1.0)	DT_TO_LDT
	DTL	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DT#1990-1-2-0:0:1.0 wird DTL#1990-1-2-0:0:1.0)	DT_TO_DTL
	TOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DT#1990-1-2-0:0:1.0 wird TOD#1990-1-2-0:0:1.0)	DT_TO_TOD
	LTOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DT#1990-1-2-0:0:1.0 wird LTOD#1990-1-2-0:0:1.0)	DT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DT_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Implizite Konvertierung von DT (Seite 404)
Übersicht über die Datentypkonvertierung (Seite 369)
DT (DATE_AND_TIME) (Seite 272)

Explizite Konvertierung von LDT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LDT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LDT	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LDT_TO_BYTE
	WORD[1)]	X		LDT_TO_WORD
	DWORD[1)]	X		LDT_TO_DWORD
	LWORD[1)]	X		LDT_TO_LWORD
	SINT	X		LDT_TO_SINT
	USINT	X		LDT_TO_USINT
	INT	X		LDT_TO_INT
	UINT	X		LDT_TO_UINT
	DINT	X		LDT_TO_DINT
	UDINT	X		LDT_TO_UDINT
	LINT	X		LDT_TO_LINT
	ULINT	X		LDT_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LDT_TO_LTIME
	DT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LDT#1990-1-2-0:0:1.0 wird DT#1990-1-2-0:0:1.0)	LDT_TO_DT
	DTL	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LDT#1990-1-2-0:0:1.0 wird DTL#1990-1-2-0:0:1.0)	LDT_TO_DTL
	TOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LDT#1990-1-2-0:0:1.0 wird TOD#1990-1-2-0:0:1.0)	LDT_TO_TOD
	LTOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LDT#1990-1-2-0:0:1.0 wird LTOD#1990-1-2-0:0:1.0)	LDT_TO_LTOD
	DATE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LDT_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

Implizite Konvertierung von LDT (Seite 406)

Übersicht über die Datentypkonvertierung (Seite 369)

LDT (DATE_AND_LTIME) (Seite 274)

Explizite Konvertierung von DTL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DTL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DTL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	LDT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DTL#1990-1-2-0:0:1.0 wird LDT#1990-1-2-0:0:1.0)	DTL_TO_LDT
	DT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus DTL#1990-1-2-0:0:1.0 wird DT#1990-1-2-0:0:1.0)	DTL_TO_DT
	TOD	X	Bei der Konvertierung wird die Tageszeit aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben.	DTL_TO_TOD, T_CONV
	LTOD	X		DTL_TO_LTOD
	DATE	X	Bei der Konvertierung wird das Datum aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	DTL_TO_DATE, T_CONV
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Implizite Konvertierung von DTL (Seite 408)
Übersicht über die Datentypkonvertierung (Seite 369)
DTL (Seite 274)

Explizite Konvertierung von DATE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DATE	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DATE_TO_BYTE
	WORD[1)]	X		DATE_TO_WORD
	DWORD[1)]	X		DATE_TO_DWORD
	LWORD[1)]	X		DATE_TO_LWORD
	SINT	X	Als Ergebnis wird die Anzahl der Tage seit dem 1.1.1990 geliefert.	DATE_TO_SINT
	USINT	X		DATE_TO_USINT
	INT	X		DATE_TO_INT
	UINT	X		DATE_TO_UINT
	DINT	X		DATE_TO_DINT
	UDINT	X		DATE_TO_UDINT
	LINT	X		DATE_TO_LINT
	ULINT	X		DATE_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	X	Die Konvertierung setzt das Datum im Zieltyp.	DATE_TO_DT
	LDT	X		DATE_TO_LDT
	DTL [2)]	X		DATE_TO_DTL
	TOD	-	Keine explizite Konvertierung	-
	LTOD	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

[2)] Der Datentyp DTL wird wie folgt initialisiert: 1970-1-1-0:0:0. Eine Konvertierung von DATE_TO_DTL setzt nur den entsprechenden Teil des DTL. Der Rest des DTLs bleibt, wie er initialisiert wurde. Daraus ergeben sich folgende Unterschiede bei den Anweisungen "T_COMBINE" und "T_CONV":

- T_COMBINE(D#2015-24-12, LTOD#12:13:14) liefert ein DTL#2015-24-12-12:13:14
- myDTL := T_CONV(D#2015-24-12); myDTL := T_CONV(LTOD#12:13:14) hat als Ergebnis im myDTL = DTL#1970-1-1-12:13:14

Siehe auch

Implizite Konvertierung von DATE (Seite 410)
Übersicht über die Datentypkonvertierung (Seite 369)
DATE (Seite 271)

Explizite Konvertierung von TOD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TOD	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	TOD_TO_BYTE
	WORD[1)]	X		TOD_TO_WORD
	DWORD[1)]	X		TOD_TO_DWORD
	LWORD[1)]	X		TOD_TO_LWORD
	SINT	X	Als Ergebnis wird die Anzahl der Millisekunden seit Mitternacht geliefert.	TOD_TO_SINT
	USINT	X		TOD_TO_USINT
	INT	X		TOD_TO_INT
	UINT	X		TOD_TO_UINT
	DINT	X		TOD_TO_DINT
	UDINT	X	Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).	TOD_TO_UDINT, T_CONV
	LINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	TOD_TO_LINT
	ULINT	X		TOD_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	X	Als Ergebnis wird die Zeitdauer seit Mitternacht geliefert.	TOD_TO_TIME
	LTIME	-	Keine explizite Konvertierung	-
	DT	X	Die Konvertierung setzt die Uhrzeit im Zieldatentyp.	TOD_TO_DT
	LDT	X		TOD_TO_LDT
	DTL	X		TOD_TO_DTL
	DATE	-	Keine explizite Konvertierung	-
	LTOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus TOD#1:0:0.0 wird LTOD#1:0:0.0)	TOD_TO_LTOD
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von TOD (Seite 412)
Übersicht über die Datentypkonvertierung (Seite 369)
TOD (TIME_OF_DAY) (Seite 271)

Explizite Konvertierung von LTOD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LTOD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LTOD	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LTOD_TO_BYTE
	WORD[1)]	X		LTOD_TO_WORD
	DWORD[1)]	X		LTOD_TO_DWORD
	LWORD[1)]	X		LTOD_TO_LWORD
	SINT	X	Als Ergebnis wird die Anzahl der Nanosekunden seit Mitternacht geliefert.	LTOD_TO_SINT
	USINT	X		LTOD_TO_USINT
	INT	X		LTOD_TO_INT
	UINT	X		LTOD_TO_UINT
	DINT	X		LTOD_TO_DINT
	UDINT	X		LTOD_TO_UDINT
	LINT	X		LTOD_TO_LINT
	ULINT	X		LTOD_TO_ULINT, T_CONV
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	LTOD_TO_LTIME
	DT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTOD#1:0:0.0 wird DT#1:0:0.0)	LTOD_TO_DT
	LDT	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTOD#1:0:0.0 wird LDT#1:0:0.0)	LTOD_TO_LDT
	DTL	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTOD#1:0:0.0 wird DTL#1:0:0.0)	LTOD_TO_DTL
	DATE	-	Keine explizite Konvertierung	-
	TOD	X	Hier findet eine Wandlung des Quellwerts in den Zieldatentyp statt (z. B. aus LTOD#1:0:0.0 wird TOD#1:0:0.0)	LTOD_TO_TOD
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und die Datentypen CHAR und WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Implizite Konvertierung von LTOD (Seite 414)
Übersicht über die Datentypkonvertierung (Seite 369)
LTOD (LTIME_OF_DAY) (Seite 272)

3.15.3.6 Zeichenfolgen

Explizite Konvertierung von CHAR

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
CHAR	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	CHAR_TO_BYTE
	WORD[1)]	X		CHAR_TO_WORD
	DWORD[1)]	X		CHAR_TO_DWORD
	LWORD[1)]	X		CHAR_TO_LWORD
	SINT	X		CHAR_TO_SINT
	USINT	X		CHAR_TO_USINT
	INT	X		CHAR_TO_INT
	UINT	X		CHAR_TO_UINT
	DINT	X		CHAR_TO_DINT
	UDINT	X		CHAR_TO_UDINT
	LINT	X		CHAR_TO_LINT
	ULINT	X		CHAR_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in das erste Zeichen der Zeichenkette (STRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert.	CHAR_TO_STRING, S_CONV
	WSTRING	-	Keine explizite Konvertierung	-
	WCHAR	X		CHAR_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich [1] Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp WCHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

Implizite Konvertierung von CHAR (Seite 416)

Übersicht über die Datentypkonvertierung (Seite 369)

CHAR (Seite 276)

Explizite Konvertierung von WCHAR

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WCHAR:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WCHAR	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WCHAR_TO_BYTE
	WORD[1)]	X		WCHAR_TO_WORD
	DWORD[1)]	X		WCHAR_TO_DWORD
	LWORD[1)]	X		WCHAR_TO_LWORD
	SINT	X		WCHAR_TO_SINT
	USINT	X		WCHAR_TO_USINT
	INT	X		WCHAR_TO_INT
	UINT	X		WCHAR_TO_UINT
	DINT	X		WCHAR_TO_DINT
	UDINT	X		WCHAR_TO_UDINT
	LINT	X		WCHAR_TO_LINT
	ULINT	X		WCHAR_TO_ULINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	X	Der Wert wird in das erste Zeichen der Zeichenkette (WSTRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert.	WCHAR_TO_WSTRING
	CHAR	X		WCHAR_TO_CHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bits kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

Übersicht über die Datentypkonvertierung (Seite 369)
WCHAR (Seite 276)
Implizite Konvertierung von WCHAR (Seite 418)

Explizite Konvertierung von STRING

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp STRING:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
STRING	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	X	Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (STRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen: • Ziffer • Vorzeichen • Punkt Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	STRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		STRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		STRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		STRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		STRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		STRING_TO_UDINT, S_CONV, STRG_VAL
	LINT	X		STRING_TO_LINT, S_CONV, STRG_VAL
	ULINT	X		STRING_TO_ULINT, S_CONV, STRG_VAL
	REAL	X		STRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		STRING_TO_LREAL, S_CONV, STRG_VAL
	S5TIME	-	Keine explizite Konvertierung	-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	CHAR	X	Das erste Zeichen der Zeichenkette (STRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird der Wert "0" in den Zieldatentyp geschrieben.	STRING_TO_CHAR, S_CONV
	WCHAR	-	Keine explizite Konvertierung	-
	WSTRING	X		STRING_TO_WSTRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Implizite Konvertierung von STRING (Seite 419)
Übersicht über die Datentypkonvertierung (Seite 369)
STRING (Seite 277)

Explizite Konvertierung von WSTRING

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WSTRING:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WSTRING	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	X	Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (WSTRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen: • Ziffer • Vorzeichen • Punkt Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	WSTRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		WSTRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		WSTRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		WSTRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		WSTRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		WSTRING_TO_UDINT, S_CONV, STRG_VAL
	LINT	X		WSTRING_TO_LINT, S_CONV, STRG_VAL
	ULINT	X		WSTRING_TO_ULINT, S_CONV, STRG_VAL
	REAL	X		WSTRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		WSTRING_TO_LREAL, S_CONV, STRG_VAL
	S5TIME	-	Keine explizite Konvertierung	-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	CHAR	-		-
	WCHAR	X	Das erste Zeichen der Zeichenkette (WSTRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird der Wert "0" in den Zieldatentyp geschrieben.	WSTRING_TO_WCHAR
	STRING	X		WSTRING_TO_STRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Übersicht über die Datentypkonvertierung (Seite 369)

WSTRING (Seite 280)

Implizite Konvertierung von WSTRING (Seite 421)

3.16 Datentypkonvertierungen bei S7-1200

3.16.1 Übersicht über die Datentypkonvertierung

Einführung

Wenn Sie mehrere Operanden in einer Anweisung verknüpfen, müssen Sie die Verträglichkeit ihrer Datentypen beachten. Dies gilt auch bei Zuweisungen oder beim Versorgen von Bausteinparametern. Sind die Operanden ungleichen Datentyps, muss eine Konvertierung durchgeführt werden.

Es gibt zwei Möglichkeiten der Konvertierung:

- Implizite Konvertierung
  Die Konvertierung erfolgt automatisch beim Ausführen der Anweisung.
- Explizite Konvertierung
  Sie verwenden eine explizite Konvertierungsanweisung, bevor die eigentliche Anweisung ausgeführt wird.

Hinweis

Die beschriebenen Möglichkeiten der Datentypkonvertierungen beziehen sich immer auf den neuesten Stand der CPU (V. 4). Es kann sein, dass als möglich gekennzeichnete Konvertierungen in den CPU Versionen 1 - 3 nicht zur Verfügung stehen.

Hinweis

Konvertieren von Bitfolgen in SCL

Alle Bitfolgen (BYTE, WORD und DWORD) werden in Ausdrücken wie die entsprechenden Ganzzahlen ohne Vorzeichen (USINT, UINT und UDINT) behandelt. Daher wird z. B. die implizite Konvertierung von DWORD nach REAL wie eine Konvertierung von UDINT nach REAL durchgeführt.

Implizite Konvertierung

Eine implizite Konvertierung wird automatisch durchgeführt, wenn die Datentypen der Operanden kompatibel sind. Die Prüfung der Kompatibilität kann nach mehr oder weniger strengen Kriterien erfolgen:

- Mit IEC-Prüfung (Voreinstellung)
  Bei eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, BYTE, WORD, DINT, INT, SINT, UDINT, UINT, USINT, TIME, DT, STRING, CHAR und WCHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp BYTE erwartet wird.
- Ohne IEC-Prüfung
  Bei nicht eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, LREAL, BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, TIME, DTL, TOD, DATE, STRING, CHAR und WCHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp DWORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp WORD erwartet wird.
  - An Durchgangsparametern (InOut) muss die Bitlänge eines angegebenen Operanden mit der programmierten Bitlänge des jeweiligen Parameters übereinstimmen.

Hinweis

Implizite Konvertierung ohne IEC-Prüfung

Der Programmiereditor kennzeichnet Operanden, die implizit konvertiert werden, mit einem grauen Rechteck. Ein dunkelgraues Rechteck signalisiert, dass eine implizite Konvertierung ohne Genauigkeitsverlust möglich ist, z. B., wenn Sie den Datentyp SINT nach INT konvertieren. Ein hellgraues Rechteck signalisiert, dass eine implizite Konvertierung möglich ist, aber während der Laufzeit können Fehler auftreten. Wenn Sie z. B. den Datentyp DINT nach INT konvertieren und es zu einem Überlauf kommt, wird der Freigabeausgang ENO auf "0" gesetzt.

Weitere Informationen zu der Einstellung der IEC-Prüfung und der impliziten Konvertierung finden Sie unter "Siehe auch".

Explizite Konvertierung

Wenn die Operanden nicht kompatibel sind und deshalb eine implizite Konvertierung nicht möglich ist, können Sie eine explizite Konvertierungsanweisung nutzen. Die Konvertierungsanweisungen finden Sie in der Task Card "Anweisungen".

Ein möglicher Überlauf wird am Freigabeausgang ENO angezeigt. Ein Überlauf entsteht, wenn z. B. der Wert des Quelldatentyps größer als der Wert des Zieldatentyps ist.

Hinweis

Verschieben von Bitmustern

Wenn es sich bei der expliziten Konvertierung um das Verschieben eines Bitmusters handelt, dann wird der Freigabeausgang ENO nicht gesetzt.

Weitere Informationen zur expliziten Konvertierung finden Sie unter "Siehe auch".

Das folgende Bild zeigt ein Beispiel, in dem eine explizite Datentypkonvertierung durchgeführt werden muss:

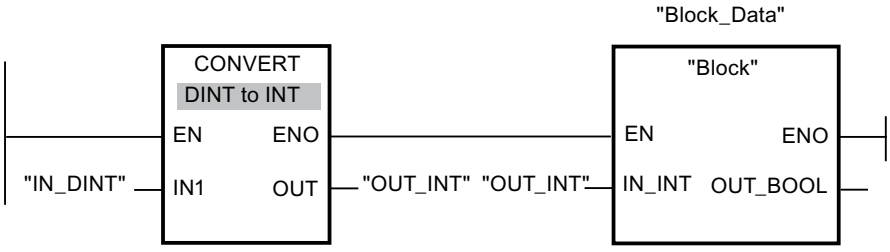

Der Funktionsbaustein "Block" erwartet am Eingangsparameter "IN_INT" eine Variable vom Datentyp INT. Der Wert der Variablen "IN_DINT" muss daher zunächst von DINT nach INT konvertiert werden. Befindet sich der Wert der Variablen "IN_DINT" innerhalb des zulässigen Wertebereichs des Datentyps INT, dann findet eine Konvertierung statt. Andernfalls wird ein Überlauf gemeldet. Doch auch im Falle eines Überlaufs findet trotzdem eine Konvertierung statt, aber die Werte werden abgeschnitten und der Freigabeausgang ENO auf "0" gesetzt.

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Implizite Konvertierungen (Seite 509)

Explizite Konvertierungen (Seite 530)

3.16.2 Implizite Konvertierungen

3.16.2.1 IEC-Prüfung aktivieren oder deaktivieren

Es wird geprüft, ob die Datentypen der verwendeten Operanden kompatibel sind. Diese Kompatibilitätsprüfung kann nach mehr oder weniger strengen Kriterien erfolgen. Bei aktivierter "IEC-Prüfung für Codebausteine" werden strengere Kriterien angewandt.

Sie können die IEC-Prüfung zentral für alle neuen Bausteine des Projekts oder für einzelne Bausteine einstellen.

IEC-Prüfung für neue Bausteine einstellen

Um die IEC-Prüfung für alle neuen Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Aktivieren oder deaktivieren Sie in der Gruppe "Voreinstellung für neue Bausteine" das Optionskästchen "IEC-Prüfung für Codebausteine".
   Die IEC-Prüfung wird für alle neuen Bausteine im Programm aktiviert oder deaktiviert.

IEC-Prüfung für einen Baustein einstellen

Um die IEC-Prüfung für einen Baustein einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "IEC-Prüfung".
   Die IEC-Prüfung wird für diesen Baustein aktiviert oder deaktiviert. Die Einstellung wird zusammen mit dem Projekt gespeichert.

3.16.2.2 Binärzahlen

Implizite Konvertierung von BOOL

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp BOOL ist nicht möglich.

Siehe auch

BOOL (Bit) (Seite 248)

Bitfolgen

Implizite Konvertierung von BYTE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
BYTE	BOOL	-	-	Keine implizite Konvertierung
	WORD	x	x	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DWORD	x	x	
	SINT	-	x	
	USINT	-	x	
	INT	-	x	
	UINT	-	x	
	DINT	-	x	
	UDINT	-	x	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	x	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	WCHAR	-	x	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

BYTE (Seite 249)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von BYTE (Seite 531)

Implizite Konvertierung von WORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert.
	DWORD	X	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	SINT	-	X	Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert.
	USINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

WORD (Seite 250)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von WORD (Seite 533)

Implizite Konvertierung von DWORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DWORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Die rechten Byte werden in den Zieldatentyp übertragen, die linken Byte werden ignoriert.
	WORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	UDINT	-	X	
	REAL	-	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	-	-	Keine implizite Konvertierung
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DTL	-	-	Keine implizite Konvertierung
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DATE	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DWORD (Seite 251)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von DWORD (Seite 536)

3.16.2.3 Ganzzahlen

Implizite Konvertierung von SINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp SINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
SINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	USINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertübertragung von z. B. SINT #-1 -> INT #-1, es wird nicht mit "0" aufgefüllt.)
	INT	X	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

SINT (8-Bit-Ganzzahlen) (Seite 254)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von SINT (Seite 540)

Implizite Konvertierung von USINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp USINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
USINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. USINT #10 -> DINT #10, oder USINT #128 -> SINT #-128)
	INT	X	X	
	UINT	X	X	
	DINT	X	X	
	UDINT	X	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

USINT (8-Bit-Ganzzahlen) (Seite 255)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von USINT (Seite 543)

Implizite Konvertierung von INT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
INT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. INT #-1 -> SINT #-1, oder INT #-32 767 -> UINT #32 769)
	USINT	-	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

INT (16-Bit-Ganzzahlen) (Seite 256)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von INT (Seite 546)

Implizite Konvertierung von UINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
UINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. UINT #100 -> DINT #100, oder UINT #60 000 -> INT #-5536)
	USINT	-	X	
	INT	-	X	
	DINT	X	X	
	UDINT	X	X	
	REAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)
	LREAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

UINT (16-Bit-Ganzzahlen) (Seite 257)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von UINT (Seite 550)

Implizite Konvertierung von DINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	UDINT	-	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DTL	-	-	Keine implizite Konvertierung
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DATE	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DINT (32-Bit-Ganzzahlen) (Seite 258)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von DINT (Seite 553)

Implizite Konvertierung von UDINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UDINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
UDINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	REAL	-	X	Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608)
	LREAL	X	X	Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DTL	-	-	Keine implizite Konvertierung
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DATE	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	WCHAR	-	X	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

UDINT (32-Bit-Ganzzahlen) (Seite 259)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von UDINT (Seite 557)

3.16.2.4 Gleitpunktzahlen

Implizite Konvertierung von REAL

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp REAL:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
REAL	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	SINT	-	X	Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. REAL #2.5 -> INT #2, oder negative Zahlen REAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LREAL	X	X	Der Wert wird in den Zieldatentyp übertragen.
	TIME	-	-	Keine implizite Konvertierung
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

REAL (Seite 264)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von REAL (Seite 560)

Implizite Konvertierung von LREAL

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LREAL:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
LREAL	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	SINT	-	X	Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. REAL #2.5 -> INT #2, oder negative Zahlen REAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

Explizite Konvertierung von LREAL (Seite 563)

3.16.2.5 Zeiten

Implizite Konvertierung von TIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	SINT	-	-	Keine implizite Konvertierung
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	DTL	-	-	
	TOD	-	X	Wenn der Quellwert kleiner als 24 Stunden (86 400 00 ms) ist, dann wird das Bitmuster des Quellwerts ohne Änderung in den Zieldatentyp übertragen. Ansonsten wird der Zielwert nicht verändert. Das Ergebnis der Konvertierung zeigt die Zeit, die seit Mitternacht vergangen ist.
	DATE	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

TIME (IEC-Zeit) (Seite 269)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Übersicht über die Datentypkonvertierung (Seite 507)
Explizite Konvertierung von TIME (Seite 566)

3.16.2.6 Datum und Uhrzeit

Implizite Konvertierung von DATE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DATE	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	DWORD	-	-	Keine implizite Konvertierung
	SINT	-	-	
	USINT	-	-	
	INT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	UINT	-	X	
	DINT	-	-	Keine implizite Konvertierung
	UDINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	DTL	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	TOD	-	-	Keine implizite Konvertierung
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DATE (Seite 271)
IEC-Prüfung aktivieren oder deaktivieren (Seite 509)
Übersicht über die Datentypkonvertierung (Seite 507)
Explizite Konvertierung von DATE (Seite 569)

Implizite Konvertierung von TOD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TOD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	SINT	-	-	Keine implizite Konvertierung
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	DTL	-	-	Keine implizite Konvertierung
	DATE	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

TOD (TIME_OF_DAY) (Seite 271)
IEC-Prüfung aktivieren oder deaktivieren (Seite 509)
Übersicht über die Datentypkonvertierung (Seite 507)
Explizite Konvertierung von TOD (Seite 570)

Implizite Konvertierung von DTL

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp DTL ist nicht möglich.

Siehe auch

Explizite Konvertierung von DTL (Seite 571)

3.16.2.7 Zeichenfolgen

Implizite Konvertierung von CHAR

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
CHAR	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden von links mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	WCHAR	-	-	
	STRING	X	X	Der STRING wird auf die Länge 1 gekürzt und enthält das Zeichen.
	WSTRING	-	-	Keine implizite Konvertierung
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

CHAR (Seite 276)
IEC-Prüfung aktivieren oder deaktivieren (Seite 509)
Übersicht über die Datentypkonvertierung (Seite 507)
Explizite Konvertierung von CHAR (Seite 573)

Implizite Konvertierung von WCHAR

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WCHAR:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WCHAR	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden von links mit "0" aufgefüllt.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	CHAR	-	-	
	STRING	-	-	
	WSTRING	X	X	Der WSTRING wird auf die Länge 1 gekürzt und enthält das Zeichen.
x: Konvertierung möglich -: Konvertierung nicht möglich				

Implizite Konvertierung von STRING

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp STRING:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
STRING	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	DATE	-	-	
	TOD	-	-	
	CHAR	-	X	Das erste Zeichen des STRING wird zurückgegeben, wenn der STRING ein oder mehrere Zeichen enthält. Sonst wird das Zeichen mit der Codierung $00 ausgegeben.
	WCHAR	-	-	Keine implizite Konvertierung
	WSTRING	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

Explizite Konvertierung von STRING (Seite 575)

Implizite Konvertierung von WSTRING

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WSTRING:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WSTRING	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	DATE	-	-	
	TOD	-	-	
	CHAR	-	-	
	WCHAR	-	X	Das erste Zeichen des WSTRING wird zurückgegeben, wenn der WSTRING ein oder mehr Zeichen enthält. Sonst wird das Zeichen mit der Codierung $0000 ausgegeben.
	STRING	-	-	Keine implizite Konvertierung
x: Konvertierung möglich -: Konvertierung nicht möglich				

3.16.3 Explizite Konvertierungen

3.16.3.1 Binärzahlen

Explizite Konvertierung von BOOL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BOOL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BOOL	BYTE	X	Im Zieldatentyp wird nur das LSB (Least Significant Bit) gesetzt. Der Freigabeausgang ENO ist immer "1".	BOOL_TO_BYTE
	WORD	X		BOOL_TO_WORD
	DWORD	X		BOOL_TO_DWORD
	SINT	X		BOOL_TO_SINT
	USINT	X		BOOL_TO_USINT
	INT	X		BOOL_TO_INT
	UINT	X		BOOL_TO_UINT
	DINT	X		BOOL_TO_DINT
	UDINT	X		BOOL_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	-		-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

BOOL (Bit) (Seite 248)

Implizite Konvertierung von BYTE (Seite 511)

Übersicht über die Datentypkonvertierung (Seite 507)

Bitfolgen

Explizite Konvertierung von BYTE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BYTE[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	BYTE_TO_BOOL
	WORD[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_WORD
	DWORD[1)]	X		BYTE_TO_DWORD
	SINT	X		BYTE_TO_SINT
	USINT	X		BYTE_TO_USINT
	INT	X		BYTE_TO_INT
	UINT	X		BYTE_TO_UINT
	DINT	X		BYTE_TO_DINT
	UDINT	X		BYTE_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_TIME
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_TOD
	DATE	X		BYTE_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	BYTE_TO_CHAR
	WCHAR	X		BYTE_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.				

Siehe auch

BYTE (Seite 249)

Implizite Konvertierung von BYTE (Seite 511)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von WORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WORD[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	WORD_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_BYTE
	DWORD[1)]	X		WORD_TO_DWORD

	SINT	X	ENO = TRUE
			#sint1 := WORD_TO_SINT(16#FFFF); // -1 bis
			#sint1 := WORD_TO_SINT(16#FF80); // -128
			#sint1 := WORD_TO_SINT(16#0); // 0 bis
			#sint1 := WORD_TO_SINT(16#007F); // 127
			
			ENO = FALSE
			#sint1 := WORD_TO_SINT(16#FF7F); // -129 bis
			#sint1 := WORD_TO_SINT(16#8000); // -32768
			#sint1 := WORD_TO_SINT(16#0080); // 128 bis
			#sint1 := WORD_TO_SINT(16#7FFF); // 32767
				WORD_TO_SINT

	USINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_USINT
	INT	X		WORD_TO_INT
	UINT	X		WORD_TO_UINT
	DINT	X		WORD_TO_DINT
	UDINT	X		WORD_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_TIME
	DTL	-	Keine explizite Konvertierung	-

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_TOD
	DATE	X		WORD_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_CHAR
	WCHAR	X		WORD_TO_WCHAR
WORD_BCD16	INT	X	Der zu konvertierende Wert hat den Datentyp WORD und wird als BCD-codierter Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ INT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	WORD_BCD16_TO_INT
BCD16	INT	X		BCD16_TO_INT

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.

Siehe auch

WORD (Seite 250)

Implizite Konvertierung von WORD (Seite 512)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von DWORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DWORD[1)]	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	DWORD_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_BYTE
	WORD[1)]	X		DWORD_TO_WORD

	SINT	X	ENO = TRUE
			#sint1 := DWORD_TO_SINT(16#FFFF_FFFF); // -1 bis
			#sint1 := DWORD_TO_SINT(16#FFFF_FF80); // -128
			#sint1 := DWORD_TO_SINT(16#0); // 0 bis
			#sint1 := DWORD_TO_SINT(16#0000_007F); // 127
			
			ENO = FALSE
			#sint1 := DWORD_TO_SINT(16#FFFF_FF7F); // -129
			#sint1 := DWORD_TO_SINT(16#8000_0000); // -2147483648
			#sint1 := DWORD_TO_SINT(16#0000_0080); // 128 bis
			#sint1 := DWORD_TO_SINT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_SINT

	USINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_USINT
	INT	X	ENO = TRUE	DWORD_TO_INT

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung

			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 bis
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 bis
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647	

	UINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_UINT
	DINT	X		DWORD_TO_DINT
	UDINT	X		DWORD_TO_UDINT
	REAL	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand von ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand von ENO = 0.	DWORD_TO_REAL
	LREAL	-	Keine explizite Konvertierung	-
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_TIME
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_TOD
	DATE	X		DWORD_TO_DATE
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DWORD_TO_CHAR
	WCHAR	X		DWORD_TO_WCHAR

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DWORD_BCD32	DINT	X	Der zu konvertierende Wert hat den Datentyp DWORD und wird als BCD-codierter Wert zwischen -9999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ DINT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	DWORD_BCD32_TO_DINT
BCD32	DINT	X		BCD32_TO_DINT

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.

Siehe auch

DWORD (Seite 251)

Implizite Konvertierung von DWORD (Seite 513)

Übersicht über die Datentypkonvertierung (Seite 507)

3.16.3.2 Ganzzahlen

Explizite Konvertierung von SINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp SINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
SINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	SINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_BYTE
	WORD[1)]	X		SINT_TO_WORD
	DWORD[1)]	X		SINT_TO_DWORD
	USINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_USINT
	INT	X		SINT_TO_INT
	UINT	X		SINT_TO_UINT
	DINT	X		SINT_TO_DINT
	UDINT	X		SINT_TO_UDINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	SINT_TO_REAL, NORM_X
	LREAL	X		SINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	SINT_TO_TIME
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)	SINT_TO_TOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)	SINT_TO_DATE

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • AWL / SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		SINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	SINT_TO_CHAR
	WCHAR[1)]	X		SINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

SINT (8-Bit-Ganzzahlen) (Seite 254)

Implizite Konvertierung von SINT (Seite 514)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von USINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp USINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
USINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	USINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	USINT_TO_BYTE
	WORD[1)]	X		USINT_TO_WORD
	DWORD[1)]	X		USINT_TO_DWORD
	SINT	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichen wechselt, wird der Freigabeausgang ENO auf "0" gesetzt.	USINT_TO_SINT
	INT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	USINT_TO_INT
	UINT	X		USINT_TO_UINT
	DINT	X		USINT_TO_DINT
	UDINT	X		USINT_TO_UDINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	USINT_TO_REAL, NORM_X
	LREAL	X		USINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	USINT_TO_TIME
	DTL	-	Keine Explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	USINT_TO_TOD
	DATE	X		USINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	USINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		USINT_TO_WSTRING

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen.	USINT_TO_CHAR
	WCHAR[1)]	X		USINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

USINT (8-Bit-Ganzzahlen) (Seite 255)

Implizite Konvertierung von USINT (Seite 515)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von INT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
INT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	INT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_BYTE
	WORD[1)]	X		INT_TO_WORD
	DWORD[1)]	X		INT_TO_DWORD
	SINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_SINT
	USINT	X		INT_TO_USINT
	UINT	X		INT_TO_UINT
	DINT	X		INT_TO_DINT
	UDINT	X		INT_TO_UDINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	INT_TO_REAL, NORM_X
	LREAL	X		INT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	INT_TO_TIME
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	INT_TO_TOD
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp	INT_TO_DATE

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • AWL / SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_STRING, S_CONV, VAL_STRG[)]
	WSTRING	X		INT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	INT_TO_CHAR
	WCHAR[1)]	X		INT_TO_WCHAR
	BCD16	X	Der zu konvertierende Wert hat den Typ INT und wird als Ganzzahl mit einem Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ WORD vor. Es findet eine echte Wandlung statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	INT_TO_BCD16
	BCD16_WORD	X		INT_TO_BCD16_WORD

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

INT (16-Bit-Ganzzahlen) (Seite 256)
Implizite Konvertierung von INT (Seite 516)
Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von UINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
UINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	UINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn dabei Bit verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_BYTE
	WORD[1)]	X		UINT_TO_WORD
	DWORD[1)]	X		UINT_TO_DWORD
	SINT	X		UINT_TO_SINT
	USINT	X		UINT_TO_USINT
	INT	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_INT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	UINT_TO_DINT
	UDINT	X		UINT_TO_UDINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	UINT_TO_REAL, NORM_X
	LREAL	X		UINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	UINT_TO_TIME
	DTL	-	Keine Explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" ge-	UINT_TO_TOD

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			setzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	UINT_TO_DATE, T_CONV
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		UINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	UINT_TO_CHAR
	WCHAR[1)]	X		UINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

UINT (16-Bit-Ganzzahlen) (Seite 257)

Implizite Konvertierung von UINT (Seite 517)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von DINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	DINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_BYTE
	WORD[1)]	X		DINT_TO_WORD
	DWORD[1)]	X		DINT_TO_DWORD
	SINT	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_SINT
	USINT	X		DINT_TO_USINT
	INT	X		DINT_TO_INT
	UINT	X		DINT_TO_UINT
	UDINT	X		DINT_TO_UDINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).	DINT_TO_REAL, NORM_X
	LREAL	X		DINT_TO_LREAL, NORM_X
	TIME	X	Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.	DINT_TO_TIME, T_CONV
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0"	DINT_TO_TOD

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			gesetzt. (Interpretation in Millisekunden seit 0:0)	
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)	DINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. • KOP / FUP: Die ersten Zeichen der Zeichenkette werden mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts. Positive numerische Werte werden ohne Vorzeichen ausgegeben. • AWL / SCL: Es werden keine Leerzeichen eingefügt. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		DINT_TO_WSTRING
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF)). Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_CHAR
	WCHAR[1)]	X		DINT_TO_WCHAR
	BCD32	X	Der zu konvertierende Wert hat den Typ DINT und wird als Ganzzahl mit einem Wert zwischen –999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ DWORD vor. Bei einem Überlauf wird der Freigabeausgang auf "0" gesetzt. Es findet eine echte Wandlung	DINT_TO_BCD32
	BCD32_DWORD	X		DINT_TO_BCD32_DWORD

Quelle	Ziel	Konver-tierung	Erläuterung	Mnemonik der Anwei-sung
			statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt.	
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.				

Siehe auch

DINT (32-Bit-Ganzzahlen) (Seite 258)

Implizite Konvertierung von DINT (Seite 518)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von UDINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UDINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
UDINT	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	UDINT_TO_BOOL
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn dabei Bit verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_BYTE
	WORD[1)]	X		UDINT_TO_WORD
	DWORD[1)]	X		UDINT_TO_DWORD
	SINT	X		UDINT_TO_SINT
	USINT	X		UDINT_TO_USINT
	INT	X		UDINT_TO_INT
	UINT	X		UDINT_TO_UINT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_DINT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	UDINT_TO_REAL, NORM_X
	LREAL	X		UDINT_TO_LREAL, NORM_X
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen.	UDINT_TO_TIME
	DTL	-	Keine explizite Konvertierung	-
	TOD	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" ge-	UDINT_TO_TOD, T_CONV

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
			setzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)	
	DATE	X	Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)	UDINT_TO_DATE
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		UDINT_TO_WCHAR
	CHAR[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	UDINT_TO_CHAR
	WCHAR[1)]	X		UDINT_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

UDINT (32-Bit-Ganzzahlen) (Seite 259)

Implizite Konvertierung von UDINT (Seite 519)

Übersicht über die Datentypkonvertierung (Seite 507)

3.16.3.3 Gleitpunktzahlen

Explizite Konvertierung von REAL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp REAL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
REAL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.	REAL_TO_DWORD
	SINT	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich des Zieldatentyps überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	REAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		REAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		REAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		REAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		REAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		REAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LREAL	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab, z. B. TRUNC(2.5) = 2.0; CEIL(2.5) = 3.0	REAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 14 Zeichen.	REAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		REAL_TO_WSTRING
	CHAR	-	Keine explizite Konvertierung	-

Quelle	Ziel	Konvertie-rung	Erläuterung	Mnemonik der Anwei-sung
	WCHAR	-		-
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

REAL (Seite 264)

Implizite Konvertierung von REAL (Seite 520)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von LREAL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LREAL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
LREAL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	SINT	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	LREAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		LREAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		LREAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		LREAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		LREAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		LREAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	REAL	X	Der Wert wird in den Zieldatentyp konvertiert. Wenn bei der Konvertierung der erlaubte Wertebereich überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Ein Genauigkeitsverlust wird toleriert.	LREAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 21 Zeichen.	LREAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		LREAL_TO_WSTRING
	CHAR	-	Keine explizite Konvertierung	-
	WCHAR	-		-

Quelle	Ziel	Konvertie-rung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

LREAL (Seite 265)

Übersicht über die Datentypkonvertierung (Seite 507)

3.16.3.4 Zeiten

Explizite Konvertierung von TIME

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.	TIME_TO_BYTE
	WORD[1]	X		TIME_TO_WORD
	DWORD[1]	X		TIME_TO_DWORD
	SINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen.	TIME_TO_SINT
	USINT	X		TIME_TO_USINT
	INT	X		TIME_TO_INT
	UINT	X		TIME_TO_UINT
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.	TIME_TO_DINT, T_CONV
	UDINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen. Ein Vorzeichenwechsel führt dazu, dass der Freigabeausgang ENO "0" ist.	TIME_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	DTL	-		-
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn der Quellwert außerhalb des Wertebereichs von TOD liegt, dann wird der Zieldatentyp nicht verändert.	TIME_TO_TOD
	DATE	-	Keine explizite Konvertierung	-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

-: Konvertierung nicht möglich

[1] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

TIME (IEC-Zeit) (Seite 269)
Implizite Konvertierung von TIME (Seite 522)
Übersicht über die Datentypkonvertierung (Seite 507)

3.16.3.5 Datum und Uhrzeit

Explizite Konvertierung von DATE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DATE	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DATE_TO_BYTE
	WORD[1)]	X		DATE_TO_WORD
	DWORD[1)]	X		DATE_TO_DWORD
	SINT	X	Als Ergebnis wird die Anzahl der Tage seit dem 1.1.1990 geliefert.	DATE_TO_SINT
	USINT	X		DATE_TO_USINT
	INT	X		DATE_TO_INT
	UINT	X		DATE_TO_UINT
	DINT	X		DATE_TO_DINT
	UDINT	X		DATE_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	-		-
	DTL [2)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	DATE_TO_DTL
	TOD	-	Keine explizite Konvertierung	-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

[2)] Der Datentyp DTL wird wie folgt initialisiert: 1970-1-1-0:0:0. Eine Konvertierung von DATE_TO_DTL setzt nur den entsprechenden Teil des DTL. Der Rest des DTLs bleibt, wie er initialisiert wurde. Daraus ergeben sich folgende Unterschiede bei den Anweisungen "T_COMBINE" und "T_CONV":

- T_COMBINE(D#2015-24-12, LTOD#12:13:14) liefert ein DTL#2015-24-12-12:13:14
- myDTL := T_CONV(D#2015-24-12); myDTL := T_CONV(LTOD#12:13:14) hat als Ergebnis im myDTL = DTL#1970-1-1-12:13:14

Siehe auch

DATE (Seite 271)

Implizite Konvertierung von DATE (Seite 523)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von TOD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TOD	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.	TOD_TO_BYTE
	WORD[1)]	X		TOD_TO_WORD
	DWORD[1)]	X		TOD_TO_DWORD
	SINT	X	Als Ergebnis wird die Anzahl der Millisekunden seit Mitternacht geliefert.	TOD_TO_SINT
	USINT	X		TOD_TO_USINT
	INT	X		TOD_TO_INT
	UINT	X		TOD_TO_UINT
	DINT	X		TOD_TO_DINT
	UDINT	X	Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).	TOD_TO_UDINT, T_CONV
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	X	Als Ergebnis wird die Zeitdauer seit Mitternacht geliefert.	TOD_TO_TIME
	DTL	X	Als Ergebnis wird der Tag auf den 1.1.1970 gesetzt.	TOD_TO_DTL
	DATE	-	Keine explizite Konvertierung	-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Konvertierung möglich

-: Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

TOD (TIME_OF_DAY) (Seite 271)

Implizite Konvertierung von TOD (Seite 524)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von DTL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DTL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DTL	BYTE	-	Keine explizite Konvertierung	-
	WORD	-		-
	DWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	REAL	-		-
	LREAL	-		-
	TIME	-		-
	TOD	X	Bei der Konvertierung wird die Tageszeit aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben.	DTL_TO_TOD, T_CONV
	DATE	X	Bei der Konvertierung wird das Datum aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	DTL_TO_DATE, T_CONV
	STRING	-	Keine explizite Konvertierung	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DTL (Seite 274)
Übersicht über die Datentypkonvertierung (Seite 507)

3.16.3.6 Zeichenfolgen

Explizite Konvertierung von CHAR

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
CHAR	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	CHAR_TO_BYTE
	WORD[1)]	X		CHAR_TO_WORD
	DWORD[1)]	X		CHAR_TO_DWORD
	SINT	X		CHAR_TO_SINT
	USINT	X		CHAR_TO_USINT
	INT	X		CHAR_TO_INT
	UINT	X		CHAR_TO_UINT
	DINT	X		CHAR_TO_DINT
	UDINT	X		CHAR_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	-		-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	X	Der Wert wird in das erste Zeichen der Zeichenkette (STRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert.	CHAR_TO_STRING
	WSTRING	-	Keine explizite Konvertierung	-
	WCHAR	X		CHAR_TO_WCHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Siehe auch

CHAR (Seite 276)

Implizite Konvertierung von CHAR (Seite 526)

Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von WCHAR

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WCHAR:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WCHAR	BOOL	-	Keine explizite Konvertierung	-
	BYTE[1)]	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WCHAR_TO_BYTE
	WORD[1)]	X		WCHAR_TO_WORD
	DWORD[1)]	X		WCHAR_TO_DWORD
	SINT	X		WCHAR_TO_SINT
	USINT	X		WCHAR_TO_USINT
	INT	X		WCHAR_TO_INT
	UINT	X		WCHAR_TO_UINT
	DINT	X		WCHAR_TO_DINT
	UDINT	X		WCHAR_TO_UDINT
	REAL	-	Keine explizite Konvertierung	-
	LREAL	-		-
	TIME	-		-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	X	Der Wert wird in das erste Zeichen der Zeichenkette (WSTRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert.	WCHAR_TO_WSTRING
	CHAR	X		WCHAR_TO_CHAR

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

Explizite Konvertierung von STRING

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp STRING:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
STRING	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	SINT	X	Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (STRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen: • Ziffer • Vorzeichen • Punkt Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	STRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		STRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		STRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		STRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		STRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		STRING_TO_UDINT, S_CONV, STRG_VAL
	REAL	X		STRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		STRING_TO_LREAL, S_CONV, STRG_VAL
	TIME	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	CHAR	X	Das erste Zeichen der Zeichenkette (STRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird den Wert "0" in den Zieldatentyp geschrieben.	STRING_TO_CHAR, S_CONV
	WCHAR	-	Keine explizite Konvertierung	-
	WSTRING	X		STRING_TO_WSTRING
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

STRING (Seite 277)
Übersicht über die Datentypkonvertierung (Seite 507)

Explizite Konvertierung von WSTRING

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WSTRING:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WSTRING	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	SINT	X	Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (STRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen: • Ziffer • Vorzeichen • Punkt Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	WSTRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		WSTRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		WSTRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		WSTRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		WSTRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		WSTRING_TO_UDINT, S_CONV, STRG_VAL
	REAL	X		WSTRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		WSTRING_TO_LREAL, S_CONV, STRG_VAL
	TIME	-	Keine explizite Konvertierung	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	CHAR	-		-
	WCHAR	X	Das erste Zeichen der Zeichenkette (WSTRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird den Wert "0" in den Zieldatentyp geschrieben.	WSTRING_TO_WCHAR
	STRING	X		WSTRING_TO_STRING

x: Konvertierung möglich
- : Konvertierung nicht möglich

3.17 Datentypkonvertierungen bei S7-300, S7-400

3.17.1 Übersicht über die Datentypkonvertierungen

Einführung

Wenn Sie mehrere Operanden in einer Anweisung verknüpfen, müssen Sie die Verträglichkeit ihrer Datentypen beachten. Dies gilt auch bei Zuweisungen oder beim Versorgen von Bausteinparametern. Sind die Operanden ungleichen Datentyps, muss eine Konvertierung durchgeführt werden.

Es gibt zwei Möglichkeiten der Konvertierung:

- Implizite Konvertierung
  Die implizite Konvertierung wird von den Programmiersprachen KOP, FUP, SCL und GRAPH unterstützt. In der Programmiersprache AWL ist die implizite Konvertierung nicht möglich.
- Explizite Konvertierung

Hinweis

Konvertieren von Bitfolgen in SCL

Alle Bitfolgen (BYTE, WORD und DWORD) werden in Ausdrücken wie die entsprechenden Ganzzahlen ohne Vorzeichen (USINT, UINT und UDINT) behandelt. Daher wird z. B. die implizite Konvertierung von DWORD nach REAL wie eine Konvertierung von UDINT nach REAL durchgeführt.

Implizite Konvertierung

Eine implizite Konvertierung wird automatisch durchgeführt, wenn die Datentypen der Operanden kompatibel sind. Die Prüfung der Kompatibilität kann nach mehr oder weniger strengen Kriterien erfolgen:

- Mit IEC-Prüfung
  In den Programmiersprachen KOP, FUP und GRAPH werden bei eingestellter IEC-Prüfung die folgenden Regeln angewandt:
  - Nur die Datentypen BYTE und WORD können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp BYTE erwartet wird.

  In der Programmiersprache SCL werden bei eingestellter IEC-Prüfung die folgenden Regeln angewandt:
  - Die implizite Konvertierung von Bitfolgen in anderen Datentypen ist nicht möglich. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp INT erwartet wird.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp BYTE erwartet wird.
- Ohne IEC-Prüfung (Voreinstellung)
  In den Programmiersprachen KOP, FUP und GRAPH werden bei nicht eingestellter IEC-Prüfung die folgenden Regeln angewandt:
  - Die Datentypen BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, TOD, DATE und CHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp DWORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp WORD erwartet wird.

  In der Programmiersprache SCL werden bei nicht eingestellter IEC-Prüfung die folgenden Regeln angewandt:
  - Die implizite Konvertierung von Bitfolgen in anderen Datentypen ist möglich. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter angegeben werden, wenn an diesem der Datentyp INT erwartet wird.
  - Die implizite Konvertierung von Bitfolgen in Gleitpunktzahlen ist nicht möglich. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp REAL erwartet wird.
  - Die implizite Konvertierung von Bitfolgen in den Datentypen TIME, TOD, DATE und CHAR ist nur dann möglich, wenn diese die gleiche Bitlänge aufweisen. Ein Operand vom Datentyp DWORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp DATE erwartet wird.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp DINT kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp INT erwartet wird.
  - An Durchgangsparametern (InOut) muss die Bitlänge eines angegebenen Operanden mit der programmierten Bitlänge für den jeweiligen Parameter übereinstimmen.

Explizite Konvertierung

Wenn die Operanden nicht kompatibel sind und deshalb eine implizite Konvertierung nicht möglich ist, können Sie eine explizite Konvertierungsanweisung nutzen. Die Konvertierungsanweisungen finden Sie in der Task Card "Anweisungen".

Ein möglicher Überlauf wird am Freigabeausgang ENO angezeigt. Ein Überlauf entsteht, wenn z. B. der Wert des Quelldatentyps größer als der Wert des Zieldatentyps ist.

Hinweis

Verschieben von Bitmustern

Wenn es sich bei der expliziten Konvertierung um das Verschieben eines Bitmusters handelt, dann wird der Freigabeausgang ENO nicht gesetzt.

Weitere Informationen zur expliziten Konvertierung finden Sie unter "Siehe auch".

Das folgende Bild zeigt ein Beispiel, in dem eine explizite Datentypkonvertierung durchgeführt werden muss:

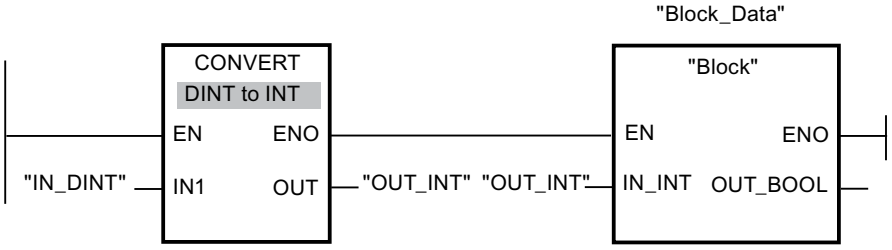

Der Funktionsbaustein "Block" erwartet am Eingangsparameter "IN_INT" eine Variable vom Datentyp INT. Der Wert der Variablen "IN_DINT" muss daher zunächst von DINT nach INT konvertiert werden. Befindet sich der Wert der Variablen "IN_DINT" innerhalb des zulässigen Wertebereichs des Datentyps INT, dann findet eine Konvertierung statt. Andernfalls wird ein Überlauf gemeldet. Doch auch im Falle eines Überlaufs findet trotzdem eine Konvertierung statt, aber die Werte werden abgeschnitten und der Freigabeausgang ENO auf "0" gesetzt.

Siehe auch

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierungen (Seite 580)

Explizite Konvertierungen (Seite 592)

3.17.2 Implizite Konvertierungen

3.17.2.1 IEC-Prüfung aktivieren oder deaktivieren

Es wird geprüft, ob die Datentypen der verwendeten Variablen kompatibel sind. Diese Kompatibilitätsprüfung kann nach mehr oder weniger strengen Kriterien erfolgen. Bei aktivierter "IEC-Prüfung für Codebausteine" werden strengere Kriterien angewandt.

Sie können die IEC-Prüfung zentral für alle neuen Bausteine des Projekts oder für einzelne Bausteine einstellen.

IEC-Prüfung für neue Bausteine einstellen

Um die IEC-Prüfung für alle neuen Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Aktivieren oder deaktivieren Sie in der Gruppe "Voreinstellung für neue Bausteine" das Optionskästchen "IEC-Prüfung für Codebausteine".
   Die IEC-Prüfung wird für alle neuen Bausteine im Programm aktiviert oder deaktiviert.

IEC-Prüfung für einen Baustein einstellen

Um die IEC-Prüfung für einen Baustein einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "IEC-Prüfung".
   Die IEC-Prüfung wird für diesen Baustein aktiviert oder deaktiviert. Die Einstellung wird zusammen mit dem Projekt gespeichert.

Siehe auch

Übersicht über die Datentypkonvertierungen (Seite 578)

3.17.2.2 Binärzahlen

Implizite Konvertierung von BOOL

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp BOOL ist nicht möglich.

Siehe auch

BOOL (Bit) (Seite 248)

Bitfolgen

Implizite Konvertierung von BYTE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
BYTE	BOOL	-	-	Keine implizite Konvertierung
	WORD	X	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	DWORD	X	X	
	INT	-	-	Keine implizite Konvertierung
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	REAL	-	-	Keine implizite Konvertierung
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

BYTE (Seite 249)

Übersicht über die Datentypkonvertierungen (Seite 578)

Explizite Konvertierung von BYTE (Seite 593)

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von WORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
WORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Nur das niederwertige Byte wird in den Zieldatentyp übertragen.
	DWORD	X	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.
	INT	-	X	
	DINT	-	X	
	REAL	-	-	Keine implizite Konvertierung
	TIME	-	-	
	S5TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	DT	-	-	Keine implizite Konvertierung
	TOD	-	-	
	DATE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	STRING	-	-	Keine implizite Konvertierung
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

WORD (Seite 250)

Übersicht über die Datentypkonvertierungen (Seite 578)

Explizite Konvertierung von WORD (Seite 595)

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von DWORD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DWORD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	WORD	-	X	
	INT	-	-	Keine implizite Konvertierung
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	REAL	-	-	Keine implizite Konvertierung
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	S5TIME	-	-	Keine implizite Konvertierung
	DT	-	-	
	TOD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	DATE	-	-	Keine implizite Konvertierung
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DWORD (Seite 251)

Übersicht über die Datentypkonvertierungen (Seite 578)

Explizite Konvertierung von DWORD (Seite 597)

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.2.3 Ganzzahlen

Implizite Konvertierung von INT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
INT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	DWORD	-	-	Keine implizite Konvertierung
	DINT [1)]	X	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	REAL	X	X	
	TIME	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	

x: Konvertierung möglich
-: Konvertierung nicht möglich
[1)]: Nur in SCL möglich

Siehe auch

INT (16-Bit-Ganzzahlen) (Seite 256)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von INT (Seite 600)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von DINT

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DINT	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	INT	-	-	Keine implizite Konvertierung
	REAL	-	-	
	TIME	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	S5TIME	-	-	Keine implizite Konvertierung
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DINT (32-Bit-Ganzzahlen) (Seite 258)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von STRING (Seite 613)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.2.4 Gleitpunktzahlen

Implizite Konvertierung von REAL

Möglichkeiten zur impliziten Konvertierung

Die Implizite Konvertierung vom Datentyp REAL ist nicht möglich.

Siehe auch

REAL (Seite 264)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von CHAR (Seite 612)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.2.5 Zeiten

Implizite Konvertierung von TIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	INT	-	-	Keine implizite Konvertierung
	DINT	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	REAL	-	-	Keine implizite Konvertierung
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

TIME (IEC-Zeit) (Seite 269)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von TIME (Seite 607)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von S5TIME

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp S5TIME:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
S5TIME	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.
	DWORD	-	-	Keine implizite Konvertierung
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

S5TIME (Zeitdauer) (Seite 268)

Übersicht über die Datentypkonvertierungen (Seite 578)

Explizite Konvertierung von S5TIME (Seite 608)

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.2.6 Datum und Uhrzeit

Implizite Konvertierung von DATE

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
DATE	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990.
	DWORD	-	-	Keine implizite Konvertierung
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

DATE (Seite 271)

Übersicht über die Datentypkonvertierungen (Seite 578)

IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von TOD

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
TOD	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr).
	INT	-	-	Keine implizite Konvertierung
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

TOD (TIME_OF_DAY) (Seite 271)
Übersicht über die Datentypkonvertierungen (Seite 578)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von DT

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp DT ist nicht möglich.

Siehe auch

DT (DATE_AND_TIME) (Seite 272)
Übersicht über die Datentypkonvertierungen (Seite 578)

Explizite Konvertierung von DT (Seite 611)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.2.7 Zeichenfolgen

Implizite Konvertierung von CHAR

Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Mit IEC-Prüfung	Ohne IEC-Prüfung	Erläuterung
CHAR	BOOL	-	-	Keine implizite Konvertierung
	BYTE	-	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
	WORD	-	-	Keine implizite Konvertierung
	DWORD	-	-	
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	X	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.
x: Konvertierung möglich -: Konvertierung nicht möglich				

Siehe auch

CHAR (Seite 276)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von CHAR (Seite 612)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

Implizite Konvertierung von STRING

Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp STRING ist nicht möglich.

Siehe auch

STRING (Seite 277)
Übersicht über die Datentypkonvertierungen (Seite 578)
Explizite Konvertierung von STRING (Seite 613)
IEC-Prüfung aktivieren oder deaktivieren (Seite 580)

3.17.3 Explizite Konvertierungen

3.17.3.1 Binärzahlen

Explizite Konvertierung von BOOL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BOOL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BOOL	BYTE	X	Im Zieldatentyp wird nur das LSB (Least Significant Bit) gesetzt. Der Freigabeausgang ENO ist immer "1".	CONVERT
	WORD	X		
	DWORD	X		
	INT	X		BOOL_TO_INT
	DINT	X		BOOL_TO_DINT
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

Übersicht über die Datentypkonvertierungen (Seite 578)
BOOL (Bit) (Seite 248)
Implizite Konvertierungen (Seite 580)

Bitfolgen

Explizite Konvertierung von BYTE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BYTE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
BYTE	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	CONVERT
	WORD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden auf "0" gesetzt.	CONVERT
	DWORD	X		
	INT	X		
	DINT	X		
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden auf "0" gesetzt.	CONVERT
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

BYTE (Seite 249)
Übersicht über die Datentypkonvertierungen (Seite 578)
Implizite Konvertierung von BYTE (Seite 582)

Explizite Konvertierung von WORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WORD	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	CONVERT
	BYTE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn der erlaubte Wertebereich des Zieldatentyps überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. In diesem Fall ist das Ergebnis der Konvertierung ungültig.	
	DWORD	X		
	INT	X		
	DINT	X		
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	MOVE
	DT	-	Keine explizite Konvertierung	-
	TOD	-		
	DATE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	MOVE
	STRING	-	Keine explizite Konvertierung	-
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	WORD_TO_CHAR
	BLOCK_DB	X	Das Bitmuster von WORD wird als Datenbausteinnummer interpretiert.	WORD_TO_BLOCK_DB

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
WORD_BCD[1) 2)]	INT	X	Wenn der erlaubte Wertebereich des Zieldatentyps überschritten wird oder eine ungültige Tetrade im Bereich A..F vorhanden ist, wird der Freigabeausgang ENO auf "0" gesetzt. In diesem Fall ist das Ergebnis der Konvertierung ungültig.	WORD_BCD_TO_INT
BCD[1) 2)]	INT	X		BCD_TO_INT

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Der zu konvertierende Wert hat den Datentyp WORD und wird als BCD-codierter Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ INT vor.

[2)] Gültig für SCL: Wenn im BCD-Code ein BCD-Fehler enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in STOP.

Siehe auch

WORD (Seite 250)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von WORD (Seite 583)

Explizite Konvertierung von DWORD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DWORD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DWORD	BOOL	X	Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	CONVERT
	BYTE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	
	WORD	X		

	INT	X	ENO = TRUE
			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 bis
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 bis
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647	

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	MOVE
	REAL	X		
	TIME	X		
	S5TIME	-	Keine explizite Konvertierung	-
	DT	-		
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	MOVE
	DATE	-	Keine explizite Konvertierung	-
	STRING	-		
	CHAR	-		
DWORD_BCD[1) 2)]	DINT	X	Wenn der erlaubte Wertebereich des Zieldatentyps überschritten wird oder eine ungültige Tetrade im Bereich A..F vorhanden ist, wird der Freigabeausgang ENO auf "0" gesetzt. In diesem Fall ist das Ergebnis der Konvertierung ungültig.	DWORD_BCD_TO_DINT
BCD[1) 2)]	DINT	X		BCD_TO_DINT

x: Konvertierung möglich

- : Konvertierung nicht möglich

[1)] Der zu konvertierende Wert hat den Datentyp DWORD und wird als BCD-codierter Wert zwischen -9999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ DINT vor.

[2)] Gültig für SCL: Wenn im BCD-Code ein BCD-Fehler enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in STOP.

Siehe auch

DWORD (Seite 251)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von DWORD (Seite 584)

3.17.3.2 Ganzzahlen

Explizite Konvertierung von INT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp INT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
INT	BOOL	X	Der Wert wird intern zunächst nach WORD und anschließend nach BOOL konvertiert. Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	INT_TO_BOOL
	BYTE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	CONVERT
	WORD	X		MOVE
	DWORD	X		
	DINT	X		CONVERT, ITD
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	CONVERT, SCALE, ITR
	TIME	-	Keine explizite Konvertierung	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	S_CONV, CONVERT
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei der Konvertierung von negativen Zahlen oder bei einem Überlauf, wird der Freigabeausgang ENO auf "0" gesetzt.	CONVERT
	BCD[1) 2)]	X	Wenn der erlaubte Wertebereich des Zieldatentyps überschritten wird oder eine ungültige Tetrade im Bereich A..F vorhanden ist, wird der Freigabeausgang ENO auf "0" gesetzt. In diesem Fall ist das Ergebnis der Konvertierung ungültig.	INT_TO_BCD
	BCD_WORD [1) 2)]	X		INT_TO_BCD_WORD

Quelle	Ziel	Konvertie-rung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Der zu konvertierende Wert hat den Typ INT und wird als Ganzzahl mit einem Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ WORD vor. 2) Gültig für SCL: Wenn im BCD-Code ein BCD-Fehler enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in STOP.				

Siehe auch

INT (16-Bit-Ganzzahlen) (Seite 256)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von INT (Seite 585)

Explizite Konvertierung von DINT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DINT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DINT	BOOL	X	Der Wert wird intern zunächst nach DWORD und anschließend nach BOOL konvertiert. Die folgenden Möglichkeiten können auftreten: • Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1". • Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1". • Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".	DINT_TO_BOOL
	BYTE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	CONVERT
	WORD	X		
	DWORD	X		MOVE
	INT	X		CONVERT
	REAL	X	Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).	CONVERT, DTR, SCALE
	TIME	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.	T_CONV, MOVE
	S5TIME	-	Keine explizite Konvertierung	-
	DT	-		
	TOD	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	CONVERT
	DATE	X		
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt.	S_CONV, CONVERT
	CHAR	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei der Konvertierung von negativen Zahlen oder bei einem Überlauf, wird der Freigabeausgang ENO auf "0" gesetzt.	DINT_TO_CHAR
	BCD[1) 2)]	X	Wenn der erlaubte Wertebereich des Zieldatentyps überschritten wird oder eine ungültige Tetrade im Bereich A..F vorhanden ist, wird der Freigabeausgang ENO auf "0" gesetzt. In diesem Fall ist das Ergebnis der Konvertierung ungültig.	DINT_TO_BCD
	BCD_DWORD[1) 2)]	X		DINT_TO_BCD_DWORD

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
x: Konvertierung möglich - : Konvertierung nicht möglich 1) Der zu konvertierende Wert hat den Datentyp DWORD und wird als BCD-codierter Wert zwischen -9999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ DINT vor. 2) Gültig für SCL: Wenn im BCD-Code ein BCD-Fehler enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in STOP.				

Siehe auch

DINT (32-Bit-Ganzzahlen) (Seite 258)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierungen (Seite 580)

3.17.3.3 Gleitpunktzahlen

Explizite Konvertierung von REAL

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp REAL:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
REAL	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	-		
	DWORD	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.	CONVERT, MOVE
	INT	X	Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich des Zieldatentyps überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	CONVERT, ROUND, RND, CEIL, RND+, FLOOR, RND-, TRUNC, UNSCALE
	DINT	X		
	TIME	-	Keine explizite Konvertierung	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt.	S_CONV, CONVERT
	CHAR	-	Keine explizite Konvertierung	-
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

REAL (Seite 264)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierungen (Seite 580)

3.17.3.4 Zeiten

Explizite Konvertierung von TIME

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	-		
	DWORD	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.	CONVERT
	INT	-	Keine explizite Konvertierung	-
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.	T_CONV, CONVERT
	REAL	-	Keine explizite Konvertierung	-
	S5TIME	X	Der Wert wird in das Format des Zieldatentyps konvertiert. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.	T_CONV, CONVERT
	DT	-	Keine explizite Konvertierung	-
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

TIME (IEC-Zeit) (Seite 269)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von TIME (Seite 587)

Explizite Konvertierung von S5TIME

Möglichkeiten zur expliziten Konvertierung in KOP, FUP, AWL und GRAPH

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp S5TIME:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
S5TIME	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	X	Der Wert wird in das Format des Zieldatentyps konvertiert.	S5TIME_TO_WORD
	DWORD	-	Keine explizite Konvertierung	-
	INT	-		
	DINT	-		
	REAL	-		
	TIME	X	Der Wert wird in das Format des Zieldatentyps konvertiert.	T_CONV, CONVERT
	DT	-	Keine explizite Konvertierung	-
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

S5TIME (Zeitdauer) (Seite 268)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von S5TIME (Seite 588)

3.17.3.5 Datum und Uhrzeit

Explizite Konvertierung von DATE

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DATE:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DATE	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	X	Der Wert wird in das Format des Zieldatentyps konvertiert.	DATE_TO_WORD
	DWORD	-	Keine explizite Konvertierung	-
	INT	X	Der Wert wird in das Format des Zieldatentyps konvertiert.	DATE_TO_INT
	DINT	X		DATE_TO_DINT
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	STRING	-		
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

DATE (Seite 271)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von DATE (Seite 589)

Explizite Konvertierung von TOD

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TOD:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
TOD	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	-		
	DWORD	X	Der Wert wird in das Format des Zieldatentyps konvertiert.	TOD_TO_DWORD
	INT	-	Keine explizite Konvertierung	-
	DINT	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	TOD_TO_DINT
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	-		
	DT	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

TOD (TIME_OF_DAY) (Seite 271)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von TOD (Seite 590)

Explizite Konvertierung von DT

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DT:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
DT	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	-		
	DWORD	-		
	INT	-		
	DINT	-		
	REAL	-		
	TIME	-		
	S5TIME	-		
	TOD	X	Bei der Konvertierung werden die Zeitangaben aus dem Format DTL extrahiert und in den Zieldatentyp übertragen.	T_CONV, CONVERT
	DATE	X	Bei der Konvertierung werden die Angaben zum Datum aus dem Format DTL extrahiert und in den Zieldatentyp übertragen.	
	STRING	-	Keine explizite Konvertierung	-
	CHAR	-		
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

DT (DATE_AND_TIME) (Seite 272)

Übersicht über die Datentypkonvertierungen (Seite 578)

3.17.3.6 Zeichenfolgen

Explizite Konvertierung von CHAR

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp CHAR:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
CHAR	BOOL	-	Keine explizite Konvertierung	-
	BYTE	X	Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.	CONVERT
	WORD	X		
	DWORD	X		
	INT	X		
	DINT	X		
	REAL	-	Keine explizite Konvertierung	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	Der Wert wird in das erste Zeichen der Zeichenkette (STRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert.	S_CONV, CONVERT
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

CHAR (Seite 276)

Übersicht über die Datentypkonvertierungen (Seite 578)

Implizite Konvertierung von CHAR (Seite 591)

Explizite Konvertierung von STRING

Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp STRING:

Quelle	Ziel	Konvertierung	Erläuterung	Mnemonik der Anweisung
STRING	BOOL	-	Keine explizite Konvertierung	-
	BYTE	-		
	WORD	-		
	DWORD	-		
	INT	X	Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (STRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen: • Ziffer • Vorzeichen • Punkt Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.	S_CONV, CONVERT
	DINT	X		
	REAL	X		
	TIME	-	Keine explizite Konvertierung	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	CHAR	X	Das erste Zeichen der Zeichenkette (STRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird den Wert "0" in den Zieldatentyp geschrieben.	S_CONV, CONVERT
x: Konvertierung möglich - : Konvertierung nicht möglich				

Siehe auch

STRING (Seite 277)
Übersicht über die Datentypkonvertierungen (Seite 578)
Implizite Konvertierungen (Seite 580)

3.17.3.7 Weitere Konvertierungsfunktionen

Weitere explizite Konvertierungsfunktionen in SCL

Weitere Möglichkeiten zur expliziten Konvertierung in SCL

Die folgende Tabelle zeigt die weiteren Möglichkeiten und Anweisungen zur expliziten Konvertierung in SCL:

Quelle	Ziel	Erläuterung	Mnemonik der Anweisung
WORD	BLOCK_DB	Das Bitmuster von WORD wird als Datenbausteinnummer interpretiert.	WORD_TO_BLOCK_DB
BLOCK_DB	WORD	Die Datenbausteinnummer wird als Bitmuster von WORD interpretiert.	BLOCK_DB_TO_WORD

Anweisungen

4

4.1 Anweisungen

4.1.1 Beispielbibliotheken

4.1.1.1 Sample Library for Instructions

Die “Sample Library for Instructions” ist die globale Bibliothek für die Programmbeispiele zu den Anweisungen in STEP 7 (TIA Portal). Diese Bibliothek bietet Ihnen mit ihren kompakten, leicht verständlichen Programmbeispielen in der Programmiersprache KOP oder SCL eine Hilfestellung bei der Programmierung. Voraussetzung zur Nutzung der Bibliothek ist mindestens eine CPU der Baureihe SIMATIC S7-1200 / S7-1500.

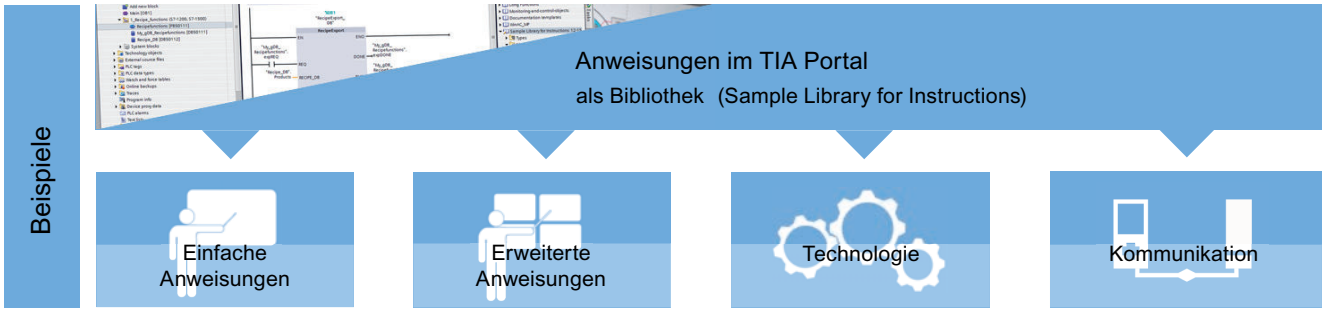

Programmbeispiele im TIA Portal Informationssystem

Die Programmbeispiele der “Sample Library for Instructions” decken sich mit den Programmbeispielen zu den Anweisungen des TIA Portal Informationssystems. Dementsprechend können Sie über die Hilfe (<F1>) im TIA Portal die Dokumentation des jeweiligen Programmbeispiels aufrufen.

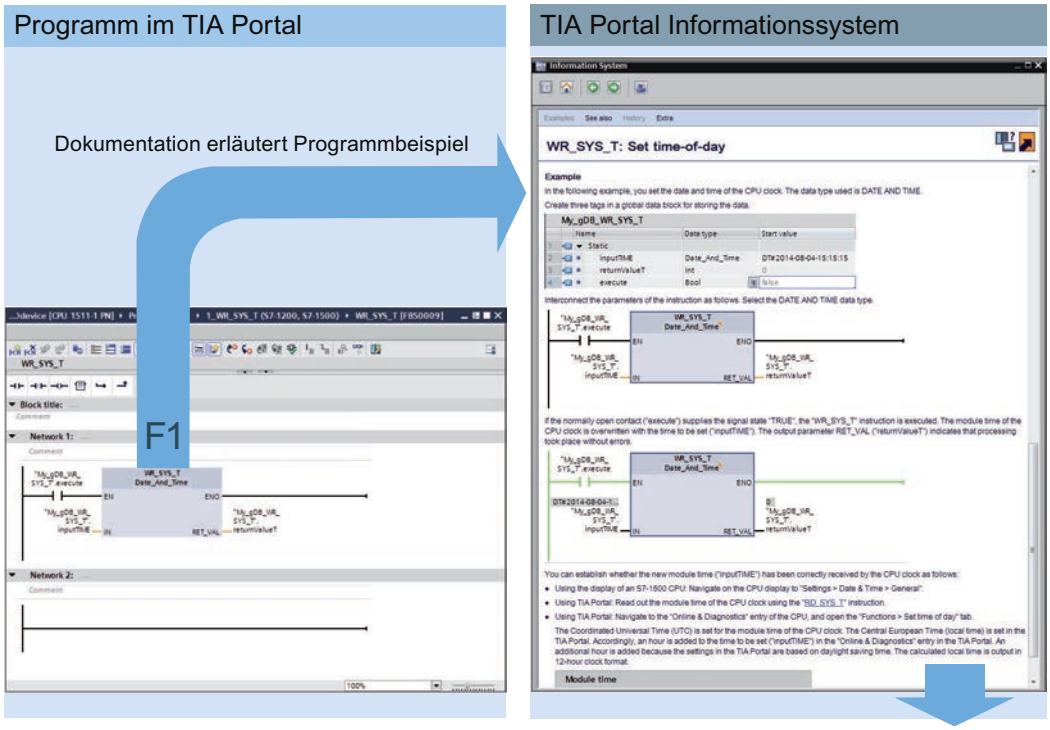

Bibliothek enthält Programmbeispiel aus der Dokumentation

Download der Bibliothek

Über die FAQ-Seite (https://support.industry.siemens.com/cs/ww/de/view/109476781) ist die “Sample Library for Instructions” hinterlegt.

Hinweis

Programmbeispiele korrekt in ein Projekt einfügen

Aufgrund unterschiedlicher Vorbedingungen und unterschiedlicher Funktionen können Programmbeispiele neben Programmbausteinen auch andere Komponenten (z. B. PLC-Variablen, PLC-Datentypen, Watchtables o. ä.) enthalten. Ebenso gibt es Programmbeispiele, die spezielle Hardware (mehrere CPUs, Dezentrale Peripherie, Eingabebaugruppen o. ä.) benötigen.

Halten Sie sich bitte an die folgenden Regeln:

- Fügen Sie niemals einfach alle Programmbeispiele bzw. den gesamten Ordner der “Sample Library for Instructions” in eine CPU ein.
- Vergewissern Sie sich, dass das entsprechende Programmbeispiel nicht aus Komponenten für mehrere CPUs besteht.
- Fügen Sie die einzelnen Komponenten der Programmbeispiele stets einzeln in den passenden Ordner Ihres Projekts ein. Achten Sie dabei auch auf die passende CPU.

4.1.1.2 Library of General Functions

Die “Library of General Functions” ist die globale Bibliothek für ergänzende Funktionen zu den Anweisungen in STEP 7 (TIA Portal). Diese Bibliothek bietet Ihnen, als Erweiterung des Umfangs an Anweisungen, Bausteine mit nützlichen Basisfunktionen, die oft in Automatisierungs-Projekten benötigt werden. Alle Inhalte der Bibliothek sind verwendbar mit einer CPU der Baureihe SIMATIC S7-1200(F) / S7-1500(F).

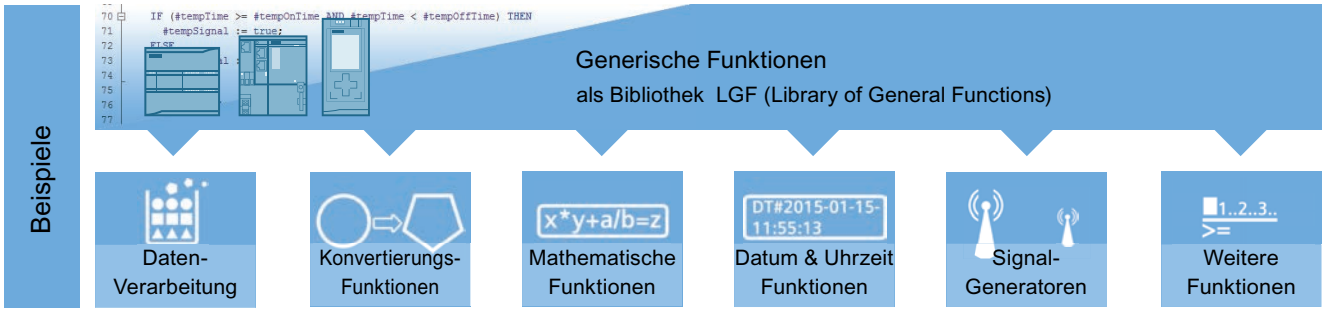

Allgemein verwendbare Funktionen

Die allgemein verwendbaren Funktionen der Bibliothek "Library of General Functions" werden Ihnen als Bausteine in der Programmiersprache SCL zur Verfügung gestellt. In der Bibliothek sind unter anderem folgende Funktionen enthalten: FIFO, Suchfunktion, Matrixberechnungen, Astrouhr. Sie können die Funktionen per Parametrierung sofort verwenden und universell einsetzen.

Download der Bibliothek

Über die FAQ-Seite (https://support.industry.siemens.com/cs/ww/de/view/109479728) ist die “Library of General Functions” hinterlegt.

4.1.2 Asynchron arbeitende Anweisungen

4.1.2.1 Unterschied zwischen synchron und asynchron arbeitenden Anweisungen

Asynchron arbeitende Anweisungen

In der Programmbearbeitung wird zwischen synchron und asynchron arbeitenden Anweisungen unterschieden.

Die Eigenschaften "synchron" bzw. "asynchron" beziehen sich auf den zeitlichen Zusammenhang zwischen Aufruf und Ausführung der Anweisung:

- Für synchrone Anweisungen gilt: Wenn der Aufruf einer synchron arbeitenden Anweisung beendet ist, ist auch die Ausführung beendet.
- Anders bei asynchronen Anweisungen: Die Ausführung einer asynchronen Anweisung kann sich über mehrere Aufrufe erstrecken. Die CPU bearbeitet asynchrone Anweisungen parallel zum zyklischen Anwenderprogramm. Asynchron arbeitende Anweisungen belegen während ihrer Bearbeitung Ressourcen in der CPU.

Bei asynchron arbeitenden Anweisungen handelt es sich in der Regel um Anweisungen für die Übertragung von Daten (Datensätze für Module, Kommunikationsdaten, Diagnosedaten, ...).

Weitere Informationen zu asynchron arbeitenden Anweisungen finden Sie hier: SIMATIC S7-1500 / ET 200MP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/86140384) im Systemhandbuch unter Grundlagen zur Programmbearbeitung > Asynchron arbeitende Anweisungen.

Identifikation des Auftrags

Um eine Anweisung über mehrere Aufrufe auszuführen, muss die CPU einen Folgeaufruf einem bereits laufenden Auftrag der Anweisung eindeutig zuordnen können. Für die Zuordnung Aufruf zu Auftrag nutzt die CPU, abhängig vom Typ der Anweisung, einen der beiden folgenden Mechanismen:

- Über die Instanz der Anweisung (bei Typ "SFB" Systemfunktionsbaustein)
- Über den Auftrag identifizierende Eingangsparameter der Anweisung

Falls Sie mit asynchron arbeitenden Anweisungen die Auslösung eines Prozessalarms oder die Ausgabe von Steuerkommandos an DP-Slaves oder eine Datenübertragung oder den Abbruch einer nichtprojektierten Verbindung angestoßen haben und Sie diese Anweisung erneut aufrufen, bevor der laufende Auftrag beendet wurde, hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt.

Übergabe von Parametern an asynchron auszuführende Programmbausteine

Codebausteine (FB/FC) und Datenbausteine (DB) können mit unterschiedlicher Zugriffsart erstellt werden ("Standard" und "Optimiert"). In den Codebausteinen können Sie beliebige Anweisungen aufrufen. Einige Anweisungen (z. B. "WRIT_DBL" und "READ_DBL") werden asynchron ausgeführt. Diese Programmbausteine können nicht mit Variablen aus TEMP versorgt werden, da die Daten während der Ausführung nicht geändert werden dürfen.

Stellen Sie sicher, dass Sie diese Anweisungen nicht in Programmen verwenden, in denen sich Codebausteine mit unterschiedlicher Zugriffsart gegenseitig aufrufen, so dass folgende Fälle eintreten würden:

- Eine Struktur aus einem Standard-Datenbaustein wird unmittelbar oder mittelbar an einen optimierten Codebaustein übergeben, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Bausteine weitergibt.
- Der umgekehrte Fall, dass eine Struktur aus einem optimierten Codebaustein unmittelbar oder mittelbar an einen Standard-Datenbaustein übergeben wird, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Programmbausteine weitergibt.

Andernfalls wird eine versteckte Kopie der übergebenen Daten in TEMP angelegt, die dann dazu führt, dass die asynchron arbeitenden Programmbausteine eine negative Quittung liefern.

Parameter REQ

Der Eingangsparameter REQ (request) dient ausschließlich dem Anstoß des Auftrags bzw. der Vorbereitung des Anstoßes:

- Einige Anweisungen wie z. B. RecipeExport und RecipeImport werden durch eine steigende Flanke an REQ aktiviert. Für diese Anweisungen ist ein vorbereitender Aufruf mit REQ = 0 notwendig, bevor Sie den Erstaufruf mit REQ = 1 durchführen.
- Stoßen Sie den Auftrag an, indem Sie den Eingangsparameter REQ gleich "1" setzen (Fall 1).
- Ist ein bestimmter Auftrag angestoßen und noch nicht abgeschlossen und Sie rufen die Anweisung zum gleichen Auftrag erneut auf (z. B. in einem Weckalarm-OB), so wird REQ durch die Anweisung nicht ausgewertet (Fall 2).
- Ist der Auftrag beendet, aber beim nächsten Aufruf der Eingangsparameter REQ immer noch auf "1" gesetzt, so wird der Auftrag sofort wieder gestartet.

Hinweis

REQ-Bit = "1"

Bitte beachten Sie, dass bei gesetztem REQ-Bit die asynchrone Anweisung nach der Beendigung des vorherigen Aufrufs wieder, möglicherweise ungewollt, erneut gestartet wird. Für die bessere Wartbarkeit / Übersichtlichkeit des Programms ist es damit ratsam, das REQ-Bit so früh wie möglich wieder auf "0" zurückzusetzen.

Parameter RET_VAL und BUSY

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand der Auftragsausführung angezeigt.

Beachten Sie den Hinweis in dem Abschnitt: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

- Beim Aufruf mit REQ = 0 und freien Systemressourcen wird in RET_VAL W#16#7000 eingetragen und BUSY erhält den Wert "0".

  Hinweis

  Aufruf mit REQ = 0 bei belegten Systemressourcen

  Falls bei einer S7-1500-CPU eine asynchrone Anweisung mit REQ = 0 aufgerufen wird und zu diesem Zeitpunkt alle verfügbaren Systemressourcen belegt sind und dieser Aufruf kein Zwischenaufruf eines bereits laufenden Auftrags ist, wird in RET_VAL der Fehlercode W#16#80C3 (temporärer Ressourcenfehler) eingetragen.

  Bei einer S7-300-CPU wird unter diesen Randbedingungen in RET_VAL der Fehlercode W#16#7000 (Erstaufruf mit REQ = 0) eingetragen.

- Im Fall 1 (Erstaufruf mit REQ = 1) wird bei freien Systemressourcen und korrekter Versorgung der Eingangsparameter in RET_VAL der Code W#16#7001 eingetragen und BUSY wird auf "1" gesetzt.
  Falls die benötigten Systemressourcen momentan belegt sind oder ein Fehler in den Eingangsparametern vorliegt, wird in RET_VAL der zugehörige Fehlercode eingetragen und BUSY wird mit "0" beschrieben.
- Im Fall 2 (Zwischenaufruf) wird der Code W#16#7002 in RET_VAL eingetragen (Dies entspricht der Nachricht, dass der Auftrag derzeit noch bearbeitet wird.) und BUSY wird auf "1" gesetzt.
- Beim letzten Aufruf für einen Auftrag gilt:
  - Bei der Anweisung "DPNRM_DG (Seite 2450)" wird bei fehlerfreier Datenübertragung in RET_VAL die Anzahl gelieferter Daten in Bytes als positive Zahl eingetragen. BUSY wird in diesem Fall mit "0" beschrieben.
    Im Fehlerfall wird in RET_VAL die Fehlerinformation eingetragen. BUSY dürfen Sie in diesem Fall nicht auswerten.
  - Bei allen anderen Anweisungen wird bei fehlerfreier Auftragsausführung in RET_VAL "0" eingetragen und BUSY wird mit "0" beschrieben. Im Fehlerfall wird in RET_VAL der Fehlercode eingetragen und BUSY wird mit "0" beschrieben.

Hinweis

Wenn der erste und der letzte Aufruf zusammenfallen, dann gilt für RET_VAL und BUSY das gleiche wie beim letzten Aufruf.

Überblick

Die folgende Tabelle gibt Ihnen einen Überblick über die oben beschriebenen Zusammenhänge. Sie zeigt insbesondere die möglichen Werte der Ausgangsparameter an, falls die Auftragsausführung nicht nach einem Anweisungs-Aufruf abgeschlossen ist.

Hinweis

Sie müssen in Ihrem Programm nach jedem Aufruf die relevanten Ausgangsparameter bewerten.

Zusammenhang zwischen Aufruf, REQ, RET_VAL und BUSY bei einem "laufenden" Auftrag:

Lfd. Nr. des Aufrufs	Aufrufart	REQ	RET_VAL	BUSY	DONE	ERROR	ENO
0	vorbereitender Aufruf	0	W#16#7000	0	0	0	True
1	erster Aufruf	1	W#16#7001	1	0	0	True
			Fehlercode	0	0	1	False
2 bis (n - 1)	Zwischenaufruf	irrelevant	W#16#7002	1	0	0	True
n	letzter Aufruf	irrelevant	W#16#0000 (Ausnahmen: RD_REC (Seite 2428) und RD_DPARA (Seite 2528): die erfolgreich gelesene Datenlänge wird in den Parameter STATUS geschrieben.)	0	1	0	True
			Fehlercode, falls Fehler aufgetreten sind.	0	0	1	False

4.1.3 Einfache Anweisungen

4.1.3.1 KOP

Bitverknüpfungen

---| |---: Schließerkontakt

Beschreibung

Die Aktivierung eines Schließerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Schließerkontakt geschlossen und der Signalzustand des Eingangs auf den Ausgang übertragen.

Wenn der Operand den Signalzustand "0" hat, wird der Schließerkontakt nicht aktiviert und der Signalzustand am Ausgang der Anweisung auf "0" gesetzt.

In einer Reihenschaltung werden zwei oder mehrere Schließerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Schließerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung.

Legen Sie dazu einen globalen Datenbaustein mit folgendem Inhalt an:

Name des Bausteins: SLI_gDB_NOContact	
Name	Datentyp
start1	BOOL
start2	BOOL
start3	BOOL
startOut	BOOL

Schreiben Sie folgenden Programmcode:

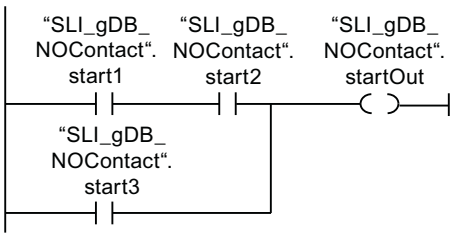

Der Operand "SLI_gDB_NOContact".startOut wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "SLI_gDB_NOContact".start1 und "SLI_gDB_NOContact".start2 haben den Signalzustand "1".
- Der Operand "SLI_gDB_NOContact".start3 hat den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---| / |---: Öffnerkontakt

Beschreibung

Die Aktivierung eines Öffnerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Öffnerkontakt geöffnet und der Signalzustand am Ausgang der Anweisung auf "0" zurückgesetzt.

Wenn der Operand den Signalzustand "0" hat, wird der Öffnerkontakt nicht aktiviert und der Signalzustand des Eingangs auf den Ausgang übertragen.

In einer Reihenschaltung werden zwei oder mehrere Öffnerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Öffnerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

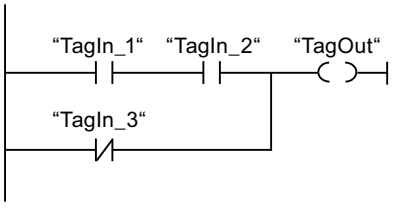

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" haben den Signalzustand "1".
- Der Operand "TagIn_3" hat den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

--|NOT|--: VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE). Wenn am Eingang der Anweisung der Signalzustand "1" ansteht, liefert der Ausgang der Anweisung den Signalzustand "0". Ist der Signalzustand am Eingang der Anweisung "0", liefert der Ausgang den Signalzustand "1".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

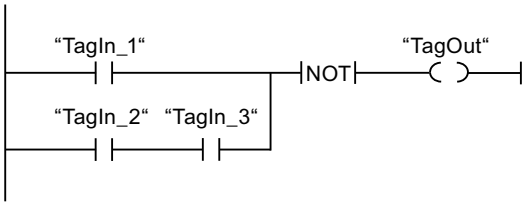

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Signalzustand an der Operanden "TagIn_2" und "TagIn_3" ist "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---( )---: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" setzen Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Eingang der Spule "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Eingang der Spule wird direkt auf den Ausgang übertragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

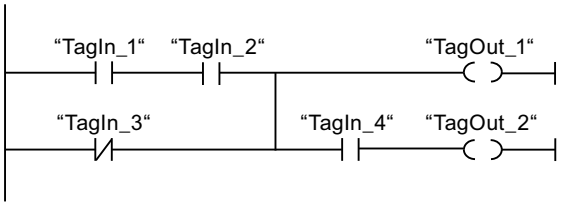

Der Operand "TagOut_1" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Der Operand "TagOut_2" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" sowie der Operand "TagIn_4" liefern den Signalzustand "1"
- Der Signalzustand des Operanden "TagIn_3" ist "0" und der des Operanden "TagIn_4" ist "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

--(/)--: Zuweisung negieren

Beschreibung

Mit der Anweisung "Zuweisung negieren" kehren Sie das Verknüpfungsergebnis (VKE) um und weisen es dem angegebenen Operanden zu. Wenn das VKE am Eingang der Spule "1" ist, wird der Operand zurückgesetzt. Wenn das VKE am Eingang der Spule "0" ist, wird der Operand auf den Signalzustand "1" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung negieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das negierte VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

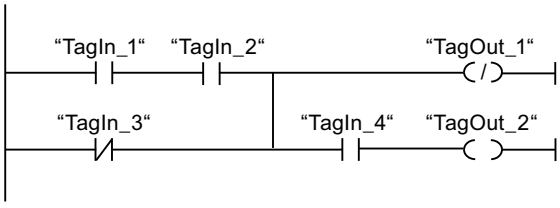

Der Operand "TagOut_1" wird zurücksetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---(R)---: Ausgang rücksetzen

Beschreibung

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "0" zurückgesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfluss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Output	BOOL	E, A, M, D, L	E, A, M, D, L, T, Z	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

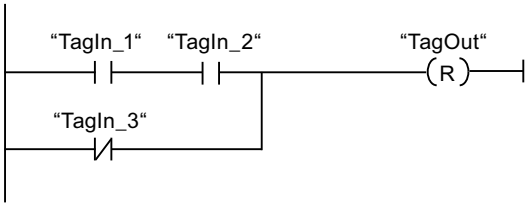

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---(S)---: Ausgang setzen

Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfluss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

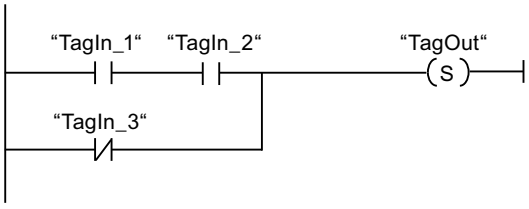

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

SET_BF: Bitfeld setzen

Beschreibung

Mit der Anweisung "Bitfeld setzen" setzen Sie mehrere Bits ab einer bestimmten Adresse.

Die Anzahl der zu setzenden Bits bestimmen Sie durch den Wert von <Operand1>. Die Adresse des ersten zu setzenden Bits wird durch <Operand2> festgelegt. Der Wert von <Operand1> darf nicht größer als die Anzahl der Bits in einem selektierten Byte sein. Ist dieser Wert größer, wird die Anweisung nicht ausgeführt und die Fehlermeldung "Bereichsüberschreitung für Index <Operand1>" wird angezeigt. Die Bits bleiben so lange gesetzt, bis sie explizit durch eine andere Anweisung zurückgesetzt werden.

<Operand1> geben Sie am Operandenplatzhalter unterhalb der Anweisung an. <Operand2> geben Sie am Operandenplatzhalter oberhalb der Anweisung an.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Bei einem VKE von "0" am Eingang der Spule wird die Anweisung nicht ausgeführt.

Bitfelder vom Datentyp PLC-Datentyp, STRUCT oder ARRAY

Bei Strukturen vom Datentyp PLC-Datentyp, STRUCT oder ARRAY werden maximal so viele Bits gesetzt, wie die Struktur Bits enthält:

- Wenn Sie z. B. am <Operand1> den Wert "20" angeben und die Struktur nur 10 Bits enthält, dann werden auch nur diese 10 Bits gesetzt.
- Wenn Sie z. B. am <Operand1> den Wert "5" angeben und die Struktur 10 Bits enthält, dann werden genau 5 Bits gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	UINT	Konstante	Anzahl der zu setzenden Bits
<Operand2>	Output	BOOL	E, A, M Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL	Zeiger auf das erste Bit, das gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

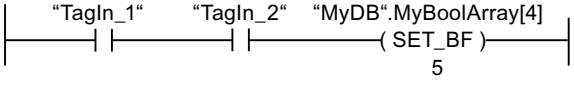

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

RESET_BF: Bitfeld rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld rücksetzen" setzen Sie mehrere Bits ab einer bestimmten Adresse zurück.

Die Anzahl der zurückzusetzenden Bits bestimmen Sie durch den Wert von <Operand1>. Die Adresse des ersten zurückzusetzenden Bits wird durch <Operand2> festgelegt. Der Wert von <Operand 1> darf nicht größer als die Anzahl der Bits in einem selektierten Byte sein. Ist dieser Wert größer, wird die Anweisung nicht ausgeführt und die Fehlermeldung "Bereichsüberschreitung für Index <Operand 1>" wird angezeigt. Die Bits bleiben so lange zurückgesetzt, bis sie explizit durch eine andere Anweisung gesetzt werden.

<Operand1> geben Sie am Operandenplatzhalter unterhalb der Anweisung an. <Operand2> geben Sie am Operandenplatzhalter oberhalb der Anweisung an.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Bei einem VKE von "0" am Eingang der Spule wird die Anweisung nicht ausgeführt.

Bitfelder vom Datentyp PLC-Datentyp, STRUCT oder ARRAY

Bei Strukturen vom Datentyp PLC-Datentyp, STRUCT oder ARRAY werden maximal so viele Bits zurückgesetzt, wie die Struktur Bits enthält:

- Wenn Sie z. B. am <Operand1> den Wert "20" angeben und die Struktur nur 10 Bits enthält, dann werden auch nur diese 10 Bits zurückgesetzt.
- Wenn Sie z. B. am <Operand1> den Wert "5" angeben und die Struktur 10 Bits enthält, dann werden genau 5 Bits zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	UINT	Konstante	Anzahl der zurückzusetzenden Bits
<Operand2>	Output	BOOL	E, A, M Bei einem DB oder einem Instanz-DB ein Element eines ARRAY[..] of BOOL	Zeiger auf das erste Bit, das zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

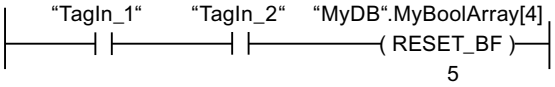

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, werden 5 Bit ab der Adresse des Operanden "MyDB".MyBoolArray[4] zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

SR: Flipflop setzen/rücksetzen

Beschreibung

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
S	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Setzen freigeben
R1	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Rücksetzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Operand, der gesetzt oder zurückgesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

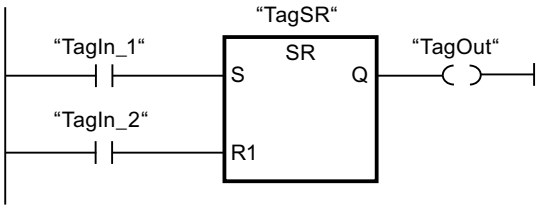

Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagSR " und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

RS: Flipflop rücksetzen/setzen

Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen, R und S1, wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
R	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Rücksetzen freigeben
S1	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Setzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Operand, der zurückgesetzt oder gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

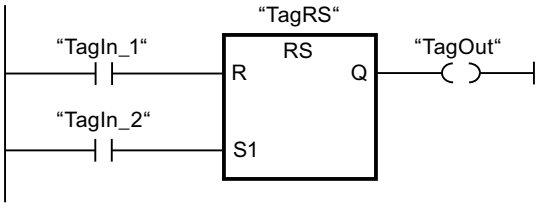

Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

--|P|--: Operand auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

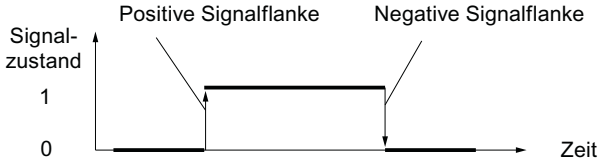

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

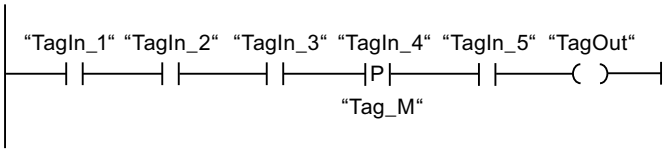

Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" liefern den Signalzustand "1".
- Am Operanden "TagIn_4" liegt eine steigende Flanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag_M" gespeichert.
- Der Signalzustand des Operanden "TagIn_5" ist "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

--|N|--: Operand auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker <Operand2> gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

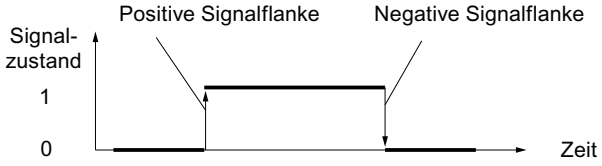

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie im Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie im Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

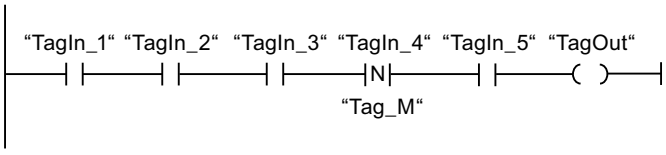

Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" liefern den Signalzustand "1".
- Am Operanden "TagIn_4" liegt eine negative Signalflanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag_M" gespeichert.
- Der Signalzustand des Operanden "TagIn_5" ist "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

--(P)--: Operand bei positiver Signalflanke setzen

Beschreibung

Mit der Anweisung "Operand bei positiver Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei positiver Signalflanke setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Output	BOOL	E, A, M, D, L	Operand, der bei einer positiven Flanke gesetzt wird.
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

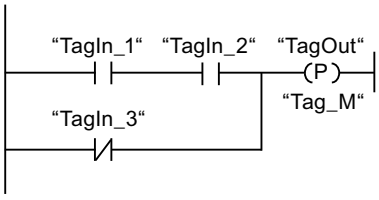

Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Spule von "0" auf "1" wechselt (positive Signalflanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

--(N)--: Operand bei negativer Signalflanke setzen

Beschreibung

Mit der Anweisung "Operand bei negativer Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei negativer Signalflanke setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Output	BOOL	E, A, M, D, L	Operand, der bei einer negativen Flanke gesetzt wird.
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

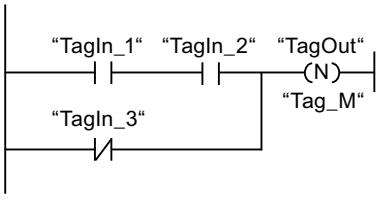

Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Spule von "1" auf "0" wechselt (negative Signalflanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

P_TRIG: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine positive Signalflanke erfasst wird. In allen anderen Fällen liefert der Ausgang den Signalzustand "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

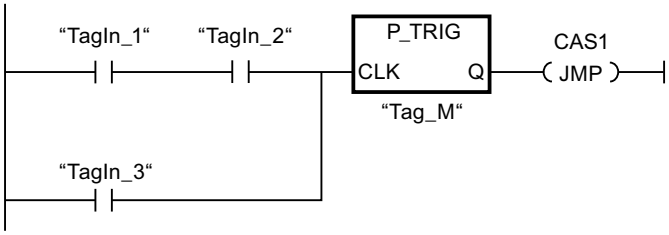

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

N_TRIG: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den

Signalzustand "1", sobald eine negative Signalflanke erfasst wird. In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

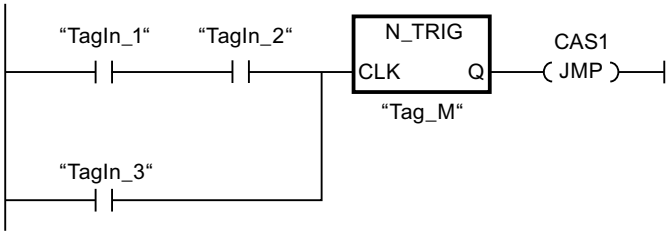

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

R_TRIG: Positive Signalflanke erkennen

Beschreibung

Mit der Anweisung "Positive Signalflanke erkennen" können Sie eine Zustandsänderung von "0" auf "1" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "0" auf "1" erkennt, wird am Ausgang Q eine positive Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Positive Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

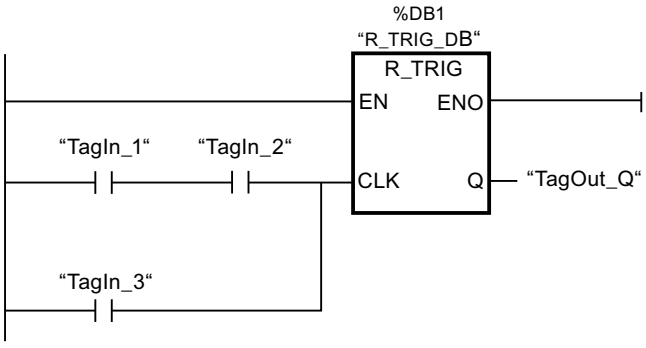

In der Variablen "R_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3" eine Änderung im Signalzustand von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

F_TRIG: Negative Signalflanke erkennen

Beschreibung

Mit der Anweisung "Negative Signalflanke erkennen" können Sie eine Zustandsänderung von "1" auf "0" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "1" auf "0" erkennt, wird am Ausgang Q eine negative Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negative Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

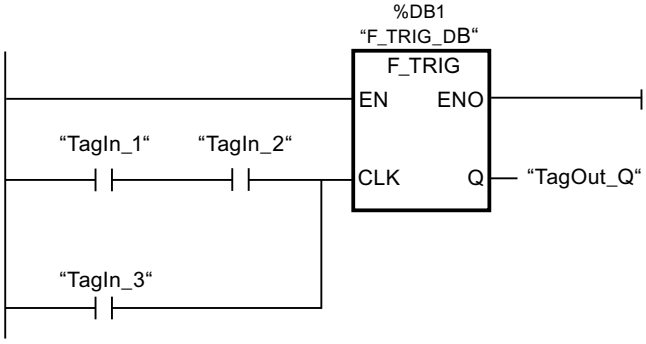

In der Variablen "F_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3"

eine Änderung im Signalzustand von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TP_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Impuls erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Impulsausgang
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

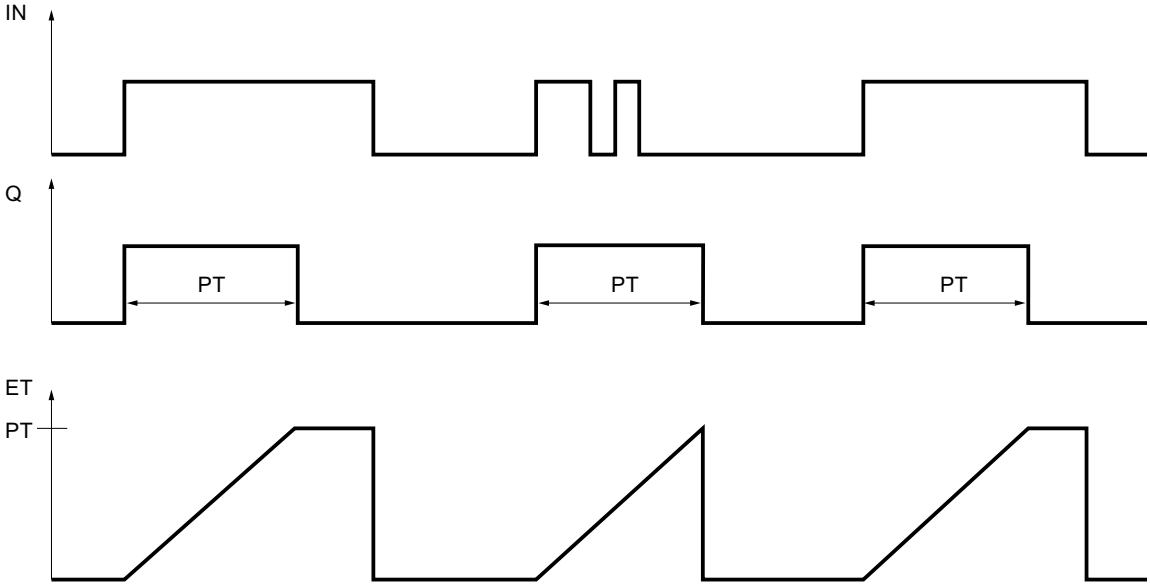

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

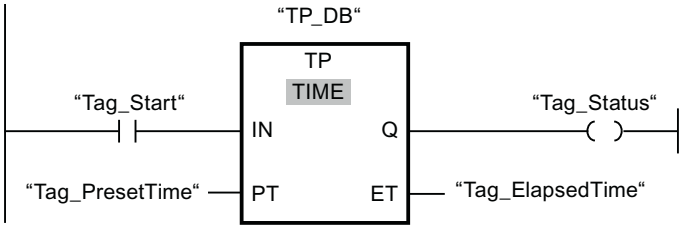

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	TRUE
ET	Tag_ElapsedTime	von T#0s => T#10s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert. Wenn die Zeit abgelaufen ist, dann wird der Operand "Tag_Status" wieder auf den Signalzustand "0" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7490)

Grundlagen zu KOP (Seite 7425)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TON_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME, TON_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Einschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

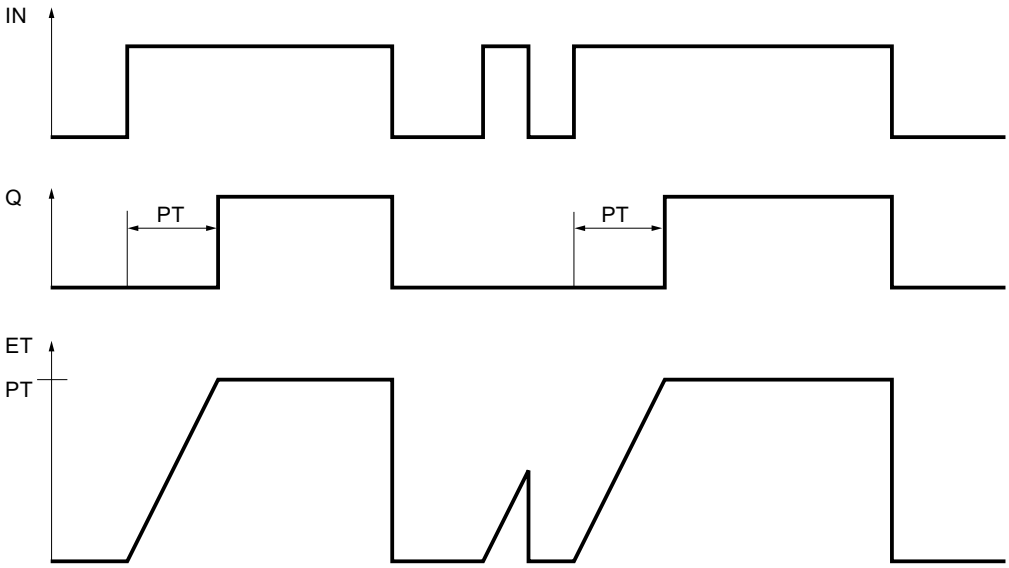

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

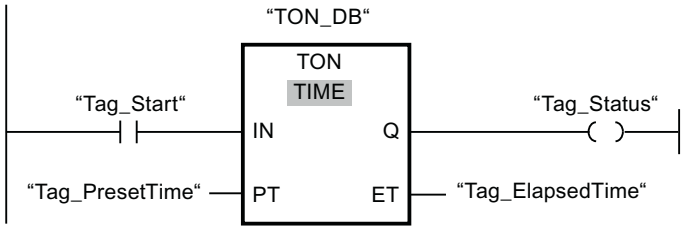

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	FALSE; nach 10s => TRUE
ET	Tag_ElapsedTime	von T#0s => T#10s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert. Wenn der Signalzustand am Operand "Tag_Start" von "1" auf "0" wechselt, wird der Operand "Tag_Status" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TOF_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME, TOF_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Ausschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

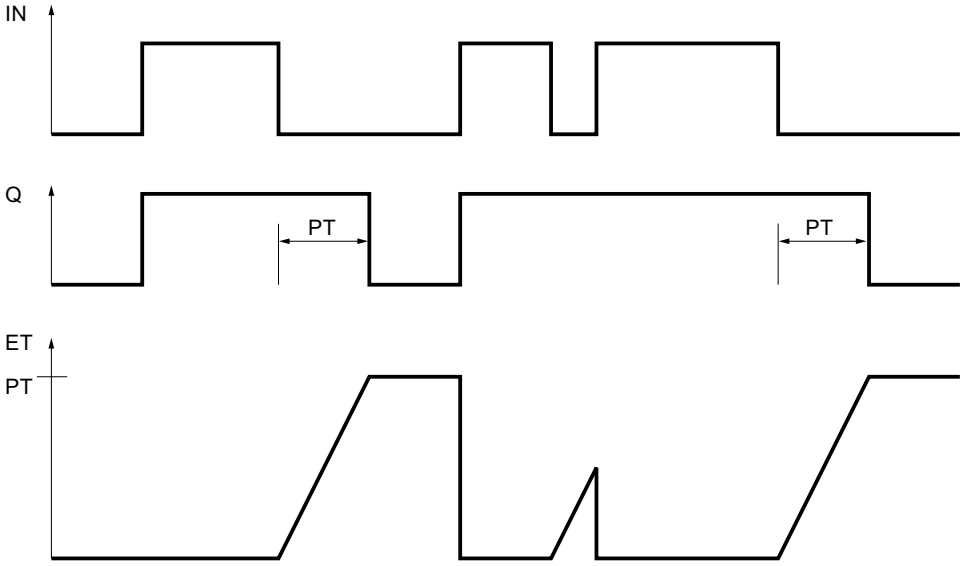

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

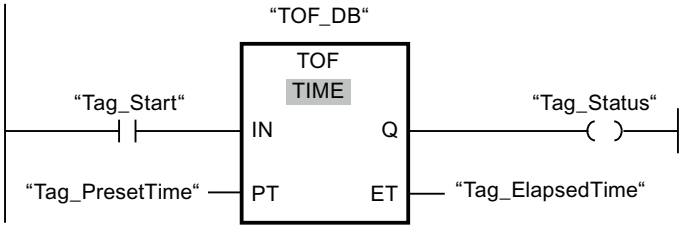

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"; Signalwechsel "1" => "0"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	TRUE
ET	Tag_ElapsedTime	von T#10s => T#0s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" auf TRUE gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf FALSE zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Eingang IN von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt (negative Signalflanke).

Der Eingang R setzt die Ausgänge ET und Q, unabhängig vom Signalzustand am Starteingang, zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TONR_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME, TONR_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Starteingang
R	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Akkumulierte Zeit

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":

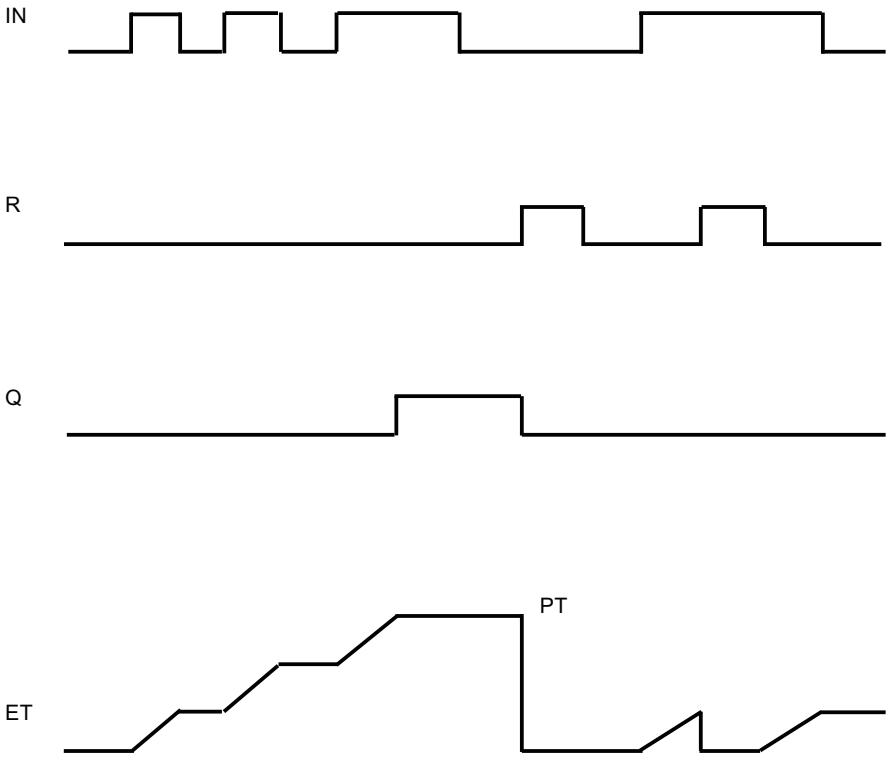

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

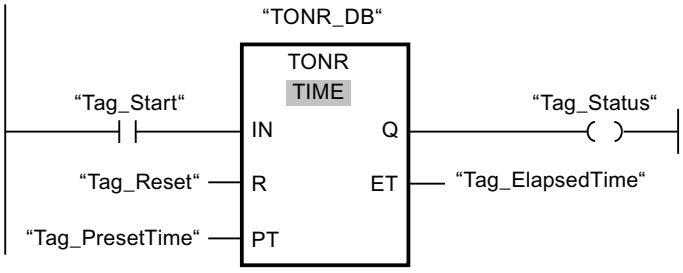

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s

Parameter	Operand	Wert
Q	Tag_Status	FALSE; nach 10s => TRUE
ET	Tag_ElapsedTime	Signalwechsel von "0" => "1": Die Zeit T#10s läuft ab. Nach 5s Signalwechsel von "1" => "0": Die Zeit am Operanden "Tag_ElapsedTime" bleibt bei T#5s stehen. Nach ca. 2s erneuter Signalwechsel von "0" => "1": Die Zeit am Operanden "Tag_ElapsedTime" läuft bei T#5s weiter ab.

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange der Operand "Tag_Start" den Signalzustand "1" hat, läuft die Zeit ab. Wenn der Signalzustand am Operanden "Tag_Start" von "1" auf "0" wechselt, bleibt die Zeit stehen und der aktuelle Zeitwert am Operanden "Tag_ElapsedTime" wird gemerkt. Wechselt der Signalzustand am Operanden "Tag_Start" wieder von "0" auf "1", dann läuft die Zeit ab dem Zeitwert weiter, der beim Wechsel von "1" auf "0" gemerkt worden war. Wenn der am Parameter PT angegebene Zeitwert erreicht ist, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(TP)---: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Impuls. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab, unabhängig vom weiteren Verlauf des VKE. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Ablauf der IEC-Zeit nicht. Solange die IEC-Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn die IEC-Zeit abgelaufen ist, liefert der Zeitstatus den Signalzustand "0".

Am <Operand1> (Zeitdauer) geben Sie unterhalb der Anweisung die Zeitdauer des Impulses an und am <Operand2> (IEC-Zeit) oberhalb der Anweisung geben Sie die IEC-Zeit an, die gestartet wird.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TP_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_LTIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente Q der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen.

Die Ausführung der Anweisung "Zeit als Impuls starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TP_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

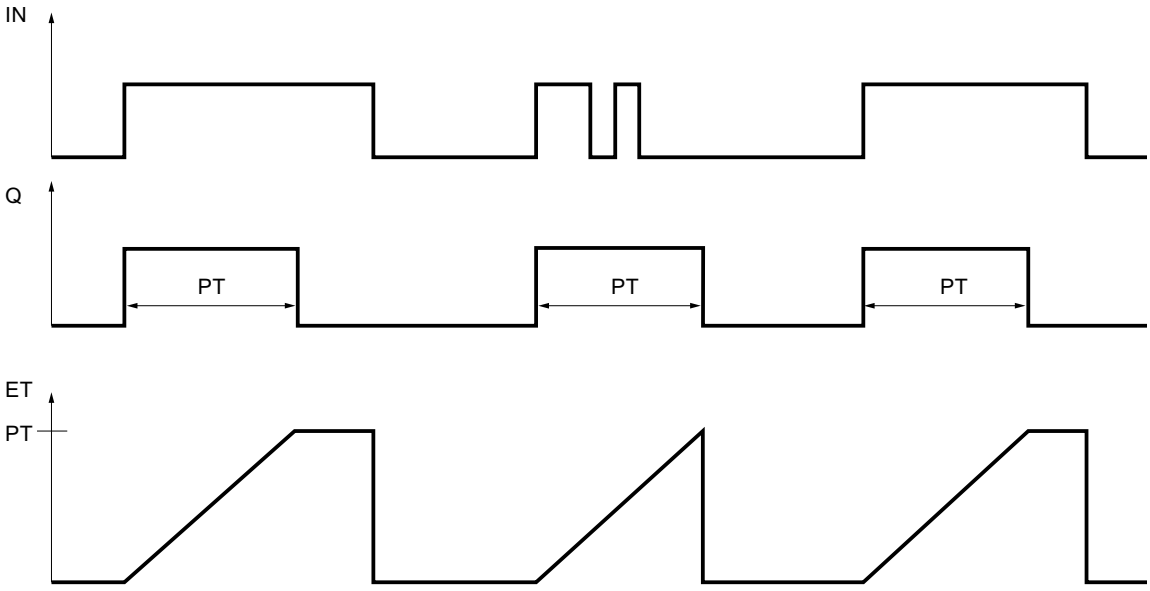

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

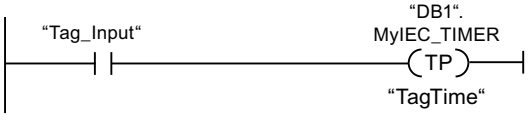

Die Anweisung "Zeit als Impuls starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "0" auf "1" wechselt. Die Zeit "DB1".MyIEC_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.

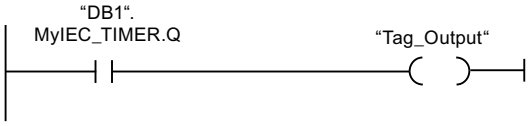

Solange die Zeit "DB1".MyIEC_TIMER läuft, liefert der Zeitstatus ("DB1".MyIEC_TIMER.Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn die IEC-Zeit abgelaufen ist, wechselt der Signalzustand des Zeitstatus auf "0" und der Operand "Tag_Output" wird zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(TON)---: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Einschaltverzögerung. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab. Der Ausgang liefert den Signalzustand "1", wenn das VKE am Eingang der Anweisung den Signalzustand "1" führt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die laufende IEC-Zeit zurückgesetzt. Die Abfrage des Zeitstatus auf "1" liefert in diesem Fall den Signalzustand "0". Mit der Erfassung der nächsten positiven Signalflanke am Eingang der Anweisung startet die IEC-Zeit wieder.

Am <Operand1> (Zeitdauer) geben Sie unterhalb der Anweisung die Zeitdauer der Einschaltverzögerung an und am <Operand2> (IEC-Zeit) oberhalb der Anweisung geben Sie die IEC-Zeit an, die gestartet wird.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TON_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME, TON_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen.

Die Ausführung der Anweisung "Zeit als Einschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TON_TIME	IEC_TIMER, IEC_LTIMER, TON_TIME, TON_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

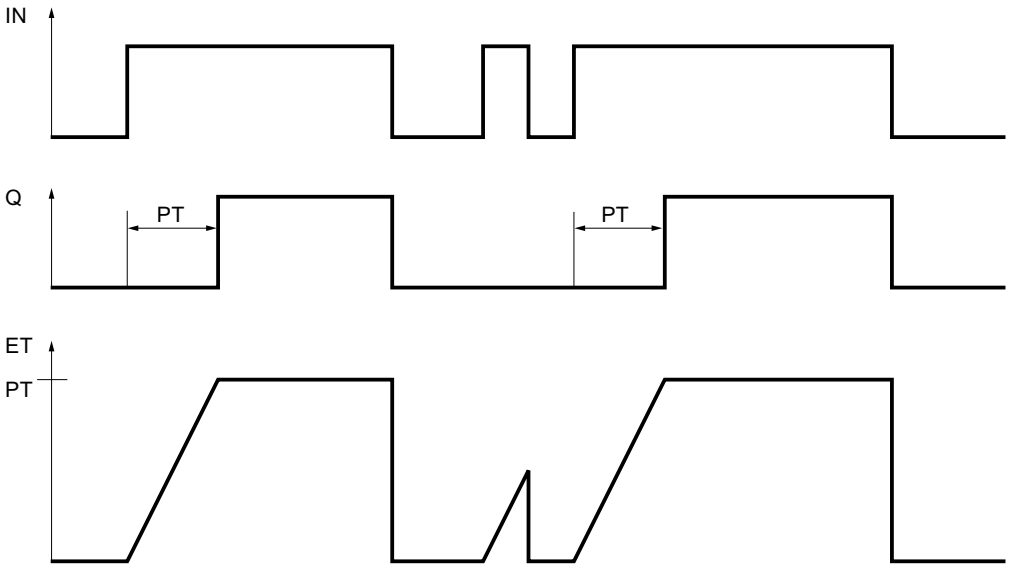

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

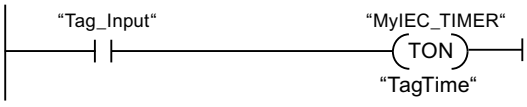

Die Anweisung "Zeit als Einschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "0" auf "1" wechselt. Die Zeit "MyIEC_TIMER" wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.

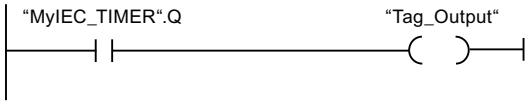

Wenn die Zeit "MyIEC_TIMER" abgelaufen ist und der Operand "Tag_Input" den Signalzustand "1" führt, liefert die Abfrage des Zeitstatus ("MyIEC_TIMER".Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn der Signalzustand des Operanden "Tag_Input" auf "0" wechselt, liefert die Abfrage des Zeitstatus den Signalzustand "0" und der Operand "Tag_Output" wird zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(TOF)---: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Ausschaltverzögerung. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "0" führt. Wenn das VKE von "1" auf "0" (negative Signalflanke) wechselt, wird die IEC-Zeit mit der angegebenen Zeitdauer gestartet. Solange die IEC-Zeit läuft, bleibt der Zeitstatus auf den Signalzustand "1". Wenn die Zeit abgelaufen ist und das VKE am Eingang der Anweisung den Signalzustand "0" führt, wird der Zeitstatus auf den Signalzustand "0" gesetzt. Wechselt das VKE vor dem Ablauf der Zeit auf "1", wird die laufende IEC-Zeit zurückgesetzt und der Zeitstatus bleibt auf dem Signalzustand "1".

Am <Operand1> (Zeitdauer) geben Sie unterhalb der Anweisung die Zeitdauer der Ausschaltverzögerung an und am <Operand2> (IEC-Zeit) oberhalb der Anweisung geben Sie die IEC-Zeit an, die gestartet wird.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TOF_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME, TOF_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen.

Die Ausführung der Anweisung "Zeit als Ausschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TOF_TIME	IEC_TIMER, IEC_LTIMER, TOF_TIME, TOF_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

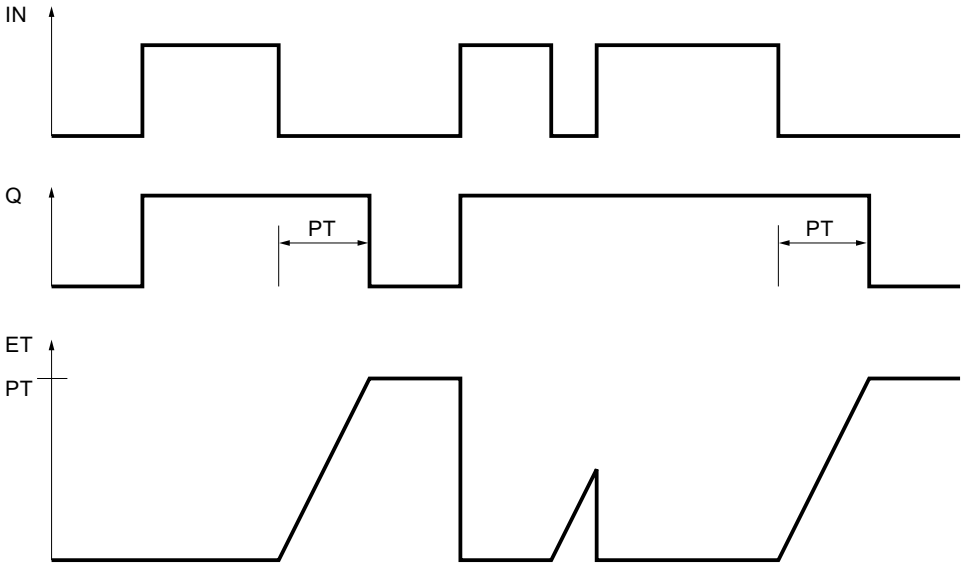

Beispiel

Solange die Zeit #MyIEC_TIMER läuft, liefert die Abfrage des Zeitstatus (#MyIEC_TIMER.Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn die Zeit abgelaufen ist und der Operand "Tag_Input" den Signalzustand "0" führt, liefert die Abfrage des Zeitstatus den Signalzustand "0". Wechselt der Signalzustand des Operanden "Tag_Input" vor dem Ablauf der Zeit #MyIEC_TIMER auf "1", wird die Zeit zurückgesetzt. Beim Signalzustand "1" des Operanden "Tag_Input" liefert die Abfrage des Zeitstatus den Signalzustand "1".

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

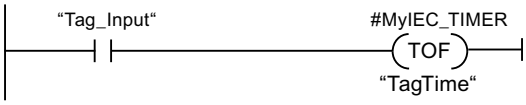

Die Anweisung "Zeit als Ausschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "1" auf "0" wechselt. Die Zeit #MyIEC_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.

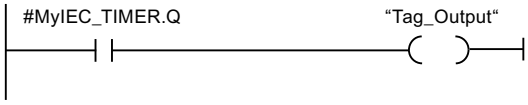

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(TONR)---: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" erfassen Sie, wie lange das Signal am Eingang der Anweisung "1" ist. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Solange das VKE "1" ist, wird die Zeit erfasst. Wechselt das VKE auf "0", wird die Anweisung angehalten. Wechselt das VKE erneut auf "1", wird die Zeiterfassung fortgesetzt. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn die erfasste Zeit den Wert der angegebenen Zeitdauer überschreitet und das VKE am Eingang der Spule "1" führt.

Der Zeitstatus und die aktuell abgelaufene Zeit können durch die Anweisung "Zeit rücksetzen" auf "0" zurückgesetzt werden.

Am <Operand1> (Zeitdauer) geben Sie unterhalb der Anweisung die Zeitdauer an und am <Operand2> (IEC-Zeit) oberhalb der Anweisung geben Sie die IEC-Zeit an, die gestartet wird.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Für CPU S7-1200

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TONR_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME, TONR_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TONR_TIME	IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

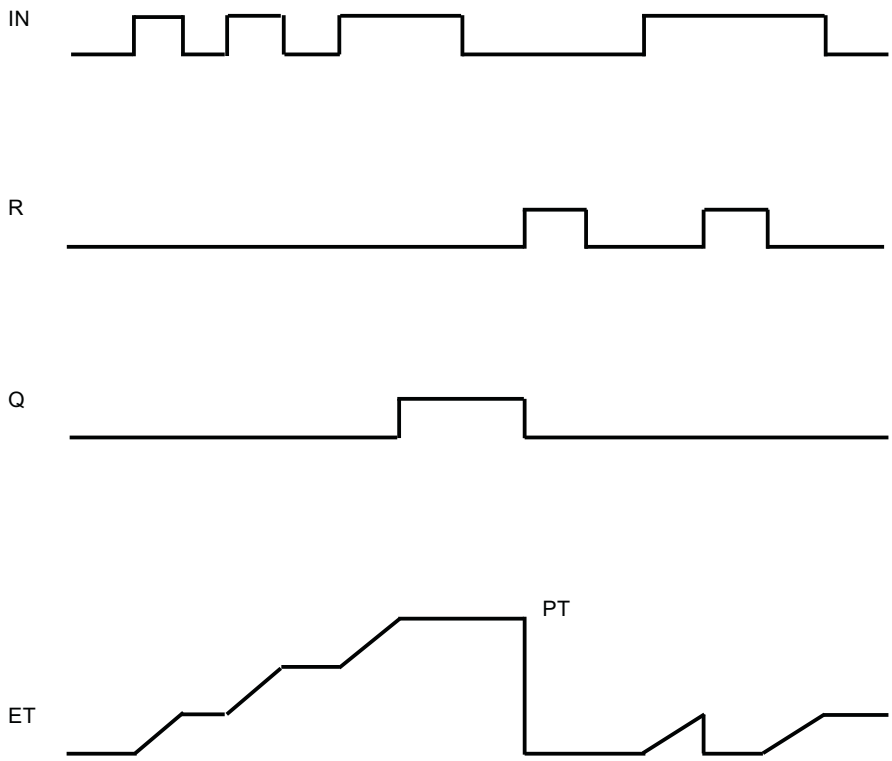

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

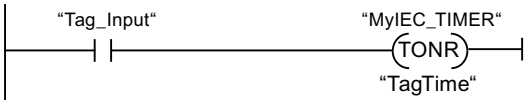

Die Anweisung "Zeit akkumulieren" wird bei einer positiven Signalflanke im VKE ausgeführt. Solange der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Zeit erfasst.

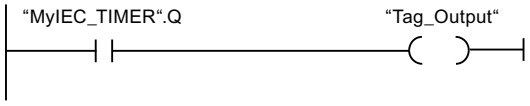

Wenn die erfasste Zeit den Wert des Operanden "TagTime" überschreitet, liefert die Abfrage des Zeitstatus ("MyIEC_TIMER".Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(RT)---: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom in die Spule (VKE ist "1"), werden die Strukturkomponenten der Zeit im angegebenen Datenbaustein auf "0" zurückgesetzt. Wenn das VKE am Eingang der Anweisung "0" ist, bleibt die Zeit unverändert.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Eingang der Spule wird direkt auf den Ausgang der Spule übertragen.

Der Anweisung "Zeit rücksetzen" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
<IEC-Zeit>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die zurückgesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

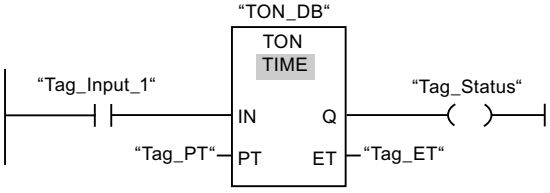

Wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PT" vorgegeben ist.

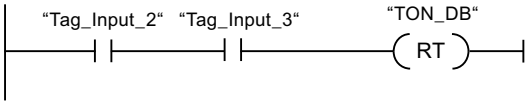

Wenn die Operanden "Tag_Input_2" und "Tag_Input_3" den Signalzustand "1" liefern, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Datenbaustein "TON_DB" abgelegte Zeit zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(PT)---: Zeitdauer laden

Beschreibung

Mit der Anweisung "Zeitdauer laden" stellen Sie die Zeitdauer einer IEC-Zeit ein. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegeben IEC-Zeit.

Am <Operand1> (Zeitdauer) geben Sie unterhalb der Anweisung die Zeitdauer an, die geladen wird, und am <Operand2> (IEC-Zeit) oberhalb der Anweisung geben Sie die IEC-Zeit an, die gestartet wird.

Hinweis

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC_TIMER-Struktur.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, deren Dauer eingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

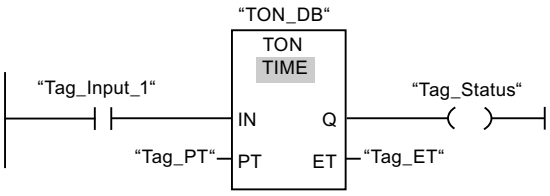

Wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PT" vorgegeben ist.

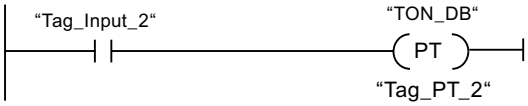

Wenn der Operand "Tag_Input_2" den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer "Tag_PT_2" in den Instanz-Datenbaustein "TON_DB" und überschreibt dabei den Wert des Operanden "Tag_PT" innerhalb des Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim Zugriff auf "MyTimer".Q oder "MyTimer".ET ändern.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Legacy

S_IMPULS: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird,

gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

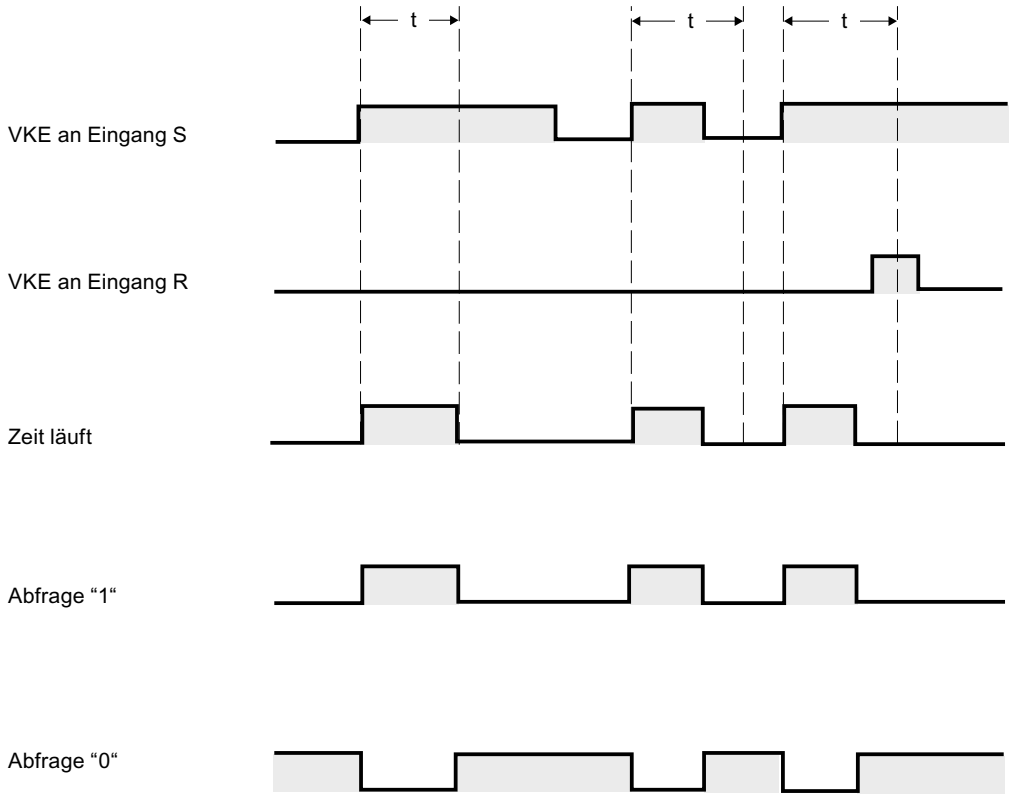

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

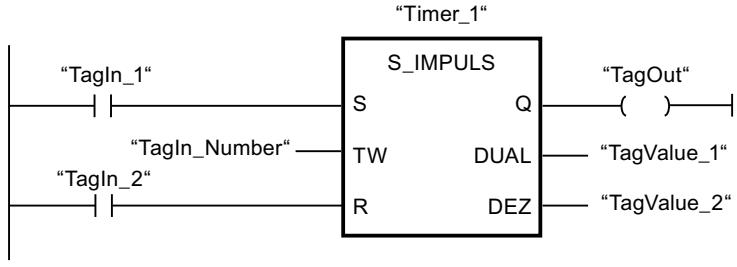

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Operand "TagIn_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

S_VIMP: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

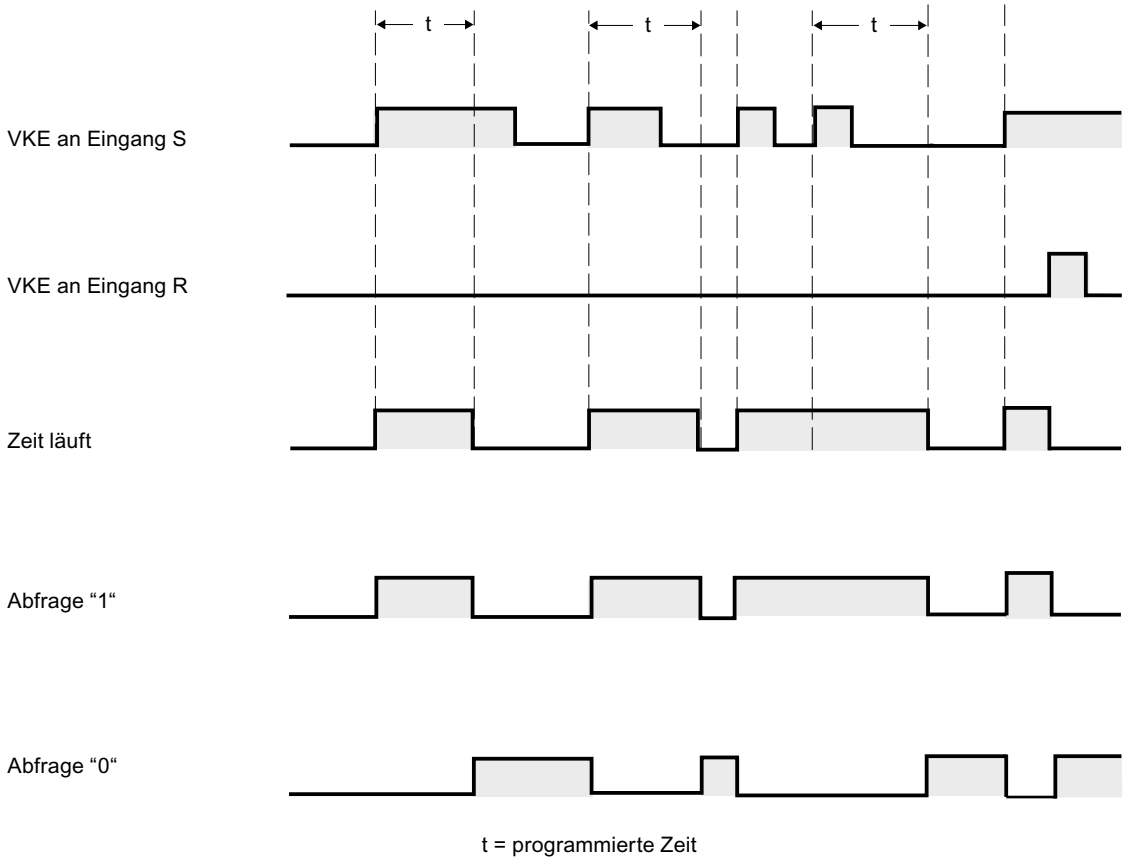

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

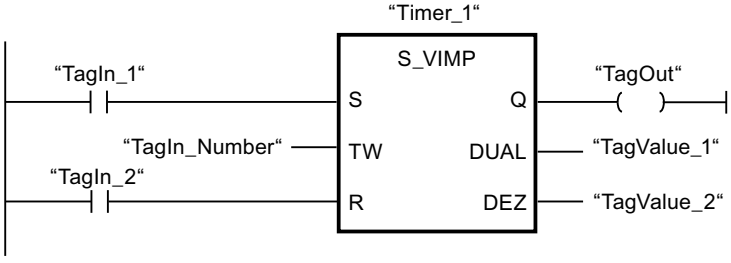

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Zeit der Anweisung geben Sie in den Platzhalter oberhalb der Box an. Die Zeit muss mit dem Datentyp TIMER deklariert sein.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

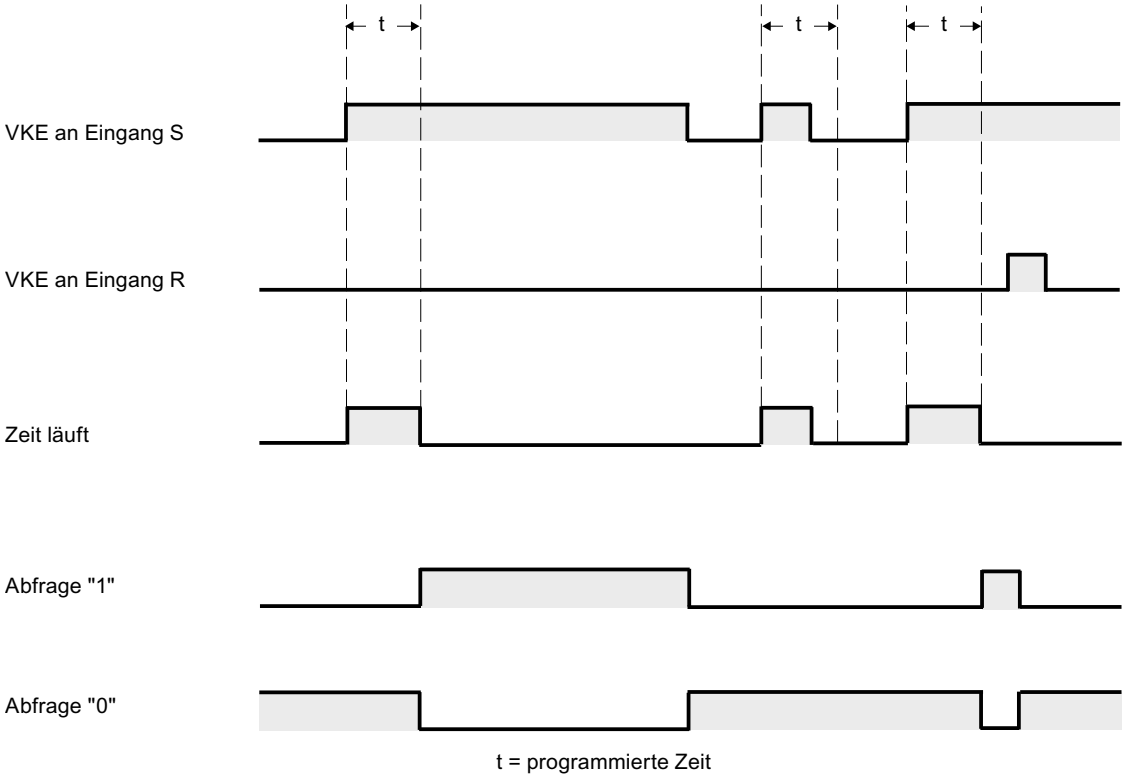

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

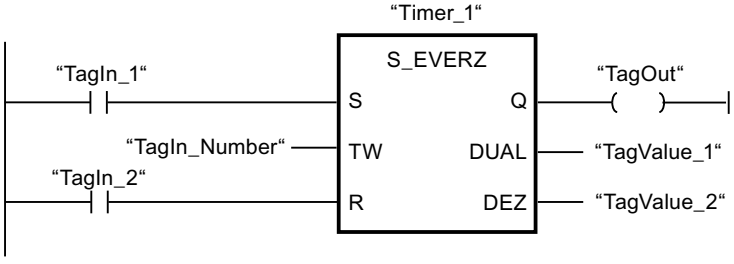

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

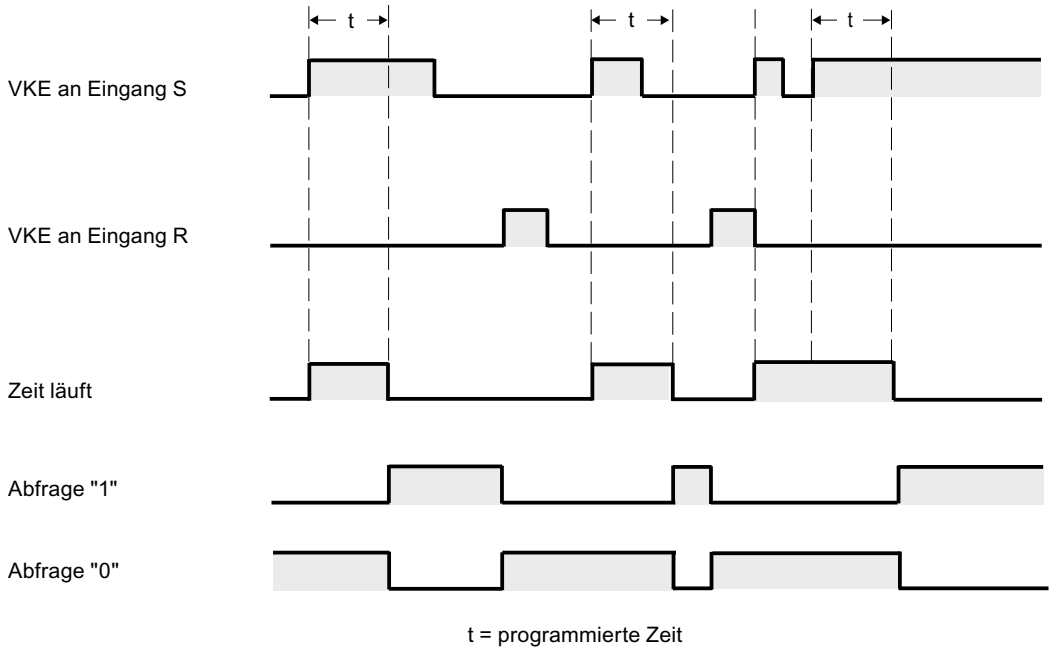

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

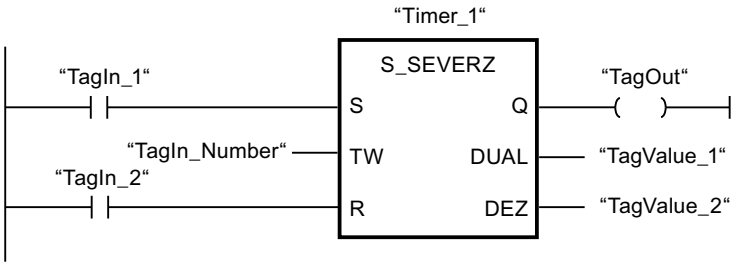

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, auch wenn der Signalzustand des Operanden "TagIn_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand am Eingang S "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

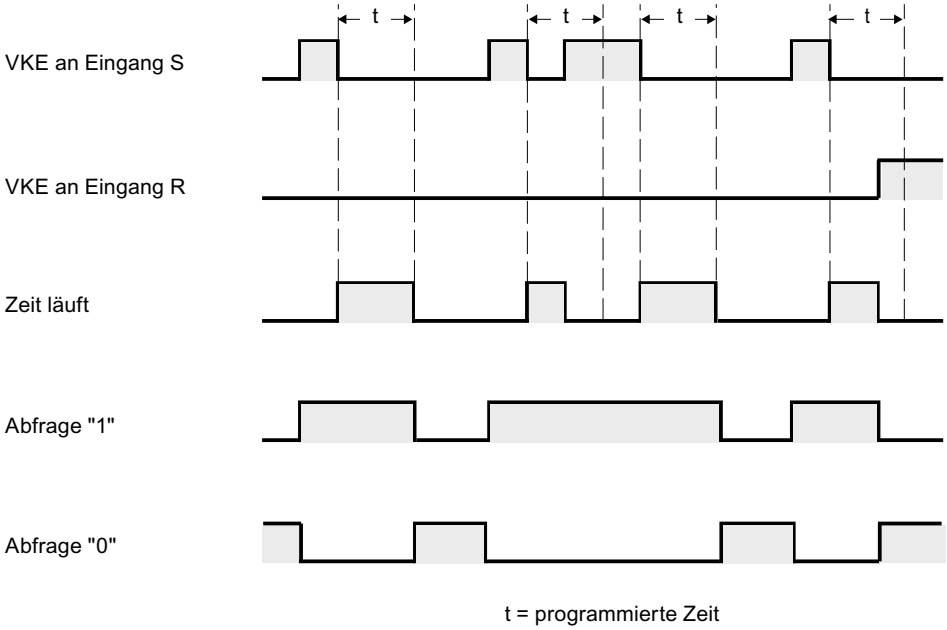

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

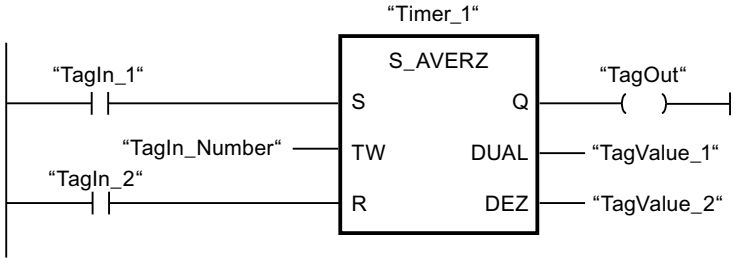

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

---(SI): Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie die Zeitdauer an mit der die Zeit ablaufen wird und in <Operand2> oberhalb der Anweisung geben Sie die Zeit an, die gestartet wird.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Operand2>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

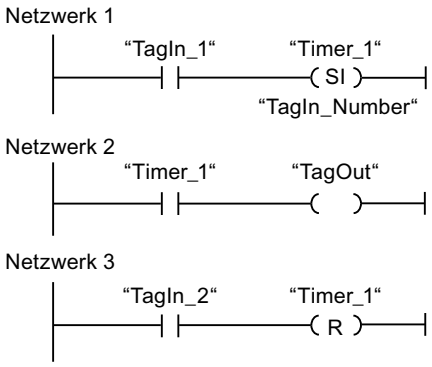

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Signalzustand des Operanden "TagIn_1" "1" ist. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_2" von "0" auf "1", dann wird die Zeit zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

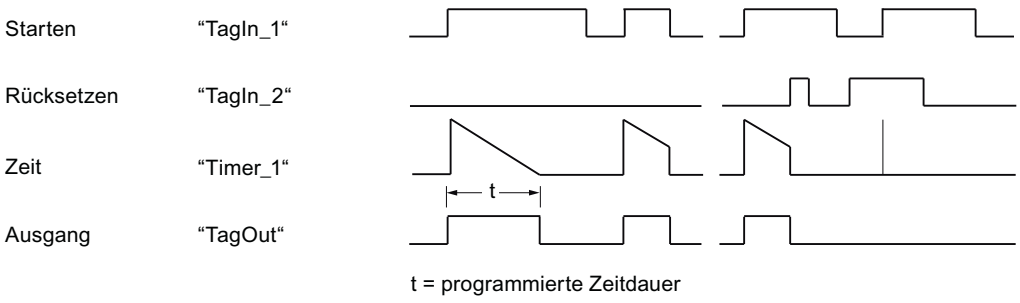

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

---(SV): Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" startet eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer neu gestartet. Die Abfrage des Zeitstatus auf "1" ergibt den Signalzustand "0", wenn die Zeit abgelaufen ist.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie die Zeitdauer an mit der die Zeit ablaufen wird und in <Operand2> oberhalb der Anweisung geben Sie die Zeit an, die gestartet wird.

Die Anweisung "Zeit als verlängerten Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Operand2>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

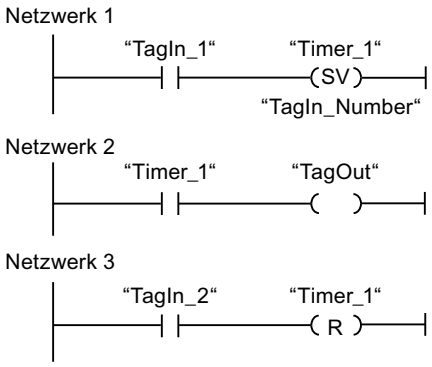

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer negativen Signalflanke im VKE beeinflusst zu werden. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_1" erneut von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

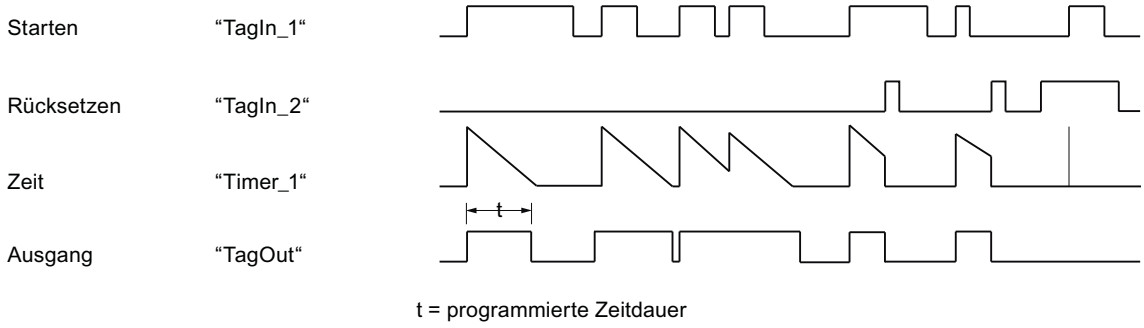

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SE): Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der angegebenen Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie die Zeitdauer an mit der die Zeit ablaufen wird und in <Operand2> oberhalb der Anweisung geben Sie die Zeit an, die gestartet wird.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Operand2>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

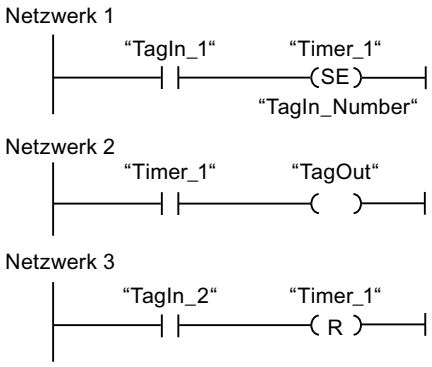

Netzwerk 1:

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab. Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit zurückgesetzt.

Netzwerk 2:

Wenn die Zeit abgelaufen ist, der Operand #TagIn_1 am Starteingang den Signalzustand "1" hat und die Zeit nicht zurückgesetzt wurde, dann ist der Operand #TagOut "1".

Netzwerk 3:

Wenn der Signalzustand des Operanden #TagIn_2 den Signalzustand "1" hat, dann wird die Zeit #Timer_1 und der Ausgang #TagOut zurückgesetzt.

Wenn Sie den #Timer_1 erneut starten wollen, dann muss der Signalzustand am Operand #TagIn_2 "0" sein und der Signalzustand am Starteingang #TagIn_1 von "0" auf "1" wechseln.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

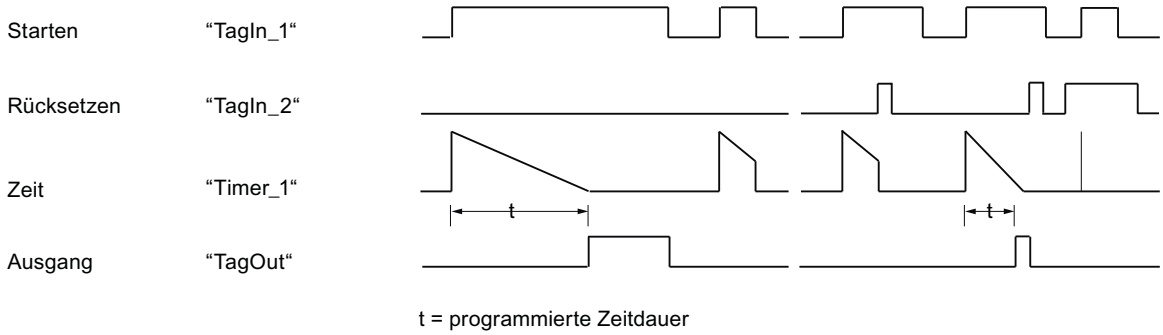

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SS): Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie die Zeitdauer an mit der die Zeit ablaufen wird und in <Operand2> oberhalb der Anweisung geben Sie die Zeit an, die gestartet wird.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Operand2>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

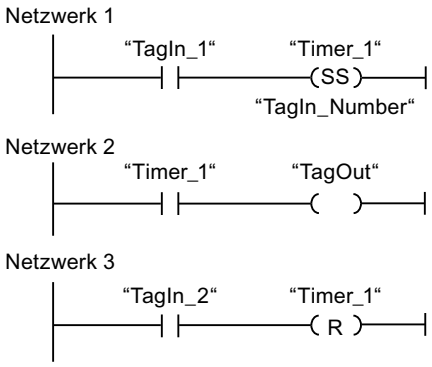

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

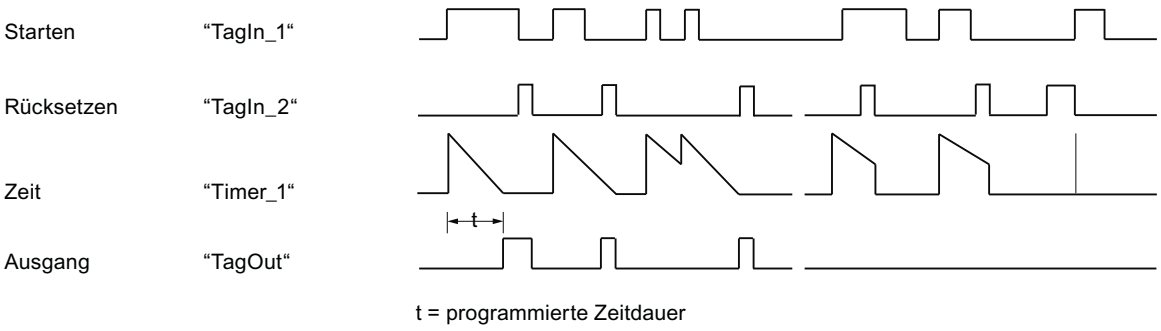

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SA): Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (Negative Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist. Wenn das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie die Zeitdauer an mit der die Zeit ablaufen wird und in <Operand2> oberhalb der Anweisung geben Sie die Zeit an, die gestartet wird.

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Operand2>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

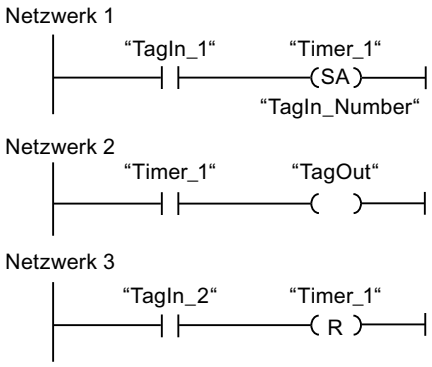

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Solange die Zeit läuft, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

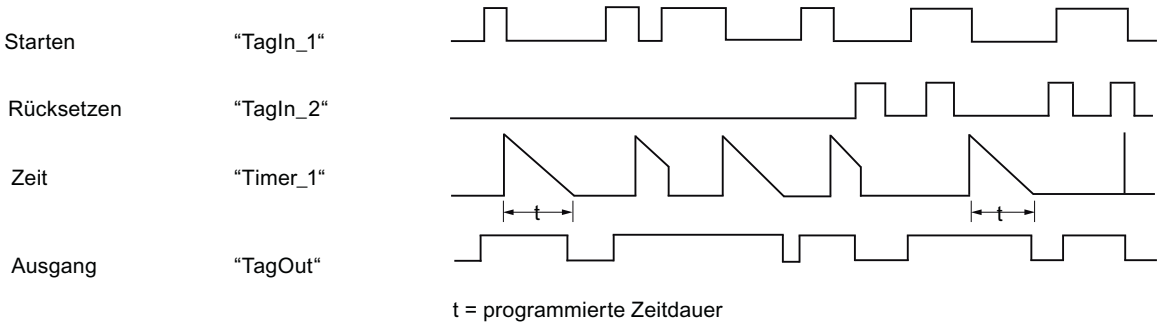

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu KOP (Seite 7425)

Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf Null zurückgesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTU_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CU	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Zähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

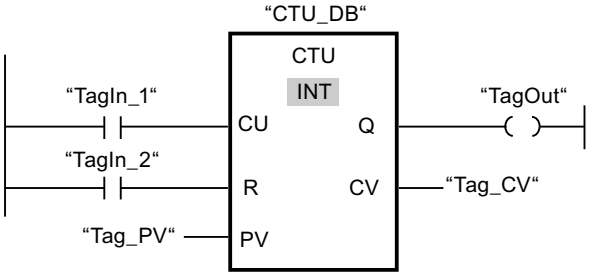

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CV" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz von lokalen Variablen (Seite 7269)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Grundlagen zu KOP (Seite 7425)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Eingang LD auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CD	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

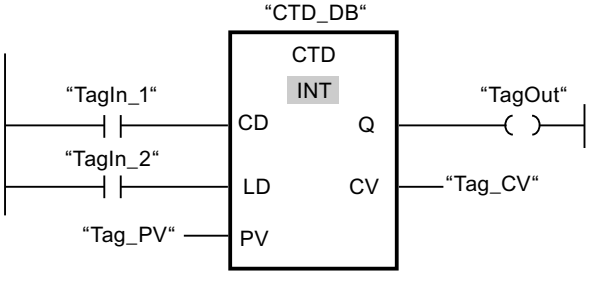

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert am Ausgang "Tag_CV" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu KOP (Seite 7425)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0".

Am Ausgang QD können Sie den Status des Rückwertszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CU	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

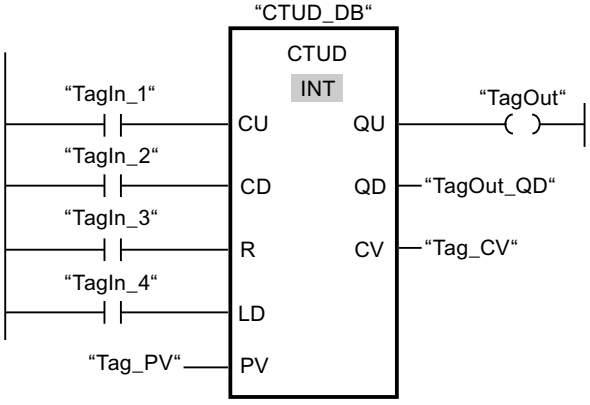

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Der Ausgang "TagOut_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut_QD" den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Grundlagen zu KOP (Seite 7425)

Legacy

Z_VORW: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L oder Konstante	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

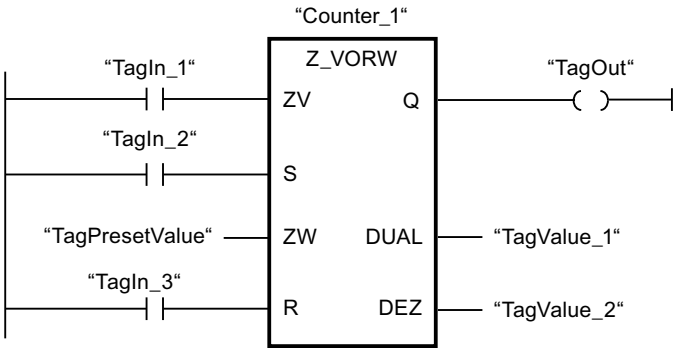

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Z_RUECK: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZR	Input	BOOL	E, A, M, D, L oder Konstante	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

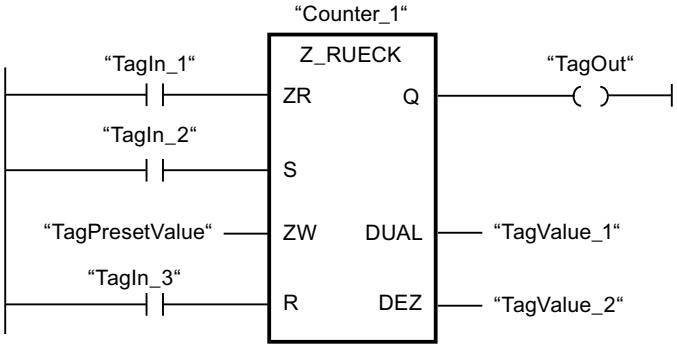

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L oder Konstante	Vorwärtszähleingang
ZR	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

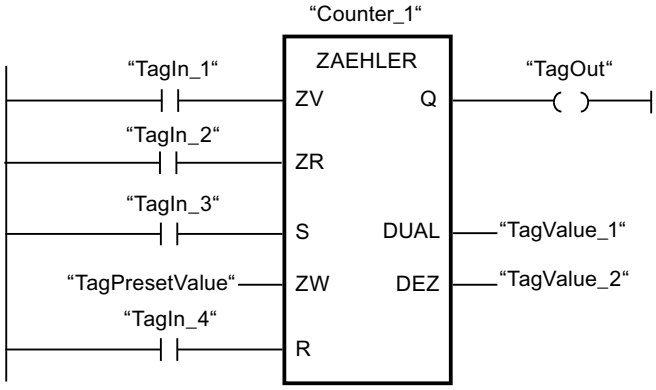

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SZ): Zähleranfangswert setzen

Beschreibung

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Am Operandenplatzhalter <Operand1> unterhalb der Anweisung geben Sie den Wert an mit der Zähler voreingestellt wird und in <Operand2> oberhalb der Anweisung geben Sie den Zähler an.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	WORD	E, A, M, D, L oder Konstante	Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999)
<Operand2>	InOut/Input	COUNTER	Z	Zähler, der voreingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

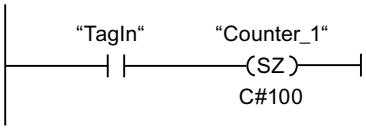

Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(ZV): Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

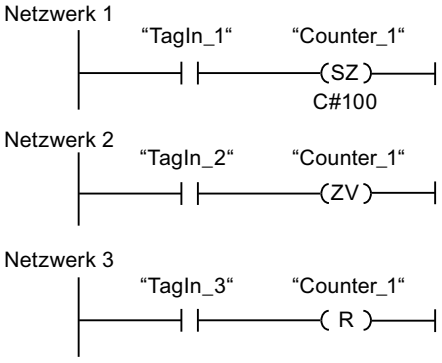

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(ZR): Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

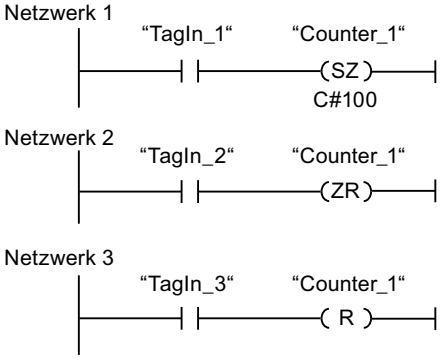

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Vergleicher

CMP ==: Gleich

Beschreibung

Mit der Anweisung "Gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) gleich dem zweiten Vergleichswert (<Operand2>) ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die

Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite der Operanden gleich sein.

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann empfehlen wir Ihnen, statt der Anweisung "CMP ==: Gleich" die Anweisung "IN_RANGE: Wert innerhalb Bereich" zu verwenden.

Bei Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefert die Anweisung "CMP ==: Gleich" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'AA'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'aa'	'aaa'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP ==: Gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP ==: Gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP ==: Gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP ==: Gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP ==: Gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefert die Anweisung "CMP ==: Gleich" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Vergleich von Hardware-Datentypen

Um Operanden vom Datentyp PORT vergleichen zu können, müssen Sie aus der Klappliste der Anweisungsbox den Datentyp WORD auswählen.

Wenn Sie die beiden Hardware-Datentypen HW_IO und HW_DEVICE vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Wählen Sie für den Vergleich von Strukturen den Datentyp VARIANT aus der Klappliste der Anweisungsbox aus. Variablen aus folgenden Datentypen sind möglich:

- PLC-Datentyp (UDT)
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Instanzdatenbausteine und Variablen von anonymen Strukturen sind nicht zulässig.)
- Datenbausteine, die von einem PLC-Datentyp abgeleitet sind.
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY bei der Auswahl VARIANT miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Hinweis

ARRAY of BOOL

Wenn Sie zwei Operanden vom Datentyp ARRAY of BOOL miteinander vergleichen, aus der Klappliste den Datentyp VARIANT ausgewählt haben und die Anzahl der Elemente nicht durch 8 teilbar ist, dann werden auch die Füll-Bits verglichen. Das kann das Vergleichsergebnis beeinflussen.

Wenn ein Element innerhalb der Strukturen, die Sie vergleichen möchten, ein ungültiger STRING/WSTRING, eine ungültige Zeit- oder Datumsangabe oder eine ungültige Gleitpunktzahl ist, dann liefert das Vergleichsergebnis im VKE (Verknüpfungsergebnis) den Signalzustand "0".

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen:

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert
Wie oben beschrieben, stehen Ihnen die Datentypen ARRAY, STRUCT (innerhalb eines PLC-Datentyps), VARIANT, ANY und PLC-Datentyp (UDT) erst ab den Firmware Versionen 2.0 bzw. 4.2 zur Verfügung.				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

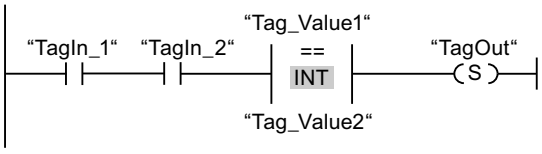

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" = "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

CMP <>: Ungleich

Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) ungleich dem zweiten Vergleichswert (<Operand2>) ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite der Operanden gleich sein.

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann empfehlen wir Ihnen, statt der Anweisung "CMP <>: Ungleich" die Anweisung "OUT_RANGE: Wert außerhalb Bereich" zu verwenden.

Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefert die Anweisung "CMP <>: Ungleich" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'Hallo Welt'	'HalloWelt'	1

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'AA'	0
'aa'	'aaa'	1

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <>: Ungleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <>: Ungleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP <>: Ungleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <>: Ungleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <>: Ungleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefert die Anweisung "CMP <>: Ungleich" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Vergleich von Hardware-Datentypen

Um Operanden vom Datentyp PORT vergleichen zu können, müssen Sie aus der Klappliste der Anweisungsbox den Datentyp WORD auswählen.

Wenn Sie die beiden Hardware-Datentypen HW_IO und HW_DEVICE vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Wählen Sie für den Vergleich von Strukturen den Datentyp VARIANT aus der Klappliste der Anweisungsbox aus. Die folgenden Datentypen sind möglich:

- PLC-Datentyp
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp (UDT) enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Anonyme Strukturen sind nicht zulässig.)
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY bei der Auswahl VARIANT miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Hinweis

ARRAY of BOOL

Wenn Sie zwei Operanden vom Datentyp ARRAY of BOOL miteinander vergleichen, aus der Klappliste den Datentyp VARIANT ausgewählt haben und die Anzahl der Elemente nicht durch 8 teilbar ist, dann werden auch die Füll-Bits verglichen. Das kann das Vergleichsergebnis beeinflussen.

Wenn ein Element innerhalb der Strukturen, die Sie vergleichen möchten, ein ungültiger STRING/WSTRING, eine ungültige Zeit- oder Datumsangabe oder eine ungültige Gleitpunktzahl ist, dann liefert das Vergleichsergebnis im VKE (Verknüpfungsergebnis) den Signalzustand "0".

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen:

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert
Wie oben beschrieben, stehen Ihnen die Datentypen ARRAY, STRUCT (innerhalb eines PLC-Datentyps), VARIANT, ANY und PLC-Datentyp (UDT) erst ab den Firmware Versionen 2.0 bzw. 4.2 zur Verfügung.				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

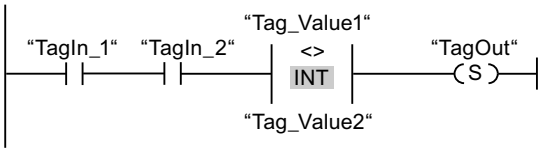

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <> "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

CMP >=: Größer gleich

Beschreibung

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'BB'	'AA'	1
'AAA'	'AA'	1
'Hallo Welt'	'Hallo Welt'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'AAA'	'a'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >=: Größer gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >=: Größer gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP >=: Größer gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >=: Größer gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP >=: Größer gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> größer (jünger) oder gleich dem Zeitpunkt am <Operand2> ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

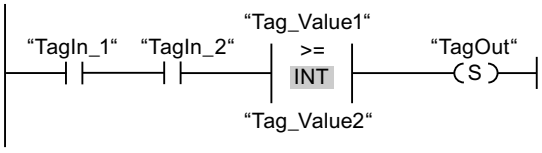

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" >= "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Grundlagen zu KOP (Seite 7425)

CMP <=: Kleiner gleich

Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'AAA'	'a'	1
'Hallo Welt'	'Hallo Welt'	1
'HalloWelt'	'Hallo Welt'	0
'BB'	'AA'	0
'AAA'	'AA'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <=: Kleiner gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <=: Kleiner gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP <=: Kleiner gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <=: Kleiner gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <=: Kleiner gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> kleiner (älter) oder gleich dem Zeitpunkt am <Operand2> ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

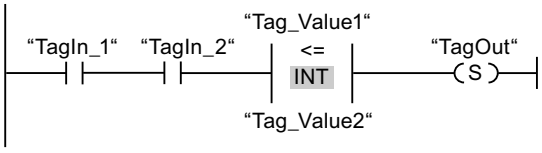

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <= "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

CMP >: Größer

Beschreibung

Mit der Anweisung "Größer" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die

Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'BB'	'AA'	1
'AAA'	'AA'	1
'AA'	'aa'	0
'AAA'	'a'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >: Größer" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >: Größer" inaktiv.

Ergebnis:

Für die Anweisung "CMP >: Größer" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >: Größer" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP >: Größer" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> größer (jünger) als der Zeitpunkt am <Operand2> ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

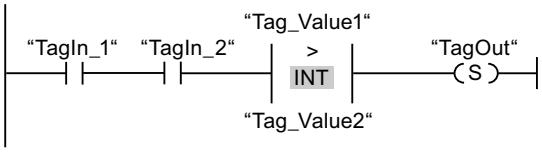

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" > "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

CMP <: Kleiner

Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'AAA'	'a'	1
'BB'	'AA'	0
'AAA'	'AA'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <: Kleiner" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <: Kleiner" inaktiv.

Ergebnis:

Für die Anweisung "CMP <: Kleiner" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <: Kleiner" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <: Kleiner" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> kleiner (älter) als der Zeitpunkt am <Operand2> ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

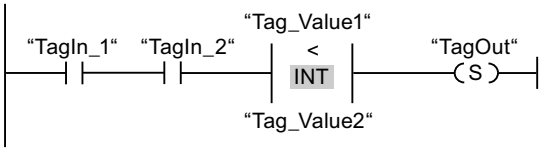

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" < "Tag_Value2" ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Grundlagen zu KOP (Seite 7425)

IN_RANGE: Wert innerhalb Bereich

Beschreibung

Mit der Anweisung "Wert innerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL innerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert innerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der

Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich MIN <= VAL oder VAL <= MAX erfüllt, liefert der Boxausgang den Signalzustand "1". Bei nicht erfülltem Vergleich steht am Boxausgang der Signalzustand "0".

Wenn der Boxeingang den Signalzustand "0" führt, wird die Anweisung "Wert innerhalb Bereich" nicht bearbeitet.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind und der Boxeingang beschaltet ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert innerhalb Bereich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Boxeingang	Input	BOOL	E, A, M, D, L oder Konstante	Ergebnis der vorherigen Verknüpfung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VAL	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Vergleichswert
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
Boxausgang	Output	BOOL	E, A, M, D, L	Ergebnis des Vergleichs

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

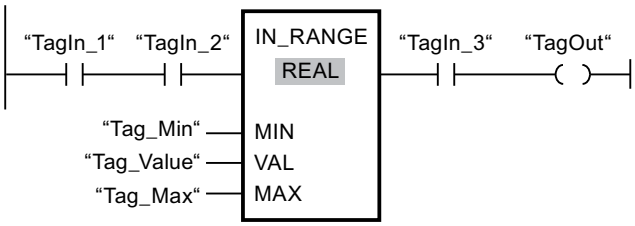

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Wert des Operanden "Tag_Value" liegt innerhalb des Wertebereichs, der durch die aktuellen Werte der Operanden "Tag_Min" und "Tag_Max" vorgegeben ist (MIN <= VAL oder VAL <= MAX).
- Der Operand "TagIn_3" liefert den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

OUT_RANGE: Wert außerhalb Bereich

Beschreibung

Mit der Anweisung "Wert außerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL außerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert außerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich MIN > VAL oder VAL > MAX erfüllt, liefert der Boxausgang den Signalzustand "1". Der Boxausgang liefert auch den Signalzustand "1", wenn ein angegebener Operand vom Datentyp REAL einen ungültigen Wert aufweist.

Der Boxausgang liefert den Signalzustand "0", wenn der Wert am Eingang VAL die Bedingung MIN > VAL oder VAL > MAX nicht erfüllt.

Wenn der Boxeingang den Signalzustand "0" führt, wird die Anweisung "Wert außerhalb Bereich" nicht bearbeitet.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind und der Boxeingang beschaltet ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert außerhalb Bereich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Boxeingang	Input	BOOL	E, A, M, D, L oder Konstante	Ergebnis der vorherigen Verknüpfung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VAL	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Vergleichswert
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
Boxausgang	Output	BOOL	E, A, M, D, L	Ergebnis des Vergleichs

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

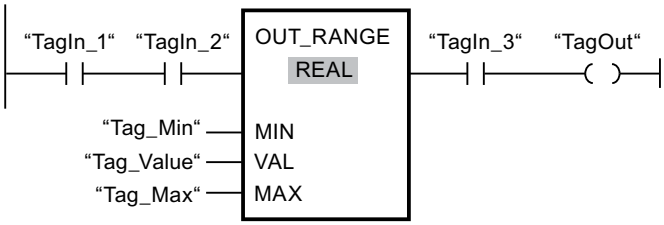

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Wert des Operanden "Tag_Value" liegt außerhalb des Wertebereichs, der durch die Werte der Operanden "Tag_Min" und "Tag_Max" vorgegeben ist (MIN > VAL oder VAL > MAX).
- Der Operand "TagIn_3" liefert den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

----I OK I----: Gültigkeit prüfen

Beschreibung

Mit der Anweisung "Gültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine gültige Gleitpunktzahl ist. Die Abfrage wird bei jedem Programmzyklus gestartet, wenn der Eingang der Anweisung den Signalzustand "1" hat.

Der Ausgang der Anweisung hat den Signalzustand "1", wenn der Wert des Operanden zum Zeitpunkt der Abfrage eine gültige Gleitpunktzahl ist und der Eingang der Anweisung den Signalzustand "1" hat. In allen anderen Fällen hat der Ausgang der Anweisung "Gültigkeit prüfen" den Signalzustand "0".

Die Anweisung "Gültigkeit prüfen" können Sie in Verbindung mit dem EN-Mechanismus nutzen. Wenn Sie die Anweisungsbox an einen Freigabeeingang EN anschließen, wird der Freigabeeingang nur bei einer positiven Abfrage der Wertgültigkeit gesetzt. Durch diese Funktionalität können Sie sicherstellen, dass eine Anweisung nur dann aktiviert wird, wenn der Wert des angegebenen Operanden eine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gültigkeit prüfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L	Wert, der abgefragt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

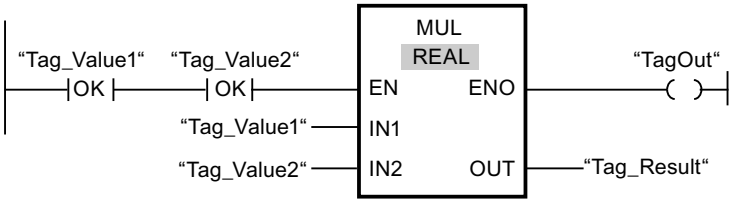

Wenn die Werte der Operanden "Tag_Value1" und "Tag_Value2" gültige Gleitpunktzahlen aufweisen, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Multiplikationsergebnis wird dann in den Operanden "Tag_Result" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

----I NOT_OK I----: Ungültigkeit prüfen

Beschreibung

Mit der Anweisung "Ungültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine ungültige Gleitpunktzahl ist. Die Abfrage wird bei jedem Programmzyklus gestartet, wenn der Eingang der Anweisung den Signalzustand "1" hat.

Der Ausgang der Anweisung hat den Signalzustand "1", wenn der Wert des Operanden zum Zeitpunkt der Abfrage eine ungültige Gleitpunktzahl ist und der Eingang der Anweisung den Signalzustand "1" hat. In allen anderen Fällen hat der Ausgang der Anweisung "Ungültigkeit prüfen" den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungültigkeit prüfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L	Wert, der abgefragt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

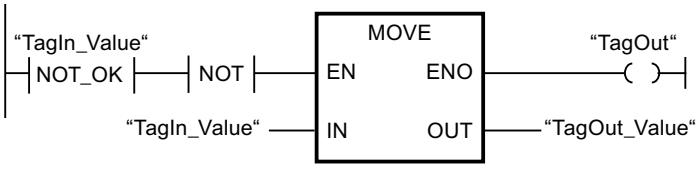

Wenn der Wert des Operanden "TagIn_Value" eine ungültige Gleitpunktzahl ist, wird die Anweisung "Wert kopieren" nicht ausgeführt. Der Operand "TagOut" am Freigabeausgang ENO wird auf den Signalzustand "0" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

VARIANT

EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable (<Operand1>), die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp einer Variable (<Operand2>) auf "Gleich".

Der <Operand1> muss den Datentyp VARIANT haben. Der <Operand2> kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die

Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP ==" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP ==: Gleich (Seite 712)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

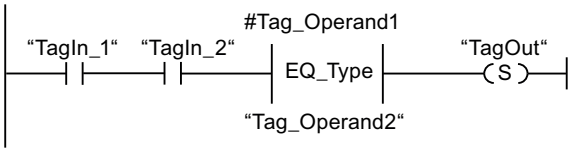

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand1 ist gleich "Tag_Operand2".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie den Datentyp ab, welchen eine Variable nicht hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable (<Operand1>), die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp einer Variable (<Operand2>) auf "Ungleich".

Der <Operand1> muss den Datentyp VARIANT haben. Der <Operand2> kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP <>" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP <>: Ungleich (Seite 718)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

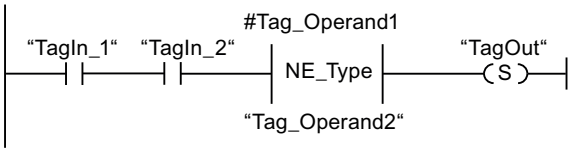

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand1 ist ungleich "Tag_Operand2".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

EQ_ElemType: Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable (<Operand1>), die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp einer Variable (<Operand2>) auf "Gleich".

Der <Operand1> muss den Datentyp VARIANT haben. Der <Operand2> kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable (<Operand1>) um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP ==" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP ==: Gleich (Seite 712)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

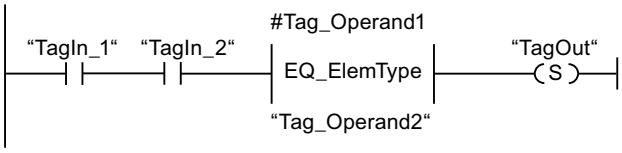

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand1 ist gleich "Tag_Operand2".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

NE_ElemType: Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen" fragen Sie ab, welchen Datentyp eine Variable nicht hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable (<Operand1>), die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp einer Variable (<Operand2>) auf "Ungleich".

Der <Operand1> muss den Datentyp VARIANT haben. Der <Operand2> kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable (<Operand1>) um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP <>" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP <>: Ungleich (Seite 718)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Baustein-schnittstelle möglich.)		Erster Operand
<Operand2>	Input	Bitfolgen, Ganz-zahlen, Gleitpunkt-zahlen, Zeiten, Da-tum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Da-tentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

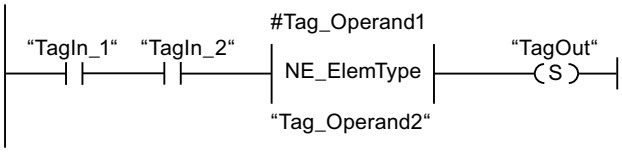

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand1 ist ungleich "Tag_Operand2".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

IS_NULL: Auf GLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf GLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz auf einen NULL-Zeiger verweist und damit nicht auf ein Objekt zeigt.

Der <Operand> muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "0", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf GLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf GLEICH NULL verglichen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

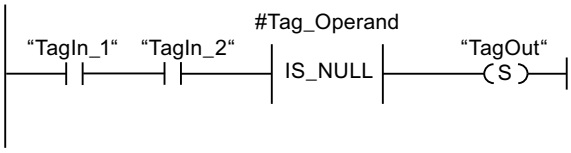

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand zeigt auf kein Objekt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

NOT_NULL: Auf UNGLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz nicht auf einen NULL-Zeiger verweist und damit auf ein Objekt zeigt.

Der <Operand> muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "1", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf UNGLEICH NULL verglichen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

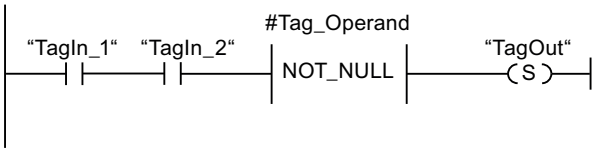

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand zeigt auf ein Objekt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

IS_ARRAY: Auf ARRAY abfragen

Beschreibung

Mit der Anweisung "Auf ARRAY abfragen" fragen Sie ab, ob der VARIANT auf eine Variable vom Datentyp ARRAY zeigt.

Der <Operand> muss den Datentyp VARIANT haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf ARRAY abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf ARRAY abgefragt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

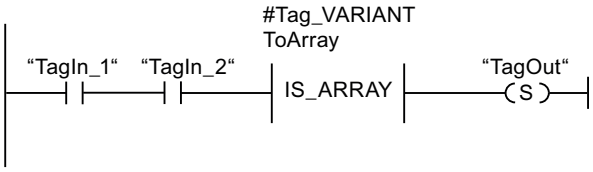

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_VARIANTToArray ist vom Datentyp ARRAY.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

EQ_TypeOfDB: Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable (<Operand1>) adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit einem Datentyp (<Operand2>) auf "Gleich".

Der <Operand1> muss den Datentyp DB_ANY haben. Der <Operand2> kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
<Operand2>	Input	TYPE_ID	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

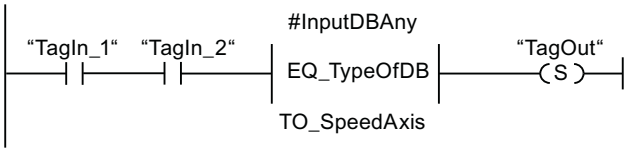

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Datentyp des Operanden #InputDBAny adressierten DB ist gleich dem Datentyp TO_SpeedAxis.

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu KOP (Seite 7425)

NE_TypeOfDB: Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein nicht hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable (<Operand1>) adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit dem Datentyp (<Operand2>) auf "Ungleich".

Der <Operand1> muss den Datentyp DB_ANY haben. Der <Operand2> kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den <Operand1> geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den <Operand2> geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
<Operand2>	Input	TYPE_ID	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

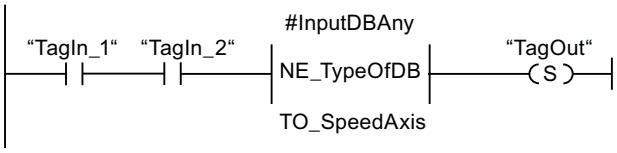

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Datentyp des Operanden #InputDBAny adressierten DB ist ungleich dem Datentyp TO_SpeedAxis.

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu KOP (Seite 7425)

Mathematische Funktionen

CALCULATE: Berechnen

Beschreibung

Mit der Anweisung "Berechnen" definieren und führen Sie, abhängig vom gewählten Datentyp, einen Ausdruck zum Berechnen mathematischer Operationen oder komplexer Verknüpfungen aus.

Den Datentyp der Anweisung wählen Sie aus der Klappliste "???" der Anweisungsbox. Abhängig vom ausgewählten Datentyp können Sie die Funktionalität bestimmter Anweisungen kombinieren, um eine komplexe Berechnung auszuführen. Die Angabe des zu berechnenden Ausdrucks erfolgt über einen Dialog, den Sie über das Symbol "Taschenrechner" am rechten oberen Rand der Anweisungsbox öffnen können. Im Ausdruck können die Namen der Eingangsparameter und die Syntax der Anweisungen enthalten sein. Die Angabe von Operandennamen oder -adressen ist nicht zulässig.

Die Anweisungsbox enthält im Grundzustand mindestens 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert.

Die Werte der Eingänge werden zur Ausführung des angegebenen Ausdrucks verwendet. Im Ausdruck müssen dabei nicht alle definierten Eingänge verwendet werden. Das Ergebnis der Anweisung wird an den Ausgang OUT übertragen.

Hinweis

Wenn eine der mathematischen Operationen im Ausdruck fehlschlägt, dann wird kein Ergebnis an den Ausgang OUT übertragen und der Freigabeausgang ENO liefert den Signalzustand "1".

Wenn Sie im Ausdruck Eingänge verwenden, die in der Box nicht verfügbar sind, werden diese automatisch eingefügt. Voraussetzung dafür ist, dass die Nummerierung der neu zu definierenden Eingänge im Ausdruck lückenlos ist. Sie können zum Beispiel den Eingang IN4 im Ausdruck nicht verwenden, wenn der Eingang IN3 nicht definiert ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis oder ein Zwischenergebnis der Anweisung "Berechnen" liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.
- Bei der Ausführung einer der im Ausdruck angegebenen Anweisungen ist ein Fehler aufgetreten.

Die folgende Tabelle zeigt die Anweisungen, die, abhängig vom gewählten Datentyp im Ausdruck der Anweisung "Berechnen", kombiniert ausgeführt werden können:

Datentyp	Anweisung	Syntax	Beispiel
Bitfolgen	AND: UND verknüpfen	AND	IN1 AND IN2 OR IN3
	OR: ODER verknüpfen	OR	
	XOR: EXCLUSIV ODER verknüpfen	XOR	
	INV: Einerkomplement erzeugen	NOT	
	SWAP: Anordnung ändern [1]	SWAP	
Ganzzahlen	ADD: Addieren	+	(IN1 + IN2) * IN3; (ABS(IN2)) * (ABS(IN1))
	SUB: Subtrahieren	-	
	MUL: Multiplizieren	*	
	DIV: Dividieren	/	
	MOD: Divisionsrest gewinnen	MOD	
	INV: Einerkomplement erzeugen	NOT	
	NEG: Zweierkomplement erzeugen	-(in1)	
	ABS: Absolutwert bilden	ABS( )	
Gleitpunktzahlen	ADD: Addieren	+	((SIN(IN2) * SIN(IN2) + (SIN(IN3) * SIN(IN3)) / IN3)); (SQR(SIN(IN2)) + (SQR(COS(IN3)) / IN2))
	SUB: Subtrahieren	-	
	MUL: Multiplizieren	*	
	DIV: Dividieren	/	
	EXPT: Potenzieren	**	
	ABS: Absolutwert bilden	ABS( )	
	SQR: Quadrat bilden	SQR( )	
	SQRT: Quadratwurzel bilden	SQRT( )	
	LN: Natürlichen Logarithmus bilden	LN( )	
	EXP: Exponentialwert bilden	EXP( )	
	FRAC: Nachkommastellen ermitteln	FRAC( )	
	SIN: Sinuswert bilden	SIN( )	
	COS: Cosinuswert bilden	COS( )	
	TAN: Tangenswert bilden	TAN( )	
	ASIN: Arcussinuswert bilden	ASIN( )	
	ACOS: Arcuscosinuswert bilden	ACOS( )	
	ATAN: Arcustangenswert bilden	ATAN( )	
	NEG: Zweierkomplement erzeugen	-(in1)	
	TRUNC: Ganzzahl erzeugen	TRUNC( )	
	ROUND: Zahl runden	ROUND( )	
	CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen	CEIL( )	
	FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen	FLOOR( )	

[1] Nicht für den Datentyp BYTE möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Berechnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster verfügbarer Eingang
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter verfügbarer Eingang
INn	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ausgang, an den das Endergebnis übertragen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

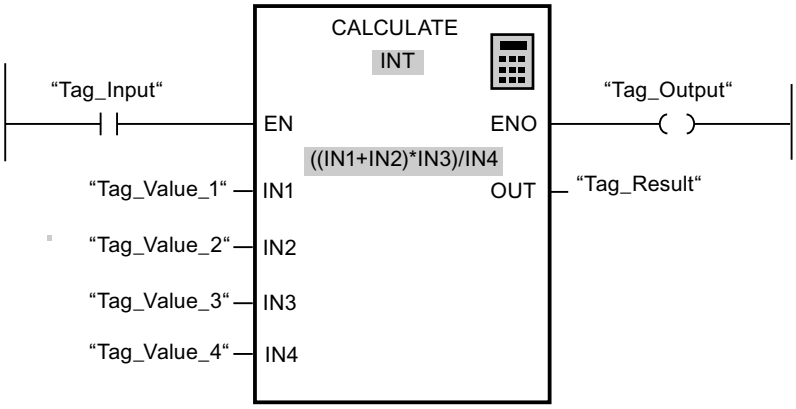

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value_1	4
IN2	Tag_Value_2	4
IN3	Tag_Value_3	3
IN4	Tag_Value_4	2
OUT	Tag_Result	12

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung "Berechnen" ausgeführt. Der Wert des Operanden "Tag_Value_1" wird mit dem Wert des Operanden "Tag_Value_2" addiert. Die Summe wird mit dem Wert des Operanden "Tag_Value_3" multipliziert. Das Produkt wird durch den Wert des Operanden "Tag_Value_4" dividiert. Der Quotient wird als Endergebnis in den Operanden "Tag_Result" am Ausgang OUT der Anweisung übertragen. Wenn keine Fehler während der Bearbeitung der einzelnen Anweisungen auftreten, werden der Freigabeausgang ENO und der Operand "Tag_Output" auf "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anweisung "Berechnen" verwenden (Seite 7454)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Beispiel für das Berechnen einer Gleichung (Seite 7487)

Grundlagen zu KOP (Seite 7425)

ADD: Addieren

Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragt die Summe am Ausgang OUT ab (OUT := IN1+IN2).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter addiert. Die Summe wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Summand
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Summand
INn	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte, die addiert werden.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Summe

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

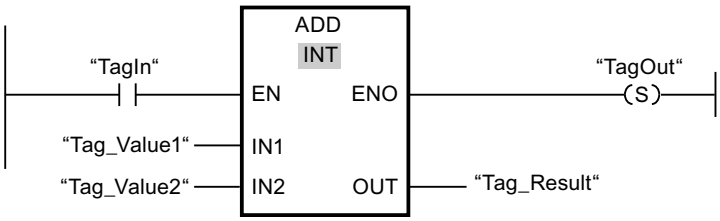

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Addieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Auswahl eines Datentyps (Seite 7444)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

SUB: Subtrahieren

Beschreibung

Mit der Anweisung "Subtrahieren" ziehen Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 ab und fragen die Differenz am Ausgang OUT ab (OUT := IN1-IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Differenz

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

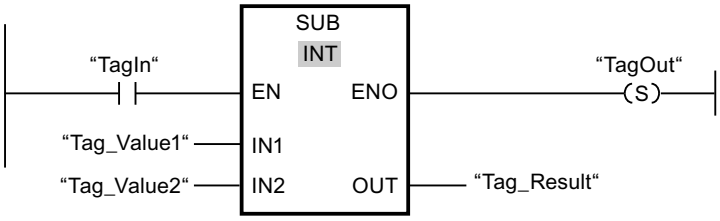

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Subtrahieren" ausgeführt. Der Wert des Operanden "Tag_Value2" wird vom Wert des Operanden "Tag_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MUL: Multiplizieren

Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und das Produkt am Ausgang OUT abfragen (OUT := IN1*IN2).

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter multipliziert. Das Produkt wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikator
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikand
INn	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte, die multipliziert werden können.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Produkt

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

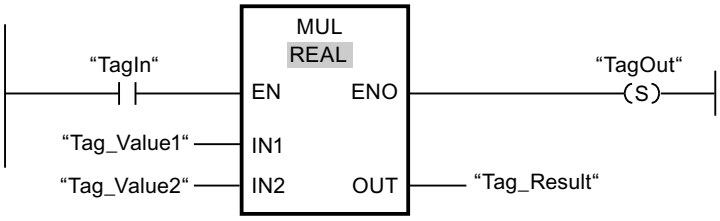

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Multiplikationsergebnis wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

DIV: Dividieren

Beschreibung

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab (OUT := IN1/IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

Division durch den Wert "0"

Die Division eines Dividend (IN1) durch einen Divisor (IN2) mit dem Wert "0" setzt den Freigabeausgang (ENO) in den Signalzustand "TRUE".

Der Wert des Quotienten (OUT) wird bei der Division durch "0" wie folgt durch den Datentyp beeinflusst:

- Der Datentyp ist INT oder LREAL: Der Wert des Quotienten (OUT) ist "0".
- Der Datentyp ist REAL: Der Wert des Quotienten (OUT) ist der Maximalwert (2143289344).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Wert des Quotienten

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

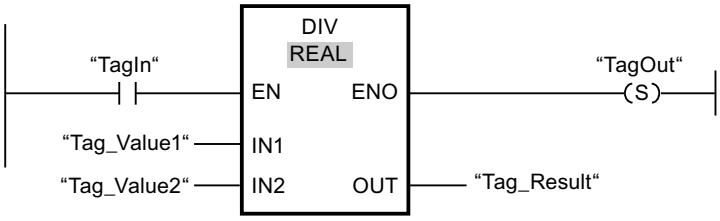

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Dividieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen	E, A, M, D, L, P	Divisionsrest

Aus der Klappliste "Auto (???)" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

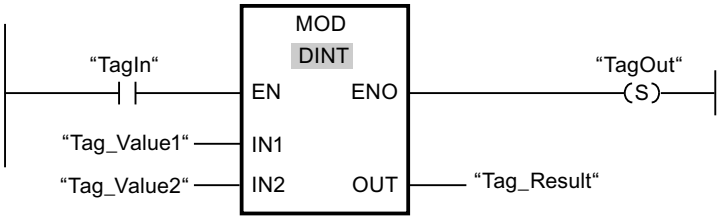

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Divisionsrest gewinnen" ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag_Result"

gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	Zweierkomplement des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

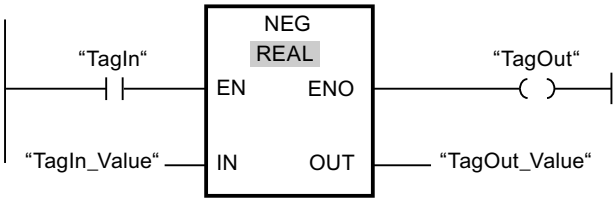

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zweierkomplement erzeugen" ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

INC: Inkrementieren

Beschreibung

Mit der Anweisung "Inkrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst höheren Wert und fragen das Ergebnis ab. Die Bearbeitung der Anweisung "Inkrementieren" wird nur gestartet, wenn der Signalzustand am Freigabeeingang EN "1" ist. Wenn kein Überlauf während der Bearbeitung auftritt, führt der Freigabeausgang ENO ebenfalls den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN/OUT	InOut	Ganzzahlen	E, A, M, D, L	Wert, der inkrementiert wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

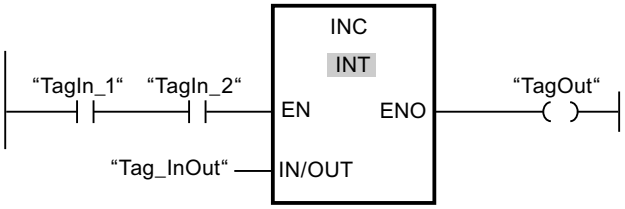

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag_InOut" um eins erhöht und der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

DEC: Dekrementieren

Beschreibung

Mit der Anweisung "Dekrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst niedrigeren Wert und fragen das Ergebnis ab. Die Bearbeitung der Anweisung "Dekrementieren" wird gestartet, wenn der Signalzustand am Freigabeeingang EN "1" ist. Wenn der Wertebereich des ausgewählten Datentyps während der Bearbeitung nicht unterschritten wird, führt der Ausgang ENO ebenfalls den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN/OUT	InOut	Ganzzahlen	E, A, M, D, L	Wert, der dekrementiert wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

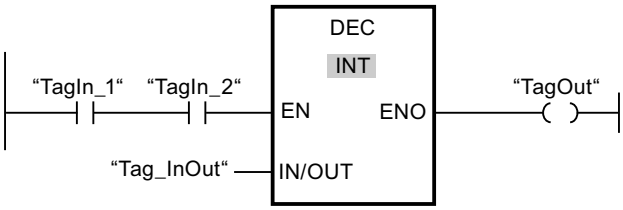

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag_InOut" um eins verringert und der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	Absolutwert des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

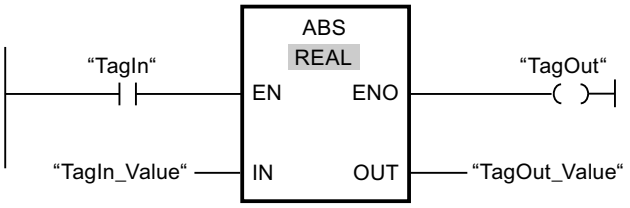

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Absolutwert bilden" ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn_Value" und gibt das Ergebnis am Ausgang "TagOut_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den kleinsten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden in der Box aufsteigend nummeriert.

Für die Ausführung der Anweisung müssen mindestens zwei und können maximal 100 Eingänge angegeben werden.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
OUT	Output	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

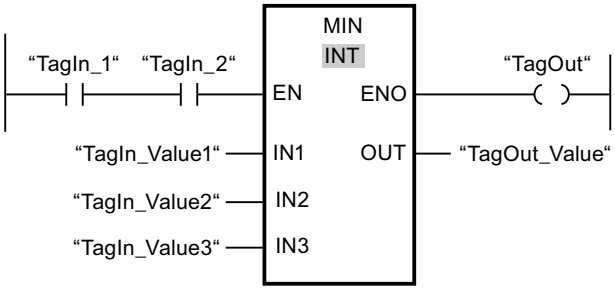

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den größten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden in der Box aufsteigend nummeriert.

Für die Ausführung der Anweisung müssen mindestens zwei und können maximal 100 Eingänge angegeben werden.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
OUT	Output	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

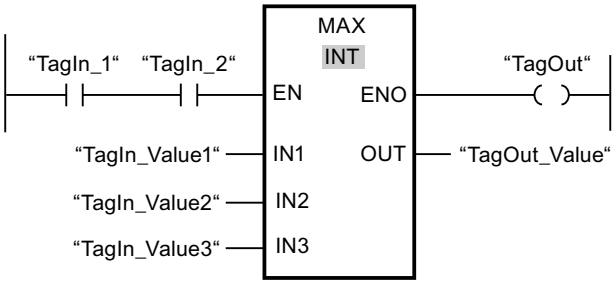

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis der am Parmeter IN angegebene Wert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	Eingangswert

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.					

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

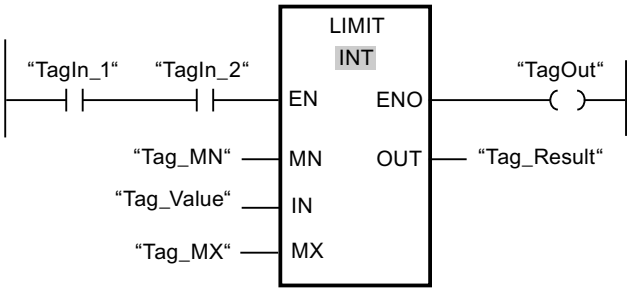

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_MN" und "Tag_MX" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_MN" in den Ausgang

"Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert einer Gleitpunktzahl und schreiben das Ergebnis in den Ausgang OUT.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswertes

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

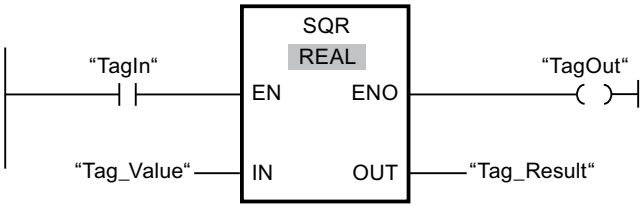

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadrat bilden" ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Datentyp eines KOP-Elements auswählen (Seite 7444)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert einer Gleitpunktzahl die Quadratwurzel und schreiben das Ergebnis in den Ausgang OUT. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadratwurzel des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

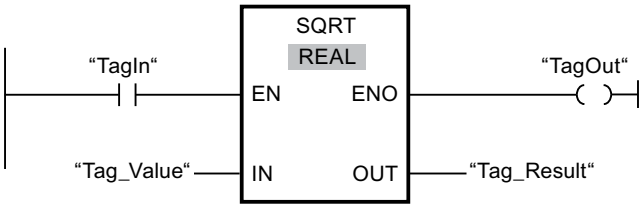

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadratwurzel bilden" ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Datentyp eines KOP-Elements auswählen (Seite 7444)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis e (e = 2,718282). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

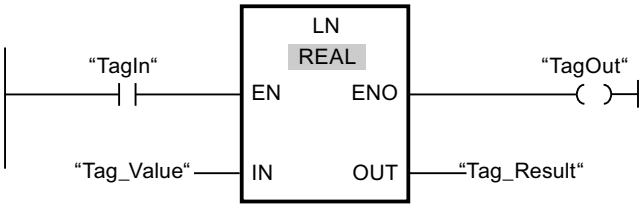

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Natürlichen Logarithmus bilden" ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag_Value" und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden (OUT = e^{IN}).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

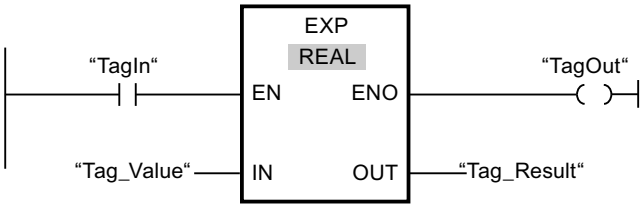

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Exponentialwert bilden" ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Sinus des angegebenen Winkels

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

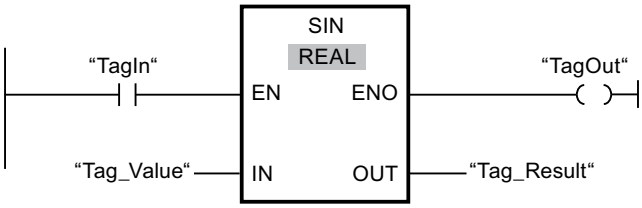

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	1.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Sinuswert bilden" ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

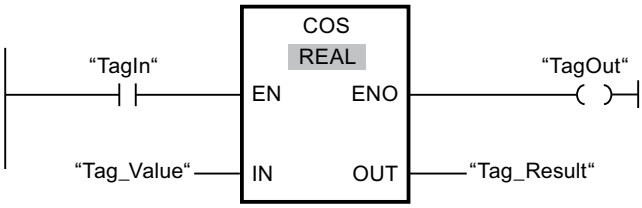

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Cosinuswert bilden" ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag_Value"

angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Tangens des angegebenen Winkels

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

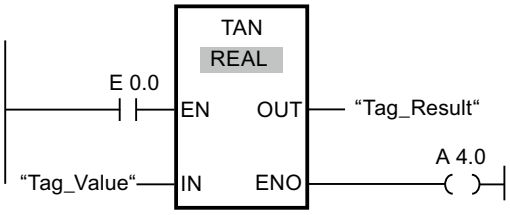

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Sinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

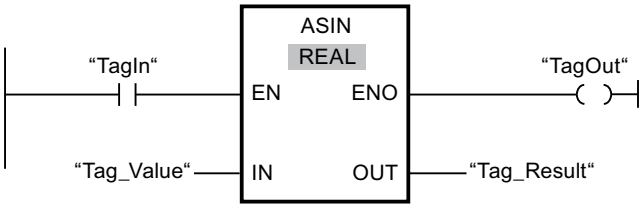

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcussinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Cosinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

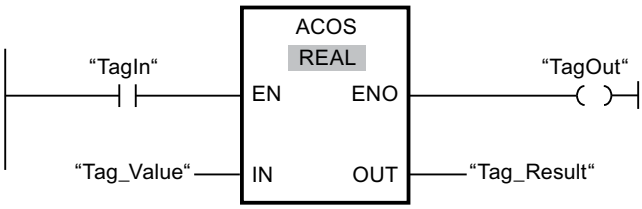

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcuscosinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Tangenswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

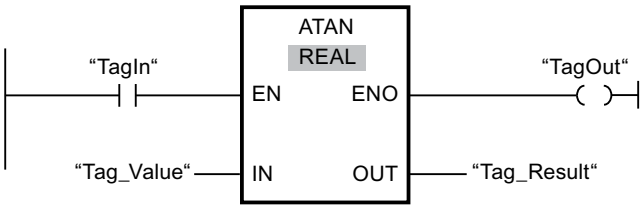

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+0.785398 (π/4)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcustangenswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu KOP (Seite 7425)

FRAC: Nachkommastellen ermitteln

Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" ermitteln Sie die Nachkommastellen des Wertes am Eingang IN. Das Ergebnis der Abfrage wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden. Wenn am Eingang IN z. B. der Wert 123,4567 ansteht, liefert der Ausgang OUT den Wert 0,4567.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Nachkommastellen ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Wert, dessen Nachkommastellen ermittelt werden.
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Nachkommastellen des Werts am Eingang IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

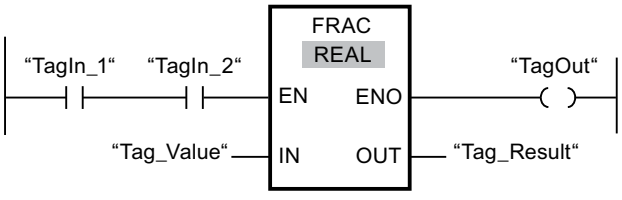

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	2.555
OUT	Tag_Result	0.555

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung "Nachkommastellen ermitteln" gestartet. Die Nachkommastellen aus dem Wert

des Operanden "Tag_Value" werden in den Operanden "Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

EXPT: Potenzieren

Beschreibung

Mit der Anweisung "Potenzieren" potenzieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2. Das Ergebnis der Anweisung wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden (OUT = $IN1^{IN2}$).

Der Eingang IN1 kann nur mit gültigen Gleitpunktzahlen belegt werden. Zur Belegung des Eingangs IN2 sind auch Ganzzahlen zulässig.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. es tritt einen Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Potenzieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Basiswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Wert, mit dem der Basiswert potenziert wird
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

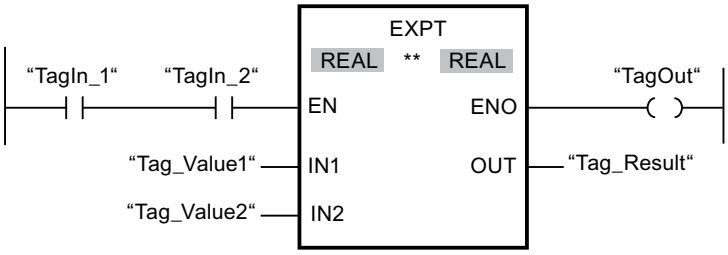

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung "Potenzieren" gestartet. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" potenziert. Das Ergebnis wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

Verschieben

MOVE: Wert kopieren

Beschreibung

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Datentyp am Parameter IN kann nicht in den angegebenen Datentyp am Parameter OUT1 konvertiert werden.

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1200:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE ,TOD, CHAR
WORD	WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
USINT	USINT, UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
INT	INT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UINT	UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UDINT	UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LREAL
TIME	TIME	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME
DATE	DATE	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE
TOD	TOD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, CHAR, Zeichen einer Zeichenkette[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, CHAR, WCHAR, Zeichen einer Zeichenkette[1)]
Zeichen einer Zeichenkette[1)]	Zeichen einer Zeichenkette	CHAR, WCHAR, Zeichen einer Zeichenkette
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
PLC-Datentyp (UDT)	PLC-Datentyp (UDT)	PLC-Datentyp (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
IEC_UCOUN-TER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUN-TER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUN-TER	IEC_UDCOUNTER	IEC_UDCOUNTER

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1500:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE ,TOD, LTOD, CHAR
WORD	WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
DWORD	DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
LWORD	LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
USINT	USINT, UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
INT	INT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UINT	UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD. LTOD
DINT	DINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UDINT	UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
LINT	LINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
ULINT	ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LWORD, LREAL

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
S5TIME	S5TIME	WORD, S5TIME
TIME	TIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME
LTIME	LTIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME
DATE	DATE	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE
DT	DT	DT
LDT	LDT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT
TOD	TOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD
LTOD	LTOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, LWORD, CHAR, Zeichen einer Zeichenkette[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, LWORD, CHAR, WCHAR, Zeichen einer Zeichenkette[1)]
Zeichen einer Zeichenkette[1)]	Zeichen einer Zeichenkette	CHAR, WCHAR, Zeichen einer Zeichenkette
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
COUNTER	COUNTER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	TIMER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
PLC-Datentyp (UDT)	PLC-Datentyp (UDT)	PLC-Datentyp (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER	IEC_LTIMER
IEC_SCOUN-TER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUN-TER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUN-TER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUN-TER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUN-TER	IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUN-TER	IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUN-TER	IEC_ULCOUNTER	IEC_ULCOUNTER
REF()	Es findet keine Datentypkonvertierung statt. Die übergebene Variable muss genau dem Zieldatentyp entsprechen.	

[1] Mit der Anweisung "Wert kopieren" können Sie auch einzelne Zeichen einer Zeichenkette zu Operanden vom Datentyp CHAR oder WCHAR übertragen. Die Nummer des zu übertragenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" übertragen. Die Übertragung vom Operanden des Datentyps CHAR oder WCHAR zu einzelnen Zeichen einer Zeichenkette ist ebenfalls möglich. Sie können auch ein bestimmtes Zeichen einer Zeichenkette durch das Zeichen einer anderen Zeichenkette ersetzen.

[2] Das Übertragen von ganzen Feldern (ARRAY) ist nur dann möglich, wenn die Feldkomponenten der Operanden am Eingang IN und am Ausgang OUT1 vom gleichen Datentyp sind.

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen überschrieben.

Die Anweisungsbox enthält im Grundzustand 1 Ausgang (OUT1). Die Anzahl der Ausgänge ist erweiterbar. Die eingefügten Ausgänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung wird der Inhalt des Operanden am Eingang IN zu allen verfügbaren Ausgängen übertragen. Die Anweisungsbox kann nicht erweitert werden, wenn strukturierte Datentypen (DTL, STRUCT, ARRAY) oder Zeichen einer Zeichenkette übertragen werden.

Zum Kopieren von Operanden des Datentyps ARRAY können auch die Anweisungen "Bereich kopieren" (MOVE_BLK) und "Bereich ununterbrechbar kopieren" (UMOVE_BLK) verwendet werden. Operanden des Datentyps STRING oder WSTRING können mit der Anweisung "Zeichenkette verschieben" (S_MOVE) kopiert werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT)	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT)	E, A, M, D, L oder Konstante	Quellwert
OUT1	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT)	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT)	E, A, M, D, L	Operand, in den der Quellwert übertragen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

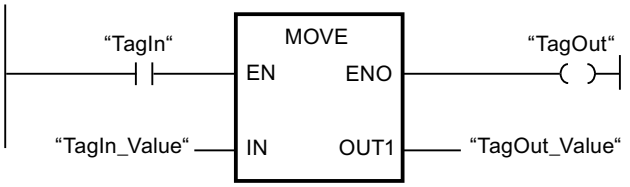

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn_Value" zum Operanden "TagOut_Value" und setzt den Ausgang "TagOut" auf den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Hinweis

Weitere Informationen zur Anweisung MOVE im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

MOVE_BLK: Bereich kopieren (Seite 804)

UMOVE_BLK: Bereich ununterbrechbar kopieren (Seite 811)

S_MOVE: Zeichenkette verschieben (Seite 2215)

Grundlagen zu KOP (Seite 7425)

Deserialize: Deserialisieren

Beschreibung

Mit der Anweisung "Deserialisieren" wandeln Sie die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder um und füllen dessen kompletten Inhalt.

Der Speicherbereich, in dem die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> liegt, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Speicherbereiche zugelassen. Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Stellen Sie vor der Umwandlung sicher, dass genügend Speicherplatz vorhanden ist. Wenn der Speicherbereich mithilfe der Anweisung "Serialisieren" befüllt wurde und eventuell Füll-Bytes eingefügt wurden, dann werden diese bei der Umwandlung nicht berücksichtigt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Mithilfe der Anweisung können Sie mehrere sequenzielle Darstellungsformen von umgewandelten Daten schrittweise wieder in ihren ursprünglichen Zustand umwandeln.

Wenn Sie nur eine einzelne sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder umwandeln möchten, dann können Sie dazu

auch direkt die Anweisung "TRCV: Daten über Kommunikationsverbindung empfangen" verwenden.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das BOOL eine Länge von 1 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Optimierter Speicherbereich

Um größere Strukturen zu deserialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich für die sequenzielle Darstellung auch mit optimiertem Zugriff deklarieren. Der Inhalt der sequenziellen Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich. Der Zugriff auf die Bytes des ARRAYs ist ausschließlich symbolisch möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deserialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC_ARRAY	Input	VARIANT ARRAY[*] of BYTE [1)]	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der Datenstrom gespeichert ist.
DEST_VARIABLE	InOut	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die deserialisiert werden soll.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
[1)] Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 möglich				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_ARRAY und DEST_VARIABLE überschneiden sich.
8136	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8150	Der Datentyp VARIANT am Parameter SRC_ARRAY enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_ARRAY
8153	Am Parameter SRC_ARRAY steht nicht genügend Speicherplatz zur Verfügung.
8154	Ungültiger Datentyp am Parameter SRC_ARRAY
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben
8251	Keine gültige Referenz am Parameter DEST_VARIABLE
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehlercode* (W#16#...)	Erläuterung
8136	Der Zugriff am Parameter SRC_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle eines FBs oder FCs
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle eines FBs oder FCs
Error	INT	
Label	STRING[4]	

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Target	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

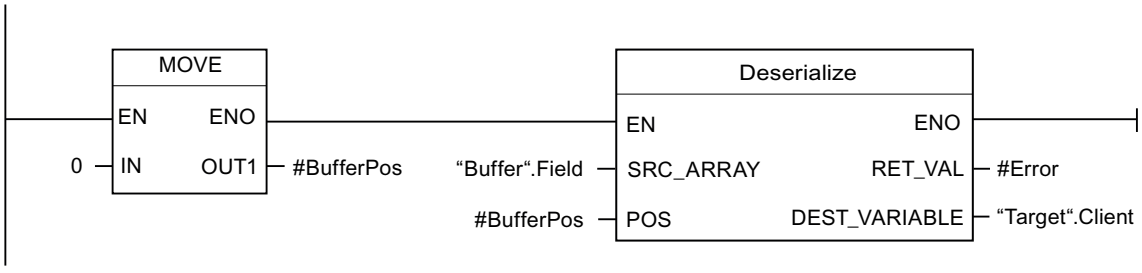

Die Anweisung "Wert kopieren" kopiert den Wert "0" in den Operanden #BufferPos. Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform der Kundendaten aus der Variable "Buffer" und schreibt sie in die Variable "Target". Der Operand #BufferPos speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen.

Netzwerk 2:

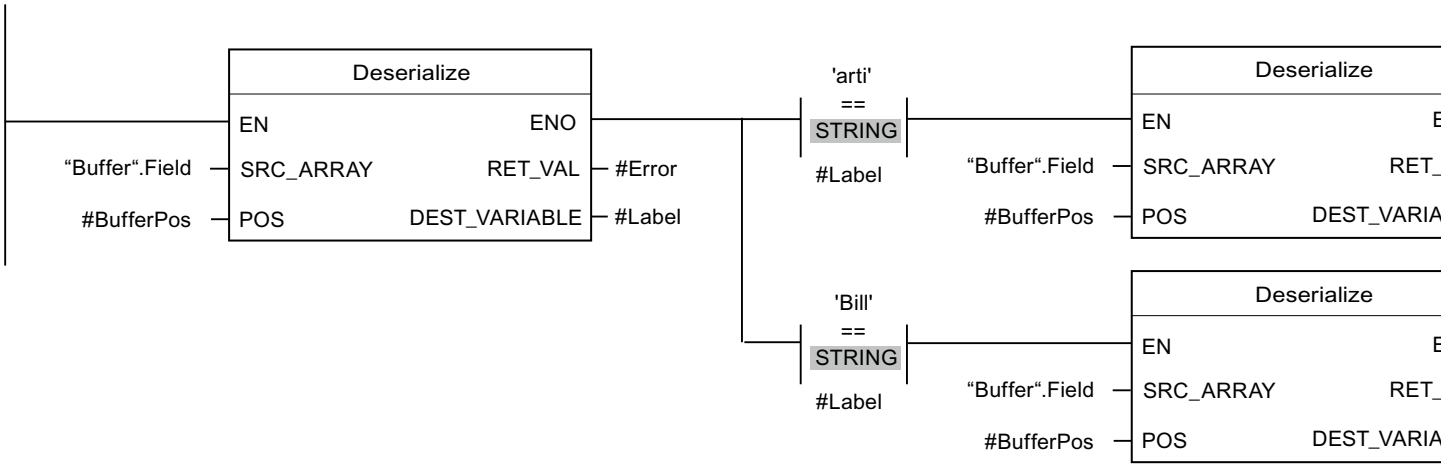

Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform des Trennblatts, das nach den Kundendaten in der sequenziellen Darstellungsform abgelegt wurde, aus der Variable "Buffer" und schreibt die Zeichen in den Operanden #Label. Die Zeichen werden mit Hilfe von Vergleichsanweisungen auf "arti" und "Bill" verglichen. Ist der Vergleich auf "arti" = TRUE, dann handelt es sich Artikeldaten, die deserialisiert und in die Variable "Target" geschrieben werden. Ist der Vergleich auf "Bill" = TRUE, dann handelt es sich um Rechnungsdaten, die deserialisiert und in die Variable "Target" geschrieben werden.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu KOP (Seite 7425)

Serialize: Serialisieren

Beschreibung

Mit der Anweisung "Serialisieren" wandeln Sie mehrere PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> in eine sequenzielle Darstellungsform um, ohne dass Teile ihrer Struktur verloren gehen.

Mithilfe der Anweisung können Sie mehrere strukturierte Daten aus Ihrem Programm in einem Puffer, der vorzugsweise in einem globalen Datenbaustein liegt, zwischenspeichern und an eine andere CPU senden. Der Speicherbereich, in dem die umgewandelten Daten abgelegt werden, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Daten zugelassen. Fülldaten des Datenbereichs der Quelle sind im Ziel-ARRAY undefiniert. Diese können sowohl Füll-Bytes bzw. Füll-Bits eines Datenbereichs (z. B. ARRAY, STRUCT oder PLC-Datentyp (UDT)) sowie die aktuell ungenutzten Zeichen eines Strings sein.

Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Strukturen, die gemäß der Regeln für standard Speicherbereiche größer als 64 KB sind, können nicht serialisiert werden, wenn der Operand am Parameter DEST_ARRAY in einem standard Speicherbereich liegt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Der Operand am Parameter POS enthält die Information, wie viele Bytes die Daten belegen.

Wenn Sie einen einzelnen PLC-Datentyp (UDT), STRUCT oder ARRAY of <Datentyp> versenden möchten, dann können Sie direkt die Anweisung "TSEND: Daten über Kommunikationsverbindung senden" aufrufen.

Hinweis

Vergleich von Strukturen

Um Strukturen zu vergleichen ist es nicht notwendig, diese zuerst zu serialisieren. Verwenden Sie stattdessen die Anweisungen "CMP".

Weitere Informationen finden Sie im Kapitel "Vergleicher".

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hängt die Länge eines BOOL davon ab, welcher Datentyp folgt. Das bedeutet, wenn z. B. ein BYTE folgt, hat das BOOL eine Länge von 1 Byte. Wenn z. B. ein WORD folgt, dann hat das BOOL eine Länge von 2 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Daher ist es empfehlenswert, dass der Quelldatenbereich für das Serialisieren mit den großen Datentypen beginnt und mit booleschen Elementen endet. Dadurch wird das Auffüllen mit Füll-Bits wesentllich reduziert.

Optimierter Speicherbereich

Um größere Strukturen zu serialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich auch mit optimiertem Zugriff deklarieren. Die sequenzielle Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Serialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC_VARIABLE	Input	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die serialisiert werden soll.
DEST_ARRAY	InOut	VARIANT	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der erzeugte Datenstrom gespeichert wird.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Gesamtzahl der Bytes, die die umgewandelten Kundendaten belegt haben. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_VARIABLE und DEST_ARRAY überschneiden sich.
8150	Der Datentyp VARIANT am Parameter SRC_VARIABLE enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_VARIABLE
8236	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.

Fehler-code* (W#16#...)	Erläuterung
8251	Keine gültige Referenz am Parameter DEST_ARRAY
8253	Die Variable am Parameter DEST_ARRAY bietet für den Inhalt der Variable am Parameter SRC_VARIABLE nicht genügend Speicherplatz. Durch den Eingangswert der Variable am Parameter POS wird der zur Verfügung stehende Speicherplatz reduziert. Der Eingangswert des Parameters POS bestimmt, wo innerhalb der Variable am Parameter DEST_ARRAY begonnen wird.
8254	Ungültiger Datentyp am Parameter DEST_ARRAY
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8236	Der Zugriff am Parameter DEST_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

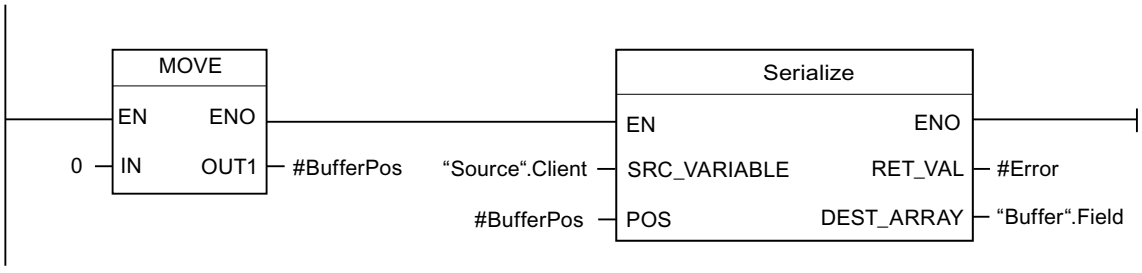

Die Anweisung "Wert kopieren" kopiert den Wert "0" in den Operanden #BufferPos. Die Anweisung "Serialisieren" serialisiert die Kundendaten aus der Variable "Source" und schreibt sie als sequenzielle Darstellung in die Variable "Buffer". Der Index des nächsten unbeschriebenen Bytes des Operanden "Buffer".Field wird im Operanden #BufferPos gespeichert.

Netzwerk 2:

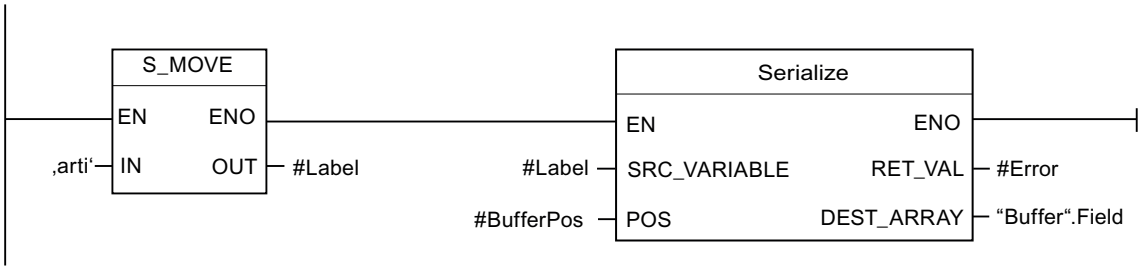

Um die sequenzielle Darstellungsform später wieder leichter deserialisieren zu können, wird jetzt eine Art Trennblatt eingefügt. Die Anweisung "Zeichenkette verschieben" kopiert die Zeichen "arti" in den Operanden #Label. Die Anweisung "Serialisieren" schreibt diese Zeichen nach den Kundendaten in die Variable "Buffer". Der Wert des Operanden #BufferPos wird entsprechend erhöht.

Netzwerk 3:

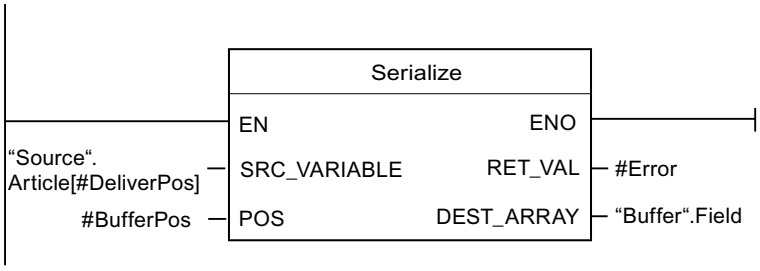

Die Anweisung "Serialisieren" serialisiert die Daten eines bestimmten Artikels, der zur Laufzeit berechnet wird, aus der Variable "Source" und schreibt sie als sequenzielle Darstellung nach den Zeichen "arti" in die Variable "Buffer".

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle
Error	INT	Im Abschnitt "Temp" der Bausteinschnittstelle
Label	STRING[4]	Im Abschnitt "Temp" der Bausteinschnittstelle

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Source	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
Buffer	Field	Array[0..294] of BYTE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu KOP (Seite 7425)

MOVE_BLK: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Eingang COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die

Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden.
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

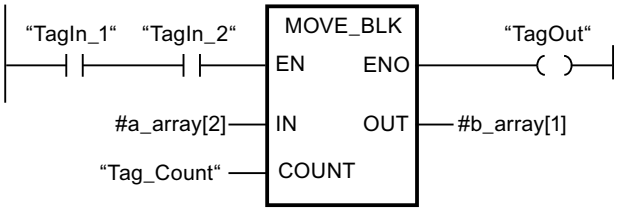

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MOVE_BLK_VARIANT: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können ein komplettes ARRAY oder Elemente eines ARRAYs in ein anderes ARRAY des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-ARRAY dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines ARRAYs umkopieren oder einzelne Elemente kopieren.

Die Anzahl der Elemente, die kopiert werden sollen, darf den selektierten Quellbereich bzw. Zielbereich nicht überschreiten.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss das ARRAY noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC_INDEX und DEST_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des ARRAYs.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als zur Verfügung gestellt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC	Input [2)]	VARIANT (der auf ein ARRAY oder ein einzelnes ARRAY-Element zeigt), ARRAY of <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Der Quellbereich, aus dem kopiert wird
COUNT	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Elemente, die kopiert werden Geben Sie dem Parameter COUNT den Wert "1", wenn am Parameter SRC oder am Parameter DEST kein ARRAY angegeben ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_INDEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert das erste Element, das kopiert wird: • Der Parameter SRC_INDEX wird nullbasiert berechnet. Wenn am Parameter SRC ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter SRC_INDEX das erste Element innerhalb des Quellbereichs an, aus dem kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter SRC kein ARRAY oder nur ein einzelnes Element eines ARRAYs angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert "0".
DEST_IN-DEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert den Anfang des Ziel-speicherbereichs: • Der Parameter DEST_INDEX wird nullbasiert berechnet. Wenn am Parameter DEST ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter DEST_INDEX das erste Element innerhalb des Zielbereichs an, in das kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter DEST kein ARRAY angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zielbereich, in den die Inhalte des Quellbereichs kopiert werden
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter DEST ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden. 2) Am Parameter SRC sind die Datentypen BOOL und ARRAY of BOOL nicht zulässig.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Die Datentypen stimmen nicht überein. Verwenden Sie statt eines ARRAY of Struct ein ARRAY of PLC-Datentyp(UDT).
8151	Der Zugriff auf den Parameter SRC ist nicht möglich.
8152	Der Operand am Parameter SRC ist nicht typisiert.
8153	Codeerzeugungsfehler am Parameter SRC
8154	Der Operand am Parameter SRC hat den Datentyp BOOL.
8281	Ungültiger Wert am Parameter COUNT
8382	Der Wert am Parameter SRC_INDEX ist kleiner NULL.
8383	Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8482	Der Wert am Parameter DEST_INDEX ist kleiner NULL.
8483	Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8534	Der Parameter DEST ist schreibgeschützt
8551	Der Zugriff auf den Parameter DEST ist nicht möglich.
8552	Der Operand am Parameter DEST ist nicht typisiert.
8553	Codeerzeugungsfehler am Parameter DEST
8554	Der Operand am Parameter DEST hat den Datentyp BOOL.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

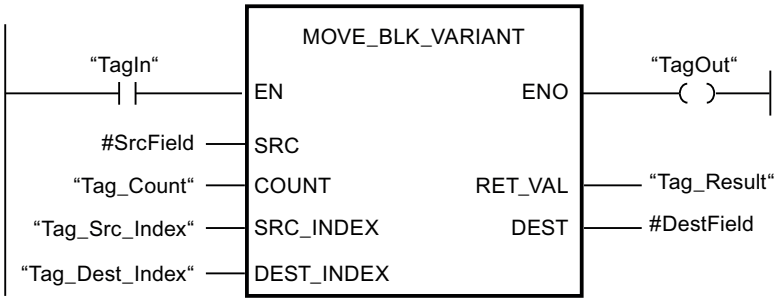

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
SRC	Input	#SrcField	Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[0..10] of "MOVE_UDT"
COUNT	Input	Tag_Count	2
SRC_INDEX	Input	Tag_Src_Index	3
DEST_INDEX	Input	Tag_Dest_Index	3
DEST	InOut	#DestField	Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[10..20] of "MOVE_UDT"

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. 2 Elemente werden aus dem Quellbereich, beginnend mit dem vierten Element des ARRAY[0..10] of MOVE_UDT, in den Zielbereich kopiert. Die Kopien werden im ARRAY[10..20] of MOVE_UDT ab dem vierten Element eingefügt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK_VARIANT im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

VariantGet: Wert einer VARIANT-Variable auslesen (Seite 854)

Beispiel für das Verschieben von Daten (Seite 348)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

UMOVE_BLK: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN [1)]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden.

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

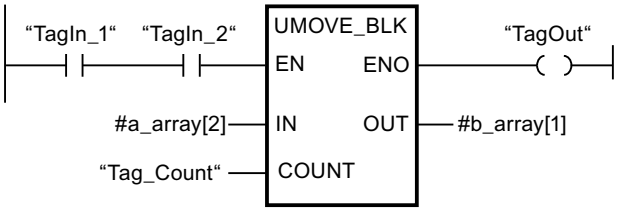

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Hinweis

Weitere Informationen zur Anweisung UMOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

FILL_BLK: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die

Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Adresse im Zielbereich, ab der befüllt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

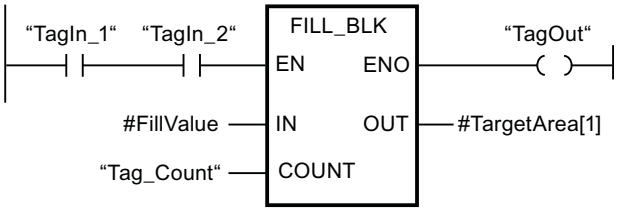

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1		Datentyp
MyStruct1		STRUCT
	Member_1	INT
	Member_2	INT
	Member_3	INT
	Member_4	INT
MyStruct2		STRUCT

Data_block_1					Datentyp
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

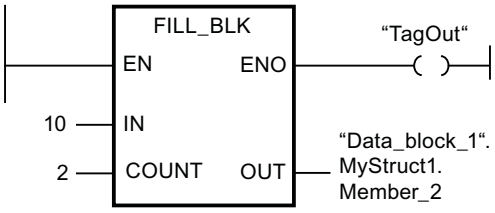

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

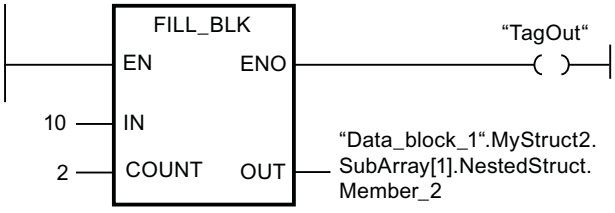

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

UFILL_BLK: Bereich ununterbrechbar befüllen

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird.
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Adresse im Zielbereich, ab der befüllt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

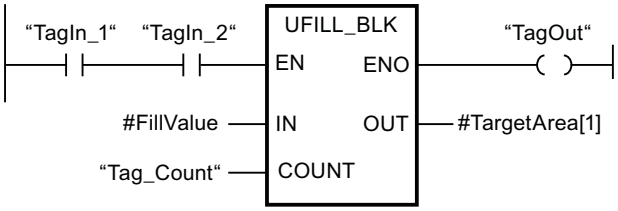

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

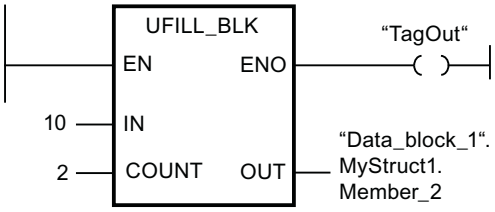

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

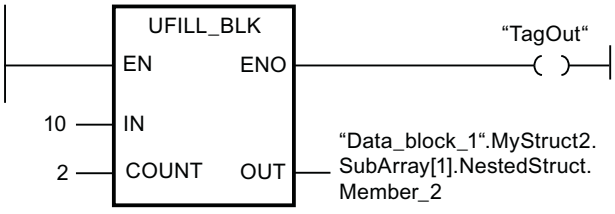

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SCATTER: Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Bitfolge in einzelne Bits zerlegen" zerlegen Sie eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Bitfolge in einzelne Bits zerlegen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der anonyme STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. ein Statuswort zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der STRUCT oder der PLC-Datentyp nicht genügend BOOL-Elemente zur Verfügung stellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Bitfolge, die zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das bzw. den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceWord		WORD
EnableOut	Output	BOOL
DestinationArray		ARRAY[0..15] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

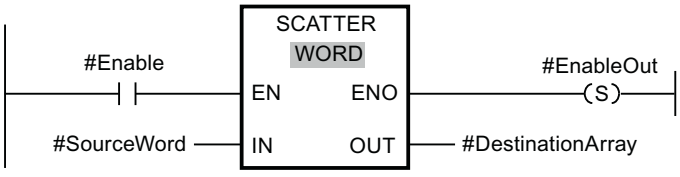

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationArray	Der Operand "DestinationArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationArray zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel mit einem PLC-Datentyp

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceWord		WORD
EnableOut	Output	BOOL
DestinationUDT		"myBits"

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

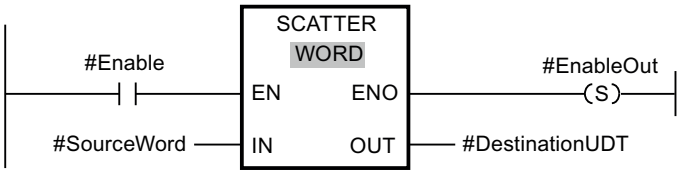

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationUDT	Der Operand "DestinationUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationUDT zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen" zerlegen Sie ein oder mehrere Elemente eines ARRAY of BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen. Am Parameter COUNT_IN geben Sie an, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Das Quell-ARRAY am Parameter IN darf mehr Elemente haben, als am Parameter COUNT_IN angegeben sind. Das ARRAY of BOOL, der anonyme STRUCT oder der PLC-Datentyp muss genügend Elemente haben, um die Bits der zerlegten Bitfolgen speichern zu können. Der Zielspeicherbereich darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter IN ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter OUT ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Das ARRAY of WORD wird in 48 einzelne Bits zerlegt.

Hinweis

Wenn die untere ARRAY-Grenze des Ziel-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. Statuswörter zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Quell-ARRAY hat weniger Elemente als am Parameter COUNT_IN angegeben sind.
- Der Index des Ziel-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of BOOL geschrieben.
- Das ARRAY[*] of BOOL, der STRUCT oder der PLC-Datentyp stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zerlegt und in das ARRAY of BOOL, den anonymen STRUCT oder den PLC-Datentyp geschrieben. Die restlichen Bitfolgen werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, das zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
COUNT_IN	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das/den die einzelnen Bits gespeichert werden

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayWord		ARRAY[0..5] of WORD
CounterInput		UDINT
EnableOut	Output	BOOL
DestinationArrayBool		ARRAY[0..95] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

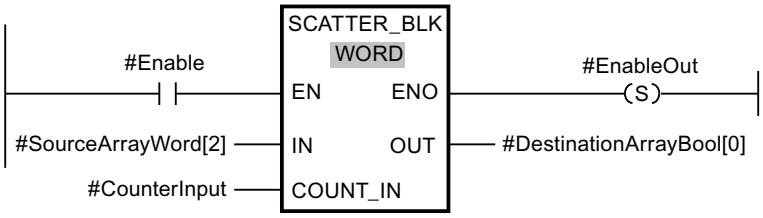

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[0]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 1. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayWord		ARRAY[0..5] of WORD
CounterInput		UDINT
EnableOut	Output	BOOL
DestinationArrayBool		ARRAY[-2..93] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

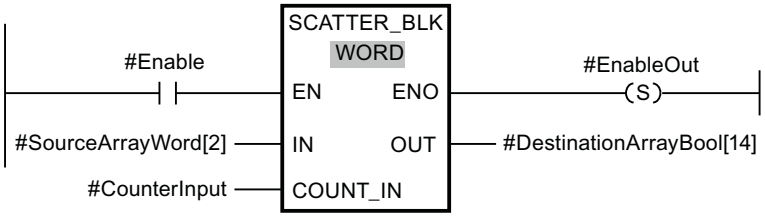

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[14]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 16. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0". Die restlichen 32 Bits werden nicht beschrieben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einer Bitfolge zusammen. Die Bitfolge wird in einer Variable vom Datentyp BYTE, WORD, DWORD oder LWORD gespeichert.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der anonyme STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der anonyme STRUCT oder der PLC-Datentyp (UDT) weniger oder mehr BOOL-Elemente hat als die Bitfolge vorgibt, dann werden sie nicht übertragen und das ENO liefert den Signalzustand "0". Das ENO ist ebenfalls "0", wenn weniger als die notwendige Anzahl an Bits vorhanden ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, dessen Bits zu einer Bitfolge zusammengefügt werden. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Zusammengefügte Bitfolge, gespeichert in einer Variable

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArray		ARRAY[0..15] of BOOL
EnableOut	Output	BOOL
DestinationWord		WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

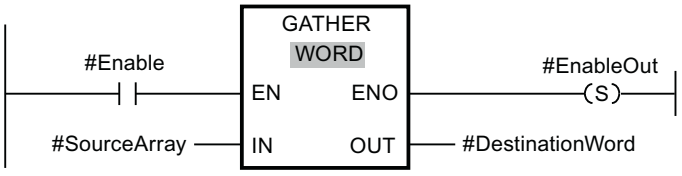

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArray	Der Operand "SourceArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Die Bits des Operanden #SourceArray werden zu einem WORD zusammengefügt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceUDT		"myBits"
EnableOut	Output	BOOL
DestinationWord		WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

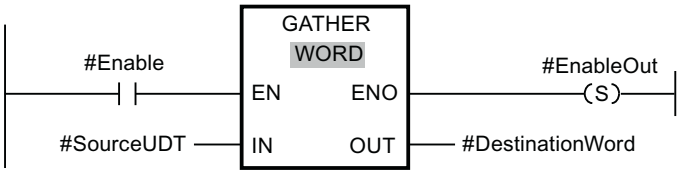

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceUDT	Der Operand "SourceUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Die Bits des Operanden #SourceUDT werden zu einem WORD

zusammengefügt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einem oder mehreren Elementen eines ARRAY of <Bitfolge> zusammen. Am Parameter COUNT_OUT geben Sie an, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Damit geben Sie auch implizit vor, wie viele Elemente des ARRAY of BOOL, des anonymen STRUCTs oder des PLC-Datentyps benötigt werden. Das Ziel-ARRAY am Parameter OUT darf mehr Elemente haben, als am Parameter COUNT_OUT angegeben sind. Das ARRAY of <Bitfolge> muss genügend Elemente haben, um die Bits, die zusammengefügt werden sollen, speichern zu können. Das Ziel-ARRAY darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter OUT ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter IN ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Aus dem ARRAY of BOOL werden 48 einzelne Bits zusammengefügt.

Hinweis

Wenn die untere ARRAY-Grenze des Quell-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Index des Quell-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of <Bitfolge> geschrieben.
- Das ARRAY[*] of <Bitfolge> stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zusammengefügt und in das ARRAY of <Bitfolge> geschrieben. Die restlichen Bits werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY of BOOL, STRUCT oder PLC-Datentyp, dessen Bits zusammengefügt werden (Quell-ARRAY). Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_OUT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, in das die Bits gespeichert werden (Ziel-ARRAY)

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayBool		ARRAY[0..95] of BOOL
CounterOutput		UDINT
EnableOut	Output	BOOL
DestinationArrayWord		ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

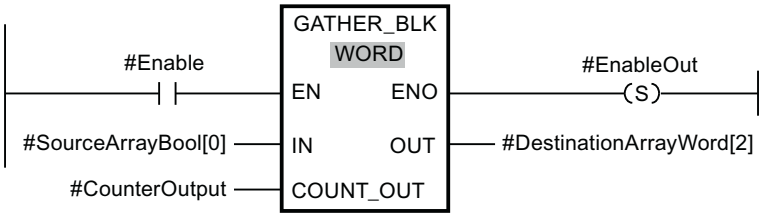

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[0]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 1. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits in das 3. Wort, die zweiten 16 Bits in das 4. Wort und die dritten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayBool		ARRAY[-2..93] of BOOL
CounterOutput		UDINT
EnableOut	Output	BOOL
DestinationArrayWord		ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

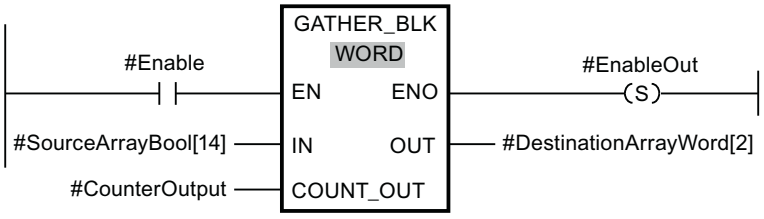

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[14]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Da aber erst ab dem 16. Element begonnen wird, stehen nur 80 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 16. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits des Quell-ARRAYs nicht berücksichtigt werden. Die zweiten 16. Bits werden in das 3. Wort, die dritten 16 Bits in das 4. Wort und die vierten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Die restlichen 64 Bits des Quell-ARRAYs werden ebenfalls nicht berücksichtigt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

AssignmentAttempt: Zuweisung eines VARIANT auf eine Referenz versuchen

Beschreibung

Mit der Anweisung "AssignmentAttempt" versuchen Sie die Zuweisung einer VARIANT-Variable auf eine Referenzvariable. Der Datentyp einer Referenzvariable wird zum Zeitpunkt der Deklaration festgelegt, während der Datentyp einer VARIANT-Variable während der Laufzeit ermittelt wird. Eine implizite Datentypkonvertierung ist bei Referenzvariablen nicht zulässig. Um die beiden Datentypen einander zuzuweisen, nutzen Sie deshalb den Zuweisungsversuch.

Beim Zuweisungsversuch wird zur Laufzeit geprüft, ob die VARIANT-Variable vom richtigen Datentyp ist. Wenn dies der Fall ist, wird die Zuweisung ausgeführt. Nach einer erfolgreichen Ausführung steht eine gültige Referenz in der Zielvariablen, andernfalls eine NULL.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL		Freigabeeingang
ENO	Output	BOOL		Freigabeausgang
SRC	Input	VARIANT	• Bausteinschnittstelle eines FC: Input, Output, InOut, Temp • Bausteinschnittstelle eines FB: Input, InOut, Temp	Zeiger auf die Quellvariable, deren Adresse ausgelesen wird
DST	Output	Referenz auf: • Bitfolgen, außer BOOL, • Ganzzahlen, • Gleitpunktzahlen, • Zeichenfolgen, • PLC-Datentypen (UDT), • Systemdatentypen (SDT), • ARRAYs aus den genannten Datentypen	• Bausteinschnittstelle eines FC: Input, Output, Temp, Return • Bausteinschnittstelle eines FB: Temp	Referenz, an die die Adresse der Quellvariable übergeben wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Regeln

Für den Zuweisungsversuch gelten die folgenden Regeln. VARIANT-Variablen, die nicht diesen Regeln entsprechen, liefern zur Laufzeit den Wert "NULL" zurück.

- Der VARIANT muss auf eine Adresse in einem optimierten Speicherbereich zeigen.
- Der VARIANT darf nicht auf eine Adresse in einem temporären Speicherberiech zeigen.
- Wenn Sie einer Referenz auf ein ARRAY einen VARIANT zuweisen wollen, gelten folgende Regeln:
  - Die VARIANT-Variable muss auf ein ARRAY zeigen, dessen Grenzen exakt mit denen der deklarierten Referenz übereinstimmen. Eine VARIANT-Variable, die auf ein ARRAY [0..9] zeigt, passt nicht zu einer Variable REF_TO ARRAY[1..10].
  - Darüberhinaus sollten Sie die Bausteine, die den Wert der VARIANT-Variable bilden, einmalig in einer CPU der Baureihe S7-1500, Firmwarestand V2.5, übersetzen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

In der Schnittstelle des Bausteins wurden der VARIANT "myVariant" und die Referenzvariable "myReference" deklariert.

	Name	Data type
	Input	
	Output	
	InOut	
	myVariant	Variant
	Static	
	Temp	
	myReference	REF_TO Int
	Constant	

Im Programmcode wird versucht "myVariant" der Referenzvariable "myReference" zuzuweisen. Hat "myVariant" zur Laufzeit den Datentyp "Int", steht in "myReference" eine gültige Referenz auf die Quellvariable des VARIANT, andernfalls eine NULL. Wenn der Zuweisungsversuch erfolgreich war, führt der Freigabeausgang "ENO" Signalzustand "1" und der Wert "10" kann in die Zielvariable geschrieben werden.

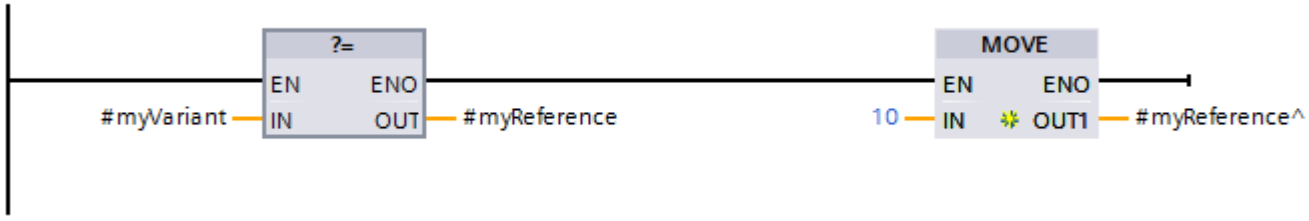

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Zuweisungsversuch von VARIANT auf eine Referenz (Seite 329)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Sample Library for Instructions (Seite 615)

Grundlagen zu KOP (Seite 7425)

SWAP: Anordnung ändern

Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes am Eingang IN und fragt das Ergebnis am Ausgang OUT ab.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" vertauscht werden:

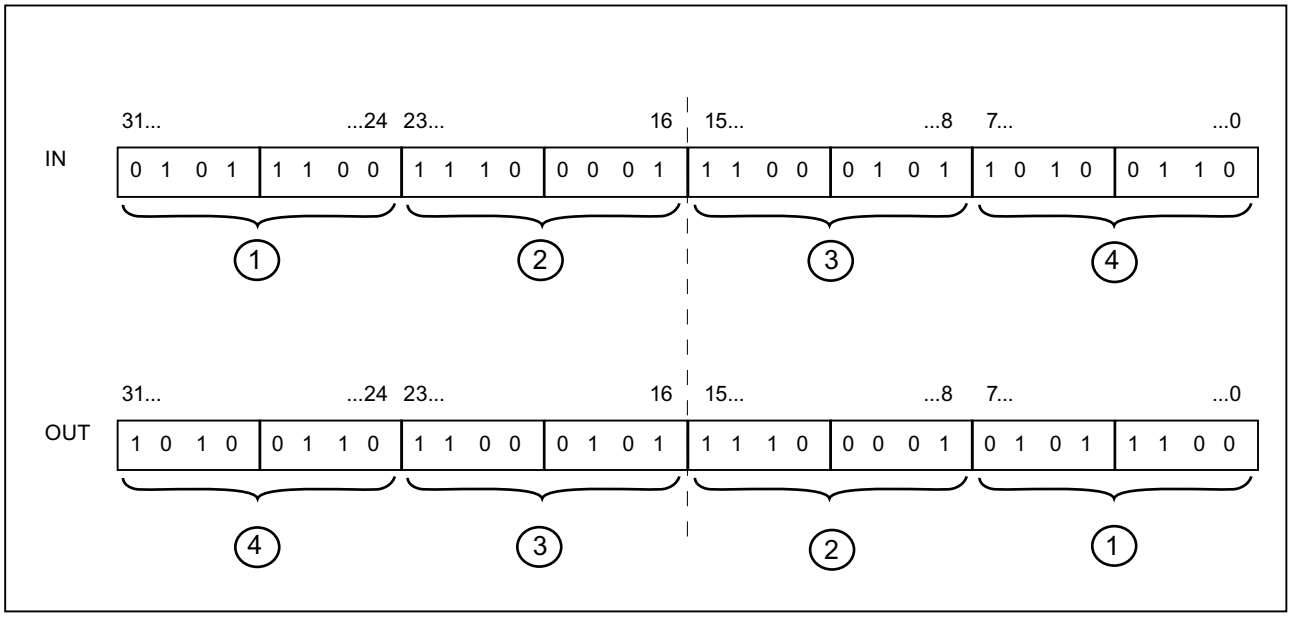

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anordnung ändern":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L oder, P Konstante	Operand, dessen Bytes vertauscht werden.
OUT	Output	WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

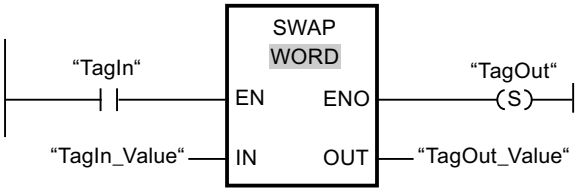

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0000 1111 0101 0101
OUT	TagOut_Value	0101 0101 0000 1111

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Anordnung ändern" ausgeführt. Die Anordnung der Bytes wird geändert und im Operanden "TagOut_Value" abgelegt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ARRAY-DB

ReadFromArrayDB: Aus Array-Datenbaustein lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie das Element aus einem Datenbaustein vom Bausteintyp ARRAY-DB, auf das der Index verweist, und schreiben den Wert des Elements in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Der Freigabeausgang ENO liefert den Signalzustand FALSE, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand FALSE.
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der gelesen und ausgegeben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter VALUE ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz, schreibgeschützt oder liegt im Ladespeicher.
8135	Der ARRAY-Datenbaustein enthält ungültige Werte.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8452	Codeerzeugungsfehler
8453	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

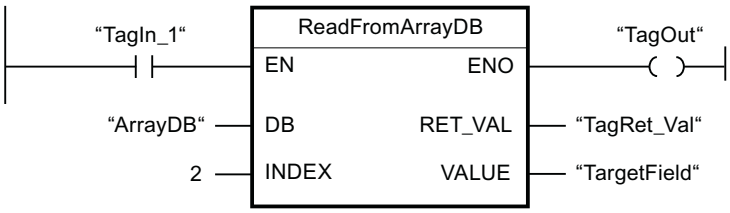

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.

Wenn der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und in den Operanden "TargetField" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

WriteToArrayDB: In Array-Datenbaustein schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Der Freigabeausgang ENO liefert den Signalzustand FALSE, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand FALSE.
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der geschrieben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8134	Der Datenbaustein ist schreibgeschützt.
8135	Der Datenbaustein ist kein ARRAY-Datenbaustein.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8350	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8352	Codeerzeugungsfehler

Fehler-code* (W#16#...)	Erläuterung
8353	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

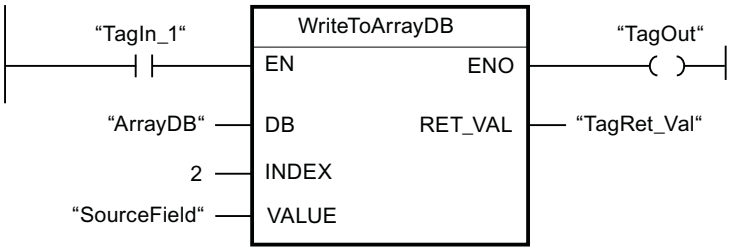

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Wenn der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "SourceField" wird in das 2. Element des ARRAY-DB geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie aus einem Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher das Element, auf das der Index verweist, und schreiben sie in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Der Freigabeausgang ENO liefert den Signalzustand FALSE, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand FALSE.
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Auslesen des ARRAY-DBs beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, aus dem gelesen wird

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1]	InOut	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Auslesen des ARRAY-DBs ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8750	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler

Fehler-code* (W#16#...)	Erläuterung
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

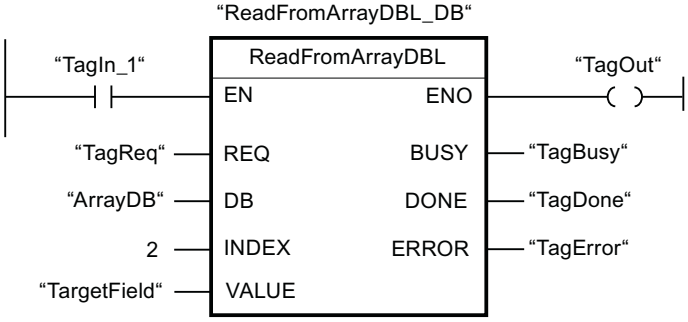

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn der Operand "TagIn1" den Signalzustand "1" liefert und am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und am Parameter VALUE ausgegeben. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird nicht mehr verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu KOP (Seite 7425)

WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Schreiben in den Array-DB beginnen
DB [1]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1]	Input	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.

Fehler-code* (W#16#...)	Erläuterung
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8234	Der Datenbaustein ist schreibgeschützt.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

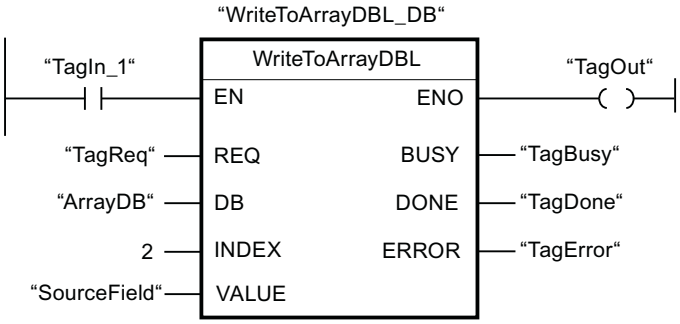

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Parameter	Operand	Wert
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn der Operand "TagIn1" den Signalzustand "1" liefert und am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird in das 2. Element des "ArrayDB" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu KOP (Seite 7425)

VARIANT

VariantGet: Wert einer VARIANT-Variable auslesen

Beschreibung

Mit der Anweisung "Wert einer VARIANT-Variable auslesen" lesen Sie den Wert der Variable aus, auf die der VARIANT am Parameter SRC zeigt, und schreiben diesen in die Variable am Parameter DST.

Der Parameter SRC hat den Datentyp VARIANT. Am Parameter DST kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable, die am Parameter DST angegen ist, muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Datentypen stimmen nicht überein. (Es wird kein Wert übertragen.)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert einer VARIANT-Variable auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
SRC	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Variable, die ausgelesen wird
DST	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

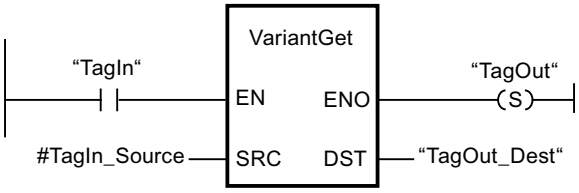

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen und in den Operand "TagOut_Dest" geschrieben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 807)

Grundlagen zu KOP (Seite 7425)

VariantPut: Wert in eine VARIANT-Variable schreiben

Beschreibung

Mit der Anweisung "Wert in eine VARIANT-Variable schreiben" schreiben Sie den Wert der Variable, die am Parameter SRC angegeben ist, in den Speicher am Parameter DST, auf den der VARIANT zeigt.

Der Parameter DST hat den Datentyp VARIANT. Am Parameter SRC kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable am Parameter SRC muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Datentypen stimmen nicht überein. (Es wird kein Wert übertragen.)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert in eine VARIANT-Variable schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
SRC	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Variable, die ausgelesen wird
DST	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

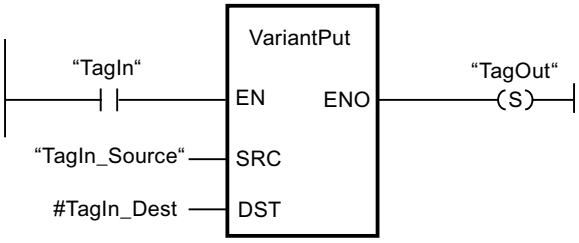

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "TagIn_Source" wird in die Variable, auf die der VARIANT am Operand #TagIn_Dest zeigt, geschrieben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 807)

Grundlagen zu KOP (Seite 7425)

CountOfElements: Anzahl der ARRAY-Elemente abfragen

Beschreibung

Mit der Anweisung "Anzahl der ARRAY-Elemente abfragen" fragen Sie ab, wie viele ARRAY-Elemente eine Variable hat, auf die der VARIANT zeigt.

Wenn es sich um ein eindimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl der ARRAY-Elemente ausgegeben. (Die Differenz zwischen dem oberen und dem unteren Grenzwert + 1). Wenn es sich um ein mehrdimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl aller Dimensionen ausgegeben.

Wenn Sie die Elemente eines ARRAY-DB abfragen möchten, dann sollten Sie die Anweisungen "ReadFromArrayDB" bzw. "WriteFromArrayDB" verwenden, da hier eine genauere Fehlerauswertung für die Anzahl der Elemente möglich ist.

Hinweis

Instanzen

Der VARIANT-Zeiger kann auf keine Instanz und damit auch auf keine Multiinstanz oder ein ARRAY of Multiinstanzen zeigen.

Hinweis

ARRAY innerhalb eines Datenbausteins

Wenn Sie die Anzahl der Elemente eines ARRAYs, das in einem Datenbaustein liegt, abfragen möchten, dann darf bei diesem Datenbaustein nicht das Bausteinattribut "Datenbaustein im Gerät schreibgeschützt" aktiviert sein. Ansonsten liefert der Parameter RET_VAL das Ergebnis "0", unabhängig davon, wie viele Elemente das ARRAY enthält.

Das Ergebnis ist ebenfalls "0", wenn die VARIANT-Variable kein ARRAY ist.

Wenn der VARIANT auf ein ARRAY of BOOL zeigt, dann werden auch die Füllelemente mitgezählt. (Z. B. bei einem ARRAY[0..1] of BOOL wird 8 zurückgegeben).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die VARIANT-Variable ist kein ARRAY. (Das Ergebnis ist "0".)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der ARRAY-Elemente abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die abgefragt wird
RET_VAL	Output	UDINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

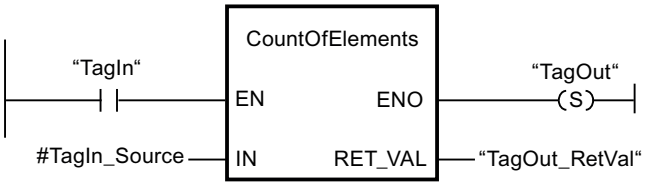

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anzahl der ARRAY-Elemente der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen und am Operand "TagOut_RetVal" ausgegeben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ARRAY[*]

LOWER_BOUND: Untere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, untere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Untere ARRAY-Grenze auslesen" zur Verfügung.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Dimension, die am Eingang DIM angegeben wurde, existiert nicht.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Untere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, untere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, untere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

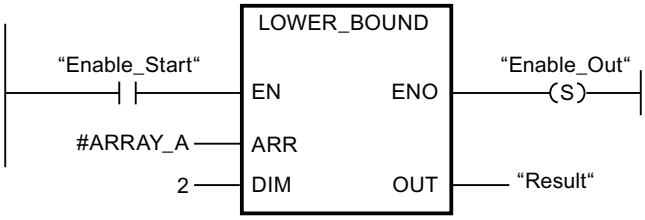

Wenn der Operand "Enable_Start" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Sie liest die variable, untere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben und der Operand "Enable_Out" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu KOP (Seite 7425)

UPPER_BOUND: Obere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, obere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Obere ARRAY-Grenze auslesen" zur Verfügung.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Dimension, die am Eingang DIM angegeben wurde, existiert nicht.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Obere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, obere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, obere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

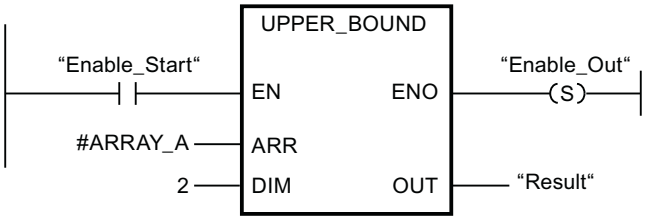

Wenn der Operand "Enable_Start" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Sie liest die variable, obere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben und der Operand "Enable_Out" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu KOP (Seite 7425)

Legacy

FieldRead: Feld lesen

Beschreibung

Mit der Anweisung "Feld lesen" lesen Sie eine bestimmte Komponente aus dem am Eingang MEMBER angegebenen Feld aus und übertragen deren Inhalt in die Variable am Ausgang VALUE. Den Index der zu lesenden Feldkomponente legen Sie am Eingang INDEX fest. Am Eingang MEMBER geben Sie die erste Komponente des Felds an, aus dem gelesen wird.

Die Datentypen der Feldkomponente am Parameter MEMBER, des Index und der Variablen am Parameter VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen, da eine implizite Konvertierung nicht möglich ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Eingang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld lesen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
INDEX	Input	DINT	DINT	E, A, M, D, L, P oder Konstante	Index der Feldkomponente, deren Inhalt ausgelesen wird.
MEMBER	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	D, L	Erste Komponente des Felds, aus dem gelesen wird.
VALUE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR, WCHAR	E, A, M, D, L, P	Operand, in den der Inhalt der Feldkomponente übertragen wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

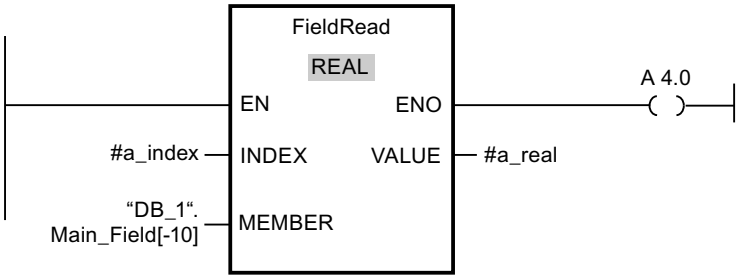

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Variable	Wert
INDEX	a_index	4
MEMBER	"DB_1".Main_Field[-10]	Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1"
VALUE	a_real	Komponente mit Index 4 des Felds "Main_Field[-10..10] of REAL"

Die Feldkomponente mit Index 4 wird aus dem Feld "Main_Field[-10...10] of REAL" ausgelesen und in die Variable "a_real" geschrieben. Die zu lesende Feldkomponente wird durch den Wert am Eingang INDEX festgelegt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

FieldWrite: Feld schreiben

Beschreibung

Mit der Anweisung "Feld schreiben" übertragen Sie den Inhalt der Variablen am Eingang VALUE in eine bestimmte Komponente des Felds am Ausgang MEMBER. Den Index der Feldkomponente, die beschrieben wird, legen Sie durch den Wert am Eingang INDEX fest. Am Ausgang MEMBER geben Sie die erste Komponente des Felds an, in das geschrieben wird.

Die Datentypen der Feldkomponente am Parameter MEMBER, des Index und der Variablen am Parameter VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen, da eine implizite Konvertierung nicht möglich ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Ausgang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld schreiben":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
INDEX	Input	DINT	DINT	E, A, M, D, L, P oder Konstante	Index der Feldkomponente, die mit dem Inhalt von VALUE beschrieben wird.
VALUE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD,CHAR, WCHAR	E, A, M, D, L, P oder Konstante	Operand, dessen Inhalt kopiert wird.
MEMBER	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	D, L	Erste Komponente des Felds, in welches der Inhalt von VALUE geschrieben wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

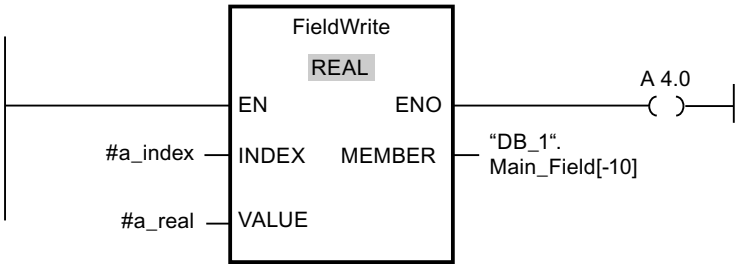

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
INDEX	a_index	4
VALUE	a_real	10.54
MEMBER	"DB_1".Main_Field[-10]	Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1"

Der Wert "10.54" der Variablen "a_real" wird in die Feldkomponente mit Index 4 des Felds "Main_Field[-10..10] of REAL" geschrieben. Der Index der Feldkomponente, in die der Inhalt der Variablen "a_real" übertragen wird, wird durch den Wert am Eingang INDEX festgelegt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

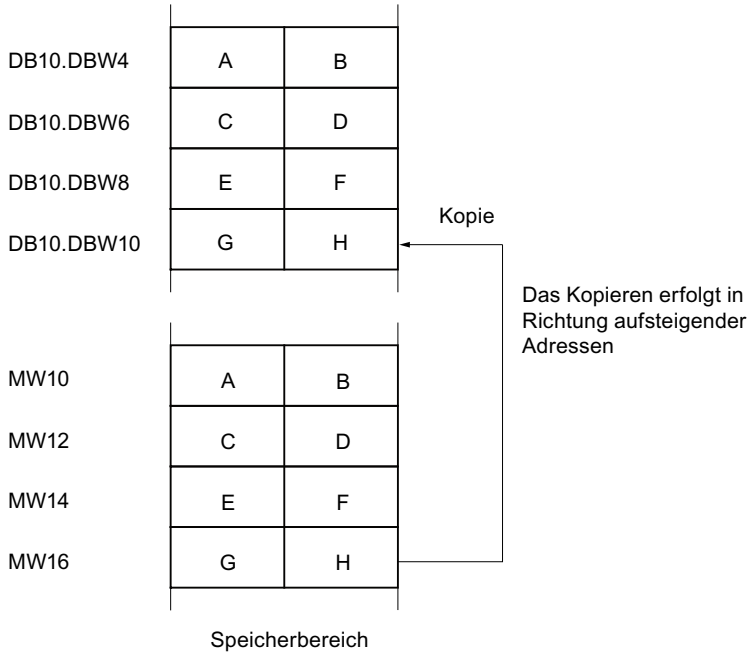

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.

Fehlercode* (W#16#...)	Erläuterung
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

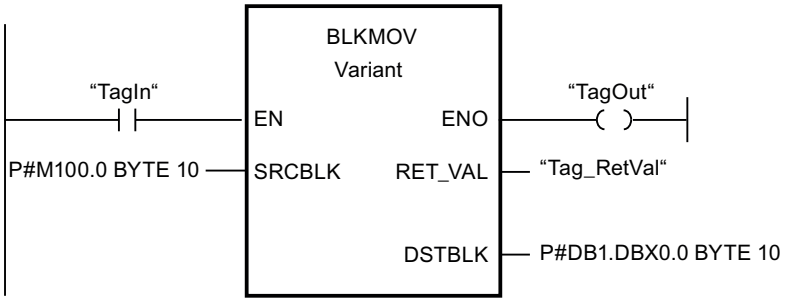

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.

Fehlercode* (W#16#...)	Erläuterung
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

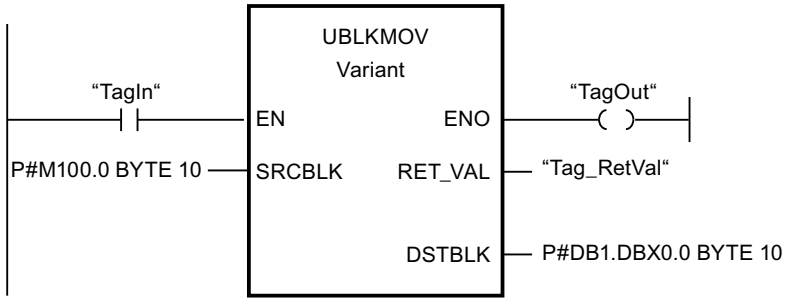

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

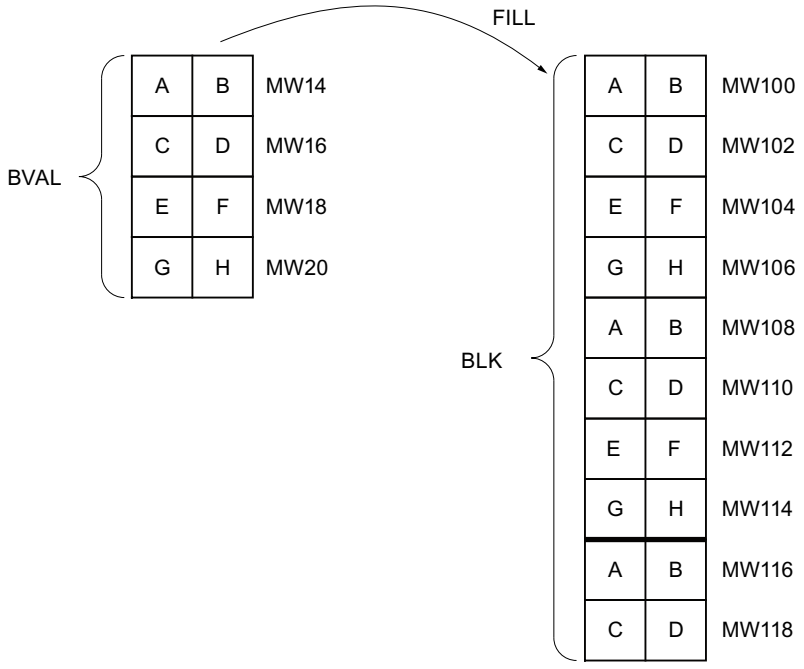

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
BVAL	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter BVAL werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter BLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

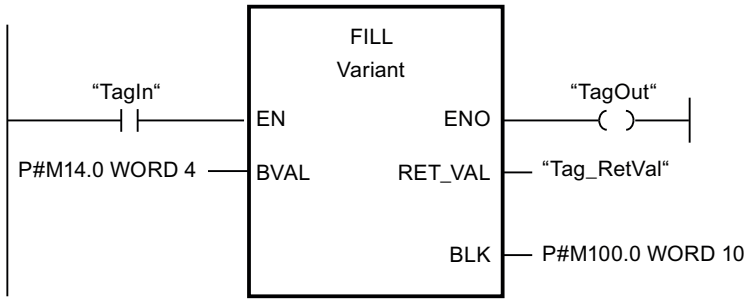

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend den in der Anweisungsbox ausgewählten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Konvertierungsmöglichkeiten von Bitfolgen

Die Bitfolgen BYTE und WORD sind in der Anweisungsbox nicht auswählbar. Es ist allerdings möglich, einen Operanden vom Datentyp DWORD oder LWORD an einem Parameter der Anweisung anzugeben, wenn die Länge des Eingangs- und des Ausgangsoperanden übereinstimmt. Dann wird der Operand vom Datentyp einer Bitfolge entsprechend des Datentyps des Eingangs- oder Ausgangsparameters interpretiert und implizit konvertiert. Der Datentyp DWORD wird z. B. als DINT/UDINT und LWORD als LINT/ULINT interpretiert. Diese Konvertierungsmöglichkeiten stehen Ihnen auch bei aktivierter IEC-Prüfung zur Verfügung.

Hinweis

Für CPUs der Baureihen S7-1500 gilt: Die Datentypen DWORD und LWORD können nur von bzw. nach Datentyp REAL oder LREAL konvertiert werden.

Dabei wird das Bitmuster des Quellwerts ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand des Freigabeausgangs ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand des Freigabeausgangs ENO = 0.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Beispiele

Das folgende Beispiel zeigt die Konvertierung einer Ganzzahl (16 Bit) in eine andere Ganzzahl (32 Bit):

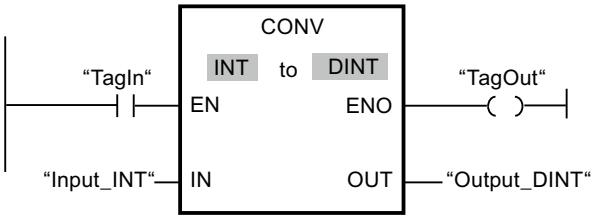

Das folgende Beispiel zeigt die Konvertierung eines Bytes (8 Bit) in die Ganzzahl SINT (8 Bit):

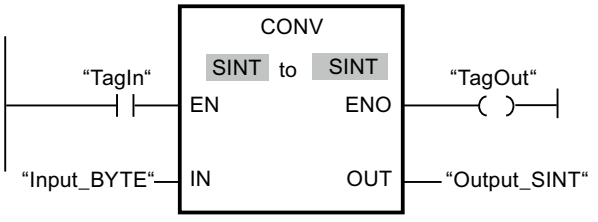

Das folgende Beispiel zeigt die Konvertierung eines Bytes (8 Bit) in die vorzeichenlose Ganzzahl USINT (8 Bit):

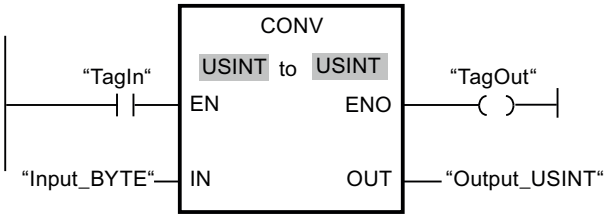

Die Konvertierungen sind möglich, da die Länge der beiden Operanden gleich ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Datentypkonvertierungen bei S7-1200 (Seite 507)

Grundlagen zu KOP (Seite 7425)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl vom Datentyp DINT um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Rundung

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

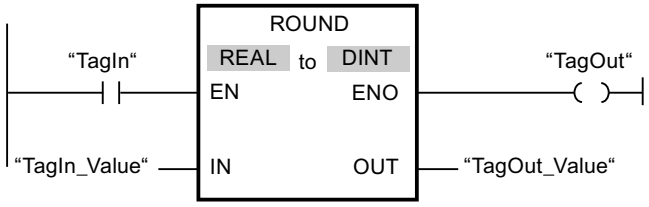

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächsten geraden Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

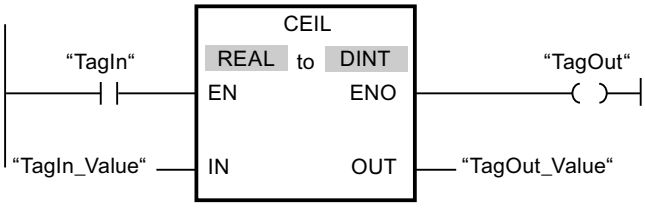

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

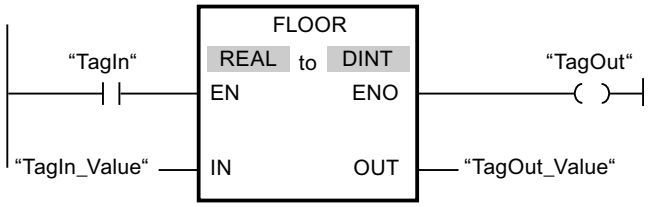

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Eingangwert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Ganzzahliger Teil des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

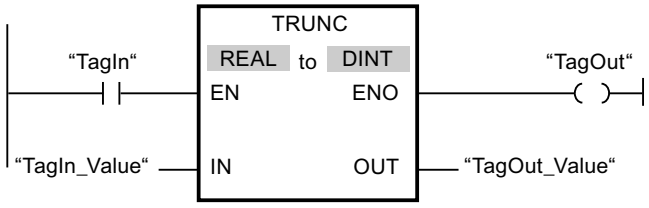

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SCALE_X: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Eingang VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung "Skalieren" wird der Gleitpunktwert am Eingang VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang OUT abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:

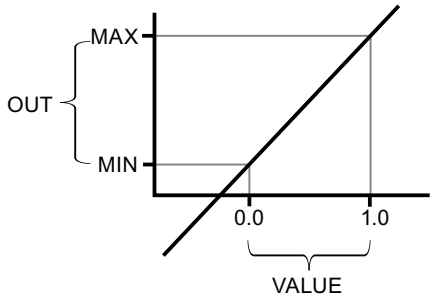

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [VALUE * (MAX – MIN)] + MIN

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Eingang VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALUE	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Wert, der skaliert wird. Bei Angabe einer Konstante müssen Sie diese deklarieren.
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Skalierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

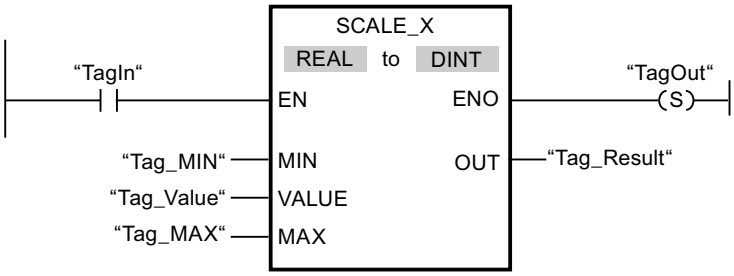

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_MIN	10
VALUE	Tag_Value	0.5
MAX	Tag_MAX	30
OUT	Tag_Result	20

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert am Eingang "Tag_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Eingängen "Tag_MIN" und "Tag_MAX" definiert wurde. Das Ergebnis wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

NORM_X: Normieren

Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert der Ausgang OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert der Ausgang OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:

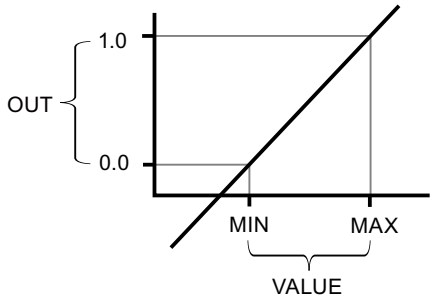

Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

OUT = (VALUE – MIN) / (MAX – MIN)

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.

- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Der Wert am Eingang VALUE ist NaN (Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Normieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MIN [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Wert, der normiert wird.
MAX [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
OUT	Output	Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Normierung
[1)] Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren.				

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

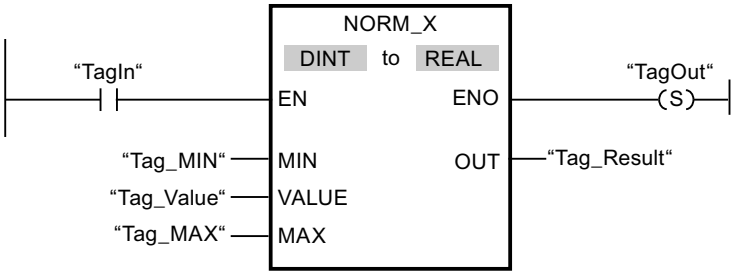

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_MIN	10
VALUE	Tag_Value	20
MAX	Tag_MAX	30
OUT	Tag_Result	0.5

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert am Eingang "Tag_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Eingängen "Tag_MIN" und "Tag_MAX" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Eingang "Tag_Value" normiert. Das Ergebnis wird am Ausgang "Tag_Result" als Gleitpunktzahl abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

Legacy

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1"
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

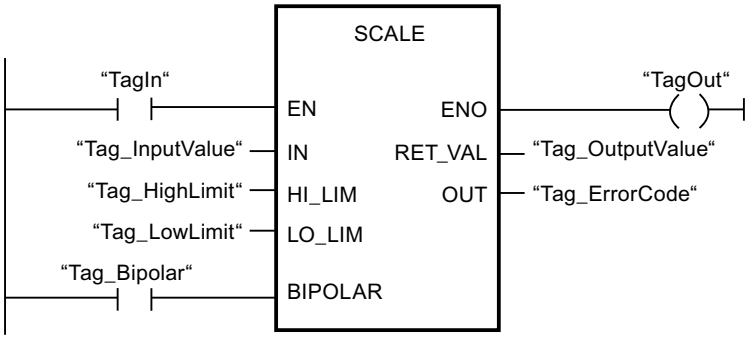

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Input	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

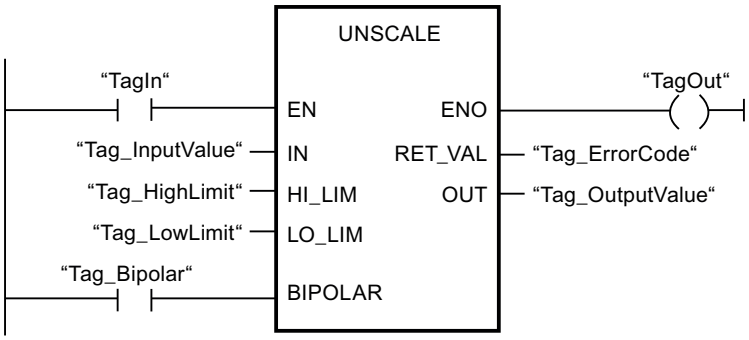

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

Programmsteuerung

---(JMP): Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

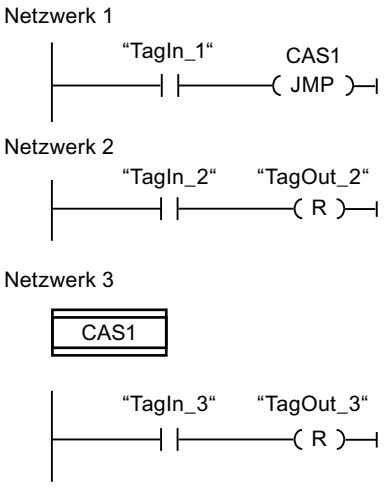

Wenn der Operand "TagIn_1"den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(JMPN): Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das Verknüpfungsergebnis am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

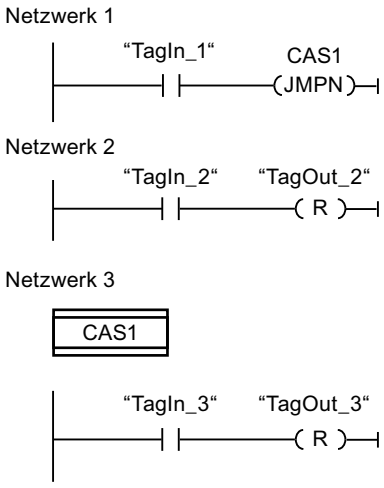

Wenn der Operand "TagIn_1" den Signalzustand "0" liefert, wird die Anweisung "Springen bei VKE = 0" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

LABEL: Sprungmarke

Beschreibung

Mit einer Sprungmarke kennzeichnen Sie ein Zielnetzwerk, in dem bei einem ausgeführten Sprung die Programmbearbeitung fortgesetzt wird.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Sie können maximal 32 Sprungmarken bei Verwendung einer CPU S7-1200 und maximal 256 Sprungmarken bei Verwendung einer CPU S7-1500 deklarieren.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

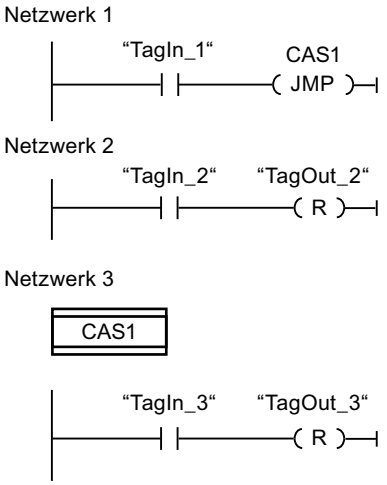

Wenn der Operand "TagIn_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

JMP_LIST: Sprungliste definieren

Beschreibung

Mit der Anweisung "Sprungliste definieren" definieren Sie mehrere bedingte Sprünge und setzen die Programmbearbeitung, abhängig vom Wert des Parameters K, in einem bestimmten Netzwerk fort.

Die Sprünge definieren Sie durch Sprungmarken (LABEL), die Sie an den Ausgängen der Anweisungsbox angeben. Die Anzahl der Ausgänge ist in der Anweisungsbox erweiterbar. Sie können maximal 32 Ausgänge bei Verwendung einer CPU S7-1200 und maximal 256 Ausgänge bei Verwendung einer CPU S7-1500 deklarieren.

Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung können nur Sprungmarken angegeben werden. Die Angabe von Anweisungen oder Operanden ist nicht zulässig.

Durch den Wert des Parameters K geben Sie die Nummer des Ausgangs und damit die Sprungmarke an, bei der die Programmbearbeitung fortgesetzt wird. Wenn der Wert am Parameter K größer als die Anzahl der verfügbaren Ausgänge ist, wird die Programmbearbeitung im nächsten Netzwerk des Bausteins fortgesetzt.

Die Anweisung "Sprungliste definieren" wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungliste definieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
K	Input	UINT	E, A, M, D, L oder Konstante	Gibt die Nummer des Ausgangs und damit den Sprung an, der ausgeführt wird.
DEST0	-	-	-	Erste Sprungmarke
DEST1	-	-	-	Zweite Sprungmarke
DESTn	-	-	-	Optionale Sprungmarken

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

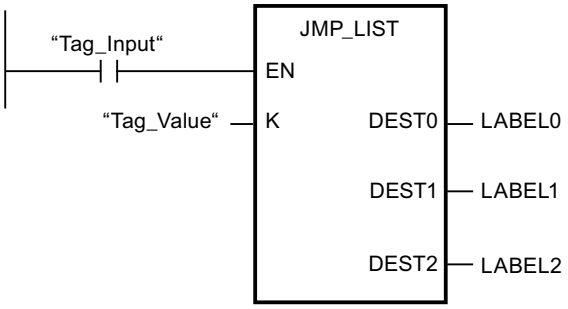

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand/Sprungmarke	Wert
K	Tag_Value	1
Dest0	LABEL0	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL0" gekennzeichnet ist.
Dest1	LABEL1	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.
Dest2	LABEL2	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL2" gekennzeichnet ist.

Wenn der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung "Sprungliste definieren" ausgeführt. Die Programmbearbeitung wird entsprechend dem Wert des Operanden "Tag_Value" in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

SWITCH: Sprungverteiler

Beschreibung

Mit der Anweisung "Sprungverteiler" definieren Sie mehrere Programmsprünge, die abhängig vom Ergebnis einer oder mehrer Vergleichsanweisungen ausgeführt werden.

Den zu vergleichenden Wert geben Sie am Parameter K an. Dieser Wert wird mit den Werten verglichen, die die einzelnen Eingänge liefern. Die Art des Vergleichs wählen Sie für jeden Eingang einzeln aus. Die Verfügbarkeit der verschiedenen Vergleichsanweisungen hängt dabei vom Datentyp der Anweisung ab.

Die folgende Tabelle zeigt die Vergleichsanweisungen, die abhängig vom gewählten Datentyp verfügbar sind:

Datentyp		Anweisung	Syntax
S7-1200	S7-1500		
Bitfolgen	Bitfolgen	Gleich	==
		Ungleich	<>

Datentyp		Anweisung	Syntax
S7-1200	S7-1500		
Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT	Gleich	==
		Ungleich	<>
		Größer gleich	>=
		Kleiner gleich	<=
		Größer	>
		Kleiner	<

Den Datentyp der Anweisung wählen Sie aus der Klappliste "???" der Anweisungsbox. Wenn Sie eine Vergleichsanweisung auswählen und der Datentyp der Anweisung noch nicht festgelegt ist, werden in der Klappliste "???" nur die Datentypen aufgelistet, die für die ausgewählte Vergleichsanweisung zulässig sind.

Die Bearbeitung der Anweisung beginnt mit dem ersten Vergleich und wird so lange ausgeführt, bis eine Vergleichsbedingung erfüllt ist. Ist eine Vergleichsbedingung erfüllt, werden die nachfolgenden Vergleichsbedingungen nicht berücksichtigt. Wenn keine der angegebenen Vergleichsbedingungen erfüllt ist, wird der Sprung am Ausgang ELSE ausgeführt. Wenn am Ausgang ELSE kein Programmsprung definiert ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Die Anzahl der Ausgänge ist in der Anweisungsbox erweiterbar. Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung geben Sie Sprungmarken (LABEL) an. Die Angabe von Anweisungen oder Operanden ist an den Ausgängen der Anweisung nicht zulässig.

Zu jedem zusätzlichen Ausgang wird automatisch auch ein Eingang eingefügt. Der an einem Ausgang programmierte Sprung wird ausgeführt, wenn die Vergleichsbedingung des entsprechenden Eingangs erfüllt ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungverteiler":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
K	Input	UINT	UINT	E, A, M, D, L oder Konstante	Gibt den zu vergleichenden Wert an.
<Vergleichswerte>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Eingangswerte, mit denen der Wert des Parameters K verglichen wird.
DEST0	-	-	-	-	Erste Sprungmarke
DEST1	-	-	-	-	Zweite Sprungmarke

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
DEST(n)	-	-	-	-	Optionale Sprungmarken: • S7-1200: n = 2 bis 32 • S7-1500: n = 2 bis 256
ELSE	-	-	-	-	Programmsprung, der ausgeführt wird, wenn keine der Vergleichsbedingungen erfüllt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

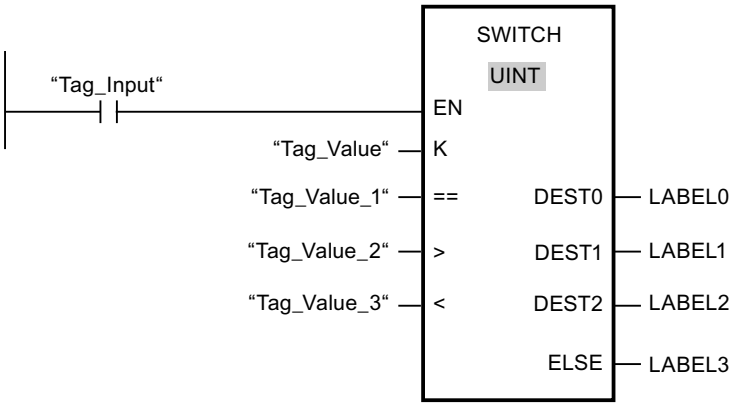

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand/Sprungmarke	Wert
K	Tag_Value	23
==	Tag_Value_1	20
>	Tag_Value_2	21
<	Tag_Value_3	19
DEST0	LABEL0	Sprung zur Sprungmarke "LABEL0", wenn der Wert des Parameters K gleich 20 ist.
DEST1	LABEL1	Sprung zur Sprungmarke "LABEL1", wenn der Wert des Parameters K größer 21 ist.

Parameter	Operand/Sprungmarke	Wert
DEST2	LABEL2	Sprung zur Sprungmarke "LABEL2", wenn der Wert des Parameters K kleiner als 19 ist.
ELSE	LABEL3	Sprung zur Sprungmarke "LABEL3", wenn keine der Vergleichsbedingungen erfüllt ist.

Wenn der Operand "Tag_Input" auf den Signalzustand "1" wechselt, wird die Anweisung "Sprungverteiler" ausgeführt. Die Programmbearbeitung wird in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

--(RET): Zurück springen

Beschreibung

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Programmbausteins bedingt oder absolut. Der Signalzustand des Rückgabewerts (Operand) wird beim Verlassen des Programmbausteins im Freigabeausgang ENO des aufrufenden Programmbausteins abgebildet.

Es gibt drei Arten, wie die Bearbeitung eines Programmbausteins beendet werden kann:

Beendigung des Programmbausteins	Erklärung
Ohne Aufruf der Anweisung	Nach der Ausführung des letzten Netzwerks wird der Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins wird auf den Signalzustand "1" gesetzt.
Aufruf der Anweisung mit Vorverknüpfung (siehe Beispiel)	Ist die Vorverknüpfung erfüllt, wird die Programmbearbeitung im aktuell aufgerufenen Programmbaustein beendet und der Programmbaustein verlassen. (Bedingtes Programmbausteinende) Die Programmbearbeitung wird im aufrufenden Programmbaustein (z. B. einem OB) nach dem Programmbausteinaufruf fortgesetzt. Der Freigabeausgang ENO des aufrufenden Programmbausteins entspricht dem Operanden.
Aufruf der Anweisung ohne Vorverknüpfung oder die Anweisung ist direkt an der linken Stromschiene angeschlossen	Der Programmbaustein wird unbedingt verlassen. (Absolutes Programmbausteinende) Der Freigabeausgang ENO des aufrufenden Porgrammbausteins entspricht dem Operanden.

Wenn ein Organisationsbaustein (OB) beendet wird, dann wird durch das Ablaufebenensystem ein anderer Programmbaustein ausgewählt und gestartet bzw. weiter ausgeführt:

- Wenn der Programmzyklus-OB beendet wird, wird dieser wieder neu gestartet.
- Wenn ein OB beendet wird, der einen anderen Programmbaustein unterbrochen hat (z. B. einen Alarm-OB), so wird der unterbrochene Programmbaustein (z. B. Programmzyklus-OB) weiter ausgeführt.

Hinweis

Anweisung RET im Zusammenhang mit JMP und JMPN

Wenn in einem Netzwerk bereits die Sprunganweisung "JMP: Springen bei VKE = 1" oder "JMPN: Springen bei VKE = 0" verwendet wird, dann darf die Anweisung "RET: Zurück springen" nicht verwendet werden. In einem Netzwerk darf nur eine springende Spule verwendet werden.

Einstellmöglichkeiten des Rückgabewerts (Operand)

Der Rückgabewert der Anweisung kann die folgenden Werte annehmen:

- Ret (RLO, entspricht dem Verknüpfungsergebnis VKE. Es wird der Signalzustand "1" an den Freigabeausgang ENO aufrufenden Programmbaustein geliefert, da die Anweisung RET als bedingte Anweisung nur dann ausgeführt wird, wenn die Bedingung TRUE ist.)
- Ret True bzw. Ret False (Der jeweilige Wert der Konstanten, TRUE bzw. FALSE, wird an den aufrufenden Programmbaustein geliefert.)
- Ret Value (Der Wert der Booleschen Variablen <Operand> wird an den aufrufenden Programmbaustein geliefert.)

Um den Rückgabewert der Anweisung einzustellen, klicken Sie auf das kleine gelbe Dreieck der Anweisung und wählen Sie in der Klappliste den gewünschten Wert aus.

Die folgende Tabelle zeigt den Status des aufrufenden Programmbausteins, wenn die Anweisung in einem Netzwerk innerhalb des aufgerufenen Programmbausteins programmiert ist:

VKE	Rückgabewert	ENO des aufrufenden Programmbausteins
1	RLO	1
	TRUE	1
	FALSE	0
	<Operand> Boolesche Variable mit den Speicherbereichsmöglichkeiten E, A, M, D, L, T und Z	<Operand>
0	RLO	Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Programmbausteins fortgesetzt.
	TRUE	
	FALSE	
	<Operand>	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

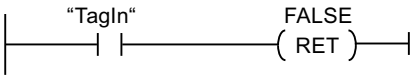

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Programmbearbeitung wird im aufgerufenen Programmbaustein beendet und im aufrufenden Programmbaustein fortgesetzt. Der Freigabeausgang ENO des aufrufenden Programmbausteins wird auf den Signalzustand "0" zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für den EN-/ENO-Mechanismus in KOP (Seite 154)

Grundlagen zu KOP (Seite 7425)

Laufzeitsteuerung

ENDIS_PW: Passwort-Legitimierung einschränken und freigeben

Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU projektierte Passwörter legitimiert sind oder nicht. Dadurch können Sie legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "0" hat, wird der aktuell eingestellte Zustand an den Ausgangsparametern angezeigt. Falls Änderungen an den Eingangsparametern vorgenommen wurden, werden diese nicht an die Ausgangsparameter übergeben.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F_PWD, FULL_PWD, R_PWD, HMI_PWD) übernommen:

- Wenn der Signalzustand "0" vorliegt, ist die Legitimation per Passwort nicht erlaubt.
- Wenn der Signalzustand "1" vorliegt, ist das Passwort verwendbar.

Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer das Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Nicht konfigurierte Passwörter müssen am Eingang den Signalzustand TRUE haben und liefern den Signalzustand TRUE am Ausgang. Das Failsafe Passwort ist nur für eine F-CPU

parametrierbar und muss daher in einer Standard CPU immer mit dem Signalzustand TRUE verschaltet werden. Wenn die Anweisung einen Fehler zurückliefert, dann bleibt der Aufruf ohne Auswirkung, d. h. die vorherige Sperre ist unverändert wirksam.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken einer leeren Speicherkarte (Übertragungskarte oder Programmkarte) in eine S7-1200-CPU.
- Der Übergang von NETZ AUS - NETZ EIN deaktiviert in der S7-1200-CPU den Schutz. Die Anweisung "Passwort-Legitimierung einschränken und freigeben" muss im Programm erneut (z. B. im Anlauf-OB) aufgerufen werden.

Hinweis

Wenn das HMI Passwort nicht freigegeben ist, dann sperrt die Anweisung "Passwort-Legitimierung einschränken und freigeben" den Zugriff von HMI-Systemen.

Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

Ungewolltes Aussperren bei einer S7-1500-CPU verhindern

Die Einstellungen können im Frontpanel der S7-1500-CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Ungewolltes Aussperren bei einer S7-1200-CPU verhindern

Da es bei einer S7-1200-CPU keinen Betriebsartenschalter gibt, wird der Schutz bei NETZ AUS - NETZ EIN deaktiviert. Damit ist es möglich und empfehlenswert, ein ungewolltes Aussperren mithilfe bestimmter Programmsequenzen innerhalb Ihres Programms zu verhindern.

Programmieren Sie dazu eine Zeitsteuerung entweder über einen Weckalarm-OB oder einen Timer im Main OB (OB 1). Dadurch haben Sie die Möglichkeit, nach einem Übergang von

NETZ AUS - NETZ EIN und der damit verbundenen Deaktivierung des Schutzes, die Anweisung "Passwort-Legitimierung einschränken und freigeben" in dem entsprechenden OB (z. B. OB 1 oder OB 35) zeitnah wieder aufzurufen. Um das Zeitfenster, in dem die Anweisung nicht aktiv ist und damit keine Einschränkungen bei der Passwort-Legitimierung bestehen, möglichst klein zu halten, rufen Sie die Anweisung im Anlauf-OB (OB 100) auf. Diese Vorgehensweise bietet Ihnen den größtmöglichen Schutz vor unbefugtem Zugriff.

Wenn es zu einer ungewollten Aussperrung gekommen ist, dann können Sie den Aufruf im Anlauf-OB überspringen (z. B. durch die Abfrage eines Eingangsparameters) und haben die eingestellte Zeit (z. B. 10 Sekunden bis 1 Minute), um eine Verbindung zur CPU aufzubauen, bevor die Sperre wieder aktiv ist.

Wenn Sie in Ihrem Programmcode keinen Timer vorgesehen haben und es zu einer Aussperrung kam, dann stecken Sie eine leere Übertragungskarte oder eine leere Programmkarte in die CPU. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Danach müssen Sie das Anwenderprogramm aus STEP 7 erneut in die CPU laden.

Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte oder Programmkarte. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

⚠ WARNUNG

Leere Übertragungskarte stecken

Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Nach dem Ziehen der Übertragungskarte ist der Inhalt der Übertragungskarte im internen Ladespeicher. Achten Sie an dieser Stelle darauf, dass die Karte kein Programm enthält.

⚠ WARNUNG

Leere Programmkarte stecken

Wenn Sie eine Programmkarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Achten Sie darauf, dass die Programmkarte leer ist. Der interne Ladespeicher wird auf die leere Programmkarte kopiert. Nach dem Ziehen der ehemals leeren Programmkarte ist der interne Ladespeicher leer.

Sie müssen die Übertragungskarte oder Programmkarte ziehen, bevor Sie die CPU in RUN versetzen.

Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

Aktion	Passwortschutz durch die Anweisung
Grundzustand nach • Betriebsartenschalter auf STOP • Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control) • Zurücksetzen auf Werkseinstellung	Nicht aktiviert (Keine Einschränkungen)
Grundzustand nach NETZ EIN	• S7-1200-CPU: Die Sperre ist deaktiviert und die Anweisung muss im Programm erneut aufgerufen werden (z. B. im Anlauf-OB). • S7-1500-CPU: Aktiviert (wenn vor NETZ AUS eine Sperre aktiviert war). Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent.
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT	Aktiviert Die Passwörter dürfen weiterhin nicht verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.
F_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff inklusive Failsafe • F_PWD = "0": Passwort nicht erlauben • F_PWD = "1": Passwort erlauben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FULL_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff • FULL_PWD = "0": Passwort nicht erlauben • FULL_PWD = "1": Passwort erlauben
R_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lesezugriff • R_PWD = "0": Passwort nicht erlauben • R_PWD = "1": Passwort erlauben
HMI_PWD	Input	BOOL	E, A, M, D, L oder Konstante	HMI-Zugriff • HMI_PWD = "0": Passwort nicht erlauben • HMI_PWD = "1": Passwort erlauben
F_PWD_ON	Output	BOOL	E, A, M, D, L oder Konstante	Status Lese-/Schreibzugriff inklusive Failsafe • F_PWD_ON = "0": Passwort nicht erlaubt • F_PWD_ON = "1": Passwort erlaubt
FULL_PWD_ON	Output	BOOL	E, A, M, D, L oder Konstante	Status Lese-/Schreibzugriff • FULL_PWD_ON = "0": Passwort nicht erlaubt • FULL_PWD_ON = "1": Passwort erlaubt
R_PWD_ON	Output	BOOL	E, A, M, D, L oder Konstante	Status Lesezugriff • R_PWD_ON = "0": Passwort nicht erlaubt • R_PWD_ON = "1": Passwort erlaubt
HMI_PWD_ON	Output	BOOL	E, A, M, D, L oder Konstante	Status HMI-Zugriff • HMI_PWD_ON = "0": Passwort nicht erlaubt • HMI_PWD_ON = "1": Passwort erlaubt
RET_VAL	Output	WORD	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8090	Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt
80D0	Das Passwort für Failsafe ist nicht konfiguriert. Bei Standard-CPUs muss der Signalzustand TRUE sein.
80D1	Der Lese-/Schreibzugriff ist nicht konfiguriert
80D2	Der Lesezugriff ist nicht konfiguriert
80D3	Der HMI-Zugriff ist nicht konfiguriert
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

PC-Systeme

SHUT_DOWN: Zielsystem herunterfahren

Beschreibung

Mit der Anweisung “SHUT_DOWN: Zielsystem herunterfahren” fahren Sie das PC-basierte Automatisierungssystem herunter, starten den S7-Software Controller CPU 150xS oder Windows auf dem PC-basierte Automatisierungssystem neu.

Die Anweisung finden Sie in der Task Card "Anweisungen" unter Einfache Anweisungen > Programmsteuerung > Laufzeitsteuerung.

Ein Neustart ist z. B. in den folgenden Situationen sinnvoll:

- Eine Industrie-USV (unterbrechungsfreie Stromversorgung) meldet über einen digitalen Eingang einen Spannungsausfall.
- Windows reagiert nicht mehr oder liefert einen „Blue Screen“.
- Im Anwenderprogramm werden zu viele Fehler-OBs aufgerufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zielsystem herunter fahren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	MODE = 1: CPU 150xS und Windows herunter fahren Die CPU geht in STOP und speichert die remanenten Daten. Anschließend werden CPU und Windows herunter gefahren. Das System muss manuell gestartet werden. MODE = 2: CPU 150xS neu starten Die CPU geht in STOP und speichert die remanenten Daten. Anschließend wird die CPU heruntergefahren und neu gestartet. MODE = 3: Windows neu starten. Die CPU bleibt in RUN. Windows wird neu gestartet (ab TIA Portal V14 ist MODE 3 nur für Abwärtskompatibilität gelassen. Es wird empfohlen, MODE 4 oder MODE 5 zu verwenden). MODE = 4: Windows wird korrekt heruntergefahren und neu gestartet. Die CPU bleibt im RUN. MODE = 5: Windows neu starten. (Vergleichbar mit MODE 3; Ausnahme: MODE 5 ist nur dann zu verwenden, wenn Windows abgestürzt ist).
COM-MENT	Input	STRING	E, A, M, D, L	Bei Mode = 1, 3 und 4 können Sie eine Begründung für den Neustart mitgeben. Die Begründung wird im Windows Event Log ausgegeben.
Ret_Val	Return	WORD	A, M, D, L	Ret_Val = 0: kein Fehler Ret_Val = 8090: Der übergebene Wert an MODE wird nicht unterstützt. Ret_Val = 8091: Windows reagiert nicht auf den Aufruf der Anweisung Shut_Down (gültig nur für Mode 3 und 4). Ret_Val = 8092: Bei Auftreten dieses Fehlers, wenden Sie sich bitte an den SIMATIC Customer Support (gültig nur für Mode 3 und 4).

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Aufruf der Anweisung

Es gelten die folgenden Aufrufbedingungen:

- Für eine CPU der Baureihe S7-1200 gilt:
  In Firmware Versionen < 2.2 können Sie die Anweisung nur in einem Zyklus-Organisationsbaustein der Priorität 1 aufrufen. Das entspricht der niedrigsten Priorität aller Organisationsbausteine. Wenn Sie die Anweisung in einem Organisationsbaustein mit höherer Priorität aufrufen, dann wird sie nicht ausgeführt und das Ergebnis (Bie-Bit, Freigabeausgang ENO) ist immer "0".
  In Firmware Versionen >= 2.2 können Sie die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.
- Für eine CPU der Baureihe S7-1500 gilt:
  Sie können die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

GET_ERROR: Fehler lokal abfragen

Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Programmbausteins ab. Dabei handelt es sich in der Regel um Programmier- oder Zugriffsfehler. Wenn das System während der Programmbausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler, innerhalb der Ausführung des Programmbausteins seit der letzten Ausführung der Anweisung, eine ausführliche Information im Operanden am Ausgang ERROR gespeichert.

Am Ausgang ERROR können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Treten im Programmbaustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehlerinformation zum nächsten aufgetretenen Fehler von der Anweisung ausgegeben.

Hinweis

Ausgang ERROR

Der Ausgang ERROR wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.
- Fragen Sie den Freigabeausgang ENO ab.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung nicht beeinflusst.

Hinweis

Aktivierung der lokalen Fehlerbehandlung

Sobald Sie die Anweisung in den Programmcode eines Programmbausteins einfügen, wird die lokale Fehlerbehandlung aktiviert und voreingestellte Systemreaktionen werden beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ERROR	Output	ErrorStruct	D, L	Fehlerinformationen

Datentyp "ErrorStruct"

Der Datentyp "ErrorStruct" kann in einen globalen Datenbaustein oder in die Bausteinschnittstelle eingefügt werden. Sie können den Datentyp auch mehrfach einfügen, wenn Sie jedes Mal einen anderen Namen für die Datenstruktur vergeben. Die Datenstruktur und der Name einzelner Strukturelemente sind nicht änderbar. Wenn Sie die Fehlerinformation in einem globalen Datenbaustein speichern, kann sie auch von anderen Programmbausteinen ausgelesen werden.

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

Strukturkomponente	Datentyp	Beschreibung
ERROR_ID	WORD	Fehler-ID
FLAGS	BYTE	Zeigt, ob der Fehler während eines Programmbausteinaufrufs aufgetreten ist. 16#01: Fehler während eines Programmbausteinaufrufs 16#00: Kein Fehler während eines Programmbausteinaufrufs

Strukturkomponente		Datentyp	Beschreibung
REACTION		BYTE	Voreingestellte Reaktion: 0: Ignorieren (Schreibfehler) 1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler) 2: Anweisung überspringen (Systemfehler)
CODE_ADDRESS		CREF	Informationen zur Adresse und Art des Programmbausteins
	BLOCK_TYPE	BYTE	Programmbausteintyp, in dem der Fehler aufgetreten ist: 1: Organisationsbaustein (OB) 2: Funktion (FC) 3: Funktionsbaustein (FB)
	CB_NUMBER	UINT	Nummer des Codebausteins
	OFFSET	UDINT	Verweis auf den internen Speicher
MODE		BYTE	Informationen zur Adresse eines Operanden
OPERAND_NUMBER		UINT	Operandennummer des Maschinenbefehls
POINTER_NUMBER_LOCATION		UINT	(A) Interner Zeiger
SLOT_NUMBER_SCOPE		UINT	(B) Ablagebereich im internen Speicher
DATA_ADDRESS		NREF	Informationen zur Adresse eines Operanden
	AREA	BYTE	(C) Speicherbereich: L: 16#40...16#7F, 16#86, 16#87, 16#8E, 16#8F, 16#C0...16#FF E: 16#81 A: 16#82 M: 16#83 DB: 16#40, 16#84, 16#85, 16#8A, 16#8B PE: 16#01 PA: 16#02 Technologische Objekte: 16#04
	DB_NUMBER	UINT	(D) Nummer des Datenbausteins
	OFFSET	UDINT	(E) Relative Adresse des Operanden

Strukturkomponente "ERROR_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR_ID" ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
2576	9590	Fehler in der Lokaldatenverteilung
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

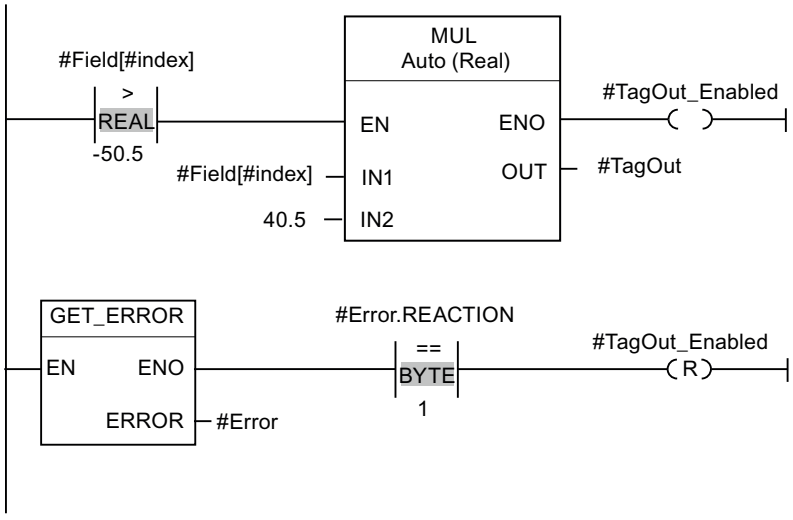

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Freigabeausgang ENO der Anweisung "Multiplizieren" und der Operand #TagOut_Enabled liefern trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler lokal abfragen" nach der Anweisung "Multiplizieren" zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler lokal abfragen" liefert, wird mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Wenn die Strukturkomponente #Error.REACTION den Wert "1" hat, dann handelt es sich um einen Lese-/Zugriffsfehler und der Ausgang #TagOut_Enabled wird wieder zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu KOP (Seite 7425)

GET_ERR_ID: Fehler-ID lokal abfragen

Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Dabei handelt es sich in der Regel um Zugriffsfehler. Wenn das System während der Bausteinbearbeitung Fehler innerhalb der Ausführung des Bausteins seit der letzten Ausführung der Anweisung meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang ID gespeichert.

Am Ausgang ID können nur Operanden vom Datentyp WORD angegeben werden. Treten im Baustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehler-ID des nächsten aufgetretenen Fehlers von der Anweisung ausgegeben.

Hinweis

Der Ausgang ID wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.
- Fragen Sie den Freigabeausgang ENO ab.

Der Freigabeausgang ENO der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Ein Beispiel, wie Sie die Anweisung auch in Kombination mit weiteren Fehlerbehandlungsmöglichkeiten realisieren können, finden Sie unter "Siehe auch".

Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ID	Output	WORD	E, A, M, D, L	Fehler-ID

Parameter ID

Die folgende Tabelle zeigt die Werte, die am Parameter ID ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

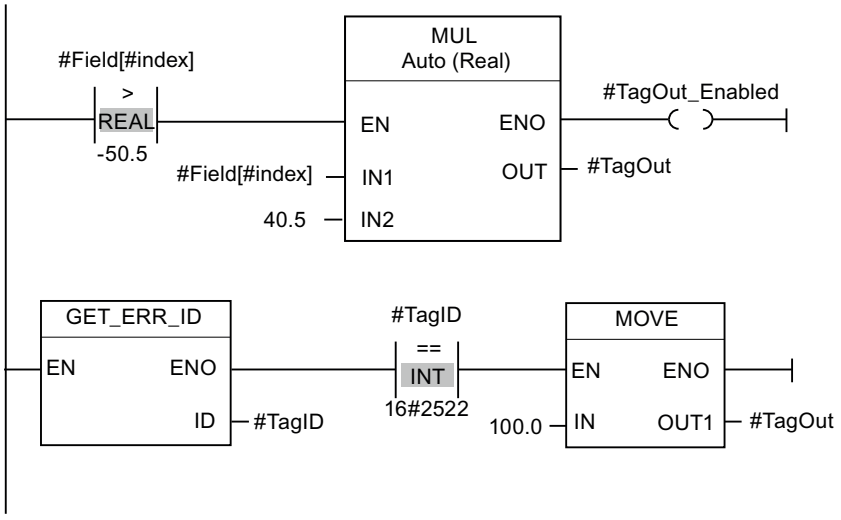

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Freigabeausgang ENO der Anweisung "Multiplizieren" und der Operand #TagOut_Enabled liefern trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler-ID lokal abfragen" nach der Anweisung "Multiplizieren" zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler-ID lokal abfragen" liefert, wird mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Wenn der Operand #TagID die ID 2522 liefert, dann handelt es sich um einen Lese-/Zugriffsfehler und der Wert "100.0" wird in den Ausgang #TagOut geschrieben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu KOP (Seite 7425)

INIT_RD: Alle remanenten Daten zurücksetzen

Beschreibung

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B5	Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

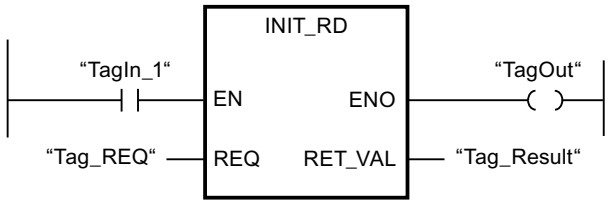

Wenn die Operanden "TagIn_1" und "Tag_REQ" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-

Zeiten und -Zähler werden zurückgesetzt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1".

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu +32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Beispiel zum Einfluss der geplanten Verzögerungszeit

Im folgenden Beispiel sehen Sie den Einfluss der Verzögerungszeit der Anweisung "WAIT" in verschiedenen Szenarien.

Das folgende Bild stellt die Szenarien schematisch dar:

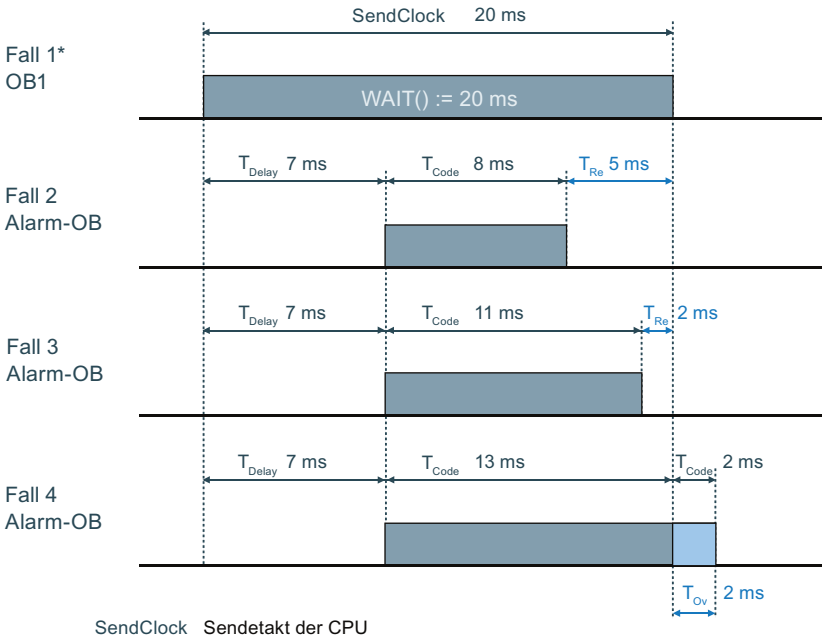

SendClock Sendetakt der CPU

T_{Delay} Verzögerungszeit des Alarm-OBs

T_{Code} Zeit der Programmausführung

T_{Re} Verbleibende Zeit

T_{Ov} Zeit der Überlänge, um welche sich der Sendetakt erhöht

* Der OB1 mit der Anweisung ”WAIT” ist ebenso Teil von Fall 2, 3 und 4 und muss mit berücksichtigt werden.

Verbleibende Zeit = Die Spanne vom Ende der geplanten Verzögerungszeit (via "WAIT") bis zur Beendigung des Alarm-OBs

Zeit der Überlänge = Die Spanne von der Beendigung des Alarm-OBs bis zum Ende der geplanten Verzögerungszeit (via "WAIT").

Fall 1:

Die Anweisung "WAIT" wird in einem OB1 aufgerufen. Die Anweisung "WAIT" kann von höher prioren OBs oder höher prioren Prozessen (z.B. System Threads) unterbrochen werden. Die Verzögerungszeit der Anweisung "WAIT" ist aber weder verändert noch aufgeschoben.

Fall 2 und Fall 3:

Die Verarbeitung des Programms im OB1 wird nach einer Zeitverzögerung von 20 ms fortgesetzt. Diese Zeitverzögerung wird durch den Aufruf der Anweisung "WAIT" im OB1 berechnet (siehe OB1 mit WAIT). Innerhalb dieser 20 ms kann ein Alarm-OB seinen eigenen Programmcode ausführen. Der Sendetakt der CPU wird nicht verändert.

Fall 4:

Die Verarbeitung des Programms im OB1 wird nach Beendingung des höher prioren Prozesses fortgesetzt. Die 20 ms Verzögerungszeit im OB1 sind verstrichen, aber der höher priore Prozess muss noch beendet werden. Der Sendetakt der CPU wird erhöht.

Hinweis

Ausführungsreihenfolge von System- oder Kommunikationsprozessen (System Threads)

Die System Threads verwenden meistens die Priorität "15". Ebenso gibt es System Threads mit einer höheren Priorität als "26", doch diese Prozesse haben eine geringere CPU-Auslastung. System Threads werden im Bild nicht dargestellt.

Laufzeitmessung des OB1 mit Hilfe der Anweisung "RT_INFO":

Fall 2: 20 ms - 8 ms - System Threads = <12 ms. Sendetakt: ~20 ms.

Fall 3: 20 ms - 11 ms - System Threads - <9 ms. Sendetakt: ~20 ms.

Fall 4: 20 ms - 15 ms - System Threads - <7 ms. Sendetakt: ~22 ms.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

RUNTIME: Laufzeitmessung

Beschreibung

Mit der Anweisung "Laufzeitmessung" messen Sie die Laufzeit des gesamten Programms, einzelner Bausteine oder Befehlssequenzen.

Wenn Sie die Laufzeit Ihres gesamten Programms messen wollen, dann rufen Sie die Anweisung "Laufzeitmessung" im OB1 auf. Der Startpunkt der Laufzeitmessung wird mit dem ersten Aufruf gesetzt und der Ausgang RET_VAL liefert nach dem zweiten Aufruf die Laufzeit des Programms. Die gemessene Laufzeit enthält alle möglichen Vorgänge der CPU, die innerhalb des Programmablaufs statt gefunden haben, wie z. B. Unterbrechungen durch höher priore Ereignisse oder Kommunikation. Die Anweisung "Laufzeitmessung" liest einen internen Zähler der CPU aus und schreibt dessen Wert in den Durchgangsparameter MEM. Die Anweisung berechnet die aktuelle Laufzeit des Programms gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Wenn Sie die Laufzeit einzelner Bausteine oder einzelner Befehlssequenzen messen wollen, dann benötigen Sie drei separate Netzwerke. Rufen Sie die Anweisung "Laufzeitmessung" in einem einzelnen Netzwerk innerhalb Ihres Programms auf. Mit diesem ersten Aufruf der Anweisung setzen Sie den Startpunkt der Laufzeitmessung. Anschließend rufen Sie im nächsten Netzwerk den gewünschten Programmbaustein oder die Befehlssequenz auf. Rufen Sie dann in einem weiteren Netzwerk die Anweisung "Laufzeitmessung" zum zweiten Mal auf und weisen Sie dem Durchgangsparameter MEM den gleichen Speicher wie beim ersten Aufruf der Anweisung zu. Die Anweisung "Laufzeitmessung" im dritten Netzwerk liest einen internen Zähler der CPU aus und berechnet die aktuelle Laufzeit des Programmbausteins oder der Befehlssequenz gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Die Anweisung "Laufzeitmessung" verwendet einen internen hochfrequenten Zähler um die Zeit zu berechnen. Wenn der Zähler überläuft, gibt die Anweisung Werte <= 0.0 zurück. Dies kann für S7-1200-CPUs mit Firmware Stand <V4.2 bis zu einmal pro Minute vorkommen. Diese Runtimewerte sind zu ignorieren.

Hinweis

Da beim optimierten Übersetzen des Programms die Reihenfolge der Anweisungen innerhalb einer Befehlssequenz geändert wird, kann die Laufzeit einer Befehlssequenz nicht genau bestimmt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Laufzeitmessung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MEM	InOut	LREAL	E, A, M, D, L	Der Inhalt ist nur für interne Zwecke vorgesehen.
RET_VAL	Output	LREAL	E, A, M, D, L	Liefert die gemessene Laufzeit in Sekunden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung anhand einer Berechnung der Laufzeit eines Programmbausteins:

Netzwerk 1:

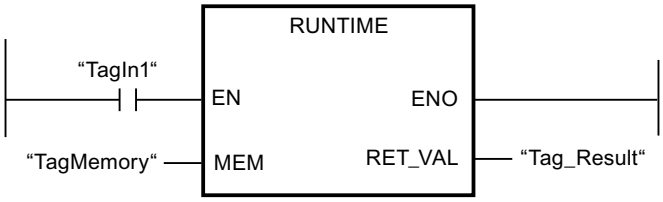

Netzwerk 2:

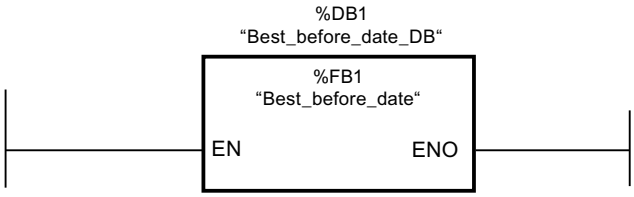

Netzwerk 3:

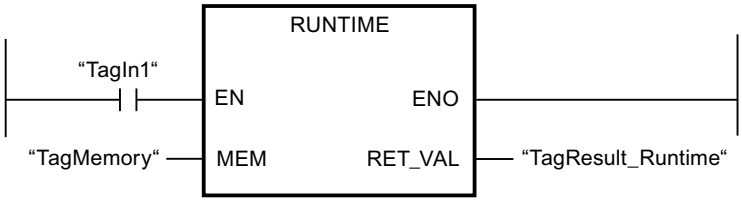

Wenn der Operand "TagIn1" im Netzwerk 1 den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Mit dem ersten Aufruf der Anweisung wird der Startpunkt für die Laufzeitmessung gesetzt und als Referenz für den zweiten Aufruf der Anweisung im Operanden "TagMemory" zwischengespeichert.

Im Netzwerk 2 wird der Programmbaustein FB1 "Best_before_date" aufgerufen.

Wenn der Programmbaustein FB1 abgearbeitet ist und der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung im Netzwerk 3 ausgeführt. Der zweite Aufruf der Anweisung berechnet die Laufzeit des Programmbausteins und schreibt das Ergebnis in den Ausgang RET_VAL.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Ein ausführlicheres Beispiel, wie Sie die Gesamtzykluszeit eines Programms messen können, finden Sie im Siemens Industry Online Support unter folgender Beitrags-ID: 87668055 (https://support.industry.siemens.com/cs/ww/de/view/87668055)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

Wortverknüpfungen

AND: UND verknüpfen

Beschreibung

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch UND verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Weitere Eingänge, deren Werte verknüpft werden.
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

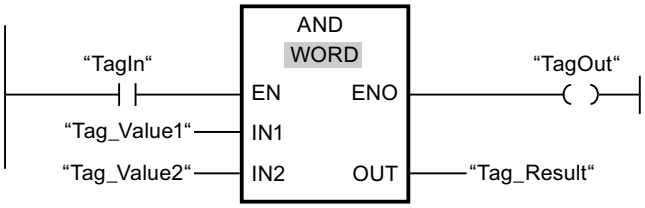

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

OR: ODER verknüpfen

Beschreibung

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüp-fung
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Weitere Eingänge, deren Werte verknüpft werden.
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

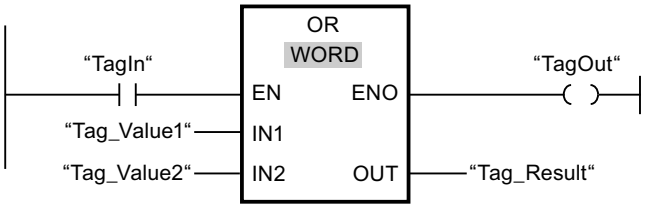

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

XOR: EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch EXKLUSIV ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Weitere Eingänge, deren Werte verknüpft werden.
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

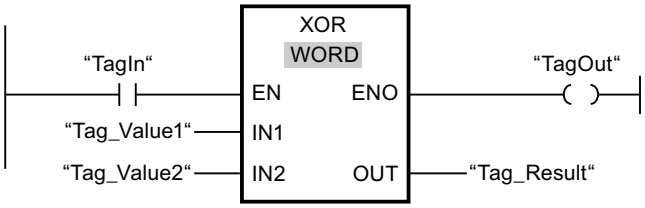

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

INVERT: Einerkomplement erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L, P	Einerkomplement des Wertes am Eingang IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

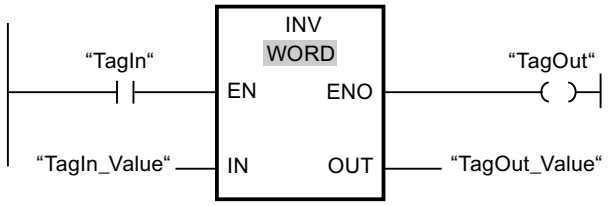

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	W#16#000F	W#16#7E
OUT	TagOut_Value	W#16#FFF0	W#16#81

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn_Value" um und schreibt das Ergebnis in den Ausgang "TagOut_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	UINT	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

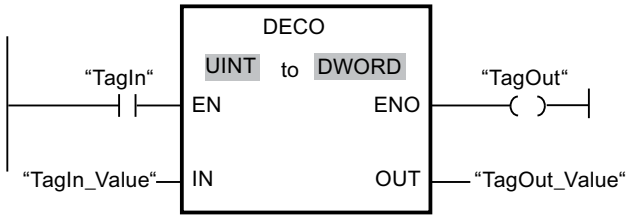

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

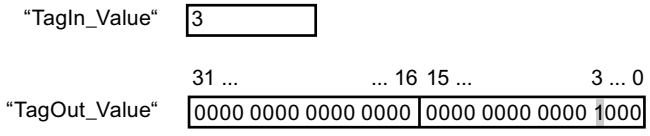

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn_Value" am Eingang ab und setzt das dritte Bit im Wert des Operanden "TagOut_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

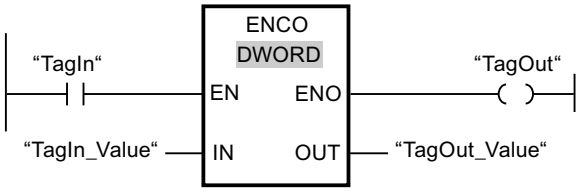

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

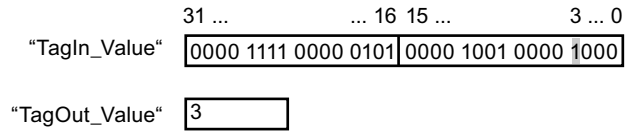

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung selektiert das niederwertigste gesetzte Bit am Eingang "TagIn_Value" und schreibt die Bitposition "3" in die Variable am Ausgang "TagOut_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Alle Variablen an allen Parametern müssen den gleichen Datentyp haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
G	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Schalter

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, CHAR, WCHAR, DATE	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, WCHAR, DATE	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, CHAR, WCHAR, DATE	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, WCHAR, DATE	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, CHAR, WCHAR, DATE	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

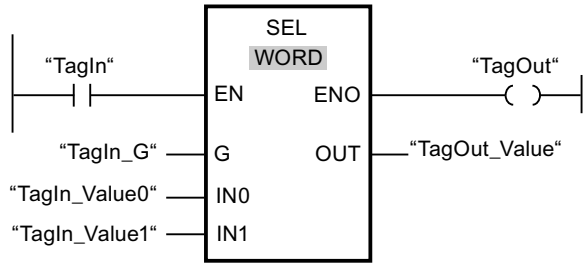

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn_G" wird der Wert am Eingang "TagIn_Value0" oder "TagIn_Value1" selektiert und in den Ausgang "TagOut_Value" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MUX: Multiplexen

Beschreibung

Mit der Anweisung "Multiplexen" kopieren Sie den Inhalt eines ausgewählten Eingangs in den Ausgang OUT. Die Anzahl der auswählbaren Eingänge der Anweisungsbox ist erweiterbar. Sie können maximal 32 Eingänge deklarieren.

Die Eingänge werden in der Box automatisch nummeriert. Die Nummerierung fängt mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Eingang, dessen Inhalt in den Ausgang OUT kopiert wird. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Eingänge ist, wird der Inhalt des Parameters ELSE in den Ausgang OUT kopiert und dem Freigabeausgang ENO den Signalzustand "0" zugewiesen.

Die Anweisung "Multiplexen" kann nur ausgeführt werden, wenn die Variablen an allen Eingängen sowie am Ausgang OUT vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, für den nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Eingang am Parameter K liegt außerhalb der verfügbaren Eingänge. Diese Reaktion ist unabhängig davon, ob der Eingang ELSE verwendet wird oder nicht. Der Wert am Ausgang OUT bleibt unverändert.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplexen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P oder Konstante	Gibt den Eingang an, dessen Inhalt kopiert wird. • Wenn K = 0 => Parameter IN0 • Wenn K = 1 => Parameter IN1, etc.
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
INn	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte
ELSE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Gibt den Wert an, der bei K > n kopiert wird
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ausgang, in den der Wert kopiert wird

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

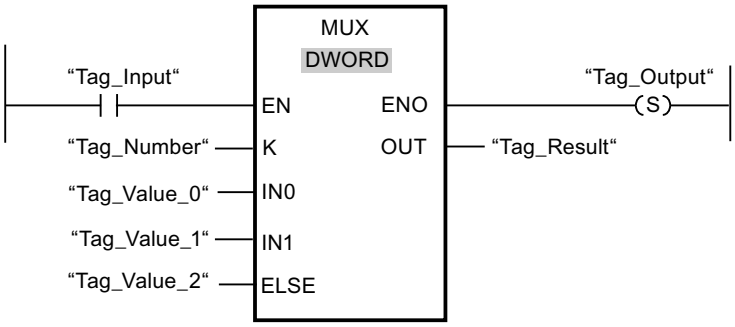

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
K	Tag_Number	1
IN0	Tag_Value_0	DW#16#00000000
IN1	Tag_Value_1	DW#16#003E4A7D
ELSE	Tag_Value_2	DW#16#FFFF0000
OUT	Tag_Result	DW#16#003E4A7D

Wenn der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Entsprechend dem Wert des Operanden "Tag_Number" wird der Wert am Eingang "Tag_Value_1" kopiert und dem Operanden am Ausgang "Tag_Result" zugewiesen. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden der Freigabeausgang ENO und "Tag_Output" gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

DEMUX: Demultiplexen

Beschreibung

Mit der Anweisung "Demultiplexen" kopieren Sie den Inhalt des Eingangs IN in einen ausgewählten Ausgang. Die Anzahl der auswählbaren Ausgänge ist in der Anweisungsbox erweiterbar. Die Ausgänge werden in der Box automatisch nummeriert. Die Nummerierung fängt bei OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Ausgang, in welchen der Inhalt des Eingangs IN kopiert wird. Die anderen Ausgänge werden nicht verändert. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Ausgänge ist, wird der Inhalt des Eingangs IN in den Parameter ELSE kopiert und dem Freigabeausgang ENO der Signalzustand "0" zugewiesen.

Die Anweisung "Demultiplexen" kann nur ausgeführt werden, wenn die Variablen am Eingang IN sowie an allen Ausgängen vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, an welchem nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Demultiplexen":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P oder Konstante	Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. • Wenn K = 0 => Parameter OUT0 • Wenn K = 1 => Parameter OUT1, etc.
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Eingangswert
OUT0	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Erster Ausgang
OUT1	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Zweiter Ausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
OUTn	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Optionale Ausgänge
ELSE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ausgang, in welchen der Eingangswert (IN) bei K > n kopiert wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

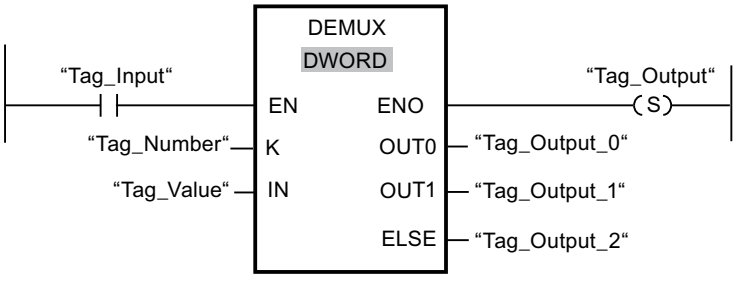

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Tabelle 4-1 Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

Parameter	Operand	Werte	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Tabelle 4-2 Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

Parameter	Operand	Werte	
OUT0	Tag_Output_0	unverändert	unverändert
OUT1	Tag_Output_1	DW#16#FFFFFFFF	unverändert
ELSE	Tag_Output_2	unverändert	DW#16#003E4A7D

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Abhängig vom Wert des Operanden "Tag_Number" wird der Wert am Eingang IN in den entsprechenden Ausgang kopiert.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:

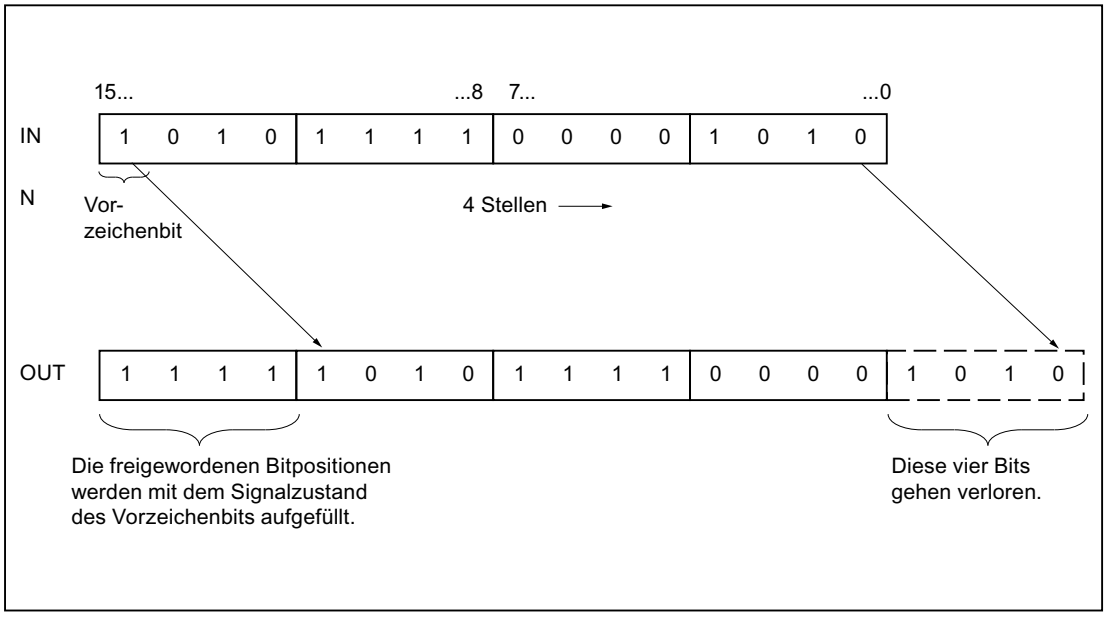

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der verschoben wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

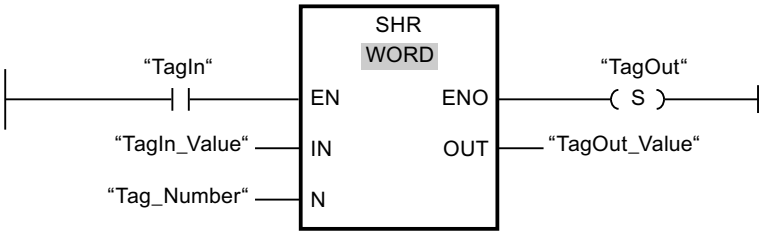

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts schieben" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

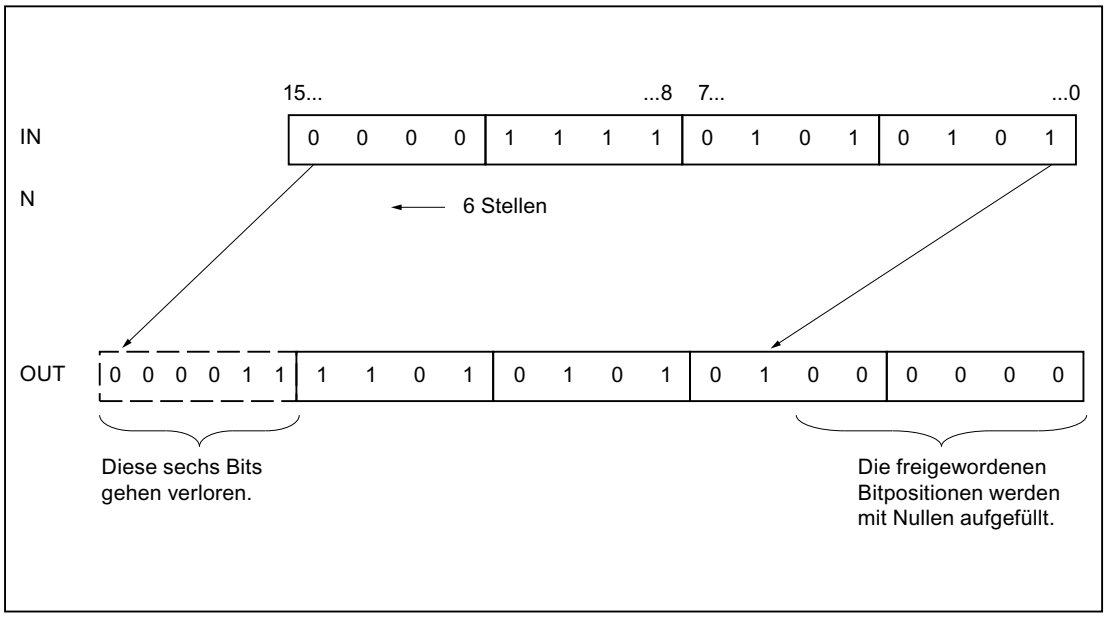

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der verschoben wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

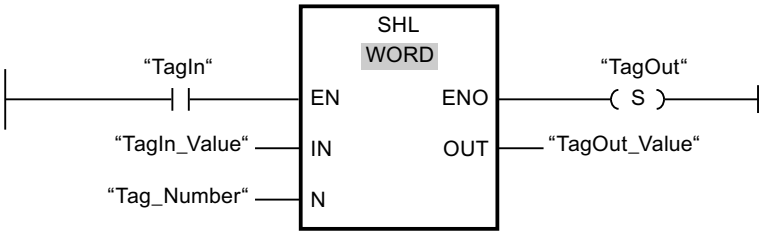

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links schieben" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

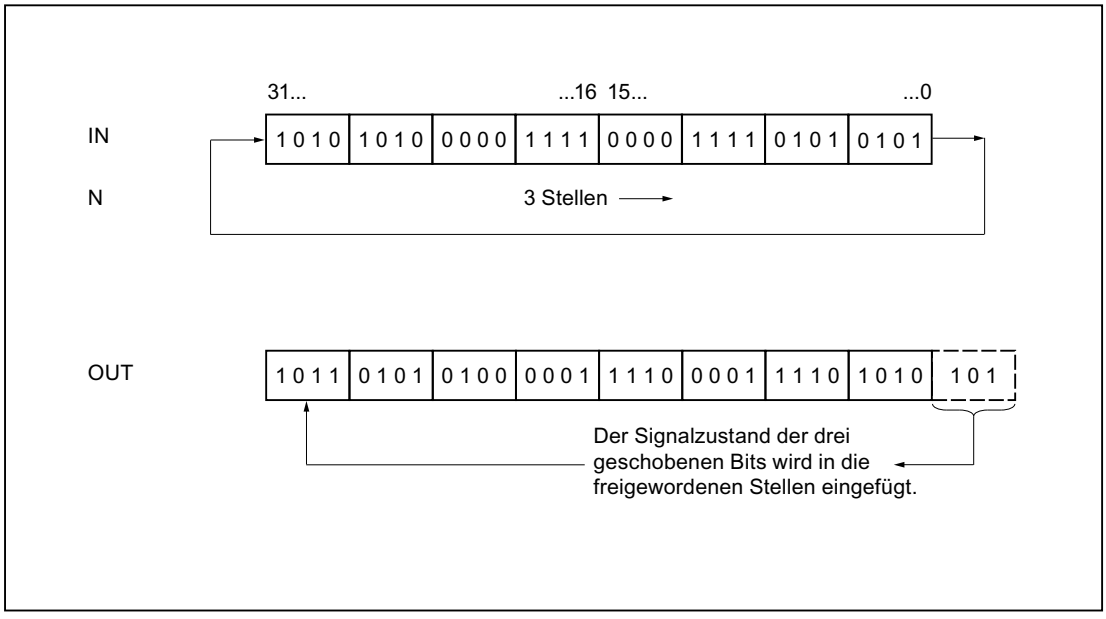

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der rotiert wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

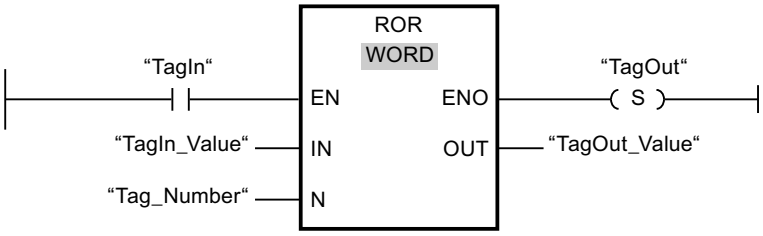

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0000 1111 1001 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1000 0111 1100

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts rotieren" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

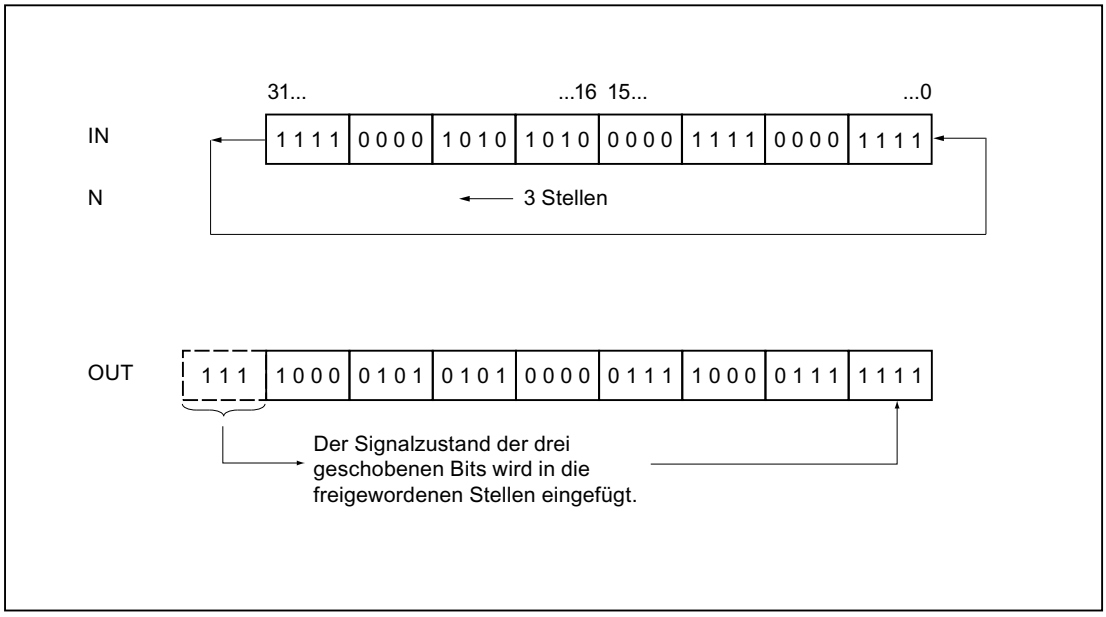

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der rotiert wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

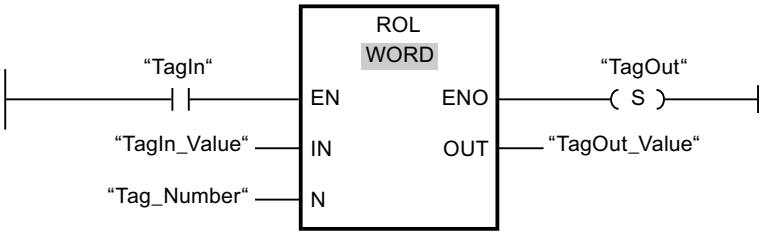

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1010 1000 1111 0110
N	Tag_Number	5
OUT	TagOut_Value	0001 1110 1101 0101

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links rotieren" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

Legacy

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L oder Konstante	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Maximale Schrittnummer (z. B.: LST_STEP = 16#08; Es sind maximal 8 Schritte möglich.)
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L oder Konstante	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P oder Konstante	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Vorherige Systemzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

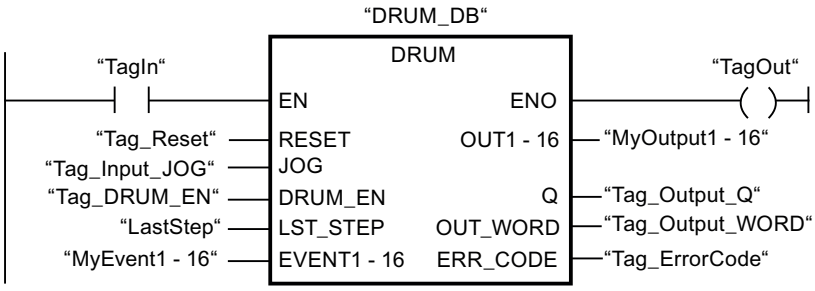

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu KOP (Seite 7425)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET (nur wenn ET < PT ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn ET < PT ist), OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CMD	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
CMD_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit zu CMD

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

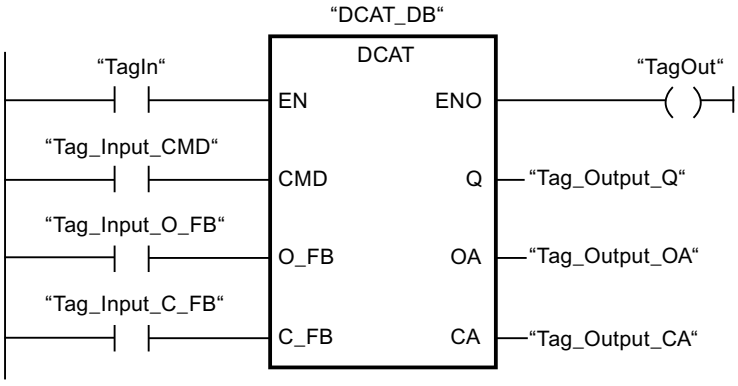

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
O_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
O_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Schließen"

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop

Eingangsparameter								Ausgangsparameter								
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT

Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

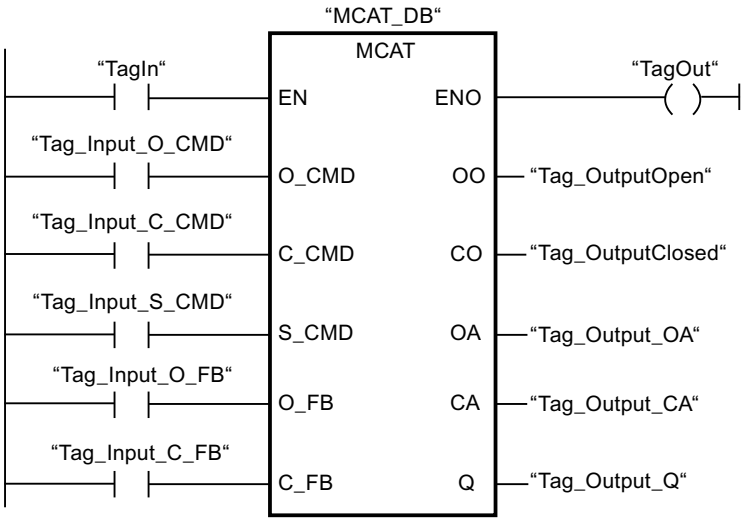

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P oder Konstante	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu KOP (Seite 7425)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird. Am Parameter IN können auch Konstanten angegeben werden.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit Am Parameter SAMPLE_T können auch Konstanten angegeben werden.
OUT	Output	REAL	E, A, M, D, L	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit.
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

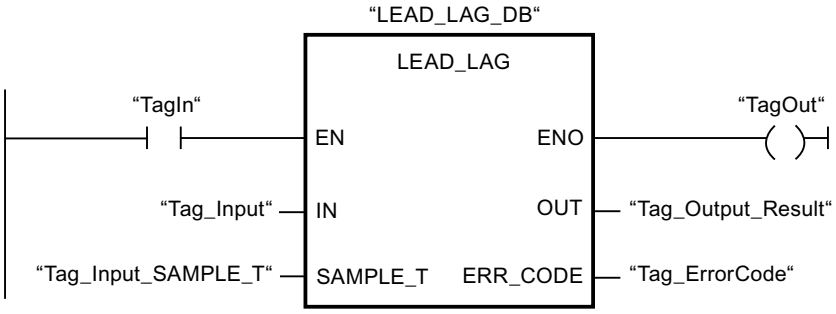

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu KOP (Seite 7425)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	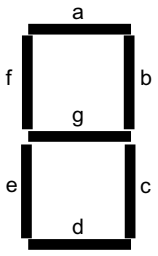
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

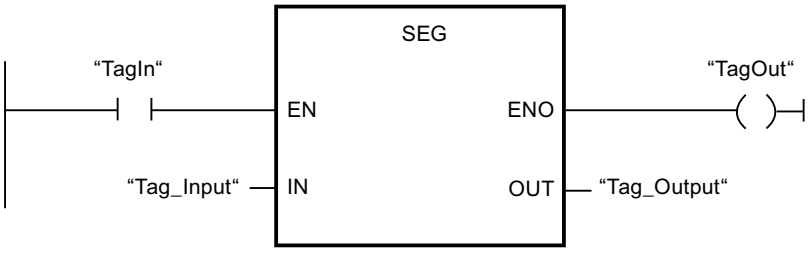

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Anzeige: 1234

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bifolgen	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

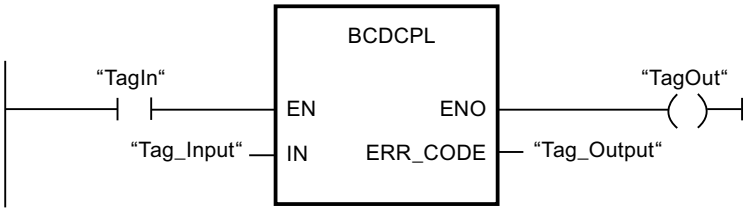

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu KOP (Seite 7425)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzten Bits gezählt werden.
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bits

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

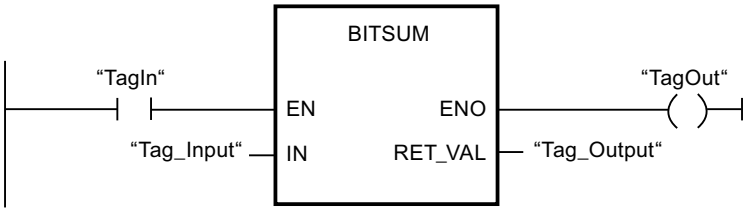

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

4.1.3.2 FUP

Bitverknüpfungen

&: UND-Verknüpfung

Beschreibung

Mit der Anweisung "UND-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der UND-Wahrheitstabelle aus.

Beträgt der Signalzustand aller Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand eines Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

Hinweis

Als "inaktiv" angezeigte Anweisungen an den Operanden

Unter den folgenden Bedingungen gilt:

- Die Anweisung "UND-Verknüpfung" ist an ihren Eingängen verschalten mit mehreren Vergleichern (z. B. CMP ==) mit dem Datentyp String, WString oder Variant.
- Für die Anweisung "UND-Verknüpfung" steht bereits das Ergebnis "FALSE" durch gleichartige Vergleicher (Datentyp String, WString oder Variant) mit dem Signalzustand "FALSE" fest.

Ergebnis: Im Onlinemodus werden die nachfolgenden Vergleicher (Datentyp String, WString oder Variant) der Verschaltung als "inaktiv" angezeigt.

Beispiel:

Operand 1: CMP == (Datentyp String) liefert den Signalzustand "TRUE". CMP == wird als "aktiv" angezeigt.

Operand 2: CMP == (Datentyp INT) liefert den Signalzustand "TRUE". CMP == wird als "aktiv" angezeigt.

Operand 3: CMP == (Datentyp String) liefert den Signalzustand "FALSE". CMP == wird als "aktiv" angezeigt.

Operand 4: CMP == (Datentyp String) liefert den Signalzustand "TRUE". CMP == wird als "inaktiv" angezeigt.

Ergebnis der "UND-Verknüpfung": Signalzustand "FALSE".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

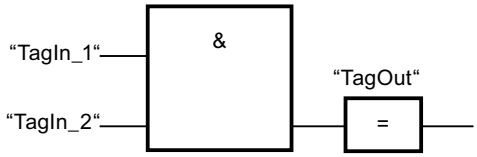

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand der Operanden "TagIn_1" und "TagIn_2" "1" ist und zurückgesetzt, wenn der Signalzustand der Operanden "TagIn_1" und "TagIn_2" "0" ist.

Siehe auch

UND-Wahrheitstabelle (Seite 978)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

Übersicht über die gültigen Datentypen (Seite 243)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Eingang einfügen (Seite 983)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

UND-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch UND entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	1	1
0	1	0
1	0	0
0	0	0

Siehe auch

&: UND-Verknüpfung (Seite 976)

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

>=1: ODER-Verknüpfung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der ODER–Wahrheitstabelle aus.

Beträgt der Signalzustand eines der Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand aller Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

Hinweis

Als "inaktiv" angezeigte Anweisungen an den Operanden

Unter den folgenden Bedingungen gilt:

- Die Anweisung "ODER-Verknüpfung" ist an ihren Eingängen verschalten mit mehreren Vergleichern (z. B. CMP ==) mit dem Datentyp String, WString oder Variant.
- Das Ergebnis ("TRUE") für die Anweisung "ODER-Verknüpfung" ist bereits durch gleichartige Vergleicher (Datentyp String, WString oder Variant) mit dem Signalzustand "TRUE" erfüllt.

Ergebnis: Im Onlinemodus werden die nachfolgenden Vergleicher (Datentyp String, WString oder Variant) der Verschaltung als "inaktiv" angezeigt.

Beispiel:

Operand 1: CMP == (Datentyp String) liefert den Signalzustand "FALSE". CMP == wird als "aktiv" angezeigt.

Operand 2: CMP == (Datentyp String) liefert den Signalzustand "TRUE". CMP == wird als "aktiv" angezeigt.

Operand 3: CMP == (Datentyp String) liefert den Signalzustand "TRUE". CMP == wird als "inaktiv" angezeigt.

Operand 4: CMP == (Datentyp INT) liefert den Signalzustand "TRUE". CMP == wird als "aktiv" angezeigt.

Ergebnis der "ODER-Verknüpfung": Signalzustand "TRUE".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

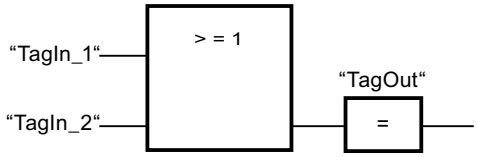

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand des Operanden "TagIn_1" oder des Operanden "TagIn_2" "1" ist.

Siehe auch

ODER-Wahrheitstabelle (Seite 981)

Beispiel für das Steuern eines Förderbands (Seite 7543)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Übersicht über die gültigen Datentypen (Seite 243)

Eingang einfügen (Seite 983)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

ODER-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	0	1
0	1	1
1	1	1
0	0	0

Siehe auch

>=1: ODER-Verknüpfung (Seite 979)

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

X: EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" fragen Sie das Ergebnis einer Signalzustandsabfrage entsprechend der EXKLUSIV ODER-Wahrheitstabelle ab.

Bei der Anweisung "EXKLUSIV ODER-Verknüpfung" ist der Signalzustand "1", wenn der Signalzustand eines der beiden angegebenen Operanden "1" ist. Bei der Abfrage von mehr als zwei Operanden ist das gemeinsame Verknüpfungsergebnis "1", wenn eine ungerade Anzahl der abgefragten Operanden das Abfrageergebnis "1" liefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

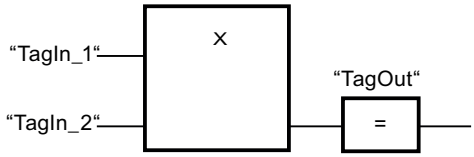

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand eines der beiden Operanden "TagIn_1" und "TagIn_2" "1" ist. Wenn beide Operanden den Signalzustand "1" oder "0" liefern, wird der Ausgang "TagOut" zurückgesetzt.

Siehe auch

EXKLUSIV ODER-Wahrheitstabelle (Seite 982)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Übersicht über die gültigen Datentypen (Seite 243)

Eingang einfügen (Seite 983)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

EXKLUSIV ODER-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch EXKLUSIV ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	0	1
0	1	1
1	1	0
0	0	0

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von drei Operanden durch EXKLUSIV ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Signalzustand des dritten Operanden	Verknüpfungsergebnis
1	0	0	1
0	1	1	0
0	1	0	1
1	0	1	0

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Signalzustand des dritten Operanden	Verknüpfungsergebnis
0	0	1	1
1	1	0	0
1	1	1	1
0	0	0	0

Siehe auch

X: EXKLUSIV ODER-Verknüpfung (Seite 981)

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Eingang einfügen

Beschreibung

Mit der Anweisung "Eingang einfügen" erweitern Sie die Box einer der folgenden Anweisungen durch einen Eingang:

- "UND-Verknüpfung"
- "ODER-Verknüpfung"
- "EXKLUSIV ODER-Verknüpfung"

Durch die Erweiterung einer Anweisungsbox können Sie den Signalzustand mehrerer Operanden abfragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingang einfügen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

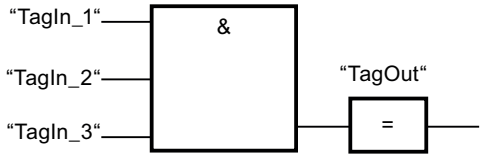

Die Box der Anweisung "UND-Verknüpfung" wurde durch einen zusätzlichen Eingang erweitert, an dem der Signalzustand des Operanden "TagIn_3" abgefragt wird. Der Ausgang "TagOut" wird gesetzt, wenn die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" den Signalzustand "1" liefern.

Siehe auch

&: UND-Verknüpfung (Seite 976)

>=1: ODER-Verknüpfung (Seite 979)

X: EXKLUSIV ODER-Verknüpfung (Seite 981)

Übersicht über die gültigen Datentypen (Seite 243)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE).

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

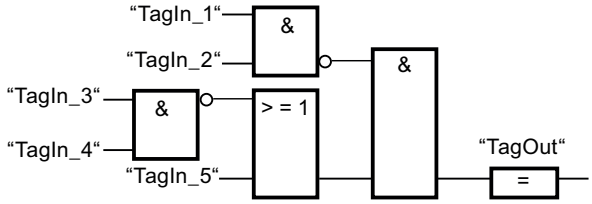

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Eingang "TagIn_1" und/oder "TagIn_2" liefert den Signalzustand "0".
- Der Eingang "TagIn_3" und/oder "TagIn_4" liefert den Signalzustand "0" oder der Eingang "TagIn_5" liefert den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Beispiel für das Erfassen der Raumtemperatur (Seite 7553)

Programmierbeispiele FUP (Seite 7543)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

=: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" setzten Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Boxeingang "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Boxeingang wird direkt dem über der Zuweisen-Box stehenden Operanden zugewiesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

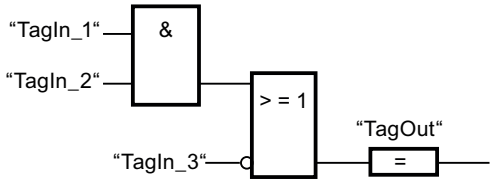

Der Operand "TagOut" am Ausgang der Anweisung "Zuweisung" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Eingänge "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand am Eingang "TagIn_3" ist "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

Programmierbeispiele FUP (Seite 7543)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

/=: Zuweisung negieren

Beschreibung

Mit der Anweisung "Zuweisung negieren" kehren Sie das Verknüpfungsergebnis (VKE) um und weisen es dem über der Box stehenden Operanden zu. Wenn das VKE am Eingang der Box "1" ist, wird der Operand zurückgesetzt. Wenn das VKE am Eingang der Box "0" ist, wird der Operand auf den Signalzustand "1" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung negieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das negierte VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

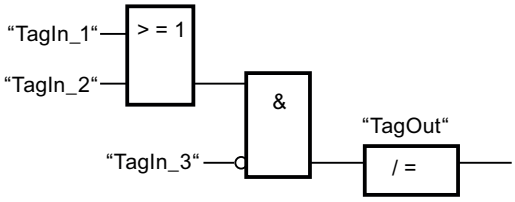

Der Operand "TagOut" wird zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" oder "TagIn_2" hat den Signalzustand "1".
- Der Operand "TagIn_3" hat den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

R: Ausgang rücksetzen

Beschreibung

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "0" zurückgesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand>	Output	BOOL	E, A, M, D, L	E, A, M, D, L, T, Z	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

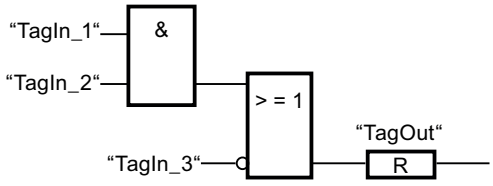

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "0".

Siehe auch

Beispiel für das Steuern eines Förderbands (Seite 7543)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

S: Ausgang setzen

Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

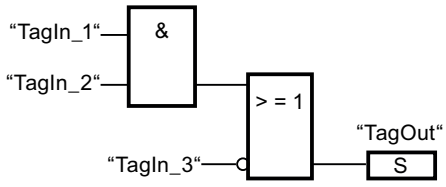

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

SET_BF: Bitfeld setzen

Beschreibung

Mit der Anweisung "Bitfeld setzen" setzen Sie mehrere Bits ab einer bestimmten Adresse.

Die Anzahl der zu setzenden Bits bestimmen Sie durch den Wert der Konstante am Eingang N. Die Adresse des ersten zu setzenden Bits wird durch <Operand> festgelegt. Der Wert von <Operand> darf nicht größer als die Anzahl der Bits in einem selektierten Byte sein. Ist dieser Wert größer, wird die Anweisung nicht ausgeführt und die Fehlermeldung "Bereichsüberschreitung für Index <Operand1>" wird angezeigt. Die Bits bleiben so lange gesetzt, bis sie explizit durch eine andere Anweisung zurückgesetzt werden.

Die Anweisung wird nur ausgeführt, wenn der Freigabeeingang EN den Signalzustand "1" liefert.

Bitfelder vom Datentyp PLC-Datentyp, STRUCT oder ARRAY

Bei Strukturen vom Datentyp PLC-Datentyp, STRUCT oder ARRAY werden maximal so viele Bits gesetzt, wie die Struktur Bits enthält:

- Wenn Sie z. B. am Eingang N den Wert "20" angeben und die Struktur nur 10 Bits enthält, dann werden auch nur diese 10 Bits gesetzt.
- Wenn Sie z. B. am Eingang N den Wert "5" angeben und die Struktur 10 Bits enthält, dann werden genau 5 Bits gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld setzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
N	Input	UINT	Konstante	Konstante	Anzahl der zu setzenden Bits
<Operand>	Output	BOOL	E, A, M Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL	E, A, M Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL	Zeiger auf das erste Bit, das gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

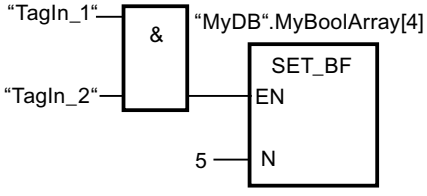

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

RESET_BF: Bitfeld rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld rücksetzen" setzen Sie mehrere Bits ab einer bestimmten Adresse zurück.

Die Anzahl der zurückzusetzenden Bits bestimmen Sie durch den Wert der Konstante am Eingang N. Die Adresse des ersten zurückzusetzenden Bits wird durch <Operand> festgelegt. Der Wert der Konstante darf nicht größer als die Anzahl der Bits in einem selektierten Byte sein. Ist dieser Wert größer, wird die Anweisung nicht ausgeführt und die Fehlermeldung "Bereichsüberschreitung für Index <Operand 1>" wird angezeigt. Die Bits bleiben so lange zurückgesetzt, bis sie explizit durch eine andere Anweisung gesetzt werden.

Die Anweisung wird nur ausgeführt, wenn der Freigabeeingang EN den Signalzustand "1" liefert.

Bitfelder vom Datentyp PLC-Datentyp, STRUCT oder ARRAY

Bei Strukturen vom Datentyp PLC-Datentyp, STRUCT oder ARRAY werden maximal so viele Bits zurückgesetzt, wie die Struktur Bits enthält:

- Wenn Sie z. B. am Eingang N den Wert "20" angeben und die Struktur nur 10 Bits enthält, dann werden auch nur diese 10 Bits zurückgesetzt.
- Wenn Sie z. B. am Eingang N den Wert "5" angeben und die Struktur 10 Bits enthält, dann werden genau 5 Bits zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
N	Input	UINT	Konstante	Konstante	Anzahl der zurückzusetzenden Bits
<Operand>	Output	BOOL	E, A, M Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL	E, A, M Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL	Zeiger auf das erste Bit, das zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

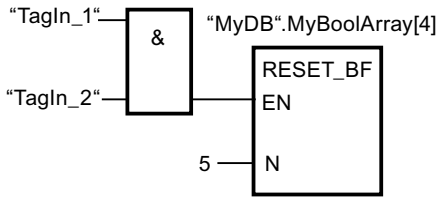

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SR: Flipflop setzen/rücksetzen

Beschreibung

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
S	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Setzen freigeben
R1	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Rücksetzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Operand, der gesetzt oder zurückgesetzt wird
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

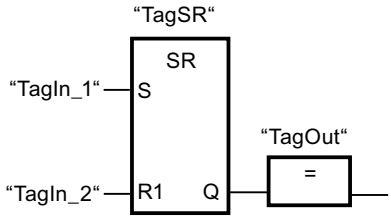

Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagSR" und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

RS: Flipflop rücksetzen/setzen

Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen R und S1 wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
R	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Rücksetzen freigeben
S1	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Setzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Operand, der zurückgesetzt oder gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

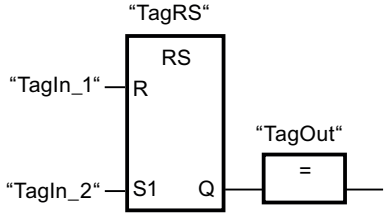

Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

P: Operand auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

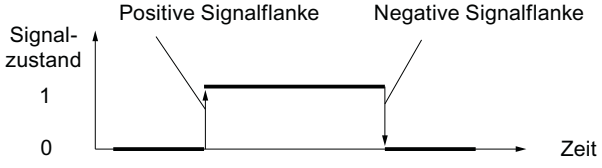

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

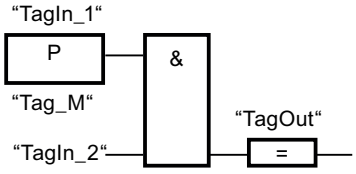

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn_1" liegt eine positive Signalflanke vor.
- Der Signalzustand des Operanden "TagIn_2" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Programmierbeispiele FUP (Seite 7543)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

N: Operand auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

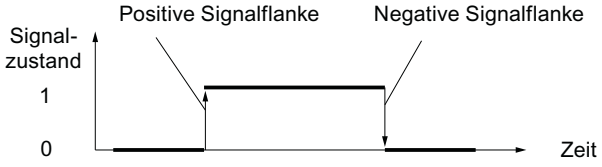

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
<Operand1>	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

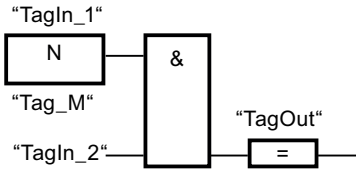

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn_1" liegt eine negative Signalflanke vor.
- Der Signalzustand des Operanden "TagIn_2" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

FUP-Elemente bearbeiten (Seite 7526)

P=: Operand bei positiver Signalflanke setzen

Beschreibung

Mit der Anweisung "Operand bei positiver Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand2>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand1>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand2> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand2>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand1>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei positiver Signalflanke setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand2>	Output	BOOL	E, A, M, D, L	Operand, der bei einer positiven Signalflanke gesetzt wird.
<Operand1>	InOut	BOOL	E, A, M, D, L	Flankenmerker

Beispiel

Das folgende Beispiel zeigt die Parameter der Anweisung:

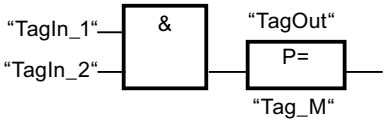

Der Ausgang "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Anweisungsbox von "0" auf "1" wechselt (positive Signalflanke). In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

N=: Operand bei negativer Signalflanke setzen

Beschreibung

Mit der Anweisung "Operand bei negativer Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei negativer Signalflanke setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Output	BOOL	E, A, M, D, L	Operand, der bei einer negativen Signalflanke gesetzt wird.
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

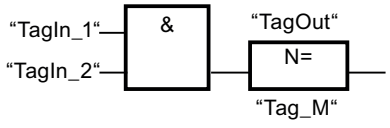

Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Anweisungsbox von "1" auf "0" wechselt (negative Signalflanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

FUP-Elemente bearbeiten (Seite 7526)

P_TRIG: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine positive Signalflanke erfasst wird. In allen anderen Fällen liefert der Ausgang den Signalzustand "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

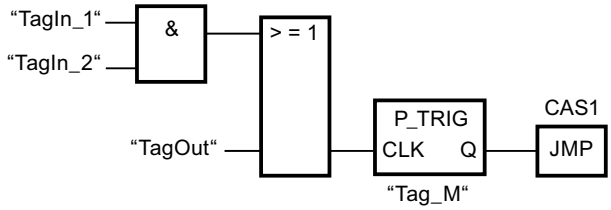

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

N_TRIG: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker (<Operand>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine negative Signalflanke erfasst wird. In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

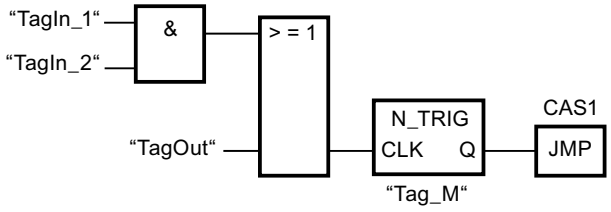

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

R_TRIG: Positive Signalflanke erkennen

Beschreibung

Mit der Anweisung "Positive Signalflanke erkennen" können Sie eine Zustandsänderung von "0" auf "1" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "0" auf "1" erkennt, wird am Ausgang Q eine positive Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Positive Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
CLK	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

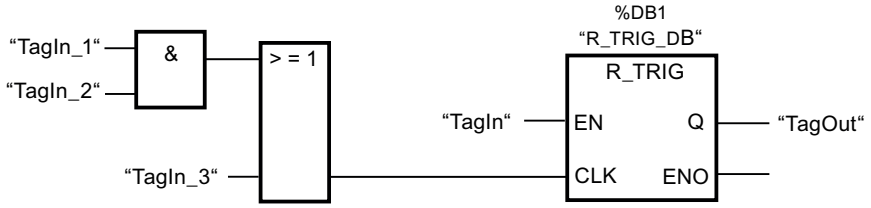

In der Variablen "R_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3" eine Änderung im Signalzustand von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

F_TRIG: Negative Signalflanke erkennen

Beschreibung

Mit der Anweisung "Negative Signalflanke erkennen" können Sie eine Zustandsänderung von "1" auf "0" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "1" auf "0" erkennt, wird am Ausgang Q eine negative Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negative Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
CLK	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

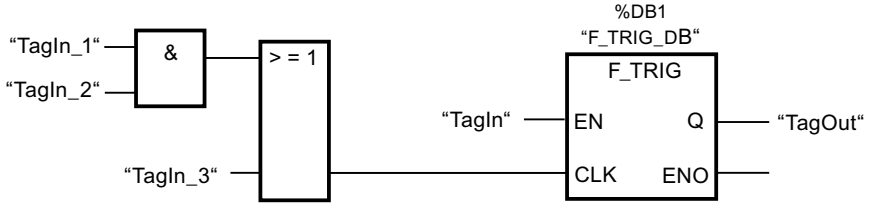

In der Variablen "F_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3" eine Änderung im Signalzustand von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die parametrierte Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TP_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Impuls erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Impulsausgang
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

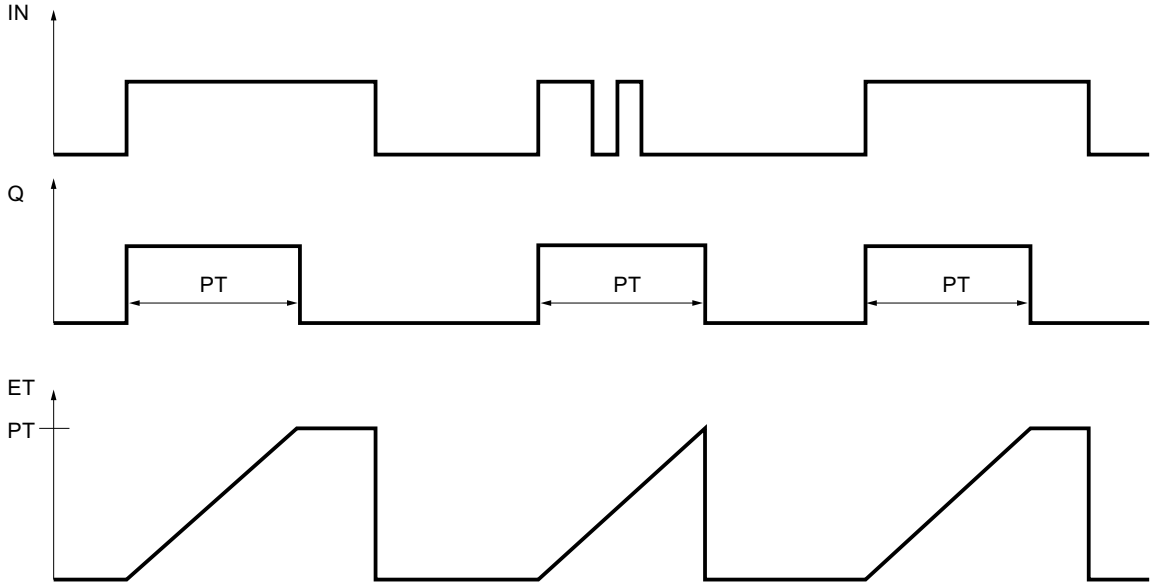

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

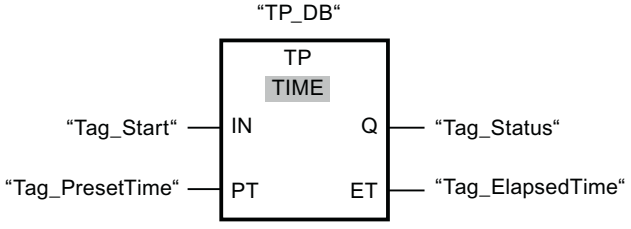

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	TRUE
ET	Tag_ElapsedTime	von T#0s => T#10s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert. Wenn die Zeit abgelaufen ist, dann wird der Operand "Tag_Status" wieder auf den Signalzustand "0" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die parametrierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TON_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME, TON_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Einschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

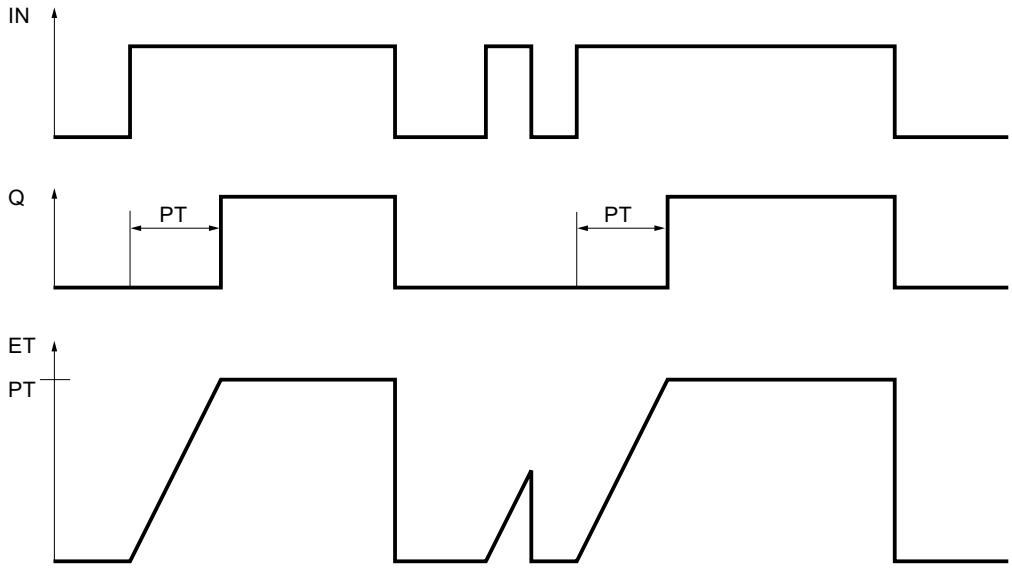

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

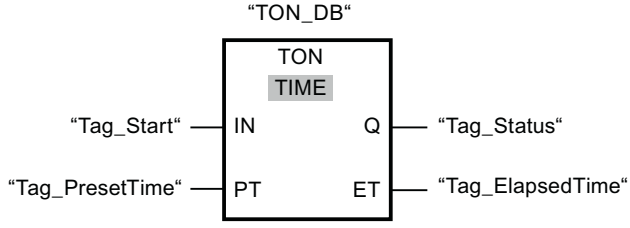

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	FALSE; nach 10s => TRUE
ET	Tag_ElapsedTime	von T#0s => T#10s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert. Wenn der Signalzustand am Operand "Tag_Start" von "1" auf "0" wechselt, wird der Operand "Tag_Status" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die parametrierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt (positive Signalflanke), läuft die parametrierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TOF_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME, TOF_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Ausschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

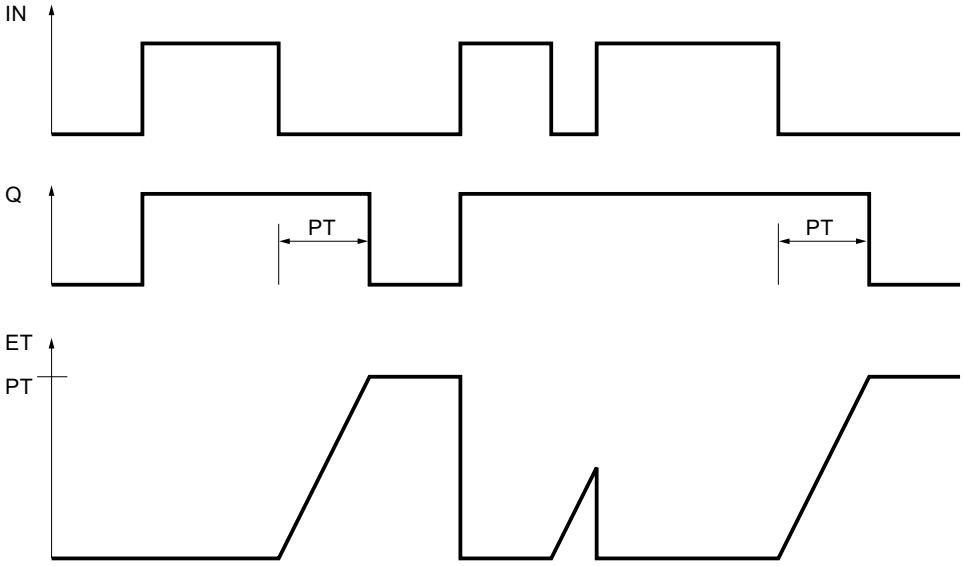

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

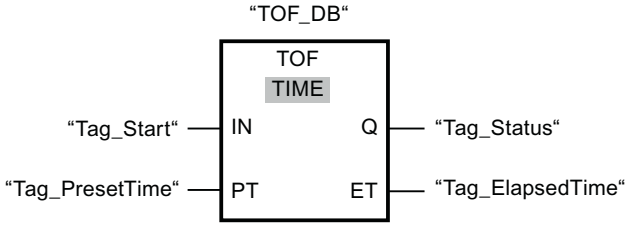

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"; Signalwechsel "1" => "0"
PT	Tag_PresetTime	T#10s
Q	Tag_Status	TRUE
ET	Tag_ElapsedTime	von T#10s => T#0s

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" auf TRUE gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf FALSE zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und die parametrierte Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn der aktuelle Zeitwert PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Eingang IN auf "0" wechselt.

Der Eingang R setzt die Ausgänge ET und Q unabhängig vom Signalzustand am Starteingang zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TONR_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME, TONR_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Starteingang
R	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":

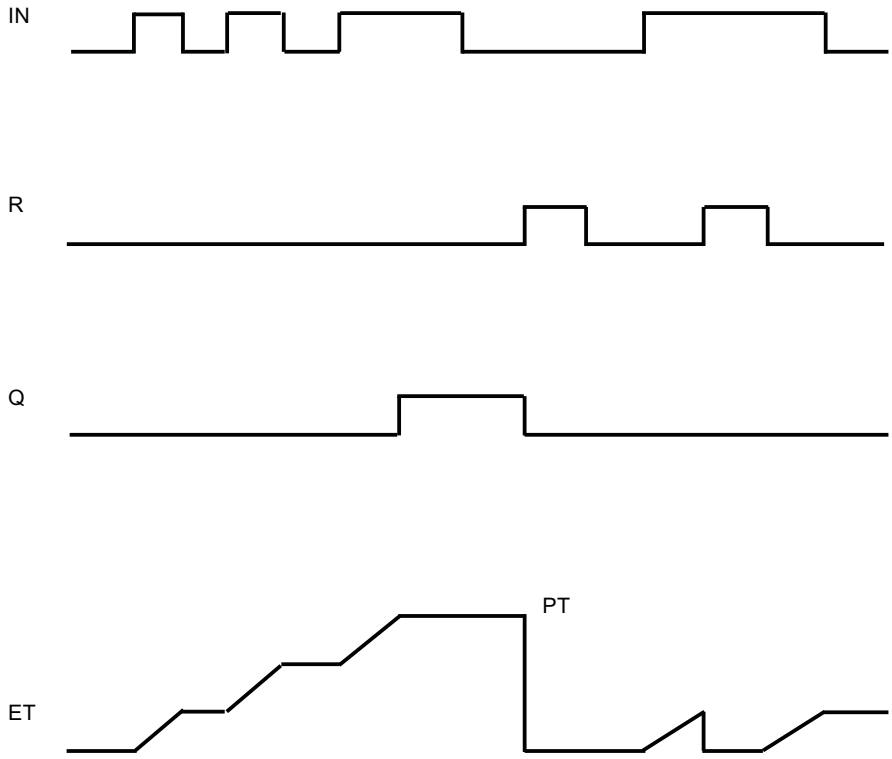

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

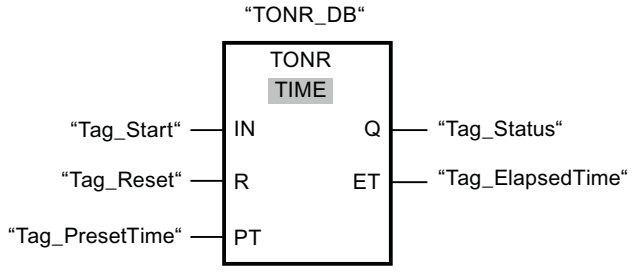

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Start	Signalwechsel "0" => "1"
PT	Tag_PresetTime	T#10s

Parameter	Operand	Wert
Q	Tag_Status	FALSE; nach 10s => TRUE
ET	Tag_ElapsedTime	Signalwechsel von "0" => "1": Die Zeit T#10s läuft ab. Nach 5s Signalwechsel von "1" => "0": Die Zeit am Operanden "Tag_ElapsedTime" bleibt bei T#5s stehen. Nach ca. 2s erneuter Signalwechsel von "0" => "1": Die Zeit am Operanden "Tag_ElapsedTime" läuft bei T#5s weiter ab.

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange der Operand "Tag_Start" den Signalzustand "1" hat, läuft die Zeit ab. Wenn der Signalzustand am Operanden "Tag_Start" von "1" auf "0" wechselt, bleibt die Zeit stehen und der aktuelle Zeitwert am Operanden "Tag_ElapsedTime" wird gemerkt. Wechselt der Signalzustand am Operanden "Tag_Start" wieder von "0" auf "1", dann läuft die Zeit ab dem Zeitwert weiter, der beim Wechsel von "1" auf "0" gemerkt worden war. Wenn der am Parameter PT angegebene Zeitwert erreicht ist, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TP: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Impuls. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab, unabhängig vom weiteren Verlauf des VKE. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Ablauf der IEC-Zeit nicht. Solange die IEC-Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn die IEC-Zeit abgelaufen ist, liefert der Zeitstatus den Signalzustand "0".

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TP_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente Q der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen.

Die Ausführung der Anweisung "Zeit als Impuls starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp		Speicherbe-reich	Beschreibung
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TP_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

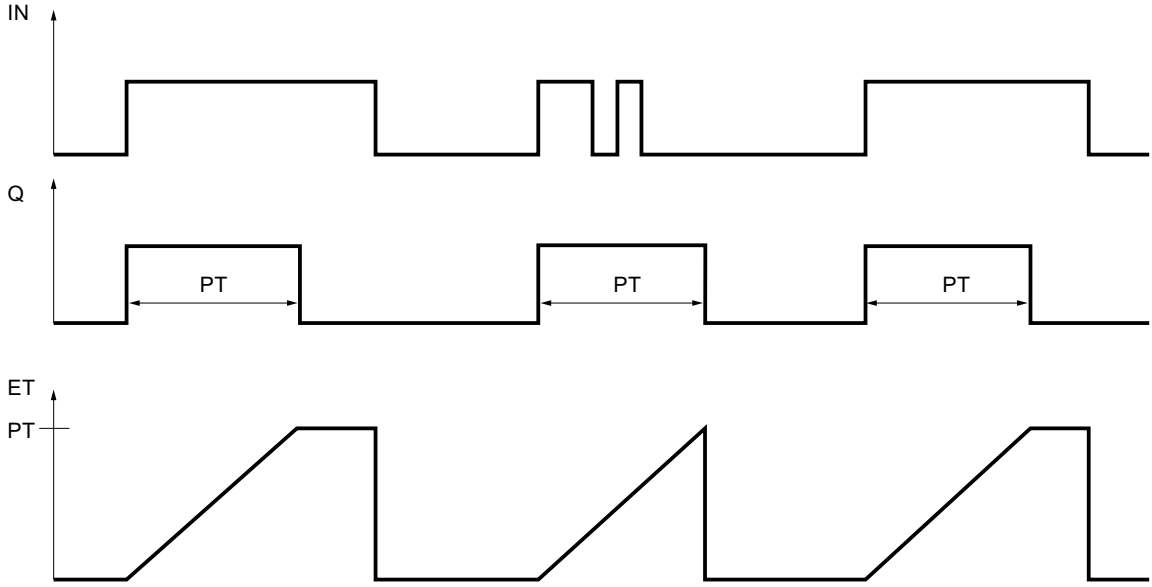

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

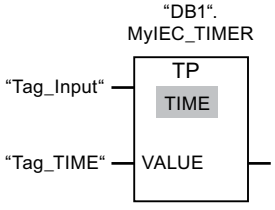

Die Anweisung "Zeit als Impuls starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "0" auf "1" wechselt. Die Zeit "DB1".MyIEC_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.

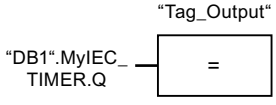

Solange die Zeit "DB1".MyIEC_TIMER läuft, liefert der Zeitstatus ("DB1".MyIEC_TIMER.Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn die IEC-Zeit abgelaufen ist, wechselt der Signalzustand des Zeitstatus auf "0" und der Operand "Tag_Output" wird zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TON: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Einschaltverzögerung. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab. Der Ausgang liefert den Signalzustand "1", wenn das VKE am Eingang der Anweisung den Signalzustand "1" führt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die laufende IEC-Zeit zurückgesetzt. Die Abfrage des Zeitstatus auf "1" liefert in diesem Fall den Signalzustand "0". Mit der Erfassung der nächsten positiven Signalflanke am Eingang der Anweisung startet die IEC-Zeit wieder.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TON_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME, TON_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen.

Die Ausführung der Anweisung "Zeit als Einschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TON_TIME	IEC_TIMER, IEC_LTIMER, TON_TIME, TON_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

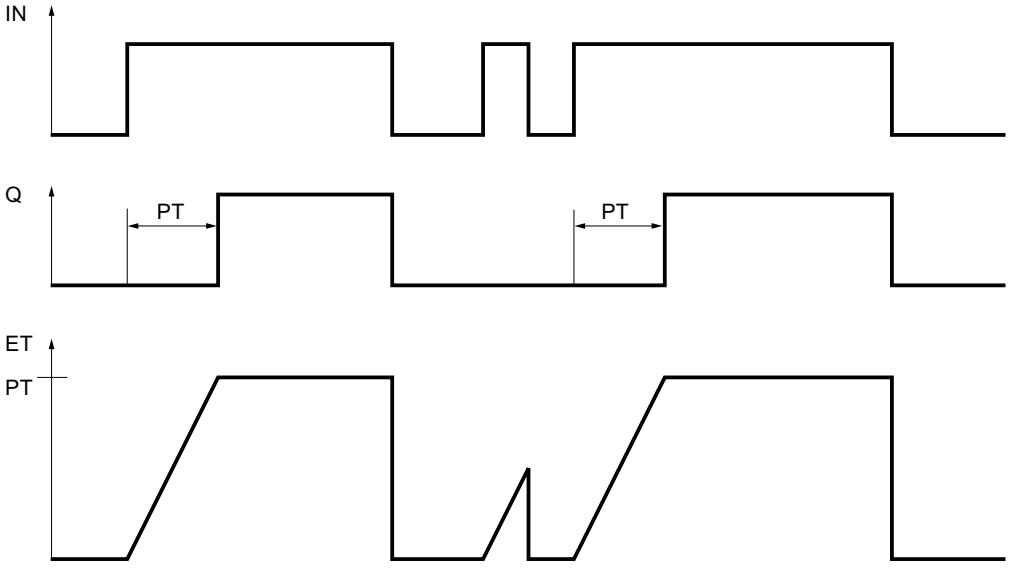

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

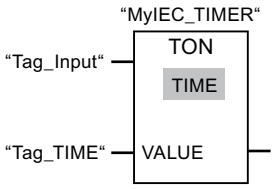

Die Anweisung "Zeit als Einschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "0" auf "1" wechselt. Die Zeit "MyIEC_TIMER" wird für die Zeitdauer gestartet, die im Operanden "Tag_TIME" gespeichert ist.

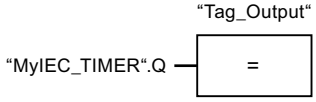

Wenn die Zeit "MyIEC_TIMER" abgelaufen ist und der Operand "Tag_Input" den Signalzustand "1" führt, liefert die Abfrage des Zeitstatus ("MyIEC_TIMER".Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn der Signalzustand des Operanden "Tag_Input" auf "0" wechselt, liefert die Abfrage des Zeitstatus den Signalzustand "0" und der Operand "Tag_Output" wird zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TOF: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Ausschaltverzögerung. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "0" führt. Wenn das VKE von "1" auf "0" (negative Signalflanke) wechselt, wird die IEC-Zeit mit der angegebenen Zeitdauer gestartet. Solange die IEC-Zeit läuft, bleibt der Zeitstatus auf den Signalzustand "1". Wenn die Zeit abgelaufen ist und das VKE am Eingang der Anweisung den Signalzustand "0" führt, wird der Zeitstatus auf den Signalzustand "0" gesetzt. Wechselt das VKE vor dem Ablauf der Zeit auf "1", wird die laufende IEC-Zeit zurückgesetzt und der Zeitstatus bleibt auf den Signalzustand "1".

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TOF_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME, TOF_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen.

Die Ausführung der Anweisung "Zeit als Ausschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp		Speicherbe-reich	Beschreibung
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TOF_TIME	IEC_TIMER, IEC_LTI-MER, TOF_TIME, TOF_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

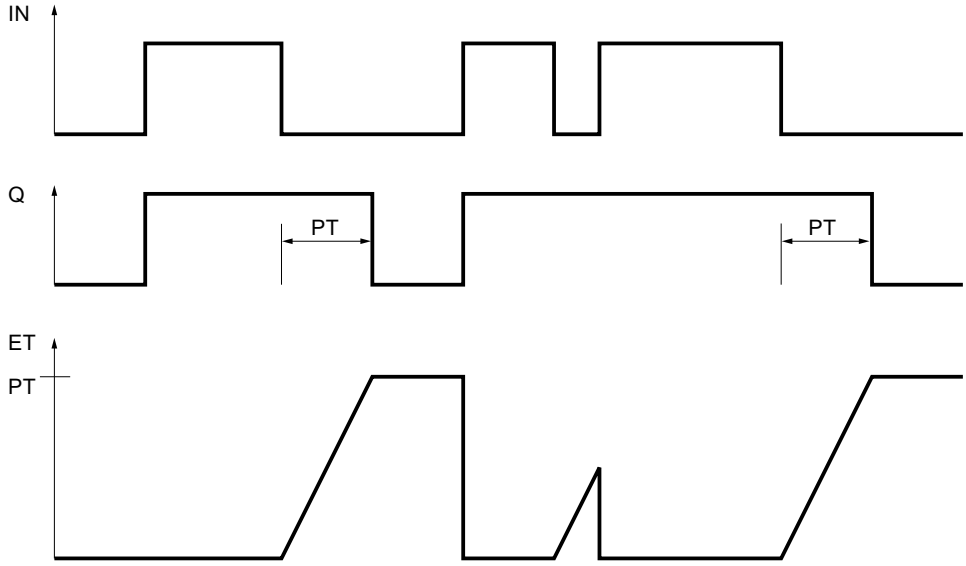

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

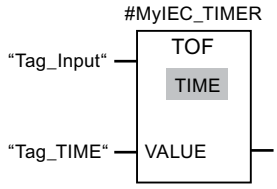

Die Anweisung "Zeit als Ausschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag_Input" von "1" auf "0" wechselt. Die Zeit #MyIEC_TIMER wird für die Zeitdauer gestartet, die im Operanden "Tag_TIME" gespeichert ist.

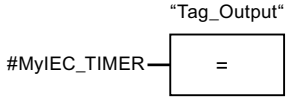

Solange die Zeit #MyIEC_TIMER läuft, liefert die Abfrage des Zeitstatus (#MyIEC_TIMER.Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt. Wenn die Zeit abgelaufen ist und der Operand "Tag_Input" den Signalzustand "0" führt, liefert die Abfrage des Zeitstatus den Signalzustand "0". Wechselt der Signalzustand des Operanden "Tag_Input" vor dem Ablauf der Zeit #MyIEC_TIMER auf "1", wird die Zeit zurückgesetzt. Beim Signalzustand "1" des Operanden "Tag_Input" liefert die Abfrage des Zeitstatus den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" erfassen Sie, wie lange das Signal am Eingang der Anweisung "1" ist. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Solange das VKE "1" ist, wird die Zeit erfasst. Wechselt das VKE auf "0", wird die Anweisung angehalten. Wechselt das VKE erneut auf "1", wird die Zeiterfassung fortgesetzt. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn die erfasste Zeit den Wert der angegebenen Zeitdauer überschreitet und das VKE am Eingang der Spule "1" führt.

Der Zeitstatus und die aktuell abgelaufene Zeit können durch die Anweisung "Zeit rücksetzen" auf "0" zurückgesetzt werden.

Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC_TIMER-Struktur aktualisiert.

Für CPU S7-1200

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER oder TONR_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME oder IEC_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Für CPU S7-1500

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME, TONR_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung wird die IEC_TIMER-Struktur aktualisiert. Der Zeitwert am Ausgang ET wird nur dann aktualisiert, wenn die Ausgänge ET oder Q abgefragt werden (z. B. "MyTimer".Q bzw. "MyTimer".ET).
- Bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	InOut	IEC_TIMER, TONR_TIME	IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung:

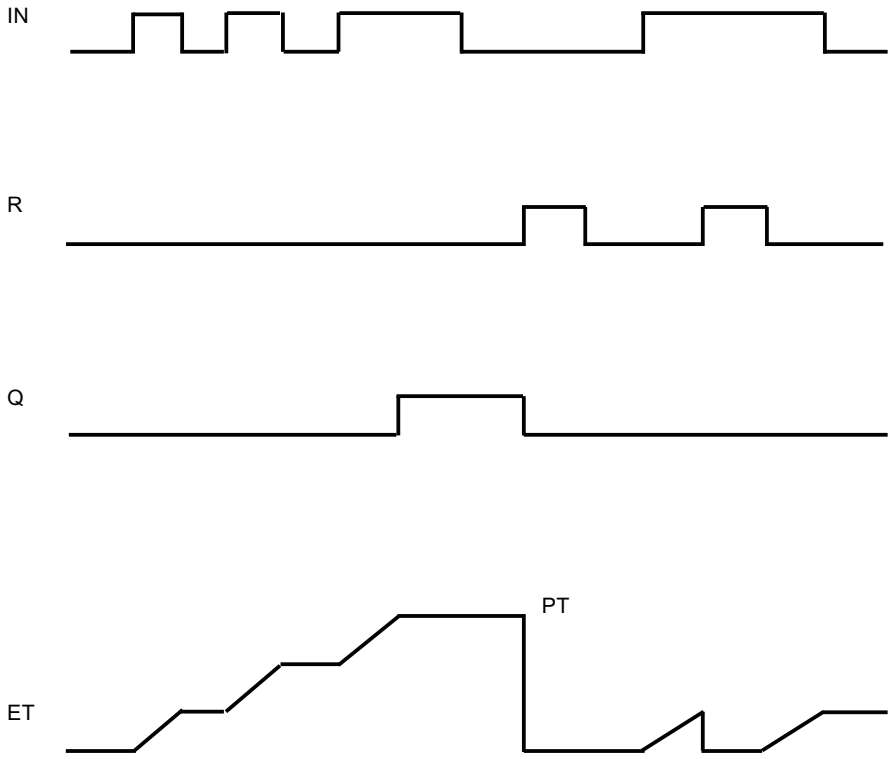

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

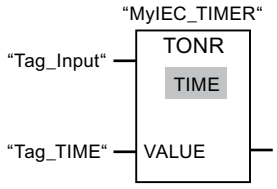

Die Anweisung "Zeit akkumulieren" wird bei einer positiven Signalflanke im VKE ausgeführt. Solange der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Zeit erfasst.

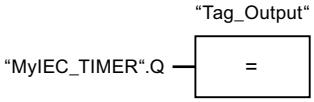

Wenn die erfasste Zeit den Wert des Operanden "Tag_TIME" überschreitet, liefert die Abfrage des Zeitstatus ("MyIEC_TIMER".Q) den Signalzustand "1" und der Operand "Tag_Output" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

RT: Zeit rücksetzen (Seite 1029)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

RT: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die IEC-Zeit, die zurückgesetzt werden soll, geben Sie an, indem Sie in den Platzhalter über der Anweisung den Namen des Datenbausteins eintragen, der die Struktur der IEC-Zeit enthält.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn die Anweisung ausgeführt wird, werden die Strukturkomponenten der IEC-Zeit im angegebenen Datenbaustein auf "0" zurückgesetzt. Wenn das VKE am Boxeingang "0" ist, wird die Anweisung nicht ausgeführt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Boxeingang wird direkt auf den Boxausgang übertragen.

Der Anweisung "Zeit rücksetzen" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

Parameter	Deklaration	Datentyp		Speicherbe-reich	Beschreibung
		S7-1200	S7-1500		
<IEC-Zeit>	InOut	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die zurückgesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

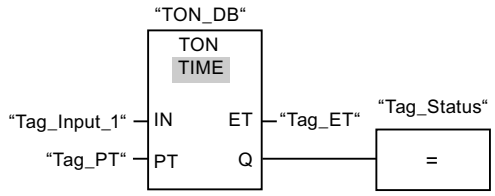

Wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PT" vorgegeben ist.

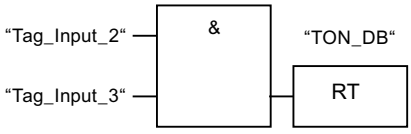

Wenn die Operanden "Tag_Input_2" und "Tag_Input_3" den Signalzustand "1" liefern, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Datenbaustein "TON_DB" abgelegte IEC-Zeit zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

PT: Zeitdauer laden

Beschreibung

Mit der Anweisung "Zeitdauer laden" parametriert die Zeitdauer einer IEC-Zeit. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisungsbox den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegeben IEC-Zeit.

Hinweis

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC_TIMER-Struktur.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
PT	Input	TIME	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer
<IEC-Zeit>	InOut	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, deren Dauer eingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

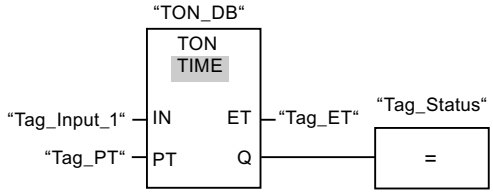

Wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PT" vorgegeben ist.

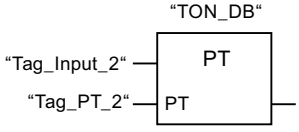

Wenn der Operand "Tag_Input_2" den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer "Tag_PT_2" in den Instanz-Datenbaustein "TON_DB" und überschreibt dabei den Wert des Operanden "Tag_PT"

innerhalb des Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim Zugriff auf "MyTimer".Q oder "MyTimer".ET ändern.

Hinweis

Der "Tag_Input_2" wird als Impulsmerker ausgeführt, damit das Laden der Zeitdauer nur über einen Programmzyklus erfolgt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

Legacy

S_IMPULS: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

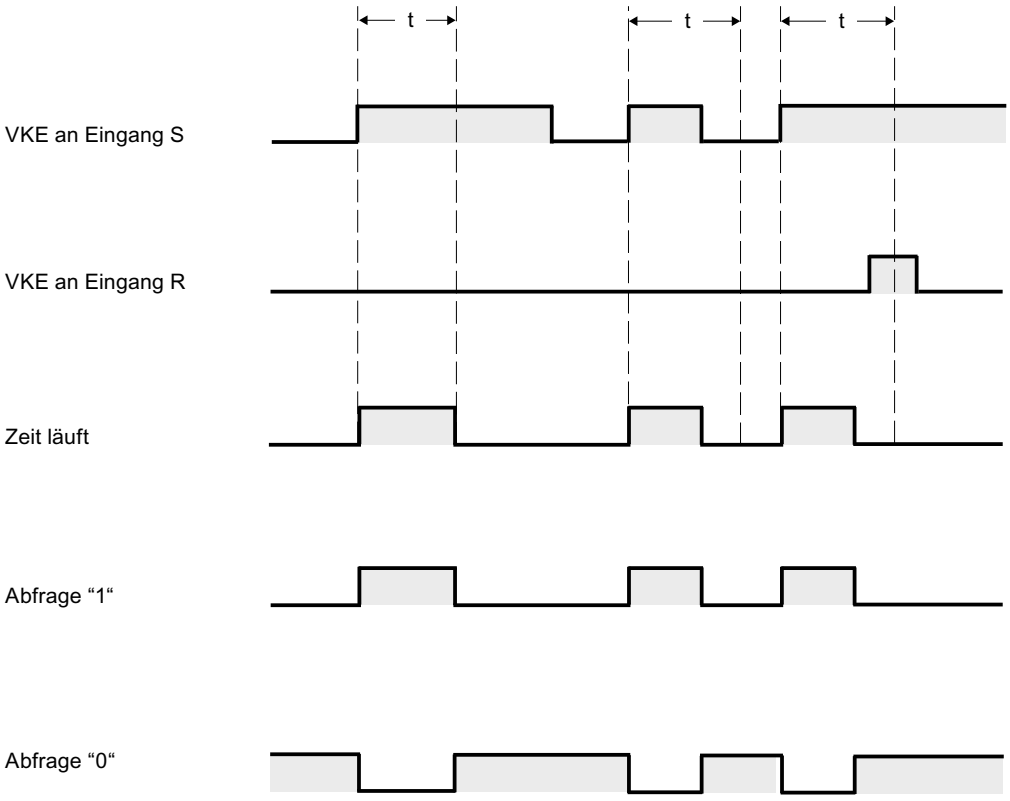

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

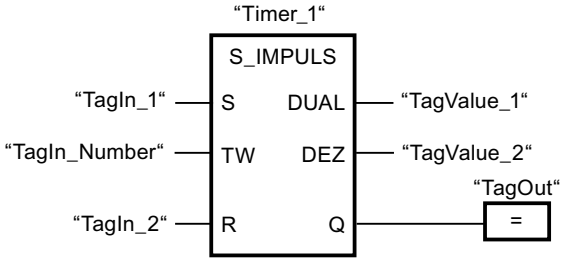

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Operand "TagIn_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

S_VIMP: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

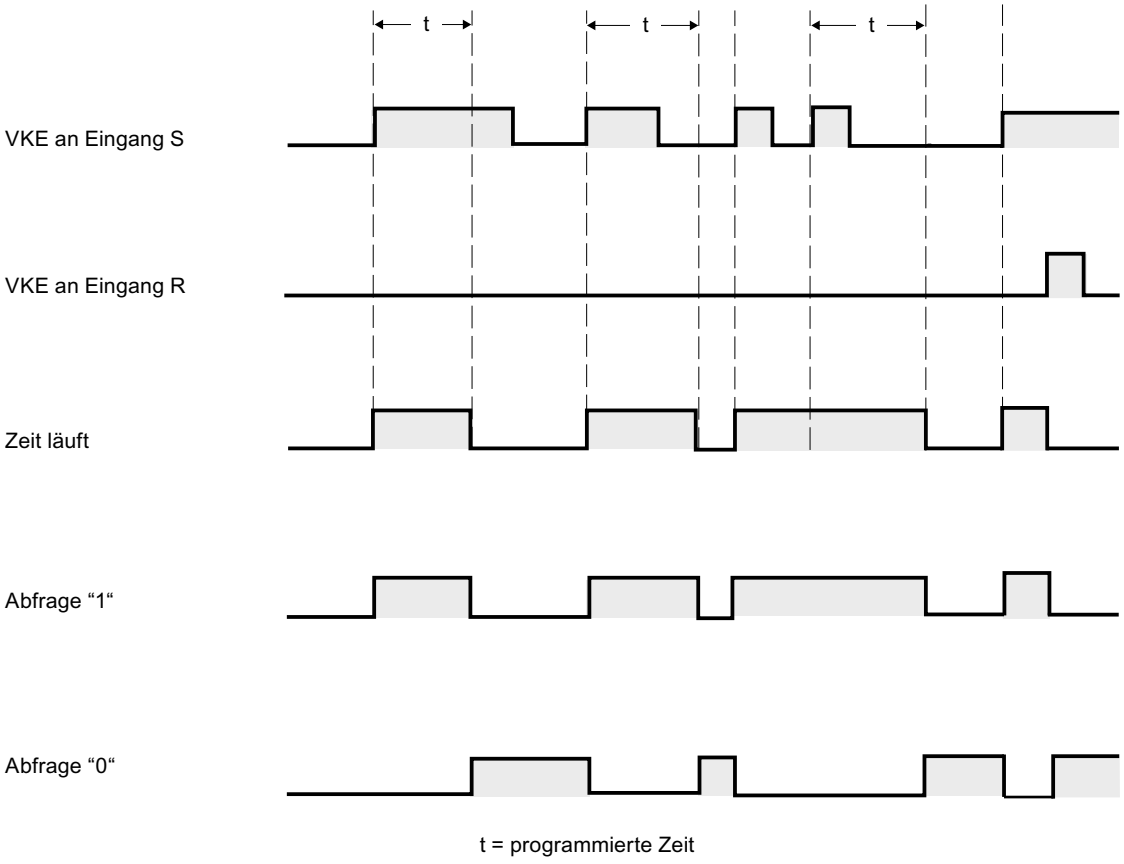

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

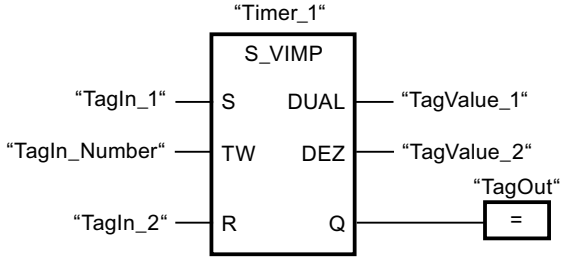

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

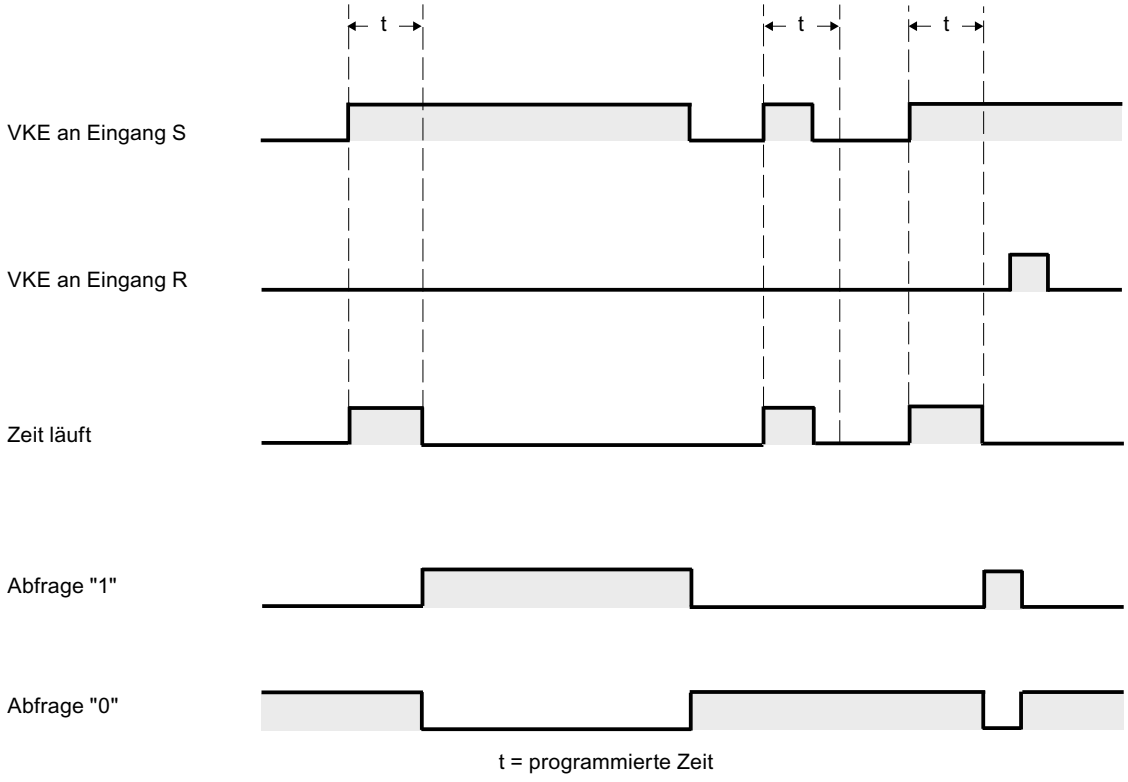

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

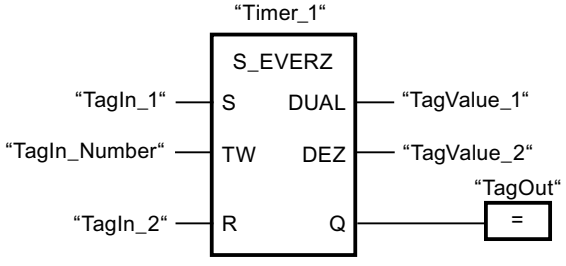

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

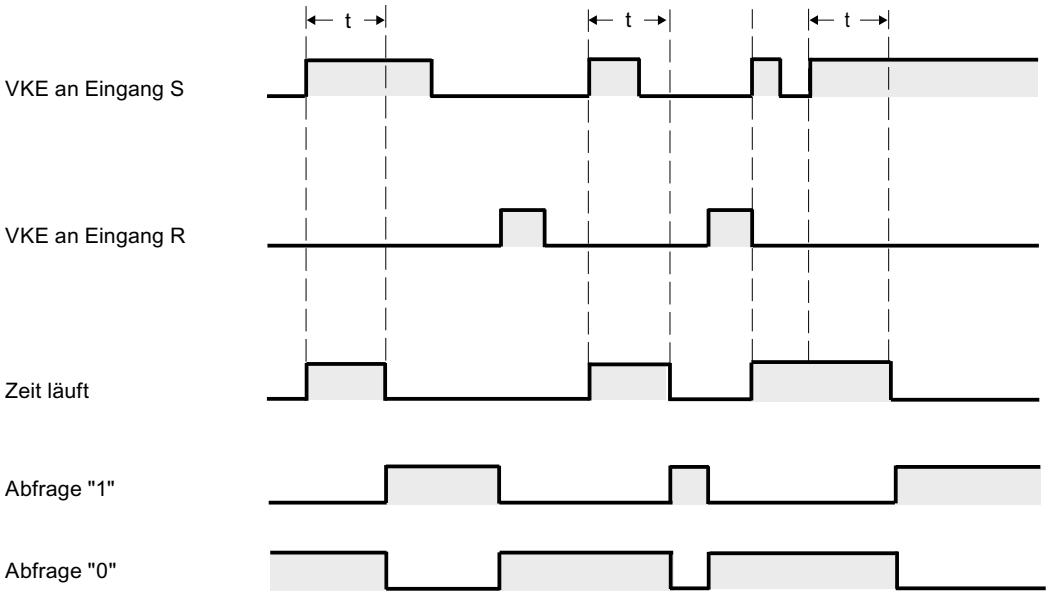

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

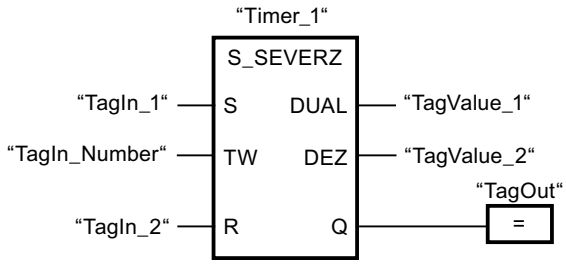

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, auch wenn der Signalzustand des Operanden "TagIn_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P oder Konstante	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

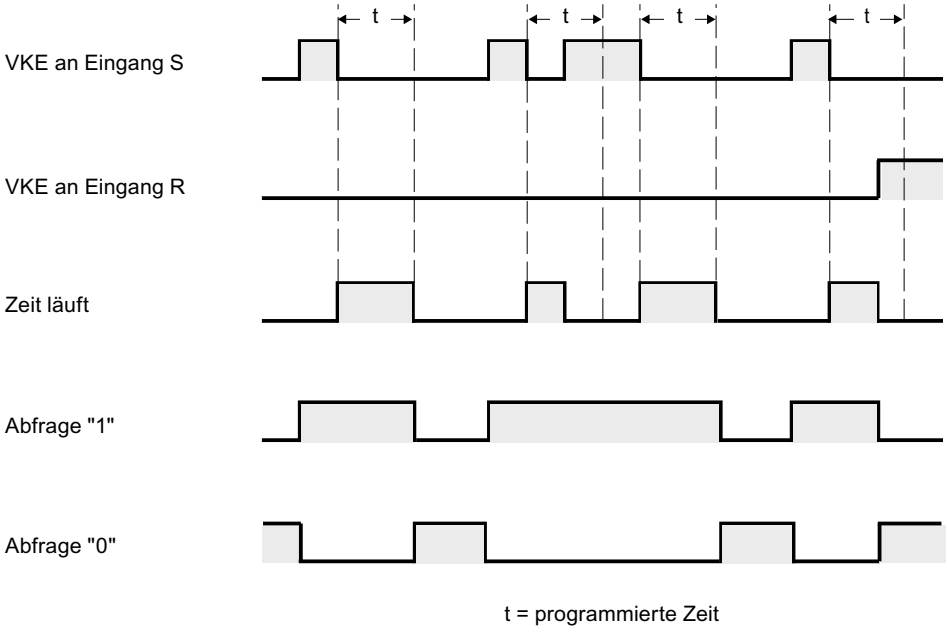

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

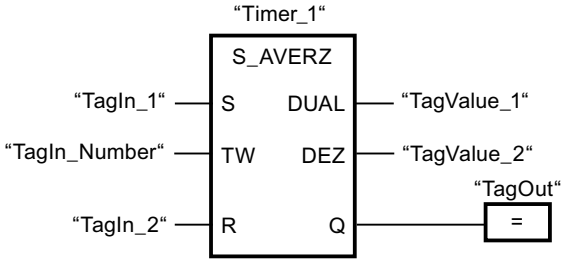

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

SI: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

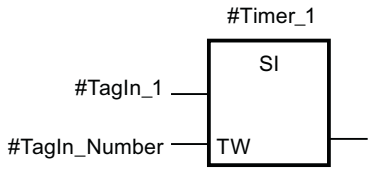

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab, solange der Signalzustand des Operanden #TagIn_1 "1" ist. Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten.

Netzwerk 2:

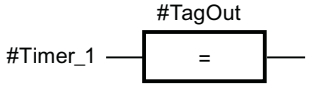

Solange die Zeit läuft, liefert der Operand #TagOut den Signalzustand "1".

Netzwerk 3:

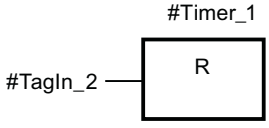

Wechselt der Signalzustand des Operanden #TagIn_2 von "0" auf "1", dann wird die Zeit zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

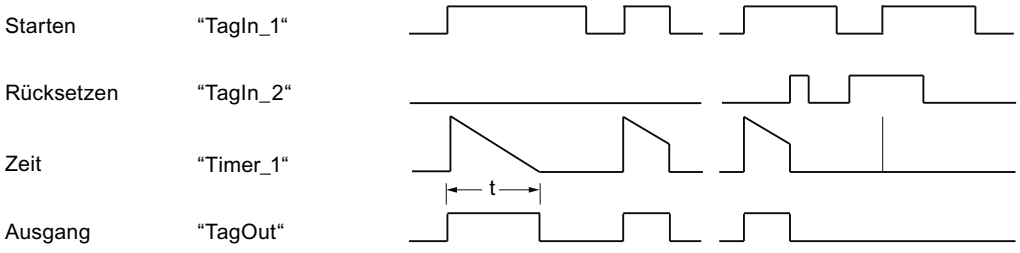

t = programmierte Zeitdauer

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu FUP (Seite 7493)

SV: Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer neu gestartet. Die Abfrage des Zeitstatus auf "1" ergibt den Signalzustand "0", wenn die Zeit abgelaufen ist.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

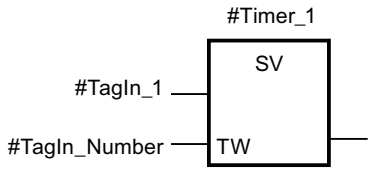

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab, ohne von einer negativen Signalflanke im VKE beeinflusst zu werden.

Netzwerk 2:

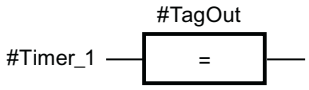

Solange die Zeit läuft, liefert der Operand #TagOut den Signalzustand "1".

Netzwerk 3:

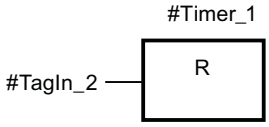

Wechselt der Signalzustand des Operanden #TagIn_1 erneut von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

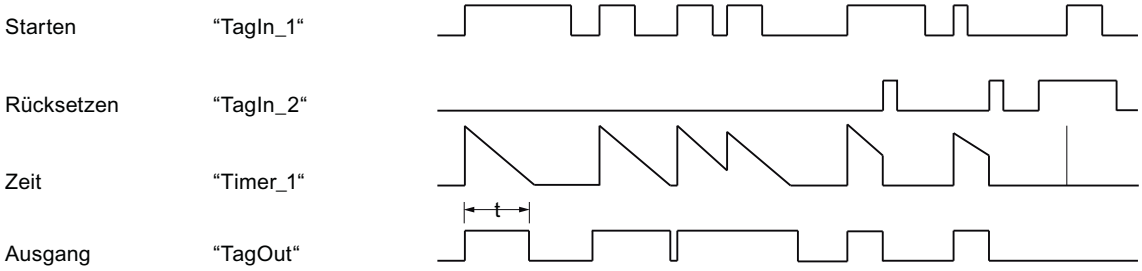

t = programmierte Zeitdauer

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SE: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der angegebenen Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

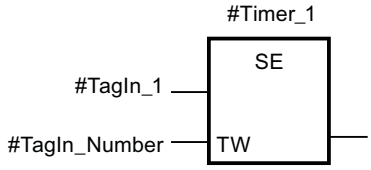

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab. Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit zurückgesetzt.

Netzwerk 2:

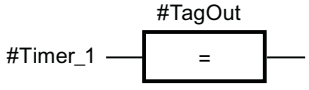

Wenn die Zeit abgelaufen ist, der Operand #TagIn_1 am Starteingang den Signalzustand "1" hat und die Zeit nicht zurückgesetzt wurde, dann ist der Operand #TagOut "1".

Netzwerk 3:

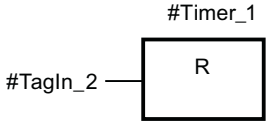

Wenn der Signalzustand des Operanden #TagIn_2 den Signalzustand "1" hat, dann wird die Zeit #Timer_1 und der Ausgang #TagOut zurückgesetzt.

Wenn Sie den #Timer_1 erneut starten wollen, dann muss der Signalzustand am Operand #TagIn_2 "0" sein und der Signalzustand am Starteingang #TagIn_1 von "0" auf "1" wechseln.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

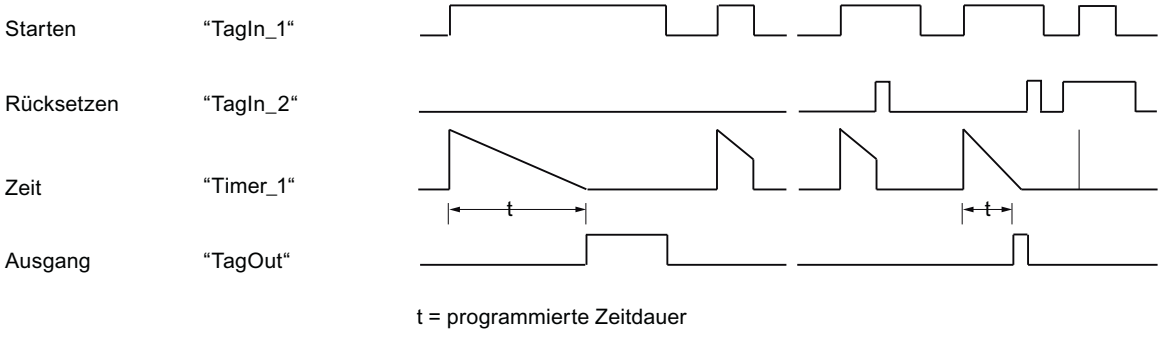

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SS: Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

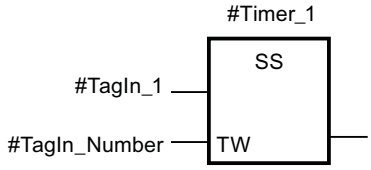

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab.

Netzwerk 2:

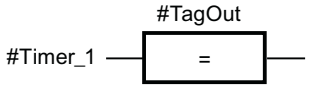

Wenn die Zeit abgelaufen ist, wird der Operand #TagOut auf "1" gesetzt.

Netzwerk 3:

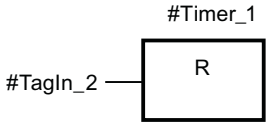

Wechselt der Signalzustand des Operanden #TagIn_1 von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden #TagIn_2 "1" ist, dann wird die Zeit #Timer_1 zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

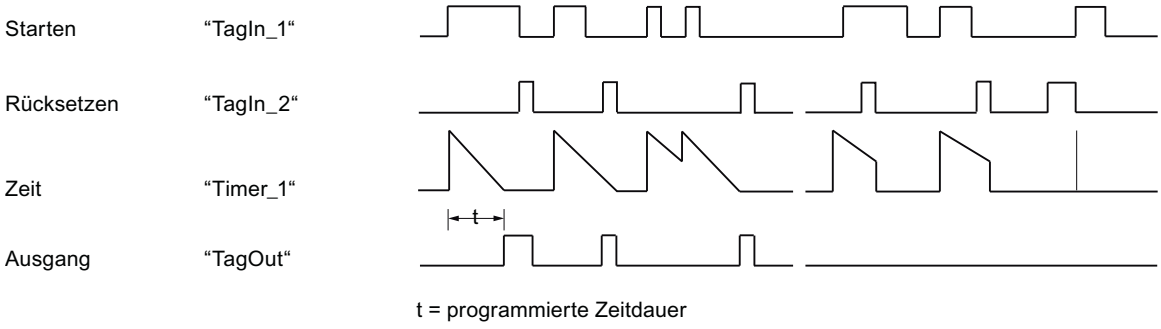

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SA: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist. Wenn das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	InOut/Input	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

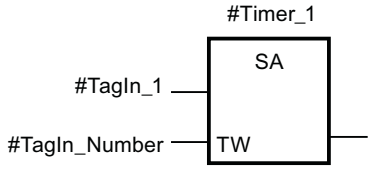

Wenn der Signalzustand des Operanden #TagIn_1 von "1" auf "0" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab.

Netzwerk 2:

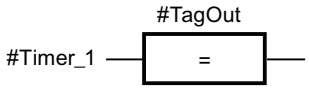

Solange die Zeit läuft, wird der Operand #TagOut auf "1" gesetzt.

Netzwerk 3:

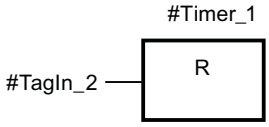

Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden #TagIn_2 "1" ist, dann wird die Zeit #Timer_1 zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

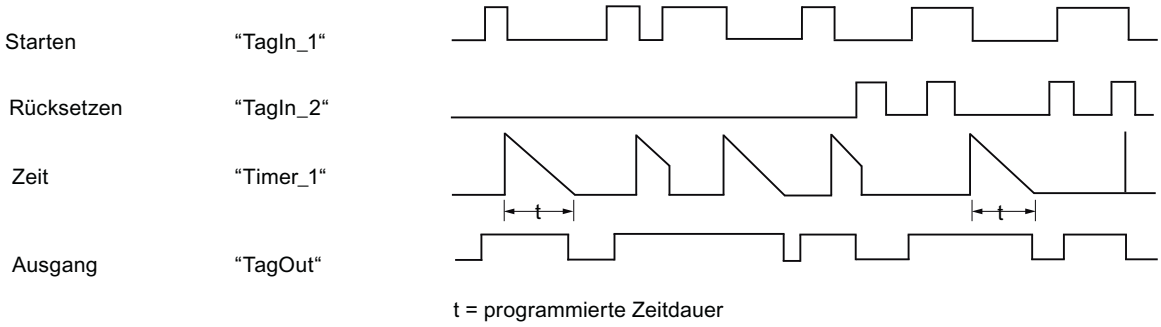

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf "0" zurückgesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTU_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CU	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Zähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

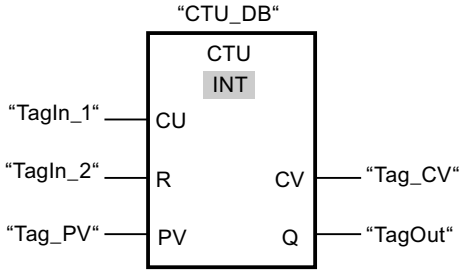

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CV" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des angegebenen Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PV" ist. In allen anderen Fällen liefert der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich "0" ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang LD von "0" auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und

die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CD	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

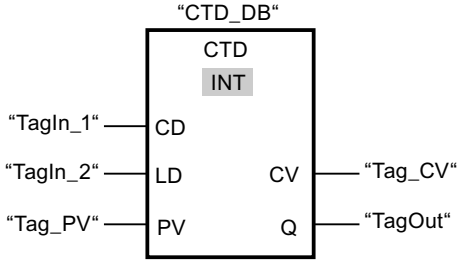

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert am Ausgang "Tag_CV" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich "0" ist. In allen anderen Fällen liefert der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt und in einem Flankenmerker gespeichert. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf "0" gesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Ausgang QD können Sie den Status des Rückwertszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Für CPU S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler< (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
CU	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, T, Z, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Status Vorwärtszähler
QD	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Status Rückwärtszähler
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

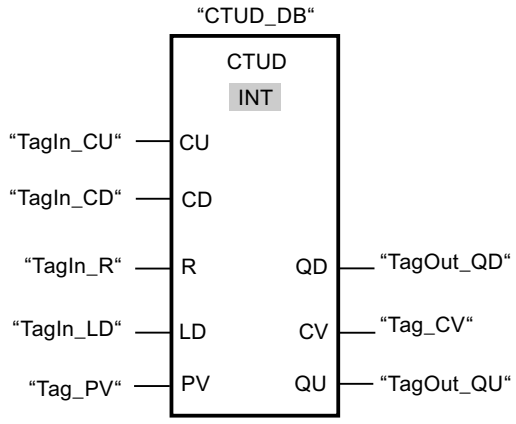

Wenn der Signalzustand am Eingang "TagIn_CU" oder am Eingang "TagIn_CD" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_CU" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn_CD" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert (INT = 32767) erreicht hat. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert (INT = -32768) erreicht hat.

Der Ausgang "TagOut_GU" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PV" ist. In allen anderen Fällen liefert der Ausgang "TagOut_QU" den Signalzustand "0".

Der Ausgang "TagOut_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich "0" ist. In allen anderen Fällen führt der Ausgang "TagOut_QD" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

Legacy

Z_VORW: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

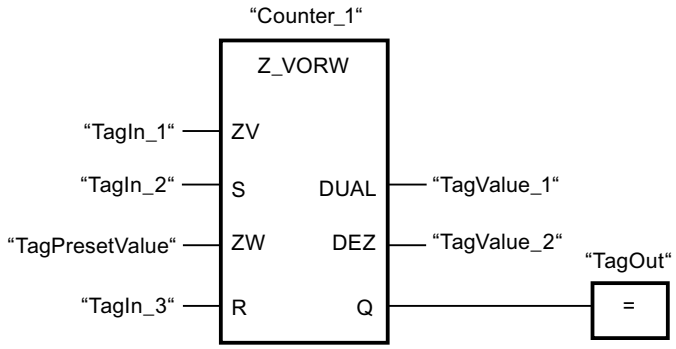

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

Z_RUECK: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZR	Input	BOOL	E, A, M, D, L oder Konstante	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

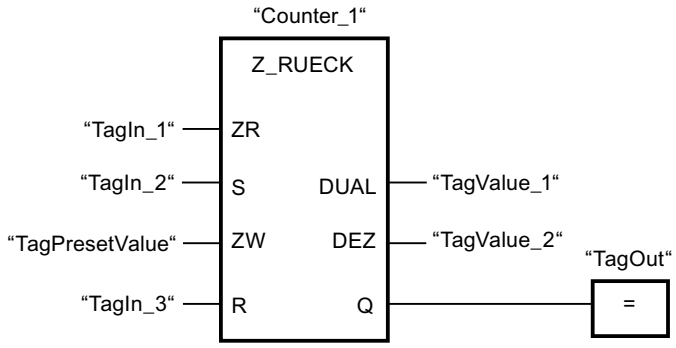

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen

ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Vorwärtszähleingang
ZR	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

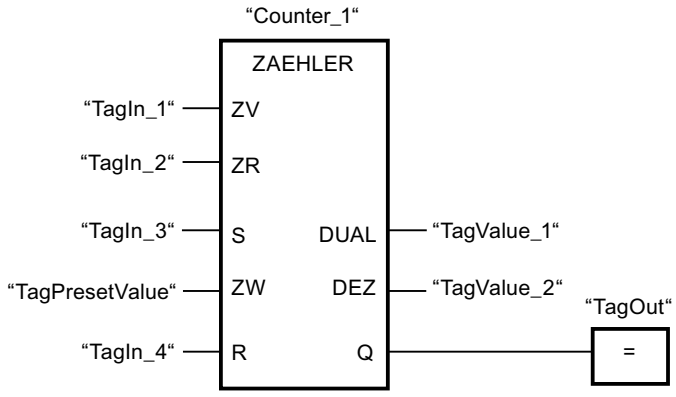

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SZ: Zähleranfangswert setzen

Beschreibung

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Starteingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
ZW	Input	WORD	E, A, M, D, L oder Konstante	Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999)
<Zähler>	InOut/Input	COUNTER	Z	Zähler, der voreingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

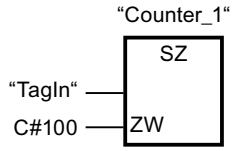

Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

ZV: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

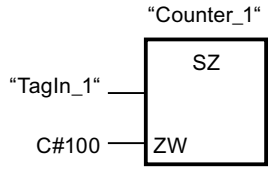

Netzwerk 2

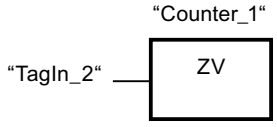

Netzwerk 3

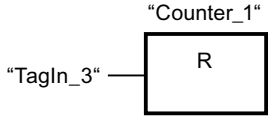

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

ZR: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

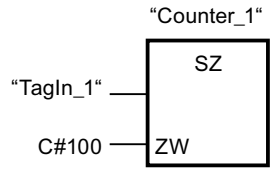

Netzwerk 2

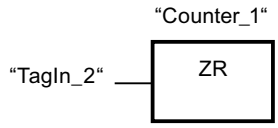

Netzwerk 3

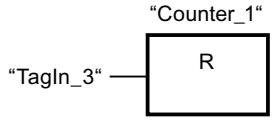

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

Vergleicher

CMP ==: Gleich

Beschreibung

Mit der Anweisung "Gleich" fragen Sie ab, ob der Wert am Eingang IN1 gleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite der Operanden gleich sein.

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann empfehlen wir Ihnen, statt der Anweisung "CMP ==: Gleich" die Anweisung "IN_RANGE: Wert innerhalb Bereich" zu verwenden.

Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefert die Anweisung "CMP ==: Gleich" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

IN1	IN2	VKE der Anweisung
'AA'	'AA'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'aa'	'aaa'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP ==: Gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP ==: Gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP ==: Gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP ==: Gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP ==: Gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefert die Anweisung "CMP ==: Gleich" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Vergleich von Hardware-Datentypen

Um Operanden vom Datentyp PORT vergleichen zu können, müssen Sie aus der Klappliste der Anweisungsbox den Datentyp WORD auswählen.

Wenn Sie die beiden Datentypen vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Wählen Sie für den Vergleich von Strukturen den Datentyp VARIANT aus der Klappliste der Anweisungsbox aus. Variablen aus folgenden Datentypen sind möglich:

- PLC-Datentyp (UDT)
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Instanzdatenbausteine und Variablen von anonymen Strukturen sind nicht zulässig.)
- Datenbausteine, die von einem PLC-Datentyp abgeleitet sind.
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY bei der Auswahl VARIANT miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Hinweis

ARRAY of BOOL

Wenn Sie zwei Operanden vom Datentyp ARRAY of BOOL miteinander vergleichen, aus der Klappliste den Datentyp VARIANT ausgewählt haben und die Anzahl der Elemente nicht durch 8 teilbar ist, dann werden auch die Füll-Bits verglichen. Das kann das Vergleichsergebnis beeinflussen.

Wenn ein Element innerhalb der Strukturen, die Sie vergleichen möchten, ein ungültiger STRING/WSTRING, eine ungültige Zeit- oder Datumsangabe oder eine ungültige Gleitpunktzahl ist, dann liefert das Vergleichsergebnis im VKE (Verknüpfungsergebnis) den Signalzustand "0".

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen:

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert
Wie oben beschrieben, stehen Ihnen die Datentypen ARRAY, STRUCT (innerhalb eines PLC-Datentyps), VARIANT, ANY und PLC-Datentyp (UDT) erst ab den Firmware Versionen 2.0 bzw. 4.2 zur Verfügung.				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

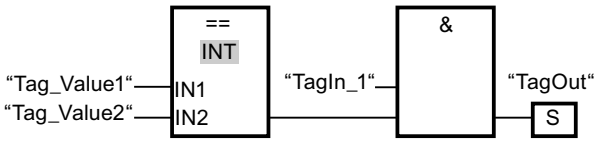

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" = "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

CMP <>: Ungleich

Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der Wert am Eingang IN1 ungleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite der Operanden gleich sein.

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann empfehlen wir Ihnen, statt der Anweisung "CMP <>: Ungleich" die Anweisung "OUT_RANGE: Wert außerhalb Bereich" zu verwenden.

Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefert die Anweisung "CMP <>: Ungleich" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

IN1	IN2	VKE der Anweisung
'AA'	'aa'	1
'Hallo Welt'	'HalloWelt'	1
'AA'	'AA'	0
'aa'	'aaa'	1

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <>: Ungleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <>: Ungleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP <>: Ungleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <>: Ungleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <>: Ungleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefert die Anweisung "CMP <>: Ungleich" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Vergleich von Hardware-Datentypen

Um Operanden vom Datentyp PORT mit der Anweisung "Ungleich" vergleichen zu können, müssen Sie aus der Klappliste der Anweisungsbox den Datentyp WORD auswählen.

Wenn Sie die beiden Hardware-Datentypen HW_IO und HW_DEVICE vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Wählen Sie für den Vergleich von Strukturen den Datentyp VARIANT aus der Klappliste der Anweisungsbox aus. Die folgenden Datentypen sind möglich:

- PLC-Datentyp
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp (UDT) enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Anonyme Strukturen sind nicht zulässig.)
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY bei der Auswahl VARIANT miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Hinweis

ARRAY of BOOL

Wenn Sie zwei Operanden vom Datentyp ARRAY of BOOL miteinander vergleichen, aus der Klappliste den Datentyp VARIANT ausgewählt haben und die Anzahl der Elemente nicht durch 8 teilbar ist, dann werden auch die Füll-Bits verglichen. Das kann das Vergleichsergebnis beeinflussen.

Wenn ein Element innerhalb der Strukturen, die Sie vergleichen möchten, ein ungültiger STRING/WSTRING, eine ungültige Zeit- oder Datumsangabe oder eine ungültige Gleitpunktzahl ist, dann liefert das Vergleichsergebnis im VKE (Verknüpfungsergebnis) den Signalzustand "0".

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen:

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert
Wie oben beschrieben, stehen Ihnen die Datentypen ARRAY, STRUCT (innerhalb eines PLC-Datentyps), VARIANT, ANY und PLC-Datentyp (UDT) erst ab den Firmware Versionen 2.0 bzw. 4.2 zur Verfügung.				

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

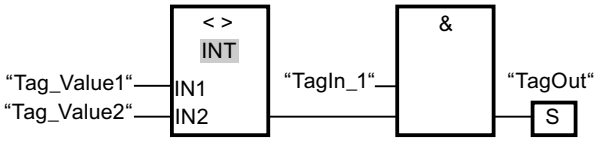

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <> "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

CMP >=: Größer gleich

Beschreibung

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der Wert am Eingang IN1 größer oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

IN1	IN2	VKE der Anweisung
'BB'	'AA'	1
'AAA'	'AA'	1
'Hallo Welt'	'Hallo Welt'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'AAA'	'a'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >=: Größer gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >=: Größer gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP >=: Größer gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >=: Größer gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP >=: Größer gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 größer (jünger) oder gleich dem Zeitpunkt am Eingang IN2 ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

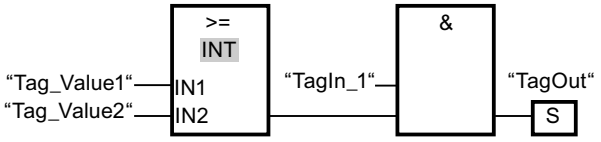

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" >= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

CMP <=: Kleiner gleich

Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der Wert am Eingang IN1 kleiner oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

IN1	IN2	VKE der Anweisung
'AA'	'aa'	1
'AAA'	'a'	1
'Hallo Welt'	'Hallo Welt'	1
'HalloWelt'	'Hallo Welt'	0
'BB'	'AA'	0
'AAA'	'AA'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <=: Kleiner gleich" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <=: Kleiner gleich" inaktiv.

Ergebnis:

Für die Anweisung "CMP <=: Kleiner gleich" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <=: Kleiner gleich" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <=: Kleiner gleich" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 kleiner (älter) oder gleich dem Zeitpunkt am Eingang IN2 ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

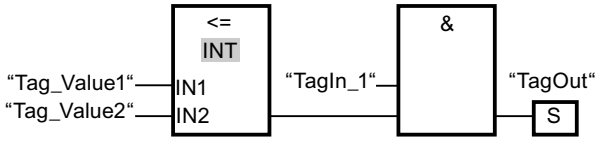

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

CMP >: Größer

Beschreibung

Mit der Anweisung "Größer" fragen Sie ab, ob der Wert am Eingang IN1 größer als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

IN1	IN2	VKE der Anweisung
'BB'	'AA'	1
'AAA'	'AA'	1
'AA'	'aa'	0
'AAA'	'a'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen

angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >: Größer" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >: Größer" inaktiv.

Ergebnis:

Für die Anweisung "CMP >: Größer" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >: Größer" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP >: Größer" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 größer (jünger) als der Zeitpunkt am Eingang IN2 ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

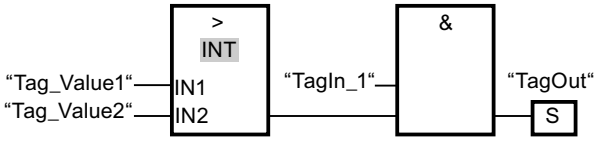

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" > "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

CMP <: Kleiner

Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der Wert am Eingang IN1 kleiner als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'AAA'	'a'	1
'BB'	'AA'	0
'AAA'	'AA'	0

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP <: Kleiner" (mit Datentyp STRING, WSTRING oder VARIANT) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP <: Kleiner" inaktiv.

Ergebnis:

Für die Anweisung "CMP <: Kleiner" (Datentyp STRING, WSTRING oder VARIANT) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP <: Kleiner" (Datentyp STRING, WSTRING oder VARIANT) angezeigt. Die Anweisung "CMP <: Kleiner" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Vergleich von Zeiten, Datum und Uhrzeit

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 kleiner (älter) als der Zeitpunkt am Eingang IN2 ist.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren

Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

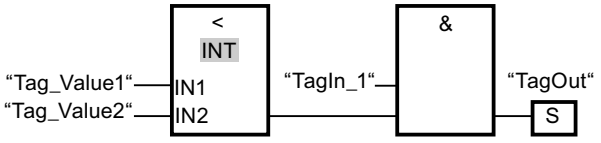

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" < "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

IN_RANGE: Wert innerhalb Bereich

Beschreibung

Mit der Anweisung "Wert innerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL innerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert innerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich MIN <= VAL oder VAL <= MAX erfüllt, liefert der Boxausgang den Signalzustand "1". Bei nicht erfülltem Vergleich steht am Boxausgang der Signalzustand "0".

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert innerhalb Bereich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VAL	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Vergleichswert
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
Boxausgang	Output	BOOL	E, A, M, D, L	Ergebnis des Vergleichs

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

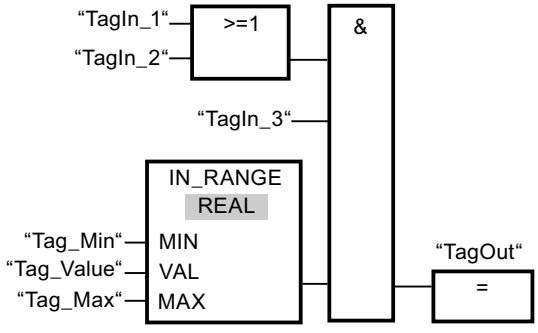

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" oder "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "1".
- Der Wert des Operanden "Tag_Value" liegt innerhalb des Wertebereichs, der durch die aktuellen Werte der Operanden "Tag_Min" und "Tag_Max" vorgegeben ist (MIN <= VAL oder VAL <= MAX).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

OUT_RANGE: Wert außerhalb Bereich

Beschreibung

Mit der Anweisung "Wert außerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL außerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert außerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich MIN > VAL oder VAL > MAX erfüllt, liefert der Boxausgang den Signalzustand "1". Der Boxausgang liefert auch den Signalzustand "1", wenn ein angegebener Operand vom Datentyp REAL einen ungültigen Wert aufweist.

Der Boxausgang liefert den Signalzustand "0", wenn der Wert am Eingang VAL die Bedingung MIN > VAL oder VAL > MAX nicht erfüllt.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert außerhalb Bereich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VAL	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Vergleichswert
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
Boxausgang	Output	BOOL	E, A, M, D, L	Ergebnis des Vergleichs

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

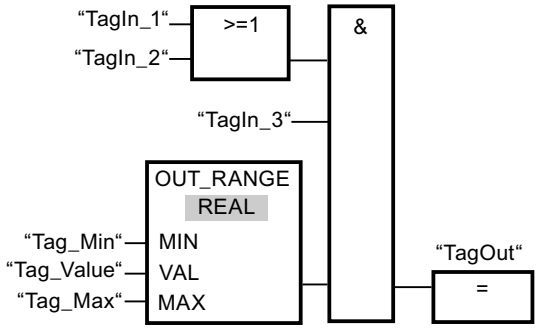

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "1".
- Der Wert des Operanden "Tag_Value" liegt außerhalb des Wertebereichs, der durch die Werte der Operanden "Tag_Min" und "Tag_Max" vorgegeben ist (MIN > VAL oder VAL > MAX).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

OK: Gültigkeit prüfen

Beschreibung

Mit der Anweisung "Gültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine gültige Gleitpunktzahl ist. Die Prüfung wird in jedem Programmzyklus durchgeführt. Wenn der Operandenwert zum Zeitpunkt der Abfrage eine gültige Gleitpunktzahl ist, liefert der Boxausgang den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "Gültigkeit prüfen" "0".

Die Anweisung "Gültigkeit prüfen" können Sie in Verbindung mit dem EN-Mechanismus nutzen. Wenn Sie die Anweisungsbox an einen Freigabeeingang EN anschließen, wird der Freigabeeingang nur bei einer positiven Abfrage der Wertgültigkeit gesetzt. Durch diese Funktionalität können Sie sicherstellen, dass eine Anweisung nur dann aktiviert wird, wenn der Wert des angegebenen Operanden eine gültige Gleitpunktzahl ist.

Parameter

Das folgende Beispiel zeigt die Funktionsweise der Anweisung "Gültigkeit prüfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L	Wert, der geprüft wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

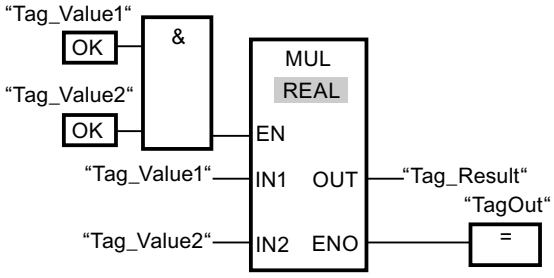

Wenn die Werte der Operanden "Tag_Value1" und "Tag_Value2" gültige Gleitpunktzahlen aufweisen, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Multiplikationsergebnis wird dann in den Operanden "Tag_Result" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

NOT_OK: Ungültigkeit prüfen

Beschreibung

Mit der Anweisung "Ungültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine ungültige Gleitpunktzahl ist. Die Prüfung wird in jedem Programmzyklus durchgeführt. Wenn der Operandenwert zum Zeitpunkt der Abfrage eine ungültige Gleitpunktzahl ist, liefert der Boxausgang den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Boxausgang "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungültigkeit prüfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L	Wert, der geprüft wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

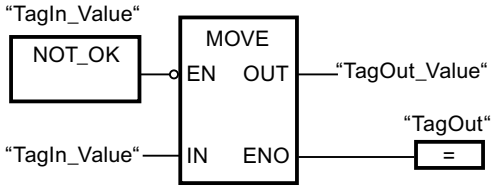

Wenn der Wert des Operanden "TagIn_Value" eine ungültige Gleitpunktzahl ist, wird die Anweisung "Wert kopieren" nicht ausgeführt. Der Operand "TagOut" am Freigabeausgang ENO wird auf den Signalzustand "0" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

VARIANT

EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP ==" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP ==: Gleich (Seite 1074)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

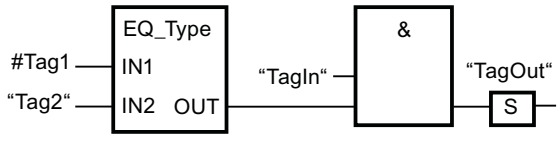

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag1 ist gleich "Tag2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie den Datentyp ab, welchen eine Variable nicht hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP <>" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP <>: Ungleich (Seite 1079)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

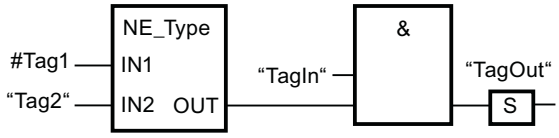

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag1 ist ungleich "Tag2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

EQ_ElemType: Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP ==" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP ==: Gleich (Seite 1074)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

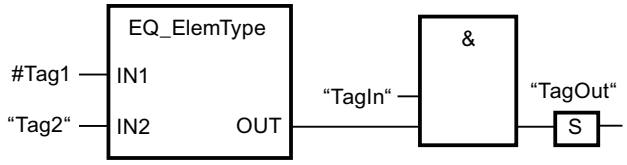

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag1 ist gleich "Tag2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

NE_ElemType: Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen" fragen Sie ab, welchen Datentyp eine Variable nicht hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CMP <>" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CMP <>: Ungleich (Seite 1079)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Baustein-schnittstelle möglich.)		Erster Operand
IN2	Input	Bitfolgen, Ganz-zahlen, Gleitpunkt-zahlen, Zeiten, Da-tum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Da-tentypen	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergeb-nis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

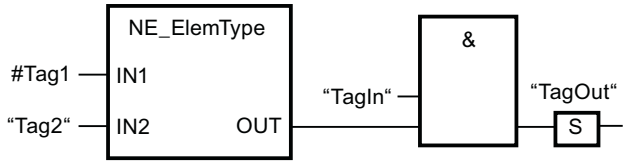

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag1 ist ungleich "Tag2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

IS_NULL: Auf GLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf GLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz auf einen NULL-Zeiger verweist und damit nicht auf ein Objekt zeigt.

Der <Operand> muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "0", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf GLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf GLEICH NULL verglichen wird
OUT	Output	BOOL	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

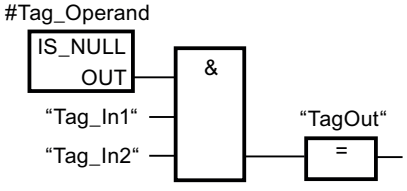

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "Tag_In1" und "Tag_In2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand zeigt auf kein Objekt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

NOT_NULL: Auf UNGLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz nicht auf einen NULL-Zeiger verweist und damit auf ein Objekt zeigt.

Der <Operand> muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "1", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf UNGLEICH NULL verglichen wird
OUT	Output	BOOL	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

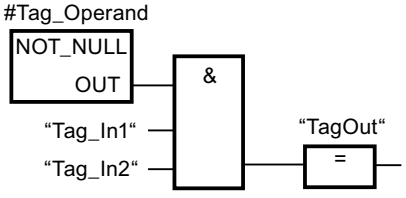

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "Tag_In1" und "Tag_In2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_Operand zeigt auf ein Objekt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

IS_ARRAY: Auf ARRAY abfragen

Beschreibung

Mit der Anweisung "Auf ARRAY abfragen" fragen Sie ab, ob der VARIANT auf eine Variable vom Datentyp ARRAY zeigt.

Der <Operand> muss den Datentyp VARIANT haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf ARRAY abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf ARRAY abgefragt wird
OUT	Output	BOOL	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

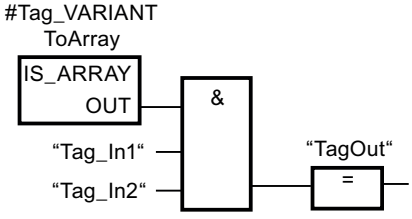

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "Tag_In1" und "Tag_In2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Operand #Tag_VARIANTToArray ist vom Datentyp ARRAY.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

EQ_TypeOfDB: Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable am Parameter IN1 adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit einem Datentyp am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp DB_ANY haben. Die Variable am Parameter IN2 kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
IN2	Input	TYPE_ID	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

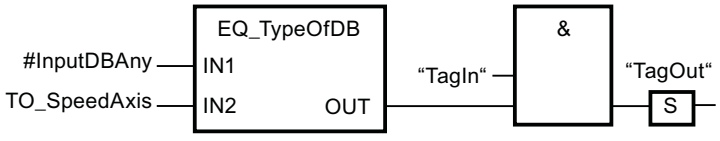

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Datentyp des Operanden #InputDBAny adressierten DB ist gleich dem Datentyp TO_SpeedAxis.

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu FUP (Seite 7493)

NE_TypeOfDB: Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein nicht hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable am Parameter IN1 adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit dem Datentyp am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp DB_ANY haben. Die Variable am Parameter IN2 kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Erster Operand
IN2	Input	TYPE_ID	E, A, M, D, L	E, A, M, D, L, P	Zweiter Operand
OUT	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

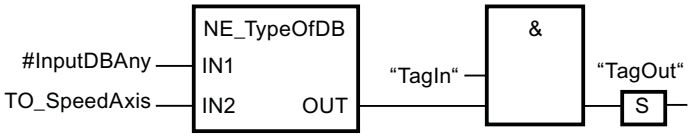

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, d. h. der Datentyp des Operanden #InputDBAny adressierten DB ist ungleich dem Datentyp TO_SpeedAxis.

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu FUP (Seite 7493)

Mathematische Funktionen

CALCULATE: Berechnen

Beschreibung

Mit der Anweisung "Berechnen" definieren und führen Sie, abhängig vom gewählten Datentyp, einen Ausdruck zum Berechnen mathematischer Operationen oder komplexer Verknüpfungen aus.

Den Datentyp der Anweisung wählen Sie aus der Klappliste "???" der Anweisungsbox. Abhängig vom ausgewählten Datentyp können Sie die Funktionalität bestimmter Anweisungen kombinieren, um eine komplexe Berechnung auszuführen. Die Angabe des zu berechnenden Ausdrucks erfolgt über einen Dialog, den Sie über das Symbol "Taschenrechner" am rechten oberen Rand der Anweisungsbox öffnen können. Im Ausdruck können die Namen der Eingangsparameter und die Syntax der Anweisungen enthalten sein. Die Angabe von Operandennamen oder -adressen ist nicht zulässig.

Die Anweisungsbox enthält im Grundzustand mindestens 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert.

Die Werte der Eingänge werden zur Ausführung des angegebenen Ausdrucks verwendet. Im Ausdruck müssen dabei nicht alle definierten Eingänge verwendet werden. Das Ergebnis der Anweisung wird an den Boxausgang OUT übertragen.

Hinweis

Wenn eine der mathematischen Operationen im Ausdruck fehlschlägt, dann wird kein Ergebnis an den Ausgang OUT übertragen und der Freigabeausgang ENO liefert den Signalzustand "1".

Wenn Sie im Ausdruck Eingänge verwenden, die in der Box nicht verfügbar sind, werden diese automatisch eingefügt. Voraussetzung dafür ist, dass die Nummerierung der neu zu definierenden Eingänge im Ausdruck lückenlos ist. Sie können zum Beispiel den Eingang IN4 im Ausdruck nicht verwenden, wenn der Eingang IN3 nicht definiert ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis oder ein Zwischenergebnis der Anweisung "Berechnen" liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.
- Bei der Ausführung einer der im Ausdruck angegebenen Anweisungen ist ein Fehler aufgetreten.

Die folgende Tabelle zeigt die Anweisungen, die, abhängig vom gewählten Datentyp, im Ausdruck der Anweisung "Berechnen" kombiniert werden, ausgeführt werden können:

Datentyp	Anweisung	Syntax	Beispiel
Bitfolgen	AND: UND verknüpfen	AND	IN1 AND IN2 OR IN3
	OR: ODER verknüpfen	OR	
	XOR: EXCLUSIV ODER verknüpfen	XOR	
	INV: Einerkomplement erzeugen	NOT	
	SWAP: Anordnung ändern [1)]	SWAP	
Ganzzahlen	ADD: Addieren	+	(IN1 + IN2) * IN3; (ABS(IN2)) * (ABS(IN1))
	SUB: Subtrahieren	-	
	MUL: Multiplizieren	*	
	DIV: Dividieren	/	
	MOD: Divisionsrest gewinnen	MOD	
	INV: Einerkomplement erzeugen	NOT	
	NEG: Zweierkomplement erzeugen	-(in1)	
	ABS: Absolutwert bilden	ABS( )	

Datentyp	Anweisung	Syntax	Beispiel
Gleitpunktzahlen	ADD: Addieren	+	((SIN(IN2) * SIN(IN2) + (SIN(IN3) * SIN(IN3)) / IN3)); (SQR(SIN(IN2)) + (SQR(COS(IN3)) / IN2))
	SUB: Subtrahieren	-	
	MUL: Multiplizieren	*	
	DIV: Dividieren	/	
	EXPT: Potenzieren	**	
	ABS: Absolutwert bilden	ABS( )	
	SQR: Quadrat bilden	SQR( )	
	SQRT: Quadratwurzel bilden	SQRT( )	
	LN: Natürlichen Logarithmus bilden	LN( )	
	EXP: Exponentialwert bilden	EXP( )	
	FRAC: Nachkommastellen ermitteln	FRAC( )	
	SIN: Sinuswert bilden	SIN( )	
	COS: Cosinuswert bilden	COS( )	
	TAN: Tangenswert bilden	TAN( )	
	ASIN: Arcussinuswert bilden	ASIN( )	
	ACOS: Arcuscosinuswert bilden	ACOS( )	
	ATAN: Arcustangenswert bilden	ATAN( )	
	NEG: Zweierkomplement erzeugen	-(in1)	
	TRUNC: Ganzzahl erzeugen	TRUNC( )	
	ROUND: Zahl runden	ROUND( )	
	CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen	CEIL( )	
	FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen	FLOOR( )	
[1] Nicht für den Datentyp BYTE möglich.			

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Berechnen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster verfügbarer Eingang
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter verfügbarer Eingang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
INn	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Ausgang, an den das Endergebnis übertragen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

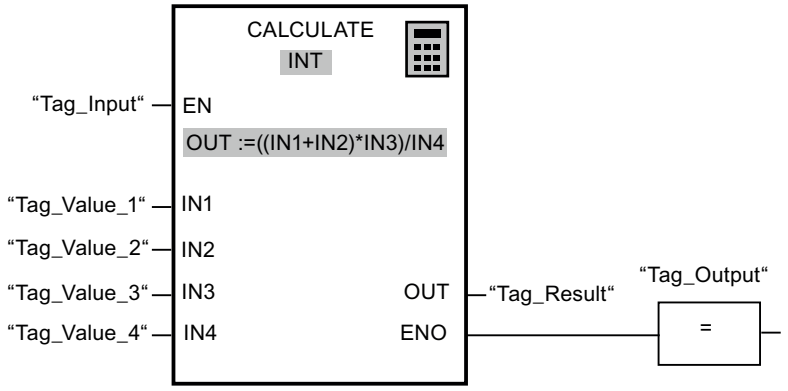

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value_1	4
IN2	Tag_Value_2	4
IN3	Tag_Value_3	3
IN4	Tag_Value_4	2
OUT	Tag_Result	12

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value_1" wird mit dem Wert des Operanden "Tag_Value_2" addiert. Die Summe wird mit dem Wert des Operanden "Tag_Value_3" multipliziert. Das Produkt wird durch den Wert des Operanden "Tag_Value_4" dividiert. Der Quotient wird als Endergebnis in den Operanden "Tag_Result" am Ausgang OUT der Anweisung übertragen. Wenn keine Fehler während der Bearbeitung der einzelnen Anweisungen auftreten, werden der Freigabeausgang ENO und der Operand "Tag_Output" auf "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Beispiel für das Berechnen einer Gleichung (Seite 7551)

Grundlagen zu FUP (Seite 7493)

ADD: Addieren

Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen die Summe am Ausgang OUT ab (OUT := IN1+IN2).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter addiert. Die Summe wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Summand
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Summand
INn	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte, die addiert werden.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Summe

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

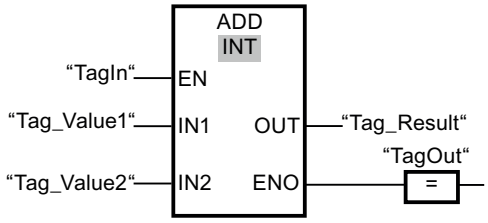

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SUB: Subtrahieren

Beschreibung

Mit der Anweisung "Subtrahieren" subtrahieren Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 und fragen die Differenz am Ausgang OUT ab (OUT := IN1-IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Differenz

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

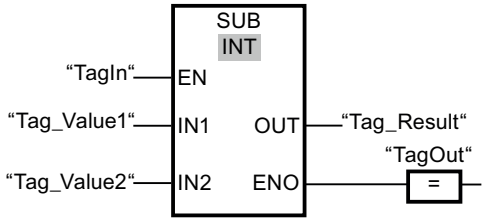

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value2" wird vom Wert des Operanden "Tag_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MUL: Multiplizieren

Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen das Produkt am Ausgang OUT ab (OUT := IN1*IN2).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter multipliziert. Das Produkt wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Wert der Multiplikation
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Wert der Multiplikation
INn	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte, die multipliziert werden.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Produkt

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

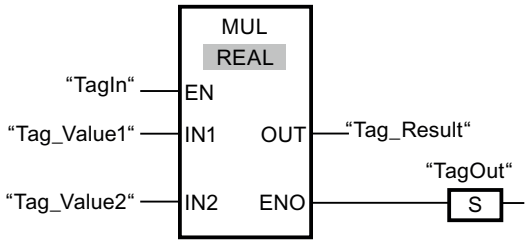

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Ergebnis der Multiplikation wird im Operanden "Tag_Result" abgelegt. Wenn

keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

DIV: Dividieren

Beschreibung

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab (OUT := IN1/IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

Division durch den Wert "0"

Die Division eines Dividend (IN1) durch einen Divisor (IN2) mit dem Wert "0" setzt den Freigabeausgang (ENO) in den Signalzustand "TRUE".

Der Wert des Quotienten (OUT) wird bei der Division durch "0" wie folgt durch den Datentyp beeinflusst:

- Der Datentyp ist INT oder LREAL: Der Wert des Quotienten (OUT) ist "0".
- Der Datentyp ist REAL: Der Wert des Quotienten (OUT) ist der Maximalwert (2143289344).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Wert des Quotienten

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

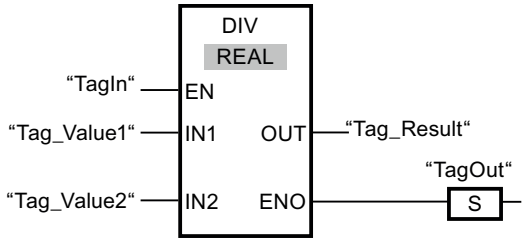

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Divisionsrest

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

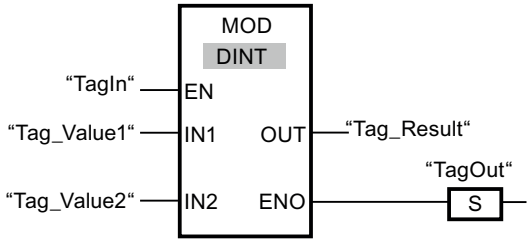

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Zweierkomplement des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

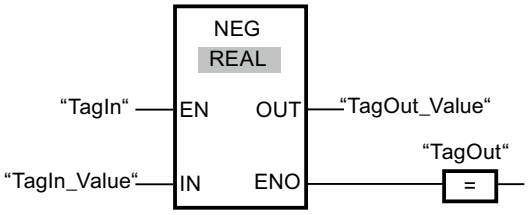

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut_Value" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

INC: Inkrementieren

Beschreibung

Mit der Anweisung "Inkrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst höheren Wert und fragen das Ergebnis ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN/OUT	InOut	Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Wert, der inkrementiert wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

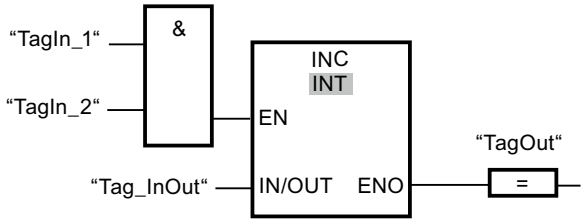

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag_InOut" um eins erhöht und der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

DEC: Dekrementieren

Beschreibung

Mit der Anweisung "Dekrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst niedrigeren Wert und fragen das Ergebnis ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN/OUT	InOut	Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Wert, der dekrementiert wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

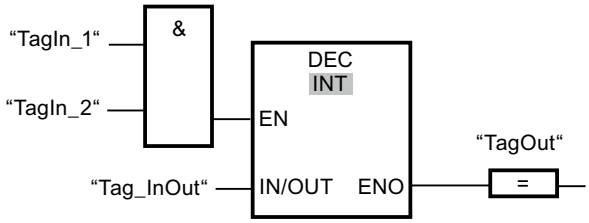

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag_InOut" um eins verringert und der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Absolutwert des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

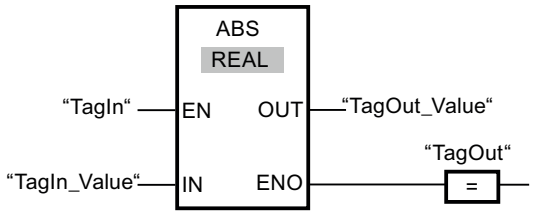

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn_Value" und gibt das Ergebnis am Ausgang "TagOut_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den kleinsten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden an der Box aufsteigend nummeriert.

Im Grundzustand enthält die Anweisung min. zwei Eingänge (IN1 und IN2) bis max. 100 Eingänge.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
OUT	Output	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)					

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

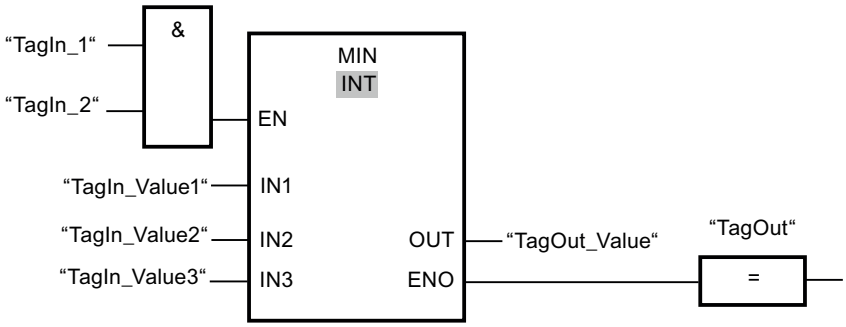

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den größten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden an der Box aufsteigend nummeriert.

Im Grundzustand enthält die Anweisung min. zwei Eingänge (IN1 und IN2) bis max. 100 Eingänge.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden.
OUT	Output	Ganzzahlen, Gleitpunktzahlen, DTL, DT	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)					

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

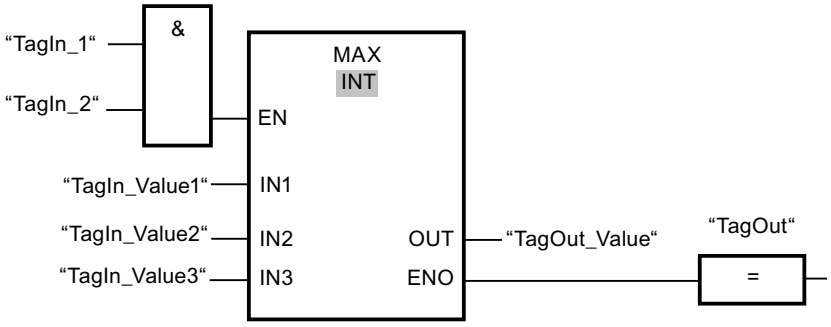

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis der am Parmeter IN angegebene Wert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Obergrenze

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL, DT	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DTL, DT	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.						

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

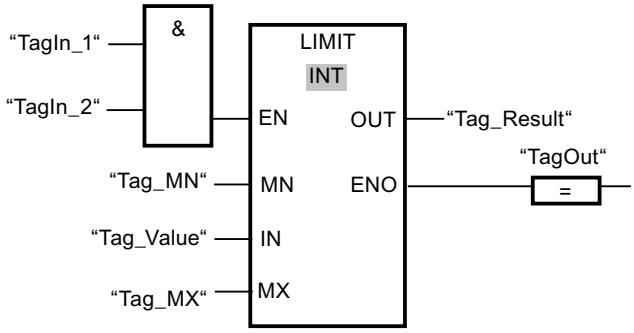

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_MN" und "Tag_MX" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_MN" in den Ausgang "Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert einer Gleitpunktzahl und schreiben das Ergebnis in den Ausgang OUT.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Quadrat des Eingangswertes

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

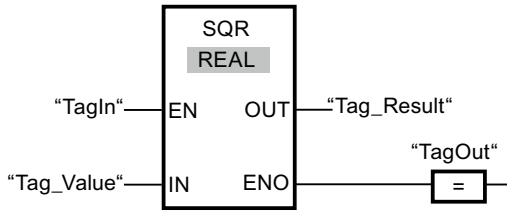

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert einer Gleitpunktzahl die Quadratwurzel und schreiben das Ergebnis in den Ausgang OUT. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L	E, A, M, D, L	Quadratwurzel des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

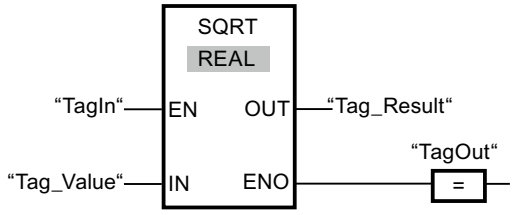

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis e (e = 2,718282). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

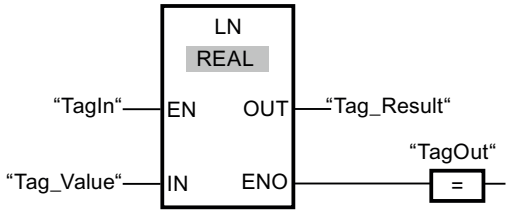

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag_Value" und legt

das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden (OUT = e^{IN}).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Exponentialwert des Eingangswerts IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

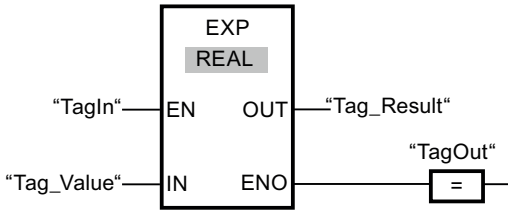

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Sinus des angegebenen Winkels

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

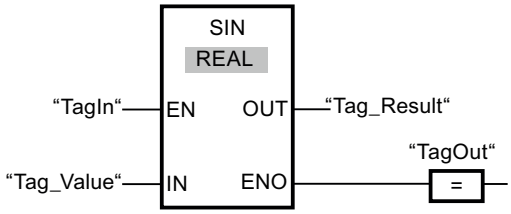

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	1.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

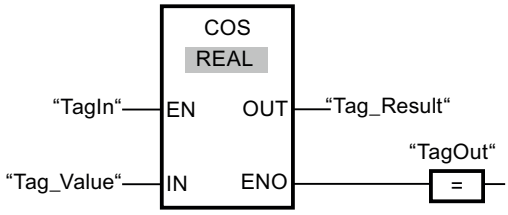

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Tangens des angegebenen Winkels

Aus der Klappliste "<???>" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

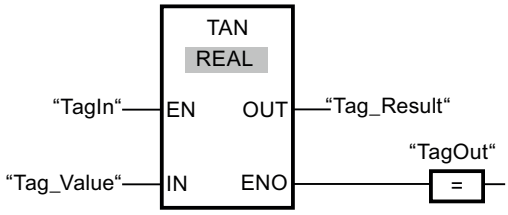

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Tangenswert bilden" ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Sinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

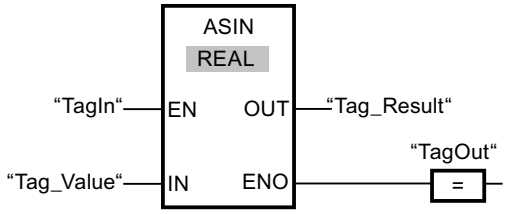

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Cosinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

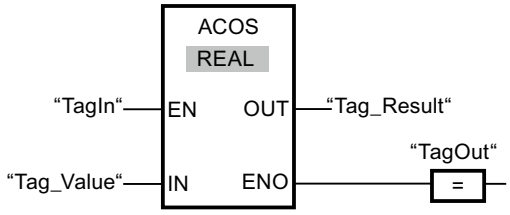

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Tangenswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

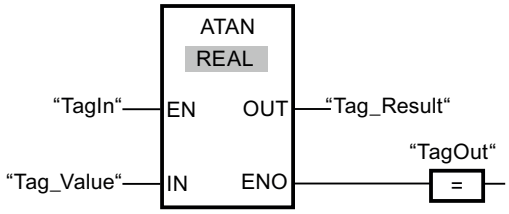

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+0.785398 (π/4)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu FUP (Seite 7493)

FRAC: Nachkommastellen ermitteln

Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" ermitteln Sie die Nachkommastellen des Wertes am Eingang IN. Das Ergebnis der Abfrage wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden. Wenn am Eingang IN z. B. der Wert 123,4567 ansteht, liefert der Ausgang OUT den Wert 0,4567.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. am Eingang liegt keine gültige Gleitpunktzahl an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Nachkommastellen ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert, dessen Nachkommastellen ermittelt werden.
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Nachkommastellen des Eingangswerts am Eingang IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

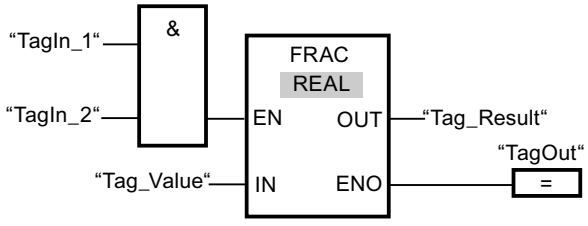

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	2.555
OUT	Tag_Result	0.555

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung gestartet. Die Nachkommastellen aus dem Wert des Operanden "Tag_Value" werden in den Operanden "Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

EXPT: Potenzieren

Beschreibung

Mit der Anweisung "Potenzieren" potenzieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2. Das Ergebnis der Anweisung wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden (OUT = $IN1^{IN2}$).

Der Eingang IN1 kann nur mit gültigen Gleitpunktzahlen belegt werden. Zur Belegung des Eingangs IN2 sind auch Ganzzahlen zulässig.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Potenzieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN1	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Basiswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, mit dem der Basiswert potenziert wird
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

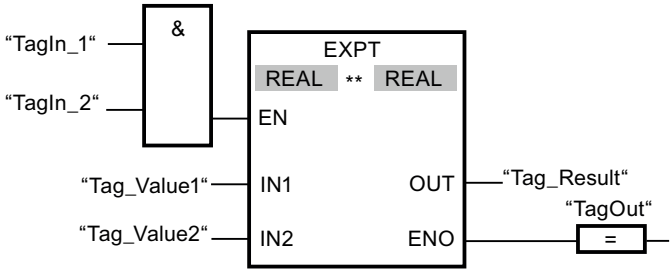

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung "Potenzieren" gestartet. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" potenziert. Das Ergebnis wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Verschieben

MOVE: Wert kopieren

Beschreibung

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Datentyp am Parameter IN kann nicht in den angegebenen Datentyp am Parameter OUT1 konvertiert werden.

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1200:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE ,TOD, CHAR
WORD	WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
USINT	USINT, UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
INT	INT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UINT	UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UDINT	UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LREAL
TIME	TIME	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME
DATE	DATE	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE
TOD	TOD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD
DTL	DTL	DTL

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
CHAR	CHAR	BYTE, WORD, DWORD, CHAR, Zeichen einer Zeichenkette[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, CHAR, WCHAR, Zeichen einer Zeichenkette[1)]
Zeichen einer Zeichenkette[1)]	Zeichen einer Zeichenkette	CHAR, WCHAR, Zeichen einer Zeichenkette
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
PLC-Datentyp (UDT)	PLC-Datentyp (UDT)	PLC-Datentyp (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER	IEC_UDCOUNTER

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1500:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE ,TOD, LTOD, CHAR
WORD	WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
DWORD	DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
LWORD	LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
USINT	USINT, UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
INT	INT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
UINT	UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD. LTOD
DINT	DINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UDINT	UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
LINT	LINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
ULINT	ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LWORD, LREAL
S5TIME	S5TIME	WORD, S5TIME
TIME	TIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME
LTIME	LTIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME
DATE	DATE	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE
DT	DT	DT
LDT	LDT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT
TOD	TOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD
LTOD	LTOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, LWORD, CHAR, Zeichen einer Zeichenkette[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, LWORD, CHAR, WCHAR, Zeichen einer Zeichenkette[1)]
Zeichen einer Zeichenkette[1)]	Zeichen einer Zeichenkette	CHAR, WCHAR, Zeichen einer Zeichenkette
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
COUNTER	COUNTER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	TIMER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
PLC-Datentyp (UDT)	PLC-Datentyp (UDT)	PLC-Datentyp (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER	IEC_LTIMER

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
IEC_SCOUN-TER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUN-TER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUN-TER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUN-TER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUN-TER	IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUN-TER	IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUN-TER	IEC_ULCOUNTER	IEC_ULCOUNTER
REF()	Es findet keine Datentypkonvertierung statt. Die übergebene Variable muss genau dem Zieldatentyp entsprechen.	

1) Mit der Anweisung "Wert kopieren" können Sie auch einzelne Zeichen einer Zeichenkette zu Operanden vom Datentyp CHAR oder WCHAR übertragen. Die Nummer des zu übertragenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" übertragen. Die Übertragung vom Operanden des Datentyps CHAR oder WCHAR zu einzelnen Zeichen einer Zeichenkette ist ebenfalls möglich. Sie können auch ein bestimmtes Zeichen einer Zeichenkette durch das Zeichen einer anderen Zeichenkette ersetzen.

2) Das Übertragen von ganzen Feldern (ARRAY) ist nur dann möglich, wenn die Feldkomponenten der Operanden am Eingang IN und am Ausgang OUT1 vom gleichen Datentyp sind.

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen überschrieben.

Die Anweisungsbox enthält im Grundzustand 1 Ausgang (OUT1). Die Anzahl der Ausgänge ist erweiterbar. Die eingefügten Ausgänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung wird der Inhalt des Operanden am Eingang IN zu allen verfügbaren Ausgängen übertragen. Die Anweisungsbox kann nicht erweitert werden, wenn strukturierte Datentypen (DTL, STRUCT, ARRAY) oder Zeichen einer Zeichenkette übertragen werden.

Zum Kopieren von Operanden des Datentyps ARRAY können auch die Anweisungen "Bereich kopieren" (MOVE_BLK) und "Bereich ununterbrechbar kopieren" (UMOVE_BLK) verwendet werden. Operanden des Datentyps STRING oder WSTRING können mit der Anweisung "Zeichenkette verschieben" (S_MOVE) kopiert werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT)	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT)	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Element, mit dem die Zieladresse überschrieben wird.
OUT1	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT)	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, CHAR, WCHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT)	E, A, M, D, L	E, A, M, D, L	Zieladresse

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

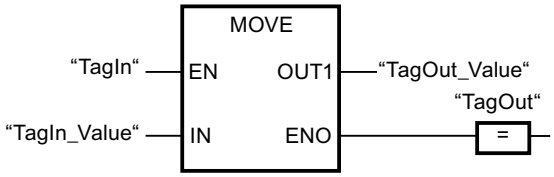

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Wenn der Operand "TagIn" den Signalzustand TRUE liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn_Value" zum Operanden "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden der Freigabeausgang ENO und "TagOut" auf den Signalzustand TRUE gesetzt.

Hinweis

Weitere Informationen zur Anweisung MOVE im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)

MOVE_BLK: Bereich kopieren (Seite 1164)

UMOVE_BLK: Bereich ununterbrechbar kopieren (Seite 1172)

S_MOVE: Zeichenkette verschieben (Seite 2215)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Deserialize: Deserialisieren

Beschreibung

Mit der Anweisung "Deserialisieren" wandeln Sie die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder um und füllen dessen kompletten Inhalt.

Der Speicherbereich, in dem die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> liegt, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Speicherbereiche zugelassen. Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Stellen Sie vor der Umwandlung sicher, dass genügend Speicherplatz vorhanden ist. Wenn der Speicherbereich mithilfe der Anweisung "Serialisieren" befüllt wurde und eventuell Füll-Bytes eingefügt wurden, dann werden diese bei der Umwandlung nicht berücksichtigt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Mithilfe der Anweisung können Sie mehrere sequenzielle Darstellungsformen von umgewandelten Daten schrittweise wieder in ihren ursprünglichen Zustand umwandeln.

Wenn Sie nur eine einzelne sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder umwandeln möchten, dann können Sie dazu auch direkt die Anweisung "TRCV: Daten über Kommunikationsverbindung empfangen" verwenden.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das BOOL eine Länge von 1 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Optimierter Speicherbereich

Um größere Strukturen zu deserialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich für die sequenzielle Darstellung auch mit optimiertem Zugriff deklarieren. Der Inhalt der sequenziellen Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich. Der Zugriff auf die Bytes des ARRAYs ist ausschließlich symbolisch möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deserialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC_ARRAY	Input	VARIANT ARRAY[*] of BYTE [1]	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der Datenstrom gespeichert ist.
DEST_VARIABLE	InOut	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die deserialisiert werden soll.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
[1] Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 möglich				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_ARRAY und DEST_VARIABLE überschneiden sich.
8136	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8150	Der Datentyp VARIANT am Parameter SRC_ARRAY enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_ARRAY
8153	Am Parameter SRC_ARRAY steht nicht genügend Speicherplatz zur Verfügung.
8154	Ungültiger Datentyp am Parameter SRC_ARRAY
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_VARIABLE
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8136	Der Zugriff am Parameter SRC_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle eines FBs oder FCs
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle eines FBs oder FCs
Error	INT	
Label	STRING[4]	

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Target	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

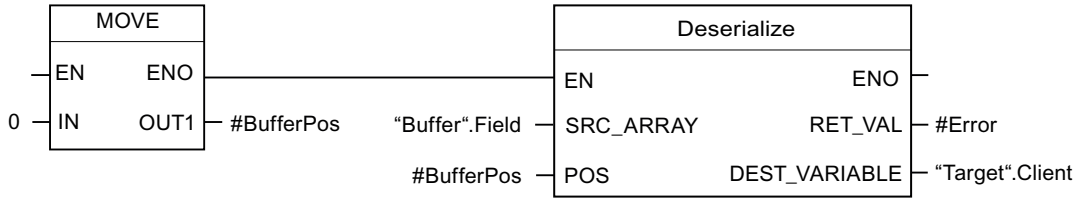

Die Anweisung "Wert kopieren" kopiert den Wert "0" in den Operanden #BufferPos. Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform der Kundendaten aus der Variable "Buffer" und schreibt sie in die Variable "Target". Der Operand #BufferPos speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen.

Netzwerk 2:

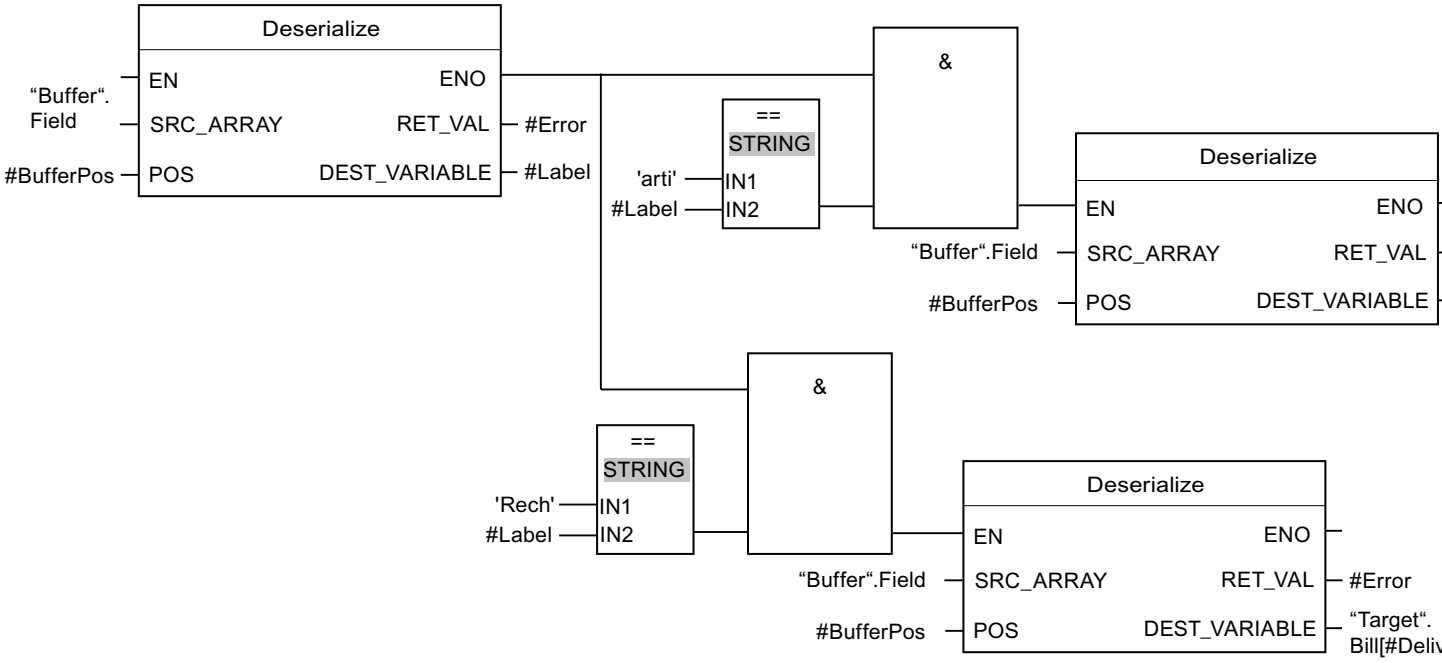

Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform des Trennblatts, das nach den Kundendaten in der sequenziellen Darstellungsform abgelegt wurde, aus der Variable "Buffer" und schreibt die Zeichen in den Operanden #Label. Die Zeichen werden mit Hilfe von Vergleichsanweisungen auf "arti" und "Bill" verglichen. Ist der Vergleich auf "arti" = TRUE, dann handelt es sich Artikeldaten, die deserialisiert und in die Variable "Target" geschrieben werden. Ist der Vergleich auf "Bill" = TRUE, dann handelt es sich um Rechnungsdaten, die deserialisiert und in die Variable "Target" geschrieben werden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu FUP (Seite 7493)

Serialize: Serialisieren

Beschreibung

Mit der Anweisung "Serialisieren" wandeln Sie mehrere PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> in eine sequenzielle Darstellungsform um, ohne dass Teile ihrer Struktur verloren gehen.

Mithilfe der Anweisung können Sie mehrere strukturierte Daten aus Ihrem Programm in einem Puffer, der vorzugsweise in einem globalen Datenbaustein liegt, zwischenspeichern und an eine andere CPU senden. Der Speicherbereich, in dem die umgewandelten Daten abgelegt werden, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Daten zugelassen. Fülldaten des Datenbereichs der Quelle sind im Ziel-ARRAY undefiniert. Diese können sowohl Füll-Bytes bzw. Füll-Bits eines Datenbereichs (z. B. ARRAY, STRUCT oder PLC-Datentyp (UDT)) sowie die aktuell ungenutzten Zeichen eines Strings sein.

Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Strukturen, die gemäß der Regeln für standard Speicherbereiche größer als 64 KB sind, können nicht serialisiert werden, wenn der Operand am Parameter DEST_ARRAY in einem standard Speicherbereich liegt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Der Operand am Parameter POS enthält die Information, wie viele Bytes die umgewandelten Daten belegen.

Wenn Sie einen einzelnen PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> versenden möchten, dann können Sie direkt die Anweisung "TSEND: Daten über Kommunikationsverbindung senden" aufrufen.

Hinweis

Vergleich von Strukturen

Um Strukturen zu vergleichen ist es nicht notwendig, diese zuerst zu serialisieren. Verwenden Sie stattdessen die Anweisungen "CMP".

Weitere Informationen finden Sie im Kapitel "Vergleicher".

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hängt die Länge eines BOOL davon ab, welcher Datentyp folgt. Das bedeutet, wenn z. B. ein BYTE folgt, hat das BOOL eine Länge von 1 Byte. Wenn z. B. ein WORD folgt, dann hat das BOOL eine Länge von 2 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Daher ist es empfehlenswert, dass der Quelldatenbereich für das Serialisieren mit den großen Datentypen beginnt und mit booleschen Elementen endet. Dadurch wird das Auffüllen mit Füll-Bits wesentllich reduziert.

Optimierter Speicherbereich

Um größere Strukturen zu serialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich auch mit optimiertem Zugriff deklarieren. Die sequenzielle Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Serialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC_VARI-ABLE	Input	VARIANT	D, Bausteinschnitt-stelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die serialisiert werden soll.
DEST_AR-RAY	InOut	VARIANT	D, Bausteinschnitt-stelle eines FB (Die Abschnitte In-put, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der erzeugte Datenstrom gespeichert wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Gesamtzahl der Bytes, die die umgewandelten Kundendaten belegt haben. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_VARIABLE und DEST_ARRAY überschneiden sich.
8150	Der Datentyp VARIANT am Parameter SRC_VARIABLE enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_VARIABLE
8236	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben
8251	Keine gültige Referenz am Parameter DEST_ARRAY
8253	Die Variable am Parameter DEST_ARRAY bietet für den Inhalt der Variable am Parameter SRC_VARIABLE nicht genügend Speicherplatz. Durch den Eingangswert der Variable am Parameter POS wird der zur Verfügung stehende Speicherplatz reduziert. Der Eingangswert des Parameters POS bestimmt, wo innerhalb der Variable am Parameter DEST_ARRAY begonnen wird.
8254	Ungültiger Datentyp am Parameter DEST_ARRAY
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehlercode* (W#16#...)	Erläuterung
8236	Der Zugriff am Parameter DEST_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

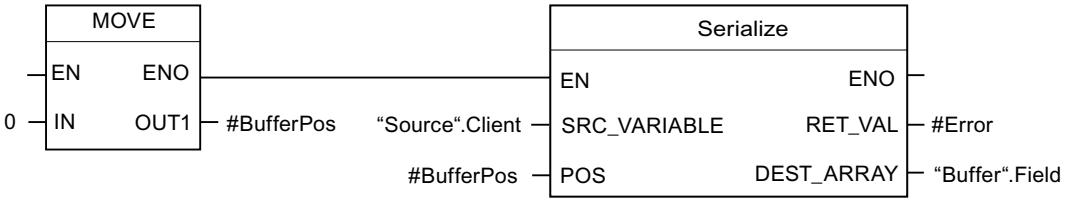

Die Anweisung "Wert kopieren" kopiert den Wert "0" in den Operanden #BufferPos. Die Anweisung "Serialisieren" serialisiert die Kundendaten aus der Variable "Source" und schreibt sie als sequenzielle Darstellung in die Variable "Buffer". Der Index des nächsten unbeschriebenen Bytes des Operanden "Buffer".Field wird im Operanden #BufferPos gespeichert.

Netzwerk 2:

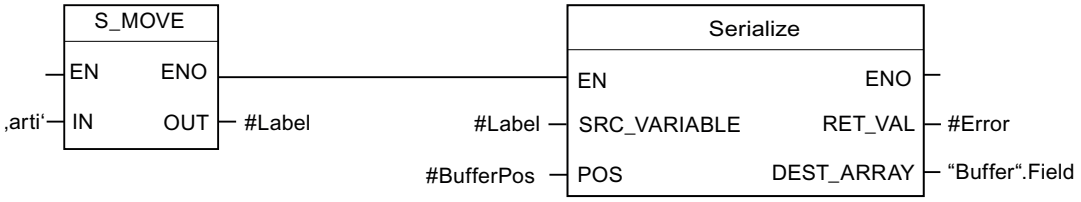

Um die sequenzielle Darstellungsform später wieder leichter deserialisieren zu können, wird jetzt eine Art Trennblatt eingefügt. Die Anweisung "Zeichenkette verschieben" kopiert die Zeichen "arti" in den Operanden #Label. Die Anweisung "Serialisieren" schreibt diese Zeichen nach den Kundendaten in die Variable "Buffer". Der Wert des Operanden #BufferPos wird entsprechend erhöht.

Netzwerk 3:

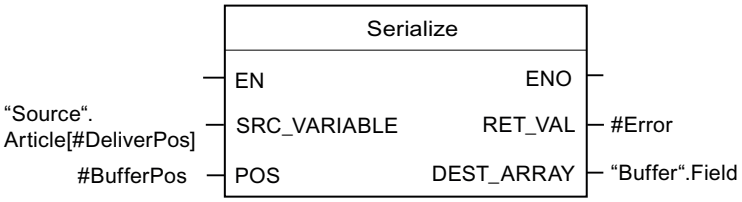

Die Anweisung "Serialisieren" serialisiert die Daten eines bestimmten Artikels, der zur Laufzeit berechnet wird, aus der Variable "Source" und schreibt sie als sequenzielle Darstellung nach den Zeichen "arti" in die Variable "Buffer".

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle

Operand	Datentyp	Deklaration
Error	INT	Im Abschnitt "Temp" der Bausteinschnittstelle
Label	STRING[4]	Im Abschnitt "Temp" der Bausteinschnittstelle

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Source	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
Buffer	Field	Array[0..294] of BYTE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu FUP (Seite 7493)

MOVE_BLK: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN [1)]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.						

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

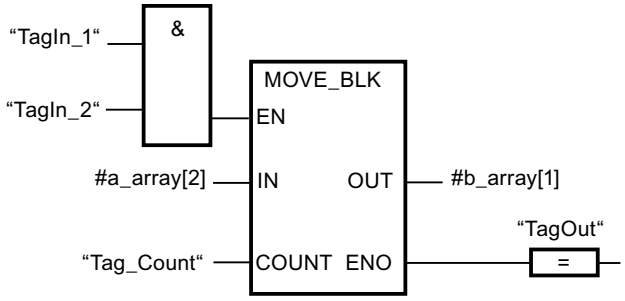

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Wenn keine Fehler während der Bearbeitung der

Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

MOVE_BLK_VARIANT: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können ein komplettes ARRAY oder Elemente eines ARRAYs in ein anderes ARRAY des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-ARRAY dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines ARRAYs umkopieren oder einzelne Elemente kopieren.

Die Anzahl der Elemente, die kopiert werden sollen, darf den selektierten Quellbereich bzw. Zielbereich nicht überschreiten.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss das ARRAY noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC_INDEX und DEST_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des ARRAYs.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als zur Verfügung gestellt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRC	Input [2)]	VARIANT (der auf ein ARRAY oder ein einzelnes ARRAY-Element zeigt), ARRAY of <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Der Quellbereich, aus dem kopiert wird
COUNT	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Elemente, die kopiert werden Geben Sie dem Parameter COUNT den Wert "1", wenn am Parameter SRC oder am Parameter DEST kein ARRAY angegeben ist.
SRC_INDEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert das erste Element, das kopiert wird: • Der Parameter SRC_INDEX wird nullbasiert berechnet. Wenn am Parameter SRC ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter SRC_INDEX das erste Element innerhalb des Quellbereichs an, aus dem kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter SRC kein ARRAY oder nur ein einzelnes Element eines ARRAYs angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST_IN-DEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert den Anfang des Zielspeicherbereichs: • Der Parameter DEST_INDEX wird nullbasiert berechnet. Wenn am Parameter DEST ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter DEST_INDEX das erste Element innerhalb des Zielbereichs an, in das kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter DEST kein ARRAY angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert "0".
DEST	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zielbereich, in den die Inhalte des Quellbereichs kopiert werden
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter DEST ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden. 2) Am Parameter SRC sind die Datentypen BOOL und ARRAY of BOOL nicht zulässig.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Die Datentypen stimmen nicht überein. Verwenden Sie statt eines ARRAY of Struct ein ARRAY of PLC-Datentyp(UDT).
8151	Der Zugriff auf den Parameter SRC ist nicht möglich.
8152	Der Operand am Parameter SRC ist nicht typisiert.
8153	Codeerzeugungsfehler am Parameter SRC
8154	Der Operand am Parameter SRC hat den Datentyp BOOL.
8281	Ungültiger Wert am Parameter COUNT
8382	Der Wert am Parameter SRC_INDEX ist kleiner NULL.
8383	Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8482	Der Wert am Parameter DEST_INDEX ist kleiner NULL.
8483	Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8534	Der Parameter DEST ist schreibgeschützt
8551	Der Zugriff auf den Parameter DEST ist nicht möglich.
8552	Der Operand am Parameter DEST ist nicht typisiert.
8553	Codeerzeugungsfehler am Parameter DEST
8554	Der Operand am Parameter DEST hat den Datentyp BOOL.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

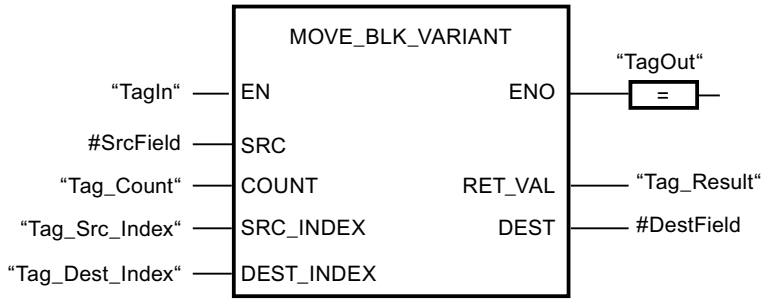

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
SRC	Input	#SrcField	Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[0..10] of "MOVE_UDT"
COUNT	Input	Tag_Count	2
SRC_INDEX	Input	Tag_Src_Index	3
DEST_INDEX	Input	Tag_Dest_Index	3
DEST	InOut	#DestField	Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[10..20] of "MOVE_UDT"

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. 2 Elemente werden aus dem Quellbereich, beginnend mit dem vierten Element des ARRAY[0..10] of MOVE_UDT, in den Zielbereich kopiert. Die Kopien werden im ARRAY[10..20] of MOVE_UDT ab dem vierten Element eingefügt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK_VARIANT im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

VariantGet: Wert einer VARIANT-Variable auslesen (Seite 1214)

Beispiel für das Verschieben von Daten (Seite 348)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu FUP (Seite 7493)

UMOVE_BLK: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.						

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

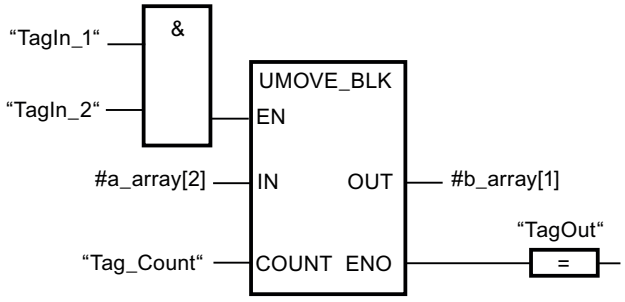

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Hinweis

Weitere Informationen zur Anweisung UMOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

FILL_BLK: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Adresse im Zielbereich, ab der befüllt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

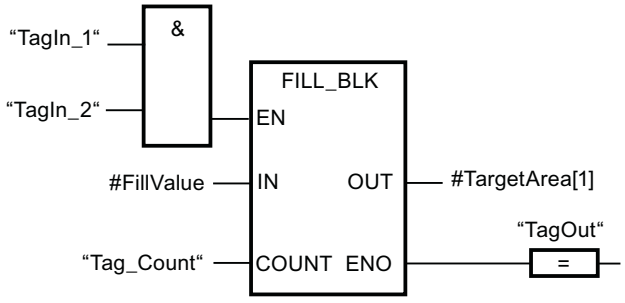

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

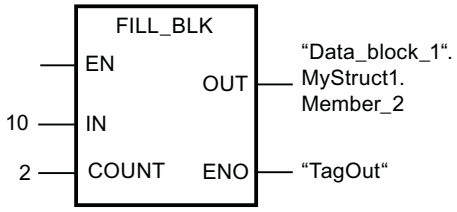

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

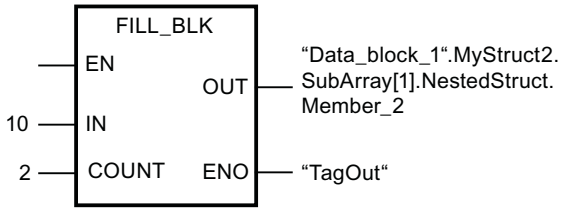

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

UFILL_BLK: Bereich ununterbrechbar befüllen

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Wenn ein ARRAY of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	D, L	Adresse im Zielbereich, ab der befüllt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

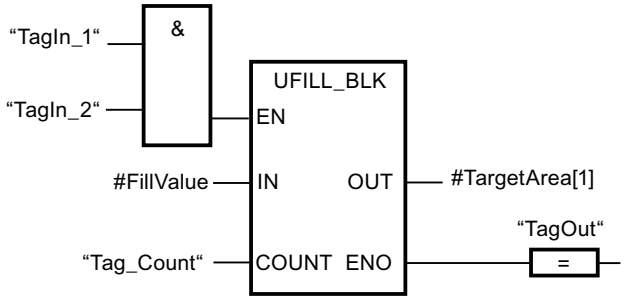

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" am Freigabeausgang ENO auf den Signalzustand "1" gesetzt.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

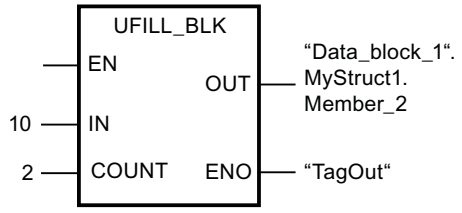

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

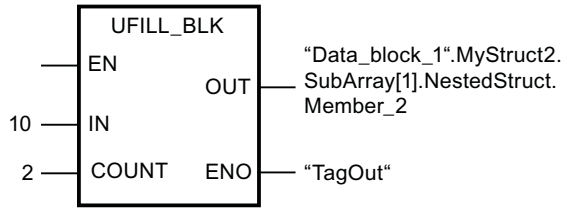

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen

Elemente, Member_1 und Member_4, werden nicht verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Operand "TagOut" auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SCATTER: Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Bitfolge in einzelne Bits zerlegen" zerlegen Sie eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Bitfolge in einzelne Bits zerlegen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der anonyme STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. ein Statuswort zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der STRUCT oder der PLC-Datentyp nicht genügend BOOL-Elemente zur Verfügung stellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Bitfolge, die zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das bzw. den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceWord		WORD
EnableOut	Output	BOOL
DestinationArray		ARRAY[0..15] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

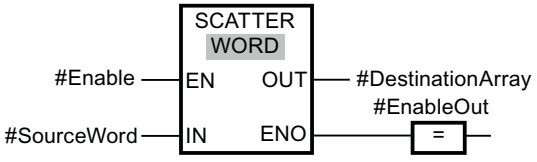

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWORD	WORD (16 Bits)
OUT	DestinationArray	Der Operand "DestinationArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationArray zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

		Name	Datentyp
1		GeneralError	Bool
2		DeviceError	Bool
3		CommError	Bool
4		myError1	Bool
5		myError2	Bool
6		myError3	Bool
7		myError4	Bool
8		myError5	Bool
9		myError6	Bool
10		myError7	Bool
11		myError8	Bool
12		myError9	Bool
13		myError10	Bool
14		myError11	Bool
15		myError12	Bool
16		myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceWord		WORD
EnableOut	Output	BOOL
DestinationUDT		"myBits"

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

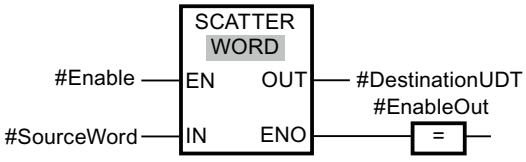

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationUDT	Der Operand "DestinationUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationUDT zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen" zerlegen Sie ein oder mehrere Elemente eines ARRAY of BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen. Am Parameter COUNT_IN geben Sie an, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Das Quell-ARRAY am Parameter IN darf mehr Elemente haben, als am Parameter COUNT_IN angegeben sind. Das ARRAY of BOOL, der anonyme STRUCT oder der PLC-Datentyp muss genügend Elemente haben, um die Bits der zerlegten Bitfolgen speichern zu können. Der Zielspeicherbereich darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter IN ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter OUT ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Das ARRAY of WORD wird in 48 einzelne Bits zerlegt.

Hinweis

Wenn die untere ARRAY-Grenze des Ziel-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. Statuswörter zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Quell-ARRAY hat weniger Elemente als am Parameter COUNT_IN angegeben sind.
- Der Index des Ziel-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of BOOL geschrieben.
- Das ARRAY[*] of BOOL, der STRUCT oder der PLC-Datentyp stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zerlegt und in das ARRAY of BOOL, den anonymen STRUCT oder den PLC-Datentyp geschrieben. Die restlichen Bitfolgen werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, das zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
COUNT_IN	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das/den die einzelnen Bits gespeichert werden

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayWord		ARRAY[0..5] of WORD
CounterInput		UDINT
EnableOut	Output	BOOL
DestinationArrayBool		ARRAY[0..95] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

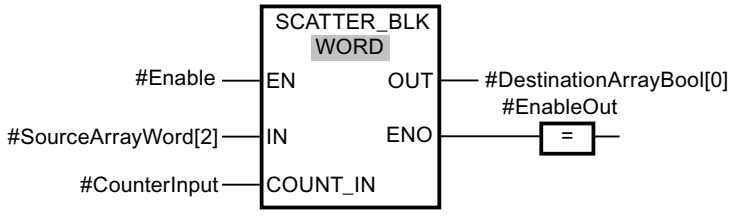

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[0]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 1. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayWord		ARRAY[0..5] of WORD
CounterInput		UDINT
EnableOut	Output	BOOL
DestinationArrayBool		ARRAY[-2..93] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

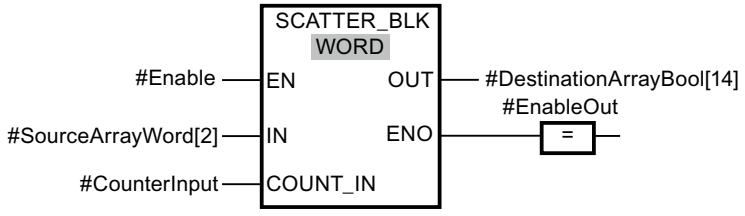

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[14]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 16. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0". Die restlichen 32 Bits werden nicht beschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einer Bitfolge zusammen. Die Bitfolge wird in einer Variable vom Datentyp BYTE, WORD, DWORD oder LWORD gespeichert.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der anonyme STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der anonyme STRUCT oder der PLC-Datentyp (UDT) weniger oder mehr BOOL-Elemente hat als die Bitfolge vorgibt, dann werden sie nicht übertragen und das ENO liefert den Signalzustand "0". Das ENO ist ebenfalls "0", wenn weniger als die notwendige Anzahl an Bits vorhanden ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, dessen Bits zu einer Bitfolge zusammengefügt werden. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Zusammengefügte Bitfolge, gespeichert in einer Variable

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArray		ARRAY[0..15] of BOOL
EnableOut	Output	BOOL
DestinationWord		WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

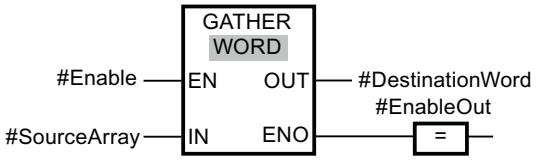

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArray	Der Operand "SourceArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Die Bits des Operanden #SourceArray werden zu einem WORD zusammengefügt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceUDT		"myBits"
EnableOut	Output	BOOL
DestinationWord		WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

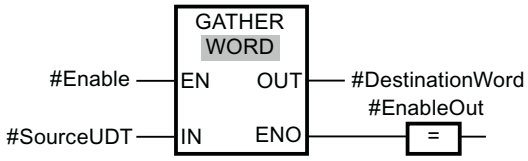

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceUDT	Der Operand "SourceUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Die Bits des Operanden #SourceUDT werden zu einem WORD

zusammengefügt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einem oder mehreren Elementen eines ARRAY of <Bitfolge> zusammen. Am Parameter COUNT_OUT geben Sie an, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Damit geben Sie auch implizit vor, wie viele Elemente des ARRAY of BOOL, des anonymen STRUCTs oder des PLC-Datentyps benötigt werden. Das Ziel-ARRAY am Parameter OUT darf mehr Elemente haben, als am Parameter COUNT_OUT angegeben sind. Das ARRAY of <Bitfolge> muss genügend Elemente haben, um die Bits, die zusammengefügt werden sollen, speichern zu können. Das Ziel-ARRAY darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter OUT ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter IN ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Aus dem ARRAY of BOOL werden 48 einzelne Bits zusammengefügt.

Hinweis

Wenn die untere ARRAY-Grenze des Quell-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Index des Quell-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of <Bitfolge> geschrieben.
- Das ARRAY[*] of <Bitfolge> stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zusammengefügt und in das ARRAY of <Bitfolge> geschrieben. Die restlichen Bits werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY of BOOL, STRUCT oder PLC-Datentyp, dessen Bits zusammengefügt werden (Quell-ARRAY). Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_OUT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, in das die Bits gespeichert werden (Ziel-ARRAY)

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayBool		ARRAY[0..95] of BOOL
CounterOutput		UDINT
EnableOut	Output	BOOL
DestinationArrayWord		ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

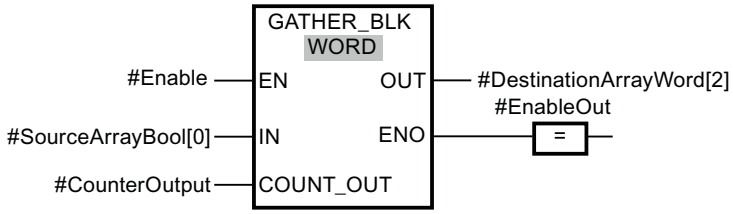

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[0]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 1. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits in das 3. Wort, die zweiten 16 Bits in das 4. Wort und die dritten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
Enable	Input	BOOL
SourceArrayBool		ARRAY[-2..93] of BOOL
CounterOutput		UDINT
EnableOut	Output	BOOL
DestinationArrayWord		ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

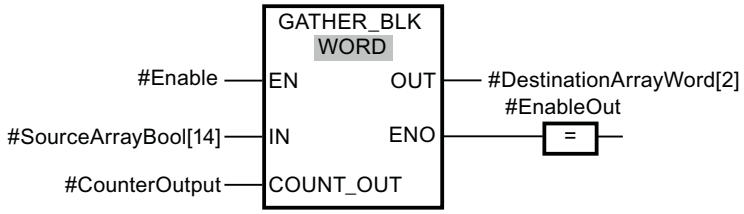

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[14]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Da aber erst ab dem 16. Element begonnen wird, stehen nur 80 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 16. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits des Quell-ARRAYs nicht berücksichtigt werden. Die zweiten 16. Bits werden in das 3. Wort, die dritten 16 Bits in das 4. Wort und die vierten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Die restlichen 64 Bits des Quell-ARRAYs werden ebenfalls nicht berücksichtigt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

AssignmentAttempt: Zuweisung eines VARIANT auf eine Referenz versuchen

Beschreibung

Mit der Anweisung "AssignmentAttempt" versuchen Sie die Zuweisung einer VARIANT-Variable auf eine Referenzvariable. Der Datentyp einer Referenzvariable wird zum Zeitpunkt der Deklaration festgelegt, während der Datentyp einer VARIANT-Variable während der Laufzeit ermittelt wird. Eine implizite Datentypkonvertierung ist bei Referenzvariablen nicht zulässig. Um die beiden Datentypen einander zuzuweisen, nutzen Sie deshalb den Zuweisungsversuch.

Beim Zuweisungsversuch wird zur Laufzeit geprüft, ob die VARIANT-Variable vom richtigen Datentyp ist. Wenn dies der Fall ist, wird die Zuweisung ausgeführt. Nach einer erfolgreichen Ausführung steht eine gültige Referenz in der Zielvariablen, andernfalls eine NULL.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL		Freigabeeingang
ENO	Output	BOOL		Freigabeausgang
SRC	Input	VARIANT	• Bausteinschnittstelle eines FC: Input, Output, InOut, Temp • Bausteinschnittstelle eines FB: Input, InOut, Temp	Zeiger auf die Quellvariable, deren Adresse ausgelesen wird
DST	Output	Referenz auf: • Bitfolgen, außer BOOL, • Ganzzahlen, • Gleitpunktzahlen, • Zeichenfolgen, • PLC-Datentypen (UDT), • Systemdatentypen (SDT), • ARRAYs aus den genannten Datentypen	• Bausteinschnittstelle eines FC: Input, Output, Temp, Return • Bausteinschnittstelle eines FB: Temp	Referenz, an die die Adresse der Quellvariable übergeben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Regeln

Für den Zuweisungsversuch gelten die folgenden Regeln. VARIANT-Variablen, die nicht diesen Regeln entsprechen, liefern zur Laufzeit den Wert "NULL" zurück.

- Der VARIANT muss auf eine Adresse in einem optimierten Speicherbereich zeigen.
- Der VARIANT darf nicht auf eine Adresse in einem temporären Speicherberiech zeigen.
- Wenn Sie einer Referenz auf ein ARRAY einen VARIANT zuweisen wollen, gelten folgende Regeln:
  - Die VARIANT-Variable muss auf ein ARRAY zeigen, dessen Grenzen exakt mit denen der deklarierten Referenz übereinstimmen. Eine VARIANT-Variable, die auf ein ARRAY [0..9] zeigt, passt nicht zu einer Variable REF_TO ARRAY[1..10].
  - Darüberhinaus sollten Sie die Bausteine, die den Wert der VARIANT-Variable bilden, einmalig in einer CPU der Baureihe S7-1500, Firmwarestand V2.5, übersetzen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

In der Schnittstelle des Bausteins wurden der VARIANT "myVariant" und die Referenzvariable "myReference" deklariert.

	Name	Data type
	Input	
	Output	
	InOut	
	myVariant	Variant
	Static	
	Temp	
	myReference	REF_TO Int
	Constant	

Im Programmcode wird versucht "myVariant" der Referenzvariable "myReference" zuzuweisen. Hat "myVariant" zur Laufzeit den Datentyp "Int", steht in "myReference" eine gültige Referenz auf die Quellvariable des VARIANT, andernfalls eine NULL. Wenn der Zuweisungsversuch erfolgreich war, führt der Freigabeausgang "ENO" Signalzustand "1" und der Wert "10" kann in die Zielvariable geschrieben werden.

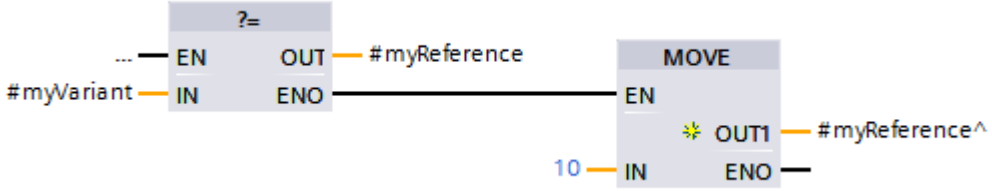

Siehe auch

Zuweisungsversuch von VARIANT auf eine Referenz (Seite 329)

Grundlagen zu Referenzen (Seite 315)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Sample Library for Instructions (Seite 615)

SWAP: Anordnung ändern

Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes am Eingang IN und fragt das Ergebnis am Ausgang OUT ab.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" geändert werden:

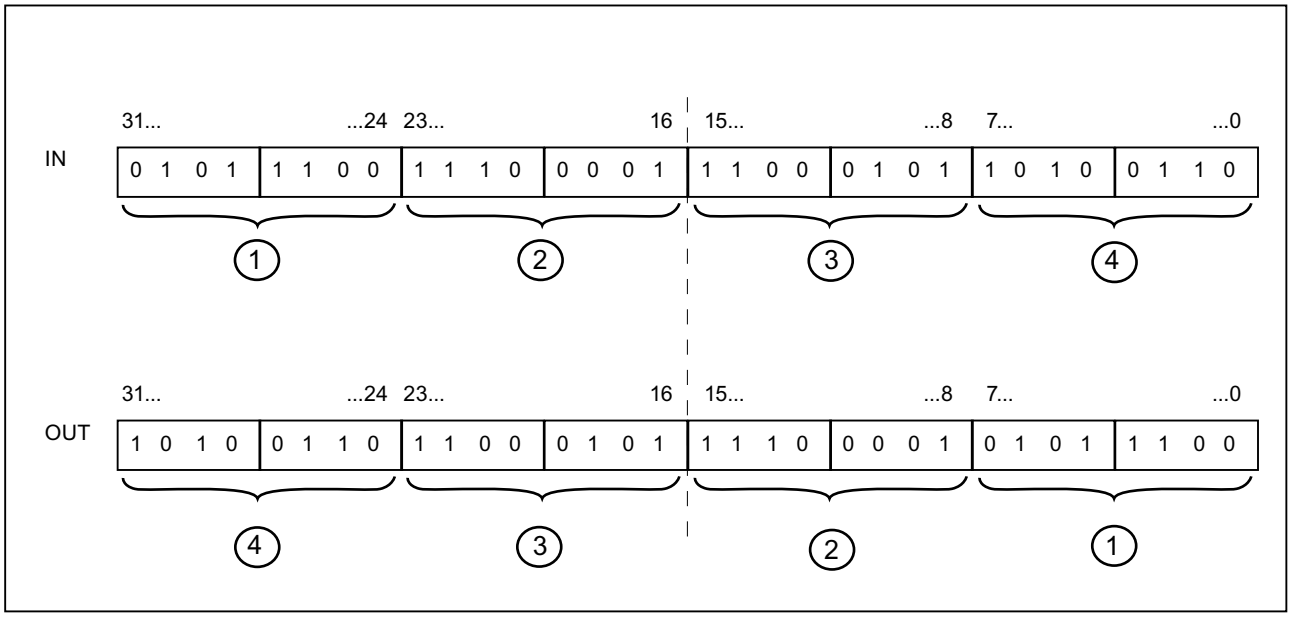

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anordnung ändern":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Operand, dessen Bytes geändert werden.
OUT	Output	WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

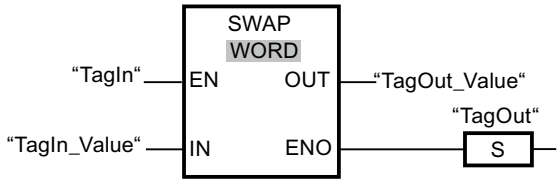

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0000 1111 0101 0101
OUT	TagOut_Value	0101 0101 0000 1111

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anordnung der Bytes wird geändert und im Operanden "TagOut_Value" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ARRAY-DB

ReadFromArrayDB: Aus Array-Datenbaustein lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie das Element aus einem Datenbaustein vom Bausteintyp ARRAY-DB, auf das der Index verweist, und schreiben den Wert des Elements in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Der Freigabeausgang ENO liefert den Signalzustand FALSE, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand FALSE.
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der gelesen und ausgegeben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter VALUE ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz, schreibgeschützt oder liegt im Ladespeicher.
8135	Der ARRAY-Datenbaustein enthält ungültige Werte.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8452	Codeerzeugungsfehler
8453	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

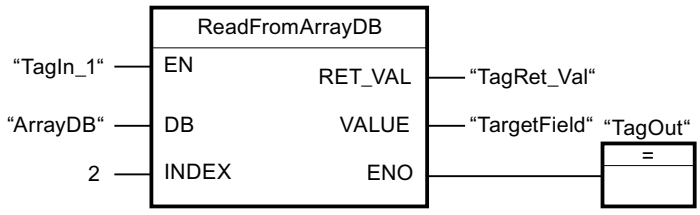

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.

Wenn der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und in den Operanden "TargetField" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu FUP (Seite 7493)

WriteToArrayDB: In Array-Datenbaustein schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Der Freigabeausgang ENO liefert den Signalzustand FALSE, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand FALSE.
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der geschrieben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8134	Der Datenbaustein ist schreibgeschützt.
8135	Der Datenbaustein ist kein ARRAY-Datenbaustein.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8350	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8352	Codeerzeugungsfehler

Fehler-code* (W#16#...)	Erläuterung
8353	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

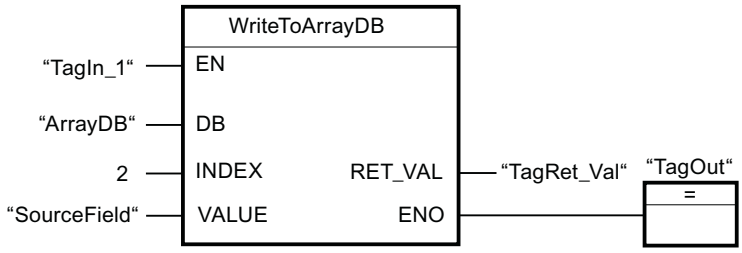

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Wenn der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "SourceField" wird in das 2. Element des ARRAY-DB geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu FUP (Seite 7493)

ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie aus einem Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher das Element, auf das der Index verweist, und schreiben sie in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Auslesen des Array-DBs beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALUE [1]	InOut	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8750	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters s stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

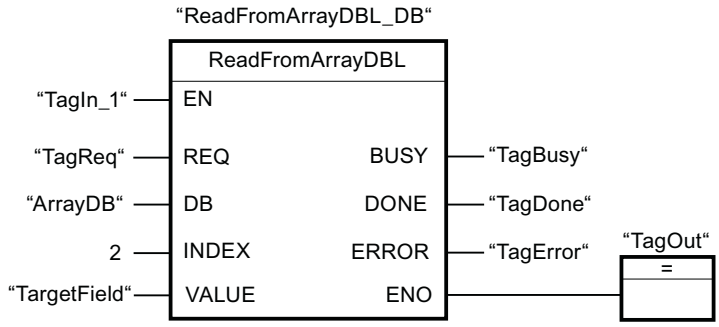

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn der Operand "TagIn1" den Signalzustand "1" liefert und am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und am Parameter VALUE ausgegeben. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird nicht mehr verändert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu FUP (Seite 7493)

WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Schreiben in den Array-DB beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALUE [1]	Input	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8234	Der Datenbaustein ist schreibgeschützt.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

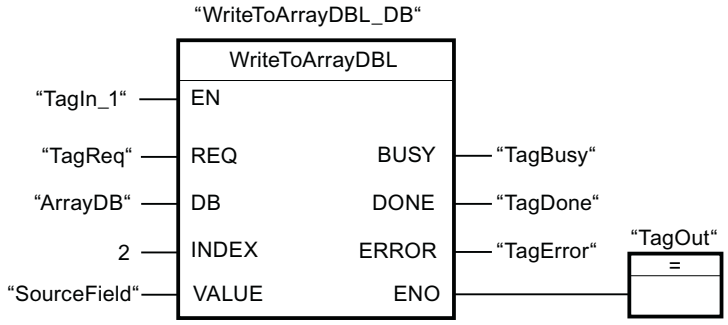

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn der Operand "TagIn1" den Signalzustand "1" liefert und am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird in das 2. Element des "ArrayDB" geschrieben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand TRUE und der Ausgang "TagOut" wird gesetzt. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zum VARIANT (Seite 339)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu FUP (Seite 7493)

VARIANT

VariantGet: Wert einer VARIANT-Variable auslesen

Beschreibung

Mit der Anweisung "Wert einer VARIANT-Variable auslesen" lesen Sie den Wert der Variable aus, auf die der VARIANT am Parameter SRC zeigt, und schreiben diesen in die Variable am Parameter DST.

Der Parameter SRC hat den Datentyp VARIANT. Am Parameter DST kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable, die am Parameter DST angegeben ist, muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Datentypen stimmen nicht überein. (Es wird kein Wert übertragen.)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert einer VARIANT-Variable auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
SRC	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Variable, die ausgelesen wird
DST	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

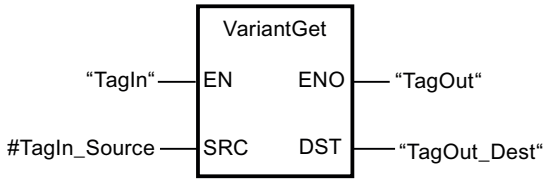

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen und in den Operand "TagOut_Dest" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1167)

Grundlagen zu FUP (Seite 7493)

VariantPut: Wert in eine VARIANT-Variable schreiben

Beschreibung

Mit der Anweisung "Wert in eine VARIANT-Variable schreiben" schreiben Sie den Wert der Variable, die am Parameter SRC angegeben ist, in den Speicher am Parameter DST, auf den der VARIANT zeigt.

Der Parameter DST hat den Datentyp VARIANT. Am Parameter SRC kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable am Parameter SRC muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Datentypen stimmen nicht überein. (Es wird kein Wert übertragen.)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert in eine VARIANT-Variable schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
SRC	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Variable, die ausgelesen wird
DST	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

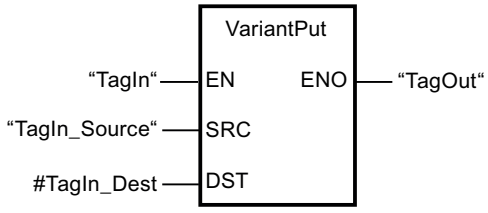

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "TagIn_Source" wird in die Variable, auf die der VARIANT am Operand #TagIn_Dest zeigt, geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1167)

Grundlagen zu FUP (Seite 7493)

CountOfElements: Anzahl der ARRAY-Elemente abfragen

Beschreibung

Mit der Anweisung "Anzahl der ARRAY-Elemente abfragen" fragen Sie ab, wie viele ARRAY-Elemente eine Variable hat, auf die der VARIANT zeigt.

Wenn es sich um ein eindimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl der ARRAY-Elemente ausgegeben. (Die Differenz zwischen dem oberen und dem unteren Grenzwert + 1). Wenn es sich um ein mehrdimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl aller Dimensionen ausgegeben.

Wenn Sie die Elemente eines ARRAY-DB abfragen möchten, dann sollten Sie die Anweisungen "ReadFromArrayDB" bzw. "WriteFromArrayDB" verwenden, da hier eine genauere Fehlerauswertung für die Anzahl der Elemente möglich ist.

Hinweis

Instanzen

Der VARIANT-Zeiger kann auf keine Instanz und damit auch auf keine Multiinstanz oder ein ARRAY of Multiinstanzen zeigen.

Hinweis

ARRAY innerhalb eines Datenbausteins

Wenn Sie die Anzahl der Elemente eines ARRAYs, das in einem Datenbaustein liegt, abfragen möchten, dann darf bei diesem Datenbaustein nicht das Bausteinattribut "Datenbaustein im Gerät schreibgeschützt" aktiviert sein. Ansonsten liefert der Parameter RET_VAL das Ergebnis "0", unabhängig davon, wie viele Elemente das ARRAY enthält.

Das Ergebnis ist ebenfalls "0", wenn die VARIANT-Variable kein ARRAY ist.

Wenn der VARIANT auf ein ARRAY of BOOL zeigt, dann werden auch die Füllelemente mitgezählt. (Z. B. bei einem ARRAY[0..1] of BOOL wird 8 zurückgegeben).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die VARIANT-Variable ist kein ARRAY. (Das Ergebnis ist "0".)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der ARRAY-Elemente abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die abgefragt wird
RET_VAL	Output	UDINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

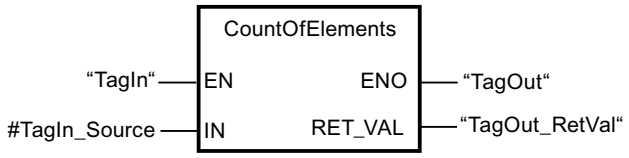

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anzahl der ARRAY-Elemente der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen und am Operand "TagOut_RetVal" ausgegeben.

Siehe auch

Grundlagen zum ARRAY (Seite 296)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu FUP (Seite 7493)

ARRAY[*]

LOWER_BOUND: Untere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, untere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Untere ARRAY-Grenze auslesen" zur Verfügung.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Dimension, die am Eingang DIM angegeben wurde, existiert nicht.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Untere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, untere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, untere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

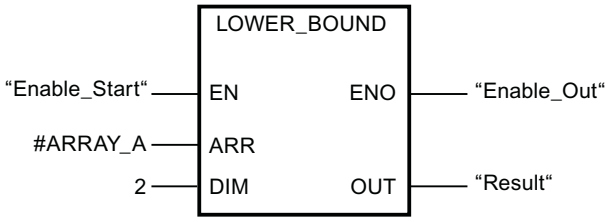

Wenn der Operand "Enable_Start" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Sie liest die variable, untere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben und der Operand "Enable_Out" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu FUP (Seite 7493)

UPPER_BOUND: Obere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, obere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Obere ARRAY-Grenze auslesen" zur Verfügung.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die Dimension, die am Eingang DIM angegeben wurde, existiert nicht.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Obere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, obere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, obere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

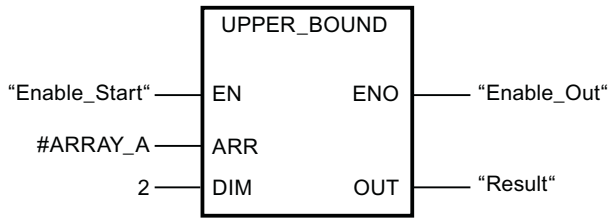

Wenn der Operand "Enable_Start" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Sie liest die variable, obere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben und der Operand "Enable_Out" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu FUP (Seite 7493)

Legacy

FieldRead: Feld lesen

Beschreibung

Mit der Anweisung "Feld lesen" lesen Sie eine bestimmte Komponente aus dem am Parameter MEMBER angegebenen Feld aus und übertragen deren Inhalt in die Variable am Parameter VALUE. Den Index der zu lesenden Feldkomponente legen Sie am Parameter INDEX fest. Am Parameter MEMBER geben Sie die erste Komponente des Felds an, aus dem gelesen wird.

Die Datentypen der Feldkomponente am Parameter MEMBER, des Index und der Variablen am Parameter VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen, da eine implizite Konvertierung nicht möglich ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Parameter INDEX angegebene Feldkomponente ist in dem am Parameter MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld lesen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
INDEX	Input	DINT	DINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Index der Feldkomponente, deren Inhalt ausgelesen wird

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
MEMBER	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	D, L	D, L	Erste Komponente des Felds, aus dem gelesen wird
VALUE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR, WCHAR	E, A, M, D, L, P	E, A, M, D, L, P	Operand, in den der Inhalt der Feldkomponente übertragen wird

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

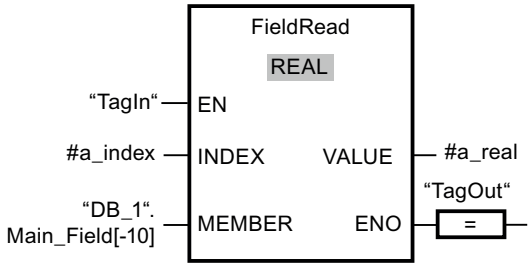

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Variable	Wert
INDEX	a_index	4
MEMBER	"DB_1".Main_Field[-10]	Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1"
VALUE	a_real	Komponente mit Index 4 des Felds "Main_Field[-10..10] of REAL"

Die Feldkomponente mit Index 4 wird aus dem Feld "Main_Field[-10...10] of REAL" ausgelesen und in die Variable "a_real" geschrieben. Die zu lesende Feldkomponente wird durch den Wert am Parameter INDEX festgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

FieldWrite: Feld schreiben

Beschreibung

Mit der Anweisung "Feld schreiben" übertragen Sie den Inhalt der Variablen am Eingang VALUE in eine bestimmte Komponente des Felds am Ausgang MEMBER. Den Index der Feldkomponente, die beschrieben wird, legen Sie durch den Wert am Eingang INDEX fest. Am Ausgang MEMBER geben Sie die erste Komponente des Felds an, in das geschrieben wird.

Die Datentypen der Feldkomponente am Parameter MEMBER, des Index und der Variablen am Parameter VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen, da eine implizite Konvertierung nicht möglich ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Ausgang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld schreiben":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
INDEX	Input	DINT	DINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Index der Feldkomponente, die mit dem Inhalt von VALUE beschrieben wird
VALUE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Operand, dessen Inhalt kopiert wird
MEMBER	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD, CHAR und WCHAR als Komponente einer ARRAY-Variable	D, L	D, L	Erste Komponente des Felds, in welches der Inhalt von VALUE geschrieben wird.

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

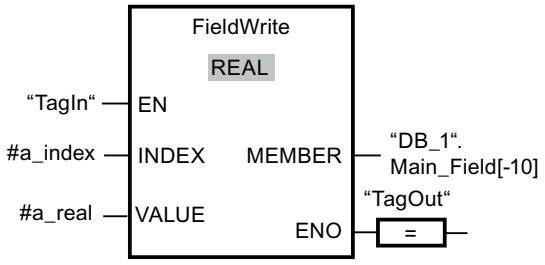

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
INDEX	a_index	4
VALUE	a_real	10.54
MEMBER	"DB_1".Main_Field[-10]	Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1"

Der Wert "10.54" der Variablen "a_real" wird in die Feldkomponente mit Index 4 des Felds "Main_Field[-10 ... 10] of REAL" geschrieben. Der Index der Feldkomponente, in die der Inhalt der Variablen "a_real" übertragen wird, wird durch den Wert am Eingang INDEX festgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

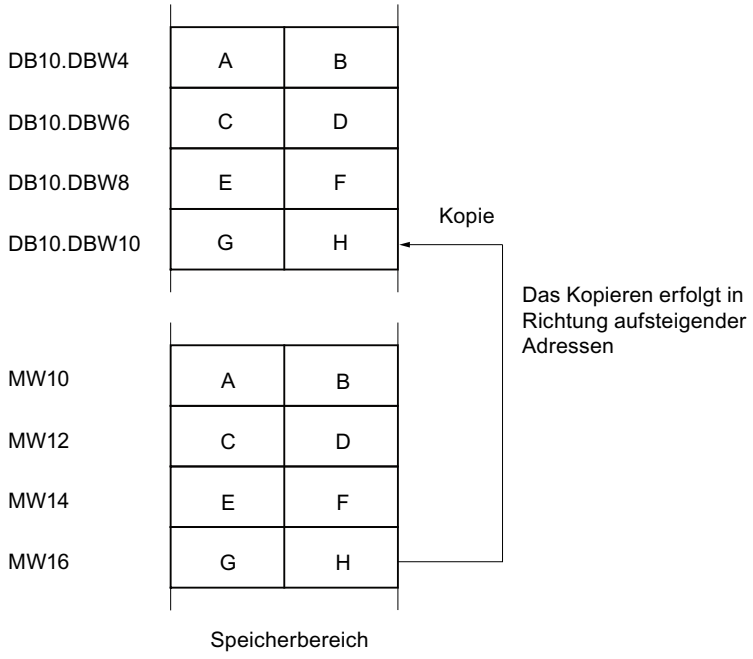

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.

Fehlercode* (W#16#...)	Erläuterung
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

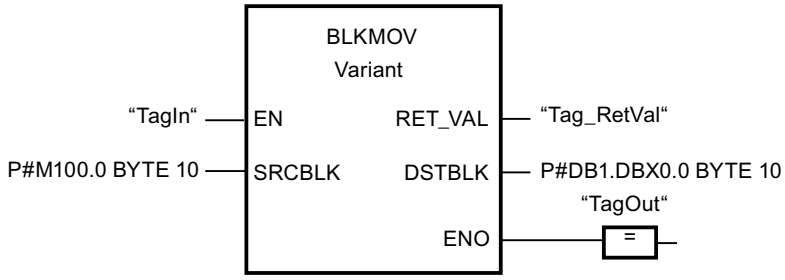

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.

Fehlercode (W#16#...)	Erläuterung
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

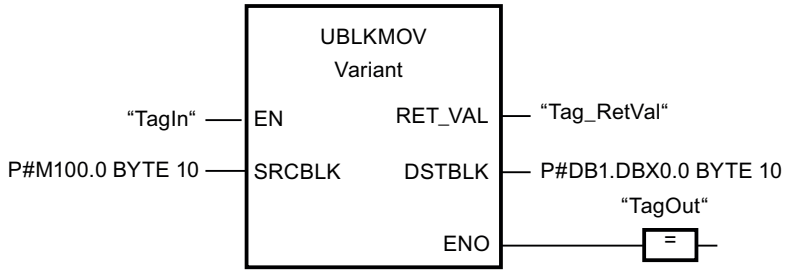

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

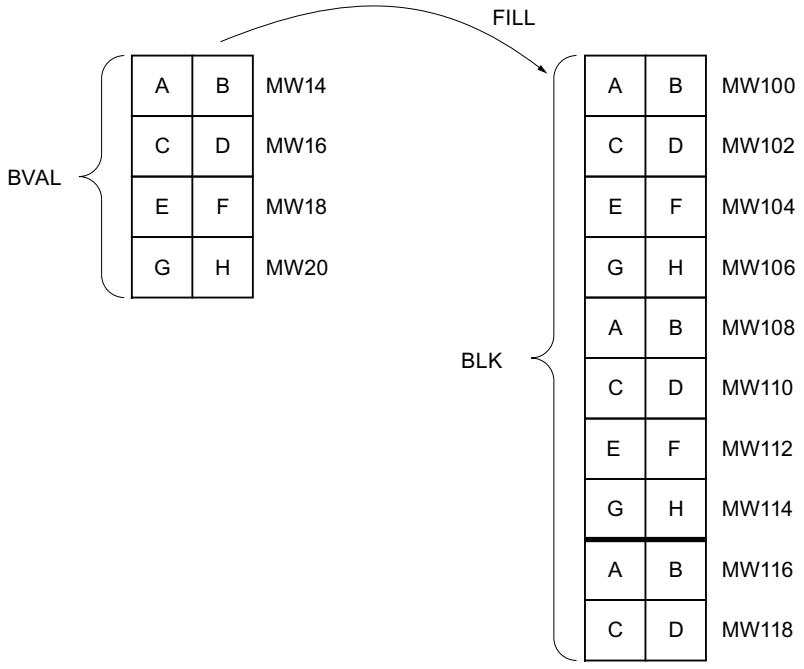

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker

- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
BVAL	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter BVAL werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter BLK werden dieDatentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

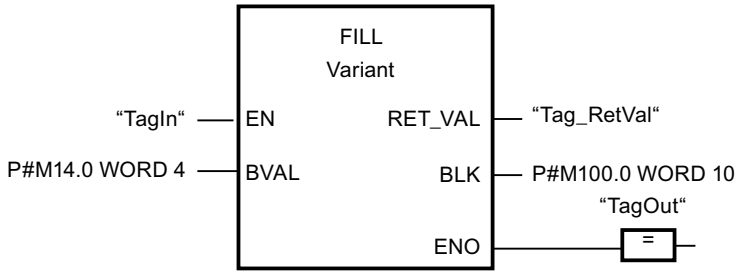

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend den in der Anweisungsbox parametrierten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Konvertierungsmöglichkeiten von Bitfolgen

Die Bitfolgen BYTE und WORD sind in der Anweisungsbox nicht auswählbar. Es ist allerdings möglich, einen Operanden vom Datentyp DWORD oder LWORD an einem Parameter der Anweisung anzugeben, wenn die Länge des Eingangs- und des Ausgangsoperanden übereinstimmt. Dann wird der Operand vom Datentyp einer Bitfolge entsprechend des Datentyps des Eingangs- oder Ausgangsparameters interpretiert und implizit konvertiert. Der Datentyp DWORD wird z. B. als DINT/UDINT und LWORD als LINT/ULINT interpretiert. Diese Konvertierungsmöglichkeiten stehen Ihnen auch bei aktivierter IEC-Prüfung zur Verfügung.

Hinweis

Für CPUs der Baureihen S7-1500 gilt: Die Datentypen DWORD und LWORD können nur von bzw. nach Datentyp REAL oder LREAL konvertiert werden.

Dabei wird das Bitmuster des Quellwerts ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand des Freigabeausgangs ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand des Freigabeausgangs ENO = 0.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Konvertierung einer Ganzzahl (16 Bit) in eine andere Ganzzahl (32 Bit):

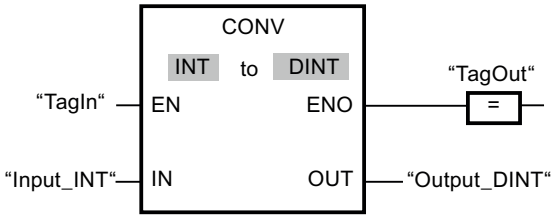

Das folgende Beispiel zeigt die Konvertierung eines Bytes (8 Bit) in die Ganzzahl SINT (8 Bit):

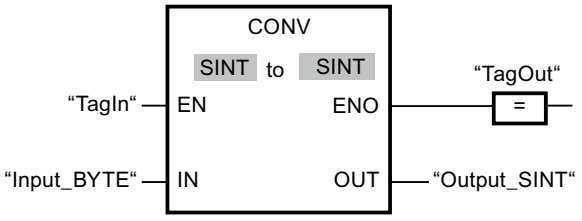

Das folgende Beispiel zeigt die Konvertierung eines Bytes (8 Bit) in die vorzeichenlose Ganzzahl USINT (8 Bit):

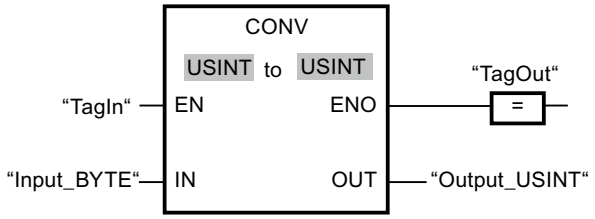

Die Konvertierungen sind möglich, da die Länge der beiden Operanden gleich ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Datentypkonvertierungen bei S7-1200 (Seite 507)

Grundlagen zu FUP (Seite 7493)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die am nächsten liegende Ganzzahl um. Wenn der Eingangswert zwischen zwei Zahlen liegt, wird die gerade Zahl gewandelt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird.
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Rundung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

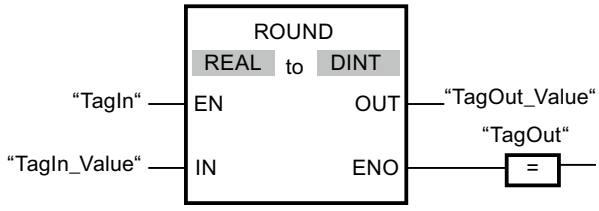

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächstliegenden geraden Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Explizite Konvertierungen (Seite 530)

Grundlagen zu FUP (Seite 7493)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert als Gleitpunktzahl
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

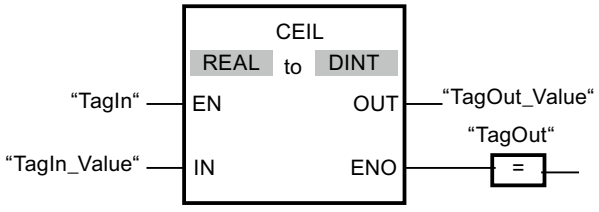

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Explizite Konvertierungen (Seite 530)

Grundlagen zu FUP (Seite 7493)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert als Gleitpunktzahl
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

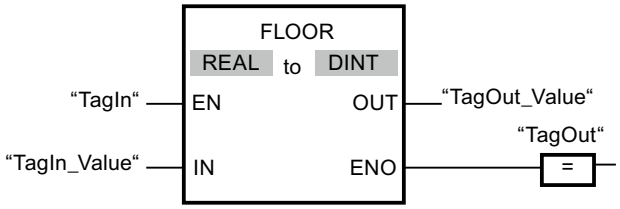

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Explizite Konvertierungen (Seite 530)

Grundlagen zu FUP (Seite 7493)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Eingangwert als Gleitpunkt-zahl
OUT	Output	Ganzzahlen, Gleit-punktzahlen	E, A, M, D, L	Ergebnis mit ganzzahligem Anteil der Gleitpunktzahl

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

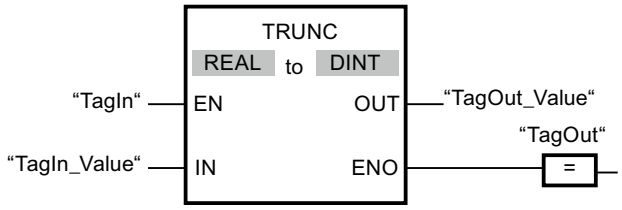

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Explizite Konvertierungen (Seite 530)

Grundlagen zu FUP (Seite 7493)

SCALE_X: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Eingang VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung "Skalieren" wird der Gleitpunktwert am Eingang VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang OUT abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:

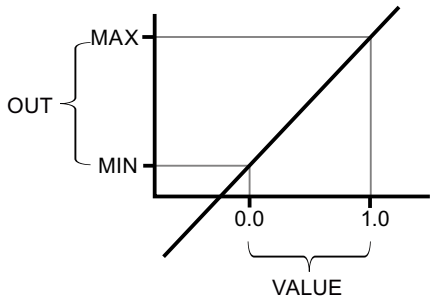

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [VALUE * (MAX – MIN)] + MIN

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Eingang VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der skaliert wird. Bei Angabe einer Konstante müssen Sie diese deklarieren.
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Skalierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

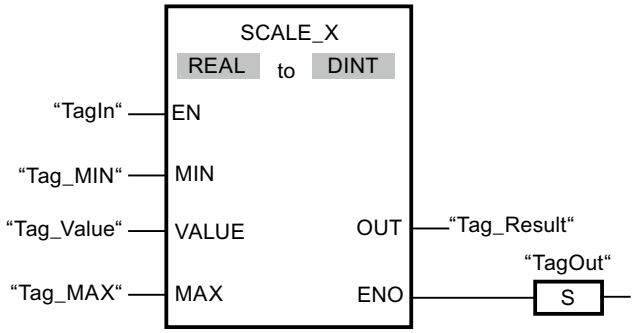

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_MIN	10
VALUE	Tag_Value	0.5
MAX	Tag_MAX	30
OUT	Tag_Result	20

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert am Eingang "Tag_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Eingängen "Tag_MIN" und "Tag_MAX" definiert wurde. Das Ergebnis wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

NORM_X: Normieren

Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen des Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert der Ausgang OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert Ausgang OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:

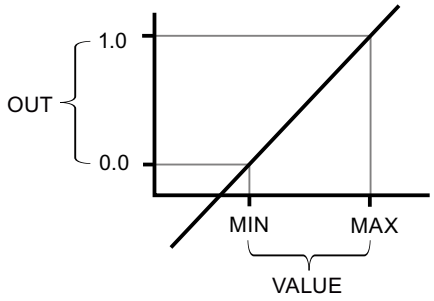

Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

OUT = (VALUE – MIN) / (MAX – MIN)

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Der Wert am Eingang VALUE ist NaN (Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Normieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
MIN [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der normiert wird.
MAX [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
OUT	Output	Gleitpunktzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Normierung
[1] Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren.					

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

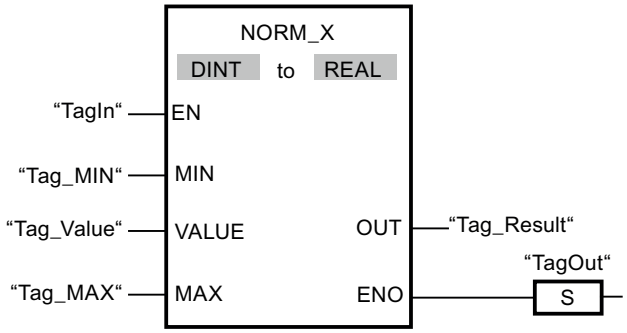

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_MIN	10
VALUE	Tag_Value	20
MAX	Tag_MAX	30
OUT	Tag_Result	0.5

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert am Eingang "Tag_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Eingängen "Tag_MIN" und "Tag_MAX" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Eingang "Tag_Value" normiert. Das Ergebnis wird am Ausgang "Tag_Result" als Gleitpunktzahl abgelegt. Wenn keine Fehler während der

Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Legacy

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	INT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

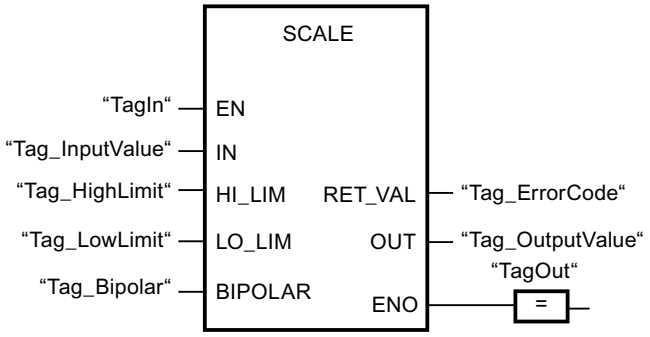

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandelt sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Input	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
OUT	Output	INT	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

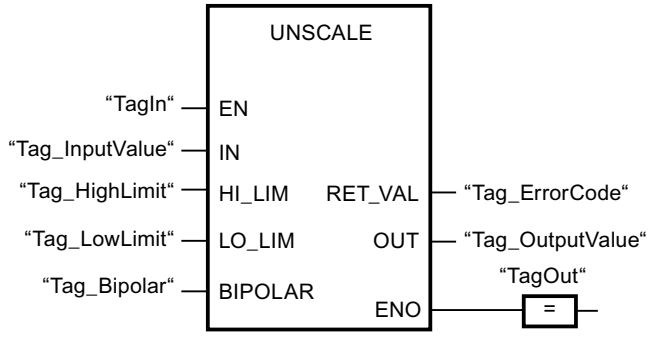

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

Programmsteuerung

JMP: Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

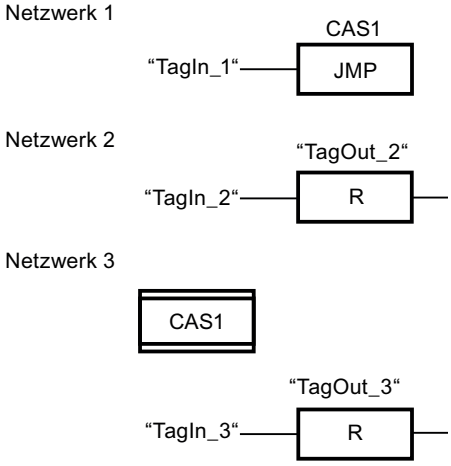

Wenn der Operand "TagIn_1"den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

JMPN: Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das VKE am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

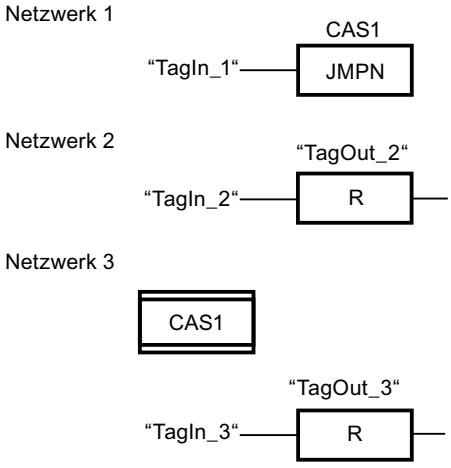

Wenn der Operand "TagIn_1" den Signalzustand "0" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

LABEL: Sprungmarke

Beschreibung

Die Sprungmarke kennzeichnet ein Zielnetzwerk, in dem nach einer ausgeführten Sprunganweisung die Programmbearbeitung fortgesetzt werden soll.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Sie können maximal 32 Sprungmarken bei Verwendung einer CPU S7-1200 und maximal 256 Sprungmarken bei Verwendung einer CPU S7-1500 deklarieren.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

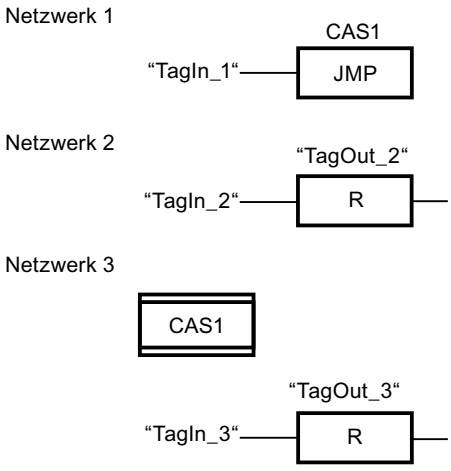

Wenn der Operand "TagIn_1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

JMP_LIST: Sprungliste definieren

Beschreibung

Mit der Anweisung "Sprungliste definieren" definieren Sie mehrere bedingte Sprünge und setzen die Programmbearbeitung, abhängig vom Wert des Parameters K, in einem bestimmten Netzwerk fort.

Die Sprünge definieren Sie durch Sprungmarken (LABEL), die Sie an den Ausgängen der Anweisungsbox angeben. Die Anzahl der Ausgänge ist in der Anweisungsbox erweiterbar. Sie können maximal 32 Ausgänge bei Verwendung einer CPU S7-1200 und maximal 256 Ausgänge bei Verwendung einer CPU S7-1500 deklarieren.

Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung können nur Sprungmarken angegeben werden. Die Angabe von Anweisungen oder Operanden ist nicht zulässig.

Durch den Wert des Parameters K geben Sie die Nummer des Ausgangs und damit die Sprungmarke an, bei der die Programmbearbeitung fortgesetzt wird. Wenn der Wert am Parameter K größer als die Anzahl der verfügbaren Ausgänge ist, wird die Programmbearbeitung im nächsten Netzwerk des Bausteins fortgesetzt.

Die Anweisung "Sprungliste definieren" wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungliste definieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, L, D oder Konstante	Freigabeeingang
K	Input	UINT	E, A, M, L, D oder Konstante	Gibt die Nummer des Ausgangs und damit den Sprung an, der ausgeführt wird.
DEST0	-	-	-	Erste Sprungmarke
DEST1	-	-	-	Zweite Sprungmarke
DESTn	-	-	-	Optionale Sprungmarken

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

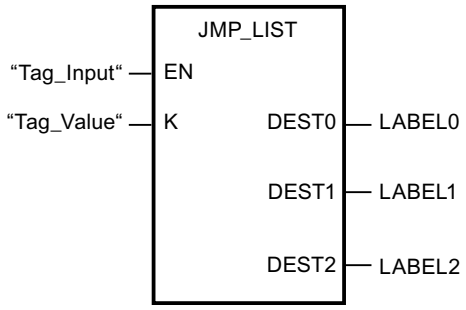

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand/Sprungmarke	Wert
K	Tag_Value	1
DEST0	LABEL0	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL0" gekennzeichnet ist.
DEST1	LABEL1	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.
DEST2	LABEL2	Sprung in das Netzwerk, das mit der Sprungmarke "LABEL2" gekennzeichnet ist.

Wenn der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Programmbearbeitung wird entsprechend dem Wert des Operanden "Tag_Value" in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SWITCH: Sprungverteiler

Beschreibung

Mit der Anweisung "Sprungverteiler" definieren Sie mehrere Programmsprünge, die abhängig vom Ergebnis einer oder mehrer Vergleichsanweisungen ausgeführt werden.

Den zu vergleichenden Wert geben Sie am Parameter K an. Dieser Wert wird mit den Werten verglichen, die die einzelnen Eingänge liefern. Die Art des Vergleichs wählen Sie für jeden Eingang einzeln aus. Die Verfügbarkeit der verschiedenen Vergleichsanweisungen hängt dabei vom Datentyp der Anweisung ab.

Die folgende Tabelle zeigt die Vergleichsanweisungen, die abhängig vom gewählten Datentyp verfügbar sind:

Datentyp		Anweisung	Syntax
S7-1200	S7-1500		
Bitfolgen	Bitfolgen	Gleich	==
		Ungleich	<>
Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT	Gleich	==
		Ungleich	<>
		Größer gleich	>=
		Kleiner gleich	<=
		Größer	>
		Kleiner	<

Den Datentyp der Anweisung wählen Sie aus der Klappliste "???" der Anweisungsbox. Wenn Sie eine Vergleichsanweisung auswählen und der Datentyp der Anweisung noch nicht festgelegt ist, werden in der Klappliste "???" nur die Datentypen aufgelistet, die für die ausgewählte Vergleichsanweisung zulässig sind.

Die Bearbeitung der Anweisung beginnt mit dem ersten Vergleich und wird so lange ausgeführt, bis eine Vergleichsbedingung erfüllt ist. Ist eine Vergleichsbedingung erfüllt, werden die nachfolgenden Vergleichsbedingungen nicht berücksichtigt. Wenn keine der angegebenen Vergleichsbedingungen erfüllt ist, wird der Sprung am Ausgang ELSE ausgeführt. Wenn am Ausgang ELSE keine Sprungmarke definiert ist, wird die lineare Bearbeitung des Programms nicht unterbrochen, sondern im nächsten Netzwerk fortgesetzt.

Die Anweisungsbox enthält im Grundzustand min. 2 Ausgänge (DEST0 und DEST1). Die Anzahl der Ausgänge ist erweiterbar. Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung geben Sie Sprungmarken (LABEL) an. Die Angabe von Anweisungen oder Operanden ist an den Ausgängen der Anweisung nicht zulässig.

Zu jedem zusätzlichen Ausgang wird automatisch auch ein Eingang eingefügt. Der an einem Ausgang programmierte Sprung wird ausgeführt, wenn die Vergleichsbedingung des entsprechenden Eingangs erfüllt ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungverteiler":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
K	Input	UINT	UINT	E, A, M, D, L oder Konstante	Gibt den zu vergleichenden Wert an.
<Vergleichswerte>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Eingangswerte, mit denen der Wert des Parameters K verglichen wird.
DEST0	-	-	-	-	Erste Sprungmarke
DEST1	-	-	-	-	Zweite Sprungmarke
DEST(n)	-	-	-	-	Optionale Sprungmarken: • S7-1200: n = 2 bis 32 • S7-1500: n = 2 bis 256
ELSE	-	-	-	-	Programmsprung, der ausgeführt wird, wenn keine der Vergleichsbedingungen erfüllt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

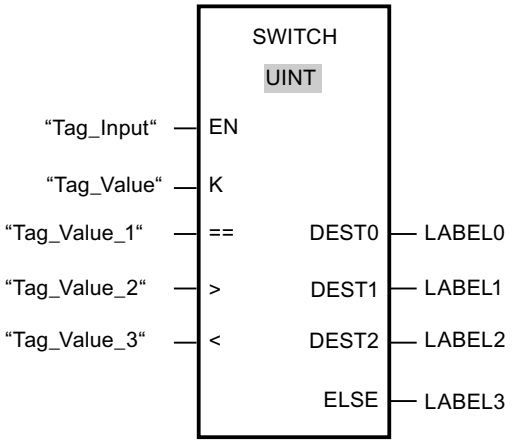

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand/Sprungmarke	Wert
K	Tag_Value	23
==	Tag_Value_1	20
>	Tag_Value_2	21
<	Tag_Value_3	19
DEST0	LABEL0	Sprung zur Sprungmarke "LABEL0", wenn der Wert des Parameters K gleich 20 ist.
DEST1	LABEL1	Sprung zur Sprungmarke "LABEL1", wenn der Wert des Parameters K größer 21 ist.
DEST2	LABEL2	Sprung zur Sprungmarke "LABEL2", wenn der Wert des Parameters K kleiner als 19 ist.
ELSE	LABEL3	Sprung zur Sprungmarke "LABEL3", wenn keine der Vergleichsbedingungen erfüllt ist.

Wenn der Operand "Tag_Input" auf den Signalzustand "1" wechselt, wird die Anweisung ausgeführt. Die Programmbearbeitung wird in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

RET: Zurück springen

Beschreibung

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Programmbausteins bedingt oder absolut. Der Signalzustand des Rückgabewerts (Operand) wird beim Verlassen des Programmbausteins im Freigabeausgang ENO des aufrufenden Programmbausteins abgebildet.

Es gibt drei Arten, wie die Bearbeitung eines Programmbausteins beendet werden kann:

Beendigung des Programmbausteins	Erklärung
Ohne Aufruf der Anweisung	Nach der Ausführung des letzten Netzwerks wird der Programmbaustein verlassen. Der Freigabeausgang ENO des aufrufenden Programmbausteins wird auf den Signalzustand "1" gesetzt.
Aufruf der Anweisung mit Vorverknüpfung (siehe Beispiel)	Ist die Vorverknüpfung erfüllt, wird die Programmbearbeitung im aktuell aufgerufenen Programmbaustein beendet und der Programmbaustein verlassen. (Bedingtes Programmbausteinende) Die Programmbearbeitung wird im aufrufenden Programmbaustein (z. B. einem OB) nach dem Programmbausteinaufruf fortgesetzt. Der Freigabeausgang ENO des aufrufenden Programmbausteins entspricht dem Operanden.
Aufruf der Anweisung ohne Vorverknüpfung oder die Anweisung ist direkt an der linken Stromschiene angeschlossen	Der Programmbaustein wird unbedingt verlassen. (Absolutes Programmbausteinende) Der Freigabeausgang ENO des aufrufenden Porgrammbausteins entspricht dem Operanden.

Wenn ein Organisationsbaustein (OB) beendet wird, dann wird durch das Ablaufebenensystem ein anderer Programmbaustein ausgewählt und gestartet bzw. weiter ausgeführt:

- Wenn der Programmzyklus-OB beendet wird, wird dieser wieder neu gestartet.
- Wenn ein OB beendet wird, der einen anderen Programmbaustein unterbrochen hat (z. B. einen Alarm-OB), so wird der unterbrochene Programmbaustein (z. B. Programmzyklus-OB) weiter ausgeführt.

Hinweis

Anweisung RET im Zusammenhang mit JMP und JMPN

Wenn in einem Netzwerk bereits die Sprunganweisung "JMP: Springen bei VKE = 1" oder "JMPN: Springen bei VKE = 0" verwendet wird, dann darf die Anweisung "RET: Zurück springen" nicht verwendet werden. In einem Netzwerk darf nur eine springende Spule verwendet werden.

Einstellmöglichkeiten des Rückgabewerts (Operand)

Der Rückgabewert der Anweisung kann die folgenden Werte annehmen:

- Ret (RLO, entspricht dem Verknüpfungsergebnis VKE. Es wird der Signalzustand "1" an den Freigabeausgang ENO aufrufenden Programmbaustein geliefert, da die Anweisung RET als bedingte Anweisung nur dann ausgeführt wird, wenn die Bedingung TRUE ist.)
- Ret True bzw. Ret False (Der jeweilige Wert der Konstanten, TRUE bzw. FALSE, wird an den aufrufenden Programmbaustein geliefert.)
- Ret Value (Der Wert der Booleschen Variablen <Operand> wird an den aufrufenden Programmbaustein geliefert.)

Um den Rückgabewert der Anweisung einzustellen, klicken Sie auf das kleine gelbe Dreieck der Anweisung und wählen Sie in der Klappliste den gewünschten Wert aus.

Die folgende Tabelle zeigt den Status des aufrufenden Programmbausteins, wenn die Anweisung in einem Netzwerk innerhalb des aufgerufenen Programmbausteins programmiert ist:

VKE	Rückgabewert	ENO des aufrufenden Programmbausteins
1	RLO	1
	TRUE	1
	FALSE	0
	<Operand> Boolesche Variable mit den Speicherbereichsmöglichkeiten E, A, M, D, L, T und Z	<Operand>
0	RLO	Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Programmbausteins fortgesetzt.
	TRUE	
	FALSE	
	<Operand>	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

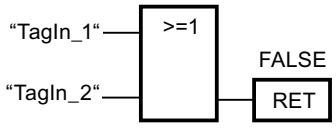

Wenn einer der Operanden "TagIn_1" oder "TagIn_2" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Programmbearbeitung im aufgerufenen Programmbaustein wird beendet und im aufrufenden Programmbaustein fortgesetzt. Der Freigabeausgang ENO des aufrufenden Programmbausteins wird auf den Signalzustand "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel für den EN-/ENO-Mechanismus in FUP (Seite 159)

Grundlagen zu FUP (Seite 7493)

Laufzeitsteuerung

ENDIS_PW: Passwort-Legitimierung einschränken und freigeben

Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU projektierte Passwörter legitimiert sind oder nicht. Dadurch können Sie legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "0" hat, wird der aktuell eingestellte Zustand an den Ausgangsparametern angezeigt. Falls Änderungen an den Eingangsparametern vorgenommen wurden, werden diese nicht an die Ausgangsparameter übergeben.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F_PWD, FULL_PWD, R_PWD, HMI_PWD) übernommen:

- Wenn der Signalzustand "0" vorliegt, ist die Legitimation per Passwort nicht erlaubt.
- Wenn der Signalzustand "1" vorliegt, ist das Passwort verwendbar.

Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer das Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Nicht konfigurierte Passwörter müssen am Eingang den Signalzustand TRUE haben und liefern den Signalzustand TRUE am Ausgang. Das Failsafe Passwort ist nur für eine F-CPU parametrierbar und muss daher in einer Standard CPU immer mit dem Signalzustand TRUE verschaltet werden. Wenn die Anweisung einen Fehler zurückliefert, dann bleibt der Aufruf ohne Auswirkung, d. h. die vorherige Sperre ist unverändert wirksam.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.

- Das Stecken einer leeren Speicherkarte (Übertragungskarte oder Programmkarte) in eine S7-1200-CPU.
- Der Übergang von NETZ AUS - NETZ EIN deaktiviert in der S7-1200-CPU den Schutz. Die Anweisung "Passwort-Legitimierung einschränken und freigeben" muss im Programm erneut (z. B. im Anlauf-OB) aufgerufen werden.

Hinweis

Wenn das HMI Passwort nicht freigegeben ist, dann sperrt die Anweisung "Passwort-Legitimierung einschränken und freigeben" den Zugriff von HMI-Systemen.

Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

Ungewolltes Aussperren bei einer S7-1500-CPU verhindern

Die Einstellungen können im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Ungewolltes Aussperren bei einer S7-1200-CPU verhindern

Da es bei einer S7-1200-CPU keinen Betriebsartenschalter gibt, wird der Schutz bei NETZ AUS - NETZ EIN deaktiviert. Damit ist es möglich und empfehlenswert, ein ungewolltes Aussperren mithilfe bestimmter Programmsequenzen innerhalb Ihres Programms zu verhindern.

Programmieren Sie dazu eine Zeitsteuerung entweder über einen Weckalarm-OB oder einen Timer im Main OB (OB 1). Dadurch haben Sie die Möglichkeit, nach einem Übergang von NETZ AUS - NETZ EIN und der damit verbundenen Deaktivierung des Schutzes, die Anweisung "Passwort-Legitimierung einschränken und freigeben" in dem entsprechenden OB (z. B. OB 1 oder OB 35) zeitnah wieder aufzurufen. Um das Zeitfenster, in dem die Anweisung nicht aktiv ist und damit keine Einschränkungen bei der Passwort-Legitimierung bestehen, möglichst klein zu halten, rufen Sie die Anweisung im Anlauf-OB (OB 100) auf. Diese Vorgehensweise bietet Ihnen den größtmöglichen Schutz vor unbefugtem Zugriff.

Wenn es zu einer ungewollten Aussperrung gekommen ist, dann können Sie den Aufruf im Anlauf-OB überspringen (z. B. durch die Abfrage eines Eingangsparameters) und haben die eingestellte Zeit (z. B. 10 Sekunden bis 1 Minute), um eine Verbindung zur CPU aufzubauen, bevor die Sperre wieder aktiv ist.

Wenn Sie in Ihrem Programmcode keinen Timer vorgesehen haben und es zu einer Aussperrung kam, dann stecken Sie eine leere Übertragungskarte oder eine leere Programmkarte in die CPU. Die leere Übertragungskarte oder Programmkarte löscht den

internen Ladespeicher der CPU. Danach müssen Sie das Anwenderprogramm aus STEP 7 erneut in die CPU laden.

Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte oder Programmkarte. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

⚠ WARNUNG

Leere Übertragungskarte stecken

Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Nach dem Ziehen der Übertragungskarte ist der Inhalt der Übertragungskarte im internen Ladespeicher. Achten Sie an dieser Stelle darauf, dass die Karte kein Programm enthält.

⚠ WARNUNG

Leere Programmkarte stecken

Wenn Sie eine Programmkarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Achten Sie darauf, dass die Programmkarte leer ist. Der interne Ladespeicher wird auf die leere Programmkarte kopiert. Nach dem Ziehen der ehemals leeren Programmkarte ist der interne Ladespeicher leer.

Sie müssen die Übertragungskarte oder Programmkarte ziehen, bevor Sie die CPU in RUN versetzen.

Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

Aktion	Passwortschutz durch die Anweisung
Grundzustand nach • Betriebsartenschalter auf STOP • Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control) • Zurücksetzen auf Werkseinstellung	Nicht aktiviert (Keine Einschränkungen)
Grundzustand nach NETZ-EIN	• S7-1200-CPU: Die Sperre ist deaktiviert und die Anweisung muss im Programm erneut aufgerufen werden (z. B. im Anlauf-OB). • S7-1500-CPU: Aktiviert (wenn vor NETZ AUS eine Sperre aktiviert war). Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent.
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT	Aktiviert Die Passwörter dürfen weiterhin nicht verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.
F_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff inklusive Failsafe • F_PWD = "0": Passwort nicht erlauben • F_PWD = "1": Passwort erlauben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FULL_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff • FULL_PWD = "0": Passwort nicht erlauben • FULL_PWD = "1": Passwort erlauben
R_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lesezugriff • R_PWD = "0": Passwort nicht erlauben • R_PWD = "1": Passwort erlauben
HMI_PWD	Input	BOOL	E, A, M, D, L oder Konstante	HMI-Zugriff • HMI_PWD = "0": Passwort nicht erlauben • HMI_PWD = "1": Passwort erlauben
F_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff inklusive Failsafe • F_PWD_ON = "0": Passwort nicht erlaubt • F_PWD_ON = "1": Passwort erlaubt
FULL_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff • FULL_PWD_ON = "0": Passwort nicht erlaubt • FULL_PWD_ON = "1": Passwort erlaubt
R_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lesezugriff • R_PWD_ON = "0": Passwort nicht erlaubt • R_PWD_ON = "1": Passwort erlaubt
HMI_PWD_ON	Output	BOOL	E, A, M, D, L	Status HMI-Zugriff • HMI_PWD_ON = "0": Passwort nicht erlaubt • HMI_PWD_ON = "1": Passwort erlaubt
RET_VAL	Output	WORD	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8090	Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt
80D0	Das Passwort für Failsafe ist nicht konfiguriert. Bei Standard-CPUs muss der Signalzustand TRUE sein.
80D1	Der Lese-/Schreibzugriff ist nicht konfiguriert
80D2	Der Lesezugriff ist nicht konfiguriert
80D3	Der HMI-Zugriff ist nicht konfiguriert
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

PC-Systeme

SHUT_DOWN: Zielsystem herunterfahren

Beschreibung

Mit der Anweisung “SHUT_DOWN: Zielsystem herunterfahren” fahren Sie das PC-basierte Automatisierungssystem herunter, starten den S7-Software Controller CPU 150xS oder Windows auf dem PC-basierte Automatisierungssystem neu.

Die Anweisung finden Sie in der Task Card "Anweisungen" unter Einfache Anweisungen > Programmsteuerung > Laufzeitsteuerung.

Ein Neustart ist z. B. in den folgenden Situationen sinnvoll:

- Eine Industrie-USV (unterbrechungsfreie Stromversorgung) meldet über einen digitalen Eingang einen Spannungsausfall.
- Windows reagiert nicht mehr oder liefert einen „Blue Screen“.
- Im Anwenderprogramm werden zu viele Fehler-OBs aufgerufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zielsystem herunter fahren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	MODE = 1: CPU 150xS und Windows herunter fahren Die CPU geht in STOP und speichert die remanenten Daten. Anschließend werden CPU und Windows herunter gefahren. Das System muss manuell gestartet werden. MODE = 2: CPU 150xS neu starten Die CPU geht in STOP und speichert die remanenten Daten. Anschließend wird die CPU heruntergefahren und neu gestartet. MODE = 3: Windows neu starten. Die CPU bleibt in RUN. Windows wird neu gestartet (ab TIA Portal V14 ist MODE 3 nur für Abwärtskompatibilität gelassen. Es wird empfohlen, MODE 4 oder MODE 5 zu verwenden). MODE = 4: Windows wird korrekt heruntergefahren und neu gestartet. Die CPU bleibt im RUN. MODE = 5: Windows neu starten. (Vergleichbar mit MODE 3; Ausnahme: MODE 5 ist nur dann zu verwenden, wenn Windows abgestürzt ist).
COMMENT	Input	STRING	E, A, M, D, L	Bei Mode = 1, 3 und 4 können Sie eine Begründung für den Neustart mitgeben. Die Begründung wird im Windows Event Log ausgegeben.
Ret_Val	Return	WORD	A, M, D, L	Ret_Val = 0: kein Fehler Ret_Val = 8090: Der übergebene Wert an MODE wird nicht unterstützt. Ret_Val = 8091: Windows reagiert nicht auf den Aufruf der Anweisung Shut_Down (gültig nur für Mode 3 und 4). Ret_Val = 8092: Bei Auftreten dieses Fehlers, wenden Sie sich bitte an den SIMATIC Customer Support (gültig nur für Mode 3 und 4).

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Aufruf der Anweisung

Es gelten die folgenden Aufrufbedingungen:

- Für eine CPU der Baureihe S7-1200 gilt:
  In Firmware Versionen < 2.2 können Sie die Anweisung nur in einem Zyklus-Organisationsbaustein der Priorität 1 aufrufen. Das entspricht der niedrigsten Priorität aller Organisationsbausteine. Wenn Sie die Anweisung in einem Organisationsbaustein mit höherer Priorität aufrufen, dann wird sie nicht ausgeführt und das Ergebnis (Bie-Bit, Freigabeausgang ENO) ist immer "0".
  In Firmware Versionen >= 2.2 können Sie die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.
- Für eine CPU der Baureihe S7-1500 gilt:
  Sie können die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOPP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang vom RUN in STOPP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOPP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

GET_ERROR: Fehler lokal abfragen

Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Programmbausteins ab. Dabei handelt es sich in der Regel um Programmier- oder Zugriffsfehler. Wenn das System während der Programmbausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler, innerhalb der Ausführung des Programmbausteins seit der letzten Ausführung der Anweisung, eine ausführliche Information im Operanden am Ausgang ERROR gespeichert.

Am Ausgang ERROR können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Treten im Programmbaustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehlerinformation zum nächsten aufgetretenen Fehler von der Anweisung ausgegeben.

Hinweis

Ausgang ERROR

Der Ausgang ERROR wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.
- Fragen Sie den Freigabeausgang ENO ab.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung nicht beeinflusst.

Hinweis

Aktivierung der lokalen Fehlerbehandlung

Sobald Sie die Anweisung in den Programmcode eines Programmbausteins einfügen, wird die lokale Fehlerbehandlung aktiviert und voreingestellte Systemreaktionen werden beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ERROR	Output	ErrorStruct	D, L	Fehlerinformationen

Datentyp "ErrorStruct"

Der Datentyp "ErrorStruct" kann in einen globalen Datenbaustein oder in die Bausteinschnittstelle eingefügt werden. Sie können den Datentyp auch mehrfach einfügen, wenn Sie jedes Mal einen anderen Namen für die Datenstruktur vergeben. Die Datenstruktur und der Name einzelner Strukturelemente sind nicht änderbar. Wenn Sie die Fehlerinformation in einem globalen Datenbaustein speichern, kann sie auch von anderen Programmbausteinen ausgelesen werden.

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

Strukturkomponente		Datentyp	Beschreibung
ERROR_ID		WORD	Fehler-ID
FLAGS		BYTE	Zeigt, ob der Fehler während eines Programmbausteinaufrufs aufgetreten ist. 16#01: Fehler während eines Programmbausteinaufrufs 16#00: Kein Fehler während eines Programmbausteinaufrufs
REACTION		BYTE	Voreingestellte Reaktion: 0: Ignorieren (Schreibfehler) 1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler) 2: Anweisung überspringen (Systemfehler)
CODE_ADDRESS		CREF	Informationen zur Adresse und Art des Programmbausteins
	BLOCK_TYPE	BYTE	Programmbausteintyp, in dem der Fehler aufgetreten ist: 1: Organisationsbaustein (OB) 2: Funktion (FC) 3: Funktionsbaustein (FB)

Strukturkomponente		Datentyp	Beschreibung
	CB_NUMBER	UINT	Nummer des Codebausteins
	OFFSET	UDINT	Verweis auf den internen Speicher
MODE		BYTE	Informationen zur Adresse eines Operanden
OPERAND_NUMBER		UINT	Operandennummer des Maschinenbefehls
POINTER_NUMBER_LOCATION		UINT	(A) Interner Zeiger
SLOT_NUMBER_SCOPE		UINT	(B) Ablagebereich im internen Speicher
DATA_ADDRESS		NREF	Informationen zur Adresse eines Operanden
	AREA	BYTE	(C) Speicherbereich: L: 16#40...16#7F, 16#86, 16#87, 16#8E, 16#8F, 16#C0...16#FF E: 16#81 A: 16#82 M: 16#83 DB: 16#40, 16#84, 16#85, 16#8A, 16#8B PE: 16#01 PA: 16#02 Technologische Objekte: 16#04
	DB_NUMBER	UINT	(D) Nummer des Datenbausteins
	OFFSET	UDINT	(E) Relative Adresse des Operanden

Strukturkomponente "ERROR_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR_ID" ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

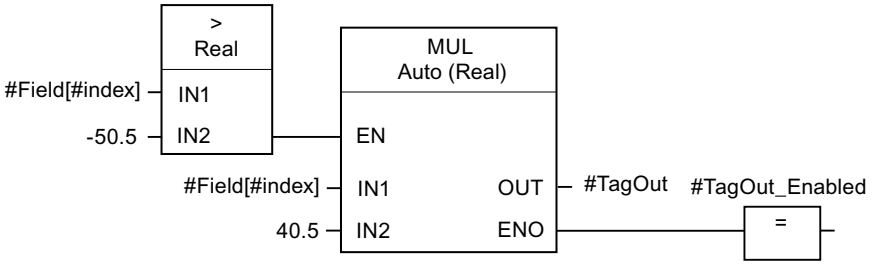

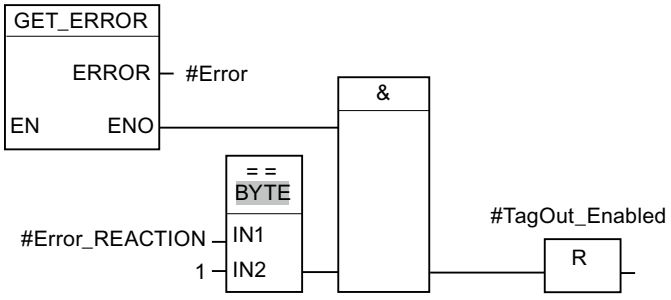

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Freigabeausgang ENO der Anweisung "Multiplizieren" und der Operand #TagOut_Enabled liefern trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0"

ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler lokal abfragen" nach der Anweisung "Multiplizieren" zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler lokal abfragen" liefert, wird mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Wenn die Strukturkomponente #Error.REACTION den Wert "1" hat, dann handelt es sich um einen Lese-/Zugriffsfehler und der Ausgang #TagOut_Enabled wird wieder zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

GET_ERR_ID: Fehler-ID lokal abfragen

Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Dabei handelt es sich in der Regel um Zugriffsfehler. Wenn das System während der Bausteinbearbeitung Fehler innerhalb der Ausführung des Bausteins seit der letzten Ausführung der Anweisung meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang ID gespeichert.

Am Ausgang ID können nur Operanden vom Datentyp WORD angegeben werden. Treten im Baustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehler-ID des nächsten aufgetretenen Fehlers von der Anweisung ausgegeben.

Hinweis

Der Ausgang ID wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.
- Fragen Sie den Freigabeausgang ENO ab.

Der Freigabeausgang ENO der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Ein Beispiel, wie Sie die Anweisung auch in Kombination mit weiteren Fehlerbehandlungsmöglichkeiten realisieren können, finden Sie unter "Siehe auch".

Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
ID	Output	WORD	E, A, M, D, L	Fehler-ID

Parameter ID

Die folgende Tabelle zeigt die Werte, die am Parameter ID ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

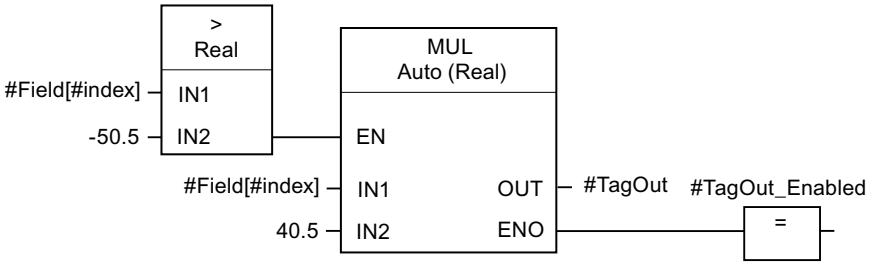

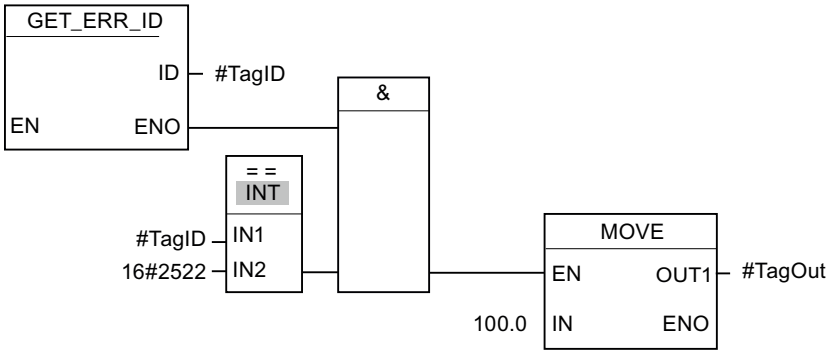

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Freigabeausgang ENO der Anweisung "Multiplizieren" und der Operand #TagOut_Enabled liefern trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler-ID lokal abfragen" nach der Anweisung "Multiplizieren" zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler-ID lokal abfragen" liefert, wird mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Wenn der Operand #TagID die ID 2522 liefert, dann handelt es sich um einen Lese-/Zugriffsfehler und der Wert "100.0" wird in den Ausgang #TagOut geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

INIT_RD: Alle remanenten Daten zurücksetzen

Beschreibung

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B5	Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

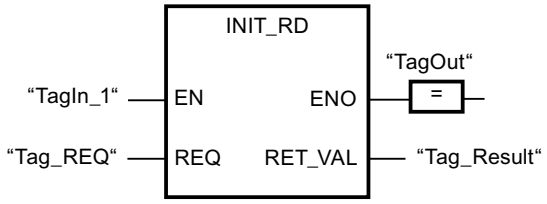

Wenn die Operanden "TagIn_1" und "Tag_REQ" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu +32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Beispiel zum Einfluss der geplanten Verzögerungszeit

Im folgenden Beispiel sehen Sie den Einfluss der Verzögerungszeit der Anweisung "WAIT" in verschiedenen Szenarien.

Das folgende Bild stellt die Szenarien schematisch dar:

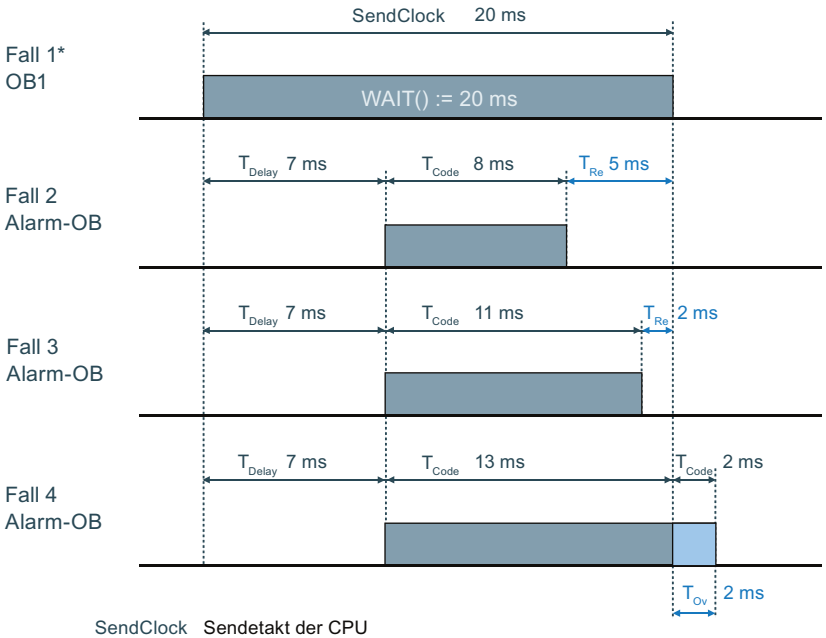

SendClock Sendetakt der CPU

T_{Delay} Verzögerungszeit des Alarm-OBs

T_{Code} Zeit der Programmausführung

T_{Re} Verbleibende Zeit

T_{Ov} Zeit der Überlänge, um welche sich der Sendetakt erhöht

* Der OB1 mit der Anweisung "WAIT" ist ebenso Teil von Fall 2, 3 und 4 und muss mit berücksichtigt werden.

Verbleibende Zeit = Die Spanne vom Ende der geplanten Verzögerungszeit (via "WAIT") bis zur Beendigung des Alarm-OBs

Zeit der Überlänge = Die Spanne von der Beendigung des Alarm-OBs bis zum Ende der geplanten Verzögerungszeit (via "WAIT").

Fall 1:

Die Anweisung "WAIT" wird in einem OB1 aufgerufen. Die Anweisung "WAIT" kann von höher prioren OBs oder höher prioren Prozessen (z.B. System Threads) unterbrochen werden. Die Verzögerungszeit der Anweisung "WAIT" ist aber weder verändert noch aufgeschoben.

Fall 2 und Fall 3:

Die Verarbeitung des Programms im OB1 wird nach einer Zeitverzögerung von 20 ms fortgesetzt. Diese Zeitverzögerung wird durch den Aufruf der Anweisung "WAIT" im OB1 berechnet (siehe OB1 mit WAIT). Innerhalb dieser 20 ms kann ein Alarm-OB seinen eigenen Programmcode ausführen. Der Sendetakt der CPU wird nicht verändert.

Fall 4:

Die Verarbeitung des Programms im OB1 wird nach Beendingung des höher prioren Prozesses fortgesetzt. Die 20 ms Verzögerungszeit im OB1 sind verstrichen, aber der höher priore Prozess muss noch beendet werden. Der Sendetakt der CPU wird erhöht.

Hinweis

Ausführungsreihenfolge von System- oder Kommunikationsprozessen (System Threads)

Die System Threads verwenden meistens die Priorität "15". Ebenso gibt es System Threads mit einer höheren Priorität als "26", doch diese Prozesse haben eine geringere CPU-Auslastung. System Threads werden im Bild nicht dargestellt.

Laufzeitmessung des OB1 mit Hilfe der Anweisung "RT_INFO":

Fall 2: 20 ms - 8 ms - System Threads = <12 ms. Sendetakt: ~20 ms.

Fall 3: 20 ms - 11 ms - System Threads - <9 ms. Sendetakt: ~20 ms.

Fall 4: 20 ms - 15 ms - System Threads - <7 ms. Sendetakt: ~22 ms.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

RUNTIME: Laufzeitmessung

Beschreibung

Mit der Anweisung "Laufzeitmessung" messen Sie die Laufzeit des gesamten Programms, einzelner Bausteine oder Befehlssequenzen.

Wenn Sie die Laufzeit Ihres gesamten Programms messen wollen, dann rufen Sie die Anweisung "Laufzeitmessung" im OB1 auf. Der Startpunkt der Laufzeitmessung wird mit dem

ersten Aufruf gesetzt und der Ausgang RET_VAL liefert nach dem zweiten Aufruf die Laufzeit des Programms. Die gemessene Laufzeit enthält alle möglichen Vorgänge der CPU, die innerhalb des Programmablaufs statt gefunden haben, wie z. B. Unterbrechungen durch höher priore Ereignisse oder Kommunikation. Die Anweisung "Laufzeitmessung" liest einen internen Zähler der CPU aus und schreibt dessen Wert in den Durchgangsparameter MEM. Die Anweisung berechnet die aktuelle Laufzeit des Programms gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Wenn Sie die Laufzeit einzelner Bausteine oder einzelner Befehlssequenzen messen wollen, dann benötigen Sie drei separate Netzwerke. Rufen Sie die Anweisung "Laufzeitmessung" in einem einzelnen Netzwerk innerhalb Ihres Programms auf. Mit diesem ersten Aufruf der Anweisung setzen Sie den Startpunkt der Laufzeitmessung. Anschließend rufen Sie im nächsten Netzwerk den gewünschten Programmbaustein oder die Befehlssequenz auf. Rufen Sie dann in einem weiteren Netzwerk die Anweisung "Laufzeitmessung" zum zweiten Mal auf und weisen Sie dem Durchgangsparameter MEM den gleichen Speicher wie beim ersten Aufruf der Anweisung zu. Die Anweisung "Laufzeitmessung" im dritten Netzwerk liest einen internen Zähler der CPU aus und berechnet die aktuelle Laufzeit des Programmbausteins oder der Befehlssequenz gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Die Anweisung "Laufzeitmessung" verwendet einen internen hochfrequenten Zähler um die Zeit zu berechnen. Wenn der Zähler überläuft, gibt die Anweisung Werte <= 0.0 zurück. Dies kann für S7-1200-CPUs mit Firmware Stand <V4.2 bis zu einmal pro Minute vorkommen. Diese Runtimewerte sind zu ignorieren.

Hinweis

Da beim optimierten Übersetzen des Programms die Reihenfolge der Anweisungen innerhalb einer Befehlssequenz geändert wird, kann die Laufzeit einer Befehlssequenz nicht genau bestimmt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Laufzeitmessung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MEM	InOut	LREAL	E, A, M, D, L	Der Inhalt ist nur für interne Zwecke vorgesehen.
RET_VAL	Output	LREAL	E, A, M, D, L	Liefert die gemessene Laufzeit in Sekunden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung anhand einer Berechnung der Laufzeit eines Programmbausteins:

Netzwerk 1:

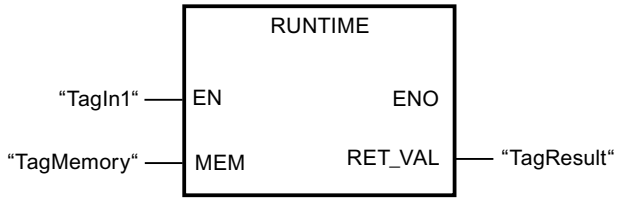

Netzwerk 2:

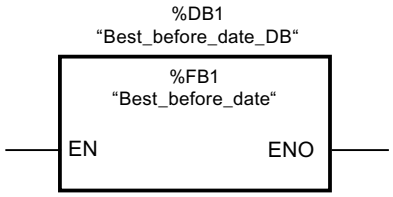

Netzwerk 3:

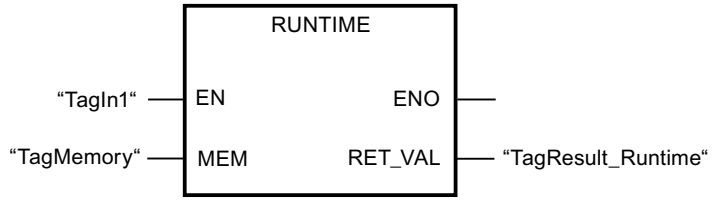

Wenn der Operand "TagIn1" im Netzwerk 1 den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Mit dem ersten Aufruf der Anweisung wird der Startpunkt für die Laufzeitmessung gesetzt und als Referenz für den zweiten Aufruf der Anweisung im Operanden "TagMemory" zwischengespeichert.

Im Netzwerk 2 wird der Programmbaustein FB1 "Best_before_date" aufgerufen.

Wenn der Programmbaustein FB1 abgearbeitet ist und der Operand "TagIn1" den Signalzustand "1" liefert, wird die Anweisung im Netzwerk 3 ausgeführt. Der zweite Aufruf der Anweisung berechnet die Laufzeit des Programmbausteins und schreibt das Ergebnis in den Ausgang RET_VAL.

Ein ausführlicheres Beispiel, wie Sie die Gesamtzykluszeit eines Programms messen können, finden Sie im Siemens Industry Online Support unter folgender Beitrags-ID: 87668055 (https://support.industry.siemens.com/cs/ww/de/view/87668055)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Wortverknüpfungen

AND: UND verknüpfen

Beschreibung

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch UND verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte
OUT	Output	Bitfolgen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

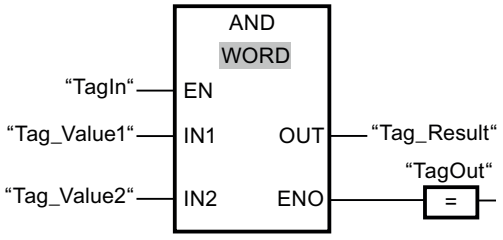

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

OR: ODER verknüpfen

Beschreibung

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte
OUT	Output	Bitfolgen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

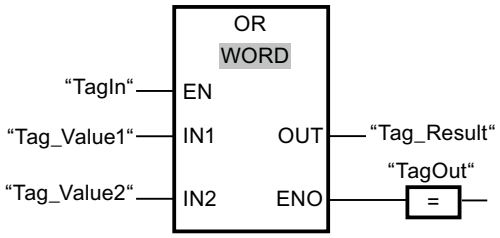

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch

ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

XOR: EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch EXKLUSIV ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN1	Input	Bitfolgen	E, A, M, D, L, P oder Konstant	E, A, M, D, L, P oder Konstant	Erster Wert der Verknüpfung
IN2	Input	Bitfolgen	E, A, M, D, L, P oder Konstant	E, A, M, D, L, P oder Konstant	Zweiter Wert der Verknüpfung

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
INn	Input	Bitfolgen	E, A, M, D, L, P oder Konstant	E, A, M, D, L, P oder Konstant	Optionale Eingangswerte
OUT	Output	Bitfolgen	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

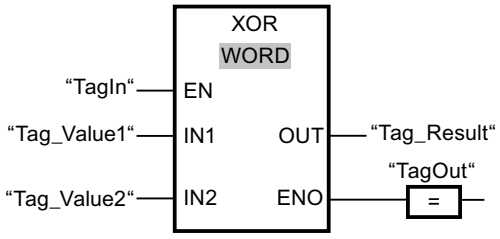

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

INVERT: Einerkomplement erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L, P	E, A, M, D, L, P	Einerkomplement des Wertes am Eingang IN

Aus der Klappliste "<???>" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

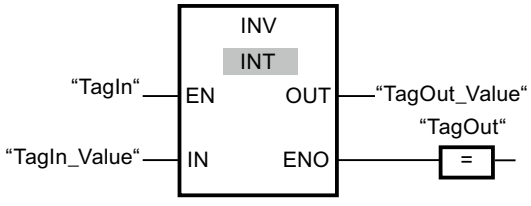

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	W#16#000F	W#16#7E
OUT	TagOut_Value	W#16#FFF0	W#16#81

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Einerkomplement erzeugen" ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn_Value" um und schreibt das Ergebnis in den Ausgang "TagOut_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	UINT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	Bitfolgen	E, A, M, D, L, P	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

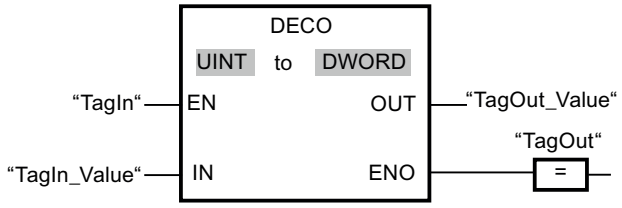

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

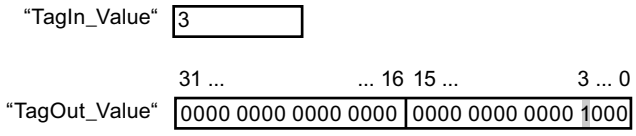

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn_Value" am Eingang ab und setzt das dritte Bit im Wert des Operanden "TagOut_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und gibt ihn am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
EN	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

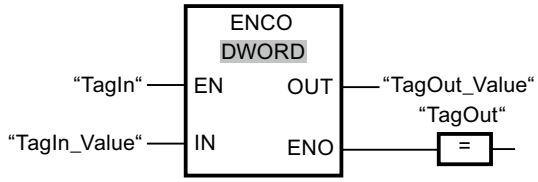

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

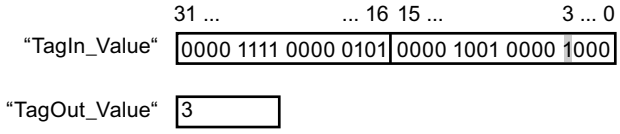

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung selektiert Bitposition "3" als niederwertigstes gesetztes Bit am Eingang "TagIn_Value" und schreibt den Wert "3" in die Variable am Ausgang "TagOut_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig vom Signalzustand am Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt den Signalzustand "1" am Freigabeeingang EN voraus und dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
G	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, LDT, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Eingangswert

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, LDT, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, LDT, CHAR, WCHAR	E, A, M, D, L, P	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

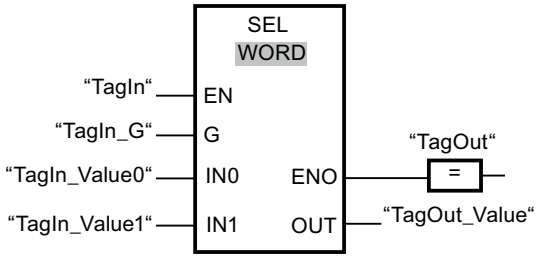

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn_G" wird der Wert am Eingang "TagIn_Value0" oder "TagIn_Value1" selektiert und in den Ausgang "TagOut_Value" kopiert.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MUX: Multiplexen

Beschreibung

Mit der Anweisung "Multiplexen" kopieren Sie den Inhalt eines ausgewählten Eingangs in den Ausgang OUT. Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN0 und IN1). Die Anzahl der auswählbaren Eingänge der Anweisungsbox ist erweiterbar. Sie können maximal 32 Eingänge deklarieren.

Die Eingänge werden in der Box automatisch nummeriert. Die Nummerierung fängt mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Eingang, dessen Inhalt in den Ausgang OUT kopiert wird. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Eingänge ist, wird der Inhalt des Parameters ELSE in den Ausgang OUT kopiert und dem Freigabeausgang ENO den Signalzustand "0" zugewiesen.

Die Anweisung "Multiplexen" kann nur ausgeführt werden, wenn die Variablen an allen Eingängen sowie am Ausgang OUT vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, für den nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Eingang am Parameter K liegt außerhalb der verfügbaren Eingänge. Diese Reaktion ist unabhängig davon, ob der Eingang ELSE verwendet wird oder nicht. Der Wert am Ausgang OUT bleibt unverändert.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplexen":

Parameter	Deklarieren	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklarieren	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Gibt den Eingang an, dessen Inhalt kopiert wird. • Wenn K = 0 => Parameter IN0 • Wenn K = 1 => Parameter IN1, etc.
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
INn	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Optionale Eingangswerte

Parameter	Deklarieren	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
ELSE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Gibt den Wert an, der bei K > n kopiert wird
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	E, A, M, D, L, P	Ausgang, in den der Wert kopiert wird

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

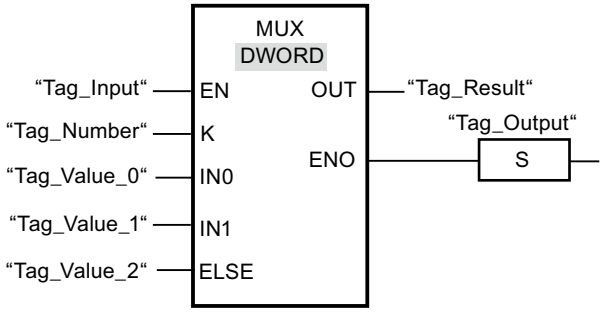

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
K	Tag_Number	1
IN0	Tag_ValueI_0	DW#16#00000000
IN1	Tag_Value_1	DW#16#003E4A7D
ELSE	Tag_Value_2	DW#16#FFFF0000
OUT	Tag_Result	DW#16#003E4A7D

Wenn der Operand "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Entsprechend dem Wert des Operanden "Tag_Number" wird der Wert am Eingang

"Tag_Value_1" kopiert und dem Operanden am Ausgang "Tag_Result" zugewiesen. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "Tag_Output" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

DEMUX: Demultiplexen

Beschreibung

Mit der Anweisung "Demultiplexen" kopieren Sie den Inhalt des Eingangs IN in einen ausgewählten Ausgang. Die Anweisungsbox enthält im Grundzustand min. 2 Ausgänge (OUT0 und OUT1). Die Anzahl der auswählbaren Ausgänge ist in der Anweisungsbox erweiterbar. Die Ausgänge werden in der Box automatisch nummeriert. Die Nummerierung fängt bei OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Ausgang, in welchen der Inhalt des Eingangs IN kopiert wird. Die anderen Ausgänge werden nicht verändert. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Ausgänge ist, wird der Inhalt des Eingangs IN in den Parameter ELSE kopiert und dem Freigabeausgang ENO der Signalzustand "0" zugewiesen.

Die Anweisung "Demultiplexen" kann nur ausgeführt werden, wenn die Variablen am Eingang IN sowie an allen Ausgängen vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, an welchem nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Demultiplexen":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. • Wenn K = 0 => Parameter OUT0 • Wenn K = 1 => Parameter OUT1, etc.
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	E, A, M, D, L, P oder Konstante	Eingangswert
OUT0	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	E, A, M, D, L, P	Erster Ausgang
OUT1	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	E, A, M, D, L, P	Zweiter Ausgang

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
OUTn	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	E, A, M, D, L, P	Optionale Ausgänge
ELSE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, DATE	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	E, A, M, D, L, P	Ausgang, in welchen der Eingangswert (IN) bei K > n kopiert wird.

Aus der Klappliste "<???>" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

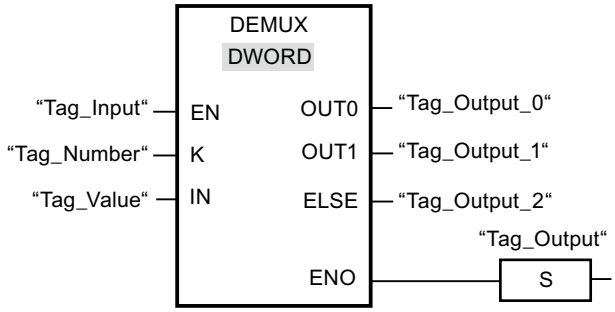

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Tabelle 4-3 Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

Parameter	Operand	Werte	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Tabelle 4-4 Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

Parameter	Operand	Werte	
OUT0	Tag_Output_0	unverändert	unverändert
OUT1	Tag_Output_1	DW#16#FFFFFFFF	unverändert
ELSE	Tag_Output_2	unverändert	DW#16#003E4A7D

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung "Demultiplexen" ausgeführt. Abhängig vom Wert des Operanden "Tag_Number" wird der Wert am Eingang IN in den entsprechenden Ausgang kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Eingang N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:

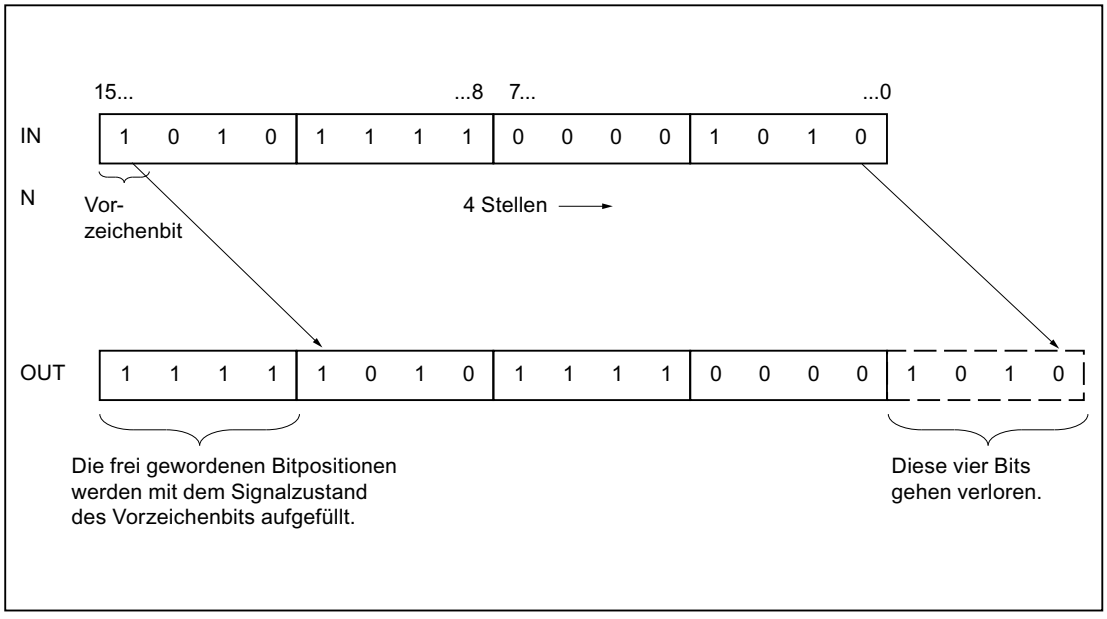

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der verschoben wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

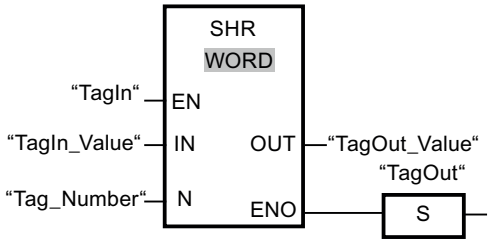

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Eingang N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

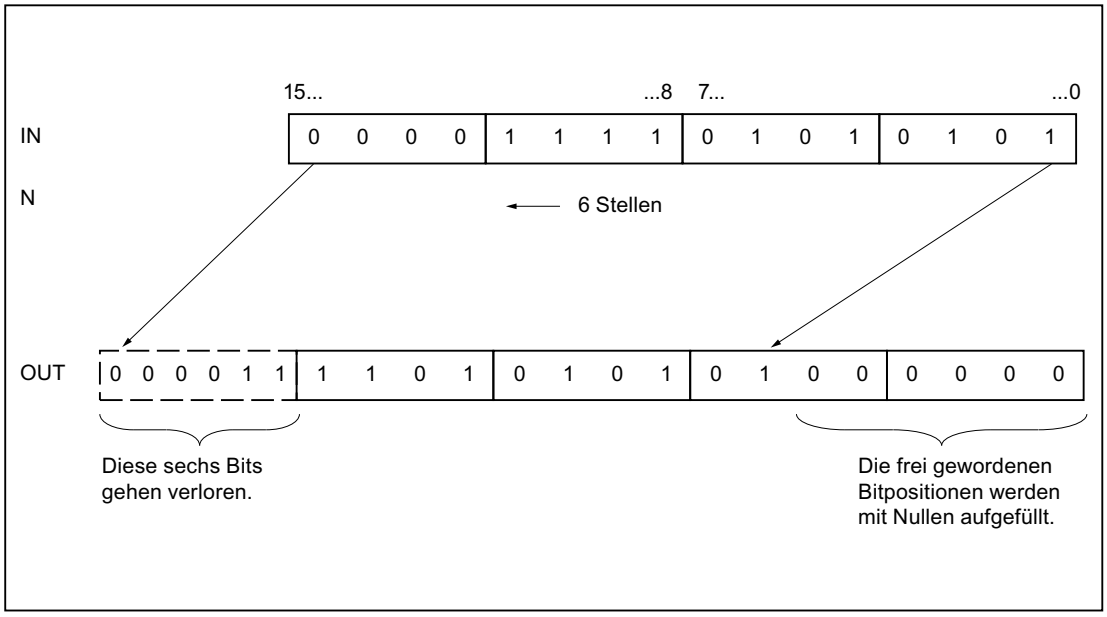

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der verschoben wird.
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird.
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

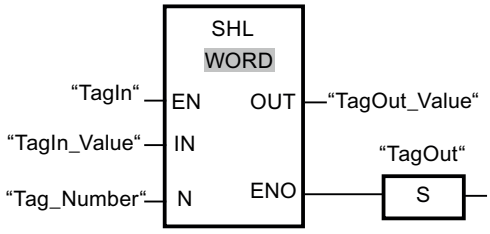

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen auf der linken Seite werden mit den hinausgeschobenen Bitstellen von der rechten Seite positionsgetreu aufgefüllt.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

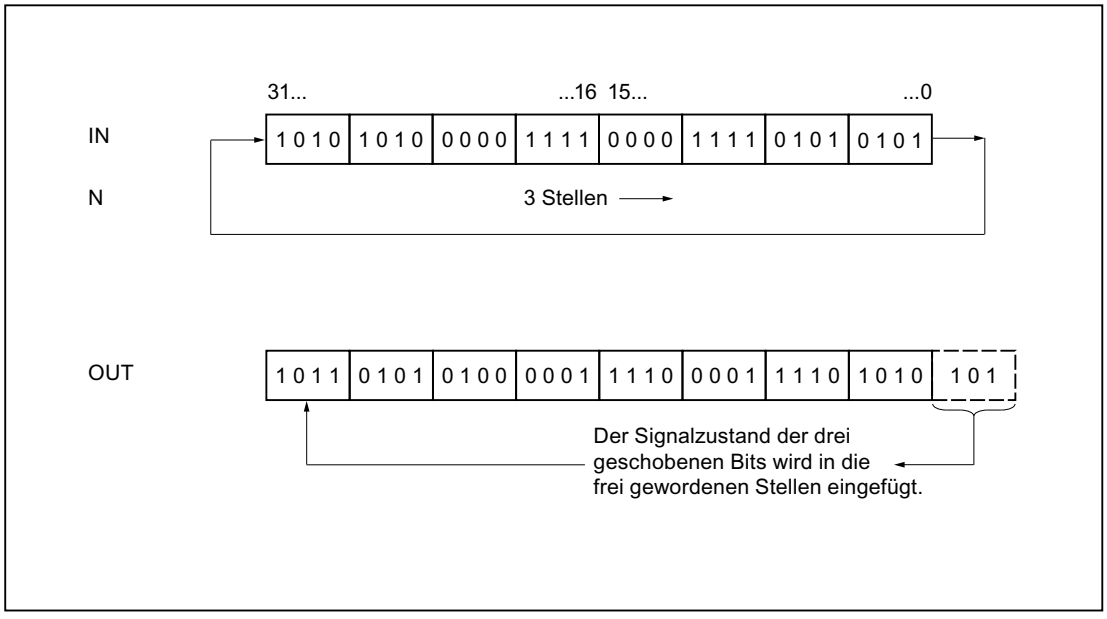

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der rotiert wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

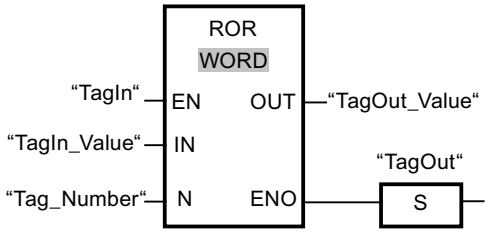

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0000 1111 1001 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1000 0111 1100

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen auf der rechten Seite werden mit den hinausgeschobenen Bitstellen von der linken Seite positionsgetreu aufgefüllt.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

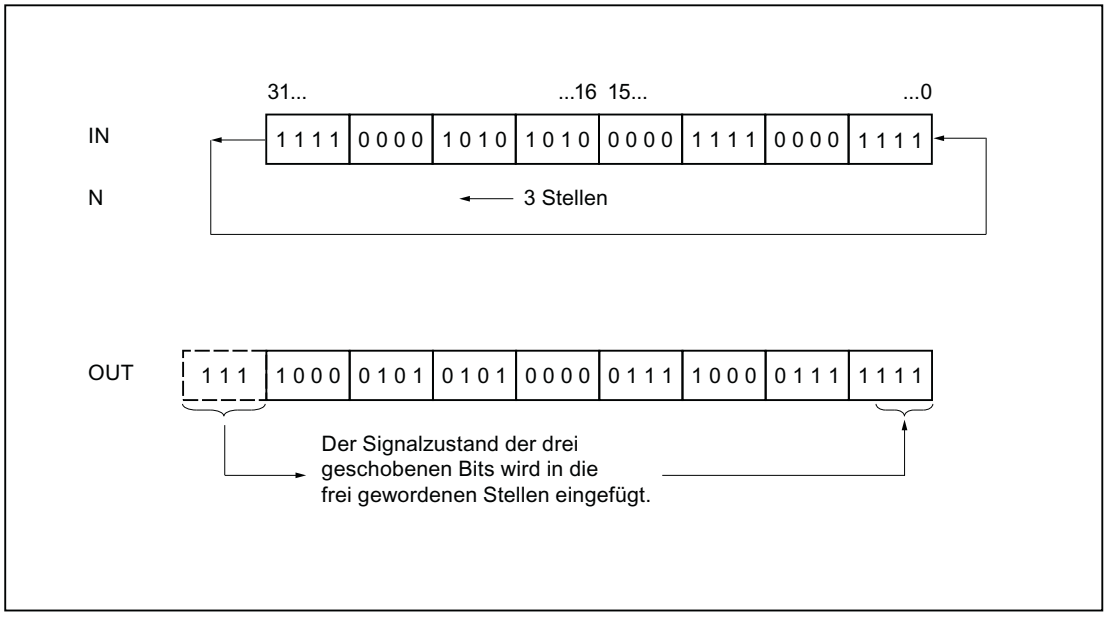

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

Parameter	Deklaration	Datentyp		Speicherbereich		Beschreibung
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	BOOL	E, A, M, D, L	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Wert, der rotiert wird.
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird.
OUT	Output	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

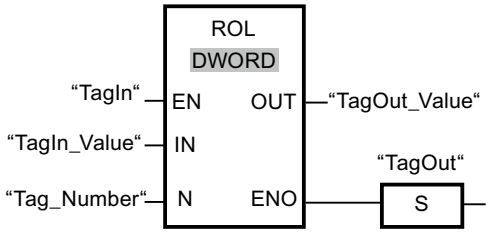

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1010 1000 1111 0110
N	Tag_Number	5
OUT	TagOut_Value	0001 1110 1101 0101

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

Legacy

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L oder Konstante	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Maximale Schrittnummer (z. B.: LST_STEP = 16#08; Es sind maximal 8 Schritte möglich.)
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L oder Konstante	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L oder Konstante	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P oder Konstante	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

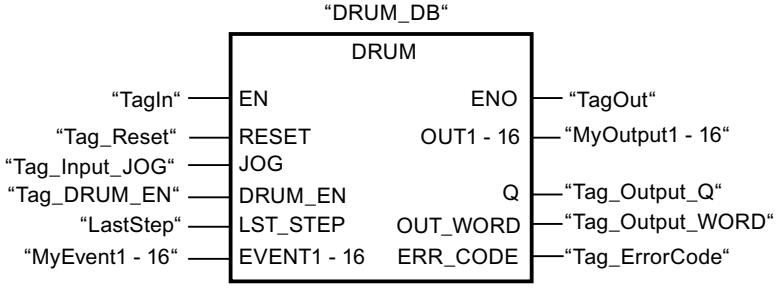

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08

Parameter	Operand	Adresse	Wert
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE

Parameter	Adresse	Wert
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE

Parameter	Operand	Adresse	Wert
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET (nur wenn ET < PT ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn ET < PT ist), OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS der Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB den Signalzustand "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CMD	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
CMD_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit zu CMD

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

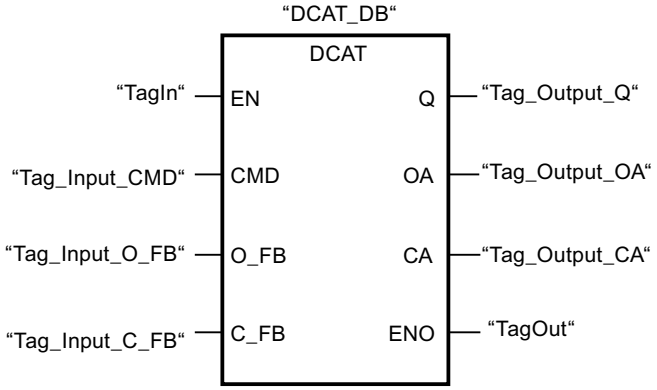

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
O_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Stoppen"

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
O_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Schließen"

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen

Eingangsparameter								Ausgangsparameter								
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt
Legende:																
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren															
PT	PT wird auf den gleichen Wert wie ET gesetzt															
X	Nicht anwendbar															
< PT	ET < PT															
>= PT	ET >= PT															
Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE																

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

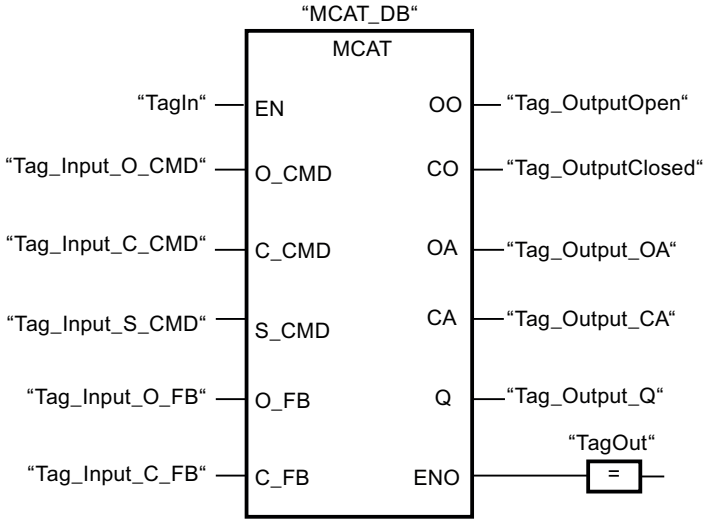

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Übereinstimmung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Übereinstimmung in den Parameter OUT_STEP angegeben. Wird keine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Übereinstimmung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P oder Konstante	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

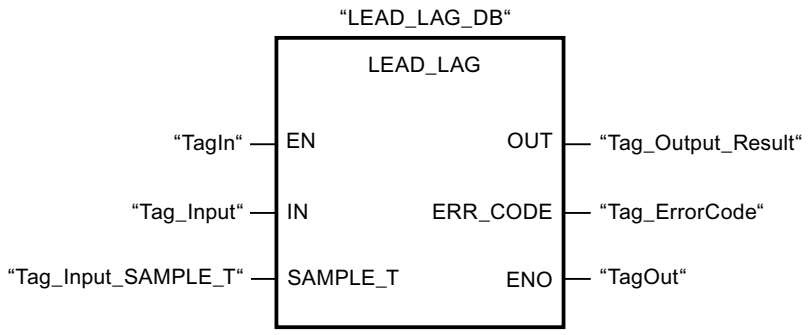

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	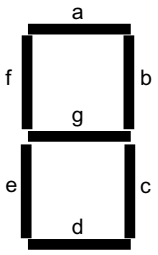
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

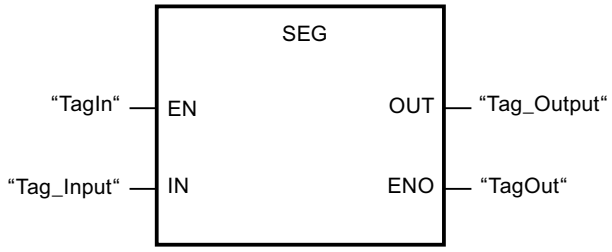

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

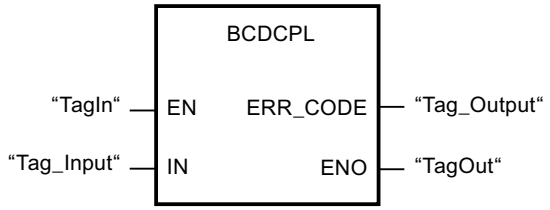

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bits gezählt werden.
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bits

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

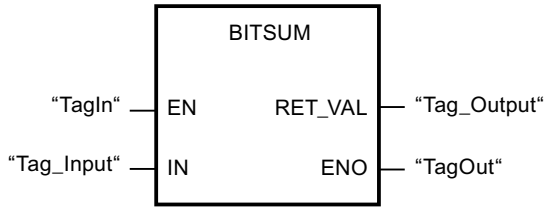

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu FUP (Seite 7493)

4.1.3.3 AWL

Bitverknüpfungen

R_TRIG: Positive Signalflanke erkennen

Beschreibung

Mit der Anweisung "Positive Signalflanke erkennen" können Sie eine Zustandsänderung von "0" auf "1" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "0" auf "1" erkennt, wird am Ausgang Q eine positive Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Positive Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL R_TRIG, "R_TRIG_DB"     // Die Anweisung wird aufgerufen.
CLK := "TagIn"               // Positive Signalflanke wird erfasst.
Q := "TagOut"                // Signalzustand "1" bei positiver Signalflanke.
```

In der Variablen "R_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3" eine Änderung im Signalzustand von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

F_TRIG: Negative Signalflanke erkennen

Beschreibung

Mit der Anweisung "Negative Signalflanke erkennen" können Sie eine Zustandsänderung von "1" auf "0" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "1" auf "0" erkennt, wird am Ausgang Q eine negative Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negative Signalflanke erkennen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L oder Konstante	Eingehendes Signal, dessen Flanke abgefragt wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL F_TRIG, "F_TRIG_DB"     // Die Anweisung wird aufgerufen.
CLK := "TagIn"               // Negative Signalflanke wird erfasst.
Q := "TagOut"                // Signalzustand "1" bei negativer Signalflanke.
```

In der Variablen "F_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn am Operand "TagIn" eine Änderung im Signalzustand von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Im Programmcode müssen Sie die Anweisung "Impuls erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME oder TP_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTP_TIMER)

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am

Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Impulsausgang
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Impuls erzeugen" nach dem Starten:

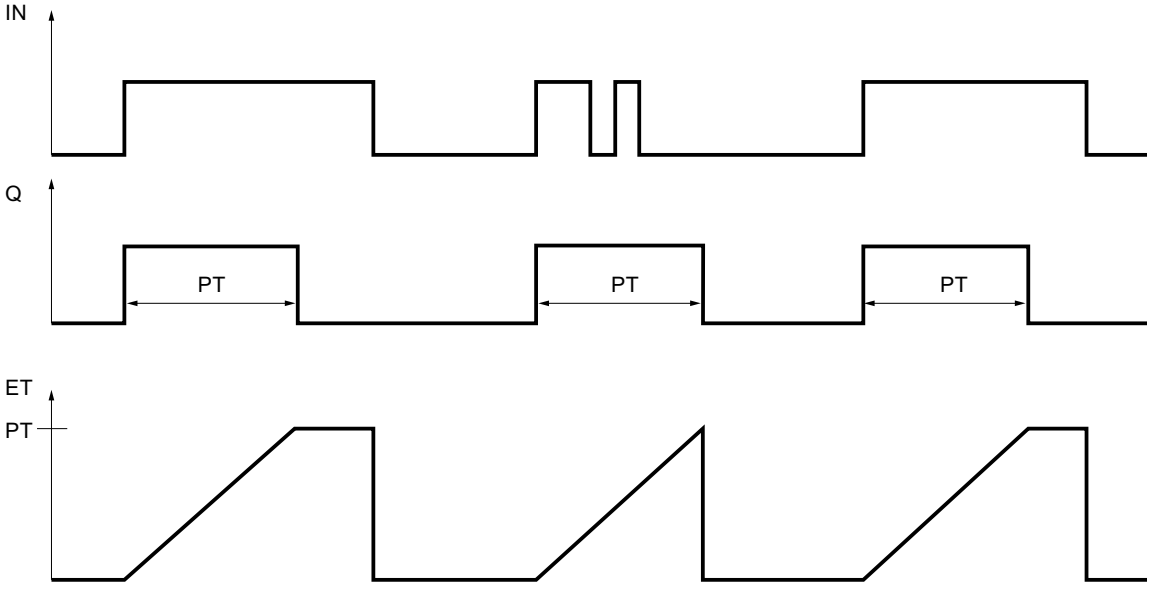

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL TP, "TP_DB"`	`// Die Anweisung wird aufgerufen. Der Datenbaustein "TP_DB" wird der Anweisung zugeordnet.` `// Wählen Sie den gewünschten Datentyp aus der Klapp-liste "???" aus.`

AWL	Erläuterung
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Zeitdauer des Impulses
Q := "Tag_Output"	// Der Operand wird für die Zeitdauer gesetzt, die durch den Operanden "Tag_PresetTIME" festgelegt wurde.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt so lange gesetzt, wie der Starteingang IN noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Parameter ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Im Programmcode müssen Sie die Anweisung "Einschaltverzögerung erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine

Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME oder TON_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTON_TIMER)

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Signalzustand, der um die Zeit PT verzögert wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Einschaltverzögerung erzeugen" nach dem Starten:

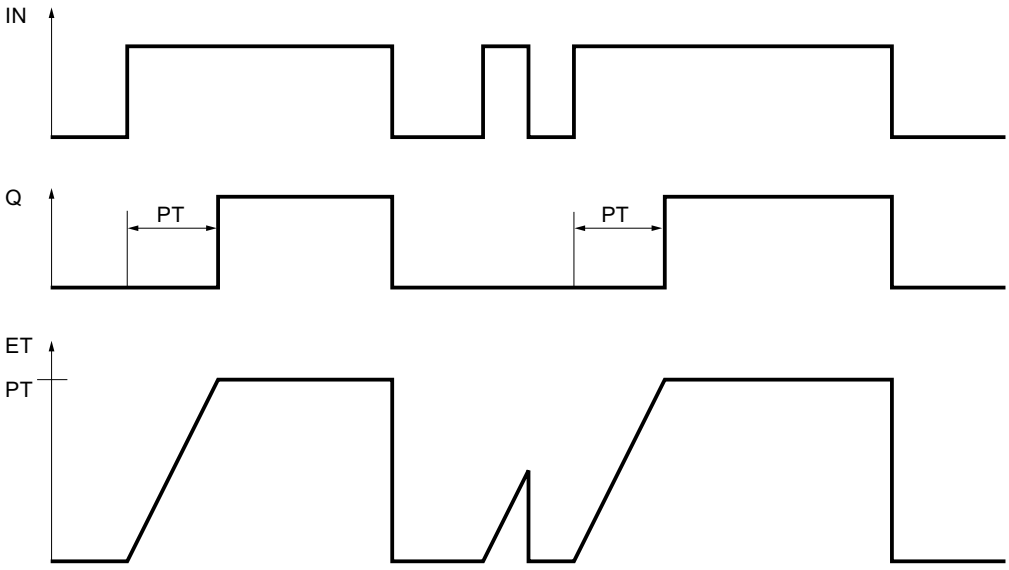

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL "TON", "TON_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TON_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Gibt die Zeitdauer an, um die die positive Signalflanke am Parameter IN verzögert wird.
Q := "Tag_Output"	// Der Operand wird gesetzt, wenn die durch die Variable "Tag_PresetTIME" vorgegebene Zeitdauer PT abgelaufen ist.
	// Der Parameter Q bleibt so lange gesetzt, wie die Variable "Tag_Start" noch "1" führt.
	// Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET solange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Im Programmcode müssen Sie die Anweisung "Ausschaltverzögerung erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME oder TOF_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTOF_TIMER)

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Signalzustand, der um die Zeit PT verzögert wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Ausschaltverzögerung erzeugen" nach dem Starten:

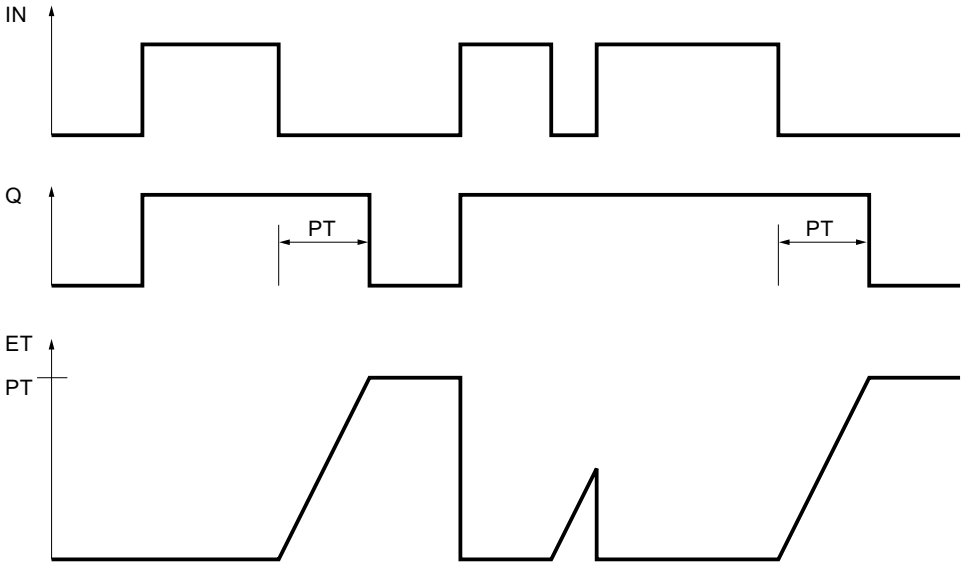

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TOF, "TOF_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TOF_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Gibt die Zeitdauer an, um die die negative Signalflanke am Parameter IN verzögert wird.
Q := "Tag_Output"	// Der Operand wird gesetzt, wenn die Anweisung durch eine positive Signalflanke am Parameter IN gestartet wird.
	// Bei einem Wechsel von "1" auf "0" am Parameter IN bleibt der Operand "Tag_Output" gesetzt, solange die durch die Variable "Tag_PresetTIME" vorgegebene Zeitdauer läuft.
	// Wenn die Zeit am Parameter PT abgelaufen ist, wird der Operand zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Eingang IN von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt (negative Signalflanke).

Der Eingang R setzt die Ausgänge ET und Q, unabhängig vom Signalzustand am Starteingang, zurück.

Im Programmcode müssen Sie die Anweisung "Zeit akkumulieren" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME oder TONR_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTONR_TIMER)

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Akkumulierte Zeit

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":

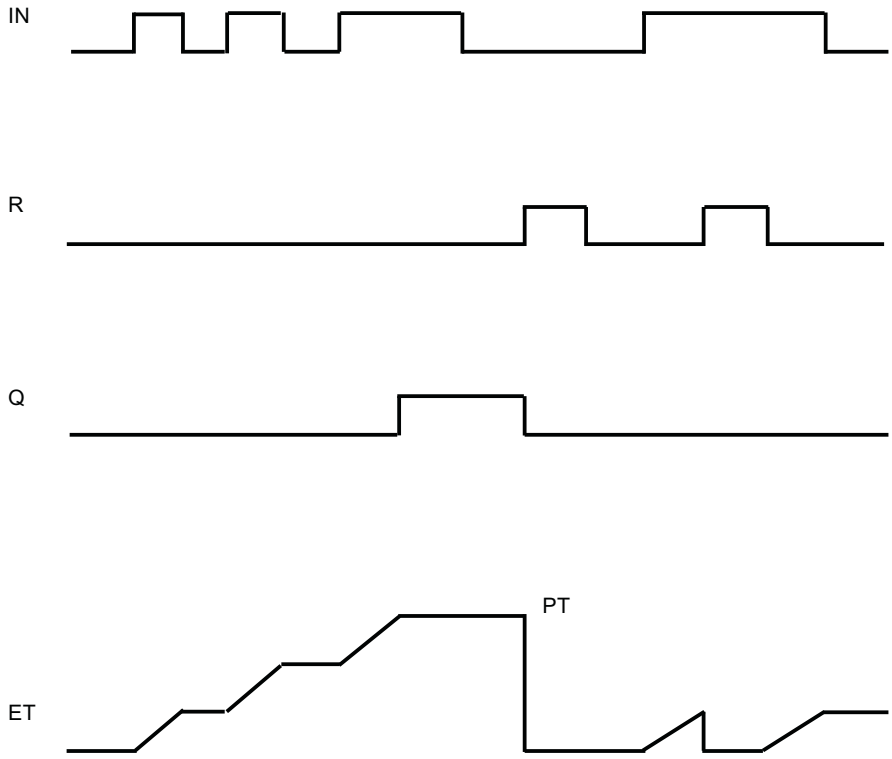

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TONR, "TONR_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TONR_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt und die Zeitdauer am Eingang PT gestartet.
	Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden.
R := "Tag_Reset"	// Der Eingang R setzt die Ausgänge ET und Q, unabhängig vom Signalzustand am Starteingang, zurück.
PT := "Tag_PresetTIME"	// Gibt die Zeitdauer an, wie lange die Zeitwerte akkumuliert werden.

AWL	Erläuterung
Q := "Tag_Output"	// Der Operand wird gesetzt, wenn die Zeitdauer PT erreicht ist. // Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt (negative Signalflanke).
ET := "Tag_ElapsedTIME"	// Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

RESET_TIMER: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die Strukturkomponenten der Zeit im angegebenen Datenbaustein werden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Am Parameter TIMER wird der Anweisung "Zeit rücksetzen" eine im Programm deklarierte IEC-Zeit zugewiesen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TIMER	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die zurückgesetzt wird

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TON, "TON_DB"	// Die Anweisung wird aufgerufen. Der Instanz-Datenbaustein "TON_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden "Tag_Start" wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PresetTIME" vorgegeben ist.
Q := "Tag_Output"	// Der Operand "Tag_Output" wird gesetzt, wenn die durch den Operanden "Tag_PresetTIME" vorgegebene Zeitdauer PT abgelaufen ist.
	// Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag_Start" noch den Signalzustand "1" führt.
	// Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert
U "Tag_Input_1"	// Wenn der Operand "Tag_Input_1" und
U "Tag_Input_2"	// der Operand "Tag_Input_2" den Signalzustand "1" liefern,
CALL RESET_TIMER	// wird die Anweisung "Zeit rücksetzen" aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
TIMER := "TON_DB"	// IEC-Zeit wird zurückgesetzt

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

PRESET_TIMER: Zeitdauer laden

Beschreibung

Mit der Anweisung "Zeitdauer laden" stellen Sie die Zeitdauer einer IEC-Zeit ein. Die Anweisung wird in jedem Zyklus ausgeführt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegeben IEC-Zeit.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC_TIMER-Struktur.

Hinweis

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	TIME, LTIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der die IEC-Zeit läuft
<IEC-Zeit>	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, deren Dauer eingestellt wird

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TON, "TON_DB"	// Die Anweisung wird aufgerufen. Der Instanz-Datenbaustein "TON_DB" wird der Anweisung zugeordnet. // Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.

AWL	Erläuterung
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden "Tag_Start" wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PresetTIME" vorgegeben ist.
Q := "Tag_Output"	// Der Operand "Tag_Output" wird gesetzt, wenn die durch den Operanden "Tag_PresetTIME" vorgegebene Zeitdauer PT abgelaufen ist.
	// Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag_Start" noch den Signalzustand "1" führt.
	// Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert
U "Tag_Input_1"	// Wenn der Operand "Tag_Input_1" und
U "Tag_Input_2"	// der Operand "Tag_Input_2" den Signalzustand "1" liefern,
CALL PRESET_TIMER	// wird die Anweisung "Zeitdauer laden" aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
PT := "Tag_PresetTIME_new"	// Die Anweisung schreibt die Zeitdauer "Tag_PresetTIME_new" in den Instanz-Datenbaustein "TON_DB" und überschreibt dabei den Zeitwert des Operanden "Tag_PresetTIME" innerhalb des Instanz-Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim Zugriff auf "MyTimer".Q oder "MyTimer".ET ändern.
TIMER := "TON_DB"	// IEC-Zeit wird zurückgesetzt

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

Im Programmcode müssen Sie die Anweisung "Vorwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Instanz-Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB / Einzelinstanz)

Der Systemdatentyp des Instanz-Datenbausteins wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Systemdatentyp des Instanz-Datenbausteins (Shared DB)
SINT / USINT	IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Einzelinstanz anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Lokale Variable (Multiinstanz)

Der Datentyp der lokalen Variable wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Datentyp der lokalen Variable
SINT / USINT	CTU_SINT / CTU_USINT / IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	CTU_INT / CTU_UINT / IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	CTU_DINT / CTU_UDINT / IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	CTU_LINT / CTU_ULINT / IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Multiinstanz in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler deklarieren

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet. Anschließend können Sie den IEC-Zähler wie folgt deklarieren:

- Einzelinstanz: Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Multiinstanz: Deklaration als lokale Variable vom Typ CTU_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L oder Konstante	Zähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL CTU, "CTU_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "CTU_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
CU := "Tag_StartCTU"	// Wenn der Signalzustand des Operanden "Tag_StartCTU" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins erhöht.
	// Der Zählwert wird so lange erhöht, bis er den oberen Grenzwert von INT = 32767 erreicht.
R := "Tag_ResetCounter"	// Wenn der Signalzustand des Operanden "Tag_ResetCounter" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf "0" zurückgesetzt.
PV := "Tag_PresetValue"	// Der Operand bestimmt, ab welchem Wert der Operand am Parameter Q gesetzt wird.
Q := "Tag_CounterStatus"	// Der Operand ist gesetzt, solange der aktuelle Zählwert größer oder gleich dem Wert am Parameter PV ist.
CV := "Tag_CounterValue"	// Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

Im Programmcode müssen Sie die Anweisung "Rückwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben. Falls Instanzen der Anweisung "Rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Instanz-Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB / Einzelinstanz)

Der Systemdatentyp des Instanz-Datenbausteins wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Systemdatentyp des Instanz-Datenbausteins (Shared DB)
SINT / USINT	IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Einzelinstanz anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Lokale Variable (Multiinstanz)

Der Datentyp der lokalen Variable wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Datentyp der lokalen Variable
SINT / USINT	CTD_SINT / CTD_USINT / IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	CTD_INT / CTD_UINT / IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	CTD_DINT / CTD_UDINT / IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	CTD_LINT / CTD_ULINT / IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Multiinstanz in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler deklarieren

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet. Anschließend können Sie den IEC-Zähler wie folgt deklarieren:

- Einzelinstanz: Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Multiinstanz: Deklaration als lokale Variable vom Typ CTD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L oder Konstante	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	**Erläuterung**
`CALL CTD, "CTD_DB"`	`// Die Anweisung wird aufgerufen. Der Datenbaustein "CTD_DB" wird der Anweisung zugeordnet.`
	`// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.`
`CD := "Tag_StartCTD"`	`// Wenn der Signalzustand des Operanden "Tag_StartCTD" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins verringert.`
	`// Der Zählwert am Parameter CV wird so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.`
`LD := "Tag_LoadPV"`	`// Wenn der Signalzustand des Operanden "Tag_LoadPV" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf den Wert des Operanden "Tag_PresetValue" gesetzt.`
`PV := "Tag_PresetValue"`	`// Gibt den Wert an, auf welchen der Zähler bei einem Signalzustand "1" am Parameter LD gesetzt wird.`

AWL	Erläuterung
Q := "Tag_CounterStatus"	// Der Operand wird gesetzt, wenn der aktuelle Zählwert kleiner oder gleich Null ist.
CV := "Tag_CounterValue"	// Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Parameter CV abgelegt. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

Im Programmcode müssen Sie die Anweisung "Vorwärts und rückwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Instanz-Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB / Einzelinstanz)

Der Systemdatentyp des Instanz-Datenbausteins wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Systemdatentyp des Instanz-Datenbausteins (Shared DB)
SINT / USINT	IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Einzelinstanz anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Lokale Variable (Multiinstanz)

Der Datentyp der lokalen Variable wird vom Datentyp der Anweisung abgeleitet:

Datentyp der Anweisung	Datentyp der lokalen Variable
SINT / USINT	CTUD_SINT / CTUD_USINT / IEC_SCOUNTER / IEC_USCOUNTER
INT / UINT	CTUD_INT / CTUD_UINT / IEC_COUNTER / IEC_UCOUNTER
DINT / UDINT	CTUD_DINT / CTUD_UDINT / IEC_DCOUNTER / IEC_UDCOUNTER
LINT / ULINT	CTUD_LINT / CTUD_ULINT / IEC_LCOUNTER / IEC_ULCOUNTER

Wenn Sie den IEC-Zähler als Multiinstanz in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler deklarieren

Nach der Auswahl des Datentyps aus der Klappliste "???" wird automatisch der Dialog "Aufrufoptionen" geöffnet. Anschließend können Sie den IEC-Zähler wie folgt deklarieren:

- Einzelinstanz: Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Multiinstanz: Deklaration als lokale Variable vom Typ CTUD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L oder Konstante	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L oder Konstante	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL CTUD, "CTUD_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "CTUD_DB" wird der Anweisung zugeordnet.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
CU := "Tag_StartCTU"	// Wenn der Signalzustand des Operanden "Tag_StartCTU" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins erhöht.
	// Der Zählwert wird bei einer positiven Signalflanke am Parameter CU so lange erhöht, bis er den oberen Grenzwert von INT = 32767 erreicht.
CD := "Tag_StartCTD"	// Wenn der Signalzustand des Operanden "Tag_StartCTD" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins verringert.
	// Der Zählwert am Parameter CV wird so lange verringert, bis er den unteren Grenzwert von -32768 erreicht.
R := "Tag_ResetCounter"	// Wenn der Signalzustand des Operanden "Tag_ResetCounter" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf "0"zurückgesetzt.
LD := "Tag_LoadPV"	// Wenn der Signalzustand des Operanden "Tag_LoadPV" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf den Wert des Operanden "Tag_PresetValue" gesetzt.
PV := "Tag_PresetValue"	// Gibt den Wert an, auf welchen der Zähler bei einem Signalzustand "1" am Parameter LD gesetzt wird.
QU := "Tag_CounterStatus"	// Der Operand ist gesetzt, solange der aktuelle Zählwert größer oder gleich dem Wert am Parameter PV ist.
QD := "Tag_CounterStatus"	// Der Operand wird gesetzt, wenn der aktuelle Zählwert kleiner oder gleich Null ist.
CV := "Tag_CounterValue"	//Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

Vergleicher

CompType: Variablen strukturierten Datentyps vergleichen

Beschreibung

Mit der Anweisung "Variablen strukturierten Datentyps vergleichen" fragen Sie ab, ob der erste Vergleichswert einer strukturierten Variable (IN1) gleich oder ungleich dem zweiten Vergleichswert einer anderen strukturierten Variable (IN2) ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung am Parameter OUT das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Vergleich von Gleitpunktzahlen

Bei Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefern sowohl die Anweisung "CompType EQ" als auch "CompType NE" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für EQ-Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'AA'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'aa'	'aaa'	0

Die folgende Tabelle zeigt Beispiele für NE-Vergleiche von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'Hallo Welt'	'HalloWelt'	1
'AA'	'AA'	0
'aa'	'aaa'	1

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefern sowohl die Anweisung "CompType EQ" als auch "CompType NE" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Die folgenden Datentypen sind möglich:

- PLC-Datentyp
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp (UDT) enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Anonyme Strukturen sind nicht zulässig.)
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Die folgende Tabellen zeigen Beispiele für einen Vergleich von Strukturen auf "Gleich":

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Die folgende Tabellen zeigen Beispiele für einen Vergleich von Strukturen auf "Ungleich":

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit Variable vom Datentyp A)		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variablen strukturierten Datentyps vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P	Erster Vergleichswert
IN2	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, Datum und Uhrzeit, ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen, STRUCT, VARIANT, ANY, PLC-Datentyp	E, A, M, D, L, P	Zweiter Vergleichswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung
Wie oben beschrieben, stehen Ihnen die Datentypen ARRAY, STRUCT (innerhalb eines PLC-Datentyps), VARIANT, ANY und PLC-Datentyp (UDT) erst ab der Firmware Version 2.0 zur Verfügung.				

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL CompType	// Die Anweisung wird aufgerufen.
	// Wählen Sie die gewünschte Funktion. Entweder "EQ" oder "NE" aus der Klappliste aus.
IN1 := "Tag_Operand1"	// Erster Vergleichswert
IN2 := "Tag_Operand2"	// Zweiter Vergleichswert
OUT := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand "Tag_Operand1" ist gleich "Tag_Operand2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

VARIANT

EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie ab, welchen Datentyp die Variable hat, auf die der VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CompType" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CompType: Variablen strukturierten Datentyps vergleichen (Seite 1361)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL EQ_Type	// Die Anweisung wird aufgerufen.
IN1 := #Tag_Operand1	// Erster zu vergleichender Operand
IN2 := "Tag_Operand2"	// Zweiter zu vergleichender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand #Tag_Operand1 ist gleich "Tag_Operand2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen

Beschreibung

Mit der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen" fragen Sie den Datentyp ab, welchen eine Variable nicht hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CompType" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CompType: Variablen strukturierten Datentyps vergleichen (Seite 1361)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Ergebnis

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                                 Erläuterung
CALL NE_Type                        // Die Anweisung wird aufgerufen.
IN1 := #Tag_Operand1                // Erster zu vergleichender Operand
IN2 := "Tag_Operand2"               // Zweiter zu vergleichender Operand
RET_VAL := "Tag_Result"             // Der Ausgang "Tag_Result" liefert den Signalzu-
                                    stand "1", wenn die Bedingung der Vergleichsanwei-
                                    sung erfüllt ist. D. h. der Operand #Tag_Operand1
                                    ist ungleich "Tag_Operand2".
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

EQ_ElemType: Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen" fragen Sie ab, welchen Datentyp die Variable hat, auf die der VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CompType" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CompType: Variablen strukturierten Datentyps vergleichen (Seite 1361)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Ergebnis

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL EQ_ElemType	// Die Anweisung wird aufgerufen.
IN1 := #Tag_Operand1	// Erster zu vergleichender Operand
IN2 := "Tag_Operand2"	// Zweiter zu vergleichender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand #Tag_Operand1 ist gleich "Tag_Operand2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

NE_ElemType: Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen " fragen Sie ab, welchen Datentyp die Variable nicht hat, auf die der VARIANT zeigt. Sie vergleichen den Datentyp der Variable am Parameter IN1, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp der Variable am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp VARIANT haben. Die Variable am Parameter IN2 kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Vergleich von Zeiten, Datum und Uhrzeit

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Strukturen

Für den Vergleich von Strukturen steht Ihnen die Anweisung "CompType" zur Verfügung. Anonyme Strukturen können generell nicht verglichen werden, außer, sie sind Bestandteil des gleichen ARRAYs.

Weitere Informationen zum Vergleich von Strukturen finden Sie hier: CompType: Variablen strukturierten Datentyps vergleichen (Seite 1361)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, PLC-Datentypen	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Ergebnis

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL NE_ElemType	// Die Anweisung wird aufgerufen.
IN1 := #Tag_Operand1	// Erster zu vergleichender Operand
IN2 := "Tag_Operand2"	// Zweiter zu vergleichender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand #Tag_Operand1 ist ungleich "Tag_Operand2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

IS_NULL: Auf GLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf GLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz auf einen NULL-Zeiger verweist und damit nicht auf ein Objekt zeigt.

Die Variable am Parameter OPERAND muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "0", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf GLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OPERAND	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf GLEICH NULL verglichen wird
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL IS_NULL	// Die Anweisung wird aufgerufen.
OPERAND := #Tag_Operand	// Zu vergleichender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand #Tag_Operand zeigt auf kein Objekt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

NOT_NULL: Auf UNGLEICH NULL-Zeiger abfragen

Beschreibung

Mit der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen" fragen Sie ab, ob der VARIANT oder die Referenz nicht auf einen NULL-Zeiger verweist und damit auf ein Objekt zeigt.

Die Variable am Parameter OPERAND muss den Datentyp VARIANT oder REF_TO <Datentyp> haben.

Hinweis

VARIANT-Variable zeigt auf einen ANY-Zeiger

Wenn die VARIANT-Variable auf einen ANY-Zeiger zeigt, dann liefert die Anweisung in jedem Fall das Ergebnis VKE = "1", auch wenn der ANY-Zeiger den Wert NULL hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf UNGLEICH NULL-Zeiger abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OPERAND	Input	VARIANT oder REF_TO <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf UNGLEICH NULL verglichen wird
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL NOT_NULL	// Die Anweisung wird aufgerufen.
OPERAND := #Tag_Operand	// Zu vergleichender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. der Operand #Tag_Operand zeigt auf ein Objekt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

IS_ARRAY: Auf ARRAY abfragen

Beschreibung

Mit der Anweisung "Auf ARRAY abfragen" fragen Sie ab, ob der VARIANT auf eine Variable vom Datentyp ARRAY zeigt.

Die Variable am Parameter OPERAND hat den Datentyp VARIANT. Das Ergebnis der Abfrage wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf ARRAY abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OPERAND	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf ARRAY abgefragt wird
RET_VAL	Output	BOOL	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL IS_ARRAY	// Die Anweisung wird aufgerufen.
OPERAND := #Tag_Input	// Auf ARRAY abzufragender Operand
RET_VAL := "Tag_Result"	// Der Ausgang "Tag_Result" liefert den Signalzustand "1", wenn die Bedingung der Vergleichsanweisung erfüllt ist. D. h. wenn der VARIANT am Operand #Tag_Input auf ein ARRAY zeigt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum VARIANT (Seite 339)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

EQ_TypeOfDB: Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf GLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable am Parameter IN1 adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit einem Datentyp am Parameter IN2 auf "Gleich".

Die Variable am Parameter IN1 muss den Datentyp DB_ANY haben. Die Variable am Parameter IN2 kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	TYPE_ID	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL EQ_TypeOfDB	// Aufruf der Anweisung
IN1 := #InputDBAny	// Auf DB_ANY abzufragender Operand
IN2 := TO_SpeedAxis	// Die Bedingung der Vergleichsanweisung ist erfüllt, wenn der Datentyp des Operanden #InputDBAny adressierten DB gleich dem Datentyp TO_SpeedAxis ist.
RET_VAL := "TagOut"	// Vergleichsergebnis

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

NE_TypeOfDB: Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen

Beschreibung

Mithilfe der Anweisung "Datentyp eines indirekt adressierten DB auf UNGLEICH mit einem Datentyp vergleichen" fragen Sie ab, welchen Datentyp der Datenbaustein nicht hat, den die Variable vom Datentyp DB_ANY adressiert. Sie vergleichen den Datentyp des durch die Variable am Parameter IN1 adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit dem Datentyp am Parameter IN2 auf "Ungleich".

Die Variable am Parameter IN1 muss den Datentyp DB_ANY haben. Die Variable am Parameter IN2 kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Erster Operand
IN2	Input	TYPE_ID	E, A, M, D, L, P	Zweiter Operand
RET_VAL	Output	BOOL	E, A, M, D, L	Vergleichsergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL NE_TypeOfDB	// Aufruf der Anweisung
IN1 := #InputDBAny	// Auf DB_ANY abzufragender Operand
IN2 := TO_SpeedAxis	// Die Bedingung der Vergleichsanweisung ist erfüllt, wenn der Datentyp des Operanden #InputDBAny adressierten DB ungleich dem Datentyp TO_SpeedAxis ist.
RET_VAL := "TagOut"	// Vergleichsergebnis

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Datentyp DB_ANY verwenden (Seite 215)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

Mathematische Funktionen

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den kleinsten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Parameter OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)				

Aus der Klappliste "???" können Sie den Datentyp für die Parameter INn und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL MIN	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN1 := "TagIn_Value1"	// Erster Eingangswert, der verglichen wird.
IN2 := "TagIn_Value2"	// Zweiter Eingangswert, der verglichen wird.
IN3 := "TagIn_Value3"	// Dritter Eingangswert, der verglichen wird.
OUT := "Tag_Minimum"	// Der Operand "Tag_Minimum" wird mit dem Wert des Operanden "TagIn_Value1" beschrieben, da dieser den kleinsten Wert aufweist.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den größten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis
Wenn die IEC-Prüfung nicht aktiviert ist, können Sie auch Variablen vom Datentyp TIME, LTIME, TOD, LTOD, DATE und LDT verwenden, indem Sie als Datentyp der Anweisung eine gleich lange Bitfolge oder Ganzzahl auswählen. (Z. B. statt TIME => DINT, UDINT oder DWORD = 32 Bit)				

Aus der Klappliste "???" können Sie den Datentyp für die Parameter INn und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL MAX	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN1 := "TagIn_Value1"	// Erster Eingangswert, der verglichen wird.
IN2 := "TagIn_Value2"	// Zweiter Eingangswert, der verglichen wird.
IN3 := "TagIn_Value3"	// Dritter Eingangswert, der verglichen wird.
OUT := "Tag_Maximum"	// Der Operand "Tag_Maximum" wird mit dem Wert des Operanden "TagIn_Value2" beschrieben, da dieser den größten Wert aufweist.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Maximum	14444

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Die Ausführung der Anweisung setzt voraus, dass alle Variablen an den Eingängen vom gleichen Datentyp sind.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis der am Parmeter IN angegebene Wert.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Parameter MN ist größer als der Wert am Parameter MX.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.				

Aus der Klappliste "???" können Sie den Datentyp für die Parameter MN, IN, MX und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	**Erläuterung**
`CALL LIMIT`	`// Die Anweisung wird aufgerufen.`
	`// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.`
`MN := "Tag_LowLimit"`	`// Untergrenze`
`IN := "Tag_InputValue"`	`// Eingangswert`
`MX := "Tag_HighLimit"`	`// Obergrenze`
`OUT := "Tag_Result"`	`// Der Operand "Tag_Result" wird mit dem Wert des Operanden "Tag_LowLimit" beschrieben, da der Wert des Operanden "Tag_InputValue" außerhalb des definierten Grenzbereichs liegt.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

Verschieben

MOVE: Wert kopieren

Beschreibung

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT. Die Operanden am Eingang IN und am Ausgang OUT müssen den identischen Datentyp haben.

Die folgende Tabelle zeigt die möglichen Übertragungen:

Quelle (IN)	Ziel (OUT)
BYTE	BYTE
WORD	WORD
DWORD	DWORD
LWORD	LWORD
SINT	SINT
USINT	USINT
INT	INT
UINT	UINT
DINT	DINT
UDINT	UDINT
LINT	LINT
ULINT	ULINT
REAL	REAL
LREAL	LREAL
S5TIME	S5TIME
TIME	TIME

Quelle (IN)	Ziel (OUT)
LTIME	LTIME
DATE	DATE
DT	DT
LDT	LDT
TOD	TOD
LTOD	LTOD
DTL	DTL
CHAR	CHAR
WCHAR	WCHAR
Zeichen einer Zeichenkette[1)]	Zeichen einer Zeichenkette
ARRAY	ARRAY
STRUCT	STRUCT
PLC-Datentyp (UDT)	PLC-Datentyp (UDT)
IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER
IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUNTER	IEC_ULCOUNTER
DB_ANY	DB_ANY
REF()	Es findet keine Datentypkonvertierung statt. Die übergebene Variable muss genau dem Zieldatentyp entsprechen.

[1)] Mit der Anweisung "Wert kopieren" können Sie auch einzelne Zeichen einer Zeichenkette zu Operanden vom Datentyp CHAR oder WCHAR übertragen. Die Nummer des zu übertragenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" übertragen. Die Übertragung vom Operanden des Datentyps CHAR zu einzelnen Zeichen einer Zeichenkette ist ebenfalls möglich. Sie können auch ein bestimmtes Zeichen einer Zeichenkette durch das Zeichen einer anderen Zeichenkette ersetzen.

Zum Kopieren von Operanden des Datentyps ARRAY können auch die Anweisungen "Bereich kopieren" (MOVE_BLK), "Bereich ununterbrechbar kopieren" (UMOVE_BLK) und "Bereich kopieren" (MOVE_BLK_VARIANT) verwendet werden. Operanden des Datentyps STRING können mit der Anweisung "Zeichenkette verschieben" (S_MOVE) kopiert werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT), DB_ANY	E, A, M, D, L oder Konstante	Element, mit dem die Zieladresse überschrieben wird
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT), DB_ANY	E, A, M, D, L	Zieladresse

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL MOVE                    // Die Anweisung wird aufgerufen.
IN := "TagIn_LREAL"          // Die Inhalte des Operanden "TagIn_LREAL" werden ko-
                             piert.
OUT := "TagOut_LREAL"        // Und in den Operanden "TagOut_LREAL" übertragen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_LREAL	1.23456789098e55
OUT	TagOut_LREAL	1.23456789098e55

Hinweis

Weitere Informationen zur Anweisung MOVE finden Sie im Siemens Industry Online Support in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Deserialize: Deserialisieren

Beschreibung

Mit der Anweisung "Deserialisieren" wandeln Sie die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder um und füllen dessen kompletten Inhalt.

Der Speicherbereich, in dem die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> liegt, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Speicherbereiche zugelassen. Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Stellen Sie vor der Umwandlung sicher, dass genügend Speicherplatz vorhanden ist. Wenn der Speicherbereich mithilfe der Anweisung "Serialisieren" befüllt wurde und eventuell Füll-Bytes eingefügt wurden, dann werden diese bei der Umwandlung nicht berücksichtigt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Mithilfe der Anweisung können Sie mehrere sequenzielle Darstellungsformen von umgewandelten Daten schrittweise wieder in ihren ursprünglichen Zustand umwandeln.

Wenn Sie nur eine einzelne sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder umwandeln möchten, dann können Sie dazu auch direkt die Anweisung "TRCV: Daten über Kommunikationsverbindung empfangen" verwenden.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das BOOL eine Länge von 1 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Optimierter Speicherbereich

Um größere Strukturen zu deserialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich für die sequenzielle Darstellung auch mit optimiertem Zugriff deklarieren. Der Inhalt der sequenziellen Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich. Der Zugriff auf die Bytes des ARRAYs ist ausschließlich symbolisch möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deserialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_ARRAY	Input	VARIANT ARRAY[*] of BYTE [1]	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der Datenstrom gespeichert ist.
DEST_VARIABLE	InOut	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die deserialisiert werden soll.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
[1] Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 möglich				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_ARRAY und DEST_VARIABLE überschneiden sich.
8136	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8150	Der Datentyp VARIANT am Parameter SRC_ARRAY enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_ARRAY
8153	Am Parameter SRC_ARRAY steht nicht genügend Speicherplatz zur Verfügung.
8154	Ungültiger Datentyp am Parameter SRC_ARRAY

Fehler-code* (W#16#...)	Erläuterung
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_VARIABLE
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8136	Der Zugriff am Parameter SRC_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle eines FBs oder FCs
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle eines FBs oder FCs
Error	INT	
Label	STRING[4]	
cmp	BOOL	

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Target	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erklärung
L 0	// Der Wert "0" wird in den Akkumulator 1 geladen.
T #BufferPos	// Der Wert "0" wird in den Operanden #BufferPos transferiert.
CALL Deserialize	// Die Anweisung wird aufgerufen.
SRC_ARRAY := "Buffer".Field	// Die Kundendaten aus der Variable "Buffer" werden deserialisiert.
RET_VAL := #Error	// Fehlerinformation
DEST_VARIABLE := "Target".Client	// Die deserialisierten Kundendaten werden in die Variable "Target" geschrieben.
POS := #BufferPos	// Der Operand #BufferPos speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen.
UN BIE SPB end	// Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann springt die Programmbearbeitung ans Ende.
CALL Deserialize	// Die Anweisung wird aufgerufen.
SRC_ARRAY := "Buffer".Field	// Das Trennblatt aus der Variable "Buffer" wird deserialisiert. Das Trennblatt, in Form der Zeichen 'arti' oder 'Bill' wurde eingefügt, um die Kundendaten besser differenzieren zu können.
RET_VAL := #Error	// Fehlerinformation
DEST_VARIABLE := #Label	// Die deserialisierten Zeichen des Trennblatts werden in den Operanden #Label geschrieben.
POS := #BufferPos	// Die Anzahl der Bytes, die die umgewandelten Zeichen belegen, wird im Operanden #BufferPos gespeichert.
UN BIE SPB end	// Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann springt die Programmbearbeitung ans Ende.

AWL	Erklärung
CALL S_COMP	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN1 := 'arti' IN2 := #Label	// Vergleich der Zeichen 'arti' mit den Zeichen, die im Operanden #Label gespeichert wurden.
OUT := #cmp	// Ergebnis der Anweisung
U #cmp SPBN noarticle	// Wenn der Vergleich den Signalzustand "0" liefert, dann handelt es sich im Operanden #Label nicht um die Zeichen 'arti' und die Programmbearbeitung springt zur Sprungmarke "noarticle".
CALL Deserialize	// War das Ergbnis des Vergleichs = "1", dann wird die Anweisung aufgerufen.
SRC_ARRAY := "Buffer".Field	// Die Artikeldaten werden deserialisiert.
RET_VAL := #Error	// Fehlerinformation
DEST_VARIABLE := "Target".Article[#DeliverPos]	// Die deserialisierten Artikeldaten werden in die Variable "Target" geschrieben.
POS := #BufferPos	// Die Anzahl der Bytes, die die umgewandelten Artikeldaten belegen, werden in den Operanden #BufferPos gespeichert.
UN BIE SPB end BE	// Wenn während der Bearbeitung kein Fehler aufgetreten ist, dann ist hier das Bausteinende erreicht.
noarticle: NOP 0	// Wenn der Vergleich auf 'arti' = "0" war,dann geht hier die Programmbearbeitung weiter.
CALL S_COMP	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN1 := 'Bill' IN2 := #Label	// Vergleich der Zeichen 'Bill' mit den Zeichen, die im Operanden #Label gespeichert wurden.
OUT := #cmp	// Ergebnis der Anweisung
U #cmp SPBN end	// Wenn der Vergleich den Signalzustand "1" liefert, dann handelt es sich im Operanden #Label um die Zeichen 'Bill' und die Programmbearbeitung geht weiter.
CALL Deserialize	// War das Ergbnis des Vergleichs = "1", dann wird die Anweisung aufgerufen.

AWL	Erklärung
SRC_ARRAY := "Buffer".Field	// Die Rechnungsdaten werden deserialisiert.
RET_VAL := #Error	// Fehlerinformation
DEST_VARIABLE := "Target".Bill[#DeliverPos]	// Die deserialisierten Kundendaten werden in die Variable "Target" geschrieben.
POS := #BufferPos	// Die Anzahl der Bytes, die die umgewandelten Artikeldaten belegen, werden in den Operanden #BufferPos gespeichert.
UN BIE SPB end BE	// Wenn während der Bearbeitung kein Fehler aufgetreten ist, dann ist hier das Bausteinende erreicht.
end: CLR	// Sprungmarke "end" wird im Fehlerfall angesprungen.
SAVE	// Das VKE wird im BIE-Bit gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu AWL (Seite 7557)

Serialize: Serialisieren

Beschreibung

Mit der Anweisung "Serialisieren" wandeln Sie mehrere PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> in eine sequenzielle Darstellungsform um, ohne dass Teile ihrer Struktur verloren gehen.

Mithilfe der Anweisung können Sie mehrere strukturierte Daten aus Ihrem Programm in einem Puffer, der vorzugsweise in einem globalen Datenbaustein liegt, zwischenspeichern und an eine andere CPU senden. Der Speicherbereich, in dem die umgewandelten Daten abgelegt werden, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Daten zugelassen. Fülldaten des Datenbereichs der Quelle sind im Ziel-ARRAY undefiniert. Diese

können sowohl Füll-Bytes bzw. Füll-Bits eines Datenbereichs (z. B. ARRAY, STRUCT oder PLC-Datentyp (UDT)) sowie die aktuell ungenutzten Zeichen eines Strings sein.

Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Strukturen, die gemäß der Regeln für standard Speicherbereiche größer als 64 KB sind, können nicht serialisiert werden, wenn der Operand am Parameter DEST_ARRAY in einem standard Speicherbereich liegt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Der Operand am Parameter POS enthält die Information, wie viele Bytes die umgewandelten Daten belegen.

Wenn Sie einen einzelnen PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> versenden möchten, dann können Sie direkt die Anweisung "TSEND: Daten über Kommunikationsverbindung senden" aufrufen.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hängt die Länge eines BOOL davon ab, welcher Datentyp folgt. Das bedeutet, wenn z. B. ein BYTE folgt, hat das BOOL eine Länge von 1 Byte. Wenn z. B. ein WORD folgt, dann hat das BOOL eine Länge von 2 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Daher ist es empfehlenswert, dass der Quelldatenbereich für das Serialisieren mit den großen Datentypen beginnt und mit booleschen Elementen endet. Dadurch wird das Auffüllen mit Füll-Bits wesentllich reduziert.

Optimierter Speicherbereich

Um größere Strukturen zu serialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich auch mit optimiertem Zugriff deklarieren. Die sequenzielle Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Serialisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_VARIABLE	Input	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die serialisiert werden soll.
DEST_ARRAY	InOut	VARIANT	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der erzeugte Datenstrom gespeichert wird.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Gesamtzahl der Bytes, die die umgewandelten Kundendaten belegt haben. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_VARIABLE und DEST_ARRAY überschneiden sich.
8150	Der Datentyp VARIANT am Parameter SRC_VARIABLE enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_VARIABLE
8236	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_ARRAY
8253	Die Variable am Parameter DEST_ARRAY bietet für den Inhalt der Variable am Parameter SRC_VARIABLE nicht genügend Speicherplatz. Durch den Eingangswert der Variable am Parameter POS wird der zur Verfügung stehende Speicherplatz reduziert. Der Eingangswert des Parameters POS bestimmt, wo innerhalb der Variable am Parameter DEST_ARRAY begonnen wird.
8254	Ungültiger Datentyp am Parameter DEST_ARRAY
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8236	Der Zugriff am Parameter DEST_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L 0	// Der Wert "0" wird in den Akkumulator 1 geladen.
T #BufferPos	// Der Wert "0" wird in den Operanden #BufferPos transferiert.
CALL Serialize	// Die Anweisung wird aufgerufen.
SRC_VARIABLE := "Source".Client	// Die Kundendaten aus der Variable "Source" werden serialisiert.
RET_VAL := #Error	// Fehlerinformation
DEST_ARRAY := "Buffer".Field	// Die sequenzielle Darstellung wird in die Variable "Buffer" geschrieben.
POS := #BufferPos	// Der Index des nächsten unbeschriebenen Bytes des Operanden "Buffer".Field wird im Operanden #BufferPos gespeichert.
CALL S_MOVE	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := 'arti' OUT := #Label	// Die Zeichen 'arti' werden in den Operanden #Label kopiert.
CALL Serialize	// Die Anweisung wird aufgerufen.
SRC_VARIABLE := #Label	// Die Zeichen werden serialisiert.
RET_VAL := #Error	// Fehlerinformation
DEST_ARRAY := "Buffer".Field	// Die sequenzielle Darstellung wird nach den Kundendaten in die Variable "Buffer".Field geschrieben.
POS := #BufferPos	// Der Wert des Operanden #BufferPos wird entsprechend erhöht.
CALL Serialize	// Die Anweisung wird aufgerufen.
SRC_VARIABLE := "Source".Article[#DeliverPos]	// Die Daten eines bestimmten Artikels aus der Variable "Source" werden serialisiert.

AWL	Erläuterung
RET_VAL := #Error	// Fehlerinformation
DEST_ARRAY := "Buffer".Field	// Die sequenzielle Darstellung wird nach den Zeichen 'arti' in die Variable "Buffer" geschrieben.
POS := #BufferPos	// Anzahl der Bytes

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle
Error	INT	Im Abschnitt "Temp" der Bausteinschnittstelle
Label	STRING[4]	Im Abschnitt "Temp" der Bausteinschnittstelle

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Source	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
Buffer	Field	Array[0..294] of BYTE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu AWL (Seite 7557)

MOVE_BLK: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Parameter IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.				

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL MOVE_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := #a_array[2]	// Erstes Element, ab dem kopiert wird.
COUNT := "Tag_Count"	// Anzahl der Elemente, die in den Zielbereich kopiert werden.
OUT := #b_array[1]	// Erstes Element, in das kopiert wird.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

MOVE_BLK_VARIANT: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können ein komplettes ARRAY oder Elemente eines ARRAYs in ein anderes ARRAY des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-ARRAY dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines ARRAYs umkopieren oder einzelne Elemente kopieren.

Die Anzahl der Elemente, die kopiert werden sollen, darf den selektierten Quellbereich bzw. Zielbereich nicht überschreiten.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss das ARRAY noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC_INDEX und DEST_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des ARRAYs.

Die Anweisung wird nicht ausgeführt, wenn mehr Daten kopiert werden, als zur Verfügung gestellt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input [2)]	VARIANT (der auf ein ARRAY oder ein einzelnes ARRAY-Element zeigt), ARRAY of <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Der Quellbereich, aus dem kopiert wird
COUNT	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Elemente, die kopiert werden Geben Sie dem Parameter COUNT den Wert "1", wenn am Parameter SRC oder am Parameter DEST kein ARRAY angegeben ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_INDEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert das erste Element, das kopiert wird: • Der Parameter SRC_INDEX wird nullbasiert berechnet. Wenn am Parameter SRC ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter SRC_INDEX das erste Element innerhalb des Quellbereichs an, aus dem kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter SRC kein ARRAY oder nur ein einzelnes Element eines ARRAYs angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert "0".
DEST_IN-DEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert den Anfang des Ziel-speicherbereichs: • Der Parameter DEST_INDEX wird nullbasiert berechnet. Wenn am Parameter DEST ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter DEST_INDEX das erste Element innerhalb des Zielbereichs an, in das kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter DEST kein ARRAY angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST	Output [1]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zielbereich, in den die Inhalte des Quellbereichs kopiert werden
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter DEST ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden. 2) Am Parameter SRC sind die Datentypen BOOL und ARRAY of BOOL nicht zulässig.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Die Datentypen stimmen nicht überein. Verwenden Sie statt eines ARRAY of Struct ein ARRAY of PLC-Datentyp(UDT).
8151	Der Zugriff auf den Parameter SRC ist nicht möglich.
8152	Der Operand am Parameter SRC ist nicht typisiert.
8153	Codeerzeugungsfehler am Parameter SRC
8154	Der Operand am Parameter SRC hat den Datentyp BOOL.
8281	Ungültiger Wert am Parameter COUNT
8382	Der Wert am Parameter SRC_INDEX ist kleiner NULL.
8383	Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8482	Der Wert am Parameter DEST_INDEX ist kleiner NULL.
8483	Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8534	Der Parameter DEST ist schreibgeschützt
8551	Der Zugriff auf den Parameter DEST ist nicht möglich.
8552	Der Operand am Parameter DEST ist nicht typisiert.
8553	Codeerzeugungsfehler am Parameter DEST
8554	Der Operand am Parameter DEST hat den Datentyp BOOL.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL MOVE_BLK_VARIANT	// Die Anweisung wird aufgerufen.
SRC := #SrcField	// Erstes Element, das kopiert wird
COUNT := "Tag_Count"	// Anzahl der Elemente, die in den Zielbereich kopiert werden.
SRC_INDEX := "Tag_Src_Index"	// Erstes Element am Parameter SRC, das kopiert wird
DEST_INDEX := "Tag_Dest_Index"	// Erstes Element am Parameter DEST, das kopiert wird
DEST := #DestField	// Erstes Element des Quellbereichs, das kopiert wird.
RET_VAL := "Tag_Result"	// Fehlerinformationen

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
SRC	Input	#SrcField	Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[0..10] of "MOVE_UDT"
COUNT	Input	Tag_Count	2
SRC_INDEX	Input	Tag_Src_Index	3
DEST_INDEX	Input	Tag_Dest_Index	3
DEST	InOut	#DestField	Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[10..20] of "MOVE_UDT"

2 Elemente werden aus dem Quellbereich, beginnend mit dem vierten Element des ARRAY[0..10] of MOVE_UDT, in den Zielbereich kopiert. Die Kopien werden im ARRAY[10..20] of MOVE_UDT ab dem vierten Element eingefügt.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK_VARIANT im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu AWL (Seite 7557)

UMOVE_BLK: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Wert am Ausgang OUT ist ungültig, wenn mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.				

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL UMOVE_BLK`	`// Die Anweisung wird aufgerufen.`
	`// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.`
`IN := #a_array[2]`	`// Erstes Element, ab dem kopiert wird.`
`COUNT := "Tag_Count"`	`// Anzahl der Elemente, die in den Zielbereich kopiert werden.`
`OUT := #b_array[1]`	`// Erstes Element, in das kopiert wird.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Hinweis

Weitere Informationen zur Anweisung UMOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

FILL_BLK: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Adresse im Zielbereich, ab der befüllt wird

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

```
AWL                          Erläuterung
CALL FILL_BLK                // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
   IN := #FillValue          // Zu kopierender Wert
   COUNT := "Tag_Count"      // Anzahl der Kopierwiederholungen
   OUT := #TargetArea[1]     // Der Zielbereich wird ab dem ersten Element des AR-
                             RAYs #TargetArea[1..5] of INT befüllt
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1		Datentyp
MyStruct1		STRUCT
	Member_1	INT
	Member_2	INT
	Member_3	INT
	Member_4	INT
MyStruct2		STRUCT

Data_block_1					Datentyp
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

AWL	Erläuterung
CALL FILL_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := 10	// Zu kopierender Wert
COUNT := 2	// Anzahl der Kopierwiederholungen
OUT := "Data_block_1".MyStruct1.Member_2	// Ab Member_2 wird die Struktur befüllt. Das bedeutet, dass Member_1 und Member_4 nicht verändert werden.

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

AWL	Erläuterung
CALL FILL_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := 10	// Zu kopierender Wert
COUNT := 2	// Anzahl der Kopierwiederholungen
OUT := "Data_block_1".MyStruct2.SubArray[1].Member_2	// Ab Member_2 wird die Struktur befüllt. Das bedeutet, dass Member_1 und Member_4 nicht verändert werden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

UFILL_BLK: Bereich ununterbrechbar befüllen

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Adresse im Zielbereich, ab der befüllt wird

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

```
AWL                              Erläuterung
CALL UFILL_BLK                   // Die Anweisung wird aufgerufen.
                                 // Wählen Sie den gewünschten Datentyp aus der Klapp-
                                 liste "???" aus.
   IN := #FillValue              // Zu kopierender Wert
   COUNT := "Tag_Count"          // Anzahl der Kopierwiederholungen
   OUT := #TargetArea[1]         // Der Zielbereich wird ab dem ersten Element des AR-
                                 RAYs #TargetArea[1..5] of INT befüllt
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

AWL	Erläuterung
CALL UFILL_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := 10	// Zu kopierender Wert
COUNT := 2	// Anzahl der Kopierwiederholungen
OUT := "Data_block_1".MyStruct1.Member_2	// Ab Member_2 wird die Struktur befüllt. Das bedeutet, dass Member_1 und Member_4 nicht verändert werden.

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

AWL	Erläuterung
CALL UFILL_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
IN := 10	// Zu kopierender Wert
COUNT := 2	// Anzahl der Kopierwiederholungen

AWL	Erläuterung
OUT := "Data_block_1".MyStruct2.SubArray[1].Member_2	// Ab Member_2 wird die Struktur befüllt. Das bedeutet, dass Member_1 und Member_4 nicht verändert werden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SCATTER: Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Bitfolge in einzelne Bits zerlegen" zerlegen Sie eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Bitfolge in einzelne Bits zerlegen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der anonyme STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. ein Statuswort zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Bitfolge, die zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das bzw. den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationArray	Output	ARRAY[0..15] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL SCATTER                                // Aufruf der Anweisung
                                            // Wählen Sie den gewünschten Datentyp
                                            aus der Klappliste "???" aus.
    IN := #SourceWord                       // Der Operand #SourceWord vom Datentyp
    OUT := #DestinationArray                WORD wird in seine einzelnen Bits (16)
                                            zerlegt und den einzelnen Elementen des
                                            Operanden #DestinationArray zugewiesen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationArray	Der Operand "DestinationArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationUDT	Output	"myBits"

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL SCATTER                          // Aufruf der Anweisung
                                      // Wählen Sie den gewünschten Datentyp
                                      aus der Klappliste "???" aus.
    IN := #SourceWord                 // Der Operand #SourceWord vom Datentyp
    OUT := #DestinationUDT            WORD wird in seine einzelnen Bits (16)
                                      zerlegt und den einzelnen Elementen des
                                      Operanden #DestinationUDT zugewiesen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationUDT	Der Operand "DestinationUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen" zerlegen Sie ein oder mehrere Elemente eines ARRAY of BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen. Am Parameter COUNT_IN geben Sie an, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Das Quell-ARRAY am Parameter IN darf mehr Elemente haben, als am Parameter COUNT_IN angegeben sind. Das ARRAY of BOOL, der anonyme STRUCT oder der PLC-Datentyp muss genügend Elemente haben, um die Bits der zerlegten Bitfolgen speichern zu können. Der Zielspeicherbereich darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter IN ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter OUT ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Das ARRAY of WORD wird in 48 einzelne Bits zerlegt.

Hinweis

Wenn die untere ARRAY-Grenze des Ziel-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD- oder DWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. Statuswörter zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, das zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_IN	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das/den die einzelnen Bits gespeichert werden

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[0..95] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL SCATTER_BLK                        // Aufruf der Anweisung
                                        // Wählen Sie den gewünschten Datentyp
                                        aus der Klappliste "???" aus.
    IN := #SourceArrayWord[2]           // Das 3., 4. und 5. WORD des Operanden
                                        #SourceArrayWord wird in seine einzelnen
                                        Bits (48) zerlegt.
    COUNT_IN := #CounterInput           // 3 Wörter werden zerlegt
```

```
AWL
    OUT := DestinationArrayBool[0]        // Und ab dem 1. Element den einzelnen
                                          Elementen des Operanden #DestinationAr-
                                          rayBool zugewiesen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[0]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[-2..93] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL
CALL SCATTER_BLK                          // Aufruf der Anweisung
                                          // Wählen Sie den gewünschten Datentyp
                                          aus der Klappliste "???" aus.
    IN := #SourceArrayWord[2]             // Das 3., 4. und 5. WORD des Operanden
                                          #SourceArrayWord wird in seine einzelnen
                                          Bits (48) zerlegt.
    COUNT_IN := #CounterInput             // 3 Wörter werden zerlegt
    OUT := DestinationArrayBool[14]       // Und ab dem 16. Element den einzelnen
                                          Elementen des Operanden #DestinationAr-
                                          rayBool zugewiesen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[14]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einer Bitfolge zusammen. Die Bitfolge wird in einer Variable vom Datentyp BYTE, WORD, DWORD oder LWORD gespeichert.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der anonyme STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, dessen Bits zu einer Bitfolge zusammengefügt werden. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Zusammengefügte Bitfolge, gespeichert in einer Variable

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArray	Input	ARRAY[0..15] of BOOL
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL GATHER                                // Aufruf der Anweisung
                                           // Wählen Sie den gewünschten Datentyp
                                           aus der Klappliste "???" aus.
    IN := #SourceArray                     // Die Bits des Operanden #SourceArray
                                           werden zu einem WORD zusammengefügt.
    OUT := #DestinationWord                // Die 16 zusammengefügten Bits werden im
                                           Operanden #DestinationWord gespeichert.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArray	Der Operand "SourceArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceUDT	Input	"myBits"
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL GATHER                                // Aufruf der Anweisung
                                           // Wählen Sie den gewünschten Datentyp
                                           aus der Klappliste "???" aus.
    IN := #SourceUDT                       // Die Bits des Operanden #SourceUDT wer-
                                           den zu einem WORD zusammengefügt.
    OUT := #DestinationWord                // Die 16 zusammengefügten Bits werden im
                                           Operanden #DestinationWord gespeichert.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceUDT	Der Operand "SourceUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einem oder mehreren Elementen eines ARRAY of <Bitfolge> zusammen. Am Parameter COUNT_OUT geben Sie an, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Damit geben Sie auch implizit vor, wie viele Elemente des ARRAY of BOOL, des anonymen STRUCTs oder des PLC-Datentyps benötigt werden. Das Ziel-ARRAY am Parameter OUT darf mehr Elemente haben, als am Parameter COUNT_OUT angegeben sind. Das ARRAY of <Bitfolge> muss genügend Elemente haben, um die Bits, die zusammengefügt werden sollen, speichern zu können. Das Ziel-ARRAY darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter OUT ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter IN ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Aus dem ARRAY of BOOL werden 48 einzelne Bits zusammengefügt.

Hinweis

Wenn die untere ARRAY-Grenze des Quell-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD- oder DWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY of BOOL, STRUCT oder PLC-Datentyp, dessen Bits zusammengefügt werden (Quell-ARRAY). Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_OUT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, in das die Bits gespeichert werden (Ziel-ARRAY)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[0..95] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL GATHER_BLK                          // Aufruf der Anweisung
                                         // Wählen Sie den gewünschten Datentyp
                                         aus der Klappliste "???" aus.
IN := SourceArrayBool[0]                 // Ab dem 1. Element des Operanden #Sour-
                                         ceArrayBool werden 48 Bits im Operanden
                                         #DestinationArrayWord zusammengefügt.
COUNT_OUT := #CounterOutput              // 3 Wörter werden zusammengefügt
```

AWL

```
OUT := DestinationArrayWord[2]          // Begonnen wird dabei im Ziel-ARRAY ab
                                        dem 3. Element. Das bedeutet, dass die
                                        ersten 16 Bits in das 3. Wort, die zwei-
                                        ten 16 Bits in das 4. Wort und die dritten
                                        16 Bits in das 5. Wort des Ziel-ARRAYs ge-
                                        schrieben werden.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[0]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[-2..93] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL

```
CALL GATHER_BLK                         // Aufruf der Anweisung
                                        // Wählen Sie den gewünschten Datentyp
                                        aus der Klappliste "???" aus.
IN := SourceArrayBool[14]               // Ab dem 16. Element des Operanden #Sour-
                                        ceArrayBool werden 48 Bits im Operanden
                                        #DestinationArrayWord zusammengefügt.
COUNT_OUT := #CounterOutput             // 3 Wörter werden zusammengefügt
```

```
AWL
OUT := DestinationArrayWord[2]          // Begonnen wird dabei im Ziel-ARRAY ab
                                        dem 3. Element. Das bedeutet, dass die
                                        ersten 16 Bits des Quell-ARRAYs nicht be-
                                        rücksichtigt werden. Die zweiten 16. Bits
                                        werden in das 3. Wort, die dritten 16
                                        Bits in das 4. Wort und die vierten 16
                                        Bits in das 5. Wort des Ziel-ARRAYs ge-
                                        schrieben werden. Die restlichen 64 Bits
                                        des Quell-ARRAYs werden ebenfalls nicht
                                        berücksichtigt
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[14]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Da aber erst ab dem 16. Element begonnen wird, stehen nur 80 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

AssignmentAttempt: Zuweisung eines VARIANT auf eine Referenz versuchen

Beschreibung

Mit der Anweisung "AssignmentAttempt" versuchen Sie die Zuweisung einer VARIANT-Variable auf eine Referenzvariable. Der Datentyp einer Referenzvariable wird zum Zeitpunkt der Deklaration festgelegt, während der Datentyp einer VARIANT-Variable während der Laufzeit ermittelt wird. Eine implizite Datentypkonvertierung ist bei Referenzvariablen nicht zulässig. Um die beiden Datentypen einander zuzuweisen, nutzen Sie deshalb den Zuweisungsversuch.

Beim Zuweisungsversuch wird zur Laufzeit geprüft, ob die VARIANT-Variable vom richtigen Datentyp ist. Wenn dies der Fall ist, wird die Zuweisung ausgeführt. Nach einer erfolgreichen Ausführung steht eine gültige Referenz in der Zielvariablen, andernfalls eine NULL.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	VARIANT	• Bausteinschnittstelle eines FC: Input, Output, InOut, Temp • Bausteinschnittstelle eines FB: Input, InOut, Temp	Zeiger auf die Quellvariable, deren Adresse ausgelesen wird
DST	Output	Referenz auf: • Bitfolgen, außer BOOL, • Ganzzahlen, • Gleitpunktzahlen, • Zeichenfolgen, • PLC-Datentypen (UDT), • Systemdatentypen (SDT), • ARRAYs aus den genannten Datentypen	• Bausteinschnittstelle eines FC: Input, Output, Temp • Bausteinschnittstelle eines FB: Temp	Referenz, an die die Adresse der Quellvariable übergeben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Regeln

Für den Zuweisungsversuch gelten die folgenden Regeln. VARIANT-Variablen, die nicht diesen Regeln entsprechen, liefern zur Laufzeit den Wert "NULL" zurück.

- Der VARIANT muss auf eine Adresse in einem optimierten Speicherbereich zeigen.
- Der VARIANT darf nicht auf eine Adresse in einem temporären Speicherberiech zeigen.
- Wenn Sie einer Referenz auf ein ARRAY einen VARIANT zuweisen wollen, gelten folgende Regeln:
  - Die VARIANT-Variable muss auf ein ARRAY zeigen, dessen Grenzen exakt mit denen der deklarierten Referenz übereinstimmen. Eine VARIANT-Variable, die auf ein ARRAY [0..9] zeigt, passt nicht zu einer Variable REF_TO ARRAY[1..10].
  - Darüberhinaus sollten Sie die Bausteine, die den Wert der VARIANT-Variable bilden, einmalig in einer CPU der Baureihe S7-1500, Firmwarestand V2.5, übersetzen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

In der Schnittstelle des Bausteins wurden der VARIANT "myVariant" und die Referenzvariable "myReference" deklariert.

		Name	Data type
1		Input	
2		Output	
3		Tag_Out	Bool
4		InOut	
5		myVariant	Variant
6		Static	
7		Temp	
8		myReference	REF_TO Int
9		Constant	

Im Programmcode wird versucht "myVariant" der Referenzvariable "myReference" zuzuweisen. Hat "myVariant" zur Laufzeit den Datentyp "Int", steht in "myReference" eine gültige Referenz auf die Quellvariable des VARIANT, andernfalls eine NULL. Ist die anschließende Abfrage auf "NOT_NULL" wahr, bedeutet dies, dass die Zuweisung erfolgreich war und der Ausgangsparameter "Tag_Out" geschrieben werden kann.

```
1        CALL  AssignmentAttempt
2           IN  :=#myVariant
3           OUT :=#myReference
4        CALL  NOT_NULL
5           OPERAND :=#myReference
6           Ret_Val :=#Tag_Out
7
```

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Zuweisungsversuch von VARIANT auf eine Referenz (Seite 329)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ARRAY-DB

ReadFromArrayDB: Aus Array-Datenbaustein lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie das Element aus einem Datenbaustein vom Bausteintyp ARRAY-DB, auf das der Index verweist, und schreiben den Wert des Elements in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der gelesen und ausgegeben wird

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter VALUE ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz, schreibgeschützt oder liegt im Ladespeicher.
8135	Der ARRAY-Datenbaustein enthält ungültige Werte.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8452	Codeerzeugungsfehler
8453	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL ReadFromArrayDB`	`// Aufruf der Anweisung`
`DB := "ArrayDB"`	`// Der Datenbaustein, aus dem gelesen wird.`
`INDEX := "ArrayDB".THIS[2]`	`// Das 2. Element wird aus dem Datenbaustein gelesen.`

AWL	Erläuterung
RET_VAL := "TagRet_Val"	// Fehlerinformationen
VALUE := "TargetField"	// Der gelesene Wert wird am Parameter VALUE ausgegeben.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.

Aus dem "ArrayDB" wird das 2. Element ausgelesen und in den Operanden "TargetField" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu AWL (Seite 7557)

WriteToArrayDB: In Array-Datenbaustein schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der geschrieben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8134	Der Datenbaustein ist schreibgeschützt.
8135	Der Datenbaustein ist kein ARRAY-Datenbaustein.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8350	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8352	Codeerzeugungsfehler

Fehler-code* (W#16#...)	Erläuterung
8353	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                                    Erläuterung
CALL WriteToArrayDB                    // Aufruf der Anweisung
DB := "ArrayDB"                        // Der Datenbaustein, in den geschrieben
                                       wird.
INDEX := 2                             // Der Wert wird in das Element [2] des
                                       Datenbausteins geschrieben.
RET_VAL := "TagRet_Val"                // Fehlerinformationen
VALUE := "SourceField"                 // Der Wert, der in den Datenbaustein ge-
                                       schrieben wird.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Der Wert des Operanden "SourceField" wird in das 2. Element des ARRAY-DB geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu AWL (Seite 7557)

ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie aus einem Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher das Element, auf das der Index verweist, und schreiben sie in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Auslesen des Array-DBs beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1)]	InOut	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Kontanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8750	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL ReadFromArrayDBL, "ReadFromArrayDBL_DB"`	`// Aufruf der Anweisung`
`REQ := "TagReg"`	`// Bei positiver Signalflanke wird die Anweisung ausgeführt.`
`DB := "ArrayDB"`	`// Der Datenbaustein, aus dem gelesen wird.`

AWL	Erläuterung
INDEX := 2	// Das Element [2] wird aus dem Datenbaustein gelesen.
VALUE := "TargetField"	// Der gelesene Wert wird am Parameter VALUE ausgegeben.
BUSY := "TagBusy"	// Zeigt an, ob die Anweisung noch in Bearbeitung ist.
DONE := "TagDone"	// Zeigt an, ob die Anweisung beendet ist.
ERROR := "TagError"	// Fehlerinformationen

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und am Parameter VALUE ausgegeben. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird nicht mehr verändert. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu AWL (Seite 7557)

WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Schreiben in den ARRAY-DB beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1)]	Input	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Schreiben in den ARRAY-DB ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8234	Der Datenbaustein ist schreibgeschützt.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL WriteToArrayDBL, "WriteToArrayDBL_DB"`	`// Aufruf der Anweisung`
`REQ := "TagReg"`	`// Bei positiver Signalflanke wird die Anweisung ausgeführt.`
`DB := "ArrayDB"`	`// Der Datenbaustein, aus dem gelesen wird.`

AWL	Erläuterung
INDEX := 2	// Das Element [2] wird aus dem Datenbaustein gelesen.
VALUE := "SourceField"	// Der gelesene Wert wird am Parameter VALUE ausgegeben.
BUSY := "TagBusy"	// Zeigt an, ob die Anweisung noch in Bearbeitung ist.
DONE := "TagDone"	// Zeigt an, ob die Anweisung beendet ist.
ERROR := "TagError"	// Fehlerinformationen

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird in das 2. Element des "ArrayDB" geschrieben. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu AWL (Seite 7557)

Lese-/Schreibzugriff

PEEK: Speicheradresse lesen

Beschreibung

Mit der Anweisung "Speicheradresse lesen" lesen Sie eine Speicheradresse ohne Angabe eines Datentyps aus einem Standard-Speicherbereich aus.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

```
CALL PEEK
AREA := 16#84
DBNUMBER := 1
BYTEOFFSET := 2
RET_VAL := #Peeker

JBI end
L 16#ffff
T #Peeker
end: NOP 0
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Speicheradresse lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#1: Peripherieeingang
DBNUMBER	Input	DINT, DB_ANY	E, A, M, D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BYTEOFF-SET	Input	DINT	E, A, M, D	Adresse, an der ausgelesen wird Es werden nur die 16 niederstwertigen Bits verwendet.
RET_VAL	Output	Bitfolgen	E, A, M, D	Ergebnis der Anweisung

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Hinweis

Wenn Sie die Speicheradresse aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL PEEK                    // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
AREA := "Tag_Area"           // Bereich Datenbaustein ausgewählt
DBNUMBER := "Tag_DBNumber"   // Nummer des Datenbaustein
BYTEOFFSET := "Tag_Byte"     // Adresse, an der ausgelesen wird
RET_VAL := "Tag_Result"      // Ergebnis
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
RET_VAL	Tag_Result	Wert des Byte "20"

Die Anweisung liest den Wert an der Adresse "20" aus dem Operanden "Tag_Byte" aus dem Datenbaustein "5" aus und liefert das Ergebnis am Operanden "Tag_Result" zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

PEEK_BOOL: Speicherbit lesen

Beschreibung

Mit der Anweisung "Speicherbit lesen" lesen Sie ein Speicherbit ohne Angabe eines Datentyps aus einem Standard-Speicherbereich aus.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

```
CALL PEEK_BOOL
AREA := 16#84
DBNUMBER := 1
BYTEOFFSET := 2
BITOFFSET := 1
RET_VAL := #Peeker_BOOL

SPBI end
L 16#ffff
T #Peeker_BOOL
end: NOP 0
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Speicherbit lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#1: Peripherieeingang
DBNUMBER	Input	DINT, DB_ANY	E, A, M, D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFFSET	Input	DINT	E, A, M, D	Adresse, an der ausgelesen wird Es werden nur die 16 niederstwertigen Bits verwendet.
BITOFFSET	Input	INT	E, A, M, D	Bit, an dem ausgelesen wird
RET_VAL	Output	BOOL	E, A, M, D	Ergebnis der Anweisung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Hinweis

Wenn Sie das Speicherbit aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                              Erläuterung
CALL PEEK_BOOL                   // Die Anweisung wird aufgerufen.
                                 // Wählen Sie den gewünschten Datentyp aus der Klapp-
                                 liste "???" aus.
AREA := "Tag_Area"               // Bereich Datenbaustein ausgewählt
DBNUMBER := "Tag_DBNumber"       // Nummer des Datenbaustein
BYTEOFFSET := "Tag_Byte"         // Adresse, an der ausgelesen wird
BITOFFSET := "Tag_Bit"           // Bit, an dem ausgelesen wird
RET_VAL := "Tag_Result"          // Ergebnis
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
RET_VAL	Tag_Result	3

Die Anweisung liest den Wert am Speicherbit "3" aus dem Operanden "Tag_Bit" aus dem Datenbaustein "5" aus dem Byte "20" aus und liefert das Ergebnis am Operanden "Tag_Result" zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

POKE: Speicheradresse schreiben

Beschreibung

Mit der Anweisung "Speicheradresse schreiben" schreiben Sie eine Speicheradresse ohne Angabe eines Datentyps in einen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Speicheradresse schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#2: Peripherieausgang
DBNUMBER	Input	DINT, DB_ANY	E, A, M, D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFF-SET	Input	DINT	E, A, M, D	Adresse, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
VALUE	Input	Bitfolgen	E, A, M, D	Wert, der geschrieben wird

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Hinweis

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL POKE                    // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
AREA := "Tag_Area"           // Bereich Datenbaustein ausgewählt
DBNUMBER := "Tag_DBNumber"   // Nummer des Datenbaustein
BYTEOFFSET := "Tag_Byte"     // Adresse, an der ausgelesen wird
VALUE := "Tag_Value"         // Wert, der geschrieben wird.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
VALUE	Tag_Value	16#11

Die Anweisung überschreibt die Speicheradresse "20" im Datenbaustein "5" mit dem Wert "16#11".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

POKE_BOOL: Speicherbit schreiben

Beschreibung

Mit der Anweisung "Speicherbit schreiben" schreiben Sie ein Speicherbit ohne Angabe eines Datentyps in einen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Speicherbit schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#2: Peripherieausgang
DBNUMBER	Input	DINT, DB_ANY	E, A, M, D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFFSET	Input	DINT	E, A, M, D	Adresse, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
BITOFFSET	Input	INT	E, A, M, D	Bit, das geschrieben wird
VALUE	Input	BOOL	E, A, M, D	Wert, der geschrieben wird

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Hinweis

Wenn Sie das Speicherbit in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                              Erläuterung
CALL POKE_BOOL                   // Die Anweisung wird aufgerufen.
                                 // Wählen Sie den gewünschten Datentyp aus der Klapp-
                                 liste "???" aus.
AREA := "Tag_Area"               // Bereich Datenbaustein ausgewählt
DBNUMBER := "Tag_DBNumber"       // Nummer des Datenbaustein
BYTEOFFSET := "Tag_Byte"         // Adresse, an der ausgelesen wird.
BITOFFSET := "Tag_Bit"           // Bit, an dem ausgelesen wird.
VALUE := "Tag_Value"             // Wert, der geschrieben wird.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
VALUE	Tag_Value	M0.0

Die Anweisung überschreibt das Speicherbit "3" im Datenbaustein "5" im Byte "20" mit dem Wert "M0.0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

POKE_BLK: Speicherbereich schreiben

Beschreibung

Mit der Anweisung "Speicherbereich schreiben" schreiben Sie einen Speicherbereich ohne Angabe eines Datentyps in einen anderen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Speicherbereich schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA_SRC	Input	BYTE	E, A, M, D	Folgende Bereiche können Sie im Quellspeicherbereich auswählen: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB
DBNUM-BER_SRC	Input	DINT, DB_ANY [1]	E, A, M, D	Nummer des Datenbausteins im Quellspeicherbereich, wenn AREA = DB, ansonsten "0"
BYTEOFF-SET_SRC	Input	DINT	E, A, M, D	Adresse im Quellspeicherbereich, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
AREA_DEST	Input	BYTE	E, A, M, D	Folgende Bereiche können Sie im Zielspeicherbereich auswählen: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB
DBNUM-BER_DEST	Input	DINT, DB_ANY [1]	E, A, M, D	Nummer des Datenbausteins im Zielspeicherbereich, wenn AREA = DB, ansonsten "0" Es werden nur die 16 niederstwertigen Bits verwendet.
BYTEOFF-SET_DEST	Input	DINT	E, A, M, D	Adresse im Zielspeicherbereich, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
COUNT	Input	DINT	E, A, M, D	Anzahl der Byte, die kopiert werden
[1] Die Datentypen der Parameter DBNUMBER_SRC und DBNUMBER_DEST müssen den gleichen Datentyp haben. D. h., dass beide Variablen entweder vom Datentyp DINT oder vom Datentyp DB_ANY sein müssen.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Hinweis

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL POKE_BLK	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
AREA_SRC := "Tag_Source_Area"	// Bereich Datenbaustein im Quellspeicherbereich ausgewählt
DBNUMBER_SRC := "Tag_Source_DBNumber"	// Nummer des Datenbaustein im Quellspeicherbereich
BYTEOFFSET_SRC := "Tag_Source_Byte"	// Adresse im Quellspeicherbereich, an der ausgelesen wird
AREA_DEST := "Tag_Source_Area"	// Bereich Datenbaustein im Zielspeicherbereich ausgewählt
DBNUMBER_DEST := "Tag_Source_DBNumber"	// Nummer des Datenbaustein im Zielspeicherbereich
BYTEOFFSET_DEST := "Tag_Source_Byte"	// Adresse im Zielspeicherbereich, an der ausgelesen wird
COUNT := "Tag_Count"	// Anzahl der kopierten Byte

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA_SRC	Tag_Source_Area	16#84
DBNUMBER_SRC	Tag_Source_DBNumber	5
BYTEOFFSET_SRC	Tag_Source_Byte	20
AREA_DEST	Tag_Destination_Area	16#83
DBNUMBER_DEST	Tag_Destination_DBNumber	0
BYTEOFFSET_DEST	Tag_Destination_Byte	30
COUNT	Tag_Count	100

Die Anweisung schreibt 100 Byte aus dem Datenbaustein "5" beginnend bei der Adresse "20" in den Merker-Speicherbereich beginnend bei der Adresse "30".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

READ_LITTLE: Daten im Little-Endian-Format lesen

Beschreibung

Mit der Anweisung "Daten im Little-Endian-Format lesen" lesen Sie Daten aus einem Speicherbereich aus und schreiben diese in eine einzelne Variable in der Byte-Reihenfolge Little-Endian. Beim Little-Endian-Format wird das Byte mit den niederstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_ARRAY und DEST_VARIABLE sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter DEST_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter SRC_ARRAY zeigt auf den Speicherbereich, aus dem gelesen wird, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Lesen begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL lesen

Wenn Sie eine Variable lesen möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL lesen möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Daten im Little-Endian-Format lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_AR-RAY	Input	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, aus dem gelesen wird
OUT	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST_VARI-ABLE	Output	Bitfolgen, Ganz-zahlen, Gleitpunkt-zahlen, Zeiten, LDT, TOD, LTOD, DATE, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L	Gelesener Wert
POS	InOut	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Lesen begonnen wird. Der Parameter POS wird nullbasiert berechnet.

Parameter OUT

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters OUT:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL READ_LITTLE	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
SRC_ARRAY := #SourceField	// Die Ganzzahl 1_295_788_826 wird aus dem Speicherbereich #SourceField ausgelesen.
OUT := "Tag_Error"	// Fehlerinformation
DEST_VARIABLE := #DINTVari-able	// Die Ganzzahl wird im Little-Endian-Format in den Operanden #DINTVariable geschrieben. Der Datentyp am Parameter DEST_VARIABLE gibt an, wie viele Byte ausgelesen werden.
POS := #TagPos	// Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_ARRAY	#SourceField	ARRAY[0..3] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
POS	#TagPos	0 => 4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

WRITE_LITTLE: Daten im Little-Endian-Format schreiben

Beschreibung

Mit der Anweisung "Daten im Little-Endian-Format schreiben" schreiben Sie die Daten einer einzelnen Variable in der Byte-Reihenfolge Little-Endian in einen Speicherbereich. Beim Little-Endian-Format wird das Byte mit den niederstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_VARIABLE und DEST_ARRAY sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter SRC_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter DEST_ARRAY zeigt auf den Speicherbereich, in den die Daten geschrieben werden, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Schreiben begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL schreiben

Wenn Sie eine Variable schreiben möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL schreiben möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Daten im Little-Endian-Format schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_VARIABLE	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, LDT, TOD, LTOD, DATE, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L	Variable, deren Daten geschrieben werden
OUT	Output	INT	E, A, M, D, L	Fehlerinformation
DEST_ARRAY	InOut	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, in den die Daten geschrieben werden
POS	InOut	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Schreiben begonnen wird. Der Parameter POS wird nullbasiert berechnet.

Parameter OUT

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters OUT:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL WRITE_LITTLE	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
SRC_VARIABLE := #DINTVariable	// Die Ganzzahl 1_295_788_826 aus dem Operanden #DINTVariable wird im Little-Endian-Format in den Speicherbereich #TargetField geschrieben. Der Datentyp am Parameter SRC_VARIABLE gibt an, wie viele Byte geschrieben werden.
OUT := "Tag_Error"	// Fehlerinformation
DEST_ARRAY := #TargetField	// Speicherbereich

AWL	Erläuterung
POS := #TagPos	// Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

READ_BIG: Daten im Big-Endian-Format lesen

Beschreibung

Mit der Anweisung "Daten im Big-Endian-Format lesen" lesen Sie die Daten aus einem Speicherbereich aus und schreiben diese in eine einzelne Variable in der Byte-Reihenfolge Big-Endian. Beim Big-Endian-Format wird das Byte mit den höchstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_ARRAY und DEST_VARIABLE sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter DEST_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter SRC_ARRAY zeigt auf den Speicherbereich, aus dem gelesen wird, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Lesen begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL lesen

Wenn Sie eine Variable lesen möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL lesen möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Daten im Big-Endian-Format lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_ARRAY	Input	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, aus dem gelesen wird
OUT	Output	INT	E, A, M, D, L	Fehlerinformation
DEST_VARIABLE	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, LDT, TOD, LTOD, DATE, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L	Gelesener Wert
POS	InOut	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Lesen begonnen wird. Der Parameter POS wird nullbasiert berechnet.

Parameter OUT

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters OUT:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL READ_BIG	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klapplis-te "???" aus.
SRC_ARRAY := #SourceField	// Die Ganzzahl 439_041_101 wird aus dem Speicherbe-reich #SourceField ausgelesen.
OUT := "Tag_Error"	// Fehlerinformation
DEST_VARIABLE := #DINTVari-able	// Die Ganzzahl wird im Big-Endian-Format in den Operan-den #DINTVariable geschrieben. Der Datentyp am Parame-ter DEST_VARIABLE gibt an, wie viele Byte ausgelesen werden.
POS := #TagPos	// Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_ARRAY	#SourceField	ARRAY[0..10] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
POS	#TagPos	0 => 4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

WRITE_BIG: Daten im Big-Endian-Format schreiben

Beschreibung

Mit der Anweisung "Daten im Big-Endian-Format schreiben" schreiben Sie die Daten einer einzelnen Variable in der Byte-Reihenfolge Big-Endian in einen Speicherbereich. Beim Big-Endian-Format wird das Byte mit den höchstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_VARIABLE und DEST_ARRAY sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter SRC_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter DEST_ARRAY zeigt auf den Speicherbereich, in den die Daten geschrieben werden, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Schreiben begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL schreiben

Wenn Sie eine Variable schreiben möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL schreiben möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Daten im Big-Endian-Format schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_VARIABLE	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen,Zeiten, LDT, TOD, LTOD, DATE, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L	Variable, deren Daten geschrieben werden
OUT	Output	INT	E, A, M, D, L	Fehlerinformation
DEST_ARRAY	InOut	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, in den die Daten geschrieben werden
POS	InOut	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Schreiben begonnen wird. Der Parameter POS wird nullbasiert berechnet.

Parameter OUT

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters OUT:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL WRITE_BIG	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.
SRC_VARIABLE := #DINTVariable	// Die Ganzzahl 439_041_101 aus dem Operanden #DINTVariable wird im Big-Endian-Format in den Speicherbereich #TargetField geschrieben. Der Datentyp am Parameter SRC_VARIABLE gibt an, wie viele Byte geschrieben werden.
OUT := "Tag_Error"	// Fehlerinformation
DEST_ARRAY := #TargetField	// Speicherbereich
POS := #TagPos	// Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

VARIANT

VariantGet: Wert einer VARIANT-Variable auslesen

Beschreibung

Mit der Anweisung "Wert einer VARIANT-Variable auslesen" lesen Sie den Wert der Variable aus, auf die der VARIANT am Parameter SRC zeigt, und schreiben diesen in die Variable am Parameter DST.

Der Parameter SRC hat den Datentyp VARIANT. Am Parameter DST kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable, die am Parameter DST angegeben ist, muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert einer VARIANT-Variable auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die ausgelesen wird
DST	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL VariantGet	// Die Anweisung wird aufgerufen
SRC := #TagIn_Source	// Der Wert der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen.
DST := "TagOut_Dest"	// Der Inhalt der Variable wird in den Operand "TagOut_Dest" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu AWL (Seite 7557)

VariantPut: Wert in eine VARIANT-Variable schreiben

Beschreibung

Mit der Anweisung "Wert in eine VARIANT-Variable schreiben" schreiben Sie den Wert der Variable, die am Parameter SRC angegeben ist, in den Speicher am Parameter DST, auf den der VARIANT zeigt.

Der Parameter DST hat den Datentyp VARIANT. Am Parameter SRC kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable am Parameter SRC muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert in eine VARIANT-Variable schreiben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L, P	Variable, die ausgelesen wird
DST	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL VariantPut	// Die Anweisung wird aufgerufen.
SRC := "TagIn_Source"	// Der Wert des Operanden "TagIn_Source" wird ausgelesen...
DST := #TagIn_Dest	// und in die Variable, auf die der VARIANT am Operand #TagIn_Dest zeigt, geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum VARIANT (Seite 339)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1398)

Grundlagen zu AWL (Seite 7557)

CountOfElements: Anzahl der ARRAY-Elemente abfragen

Beschreibung

Mit der Anweisung "Anzahl der ARRAY-Elemente abfragen" fragen Sie ab, wie viele ARRAY-Elemente eine Variable hat, auf die der VARIANT zeigt.

Wenn es sich um ein eindimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl der ARRAY-Elemente ausgegeben. (Die Differenz zwischen dem oberen und dem unteren Grenzwert + 1). Wenn es sich um ein mehrdimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl aller Dimensionen ausgegeben.

Wenn Sie die Elemente eines ARRAY-DB abfragen möchten, dann sollten Sie die Anweisungen "ReadFromArrayDB" bzw. "WriteFromArrayDB" verwenden, da hier eine genauere Fehlerauswertung für die Anzahl der Elemente möglich ist.

Hinweis

Instanzen

Der VARIANT-Zeiger kann auf keine Instanz und damit auch auf keine Multiinstanz oder ein ARRAY of Multiinstanzen zeigen.

Hinweis

ARRAY innerhalb eines Datenbausteins

Wenn Sie die Anzahl der Elemente eines ARRAYs, das in einem Datenbaustein liegt, abfragen möchten, dann darf bei diesem Datenbaustein nicht das Bausteinattribut "Datenbaustein im Gerät schreibgeschützt" aktiviert sein. Ansonsten liefert der Parameter RET_VAL das Ergebnis "0", unabhängig davon, wie viele Elemente das ARRAY enthält.

Das Ergebnis ist ebenfalls "0", wenn die VARIANT-Variable kein ARRAY ist.

Wenn der VARIANT auf ein ARRAY of BOOL zeigt, dann werden auch die Füllelemente mitgezählt. (Z. B. bei einem ARRAY[0..1] of BOOL wird 8 zurückgegeben).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der ARRAY-Elemente abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die abgefragt wird
RET_VAL	Output	UDINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL CountOfElements`	`// Die Anweisung wird aufgerufen.`
`IN := #TagIn_Source`	`// Die Anzahl der ARRAY-Elemente der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen...`
`RET_VAL := "TagOut_RetVal"`	`// und am Operand "TagOut_RetVal" ausgegeben.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu AWL (Seite 7557)

ARRAY[*]

LOWER_BOUND: Untere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, untere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Untere ARRAY-Grenze auslesen" zur Verfügung.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Untere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, untere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, untere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL LOWER_BOUND`	`// Die Anweisung wird aufgerufen.`
`ARR := #ARRAY_A`	`// Die variable, untere Grenze des ARRAYs #ARRAY_A soll ausgelesen werden.`
`DIM := 2`	`// Die variable, untere Grenze der zweiten Dimension soll ausgelesen werden.`
`OUT := "Result"`	`// Ergebnis der Anweisung`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu AWL (Seite 7557)

UPPER_BOUND: Obere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, obere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Obere ARRAY-Grenze auslesen" zur Verfügung.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Obere ARRAY-Grenze auslesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, obere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, obere Grenze ausgelesen werden soll.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL UPPER_BOUND`	`// Die Anweisung wird aufgerufen.`
`ARR := #ARRAY_A`	`// Die variable, obere Grenze des ARRAYs #ARRAY_A soll ausgelesen werden.`
`DIM := 2`	`// Die variable, obere Grenze der zweiten Dimension soll ausgelesen werden.`
`OUT := "Result"`	`// Ergebnis der Anweisung`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu AWL (Seite 7557)

Legacy

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

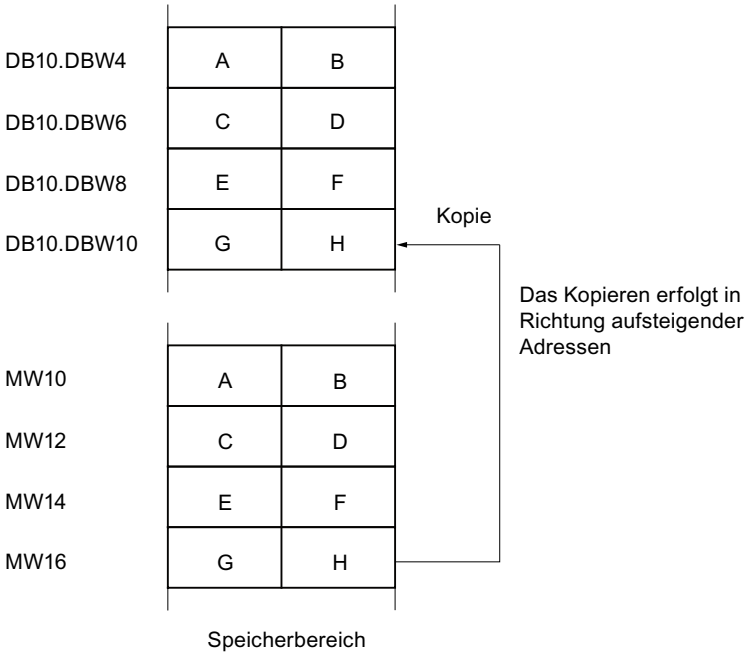

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL BLKMOV	// Die Anweisung "Bereich kopieren" wird aufgerufen.
SRCBLK := P#M100.0 BYTE 10	// 10 Bytes ab MB100 werden kopiert.
RET_VAL := "Tag_ErrorCode"	// Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in die Variable "Tag_ErrorCode" ausgegeben.
DSTBLK := P#M200.0 BYTE 10	// Die kopierten Bytes werden in die 10 Bytes ab MB200 geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Grundlagen zu AWL (Seite 7557)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL UBLKMOV	// Die Anweisung "Bereich ununterbrechbar kopieren" wird aufgerufen.
SRCBLK := P#M100.0 BYTE 10	// 10 Bytes ab MB100 werden kopiert.
RET_VAL := "Tag_ErrorCode"	// Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in die Variable "Tag_ErrorCode" ausgegeben.
DSTBLK := P#M200.0 BYTE 10	// Die kopierten Bytes werden in die 10 Bytes ab MB200 geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Grundlagen zu AWL (Seite 7557)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

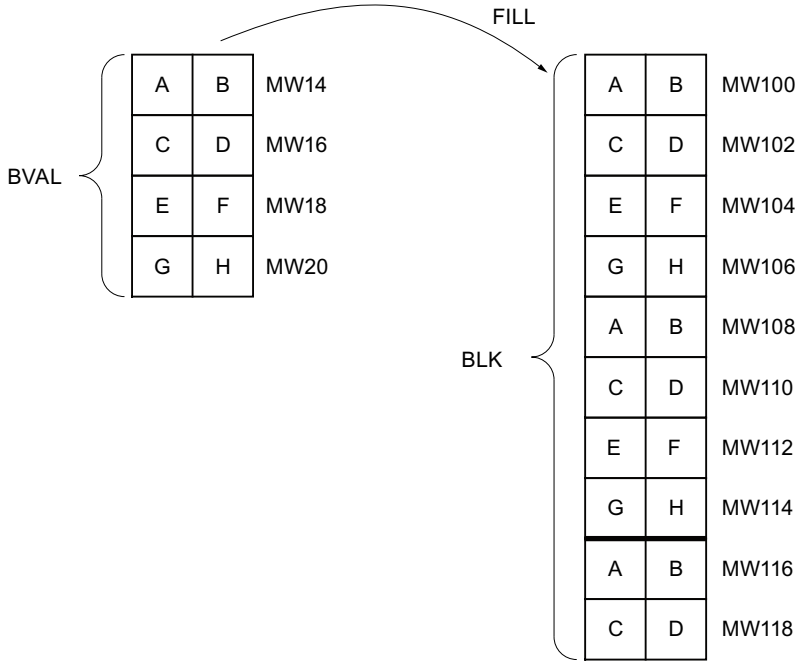

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird
1) Der Parameter BLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter BVAL werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter BLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL FILL                    // Die Anweisung "Bereich befüllen" wird aufgerufen.
BVAL := P#M14.0 WORD 4       // Der Bereich von MW14 bis MW20 wird kopiert.
RET_VAL := "Tag_ErrorCode"   // Fehlerinformationen
```

AWL	Erläuterung
BLK := P#M100.0 WORD 10	// Der Speicherbereich von MW100 bis MW118 wird mit dem Inhalt der 4 Wörter befüllt, die im Speicherbereich am Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Grundlagen zu AWL (Seite 7557)

Umwandler

SCALE_X: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Eingang VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung "Skalieren" wird der Gleitpunktwert am Eingang VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang RET_VAL abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:

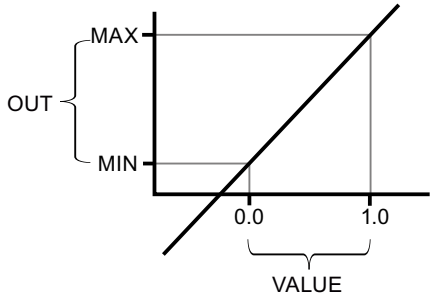

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [VALUE * (MAX – MIN)] + MIN

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Wert, der skaliert wird. Bei Angabe einer Konstante müssen Sie diese deklarieren.
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
RET_VAL	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Skalierung

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SCALE_X                 // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
MIN := "Tag_Minimum"         // Unterer Grenzwert
VALUE := "Tag_Value"         // Gleitpunktwert
MAX := "Tag_Maximum"         // Oberer Grenzwert
RET_VAL := "Tag_Result"      // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_Minimum	10
VALUE	Tag_Value	0.5
MAX	Tag_Maximum	30
OUT	Tag_Result	20

Der Wert am Eingang "Tag_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Eingängen "Tag_Minimum" und "Tag_Maximum" definiert wurde. Das Ergebnis wird am Ausgang "Tag_Result" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Globale Konstanten deklarieren (Seite 7392)

Grundlagen zu AWL (Seite 7557)

NORM_X: Normieren

Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang RET_VAL als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert der Ausgang OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem am Eingang MAX ist, liefert der Ausgang OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:

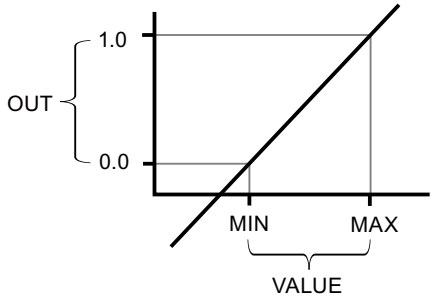

Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

OUT = (VALUE – MIN) / (MAX – MIN)

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Normieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Untere Grenze des Wertebereichs
VALUE [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Wert, der normiert wird
MAX [1]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Obere Grenze des Wertebereichs
RET_VAL	Output	Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Normierung
[1] Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren.				

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL NORM_X                  // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
MIN := "Tag_Minimum"         // Unterer Grenzwert
VALUE := "Tag_Value"         // Gleitpunktwert
MAX := "Tag_Maximum"         // Oberer Grenzwert
RET_VAL := "Tag_Result"      // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_Minimum	10
VALUE	Tag_Value	20
MAX	Tag_Maximum	30
RET_VAL	Tag_Result	0.5

Der Wert am Eingang "Tag_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Eingängen "Tag_Minimum" und "Tag_Maximum" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Eingang "Tag_Value" normiert. Das Ergebnis wird am Ausgang "Tag_Result" als Gleitpunktzahl abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Globale Konstanten deklarieren (Seite 7392)

Grundlagen zu AWL (Seite 7557)

VARIANT

VARIANT_TO_DB_ANY: VARIANT in DB_ANY konvertieren

Beschreibung

Mit der Anweisung "VARIANT in DB_ANY konvertieren" fragen Sie die Datenbaustein-Nummer ab, die der Operand, der am Parameter IN angegeben ist, adressiert. Dabei kann es sich um einen Instanz-Datenbaustein oder um einen ARRAY-Datenbaustein handeln. Der Operand am Parameter IN hat den Datentyp VARIANT, daher muss zur Zeit der Programmerstellung nicht bekannt sein, welchen Datentyp der Datenbaustein hat, dessen Nummer abgefragt werden soll. Die Datenbaustein-Nummer wird zur Laufzeit ausgelesen und in den, am Parameter RET_VAL angegebenen, Operanden geschrieben.

Voraussetzungen

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind, dann wird "0" als Datenbausteinnummer ausgegeben.

Die Ausgangsvariable...	referenziert...	Konvertierungsmöglichkeiten
VARIANT	... einen Datenbaustein, der ein Instanz-Datenbaustein eines PLC-Datentyps oder eines Systemdatentyps (SDT) ist.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.
VARIANT	... einen Datenbaustein, der ein ARRAY-DB ist.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.

Die Ausgangsvariable...	referenziert...	Konvertierungsmöglichkeiten
VARIANT	... eine Variable mit einem elementaren Datentyp.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da ein Datenbaustein keinesfalls aus nur einem elementaren Datentyp besteht.
VARIANT	... eine Struktur innerhalb eines Datenbausteins.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da es sich nur um einen Teil innerhalb des Datenbausteins handelt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VARIANT in DB_ANY konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die ausgelesen wird. (Der Funktionswert der Anweisung "DB_ANY_TO_VARIANT"). Am Parameter IN können Sie eine lokale oder eine globale Variable verwenden.
RET_VAL	Output	DB_ANY	E, A, M, D, L	Ergebnis: Nummer des DBs
ERR	Output	INT	E, A, M, D, L	Fehlerinformationen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
252C	Der Datentyp VARIANT am Parameter IN liefert den Wert "0" und die CPU geht in den Betriebszustand STOP.
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8131	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8132	Der Datenbaustein ist zu kurz und kein ARRAY-Datenbaustein.
8150	Der Datentyp VARIANT am Parameter IN liefert den Wert "0". Um diese Fehlermeldung zu erhalten, muss die Bausteineigenschaft "Lokale Fehlerbehandlung im Baustein" aktiviert sein. Ansonsten geht die CPU in den Betriebszustand STOP und gibt den Fehlercode 16#252C aus.

Fehler-code* (W#16#...)	Erläuterung
8153	Der Datentyp VARIANT am Parameter IN zeigt nicht auf den Anfang eines ARRAY-Datenbausteins oder die Länge des VARIANT stimmt nicht mit der des Datenbausteins überein.
8154	Der Datenbaustein hat den falschen Datentyp.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                                      Erläuterung
CALL VARIANT_TO_DB_ANY                   // Die Anweisung wird aufgerufen.
IN := #tempVARIANT                       // Die Nummer eines Datenbausteins, der
                                         am Operand tempVARIANT angegeben ist,
                                         wird ausgelesen. Da der Operand den Daten-
                                         typ VARIANT hat, muss zur Programmerstel-
                                         lung noch nicht bekannt sein, welchen Da-
                                         tentyp die Variable hat.
RET_VAL := "OutputDBNumber"              // Die Nummer wird in die Variable "Out-
                                         putDBNumber" geschrieben, die den Daten-
                                         typ DB_ANY hat.
ERR := "Tag_Error"                       // Fehlerinformation
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
IN	Input	tempVARIANT	-
RET_VAL	Output	OutputDBNumber	11

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum VARIANT (Seite 339)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu AWL (Seite 7557)

DB_ANY_TO_VARIANT: DB_ANY in VARIANT konvertieren

Beschreibung

Mit der Anweisung "DB_ANY in VARIANT konvertieren" erzeugen Sie aus einem Datenbaustein eine VARIANT-Variable, die die unten stehenden Voraussetzungen erfüllt. Der Operand am Parameter IN hat den Datentyp DB_ANY, daher muss zur Zeit der Programmerstellung der Datenbaustein nicht bekannt sein. Die Datenbaustein-Nummer wird zur Laufzeit ausgelesen.

Voraussetzungen

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind oder der Datenbaustein nicht existiert, dann wird am Parameter RET_VAL der Wert NULL ausgegeben. Alle weiteren Zugriffe mit der Variable RET_VAL schlagen fehl.

Die Eingangsvariable vom Datentyp ...	referenziert...	Konvertierungsmöglichkeiten
DB_ANY	...einen Datenbaustein, der ein Instanz-Datenbaustein eines PLC-Datentyps oder eines Systemdatentyps (SDT) ist.	Die Konvertierung ist möglich
DB_ANY	...einen Datenbaustein, der ein ARRAY-DB ist.	Die Konvertierung ist möglich
DB_ANY	...einen Datenbaustein, der ein Instanz-Datenbaustein eines Funktionsbausteins oder ein globaler Datenbaustein ist.	Die Konvertierung ist nicht möglich

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DB_ANY in VARIANT konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DB_ANY	E, A, M, D, L	Datenbaustein, dessen Nummer ausgelesen wird. Am Parameter IN können Sie eine lokale oder eine globale Variable verwenden.
RET_VAL	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Nummer des Datenbausteins
ERR	Output	INT	E, A, M, D, L	Fehlerinformationen
1) Der Parameter RET_VAL ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8130	Die Nummer des Datenbaustein ist "0".
8131	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8132	Der Datenbaustein ist zu kurz und kein ARRAY-Datenbaustein.
8134	Der Datenbaustein ist schreibgeschützt.
8154	Der Datenbaustein hat den falschen Datentyp.
8155	Der Datenbaustein hat einen unbekannten Datentyp. [1]
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

[1] Die Ursache für die Ausgabe des Fehlercodes #8155 liegt darin:

Es wurde ein PLC-Datentyp (UDT1) deklariert und anschließend ein Datenbaustein (DB2) vom Datentyp "UDT1" erstellt. In der Variablentabelle gibt es eine Variable (3) vom Datentyp DB_ANY. In einem Programmbaustein (4) wurde anschließend die Anweisung "DB_ANY_TO_VARIANT" aufgerufen und am Parameter IN mit der Variable (3) versorgt. Beim Ausführen liefert die Anweisung "DB_ANY_TO_VARIANT" den Fehlercode 16#8155.

Zur Auflösung des Fehlercodes gehen Sie folgendermaßen vor:

1. Legen Sie eine Funktion (FC5) an und deklarieren Sie an der InOut-Schnittstelle eine Variable vom Datentyp VARIANT.
2. Legen Sie eine weitere Funktion (FC6) an und rufen Sie darin den FC5 auf.
3. Legen Sie im FC6 in der Temp-Schnittstelle eine Variable (7) vom Datentyp "UDT1" an.
4. Versorgen Sie die InOut-Schnittstelle des FC5 mit der Variable (7).
5. Übersetzen und laden Sie die beiden Bausteine (FC5 und FC6) in Ihre CPU. Sie brauchen diese Bausteine (FC5 und FC6) nicht im Anwenderprogramm aufzurufen.

Ergebnis:

Der Fehlercode 16#8155 wird nicht mehr ausgegeben, da der Datentyp jetzt dem Anwenderprogramm bekannt ist.

Diese Vorgehensweise ist nicht notwendig, wenn Sie bereits nach dem Aufruf der Anweisung "DB_ANY_TO_VARIANT" eine der beiden Anweisungen "VariantGet" oder "VariantPut" aufrufen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                                  Erläuterung
CALL DB_ANY_TO_VARIANT               // Die Anweisung wird aufgerufen.
```

AWL	Erläuterung
IN := "InputDB"	// Die Nummer eines beliebigen Datenbausteins, der am Operand "InputDB" angegeben ist, wird genutzt um eine Variable vom Datentyp VARIANT zu erzeugen, die den Datenbaustein adressiert. Da der Operand am Parameter IN den Datentyp DB_ANY hat, muss zur Programmerstellung noch nicht bekannt sein, welcher Datenbaustein zur Laufzeit verwendet wird, weder der Name noch die Nummer des Datenbausteins.
RET_VAL := #tempVARIANT	// Da der Operand am Parameter RET_VAL den Datentyp VARIANT hat, muss zur Programmerstellung noch nicht bekannt sein, welchen Datentyp der Datenbaustein hat.
ERR := "Tag_Error"	// Fehlerinformation

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
IN	Input	InputDB	11
RET_VAL	Temp	tempVARIANT	-

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zum VARIANT (Seite 339)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu AWL (Seite 7557)

Legacy

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
allgemeine Fehlerinfor-mation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SCALE                   // Die Anweisung wird aufgerufen.
IN := "Tag_InputValue"       // Gibt den Wert an, der umgewandelt und skaliert wird.
HI_LIM := "Tag_HighLimit"    // Oberer Grenzwert
LO_LIM := "Tag_LowLimit"     // Unterer Grenzwert
BIPOLAR := "Tag_Bipolar"     // Gibt an, ob der Wert am Parameter IN als bipolar
                             oder unipolar interpretiert wird.
RET_VAL := "Tag_ErrorCode"   // Fehlerinformationen
OUT := "Tag_OutputValue"     // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu AWL (Seite 7557)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandelt sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL UNSCALE                 // Die Anweisung wird aufgerufen.
IN := "Tag_InputValue"       // Gibt den Wert an, der deskaliert wird.
HI_LIM := "Tag_HighLimit"    // Oberer Grenzwert
LO_LIM := "Tag_LowLimit"     // Unterer Grenzwert
BIPOLAR := "Tag_Bipolar"     // Gibt an, ob der Wert am Parameter IN als bipolar
                             oder unipolar interpretiert wird.
```

```
AWL                              Erläuterung
RET_VAL := "Tag_ErrorCode"       // Fehlerinformationen
OUT := "Tag_OutputValue"         // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu AWL (Seite 7557)

Programmsteuerung

Laufzeitsteuerung

ENDIS_PW: Passwort-Legitimierung einschränken und freigeben

Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU projektierte Passwörter legitimiert sind oder nicht. Dadurch können Sie legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist.

Wenn Sie die Anweisung aufrufen und der Parameter REQ den Signalzustand "0" hat, wird der aktuell eingestellte Zustand an den Ausgangsparametern angezeigt. Falls Änderungen an den Eingangsparametern vorgenommen wurden, werden diese nicht an die Ausgangsparameter übergeben.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F_PWD, FULL_PWD, R_PWD, HMI_PWD) übernommen:

- Wenn der Signalzustand "0" vorliegt, ist die Legitimation per Passwort nicht erlaubt.
- Wenn der Signalzustand "1" vorliegt, ist das Passwort verwendbar.

Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer das Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Nicht konfigurierte Passwörter müssen am Eingang den Signalzustand TRUE haben und liefern den Signalzustand TRUE am Ausgang. Das Failsafe Passwort ist nur für eine F-CPU parametrierbar und muss daher in einer Standard CPU immer mit dem Signalzustand TRUE verschaltet werden. Wenn die Anweisung einen Fehler zurückliefert, dann bleibt der Aufruf ohne Auswirkung, d. h. die vorherige Sperre ist unverändert wirksam.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken einer leeren Speicherkarte (Übertragungskarte oder Programmkarte) in eine S7-1200-CPU.
- Der Übergang von NETZ AUS - NETZ EIN deaktiviert in der S7-1200-CPU den Schutz. Die Anweisung "Passwort-Legitimierung einschränken und freigeben" muss im Programm erneut (z. B. im Anlauf-OB) aufgerufen werden.

Hinweis

Wenn das HMI Passwort nicht freigegeben ist, dann sperrt die Anweisung "Passwort-Legitimierung einschränken und freigeben" den Zugriff von HMI-Systemen.

Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

Ungewolltes Aussperren bei einer S7-1500-CPU verhindern

Die Einstellungen können im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Ungewolltes Aussperren bei einer S7-1200-CPU verhindern

Da es bei einer S7-1200-CPU keinen Betriebsartenschalter gibt, wird der Schutz bei NETZ AUS - NETZ EIN deaktiviert. Damit ist es möglich und empfehlenswert, ein ungewolltes Aussperren mithilfe bestimmter Programmsequenzen innerhalb Ihres Programms zu verhindern.

Programmieren Sie dazu eine Zeitsteuerung entweder über einen Weckalarm-OB oder einen Timer im Main OB (OB 1). Dadurch haben Sie die Möglichkeit, nach einem Übergang von NETZ AUS - NETZ EIN und der damit verbundenen Deaktivierung des Schutzes, die Anweisung "Passwort-Legitimierung einschränken und freigeben" in dem entsprechenden OB (z. B. OB 1 oder OB 35) zeitnah wieder aufzurufen. Um das Zeitfenster, in dem die Anweisung nicht aktiv ist und damit keine Einschränkungen bei der Passwort-Legitimierung bestehen, möglichst klein zu halten, rufen Sie die Anweisung im Anlauf-OB (OB 100) auf. Diese Vorgehensweise bietet Ihnen den größtmöglichen Schutz vor unbefugtem Zugriff.

Wenn es zu einer ungewollten Aussperrung gekommen ist, dann können Sie den Aufruf im Anlauf-OB überspringen (z. B. durch die Abfrage eines Eingangsparameters) und haben die eingestellte Zeit (z. B. 10 Sekunden bis 1 Minute), um eine Verbindung zur CPU aufzubauen, bevor die Sperre wieder aktiv ist.

Wenn Sie in Ihrem Programmcode keinen Timer vorgesehen haben und es zu einer Aussperrung kam, dann stecken Sie eine leere Übertragungskarte oder eine leere Programmkarte in die CPU. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Danach müssen Sie das Anwenderprogramm aus STEP 7 erneut in die CPU laden.

Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte oder Programmkarte. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

⚠ WARNUNG

Leere Übertragungskarte stecken

Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Nach dem Ziehen der Übertragungskarte ist der Inhalt der Übertragungskarte im internen Ladespeicher. Achten Sie an dieser Stelle darauf, dass die Karte kein Programm enthält.

⚠ WARNUNG

Leere Programmkarte stecken

Wenn Sie eine Programmkarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Achten Sie darauf, dass die Programmkarte leer ist. Der interne Ladespeicher wird auf die leere Programmkarte kopiert. Nach dem Ziehen der ehemals leeren Programmkarte ist der interne Ladespeicher leer.

Sie müssen die Übertragungskarte oder Programmkarte ziehen, bevor Sie die CPU in RUN versetzen.

Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

Aktion	Passwortschutz durch die Anweisung
Grundzustand nach • Betriebsartenschalter auf STOP • Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control) • Zurücksetzen auf Werkseinstellung	Nicht aktiviert (Keine Einschränkungen)
Grundzustand nach NETZ-EIN	• S7-1200-CPU: Die Sperre ist deaktiviert und die Anweisung muss im Programm erneut aufgerufen werden (z. B. im Anlauf-OB). • S7-1500-CPU: Aktiviert (wenn vor NETZ AUS eine Sperre aktiviert war). Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent.
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT	Aktiviert Die Passwörter dürfen weiterhin nicht verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.
F_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff inklusive Failsafe • F_PWD = "0": Passwort nicht erlauben • F_PWD = "1": Passwort erlauben
FULL_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff • FULL_PWD = "0": Passwort nicht erlauben • FULL_PWD = "1": Passwort erlauben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lesezugriff • R_PWD = "0": Passwort nicht erlauben • R_PWD = "1": Passwort erlauben
HMI_PWD	Input	BOOL	E, A, M, D, L oder Konstante	HMI-Zugriff • HMI_PWD = "0": Passwort nicht erlauben • HMI_PWD = "1": Passwort erlauben
F_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff inklusive Failsafe • F_PWD_ON = "0": Passwort nicht erlaubt • F_PWD_ON = "1": Passwort erlaubt
FULL_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff • FULL_PWD_ON = "0": Passwort nicht erlaubt • FULL_PWD_ON = "1": Passwort erlaubt
R_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lesezugriff • R_PWD_ON = "0": Passwort nicht erlaubt • R_PWD_ON = "1": Passwort erlaubt
HMI_PWD_ON	Output	BOOL	E, A, M, D, L	Status HMI-Zugriff • HMI_PWD_ON = "0": Passwort nicht erlaubt • HMI_PWD_ON = "1": Passwort erlaubt
RET_VAL	Output	WORD	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8090	Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt

Fehler-code* (W#16#...)	Erläuterung
80D0	Das Passwort für Failsafe ist nicht konfiguriert. Bei Standard-CPUs muss der Signalzustand TRUE sein.
80D1	Der Lese-/Schreibzugriff ist nicht konfiguriert
80D2	Der Lesezugriff ist nicht konfiguriert
80D3	Der HMI-Zugriff ist nicht konfiguriert
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

Globale Konstanten deklarieren (Seite 7391)

PC-Systeme

SHUT_DOWN: Zielsystem herunterfahren

Beschreibung

Mit der Anweisung “SHUT_DOWN: Zielsystem herunterfahren” fahren Sie das PC-basierte Automatisierungssystem herunter, starten den S7-Software Controller CPU 150xS oder Windows auf dem PC-basierte Automatisierungssystem neu.

Die Anweisung finden Sie in der Task Card "Anweisungen" unter Einfache Anweisungen > Programmsteuerung > Laufzeitsteuerung.

Ein Neustart ist z. B. in den folgenden Situationen sinnvoll:

- Eine Industrie-USV (unterbrechungsfreie Stromversorgung) meldet über einen digitalen Eingang einen Spannungsausfall.
- Windows reagiert nicht mehr oder liefert einen „Blue Screen“.
- Im Anwenderprogramm werden zu viele Fehler-OBs aufgerufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zielsystem herunter fahren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	MODE = 1: CPU 150xS und Windows herunter fahren Die CPU geht in STOP und speichert die remanenten Daten. Anschließend werden CPU und Windows herunter gefahren. Das System muss manuell gestartet werden. MODE = 2: CPU 150xS neu starten Die CPU geht in STOP und speichert die remanenten Daten. Anschließend wird die CPU heruntergefahren und neu gestartet. MODE = 3: Windows neu starten. Die CPU bleibt in RUN. Windows wird neu gestartet (ab TIA Portal V14 ist MODE 3 nur für Abwärtskompatibilität gelassen. Es wird empfohlen, MODE 4 oder MODE 5 zu verwenden). MODE = 4: Windows wird korrekt heruntergefahren und neu gestartet. Die CPU bleibt im RUN. MODE = 5: Windows neu starten. (Vergleichbar mit MODE 3; Ausnahme: MODE 5 ist nur dann zu verwenden, wenn Windows abgestürzt ist).
COM-MENT	Input	STRING	E, A, M, D, L	Bei Mode = 1, 3 und 4 können Sie eine Begründung für den Neustart mitgeben. Die Begründung wird im Windows Event Log ausgegeben.
Ret_Val	Return	WORD	A, M, D, L	Ret_Val = 0: kein Fehler Ret_Val = 8090: Der übergebene Wert an MODE wird nicht unterstützt. Ret_Val = 8091: Windows reagiert nicht auf den Aufruf der Anweisung Shut_Down (gültig nur für Mode 3 und 4). Ret_Val = 8092: Bei Auftreten dieses Fehlers, wenden Sie sich bitte an den SIMATIC Customer Support (gültig nur für Mode 3 und 4).

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Wenn Sie die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufrufen, dann wird sie nicht ausgeführt und der Freigabeausgang ENO liefert den Signalzustand "0".

Die Anweisung läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Aufruf der Anweisung

Sie können die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

GET_ERROR: Fehler lokal abfragen

Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Programmbausteins ab. Dabei handelt es sich in der Regel um Programmier- oder Zugriffsfehler. Wenn das System während der Programmbausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler, innerhalb der Ausführung des Programmbausteins seit der letzten Ausführung der Anweisung, eine ausführliche Information im Operanden am Ausgang OUT gespeichert.

Am Ausgang OUT können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Treten im Programmbaustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehlerinformation zum nächsten aufgetretenen Fehler von der Anweisung ausgegeben.

Hinweis

Ausgang OUT

Der Ausgang OUT wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.

Hinweis

Aktivierung der lokalen Fehlerbehandlung

Sobald Sie die Anweisung in den Programmcode eines Programmbausteins einfügen, wird die lokale Fehlerbehandlung aktiviert und voreingestellte Systemreaktionen werden beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	ErrorStruct	D, L	Fehlerinformationen

Datentyp "ErrorStruct"

Der Datentyp "ErrorStruct" kann in einen globalen Datenbaustein oder in die Bausteinschnittstelle eingefügt werden. Sie können den Datentyp auch mehrfach einfügen, wenn Sie jedes Mal einen anderen Namen für die Datenstruktur vergeben. Die Datenstruktur und der Name einzelner Strukturelemente sind nicht änderbar. Wenn Sie die Fehlerinformation in einem globalen Datenbaustein speichern, kann sie auch von anderen Programmbausteinen ausgelesen werden.

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

Strukturkomponente		Datentyp	Beschreibung
ERROR_ID		WORD	Fehler-ID
FLAGS		BYTE	Zeigt, ob der Fehler während eines Programmbausteinaufrufs aufgetreten ist. 16#01: Fehler während eines Programmbausteinaufrufs 16#00: Kein Fehler während eines Programmbausteinaufrufs
REACTION		BYTE	Voreingestellte Reaktion: 0: Ignorieren (Schreibfehler) 1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler) 2: Anweisung überspringen (Systemfehler)
CODE_ADDRESS		CREF	Informationen zur Adresse und Art des Programmbausteins
	BLOCK_TYPE	BYTE	Programmbausteintyp, in dem der Fehler aufgetreten ist: 1: Organisationsbaustein (OB) 2: Funktion (FC) 3: Funktionsbaustein (FB)
	CB_NUMBER	UINT	Nummer des Codebausteins
	OFFSET	UDINT	Verweis auf den internen Speicher
MODE		BYTE	Informationen zur Adresse eines Operanden
OPERAND_NUMBER		UINT	Operandennummer des Maschinenbefehls
POINTER_NUMBER_LOCATION		UINT	(A) Interner Zeiger
SLOT_NUMBER_SCOPE		UINT	(B) Ablagebereich im internen Speicher
DATA_ADDRESS		NREF	Informationen zur Adresse eines Operanden
	AREA	BYTE	(C) Speicherbereich: L: 16#40...16#7F, 16#86, 16#87, 16#8E, 16#8F, 16#C0...16#FF E: 16#81 A: 16#82 M: 16#83 DB: 16#40, 16#84, 16#85, 16#8A, 16#8B PE: 16#01 PA: 16#02 Technologische Objekte: 16#04
	DB_NUMBER	UINT	(D) Nummer des Datenbausteins
	OFFSET	UDINT	(E) Relative Adresse des Operanden

Strukturkomponente "ERROR_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR_ID" ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
LABEL: L #Field[#index]      // Zugriff auf die Variable #Field[#index]
```

AWL	Erläuterung
L 40.5	// Wert 40.5 laden
*R	// Wert der Variable mit dem Wert 40.5 multiplizieren.
T #TagOut	// Produkt in den Operanden #TagOut übertragen.
CALL GET_ERROR	// Die Anweisung wird aufgerufen.
OUT := #Error	// Im Fehlerfall wird eine ausführliche Information im Operanden #Error gespeichert.
T #Error.REACTION	// Die Strukturkomponente #Error.REACTION wird ausgelesen.
L 1	// Wert 1 laden
==I	// Der Wert der Strukturkomponente wird mit dem Wert 1 verglichen.
SPB LABEL	// Wenn der Vergleich der beiden Werte positiv ist, dann springt die Programmbearbeitung zurück zur Sprungmarke LABEL.

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Operand #TagOut liefert trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler lokal abfragen" nach der Multiplikation zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler lokal abfragen" liefert, wird mithilfe eines Vergleichs ausgewertet. Wenn die Strukturkomponente #Error.REACTION den Wert "1" hat, dann handelt es sich um einen Lese-/Zugriffsfehler und die Programmbearbeitung beginnt wieder bei der Sprungmarke LABEL.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu AWL (Seite 7557)

GET_ERR_ID: Fehler-ID lokal abfragen

Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Dabei handelt es sich in der Regel um Zugriffsfehler. Wenn das System während der Bausteinbearbeitung Fehler innerhalb der Ausführung des Bausteins seit der letzten Ausführung der Anweisung meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang RET_VAL gespeichert.

Am Ausgang RET_VAL können nur Operanden vom Datentyp WORD angegeben werden. Treten im Baustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten

aufgetretenen Fehlers die Fehler-ID des nächsten aufgetretenen Fehlers von der Anweisung ausgegeben.

Hinweis

Der Ausgang RET_VAL wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.

Der Ausgang der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn eine Fehlerinformation vorliegt. Wenn diese Bedingung nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Ein Beispiel, wie Sie die Anweisung auch in Kombination mit weiteren Fehlerbehandlungsmöglichkeiten realisieren können, finden Sie unter "Siehe auch".

Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	WORD	E, A, M, D, L	Fehler-ID

Parameter RET_VAL

Die folgende Tabelle zeigt die Werte, die am Parameter RET_VAL ausgegeben werden können:

RET_VAL* (Hexadezimal)	RET_VAL* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L #Field[#index]             // Zugriff auf die Variable #Field[#index]
```

AWL	Erläuterung
L 40.5	// Wert 40.5 laden
*R	// Wert der Variable mit dem Wert 40.5 multiplizieren.
T #TagOut	// Produkt in den Operanden #TagOut übertragen.
CALL GET_ERR_ID	// Die Anweisung wird aufgerufen.
RET_VAL := #TagID	// Im Fehlerfall wird eine Fehler-ID im Operanden #TagID gespeichert.
T #TagID	// Die Fehler-ID am Operanden #TagID wird ausgelesen.
L 16#2522	// Wert 16#2522 laden
==I	// Der Wert der Fehler-ID wird mit dem Wert 16#2522 verglichen.
CALL MOVE	// Die Anweisung wird aufgerufen.
IN := 100.0	// Der Wert 100.0 wird in den Operanden #TagOut kopiert.
OUT := #TagOut	// Ausgang der Anweisung

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Operand #TagOut liefert trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler-ID lokal abfragen" nach der Multiplikation zu programieren um den Fehler abzufangen. Die Fehler-ID, die die Anweisung "Fehler-ID lokal abfragen" liefert, wird mithilfe eines Vergleichs ausgewertet. Wenn der Operand #TagID die ID 16#2522 liefert, dann handelt es sich um einen Lese-/Zugriffsfehler und der Wert "100.0" wird in den Ausgang #TagOut kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu AWL (Seite 7557)

INIT_RD: Alle remanenten Daten zurücksetzen

Beschreibung

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B5	Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL INIT_RD                 // Die Anweisung wird aufgerufen.
REQ := "Tag_REQ"             // Wenn der Operand "Tag_REQ" den Signalzustand "1"
                             führt, werden alle remanenten Daten zurückgesetzt.
RET_VAL := "Tag_RET_VAL"     // Fehlercode
```

Wenn der Operand "Tag_REQ" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1499)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu AWL (Seite 7557)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine parametrierte Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) parametrieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Beispiel zum Einfluss der geplanten Verzögerungszeit

Im folgenden Beispiel sehen Sie den Einfluss der Verzögerungszeit der Anweisung "WAIT" in verschiedenen Szenarien.

Das folgende Bild stellt die Szenarien schematisch dar:

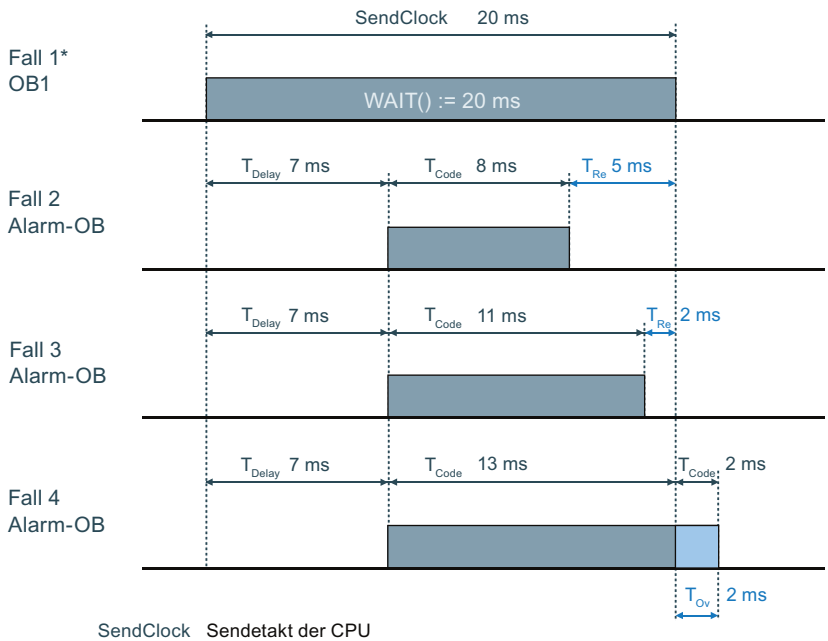

SendClock Sendetakt der CPU

T_{Delay} Verzögerungszeit des Alarm-OBs

T_{Code} Zeit der Programmausführung

T_{Re} Verbleibende Zeit

T_{Ov} Zeit der Überlänge, um welche sich der Sendetakt erhöht

* Der OB1 mit der Anweisung ”WAIT” ist ebenso Teil von Fall 2, 3 und 4 und muss mit berücksichtigt werden.

Verbleibende Zeit = Die Spanne vom Ende der geplanten Verzögerungszeit (via "WAIT") bis zur Beendigung des Alarm-OBs

Zeit der Überlänge = Die Spanne von der Beendigung des Alarm-OBs bis zum Ende der geplanten Verzögerungszeit (via "WAIT").

Fall 1:

Die Anweisung "WAIT" wird in einem OB1 aufgerufen. Die Anweisung "WAIT" kann von höher prioren OBs oder höher prioren Prozessen (z.B. System Threads) unterbrochen werden. Die Verzögerungszeit der Anweisung "WAIT" ist aber weder verändert noch aufgeschoben.

Fall 2 und Fall 3:

Die Verarbeitung des Programms im OB1 wird nach einer Zeitverzögerung von 20 ms fortgesetzt. Diese Zeitverzögerung wird durch den Aufruf der Anweisung "WAIT" im OB1 berechnet (siehe OB1 mit WAIT). Innerhalb dieser 20 ms kann ein Alarm-OB seinen eigenen Programmcode ausführen. Der Sendetakt der CPU wird nicht verändert.

Fall 4:

Die Verarbeitung des Programms im OB1 wird nach Beendingung des höher prioren Prozesses fortgesetzt. Die 20 ms Verzögerungszeit im OB1 sind verstrichen, aber der höher priore Prozess muss noch beendet werden. Der Sendetakt der CPU wird erhöht.

Hinweis

Ausführungsreihenfolge von System- oder Kommunikationsprozessen (System Threads)

Die System Threads verwenden meistens die Priorität "15". Ebenso gibt es System Threads mit einer höheren Priorität als "26", doch diese Prozesse haben eine geringere CPU-Auslastung. System Threads werden im Bild nicht dargestellt.

Laufzeitmessung des OB1 mit Hilfe der Anweisung "RT_INFO":

Fall 2: 20 ms - 8 ms - System Threads = <12 ms. Sendetakt: ~20 ms.

Fall 3: 20 ms - 11 ms - System Threads - <9 ms. Sendetakt: ~20 ms.

Fall 4: 20 ms - 15 ms - System Threads - <7 ms. Sendetakt: ~22 ms.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

RUNTIME: Laufzeitmessung

Beschreibung

Mit der Anweisung "Laufzeitmessung" messen Sie die Laufzeit des gesamten Programms, einzelner Bausteine oder Befehlssequenzen.

Wenn Sie die Laufzeit Ihres gesamten Programms messen wollen, dann rufen Sie die Anweisung "Laufzeitmessung" im OB1 auf. Der Startpunkt der Laufzeitmessung wird mit dem ersten Aufruf gesetzt und der Ausgang RET_VAL liefert nach dem zweiten Aufruf die Laufzeit des Programms. Die gemessene Laufzeit enthält alle möglichen Vorgänge der CPU, die innerhalb des Programmablaufs statt gefunden haben, wie z. B. Unterbrechungen durch höher priore Ereignisse oder Kommunikation. Die Anweisung "Laufzeitmessung" liest einen internen Zähler der CPU aus und schreibt dessen Wert in den Durchgangsparameter MEM. Die Anweisung berechnet die aktuelle Laufzeit des Programms gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Wenn Sie die Laufzeit einzelner Bausteine oder einzelner Befehlssequenzen messen wollen, dann benötigen Sie drei separate Netzwerke. Rufen Sie die Anweisung "Laufzeitmessung" in einem einzelnen Netzwerk innerhalb Ihres Programms auf. Mit diesem ersten Aufruf der Anweisung setzen Sie den Startpunkt der Laufzeitmessung. Anschließend rufen Sie im nächsten Netzwerk den gewünschten Programmbaustein oder die Befehlssequenz auf. Rufen Sie dann in einem weiteren Netzwerk die Anweisung "Laufzeitmessung" zum zweiten Mal auf und weisen Sie dem Durchgangsparameter MEM den gleichen Speicher wie beim ersten Aufruf der Anweisung zu. Die Anweisung "Laufzeitmessung" im dritten Netzwerk liest einen

internen Zähler der CPU aus und berechnet die aktuelle Laufzeit des Programmbausteins oder der Befehlssequenz gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Hinweis

Da beim optimierten Übersetzen des Programms die Reihenfolge der Anweisungen innerhalb einer Befehlssequenz geändert wird, kann die Laufzeit einer Befehlssequenz nicht genau bestimmt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Laufzeitmessung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	LREAL	E, A, M, D, L	Liefert die gemessene Laufzeit in Sekunden
MEM	InOut	LREAL	E, A, M, D, L	Der Inhalt ist nur für interne Zwecke vorgesehen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung anhand einer Berechnung der Laufzeit eines Programmbausteins:

```
AWL                              Erläuterung
CALL RUNTIME                     // Die Anweisung wird aufgerufen.
   RET_VAL := "TagResult"        // Zwischenergebnis
   MEM := "TagMemory"            // Speichert den Startpunkt der Laufzeitmessung.

CALL "Best_before_date",         // Der Baustein "Best_before_date" wird aufgerufen
"Best_before_date_DB"            und abgearbeitet.

CALL RUNTIME                     // Die Anweisung wird zum zweiten Mal aufgerufen.
   RET_VAL := "TagResult"        // Ergebnis der Anweisung
   MEM := "TagMemory"            // Nimmt den Startpunkt der Laufzeitmessung als Grund-
                                 lage zur Laufzeitberechnung.
```

Mit dem ersten Aufruf der Anweisung wird der Startpunkt für die Laufzeitmessung gesetzt und als Referenz für den zweiten Aufruf der Anweisung im Operanden "TagMemory" zwischengespeichert.

Im Anschluss wird der Programmbaustein FB1 "Best_before_date" aufgerufen.

Wenn der Programmbaustein FB1 abgearbeitet ist, wird die Anweisung zum zweiten Mal ausgeführt. Der zweite Aufruf der Anweisung berechnet die Laufzeit des Programmbausteins und schreibt das Ergebnis in den Operanden "TagResult" am Ausgang RET_VAL.

Ein ausführlicheres Beispiel, wie Sie die Gesamtzykluszeit eines Programms messen können, finden Sie im Siemens Industry Online Support unter folgender Beitrags-ID: 87668055 (https://support.industry.siemens.com/cs/ww/de/view/87668055)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Wortverknüpfungen

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit am Parameter OUT, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Parameter IN größer als 31 ist, wird eine Modulo-32-Operation ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	UINT	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" können Sie den Datentyp des Parameters OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL DECO	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" "aus.
IN := "Tag_Input"	// Eingangswert, der das zu setzende Bit im Ausgangswert vorgibt.
OUT := "Tag_Output"	// Ergebnis

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

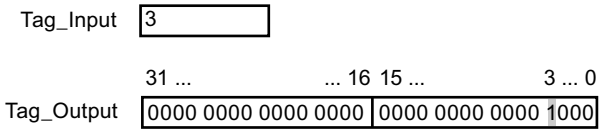

Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "Tag_Input" am Eingang ab und setzt das dritte Bit im Wert des Operanden "Tag_Output" am Ausgang.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Parameter OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Parameter IN und schreibt dessen Bitnummer in den Operanden am Parameter OUT. Wenn der Parameter IN den Wert DW#16#00000001 oder DW#16#00000000 liefert, wird am Parameter OUT der Wert "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" können Sie den Datentyp des Parameters IN wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL ENCO	// Die Anweisung wird aufgerufen.
	// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" "aus.
IN := "Tag_Input"	// Eingangswert, dessen niederwertigstes gesetztes Bit abgelesen wird.
OUT := "Tag_Output"	// Bitnummer des niederwertigsten gesetzten Bits im Eingangswert

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

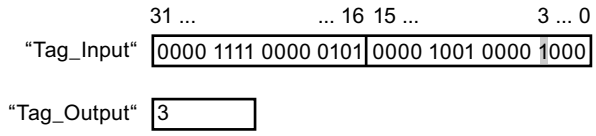

Die Anweisung selektiert das niederwertigste gesetzte Bit der Variablen "Tag_Input" und schreibt dessen Bitposition "3" in die Variable "Tag_Output".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
G	Input	BOOL	E, A, M, D, L oder Konstante	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P oder Konstante	Erster Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Parameter INn und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SEL                     // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" "aus.
G := "Tag_Input_G"           // Signalzustand des Schalters
IN0 := "Tag_Input0"          // Erster Eingangswert
IN1 := "Tag_Input1"          // Zweiter Eingangswert
OUT := "Tag_Output"          // Wert des ausgewählten Eingangs
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Legacy

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L oder Konstante	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Maximale Schrittnummer (z. B.: LST_STEP = 16#08; Es sind maximal 8 Schritte möglich.)
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L oder Konstante	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L oder Konstante	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Voreingestellter erster Schritt des Schrittschaltwerks (1 bis 16)
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P oder Konstante	Verbleibende Bearbeitungszeit für den aktuellen Schritt
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Systemzeit des vorherigen Aufrufs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL DRUM, "DRUM_DB"	// Die Anweisung "Schrittschaltwerk realisieren" wird aufgerufen und der Instanz-Datenbaustein "DRUM_DB" angelegt.
RESET := "Tag_Reset"	// Rücksetzeingang
JOG := "Tag_Input_Jog"	// Bei einer positiven Signalflanke geht die Anweisung zum nächsten Schritt über.
DRUM_EN := "Tag_Input_DrumEN"	// Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.

AWL	Erläuterung
LST_STEP := "Tag_Number_Last-Step"	// Nummer des zuletzt programmierten Schritts
EVENT1 := "MyTag_Event_1"	// Ereignisbit 1
EVENT2 := "MyTag_Event_2"	// Ereignisbit 2
EVENT3 := "MyTag_Event_3"	// Ereignisbit 3
EVENT4 := "MyTag_Event_4"	// Ereignisbit 4
EVENT5 := "MyTag_Event_5"	// Ereignisbit 5
EVENT6 := "MyTag_Event_6"	// Ereignisbit 6
EVENT7 := "MyTag_Event_7"	// Ereignisbit 7
EVENT8 := "MyTag_Event_8"	// Ereignisbit 8
EVENT9 := "MyTag_Event_9"	// Ereignisbit 9
EVENT10 := "MyTag_Event_10"	// Ereignisbit 10
EVENT11 := "MyTag_Event_11"	// Ereignisbit 11
EVENT12 := "MyTag_Event_12"	// Ereignisbit 12
EVENT13 := "MyTag_Event_13"	// Ereignisbit 13
EVENT14 := "MyTag_Event_14"	// Ereignisbit 14
EVENT15 := "MyTag_Event_15"	// Ereignisbit 15
EVENT16 := "MyTag_Event_16"	// Ereignisbit 16
OUT1 := "MyTag_Output_1"	// Ausgabebit 1
OUT2 := "MyTag_Output_2"	// Ausgabebit 2
OUT3 := "MyTag_Output_3"	// Ausgabebit 3
OUT4 := "MyTag_Output_4"	// Ausgabebit 4
OUT5 := "MyTag_Output_5"	// Ausgabebit 5
OUT6 := "MyTag_Output_6"	// Ausgabebit 6
OUT7 := "MyTag_Output_7"	// Ausgabebit 7
OUT8 := "MyTag_Output_8"	// Ausgabebit 8
OUT9 := "MyTag_Output_9"	// Ausgabebit 9
OUT10 := "MyTag_Output_10"	// Ausgabebit 10
OUT11 := "MyTag_Output_11"	// Ausgabebit 11
OUT12 := "MyTag_Output_12"	// Ausgabebit 12
OUT13 := "MyTag_Output_13"	// Ausgabebit 13
OUT14 := "MyTag_Output_14"	// Ausgabebit 14
OUT15 := "MyTag_Output_15"	// Ausgabebit 15
OUT16 := "MyTag_Output_16"	// Ausgabebit 16
Q := "Tag_Output_Q"	// Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD := "Tag_OutputWord"	// Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE := "Tag_ErrorCode"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Wirkungen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET (nur wenn ET < PT ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn ET < PT ist), OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS der Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
CMD_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit zu CMD

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

```
AWL                          Erläuterung
CALL DCAT, "DCAT_DB"         // Die Anweisung "Diskreter Steuerungszeitalarm" wird
                             aufgerufen und der Instanz-Datenbaustein "DCAT_DB" an-
                             gelegt.
CMD := "Tag_Input_CMD"       // Öffnen oder Schließen des Geräts
O_FB := "Tag_Input_O_FB"     // Feedback beim Öffnen
C_FB := "Tag_Input_C_FB"     // Feedback beim Schließen
Q := "Tag_Output_Q"          // Zeigt den Status des Parameters CMD
OA := "Tag_Output_OA"        // Alarmausgang beim Öffnen
CA := "Tag_Output_CA"        // Alarmausgang beim Schließen
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
O_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Schließen"

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt
Legende:																
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren															

Eingangsparameter	Ausgangsparameter
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT
Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL MCAT, "MCAT_DB"	// Die Anweisung "Motorsteuerungszeitalarm" wird aufgerufen und der Instanz-Datenbaustein "MCAT_DB" angelegt.
O_CMD := "Tag_Input_O_CMD"	// Befehlseingabe "Öffnen"
C_CMD := "Tag_Input_C_CMD"	// Befehlseingabe "Schließen"
S_CMD := "Tag_Input_S_CMD"	// Befehlseingabe "Stoppen"
O_FB := "Tag_Input_O_FB"	// Feedback beim Öffnen
C_FB := "Tag_Input_C_FB"	// Feedback beim Schließen
OO := "Tag_OutputOpen"	// Ausgang "Öffnen"
CO := "Tag_OutputClosed"	// Ausgang "Schließen"
OA := "Tag_Output_OA"	// Alarmausgang beim Öffnen
CA := "Tag_Output_CA"	// Alarmausgang beim Schließen
Q := "Tag_Output_Q"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für Schritt 2 verglichen. Der Signalzustand des Parameters OUT wird auf TRUE gesetzt, da die Eingabebits mit der Maske für Schritt 2 übereinstimmen.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

```
AWL                              Erläuterung
CALL IMC, "IMC_DB"               // Die Anweisung "Eingabebits mit den Bits einer Maske
                                 vergleichen" wird aufgerufen und der Instanz-Datenbau-
                                 stein "IMC_DB" angelegt.
IN_BIT0 := "Tag_Input_BIT0"      // Eingangsbit 0
IN_BIT1 := "Tag_Input_BIT1"      // Eingangsbit 1
IN_BIT2 := "Tag_Input_BIT2"      // Eingangsbit 2
IN_BIT3 := "Tag_Input_BIT3"      // Eingangsbit 3
IN_BIT4 := "Tag_Input_BIT4"      // Eingangsbit 4
IN_BIT5 := "Tag_Input_BIT5"      // Eingangsbit 5
IN_BIT6 := "Tag_Input_BIT6"      // Eingangsbit 6
IN_BIT7 := "Tag_Input_BIT7"      // Eingangsbit 7
IN_BIT8 := "Tag_Input_BIT8"      // Eingangsbit 8
IN_BIT9 := "Tag_Input_BIT9"      // Eingangsbit 9
IN_BIT10 := "Tag_Input_BIT10"    // Eingangsbit 10
IN_BIT11 := "Tag_Input_BIT11"    // Eingangsbit 11
IN_BIT12 := "Tag_Input_BIT12"    // Eingangsbit 12
IN_BIT13 := "Tag_Input_BIT13"    // Eingangsbit 13
IN_BIT14 := "Tag_Input_BIT14"    // Eingangsbit 14
IN_BIT15 := "Tag_Input_BIT15"    // Eingangsbit 15
CMP_STEP := "Tag_CMP_STEP"       // Schrittnummer der Maske
OUT := "Tag_Output"              // Zeigt an, ob eine Entsprechung gefunden wurde.
ERR_CODE := "Tag_ErrorCode"      // Fehlerinformation
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "IMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P oder Konstante	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für die Schritte 0 bis 5 verglichen, bis eine Entsprechung gefunden wird. Dabei werden nur die Masken für die Schritte 0 bis 2 abgefragt, weil die Maske für Schritt 2 dem Eingabebit entspricht.

Hinweis

Statische Parameter können Sie im Baustein initialisieren.

```
AWL                              Erläuterung
CALL SMC, "SMC_DB"               // Die Anweisung "Matrixscanner" wird aufgerufen und
                                 der Instanz-Datenbaustein "SMC_DB" angelegt.
IN_BIT0 := "Tag_Input_BIT0"      // Eingangsbit 0
IN_BIT1 := "Tag_Input_BIT1"      // Eingangsbit 1
IN_BIT2 := "Tag_Input_BIT2"      // Eingangsbit 2
IN_BIT3 := "Tag_Input_BIT3"      // Eingangsbit 3
IN_BIT4 := "Tag_Input_BIT4"      // Eingangsbit 4
IN_BIT5 := "Tag_Input_BIT5"      // Eingangsbit 5
IN_BIT6 := "Tag_Input_BIT6"      // Eingangsbit 6
IN_BIT7 := "Tag_Input_BIT7"      // Eingangsbit 7
IN_BIT8 := "Tag_Input_BIT8"      // Eingangsbit 8
IN_BIT9 := "Tag_Input_BIT9"      // Eingangsbit 9
IN_BIT10 := "Tag_Input_BIT10"    // Eingangsbit 10
IN_BIT11 := "Tag_Input_BIT11"    // Eingangsbit 11
IN_BIT12 := "Tag_Input_BIT12"    // Eingangsbit 12
IN_BIT13 := "Tag_Input_BIT13"    // Eingangsbit 13
IN_BIT14 := "Tag_Input_BIT14"    // Eingangsbit 14
IN_BIT15 := "Tag_Input_BIT15"    // Eingangsbit 15
OUT := "Tag_Output"              // Zeigt an, ob eine Entsprechung gefunden wurde.
OUT_STEP := "Tag_Output_STEP"    // Enthält die Schrittnummer mit der entsprechenden
                                 Maske.
ERR_CODE := "Tag_ErrorCode"      // Fehlerinformation
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE

Parameter	Operand	Wert
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "SMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
OUT_STEP	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL LEAD_LAG, "LEAD_LAG_DB"	// Die Anweisung "Lead- und Lag-Algurithmus" wird aufgerufen und der Instanz-Datenbaustein "LEAD_LAG_DB" angelegt.
IN := "Tag_Input"	// Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T := "Tag_Input_SAMPE_T"	// Abtastzeit
OUT := "Tag_Output_Result"	// Ergebnis der Anweisung
ERR_CODE := "Tag_ErrorCode"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in vier äquivalente Codes für eine 7-Segment-Anzeige um und schreiben diese in das Doppelwort des Ausgangs (OUT).

Die Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" erkennt keine Fehlerbedingungen.

Folgende Beziehung besteht zwischen den hexadezimalen Eingabewerten und den Bitmustern der Ausgabe:

Ziffer	-gfedcba	Anzeige
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7
1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

Sieben-Segment-Anzeige

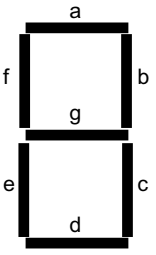

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	IN	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern.
OUT	OUT	DWORD	E, A, M, D, L, P	Bitmuster des Ziels mit vier Byte.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SEG                     // Die Anweisung "Bitmuster für 7-Segment-Anzeige er-
                             zeugen" wird aufgerufen.
IN := "Tag_Input"            // Quellwort
OUT := "Tag_Output"          // Bitmuster für die 7-Segment-Anzeige
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16#065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL BCDCPL                  // Die Anweisung wird aufgerufen.
IN := "Tag_Input"            // 7-stellige BCD-Zahl
ERR_CODE := "Tag_Output"     // Zehnerkomplement der angegebenen BCD-Zahl
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden.
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL BITSUM                  // Die Anweisung wird aufgerufen.
IN := "Tag_Input"            // Operand, dessen gesetzte Bits gezählt werden.
RET_VAL := "Tag_Output"      // Anzahl der gesetzten Bits des Operanden "Tag_In-
                             put".
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu AWL (Seite 7557)

AWL Mnemonik

Bitverknüpfungen

U: UND-Verknüpfung

Beschreibung

Mit der Anweisung "UND-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach UND. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden.

Nach der Ausführung der Anweisung ist das VKE "1", wenn beide verknüpften Signalzustände auch "1" sind. Wenn einer der verknüpften Signalzustände "0" ist, ist das VKE nach der Ausführung der Anweisung "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, müssen alle Abfrageergebnisse "1" sein, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "UND-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm		Relais-Schaltplan	
		Stromschiene	
U	“Tag_Input_1“	Signalzustand 1	Schließer
U	“Tag_Input_2“	Signalzustand 1	Schließer
=	“Tag_Output“	Signalzustand 1	Spule
Zeigt geschlossenen Schalter an.			

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Programmierbeispiele AWL (Seite 7595)

UN: Negierte UND-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte UND-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach UND. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Wenn der Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Wenn der Signalzustand des Operanden "1" ist, ist das Abfrageergebnis "0".

Nach der Ausführung der Anweisung ist das VKE "1", wenn das Abfrageergebnis und das VKE vor der Ausführung der Anweisung den Signalzustand "1" liefern.

Nach der Ausführung der Anweisung ist das VKE "0", wenn das Abfrageergebnis oder das VKE vor der Ausführung der Anweisung den Signalzustand "0" liefern.

Mit der Anweisung "Negierte UND-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

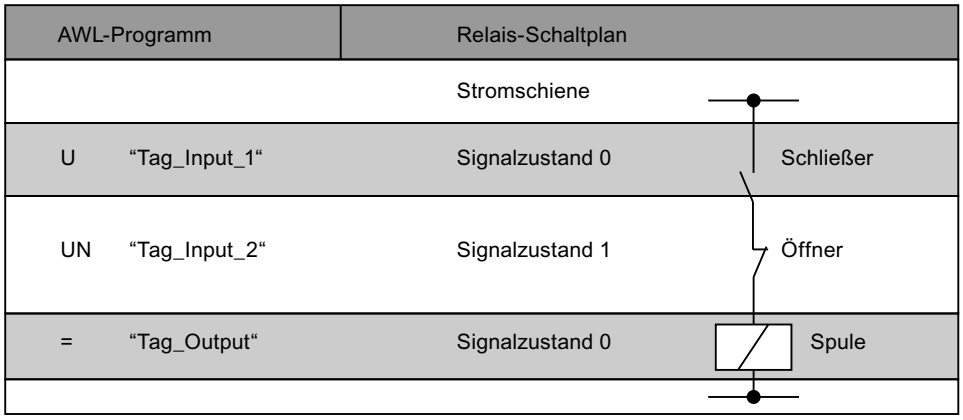

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

O: ODER-Verknüpfung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden.

Nach der Ausführung der Anweisung ist das VKE "1", wenn einer der verknüpften Signalzustände "1" ist. Wenn beide verknüpften Signalzustände "0" ist, ist das VKE nach der Ausführung der Anweisung "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Relais-Schaltplan
	Stromschiene
O "Tag_Input_1"	Signalzustand 1 Schließer ("Tag_Input_1") Signalzustand 0 Schließer ("Tag_Input_2")
O "Tag_Input_2"	
= "Tag_Output"	Signalzustand 1 Spule
	Zeigt geschlossenen Schalter an.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

ON: Negierte ODER-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Wenn der Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Wenn der Signalzustand des Operanden "1" ist, ist das Abfrageergebnis "0".

Nach der Ausführung der Anweisung ist das VKE "1", wenn das Abfrageergebnis oder das VKE vor der Ausführung der Anweisung den Signalzustand "1" liefert. Wenn beide verknüpften Signalzustände "0" sind, ist das VKE nach der Ausführung der Anweisung auch "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "Negierte ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

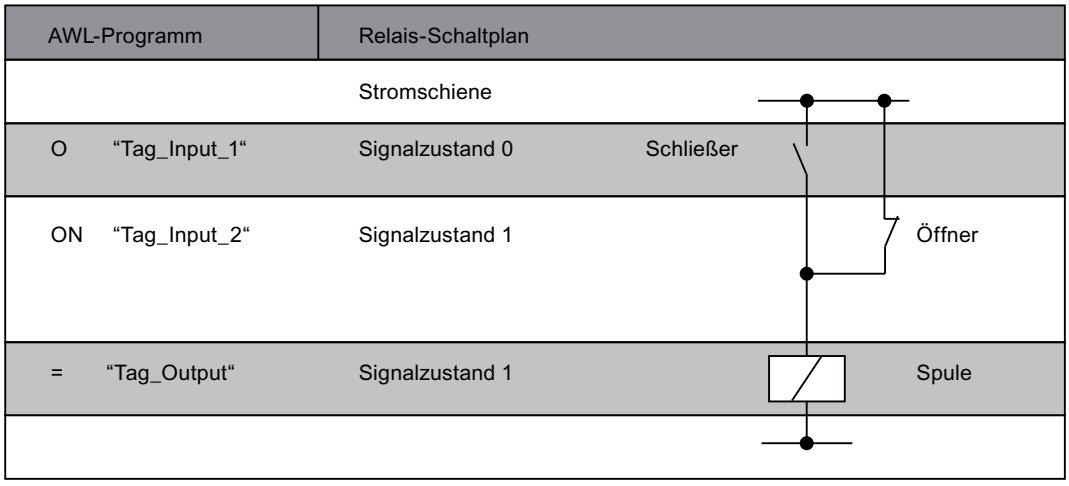

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

X: EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach EXKLUSIV ODER. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden. Nach der Ausführung der Anweisung ist das VKE "1", wenn die miteinander verknüpften Signalzustände ungleich sind. Das VKE ist "0", wenn beide Signalzustände gleich sind.

Wenn z. B. das VKE vor der Ausführung der Anweisung "1" ist und der abgefragte Operand "1" liefert, wird das VKE auf "0" zurückgesetzt. Wenn das VKE "1" ist und der Operand den Signalzustand "0" liefert, wird das VKE nach der Ausführung der Anweisung auf "1" gesetzt.

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame VKE auch "1" ist. Wenn alle Abfrageergebnisse "1" oder "0" sind, wird das VKE auf "0" zurückgesetzt.

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Relais-Schaltplan
	Stromschiene
X “Tag_Input_1“	Kontakt
X “Tag_Input_2“	Kontakt
= “Tag_Output“	Spule

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

XN: Negierte EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Nach der Ausführung der Anweisung ist das VKE "1", wenn die miteinander verknüpften Signalzustände ungleich sind. Das VKE ist "0", wenn beide Signalzustände gleich sind.

Wenn z. B. der abgefragte Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Die Verknüpfung des Abfrageergebnisses mit einem VKE "1" nach EXKLUSIV ODER ergibt den Signalzustand "0". Die Verknüpfung mit einem VKE "0" ergibt den Signalzustand "1".

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm		Relais-Schaltplan
		Stromschiene
X	“Tag_Input_1“	Kontakt
XN	“Tag_Input_2“	Kontakt
=	“Tag_Output“	Spule

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

O: ODER-Verknüpfung von UND-Funktionen

Beschreibung

Mit der Anweisung "ODER-Verknüpfung von UND-Funktionen" bewirken Sie, dass mehrere UND-Verknüpfungen vor einer ODER-Verknüpfung ausführt werden. Das Gesamtergebnis der UND-Verknüpfungen wird in einem Zwischenspeicher gespeichert und mit den weiteren Signalabfragen nach ODER verknüpft.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

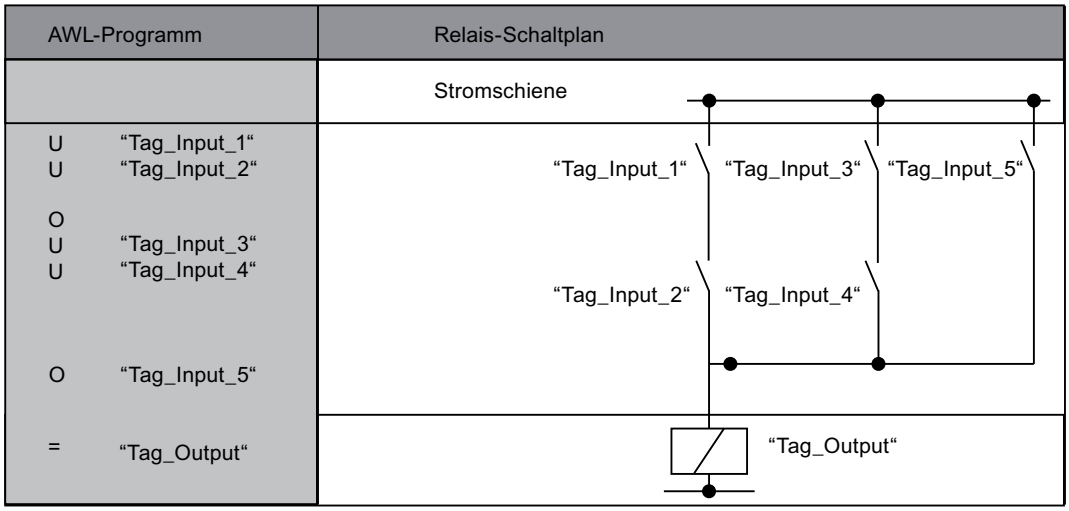

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

U(: UND-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "UND-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer UND-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Das gespeicherte VKE wird mit dem Verknüpfungsergebnis des gesamten Klammerausdrucks nach UND verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der UND-Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

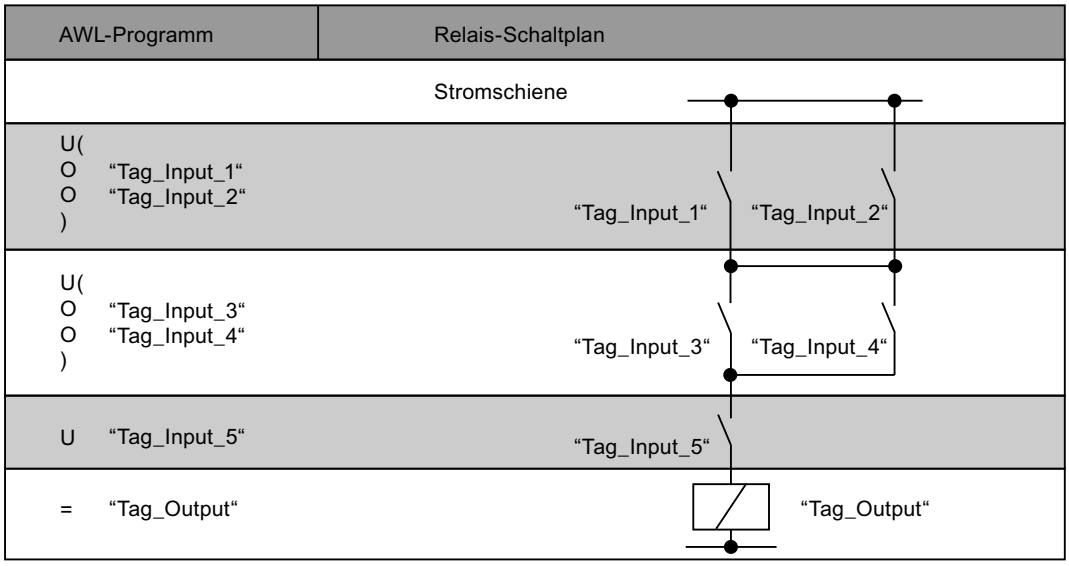

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

UN(: Negierte UND-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte UND-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer UND-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen wird das VKE des gesamten Klammerausdrucks negiert und mit dem gespeicherten VKE nach UND verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der UND-Anweisungen verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
UN(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach UND verknüpfen und das Gesamtergebnis negieren.
O "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach ODER verknüpfen.
O "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach ODER verknüpfen.
)	// Ende des Klammerausdrucks
UN "Tag_Input_3"	// Signalzustand des Operanden auf "0" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

O(: ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer ODER-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Das gespeicherte VKE wird nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen mit dem VKE des gesamten Klammerausdrucks nach ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der ODER-Anweisung verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
O(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach ODER verknüpfen.

AWL	Erläuterung
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ON(: Negierte ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer ODER-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Nach der Bearbeitung der Anweisungen im Klammerausdruck wird das VKE des Klammerausdrucks negiert und mit dem gespeicherten VKE nach ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der ODER-Anweisung verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ON(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach ODER verknüpfen und das Gesamtergebnis negieren.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer EXKLUSIV ODER-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Das gespeicherte VKE wird nach der Bearbeitung der Anweisungen im Klammerausdruck mit dem VKE des gesamten Klammerausdrucks nach EXKLUSIV ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der EXKLUSIV ODER-Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
X(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach EXKLUSIV ODER verknüpfen.
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
X "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer EXKLUSIV ODER-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE. Nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen wird das VKE des Klammerausdrucks negiert und mit dem gespeicherten VKE nach EXKLUSIV ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der EXKLUSIV ODER -Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`XN(`	`// Anfang des Klammerausdrucks`
	`// VKE des Klammerausdrucks nach EXKLUSIV ODER verknüpfen und das Gesamtergebnis negieren.`
`U "Tag_Input_1"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`U "Tag_Input_2"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`)`	`// Ende des Klammerausdrucks`
`XN "Tag_Input_3"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`= "Tag_Output"`	`// Signalzustand des VKE dem Operanden zuweisen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

): Verzweigung schließen

Beschreibung

Mit der Anweisung "Verzweigung schließen" legen Sie das Ende eines Klammerausdrucks fest. Nach der Ausführung der Anweisung wird das Gesamtergebnis des Klammerausdrucks nach der Anweisung verknüpft, die zum Öffnen des Klammerausdrucks verwendet wurde.

Zum Öffnen eines Klammerausdrucks können Sie die folgenden Anweisungen verwenden:

- U(: UND-Verknüpfung mit Verzweigung
- UN(: Negierte UND-Verknüpfung mit Verzweigung
- O(: ODER-Verknüpfung mit Verzweigung
- ON(: Negierte ODER-Verknüpfung mit Verzweigung
- X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung
- XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung

Nach der Anweisung "Verzweigung schließen" setzt die CPU das Binärergebnis BIE auf den Signalzustand, den es vor dem Klammerausdruck hatte.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

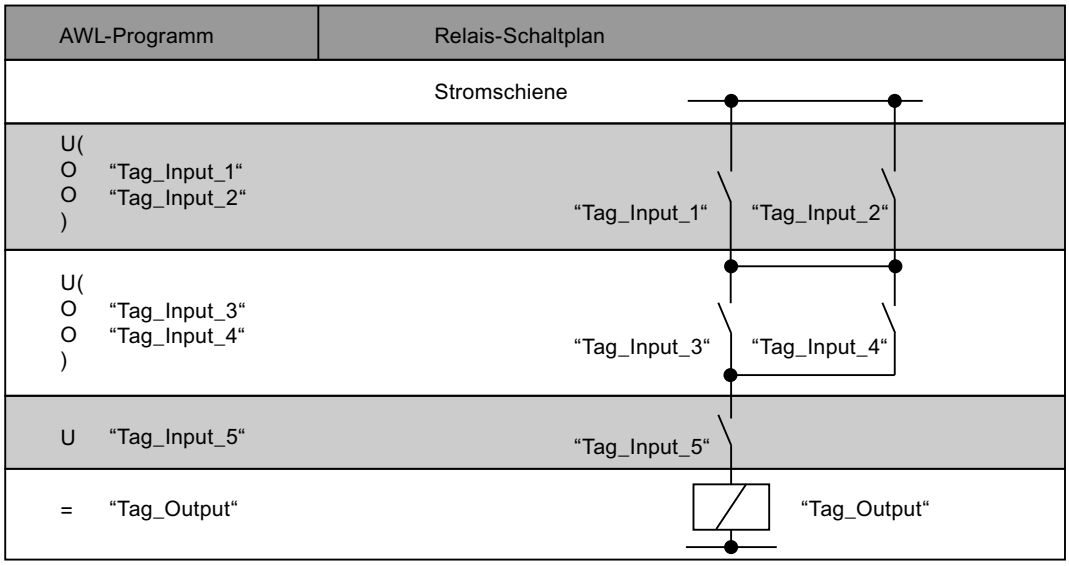

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

=: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" weisen Sie einem angegebenen Operanden den Signalzustand des in der CPU gespeicherten Verknüpfungsergebnisses (VKE) zu. Wenn das VKE den Signalzustand "1" hat, wird der Operand gesetzt. Wenn der Signalzustand "0" ist, wird der Operand auf "0" zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das aktuelle VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

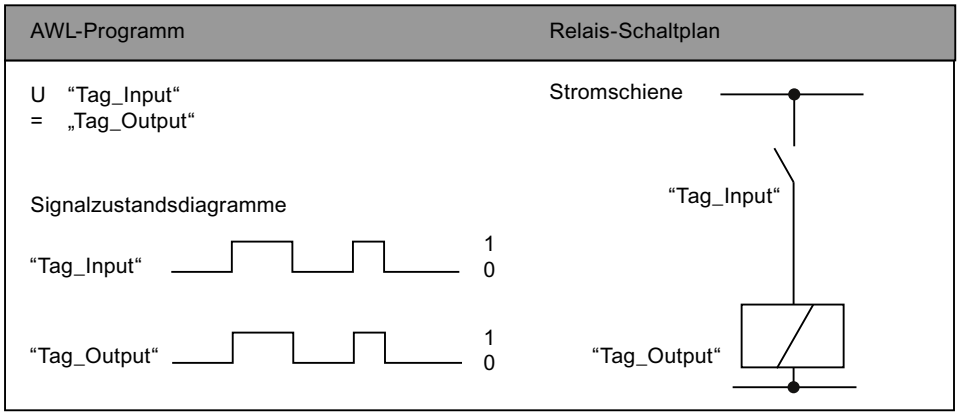

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

R: Rücksetzen

Beschreibung

Mit der Anweisung "Rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Nach der Ausführung der Anweisung wird der angegebene Operand auf "0" zurückgesetzt. Bei einem aktuellen VKE von "0", bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "Tag_Output"	// Operanden auf "0" zurücksetzen, wenn das VKE = "1" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

S: Setzen

Beschreibung

Mit der Anweisung "Setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Nach der Ausführung der Anweisung wird der angegebene Operand auf "1" gesetzt. Bei einem aktuellen VKE von "0", bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_Input_1"              // Signalzustand des Operanden auf "1" abfragen und
                             mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"              // Signalzustand des Operanden auf "1" abfragen und
                             mit dem aktuellen VKE nach UND verknüpfen.
S "Tag_Output"               // Operanden auf "1" setzen, wenn das VKE = "1" ist.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

NOT: VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE). Die Anweisung "VKE invertieren" können Sie an beliebiger Stelle, auch innerhalb einer Verknüpfung, anwenden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_Input_1"              // Signalzustand des Operanden "Tag_Input_1" auf "1"
                             abfragen und mit dem aktuellen VKE nach UND verknüpfen.
```

AWL	Erläuterung
U "Tag_Input_2"	// Signalzustand des Operanden "Tag_Input_2" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"	// Signalzustand des Operanden "Tag_Input_3" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
NOT	// Signalzustand des VKE invertieren
= "Tag_Output"	// Signalzustand des VKE dem Operanden "Tag_Output" zuweisen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Input_1	1	1	1	0	0	0	1	0
Tag_Input_2	0	1	1	0	1	1	0	1
Tag_Input_3	0	0	1	0	0	1	1	1
Tag_Output	1	1	0	1	1	1	1	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SET: VKE auf 1 setzen

Beschreibung

Mit der Anweisung "VKE auf 1 setzen" setzen Sie das aktuelle Verknüpfungsergebnis (VKE) auf den Signalzustand "1".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Signalzustand	Verknüpfungsergebnis (VKE)
SET		1
= “Tag_Output_1“	1	
= “Tag_Output_2“	1	
= “Tag_Output_3“	1	
CLR		0
= “Tag_Output_4“	0	
= “Tag_Output_5“	0	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

CLR: VKE auf 0 rücksetzen

Beschreibung

Mit der Anweisung "VKE auf 0 rücksetzen" setzen Sie das aktuelle Verknüpfungsergebnis (VKE) auf den Signalzustand "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

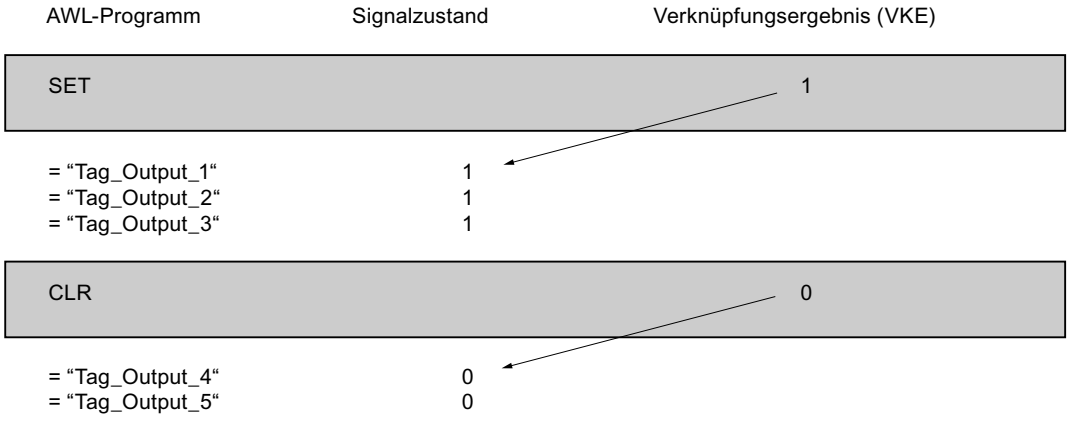

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SAVE: VKE im BIE-Bit speichern

Beschreibung

Mit der Anweisung "VKE im BIE-Bit speichern" sichern Sie das Verknüpfungsergebnis (VKE) im Binärergebnis (BIE). In der Ausführung überträgt die Anweisung den Signalzustand des aktuellen Verknüpfungsergebnisses zum Statusbit BIE. Die Anweisung arbeitet unabhängig von Bedingungen und beeinflusst keine weiteren Statusbits.

Hinweis

Das BIE-Bit kann nach der Anwendung der Anweisung "VKE im BIE-Bit speichern" durch nachfolgende Anweisungen im gleichen oder in unterlagerten Bausteinen wieder verändert werden.

Sie können die Anweisung "VKE im BIE-Bit speichern" nutzen, um den Ausführungsstatus eines Bausteins zu kontrollieren. Wenn Sie z. B. die Anweisung "VKE im BIE-Bit speichern" am Ende eines Bausteins anwenden, wird das BIE-Bit auf den Signalzustand des aktuellen Verknüpfungsergebnisses des Bausteins gesetzt.

Das BIE-Bit hilft bei der Realisierung des EN-/ENO-Mechanismus bei Bausteinaufrufen. Weitere Informationen dazu finden sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand des Operanden "Tag_Input" auf "1" abfragen.
SAVE	// Signalzustand des VKE in das BIE-Bit übertragen.
BEB	// Baustein bei VKE = "1" beenden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

FN: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand des Verknüpfungsergebnisses (VKE) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker <Operand> gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im VKE bei einer negativen und einer positiven Signalflanke:

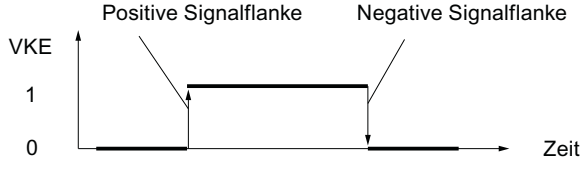

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird das VKE nach der Flankenauswertung einen Programmzyklus lang auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand des VKE "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Flankenmerker , in dem das VKE der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

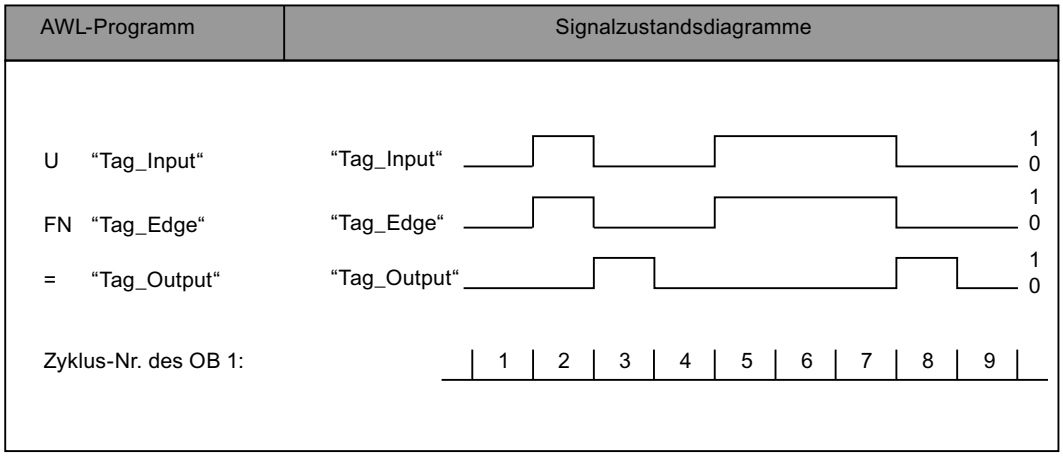

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

FP: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand des Verknüpfungsergebnisses (VKE) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker <Operand> gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im VKE bei einer negativen und einer positiven Signalflanke:

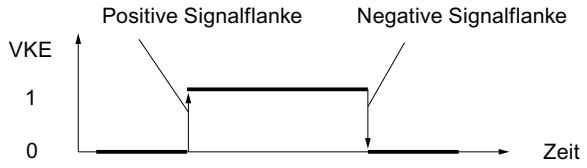

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird das VKE nach der Flankenauswertung einen Programmzyklus lang auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand des VKE "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Flankenmerker , in dem das VKE der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

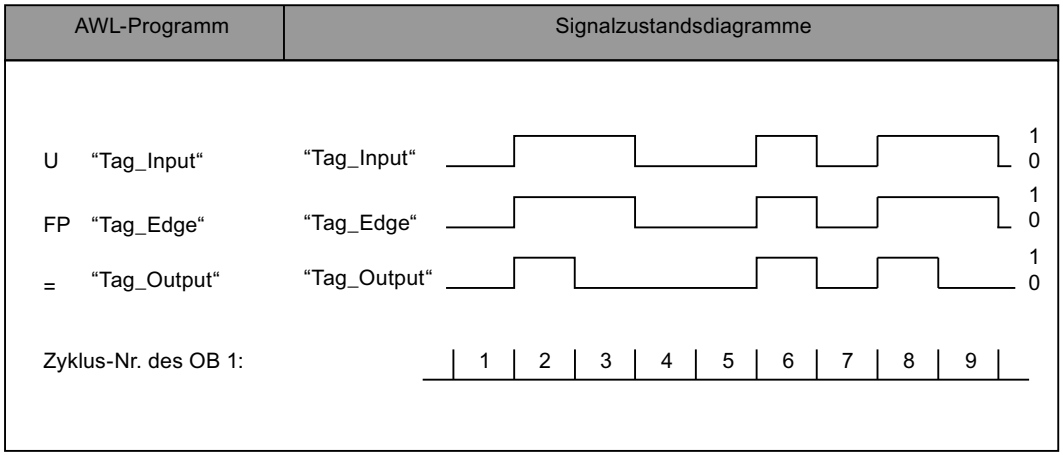

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

Zeiten

FR: Zeit freigeben

Beschreibung

Mit der Anweisung "Zeit freigeben" bewirken Sie den Neustart einer Zeit. Die Anweisung wird mit einer positiven Signalflanke ausgeführt und setzt den internen Flankenmerker für das Starten einer Zeit zurück. Bei der Bearbeitung einer Anweisung zum Starten der Zeit mit einem Verknüpfungsergebnis "1" läuft die Zeit dann neu an, auch wenn keine positive Signalflanke vor der Startanweisung vorliegt. Bei einem Verknüpfungsergebnis "0" vor einer Startanweisung hat das Rücksetzen des internen Flankenmerkers keinen Einfluss auf die Zeit.

Ein Wechsel des Verknüpfungsergebnisses auf "0" vor der Anweisung "Zeit freigeben" hat nach dem Neustart der Zeit keine Wirkung.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit freigeben":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die freigegeben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_EnableInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyTimer"	// Bei einer positiven Signalflanke Zeit freigeben.
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

L: Zeitwert laden (Seite 1567)

Grundlagen zu AWL (Seite 7557)

L: Zeitwert laden

Beschreibung

Mit der Anweisung "Zeitwert laden" laden Sie den Zeitwert einer angegebenen Zeit in Akkumulator 1. Der dualcodierte Zeitwert wird dabei ohne die Zeitbasis geladen. Anstelle der Zeitbasis werden in den Akkumulator 1 Nullen geschrieben.

Nach dem Laden entspricht der im Akkumulator 1 stehende Wert einer positiven Zahl des Datentyps INT.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Das folgende Bild zeigt beispielweise, wie der Zeitwert in den Akkumulator geladen wird:

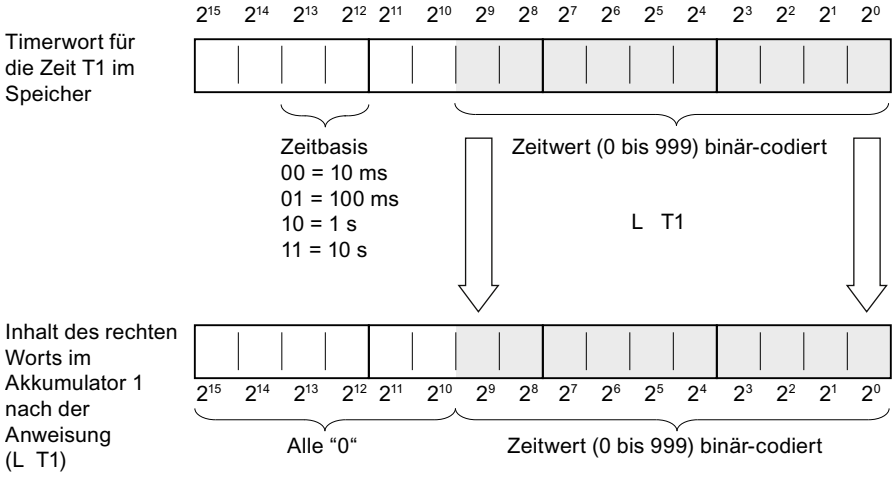

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitwert laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, deren aktueller Zeitwert geladen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_EnableInput"          // Operanden auf "1" abfragen und mit dem aktuellen
                             VKE nach UND verknüpfen.
FR "MyTimer"                 // Bei einer positiven Flanke Zeit freigeben.
U "Tag_StartInput"           // Operanden auf "1" abfragen und mit dem aktuellen
                             VKE nach UND verknüpfen.
L S5T#10s                    // Zeitdauer(10s) laden
SI "MyTimer"                 // Zeit als Impuls starten, wenn eine positive Flanke
                             am Operanden "Tag_StartInput" oder am Operanden
                             "Tag_EnableInput" vorliegt und der Operand "Tag_Start-
                             Input" den Signalzustand "1" führt.
U "Tag_Reset"                // Operanden auf "1" abfragen und mit dem aktuellen
                             VKE nach UND verknüpfen.
R "MyTimer"                  // Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"                  // Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"          // Zeitstatus dem Operanden zuweisen.
L "MyTimer"                  // Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
```

AWL	Erläuterung
`T "Tag_TimerValue"`	`// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

LC: Zeitwert im BCD-Format laden

Beschreibung

Mit der Anweisung "Zeitwert im BCD-Format laden" laden Sie den Zeitwert einer angegebenen Zeit, der dualcodiert vorliegt, im BCD-Format in Akkumulator 1. Beim Laden wird dabei auch die Zeitbasis in Akkumulator 1 übertragen.

Nach dem Laden entspricht der im Akkumulator 1 stehende Wert einer Zeitdauer im Format S5TIME. Das linke Wort im Akkumulator 1 wird mit Nullen belegt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Das folgende Bild zeigt, wie der Zeitwert in den Akkumulator geladen wird:

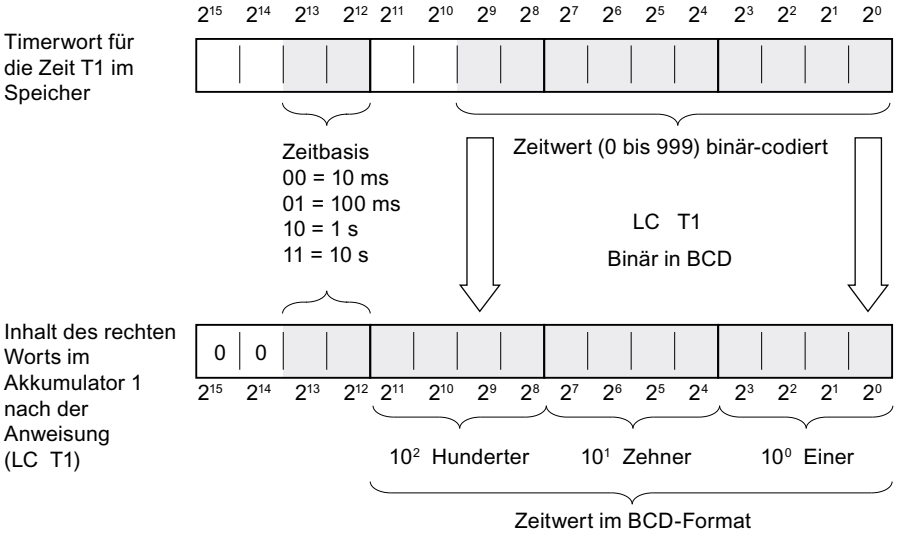

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitwert im BCD-Format laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, deren aktueller Zeitwert im BCD-Format geladen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_EnableInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyTimer"	// Bei einer positiven Flanke Zeit freigeben.
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer(10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Flanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_Start-Input" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
LC "MyTimer"	// Aktuellen Zeitwert der Zeit im BCD-Format in Akku-mulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerVa-lue" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

L: Zeitwert laden (Seite 1567)

Grundlagen zu AWL (Seite 7557)

R: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine angegebene Zeit auf "0" zurück. Die Anweisung wird ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Solange das VKE "1" vor der Anweisung ansteht, liefert die angegebene Zeit den Wert "0". Die Anweisung setzt auch den Zeitwert und die Zeitbasis der programmierten Zeitdauer auf "0" zurück.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Hinweis

Die Anweisung "Zeit rücksetzen" setzt den internen Flankenmerker nicht zurück. Zum Zurücksetzen des internen Flankenmerkers müssen Sie die Anweisung "Zeit freigeben" programmieren oder eine Anweisung zum Starten der Zeit mit dem Signalzustand "0" ausführen.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die zurückgesetzt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

L: Zeitwert laden (Seite 1567)

Grundlagen zu AWL (Seite 7557)

SI: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, solange das Verknüpfungsergebnis (VKE) vor der Anweisung "1" bleibt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die Zeit angehalten. Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", solange die Zeit läuft.

Ein Neustart der Zeit kann mit einer positiven Signalflanke vor der Anweisung "Zeit als Impuls starten" oder mithilfe der Anweisung "Zeit freigeben" bewirkt werden. Voraussetzung dafür ist, dass die Zeit nicht zurückgesetzt ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als Impuls starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Impuls gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

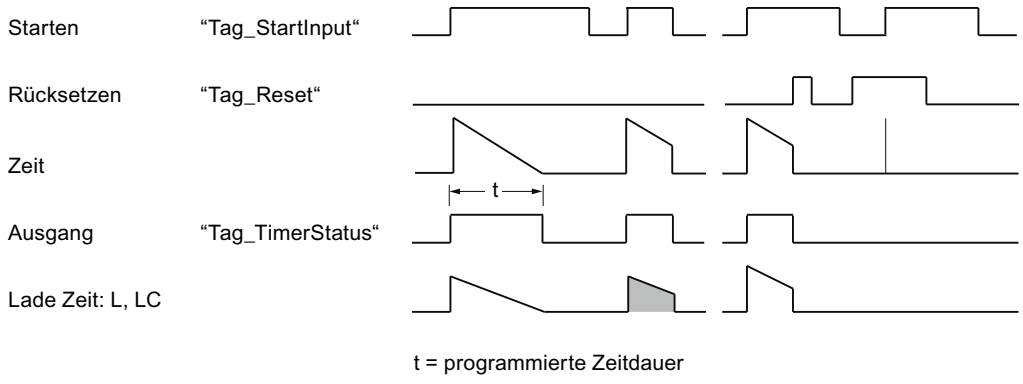

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

L: Zeitwert laden (Seite 1567)

Grundlagen zu AWL (Seite 7557)

SV: Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, auch wenn das Verknüpfungsergebnis vor der Anweisung auf "0" wechselt. Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", solange die Zeit läuft.

Die Anweisung startet bei jeder positiven Signalflanke die Zeit mit der programmierten Zeitdauer neu, auch wenn die Zeit noch nicht abgelaufen ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als verlängerten Impuls starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als verlängerter Impuls gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_StartInput"           // Operanden auf "1" abfragen und mit dem aktuellen
                             VKE nach UND verknüpfen.
L S5T#10s                    // Zeitdauer (10s) laden
```

AWL	Erläuterung
SV "MyTimer"	// Zeit als verlängerter Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

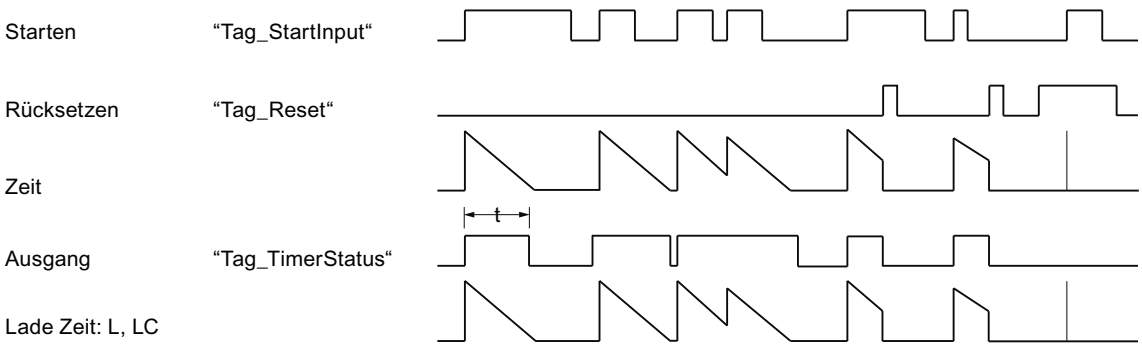

t = programmierte Zeitdauer

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)
Grundlagen zum Statuswort (Seite 196)
Programmierbeispiele AWL (Seite 7595)
Grundlagen zu AWL (Seite 7557)

SE: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der in Akkumulator 1 stehenden Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer im Akkumulator 1 setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Einschaltverzögerung gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SE "MyTimer"	// Zeit als Einschaltverzögerung starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert an den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format in den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

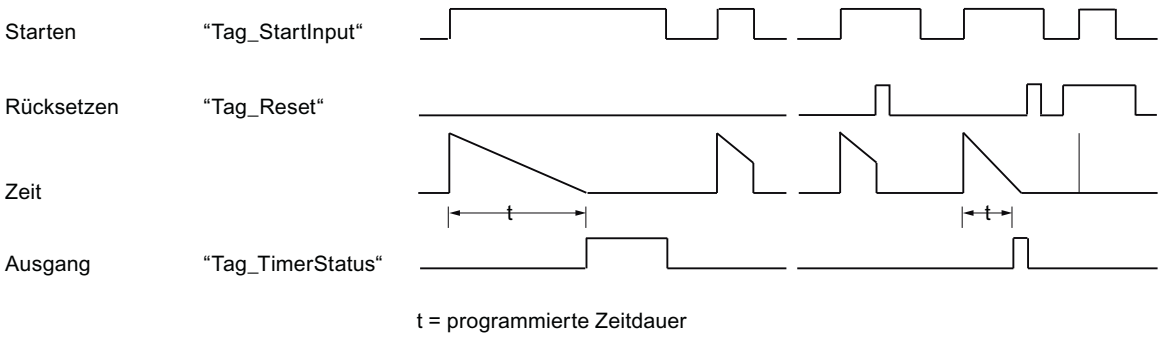

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

L: Zeitwert laden (Seite 1567)

Grundlagen zu AWL (Seite 7557)

SS: Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, auch wenn das Verknüpfungsergebnis (VKE) vor der Anweisung auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "1". Das Abfrageergebnis wird dabei durch den Signalzustand des aktuellen Verknüpfungsergebnisses vor der Anweisung nicht beeinflusst.

Die Anweisung startet bei jeder positiven Flanke die Zeit mit der programmierten Zeitdauer neu, auch wenn die Zeit noch nicht abgelaufen ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als speichernde Einschaltverzögerung gestartet wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SS "MyTimer"	// Zeit als speichernde Einschaltverzögerung starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format in den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

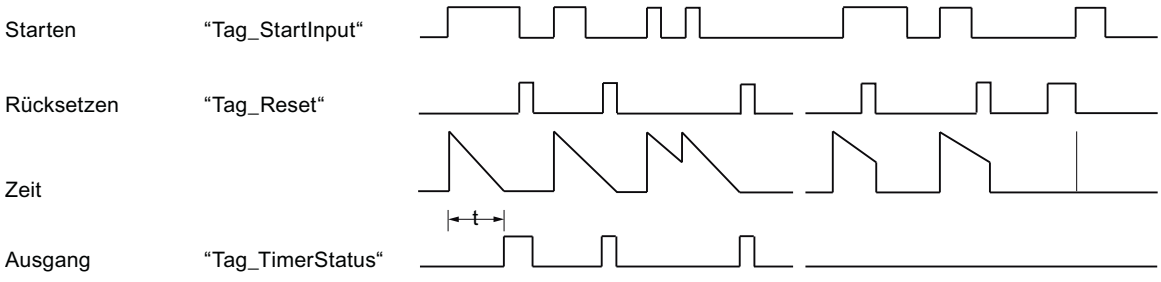

t = programmierte Zeitdauer

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SA: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine angegebene Zeit bei einer negativen Signalflanke. Wenn eine negative Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab. Wenn das Verknüpfungsergebnis (VKE) vor dem Ablauf der Zeit auf "1" wechselt, wird die Zeit zurückgesetzt. Erst wenn eine negative Signalflanke bei der Ausführung der Anweisung erfasst wird, wird die Zeit wieder gestartet.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist oder die programmierte Zeit läuft. Wenn die Zeit nicht läuft oder das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als Ausschaltverzögerung starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 1567)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Ausschaltverzögerung gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SA "MyTimer"	// Zeit als Ausschaltverzögerung starten, wenn eine negative Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert an den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels.

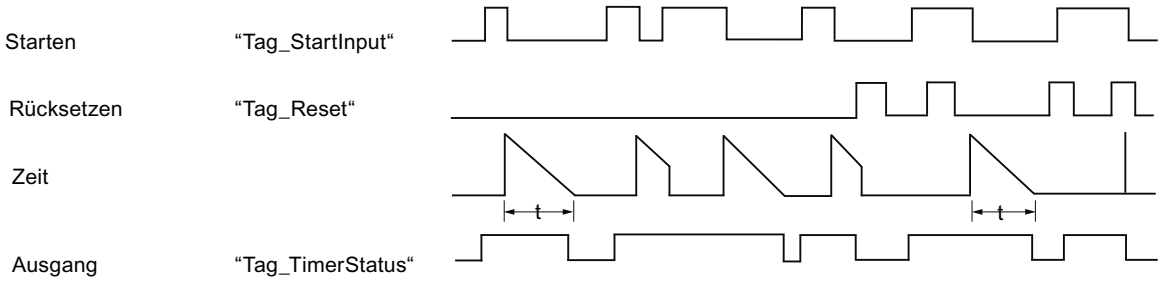

t = programmierte Zeitdauer

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Zähler

FR: Zähler freigeben

Beschreibung

Mit der Anweisung "Zähler freigeben" bewirken Sie, dass die Anweisungen "Zähler setzen", "Vorwärts zählen" und "Rückwärts zählen" ohne eine aktuell vorliegende positive Signalflanke ausgeführt werden. Bei der Ausführung setzt die Anweisung den internen Flankenmerker auf den Signalzustand "0" zurück. Die Abfrage vor der entsprechenden Zählanweisung muss weiterhin den Signalzustand "1" führen.

Die Anweisung "Zähler freigeben" wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt.

Das Freigeben eines Zählers ist für die Ausführung einer Zählanweisung nicht erforderlich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler freigeben":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, der freigegeben wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyCounter"	// Zähler freigeben
U "Tag_Input_2"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ZR "MyCounter"	// Rückwärts zählen
U "MyCounter"	// Zählerstatus abfragen
= "Tag_Output"	// Abfrageergebnis dem Operanden zuweisen.

Im Beispiel wird der Zähler "MyCounter" freigegeben, wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt. Wenn der Operand "Tag_Input_2" den Signalzustand "1" führt, wird der Zählwert beim Freigeben des Zählers um eins verringert. Bei einem Signalzustand "0" des Operanden "Tag_Input_2", bleibt der Zählwert unverändert. Der Zählwert wird auch unabhängig von der Zählerfreigabe um Eins verringert, wenn der Signalzustand des Operanden "Tag_Input_2" von "0" auf "1" wechselt.

Die Abfrage des Zählerstatus liefert das Ergebnis "1", wenn Zählerstand größer Null ist. Wenn der Zählerstand gleich Null ist, ist das Abfrageergebnis "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

L: Zählwert laden

Beschreibung

Mit der Anweisung "Zählwert laden" übertragen Sie den aktuellen Wert eines angegebenen Zählers dualcodiert in das rechte Wort des Akkumulators 1. Der geladene Zählwert steht dann als 16-Bit Ganzzahl zur weiteren Bearbeitung zur Verfügung.

Das folgende Bild zeigt die Funktionsweise der Anweisung:

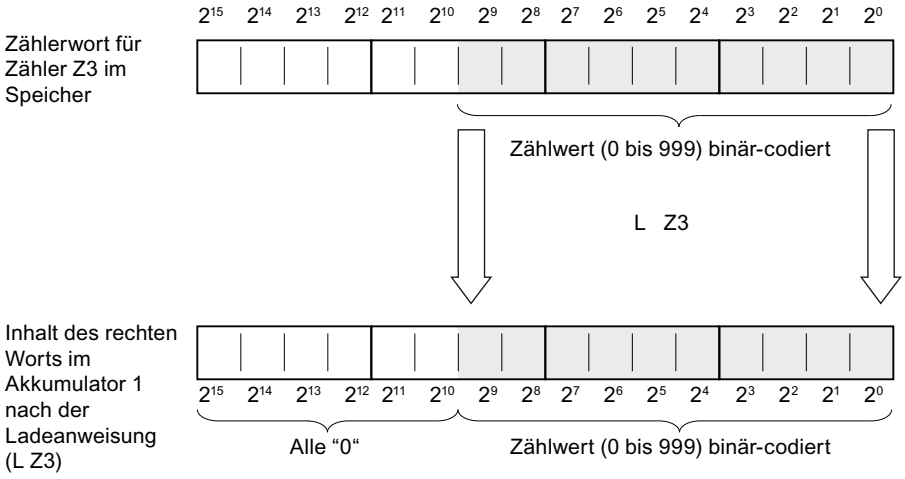

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zählwert laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen aktueller Wert geladen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "MyCounter"	// Aktuellen Zählwert dualcodiert in den Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value"	// Zählwert in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value" in Akkumulator 1 laden.
>=I	// Abfrage, ob der Zählwert größer oder gleich dem Wert des Operanden "Tag_Value" ist.
= "Tag_Output"	// Abfrageergebnis dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

LC: Zählwert im BCD-Format laden

Beschreibung

Mit der Anweisung "Zählwert im BCD-Format laden" übertragen Sie den aktuellen Wert eines angegebenen Zählers als BCD in das rechte Wort des Akkumulators 1. Der geladene Zählwert steht dann in BCD-Form zur weiteren Bearbeitung zur Verfügung.

Das folgende Bild zeigt die Funktionsweise der Anweisung "Zählwert im BCD-Format laden":

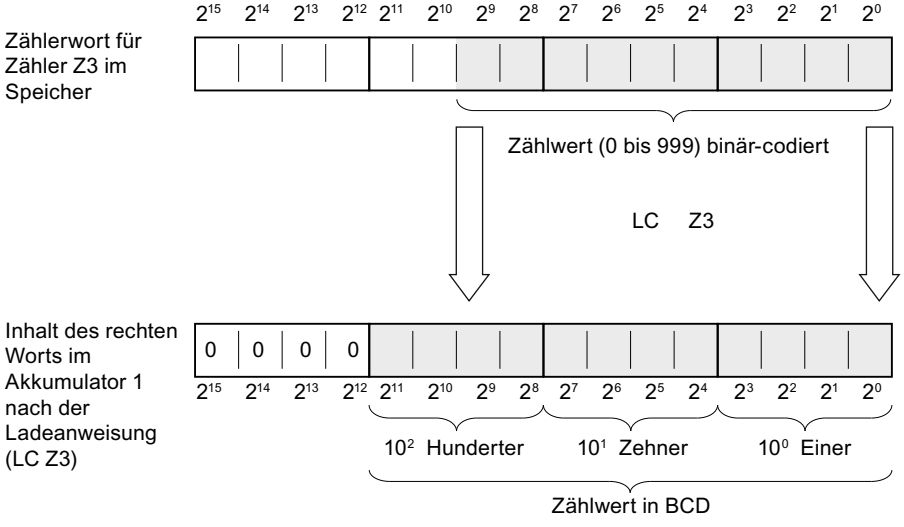

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zählwert im BCD-Format laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen aktueller Wert geladen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`LC "MyCounter"`	`// Aktuellen Zählwert als BCD in den Akkumulator 1 laden.`
`T "Tag_Output"`	`// Aktuellen Zählwert in den Operanden "Tag_Output" transferieren und speichern.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

R: Zähler rücksetzen

Beschreibung

Mit der Anweisung "Zähler rücksetzen" setzen Sie den aktuellen Zählwert eines angegebenen Zählers auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung den Signalzustand "1" führt. Solange das VKE vor der Anweisung "Zähler rücksetzen" auf "1" steht, liefern die Abfragen des Zählerstatus auf "1" das Ergebnis "0".

Durch das Zurücksetzen eines Zählers wird der interne Flankenmerker für die Anweisungen "Zähler setzen", "Vorwärts zählen" und "Rückwärts zählen" nicht zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler rücksetzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert auf "0" zurückgesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyCounter"	// Zähler zurücksetzen

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

S: Zähler setzen

Beschreibung

Mit der Anweisung "Zähler setzen" setzen Sie einen angegebenen Zähler auf den Wert, der sich zum Zeitpunkt der Abfrage im Akkumulator 1 befindet.

Die Anweisung "Zähler setzen" wird nur ausgeführt, wenn das Verknüpfungsergebnis vor der Anweisung von "0" auf "1" wechselt. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Zum Setzen eines Zählers muss im Akkumulator 1 ein gültiger Zählwert stehen. Gültige Zählwerte sind Werte im BCD-Format mit drei Dekaden. Es sind dabei nur positive Zählwerte erlaubt, negative Zählwerte kann die Zählanweisung nicht bearbeiten. Wenn im Akkumulator 1 ein gültiger BCD-Wert vorliegt, dann hat das Statusbit OV den Signalzustand "0". Wenn kein gültiger BCD-Wert vorliegt, hat das Statusbit OV den Signalzustand "1". Es wird keine Synchronfehlerbehandlung gestartet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler setzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, der gesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_Input"                // Signalzustand auf "1" abfragen und mit dem aktuel-
                             len VKE nach UND verknüpfen.
L "Tag_CountValue"           // Zählwert in Akkumulator 1 laden.
S "MyCounter"                // Zähler auf den Wert setzen, der im Akkumulator 1
                             steht.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ZV: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den aktuellen Zählwert um eins, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt. Der Zählwert wird bei jeder positiven Signalflanke weiterhin erhöht, bis er den oberen Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, hat das VKE vor der Anweisung keinen Einfluss mehr auf den Zählwert.

Zur Ausführung der Anweisung "Vorwärts zählen" ist immer eine positive Signalflanke erforderlich. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert bei einer positiven Flanke um eins erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_Input"                // Signalzustand auf "1" abfragen und mit dem aktuel-
                             len VKE nach UND verknüpfen.
```

AWL	Erläuterung
ZV "MyCounter"	// Zählwert bei einer positiven Flanke um eins erhöhen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

ZR: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den aktuellen Zählwert um eins, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt. Der Zählwert wird bei jeder positiven Signalflanke weiterhin verringert, bis er den unteren Grenzwert "0" erreicht. Wenn der Grenzwert erreicht ist, hat das VKE vor der Anweisung keinen Einfluss mehr auf den Zählwert. Ein Zählen mit negativem Zählwert findet nicht statt.

Zur Ausführung der Anweisung "Rückwärts zählen" ist immer eine positive Signalflanke erforderlich. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert bei einer positiven Flanke um eins verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ZR "MyCounter"	// Zählwert bei einer positiven Flanke um eins verringern.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

Vergleicher

? I: 16-Bit Ganzzahlen vergleichen

Beschreibung

Mit der Anweisung "16-Bit Ganzzahlen vergleichen" vergleichen Sie den Inhalt des rechten Wortes im Akkumulator 2 mit dem Inhalt des rechten Worts im Akkumulator 1. Die Inhalte der Akkumulatoren 1 und 2 werden als 16-Bit Ganzzahlen (INT) interpretiert. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Hinweis

Datentypen ohne Vorzeichen (z. B. UINT) werden nicht unterstützt.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==I	16-Bit Ganzzahlen auf "gleich" vergleichen
<>I	16-Bit Ganzzahlen auf "ungleich" vergleichen
>I	16-Bit Ganzzahlen auf "größer" vergleichen
<I	16-Bit Ganzzahlen auf "kleiner" vergleichen
>=I	16-Bit Ganzzahlen auf "größer oder gleich" vergleichen
<=I	16-Bit Ganzzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==I	0	1	0
<>I	1	0	1
>I	1	0	0
<I	0	0	1
>=I	1	1	0
<=I	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input_1"`	`// Wert des Operanden "Tag_Input_1" in Akkumulator 1 laden.`
`L "Tag_Input_2"`	`// Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumulator 2 verschieben.`
	`// Wert des Operanden "Tag_Input_2" in Akkumulator 1 laden.`
`>I`	`// Vergleich, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.`
`= "Tag_Output"`	`// Ergebnis des Vergleichs dem Operanden "Tag_Output" zuweisen.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	10	10	5
Tag_Input_2	5	10	10
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

? D: 32-Bit Ganzzahlen vergleichen

Beschreibung

Mit der Anweisung "32-Bit Ganzzahlen vergleichen" vergleichen Sie den Inhalt des Akkumulators 2 (AKKU 2) mit dem Inhalt des Akkumulators 1 (AKKU 1). Die Inhalte der Akkumulatoren 1 und 2 werden als 32-Bit Ganzzahlen (DINT) interpretiert. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Hinweis

Datentypen ohne Vorzeichen (z. B. UDINT) werden nicht unterstützt.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==D	32-Bit Ganzzahlen auf "gleich" vergleichen
<>D	32-Bit Ganzzahlen auf "ungleich" vergleichen
>D	32-Bit Ganzzahlen auf "größer" vergleichen
<D	32-Bit Ganzzahlen auf "kleiner" vergleichen
>=D	32-Bit Ganzzahlen auf "größer oder gleich" vergleichen
<=D	32-Bit Ganzzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf den Signalzustand "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==D	0	1	0
<>D	1	0	1
>D	1	0	0
<D	0	0	1
>=D	1	1	0
<=D	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Input_1"              // Wert des Operanden "Tag_Input_1" in Akkumulator 1
                             laden.
L "Tag_Input_2"              // Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumu-
                             lator 2 verschoben.
                             // Wert des Operanden "Tag_Input_2" in Akkumulator 1
                             laden.
>D                           // Vergleich, ob der Wert des Operanden "Tag_Input_1"
                             größer als der Wert des Operanden "Tag_Input_2" ist.
= "Tag_Output"               // Ergebnis des Vergleichs dem Operanden "Tag_Output"
                             zuweisen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	150 000	150 000	100 000
Tag_Input_2	100 000	150 000	150 000
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

? R: Gleitpunktzahlen vergleichen

Beschreibung

Mit der Anweisung "Gleitpunktzahlen vergleichen" vergleichen Sie den Inhalt des Akkumulators 2 (AKKU 2) mit dem Inhalt des Akkumulators 1 (AKKU 1). Die Inhalte der Akkumulatoren 1 und 2 werden als Gleitpunktzahlen interpretiert. Zur Ausführung des Vergleichs müssen in den Akkumulatoren 1 und 2 gültige Gleitpunktzahlen stehen. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==R	Gleitpunktzahlen auf "gleich" vergleichen
<>R	Gleitpunktzahlen auf "ungleich" vergleichen
>R	Gleitpunktzahlen auf "größer" vergleichen
<R	Gleitpunktzahlen auf "kleiner" vergleichen

Vergleichsfunktion	Bedeutung
>=R	Gleitpunktzahlen auf "größer oder gleich" vergleichen
<=R	Gleitpunktzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf den Signalzustand "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==R	0	1	0
<>R	1	0	1
>R	1	0	0
<R	0	0	1
>=R	1	1	0
<=R	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Hinweis

Bei einem Vergleich ungültiger Gleitpunktzahlen setzt die CPU das Verknüpfungsergebnis auf "0" und die Statusbits A0, A1, OV und OS auf "1".

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Wert des Operanden "Tag_Input_1" in Akkumulator 1 laden.
L "Tag_Input_2"	// Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Input_2" in Akkumulator 1 laden.
>R	// Vergleich, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
= "Tag_Output"	// Ergebnis des Vergleichs dem Operanden "Tag_Output" zuweisen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	10 515	10 515	5 232
Tag_Input_2	5 232	10 515	10 515
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Mathematische Funktionen

Ganzzahlen

+I: Ganzzahlen (16 Bit) addieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) addieren" addieren Sie die Werte, die sich in den rechten Wörtern der Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Summe speichert die Anweisung im rechten Wort des Akkumulators 1. Das linke Wort des Akkumulators 1 bleibt unverändert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Ersten Wert der Addition laden.
L "Tag_Value_2"             // Zweiten Wert der Addition laden.
+I                          // Werte addieren
T "Tag_Result"              // Summe in den Operanden "Tag_Result" übertragen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

-I: Ganzzahlen (16 Bit) subtrahieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) subtrahieren" subtrahieren Sie den Wert des rechten Worts des Akkumulators 1 vom Wert des rechten Worts im Akkumulator 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Differenz speichert die Anweisung im rechten Wort des Akkumulators 1. Das linke Wort des Akkumulators 1 bleibt unverändert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Addition in Akkumulator 1 laden.
-I	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

*I: Ganzzahlen (16 Bit) multiplizieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) multiplizieren" multiplizieren Sie die Werte, die sich in den rechten Wörtern der Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 16-Bit Ganzzahlen.

Das Produkt speichert die Anweisung im Akkumulator 1 als 32-Bit Ganzzahl (DINT).

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Multiplikation laden.
L "Tag_Value_2"	// Zweiten Wert der Multiplikation laden.
*I	// Werte multiplizieren
T "Tag_Result"	// Produkt in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

/I: Ganzzahlen (16 Bit) dividieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) dividieren" dividieren Sie den Wert des rechten Worts im Akkumulator 2 durch den Wert des rechten Worts im Akkumulator 1. Die Werte der Akkumulatoren interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Anweisung liefert zwei Ergebnisse und zwar den Quotienten und den übriggebliebenen Rest der Division. Der Quotient ist das ganzzahlige Ergebnis der Division. Diesen speichert die Anweisung im rechten Wort des Akkumulators 1. Den Rest der Division speichert die Anweisung im linken Wort des Akkumulators 1. Bei einem negativen Dividenden ist der Rest der Division auch negativ.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten und als Rest jeweils den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
/I	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Ergebnis in den Operanden "Tag_Result" übertragen.

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der rechten Wörter in den Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Ganzzahlen (16 Bit) dividieren":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Ganzzahlen (16 Bit) dividieren":

Akkumulator 1	Wert
Bit 0 bis 15 (Rechtes Wort)	3
Bit 16 bis 31 (Linkes Wort)	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Programmierbeispiele AWL (Seite 7595)

Grundlagen zu AWL (Seite 7557)

+D: Ganzzahlen (32 Bit) addieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) addieren" addieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 32-Bit Ganzzahlen. Die Summe speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Addition laden.
L "Tag_Value_2"	// Zweiten Wert der Addition laden.
+D	// Werte addieren
T "Tag_Result"	// Summe in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

-D: Ganzzahlen (32 Bit) subtrahieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) subtrahieren" subtrahieren Sie den Wert des Akkumulators 1 vom Wert des Akkumulators 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 32-Bit Ganzzahlen. Das Ergebnis der Subtraktion speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Subtraktion in Akkumulator 1 laden.
-D	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

*D: Ganzzahlen (32 Bit) multiplizieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) multiplizieren" multiplizieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 32-Bit Ganzzahlen. Das Produkt speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Multiplikation laden.
L "Tag_Value_2"	// Zweiten Wert der Multiplikation laden.
*D	// Werte multiplizieren
T "Tag_Result"	// Produkt in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

/D: Ganzzahlen (32 Bit) dividieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) dividieren" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 32-Bit Ganzzahlen. Das ganzzahlige Ergebnis der Division (Quotient) speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
/D	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Quotienten in den Operanden "Tag_Result" übertragen.

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Ganzzahlen (32 Bit) dividieren":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Ganzzahlen (32 Bit) dividieren":

Akkumulator	Wert
Akkumulator 1	3

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

+: Konstanten addieren

Beschreibung

Mit der Anweisung "Konstanten addieren" addieren Sie den Wert einer angegebenen Konstante zum Inhalt des Akkumulators 1. Den Konstantenwert können Sie als 16- oder 32-Bit Ganzzahl angeben. Das Ergebnis speichert die Anweisung im Akkumulator 1. Bei Addition von 16-Bit Ganzzahlen beeinflusst die Anweisung nur das rechte Wort des Akkumulators 1.

Um die Addition einer 16-Bit Ganzzahl als DINT-Rechnung auszuführen, müssen Sie vor der Konstante "L#" eingeben. Die Anweisung "Konstanten addieren" erlaubt die Verwendung negativer Zahlen.

Die Anweisung beeinflusst keine Statusbits.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Konstanten addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	INT, DINT	Konstante, die addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden "Tag_Value_1" laden.
+ 25                        // Den Wert des Akkumulators 1 um 25 erhöhen.
T "Tag_Result_1"            // Inhalt des Akkumulators 1 an den Operanden "Tag_Re-
                            sult_1" übertragen.
L "Tag_Value_2"             // Wert des Operanden "Tag_Value_2" in Akkumulator 1
                            laden.
```

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
+ -100	// Den Wert des Akkumulators 1 um 100 verringern.
<I	// Abfragen, ob der Wert des Akkumulators 2 kleiner als der Wert des Akkumulators 1 ist.
SPB NEXT	// Bei einem Abfrageergebnis = "1" zur Sprungmarke NEXT springen.
	// Bei einem Abfrageergebnis = "0" die nächstfolgende Anweisung bearbeiten.
T "Tag_Result_2"	Inhalt des Akkumulators 1 in den Operanden "Tag_Result_2" übertragen.
L "Tag_Value_4"	// Wert des Operanden "Tag_Value_4" laden.
+ L#20	// Inhalt des Akkumulators 1 um 20 erhöhen.
	// Die Addition erfolgt entsprechend einer DINT-Rechnung.
T "Tag_Result_3"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result_3" übertragen.
NEXT: L "Tag_Value_5"	// Sprungmarke "NEXT"
	// Wert des Operanden "Tag_Value_5" laden.
+D	// Werte des Akkumulators 1 und 2 addieren.
T "Tag_Result_4"	// Summe in den Operanden "Tag_Result_4" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

INC: Inkrementieren

Beschreibung

Mit der Anweisung "Inkrementieren" erhöhen Sie den im Akkumulator 1 stehenden Wert um den Wert, der als Parameter der Anweisung angegeben wird. Der angegebene Wert kann in einem Wertebereich von 0 bis 255 liegen. Das Erhöhen erfolgt nur im rechten Byte des Akkumulators 1. Ein Übertrag auf die links liegenden Bytes findet nicht statt.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Der Inhalt des Akkumulators 2 bleibt unverändert.

Wenn der Parameterwert außerhalb des Wertebereichs liegt, dann wird das Ergebnis verfälscht ohne eine Fehlermeldung zu generieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

Parameter	Format	Beschreibung
<Konstante>	8-Bit Ganzzahl	Wert, um den erhöht wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value"                // Wert des Operanden in Akkumulator 1 laden.
INC 5                        // Wert des Akkumulators 1 um 5 erhöhen.
T "Tag_Result"               // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

DEC: Dekrementieren

Beschreibung

Mit der Anweisung "Dekrementieren" verringern Sie den im Akkumulator 1 stehenden Wert um den Wert, der als Parameter der Anweisung angegeben wird. Der angegebene Wert kann in einem Wertebereich von 0 bis 255 liegen. Das Verringern erfolgt nur im rechten Byte des Akkumulators 1. Ein Übertrag auf die links liegenden Bytes findet nicht statt.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Der Inhalt des Akkumulators 2 bleibt unverändert.

Wenn der Parameterwert außerhalb des Wertebereichs liegt, dann wird das Ergebnis verfälscht ohne eine Fehlermeldung zu generieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

Parameter	Format	Beschreibung
<Konstante>	8-Bit Ganzzahl	Wert, um den verringert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden in Akkumulator 1 laden.
DEC 5	// Wert des Akkumulators 1 um 5 verringern.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1 und speichern den Divisionsrest.

Die Anweisung interpretiert die Werte der Akkumulatoren 1 und 2 als 32-Bit Ganzzahlen. Den Divisionsrest speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Divisionsrest negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Ergebnis den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
MOD	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Divisionsrest in den Operanden "Tag_Result" übertragen.

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Divisionsrest gewinnen":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Divisionsrest gewinnen":

Akkumulator	Wert
Akkumulator 1	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Gleitpunktzahlen

+R: Gleitpunktzahlen addieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen addieren" addieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als Gleitpunktzahlen. Die Summe speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Addition laden.
L "Tag_Value_2"	// Zweiten Wert der Addition laden.
+R	// Werte addieren
T "Tag_Result"	// Summe in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

-R: Gleitpunktzahlen subtrahieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen subtrahieren" subtrahieren Sie den Wert des Akkumulators 1 vom Wert des Akkumulators 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als Gleitpunktzahlen. Die Differenz speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Subtraktion in Akkumulator 1 laden.
-R	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

*R: Gleitpunktzahlen multiplizieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen multiplizieren" multiplizieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als Gleitpunktzahlen. Das Produkt speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Multiplikation laden.
L "Tag_Value_2"	// Zweiten Wert der Multiplikation laden.
*R	// Werte multiplizieren
T "Tag_Result"	// Produkt in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

/R: Gleitpunktzahlen dividieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen dividieren" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als Gleitpunktzahlen. Das ganzzahlige Ergebnis der Division (Quotient) speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Der Inhalt des Akkumulators 2 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
/R	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Quotienten in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" bilden Sie den Absolutwert des im Akkumulator 1 stehenden Werts. Die Anweisung interpretiert den Inhalt des Akkumulators 1 als eine Gleitpunktzahl und setzt das Vorzeichen der Mantisse auf "0". Das Ergebnis wird im Akkumulator 1 gespeichert.

Die Anweisung "Absolutwert bilden" beeinflusst keine Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Gleitpunktzahl in Akkumulator 1 laden.

AWL	Erläuterung
ABS	// Absolutwert bilden
T "Tag_Result"	// Ergebnis in den Operanden "Tag_Result" transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	-1.5E+02
Tag_Result	1.5E+02

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Gleitpunktzahlen erweitert

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" berechnen Sie das Quadrat einer im Akkumulator 1 stehenden Gleitpunktzahl. Das Ergebnis wird im Akkumulator 1 gespeichert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
SQR	// Quadrat bilden
	// Ergebnis im Akkumulator 1 speichern.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie die Quadratwurzel der Gleitpunktzahl, die in Akkumulator 1 steht. Das Ergebnis wird im Akkumulator 1 gespeichert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn im Akkumulator 1 ein Wert kleiner Null oder eine ungültige Gleitpunktzahl steht, schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Wenn im Akkumulator 1 der Wert "0" steht, liefert die Anweisung den Wert "0" zurück.

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
+R	// Werte der Akkumulatoren 1 und 2 addieren.
	// Summe im Akkumulator 1 speichern.
SQRT	// Quadratwurzel ziehen

AWL	Erläuterung
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2.718282) und der im Akkumulator 1 stehenden Gleitpunktzahl. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
EXP	// Exponentialwert bilden
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus der im Akkumulator 1 stehenden Gleitpunktzahl den natürlichen Logarithmus zur Basis e (e = 2.718282).

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn im Akkumulator 1 ein Wert kleiner oder gleich Null steht, schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.`
`LN`	`// Natürlichen Logarithmus bilden.`
	`// Ergebnis im Akkumulator 1 speichern.`
`T "Tag_Result"`	`// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels, der im Akkumulator 1 steht. Der Winkel muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
SIN	// Sinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl in einem Wertebereich von "-1" bis "1" vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von $-\pi/2$ bis $+\pi/2$ ($\pi = 3{,}14159$) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
ASIN	// Arcussinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus des Winkels, der im Akkumulator 1 steht. Die Größe des Winkels muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
COS	// Cosinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl in einem Wertebereich von "-1" bis "1" vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von "0" bis "π" (π = 3,14159) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
ACOS	// Arcuscosinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens des Winkels, der im Akkumulator 1 steht. Die Größe des Winkels muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
TAN	// Tangenswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.

AWL	Erläuterung
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von $-\pi/2$ bis $+\pi/2$ ($\pi = 3{,}14159$) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte des Akkumulators 2 bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
ATAN	// Arcustangenswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.

AWL	Erläuterung
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu AWL (Seite 7557)

Laden und Transferieren

Laden

L: Laden

Beschreibung

Mit der Anweisung "Laden" laden Sie den Inhalt eines angegebenen Operanden in den Akkumulator 1. Der zu ladende Operand kann byte-, wort- oder doppelwortweise in einem der folgenden Speicherbereiche adressiert sein:

- Prozessabbild der Eingänge und Ausgänge (E, A)
- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)
- Zeiger
- Peripherie (PE)
- Timer (T)
- Zähler (Z)

Der Speicherbereich des Akkumulators 1 ist byteweise organisiert und 32 Bits breit.

Den Inhalt eines zu ladenden Operanden im Byte-Format schreibt die Anweisung rechtsbündig in den Akkumulator 1. Die restlichen Bytes im Akkumulator 1 werden mit "0" aufgefüllt.

Den Inhalt eines zu ladenden Operanden im Wort-Format schreibt die Anweisung in das rechte Wort des Akkumulators 1. Das höher adressierte Byte wird hierbei in das rechte Byte (Bits 0 bis 7) des Akkumulators 1 übertragen. Das niedriger adressierte Byte wird links daneben geschrieben. Die restlichen Bytes im linken Wort des Akkumulators 1 werden mit "0" aufgefüllt.

Den Inhalt eines zu ladenden Operanden im Doppelwort-Format schreibt die Anweisung in den 32 Bits des Akkumulators 1. Das am höchsten adressierte Byte wird hierbei in das rechte

Byte (Bits 0 bis 7) des Akkumulators 1 übertragen. Das am niedrigsten adressierte Byte wird in die Bits 24 bis 31 geschrieben.

Die folgende Tabelle zeigt beispielsweise, wie der Inhalt des Akkumulators 1 beim Laden von Operanden in Byte-, Wort-, und Doppelwortformat verändert wird:

Anweisung	Akkumulator 1							
	31 . . .	. . . 24	23 . . .	. . . 16	15 . . .	. . . 8	7 . . .	. . . 0
Laden eines Bytes: L MB10	0000	0000	0000	0000	0000	0000	<MB10>	
Laden eines Worts: L MW10	0000	0000	0000	0000	<MB10>		<MB11>	
Laden eines Doppelworts: L MD10	<MB10>		<MB11>		<MB12>		<MB13>	

Beim Laden des angegebenen Operanden wird der vorherige Inhalt des Akkumulators 1 in den Akkumulator 2 verschoben. Die Anweisung "Laden" überträgt dabei den gesamten Inhalt des Akkumulators 1 zum Akkumulator 2. Der vorherige Inhalt des Akkumulators 2 geht dabei verloren.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, Zeiten, REAL, DATE, TOD, CHAR	E, A, PE, M, L, DB, DI, T, Z, Zeiger, Parameter	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
+R	// Werte der Akkumulatoren 1 und 2 addieren.
	// Summe im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

L STW: Statuswort in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Statuswort in Akkumulator 1 laden" laden Sie das Statuswort in den Akkumulator 1. Bei der Ausführung schreibt die Anweisung die einzelnen Statusbits in das rechte Wort des Akkumulators 1. Die restlichen Bits des Akkumulators 1 werden mit "0" beschrieben.

Die folgende Tabelle zeigt den Inhalt des Akkumulators nach der Ausführung der Anweisung "Statuswort in Akkumulator 1 laden":

	Akkumulator 1									
Bitnummer	31-9	8	7	6	5	4	3	2	1	0
Inhalt	0	BIE	A1	A0	OV	OS	0	0	VKE	0

Die Anweisung wird unabhängig vom Signalzustand des Statusbits ausgeführt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L STW	// Statuswort in den Akkumulator 1 laden.
T "Tag_STW"	// Inhalt des Akkumulators 1 (Statusbits) in den Operanden "Tag_STW" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

LAR1: AR1 mit Inhalt von Akkumulator 1 laden

Beschreibung

Mit der Anweisung "AR1 mit Inhalt von Akkumulator 1 laden" laden Sie das Adressregister 1 (AR1) mit dem Inhalt des Akkumulators 1. Der Wert des Akkumulators 1 muss dem Format eines Bereichszeigers (POINTER) entsprechen.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L P#10.0	// Zeiger (P#10.0) in Akkumulator 1 laden.
LAR1	// Adressregister 1 mit dem Inhalt des Akkumulators 1 laden.
L MW [AR1,P#4.0]	// MW14 in den Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR1,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

LAR1 <D>: AR1 mit Doppelwort oder Bereichszeiger laden

Beschreibung

Mit der Anweisung "AR1 mit Doppelwort oder Bereichszeiger laden" laden Sie das Adressregister 1 (AR1) mit dem Inhalt eines Doppelworts oder eines bereichsinternen oder bereichsübergreifenden Zeigers. Der Inhalt des Doppelworts muss dabei dem Format eines Bereichszeigers entsprechen.

Der Zeiger und das Doppelwort können einen der folgenden Speicherbereiche adressieren:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR1 mit Doppelwort oder Bereichszeiger laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Input	DWORD, POINTER	D, M, L	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#10.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
L MW [AR1,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR1,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

AWL	Erläuterung
= "Tag_Output"	// Bei erfüllter Bedingung (VKE="1") den Operanden "Tag_Output" auf "1" setzen.
LAR1 MD20	// Adressregister 1 mit dem Inhalt von MD20 laden (MD20 = P#30.0).
U E [AR1,P#2.1]	// Signalzustand des Bits E32.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

LAR1 AR2: AR1 mit Inhalt von AR2 laden

Beschreibung

Mit der Anweisung "AR1 mit Inhalt von AR2 laden" laden Sie den Inhalt des Adressregisters 2 (AR2) in das Adressregister 1 (AR1).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
LAR1 AR2	// Inhalt des Adressregisters 2 in das Adressregister 1 kopieren.
L MW [AR1,P#4.0]	// MW24 in Akkumulator 1 laden.
T "Tag_2"	// Transferiere Akkumulator 1 in "Tag_2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

LAR2: AR2 mit Inhalt von Akkumulator 1 laden

Beschreibung

Mit der Anweisung "AR2 mit Inhalt von Akkumulator 1 laden" laden Sie das Adressregister 2 (AR2) mit dem Inhalt des Akkumulators 1. Der Wert des Akkumulators 1 muss dem Format eines Bereichszeigers (POINTER) entsprechen.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L P#10.0	// Zeiger (P#10.0) in Akkumulator 1 laden.
LAR2	// Adressregister 2 mit dem Inhalt des Akkumulators 1 laden.
L MW [AR2,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR2,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

LAR2 <D>: AR2 mit Doppelwort oder Bereichszeiger laden

Beschreibung

Mit der Anweisung "AR2 mit Doppelwort oder Bereichszeiger laden" laden Sie das Adressregister 2 (AR2) mit dem Inhalt eines Doppelworts oder eines bereichsinternen oder bereichsübergreifenden Zeigers. Der Inhalt des Doppelworts muss dabei dem Format eines Bereichszeigers (POINTER) entsprechen.

Der Zeiger und das Doppelwort können einen der folgenden Speicherbereiche adressieren:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR2 mit Doppelwort oder Bereichszeiger laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Input	DWORD, POINTER	D, M, L	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#10.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
L MW [AR2,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR2,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.
LAR2 MD20	// Adressregister 2 mit dem Inhalt von MD20 laden (MD20 = P#30.0).
U E [AR2,P#2.1]	// Signalzustand des Bits E32.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Transferieren

T: Transferieren

Beschreibung

Mit der Anweisung "Transferieren" übertragen Sie den Inhalt des Akkumulators 1 byte-, wort-, oder doppelwortweise in einen angegebenen Operanden. Der Inhalt des Akkumulators 1 wird dabei nicht verändert. Der angegebene Operand muss in einem der folgenden Speicherbereiche adressiert sein:

- Prozessabbild der Eingänge und Ausgänge (E, A)
- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)
- Peripherie (PA)

Die Anzahl der zu übertragenden Bytes hängt vom Format des angegebenen Operanden ab. Wenn der Operand im Byte-Format vorliegt, überträgt die Anweisung den Inhalt der Bits 0 bis 7 des Akkumulators 1.

Wenn der angegebene Operand im Wort-Format vorliegt, überträgt die Anweisung den Inhalt des rechtsbündig im Akkumulator 1 stehenden Worts. Das rechte Byte des Akkumulators 1 wird in das höher adressierte Byte des Operanden übertragen. In das niedriger adressierte Byte des Operanden werden die Bits 8 bis 15 des Akkumulators 1 transferiert.

Wenn der angegebene Operand im Doppelwort-Format vorliegt, überträgt die Anweisung den gesamten Inhalt des Akkumulators 1. Das rechte Byte des Akkumulators 1 wird in das am höchsten adressierte Byte des Operanden übertragen. In das am niedrigsten adressierte Byte des Operanden werden die Bits 24 bis 31 des Akkumulators 1 transferiert.

Die folgende Tabelle zeigt ein Beispiel, wie der Inhalt des Akkumulators 1 byte-, wort- und doppelwortweise transferiert wird:

Anweisung	Akkumulator 1			
Bits	31 24	23 16	15 8	7 0
Inhalt des Akkumulators 1	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)

Anweisung	Akkumulator 1			
Transferieren eines Bytes				Byte (n)
Transferieren eines Worts			Byte (n)	Byte (n+1)
Transferieren eines Doppelworts	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, Zeiten, REAL, DATE, TOD, CHAR	E, A, PA, M, L, DB, DI	Operand, in den der Inhalt des Akkumulators 1 übertragen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden "Tag_Value_1" in Akkumulator 1
                            laden.
L "Tag_Value_2"             // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                            schieben.
                            // Wert des Operanden "Tag_Value_2" in Akkumulator 1
                            laden.
*R                          // Werte der Akkumulatoren 1 und 2 multiplizieren.
                            // Produkt im Akkumulator 1 speichern.
T "Tag_Result_1"            // Produkt (Inhalt des Akkumulators 1) in den Operan-
                            den "Tag_Result_1" übertragen.
L "Tag_Value_3"             // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                            schieben.
                            // Wert des Operanden "Tag_Value_3" in Akkumulator 1
                            laden.
+R                          // Werte der Akkumulatoren 1 und 2 addieren.
                            // Summe im Akkumulator 1 speichern.
T "Tag_Result_2"            // Summe (Inhalt des Akkumulators 1) in den Operanden
                            "Tag_Result_2" übertragen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

T STW: Akkumulator 1 in Statuswort transferieren

Beschreibung

Mit der Anweisung "Akkumulator 1 in Statuswort transferieren" übertragen Sie die Bits 0 bis 8 des Akkumulators 1 in das Statuswort.

Die folgende Tabelle zeigt die Bits im Akkumulator 1, mit denen die Statusbits nach der Ausführung der Anweisung überschrieben werden:

	Akkumulator 1								
Bitnummer	8	7	6	5	4	3	2	1	0
Statusbits	BIE	A1	A0	OV	OS	0	0	VKE	0

Die Anweisung wird unabhängig vom Signalzustand des Statusbits ausgeführt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_STW"`	`// Wert des Operanden "Tag_STW" in den Akkumulator 1 laden.`
`T STW`	`// Bit 0 bis 8 des Akkumulators 1 in das Statuswort übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR: AR1 und AR2 tauschen

Beschreibung

Mir der Anweisung "AR1 und AR2 tauschen" tauschen Sie die Inhalte der Adressregister 1 (AR1) und 2 (AR2).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#10.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
TAR	// Inhalt der Adressregister tauschen.
L MD [AR1,P#2.0]	// MD22 in Akkumulator 1 laden.
T MD [AR2,P#2.0]	// MD12 in Akkumulator 2 transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR1: AR1 in Akkumulator 1 transferieren

Beschreibung

Mit der Anweisung "AR1 in Akkumulator 1 transferieren" übertragen Sie den Inhalt des Adressregisters 1 (AR1) in den Akkumulator 1. Der Inhalt des Akkumulators 1 wird dabei in den Akkumulator 2 verschoben.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#15.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#15.0 laden.
TAR1	// Inhalt des Adressregisters 1 (P#15.0) in den Akkumulator 1 transferieren.
T "Tag_Pointer"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Pointer" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR1 <D>: AR1 in Doppelwort transferieren

Beschreibung

Mit der Anweisung "AR1 in Doppelwort transferieren" übertragen Sie den Inhalt des Adressregisters 1 (AR1) in ein Doppelwort. Das Doppelwort muss dabei in einem der folgenden Speicherbereiche adressiert sein:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR1 in Doppelwort transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Output	DWORD	D, M, L	Operand, in welchen der Inhalt des Adressregisters 1 transferiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
TAR1 %DBD20	// Inhalt des Adressregisters 1 in das Datendoppelwort DBD20 übertragen.
TAR1 %DID30	// Inhalt des Adressregisters 1 in das Instanzdoppelwort DID30 übertragen.
TAR1 %LD18	// Inhalt des Adressregisters 1 in das Lokaldaten-Doppelwort LD18 übertragen.
TAR1 %MD24	// Inhalt des Adressregisters 1 in das Merkerdoppelwort MD24 übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR1 AR2: AR1 in AR2 transferieren

Beschreibung

Mit der Anweisung "AR1 in AR2 transferieren" kopieren Sie den Inhalt des Adressregisters 1 (AR1) in das Adressregister 2 (AR2).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#10.4	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.4 laden.
TAR1 AR2	// Inhalt des Adressregisters 1 in das Adressregister 2 transferieren.
U E [AR2,P#3.2]	// Signalzustand des Bits E13.6 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag1"	// Operanden "Tag_1" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR2: AR2 in Akkumulator 1 transferieren

Beschreibung

Mit der Anweisung "AR2 in Akkumulator 1 transferieren" übertragen Sie den Inhalt des Adressregisters 2 (AR2) in den Akkumulator 1. Der Inhalt des Akkumulators 1 wird dabei in den Akkumulator 2 verschoben.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
TAR2	// Inhalt des Adressregisters 2 (P#20.0) in den Akkumulator 1 transferieren.
T "Tag_Pointer"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Pointer" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAR2 <D>: AR2 in Doppelwort transferieren

Beschreibung

Mit der Anweisung "AR2 in Doppelwort transferieren" übertragen Sie den Inhalt des Adressregisters 2 (AR2) in ein Doppelwort. Das Doppelwort muss dabei in einem der folgenden Speicherbereiche adressiert sein:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR2 in Doppelwort transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Output	DWORD	D, M, L	Operand, in welchen der Inhalt des Adressregisters 2 transferiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
TAR2 %DBD20	// Inhalt des Adressregisters 2 in das Datendoppelwort DBD20 übertragen.
TAR2 %DID30	// Inhalt des Adressregisters 2 in das Instanzdoppelwort DID30 übertragen.
TAR2 %LD18	// Inhalt des Adressregisters 2 in das Lokaldaten-Doppelwort LD18 übertragen.
TAR2 %MD24	// Inhalt des Adressregisters 2 in das Merkerdoppelwort MD24 übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Umwandler

BTI: BCD in Ganzzahl (16 Bit) umwandeln

Beschreibung

Mit der Anweisung "BCD in Ganzzahl (16 Bit) umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine 16-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine dreistellige binär codierte Dezimalzahl (BCD).

Die Bits 1 bis 11 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-999" bis "+999".

Das Vorzeichen des Ergebniswerts wird aus dem Bit 15 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" des Bits ist das Vorzeichen positiv. Wenn der Signalzustand des Bits "1" ist, ist das Vorzeichen negativ. Die Bits 12 bis 14 werden bei der Umwandlung nicht verwendet.

Wenn im BCD-Code ein BCD-Fehler enthalten ist, setzt die CPU das OV und das OS Bit im Statuswort.

Das Ergebnis der Umwandlung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// BCD in Akkumulator 1 laden.`
`BTI`	`// BCD in eine 16-Bit Ganzzahl umwandeln.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

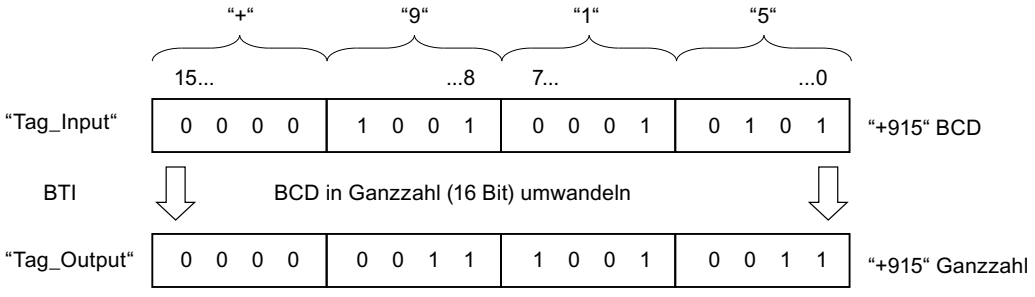

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

ITB: Ganzzahl (16 Bit) in BCD umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) in BCD umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine dreistellige binär codierte Dezimalzahl (BCD) um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 16-Bit Ganzzahl.

Die Bits 0 bis 11 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-999" bis "+999". Wenn der umzuwandelnde Wert außerhalb dieses Bereichs liegt, werden die Bits OV und OS auf den Signalzustand "1" gesetzt. Die Umwandlung wird in diesem Fall nicht durchgeführt.

Das Vorzeichen des Ergebniswerts wird aus den Bits 12 bis 15 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" der Bits ist das Vorzeichen positiv. Wenn der Signalzustand aller vier Bits "1" ist, ist das Vorzeichen negativ.

Das Ergebnis der Umwandlung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// Ganzzahl (16 Bit) in Akkumulator 1 laden.
ITB	// Ganzzahl in BCD umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

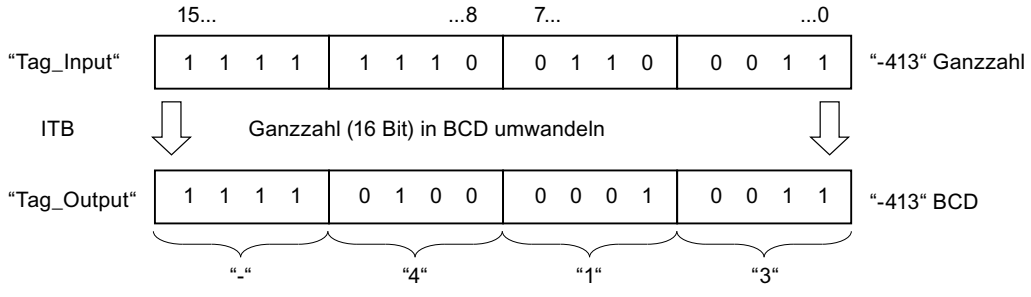

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

BTD: BCD in Ganzzahl (32 Bit) umwandeln

Beschreibung

Mit der Anweisung "BCD in Ganzzahl (32 Bit) umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine siebenstellige binär codierte Dezimalzahl (BCD).

Die Bits 0 bis 27 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-9999999" bis "+9999999".

Das Vorzeichen des Ergebniswerts wird aus dem Bit 31 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" des Bits ist das Vorzeichen positiv. Wenn der Signalzustand des Bits "1" ist, ist das Vorzeichen negativ. Die Bits 28 bis 30 werden bei der Umwandlung nicht verwendet.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Wenn im BCD-Code ein BCD-Fehler enthalten ist, setzt die CPU das OV und das OS Bit im Statuswort.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// BCD in Akkumulator 1 laden.
BTD	// BCD wird in eine 32-Bit Ganzzahl umgewandelt.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

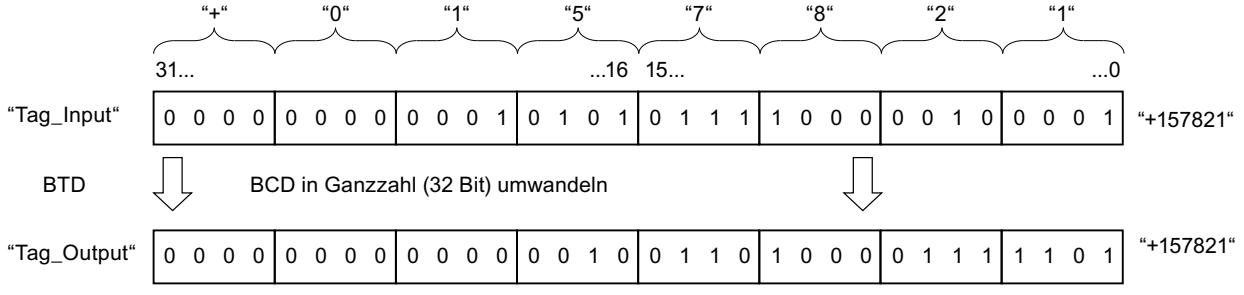

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

ITD: Ganzzahl (16 Bit) in Ganzzahl (32 Bit) umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) in Ganzzahl (32 Bit) umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 16-Bit Ganzzahl.

Bei der Umwandlung werden die Bits 0 bis 15 unverändert übernommen. Die Bits 16 bis 31 des Ergebniswerts werden mit dem Signalzustand des Bits 15 aufgefüllt.

Der Ergebniswert wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// Ganzzahl (16 Bit) in Akkumulator 1 laden .
ITD	// 16-Bit Ganzzahl in eine 32-Bit Ganzzahl umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Akkumulator 1							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Vor der Ausführung	XXXX	XXXX	XXXX	XXXX	1111	1111	1111	0110
Nach der Ausführung	1111	1111	1111	1111	1111	1111	1111	0110
	X : Der Signalzustand der Bits ist für die Umwandlung irrelevant							

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

DTB: Ganzzahl (32 Bit) in BCD umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) in BCD umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine siebenstellige binär codierte Dezimalzahl (BCD) um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 32-Bit Ganzzahl.

Die Bits 0 bis 27 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von"-9999999" und "+9999999". Wenn der umzuwandelnde Wert außerhalb dieses Bereichs liegt, werden die Bits OV und OS auf den Signalzustand "1" gesetzt. Die Umwandlung wird in diesem Fall nicht durchgeführt.

Das Vorzeichen des Ergebniswerts wird aus den Bits 28 bis 31 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" der Bits ist das Vorzeichen positiv. Wenn der Signalzustand aller vier Bits "1" ist, ist das Vorzeichen negativ.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// 32-Bit Ganzzahl in Akkumulator 1 laden.
DTB	// 32-Bit Ganzzahl in BCD umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

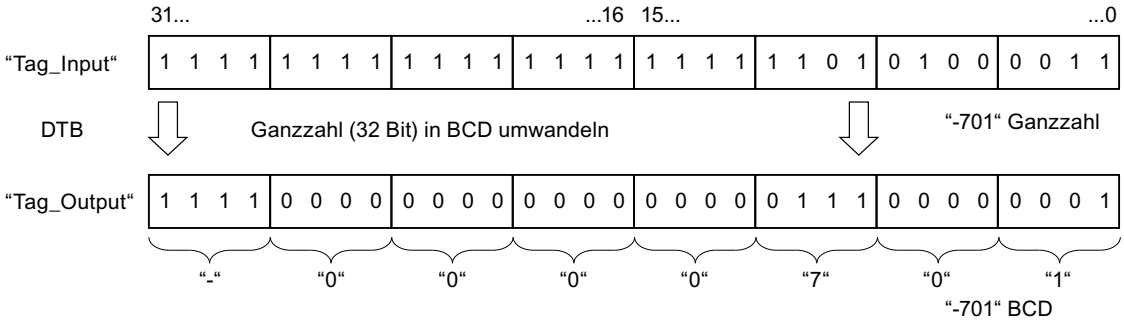

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

DTR: Ganzzahl (32 Bit) in Gleitpunktzahl umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) in Gleitpunktzahl umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine Gleitpunktzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 32-Bit Ganzzahl.

Da eine 32-Bit Ganzzahl eine höhere Genauigkeit als eine Gleitpunktzahl hat, wird das Ergebnis der Umwandlung zur nächsten darstellbaren Zahl gerundet.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// 32-Bit Ganzzahl in Akkumulator 1 laden.`
`DTR`	`// 32-Bit Ganzzahl in eine Gleitpunktzahl umwandeln.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

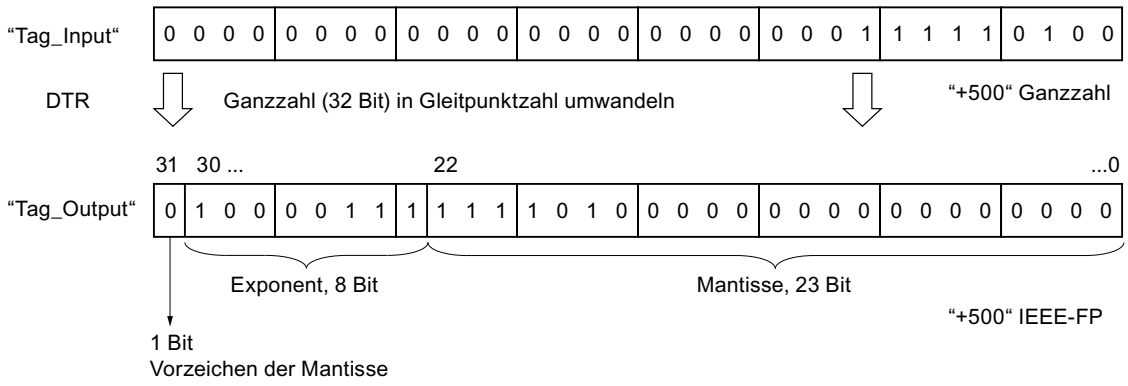

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

INVI: Einerkomplement zu Ganzzahl (16 Bit) erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement zu Ganzzahl (16-Bit) erzeugen" negieren Sie den im rechten Wort des Akkumulators 1 stehenden Wert Bit für Bit.

Bei der Ausführung invertiert die Anweisung den Signalzustand jedes einzelnen Bits im rechten Wort des Akkumulators 1. Die Einsen werden durch Nullen ersetzt und die Nullen durch Einsen.

Das Ergebnis wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Wert in Akkumulator 1 laden.`
`INVI`	`// Einerkomplement erzeugen`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	0011	1010	1110
Tag_Output	1001	1100	0101	0001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

INVD: Einerkomplement zu Ganzzahl (32 Bit) erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement zu Ganzzahl (32-Bit) erzeugen" negieren Sie den im Akkumulator 1 stehenden Wert Bit für Bit.

Bei der Ausführung invertiert die Anweisung den Signalzustand jedes einzelnen Bits im Akkumulator 1. Die Einsen werden durch Nullen ersetzt und die Nullen durch Einsen.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Input"                // Wert in Akkumulator 1 laden.
INVD                         // Einerkomplement erzeugen
T "Tag_Output"               // Ergebnis in den Operanden "Tag_Output" transferie-
                             ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	1111	1000	1100	0110	0011	1010	1110
Tag_Output	1001	0000	0111	0011	1001	1100	0101	0001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

NEGI: Ganzzahl (16 Bit) negieren

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) negieren" wechseln Sie das Vorzeichen des im rechten Wort des Akkumulators 1 stehenden Werts durch Zweierkomplementbildung. Den Wert des Akkumulators 1 interpretiert die Anweisung als 16-Bit Ganzzahl.

Die Ausführung der Anweisung ist gleichbedeutend mit einer Multiplikation mit "-1". Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Die Anweisung "Ganzzahl (16 Bit) negieren" beeinflusst die Statusbits A0, A1, OV und OS. Die folgende Tabelle zeigt, wie die Anweisung die Statusbit abhängig vom Ergebnis beeinflusst:

Ergebnis:	A1	A0	OV	OS
+1 bis +32 767	1	0	0	-
0	0	0	0	-
-1 bis -32 767	0	1	0	-
(-) 32 768	0	1	1	1

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Input"                // Wert in Akkumulator 1 laden.
NEGI                         // 16-Bit Ganzzahl negieren
T "Tag_Output"               // Ergebnis in den Operanden "Tag_Output" transferie-
                             ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1101	0011	1000
Tag_Output	1010	0010	1100	1000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

NEGD: Ganzzahl (32 Bit) negieren

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) negieren" wechseln Sie das Vorzeichen des im Akkumulator 1 stehenden Werts durch Zweierkomplementbildung. Den Wert des Akkumulators 1 interpretiert die Anweisung als 32-Bit Ganzzahl.

Die Ausführung der Anweisung ist gleichbedeutend mit einer Multiplikation mit "-1". Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Anweisung "Ganzzahl (32 Bit) negieren" beeinflusst die Statusbits A0, A1, OV und OS. Die folgende Tabelle zeigt, wie die Anweisung die Statusbit abhängig vom Ergebnis beeinflusst:

Ergebnis	A1	A0	OV	OS
+1 bis +2147483647	1	0	0	-
0	0	0	0	-
-1 bis -2147483647	0	1	0	-
(-) 2147483648	0	1	1	1

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Input"                // Wert in Akkumulator 1 laden.
NEGD                         // 32-Bit Ganzzahl negieren.
T "Tag_Output"               // Ergebnis in den Operanden "Tag_Output" transferie-
                             ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Status	Wert							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1111	0110	0100	0101	1101	0011	1000
Tag_Output	1010	0000	1001	1011	1010	0010	1100	1000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

NEGR: Gleitpunktzahl negieren

Beschreibung

Mit der Anweisung "Gleitpunktzahl negieren" invertieren Sie das Vorzeichen der Mantisse und ist gleichbedeutend mit einer Multiplikation mit "-1".

Den Wert des Akkumulators 1 interpretiert die Anweisung als eine Gleitpunktzahl. Im Bit 31 der Gleitpunktzahl steht das Vorzeichen der Mantisse.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Anweisung beeinflusst keine Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`NEGR`	`// Gleitpunktzahl negieren.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Input	1.5E+02
Tag_Output	-1.5E+02

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAW: Bytes im rechten Wort des Akkumulators 1 tauschen

Beschreibung

Mit der Anweisung "Bytes im rechten Wort des Akkumulators 1 tauschen" tauschen Sie die Reihenfolge der beiden rechten Bytes im rechten Wort des Akkumulators 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert B	Wert D	Wert C

Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Die Bytes im linken Wort des Akkumulators 1 werden durch die Anweisung nicht beeinflusst und bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Wert in Akkumulator laden.`
`TAW`	`// Bytes im rechten Wort des Akkumulators 1 tauschen.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Input	0000	1111	0000	1111
Tag_Output	0000	1111	1111	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TAD: Bytes im gesamten Akkumulator 1 tauschen

Beschreibung

Mit der Anweisung "Bytes im gesamten Akkumulator 1 tauschen" tauschen Sie die Reihenfolge der Bytes im Akkumulator 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert D	Wert C	Wert B	Wert A

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Input"                // Wert des Operanden in Akkumulator 1 laden.
TAD                          // Bytes des Akkumulators 1 tauschen.
T "Tag_Output"               // Ergebnis in den Operanden "Tag_Output" transferie-
                             ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Input	1111	0000	0000	1111	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000	0000	1111	1111	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

RND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächstliegenden Ganzzahl.

Wenn die Gleitpunktzahl genau zwischen gerader und ungerader Zahl liegt, wird die gerade Zahl als Ergebnis gewählt.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND`	`// Gleitpunktzahl zu einer Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	101.5	-101.5
Tag_Output	102	-102

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl umwandeln. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und liefert deren ganzzahligen Anteil als Ergebnis zurück.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`TRUNC`	`// Gleitpunktzahl in eine Ganzzahl vom Datentyp DINT umwandeln.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	101.5	-101.5
Tag_Output	101	-101

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

RND+: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächst höheren Ganzzahl. Das Ergebnis der Anweisung ist eine Zahl vom Datentyp DINT, die größer oder gleich der umzuwandelnden Gleitpunktzahl ist.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND+`	`// Gleitpunktzahl zur nächst höheren Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	100.5	-100.5
Tag_Output	101	-100

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

RND-: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächst niederen Ganzzahl. Das Ergebnis der Anweisung ist eine Zahl vom Datentyp DINT, die kleiner oder gleich der umzuwandelnden Gleitpunktzahl ist.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND-`	`// Gleitpunktzahl zur nächst niederen Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	100.5	-100.5
Tag_Output	100	-101

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmsteuerung

Sprünge

Sprungmarke

Beschreibung

Mit einer Sprungmarke können Sie die Programmstelle kennzeichnen, an der bei einem ausgeführten Sprung die Programmbearbeitung fortgesetzt wird. Die Bezeichnung der Sprungmarke kann aus bis zu 128 Buchstaben, Ziffern oder Unterstrichen bestehen.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden. Sie können maximal 256 Sprungmarken deklarieren.

Es kann sowohl vorwärts als auch rückwärts gesprungen werden.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL	// Bei einem Abfrageergebnis "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	//Bei einem Abfrageergebnis "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" laden.
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Operanden "Tag_Input_4" auf "1" abfragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPA: Absolut springen

Beschreibung

Mit der Anweisung "Absolut springen" unterbrechen Sie die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Die Anweisung wird immer, unabhängig von Bedingungen ausgeführt.

Die Anweisung "Absolut springen" beeinflusst die Statusbits nicht.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolut springen":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL_1	// Bei einem Abfrageergebnis "1" zur Sprungmarke "MyLABEL_1" springen und die Programmbearbeitung dort fortsetzen.
	//Bei einem Abfrageergebnis "0" lineare Programmbearbeitung fortsetzen.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
SPA MyLABEL_2	// Zur Sprungmarke "MyLABEL_2" springen und die Programmbearbeitung dort fortsetzen.

AWL	Erläuterung
MyLABEL_1:	// Sprungmarke
L "Tag_Input_4"	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.
MyLABEL_2: U "Tag_Input_5"	// Sprungmarke "MyLABEL_2"
	// Operanden "Tag_Input_5" auf "1" abfragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

SPB: Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie, abhängig vom Verknüpfungsergebnis, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "1" ist. Wenn das aktuelle VKE vor der Anweisung "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei VKE = 1" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE auf "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL	// Bei VKE = "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
MyLABEL: L "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

SPBN: Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "0" ist. Wenn das aktuelle VKE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei VKE = 0" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE auf "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 0":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Operanden "Tag_Input_1" auf "1" abfragen und nach UND verknüpfen.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und nach UND verknüpfen.
SPBN MyLABEL	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "1" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" laden.
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Den Operanden "Tag_Input_4" auf "1" abfragen und mit dem VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

SPBB: Springen bei VKE = 1 und VKE speichern

Beschreibung

Mit der Anweisung "Springen bei VKE = 1 und VKE speichern" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Parallel dazu wird der Signalzustand des aktuellen VKE in das Binärergebnis (BIE) kopiert.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE 1 ist. In diesem Fall setzt die Anweisung das Binärergebnis auf "1".

Wenn das aktuelle VKE vor der Anweisung "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt. In diesem Fall weist die Anweisung dem Binärergebnis den Signalzustand "0" zu.

Die Anweisung "Springen bei VKE = 1 und VKE speichern" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE auf "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 1 und VKE speichern":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPBB MyLABEL	// Aktuelles VKE in BIE kopieren.
	// Bei VKE = "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	//Bei VKE = "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
MyLABEL: L "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPBNB: Springen bei VKE = 0 und VKE speichern

Beschreibung

Mit der Anweisung "Springen bei VKE = 0 und VKE speichern" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Parallel dazu wird der Signalzustand des aktuellen Verknüpfungsergebnisses in das Binärergebnis (BIE) kopiert.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "0" ist. In diesem Fall setzt die Anweisung das Binärergebnis auf "0".

Wenn das aktuelle VKE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt. In diesem Fall weist die Anweisung dem Binärergebnis den Signalzustand "1" zu.

Die Anweisung "Springen bei VKE = 0 und VKE speichern" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE auf "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 0 und VKE speichern":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Operanden "Tag_Input_1" auf "1" abfragen und nach UND verknüpfen.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und nach UND verknüpfen.
SPBNB MyLABEL	// Aktuelles VKE in BIE kopieren.
	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "1" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" transferieren.

AWL	Erläuterung
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen. // Den Operanden "Tag_Input_4" auf "1" abfragen und mit dem VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPBI: Springen bei BIE = 1

Beschreibung

Mit der Anweisung "Springen bei BIE = 1" unterbrechen Sie abhängig vom Binärergebnis die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit BIE "1" ist. Wenn das Statusbit BIE "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei BIE = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
SET	// VKE am Anfang des Bausteins auf "1" setzen.
SAVE	// Signalzustand des VKE in das BIE-Bit übertragen.
....	// Beliebiges Programm
UN OV	// OV-Bit am Ende des Bausteins abfragen.
SAVE	// Bei einem Überlauf BIE-Bit auf "0" setzen.
	// BIE auf "1" setzen, wenn kein Überlauf vorliegt.

AWL	Erläuterung
SPBI END	// Bei BIE = "1" zur Sprungmarke END springen.
	// Bei BIE = "0" die nächstfolgende Anweisung bearbeiten.
R "Tag_Output_1"	// Operanden "Tag_Output_1" auf "0" zurücksetzen.
END: S "Tag_Output_2"	// Operanden "Tag_Output_2" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPBIN: Springen bei BIE = 0

Beschreibung

Mit der Anweisung "Springen bei BIE = 0" unterbrechen Sie, abhängig vom Binärergebnis (BIE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit BIE "0" ist. Wenn das Statusbit BIE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei BIE = 0":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
SET	// VKE am Anfang des Bausteins auf "1" setzen.
SAVE	// Signalzustand des VKE in das BIE-Bit übertragen.
....	// Beliebiges Programm
UN OV	// OV-Bit am Ende des Bausteins abfragen.

AWL	Erläuterung
SAVE	// Bei einem Überlauf BIE-Bit auf "0" setzen.
	// BIE auf "1" setzen, wenn kein Überlauf vorliegt.
SPBIN END	// Bei BIE = "0" zur Sprungmarke END springen.
	// Bei BIE = "1" die nächstfolgende Anweisung bearbeiten.
S "Tag_Output"	// Operanden "Tag_Output" auf "1" setzen.
END: BE	// Baustein beenden

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPO: Springen bei OV = 1

Beschreibung

Mit der Anweisung "Springen bei OV = 1" unterbrechen Sie, abhängig vom Statusbit OV, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit OV "1" ist. Das Statusbit OV wird gesetzt, wenn z. B. das Ergebnis einer Berechnung außerhalb des zulässigen Bereichs liegt oder beim Vergleichen von Gleitpunktzahlen ein ungültiger Wert vorliegt.

Wenn das Statusbit OV "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei OV = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert laden
L "Tag_Value_2"	// Wert laden
*I	// Werte multiplizieren
SPO OVER	// Bei einem Signalzustand "1" des Statusbits OV zur Sprungmarke "OVER" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem Signalzustand "0" des Statusbits OV nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Ergebnis der Multiplikation in den Operanden "Tag_Result" übertragen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
OVER: SET	// Programmbearbeitung bei erfüllter Bedingung (OV = "1")an dieser Stelle fortsetzen.
	// VKE auf den Signalzustand "1" setzen.
R "Tag_Output"	// Operanden "Tag_Output" auf "0" zurücksetzen.
NEXT: U "MyTag_1"	// Sprungmarke "NEXT"
	// Operanden "MyTag_1" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_2"	// Operanden "MyTag_2" auf "1" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = 1 den Operanden "Tag_Output_2" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPS: Springen bei OS = 1

Beschreibung

Mit der Anweisung "Springen bei OS = 1" unterbrechen Sie, abhängig vom Statusbit OS, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit OS "1" ist. Das Statusbit OS wird immer gesetzt, wenn ein Zahlenbereichsüberlauf das Statusbit OV auf "1" setzt. Im Gegensatz zu Statusbit OV bleibt das Statusbit OS gesetzt, auch wenn anschließend ein Ergebnis im erlaubten Zahlenbereich liegt.

Wenn das Statusbit OS "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei OS = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert laden
L "Tag_Value_2"              // Wert laden
*I                           // Geladene Werte multiplizieren.
                             // Produkt im Akkumulator 1 speichern.
L "Tag_Value_3"              // Wert laden
+I                           // Geladenen Wert mit dem Produkt addieren.
                             // Produkt im Akkumulator 1 speichern.
L "Tag_Value_4"              // Wert laden
-I                           // Geladenen Wert von der berechneten Summe subtrahie-
                             ren.
                             // Produkt im Akkumulator 1 speichern.
SPS OVER                     // Bei einem Überlauf in einer der drei vorangehenden
                             Anweisungen wird das Statusbit OS auf "1" gesetzt.
                             // Bei einem Signalzustand "1" des Statusbits OS wird
                             die Programmbearbeitung an der Stelle der Sprungmarke
                             "OVER" fortsetzt.
                             // Bei einem Signalzustand "0" des Statusbits OS wird
                             die nächstfolgende Anweisung bearbeitet.
T "Tag_Result"               // Ergebnis der gesamten Berechnung in den Operanden
                             "Tag_Result" übertragen.
SPA NEXT                     // Zur Sprungmarke "NEXT" springen und die Programmbe-
                             arbeitung dort fortsetzen.
OVER: SET                    // Programmbearbeitung bei erfüllter Bedingung (OV =
                             "1")an dieser Stelle fortsetzen.
                             // VKE auf den Signalzustand "1" setzen.
R "Tag_Output"               // Operanden "Tag_Output" auf "0" zurücksetzen.
NEXT: U "MyTag_1"            // Sprungmarke "NEXT"
                             // Operanden "MyTag_1" auf "1" abfragen und nach UND
                             verknüpfen.
```

AWL	Erläuterung
U "MyTag_2"	// Operanden "MyTag_2" auf "1" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = 1 den Operanden "Tag_Output_2" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPZ: Springen bei Ergebnis Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf hat der Akkumulator 1 den Wert "0".
- Bei der Ausführung der Anweisung "Ganzahlen (16Bit) addieren" (+I) oder "Ganzahlen (32Bit) addieren" (+D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Der Inhalt des Akkumulators 2 ist nach der Ausführung einer Vergleichsanweisung gleich dem Inhalt des Akkumulators 1.
- Der Inhalt des Akkumulators 1 ist nach der Ausführung einer Wortverknüpfung Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "0".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value"                // Wert des Operanden "Tag_Value" laden.
SRW 1                        // Inhalt des rechten Worts im Akkumulator 1 um eine
                             Stelle nach rechts schieben.
SPZ ZERO                     // Bei einem Signalzustand "0" des zuletzt hinausgesc-
                             hobenen Bits zur Sprungmarke "OVER" springen und die
                             Programmbearbeitung dort fortsetzen.
                             // Bei einem Signalzustand "1" des zuletzt hinausgesc-
                             hobenen Bits nächstfolgende Anweisung bearbeiten.
L "MyTag_1"                  // Wert des Operanden "MyTag_1" in Akkumulator 1 laden.
INC 1                        // Rechtes Byte des Akkumulators 1 mit dem Wert "1"
                             addieren.
T "Tag_Result_1"             // Ergebnis in den Operanden "Tag_Result_1" übertra-
                             gen.
SPA NEXT                     // Zur Sprungmarke "NEXT" springen und die Programmbe-
                             arbeitung dort fortsetzen.
ZERO: L "MyTag_2"            // Sprungmarke "ZERO"
                             // Wert des Operanden "MyTag_2" in Akkumulator 1 laden.
INC 1                        // Rechtes Byte des Akkumulators 1 mit dem Wert "1"
                             addieren.
T "Tag_Result_2"             // Ergebnis in den Operanden "Tag_Result_2" übertra-
                             gen.
NEXT: U "MyTag_3"            // Sprungmarke "NEXT"
                             // Operanden "MyTag_3" auf "1" abfragen und nach UND
                             verknüpfen.
U "MyTag_4"                  // Operanden "MyTag_4" auf "1" abfragen und nach UND
                             verknüpfen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPN: Springen bei Ergebnis nicht Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis nicht Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 unterschiedlich ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 ungleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, *I, +D, -D, *D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, *I, /I, +D, -D, *D, /D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven oder negativen Bereich auf.
- Der Inhalt des Akkumulators 2 ist nach der Ausführung einer Vergleichsanweisung ungleich dem Inhalt des Akkumulators 1.
- Der Inhalt des Akkumulators 1 ist nach der Ausführung einer Wortverknüpfung nicht Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "1".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis nicht Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Valu_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
XOW                          // Nach EXKLUSIV ODER verknüpfen
```

AWL	Erläuterung
SPN NOZERO	// Wenn der Wert im Akkumulator 1 ungleich Null ist, wird der Sprung zur Sprungmarke "NOZERO" ausgeführt und die Programmbearbeitung dort fortsetzt.
	// Wenn der Wert im Akkumulator 1 gleich Null ist, wird die nächstfolgende Anweisung bearbeitet.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
NOZERO: UN "MyTag_2"	// Sprungmarke "NOZERO"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei einem VKE "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPP: Springen bei Ergebnis größer Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis größer Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "0" und des Statusbits A1 "1" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 größer Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, /I, *D, /D) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven Bereich auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 größer als der Inhalt des Akkumulators 1.
- Nach der Ausführung einer Wortverknüpfung ist der Inhalt des Akkumulators 1 nicht Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "1".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis größer Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
-I	// Wert des Operanden "Tag_Value_2" vom Wert des Operanden "Tag_Value_1" subtrahieren.
SPP POSITIVE	// Bei einem positiven Ergebniswert im Akkumulator 1 zur Sprungmarke "POSITIVE" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem negativen Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.

AWL	Erläuterung
POSITIVE: UN "MyTag_2"	// Sprungmarke "POSITIVE"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPM: Springen bei Ergebnis kleiner Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis kleiner Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "1" und des Statusbits A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 kleiner Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, *D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im negativen Bereich auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 kleiner als der Inhalt des Akkumulators 1.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis kleiner Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
-I                           // Wert des Operanden "Tag_Value_2" vom Wert des Ope-
                             randen "Tag_Value_1" subtrahieren.
SPM NEGATIVE                 // Bei einem negativen Ergebniswert im Akkumulator 1
                             zur Sprungmarke "NEGATIVE" springen und die Programm-
                             bearbeitung dort fortsetzen.
                             // Bei einem positiven Ergebniswert im Akkumulator 1
                             nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"                 // Operanden "MyTag_1" auf "0" abfragen und nach UND
                             verknüpfen.
S "Tag_Output_1"             // Bei VKE = "1" den Operanden "Tag_Output_1" auf "1"
                             setzen.
SPA NEXT                     // Zur Sprungmarke "NEXT" springen und die Programmbe-
                             arbeitung dort fortsetzen.
NEGATIVE: UN "MyTag_2"       // Sprungmarke "NEGATIVE"
                             // Operanden "MyTag_2" auf "0" abfragen und nach UND
                             verknüpfen.
S "Tag_Output_2"             // Bei VKE = "1" den Operanden "Tag_Output_2" auf "1"
                             setzen.
NEXT: U "MyTag_3"            // Sprungmarke "NEXT"
                             // Operanden "MyTag_3" auf "1" abfragen und nach UND
                             verknüpfen.
U "MyTag_4"                  // Operanden "MyTag_4" auf "1" abfragen und nach UND
                             verknüpfen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPPZ: Springen bei Ergebnis größer oder gleich Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis größer oder gleich Null" unterbrechen Sie, abhängig vom Statusbit A0, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 größer oder gleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, /I, *D, /D) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 größer oder gleich dem Inhalt des Akkumulators 1.
- Eine Anweisung für Wortverknüpfung wurde ausgeführt.
- Eine Schiebeanweisung wurde ausgeführt.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis größer oder gleich Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
-I                           // Wert des Operanden "Tag_Value_2" vom Wert des Ope-
                             randen "Tag_Value_1" subtrahieren.
```

AWL	Erläuterung
SPPZ REGULAR	// Bei einem positiven Ergebniswert oder Null im Akkumulator 1 zur Sprungmarke "REGULAR" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem negativen Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
REGULAR: UN "MyTag_2"	// Sprungmarke "REGULAR"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPMZ: Springen bei Ergebnis kleiner oder gleich Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis kleiner oder gleich Null" unterbrechen Sie, abhängig vom Statusbit A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 kleiner oder gleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, *I, +D, *D) tritt ein Überlauf im negativen Bereich auf.

- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 kleiner oder gleich dem Inhalt des Akkumulators 1.
- Nach der Ausführung einer Wortverknüpfung ist der Wert des Akkumulators 1 gleich Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits gleich Null.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis kleiner oder gleich Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
-I	// Wert des Operanden "Tag_Value_2" vom Wert des Operanden "Tag_Value_1" subtrahieren.
SPMZ MyLABEL	// Bei einem negativen Ergebniswert oder Null im Akkumulator 1 zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem positiven Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Be VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL: UN "MyTag_2"	// Sprungmarke "MyLABEL"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE="1" den Operanden "Tag_Output_2" auf "1" setzen

AWL	Erläuterung
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPU: Springen bei Ergebnis ungültig

Beschreibung

Mit der Anweisung "Springen bei Ergebnis ungültig" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 "1" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Bei einer mathematischen Anweisung (/I, /D, MOD) wird es durch Null dividiert.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf auf und der Ergebniswert ist eine ungültige Gleitpunktzahl.
- Bei der Ausführung einer Vergleichsanweisung mit Gleitpunktzahlen wurde eine ungültige Gleitpunktzahl verwendet oder ist als Ergebniswert entstanden.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis ungültig":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
/I	// Wert des Operanden "Tag_Value_1" durch den Wert des Operanden "Tag_Value_2" dividieren.
SPU ERROR	// Bei einer Division durch Null zur Sprungmarke "ERROR" springen und die Programmbearbeitung dort fortsetzen.
	// In anderem Fall die nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.
U "MyTag_1"	// Operanden "MyTag_1" auf "1" abfragen und nach UND verknüpfen.
R "MyTag_1"	// Bei VKE = "1" den Operanden "MyTag_1" auf "0" zurücksetzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
ERROR: UN "MyTag_1"	// Sprungmarke "ERROR"
	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "MyTag_1"	// Bei VKE = "1" den Operanden "MyTag_1" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

SPL: Sprungliste definieren

Beschreibung

Mit der Anweisung "Sprungliste definieren" programmieren Sie eine Liste, die aus mehreren Einträgen der Anweisung "Absolut springen" (SPA) besteht. Die Liste beginnt unmittelbar nach der Anweisung "Sprungliste definieren" und darf maximal 255 Einträge enthalten. Die Nummerierung der Sprungfunktionen in der Liste beginnt bei Null. Die Sprungliste muss lückenlos programmiert werden. Das Ende der Sprungliste wird durch eine Sprungmarke gekennzeichnet, die bei der Anweisung "Sprungliste definieren" angegeben wird.

Welche Sprungfunktion der Liste ausgeführt wird, hängt vom Wert im rechten Byte des Akkumulators 1 ab. Wenn z. B. im Akkumulator 1 der Wert "0" steht, wird die erste Sprungfunktion ausgeführt. Bei einem Wert "1" im Akkumulator 1, wird die zweite Sprungfunktion ausgeführt. Wenn der Wert im Akkumulator "1" größer als die Anzahl der Listeneinträge ist, verweist die Anweisung "Sprungliste definieren" zum Ende der Liste.

Die Anweisung "Sprungverteiler"' wird unabhängig von Bedingungen ausgeführt und beeinflusst die Statusbits nicht.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Sprungnummer in Akkumulator 1 laden.
SPL END	// Anfang der Sprungliste
SPA MyLABEL_1	// Bei einem Wert "0" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_1" springen.
SPA MyLABEL_2	// Bei einem Wert "1" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_2" springen.
SPA MyLABEL_3	// Bei einem Wert "2" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_3" springen.
END: L "MyTag_1"	// Ende der Sprungliste
	// Inhalt des Operanden "MyTag_1" in Akkumulator 1 laden.
L "MyTag_2"	// Inhalt des Operanden "MyTag_2" in Akkumulator 1 laden.
+I	// Werte addieren
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.
MyLABEL_1: ...	// Sprungmarke "MyLABEL_1"
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL_2: ...	// Sprungmarke "MyLABEL_2"
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL_3: ...	// Sprungmarke "MyLABEL_3"

AWL	Erläuterung
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
NEXT: ...	// Sprungmarke "NEXT"

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

LOOP: Schleifensprung

Beschreibung

Mit der Anweisung "Schleifensprung" programmieren Sie Programmschleifen.

Die Anweisung interpretiert das rechte Wort des Akkumulators 1 als eine vorzeichenlose 16-Bit Ganzzahl im Bereich von 0 bis 65535. In der Ausführung dekrementiert die Anweisung zuerst den Inhalt des Akkumulators 1 um eins. Wenn der Wert im Akkumulator 1 nach dem Dekrementieren nicht Null ist, wird der Sprung zur angegebenen Sprungmarke ausgeführt. Wenn der Wert Null ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der unmittelbar darauffolgenden Anweisung fortgesetzt. Der Wert im Akkumulator 1 gibt somit die Anzahl der zu durchlaufenden Programmschleifen vor. Diese Anzahl müssen Sie in einem Schleifenzähler speichern.

Der Operand <Anzahl> enthält die Anzahl der Schleifendurchläufe. Der Operand <Schleifenzähler> enthält noch auszuführende Schleifendurchläufe. Am Ende der Programmschleife wird der Inhalt des Schleifenzählers in den Akkumulator 1 geladen und von der Anweisung um eins dekrementiert. Der Sprung zur Sprungmarke am Anfang der Programmschleife wird ausgeführt, wenn der Wert im Akkumulator nach dem Dekrementieren ungleich Null ist.

Die Anweisung "Schleifensprung" beeinflusst keine Statusbits.

⚠ WARNUNG

Vorbelegung des Schleifenzählers

Wenn der Schleifenzähler nicht vorbelegt ist oder mit dem Wert "0" vorbelegt ist, wird der Schleifenzähler immer weiter ins Negative dekrementiert, wodurch eine endlose Schleife entsteht, die die CPU in den Betriebszustand STOP versetzen kann.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Anzahl der auszuführenden Schleifendurchläufe in das rechte Wort des Akkumulators 1 laden.
START: T "Tag_Counter"	// Anfang der Programmschleife
	// Inhalt des Akkumulators 1 in den Schleifenzähler transferieren.
L "MyTag_1"	// Wert des Operanden "MyTag_1" laden.
L "MyTag_2"	// Wert des Operanden "MyTag_2" laden.
*D	// Werte multiplizieren
T "MyTag_1"	// Ergebnis der Multiplikation in den Operanden "MyTag_1" transferieren.
L "Tag_Counter"	// Inhalt des Schleifenzählers in den Akkumulator 1 laden.
LOOP START	// Wert im Akkumulator 1 um eins dekrementieren.
	// Bei einem Wert ungleich Null zum Anfang der Programmschleife springen.
	// Bei einem Wert Null die nächstfolgende Anweisung bearbeiten.
L "MyTag_2"	// Wert des Operanden "MyTag_2" laden.
L 100	// Den Wert 100 laden.
>I	// Vergleichen, ob der Wert des Operanden "MyTag_2" größer als 200 ist.
= "MyTag_3"	// Vergleichergebnis in den Operanden "MyTag_3" schreiben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Datenbausteine

AUF: Datenbaustein im DB-Register aufschlagen

Beschreibung

Mit der Anweisung "Datenbaustein im DB-Register aufschlagen" schlagen Sie einen globalen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Die darauf folgenden DB-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Die Anweisung "Datenbaustein im DB-Register aufschlagen" wird unabhängig von Bedingungen ausgeführt und beeinflusst weder das Verknüpfungsergebnis noch die Akkumulatorinhalte.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datenbaustein im DB-Register aufschlagen":

Operand	Deklaration	Datenbausteintyp	Beschreibung
<Datenbau-stein>	-	DB	Datenbaustein, der geöffnet wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`AUF "GlobalDataBlock"`	`// Globalen Datenbaustein öffnen und die Nummer des Bausteins in das DB-Register übertragen.`
`L %DBW0`	`// Datenwort DBW0 des geöffneten globalen Datenbausteins in Akkumulator 1 laden.`
`T "MyTag"`	`// Inhalt des Akkumulators 1 in den Operanden "MyTag_1" transferieren.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

AUFDI: Datenbaustein im DI-Register aufschlagen

Beschreibung

Mit der Anweisung "Datenbaustein im DI-Register aufschlagen" schlagen Sie einen beliebigen Datenbaustein auf. Die Nummer des Datenbausteins wird in das DI-Register übertragen. Die darauf folgenden DI-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Bei der symbolischen Adressierung eines lokalen Formalparameters aus der Bausteinschnittstelle greifen Sie immer auf den Datenbaustein zu, dessen Nummer Sie beim Bausteinaufruf angegeben haben.

Die Anweisung "Datenbaustein im DI-Register aufschlagen" wird unabhängig von Bedingungen ausgeführt und beeinflusst weder das Verknüpfungsergebnis noch die Akkumulatorinhalte.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datenbaustein im DI-Register aufschlagen":

Operand	Deklaration	Datenbausteintyp	Beschreibung
<Datenbau-stein>	-	DI	Datenbaustein, der geöffnet wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUFDI "DataBlock"	// Datenbaustein öffnen und die Nummer des Datenbausteins in das DI-Register übertragen.
L %DIW0	// Datenwort DIW0 des geöffneten Datenbausteins in Akkumulator 1 laden.
T "MyTag"	// Inhalt des Akkumulators 1 in den Operanden "MyTag" transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

TDB: Datenbausteinregister tauschen

Beschreibung

Mit der Anweisung "Datenbausteinregister tauschen" tauschen Sie die Inhalte der Datenbausteinregister. Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst die Statusbits nicht.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUF "GlobalDataBlock"	// Globalen Datenbaustein öffnen und die Nummer des Datenbausteins in das DB-Register übertragen.
AUFDI "DataBlock"	// Datenbaustein öffnen und die Nummer des Datenbausteins in das DI-Register übertragen.
TDB	// Datenbausteinregister tauschen
	// Das DB-Register verweist auf "DataBlock" und das DI-Register verweist auf "GlobalDataBlock".
L %DIW0	// DW0 aus "GlobalDataBlock" in Akkumulator 1 laden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

L DBLG: Länge eines Global-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Länge eines Global-Datenbausteins in Akkumulator 1 laden" laden Sie die Länge eines über das Datenbausteinregister geöffneten Global-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Global-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen. Die Länge des Datenbausteins ist gleichbedeutend mit der Anzahl der Datenbytes.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUF "GlobalDataBlock"	// Globalen Datenbaustein öffnen und die Nummer des Datenbausteins in das DB-Register übertragen.
L DBLG	// Länge des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
<D	// Vergleichen, ob die Länge des Datenbausteins kleiner als der Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis in den Operanden "MyTag_2" schreiben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

L DBNO: Nummer eines Global-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Nummer eines Global-Datenbausteins in Akkumulator 1 laden" laden Sie die Nummer eines über das Datenbausteinregister geöffneten Global-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Global-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUF "GlobalDataBlock"	// Globalen Datenbaustein öffnen und die Nummer des Datenbausteins in das DB-Register übertragen.
L DBNO	// Nummer des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
==I	// Vergleichen, ob die Nummer des Datenbausteins gleich dem Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis in den Operanden "MyTag_2" schreiben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

L DILG: Länge eines Instanz-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Länge eines Instanz-Datenbausteins in Akkumulator 1 laden" laden Sie die Länge eines über das Datenbausteinregister geöffneten Instanz-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Instanz-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen. Die Länge des Datenbausteins ist gleichbedeutend mit der Anzahl der Datenbytes.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUFDI "InstanceDataBlock"	// Datenbaustein öffnen und die Nummer des Datenbausteins in das DI-Register übertragen.

AWL	Erläuterung
L DILG	// Länge des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
<I	// Vergleichen, ob die Länge des Datenbausteins kleiner als der Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis in den Operanden "MyTag_2" schreiben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

L DINO: Nummer eines Instanz-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Nummer eines Instanz-Datenbausteins in Akkumulator 1 laden" laden Sie die Nummer eines über das Datenbausteinregister geöffneten Instanz-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Instanz-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUFDI "InstanceDataBlock"	// Datenbaustein öffnen und die Nummer des Datenbausteins in das DI-Register übertragen.
L DINO	// Nummer des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
==I	// Vergleichen, ob die Nummer des Datenbausteins gleich dem Wert des Operanden "MyTag_1" ist.

AWL	Erläuterung
= "MyTag_2"	// Abfrageergebnis dem Operanden "MyTag_2" zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Codebausteine

BE: Bausteinende

Beschreibung

Mit der Anweisung "Bausteinende" beenden Sie die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird. Die Programmbearbeitung wird dann mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Die Anweisung "Bausteinende" wird unabhängig von Bedingungen ausgeführt. Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
SIN	// Sinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.
BE	// Baustein beenden

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

BEB: Bedingtes Bausteinende

Beschreibung

Mit der Anweisung "Bedingtes Bausteinende" beenden Sie, abhängig vom Verknüpfungsergebnis (VKE), die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird.

Wenn das VKE "1" ist, wird die Anweisung ausgeführt. Der aktuell bearbeitete Baustein wird beendet und die Programmbearbeitung im aufrufenden Baustein fortgesetzt. Die Programmbearbeitung wird mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Wenn das VKE bei der Bearbeitung der Anweisung "0" ist, wird die Anweisung nicht ausgeführt. Die CPU setzt in diesem Fall das VKE auf "1" und bearbeitet die nächstfolgende Anweisung.

Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U "Tag_Input"	// Operanden "Tag_Input" auf "1" abfragen und mit dem VKE nach UND verknüpfen.
BEB	// Bei erfüllter Bedingung (VKE = "1") Baustein beenden.
	// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

BEA: Absolutes Bausteinende

Beschreibung

Mit der Anweisung "Absolutes Bausteinende" beenden Sie die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird. Die Programmbearbeitung wird dann mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Die Anweisung "Absolutes Bausteinende" wird unabhängig von Bedingungen ausgeführt und kann mehrfach innerhalb eines Bausteins programmiert werden. Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U "Tag_Input"	// Operanden "Tag_Input" auf "1" abfragen und mit dem VKE nach UND verknüpfen.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1") Programmbearbeitung bei der Sprungmarke "NEXT" fortsetzen.
	// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.
BEA	// Baustein beenden
NEXT: T "Tag_Result"	// Sprungmarke "NEXT"
	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

CALL: Baustein aufrufen

Beschreibung

Mit der Anweisung "Baustein aufrufen" rufen Sie die folgenden Bausteinarten im Programm auf:

- Funktionen
- Funktionsbausteine

Die Anweisung "Baustein aufrufen" wird unabhängig von Bedingungen ausgeführt. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt.

Einen aufgerufenen Baustein können Sie mit Daten versorgen. Die Datenübergabe erfolgt über die Bausteinparameter. Die Parameter des aufgerufenen Bausteins werden nach der Aufrufanweisung im aufrufenden Baustein aufgelistet. Diesen können Sie die benötigten Aktualparameter zuordnen. Bei der Ausführung der Anweisung "Baustein aufrufen" werden

die Daten an den aufgerufenen Baustein übergeben. Durch die Datenübergabe werden die Inhalte des Statusworts sowie der Adress- und Datenbaustein-Register verändert.

Die nicht versorgten Parameter eines Funktionsbausteins behalten ihren aktuellen Wert bei. Bei Aufruf von Funktionen müssen alle Parameter versorgt werden. Falls ein aufgerufener Baustein keine Parameter hat, wird keine Parameterliste angezeigt.

Wenn der aufgerufene Baustein einen Instanz-Datenbaustein benötigt, geben Sie ihn durch ein Komma getrennt beim Aufruf an. Der angegebene Datenbaustein muss vor dem Aufruf angelegt sein.

Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Hinweis

Die Anweisung "Baustein aufrufen" kann nur dann verwendet werden, wenn die Bausteineigenschaft "Parameterversorgung über Register" nicht aktiviert ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL "MyFunction"	// Funktion "MyFunction" aufrufen
Input_1 := "Tag_Input_1",	// Aktualparameter zuordnen
Input_2 := "Tag_Input_2"	
Output_1 := "Tag_Output_1"	
Output_2 := "Tag_Output_1"	
CALL "MyFunctionBlock", "MyFB_DB"	// Funktionsbaustein "MyFunctionBlock" aufrufen
Value_1 := "Tag_Value_1"	// Aktualparameter zuordnen
Value_2 := "Tag_Value_2"	
Output := "Tag_Output"	
CALL "LIMIT"	// Anweisung "Limitieren" aufrufen
MN := "Tag_LowLimit"	// Aktualparameter zuordnen
IN := "Tag_InputValue"	
MX := "Tag_HighLimit"	
OUT := "Tag_Output"	
CALL "CTU", "CTU_DB"	// Zähler "Vorwärts zählen" aufrufen
CU := "Tag_StartCTU"	// Aktualparameter zuordnen
R := "Tag_ResetCounter"	
PV := "Tag_PresetValue"	
Q := "Tag_CounterStatus"	
CV := "Tag_CounterValue"	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

CC: Baustein bedingt aufrufen

Beschreibung

Mit der Anweisung "Baustein bedingt aufrufen" rufen Sie abhängig vom Verknüpfungsergebnis (VKE) Funktionen (FC) und Funktionsbausteine (FB) auf, die keine Parameter und keine Instanz-Datenbausteine haben.

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) vor der Bearbeitung der Anweisung "1" ist. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt. Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Beim Wechseln des Bausteins wird das Statusbit OS auf "0" zurückgesetzt. Die Statusbits A0, A1 und OV werden entsprechend des aufrufenden Bausteins beeinflusst.

Hinweis

Die Anweisung "Baustein bedingt aufrufen" kann nur dann verwendet werden, wenn die Bausteineigenschaft "Parameterversorgung über Register" aktiviert ist. Dadurch sinkt aber die Performance.

Der Inhalt der Akkumulatoren und der Adressregister wird durch die Anweisung "Baustein bedingt aufrufen" nicht verändert.

Wenn das aktuelle VKE "0" ist, wird die Anweisung und damit auch der Bausteinaufruf nicht ausgeführt und das VKE auf "1" gesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input_1"`	`// Operanden "Tag_Input_1" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`

AWL	Erläuterung
CC "MyFunction"	// Bei erfüllter Bedingung (VKE = "1") die Funktion "MyFunction" aufrufen. // Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
CC "MyFunctionBlock"	// Bei erfüllter Bedingung (VKE = "1") den Funktionsbaustein "MyFunctionBlock" aufrufen. // Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Übersicht über die Bausteineigenschaften (Seite 7215)

Parameterversorgung über Register beim Bausteinaufruf in AWL (Seite 7559)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

UC: Baustein unbedingt aufrufen

Beschreibung

Mit der Anweisung "Baustein unbedingt aufrufen" rufen Sie Funktionen (FC) und Funktionsbausteine (FB) auf, die keine Parameter und keine Instanz-Datenbausteine haben.

Die Anweisung wird unabhängig von Bedingungen ausgeführt. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt. Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Beim Wechseln des Bausteins wird das Statusbit OS auf "0" zurückgesetzt. Die Statusbits A0, A1 und OV werden entsprechend des aufrufenden Bausteins beeinflusst.

Hinweis

Die Anweisung "Baustein unbedingt aufrufen" kann nur dann verwendet werden, wenn die Bausteineigenschaft "Parameterversorgung über Register" aktiviert ist. Dadurch sinkt aber die Performance.

Der Inhalt der Akkumulatoren und der Adressregister wird durch die Anweisung "Baustein unbedingt aufrufen" nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Operanden "Tag_Input_1" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
UC "MyFunction"	// Funktion "MyFunction" aufrufen.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
UC "MyFunctionBlock"	// Funktionsbaustein "MyFunctionBlock" aufrufen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Übersicht über die Bausteineigenschaften (Seite 7215)

Parameterversorgung über Register beim Bausteinaufruf in AWL (Seite 7559)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Wortverknüpfungen

UW: Wortweise nach UND verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach UND verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach UND. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn beide zu verknüpfenden Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach UND gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	0	0	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach UND verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach UND verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
UW W#16#F6B5	// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach UND verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
UW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach UND verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	0101	0000	0011	0001
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	0100	1000	0000	0010

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

OW: Wortweise nach ODER verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach ODER verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach ODER. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
OW W#16#F6B5	// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
OW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1111	1111	1011	1111
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1111	1110	1011	1011

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Programmierbeispiele AWL (Seite 7595)

XOW: Wortweise nach EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach EXKLUSIV ODER. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn der Signalzustand der beiden zu verknüpfenden Bits unterschiedlich ist. Wenn beide zu verknüpfende Bits den gleichen Signalzustand führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach EXKLUSIV ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	0

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach EXKLUSIV ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach EXKLUSIV ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
XOW W#16#F6B5	// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
XOW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1010	1111	1000	1110
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1011	0110	1011	1001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

UD: Doppelwortweise nach UND verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach UND verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach UND. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn beide zu verknüpfenden Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach UND gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	0	0	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach UND verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach UND verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in den Akkumulator 1 laden.
UD DW#16#39C657AC            // Wert des Akkumulators 1 mit der Konstante
                             (DW#16#39C657AC) nach UND verknüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"             // Ergebnis in den Operanden "Tag_Result_1" übertra-
                             gen.
L "Tag_Value_2"              // Wert des Operanden in den Akkumulator 1 laden.
L "Tag_Value_3"              // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                             schieben.
                             // Wert des Operanden "Tag_Value_3" in den Akkumulator
                             1 laden.
UD                           // Wert des Akkumulators 2 ("Tag_Value_2") mit dem
                             Wert des Akkumulators 1 ("Tag_Value_3")nach UND ver-
                             knüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_2"             // Ergebnis in den Operanden "Tag_Result_2" übertra-
                             gen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39 C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_1	0001	1001	0100	0100	0001	0101	0010	1000
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011

Operand	Wert							
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_2	0010	0001	0100	0010	0101	0101	0010	1010

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

OD: Doppelwortweise nach ODER verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach ODER verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach ODER. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in den Akkumulator 1 laden.
OD DW#16#39C657AC            // Wert des Akkumulators 1 mit der Konstante
                             (DW#16#39C657AC) nach ODER verknüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"             // Ergebnis in den Operanden "Tag_Result_1" übertra-
                             gen.
L "Tag_Value_2"              // Wert des Operanden in den Akkumulator 1 laden.
L "Tag_Value_3"              // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                             schieben.
                             // Wert des Operanden "Tag_Value_3" in den Akkumulator
                             1 laden.
OD                           // Wert des Akkumulators 2 ("Tag_Value_2") mit dem
                             Wert des Akkumulators 1 ("Tag_Value_3")nach ODER ver-
                             knüpfen.
                             // Ergebnis im Akkumulator 1 speichern
T "Tag_Result_2"             // Ergebnis in den Operanden "Tag_Result_2" übertra-
                             gen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39 C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_1	0111	1111	1110	0110	1101	1111	1011	1111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_2	0111	1101	1100	0111	0101	1111	1110	1111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

XOD: Doppelwortweise nach EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach EXKLUSIV ODER verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach EXKLUSIV ODER. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn der Signalzustand der beiden zu verknüpfenden Bits unterschiedlich ist. Wenn beide zu verknüpfende Bits den gleichen Signalzustand führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach EXKLUSIV ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	0

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach EXKLUSIV ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach EXKLUSIV ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in den Akkumulator 1 laden.
XOD DW#16#39C657AC	// Wert des Akkumulators 1 mit der Konstante (DW#16#39C657AC) nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in den Akkumulator 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in den Akkumulator 1 laden.
XOD	// Wert des Akkumulators 2 ("Tag_Value_2") mit dem Wert des Akkumulators 1 ("Tag_Value_3")nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39 C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0110	0110	1010	0010	1100	1010	1001	0111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0101	1100	1000	0101	0000	1010	1000	0111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Setzen der Statusbits bei Wortverknüpfungen (Seite 201)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Schieben und Rotieren

Schieben

SSI: Mit Vorzeichen wortweise schieben

Beschreibung

Mit der Anweisung "Mit Vorzeichen wortweise schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit dem Signalzustand des Bits 15 (Vorzeichenbit bei INT-Zahlen) aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SSI <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SSI: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit dem Signalzustand des Bits 15 aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SSI 6                        // Bits 0 bis 15 des Akkumulators 1 um 6 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit dem Zustand des Bits
                             15 auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 3                          // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Schiebezahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
SSI                          // Bits 0 bis 15 des Akkumulators 1 um 3 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit dem Zustand des Bits
                             15 auffüllen.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1111	1110	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	1011	1010	0101

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

SSD: Mit Vorzeichen doppelwortweise schieben

Beschreibung

Mit der Anweisung "Mit Vorzeichen doppelwortweise schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit dem Signalzustand des Bits 31 (Vorzeichenbit bei DINT-Zahlen) aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SSD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SSD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit dem Signalzustand des Bits 31 aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Mit Vorzeichen doppelwortweise schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT,	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
```

AWL	Erläuterung
SSD 7	// Bits 0 bis 31 des Akkumulators 1 um 7 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit dem Zustand des Bits 31 auffüllen.
T "Tag_Result_1"	// Inhalt des Akkumulators 1 in den Operanden transferieren.
L 4	// Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SSD	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit dem Zustand des Bits 31 auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	1000	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	1111	0001	1110	1100	1000	1011	1010
Tag_Value_2	0010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	0010	1000	1010	0010	1001	1011	1100

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

SLW: Wortweise nach links schieben

Beschreibung

Mit der Anweisung "Wortweise nach links schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach links. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SLW <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SLW: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach links schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SLW 5                        // Bits 0 bis 15 des Akkumulators 1 um 5 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 4                          // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Schiebezahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
SLW                          // Bits 0 bis 15 des Akkumulators 1 um 4 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1010	0111	0110	0000
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	1101	0010	1011	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

SRW: Wortweise nach rechts schieben

Beschreibung

Mit der Anweisung "Wortweise nach rechts schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SRW <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SRW: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach rechts schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SRW 6                        // Bits 0 bis 15 des Akkumulators 1 um 6 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 4                          // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Schiebezahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
SRW                          // Bits 0 bis 15 des Akkumulators 1 um 4 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	0000	0001	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	0101	1101	0010

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

SLD: Doppelwortweise nach links schieben

Beschreibung

Mit der Anweisung "Doppelwortweise nach links schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach links. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SLD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SLD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach links schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SLD 5                        // Bits 0 bis 31 des Akkumulators 1 um 5 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

AWL	Erläuterung
L 4	// Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SLD	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach links schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1110	1100	1000	1011	1010	0111	0110	0000
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	1000	1010	0010	1001	1011	1100	1101	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

SRD: Doppelwortweise nach rechts schieben

Beschreibung

Mit der Anweisung "Doppelwortweise nach rechts schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SRD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SRD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach rechts schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in Akkumulator 1 laden.
SRD 7	// Bits 0 bis 31 des Akkumulators 1 um 7 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"	// Inhalt des Akkumulators 1 in den Operanden transferieren.
L 4	// Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SRD	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Re-sult_1	0000	0000	1011	1110	1100	1000	1011	1010
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Re-sult_2	0000	1010	1000	1010	0010	1001	1011	1100

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

Rotieren

RLD: Doppelwortweise nach links rotieren

Beschreibung

Mit der Anweisung "Doppelwortweise nach links rotieren" rotieren Sie den gesamten Inhalt des Akkumulators 1 bitweise nach links. Bei der Ausführung schiebt die Anweisung die Bits 0 bis 31 des Akkumulators 1 bitweise nach links und füllt die beim Schieben frei werdenden Stellen mit den hinausgeschobenen Bitstellen auf. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die rotiert wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- RLD <Anzahl>: Die Rotierzahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 32. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- RLD: Die Rotierzahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 32 wird die Rotierzahl mit einer Modulo-Division durch 32 berechnet. Wenn z. B. die Rotierzahl 34 ist, werden die Bits des Akkumulators 1 um 2 Bitstellen rotiert.

Wenn die angegebene Rotierzahl 32 ist, bleibt der Inhalt des Akkumulators 1 unverändert.

Wenn die angegebene Rotierzahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach links rotieren":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die rotiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
RLD 4                        // Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach
                             links rotieren.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 6                          // Rotierzahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Rotierzahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
RLD                          // Bits 0 bis 31 des Akkumulators 1 um 6 Stellen nach
                             links rotieren.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	**0101**	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	0110	0100	0101	1101	0011	1011	**0101**
Tag_Value_2	**1010**	**10**00	1010	0010	1001	1011	1100	1101
Tag_Result_2	0010	1000	1010	0110	1111	0011	01**10**	**1010**

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

RRD: Doppelwortweise nach rechts rotieren

Beschreibung

Mit der Anweisung "Doppelwortweise nach rechts rotieren" rotieren Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Bei der Ausführung schiebt die Anweisung die Bits 0 bis 31 des Akkumulators 1 bitweise nach rechts und füllt die beim Schieben frei werdenden Stellen mit den hinausgeschobenen Bitstellen auf. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die rotiert wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- RRD <Anzahl>: Die Rotierzahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 32. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- RRD: Die Rotierzahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 32 wird die Rotierzahl mit einer Modulo-Division durch 32 berechnet. Wenn z. B. die Rotierzahl 34 ist, werden die Bits des Akkumulators 1 um 2 Bitstellen rotiert.

Wenn die angegebene Rotierzahl 32 ist, bleibt der Inhalt des Akkumulators 1 unverändert.

Wenn die angegebene Rotierzahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach rechts rotieren":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die rotiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in Akkumulator 1 laden.
RRD 4	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach rechts rotieren.
T "Tag_Result_1"	// Inhalt des Akkumulators 1 in den Operanden transferieren.
L 6	// Rotierzahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Rotierzahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
RRD	// Bits 0 bis 31 des Akkumulators 1 um 6 Stellen nach rechts rotieren.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	**1011**
Tag_Result_1	**1011**	0101	1111	0110	0100	0101	1101	0011
Tag_Value_2	1010	1000	1010	0010	1001	1011	11**00**	**1101**
Tag_Result_2	**0011**	**01**10	1010	0010	1000	1010	0110	1111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

RLDA: Durch Statusbit A1 nach links rotieren

Beschreibung

Mit der Anweisung "Durch Statusbit A1 nach links rotieren" schieben Sie den Inhalt des Akkumulators 1 um 1 Bit nach links. Die beim Schieben freiwerdende Bitstelle (0) wird mit dem Signalzustand des Statusbits A1 aufgefüllt. Das Statusbit A1 erhält den Signalzustand des hinausgeschobenen Bits (31).

Die Anweisung wird unabhängig vom VKE ausgeführt. Sie setzt das Statusbit A0 auf "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden in Akkumulator 1 laden.
RLDA	// Bits 0 bis 31 des Akkumulators 1 um 1 Stelle nach links rotieren.
	// Bit 0 mit dem Signalzustand des Statusbits A1 auffüllen.
	// Hinausgeschobenes Bit in das Statusbit A1 schreiben.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	A1	Wert							
Tag_Value	**1**	**0**101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	**0**	1011	1110	1100	1000	1011	1010	0111	011**1**

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

RRDA: Durch Statusbit A1 nach rechts rotieren

Beschreibung

Mit der Anweisung "Durch Statusbit A1 nach rechts rotieren" schieben Sie den Inhalt des Akkumulators 1 um 1 Bit nach rechts. Die beim Schieben freiwerdende Bitstelle (31) wird mit dem Signalzustand des Statusbits A1 aufgefüllt. Das Statusbit A1 erhält den Signalzustand des hinausgeschobenen Bits (0).

Die Anweisung wird unabhängig vom VKE ausgeführt. Sie setzt die Statusbits A0 und OV auf "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden in Akkumulator 1 laden.

AWL	Erläuterung

RRDA	// Bits 0 bis 31 des Akkumulators 1 um 1 Stelle nach rechts rotieren.
	// Bit 31 mit dem Signalzustand des Statusbits A1 auffüllen.
	// Hinausgeschobenes Bit in das Statusbit A1 schreiben.

T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	A1	Wert							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1010	1111	1011	0010	0010	1110	1001	1101

Das Statusbit A1 hat den Wert "1", da das Bit 0 des Operanden "Tag_Value" auch den Wert "1" hat.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Grundlagen zu AWL (Seite 7557)

Weitere Anweisungen

Akkumulator

TAK: Inhalte der Akkumulatoren 1 und 2 tauschen

Beschreibung

Mit der Anweisung "Inhalte der Akkumulatoren 1 und 2 tauschen" tauschen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in den Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1")zur Sprungmarke "NEXT" springen.
	// Bei nicht erfüllter Bedingung (VKE = "0")die nächstfolgende Anweisung bearbeiten.
TAK	// Inhalte der Akkumulatoren 1 und 2 tauschen.
NEXT: -I	// Sprungmarke "NEXT"
	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

PUSH: Inhalt in den nächst höheren Akkumulator schieben

Beschreibung

Mit der Anweisung "Inhalt in den nächst höheren Akkumulator schieben" schieben Sie die Inhalte des Akkumulators 1 in den Akkumulator 2.

Der Inhalt des Akkumulators 1 wird durch die Anweisung nicht beeinflusst und bleibt nach der Ausführung der Anweisung unverändert. Der Inhalt des Akkumulators 2 geht verloren.

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor und nach der Ausführung der Anweisung:

Status	Akkumulator	
	1	2
Vor der Ausführung	Wert A	Wert B
Nach der Ausführung	Wert A	Wert A

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in Akkumulator 1 laden.
PUSH	// Inhalt der Akkumulatoren 1 bis 2 jeweils in den nächst höheren Akkumulator schieben.
L "Tag_Value_2"	// Wert des Operanden in Akkumulator 1 laden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

POP: Inhalt in den nächst niedrigeren Akkumulator schieben

Beschreibung

Mit der Anweisung "Inhalt in den nächst niedrigeren Akkumulator schieben" schieben Sie den Inhalt des Akkumulators 2 in den Akkumulator 1.

Der Inhalt des Akkumulators 2 wird durch die Anweisung nicht beeinflusst und bleibt nach der Ausführung der Anweisung unverändert. Der Inhalt des Akkumulators 1 geht verloren.

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor und nach der Ausführung der Anweisung:

Status	Akkumulator	
	1	2
Vor der Ausführung	Wert A	Wert B
Nach der Ausführung	Wert B	Wert B

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
T "Tag_Value_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Value_1" übertragen.

AWL	Erläuterung
POP	// Inhalt des Akkumulators 2 in den darunter liegenden Akkumulator 1 schieben.
T "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Value_2" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Adressregister

+AR1: Akkumulator 1 zu AR1 addieren

Beschreibung

Mit der Anweisung "Akkumulator 1 zu AR1 addieren" addieren Sie einen Wert zum Inhalt des Adressregisters 1. Die Art des im Adressregister 1 stehenden Zeigers und der Operandenbereich bleiben dabei erhalten.

Zur Angabe des zu addierenden Werts gibt es die folgenden Möglichkeiten:

- Angabe über eine Konstante: Die Anweisung addiert den Wert der Konstanten zum Adressregister 1. Der Wert der Konstante muss dem Format eines bereichsinternen Zeigers (POINTER) entsprechen.
- Angabe über den Wert des rechten Worts im Akkumulator 1: Die Anweisung interpretiert den Wert des rechten Worts im Akkumulator 1 als eine 16-Bit Ganzzahl und erweitert diese vorzeichenrichtig auf 24 Bit. Den Wert des Akkumulators 1 addiert die Anweisung anschließend zum Adressregister 1. Zulässig sind Werte von -32 768 bis +32 767.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Akkumulator 1 zu AR1 addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	POINTER	Wert, der zum Adressregister 1 addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
+AR1 P#10.0	// Zeiger zum Adressregister 1 addieren.
L "Tag_Value"	// Wert des Operanden "Tag_Value" in das rechte Wort des Akkumulators 1 laden.
+AR1	// Wert des Akkumulators 1 zum Adressregister 1 addieren.
TAR1 %MD24	// Inhalt des Adressregisters 1 in das Doppelwort MD24 übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

+AR2: Akkumulator 1 zu AR2 addieren

Beschreibung

Mit der Anweisung "Akkumulator 1 zu AR2 addieren" addieren Sie einen Wert zum Inhalt des Adressregisters 2. Die Art des im Adressregister 2 stehenden Zeigers und der Operandenbereich bleiben dabei erhalten.

Zur Angabe des zu addierenden Werts, gibt es die folgenden Möglichkeiten:

- Angabe über eine Konstante: Die Anweisung addiert den Wert der Konstanten zum Adressregister 2. Der Wert der Konstante muss dem Format eines bereichsinternen Zeigers (POINTER) entsprechen.
- Angabe über den Wert des rechten Worts im Akkumulator 1: Die Anweisung interpretiert den Wert des rechten Worts im Akkumulator 1 als eine 16-Bit Ganzzahl und erweitert diese vorzeichenrichtig auf 24 Bit. Den Wert des Akkumulators 1 addiert die Anweisung anschließend zum Adressregister 2. Zulässig sind Werte von -32 768 bis +32 767.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Hinweis

Das Adressregister AR2 wird bei der Bearbeitung von Multiinstanzen verwendet. Wenn Sie die Anweisung "Akkumulator 1 zu AR2 addieren" programmieren, müssen Sie zuvor den Inhalt des Adressregisters 2 "retten" und später wieder laden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Akkumulator 1 zu AR2 addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	POINTER	Wert, der zum Adressregister 1 addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`+AR2 P#10.0`	`// Zeiger zum Adressregister 2 addieren.`
`L "Tag_Value"`	`// Wert des Operanden "Tag_Value" in das rechte Wort des Akkumulators 1 laden.`
`+AR2`	`// Wert des Akkumulators 1 zum Adressregister 2 addieren.`
`TAR2 %MD24`	`// Inhalt des Adressregisters 1 in das Doppelwort MD24 übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

Nullanweisungen

BLD: Bildaufbau (Nullanweisung)

Beschreibung

Mit der Anweisung "Bildaufbau (Nullanweisung)" führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Die Anweisung dient zur Wiedererkennung von Codesequencen bei einer Parameterübergabe oder bei KOP/FUP-Netzwerken. Sie wird automatisch erzeugt, wenn ein KOP- oder FUP-Programm in AWL angezeigt wird. Der Parameterwert ist die Kennnummer der Anweisung und wird vom Programmiergerät erzeugt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bildaufbau (Nullanweisung)":

Parameter	Datentyp	Beschreibung
<Kennnum-mer>	WORD	Kennnummer der Anweisung

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

NOP 0: Nullanweisung

Beschreibung

Mit der Anweisung "Nullanweisung" mit einem Parameter 0 führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Der Anweisungscode enthält ein Bitmuster mit 16 Nullen. Die Anweisung ist nur für das Programmiergerät wichtig, wenn ein Programm angezeigt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in den Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1")zur Sprungmarke "NEXT" springen.
	// Bei nicht erfüllter Bedingung (VKE = "0")die nächstfolgende Anweisung bearbeiten.
TAK	// Inhalte der Akkumulatoren 1 und 2 tauschen.
NEXT: NOP 0	// Sprungmarke "NEXT"

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

NOP 1: Nullanweisung

Beschreibung

Mit der Anweisung "Nullanweisung" mit einem Parameter 1 führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Der Anweisungscode enthält ein Bitmuster mit 16 Einsen. Die Anweisung ist nur für das Programmiergerät wichtig, wenn ein Programm angezeigt wird.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Abfragen und Setzen der Statusbits in AWL (Seite 197)

4.1.3.4 SCL

Bitverknüpfungen

R_TRIG: Positive Signalflanke erkennen

Beschreibung

Mit der Anweisung "Positive Signalflanke erkennen" können Sie eine Zustandsänderung von "0" auf "1" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "0" auf "1" erkennt, wird am Ausgang Q eine positive Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Eingehendes Signal, dessen Flanke abgefragt wird
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"R_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

In der Variablen "R_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn an den Operanden "TagIn_1" und "TagIn_2" oder im Operanden "TagIn_3" eine Änderung im Signalzustand von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut_Q" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

F_TRIG: Negative Signalflanke erkennen

Beschreibung

Mit der Anweisung "Negative Signalflanke erkennen" können Sie eine Zustandsänderung von "1" auf "0" am Eingang CLK erkennen. Die Anweisung vergleicht den aktuellen Wert am Eingang CLK mit dem Zustand der vorherigen Abfrage (Flankenmerker), die in der angegebenen Instanz gespeichert ist. Wenn die Anweisung einen Zustandswechsel am Eingang CLK von "1" auf "0" erkennt, wird am Ausgang Q eine negative Signalflanke erzeugt, d. h. der Ausgang führt genau einen Zyklus lang den Wert TRUE oder "1".

In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Eingehendes Signal, dessen Flanke abgefragt wird
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"F_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

In der Variablen "F_TRIG_DB" wird der vorherige Zustand der Variablen am Eingang CLK gespeichert. Wenn am Operand "TagIn" eine Änderung im Signalzustand von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut" einen Zyklus lang den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Zeiten

Aufruf von IEC-Zeiten

Beschreibung

Sie haben die Möglichkeit, die IEC-Zeit als Einzel- oder Multiinstanz zu deklarieren und im Programmcode aufzurufen.

Die folgenden Syntaxmöglichkeiten für die Deklaration einer IEC-Zeit als Multiinstanz innerhalb einer Struktur in der Bausteinschnittstelle stehen Ihnen zur Verfügung:

IEC-Zeit als ARRAY-Element

Deklaration in der Bausteinschnittstelle:

Static	
MyARRAY	Array[0..1] of IEC_TIMER
MyARRAY[0]	IEC_TIMER
MyARRAY[1]	IEC_TIMER

Programmcode:

SCL

```
#MyARRAY[1].TOF(IN := <Operand>, PT := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

Static	
MyARRAY	Array[0..1] of TON_TIME
MyARRAY[0]	TON_TIME
MyARRAY[1]	TON_TIME

Programmcode:

SCL

```
#MyARRAY[#index](IN := <Operand>, PT := <Operand>)
```

IEC-Zeit innerhalb einer anonymen Struktur

Deklaration in der Bausteinschnittstelle:

Static	
MyStruct	Struct
FirstTime	IEC_TIMER
PT	Time
ET	Time
IN	Bool
Q	Bool

Programmcode:

SCL

```
#MyStruct.FirstTime.TOF(IN := <Operand>, PT := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

	Static	
	MyStruct	Struct
	FirstTime	TON_TIME
	PT	Time
	ET	Time
	IN	Bool
	Q	Bool

Programmcode:

SCL

```
#MyStruct.FirstTime(IN := <Operand>, PT := <Operand>)
```

IEC-Zeit im globalen Datenbaustein

Deklaration im Datenbaustein:

MyGlobalDB

	Name	Datentyp
1	Static	
2	Timer	"MyUDT"
3	FirstTime	IEC_TIMER
4	PT	Time
5	ET	Time
6	IN	Bool
7	Q	Bool

Programmcode:

SCL

```
"MyGlobalDB".Timer.FirstTime.TOF(IN := <Operand>, PT := <Operand>)
```

Deklaration im Datenbaustein:

MyGlobalDB

	Name	Datentyp
1	Static	
2	Timer	"MyUDT"
3	FirstTime	IEC_TIMER
4	SecondTime	TON_TIME
5	PT	Time
6	ET	Time
7	IN	Bool
8	Q	Bool

Programmcode:

SCL
```
"MyGlobalDB".Timer.SecondTime(IN := <Operand>, PT := <Operand>)
```

IEC-Zeit als Element in der Bausteinschnittstelle

Deklaration in der Bausteinschnittstelle:

Static	
Timer	"MyUDT"
FirstTime	IEC_TIMER
PT	Time
ET	Time
IN	Bool
Q	Bool

Programmcode:

SCL
```
#Timer.FirstTime.TOF(IN := <Operand>, PT := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

Static	
Timer	"MyUDT"
FirstTime	IEC_TIMER
SecondTime	TON_TIME
PT	Time
ET	Time
IN	Bool
Q	Bool

Programmcode:

SCL
```
#Timer.SecondTime(IN := <Operand>, PT := <Operand>)
```

IEC-Zeit innerhalb eines ARRAY-DBs

Deklaration im ARRAY-DB:

MyARRAYDB

	Name	Datentyp
1	MyARRAYDB	Array[0..1] of "MyUDT"
2	MyARRAYDB[0]	"MyUDT"
3	FirstTime	IEC_TIMER
4	PT	Time
5	ET	Time
6	IN	Bool
7	Q	Bool

Programmcode:

SCL
```
"MyARRAYDB"."THIS"[0].FirstTime.TOF(IN := <Operand>, PT := <Operand>)
```

Deklaration im ARRAY-DB:

MyARRAYDB			
		Name	Datentyp
1		MyARRAYDB	Array[0..1] of "MyUDT"
2		MyARRAYDB[0]	"MyUDT"
3		FirstTime	IEC_TIMER
4		SecondTime	TON_TIME
5		PT	Time
6		ET	Time
7		IN	Bool
8		Q	Bool

Programmcode:

SCL
```
"MyARRAYDB"."THIS"[0].SecondTime(IN := <Operand>, PT := <Operand>)
```

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Parameter Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Parameter IN "0" ist, wird der Parameter ET zurückgesetzt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Informationen zum Aufruf von IEC-Zeiten innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zeiten (Seite 1728)

Für CPUs der Baureihe S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TP_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TP_TIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

Für CPUs der Baureihe S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_LTIMER_DB")
- Deklaration als lokale Variable vom Datentyp TP_TIME oder TP_LTIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_LTIMER_Instance)

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TP();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BOOL	BOOL	E, A, M, D, L, P	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L, P	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L, P	Operand, der für die Zeitdauer PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

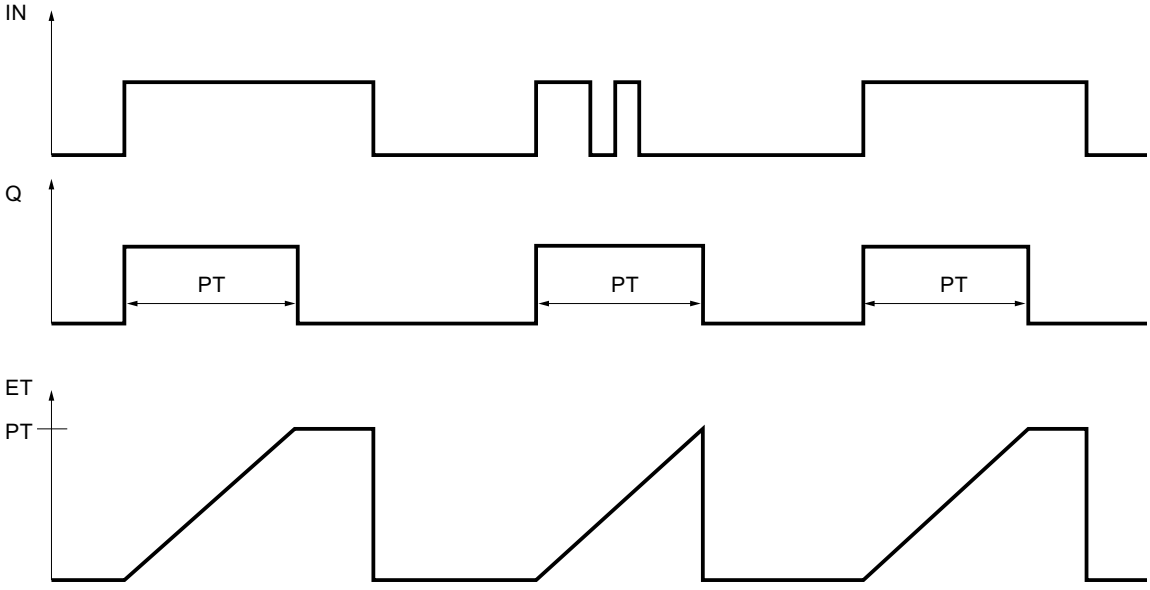

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"TP_DB".TP(IN := "Tag_Start",
           PT := "Tag_PresetTime",
           Q => "Tag_Status",
           ET => "Tag_ElapsedTime");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Parameters Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Parameter IN erfasst wird.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeit PT erreicht ist. Der Parameter ET wird zurückgesetzt, sobald der Signalzustand am Parameter IN auf "0" wechselt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Informationen zum Aufruf von IEC-Zeiten innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zeiten (Seite 1728)

Für CPUs der Baureihe S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TON_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TON_TIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

Für CPUs der Baureihe S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TON_TIME oder TON_LTIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TON();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BOOL	BOOL	E, A, M, D, L, P	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L, P	Zeitdauer der Einschaltverzögerung. Der Wert am Parameter PT muss positiv sein.

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
Q	Output	BOOL	BOOL	E, A, M, D, L, P	Operand, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

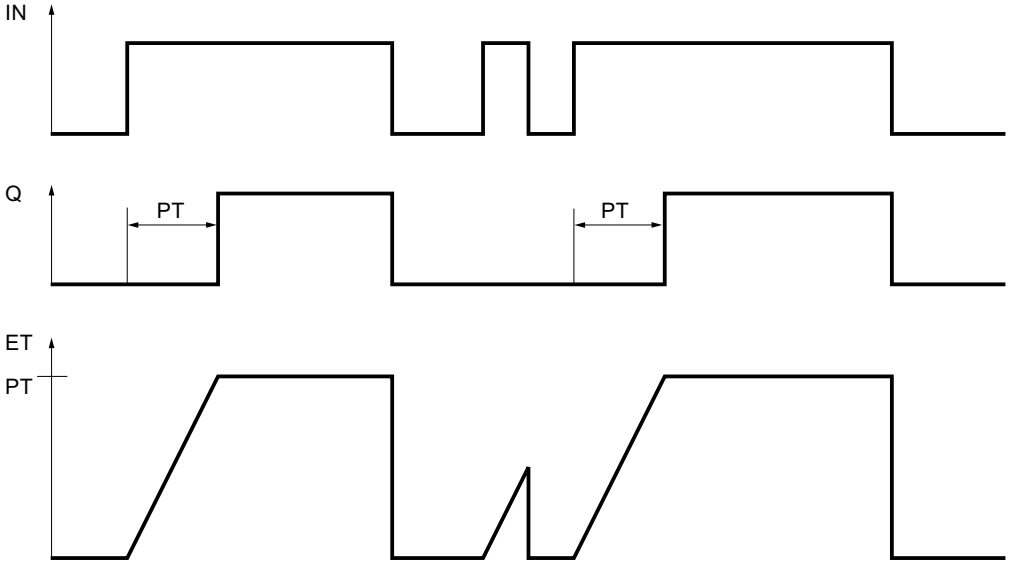

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt. Wenn der Signalzustand am Operand "Tag_Start" von "1" auf "0" wechselt, wird der Operand "Tag_Status" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Parameters Q um die programmierte Zeitdauer PT. Der Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeit PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET so lange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt. Wenn der Parameter IN vor dem Ablauf der Zeit PT auf "1" wechselt, wird der Parameter ET auf den Wert T#0s zurückgesetzt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Informationen zum Aufruf von IEC-Zeiten innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zeiten (Seite 1728)

Für CPUs der Baureihe S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TOF_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TOF_TIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

Für CPUs der Baureihe S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TOF_TIME oder TOF_LTIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TOF();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BOOL	BOOL	E, A, M, D, L, P	Starteingang
PT	Input	TIME	TIME, LTIME	E, A, M, D, L, P	Zeitdauer der Ausschaltverzögerung. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L, P	Operand, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

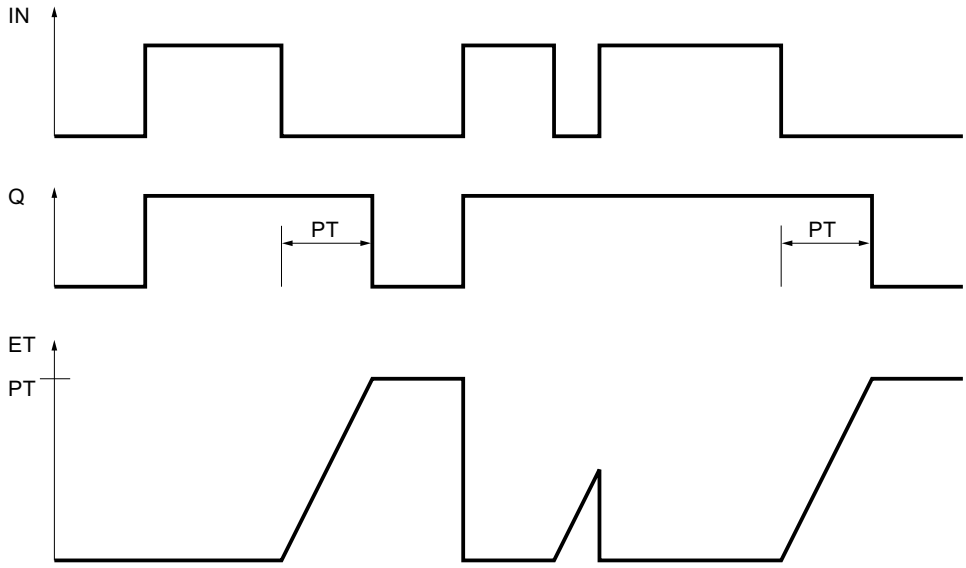

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TOF_DB".TOF(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Bei einem Wechsel im Signalzustand des Operanden "Tag_Start" von "0" auf "1" wird der Operand "Tag_Status" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Parameter IN auf "1" wechselt, wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Parameter IN erfasst werden. Die akkumulierte Zeit wird am Parameter ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN auf "0" wechselt.

Der Parameter R setzt die Parameter ET und Q unabhängig vom Signalzustand am Parameter IN zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Informationen zum Aufruf von IEC-Zeiten innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zeiten (Seite 1728)

Für CPUs der Baureihe S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER oder TONR_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TONR_TIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

Für CPUs der Baureihe S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TONR_TIME oder TONR_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER_DB")
- Deklaration als lokale Variable vom Datentyp TONR_TIME oder TONR_LTIME im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_TIMER_Instance)

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TONR();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BOOL	BOOL	E, A, M, D, L, P	Starteingang
R	Input	BOOL	BOOL	E, A, M, D, L, P	Rücksetzen der Parameter ET und Q
PT	Input	TIME	TIME, LTIME	E, A, M, D, L, P	Maximale Dauer der Zeiterfassung. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	BOOL	E, A, M, D, L, P	Operand, der nach dem Ablauf der Zeit PT gesetzt bleibt.
ET	Output	TIME	TIME, LTIME	E, A, M, D, L, P	Akkumulierte Zeit

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":

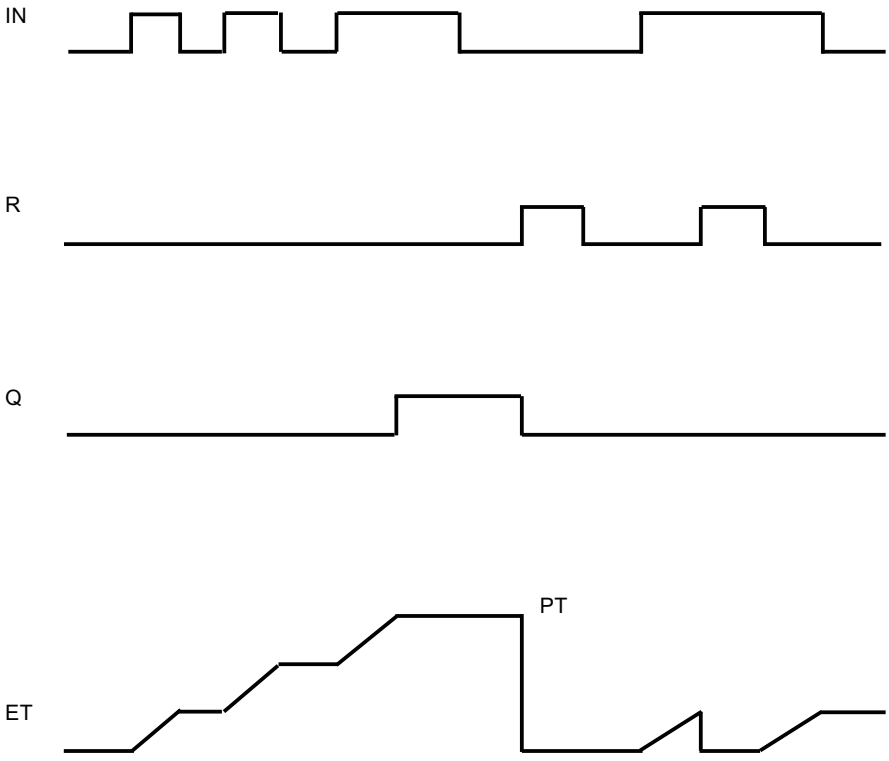

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TONR_DB".TONR(IN := "Tag_Start",
               R := "Tag_Reset",
               PT := "Tag_PresetTime",
               Q => "Tag_Status",
               ET => "Tag_Time");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Während des Ablaufs der Zeit werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" des Operanden "Tag_Start" erfasst werden. Die akkumulierte Zeit wird im Operanden "Tag_Time" gespeichert. Wenn der am Parameter PT angegebene Zeitwert erreicht ist, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_Time" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RESET_TIMER: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die Strukturkomponenten der Zeit im angegebenen Datenbaustein werden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Am Parameter TIMER wird der Anweisung "Zeit rücksetzen" eine im Programm deklarierte IEC-Zeit zugewiesen. Die Anweisung muss innerhalb einer IF-Anweisung programmiert werden. Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<IEC-Zeit>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, die zurückgesetzt wird

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF #started = false THEN
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
#started := true;
END_IF;
```

SCL

```
IF "TON_DB".ET < T#25s THEN
RESET_TIMER(TIMER := "TON_DB");
#started := false;
END_IF;
```

Wenn die Variable #started den Signalzustand "0" führt, dann wird die Anweisung "Einschaltverzögerung erzeugen" bei einer positiven Signalflanke am Operanden "Tag_Start" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PresetTime" vorgegeben ist. Der Operand "Tag_Status" wird gesetzt, wenn die durch den Operanden "Tag_PresetTime" vorgegebene Zeitdauer abgelaufen ist. Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag_Start" noch den Signalzustand "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.

Wenn die abgelaufene Zeit der IEC-Zeit "TON_DB" kleiner als 25s ist, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Instanz-Datenbaustein "TON_DB" abgelegte Zeit zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

PRESET_TIMER: Zeitdauer laden

Beschreibung

Mit der Anweisung "Zeitdauer laden" stellen Sie die Zeitdauer einer IEC-Zeit ein. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegeben IEC-Zeit.

Hinweis

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC_TIMER-Struktur.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Zeitdauer>	Input	TIME	TIME, LTIME	E, A, M, D, L	Zeitdauer, mit der die IEC-Zeit läuft.
<IEC-Zeit>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	IEC-Zeit, deren Dauer eingestellt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF #started = false THEN
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
#started := true;
#preset = true
END_IF;

IF "TON_DB".ET < T#10s AND #preset = true THEN
PRESET_TIMER(PT := T#25s,
             TIMER := "TON_DB");
#preset := false;
END_IF;
```

Wenn die Variable #started den Signalzustand "0" und am Operanden "Tag_Start" eine positive Signalflanke anliegt, dann wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag_PresetTime" vorgegeben ist. Der Operand "Tag_Status" wird gesetzt, wenn die durch den Operanden "Tag_PresetTime" vorgegebene

Zeitdauer PT abgelaufen ist. Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag_Start" noch den Signalzustand "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.

Wenn die abgelaufene Zeit der IEC-Zeit "TON_DB" kleiner als 10s ist und die Variable #preset den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer, die am Parameter PT angegeben ist, in den Instanz-Datenbaustein "TON_DB" und überschreibt dabei den Zeitwert des Operanden "Tag_PresetTime" innerhalb des Instanz-Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim nächsten Zugriff auf "TON_DB".Q oder "TON_DB".ET ändern.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Legacy

S_PULSE: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie die am Parameter T_NO programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten und der Parameter Q auf "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Der Parameter Q liefert den Signalzustand "1", solange die Zeit läuft und der Signalzustand am Parameter S "1" ist. Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, liefert der Parameter Q den Signalzustand "0". Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Parameter Q ebenfalls den Signalzustand "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

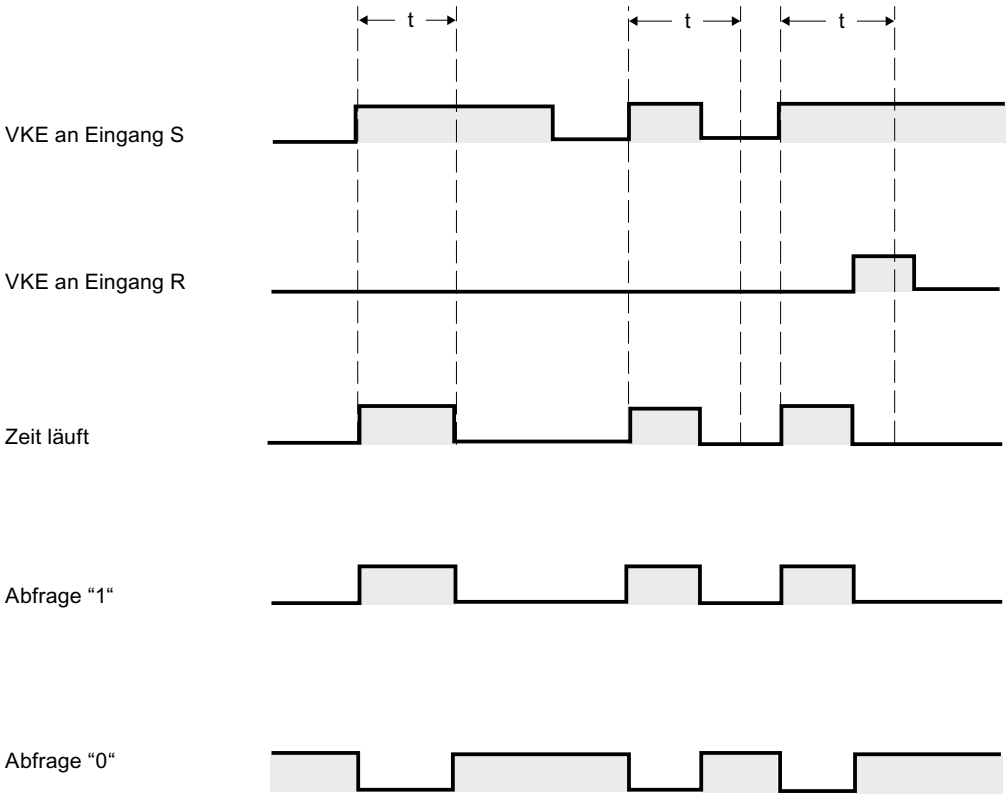

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_PULSE(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q => "Tag_Status",
                        BI => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "Tag_Number" ab, solange der Operand "Tag_1" den Signalzustand "1" liefert.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird der Operand "Tag_Status" auf "0" zurückgesetzt. Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Operand "Tag_Status" ebenfalls den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu SCL (Seite 7607)

S_PEXT: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt. Solange die Zeit läuft, liefert der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist, wird der Parameter Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der am Parameter TV programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Parameter R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Parameter R hat keine Auswirkungen.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

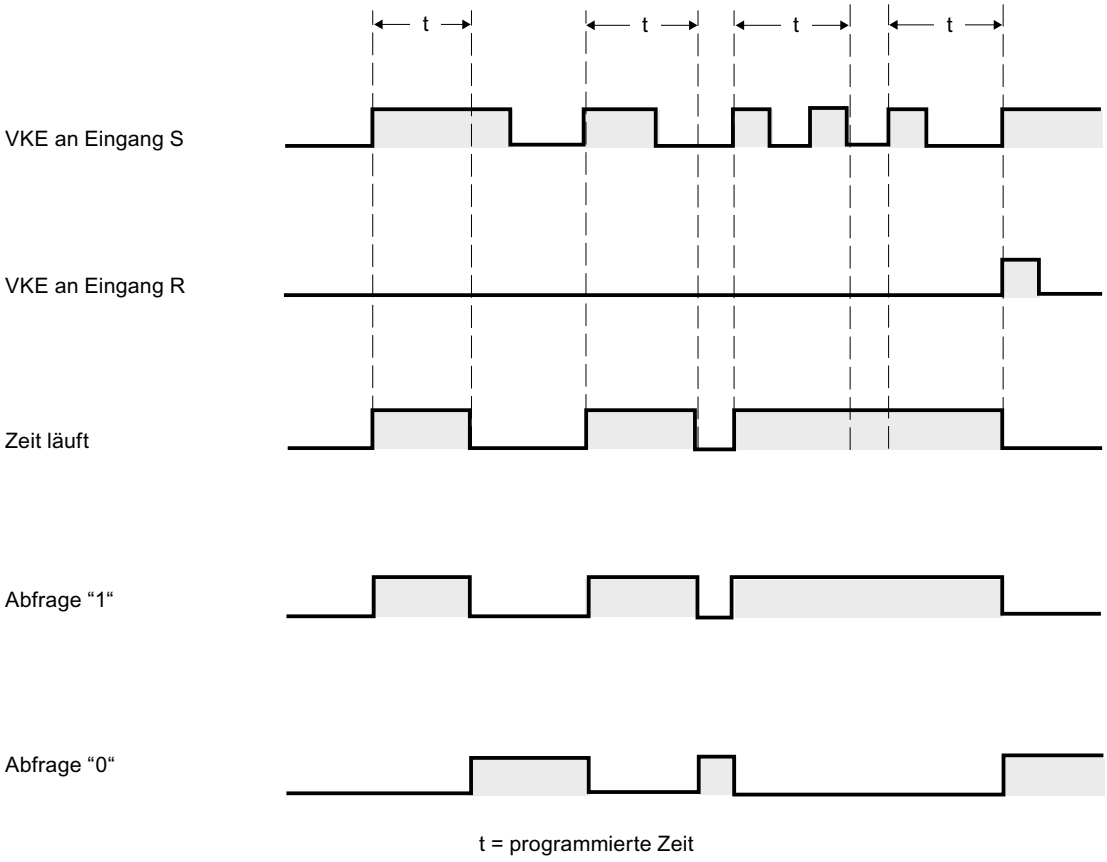

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_PEXT(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q => "Tag_Status",
                       BI => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Solange die Zeit läuft, liefert der Operand "Tag_Status" den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Operand "Tag_Status" auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu SCL (Seite 7607)

S_ODT: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und am Parameter S der Signalzustand "1" noch anliegt, liefert der Parameter Q den Signalzustand "1". Wechselt der Signalzustand am Parameter S von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Parameter Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Parameter R zurückgesetzt, auch wenn die Zeit nicht läuft und das Verknüpfungsergebnis (VKE) am Parameter S "1" ist.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

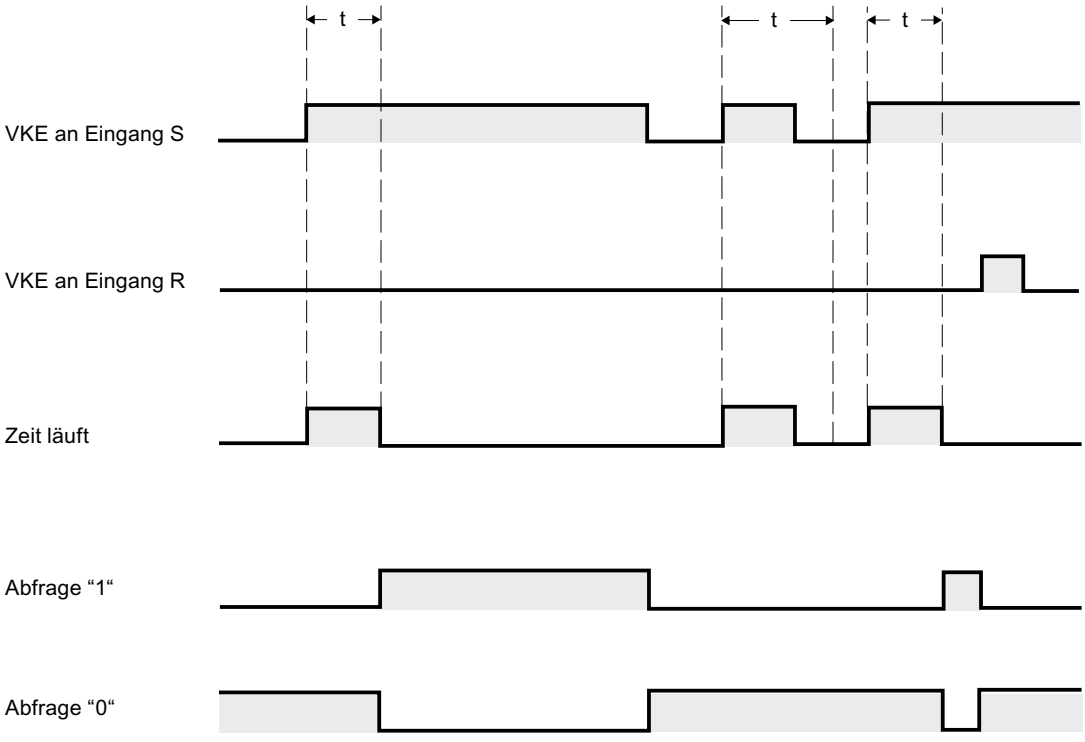

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := S_ODT(T_NO := "Timer_1",
                      S := "Tag_1",
                      TV := "Tag_Number",
                      R := "Tag_Reset",
                      Q => "Tag_Status",
                      BI => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab, solange der Signalzustand des Operanden "Tag_1" "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und der Operand "Tag_Status" den Signalzustand "1" hat, wird der Operand "Tag_Status" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "Tag_1" von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Operand "Tag_Status" liefert in diesem Fall den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu SCL (Seite 7607)

S_ODTS: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt.

Wenn die Zeit abgelaufen ist, liefert der Parameter Q den Signalzustand "1" unabhängig vom Signalzustand am Parameter S. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer TV neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Parameter S. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

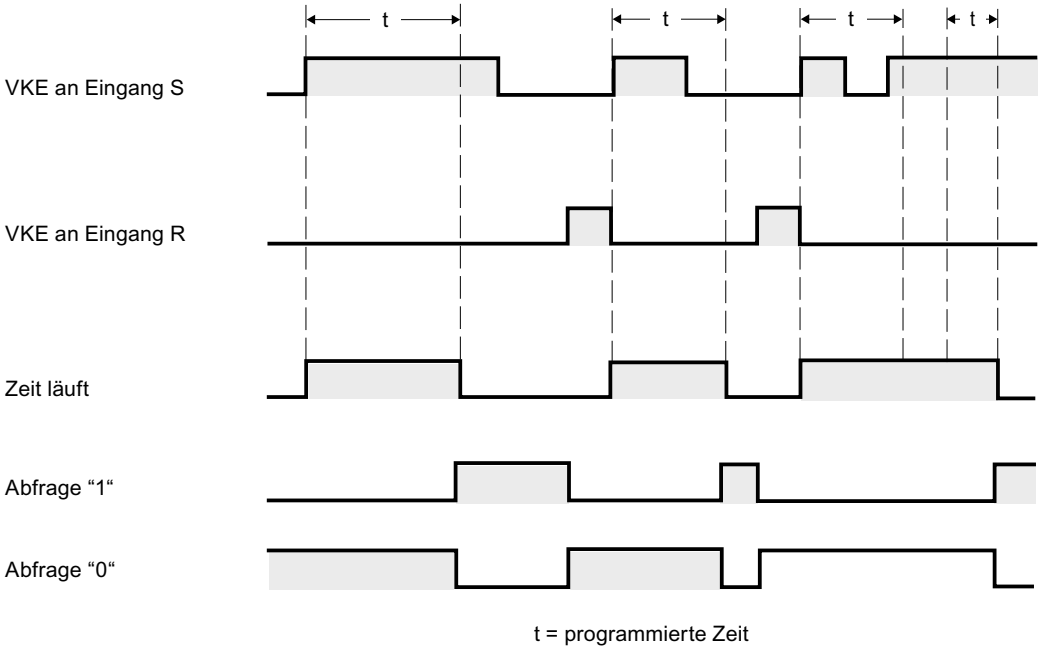

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_ODTS(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q => "Tag_Status",
                       BI => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab.

Wenn die Zeit abgelaufen ist, liefert der Operand "Tag_Status" den Signalzustand "1" unabhängig vom Signalzustand des Operanden "Tag_1". Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu SCL (Seite 7607)

S_OFFDT: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine negative Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab. Solange die Zeit läuft oder der Parameter S den Signalzustand "1" liefert, führt der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Parameter Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

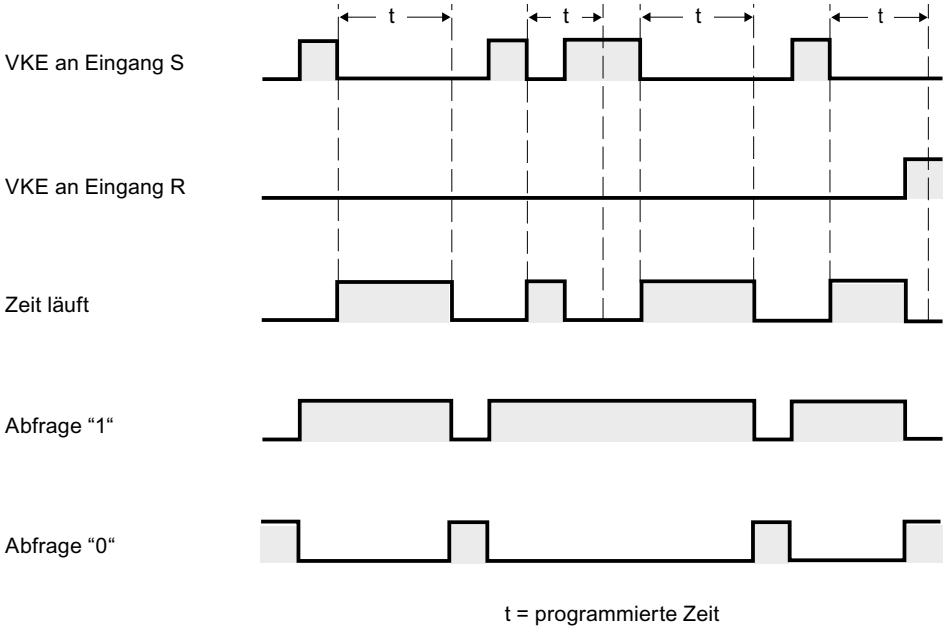

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_OFFDT(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q => "Tag_Status",
                        BI => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab. Solange die Zeit läuft

oder der Operand "Tag_1" den Signalzustand "1" liefert, führt der Operand "Tag_Status" den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "Tag_1" "0" ist, wird der Operand "Tag_Status" auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1" während die Zeit läuft, wird die Zeit zurückgesetzt. Erst wenn eine negative Flanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 1567)

Grundlagen zu SCL (Seite 7607)

Zähler

Aufruf von IEC-Zählern

Beschreibung

Sie haben die Möglichkeit, den IEC-Zähler als Einzel- oder Multiinstanz zu deklarieren und im Programmcode aufzurufen.

Die folgenden Syntaxmöglichkeiten für die Deklaration eines IEC-Zählers als Multiinstanz innerhalb einer Struktur in der Bausteinschnittstelle stehen Ihnen zur Verfügung:

IEC-Zähler als ARRAY-Element

Deklaration in der Bausteinschnittstelle:

Static	
MyARRAY	Array[0..1] of IEC_COUNTER
MyARRAY[0]	IEC_COUNTER
MyARRAY[1]	IEC_COUNTER

Programmcode:

SCL

```
#MyARRAY[1].CTU(CU := <Operand>, PV := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

Static	
MyARRAY	Array[0..1] of CTU_INT
MyARRAY[0]	CTU_INT
MyARRAY[1]	CTU_INT

Programmcode:

SCL
```
#MyARRAY[#index](CU := <Operand>, PV := <Operand>)
```

IEC-Zähler innerhalb einer anonymen Struktur

Deklaration in der Bausteinschnittstelle:

Static	
MyStruct	Struct
FirstTime	IEC_COUNTER
CU	Bool
CD	Bool
R	Bool
LD	Bool
QU	Bool
QD	Bool
PV	Int
CV	Int

Programmcode:

SCL
```
#MyStruct.FirstTime.CTU(CU := <Operand>, PV := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

Static	
MyStruct	Struct
FirstTime	CTU_INT
CU	Bool
CD	Bool
R	Bool
LD	Bool
QU	Bool
QD	Bool
PV	Int
CV	Int

Programmcode:

SCL
```
#MyStruct.FirstTime(CU := <Operand>, PV := <Operand>)
```

IEC-Zähler im globalen Datenbaustein

Deklaration im Datenbaustein:

MyGlobalDB		
	Name	Datentyp
1	Static	
2	Counter	"MyUDT"
3	FirstCount	IEC_COUNTER
4	CU	Bool
5	CD	Bool
6	R	Bool
7	LD	Bool
8	QU	Bool
9	QD	Bool
10	PV	Int
11	CV	Int

Programmcode:

SCL

```
"MyGlobalDB".Counter.FirstCount.CTU(CU := <Operand>, PV := <Operand>)
```

Deklaration im Datenbaustein:

MyGlobalDB		
	Name	Datentyp
1	Static	
2	Counter	"MyUDT"
3	FirstCount	IEC_COUNTER
4	SecondCount	CTU_INT
5	CU	Bool
6	CD	Bool
7	R	Bool
8	LD	Bool
9	QU	Bool
10	QD	Bool
11	PV	Int
12	CV	Int

Programmcode:

SCL

```
"MyGlobalDB".Counter.SecondCount(CU := <Operand>, PV := <Operand>)
```

IEC-Zähler als Element in der Bausteinschnittstelle

Deklaration in der Bausteinschnittstelle:

Static	
Counter	"MyUDT"
FirstCount	IEC_COUNTER
CU	Bool
CD	Bool
R	Bool
LD	Bool
QU	Bool
QD	Bool
PV	Int
CV	Int

Programmcode:

SCL

```
#Counter.FirstCount.CTU(CU := <Operand>, PV := <Operand>)
```

Deklaration in der Bausteinschnittstelle:

Static	
Counter	"MyUDT"
FirstCount	IEC_COUNTER
SecondCount	CTU_INT
CU	Bool
CD	Bool
R	Bool
LD	Bool
QU	Bool
QD	Bool
PV	Int
CV	Int

Programmcode:

SCL

```
#Counter.SecondCount(CU := <Operand>, PV := <Operand>)
```

IEC-Zähler innerhalb eines ARRAY-DBs

Deklaration im ARRAY-DB:

MyARRAYDB		
	Name	Datentyp
1	MyARRAYDB	Array[0..1] of "MyUDT"
2	MyARRAYDB[0]	"MyUDT"
3	FirstCount	IEC_COUNTER
4	CU	Bool
5	CD	Bool
6	R	Bool
7	LD	Bool
8	QU	Bool
9	QD	Bool
10	PV	Int
11	CV	Int

Programmcode:

SCL

```
"MyARRAYDB"."THIS"[0].FirstCount.CTU(CU := <Operand>, PV := <Operand>)
```

Deklaration im ARRAY-DB:

MyARRAYDB		
	Name	Datentyp
1	MyARRAYDB	Array[0..1] of "MyUDT"
2	MyARRAYDB[0]	"MyUDT"
3	FirstCount	IEC_COUNTER
4	SecondCount	CTU_INT
5	CU	Bool
6	CD	Bool
7	R	Bool
8	LD	Bool
9	QU	Bool
10	QD	Bool
11	PV	Int
12	CV	Int

Programmcode:

SCL

```
"MyARRAYDB"."THIS"[0].SecondCount(CU := <Operand>, PV := <Operand>)
```

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPUs der Baureihe S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT

Für CPUs der Baureihe S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER_DB")
- Deklaration als lokale Variable vom Datentyp CTU_<Datentyp> im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_COUNTER_Instance)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTU();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Informationen zum Aufruf von IEC-Zählern innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zählern (Seite 1759)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P	Wert, bei dem der Ausgang Q gesetzt wird
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"IEC_COUNTER_DB".CTU(CU := "Tag_Start",
                     R := "Tag_Reset",
                     PV := "Tag_PresetValue",
                     Q => "Tag_Status",
                     CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CounterValue" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des angegebenen Datentyps (INT = 32767) erreicht ist.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag_CounterValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Grundlagen zu SCL (Seite 7607)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange am Parameter LD der

Signalzustand "1" ansteht, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPUs der Baureihe S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT

Für CPUs der Baureihe S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER_DB")
- Deklaration als lokale Variable vom Datentyp CTD_<Datentyp> im Abschnitt "Static" eines Programmbausteins (z. B. #MyIEC_COUNTER_Instance)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTD();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Informationen zum Aufruf von IEC-Zählern innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zählern (Seite 1759)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"IEC_SCOUNTER_DB".CTD(CD := "Tag_Start",
                      LD := "Tag_Load",
                      PV := "Tag_PresetValue",
                      Q => "Tag_Status",
                      CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert des Operanden "Tag_CounterValue" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (-128) erreicht ist.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag_CounterValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Grundlagen zu SCL (Seite 7607)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Parametern CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPUs der Baureihe S7-1200

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT

Für CPUs der Baureihe S7-1500

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)	Lokale Variable
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Instanz-Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER_DB")
- Deklaration als lokale Variable vom Datentyp CTUD_<Datentyp> im Abschnitt "Static" eines Programmbausteins (z. B. #MyCTUD_COUNTER_Instance)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert.

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTUD();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Informationen zum Aufruf von IEC-Zählern innerhalb von Strukturen (Multiinstanz) finden Sie hier: Aufruf von IEC-Zählern (Seite 1759)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"IEC_COUNTER_DB".CTUD(CU := "Tag_Start1",
                      CD := "Tag_Start2",
                      LD := "Tag_Load",
                      R := "Tag_Reset",
                      PV := "Tag_PresetValue",
                      QU => "Tag_CU_Status",
                      QD => "Tag_CD_Status",
                      CV => "Tag_CounterValue");
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_Start1" vorliegt, wird der aktuelle Zählwert um eins erhöht und im Operanden "Tag_CounterValue" abgelegt. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_Start2" vorliegt, wird der Zählwert um eins verringert und ebenfalls im Operanden "Tag_CounterValue" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Parameter CU so lange erhöht, bis er den oberen Grenzwert des angegebenen Datentyps (INT) erreicht. Bei einer positiven Signalflanke am Parameter CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert des angegebenen Datentyps (INT) erreicht.

Der Operand "Tag_CU_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_CU_Status" den Signalzustand "0".

Der Operand "Tag_CD_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_CD_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Grundlagen zu SCL (Seite 7607)

Legacy

S_CU: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Eingang CU "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CU und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig.
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L	Aktueller Zählwert
Funktionswert		WORD	E, A, M, D, L	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CU(C_NO := "Counter_1",
                     CU := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Wenn der Signalzustand am Parameter "Tag_Start" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "Tag_Value" gespeichert.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

S_CD: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Parameter CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig.
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L	Aktueller Zählwert
Funktionswert		WORD	E, A, M, D, L	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CD := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag_Value" gespeichert.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

S_CUD: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Wenn in einem Programmzyklus an den Parametern CU und CD eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) an den Parametern CU und CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Parametern CU, CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig.
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L	Aktueller Zählwert (Hexadezimal)
Funktionswert		WORD	E, A, M, D, L	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CU := "Tag_CU",
                     CD := "Tag_CD",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_CU" erfasst wird und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_CD" erfasst wird und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag_Value" gespeichert.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Vergleicher

TypeOf: Datentyp einer VARIANT-Variable abfragen

Beschreibung

Mit der Anweisung "Datentyp einer VARIANT-Variable abfragen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie können den Datentyp der Variable, die Sie in der Bausteinschnittstelle deklariert haben, entweder mit dem Datentyp einer anderen Variable oder direkt mit einem Datentyp auf "Gleich" oder "Ungleich" vergleichen.

Der Vergleichsoperand kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Die Anweisung "Datentyp einer VARIANT-Variable abfragen" können Sie nur innerhalb einer IF- oder CASE-Anweisung verwenden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Abzufragender Operand

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt den Vergleich mit einer anderen Variable:

SCL

```
IF TypeOf(#TagVARIANT) = TypeOf("TagBYTE") THEN
...;
END_IF;
```

Das folgende Beispiel zeigt den Vergleich mit einem Datentyp:

SCL

```
IF TypeOf(#TagVARIANT) = BYTE THEN
...;
END_IF;
```

Wenn der Operand, auf den der VARIANT #TagVARIANT zeigt, den Datentyp BYTE hat, dann ist die Vergleichsbedingung erfüllt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiel für das Programmieren einer Warteschlange (FIFO) (Seite 351)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu SCL (Seite 7607)

TypeOfElements: Datentyp eines ARRAY-Elements einer VARIANT-Variable abfragen

Beschreibung

Mit der Anweisung "Datentyp eines ARRAY-Elements einer VARIANT-Variable abfragen" fragen Sie ab, welchen Datentyp eine Variable hat, auf die ein VARIANT zeigt. Sie vergleichen den Datentyp der Variable, die Sie in der Bausteinschnittstelle deklariert haben, mit dem Datentyp einer Variablen auf "Gleich" oder "Ungleich".

Der Operand muss den Datentyp VARIANT haben. Der Vergleichsoperand kann ein elementarer Datentyp oder ein PLC-Datentyp sein.

Wenn es sich bei dem Datentyp der VARIANT-Variable um ein ARRAY handelt, dann wird der Datentyp der ARRAY-Elemente verglichen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Abzufragender Operand

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF TypeOfElements("Tag_Variant") = TypeOF("GlobalDB".Product[1]) THEN
"Tag_Variant" := "GlobalDB".Product[1] * 3;
END_IF;
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
"GlobalDB".Product[1]	1.5
Tag_Variant	4.5

Wenn die Variable, auf die der VARIANT zeigt, und der Operand "GlobalDB".Product[1] den Datentyp REAL haben, dann wird der Operand "GlobalDB".Product[1] mit 3 multipliziert und das Ergebnis in den Operand "Tag_Variant" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiel für das Programmieren einer Warteschlange (FIFO) (Seite 351)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu SCL (Seite 7607)

IS_ARRAY: Auf ARRAY abfragen

Beschreibung

Mit der Anweisung "Auf ARRAY abfragen" fragen Sie ab, ob der VARIANT auf eine Variable vom Datentyp ARRAY zeigt.

Die Anweisung "Auf ARRAY abfragen" können Sie nur innerhalb einer IF-Anweisung verwenden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Auf ARRAY abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Operand, der auf ARRAY abgefragt wird
Funktionswert		UDINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF IS_ARRAY(#Tag_VARIANTToArray) THEN
"Tag_Result" := CountOfElements(#Tag_VARIANTToArray);
END_IF;
```

Wenn es sich bei der Variable, auf die der VARIANT zeigt, um ein ARRAY handelt, dann wird die Anzahl der ARRAY-Elemente ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiel für das Programmieren einer Warteschlange (FIFO) (Seite 351)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu SCL (Seite 7607)

TypeOfDB: Datentyp eines DB abfragen

Beschreibung

Mithilfe der Anweisung "Datentyp eines DB abfragen" fragen Sie ab, welchen Datentyp der Datenbaustein hat, den die Variable vom Datentyp DB_ANY adressiert. Sie können den Datentyp des durch die Variable <Operand> adressierten DB entweder mit dem Datentyp einer anderen Variable oder direkt mit einem Datentyp auf "Gleich" oder "Ungleich" vergleichen.

Die Variable muss den Datentyp DB_ANY haben. Der Vergleichsoperand kann z. B. ein PLC-Datentyp, ein Systemdatentyp, eine Achse oder ein FB sein.

Die Anweisung "Datentyp eines DB abfragen" können Sie nur innerhalb einer IF- oder CASE-Anweisung verwenden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	DB_ANY	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Abzufragender Operand

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF TypeOfDB(#InputDBAny) = TO_SpeedAxis THEN
"TagOut" := 1;
END_IF;
```

Der Ausgang "TagOut" wird gesetzt, wenn der Datentyp des Operanden #InputDBAny adressierten DB ist gleich dem Datentyp TO_SpeedAxis.

Der Ausgang "TagOut" wird nicht gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Nummer des Datenbausteins ist "0".
- Der Datenbaustein existiert nicht.
- Der Datenbaustein ist ein ARRAY-DB.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Datentyp DB_ANY verwenden (Seite 215)

Grundlagen zu SCL (Seite 7607)

Mathematische Funktionen

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag eines Eingangswertes und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Ausdruck>	Input	SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		SINT, INT, DINT, Gleitpunktzahlen	SINT, INT, DINT, LINT, Gleitpunktzahlen	E, A, M, D, L, P	Absolutwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

Der Absolutbetrag des Eingangswerts wird im Format des Eingangswerts als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	-2
Tag_Result1	2
Tag_Value1	4
Tag_Value2	-1
Tag_Result2	4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und geben den kleinsten Wert als Ergebnis aus.

An der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Das Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Ergebnis der Anweisung
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := MIN(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	12222
IN2	Tag_Value2	14444
IN3	Tag_Value3	13333
Funktionswert	Tag_Result	12222

Die Anweisung vergleicht die Werte der verfügbaren Eingänge und kopiert den kleinsten Wert ("Tag_Value1") in den Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und geben den größten Wert als Ergebnis aus.

An der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Das Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Ergebnis der Anweisung
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := MAX(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	12 222
IN2	Tag_Value2	14 444
IN3	Tag_Value3	13 333
Funktionswert	Tag_Result	14 444

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("Tag_Value2") in den Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert des Parameters IN auf die Werte der Parameter MN und MX. Der Wert des Parameters MN darf dabei nicht größer als der Wert des Parameters MX sein.

Wenn der Wert des Parameters IN die Bedingung MN <= IN <= MX erfüllt, wird er als Ergebnis der Anweisung ausgeben. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Wert des Parameters MN als Ergebnis ausgegeben. Bei einer Überschreitung der Obergrenze MX wird der Wert des Parameters MX als Ergebnis ausgegeben.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis der am Parmeter IN angegebene Wert und der Freigabeausgang ENO ist "0".

Die Ausführung der Anweisung setzt voraus, dass die Operanden aller Parameter vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Obergrenze
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE, DTL	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	E, A, M, D, L, P	Ergebnis der Anweisung
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := LIMIT(MN := "Tag_Minimum",
                      IN := "Tag_Value",
                      MX := "Tag_Maximum");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_Minimum	12 000
IN	Tag_Value	8 000
MX	Tag_Maximum	16 000
Funktionswert	Tag_Result	12 000

Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_Minimum" und "Tag_Maximum" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_Minimum" in den Operanden "Tag_Result" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den Eingangswert und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := SQR("Tag_Value");
"Tag_Result2" := SQR((SQR("Tag_Value1"))*"Tag_Value2");
```

Das Quadrat des Eingangswerts wird im Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	2.5
Tag_Result1	6.25
Tag_Value1	6.0
Tag_Value2	2.0
Tag_Result2	5184.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem Eingangswert die Quadratwurzel und speichern das Ergebnis im angegebenen Operanden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück. Wenn der Eingangswert "0" ist, ist das Ergebnis auch "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := SQRT("Tag_Value");
"Tag_Result2" := SQRT((SQR("Tag_Value1"))+"Tag_Value2");
```

Die Quadratwurzel des Eingangswerts wird im Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	4.0
Tag_Result1	2.0
Tag_Value1	3.0
Tag_Value2	16.0
Tag_Result2	5.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem Eingangswert den natürlichen Logarithmus zur Basis e (e = 2,718282). Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := LN("Tag_Value");
"Tag_Result2" := LN("Tag_Value1"+"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	2.5
Tag_Result1	0.916
Tag_Value1	1.5
Tag_Value2	3.2
Tag_Result2	1.548

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem Eingangswert und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := EXP("Tag_Value");
"Tag_Result2" := EXP("Tag_Value1"/"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	20.5
Tag_Result1	799 902 200
Tag_Value1	15.5
Tag_Value2	30.2
Tag_Result2	1.671

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SIN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+1.570796 (π/2)
Tag_Result	1.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := COS("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+1.570796 (π/2)
Tag_Result	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := TAN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+3.141593 (π)
Tag_Result	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus einem Sinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Sinuswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ASIN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	1.0
Tag_Result	+1.570796 (π/2)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus einem Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Cosinuswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ACOS("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	0
Tag_Result	+1.570796 (π/2)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus einem Tangenswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Tangenswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ATAN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	1.0
Tag_Result	+0.785398 (π/4)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu SCL (Seite 7607)

FRAC: Nachkommastellen ermitteln

Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" geben Sie die Nachkommastellen eines Wertes als Ergebnis aus. Beispielsweise liefert der Eingangswert 123.4567 als Ergebnis den Wert 0.4567.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

FRAC_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
_<Datentyp>		Gleitpunktzahlen Vorbelegung: REAL	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Nachkommastellen des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := FRAC("Tag_Value");
"Tag_Result2" := FRAC_LREAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	2.555	-1.4421
Tag_Result1	0.555	-0.4421

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu SCL (Seite 7607)

Verschieben

Deserialize: Deserialisieren

Beschreibung

Mit der Anweisung "Deserialisieren" wandeln Sie die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder um und füllen dessen kompletten Inhalt.

Der Speicherbereich, in dem die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> liegt, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Speicherbereiche zugelassen. Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Stellen Sie vor der Umwandlung sicher, dass genügend Speicherplatz vorhanden ist. Wenn der Speicherbereich mithilfe der Anweisung "Serialisieren" befüllt wurde und eventuell Füll-Bytes eingefügt wurden, dann werden diese bei der Umwandlung nicht berücksichtigt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Mithilfe der Anweisung können Sie mehrere sequenzielle Darstellungsformen von umgewandelten Daten schrittweise wieder in ihren ursprünglichen Zustand umwandeln.

Wenn Sie nur eine einzelne sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder umwandeln möchten, dann können Sie dazu auch direkt die Anweisung "TRCV: Daten über Kommunikationsverbindung empfangen" verwenden.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das BOOL eine Länge von 1 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Optimierter Speicherbereich

Um größere Strukturen zu deserialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich für die sequenzielle Darstellung auch mit optimiertem Zugriff deklarieren. Der Inhalt der sequenziellen Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich. Der Zugriff auf die Bytes des ARRAYs ist ausschließlich symbolisch möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_ARRAY	Input	VARIANT ARRAY[*] of BYTE [1]	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der Datenstrom gespeichert ist.
DEST_VARIABLE	InOut	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die deserialisiert werden soll.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen. Der Parameter POS wird nullbasiert berechnet.
Funktionswert		INT	E, A, M, D, L	Fehlerinformation
[1] Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 möglich				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_ARRAY und DEST_VARIABLE überschneiden sich.
8136	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8150	Der Datentyp VARIANT am Parameter SRC_ARRAY enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_ARRAY
8153	Am Parameter SRC_ARRAY steht nicht genügend Speicherplatz zur Verfügung.
8154	Ungültiger Datentyp am Parameter SRC_ARRAY
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_VARIABLE
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8136	Der Zugriff am Parameter SRC_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle eines FBs oder FCs
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle eines FBs oder FCs
Error	INT	
Label	STRING[4]	

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Target	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Client,
                           POS := #BufferPos);

#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := #Label,
                           POS := #BufferPos);

IF #Label = 'arti' THEN
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Article[#DeliverPos],
                           POS := #BufferPos);
ELSIF #Label = 'Bill' THEN
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Bill[#DeliverPos],
                           POS := #BufferPos);
;
ELSE
;
END_IF;
```

Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform der Kundendaten aus der Variable "Buffer" und schreibt sie in die Variable "Target". Der Operand #BufferPos speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen.

Die Anweisung "Deserialisieren" deserialisiert die sequenzielle Darstellungsform des Trennblatts, das nach den Kundendaten in der sequenziellen Darstellungsform abgelegt wurde, aus der Variable "Buffer" und schreibt die Zeichen in den Operanden #Label. Die Zeichen werden mit Hilfe von Vergleichsanweisungen auf "arti" und "Bill" verglichen. Ist der Vergleich auf "arti" = TRUE, dann handelt es sich Artikeldaten, die deserialisiert und in die Variable "Target" geschrieben werden. Ist der Vergleich auf "Bill" = TRUE, dann handelt es sich um Rechnungsdaten, die deserialisiert und in die Variable "Target" geschrieben werden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu SCL (Seite 7607)

Serialize: Serialisieren

Beschreibung

Mit der Anweisung "Serialisieren" wandeln Sie mehrere PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> in eine sequenzielle Darstellungsform um, ohne dass Teile ihrer Struktur verloren gehen.

Mithilfe der Anweisung können Sie mehrere strukturierte Daten aus Ihrem Programm in einem Puffer, der vorzugsweise in einem globalen Datenbaustein liegt, zwischenspeichern und an eine andere CPU senden. Der Speicherbereich, in dem die umgewandelten Daten abgelegt werden, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Daten zugelassen. Fülldaten des Datenbereichs der Quelle sind im Ziel-ARRAY undefiniert. Diese können sowohl Füll-Bytes bzw. Füll-Bits eines Datenbereichs (z. B. ARRAY, STRUCT oder PLC-Datentyp (UDT)) sowie die aktuell ungenutzten Zeichen eines Strings sein.

Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Strukturen, die gemäß der Regeln für standard Speicherbereiche größer als 64 KB sind, können nicht serialisiert werden, wenn der Operand am Parameter DEST_ARRAY in einem standard Speicherbereich liegt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Der Operand am Parameter POS enthält die Information, wie viele Bytes die umgewandelten Daten belegen.

Wenn Sie einen einzelnen PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> versenden möchten, dann können Sie direkt die Anweisung "TSEND: Daten über Kommunikationsverbindung senden" aufrufen.

Hinweis

Vergleich von Strukturen

Um Strukturen zu vergleichen ist es nicht notwendig, diese zuerst zu serialisieren. Verwenden Sie stattdessen einen Vergleichsausdruck.

Weitere Informationen finden Sie hier: Vergleichsausdrücke (Seite 7612)

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hängt die Länge eines BOOL davon ab, welcher Datentyp folgt. Das bedeutet, wenn z. B. ein BYTE folgt, hat das BOOL eine Länge von 1 Byte. Wenn z. B. ein WORD folgt, dann hat das BOOL eine Länge von 2 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Daher ist es empfehlenswert, dass der Quelldatenbereich für das Serialisieren mit den großen Datentypen beginnt und mit booleschen Elementen endet. Dadurch wird das Auffüllen mit Füll-Bits wesentllich reduziert.

Optimierter Speicherbereich

Um größere Strukturen zu serialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich auch mit optimiertem Zugriff deklarieren. Die sequenzielle Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_VARIABLE	Input	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die serialisiert werden soll.
DEST_ARRAY	InOut	VARIANT	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der erzeugte Datenstrom gespeichert wird.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Gesamtzahl der Bytes, die die umgewandelten Kundendaten belegt haben. Der Parameter POS wird nullbasiert berechnet.
Funktionswert		INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_VARIABLE und DEST_ARRAY überschneiden sich.
8150	Der Datentyp VARIANT am Parameter SRC_VARIABLE enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_VARIABLE
8236	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_ARRAY
8253	Die Variable am Parameter DEST_ARRAY bietet für den Inhalt der Variable am Parameter SRC_VARIABLE nicht genügend Speicherplatz. Durch den Eingangswert der Variable am Parameter POS wird der zur Verfügung stehende Speicherplatz reduziert. Der Eingangswert des Parameters POS bestimmt, wo innerhalb der Variable am Parameter DEST_ARRAY begonnen wird.
8254	Ungültiger Datentyp am Parameter DEST_ARRAY
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8236	Der Zugriff am Parameter DEST_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#Tag_RetVal := Serialize(SRC_VARIABLE := "Source".Client,
                         DEST_ARRAY := "Buffer".Field,
                         POS := #BufferPos);

#Label := STRING_TO_WSTRING('arti');

#Tag_RetVal := Serialize(SRC_VARIABLE := #Label,
                         DEST_ARRAY := "Buffer".Field,
                         POS := #BufferPos);

#Tag_RetVal := Serialize(SRC_VARIABLE := "Source".Article[#DeliverPos],
                         DEST_ARRAY := "Buffer".Field,
                         POS := #BufferPos);
```

Die Anweisung "Serialisieren" serialisiert die Kundendaten aus der Variable "Source" und schreibt sie als sequenzielle Darstellung in die Variable "Buffer". Der Index des nächsten unbeschriebenen Bytes des Operanden "Buffer".Field wird im Operanden #BufferPos gespeichert.

Um die sequenzielle Darstellungsform später wieder leichter deserialisieren zu können, wird jetzt eine Art Trennblatt eingefügt. Die Anweisung "Zeichenkette verschieben" kopiert die Zeichen "arti" in den Operanden #Label. Die Anweisung "Serialisieren" schreibt diese Zeichen nach den Kundendaten in die Variable "Buffer". Der Wert des Operanden #BufferPos wird entsprechend erhöht.

Die Anweisung "Serialisieren" serialisiert die Daten eines bestimmten Artikels, der zur Laufzeit berechnet wird, aus der Variable "Source" und schreibt sie als sequenzielle Darstellung nach den Zeichen "arti" in die Variable "Buffer".

Die folgende Tabelle zeigt die Deklaration der Operanden:

Operand	Datentyp	Deklaration
DeliverPos	INT	Im Abschnitt "Input" der Bausteinschnittstelle
BufferPos	DINT	Im Abschnitt "Temp" der Bausteinschnittstelle
Error	INT	Im Abschnitt "Temp" der Bausteinschnittstelle
Label	STRING[4]	Im Abschnitt "Temp" der Bausteinschnittstelle

Die folgende Tabelle zeigt die Deklaration der PLC-Datentypen:

Name der PLC-Datentypen	Name	Datentyp
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

Die folgende Tabelle zeigt die Deklaration der Datenbausteine:

Name der Datenbausteine	Name	Datentyp
Source	Client	"Client" (PLC-Datentyp)
	Article	Array[0..10] of "Article" (PLC-Datentyp)
Buffer	Field	Array[0..294] of BYTE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu SCL (Seite 7607)

MOVE_BLK: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des ersten Elements des Quellbereichs festgelegt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn die folgende Bedingung zutrifft:

- Es werden mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN [1)]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
MOVE_BLK(IN := #a_array[2],
         COUNT := "Tag_Count",
         OUT => #b_array[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MOVE_BLK_VARIANT: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können ein komplettes ARRAY oder Elemente eines ARRAYs in ein anderes ARRAY des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-ARRAY dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines ARRAYs umkopieren oder einzelne Elemente kopieren.

Die Anzahl der Elemente, die kopiert werden sollen, darf den selektierten Quellbereich bzw. Zielbereich nicht überschreiten.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss das ARRAY noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC_INDEX und DEST_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des ARRAYs.

Die Anweisung wird nicht ausgeführt, wenn mehr Daten kopiert werden, als zur Verfügung gestellt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input [2)]	VARIANT (der auf ein ARRAY oder ein einzelnes ARRAY-Element zeigt), ARRAY of <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Der Quellbereich, aus dem kopiert wird
COUNT	Input	UDINT	E, A, M, D, L	Anzahl der Elemente, die kopiert werden Geben Sie dem Parameter COUNT den Wert "1", wenn am Parameter SRC oder am Parameter DEST kein ARRAY angegeben ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_INDEX	Input	DINT	E, A, M, D, L	Definiert das erste Element, das kopiert wird: • Der Parameter SRC_INDEX wird nullbasiert berechnet. Wenn am Parameter SRC ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter SRC_INDEX das erste Element innerhalb des Quellbereichs an, aus dem kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter SRC kein ARRAY oder nur ein einzelnes Element eines ARRAYs angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert "0".
DEST_IN-DEX	Input	DINT	E, A, M, D, L	Definiert den Anfang des Ziel-speicherbereichs: • Der Parameter DEST_INDEX wird nullbasiert berechnet. Wenn am Parameter DEST ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter DEST_INDEX das erste Element innerhalb des Zielbereichs an, in das kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter DEST kein ARRAY angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST	Output [1]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zielbereich, in den die Inhalte des Quellbereichs kopiert werden
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformationen
1) Der Parameter DEST ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden. 2) Am Parameter SRC sind die Datentypen BOOL und ARRAY of BOOL nicht zulässig.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Die Datentypen stimmen nicht überein. Verwenden Sie statt eines ARRAY of Struct ein ARRAY of PLC-Datentyp(UDT).
8151	Der Zugriff auf den Parameter SRC ist nicht möglich.
8152	Der Operand am Parameter SRC ist nicht typisiert.
8153	Codeerzeugungsfehler am Parameter SRC
8154	Der Operand am Parameter SRC hat den Datentyp BOOL.
8281	Ungültiger Wert am Parameter COUNT
8382	Der Wert am Parameter SRC_INDEX ist kleiner NULL.
8383	Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8482	Der Wert am Parameter DEST_INDEX ist kleiner NULL.
8483	Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8534	Der Parameter DEST ist schreibgeschützt
8551	Der Zugriff auf den Parameter DEST ist nicht möglich.
8552	Der Operand am Parameter DEST ist nicht typisiert.
8553	Codeerzeugungsfehler am Parameter DEST
8554	Der Operand am Parameter DEST hat den Datentyp BOOL.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := MOVE_BLK_VARIANT(SRC := #SrcField,
                                 COUNT := "Tag_Count",
                                 SRC_INDEX := "Tag_Src_Index",
                                 DEST_INDEX := "Tag_Dest_Index",
                                 DEST => #DestField);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
SRC	Input	#SrcField	Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[0..10] of "MOVE_UDT"
COUNT	Input	Tag_Count	2
SRC_INDEX	Input	Tag_Src_Index	3
DEST_INDEX	Input	Tag_Dest_Index	3
DEST	InOut	#DestField	Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[10..20] of "MOVE_UDT"

2 Elemente werden aus dem Quellbereich, beginnend mit dem vierten Element des ARRAY[0..10] of MOVE_UDT, in den Zielbereich kopiert. Die Kopien werden im ARRAY[10..20] of MOVE_UDT ab dem vierten Element eingefügt.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK_VARIANT im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

VariantGet: Wert einer VARIANT-Variable auslesen (Seite 1867)

Grundlagen zu SCL (Seite 7607)

UMOVE_BLK: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des ersten Elements des Quellbereichs festgelegt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Wert am Ausgang OUT ist ungültig, wenn die folgende Bedingung zutrifft:

- Es werden mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
UMOVE_BLK(IN := #a_array[2],
          COUNT := "Tag_Count",
          OUT => #b_array[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Hinweis

Weitere Informationen zur Anweisung UMOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

FILL_BLK: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Adresse im Zielbereich, ab der befüllt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

SCL

```
FILL_BLK(IN := #FillValue,
         COUNT := "Tag_Count",
```

SCL

```
OUT => #TargetArea[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

SCL
```
FILL_BLK(IN := 10,
         COUNT := 2,
         OUT => "Data_block_1".MyStruct1.Member_2);
```

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

SCL
```
FILL_BLK(IN := 10,
         COUNT := 2,
        OUT => "Data_block_1".MyStruct2.SubArray[1].NestedStruct.Member_2);
```

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

UFILL_BLK: Bereich ununterbrechbar befüllen

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	E, A, M, D, L, P	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, DATE, CHAR, WCHAR	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, WCHAR, TOD, LTOD	D, L	Adresse im Zielbereich, ab der befüllt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

SCL

```
UFILL_BLK(IN := #FillValue,
          COUNT := "Tag_Count",
          OUT => #TargetArea[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Beispiele mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1		Datentyp
MyStruct1		STRUCT
	Member_1	INT
	Member_2	INT
	Member_3	INT
	Member_4	INT
MyStruct2		STRUCT

Data_block_1					Datentyp
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

SCL

```
UFILL_BLK(IN := 10,
          COUNT := 2,
          OUT => "Data_block_1".MyStruct1.Member_2);
```

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

SCL

```
UFILL_BLK(IN := 10,
          COUNT := 2,
          OUT => "Data_block_1".MyStruct2.SubArray[1].NestedStruct.Member_2);
```

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SCATTER: Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Bitfolge in einzelne Bits zerlegen" zerlegen Sie eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Bitfolge in einzelne Bits zerlegen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der anonyme STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. ein Statuswort zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der STRUCT oder der PLC-Datentyp nicht genügend BOOL-Elemente zur Verfügung stellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Bitfolge, die zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das bzw. den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationArray	Output	ARRAY[0..15] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SCATTER(IN := #SourceWord,
        OUT => #DestinationArray);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationArray	Der Operand "DestinationArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationArray zugewiesen.

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationUDT	Output	"myBits"

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SCATTER(IN := #SourceWord,
        OUT => #DestinationUDT);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationUDT	Der Operand "DestinationUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationUDT zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen" zerlegen Sie ein oder mehrere Elemente eines ARRAY of BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen. Am Parameter COUNT_IN geben Sie an, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Das Quell-ARRAY am Parameter IN darf mehr Elemente haben, als am Parameter COUNT_IN angegeben sind. Das ARRAY of BOOL, der anonyme STRUCT oder der PLC-Datentyp muss genügend Elemente haben, um die Bits der zerlegten Bitfolgen speichern zu können. Der Zielspeicherbereich darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter IN ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter OUT ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Das ARRAY of WORD wird in 48 einzelne Bits zerlegt.

Hinweis

Wenn die untere ARRAY-Grenze des Ziel-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. Statuswörter zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Quell-ARRAY hat weniger Elemente als am Parameter COUNT_IN angegeben sind.
- Der Index des Ziel-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of BOOL geschrieben.
- Das ARRAY[*] of BOOL, der STRUCT oder der PLC-Datentyp stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zerlegt und in das ARRAY of BOOL, den anonymen STRUCT oder den PLC-Datentyp geschrieben. Die restlichen Bitfolgen werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, das zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_IN	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das/den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[0..95] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SCATTER_BLK(IN := #SourceArrayWord[2],
            COUNT_IN := #CounterInput,
            OUT => #DestinationArrayBool[0]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[0]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 1. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen.

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[-2..93] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SCATTER_BLK(IN := #SourceArrayWord[2],
            COUNT_IN := #CounterInput,
            OUT => #DestinationArrayBool[14]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[14]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 16. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einer Bitfolge zusammen. Die Bitfolge wird in einer Variable vom Datentyp BYTE, WORD, DWORD oder LWORD gespeichert.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der anonyme STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn das ARRAY, der anonyme STRUCT oder der PLC-Datentyp (UDT) weniger oder mehr BOOL-Elemente hat als die Bitfolge vorgibt, dann werden sie nicht übertragen und das ENO liefert den Signalzustand "0". Das ENO ist ebenfalls "0", wenn weniger als die notwendige Anzahl an Bits vorhanden ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, dessen Bits zu einer Bitfolge zusammengefügt werden. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Zusammengefügte Bitfolge, gespeichert in einer Variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArray	Input	ARRAY[0..15] of BOOL
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
GATHER(IN := #SourceArray,
       OUT => #DestinationWord);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArray	Der Operand "SourceArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bit)

Die Bits des Operanden #SourceArray werden zu einem WORD zusammengefügt.

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceUDT	Input	"myBits"
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
GATHER(IN := #SourceUDT,
       OUT => #DestinationWord);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceUDT	Der Operand "SourceUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Die Bits des Operanden #SourceUDT werden zu einem WORD zusammengefügt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einem oder mehreren Elementen eines ARRAY of <Bitfolge> zusammen. Am Parameter COUNT_OUT geben Sie an, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Damit geben Sie auch implizit vor, wie viele Elemente des ARRAY of BOOL, des anonymen STRUCTs oder des PLC-Datentyps benötigt werden. Das Ziel-ARRAY am Parameter OUT darf mehr Elemente haben, als am Parameter COUNT_OUT angegeben sind. Das ARRAY of <Bitfolge> muss genügend Elemente haben, um die Bits, die zusammengefügt werden sollen, speichern zu können. Das Ziel-ARRAY darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter OUT ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter IN ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Aus dem ARRAY of BOOL werden 48 einzelne Bits zusammengefügt.

Hinweis

Wenn die untere ARRAY-Grenze des Quell-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1200 ab der Firmware Version >4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Index des Quell-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of <Bitfolge> geschrieben.
- Das ARRAY[*] of <Bitfolge> stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zusammengefügt und in das ARRAY of <Bitfolge> geschrieben. Die restlichen Bits werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY of BOOL, STRUCT oder PLC-Datentyp, dessen Bits zusammengefügt werden (Quell-ARRAY). Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_OUT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of <Bitfolge>	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, in das die Bits gespeichert werden (Ziel-ARRAY)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[0..95] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
GATHER_BLK(IN := #SourceArrayBool[0],
           COUNT_OUT := #CounterOutput,
           OUT => #DestinationArrayWord[2]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[0]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 1. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits in das 3. Wort, die zweiten 16 Bits in das 4. Wort und die dritten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[-2..93] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
GATHER_BLK(IN := #SourceArrayBool[14],
           COUNT_OUT := #CounterOutput,
           OUT => #DestinationArrayWord[2]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[14]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Da aber erst ab dem 16. Element begonnen wird, stehen nur 80 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Wenn der Operand #Enable am Freigabeeingang EN den Signalzustand "1" liefert, dann wird die Anweisung ausgeführt. Ab dem 16. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits des Quell-ARRAYs nicht berücksichtigt werden. Die zweiten 16. Bits werden in das 3. Wort, die dritten 16 Bits in das 4. Wort und die vierten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Die restlichen 64 Bits des Quell-ARRAYs werden ebenfalls nicht berücksichtigt. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, dann liefert der Operand #EnableOut am Freigabeausgang ENO den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

AssignmentAttempt: Zuweisung eines VARIANT auf eine Referenz versuchen

Beschreibung

Mit der Anweisung "AssignmentAttempt" versuchen Sie die Zuweisung einer VARIANT-Variable auf eine Referenzvariable. Der Datentyp einer Referenzvariable wird zum Zeitpunkt der Deklaration festgelegt, während der Datentyp einer VARIANT-Variable während der Laufzeit ermittelt wird. Eine implizite Datentypkonvertierung ist bei Referenzvariablen nicht zulässig. Um die beiden Datentypen einander zuzuweisen, nutzen Sie deshalb den Zuweisungsversuch.

Beim Zuweisungsversuch wird zur Laufzeit geprüft, ob die VARIANT-Variable vom richtigen Datentyp ist. Wenn dies der Fall ist, wird die Zuweisung ausgeführt. Nach einer erfolgreichen Ausführung steht eine gültige Referenz in der Zielvariablen, andernfalls eine NULL.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	VARIANT	• Bausteinschnittstelle eines FC: Input, Output, InOut, Temp • Bausteinschnittstelle eines FB: Input, InOut, Temp	Zeiger auf die Quellvariable, deren Adresse ausgelesen wird
DST	Output	Referenz auf: • Bitfolgen, außer BOOL, • Ganzzahlen, • Gleitpunktzahlen, • Zeichenfolgen, • PLC-Datentypen (UDT), • Systemdatentypen (SDT), • ARRAYs aus den genannten Datentypen	• Bausteinschnittstelle eines FC: Input, Output, Temp • Bausteinschnittstelle eines FB: Temp	Referenz, an die die Adresse der Quellvariable übergeben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Regeln

Für den Zuweisungsversuch gelten die folgenden Regeln. VARIANT-Variablen, die nicht diesen Regeln entsprechen, liefern zur Laufzeit den Wert "NULL" zurück.

- Der VARIANT muss auf eine Adresse in einem optimierten Speicherbereich zeigen.
- Der VARIANT darf nicht auf eine Adresse in einem temporären Speicherberiech zeigen.
- Wenn Sie einer Referenz auf ein ARRAY einen VARIANT zuweisen wollen, gelten folgende Regeln:
  - Die VARIANT-Variable muss auf ein ARRAY zeigen, dessen Grenzen exakt mit denen der deklarierten Referenz übereinstimmen. Eine VARIANT-Variable, die auf ein ARRAY [0..9] zeigt, passt nicht zu einer Variable REF_TO ARRAY[1..10].
  - Darüberhinaus sollten Sie die Bausteine, die den Wert der VARIANT-Variable bilden, einmalig in einer CPU der Baureihe S7-1500, Firmwarestand V2.5, übersetzen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

In der Schnittstelle des Bausteins wurden der VARIANT "myVariant" und die Referenzvariable "myReference" deklariert.

Name	Data type
Input	
Output	
InOut	
myVariant	Variant
Static	
Temp	
myReference	REF_TO Int
Constant	

Im Programmcode wird versucht "myVariant" der Referenzvariable "myReference" zuzuweisen. Hat "myVariant" zur Laufzeit den Datentyp "Int", steht in "myReference" eine gültige Referenz auf die Quellvariable des VARIANT, andernfalls eine NULL. Ist die anschließende Abfrage auf "NOT_NULL" wahr, bedeutet dies, dass die Zuweisung erfolgreich war und der Wert "10" in die Zielvariable geschrieben werden kann.

```
1  #myReference ?= #myVariant;
2
3  IF #myReference <> NULL THEN
4     #myReference^ := 10;
5  END_IF;
```

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Zuweisungsversuch von VARIANT auf eine Referenz (Seite 329)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

SWAP: Anordnung ändern

Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes eines Eingangswerts und speichern das Ergebnis im angegebenen Operanden.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" vertauscht werden:

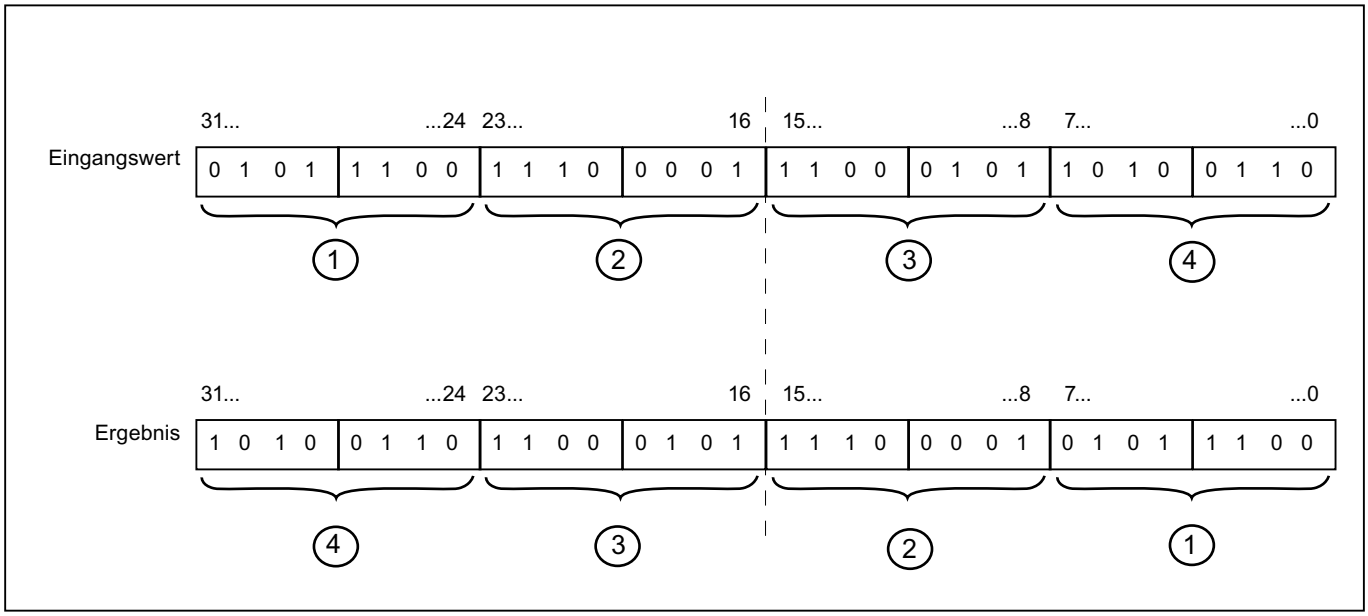

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
<Ausdruck>	Input	WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L, P	Eingangswert
Funktionswert		WORD, DWORD	WORD, DWORD, LWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SWAP("Tag_Value");
```

Das Ergebnis der Anweisung wird als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	0000 1111 0101 0101
Tag_Result	0101 0101 0000 1111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ARRAY-DB

ReadFromArrayDB: Aus Array-Datenbaustein lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie das Element aus einem Datenbaustein vom Bausteintyp ARRAY-DB, auf das der Index verweist, und schreiben den Wert des Elements in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der gelesen und ausgegeben wird
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Ergebnis der Anweisung
1) Der Parameter VALUE ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz, schreibgeschützt oder liegt im Ladespeicher.
8135	Der ARRAY-Datenbaustein enthält ungültige Werte.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8452	Codeerzeugungsfehler
8453	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TagResult" := ReadFromArrayDB(DB := "ArrayDB",
                               INDEX := 2,
                               VALUE => "TargetField");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.

Aus dem "ArrayDB" wird das 2. Element ausgelesen und in den Operanden "TargetField" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu SCL (Seite 7607)

WriteToArrayDB: In Array-Datenbaustein schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der geschrieben wird
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8134	Der Datenbaustein ist schreibgeschützt.
8135	Der Datenbaustein ist kein ARRAY-Datenbaustein.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8350	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8352	Codeerzeugungsfehler
8353	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TagResult" := WriteToArrayDB(DB := "ArrayDB",
                              INDEX := 2,
                              VALUE := "SourceField");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Der Wert des Operanden "SourceField" wird in das 2. Element des ARRAY-DB geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu SCL (Seite 7607)

ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie aus einem Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher das Element, auf das der Index verweist, und schreiben sie in den Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = "1": Mit dem Auslesen des Array-DBs beginnen
DB [1]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1]	InOut	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Kontanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8235	Der Datenbaustein ist kein ARRAY-DB.

Fehler-code* (W#16#...)	Erläuterung
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8750	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"ReadFromArrayDBL_DB"(REQ := "TagReg",
                      DB := "ArrayDB",
                      INDEX := 2,
                      VALUE := "TargetField",
                      BUSY => "TagBusy",
                      DONE => "TagDone",
                      ERROR => "TagError");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und am Parameter VALUE ausgegeben. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird nicht mehr verändert. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu SCL (Seite 7607)

WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie das Element, auf das der Index verweist, in einen Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of <Datentyp> besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = "1": Mit dem Schreiben in den Array-DB beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALUE [1]	Input	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8234	Der Datenbaustein ist schreibgeschützt.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"WriteToArrayDBL_DB"(REQ := "TagReg",
                     DB := "ArrayDB",
                     INDEX := 2,
                     VALUE := "SourceField",
                     BUSY => "TagBusy",
                     DONE => "TagDone",
                     ERROR => "TagError");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird in das 2. Element des "ArrayDB" geschrieben. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen (Seite 2825)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben (Seite 2829)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu SCL (Seite 7607)

Lese-/Schreibzugriff

PEEK: Speicheradresse lesen

Beschreibung

Mit der Anweisung "Speicheradresse lesen" lesen Sie eine Speicheradresse ohne Angabe eines Datentyps aus einem Standard-Speicherbereich aus.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

PEEK_<Datentyp>();

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

SCL

```
#Peeker := PEEK(AREA := 16#84,
                DBNUMBER := 1,
                BYTEOFFSET := 2,
                ENO => ENO);
IF NOT ENO THEN;
     #Peeker := 16#ffff;
END_IF;
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#1: Peripherieeingang (nur S7-1500)
DBNUM-BER	Input	DINT, DB_ANY	D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFF-SET	Input	DINT	E, A, M, D	Adresse, an der ausgelesen wird Es werden nur die 16 niederstwertigen Bits verwendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
_<Datentyp>		Bitfolgen Vorbelegung: BYTE	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Bitfolgen	E, A, M, D	Ergebnis der Anweisung

Hinweis

Wenn Sie die Speicheradresse aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := PEEK(AREA := "Tag_Area",
                      DBNUMBER := "Tag_DBNumber",
                      BYTEOFFSET := "Tag_Byte");
```

SCL

```
"Tag_Result2" := PEEK_WORD(AREA := "Tag_Area",
                           DBNUMBER := "Tag_DBNumber",
                           BYTEOFFSET := "Tag_Byte");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
Funktionswert	Tag_Result1	Wert des Bytes "20"
Funktionswert	Tag_Result2	Wert des Wortes "20"

Die Anweisung liest den Wert an der Adresse "20" aus dem Operanden "Tag_Byte" aus dem Datenbaustein "5" aus und liefert das Ergebnis am Operanden "Tag_Result" als Funktionswert zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

PEEK_BOOL: Speicherbit lesen

Beschreibung

Mit der Anweisung "Speicherbit lesen" lesen Sie ein Speicherbit ohne Angabe eines Datentyps aus einem Standard-Speicherbereich aus.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

SCL

```
#Peeker_BOOL := PEEK_BOOL(AREA := 16#84,
                          DBNUMBER := 1,
                          BYTEOFFSET := 2,
                          BITOFFSET := 1,
                          ENO => ENO);
IF NOT ENO THEN;
     #Peeker_BOOL := 0;
END_IF;
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#1: Peripherieeingang (nur S7-1500)
DBNUMBER	Input	DINT, DB_ANY	D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFFSET	Input	DINT	E, A, M, D	Adresse, an der ausgelesen wird Es werden nur die 16 niederstwertigen Bits verwendet.
BITOFFSET	Input	INT	E, A, M, D	Bit, an dem ausgelesen wird
Funktionswert		BOOL	E, A, M, D	Ergebnis der Anweisung

Hinweis

Wenn Sie das Speicherbit aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := PEEK_BOOL(AREA := "Tag_Area",
                          DBNUMBER := "Tag_DBNumber",
                          BYTEOFFSET := "Tag_Byte",
                          BITOFFSET := "Tag_Bit");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
Funktionswert	Tag_Result	3

Die Anweisung liest den Wert am Speicherbit "3" aus dem Operanden "Tag_Bit" aus dem Datenbaustein "5" aus dem Byte "20" aus und liefert das Ergebnis am Operanden "Tag_Result" als Funktionswert zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

POKE: Speicheradresse schreiben

Beschreibung

Mit der Anweisung "Speicheradresse schreiben" schreiben Sie eine Speicheradresse ohne Angabe eines Datentyps in einen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

SCL

```
POKE(AREA := 16#84,
     DBNUMBER := 1,
     BYTEOFFSET := 2,
     VALUE := BYTE#7,
     ENO => ENO);
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#2: Peripherieausgang (nur S7-1500)
DBNUM-BER	Input	DINT, DB_ANY	D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BYTEOFF-SET	Input	DINT	E, A, M, D	Adresse, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
VALUE	Input	Bitfolgen	E, A, M, D	Wert, der geschrieben wird

Hinweis

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
POKE(AREA := "Tag_Area",
     DBNUMBER := "Tag_DBNumber",
     BYTEOFFSET := "Tag_Byte",
     VALUE := "Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
VALUE	Tag_Value	16#11

Die Anweisung überschreibt die Speicheradresse "20" im Datenbaustein "5" mit dem Wert "16#11".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

POKE_BOOL: Speicherbit schreiben

Beschreibung

Mit der Anweisung "Speicherbit schreiben" schreiben Sie ein Speicherbit ohne Angabe eines Datentyps in einen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

SCL

```
POKE(AREA := 16#84,
     DBNUMBER := 1,
     BYTEOFFSET := 2,
     BITOFFSET := 4,
     VALUE := TRUE,
     ENO => ENO);
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA	Input	BYTE	E, A, M, D	Folgende Bereiche können ausgewählt werden: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB • 16#2: Peripherieausgang (nur S7-1500)
DBNUMBER	Input	DINT, DB_ANY	D	Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"
BYTEOFFSET	Input	DINT	E, A, M, D	Adresse, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
BITOFFSET	Input	INT	E, A, M, D	Bit, das geschrieben wird
VALUE	Input	BOOL	E, A, M, D	Wert, der geschrieben wird

Hinweis

Wenn Sie das Speicherbit in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
POKE_BOOL(AREA := "Tag_Area",
          DBNUMBER := "Tag_DBNumber",
          BYTEOFFSET := "Tag_Byte",
          BITOFFSET := "Tag_Bit",
          VALUE := "Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
VALUE	Tag_Value	M0.0

Die Anweisung überschreibt das Speicherbit "3" im Datenbaustein "5" im Byte "20" mit dem Wert "M0.0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

POKE_BLK: Speicherbereich schreiben

Beschreibung

Mit der Anweisung "Speicherbereich schreiben" schreiben Sie einen Speicherbereich ohne Angabe eines Datentyps in einen anderen Standard-Speicherbereich.

Verwendung des Speicherbereichs 16#84: DB

Wenn Sie als Speicherbereich einen Datenbaustein verwenden und dieser bei der Erstellung des Programmcodes noch nicht bekannt ist, dann haben Sie die Möglichkeit, den EN-/ENO-Mechanismus zu verwenden, um mögliche Zugriffsfehler zu erkennen:

SCL

```
POKE_BLK(AREA_SRC := 16#84,
         DBNUMBER_SRC := 1,
         BYTEOFFSET_SRC := 2,
         AREA_DEST := 16#84,
         DBNUMBER_DEST := 11,
         BYTEOFFSET_DEST := 22,
```

SCL

```
COUNT := 3,
ENO => ENO);
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
AREA_SRC	Input	BYTE	E, A, M, D	Folgende Bereiche können Sie im Quellspeicherbereich auswählen: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB
DBNUMBER_SRC	Input	DINT, DB_ANY [1]	D	Nummer des Datenbausteins im Quellspeicherbereich, wenn AREA = DB, ansonsten "0"
BYTEOFF-SET_SRC	Input	DINT	E, A, M, D	Adresse im Quellspeicherbereich, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
AREA_DEST	Input	BYTE	E, A, M, D	Folgende Bereiche können Sie im Zielspeicherbereich auswählen: • 16#81: Input • 16#82: Output • 16#83: Merker • 16#84: DB
DBNUM-BER_DEST	Input	DINT, DB_ANY [1]	D	Nummer des Datenbausteins im Zielspeicherbereich, wenn AREA = DB, ansonsten "0"
BYTEOFF-SET_DEST	Input	DINT	E, A, M, D	Adresse im Zielspeicherbereich, die geschrieben wird Es werden nur die 16 niederstwertigen Bits verwendet.
COUNT	Input	DINT	E, A, M, D	Anzahl der Bytes, die kopiert werden
[1] Die Datentypen der Parameter DBNUMBER_SRC und DBNUMBER_DEST müssen den gleichen Datentyp haben. D. h., dass beide Variablen entweder vom Datentyp DINT oder vom Datentyp DB_ANY sein müssen.				

Hinweis

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter DBNUMBER mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
POKE_BLK(AREA_SRC := "Tag_Source_Area",
         DBNUMBER_SRC := "Tag_Source_DBNumber",
         BYTEOFFSET_SRC := "Tag_Source_Byte"),
         AREA_DEST := "Tag_Destination_Area",
         DBNUMBER_DEST := "Tag_Destination_DBNumber",
         BYTEOFFSET_DEST := "Tag_Destination_Byte",
         COUNT := "Tag_Count");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
AREA_SRC	Tag_Source_Area	16#84
DBNUMBER_SRC	Tag_Source_DBNumber	5
BYTEOFFSET_SRC	Tag_Source_Byte	20
AREA_DEST	Tag_Destination_Area	16#83
DBNUMBER_DEST	Tag_Destination_DBNumber	0
BYTEOFFSET_DEST	Tag_Destination_Byte	30
COUNT	Tag_Count	100

Die Anweisung schreibt 100 Byte aus dem Datenbaustein "5" beginnend bei der Adresse "20" in den Merker-Speicherbereich beginnend bei der Adresse "30".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

READ_LITTLE: Daten im Little-Endian-Format lesen

Beschreibung

Mit der Anweisung "Daten im Little-Endian-Format lesen" lesen Sie Daten aus einem Speicherbereich aus und schreiben diese in eine einzelne Variable in der Byte-Reihenfolge Little-Endian. Beim Little-Endian-Format wird das Byte mit den niederstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_ARRAY und DEST_VARIABLE sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter DEST_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter SRC_ARRAY zeigt auf den Speicherbereich, aus dem gelesen wird, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Lesen begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL lesen

Wenn Sie eine Variable lesen möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL lesen möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200 ab V4.0	S7-1500		
SRC_ARRAY	Input	ARRAY of BYTE	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, aus dem gelesen wird
DEST_VARIABLE	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, LDT, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L	Gelesener Wert
POS	InOut	DINT	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Lesen begonnen wird. Der Parameter POS wird nullbasiert berechnet.
Funktionswert (RET_VAL)		INT	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#TagResult := READ_LITTLE(SRC_ARRAY := #SourceField,
                          DEST_VARIABLE => #DINTVariable,
                          POS := #TagPos);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_ARRAY	#SourceField	ARRAY[0..3] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
POS	#TagPos	0 => 4

Die Anweisung liest die Ganzzahl 1_295_788_826 aus dem Speicherbereich #SourceField aus und schreibt sie im Little-Endian-Format in den Operanden #DINTVariable. Der Datentyp am Parameter DEST_VARIABLE gibt an, wie viele Byte ausgelesen werden. Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Deserialize: Deserialisieren (Seite 1799)

Serialize: Serialisieren (Seite 1803)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zu SCL (Seite 7607)

WRITE_LITTLE: Daten im Little-Endian-Format schreiben

Beschreibung

Mit der Anweisung "Daten im Little-Endian-Format schreiben" schreiben Sie die Daten einer einzelnen Variable in der Byte-Reihenfolge Little-Endian in einen Speicherbereich. Beim Little-Endian-Format wird das Byte mit den niederstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_VARIABLE und DEST_ARRAY sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter SRC_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter DEST_ARRAY zeigt auf den Speicherbereich, in den die Daten geschrieben werden, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Schreiben begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL schreiben

Wenn Sie eine Variable schreiben möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL schreiben möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200 ab V4.0	S7-1500		
SRC_VARIABLE	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, LDT, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L	Variable, deren Daten geschrieben werden
DEST_ARRAY	InOut	ARRAY of BYTE	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, in den die Daten geschrieben werden
POS	InOut	DINT	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Schreiben begonnen wird. Der Parameter POS wird nullbasiert berechnet.
Funktionswert (RET_VAL)		INT	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#TagResult := WRITE_LITTLE(SRC_VARIABLE := #DINTVariable,
                           DEST_ARRAY := #TargetField,
                           POS := #TagPos);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

Die Anweisung schreibt die Ganzzahl 1_295_788_826 aus dem Operanden #DINTVariable im Little-Endian-Format in den Speicherbereich #TargetField. Der Datentyp am Parameter SRC_VARIABLE gibt an, wie viele Byte geschrieben werden. Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Deserialize: Deserialisieren (Seite 1799)

Serialize: Serialisieren (Seite 1803)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zu SCL (Seite 7607)

READ_BIG: Daten im Big-Endian-Format lesen

Beschreibung

Mit der Anweisung "Daten im Big-Endian-Format lesen" lesen Sie die Daten aus einem Speicherbereich aus und schreiben diese in eine einzelne Variable in der Byte-Reihenfolge Big-Endian. Beim Big-Endian-Format wird das Byte mit den höchstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_ARRAY und DEST_VARIABLE sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter DEST_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter SRC_ARRAY zeigt auf den Speicherbereich, aus dem gelesen wird, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Lesen begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL lesen

Wenn Sie eine Variable lesen möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL lesen möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200 ab V4.0	S7-1500		
SRC_AR-RAY	Input	ARRAY of BYTE	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, aus dem gelesen wird
DEST_VARI-ABLE	Output	Bitfolgen, Ganzzahlen, Gleitpunkt-zahlen, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunkt-zahlen, LDT, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L	Gelesener Wert
POS	InOut	DINT	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Lesen be-gonnen wird. Der Para-meter POS wird nullba-siert berechnet.
Funktionswert (RET_VAL)		INT	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#TagResult := READ_BIG(SRC_ARRAY := #SourceField,
                       DEST_VARIABLE => #DINTVariable,
                       POS := #TagPos);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_ARRAY	#SourceField	ARRAY[0..10] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
POS	#TagPos	0 => 4

Die Anweisung liest die Ganzzahl 439_041_101 aus dem Speicherbereich #SourceField aus und schreibt sie im Big-Endian-Format in den Operanden #DINTVariable. Der Datentyp am Parameter DEST_VARIABLE gibt an, wie viele Byte ausgelesen werden. Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Deserialize: Deserialisieren (Seite 1799)

Serialize: Serialisieren (Seite 1803)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zu SCL (Seite 7607)

WRITE_BIG: Daten im Big-Endian-Format schreiben

Beschreibung

Mit der Anweisung "Daten im Big-Endian-Format schreiben" schreiben Sie die Daten einer einzelnen Variable in der Byte-Reihenfolge Big-Endian in einen Speicherbereich. Beim Big-Endian-Format wird das Byte mit den höchstwertigen Bits zuerst gespeichert, d. h. an der kleinsten Speicheradresse.

Die Parameter SRC_VARIABLE und DEST_ARRAY sind vom Datentyp VARIANT. Allerdings gibt es ein paar Einschränkungen bezüglich des Datentyps, mit dem die Parameter verschaltet werden können. Der VARIANT am Parameter SRC_VARIABLE muss auf einen elementaren Datentyp zeigen. Der VARIANT am Parameter DEST_ARRAY zeigt auf den Speicherbereich, in den die Daten geschrieben werden, und dieser muss ein ARRAY of BYTE sein.

Sie können bei den Parameter SRC_ARRAY und DEST_VARIABLE auch einen Aktualparameter mit variablem ARRAY-Index verwenden.

Der Operand am Parameter POS bestimmt die Position im Speicherbereich, an der mit dem Schreiben begonnen wird.

Hinweis

Variable vom Datentyp VARIANT oder BOOL schreiben

Wenn Sie eine Variable schreiben möchten, auf die ein VARIANT zeigt, dann verwenden Sie die Anweisungen "Serialisieren" oder "Deserialisieren".

Wenn Sie eine Variable vom Datentyp BOOL schreiben möchten, dann verwenden Sie einen "Slice-Zugriff".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicher-bereich	Beschreibung
		S7-1200 ab V4.0	S7-1500		
SRC_VARI-ABLE	Input	Bitfolgen, Ganzzahlen, Gleitpunkt-zahlen, TOD, DATE, CHAR, WCHAR	Bitfolgen, Ganzzahlen, Gleitpunkt-zahlen, LDT, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L	Variable, deren Daten geschrieben werden
DEST_AR-RAY	InOut	ARRAY of BYTE	ARRAY of BYTE	E, A, M, D, L	Speicherbereich, in den die Daten geschrieben werden
POS	InOut	DINT	DINT	E, A, M, D, L	Bestimmt die Position, an der mit dem Schrei-ben begonnen wird. Der Parameter POS wird nullbasiert berechnet.
Funktionswert (RET_VAL)		INT	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der Datentyp am Parameter SRC_ARRAY ist kein ARRAY of BYTE.
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des ARRAYs.
8383	Der Wert am Parameter POS befindet sich zwar innerhalb der Grenzwerte des ARRAYs, aber die Größe des Speicherbereichs überschreitet die obere Grenze des ARRAYs.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#TagResult := WRITE_BIG(SRC_VARIABLE := #DINTVariable,
                        DEST_ARRAY := #TargetField,
                        POS := #TagPos);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
SRC_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

Die Anweisung schreibt die Ganzzahl 439_041_101 aus dem Operanden #DINTVariable im Big-Endian-Format in den Speicherbereich #TargetField. Der Datentyp am Parameter SRC_VARIABLE gibt an, wie viele Byte geschrieben werden. Am Operanden #TagPos wird die Anzahl 4 gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Deserialize: Deserialisieren (Seite 1799)

Serialize: Serialisieren (Seite 1803)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 115)

Grundlagen zu SCL (Seite 7607)

VARIANT

VariantGet: Wert einer VARIANT-Variable auslesen

Beschreibung

Mit der Anweisung "Wert einer VARIANT-Variable auslesen" lesen Sie den Wert der Variable aus, auf die der VARIANT am Parameter SRC zeigt, und schreiben diesen in die Variable am Parameter DST.

Der Parameter SRC hat den Datentyp VARIANT. Am Parameter DST kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable, die am Parameter DST angegeben ist, muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
SRC	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Variable, die ausgelesen wird
DST	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
VariantGet(SRC := #TagIn_Source,
           DST => "TagOut_Dest");
```

Der Wert der Variable, auf die der VARIANT am Operand #TagIn_Source zeigt, wird ausgelesen und in den Operand "TagOut_Dest" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1810)

Grundlagen zu SCL (Seite 7607)

VariantPut: Wert in eine VARIANT-Variable schreiben

Beschreibung

Mit der Anweisung "Wert in eine VARIANT-Variable schreiben" schreiben Sie den Wert der Variable, die am Parameter SRC angegeben ist, in den Speicher am Parameter DST, auf den der VARIANT zeigt.

Der Parameter DST hat den Datentyp VARIANT. Am Parameter SRC kann jeder Datentyp, außer VARIANT, angegeben werden.

Der Datentyp der Variable am Parameter SRC muss mit dem Datentyp übereinstimmen, auf den der VARIANT zeigt.

Hinweis

Zum Kopieren von Strukturen und ARRAYs können Sie die Anweisung "MOVE_BLK_VARIANT: Bereich kopieren" verwenden. Weitere Informationen finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
SRC	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, ARRAY-Elemente, PLC-Datentypen	E, A, M, D, L	E, A, M, D, L, P	Variable, die ausgelesen wird
DST	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)		Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
VariantPut(SRC := "TagIn_Source",
           DST := #TagIn_Dest);
```

Der Wert des Operanden "TagIn_Source" wird in die Variable, auf die der VARIANT am Operand #TagIn_Dest zeigt, geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

MOVE_BLK_VARIANT: Bereich kopieren (Seite 1810)

Grundlagen zu SCL (Seite 7607)

CountOfElements: Anzahl der ARRAY-Elemente abfragen

Beschreibung

Mit der Anweisung "Anzahl der ARRAY-Elemente abfragen" fragen Sie ab, wie viele ARRAY-Elemente eine Variable hat, auf die der VARIANT zeigt.

Wenn es sich um ein eindimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl der ARRAY-Elemente ausgegeben. (Die Differenz zwischen dem oberen und dem unteren Grenzwert + 1). Wenn es sich um ein mehrdimensionales ARRAY handelt, dann wird als Ergebnis die Anzahl aller Dimensionen ausgegeben.

Wenn Sie die Elemente eines ARRAY-DB abfragen möchten, dann sollten Sie die Anweisungen "ReadFromArrayDB" bzw. "WriteFromArrayDB" verwenden, da hier eine genauere Fehlerauswertung für die Anzahl der Elemente möglich ist.

Hinweis

Instanzen

Der VARIANT-Zeiger kann auf keine Instanz und damit auch auf keine Multiinstanz oder ein ARRAY of Multiinstanzen zeigen.

Hinweis

ARRAY innerhalb eines Datenbausteins

Wenn Sie die Anzahl der Elemente eines ARRAYs, das in einem Datenbaustein liegt, abfragen möchten, dann darf bei diesem Datenbaustein nicht das Bausteinattribut "Datenbaustein im Gerät schreibgeschützt" aktiviert sein. Ansonsten liefert der Parameter RET_VAL das Ergebnis "0", unabhängig davon, wie viele Elemente das ARRAY enthält.

Das Ergebnis ist ebenfalls "0", wenn die VARIANT-Variable kein ARRAY ist.

Wenn der VARIANT auf ein ARRAY of BOOL zeigt, dann werden auch die Füllelemente mitgezählt. (Z. B. bei einem ARRAY[0..1] of BOOL wird 8 zurückgegeben).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die abgefragt wird
Funktionswert		UDINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF IS_ARRAY(#Tag_VARIANTToArray) THEN
"Tag_Result" := CountOfElements(#Tag_VARIANTToArray);
END_IF;
```

Wenn es sich bei der Variable, auf die der VARIANT zeigt, um ein ARRAY handelt, dann wird die Anzahl der ARRAY-Elemente ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ARRAY[*]

LOWER_BOUND: Untere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, untere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Untere ARRAY-Grenze auslesen" zur Verfügung.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, untere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, untere Grenze ausgelesen werden soll.
Funktionswert		DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Result" := LOWER_BOUND(ARR := #ARRAY_A,
                        DIM := 2);
```

Die Anweisung liest die variable, untere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu SCL (Seite 7607)

UPPER_BOUND: Obere ARRAY-Grenze auslesen

Beschreibung

Sie können in der Bausteinschnittstelle eines Funktionsbausteins oder einer Funktion Variablen vom Datentyp ARRAY[*] deklarieren. Für diese lokalen Variablen können Sie die Grenzen des ARRAYs auslesen. Am Parameter DIM müssen Sie die gewünschte Dimension angeben.

Um die variable, obere Grenze des ARRAYs auszulesen, steht Ihnen die Anweisung "Obere ARRAY-Grenze auslesen" zur Verfügung.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung steht Ihnen ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ARR	Input	ARRAY[*]	FB => Abschnitt InOut FC => Abschnitte Input und InOut	ARRAY, dessen variable, obere Grenze ausgelesen werden soll.
DIM	Input	UDINT	E, A, M, D, L, P oder Konstante	Dimension des ARRAYs, deren variable, obere Grenze ausgelesen werden soll.
Funktionswert		DINT	E, A, M, D, L, P	Ergebnis

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Result" := UPPER_BOUND(ARR := #ARRAY_A,
                        DIM := 2);
```

Die Anweisung liest die variable, obere Grenze des ARRAYs #ARRAY_A aus der zweiten Dimension aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird das Ergebnis in den Operanden "Result" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum ARRAY (Seite 296)

Grundlagen zu SCL (Seite 7607)

Legacy

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

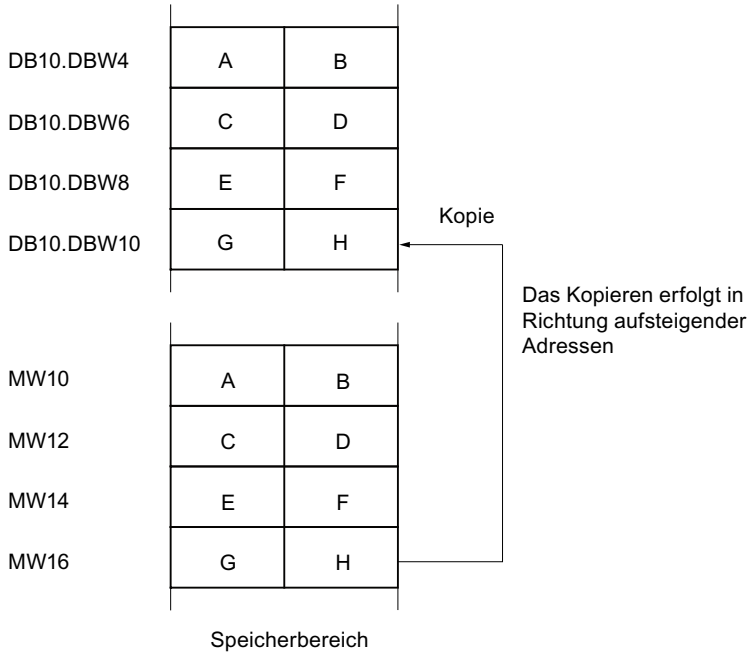

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
DSTBLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
Funktionswert (RET_VAL)		INT	E, A, M, D, L, P	Fehlerinformationen
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.

Fehlercode* (W#16#...)	Erläuterung
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := BLKMOV(SRCBLK := P#M100.0 BYTE 10,
                       DSTBLK => P#DB1.DBX0.0 BYTE 10);
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Grundlagen zu SCL (Seite 7607)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
DSTBLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
Funktionswert (RET_VAL)		INT	E, A, M, D, L, P	Fehlerinformationen
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := UBLKMOV(SRCBLK := P#M100.0 BYTE 10,
                        DSTBLK => P#DB1.DBX0.0 BYTE 10);
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Grundlagen zu SCL (Seite 7607)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Den Quell- und Zielbereich können Sie auch durch den Datentyp ANY definieren

Wenn Sie den Datentyp ANY verwenden, dann müssen Sie im Zusammenhang mit dem Datentyp STRING folgendes beachten:

- Bei einer Zuweisung von STRING (Quellbereich) über ANY nach STRING (Zielbereich) wird der Inhalt des STRING so oft in den Zielbereich kopiert, bis dieser voll ist.
  Quellbereich: 'STEP7-SCL-TIA-Portal'
  Zielbereich: 'STEP7-SCL-TIA-PortalSTEP7-SCL-TIA-PortalSTEP7-SCL'
- Bei einer Zuweisung von WSTRING (Quellbereich) über ANY nach WSTRING (Zielbereich) wird der gesamte WSTRING inklusive Typisierung nur einmal in den Zielbereich kopiert.
  Quellbereich: WSTRING#'STEP7-SCL-TIA-Portal'
  Zielbereich: WSTRING#'STEP7-SCL-TIA-Portal'

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

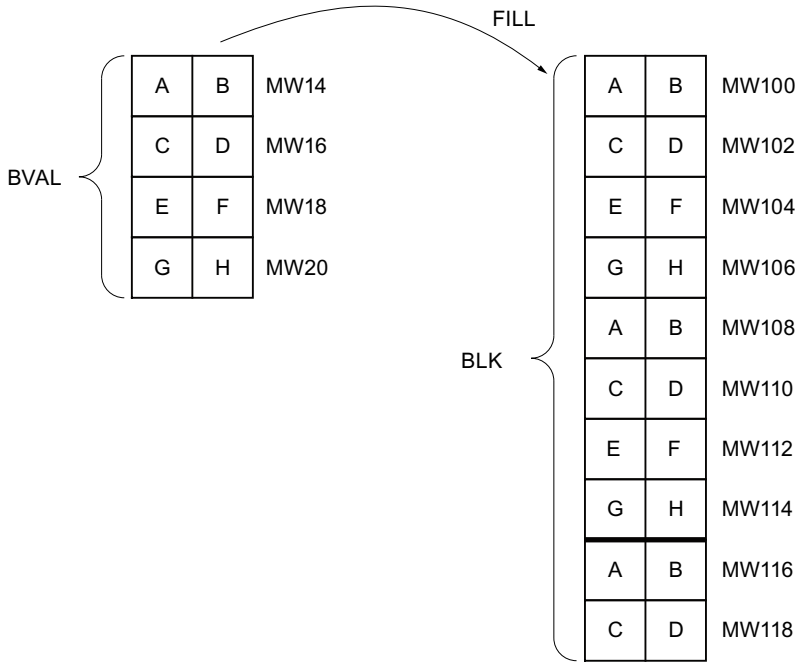

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker

- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
BLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
Funktionswert (RET_VAL)		INT	E, A, M, D, L, P	Fehlerinformationen
1) Der Parameter BLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter BVAL

Wenn Sie als Eingangsparameter eine Struktur übergeben, dann müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter BVAL werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
8352	Am Parameter BLK werden die Datentypen WSTRING, WCHAR, BOOL, ARRAY of STRING, ARRAY of WSTRING und ARRAY of WCHAR nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := FILL(BVAL := P#M14.0 WORD 4,
                     BLK => P#M100.0 WORD 10);
```

Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Grundlagen zu SCL (Seite 7607)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" programmieren Sie explizite Konvertierungen. Beim Einfügen der Anweisung wird der Dialog "CONVERT" geöffnet. In diesem geben Sie den Quelldatentyp und den Zieldatentyp der Konvertierung an. Der Quellwert wird gelesen und in den angegebenen Zieldatentyp konvertiert.

Konvertierungsmöglichkeiten von Bitfolgen

Die Bitfolgen BYTE und WORD sind in der Anweisungsbox nicht auswählbar. Es ist allerdings möglich, einen Operanden vom Datentyp DWORD oder LWORD an einem Parameter der Anweisung anzugeben, wenn die Länge des Eingangs- und des Ausgangsoperanden übereinstimmt. Dann wird der Operand vom Datentyp einer Bitfolge entsprechend des Datentyps des Eingangs- oder Ausgangsparameters interpretiert und implizit konvertiert. Der Datentyp DWORD wird z. B. als DINT/UDINT und LWORD als LINT/ULINT interpretiert. Diese Konvertierungsmöglichkeiten stehen Ihnen auch bei aktivierter IEC-Prüfung zur Verfügung.

Hinweis

Für CPUs der Baureihen S7-1500 gilt: Die Datentypen DWORD und LWORD können nur von bzw. nach Datentyp REAL oder LREAL konvertiert werden.

Dabei wird das Bitmuster des Quellwerts ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand des Freigabeausgangs ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand des Freigabeausgangs ENO = 0.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Quelltyp>	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
<Zieltyp>	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Ergebnis der Konvertierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_INT" := REAL_TO_INT("Tag_REAL");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Datentyp	Wert
Tag_REAL	REAL	20.56
Tag_INT	INT	21

Bei der Konvertierung wird der Wert des Operanden "Tag_REAL" zur nächsten Ganzzahl gerundet und im Operanden "Tag_INT" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl oder eine Gleitpunktzahl um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert, der gerundet wird.
Funktionswert		Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Rundung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROUND("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	1.50000000	-1.50000000
Tag_Result	2	-2

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie einen Wert zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst höhere ganze Zahl um. Der Funktionswert kann dabei größer oder gleich dem Eingangswert sein.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

CEIL_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
_<Datentyp>		Ganzzahlen, Gleitpunktzahlen Vorbelegung: DINT	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Aufgerundeter Eingangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := CEIL("Tag_Value");
"Tag_Result2" := CEIL_REAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0.5	-0.5
Tag_Result1	1	0
Tag_Result2	1.0	0.0

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie einen Wert zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst niedrigere ganze Zahl um. Der Funktionswert kann dabei kleiner oder gleich dem Eingangswert sein.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

FLOOR_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
_<Datentyp>		Ganzzahlen, Gleitpunktzahlen Vorbelegung: DINT	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Abgerundeter Eingangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := FLOOR("Tag_Value");
"Tag_Result2" := FLOOR_REAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0.5	-0.5
Tag_Result1	0	-1
Tag_Result2	0.0	-1.0

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Eingangswert eine Ganzzahl ohne Rundung. Die Anweisung selektiert nur den ganzzahligen Anteil des Eingangswerts und gibt diesen ohne die Nachkommastellen als Funktionswert zurück.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

TRUNC_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L	Eingangswert
_<Datentyp>		Ganzzahlen, Gleitpunktzahlen Vorbelegung: DINT	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Ganzzahliger Anteil des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := TRUNC("Tag_Value1");
"Tag_Result2" := TRUNC("Tag_Value2"+"Tag_Value3");
"Tag_Result3" := TRUNC_SINT("Tag_Value4");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value1	-1.5
Tag_Result1	-1

Operand	Wert
Tag_Value2	2.1
Tag_Value3	3.2
Tag_Result2	5
Tag_Result3	2
Tag_Value4	2.4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SCALE_X: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie eine Gleitpunktzahl, indem Sie diese auf einem bestimmten Wertebereich abbilden. Den Wertebereich legen Sie durch die Parameter MIN und MAX fest. Das Ergebnis der Skalierung ist eine Ganzzahl.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

SCALE_X_<Datentyp>();

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:

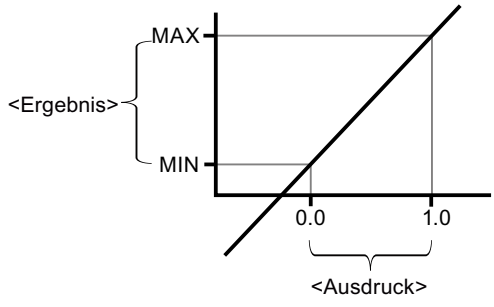

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [VALUE * (MAX – MIN)] + MIN

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Untere Grenze des Wertebereichs
VALUE	Input	Gleitpunktzahlen	E, A, M, D, L	Wert, der skaliert wird. Bei Angabe einer Konstante müssen Sie diese deklarieren.
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Obere Grenze des Wertebereichs
_<Datentyp>		Ganzzahlen, Gleitpunktzahlen Vorbelegung: INT	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Ganzzahlen, Gleitpunktzahlen	-	Ergebnis der Skalierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := SCALE_X(MIN := "Tag_Value1",
                         VALUE := "Tag_Real",
                         MAX := "Tag_Value2");
"Tag_Result2" := SCALE_X_REAL(MIN := "Tag_Value1",
                              VALUE := "Tag_Real",
                              MAX := "Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Real	0.5
Tag_Value1	10
Tag_Value2	30
Tag_Result1	20
Tag_Result2	20.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Globale Konstanten deklarieren (Seite 7392)

Grundlagen zu SCL (Seite 7607)

NORM_X: Normieren

Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert die Anweisung den Wert "0.0" als Ergebnis zurück. Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert die Anweisung den Wert "1.0" als Ergebnis zurück.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

NORM_X_<Datentyp>();

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:

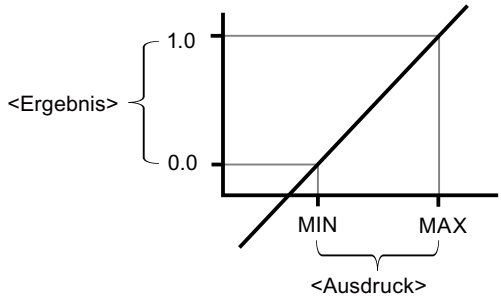

Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

OUT = (VALUE – MIN) / (MAX – MIN)

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Untere Grenze des Wertebereichs
VALUE [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Wert, der normiert wird.
MAX [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Obere Grenze des Wertebereichs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
_<Datentyp>		Gleitpunktzahlen Vorbelegung: REAL	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Normierung
[1] Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := NORM_X(MIN := "Tag_Value1",
                        VALUE := "Tag_InputValue",
                        MAX := "Tag_Value2");
"Tag_Result2" := NORM_X_LREAL(MIN := "Tag_Value1",
                              VALUE := "Tag_InputValue",
                              MAX := "Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InputValue	20
Tag_Value1	10
Tag_Value2	30

Operand	Wert
Tag_Result1	0.5
Tag_Result2	0.5

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Globale Konstanten deklarieren (Seite 7392)

Grundlagen zu SCL (Seite 7607)

REF: Referenz auf eine Variable bilden

Beschreibung

Mit dem Schlüsselwort "REF()" geben Sie an, auf welche Variable eine zuvor deklarierte Referenz zeigen soll. Als Parameter geben Sie die zu referenzierende Variable an.

Hinweis

Deklaration von Referenzen

Bevor Sie das Schlüsselwort "REF()" einsetzen, müssen Sie zunächst in der Bausteinschnittstelle eine Referenz deklarieren.

Siehe auch:

Referenzen deklarieren (Seite 318)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Bitfolgen, außer BOOL, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, PLC-Datentypen (UDT), Systemdatentypen (SDT), ARRAYs aus den genannten Datentypen	DB-Variable (optimiert), Bausteinschnittstelle eines FB (optimiert)	Variable, auf die die Referenz zeigen soll
Funktionswert		Der Datentyp des Funktionswerts entspricht dem der deklarierten Referenzvariable.	Bausteinschnittstelle eines FC: Input, Output, Temp Bausteinschnittstelle eines FB: Temp	Adresse der referenzierten Variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Regeln

Für das Referenzieren gelten folgende Regeln:

- Der Datentyp der Variable, die Sie in Klammer angeben, muss genau dem Datentyp der deklarierten Referenz entsprechen. Eine Referenz mit dem Datentyp "REF_TO Int" kann also nur auf eine Variable vom Datentyp "Int" zeigen. Eine Datentypkonvertierung findet nicht statt.
- Die Variable, die Sie als Parameter für "REF()" angeben, muss in einem optimierten Speicherbereich liegen.
- Eine Referenz darf nicht auf die folgenden Daten zeigen:
  - temporäre Daten (TEMP)
  - globale Variablen aus der PLC-Variablentabelle
  - Bausteinparameter
  - Konstanten
  - schreibgeschützte Variablen

Bei Referenzen auf Arrays gilt:

- Die Arraygrenzen und Dimensionen der Referenz und der referenzierten Variable müssen identisch sein.
- Array[*] wird nicht unterstützt.
- Referenzen auf ARRAY-DBs, die auf einem PLC-Datentyp basieren, müssen wie folgt gebildet werden:
  `REF("my_ArrayDB_UDT".THIS)`
  `REF("my_ArrayDB_UDT"."THIS"[i])`

Beispiel

In der Schnittstelle des Bausteins wurden die temporären Variablen "#myRefInt", "#myRefType" und "#myRefArray" deklariert.

	Name	Data type
	Input	
	Output	
	InOut	
	Static	
	a	Int
	Temp	
	myRefInt	REF_TO Int
	myRefType	REF_TO "myUDT"
	myRefARRAY	REF_TO Array[0..3] of Int
	Constant	

Im Programmcode werden diesen Referenzen konkrete Variablen zugewiesen:

SCL

```
#myRefInt := REF(#a);
#myRefType := REF("myDB".myUDT);
#myRefType := REF("myArrayDB_UDT"."THIS"[1]);
#myRefARRAY := REF("myDB".myArray);
```

Siehe auch

Grundlagen zu Referenzen (Seite 315)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

VARIANT

VARIANT_TO_DB_ANY: VARIANT in DB_ANY konvertieren

Beschreibung

Mit der Anweisung "VARIANT in DB_ANY konvertieren" fragen Sie die Datenbaustein-Nummer ab, die der Operand, der am Parameter IN angegeben ist, adressiert. Dabei kann es sich um einen Instanz-Datenbaustein oder um einen ARRAY-Datenbaustein handeln. Der Operand am Parameter IN hat den Datentyp VARIANT, daher muss zur Zeit der Programmerstellung nicht bekannt sein, welchen Datentyp der Datenbaustein hat, dessen Nummer abgefragt werden soll. Die Datenbaustein-Nummer wird zur Laufzeit ausgelesen und in den, am Parameter RET_VAL angegebenen, Operanden geschrieben.

Voraussetzungen

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind, dann wird "0" als Datenbausteinnummer ausgegeben.

Die Ausgangsvariable...	referenziert...	Konvertierungsmöglichkeiten
VARIANT	... einen Datenbaustein, der ein Instanz-Datenbaustein eines PLC-Datentyps oder eines Systemdatentyps (SDT) ist.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.
VARIANT	... einen Datenbaustein, der ein ARRAY-DB ist.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.
VARIANT	... eine Variable mit einem elementaren Datentyp.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da ein Datenbaustein keinesfalls aus nur einem elementaren Datentyp besteht.
VARIANT	... eine Struktur innerhalb eines Datenbausteins.	Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da es sich nur um einen Teil innerhalb des Datenbausteins handelt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Variable, die ausgelesen wird. (Der Funktionswert der Anweisung "DB_ANY_TO_VARIANT"). Am Parameter IN können Sie eine lokale oder eine globale Variable verwenden.
ERR	Output	INT	E, A, M, D, L	Fehlerinformationen
Funktionswert (RET_VAL)		DB_ANY	E, A, M, D, L	Ergebnis: Nummer des DBs

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
252C	Der Datentyp VARIANT am Parameter IN liefert den Wert "0" und die CPU geht in den Betriebszustand STOP.
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8131	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8132	Der Datenbaustein ist zu kurz und kein ARRAY-Datenbaustein.
8150	Der Datentyp VARIANT am Parameter IN liefert den Wert "0". Um diese Fehlermeldung zu erhalten, muss die Bausteineigenschaft "Lokale Fehlerbehandlung im Baustein" aktiviert sein. Ansonsten geht die CPU in den Betriebszustand STOP und gibt den Fehlercode 16#252C aus.
8153	Der Datentyp VARIANT am Parameter IN zeigt nicht auf den Anfang eines ARRAY-Datenbausteins oder die Länge des VARIANT stimmt nicht mit der des Datenbausteins überein.
8154	Der Datenbaustein hat den falschen Datentyp.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"OutputDBNumber" := VARIANT_TO_DB_ANY(IN := #tempVARIANT,
                                      ERR := "Tag_Error");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
IN	Input	tempVARIANT	-
<Funktionswert>	Output	OutputDBNumber	11

Die Nummer eines Datenbausteins, der am Operand tempVARIANT angegeben ist, wird ausgelesen. Da der Operand den Datentyp VARIANT hat, muss zur Programmerstellung noch nicht bekannt sein, welchen Datentyp die Variable hat. Die Nummer wird in die Variable "OutputDBNumber" geschrieben, die den Datentyp DB_ANY hat.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

DB_ANY_TO_VARIANT: DB_ANY in VARIANT konvertieren

Beschreibung

Mit der Anweisung "DB_ANY in VARIANT konvertieren" erzeugen Sie aus einem Datenbaustein eine VARIANT-Variable, die die unten stehenden Voraussetzungen erfüllt. Der Operand am Parameter IN hat den Datentyp DB_ANY, daher muss zur Zeit der Programmerstellung der Datenbaustein nicht bekannt sein. Die Datenbaustein-Nummer wird zur Laufzeit ausgelesen.

Voraussetzungen

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind oder der Datenbaustein nicht existiert, dann wird am Parameter RET_VAL der Wert NULL ausgegeben. Alle weiteren Zugriffe mit der Variable RET_VAL schlagen fehl.

Die Eingangsvariable vom Datentyp ...	referenziert...	Konvertierungsmöglichkeiten
DB_ANY	...einen Datenbaustein, der ein Instanz-Datenbaustein eines PLC-Datentyps oder eines Systemdatentyps (SDT) ist.	Die Konvertierung ist möglich
DB_ANY	...einen Datenbaustein, der ein ARRAY-DB ist.	Die Konvertierung ist möglich
DB_ANY	...einen Datenbaustein, der ein Instanz-Datenbaustein eines Funktionsbausteins oder ein globaler Datenbaustein ist.	Die Konvertierung ist nicht möglich

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DB_ANY	E, A, M, D, L	Datenbaustein, dessen Nummer ausgelesen wird. Am Parameter IN können Sie eine lokale oder eine globale Variable verwenden.
ERR	Output	INT	E, A, M, D, L	Fehlerinformationen
Funktionswert (RET_VAL) [1]		VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Nummer des Datenbausteins
1) Der Parameter RET_VAL ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8130	Die Nummer des Datenbaustein ist "0".
8131	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8132	Der Datenbaustein ist zu kurz und kein ARRAY-Datenbaustein.
8134	Der Datenbaustein ist schreibgeschützt.
8154	Der Datenbaustein hat den falschen Datentyp.
8155	Der Datenbaustein hat einen unbekannten Datentyp. [1]
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

[1] Die Ursache für die Ausgabe des Fehlercodes #8155 liegt darin:

Es wurde ein PLC-Datentyp (UDT1) deklariert und anschließend ein Datenbaustein (DB2) vom Datentyp "UDT1" erstellt. In der Variablentabelle gibt es eine Variable (3) vom Datentyp DB_ANY. In einem Programmbaustein (4) wurde anschließend die Anweisung "DB_ANY_TO_VARIANT" aufgerufen und am Parameter IN mit der Variable (3) versorgt. Beim Ausführen liefert die Anweisung "DB_ANY_TO_VARIANT" den Fehlercode 16#8155.

Zur Auflösung des Fehlercodes gehen Sie folgendermaßen vor:

1. Legen Sie eine Funktion (FC5) an und deklarieren Sie an der InOut-Schnittstelle eine Variable vom Datentyp VARIANT.
2. Legen Sie eine weitere Funktion (FC6) an und rufen Sie darin den FC5 auf.

3. Legen Sie im FC6 in der Temp-Schnittstelle eine Variable (7) vom Datentyp "UDT1" an.
4. Versorgen Sie die InOut-Schnittstelle des FC5 mit der Variable (7).
5. Übersetzen und laden Sie die beiden Bausteine (FC5 und FC6) in Ihre CPU. Sie brauchen diese Bausteine (FC5 und FC6) nicht im Anwenderprogramm aufzurufen.

Ergebnis:

Der Fehlercode 16#8155 wird nicht mehr ausgegeben, da der Datentyp jetzt dem Anwenderprogramm bekannt ist.

Diese Vorgehensweise ist nicht notwendig, wenn Sie bereits nach dem Aufruf der Anweisung "DB_ANY_TO_VARIANT" eine der beiden Anweisungen "VariantGet" oder "VariantPut" aufrufen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#tempVARIANT := DB_ANY_TO_VARIANT(IN := "InputDB",
                                  ERR := "Tag_Error");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnitt-stelle	Operand	Wert
IN	Input	InputDB	11
<Funktionswert>	Temp	tempVARIANT	-

Die Nummer eines beliebigen Datenbausteins, der am Operand "InputDB" angegeben ist, wird genutzt um eine Variable vom Datentyp VARIANT zu erzeugen, die den Datenbaustein adressiert. Da der Operand am Parameter IN den Datentyp DB_ANY hat, muss zur Programmerstellung noch nicht bekannt sein, welcher Datenbaustein zur Laufzeit verwendet wird, weder der Name noch die Nummer des Datenbausteins. Da der Operand am Parameter RET_VAL den Datentyp VARIANT hat, muss zur Programmerstellung noch nicht bekannt sein, welchen Datentyp der Datenbaustein hat.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Legacy

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P	Eingabewert, der skaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
Funktionswert (RET_VAL)		WORD	E, A, M, D, L, P	Fehlerinformationen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als 27 648 oder kleiner als 0 (unipolar) bzw. -27 648 (bipolar).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweiser der Anweisung:

SCL

```
"Tag_ErrorCode" := SCALE(IN := "Tag_InputValue",
                         HI_LIM := "Tag_HighLimit",
                         LO_LIM := "Tag_LowLimit",
                         BIPOLAR := "Tag_Bipolar",
                         OUT => "Tag_Result");
```

Die Fehlerinformation wird am Operanden "Tag_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
OUT	Tag_Result	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert -27648,0 und die Konstante "K2" den Wert +27648,0.
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert 0,0 und die Konstante "K2" den Wert +27648,0.

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P	Eingabewert, der in einen ganzzahligen Wert deskaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
Funktionswert (RET_VAL)		WORD	E, A, M, D, L, P	Fehlerinformationen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
"Tag_ErrorCode" := UNSCALE(IN := "Tag_InputValue",
                           HI_LIM := "Tag_HighLimit",
                           LO_LIM := "Tag_LowLimit",
                           BIPOLAR := "Tag_Bipolar",
                           OUT => "Tag_Result");
```

Die Fehlerinformation wird am Operanden "Tag_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	22
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

Programmsteuerung

IF: Bedingt ausführen

Beschreibung

Mit der Anweisung "Bedingt ausführen" verzweigen Sie den Programmfluss abhängig von einer Bedingung. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Parameter

Je nach Verzweigungsart können Sie die folgenden Formen der Anweisung programmieren:

- Verzweigung durch IF:
  IF <Bedingung> THEN <Anweisungen>
  END_IF;

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung mit der nächsten Anweisung nach END_IF fortgesetzt.

- Verzweigung durch IF und ELSE:
  IF <Bedingung> THEN <Anweisungen1>
  ELSE <Anweisungen0>
  END_IF;

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, werden die Anweisungen ausgeführt, die nach ELSE programmiert sind. Danach wird die Programmbearbeitung mit der nächsten Anweisung nach END_IF fortgesetzt.

- Verzweigung durch IF, ELSIF und ELSE:
  IF <Bedingung1> THEN <Anweisungen1>
  ELSIF <Bedingung2> THEN <Anweisung2>
  ELSE <Anweisungen0>
  END_IF;

Wenn die erste Bedingung (<Bedingung1>) erfüllt ist, werden die Anweisungen (<Anweisungen1>) nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END_IF fortgesetzt.

Wenn die erste Bedingung nicht erfüllt ist, wird die zweite Bedingung (<Bedingung2>) geprüft. Wenn die zweite Bedingung (<Bedingung2>) erfüllt ist, werden die Anweisungen <Anweisungen2> nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END_IF fortgesetzt.

Wenn keine der Bedingungen erfüllt ist, werden die Anweisungen (<Anweisungen0>) nach ELSE ausgeführt und danach wird die Programmbearbeitung nach END_IF fortgesetzt.

Sie können beliebig viele Kombinationen von ELSIF und THEN innerhalb der IF-Anweisung schachteln. Die Programmierung eines ELSE-Zweigs ist optional.

Die Syntax der IF-Anweisung besteht aus den folgenden Teilen:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der ausgewertet wird.
<Anweisungen>	-		Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn keine Bedingung innerhalb der Programmschleife erfüllt ist.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF "Tag_1" = 1
THEN "Tag_Value" := 10;
ELSIF "Tag_2" = 1
THEN "Tag_Value" := 20;
ELSIF "Tag_3" = 1
THEN "Tag_Value" := 30;
ELSE "Tag_Value" := 0;
END_IF;
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_1	1	0	0	0
Tag_2	0	1	0	0
Tag_3	0	0	1	0
Tag_Value	10	20	30	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CASE: Mehrfach verzweigen

Beschreibung

Mit der Anweisung "Mehrfach verzweigen" bearbeiten Sie abhängig vom Wert eines numerischen Ausdrucks eine von mehreren Anweisungsfolgen.

Der Wert des Ausdrucks muss eine Ganzzahl sein. Bei der Ausführung der Anweisung wird der Wert des Ausdrucks mit den Werten mehrerer Konstanten verglichen. Wenn der Wert des Ausdrucks mit dem Wert einer Konstante übereinstimmt, werden die Anweisungen ausgeführt, die direkt nach dieser Konstante programmiert sind. Die Konstanten können dabei die folgenden Werte annehmen:

- Eine Ganzzahl (z. B. 5)
- Ein Bereich aus Ganzzahlen (z. B. 15..20)
- Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)

Die Anweisung können Sie wie folgt deklarieren:

```
CASE <Variable> OF
<Konstante1>: <Anweisungen1>;
<Konstante2>: <Anweisungen2>;
<KonstanteX>: <AnweisungenX>; // X >= 3
ELSE <Anweisungen0>;
END_CASE;
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Variable>	Ganzzahlen	E, A, M, D, L	Wert, der mit den programmierten Konstantenwerten verglichen wird.
<Konstante>	Ganzzahlen	-	Konstantenwerte, die die Bedingung zur Ausführung einer Anweisungsfolge bilden. Die Konstanten können die folgenden Werte annehmen: • Eine Ganzzahl (z. B. 5) • Ein Bereich aus Ganzzahlen (z. B. 15..20) • Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)
<Anweisung>	-	-	Beliebige Anweisungen, die ausgeführt werden, wenn der Wert des Ausdrucks mit dem Wert einer Konstanten übereinstimmt. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn die Werte nicht übereinstimmen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) übereinstimmt, werden die Anweisungen (<Anweisungen1>) ausgeführt, die direkt nach der ersten Konstante programmiert sind. Danach wird die Programmbearbeitung nach END_CASE fortgesetzt.

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) nicht übereinstimmt, wird dieser mit dem Wert der Konstante verglichen, die als nächste programmiert ist. Auf diese Weise wird die CASE-Anweisung ausgeführt, bis eine Übereinstimmung der Werte vorliegt. Wenn der Wert des Ausdrucks keinem der programmierten Konstantenwerte entspricht, werden die Anweisungen (<Anweisungen0>) ausgeführt, die nach ELSE programmiert sind. ELSE ist ein optionaler Teil der Syntax und kann entfallen.

Die CASE-Anweisung kann auch geschachtelt werden, indem ein Anweisungsblock durch CASE ersetzt wird. END_CASE bildet den Abschluss der CASE-Anweisung.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
CASE "Tag_Value" OF
     0 :
          "Tag_1" := 1;
     1,3,5 :
          "Tag_2" := 1;
     6..10 :
          "Tag_3" := 1;
     16,17,20..25 :
          "Tag_4" := 1;
     ELSE
          "Tag_5" := 1;
END_CASE;
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte				
Tag_Value	0	1, 3 , 5	6, 7, 8, 9, 10	16,17, 20, 21, 22, 23, 24, 25	2
Tag_1	1	-	-	-	-
Tag_2	-	1	-	-	-
Tag_3	-	-	1	-	-
Tag_4	-	-	-	1	-
Tag_5	-	-	-	-	1
1: Der Operand wird auf den Signalzustand "1" gesetzt. -: Der Signalzustand des Operanden bleibt unverändert.					

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

FOR: In Zählschleife ausführen

Beschreibung

Mit der Anweisung "In Zählschleife ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Laufvariable innerhalb eines angegebenen Wertebereichs liegt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung

"Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Hinweis

Informationen zur Durchlaufanzahl und zur Laufvariable

Die Durchlaufanzahl kann während der Programmbearbeitung nicht verändert werden.

Aus Performance Gründen sollte die Laufvariable in der Bausteinschnittstelle im Abschnitt "Temp" deklariert werden. Darüberhinaus ist es nicht möglich, die Laufvariable aus der Schleife heraus zu ändern.

Das folgende Programmbeispiel würde demnach einen Syntaxfehler erzeugen und wäre nicht übersetzbar:

```
FOR #i := 1 TO 10 DO
#i := #i + 1;
END_FOR;
```

Grenzwerte für FOR-Anweisungen

Um "sichere" FOR-Anweisungen, die nicht endlos laufen, zu programmieren, beachten Sie folgende Regel und Grenzwerte:

FOR <Laufvariable> := <Anfangswert> TO <Endwert> BY <Schrittweite> DO <Anweisungen>;

END_FOR;

Wenn...	...dann	Bemerkung
Anfangswert < Endwert	Endwert < (PMAX - schritt)	Laufvariable läuft in positiver Richtung
Anfangswert > Endwert AND Schrittweite < O	Endwert > (NMAX - schritt)	Laufvariable läuft in negativer Richtung

Grenzwerte

Für die möglichen Datentypen gelten unterschiedliche Grenzwerte:

Datentyp	PMAX	NMAX
Laufvariable vom Typ SINT	127	-128
Laufvariable vom Typ INT	32767	-32768
Laufvariable vom Typ DINT	2147483647	-2147483648
Laufvariable vom Typ LINT	9223372036854775807	-9223372036854775808

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp		Speicherbereich	Beschreibung
	S7-1200	S7-1500		
<Laufvariable>	SINT, INT, DINT	SINT, INT, DINT, LINT	E, A, M, D, L	Operand, dessen Wert bei der Schleifenbearbeitung ausgewertet wird. Der Datentyp der Laufvariablen bestimmt den Datentyp der restlichen Parameter.
<Anfangswert>	SINT, INT, DINT	SINT, INT, DINT, LINT	E, A, M, D, L	Ausdruck, dessen Wert bei Beginn der Schleifenbearbeitung der Laufvariablen zugewiesen wird.
<Endwert>	SINT, INT, DINT	SINT, INT, DINT, LINT	E, A, M, D, L	Ausdruck, dessen Wert den letzten Durchlauf der Programmschleife bestimmt. Vor jedem Schleifendurchlauf wird überprüft, welchen Wert die Laufvariable hat: • Endwert nicht erreicht: Die Anweisungen nach DO werden ausgeführt • Endwert erreicht: Die FOR-Schleife wird ein letztes Mal ausgeführt • Endwert überschritten: Die FOR-Schleife wird beendet Eine Veränderung des Endwerts während der Ausführung der Anweisung ist nicht zulässig.
<Schrittweite>	SINT, INT, DINT	SINT, INT, DINT, LINT	E, A, M, D, L	Ausdruck, um dessen Wert die Laufvariable nach jedem Schleifendurchlauf erhöht (positive Schrittweite) oder verringert (negative Schrittweite) wird. Die Angabe der Schrittweite ist optional. Wenn keine Schrittweite angegeben ist, wird der Wert der Laufvariablen nach jedem Schleifendurchlauf um 1 erhöht. Eine Veränderung der Schrittweite während der Ausführung der Anweisung ist nicht zulässig.
<Anweisungen>	-	-		Anweisungen, die bei jedem Schleifendurchlauf ausgeführt werden, solange der Wert der Laufvariablen im Wertebereich liegt. Der Wertebereich wird durch den Anfangs- und Endwert bestimmt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i := 2 TO 8 BY 2
    DO "a_array[i] := "Tag_Value"*"b_array[i]";
END_FOR;
```

Der Operand "Tag_Value" wird mit den Elementen (2, 4, 6, 8) der ARRAY-Variablen "b_array" multipliziert. Das Ergebnis wird in den Elementen (2, 4, 6, 8) der ARRAY-Variablen "a_array" eingelesen.

Siehe auch

CONTINUE: Schleifenbedingung erneut prüfen (Seite 1918)

EXIT: Schleife sofort verlassen (Seite 1918)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

WHILE: Bei erfüllter Bedingung ausführen

Beschreibung

Mit der Anweisung "Bei erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Durchführungsbedingung erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung können Sie wie folgt deklarieren:

WHILE <Bedingung> DO <Anweisungen>;

END_WHILE;

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der vor jedem Schleifendurchlauf ausgewertet wird.
<Anweisungen>	-		Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung nach END_WHILE fortgesetzt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
WHILE
      "Tag_Value1" <> "Tag_Value2"
      DO "Tag_Result"
      := "Tag_Input";
END_WHILE;
```

Solange die Werte der Operanden "Tag_Value1" und "Tag_Value2" ungleich sind, wird dem Operanden "Tag_Result" der Wert des Operanden "Tag_Input" zugewiesen.

Siehe auch

EXIT: Schleife sofort verlassen (Seite 1918)

CONTINUE: Schleifenbedingung erneut prüfen (Seite 1918)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

REPEAT: Bei nicht erfüllter Bedingung ausführen

Beschreibung

Mit der Anweisung "Bei nicht erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Abbruchbedingung nicht erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung können Sie wie folgt deklarieren:

REPEAT <Anweisungen>;

UNTIL <Bedingung> END_REPEAT;

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Anweisungen>	-		Anweisungen, die solange ausgeführt werden, wie die programmierte Bedingung den Wert FALSE hat. Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der nach jedem Schleifendurchlauf ausgewertet wird. Wenn der Ausdruck den Wert FALSE hat, wird die Programmschleife erneut bearbeitet. Wenn der Ausdruck den Wert TRUE hat, wird die Programmbearbeitung nach END_REPEAT fortgesetzt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
REPEAT "Tag_Result"
        := "Tag_Value";
UNTIL "Tag_Error"
END_REPEAT;
```

Solange der Wert des Operanden "Tag_Error" den Signalzustand "0" hat, wird dem Operanden "Tag_Result" der Wert des Operanden "Tag_Value" zugewiesen.

Siehe auch

CONTINUE: Schleifenbedingung erneut prüfen (Seite 1918)

EXIT: Schleife sofort verlassen (Seite 1918)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CONTINUE: Schleifenbedingung erneut prüfen

Beschreibung

Mit der Anweisung "Schleifenbedingung erneut prüfen" beenden Sie den aktuellen Programmdurchlauf einer FOR-, WHILE- oder REPEAT-Schleife.

Nach der Ausführung der Anweisung werden die Bedingungen zur Fortsetzung der Programmschleife erneut ausgewertet. Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i
     := 1 TO 15 BY 2 DO
     IF (i < 5) THEN
     CONTINUE;
END_IF;
      "DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung i < 5 erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) nicht bearbeitet. Die Laufvariable (i) wird um die Schrittweite "2" erhöht und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.

Wenn die Bedingung i < 5 nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable wird ebenfalls um die Schrittweite "2" erhöht und geprüft.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

EXIT: Schleife sofort verlassen

Beschreibung

Mit der Anweisung "Schleife sofort verlassen" brechen Sie die Bearbeitung einer FOR-, WHILE- oder REPEAT-Schleife, unabhängig von Bedingungen, an beliebiger Stelle ab. Die Programmbearbeitung wird nach dem Abschluss der Schleife (END_FOR, END_WHILE, END_REPEAT) fortgesetzt.

Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i := 15 TO 1 BY -2 DO
IF (i < 5)
THEN EXIT;
END_IF;
"DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung i < 5 erfüllt ist, wird die Bearbeitung der Schleife abgebrochen. Die Programmbearbeitung wird nach END_FOR fortgesetzt.

Wenn die Bedingung i < 5 nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] :=1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable (i) wird um die Schrittweite "-2" verringert und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable (i) im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

GOTO: Springen

Beschreibung

Mit der Anweisung "Springen" setzen Sie die Programmbearbeitung an einer Stelle fort, die durch eine Sprungmarke gekennzeichnet ist.

Die Sprungmarke und die Anweisung "Springen" müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb eines Bausteins nur einmal vergeben werden. Jede Sprungmarke kann von mehreren Sprunganweisungen angesprungen werden.

Ein Sprung von "außen" in eine Programmschleife ist nicht zulässig, aber ein Sprung von einer Programmschleife nach "außen" ist möglich.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Die Anweisung können Sie wie folgt deklarieren:

GOTO <Sprungmarke>;

...

<Sprungmarke>: <Anweisungen>

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Beschreibung
<Sprungmarke>	-	Sprungmarke, zu der gesprungen wird
<Anweisungen>	-	Anweisungen, die nach dem Sprung ausgeführt werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
CASE "Tag_Value" OF
1 : GOTO MyLABEL1;
2 : GOTO MyLABEL2;
3 : GOTO MyLABEL3;
ELSE GOTO MyLABEL4;
END_CASE;
MyLABEL1: "Tag_1" := 1;
MyLABEL2: "Tag_2" := 1;
MyLABEL3: "Tag_3" := 1;
MyLABEL4: "Tag_4" := 1;
```

Abhängig vom Wert des Operanden "Tag_Value" wird die Programmbearbeitung an der Stelle fortgesetzt, die durch die entsprechende Sprungmarke gekennzeichnet ist. Wenn z. B. der Operand "Tag_Value" den Wert "2" hat, wird die Programmbearbeitung bei der Sprungmarke "MyLABEL2" fortgesetzt. Die Programmzeile, die durch die Sprungmarke "MyLABEL1" gekennzeichnet ist, wird in diesem Fall übersprungen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RETURN: Baustein verlassen

Beschreibung

Mit der Anweisung "Baustein verlassen" beenden Sie die Programmbearbeitung im aktuell bearbeiteten Baustein und setzen sie im aufrufenden Baustein fort.

Am Bausteinende kann die Anweisung entfallen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL
```
IF "Tag_Error" <>0 THEN RETURN;
END_IF;
```

Wenn der Signalzustand des Operanden "Tag_Error" ungleich Null ist, wird die Programmbearbeitung im aktuell bearbeiteten Baustein beendet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

(*...*): Kommentarabschnitt einfügen

Beschreibung

Mit der Anweisung "Kommentarabschnitt einfügen" fügen Sie einen Kommentarabschnitt ein. Der Text, der innerhalb der Klammern (*...*) steht, wird wie ein Kommentar behandelt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL
```
(*Dies ist ein Kommentarabschnitt.*)
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

REGION: Programmcode strukturieren

Beschreibung

Mit der Anweisung "Programmcode strukturieren" können Sie den Programmcode innerhalb eines SCL-Bausteins strukturieren, indem Sie ihn in unterschiedliche Bereiche einteilen.

Weitere Informationen zum Arbeiten mit Bereichen finden Sie hier: Arbeiten mit Bereichen (Seite 7646)

Die Anweisung können Sie wie folgt deklarieren:

REGION <Name>

<Anweisungen>

END_REGION

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
REGION	-	-	Schlüsselwort, mit dem der Bereich beginnt
<Name>	-	-	Text zur Bezeichnung der REGION
<Anweisungen>	-	-	Programmcode, den die REGION umschließt
END_REGION	-	-	Schlüsselwort, mit dem der Bereich abschließt

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
REGION Feeder System
     // Strukturiert den Quellcode für das Feeder-System
   IF "Variable_1" = 0 THEN
      "Variable_2" := 10;
   END_IF;
END_REGION
```

Der Bereich Feeder System umfasst den gesamten Programmcode.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Laufzeitsteuerung

ENDIS_PW: Passwort-Legitimierung einschränken und freigeben

Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU projektierte Passwörter legitimiert sind oder nicht. Dadurch können Sie legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist.

Wenn Sie die Anweisung aufrufen und der Parameter REQ den Signalzustand "0" hat, wird der aktuell eingestellte Zustand an den Ausgangsparametern angezeigt. Falls Änderungen an den Eingangsparametern vorgenommen wurden, werden diese nicht an die Ausgangsparameter übergeben.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F_PWD, FULL_PWD, R_PWD, HMI_PWD) übernommen:

- Wenn der Signalzustand "0" vorliegt, ist die Legitimation per Passwort nicht erlaubt.
- Wenn der Signalzustand "1" vorliegt, ist das Passwort verwendbar.

Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer das Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Nicht konfigurierte Passwörter müssen am Eingang den Signalzustand TRUE haben und liefern den Signalzustand TRUE am Ausgang. Das Failsafe Passwort ist nur für eine F-CPU parametrierbar und muss daher in einer Standard CPU immer mit dem Signalzustand TRUE verschaltet werden. Wenn die Anweisung einen Fehler zurückliefert, dann bleibt der Aufruf ohne Auswirkung, d. h. die vorherige Sperre ist unverändert wirksam.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken einer leeren Speicherkarte (Übertragungskarte oder Programmkarte) in eine S7-1200-CPU.
- Der Übergang von NETZ AUS - NETZ EIN deaktiviert in der S7-1200-CPU den Schutz. Die Anweisung "Passwort-Legitimierung einschränken und freigeben" muss im Programm erneut (z. B. im Anlauf-OB) aufgerufen werden.

Hinweis

Wenn das HMI Passwort nicht freigegeben ist, dann sperrt die Anweisung "Passwort-Legitimierung einschränken und freigeben" den Zugriff von HMI-Systemen.

Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

Ungewolltes Aussperren bei einer S7-1500-CPU verhindern

Die Einstellungen können im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Ungewolltes Aussperren bei einer S7-1200-CPU verhindern

Da es bei einer S7-1200-CPU keinen Betriebsartenschalter gibt, wird der Schutz bei NETZ AUS - NETZ EIN deaktiviert. Damit ist es möglich und empfehlenswert, ein ungewolltes Aussperren mithilfe bestimmter Programmsequenzen innerhalb Ihres Programms zu verhindern.

Programmieren Sie dazu eine Zeitsteuerung entweder über einen Weckalarm-OB oder einen Timer im Main OB (OB 1). Dadurch haben Sie die Möglichkeit, nach einem Übergang von NETZ AUS - NETZ EIN und der damit verbundenen Deaktivierung des Schutzes, die Anweisung "Passwort-Legitimierung einschränken und freigeben" in dem entsprechenden OB (z. B. OB 1 oder OB 35) zeitnah wieder aufzurufen. Um das Zeitfenster, in dem die Anweisung nicht aktiv ist und damit keine Einschränkungen bei der Passwort-Legitimierung bestehen, möglichst klein zu halten, rufen Sie die Anweisung im Anlauf-OB (OB 100) auf. Diese Vorgehensweise bietet Ihnen den größtmöglichen Schutz vor unbefugtem Zugriff.

Wenn es zu einer ungewollten Aussperrung gekommen ist, dann können Sie den Aufruf im Anlauf-OB überspringen (z. B. durch die Abfrage eines Eingangsparameters) und haben die eingestellte Zeit (z. B. 10 Sekunden bis 1 Minute), um eine Verbindung zur CPU aufzubauen, bevor die Sperre wieder aktiv ist.

Wenn Sie in Ihrem Programmcode keinen Timer vorgesehen haben und es zu einer Aussperrung kam, dann stecken Sie eine leere Übertragungskarte oder eine leere Programmkarte in die CPU. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Danach müssen Sie das Anwenderprogramm aus STEP 7 erneut in die CPU laden.

Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte oder Programmkarte. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

WARNUNG

Leere Übertragungskarte stecken

Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Nach dem Ziehen der Übertragungskarte ist der Inhalt der Übertragungskarte im internen Ladespeicher. Achten Sie an dieser Stelle darauf, dass die Karte kein Programm enthält.

WARNUNG

Leere Programmkarte stecken

Wenn Sie eine Programmkarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Achten Sie darauf, dass die Programmkarte leer ist. Der interne Ladespeicher wird auf die leere Programmkarte kopiert. Nach dem Ziehen der ehemals leeren Programmkarte ist der interne Ladespeicher leer.

Sie müssen die Übertragungskarte oder Programmkarte ziehen, bevor Sie die CPU in RUN versetzen.

Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

Aktion	Passwortschutz durch die Anweisung
Grundzustand nach • Betriebsartenschalter auf STOP • Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control) • Zurücksetzen auf Werkseinstellung	Nicht aktiviert (Keine Einschränkungen)
Grundzustand nach NETZ-EIN	• S7-1200-CPU: Die Sperre ist deaktiviert und die Anweisung muss im Programm erneut aufgerufen werden (z. B. im Anlauf-OB). • S7-1500-CPU: Aktiviert (wenn vor NETZ AUS eine Sperre aktiviert war). Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent.
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT	Aktiviert Die Passwörter dürfen weiterhin nicht verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.
F_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff inklusive Failsafe • F_PWD = "0": Passwort nicht erlauben • F_PWD = "1": Passwort erlauben
FULL_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff • FULL_PWD = "0": Passwort nicht erlauben • FULL_PWD = "1": Passwort erlauben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lesezugriff • R_PWD = "0": Passwort nicht erlauben • R_PWD = "1": Passwort erlauben
HMI_PWD	Input	BOOL	E, A, M, D, L oder Konstante	HMI-Zugriff • HMI_PWD = "0": Passwort nicht erlauben • HMI_PWD = "1": Passwort erlauben
F_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff inklusive Failsafe • F_PWD_ON = "0": Passwort nicht erlaubt • F_PWD_ON = "1": Passwort erlaubt
FULL_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff • FULL_PWD_ON = "0": Passwort nicht erlaubt • FULL_PWD_ON = "1": Passwort erlaubt
R_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lesezugriff • R_PWD_ON = "0": Passwort nicht erlaubt • R_PWD_ON = "1": Passwort erlaubt
HMI_PWD_ON	Output	BOOL	E, A, M, D, L	Status HMI-Zugriff • HMI_PWD_ON = "0": Passwort nicht erlaubt • HMI_PWD_ON = "1": Passwort erlaubt
Funktionswert (RET_VAL)		WORD	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8090	Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt

Fehler-code* (W#16#...)	Erläuterung
80D0	Das Passwort für Failsafe ist nicht konfiguriert. Bei Standard-CPUs muss der Signalzustand TRUE sein.
80D1	Der Lese-/Schreibzugriff ist nicht konfiguriert
80D2	Der Lesezugriff ist nicht konfiguriert
80D3	Der HMI-Zugriff ist nicht konfiguriert
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ENDIS_PW(REQ := 0,
                         F_PWD := 0,
                         FULL_PWD := 0,
                         R_PWD := 1,
                         HMI_PWD := 0,
                         F_PWD_ON => "Status_F_PWD",
                         FULL_PWD_ON => "Status_FULL_PWD",
                         R_PWD_ON => "Status_R_PWD",
                         HMI_PWD_ON => "Status_HMI_PWD");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Datentyp	Wert
REQ	BOOL	0
F_PWD	BOOL	0
FULL_PWD	BOOL	0
R_PWD	BOOL	1
HMI_PWD	BOOL	0
Status_F_PWD	BOOL	0
Status_FULL_PWD	BOOL	0
Status_R_PWD	BOOL	1
Status_HMI_PWD	BOOL	0

Die Anweisung wird ausgeführt, da der Operand REQ den Signalzustand "1" hat. Der Operand R_PWD hat den Signalzustand "1", das bedeutet, dass bei Eingabe des vergebenen Passworts ein Lesezugriff erlaubt ist. Der Statusoperand R_PWD_ON hat ebenfalls den Signalzustand "1" und signalisiert damit, dass der Operand R_PWD aktiviert ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

PC-Systeme

SHUT_DOWN: Zielsystem herunterfahren

Beschreibung

Mit der Anweisung “SHUT_DOWN: Zielsystem herunterfahren” fahren Sie das PC-basierte Automatisierungssystem herunter, starten den S7-Software Controller CPU 150xS oder Windows auf dem PC-basierte Automatisierungssystem neu.

Die Anweisung finden Sie in der Task Card "Anweisungen" unter Einfache Anweisungen > Programmsteuerung > Laufzeitsteuerung.

Ein Neustart ist z. B. in den folgenden Situationen sinnvoll:

- Eine Industrie-USV (unterbrechungsfreie Stromversorgung) meldet über einen digitalen Eingang einen Spannungsausfall.
- Windows reagiert nicht mehr oder liefert einen „Blue Screen“.
- Im Anwenderprogramm werden zu viele Fehler-OBs aufgerufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zielsystem herunter fahren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	MODE = 1: CPU 150xS und Windows herunter fahren Die CPU geht in STOP und speichert die remanenten Daten. Anschließend werden CPU und Windows herunter gefahren. Das System muss manuell gestartet werden. MODE = 2: CPU 150xS neu starten Die CPU geht in STOP und speichert die remanenten Daten. Anschließend wird die CPU heruntergefahren und neu gestartet. MODE = 3: Windows neu starten. Die CPU bleibt in RUN. Windows wird neu gestartet (ab TIA Portal V14 ist MODE 3 nur für Abwärtskompatibilität gelassen. Es wird empfohlen, MODE 4 oder MODE 5 zu verwenden). MODE = 4: Windows wird korrekt heruntergefahren und neu gestartet. Die CPU bleibt im RUN. MODE = 5: Windows neu starten. (Vergleichbar mit MODE 3; Ausnahme: MODE 5 ist nur dann zu verwenden, wenn Windows abgestürzt ist).

COM-MENT	Input	STRING	E, A, M, D, L	Bei Mode = 1, 3 und 4 können Sie eine Begründung für den Neustart mitgeben. Die Begründung wird im Windows Event Log ausgegeben.
Ret_Val	Return	WORD	A, M, D, L	Ret_Val = 0: kein Fehler Ret_Val = 8090: Der übergebene Wert an MODE wird nicht unterstützt. Ret_Val = 8091: Windows reagiert nicht auf den Aufruf der Anweisung Shut_Down (gültig nur für Mode 3 und 4). Ret_Val = 8092: Bei Auftreten dieses Fehlers, wenden Sie sich bitte an den SIMATIC Customer Support (gültig nur für Mode 3 und 4).

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Aufruf der Anweisung

Es gelten die folgenden Aufrufbedingungen:

- Für eine CPU der Baureihe S7-1200 gilt:
  In Firmware Versionen < 2.2 können Sie die Anweisung nur in einem Zyklus-Organisationsbaustein der Priorität 1 aufrufen. Das entspricht der niedrigsten Priorität aller Organisationsbausteine. Wenn Sie die Anweisung in einem Organisationsbaustein mit höherer Priorität aufrufen, dann wird sie nicht ausgeführt und das Ergebnis (Bie-Bit, Freigabeausgang ENO) ist immer "0".
  In Firmware Versionen >= 2.2 können Sie die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.
- Für eine CPU der Baureihe S7-1500 gilt:
  Sie können die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

GET_ERROR: Fehler lokal abfragen

Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Programmbausteins ab. Dabei handelt es sich in der Regel um Programmier- oder Zugriffsfehler. Wenn das System während der Programmbausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler, innerhalb der Ausführung des Programmbausteins seit der letzten Ausführung der Anweisung, eine ausführliche Information am <Operand> ausgegeben.

Die Fehlerinformation kann nur in Operanden vom Systemdatentyp "ErrorStruct" gespeichert werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Treten im Programmbaustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehlerinformation zum nächsten aufgetretenen Fehler von der Anweisung ausgegeben.

Hinweis

<Operand>

Der <Operand> wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Operanden nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie den Operanden im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie den Operanden vor dem Aufruf der Anweisung auf "0" zurück.

Hinweis

Aktivierung der lokalen Fehlerbehandlung

Sobald Sie die Anweisung in den Programmcode eines Programmbausteins einfügen, wird die lokale Fehlerbehandlung aktiviert und voreingestellte Systemreaktionen werden beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Operand>	ErrorStruct	D, L	Information zu aufgetretenen Fehlern

Datentyp "ErrorStruct"

Der Datentyp "ErrorStruct" kann in einen globalen Datenbaustein oder in die Bausteinschnittstelle eingefügt werden. Sie können den Datentyp auch mehrfach einfügen, wenn Sie jedes Mal einen anderen Namen für die Datenstruktur vergeben. Die Datenstruktur und der Name einzelner Strukturelemente sind nicht änderbar. Wenn Sie die Fehlerinformation in einem globalen Datenbaustein speichern, kann sie auch von anderen Programmbausteinen ausgelesen werden.

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

Strukturkomponente		Datentyp	Beschreibung
ERROR_ID		WORD	Fehler-ID
FLAGS		BYTE	Zeigt, ob der Fehler während eines Programmbausteinaufrufs aufgetreten ist. 16#01: Fehler während eines Programmbausteinaufrufs 16#00: Kein Fehler während eines Programmbausteinaufrufs
REACTION		BYTE	Voreingestellte Reaktion: 0: Ignorieren (Schreibfehler) 1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler) 2: Anweisung überspringen (Systemfehler)
CODE_ADDRESS		CREF	Informationen zur Adresse und Art des Programmbausteins
	BLOCK_TYPE	BYTE	Programmbausteintyp, in dem der Fehler aufgetreten ist: 1: Organisationsbaustein (OB) 2: Funktion (FC) 3: Funktionsbaustein (FB)
	CB_NUMBER	UINT	Nummer des Codebausteins
	OFFSET	UDINT	Verweis auf den internen Speicher

Strukturkomponente		Datentyp	Beschreibung
MODE		BYTE	Informationen zur Adresse eines Operanden
OPERAND_NUMBER		UINT	Operandennummer des Maschinenbefehls
POINTER_NUMBER_LOCATION		UINT	(A) Interner Zeiger
SLOT_NUMBER_SCOPE		UINT	(B) Ablagebereich im internen Speicher
DATA_ADDRESS		NREF	Informationen zur Adresse eines Operanden
	AREA	BYTE	(C) Speicherbereich: L: 16#40...16#7F, 16#86, 16#87, 16#8E, 16#8F, 16#C0...16#FF E: 16#81 A: 16#82 M: 16#83 DB: 16#40, 16#84, 16#85, 16#8A, 16#8B PE: 16#01 PA: 16#02 Technologische Objekte: 16#04
	DB_NUMBER	UINT	(D) Nummer des Datenbausteins
	OFFSET	UDINT	(E) Relative Adresse des Operanden

Strukturkomponente "ERROR_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR_ID" ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
LABEL: #TagOut := #Field[#index] * REAL#40.5;

GET_ERROR(#Error);

IF #Error.REACTION = 1 THEN
   GOTO LABEL;
   ;
END_IF;
```

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Operand #TagOut liefert trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler lokal abfragen" nach der Multiplikation zu programieren um den Fehler abzufangen. Die Fehlerinformation, die die Anweisung "Fehler lokal abfragen" liefert, wird mithilfe eines Vergleichs ausgewertet. Wenn die Strukturkomponente #Error.REACTION den Wert "1" hat, dann handelt es sich um einen Lese-/Zugriffsfehler und die Programmbearbeitung beginnt wieder bei der Sprungmarke LABEL.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu SCL (Seite 7607)

GET_ERR_ID: Fehler-ID lokal abfragen

Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Dabei handelt es sich in der Regel um Zugriffsfehler. Wenn das System während der Bausteinbearbeitung Fehler innerhalb der Ausführung des Bausteins seit der letzten Ausführung der Anweisung meldet, gibt die Anweisung die Fehler-ID des ersten aufgetretenen Fehlers aus.

Die Fehler-ID kann nur in Operanden vom Datentyp WORD gespeichert werden. Treten im Baustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehler-ID des nächsten aufgetretenen Fehlers von der Anweisung ausgegeben.

Hinweis

Der <Operand> wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Operanden nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie den Operanden im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie den Operanden vor dem Aufruf der Anweisung auf "0" zurück.

Der Ausgang der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn eine Fehlerinformation vorliegt. Wenn diese Bedingung nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Ein Beispiel, wie Sie die Anweisung auch in Kombination mit weiteren Fehlerbehandlungsmöglichkeiten realisieren können, finden Sie unter "Siehe auch".

Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
Funktions-wert	WORD	D, L	Fehler-ID

Fehler-ID

Die folgende Tabelle zeigt die Werte, die ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
#TagOut := #Field[#index] * REAL#40.5;

#TagID := GET_ERR_ID();

IF #TagID = 16#2522 THEN
   MOVE_BLK(IN := #TagArrayIn[0],
            COUNT := 1,
            OUT => #TagArrayOut[1]);
END_IF;
```

Beim Zugriff auf die Variable #Field[#index] ist ein Fehler aufgetreten. Der Operand #TagOut liefert trotz des Lese-/Zugriffsfehlers den Signalzustand "1" und die Multiplikation wird mit dem Wert "0.0" ausgeführt. In diesem Fehlerfall ist es empfehlenswert, die Anweisung "Fehler-ID lokal abfragen" nach der Multiplikation zu programieren um den Fehler abzufangen. Die Fehler-ID, die die Anweisung "Fehler-ID lokal abfragen" liefert, wird mithilfe eines Vergleichs ausgewertet. Wenn der Operand #TagID die ID 16#2522 liefert, dann handelt es sich um einen Lese-/Zugriffsfehler und der Wert "100.0" des Operand #TagArrayIn[0] wird in den Operand #TagArrayOut[1] geschrieben.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu SCL (Seite 7607)

INIT_RD: Alle remanenten Daten zurücksetzen

Beschreibung

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L	Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B5	Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := INIT_RD("Tag_REQ");
```

Wenn der Operanden "Tag_REQ" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Beispiel zum Einfluss der geplanten Verzögerungszeit

Im folgenden Beispiel sehen Sie den Einfluss der Verzögerungszeit der Anweisung "WAIT" in verschiedenen Szenarien.

Das folgende Bild stellt die Szenarien schematisch dar:

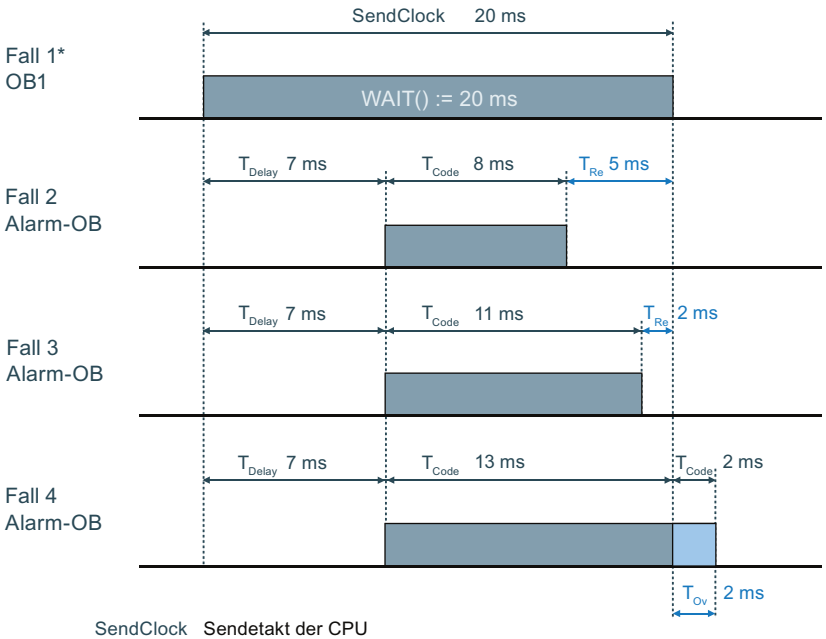

SendClock Sendetakt der CPU

T_{Delay} Verzögerungszeit des Alarm-OBs

T_{Code} Zeit der Programmausführung

T_{Re} Verbleibende Zeit

T_{Ov} Zeit der Überlänge, um welche sich der Sendetakt erhöht

* Der OB1 mit der Anweisung ”WAIT” ist ebenso Teil von Fall 2, 3 und 4 und muss mit berücksichtigt werden.

Verbleibende Zeit = Die Spanne vom Ende der geplanten Verzögerungszeit (via "WAIT") bis zur Beendigung des Alarm-OBs

Zeit der Überlänge = Die Spanne von der Beendigung des Alarm-OBs bis zum Ende der geplanten Verzögerungszeit (via "WAIT").

Fall 1:

Die Anweisung "WAIT" wird in einem OB1 aufgerufen. Die Anweisung "WAIT" kann von höher prioren OBs oder höher prioren Prozessen (z.B. System Threads) unterbrochen werden. Die Verzögerungszeit der Anweisung "WAIT" ist aber weder verändert noch aufgeschoben.

Fall 2 und Fall 3:

Die Verarbeitung des Programms im OB1 wird nach einer Zeitverzögerung von 20 ms fortgesetzt. Diese Zeitverzögerung wird durch den Aufruf der Anweisung "WAIT" im OB1 berechnet (siehe OB1 mit WAIT). Innerhalb dieser 20 ms kann ein Alarm-OB seinen eigenen Programmcode ausführen. Der Sendetakt der CPU wird nicht verändert.

Fall 4:

Die Verarbeitung des Programms im OB1 wird nach Beendingung des höher prioren Prozesses fortgesetzt. Die 20 ms Verzögerungszeit im OB1 sind verstrichen, aber der höher priore Prozess muss noch beendet werden. Der Sendetakt der CPU wird erhöht.

Hinweis

Ausführungsreihenfolge von System- oder Kommunikationsprozessen (System Threads)

Die System Threads verwenden meistens die Priorität "15". Ebenso gibt es System Threads mit einer höheren Priorität als "26", doch diese Prozesse haben eine geringere CPU-Auslastung. System Threads werden im Bild nicht dargestellt.

Laufzeitmessung des OB1 mit Hilfe der Anweisung "RT_INFO":

Fall 2: 20 ms - 8 ms - System Threads = <12 ms. Sendetakt: ~20 ms.

Fall 3: 20 ms - 11 ms - System Threads - <9 ms. Sendetakt: ~20 ms.

Fall 4: 20 ms - 15 ms - System Threads - <7 ms. Sendetakt: ~22 ms.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RUNTIME: Laufzeitmessung

Beschreibung

Mit der Anweisung "Laufzeitmessung" messen Sie die Laufzeit des gesamten Programms, einzelner Bausteine oder Befehlssequenzen.

Wenn Sie die Laufzeit Ihres gesamten Programms messen wollen, dann rufen Sie die Anweisung "Laufzeitmessung" im OB1 auf. Der Startpunkt der Laufzeitmessung wird mit dem ersten Aufruf gesetzt und der Ausgang RET_VAL liefert nach dem zweiten Aufruf die Laufzeit des Programms. Die gemessene Laufzeit enthält alle möglichen Vorgänge der CPU, die innerhalb des Programmablaufs statt gefunden haben, wie z. B. Unterbrechungen durch höher priore Ereignisse oder Kommunikation. Die Anweisung "Laufzeitmessung" liest einen internen Zähler der CPU aus und schreibt dessen Wert in den Durchgangsparameter. Die Anweisung berechnet die aktuelle Laufzeit des Programms gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Wenn Sie die Laufzeit einzelner Bausteine oder einzelner Befehlssequenzen messen wollen, dann benötigen Sie drei separate Netzwerke. Rufen Sie die Anweisung "Laufzeitmessung" in einem einzelnen Netzwerk innerhalb Ihres Programms auf. Mit diesem ersten Aufruf der Anweisung setzen Sie den Startpunkt der Laufzeitmessung. Anschließend rufen Sie im nächsten Netzwerk den gewünschten Programmbaustein oder die Befehlssequenz auf. Rufen Sie dann in einem weiteren Netzwerk die Anweisung "Laufzeitmessung" zum zweiten Mal auf und weisen Sie dem Durchgangsparameter den gleichen Speicher wie beim ersten Aufruf der Anweisung zu. Die Anweisung "Laufzeitmessung" im dritten Netzwerk liest einen internen Zähler der CPU aus und berechnet die aktuelle Laufzeit des Programmbausteins oder der

Befehlssequenz gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET_VAL.

Die Anweisung "Laufzeitmessung" verwendet einen internen hochfrequenten Zähler um die Zeit zu berechnen. Wenn der Zähler überläuft, gibt die Anweisung Werte <= 0.0 zurück. Dies kann für S7-1200-CPUs mit Firmware Stand <V4.2 bis zu einmal pro Minute vorkommen. Diese Runtimewerte sind zu ignorieren.

Hinweis

Da beim optimierten Übersetzen des Programms die Reihenfolge der Anweisungen innerhalb einer Befehlssequenz geändert wird, kann die Laufzeit einer Befehlssequenz nicht genau bestimmt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	InOut	LREAL	E, A, M, D, L	Der Inhalt ist nur für interne Zwecke vorgesehen.
Funktionswert		LREAL	E, A, M, D, L	Liefert die gemessene Laufzeit in Sekunden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung anhand einer Berechnung der Laufzeit eines Programmbausteins:

SCL

```
"Tag_Result" := RUNTIME("Tag_Memory");

"Best_before_date_DB" ();

"Tag_Result" := RUNTIME("Tag_Memory");
```

Mit dem ersten Aufruf der Anweisung wird der Startpunkt für die Laufzeitmessung gesetzt und als Referenz für den zweiten Aufruf der Anweisung im Operanden "TagMemory" zwischengespeichert.

Im Anschluss wird der Programmbaustein FB1 "Best_before_date" aufgerufen.

Wenn der Programmbaustein FB1 abgearbeitet ist, wird die Anweisung zum zweiten Mal ausgeführt. Der zweite Aufruf der Anweisung berechnet die Laufzeit des Programmbausteins und schreibt das Ergebnis in den Ausgang "Tag_Result".

Ein ausführlicheres Beispiel, wie Sie die Gesamtzykluszeit eines Programms messen können, finden Sie im Siemens Industry Online Support unter folgender Beitrags-ID: 87668055 (https://support.industry.siemens.com/cs/ww/de/view/87668055)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Wortverknüpfungen

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert des Parameters IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

DECO_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	UINT	E, A, M, D, L, P	Position des Bits im Ausgangswert, das gesetzt wird.
_<Datentyp>		Bitfolgen Vorbelegung: DWORD	-	Datentyp des Funktionswerts: 1. Sie können den Datentyp der Anweisung explizit mit Hilfe von "_" angeben. 2. Wenn Sie den Datentyp nicht explizit angeben, dann wird er durch die verwendeten Variablen oder typisierten Konstanten bestimmt. 3. Wenn Sie den Datentyp weder explizit angeben, noch definierte Variablen oder typisierte Konstanten angeben, wird der voreingestellte Datentyp verwendet.
Funktionswert		Bitfolgen	E, A, M, D, L, P	Aktueller Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := DECO(IN := "Tag_Value");
"Tag_Result2" := DECO_BYTE(IN := "Tag_Value2");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

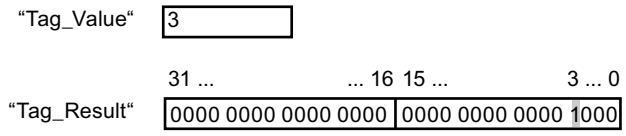

Die Anweisung liest die Zahl "3" aus dem Wert des Operanden "Tag_Value" ab und setzt das dritte Bit im Wert des Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben diese als Ergebnis aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P	Eingangswert
Funktionswert		INT	E, A, M, D, L, P	Bitnummer des Bits im Eingangswert, das ausgelesen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := ENCO(IN := "Tag_Value");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

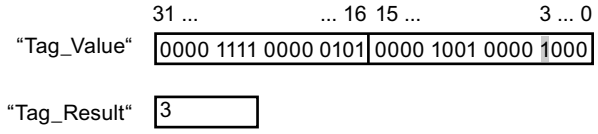

Die Anweisung liest das niederwertigste gesetzte Bit des Operanden "Tag_Value" ab und schreibt die Bitposition "3" in den Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Parameter G) einen der Parameter IN0 oder IN1 aus und geben dessen Inhalt als Ergebnis aus. Wenn der Parameter G den Signalzustand "0" hat, wird der Wert am Parameter IN0 kopiert. Wenn der Parameter G den Signalzustand "1" hat, wird der Wert am Parameter IN1 kopiert und als Funktionswert zurückgeliefert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom Datentyp der gleichen Klasse sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
G	Input	BOOL	BOOL	E, A, M, D, L	Schalter
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, TOD, LTOD, DT, LDT	E, A, M, D, L, P	Erster Eingangswert

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, TOD, LTOD, DT, LDT	E, A, M, D, L, P	Zweiter Eingangswert
Funktionswert		Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, TOD, LTOD, DT, LDT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SEL(G := "Tag_Value",
                    IN0 := "Tag_0",
                    IN1 := "Tag_1");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0	1
Tag_0	W#16#0000	W#16#4C
Tag_1	W#16#FFFF	D#16#5E
Tag_Result	W#16#0000	D#16#5E

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MUX: Multiplexen

Beschreibung

Mit der Anweisung "Multiplexen" kopieren Sie den Wert eines ausgewählten Eingangsparameters und geben diesen aus. Durch den Parameter K bestimmen Sie die Nummer des Eingangsparameters, dessen Wert kopiert wird. Die Nummerierung fängt dabei mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Sie können maximal 32 Eingänge deklarieren.

An den Eingängen sind numerische Datentypen und Zeitdatentypen zugelassen. Alle parametrierten Variablen müssen vom gleichen Datentyp sein.

Der Funktionswert ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Es treten Fehler während der Bearbeitung der Anweisung auf.
- Wenn der Eingang am Parameter K außerhalb der verfügbaren Eingänge liegt und der Eingang INELSE nicht verwendet wird, dann ist der Funktionswert ungültig.

Hinweis

Parameter K hat eine negative Ganzzahl

Wenn Sie an den Eingangsparametern eine Variable mit gültigem Datentyp angeben und der Parameter K eine negative Ganzzahl hat, dann wird der Wert der Variable verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P	Gibt den Parameter an, dessen Inhalt übertragen wird. • Wenn K = 0 => Parameter IN0 • Wenn K = 1 => Parameter IN1, etc.
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT	E, A, M, D, L, P	Erster Eingangswert

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT	E, A, M, D, L, P	Zweiter Eingangswert
INn	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT	E, A, M, D, L, P	Optionale Eingangswerte
INELSE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT	E, A, M, D, L, P	Gibt den Wert an, der bei K <> n kopiert wird.
Funktionswert		Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := MUX(K := "Tag_Number",
                    IN0 := "Tag_1",
                    IN1 := "Tag_2",
                    INELSE := "Tag_3");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Number	1	4
Tag_1	DW#16#00000000	DW#16#00000000
Tag_2	DW#16#003E4A7D	DW#16#003E4A7D
Tag_3	DW#16#FFFF0000	DW#16#FFFF0000
Tag_Result	DW#16#003E4A7D	DW#16#FFFF0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

DEMUX: Demultiplexen

Beschreibung

Mit der Anweisung "Demultiplexen" übertragen Sie den Wert des Eingangsparameters IN an einen ausgewählten Ausgangsparameter. Die Auswahl des Ausgangsparameters erfolgt abhängig vom Wert des Parameters K. Der Parameter K gibt die Nummer des Ausgangsparameters, in welchen der Wert des Eingangsparameters IN übertragen wird. Die anderen Ausgangsparameter werden nicht verändert. Die Nummerierung fängt mit OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Sie können maximal 32 Ausgangsparameter deklarieren.

Wenn der Wert des Parameters K größer als die Anzahl der Ausgangsparameter ist, wird der Inhalt des Eingangsparameters IN in den Ausgangsparameter OUTELSE kopiert und dem Freigabeausgang ENO der Signalzustand "0" zugewiesen.

Der Funktionswert ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Hinweis

Parameter K < 0 oder K > verfügbare Ausgänge

Wenn Sie am Parameter K einen Wert angeben, der außerhalb der verfügbaren Ausgänge liegt (K < 0 oder K > verfügbare Ausgänge), dann wird am Ausgangsparameter OUTELSE der Wert des Eingangsparameters IN ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
K	Input	Ganzzahlen	Ganzzahlen	E, A, M, D, L, P	Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. • Wenn K = 0 => Parameter OUT0 • Wenn K = 1 => Parameter OUT1, etc.
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Eingangswert
OUT0	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Erster Ausgang
OUT1	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Zweiter Ausgang

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
OUTn	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Optionale Ausgänge
OUTELSE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, STRING, CHAR, WCHAR, TOD, DATE, DT	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT	E, A, M, D, L, P	Ausgang, in welchen der Wert des Eingangs IN bei K > n kopiert wird.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
DEMUX(K := "Tag_Number",
      IN := "Tag_Value",
      OUT0 := "Tag_1",
      OUT1 := "Tag_2",
      OUTELSE := "Tag_3");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

Parameter	Operand	Werte	
K	Tag_Number	2	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

Parameter	Operand	Werte	
OUT0	Tag_1	unverändert	unverändert
OUT1	Tag_2	DW#16#FFFFFFFF	unverändert
OUTELSE	Tag_3	unverändert	DW#16#003E4A7D

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach rechts und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:

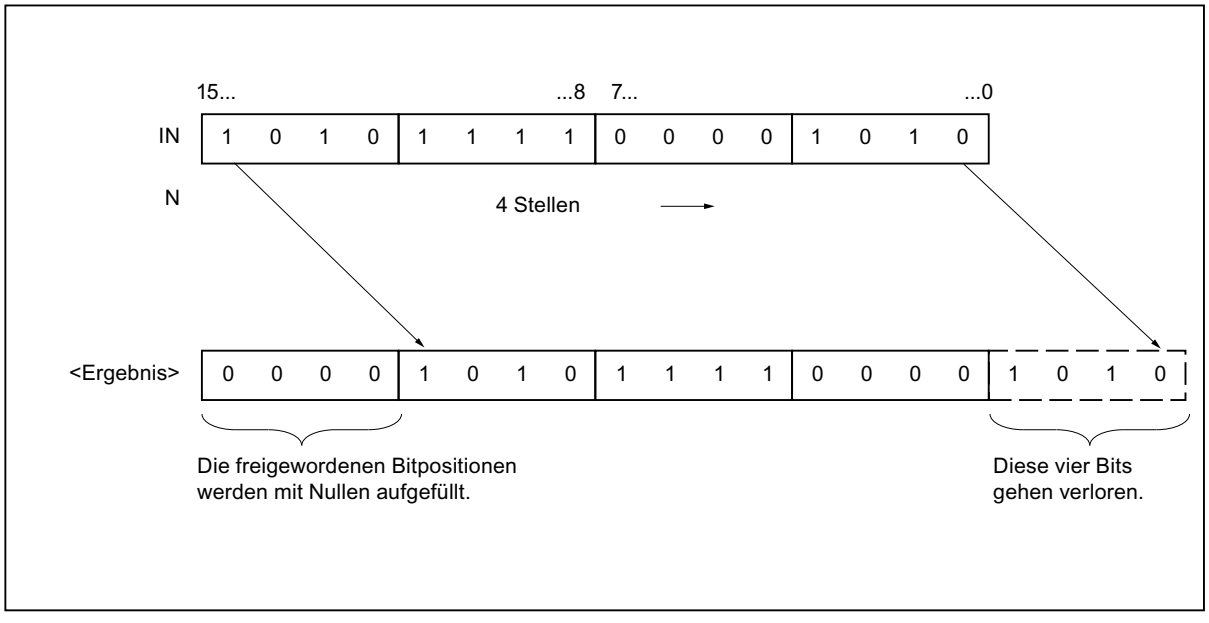

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Wert, der verschoben wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Anzahl der Bits, um die der Wert (IN) verschoben wird
Funktionswert		Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SHR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	3
Funktionswert	Tag_Result	0000 0111 1111 0101

Der Wert des Operanden "Tag_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen werden im Ergebniswert mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

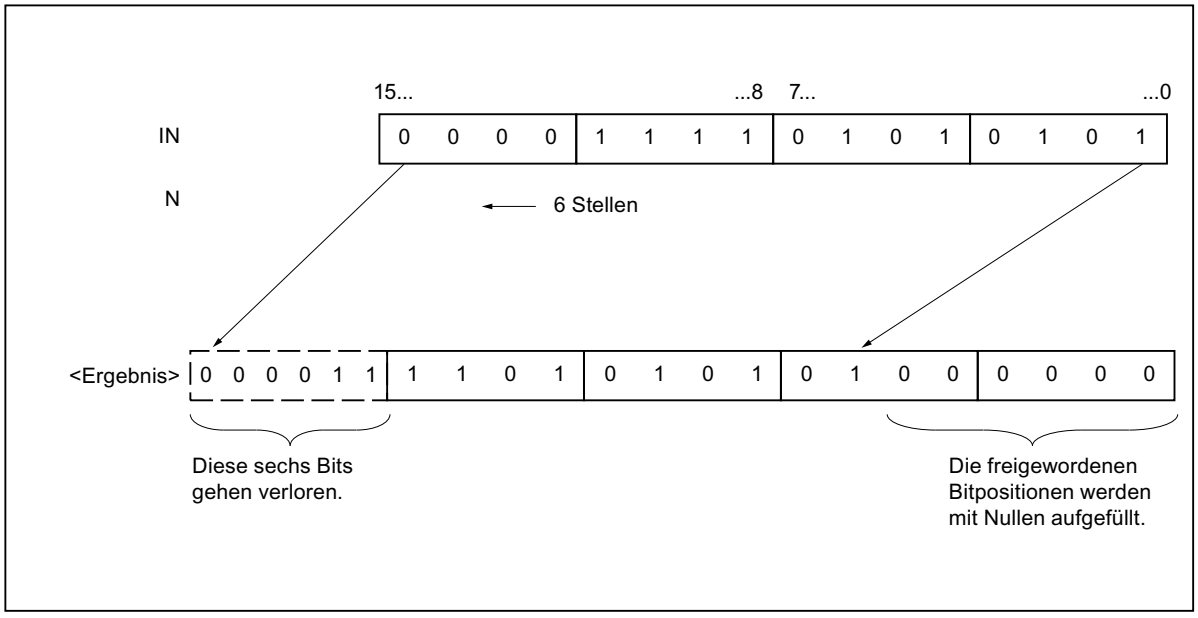

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Wert, der verschoben wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Anzahl der Bits, um die der Wert (IN) verschoben wird
Funktionswert		Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SHL(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	4
Funktionswert	Tag_Result	1111 1010 1111 0000

Der Wert des Operanden "Tag_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach rechts und weisen das Ergebnis dem angegebenen Operanden zu. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

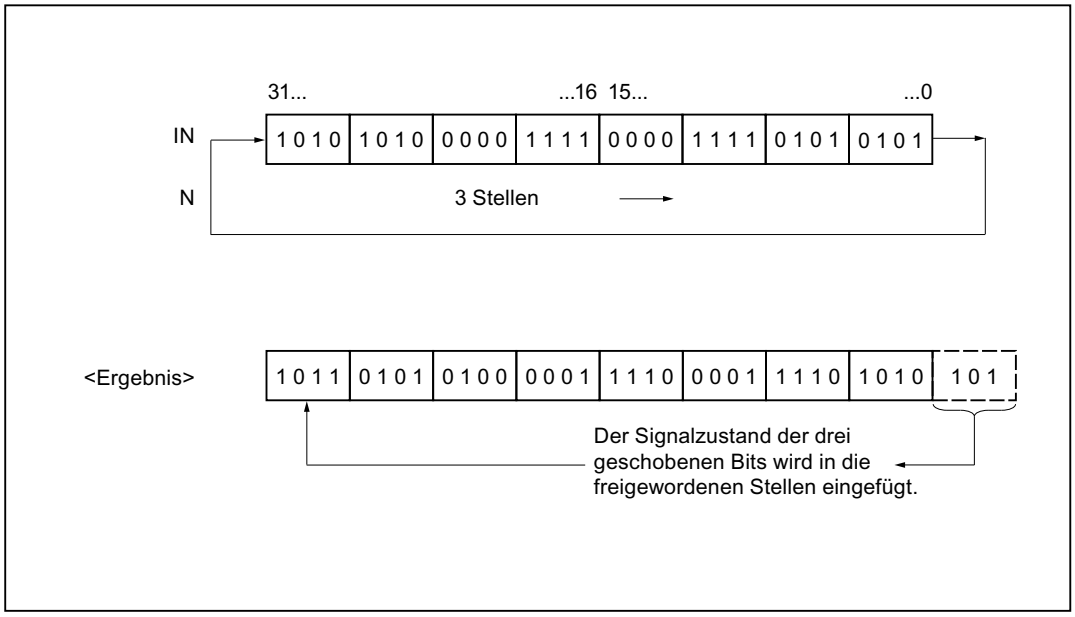

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Wert, der rotiert wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Anzahl der Bitstellen, um die der Wert (IN) rotiert wird
Funktionswert		Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0000 1111 1001 0101
N	Tag_Number	5
Funktionswert	Tag_Result	1010 1000 0111 1100

Der Inhalt des Operanden "Tag_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

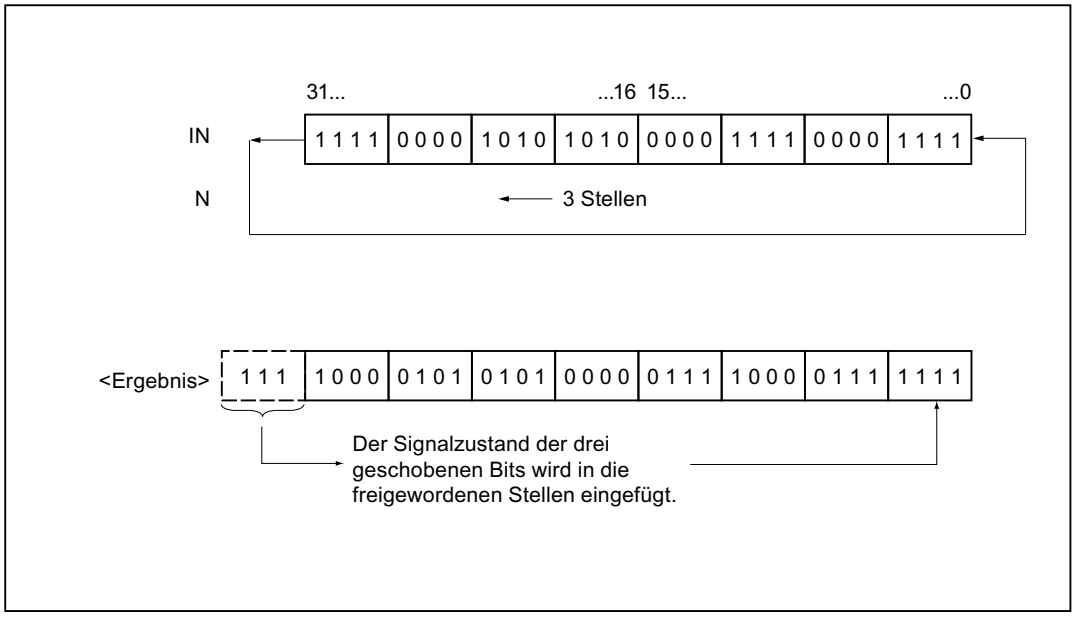

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Wert, der rotiert wird
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Anzahl der Bitstellen, um die der Wert (IN) rotiert wird
Funktionswert		Bitfolgen, Ganzzahlen	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROL(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1010 1000 1111 0110
N	Tag_Number	5
Funktionswert	Tag_Result	0001 1110 1101 0101

Der Inhalt des Operanden "Tag_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Legacy

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L	Maximale Schrittnummer (z. B.: LST_STEP = 16#08; Es sind maximal 8 Schritte möglich.)
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"DRUM_DB"(RESET := "Tag_Reset"
          JOG := "Tag_Input_Jog"
          DRUM_EN := "Tag_Input_DrumEN"
          LST_STEP := "Tag_Number_LastStep"
          EVENT1 := "MyTag_Event_1"
          EVENT2 := "MyTag_Event_2"
          EVENT3 := "MyTag_Event_3"
          EVENT4 := "MyTag_Event_4"
          EVENT5 := "MyTag_Event_5"
          EVENT6 := "MyTag_Event_6"
          EVENT7 := "MyTag_Event_7"
          EVENT8 := "MyTag_Event_8"
          EVENT9 := "MyTag_Event_9"
          EVENT10 := "MyTag_Event_10"
          EVENT11 := "MyTag_Event_11"
          EVENT12 := "MyTag_Event_12"
          EVENT13 := "MyTag_Event_13"
          EVENT14 := "MyTag_Event_14"
          EVENT15 := "MyTag_Event_15"
          EVENT16 := "MyTag_Event_16"
          OUT1 => "MyTag_Output_1"
          OUT2 => "MyTag_Output_2"
          OUT3 => "MyTag_Output_3"
          OUT4 => "MyTag_Output_4"
          OUT5 => "MyTag_Output_5"
          OUT6 => "MyTag_Output_6"
          OUT7 => "MyTag_Output_7"
          OUT8 => "MyTag_Output_8"
          OUT9 => "MyTag_Output_9"
          OUT10 => "MyTag_Output_10"
          OUT11 => "MyTag_Output_11"
          OUT12 => "MyTag_Output_12"
          OUT13 => "MyTag_Output_13"
          OUT14 => "MyTag_Output_14"
          OUT15 => "MyTag_Output_15"
          OUT16 => "MyTag_Output_16"
          Q => "Tag_Output_Q"
          OUT_WORD => "Tag_OutputWord"
          ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde(O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET (nur wenn ET < PT ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn ET < PT ist), OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	D, L	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	D, L	Verlaufsbit zu CMD

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"DCAT_DB"(CMD := "Tag_Input_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          Q => "Tag_Output_Q",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L	Vorherige Systemzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_HIS	Static	BOOL	D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	D, L	Verlaufsbit "Schließen"

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT

Eingangsparameter		Ausgangsparameter
>= PT	ET >= PT	
Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE		

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"MCAT_DB"(O_CMD := "Tag_Iput_O_CMD",
          C_CMD := "Tag_Input_C_CMD",
          S_CMD := "Tag_Input_S_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          OO => "Tag_OutputOpen",
          CO => "Tag_OutputClosed",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA",
          Q => "Tag_Output_Q");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE

Parameter	Operand	Wert
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird. Am Parameter IN können auch Konstanten angegeben werden.
SAMPLE_T	Input	INT	E, A, M, D, L, P	Abtastzeit Am Parameter SAMPLE_T können auch Konstanten angegeben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	REAL	E, A, M, D, L	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P	Vorheriger Ausgang

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"LEAD_LAG_DB"(IN := "Tag_Input",
              SAMPLE_T := "Tag_Input_SAMPLE_T",
              OUT => "Tag_Output_Result",
              ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	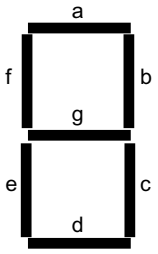
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige
Funktionswert		VOID		Leerer Funktionswert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SEG(IN := "Tag_Input",
    OUT => "Tag_Output");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
		Hexadezimal	Binär
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW16#065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Operanden angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Bitfolgen	E, A, M, D, L, P	7-stellige BCD-Zahl
Funktionswert		DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := BCDCPL("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Operand	Wert*
Tag_Input	DW#16#01234567
Tag_Result	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bit eines Operanden, die auf den Signalzustand "1" gesetzt sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	DWORD	E, A, M, D, L, P	Operand, dessen gesetzte Bit gezählt werden
Funktionswert		INT	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := BITSUM("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Operand	Wert*
Tag_Input	DW#16#12345678
Tag_Result	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

4.1.3.5 GRAPH

GRAPH-Kette

Schritt und Transition

Beschreibung

Mit dem Strukturelement "Schritt und Transition" können Sie gleichzeitig einen Schritt und eine Transition in Ihre Ablaufkette einfügen.

Siehe auch:

Schritt (Seite 1983)

Transition (Seite 1984)

Siehe auch

Schritt (Seite 1983)

Transition (Seite 1984)

Kettenende (Seite 1985)

Sprung zu Schritt (Seite 1985)

Alternativzweig (Seite 1986)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Schritte und Transitionen (Seite 7692)

Schritt

Beschreibung

Schritte dienen dazu, komplexe Automatisierungsaufgaben in übersichtliche Teilaufgaben zu untergliedern, die anschließend mithilfe von Aktionen gelöst werden können. Die einzelnen Schritte werden in Ablaufketten organisiert, sodass bei Programmablauf jeder Schritt in der festgelegten Reihenfolge abgearbeitet werden kann. Jeder Schritt muss einen eindeutigen Namen und eine eindeutige Nummer erhalten.

Damit die Bearbeitung tatsächlich erfolgt, muss der Schritt durch eine der folgenden Bedingungen aktiv werden:

- Der Schritt wurde als Initialschritt definiert.
- Die Transition des Vorgängerschritts ist erfüllt.
- Der Schritt wird durch eine ereignisabhängige Aktion aufgerufen.

Nachdem alle Aktionen bearbeitet wurden, wird der Schritt wieder inaktiv.

Schritte, in denen keine Aktionen programmiert sind, werden Leerschritte genannt. Ein solcher Leerschritt verhält sich wie ein aktiver Schritt und die nachfolgende Transition wird immer erfüllt.

Siehe auch

Grundlagen zu Schritten (Seite 7692)

Schritt und Transition (Seite 1983)

Transition (Seite 1984)

Kettenende (Seite 1985)

Sprung zu Schritt (Seite 1985)

Alternativzweig (Seite 1986)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Transition

Beschreibung

Transitionen stehen zwischen den Schritten und enthalten die Bedingungen zum Weiterschalten von einem Schritt zum nächsten. Wenn die Weiterschaltbedingungen einer Transition erfüllt sind, wird der nächste Schritt aktiv und seine Aktionen werden bearbeitet. Sie können die Bedingungen einer Transition entweder in KOP der FUP programmieren.

Siehe auch

Grundlagen zu Transitionen (Seite 7694)

Schritt und Transition (Seite 1983)

Schritt (Seite 1983)

Kettenende (Seite 1985)

Sprung zu Schritt (Seite 1985)

Alternativzweig (Seite 1986)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Kettenende

Beschreibung

Mit dem Strukturelement "Kettenende" können Sie eine Ablaufkette oder eine Verzweigung beenden. In einer Simultanverzweigung muss dem Kettenende jedoch eine Transition vorangehen.

Hinweis

Werden alle Zweige einer Ablaufkette mit einem Kettenende abgeschlossen, kann die Ablaufkette entweder durch den Parameter "INIT_SQ" oder in der Task Card "Testen" in der Palette "Kettensteuerung" mit der Schaltfläche "Initialisieren" neu gestartet werden.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 1983)

Schritt (Seite 1983)

Transition (Seite 1984)

Sprung zu Schritt (Seite 1985)

Alternativzweig (Seite 1986)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Sprung zu Schritt

Beschreibung

Mit einem Sprung können Sie die Programmausführung mit einem beliebigem Schritt innerhalb des GRAPH-Funktionsbausteins fortsetzen. Sprünge können am Ende des Hauptzweiges oder eines Alternativzweiges eingefügt werden, sodass eine zyklische Bearbeitung der Ablaufkette möglich ist. Sprung und Sprungziel werden in der Ablaufkette als Pfeile dargestellt, wobei beim Sprungziel die Absprungtransition angegeben ist.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 1983)

Schritt (Seite 1983)

Transition (Seite 1984)

Kettenende (Seite 1985)

Alternativzweig (Seite 1986)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Alternativzweig

Beschreibung

Mithilfe von Alternativzweigen können Sie ODER-Verzweigungen programmieren. Das bedeutet, dass Sie nach einem Schritt Zweige einfügen, die mit einer Transition beginnen. Abhängig davon, welche Transition zuerst erfüllt ist, wird der entsprechende Zweig abgearbeitet. Sind mehrere Transitionen gleichzeitig erfüllt, hat jeweils die am weitesten links stehende Transition die höchste Priorität und der entsprechende Zweig wird abgearbeitet. Alternativzweige enden wieder mit einer Transition.

In einer Ablaufkette können Sie bis zu 125 Alternativzweige programmieren.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 1983)

Schritt (Seite 1983)

Transition (Seite 1984)

Kettenende (Seite 1985)

Sprung zu Schritt (Seite 1985)

Simultanzweig (Seite 1986)

Verzweigung schließen (Seite 1987)

Simultanzweig

Beschreibung

Mithilfe von Simultanzweigen können Sie UND-Verzweigungen programmieren. Das bedeutet, dass Sie mit einer Transition mehrere Schritte aktiv setzen, deren Aktionen dann ausgeführt werden. Darum beginnen und enden Simultanzweige immer mit einem Schritt.

Die nachfolgenden Transitionen der Simultanzweige befinden sich auf dem Hauptzweig, wobei die verschiedenen Simultanzweige an verschiedenen Stellen des Hauptzweigs geschlossen werden können. Beachten Sie, dass Zweige, die Sie in einer Transition zusammenführen, erst dann in den nächsten Schritt schalten, wenn diese Zweige komplett abgearbeitet sind.

In einer Ablaufkette können Sie bis zu 249 Simultanzweige programmieren.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)
Schritt und Transition (Seite 1983)
Schritt (Seite 1983)
Transition (Seite 1984)
Kettenende (Seite 1985)
Sprung zu Schritt (Seite 1985)
Alternativzweig (Seite 1986)
Verzweigung schließen (Seite 1987)

Verzweigung schließen

Beschreibung

Mit dem Element "Verzweigung schließen" können Sie Simultan- oder Alternativzweige zu ihrem Elternzweig hin schließen. Dies ist notwendig, wenn Sie die Verzweigung nicht mit einem Sprung oder einem Kettenende abschließen möchten. In einem Simultanzweig können Sie "Verzweigung schließen" nur nach einem Schritt einfügen.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)
Schritt und Transition (Seite 1983)
Schritt (Seite 1983)
Transition (Seite 1984)
Kettenende (Seite 1985)
Sprung zu Schritt (Seite 1985)
Alternativzweig (Seite 1986)
Simultanzweig (Seite 1986)

GRAPH-Aktionen

Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Parameter IN "0" ist, wird der Parameter ET zurückgesetzt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Parameter ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TP_TIME oder TP_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TP_TIME, TP_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Impuls erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Impulsausgang
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen" nach dem Starten:

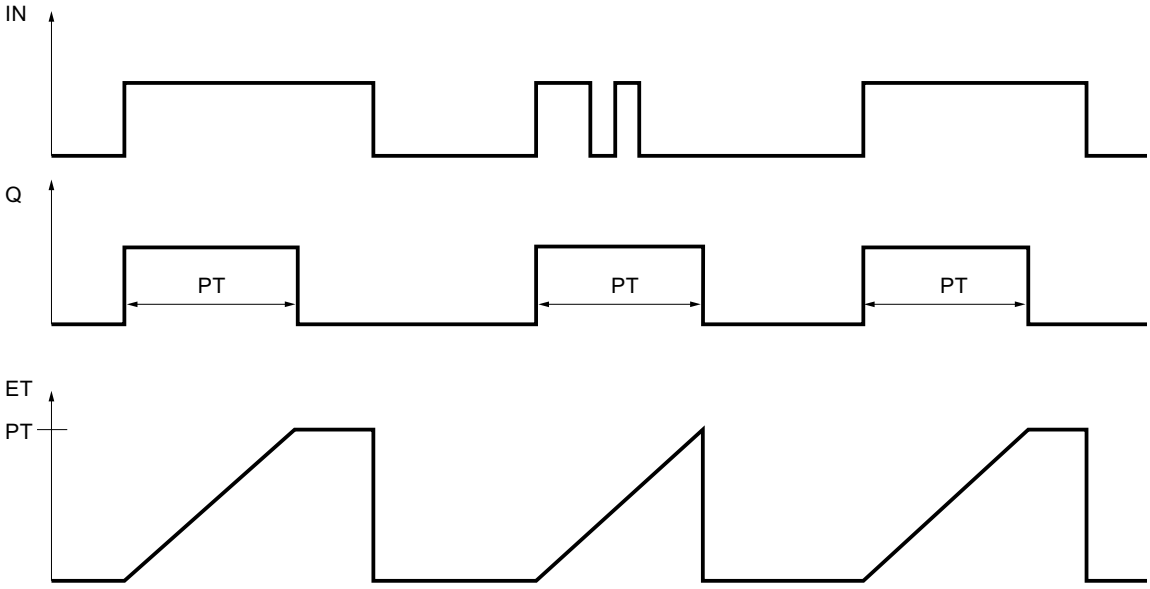

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TP TIME, "IEC_TP_DB"
      (IN := "Tag_Start"
       PT := "Tag_PresetTIME"
       Q => "Tag_Status"
       ET => "Tag_ElapsedTIME"
       )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTIME" gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt so lange gesetzt, wie der Starteingang IN noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Parameter ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TON_TIME oder TON_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TON_TIME, TON_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Einschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Signalzustand, der um die Zeit PT verzögert wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Einschaltverzögerung erzeugen" nach dem Starten:

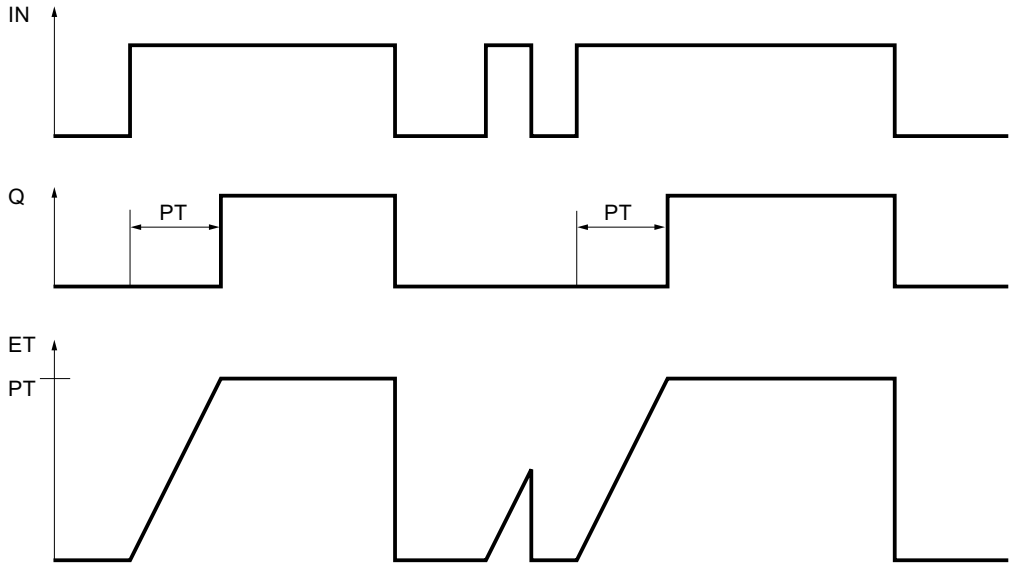

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TON TIME, "IEC_TON_DB"
```

GRAPH

```
(IN := "Tag_Start"
 PT := "Tag_PresetTIME"
 Q => "Tag_Status"
 ET => "Tag_ElapsedTIME"
 )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt. Wenn der Signalzustand am Operanden "Tag_Start" von "1" auf "0" wechselt, wird der Operand "Tag_Status" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET solange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine

Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TOF_TIME oder TOF_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TOF_TIME, TOF_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Ausschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Signalzustand, der um die Zeit PT verzögert wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Ausschaltverzögerung erzeugen" nach dem Starten:

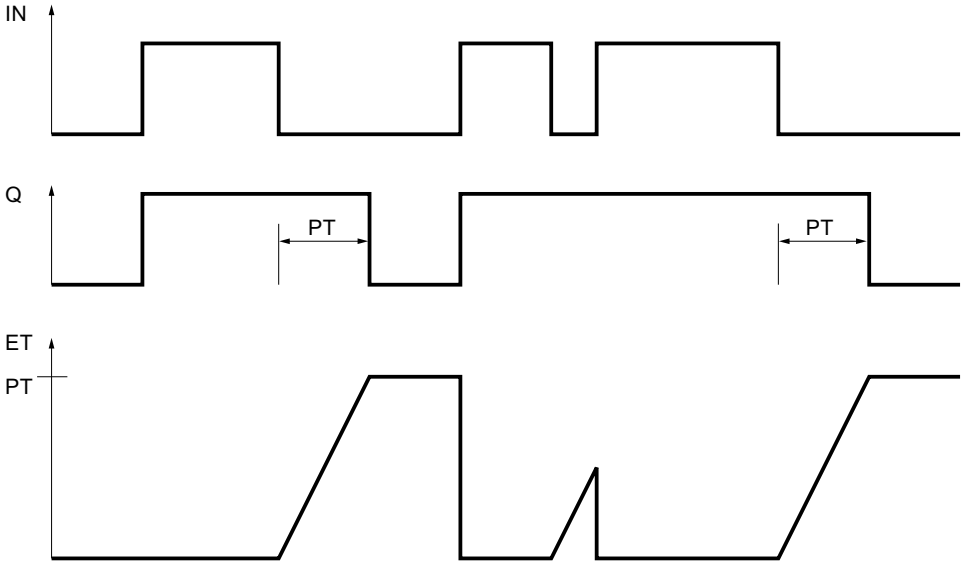

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TOF TIME, "IEC_TOF_DB"
      (IN := "Tag_Start"
       PT := "Tag_PresetTIME"
       Q => "Tag_Status"
       ET => "Tag_ElapsedTIME"
       )
```

Bei einem Wechsel im Signalzustand des Operanden "Tag_Start" von "0" auf "1" wird der Operand "Tag_Status" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTIME" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

TONR: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Eingang IN von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt (negative Signalflanke).

Der Eingang R setzt die Ausgänge ET und Q, unabhängig vom Signalzustand am Starteingang, zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit ist eine Struktur vom Datentyp IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_TIMER oder IEC_LTIMER (z. B. "MyIEC_TIMER")
- Deklaration als lokale Variable vom Typ TONR_TIME, TONR_LTIME, IEC_TIMER oder IEC_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht in den folgenden Fällen:

- Bei einem Aufruf der Anweisung, wenn die Ausgänge ET oder Q verschaltet sind. Wenn die Ausgänge nicht verschaltet sind, dann wird der aktuelle Zeitwert am Ausgang ET nicht aktualisiert.
- Bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, P oder Konstante	Starteingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PT	Input	TIME, LTIME	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L, P	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME, LTIME	E, A, M, D, L, P	Akkumulierte Zeit

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":

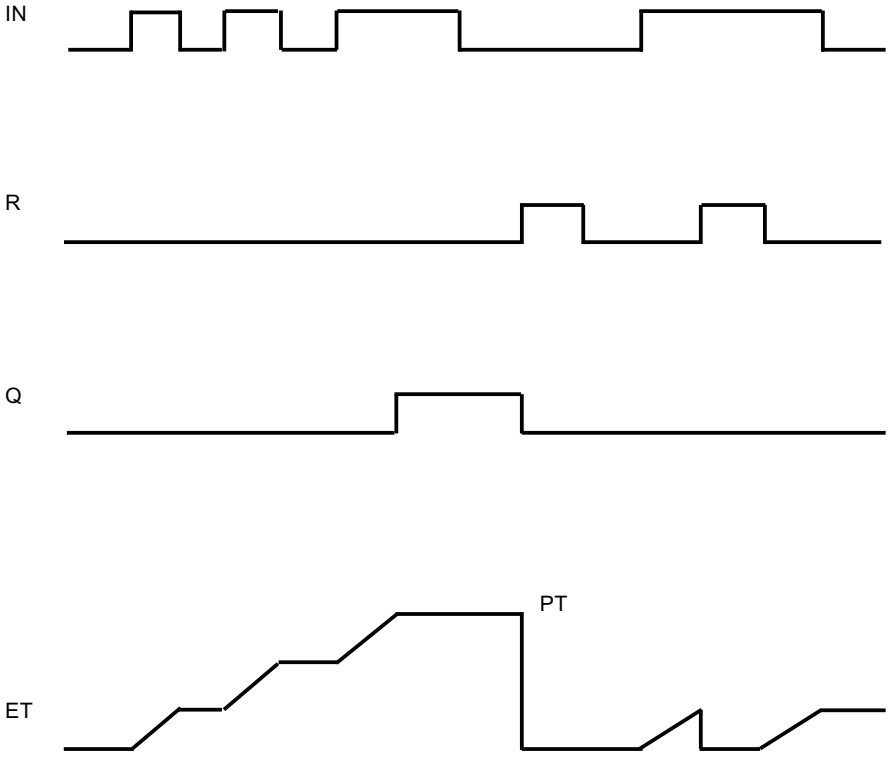

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TONR TIME, "IEC_TONR_DB"
     (IN := "Tag_Start"
      R := "Tag_Reset"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Während des Ablaufs der Zeit werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" des Operanden "Tag_Start" erfasst werden. Die akkumulierte Zeit wird im Operanden "Tag_ElapsedTIME" gespeichert. Wenn der am Parameter PT angegebene Zeitwert erreicht ist, wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTIME" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Lokale Variable

- CTU_SINT / CTU_USINT
- CTU_INT / CTU_UINT

- CTU_DINT / CTU_UDINT
- CTU_LINT / CTU_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTU_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L oder Konstante	Zähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTU INT, "IEC_CTU_DB"
```

GRAPH

```
(CU := "Tag_Start"
 R := "Tag_ResetCOUNTER"
 PV := "Tag_PresetValue"
 Q => "Tag_Status"
 CV => "Tag_CounterValue"
 )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CounterValue" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "Tag_Status" übernommen. Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu GRAPH (Seite 7685)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange am Parameter LD der

Signalzustand "1" ansteht, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Lokale Variable

- CTD_SINT / CTD_USINT
- CTD_INT / CTD_UINT
- CTD_DINT / CTD_UDINT
- CTD_LINT / CTD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L oder Konstante	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTD INT, "IEC_CTD_DB"
     (CD := "Tag_Start"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      Q => "Tag_Status"
      CV => "Tag_CounterValue"
      )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Rückwärts zählen" ausgeführt und der Wert am Ausgang "Tag_CounterValue" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu GRAPH (Seite 7685)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Wert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0".

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Lokale Variable

- CTUD_SINT / CTUD_USINT
- CTUD_INT / CTUD_UINT
- CTUD_DINT / CTUD_UDINT
- CTUD_LINT / CTUD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC_<Zähler> (z. B. "MyIEC_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD_<Datentyp> oder IEC_<Zähler> im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Wenn Sie den IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) anlegen, dann wird der Instanz-Datenbaustein per Voreinstellung mit "optimiertem Bausteinzugriff" angelegt und die einzelnen Variablen als remanent definiert. Weitere Informationen zur Einstellung der Remanenz in einem Instanz-Datenbaustein finden Sie unter "Siehe auch".

Wenn Sie den IEC-Zähler als lokale Variable (Multiinstanz) in einem Funktionsbaustein mit "optimiertem Bausteinzugriff" anlegen, dann wird diese in der Bausteinschnittstelle als remanent definiert.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L oder Konstante	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L oder Konstante	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P oder Konstante	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P oder Konstante	Ladeeingang
PV	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	Ganzzahlen, CHAR, WCHAR, DATE	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTUD INT, "IEC_CTUD_DB"
     (CU := "Tag_StartCTU"
      CD := "Tag_StartCTD"
      R := "Tag_ResetCOUNTER"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      QU => "Tag_CounterStatusUP"
      QD => "Tag_CounterStatusDOWN"
      CV => "Tag_CounterValue"
      )
```

Wenn der Signalzustand am Eingang "Tag_StartCTU" oder am Eingang "Tag_StartCTD" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "Tag_StartCTU" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CounterValue" abgelegt. Wenn eine positive Signalflanke am Eingang "Tag_StartCTD" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CounterValue" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "Tag_CounterStatusUP" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_CounterStatusUP" den Signalzustand "0".

Der Ausgang "Tag_CounterStatusDOWN" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_CounterStatusDOWN" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu GRAPH (Seite 7685)

Mathematische Funktionen

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Operandenwerts. Wenn z. B. ein positiver Wert ansteht, wird das negative Äquivalent dieses Werts ausgegeben.

Der Wert des Ausgangs OUT ist ungültig, wenn das Ergebnis der Anweisung außerhalb des Bereichs liegt, der für den am Ausgang OUT angegebenen Datentyp zulässig ist (gilt nur für Ganzzahlen).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	SINT, INT, DINT, LINT, Gleitpunkt-zahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	SINT, INT, DINT, LINT, Gleitpunkt-zahlen	E, A, M, D, L, P	Zweierkomplement des Ein-gangswerts

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := NEG_REAL("Tag_InValue")
```

Die Anweisung wechselt das Vorzeichen des Werts am Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Operandenwertes.

Der Wert des Ergebnis ist ungültig, wenn eine Gleitpunktzahl einen ungültigen Wert aufweist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	SINT, INT, DINT, LINT, Gleitpunkt-zahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	SINT, INT, DINT, LINT, Gleitpunkt-zahlen	E, A, M, D, L, P	Absolutwert des Eingangs-werts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ABS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	-6.234
Tag_OutValue	6.234

Die Anweisung berechnet den Absolutbetrag des Wertes am Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge IN1 und IN2 und schreiben den kleinsten Wert in den Ausgang OUT.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MIN INT
     (IN1 := "TagIn_Value1"
      IN2 := "TagIn_Value2"
      OUT => "Tag_Minimum"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
OUT	Tag_Minimum	12222

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "Tag_Minimum".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1 und IN2 und schreiben den größten Wert in den Ausgang OUT.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MAX INT
     (IN1 := "TagIn_Value1"
      IN2 := "TagIn_Value2"
      OUT => "Tag_Maximum"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
OUT	Tag_Maximum	14444

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "Tag_Maximum".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis der am Parmeter IN angegebene Wert. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Parameter MN ist größer als der Wert am Parameter MX.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ergebnis
Die Datentypen TOD, LTOD, DATE und LDT können nur verwendet werden, wenn die IEC-Prüfung nicht aktiviert ist.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL LIMIT INT
      (MN := "Tag_LowLimit"
       IN := "Tag_InputValue"
       MX := "Tag_HighLimit"
       OUT => "Tag_Result"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

Der Wert des Operanden "Tag_InputValue" wird mit den Werten der Operanden "Tag_LowLimit" und "Tag_HighLimit" verglichen. Da der Wert des Operanden "Tag_InputValue" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_LowLimit" in den Ausgang "Tag_Result" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den Wert der Gleitpunktzahl des Operanden und schreiben das Ergebnis in den Ausgang.

Wenn der Operand eine ungültige Gleitpunktzahl enthält, dann ist der Wert des Ergebnises ungültig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswertes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SQR("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	5.0
Tag_OutValue	25.0

Die Anweisung quadriert den Wert des Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem Eingangswert die Quadratwurzel und speichern das Ergebnis im angegebenen Operanden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück. Wenn der Wert des Operanden "0" ist, ist das Ergebnis auch "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SQRT("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	25.0
Tag_OutValue	5.0

Die Anweisung zieht aus dem Wert des Operanden "Tag_InValue" die Quadratwurzel und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu GRAPH (Seite 7685)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem Eingangswert den natürlichen Logarithmus zur Basis e (e = 2,718282). Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := LN("Tag_InValue")
```

Die Anweisung bildet den natürlichen Logarithmus des Wertes am Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu GRAPH (Seite 7685)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem Wert des Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH
```
"Tag_OutValue" := EXP("Tag_InValue")
```

Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_InValue" und gibt das Ergebnis (e^{IN}) am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu GRAPH (Seite 7685)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus eines Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Sinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH
```
"Tag_OutValue" := SIN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+1.570796 (π/2)
Tag_OutValue	1.0

Die Anweisung berechnet den Sinus des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := COS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+1.570796 (π/2)
Tag_OutValue	0

Die Anweisung berechnet den Cosinus des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Tangens des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := TAN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+3.141593 (π)
Tag_OutValue	0

Die Anweisung berechnet den Tangens des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Operanden angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Operanden ist keine gültige Gleitpunktzahl.
- Der Wert des Operanden liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Sinuswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ASIN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1.0
Tag_OutValue	+1.570796 (π/2)

Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Operanden angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße kann in einem Wertebereich von 0 bis +π liegen.

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Operanden ist keine gültige Gleitpunktzahl.
- Der Wert des Operanden liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Cosinuswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ACOS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0
Tag_OutValue	+1.570796 (π/2)

Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Operanden angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Tangenswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ATAN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1.0
Tag_OutValue	+0.785398 (π/4)

Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu GRAPH (Seite 7685)

FRAC: Nachkommastellen ermitteln

Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" ermitteln Sie die Nachkommastellen des Wertes am Operanden. Wenn am Operanden z. B. der Wert 123,4567 ansteht, liefert das Ergebnis den Wert 0,4567.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert, dessen Nachkommastellen ermittelt werden
RET_VAL	Output	Gleitpunktzahlen	E, A, M, D, L, P	Nachkommastellen des Eingangswerts

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL FRAC REAL
      (IN := "Tag_InValue"
       RET_VAL => "Tag_OutValue"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	2.555
Tag_OutValue	0.555

Die Anweisung kopiert die Nachkommastellen aus dem Wert des Operanden "Tag_InValue" in den Operanden "Tag_OutValue".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

Verschieben

Deserialize: Deserialisieren

Beschreibung

Mit der Anweisung "Deserialisieren" wandeln Sie die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder um und füllen dessen kompletten Inhalt.

Der Speicherbereich, in dem die sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> liegt, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Speicherbereiche zugelassen. Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Stellen Sie vor der Umwandlung sicher, dass genügend Speicherplatz vorhanden ist. Wenn der Speicherbereich mithilfe der Anweisung "Serialisieren" befüllt wurde und eventuell Füll-Bytes eingefügt wurden, dann werden diese bei der Umwandlung nicht berücksichtigt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Mithilfe der Anweisung können Sie mehrere sequenzielle Darstellungsformen von umgewandelten Daten schrittweise wieder in ihren ursprünglichen Zustand umwandeln.

Wenn Sie nur eine einzelne sequenzielle Darstellungsform eines PLC-Datentyps (UDT), STRUCT oder ARRAY of <Datentyp> wieder umwandeln möchten, dann können Sie dazu auch direkt die Anweisung "TRCV: Daten über Kommunikationsverbindung empfangen" verwenden.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das BOOL eine Länge von 1 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Optimierter Speicherbereich

Um größere Strukturen zu deserialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich für die sequenzielle Darstellung auch mit optimiertem Zugriff deklarieren. Der Inhalt der sequenziellen Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich. Der Zugriff auf die Bytes des ARRAYs ist ausschließlich symbolisch möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_ARRAY	Input	VARIANT	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der Datenstrom gespeichert ist.
DEST_VARIABLE	InOut	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die deserialisiert werden soll.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_ARRAY und DEST_VARIABLE überschneiden sich.
8136	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8150	Der Datentyp VARIANT am Parameter SRC_ARRAY enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_ARRAY
8153	Am Parameter SRC_ARRAY steht nicht genügend Speicherplatz zur Verfügung.
8154	Ungültiger Datentyp am Parameter SRC_ARRAY
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_VARIABLE
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehlercode* (W#16#...)	Erläuterung
8136	Der Zugriff am Parameter SRC_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL Deserialize VARIANT
```

GRAPH

```
(SRC_ARRAY := "Buffer".Field
 DEST_VARIABLE => "Target".Client
 POS := #BufferPos
 RET_VAL => #Error
 )
```

Die Anweisung deserialisiert die sequenzielle Darstellungsform der Kundendaten aus der Variable "Buffer" und schreibt sie in die Variable "Target". Der Operand #BufferPos speichert den Index des ersten Bytes nach der Anzahl der Bytes, die die umgewandelten Kundendaten belegen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu GRAPH (Seite 7685)

Serialize: Serialisieren

Beschreibung

Mit der Anweisung "Serialisieren" wandeln Sie mehrere PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> in eine sequenzielle Darstellungsform um, ohne dass Teile ihrer Struktur verloren gehen.

Mithilfe der Anweisung können Sie mehrere strukturierte Daten aus Ihrem Programm in einem Puffer, der vorzugsweise in einem globalen Datenbaustein liegt, zwischenspeichern und an eine andere CPU senden. Der Speicherbereich, in dem die umgewandelten Daten abgelegt werden, muss den Datentyp ARRAY of BYTE oder ARRAY of CHAR haben und in der Version 1.0 mit Standardzugriff deklariert sein. Ab der Version 2.0 sind auch optimierte Daten zugelassen. Fülldaten des Datenbereichs der Quelle sind im Ziel-ARRAY undefiniert. Diese können sowohl Füll-Bytes bzw. Füll-Bits eines Datenbereichs (z. B. ARRAY, STRUCT oder PLC-Datentyp (UDT)) sowie die aktuell ungenutzten Zeichen eines Strings sein.

Die Kapazität des Standard-Speicherbereichs liegt bei 64 KB. Strukturen, die gemäß der Regeln für standard Speicherbereiche größer als 64 KB sind, können nicht serialisiert werden, wenn der Operand am Parameter DEST_ARRAY in einem standard Speicherbereich liegt.

Es ist empfehlenswert, die untere ARRAY-Grenze mit "0" zu definieren, denn dann entspricht der Index innerhalb des ARRAYs dem Wert des Parameters POS, z. B. ARRAY[0] = POS 0. Die Beschreibung und das unten stehende Beispiel sind auf dieser Grundlage aufgebaut.

Der Operand am Parameter POS enthält die Information, wie viele Bytes die umgewandelten Daten belegen.

Wenn Sie einen einzelnen PLC-Datentypen (UDT), STRUCT oder ARRAY of <Datentyp> versenden möchten, dann können Sie direkt die Anweisung "TSEND: Daten über Kommunikationsverbindung senden" aufrufen.

Größe des Speicherbereichs

Aufgrund der Ausrichtungsregeln enthalten einfache Strukturen im optimierten Speicherbereich keine Füll-Bytes. Das führt dazu, dass eine Struktur im optimierten Speicherbereich kleiner ist als im standard Speicherbereich. ARRAYs von Strukturen und Strukturen, die Strukturen enthalten, beinhalten Füll-Bytes. Im Allgemeinen kann keine Aussage getroffen werden, in welchem Speicherbereich eine zusammengesetzte Struktur mehr Platz braucht.

Gültig für CPUs der Baureihe S7-1500:

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hängt die Länge eines BOOL davon ab, welcher Datentyp folgt. Das bedeutet, wenn z. B. ein BYTE folgt, hat das BOOL eine Länge von 1 Byte. Wenn z. B. ein WORD folgt, dann hat das BOOL eine Länge von 2 Byte. Dadurch kommt es vor, dass eine Struktur, die im wesentlichen aus dem Datentyp BOOL besteht, im optimierten Speicherbereich größer ist als im standard Speicherbereich. Zusammengesetzte Strukturen mit einem geringen Anteil an BOOL-Datentypen sind im optimierten Speicherbereich kleiner als im standard Speicherbereich.

Daher ist es empfehlenswert, dass der Quelldatenbereich für das Serialisieren mit den großen Datentypen beginnt und mit booleschen Elementen endet. Dadurch wird das Auffüllen mit Füll-Bits wesentllich reduziert.

Optimierter Speicherbereich

Um größere Strukturen zu serialisieren, können Sie ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 den Speicherbereich auch mit optimiertem Zugriff deklarieren. Die sequenzielle Darstellung bleibt dabei unverändert, wie bei einem standard Speicherbereich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_VARIABLE	Input	VARIANT	D, Bausteinschnittstelle eines FB	Zeiger auf eine Variable vom Datentyp STRUCT, ARRAY oder PLC-Datentyp (UDT), die serialisiert werden soll.
DEST_ARRAY	InOut	VARIANT	D, Bausteinschnittstelle eines FB (Die Abschnitte Input, Output, Static und Temp sind möglich.)	Zeiger auf ein ARRAY of BYTE oder ARRAY of CHAR, in dem der erzeugte Datenstrom gespeichert wird.
POS	InOut	DINT	E, A, M, D, L	Der Operand am Parameter POS speichert den Index des ersten Bytes nach der Gesamtzahl der Bytes, die die umgewandelten Kundendaten belegt haben. Der Parameter POS wird nullbasiert berechnet.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B0	Die Speicherbereiche an den Parametern SRC_VARIABLE und DEST_ARRAY überschneiden sich.
8150	Der Datentyp VARIANT am Parameter SRC_VARIABLE enthält einen NULL-Zeiger.
8151	Keine gültige Referenz am Parameter SRC_VARIABLE
8236	Die Variable am Parameter SRC_ARRAY liegt nicht in einem Baustein mit Standardzugriff.
8250	Am Parameter DEST_ARRAY wurde ein NULL-Zeiger übergeben.
8251	Keine gültige Referenz am Parameter DEST_ARRAY
8253	Die Variable am Parameter DEST_ARRAY bietet für den Inhalt der Variable am Parameter SRC_VARIABLE nicht genügend Speicherplatz. Durch den Eingangswert der Variable am Parameter POS wird der zur Verfügung stehende Speicherplatz reduziert. Der Eingangswert des Parameters POS bestimmt, wo innerhalb der Variable am Parameter DEST_ARRAY begonnen wird.
8254	Ungültiger Datentyp am Parameter DEST_ARRAY

Fehler-code* (W#16#...)	Erläuterung
8382	Der Wert am Parameter POS befindet sich außerhalb der Grenzwerte des Arrays.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ab der Firmware Version >= 4.2 bei einer CPU der Baureihe S7-1200 und ab der Firmware Version >= 2.0 bei einer CPU der Baureihe S7-1500 hat der folgende Fehlercode eine andere Bedeutung:

Fehler-code* (W#16#...)	Erläuterung
8236	Der Zugriff am Parameter DEST_ARRAY auf den Speicherbereich ist ungültig.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL Serialize VARIANT
     (SRC_VARIABLE := "Source".Client
      DEST_ARRAY => "Buffer".Field
      POS := #BufferPos
      RET_VAL => #Error
      )
```

Die Anweisung serialisiert die Kundendaten aus der Variable "Source" und schreibt sie als sequenzielle Darstellung in die Variable "Buffer". Der Index des nächsten unbeschriebenen Bytes des Operanden "Buffer".Field wird im Operanden #BufferPos gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

Aufbau einer ARRAY-Variablen (Seite 302)

Aufbau einer STRUCT-Variablen (Seite 293)

Aufbau einer STRING-Variablen (Seite 279)

Grundlagen zu GRAPH (Seite 7685)

MOVE_BLK: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Parameter IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Hinweis

Verwendung von ARRAYs

Die Anweisung kopiert den Inhalt ab dem definierten Element n Elemente (n = abhängig vom Wert am Parameter COUNT) aus dem Quellbereich in den Zielbereich, beginnend ab dem angegebenen Index.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MOVE_BLK INT_UINT
      (IN := #a_array[2]
       COUNT := "Tag_Count"
       OUT => #b_array[1]
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem zweiten Element.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

MOVE_BLK_VARIANT: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können ein komplettes ARRAY oder Elemente eines ARRAYs in ein anderes ARRAY des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-ARRAY dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines ARRAYs umkopieren oder einzelne Elemente kopieren.

Die Anzahl der Elemente, die kopiert werden sollen, darf den selektierten Quellbereich bzw. Zielbereich nicht überschreiten.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss das ARRAY noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC_INDEX und DEST_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des ARRAYs.

Die Anweisung wird nicht ausgeführt, wenn mehr Daten kopiert werden, als zur Verfügung gestellt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input [2)]	VARIANT (der auf ein ARRAY oder ein einzelnes ARRAY-Element zeigt), ARRAY of <Datentyp>	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Der Quellbereich, aus dem kopiert wird
COUNT	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Elemente, die kopiert werden Geben Sie dem Parameter COUNT den Wert "1", wenn am Parameter SRC oder am Parameter DEST kein ARRAY angegeben ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC_INDEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert das erste Element, das kopiert wird: • Der Parameter SRC_INDEX wird nullbasiert berechnet. Wenn am Parameter SRC ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter SRC_INDEX das erste Element innerhalb des Quellbereichs an, aus dem kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter SRC kein Array oder nur ein einzelnes Element eines ARRAYs angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert "0".
DEST_IN-DEX	Input	DINT	E, A, M, D, L oder Konstante	Definiert den Anfang des Ziel-speicherbereichs: • Der Parameter DEST_INDEX wird nullbasiert berechnet. Wenn am Parameter DEST ein ARRAY angegeben ist, dann gibt die Ganzzahl am Parameter DEST_INDEX das erste Element innerhalb des Zielbereichs an, in das kopiert werden soll. Unabhängig von den deklarierten ARRAY-Grenzen. • Wenn am Parameter DEST kein ARRAY angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEST	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zielbereich, in den die Inhalte des Quellbereichs kopiert werden
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
1) Der Parameter DEST ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden. 2) Am Parameter SRC sind die Datentypen BOOL und ARRAY of BOOL nicht zulässig.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Die Datentypen stimmen nicht überein. Verwenden Sie statt eines ARRAY of Struct ein ARRAY of PLC-Datentyp(UDT).
8151	Der Zugriff auf den Parameter SRC ist nicht möglich.
8152	Der Operand am Parameter SRC ist nicht typisiert.
8153	Codeerzeugungsfehler am Parameter SRC
8154	Der Operand am Parameter SRC hat den Datentyp BOOL.
8281	Ungültiger Wert am Parameter COUNT
8382	Der Wert am Parameter SRC_INDEX ist kleiner NULL.
8383	Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8482	Der Wert am Parameter DEST_INDEX ist kleiner NULL.
8483	Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des ARRAYs.
8534	Der Parameter DEST ist schreibgeschützt
8551	Der Zugriff auf den Parameter DEST ist nicht möglich.
8552	Der Operand am Parameter DEST ist nicht typisiert.
8553	Codeerzeugungsfehler am Parameter DEST
8554	Der Operand am Parameter DEST hat den Datentyp BOOL.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MOVE_BLK_VARIANT
      (SRC := #SrcField
       COUNT := "Tag_Count"
       SRC_INDEX := "Tag_Src_Index"
       DEST_INDEX := "Tag_Dest_Index"
       DEST => #DestField
       RET_VAL := "Tag_Result"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Deklaration in der Bausteinschnittstelle	Operand	Wert
SRC	Input	#SrcField	Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[0..10] of "MOVE_UDT"
COUNT	Input	Tag_Count	2
SRC_INDEX	Input	Tag_Src_Index	3
DEST_INDEX	Input	Tag_Dest_Index	3
DEST	InOut	#DestField	Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekannten, PLC-Datentyp. (ARRAY[10..20] of "MOVE_UDT"

2 Elemente werden aus dem Quellbereich, beginnend mit dem vierten Element des ARRAY[0..10] of MOVE_UDT, in den Zielbereich kopiert. Die Kopien werden im ARRAY[10..20] of MOVE_UDT ab dem vierten Element eingefügt.

Hinweis

Weitere Informationen zur Anweisung MOVE_BLK_VARIANT im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel für das Verschieben von Daten (Seite 348)

Grundlagen zum VARIANT (Seite 339)

Grundlagen zu GRAPH (Seite 7685)

UMOVE_BLK: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Parameter IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Ununterbrechbar kopieren

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Wert am Ausgang OUT ist ungültig, wenn mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN [1]	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Quellbereichs, das kopiert wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden
OUT [1]	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden
[1] Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL UMOVE_BLK INT_UINT
      (IN := #a_array[2]
       COUNT := "Tag_Count"
       OUT => #b_array[1]
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	a_array[2]	Der Operand "a_array" ist vom Datentyp ARRAY[0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT.
COUNT	Tag_Count	3
OUT	b_array[1]	Der Operand "b_array" ist vom Datentyp ARRAY[0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT.

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a_array und kopiert deren Inhalte in die Ausgangsvariable #b_array, beginnend bei dem

zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Hinweis

Weitere Informationen zur Anweisung UMOVE_BLK im Siemens Industry Online Support finden Sie in folgendem Beitrag:

Wie können Sie in STEP 7 (TIA Portal) Speicherbereiche und strukturierte Daten zwischen zwei Datenbausteinen kopieren?

https://support.industry.siemens.com/cs/ww/de/view/42603881 (https://support.industry.siemens.com/cs/ww/de/view/42603881)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

FILL_BLK: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Adresse im Zielbereich, ab der befüllt wird

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

GRAPH

```
CALL FILL_BLK INT_UINT
     (IN := #FillValue
      COUNT := "Tag_Count"
      OUT => #TargetArea[1]
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element.

Beispiel mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1					Datentyp
MyStruct1					STRUCT
	Member_1				INT
	Member_2				INT
	Member_3				INT
	Member_4				INT
MyStruct2					STRUCT
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

GRAPH

```
CALL FILL_BLK INT_UINT
     (IN := 10
```

GRAPH

```
      COUNT := 2
      OUT => "Data_block_1".MyStruct1.Member_2
      )
```

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

GRAPH

```
CALL FILL_BLK INT_UINT
     (IN := 10
      COUNT := 2
      OUT => "Data_block_1".MyStruct2.SubArray[1].NestedStruct.Member_2
      )
```

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

UFILL_BLK: Bereich ununterbrechbar befüllen

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Es werden maximal so viele Elemente geändert, wie das ARRAY, bzw die Struktur, Elemente hat. Wenn Sie mehr Daten kopieren, als am Ausgang OUT Elemente zur Verfügung stehen, dann erhalten Sie ein unerwünschtes Ergebnis.

Hinweis

Verwendung von ARRAYs

Die Anweisung liest den Inhalt des gewählten Elements aus dem Quellbereich und kopiert dessen Inhalt n mal (n = abhängig vom Wert am Parameter COUNT) in den Zielbereich, beginnend ab dem angegebenen Index.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Strukturen befüllen

Neben den Elementen eines ARRAYs können Sie ebenfalls mehrere Elemente einer Struktur (STRUCT, PLC-Datentyp) mit dem gleichen Wert befüllen. Die Struktur, deren Elemente Sie befüllen wollen, darf nur einzelne Elemente vom gleichen elementaren Datentyp beinhalten. Die Struktur selbst kann aber in einer anderen Struktur eingebettet sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	E, A, M, D, L, P oder Konstante	Element, mit dem der Zielbereich befüllt wird
COUNT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L, P oder Konstante	Anzahl der Kopierwiederholungen
OUT	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, DATE, CHAR, WCHAR	D, L	Adresse im Zielbereich, ab der befüllt wird

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Das folgende Beispiel zeigt die Funktionsweise der Anweisung, wenn Sie ein ARRAY befüllen wollen:

GRAPH

```
CALL UFILL_BLK INT_UINT
      (IN := #FillValue
       COUNT := "Tag_Count"
       OUT => #TargetArea[1]
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	FillValue	Der Operand ist vom Datentyp INT.
COUNT	Tag_Count	3
OUT	TargetArea	Der Operand TargetArea ist vom Datentyp ARRAY[1..5] of INT. Er besteht aus 5 Elementen des Datentyps INT.

Die Anweisung kopiert drei Mal den Wert des Operanden #FillValue in die Ausgangsvariable #TargetArea, beginnend mit dem ersten Element. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

Beispiel mit einer Struktur

Die folgenden Beispiele zeigen die Funktionsweise der Anweisung, wenn Sie eine Struktur befüllen wollen.

Erstellen Sie einen globalen Datenbaustein mit den folgenden Elementen:

Data_block_1		Datentyp
MyStruct1		STRUCT
	Member_1	INT
	Member_2	INT
	Member_3	INT
	Member_4	INT
MyStruct2		STRUCT

Data_block_1					Datentyp
	SubArray				ARRAY[1..2] of STRUCT
		SubArray[1]			STRUCT
			NestedStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT
		SubArray[2]			STRUCT
			Nest-edStruct		STRUCT
				Member_1	INT
				Member_2	INT
				Member_3	INT
				Member_4	INT

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct1 zu adressieren:

GRAPH

```
CALL UFILL_BLK INT_UINT
    (IN := 10
     COUNT := 2
     OUT => "Data_block_1".MyStruct1.Member_2
      )
```

Erstellen Sie den folgenden Programmcode, um die Variable MyStruct2 zu adressieren:

GRAPH

```
CALL UFILL_BLK INT_UINT
     (IN := 10
      COUNT := 2
      OUT => "Data_block_1".MyStruct2.SubArray[1].NestedStruct.Member_2
      )
```

In beiden Beispielen wird der Wert 10 des Parameters IN zweimal in den Operanden am Parameter OUT kopiert, beginnend mit dem Element Member_2. Das bedeutet, dass der Wert 10 jeweils in die Elemente Member_2 und Member_3 kopiert wird. Die beiden anderen Elemente, Member_1 und Member_4, werden nicht verändert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SCATTER: Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Bitfolge in einzelne Bits zerlegen" zerlegen Sie eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Bitfolge in einzelne Bits zerlegen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der anonyme STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. ein Statuswort zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn das ARRAY nicht genügend BOOL-Elemente zur Verfügung stellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Bitfolge, die zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das bzw. den die einzelnen Bits gespeichert werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationArray	Output	ARRAY[0..15] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCATTER
    (IN := #SourceWord
     OUT => #DestinationArray
     )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationArray	Der Operand "DestinationArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationArray zugewiesen.

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

	Name	Datentyp
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceWord	Input	WORD
DestinationUDT	Output	"myBits"

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCATTER
     (IN := #SourceWord
      OUT => #DestinationUDT
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceWord	WORD (16 Bits)
OUT	DestinationUDT	Der Operand "DestinationUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, das zerlegt werden soll.

Der Operand #SourceWord vom Datentyp WORD wird in seine einzelnen Bits (16) zerlegt und den einzelnen Elementen des Operanden #DestinationUDT zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen

Beschreibung

Mit der Anweisung "Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen" zerlegen Sie ein oder mehrere Elemente eines ARRAY of BYTE, WORD, DWORD oder LWORD in einzelne Bits und speichern diese in einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen. Am Parameter COUNT_IN geben Sie an, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Das Quell-ARRAY am Parameter IN darf mehr Elemente haben, als am Parameter COUNT_IN angegeben sind. Das ARRAY of BOOL, der anonyme STRUCT oder der PLC-Datentyp muss genügend Elemente haben, um die Bits der zerlegten Bitfolgen speichern zu können. Der Zielspeicherbereich darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter IN ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter OUT ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Das ARRAY of WORD wird in 48 einzelne Bits zerlegt.

Hinweis

Wenn die untere ARRAY-Grenze des Ziel-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Auf diese Weise können Sie z. B. Statuswörter zerlegen und per Index den Status der einzelnen Bits auslesen und ändern. Mittels GATHER können Sie die Bits wieder zu einer Bitfolge zusammenfügen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Das Quell-ARRAY hat weniger Elemente als am Parameter COUNT_IN angegeben sind.
- Der Index des Ziel-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of BOOL geschrieben.
- Das ARRAY[*] of BOOL, der STRUCT oder der PLC-Datentyp stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zerlegt und in das ARRAY of BOOL, den anonymen STRUCT oder den PLC-Datentyp geschrieben. Die restlichen Bitfolgen werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, das zerlegt wird. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_IN	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Quell-ARRAYs zerlegt werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, in das/den die einzelnen Bits gespeichert werden

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[0..95] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCATTER_BLK
     (IN := #SourceArrayWord[2]
      COUNT_IN := #CounterInput
      OUT => #DestinationArrayBool[0]
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[0]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 1. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen.

Beispiel für ein Ziel-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variabel	Abschnitt	Datentyp
SourceArrayWord	Input	ARRAY[0..5] of WORD
CounterInput		UDINT
DestinationArrayBool	Output	ARRAY[-2..93] of BOOL

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCATTER_BLK
     (IN := #SourceArrayWord[2]
      COUNT_IN := #CounterInput
      OUT => #DestinationArrayBool[14]
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayWord[2]	ARRAY[0..5] of WORD (Es können 96 Bits zerlegt werden.)
COUNT_IN	CounterInput = 3	UDINT3 (3 WORDs bzw. 48 Bits sollen zerlegt werden. Das bedeutet, dass im Ziel-ARRAY mindestens 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayBool[14]	Der Operand "DestinationArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung.

Das 3., 4. und 5. WORD des Operanden #SourceArrayWord wird in seine einzelnen Bits (48) zerlegt und ab dem 16. Element den einzelnen Elementen des Operanden #DestinationArrayBool zugewiesen. Die restlichen 32 Bits werden nicht beschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einer Bitfolge zusammen. Die Bitfolge wird in einer Variable vom Datentyp BYTE, WORD, DWORD oder LWORD gespeichert.

Hinweis

Mehrdimensionales ARRAY of BOOL

Bei der Anweisung "Einzelne Bits zu einer Bitfolge zusammenfügen" ist die Nutzung eines mehrdimensionalen ARRAY of BOOL nicht erlaubt.

Hinweis

Länge des ARRAYs, des STRUCTs oder des PLC-Datentyps

Das ARRAY, der STRUCT oder der PLC-Datentyp muss genau so viele Elemente haben, wie die Bitfolge vorgibt.

Das heißt, dass z. B. beim Datentyp BYTE das ARRAY, der anonyme STRUCT oder der PLC-Datentyp genau 8 Elemente groß sein muss (WORD = 16, DWORD = 32 und LWORD = 64).

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn das ARRAY, der anonyme STRUCT oder der PLC-Datentyp weniger oder mehr BOOL-Elemente hat als die Bitfolge vorgibt. Dann werden die BOOL-Elemente nicht übertragen. Das ENO ist ebenfalls "0", wenn weniger als die notwendige Anzahl an Bits vorhanden ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp *: 8, 16, 32 oder 64 Elemente	E, A, M, D, L	ARRAY, STRUCT oder PLC-Datentyp, dessen Bits zu einer Bitfolge zusammengefügt werden. Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	BYTE, WORD, DWORD, LWORD	E, A, M, D, L	Zusammengefügte Bitfolge, gespeichert in einer Variable

Aus der Klappliste "???" der Anweisungsbox können Sie die gewünschte Bitfolge wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel mit einem ARRAY

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArray	Input	ARRAY[0..15] of BOOL
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL GATHER
    (IN := #SourceArray
     OUT => #DestinationWord
    )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArray	Der Operand "SourceArray" ist vom Datentyp ARRAY[0..15] of BOOL. Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Die Bits des Operanden #SourceArray werden zu einem WORD zusammengefügt.

Beispiel mit einem PLC-Datentyp (UDT)

Legen Sie den folgenden PLC-Datentyp "myBits" an:

		Name	Datentyp
1		GeneralError	Bool
2		DeviceError	Bool
3		CommError	Bool
4		myError1	Bool
5		myError2	Bool
6		myError3	Bool
7		myError4	Bool
8		myError5	Bool
9		myError6	Bool
10		myError7	Bool
11		myError8	Bool
12		myError9	Bool
13		myError10	Bool
14		myError11	Bool
15		myError12	Bool
16		myError13	Bool

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceUDT	Input	"myBits"
DestinationWord	Output	WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL GATHER
    (IN := #SourceUDT
     OUT => #DestinationWord
    )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceUDT	Der Operand "SourceUDT" ist vom Datentyp PLC-Datentyp (UDT). Er besteht aus 16 Elementen und ist damit genauso groß wie das WORD, in dem die Bits zusammengefügt werden sollen.
OUT	DestinationWord	WORD (16 Bits)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen

Beschreibung

Mit der Anweisung "Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen" fügen Sie die Bits aus einem ARRAY of BOOL, einem anonymen STRUCT oder einem PLC-Datentyp mit ausschließlich booleschen Elementen zu einem oder mehreren Elementen eines ARRAY of <Bitfolge> zusammen. Am Parameter COUNT_OUT geben Sie an, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Damit geben Sie auch implizit vor, wie viele Elemente des ARRAY of BOOL, des anonymen STRUCTs oder des PLC-Datentyps benötigt werden. Das Ziel-ARRAY am Parameter OUT darf mehr Elemente haben, als am Parameter COUNT_OUT angegeben sind. Das ARRAY of <Bitfolge> muss genügend Elemente haben, um die Bits, die zusammengefügt werden sollen, speichern zu können. Das Ziel-ARRAY darf aber auch größer sein.

Hinweis

Mehrdimensionales ARRAY of BOOL

Wenn das ARRAY ein mehrdimensionales ARRAY of BOOL ist, dann werden die Füll-Bits der enthaltenen Dimensionen mitgezählt, auch wenn diese nicht explizit deklariert wurden.

Beispiel 1: Ein ARRAY[1..10,0..4,1..2] of BOOL wird behandelt wie ein ARRAY[1..10,0..4,1..8] of BOOL bzw. wie ein ARRAY[0..399] of BOOL.

Beispiel 2: Am Parameter OUT ist ein ARRAY[0..5] of WORD (sourceArrayWord[2]) verschalten. Der Parameter COUNT_IN hat den Wert "3". Am Parameter IN ist ein ARRAY[0..1,0..5,0..7] of BOOL (destinationArrayBool[0,0,0]) verschalten. Sowohl das Array am Parameter IN als auch am Parameter OUT ist 96 Bit groß. Aus dem ARRAY of BOOL werden 48 einzelne Bits zusammengefügt.

Hinweis

Wenn die untere ARRAY-Grenze des Quell-ARRAYs nicht "0" ist, gilt es folgendes zu beachten:

Aus Performance Gründen sollte der Index immer an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze beginnen. Das bedeutet, dass der Index vom unteren Grenzwert des ARRAYs ausgehend, berechnet werden muss. Als Grundlage für diese Berechnung dient die folgende Formel:

Gültige Indices = Untere ARRAY-Grenze + n(Anzahl der Bitfolgen) * Anzahl der Bits der gewünschten Bitfolge

Bei einem ARRAY[-2..45] of BOOL und der Bitfolge WORD sieht die Berechnung folgendermaßen aus:

- Gültiger Index (-2) = -2 + 0 * 16
- Gültiger Index (14) = -2 + 1 * 16
- Gültiger Index (30) = -2 + 2 * 16

Ein Beispiel finden Sie unten beschrieben.

Hinweis

Verfügbarkeit der Anweisung

Die Anweisung kann bei einer CPU der Baureihe S7-1500 ab der Firmware Version 2.1 verwendet werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Index des Quell-ARRAYs beginnt nicht an einer BYTE-, WORD-, DWORD- oder LWORD-Grenze. In diesem Fall wird kein Ergebnis in das ARRAY of <Bitfolge> geschrieben.
- Das ARRAY[*] of <Bitfolge> stellt nicht die benötigte Anzahl an Elementen zur Verfügung. In diesem Falls werden so viele Bitfolgen wie möglich zusammengefügt und in das ARRAY of <Bitfolge> geschrieben. Die restlichen Bits werden nicht mehr berücksichtigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Element eines ARRAY[*] of BOOL, STRUCT oder PLC-Datentyp	E, A, M, D, L	ARRAY of BOOL, STRUCT oder PLC-Datentyp, dessen Bits zusammengefügt werden (Quell-ARRAY). Die Werte dürfen nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
COUNT_OUT	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L	Zähler, wieviele Elemente des Ziel-ARRAYs beschrieben werden sollen. Der Wert darf nicht im Peripheriebereich oder im DB eines Technologieobjekts liegen.
OUT	Output	Element eines ARRAY[*] of <Bitfolge>	E, A, M, D, L	ARRAY of <Bitfolge>, in das die Bits gespeichert werden (Ziel-ARRAY)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "0"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[0..95] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL GATHER_BLK
    (IN := #SourceArrayBool[0]
     COUNT_OUT := #CounterOutput
     OUT => #DestinationArrayWord[2]
    )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[0]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[0..95] of BOOL. Damit stehen 96 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Ab dem 1. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits in das 3. Wort, die zweiten 16 Bits in das 4. Wort und die dritten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden.

Beispiel für ein Quell-ARRAY mit dem unteren Grenzwert von "-2"

Legen Sie die folgenden Variablen in der Bausteinschnittstelle an:

Variable	Abschnitt	Datentyp
SourceArrayBool	Input	ARRAY[-2..93] of BOOL
CounterOutput		UDINT
DestinationArrayWord	Output	ARRAY[0..5] of WORD

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL GATHER_BLK
    (IN := #SourceArrayBool[14]
     COUNT_OUT := #CounterOutput
     OUT => #DestinationArrayWord[2]
    )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Datentyp
IN	SourceArrayBool[14]	Der Operand "SourceArrayBool" ist vom Datentyp ARRAY[-2..93] of BOOL. Da aber erst ab dem 16. Element begonnen wird, stehen nur 80 BOOL-Elemente zur Verfügung, die wieder zu Wörtern zusammengefasst werden können.
COUNT_OUT	CounterOutput = 3	UDINT3 (Es sollen 3 Wörter beschrieben werden. Das bedeutet, dass im Quell-ARRAY 48 Bits zur Verfügung stehen müssen.)
OUT	DestinationArrayWord[2]	Der Operand "DestinationArrayWord" ist vom Datentyp ARRAY[0..5] of WORD. Damit stehen 6 WORD-Elemente zur Verfügung.

Ab dem 16. Element des Operanden #SourceArrayBool werden 48 Bits im Operanden #DestinationArrayWord zusammengefügt. Begonnen wird dabei im Ziel-ARRAY ab dem 3. Element. Das bedeutet, dass die ersten 16 Bits des Quell-ARRAYs nicht berücksichtigt werden. Die zweiten 16. Bits werden in das 3. Wort, die dritten 16 Bits in das 4. Wort und die vierten 16 Bits in das 5. Wort des Ziel-ARRAYs geschrieben werden. Die restlichen 64 Bits des Quell-ARRAYs werden ebenfalls nicht berücksichtigt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SWAP: Anordnung ändern

Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes des Operanden.

Abhängig vom verwendeten Datentyp, können Sie Bytes im gesamten Akkumulator 1 oder nur im rechten Wort des Akkumulators 1 tauschen.

Das folgende Bild zeigt, wie die Bytes des Operanden vom Datentyp DWORD vertauscht werden:

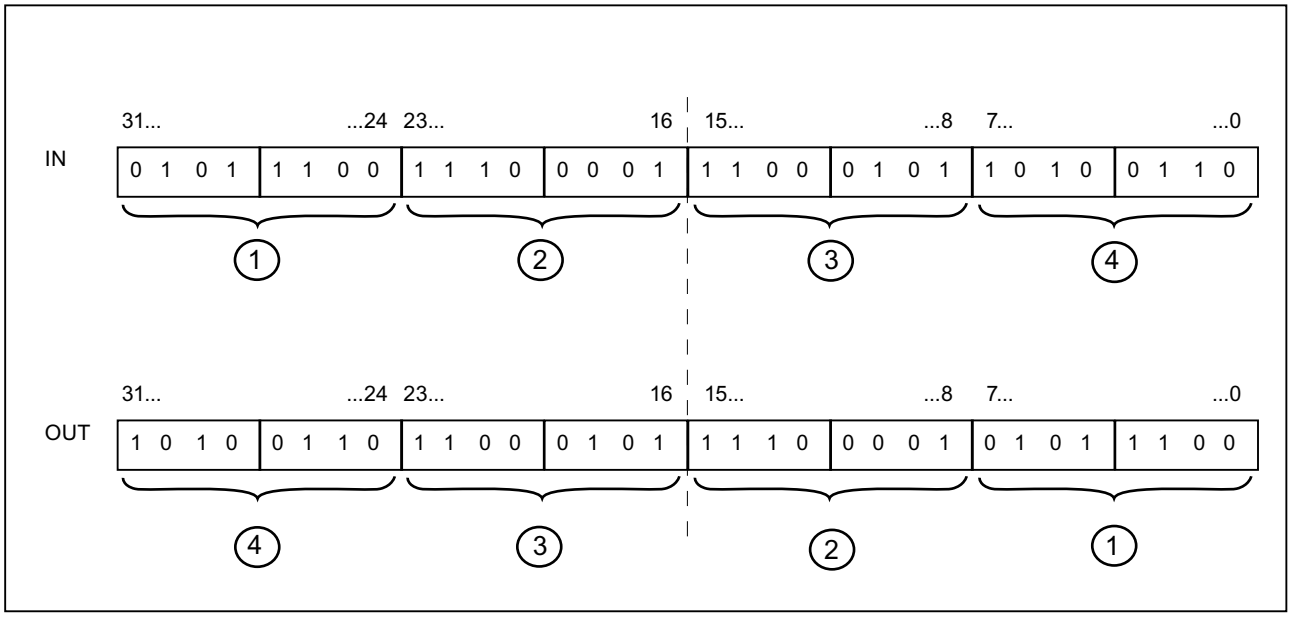

Anordnung von Bytes im rechten Wort des Akkumulators 1

Beim Datentyp WORD ändern Sie die Anordnung der Bytes im rechten Wort des Akkumulators 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert B	Wert D	Wert C

Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Die Bytes im linken Wort des Akkumulators 1 werden durch die Anweisung nicht beeinflusst und bleiben unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	WORD, DWORD, LWORD	E, A, M, D, L oder Konstante	Operand, dessen Bytes vertauscht werden
<Ergebnis>	Output	WORD, DWORD, LWORD	E, A, M, D, L	Ergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SWAP("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_InValue	0000	1111	0000	1111
Tag_OutValue	0000	1111	1111	0000

Anordnung von Bytes im gesamten Akkumulator 1 ändern:

Bei Datentyp DWORD ändern Sie die Anordnung der Bytes im gesamten Akkumulator 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert D	Wert C	Wert B	Wert A

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_InValue	1111	0000	0000	1111	0000	0000	1111	1111
Tag_OutValue	1111	1111	0000	0000	0000	1111	1111	0000

Die Anordnung der Bytes im Operanden "Tag_InValue" wird vertauscht im Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ARRAY-DB

ReadFromArrayDB: Aus Array-Datenbaustein lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie Daten aus einem Datenbaustein vom Bausteintyp ARRAY-DB und schreiben sie in einen Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of [Datentyp] besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Output [1)]	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der gelesen und ausgegeben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
1) Der Parameter VALUE ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.
8132	Der Datenbaustein existiert nicht, ist zu kurz, schreibgeschützt oder liegt im Ladespeicher.

Fehler-code* (W#16#...)	Erläuterung
8135	Der ARRAY-Datenbaustein enthält ungültige Werte.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8452	Codeerzeugungsfehler
8453	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL ReadFromArrayDB
     (DB := "ArrayDB"
      INDEX := 2
      VALUE => "TargetField"
      RET_VAL => "TagResult"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.

Aus dem "ArrayDB" wird das Element ausgelesen und in den Operanden "TargetField" geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu GRAPH (Seite 7685)

WriteToArrayDB: In Array-Datenbaustein schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie Daten in einen Datenbaustein vom Bausteintyp ARRAY-DB.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of [Datentyp] besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB	Input	DB_ANY	E, A, M, D, L	Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE	Input	VARIANT	L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Wert, der geschrieben wird
RET_VAL	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
80B5	Der Kopiervorgang wurde abgebrochen.

Fehler-code* (W#16#...)	Erläuterung
8132	Der Datenbaustein existiert nicht, ist zu kurz oder liegt im Ladespeicher.
8134	Der Datenbaustein ist schreibgeschützt.
8135	Der Datenbaustein ist kein ARRAY-Datenbaustein.
8154	Der Datenbaustein hat den falschen Datentyp.
8282	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8350	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8352	Codeerzeugungsfehler
8353	Es gibt zwei mögliche Fehlerursachen: • Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein. • Die beiden Variablen liegen nicht in einem Speicherbereich mit optimiertem Zugriff. Weitere Informationen zu den Speicherbereichzugriffsarten finden Sie hier: Grundlagen zum Bausteinzugriff (Seite 55)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL WriteToArrayDB
     (DB := "ArrayDB"
      INDEX := 2
      VALUE := "SourceField"
      RET_VAL => "TagResult"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp Array [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.

Der Wert des Operanden "SourceField" wird in das 2. Element des ARRAY-DB geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Grundlagen zu GRAPH (Seite 7685)

ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie Daten aus einem Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher und schreiben sie in einen Zielbereich.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of [Datentyp] besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Auslesen des ARRAY-DBs beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, aus dem gelesen wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, das gelesen wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALUE [1]	InOut	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Konstanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Auslesen des ARRAY-DBs ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8750	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL ReadFromArrayDBL DB_ANY, "ReadFromArrayDBL_DB"
     (REQ := "TagReq"
      DB := "ArrayDB"
      INDEX := 2
      VALUE := "TargetField"
      BUSY => "TagBusy"
      DONE => "TagDone"
      ERROR => "TagError"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	TargetField	Der Operand "TargetField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Aus dem "ArrayDB" wird das 2. Element ausgelesen und am Parameter VALUE ausgegeben. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird nicht mehr verändert. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu GRAPH (Seite 7685)

WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie Daten in einen Datenbaustein vom Bausteintyp ARRAY-DB im Ladespeicher.

Ein ARRAY-Datenbaustein ist ein Datenbaustein, der genau aus einem ARRAY of [Datentyp] besteht. Die Elemente des ARRAYs können vom Datentyp PLC-Datentyp oder jedes anderen elementaren Datentyps sein. Die Zählung des ARRAYs beginnt immer mit dem unteren Grenzwert "0".

Wenn der ARRAY-Datenbaustein mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = "1": Mit dem Schreiben in den Array-DB beginnen
DB [1)]	Input	DB_ANY	E, A, M, D, L	ARRAY-Datenbaustein, in den geschrieben wird
INDEX	Input	DINT	E, A, M, D, L, P oder Konstante	Element im DB, in das geschrieben wird. Die Angabe kann eine Konstante, eine globale Variable oder ein indizierter Wert sein.
VALUE [1)]	Input	VARIANT	D (Element eines globalen Datenbausteins) L (Die Deklaration ist in den Abschnitten "Input", "InOut" und "Temp" der Bausteinschnittstelle möglich.)	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird und dessen Wert geschrieben wird. Es dürfen keine lokalen Kontanten oder Variablen aus dem Abschnitt TEMP verwendet werden.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen
DONE	Output	BOOL	E, A, M, D, L	DONE = "1": Die Anweisung wurde erfolgreich ausgeführt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter ERROR wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
[1] Die Datenbausteine müssen mit der Bausteineigenschaft "optimiert" angelegt sein.				

Parameter ERROR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERROR:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B4	Der im ARRAY-Datenbaustein gespeicherte Element-Datentyp stimmt nicht mit dem im VARIANT übergegebenen Element-Datentyp überein.
8230	Die Datenbausteinnummer ist falsch.
8231	Der Datenbaustein existiert nicht.
8232	Der Datenbaustein ist zu kurz oder liegt nicht im Ladespeicher.
8234	Der Datenbaustein ist schreibgeschützt.
8235	Der Datenbaustein ist kein ARRAY-DB.
8254	Der Datenbaustein hat den falschen Datentyp.
8382	Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des ARRAYs.
8450	Der Datentyp VARIANT am Parameter VALUE liefert den Wert "0".
8751	Codeerzeugungsfehler
8752	Codeerzeugungsfehler
8753	Die Größe des Parameters VALUE stimmt nicht mit der Elementlänge im ARRAY-Datenbaustein überein.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Die Beschreibung der Fehlercodes, die von den Anweisungen "READ_DBL" und "WRIT_DBL" ausgelöst werden, finden Sie in den jeweiligen Anweisungsbeschreibungen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL WriteToArrayDBL DB_ANY, "WriteToArrayDBL_DB"
     (REQ := "TagReg"
      DB := "ArrayDB"
      INDEX := 2
      VALUE := "SourceField"
      BUSY => "TagBusy"
```

GRAPH

```
DONE => "TagDone"
ERROR => "TagError"
)
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
REQ	TagReq	BOOL
DB	ArrayDB	Der Operand "ArrayDB" ist ein ARRAY-DB vom Datentyp ARRAY [0..10] of INT.
INDEX	2	2. Element des "ArrayDB"
VALUE	SourceField	Der Operand "SourceField" ist eine globale Variable vom Datentyp INT.
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Wenn am Operand "TagReq" eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Sobald am Operanden "TagBusy" eine negative Signalflanke erfasst wird, wird die Anweisung beendet und der Wert am Parameter VALUE wird in das 2. Element des "ArrayDB" geschrieben. Nach der Bearbeitung der Anweisung hat der Operand "TagDone" den Signalzustand TRUE.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Grundlagen zu GRAPH (Seite 7685)

Legacy

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

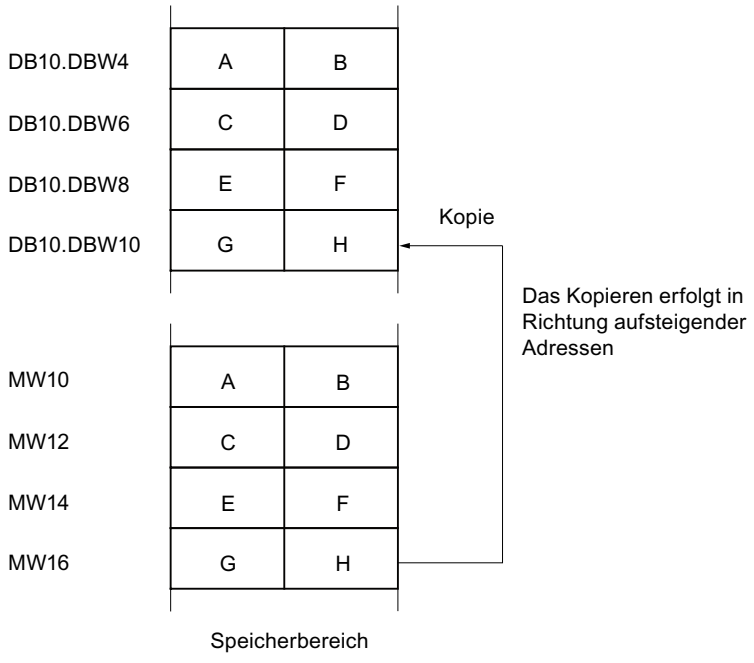

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output	VARIANT [1)]	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BLKMOV VARIANT
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_ErrorCode" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu GRAPH (Seite 7685)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter SRCBLK werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
8352	Am Parameter DSTBLK werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL UBLKMOV VARIANT
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_ErrorCode" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu GRAPH (Seite 7685)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

Hinweis

Die Variablen der Anweisung können Sie nur innerhalb von Speicherbereichen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht aktiviert ist. Das gilt für Datenbausteine (DBs), Organisationsbausteine (OBs), Funktionen (FCs), Merker (M), Eingänge (E) und Ausgänge (A).

Wenn eine Variable der Anweisung jedoch mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie diese Variable auch in Speicherbereichen "mit optimiertem Bausteinzugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

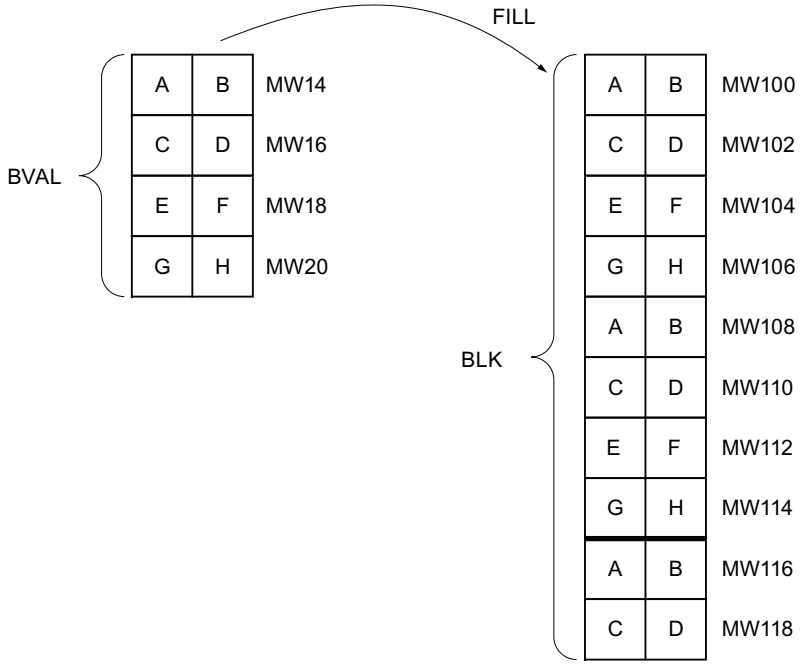

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1)]	VARIANT	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, da die Daten in die Variable fließen. Die Variable selbst muss in der Bausteinschnittstelle aber als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8152	Am Parameter BVAL werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
8352	Am Parameter BLK werden die Datentypen WSTRING, WCHAR, BOOL und ARRAY of STRING nicht unterstützt.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL FILL VARIANT
      (BVAL := P#M14.0 WORD 4
       RET_VAL => "Tag_ErrorCode"
       BLK => P#M100.0 WORD 10
       )
```

Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu GRAPH (Seite 7685)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Operanden und konvertieren ihn entsprechend der parametrierten Datentypen.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
<Ergebnis>	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, CHAR, WCHAR, BCD16, BCD32	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH
```
"Tag_OutValue" := INT_TO_DINT("Tag_InValue")
```

Der Inhalt des Operanden "Tag_InValue" wird gelesen und in eine Ganzzahl (32 Bit) umgewandelt. Das Ergebnis wird im Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

Explizite Konvertierungen (Seite 423)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Operanden zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in die am nächsten liegende Ganzzahl um. Wenn der Eingangswert einer geraden und einer ungeraden Zahl liegt, wird die gerade Zahl gewählt.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird.
<Ergebnis>	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Rundung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH
```
"Tag_OutValue" := ROUND("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	2	-2

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächstliegenden geraden Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Operanden zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis kann dabei größer oder gleich dem Eingangswert sein.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert als Gleitpunkt-zahl
<Ergebnis>	Output	Ganzzahlen, Gleit-punktzahlen	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := CEIL("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	1	0

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächst höheren Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Operanden zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis kann dabei kleiner oder gleich dem Eingangswert sein.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert als Gleitpunktzahl
<Ergebnis>	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := FLOOR("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	0	-1

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächst niedrigeren Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Operanden eine Ganzzahl. Der Wert wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen als Erg ebnis aus.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Eingangwert als Gleitpunkt-zahl
<Ergebnis>	Output	Ganzzahlen, Gleit-punktzahlen	E, A, M, D, L	Ergebnis mit ganzzahligem Anteil der Gleitpunktzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := TRUNC("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	1	-1

Der ganzzahlige Anteil der Gleitpunktzahl am Operanden "Tag_InValue" wird in eine Ganzzahl umgewandelt und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SCALE_X: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Parameter VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung wird der Gleitpunktwert am Parameter VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang OUT abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:

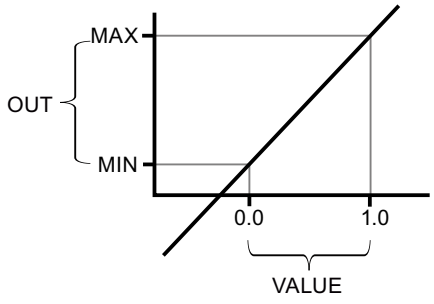

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [VALUE * (MAX – MIN)] + MIN

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert am Parameter MIN ist größer oder gleich dem Wert am Parameter MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Parameter VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Wert, der skaliert wird. Bei Angabe einer Konstante müssen Sie diese deklarieren.
MAX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs
RET_VAL	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Skalierung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCALE_X INT_REAL
     (MIN := "Tag_Minimum"
      VALUE := "Tag_Value"
      MAX := "Tag_Maximum"
      RET_VAL => "Tag_OutputValue"
      )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_Minimum	10
VALUE	Tag_Value	0.5
MAX	Tag_Maximum	30
RET_VAL	Tag_ReturnValue	20

Der Wert am Parameter "Tag_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Parametern "Tag_Minimum" und "Tag_Maximum" definiert wurde. Das Ergebnis wird am Parameter "Tag_ReturnValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

NORM_X: Normieren

Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert am Parameter VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Parameter OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Parameter MIN ist, liefert der Parameter OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem Wert am Parameter MAX ist, liefert der Parameter OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:

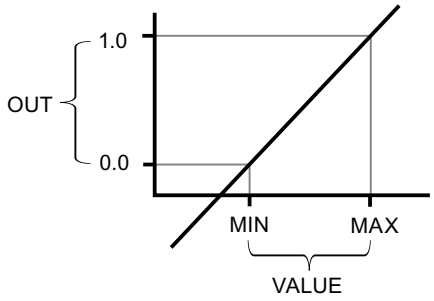

Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

OUT = (VALUE – MIN) / (MAX – MIN)

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert am Parameter MIN ist größer oder gleich dem Wert am Parameter MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Parameter VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

Hinweis

Weitere Informationen zur Umwandlung von Analogwerten finden Sie im jeweiligen Gerätehandbuch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MIN [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Untere Grenze des Wertebereichs
VALUE [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Wert, der normiert wird
MAX [1)]	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L oder Konstante	Obere Grenze des Wertebereichs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	Gleitpunktzahlen	E, A, M, D, L	Ergebnis der Normierung
[1] Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren.				

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL NORM_X INT_REAL
      (MIN := "Tag_Minimum"
       VALUE := "Tag_Value"
       MAX := "Tag_Maximum"
       RET_VAL => "Tag_OutputValue"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MIN	Tag_Minimum	10
VALUE	Tag_Value	20
MAX	Tag_Maximum	30
RET_VAL	Tag_ReturnValue	0.5

Der Wert am Parameter "Tag_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Parametern "Tag_Minimum" und "Tag_Maximum" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Parameter "Tag_Value" normiert. Das Ergebnis wird am Parameter "Tag_ReturnValue" als Gleitpunktzahl abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

Legacy

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27 648 und 27 648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27 648,0" und die Konstante "K2" den Wert "+27 648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27 648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27 648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCALE
     (IN := "Tag_InputValue"
      HI_LIM := "Tag_HighLimit"
      LO_LIM := "Tag_LowLimit"
      BIPOLAR := "Tag_Bipolar"
      RET_VAL => "Tag_ErrorCode"
      OUT => "Tag_OutputValue"
      )
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Am Operanden "Tag_InputValue" ist der Wert angegeben, der umgewandelt und skaliert werden soll. Der obere und der untere Grenzwert ist durch die Operanden "Tag_HighLimit" und "Tag_LowLimit" definiert. Mithilfe des Operanden "Tag_Bipolar" = TRUE geben Sie an, dass der Wert am Parameter IN als bipolar interpretiert werden soll. Am Operanden "Tag_OutputValue" wird das Ergebnis der Anweisung ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27 648 und 27 648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27 648,0" und die Konstante "K2" den Wert "+27 648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27 648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27 648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL UNSCALE
      (IN := "Tag_InputValue"
       HI_LIM := "Tag_HighLimit"
       LO_LIM := "Tag_LowLimit"
       BIPOLAR := "Tag_Bipolar"
       RET_VAL => "Tag_ErrorCode"
       OUT => "Tag_OutputValue"
       )
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Am Operanden "Tag_InputValue" ist der Wert angegeben, der umgewandelt und deskaliert werden soll. Der obere und der untere Grenzwert ist durch die Operanden "Tag_HighLimit" und "Tag_LowLimit" definiert. Mithilfe des Operanden "Tag_Bipolar" = TRUE geben Sie an, dass der Wert am Parameter IN als bipolar interpretiert werden soll. Am Operanden "Tag_OutputValue" wird das Ergebnis der Anweisung ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu GRAPH (Seite 7685)

Programmsteuerung

Laufzeitsteuerung

ENDIS_PW: Passwort-Legitimierung einschränken und freigeben

Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU projektierte Passwörter legitimiert sind oder nicht. Dadurch können Sie legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist.

Wenn Sie die Anweisung aufrufen und der Parameter REQ den Signalzustand "0" hat, wird der aktuell eingestellte Zustand an den Ausgangsparametern angezeigt. Falls Änderungen an den Eingangsparametern vorgenommen wurden, werden diese nicht an die Ausgangsparameter übergeben.

Wenn die Anweisung aufgerufen wird und der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F_PWD, FULL_PWD, R_PWD, HMI_PWD) übernommen:

- Wenn der Signalzustand "0" vorliegt, ist die Legitimation per Passwort nicht erlaubt.
- Wenn der Signalzustand "1" vorliegt, ist das Passwort verwendbar.

Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer dem Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Nicht konfigurierte Passwörter müssen am Eingang den Signalzustand TRUE haben und liefern den Signalzustand TRUE am Ausgang. Das Failsafe Passwort ist nur für eine F-CPU

parametrierbar und muss daher in einer Standard CPU immer mit dem Signalzustand TRUE verschaltet werden. Wenn die Anweisung einen Fehler zurückliefert, dann bleibt der Aufruf ohne Auswirkung, d. h. die vorherige Sperre ist unverändert wirksam.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken einer leeren Speicherkarte (Übertragungskarte oder Programmkarte) in eine S7-1200-CPU.
- Der Übergang von NETZ AUS - NETZ EIN deaktiviert in der S7-1200-CPU den Schutz. Die Anweisung "Passwort-Legitimierung einschränken und freigeben" muss im Programm erneut (z. B. im Anlauf-OB) aufgerufen werden.

Hinweis

Wenn das HMI Passwort nicht freigegeben ist, dann sperrt die Anweisung "Passwort-Legitimierung einschränken und freigeben" den Zugriff von HMI-Systemen.

Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

Ungewolltes Aussperren bei einer S7-1500-CPU verhindern

Die Einstellungen können im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Ungewolltes Aussperren bei einer S7-1200-CPU verhindern

Da es bei einer S7-1200-CPU keinen Betriebsartenschalter gibt, wird der Schutz bei NETZ AUS - NETZ EIN deaktiviert. Damit ist es möglich und empfehlenswert, ein ungewolltes Aussperren mithilfe bestimmter Programmsequenzen innerhalb Ihres Programms zu verhindern.

Programmieren Sie dazu eine Zeitsteuerung entweder über einen Weckalarm-OB oder einen Timer im Main OB (OB 1). Dadurch haben Sie die Möglichkeit, nach einem Übergang von

NETZ AUS - NETZ EIN und der damit verbundenen Deaktivierung des Schutzes, die Anweisung "Passwort-Legitimierung einschränken und freigeben" in dem entsprechenden OB (z. B. OB 1 oder OB 35) zeitnah wieder aufzurufen. Um das Zeitfenster, in dem die Anweisung nicht aktiv ist und damit keine Einschränkungen bei der Passwort-Legitimierung bestehen, möglichst klein zu halten, rufen Sie die Anweisung im Anlauf-OB (OB 100) auf. Diese Vorgehensweise bietet Ihnen den größtmöglichen Schutz vor unbefugtem Zugriff.

Wenn es zu einer ungewollten Aussperrung gekommen ist, dann können Sie den Aufruf im Anlauf-OB überspringen (z. B. durch die Abfrage eines Eingangsparameters) und haben die eingestellte Zeit (z. B. 10 Sekunden bis 1 Minute), um eine Verbindung zur CPU aufzubauen, bevor die Sperre wieder aktiv ist.

Wenn Sie in Ihrem Programmcode keinen Timer vorgesehen haben und es zu einer Aussperrung kam, dann stecken Sie eine leere Übertragungskarte oder eine leere Programmkarte in die CPU. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Danach müssen Sie das Anwenderprogramm aus STEP 7 erneut in die CPU laden.

Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte oder Programmkarte. Die leere Übertragungskarte oder Programmkarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

⚠ WARNUNG

Leere Übertragungskarte stecken

Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Nach dem Ziehen der Übertragungskarte ist der Inhalt der Übertragungskarte im internen Ladespeicher. Achten Sie an dieser Stelle darauf, dass die Karte kein Programm enthält.

⚠ WARNUNG

Leere Programmkarte stecken

Wenn Sie eine Programmkarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann.

Achten Sie darauf, dass die Programmkarte leer ist. Der interne Ladespeicher wird auf die leere Programmkarte kopiert. Nach dem Ziehen der ehemals leeren Programmkarte ist der interne Ladespeicher leer.

Sie müssen die Übertragungskarte oder Programmkarte ziehen, bevor Sie die CPU in RUN versetzen.

Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

Aktion	Passwortschutz durch die Anweisung
Grundzustand nach • Betriebsartenschalter auf STOP • Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control) • Zurücksetzen auf Werkseinstellung	Nicht aktiviert (Keine Einschränkungen)
Grundzustand nach NETZ-EIN	• S7-1200-CPU: Die Sperre ist deaktiviert und die Anweisung muss im Programm erneut aufgerufen werden (z. B. im Anlauf-OB). • S7-1500-CPU: Aktiviert (wenn vor NETZ AUS eine Sperre aktiviert war). Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent.
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT	Aktiviert Die Passwörter dürfen weiterhin nicht verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.
F_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff inklusive Failsafe • F_PWD = "0": Passwort nicht erlauben • F_PWD = "1": Passwort erlauben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FULL_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lese-/Schreibzugriff • FULL_PWD = "0": Passwort nicht erlauben • FULL_PWD = "1": Passwort erlauben
R_PWD	Input	BOOL	E, A, M, D, L oder Konstante	Lesezugriff • R_PWD = "0": Passwort nicht erlauben • R_PWD = "1": Passwort erlauben
HMI_PWD	Input	BOOL	E, A, M, D, L oder Konstante	HMI-Zugriff • HMI_PWD = "0": Passwort nicht erlauben • HMI_PWD = "1": Passwort erlauben
F_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff inklusive Failsafe • F_PWD_ON = "0": Passwort nicht erlaubt • F_PWD_ON = "1": Passwort erlaubt
FULL_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lese-/Schreibzugriff • FULL_PWD_ON = "0": Passwort nicht erlaubt • FULL_PWD_ON = "1": Passwort erlaubt
R_PWD_ON	Output	BOOL	E, A, M, D, L	Status Lesezugriff • R_PWD_ON = "0": Passwort nicht erlaubt • R_PWD_ON = "1": Passwort erlaubt
HMI_PWD_ON	Output	BOOL	E, A, M, D, L	Status HMI-Zugriff • HMI_PWD_ON = "0": Passwort nicht erlaubt • HMI_PWD_ON = "1": Passwort erlaubt
RET_VAL	Output	WORD	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8090	Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt
80D0	Das Passwort für Failsafe ist nicht konfiguriert. Bei Standard-CPUs muss der Signalzustand TRUE sein.
80D1	Der Lese-/Schreibzugriff ist nicht konfiguriert
80D2	Der Lesezugriff ist nicht konfiguriert
80D3	Der HMI-Zugriff ist nicht konfiguriert
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Wenn Sie die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufrufen, dann wird sie nicht ausgeführt und der Freigabeausgang ENO liefert den Signalzustand "0".

Die Anweisung läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Aufruf der Anweisung

Sie können die Anweisung unabhängig von der Priorität in allen Organisationsbausteinen aufrufen.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

GET_ERROR: Fehler lokal abfragen

Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Programmbausteins ab. Dabei handelt es sich in der Regel um Programmier- oder Zugriffsfehler. Wenn das System während der Programmbausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler, innerhalb der Ausführung des Programmbausteins seit der letzten Ausführung der Anweisung, eine ausführliche Information im Operanden am Ausgang OUT gespeichert.

Am Ausgang OUT können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Treten im Programmbaustein mehrere Fehler auf, dann wird erst nach der Behebung des

ersten aufgetretenen Fehlers die Fehlerinformation zum nächsten aufgetretenen Fehler von der Anweisung ausgegeben.

Hinweis

Ausgang OUT

Der Ausgang OUT wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.

Hinweis

Aktivierung der lokalen Fehlerbehandlung

Sobald Sie die Anweisung in den Programmcode eines Programmbausteins einfügen, wird die lokale Fehlerbehandlung aktiviert und voreingestellte Systemreaktionen werden beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	ErrorStruct	D, L	Fehlerinformationen

Datentyp "ErrorStruct"

Der Datentyp "ErrorStruct" kann in einen globalen Datenbaustein oder in die Bausteinschnittstelle eingefügt werden. Sie können den Datentyp auch mehrfach einfügen, wenn Sie jedes Mal einen anderen Namen für die Datenstruktur vergeben. Die Datenstruktur und der Name einzelner Strukturelemente sind nicht änderbar. Wenn Sie die Fehlerinformation in einem globalen Datenbaustein speichern, kann sie auch von anderen Programmbausteinen ausgelesen werden.

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

Strukturkomponente		Datentyp	Beschreibung
ERROR_ID		WORD	Fehler-ID
FLAGS		BYTE	Zeigt, ob der Fehler während eines Programmbausteinaufrufs aufgetreten ist. 16#01: Fehler während eines Programmbausteinaufrufs 16#00: Kein Fehler während eines Programmbausteinaufrufs
REACTION		BYTE	Voreingestellte Reaktion: 0: Ignorieren (Schreibfehler) 1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler) 2: Anweisung überspringen (Systemfehler)
CODE_ADDRESS		CREF	Informationen zur Adresse und Art des Programmbausteins
	BLOCK_TYPE	BYTE	Programmbausteintyp, in dem der Fehler aufgetreten ist: 1: Organisationsbaustein (OB) 2: Funktion (FC) 3: Funktionsbaustein (FB)
	CB_NUMBER	UINT	Nummer des Codebausteins
	OFFSET	UDINT	Verweis auf den internen Speicher
MODE		BYTE	Informationen zur Adresse eines Operanden
OPERAND_NUMBER		UINT	Operandennummer des Maschinenbefehls
POINTER_NUMBER_LOCATION		UINT	(A) Interner Zeiger
SLOT_NUMBER_SCOPE		UINT	(B) Ablagebereich im internen Speicher
DATA_ADDRESS		NREF	Informationen zur Adresse eines Operanden
	AREA	BYTE	(C) Speicherbereich: L: 16#40...16#7F, 16#86, 16#87, 16#8E, 16#8F, 16#C0...16#FF E: 16#81 A: 16#82 M: 16#83 DB: 16#40, 16#84, 16#85, 16#8A, 16#8B PE: 16#01 PA: 16#02 Technologische Objekte: 16#04
	DB_NUMBER	UINT	(D) Nummer des Datenbausteins
	OFFSET	UDINT	(E) Relative Adresse des Operanden

Strukturkomponente "ERROR_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR_ID" ausgegeben werden können:

ID* (Hexadezimal)	ID* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2534	9524	Bausteinnummernfehler FC
2535	9525	Bausteinnummernfehler FB
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu GRAPH (Seite 7685)

GET_ERR_ID: Fehler-ID lokal abfragen

Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Dabei handelt es sich in der Regel um Zugriffsfehler. Wenn das System während der Bausteinbearbeitung Fehler innerhalb der Ausführung des Bausteins seit der letzten Ausführung der Anweisung meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang RET_VAL gespeichert.

Am Ausgang RET_VAL können nur Operanden vom Datentyp WORD angegeben werden. Treten im Baustein mehrere Fehler auf, dann wird erst nach der Behebung des ersten aufgetretenen Fehlers die Fehler-ID des nächsten aufgetretenen Fehlers von der Anweisung ausgegeben.

Hinweis

Der Ausgang RET_VAL wird nur verändert, wenn eine Fehlerinformation vorliegt. Um den Ausgang nach der Fehlerbehandlung wieder auf "0" zu setzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Deklarieren Sie die Variable im Abschnitt "Temp" der Bausteinschnittstelle.
- Setzen Sie die Variable vor dem Aufruf der Anweisung auf "0" zurück.

Der Ausgang der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn eine Fehlerinformation vorliegt. Wenn diese Bedingung nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Ein Beispiel, wie Sie die Anweisung auch in Kombination mit weiteren Fehlerbehandlungsmöglichkeiten realisieren können, finden Sie unter "Siehe auch".

Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

Möglichkeiten der Fehlerbehandlung

Einen Überblick über die Möglichkeiten der Fehlerbehandlung finden Sie hier: Mechanismen zur Fehlerbehandlung im Überblick (Seite 146)

Ein ausführliches Beispiel zu einer lokalen Fehlerbehandlung, die mehrere Fehlerbehandlungsmöglichkeiten beinhaltet, finden Sie hier: Beispiel für die Behandlung von Programmablauffehlern (Seite 177)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	WORD	E, A, M, D, L	Fehler-ID

Parameter RET_VAL

Die folgende Tabelle zeigt die Werte, die am Parameter RET_VAL ausgegeben werden können:

Fehlercode* (Hexadezimal)	Fehlercode* (Dezimal)	Beschreibung
0	0	Kein Fehler
2503	9475	Ungültiger Zeiger
2520	9504	Ungültiger STRING
2522	9506	Lesefehler: Operand außerhalb des gültigen Bereichs
2523	9507	Schreibfehler: Operand außerhalb des gültigen Bereichs
2524	9508	Lesefehler: Ungültiger Operand
2525	9509	Schreibfehler: Ungültiger Operand
2528	9512	Lesefehler: Datenausrichtung
2529	9513	Schreibfehler: Datenausrichtung
252C	9516	Ungültiger Zeiger
2530	9520	Schreibfehler: Datenbaustein
2533	9523	Unzulässig verwendete Referenz
2534	9524	Bausteinnummernfehler FC
2535	9525	Bausteinnummernfehler FB
2538	9528	Zugriffsfehler: DB nicht vorhanden
2539	9529	Zugriffsfehler: Falscher DB verwendet
253A	9530	Globaler Datenbaustein existiert nicht
253C	9532	Fehlerhafte Angabe oder die Funktion existiert nicht
253D	9533	Systemfunktion existiert nicht
253E	9534	Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht
253F	9535	Systembaustein existiert nicht
2550	9552	Zugriffsfehler: DB nicht vorhanden
2551	9553	Zugriffsfehler: Falscher DB verwendet
2575	9589	Fehler in der Programmschachtelungstiefe
2576	9590	Fehler in der Lokaldatenverteilung
2577	9591	Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert.
25A0	9632	Interner Fehler im TP

Fehlercode* (Hexadezimal)	Fehlercode* (Dezimal)	Beschreibung
25A1	9633	Variable ist schreibgeschützt
25A2	9634	Ungültiger Zahlenwert der Variable
2942	10562	Lesefehler: Eingang
2943	10563	Schreibfehler: Ausgang
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Verwendung der Anweisungen GET_ERROR und GET_ERR_ID (Seite 176)

Grundlagen zu GRAPH (Seite 7685)

INIT_RD: Alle remanenten Daten zurücksetzen

Beschreibung

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
80B5	Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde.
allgemeine Fehlerinformation	Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL INIT_RD
     (REQ := "Tag_REQ"
      RET_VAL => "Tag_RET_VAL"
      )
```

Wenn der Operand "Tag_REQ" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2105)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine parametrierte Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) parametrieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Beispiel zum Einfluss der geplanten Verzögerungszeit

Im folgenden Beispiel sehen Sie den Einfluss der Verzögerungszeit der Anweisung "WAIT" in verschiedenen Szenarien.

Das folgende Bild stellt die Szenarien schematisch dar:

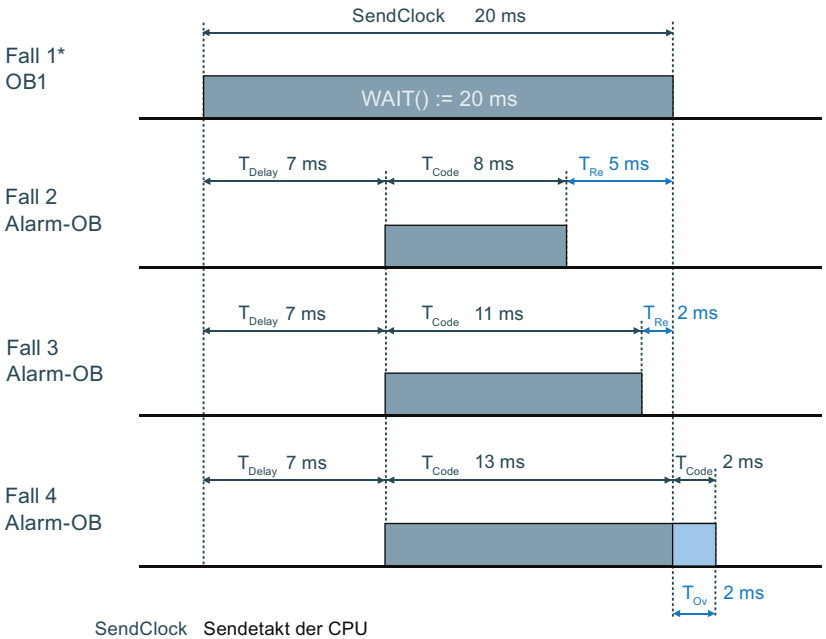

SendClock	Sendetakt der CPU
T_{Delay}	Verzögerungszeit des Alarm-OBs
T_{Code}	Zeit der Programmausführung
T_{Re}	Verbleibende Zeit
T_{Ov}	Zeit der Überlänge, um welche sich der Sendetakt erhöht

* Der OB1 mit der Anweisung ”WAIT” ist ebenso Teil von Fall 2, 3 und 4 und muss mit berücksichtigt werden.

Verbleibende Zeit = Die Spanne vom Ende der geplanten Verzögerungszeit (via "WAIT") bis zur Beendigung des Alarm-OBs

Zeit der Überlänge = Die Spanne von der Beendigung des Alarm-OBs bis zum Ende der geplanten Verzögerungszeit (via "WAIT").

Fall 1:

Die Anweisung "WAIT" wird in einem OB1 aufgerufen. Die Anweisung "WAIT" kann von höher prioren OBs oder höher prioren Prozessen (z.B. System Threads) unterbrochen werden. Die Verzögerungszeit der Anweisung "WAIT" ist aber weder verändert noch aufgeschoben.

Fall 2 und Fall 3:

Die Verarbeitung des Programms im OB1 wird nach einer Zeitverzögerung von 20 ms fortgesetzt. Diese Zeitverzögerung wird durch den Aufruf der Anweisung "WAIT" im OB1 berechnet (siehe OB1 mit WAIT). Innerhalb dieser 20 ms kann ein Alarm-OB seinen eigenen Programmcode ausführen. Der Sendetakt der CPU wird nicht verändert.

Fall 4:

Die Verarbeitung des Programms im OB1 wird nach Beendingung des höher prioren Prozesses fortgesetzt. Die 20 ms Verzögerungszeit im OB1 sind verstrichen, aber der höher priore Prozess muss noch beendet werden. Der Sendetakt der CPU wird erhöht.

Hinweis

Ausführungsreihenfolge von System- oder Kommunikationsprozessen (System Threads)

Die System Threads verwenden meistens die Priorität "15". Ebenso gibt es System Threads mit einer höheren Priorität als "26", doch diese Prozesse haben eine geringere CPU-Auslastung. System Threads werden im Bild nicht dargestellt.

Laufzeitmessung des OB1 mit Hilfe der Anweisung "RT_INFO":

Fall 2: 20 ms - 8 ms - System Threads = <12 ms. Sendetakt: ~20 ms.

Fall 3: 20 ms - 11 ms - System Threads - <9 ms. Sendetakt: ~20 ms.

Fall 4: 20 ms - 15 ms - System Threads - <7 ms. Sendetakt: ~22 ms.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

Wortverknüpfungen

NOT (INV): Invertieren

Beschreibung

Mit der Anweisung "Invertieren" invertieren Sie den Signalzustand der Bits des Operanden. Bei der Bearbeitung der Anweisung wird der Wert des Operanden mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und als Ergebnis ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L, P	Einerkomplement des Operandenwertes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"TagOut_Value" := NOT("TagIn_Value")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
TagIn_Value	W#16#000F
TagOut_Value	W#16#FFF0

Die Anweisung kehrt den Signalzustand der einzelnen Bits am Operanden "TagIn_Value" um und schreibt das Ergebnis in den Operanden "TagOut_Value".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit am Parameter OUT, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Parameter IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	UINT	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	Bitfolgen	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL DECO UINT_DWORD
      (IN := "Tag_Input"
       OUT => "Tag_Output"
      )
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

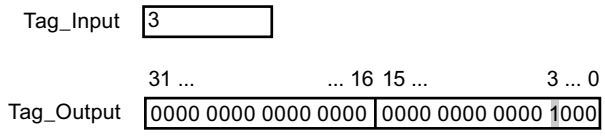

Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "Tag_Input" am Eingang ab und setzt das dritte Bit im Wert des Operanden "Tag_Output" am Ausgang.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Parameter OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Parameter IN und schreibt dessen Bitnummer in den Operanden am Parameter OUT. Wenn der Parameter IN den Wert DW#16#00000001 oder DW#16#00000000 liefert, wird am Parameter OUT der Wert "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
CALL ENCO DWORD
     (IN := "Tag_Input"
      OUT => "Tag_Output"
     )
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

	31 16	15 ... 3 ... 0
"Tag_Input"	0000 1111 0000 0101	0000 1001 0000 1000

"Tag_Output"	3

Die Anweisung selektiert das niederwertigste gesetzte Bit der Variablen "Tag_Input" und schreibt dessen Bitposition "3" in die Variable "Tag_Output".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
G	Input	BOOL	E, A, M, D, L oder Konstante	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P oder Konstante	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SEL CHAR
     (G := "Tag_Input_G"
      IN0 := "Tag_Input0"
      IN1 := "Tag_Input1"
      OUT => "Tag_Output"
     )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Abhängig vom Signalzustand am Eingang "Tag_Input_G" wird der Wert am Eingang "Tag_Input0" oder "Tag_Input1" selektiert und in den Ausgang "Tag_Output" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

MUX: Multiplexen

Beschreibung

Mit der Anweisung "Multiplexen" kopieren Sie den Inhalt eines ausgewählten Eingangs in den Ausgang RET_VAL. Die Anzahl der auswählbaren Eingänge ist bis maximal 32 erweiterbar. Die Nummerierung fängt mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Eingang, dessen Inhalt in den Ausgang RET_VAL kopiert wird. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Eingänge ist, wird der Inhalt des Parameters INELSE in den Ausgang RET_VAL kopiert.

Die Anweisung "Multiplexen" kann nur ausgeführt werden, wenn die Variablen an allen Eingängen sowie am Ausgang RET_VAL vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, für den nur Ganzzahlen angegeben werden können.

Der Wert am Parameter RET_VAL ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang am Parameter K liegt außerhalb der verfügbaren Eingänge. Diese Reaktion ist unabhängig davon, ob der Eingang INELSE verwendet wird oder nicht. Der Wert am Ausgang RET_VAL bleibt unverändert und der Freigabeausgang ENO wird auf "0" gesetzt.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
K	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Gibt den Eingang an, dessen Inhalt kopiert wird.
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L oder Konstante	Erster Eingangswert
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L oder Konstante	Zweiter Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
INn	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L oder Konstante	Optionale Eingangswerte
INELSE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L oder Konstante	Gibt den Wert an, der bei K > n kopiert wird.
RET_VAL	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L	Ausgang, in den der Wert kopiert wird.

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MUX SINT_SINT
      (K := "Tag_Number"
       IN0 := "Tag_Value_1"
       IN1 := "Tag_Value_2"
       INELSE := "Tag_Value_3"
       RET_VAL => "Tag_Result"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
K	Tag_Number	1
IN0	Tag_Value_1	DW#16#00000000
IN1	Tag_Value_2	DW#16#3E4A7D
INELSE	Tag_Value_3	DW#16#FFFF0000
RET_VAL	Tag_Result	DW#16#3E4A7D

Entsprechend dem Wert des Operanden "Tag_Number" wird der Wert am Eingang "Tag_Value_1" kopiert und dem Operanden am Ausgang "Tag_Result" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

DEMUX: Demultiplexen

Beschreibung

Mit der Anweisung "Demultiplexen" kopieren Sie den Inhalt des Eingangs IN in einen ausgewählten Ausgang. Die Anzahl der auswählbaren Ausgänge ist bis maximal 32 erweiterbar. Die Nummerierung fängt bei OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Ausgang, in welchen der Inhalt des Eingangs IN kopiert wird. Die anderen Ausgänge werden nicht verändert. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Ausgänge ist, wird der Inhalt des Eingangs IN in den Parameter OUTELSE kopiert.

Die Anweisung "Demultiplexen" kann nur ausgeführt werden, wenn die Variablen am Eingang IN sowie an allen Ausgängen vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, an welchem nur Ganzzahlen angegeben werden können.

Der Wert am Ausgang OUTELSE ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
K	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird.
IN	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L oder Konstante	Eingangswert
OUT0	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L	Erster Ausgang
OUT1	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L	Zweiter Ausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUTn	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L	Optionale Ausgänge
OUTELSE	Output	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, WCHAR, TOD, LTOD, DATE, LDT	E, A, M, D, L	Ausgang, in welchen der Eingangswert (IN) bei K > n kopiert wird.

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL DEMUX SINT_SINT
     (K := "Tag_Number"
      IN := "Tag_Value"
      OUT0 => "Tag_Output_1"
      OUT1 => "Tag_Output_2"
      OUTELSE => "Tag_Output_3"
     )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Tabelle 4-5 Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks:

Parameter	Operand	Werte	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#3E4A7D

Tabelle 4-6 Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks:

Parameter	Operand	Werte	
OUT0	Tag_Output_1	unverändert	unverändert
OUT1	Tag_Output_2	DW#16#FFFFFFFF	unverändert
OUTELSE	Tag_Output_3	unverändert	DW#16#3E4A7D

Abhängig vom Wert des Operanden "Tag_Number" wird der Wert am Eingang IN in den entsprechenden Ausgang kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operand1 bitweise nach rechts. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operand1 mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt des Operand1 vom Datentyp INT um vier Bitstellen nach rechts verschoben wird:

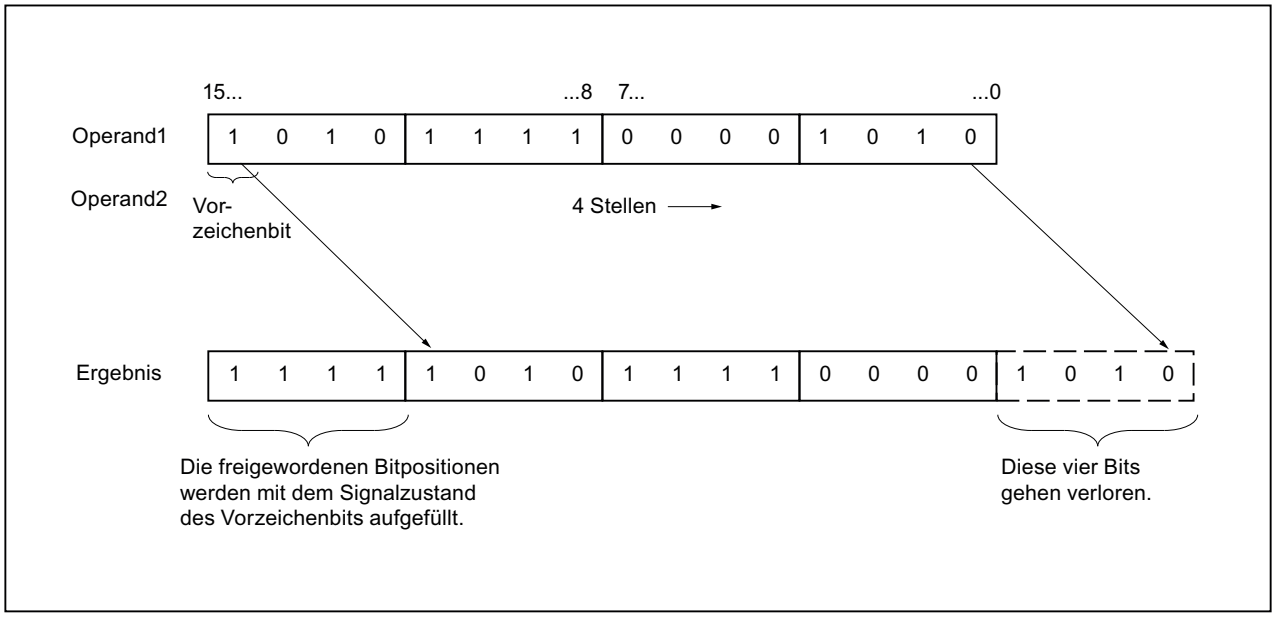

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der verschoben wird
<Operand2>	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SHR_INT("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0011 1111 1010 1111
Tag_Number	3
Tag_OutValue	0000 0111 1111 0101

Der Inhalt des Operanden "Tag_InValue" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operand1 bitweise nach links. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operand1 werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt des Operand1 vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

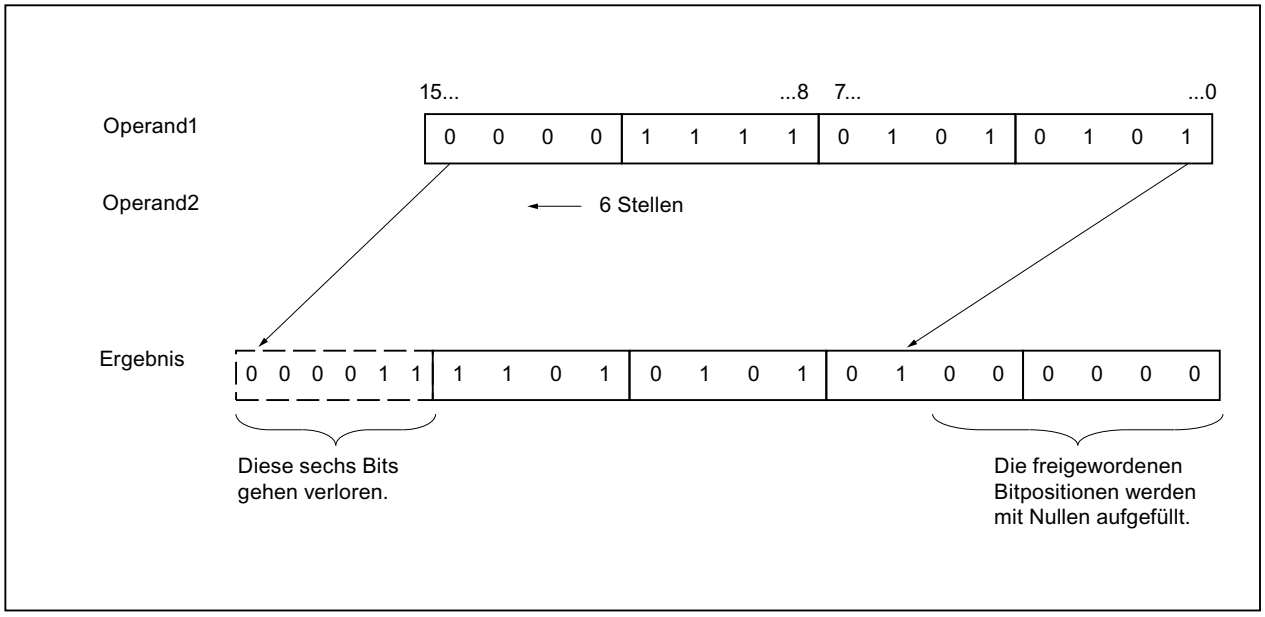

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der verschoben wird
<Operand2>	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SHL_WORD("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0011 1111 1010 1111
Tag_Number	4
Tag_OutValue	1111 1010 1111 0000

Der Inhalt des Operanden "Tag_InValue" wird um vier Bitstellen nach links verschoben und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operand1 bitweise nach rechts. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die angegebene Anzahl der Bitstellen rotiert.

Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

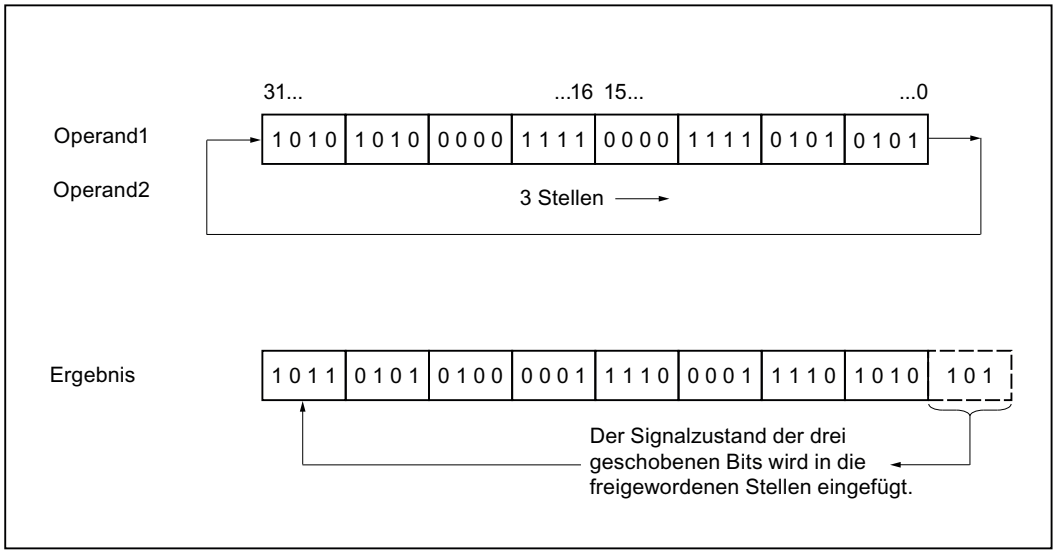

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der rotiert wird
<Operand2>	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ROR("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1010 1010 0000 1111 0000 1111 0101 0101
Tag_Number	5
Tag_OutValue	1010 1101 0101 0000 0111 1000 0111 1010

Der Inhalt des Operanden "Tag_InValue" wird um fünf Bitstellen nach rechts rotiert und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operand1 bitweise nach links. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die angegebene Anzahl der Bitstellen rotiert.

Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

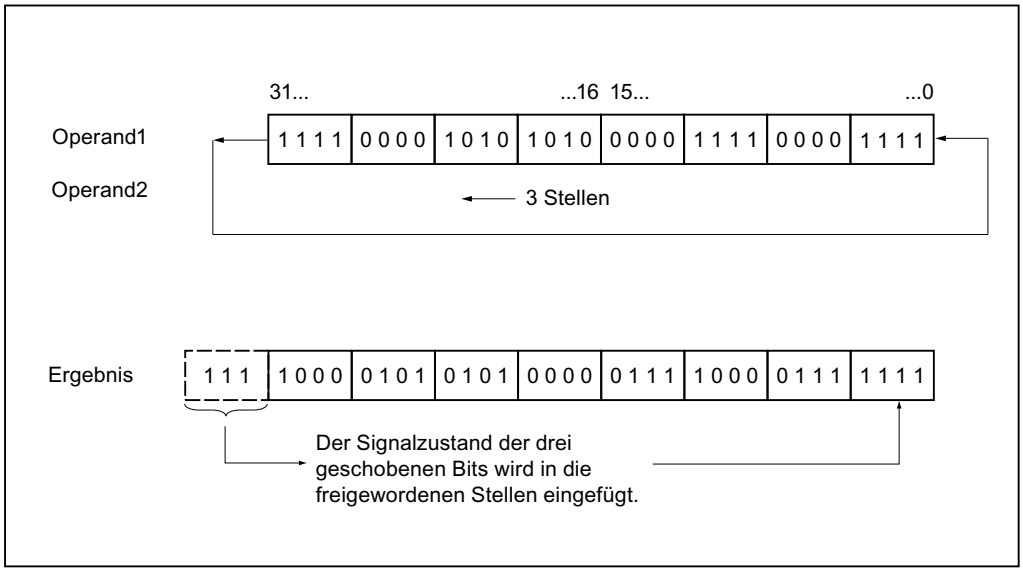

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L oder Konstante	Wert, der rotiert wird
<Operand2>	Input	USINT, UINT, UDINT, ULINT	E, A, M, D, L oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ROL("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1111 0000 1010 1010 0000 1111 0000 1111
Tag_Number	5
Tag_OutValue	0001 0101 0100 0001 1110 0001 1111 1110

Der Inhalt des Operanden "Tag_InValue" wird um fünf Bitstellen nach links rotiert und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

Legacy

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L oder Konstante	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Maximale Schrittnummer (z. B.: LST_STEP = 16#08; Es sind maximal 8 Schritte möglich.)
EVENT(i) 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L oder Konstante	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L oder Konstante	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Voreingestellter erster Schritt des Schrittschaltwerks (1 bis 16)
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P oder Konstante	Verbleibende Bearbeitungszeit für den aktuellen Schritt
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Systemzeit des vorherigen Aufrufs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L oder Konstante	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L oder Konstante	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL DRUM, "DRUM_DB"
     (RESET := "Tag_Reset"
      JOG := "Tag_Input_Jog"
      DRUM_EN := "Tag_Input_Drum_EN"
      LST_STEP := "Tag_Number_LastStep"
      EVENTn := "MyTag_Event_n"
      OUTn => "MyTag_Output_n"
      Q => "Tag_Output_Q"
```

GRAPH

```
OUT_WORD => "Tag_OutputWord"
ERR_CODE => "Tag_ErrorCode"
)
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE

Parameter	Adresse	Wert
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET (nur wenn ET < PT ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn ET < PT ist), OA, CA und CMD_HIS auf "0" zurückgesetzt.

- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L oder Konstante	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
CMD_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit zu CMD

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL DCAT, "DCAT_DB"
     (CMD := "Tag_Input_CMD"
      O_FB := "Tag_Input_O_FB"
      C_FB := "Tag_Input_C_FB"
      Q => "Tag_Output_Q"
      OA => "Tag_Output_OA"
      CA => "Tag_Output_CA"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT

Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L oder Konstante	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L oder Konstante	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an
ET	Static	DINT	D, L oder Konstante	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	D, L oder Konstante	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	D, L oder Konstante	Vorherige Systemzeit
O_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	D, L oder Konstante	Verlaufsbit "Schließen"

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL MCAT, "MCAT_DB"
      (O_CMD := "Tag_Input_O_CMD"
       C_CMD := "Tag_Input_C_CMD"
       S_CMD := "Tag_Input_S_CMD"
       O_FB := "Tag_Input_O_FB"
```

GRAPH

```
C_FB := "Tag_Input_C_FB"
OO => "Tag_OutputOpen"
CO => "Tag_OutputClosed"
OA => "Tag_Output_OA"
CA => "Tag_Output_CA"
Q => "Tag_Output_Q"
)
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für Schritt 2 verglichen. Der Signalzustand des Parameters OUT wird auf TRUE gesetzt, da die Eingabebits mit der Maske für Schritt 2 übereinstimmen.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL IMC, "IMC_DB"
     (IN_BIT0 - 15 := "Tag_Input_BITn"
      CMP_STEP := "Tag_CMP_STEP"
      OUT => "Tag_Output"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE

Parameter	Operand	Wert
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "IMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 7 wird mit Bit 7 der Maske verglichen.

Parame-ter	Deklara-tion	Datentyp	Speicherbereich	Beschreibung
IN_BIT8	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L oder Konstante	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P oder Konstante	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L oder Konstante	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für die Schritte 0 bis 5 verglichen, bis eine Entsprechung gefunden wird. Dabei werden nur die Masken für die Schritte 0 bis 2 abgefragt, weil die Maske für Schritt 2 dem Eingabebit entspricht.

Hinweis

Statische Parameter können Sie im Baustein initialisieren.

GRAPH

```
CALL SMC, "SMC_DB"
     (IN_BIT0 - 15 := "Tag_Input_BITn"
      OUT => "Tag_Output"
      OUT_STEP => "Tag_Output_STEP"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE

Parameter	Operand	Wert
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "SMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
OUT_STEP	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL LEAD_LAG, "LEAD_LAG_DB"
     (IN := "Tag_Input"
      SAMPLE_T := "Tag_Input_SAMPLE_T"
      OUT => "Tag_Output_Result"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Die Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandelt jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in vier äquivalente Codes für eine 7-Segment-Anzeige um und schreibt diese in das Doppelwort des Ausgangs (OUT).

Die Anweisung erkennt keine Fehlerbedingungen.

Folgende Beziehung besteht zwischen den hexadezimalen Eingabewerten und den Bitmustern der Ausgabe:

Ziffer	-gfedcba	Anzeige	Sieben-Segment-Anzeige
0000	00111111	0	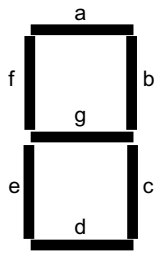
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster des Ziels mit vier Byte

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SEG
```

GRAPH

```
(IN := "Tag_Input"
 OUT => "Tag_Output"
 )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BCDCPL
      (IN := "Tag_Input"
       ERR_CODE => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BITSUM
      (IN := "Tag_Input"
       RET_VAL => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

GRAPH KOP-Anweisungen

Allgemeine KOP-Anweisungen in GRAPH

Beschreibung

Beim Programmieren in GRAPH stehen Ihnen unterschiedliche KOP-Anweisungen zur Verfügung.

Die folgende Tabelle zeigt eine Übersicht der Anweisungen und ihrer Verfügbarkeit:

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
Allgemein		
Netzwerk einfügen	x	
Leerbox	x	x
Verzweigung öffnen	x	x
Verzweigung schließen	x	x
-|: Eingang einfügen	x	
Bitverknüpfungen		
---| |---: Schließerkontakt (Seite 621)	x	x
---| / |---: Öffnerkontakt (Seite 623)	x	x

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
--| NOT |--: VKE invertieren (Seite 624)	x	x
---( )---: Zuweisung (Seite 625)	x	
---(/)---: Zuweisung negieren (Seite 626)	x	
---(R)---: Ausgang rücksetzen (Seite 627)	x	
---(S)---: Ausgang setzen (Seite 628)	x	
SET_BF: Bitfeld setzen (Seite 629)	x	
RESET_BF: Bitfeld rücksetzen (Seite 630)	x	
SR: Flipflop setzen/rücksetzen (Seite 631)	x	
RS: Flipflop rücksetzen/setzen (Seite 633)	x	
--|P|--: Operand auf positive Signalflanke abfragen (Seite 634)	x	
--|N|--: Operand auf negative Signalflanke abfragen (Seite 636)	x	
--(P)--: Operand bei positiver Signalflanke setzen (Seite 638)	x	
--(N)--: Operand bei negativer Signalflanke setzen (Seite 639)	x	
P_TRIG: VKE auf positive Signalflanke abfragen (Seite 640)	x	
N_TRIG: VKE auf negative Signalflanke abfragen (Seite 641)	x	
R_TRIG: Positive Signalflanke erkennen (Seite 643)	x	
F_TRIG: Negative Signalflanke erkennen (Seite 644)	x	
Zeiten		
IEC-Zeiten		
TP: Impuls erzeugen (Seite 645)	x	
TON: Einschaltverzögerung erzeugen (Seite 648)	x	
TOF: Ausschaltverzögerung erzeugen (Seite 651)	x	
TONR: Zeit akkumulieren (Seite 654)	x	
---(TP)---: Zeit als Impuls starten (Seite 657)	x	
---(TON)---: Zeit als Einschaltverzögerung starten (Seite 660)	x	
---(TOF)---: Zeit als Ausschaltverzögerung starten (Seite 663)	x	
---(TONR)---: Zeit akkumulieren (Seite 665)	x	
---(RT)---: Zeit rücksetzen (Seite 668)	x	
---(PT)---: Zeitdauer laden (Seite 670)	x	
SIMATIC-Zeiten (S7-1500)		
S_IMPULS: Zeit als Impuls parametrieren und starten (Seite 671)	x	
S_VIMP: Zeit als verlängerten Impuls parametrieren und starten (Seite 674)	x	
S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten (Seite 677)	x	
S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten (Seite 680)	x	
S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten (Seite 682)	x	
Zähler		
IEC-Zähler		
CTU: Vorwärts zählen (Seite 695)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
CTD: Rückwärts zählen (Seite 697)	x	
CTUD: Vorwärts und rückwärts zählen (Seite 700)	x	
SIMATIC-Zähler (S7-1500)		
Z_VORW: Parametrieren und vorwärts zählen (Seite 703)	x	
Z_RUECK: Parametrieren und rückwärts zählen (Seite 705)	x	
ZAEHLER: Parametrieren und vorwärts/rückwärts zählen (Seite 707)	x	
Vergleicher		
CMP ==: Gleich (Seite 712)	x	x
CMP <>: Ungleich (Seite 718)	x	x
CMP >=: Größer gleich (Seite 723)	x	x
CMP <=: Kleiner gleich (Seite 725)	x	x
CMP >: Größer (Seite 728)	x	x
CMP <: Kleiner (Seite 731)	x	x
IN_RANGE: Wert innerhalb Bereich (Seite 733)	x	
OUT_RANGE: Wert außerhalb Bereich (Seite 735)	x	
---| OK |---: Gültigkeit prüfen (Seite 736)	x	
---| NOT_OK |---: Ungültigkeit prüfen (Seite 737)	x	
EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen (Seite 738)	x	
NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen (Seite 740)	x	
EQ_ElemType: Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen (Seite 742)	x	
NE_ElemType: Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen (Seite 744)	x	
IS_NULL: Auf GLEICH NULL-Pointer abfragen (Seite 745)	x	
NOT_NULL: Auf UNGLEICH NULL-Pointer abfragen (Seite 747)	x	
IS_ARRAY: Auf ARRAY abfragen (Seite 748)	x	
EQ_TypeOfDB: Datentyp DB_ANY auf GLEICH mit dem Datentyp einer Variablen vergleichen (Seite 749)	x	
NE_TypeOfDB: Datentyp DB_ANY auf UNGLEICH mit dem Datentyp einer Variablen vergleichen (Seite 750)	x	
CMP>T: Größer Schrittaktivierungszeit (Seite 2159)		x
CMP>U: Größer ununterbrochene Schrittaktivierungszeit (Seite 2161)		x
Mathematische Funktionen		
CALCULATE: Berechnen (Seite 752)	x	
ADD: Addieren (Seite 755)	x	
SUB: Subtrahieren (Seite 756)	x	
MUL: Multiplizieren (Seite 758)	x	
DIV: Dividieren (Seite 759)	x	
MOD: Divisionsrest gewinnen (Seite 761)	x	
NEG: Zweierkomplement erzeugen (Seite 762)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
INC: Inkrementieren (Seite 763)	x	
DEC: Dekrementieren (Seite 764)	x	
ABS: Absolutwert bilden (Seite 766)	x	
MIN: Minimum ermitteln (Seite 767)	x	
MAX: Maximum ermitteln (Seite 769)	x	
LIMIT: Limitieren (Seite 771)	x	
SQR: Quadrat bilden (Seite 773)	x	
SQRT: Quadratwurzel bilden (Seite 774)	x	
LN: Natürlichen Logarithmus bilden (Seite 776)	x	
EXP: Exponentialwert bilden (Seite 777)	x	
SIN: Sinuswert bilden (Seite 778)	x	
COS: Cosinuswert bilden (Seite 780)	x	
TAN: Tangenswert bilden (Seite 781)	x	
ASIN: Arcussinuswert bilden (Seite 782)	x	
ACOS: Arcuscosinuswert bilden (Seite 784)	x	
ATAN: Arcustangenswert bilden (Seite 785)	x	
FRAC: Nachkommastellen ermitteln (Seite 787)		
EXPT: Potenzieren (Seite 788)	x	
Verschieben		
MOVE: Wert kopieren (Seite 789)	x	
Deserialize: Deserialisieren (Seite 795)	x	
Serialize: Serialisieren (Seite 799)	x	
FieldRead: Feld lesen (Seite 862)	x	
FieldWrite: Feld schreiben (Seite 864)	x	
MOVE_BLK: Bereich kopieren (Seite 804)	x	
SCATTER: Bitfolge in einzelne Bits zerlegen (Seite 821)	x	
SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen (Seite 826)	x	
GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen (Seite 830)	x	
GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen (Seite 834)	x	
MOVE_BLK_VARIANT: Bereich kopieren (Seite 807)	x	
UMOVE_BLK: Bereich ununterbrechbar kopieren (Seite 811)	x	
FILL_BLK: Bereich befüllen (Seite 814)	x	
UFILL_BLK: Bereich ununterbrechbar befüllen (Seite 818)	x	
?=: Zuweisungsversuch (Seite 839)	x	
SWAP: Anordnung ändern (Seite 841)	x	
ReadFromArrayDB: Aus Array-Datenbaustein lesen (Seite 843)	x	
WriteToArrayDB: In Array-Datenbaustein schreiben (Seite 845)	x	
ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen (Seite 848)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben (Seite 851)	x	
VariantGet: Wert einer VARIANT-Variable auslesen (Seite 854)	x	
VariantPut: Wert in eine VARIANT-Variable schreiben (Seite 856)	x	
CountOfElements: Anzahl der ARRAY-Elemente abfragen (Seite 858)	x	
LOWER_BOUND: Untere ARRAY-Grenze auslesen (Seite 859)	x	
UPPER_BOUND: Obere ARRAY-Grenze auslesen (Seite 861)	x	
BLKMOV: Bereich kopieren (S7-1500) (Seite 866)	x	
UBLKMOV: Bereich ununterbrechbar kopieren (S7-1500) (Seite 870)	x	
FILL: Bereich befüllen (S7-1500) (Seite 872)	x	
Umwandler		
CONVERT: Wert konvertieren (Seite 876)	x	
ROUND: Zahl runden (Seite 879)	x	
CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen (Seite 880)	x	
FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen (Seite 881)	x	
TRUNC: Ganzzahl erzeugen (Seite 883)	x	
SCALE_X: Skalieren (Seite 884)	x	
NORM_X: Normieren (Seite 887)	x	
SCALE: Skalieren (S7-1500) (Seite 889)	x	
UNSCALE: Deskalieren (S7-1500) (Seite 892)	x	
Programmsteuerung		
Laufzeitsteuerung		
ENDIS_PW: Passwort-Legitimierung einschränken und freigeben (Seite 904)	x	
RE_TRIGR: Zykluszeitüberwachungszeit neu starten (Seite 910)	x	
STP: Programm beenden (Seite 911)	x	
GET_ERROR: Fehler lokal abfragen (Seite 912)	x	
GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	x	
INIT_RD: Alle remanenten Daten rücksetzen (Seite 919)	x	
WAIT: Zeitverzögerung programmieren (Seite 921)	x	
RUNTIME: Laufzeitmessung (Seite 923)	x	
Wortverknüpfungen		
AND: UND verknüpfen (Seite 926)	x	
OR: ODER verknüpfen (Seite 927)	x	
XOR: EXKLUSIV ODER verknüpfen (Seite 929)	x	
INV: Einerkomplement erzeugen (Seite 930)	x	
DECO: Decodieren (Seite 932)	x	
ENCO: Encodieren (Seite 933)	x	
SEL: Selektieren (Seite 934)	x	
MUX: Multiplexen (Seite 936)	x	
DEMUX: Demultiplexen (Seite 939)	x	

KOP-Anweisung	Perma-nente Anwei-sung	Transition/ Supervision/ Interlock
Schieben und Rotieren		
SHR: Rechts schieben (Seite 942)	x	
SHL: Links schieben (Seite 944)	x	
ROR: Rechts rotieren (Seite 946)	x	
ROL: Links rotieren (Seite 948)	x	
Weitere Anweisungen		
DRUM: Schrittschaltwerk realisieren (Seite 951)	x	
DCAT: Diskreter Steuerungszeitalarm (Seite 957)	x	
MCAT: Motorsteuerungszeitalarm (Seite 960)	x	
IMC: Eingabebits mit den Bits der Maske vergleichen (Seite 964)	x	
SMC: Matrixscanner (Seite 967)	x	
LEAD_LAG: Lead- und Lag-Algorithmus (Seite 969)	x	
SEG: Bitmuster für 7-Segment-Anzeige erzeugen (Seite 972)	x	
BCDCPL: Zehnerkomplement erzeugen (Seite 974)	x	
BITSUM: Anzahl der gesetzten Bits zählen (Seite 975)	x	

CMP>T: Größer Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Gesamtdauer eines Schritts. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn

die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T: Größer Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschaltet.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T: Größer Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T: Größer Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T: Größer Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T: Größer Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

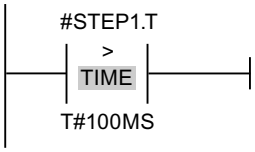

Solange die Aktivierungszeit von #STEP1.T unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

CMP>U: Größer ununterbrochene Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer ununterbrochene Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Dauer eines Schritts, abzüglich eventueller Störungen. Dabei werden etwaige Ereignisse oder Störungen zeitlich nicht mit erfasst und nur die reine Schrittdauer überwacht.

In der programmierten Bedingung wird die gesamte Aktivierungszeit des Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer ununterbrochene Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts abzüglich der Störungen
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

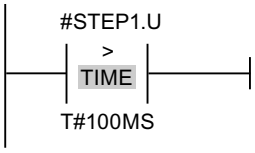

Solange die Aktivierungszeit abzüglich eventueller Störungen von #STEP1.U unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer maximale Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die maximale Gesamtdauer eines Schritts. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit der maximalen Dauer (Operand2) in Millisekunden verglichen. Wenn

die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer maximale Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

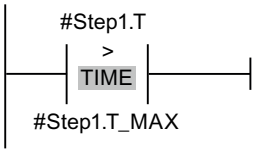

Solange die Aktivierungszeit von #STEP1.T unter der maximalen Schrittaktivierungszeit von #Step1.T_MAX liegt, ist das VKE "0". Sobald die maximale Schrittaktivierungszeit überschritten ist, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Grundlagen zu GRAPH (Seite 7685)

CMP>T_WARN: Größer Warnungszeit

Beschreibung

Mit der Anweisung "Größer Warnungszeit" überwachen Sie in Supervisionsbedingungen die Dauer eines Schritts und geben beim Überschreiten der Zeit eine Warnung aus. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis

(VKE) "1" und eine Warnung wird ausgegeben. Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T_WARN: Größer Warnungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T_WARN: Größer Warnungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T_WARN: Größer Warnungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T_WARN: Größer Warnungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T_WARN: Größer Warnungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Warnungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

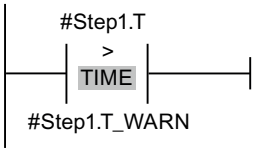

Solange die Aktivierungszeit von #Step1.T die Warnungszeit von #Step1.T_WARN nicht überschritten hat, ist das VKE "0". Sobald die Warnungszeit überschritten ist, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Grundlagen zu GRAPH (Seite 7685)

GRAPH FUP-Anweisungen

Allgemeine FUP-Anweisungen in GRAPH

Beschreibung

Beim Programmieren in GRAPH stehen Ihnen unterschiedliche FUP-Anweisungen zur Verfügung.

Die folgende Tabelle zeigt eine Übersicht der Anweisungen und ihrer Verfügbarkeit:

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
Allgemein		
Neues Netzwerk	x	
Leerbox	x	x
Abzweig: Verzweigung öffnen	x	x
-|: Eingang einfügen (Seite 983)	x	x
-o|: VKE invertieren (Seite 984)	x	x
Bitverknüpfungen		
&: UND-Verknüpfung (Seite 976)	x	x
>=1: ODER-Verknüpfung (Seite 979)	x	x
X: EXCLUSIV ODER-Verknüpfung (Seite 981)	x	
=: Zuweisung (Seite 985)	x	
/=: Zuweisung negieren (Seite 986)	x	
R: Ausgang rücksetzen (Seite 987)	x	
S: Ausgang setzen (Seite 988)	x	
SET_BF: Bitfeld setzen (Seite 989)	x	
RESET_BF: Bitfeld rücksetzen (Seite 991)	x	
SR: Flipflop setzen/rücksetzen (Seite 992)	x	
RS: Flipflop rücksetzen/setzen (Seite 994)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
P: Operand auf positive Signalflanke abfragen (Seite 995)	x	
N: Operand auf negative Signalflanke abfragen (Seite 997)	x	
P=: Operand bei positiver Signalflanke setzen (Seite 998)	x	
N=: Operand bei negativer Signalflanke setzen (Seite 1000)	x	
P_TRIG: VKE auf positive Signalflanke abfragen (Seite 1001)	x	
N_TRIG: VKE auf negative Signalflanke abfragen (Seite 1002)	x	
R_TRIG: Positive Signalflanke erkennen (Seite 1003)	x	
F_TRIG: Negative Signalflanke erkennen (Seite 1005)	x	
Zeiten		
IEC-Zeiten		
TP: Impuls erzeugen (Seite 1006)	x	
TON: Einschaltverzögerung erzeugen (Seite 1009)	x	
TOF: Ausschaltverzögerung erzeugen (Seite 1012)	x	
TONR: Zeit akkumulieren (Seite 1015)	x	
TP: Zeit als Impuls starten (Seite 1018)	x	
TON: Zeit als Einschaltverzögerung starten (Seite 1021)	x	
TOF: Zeit als Ausschaltverzögerung starten (Seite 1023)	x	
TONR: Zeit akkumulieren (Seite 1026)	x	
RT: Zeit rücksetzen (Seite 1029)	x	
PT: Zeitdauer laden (Seite 1030)	x	
SIMATIC-Zeiten		
S_IMPULS: Zeit als Impuls parametrieren und starten (Seite 1032)	x	
S_VIMP: Zeit als verlängerten Impuls parametrieren und starten (Seite 1035)	x	
S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten (Seite 1038)	x	
S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten (Seite 1041)	x	
S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten (Seite 1043)	x	
Zähler		
IEC-Zähler		
CTU: Vorwärts zählen (Seite 1056)	x	
CTD: Rückwärts zählen (Seite 1058)	x	
CTUD: Vorwärts und rückwärts zählen (Seite 1061)	x	
SIMATIC-Zähler		
Z_VORW: Parametrieren und vorwärts zählen (Seite 1065)	x	
Z_RUECK: Parametrieren und rückwärts zählen (Seite 1066)	x	
ZAEHLER: Parametrieren und vorwärts/rückwärts zählen (Seite 1068)	x	
Vergleicher		
CMP==: Gleich (Seite 1074)	x	x
CMP<>: Ungleich (Seite 1079)	x	x

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
CMP>=: Größer gleich (Seite 1084)	x	x
CMP<=: Kleiner gleich (Seite 1086)	x	x
CMP>: Größer (Seite 1089)	x	x
CMP<: Kleiner (Seite 1091)	x	x
IN_RANGE: Wert innerhalb Bereich (Seite 1094)	x	
OUT_RANGE: Wert außerhalb Bereich (Seite 1095)	x	
---| OK |---: Gültigkeit prüfen (Seite 1096)	x	
---| NOT_OK |---: Ungültigkeit prüfen (Seite 1097)	x	
EQ_Type: Datentyp auf GLEICH mit dem Datentyp einer Variablen vergleichen (Seite 1098)	x	
NE_Type: Datentyp auf UNGLEICH mit dem Datentyp einer Variablen vergleichen (Seite 1100)	x	
EQ_ElemType: Datentyp eines ARRAY-Elements auf GLEICH mit dem Datentyp einer Variable vergleichen (Seite 1102)	x	
NE_ElemType: Datentyp eines ARRAY-Elements auf UNGLEICH mit dem Datentyp einer Variable vergleichen (Seite 1104)	x	
IS_NULL: Auf GLEICH NULL-Pointer abfragen (Seite 1105)	x	
NOT_NULL: Auf UNGLEICH NULL-Pointer abfragen (Seite 1107)	x	
IS_ARRAY: Auf ARRAY abfragen (Seite 1108)	x	
EQ_TypeOfDB: Datentyp DB_ANY auf GLEICH mit dem Datentyp einer Variablen vergleichen (Seite 1109)	x	
NE_TypeOfDB: Datentyp DB_ANY auf UNGLEICH mit dem Datentyp einer Variablen vergleichen (Seite 1110)	x	
CMP>T: Größer Schrittaktivierungszeit (Seite 2171)		x
CMP>U: Größer ununterbrochene Schrittaktivierungszeit (Seite 2172)		x
Mathematische Funktionen		
CALCULATE: Berechnen (Seite 1111)	x	
ADD: Addieren (Seite 1115)	x	
SUB: Subtrahieren (Seite 1116)	x	
MUL: Multiplizieren (Seite 1117)	x	
DIV: Dividieren (Seite 1119)	x	
MOD: Divisionsrest gewinnen (Seite 1120)	x	
NEG: Zweierkomplement erzeugen (Seite 1122)	x	
INC: Inkrementieren (Seite 1123)	x	
DEC: Dekrementieren (Seite 1124)	x	
ABS: Absolutwert bilden (Seite 1125)	x	
MIN: Minimum ermitteln (Seite 1127)	x	
MAX: Maximum ermitteln (Seite 1129)	x	
LIMIT: Limitieren (Seite 1130)	x	
SQR: Quadrat bilden (Seite 1133)	x	
SQRT: Quadratwurzel bilden (Seite 1134)	x	
LN: Natürlichen Logarithmus bilden (Seite 1136)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
EXP: Exponentialwert bilden (Seite 1137)	x	
SIN: Sinuswert bilden (Seite 1138)	x	
COS: Cosinuswert bilden (Seite 1139)	x	
TAN: Tangenswert bilden (Seite 1141)	x	
ASIN: Arcussinuswert bilden (Seite 1142)	x	
ACOS: Arcuscosinuswert bilden (Seite 1144)	x	
ATAN: Arcustangenswert bilden (Seite 1145)	x	
FRAC: Nachkommastellen ermitteln (Seite 1147)	x	
EXPT: Potenzieren (Seite 1148)	x	
Verschieben		
MOVE: Wert kopieren (Seite 1150)	x	
Deserialize: Deserialisieren (Seite 1156)	x	
Serialize: Serialisieren (Seite 1160)	x	
FieldRead: Feld lesen (Seite 1222)	x	
FieldWrite: Feld schreiben (Seite 1224)	x	
MOVE_BLK: Bereich kopieren (Seite 1164)	x	
SCATTER: Bitfolge in einzelne Bits zerlegen (Seite 1182)	x	
SCATTER_BLK: Elemente eines ARRAY of Bitfolge in einzelne Bits zerlegen (Seite 1186)	x	
GATHER: Einzelne Bits zu einer Bitfolge zusammenfügen (Seite 1190)	x	
GATHER_BLK: Einzelne Bits zu mehreren Elementen eines ARRAY of Bitfolge zusammenfügen (Seite 1194)	x	
MOVE_BLK_VARIANT: Bereich kopieren (Seite 1167)	x	
UMOVE_BLK: Bereich ununterbrechbar kopieren (Seite 1172)	x	
FILL_BLK: Bereich befüllen (Seite 1175)	x	
UFILL_BLK: Bereich ununterbrechbar befüllen (Seite 1178)	x	
?=: Zuweisungsversuch (Seite 1199)	x	
SWAP: Anordnung ändern (Seite 1201)	x	
ReadFromArrayDB: Aus Array-Datenbaustein lesen (Seite 1203)	x	
WriteToArrayDB: In Array-Datenbaustein schreiben (Seite 1205)	x	
ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen (Seite 1208)	x	
WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben (Seite 1211)	x	
VariantGet: Wert einer VARIANT-Variable auslesen (Seite 1214)	x	
VariantPut: Wert in eine VARIANT-Variable schreiben (Seite 1215)	x	
CountOfElements: Anzahl der ARRAY-Elemente abfragen (Seite 1217)	x	
LOWER_BOUND: Untere ARRAY-Grenze auslesen (Seite 1219)	x	
UPPER_BOUND: Obere ARRAY-Grenze auslesen (Seite 1220)	x	
BLKMOV: Bereich kopieren (S7-1500) (Seite 1226)	x	
UBLKMOV: Bereich ununterbrechbar kopieren (S7-1500) (Seite 1230)	x	
FILL: Bereich befüllen (S7-1500) (Seite 1232)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
Umwandler		
CONVERT: Wert konvertieren (Seite 1236)	x	
ROUND: Zahl runden (Seite 1238)	x	
CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen (Seite 1240)	x	
FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen (Seite 1241)	x	
TRUNC: Ganzzahl erzeugen (Seite 1243)	x	
SCALE_X: Skalieren (Seite 1244)	x	
NORM_X: Normieren (Seite 1246)	x	
VARIANT_TO_DB_ANY: VARIANT in DB_ANY konvertieren	x	
DB_ANY_TO_VARIANT: DB_ANY in VARIANT konvertieren	x	
SCALE: Skalieren (S7-1500) (Seite 1249)	x	
UNSCALE: Deskalieren (S7-1500) (Seite 1251)	x	
Programmsteuerung		
Laufzeitsteuerung		
ENDIS_PW: Passwort-Legitimierung einschränken und freigeben (Seite 1264)	x	
RE_TRIGR: Zyklusüberwachungszeit neu starten (Seite 1270)	x	
STP: Programm beenden (Seite 1271)	x	
GET_ERROR: Fehler lokal abfragen (Seite 1272)	x	
GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1276)	x	
INIT_RD: Alle remanenten Daten zurücksetzen (Seite 1280)	x	
WAIT: Zeitverzögerung programmieren (Seite 1281)	x	
RUNTIME: Laufzeitmessung (Seite 1283)	x	
Wortverknüpfungen		
AND: UND verknüpfen (Seite 1286)	x	
OR: ODER verknüpfen (Seite 1287)	x	
XOR: EXKLUSIV ODER verknüpfen (Seite 1289)	x	
INV: Einerkomplement erzeugen (Seite 1291)	x	
DECO: Decodieren (Seite 1292)	x	
ENCO: Encodieren (Seite 1293)	x	
SEL: Selektieren (Seite 1295)	x	
MUX: Multiplexen (Seite 1297)	x	
DEMUX: Demultiplexen (Seite 1300)	x	
Schieben und Rotieren		
SHR: Rechts schieben (Seite 1303)	x	
SHL: Links schieben (Seite 1305)	x	
ROR: Rechts rotieren (Seite 1307)	x	
ROL: Links rotieren (Seite 1309)	x	
Weitere Anweisungen		
DRUM: Schrittschaltwerk realisieren (Seite 1311)	x	
DCAT: Diskreter Steuerungszeitalarm (Seite 1318)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
MCAT: Motorsteuerungszeitalarm (Seite 1321)	x	
IMC: Eingabebits mit den Bits der Maske vergleichen (Seite 1325)	x	
SMC: Matrixscanner (Seite 1327)	x	
LEAD_LAG: Lead- und Lag-Algorithmus (Seite 1329)	x	
SEG: Bitmuster für 7-Segment-Anzeige erzeugen (Seite 1332)	x	
BCDCPL: Zehnerkomplement erzeugen (Seite 1333)	x	
BITSUM: Anzahl der gesetzten Bits zählen (Seite 1334)	x	

CMP>T: Größer Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer Schrittaktivierungszeit" können Sie in Supervisionsbedingungen die Gesamtdauer eines Schritts überwachen. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T: Größer Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T: Größer Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T: Größer Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T: Größer Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T: Größer Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
IN2	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

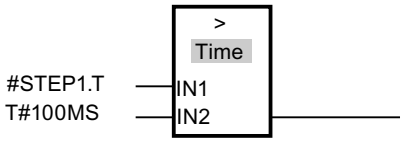

Solange die Aktivierungszeit von #STEP1.T unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

CMP>U: Größer ununterbrochene Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer ununterbrochene Schrittaktivierungszeit" können Sie in Supervisionsbedingungen die Dauer eines Schritts, abzüglich eventueller Störungen überwachen. Dabei werden etwaige Ereignisse oder Störungen zeitlich nicht mit erfasst und nur die reine Schrittdauer überwacht.

In der programmierten Bedingung wird die gesamte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des

Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >U: Größer ununterbrochene Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer ununterbrochene Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts abzüglich der Störungen
IN2	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

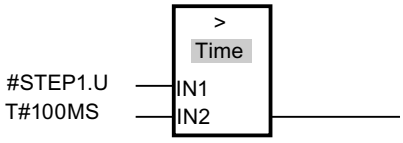

Solange die Aktivierungszeit abzüglich eventueller Störungen von #STEP1.U unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zu GRAPH (Seite 7685)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer maximale Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die maximale Gesamtdauer eines Schritts. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit der maximalen Dauer (Operand2) in Millisekunden verglichen. Wenn

die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T_MAX: Größer maximale Schrittaktivierungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer maximale Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
IN2	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

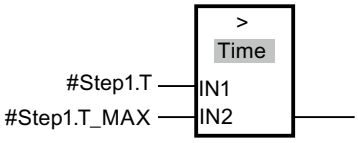

Solange die Aktivierungszeit von #STEP1.T unter der maximalen Schrittaktivierungszeit von #Step1.T_MAX liegt, ist das VKE "0". Sobald die maximale Schrittaktivierungszeit überschritten ist, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Grundlagen zu GRAPH (Seite 7685)

CMP>T_WARN: Größer Warnungszeit

Beschreibung

Mit der Anweisung "Größer Warnungszeit" überwachen Sie in Supervisionsbedingungen die Dauer eines Schritts und geben beim Überschreiten der Zeit eine Warnung aus. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis

(VKE) "1" und eine Warnung wird ausgegeben. Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Weiterhin angezeigter Zustand trotz "inaktiver" Anweisung

Unter den folgenden Voraussetzungen gilt:

- Vor der Anweisung "CMP >T_WARN: Größer Warnungszeit" (mit Datentyp STRING, WSTRING) wird eine Bedingung (z. B. eines Schließerkontakts) im Netzwerk erfragt.
- "Beobachten ein" ist eingeschalten.
- Ein neues Ergebnis der Bedingung setzt das Netzwerk wieder auf FALSE. Dadurch ist die Anweisung "CMP >T_WARN: Größer Warnungszeit" inaktiv.

Ergebnis:

Für die Anweisung "CMP >T_WARN: Größer Warnungszeit" (Datentyp STRING, WSTRING) wird weiterhin der vorherige Zustand im Netzwerk angezeigt.

Wenn Sie "Beobachten" aus- und wieder einschalten oder zu einem anderen Netzwerk scrollen, wird wieder der korrekte Zustand der Anweisung "CMP >T_WARN: Größer Warnungszeit" (Datentyp STRING, WSTRING) angezeigt. Die Anweisung "CMP >T_WARN: Größer Warnungszeit" wird im inaktiven Zustand im Netzwerk ausgegraut dargestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Warnungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
IN2	Input	Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, DATE, DT, DTL, TOD, LTOD, LDT	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

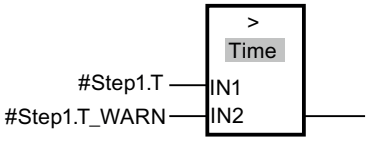

Solange die Aktivierungszeit von #Step1.T die Warnungszeit von #Step1.T_WARN nicht überschritten hat, ist das VKE "0". Sobald die Warnungszeit überschritten ist, wechselt das VKE auf "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 196)

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Grundlagen zu GRAPH (Seite 7685)

4.1.4 Erweiterte Anweisungen

4.1.4.1 Datum und Uhrzeit

T_COMP: Zeitvariablen vergleichen

Beschreibung

Mit der Anweisung vergleichen Sie die Inhalte zweier Variablen der Datentypen "Zeiten" oder "Datum und Uhrzeit".

Die Anweisung unterstützt den Vergleich der folgenden Datentypen: DATE, TIME, LTIME, TOD (TIME_OF_DAY), LTOD (LTIME_OF_DAY), DT (DATE_AND_TIME), LDT (DATE_AND_LTIME), DTL, S5Time.

Um den Vergleich durchzuführen, müssen die Datentypen in Länge und Format übereinstimmen.

Das Vergleichsergebnis wird als Rückgabewert am Parameter OUT ausgegeben. Wenn die verwendete Vergleichsbedingung erfüllt ist, wird der Parameter OUT auf "1" gesetzt.

Es gibt folgende Vergleichsmöglichkeiten:

Symbol	Beschreibung
EQ	Wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 gleich ist, führt der Rückgabewert zu Signalzustand "1".
NE	Wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 ungleich ist, führt der Rückgabewert zu Signalzustand "1".
GE	Wenn der Zeitpunkt Parameter IN1 größer (jünger) oder gleich dem Zeitpunkt am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
LE	Wenn der Zeitpunkt am Parameter IN1 kleiner (älter) oder gleich dem Zeitpunkt am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".

Symbol	Beschreibung
GT	Wenn der Zeitpunkt am Parameter IN1 größer (jünger) als am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
LT	Wenn der Zeitpunkt am Parameter IN1 kleiner (älter) als am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_COMP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL, S5Time	E, A, M, D, L, P oder Konstante	Erster zu vergleichender Wert.
IN2	Input	DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL, S5Time	E, A, M, D, L, P oder Konstante	Zweiter zu vergleichender Wert.
OUT	Output	BOOL	E, A, M, D, L, P	Rückgabewert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel vergleichen Sie mithilfe der Vergleichsmöglichkeit "größer gleich" zwei Zeiten mit dem Datentyp LTIME miteinander.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

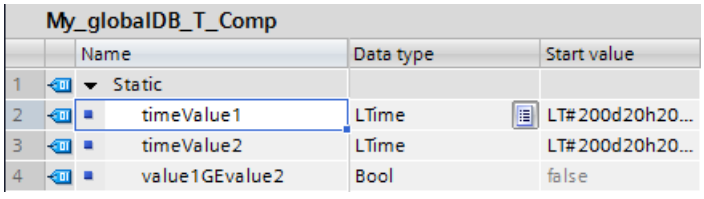

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen die Vergleichsmöglichkeit "GE".

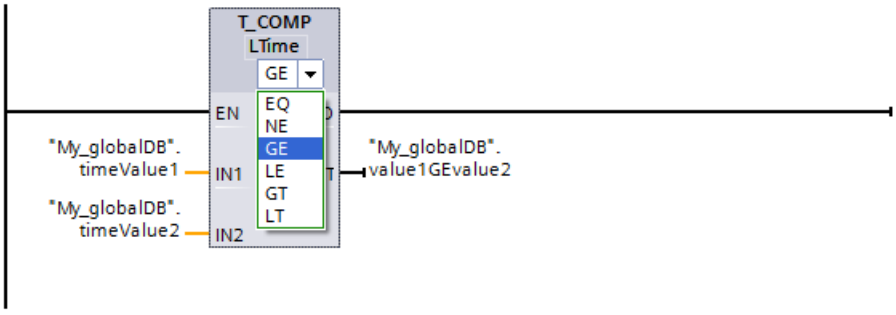

Da die Zeit des ersten zu vergleichenden Werts ("timeValue1") größer gleich dem zweiten Wert ("timeValue2") ist, zeigt der Rückgabewert ("value1GEvalue2") den Signalzustand "TRUE".

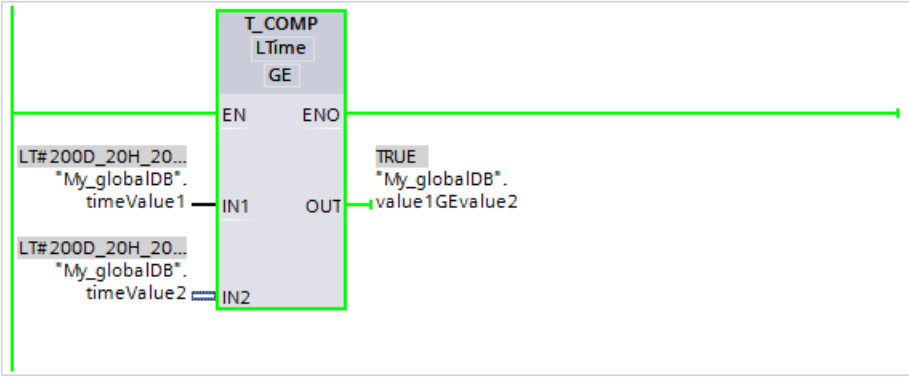

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

T_CONV: Zeiten umwandeln und extrahieren

Beschreibung

Mit der Anweisung "T_CONV" konvertieren Sie den Datentyp am Eingangsparameter IN in den Datentyp, der am Ausgang OUT ausgegeben wird. Über die Anweisungsboxen des Ein- und Ausgangs wählen Sie die Datenformate zur Konvertierung aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_CONV". Wird am Eingangs- und Ausgangsparameter der gleiche Datentyp verwendet, kopiert die Anweisung den entsprechenden Wert.

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	Ganzzahlen, TIME, Datum und Uhrzeit*	WORD, Ganzzahlen, Zeiten, Datum und Uhrzeit*	E, A, M, D, L, P oder Konstante	Zu konvertierender Wert
OUT	Return	Ganzzahlen, TIME, Datum und Uhrzeit*	WORD, Ganzzahlen, Zeiten, Datum und Uhrzeit*	E, A, M, D, L, P	Ergebnis der Konvertierung
* Der Umfang der unterstützten Datentypen ist abhängig von der CPU. Welche Datentypen von den Baugruppen S7-1200 und S7-1500 jeweils unterstützt werden, entnehmen Sie der Übersicht der jeweils gültigen Datentypen.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel konvertieren Sie eine Zeit mit dem Datentyp DATE AND TIME in eine Zeit mit dem Datentyp LTIME OF DAY um.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

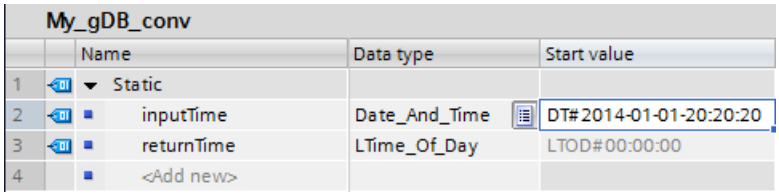

My_gDB_conv

		Name	Data type	Start value
1		Static		
2		inputTime	Date_And_Time	DT#2014-01-01-20:20:20
3		returnTime	LTime_Of_Day	LTOD#00:00:00
4		<Add new>		

Die Parameter der Anweisung verschalten Sie wie folgt und wählen die Datentypen aus.

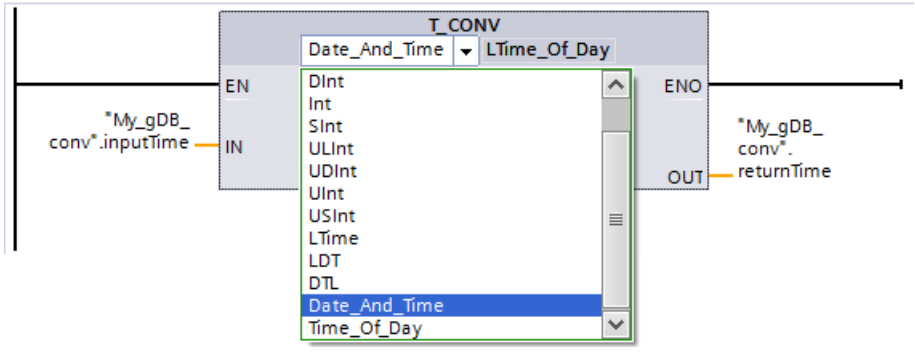

Der zu konvertierende Wert ("inputTime") wird am Ausgangsparameter als neuer Zeitwert ausgegeben ("returnTime"). Die Information für das Datum geht verloren.

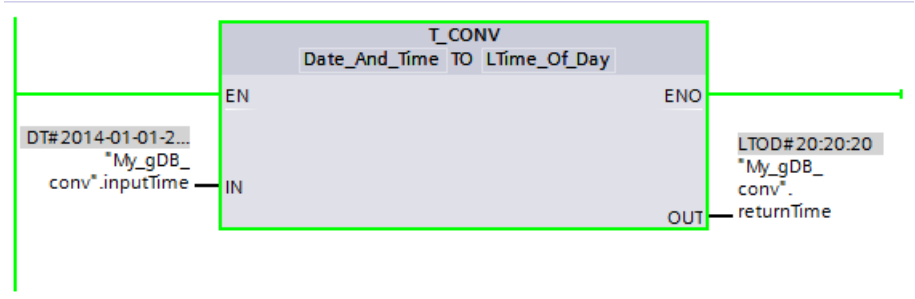

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

T_ADD: Zeiten addieren

Beschreibung

Mit der Anweisung addieren Sie die Zeitangabe am Eingangsparameter IN1 zu der Zeitangabe am Eingangsparameter IN2. Das Ergebnis können Sie am Ausgangsparameter OUT abfragen. Dabei ist die Addition folgender Formate möglich:

- Addition einer Zeitdauer zu einer anderen Zeitdauer.
  Beispiel: Addition eines Datentyps TIME zu einem weiteren Datentyp TIME.
- Addition einer Zeitdauer zu einem Zeitpunkt.
  Beispiel: Addition eines Datentyps TIME zu dem Datentyp DTL.

Den Datentyp für den Wert am Eingangsparameter IN1 und Ausgangsparameter OUT bestimmen Sie durch die Auswahl in den Anweisungsboxen des Ein- und Ausgangs. Am Eingangsparameter IN2 geben Sie nur Zeitangaben des Formats TIME (bei S7-1500er Baugruppen auch LTIME) an.

Hinweis

Überlauf / Unterlauf

Bei der Berechnung kann es zu einer Überschreitung des größtmöglichen oder einer Unterschreitung des kleinstmöglichen Werts des Ausgangsparameters OUT kommen. Werten Sie den Freigabeausgang ENO aus, um dies festzustellen.

Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "T_ADD" entsprechend der möglichen Konvertierungen:

Tabelle 4-7 Addition einer Zeitdauer zu einer anderen Zeitdauer

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Erster Summand
IN2	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zweiter Summand
OUT	Return	DINT, DWORD, TIME, TOD	TIME, LTIME,	E, A, M, D, L, P	Ergebnis der Addition Die Auswahl der Datentypen ist abhängig von den gewählten Datentypen der Eingangsparameter IN1 und IN2.

Tabelle 4-8 Addition einer Zeitdauer zu einem Zeitpunkt

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	DTL, TOD	DT, TOD, LTOD, LDT, DTL	E, A, M, D, L, P oder Konstante	Erster Summand Bei LTIME am Parameter IN2 können nur LTOD, LDT oder DTL verwendet werden.
IN2	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Zweiter Summand
OUT	Return	DINT, DWORD, TIME, TOD, UDINT, DTL	DT, DTL, LDT, TOD, LTOD	E, A, M, D, L, P	Ergebnis der Addition Die Auswahl der Datentypen ist abhängig von den gewählten Datentypen der Eingangsparameter IN1 und IN2.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel addieren Sie eine Zeitdauer mit dem Datentyp TIME zu einem Zeitpunkt mit dem Datentyp TOD.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

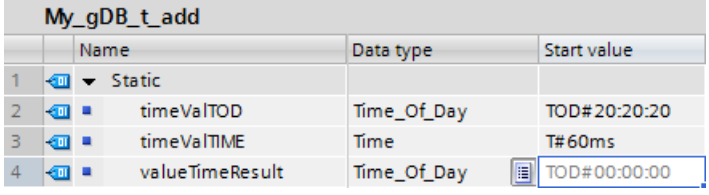

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen die Datentypen für Zeitpunkt ("timeValTOD") und für Zeitdauer ("timeValTIME") aus.

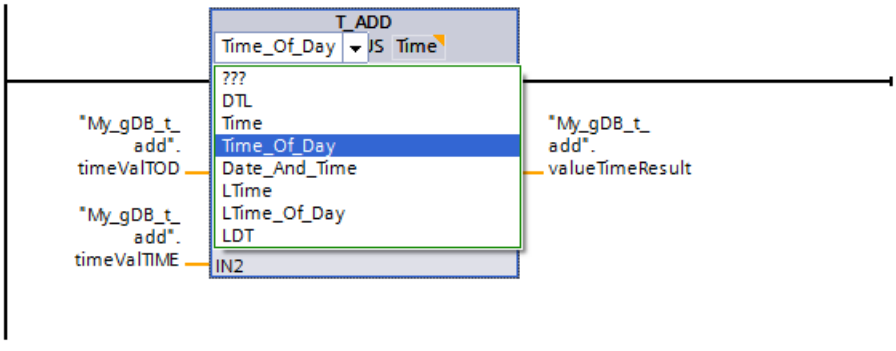

Der Zeitpunkt ("timeValTOD") und die Zeitdauer ("timeValTIME") werden addiert und deren Ergebnis als Zeitpunkt am Ausgabeparameter OUT ("valueTimeResult") angezeigt.

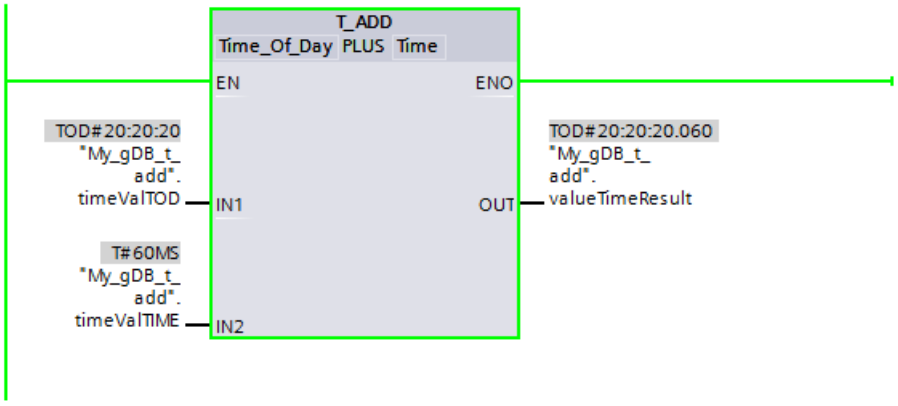

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

T_SUB: Zeiten subtrahieren

Beschreibung

Mit der Anweisung subtrahieren Sie die Zeitangabe am Eingangsparameter IN2 von der Zeitangabe am Eingangsparameter IN1. Am Ausgangsparameter OUT fragen Sie die Differenz ab. Dabei ist die Subtraktion folgender Formate möglich:

- Subtraktion einer Zeitdauer von einer anderen Zeitdauer
  Beispiel: Subtraktion einer Zeitdauer vom Datentyp TIME von einer anderen Zeitdauer vom Datentyp TIME. Das Ergebnis kann in eine Variable des Datentyps TIME ausgegeben werden.
- Subtraktion einer Zeitdauer von einem Zeitpunkt
  Beispiel: Subtraktion einer Zeitdauer vom Datentyp TIME von einem Zeitpunkt vom Datentyp DTL. Das Ergebnis kann in eine Variable des Datentyps DTL ausgegeben werden.

Die Formate der Werte am Eingangsparameter IN1 und Ausgangsparameter OUT bestimmen Sie durch die Auswahl der Datentypen für den Eingangsparameter und den Ausgangsparameter der Anweisung.

Hinweis

Überlauf / Unterlauf

Bei der Berechnung kann es zu einer Überschreitung des größtmöglichen oder einer Unterschreitung des kleinstmöglichen Werts des Ausgangsparameters OUT kommen. Werten Sie den Freigabeausgang ENO aus, um dies festzustellen.

Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "T_SUB" entsprechend der möglichen Konvertierungen:

Tabelle 4-9 Subtraktion einer Zeitdauer von einer anderen Zeitdauer

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Return	DINT, DWORD, TIME, TOD, UDINT	TIME, LTIME	E, A, M, D, L, P	Ergebnis der Subtraktion

Tabelle 4-10 Subtraktion einer Zeitdauer von einem Zeitpunkt

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	DTL, TOD	TOD, LTOD, DTL, DT, LDT	E, A, M, D, L, P oder Konstante	Minuend Bei LTIME am Parameter IN2 werden nur LTOD, LDT oder DTL verwendet.
IN2	Input	TIME	TIME, LTIME	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Return	DTL, DINT, DWORD, TIME, TOD, UDINT	TOD, LTOD, DTL, DT, LDT	E, A, M, D, L, P	Ergebnis der Subtraktion

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel subtrahieren Sie eine Zeitdauer mit dem Datentyp TIME von einem Zeitpunkt mit dem Datentyp TOD.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

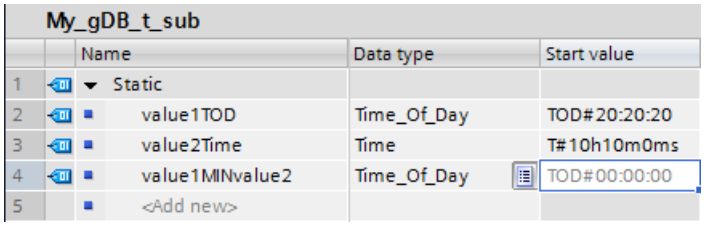

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp für den Zeitpunkt ("value1TOD") und den Datentyp für die Zeitdauer ("value2Time") aus.

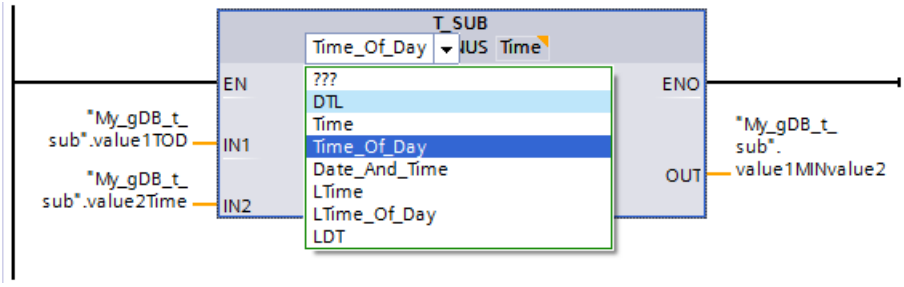

Der Zeitpunkt ("value1TOD") und die Zeitdauer ("value2Time") werden subtrahiert und deren Ergebnis als Zeitpunkt am Ausgabeparameter OUT ("value1MINvalue2") angezeigt.

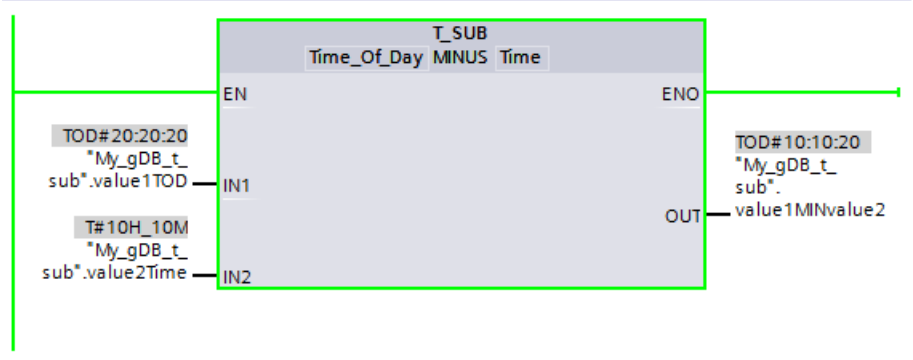

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

T_DIFF: Zeitdifferenz

Beschreibung

Mit der Anweisung subtrahieren Sie die Zeitangabe am Eingangsparameter IN2 von der Zeitangabe am Eingangsparameter IN1. Das Ergebnis wird am Ausgangsparameter OUT ausgegeben.

- Wenn die Zeitangabe am Eingangsparameter IN2 größer als die Zeitangabe am Eingangsparameter IN1 ist, wird als Ergebnis ein negativer Wert am Ausgangsparameter OUT ausgegeben.
- Wenn das Ergebnis der Subtraktion außerhalb des TIME-Zahlenbereichs liegt, wird der Freigabeausgang ENO = "0". Abhängig von den verwendeten Datentypen erhält man entweder ein verstümmeltes Ergebnis oder "0" (0:00).
  - Wenn Sie als Minuend und Subtrahend Parameter vom Datentyp DTL wählen, ist das Ergebnis vom Datentyp TIME. Es darf nicht größer als 24 Tage sein, sonst wird der Freigabeausgang ENO = "0" und das Ergebnis ist "0" .
  - Wenn Sie als Minuend und Subtrahend Parameter vom Datentyp LDT wählen, besteht diese Einschränkung nicht.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_DIFF":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	DTL, DATE, TOD	DTL, DATE, DT, TOD, LTOD, LDT	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	DTL, DATE, TOD	DTL, DATE, DT, TOD, LTOD, LDT	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Return	TIME, INT	TIME, LTIME, INT	E, A, M, D, L, P	Differenz der Eingangsparameter

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Verwendung der Anweisung "T_DIFF" in SCL

In SCL ist für den Parameter OUT der Datentyp TIME voreingestellt (siehe auch: Grundlagen zu den Datentypen von SCL-Anweisungen (Seite 7654)). Wenn Sie für OUT den Datentyp INT benutzen wollen, müssen Sie den Anweisungsnamen T_DIFF um "_INT" ergänzen (Siehe auch: Datentyp einer SCL-Anweisung ändern (Seite 7655)).

Beispiel

In dem folgenden Beispiel errechnen Sie die Differenz zweier Zeitpunkte mit dem Datentyp TOD. Die Differenz geben Sie im Datentyp TIME an.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

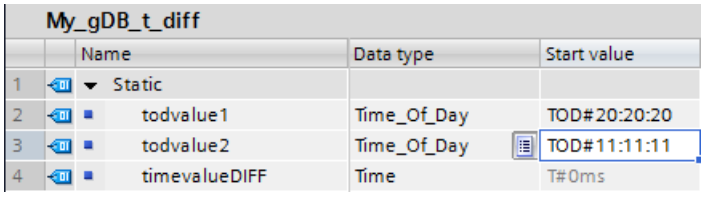

Die Parameter der Anweisung verschalten Sie wie folgt und wählen die Datentypen aus. Bei der ersten Auswahlmöglichkeit geben Sie den Datentyp der Zeitpunkte an. Bei der zweiten Auswahlmöglichkeit geben Sie den Datentyp der Differenz an.

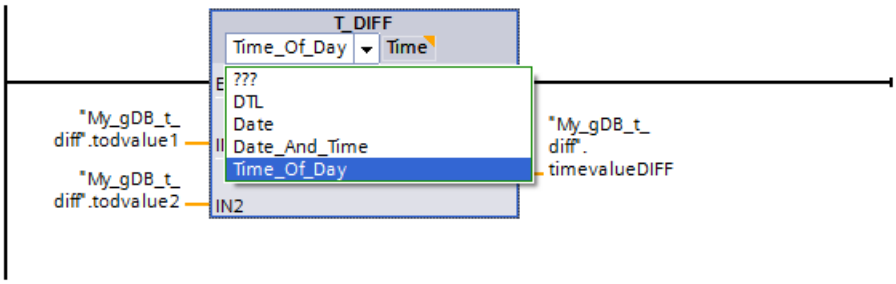

Der erste Zeitpunkt ("todvalue1") und der zweite Zeitpunkt ("todvalue2") werden subtrahiert und deren Differenz als Zeitdauer am Ausgabeparameter OUT ("timevalueDIFF") angezeigt.

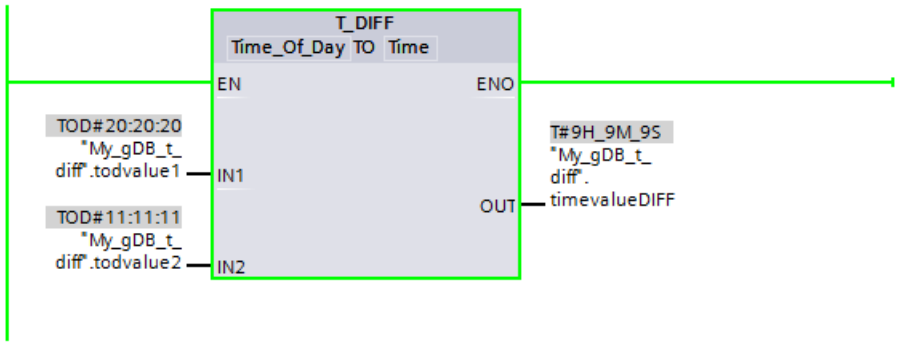

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

T_COMBINE: Zeiten zusammenfassen

Beschreibung

Die Anweisung fasst den Wert für ein Datum mit dem einer Uhrzeitangabe zusammen. Das Ergebnis ist eine kombinierte Datums- und Uhrzeitangabe.

- Am Eingangsparameter IN1 wird die Datumsangabe eingegeben. Verwenden Sie bei dem Datentyp DATE einen Wert zwischen 1990-01-01 und 2089-12-31 (wird nicht geprüft).
- Am Eingangswert IN2 wird die Uhrzeitangabe (Datentyp TOD/LTOD) eingegeben.
- Am Ausgangswert OUT wird der kombinierte Datentyp zur Datums- und Uhrzeitangabe ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_COMBINE":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN1	Input	DATE	DATE	E, A, M, D, L, P oder Konstante	Eingangsvariable des Datums
IN2	Input	TOD	TOD, LTOD	E, A, M, D, L, P oder Konstante	Eingangsvariable der Uhrzeit
OUT	Return	DTL	DT, DTL, LDT	E, A, M, D, L, P	Rückgabewert von Datum und Uhrzeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Verwendung von T_COMBINE in SCL-Programmen

Wenn Sie die Anweisung T_COMBINE aus der Task Card "Anweisungen" in das SCL-Programmierfenster ziehen, wird dort die interne Anweisung CONCAT_DATE_TOD oder CONCAT_DATE_LTOD angezeigt.

Beispiel

In dem folgenden Beispiel kombinieren Sie eine Uhrzeit mit dem Datentyp TOD und ein Datum mit dem Datentyp DATE. Den Rückgabewert geben Sie im Datentyp DT an.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

My_gDB_t_comb

	Name	Data type	Start value
1	Static		
2	valueDATE	Date	D#2014-01-01
3	valueTOD	Time_Of_Day	TOD#20:22:20
4	combTIME	Date_And_Time	DT#1990-01-01-00:00:00

Die Parameter der Anweisung verschalten Sie wie folgt und wählen die Datentypen aus. Bei der ersten Auswahlmöglichkeit geben Sie den Datentyp der Tageszeit ("valueTOD") an. Bei der zweiten Auswahlmöglichkeit geben Sie den Datentyp des Rückgabewerts ("combTIME") an.

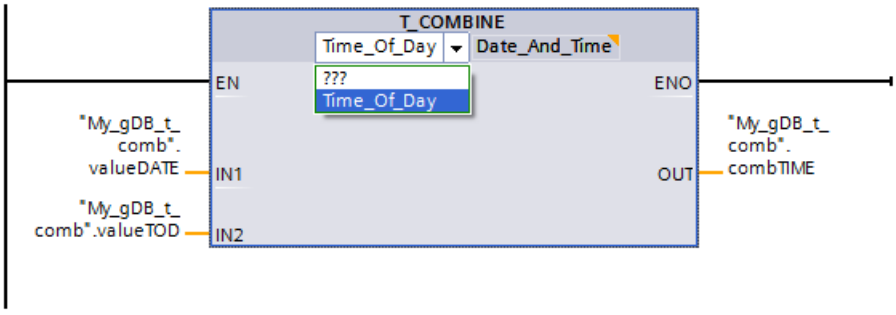

Das Datum ("valueDATE") wird um die Angabe der Uhrzeit ("valueTOD") erweitert und deren Rückgabewert am Ausgabeparameter OUT ("combTIME") angezeigt.

My_gDB_t_comb

	Name	Data type	Start value	Monitor value
1	Static			
2	valueDATE	Date	D#2014-01-01	D#2014-1-1
3	valueTOD	Time_Of_Day	TOD#20:22:20	TOD#20:22:20
4	combTIME	Date_And_Time	DT#1990-01-01-0	DT#2014-01-01-20:22:20

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Uhrzeitfunktionen

WR_SYS_T: Uhrzeit stellen

Beschreibung

Mit der Anweisung stellen Sie das Datum und die Uhrzeit (Baugruppenzeit) der CPU-Uhr ein. Die Angabe zu Datum und Uhrzeit geben Sie am Eingangsparameter IN an. Der Wert muss in dem folgenden Bereich liegen:

- Bei DT: min. DT#1990-01-01-0:0:0, max. DT#2089-12-31-23:59:59.999
- Bei LDT: min. LDT#1970-01-01-0:0:0.000000000, max. LDT#2200-12-31-23:59:59.999999999
- Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2200-12-31-23:59:59.999999999

Am Ausgangsparameter RET_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Angaben zur lokalen Zeitzone oder Sommerzeit werden mit der Anweisung "WR_SYS_T" nicht übertragen.

Hinweis

Baugruppenzeit der CPU-Uhr

Die CPU-Uhr interpretiert die Baugruppenzeit als die koordinierte Weltzeit (UTC). Dementsprechend wird die Baugruppenzeit stets ohne die Faktoren "lokale Zeitzone" oder "Sommerzeit" in der CPU-Uhr hinterlegt. Die CPU-Uhr berechnet dann auf Basis der Baugruppenzeit die Lokalzeit der CPU-Uhr.

Die Baugruppenzeit der CPU-Uhr dient als Vorlage für alle zeitlichen Prozesse, ausgehend von der CPU. Beispiele für die Verwendung sind:

- Berechnung der Lokalzeit der CPU-Uhr auf Basis der Baugruppenzeit
- Darstellung der Baugruppenzeit unter "Online & Diagnose" in Lokalzeit
- Bausteineigenschaften
- Einträge im Diagnosepuffer der CPU

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_SYS_T":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	DTL	DT **, DTL, LDT	E, A, M, D, L, P oder Konstante *	Datum und Uhrzeit
RET_VAL	Return	INT	INT	E, A, M, D, L, P	Status der Anweisung
* Die Datentypen DT und DTL werden für die Speicherbereiche Eingang, Ausgang und Merker nicht verwendet. ** Bei Verwendung des Datentyps DT werden die Millisekunden nicht zur CPU übertragen.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Fehler im Datum
8081	Fehler in der Uhrzeit
8082**	Ungültiger Wert für die Angabe des Monats (Byte 2 im Format DTL)
8083**	Ungültiger Wert für die Angabe des Tages (Byte 3 im Format DTL)
8084**	Ungültiger Wert für die Angabe der Stunde (Byte 5 im Format DTL)
8085**	Ungültiger Wert für die Angabe der Minute (Byte 6 im Format DTL)
8086**	Ungültiger Wert für die Angabe der Sekunde (Byte 7 im Format DTL)
8087**	Ungültiger Wert für die Angabe der Nanosekunde (Byte 8 bis 11 im Format DTL)
80B0	Die Echtzeituhr ist ausgefallen
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	
** Nur bei Angabe von Datum und Uhrzeit am Parameter IN im Format DTL	

Beispiel

In dem folgenden Beispiel stellen Sie Datum und Uhrzeit der CPU-Uhr ein. Der verwendete Datentyp ist DATE AND TIME.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

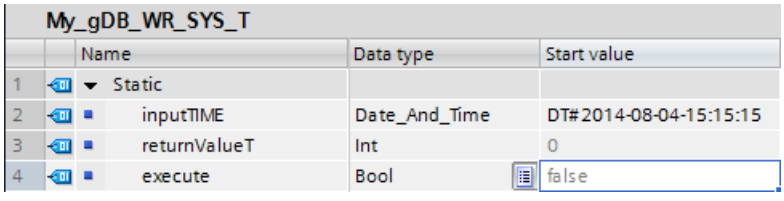

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp DATE AND TIME aus.

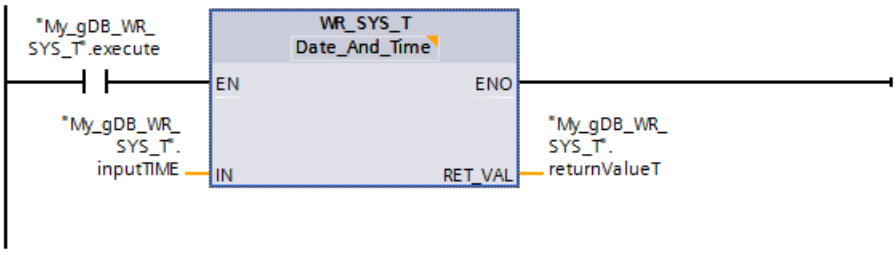

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "WR_SYS_T" ausgeführt. Die Baugruppenzeit der CPU-Uhr wird mit der einzustellenden Zeit ("inputTIME") überschrieben. Am Ausgangsparameter RET_VAL ("returnValueT") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

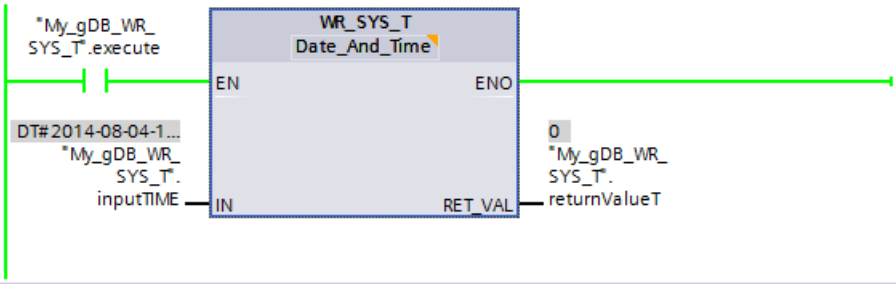

Ob die neue Baugruppenzeit ("inputTIME") von der CPU-Uhr korrekt übernommen wurde, stellen Sie wie folgt fest:

- Über das Display einer CPU S7-1500: Dazu navigieren Sie am CPU-Display zu "Einstellungen > Datum & Uhrzeit > Allgemein".
- Über TIA Portal: Sie lesen über die Anweisung "RD_SYS_T (Seite 2193)" die Baugruppenzeit der CPU-Uhr aus.
- Über TIA Portal: Sie navigieren zum Eintrag "Online & Diagnose" der CPU und öffnen das Register "Funktionen > Uhrzeit einstellen".
  Bei der CPU-Uhr ist die Baugruppenzeit als die koordinierte Weltzeit (UTC) eingestellt. Im TIA Portal ist in diesem Beispiel die Mitteleuropäische Zeitzone als Lokalzeit eingestellt. Dementsprechend wird im TIA Portal, im Eintrag "Online & Diagnose", zu der einzustellenden Zeit ("inputTIME") eine Stunde dazu addiert. Eine weitere Stunde wird bei Verwendung der Sommerzeit addiert. Die errechnete Lokalzeit wird in 12 Stunden-Darstellung ausgegeben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RD_SYS_T: Uhrzeit lesen

Beschreibung

Mit der Anweisung lesen Sie das aktuelle Datum und die aktuelle Uhrzeit (Baugruppenzeit) der CPU-Uhr aus.

Die ausgelesenen Daten werden am Ausgangsparameter OUT der Anweisung ausgegeben. Im ausgegebenen Wert sind dabei keine Angaben zur lokalen Zeitzone oder Sommerzeit enthalten.

Hinweis

Baugruppenzeit der CPU-Uhr

Die CPU-Uhr interpretiert die Baugruppenzeit als die koordinierte Weltzeit (UTC). Dementsprechend wird die Baugruppenzeit stets ohne die Faktoren "lokale Zeitzone" oder "Sommerzeit" in der CPU-Uhr hinterlegt. Die CPU-Uhr berechnet dann auf Basis der Baugruppenzeit die Lokalzeit der CPU-Uhr.

Die Baugruppenzeit der CPU-Uhr dient als Vorlage für alle zeitlichen Prozesse, ausgehend von der CPU. Beispiele für die Verwendung sind:

- Berechnung der Lokalzeit der CPU-Uhr auf Basis der Baugruppenzeit
- Darstellung der Baugruppenzeit unter "Online & Diagnose" in Lokalzeit
- Bausteineigenschaften
- Einträge im Diagnosepuffer der CPU

Am Ausgang RET_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_SYS_T":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
RET_VAL	Return	INT	INT	E, A, M, D, L, P	Status der Anweisung
OUT	Output	DTL	DT, DTL, LDT	E, A, M, D, L, P *	Datum und Uhrzeit der CPU
* Die Datentypen DT und DTL können für die Speicherbereiche Eingang, Ausgang und Merker nicht verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
8081	Die ausgelesenen Daten können nicht gespeichert werden, da sie für den gewählten Datentyp des Parameters OUT außerhalb des zulässigen Bereichs liegen. • Bei DT: min. DT#1990-01-01-0:0:0, max. DT#2089-12-31-23:59:59.999 • Bei LDT: min. LDT#1970-01-01-0:0:0.000000000, max. LDT#2262-04-11-23:47:16.854775807 • Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2262-04-11-23:47:16.854775807
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie die Baugruppenzeit der CPU-Uhr aus. Der verwendete Datentyp ist DATE AND TIME.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

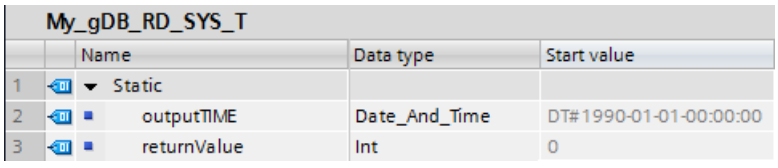

My_gDB_RD_SYS_T

	Name	Data type	Start value
1	Static		
2	outputTIME	Date_And_Time	DT#1990-01-01-00:00:00
3	returnValue	Int	0

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp DATE AND TIME aus.

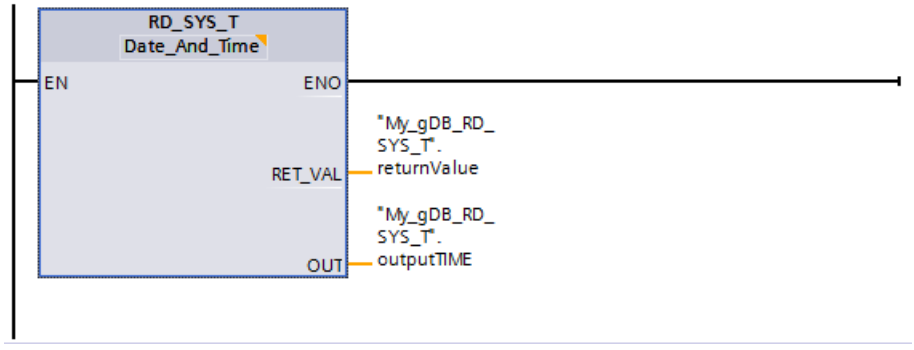

Die Baugruppenzeit der CPU-Uhr wird ausgelesen und am Ausgangsparameter OUT ("outputTIME") angezeigt. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Ausführung ohne Fehler verlief.

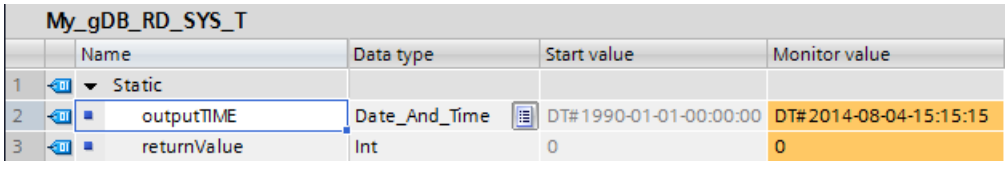

My_gDB_RD_SYS_T

	Name	Data type	Start value	Monitor value
1	Static			
2	outputTIME	Date_And_Time	DT#1990-01-01-00:00:00	DT#2014-08-04-15:15:15
3	returnValue	Int	0	0

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RD_LOC_T: Lokalzeit lesen

Beschreibung

Mit der Anweisung lesen Sie die aktuelle Lokalzeit aus der CPU-Uhr aus und geben diese am Ausgang OUT aus. Zur Ausgabe der Lokalzeit werden die Angaben zur Zeitzone sowie zum Beginn der Sommer- und Winterzeit verwendet, die Sie bei der Konfiguration der CPU-Uhr eingestellt haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_LOC_T":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
RET_VAL	Return	INT	INT	E, A, M, D, L, P	Status der Anweisung
OUT	Output	DTL	DT, LDT, DTL	E, A, M, D, L, P *	Lokalzeit
* Die Datentypen DT und DTL können für die Speicherbereiche Eingang, Ausgang und Merker nicht verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
0001	Kein Fehler. Lokalzeit wird als Sommerzeit ausgegeben.
8080	Die Lokalzeit ist nicht auslesbar.
8081	Die aktuelle Lokalzeit kann nicht gespeichert werden, da sie für den gewählten Datentyp des Parameters OUT außerhalb des zulässigen Bereichs liegt. • Bei DT: min. DT#1990-01-01-0:0:0, max. DT#2089-12-31-23:59:59.999 • Bei LDT: min. LDT#1970-01-01-0:0:0.000000000, max. LDT#2262-04-11-23:47:16.854775807 • Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2262-04-11-23:47:16.854775807
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie die Lokalzeit der CPU-Uhr aus. Der verwendete Datentyp ist DATE AND TIME.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

My_gDB_RD_LOC_T

	Name	Data type	Start value
1	Static		
2	outputLocTIME	Date_And_Time	DT#1990-01-01-00:00:00
3	returnValue	Int	0

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp DATE AND TIME aus.

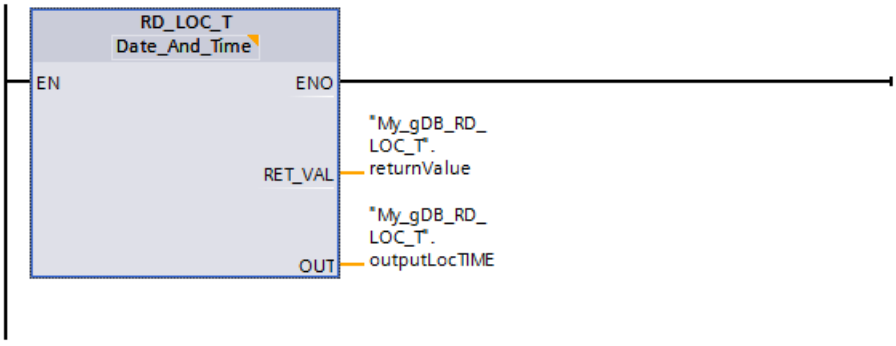

Die Lokalzeit der CPU-Uhr wird ausgelesen und am Ausgangsparameter OUT ("outputLocTIME") angezeigt. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief und die Lokalzeit bei diesem Aufruf als Sommerzeit ausgegeben wird.

My_gDB_RD_LOC_T

	Name	Data type	Start value	Monitor value
1	Static			
2	outputLocTIME	Date_And_Time	DT#1990-01-01-00:00:00	DT#2014-08-04-16:15:15.001
3	returnValue	Int	0	1

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

WR_LOC_T: Lokalzeit schreiben

Beschreibung

Mit der Anweisung "WR_LOC_T" stellen Sie Datum und Uhrzeit der CPU-Uhr ein. Die Angabe zu Datum und Uhrzeit geben Sie als Lokalzeit am Eingangsparameter LOCTIME an.

Der Wert muss in dem folgenden Bereich liegen:

- Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2200-12-31 23:59:59.999999999
- Bei LDT: min. LDT#1970-01-01-0:0:0.000000000, max. LDT#2200-12-31 23:59:59.999999999

Die Granularität der Zeitangabe für die Lokalzeit und die Systemzeit ist produktspezifisch und beträgt mindestens eine Millisekunde. Eingangswerte am Parameter LOCTIME, die kleiner sind als von der CPU unterstützt, werden bei der Berechnung der Systemzeit gerundet.

Am Ausgangsparameter RET_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_LOC_T":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
LOCTIME	Input	DTL	DTL, LDT	E, A, M, D, L, P oder Konstante *	Lokalzeit
DST	Input	BOOL	BOOL	E, A, M, D, L, P, T, Z oder Konstante	**D**aylight **S**aving **T**ime Wird nur während der "doppelten Stunde" bei der Umschaltung zur Winterzeit ausgewertet. • TRUE = Sommerzeit (erste Stunde) • FALSE = Winterzeit (zweite Stunde)
RET_VAL	Return	INT	INT	E, A, M, D, L, P	Fehlermeldung (siehe "Parameter RET_VAL")
* Der Datentyp DTL kann für die Speicherbereiche Eingang, Ausgang und Merker nicht verwendet werden.					

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter DST (Daylight Saving Time)

Der Parameter DST wird nur während der Zeitumstellung von Sommer- auf Winterzeit ausgewertet:

- Verwendung von DST bei der Umstellung von Sommer- auf Winterzeit
  Die Uhr wird morgens um 03:00[(1)] Uhr um eine Stunde zurückgestellt wird. Das bedeutet, dass es die Stunde zwischen 02:00:00:000000000 Uhr und 02:59:59:999999999 Uhr zweimal gibt.
  Bei allen Zeitangaben für LOCTIME, die zwischen 02:00:00:000000000 Uhr und 02:59:59:999999999 Uhr liegen, müssen Sie daher festlegen, ob sich die Zeitangabe auf den Zeitpunkt vor oder nach der Zeitumstellung bezieht. Dies erfolgt über den Parameter DST:
  - Bei DST=TRUE liegt die Zeitangabe in der ersten der beiden Stunden, also noch in der Sommerzeit.
  - Bei DST=FALSE liegt die Zeitangabe in der zweiten der beiden Stunden, also bereits in der Winterzeit.

  Bei allen Zeitangaben für LOCTIME, welche außerhalb der doppelten Stunde liegen, ist DST ohne Bedeutung.
- Umstellung von Winter- auf Sommerzeit
  Bei der Umstellung von Winter- auf Sommerzeit ist der Parameter DST irrelevant.
  Die Uhr wird morgens um 02:00[(1)] Uhr um eine Stunde vorgestellt. Das bedeutet, dass es die Stunde zwischen 02:00:00:000000000 Uhr und 02:59:59:999999999 Uhr nicht gibt.
  Wenn Sie an LOCTIME eine Zeitangabe innerhalb dieser Stunde vorgeben, wird der Fehlercode W#16#8089 ausgegeben.

[(1)]Die Beschreibung geht hier von der Mitteleuropäischen Zeit (MEZ) aus. Der Umschaltzeitpunkt bei MEZ ist 2:00 Uhr morgens (Beginn der Sommerzeit), bzw. 3:00 Uhr morgens (Ende der Sommerzeit). Für andere Zeitzonen können entsprechend andere Umschaltzeitpunkte gelten.

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler.
8080	Ungültiger Wert am Parameter LOCTIME.
8081	Angegebener Zeitwert am Parameter LOCTIME außerhalb des zulässigen Wertebereichs: • Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2200-12-31 23:59:59.999999999 • Bei LDT: min. LDT#1970-1-1-0:0:0.000000000, max. LDT#2200-12-31 23:59:59.999999999
8082**	Ungültiger Wert für die Angabe des Monats (Byte 2 im Format DTL).
8083**	Ungültiger Wert für die Angabe des Tages (Byte 3 im Format DTL).
8084**	Ungültiger Wert für die Angabe der Stunde (Byte 5 im Format DTL).
8085**	Ungültiger Wert für die Angabe der Minute (Byte 6 im Format DTL).
8086**	Ungültiger Wert für die Angabe der Sekunde (Byte 7 im Format DTL).
8087**	Ungültiger Wert für die Angabe der Nanosekunde (Byte 8 bis 11 im Format DTL).
8089	Zeitwert existiert nicht (abgelaufene Stunde bei der Umschaltung zur Sommerzeit).
80B0	Die Echtzeituhr ist ausgefallen.

Fehlercode* (W#16#....)	Beschreibung
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	
** Nur bei Lokalzeitangabe am Parameter LOCTIME im Format DTL.	

Beispiel

In dem folgenden Beispiel stellen Sie Lokalzeit der CPU-Uhr ein. Der verwendete Datentyp ist DTL.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

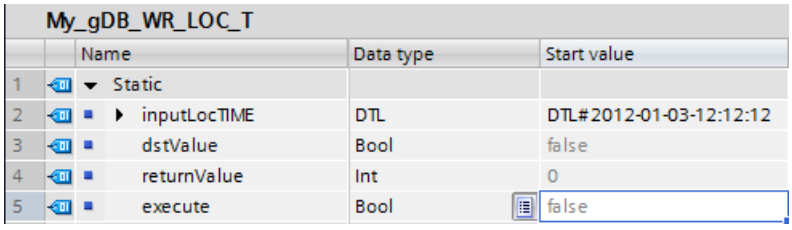

My_gDB_WR_LOC_T

	Name	Data type	Start value
1	Static		
2	inputLocTIME	DTL	DTL#2012-01-03-12:12:12
3	dstValue	Bool	false
4	returnValue	Int	0
5	execute	Bool	false

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp DTL aus.

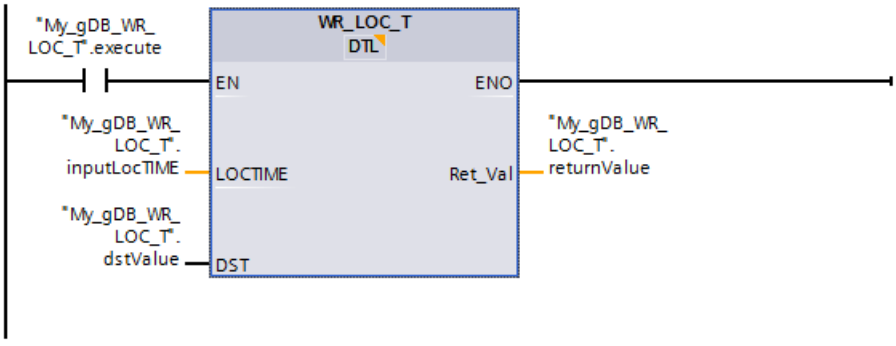

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "WR_LOC_T" ausgeführt. Die Lokalzeit der CPU-Uhr wird mit der einzustellenden Zeit ("inputLocTIME") überschrieben. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass der Vorgang ohne Fehler verlief. Der Eingangsparameter DST ("dstValue") gibt an, dass sich die Zeitangabe auf die Winterzeit bezieht. Dieser Parameter ist nur relevant zur "doppelten Stunde" (siehe Beschreibung "Parameter DST").

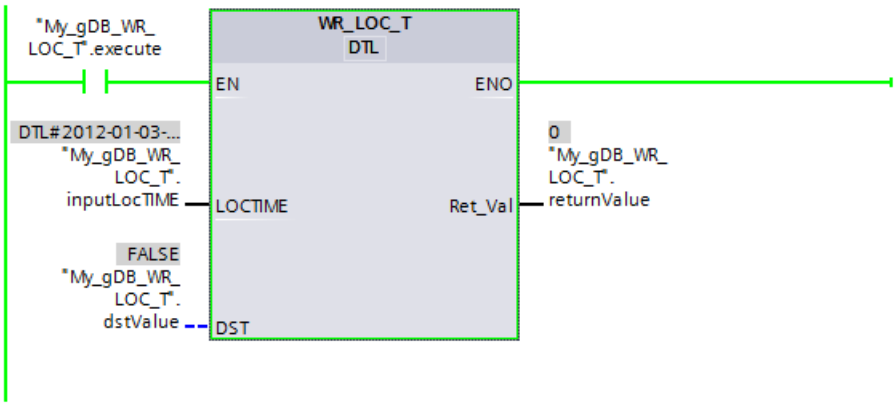

Ob die neue Lokalzeit ("inputLocTIME") von der CPU-Uhr korrekt übernommen wurde, stellen Sie wie folgt fest:

- Über das Display einer CPU S7-1500: Dazu navigieren Sie am CPU-Display zu "Einstellungen > Datum & Uhrzeit > Allgemein".
- Über TIA Portal: Sie lesen über die Anweisung "RD_LOC_T (Seite 2195)" die Lokalzeit der CPU-Uhr aus.
- Über TIA Portal: Sie navigieren zum Eintrag "Online & Diagnose" der CPU und öffnen das Register "Funktionen > Uhrzeit einstellen".
  Die Lokalzeit wird in einer 12 Stunden-Darstellung ausgegeben.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SET_TIMEZONE: Zeitzone setzen

Beschreibung

Mit der Anweisung "SET_TIMEZONE" stellen Sie die Parameter für die lokale Zeitzone und die Sommer-/Winterzeitumschaltung ein.

Die Einstellungen, die Sie mit der Anweisung "SET_TIMEZONE" vornehmen, entsprechen den Einstellungen zur Uhrzeit in den Eigenschaften der CPU. Zur Ausführung der Anweisung "SET_TIMEZONE" hinterlegen Sie die entsprechenden Parameter in dem Systemdatentyp TimeTransformationRule.

Anhand der Einstellungen für die Zeitzone und die Sommer-/Winterzeitumschaltung wird die Lokalzeit auf Basis der Systemzeit berechnet. Die Systemzeit der CPU ist die UTC-Zeit. Zur Kommunikation innerhalb des Systems wird ausschließlich die Systemzeit verwendet.

Hinweis

Aufrufen der Anweisung "SET_TIMEZONE"

Die Anweisung "SET_TIMEZONE" schreibt intern Daten in den Ladespeicher der CPU. Dementsprechend gilt die Änderung der Zeitzone auch noch nach einem Netzaus. Ein wiederholtes Aufrufen der Anweisung ist unnötig.

Rufen Sie die Anweisung jedes Mal dann einmalig auf, wenn die Zeitzone geändert werden muss. Beispielsweise empfehlenswert ist der Aufruf von "SET_TIMEZONE" in einem Anlauf-OB.

Hinweis

Verwendung mit CPU der Baureihe S7-1500

Die Anweisung "SET_TIMEZONE" ist bei einer CPU der Baureihe S7-1500 erst ab der Firmware-Version V1.7 nutzbar.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_TIMEZONE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, P oder Konstante	Einstellung der Parameter, die am Parameter TimeZone hinterlegt sind.
TimeZone	Input	TimeTransformation-Rule	D, L	Am Parameter TimeZone verschalten Sie den Systemdatentyp TimeTransformation-Rule (siehe unten).
DONE	Output	BOOL	E, A, M, D, L, P	Zustandsparameter: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L, P	Zustandsparameter: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L, P	Zustandsparameter: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L, P	Am Parameter STATUS werden detaillierte Fehler- und Statusinformationen ausgegeben. Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter TimeZone

In dem Systemdatentyp TimeTransformationRule hinterlegen Sie die Parameter für die lokale Zeitzone und die Sommer-/Winterzeitumschaltung

Die TimeTransformationRule erstellen Sie, indem Sie in einem Datenbaustein oder der lokalen Schnittstelle eines Funktionsbausteins als Datentyp TimeTransformationRule eingeben.

Die TimeTransformationRule hat den folgenden Aufbau:

Name	Datentyp	Beschreibung
TimeTransformationRule	STRUCT	

Name	Datentyp	Beschreibung
Bias	INT	Zeitdifferenz zwischen Lokalzeit und Systemzeit (UTC) in Minuten. Der Wert muss zwischen -720 und +780 Minuten (-12 bis +13 Stunden) betragen. Der Wert (UTC -12 bis +13 Stunden) entspricht den Zeitzonen, die Sie in den Eigenschaften der CPU festlegen.
DaylightBias	INT	Zeitunterschied zwischen Winter- und Sommerzeit in Minuten. Der Wert muss zwischen 0 und 120 Minuten betragen. • Der Wert "0" deaktiviert die Umschaltung zwischen Sommer- und Winterzeit. Die Werte für "DaylightStart..." und "StandardStart..." werden auf "0" gesetzt. Es wird nur der Wert für Bias ausgewertet (Zeitdifferenz Lokalzeit / Systemzeit). • Wenn der Wert nicht "0" ist, werden alle Variablen der Struktur TimeTransformationRule ausgewertet. Bei einer ungültigen Eingabe wird am Parameter STATUS der Fehlercode 808F ausgegeben.
Angabe des Zeitpunkts zur Umstellung auf Sommerzeit. Die folgenden Zeitangaben beziehen sich immer auf die Lokalzeit.		
DaylightStartMonth	USINT	Monat der Umstellung auf Sommerzeit: 1 = Januar 2 = Februar 3 = März ... 12 = Dezember
DaylightStartWeek	USINT	Woche der Umstellung auf Sommerzeit. 1 = Erstes Auftreten des Wochentags im Monat ... 5 = Letztes Auftreten des Wochentags im Monat
DaylightStartWeek-day	USINT	Wochentag der Sommerzeitumstellung: 1 = Sonntag ... 7 = Samstag
DaylightStartHour	USINT	Stunde der Sommerzeitumstellung
DaylightStartMinute	USINT	Minute der Sommerzeitumstellung
Angabe des Zeitpunkts zur Umstellung auf Winterzeit Die folgenden Zeitangaben beziehen sich immer auf die Lokalzeit.		
StandardStartMonth	USINT	Monat der Umstellung auf Winterzeit: 1 = Januar 2 = Februar 3 = März ... 12 = Dezember
StandardStartWeek	USINT	Woche der Umstellung auf Winterzeit: 1 = Erstes Auftreten des Wochentags im Monat ...

Name		Datentyp	Beschreibung
			5 = Letztes Auftreten des Wochentags im Monat
	StandardStartWeek-day	USINT	Wochentag der Winterzeitumstellung: 1 = Sonntag ... 7 = Samstag
	StandardStartHour	USINT	Stunde der Winterzeitumstellung
	StandardStartMinute	USINT	Minute der Winterzeitumstellung
	TimeZoneName	STRING[80]	Wird nicht benutzt: Die parametrierte Zeichenfolge wird ignoriert und nicht in die CPU-internen Daten geschrieben.

Parameter STATUS

Fehlercode* (W#16#....)	Beschreibung
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
808F	Aufbau, Inhalt oder die Datentypen der TimeTransformationRule am Parameter TimeZone sind ungültig oder inkonsistent.
80C3	Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an gleichzeitigen Bausteinaufrufen. Vor der Beendigung von mindestens einem Bausteinaufruf wird die "SET_TIMEZONE" nicht ausgeführt.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel stellen Sie die Parameter für die lokale Zeitzone und die Sommer-/Winterzeitumschaltung ein. Das Beispiel rufen Sie vorzugsweise in einem Anlauf-OB auf.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein Folgendes an: Die Struktur "timezone" (mit dem Datentyp TimeTransformationRule) sowie fünf weitere Variablen.

Hinweis

Wenn Sie das Beispiel in einem Anlauf-OB aufrufen wollen: Setzen Sie den Startwert der Variable "execute" auf "TRUE".

SLI_gDB_SET_TIMEZONE			
	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	timezone	TimeTransformatio...	
4	Bias	Int	120
5	DaylightBias	Int	60
6	DaylightStartMonth	USInt	3
7	DaylightStartWeek	USInt	5
8	DaylightStartWeekday	USInt	7
9	DaylightStartHour	USInt	2
10	DaylightStartMinute	USInt	0
11	StandardStartMonth	USInt	10
12	StandardStartWeek	USInt	5
13	StandardStartWeekday	USInt	1
14	StandardStartHour	USInt	3
15	StandardStartMinute	USInt	0
16	TimeZoneName	String[80]	'My_GMT+'
17	modeDONE	Bool	false
18	modeBUSY	Bool	false
19	modeERROR	Bool	false
20	statusTime	Word	16#0
21	memErrStatus	Word	16#0

Für die Verschaltung des Ausgangsparameters DONE legen Sie eine lokale Variable an.

9	statDone	Bool	false	Non-retain

Netzwerk 1: Die Parameter der Anweisung "SET_TIMEZONE" verschalten Sie wie folgt.

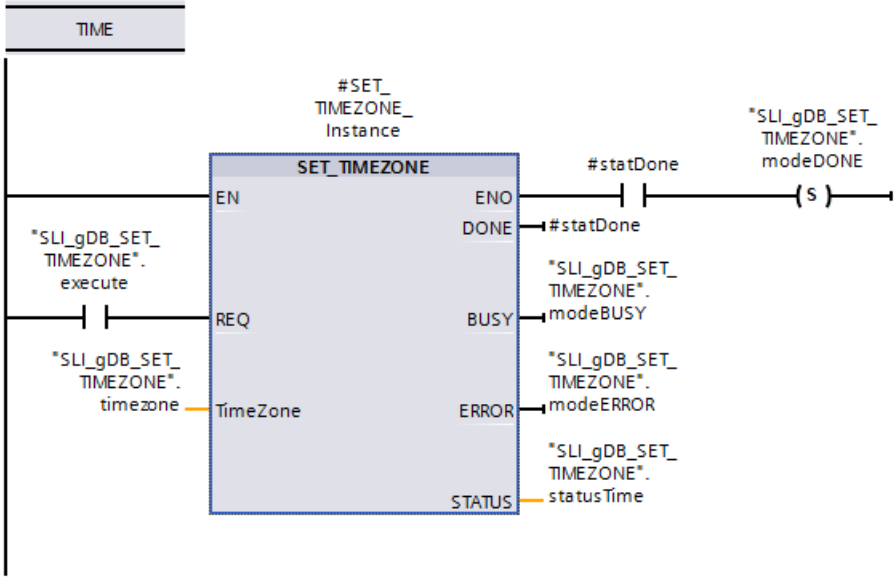

Zudem legen Sie folgende, weitere Verschaltungen an:

Netzwerk 2: Im Fehlerfall speichern Sie den Status in der Variable "memErrStatus".

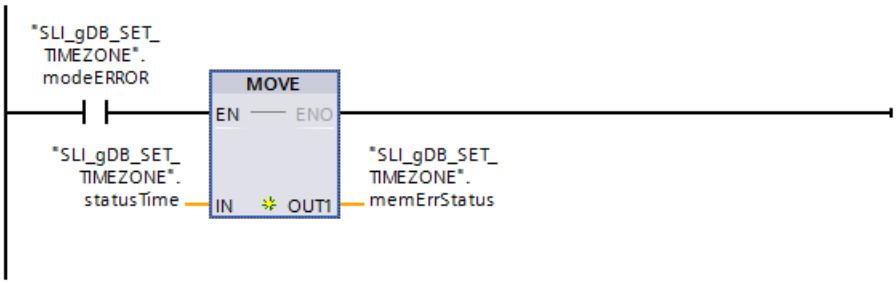

Netzwerk 3: Den Eingangsparameter REQ ("execute") lassen Sie wie folgt nach erfolgreicher Verarbeitung automatisch zurücksetzen.

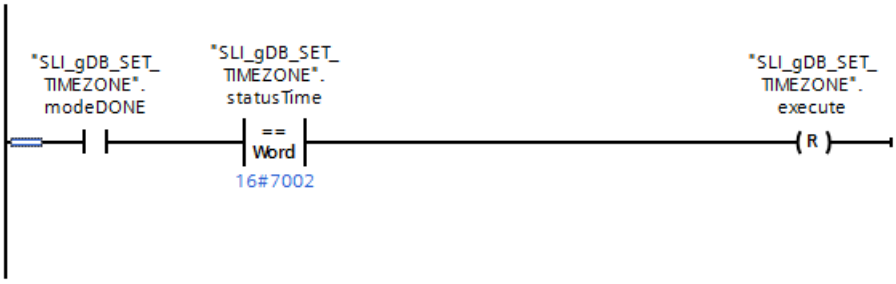

Netzwerk 4: Damit die Anweisung "SET_TIMEZONE" bei Aufruf in einem Anlauf-OB vollständig verarbeitet wird, verwenden Sie die Anweisungen JMP (im Netzwerk 4) und Label (siehe Netzwerk 1).

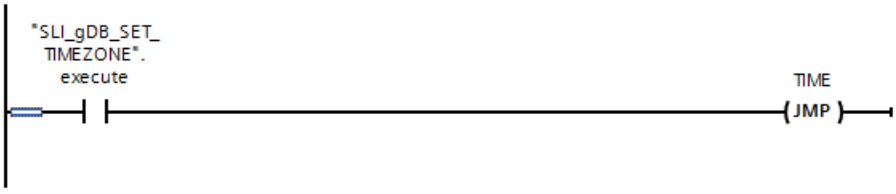

Hinweis

Wenn Sie das Beispiel in einem Anlauf-OB aufrufen wollen, muss sich die CPU für die Verarbeitung des Beispiels im Anlauf befinden. Setzen Sie die CPU in den Modus STOP, bevor Sie das Beispiel in die CPU laden. Setzen Sie die CPU anschließend in den Modus RUN.

Wenn der Eingangsparameter REQ ("execute") den Signalzustand "TRUE" liefert, werden die Daten für die Zeitzone der CPU-Uhr mit den einzustellenden Daten ("timezone") überschrieben. Dies bedeutet außerdem:

- Der Ausgangsparameter BUSY ("modeBUSY") geht in den Signalzustand "TRUE". Nach der Verarbeitung zeigt der Ausgangsparameter BUSY den Wert "FALSE" und der Ausgangsparameter DONE ("#statDone") zeigt den Wert "TRUE". Der Erfolgszustand wird in der Variable "modeDONE" gespeichert.
- Am Ausgangsparameter STATUS ("statusTime") wird angezeigt, wie die Verarbeitung verläuft*. (*Die Auftragsbearbeitung beginnt (Wert "7001") und zeigt nachfolgend die Anweisung als bereits aktiv (Wert "7002").)
- Am Ausgangsparameter ERROR ("modeERROR") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft (der Signalzustand ist "FALSE").

SLI_gDB_SET_TIMEZONE

	Name	Data type	Start value	Monitor value
1	Static			
2	execute	Bool	false	FALSE
3	timezone	TimeTransformatio...		
4	Bias	Int	120	120
5	DaylightBias	Int	60	60
6	DaylightStartMonth	USInt	3	3
7	DaylightStartWeek	USInt	5	5
8	DaylightStartWeekday	USInt	7	7
9	DaylightStartHour	USInt	2	2
10	DaylightStartMinute	USInt	0	0
11	StandardStartMonth	USInt	10	10
12	StandardStartWeek	USInt	5	5
13	StandardStartWeekday	USInt	1	1
14	StandardStartHour	USInt	3	3
15	StandardStartMinute	USInt	0	0
16	TimeZoneName	String[80]	'My_GMT+'	'My_GMT+'
17	modeDONE	Bool	false	TRUE
18	modeBUSY	Bool	false	FALSE
19	modeERROR	Bool	false	FALSE
20	statusTime	Word	16#0	16#7000
21	memErrStatus	Word	16#0	16#0000

Hinweis: Die Anweisung "SET_TIMEZONE" ist pegelgetriggert. Nur wenn der Eingangsparameter REQ ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung ausgeführt.

Nach erfolgreicher Verarbeitung von SET_TIMEZONE ("modeDONE" ist "TRUE") wird der Eingangsparameter REQ ("execute") automatisch auf den Wert "FALSE" zurückgesetzt. Dazu wird für "statusTime" der Wert "16#7002" abgefragt. Nach dem Rücksetzen wird der Statuswert "16#7000" ausgegeben.

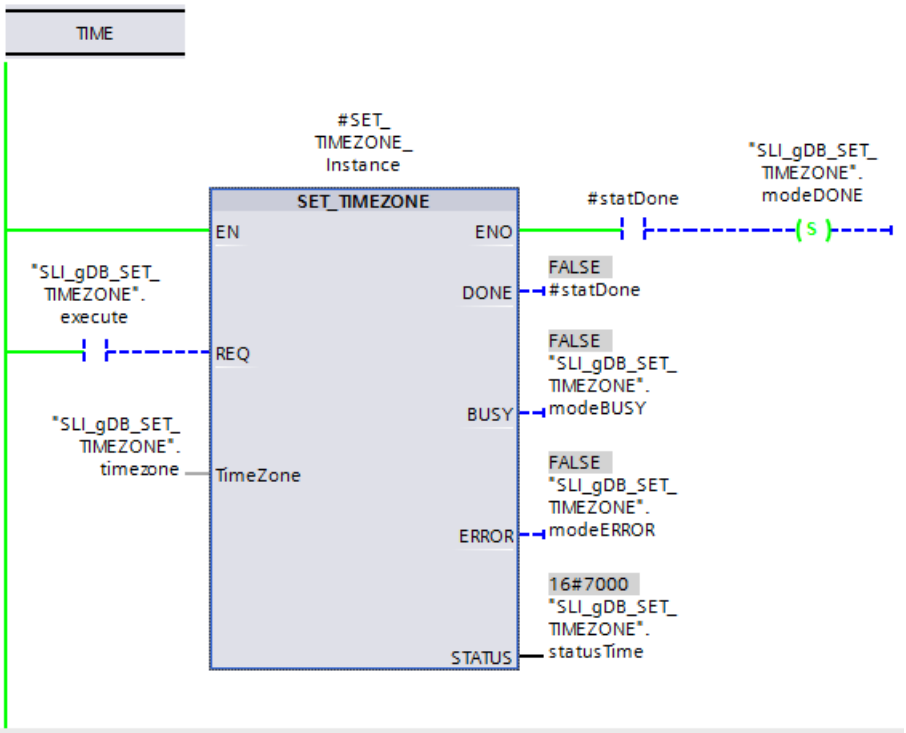

Ob die einzustellenden Daten ("timezone") von der CPU-Uhr korrekt übernommen wurden, stellen Sie wie folgt fest:

- Über das Display einer CPU S7-1500: Dazu navigieren Sie am CPU-Display zu "Einstellungen > Datum & Uhrzeit > Sommerzeit".
- Über TIA Portal: Sie lesen über die Anweisung "RD_LOC_T (Seite 2195)" die Lokalzeit der CPU-Uhr aus.
- Über TIA Portal: Sie lesen über die Anweisung "RD_SYS_T (Seite 2193)" die Baugruppenzeit der CPU-Uhr aus.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

SNC_RTCB: Uhrzeitslaves synchronisieren

Definition: Synchronisation von Uhrzeitslaves

Unter der Synchronisation von Uhrzeitslaves versteht man die Übertragung des Datums und der Uhrzeit vom Uhrzeitmaster eines Bussegments auf sämtliche Uhrzeitslaves dieses Bussegments.

Beschreibung

Mit der Anweisung synchronisieren Sie alle lokal an der CPU oder lokal an einem Bussegment vorhandenen Uhrzeitslaves unabhängig vom parametrierten Synchronisationsintervall (Die externe Schnittstelle eines CP oder CM gehört zu einem anderen Bussegment.). Voraussetzung für die erfolgreiche Synchronisation ist, dass "SNC_RTCB" auf einer CPU aufgerufen wird, deren Echtzeituhr für mindestens ein Bussegment als Uhrzeitmaster parametriert wurde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SNC_RTCB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Bei der Synchronisation ist kein Fehler aufgetreten.
0001	Die vorhandene Uhr wurde für kein Bussegment als Uhrzeitmaster parametriert.
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

TIME_TCK: Systemzeit lesen

Beschreibung

Mit der Anweisung "TIME_TCK" lesen Sie die Systemzeit der CPU. Die Systemzeit ist ein Zeitzähler, der von 0 bis max. 2147483647 ms zählt. Bei einem Überlauf der Systemzeit wird wieder ab "0" gezählt. Das Zeitraster und die Genauigkeit der Systemzeit beträgt 1 ms. Die Systemzeit wird nur von den Betriebszuständen der CPU beeinflusst. Die Systemzeit können Sie zum Beispiel nutzen, um die Dauer von Vorgängen durch Differenzbildung der Ergebnisse von zwei "TIME_TCK"-Aufrufen zu messen. Die Anweisung stellt keine Fehlerinformationen zur Verfügung.

Die folgende Tabelle gibt eine Übersicht, wie sich die Systemzeit abhängig von den Betriebszuständen der CPU ändert.

Betriebszustand	Systemzeit ...
Anlauf	... wird ständig aktualisiert.
RUN	
STOP	... wird angehalten und behält den aktuellen Wert.
Neustart (Warmstart)	... wird gelöscht und beginnt wieder von "0" an zu laufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TIME_TCK":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	TIME	E, A, M, D, L	Der Parameter RET_VAL enthält die gelesene Systemzeit im Bereich von 0 bis 2^{31} -1 ms.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel lesen Sie die Systemzeit der CPU aus. Den Rückgabewert geben Sie im Datentyp TIME an.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein eine Variable an.

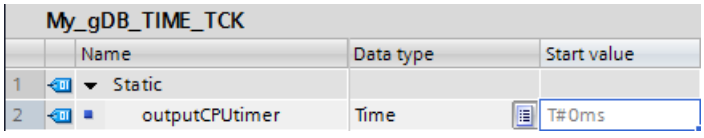

Die Parameter der Anweisung verschalten Sie wie folgt.

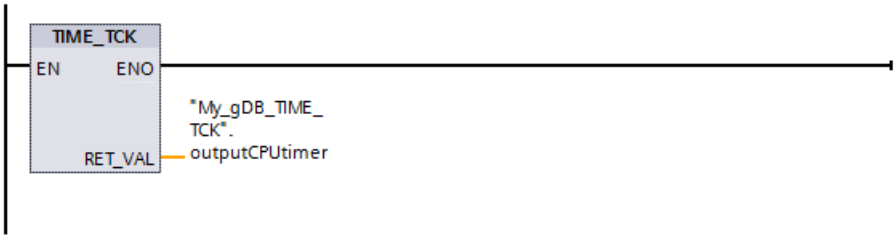

Die Systemzeit der CPU wird ausgelesen und am Ausgabeparameter RET_VAL ("outputCPUtimer") angezeigt.

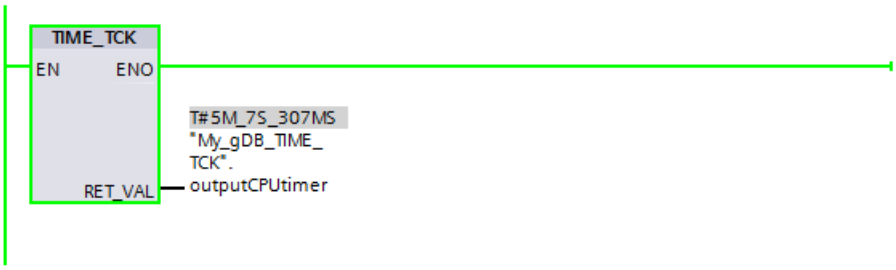

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RTM: Betriebsstundenzähler

Beschreibung

Mit der Anweisung können Sie einen 32-Bit-Betriebsstundenzähler Ihrer CPU setzen, starten, stoppen und auslesen.

Beachten Sie, dass der Betriebsstundenzähler während der Ausführung des Anwenderprogramms auch gestoppt oder neu gestartet werden kann, wodurch die gespeicherten Werte verfälscht werden können.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RTM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NR	Input	RTM	E, A, M, D, L oder Konstante	Nummer des Betriebsstundenzählers Die Nummerierung beginnt mit 0. Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den technischen Daten.
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung: • 0: auslesen (Status wird dann in CQ, aktueller Wert wird in CV eingetragen). Läuft der Betriebsstundenzähler länger als (2^31) - 1 Stunden, dann bleibt er beim letzten darstellbaren Wert stehen und liefert die Fehlermeldung "Überlauf". • 1: starten (mit dem letzten Zählerstand) • 2: stoppen • 4: setzen auf den am Parameter PV angegebenen Wert • 5: setzen auf den am Parameter PV angegebenen Wert und starten • 6: setzen auf den am Parameter PV angegebenen Wert und stoppen
PV	Input	DINT	E, A, M, D, L oder Konstante	Neuer Wert für den Betriebsstundenzähler
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
CQ	Output	BOOL	E, A, M, D, L	Status des Betriebsstundenzählers (1: läuft)
CV	Output	DINT	E, A, M, D, L	Aktueller Wert des Betriebsstundenzählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Falsche Nummer des Betriebsstundenzählers
8081	Ein negativer Wert wurde dem Parameter PV übergeben.
8082	Überlauf des Betriebsstundenzählers
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
allgemeine Fehlerinformation	Siehe auch: Fehlerauswertung über GET_ERR_ID (Seite 1935)

Beispiel

In dem folgenden Beispiel setzen Sie den Betriebsstundenzähler der CPU und lesen den Wert nach einer Stunde aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein sechs Variablen an.

My_gDB_RTM

	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	in_processValue	DInt	5
4	returnValue	Int	0
5	statusRTM	Bool	false
6	currentValue	DInt	0
7	comandMODE	Byte	5

Die Parameter der Anweisung verschalten Sie wie folgt. Am Eingangsparameter NR geben Sie die Nummer des Betriebsstundenzählers der CPU an.

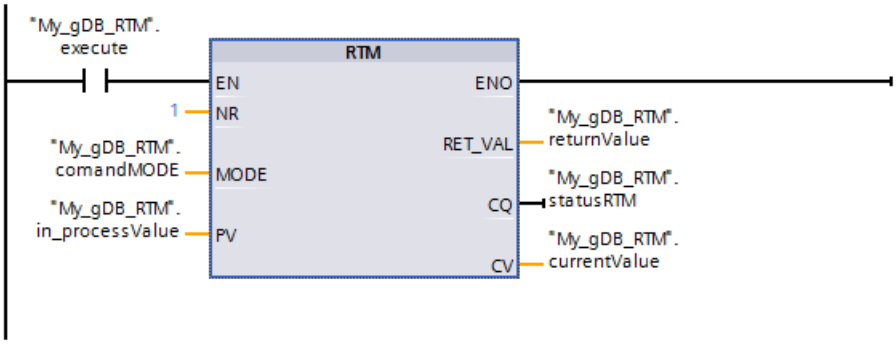

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "RTM" ausgeführt. Der Betriebsstundenzähler der CPU wird auf den einzustellenden Wert ("in_processValue") gesetzt und gestartet. Nach dem Starten des Betriebsstundenzählers setzen Sie den Wert des Eingangsparameter MODE ("comandMODE") auf "0". (Sie klicken dazu auf den Parameter und wählen "Operand steuern > auf 0 setzen".) Dadurch liest die Anweisung "RTM" nur noch den aktuellen Wert ("currentValue") des Betriebsstundenzählers aus, ohne diesen zu verändern. Der Ausgangsparameter CQ ("statusRTM") zeigt ab dem Starten des Betriebsstundenzählers, dass der Betriebsstundenzähler läuft (Wert ist "TRUE"). Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft.

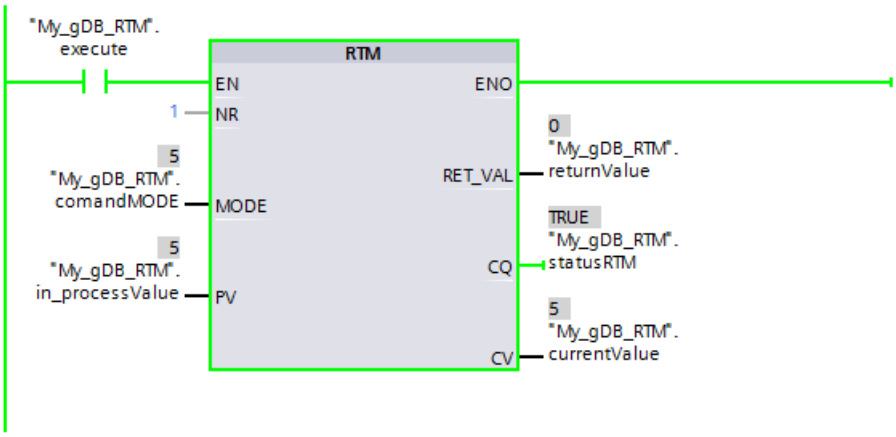

Nach einer Stunde zeigt der Ausgangsparameter CV ("currentValue") den Wert "6" an.

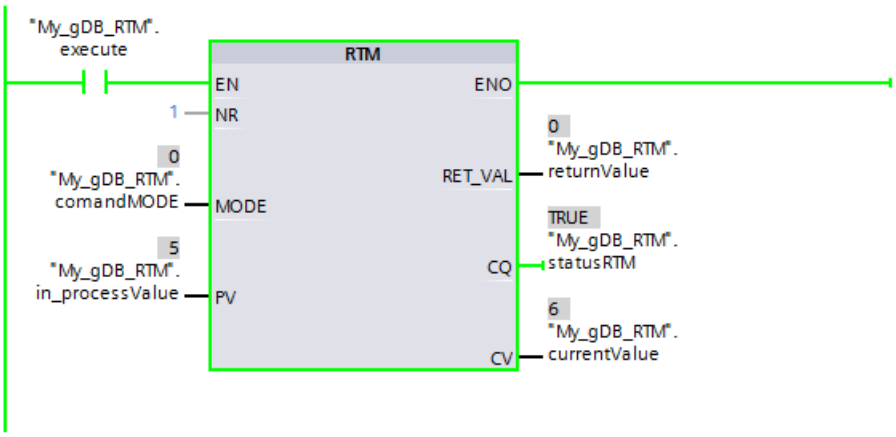

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

4.1.4.2 String + Char

S_MOVE: Zeichenkette verschieben

Beschreibung

Mit der Anweisung können Sie den Inhalt einer Zeichenkette (W)STRING von dem Parameter IN in den Datenbereich schreiben, den Sie am Parameter OUT angeben.

Zum Kopieren von Variablen des Datentyps ARRAY verwenden Sie die Anweisungen "MOVE_BLK" und "UMOVE_BLK".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S_MOVE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Quellstring
OUT	Output	STRING, WSTRING	D, L	Zielstring

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel kopieren Sie den Inhalt einer Zeichenkette vom Eingangsparameter IN auf eine andere Zeichenkette am Ausgangsparameter OUT. Der verwendete Datentyp ist STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

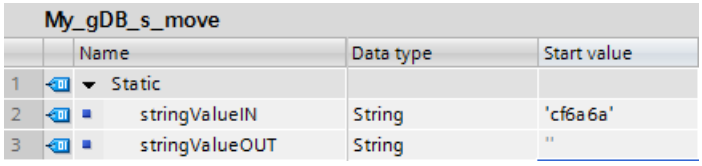

My_gDB_s_move

		Name	Data type	Start value
1		Static		
2		stringValueIN	String	'cf6a6a'
3		stringValueOUT	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

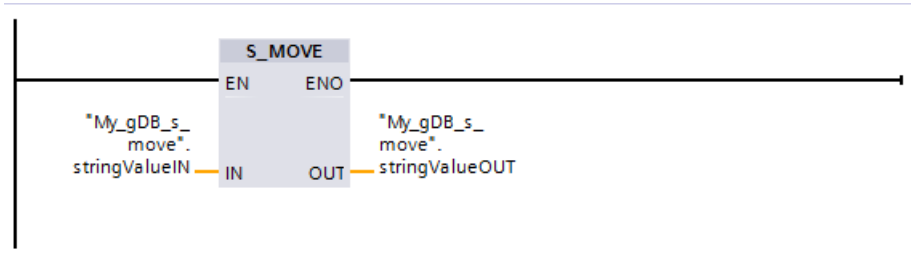

Das Ergebnis der zu kopierenden Zeichenkette ("stringValueIN") wird am Ausgangsparameter OUT ("stringValueOUT") ausgegeben.

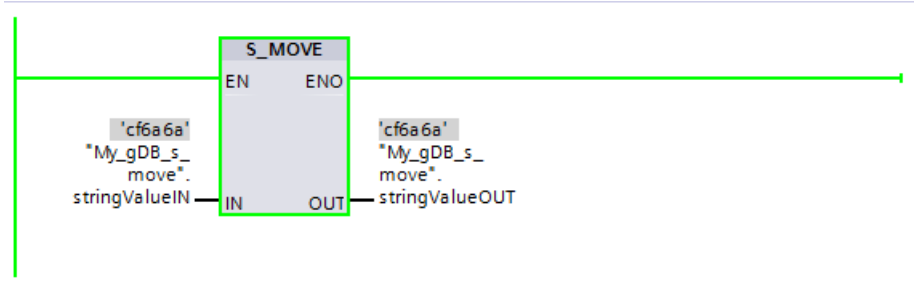

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

S_COMP: Zeichenketten vergleichen

Beschreibung

Die Anweisung vergleicht die Inhalte zweier Variablen im Format (W)STRING und gibt das Vergleichsergebnis als Rückgabewert aus. Die zu vergleichenden Variablen werden an den Eingängen IN1 und IN2 verschaltet. Die Eingangsparameter belegen Sie nur mit einer symbolisch definierten Variablen.

Über die Anweisungsbox wählen Sie die Vergleichsbedingung aus. Wenn die Vergleichsbedingung (z. B. größer oder gleich) erfüllt ist, wird am Ausgangsparameter OUT der Signalzustand auf "1" gesetzt.

Verwenden Sie eine der folgenden Vergleichsbedingungen:

Symbol	Beschreibung
EQ	Wenn die Zeichenkette am Parameter IN1 gleich der Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
NE	Wenn die Zeichenkette am Parameter IN1 ungleich der Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
GT (1)	Wenn die Zeichenkette am Parameter IN1 größer als die Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
LT (1)	Wenn die Zeichenkette am Parameter IN1 kleiner als die Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
GE (1)	Wenn die Zeichenkette am Parameter IN1 größer oder gleich der Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
LE (1)	Wenn die Zeichenkette am Parameter IN1 kleiner oder gleich der Zeichenkette am Parameter IN2 ist, führt der Rückgabewert zu Signalzustand "1".
(1) Die Zeichen werden beginnend von links über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S_COMP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING, WSTRING*	D, L oder Konstante	Eingangsvariable im Format STRING / WSTRING
IN2	Input	STRING, WSTRING*	D, L oder Konstante	Eingangsvariable im Format STRING / WSTRING
OUT	Output	BOOL	E, A, M, D, L	Vergleichsergebnis
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING / WSTRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel vergleichen Sie mit Hilfe der Vergleichsmöglichkeit "gleich" zwei Zeichenketten mit dem Datentyp STRING miteinander.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

My_gDB_s_comp

		Name	Data type	Start value
1		Static		
2		stringvalue1	String	'ThisIsAVeryLongStringWithSomeNumbers646'
3		stringvalue2	String	'ThisIsAVeryLongStringWithSomeNumbers646'
4		svalue1EQsvalue2	Bool	false

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen den Datentyp STRING und die Vergleichsmöglichkeit EQ.

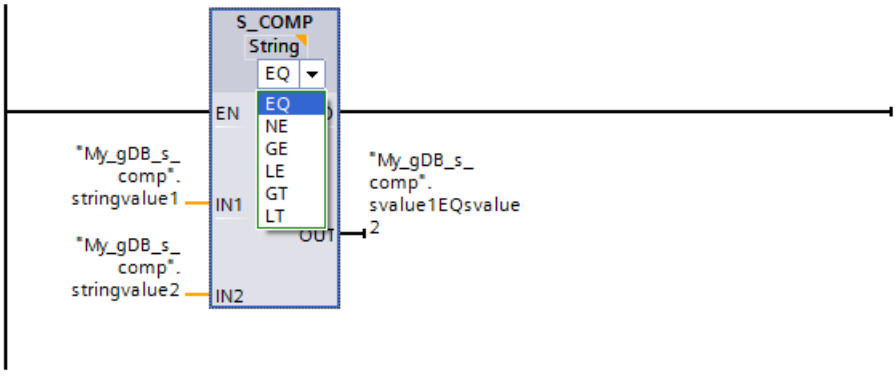

Da der Wert der ersten zu vergleichenden Zeichenkette ("stringvalue1") gleich der zweiten Zeichenkette ("stringvalue2") ist, zeigt das Vergleichsergebnis ("svalue1EQsvalue2") den Signalzustand "TRUE".

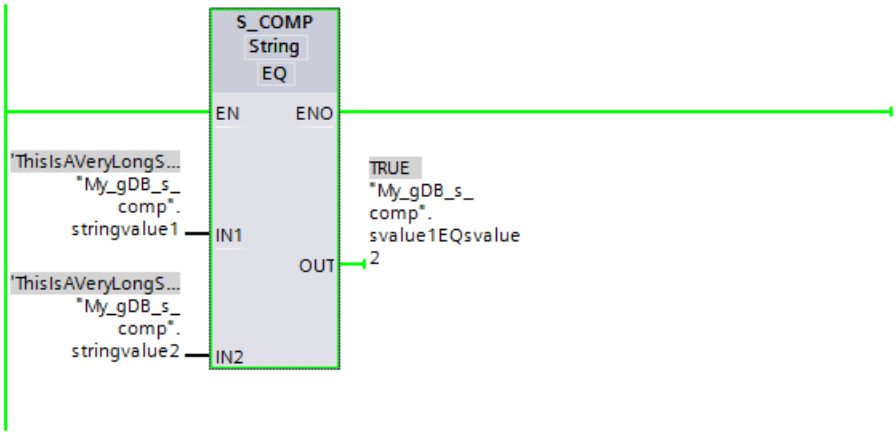

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

S_CONV: Zeichenkette konvertieren

Beschreibung

Mit der Anweisung konvertieren Sie den Wert am Eingang IN in das Datenformat, das Sie am Ausgang OUT angegeben haben. Das Ausgabeformat der Konvertierung bestimmen Sie durch die Auswahl eines Datentyps für den Ausgangsparameter OUT.

Die folgenden Konvertierungen sind möglich:

- Konvertierung einer Zeichenkette
- Konvertierung eines numerischen Werts oder Zeichens in eine Zeichenkette
- Konvertierung eines Zeichens in ein Zeichen

Hinweis

Explizite Konvertierungen

Weitere Informationen zu den expliziten Konvertierungen finden Sie hier:

- Für eine S7-1200-CPU: Explizite Konvertierungen (Seite 530)
- Für eine S7-1500-CPU: Explizite Konvertierungen (Seite 423)

Konvertierung einer Zeichenkette

- In einen numerischen Wert (Ganzzahl oder Gleitpunktzahl):
  Die Konvertierung wird für alle Zeichen der am Eingangsparameter IN angegebenen Zeichenkette ausgeführt. Erlaubte Zeichen sind die Ziffern von "0" bis "9", der Dezimalpunkt sowie das Plus- und Minus-Zeichen. Das erste Zeichen der Kette darf dabei eine gültige Ziffer oder ein Vorzeichen sein. Führende Leerzeichen und Exponentialdarstellungen werden ignoriert.
- In ein Zeichen:
  Bei der Konvertierung einer Zeichenkette in ein Zeichen wird das erste Zeichen der Zeichenkette an den Parameter OUT übertragen.
- In eine Zeichenkette

Konvertierung eines numerischen Werts oder Zeichens in eine Zeichenkette

- Das Format des zu konvertierenden numerischen Werts bestimmen Sie durch die Auswahl eines Datentyps für den Eingang IN. Geben Sie am Ausgang OUT eine gültige Variable des Datentyps (W)STRING an. Die Länge der Zeichenkette nach der Konvertierung hängt vom Wert am Eingang IN ab.
- Das Konvertierungsergebnis wird ab dem dritten Byte der Zeichenkette gespeichert. In dem ersten Byte der Zeichenkette wird die maximale Länge und im zweiten Byte die tatsächliche Länge der Zeichenkette erfasst. Positive numerische Werte werden ohne Vorzeichen ausgegeben.
- Wenn Sie den Zahlenwert 0, der als Datentyp INT oder UINT vorliegt, in eine Zeichenkette konvertieren (z. B. INT_TO_STRING(0)), hat die Zeichenkette im Ergebnis eine Länge von 6 Zeichen.

- Bei der Konvertierung eines numerischen Werts in eine Zeichenkette werden die ersten Zeichen der Zeichenkette mit Leerzeichen aufgefüllt. Die Anzahl der Leerzeichen ist abhängig von der Länge des numerischen Werts.

  Hinweis

  SCL-spezifische Information

  Bei der Konvertierung eines numerischen Werts in eine Zeichenkette werden die ersten Zeichen der Zeichenkette **nicht** mit Leerzeichen aufgefüllt.

  Wenn Sie möchten, dass die ersten Zeichen der Zeichenkette mit Leerzeichen aufgefüllt werden sollen, dann verwenden Sie bitte die Anweisung "VAL_STRG".

- Bei einer Konvertierung eines Zeichens (W)CHAR, wird das Zeichen an die erste Stelle der Zeichenkette geschrieben.

Hinweis

Exponentielle Notation bei der Konvertierung von Gleitpunktzahlen

Verwenden Sie bei der Konvertierung von Gleitpunktzahlen mit der Anweisung "S_CONV" keine exponentielle Notation ("e" oder "E"). Verwenden Sie für die Konvertierung von Gleitpunktzahlen mit exponentieller Notation stattdessen die Anweisung "STRG_VAL". Wählen Sie über den Parameter FORMAT der Anweisung das Eingangsformat der exponentiellen Notation aus.

Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "S_CONV" entsprechend der möglichen Konvertierungen:

Tabelle 4-11 Parameter bei der Konvertierung einer Zeichenkette in einen numerischen Wert:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zu konvertierender Wert
OUT	Output	CHAR, WCHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL	E, A, M, D, L	Ergebnis der Konvertierung

Tabelle 4-12 Parameter bei der Konvertierung einer Zeichenkette in eine Zeichenkette:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zu konvertierender Wert
OUT	Output	STRING, WSTRING	D, L	Ergebnis der Konvertierung (mögliche Konvertierungen: STRING nach WSTRING und umgekehrt)

Tabelle 4-13 Parameter bei der Konvertierung eines numerischen Werts oder Zeichens in eine Zeichenkette:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	CHAR, WCHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL	E, A, M, D, L oder Konstante	Zu konvertierender Wert
OUT	Output	STRING, WSTRING	D, L	Ergebnis der Konvertierung

Tabelle 4-14 Parameter bei der Konvertierung eines Zeichens in ein Zeichen:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	CHAR, WCHAR	E, A, M, D, L oder Konstante	Zu konvertierender Wert
OUT	Output	CHAR, WCHAR	E, A, M, D, L	Ergebnis der Konvertierung (mögliche Konvertierungen: CHAR nach WCHAR und umgekehrt)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

Konvertieren Sie in dem folgenden Beispiel eine Zahl mit dem Datentyp INT in eine Zeichenkette mit dem Datentyp STRING um.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

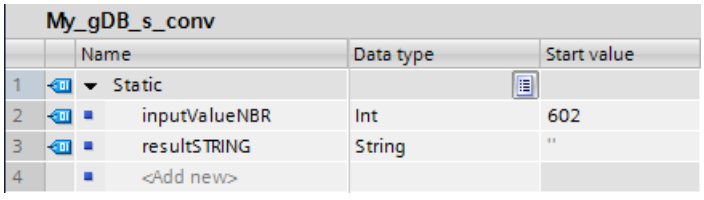

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen dabei die Datentypen aus. Bei der ersten Auswahlmöglichkeit geben Sie den Datentyp für den zu konvertierenden Wert ("inputValueNBR") an. Bei der zweiten Auswahlmöglichkeit geben Sie den Datentyp für die zu erzeugende Zeichenkette ("resultSTRING") an.

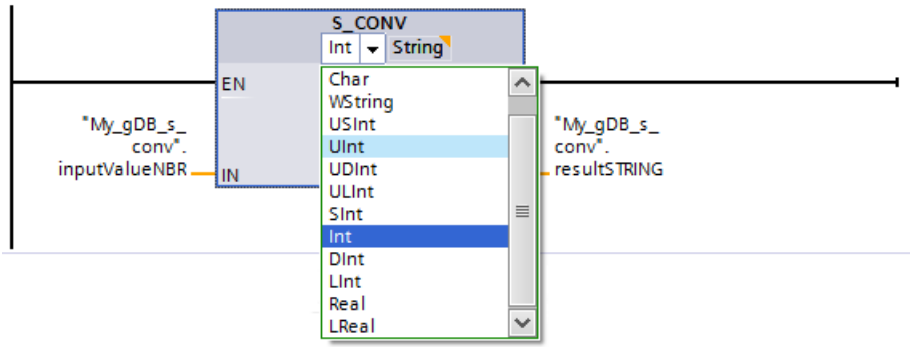

Der zu konvertierende Wert ("inputValueNBR") wird in das Ausgabeformat umgewandelt. Auf die leeren Stellen am Anfang der Zeichenkette werden Leerzeichen geschrieben. Das Ergebnis der Konvertierung wird am Ausgangsparameter OUT ("resultSTRING") als Zeichenkette ausgegeben.

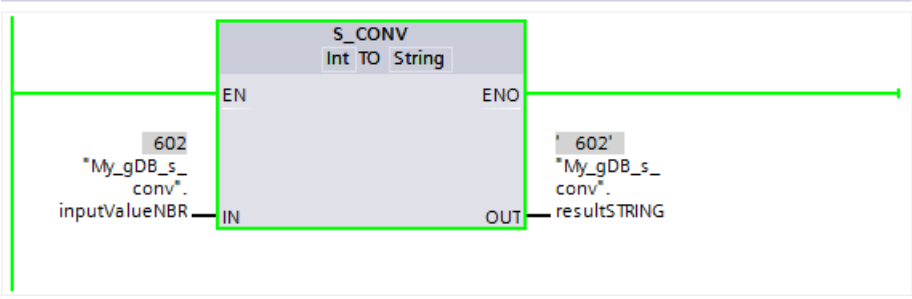

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

STRG_VAL: Zeichenkette in Zahlenwert konvertieren (Seite 2222)

STRG_VAL: Zeichenkette in Zahlenwert konvertieren

Beschreibung

Die Anweisung "STRG_VAL" wandelt eine Zahlenzeichenkette in eine Ganzzahl oder Gleitpunktzahl um:

- Die zu konvertierende Zeichenkette geben Sie am Eingangsparameter IN an.
- Das Format des Ausgabewerts bestimmen Sie durch die Auswahl eines Datentyps für den Ausgangsparameter OUT.

Erlaubte Zeichen für die Konvertierung sind die Ziffern "0" bis "9", der Dezimalpunkt, das Dezimalkomma, Notationen "E" und "e" sowie das Plus- und Minus-Zeichen. Die Konvertierung wird beendet, wenn ein ungültiges Zeichen gefunden wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "STRG_VAL":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	STRING, WSTRING	STRING, WSTRING	D, L oder Konstante	Zu konvertierende Zahlenzeichenkette
FORMAT	Input	WORD	WORD	E, A, M, D, L, P oder Konstante	Eingabeformat der Zeichen
P	Input	UINT	UINT	E, A, M, D, L, P oder Konstante	Verweis zum ersten Zeichen, das umgewandelt werden soll (erstes Zeichen = 1, der Wert "0" oder ein Wert > Länge des Strings ist ungültig)
OUT	Output	USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL	USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL	E, A, M, D, L, P	Ergebnis der Konvertierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter FORMAT

Mit dem Parameter FORMAT geben Sie vor, wie die Zeichen einer Zeichenkette zu interpretieren sind. Exponentialwerte mit der Anweisung "STRG_VAL" werden konvertiert konvertiert und dargestellt.

Die folgende Tabelle zeigt die möglichen Werte des Parameters FORMAT und deren Bedeutung:

Wert (W#16#....)	Notation	Dezimaldarstellung
0000	Dezimalbruch	"."
0001		","
0002	Exponentiell	"."
0003		","
0004 bis FFFF	Ungültige Werte	

Parameter P

Die Konvertierung beginnt ab dem Zeichen, dessen Position Sie im Parameter P angegeben haben. Wenn z. B. am Parameter P der Wert "1" angegeben ist, wird die Konvertierung ab dem ersten Zeichen der angegebenen Zeichenkette ausgeführt.

Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung einer Zeichenkette in einen numerischen Wert:

IN (STRING)	FORMAT (W#16#....)	OUT (Datentyp)	OUT (Wert)	ENO-Status
'123'	0000	INT/DINT	123	1
'-00456'	0000	INT/DINT	-456	1
'123.45'	0000	INT/DINT	123	1
'+2345'	0000	INT/DINT	2345	1
'00123AB'	0000	INT/DINT	123	1
'123'	0000	REAL	123.0	1
'-00456'	0001	REAL	-456.0	1
'+00456'	0001	REAL	456.0	1
'123.45'	0000	REAL	123.45	1
'123.45'	0001	REAL	12345.0	1
'123,45'	0000	REAL	12345.0	1
'123,45'	0001	REAL	123.45	1
'.00123AB'	0001	REAL	123.0	1
'1.23e-4'	0000	REAL	1.23	1
'1.23E-4'	0000	REAL	1.23	1
'1.23E-4'	0002	REAL	1.23E-4	1
'12,345.67'	0000	REAL	12345.67	1
'12,345.67'	0001	REAL	12.345	1
'3.4e39'	0002	REAL	W#16#7F800000	0
'-3.4e39'	0002	REAL	W#16#FF800000	0
'1.1754943e-38'	0002	REAL	0.0	1
'12345'	-/-	SINT	0	0
'A123'	-/-	-/-	0	0
''	-/-	-/-	0	0
'++123'	-/-	-/-	0	0
'+-123'	-/-	-/-	0	0

Beispiel

Konvertieren Sie in dem folgenden Beispiel eine Zahlenzeichenkette mit dem Datentyp STRING in eine Gleitpunktzahl mit dem Datentyp REAL um. Durch den Datentyp REAL wird das Ergebnis eine Länge von 32 Bits haben und ein Vorzeichen besitzen dürfen.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_STRG_VAL			
	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'567.12'
3	resultSformat	Word	16#0001
4	pointerSTRG	UInt	1
5	outputVAL	Real	0.0

Die Parameter der Anweisung verschalten Sie wie folgt. Über die linke Auswahlmöglichkeit wählen Sie den Datentyp für die Zeichenkette aus. Über die rechte Auswahlmöglichkeit wählen Sie den Datentyp für die Gleitpunktzahl aus.

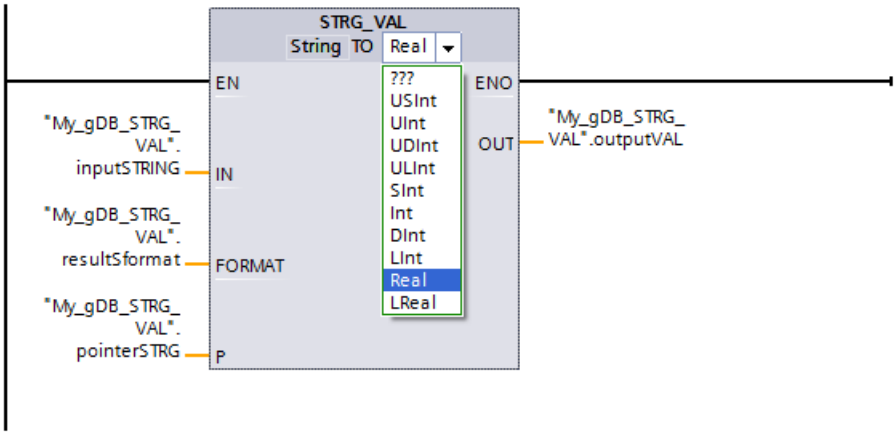

Entsprechend dem Wert "1" des Parameter P ("pointerSTRG") wird die Zahlenzeichenkette ab dem ersten Zeichen umgewandelt. Durch den Wert "0001" des Parameter FORMAT ("resultSformat") wird der Punkt in der Zahlenzeichenkette als Tausendertrennzeichen interpretiert. (Dezimaltrennzeichen für den Wert "0001" ist das Komma.) Der zu konvertierende Wert ("inputSTRING") wird am Ausgangsparameter OUT ("outputVAL") als Gleitpunktzahl ausgegeben.

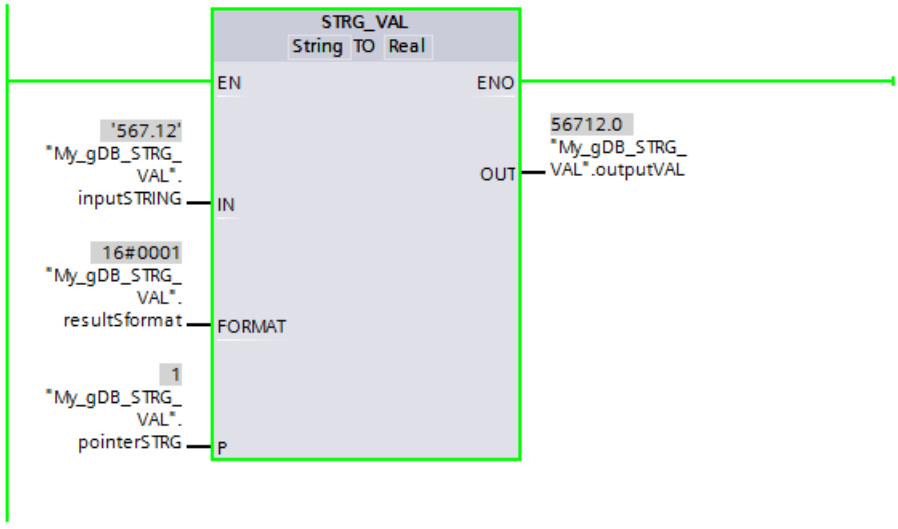

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

VAL_STRG: Zahlenwert in Zeichenkette konvertieren

Beschreibung

Mit der Anweisung "VAL_STRG" konvertieren Sie einen numerischen Wert in eine Zeichenkette.

- Den zu konvertierenden Wert geben Sie am Eingangsparameter IN an. Das Format des numerischen Werts bestimmen Sie durch die Auswahl eines Datentyps.
- Das Ergebnis der Konvertierung fragen Sie am Ausgangsparameter OUT ab.

Erlaubte Zeichen für die Konvertierung sind die Ziffern "0" bis "9", der Dezimalpunkt, das Dezimalkomma, Notationen "E" und "e" sowie das Plus- und Minus-Zeichen. Durch ungültige Zeichen wird die Konvertierung unterbrochen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VAL_STRG":

Parameter	Deklaration	Datentyp		Speicherbereich	Beschreibung
		S7-1200	S7-1500		
IN	Input	USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL	USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL	E, A, M, D, L, P oder Konstante	Zu konvertierender Wert
SIZE	Input	USINT	USINT	E, A, M, D, L, P oder Konstante	Anzahl der Zeichenplätze
PREC	Input	USINT	USINT	E, A, M, D, L, P oder Konstante	Anzahl der Nachkommastellen
FORMAT	Input	WORD	WORD	E, A, M, D, L, P oder Konstante	Ausgabeformat der Zeichen
P	InOut	UINT	UINT	E, A, M, D, L, P oder Konstante	Zeichen, ab dem das Ergebnis geschrieben wird.
OUT	Output	STRING, WSTRING	STRING, WSTRING	D, L	Ergebnis der Konvertierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter P

Durch den Parameter P geben Sie an, ab welchem Zeichen der Zeichenkette das Ergebnis geschrieben wird. Wenn beispielsweise am Parameter P der Wert "2" angegeben ist, wird der konvertierte Wert ab dem zweiten Zeichen der Zeichenkette gespeichert.

Parameter SIZE und P

Durch den Parameter SIZE legen Sie fest, wie viele Zeichen der Zeichenkette beschrieben werden. Gezählt wird dabei ab dem Zeichen, das am Parameter P angegeben ist. Wenn der Ausgabewert kürzer als die angegebene Länge ist, wird das Ergebnis rechtsbündig in die Zeichenkette geschrieben. Die leeren Zeichenstellen werden dabei mit Leerzeichen gefüllt.

Parameter FORMAT

Durch den Parameter FORMAT geben Sie vor, wie der numerische Wert bei der Konvertierung interpretiert und in die Zeichenkette geschrieben wird. Am Parameter FORMAT können Sie nur Variablen vom Datentyp USINT angeben.

Die folgende Tabelle zeigt die möglichen Werte des Parameters FORMAT und deren Bedeutung:

Wert (W#16#....)	Notation	Vorzeichen	Dezimaldarstellung
0000	Dezimalbruch	"-"	"."
0001			","
0002	Exponentiell		"."
0003			","
0004	Dezimalbruch	"+" und "-"	"."
0005			","
0006	Exponentiell		"."
0007			","
0008 bis FFFF	Ungültige Werte		

Hinweis

Relevanz der Werte 2, 3, 6 und 7 des Parameters FORMAT

Die Werte 2, 3, 6 und 7 des Parameters FORMAT sind nur dann relevant, wenn der Parameter IN vom Datentyp REAL oder LREAL ist.

Parameter PREC

Durch den Parameter PREC definieren Sie die Anzahl der Nachkommastellen bei der Konvertierung von Gleitpunktzahlen. Maximal wird eine Genauigkeit von 7 Ziffern für numerische Werte des Datentyps REAL unterstützt. Wenn der zu konvertierende Wert eine Ganzzahl ist, legen Sie mit dem Parameter PREC die Stelle fest, an der ein Dezimalpunkt gesetzt wird.

Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung von numerischen Werten in eine Zeichenkette, wenn der Parameter OUT vor dem Aufruf von VAL_STRG einen Leerstring enthält.

IN (Wert)	IN (Datentyp)	P	SIZE	FORMAT (W#16#....)	PREC	OUT (STRING)	ENO-Status
123	UINT	16	10	0000	0	`xxxxxxx123 C`	1
0	UINT	16	10	0000	2	`xxxxxx0.00 C`	1
12345678	UDINT	16	10	0000	3	`x12345.678 C`	1
12345678	UDINT	16	10	0001	3	`x12345.678 C`	1
123	INT	16	10	0004	0	`xxxxxx+123 C`	1
-123	INT	16	10	0004	0	`xxxxxx-123 C`	1
-0.00123	REAL	16	10	0004	4	`xxx-0.0012 C`	1
-0.00123	REAL	16	10	0006	4	`-1.2300E-3 C`	1
-Inf [1]	REAL	16	10	-/-	4	`xxxxxx-INF C`	0
+Inf [2]	REAL	16	10	-/-	4	`xxxxxx+INF C`	0
NaN [3]	REAL	16	10	-/-	4	`xxxxxxxNaN C`	0
12345678	UDINT	16	6	-/-	3	`xxxxxxxxxx C`	0

"x" repräsentiert Leerzeichen
[1]-Inf: Gleitpunktzahl, die einen negativen unendlichen Wert repräsentiert.
[2]+Inf: Gleitpunktzahl, die einen positiven unendlichen Wert repräsentiert.
[3]NaN: Wert, der als Ergebnis ungültiger Rechenoperationen zurückgegeben wird.

Beispiel

Im Folgenden finden Sie drei Beispiele für die Konvertierung von numerischen Werten in eine Zeichenkette, wenn der Parameter OUT vor dem Aufruf von VAL_STRG einen nicht leeren String enthält.

In allen Fällen haben die Parameter IN, SIZE, PREC, FORMAT und P dieselben Werte: IN=123, SIZE=16, PREC=0, FORMAT=4, P=10. Lediglich die Vorbelegung von OUT ist unterschiedlich.

- OUT enthält eine Zeichenkette, die kürzer als P ist.
  Die CPU befüllt OUT hinter dem bestehenden String mit Leerzeichen, und zwar bis zur (P-1)-ten Position. Ab Position P wird OUT unter Berücksichtigung der führenden Leerzeichen beschrieben.

OUT vor dem Aufruf von VAL_STRG:	ABCDEFxxxxxxxxxx xxxxxxxxxxxxxxxx
OUT nach dem Aufruf von VAL_STRG:	ABCDEFxxxxxxxxxx xxxxxx123xxxxxxx
"x" repräsentiert ein Leerzeichen	

- OUT enthält eine Zeichenkette, die länger als P+SIZE ist
  Die CPU befüllt OUT ab Position P unter Berücksichtigung der führenden Leerzeichen mit SIZE Zeichen.

OUT vor dem Aufruf von VAL_STRG:	ABCDEFGHIJKLMNOP QRSTUVWXYZABCDEF
OUT nach dem Aufruf von VAL_STRG:	ABCDEFGHIxxxxxxx xxxxxx123ZABCDEF
"x" repräsentiert ein Leerzeichen	

- OUT enthält eine Zeichenkette, die länger als P, aber kürzer als P+SIZE ist
  Die CPU befüllt OUT ab Position P unter Berücksichtigung der führenden Leerzeichen mit SIZE Zeichen.

OUT vor dem Aufruf von VAL_STRG:	ABCDEFGHIJKLMNOP xxxxxxxxxxxxxxxx
OUT nach dem Aufruf von VAL_STRG:	ABCDEFGHIxxxxxxx xxxxxx123xxxxxxx
"x" repräsentiert ein Leerzeichen	

Beispiel

In dem folgenden Beispiel konvertieren Sie eine Gleitpunktzahl mit dem Datentyp REAL in eine Zeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein sechs Variablen an.

My_gDB_VAL_STRG

	Name	Data type	Start value
1	Static		
2	inputVAL	Real	-567.12
3	pointer2STRG	UInt	16
4	sizeSTRG	USInt	10
5	precVAL	USInt	3
6	outputSTRING	String	''
7	resultV2Sformat	Word	16#0004

Die Parameter der Anweisung verschalten Sie wie folgt. Sie wählen die Datentypen aus:

- Die linke Auswahlmöglichkeit steht für den zu konvertierenden Wert.
- Die rechte Auswahlmöglichkeit steht für die zu erzeugende Zeichenkette.

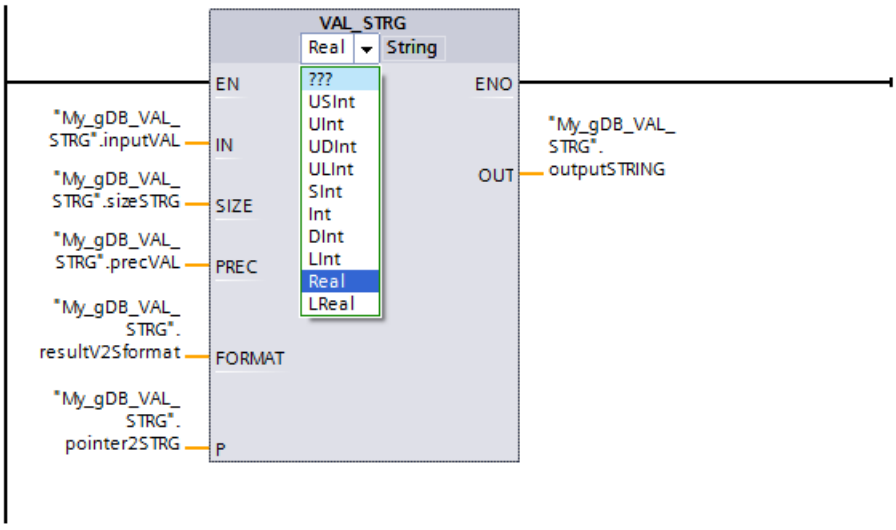

Entsprechend dem Wert "16" des Parameter P ("pointer2STRG") wird die Zeichenkette ab dem 16. Zeichen geschrieben. Ab diesem Punkt wird die Zeichenkette 10 Zeichen lang, gemäß dem Wert "10" des Parameters SIZE ("sizeSTRG"). Durch den Wert "0004" des Parameter FORMAT ("resultV2Sformat") wird der Punkt des zu konvertierenden Werts ("inputVAL") als Dezimaltrennzeichen interpretiert. Entsprechend dem Wert "3" des Parameters PREC ("precVAL") werden drei Nachkommastellen in die Zeichenkette geschrieben. Das Vorzeichen des zu konvertierenden Werts wird als Zeichen in die Zeichenkette übernommen und vor die Ziffern gesetzt. Vor dem Vorzeichen werden die restlichen der 10 Zeichen der Zeichenkette als Leerzeichen geschrieben. Die Zeichenkette wird am Ausgangsparameter OUT ("outputSTRING") ausgegeben.

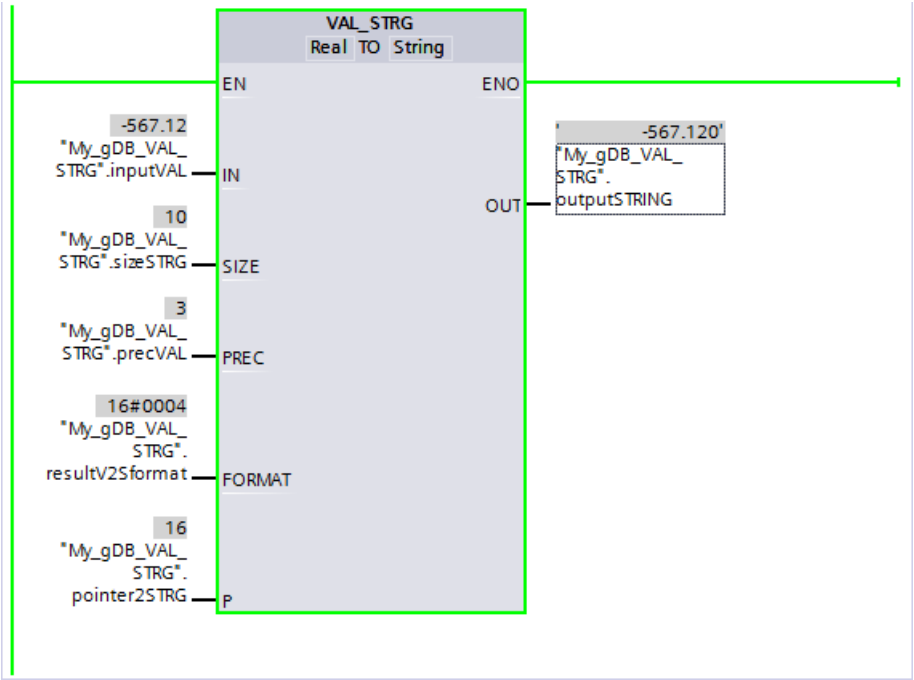

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Strg_TO_Chars: Zeichenkette in Array of CHAR konvertieren

Beschreibung

Mit der Anweisung "Strg_TO_Chars" kopieren Sie eine Zeichenkette vom Datentyp STRING in ein Array of CHAR oder Array of BYTE bzw. eine Zeichenkette vom Datentyp WSTRING in ein Array of WCHAR oder Array of WORD. Für den Kopiervorgang sind ausschließlich ASCII-Zeichen gültig.

- Die Zeichenkette geben Sie am Eingangsparameter STRG an.
- Die Zeichen werden am Parameter CHARS in einen Datentyp Array of CHAR / BYTE / WCHAR / WORD geschrieben.
  - Die Anzahl der Zeichen im Zielfeld muss mindestens so viele Zeichen umfassen, wie aus der Quellzeichenkette kopiert werden.
  - Wenn das Zielfeld weniger Zeichen als die Quellzeichenkette enthält, werden die Zeichen bis zur maximalen Länge von des Zielfelds geschrieben.
  - Wenn die Zeichenkette ein Zeichen "$00" bzw. W#16#0000 enthält, hat das auf den Kopiervorgang keine Auswirkung (siehe Grafik).
  - Die Anzahl der kopierten Zeichen wird am Parameter CNT ausgegeben.
- Über den Parameter PCHARS geben Sie an, ab welcher Position im Zielfeld geschrieben wird.
  - Beispiel: Soll der Schreibvorgang erst ab der dritten Position beginnen, verwenden Sie am Parameter PCHARS den Wert "2":

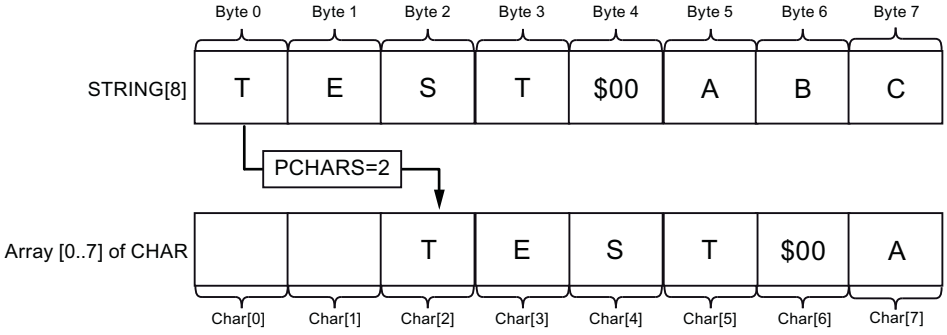

  - Der Standard für PCHARS ist "0". Mit PCHARS = 0 wird die untere Indexgrenze des Arrays verwendet (z. B. CHAR[0] bei einem Array [0..5] of CHAR). Dies gilt auch, wenn die Untergrenze des Array negativ ist (z. B. CHAR[-5] bei einem Array [-5..5] of CHAR).

Hinweis

Verwendung der Anweisung mit S7-1200 V2.0

Die S7-1200 bis zur Version 2.0 unterstützt nur Array [0 .. n] of CHAR / BYTE. Negative Indexgrenzen (z. B. Array [-3..2] of CHAR) sind nicht zulässig. Diese Einschränkung wird durch die Software nicht geprüft.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Strg_TO_Chars":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STRG	Input	STRING, WSTRING	D, L oder Konstante	Quelle des Kopiervorgangs
PCHARS	Input	DINT	E, A, M, D, L, P oder Konstante	Position in der Struktur Array of (W)CHAR / BYTE / WORD, ab der die Zeichen der Zeichenkette geschrieben werden.
CHARS	InOut	VARIANT	D, L	Ziel des Kopiervorgangs Kopieren Sie die Zeichen in eine Struktur vom Datentyp Array of (W)CHAR / BYTE / WORD.
CNT	Output	UINT	E, A, M, D, L, P	Anzahl der kopierten Zeichen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel kopieren Sie Zeichen einer Zeichenkette mit dem Datentyp STRING in eine Struktur des Datentyps Array of CHAR.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_Strg_TO_Chars

	Name	Data type	Start value
1	Static		
2	inputSTRG	String[10]	'cf7a7a#'
3	pointerCHARS	DInt	2
4	myarrayCHARS	Array[0..9] of Char	
5	myarrayCHARS[0]	Char	' '
6	myarrayCHARS[1]	Char	' '
7	myarrayCHARS[2]	Char	' '
8	myarrayCHARS[3]	Char	' '
9	myarrayCHARS[4]	Char	' '
10	myarrayCHARS[5]	Char	' '
11	myarrayCHARS[6]	Char	' '
12	myarrayCHARS[7]	Char	' '
13	myarrayCHARS[8]	Char	' '
14	myarrayCHARS[9]	Char	' '
15	countCHARS	UInt	0

Die Parameter der Anweisung verschalten Sie wie folgt und wählen den Datentyp der Zeichenkette aus.

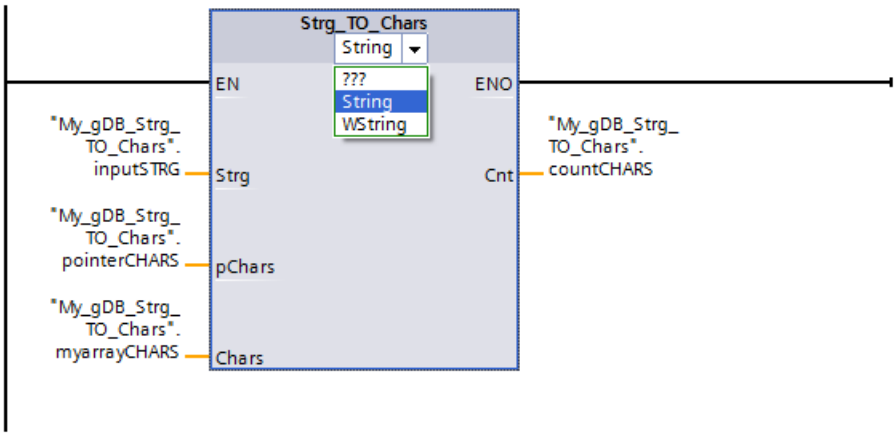

Durch den Datentyp Array of CHAR wird eine Struktur aus einzelnen Zeichen bestehend erstellt. Die Struktur CHARS ("myarrayCHARS") wird zehn Zeichen lang (Array ... [0..9]). Entsprechend dem Wert "2" des Parameters PCHARS ("pointerCHARS") wird ab dem dritten Zeichen in der Struktur geschrieben ("0" und "1" werden leer erzeugt, "2" enthält das erste Zeichen der Zeichenkette ("inputSTRG")). Nachdem die Zeichen der Zeichenkette ("inputSTRG") in die Struktur ("myarrayCHARS") geschrieben wurden, wird das letzte, zu erstellende Zeichen der Struktur leer geschrieben. Die Anzahl der kopierten Zeichen von der Zeichenkette wird am Ausgangsparameter CNT ("countCHARS") ausgegeben.

My_gDB_Strg_TO_Chars

	Name	Data type	Start value	Monitor value
1	Static			
2	inputSTRG	String[10]	'cf7a7a#'	'cf7a7a#'
3	pointerCHARS	DInt	2	2
4	myarrayCHARS	Array[0..9] of Char		
5	myarrayCHARS[0]	Char	' '	' '
6	myarrayCHARS[1]	Char	' '	' '
7	myarrayCHARS[2]	Char	' '	'c'
8	myarrayCHARS[3]	Char	' '	'f'
9	myarrayCHARS[4]	Char	' '	'7'
10	myarrayCHARS[5]	Char	' '	'a'
11	myarrayCHARS[6]	Char	' '	'7'
12	myarrayCHARS[7]	Char	' '	'a'
13	myarrayCHARS[8]	Char	' '	'#'
14	myarrayCHARS[9]	Char	' '	' '
15	countCHARS	UInt	0	7

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Chars_TO_Strg: Array of CHAR in Zeichenkette konvertieren

Beschreibung

Mit der Anweisung "Chars_TO_Strg" kopieren Sie Zeichen aus einem Array of CHAR oder Array of BYTE in eine Zeichenkette vom Datentyp STRING bzw. aus einem ARRAY of WCHAR oder Array of WORD in eine Zeichenkette vom Datentyp WSTRING. Für den Kopiervorgang sind ausschließlich ASCII-Zeichen gültig.

- Die Zeichen des Array of (W)CHAR / BYTE / WORD, die in eine Zeichenkette kopiert werden, geben Sie am Eingangsparameter CHARS an.
- Die Zeichen werden am Parameter STRG in einen Datentyp (W)STRING geschrieben.
  - Die Anzahl der Zeichen in der Zeichenkette umfasst mindestens so viele Zeichen, wie aus dem Quellfeld kopiert werden.
  - Wenn die Zeichenkette kürzer als die Anzahl der Zeichen des Quellfeldes ist, werden die Zeichen bis zur maximalen Länge der Zeichenkette geschrieben.
  - Wenn das Array of CHAR / BYTE ein Zeichen "$00" bzw. das Array of WCHAR / WORD ein Zeichen W#16#0000 enthält, erfolgt der Kopiervorgang nur bis zur entsprechenden Stelle (siehe Grafik).
- Geben Sie über den Parameter PCHARS an, ab welcher Position des Quellfelds die Zeichen kopiert werden. PCHARS = 0 ist der Standardwert und gibt immer die untere Indexgrenze des Arrays an, auch wenn diese negativ ist.
  - Beispiel: Soll der Kopiervorgang erst mit dem dritten Zeichen des Quellfelds beginnen, verwenden Sie am Parameter PCHARS den Wert "2":

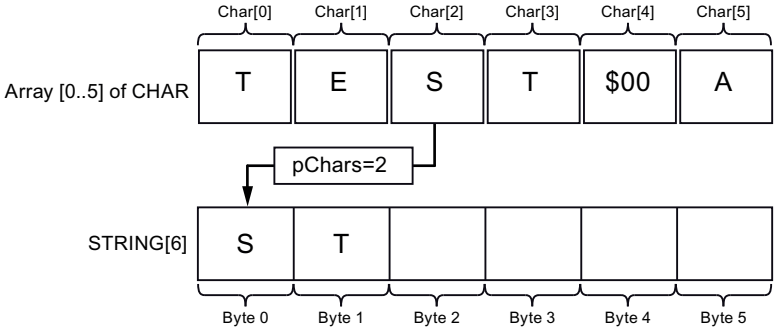

  - Wenn am Parameter PCHARS ein Index angegeben wird, welcher in der Kopierquelle nicht enthalten ist (z. B. "7" bei Array [0..5] of CHAR), wird die Anweisung nicht ausgeführt.

Hinweis

Verwendung der Anweisung mit S7-1200 V2.0

Die S7-1200 bis zur Version 2.0 unterstützt nur Array [0 .. n] of CHAR / BYTE. Negative Indexgrenzen (z. B. Array [-3..2] of CHAR) sind nicht zulässig. Diese Einschränkung wird durch die Software nicht geprüft.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Chars_TO_Strg":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CHARS	Input	VARIANT	D, L	Quelle des Kopiervorgangs Array of (W)CHAR / BYTE / WORD, aus der die Zeichen kopiert werden.
PCHARS	Input	DINT	E, A, M, D, L, P oder Konstante	Position im Array of (W)CHAR / BYTE / WORD, ab der die Zeichen kopiert werden.
CNT	Input	UINT	E, A, M, D, L, P oder Konstante	Anzahl der zu kopierenden Zeichen. Bei "0" werden alle Zeichen kopiert.
STRG	Output	STRING, WSTRING	D, L	Ziel des Kopiervorgangs Zeichenkette mit dem Datentyp (W)STRING. Beachten Sie die maximale Länge der Datentypen: • STRING: 254 Zeichen • WSTRING: 254 Zeichen (Standard) / 16382 Zeichen (Maximum) Achten Sie bei der Verwendung von WSTRING darauf, dass Sie eine Länge > 254 Zeichen explizit mit eckigen Klammern definieren müssen (z. B. WSTRING[16382]).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel kopieren Sie Zeichen einer Struktur des Datentyps Array of CHAR in eine Zeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_Chars_TO_Strg

	Name	Data type	Start value
1	Static		
2	inputArrayCHARS	Array[0..9] of Char	
3	inputArrayCHARS[0]	Char	'M'
4	inputArrayCHARS[1]	Char	'y'
5	inputArrayCHARS[2]	Char	'S'
6	inputArrayCHARS[3]	Char	'7'
7	inputArrayCHARS[4]	Char	'P'
8	inputArrayCHARS[5]	Char	'L'
9	inputArrayCHARS[6]	Char	'C'
10	inputArrayCHARS[7]	Char	' '
11	inputArrayCHARS[8]	Char	' '
12	inputArrayCHARS[9]	Char	' '
13	pointerCHARS	DInt	2
14	outputSTRG	String	''
15	countCHARS	UInt	0

Die Parameter der Anweisung verschalten Sie wie folgt und wählen den Datentyp der Zeichenkette aus.

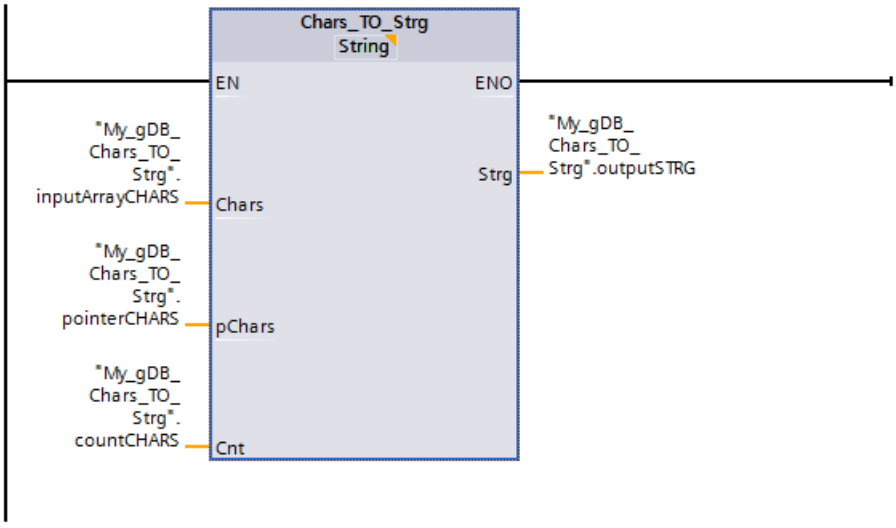

Die Struktur CHARS ("inputArrayCHARS") ist zehn Zeichen lang (Array ... [0..9]). Entsprechend dem Wert "2" des Parameter PCHARS ("pointerCHARS") wird erst ab der dritten Stelle der Struktur in die Zeichenkette ("outputSTRG") kopiert. Ab der Stelle "2" werden alle Zeichen der Struktur ("inputArrayCHARS") in die Zeichenkette ("outputSTRG") kopiert, da der Parameter CNT ("countCHARS") den Wert "0" besitzt.

My_gDB_Chars_TO_Strg

	Name	Data type	Start value	Monitor value
1	Static			
2	inputArrayCHARS	Array[0..9] of Char		
3	inputArrayCHARS[0]	Char	'M'	'M'
4	inputArrayCHARS[1]	Char	'y'	'y'
5	inputArrayCHARS[2]	Char	'S'	'S'
6	inputArrayCHARS[3]	Char	'7'	'7'
7	inputArrayCHARS[4]	Char	'P'	'P'
8	inputArrayCHARS[5]	Char	'L'	'L'
9	inputArrayCHARS[6]	Char	'C'	'C'
10	inputArrayCHARS[7]	Char	' '	' '
11	inputArrayCHARS[8]	Char	' '	' '
12	inputArrayCHARS[9]	Char	' '	' '
13	pointerCHARS	DInt	2	2
14	outputSTRG	String	''	'S7PLC '
15	countCHARS	UInt	0	0

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

MAX_LEN: Maximale Länge einer Zeichenkette ermitteln

Beschreibung

Eine Variable vom Datentyp (W)STRING enthält zwei Längen: die maximale Länge und die aktuelle Länge (das ist die Anzahl der momentan gültigen Zeichen).

- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort STRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Bytes ist um 2 größer als die maximale Länge.
- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort WSTRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Worte ist um 2 größer als die maximale Länge.
- Die aktuelle Länge repräsentiert die Anzahl der tatsächlich belegten Zeichenplätze. Die aktuelle Länge ist kleiner oder gleich der maximalen Länge.

Mit der Anweisung "MAX_LEN" ermitteln Sie die maximale Länge der am Eingangsparameter IN angegebenen Zeichenkette. Diese wird als numerischer Wert am Ausgangsparameter OUT ausgegeben.

Hinweis

Überprüfung der Zeichenkette

Die Anweisung "MAX_LEN" überprüft die Zeichenkette am Parameter "IN" nicht.

Wenn Fehler während der Bearbeitung der Anweisung auftreten, wird am Parameter OUT der Wert "0" ausgegeben.

Hinweis

Auslesen der aktuellen Länge

Mit der Anweisung "LEN (Seite 2256)" lesen Sie die aktuelle Länge einer Zeichenkette aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MAX_LEN":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	• STRING • WSTRING	D, L oder Konstante	Zeichenkette
OUT	Return	• INT • DINT	E, A, M, D, L, P	Maximale Anzahl der Zeichen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel ermitteln Sie die maximale Länge einer Zeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

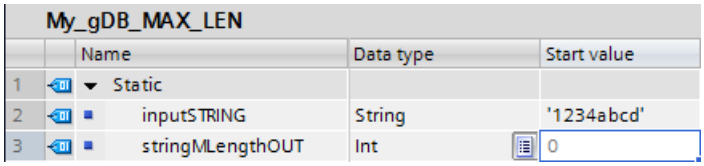

My_gDB_MAX_LEN

		Name	Data type	Start value
1		Static		
2		inputSTRING	String	'1234abcd'
3		stringMLengthOUT	Int	0

Die Parameter der Anweisung verschalten Sie wie folgt.

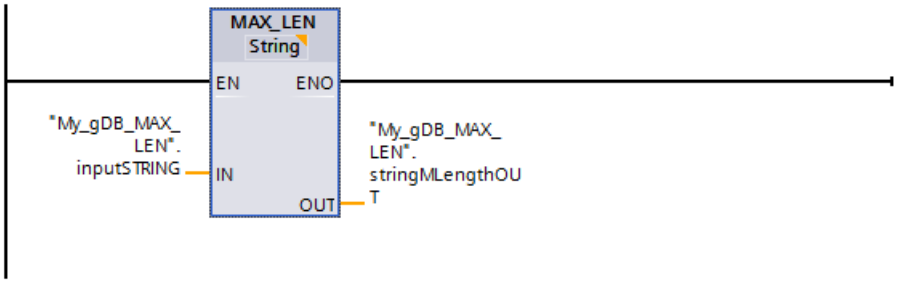

Die maximale Länge der angegebenen Zeichenkette ("inputSTRING") wird ermittelt und am Ausgangsparameter OUT ("stringMLengthOUT") als numerischer Wert ausgegeben.

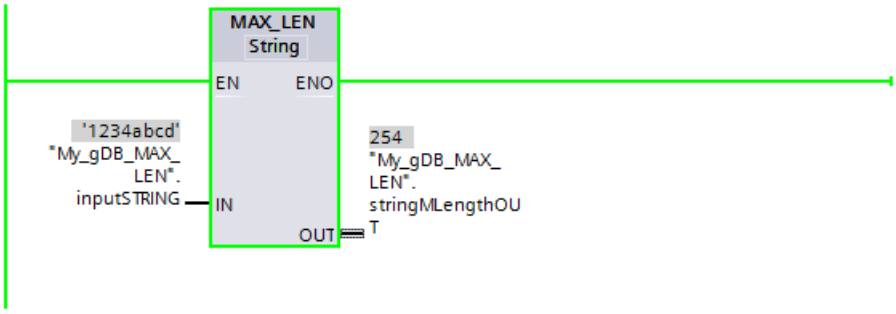

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

JOIN: Mehrere Zeichenketten verbinden

Beschreibung

Die Anweisung "JOIN" verbindet mehrere Zeichenketten zu einem Array.

Für die Umwandlung mehrerer Zeichenketten in eine Zeichenkette bietet die Anweisung folgende Funktionen:

- Formatauswahl
  Über das erste Bit am Parameter Mode geben Sie an, ob die Aneinanderreihung der Quellzeichenketten zu einem Array im Format CSV oder FSR erfolgen soll.
  Im folgenden Beispiel sind zwei Quellzeichenketten durch die beiden Spalten der Tabelle gegeben. Die maximale Zeichenanzahl für die erste Quellzeichenkette betrage 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen.

1963	1974
Miller	Jackson
John	Peter
Roadname	VeryLongRoadna

  – Bei CSV (Comma Separated Values) werden die Quellzeichenketten hintereinander in das Ziel-Array geschrieben und jeweils durch ein Trennzeichen getrennt. (Beispiele siehe unten).
  – Bei FSR (Fixed Size Records) ist im Ziel-Array für jede Quellzeichenkette eine bestimmte Anzahl von Zeichen definiert. Wenn die Zeichen einer Quellzeichenkette den für sie reservierten Platz im Ziel-Array nicht benötigen, werden die zugehörigen Feldelemente mit Trennzeichen aufgefüllt. Wenn die Anzahl der Zeichen einer Quellzeichenkette jedoch größer ist als der für sie reservierte Platz, werden die zugehörigen Feldelemente von vorne her aufgefüllt, und die überzähligen Zeichen der Quellzeichenkette werden abgeschnitten (Beispiel siehe unten).

- Auswahl von Trennzeichen für die Quellzeichenketten
  Über den Parameter RecSeparator wählen Sie aus, welches Trennzeichen für die einzelnen Strings verwendet wird. Welches Zeichen Sie auswählen, sollten Sie vom Inhalt der Eingangszeichenketten am Parameter SrcStruct abhängig machen. Wenn die Eingangszeichenketten z. B. innerhalb eines Strings ein Komma enthalten, verwenden Sie das Komma nicht als Trennzeichen. Der Datentyp, den Sie für das Trennzeichen verwenden, muss dem des Ziel-Arrays am Parameter DstArray entsprechen. Dadurch können die Trennzeichen mit in das Array geschrieben werden.

- Auswahl von Trennzeichen für das Ende der aller Quellzeichenketten
  Über das dritte Bit am Parameter Mode wählen Sie aus, ob im Ziel-Array (Parameter DstArray) ein zusätzliches Zeichen als Trennzeichen ans Ende der kopierten Zeichen geschrieben wird. Am Parameter EndSeparator geben Sie das Zeichen ein, das als Trennzeichen verwendet wird. Achten Sie darauf, ein anderes Zeichen zu verwenden als am Parameter RecSeparator (Trennzeichen für einzelne Zeichenketten). Bei einer Zurückwandlung über die Anweisung "SPLIT" erhalten Sie sonst unerwartete Ergebnisse, wenn die beiden Trennzeichen nicht unterscheidbar sind.

- Auswahl der Quellzeichenketten
  Die Quellzeichenketten geben Sie am Parameter SrcStruct an. Verwenden können Sie als Datentyp Array of STRING oder Array of WSTRING oder Strukturen, die ausschließlich den Datentyp STRING oder WSTRING enthalten. Dies gilt auch für Anwenderdatentypen oder verschachtelte Strukturen. Solange diese ausschließlich den Datentyp STRING respektive WSTRING enthalten, können Sie diese verwenden.

- **Angabe der Anzahl der verbundenen Zeichenketten**
  Wenn Sie am Parameter SrcStruct (Quellzeichenketten) ein Array of STRING oder Array of WSTRING verwenden (keine verschachtelten Strukturen), können Sie über den Parameter Count die Anzahl der Quellzeichenketten angeben, die zu einer Zeichenkette verbunden werden. Wenn Sie am Parameter SrcStruct einen anderen Datentyp als Array of (W)STRING verwenden, wird der Parameter Count ignoriert. So lässt sich nur ein größerer Teil eines Arrays verbinden.
- **Auswahl des Zielbereichs zum Schreiben des Arrays**
  An dem Parameter DstArray verwenden Sie den Datentyp Array of (W)CHAR. Die Verwendung der Datentypen STRING oder WSTRING ist in diesem Fall nicht möglich, da bei STRING die Länge auf 254 Zeichen oder 256 Bytes begrenzt ist.
- **Index der Position in dem Array (Zielparameter DestArray)**
  Die Umwandlung beginnt an dieser Position. Die Anweisung gibt über den Parameter Position aus, an welcher Position die Umwandlung abgeschlossen wurde. Dies ermöglicht Folgeaufrufe der Anweisung zur Befüllung des Arrays.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "JOIN":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Mode	Input	DWORD	E, A, M, D, L oder Konstante	Spezifiziert, wie die Zusammenführung zu einer Zeichenkette durchgeführt wird (siehe "Parameter Mode").
RecSeparator	Input	VARIANT	E, A, M, D, L	Trennzeichen für die Quellzeichenketten • Bei CSV: Zeichen, das als Trennzeichen für die einzelnen Strings verwendet wird. • Bei FSR: Zeichen, das als Füllzeichen für die einzelnen Strings verwendet wird.
EndSeparator	Input	VARIANT	E, A, M, D, L	Trennzeichen für das Ende der Umwandlung Trennzeichen, das am Ende der Zeichen geschrieben wird, falls für Parameter Mode das Bit 3 = 1 gesetzt wurde.
SrcStruct	Input	VARIANT	E, A, M, D, L	Zeiger auf die Quellzeichenketten.
Count	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Zeichenketten, die zusammengeführt werden. Der Parameter Count kann nur verwendet werden, wenn am Parameter SrcStruct ein Array of (W)STRING verwendet wurde.
DestArray	InOut	VARIANT	E, A, M, D, L	Bereich, in den die Zeichen nach der Umwandlung geschrieben werden. Verwenden Sie Parameter DestArray den Datentyp Array [0 .. x] of CHAR/WCHAR. Die Länge (x) des Arrays wählen Sie nach der Länge der Quellzeichenketten am Parameter SrcStruct aus.
Position	InOut	UDINT	E, A, M, D, L	Index der Position in der Gesamtzeichenkette
Ret_Val	Return	INT	E, A, M, D, L	Status der Anweisung (siehe Tabelle "Parameter RET_VAL").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter Mode

Bit	Bitwert "0"	Bitwert "1"	Beschreibung
0	CSV-Format (Comma Separated Values)	FSR-Format (Fixed Size Record)	Auswahl des Formats: • Bei CSV werden die Quellzeichenketten durch ein Trennzeichen im Ziel-Array getrennt. • Bei FSR werden die Quellzeichenketten mit denen am Parameter RecSeparator definierten Füllzeichen in das Ziel-Array geschrieben.
1	-	-	Für die Anweisung "JOIN" nicht relevant.
2	-	-	Reserviert (Bitwert irrelevant)
3	Kein zusätzliches Trennzeichen schreiben.	Zeichen am Ende der gelesenen Zeichen schreiben, das über den Parameter EndSeparator definiert wurde.	Auswahl, ob in dem Array (Parameter DestArray) ein zusätzliches Zeichen als Separator ans Ende der Zeichen geschrieben wird.
4	-	-	Für die Anweisung "JOIN" nicht relevant.

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler.
8190	Auswahl am Parameter Mode wird nicht unterstützt.
8x20	Quellzeichenketten ungültig.
8x53	VARIANT verweist auf zu kurze Datenstruktur.
8x54	Ungültiger Datentyp
8082	Der Wert am Parameter Count ist größer als die Anzahl der Quellzeichenketten in SrcStruct.
8xB4	Unterschiedliche Datentypen an den Parametern SrcStruct (Quelle) und DestArray (Ziel) oder den Trennzeichen (Parameter RecSeparator und EndSeparator).
80B5	Pufferüberlauf bei der Anweisung. Am Parameter DestArray werden die Zeichen unvollständig ausgegeben, oder der Wert am Parameter Position liegt außerhalb von DestArray.
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

* Beachten Sie zu den Fehlercodes Folgendes:

- Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".
- Das "x" an zweiter Stelle der aufgelisteten Fehlercodes steht für den Parameter, der den Fehler verursacht hat. Beispiel: Fehlercode 8352 hex = Fehler am 3. Parameter (EndSeparator) aufgetreten, siehe Parametertabelle.
- Wenn der Fehler nicht eindeutig einem Parameter zugeordnet wird, so wird eine "0" ausgegeben. Beispiel: Für das Trennzeichen (Parameter RecSeparator) wird als Datentyp CHAR verwendet. Für das Array am Parameter DestArray wird als Datentyp WCHAR verwendet. In diesem Fall wird als Fehlercode 80B4 ausgegeben.

Zwei Beispiele zur Anweisung JOIN, wenn das Ziel-ARRAY das Format CSV haben soll

- Erstes Beispiel
  Folgende Quellzeichenketten sind gegeben:
  - `1963`
  - `Miller`
  - `John`
  - `Roadname`

  Wenn SIe als Trennzeichen "," wählen, dann liefert der Aufruf von JOIN das folgende Ziel-Array:
  `1963,Miller,John,Roadname`
- Zweites Beispiel
  Folgende Quellzeichenketten sind gegeben:
  - `1974`
  - `Jackson`
  - `Peter`
  - `VeryLongRoadname`

  Wenn SIe als Trennzeichen "," wählen, dann liefert der Aufruf von JOIN das folgende Ziel-Array:
  `1974,Jackson,Peter,VeryLongRoadname`

Zwei Beispiele zur Anweisung JOIN, wenn das Ziel-ARRAY das Format FSR haben soll

- Erstes Beispiel
  Folgende Quellzeichenketten sind gegeben:
  - `1963`
  - `Miller`
  - `John`
  - `Roadname`

  Die Anzahl der reservierten Zeichen im Ziel-Array beträgt für die erste Quellzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen. Wenn SIe als Füllzeichen "," wählen, dann liefert der Aufruf von JOIN das folgende Ziel-Array:
  `1963Miller,,,,,,,John,,,,,,Roadname,,,,,,`
- Zweites Beispiel
  Folgende Quellzeichenketten sind gegeben:
  - `1974`
  - `Jackson`
  - `Peter`
  - `VeryLongRoadname`

  Die Anzahl der reservierten Zeichen im Ziel-Array beträgt für die erste Quellzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen. Wenn SIe als Füllzeichen "," wählen, dann liefert der Aufruf von JOIN das folgende Ziel-Array:
  `1974Jackson,,,,,,Peter,,,,,VeryLongRoadna`

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SPLIT: Array von Zeichen in mehrere Zeichenketten aufteilen

Beschreibung

Die Anweisung "SPLIT" wandelt ein Array (Array of CHAR / WCHAR) in mehrere getrennte Zeichenketten um (Array of STRING / WSTRING oder Struktur).

Für die Umwandlung des Arrays in mehrere Zeichenketten geben Sie die folgenden Informationen an:

- **Auswahl des zu lesenden Arrays**
  Das zu lesende Array geben Sie am Parameter SrcArray an. Achten Sie bei den weiteren Parametern darauf, dass die verwendeten Datentypen der Ein- und Ausgangsparameter übereinstimmen. Wenn Sie zum Beispiel am Parameter SrcArray ein Array mit dem Datentyp CHAR verwenden, müssen Sie sowohl für die Trennzeichen (Rec-/EndSeparator) den Datentyp CHAR als auch die Struktur am Parameter DestStruct nur Zeichenketten vom Datentyp STRING enthalten.
- **Formatauswahl für das Quell-Array**
  Über das erste Bit am Parameter Mode geben Sie an, ob das zu lesende Array das Format CSV oder FSR hat.
  - Bei CSV (Comma Separated Values) werden zusammengehörende Zeichen des Quell-Arrays durch ein Trennzeichen von den nächsten zusammengehörenden Zeichen getrennt.
    Zwei Beispiele für Quell-Arrays:
    `1963,Miller,John,CitynameA,Roadname`
    `1974,Jackson,Peter,CitynameB,VeryLongRoadname`
  - Bei FSR (Fixed Size Records) ist für jede logische Information des Quell-Arrays eine bestimmte Anzahl von Zeichen definiert. Jede Information soll in ihren definierten Platz passen. Wenn eine Information den für sie definierten Platz nicht benötigt, wird mit Trennzeichen aufgefüllt.
    Zwei Beispiele für Quell-Arrays, wobei die Länge der Informationen für die erste Information (Geburtsjahr) 4 Zeichen, für die zweite (Name) 13 Zeichen, für die dritte (Vorname) 10 Zeichen, für die vierte (Wohnort) 9 Zeichen und für die fünfte (Straße) 16 Zeichen betrage:
    `1963Miller,,,,,,,John,,,,,,CitynameARoadname,,,,,,,,`
    `1974Jackson,,,,,,Peter,,,,,CitynameBVeryLongRoadname`
- **Verwendetes Trennzeichen für das zu lesende Array**
  - Wenn das zu lesende Array das Format CSV hat, geben Sie über den Parameter RecSeparator an, welches Trennzeichen verwendet wurde.
  - Wenn das zu lesende Array das Format FSR hat, geben Sie über den Parameter RecSeparator an, welches Füllzeichen verwendet wurde.
- **Verwendetes Trennzeichen für das Ende der Gesamtzeichenkette**
  Am Parameter EndSeparator geben Sie das Trennzeichen an, nach welchem das Array nicht weiter ausgelesen wird. Die Anweisung "SPLIT" bricht an dieser Stelle ab und gibt die gefundenen Zeichenketten aus. Beachten Sie, dass das Trennzeichen EndSeparator vorrangig vor dem Trennzeichen RecSeparator ausgewertet wird. Wenn das Trennzeichen am Parameter EndSeparator in einer zu lesenden Zeichenkette verwendet wurde (zwischen zwei Trennzeichen RecSeparator), werden alle Inhalte nach dem Trennzeichen EndSeparator nicht mehr ausgegeben.

- **Angabe der Position, ab welcher das Array gelesen wird**
  Die Umwandlung beginnt an dieser Position im Array und die Anweisung gibt über den Parameter Position aus, an welcher Position die Umwandlung abgeschlossen wurde. Dies ermöglicht Folgeaufrufe der Anweisung zur Befüllung verschiedener Zeichenketten am Parameter DestStruct.
- **Ausgabe der Anzahl der gelesenen Zeichenketten**
  Wenn Sie am Parameter DestStruct ein Array of STRING verwenden, können Sie über den Parameter Count die Anzahl der gelesenen Zeichenketten ausgeben lassen. Gezählt werden nur die Zeichenketten mit Inhalt. Wenn Sie am Parameter DestStruct einen anderen Datentyp als Array of STRING verwenden, wird am Parameter Count "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SPLIT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Mode	Input	DWord	E, A, M, D, L oder Konstante	Spezifiziert, wie die Aufteilung in mehrere Zeichenketten durchgeführt wird (siehe "Parameter Mode").
RecSeparator	Input	Variant	E, A, M, D, L	Trenn- bzw. Füllzeichen • Bei CSV: Zeichen, das in dem zu lesenden Array zur Identifikation der einzelnen Strings verwendet wurde. • Bei FSR: Zeichen, das als Füllzeichen in dem zu lesenden Array verwendet wurde.
EndSeparator	Input	Variant	E, A, M, D, L	Trennzeichen, über das in dem zu lesenden Array das Ende des Gesamtstrings definiert wurde.
SrcArray	Input	Variant	E, A, M, D, L	Zeiger auf das zu lesende Array (Array of CHAR/WCHAR)
DestStruct	InOut	Variant	E, A, M, D, L	Struktur, welche die umgewandelten Zeichenketten enthält (Array of STRING / WSTRING).
Position	InOut	UDInt	E, A, M, D, L	Position, ab der das Array am Parameter SrcArray gelesen wird.
Ret_Val	Return	Int	E, A, M, D, L	Ergebnis der Anweisungsausführung / Fehlercode (siehe Tabelle "Parameter Ret_Val")
Count	Output	UDInt	E, A, M, D, L	Anzahl der Zeichenketten, die gefunden wurden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter "Mode"

Bit	Bitwert "0"	Bitwert "1"	Beschreibung
0	CSV-Format (Comma Separated Values)	FSR-Format (Fixed Size Record)	Basismodus: Auswahl ob CSV oder FSR
1	• Bei CSV: Zusätzliche Zeichen führen zu einem Fehler. • Bei FSR: Zusätzliche Füllzeichen bleiben in String.	• Bei CSV: Zusätzliche Zeichen werden ignoriert. • Bei FSR: Zusätzliche Füllzeichen werden entfernt.	Über das Bit 1 wählen Sie aus, wie mit zusätzlichen Zeichen verfahren wird: • Bei CSV: – Wenn das Bit gesetzt ist, werden zusätzliche Zeichen, die nicht mehr in die Zielzeichenkette passen, ignoriert. Beispiel: Die Anweisung schreibt in eine Zeichenkette mit der Länge 16 (Datentyp STRING[16]). In der Quelle ist nach den ersten 16 Zeichen kein Trennzeichen enthalten. Wenn Bit 1 gesetzt ist, werden die überschüssigen Zeichen ignoriert und die Anweisung fährt fort, das Array auszulesen. – Wenn das Bit nicht gesetzt ist, bricht die Anweisung in so einem Fall ab und gibt eine Fehlermeldung am Parameter Ret_Val aus. • Bei FSR: – Wenn das Bit gesetzt ist, werden beim Übertragen des Quell-Arrays in die Zielzeichenketten diejenigen Füllzeichen, die rechts von Zeichen mit Informationsgehalt stehen, nicht in die Zielzeichenkette geschrieben (siehe Beispiel). – Wenn das Bit nicht gesetzt ist, werden beim Übertragen des Quell-Arras in die Zielzeichenketten diejenigen Füllzeichen, die rechts von Zeichen mit Informationsgehalt stehen, in die Zielzeichenkette geschrieben (siehe Beispiel).
2	-	-	Reserviert zur Verwendung in zukünftigen Versionen.
3	EndSeparator wird nicht gezählt.	EndSeparator wird gezählt.	Auswahl, ob EndSeparator beim Parameter Position mitgezählt wird.
4	Nicht beschriebene Zeichenketten (STRING) auf Länge belassen.	Nicht beschriebene Zeichenketten (STRING) auf Länge "0" setzten.	Auswahl, ob die nicht verwendeten Zeichenketten (STRING) am Parameter DestStruct auf die Länge "0" gesetzt werden.

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler.
0001	Zusätzliche Zeichen werden ignoriert.
0002	Nur bei CSV: • Im zu lesenden Array (festgelegt durch SrcArray) wurde kein Trennzeichen EndSeparator gefunden. Die Umwandlung wurde wegen eines anderen Grunds abgebrochen, z. B. weil DestStruct vollständig gefüllt ist. • RecSeparator und EndSeparator sind identisch.
8190	Auswahl am Parameter Mode wird nicht unterstützt.

Fehlercode* (W#16#...)	Erläuterung
8x20	Quellzeichenketten ungültig.
8x53	VARIANT verweist auf zu kurze Datenstruktur.
8x54	Ungültiger Datentyp
8882	Position nicht im zulässigen Bereich
8x84	Zusätzliche Zeichen gefunden.
8xB4	Unterschiedliche Datentypen an den Parametern SrcArray (Quelle) und DestStruct (Ziel) oder den Trennzeichen (Parameter RecSeparator und EndSeparator).
80B5	Das zu lesende Array SrcArray ist zu klein.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

* Beachten Sie zu den Fehlercodes Folgendes:

- Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".
- Das "x" an zweiter Stelle der aufgelisteten Fehlercodes steht für den Parameter, der den Fehler verursacht hat. Beispiel: Fehlercode 8352 hex = Fehler am 3. Parameter (EndSeparator) aufgetreten, siehe Parametertabelle.
- Wenn der Fehler nicht eindeutig einem Parameter zugeordnet wird, so wird eine "0" ausgegeben. Beispiel: Für das Trennzeichen (Parameter RecSeparator) wird als Datentyp CHAR verwendet. Für das Array am Parameter DestArray wird als Datentyp WCHAR verwendet. In diesem Fall wird der Fehlercode 80B4 ausgegeben.

Zwei Beispiele zur Anweisung SPLIT, wenn das zu lesende ARRAY das Format CSV hat

Im ersten Schritt wird das Quell-Array so in Zeichenketten aufgespalten, wie es die Trennzeichen (Parameter RecSeparator, z. B. "," vorgeben.

Im zweiten Schritt werden die nach der Aufspaltung entstandenen Zeichenketten in Zielzeichenketten gespeichert. Deren Längen geben Sie über den Parameter DestStruct vor.

Wenn in den durch die Aufspaltung entstandenen Zeichenketten mehr Zeichen enthalten sind, als in die Zielzeichenketten passen, dann hängt das weitere Verhalten von SPLIT vom Parameter Mode ab. Bei Mode=2#00 führt das zu einem Fehler: Der zugehörige Fehlercode wird ausgegeben, und es werden keine weiteren Zielzeichenketten befüllt. Bei Mode=2#10

werden die überzähligen Zeichen ignoriert, und die nächste Zielzeichenkette wird mit den Zeichen nach dem nächsten Trennzeichen befüllt.

- Erstes Beispiel
  In diesem Beispiel gibt es diese zwei Quell-Arrays:
  - `1963,Miller,John,Roadname`
  - `1974,Jackson,Peter,VeryLongRoadname`

  Die Länge der Zielzeichenketten beträgt für die erste Zielzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen.
  Der erste Aufruf von SPLIT liefert mit Mode=2#10 die folgenden Zielzeichenketten:
  - `1963`
  - `Miller`
  - `John`
  - `Roadname`

  Der zweite Aufruf von SPLIT liefert mit Mode=2#10 die folgenden Zielzeichenketten:
  - `1974`
  - `Jackson`
  - `Peter`
  - `VeryLongRoadna`
- Zweites Beispiel
  In diesem Beispiel liegen die logischen Informationen in den Quell-Arrays nicht in der erwarteten Reihenfolge vor. Es gibt diese zwei Quell-Arrays:
  - `1963,Miller,Roadname,John`
  - `1974,Jackson,VeryLongRoadname,Peter`

  Die Länge der Zielzeichenketten beträgt wie im ersten Beispiel für die erste Zielzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen.
  Der erste Aufruf von SPLIT liefert mit Mode=2#10 die folgenden Zielzeichenketten:
  - `1963`
  - `Miller`
  - `Roadname`
  - `John`

  Der zweite Aufruf von SPLIT liefert mit Mode=2#10 die folgenden Zielzeichenketten:
  - `1974`
  - `Jackson`
  - `VeryLongRo`
  - `Peter`

Zwei Beispiele zur Anweisung SPLIT, wenn das zu lesende ARRAY das Format FSR hat

Das Quell-Array wird so in Zeichenketten aufgespalten, wie Sie es im Parameter DestStruct für die Länge der Zielzeichenketten vorgegeben haben.

Hinsichtlich der Füllzeichen (Parameter RecSeparator, z. B. ",") hängt das Verhalten von SPLIT vom Parameter Mode ab: Bei Mode=2#01 werden die Füllzeichen in die Zielzeichenketten übernommen, bei Mode=2#11 werden diese nicht übernommen.

- Erstes Beispiel
  In diesem Beispiel gibt es diese zwei Quell-Arrays:
  - `1963Miller,,,,,,,John,,,,,,Roadname,,,,,,`
  - `1974Jackson,,,,,,Peter,,,,,VeryLongRoadname`

  Die Länge der Zielzeichenketten beträgt für die erste Zielzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen.
  Der erste Aufruf von SPLIT liefert mit Mode=2#01 die folgenden Zielzeichenketten:
  - `1963`
  - `Miller,,,,,,,`
  - `John,,,,,,`
  - `Roadname,,,,,,`

  Der zweite Aufruf von SPLIT liefert mit Mode=2#01 die folgenden Zielzeichenketten:
  - `1974`
  - `Jackson,,,,,,`
  - `Peter,,,,,`
  - `VeryLongRoadna`
- Zweites Beispiel
  In diesem Beispiel liegen die logischen Informationen in den Quell-Arrays nicht in der erwarteten Reihenfolge vor. Es gibt diese zwei Quell-Arrays:
  - `1963Miller,,,,,,,Roadname,,,,,,John,,,,,,`
  - `1974Jackson,,,,,,VeryLongRoadnamePeter,,,,,`

  Die Länge der Zielzeichenketten beträgt wie im ersten Beispiel für die erste Zielzeichenkette 4 Zeichen, für die zweite 13 Zeichen, für die dritte 10 Zeichen und für die vierte 14 Zeichen.
  Der erste Aufruf von SPLIT liefert mit Mode=2#11 die folgenden Zielzeichenketten:
  - `1963`
  - `Miller`
  - `Roadname`
  - `,,,,John`

  Erläuterung: Die Anweisung SPLIT legt quasi Schablonen über das Quell-Array, deren Länge durch die Länge der Zielzeichenketten vorgegeben wird. Beim Übertragen in die Zielzeichenketten werden nur jene Kommas als Füllzeichen bewertet, die rechts von Zeichen mit Informationsgehalt stehen. Bei Mode=2#11 werden die Füllzeichen nicht übernommen. Im ersten Aufruf sind die Kommas rechts von "John" Füllzeichen. Die Kommas links von "John" hingegen sind keine Füllzeichen, weil vor Erreichen des Endes der zugehörigen Zielzeichenkette noch die relevanten Zeichen "John" folgen. Die Kommas links von "John" erscheinen daher in der Zielzeichenkette.
  Der zweite Aufruf von SPLIT liefert mit Mode=2#11 die folgenden Zielzeichenketten:

- 1974
- Jackson
- VeryLongRo
- adnamePeter

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

JOIN: Mehrere Zeichenketten verbinden (Seite 2238)

ATH: ASCII-Zeichenkette in Hexadezimalzahl konvertieren

Beschreibung

Mit der Anweisung "ATH" konvertieren Sie die am Eingangsparameter IN angegebene ASCII-Zeichenkette in eine Hexadezimalzahl. Das Ergebnis der Konvertierung wird am Ausgangsparameter OUT ausgegeben.

- Über den Zeiger am Parameter IN (ASCII) können Sie auf die folgenden Datentypen verweisen: STRING, WSTRING, CHAR, BYTE, Array of CHAR, Array of BYTE, WCHAR, Array of WCHAR, Array of WORD.
- Über den Zeiger am Parameter OUT (hexadezimal) können Sie auf die folgenden Datentypen verweisen: Array of CHAR, Array of BYTE, Array of WORD, STRING, BYTE, CHAR, WORD, INT, DWORD, DINT, SINT, USINT, UINT, UDINT. Nur bei S7-1500: Array of WCHAR, WSTRING, WCHAR, ULINT, LINT, LWORD

Die Anzahl der zu konvertierenden ASCII-Zeichen legen Sie durch den Parameter N fest. Maximal können 32767 gültige ASCII-Zeichen konvertiert werden. Nur die Ziffern "0" bis "9" die Großbuchstaben "A" bis "F" und die Kleinbuchstaben "a" bis "f" können interpretiert werden. Alle anderen Zeichen werden in Nullen umgewandelt.

Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist das Ausgabewort nur halb so lang wie das Eingabewort. Die ASCII-Zeichen werden nach dem Umwandeln in dem Ausgang in der gleichen Reihenfolge angeordnet, in der sie eingelesen wurden. Wenn es sich um eine ungerade Anzahl von ASCII-Zeichen handelt, wird die Hexadezimalzahl in dem Halbbyte rechts von der zuletzt umgewandelten Hexadezimalzahl mit Nullen aufgefüllt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATH":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	D, L	Zeiger auf ASCII-Zeichenkette
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu konvertierenden ASCII-Zeichen
RET_VAL	Return	WORD	E, A, M, D, L	Status der Anweisung
OUT	Output	VARIANT	E, A, M, D, L	Hexadezimalzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#....)*	Beschreibung
0000	Kein Fehler
0007	Ungültiges Zeichen. Nur die folgenden ASCII-Zeichen dürfen verwendet werden: Ziffern "0" bis "9", Großbuchstaben "A" bis "F", Kleinbuchstaben "a" bis "f".
8182	Eingangspuffer ist zu klein für Daten am Parameter N.
8120	Ungültiges Format am Parameter IN.
8151	Nicht unterstützter Datentyp am Parameter IN.
8482	Ausgangspuffer ist zu klein für Daten am Parameter N.
8420	Ungültiges Format am Parameter OUT.
8451	Nicht unterstützter Datentyp am Parameter OUT.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

ASCII-Zeichen und Hexadezimalwerte

Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte:

ASCII-Zeichen	ASCII-codierter Hexadezimalwert	Hexadezimalziffer
"0"	30	0
"1"	31	1
"2"	32	2
"3"	33	3
"4"	34	4
"5"	35	5
"6"	36	6
"7"	37	7
"8"	38	8
"9"	39	9
"A"	41	A
"B"	42	B
"C"	43	C
"D"	44	D
"E"	45	E
"F"	46	F

Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung von ASCII-Zeichenketten in Hexadezimalzahl:

Datentyp von IN	IN	N	OUT	ENO-Status
STRING, WSTRING	'0a23'	4	• WORD#16#0A23 • UINT#10#2595	1
STRING, WSTRING	‘123aFx1a23’	10	• LWORD#16#123AF01A_23000000 • ULINT#10#1 313 626 236 378 939 392	0
STRING, WSTRING	‘a23’	3	WORD#16#A230	1
STRING, WSTRING	‘0a23’	4	• WORD#[16#0A23, 16#0000] • BYTE#[16#0A, 16#23]	1
STRING, WSTRING	‘aFbE’	1	• CHAR#‘A’ • WCHAR#‘A’	1
STRING, WSTRING	‘0a23’	4	• CHAR#[‘0’, ‘A’, ‘2’, ‘3’] • WCHAR#[‘0’, ‘A’, ‘2’, ‘3’]	1
STRING, WSTRING	‘123aFC1a23’	10	• STRING#‘123AFC1A23’ • WSTRING#‘123AFC1A23’	1
ArrayOfCHAR, Array-OfWCHAR	[‘a’, ‘B’, ‘E’, ‘3’]	4	WORD#16#ABE3	1
ArrayOfBYTE	[16#39, 16#32, 16#35]	3	WORD#16#9250	1
ArrayOfWORD	[16#1234, 16#0032, 16#0031, 16#0000]	3	WORD#16#0210	0
ArrayOfWORD	[16#0031, 16#0032, 16#0033, 16#0034]	4	WORD#16#1234	1
ArrayOfWORD	[16#AB31, 16#32AF, 16#4333, 16#0034]	4	WORD#16#0004	0
ArrayOfWORD	[16#0031, 16#0032, 16#0033, 16#0034]	4	[16#1234, 16#0000, 16#0000, 16#0000]	1
ArrayOfWORD	[16#0034, 16#0035, 16#0036, 16#0037, 16#0041, 16#0042, 16#0043, 16#0044]	8	DWORD#16#4567_ABCD	1

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

HTA: Hexadezimalzahl in ASCII-Zeichenkette konvertieren

Beschreibung

Mit der Anweisung "HTA" konvertieren Sie die am Eingang IN angegebene Hexadezimalzahl in eine ASCII-Zeichenkette. Das Ergebnis der Konvertierung wird an der am Parameter OUT angegebenen Adresse abgelegt.

- Über den Zeiger am Parameter IN (hexadezimal) können Sie auf die folgenden Datentypen verweisen: Array of CHAR, Array of BYTE, STRING, BYTE, CHAR, WORD, Array of WORD, INT, DWORD, Array of DWORD, DINT, SINT, USINT, UINT, UDINT. Nur bei S7-1500: Array of WCHAR, WSTRING, WCHAR, ULINT, LINT, LWORD, Array of LWORD
- Über den Zeiger am Parameter OUT (ASCII) können Sie auf die folgenden Datentypen verweisen: STRING, WSTRING, Array of CHAR, Array of WCHAR, Array of BYTE, Array of WORD

Die Anzahl der zu konvertierenden Hexadezimalbytes legen Sie durch den Parameter N fest. Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist der Ausgabewert doppelt so lang wie der Eingabewert. Jedes Halbbyte der Hexadezimalzahl wird in ein Zeichen unter Beibehaltung der ursprünglichen Reihenfolge umgewandelt.

In die ASCII-Zeichenkette werden maximal 32767 Zeichen geschrieben. Das Ergebnis der Konvertierung wird mit den Ziffern "0" bis "9" und Großbuchstaben "A" bis "F" dargestellt.

Kann das vollständige Ergebnis der Konvertierung nicht am Parameter OUT angezeigt werden, wird das Ergebnis nur teilweise in den Parameter geschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "HTA":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	VARIANT	E, A, M, D, L	Anfangsadresse der Hexadezimalziffer
N	Input	UINT	E, A, M, D, L oder Konstante	Anzahl der zu konvertierenden Hexadezimalbytes
RET_VAL	Return	WORD	E, A, M, D, L	Fehlermeldung
OUT	Output	VARIANT	D, L	Adresse, an der das Ergebnis abgelegt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
8182	Eingangspuffer ist zu klein für Daten am Parameter N.
8120	Ungültiges Format am Parameter IN.

Fehlercode* (W#16#....)	Beschreibung
8151	Nicht unterstützter Datentyp am Parameter IN.
8482	Ausgangspuffer ist zu klein für Daten am Parameter N.
8420	Ungültiges Format am Parameter OUT.
8451	Nicht unterstützter Datentyp am Parameter OUT.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

ASCII-Zeichen und Hexadezimalwerte

Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte:

Hexadezimalziffer	ASCII-codierter Hexadezimal-wert	ASCII-Zeichen
0	30	"0"
1	31	"1"
2	32	"2"
3	33	"3"
4	34	"4"
5	35	"5"
6	36	"6"
7	37	"7"
8	38	"8"
9	39	"9"
A	41	"A"
B	42	"B"
C	43	"C"
D	44	"D"
E	45	"E"
F	46	"F"

Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung von Hexadezimalzahlen in ASCII-Zeichenketten:

IN	N	OUT	Datentyp von OUT	ENO-Status
• WORD#16#0123 • UINT#10#291	2	'0123'	STRING, WSTRING	1
• DWORD#16#123AF012 • INT#10#305852434	4	'123AF012'	STRING, WSTRING	1
WORD#16#ABE3	2	CHAR#[‘A’, ‘B’, ‘E’, ‘3’]	ArrayOfCHAR, ArrayOfW-CHAR	1
WORD#16#9250	2	[16#39, 16#32, 16#35, 16#30]	ArrayOfBYTE	1

IN	N	OUT	Datentyp von OUT	ENO-Status
ArrayOfWORD#[16#123A, 16#0000, 16#ABCD, 16#0000]	2	[16#0031, 16#0032, 16#0033, 16#0041]	ArrayOfWORD	1
WORD#16#1234	2	[16#0031, 16#0032, 16#0033, 16#0034]	ArrayOfWORD	1
DWORD#16#4567ABCD	4	[16#0034, 16#0035, 16#0036, 16#0037, 16#0041, 16#0042, 16#0043, 16#0044]	ArrayOfWORD	1
ArrayOfD-WORD#[16#A800_0037, 16#0000_0041]	1	‘A8’	STRING, WSTRING	1
• CHAR#[‘0’, ‘A’, ‘2’, ‘3’] • WCHAR#[‘0’, ‘A’, ‘2’, ‘3’]	2	‘3041’	STRING, WSTRING	1
• STRING#‘123AFC1A23’ • WSTRING#‘123AFC1A23’	5	‘3132334146’	STRING, WSTRING	1
• Char#‘A‘ • Byte#16#41	1	‘41’	STRING, WSTRING	1

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

LEN: Länge einer Zeichenkette ermitteln

Beschreibung

Eine Variable vom Datentyp (W)STRING enthält zwei Längen: die maximale Länge und die aktuelle Länge (das ist die Anzahl der momentan gültigen Zeichen).

- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort STRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Bytes ist um 2 größer als die maximale Länge.
- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort WSTRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Worte ist um 2 größer als die maximale Länge.
- Die aktuelle Länge repräsentiert die Anzahl der tatsächlich belegten Zeichenplätze. Die aktuelle Länge ist kleiner oder gleich der maximalen Länge.

Mit der Anweisung "LEN" fragen Sie die aktuelle Länge der am Eingangsparameter IN angegebenen Zeichenkette ab. Diese wird als numerischer Wert am Ausgangsparameter OUT ausgegeben. Ein Leerstring (") hat die Länge Null.

Wenn Fehler während der Bearbeitung der Anweisung auftreten, wird am Parameter OUT der Wert "0" ausgegeben.

Hinweis

Auslesen der maximalen Länge

Mit der Anweisung "MAX_LEN (Seite 2237)" lesen Sie die maximale Länge einer Zeichenkette aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEN":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
OUT	Return	INT	E, A, M, D, L	Anzahl der gültigen Zeichen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel ermitteln Sie die Länge einer Zeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

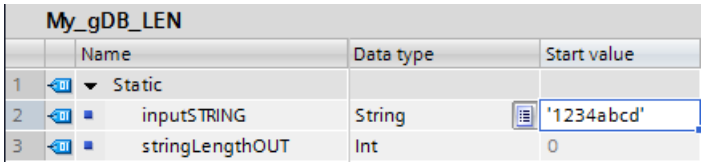

Die Parameter der Anweisung verschalten Sie wie folgt.

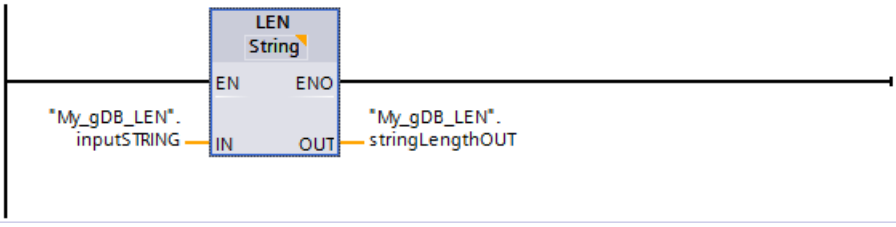

Die Anzahl der tatsächlich belegten Zeichen der Zeichenkette ("inputSTRING") wird ermittelt und am Ausgangsparameter OUT ("stringLengthOUT") als numerischer Wert ausgegeben.

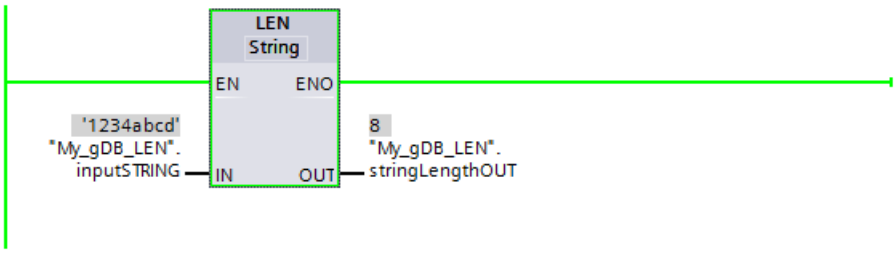

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

CONCAT: Zeichenketten zusammenfassen

Beschreibung

Mit der Anweisung "CONCAT" verbinden Sie die Zeichenkette am Eingangsparameter IN1 mit der Zeichenkette am Eingangsparameter IN2. Das Ergebnis wird am Ausgangsparameter OUT in Format (W)STRING ausgegeben. Wenn die Ergebniszeichenkette länger als die am Ausgangsparameter OUT angegebene Variable ist, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.

Wenn während der Bearbeitung der Anweisung Fehler auftreten und es möglich ist den Ausgangsparameter OUT zu beschreiben, wird ein Leerstring ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CONCAT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
IN2	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
OUT	Return	STRING, WSTRING	D, L	Ergebniszeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel verbinden Sie zwei Zeichenketten mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

My_gDB_CONCAT

		Name	Data type	Start value
1		Static		
2		inputstring1	String	'1234'
3		inputstring2	String	'abcd'
4		string1CONCstring2	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

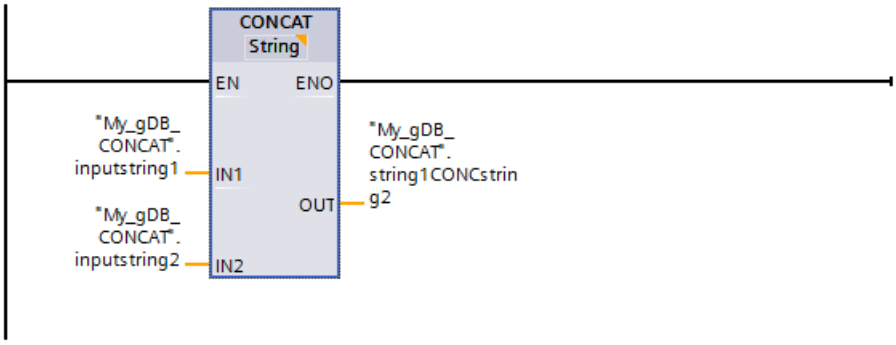

Die Zeichen der zweiten Zeichenkette ("inputstring2") werden an die erste Zeichenkette ("inputstring1") angehängt und das Ergebnis am Ausgangsparameter OUT ("string1CONCstring2") ausgegeben.

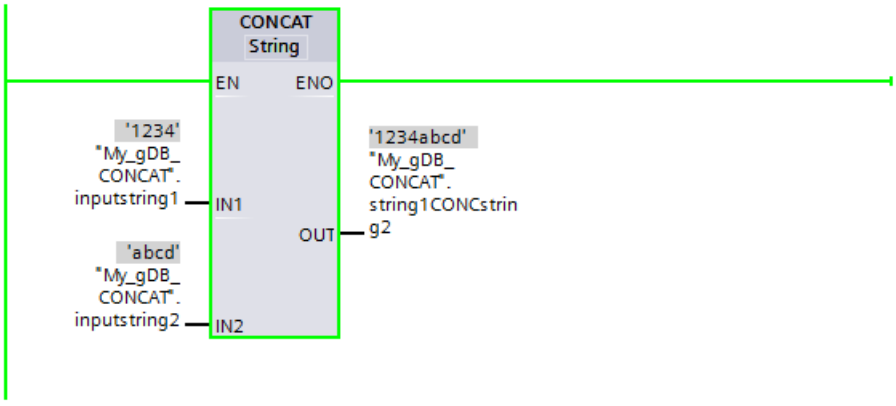

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

LEFT: Linke Zeichen einer Zeichenkette lesen

Beschreibung

Mit der Anweisung "LEFT" extrahieren Sie eine Teilzeichenkette ab dem ersten Zeichen der Zeichenkette am Eingangsparameter IN. Die Anzahl der zu extrahierenden Zeichen legen Sie am Parameter L fest. Die extrahierten Zeichen werden am Ausgangsparameter OUT in Format (W)STRING ausgegeben.

Wenn die Anzahl der zu extrahierenden Zeichen größer als die aktuelle Länge der Zeichenkette ist, liefert der Ausgangsparameter OUT die Eingangszeichenkette als Ergebnis

zurück. Beim Wert "0" am Parameter L oder bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert. Wenn der Wert am Parameter L negativ ist, wird ein Leerstring ausgegeben.

Wenn während der Bearbeitung der Anweisung Fehler auftreten und es möglich ist den Ausgangsparameter OUT zu beschreiben, wird ein Leerstring ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEFT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
L	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Anzahl der zu extrahierenden Zeichen
OUT	Return	STRING, WSTRING	D, L	Extrahierte Teilzeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel extrahieren Sie aus einer Zeichenkette ab dem ersten Zeichen eine Teilzeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

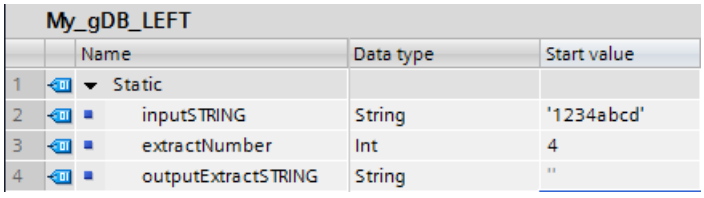

My_gDB_LEFT

	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	extractNumber	Int	4
4	outputExtractSTRING	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

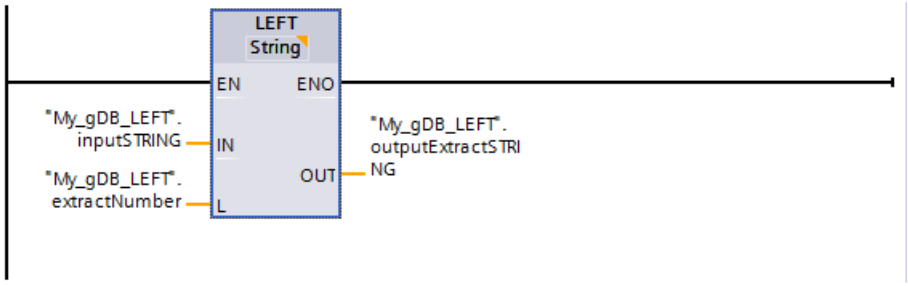

Entsprechend dem Wert "4" des Parameter L ("extractNumber") extrahieren Sie ab den ersten Zeichen von links von der Zeichenkette ("inputSTRING") eine vier Zeichen lange Teilzeichenkette. Die extrahierte Teilzeichenkette wird am Ausgangsparameter OUT ("outputExtractSTRING") ausgegeben.

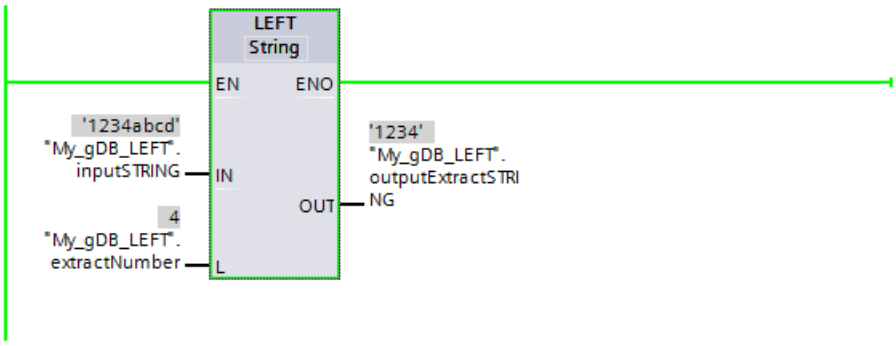

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

RIGHT: Rechte Zeichen einer Zeichenkette lesen

Beschreibung

Mit der Anweisung extrahieren Sie die letzten L Zeichen einer Zeichenkette am Eingangsparameter IN. Die Anzahl der zu extrahierenden Zeichen legen Sie am Parameter L fest. Die extrahierten Zeichen werden am Ausgangsparameter OUT in Format (W)STRING ausgegeben.

Wenn die Anzahl der zu extrahierenden Zeichen größer als die aktuelle Länge der Zeichenkette ist, liefert der Ausgangsparameter OUT die Eingangszeichenkette als Ergebnis zurück. Beim Wert "0" am Parameter L oder bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert. Wenn der Wert am Parameter L negativ ist, wird ein Leerstring ausgegeben.

Wenn während der Bearbeitung der Anweisung Fehler auftreten und es möglich ist den Ausgangsparameter OUT zu beschreiben, wird ein Leerstring ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RIGHT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
L	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Anzahl der zu extrahierenden Zeichen
OUT	Return	STRING, WSTRING	D, L	Extrahierte Teilzeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel extrahieren Sie aus einer Zeichenkette ab dem letzten Zeichen eine Teilzeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

My_gDB_RIGHT

	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	extractNumber	Int	4
4	outputExtractSTRING	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

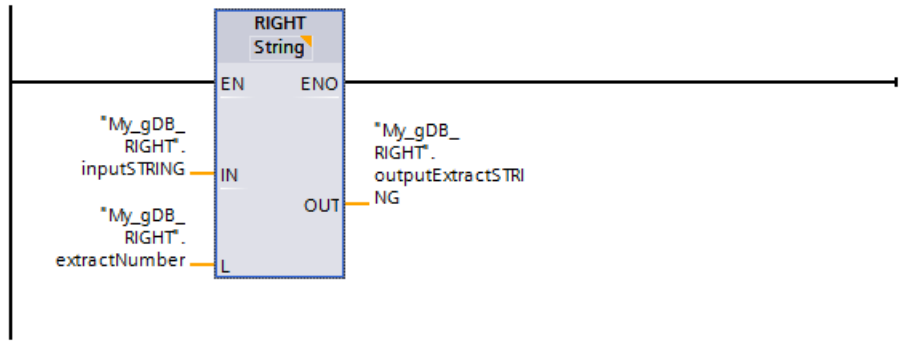

Entsprechend dem Wert "4" des Eingangsparameter L ("extractNumber") extrahieren Sie ab dem ersten Zeichen von rechts von der Zeichenkette ("inputSTRING") eine vier Zeichen lange Teilzeichenkette. Die extrahierte Teilzeichenkette wird am Ausgangsparameter OUT ("outputExtractSTRING") ausgegeben.

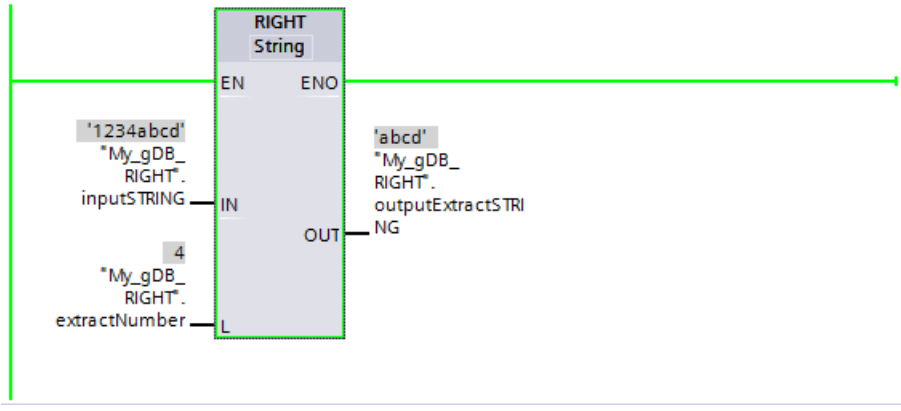

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

MID: Mittlere Zeichen einer Zeichenkette lesen

Beschreibung

Mit der Anweisung extrahieren Sie einen Teil der Zeichenkette am Eingangsparameter IN. Mit dem Parameter P legen Sie die Position des ersten zu extrahierenden Zeichens fest. Die Länge der zu extrahierenden Zeichenkette bestimmen Sie durch den Parameter L. Die extrahierte Teilzeichenkette wird am Ausgangsparameter OUT ausgegeben.

Beachten Sie bei der Ausführung der Anweisung folgende Regeln:

- Wenn die Anzahl der zu extrahierenden Zeichen die aktuelle Länge des Zeichenkette am Eingangsparameter IN überschreitet, wird eine Teilzeichenkette ausgegeben. Die Teilzeichenkette beginnt an der Zeichenposition P und wird bis zum Ende der Zeichenkette fortgesetzt.
- Wenn die am Parameter P angegebene Zeichenposition außerhalb der aktuellen Länge der Zeichenkette am Eingangsparameter IN liegt, wird eine leere Zeichenkette am Ausgangsparameter OUT ausgegeben.
- Wenn der Wert des Parameters P oder L gleich null oder negativ ist, wird eine leere Zeichenkette am Ausgangsparameter OUT ausgegeben.

Wenn während der Bearbeitung der Anweisung Fehler auftreten und es möglich ist den Ausgangsparameter OUT zu beschreiben, wird ein Leerstring ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MID":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
L	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Länge der zu extrahierenden Zeichenkette
P	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Position des ersten zu extrahierenden Zeichens (erstes Zeichen = 1)
OUT	Return	STRING, WSTRING	D, L	Extrahierte Teilzeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel extrahieren Sie aus der Mitte einer Zeichenkette eine Teilzeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_MID			
	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	startingPoint	Int	3
4	extractNumber	Int	4
5	outputExtractSTRING	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

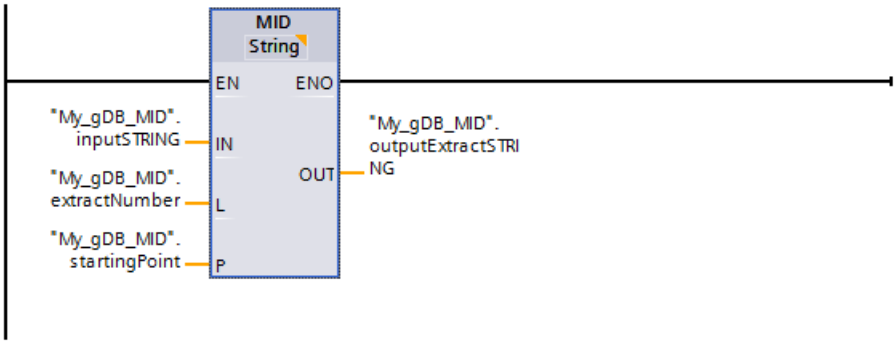

Entsprechend dem Wert "4" des Eingangsparameter L ("extractNumber") extrahieren Sie eine vier Zeichen lange Teilzeichenkette von der Zeichenkette ("inputSTRING"). Die Extraktion beginnt ab dem dritten Zeichen ("startingPoint" hat den Wert "3") der Zeichenkette ("inputSTRING"). Die extrahierte Teilzeichenkette wird am Ausgangsparameter OUT ("outputExtractSTRING") ausgegeben.

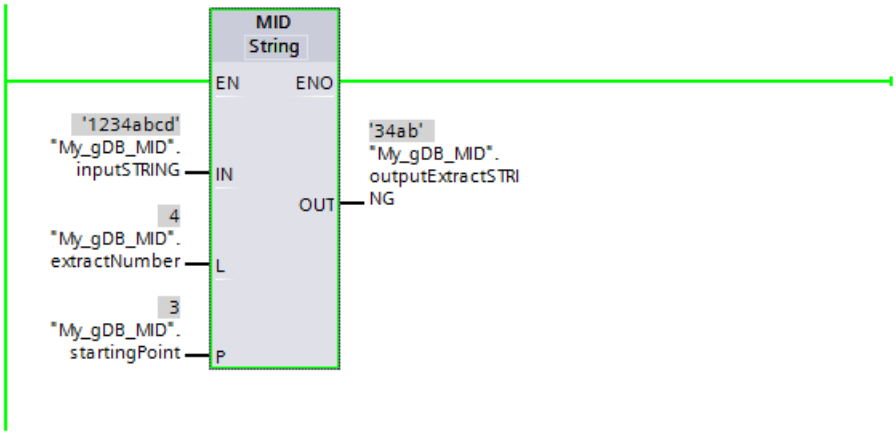

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

DELETE: Zeichen in Zeichenkette löschen

Beschreibung

Mit der Anweisung löschen Sie einen Teil der Zeichenkette am Eingangsparameter IN. Die Position des ersten zu löschenden Zeichens legen Sie mit dem Parameter P fest. Am Parameter L geben Sie die Anzahl der zu löschenden Zeichen an. Am Ausgangsparameter OUT wird die verbliebene Teilzeichenkette in Format (W)STRING ausgegeben.

Beachten Sie bei der Ausführung der Anweisung folgende Regeln:

- Wenn der Wert am Parameter P kleiner oder gleich null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn der Wert am Parameter P größer als die aktuelle Länge der Zeichenkette am Eingang IN ist, wird die Eingangszeichenkette am Ausgangsparameter OUT zurückgeliefert.
- Wenn der Wert am Parameter L gleich null ist, wird die Eingangszeichenkette am Ausgangsparameter OUT zurückgeliefert.
- Wenn die Anzahl der zu löschenden Zeichen am Parameter L größer als die Länge der Zeichenkette am Eingangsparameter IN ist, werden die Zeichen ab der Position, die durch den Parameter P vorgegeben sind, gelöscht. Die dadurch resultierende Zeichenkette wird ausgegeben.
- Wenn der Wert am Parameter L negativ ist, wird eine leere Zeichenkette ausgegeben.

Wenn während der Bearbeitung der Anweisung Fehler auftreten und es möglich ist den Ausgangsparameter OUT zu beschreiben, wird ein Leerstring ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DELETE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
L	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Anzahl der zu löschenden Zeichen
P	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Position des ersten zu löschenden Zeichens
OUT	Return	STRING, WSTRING	D, L	Ergebniszeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel löschen Sie Zeichen aus einer Zeichenkette mit dem Datentyp STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_DELETE			
	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	startingPoint	Int	3
4	deleteNumber	Int	4
5	outputResidualSTRING	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

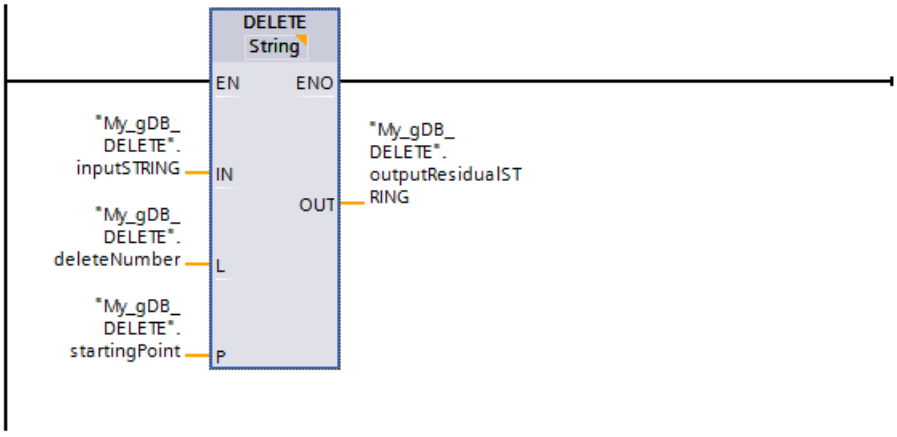

Entsprechend dem Wert "4" des Eingangsparameter L ("deleteNumber") löschen Sie vier Zeichen ab dem dritten Zeichen ("startingPoint" hat den Wert "3") von der Zeichenkette ("inputSTRING"). Die verbleibende Zeichenkette wird am Ausgangsparameter OUT ("outputResidualSTRING") ausgegeben.

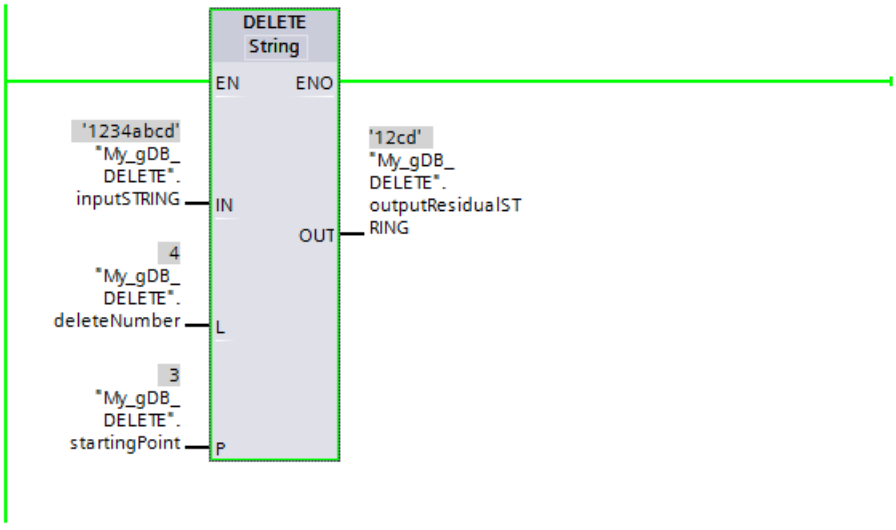

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

INSERT: Zeichen in Zeichenkette einfügen

Beschreibung

Mit der Anweisung fügen Sie die Zeichenkette am Eingangsparameter IN2 in die Zeichenkette am Eingangsparameter IN1 ein. Durch den Parameter P legen Sie die Position des Zeichens fest, ab dem die Zeichen eingefügt werden. Das Ergebnis wird am Ausgangsparameter OUT im Format (W)STRING ausgegeben.

Beachten Sie bei der Ausführung der Anweisung folgende Regeln:

- Wenn der Wert am Parameter P die aktuelle Länge der Zeichenkette am Eingangsparameter IN1 überschreitet, wird die Zeichenkette des Eingangsparameters IN2 an die des Eingangsparameters IN1 angehängt.
- Wenn der Wert am Parameter P null ist, wird am Ausgang OUT die Zeichenkette am Parameter IN2 ausgegeben, gefolgt von der Zeichenkette am Parameter IN1.
- Wenn der Wert am Parameter P negativ ist, wird am Ausgang OUT eine leere Zeichenkette ausgegeben.
- Wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter OUT angegebene Variable, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "INSERT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette
IN2	Input	STRING, WSTRING	D, L oder Konstante	Einzufügende Zeichenkette
P	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Einfügeposition
OUT	Return	STRING, WSTRING	D, L	Ergebniszeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel fügen Sie eine Zeichenkette in eine andere Zeichenkette ein. Der verwendete Datentyp ist STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

My_gDB_INSERT			
	Name	Data type	Start value
1	Static		
2	input1_STRING	String	'1234'
3	input2_STRING	String	'abcd'
4	startingPoint	Int	3
5	outputStrg1AndStrg2	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

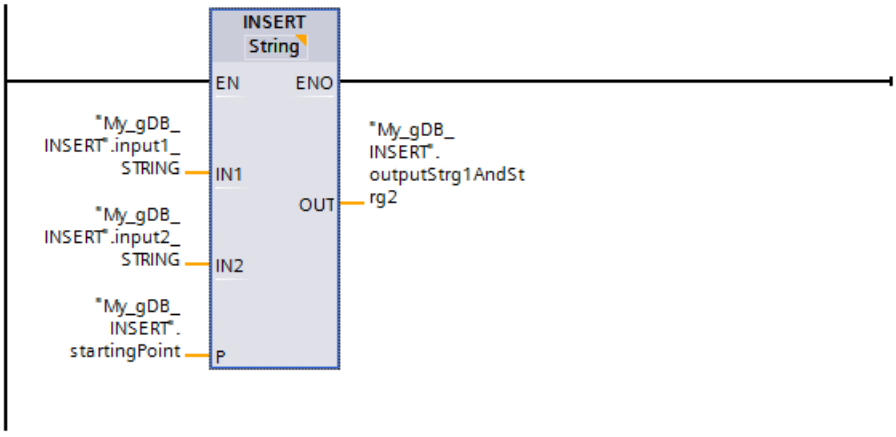

Die Zeichen der zweiten Zeichenkette ("input2_STRING") werden in die erste Zeichenkette ("input1_STRING") eingefügt. Entsprechend dem Wert "3" des Parameter P ("startingPoint") beginnt das Einfügen nach dem dritten Zeichen der ersten Zeichenkette ("input1_STRING"). Das Ergebnis wird am Ausgangsparameter OUT ("outputStrg1AndStrg2") ausgegeben.

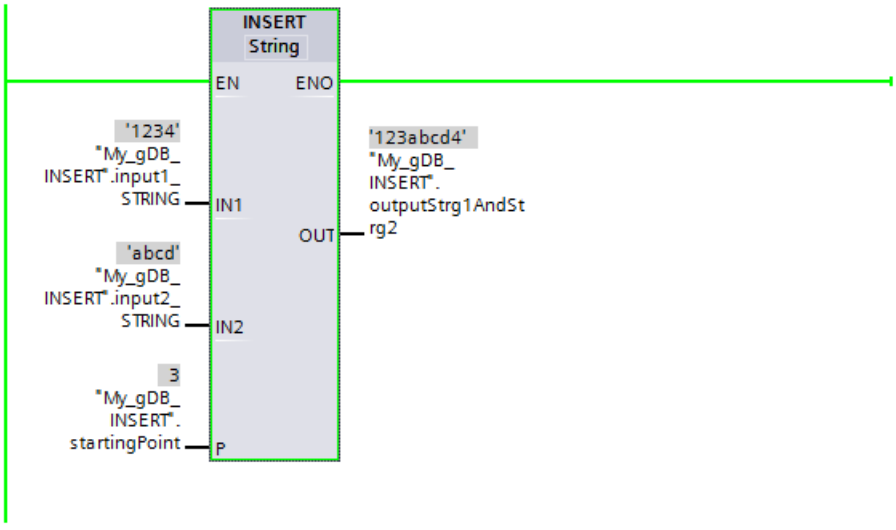

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

REPLACE: Zeichen in Zeichenkette ersetzen

Beschreibung

Mit der Anweisung ersetzen Sie einen Teil der Zeichenkette am Eingang IN1 durch die Zeichenkette des Eingangsparameters IN2. Die Position des ersten zu ersetzenden Zeichens legen Sie am Parameter P fest. Die Anzahl der zu ersetzenden Zeichen geben Sie am Parameter L an. Das Ergebnis wird am Ausgangsparameter OUT in Format (W)STRING ausgegeben.

Beachten Sie bei der Ausführung der Anweisung folgende Regeln:

- Wenn der Wert am Parameter P kleiner oder gleich null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn der Wert am Parameter L kleiner als null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn P gleich eins ist, wird die Zeichenkette am Eingang IN1 ab dem ersten Zeichen (einschließlich) ersetzt.
- Wenn der Wert am Parameter P die aktuelle Länge der Zeichenkette am Eingangsparameter IN1 überschreitet, wird die Zeichenkette des Eingangsparameters IN2 an die des Eingangsparameters IN1 angehängt.
- Wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter OUT angegebene Variable, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.
- Wenn der Wert am Parameter L gleich null ist, werden keine Zeichen ersetzt, sondern eingefügt. Es gelten ähnliche Bedingungen wie bei der Anweisung INSERT. Siehe auch: INSERT: Zeichen in Zeichenkette einfügen (Seite 2267)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "REPLACE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette, deren Zeichen ersetzt werden.
IN2	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette, deren Zeichen eingefügt werden.
L	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Anzahl der zu ersetzenden Zeichen
P	Input	BYTE, INT, SINT, USINT	E, A, M, D, L oder Konstante	Position des ersten zu ersetzenden Zeichens
OUT	Return	STRING, WSTRING	D, L	Ergebniszeichenkette

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel ersetzen Sie einen Teil einer Zeichenkette durch eine andere Zeichenkette. Der verwendete Datentyp ist STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein fünf Variablen an.

My_gDB_REPLACE			
	Name	Data type	Start value
1	Static		
2	input1_STRING	String	'1234'
3	input2_STRING	String	'abcd'
4	replaceNumber	Int	2
5	startingPoint	Int	3
6	outputNewSTRING	String	''

Die Parameter der Anweisung verschalten Sie wie folgt.

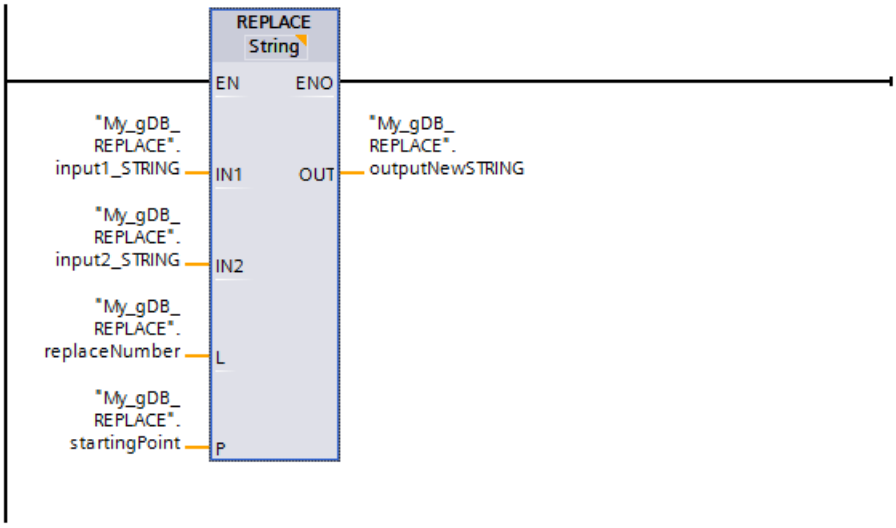

Der ersten Zeichenkette ("input1_STRING") wird ab dem dritten Zeichen ("startingPoint" hat den Wert "3") die zweite Zeichenkette ("input2_STRING") hinzugefügt. Entsprechend dem Wert "2" des Parameter L ("replaceNumber") werden dabei das dritte und vierte Zeichen der ersten Zeichenkette ("input1_STRING") ersetzt. Das Ergebnis wird am Ausgangsparameter OUT ("outputNewSTRING") ausgegeben.

My_gDB_REPLACE				
	Name	Data type	Start value	Monitor value
1	Static			
2	input1_STRING	String	'1234'	'1234'
3	input2_STRING	String	'abcd'	'abcd'
4	replaceNumber	Int	2	2
5	startingPoint	Int	3	3
6	outputNewSTRING	String	''	'12abcd'

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

FIND: Zeichen in Zeichenkette finden

Beschreibung

Mit der Anweisung durchsuchen Sie die Zeichenkette am Eingangsparameter IN1 nach einer bestimmten Zeichenfolge.

- Den zu suchenden Wert geben Sie am Eingangsparameter IN2 an. Die Suche verläuft von links nach rechts.
- Am Ausgangsparameter OUT wird die Position des ersten Auftretens ausgegeben. Wenn die Suche keine Übereinstimmung ergibt, wird am Ausgangsparameter OUT der Wert "0" ausgegeben.

Wenn am Parameter IN2 ein ungültiges Zeichen angegeben wird oder während der Bearbeitung ein Fehler auftritt, wird am Parameter OUT der Wert "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FIND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING, WSTRING	D, L oder Konstante	Zeichenkette, die durchsucht wird
IN2	Input	STRING, WSTRING	D, L oder Konstante	Zu suchende Zeichenfolge
OUT	Return	INT	E, A, M, D, L	Zeichenposition

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel suchen Sie nach einer Zeichenkette in einer anderen Zeichenkette. Der verwendete Datentyp ist STRING.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

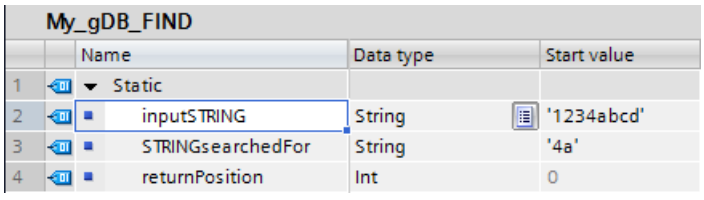

Die Parameter der Anweisung verschalten Sie wie folgt.

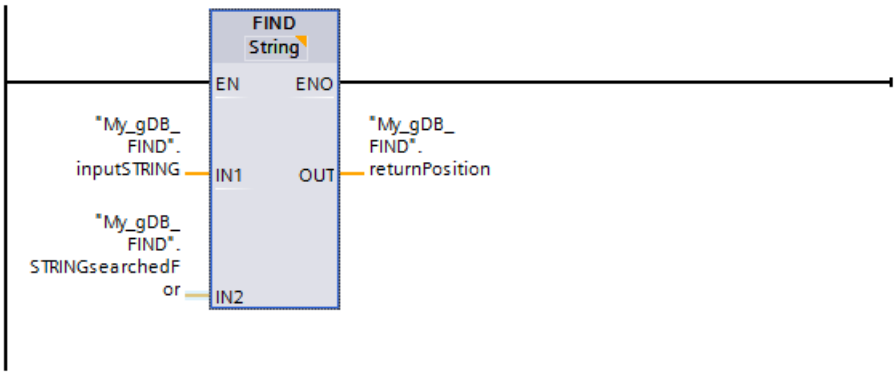

Die erste Zeichenkette ("inputSTRING") wird nach dem Wert "4a" der zweiten Zeichenkette ("STRINGsearchedFor") durchsucht. Die Zeichenposition ("4"), an der die gesuchte Zeichenkette beginnt, wird am Ausgangsparameter OUT ("returnPosition") ausgegeben.

My_gDB_FIND

		Name	Data type	Start value	Monitor value
1		Static			
2		inputSTRING	String	'1234abcd'	'1234abcd'
3		STRINGsearchedFor	String	'4a'	'4a'
4		returnPosition	Int	0	4

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Laufzeit-Informationen

GetSymbolName: Namen einer Variable am Eingangsparameter auslesen

Beschreibung

Mit der Anweisung "GetSymbolName" lesen Sie über den Eingangsparameter eines Bausteins den Namen einer am Parameter VARIABLE verschalteten Variable aus.

Hinweis

Negierte Variable am Eingangsparameter eines Bausteins

Wenn SIe "GetSymbolName" auf einen Eingangsparameter eines Bausteins anwenden, an dem eine negierte Variable verschaltet ist, wird stets ein leerer String zurückgeliefert.

Hinweis

Erhöhter Codespeicherbedarf

Die Verwendung von GetSymbolName führt zu einem erhöhten Codespeicherbedarf. Er ist um so größer, je häufiger die Anweisung aufgerufen wird und je länger die Aktualsymbole sind.

Wenn ein Baustein mehrmals im Projekt verwendet und von jeweils unterschiedlichen Variablen aufgerufen wird, haben Sie mit der Anweisung "GetSymbolName" die Möglichkeit,

den Namen der aufrufenden Variable auszuwerten. Der Prozesswert der Variablen spielt hierbei keine Rolle.

- Am Parameter VARIABLE der Anweisung geben Sie den Eingangsparameter der Bausteinschnittstelle an. Verwenden Sie an diesem Parameter keine PLC- oder Datenbaustein-Variablen, sondern ausschließlich einen Schnittstellenparameter.
- Über den Parameter SIZE beschränken Sie die Länge des gelesenen Variablennamens. Wurde der Name abgeschnitten, wird dies durch das Zeichen "..." (Unicode-Zeichen 16#2026) am Ende des Namens angezeigt. Beachten Sie, dass dieses Zeichen die Länge 1 hat.
  Ein Beispiel, um die Bedeutung des Parameters SIZE zu verdeutlichen. Über die Bausteinschnittstelle wird folgender Variablenname ausgelesen: "MyPLCTag" (Die doppelten Anführungszeichen am Anfang und am Ende gehören zum Namen dazu.)

SIZE	GetSymbolName liefert	Erklärung
1	'...'	• Anfangszeichen des WSTRING: ' • Kennung, dass der Name abgeschnitten wurde: ... • Endezeichen des WSTRING: '
2	'"...'	• Anfangszeichen des WSTRING: ' • Das erste Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "... • Endezeichen des WSTRING: '
3	'"M...'	• Anfangszeichen des WSTRING: ' • Die ersten beiden Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "M... • Endezeichen des WSTRING: '
6	'"MyPL...'	• Anfangszeichen des WSTRING: ' • Die ersten fünf Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "MyPL... • Endezeichen des WSTRING: '
0	'"MyPLCTag"'	• Anfangszeichen des WSTRING: ' • Alle Zeichen des Namens: "MyPLCTag" • Endezeichen des WSTRING: '

- An dem Parameter OUT wird der gelesene Name ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetSymbolName":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VARIABLE	Input	PARAMETER	Parameter-Abschnitte Input, Output, InOut	Auswahl der lokalen Schnittstelle, zu der Sie den Namen der Eingangsparameterversorgung auslesen.
SIZE	Input	DINT	E, A, M, D, L	Begrenzt die Anzahl der ausgegeben Zeichen am Parameter OUT: • SIZE > 0: GetSymbolName liefert die ersten SIZE Zeichen des Namens. • SIZE = 0: GetSymbolName liefert den gesamten Namen. • SIZE < 0: GetSymbolName liefert die letzten SIZE Zeichen des Namens.
OUT	Return	WSTRING	D, L	Ausgabe des Variablennamens der Eingangsparameterversorgung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel lesen Sie den Namen einer Variable aus, die über den Eingangsparameter eines Bausteins verschaltet ist.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

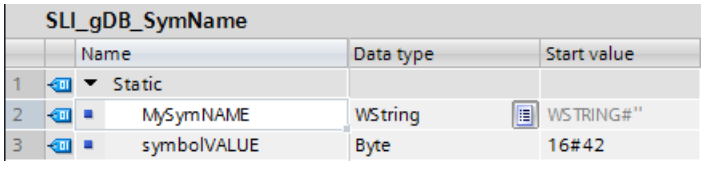

Sie erstellen einen FC "SLI_FC_GetSymbolName". In dem Baustein legen Sie die folgende lokale Variable an.

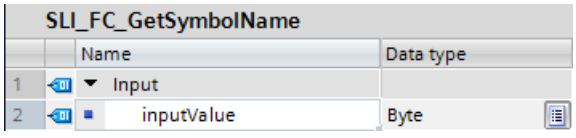

Die Anweisung "GetSymbolName" rufen Sie in dem FC auf und verschalten "GetSymbolName" wie folgt.

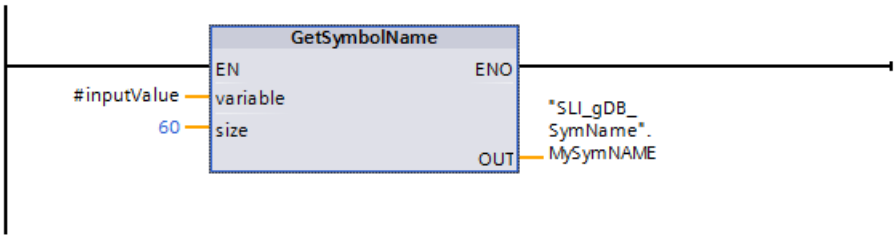

Sie erstellen einen FC "SLI_FC_FCCall_GetSymbolName". In dem FC rufen Sie den FC "SLI_FC_GetSymbolName" auf und verschalten diesen wie folgt.

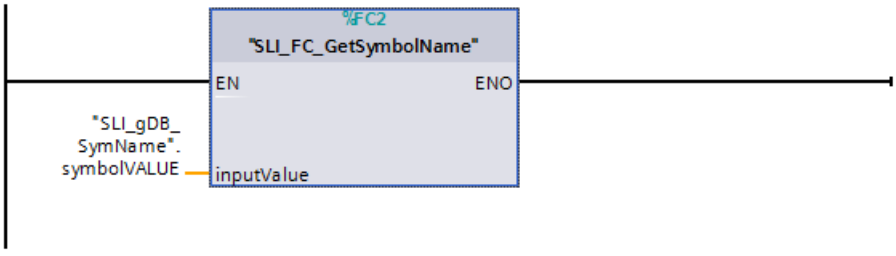

Im Baustein "SLI_FC_GetSymbolName" wird die Anweisung "GetSymbolName" ausgeführt. Über den Eingangsparameter VARIABLE der Anweisung wird der Eingangsparameter inputValue des Bausteins "SLI_FB_GetSymbolName" nach seiner Verschaltung untersucht. Dabei wird die Variable "symbolVALUE" ausgelesen und am Ausgangsparameter OUT ("MySymNAME") als Zeichenkette ausgegeben. Die Länge der Zeichenkette ist entsprechend dem Wert des Eingangsparameters SIZE auf 60 Zeichen beschränkt.

SLI_gDB_SymName

	Name	Data type	Start value	Monitor value
1	Static			
2	MySymNAME	WString	WSTRING#''	WSTRING#'"SLI_gDB_SymName".symbolVALUE'
3	symbolVALUE	Byte	16#42	16#42

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetSymbolPath: Aktualparameter am Beginn eines Aufrufpfades abfragen

Beschreibung

Mit der Anweisung "GetSymbolPath" lesen Sie den Namen eines Parameters, der am Beginn eines Aufrufpfades übergeben und über mehrere Bausteinaufrufe durchgereicht wird.

Hinweis

Erhöhter Codespeicherbedarf

Die Verwendung von GetSymbolPath führt zu einem erhöhten Codespeicherbedarf. Er ist um so größer, je häufiger die Anweisung aufgerufen wird und je länger die Aktualsymbole sind.

- Am Parameter VARIABLE der Anweisung geben Sie den Formalparameternamen an, von dem zur Laufzeit der Name des beim Aufruf verschalteten Aktualparameters gelesen werden soll.
  - Wenn zur Eingangsparameterversorgung die Variable eines Datenbausteins verwendet wird, so wird diese Variable - bestehend aus Name des DB und Zugriffspfad auf die eventuell strukturierte Variable - ausgegeben.
  - Wenn zur Eingangsparameterversorgung eine PLC-Variable verwendet wird, so wird der Name der PLC-Variable ausgegeben.
  - Wenn zur Eingangsparameterversorgung eine Konstante verwendet wird, so wird diese Konstante ausgegeben.
- Über den Parameter SIZE beschränken Sie die Länge des gelesenen Variablennamens. Wenn der Name abgeschnitten wurde, wird dies durch das Zeichen "..." (Unicode-Zeichen 16#2026) am Ende des Namens angezeigt. Beachten Sie, dass dieses Zeichen die Länge 1 hat.
  Ein Beispiel, um die Bedeutung des Parameters SIZE zu verdeutlichen. Über die Bausteinschnittstelle werde folgender Variablenname ausgelesen: "MyPLCTag" (Die doppelten Anführungszeichen am Anfang und am Ende gehören zum Namen dazu.)

SIZE	GetSymbolPath liefert	Erklärung
1	'...'	• Anfangszeichen des WSTRING: ' • Kennung, dass der Name abgeschnitten wurde: ... • Endezeichen des WSTRING: '
2	'"...'	• Anfangszeichen des WSTRING: ' • Das erste Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "... • Endezeichen des WSTRING: '
3	'"M...'	• Anfangszeichen des WSTRING: ' • Die ersten beiden Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "M... • Endezeichen des WSTRING: '
6	'"MyPL...'	• Anfangszeichen des WSTRING: ' • Die ersten fünf Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "MyPL... • Endezeichen des WSTRING: '
0	'"MyPLCTag"'	• Anfangszeichen des WSTRING: ' • Alle Zeichen des Namens: "MyPLCTag" • Endezeichen des WSTRING: '

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetSymbolPath":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VARIABLE	Input	PARAMETER	Parameter-Abschnitte Input, Output, InOut	Auswahl des Formalparameters, zu dem Sie den Namen des Aktualparameters am Beginn des Aufrufpfades auslesen möchten.
SIZE	Input	DINT	E, A, M, D, L oder Konstante	Begrenzt die Anzahl der ausgegebenen Zeichen am Parameter OUT: • SIZE > 0: GetSymbolPath liefert die ersten SIZE Zeichen des Namens. • SIZE = 0: GetSymbolPath liefert den gesamten Namen. • SIZE < 0: GetSymbolPath liefert die letzten SIZE Zeichen des Namens.
OUT	Output	WSTRING	D, L	Ausgabe des Variablennamens der Eingangsparameterversorgung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

Das folgende Beispiel zeigt die Verwendung von GetSymbolPath über mehrere Aufrufebenen:

- Aus dem Organisationsbaustein OB1 wird der Baustein FB_Level_1 aufgerufen. Aus diesem wird wiederum der Baustein FB_Level_2 aufgerufen.
- In dem Baustein FB_Level_2 wird GetSymbolPath ausgeführt, um den Namen des Parameters auszulesen, dessen Wert am Beginn des Aufrufpfades an REQ2 durchgereicht wird.
- Am Parameter REQ1 wird die Variable MyStarterBit als Aktualparameter verschaltet. Das Bit liegt im Datenbaustein MyDatablock in der Struktur MySTRUCT.
  Diese Information wird durch GetSymbolPath ausgelesen und an dem Parameter OUT ausgegeben.

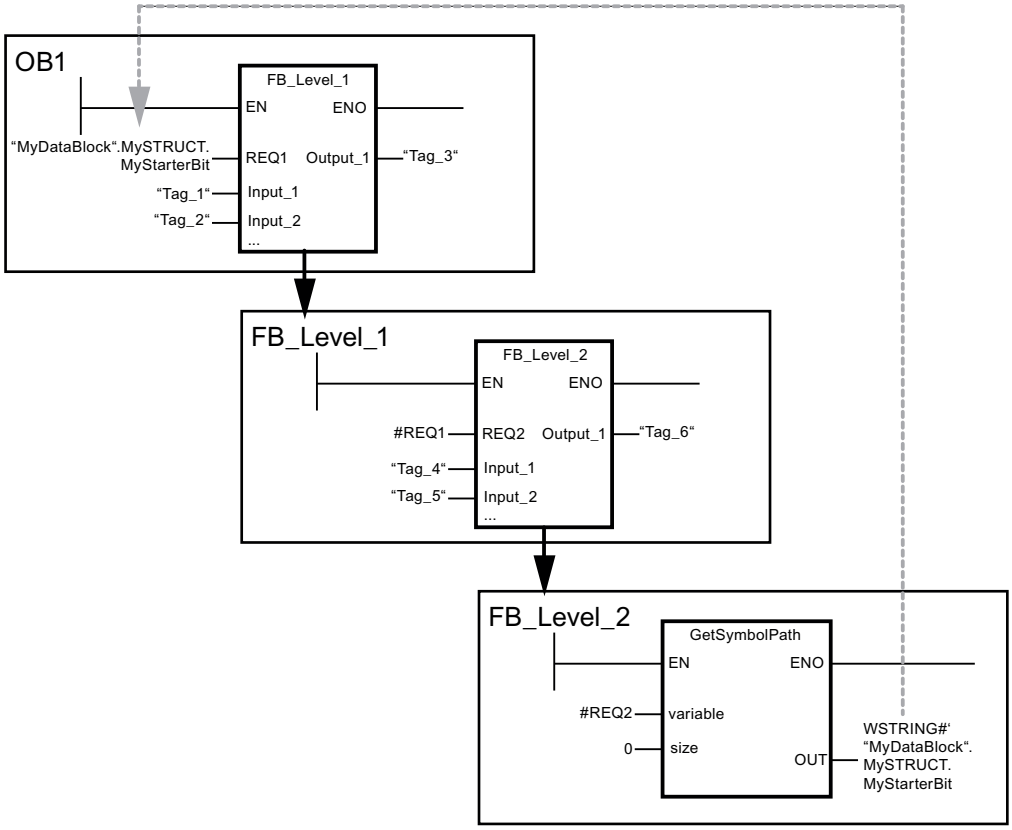

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetInstanceName: Namen der Baustein-Instanz auslesen

Beschreibung

Über die Anweisung "GetInstanceName" lesen Sie innerhalb eines Funktionsbausteins den Namen des Instanz-Datenbausteins aus.

Hinweis

Erhöhter Codespeicherbedarf

Die Verwendung von GetInstanceName führt zu einem erhöhten Codespeicherbedarf. Er ist um so größer, je häufiger die Anweisung aufgerufen wird, je länger der Name des Instanz-Datenbausteins ist und je weniger Sie diese Länge beschränken.

- Über den Parameter SIZE beschränken Sie die Länge des gelesenen Instanz-Namens. Wenn der Name abgeschnitten wurde, wird dies durch das Zeichen "..." (Unicode-Zeichen 16#2026) am Ende des Namens angezeigt. Beachten Sie, dass dieses Zeichen die Länge 1 hat.
  Ein Beispiel, um die Bedeutung des Parameters SIZE zu verdeutlichen. Über die Bausteinschnittstelle wird folgender Instanz-Name ausgelesen: "Level1_DB" (Die doppelten Anführungszeichen am Anfang und am Ende gehören zum Namen dazu.)

SIZE	GetInstanceName liefert	Erklärung
1	'...'	• Anfangszeichen des WSTRING: ' • Kennung, dass der Name abgeschnitten wurde: ... • Endezeichen des WSTRING: '
2	'"...'	• Anfangszeichen des WSTRING: ' • Das erste Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "... • Endezeichen des WSTRING: '
3	'"L...'	• Anfangszeichen des WSTRING: ' • Die ersten beiden Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "L... • Endezeichen des WSTRING: '
6	'"Leve...'	• Anfangszeichen des WSTRING: ' • Die ersten fünf Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "Leve... • Endezeichen des WSTRING: '
0	'"Level1_DB"'	• Anfangszeichen des WSTRING: ' • Alle Zeichen des Namens: "Level1_DB" • Endezeichen des WSTRING: '

- An dem Parameter OUT wird der Name des Instanz-Datenbausteins geschrieben. Wenn der Name des Instanz-Datenbausteins länger als die maximale Länge von WSTRING ist, wird der Name abgeschnitten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetInstanceName":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIZE	Input	DINT	E, A, M, D, L oder Konstante	Begrenzt die Anzahl der ausgegebenen Zeichen am Parameter OUT: • SIZE > 0: GetInstanceName liefert die ersten SIZE Zeichen des Namens. • SIZE = 0: GetInstanceName liefert den gesamten Namen. • SIZE < 0: GetInstanceName liefert die letzten SIZE Zeichen des Namens.
OUT	Output	WSTRING	D, L	Gelesener Name des Instanz-Datenbausteins

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel lesen Sie den Namen eines Instanz-Datenbausteins aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

SLI_gDB_GetInstanceName

	Name	Data type	Start value
1	Static		
2	limitSIZE	DInt	0
3	outputInstName	WString	WSTRING#''

Sie legen einen FB "SLI_FB_GetInstanceName" an. Die Parameter der Anweisung "GetInstanceName" verschalten Sie wie folgt.

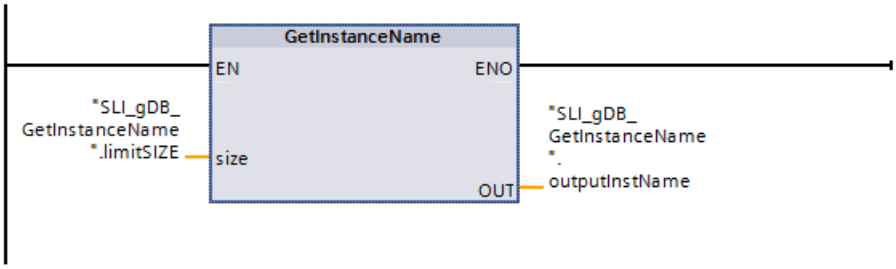

Im Baustein "SLI_FB_GetInstanceName" wird die Anweisung "GetInstanceName" ausgeführt. Über die Anweisung "GetInstanceName" wird der zugehörige Instanz-Datenbaustein des Bausteins "SLI_FB_GetInstanceName" ermittelt und am Ausgangsparameter OUT ("outputInstName") als Zeichenkette ausgegeben. Die Länge der Zeichenkette ist entsprechend dem Wert "0" des Parameters SIZE ("limitSIZE") unbeschränkt.

SLI_gDB_GetInstanceName

	Name	Data type	Start value	Monitor value
1	Static			
2	limitSIZE	DInt	0	0
3	outputInstName	WString	WSTRING#''	WSTRING#'"SLI_FB_GetInstanceName_DB"'

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetInstancePath: Zusammengesetzten globalen Namen der Baustein-Instanz abfragen

Beschreibung

Über die Anweisung "GetInstancePath" lesen Sie innerhalb eines Funktionsbausteins den zusammengesetzten globalen Namen der Baustein-Instanz aus. Der zusammengesetzte globale Name der Baustein-Instanz ist der Pfad der gesamten Aufrufhierarchie bei Verwendung von Multiinstanzen.

Hinweis

Erhöhter Codespeicherbedarf

Die Verwendung von GetInstancePath führt zu einem erhöhten Codespeicherbedarf. Er ist um so größer, je länger die Name und je größer die Parameteranzahl ist.

In dem folgenden Beispiel wird die Anweisung "GetInstancePath" in dem Funktionsbaustein FB_Level_3 aufgerufen.

- Der Funktionsbaustein FB_Level_3 legt seine Daten in dem aufrufenden Funktionsbaustein FB_Level_2 ab.
- Der Funktionsbaustein FB_Level_2 legt wiederum seine Daten in dem aufrufenden Funktionsbaustein FB_Level_1 ab.
- Der Funktionsbaustein FB_Level_1 legt seine Daten wiederum in seinem Instanz-Datenbaustein IDB_LEVEL_1 ab. Durch die Verwendung von Multiinstanzen enthält der Instanz-Datenbaustein von FB_Level_1 alle Daten der drei Funktionsbausteine.

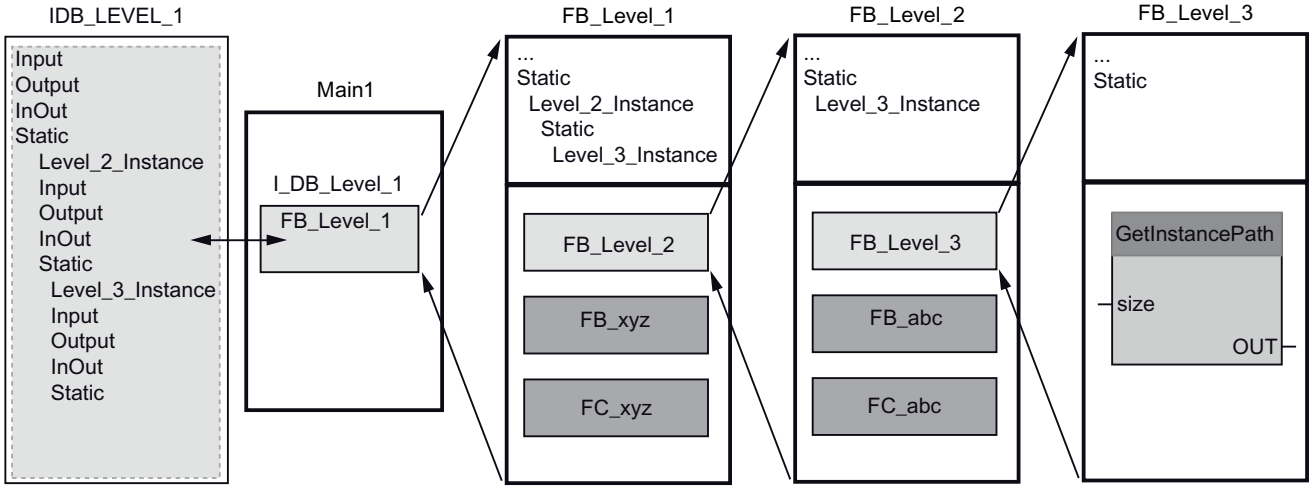

Die Anweisung "GetInstancePath" liefert für dieses Beispiel die folgenden Werte zurück:

SIZE	GetInstancePath liefert	Erklärung
1	'...'	• Anfangszeichen des WSTRING: ' • Kennung, dass der Name abgeschnitten wurde: ... • Endezeichen des WSTRING: '
2	'"...'	• Anfangszeichen des WSTRING: ' • Das erste Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "... • Endezeichen des WSTRING: '
3	'"I...'	• Anfangszeichen des WSTRING: ' • Die ersten beiden Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "I... • Endezeichen des WSTRING: '
6	'"IDB_...'	• Anfangszeichen des WSTRING: ' • Die ersten fünf Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "IDB_... • Endezeichen des WSTRING: '
0	'"IDB_LEVEL_1".Level_2_ Instance.Level_3_Instance'	• Anfangszeichen des WSTRING: ' • Alle Zeichen des Namens: "IDB_LEVEL_1".Level_2_Instance.Level_3_Instance • Endezeichen des WSTRING: '

Hinweis

Verwendung von "GetInstancePath" in Funktionsbausteinen mit Einzel-Instanz

Speichert der Funktionsbaustein in dem Sie "GetInstancePath" aufrufen in einem eigenen Instanz-Datenbaustein, wird als globaler Name der Name der Einzel-Instanz ausgegeben. Das Ergebnis am Parameter OUT entspricht in diesem Fall der Anweisung "GetInstanceName (Seite 2279)".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetInstancePath":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIZE	Input	DINT	E, A, M, D, L oder Konstante	Begrenzt die Anzahl der ausgegebenen Zeichen am Parameter OUT: • SIZE > 0: GetInstancePath liefert die ersten SIZE Zeichen des Namens. • SIZE = 0: GetInstancePath liefert den gesamten Namen. • SIZE < 0: GetInstancePath liefert die letzten SIZE Zeichen des Namens.
OUT	Output	WSTRING	D, L	Gelesener globaler Name der Baustein-Instanz. Ist der globale Name der Baustein-Instanz länger als die maximale Länge von WSTRING (254 Zeichen), wird der Name abgeschnitten.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel lesen Sie den zusammengesetzten globalen Namen der Baustein-Instanz aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein eine Variable an.

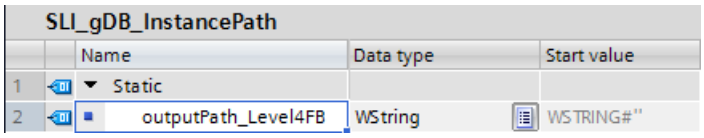

Sie legen einen FB "SLI_FB_GIP_Level4" an und rufen darin die Anweisung "GetInstancePath" auf.

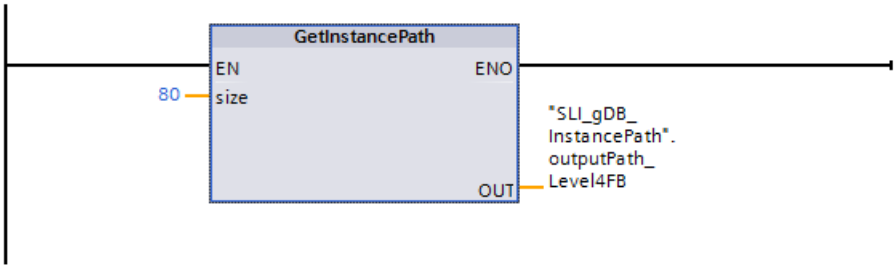

Sie legen einen FB "SLI_FB_GIP_Level3" an und rufen darin den FB "SLI_FB_GIP_Level4" auf.

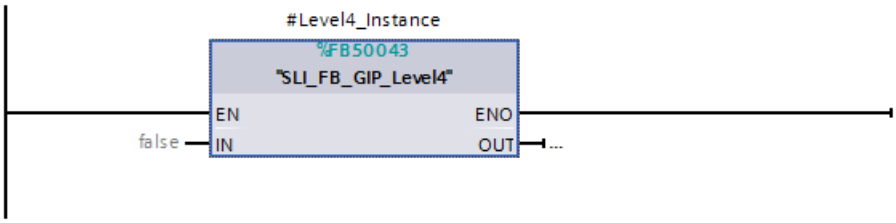

Sie legen einen FB "SLI_FB_GIP_Level2" an und rufen darin den FB "SLI_FB_GIP_Level3" auf.

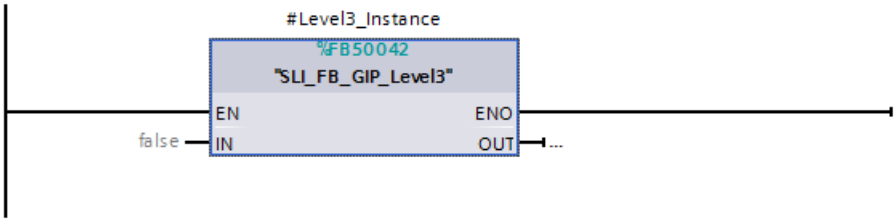

Sie legen einen FB "SLI_FB_GIP_Level1" an und rufen darin den FB "SLI_FB_GIP_Level2" auf.

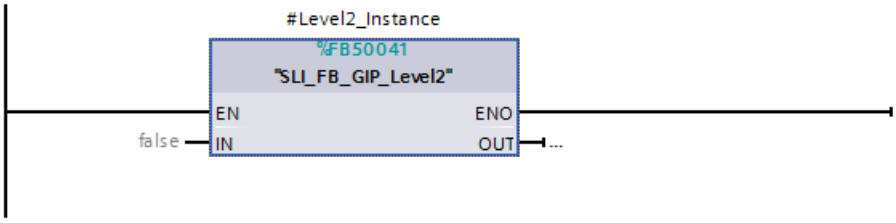

In dem FB "SLI_FB_GIP_Level1" legen Sie folgende lokale Variablen an.

SLI_FB_GIP_Level1

		Name	Data type	Default value
1		Input		
2		IN	Bool	false
3		inputValue	Byte	16#0
4		Output		
5		OUT	Bool	false
6		InOut		
7		<Add new>		
8		Static		
9		Level2_Instance	"SLI_FB_GIP_Level2"	

Sie legen einen Zyklus-OB ("OB123") an und rufen darin den FB "SLI_FB_GIP_Level1" auf. Den FB verschalten Sie wie folgt.

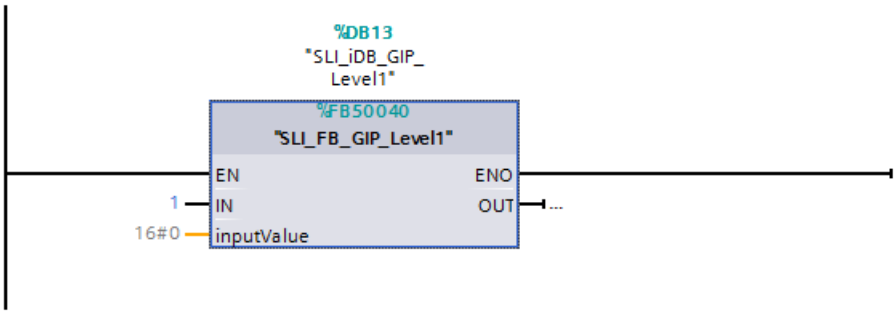

Im OB123 wird die Anweisung "GetInstancePath" ausgeführt. Die Anweisung "GetInstancePath" liesst die Instanzen aller Funktionsbausteine aus, vom Aufruf der Anweisung im FB "SLI_FB_GIP_Level4" bis zum Aufruf im OB123. Die Namen der Instanzen werden als Zeichenkette am Ausgangsparameter OUT ("outputPath_Level4FB") der Anweisung "GetInstancePath" ausgegeben. Die Länge der Zeichenkette ist entsprechend dem Wert des Eingangsparameters SIZE der Anweisung "GetInstancePath" auf 80 Zeichen beschränkt.

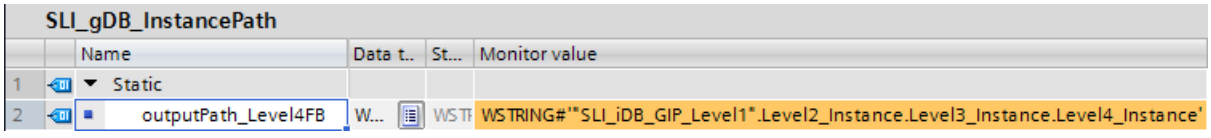

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetBlockName: Name des Bausteins auslesen

Beschreibung

Über die Anweisung "GetBlockName" lesen Sie den Namen des Bausteins aus, in dem die Anweisung aufgerufen wird.

- Wenn die Länge des Bausteinnamens auf eine bestimmte Anzahl Zeichen begrenzt werden soll, geben Sie am Parameter SIZE die maximale Länge an. Wenn der Name abgeschnitten wurde, wird dies durch das Zeichen "..." (Unicode-Zeichen 16#2026) am Ende des Namens angezeigt. Beachten Sie, dass dieses Zeichen die Länge 1 hat.
  Ein Beispiel, um die Bedeutung des Parameters SIZE zu verdeutlichen. Folgender Baustein-Name wird ausgelesen: "Level1_gbn" (Die doppelten Anführungszeichen am Anfang und am Ende gehören zum Namen dazu.)

SIZE	GetBlockName liefert	Erklärung
1	'...'	• Anfangszeichen des WSTRING: ' • Kennung, dass der Name abgeschnitten wurde: ... • Endezeichen des WSTRING: '
2	'"...'	• Anfangszeichen des WSTRING: ' • Das erste Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "... • Endezeichen des WSTRING: '
3	'"L...'	• Anfangszeichen des WSTRING: ' • Die ersten beiden Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "L... • Endezeichen des WSTRING: '
6	'"Leve...'	• Anfangszeichen des WSTRING: ' • Die ersten fünf Zeichen des Namens und Kennung, dass der Name abgeschnitten wurde: "Leve... • Endezeichen des WSTRING: '
0	'"Level1_gbn"'	• Anfangszeichen des WSTRING: ' • Alle Zeichen des Namens: "Level1_gbn" • Endezeichen des WSTRING: '

- An dem Parameter RET_VAL wird der Name des Bausteins geschrieben. Wenn der Name des Bausteins länger als die maximale Länge von WSTRING ist, wird der Name abgeschnitten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetBlockName":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIZE	Input	UINT	E, A, M, D, L oder Konstante	Begrenzt die Anzahl der ausgegebenen Zeichen am Parameter RET_VAL: • SIZE > 0: GetBlockName liefert die ersten SIZE Zeichen des Namens. • SIZE = 0: GetBlockName liefert den gesamten Namen. • SIZE < 0: GetBlockName liefert die letzten SIZE Zeichen des Namens.
RET_VAL	Output	WSTRING	D, L	Gelesener Name des Instanz-Datenbausteins

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

In dem folgenden Beispiel lesen Sie den Bausteinnamen aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwei Variablen an.

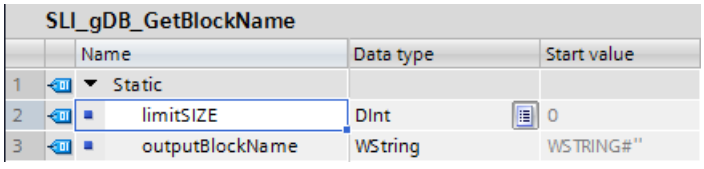

Die Parameter der Anweisung verschalten Sie wie folgt.

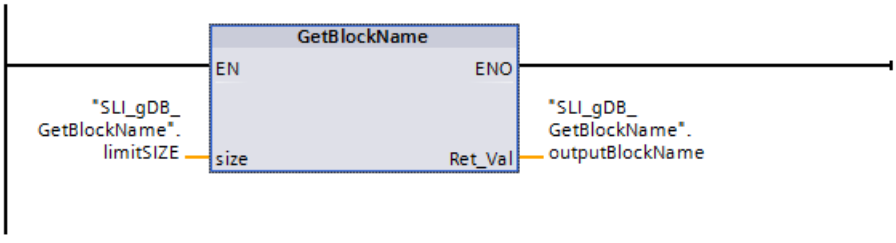

Im Baustein "SLI_FC_GetBlockName" wird die Anweisung "GetBlockName" ausgeführt. Über die Anweisung "GetBlockName" wird der Name des Bausteins "SLI_FC_GetBlockName" ausgelesen und am Ausgangsparameter RET_VAL ("outputBlockName") als Zeichenkette ausgegeben. Die Länge der Zeichenkette ist entsprechend dem Wert "0" des Parameters SIZE ("limitSIZE") unbeschränkt.

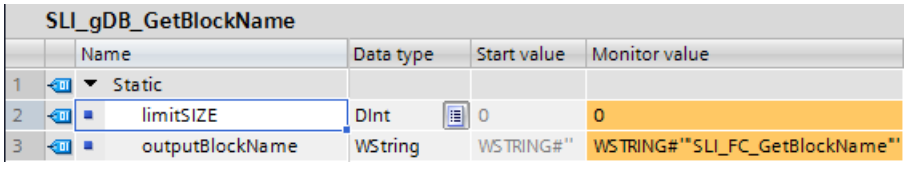

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.4.3 Prozessabbild

UPDAT_PI: Prozessabbild der Eingänge aktualisieren

Beschreibung

Mit der Anweisung aktualisieren Sie ein per Projektierung definiertes Teilprozessabbild der Eingänge.

Falls Sie als Meldeverfahren für die systemseitige Prozessabbildaktualisierung das wiederholte Melden aller Peripheriezugriffsfehler projektiert haben, wird die Aktualisierung des ausgewählten Prozessabbilds stets durchgeführt.

Sonst wird diese Aktualisierung nur dann durchgeführt, wenn das ausgewählte Teilprozessabbild nicht systemseitig aktualisiert wird, d. h. wenn Sie dieses Teilprozessabbild keinem Alarm-OB zugeordnet haben.

Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Eingänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Eingänge.
Ein Teilprozessabbild, das Sie mit "UPDAT_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC_PI (Seite 2295)" aktualisieren.

Die systemseitige Aktualisierung des OB 1-Prozessabbilds der Eingänge und der Teilprozessabbilder der Eingänge, die Sie einem Alarm-OB zugeordnet haben, findet unabhängig von "UPDAT_PI"-Aufrufen statt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT_PI":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PART	Input	PIP	E, A, M, D, L oder Konstante	Nummer des zu aktualisierenden Teilprozessabbilds der Eingänge. Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 1 <= n <= 31 (bedeutet Teilprozessabbild n).
RET_VAL	Return	INT	E, A, M, D, L, P	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L, P	Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Unzulässiger Wert beim Parameter PART
8091	Das angegebene Teilprozessabbild wurde noch nicht definiert.
8092	Das Teilprozessabbild wird systemseitig aktualisiert und darf nicht für den Parameter PART verwendet.
8093	Die Aktualisierung des Teilprozessabbilds wird momentan in einem anderen OB verarbeitet.
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, sind auch Fehlercodes der Anweisung "DPRD_DAT (Seite 2434)" möglich. Dies ist der Fall, wenn Sie für die Konsistenzbereiche größer als 32 Bytes definiert haben.

Beispiel

In dem folgenden Beispiel aktualisieren Sie das per Projektierung definierte Teilprozessabbild "1" der Eingange.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

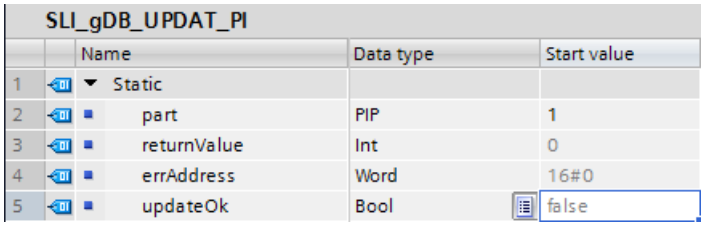

SLI_gDB_UPDAT_PI

	Name	Data type	Start value
1	Static		
2	part	PIP	1
3	returnValue	Int	0
4	errAddress	Word	16#0
5	updateOk	Bool	false

Die Parameter der Anweisung verschalten Sie wie folgt.

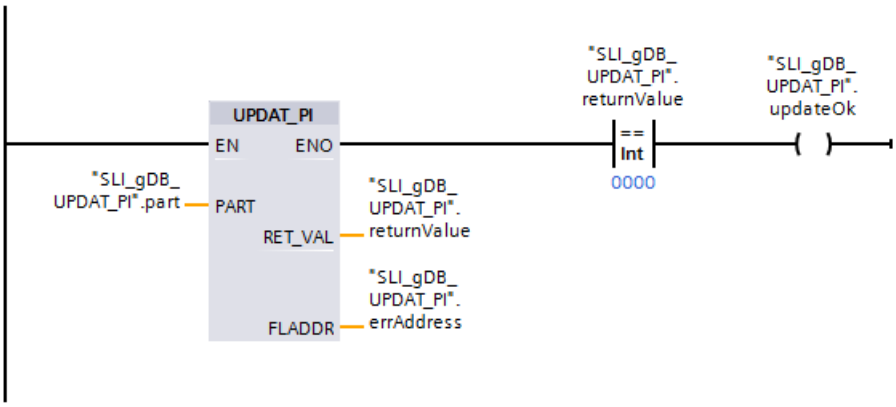

Sie legen zu Ihrer CPU eine Eingabebaugruppe (DI) an und projektieren unter "Eigenschaften > Eingang 0 - 15 > E/A-Adressen" das Teilprozessabbild "1". Zudem ordnen Sie das Teilprozessabbild keinem OB zu.

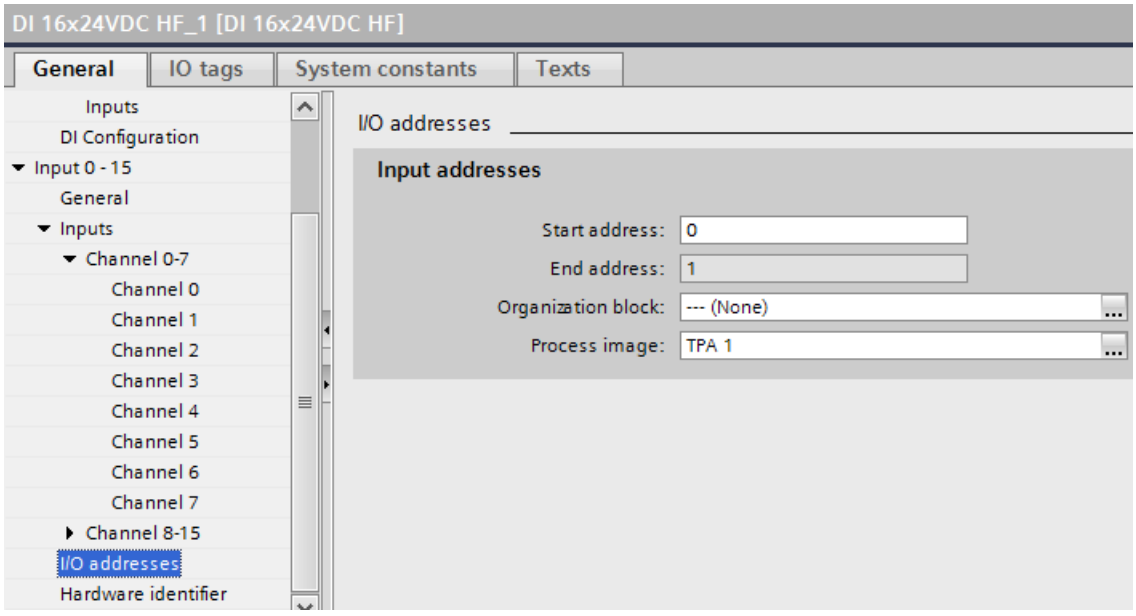

Uber den Eingangsparameter PART ("part") der Anweisung "UPDAT_PI" ist die Nummer des zu aktualisierenden Teilprozessabbilds hinterlegt. Beim Aufruf der Anweisung "UPDAT_PI" wird das angegebene Teilprozessabbild aktualisiert.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Die fehlerlose Verarbeitung setzt die Spule ("updateOk") auf "TRUE". Da kein

Zugriffsfehler auftritt, zeigt die Fehleradresse ("errAddress") der Anweisung "UPDAT_PI" den Wert "16#0000".

SLI_gDB_UPDAT_PI				
	Name	Data type	Start value	Monitor value
1	Static			
2	part	PIP	1	1
3	returnValue	Int	0	0
4	errAddress	Word	16#0	16#0000
5	updateOk	Bool	false	TRUE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

UPDAT_PO: Prozessabbild der Ausgänge aktualisieren

Beschreibung

Mit der Anweisung übertragen Sie die Signalzustände eines per Projektierung definierten Teilprozessabbilds der Ausgänge zu den Ausgabebaugruppen.

Falls Sie für das ausgewählte Teilprozessabbild einen Konsistenzbereich definiert haben, werden die zugehörigen Daten konsistent an die entsprechende Peripheriebaugruppe übertragen.

Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Ausgänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Ausgänge. Ausgänge, die Sie mit "UPDAT_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC_PO (Seite 2297)" aktualisieren.

Die systemseitige Übertragung zu den Ausgabebaugruppen von dem OB 1-Prozessabbilds der Ausgänge und der Teilprozessabbilder der Ausgänge, die Sie einem Alarm-OB zugeordnet haben, erfolgt unabhängig von "UPDAT_PO"-Aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT_PO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PART	Input	PIP	E, A, M, D, L oder Konstante	Nummer des zu übertragenden Teilprozessabbilds der Ausgänge. Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 1 <= n <= 31 (bedeutet Teilprozessabbild n)
RET_VAL	Return	INT	E, A, M, D, L, P	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L, P	Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Unzulässiger Wert beim Parameter PART
8091	Das angegebene Teilprozessabbild wurde noch nicht definiert.
8092	Das Teilprozessabbild wird systemseitig aktualisiert und darf nicht für den Parameter PART verwendet werden.
8093	Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, sind auch Fehlercodes der Anweisung "DPWR_DAT (Seite 2436)" möglich. Dies ist der Fall, wenn Sie für die Konsistenzbereiche größer als 32 Bytes definiert haben.

Beispiel

In dem folgenden Beispiel aktualisieren Sie das per Projektierung definierte Teilprozessabbild "1" der Ausgänge.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

SLI_gDB_UPDAT_PO

		Name	Data type	Start value
1		Static		
2		part	PIP	1
3		returnValue	Int	0
4		errAddress	Word	16#0
5		updateOk	Bool	false

Die Parameter der Anweisung verschalten Sie wie folgt.

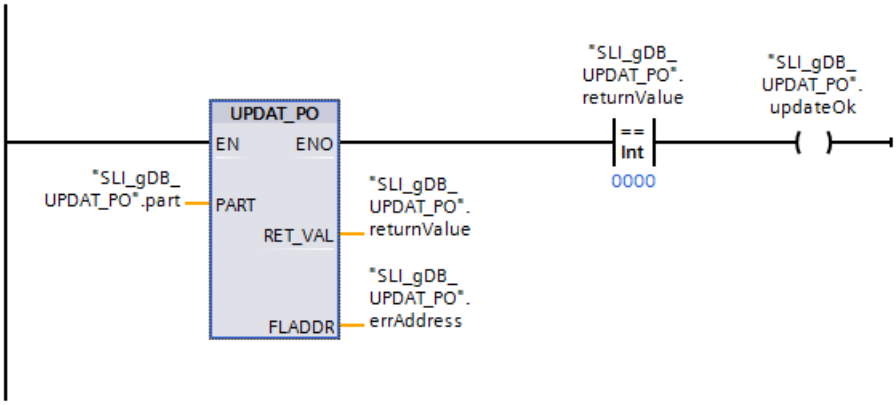

Sie legen zu Ihrer CPU eine Ausgabebaugruppe (DQ) an und projektieren unter "Eigenschaften > Ausgang 0 - 15 > E/A-Adressen" das Teilprozessabbild "1". Zudem ordnen Sie das Teilprozessabbild keinem OB zu.

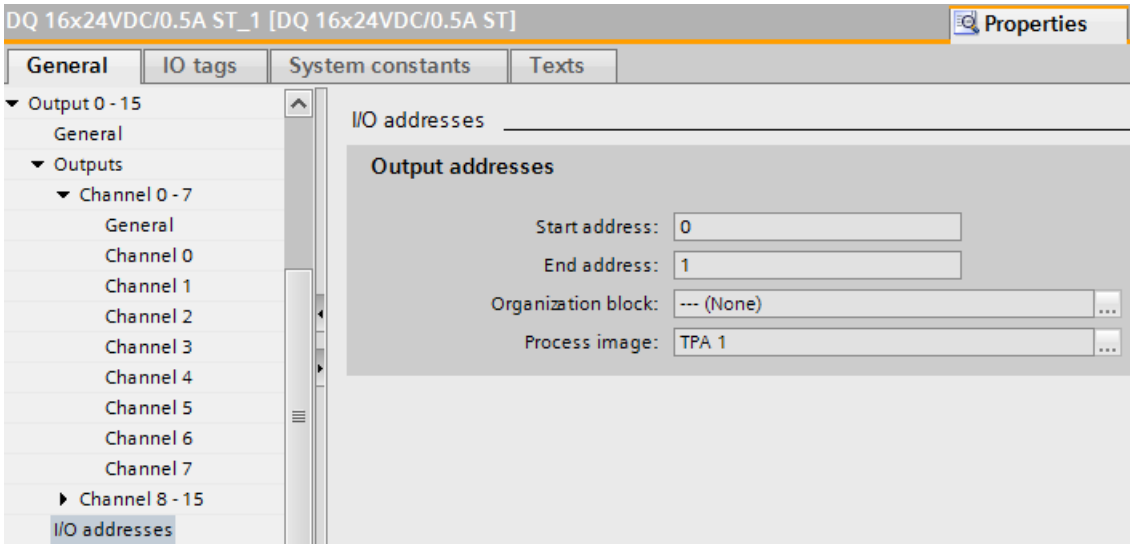

Uber den Eingangsparameter PART ("part") der Anweisung "UPDAT_PO" ist die Nummer des zu aktualisierenden Teilprozessabbilds hinterlegt. Beim Aufruf der Anweisung "UPDAT_PO" wird das angegebene Teilprozessabbild aktualisiert.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Die fehlerlose Verarbeitung setzt die Spule ("updateOk") auf "TRUE". Da kein Zugriffsfehler auftritt, zeigt die Fehleradresse ("errAddress") der Anweisung "UPDAT_PO" den Wert "16#0000".

SLI_gDB_UPDAT_PO

	Name	Data type	Start value	Monitor value
1	Static			
2	part	PIP	1	1
3	returnValue	Int	0	0
4	errAddress	Word	16#0	16#0000
5	updateOk	Bool	false	TRUE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SYNC_PI: Prozessabbild der Eingänge synchronisieren

Beschreibung

Mit der Anweisung "SYNC_PI" aktualisieren Sie ein Teilprozessabbild der Eingänge taktsynchron. Ein an einen DP-Takt bzw. PN-Sendetakt angebundenes Anwenderprogramm aktualisiert mit dieser Anweisung die erfassten Eingangsdaten in einem Teilprozessabbild der Eingänge taktsynchron und konsistent.

Aufruf

"SYNC_PI" ist unterbrechbar und nur in den OBs 61, 62, 63 und 64 aufzurufen.

Hinweis

Wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben, ist der Aufruf der Anweisung "SYNC_PI" nur in den OBs 61 bis 64 erlaubt. Ein Teilprozessabbild, das Sie mit "SYNC_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT_PI (Seite 2289)" aktualisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC_PI":

Parameter	Deklaration	Datentyp	Speicherbereich	Wertebereich	Beschreibung
PART	Input	PIP	E, A, M, D, L oder Konstante	1 bis 31	Nummer des Teilprozessabbildes der Eingänge, welches taktsynchron zu aktualisieren ist.
RET_VAL	Return	INT	E, A, M, D, L, P	-	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L, P		Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)*	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Submoduls sind jedoch konsistent übertragen worden.
8090	Ein unzulässiger Wert am Parameter PART oder eine Aktualisierung des angegebenen Teilprozessabbilds der Eingänge ist in diesem OB nicht erlaubt. Möglicherweise wurde das angegebene Teilprozessabbild noch nicht definiert. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.
8092	Das Teilprozessabbild wird systemseitig aktualisiert und darf nicht für den Parameter PART nicht verwendet.
8093	Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.
80A0	Bei der Aktualisierung wurde ein Zugriffsfehler erkannt. Betroffene Eingänge wurden auf "0" gesetzt.
80A1	Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert. Der DP- bzw. PN-Zyklus ist zu kurz, um ausreichend Zeit für die Bearbeitung der Anweisung zu gewährleisten. Sie müssen daher die Zeiten TDP (auch bekannt als T_DC), Ti und To vergrößern.
80A2	Zugriffsfehler mit Konsistenzwarnung Bei der Aktualisierung des angegebenen Teilprozessabbilds der Eingänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. • Die Daten der fehlerhaften Eingänge wurden nicht von der Peripherie gelesen. Im Teilprozessabbild der Eingänge werden die betroffenen Eingänge auf "0" gesetzt. • Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Eingangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.
80C1	Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Falls Sie die Anweisung "SYNC_PI" für Prozessabbildteile von DP-Normslaves einsetzen, sind auch Fehlercodes der Anweisung "DPRD_DAT" möglich. Dies ist der Fall, wenn Sie für die Konsistenzbereiche größer als 32 Bytes definiert haben.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen (Seite 2434)

SYNC_PO: Prozessabbild der Ausgänge synchronisieren

Beschreibung

Mit der Anweisung "SYNC_PO" aktualisieren Sie ein Teilprozessabbild der Ausgänge taktsynchron. Ein an einen DP-Takt bzw. an ein PN-Sendetakt angebundenes Anwenderprogramm kann mit dieser Anweisung die berechneten Ausgangsdaten eines Teilprozessabbildes der Ausgänge taktsynchron und konsistent in die Peripherie übertragen.

Aufruf

"SYNC_PO" ist unterbrechbar und nur in den OBs 61, 62, 63 und 64 auszurufen.

Hinweis

Wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben, ist der Aufruf der Anweisung "SYNC_PO" nur in den OBs 61 bis 64 erlaubt. Ein Teilprozessabbild, das Sie mit "SYNC_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT_PO (Seite 2292)" aktualisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC_PO":

Parameter	Deklaration	Datentyp	Speicherbereich	Wertebereich	Beschreibung
PART	Input	PIP	E, A, M, D, L oder Konstante	1 bis 31	Nummer des Teilprozessabbildes der Ausgänge, welches taktsynchron zu aktualisieren ist.
RET_VAL	Return	INT	E, A, M, D, P	-	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
FLADDR	Output	WORD	E, A, M, D, L, P	-	Adresse des ersten Fehler verursachenden Bytes.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)*	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Submoduls sind jedoch konsistent übertragen worden.
8090	Ein unzulässiger Wert am Parameter PART oder eine Aktualisierung des angegebenen Teilprozessabbildes der Ausgänge ist in diesem OB nicht erlaubt. Möglicherweise wurde das angegebene Teilprozessabbild noch nicht definiert. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.

Fehlercode (W#16#...)*	Erläuterung
8092	Das Teilprozessabbild wird systemseitig aktualisiert und für den Parameter PART nicht verwendet.
8093	Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.
80A0	Bei der Aktualisierung des angegebenen Teilprozessabbilds der Ausgänge wurde ein Zugriffsfehler erkannt. Fehlerhafte Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben diese Ausgänge unverändert.
80A1	Zugriffsfehler mit Konsistenzwarnung Bei der Aktualisierung des angegebenen Teilprozessabbilds der Ausgänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. • Die Daten der fehlerhaften Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben die betroffenen Ausgänge unverändert. • Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Ausgangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.
80A2	Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
80C1	Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Falls Sie die Anweisung "SYNC_PO" für Prozessabbildteile von DP-Normslaves einsetzen, sind auch Fehlercodes der Anweisung "DPWR_DAT" möglich. Dies ist der Fall, wenn Sie für die Konsistenzbereiche größer als 32 Bytes definiert haben.

Siehe auch

DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben (Seite 2436)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.4.4 Dezentrale Peripherie

Übersicht über die Typen von Datensätzen

Einleitung

Der Datensatz als Eigenschaft der Hardware entspricht der Zusammenfassung von Daten, welche bestimmte Eigenschaften der Hardware widerspiegeln. Jede Hardware kann eine Vielzahl von Datensätzen als Eigenschaften besitzen. Dabei lassen sich die Datensätze anhand ihres Nutzens und ihres Informationsumfangs unterscheiden.

Im Folgenden machen Sie sich mit den Typen von Datensätzen vertraut und erhalten einen Wegweiser für das Arbeiten mit Datensätzen durch die Zuordnung zu Modulen, Systemen und Anweisungen.

Die nachfolgenden Ausführungen gelten für die Automatisierungssysteme S7-1200 und S7-1500 sowie die Dezentrale Peripherie ET 200MP, ET 200SP.

Typen von Datensätzen

Es gibt die folgenden Typen von Datensätzen:

Typen		Nutzen
Parameterdatensätze	a) für Eingabemodule	Parameterdatensätze enthalten die Eigenschaften der Eingabe-/Ausgabefunktion des betreffenden Peripheriemoduls. Sie können mit dem Datensatz z. B. einen Drahtbruch auslesen oder die Eingangsverzögerung ändern. Parameterdatensätze sind lesbar und schreibbar.
	b) für Technologiemodule oder Module mit Nutzung der Technologiefunktion	
	c) für Ausgabemodule	
Steuerdatensätze	a) zum Konfigurieren eines Zentralen oder Dezentralen Aufbaus	Der Steuerdatensatz enthält die Hardwarekonfiguration des Systems bzw. Aufbaus und ermöglicht das Umkonfigurieren eines Moduls oder eines IO-Systems über das Anwenderprogramm. Der Steuerdatensatz ist lesbar und schreibbar.
	b) zum Umkonfigurieren eines IO-Systems	
I&M-Datensätze	a) für Zentralen Aufbau oder Dezentralen Aufbau über PROFINET IO	Die I&M-Datensätze enthalten die Identifikations- und Wartungsdaten eines Moduls (I&M = Identification & Maintenance). Die Daten unterscheiden sich dabei wie folgt: • Identifikationsdaten: – abhängig von Herstellerdaten – nur lesbar – Sie können damit z. B. die Seriennummer auslesen. • Wartungsdaten: – abhängig von der Anlage und Verwendung des Geräts – lesbar und schreibbar – Sie können damit z. B. den Einbauort des Moduls ändern.
	b) für Zentralen Aufbau oder Dezentralen Aufbau über PROFIBUS DP	

Typen		Nutzen
Diagnosedatensätze & Konfigurationsdatensätze	a) für Dezentralen Aufbau über PROFINET IO b) für Dezentralen Aufbau über PROFIBUS DP	Diagnosedatensätze und Konfigurationsdatensätze sind jeweils nach gleicher Struktur aufgebaut und variieren lediglich in Inhalt und Größe. Die Datensätze lassen sich wie folgt unterscheiden: • Diagnosedatensätze: – Die Diagnosedatensätze enthalten die Diagnose- und Überwachungsdaten eines Moduls. – Kanaldiagnosedatensätze werden generiert, wenn ein Kanal einen Fehler aufweist und/oder einen Alarm ausgelöst hat. Wenn kein Fehler vorliegt, wird ein Diagnosedatensatz mit der Länge 0 geliefert. – Struktur und Größe der herstellerspezifischen Diagnosedatensätze sind vom jeweiligen Hersteller abhängig. – Diagnosedatensätze sind nur lesbar. • Konfigurationsdatensätze: – Die Konfigurationsdatensätze enthalten die Daten der Ist-Konfiguration oder Soll-Konfiguration eines Moduls. – Konfigurationsdatensätze sind lesbar (Ist-Konfiguration, Soll-Konfiguration) und schreibbar (Soll-Konfiguration).

Hinweis

Die bei WinCC verwendeten Rezepturdatensätze sind keine Eigenschaften des Moduls und haben mit den hier genannten Datensätzen nichts zu tun.

Verwendung eines Datensatzes im Projekt

Bevor Sie einen Datensatz in Ihrem Projekt verwenden, lesen Sie bitte im passenden Handbuch wegen der Datensatzstruktur und etwaigen Vorraussetzungen nach. Auf das jeweils passende Handbuch verweist der jeweilige Abschnitt "Weitere informationen".

Parameterdatensätze

Hinweis

Kanäle auslesen

Die Datensätze 0..X und 64..64+X enthalten kanalspezifische Parameter. Dementsprechend enthält z. B. der Datensatz 0 die Informationen des Kanal 0.

Der Datensatz 128 enthält die Informationen aller Kanäle des betreffenden Moduls aus der Hardware-Konfiguration.

Die Anweisung RD_DPAR kann nur auf einer CPU S7-1500 oder einer Dezentralen Peripherie ET 200 ausgeführt werden.

Typ:	Parameterdatensätze (a)				
Betrifft das Modul:	Eingabemodul				
Ort:	Zentraler Aufbau (S7-1500) oder ET 200MP		S7-1200 oder ET 200SP (*)		
Anweisungen:	RDREC, WRREC	RD_DPAR	RDREC, WRREC		RD_DPAR (nicht für S7-1200)
Datensatz (DS):	DS 0..X (DS = Kanal; X = jeweils letzte Kanal des Eingabemoduls)	DS 128	DS 128	*Für ET 200SP Namur: DS 0..X (DS = Kanal; X = jeweils letzte Kanal des Eingabemoduls) oder DS 128	DS 128
HW-ID:	HW-ID des Eingabemoduls verwenden.				
Beispiel:	z. B. S7-1500 DI16x24VDC HF --> Kanäle 0..15; (Z. B. DS 13 verwenden). Siehe auch "Programmbeispiel zu Parameterdatensatz mit WRREC & RDREC verwenden (Seite 2328)".	z. B. S7-1500 DI16x24VDC HF --> DS 128. Siehe auch "Programmbeispiel zu RD_DPAR (Seite 2522)".	1. Beispiel: Eingabemodul einer CPU S7-1212C --> HW-ID der CPU, DS 128 und RDREC / WRREC verwenden. 2. Beispiel: ET 200SP DI8x24VDC HF --> HW-ID des DI-Moduls, DS 128 und RDREC / WRREC oder RD_DPAR verwenden. Siehe auch "Programmbeispiel zu RD_DPAR (Seite 2522)".		

Typ:	Parameterdatensätze (b)		
Betrifft das Modul:	Technologiemodul oder Modul mit Nutzung der Technologiefunktion		
Ort:	Zentraler Aufbau (S7-1500 oder S7-1200) oder Dezentraler Aufbau (ET 200MP oder ET 200SP)		
Anweisungen:	RDREC, WRREC		RD_DPAR (nicht für S7-1200)
Datensatz (DS):	DS 128	Alternativ bei DI-Modulen der S7-1500: DS 0..X (DS = Kanal; X = jeweils letzte Kanal des Eingabemoduls)	DS 128
HW-ID:	HW-ID des betreffenden Moduls / Peripheriemoduls verwenden.		
Beispiel:	1) Beispiel: CPU S7-1511 PN; High Speed Counter der S7-1500 DI16x24VDC HF --> z. B. DS 1; HW-ID des DI-Moduls (HSC) verwenden. WRREC / RDREC verwenden. 2) Beispiel: TM Count 2x24V --> DS 128; HW-ID des TM verwenden. WRREC / RDREC oder RD_DPAR verwenden. 3) Beispiel: High Speed Counter einer CPU S7-1200 --> DS 128; HW-ID der CPU und WRREC / RDREC verwenden.		

Typ:	Parameterdatensätze (c)	
Betrifft das Modul:	Ausgabemodul	
Ort:	Zentraler Aufbau (S7-1500) oder ET 200MP	S7-1200 oder ET 200SP (*)

Typ:	Parameterdatensätze (c)				
Anweisungen:	RDREC, WRREC	RD_DPAR	RDREC, WRREC		RD_DPAR (nicht für S7-1200)
Datensatz (DS):	DS 64..64 + X (DS 64 = Kanal 0; X = jeweils letzte Kanal des Ausgabemoduls)	DS 128	DS 128	*Für ET 200SP Namur: DS 64..64 + X (DS 64 = Kanal 0; X = jeweils letzte Kanal des Ausgabemoduls) oder DS 128	DS 128
HW-ID:	HW-ID des Ausgabemoduls verwenden.				
Beispiel:	z. B. DQ16x24VDC/0.5A ST --> Kanäle 0..15; (Z. B. für Kanal 13 den DS 77 verwenden).	z. B. DQ16x24VDC/ 0.5A ST --> DS 128.	z. B. ET 200SP DQ8x24VDC/0.5A ST --> DS 128.		

Hinweis

Weitere Datensätze

Gegebenenfalls gibt es noch weitere Datensätze. Z. B. Für die Eingabemodule:

- Bei der ET 200SP AI RTD gibt es den DS 237 zum Deaktivieren der Drahtbruchprüfung.
- Bei der ET 200SP AI Energy Meter ST gibt es den DS 142 zum Lesen der Messdaten.

Hinweis

Interfacemodul mit PROFIBUS DP

Wenn Sie ein Interfacemodul mit PROFIBUS DP verwenden, können Sie von dem dahinter liegenden Ein-/Ausgabemodul die Parameterdatensätze "0" und "1" nicht auslesen. Statt dessen lesen Sie darüber den Inhalt der Diagnosedatensätze "0" und "1" aus.

Zum Beispiel ist bei Ausgabemodulen der "Parameterdatensatz 0" der "DS 64" und der "Parameterdatensatz 1" der "DS 65".

Weitere Informationen

Weitere Informationen zum jeweiligen Parameterdatensatz finden Sie im Gerätehandbuch des betreffenden Peripheriemoduls (z. B. S7-1500 DI16x24VDC HF (https://support.industry.siemens.com/cs/de/de/view/59193001)).

Steuerdatensätze

Typ:	Steuerdatensätze			
Betrifft das Modul:	alle zu verwendenden Module des Aufbaus			IO-System und gewünschte IO-Devices
Ort:	Zentraler Aufbau (S7-1200, S7-1500)	Dezentraler Aufbau (z. B. ET 200MP, ET 200SP)		PROFINET IO
Anweisungen:	WRREC, RDREC	WRREC, RDREC	RDREC	ReconfigIOSystem

Typ:	Steuerdatensätze			
Datensatz (DS):	DS 196	DS 196	DS 197 (Rückmeldedatensatz)	DS "CTRLREC" (Parameter von ReconfigIOSystem)
HW-ID:	HW-ID 33 verwenden.	HW-ID des Kopfes [Head] der IM der Dezentralen Peripherie verwenden.		HW-ID der PROFINET-Schnittstelle (IO-Controller) verwenden. Entsprechend einzurichtender Verbindungen und IO-Devices die zugehörigen HW-IDs über CTRLREC angeben.
Beispiel:	z. B. Für CPU S7-1511-1 PN mit DI16x24VDC HF. Siehe auch "Programmbeispiel zu Steuerdatensatz mit WRREC & RDREC verwenden (Seite 2338)".	z. B. DI16x24VDC HF, ET 200MP IM 155-5 PN ST --> von "<IO-device_V2.0[Head]>" die ID verwenden. Der reale Aufbau enthält eine weitere DI16x24VDC HF, welche bei der Konfiguration nicht berücksichtig werden soll.		z. B. Eine CPU S7-1500 (IO-Controller) mit einer ET 200MP (IO-Device) und einer ET 200SP (als optionales IO-Device) verbinden und aktivieren. Siehe auch "Programmbeispiel für ReconfigIOSystem (Seite 2416)".

Weitere Informationen

Weitere Informationen zum Steuerdatensatz 196 finden Sie im Systemhandbuch des betreffenden Automatisierungssystems (z. B. "Automatisierungssystem S7-1500 (https://support.industry.siemens.com/cs/ww/de/view/59191792)").

Weitere Informationen zum Steuerdatensatz 197 finden Sie im Gerätehandbuch des betreffenden Interfacemoduls (z. B. "ET 200MP IM 155-5 PN ST (https://support.industry.siemens.com/cs/ww/de/view/59193106)").

Weitere Informationen zum Steuerdatensatz "CTRLREC" finden Sie bei der Anweisung "ReconfigIOSystem: IO-System umkonfigurieren (Seite 2411)".

I&M-Datensätze

Berücksichtigen Sie die folgende Verwendung der I&M-Datensätze:

- Der Datensatz "I&M0" ("16#AFF0") dient zum Übermitteln allgemeiner Informationen über das Modul oder Gerät.
- Der Datensatz "I&M1" ("16#AFF1") dient zum Übermitteln von Funktion und Einbauort des Moduls oder Geräts.
- Der Datensatz "I&M2" ("16#AFF2") dient zum Übermitteln des Installationsdatums des Moduls oder Geräts.
- Der Datensatz "I&M3" ("16#AFF3") dient zum Übermitteln von individuellen Zusatzinformationen über das Modul oder Gerät.

Hinweis

Der Datensatz "I&M0" ("16#AFF0") wird von jedem Modul unterstützt. Alle anderen I&M-Datensätze sind optional.

Typ:	I&M-Datensätze				
Betrifft das Modul:	Gerät / Modul				
Ort:	Zentraler Aufbau oder Dezentraler Aufbau über PROFINET IO oder PROFIBUS DP				
Anweisungen:	RDREC, WRREC				
Datensatz (DS):	DS 16#AFF0	DS 16#AFF1	DS 16#AFF2	DS 16#AFF3	Alternativ bei Dezentralem Aufbau über PROFIBUS DP: DS 255
HW-ID:	HW-ID des Moduls verwenden.				
Beispiel:	1) Beispiel: Für CPU S7-1511-1 PN --> DS 16#AFF0; HW-ID der CPU verwenden. Siehe auch "Programmbeispiel zu I&M-Datensatz mit RDREC lesen (Seite 2309)".				2) Beispiel: Für ET 200SP mit IM 155-6 DP ST --> DS 255; HW-ID des IM verwenden.

Weitere Informationen

Weitere Informationen zum jeweiligen I&M-Datensatz finden Sie:

- im Systemhandbuch des betreffenden Automatisierungssystems (z. B. "Automatisierungssystem S7-1500 (https://support.industry.siemens.com/cs/ww/de/view/59191792)").
- im Programmierhandbuch "SIMATIC NET PROFINET IO-Base-Anwenderprogrammierschnittstelle (https://support.industry.siemens.com/cs/ww/de/view/26435491)".

Diagnosedatensätze & Konfigurationsdatensätze

Typ:	Diagnosedatensätze & Konfigurationsdatensätze			
Betrifft das Modul:	Gerät / Modul oder Submodul im IO-System			
Ort:	Dezentraler Aufbau (PROFINET IO, PROFIBUS DP)			
Anweisungen:	RDREC, WRREC	RDREC		
Datensatz (DS):	Konfiguration (Ist / Soll):	Kanaldiagnose:	Herstellerspezifische Diagnose über Interfacemodul:	
	DS für das betreffende Modul verwenden. Siehe dazu die Adressierungsebene & DS zu "PROFINET IO (https://support.industry.siemens.com/cs/ww/de/view/19289930)".		für PROFINET IO: Siehe Interfacemodul	für PROFIBUS DP: DS 0, 1

Typ:	Diagnosedatensätze & Konfigurationsdatensätze		
HW-ID:	HW-ID des betreffenden Moduls (HW-ID mit Datentyp "Hw_SubModule") verwenden.		HW-ID des betreffenden Moduls (HW-ID mit Datentyp "Hw_SubModule") verwenden.
Beispiel:	z. B. IO-Controller S7-1513-1 PN; HW-ID des Moduls auf Steckplatzebene für das IO-Device (ET 200MP IM 155-5 PN HF) verwenden. Mit DS 16#C001 die Ist-Konfiguration auslesen.	z. B. IO-Device (ET 200MP IM 155-5 PN HF) mit DQ-Modul erzeugt Drahtbruch. HW-ID des Kopfmoduls des IO-Device verwenden. IO-Controller ist S7-1513-1 PN. DS 16#E00A zur Kanaldiagnose verwenden. Siehe auch "Programmbeispiel zu Diagnosedatensatz mit RDREC lesen (Seite 2313)".	z. B. PROFIBUS DP: ET 200MP mit IM 155-5 DP ST; DI-Modul lesen --> HW-ID (Datentyp "Hw_SubModule") des DI-Moduls und den DS 1 zur Diagnose verwenden.

Weitere Informationen

Die Adressierungsebene, den Aufbau und wichtige Datensätze zu Diagnose & Konfiguration bei PROFINET IO finden Sie:

- im Programmierhandbuch "Von PROFIBUS DP nach PROFINET IO (https://support.industry.siemens.com/cs/ww/de/view/19289930)".

Weitere und allgemeine Informationen zu Kanaldiagnosedatensätzen finden Sie:

- in der Online-Hilfe von STEP 7 unter "Diagnose von PROFINET-Geräten".
- im Systemhandbuch zu PROFINET (z. B. "PROFINET mit STEP 7 ab V14 (https://support.industry.siemens.com/cs/ww/de/view/49948856)").

Weitere Informationen zu herstellerspezifischen Diagnosedatensätzen finden Sie:

- im Gerätehandbuch des betreffenden Interfacemoduls (z. B. ET 200MP IM 155-5 DP ST (https://support.industry.siemens.com/cs/ww/de/view/78324181)).

Siehe auch

Sample Library for Instructions (Seite 615)

RDREC: Datensatz lesen (Seite 2306)

WRREC: Datensatz schreiben (Seite 2326)

FAQ-Seite der Library of PLC Datatypes (https://support.industry.siemens.com/cs/de/de/view/109482396)

RDREC: Datensatz lesen

RDREC: Datensatz lesen

Beschreibung

Mit der Anweisung "RDREC" lesen Sie den Datensatz mit der Nummer INDEX von der mittels ID adressierten Baugruppe. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Baugruppe (PROFIBUS DP oder PROFINET IO) handeln.

- Über den Parameter ID wählen Sie aus, von welcher Baugruppe (DP/PROFINET IO) ein Datensatz gelesen werden soll. Verwenden Sie für den Parameter ID die Hardware-Kennung (HW-ID) der Baugruppe. Die Hardware-Kennung der Baugruppe finden Sie unter "PLC-Variablen > Systemkonstanten". Z. B. Für eine S7-1500 DI-Baugruppe: Diese wird als <Name der DI-Baugruppe> mit dem Datentyp "Hw_SubModule" angezeigt.
- Über den Parameter INDEX wählen Sie aus, welcher Datensatz von der Baugruppe gelesen wird. Welche Datensätze mit welcher Datensatznummer gelesen werden können, ist abhängig von der Baugruppe. Weitere Informationen hierzu finden Sie in der Dokumentation zur jeweiligen Baugruppe.
- Die über "RDREC" lesbaren Datensätze haben eine unterschiedliche Länge. Mit MLEN geben Sie vor, wie viele Bytes des Datensatzes Sie maximal lesen möchten. Wenn Sie am Parameter MLEN als Länge "0" wählen, wird der vollständige Datensatz an dem Parameter RECORD geschrieben.
- Wählen Sie den Zielbereich RECORD mindestens MLEN Bytes lang. Wenn Sie mit MLEN=0 den vollständigen Datensatz auslesen, verwenden Sie für RECORD die maximale Länge des Datensatzes. Die Struktur (Aufbau, Datentypen und Länge), die Sie am Parameter RECORD verwenden ist ebenfalls abhängig davon, welchen Datensatz Sie von welcher Baugruppe auslesen.
- Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.
- Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Hinweis

Die Schnittstelle der Anweisung "RDREC" ist identisch mit der in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "RDREC".

Arbeitsweise

"RDREC" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "RDREC" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat, ist die Datensatzübertragung abgeschlossen.

Aufruf von RDREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems

Wenn Sie die Anweisung RDREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems aufrufen, werden stets die Fehlercodes 0x7001 und 0x7002 zurückgeliefert. Auch im Fehlerfall, wenn z. B. die Datensatzquelle nicht vorhanden ist, wird kein anderer Fehlercode geliefert. In diesem Fall wird der korrekte Fehlercode erst nach Beenden des SYNCUP geliefert.

Wenn Sie die Anweisung RDREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems aufrufen, wird kein Datensatzauftrag angestoßen. Der Anstoß des Datensatzauftrags efolgt erst, wenn Sie die Anweisung nach Beendigung des SYNCUP erneut aufrufen. Es wird daher empfohlen, RDREC stets zyklisch aufzurufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RDREC":

Parameter	Deklaration	Datentyp*	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T**, Z** oder Konstante	REQ = 1: Datensatzübertragung durchführen
ID	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Hardware-Baugruppe (DP/PROFINET IO) Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Baugruppe oder des Interface in der Hardware-Konfiguration hinterlegt.
INDEX	Input	DINT	E, A, M, D, L oder Konstante	Datensatznummer
MLEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Länge der zu lesenden Datensatz-information in Byte.
VALID	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde empfangen und ist gültig.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht be-endet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Lesevorgang trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus bzw. Fehlerinformation.
LEN	Output	UINT	E, A, M, D, L	Länge der gelesenen Datensatzinformation.

Parameter	Deklaration	Datentyp*	Speicherbereich	Beschreibung
RECORD	InOut	VARIANT	E, A, M, D, L	Zielbereich für den gelesenen Datensatz. Falls Sie für den Zielbereich den Systemdatentyp NREF in einem Datenbaustein mit optimiertem Zugriff verwenden, werden in den Zielbereich keine Werte geschrieben.
* Bei AWL erfolgt keine implizierte Konvertierung, weshalb der Umfang der zulässigen Datentypen eingeschränkt sein kann. Beachten Sie bei der Programmierung in AWL die jeweils zulässigen Datentypen im Tooltipp des Parameters ** Nur bei S7-1500.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie "RDREC" für das Lesen eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX, MLEN und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2370).

Sonderfall für PC-Systeme: Zustand von Windows ermitteln

Folgendes Szenario hilft Ihnen, Informationen über den Zustand von Windows ihres PC-Systems zu erhalten:

- Rufen Sie die Anweisung "RDREC" (SFB52) über einen zyklischen OB, z. B. OB1 auf

Um über die Anweisung "RDREC" den Windows-Status zu ermitteln, ist ein Data Record index (0x7FFE) für den Software-Controller definiert.

Mit folgenden Eingangsparametern wird die Anweisung "RDREC" aufgerufen:

Parameter	Deklaration	Datentyp	Speicherbereich	Wert/Beschreibung
(HIC) ID	Input	HW_IO	E, A, M, D, L oder Konstante	59 (Host Interface RID)
INDEX	Input	DINT	E, A, M, D, L oder Konstante	0x7FFE
MLEN	Input	UINT	E, A, M, D, L oder Konstante	9
RECORD	InOut	VARIANT	E, A, M, D, L	Adresse zu jedem Bereich von 9 Byte

Der Data Record zum Windows-Status wird über das achte Byte ausgegeben und kann folgenden Inhalt haben:

- 0x01: Windows läuft
- 0x02: Windows läuft nicht
- 0x03: Bluescreen

Hinweis

Bei einem Systemabsturz reagiert Windows nicht mehr und zeigt einen Bluescreen an, der Angaben über den Grund und den Zeitpunkt des Systemabsturzes liefert. Standardmäßig bleibt der Bluescreen dauerhaft sichtbar.

Nach dem Systemfehler wird das System gestoppt und es können folgende Szenarien eintreten:

- Windows sendet keine Signale mehr an die konfigurierte Software-CPU
- Kein Aufruf des OB82
- Kein Erstellen eines Diagnoseeintrags

Beispiele

Ein Beispiel finden Sie hier: Programmbeispiel zu I&M-Datensatz mit RDREC lesen (Seite 2309).

Ein weiteres Beispiel finden Sie hier: Programmbeispiel zu Diagnosedatensatz mit RDREC lesen (Seite 2313).

Weitere Informationen und den Programmcode zu den Beispielen finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Grundlagen zum Bausteinzugriff (Seite 55)

Übersicht über die Typen von Datensätzen (Seite 2298)

Programmbeispiel zu I&M-Datensatz mit RDREC lesen

Einleitung

In dem folgenden Beispiel lesen Sie den I&M0-Datensatz einer CPU S7-1500 aus.

Weitere Informationen zu Datensätzen finden Sie in der Übersicht über die Typen von Datensätzen (Seite 2298).

Voraussetzung

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein neun Variablen sowie eine Struktur an.

Die Hardware-Kennung der CPU lesen Sie unter "PLC-Variablen > Systemkonstanten" aus. Die CPU sollte dort als <Local> mit dem Datentyp "Hw_SubModule" angezeigt werden. Die Hardware-Kennung hinterlegen Sie über den Eingangsparameter ID ("id").

SLI_gDB_RDREC_I-M

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	id	HW_IO	49
4	dataRecNbr	DInt	16#AFF0
5	maxLen	UInt	0
6	checkedValid	Bool	false
7	busy	Bool	false
8	error	Bool	false
9	status	DWord	16#0
10	len	UInt	0
11	recordIM0	Struct	
12	BlockType	Word	16#0
13	BlockLength	UInt	0
14	BlockVersion_HL	Word	16#0
15	VendorID_HL	Word	16#0
16	OrderID	Array[0..19] of Byte	
17	IM_SerialNbr	Array[0..15] of Byte	
18	IM_HWrev	UInt	0
19	IM_SWrev	Struct	
20	IM_RevCounter	UInt	0
21	IM_ProfileID	UInt	0
22	IM_ProfileSpecType	Word	16#0
23	IM_Version	Word	16#0
24	IM_Supported	Word	16#0

In einem FB legen Sie die folgende lokale Variable an.

memErrStatus	DWord	16#0	Set in IDB

Parameter verschalten

Netzwerk 1: Die Parameter der Anweisung "RDREC" verschalten Sie wie folgt.

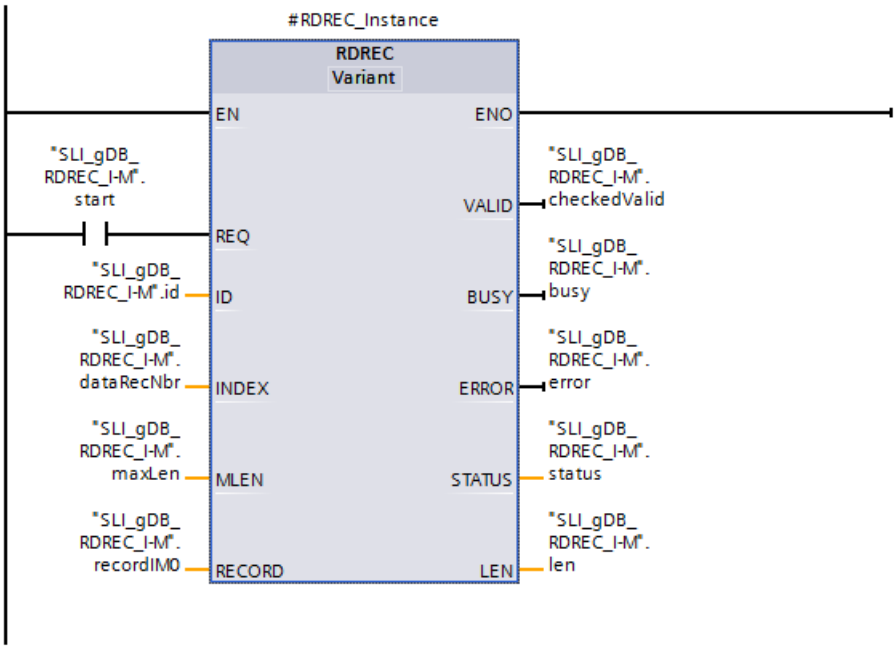

Netzwerk 2: Zum Speichern des Status im Fehlerfall verschalten Sie die Variable "status" wie folgt.

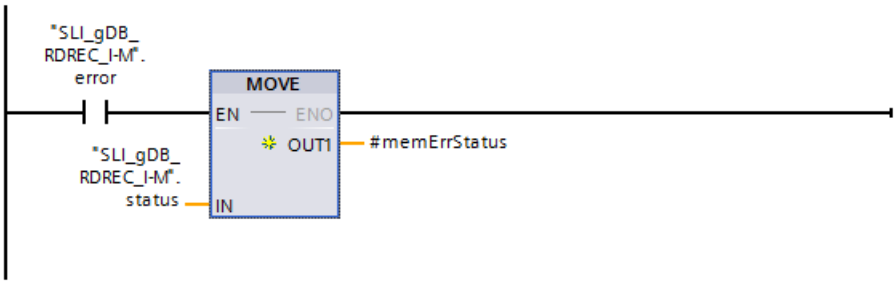

Ergebnis

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "RDREC" gestartet. Über den Eingangsparameter ID ("id") ruft die Anweisung "RDREC" die Baugruppe auf. Durch den Eingangsparameter INDEX ("dataRecNbr") wird der I&M0-Datensatz der CPU aufgerufen.

Über mehrere Aufrufe hinweg liesst die Anweisung "RDREC" den I&M0-Datensatz aus und speichert den Datensatz über den Parameter RECORD ("recordIM0") ab.

Der I&M0-Datensatz enthält u. a. folgende Informationen:

- den Herstellernamen ("VendorID_HL")
- die Artikel- ("OrderID") & Gerätenummer ("IM_SerialNbr")

- Hardware- ("IM_HWrev") & Software-Informationen ("IMSWrev")
- IM-Version ("IM_Version") und die Bandbreite unterstützter I&M-Datensätze ("IM_Supported")

recordIM0	Struct		
BlockType	Word	16#0	16#0020
BlockLength	UInt	0	56
BlockVersion_HL	Word	16#0	16#0100
VendorID_HL	Word	16#0	16#002A
OrderID	Array[0..19] of Byte		
IM_SerialNbr	Array[0..15] of Byte		
IM_HWrev	UInt	0	3
IM_SWrev	Struct		
Refix	Byte	16#0	16#56
FunctEnh	Byte	16#0	16#01
BugFix	Byte	16#0	16#05
InternalChange	Byte	16#0	16#00
IM_RevCounter	UInt	0	0
IM_ProfileID	UInt	0	0
IM_ProfileSpecType	Word	16#0	16#0000
IM_Version	Word	16#0	16#0101
IM_Supported	Word	16#0	16#001E

Die Länge des gelesenen Datensatzes wird über den Ausgangsparameter LEN ("len") festgehalten. Entsprechend des Werts des Eingangsparameters MLEN ("maxLen" hat den Wert "0") ist die zu lesende Datensatzinformation unbeschränkt.

Während der Verarbeitung ist der Ausgangsparameter BUSY ("busy") auf "TRUE" und VALID ("checkedValid") auf "FALSE" gesetzt. Ein Erfolg beim Lesen ("checkedValid" auf "TRUE") wird erst zum Ende der Verarbeitung ausgegeben. Am Ausgangsparameter STATUS ("status") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

SLI_gDB_RDREC_I-M

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	id	HW_IO	49	16#0031
4	dataRecNbr	DInt	16#AFF0	45040
5	maxLen	UInt	0	0
6	checkedValid	Bool	false	TRUE
7	busy	Bool	false	FALSE
8	error	Bool	false	FALSE
9	status	DWord	16#0	16#0000_0000
10	len	UInt	0	60
11	recordIM0	Struct		

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

RDREC: Datensatz lesen (Seite 2306)

Programmbeispiel zu Diagnosedatensatz mit RDREC lesen

Einleitung

In dem folgenden Beispiel lesen Sie über RDREC den Diagnosedatensatz "16#E00A" eines IO-Device aus. Bei Auslösen eines Drahtbruchs werden die Fehlerinformationen als Diagnosedatensatz ausgegeben.

Weitere Informationen zu Datensätzen finden Sie in der Übersicht über die Typen von Datensätzen (Seite 2298).

Hinweis

Anzahl der Diagnosedatensätze

Jeder Steckplatz eines Fehler auslösenden Moduls generiert seinen eigenen Datensatz für das IO-Device. Die Diagnose kann also mehrere Datensätze enthalten.

Das Beispiel ist dazu gedacht, mehrere Hardware-Fehler (Kanalfehler) mehrerer Module auszulesen und in die passende Struktur eines Speicher-Arrays zu kopieren. Die Speicher-Arrays ("recordA", "recordB") können nach vorgegebener Einstellung maximal sechs Datensätze mit jeweils maximal 16 Kanälen aufnehmen.

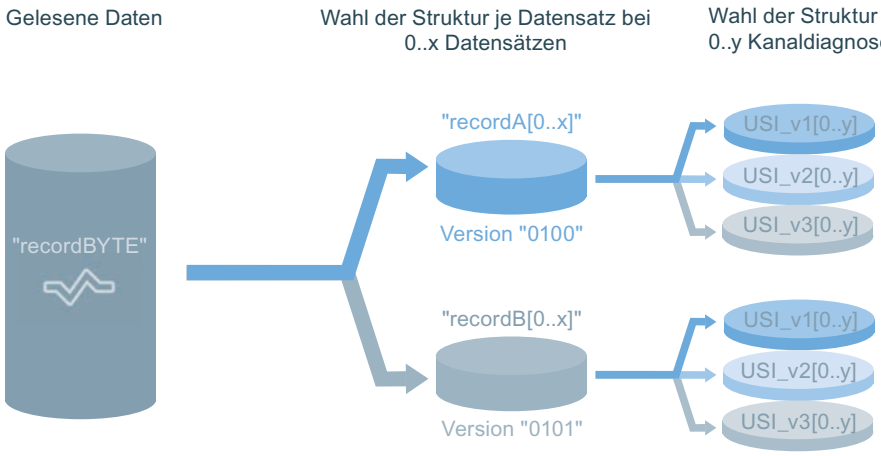

Voraussetzungen

Hardware installiert

Folgende Hardware wird benötigt:

- Ein IO-Controller (z. B. S7-1513-1 PN)
- Ein IO-Device (z. B. ET 200MP IM 155-5 PN HF)
- Ein oder mehrere digitale Ausgabemodule des Typs HF (z. B. DQ 4x24VDC HF)

Hinweis

Drahtbruch auslösen

Stellen Sie sicher, dass die Kanäle des DQ-Moduls, welche den Drahtbruch auslösen sollen, nicht verdrahtet sind.

Hinweis

Vorgehen zur Konfiguration der Hardware

Konfigurieren Sie die Hardware wie folgt:

1. Verbinden Sie die CPU und das IO-Device über PROFINET und weisen Sie die CPU als IO-Controller dem IO-Device zu.
2. Öffnen Sie für das jeweilige DQ-Modul den Pfad "Eigenschaften > Ausgang 0-x > Ausgänge > Kanal 0-y".
3. Wählen jeweils den zu konfigurierenden Kanal und aktivieren Sie im Bereich "Diagnose" die Option "Drahtbruch".

PLC-Datentypen erstellt

Sie benötigen die folgenden PLC-Datentypen für die Weitergabe und Ablage der Datensätze:

- für den Body-Inhalt der Struktur "A"/"0100":

Rd_0100body

	Name	Data type	Default value
1	BlockType	Word	16#0
2	BlockLength	UInt	0
3	BlockVersion_HL	Word	16#0
4	SlotNumber	UInt	0
5	SubSlotNumber	UInt	0
6	ChannelLocation	Word	16#0
7	ChannelProperties	Array[0..15] of Bool	
8	UserStructureIdentifier	Word	16#0

- für den Body-Inhalt der Struktur "B"/"0101":

Rd_0101body

	Name	Data type	Default value
1	BlockType	Word	16#0
2	BlockLength	UInt	0
3	BlockVersion_HL	Word	16#0
4	API	DWord	16#0
5	SlotNumber	UInt	0
6	SubSlotNumber	UInt	0
7	ChannelLocation	Word	16#0
8	ChannelProperties	Array[0..15] of Bool	
9	UserStructureIdentifier	Word	16#0

- für die Struktur des Typs 1 entsprechend dem Wert von "UserStructureIdentifier":
  Der Typ 1 der Struktur wird für die einfache Kanaldiagnose verwendet.

USI_v1

	Name	Data type	Default value
1	ChannelNumber	UInt	0
2	ChannelProperties	Array[0..15] of Bool	
3	ChannelErrorType	Word	16#0

- für die Struktur des Typs 2 entsprechend dem Wert von "UserStructureIdentifier":
  Der Typ 2 der Struktur wird für die erweiterte Kanaldiagnose verwendet.

USI_v2

	Name	Data type	Default value
1	ChannelNumber	UInt	0
2	ChannelProperties	Array[0..15] of Bool	
3	ChannelErrorType	Word	16#0
4	ExtChErrorType	Word	16#0
5	ExtChAddValue	DWord	16#0

- für die Struktur des Typs 3 entsprechend dem Wert von "UserStructureIdentifier":
  Der Typ 3 der Struktur wird für eine herstellerspezifische Kanaldiagnose verwendet. Der Aufbau der Struktur wird in diesem Beispiel nicht definiert. Für dieses Beispiel wird für den Typ 3 von einer Länge von 28 Byte ausgegangen.

USI_v3

	Name	Data type	Default value
1	ChannelDiagData3	Array[0..27] of Byte	

- für die Zusammenfassung der Struktur "A"/"0100":

RECdiag_0100

	Name	Data type
1	Body0100	"Rd_0100body"
2	USI_DetailsType1	Array[0..15] of "USI_v1"
3	USI_DetailsType2	Array[0..15] of "USI_v2"
4	USI_DetailsType3	Array[0..15] of "USI_v3"

- für die Zusammenfassung der Struktur "B"/"0101":

RECdiag_0101

	Name	Data type
1	Body0101	"Rd_0101body"
2	USI_DetailsType1	Array[0..15] of "USI_v1"
3	USI_DetailsType2	Array[0..15] of "USI_v2"
4	USI_DetailsType3	Array[0..15] of "USI_v3"

Hinweis

Größe der Arrays

Wenn Sie die Array-Größen (für Lesespeicher, Datensatz oder Kanäle) ändern wollen, müssen Sie Folgendes tun:

- Für Kanäle: Passen Sie die Größe der Arrays ("Array[0..y] of USI_v..") in den PLC-Datentypen ("RECdiag_0100", "RECdiag_0101") an. Passen Sie den Wert von "maxChannelPos" im gDB an.
- Für Datensätze: Passen Sie die Größe der Arrays ("recordA", "recordB") im gDB an. Passen Sie den Wert von "maxRecordPos" im gDB an.
- Für Lesespeicher ("recordBYTE"): Passen Sie die Größe des Arrays ("recordBYTE") und den Wert von "maxPosition" im gDB an. Beachten Sie, dass der Diagnosedatensatz "16#E00A" insgesamt maximal 4176 Byte lang sein darf.

Überwachungstabelle angelegt

Zum Auslösen eines Drahtbruchs verwenden Sie die Funktion "Forcen" einer Überwachungstabelle.

Erzwingen Sie z. B. für den Ausgang "%QW0" den Wert "16#FFFF", lösen alle den Ausgang verwendenden Kanäle mit konfiguriertem Drahtbruch einen Hardware-Fehler aus.

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein (gDB) folgende Variablen und Strukturen an.

SLI_gDB_RDREC_Diag			
	Name	Data type	Start value
1	Static		
2	id	HW_IO	262
3	dataRecNbr	DInt	16#E00A
4	startRead	Bool	false
5	maxLength	UInt	0
6	maxPosition	UInt	4175
7	recordBYTE	Array[0..4175] of Byte	
8	readLength	UInt	0
9	checkedValid	Bool	false
10	busy	Bool	false
11	error	Bool	false
12	memErrStatus	DWord	16#0
13	countCalls	UInt	0
14	maxChannelPos	UInt	15
15	maxRecordPos	UInt	5
16	recordA	Array[0..5] of "RECdiag_0100"	
17	recordB	Array[0..5] of "RECdiag_0101"	
18	data_sorting_done	Bool	false
19	outputOverflow	Struct	
20	Too_many_bytes	Bool	false
21	Too_many_records	Bool	false
22	Too_many_channels	Bool	false
23	different_structs_used	Bool	false
24	countRecs	UInt	0
25	chosenStruct	UInt	0
26	posForRec	UInt	0
27	posForChannel	UInt	0
28	resetOps	Struct	
29	reset	Bool	false

Die Hardware-Kennung des Kopfs des IO-Device lesen Sie aus und hinterlegen die Hardware-Kennung über die Variable "id". Die HW-Kennung für das Kopfmodul finden Sie über "PLC-Variablen > Systemkonstanten" und über einen Eintrag wie <Head~IO_Device>. Der verwendete Datentyp ist "HW_SubModule".

FC "SLI_FC_start_RDiag": Parameter verschalten

Zum Starten der Anweisung "RDREC", basierend auf einem Hardware-Fehler, erstellen Sie den FC "SLI_FC_start_RDiag".

Im FC legen Sie die folgende Verschaltung an.

Den FC "SLI_FC_start_RDiag" rufen Sie in einem Diagnosealarm-OB (OB82) auf.

FC "SLI_FC_reset_RDiag": Parameter verschalten

Zum Rücksetzen von Prozesswerten nach einem Fehler erstellen Sie den FC "SLI_FC_reset_RDiag".

Im FC legen Sie die folgende lokale Variable an.

	Temp	
	tempReset	Bool

Im FC legen Sie die folgende Verschaltungen an.

Netzwerk 1: Zum Rücksetzen der Prozesswerte legen Sie folgende Verschaltungen an. Teil 1:

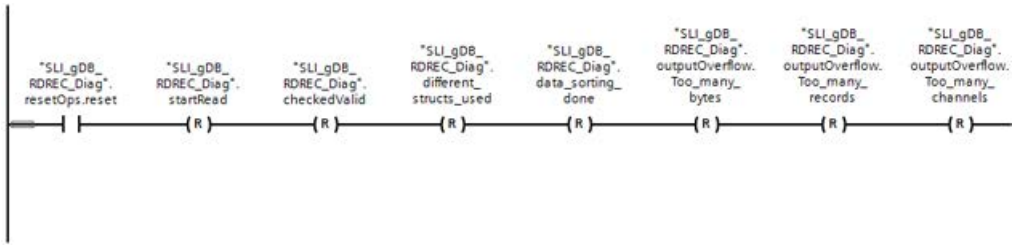

Teil 2:

Teil 3:

Teil 4:

Netzwerk 2: Zum Rücksetzen der Variable "reset" legen Sie folgende Verschaltungen an.

FB "SLI_FB_chooseStruct_RDiag": Parameter verschalten

Den FB "SLI_FB_chooseStruct_RDiag" erstellen Sie zur Auswahl der passenden Struktur für den Datensatz und zum Kopieren der Daten auf die passende Stelle im Array.

Weitere Informationen finden Sie im Abschnitt "Ergebnis" ("Netzwerk 6") oder im Kommentarbereich der Netzwerke (siehe "Programmcode").

FB "SLI_FB_RDREC_Diag": Parameter verschalten

Den FB "SLI_FC_start_RDiag" erstellen Sie zum Aufrufen und Verarbeiten der Anweisung "RDREC" sowie aller nachfolgenden Prozesse.

Im FB legen Sie folgende lokale Variablen an.

Static			
statusExe	DWord	16#0	Nicht re...
checkExe	Bool	false	Nicht rema...
lengthOut	UInt	0	Nicht rema...
RDREC_Instance00	RDREC		

Im FB legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Sie rufen den FC "SLI_FC_reset_RDiag" auf und legen die Anweisung "Label" mit dem Namen "RDREC" an.

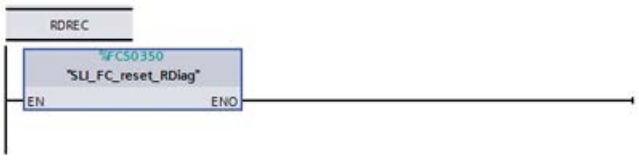

Netzwerk 2: Die Parameter der Anweisung "RDREC" verschalten Sie wie folgt.

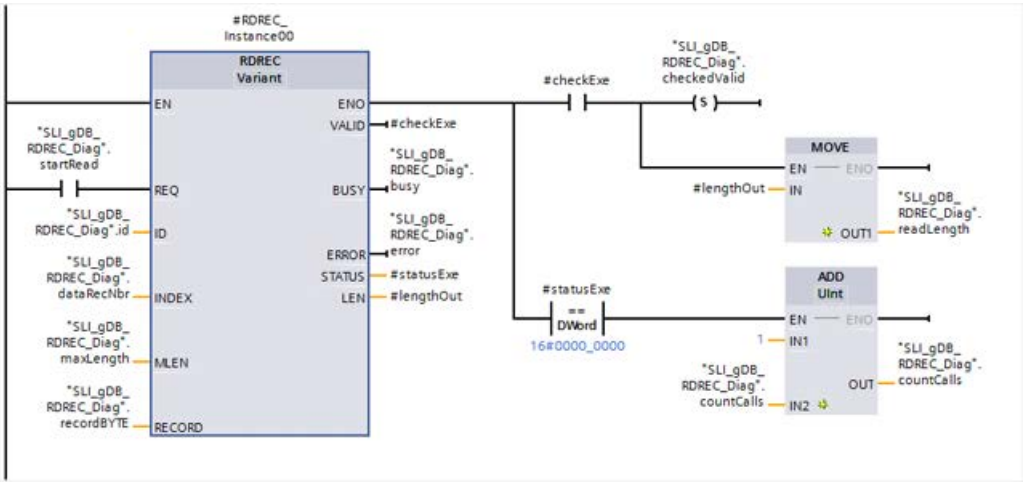

Netzwerk 3: Zum Speichern des Status im Fehlerfall von RDREC verschalten Sie den Parameter wie folgt.

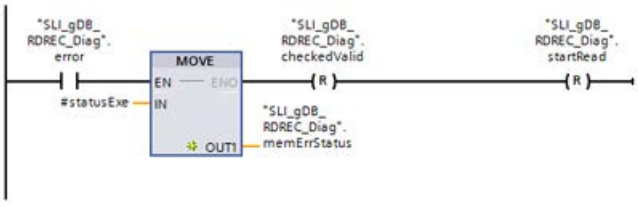

Netzwerk 4: Zum Beenden der Verarbeitung von RDREC legen Sie die folgenden Verschaltungen an.

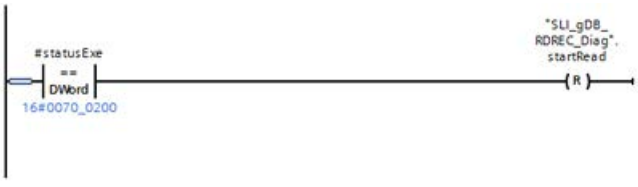

Netzwerk 5: Zum Zurückspringen nach Netzwerk 1 legen Sie die folgenden Verschaltungen an.

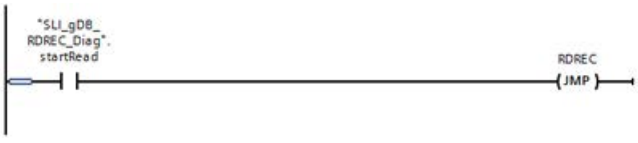

Netzwerk 6: Sie rufen den FB "SLI_FB_chooseStruct_RDiag" wie folgt auf.

Den FB "SLI_FB_RDREC_Diag" rufen Sie in einem Zyklus-OB (OB1) auf.

Ergebnis - FB "SLI_FB_RDREC_Diag"

Netzwerk 1:

Wenn der Schließerkontakt ("reset") den Signalzustand "TRUE" liefert, werden die Prozesswerte des FBs "SLI_FB_chooseStruct_RDiag" und die Prozesswerte des globalen Datenbausteins rückgesetzt.

Netzwerk 2:

Wenn der Eingangsparameter REQ ("startRead") den Signalzustand "TRUE" liefert, wird die Anweisung "RDREC" gestartet.

Hinweis

Aufruf von "startRead"

Entsprechend dem Aufruf durch den OB82 wird "startRead" jedes Mal auf "TRUE" gesetzt, wenn ein Hardware-Fehler erkannt wird. Die erfolgreichen Aufrufe von RDREC werden über die Variable "countCalls" gezählt. Nach drei ausgelösten Kanalfehlern steht der Wert von "countCalls" auf "3".

Über den Eingangsparameter ID ("id") ruft die Anweisung "RDREC" das Kopfmodul des IO-Device auf. Durch den Eingangsparameter INDEX ("dataRecNbr") wird der Diagnosedatensatz "16#E00A" des IO-Device aufgerufen.

Über mehrere Aufrufe hinweg liesst die Anweisung "RDREC" die Diagnosedaten aus und speichert die gelesenen Daten (x Datensätze) über den Parameter RECORD ("recordBYTE") ab. Die Länge der gelesenen Daten wird über den Ausgangsparameter LEN ("#lengthOut") festgehalten und für weiteren Gebrauch über die Variable "readLength" gespeichert. Entsprechend des Werts des Eingangsparameters MLEN ("maxLength" hat den Wert "0") ist die Länge der zu lesenden Daten unbeschränkt.

Während der Verarbeitung ist der Ausgangsparameter BUSY ("busy") auf "TRUE" und VALID ("checkedValid") auf "FALSE" gesetzt. Ein Erfolg ("checkedValid" auf "TRUE") beim Lesen wird erst zum Ende der Verarbeitung ausgegeben und dient dem FB "SLI_FB_chooseStruct_RDiag" als Startbedingung. Zum Ende der Verarbeitung des FB "SLI_FB_chooseStruct_RDiag" wird die Variable "checkedValid" auf "FALSE" gesetzt.

Am Ausgangsparameter STATUS ("#statusExe") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft im Beispiel ohne Fehler.

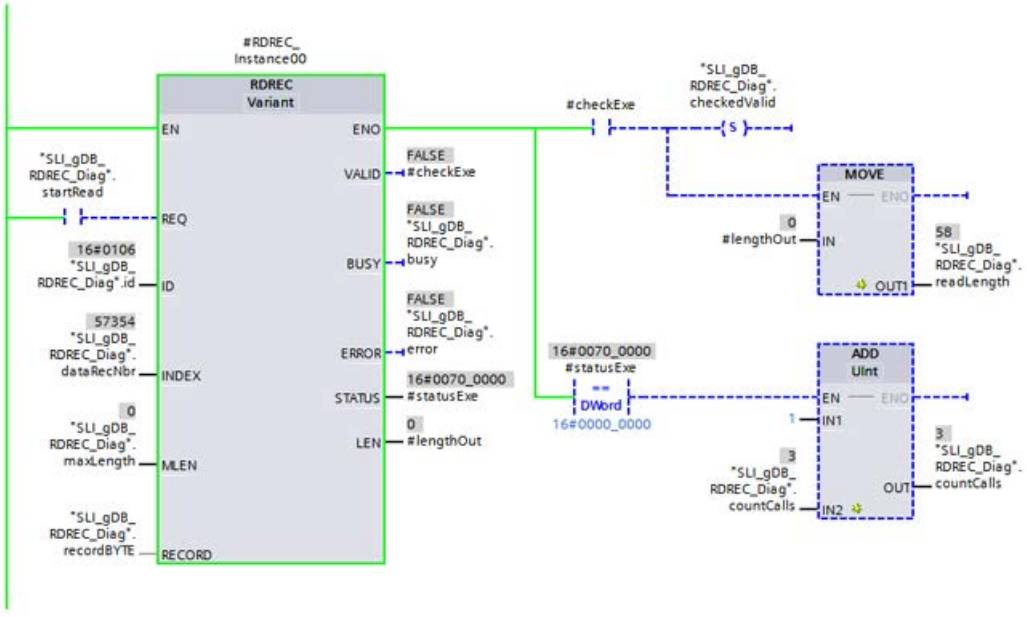

Netzwerk 3:

Wenn der Schließerkontakt ("error") den Signalzustand "TRUE" liefert, wird der Status ("#statusExe") im Fehlerfall gespeichert ("memErrStatus"). Zudem werden die Variablen "checkedValid" und "startRead" rückgesetzt. Die Verarbeitung von RDREC wird beendet. Die Verarbeitung des FB "SLI_FB_chooseStruct_RDiag" kann nicht gestartet werden.

Netzwerk 4:

Wenn die Verarbeitung von RDREC bereits komplett durchlaufen wurde ("#statusExe" hat den Wert "16#0070_0200"), wird "startRead" auf "FALSE" gesetzt. Die Verarbeitung von RDREC wird beendet.

Netzwerk 5:

Wenn der Schließerkontakt ("startRead") den Signalzustand "TRUE" liefert, springt die Verarbeitung (durch die Anweisung "JMP") gemäß dem Label "RDREC" zum Netzwerk 1.

Netzwerk 6:

Wenn die Variable "checkedValid" den Signalzustand "TRUE" und die Variable "reset" den Signalzustand "FALSE" liefert, wird der Sortiervorgang in dem FB "SLI_FB_chooseStruct_RDiag" gestartet.

Im FB "SLI_FB_chooseStruct_RDiag" wird der Strukturtyp des jeweiligen Datensatzes und der Strukturtyp entsprechend dem UserStructureIdentifier gewählt. Die Daten des Datensatzes werden in das passende Array, an passende Stelle kopiert. Zudem wird die Verarbeitung entsprechend der Größe des jeweiligen Datensatzes unter Umständen mehrfach durchlaufen. Dabei werden die Informationen eines oder mehrerer Kanäle ausgelesen.

Wenn zum Ende der Verarbeitung des Datensatz-Inhalts die "Länge des Datensatzes + vorhergehender Längen" ("#allRecLEN") kleiner ist als die Länge aller gelesenen Daten ("readLength"), beginnt die Verarbeitung des FBs von vorne. Ein weiterer Datensatz wird ausgelesen, seine Inhalte verglichen und seine Daten in das passende Array, an passende Stelle kopiert.

Der Sortiervorgang wird beendet, indem lokale Variablen ("#readComplete", "#dbComplete", "#limitReach") und die globale Variable "checkedValid" auf "FALSE" gesetzt werden.

Anhand der folgenden Variablen wird der Sortiervorgang dokumentiert:

- "data_sorting_done": Diese Variable zeigt an, dass die Verarbeitung des FB "SLI_FB_chooseStruct_RDiag" vollständig durchlaufen und beendet wurde. Die Variable "checkedValid" wird rückgesetzt, wodurch die Verarbeitung des FBs nicht von Neuem durchlaufen werden kann.
- "outputOverflow": Diese Struktur enthält drei Variablen, welche jeweils eine andere Überschreitung von Array-Grenzen festhält. Zum Beispiel könnten mehr Datensätze ausgelesen worden sein, als das entsprechende Array ("recordA" oder "recordB") zu speichern vermag.
- "different_structs_used": Wenn sowohl "recordA" als auch "recordB" zur Ablage von Datensätzen einer Diagnose verwendet wurden, liefert die Variable "different_structs_used" den Wert "TRUE".
- "countRecs": Diese Variable zählt die kopierten Datensätze.
- "chooseStruct": Diese Variable zeigt die zuletzt verwendete Struktur an. Der Wert "4" gibt an, dass zuletzt in der Struktur "recordB" mit der Ausprägung "USI Typ 1" ein Datensatz abgelegt wurde.
- "posForRec": Diese Variable speichert die zuletzt verwendete Stelle im Array der Datensätze ("recordA" oder "recordB"). Der Wert "1" zeigt an, dass zuletzt im "recordB[1]" (siehe "chooseStruct") ein Datensatz abgespeichert wurde.
- "posForChannel": Diese Variable speichert die zuletzt verwendete Stelle im Array der Kanäle ("USI_DetailsType[x]"). Der Wert "0" zeigt an, dass in "recordB[1]/ USI_DetailsType[0]" (siehe "chooseStruct" und "posForRec") ein Kanal abgespeichert wurde.

SLI_gDB_RDREC_Diag

	Name	Data type	Start value	Monitor value
1	Static			
2	id	HW_IO	262	16#0106
3	dataRecNbr	DInt	16#E00A	57354
4	startRead	Bool	false	FALSE
5	maxLength	UInt	0	0
6	maxPosition	UInt	4175	4175
7	recordBYTE	Array[0..4175] of B...		
8	readLength	UInt	0	58
9	checkedValid	Bool	false	FALSE
10	busy	Bool	false	FALSE
11	error	Bool	false	FALSE
12	memErrStatus	DWord	16#0	16#0000_0000
13	countCalls	UInt	0	3
14	maxChannelPos	UInt	15	15
15	maxRecordPos	UInt	5	5
16	recordA	Array[0..5] of "RECd...		
17	recordB	Array[0..5] of "RECd...		
18	data_sorting_done	Bool	false	TRUE
19	outputOverflow	Struct		
20	different_structs_used	Bool	false	FALSE
21	countRecs	UInt	0	2
22	chosenStruct	UInt	0	4
23	posForRec	UInt	0	1
24	posForChannel	UInt	0	0
25	resetOps	Struct		
26	reset	Bool	false	FALSE

Im Folgenden ist der Datensatz mit der Struktur "B"/"0101" auf der Position "0" ("recordB[0]"), mit zwei gelesenen Kanälen zu sehen.

Der Wert "16#8000" von "ChannelLocation" zeigt eine Diagnose auf Submodul-Ebene an. Laut dem Wert "16#0006" von "ChannelErrorType" liegt ein Drahtbruch vor, an den Kanälen 0 und 1 (siehe "ChannelNumber") des Submoduls 1 (siehe "SubSlotNumber"), des Moduls auf Steckplatz 2 (siehe "SlotNumber").

17	recordB	Array[0..5] of "RECd...		
18	recordB[0]	"RECdiag_0101"		
19	Body0101	"Rd_0101body"		
20	BlockType	Word	16#0	16#0010
21	BlockLength	UInt	0	28
22	BlockVersion_HL	Word	16#0	16#0101
23	API	DWord	16#0	16#0000_0000
24	SlotNumber	UInt	0	2
25	SubSlotNumber	UInt	0	1
26	ChannelLocation	Word	16#0	16#8000
27	ChannelProperties	Array[0..15] of Bool		
28	UserStructureIdentifier	Word	16#0	16#8000
29	USI_DetailsType1	Array[0..15] of "USI...		
30	USI_DetailsType1[0]	"USI_v1"		
31	ChannelNumber	UInt	0	0
32	ChannelProperties	Array[0..15] of Bool		
33	ChannelErrorType	Word	16#0	16#0006
34	USI_DetailsType1[1]	"USI_v1"		
35	ChannelNumber	UInt	0	1
36	ChannelProperties	Array[0..15] of Bool		
37	ChannelProperties[0]	Bool	false	FALSE
38	ChannelProperties[1]	Bool	false	FALSE
39	ChannelProperties[2]	Bool	false	FALSE
40	ChannelProperties[3]	Bool	false	TRUE
41	ChannelProperties[4]	Bool	false	FALSE
42	ChannelProperties[5]	Bool	false	FALSE
43	ChannelProperties[6]	Bool	false	TRUE
44	ChannelProperties[7]	Bool	false	FALSE
45	ChannelProperties[8]	Bool	false	TRUE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

RDREC: Datensatz lesen (Seite 2306)

WRREC: Datensatz schreiben

WRREC: Datensatz schreiben

Beschreibung

Mit der Anweisung "WRREC" übertragen Sie den Datensatz RECORD zu der mittels ID adressierten Komponente. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

- Über den Parameter ID wählen Sie aus, für welche Baugruppe ein Datensatz geschrieben werden soll. Verwenden Sie für den Parameter ID die Hardware-Kennung (HW-ID) der Baugruppe. Die Hardware-Kennung der Baugruppe finden Sie unter "PLC-Variablen > Systemkonstanten". Z. B. Für eine S7-1500 DI-Baugruppe: Diese wird als <Name der DI-Baugruppe> mit dem Datentyp "Hw_SubModule" angezeigt.
- Über den Parameter INDEX wählen Sie aus, welcher Datensatz geschrieben wird. Welche Datensätze mit welcher Datensatznummer geschrieben werden können, ist abhängig von der Baugruppe. Weitere Informationen hierzu finden Sie in dem Gerätehandbuch der jeweiligen Baugruppe.
- Mit LEN geben Sie die Länge des zu übertragenden Datensatzes in Bytes vor. Wählen Sie den Quellbereich RECORD mindestens LEN Bytes lang.
- Der Wert TRUE des Ausgangsparameters DONE zeigt an, dass der Datensatz erfolgreich übertragen wurde.
- Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Hinweis

Die Schnittstelle der Anweisung "WRREC" ist identisch mit der des in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "WRREC".

Arbeitsweise

"WRREC" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "WRREC" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Beachten Sie, dass Sie dem Aktualparameter von RECORD bei allen zu ein und demselben Auftrag gehörenden Aufrufen von "WRREC" denselben Wert zuweisen. Dasselbe gilt für die Aktualparameter von LEN.

Wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat, ist die Datensatzübertragung abgeschlossen.

Aufruf von WRREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems

Wenn Sie die Anweisung WRREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems aufrufen, werden stets die Fehlercodes 0x7001 und 0x7002 zurückgeliefert. Auch im Fehlerfall, wenn z. B. das Datensatzziel nicht vorhanden ist, wird kein anderer Fehlercode geliefert. In diesem Fall wird der korrekte Fehlercode erst nach Beenden des SYNCUP geliefert.

Wenn Sie die Anweisung WRREC im Systemzustand SYNCUP eines S7-1500-R/H-Systems aufrufen, wird kein Datensatzauftrag angestoßen. Der Anstoß des Datensatzauftrags efolgt erst, wenn Sie die Anweisung nach Beendigung des SYNCUP erneut aufrufen. Es wird daher empfohlen, WRREC stets zyklisch aufzurufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRREC":

Parameter	Deklaration	Datentyp**	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T*, Z* oder Konstante	REQ = 1: Datensatzübertragung durchführen
ID	Input	HW_IO	E, A, M, D, L oder Konstante	Identifikationsnummer der Hardware-Komponente (DP/PROFINET IO) Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Komponente oder des Interface in der Hardware-Konfiguration hinterlegt.
INDEX	Input	DINT	E, A, M, D, L oder Konstante	Datensatznummer
LEN	Input	UINT	E, A, M, D, L oder Konstante	(ausgeblendet) maximale Länge des zu übertragenden Datensatzes in Bytes
DONE	Output	BOOL	E, A, M, D, L	Datensatz wurde übertragen
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Schreibvorgang trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus bzw. Fehlerinformation Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2370).
RECORD	InOut	VARIANT	E, A, M, D, L	Datensatz

* Nur bei S7-1500.

** Bei AWL erfolgt keine implizierte Konvertierung, weshalb der Umfang der zulässigen Datentypen eingeschränkt sein kann. Beachten Sie bei der Programmierung in AWL die jeweils zulässigen Datentypen im Tooltipp des Parameters.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie "WRREC" für das Schreiben eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2370).

Beispiele

Ein Beispiel finden Sie hier: Programmbeispiel zu Parameterdatensatz mit WRREC & RDREC verwenden (Seite 2328).

Ein weiteres Beispiel finden Sie hier: Programmbeispiel zu Steuerdatensatz mit WRREC & RDREC verwenden (Seite 2338).

Weitere Informationen und den Programmcode zu den Beispielen finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die Typen von Datensätzen (Seite 2298)

Programmbeispiel zu Parameterdatensatz mit WRREC & RDREC verwenden

Einleitung

In dem folgenden Beispiel lesen Sie über RDREC den Parameterdatensatz einer digitalen Eingabebaugruppe aus und schreiben über WRREC den Datensatz neu. Sie ändern dabei die Eingangsverzögerung.

Weitere Informationen zu Datensätzen finden Sie in der Übersicht über die Typen von Datensätzen (Seite 2298).

Voraussetzung

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein folgende Variablen an:

- zwei Variablen für allgemeinen Gebrauch und sieben Variablen für WRREC:

SLI_gDB_WRREC_Para			
	Name	Data type	Start value
1	Static		
2	id	HW_IO	257
3	dataRecNbr	DInt	3
4	startWrite	Bool	false
5	maxWriteLength	UInt	4
6	writeCheckedValid	Bool	false
7	writeBusy	Bool	false
8	writeError	Bool	false
9	writeStatus	DWord	16#0
10	wrMemErrStat	DWord	16#0

- eine Struktur für den Datensatz an WRREC:

11	writeParaRecord	Struct	
12	Version	USInt	16
13	LengthPerChannel	USInt	2
14	ClearanceDiagnosis	Array[0..7] of Bool	
15	InputDelay_ProcessEvent	Array[0..7] of Bool	
16	InputDelay_ProcessEvent[0]	Bool	1
17	InputDelay_ProcessEvent[1]	Bool	false
18	InputDelay_ProcessEvent[2]	Bool	false
19	InputDelay_ProcessEvent[3]	Bool	false
20	InputDelay_ProcessEvent[4]	Bool	false
21	InputDelay_ProcessEvent[5]	Bool	false
22	InputDelay_ProcessEvent[6]	Bool	false
23	InputDelay_ProcessEvent[7]	Bool	false

- acht Variablen für RDREC:

24	startRead	Bool	false
25	maxReadLength	UInt	4
26	readLength	UInt	0
27	readCheckedValid	Bool	false
28	readBusy	Bool	false
29	readError	Bool	false
30	readStatus	DWord	16#0
31	reMemErrStat	DWord	16#0

- eine Struktur für den Datensatz an RDREC:

32	readParaRecord	Struct	
33	Version	USInt	0
34	LengthPerChannel	USInt	0
35	ClearanceDiagnosis	Array[0..7] of Bool	
36	InputDelay_ProcessEvent	Array[0..7] of Bool	
37	InputDelay_ProcessEvent[0]	Bool	false
38	InputDelay_ProcessEvent[1]	Bool	false
39	InputDelay_ProcessEvent[2]	Bool	false
40	InputDelay_ProcessEvent[3]	Bool	false
41	InputDelay_ProcessEvent[4]	Bool	false
42	InputDelay_ProcessEvent[5]	Bool	false
43	InputDelay_ProcessEvent[6]	Bool	false
44	InputDelay_ProcessEvent[7]	Bool	false

In einem FB "SLI_FB_WRREC_Para" legen Sie folgende lokale Variablen an.

7	Static		
8	WRREC_Instance	WRREC	
9	RDREC_Instance	RDREC	
10	readCheck	Bool	false
11	writeCheck	Bool	false
12	readLength	UInt	0
13	readBYTE	Struct	
14	writeBYTE	Struct	

Parameter verschalten: FC "SLI_itemisePara_WRREC_Para"

Zur Weitergabe der Werte einer Bitfolge (0..7) legen Sie den FC "SLI_itemisePara_WRREC_Para" an.

Darin legen Sie die folgenden lokalen Variablen an.

SLI_itemisePara_WRREC_Para		
	Name	Data type
1	Input	
2	in0	Bool
3	in1	Bool
4	in2	Bool
5	in3	Bool
6	in4	Bool
7	in5	Bool
8	in6	Bool
9	in7	Bool
10	Output	
11	out0	Bool
12	out1	Bool
13	out2	Bool
14	out3	Bool
15	out4	Bool
16	out5	Bool
17	out6	Bool
18	out7	Bool

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) Verschaltungen an.

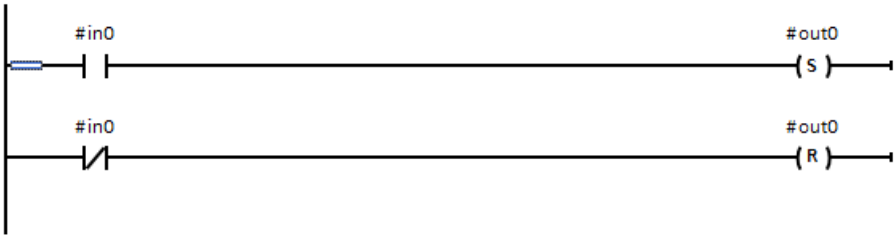

Parameter verschalten: FB "SLI_FB_WRREC_Para"

In dem FB "SLI_FB_WRREC_Para" legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Die Parameter der Anweisung "RDREC" verschalten Sie wie folgt.

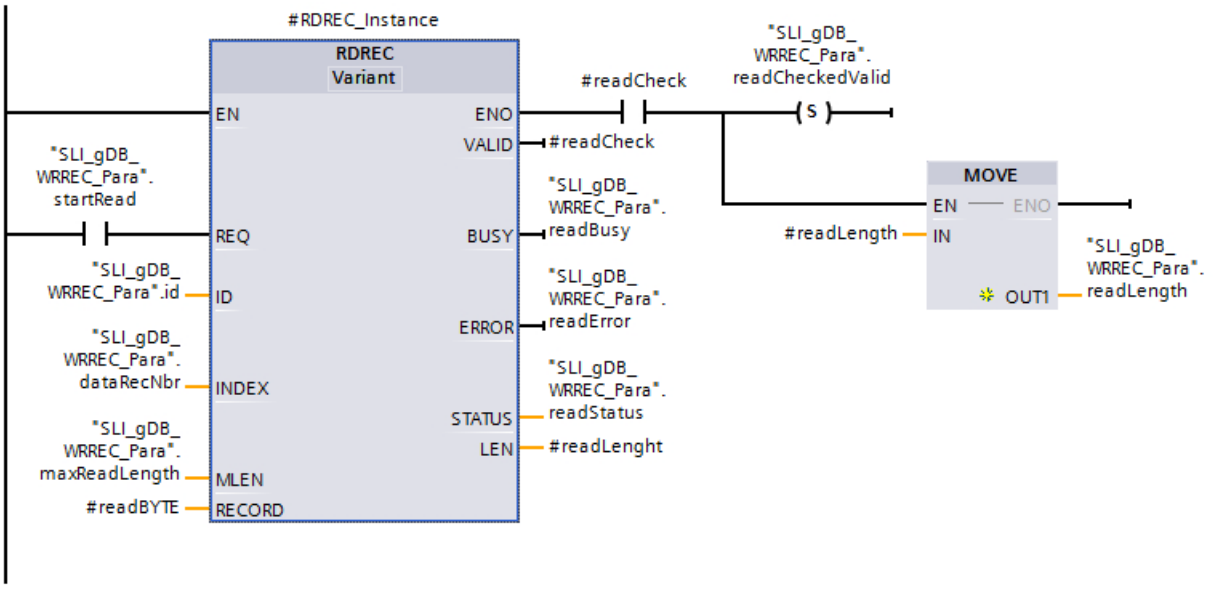

Netzwerk 2: Im Folgenden verschalten Sie alle Variablen der Struktur "#readBYTE" mit der Struktur "readParaRecord". Teil 1 des Netzwerks:

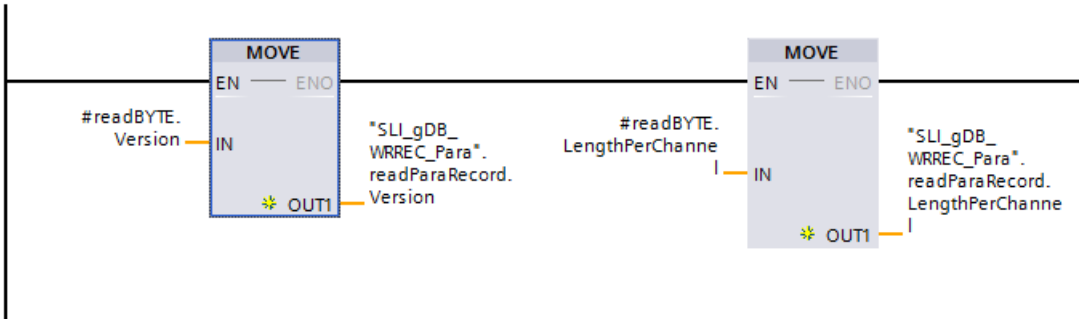

Teil 2 des Netzwerks: Die Variablen "ClearanceDiagnosis" und "InputDelay_ProcessEvent" verschalten Sie nach folgendem Vorbild.

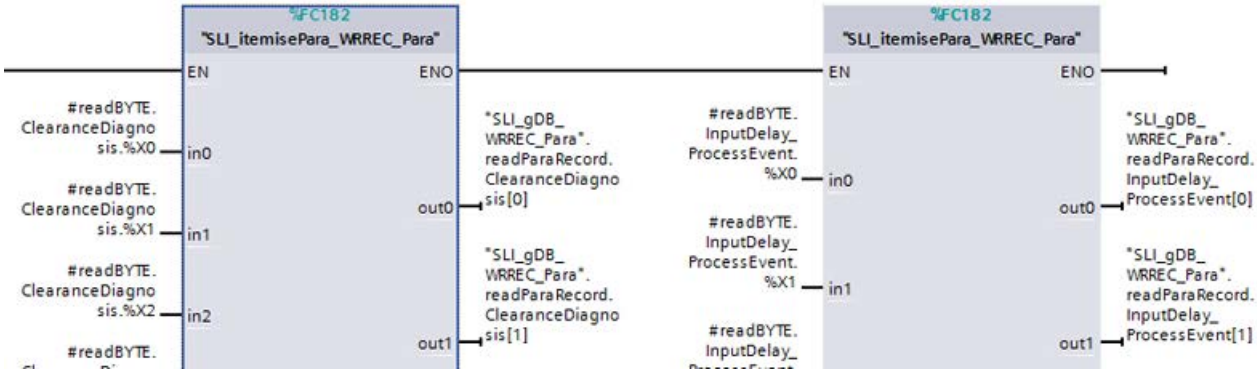

Netzwerk 3: Im Folgenden verschalten Sie die Variablen für den Fehlerfall von RDREC.

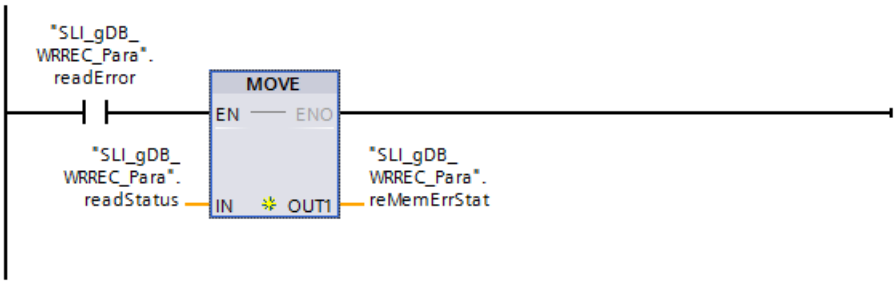

Netzwerk 4: Im Folgenden verschalten Sie die Variablen zum Beenden von RDREC.

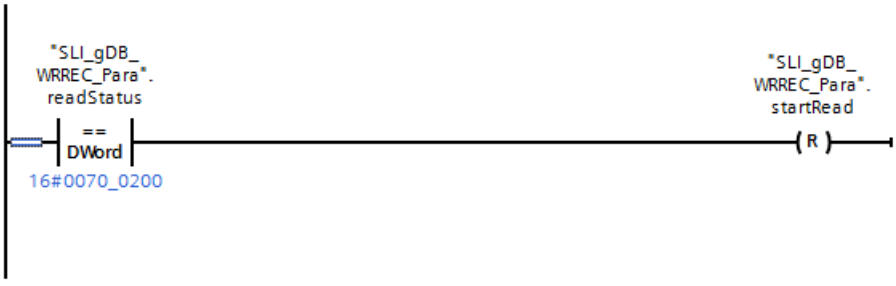

Netzwerk 5: Im Folgenden verschalten Sie alle Variablen der Struktur "writeParaRecord" mit der Struktur "#writeBYTE". Teil 1 des Netzwerks:

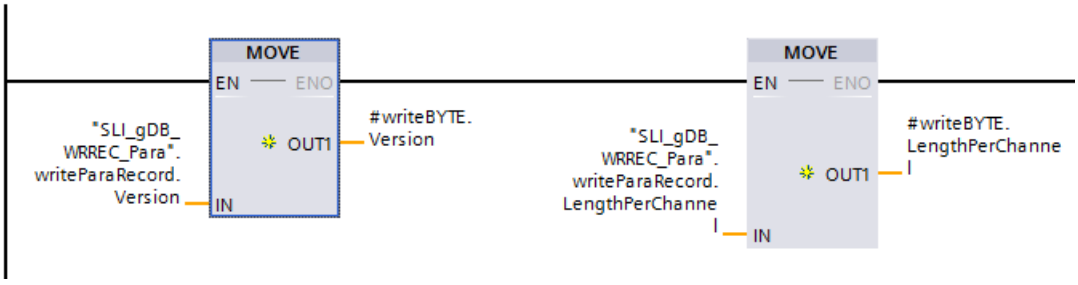

Teil 2 des Netzwerks:

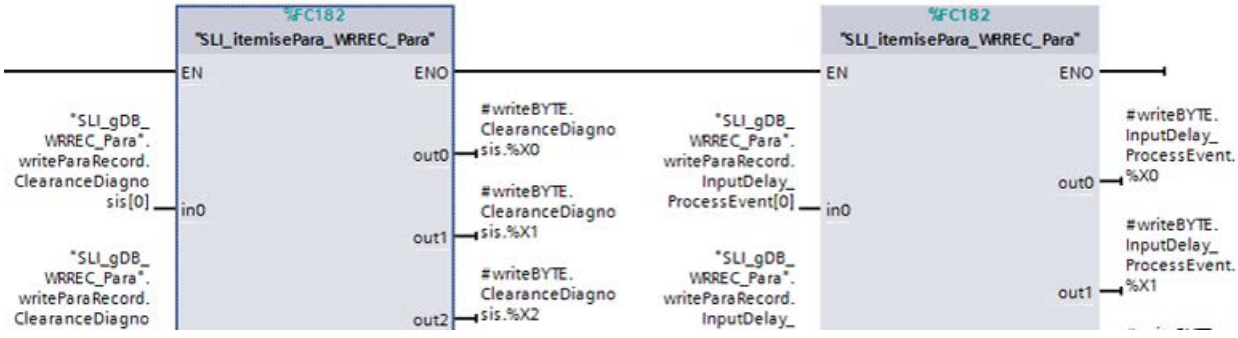

Netzwerk 6: Im Folgenden verschalten Sie die Parameter der Anweisung "WRREC".

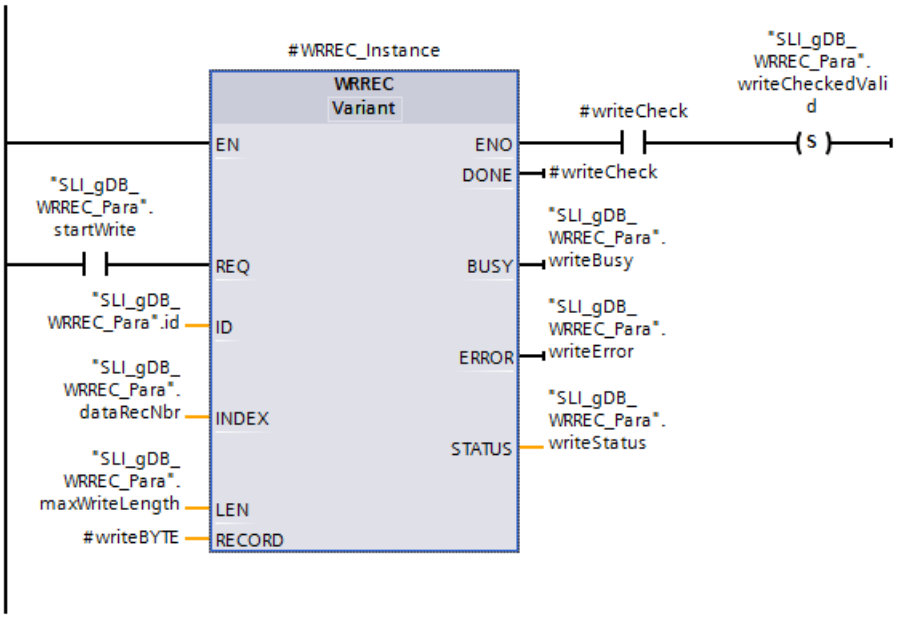

Netzwerk 7: Im Folgenden verschalten Sie die Variablen für den Fehlerfall von WRREC.

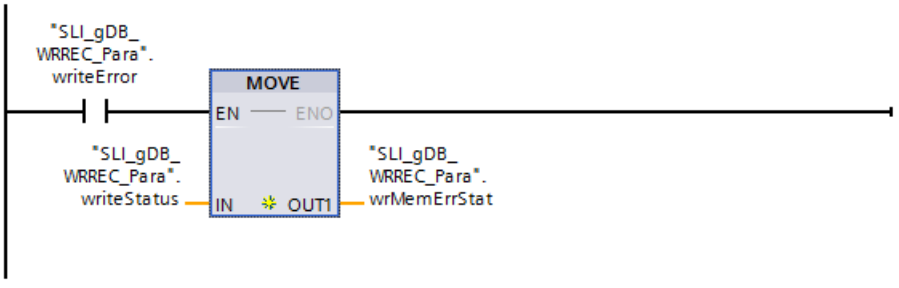

Netzwerk 8: Im Folgenden verschalten Sie die Variablen zum Beenden von WRREC und dem erneuten Aufrufen von RDREC.

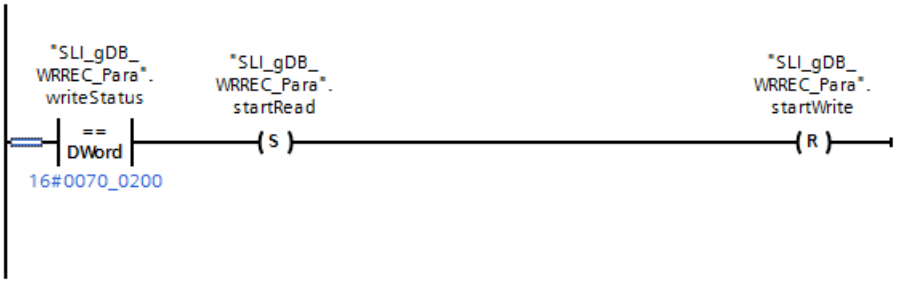

Ergebnis von Netzwerk 1:

Sie legen eine Eingabebaugruppe an und lesen die Hardware-Kennung in den Eigenschaften der Eingabebaugruppe aus. Die Hardware-Kennung hinterlegen Sie über den Eingangsparameter ID ("id"). Durch den Eingangsparameter INDEX ("dataRecNbr") wird der Datensatz der Nummer "3" der Baugruppe aufgerufen. Der Datensatz entspricht dem gleichnamigen Kanal.

SLI_gDB_WRREC_Para

	Name	Data type	Start value	Monitor value
1	Static			
2	id	HW_IO	257	16#0101
3	dataRecNbr	DInt	3	3

Wenn der Eingangsparameter REQ ("startRead") den Signalzustand "TRUE" liefert, wird die Anweisung "RDREC" gestartet. Über den Eingangsparameter ID ("id") ruft die Anweisung "RDREC" die Eingabebaugruppe auf.

Über mehrere Aufrufe hinweg liesst die Anweisung "RDREC" den Datensatz aus und speichert den Datensatz über den Parameter RECORD ("#readBYTE") ab. Die Länge des gelesenen Datensatzes wird über den Ausgangsparameter LEN ("#readLength") festgehalten. Entsprechend des Werts des Eingangsparameters MLEN ("maxReadLength") ist die zu lesende Datensatzinformation auf 4 Byte beschränkt.

Während der Verarbeitung ist der Ausgangsparameter BUSY ("readBusy") auf "TRUE" und VALID ("#readCheck") auf "FALSE" gesetzt.

Ein Erfolg beim Lesen wird wie folgt zum Ende der Verarbeitung ausgegeben:

- Der Ausgangsparameter VALID ("#readCheck") wird auf "TRUE" gesetzt und setzt die Variable "readCheckedValid" dauerhaft auf "TRUE".
- Zudem wird der Wert der lokalen Variable "#readLength" in der Variable "readLength" dauerhaft gespeichert.

Am Ausgangsparameter STATUS ("readStatus") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

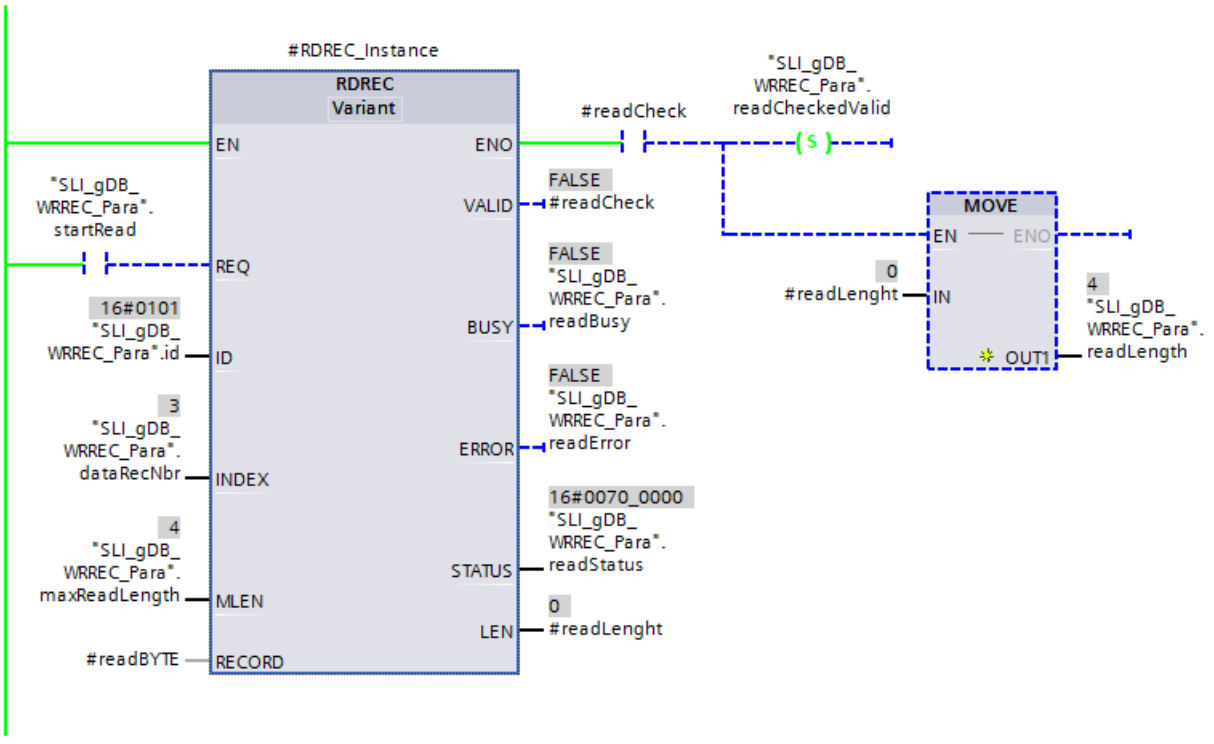

Ergebnis von Netzwerk 2:

Über MOVE-Anweisungen und den FC "SLI_itemisePara_WRREC_Para" werden die Werte der Variablen von der Struktur "#readBYTE" an die gleichnamigen Variablen der Struktur "readParaRecord" übergeben. Zum besseren Verständnis sind die Variablen der Struktur "readParaRecord" als Klarnamen vertreten. Der Aufbau ist im Gerätehandbuch der Baugruppe beschrieben. Angezeigt wird z. B. durch "InputDelay_ProcessEvent" eine Eingangsverzögerung von 3,2 ms. Für den Zustand des Kanals ("ClearanceDiagnosis") wird keine Störung ausgegeben (Bit 0 bis 8 sind "FALSE").

32	readParaRecord	Struct		
33	Version	USInt	0	16
34	LengthPerChannel	USInt	0	2
35	ClearanceDiagnosis	Array[0..7] of Bool		
36	InputDelay_ProcessEvent	Array[0..7] of Bool		
37	InputDelay_ProcessEvent[0]	Bool	false	FALSE
38	InputDelay_ProcessEvent[1]	Bool	false	TRUE
39	InputDelay_ProcessEvent[2]	Bool	false	TRUE
40	InputDelay_ProcessEvent[3]	Bool	false	FALSE
41	InputDelay_ProcessEvent[4]	Bool	false	FALSE
42	InputDelay_ProcessEvent[5]	Bool	false	FALSE
43	InputDelay_ProcessEvent[6]	Bool	false	FALSE
44	InputDelay_ProcessEvent[7]	Bool	false	FALSE

Ergebnis von Netzwerk 3:

Wenn "readError" den Signalzustand "TRUE" liefert, wird der Wert der Variable "readStatus" in der Variable "reMemErrStat" gespeichert.

Ergebnis von Netzwerk 4:

Wenn die Verarbeitung der Anweisung "RDREC" durchgelaufen ist, liefert "readStatus" den Wert "16#0070_0200". Daraufhin setzt ein Rücksetzer die Variable "startRead" auf "FALSE".

Ergebnis von Netzwerk 5:

Über MOVE-Anweisungen und den FC "SLI_itemisePara_WRREC_Para" werden die Werte der Variablen von der Struktur "writeParaRecord" an die gleichnamigen Variablen der Struktur "#writeBYTE" übergeben. Zum besseren Verständnis sind die Variablen der Struktur "writeParaRecord" als Klarnamen vertreten.

Ergebnis von Netzwerk 6:

Wenn der Eingangsparameter REQ ("startWrite") den Signalzustand "TRUE" liefert, wird die Anweisung "WRREC" gestartet.

Am Parameter RECORD ("#writeBYTE") sind die neuen Daten für den Datensatz hinterlegt. Über mehrere Aufrufe hinweg schreibt die Anweisung "WRREC" die neuen Daten in den Datensatz "3". Entsprechend des Werts des Eingangsparameters LEN ("maxWriteLength") ist die zu schreibende Datensatzinformation auf 4 Byte beschränkt.

Während der Verarbeitung ist der Ausgangsparameter BUSY ("writeBusy") auf "TRUE" und DONE ("#writeCheck") auf "FALSE" gesetzt.

Ein Erfolg beim Schreiben wird wie folgt zum Ende der Verarbeitung ausgegeben:

- Der Ausgangsparameter DONE ("#writeCheck") wird auf "TRUE" gesetzt.
- Wenn "#writeCheck" den Signalzustand "TRUE" liefert, wird die Variable "writeCheckedValid" dauerhaft auf "TRUE" gesetzt.

Am Ausgangsparameter STATUS ("writeStatus") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

4		startWrite	Bool	false	FALSE
5		maxWriteLength	UInt	4	4
6		writeCheckedValid	Bool	false	TRUE
7		writeBusy	Bool	false	FALSE
8		writeError	Bool	false	FALSE
9		writeStatus	DWord	16#0	16#0070_0000
10		wrMemErrStat	DWord	16#0	16#0000_0000

Ergebnis von Netzwerk 7:

Wenn "writeError" den Signalzustand "TRUE" liefert, wird der Wert der Variable "writeStatus" in der Variable "wrMemErrStat" gespeichert.

Ergebnis von Netzwerk 8:

Wenn die Verarbeitung der Anweisung "WRREC" durchgelaufen ist, liefert "writeStatus" den Wert "16#0070_0200". Daraufhin setzt der Rücksetzer die Variable "startWrite" auf "FALSE". Die Variable "startRead" wird auf "TRUE" gesetzt, woraufhin die Anweisung "RDREC" von Neuem abgearbeitet wird. Die Struktur "readParaRecord" zeigt die neu geschriebenen Werte des Datensatzes an. Da in "InputDelay_ProcessEvent" das Bit 0 den Signalzustand "TRUE" liefert, besitzt der Kanal 3 nun eine Eingangsverzögerung von 0,1 ms.

32	readParaRecord	Struct		
33	Version	USInt	0	16
34	LengthPerChannel	USInt	0	2
35	ClearanceDiagnosis	Array[0..7] of Bool		
36	InputDelay_ProcessEvent	Array[0..7] of Bool		
37	InputDelay_ProcessEvent[0]	Bool	false	TRUE
38	InputDelay_ProcessEvent[1]	Bool	false	FALSE
39	InputDelay_ProcessEvent[2]	Bool	false	FALSE
40	InputDelay_ProcessEvent[3]	Bool	false	FALSE
41	InputDelay_ProcessEvent[4]	Bool	false	FALSE
42	InputDelay_ProcessEvent[5]	Bool	false	FALSE
43	InputDelay_ProcessEvent[6]	Bool	false	FALSE
44	InputDelay_ProcessEvent[7]	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

WRREC: Datensatz schreiben (Seite 2326)

RDREC: Datensatz lesen (Seite 2306)

Programmbeispiel zu Steuerdatensatz mit WRREC & RDREC verwenden

Einleitung

In dem folgenden Beispiel schreiben Sie den Steuerdatensatz 196 über WRREC und lesen zur Überprüfung über RDREC den Datensatz aus. Zur Konfiguration wählen Sie den zentralen Aufbau einer S7-1500.

Weitere Informationen zu Datensätzen finden Sie in der Übersicht über die Typen von Datensätzen (Seite 2298).

Voraussetzung

Hinweis

Konfigurationssteuerung aktivieren

Damit der Steuerdatensatz geschrieben werden kann, muss die Konfigurationssteuerung (Optionenhandling) der CPU aktiviert sein. Diese finden Sie unter "CPU-Eigenschaften > Konfigurationssteuerung".

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein folgende Variablen an:

- drei Variablen für allgemeine Nutzung und sechs für WRREC:

SLI_gDB_WRREC_Control

	Name	Data type	Start value
1	Static		
2	id	HW_IO	33
3	dataRecNbr	DInt	196
4	maxLength	UInt	0
5	configDone	Bool	false
6	checkedValid_WR	Bool	false
7	busyWR	Bool	false
8	errorWR	Bool	false
9	statusWR	DWord	16#0
10	memErrStatusWR	DWord	16#0

- eine Struktur für die Daten an WRREC (Datensatz 196):

11	recConfigCtrl_WR	Struct	
12	BlockLength	USInt	8
13	BlockID	USInt	196
14	Version	USInt	4
15	Subversion	USInt	0
16	Slot_0	USInt	0
17	Slot_1	USInt	1
18	Slot_2	USInt	2
19	Slot_3	USInt	3

Der Datensatz ist wie folgt aufgebaut:

– "BlockLength" gibt die Länge des Datensatzes in Byte an.

– "BlockID" gibt die Datensatznummer an.

– "Version" gibt die Art des Aufbaus an. Für das Beispiel wird ein Zentraler Aufbau verwendet. Bei einem Zentralen Aufbau hat "Subversion" den Wert "0".

– "Slot_0" bis "Slot_.." entsprechen den verwendeten Slots. Die Werte der Slots geben die zu verwendende Nummierung an.

- sechs Variablen für RDREC:

20		startRead	Bool	false
21		checkedValid_RD	Bool	false
22		lengthRD	UInt	0
23		busyRD	Bool	false
24		errorRD	Bool	false
25		statusRD	DWord	16#0

- eine Struktur für die Daten an RDREC:

26		recConfigCtrl_RD	Struct	
27		BlockLength	USInt	0
28		BlockID	USInt	0
29		Version	USInt	0
30		Subversion	USInt	0
31		Slot_0	USInt	0
32		Slot_1	USInt	0
33		Slot_2	USInt	0
34		Slot_3	USInt	0

Die Anweisung "WRREC", zum Schreiben des Steuerdatensatzes, muss in einem Anlauf-OB aufgerufen werden. In dem Anlauf-OB legen Sie für die Datenweitergabe folgende lokale Variable an:

4		Temp	
5		checkWR	Bool
6		statusWR	DWord

In einem FB legen Sie die folgenden lokalen Variablen an. Den FB rufen Sie in einem Programmzyklus-OB (OB1) auf.

7		Static			
8		RDREC_Instance	RDREC		
9		memErrStatus	DWord	16#0	Set in IDB
10		checkRD	Bool	false	Non-retain
11		lengthRD	UInt	0	Non-retain

Parameter verschalten im Anlauf-OB

Netzwerk 1: Im Folgenden verschalten Sie die Parameter der Anweisung "WRREC".

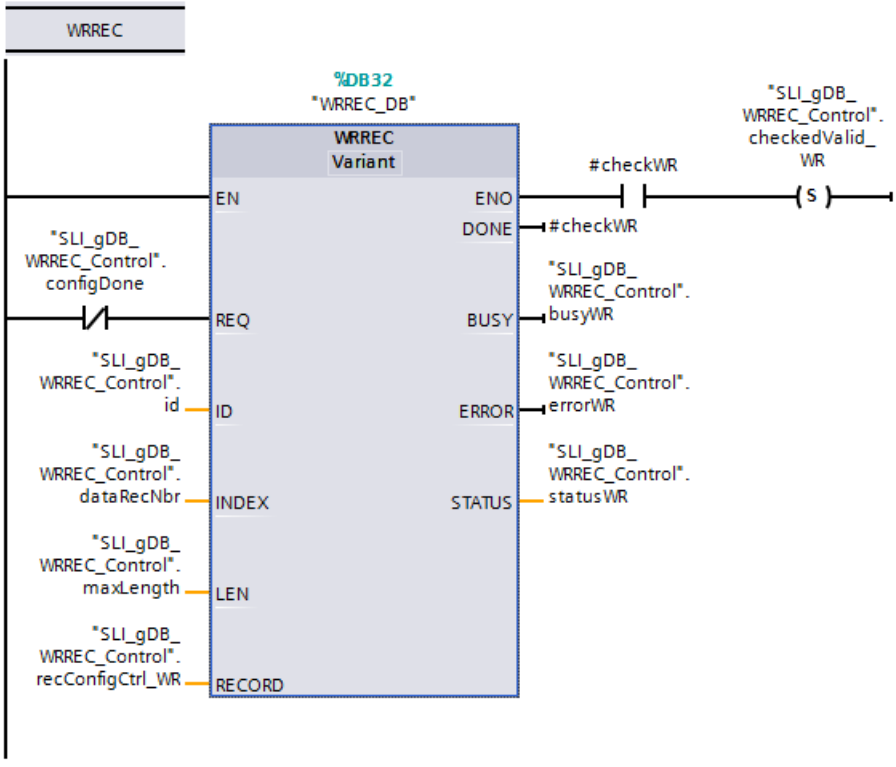

Netzwerk 2: Im Folgenden verschalten Sie die Parameter für den Fehlerfall von WRREC.

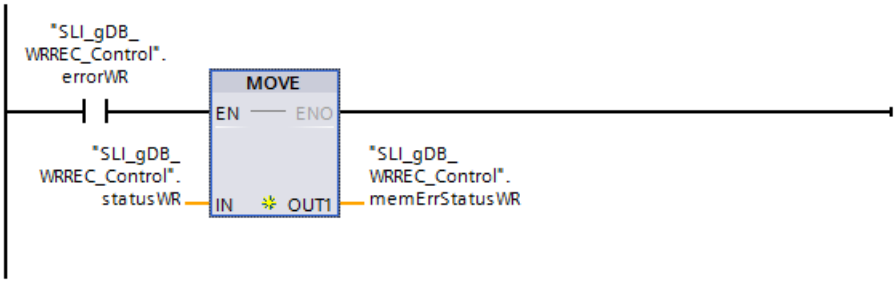

Netzwerk 3: Im Folgenden verschalten Sie die Parameter zum Beenden von WRREC.

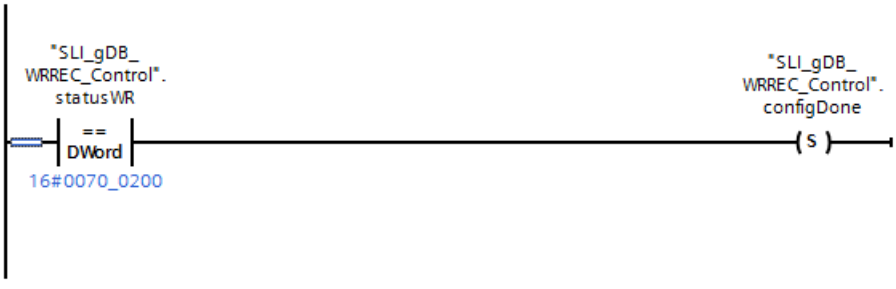

Netzwerk 4: Im Folgenden verschalten Sie die Parameter zur Abfrage des wiederholten Aufruf von WRREC.

Parameter verschalten im FB

Netzwerk 1: Im Folgenden verschalten Sie die Parameter der Anweisung "RDREC".

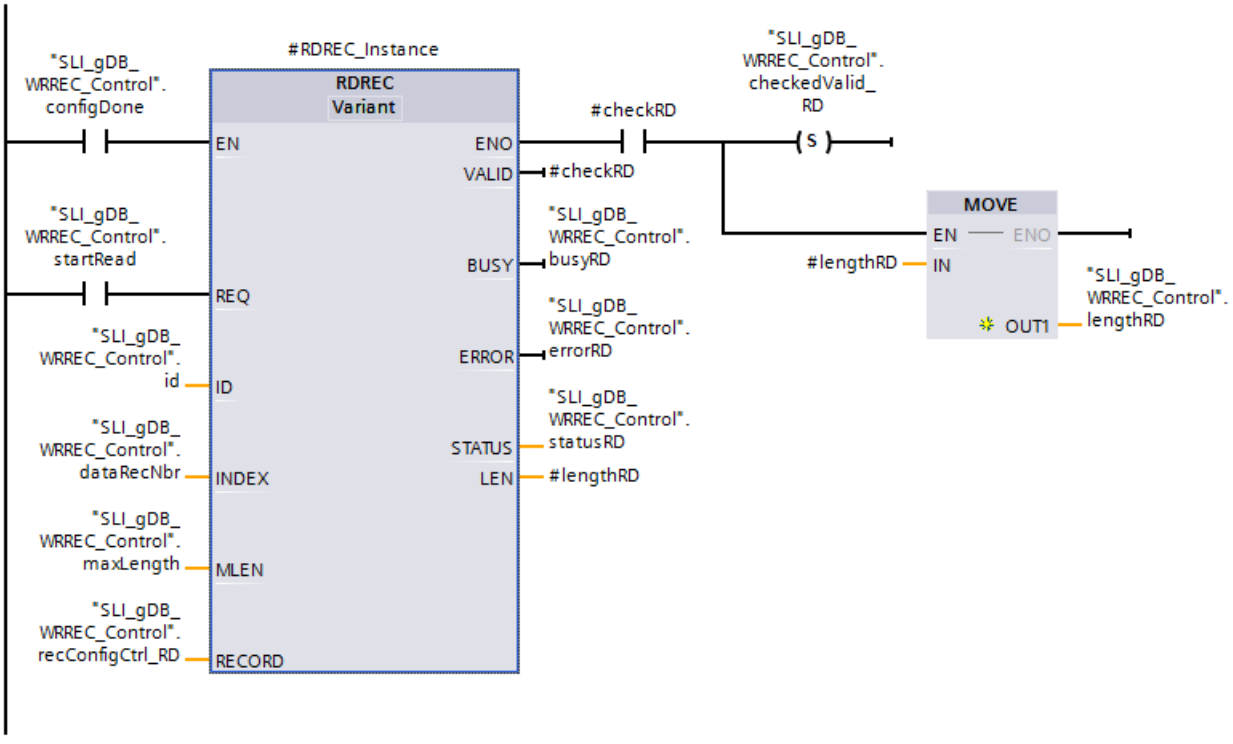

Netzwerk 2: Im Folgenden verschalten Sie die Parameter für den Fehlerfall von RDREC.

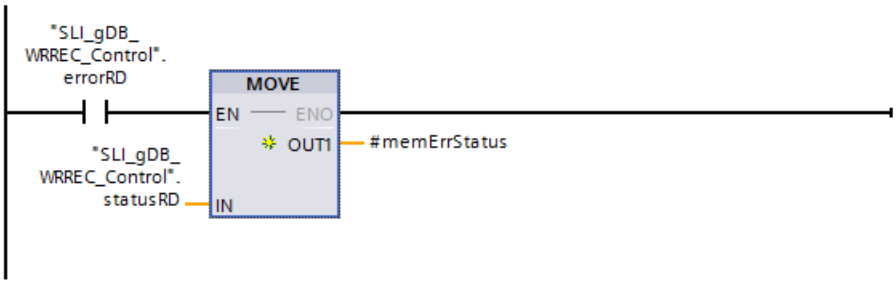

Netzwerk 3: Im Folgenden verschalten Sie die Parameter zum Beenden von RDREC.

Allgemeine Parameter

Die Hardware-Kennung der Konfiguration der CPU ("33") hinterlegen Sie über den Eingangsparameter ID ("id"). Den Steuerdatensatz ("196") hinterlegen Sie über den Eingangsparameter INDEX ("dataRecNbr"). Entsprechend des Werts der Variablen maxLength" (mit dem Wert "0") ist die zu berücksichtigende Datensatzinformationsmenge unbeschränkt.

Hinweis

Laden in die CPU und Verarbeitung des Programms im Anlauf

1. Zum Ladens des Programms setzen Sie die CPU in den Modus "STOPP".
2. Setzen Sie das Projekt in den Modus "Online".
3. Setzen Sie die CPU in den Modus "RUN". Der Modus "Anlauf" wird abgearbeitet.

Im Modus "Anlauf" arbeitet die CPU die im Anlauf-OB befindlichen Programmteile ab.

Anlauf-OB: Ergebnis des Netzwerk 1

Da der Eingangsparameter REQ den Signalzustand "TRUE" liefert, wird die Anweisung "WRREC" gestartet. Der Öffnerkontakt ("configDone") hat den Signalzustand "FALSE".

Am Parameter RECORD ("recConfigCtrl_WR") sind die Daten für den Steuerdatensatz hinterlegt. Zum besseren Verständnis sind die Variablen der Struktur "recConfigCtrl_WR" als Klarnamen vertreten und besitzen denselben Aufbau wie im Systemhandbuch beschrieben.

11	recConfigCtrl_WR	Struct		
12	BlockLength	USInt	8	8
13	BlockID	USInt	196	196
14	Version	USInt	4	4
15	Subversion	USInt	0	0
16	Slot_0	USInt	0	0
17	Slot_1	USInt	1	1
18	Slot_2	USInt	2	2
19	Slot_3	USInt	3	3

Über mehrere Aufrufe hinweg schreibt die Anweisung "WRREC" die Daten in den Steuerdatensatz. Während der Verarbeitung von WRREC ist der Ausgangsparameter BUSY ("busyWR") auf "TRUE" und DONE ("#checkWR") auf "FALSE" gesetzt.

Ein Erfolg beim Schreiben wird wie folgt zum Ende der Verarbeitung ausgegeben:

- Der Ausgangsparameter DONE ("#checkWR") wird auf "TRUE" gesetzt.und setzt die Variable "checkedValid_WR" dauerhaft auf "TRUE".

Am Ausgangsparameter STATUS ("statusWR") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

SLI_gDB_WRREC_Control

	Name	Data type	Start value	Monitor value
1	Static			
2	id	HW_IO	33	16#0021
3	dataRecNbr	DInt	196	196
4	maxLength	UInt	0	0
5	configDone	Bool	false	TRUE
6	checkedValid_WR	Bool	false	TRUE
7	busyWR	Bool	false	FALSE
8	errorWR	Bool	false	FALSE
9	statusWR	DWord	16#0	16#0000_0000
10	memErrStatusWR	DWord	16#0	16#0000_0000

Anlauf-OB: Ergebnis des Netzwerk 2

Im Fall eines Fehlers bei der Verarbeitung von WRREC:

Wenn "errorWR" den Signalzustand "TRUE" liefert, wird der Wert der Variable "statusWR" in der Variable "memErrStatusWR" gespeichert.

Anlauf-OB: Ergebnis des Netzwerk 3

Wenn die Verarbeitung der Anweisung "WRREC" durchgelaufen ist, liefert "statusWR" den Wert "16#0070_0200". Daraufhin wird die Variable "configDone" auf "TRUE" gesetzt. Die Verarbeitung von WRREC ist damit beendet.

Anlauf-OB: Ergebnis des Netzwerk 4

Solange die Verarbeitung der Anweisung "WRREC" noch nicht beendet ist ("configDone" hat den Wert "FALSE" und "busyWR" hat den Wert "TRUE") springt die Verarbeitung zurück zum Netzwerk 1.

FB: Ergebnis des Netzwerk 1

Nach der Verarbeitung der im Anlauf-OB befindlichen Programmteile befindet sich die CPU im Modus "RUN".

Wenn der Eingangsparameter EN ("configDone") den Signalzustand "TRUE" liefert, ist die Anweisung "RDREC" zu Verarbeitung freigegeben. Wenn der Eingangsparameter REQ ("startRead") den Signalzustand "TRUE" liefert, wird die Anweisung "RDREC" gestartet.

Über mehrere Aufrufe hinweg liesst die Anweisung "RDREC" den Steuerdatensatz aus und speichert den Datensatz uber den Parameter RECORD ("recConfigCtrl_RD") ab.

26	recConfigCtrl_RD	Struct		
27	BlockLength	USInt	0	8
28	BlockID	USInt	0	196
29	Version	USInt	0	4
30	Subversion	USInt	0	0
31	Slot_0	USInt	0	0
32	Slot_1	USInt	0	1
33	Slot_2	USInt	0	2
34	Slot_3	USInt	0	3

Die Länge des gelesenen Datensatzes wird über den Ausgangsparameter LEN ("#readLength") festgehalten. Während der Verarbeitung von RDREC ist der Ausgangsparameter BUSY ("busyRD") auf "TRUE" und VALID ("#checkRD") auf "FALSE" gesetzt.

Ein Erfolg beim Lesen wird wie folgt zum Ende der Verarbeitung ausgegeben:

- Der Ausgangsparameter VALID ("#checkRD") wird auf "TRUE" gesetzt. Daraufhin setzt #checkRD die Variable "checkedValid_RD" dauerhaft auf "TRUE".
- Zudem wird der Wert der lokalen Variable "#lengthRD" in der Variable "lengthRD" dauerhaft gespeichert.

Am Ausgangsparameter STATUS ("statusRD") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

20	startRead	Bool	false	FALSE
21	checkedValid_RD	Bool	false	TRUE
22	lengthRD	UInt	0	8
23	busyRD	Bool	false	FALSE
24	errorRD	Bool	false	FALSE
25	statusRD	DWord	16#0	16#0070_0000

FB: Ergebnis des Netzwerk 2

Im Fall eines Fehlers bei der Verarbeitung von RDREC:

Wenn "errorRD" den Signalzustand "TRUE" liefert, wird der Wert der Variable "statusRD" in der lokalen Variable "#memErrStatus" gespeichert.

FB: Ergebnis des Netzwerk 3

Wenn die Verarbeitung der Anweisung "RDREC" durchgelaufen ist und die Daten geprüft hat, liefert "#checkRD" den Wert "TRUE". Daraufhin wird die Variable "startRead" auf "FALSE" gesetzt. Die Verarbeitung der Anweisung "RDREC" wird beendet.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

WRREC: Datensatz schreiben (Seite 2326)

RDREC: Datensatz lesen (Seite 2306)

GETIO: Prozessabbild lesen

Beschreibung

Mit der Anweisung "GETIO" lesen Sie alle Eingänge eines DP-Normslaves / PROFINET IO-Devices konsistent aus. Die Anweisung "GETIO" ruft dabei die Anweisung "DPRD_DAT (Seite 2434)" auf. Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den Zielbereich am Parameter INPUTS eingetragen.

Für den Zielbereich gilt:

- Falls Sie für die selektierte Komponente eine Länge von genau einem Byte projektiert haben, müssen Sie als Zielbereich ein einzelnes Byte benutzen.
- Falls Sie für die selektierte Komponente eine Länge von mehreren Bytes projektiert haben, müssen Sie als Zielbereich ein ARRAY of BYTE benutzen, dessen Länge mit der Anzahl der projektierten Bytes identisch ist.

Benutzen Sie als Datentyp des Zielbereichs ausschließlich BYTE oder ARRAY of BYTE. Benutzen Sie insbesondere kein ARRAY of STRUCT. Falls Sie ein ARRAY of STRUCT benutzen, entstehen Lücken im Zielbereich.

Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen greifen Sie mit einem "GETIO"-Aufruf jeweils nur auf die Daten einer Komponente / DP-Kennung unter der projektierten Anfangsadresse zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Hardware-ID des DP-Normslave / PROFINET IO-Device.
STATUS	Output	DWORD	E, A, M, D, L	Enthält die Fehlerinformation von "DPRD_DAT (Seite 2434)" in der Form DW#16#40xxxx00.
LEN	Output	INT	E, A, M, D, L	Anzahl der gelesenen Daten in Bytes.
INPUTS	InOut	VARIANT	E, A, M, D, L	Zielbereich für die gelesenen Daten. Dieser muss genauso lang sein, wie Sie für den selektierten DP-Normslave / PROFINET IO-Device projektiert haben. Nur der Datentyp BYTE bzw. ARRAY of BYTE ist zulässig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Siehe auch: DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen (Seite 2434).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für GETIO & SETIO (Seite 2348).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

SETIO: Prozessabbild übertragen

Beschreibung

Mit der Anweisung "SETIO" übertragen Sie die Daten. Der Quellbereich der Übertragung wid über den Parameter OUTPUTS definitert. Die Daten werden konsistent zum adressierten DP-Normslave / PROFINET IO-Device übertragen. Wenn Sie den Adressbereich des DP-Normslave / PROFINET IO-Device als Konsistenzbereich in einem Prozessabbild projektiert haben, werden die Daten ins ins Prozessabbild übertragen. "SETIO" ruft dabei die Anweisung "DPWR_DAT (Seite 2436)" auf.

Der Quellbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Komponente projektiert haben.

Bei einem DP-Normslave / PROFINET IO-Device mit modularem Aufbau bzw. mit mehreren Kennungen können Sie mit einem "SETIO"-Aufruf nur auf eine DP-Kennung / Komponente zugreifen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Hardware-ID des DP-Normslave / PRO-FINET IO-Device.
STATUS	Output	DWORD	E, A, M, D, L	Enthält die Fehlerinformation von "DPWR_DAT (Seite 2436)" in der Form DW#16#40xxxx00.
OUTPUTS	InOut	VARIANT	E, A, M, D, L	Quellbereich für die zu schreibenden Daten. Er muss genauso lang sein, wie Sie für den selektierten DP-Normslave / PRO-FINET IO-Device projektiert haben. Nur der Datentyp BYTE ist zulässig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Siehe auch: DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben (Seite 2436).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für GETIO & SETIO (Seite 2348).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Programmbeispiel für GETIO & SETIO

Einleitung

In dem folgenden Beispiel lesen Sie über GETIO die Känale eines digitalen Eingabemoduls aus und schreiben über SETIO neue Werte auf die Kanäle eines digitalen Ausgabemoduls. Anschließend lesen Sie die Änderungen über GETIO und über das digitale Eingabemodul aus.

Voraussetzungen

- Ein IO-Device / DP-Slave mit einem digitalen Eingabemodul und einem digitalen Ausgabemodul ist angelegt.

Hinweis

Mit SETIO ändern, mit GETIO die Änderungen auslesen

Das digitale Eingabemodul muss mit dem digitalen Ausgabemodul verbunden sein, damit die über SETIO geschriebenen Werte mit Hilfe von GETIO lesbar sind.

- Zur Ablage der Daten legen Sie in einem globalen Datenbaustein neun Variablen und die beiden Strukturen "readinputs", "writeOutputs" an.
  Über die Variable "id_Read" hinterlegen Sie die HW-Kennung des digitalen Eingabemoduls. Über die Variable "id_Write" hinterlegen Sie die HW-Kennung des digitalen Ausgabemoduls.

Hinweis

HW-Kennung finden

Die HW-Kennung finden Sie über "PLC-Variablen > Systemkonstanten". Z. B. für das digitale Eingabemodul suchen Sie nach "Local~<Name des DI-Moduls>", mit dem Datentyp "HW_Submodule". Die zugehörige Zelle "Wert" enthält die HW-Kennung.

SLI_gDB_GETIO

	Name	Data type	Start value
1	Static		
2	id_Read	HW_SUBMODULE	257
3	executeRead	Bool	false
4	readInputs	Struct	
5	Channel_0-7	Array[0..7] of Bool	
6	Channel_8-15	Array[0..7] of Bool	
7	Channel_16-23	Array[0..7] of Bool	
8	Channel_24-31	Array[0..7] of Bool	
9	statusRead	DWord	16#0
10	lengthRead	Int	0
11	doneRead	Bool	false
12	id_Write	HW_SUBMODULE	258
13	executeWrite	Bool	false
14	writeOutputs	Struct	
15	Channel_0-7	Array[0..7] of Bool	
16	Channel_8-15	Array[0..7] of Bool	
17	Channel_16-23	Array[0..7] of Bool	
18	Channel_24-31	Array[0..7] of Bool	
19	statusWrite	DWord	16#0
20	doneWrite	Bool	false

- Für die Datenweitergabe legen Sie die folgenden lokalen Variablen in einem FB "SLI_FB_GETIO_SETIO" an.

7	Static			
8	GETIO_Instance	GETIO		
9	SETIO_Instance	SETIO		
10	readBYTE	Struct		Non-retain
11	Channel_0-7	Byte	16#0	Non-retain
12	Channel_8-15	Byte	16#0	Non-retain
13	Channel_16-23	Byte	16#0	Non-retain
14	Channel_24-31	Byte	16#0	Non-retain
15	writeBYTE	Struct		Non-retain
16	Channel_0-7	Byte	16#0	Non-retain
17	Channel_8-15	Byte	16#0	Non-retain
18	Channel_16-23	Byte	16#0	Non-retain
19	Channel_24-31	Byte	16#0	Non-retain

- Zum Ändern der Kanalzustände setzen Sie folgende Bits (Kanal 1 und 2) in dem globalen Datenbaustein auf "TRUE":

14	writeOutputs	Struct	
15	Channel_0-7	Array[0..7] of Bool	
16	Channel_0-7[0]	Bool	false
17	Channel_0-7[1]	Bool	1
18	Channel_0-7[2]	Bool	1
19	Channel_0-7[3]	Bool	false
20	Channel_0-7[4]	Bool	false
21	Channel_0-7[5]	Bool	false
22	Channel_0-7[6]	Bool	false
23	Channel_0-7[7]	Bool	false

Parameter verschalten: FC "SLI_itemiseByte_gsIO"

Zur Weitergabe der Werte einer Bitfolge (0..7) legen Sie den FC "SLI_itemiseByte_gsIO" an.

Darin legen Sie die folgenden lokalen Variablen an.

SLI_itemiseByte_gsIO		
	Name	Data type
1	Input	
2	in0	Bool
3	in1	Bool
4	in2	Bool
5	in3	Bool
6	in4	Bool
7	in5	Bool
8	in6	Bool
9	in7	Bool
10	Output	
11	out0	Bool
12	out1	Bool
13	out2	Bool
14	out3	Bool
15	out4	Bool
16	out5	Bool
17	out6	Bool
18	out7	Bool

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) Verschaltungen an.

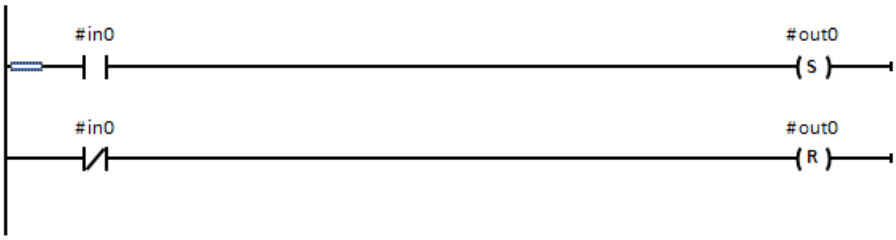

Parameter verschalten: FB "SLI_FB_GETIO_SETIO"

In dem FB "SLI_FB_GETIO_SETIO" legen Sie die folgenden Verschaltungen an. Auf dem IO-Device / DP-Slave rufen Sie den FB "SLI_FB_GETIO_SETIO" anschließend in einem Programmzyklus-OB (OB1) auf.

Netzwerk 1: Die Parameter der Anweisung "GETIO" verschalten Sie wie folgt.

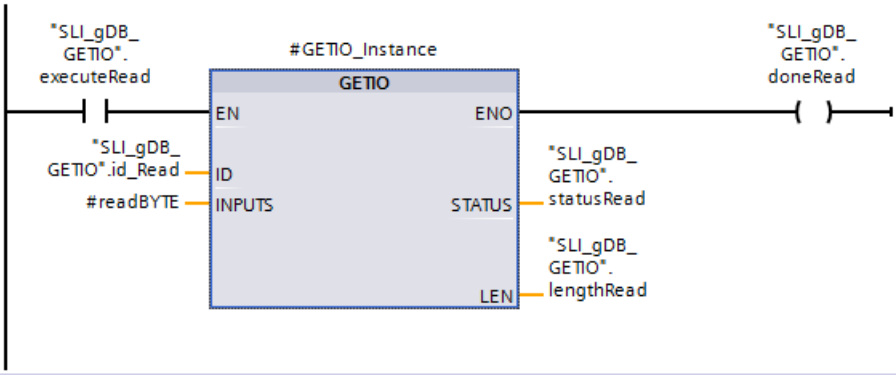

Netzwerk 2: Die Anweisung "GETIO" beenden Sie wie folgt.

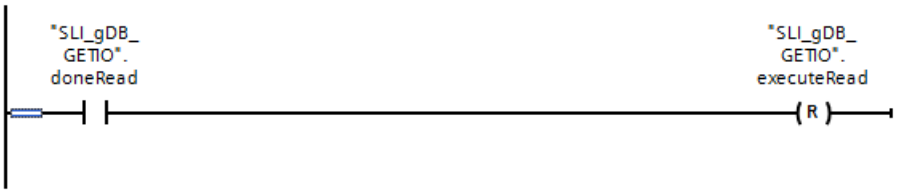

Netzwerk 3: Zur Umwandlung der Kanalwerte von GETIO legen Sie Verschaltungen nach folgendem Vorbild an. Dabei werden alle Ein- und Ausgänge (0..7) des FC "SLI_itemiseByte_gsIO" verschalten.

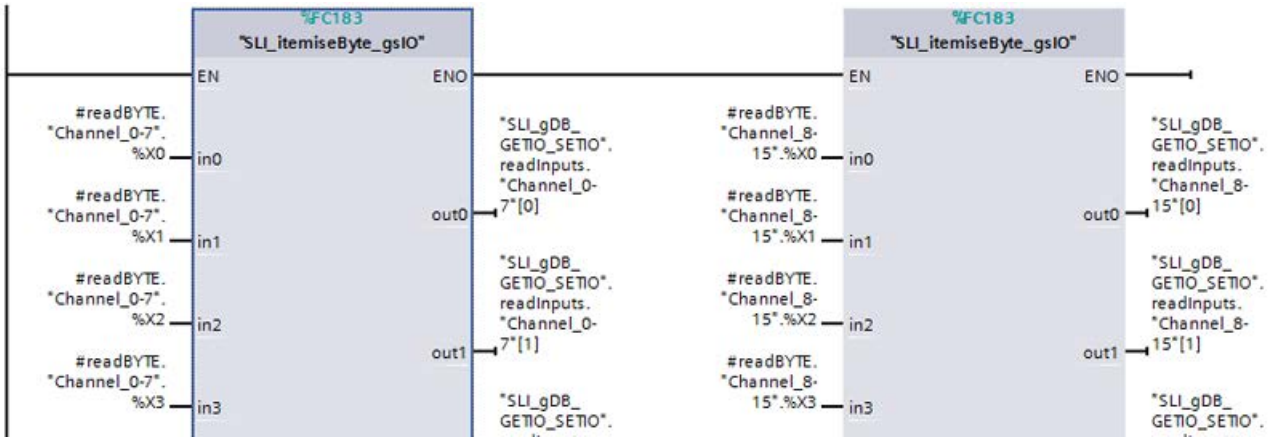

Netzwerk 4: Zur Umwandlung der Kanalwerte von SETIO legen Sie Verschaltungen nach folgendem Vorbild an. Dabei werden alle Ein- und Ausgänge (0..7) des FC "SLI_itemiseByte_gsIO" verschalten.

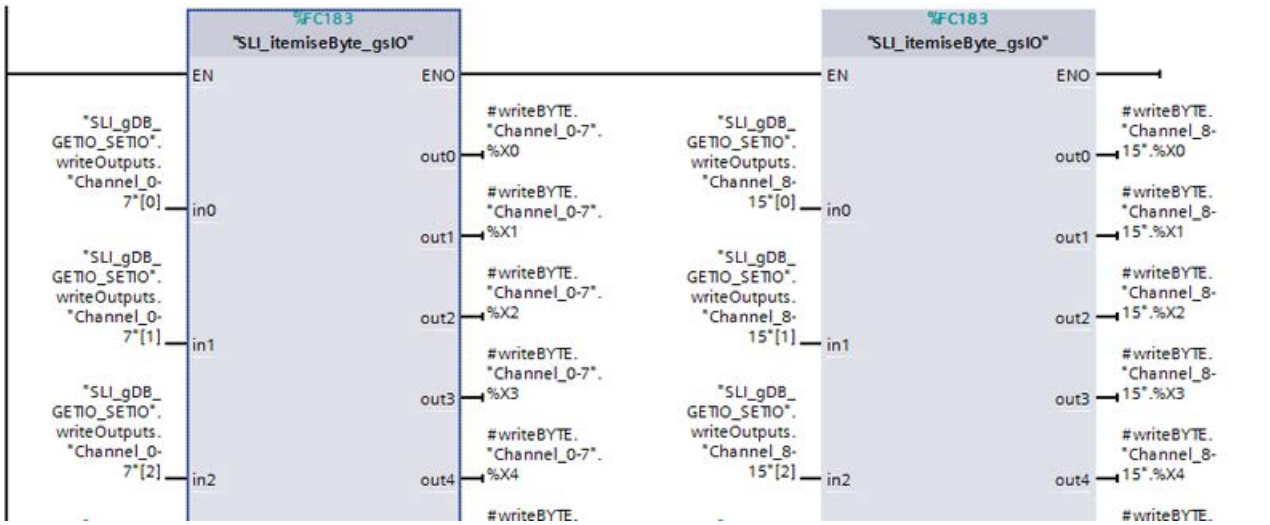

Netzwerk 5: Die Parameter der Anweisung "SETIO" verschalten Sie wie folgt.

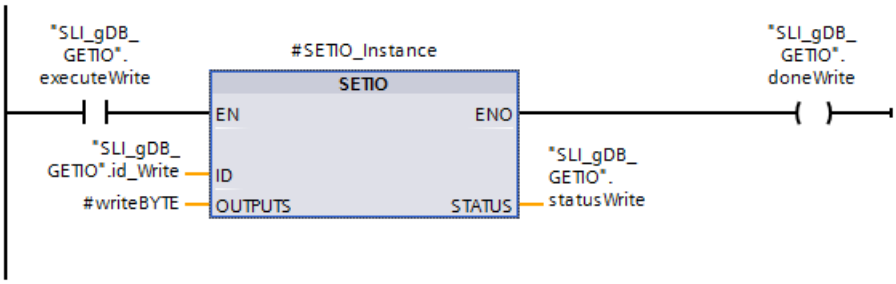

Netzwerk 6: Die Anweisung "SETIO" beenden Sie wie folgt.

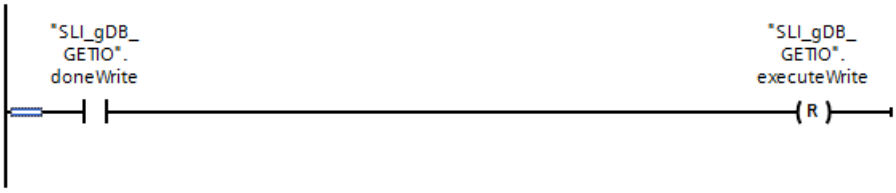

Ergebnis von Netzwerk 1 (GETIO)

Wenn der Eingangsparameter EN ("executeRead") den Signalzustand "TRUE" liefert, wird die Anweisung "GETIO" gestartet. Über den Eingangsparameter ID ("id_Read") ruft die Anweisung "GETIO" das Eingabemodul auf. Über den Eingangsparameter INPUTS ("#readBYTE") werden die Kanäle des Eingabemoduls ausgelesen. Die Länge der gelesenen Kanäle in BYTE wird über den Ausgangsparameter LEN ("lengthRead") festgehalten. Der Wert "2" weisst auf 16 Eingangskanäle hin.

Am Ausgangsparameter STATUS ("statusRead") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

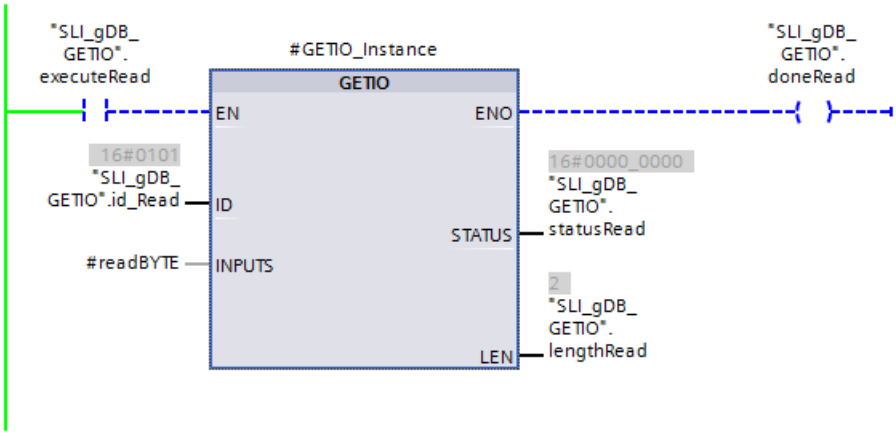

Ergebnis von Netzwerk 2 (GETIO)

Wenn die Verarbeitung der Anweisung "GETIO" durchgelaufen ist, liefert der Ausgangsparameter ENO ("doneRead") den Signalzustand "TRUE". Daraufhin setzt der Rücksetzer die Variable "executeRead" auf "FALSE" und beendet die Verarbeitung von GETIO.

Ergebnis von Netzwerk 3 (GETIO)

Über MOVE-Anweisungen und den FC "SLI_itemiseByte_gsIO" werden die Werte der Variablen von der Struktur "#readBYTE" an Variablen der Struktur "readInputs" übergeben. Dabei werden die Werte der Kanäle 0..7 und 8..15 von BYTE in Array of BOOL umgewandelt.

Zum besseren Verständnis sind die Variablen der Struktur "readInputs" als Klarnamen vertreten.

Ergebnis von Netzwerk 4 (SETIO)

Über MOVE-Anweisungen und den FC "SLI_itemiseByte_gsIO" werden die Werte der Variablen von der Struktur "writeOutputs" an Variablen der Struktur "#writeBYTE" übergeben. Dabei werden die Werte der Kanäle 0..7 und 8..15 von Array of BOOL in BYTE umgewandelt.

Zum besseren Verständnis sind die Variablen der Struktur "writeOutputs" als Klarnamen vertreten.

Ergebnis von Netzwerk 5 (SETIO)

Wenn der Eingangsparameter EN ("executeWrite") den Signalzustand "TRUE" liefert, wird die Anweisung "SETIO" gestartet. Über den Eingangsparameter ID ("id_Write") ruft die Anweisung "SETIO" das Ausgabemodul auf. Über den Eingangsparameter OUTPUTS ("#writeBYTE") werden auf die Kanäle des Ausgabemoduls neue Werte geschrieben.

Am Ausgangsparameter STATUS ("statusWrite") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

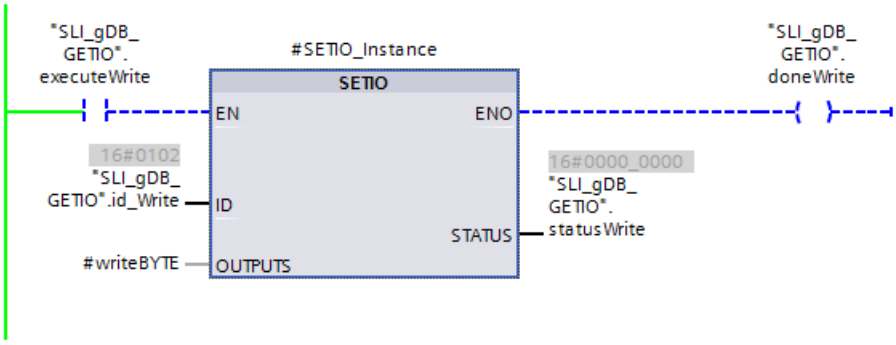

Ergebnis von Netzwerk 6 (SETIO)

Wenn die Verarbeitung der Anweisung "SETIO" durchgelaufen ist, liefert der Ausgangsparameter ENO ("doneWrite") den Signalzustand "TRUE". Daraufhin setzt der Rücksetzer die Variable "executeWrite" auf "FALSE" und beendet die Verarbeitung von SETIO.

Ergebnis einer erneuten Verarbeitung von GETIO

Wenn die Variable "executeRead" von Neuem auf "TRUE" gesetzt wird, zeigt die Struktur "readInputs" nach der Verarbeitung von GETIO die neuen Werte an den Kanälen des Eingabemoduls an. Die Kanäle 1 und 2 ("Channel_0-7[1]..[2]") liefern nun den Signalzustand "TRUE".

4	readInputs	Struct		
5	Channel_0-7	Array[0..7] of Bool		
6	Channel_0-7[0]	Bool	false	FALSE
7	Channel_0-7[1]	Bool	false	TRUE
8	Channel_0-7[2]	Bool	false	TRUE
9	Channel_0-7[3]	Bool	false	FALSE
10	Channel_0-7[4]	Bool	false	FALSE
11	Channel_0-7[5]	Bool	false	FALSE
12	Channel_0-7[6]	Bool	false	FALSE
13	Channel_0-7[7]	Bool	false	FALSE
14	Channel_8-15	Array[0..7] of Bool		
15	Channel_16-23	Array[0..7] of Bool		
16	Channel_24-31	Array[0..7] of Bool		

Hinweis

Signalzustand eines Kanals ändern

Wenn ein Kanal, bei dem der Signalzustand geändert werden soll, bereits mit einem Wert belegt wird (z. B. durch eine PLC-Variable), kann diese Vorbelegung eine Änderung durch SETIO unwirksam machen. Achten Sie darauf, dass der Signalzustand eines zu ändernden Kanals nicht bereits durch andere Faktoren bestimmt wird.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

GETIO: Prozessabbild lesen (Seite 2346)

SETIO: Prozessabbild übertragen (Seite 2347)

GETIO_PART: Prozessabbildbereich lesen

Beschreibung

Mit der Anweisung "GETIO_PART" lesen Sie einen zusammenhängenden Teil der Eingänge einer EA-Baugruppe konsistent aus. "GETIO_PART" ruft dabei die Anweisung "DPRD_DAT (Seite 2434)" auf.

Über den Eingangsparameter ID wählen Sie die EA-Baugruppe anhand der Hardware-Kennung aus.

Über die Parameter OFFSET und LEN legen Sie die auszulesende Teilmenge des Prozessabbildbereichs fest. Falls der mithilfe von OFFSET und LEN aufgespannte Eingangsbereich nicht komplett von der Baugruppe abgedeckt wird, liefert der Baustein den Fehlercode DW#16#40**80B7**00.

Die Länge des Zielbereichs muss größer als oder gleich der Anzahl der zu lesenden Byte sein:

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE. Die gelesenen Daten werden in den am Parameter INPUTS definierten Zielbereich geschrieben.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE. Der Parameter STATUS erhält die Fehlerinformation von "DPRD_DAT".
- Falls der Zielbereich größer ist als LEN, werden die ersten LEN Byte des Zielbereichs beschrieben. ERROR erhält den Wert FALSE.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO_PART":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe
OFFSET	Input	INT	E, A, M, D, L oder Konstante	Nummer des ersten zu lesenden Byte im Prozessabbild der Komponente (kleinstmöglicher Wert: 0).
LEN	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu lesenden Byte.
STATUS	Output	DWORD	E, A, M, D, L	Enthält die Fehlerinformation von "DPRD_DAT" in der Form DW#16#40xxxx00, falls ERROR = TRUE.
ERROR	Output	BOOL	E, A, M, D, L	Fehleranzeige: ERROR = TRUE, falls beim Aufruf von "DPRD_DAT" ein Fehler auftritt.
INPUTS	InOut	VARIANT	E, A, M, D, L	Zielbereich für die gelesenen Daten: Falls der Zielbereich größer ist als LEN, werden die ersten LEN Byte des Zielbereichs beschrieben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Siehe Parameter RET_VAL der Anweisung "DPRD_DAT (Seite 2434)".

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für GETIO_PART & SETIO_PART (Seite 2358).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

SETIO_PART: Prozessabbildbereich übertragen

Beschreibung

Mit der Anweisung "SETIO_PART" schreiben Sie die Daten aus dem durch OUTPUTS aufgespannten Quellbereich konsistent auf die Ausgänge einer EA-Baugruppe. "SETIO_PART" ruft dabei die Anweisung "DPWR_DAT (Seite 2436)" auf.

Über den Eingangsparameter ID wählen Sie die EA-Baugruppe anhand der Hardware-Kennung aus.
Über die Parameter OFFSET und LEN legen Sie die zu beschreibende Teilmenge des Prozessabbildbereichs der über ID adressierten Komponente fest. Falls der mithilfe von OFFSET und LEN aufgespannte Ausgangsbereich nicht komplett von der Baugruppe abgedeckt wird, liefert der Baustein den Fehlercode DW#16#40**80B7**00.

Die Länge des Zielbereichs muss größer als oder gleich der Anzahl der zu lesenden Byte sein:

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE, und STATUS erhält die Fehlerinformation von "DPWR_DAT".
- Falls der Quellbereich größer ist als LEN, werden die ersten LEN Byte aus OUTPUTS übertragen. ERROR erhält den Wert FALSE.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO_PART":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Hardware-Kennung der EA-Baugruppe.
OFFSET	Input	INT	E, A, M, D, L oder Konstante	Nummer des ersten zu beschreibenden Byte im Prozessabbild der Komponente (kleinstmöglicher Wert: 0).
LEN	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu schreibenden Byte.
STATUS	Output	DWORD	E, A, M, D, L	Enthält die Fehlerinformation von "DPWR_DAT" in der Form DW#16#40xxxx00, falls ERROR = TRUE.
ERROR	Output	BOOL	E, A, M, D, L	Fehleranzeige: ERROR = TRUE, falls beim Aufruf von "DPWR_DAT" ein Fehler auftritt.
OUTPUTS	InOut	VARIANT	E, A, M, D, L	Quellbereich für die zu schreibenden Daten: Falls der Quellbereich größer ist als LEN, werden die ersten LEN Byte aus OUTPUTS übertragen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS und ERROR

Siehe Anweisung "DPWR_DAT (Seite 2436)".

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für GETIO_PART & SETIO_PART (Seite 2358).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Programmbeispiel für GETIO_PART & SETIO_PART

Einleitung

In dem folgenden Beispiel lesen Sie über GETIO_PART die Känale eines digitalen Eingabemoduls aus und schreiben über SETIO_PART neue Werte auf einen Teil der Kanäle eines digitalen Ausgabemoduls. Anschließend lesen Sie die Änderungen über GETIO_PART und über das digitale Eingabemodul aus.

Voraussetzungen

- Ein IO-Device / DP-Slave mit einem digitalen Eingabemodul und einem digitalen Ausgabemodul ist angelegt.

Hinweis

Mit SETIO_PART ändern, mit GETIO_PART die Änderungen auslesen

Das digitale Eingabemodul muss mit dem digitalen Ausgabemodul verbunden sein, damit die über SETIO_PART geschriebenen Werte mit Hilfe von GETIO_PART lesbar sind.

- Zur Ablage der Daten legen Sie in einem globalen Datenbaustein 14 Variablen und die beiden Strukturen "readinputs", "writeOutputs" an.
  Über die Variable "id_Read" hinterlegen Sie die HW-Kennung des digitalen Eingabemoduls. Über die Variable "id_Write" hinterlegen Sie die HW-Kennung des digitalen Ausgabemoduls.

Hinweis

HW-Kennung finden

Die HW-Kennung finden Sie über "PLC-Variablen > Systemkonstanten". Z. B. für das digitale Eingabemodul suchen Sie nach "Local~<Name des DI-Moduls>", mit dem Datentyp "HW_Submodule". Die zugehörige Zelle "Wert" enthält die HW-Kennung.

SLI_gDB_GETIO_SETIO_PART

	Name	Data type	Start value
1	Static		
2	id_Read	HW_SUBMODULE	257
3	executeRead	Bool	false
4	firstByteRead	Int	0
5	maxLengthRead	Int	2
6	readInputs	Struct	
7	Channel_0-7	Array[0..7] of Bool	
8	Channel_8-15	Array[0..7] of Bool	
9	Channel_16-23	Array[0..7] of Bool	
10	Channel_24-31	Array[0..7] of Bool	
11	errorRead	Bool	false
12	statusRead	DWord	16#0
13	doneRead	Bool	false
14	id_Write	HW_SUBMODULE	258
15	executeWrite	Bool	false
16	firstByteWrite	Int	0
17	maxLengthWrite	Int	1
18	writeOutputs	Struct	
19	Channel_0-7	Array[0..7] of Bool	
20	Channel_8-15	Array[0..7] of Bool	
21	Channel_16-23	Array[0..7] of Bool	
22	Channel_24-31	Array[0..7] of Bool	
23	errorWrite	Bool	false
24	statusWrite	DWord	16#0
25	doneWrite	Bool	false

- Zum Ändern der Kanalzustände setzen Sie folgende Bits (für BYTE 0 und BYTE 1) in dem globalen Datenbaustein auf "TRUE":

14	writeOutputs	Struct	
15	Channel_0-7	Array[0..7] of Bool	
16	Channel_0-7[0]	Bool	false
17	Channel_0-7[1]	Bool	1
18	Channel_0-7[2]	Bool	1
19	Channel_0-7[3]	Bool	false
20	Channel_0-7[4]	Bool	false
21	Channel_0-7[5]	Bool	false
22	Channel_0-7[6]	Bool	false
23	Channel_0-7[7]	Bool	false
24	Channel_8-15	Array[0..7] of Bool	
25	Channel_8-15[0]	Bool	false
26	Channel_8-15[1]	Bool	1

- Für die Datenweitergabe legen Sie die folgenden lokalen Variablen in einem FB "SLI_FB_GETIO_SETIO_PART" an.

7	Static			
8	GETIO_PART_Instance	GETIO_PART		
9	SETIO_PART_Instance	SETIO_PART		
10	readBYTE	Struct		Non-retain
11	writeBYTE	Struct		Non-retain

Parameter verschalten: FC "SLI_itemiseByte_gpspIO"

Zur Weitergabe der Werte einer Bitfolge (0..7) legen Sie den FC "SLI_itemiseByte_gpspIO" an.

Darin legen Sie die folgenden lokalen Variablen an.

SLI_itemiseByte_gpspIO		
	Name	Data type
1	Input	
2	in0	Bool
3	in1	Bool
4	in2	Bool
5	in3	Bool
6	in4	Bool
7	in5	Bool
8	in6	Bool
9	in7	Bool
10	Output	
11	out0	Bool
12	out1	Bool
13	out2	Bool
14	out3	Bool
15	out4	Bool
16	out5	Bool
17	out6	Bool
18	out7	Bool

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) Verschaltungen an.

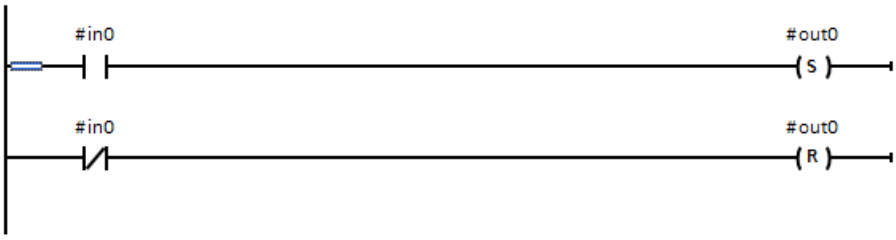

Parameter verschalten: FB "SLI_FB_GETIO_SETIO_PART"

In dem FB "SLI_FB_GETIO_SETIO_PART" legen Sie die folgenden Verschaltungen an. Auf dem IO-Device / DP-Slave rufen Sie den FB "SLI_FB_GETIO_SETIO_PART" anschließend in einem Programmzyklus-OB (OB1) auf.

Netzwerk 1: Die Parameter der Anweisung "GETIO_PART" verschalten Sie wie folgt.

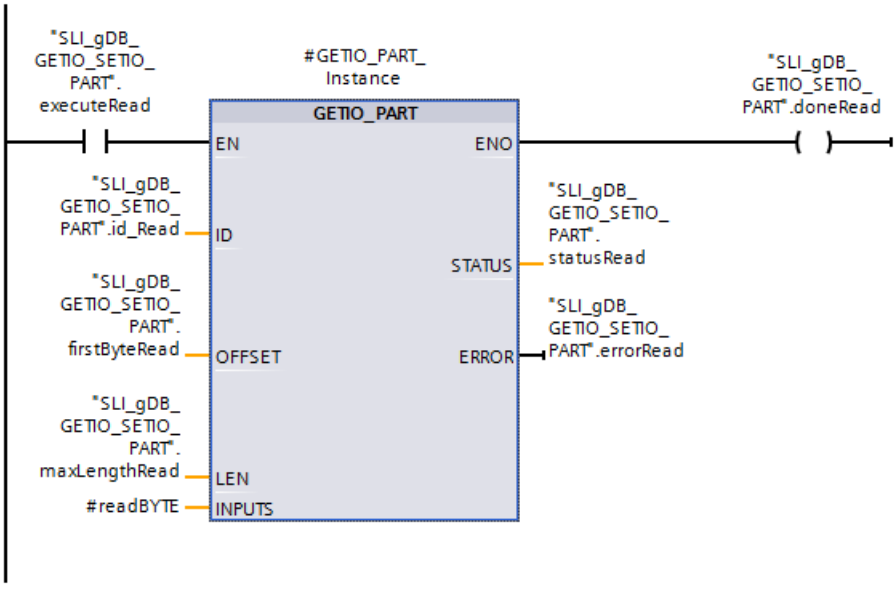

Netzwerk 2: Die Anweisung "GETIO_PART" beenden Sie wie folgt.

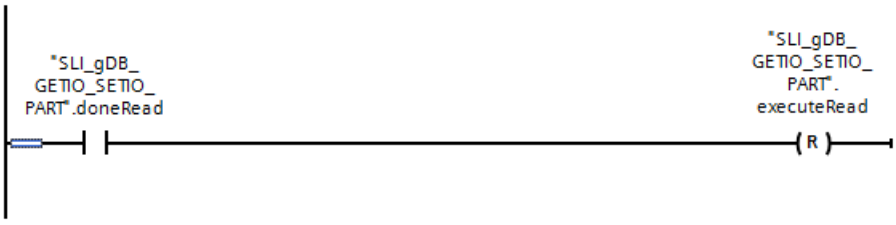

Netzwerk 3: Zur Umwandlung der Kanalwerte von GETIO_PART legen Sie Verschaltungen nach folgendem Vorbild an. Dabei werden alle Ein- und Ausgänge (0..7) des FC "SLI_itemiseByte_gpspIO" verschalten.

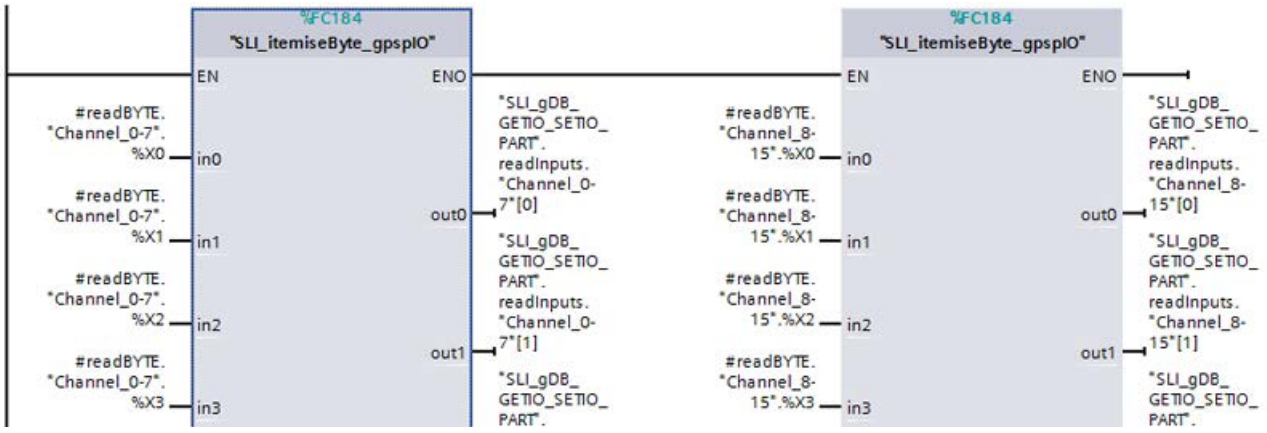

Netzwerk 4: Zur Umwandlung der Kanalwerte von SETIO_PART legen Sie Verschaltungen nach folgendem Vorbild an. Dabei werden alle Ein- und Ausgänge (0..7) des FC "SLI_itemiseByte_gpspIO" verschalten.

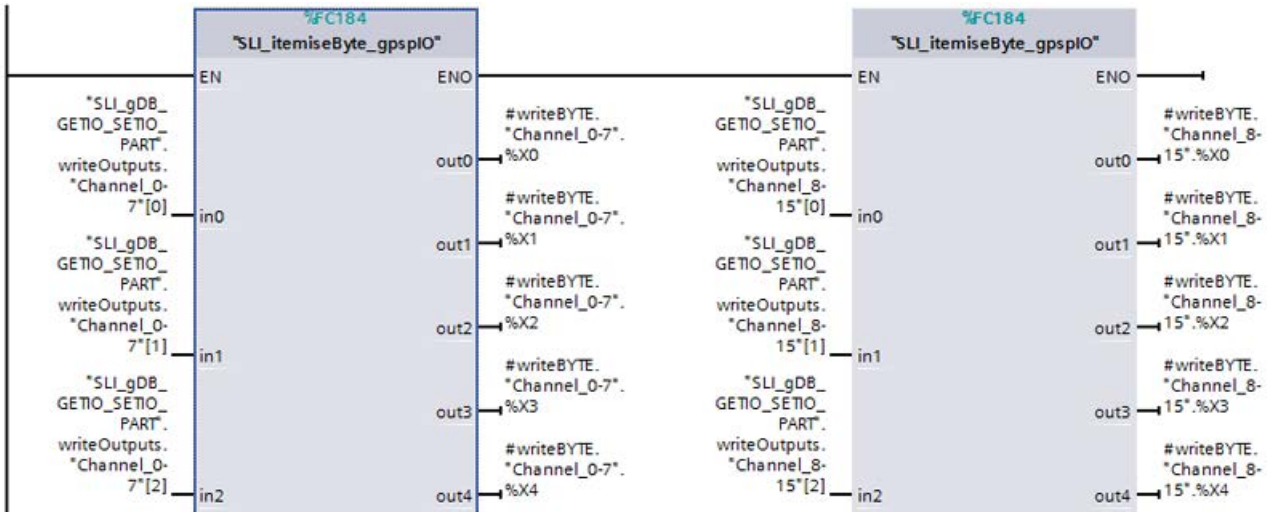

Netzwerk 5: Die Parameter der Anweisung "SETIO_PART" verschalten Sie wie folgt.

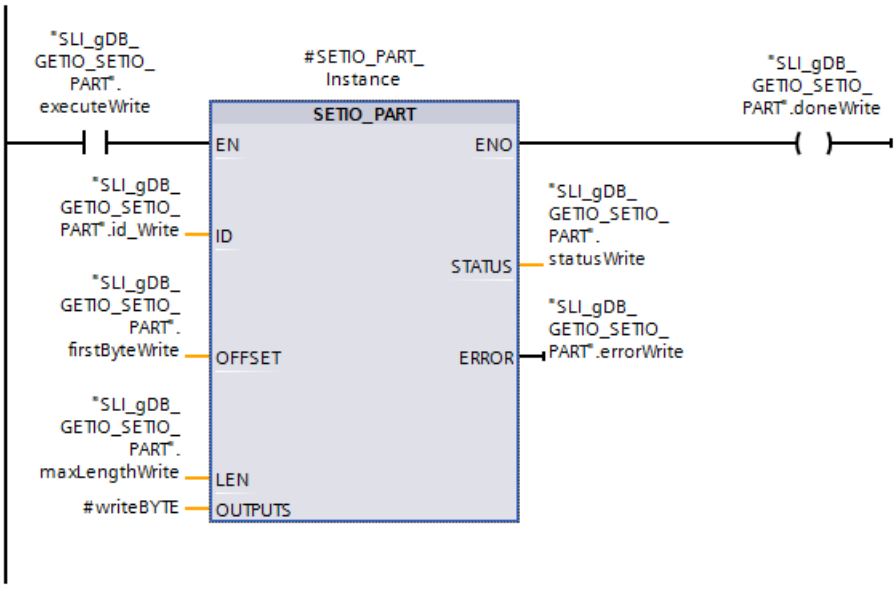

Netzwerk 6: Die Anweisung "SETIO_PART" beenden Sie wie folgt.

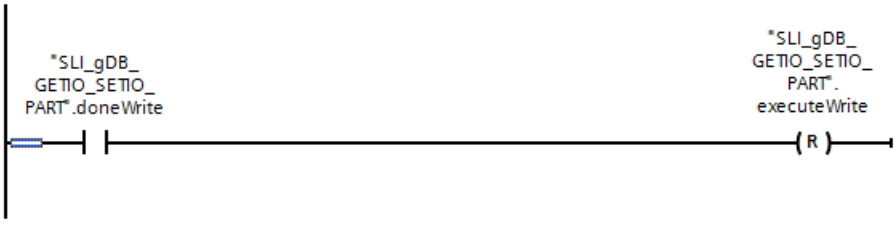

Ergebnis von Netzwerk 1 (GETIO_PART)

Wenn der Eingangsparameter EN ("executeRead") den Signalzustand "TRUE" liefert, wird die Anweisung "GETIO_PART" gestartet. Über den Eingangsparameter ID ("id_Read") ruft die Anweisung "GETIO_PART" das Eingabemodul auf. Über den Eingangsparameter INPUTS ("#readBYTE") werden die Kanäle des Eingabemoduls ausgelesen.

Durch den Eingangsparameter OFFSET ("firstByteRead") mit dem Wert "0" beginnt die Anweisung "GETIO_PART" ab dem "BYTE 0" die Kanäle des Eingabemoduls zu lesen. Entsprechend dem Wert "2" des Eingangsparameters LEN ("maxLengthRead") werden zwei BYTE (16 Kanäle) ausgelesen. In diesem Beispiel entspricht dieser Wert den Kanälen 0..15.

Am Ausgangsparameter STATUS ("statusRead") und ERROR ("errorRead") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

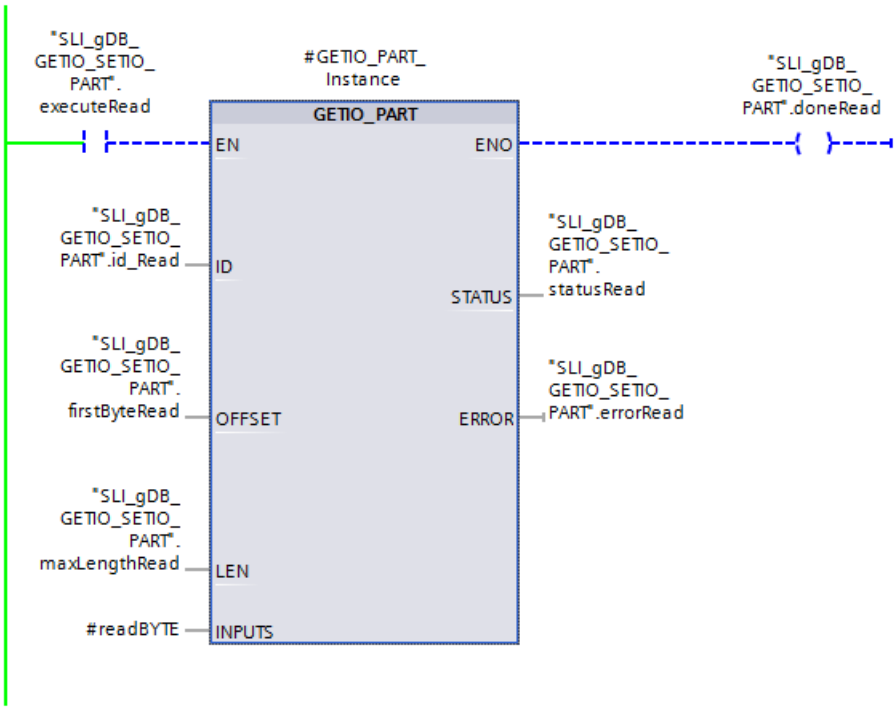

Ergebnis von Netzwerk 2 (GETIO_PART)

Wenn die Verarbeitung der Anweisung "GETIO_PART" durchgelaufen ist, liefert der Ausgangsparameter ENO ("doneRead") den Signalzustand "TRUE". Daraufhin setzt der Rücksetzer die Variable "executeRead" auf "FALSE" und beendet die Verarbeitung von GETIO_PART.

SLI_gDB_GETIO_SETIO_PART

	Name	Data type	Start value	Monitor value
1	Static			
2	id_Read	HW_SUBMODULE	257	16#0101
3	executeRead	Bool	false	FALSE
4	firstByteRead	Int	0	0
5	maxLengthRead	Int	2	2
6	readInputs	Struct		
7	errorRead	Bool	false	FALSE
8	statusRead	DWord	16#0	16#0000_0000
9	doneRead	Bool	false	FALSE

Ergebnis von Netzwerk 3 (GETIO_PART)

Über MOVE-Anweisungen und den FC "SLI_itemiseByte_gpspIO" werden die Werte der Variablen von der Struktur "#readBYTE" an Variablen der Struktur "readInputs" übergeben. Dabei werden die Werte der Kanäle 0..7 und 8..15 von BYTE in Array of BOOL umgewandelt.

Zum besseren Verständnis sind die Variablen der Struktur "readInputs" als Klarnamen vertreten.

Ergebnis von Netzwerk 4 (SETIO_PART)

Über MOVE-Anweisungen und den FC "SLI_itemiseByte_gpspIO" werden die Werte der Variablen von der Struktur "writeOutputs" an Variablen der Struktur "#writeBYTE" übergeben. Dabei werden die Werte der Kanäle 0..7 und 8..15 von Array of BOOL in BYTE umgewandelt.

Zum besseren Verständnis sind die Variablen der Struktur "writeOutputs" als Klarnamen vertreten.

Ergebnis von Netzwerk 5 (SETIO_PART)

Wenn der Eingangsparameter EN ("executeWrite") den Signalzustand "TRUE" liefert, wird die Anweisung "SETIO_PART" gestartet. Über den Eingangsparameter ID ("id_Write") ruft die Anweisung "SETIO_PART" das Ausgabemodul auf. Über den Eingangsparameter OUTPUTS ("#writeBYTE") werden auf die Kanäle des Ausgabemoduls neue Werte geschrieben.

Durch den Eingangsparameter OFFSET ("firstByteWrite") mit dem Wert "0" beginnt die Anweisung "SETIO_PART" ab dem "BYTE 0" auf die Kanäle des Ausgabemoduls zu schreiben. Entsprechend dem Wert "1" des Eingangsparameters LEN ("maxLengthWrite") wird nur ein BYTE (8 Kanäle) geschrieben. In diesem Beispiel entspricht dieser Wert den Kanälen 0..7. Der auf den Wert "TRUE" gesetzte Kanal 9 (siehe "writeOutputs", "Channel_8-15[1]") wird beim Schreiben auf die Kanäle des Ausgabemoduls also nicht berücksichtigt.

Am Ausgangsparameter STATUS ("statusWrite") und ERROR ("errorWrite") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

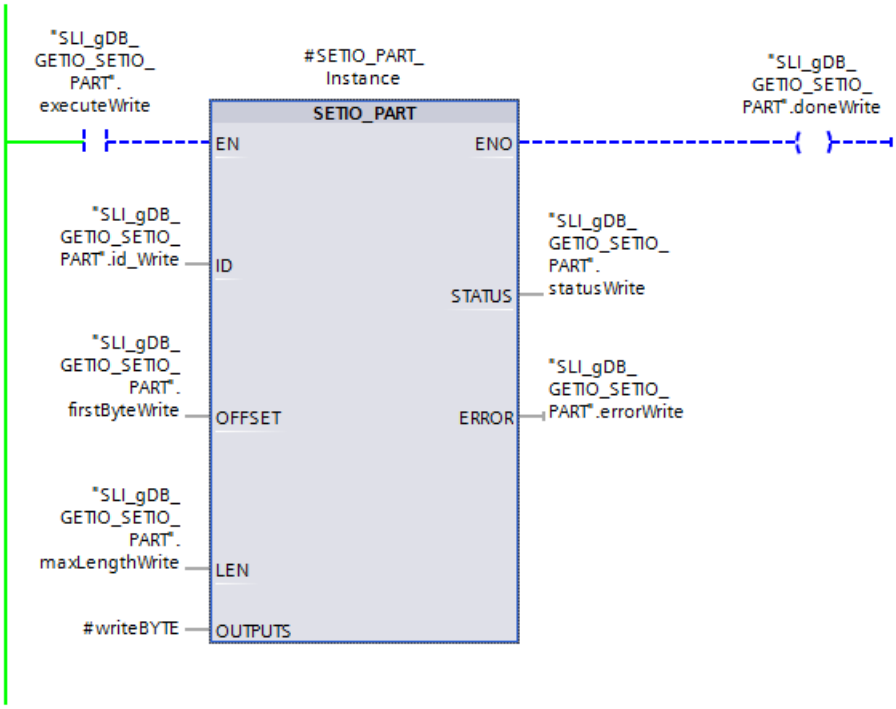

Ergebnis von Netzwerk 6 (SETIO_PART)

Wenn die Verarbeitung der Anweisung "SETIO_PART" durchgelaufen ist, liefert der Ausgangsparameter ENO ("doneWrite") den Signalzustand "TRUE". Daraufhin setzt der Rücksetzer die Variable "executeWrite" auf "FALSE" und beendet die Verarbeitung von SETIO_PART.

10	id_Write	HW_SUBMODULE	258	16#0102
11	executeWrite	Bool	false	FALSE
12	firstByteWrite	Int	0	0
13	maxLengthWrite	Int	1	1
14	writeOutputs	Struct		
15	errorWrite	Bool	false	FALSE
16	statusWrite	DWord	16#0	16#0000_0000
17	doneWrite	Bool	false	FALSE

Ergebnis einer erneuten Verarbeitung von GETIO_PART

Wenn die Variable "executeRead" von Neuem auf "TRUE" gesetzt wird, zeigt die Struktur "readInputs" nach der Verarbeitung von GETIO_PART die neuen Werte an den Kanälen des Eingabemoduls an. Entsprechend den Werten der Variablen "firstByteWrite" und "maxLengthWrite" betrifft eine Änderung nur die Kanäle 0..7. Die Kanäle 1 und 2 ("Channel_0-7[1]..[2]") liefern nun den Signalzustand "TRUE". Der Signalzustand des Kanal 9 ("Channel_8-15[1]") ist unverändert.

6	readInputs	Struct		
7	Channel_0-7	Array[0..7] of Bool		
8	Channel_0-7[0]	Bool	false	FALSE
9	Channel_0-7[1]	Bool	false	TRUE
10	Channel_0-7[2]	Bool	false	TRUE
11	Channel_0-7[3]	Bool	false	FALSE
12	Channel_0-7[4]	Bool	false	FALSE
13	Channel_0-7[5]	Bool	false	FALSE
14	Channel_0-7[6]	Bool	false	FALSE
15	Channel_0-7[7]	Bool	false	FALSE
16	Channel_8-15	Array[0..7] of Bool		
17	Channel_8-15[0]	Bool	false	FALSE
18	Channel_8-15[1]	Bool	false	FALSE

Hinweis

Signalzustand eines Kanals ändern

Wenn ein Kanal, bei dem der Signalzustand geändert werden soll, bereits mit einem Wert belegt wird (z. B. durch eine PLC-Variable), kann diese Vorbelegung eine Änderung durch SETIO_PART unwirksam machen. Achten Sie darauf, dass der Signalzustand eines zu ändernden Kanals nicht bereits durch andere Faktoren bestimmt wird.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

GETIO_PART: Prozessabbildbereich lesen (Seite 2355)

SETIO_PART: Prozessabbildbereich übertragen (Seite 2357)

RALRM: Alarm empfangen

Beschreibung RALRM

Beschreibung

Die Anweisung empfängt einen Alarm samt der zugehörigen Information von einer Peripheriebaugruppe (zentraler Aufbau) bzw. eines Moduls eines DP-Slaves oder eines PROFINET IO-Devices. Diese Informationen werden an seinen Ausgangsparametern zur Verfügung gestellt.

Die Information in den Ausgangsparametern enthält sowohl die Startinformation des aufgerufenen OB als auch Informationen aus der Alarmquelle.

Der Aufbau der Datenstruktur des Zielbereichs AINFO entspricht dem für einen zentralen Aufbau der Datenstruktur von PROFINET IO.

Rufen Sie "RALRM" nur innerhalb desjenigen Alarm-OBs auf, den das Betriebssystem der CPU aufgrund des zu untersuchenden Alarms aus der Peripherie gestartet hat.

Hinweis

Wenn Sie "RALRM" in einem OB aufrufen, dessen Startereignis kein Alarm aus der Peripherie ist, stellt Ihnen die Anweisung an ihren Ausgängen entsprechend weniger Information zur Verfügung.
Achten Sie darauf, dass Sie beim Aufruf von "RALRM" in verschiedenen OBs unterschiedliche Instanz-DBs verwenden. Falls Sie die aus einem "RALRM"-Aufruf resultierenden Daten außerhalb des zugehörigen Alarm-OB auswerten, benutzen Sie pro OB-Startereignis einen eigenen Instanz-DB.

Hinweis

Die Schnittstelle der Anweisung "RALRM" ist identisch mit der in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "RALRM".

Aufruf von RALRM

"RALRM" können Sie in drei verschiedenen Betriebsarten (Parameter MODE) aufrufen, die in der folgenden Tabelle erläutert werden.

MODE	RALRM ...
0	... zeigt die Alarm auslösende Komponente im Ausgangsparameter ID an und beschreibt den Ausgangsparameter NEW mit TRUE.
1	... beschreibt sämtliche Ausgangsparameter unabhängig von der Alarm auslösenden Komponente.
2	... prüft, ob die im Eingangsparameter F_ID angegebene Komponente den Alarm ausgelöst hat. • Falls nein, erhält NEW den Wert FALSE • Falls ja, erhält NEW den Wert TRUE, und alle anderen Ausgangsparameter werden beschrieben

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RALRM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe oder des Moduls Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Komponente oder des Interface in der Hardware-Konfiguration hinterlegt.
MLEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Länge der zu empfangenen Alarminformation in Bytes. Bei MLEN = 0 werden alle am Parameter AINFO spezifizierten Daten gelesen.
NEW	Output	BOOL	E, A, M, D, L	Ein neuer Alarm wurde empfangen.
STATUS (Seite 2370)	Output	DWORD	E, A, M, D, L	Fehlercode
ID	Output	HW_IO	E, A, M, D, L	Hardware-Kennung der Baugruppe oder des Moduls, von der ein Alarm empfangen wurde.
LEN	Output	UINT	E, A, M, D, L	Länge der empfangenen Alarminformation
TINFO (Seite 2374)	InOut	VARIANT	D, L	Zielbereich für OB-Startinformation und Verwaltungsinformation
AINFO (Seite 2393)	InOut	VARIANT	E, A, M, D, L	Zielbereich für Kopfinformation und Alarmzusatzinformation Sehen Sie für AINFO eine Länge von mindestens MLEN Byte vor.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie den Zielbereich (Seite 2400)TINFO bzw. AINFO zu kurz wählen, trägt RALRM nicht die gesamte Information ein.

Sonderfall für PC-Systeme: Zustand von Windows ermitteln

Folgendes Szenario hilft Ihnen, Informationen über den Zustand von Windows ihres PC-Systems zu erhalten:

- Rufen Sie die Anweisung "RALRM" (SFB54) im OB82 auf

Mit folgenden Eingangsparametern wird die Anweisung "RALRM" aufgerufen:

Parameter	Deklaration	Datentyp	Speicherbereich	Wert/Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	2
F_ID	Input	HW_IO	E, A, M, D, L oder Konstante	59 (Host Interface RID)
MLEN	UINT	UINT	E, A, M, D, L oder Konstante	0
TINFO	InOut	VARIANT	D, L	Adresse zu jedem Bereich von 54 Byte
AINFO	InOut	VARIANT	E, A, M, D, L	Adresse zu jedem Bereich von 54 Byte

Im Fall eines Software-Controllers können Sie über RALRM auch Information über die Zustand von Windows erhalten.
Das Byte 50 und 51 vom Parameter "AINFO" verwenden Sie, um detaillierte Information zum Zustand von Windows zu erhalten. Wenn das Byte 50 den Wert W#16#40 anzeigt, dann kann im Byte 51 die folgende detaillierte Information zum Zustand von Windows ausgelesen werden:

- 0xA1: Bluescreen
- 0xA2: Windows läuft
- 0xA3: Windows läuft nicht

Hinweis

Bei einem Systemabsturz reagiert Windows nicht mehr und zeigt einen Bluescreen an, der Angaben über den Grund und den Zeitpunkt des Systemabsturzes liefert. Standardmäßig bleibt der Bluescreen dauerhaft sichtbar.

Nach dem Systemfehler wird das System gestoppt und es kann folgendes Szenario eintreten:

- Windows sendet keine Signale mehr an die konfigurierte Software-CPU

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für RALRM (Seite 2401).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Parameter STATUS

Beschreibung

Der Ausgangsparameter STATUS enthält Fehlerinformationen. Wenn er als ARRAY[1...4] of BYTE interpretiert wird, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	• B#16#00, falls kein Fehler • Funktionskennung aus DPV1-PDU: Bei einem Fehler wird B#16#80 ausgegeben (bei einem Fehler beim Lesen eines Datensatzes B#16#DE und beim Schreiben eines Datensatzes B#16#DF). Falls kein DPV1-Protokollelement benutzt wird, wird B#16#C0 ausgegeben.
STATUS[2]	Error_Decode	Ort der Fehlerkennung
STATUS[3]	Error_Code_1	Fehlerkennung
STATUS[4]	Error_Code_2	Herstellerspezifische Erweiterung der Fehlerkennung

Feldelement STATUS[2]

STATUS[2] kann folgende Werte annehmen:

Error_Decode (B#16#...)	Quelle	Bedeutung
00 bis 7F	CPU	Kein Fehler oder keine Warnung
80	DPV1	Fehler nach IEC 61158-6
81 bis 8F	CPU	B#16#8x zeigt einen Fehler im x-ten Aufrufparameter der Anweisung an.
FE, FF	DP Profile	Profilspezifischer Fehler

Feldelement STATUS[3]

STATUS[3] kann folgende Werte annehmen:

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
00	00		Kein Fehler, keine Warnung
70	00	reserved, reject	Erstaufruf; keine Datensatzübertragung aktiv
	01	reserved, reject	Erstaufruf; Datensatzübertragung angestoßen
	02	reserved, reject	Zwischenaufruf; Datensatzübertragung ist bereits aktiv
80	81		Der Systemdatentyp am Parameter TINFO passt nicht zur Aufrufumgebung der Anweisung. Der verwendete Systemdatentyp muss zum Organisationsbaustein im Anwenderprogramm passen (Beispiel: für einen Verzögerungsalarm-OB benötigen Sie den Systemdatentyp TI_Delay).

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	90	reserved, pass	• Logische Anfangsadresse ungültig • Anweisungen RDREC und WRREC bei einer S7-1500-R/H-CPU: Kein Zugriff auf den Datensatz möglich, da sich die Partner-CPU im Betriebszustand STOP befindet.
	92	reserved, pass	Unzulässiger Typ bei VARIANT-Pointer
	93	reserved, pass	Die mittels ID bzw. F_ID adressierte DP-Komponente ist nicht konfiguriert.
	96		Der "RALRM (Seite 2367)" liefert weder die OB-Startinformation noch die Verwaltungsinformation noch die Kopfinformation noch die Alarmzusatzinformation. Bei den OBs 4x, 55, 56, 57, 82 und 83 können Sie mithilfe der Anweisung "DPNRM_DG (Seite 2450)" das aktuelle Diagnosetelegramm des betroffenen DP-Slaves asynchron lesen (Adressinformation aus OB-Startinformation).
	A0	read error	Negative Quittung beim Lesen von der Baugruppe
	A1	write error	Negative Quittung beim Schreiben zur Baugruppe
	A2	module failure	DP-Protokollfehler bei Layer 2 (z. B. Slaveausfall oder Busstörungen)
	A3	reserved, pass	Allgemeiner Kommunikationsfehler oder IO-Device / DP-Slave nicht erreichbar
	A4	reserved, pass	Kommunikation am PBUS+ gestört
	A5	reserved, pass	–
	A7	reserved, pass	DP-Slave oder Baugruppe ist beschäftigt (temporärer Fehler)
	A8	version conflict	DP-Slave oder Baugruppe meldet nicht passende Versionen
	A9	feature not supported	Funktion wird vom DP-Slave oder der Baugruppe nicht unterstützt
	AA bis AF	user specific	DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler seiner Anwendung. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.
	B0	invalid index	Baugruppe kennt den Datensatz nicht Datensatznummer ≥256 ist unzulässig
	B1	write length error	Fehler bei Längenangabe: • bei "RALRM (Seite 2367)": Längenfehler in AINFO (Seite 2393) • bei "RDREC (Seite 2306)": Längenfehler in MLEN • bei "WRREC (Seite 2326)": Längenfehler in LEN
	B2	invalid slot	• Der projektierte Steckplatz ist nicht belegt. • Für PROFINET IO und PROFIBUS DP: IO-Device / DP-Slave nicht erreichbar
	B3	type conflict	Ist-Baugruppentyp ungleich Soll-Baugruppentyp
	B4	invalid area	DP-Slave oder Baugruppe meldet einen Zugriff auf einen unzulässigen Bereich

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	B5	state conflict	DP-Slave oder Baugruppe ist nicht bereit
	B6	access denied	DP-Slave oder Baugruppe verweigert den Zugriff
	B7	invalid range	DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Werts
	B8	invalid parameter	DP-Slave oder Baugruppe meldet einen unzulässigen Parameter
	B9	invalid type	DP-Slave oder Baugruppe meldet einen unzulässigen Typ Bei "RDREC (Seite 2306)": Puffer zu klein (Lesen von Teilmengen ist nicht möglich) Bei "WRREC (Seite 2326)": Puffer zu klein (Schreiben von Teilmengen ist nicht möglich)
	BA		S7-1500-R/H-CPU: Während des Lesens oder Schreibens eines Datensatzes fällt die Primary-CPU aus. Die neue Primary-CPU quittiert den Auftrag mit 0x80BA. Der Datensatzauftrag wird auf der neuen Primary-CPU nicht wiederholt.
	BB bis BF	user specific	DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler beim Zugriff. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.
	C0	read constrain conflict	Bei "WRREC (Seite 2326)": Die Daten werden nur im STOP-Zustand der CPU geschrieben. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Die Daten schreiben nur online mit PG/PC. Bei "RDREC (Seite 2306)": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. Daher sind die Daten nur online mit PG/PC zu lesen.
	C1	write constrain conflict	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.
	C2	resource busy	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.
	C3	resource unavailable	Die benötigten Betriebsmittel sind momentan belegt.
	C4		Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.
	C5		DP-Slave oder Baugruppe nicht verfügbar.
	C6		Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs
	C7		Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	C8 bis CF		DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler zu seinen Ressourcen. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.
	Dx	user specific	DP-Slave-spezifisch. Siehe Beschreibung des DP-Slaves.
	Ex	user specific	Herstellerspezifisch. Bitte sehen Sie in der Dokumentation der Baugruppe nach.
81	00 bis FF		Fehler im ersten Aufrufparameter (bei "RALRM (Seite 2367)": MODE)
	00		Unzulässige Betriebsart
82	00 bis FF		Fehler im zweiten Aufrufparameter
:	:		:
88	00 bis FF		Fehler im achten Aufrufparameter (bei "RALRM (Seite 2367)": TINFO (Seite 2374))
	01		Syntaxkennung falsch
	23		Überschreitung des Mengengerüsts oder Zielbereich zu klein
	24		Bereichskennung falsch
	32		DB/DI-Nr. außerhalb des Anwenderbereichs
	3A		DB/DI-Nr. ist NULL bei Bereichskennung DB/DI oder angegebener DB/DI nicht vorhanden
89	00 bis FF		Fehler im neunten Aufrufparameter (bei "RALRM (Seite 2367)": AINFO (Seite 2393))
	01		Syntaxkennung falsch
	23		Überschreitung des Mengengerüsts oder Zielbereich zu klein
	24		Bereichskennung falsch
	32		DB/DI-Nr. außerhalb des Anwenderbereichs
	3A		DB/DI-Nr. ist NULL bei Bereichskennung DB/DI oder angegebener DB/DI nicht vorhanden
8A	00 bis FF		Fehler im 10. Aufrufparameter
:	:		:
8F	00 bis FF		Fehler im 15. Aufrufparameter
FE, FF	00 bis FF		Profilspezifischer Fehler

Parameter TINFO

Datenstruktur des Zielbereichs TINFO

Die Datenstruktur des Zielbereichs TINFO enthält die Startinformationen des Organisationsbausteins, in dem "RALRM" aktuell aufgerufen wurde.

Der Zielbereich TINFO kann die Startinformation mit Standardzugriff oder optimiertem Zugriff enthalten. Das Format der Startinformationen im Zielbereich TINFO muss immer den Startinformationen des entsprechenden Organisationsbausteins entsprechen.

- Die Startinformationen eines OB mit Standardzugriff liegen in der Bausteinschnittstelle immer in den ersten 20 Byte des Abschnitts "Temp". Für diese verwenden Sie die Datenstruktur "TI_Classic".
- Die Startinformationen eines OB mit optimiertem Zugriff werden im Abschnitt "Input" abgelegt. Die Startinformationen werden in für den OB spezifischen Datenstrukturen abgelegt. Verwenden Sie jeweils die für den OB spezifische Datenstruktur.

Durch die Umstellung des Bausteinzugriffs (standard/optimiert) ändert sich auch die Bausteinschnittstelle.

Die folgende Tabelle enthält eine Übersicht der Datenstrukturen, die in Abhängigkeit von dem Organisationsbaustein am Parameter TINFO verwendet werden.

Name der Datenstruktur		S7-1200 CPUs ab Version	S7-1500 CPUs ab Version	Datenstruktur verwendet für:
Datenstruktur für Organisationsbausteine mit Standardzugriff				
	TI_Classic	-	V1	Organisationsbausteine ohne optimierten Bausteinzugriff.
Datenstruktur für Organisationsbausteine mit optimiertem Bausteinzugriff				

Name der Datenstruktur		S7-1200 CPUs ab Version	S7-1500 CPUs ab Version	Datenstruktur verwendet für:
	TI_ProgramCycle	V2	V1	Zyklus-OB (Program cycle)
	TI_Startup	V2	V1	Anlauf-OB (Startup)
	TI_Delay	V2	V1	Verzögerungsalarm-OB (Time delay interrupt)
	TI_Cyclic	V2	V1	Weckalarm-OB (Cyclic interrupt)
	TI_HWInterrupt	V2	V1	Prozessalarm-OB (Hardware interrupt)
	TI_HWInterrupt_Extended	-	V2.0	Prozessalarm-OB (Hardware interrupt)
	TI_TimeError	V2	V1	Zeitfehler-OB (Time error interrupt)
	TI_DiagnosticInterrupt	V2	V1	Diagnosealarm-OB (Diagnostic error interrupt)
	TI_PlugPullModule	V2	V1	Ziehen/Stecken-OB (Pull or plug of modules)
	TI_StationFailure	V2	V1	Baugruppenträgerausfall-OB (Rack or station failure)
	TI_ProgIOAccessError	V2	V1	• Programmierfehler-OB (Programming error) • Peripheriezugriffsfehler-OB (IO access error)
	TI_TimeOfDay	V2	V1	Uhrzeitalarm-OB (Time of day)
	TI_SynchCycle	-	V1	Taktsynchronalarm-OB (Synchronous Cycle)
	TI_Submodule	V2	V1	• Statusalarm-OB (Status) • Update-Alarm OB (Update) • OB für hersteller- bzw. profilspezifischen Alarm (Profile)
	TI_CPURedundancyError_V2	-	V2.6	CPU-Redundanzfehler-OB

Datenstruktur für Organisationsbausteine mit Standardzugriff

Die Datenstruktur TI_Classic hat den folgenden Aufbau:

Parameter	Datentyp	Byte	Beschreibung
TI_Classic			Datenstruktur für Organisationsbausteine ohne optimierten Bausteinzugriff.

Parameter	Datentyp	Byte	Beschreibung
Byte 0 bis 19: Startinformation des OB, in dem "RALRM" aufgerufen wurde.*			
EV_CLASS	BYTE	0	Ereignisklasse Beispiel OB1: • Bits 0-3: Kennung des Ereignis (1=kommendes Ereignis) • Bits 4-7: Ereignisklasse (1=Ereignisklasse 1)
EV_NUM	BYTE	1	Ereignisnummer (abhängig von OB-Typ) Beispiel OB1 (SCAN_1): • SCAN_1 = 1 bei Erstaufruf • SCAN_1 = 3 bei allen weiteren Aufrufen
PRIORITY	BYTE	2	Prioritätsklasse
NUM	BYTE	3	OB-Nummer
TYP2_3	BYTE	4	Zusatzinformation Abhängig von dem verwendeten OB-Typ werden in den BYTES "TYP2_3" und "TYP1" unterschiedliche Informationen abgelegt. Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben. Beispiel (OB1): • TYP2_3: OB1_RESERVED_1 (Reserviert) • TYP1: OB1_RESERVED_2 (Reserviert)
TYP1	BYTE	5	
ZI1	WORD	6 bis 7	Zusatzinformation Abhängig von dem verwendeten OB-Typ werden in "ZI1" unterschiedliche Informationen abgelegt. Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben. Beispiel (OB1): • ZI1: OB1_PREV_CYCLE (Laufzeit des vorherigen Zyklus in ms)
ZI2_3	DWORD	8 bis 11	Zusatzinformation Abhängig von dem verwendeten OB-Typ werden in "ZI2_3" unterschiedliche Informationen abgelegt. Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben. Beispiel (OB1): • ZI2: OB1_MIN_CYCLE (minimale Zykluszeit (ms) seit dem letzten Anlauf) • ZI3: OB1_MAX_CYCLE (maximale Zykluszeit (ms) seit dem letzten Anlauf)
OB_DATE_TIME	DATE_AND_TIME (DT)	12 bis 19	Datum und Uhrzeit, zu denen der OB angefordert wurde.
Byte 20 und 21: Adressinformationen			
address	WORD	20 und 21	Adressinformationen wie bei S7-300/400 CPUs: • Bei zentralem Aufbau die Baugruppenträgernummer (0-31):

Parameter	Datentyp	Byte	Beschreibung
			Bit: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 0 — Baugruppenträgernummer • Bei dezentralem Aufbau mit PROFIBUS DP: – DP-Mastersystem-ID (1-31) – Stationsnummer (0-127). Bit: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 0 — DP-Mastersystem-ID — Stationsnummer • Bei dezentralem Aufbau mit PROFINET IO: – Die letzten beiden Stellen der PROFINET IO-System-ID (0-15), um die vollständige PROFINET IO-System-ID zu erhalten, müssen Sie 100 (dezimal) dazu addieren – Stationsnummer (0-2047). Bit: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 1 — IO-System-ID — Stationsnummer
Byte 22 bis 31: Verwaltungsinformation			
slv_prfl	BYTE	22	Slaveprofil wie bei S7-300/400 CPUs: • Bei zentralem Aufbau: 0 (Datensatz 0 oder Datensatz 1) • Bei dezentralem Aufbau: – Bit 0 bis 3: Slavetyp - 0000: DP (Struktur Datensatz 0) - 0001: DPS7 (Struktur Datensatz 0 oder Datensatz 1) - 0010: DPS7 V1 (Struktur Datensatz 0 oder Datensatz 1) - 0011: DPV1 (Struktur gemäß PROFIBUS DP Standard) - 0100 – 0111: reserviert - 1000: PROFINET IO (Struktur gemäß PROFINET IO Standard) - ab 1001: reserviert – Bit 4 bis 7: Profiltyp (reserviert)
intr_type	BYTE	23	Alarminfotyp wie bei S7-300/400 CPUs: • Bei zentralem Aufbau: 0 • Bei dezentralem Aufbau: – Bit 0 bis 3: Alarminfotyp - 0000: Alarm kommt von einer projektierten dezentralen Baugruppe - 0001: Alarm eines Nicht-DPV1-Slaves / Nicht-IO-Device oder eines nicht projektierten Steckplatzes - 0010: In der CPU erzeugter Alarm - ab 0011: reserviert

Parameter	Datentyp	Byte	Beschreibung
			– Bit 4 bis 7: Strukturversion - 0000: Initial - ab 0001: reserviert
flags1	BYTE	24	Flags der PROFIBUS DP-Master-Anschaltung/PROFINET IO-Controller-Anschaltung wie bei S7-300/400 CPUs: • Bei zentralem Aufbau: 0 • Bei dezentralem Aufbau: – Bit 0 = 0: Alarm von einer integrierten Anschaltung (PROFINET IO oder PROFIBUS DP) – Bit 0 = 1: Alarm von einer externen Anschaltung (PROFINET IO oder PROFIBUS DP) – Bit 1 bis 7: reserviert
flags2	BYTE	25	Flags der PROFIBUS DP-Master-Anschaltung/PROFINET IO-Controller-Anschaltung wie bei S7-300/400 CPUs: • Bei zentralem Aufbau: 0 • Bei dezentralem Aufbau mit PROFIBUS DP: – Bit 0: EXT_DIAG_FLAG aus dem Diagnosetelegramm bzw. 0, falls dieses Bit beim Alarm nicht vorliegt. Wenn der DP-Slave gestört ist, beträgt das Bit 1. – Bit 1 bis 7: reserviert • Bei dezentralem Aufbau mit PROFINET IO: – Bit 0: ARDiagnosisstate bzw. 0, falls keine Information beim Alarm vorliegt. Wenn das IO-Device gestört ist, beträgt das Bit ist 1. – Bit 1 bis 7: reserviert
id	UINT	26 und 27	Verwaltungsinformation • Bei zentralem Aufbau: 0 • Bei dezentralem Aufbau mit PROFIBUS DP: PROFIBUS Identnummer als eindeutige Identifikation des PROFIBUS DP-Slave • Bei dezentralem Aufbau mit PROFINET IO: PROFINET IO-Device Identnummer als eindeutige Identifikation des PROFINET IO-Devices
manufactu-rer	UINT	28 und 29	Herstellerkennung (nur bei dezentralem Aufbau mit PROFINET IO).
instance	UINT	30 und 31	Identnummer der Instanz (nur bei dezentralem Aufbau mit PROFINET IO).
*Die Startinformationen sind abhängig von dem verwendeten OB. Die Startinformationen des jeweiligen OB-Typs entnehmen Sie der Schnittstelle oder der Dokumentation des OBs.			

Datenstrukturen für Organisationsbausteine mit optimiertem Bausteinzugriff

Die Datenstrukturen für Organisationsbausteine mit optimiertem Bausteinzugriff haben den folgenden Aufbau:

- Byte 0 bis 3: Format der Startinformationen, Klasse und Nummer des aufgerufenen OBs (gleicher Aufbau für alle Datenstrukturen).
- Byte 4 bis 19: Optimierte Startinformation (Aufbau abhängig von dem OB-Typ). Die Daten in Byte 4 bis 19 entsprechen in Aufbau und Inhalt der entsprechenden OB-Schnittstelle.
- Byte 20 bis 31: Bei bestimmten OBs zusätzlich die Adress- und Verwaltungsinformation. Die Daten in Byte 20 bis 31 entsprechen den Daten der Bytes 20 bis 31 der Datenstruktur TI_Classic.

In den folgenden Tabellen ist der Aufbau der Datenstrukturen beschrieben.

Tabelle 4-15 **Zyklus-OB: Datenstruktur TI_ProgramCycle**

Parameter		Datentyp	Byte	Beschreibung
TI_ProgramCycle				Datenstruktur für Zyklus-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=1)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	Initial_Call	BOOL	4	=TRUE im ersten Aufruf dieses OB bei: • Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
	Remanence	BOOL	5	=TRUE, falls remanente Daten verfügbar sind.

Tabelle 4-16 **Anlauf-OB: Datenstruktur TI_Startup**

Parameter		Datentyp	Byte	Beschreibung
TI_Startup				Datenstruktur für Anlauf-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=100)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	LostRetentive	BOOL	4	=TRUE, falls die Inhalte remanenter Datenbereiche verloren gegangen sind.
	LostRTC	BOOL	5	=TRUE, falls die Uhrzeit der Echtzeituhr verloren gegangen ist.

Tabelle 4-17 **Verzögerungsalarm-OB: Datenstruktur TI_Delay**

Parameter		Datentyp	Byte	Beschreibung
TI_Delay				Datenstruktur für Verzögerungsalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=20)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	Sign	WORD	4	Anwenderkennung: Eingangsparameter SIGN aus dem Aufruf der Anweisung "SRT_DINT (Seite 2570)".

Tabelle 4-18 **Weckalarm-OB: Datenstruktur TI_Cyclic**

Parameter		Datentyp	Byte	Beschreibung
TI_Cyclic				Datenstruktur für Weckalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=30)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	Initial_Call	BOOL	4	=TRUE im ersten Aufruf dieses OB • Beim Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
	Event_Count	INT	6	Anzahl der verworfenen Startereignisse seit dem letzten Start dieses OBs.

Tabelle 4-19 **Prozessalarm-OB: Datenstruktur TI_HWInterrupt**

Parameter		Datentyp	Byte	Beschreibung
TI_HWInterrupt				Datenstruktur für Prozessalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=40)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	LADDR	HW_IO	4	HW-Kennung des Moduls, das den Prozessalarm auslöst.
	USI	WORD	6	Kennung für zukünftige Erweiterungen (nicht anwenderrelevant).
	IChannel	USINT	8	Nummer des Kanals, der den Prozessalarm auslöst.
	EventType	BYTE	9	Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke). Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen.
	address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
	slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
	intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
	flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
	flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
	id	UNIT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
	manufacturer	UNIT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
	instance	UNIT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-20 **Prozessalarm-OB: Datenstruktur TI_HWInterrupt_Extended**

Parameter		Datentyp	Byte	Beschreibung
TI_HWInterrupt_Extended				Datenstruktur für Prozessalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=40)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	LADDR	HW_IO	4	HW-Kennung des Moduls, das den Prozessalarm auslöst.
	USI	WORD	6	Kennung für zukünftige Erweiterungen (nicht anwenderrelevant).
	IChannel	USINT	8	Nummer des Kanals, der den Prozessalarm auslöst.
	EventType	BYTE	9	Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke). Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen.
	PointAddr	DWORD	12	• Bei Digitalbaugruppen: Bitfeld mit denjenigen Eingängen auf der Baugruppe, die den Prozessalarm ausgelöst haben • Bei Analogbaugruppen: Bitfeld mit der Information, welcher Kanal welche Grenze überschritten hat • Bei CPs oder IMs: Alarmzustand der Baugruppe (nicht anwenderrelevant)
	address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
	slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
	intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
	flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
	flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
	id	UNIT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
	manufacturer	UNIT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
	instance	UNIT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-21 **Zeitfehler-OB: Datenstruktur TI_TimeError**

Parameter		Datentyp	Byte	Beschreibung
TI_TimeError				Datenstruktur für Zeitfehler-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=80)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	Csg_OBnr	OB_ANY	4	Nummer des OB, der bearbeitet wurde, als der Zeitfehler auftrat.
	Fault_ID	BYTE	7	Fehlercode. Mögliche Werte: • B#16#01: Zykluszeit überschritten • B#16#02: Der angeforderte OB ist noch in Bearbeitung. • B#16#05: Abgelaufener Uhrzeitalarm durch Uhrzeitsprung. • B#16#06: Abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN nach HALT. • B#16#07: Überlauf des OB-Anforderungspuffers für die aktuelle Prioritätsklasse. • B#16#08: Taktsynchronalarm-Zeitfehler. • B#16#09: Alarmverlust durch zu hohe Alarmlast • B#16#0B: Technologiesynchronalarm-Zeitfehler
	Csg_Prio	UNIT	8	Priorität des OB, der bearbeitet wurde, als der Zeitfehler auftrat.

Tabelle 4-22 **Diagnosealarm-OB: Datenstruktur TI_DiagnosticInterrupt**

Parameter		Datentyp	Byte	Beschreibung
TI_DiagnosticInterrupt				Datenstruktur für Diagnosealarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=82)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	LADDR	HW_ANY	4	Hardware-Kennung des Hardware-Objekts, das den Diagnosealarm ausgelöst hat.
	IO_State	WORD	6	Zustand des Hardware-Objekts: • Bit 0: Gut • Bit 1: Deaktiviert • Bit 2: Wartungsbedarf • Bit 3: Wartungsanforderung • Bit 4: Fehler • Bit 5: Nicht erreichbar • Bit 6: Qualifiziert • Bit 7: Nicht verfügbar
	Channel	UINT	8	Kanalnummer
	MultiError	BOOL	10	=TRUE, falls mehr als ein Fehler vorliegt.
	address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
	slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
	intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
	flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
	flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
	id	UINT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
	manufacturer	UINT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
	instance	UINT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-23 **Ziehen/Stecken-OB: Datenstruktur** TI_PlugPullModule

Parameter	Datentyp	Byte	Beschreibung
TI_PlugPullModule			Datenstruktur für Ziehen/Stecken-OB
SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
OB_Class	USINT	1	OB-Klasse (=83)
OB_Nr	UINT	2	OB-Nummer (1...32767).
LADDR	HW_IO	4	Hardware-Kennung des betroffenen Moduls bzw. Submoduls
Event_Class	BYTE	6	• B#16#38: (Sub-) Modul gesteckt • B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar
Fault_ID	BYTE	7	Fehlercode Über den Fehlercode wird ausgegeben, welches Ereignis den Start des Ziehen/Stecken-OB verursacht hat. • Bei Event_Class = B#16#38 - (Sub-) Modul bzw. Baugruppe gesteckt: – B#16#54: Submodul gesteckt und entspricht parametriertem Submodul – B#16#55: Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul – B#16#56: Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung – B#16#57: Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung – B#16#58: Submodul Zugriffsfehler beseitigt – B#16#61: Baugruppe gesteckt • Bei Event_Class = B#16#39 - (Sub-) Modul bzw. Baugruppe gezogen bzw. nicht ansprechbar: – B#16#51: Modul gezogen – B#16#54: Submodul gezogen – B#16#61: Baugruppe gezogen bzw. nicht ansprechbar
address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
id	UINT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
manufacturer	UINT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
instance	UINT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-24 **Baugruppenträgerfehler-OB: Datenstruktur TI_StationFailure**

Parameter	Datentyp	Byte	Beschreibung
TI_StationFailure			Datenstruktur für Baugruppenträgerfehler-OB
SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
OB_Class	USINT	1	OB-Klasse (=86)
OB_Nr	UINT	2	OB-Nummer (1...32767).
LADDR	HW_Device	4	Hardware-Kennung des fehlerhaften Hardware-Objekts.
Event_Class	BYTE	6	• B#16#32: Aktivierung einer Station mit der Anweisung "D_ACT_DP" • B#16#33: Deaktivierung einer Station mit der Anweisung "D_ACT_DP" • B#16#38: gehendes Ereignis • B#16#39: kommendes Ereignis
Fault_ID	BYTE	7	Fehlercode Über den Fehlercode wird ausgegeben, welches Ereignis den Start des Baugruppenträgerfehler-OB verursacht hat. • Event_Class = B#16#39, FAULT_ID = C3: Dezentrale Peripherie: Ausfall eines DP-Mastersystems • Event_Class = B#16#38, FAULT_ID = C4: Ausfall einer DP-Station • Event_Class = B#16#39, FAULT_ID = CA: PROFINET IO-Systemausfall • Event_Class = B#16#39/38, FAULT_ID = CB: PROFINET IO-Stationsausfall/ Stationswiederkehr • Event_Class = B#16#38, FAULT_ID = CC: PROFINET IO-Stationswiederkehr mit Störung oder Wartung • Event_Class = B#16#38, FAULT_ID = CD: PROFINET IO-Stationswiederkehr, Sollausbau weicht von Istausbau ab • Event_Class = B#16#38, FAULT_ID = CE: PROFINET IO-Stationswiederkehr, Fehler bei der Baugruppenparametrierung • Event_Class = B#16#32/33, FAULT_ID = CF: Aktivierung/Deaktivierung eines PROFINET IO-DEvices mit der Anweisung "D_ACT_DP" • Event_Class = B#16#39/38, FAULT_ID = F8: Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Device • Event_Class = B#16#38, FAULT_ID = F9: Wiederkehr eines Teils der Submodule eines PROFINET I-Device mit Device-Konfigurationsunterschied
address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
id	UINT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
manufacturer	UINT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
instance	UINT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-25 **Programmierfehler-OB / Peripheriezugriffsfehler-OB: Datenstruktur TI_ProgIOAccessError**

Parameter	Datentyp	Byte	Beschreibung*
TI_ProgIOAccessError			Datenstruktur für Programmierfehler-OB und Peripheriezugriffsfehler-OB

Parameter	Datentyp	Byte	Beschreibung*
SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
OB_Class	USINT	1	OB-Klasse: • =120 für Programmierfehler-OB • =120 für Peripheriezugriffsfehler-OB
OB_Nr	UINT	2	OB-Nummer (1...32767).
BlockNr	UINT	4	Nummer des Bausteins, in dem der Programmierfehler auftrat.
Reaction	USINT	6	• 0: Fehler ignorieren • 1: Fehlerhaften Wert ersetzen • 2: Befehl überspringen
Fault_ID	BYTE	7	Fehlercode • B#16#21: BCD-Konvertierungsfehler • B#16#22: Bereichslängenfehler beim Lesen • B#16#23: Bereichslängenfehler beim Schreiben • B#16#24: Bereichsfehler beim Lesen • B#16#25: Bereichsfehler beim Schreiben • B#16#26: Fehler bei Timer-Nr. • B#16#27: Fehler bei Zähler-Nr. • B#16#28: Lesender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich "0" ist • B#16#29: Schreibender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich "0" ist • B#16#:30: Schreibender Zugriff auf einen schreibgeschützten Global-DB • B#16#31: Schreibender Zugriff auf einen schreibgeschützten Instanz-DB • B#16#32: DB-Nummernfehler beim Zugriff auf einen Global-DB • B#16#33: DB-Nummernfehler beim Zugriff auf einen Instanz-DB • B#16#34: Nummernfehler bei FC-Aufruf • B#16#35: Nummernfehler bei FB-Aufruf • B#16#42: Peripheriezugriffsfehler, lesend • B#16#43: Peripheriezugriffsfehler, schreibend • B#16#3A: Zugriff auf einen nicht geladenen DB; die DB-Nummer liegt im zulässigen Bereich • B#16#3C: Zugriff auf eine nicht geladene FC; die FC-Nummer liegt im zulässigen Bereich • B#16#3D: Zugriff auf eine nicht geladene Anweisung (SFC); die SFC-Nummer liegt im zulässigen Bereich • B#16#3E: Zugriff auf einen nicht geladenen FB; die FB-Nummer liegt im zulässigen Bereich • B#16#3F: Zugriff auf einen nicht vorhandenen SFB; die SFB-Nummer liegt im zulässigen Bereich
BlockType	USINT	8	Art des Bausteins, in dem der Fehler aufgetreten ist:

Parameter	Datentyp	Byte	Beschreibung*
			• 1: OB • 2: FC • 3: FB • 4: SFC • 5: SFB • 6: DB
Area	USINT	9	Speicherbereich, bei dem der fehlerhafte Zugriff auftrat: • Lokaldaten: B#16#40 bis 4E, 86, 87, 8E, 8F, C0 bis CE • Prozessabbild der Eingänge: B#16#01 • Prozessabbild der Ausgänge: B#16#02 • Technologie-DB: B#16#04 • E: B#16#81 • A: B#16#82 • M: B#16#83 • DB: B#16#84, 85, 8A, 8B
DBNr	DB_ANY	10	DB-Nr, falls Area = DB (Global-DB oder Instanz-DB) Nur relevant bei Programmierfehler-OB.
Csg_OBNr	OB_ANY	12	OB-Nummer: • 121: Programmierfehler-OB • 122: Peripheriezugriffsfehler-OB
Csg_Prio	USINT	14	OB-Priorität
Width	USINT	15	Art des Zugriffs, bei dem der Fehler auftrat: • Bit: B#16#00 • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04
* Abhängig davon, zu welchem OB (Peripheriezugriffsfehler-OB oder Programmierfehler-OB) Informationen ausgelesen werden, sind nur bestimmte Ausgabewerte möglich.			

Tabelle 4-26 **Uhrzeitalarm-OB: Datenstruktur TI_TimeOfDay**

Parameter		Datentyp	Byte	Beschreibung
TI_TimeOfDay				Datenstruktur für Uhrzeitalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=10)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	CaughtUp	BOOL	4	=TRUE, wenn ein OB-Aufruf wegen vorgestellter Uhrzeit nachgeholt wurde
	SecondTime	BOOL	5	=TRUE, wenn ein OB wegen zurückgestellter Uhrzeit ein zweites Mal aufgerufen wurde. Hinweis: SecondTime wird bei zurückgestellter Uhrzeit nur ein einziges Mal gesetzt.

Tabelle 4-27 **Taktsynchronalarm-OB: Datenstruktur TI_SynchCycle**

Parameter		Datentyp	Byte	Beschreibung
TI_SynchCycle				Datenstruktur für Taktsynchronalarm-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=61)
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	Initial_Call	BOOL	4	=TRUE im ersten Aufruf dieses OB bei: • Übergang von STOP bzw. HALT nach RUN • Nach Neuladen
	IO _System	USINT	5	Nummer des Alarm auslösenden Dezentralen Peripheriesystems
	Event_Count	INT	6	• = n: Anzahl der verlorenen Takte • = -1: Es ist eine unbekannte Anzahl von Takten verloren gegangen (z. B. weil der Takt sich geändert hat).
	PIP_Input	BOOL	10	=TRUE: Das zugehöriges Prozessabbild der Eingänge ist aktuell
	PIP_Output	BOOL	11	=TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen
	SyncCycle-Time	LTIME	16	Berechnete Zykluszeit

Tabelle 4-28 **Statusalarm-OB / Update-Alarm-OB / OB für hersteller- bzw. profilspezifischen Alarm: Datenstruktur TI_Submodule**

Parameter		Datentyp	Byte	Beschreibung
TI_Submodule				Datenstruktur für Statusalarm-OB, Update-Alarm-OB und OB für hersteller- bzw. profilspezifischen Alarm
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse • =55 für Statusalarm-OB • =56 für Update-Alarm OB • =57 für hersteller- bzw. profilspezifischen Alarm-OB
	OB_Nr	UINT	2	OB-Nummer (1...32767).
	LADDR	HW_IO	4	Hardware-Adresse der Alarm auslösenden Komponente
	Slot	UINT	6	Steckplatz-Nr. der Alarm auslösenden Komponente
	Specifier	WORD	8	Alarm Specifier aus dem Alarm-Telegramm
	address	WORD	20	Siehe Parameter "address" der Datenstruktur TI_Cassic.
	slv_prfl	BYTE	22	Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.
	intr_type	BYTE	23	Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.
	flags1	BYTE	24	Siehe Parameter "flags1" der Datenstruktur TI_Cassic.
	flags2	BYTE	25	Siehe Parameter "flags2" der Datenstruktur TI_Cassic.
	id	UINT	26	Siehe Parameter "id" der Datenstruktur TI_Cassic.
	manufacturer	UINT	28	Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.
	instance	UINT	30	Siehe Parameter "instance" der Datenstruktur TI_Cassic.

Tabelle 4-29 **CPU-Redundanzfehler-OB: Datenstruktur TI_CPURedundancyError_V2**

Parameter		Datentyp	Byte	Beschreibung
TI_CPURedundancyError_V2				Datenstruktur für CPU-Redundanzfehler-OB
	SI_Format	USINT	0	Format der Startinformationen: • 16#FF: Keine • 16#FE: Optimierte Startinformation
	OB_Class	USINT	1	OB-Klasse (=72)
	OB_Nr	UINT	2	OB-Nummer (1...32767)
	LADDR	HW_ANY	4	Hardware-Kennung des fehlerhaften Hardware-Objekts
	Event_Class	BYTE	6	Ereignisklasse und Kennungen (B#16#73: kommendes Ereignis)
	Fault_ID	BYTE	7	Fehlecode

Parameter AINFO

Datenstruktur des Zielbereiches AINFO bei Alarmen von PROFIBUS DP

Byte	Bedeutung	
0 bis 3	Kopfinformation, genaue Beschreibung s. u.	
4 bis 63	Alarmzusatzinformation: Daten zum jeweiligen Alarm:	
	Dezentral:	ARRAY[0] bis ARRAY[59]

Aufbau der Kopfinformation bei Alarmen von PROFIBUS DP

Byte	Datentyp	Bedeutung		
0	BYTE	Länge der empfangenen Alarminformation in Bytes		
		Zentral:	4 bis 224	
		Dezentral:	4 bis 63	
1	BYTE	Zentral:	reserviert	
		Dezentral:	Kennung für den Alarmtyp	
			1: 2: 3: 4: 5: 6: 31	Diagnosealarm Prozessalarm Ziehen-Alarm Stecken-Alarm Statusalarm Update-Alarm Ausfall eines Erweiterungsgeräts, eines DP-Mastersystems oder einer DP-Station
			32 bis 126:	herstellerspezifischer Alarm
2	BYTE	Steckplatznummer der Alarm auslösenden Komponente		
3	BYTE	Zentral:	reserviert	
		Dezentral:	Specifier	
			Bits 0 und 1:	0: keine weitere Information; 1: Steckplatz gestört 2: Steckplatz nicht gestört
			Bit 2:	Add_Ack
			Bits 3 bis 7:	Sequenznummer

Datenstruktur des Zielbereichs AINFO bei Alarmen von PROFINET IO oder zentraler Peripherie

Byte	Bedeutung
0 bis 25	Kopfinformation, genaue Beschreibung s. u.
26 bis 1431	Alarmzusatzinformation: Genormte Diagnosedaten zum jeweiligen Alarm: ARRAY[0] bis ARRAY[1405] Hinweis: Die Alarmzusatzinformation kann auch entfallen.

Aufbau der Kopfinformation bei Alarmen von PROFINET IO oder zentraler Peripherie

Byte	Datentyp	Bedeutung
0 und 1	WORD	• Bits 0 bis 7: Bausteintyp • Bits 8 bis 15: reserviert
2 und 3	WORD	Bausteinlänge
4 und 5	WORD	Version: • Bits 0 bis 7: low byte • Bits 8 bis 15: high byte
6 und 7	WORD	Kennung für den Alarmtyp: • 1: Diagnosealarm (kommend oder gehend) • 2: Prozessalarm • 3: Ziehen-Alarm • 4: Stecken-Alarm • 5: Status-Alarm • 6: Update-Alarm • 7: Redundanz-Alarm • 8: Vom Supervisor gesteuert (Controlled_by_Supervisor) • 9: Vom Supervisor freigegeben (Released_by_Supervisor) • 10: Nicht die projektierte Baugruppe gesteckt • 11: Wiederkehr des Submoduls • 12: Diagnosealarm (gehend) • 13: Querverkehr-Verbindungsmeldung • 14: Nachbarschaftsänderungsmeldung • 15: Taktsynchronisationsmeldung (busseitig) • 16: Taktsynchronisationsmeldung (geräteseitig) • 17: Netzwerk-Komponenten-Meldung • 18: Uhrzeit-Synchronisations-Meldung (busseitig) • 19 bis 29: Reserviert • 30: Upload-and-Retrieval-Alarm • 31: Reserviert • 32 bis 127: Herstellerspezifischer Alarm • 128 bis 65535: Reserviert
8 bis 11	DWORD	API (Application Process Identifier)
12 bis 13	WORD	Steckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)
14 bis 15	WORD	Submodulsteckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)
16 bis 19	DWORD	Modul-Identifikation; eindeutige Information der Alarmquelle

Byte	Datentyp	Bedeutung
20 bis 23	DWORD	Submodul-Identifikation; eindeutige Information der Alarmquelle
24 bis 25	WORD	Alarm Specifier: • Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) • Bit 11: Kanaldiagnose: 0: Keine Kanaldiagnose vorhanden 1: Kanaldiagnose vorhanden • Bit 12: Status der herstellerspezifischen Diagnose: 0: Keine herstellerspezifische Statusinformation vorhanden 1: Herstellerspezifische Statusinformation vorhanden • Bit 13: Status der Diagnose beim Submodul: 0: Keine Statusinformation vorhanden, alle Fehler wurden beseitigt 1: Mindestens eine Kanaldiagnose und/oder Statusinformation vorhanden • Bit 14: Reserviert • Bit 15: Application Relationship Diagnosis State: – 0: Keine der innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose – 1: Mindestens eine innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose

Aufbau der Alarmzusatzinformation bei Alarmen von PROFINET IO oder zentraler Peripherie

Die Alarmzusatzinformation bei PROFINET IO ist abhängig von der Formatkennung. Sie kann aus mehreren Datenblöcken mit gleicher oder unterschiedlicher Formatkennung bestehen. Folgende Formatkennungen sind möglich:

- W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose

Byte	Datentyp	Bedeutung
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose
2 bis n	BYTE	Siehe Handbuch des Herstellers.

- W#16#8000: Kanaldiagnose
  Die Kanaldiagnose wird in Blöcken zu jeweils 6 Byte ausgegeben. Die Alarmzusatzinformation (ohne Formatkennung) wird nur für die gestörten Kanäle ausgegeben.

Byte	Datentyp	Bedeutung	
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#8000: Kanaldiagnose	
2 bis 3	WORD	Kanalnummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535): • W#16#0000 bis W#16#7FFF: Kanalnummer des Submoduls/ Moduls • W#16#8000: ist der Stellvertreter für das gesamte Submodul • W#16#8001 bis W#16#FFFF: reserviert	
4	BYTE	Bits 0 bis 2:	Reserviert
		Bits 3 bis 4:	Fehlerart: • 0: Reserviert • 1: Kommender Fehler • 2: Gehender Fehler • 3: Gehender Fehler, weitere Fehler vorhanden
		Bits 5 bis 7:	Kanalart: • 0: Reserviert • 1: Eingabekanal • 2: Ausgabekanal • 3: Ein-/Ausgabekanal

Byte	Datentyp	Bedeutung
5	BYTE	Datenformat: • B#16#00: Freies Datenformat • B#16#01: Bit • B#16#02: 2 Bit • B#16#03: 4 Bit • B#16#04: Byte • B#16#05: Wort • B#16#06: Doppelwort • B#16#07: 2 Doppelworte • B#16#08 bis B#16#FF: reserviert

Byte	Datentyp	Bedeutung
6 bis 7	WORD	Fehlertyp: • W#16#0000: Reserviert • W#16#0001: Kurzschluss • W#16#0002: Unterspannung • W#16#0003: Überspannung • W#16#0004: Überlast • W#16#0005: Übertemperatur • W#16#0006: Leitungsbruch • W#16#0007: Oberer Grenzwert überschritten • W#16#0008: Unterer Grenzwert überschritten • W#16#0009: Fehler • W#16#000A: Simulation aktiv • W#16#000B bis W#16#000E: Reserviert • W#16#000F bis W#16#001F: Herstellerspezifisch • W#16#0020 bis W#16#00FF: Reserviert • W#16#0100 bis w#16#7FFF: Herstellerspezifisch • W#16#8000: Datenübertragung nicht möglich • W#16#8001: Fehler beim Partner • W#16#8002: Medienredundanz-Fehler • W#16#8003: Synchronisationsverlust • W#16#8004: Taktsynchronitätsfehler • W#16#8005: Multicast CR-Fehler • W#16#8006: reserviert • W#16#8007: Fiber Optic-Fehler • W#16#8008: Funktionsfehler der Netzwerkkomponente • W#16#8009: Zeitfehler • W#16#800A: DFP-Problem (DFP: dynamic frame packing) • W#16#800B: MRPD-Problem (MRPD: media redundancy with planned duplication) • W#16#800C: Systemredundanzfehler • W#16#800D bis W#16#FFFF: Reserviert Nicht jeder Kanal unterstützt jeden Fehlertyp. Details finden Sie in der Beschreibung der Diagnosedaten des entsprechenden Geräts.

Hinweis

Der Teil "Kanalnummer" bis "Fehlertyp" kann 0 bis n mal auftreten.

- W#16#8001: MULTIPLE (unterschiedliche Arten von Diagnoseinformationen werden übertragen). Die Alarmzusatzinformation wird in diesem Fall in Blockstruktur mit variabler Länge übertragen.

Byte	Datentyp	Bedeutung
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#8001: Herstellerspezifische Diagnose und/oder Kanaldiagnose
2 bis 3	WORD	Bausteintyp
4 bis 5	WORD	Bausteinlänge
6	BYTE	Version: high byte
7	BYTE	Version: low byte
8 bis 11	DWORD	API (nur falls das low byte von Version = 1)
12 bis 13	WORD	Steckplatznummer
14 bis 15	WORD	Substeckplatz-Nummer
16 bis 17	WORD	Kanalnummer
18 bis 19	WORD	Kanaleigenschaften
20 bis 21	WORD	Formatkennung: • W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose • W#16#8000: Kanaldiagnose • W#16#8002: Erweiterte Kanaldiagnose • W#16#8003: Abgestufte erweiterte Kanaldiagnose • W#16#8004 bis W#16#80FF: reserviert
22 bis n	BYTE	Daten abhängig von der Formatkennung

Hinweis

Der Teil ab "Bausteintyp" kann 1 bis n mal auftreten.

- W#16#8002: Erweiterte Kanaldiagnose

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8002
2 bis 3	Kanalnummer
4 bis 5	Kanaleigenschaften
6 bis 7	Fehlertyp
8 bis 9	Zusätzlicher Fehlerwert
10 bis 13	Zusätzliche Fehlerinformation

- W#16#8003: Abgestufte erweiterte Kanaldiagnose

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8003
2 bis 3	Kanalnummer
4 bis 5	Kanaleigenschaften
6 bis 7	Fehlertyp
8 bis 9	Zusätzlicher Fehlerwert
10 bis 13	Zusätzliche Fehlerinformation
14 bis 17	Qualified Channel Qualifier

- W#16#8100: Wartungsinformation

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8100
2 bis 3	Bausteintyp
4 bis 5	Bausteinlänge
6 bis 7	Bausteinversion
8 bis 9	Reserviert Hinweis: Bei zentralen Baugruppen fehlen diese beiden Bytes. Dezentral sind sie vorhanden (entsprechend dem PROFINET Standard).
10 bis 13	Wartungszustand

Hinweis

Genauere Informationen zum Aufbau der Alarmzusatzinformation entnehmen Sie dem *Programmierhandbuch SIMATIC PROFINET IO Von PROFIBUS DP nach PROFINET IO* und dem aktuellen Stand der Norm IEC 61158-6-10-1.

Zielbereich TINFO und AINFO

Zielbereich TINFO und AINFO

Abhängig vom jeweiligen OB, in dem "RALRM (Seite 2367)" aufgerufen wird, werden die Zielbereiche TINFO und AINFO nur teilweise beschrieben. Welche Information jeweils eingetragen wird, entnehmen Sie der folgenden Tabelle.

Alarmtyp	OB-Klasse	TINFO OB-Startinformation	TINFO Verwaltungsinformation	AINFO Kopfinformation	AINFO Alarmzusatzinformation	
Prozessalarm	4x	Ja	Ja	Ja	Zentral:	Nein
					Dezentral:	Wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Statusalarm	55	Ja	Ja	Ja	Ja	Ja
Update-Alarm	56	Ja	Ja	Ja	Ja	Ja
Herstellerspezifischer Alarm	57	Ja	Ja	Ja	Ja	Ja

Alarmtyp	OB-Klasse	TINFO OB-Startinformation	TINFO Verwaltungsinformation	AINFO Kopfinformation	AINFO Alarmzusatzinformation	
Peripherie-Redundanzfehler	70	Ja	Ja	Nein	Nein	Nein
Diagnosealarm	82	Ja	Ja	Ja	Zentral:	Aufbau gemäß PROFINET IO-Norm
					Dezentral:	Wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Ziehen-/Stecken-Alarm	83	Ja	Ja	Ja	Zentral:	Nein
					Dezentral:	Wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Sonderform des Ziehen-Alarms: Vom Supervisor gesteuert	83	Ja	Ja	Ja	Nur PROFINET IO	
Sonderform des Stecken-Alarms: Vom Supervisor freigegeben	83	Ja	Ja	Ja	Nur PROFINET IO	
Nicht projektierte Baugruppe gesteckt	83	Ja	Ja	Ja	Nur PROFINET IO	
Baugruppenträger-Ausfall/ Stationsausfall	86	Ja	Ja	Nein	Nein	
... alle anderen OBs		Ja	Nein	Nein	Nein	

Programmbeispiel für RALRM

Einleitung

In dem folgenden Beispiel generieren Sie einen Drahtbruch über eine Eingabebaugruppe und empfangen den Alarm, samt zugehöriger Informationen. Das Beispiel gilt für eine CPU S7-1500 bzw. PROFINET IO.

Hinweis

Beispiel platzieren

Das Beispiel muss in dem Aufbau abgelegt werden, in welchem die Eingabebaugruppe einen Alarm (Drahtbruch) erzeugen soll.

Voraussetzungen

- Eine Eingabebaugruppe ist angelegt und die Hardware-Kennung ist in den Eigenschaften der Eingabebaugruppe ausgelesen. Die Hardware-Kennung hinterlegen Sie über den Eingangsparameter F_ID ("alarmCheckID").
- Um einen Alarm zu generieren, aktivieren Sie für die Eingangskanäle die Option "Drahtbruch". Diese Option finden Sie bei der Eingabebaugruppe unter "Baugruppenparameter > Kanal-Vorlage > Eingänge".

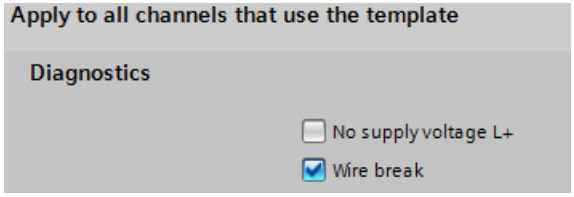

- Zur Ablage der Daten legen Sie in einem globalen Datenbaustein sieben Variablen sowie die Strukturen "TI_Diagnostic_Interrupt" und "Additional_Diag" an. Die Struktur "TI_Diagnostic_Interrupt" legen Sie dabei mit dem Datentyp "TI_DiagnosticInterrupt" an.

SLI_gDB_RALRM

	Name	Data type	Start value
1	Static		
2	alarmMode	Int	2
3	alarmCheckID	HW_IO	257
4	alarmStatus	DWord	16#0
5	alarmFromID	HW_IO	16#0
6	alarmReceived	Bool	false
7	alarmMaxLength	UInt	0
8	alarmLength	UInt	0
9	TI_Diagnostic_Interrupt	TI_DiagnosticInterrupt	
10	Additional_Diag	Struct	
11	header	Struct	
12	BlockType	Word	16#0
13	BlockLength	UInt	0
14	Version	Word	16#0
15	InterruptType	UInt	0
16	API	DWord	16#0
17	SlotNumber	UInt	0
18	SubSlotNumber	UInt	0
19	ModulSpecInfo	DWord	16#0
20	SubmodulSpecInfo	DWord	16#0
21	InterruptSpecifier	Word	16#0
22	addInfo	Struct	
23	FormatIdentifier	Word	16#0
24	ChannelNumber	Word	16#0
25	ChannelErrorType	Byte	16#0
26	DataFormat	Byte	16#0
27	ErrorCode	Word	16#0

Parameter verschalten

Die Anweisung "RALRM" rufen Sie in einem Diagnosealarm-OB (OB82) auf. Die Parameter der Anweisung verschalten Sie wie folgt.

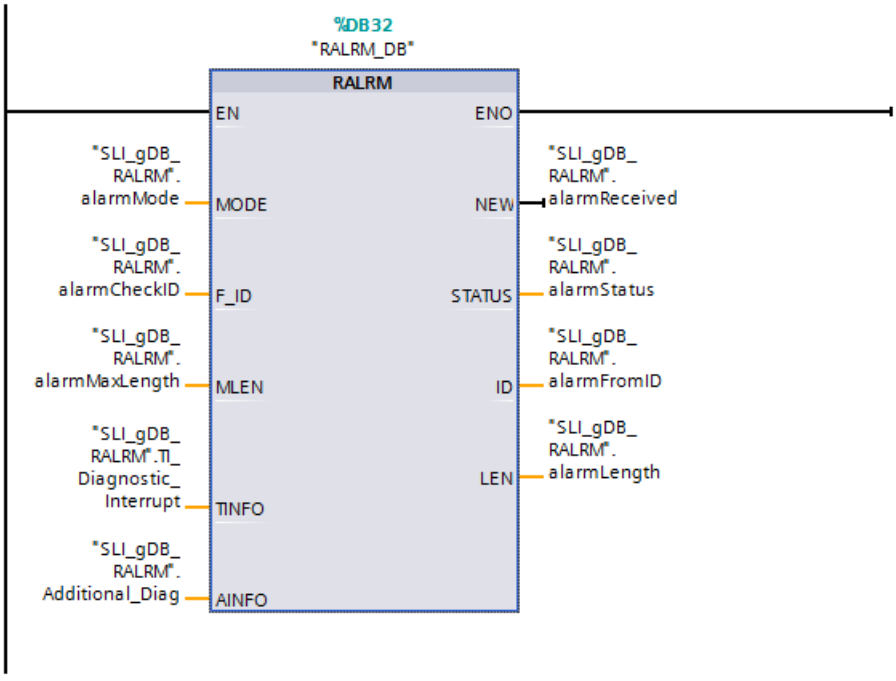

Ergebnis von RALRM

Hinweis

Drahbruch auslösen

Um den Alarm auszulösen, ziehen Sie eines der Stromkabel der Eingänge der Eingabebaugruppe und stecken das Kabel anschließend wieder hinein. Alternativ können Sie auch eine Potentialbrücke der Eingabebaugruppe ziehen und wieder hineinstecken.

Wenn ein Drahtbruch generiert wird, erzeugt die Eingabebaugruppe einen Alarm. Daraufhin wird der Diagnosealarm-OB aufgerufen und die Anweisung "RALRM" gestartet.

Über den Eingangsparameter F_ID ("alarmCheckID") ruft die Anweisung "RALRM" die Eingabebaugruppe auf. Entsprechend dem Wert "2" des Eingangsparameters MODE ("alarmMode") prüft die Anweisung "RALRM", ob der Alarm von der Eingabebaugruppe stammt. Wenn der Alarm von der Eingabebaugruppe stammt ("alarmCheckID" und "alarmFromID" sind gleich), wird der Ausgangsparameter NEW ("alarmReceived") auf "TRUE" gesetzt und die Alarminformation über die Parameter TINFO ("TI_Diagnostivc_Interrupt") und AINFO ("Additional_Diag") abgespeichert. Die Hardware-Kennung der Alarm generierenden Baugruppe (der Eingabebaugruppe) wird am Ausgangsparameter ID ("alarmFromID") angezeigt.

Die Länge der empfangenen Alarminformation in Bytes wird über den Ausgangsparameter LEN ("alarmLength") festgehalten. Entsprechend des Werts des Eingangsparameters MLEN

("alarmMaxLength" hat den Wert "0") ist die zu lesende Information für AINFO ("Additional_Diag") unbeschränkt.

Am Ausgangsparameter STATUS ("alarmStatus") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

SLI_gDB_RALRM

	Name	Data type	Start value	Monitor value
1	Static			
2	alarmMode	Int	2	2
3	alarmCheckID	HW_IO	257	16#0101
4	alarmStatus	DWord	16#0	16#0000_0000
5	alarmFromID	HW_IO	16#0	16#0101
6	alarmReceived	Bool	false	TRUE
7	alarmMaxLength	UInt	0	0
8	alarmLength	UInt	0	76

Die Auswertung zu TINFO und AINFO finden Sie in den nachfolgenden Abschnitten.

Details zu Parameter TINFO ("TI_Diagnostivc_Interrupt")

Am Parameter TINFO ("TI_Diagnostivc_Interrupt") werden die OB-Startinformationen und Verwaltungsinformationen abgespeichert.

Der generierte Alarm enthält u.a. folgende Informationen:

- OB-Startinformationen: Der OB82 verwendet optimierte Startinformationen ("SI_Format" mit dem Wert "254").
- Verwaltungsinformationen:
  - Die Eingabebaugruppe (siehe "LADDR") mit dem Status "Fehler" (siehe den Wert von "IO_State") wird ausgelesen.
  - Zuletzt wurde für den Kanal "15" (siehe den Wert von "Channel") ein Alarm erfasst.
  - Es wurden mehrere Fehler erfasst ("MutliError" mit dem Wert "TRUE").
  - Zuletzt wurde für Station "0" (siehe den Wert von "address") ein Alarm erfasst.
  - Es wurde eine Struktur gemäß Diagnosedatensatz 0 erfasst (siehe den Wert von "slv_prfl").
  - Der Alarm wurde von einer projektierten, dezentralen Baugruppe ausgelöst (siehe den Wert von "intr_type").

9	TI_Diagnostic_Interrupt	TI_DiagnosticInter...		
10	SI_Format	USInt	0	254
11	OB_Class	USInt	82	82
12	OB_Nr	UInt	0	82
13	LADDR	HW_ANY	16#0	16#0101
14	IO_State	Word	16#0	16#0010
15	Channel	UInt	0	15
16	MultiError	Bool	false	TRUE
17	address	Word	16#0	16#0000
18	slv_prfl	Byte	16#0	16#00
19	intr_type	Byte	16#0	16#00
20	flags1	Byte	16#0	16#00
21	flags2	Byte	16#0	16#00
22	id	UInt	0	0
23	manufacturer	UInt	0	0
24	instance	UInt	0	0

Details zu Parameter AINFO ("Additional_Diag")

Am Parameter AINFO ("Additional_Diag") werden die Kopfinformationen und Zusatzinformationen zum Alarm abgespeichert.

Hinweis

Aufbau und Umfang der Alarminformation

Dieses Beispiel ist zum Empfangen eines einzigen Hardware-Fehlers (Drahtbruch) gedacht. Dementsprechend wird bei Auftreten eines Multierrors nur die Alarminformation zu einem der betreffenden Kanäle angezeigt.

- Um für jeden Kanal einzeln die Zusatzinformationen aufzurufen, verwenden Sie einen Mehrfachaufruf von RALRM: Fragen Sie je Aufruf den Kanal ("TI_Diagnostivc_Interrupt.Channel") ab und hinterlegen Sie für jeden Kanal die Informationen der Variablen "ChannelErrorType" bis "ErrorCode". Siehe auch "FAQ zu Kanaldiagnose mit RALRM (https://support.industry.siemens.com/cs/de/de/view/109480387)".
- Beachten Sie die nach Anwendungsfall am Parameter AINFO (Seite 2393) zu hinterlegende Struktur.

Der generierte Alarm enthält u.a. folgende Informationen:

- Der verwendete OB ist ein Diagnosealarm-OB ("InterruptType" mit dem Wert "1").
- Der Alarm wird für eine Baugruppe auf dem Steckplatz "2" ("SlotNumber") und für die Kanäle 8..15 (BYTE 1, "SubSlotNumber") gemeldet.
- Es ist mindestens eine Kanaldiagnose und/oder Statusinformation für das Submodul vorhanden ("InterruptSpecifier" mit dem Wert "16#2000").
- In diesem Beispiel wird die Struktur der Kanaldiagnose verwendet ("FormatIdentifier" mit dem Wert "16#8000").
- Die Kanalnummer der Alarm auslösenden Komponente ist dem Submodul zuzuordnen ("ChannelNumber" mit dem Wert "16#0008").

- Es wird ein kommender Fehler für einen Eingabekanal ("ChannelErrorType" mit dem Wert "16#28") diagnostiziert.
- Das verwendete Datenformat ist ein Bit ("DataFormat" mit dem Wert "16#01").
- Es wurde ein Drahtbruch diagnostiziert ("ErrorCode" mit dem Wert "16#0006").

25	Additional_Diag	Struct		
26	header	Struct		
27	BlockType	Word	16#0	16#0002
28	BlockLength	UInt	0	72
29	Version	Word	16#0	16#0100
30	InterruptType	UInt	0	1
31	API	DWord	16#0	16#0000_0000
32	SlotNumber	UInt	0	2
33	SubSlotNumber	UInt	0	1
34	ModulSpecInfo	DWord	16#0	16#0000_0000
35	SubmodulSpecInfo	DWord	16#0	16#0000_0000
36	InterruptSpecifier	Word	16#0	16#2000
37	addInfo	Struct		
38	FormatIdentifier	Word	16#0	16#8000
39	ChannelNumber	Word	16#0	16#0008
40	ChannelErrorType	Byte	16#0	16#28
41	DataFormat	Byte	16#0	16#01
42	ErrorCode	Word	16#0	16#0006

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Weiteres Beispiel

Ein komplexeres Beispiel finden Sie hier: "FAQ zu Kanaldiagnose mit RALRM (https://support.industry.siemens.com/cs/de/de/view/109480387)".

D_ACT_DP: DP-Slaves deaktivieren / aktivieren

Beschreibung

Mit der Anweisung "D_ACT_DP" deaktivieren und aktivieren Sie gezielt projektierte DP-Slaves/PROFINET IO-Devices. Darüber hinaus können Sie für jeden eingesetzten DP-Slave bzw. für jedes eingesetzte PROFINET IO-Device ermitteln, ob diese Komponente momentan aktiviert oder deaktiviert ist.

Hinweis

S7-1200-CPUs

In S7-1200-CPUs können Sie die Anweisung D_ACT_DP nur bei PROFINET IO-Devices verwenden. Sie können sie nicht bei PROFIBUS DP-Slaves verwenden.

Sie deaktivieren / aktivieren einen Netzübergang vom Typ IE/PB Link PN IO nicht mit der Anweisung "D_ACT_DP". Falls Sie "D_ACT_DP" dennoch auf den genannten Netzübergang anwenden, liefert die CPU den Rückgabewert W#16#8093 (Zu der in LADDR angegebenen Adresse gehört kein Hardware-Objekt, das aktiviert oder deaktiviert werden kann.).

Ab Firmware-Stand V1.8 der S7-1500-CPUs gilt: Sie deaktivieren / aktivieren DP-Slaves, die an einem Netzübergang vom Typ IE/PB Link PN IO angeschlossen sind, mit der Anweisung "D_ACT_DP".

Die Anweisung ist nicht anwendbar auf PROFIBUS PA-Feldgeräte, die über DP/PA Link an ein DP-Mastersystem angeschlossen sind.

Hinweis

Für die Bearbeitung eines Deaktivier- bzw. Aktivierauftrags sind mehrere Durchläufe durch den Zykluskontrollpunkt notwendig. Sie warten daher nicht in einer programmierten Schleife auf das Ende eines solchen Auftrags.

Arbeitsweise

"D_ACT_DP" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "D_ACT_DP" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Anwendungsbereich

Wenn Sie in einer CPU DP-Slaves/PROFINET IO-Devices konfigurieren, die real nicht vorhanden sind oder aktuell nicht benötigt werden, greift die CPU dennoch regelmäßig auf diese DP-Slaves/PROFINET IO Devices zu. Nach deren Deaktivierung unterbleiben weitere CPU-Zugriffe. Bei PROFIBUS DP kann dadurch der schnellstmögliche DP-Buszyklus erreicht werden. Die entsprechenden Fehlerereignisse treten nicht mehr auf.

Beispiele

Im Serienmaschinenbau ist aus Sicht des Maschinenbauers eine Vielzahl von Maschinenoptionen möglich. Jede ausgelieferte Maschine beinhaltet jedoch nur eine individuelle Kombination ausgewählter Optionen.

Sämtliche möglichen Maschinenoptionen sind vom Hersteller als DP-Slaves/PROFINET IO Devices projektiert, um ein gemeinsames Anwenderprogramm über alle möglichen Optionen erstellen und pflegen zu können. Mit "D_ACT_DP" deaktivieren Sie im Anlauf der Maschine alle nicht vorhandenen DP-Slaves/PROFINET IO Devices deaktivieren.

Eine ähnliche Situation besteht bei Werkzeugmaschinen, die viele Werkzeuge vorhalten, aber zu einem Zeitpunkt nur wenige einsetzen. Hier werden die Werkzeuge als DP-Slaves/ PROFINET IO Devices realisiert. Das Anwenderprogramm aktiviert mit Hilfe von "D_ACT_DP" die aktuell benötigten und deaktiviert die erst später wieder einzusetzenden Werkzeuge.

Identifikation eines Auftrags

Falls Sie einen Deaktivierungs- bzw. Aktivierungsauftrag angestoßen haben und Sie "D_ACT_DP" erneut aufrufen, bevor dieser beendet wurde, hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Wenn der Eingangsparameter LADDR übereinstimmt, so gilt der Aufruf als Folgeaufruf.

Deaktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO Device mit "D_ACT_DP" deaktivieren, werden dadurch dessen Prozessausgänge auf die projektierten Ersatzwerte bzw. auf 0 gesetzt (sicherer Zustand). Der zugehörige DP-Master/PROFINET IO Controller spricht diese Komponente im Weiteren nicht mehr an. Deaktivierte DP-Slaves/PROFINET IO-Devices werden an den Fehler-LEDs des DP-Masters/PROFINET IO-Controllers oder der CPU nicht als gestört oder fehlend gekennzeichnet.

Das Prozessabbild der Eingänge von deaktivierten DP-Slaves/PROFINET IO-Devices wird mit 0 aktualisiert. Daher wird es wie bei ausgefallenen DP-Slaves/PROFINET IO-Devices behandelt.

Falls Sie in Ihrem Programm durch Direktzugriff auf die Nutzdaten eines zuvor deaktivierten DP-Slaves/PROFINET IO-Devices zugreifen, wird der Peripheriezugriffsfehler-OB aufgerufen und das zugehörige Startereignis in den Diagnosepuffer eingetragen. Falls Sie per Anweisung (z. B. "RD_REC (Seite 2428)") auf einen deaktivierten DP-Slave bzw. ein deaktiviertes PROFINET IO-Device zugreifen, erhalten Sie in RET_VAL dieselbe Fehlerinformation wie bei einem nicht verfügbaren DP-Slave/PROFINET IO-Device.

Das Deaktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OB, auch wenn dessen Ein- bzw. Ausgänge zum systemseitig zu aktualisierenden Prozessabbild gehören.

Fällt eine DP-Station/PNIO-Station aus, nachdem Sie sie mit "D_ACT_DP" deaktiviert haben, wird der Ausfall vom Betriebssystem nicht erkannt.

Bei PROFIBUS DP gilt: Falls Sie DP-Slaves deaktivieren wollen, die als Sender am Querverkehr beteiligt sind, wird empfohlen, zuerst die Empfänger (Mithörer) zu deaktivieren, die mithören, welche Eingangsdaten der Sender seinem DP-Master schickt. Erst im Anschluss daran deaktivieren Sie den Sender.

Aktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO-Device mit "D_ACT_DP" wieder aktivieren, wird diese Komponente vom zugehörigen DP-Master/PROFINET IO-Controller konfiguriert und parametriert (wie bei der Wiederkehr einer ausgefallenen DP-Station/PROFINET IO-Station). Die Aktivierung ist abgeschlossen, wenn die Komponente Nutzdaten transferieren kann.

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OBs, auch wenn dessen Ein- bzw. Ausgänge zum systemseitig zu aktualisierenden Prozessabbild gehören.

Wenn Sie versuchen, einen DP-Slave/PROFINET IO-Device mit "D_ACT_DP" zu aktivieren, der nicht erreichbar ist (z. B. weil dieser physikalisch vom Bus getrennt wurde), liefert die Anweisung nach Ablauf der konfigurierten Parametrierungszeit für dezentrale Peripherie den

Fehlercode W#16#80A7. Der DP-Slave / das PROFINET IO-Device ist aktiviert und die Nichterreichbarkeit des aktivierten DP-Slaves / PROFINET IO-Device führt zur entsprechenden Anzeige in der Systemdiagnose.

Ist der DP-Slave/PROFINET IO-Device danach wieder erreichbar, resultiert dies in dem normalen Systemverhalten (z. B. in einem Aufruf des hierfür projektierten OBs).

Hinweis

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices kann längere Zeit dauern. Falls Sie einen laufenden Aktivierungsauftrag abbrechen wollen, starten Sie "D_ACT_DP" mit dem gleichen Wert für LADDR und MODE = 2. Sie wiederholen den Aufruf von "D_ACT_DP" mit MODE = 2 so lange, bis der erfolgreiche Abbruch des Aktivierungsauftrags mit RET_VAL = 0 angezeigt wird.

Falls Sie DP-Slaves aktivieren wollen, die am Querverkehr beteiligt sind, wird empfohlen, zuerst die Sender und anschließend die Empfänger (Mithörer) zu aktivieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "D_ACT_DP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Pegelgetriggerter Steuerparameter REQ=1: Aktivieren bzw. Deaktivieren durchführen
MODE	Input	USINT	E, A, M, D, L oder Konstante	Auftragskennung Mögliche Werte: • 0: Auskunft einholen, ob die angesprochene Komponente aktiviert oder deaktiviert ist (Ausgabe über Parameter RET_VAL) • 1: DP-Slave/PROFINET IO-Device aktivieren • 2: DP-Slave/PROFINET IO-Device deaktivieren
LADDR	Input	HW_DEVICE	E, A, M, D, L oder Konstante	Hardware-Kennung des DP-Slaves (HW_DPSlave)/PROFINET IO-Devices (HW_Device) Die Nummer kann in der Netzsicht aus den Eigenschaften DP-Slaves/PROFINET IO-Devices oder dem Register "Systemkonstanten" der Standard-Variablentabelle übernommen werden. Falls dort sowohl die Kennung für die Gerätediagnose als auch die Kennung für Betriebszustandsübergänge angegeben ist, müssen Sie die Kennung für die Gerätediagnose verwenden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	Aktivkennung: • BUSY=1: Der Auftrag ist noch aktiv. • BUSY=0: Der Auftrag wurde beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
0001	DP-Slave/PROFINET IO-Device ist aktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)
0002	DP-Slave/PROFINET IO-Device ist deaktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)
7000	Erstaufruf mit REQ=0. Der über LADDR festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1. Der über LADDR festgelegte Auftrag wurde angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant). Der aktivierte Auftrag ist noch in Bearbeitung; BUSY hat den Wert "1".
8090	• Sie haben keine Baugruppe mit der in LADDR angegebenen Adresse projektiert. • Sie betreiben Ihre CPU als I-Slave / I-Device und haben in LADDR eine Adresse dieses I-Slaves / I-Device angegeben.
8092	Der laufende Deaktiviervorgang eines DP-Slaves/PROFINET IO-Devices (MODE=2) kann nicht durch seine Aktivierung (MODE=1) abgebrochen werden. Aktivieren Sie die Komponente zu einem späteren Zeitpunkt.
8093	Zu der in LADDR angegebenen Adresse gehört kein DP-Slave/PROFINET IO-Device, der aktiviert oder deaktiviert werden kann, oder der Parameter MODE ist nicht bekannt.
8094	Sie haben versucht, ein Gerät zu aktivieren, das ein möglicher Partner eines Werkzeugwechselports ist. Zu diesem Zeitpunkt ist an diesem Werkzeugwechselport aber bereits ein anderes Gerät aktiviert. Das aktivierte Gerät bleibt aktiviert.
80A0	Fehler bei der Kommunikation zwischen der CPU und dem IO-Controller.
80A3	Der betroffene DP-Master/PROFINET IO-Controller unterstützt diese Funktion nicht.
80A4	Die CPU unterstützt diese Funktion bei externen DP-Mastern/PROFINET IO-Controllern nicht.
80A7	Während der Aktivierung ist eine Zeitüberschreitung aufgetreten: Das entfernte Gerät ist nicht erreichbar oder Sie haben die Parametrierungszeit für zentrale und dezentrale Peripherie zu klein eingestellt. Der Zustand des entfernten Geräts ist "aktiviert", aber es ist noch nicht erreichbar.
80AA	Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Unterschiede in der Konfiguration
80AB	Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Parametrierfehler
80AC	Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Wartungsbedarf erforderlich
80C3	• Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an Aktivierungs-/Deaktivierungsaufträgen (8). • Die CPU erhält gerade eine geänderte Konfiguration. Das Deaktivieren und Aktivieren von DP-Slaves/PROFINET IO-Devices ist daher momentan nicht möglich.

Fehlercode* (W#16#...)	Erläuterung
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ReconfigIOSystem: IO-System umkonfigurieren

ReconfigIOSystem: IO-System umkonfigurieren

Beschreibung

Mit der Anweisung "ReconfigIOSystem" übertragen Sie einen Datensatz an die PROFINET-Schnittstelle der CPU. Dieser Datensatz (Parameter "CTRLREC") enthält folgende Informationen zum Steuern der Konfiguration:

- Eine Liste der zu aktivierenden optionalen IO-Devices
- Eine Liste der zu spezifizierenden Partner-Ports, falls Sie bei den Port-Eigenschaften von IO-Devices die Option "Partner im Anwenderprogramm gesetzt" eingestellt haben

Hinweis

Die Anweisung "ReconfigIOSystem" nutzt im MODE 1 und im MODE 3 intern die Anweisung "D_ACT_DP" zum Aktivieren/Deaktivieren von IO-Devices. Beachten Sie daher die Regeln und Hinweise in der Beschreibung zu dieser Anweisung.

Siehe auch "D_ACT_DP: Dezentrale Peripheriegeräte deaktivieren/aktivieren (Seite 2406)".

Abhängigkeiten zwischen Anweisungs-Versionen

Die Version V1.1 der Anweisung "ReconfigIOSystem" erfordert die Version V1.1 der Anweisung "WRREC".

Die Version V1.0 der Anweisung "ReconfigIOSystem" erfordert die Version V1.1 der Anweisung "WRREC".

Arbeitsweise

"ReconfigIOSystem" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "ReconfigIOSystem" mit REQ=1 aufrufen.

Über die Ausgangsparameter STATUS und BUSY wird der Zustand des Auftrags angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ReconfigIOSystem":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Flankengetriggerter Steuerparameter REQ=1: Datenübertragung durchführen
MODE	Input	UINT	E, A, M, D, L oder Konstante	Über den Parameter MODE steuern Sie, wie die Anweisung arbeitet. Die detaillierte Beschreibung der Funktionsweise finden Sie unter dieser Tabelle erläutert. Mögliche Werte: • 1: Für die Umrüstphase alle IO-Devices des IO-Systems deaktivieren • 2: Umkonfigurieren des IO-Systems gemäß den Datensatz-Einstellungen (CTRLREC) • 3: Nach Umkonfigurieren alle IO-Devices des IO-Systems wieder aktivieren
LADDR	Input	HW_INTER-FACE	E, A, M, D, L oder Konstante	HW-Kennung der PROFINET-Schnittstelle (IO-Controller)
CTRLREC	Input	VARIANT	E, A, M, D, L	Datensatz zur Steuerung der tatsächlichen Konfiguration des IO-Systems
DONE	Output	BOOL	E, A, M, D, L	0: Anweisung noch nicht beendet 1: Anweisung beendet
BUSY	Output	BOOL	E, A, M, D, L	Aktivkennung: 0: Anweisung aktiv 1: Anweisung beendet
ERROR	Output	BOOL	E, A, M, D, L	0: Anweisung ohne Fehler beendet 1: Anweisung mit Fehler beendet
STATUS	Output	DWORD	E, A, M, D, L	Ergebnis/Fehlerkennung
ERR_INFO	Output	WORD	E, A, M, D, L	HW-Kennung des zuletzt ermittelten fehlerverursachenden IO-Devices.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

Für den Eingangsparameter MODE sind folgende Werte möglich:

MODE	Beschreibung
1	Alle IO-Devices des IO-Systems werden durch Aufruf der Anweisung mit dem Mode 1 deaktiviert. Intern verwendet die Anweisung "ReconfigIOSystem" die Anweisung "D_ACT_DP". Fehler, die durch D_ACT_DP ermittelt werden, gibt "ReconfigIOSystem" in folgenden Ausgangsparametern wieder: • STATUS (Fehlercode) • ERR_INFO (HW-Kennung des fehlerverursachenden IO-Devices). In STATUS und ERR_INFO trägt die CPU den zuletzt ermittelten Fehler / die zuletzt ermittelte HW-Kennung ein und überschreibt dabei einen bereits vorhandenen Fehlercode. Daher können neben dem eingetragenen Fehler weitere Fehler vorliegen.
2	Die Anweisung überträgt den Datensatz zur Steuerung der tatsächlichen Konfiguration des IO-Systems an die PROFINET-Schnittstelle, die durch LADDR adressiert ist. Zum Aufbau des Datensatzes siehe nachfolgenden Abschnitt. Zur Fehlerauswertung siehe nachfolgenden Abschnitt zum Parameter STATUS.
3	Alle nicht optionalen IO-Devices des IO Systems und optionale IO-Devices, die im Steuerdatensatz CTRLREC aufgelistet sind, werden aktiviert. Die optionalen IO-Devices, die **nicht** im Datensatz CTRLREC aufgeführt sind, bleiben deaktiviert. Wenn IO-Devices, die Teil von Docking Units sind, im Steuerdatensatz CTRLREC aufgeführt sind, verhält sich das PN IO-System wie folgt: • IO-Devices der Docking Units bleiben beim Aufruf von ReconfigIOSystem mit Mode 3 deaktiviert. Dieses Verhalten entspricht dem Verhalten einer Projektierung ohne konfigurationsgesteuerte IO-Devices. IO-Devices von Docking Units sind in der Grundeinstellung deaktiviert und müssen im Anwenderprogramm aktiviert werden.

Aufbau des Steuerungsdatensatzes

Mithilfe des Steuerungsdatensatzes (Parameter "CTRLREC") teilen Sie der PROFINET-Schittstelle der CPU mit, welche der als optional parametrierten IO-Devices im realen IO-Systemaufbau vorhanden sind und welche Portverschaltungen zu setzen sind.

Voraussetzung ist eine Projektierung, die eine Anpassung des IO-Systemaufbaus zulässt:

- Die im Steuerungsdatensatz aufgeführten IO-Devices müssen als "Optionales IO-Device" aktiviert sein (Eigenschaften IO-Device: PROFINET-Schnittstelle [X1] > Erweiterte Optionen > Schnittstellen-Optionen).
- Die im Steuerungsdatensatz aufgeführten Portverschaltungen sind nur möglich, wenn bei den entsprechenden Ports der Partnerport eingestellt ist auf "Partner im Anwenderprogramm gesetzt".

Der Parameter "CTRLREC" ist vom Datentyp "VARIANT". Der Steuerungsdatensatz "CTRLREC" muss folgendermaßen aufgebaut sein:

- Bis Version V1.1 von ReconfigIOSystem: Array aus Elementen vom Datentyp Word (für numerische Adressierung)
- Ab Version V1.2 von ReconfigIOSystem: Array aus Elementen vom Datentyp UInt (für symbolische oder numerische Adressierung) oder Array aus Elementen vom Datentyp Word (für numerische Adressierung)

Im Folgenden ist der grundsätzliche strukturelle Aufbau "CTRLREC" mit dem Element-Datentyp Word beschrieben.

Die benötigten HW-Kennungen lesen Sie direkt im Register "Systemkonstanten" in der Netzsicht bzw. in der Gerätesicht ab. Markieren Sie dazu das Objekt in der Netzsicht (IO-Device) oder in der Gerätesicht (PROFINET-Schnittstelle).

Empfehlung: Nutzen Sie die symbolische Adressierung der HW-Kennungen, indem Sie die Namen der HW-Kennungen verwenden!

Name	Datentyp	Kommentar
Version_High, Version_Low	Word	Version des Steuerdatensatzes(High Byte: 01 Low Byte: 00
Number_of_opt_Devices_used	Word	Anzahl der optionalen IO-Devices, die im realen IO-Systemaufbau verwendet sind. Optionale IO-Devices, die im Folgenden nicht aufgeführt sind, bleiben deaktiviert.
Activate_opt_Device_1	UINT / Hw_Device	Die optionalen IO-Devices, die in der realen Konfiguration vorhanden sind, führen Sie mit ihren HW-Kennungen auf. Verwenden Sie die Systemkonstante des IO-Device-Objekts. Beispiel: Das IO-Device-Objekt hat den Namen "IO-Device_4~IODevice" und ist vom Typ "Hw_Device".
Activate_opt_Device_2	UINT / Hw_Device	2. optionales IO-Device (z. B. mit dem Wert 262)
...	...	...
Activate_opt_Device_n	UINT / Hw_Device	n. optionales IO-Device (z. B. mit dem Wert 282)
Number_of_Port_Interconnections_used	Word	Anzahl der im Folgenden aufgeführten Portverschaltungen. Wenn Sie keine Portverschaltungen spezifizieren, tragen Sie eine "0" ein. Für alle Ports, bei denen Sie projektiert haben "Partner im Anwenderprogramm gesetzt" und die im Folgenden nicht aufgeführt sind, verwendet die CPU die Einstellung "Beliebiger Partner".
Port_Interconnection_1_Local	UINT / Hw_Interface	1. Portverschaltung, HW-Kennung lokaler Port Verwenden Sie die Systemkonstante des Port-Objekts. Beispiel: Das Port-Objekt hat den Namen "IO-Device_2~PROFINET_interface~Port_2" und ist vom Typ "Hw_Interface".
Port_Interconnection_1_Remote	UINT / Hw_Interface	1. Portverschaltung, HW-Kennung Partnerport
Port_Interconnection_2_Local	UINT / Hw_Interface	2. Portverschaltung, HW-Kennung lokaler Port
Port_Interconnection_2_Remote	UINT / Hw_Interface	2. Portverschaltung, HW-Kennung Partnerport
...	...	...
Port_Interconnection_n_Local	UINT / Hw_Interface	n. Portverschaltung, HW-Kennung lokaler Port
Port_Interconnectio_n_Remote	UINT / Hw_Interface	n. Portverschaltung, HW-Kennung Lokaler Port

Parameter STATUS

Der Ausgangsparameter STATUS enthält Fehlerinformationen. Die Liste der möglichen Fehlercodes im Detail finden Sie im nächsten Abschnitt. Wenn Sie STATUS als ARRAY[1...4] of BYTE interpretieren, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	B#16#DF: Fehler beim Schreiben eines Datensatzes, sonst B#16#00.
STATUS[2]	Error_Decode	Bei einem Fehler wird B#16#80 ausgegeben (entsprechend IEC 61158-6 im Kontext Lesen und Schreiben von Datensätzen).
STATUS[3]	Error_Code_1	B#16#AA im Fall eines Fehlers in der Struktur des Datensatzes
STATUS[4]	Error_Code_2	Herstellerspezifische Erweiterung der Fehlerkennung: • B#16#00 Fehler im Datensatz (z. B. falsche Füllbyte-Werte (ungleich 0) • B#16#01 Reserviert • B#16#02 Mindestens eine Stationsnummer für ein IO-Device im Datensatz ist ungültig (nicht projektiert oder bezeichnet ein nicht optionales IO-Device oder hat den Wert 0 für einen IO-Controller) • B#16#03 Mindestens ein Partnerport, der im Datensatz spezifiziert ist, ist ungültig: Beispiele: – Substeckplatz-Adresse eines Partnerports ist nicht vorhanden – Partnerport ist falsch projektiert (richtig ist die Einstellung: "Partner im Anwenderprogramm gesetzt") – Partnerport gehört zu einem IO-Device, das deaktiviert ist • B#16#10 Version des Steuerdatensatzes CTRLREC ungültig (angegebene Version wird nicht unterstützt) • B#16#11 Die Anzahl zu aktivierender optionaler IO-Devices im Steuerdatensatz CTRLREC wird nicht unterstützt • B#16#12 Die Anzahl der spezifizierten Verschaltungen ("Partner im Anwenderprogramm gesetzt") im Steuerdatensatzes CTRLREC wird nicht unterstützt • B#16#13 Interne Umrechnung der HW-Kennung in Gerätenummer fehlgeschlagen. Der Ausgangsparameter ERR_INFO enthält die HW-Kennung des fehlerverursachenden Gerätes. • B#16#14 Konsistenzfehler: Die Länge des Steuerdatensatzes CTRLREC passt nicht mit den Angaben im Steuerdatensatz zusammen. Beispiel: Angegeben sind 20 optionale IO-Devices, der Steuerdatensatz hat aber nur eine Länge von 10 Byte.

Fehlercodes (Parameter STATUS)

Fehlercode	Beschreibung
16#0000_0000	Auftrag fertig ohne Fehler
16#0070_0000	Kein Auftrag aktiv
16#0070_0100	Erstaufruf der Anweisung
16#0070_0200	Folgeaufruf der Anweisung (Anweisung läuft noch, BUSY=1)

Fehlercode	Beschreibung
16#0080_8000	MODE wird nicht unterstützt
16#0084_5100	Falscher Datentyp des Datensatzes CRTLREC. Verwenden Sie ein Array of UInt oder ein Array of Word.
16#0080_9100	Der Parameter LADDR adressiert keine PROFINET-Schnittstelle (existiert nicht oder hat den falschen Typ, z. B. PROFIBUS-Schnittstelle). Die PROFINET-Schnittstelle unterstützt keine Konfiguationssteuerung von IO-Systemen.
16#0080_Cx00	Temporärer Fehler, z. B. wegen kurzzeitigem Ressourcenmangel.
16#DF80_AAxy	Fehler in der Struktur des Datensatzes (MODE 2). Zur Bedeutung von "xy" siehe oben, Definition von STATUS[4] Error_Code_2.
16#DF80_B6xy	Konfigurationssteuerung nicht möglich, weil entweder kein optionales IO-Device projektiert und kein Port mit "Partner im Anwenderprogramm gesetzt" parametriert wurde. Diese Projektierung ist Voraussetzung für den Aufruf der Anweisung. "xy" ist irrelevant.
16#0080_9400 16#0080_A000 16#0080_A700 16#0080_AA00 16#0080_AB00 16#0080_AC00	Durchgereichte Fehlercodes der intern aufgerufenen Anweisung D_ACT_DP. Zur Bedeutung dieser Fehlercodes siehe "D_ACT_DP (Seite 2406)". Die HW-Kennungen fehlerverursachender IO-Devices werden in ERR_INFO eingetragen (Eintrag wird durch weitere Fehler fortlaufend überschrieben). Wenn mehrere IO-Devices betroffen sind, empfiehlt sich die Online-Diagnose mit STEP 7. Bei IRT-Projektierung: Die Gerätenummern der IO-Devices sollten vom IO-Controller aus in aufsteigender Reihenfolge der topologischen Verschaltung folgen, siehe hier.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für ReconfigIOSystem (Seite 2416).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für ReconfigIOSystem

Einleitung

In dem folgenden Beispiel konfigurieren Sie ein IO-System über das Programm mit Hilfe des Steuerdatensatzes "CTRLREC". Die Ziel-Konfiguration soll aus einer CPU S7-1500, einem IO-Device (z. B. ET 200MP) und einem optionalen IO-Device (z. B. ET 200SP) bestehen. Dazu deaktivieren Sie ein bestehendes IO-System, übertragen die Konfigurationswerte an die PROFINET-Schnittstelle der CPU und aktivieren die IO-Devices.

Weitere Informationen zu Datensätzen finden Sie in der Übersicht über die Typen von Datensätzen (Seite 2298).

Voraussetzungen

- Eine CPU S7-1500 ist angelegt
- Zwei ET 200 sind angelegt
- Eine PROFINET-Verbindung ist angelegt und die Hardware-Einstellungen sind vorgenommen:
  - IO-Device (z. B. ET 200MP IM 155-5 PN HF)
  - optionales IO-Device (z. B. ET 200SP IM 155-6 PN ST)
  - IO-System wird durch das Programm eingerichtet

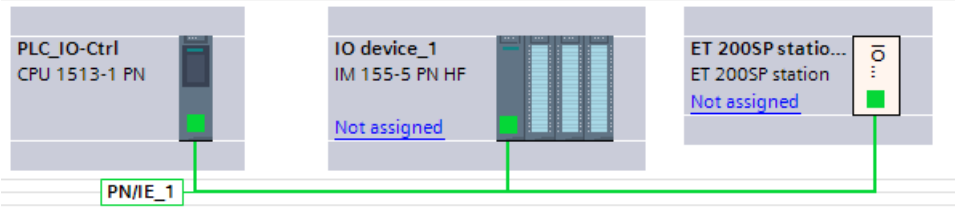

Hinweis

Hardware-Einstellungen für ein durch das Programm einzurichtendes IO-System

Gehen Sie wie folgt vor:

1. Verbinden Sie die zwei ET 200 über PROFINET mit der CPU.
2. Weisen Sie beiden ET 200 die CPU als IO-Controller zu. Öffnen Sie anschließend die CPU-Eigenschaften.
3. Öffnen Sie den Eintrag "PROFINET Schnittstelle [X1] > Erweiterte Optionen > Port X1 P1 > Portverschaltung > Partnerport". Wählen Sie die Option "Partner durch Anwenderprogramm einstellen". Machen Sie dasselbe für "Port X1 P2".
4. Öffnen Sie die Eigenschaften der ET 200MP (...> Port X1 P1) und wählen Sie die Option "Partner durch Anwenderprogramm einstellen" (siehe Schritt 3).
5. Öffnen Sie die Eigenschaften der ET 200SP (...> Port X1 P1) und machen Sie folgende Einstellungen:
   - Wählen Sie die Option "Partner durch Anwenderprogramm einstellen" (siehe Schritt 3).
   - Öffnen Sie "PROFINET Schnittstelle [X1] > Erweiterte Optionen > Schnittstellen-Optionen" und aktivieren Sie die Option "Optionales IO-Device".

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die beiden Strukturen "operateMode" und "ctrlRec" sowie 13 weitere Variablen an.

Die Struktur "ctrlRec" enthält den Steuerdatensatz "CTRLREC".

SLI_gDB_ReconfigIOSystem

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	startDone	Bool	false
4	reset	Bool	false
5	getExecute	Bool	false
6	operateMode	Struct	
7	value	UInt	1
8	Disable_all_IO_devices_done	Bool	false
9	Reconfigure_IO_system_done	Bool	false
10	Enable_all_IO_devices_done	Bool	false
11	laddr_interface	HW_INTERFACE	64
12	ctrlRec	Struct	
13	Version	Word	16#0100
14	Number_of_opt_Devices_used	Word	1
15	Activate_opt_Device_1	HW_DEVICE	264
16	Number_of_Port_Interconnections_used	Word	2
17	Port_Interconnection_1_Local	HW_INTERFACE	65
18	Port_Interconnection_1_Remote	HW_INTERFACE	259
19	Port_Interconnection_2_Local	HW_INTERFACE	66
20	Port_Interconnection_2_Remote	HW_INTERFACE	273
21	status	DWord	16#0
22	busy	Bool	false
23	done	Bool	false
24	error	Bool	false
25	errorDevice	Word	16#0
26	memErr	Bool	false
27	memErrDevice	Word	16#0
28	memErrStatus	DWord	16#0

Hinweis

Die Werte der HW-Kennungen ("laddr_interface", "ctrlRec") müssen Sie passend zu Ihrem System angeben.

Z. B die HW-Kennung der Schnittstelle der CPU ("laddr_interface") ermitteln Sie wie folgt:

Öffnen Sie in der CPU den Eintrag "PLC-Variablen > Alle variablen anzeigen > Systemkonstanten" und suchen Sie den Namen <Local~PROFINET-Interface_1> mit dem Datentyp "Hw_Interface". Die zugehörige Zelle "Wert" enthält die HW-Kennung.

Hinweis

Achten Sie auf die verwendeten Schnittstellen!

- Der Port X1 P1 des IO-Device (ET 200MP / "<IO_device_1>") ist mit dem Port X1 P1 der CPU ("<Local>") verbunden.
- Der Port X1 P1 des optionalen IO-Device (ET 200SP / "<IO_device_2>") ist mit dem Port X1 P2 der CPU ("<Local>") verbunden. Vom optionalen IO-Device benötigen Sie zudem die HW-Kennung des Geräts (Datentyp "Hw_Device").

35	Local	Hw_SubModule	257	
36	Local~Common	Hw_SubModule	50	
37	Local(1)	Hw_SubModule	49	
38	OB_Main	OB_PCYCLE	1	
39	Local~Display	Hw_SubModule	54	
40	Local~MC	Hw_SubModule	51	
41	Local~Exec	Hw_SubModule	52	
42	Local~PROFINET_interface_1	Hw_Interface	64	-->laddr
43	Local~PROFINET_interface_1~Port_1	Hw_Interface	65	-->Port 1 local
44	Local~PROFINET_interface_1~Port_2	Hw_Interface	66	-->Port 2 local
45	Local~PROFINET_IO-System	Hw_IoSystem	258	
46	IO_device_1~Proxy	Hw_SubModule	267	
47	IO_device_1~IODevice	Hw_Device	265	
48	IO_device_1~PROFINET_interface	Hw_Interface	268	
49	IO_device_1~PROFINET_interface~Port_1	Hw_Interface	259	-->Port 1 remote
50	IO_device_1~PROFINET_interface~Port_2	Hw_Interface	260	
51	IO_device_1~Head	Hw_SubModule	261	
52	IO_device_1~DI_16x24VDC_HF_1	Hw_SubModule	262	
53	IO_device_1~DQ_8x24VDC_2A_HF_1_1	Hw_SubModule	263	
54	IO_device_2~Proxy	Hw_SubModule	270	
55	IO_device_2~IODevice	Hw_Device	264	-->Used as optional
56	IO_device_2~Head	Hw_SubModule	271	IO device
57	IO_device_2~PROFINET_interface	Hw_Interface	272	
58	IO_device_2~PROFINET_interface~Port_1	Hw_Interface	273	-->Port 2 remote
59	IO_device_2~PROFINET_interface~Port_2	Hw_Interface	274	
60	IO_device_2~DQ_8x24VDC_0_5A_HF_1	Hw_SubModule	276	

Parameter verschalten im FC "SLI_FC_start_reset_ReconfigIOS"

Zum Starten oder Rücksetzen der Anweisung "ReconfigIOSystem" legen Sie den FC "SLI_FC_start_reset_ReconfigIOS" an. Für die Datenweitergabe legen Sie folgende lokale Variablen in dem FC an.

SLI_FC_start_reset_ReconfigIOS

	Name	Data type
1	Input	
2	start	Bool
3	reset	Bool
4	Output	
5	<Add new>	
6	InOut	
7	startDone	Bool
8	Temp	
9	tempStart	Bool
10	resetDone	Bool

Netzwerk 1: Zum Setzen des Eingangsparameters REQ ("getExecute") verschalten Sie die Variable "start".

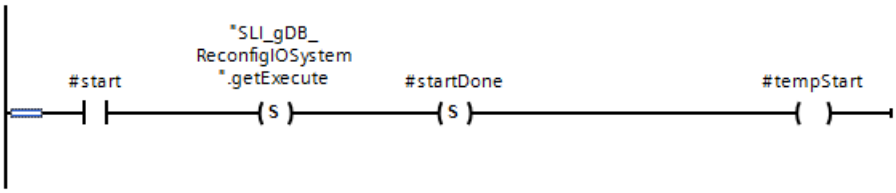

Netzwerk 2: Zum Rücksetzen von "start" legen Sie folgende Verschaltungen an.

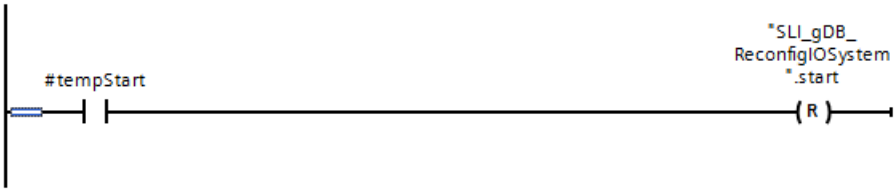

Netzwerk 3: Um alle gesetzten Prozesswerte wieder zurück zu setzen, legen Sie folgende Verschaltungen an. Teil 1 des Netzwerks:

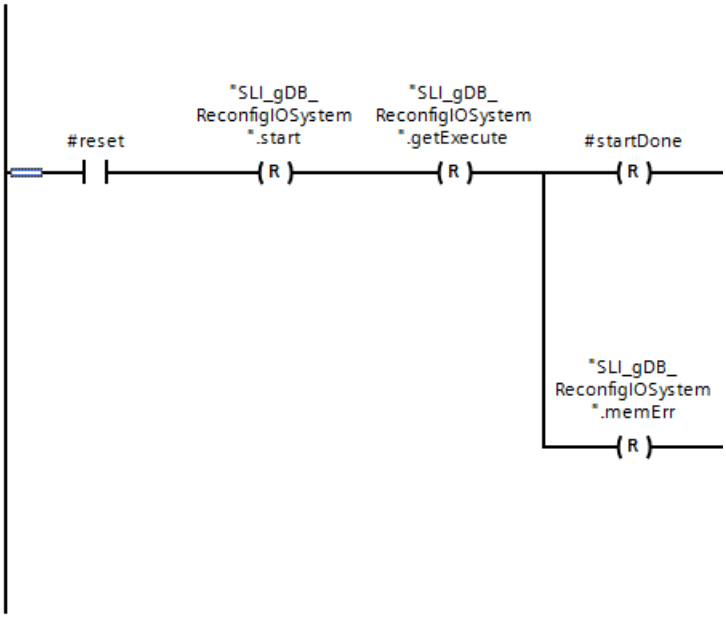

Teil 2 des Netzwerks:

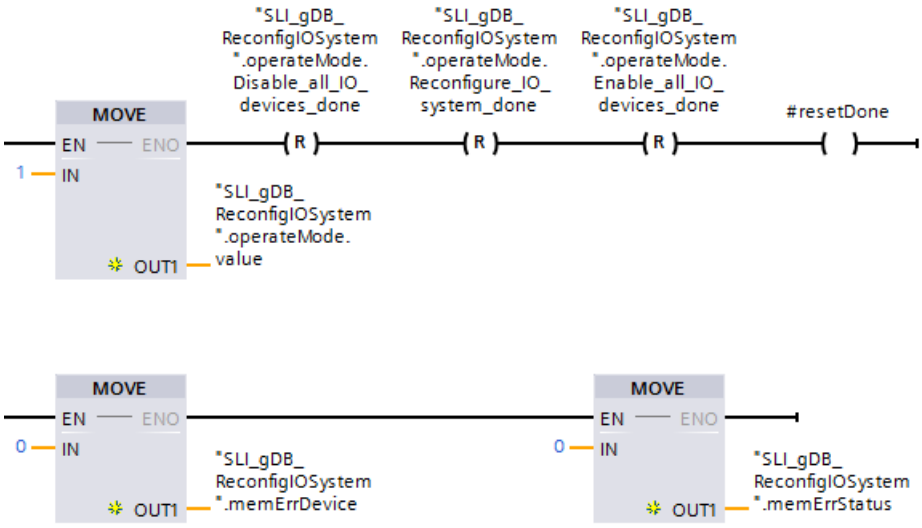

Netzwerk 4: Zum Rücksetzen von "reset" legen Sie folgende Verschaltungen an.

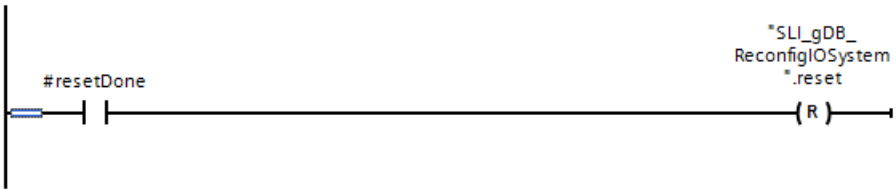

Parameter verschalten im FC "SLI_FC_switchMode_ReconfigIOS"

Für ein sicheres Umkonfigurieren des IO-System lassen Sie alle drei Modi von "ReconfigIOSystem" durchlaufen. Damit Sie die Modi der Anweisung "ReconfigIOSystem" nicht selbst umschalten müssen, legen Sie in dem FC "SLI_FC_switchMode_ReconfigIOS" einen Automatismus für das Umschalten der Modi an.

Netzwerk 1: Zum Speichern der Informationen im Fehlerfall legen Sie folgende Verschaltungen an.

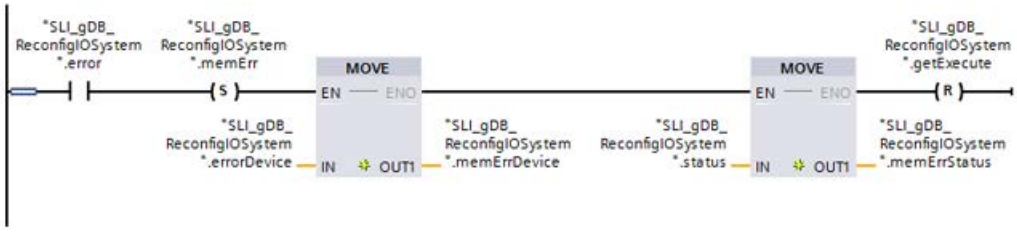

Netzwerk 2: Um im Modus 2 oder Modus 3 die Anweisung "ReconfigIOSystem" automatisch zu starten, legen Sie folgende Verschaltungen an.

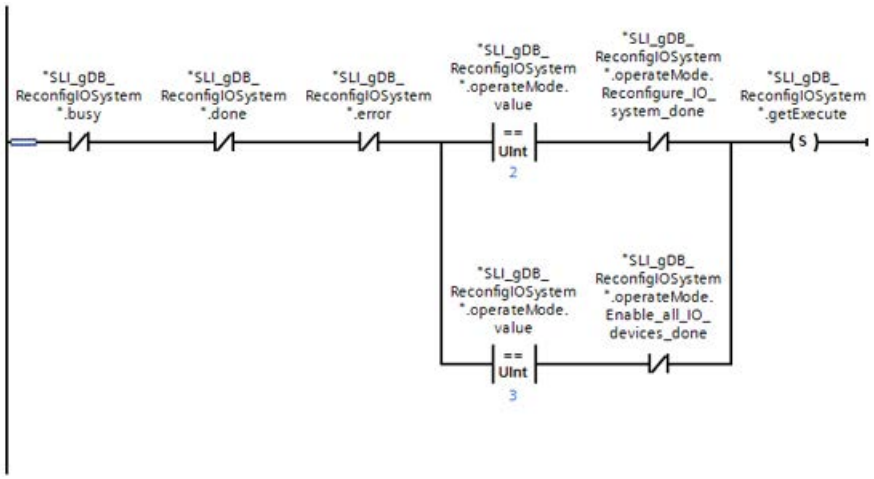

Netzwerk 3: Den Erfolg der Verarbeitung von ReconfigIOSystem im Modus 1 erfassen Sie wie folgt.

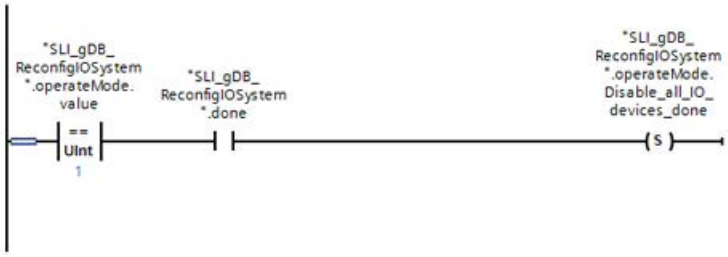

Netzwerk 4: Zum sicheren Umschalten von Modus 1 in den Modus 2 legen Sie folgende Verschaltungen an.

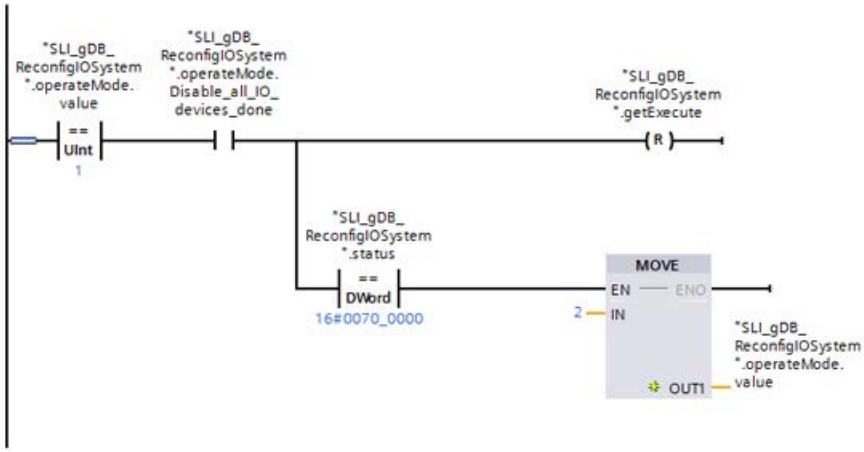

Netzwerk 5: Den Erfolg der Verarbeitung von ReconfigIOSystem im Modus 2 erfassen Sie wie folgt.

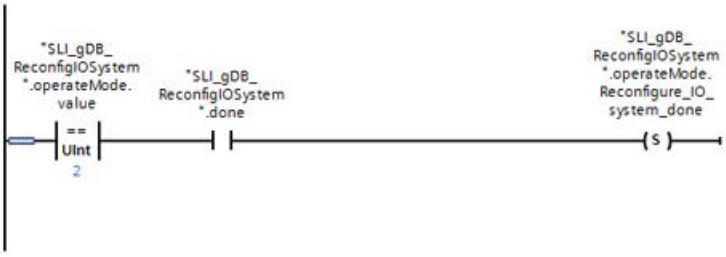

Netzwerk 6: Zum sicheren Umschalten von Modus 2 in den Modus 3 legen Sie folgende Verschaltungen an.

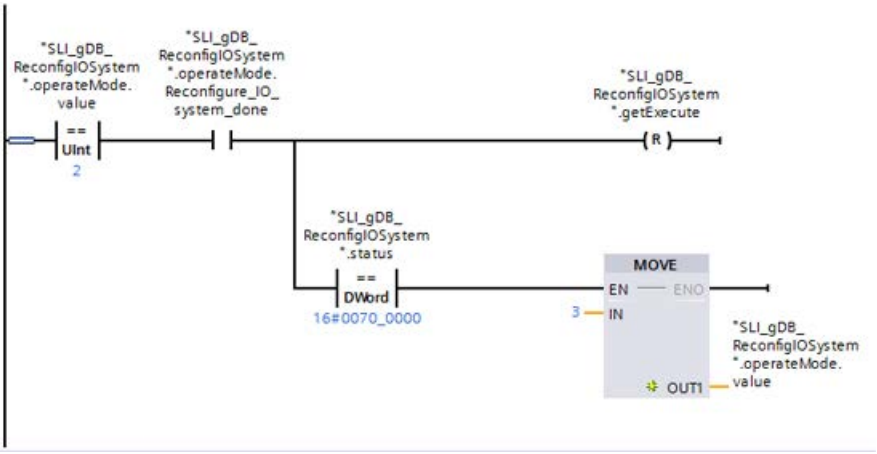

Netzwerk 7: Den Erfolg der Verarbeitung von ReconfigIOSystem im Modus 3 erfassen Sie wie folgt.

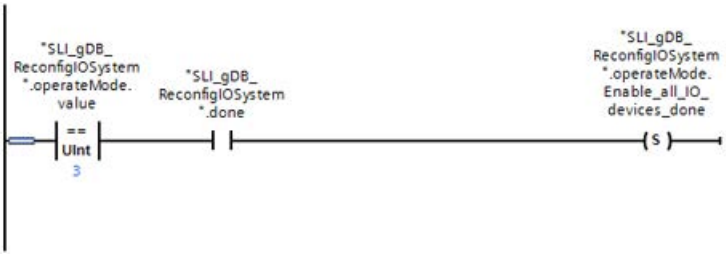

Netzwerk 8: Um die Verarbeitung von ReconfigIOSystem zu beenden, legen Sie folgende Verschaltungen an.

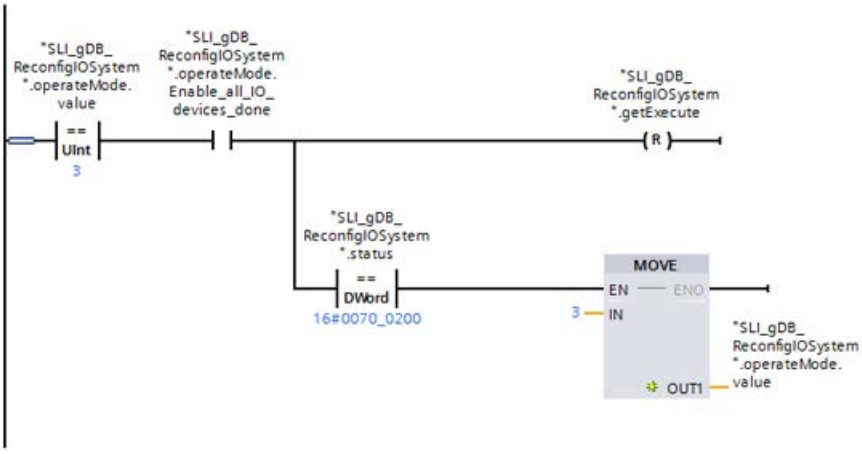

Hinweis

Alternativ könnten Sie nach der Verarbeitung des Modus 3 in den Modus 1 umzuschalten. Tragen Sie dazu bei der Anweisung "MOVE" den Wert "1" ein.

Parameter verschalten im FB "SLI_FB_ReconfigIOSystem"

Die Anweisung "ReconfigIOSystem" rufen Sie in einem FB ("SLI_FB_ReconfigIOSystem") auf. Für die Datenweitergabe legen Sie folgende lokale Variablen in dem FB an. Die Variablen "#recordAoWord" und "#recordStruct" legen Sie als AT-Konstrukt an. Die Variablen müssen die gleiche Größe (wie auch "ctrlRec") besitzen.

7		Static			
8		ReconfigIOSystem_Instance_1	ReconfigIOSyste...		
9		recordAoWord	Array[0..7] of Word		Set in IDB
10		recordStruct AT "recordAoWord"	Struct		Set in IDB
11		Version	Word		Set in IDB
12		Number_of_opt_Devices_used	Word		Set in IDB
13		Activate_opt_Device_1	HW_DEVICE		Set in IDB
14		Number_of_Port_Interconnections_...	Word		Set in IDB
15		Port_Interconnection_1_Local	HW_INTERFACE		Set in IDB
16		Port_Interconnection_1_Remote	HW_INTERFACE		Set in IDB
17		Port_Interconnection_2_Local	HW_INTERFACE		Set in IDB
18		Port_Interconnection_2_Remote	HW_INTERFACE		Set in IDB

Netzwerk 1: Um den Datensatz ("ctrlRec") zu übertragen, legen Sie folgende Verschaltung an.

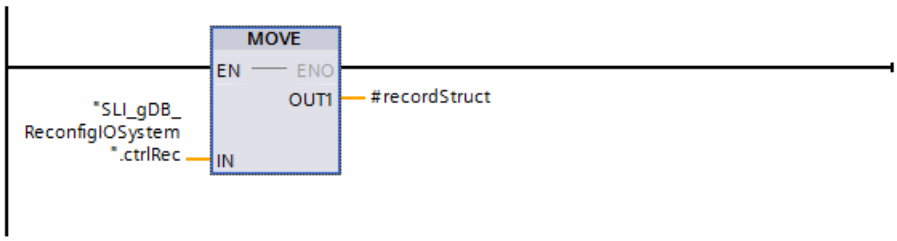

Netzwerk 2: Den FC "SLI_FC_start_reset_ReconfigIOS" verschalten Sie wie folgt.

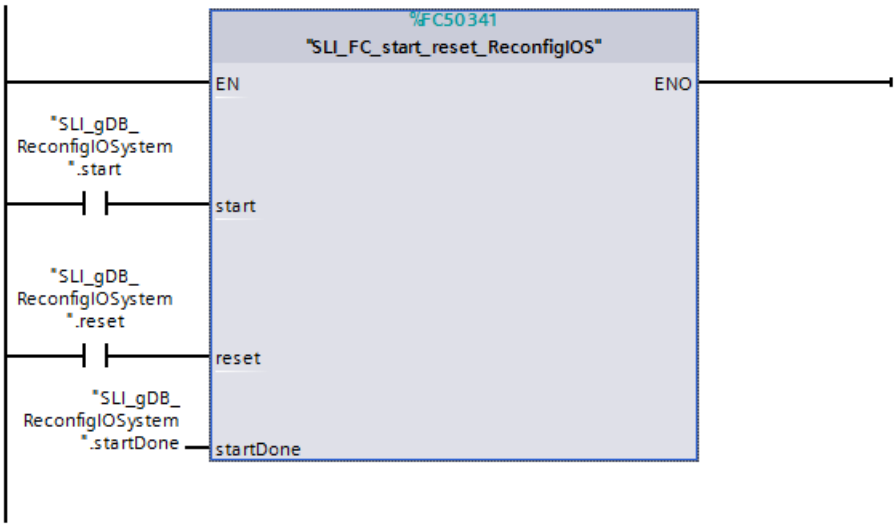

Netzwerk 3: Die Anweisung "ReconfigIOSystem" verschalten Sie wie folgt.

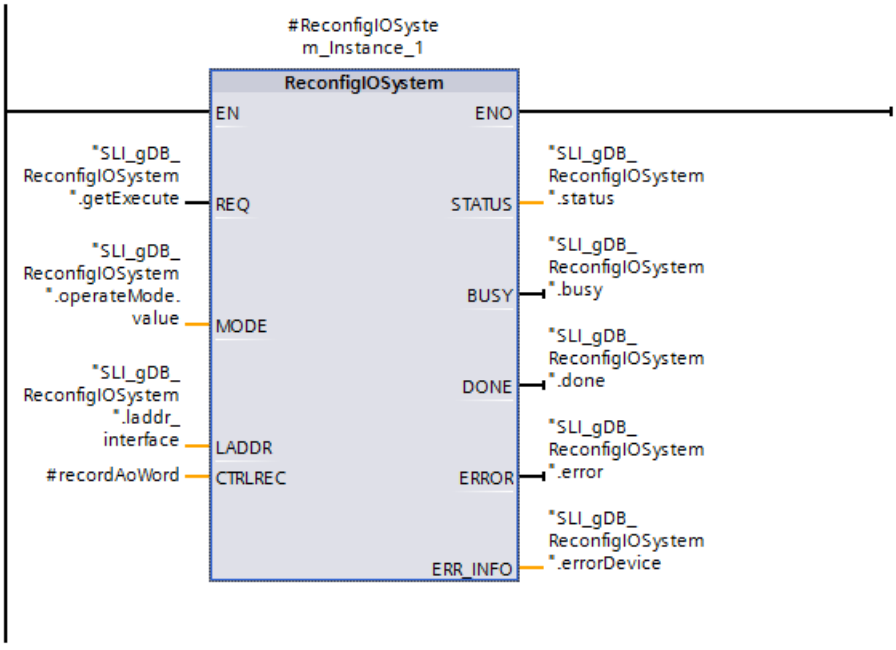

Netzwerk 4: Den FC "SLI_FC_switchMode_ReconfigIOS" verschalten Sie wie folgt.

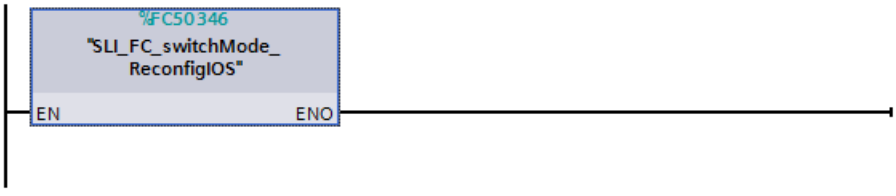

Den FB ("SLI_FB_ReconfigIOSystem") rufen Sie einem OB1 auf.

Ergebnis (FB "SLI_FB_ReconfigIOSystem")

FB Netzwerk 1:

Über eine MOVE-Anweisung werden die Werte des Datensatzes ("ctrlRec") auf ein Array of WORD ("#recordAoWord") übertragen. Zum besseren Verständnis sind die Variablen der Strukturen "ctrlRec" und "#recordStruct" als Klarnamen vertreten.

FB Netzwerk 2:

Wenn der Eingangsparameter start ("start") den Signalzustand "TRUE" liefert, werden die Variable "getExecute" und der Ausgangsparameter startDone ("startDone") auf "TRUE" gesetzt. Durch die Variable "getExecute" auf "TRUE" wird die Verarbeitung der Anweisung "ReconfigIOSystem" gestartet.

Wenn der Eingangsparameter reset ("reset") den Signalzustand "TRUE" liefert, werden der Startbefehl von ReconfigIOSystem, die Prozesswerte der Verarbeitung und die Informationen des Fehlerfalls zurückgesetzt.

FB Netzwerk 3:

Wenn der Eingangsparameter REQ ("getExecute") den Signalzustand "TRUE" liefert, wird die Anweisung "ReconfigIOSystem" gestartet. Über den Eingangsparameter LADDR ("laddr_interface") ruft die Anweisung "ReconfigIOSystem" die Schnittstelle der CPU auf.

Der Anweisung "ReconfigIOSystem" wird der Modus für die Betriebsart über den Eingangsparameter MODE ("operateMode.value") übermittelt. Entsprechend dem Startwert "1" deaktiviert ReconfigIOSystem alle bestehenden IO-Systeme der CPU. Am Ausgangsparameter STATUS ("status") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft im Beispiel ohne Fehler.

Erst im Modus 2 ruft die Anweisung "ReconfigIOSystem" den Datensatz am Eingangsparameter CTRLREC ("#recordAoWord") ab. Dabei werden die Konfigurationswerte für das einzurichtende IO-System von ReconfigIOSystem an die CPU (IO-Controller) übermittelt.

Im Modus 3 aktiviert die Anweisung "ReconfigIOSystem" das IO-System. Das Umschalten der Eingangsparameter REQ ("getExecute") und MODE ("operateMode.value") wird im Netzwerk 4 ("SLI_FC_switchMode_ReconfigIOS") umgesetzt.

SLI_gDB_ReconfigIOSystem				
	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	FALSE
3	startDone	Bool	false	TRUE
4	reset	Bool	false	FALSE
5	getExecute	Bool	false	FALSE
6	operateMode	Struct		
7	value	UInt	1	3
8	Disable_all_IO_devices_done	Bool	false	TRUE
9	Reconfigure_IO_system_done	Bool	false	TRUE
10	Enable_all_IO_devices_done	Bool	false	TRUE
11	laddr_interface	HW_INTERFACE	64	16#0040
12	ctrlRec	Struct		
13	Version	Word	16#0100	16#0100
14	Number_of_opt_Devices_used	Word	1	16#0001
15	Activate_opt_Device_1	HW_DEVICE	264	16#0108
16	Number_of_Port_Interconnections_used	Word	2	16#0002
17	Port_Interconnection_1_Local	HW_INTERFACE	65	16#0041
18	Port_Interconnection_1_Remote	HW_INTERFACE	259	16#0103
19	Port_Interconnection_2_Local	HW_INTERFACE	66	16#0042
20	Port_Interconnection_2_Remote	HW_INTERFACE	273	16#0111
21	status	DWord	16#0	16#0070_0000
22	busy	Bool	false	FALSE
23	done	Bool	false	FALSE
24	error	Bool	false	FALSE
25	errorDevice	Word	16#0	16#0000
26	memErr	Bool	false	FALSE
27	memErrDevice	Word	16#0	16#0000
28	memErrStatus	DWord	16#0	16#0000_0000

FB Netzwerk 4:

Wenn die Anweisung "ReconfigIOSystem" im Modus 1 erfolgreich verarbeitet wurde (die Variable "done" hat den Wert "TRUE"), wird die Variable "operateMode.Disable_all_IO_devices_done" auf "TRUE" gesetzt. Dadurch wird daraufolgend der Eingangsparameter REQ ("getExecute") auf "FALSE" rückgesetzt. Sobald ReconfigIOSystem inaktiv ist ("status" hat den Wert "16#0070_0000") wird der Variable "operateMode.value" der Wert "2" übergeben. ReconfigIOSystem befindet sich danach im Modus 2.

Im Modus 2 wird der Eingangsparameter REQ ("getExecute") automatisch auf "TRUE" gesetzt, wenn für die Anweisung "ReconfigIOSystem" Folgendes gilt:

- ReconfigIOSystem ist inaktiv ("busy" und "done" sind "FALSE").
- Kein Fehler tritt auf ("error" ist "FALSE").
- Der Erfolgszustand des Modus 2 ist nicht erreicht ("operateMode.Reconfigure_IO_system_done" ist "FALSE").

Wenn die Anweisung "ReconfigIOSystem" im Modus 2 erfolgreich verarbeitet wurde (die Variable "done" hat den Wert "TRUE"), wird die Variable

"operateMode.Reconfigure_IO_system_done" auf "TRUE" gesetzt. Nachfolgend passiert Ähnliches wie bei Modus 1:

- Die Variable "getExecute" wird auf "FALSE" rückgesetzt.
- ReconfigIOSystem ("operateMode.value") wird in den Modus 3 gesetzt.

Im Modus 3 wird der Eingangsparameter REQ ("getExecute") automatisch auf "TRUE" gesetzt, wenn für die Anweisung "ReconfigIOSystem" Gleiches gilt wie für Modus 2 - mit einer Ausnahme:

- Der Erfolgszustand des Modus 3 ist nicht erreicht ("operateMode.Enable_all_IO_devices_done" ist "FALSE").

Der Erfolgstatus im Modus 3 (die Variable "done" hat den Wert "TRUE") wird auf gleiche Weise abgefragt wie für Modus 2. Nach der erfolgreichen Verarbeitung verbleibt ReconfigIOSystem im Modus 3 ("operateMode.value" behält den Wert).

Nach beendeter Verarbeitung der drei Modi ist die Anweisung "ReconfigIOSystem" inaktiv ("status" hat den Wert "16#0070_0000"). Das IO-System ist eingerichtet: eine CPU S7-1500 mit einem IO-Device und einem optionalen IO-Device. Beide IO-Devices sind aktiviert.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

ReconfigIOSystem: IO-System umkonfigurieren (Seite 2411)

Weitere

RD_REC: Datensatz von Peripherie lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM von der adressierten Baugruppe. Sie starten den Lesevorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Lesevorgang sofort ausgeführt wird, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Lesevorgang noch nicht abgeschlossen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618). Der gelesene Datensatz wird nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen.

Hinweis

Falls Sie einen Datensatz, dessen Nummer größer als eins ist, aus einer FM oder einem CP auslesen, die Sie vor Februar 1997 erworben haben (im Folgenden als "alte Baugruppen" bezeichnet), dann verhält sich "RD_REC" anders als bei einer neuen Baugruppe. Dieser Sonderfall wird im Abschnitt "Verwendung alter S7-300-FMs und -CPs bei DS-Nr. >1" (siehe unten) behandelt.

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "RD_REC" von den E/A-Baugruppen gelesen werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_REC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 0 bis 240)
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode. Zusätzlich: Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz und bei der Übertragung kein Fehler auftrat.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.
RECORD	Output	VARIANT	E, A, M, D, L	Zielbereich für den gelesenen Datensatz. Sie müssen bei asynchroner Bearbeitung von "RD_REC" darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen dieselbe Längeninformation haben. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RECORD

Hinweis

Wenn Sie sicherstellen wollen, dass immer der gesamte Datensatz gelesen wird, wählen Sie einen Zielbereich mit der Länge 241 Bytes. Bei einer fehlerfreien Datenübertragung steht dann in RET_VAL die tatsächliche Datensatzlänge.

Verwendung alter S7-300-FMs und -CPs bei DS-Nr. > 1

Falls Sie mit der Anweisung "RD_REC" aus einer alten S7-300-FM oder einem alten S7-300-CP einen Datensatz auslesen wollen, dessen Nummer größer als eins ist, müssen Sie Folgendes beachten:

- Falls der Zielbereich größer als die tatsächliche Länge des gewünschten Datensatzes ist, werden keine Daten in RECORD eingetragen. RET_VAL wird mit W#16#80B1 beschrieben.
- Falls der Zielbereich kleiner als die tatsächliche Länge des gewünschten Datensatzes ist, liest die CPU so viele Bytes ab Datensatzanfang, wie in der Längenangabe von RECORD angegeben ist, und trägt sie in RECORD ein. RET_VAL wird mit "0" beschrieben.
- Falls die Längenangabe von RECORD gleich der tatsächlichen Länge des gewünschten Datensatzes ist, liest die CPU den gewünschten Datensatz und trägt ihn in RECORD ein, RET_VAL wird mit "0" beschrieben.

Parameter RET_VAL

- Wenn während der Bearbeitung der Funktion ein Fehler auftrat, enthält der Rückgabewert einen Fehlercode.
- Falls bei der Übertragung kein Fehler auftrat, enthält RET_VAL
  - 0, falls der gesamte Zielbereich mit Daten aus dem selektierten Datensatz gefüllt wurde (Der Datensatz kann aber unvollständig sein.).
  - die Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz.

  Hinweis

  Falls der allgemeine Fehler W#16#8745 auftritt, zeigt dies nur an, dass beim Schreibvorgang auf das Prozessabbild der Zugriff auf mindestens ein Byte nicht möglich war. Der Datensatz wurde ordnungsgemäß von der Baugruppe gelesen und in den E/A-Speicherbereich geschrieben.

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) der folgenden Tabelle sind zwei unterschiedliche Fälle aufgeführt:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A3, 80Cx):
  Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt. Daher ist es sinnvoll, dass Sie die Anweisung erneut aufrufen (wenn nötig: mehrfach).
  Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#809x, 80A0, 80A1, 80Bx):
  Bei dieser Fehlerart behebt sich der Fehler nicht ohne Ihr Zutun. Wenn Sie den Fehler beseitigt haben, ist ein erneuter Aufruf der Weisung erforderlich. Beispiel für einen permanenten Fehler: Falsche Längenangabe in RECORD (W#16#80B1).

Hinweis

Wenn Sie mit "WR_REC (Seite 2433)" Datensätze zu einem DPV1-Slave übertragen oder mit RD_REC Datensätze von einem DPV1-Slave lesen und dieser im DPV1-Mode arbeitet, bewertet der DP-Master die vom Slave erhaltene Fehlerinformation wie folgt:

Wenn die Fehlerinformation in den Bereichen W#16#8000 bis W#16#80FF oder W#16#F000 bis W#16#FFFF liegt, reicht der DP-Master die Fehlerinformation an die Anweisung weiter. Wenn sie außerhalb liegt, gibt er den Wert W#16#80A2 an die Anweisung und suspendiert den Slave.

Zur Beschreibung der von DPV1-Slaves stammenden Fehlerinformationen siehe STATUS[3] Parameter STATUS (Seite 2370).

Parameter RET_VAL für WR_REC und RD_REC

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
0000	Kein Fehler	-
7000	Erstaufruf mit REQ=0: Keine Datenübertragung aktiv; BUSY hat den Wert 0.	-
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert 1.	Dezentrale Peripherie
8090	Angegebene Adresse am Parameter ADDR ungültig.	-
8092	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.	-
8093	Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig.	-
80A0	Negative Quittung beim Lesen von Baugruppe: Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt	Nur bei "RD_REC"
80A1	Negative Quittung beim Schreiben zur Baugruppe: Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt	Nur bei "WR_REC (Seite 2433)"
80A2	• DP-Protokollfehler bei Layer2 (z. B. Slaveausfall oder Busstörungen) • Bei ET200S ist im DPV0-Mode kein Datensatz-Lesen möglich.	Dezentrale Peripherie
80A3	DP-Protokollfehler bei User-Interface/User	Dezentrale Peripherie

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
80A4	Kommunikation am PROFIBUS gestört.	-
80B0	• Anweisung für Baugruppentyp nicht möglich • Baugruppe kennt den Datensatz nicht. • Datensatznummer 241 ist unzulässig. • Bei "WR_REC (Seite 2433)" sind die Datensätze 0 und 1 nicht erlaubt.	-
80B1	Die Längenangabe in Parameter RECORD ist falsch.	• Bei "WR_REC (Seite 2433)": Länge falsch • bei "RD_REC" (nur möglich bei Verwendung alter S7-300-FMs und S7-300-CPs):Angabe > DS-Länge • Bei DPNRM_DG: Angabe < DS-Länge
80B2	Der projektierte Steckplatz ist nicht belegt.	-
80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp	-
80B5	DP-Slave oder Baugruppe ist nicht bereit.	-
80B7	DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Wertes.	Nur bei "RD_REC"
80C0	Bei "WR_REC (Seite 2433)": Die Daten werden nur im STOP-Zustand der CPU geschrieben. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Sie können die Daten nur online mit PG/PC schreiben. Bei "RD_REC": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. In diesem Fall können Sie die Daten nur online mit PG/PC lesen. Bei "DPNRM_DG (Seite 2450)": Keine Diagnosedaten vorhanden.	Bei "WR_REC (Seite 2433)" oder "RD_REC" oder "DPNRM_DG (Seite 2450)"
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.	-
80C2	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.	-
80C3	Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.	-
80C4	Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.	-
80C5	Dezentrale Peripherie nicht verfügbar.	Dezentrale Peripherie
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)	Dezentrale Peripherie
80Ex	Herstellerspezifisch. Bitte sehen Sie in der Dokumentation der Baugruppe nach.	-

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

WR_REC: Datensatz in Peripherie schreiben

Beschreibung

Mit der Anweisung "WR_REC" übertragen Sie den Datensatz RECORD zu der adressierten Baugruppe.

Die zu übertragenden Daten werden aus dem Parameter RECORD beim ersten Aufruf gelesen. Falls die Übertragung des Datensatzes länger als einen Aufruf dauert, so ist bei den Folgeaufrufen der Anweisung (zum gleichen Auftrag) der Inhalt des Parameters RECORD nicht mehr relevant.

Sie starten den Schreibvorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Schreibvorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Schreibvorgang noch nicht abgeschlossen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_REC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	REQ = 1: Anforderung zum Schreiben
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer
RECORD	Input	VARIANT	E, A, M, D, L	Datensatz
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Siehe auch: RD_REC: Datensatz von Peripherie lesen (Seite 2428)

Hinweis

Falls der allgemeine Fehler W#16#8544 auftritt, zeigt dies nur an, dass der Zugriff auf mindestens ein Byte des den Datensatz enthaltenden E/A-Speicherbereichs gesperrt war. Die Datenübertragung wurde fortgesetzt.

DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen

Beschreibung

Mit der Anweisung "DPRD_DAT" lesen Sie konsistente Daten eines EA-Moduls aus.

Die Anweisung kann sowohl für Module der Zentralbaugruppe als auch DP-Normslaves und PROFINET IO-Devices verwendet werden.

Sie benötigen "DPRD_DAT", weil Sie mit den Ladebefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Eingänge zugreifen, maximal vier Bytes zusammenhängend auslesen können. Sie können konsistente Daten ggf. auch über das Prozessabbild der Eingänge einlesen. Ob Ihre CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation. Weitere Informationen zu konsistenten Daten eines DP-Normslaves/ PROFINET IO-Devices finden Sie in dem Abschnitt "Datenkonsistenz (Seite 3589)".

Verwenden Sie die Anweisung "DPRD_DAT" bei Bedarf auch für einen Datenbereich ab 1 Byte. Die Maximallänge der Daten entnehmen Sie der Dokumentation Ihrer CPU (z. B. 64 Byte bei einer S7-1214).

- Über den Parameter LADDR wählen Sie das Modul des DP-Normslave/PROFINET IO-Device aus. Bei einem Zugriffsfehler wird der Fehlercode W#16#8090 gemeldet.
- Über den Parameter RECORD definieren Sie den Zielbereich der gelesenen Daten:
  - Der Zielbereich muss mindestens genauso lang sein wie die Eingänge der selektierten Baugruppe. Wenn der Zielbereich länger ist als die Eingänge der Baugruppe, wird er von vorne mit den Werten der Eingänge beschrieben; bei S7-1200-CPUs bleiben die übrigen Bytes des Zielbereichs unverändert, bei S7-1500-CPUs werden sie mit Nullen aufgefüllt. Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "DPRD_DAT" -Aufruf jeweils nur auf die Daten einer Baugruppe der projektierten Hardware-Kennung zugreifen. Wenn der Zielbereich zu klein gewählt wird, wird am Parameter RET_VAL der Fehlercode 80B1 ausgegeben.
  - Die folgenden Datentypen können verwendet werden: BOOL, BYTE, CHAR, WCHAR, WORD, LWORD, DWORD, INT, UINT, USINT, SINT, LINT, ULINT, DINT, UDINT, REAL, LREAL. Eine Verwendung dieser Datentypen in einer Datenstruktur vom Typ ARRAY oder STRUCT ist ebenfalls zulässig.
  - Die Datentypen STRING und WSTRING werden nicht unterstützt.
  - PLC-Datentypen (UDT) und Systemdatentypen werden unterstützt.
- Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den am Parameter RECORD definierten Zielbereich eingetragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPRD_DAT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, L oder Konstante	Hardware-Kennung des Moduls, aus dem die Daten gelesen werden sollen. Die Hardware-Kennung entnehmen sie den Eigenschaften des Moduls in der Gerätesicht oder den Systemtemkonstanten.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
RECORD	Output	VARIANT	E, A, M, D, L	Zielbereich für die gelesenen Nutzdaten.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8090	• Für die angegebene HW-Kennung haben sie keine Baugruppe projektiert, oder • Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet, oder • Sie haben als Adresse am Parameter LADDR keine HW-Kennung angegeben.
8092	Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.
8093	Für die unter LADDR angegebene HW-Kennung existiert keine DP-Baugruppe / kein PROFINET IO-Device, von der/dem Sie konsistente Daten lesen können. Wenn die über LADDR adressierte Baugruppe nicht über Eingänge verfügt, tritt dieser Fehlercode auch auf.
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.
80B1	Die Länge des angegebenen Zielbereichs am Parameter RECORD ist kürzer als die projektierte Nutzdatenlänge.
80C0	Die Daten wurden noch nicht gelesen.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Parameter STATUS (Seite 2370)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben

Beschreibung

Mit der Anweisung "DPWR_DAT" übertragen Sie die Daten am Parameter RECORD konsistent zum adressierten Modul der Zentralbaugruppe oder des DP-Normslave/PROFINET IO-Device und ggf. ins Prozessabbild (nämlich dann, wenn Sie den betroffenen Adressbereich des DP-Normslaves als Konsistenzbereich in einem Prozessabbild projektiert haben).

Sie benötigen "DPWR_DAT", weil Sie mit den Transferbefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Ausgänge zugreifen, maximal vier Bytes zusammenhängend schreiben können. Wenn erforderlich, schreiben Sie konsistente Daten auch über das Prozessabbild der Ausgänge. Ob Ihre CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation. Nutzen Sie beim Schreiben konsistenter Daten nicht beide Möglichkeiten gleichzeitig: Nutzen Sie entweder "DPWR_DAT" oder das Schreiben über das Prozessabbild der Ausgänge. Weitere Informationen zu konsistenten Daten eines DP-Normslaves/PROFINET IO-Devices finden Sie in dem Abschnitt "Datenkonsistenz (Seite 3589)". Bei einem DP-Normslave mit modularem Aufbau greifen Sie nur auf eine Baugruppe des DP-Slaves zu.

⚠ VORSICHT
Peripheriezugriff Vermeiden Sie bei Verwendung von "DPWR_DAT" Zugriffe auf Peripheriebereiche, deren Teilprozessabbilder mit OB6x-Anbindung (Taktsynchronalarme) zugeordnet sind.

Die Anweisung "DPRD_DAT" kann bei Bedarf auch für einen Datenbereich ab 1 Byte verwendet werden. Die Maximallänge der Daten entnehmen Sie der Dokumentation Ihrer CPU (z. B. 64 Byte bei einer S7-1214).

- Über den Parameter LADDR wählen Sie den DP-Normslave/PROFINET IO-Device aus. Bei einem Zugriffsfehler auf die adressierte Baugruppe wird der Fehlercode 8090 ausgegeben.
- Über den Parameter RECORD definieren Sie den Quellbereich der zu schreibenden Daten:
  - Der Quellbereich muss mindestens genauso lang sein wie die Ausgänge der selektierten Baugruppe. Nur die Ausgänge werden übertragen, die übrigen Bytes werden nicht berücksichtigt. Wenn der Quellbereich am Parameter RECORD länger ist als die Ausgänge der projektierten Baugruppe, werden nur die Daten bis zur maximalen Länge der Ausgänge übertragen. Wenn der Quellbereich am Parameter RECORD kürzer ist als die Ausgänge der projektierten Baugruppe, wird der Fehlercode 80B1 ausgegeben.
  - Die folgenden Datentypen können verwendet werden: BOOL, BYTE, CHAR, WCHAR, WORD, LWORD, DWORD, INT, UINT, USINT, SINT, LINT, ULINT, DINT, UDINT, REAL, LREAL. Eine Verwendung dieser Datentypen in einer Datenstruktur vom Typ ARRAY oder STRUCT ist ebenfalls zulässig.
  - Die Datentypen STRING und WSTRING werden nicht unterstützt.
  - PLC-Datentypen (UDT) und Systemdatentypen werden unterstützt.

Die Datenübertragung erfolgt synchron. Nach Beendigung der Anweisung ist der Schreibvorgang abgeschlossen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPWR_DAT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, L oder Konstante	Hardware-Kennung des Moduls, auf das die Daten geschrieben werden sollen. Die Hardware-Kennung entnehmen Sie den Eigenschaften des Moduls in der Gerätesicht oder den Systemtemkonstanten.
RECORD	Input	VARIANT	E, A, M, D, L	Quellbereich für die zu schreibenden Nutzdaten.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8090	• Für die angegebene HW-Kennung haben sie keine Baugruppe projektiert, oder • Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet, oder • Sie haben keine HW-Kennung am Parameter LADDR angegeben.
8092	Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.
8093	Für die unter LADDR angegebene HW-Kennung existiert keine DP-Baugruppe/kein PROFINET IO-Device, auf die/das Sie konsistente Daten schreiben können. Dieser Fehlercode tritt auch auf, wenn der über LADDR adressierte DP-Normslave/PROFINET IO-Device nicht über Ausgänge verfügt.
80A1	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt
80B1	Die Länge des angegebenen Quellbereichs am Parameter RECORD ist kürzer als die Ausgänge des projektierten DP-Normslave/PROFINET IO-Device.
80C1	Die Daten des vorangegangenen Schreibauftrags sind von dem DP-Normslave/PROFINET IO-Device noch nicht bearbeitet.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Parameter STATUS (Seite 2370)

iDevice / iSlave

RCVREC: Datensatz empfangen

Beschreibung

Ein I-Device kann vom übergeordneten Controller einen Datensatz empfangen. Der Empfang erfolgt im Anwendungsprogramm mit der Anweisung "RCVREC" (receive record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für einen Datensatzempfang vorliegt.
- Bereitstellen des Datensatzes an den Ausgangsparametern.
- Verschicken einer Antwort an den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).

Das I-Device muss sich im Betriebszustand RUN oder im Betriebszustand ANLAUF befinden.

Mit MLEN geben Sie vor, wie viele Bytes Sie maximal empfangen möchten. Wählen Sie den Zielbereich RECORD mindestens MLEN Bytes lang.

Wenn ein Datensatz empfangen wurde (MODE=1 oder MODE=2), wird im Ausgangsparameter NEW angezeigt, dass der Datensatz in RECORD gespeichert wurde. Achten Sie dabei auf eine ausreichende Länge von RECORD. Der Ausgangsparameter LEN enthält die tatsächliche Länge des empfangenen Datensatzes in Bytes.

Für die positive Antwort an den übergeordneten Controller beschreiben Sie CODE1 und CODE2 mit Null. Wenn der empfangene Datensatz abgelehnt wird, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

Hinweis

Wenn das I-Device eine Anforderung für einen Datensatzempfang erhalten hat, müssen Sie die Zustellung dieser Anforderung innerhalb einer bestimmten Zeitdauer erkannt haben. Nach dem Erkennen müssen Sie dem übergeordneten Controller innerhalb derselben Zeitdauer eine Antwort geschickt haben. Andernfalls tritt beim I-Device ein Timeout-Fehler auf, der zur Folge hat, dass das Betriebssystem des I-Device eine negative Antwort an den übergeordneten Controller schickt. Den Wert für die Zeitdauer können Sie der Beschreibung von PRVREC (Seite 2440) entnehmen.

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

Betriebsarten

Die Betriebsart der Anweisung "RCVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

MODE	Bedeutung
0	Überprüfen, ob eine Anforderung für einen Datensatzempfang vorliegt Falls auf dem I-Device ein Datensatz vom übergeordneten Controller vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, SUBSLOT, INDEX und LEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung.
1	Empfangen eines Datensatzes für irgendeinen Subslot des I-Device Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.
2	Empfangen eines Datensatzes für einen bestimmten Subslot des I-Device Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.
3	Positive Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, akzeptiert den vorliegenden Datensatz und schickt eine positive Quittung an den übergeordneten Controller.
4	Negative Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, lehnt den vorliegenden Datensatz ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.

Hinweis

Nach dem Eintreffen eines Datensatzes (NEW=1) müssen Sie "RCVREC" noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2 (muss innerhalb einer Sekunde nach Eintreffen der Anforderung beendet sein, sonst wird die Anweisung des anfordernden IO-Controllers negativ quittiert, d. h. Parameter STATUS (Feldelement STATUS[3] entspricht Error_code_1): C2 (Resource busy))
- Zweiter Aufruf mit MODE=3 oder MODE=4 (muss innerhalb von 10 s nach erstem Aufruf mit MODE=1 oder MODE=2 beendet sein. Nach abgelaufener Wartezeit erhält der übergeordnete IO-Controller im Parameter STATUS seiner anfordernden Anweisung (RDREC) folgenden Fehler: Parameter STATUS (Feldelement STATUS[3] entspricht Error_code_1): A7 (Application busy))

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCVREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	HW_SUBMODULE	E, A, M, D, L oder Konstante	Subslot im Transferbereich des I-Device für den zu erhaltenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen.
MLEN	Input	INT*	E, A, M, D, L oder Konstante	Maximale Länge des zu empfangenden Datensatzes in Bytes.
CODE1	Input	BYTE	E, A, M, D, L oder Konstante	"0" (bei MODE=3) bzw. Error Code 1 (bei MODE=4)
CODE2	Input	BYTE	E, A, M, D, L oder Konstante	"0" (bei MODE=3) bzw. Error Code 2 (bei MODE=4)
NEW	Output	BOOL	E, A, M, D, L	• MODE=0: Neuer Datensatz wurde empfangen • MODE=1 oder 2: Datensatz wurde in RECORD übertragen
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation
SLOT	Output	HW_SUBMODULE	E, A, M, D, L	Identisch zu F_ID
SUBSLOT	Output	HW_SUBMODULE	E, A, M, D, L	Identisch zu F_ID
INDEX	Output	UINT	E, A, M, D, L	Nummer des empfangenen Datensatzes
LEN	Output	UINT	E, A, M, D, L	Länge des empfangenen Datensatzes
RECORD	InOut	VARIANT	E, A, M, D, L	Zielbereich für den empfangenen Datensatz.
* Verwenden Sie in der Programmiersprache AWL den Datentyp UINT.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 2370)

PRVREC: Datensatz bereitstellen

Beschreibung

Ein I-Device kann vom übergeordneten Controller eine Anforderung erhalten, einen Datensatz bereitzustellen. Die Bereitstellung erfolgt im Anwenderprogramm mit der Anweisung "PRVREC" (provide record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für eine Datensatzbereitstellung vorliegt.
- Übertragung des angeforderten Datensatzes an den übergeordneten Controller.
- Schicken einer Antwort an den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).

Die Anweisung PRVREC kann nur im Betriebszustand RUN oder im Betriebszustand ANLAUF im I-Device ausgeführt werden.

Wenn sich das I-Device im Betriebszustand STOP befindet, kann PRVREC nicht ausgeführt werden und der übergeordnete IO-Controller erhält im STATUS seiner anfordernden Anweisung (RDREC) folgenden Fehler:

Parameter STATUS (Feldelement STATUS[3] entspr. Error_code_1): B5 (State conflict).

Wenn sich das I-Device in ANLAUF oder RUN befindet, hat das I-Device zunächst 1s Zeit, um nach dem Empfang des Read-Requests mit PRVREC(MODE=1, …) oder PRVREC(MODE=2, …) zu antworten. Sonst wird an den übergeordneten IO-Controller der Fehler C2 (Resource busy) gemeldet.

Nach Aufruf von PRVREC (Mode=1/2) hat das I-Device 10 s Zeit, um die Anweisung mit PRVREC (MODE=3) oder PRVREC (MODE=4) abzuschließen. Sonst wird an den übergeordneten IO-Controller der Fehler A7 (Application busy) gemeldet.

Mit LEN geben Sie vor, wie viele Bytes der zu versendende Datensatz maximal haben soll. Wählen Sie den Zielbereich RECORD mindestens LEN Bytes lang.

Wenn eine Anforderung für eine Datensatzbereitstellung vorliegt (MODE=0), wird der Ausgangsparameter NEW auf TRUE gesetzt.

Wenn die Anforderung für eine Datensatzbereitstellung akzeptiert wird, beschreiben Sie für die positive Antwort an den übergeordneten Controller RECORD mit dem angeforderten Datensatz und CODE1 und CODE2 mit Null. Wenn die Anforderung für eine Datensatzbereitstellung abgelehnt werden soll, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

Betriebsarten

Die Betriebsart der Anweisung "PRVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

MODE	Bedeutung
0	Überprüfen, ob eine Anforderung für eine Datensatzbereitstellung vorliegt Falls auf dem I-Device eine Anforderung vom übergeordneten Controller für eine Datensatzbereitstellung vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, SUBSLOT, INDEX und RLEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung.
1	Empfangen einer Anforderung für eine Datensatzbereitstellung für irgendeinen Subslot des I-Device Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.

MODE	Bedeutung
2	Empfangen einer Anforderung für eine Datensatzbereitstellung für einen bestimmten Subslot des I-Device Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.
3	Datensatz bereitstellen und positive Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, stellt den angeforderten Datensatz in RECORD zur Verfügung und schickt eine positive Quittung an den übergeordneten Controller.
4	Negative Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, lehnt diese Anforderung ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.

Hinweis

Nach dem Eintreffen einer Anforderung (NEW=1) müssen Sie die Anweisung noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2 (muss innerhalb einer Sekunde nach Eintreffen der Anforderung beendet sein, sonst wird die Anweisung des anfordernden IO-Controllers negativ quittiert, d. h. Parameter STATUS (Feldelement STATUS[3] entspricht Error_code_1): C2 (Resource busy))
- Zweiter Aufruf mit MODE=3 oder MODE=4 (muss innerhalb von 10 s nach erstem Aufruf mit MODE=1 oder MODE=2 beendet sein. Nach abgelaufener Wartezeit erhält der übergeordnete IO-Controller im Parameter STATUS seiner anfordernden Anweisung (RDREC) folgenden Fehler: Parameter STATUS (Feldelement STATUS[3] entspricht Error_code_1): A7 (Application busy))

Tipp

Programmieren Sie den ersten Aufruf und den zweiten Aufruf direkt hintereinander!

Hintergrund: Nach dem ersten Aufruf der Anweisung mit Mode 1 bzw. Mode 2 kann die CPU keine weitere Instanz der Anweisung, d. h. keinen weiteren Datensatz-Auftrag bearbeiten. Erst nachdem der zweite Aufruf mit Mode 3 bzw. Mode 4 bearbeitet wurde, kann die CPU einen anderen Datensatz-Auftrag bearbeiten. Wenn nach dem ersten Aufruf der Anweisung mit Mode 1 bzw. Mode 2 eine weitere Instanz der Anweisung aufgerufen wird, liefert der Status den Wert 0xC080C300 (= Fehler, Resource unavailable).

Beispiel: Sie verwenden sowohl die Funktionen Asset Management Datensatz als auch die Funktion PROFIenergy im I-Device-Anwenderprogramm. Asset Management nutzt die Anweisung PRVREC und PROFIenergy nutzt die Anweisung PE_I_DEV. PE_I_DEV nutzt intern ebenfalls die Anweisungen PRVREC und RCVREC.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PRVREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Subslot im Transferbereich des I-Device für den zu versendenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen. Wenn Sie PRVREC aufrufen, um einen Asset-Management Datensatz an den IO-Controller zu liefern (Datensatz-Nr. 0xF880), dann müssen Sie F_ID = 0 setzen. Mit F_ID = 0 wird das Device adressiert. (Nur für MODE=2 relevant. Achtung: Bei Aufruf mit MODE=0 wird für INDEX=0xF880 von PRVREC für SLOT und SUBSLOT der Wert 0 zurückgeliefert.)
CODE1	Input	BYTE	E, A, M, D, L oder Konstante	"0" (bei MODE=3) bzw. Error Code 1 (bei MO-DE=4)
CODE2	Input	BYTE	E, A, M, D, L oder Konstante	"0" (bei MODE=3) bzw. Error Code 2 (bei MO-DE=4)
LEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Länge des zu versendenden Datensat-zes in Bytes
NEW	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde vom übergeordneten Controller angefordert.
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation
SLOT	Output	HW_SUBMO-DULE	E, A, M, D, L	Identisch zu F_ID
SUBSLOT	Output	HW_SUBMO-DULE	E, A, M, D, L	Identisch zu F_ID
INDEX	Output	UINT	E, A, M, D, L	Nummer des zu versendenden Datensatzes
RLEN	Output	UINT	E, A, M, D, L	Länge des zu versendenden Datensatzes
RECORD	InOut	VARIANT	E, A, M, D, L	Bereitgestellter Datensatz

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 2370)

PROFIBUS

DPSYC_FR: DP-Slaves synchronisieren / Eingänge einfrieren

Beschreibung

Mit der Anweisung können Sie eine oder mehrere Gruppen von DP-Slaves synchronisieren.

Dazu schicken Sie eines der folgenden Steuerkommandos oder eine Kombination davon an die betroffenen Gruppen:

- SYNC (gleichzeitiges Ausgeben und Einfrieren von Ausgangszuständen bei den DP-Slaves)
- UNSYNC (hebt das Steuerkommando SYNC wieder auf)
- FREEZE (Einfrieren von Eingangszuständen bei den DP-Slaves und Einlesen der eingefrorenen Eingänge)
- UNFREEZE (hebt das Steuerkommando FREEZE wieder auf)

Bevor Sie die oben genannten Steuerkommandos abschicken, müssen Sie die DP-Slaves per Projektierung in Gruppen eingeteilt haben. Sie müssen wissen, welcher DP-Slave welcher Gruppe mit welcher Nummer zugeordnet ist und welche Eigenschaften hinsichtlich SYNC-/FREEZE-Verhalten die einzelnen Gruppen haben.

Hinweis

Beachten Sie, dass die Steuerkommandos SYNC und FREEZE auch bei einem Neustart/Kaltstart ihre Gültigkeit behalten.

Beachten Sie außerdem, dass zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag bzw. nur ein FREEZE/UNFREEZE-Auftrag angestoßen sein darf.

Arbeitsweise

"DPSYC_FR" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "DPSYC_FR" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Identifikation eines Auftrags

Falls Sie einen SYNC-/FREEZE-Auftrag angestoßen haben und Sie "DPSYC_FR" erneut aufrufen, bevor dieser beendet wurde, dann hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Stimmen die Eingangsparameter LADDR, GROUP und MODE überein, so gilt der Aufruf als Folgeaufruf.

Ausgänge von DP-Baugruppen beschreiben

Das Beschreiben der Ausgänge von DP-Baugruppen wird angestoßen durch

- Transfer-Befehle auf DP-Peripherie,
- das Schreiben des Prozessabbilds der Ausgänge in die Baugruppen (durch das Betriebssystem am Ende des OB 1 oder durch Aufruf der Anweisung "UPDAT_PO (Seite 2292)"),
- Aufruf der Anweisung "DPWR_DAT (Seite 2436)".

Im Normalfall überträgt der DP-Master die Ausgangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) an die Ausgänge der DP-Slaves.

Wenn Sie bestimmte Ausgangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig auf die Ausgänge an den Prozess ausgeben wollen, schicken Sie mithilfe der Anweisung "DPSYC_FR" das Steuerkommando SYNC an den zugehörigen DP-Master.

Was bewirkt SYNC?

Mit dem Steuerkommando SYNC werden die DP-Slaves der genannten Gruppen in den Sync-Modus geschaltet. Der DP-Master überträgt die aktuellen Ausgangsdaten und veranlasst die betroffenen DP-Slaves, die Ausgänge einzufrieren. Bei den folgenden Ausgabetelegrammen speichern die DP-Slaves die Ausgangsdaten in einem internen Puffer; der Zustand der Ausgänge bleibt unverändert.

Nach jedem Steuerkommando SYNC legen die DP-Slaves der selektierten Gruppen die Ausgangsdaten ihres internen Puffers auf die Ausgänge an den Prozess.

Wenn Sie mithilfe der Anweisung "DPSYC_FR" das Steuerkommando UNSYNC absetzen, werden die Ausgänge wieder zyklisch aktualisiert.

Hinweis

Falls sich DP-Slaves der ausgewählten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den SYNC-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

Eingänge von DP-Baugruppen lesen

Die Eingangsdaten von DP-Baugruppen werden gelesen

- mit Ladebefehlen auf DP-Peripherie,
- bei der Aktualisierung des Prozessabbilds der Eingänge (durch das Betriebssystem am Anfang des OB 1 oder durch Aufruf der Anweisung "UPDAT_PI (Seite 2289)"),
- durch Aufruf der Anweisung "DPRD_DAT (Seite 2434)".

Im Normalfall erhält der DP-Master diese Eingangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) von seinen DP-Slaves und stellt sie der CPU zur Verfügung.

Wenn Sie bestimmte Eingangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig vom Prozess einlesen wollen, schicken Sie mithilfe der Anweisung "DPSYC_FR" das Steuerkommando FREEZE an den zugehörigen DP-Master.

Was bewirkt FREEZE?

Mit dem Steuerkommando FREEZE werden die betroffenen DP-Slaves in den Freeze-Modus geschaltet. Der DP-Master veranlasst die betroffenen DP-Slaves, den aktuellen Zustand der Eingänge einzufrieren. Anschließend überträgt er die eingefrorenen Daten in den Eingangsbereich der CPU.

Nach jedem Steuerkommando FREEZE frieren die DP-Slaves den Zustand der Eingänge erneut ein.

Wenn Sie mithilfe der Anweisung "DPSYC_FR" das Steuerkommando UNFREEZE absetzen, erhält der DP-Master wieder zyklisch den aktuellen Zustand der Eingänge.

Hinweis

Falls sich DP-Slaves der ausgewälten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den Freeze-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

Datenkonsistenz

Aufgrund der asynchronen Arbeitsweise und der Unterbrechbarkeit durch höhere Prioritätsklassen sollten Sie bei Einsatz der Anweisung "DPSYC_FR" darauf achten, dass die Prozessabbilder konsistent zu den tatsächlichen Ein- und Ausgängen der Peripherie sind.

Wenn Sie eine der im Folgenden genannten Konsistenzregeln beachten, ist dies gewährleistet:

- Definieren Sie für die "SYNC-Ausgänge" und die "FREEZE-Eingänge" geeignete Teilprozessabbilder (nur möglich bei S7-400). Rufen Sie die Anweisung "UPDAT_PO (Seite 2292)" unmittelbar vor dem jeweiligen Erstaufruf eines SYNC-Auftrags auf. Rufen Sie die Anweisung "UPDAT_PI (Seite 2289)" unmittelbar nach dem jeweiligen Letztaufruf eines FREEZE-Auftrags auf.
- Alternativ dazu: Verwenden Sie für Ausgänge, die von einem SYNC-Auftrag betroffen sind, und für Eingänge, die von einem FREEZE-Auftrag betroffen sind, nur direkte Peripheriezugriffe. Sie dürfen bei einem laufenden SYNC-Auftrag die betroffenen Ausgänge nicht beschreiben und bei einem laufenden FREEZE-Auftrag die betroffenen Eingänge nicht einlesen.

Einsatz von DPWR_DAT und DPRD_DAT

Falls Sie die Anweisung "DPWR_DAT (Seite 2436)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Ausgänge einen SYNC-Auftrag abschicken.

Falls Sie die Anweisung "DPRD_DAT (Seite 2434)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Eingänge einen FREEZE-Auftrag abschicken.

Anlauf und "DPSYC_FR"

Das Absetzen der Steuerkommandos SYNC und FREEZE in den Anlauf-OBs liegt in alleiniger Verantwortung des Anwenders.

Falls die Ausgänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im SYNC-Modus arbeiten sollen, müssen Sie im Anlauf diese Ausgänge initialisieren und die Anweisung "DPSYC_FR" mit dem Steuerkommando SYNC vollständig abarbeiten.

Falls die Eingänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im FREEZE-Modus arbeiten sollen, müssen Sie im Anlauf für diese Eingänge die Anweisung "DPSYC_FR" mit dem Steuerkommando FREEZE vollständig abarbeiten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPSYC_FR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Pegelgetriggerter Steuerparameter REQ=1: Anstoß des SYNC-/FREEZE-Auftrags
LADDR	Input	HW_INTER-FACE	E, A, M, D, L oder Konstante	Hardware-Kennung der DP-Master-Schnittstelle Die Nummer kann in der Netzsicht aus den Eigenschaften der DP-Master-Schnittstelle oder dem Register "Systemkonstanten" der Standard-Variablentabelle übernommen werden.
GROUP	Input	BYTE	E, A, M, D, L oder Konstante	Gruppenauswahl Bit 0 = 1: Gruppe 1 ausgewählt Bit 1 = 1: Gruppe 2 ausgewählt : Bit 7 = 1: Gruppe 8 ausgewählt Pro Auftrag können Sie mehrere Gruppen auswählen. Der Wert B#16#0 ist nicht zugelassen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung (Codierung gemäß EN 50 170 Volume 2, PROFIBUS) Bit 0: reserviert (Wert 0) Bit 1: reserviert (Wert 0) Bit 2: • = 1: UNFREEZE wird ausgeführt • = 0: keine Bedeutung Bit 3: • = 1: FREEZE wird ausgeführt • = 0: keine Bedeutung Bit 4: • = 1: UNSYNC wird ausgeführt • = 0: keine Bedeutung Bit 5: • = 1: SYNC wird ausgeführt • = 0: keine Bedeutung Bit 6: reserviert (Wert 0) Bit 7: reserviert (Wert 0) Mögliche Werte: • bei genau einer Kennung pro Auftrag: – B#16#04 (UNFREEZE) – B#16#08 (FREEZE) – B#16#10 (UNSYNC) – B#16#20 (SYNC) • bei mehreren Kennungen pro Auftrag: – B#16#14 (UNSYNC, UNFREEZE) – B#16#18 (UNSYNC, FREEZE) – B#16#24 (SYNC, UNFREEZE) – B#16#28 (SYNC, FREEZE)
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode. Sie müssen RET_VAL nach jedem Bausteindurchlauf auswerten.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der SYNC-/FREEZE-Auftrag ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Hinweis

Falls Sie auf DPV1-Slaves zugreifen, können Fehlerinformationen dieser Slaves vom DP-Master an die Anweisung weitergereicht werden. Zur Beschreibung dieser Fehlerinformationen siehe STATUS[3], Parameter STATUS (Seite 2370).

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Der über LADDR, GROUP und MODE festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert 0.
7001	Erstaufruf mit REQ=1. Der über LADDR, GROUP und MODE festgelegte Auftrag wurde angestoßen; BUSY hat den Wert 1.
7002	Zwischenaufruf (REQ irrelevant). Der aktivierte SYNC-/FREEZE-Auftrag läuft noch; BUSY hat den Wert 1.
8090	Die über LADDR ausgewählte Baugruppe ist kein DP-Master.
8093	Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig (Projektierung oder Version des DP-Masters).
8094	Parameter GROUP falsch
8095	Parameter MODE falsch
80A4	Kommunikation am PROFIBUS gestört.
80B0	Die über GROUP ausgewählte Gruppe ist nicht projektiert.
80B1	Die über GROUP ausgewählte Gruppe ist dieser CPU nicht zugeordnet.
80B2	Der über MODE spezifizierte SYNC-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.
80B3	Der über MODE spezifizierte FREEZE-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.
80C2	Temporärer Ressourcenmangel des DP-Masters: Der DP-Master bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.
80C3	Dieser SYNC-/UNSYNC-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm.
80C4	Dieser FREEZE-/UNFREEZE-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein FREEZE-/UNFREEZE-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm.
80C5	Kurzschluss unmittelbar an der DP-Schnittstelle
80C6	Auftragsabbruch wegen Peripherieabwurf durch die CPU
80C7	Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DPNRM_DG: Diagnosedaten eines DP-Slaves lesen

Beschreibung

Mit der Anweisung "DPNRM_DG" lesen Sie die aktuellen Diagnosedaten eines DP-Slaves in der Form, wie sie durch EN 50 170 Volume 2, PROFIBUS festgelegt sind.

Den prinzipiellen Aufbau der Slave-Diagnose entnehmen Sie folgender Tabelle, weitergehende Informationen den Handbüchern zu den DP-Slaves.

Byte	Bedeutung
0	Stationsstatus 1
1	Stationsstatus 2
2	Stationsstatus 3
3	Master-Stationsnummer
4	Herstellerkennung (high byte)
5	Herstellerkennung (low byte)
6 ...	Weitere slavespezifische Diagnose

Die gelesenen Daten werden nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen. Sie starten den Lesevorgang, indem Sie beim Aufruf der Anweisung "DPNRM_DG" den Eingangsparameter REQ mit "1" belegen.

Arbeitsweise

Der Lesevorgang wird asynchron ausgeführt. Er kann sich über mehrere Aufrufe erstrecken. Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPNRM_DG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen
LADDR	Input	HW_DPSLAVE	E, A, M, D, L oder Konstante	Hardware-Kennung des DP-Slaves Hinweis: Die Kennung muss hexadezimal angegeben werden. Z. B. Hardware-Kennung 1022 bedeutet: LADDR:=W#16#3FE.
RET_VAL	Return	DINT, INT, LREAL, REAL	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode. Falls kein Fehler auftrat, steht in RET_VAL die Länge der tatsächlich übertragenen Daten.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RECORD	Output	VARIANT	E, A, M, D, L	Zielbereich für die gelesenen Diagnosedaten. Nur der Datentyp BYTE ist zulässig. Die Mindestlänge des zu lesenden Datensatzes bzw. des Zielbereichs beträgt 6. Die Maximallänge des zu lesenden Datensatzes beträgt 240. Bei Normslaves, bei denen die Anzahl der Normdiagnosedaten größer als 240 Byte ist und maximal 244 Bytes beträgt, werden die ersten 240 Byte in den Zielbereich übertragen. Danach wird das entsprechende Overflow-Bit in den Daten gesetzt.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Informationen zur Datentypkonvertierung in den einzelnen Programmiersprachen finden Sie unter "Übersicht über die Datentypkonvertierungen (Seite 578)".

Parameter RECORD

Die CPU bewertet die tatsächliche Länge der gelesenen Diagnosedaten:

Falls die Längenangabe von RECORD

- kleiner als die Anzahl der gelieferten Daten ist, werden die Daten verworfen. In RET_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich der Anzahl der gelieferten Daten ist, werden die Daten in den Zielbereich übernommen, und in RET_VAL wird die tatsächliche Länge als positiver Wert eingetragen.

Hinweis

Sie müssen darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen, die zu einem Auftrag gehören, übereinstimmen.

Ein Auftrag ist eindeutig festgelegt durch den Eingangsparameter LADDR.

Normslaves mit mehr als 240 Bytes Diagnosedaten

Bei Normslaves, bei denen die Anzahl der Normdiagnosedaten zwischen 241 und 244 Bytes liegt, müssen Sie folgendes beachten:

Falls die Längenangabe von RECORD

- kleiner als 240 Byte ist, werden die Daten verworfen, und in RET_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich 240 Byte ist, werden die ersten 240 Byte der Normdiagnosedaten in den Zielbereich übertragen und das entsprechende Overflow-Bit in den Daten gesetzt.

Parameter RET_VAL

- Wenn während der Bearbeitung der Funktion ein Fehler auftrat, enthält der Rückgabewert einen Fehlercode.
- Falls bei der Übertragung kein Fehler auftrat, enthält RET_VAL die Länge der gelesenen Daten in Bytes als positive Zahl.

Hinweis

Die Anzahl der gelesenen Daten ist bei einem DP-Slave von seinem Diagnosezustand abhängig.

Zur Auswertung der Fehlerinformationen des Parameters RET_VAL siehe folgende Tabelle.

Fehlercode (W#16#...)	Erläuterung	Einschränkung
7000	Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".	-
7001	Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".	Dezentrale Peripherie
8090	Angegebene Adresse am Parameter LADDR ungültig.	-
8093	Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig.	-
80A2	• DP-Protokollfehler bei Layer2 (z. B. Slaveausfall oder Busstörungen) • Bei ET200S ist im DPV0-Mode kein Datensatz-Lesen möglich.	Dezentrale Peripherie
80A3	DP-Protokollfehler bei User-Interface/User	Dezentrale Peripherie
80A4	Kommunikation am PROFIBUS gestört.	Dezentrale Peripherie
80B0	• Anweisung für Baugruppentyp nicht möglich • Baugruppe kennt den Datensatz nicht. • Datensatznummer 241 ist unzulässig. • Bei "WR_REC (Seite 2326)" sind die Datensätze 0 und 1 nicht erlaubt.	-
80B1	Die Längenangabe in Parameter RECORD ist falsch.	Angabe < DS-Länge
80B2	Der projektierte Steckplatz ist nicht belegt.	-
80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp	-
80C0	Es liegen keine Diagnosedaten vor.	-
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.	-
80C2	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.	-
80C3	Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.	-
80C4	Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.	-
80C5	Dezentrale Peripherie nicht verfügbar.	Dezentrale Peripherie

Fehlercode (W#16#...)	Erläuterung	Einschränkung
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)	Dezentrale Peripherie
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Übersicht über die Datentypkonvertierungen (Seite 578)

DP_TOPOL: Topologie für DP-Mastersystem ermitteln

Beschreibung

Mit der Anweisung stoßen Sie die Topologieermittlung für ein ausgewähltes DP-Mastersystem an. Mit dem Aufruf von der Anweisung werden alle Diagnose-Repeater an einem DP-Mastersystem angesprochen.

Hinweis

Die Topologieermittlung kann zu einem Zeitpunkt nur für ein DP-Mastersystem durchgeführt werden.

Die Topologieermittlung ist die Voraussetzung für die detaillierte Anzeige des Fehlerorts bei auftretenden Leitungsfehlern. Führen Sie nach dem Aufbau und nach jeder Änderung des physikalischen Aufbaus eines DP-Mastersystems die Topologieermittlung mit der Anweisung "DP_TOPOL" erneut durch.

Änderungen des physikalischen Aufbaus sind:

- Änderung von Leitungslängen
- Hinzufügen oder Entfernen von Teilnehmern oder Komponenten mit Repeaterfunktion
- Ändern von Teilnehmeradressen

Wenn von einem Diagnose-Repeater ein Fehler gemeldet wird, beschreibt "DP_TOPOL" die Ausgänge DPR und DPRI für die Dauer eines "DP_TOPOL"-Durchlaufs. Wenn von mehreren Diagnose-Repeatern des ausgewählten DP-Mastersystems Fehler gemeldet werden, schreibt "DP_TOPOL" in DPR und DPRI Informationen zum ersten fehlermeldenden Diagnose-Repeater. Die vollständige Diagnoseinformation können Sie mit der Anweisung "DPNRM_DG (Seite 2450)" oder am PG auslesen. Falls kein Diagnose-Repeater einen Fehler meldet, haben die Ausgänge DPR und DPRI den Wert "0".

Wenn Sie nach dem Auftreten eines Fehlers erneut eine Topologieermittlung anstoßen wollen, müssen Sie "DP_TOPOL" zunächst zurücksetzen. Dies geschieht, indem Sie "DP_TOPOL" mit REQ=0 und R=1 aufrufen.

Arbeitsweise

"DP_TOPOL" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Ermittlung der Bustopologie, indem Sie "DP_TOPOL" mit REQ=1 aufrufen. Falls Sie den Vorgang abbrechen möchten, rufen Sie "DP_TOPOL" mit R=1 auf.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Hinweis

Die Topologieermittlung kann mehrere Minuten dauern.

Identifikation eines Auftrags

Der Eingangsparameter DP_ID legt einen Auftrag eindeutig fest.

Falls Sie "DP_TOPOL" erneut aufrufen, bevor die Topologieermittlung abgeschlossen ist, hängt das weitere Verhalten der Anweisung davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt: Stimmt der Parameter DP_ID mit einem noch nicht abgeschlossenen Auftrag überein, so wird der Aufruf als Folgeaufruf interpretiert. In RET_VAL wird der Wert W#16#7002 eingetragen. Handelt es sich hingegen um einen weiteren Auftrag, weist die CPU diesen ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DP_TOPOL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ=1: Topologieermittlung anstoßen
R	Input	BOOL	E, A, M, D, L oder Konstante	R=1: Abbruch der Topologieermittlung
DP_ID	Input	HW_IOSYS-TEM	E, A, M, D, L oder Konstante	DP-Mastersystem-ID desjenigen DP-Mastersystems, dessen Topologie ermittelt werden soll
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Topologieermittlung ist noch nicht abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DPR	Output	BYTE	E, A, M, D, L	PROFIBUS-Adresse des Fehler meldenden Diagnose-Repeaters
DPRI	Output	BYTE	E, A, M, D, L	Messsegment des Fehler meldenden Diagnose-Repeaters: • Bit 0 = 1: Temporäre Störungen auf Segment DP2 • Bit 1 = 1: Dauerhafte Störungen auf Segment DP2 • Bit 4 = 1: Temporäre Störungen auf Segment DP3 • Bit 5 = 1: Dauerhafte Störungen auf Segment DP3

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) ist zwischen zwei Fällen zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A4, 80C3, 80C5):
  Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt. Es ist sinnvoll, dass Sie "DP_TOPOL" erneut aufrufen (ggf. mehrfach).
  Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#8082, 80B0, 80B2):
  Bei dieser Fehlerart müssen Sie den Fehler beheben. Wenn Sie den Fehler beseitigt haben, ist ein erneuter Aufruf von "DP_TOPOL". Beispiel für einen permanenten Fehler: DP-Master / CPU unterstützt diesen Dienst nicht. (W#16#80B0).

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Es wird keine Topologieerrmittlung angestoßen. BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1. Der Auftrag, die Topologieermittlung durchzuführen, wurde gestellt. BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Die Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".
7010	Sie haben versucht, die Topologieermittlung abzubrechen. Es gibt aber keinen laufenden Auftrag mit der angegebenen DP_ID. BUSY hat den Wert "0".
7011	Erstaufruf mit R=1. Der Abbruch der Topologieermittlung wurde angestoßen. BUSY hat den Wert "1".
7012	Zwischenaufruf: Der Abbruch der Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".
7013	Letztaufruf: Die Topologieermittlung wurde abgebrochen. BUSY hat den Wert "0".
8082	Kein DP-Mastersystem ist mit der angegebenen DP_ID projektiert.

Fehlercode* (W#16#...)	Erläuterung
80A2	Fehler bei Topologieermittlung; genauere Informationen entnehmen Sie den Ausgangsparametern DPR und DPRI.
80A3	Fehler bei Topologieermittlung: Überwachungszeit ist abgelaufen (Timeout).
80A4	Kommunikation am PROFIBUS gestört.
80B0	DP-Master / CPU unterstützt diesen Dienst nicht.
80B2	Fehler bei Topologieermittlung: Am ausgewählten DP-Mastersystem wurde kein Diagnose-Repeater erkannt.
80C3	Benötigte Betriebsmittel sind momentan belegt. Mögliche Ursache: Sie haben eine zweite Topologieermittlung angestoßen (nur eine Topologieermittlung ist zu einem Zeitpunkt zulässig).
80C5	DP-Mastersystem ist momentan nicht verfügbar.
Allgemeine Fehlerinformationen	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ASi

ASI_CTRL: AS-i-Masterverhalten steuern

Beschreibung ASI_CTRL

Beschreibung

Mit der Anweisung "ASI_CTRL" wird das Verhalten des AS-i-Masters über das Anwenderprogramm der PLC gesteuert. Die Anweisung bearbeitet das Kommandoprotokoll selbstständig und ermöglicht auch die Parametrierungen an SIMATIC AS-i-Mastern sowie das Auslesen von Informationsdaten. Die Möglichkeiten und die Arbeitsweise der Kommandoschnittstelle sind im Handbuch des AS-i-Masters beschrieben.

Unterstützt werden sowohl zentral steckende AS-i-Master als auch dezentrale AS-i-Master über PROFIBUS DP. Auch Kombinationen mit PROFINET IO (Beispiel: IE/PB Link PN IO) sind möglich.

Die folgende Abbildung zeigt schematisch die Funktionalität der Anweisung "ASI_CTRL":

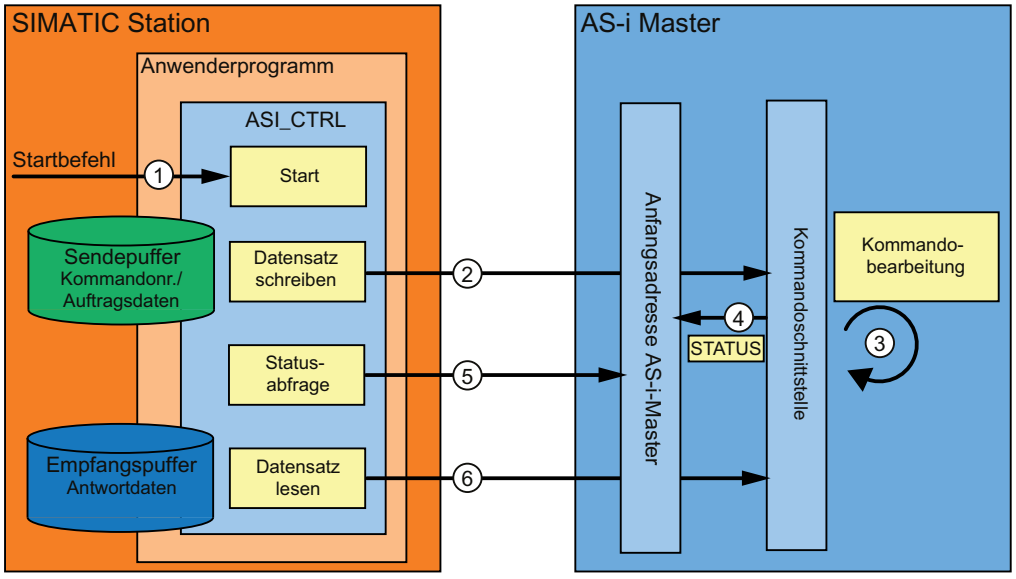

① Start der Bearbeitung am Parameter REQ.

② Das Programm setzt das gewünschte Kommando über die Anweisung "RDREC" an den AS-i-Master ab.

③ Der AS-i-Master bearbeitet das Kommando.

④ Der aktuelle Status des AS-i-Master wird im Eingangsbereich der Binärdaten (logische Basisadresse) abgelegt.

⑤ Die Anweisung "ASI_CTRL" fragt zyklisch die 4 Statusbits ab und wertet diese aus.

⑥ Wenn die Kommandobearbeitung beendet ist, wird der Kommandoauftrag über "RDREC" abgeschlossen. Im Datenfeld von "RDREC" befinden sich je nach Kommando die Antwortdaten des Kommandos oder weitere Statusinformationen.

Abhängigkeiten zwischen Anweisungs-Versionen

Die Version V1.3 der Anweisung "ASI_CTRL" erfordert die Version V1.1 der Anweisung "WRREC".

Die Version V1.2 der Anweisung "ASI_CTRL" erfordert die Version V1.1 der Anweisung "WRREC".

Unterschiede im Kommandoaufruf bei IE/ AS-i Link und DP/AS-i Links

Für den Kommandoaustausch einer Steuerung mit einem AS-i Master gibt es wesentliche Unterschiede in der Vorgehensweise.

- Beim **IE/AS-i Link** (PROFINET) wird die Datensatzschnittstelle verwendet. Die verschiedenen Kommandos werden entweder durch "Datensatz schreiben" (Anweisung "WRREC") oder durch "Datensatz lesen" (Anweisung "RDREC") von verschiedenen Datensatznummern aufgerufen.
- **DP/AS-i Links** (PROFIBUS) verwenden die Kommandoschnittstelle. Alle Kommandos werden durch ein paarweises "Datensatz schreiben" (Anweisung "WRREC") und "Datensatz lesen" (Anweisung "RDREC") von Datensatznummer 2 aufgerufen. Der Typ des Kommandos wird durch den Dateninhalt beim Schreibauftrag definiert.

Arbeitsweise der Anweisung "ASI_CTRL"

Die Anweisung "ASI_CTRL" ist ein asynchron arbeitender Funktionsbaustein. Die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Gestartet wird ein Auftrag mit REQ = TRUE.
- Über die Ausgangsparameter BUSY und die mittleren zwei Byte des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt.
- Während einer Auftragsbearbeitung ist der Parameter BUSY gesetzt. Bei Erstaufruf enthält STATUS dabei den Wert 00700100_H. Bei allen Folgeaufrufen, welche zu diesem Auftrag gehören, den Wert 00700200_H. Mit Abschluss des Auftrages wird das Ergebnis an den Parametern DONE bzw. ERROR ausgegeben.
  - Wenn kein Fehler aufgetreten ist, wird DONE gesetzt. Bei Aufträgen mit Antwortdaten vom AS-i-Master werden diese im angegebenen Empfangspuffer zur Verfügung gestellt. In diesem Fall wird im Parameter STATUS zusätzlich die Menge der gelieferten Daten in Bytes angezeigt. Bei Aufträgen ohne Antwortdaten wird in STATUS der Wert 00000000_H eingetragen.
  - Wenn während der Auftragsbearbeitung ein Fehler auftritt, wird ERROR gesetzt. In diesem Fall ist der Inhalt des Empfangspuffers ungültig. Zur näheren Beschreibung des aufgetretenen Fehlers wird im Parameter STATUS ein Fehlercode eingetragen.

Anzahl Kommandoaufrufe

Wenn Sie die Anweisung "ASI_CTRL" zur Kommandobeauftragung nutzen, dürfen Sie nicht gleichzeitig weitere Kommandos über "RDREC (Seite 2306)" und "WRREC (Seite 2326)" an denselben AS-i-Master absetzen. Dies gilt auch für Mehrfachaufrufe der Anweisung an denselben AS-i-Master.

Die Anweisung "ASI_CTRL" ist nicht unterbrechbar ablauffähig. Aufrufe dürfen daher nicht in Programmablaufebenen programmiert werden, die sich gegenseitig unterbrechen (z. B. durch Aufruf in OB 1 und in OB 35).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ASI_CTRL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, Konstante	Mit REQ = TRUE wird ein neuer Auftrag gestartet, falls nicht bereits ein Auftrag in Bearbeitung ist. Es findet keine Flankenauswertung statt.
LADDR	Input	HW_IO	E, A, M, D, L, Konstante	Hardwarekennung des AS-i Masters. Die Adresse können Sie aus den Modul-Eigenschaften entnehmen.
SD	Input	VARIANT	E, A, M, D, L	Sendepuffer Der Parameter verweist auf einen Speicherbereich, in dem das Kommando zu spezifizieren ist (siehe "ASi-Kommandos (Seite 2461)").
RD	Input	VARIANT	E, A, M, D, L	Empfangspuffer Dieser Puffer ist nur bei Kommandos relevant, die Antwortdaten liefern. Der Parameter verweist auf einen Speicherbereich, in dem eine Kommandoantwort abgelegt wird (siehe "ASi-Kommandos (Seite 2461)").
DONE	Output	BOOL	E, A, M, D, L	DONE = TRUE: Auftrag ohne Fehler beendet.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = TRUE: Auftrag in Bearbeitung.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = TRUE: Auftrag mit Fehler abgebrochen.
STATUS	Output	DWORD	E, A, M, D, L	Auftragsstatus / Fehlercode Siehe Beschreibung "Parameter STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter LADDR, SD und RD

Die Parameter LADDR, SD und RD dürfen während einer Auftragsbearbeitung in keinem Bausteindurchlauf verändert werden, sondern müssen konstant bleiben.

Parameter STATUS

Die folgende Tabelle enthält die möglichen Anzeigen von STATUS in Abhängigkeit von DONE und ERROR.

DONE	BUSY	ERROR	STATUS	Bedeutung
0	0	0	00700000_H	Erstaufruf mit REQ = FALSE; keine Auftrag aktiv.
0	1	0	00700100_H	Erstaufruf mit REQ = TRUE; Auftrag angestoßen.
0	1	0	00700200_H	Folgeaufruf (REQ irrelevant); Auftrag wird derzeit noch bearbeitet.
1	0	0	00000000_H	Auftrag ohne Fehler abgeschlossen. Keine Antwortdaten.
1	0	0	$0000xx00_H$	Auftrag ohne Fehler abgeschlossen. Anzahl xx-Bytes Antwortdaten.

DONE	BUSY	ERROR	STATUS	Bedeutung
0	0	1	C0818400 H	Datentyp des Formaloperanden RD unzulässig.
0	0	1	C0818500 H	Kommunikationsfehler mit AS-i-Master (falsche Adresse am Parameter LADDR projektiert).
0	0	1	C0838100 H	Falsche AS-i-Slave-Adresse.
0	0	1	C0838200 H	Der AS-i-Slave ist nicht aktiviert (nicht in LAS).
0	0	1	C0838300 H	Fehler am AS−Interface (der Parameter SD ist eventuell zu klein eingestellt).
0	0	1	C0838400 H	Das Kommando ist im aktuellen Zustand des AS-i-Masters nicht zulässig.
0	0	1	C0838500 H	Ein AS-i-Slave mit Adresse "0" existiert.
0	0	1	C0838600 H	Der AS-i-Slave hat unzulässige Konfigurationsdaten (E/A bzw. ID−Codes).
0	0	1	C083A100 H	Der angesprochene AS-i-Slave wurde am AS−Interface nicht gefunden.
0	0	1	C083A200 H	Ein AS-i-Slave mit Adresse "0" existiert.
0	0	1	C083A300 H	Ein AS-i-Slave mit der neuen Adresse ist bereits am AS−Interface vorhanden.
0	0	1	C083A400 H	Die AS-i-Slave-Adresse lässt sich nicht löschen.
0	0	1	C083A500 H	Die AS-i-Slave-Adresse lässt sich nicht setzen.
0	0	1	C083A600 H	Die AS-i-Slave-Adresse lässt sich nicht permanent speichern.
0	0	1	C083A700 H	Fehler beim Lesen des Extended ID1−Codes.
0	0	1	C083A800 H	Die Zieladresse ist nicht plausibel (z. B. wird eine B-Slave-Adresse für einen Standard−Slave verwendet).
0	0	1	C083B100 H	Ein Längenfehler ist beim Stringtransfer aufgetreten.
0	0	1	C083B200 H	Ein Protokollfehler ist beim Stringtransfer aufgetreten.
0	0	1	C083F800 H	Die Auftragsnummer oder der Auftragsparameter ist unbekannt.
0	0	1	C083F900 H	Der AS-i-Master hat einen EEPROM-Fehler festgestellt.

Siehe auch

RDREC: Datensatz lesen (Seite 2306)

WRREC: Datensatz schreiben (Seite 2326)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

ASi-Kommandos

Beschreibung

Durch die Kommandoschnittstelle ist die Möglichkeit gegeben, Parametrier- und Informationsdaten zwischen Steuerung und AS-i-Master auszutauschen.

Mit diesen Kommandos

- wird die komplette Funktionalität des Masterprofiles M4 der AS-i-Masterspezifikation zur Verfügung gestellt.
- kann der AS-i-Master komplett von der Steuerung aus projektiert werden.

Hinweis

Unterstützte AS-i-Kommandos

Welche AS-i Kommandos unterstützt werden sowie eine detaillierte Beschreibung finden Sie im Handbuch des jeweiligen AS-i-Masters.

Allgemeine Struktur des Sendepuffers

Die allgemeine Struktur des Sendepuffers für Kommandos und Auftragsdaten zeigt die folgende Tabelle. Der Bereich für die Kommandonummer muss immer belegt sein. Die Anzahl Byte für die Auftragsdaten sind in der Beschreibung des jeweiligen Kommandos enthalten (siehe Dokumentation des AS-i-Masters). "q" entspricht dabei der Anfangsadresse des Sendepuffers.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Bedeutung							
q + 0	Kommandonummer							
q + 1	Auftragsdaten							
q + 2	Auftragsdaten							
q + ...	Auftragsdaten							

Allgemeine Struktur des Empfangspuffers

Die allgemeine Struktur des Empfangspuffers für die Antwortdaten des Kommandos zeigt folgende Tabelle. Die Anzahl Bytes für die Antwortdaten ist vom jeweiligen Kommando abhängig. Einige Kommandos liefern keine Antwortdaten und benötigen somit lediglich die Angabe eines virtuellen Empfangspuffers, der nicht mit Daten befüllt wird. "n" entspricht dabei der Anfangsadresse des Empfangspuffers.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Bedeutung							
n + 0	Antwortdaten							
n + 1	Antwortdaten							
n + 2	Antwortdaten							
n + ...	Antwortdaten							

ACHTUNG

Speicherbereiche können überschrieben werden

Wenn der Empfangspuffer der Anweisung "ASI_CTRL" zu kurz ist, werden eventuell angrenzende Speicherbereiche überschrieben. Die Längenangabe im ANY-Zeiger des Parameters RD beim Aufruf der Anweisung "ASI_CTRL" ist irrelevant. Die benötigte Länge des Empfangspuffers ist der Beschreibung des Kommandos zu entnehmen.

Für die Kommandonummern 39_H, 41_H, 42_H, 43_H und 44_H gilt:

Der Empfangspuffer muss eine Länge von 221 Byte haben (Byte 0 bis 220), auch wenn das Kommando weniger Daten zurückliefert. Je nach Kommando werden die im Empfangspuffer enthaltenen höchsten Bytes vom AS-i-Master eventuell mit Nullwerten überschrieben.

AS-i-Kommandos

Die folgende Tabelle zeigt eine Auswahl möglicher AS-i-Kommandos.

Name	Parameter	Rückgabe	Codierung
Parameterwert_projektieren (Set_Permanent_Parameter)	Slave–Adresse, Parameter		00_H
Projektierten_Parameterwert_lesen (Get_Permanent_Parameter)	Slave–Adresse	Parameter	01_H
Parameterwert_schreiben (Write_Parameter)	Slave–Adresse, Parameter	Parameterecho	02_H
Parameterwert_lesen (Read_Parameter)	Slave–Adresse	Parameterwert	03_H
Ist_Parameterwerte_projektieren (Store_Actual_Parameters)			04 H
Konfigurationsdaten_projektieren	Slave–Adresse, Konfiguration		25 H

Name	Parameter	Rückgabe	Codierung
Projektierte_Konfigurationsdaten_lesen	Slave–Adresse	Projektierte Konfigurationsdaten	26 H
Ist_Konfigurationsdaten_projektieren (Store_Actual_Configuration)			07 $_H$
Ist-Konfigurationsdaten_lesen	Slave-Adresse	Ist-Konfigurationsdaten	28 $_H$
LPS_projektieren	LPS		29 $_H$
Offlinemodus_setzen	Mode		0A $_H$
Autoprogrammieren_wählen	Mode		0B $_H$
Betriebsmodus_setzen	Mode		0C $_H$
AS–iSlave–Adresse_ändern (Change_AS–iSlave_Address)	Adresse1, Adresse2		0D $_H$
AS–iSlavestatus_lesen	Slave–Adresse	Fehlerrecord des AS-iSlaves	0F $_H$
Listen_und_Flags_lesen		LDS, LAS, LPS, Flags	30 $_H$
Gesamtkonfiguration_lesen		Ist-Konfigurationsdaten, aktuelle Parameter, LAS, Flags	39 $_H$
Gesamtkonfiguration_projektieren	Gesamtkonfiguration		3A $_H$
Parameterliste_schreiben	Parameter Liste		3C $_H$
Parameterecholiste_lesen		Parameterecho-Liste	33 $_H$
CTT2-Request_schreiben	Slave Adresse CTT2-String	CTT2-String	44 $_H$
Versionskennung_lesen		Versions-String	14 $_H$
AS–iSlave-ID_lesen	Slave–Adresse	ID-Code	17 $_H$
AS–iSlave-Extended–ID1_lesen	Slave–Adresse	Extended ID1–Code	37 $_H$
AS–iSlave-Extended–ID1_schreiben	Extended ID1–Code		3F $_H$
AS–iSlave-Extended–ID2_lesen	Slave–Adresse	Extended ID2–Code	38 $_H$
AS–iSlave-EA_lesen	Slave–Adresse	E/A-Konfiguration	18 $_H$
Peripheriefehlerliste_lesen		LPF	3E $_H$
AS–i–Slave_Parameter–String_schreiben	Slave–Adresse, Parameter–String		40 $_H$
AS–iSlave_Parameter–String_lesen	Slave–Adresse	Parameter–String	41 $_H$
AS–iSlave_ID–String_lesen	Slave–Adresse	ID–String	42 $_H$
AS–iSlave_Diagnose–String_lesen	Slave–Adresse	Diagnose–String	43 $_H$
AS–iStrang_Fehlerzähler_lesen			4A $_H$
AS–iStrang_Fehlerzähler_lesen_und_löschen			4B $_H$
AS–iSlave_Fehlerzähler_lesen	Slave–Adresse		4C $_H$
AS–iSlave_Fehlerzähler_lesen_und_löschen	Slave–Adresse		4D $_H$
Zusätzliches Kommando für den DP/ AS-i F-Link:			
AS–i-Status/Diag_der_F-Slaves		Status / Diagnose aller ASIsafe-Salves	51 $_H$

Hinweis

Erneute Initialisierung der Kommandoschnittstelle des AS-i-Masters

Ein weiteres Kommando, welches in der Tabelle nicht erwähnt wird, ist das Kommando 77 $_H$. Mit diesem Aufruf wird die Kommandoschnittstelle des AS-i-Masters neu initialisiert. Falls der angegebene AS-i-Master zu diesem Zeitpunkt mit einem Kommando beschäftigt sein sollte, wird dieses abgebrochen.

Ab Version V2.1.20 des DP/AS-i LINK Advanced wird zusätzlich das Kommando 0E $_H$ angeboten. Durch diesen Aufruf kann die Funktion Erdschlussüberwachung eines Strangs frei gegeben oder gesperrt werden.

4.1.4.5 PROFIenergy

Beschreibung PROFIenergy

PROFIenergy

PROFIenergy ist ein hersteller- und geräteunabhängiges Profil für das Energiemanagement mit PROFINET. Um während Produktionspausen oder ungeplanten Unterbrechungen den Stromverbrauch zu reduzieren, lassen sich Geräte mit PROFIenergy koordiniert und zentral abschalten.

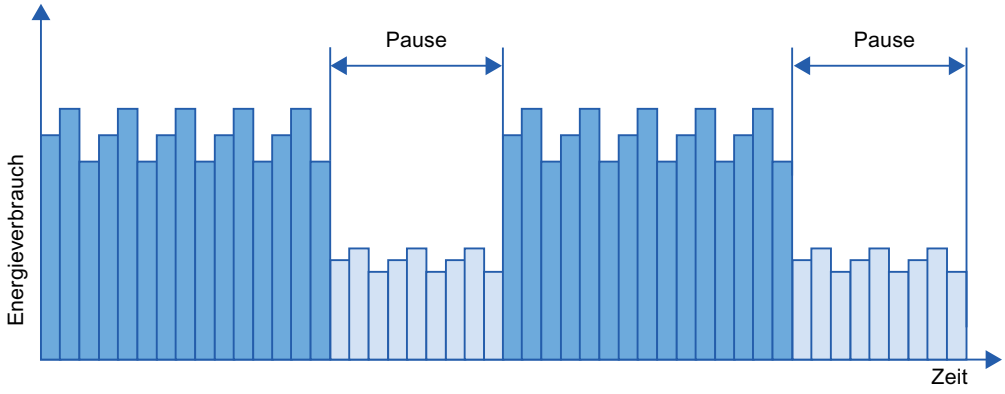

Die Abschaltung der PROFINET-Geräte bzw. der Powermodule erfolgt über spezielle Kommandos im Anwenderprogramm des PROFINET IO-Controllers. Es wird keine zusätzliche Hardware benötigt. Die PROFIenergy-Kommandos werden direkt von den PROFINET-Geräten interpretiert.

PROFIenergy-Controller (PE-Controller)

Der PE-Controller ist eine PLC, die bei unterlagerten Geräten den Ruhezustand aktiviert bzw. deaktiviert. Die Abschaltung bzw. Reaktivierung von einzelnen Produktionskomponenten oder ganzen Produktionslinien erfolgt dabei über das Anwenderprogramm. Über entsprechende Anweisungen (Funktionsbausteine) werden Kommandos (z. B. "Start_Pause" oder "End_Pause") an das unterlagerte Gerät abgesetzt. Das Senden der Kommandos erfolgt über das PROFINET-Kommunikationsprotokoll.

PROFIenergy-Entity (PE-Entity)

Die PE-Entity empfängt die PROFIenergy-Kommandos des PE-Controllers und setzt die Kommandos entsprechend um (z.B. durch Rücksenden eines Messwerts oder Aktivierung eines Energiesparmodus). Die Umsetzung der PE-Entity in einem PROFIenergy-fähigem Gerät ist geräte- und herstellerspezifisch.

Die PE-Entity kann z. B. umgesetzt werden:

- Innerhalb des Proxys eines Moduls: Die PE-Kommandos sind jeweils gültig für das adressierte Modul und ggf. unterlagerte Module.
- Innerhalb des Proxys einer Baugruppe: Die PE-Kommandos sind jeweils gültig für verschiedene Module innerhalb der Baugruppe.

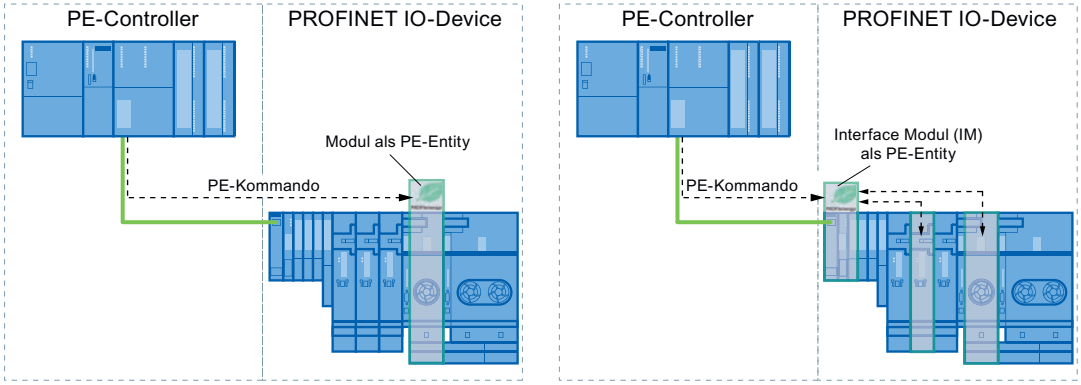

- Innerhalb eines Moduls ohne Proxy-Funktion innerhalb des Netzwerks: Die PE-Kommandos sind in diesem Fall nur für das jeweilige Modul gültig.

PROFIenergy-Anweisungen

- Anweisungen für IO-Controller
  - Mit der Anweisung "PE_START_END (Seite 2467)" wird auf einfachste Art und Weise der Ruhezustand der PROFINET-Geräte aktiviert oder deaktiviert (PROFIenergy-Kommandos "Start_Pause" und "End_Pause"). Das passiert über eine positive und negative Signalflanke in der Anweisung.
  - Mit der Anweisung "PE_CMD (Seite 2472)" lassen sich alle PROFIenergy-Kommandos einschließlich "Start_Pause" und "End_Pause" übertragen. Darüber hinaus kann mit weiteren Kommandos z. B. der aktuelle Status der PROFINET-Geräte oder das Verhalten während der Pausen abgefragt werden.
  - Mit der Anweisung "PE_DS3_Write_ET200S (Seite 2477)" werden die Einstellungen für das Schaltverhalten von bis zu 8 Steckplätzen der ET 200S festgelegt. Die Anweisung gehört nicht zu den PROFIenergy-Anweisungen, ergänzt aber die PROFIenergy-Funktionen für eine ET 200S.
- Anweisung für iDevices
  Mit der Anweisung "PE_I_DEV (Seite 2503)" lässt sich PROFIenergy auch auf iDevices realisieren. Die Anweisung empfängt PROFIenergy-Kommandos auf dem iDevice und gibt sie an das Anwenderprogramm zur Bearbeitung weiter. Nach der Bearbeitung des Kommandos durch das Anwenderprogramm ruft es die Anweisung "PE_I_DEV (Seite 2503)" erneut auf, um die Quittung an den IO-Controller zu senden. Für diese Antworten steht Ihnen für jedes Kommando ein entsprechender Hilfsbaustein zur Verfügung, der die Anweisung "PE_I_DEV (Seite 2503)" mit den Antwortdaten versorgt.

PROFIenergy-Kommandos (PE-Kommandos)

PE-Kommandos werden von dem PE-Controller an die PE-Entity gesendet. Das PE-Kommando kann entweder ein Steuerungskommando sein, um eine PE-Entity in einen Energiesparmodus zu schalten oder ein Kommando zum Auslesen eines Status oder Messwertes:

- PE-Kommandos zur Steuerung
  PROFIenergy unterstützt zwei Steuerungskommandos, die entweder über die Anweisung "PE_Start_End (Seite 2467)" oder die Anweisung "PE_CMD (Seite 2472)" ausgeführt werden können:
  - Start_Pause: Starten eines geeigneten Energiesparmodus (PE Energy-saving mode)
  - End_Pause: Beenden des Energiesparmodus (Wechsel in den Modus PE_ready_to_operate)
- PE-Kommandos zum Auslesen eines Status oder Messwerts
  Über die folgenden Statuskommandos können mit der Anweisung "PE_CMD (Seite 2472)" bestimmte Zustandsinformationen durch die Steuerung ausgelesen werden:
  - PE_Identify: Auslesen, welche PE-Kommandos von der PE-Entity unterstützt werden.
  - PEM_Status: Auslesen des momentan aktiven Modus einer PE-Entity (z. B. PE_ready_to_operate).
  - Query_Modes: Ausgabe einer Übersicht zu allen unterstützten Energiesparmodi, inklusive der Zeit- und Energieinformationen
  - Query_Measurement: Ausgabe der Messwerte einer PE-Entity

Applikationsbeispiele

Beispiele zur Anwendung der PROFIenergy-Anweisungen stehen über den Industry Online Support unter dem Beitrag "PROFIenergy - Energie sparen mit SIMATIC S7 (http://support.automation.siemens.com/WW/view/de/41986454)" zur Verfügung.

Siehe auch

Service und Support (http://support.automation.siemens.com/)

IO-Controller

PE_START_END: Energiesparmodus starten und beenden

Beschreibung

Mit der Anweisung "PE_START_END" wird für die angegebene PE-Entity (z. B. die ET 200S) die Energiesparpause gestartet und beendet.

Die Anweisung "PE_START_END" wird im PE-Controller verwendet, vorzugsweise wenn an den zugeordneten PE-Devices nur Feldgeräte angeschlossen sind, bei denen keine Energiedaten ausgelesen werden müssen. Zum Lesen der Energiedaten kann alternativ die Anweisung "PE_CMD (Seite 2472)" verwendet werden.

Die Energiesparmodi werden im Anwenderprogramm des PE-Controllers projektiert. Der tatsächlich eingenommene Energiesparmodus wird nach Ausführung von "PE_START_END" von der PE-Entity zurückgemeldet und am Parameter PE_MODE_ID ausgegeben.

Schreib- und Leseaufträge der Anweisung "PE_START_END"

Die Anweisung "PE_START_END" versendet intern mittels "WRREC (Seite 2326)" einen PROFIenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE_START_END" auf die Quittung von der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 2306)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 2306)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:

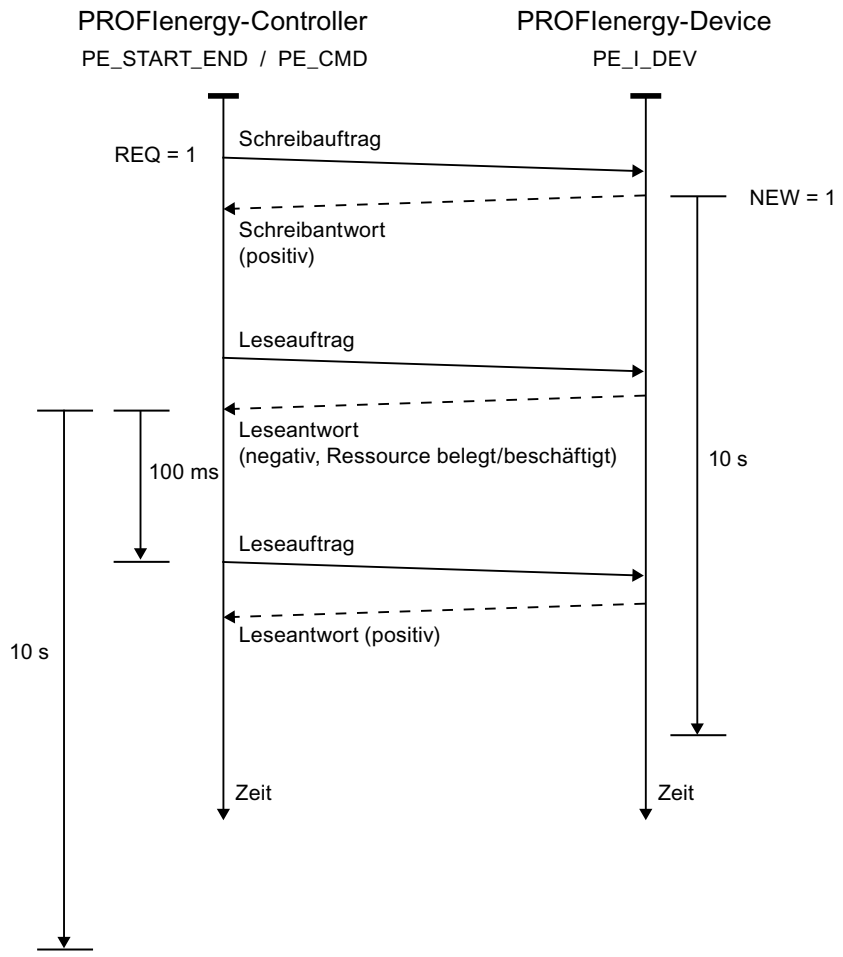

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_START_END":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
START	Input	BOOL	E, A, M, D, L oder Konstante	Senden des PE-Kommandos "Start_Pause" an die PE-Entity mit der Adresse am Parameter ID.
END	Input	BOOL	E, A, M, D, L oder Konstante	Senden des PE-Kommandos "End_Pause" an die PE-Entity mit der Adresse am Parameter ID.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	HW_ SUBMODULE	E, A, M, D, L oder Konstante	Adresse der PE-Entity Verwenden Sie bei einem PROFINET-IO Device die Hardware-Kennung der Kopfbaugruppe. Die Hardware-Kennung entnehmen Sie den Systemkonstanten des zugewiesenen IO-Controllers. Der Name der Kopfbaugruppe ist aus dem Namen des IO-Device und dem Zusatz [Head] zusammengesetzt (Beispiel: "IO_Device_1[Head]"). Handelt es sich bei der PE-Entity um ein I-Device, geben Sie stattdessen die Hardware-Kennung eines Transferbereichs an.
PAUSE_TIME	Input	TIME	E, A, M, D, L, P oder Konstante	Geplante Pausendauer. • Bereich: T#1MS bis T#24D20H31M23S647MS • Startwert: T#0MS
VALID	Output	BOOL	E, A, M, D, L	PE-Kommando erfolgreich abgesetzt.
BUSY	Output	BOOL	E, A, M, D, L	Bearbeitung des PE-Kommandos läuft noch.
ERROR	Output	BOOL	E, A, M, D, L	Bei der Bearbeitung trat ein Fehler auf. Die Fehlermeldung wird am Parameter STATUS ausgegeben.
STATUS	Output	DWORD	E, A, M, D, L, P	Bausteinstatus / Fehlernummer (siehe "Parameter STATUS").
PE_MODE_ID	Output	BYTE	E, A, M, D, L, P	Identifikationsnummer des Energiesparmodus (Energiesparlevel, der während der Pausendauer eingenommen wird).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter PAUSE_TIME

Mit dem Parameter PAUSE_TIME wird der PE-Entity die Dauer der Energiesparpause vorgegeben. In der PE-Entity wird geprüft, ob die vorgegebene Pausendauer ausreichend groß ist und umgesetzt werden kann. Die Mindest-Pausendauer (Time_min_Pause) muss dabei größer sein, als die Summe der Zeiten, die das Gerät zum Wechsel in den Energiesparmodus (Time_to_Pause) und zum Wechsel zurück in den Betriebsmodus (Time_to_Operate) benötigt.

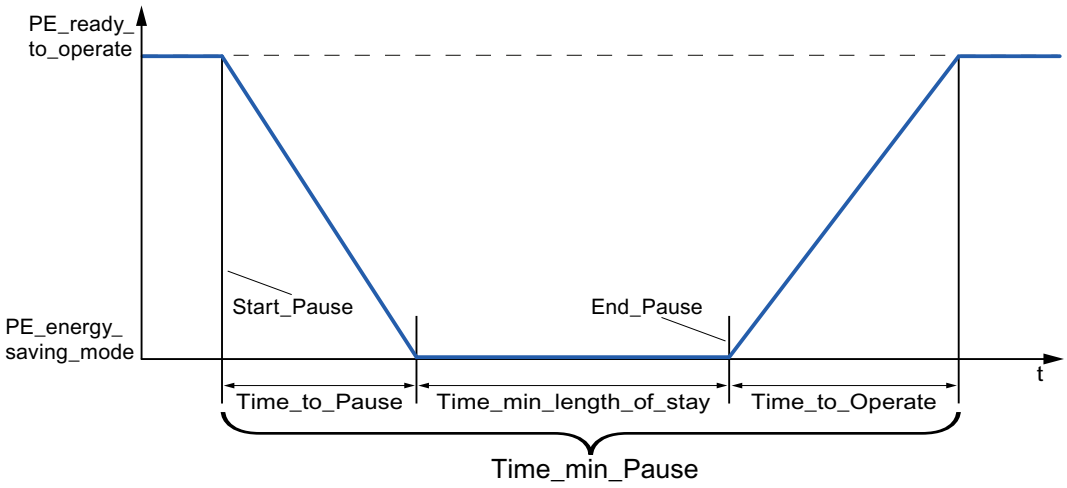

Bei der ET 200S erfolgt eine Prüfung, ob die geplante Pausendauer größer oder gleich der minimalen Pausendauer (PM-E_Pause_Min) ist, die auf der ET 200S hinterlegt ist. Diese beträgt fest 10 Sekunden. Wenn eine kleinere Pause verwendet wird, bleiben die Powermodule (PM-E) der ET 200S eingeschaltet.

Es erfolgt kein automatisches Wiedereinschalten nach Ablauf der Pausenzeit, die Baugruppe bleibt bis zu dem "END"-Befehl im Zustand AUS. Damit wird ein unkoordiniertes Wiedereinschalten, was zu ungewollten Lastspitzen führen kann, verhindert.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wenn er als ARRAY[1...4] of BYTE interpretiert wird, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#00: Kein Fehler • B#16#DE: Fehler beim Lesen des Datensatzes • B#16#DF: Fehler beim Schreiben des Datensatzes • B#16#C0: Fehlermeldung durch die Anweisung oder durch die intern verwendeten Kommunikationsanweisungen "RDREC (Seite 2306)" und "WRREC (Seite 2326)".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • 80: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch • FE: DP/PNIO-Profile - PROFIenergy-spezifischer Fehler

Feldelement	Name	Bedeutung
STATUS[3]	Error_Code_1	Fehlerkennung • Bei Error_Decode = **80**: – **80**: Gleichzeitig eine steigende Flanke an den Eingangsparametern START und END. – **81**: Längenkonflikt bei den Parametern CMD_PARAM und CMD_PARAM_LEN. – **82-8F**: weitere Fehlermeldungen (reserviert) • Bei Error_Decode = **FE**: – **01**: "Service Request ID" ungültig – **02**: Falsche "Request_Reference" – **03**: "Modifier" ungültig – **04**: "Data Structure Identifier RQ" ungültig – **05**: "Data Structure Identifier RS" ungültig – **06**: "PE energy-saving modes" werden nicht unterstützt – **07**: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten) – **08**: "Count" ungültig – **50**: Es steht kein passender "energy mode" zur Verfügung. – **51**: Angegebener Zeitwert wird nicht unterstützt. – **52**: "PE_Mode_ID" unzulässig – **53**: Wechsel in „PE energy-saving mode” nicht möglich, da Gerät in Betrieb – **54**: Funktion zur Zeit nicht möglich. Gerät falsch parametriert oder fehlerhafter Setup. – **55** bis **FF**: Reserviert
STATUS[4]	Error_Code_2	Herstellerspezifische Erweiterung der Fehlerkennung

Hinweis

Fehlermeldungen der Anweisungen RDREC und WRREC

Die Anweisung "PE_START_END" verwendet zur Kommunikation die Anweisungen "WRREC (Seite 2326)" und "RDREC (Seite 2306)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 2326)" und "RDREC (Seite 2306)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2370).

Siehe auch

Beschreibung PROFIenergy (Seite 2464)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

PE_CMD: Energiesparmodus starten und beenden / Zustandsinformationen auslesen

Beschreibung

Die Anweisung "PE_CMD" wird im PE-Controller verwendet und löst in der PE-Entity eine Energiesparpause aus oder beendet eine Pause. Zusätzlich werden mit "PE_CMD" weitere Informationen und Energiemesswerte aus einer PE-Entity ausgelesen.

Die Anweisung kann vorzugsweise bei PE-Controllern eingesetzt werden, an deren zugeordneten PE-Devices Feldgeräte angeschlossen sind, von denen Energie-Messwerte ausgelesen werden sollen. Ist dies nicht der Fall, kann zum starten und beenden der Pausen auch die Anweisung "PE_START_END (Seite 2467)" verwendet werden.

Übertragung der PROFIenergy-Kommandos (PE-Kommandos)

Die Anweisung "PE_CMD" überträgt ein PROFIenergy-Kommando an eine PE-Entity.

Wenn das PROFIenergy-Profil zukünftig um weitere Kommandos erweitert wird, kann die Anweisung auch eingesetzt werden. Die Kommandos, die nach dem aktuellen PROFIenergy-Profil verwendet werden können, sind in der Beschreibung der Parameter CMD und CMD_MODIFIER aufgelistet (siehe Tabelle "Parameter CMD und CMD_MODIFIER").

- Den einzelnen PE-Kommandos, die mit der Anweisung an die PE-Entity übertragen werden, sind definierte "Service_Request_IDs" zugeordnet. Die Service_Request_IDs 01...05 und 16 werden am Parameter CMD zugewiesen.
- Über den Parameter CMD_MODIFIER werden die beiden PE-Kommandos 04 (Query_Modes) und 16 (Query_Measurement) näher spezifiziert.
- Für einzelne PE-Kommandos werden am Parameter CMD_PARA zusätzliche Werte übergeben (siehe Beschreibung der einzelnen PE-Kommandos). Der Parameter CMD_PARA_LEN legt die Länge der Daten am Parameter CMD_PARA fest.

Die Kommandos werden ohne Plausibilitätstest übertragen. Die Antwortdaten der PE-Entity werden in dem durch VARIANT-Zeiger adressierten Datenbereich RESPONSE_DATA abgelegt (die Inhalte der Antworttelegramme sind in den Beschreibungen des jeweiligen PE-Kommandos enthalten).

Schreib- und Leseaufträge der Anweisung "PE_CMD"

Die Anweisung "PE_CMD" versendet mittels "WRREC (Seite 2326)" einen PROFIenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE_CMD" auf die Quittung der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 2306)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 2306)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:

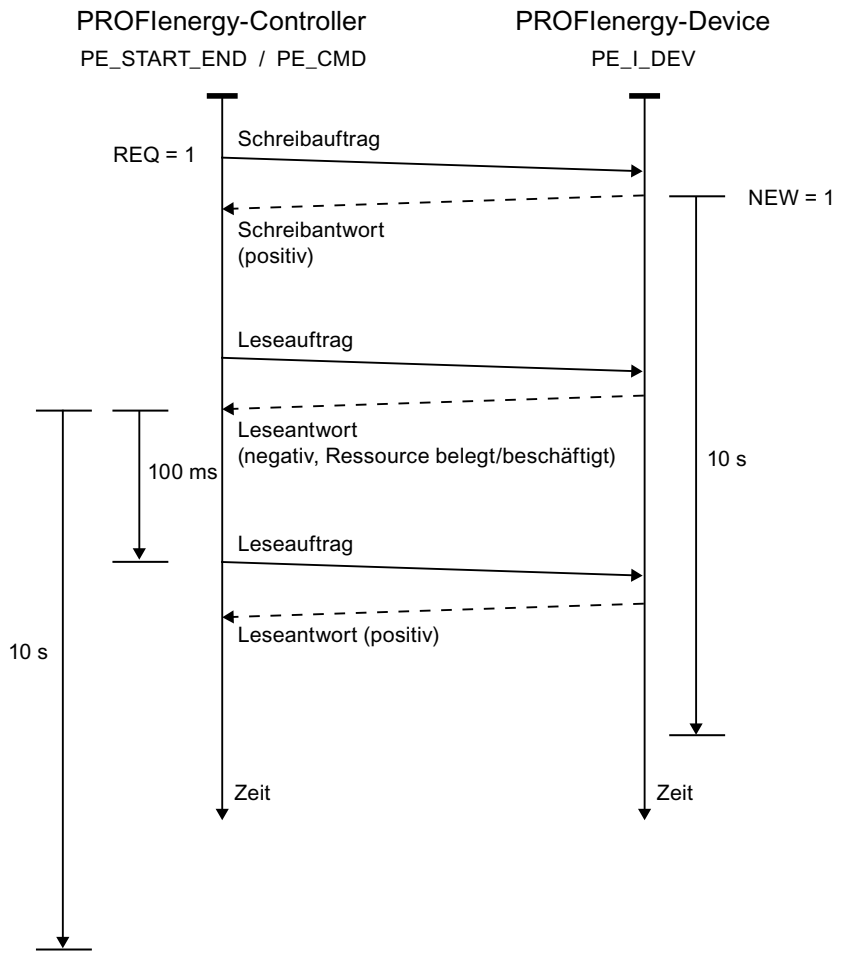

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_CMD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Startet die Übertragung des PE- Kommandos bei positiver Flanke.
ID	Input	HW_ SUBMODULE	E, A, M, D, L oder Konstante	Adresse der PE-Entity Verwenden Sie bei einem PROFINET-IO Device die Hardware-Kennung der Kopfbaugruppe. Die Hardware-Kennung entnehmen Sie den Systemkonstanten des zugewiesenen IO-Controllers. Der Name der Kopfbaugruppe ist aus dem Namen des IO-Device und dem Zusatz [Head] zusammengesetzt (Beispiel: "IO_Device_1[Head]"). Handelt es sich bei der PE-Entity um ein I-Device, geben Sie stattdessen die Hardware-Kennung eines Transferbereichs an.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BYTE	E, A, M, D, L, P oder Konstante	Service-Request-ID des PROFIenergy-Kommandos entsprechend dem PROFIenergy-Profil (siehe "Parameter CMD und CMD_MODIFIER"). Nach Erweiterungen des PROFIenergy-Profils sind weitere Service-Request-IDs möglich.
CMD_MODIFIER	Input	BYTE	E, A, M, D, L, P oder Konstante	PROFIenergy-Unterkommando (nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER") Nach Erweiterungen des PROFIenergy-Profils sind weitere Unterkommandos möglich.
CMD_PARA	Input	VARIANT	E, A, M, D, L	Parameter für die PE-Kommandos: • Get mode: PE_mode_ID • Get measurement values: List of Measurement_Ids Es wird der komplette Service Data Request eingetragen.
CMD_PARA_LEN	Input	INT	E, A, M, D, L, P oder Konstante	Tatsächliche Länge der Parameter zu dem Kommando (<= Länge in CMD_PARA, wird von der Anweisung geprüft).
CMD_DATA_STRUCTURE_IDENTIFIER_RQ	Input	BYTE	E, A, M, D, L, P oder Konstante	Wert des Data Structure Identifiers aus der Norm Hinweis: Den Parameter CMD_DATA_STRUCTURE_IDENTIFIER_RQ gibt es ab der Anweisungs-Version V2.0.
RESPONSE_DATA	InOut	VARIANT	E, A, M, D, L	PROFIenergy-Information Je nach Kommando gesamtes Antwort-Telegramm einschließlich Block-Header. Hinweis: Wenn der Puffer zu klein ist, werden nur die Anzahl Bytes eingetragen, wie im VARIANT-Zeiger angegeben.
VALID	Output	BOOL	E, A, M, D, L	Kommando erfolgreich abgesetzt.
BUSY	Output	BOOL	E, A, M, D, L	Kommandobearbeitung läuft noch.
ERROR	Output	BOOL	E, A, M, D, L	Bei der Bearbeitung trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus / Fehlernummer (siehe "Parameter STATUS"):

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter CMD und CMD_MODIFIER

CMD	CMD_ MODIFIER	PROFIenergy-Kommando	Beschreibung
01	0	Start_Pause (Seite 2493)	Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus.
02	0	End_Pause (Seite 2493)	Beenden des Energiesparmodus.
03	1	Query_Modes - List energy saving Modes (Seite 2494)	Ausgeben der unterstützten Energiesparmodi.
	2	Query_Modes - Get Mode (Seite 2495)	Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.
04	0	PEM_Status (Seite 2497)	Status des Energiesparmodus abfragen.
05	0	PE_Identify (Seite 2499)	Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.
16	1	Query_Measurement - Get_Measurement_List (Seite 2500)	Auflistung der durch die PE-Entity unterstützten Messwerte.
	2	Query_Measurement - Get_Measurement_Values (Seite 2502)	Ausgabe der Messwerte der PE-Entity.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wenn er als ARRAY[1...4] of BYTE interpretiert wird, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#00: Kein Fehler • B#16#DE: Fehler beim Lesen des Datensatzes • B#16#DF: Fehler beim Schreiben des Datensatzes • B#16#C0: Fehlermeldung der intern verwendeten Kommunikationsanweisungen "RDREC (Seite 2306)" und "WRREC (Seite 2326)".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • 80: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch • FE: DP/PNIO-Profile - PROFIenergy-spezifischer Fehler

Feldelement	Name	Bedeutung
STATUS[3]	Error_Code_1	Fehlerkennung • Bei Error_Decode = **80**: – **81**: Längenkonflikt bei den Parametern CMD_PARA und CMD_PARA_LEN bzw. maximale Datensatzlänge (4095 Byte) überschritten. – **82-8F**: weitere Fehlermeldungen (reserviert) • Bei Error_Decode = **FE**: – **01**: "Service Request ID" ungültig – **02**: Falsche "Request_Reference" – **03**: "Modifier" ungültig – **04**: "Data Structure Identifier RQ" ungültig – **05**: "Data Structure Identifier RS" ungültig – **06**: "PE energy-saving modes" werden nicht unterstützt – **07**: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten) – **08**: "Count" ungültig – **50**: Es steht kein passender Energiesparmodus (energy mode) zur Verfügung. – **51**: Angegebener Zeitwert wird nicht unterstützt. – **52**: "PE_Mode_ID" ungültig
STATUS[4]	Error_Code_2	Herstellerspezifische Erweiterung der Fehlerkennung

Hinweis

Fehlermeldungen der Anweisungen RDREC und WRREC

Die Anweisung "PE_CMD" verwendet zur Kommunikation die Anweisungen "WRREC (Seite 2326)" und "RDREC (Seite 2306)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 2326)" und "RDREC (Seite 2306)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2370).

Siehe auch

Beschreibung PROFIenergy (Seite 2464)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

PE_DS3_Write_ET200S: Schaltverhalten der Powermodule einstellen

Beschreibung

Die Anweisung "PE_DS3_Write_ET200S" sendet grundsätzliche Einstellungen zum Schaltverhalten der Powermodule an die ET 200S. Mit der Anweisung "PE_DS3_Write_ET200S" wird das Schaltverhalten für bis zu 8 Slots der ET 200S (z. B. für Powermodule) festgelegt.

Hinweis

Diese Anweisung ist nicht Bestandteil des PROFIenergy-Profils, sondern ergänzt SIMATIC-spezifische Funktionen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_DS3_Write_ET200S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ENABLE	Input	BOOL	E, A, M, D, L oder Konstante	Eine positive Flanke stößt die Übertragung des Datensatzes an. Übertragen Sie den Datensatz nach Spannung AUS/EIN erneut.
ID	Input	HW_SUBMODULE	E, A, M, D, L oder Konstante	Adresse der ET 200S Übernehmen Sie die Adresse aus der Hardware-Konfiguration.
SLOT_NO_X	Input	INT	E, A, M, D, L, P oder Konstante	Steckplatznummer des X. schaltbaren Powermoduls.
FUNC_X	Input	INT	E, A, M, D, L, P oder Konstante	Funktion der Baugruppe auf diesem Steckplatz. Über den Parameter FUNC_X legen Sie das Schaltverhalten des PM-E (Powermodul der ET 200S) fest: • FALSE: – Bei "PAUSE_START": -Kein Einfluss auf PM-E -PM-E bleibt eingeschaltet – Bei "PAUSE_STOP": -Schaltet PM_E wieder ein • TRUE: – Bei "PAUSE_START": -Schaltet PM_E aus – Bei "PAUSE_STOP": -Schaltet PM-E wieder ein
BUSY	Output	BOOL	E, A, M, D, L	Übertragung noch nicht abgeschlossen.
DONE	Output	BOOL	E, A, M, D, L	Übertragung ohne Fehler beendet.
ERROR	Output	BOOL	E, A, M, D, L	Übertragung mit Fehler beendet.
STATUS	Output	DWORD	E, A, M, D, L, P	Fehlernummer (siehe Parameter STATUS der Anweisung "PE_Start_End (Seite 2467)")

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Siehe auch

Beschreibung PROFIenergy (Seite 2464)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

PE_WOL: Energiesparmodus über WakeOnLan starten und beenden

Beschreibung PE_WOL

Beschreibung

Die Anweisung "PE_WOL" sendet die PROFIenergy-Kommandos "Start_Pause" und "End_Pause" an mehrere PROFIenergy-fähige Devices im PROFINET I/O Systemen.

Über die Anweisung ist die Koordination mehrerer PE-Devices möglich, vorausgesetzt die PE-Devices unterstützen die Funktion "Wake on LAN" über eine UDP-Verbindung.

Die Anweisung "PE_WOL" ist nur auf einer CPU mit integrierter Ethernet-Schnittstelle ausführbar. Diese CPU muss in der Lage sein, Bausteine mit einer Größe von ca. 400KB zu laden. Wenn diese über einen Ethernet CP angebunden sind, können Sie den Baustein nicht mit PROFINET I/O Systemen einsetzen.

Die Anweisung "PE_WOL" wird asynchron abgearbeitet.

Die Anweisung "PE_WOL" unterstützt nur Devices mit einer Gerätenummer kleiner 256. Devices mit einer größeren Gerätenummer werden nicht berücksichtigt.

Definition: Wake on LAN

Über Wake On LAN können abgeschaltete Geräte durch den Empfang eines speziellen Ethernet-Pakets wieder aktiviert werden.

Damit dieses Verfahren funktionieren kann, muss die Datenverarbeitungseinrichtung über einen Netzwerk-Controller verfügen, der für den Empfang eines solchen Pakets ausgerüstet ist.

Dieses Paket (Magic Packet™) hat ein spezielles Format. Es beinhaltet 15mal die MAC Adresse des Netzwerkadapters.

Auswahl der Devices

Die Auswahl der Devices erfolgt über den Anwender-Datenbaustein am Parameter PENERGY (Typ: “PE_PLUS“). Der Anwender-DB stellt die Datenbasis für die Bearbeitung mehrerer Devices dar.

Vor der Initialisierung von "PE_WOL" müssen Sie in dem Anwender-DB mindestens folgende Informationen hinterlegen:

- ID des PROFINET I/O Systems
- Daten der Verbindung, die für "Wake On LAN" genutzt wird
- Port-Nummer, die für "Wake On LAN" genutzt wird
- Für jedes Device
  - Pausenzeit (PauseTime)
  - Schalten des Device in den PE_SLEEP_MODE (EnableSleep)

Über den Parameter COM_RST initialisieren Sie die Anweisung "PE_WOL". Nach der Initialisierung werden die im Anwender-DB hinterlegten Aufträge nacheinander abgearbeitet.

Die folgende Grafik zeigt beispielhaft, wie das PE-Kommando "Start_Pause" an mehrere Devices übertragen wird:

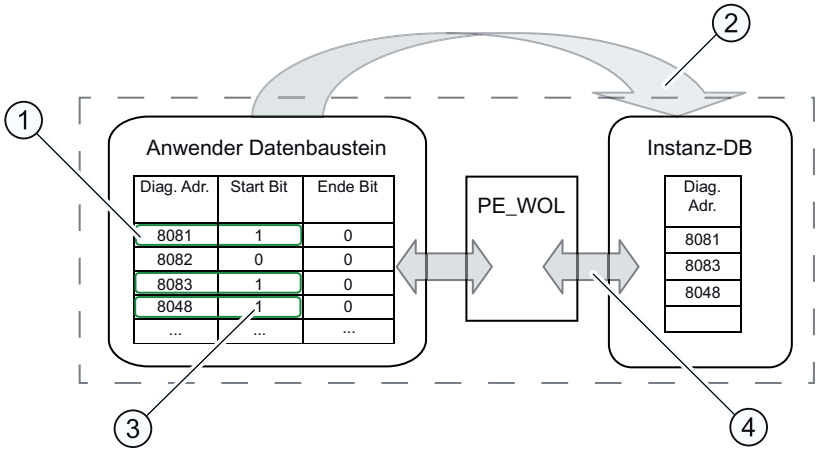

(1) Schritt 1: Das Bit "CmdStartPause" der abzuschaltenden Devices wird durch den Anwender auf "1" gesetzt.

(2) Schritt 2: Die Diagnose-Adressen der abzuschaltenden Devices (CmdStartPause = "1") werden in die Warteschlange eingekettet.

(3) Schritt 3: Das Bit "CmdStartPause" wird automatisch zurückgesetzt, nachdem die Aufträge eingekettet wurden.

(4) Schritt 4: Die Anweisung "PE_WOL" bearbeitet die Aufträge, sobald diese eingekettet sind.

Über die Parameter START und END kann ein PROFIenergy-Kommando "CmdStartPause" bzw. "CmdEndPause" für alle erkannten Geräte im PROFINET IO System gesendet werden.

Über den Parameter STATUS wird der Stand der Auftragsbearbeitung sowie mögliche Fehler bei der Bearbeitung ausgegeben.

Bedienung der Anweisung über den Anwender-DB

Die Bedienung der Anweisung "PE_WOL" erfolgt ausschließlich über den Anwender-DB. Hierbei gilt ein grundsätzlicher Ablauf:

1. Anwahl des auszuführenden Kommandos für ein Device:
   - START_PAUSE ("CmdStartPause" im Anwender-DB)
   - ENDE_PAUSE ("CmdEndPause" im Anwender-DB)
   - UPDATE_STATUS ("CmdUpdateStatus" im Anwender-DB)
2. Setzen des Bits zur Aktualisierung ("Update" im Header des Anwender-DBs)
   Zwischen zwei Aktualisierungen sollte mindestens ein CPU Zyklus verstreichen mit einem "Update" = False, da sonst eine Flankenerkennung nicht gewährleistet werden kann.

Priorisierung der PE-Kommandos

Die folgende Grafik zeigt den zeitlichen Ablauf der drei möglichen Kommandos.

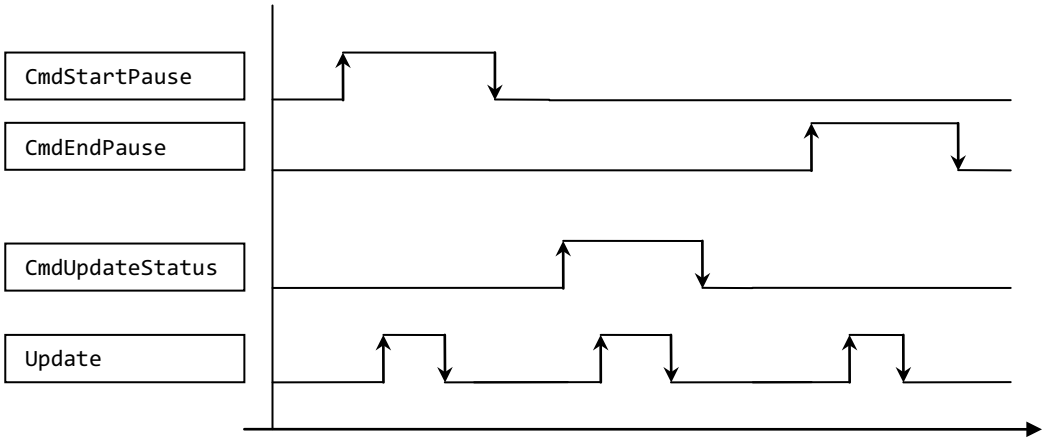

Diese werden nacheinander abgearbeitet, unabhängig davon ob der jeweils vorherige Kommandoaufruf erfolgreich war oder mit Fehler beendet wurde.

Bei gleichzeitigem Setzen von, z. B. "CmdEndPause" und "CmdUpdateStatus" wird nur ein Kommando ausgeführt. Innerhalb des Bausteins liegt eine Priorisierung vor:

- Das Kommando "CmdStartPause" hat die höchste Priorität und wird folglich immer ausgeführt, sofern angewählt.
- Das Kommando "CmdEndPause" hat die zweithöchste Priorität.
- Das Kommando "CmdUpdateStatus" hat die niedrigste Priorität.

Wenn alle drei Kommandos gleichzeitig gesetzt werden, bleiben die nicht ausgeführten Kommandos vorgewählt. In diesem Falle wird die nächste steigende Flanke das nächste Kommando ausführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_WOL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
COM_RST (Seite 2481)	Input	BOOL	E, A, M, D, L	Setzt den Baustein zurück und führt eine Neuinitialisierung durch. Solange hier ein True gesetzt ist, wird die Initialisierung eingeleitet, jedoch noch nicht vollständig durchgeführt. Erst die fallende Flanke setzt die Initialisierung fort und wechselt nach der Initialisierung in den normalen Betriebsmodus.
START (Seite 2483)	Input	BOOL	E, A, M, D, L	Eine steigende Flanke führt ein PROFIenergy-Kommando "CmdStartPause" für alle erkannten Geräte durch, die diese Funktion unterstützen.
END (Seite 2484)	Input	BOOL	E, A, M, D, L	Eine steigende Flanke führt ein PROFIenergy-Kommando "CmdEndPause" für alle erkannten Geräte durch, die diese Funktion unterstützen.
PENERGY (Seite 2484)	InOut	PE_PLUS	D	Zeiger auf den Anwender-DB, der die Datenbasis für die Bearbeitung mehrerer Devices enthält.
STATUS (Seite 2489)	Output	DWORD	E, A, M, D, L	Status-/Fehlernummer zum aktuellen Zustand der Anweisung (siehe "Parameter STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter COM_RST

Ablauf der Initialisierungsroutine

Über den Parameter COM_RST starten Sie die Initialisierung der Anweisung "PE_WOL".

Das folgende Ablaufdiagramm zeigt die Initialisierungsroutine.

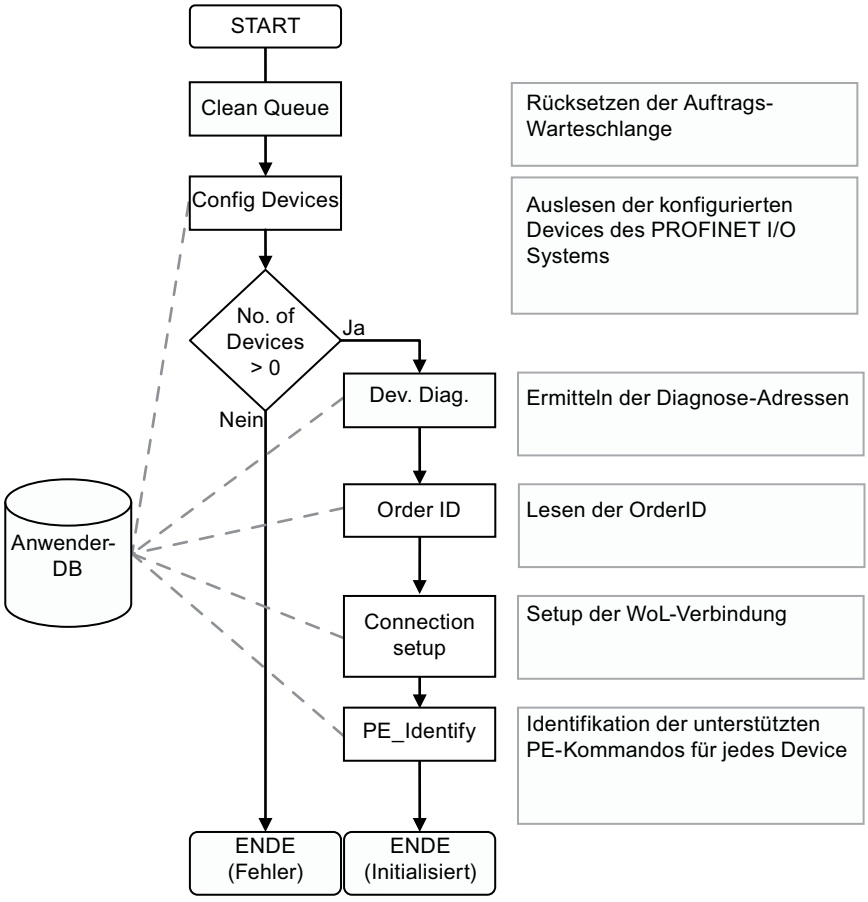
START
Clean Queue
Rücksetzen der Auftrags-Warteschlange
Config Devices
Auslesen der konfigurierten Devices des PROFINET I/O Systems
No. of Devices > 0
Ja
Nein
Dev. Diag.
Ermitteln der Diagnose-Adressen
Order ID
Lesen der OrderID
Anwender-DB
Connection setup
Setup der WoL-Verbindung
PE_Identify
Identifikation der unterstützten PE-Kommandos für jedes Device
ENDE (Fehler)
ENDE (Initialisiert)

Parameter START

Ablauf des CmdStartPause Kommandos

Das Ablaufdiagramm zeigt die intern genutzten Funktionen und Interaktion mit einem Device beim Ausführen des CmdStartPause Kommandos.

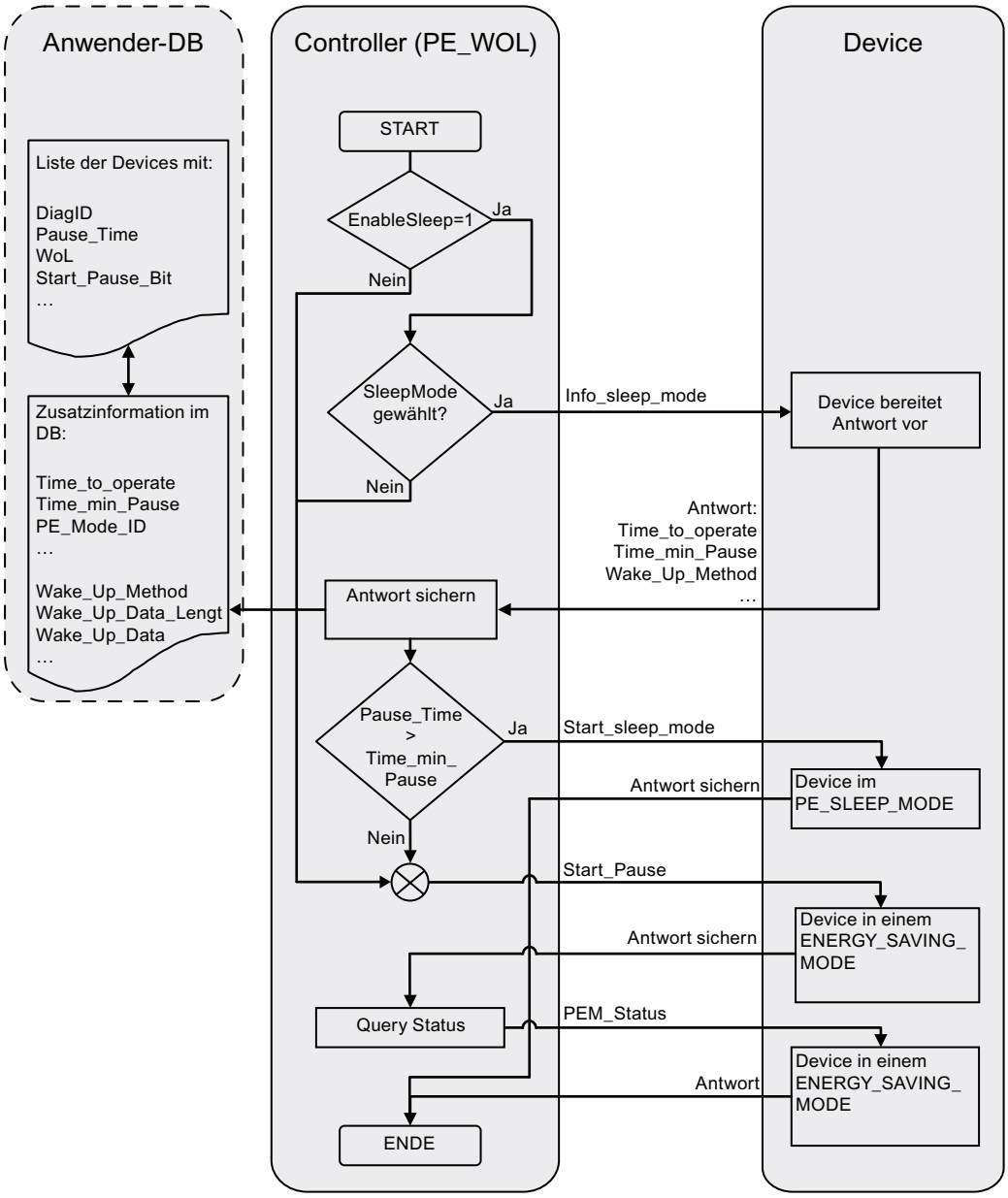

Parameter END

Ablauf des CmdEndPause Kommandos

Das Ablaufdiagramm zeigt die intern genutzten Funktionen und Interaktion mit einem Device beim Ausführen des CmdEndPause Kommandos.

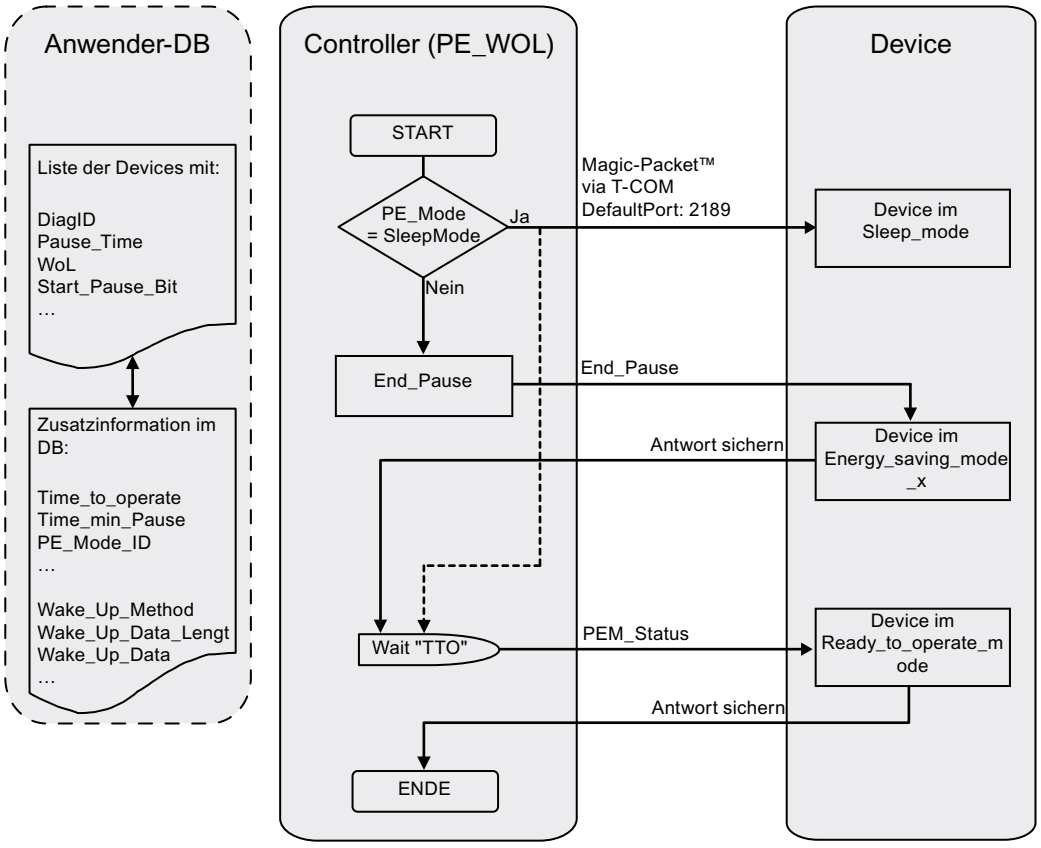

Parameter PENERGY

Datenbaustein am Parameter PENERGY

Der Anwender-DB für die PROFIenergy Anweisung "PE_WOL" stellt eine Datenbasis für die Bearbeitung mehrerer Devices dar.

Der Datenbaustein ist generell in zwei Teile gegliedert. Diese sind:

- Kopfteil 110 Byte (Header)
- Device-Teil für maximal 256 Devices mit je 100 Byte (Device). Diese enthalten:
  - Device-spezifische Daten (Device)
  - PROFIenergy-spezifische Daten (PE)
  - Daten der Auftragsbearbeitung (Task)
  - Benutzerdaten (UserData)

Der Datenbaustein arbeitet mit optimiertem Zugriff.

Verbindungsparameter "Connection"

Die Anweisung "PE_WOL" reserviert eine Verbindungsressource aus dem Bereich der "Open User Communication". Diese wird als UDP-Verbindung genutzt. Hierfür müssen im Datenbaustein folgende Parameter definiert werden:

- Verbindungs-ID (Parameter "Connection.id")
  Die Verbindungs-ID ist eine ganze Zahl, die im Bereich von 1 bis 4095 liegt. Sie dient der Identifikation der firmwareseitig zugeordneten Kommunikationsressourcen, wie Sende- und Empfangspuffer.
  Die Verbindungs-ID muss CPU-weit eindeutig vergeben werden.
- Port-Nummer, die für die "Wake On LAN" Funktion genutzt wird (Parameter "Header.PortNo")
  Nummer des UDP-Ports über die ein "Wake On LAN" Paket versendet wird. Diese Port-Nummern sind Bestandteil der Kommunikationsressourcen, die firmwareseitig durch die Verbindungs-ID identifiziert und bereitgestellt werden. Die hier verwendete Standardeinstellung für den Port von 2189 ist durch die IANA zurzeit nicht zugeordnet. Die Port-Nummer wird auf die Verbindungsprojektierung übertragen und für den Parameter "Connection.local_tsap_id[1]" übernommen. Die Port-Nummer und Länge des entfernten Ports werden über die Parameter "Connection.rem_tsap_id[1]" und "Connection.rem_tsap_id_LEN" definiert und müssen händisch vergeben werden.

Aufbau des Datenbausteins

Der Datenbaustein hat den folgenden Aufbau:

Name	Datentyp	Offset	Kommentar
Header	PE_HEADER	-	Kopfinformationen
Update (1)	BOOL	-	Signal zur Anzeige einer Änderung im Datenbereich. • True= zeigt eine Änderung durch den Anwender an. • False= zeigt die Übernahme der Änderungen an.
Initialized	BOOL	-	Signal zur Anzeige der abgeschlossenen Initialisierung. • True=Initialisierung fertig. • False= zeigt an, dass der Baustein nicht initialisiert ist.
LinkUp	BOOL	-	Zeigt eine erfolgreiche Konfiguration der Ethernetschnittstelle an. • True=Schnittstelle einsatzbereit. • False=Schnittstelle noch nicht konfiguriert.
LinkDown	BOOL	-	Zeigt eine unkonfigurierte Schnittstelle an. • True=Schnittstelle ist nicht konfiguriert. • False=Schnittstelle wird gerade oder ist konfiguriert.
PROFINET_ID (1)	INT	-	ID des PROFINET I/O Systems
Reserved	ARRAY [1..37] OF BYTE	-	Reserviert
LastDeviceID	INT	-	Enthält die größte Device-ID in diesem PROFINET I/O System.
PortNo (1)	WORD	-	Port-Nummer, die für die "Wake On LAN" Funktion genutzt wird (Default= 2189).
Connection	TCON_Param	-	Enthält die Verbindungsprojektierung der "Wake on LAN" Verbindung.

Name		Datentyp	Offset	Kommentar
	BLOCK_LENGTH	UInt	-	Länge der Struktur (immer B#16#40).
	ID (1)	CONN_OUC	-	Verbindungs-ID
	CONNECTION_TYPE (1)	USINT	-	Verbindungstyp (UDP = B#16#13)
	ACTIVE_EST (1)	BOOL	-	Aktiver Verbindungsaufbau (bei UDP immer passiv)
	LOCAL_DEVICE_ID (1)	USINT	-	Enthält die Schnittstellen-ID (CPU abhängig)
	LOCAL_TSAP_ID_LEN (1)	USINT	-	Enthält die Länge in Bytes des eigenen/lokalen UDP-Ports
	REM_SUBNET_ID_LEN (1)	USINT	-	Ungenutzt (immer B#16#00)
	REM_STADDR_LEN (1)	USINT	-	Enthält die Länge der entfernten IP-Adresse oder B#16#00.
	REM_TSAP_ID_LEN (1)	USINT	-	Enthält die Länge in Bytes des entfernten UDP-Ports. Die Längenangabe muss händisch eingetragen werden.
	NEXT_STADDR_LEN (1)	USINT	-	Enthält die Länge der Default-Router-Adresse (nicht relevant).
	LOCAL_TSAP_ID (1)	ARRAY[1..16] OF BYTE	-	Enthält die eigene/lokale Port-Nummer. Der Wert wird bei der Initialisierung aus dem Parameter PortNo übernommen.
	REM_SUBNET_ID (1)	ARRAY[1..6] OF USINT	-	Ungenutzt (immer B#16#00)
	REM_STADDR (1)	ARRAY[1..6] OF USINT	-	Enthält die entfernte IP-Adresse.
	REM_TSAP_ID (1)	ARRAY[1..16] OF BYTE	-	Enthält die entfernte UDP-Port-Nummer. Die Port-Nummer muss händisch eingetragen werden.
	NEXT_STADDR (1)	ARRAY[1..6] OF BYTE	-	Irrelevant
	SPARE (1)	WORD	-	
Device		ARRAY[1..256] OF PE_DEVICE	-	Array der Devices
	Device	PE_DEV	-	Enthält Daten für jedes Device
	DeviceID	HW_DEVICE	-	Hardware-Kennung des Device. Wird durch die Hardware-Konfiguration zugewiesen.
	PE_EntityID	HW_IO	-	Hardware-Kennung der PROFIenergy-Entity. Wird durch die Hardware-Konfiguration zugewiesen.
	MACAdr	ARRAY[1..6] OF BYTE	-	Enthält die MAC-Adresse des Device.
	IPAdr	ARRAY[1..4] OF BYTE	-	Enthält die IP-Adresse des Device.
	OrderID_MxLen	Byte	-	Enthält maximale Länge der OrderID.
	OrderId_ActLen	Byte		Enthält aktuelle Länge der OrderID.
	OrderID_Data	ARRAY[1..20] OF CHAR		Enthält die OrderID des Device.
	PE	PE_PE	-	PROFIenergy spezifische Daten.

Name	Datentyp	Offset	Kommentar
ModeID	BYTE	-	PE_MODE_ID nach PROFIenergy Spezifikation.
Result	BYTE	-	PE ErrorCode nach PROFIenergy Spezifikation.
PauseTime (1)	TIME	-	Enthält die Pausenzeit in ms.
TimeToPause	TIME	-	Enthält die Zeit, die das Device benötigt, um in den Pause-Modus zu gelangen.
TimeToOperate	TIME	-	Enthält die Zeit, die das Device benötigt, um in den Betriebsmodus zu gelangen.
MinSleepTime	TIME	-	Enthält die minimale Zeit des Devices im PE_SLEEP_MODE.
SleepToOperate	TIME	-	Enthält die Zeit, die das Device benötigt um vom PE_SLEEP_MODE betriebsbereit zu werden.
StatusOperate	BOOL	-	Zeigt den Betriebs-Modus des Device an.
StatusPause	BOOL	-	Zeigt den Pause-Modus des Devices an.
StatusSleep	BOOL	-	Zeigt den PE_SLEEP_MODE des Device an.
StatusTransitOK	BOOL	-	Zeigt den Übergang von einem Energiezustand in einen anderen an.
StatusInTransit	BOOL	-	Zeigt einen aktuellen Zustandsübergang an.
StatusTransitNOK	BOOL	-	Zeigt an, dass der Zustandswechsel nicht erfolgreich war.
StatusError	BOOL	-	Zeigt einen Fehler mit dem Device an.
StatusRetryEx	BOOL	-	Zeigt eine nicht erfolgreiche Ausführung eines Kommandos an. Es wird nicht mehr versucht, dieses Kommando erneut auszuführen.
CmdStartPause (1)	BOOL	-	Platziert ein START_PAUSE Kommando für dieses Device in die Warteschleife.
CmdEndPause (1)	BOOL	-	Platziert ein END_PAUSE Kommando für dieses Device in die Warteschleife.
CmdUpdateStatus (1)	BOOL	-	Platziert ein PEM_STATUS Kommando für dieses Device in die Warteschleife.
EnableSleep (1)	BOOL	-	Erlaubt den PE_SLEEP_MODE für dieses Device. • True= Device soll in den PE_SLEEP_MODE, wenn die Pausenzeit lang genug ist • False=Device darf nicht in den PE_SLEEP_MODE.
Services	WORD	-	Zeigt alle unterstützten PROFIenergy Services an.
UserData (2)	ARRAY[1..24] OF BYTE	-	Nutzer definierte Daten
Task	PE_TASK	-	Auftragsbearbeitung

Name		Datentyp	Offset	Kommentar
	Cmd	BYTE	-	Interne Bits für die Auftragsbearbeitung
	CmdJ	BYTE	-	Internal Bits für die Auftragsbearbeitung
	TimeStart	BOOL	-	Startet eine Verzögerungszeit.
	TimeStarted	BOOL	-	Die Verzögerungszeit wurde gerade gestartet.
	TimeDone	BOOL	-	Zeigt den Ablauf der Verzögerungszeit an.
	Done	BOOL	-	Zeigt an, dass dieser Auftrag beendet wird.
	DelayedCmd	BOOL	-	Zeigt an, dass ein verzögertes Kommando noch ansteht.
	IsV1_0	BOOL	-	Zeigt an, dass dieses Gerät ein Spec. V1.0 Gerät ist.
	IsWakeOnLAN	BOOL	-	Zeigt an, dass dieses Gerät per "Wake On LAN" geweckt wird.
	RetryCount	BYTE	-	Wiederholungszähler für PE_COMMANDS
	Duration	TIME	-	Enthält den Zeitwert für die Verzögerung in ms.
	StartTime	TIME	-	Enthält den Startzeitpunkt der Verzögerungszeit.
	MachineState	INT	-	Enthält den internen Status des Auftrags.
(1) Durch den Anwender zu befüllen. (2) Für den Anwender frei verwendbar.				

Parameter STATUS

Parameter STATUS

Der Ausgabewert am Parameter STATUS ist in drei Bereiche unterteilt:

- Bit 31 bis 24: MESSAGE
- Bit 23 bis 16: LOCATION
- Bit 15 bis 0: INFORMATION

Die Bedeutung der einzelnen Fehlercodes der drei Bereiche entnehmen Sie den folgenden Tabellen:

Tabelle 4-30 Mögliche Werte für MESSAGE

Fehlercode (W#16#...)	Beschreibung
00	Kein Fehler.
50	Anweisung initialisiert.
51	Konfiguration des PROFINET I/O Systems wird ermittelt.
52	Die Anweisung konnte keine konfigurierten Devices am PROFINET I/O System ermitteln.
53	Die logischen Adressen der konfigurierten Devices werden ermittelt.
54	Schnittstelleninformationen der Devices werden ausgelesen.
55	I&M Daten (nur Datensatz 0) der konfigurierten Devices werden ermittelt.
56	PROFINET-Schnittstelle für den Versand des "Wake on LAN" MagicPaket™ über UDP wird konfiguriert.
57	PROFIenergy-Fähigkeiten der angeschlossenen Geräte wird ermittelt.
62	Ungültige PROFINET I/O System ID erkannt. Die verursachende Nummer wird im Feld INFORMATION angezeigt.

Fehlercode (W#16#...)	Beschreibung
70	Die Anweisung ist initialisiert und bearbeitet Aufträge. Der Wert im Feld INFORMATION enthält die Anzahl aktuell aktiver Aufträge.
80	Die Anweisung hat während der Bearbeitung von Aufträgen die Initialisierung verloren. Dies tritt im Allgemeinen dann auf, wenn entweder der Instanz-DB oder der Anwender-DB neu geladen werden.
FF	Unbekannter Fehler aufgetreten.

Tabelle 4-31 Mögliche Werte für LOCATION

Fehlercode (W#16#...)	Beschreibung
00	Die Anweisung ist entweder nicht initialisiert oder im Leerlauf.
70	Die Anweisung wartet auf Aufträge.
71	Die Anweisung hängt einen Auftrag in die Auftragsliste ein.
72	Die Anweisung bereitet den Auftragsversand vor.
73	Die Anweisung versendet einen Auftrag an ein Device.
74	Die Anweisung wartet auf eine Antwort vom Device.
75	Die Anweisung wertet die Antwort des Devices aus.
76	Die Anweisung entfernt den Auftrag aus der Auftragsliste.
FF	Unbekannter Fehler aufgetreten.

Tabelle 4-32 Mögliche Werte für INFORMATION

Fehlercode (W#16#...)	Beschreibung
0000	Keine Zusatzinformationen vorhanden und keine Aufträge aktiv.
0001 -00FF	Es werden aktuell 1 – 255 Aufträge bearbeitet.
7000	Neuinitialisierung mit COM_RST eingeleitet, jedoch noch nicht vollständig durchgeführt.
8001	Fehler am 1. Parameter
8002	Fehler am 2. Parameter
8003	Fehler am 3. Parameter
8004	Fehler am 4. Parameter
8005	Fehler am 5. Parameter. Dieser Fehler wird gemeldet, sofern es keine oder eine ungültige Verschaltung zum Anwender-DB gibt. Mögliche Ursachen: • Der Anwender-DB ist zu klein. • Der Anwender-DB ist schreibgeschützt. • Der Anwender-DB ist nicht im RAM vorhanden. • Der Anwender-DB ist ungültig für die eingesetzte CPU.
8085 bis 80CE	Fehler beim Aufbau der Verbindung. Es werden die Fehlermeldungen der intern verwendeten Anweisung TCON ausgegeben. Die Beschreibung der Fehlermeldungen entnehmen Sie der Tabelle zum Parameter STATUS (Seite 3692).

Fehlercode (W#16#...)	Beschreibung
8100	Es wurde versucht, mehr als die maximal 256 möglichen Aufträge zu platzieren. Es handelt sich hierbei um einen temporären Fehler, der sich mit dem Abschluss einiger Aufträge behebt. Der platzierte Auftrag wurde nicht entgegen genommen und muss erneut platziert werden.
8200	Es wurde versucht, ein ungültiges oder nicht unterstütztes PROFIenergy-Kommando (PE_COMMAND) zu senden.
8400	Verbindungs-ID außerhalb des erlaubten Bereichs. Initialisierung abgebrochen. Kontrollieren Sie die ID der Verbindungsprojektierung für "Wake on LAN". Siehe Datenbaustein am Parameter PENERGY > Header > Connection > ID.
84xx	Es ist ein Kommunikationsfehler aufgetreten. Die Nummer des Devices, das den Fehler erzeugt hat, wird an "xx" ausgegeben.
85xx	Es wurde ein Fehler vom Device xx zurückgemeldet. Die Nummer des Devices, das den Fehler erzeugt hat, wird an "xx" ausgegeben.
8600	Die angeforderte WakeUp-Methode wird derzeit nicht unterstützt.
FFFF	Unbekannter Fehler aufgetreten.

Siehe auch

Parameter PENERGY (Seite 2484)

PROFIenergy-Kommandos

Aufbau der Anworttelegramme

Aufbau des Antworttelegramms entsprechend dem PROFIenergy-Profil

Die folgende Tabelle zeigt den grundsätzlichen Aufbau des Antworttelegramms gemäß des PROFIenergy-Profils. Das Antworttelegramm besteht aus einem allgemeinen Teil (Header) und einem spezifischen Teil (Service Data Response). Der Inhalt des spezifischen Teils des Antworttelegramms ist jeweils in der Beschreibung des entsprechenden PROFIenergy-Kommandos enthalten.

Blockdefinition	Attribute	Wert	Datentyp	Beschreibung
BlockHeader	BlockType	801_{hex}	WORD	
	BlockLength		WORD	Anzahl der Bytes ohne Berücksichtigung der Felder BlockType und BlockLength.
	BlockVersionHigh	1_{hex}	BYTE	
	BlockVersionLow	0_{hex}	BYTE	

Blockdefinition	Attribute	Wert	Datentyp	Beschreibung
Response Header	Service_Request_ID	1_{hex} bis FF $_{hex}$	BYTE	ID des durchgeführten PE-Kommandos. Die ID des durch die PE-Entity bearbeiteten PE-Kommandos wird im Antworttelegramm zurückgesendet: • 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identify • 06 bis 09: reserviert • 16: Query_Measurement • 11 bis CF: reserviert • D0 bis FF: herstellerspezifisch
	Request_Reference	1_{hex} bis FF $_{hex}$	BYTE	Eindeutige Nummer zur Identifikation des Anfrage/Antwort-Paars (wird vom Server in der Antwort zurückgeliefert).
Service Header Response	Status	1_{hex} bis FF $_{hex}$	BYTE	Information, ob das PE-Kommando ausgeführt wurde: • 00: reserviert • 01: abgeschlossen • 02: mit Fehler abgeschlossen • 03: Daten unvollständig • 04 bis CF: reserviert • D0 bis FF: Abhängig von der Service_Request_ID
	Data_Structure_ Identifier_RS	1_{hex} bis FF $_{hex}$	BYTE	• 00: reserviert • 01 bis FF: Datenstruktur abhängig von der Service_Request_ID • 0xFF - Fehler
Service Data Response				Abhängig von der Service-Request-ID: • Zu den Service-Request-IDs, siehe Parameter CMD und CMD_MODIFIER der Anweisung "PE_CMD (Seite 2472)". • Die spezifischen Inhalte des Antworttelegramms sind jeweils in der Beschreibung des PE-Kommandos enthalten (siehe z. B. Kommando "Start_Pause (Seite 2493)").

PE-Kommando "Start_Pause"

Beschreibung

Mit dem PE-Kommando "Start_Pause" starten Sie den Energiesparmodus. Das Kommando Start_Pause kann verwendet werden um:

- Die PE-Entity vom Status "Betriebsbereit" (PE_ready_to_operate) in einen Energiesparmodus (PE_energy_saving_mode) zu schalten.
- Die PE-Entity von einem Energiesparmodus in einen anderen zu schalten.
  Der Energieverbrauch kann durch das Umschalten des Energiesparmodus steigen oder sinken.

Aufruf des PE-Kommandos "Start_Pause"

Der Aufruf des Kommandos "Start_Pause" mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	1	Aufruf des PE-Kommandos "Start_Pause".
CMD_MODIFIER	0	Bei Kommando "Start_Pause" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	4	Länge des Parameters CMD_PARA von 4 Byte.
CMD_PARA	VARIANT	VARIANT-Zeiger auf Wert für "Pause_Time" (TIME).

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe Anweisung "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID	1 bis 255	BYTE	Identifikationsnummer des Energiesparmodus
Reserviert	0	BYTE	-

PE-Kommando "End_Pause"

Beschreibung

Mit dem PE-Kommando "End_Pause" beenden Sie den Energiesparmodus der PE-Entity.

Aufruf des PE-Kommandos "End_Pause"

Der Aufruf des Kommandos "End_Pause" mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	2	Aufruf des PE-Kommandos "End_Pause".
CMD_MODIFIER	0	Bei Kommando "End_Pause" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Time_to_operate	-	DWORD	Erwartete Zeit, um in den Modus "PE_ready_to_operate" umzuschalten.

PE-Kommando "Query_modes" - "List_Energy_Saving_Modes"

Beschreibung

Mit dem PE-Kommando "Query_modes" und dem Unterkommando (Modifier) "List_Energy_Saving_Modes" geben Sie alle durch die PE-Entity unterstützten Energiesparmodi (PE_Mode_ID) aus.

Das Abfrageergebnis wird in Form des Antworttelegramms in den über den Parameter RESPONSE_DATA referenzierten Datenbaustein geschrieben.

Aufruf des PE-Kommandos "Query_modes" - "List_Energy_Saving_Modes"

Der Aufruf des Kommandos"List_Energy_Saving_Modes" mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	3	Aufruf des PE-Kommandos "Query_modes".
CMD_MODIFIER	1	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "List_Energy_Saving_Modes" zur Ausgabe der Anzahl und Typen der unterstützten Energiesparmodi.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Number_of_PE_ Mode_IDs	1	BYTE	Die Anzahl der Energiesparmodi.
PE_Mode_IDs	-	Array [...] of BYTE	Array mit den IDs der unterstützten Energiesparmodi. Die Bedeutung der einzelnen IDs ist abhängig von der PE-Entity.

PE-Kommando "Query_modes" - "Get_Mode"

Beschreibung

Mit dem PE-Kommando "Query_modes" und dem Unterkommando (Modifier) "Get_Mode" geben Sie die Attribute zu dem momentan aktivierten Energiesparmodus aus.

Aufruf des PE-Kommandos "Query_modes" - "Get_Mode"

Der Aufruf des Kommandos mit dem mit der Anweisung "PE_CMD" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	3	Aufruf des PE-Kommandos "Query_modes"
CMD_MODIFIER	2	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Mode" zur Ausgabe des Status des momentan aktivierten Modus
CMD_PARA_LEN	1	Länge des Parameters CMD_PARA von 1 Byte
CMD_PARA	VARIANT	VARIANT -Zeiger auf den Wert für PE_MODE_ID

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID	• 0 Modus "PE_power_off" • 1...254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	ID des momentan aktiven Energiesparmodus
PE_Mode_Attributes	Bit 0: • = 0: Nur statische Energieverbrauchs- und Zeitwerte verfügbar. • = 1: Dynamische Energieverbrauchs- und Zeitwerte verfügbar. Bit 1 bis 7: • Reserviert	BYTE	
Time_min_Pause [1]	Zeitdifferenz ohne Datum	DWORD	Minimale Pausenzeit für den PE-Modus. Die minimale Pausenzeit ist die Summe aus den Werten der folgenden Attribute: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay Siehe Beschreibung "Parameter PAUSE_TIME" der Anweisung "PE_START_END: Energiesparmodus starten und beenden (Seite 2467)".
Time_to_Pause [1]	Zeitdifferenz ohne Datum	DWORD	Ausschaltzeit: Zeitdauer vom Aufruf des Energiesparmodus bis zum Start des Energiesparmodus (Übergangsdauer von PE_ready_to_operate zu PE_energy_saving_mode). Die Ausschaltzeit ist abhängig von der PE-Entity.
Time_to_operate [1]	Zeitdifferenz ohne Datum	DWORD	Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate). Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet.
Time_min_length_of_stay [1]		DWORD	Mindestdauer für den Energiesparmodus in der PE-Entity
Time_max_length_of_stay [1]		DWORD	Maximale Dauer für den Energiesparmodus in der PE-Entity

Attribut	Wert	Datentyp	Beschreibung
Mode_Power_ Consumption [2]		REAL	Stromverbrauch der PE-Entity im aktivierten Energiesparmodus Einheit: kW
Energy_ Consumption_ to_pause [2]		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Bereitschaftsmodus (PE_ready_to_operate) zum Energiesparmodus (PE_energy_saving_mode) Einheit: kWh
Energy_ Consumption_ to_operate [2]		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate) Einheit: kWh
[1] Wenn die Zeitdauer unendlich ist, wird 0xFFFFFFFF ausgegeben. Wenn die Zeitdauer null ist, wird "0" ausgegeben. [2] Wenn die Daten zu Energie- und Stromverbrauch durch die PE-Entity nicht definiert sind, wird als Wert "0,0" ausgegeben.			

PE-Kommando "PEM_Status"

Beschreibung

Mit dem PE-Kommando "PEM_Status" fragen Sie den Status eines momentan aktiven Energiesparmodus einer PE-Entity ab.

Aufruf des PE-Kommandos "PEM_Status"

Der Aufruf des Kommandos "PEM_Status" mit der Anweisung "PE_CMD" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	4	Aufruf des PE-Kommandos "PEM_Status".
CMD_MODIFIER	0	Bei Kommando "PEM_Status" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID_ Source	• 0 Modus "PE_power_off" • 1 bis 254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	Modus, in dem sich die PE-Entity vor dem Absetzen eines PE-Kommandos befindet.
PE_Mode_ID_ Destination	• 0 Modus "PE_power_off" • 1 bis 254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	Modus, in dem sich die PE-Entity nach der Durchführung eines PE-Kommandos befindet.
Time_to_operate	Zeitdifferenz ohne Datum.	DWORD	Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate). Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet.
Remaining_time_to_ destination	Zeitdifferenz ohne Datum.	DWORD	Verbleibende Zeit zur Umschaltung in den anderen Modus.
Mode_Power_ Consumption		REAL	Stromverbrauch der PE-Entity im aktivierten Energiesparmodus. Einheit: kW
Energy_ Consumption_ to_Destination		REAL	Energieverbrauch für momentane PE-Transition Einheit: kWh
Energy_ Consumption_ to_operate		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy-saving mode) zum Bereitschaftsmodus (PE_ready_to_operate) Einheit: kWh

PE-Kommando "PE_Identify"

Beschreibung

Mit dem PE-Kommando "PE_Identify" lesen sie die Anzahl und die Beschreibung der durch die PE-Entity unterstützten PE-Kommandos aus. Welche und wie viele Kommandos unterstützt werden, ist abhängig von der PE-Entity. Da PE_Identify selbst ein PE-Kommando ist, werden bei einer positiven Antwort mindestens drei unterstützte PE-Kommandos ausgegeben: Start_Pause, End_Pause und PE_Identify.

Aufruf des PE-Kommandos "PE_Identify"

Der Aufruf des Kommandos "PE_Identify" mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	5	Aufruf des Kommandos "PE_Identify".
CMD_MODIFIER	0	Bei Kommando "PE_Identify" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count [1]	6	BYTE	Anzahl der unterstützten PROFIenergy-Kommandos
Start_Pause	1	BYTE	Erstes unterstütztes PE-Kommando (Service_Request_ID)
End_Pause	2	BYTE	...
Query_Modes	3	BYTE	...
PEM_Status	4	BYTE	...
PE_Identify	5	BYTE	...
Query_Measurement	16	BYTE	Letztes unterstütztes PE-Kommando (Service_Request_ID)

[1] Die Anzahl der unterstützten Kommandos ist herstellerspezifisch und hängt von der verwendeten PE-Entity ab. Wenn alle 6 PE-Kommandos unterstützt werden, sind die angegebenen Werte ein Beispiel für ein Antworttelegramm.

PE-Kommando "Query_Measurement" - "Get_Measurement_list"

Beschreibung

Mit dem PE-Kommando "Query_Measurement" und dem Unterkommando (Modifier) "Get_measurement_list" führen Sie eine Abfrage durch, welche Messwerte durch die PE-Entity unterstützt werden. Die unterstützten Messwerte werden als Liste in dem über den Parameter RESPONSE_DATA referenzierten Datenbaustein hinterlegt.

Aufruf des PE-Kommandos "Query_Measurement" - "Get_Measurement_list"

Der Aufruf des Kommandos mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	16	Aufruf des Kommandos "Query_Measurement"
CMD_MODIFIER	1	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Measurement_List" zur Ausgabe einer Liste der unterstützten Messwerte.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count	-	BYTE	Anzahl der Measurement-IDs
reserved	-	BYTE	
...			
Measurement_ID	-	WORD	Erste unterstützte Measurement_ID. Die Measurement_ID ist herstellerspezifisch. Weitere Informationen entnehmen Sie dem Handbuch der jeweiligen PE-Entity.
Accuracy_Domain	-	BYTE	Siehe Tabelle "Genauigkeitsbereiche".
Accuracy_Class	-	BYTE	Siehe Tabellen "Genauigkeitsklassen".
Range	-	REAL	Spezifiziert den Skalen-Endwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet).
...			
Measurement_ID	-	WORD	Letzte unterstützte Measurement_ID
Accuracy_Domain	-	BYTE	Siehe Tabelle "Genauigkeitsbereiche".
Accuracy_Class	-	BYTE	Siehe Tabellen "Genauigkeitsklassen".
Range	-	REAL	Spezifiziert den Skalenendwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet).

Genauigkeitsbereiche

Genauigkeitsbereich	Beschreibung
0	Reserviert
1	Die Genauigkeitsabweichung wird in Prozent des Skalenendwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).
2	Die Genauigkeitsabweichung wird in Prozent des aktuellen Messwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).
3	Die Messgenauigkeit erfolgt nach der Norm IEC 61557-12. Die Funktions-Leistungsklassen für Geräte zur Leistungs-Messung und Überwachung (PMD) ohne externe Sensoren und System-Leistungsklassen für PMD mit externen Sensoren sind in der entsprechend der Tabelle "Genauigkeitsklassen des Genauigkeitsbereichs 3" codiert.
4	Die Angabe der Genauigkeit erfolgt entsprechend der Norm EN 50470-3, Kapitel 8 (siehe auch Tabelle: Genauigkeitsklassen des Genauigkeitsbereichs 4).

Genauigkeitsklassen

Tabelle 4-33 Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2

Genauigkeitsklasse	0	1	2	3	4	5	6	7	8
Bedeutung	reserviert	0,01%	0,02%	0,05%	0,1%	0,2%	0,5%	1%	1,5%

Genauigkeitsklasse	9	10	11	12	13	14	15	>15
Bedeutung	2%	2,5%	3%	5%	10%	20%	>20%	Nicht definiert

Tabelle 4-34 Genauigkeitsklassen des Genauigkeitsbereichs 3

Genauigkeitsklasse	0	1	2	3	4	5	6	7	8
Bedeutung	reserviert	0,02	0,05	0,1	0,2	0,5	1	1,5	2

Genauigkeitsklasse	9	10	11	12	13	14	>13
Bedeutung	2,5	3	5	10	20	20%	Nicht definiert

Tabelle 4-35 Genauigkeitsklassen des Genauigkeitsbereichs 4

Genauigkeitsklasse	0	1	2	3	4	5	6	>7
Bedeutung	reserviert	0,5	1,0	1,5	2,0	2,5	3,0	Nicht definiert

PE-Kommando "Query_Measurement" - "Get_Measurement_values"

Beschreibung

Mit dem PE-Kommando "Query_Measurement" und dem Unterkommando (Modifier) "Get_measurement_values" geben Sie die von der PE-Entity unterstützten Messwerte aus. Die Messwerte werden als Liste in dem über den Parameter RESPONSE_DATA referenzierten Datenbaustein hinterlegt.

Aufruf des PE-Kommandos "Query_Measurement" - "Get_Measurement_values"

Der Aufruf des Kommandos mit der Anweisung "PE_CMD (Seite 2472)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	16	Aufruf des Kommandos "Query_Measurement"
CMD_MODIFIER	2	Spezifikation des Kommandoaufrufs: Auswahl des Kommandos "Get_Measurement_Values" zur Ausgabe einer Liste der unterstützten Messwerte.
CMD_PARA_LEN	0	Abhängig von der Anzahl der Messwerte. Die Länge des Parameters ergibt sich aus dem Attribut count und der Summe der Längen der Attribute für die übertragenen Messwerte.
CMD_PARA	VARIANT	VARIANT-Zeiger auf Datenstruktur mit Auflistung der abzufragenden Messwerte (siehe "Parameter CMD_PARA").

Parameter CMD_PARA

Die Struktur, die über den VARIANT-Zeiger am Parameter CMD_PARA angegeben wird, muss den folgenden Aufbau haben:

Attribut	Wert	Datentyp	Beschreibung
Count	-	BYTE	Anzahl der Messwerte (Measurement-IDs)
reserved	0	BYTE	Nicht verwendet
Measurement_ID	-	WORD	Erster abgefragter Messwert
...			
Measurement_ID	-	WORD	Letzter abgefragter Messwert

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 2472)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count [1]	-	BYTE	Anzahl der Messwerte (Measurement-IDs)
reserved	0	BYTE	Nicht verwendet
Length_of_Structure	2 bis 65535	WORD	Länge der Struktur in Bytes.

Attribut	Wert	Datentyp	Beschreibung
Measurement_Data_Structure_ID	1 = simple value	BYTE	Definiert die folgende Struktur.
Measurement_ID	0 bis 65535	WORD	ID des unterstützten Messwertes.
Status_of_ Measurement_Value	1 bis 3	BYTE	Status des Messwerts: • 1: Gültig • 2: Nicht unterstützt • 3: Nicht gültig
Transmission_Data_Type	-	REAL	
End_of_demand	-	TOD	Optionale Zeitstempelung mit dem Datentyp TimeOfDay.
...			
Length_of_Structure	-	WORD	Länge der Struktur in Bytes.
Measurement_Data_Structure_ID	-	BYTE	Definiert die folgende Struktur.
Measurement_ID	-	WORD	ID des unterstützten Messwertes.
Status_of_Measurement_Value	-	BYTE	Status des Messwerts: • 1: Gültig • 2: Nicht unterstützt • 3: Nicht gültig
Transmission_Data_Type	-	REAL	
End_of_demand	-	TOD	Optionale Zeitstempelung mit dem Datentyp TimeOfDay.
[1] Wenn die Datenlänge der abgefragten Messwerte die Größe der PDU (Protocol Data Unit) der Protokollschicht übersteigt, werden die Daten unvollständig übertragen und nur die unterstützten Messwerte ausgegeben.			

iDevice / iSlave

PE_I_DEV: PROFIenergy-Kommandos im iDevice steuern

Beschreibung

Die Anweisung "PE_I_DEV" wird zur Abwicklung des PROFIenergy-Profils im intelligenten IO-Device (iDevice) verwendet.

- Die Anweisung "PE_I_DEV" wird zyklisch vom Anwenderprogramm des iDevice aufgerufen und empfängt alle PROFIenergy-Kommandos.
- Die Generierung der PROFIenergy-Antwort erfolgt durch Parametrierung eines Hilfsbausteins. Die Reaktion in der Pause ist frei programmierbar. Die Antwortdaten müssen innerhalb von 10 Sekunden bereitgestellt werden, da sonst am Parameter STATUS der Anweisung im IO-Controller der "Stateconflict 0x80B5" auftritt.

Hinweis: Diese Funktionen werden in einem normalen PROFIenergy-fähigen IO-Device, z. B. der ET 200S, von der Firmware erledigt.

Zur Verwendung der Anweisung sind keine expliziten Kenntnisse des PROFIenergy-Standards erforderlich.

PROFIenergy-Hilfsbausteine (PE-Hilfsbausteine)

Über die PE-Hilfsbausteine generieren Sie das Antworttelegramm. Hierzu geben Sie die Antwortdaten (im Klartext) an den Eingangsparametern des jeweiligen Bausteins an.

- Für jedes PROFIenergy-Kommando gibt es einen entsprechenden Hilfsbaustein für eine positive Antwort:
  - PE-Kommando "Start_Pause": PE_Start_RSP (Seite 2508)
  - PE-Kommando "End_Pause": PE_End_RSP (Seite 2510)
  - PE-Kommando "Query_modes" - "List_Energy_Saving_Modes": PE_List_Modes_RSP (Seite 2511)
  - PE-Kommando "Query_modes" - "Get_Mode": PE_Get_Mode_RSP (Seite 2512)
  - PE-Kommando "PEM_Status": PE_PEM_Status_RSP (Seite 2514)
  - PE-Kommando "PE_Identify": PE_Identify_RSP (Seite 2516)
  - PE-Kommando "Query_Measurement" - "Get_Measurement_list": PE_Measurement_List_RSP (Seite 2518)
  - PE-Kommando "Query_Measurement" - "Get_Measurement_values": PE_Measurement_Value_RSP (Seite 2519)
- Unabhängig vom PROFIenergy-Kommando gibt es zusätzlich einen gemeinsamen Hilfsbaustein für eine negative Antwort (siehe PE_Error_RSP (Seite 2507)).

Verschaltung der Hilfsbausteine

Die Anweisung "PE_I_DEV" und die Hilfsbausteine sind aufeinander abgestimmt. Die Parameter werden teilweise einfach verschaltet. Die folgende Grafik zeigt, welche der Parameter verschaltet werden müssen.

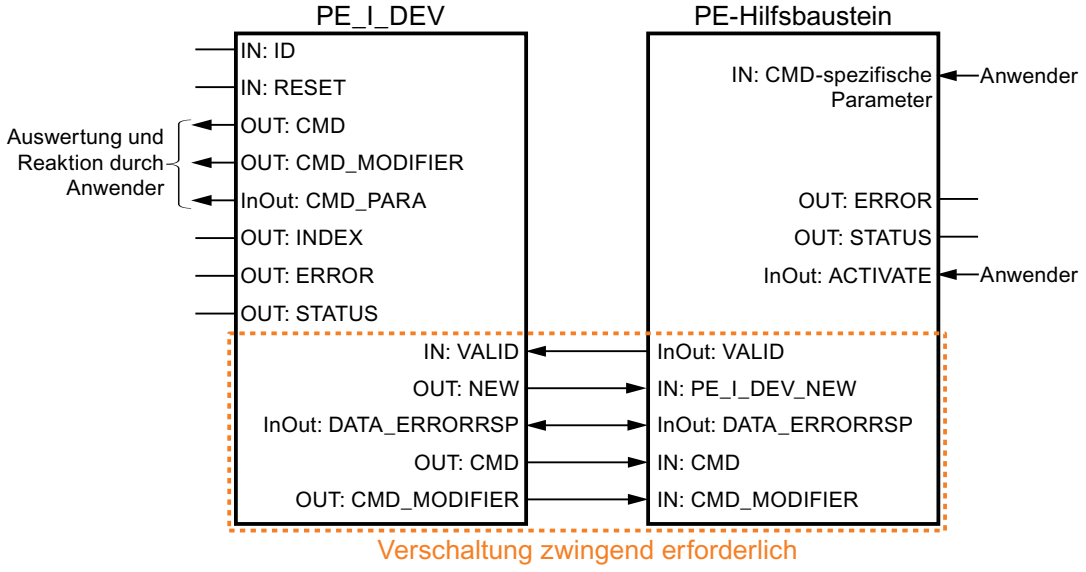

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_I_DEV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L oder Konstante	Rücksetzten der Anweisung.
ID	Input	HW_SUBMO-DULE	E, A, M, D, L oder Konstante	Adresse des Transferbereichs, welcher im IO-Controller mit den Daten für PROFIenergy versorgt wird. Die Hardware-Kennung entnehmen Sie den Systemkonstanten.
VALID	Input	BOOL	E, A, M, D, L oder Konstante	Die Antwortdaten für den PROFIenergy-Controller stehen bereit und können gesendet werden.
CMD_PARA	Output	VARIANT	E, A, M, D, L	Parameter für: • Get mode: PE_mode_ID • Get measurement values: Liste der Measurement_IDs (Liste der IDs der zu lesenden Variablen; es können entweder eine einzelne Variable oder auch mehrere Variablen auf einmal gelesen werden.) Maximale Länge: 234 Byte Hinweis: Bei S7-1200 darf der Zeiger nicht auf 64-Bit-Datentypen verweisen (also nicht auf die Datentypen LWORD, LINT und ULINT).
DATA_ER-RORRSP	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Datenbereich, der die Quittungsdaten für den PROFIenergy-Controller enthält. Dieser muss mit dem Zeiger übereinstimmen, der auch bei den Hilfsbausteinen verwendet wird. Hinweis: Bei S7-1200 darf der Zeiger nicht auf 64-Bit-Datentypen verweisen (also nicht auf die Datentypen LWORD, LINT und ULINT).
INDEX	Output	INT	E, A, M, D, L	Datensatznummer des PROFIenergy Records (0x80A0 fest)
CMD	Output	INT	E, A, M, D, L	Service-Request-ID des PROFIenergy-Kommandos entsprechend dem PROFIenergy-Profil (siehe "Parameter CMD und CMD_MODIFIER"). Nach Erweiterungen des PROFIenergy-Profils sind weitere PE- Kommandos (Service-Request-IDs) möglich.
CMD_ MODIFIER	Output	INT	E, A, M, D, L	PROFIenergy-Unterkommando: • Nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER". • Bei allen anderen Kommandos: "0". Nach Erweiterungen des PROFIenergy-Profils sind weitere Unterkommandos möglich.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NEW	Output	BOOL	E, A, M, D, L	Neue Daten vom PROFIenergy-Controller verfügbar.
ERROR	Output	BOOL	E, A, M, D, L	Befehl mit Fehler beendet.
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation (siehe "Parameter STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter CMD und CMD_MODIFIER

CMD	CMD_ MODIFIER	PROFIenergy-Kommando	Beschreibung
01	0	Start_Pause	Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus.
02	0	End_Pause	Beenden des Energiesparmodus.
03	1	Query_Modes - List energy saving Modes	Ausgeben der unterstützten Energiesparmodi.
	2	Query_Modes - Get Mode	Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.
04	0	PEM_Status	Status des Energiesparmodus abfragen.
05	0	PE_Identify	Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.
16	1	Query_Measurement - Get_Measurement_List	Auflistung der durch die der PE-Entity unterstützten Messwerte.
	2	Query_Measurement - Get_Measurement_Values	Ausgabe der Messwerte der der PE-Entity.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#**00**: Kein Fehler • B#16#**DE**: Fehler beim Lesen des Datensatzes • B#16#**DF**: Fehler beim Schreiben des Datensatzes • B#16#**C0**: Fehlermeldung durch die Anweisung "PE_I_DEV" oder der intern verwendeten Kommunikationsanweisungen "PRVREC (Seite 2440)" und "RCVREC (Seite 2438)".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • **80**: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch

Feldelement	Name	Bedeutung
STATUS[3]	Error_Code_1	Fehlerkennung (bei Error_Decode = **80**): • **B1**: Write length error (Fehler bei der Schreiblänge oder unzureichende Längenangabe über den Datentyp VARIANT).
STATUS[4]	Error_Code_2	Bei PROFINET-Fehler: Ausgabe der Fehlermeldung des IO-Controllers. Ist kein PROFINET-Fehler aufgetreten ist der Wert in STATUS[4] = "0".

Hinweis

Fehlermeldungen der Anweisungen "PRVREC" und "RCVREC"

Die Anweisung "PE_I_DEV" verwendet zur Kommunikation die Anweisungen "PRVREC (Seite 2440)" und "RCVREC (Seite 2438)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "PRVREC" und "RCVREC" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2370).

Siehe auch

WRREC: Datensatz schreiben (Seite 2326)

RDREC: Datensatz lesen (Seite 2306)

Beschreibung PROFIenergy (Seite 2464)

Hilfsbausteine der Anweisung PE_I_DEV

PE_Error_RSP: Negative Antwort auf Kommando generieren

Beschreibung

Der Hilfsbaustein "PE_Error_RSP" (Response with failure) generiert eine negative Antwort, falls das angeforderte Kommando generell oder temporär nicht unterstützt wird. Die Generierung der Antwort erfolgt unabhängig vom angeforderten Kommando.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Error_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
ERROR_CODE	Input	BYTE	E, A, M, D, L oder Konstante	Fehlernummer

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Start_RSP: Antwort auf Kommando zum Pausenstart generieren

Beschreibung

Der Hilfsbaustein "PE_Start_RSP" (Start Pause) generiert die Antwort auf das PE-Kommando Start_Pause (Seite 2493). Die Anweisung gibt zurück, welchen Energiesparzustand das Gerät einnimmt (Parameter PE_MODE_ID).

Wenn Sie bei verschieden langen Pausen unterschiedlich reagieren, können Sie das in der Rückmeldung über den eingenommenen Energiesparzustand zurückmelden (PE_Mode_ID = 1 für eine kurze Pause, PE_Mode_ID = 2 für eine längere Pause usw.).

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Start_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
PE_MODE_ID	Input	BYTE	E, A, M, D, L oder Konstante	PE-Modus, den der Prozess einnimmt
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_End_RSP: Antwort auf Kommando zum Pausenende generieren

Beschreibung

Der Hilfsbaustein "PE_End_RSP" generiert die Antwort auf das PE-Kommando End_Pause (Seite 2493). Um von dem aktuellen Betriebszustand in den Betriebszustand "Ready_To_Operate" zu wechseln, wird als Antwort die benötigte Zeitdauer zurückgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_End_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
Time_to_Operate	Input	DWORD	E, A, M, D, L oder Konstante	Zeitdauer, um vom aktuellen Betriebszustand nach "Ready_To_Operate" zu wechseln.
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_List_Modes_RSP: Abgefragte Energiesparmodi als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_List_Modes_RSP" generiert die Antwort auf das PE-Kommando List_Energy_Saving_Modes (Seite 2494). Die generierte Antwort enthält die Anzahl und die IDs der unterstützten Energiesparzustände.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_List_Modes_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein wird bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Number_of_PE_Mode_IDs	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der unterstützen Energiesparzustände. Zulässige Werte: 1 bis 254
PE_MODE_ID	Input	VARIANT	E, A, M, D, L	Zeigt auf den Bereich, in dem die IDs der unterstützen Energiesparzustände (PE_Mode_ID) hinterlegt sind. Zulässiger Bereich: 1 bis 254.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Get_Mode_RSP: Abgefragte Energiedaten als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Get_Mode_RSP" generiert die Antwort auf das Kommando Get_Mode (Seite 2495). Die Antwort enthält die Zeiten und Leistungs- bzw. Energiedaten eines einzelnen Energiesparzustands.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Get_Mode_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
PE_Mode_ID	Input	BYTE	E, A, M, D, L oder Konstante	ID des momentan aktiven Energiesparmodus.
PE_Mode_Attributes	Input	BYTE	E, A, M, D, L oder Konstante	Zusatzinformation zum Energiesparmodus
Time_min_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Mindest-Pausendauer für diesen PE-Energiesparmodus. Diese bildet die Summe der drei Parameter: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay
Time_to_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Zeitdauer von der Flanke am Parameter START (siehe "PE_I_DEV (Seite 2503)") bis zum Erreichen des angeforderten PE-Energiesparmodus.
Time_to_Operate	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Einschaltdauer bis "PE_ready_to_operate". Der Parameter "Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximalwert sein oder durch die PE-Entity dynamisch berechnet werden.
Time_min_Lenght_of_stay	Input	DWORD	E, A, M, D, L oder Konstante	Minimale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben muss.
Time_max_Lenght_of_stay	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben kann.
Mode_Power_Consumption	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch im aktuellen PE-Modus in [kW].
Energy_Consum_to_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch von "PE_ready_to_operate" bis zum aktuellen PE –Modus in [kWh].
Energy_Consum_to_operate	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_PEM_Status_RSP: PEM-Status als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_PEM_Status_RSP" generiert die Antwort auf das Kommando PEM_Status (Seite 2497).

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_PEM_Status_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
PE_MODE_ID_ Source	Input	BYTE	E, A, M, D, L oder Konstante	Source und Destination für PEM_STATUS. Werte: • 0x00: PE_POWER_OFF • 0x01 – 0xFE: Herstellerspezifisch • 0xFF: PE_READY_TO_OPERATE
PE_MODE_ID_ Destination	Input	BYTE	E, A, M, D, L oder Konstante	
Time_to_Operate[1]	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Einschaltdauer bis "PE_ready_to_operate". "Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximal-Wert sein oder durch die PE-Entity dynamisch berechnet werden.
Remaining_time_ to_destination[1]	Input	DWORD	E, A, M, D, L oder Konstante	Optional: Verbleibende Zeit bis zum angeforderten PE-Modus. Dynamischer Wert oder statischer Maximal-Wert
Mode_Power_ Consumption[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch im aktuellen PE-Modus in [kW].
Energy_ Consumption_ to_Destination[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch bis zum angeforderten PE-Modus in [kWh].
Energy_ Consumption_ to_operate[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich
[1] Wenn die Zeitdauer unbegrenzt ist, kann der max. Wert "0xFFFFFFFF" angegeben werden. Wenn die Zeitdauer "Null" ist, kann "0x00" verwendet werden. [2] Wenn ein Energieverbrauchswert nicht definiert ist, kann "0,0"angegeben werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Identify_RSP: Unterstützte PROFIenergy-Kommandos als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Identify_RSP" generiert die Antwort auf das Kommando PE_Identify (Seite 2499). In der Antwort auf das Kommando geben Sie an, welche PROFIenergy-Kommandos unterstützt werden. Beachten Sie, dass PE_IDENTIFY selbst ein PE-Kommando ist und in der Antwort entsprechend mit angegeben wird.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Identify_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
Start_Pause	Input	BOOL	E, A, M, D, L oder Konstante	Jeweils ein Parameter für jedes der entsprechenden PROFIenergy-Kommandos: • 0: PE-Kommando wird nicht unterstützt • 1: PE-Kommando wird unterstützt
End_Pause	Input	BOOL	E, A, M, D, L oder Konstante	
Query_Modes	Input	BOOL	E, A, M, D, L oder Konstante	
PEM_Status	Input	BOOL	E, A, M, D, L oder Konstante	
PE_Identify	Input	BOOL	E, A, M, D, L oder Konstante	
Query_Measurement	Input	BOOL	E, A, M, D, L oder Konstante	
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Measurement_List_RSP: Liste der unterstützten Messwerte als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Measurement_List_RSP" generiert die Antwort auf das Kommando Get_measurement_list (Seite 2500). In der Antwort geben Sie an, welche Messwerte (Measurement-IDs) unterstützt werden.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Measurement_List_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Count	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der unterstützten Messwerte (Measurement-IDs)
Measurement_List	Input	VARIANT	D	Zeiger auf das Array mit den unterstützten Measurement_IDs. Zum Aufbau des Arrays gemäß PROFIenergy Profil siehe: PE-Kommando "Query_Measurement" - "Get_Measurement_list" (Seite 2500)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Measurement_Value_RSP: Angeforderte Messwerte als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Measurement_Value_RSP" generiert die Antwort auf das Kommando Get_measurement_values (Seite 2502). In der Antwort geben Sie die Werte der angeforderten Messwerte zurück.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Measurement_Value_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn am Parameter der Wert "1" ansteht, wird der Hilfsbaustein bearbeitet.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Count	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der Messwerte (Measurement_Values).
Measurement_Values	Input	VARIANT	D	Zeiger auf das Array mit den Messwerten (Measurement_IDs). Zum Aufbau des Arrays gemäß PROFIenergy Profil, siehe PE-Kommando "Query_Measurement" - "Get_Measurement_values" (Seite 2502)
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2503)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2503)" verschaltet werden. Wenn die Antwortdaten für den PROFIenergy-Controller bereit stehen und gesendet werden können, wird der Parameter vom Hilfsbaustein gesetzt.
DATA_ERRORRSP	InOut	VARIANT	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2503)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

4.1.4.6 Baugruppenparametrierung

Datensätze schreiben und lesen

Prinzip

Es gibt Baugruppen, die über einen Systemdatenbereich verfügen, auf den Sie von Ihrem Programm aus nur schreibend zugreifen können. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt (siehe folgende Tabelle).

Darüber hinaus können Baugruppen auch einen Systemdatenbereich besitzen. Auf den Systemdatenbereich können Sie aus Ihrem Programm nur lesend zugreifen. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt.

Hinweis

Es gibt Baugruppen, die über beide Systemdatenbereiche verfügen. Dabei handelt es sich um physikalisch unterschiedliche Bereiche, die lediglich die logische Aufteilung in Datensätze gemeinsam haben.

Nur beschreibbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur beschreibbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie geschrieben werden können.

Datensatz-Nummer	Inhalt	Größe	Beschreibbar mit Anweisung
0	Parameter	-	WR_DPARM (Seite 2532)
1	Parameter	-	WR_DPARM (Seite 2532)
2 bis 127	Anwenderdaten	je ≤ 240 Bytes	WR_DPARM (Seite 2532) WR_REC (Seite 2433)
128 bis 240	Parameter	je ≤ 240 Bytes	WR_DPARM (Seite 2532) WR_REC (Seite 2433)

Nur lesbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur lesbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie gelesen werden können.

Datensatz-Nummer	Inhalt	Größe	Lesbar mit Anweisung
0	baugruppenspezifische Diagnosedaten (systemeinheitlich festgelegt)	4 Bytes	RD_REC (Seite 2428)
1	kanalspezifische Diagnosedaten (incl. Datensatz 0)	4 bis 220 Bytes	RD_REC (Seite 2428)
2 bis 127	Anwenderdaten	je ≤ 240 Bytes	RD_REC (Seite 2428)
128 bis 240	Diagnosedaten	je ≤ 240 Bytes	RD_REC (Seite 2428)

Systemressourcen

Wenn Sie mehrere asynchron laufende Datensatzübertragungen kurz nacheinander anstoßen, so wird gewährleistet, dass alle Aufträge durchgeführt werden und keine gegenseitige Beeinflussung stattfindet.

Wenn die Begrenzung der Systemressourcen erreicht wird, so erhalten Sie dies in RET_VAL mitgeteilt. Beheben Sie den temporären Fehlerfall durch eine Wiederholung des Auftrags.

Die maximale Anzahl "gleichzeitig" aktiver Aufträge eines Anweisungstyps ist CPU-abhängig.

RD_DPAR: Datensatz einer Baugruppe lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer INDEX der adressierten Komponente aus den projektierten Systemdaten. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.

Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Hinweis

Informationen zu den Datensätzen

Die Informationen zu den Datensatz-Nummern und zur Datensatz-Struktur entnehmen Sie dem Gerätehandbuch der jeweiligen Baugruppe.

Arbeitsweise

Die Anweisung "RD_DPAR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "RD_DPAR" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die Bytes 2 und 3 des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die Bytes 2 und 3 von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat, ist die Datensatzübertragung abgeschlossen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPAR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe (z. B. Eingabebaugruppe). Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Baugruppe unter "HW-Kennung" (z. B. "Eingang 0 -15 > HW-Kennung") hinterlegt.
INDEX	Input	INT	E, A, M, D, L oder Konstante	Datensatznummer
RECORD	InOut	VARIANT	E, A, M, D, L	Zielbereich für den gelesenen Datensatz
VALID	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde empfangen und ist gültig
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Auftrag ist noch nicht beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Lesevorgang trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	• Aufrufkennung (Bytes 2 und 3) bzw. Fehlercode • Byte 1: B#16#00, falls kein Fehler. Andernfalls Funktionskennung aus DPV1-PDU: Im Fehlerfall bei Datensatz lesen B#16#DE, im Fehlerfall bei Datensatz schreiben B#16#DF. Falls kein DPV1-Protokollelement benutzt wird: B#16#C0. • Byte 4: herstellerspezifische Erweiterung der Fehlerkennung
LEN	Output	INT	E, A, M, D, L	Länge der gelesenen Datensatzinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
0000	Kein Fehler	-
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".	-
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".	Dezentrale Peripherie
8092	Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.	-
8093	Angegebene Adresse am Parameter LADDR ist ungültig.	-
80B0	Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.	-
80B1	Die Länge des zu übertragenden Datensatzes ist falsch.	-
80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.	
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

In dem folgenden Beispiel lesen Sie den kompletten Parameterdatensatz einer digitalen Eingabebaugruppe.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein neun Variablen und die Struktur "record" an. (Den Aufbau der Struktur finden Sie weiter unten.)

SLI_gDB_RD_DPAR

		Name	Data type	Start value
1		Static		
2		start	Bool	false
3		laddr	HW_IO	257
4		dataRecNbr	Int	128
5		checkedValid	Bool	false
6		busy	Bool	false
7		error	Bool	false
8		status	DWord	16#0
9		lengthRec	Int	0
10		record	Struct	
11		Mode_C0_Bool	Array[0..7] of Bool	

Parameter verschalten: FC "SLI_itemiseByte_RD_DPAR"

Zur Weitergabe der Werte einer Bitfolge (0..7) legen Sie den FC "SLI_itemiseByte_RD_DPAR" an.

Darin legen Sie die folgenden lokalen Variablen an.

	Name	Data type
1	Input	
2	in0	Bool
3	in1	Bool
4	in2	Bool
5	in3	Bool
6	in4	Bool
7	in5	Bool
8	in6	Bool
9	in7	Bool
10	Output	
11	out0	Bool
12	out1	Bool
13	out2	Bool
14	out3	Bool
15	out4	Bool
16	out5	Bool
17	out6	Bool
18	out7	Bool

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) Verschaltungen an.

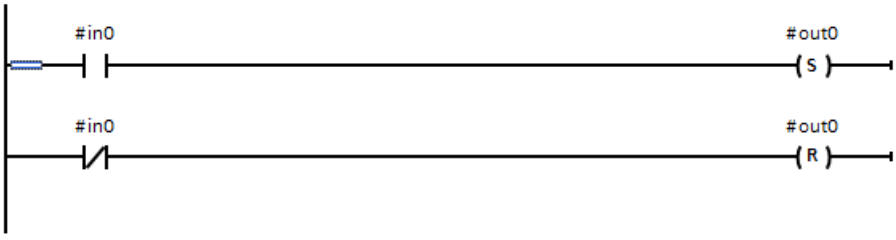

Parameter verschalten: FB "SLI_FB_RD_DPAR"

Im FB "SLI_FB_RD_DPAR" legen Sie folgende lokale Variablen an.

7	Static	
8	RD_DPAR_Instance	RD_DPAR

Netzwerk 1: Die Parameter der Anweisung "RD_DPAR" verschalten Sie wie folgt.

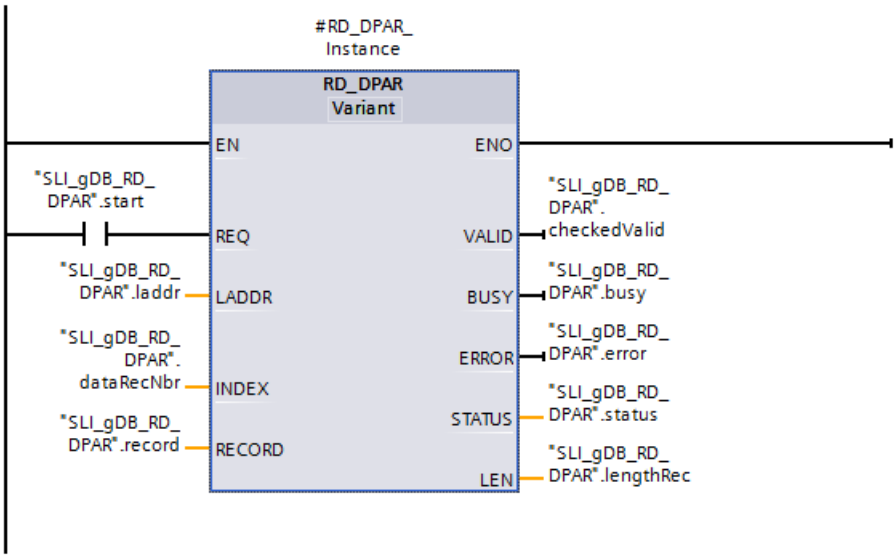

Netzwerk 2: Zur beispielhaften Umwandlung der Variable "Mode_Channel0" verschalten Sie die Variable wie folgt. Dabei werden alle Ein- und Ausgänge (0..7) des FC "SLI_itemiseByte_RD_DPAR" verschalten.

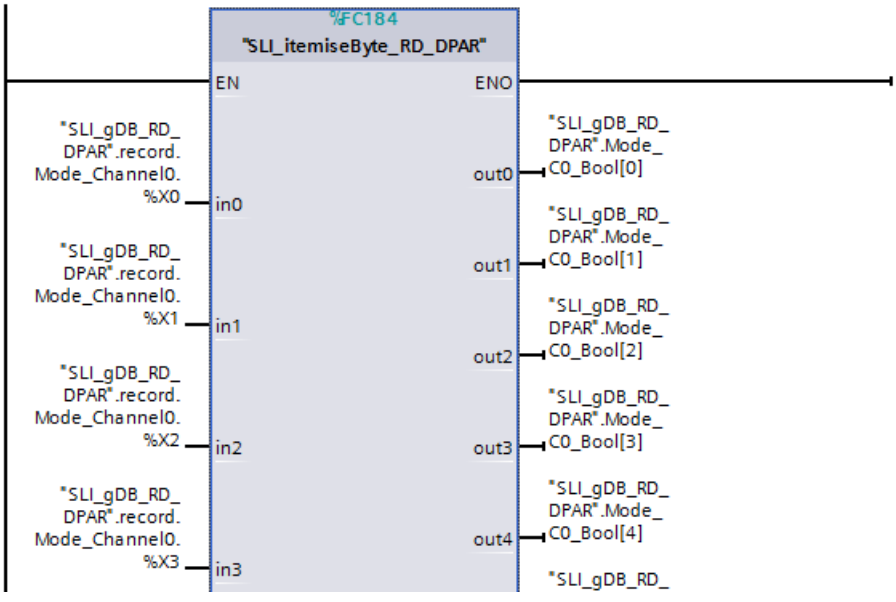

Hardware-Kennung

Sie legen eine Eingabebaugruppe an und lesen die Hardware-Kennung in den Eigenschaften der Eingabebaugruppe aus. Die Hardware-Kennung hinterlegen Sie über den Eingangsparameter LADDR ("laddr").

Ergebnis (FB "SLI_FB_RD_DPAR")

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "RD_DPAR" gestartet. Über den Eingangsparameter LADDR ("laddr") ruft die Anweisung "RD_DPAR" die Baugruppe auf. Über den Eingangsparameter INDEX ("dataRecNbr") wird der Datensatz der Nummer "128" der Baugruppe aufgerufen. Die

Anweisung "RD_DPAR" liesst während mehrerer Aufrufe den Datensatz aus und speichert den Datensatz über den Parameter RECORD ("record") ab.

Die Länge des gelesenen Datensatzes wird über den Ausgangsparameter LEN ("lengthRec") festgehalten. Entsprechend des Werts des Eingangsparameters MLEN ("maxLen" hat den Wert "0") ist die zu lesende Datensatzinformation unbeschränkt.

Während der Verarbeitung ist der Ausgangsparameter BUSY ("busy") auf "TRUE" und VALID ("checkedValid") auf "FALSE" gesetzt. Ein Erfolg ("checkedValid" auf "TRUE") beim Lesen wird erst zum Ende der Verarbeitung ausgegeben. Am Ausgangsparameter STATUS ("status") wird der Bausteinstatus angezeigt. Die Verarbeitung verläuft ohne Fehler.

SLI_gDB_RD_DPAR

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	laddr	HW_IO	257	16#0101
4	dataRecNbr	Int	128	128
5	checkedValid	Bool	false	TRUE
6	busy	Bool	false	FALSE
7	error	Bool	false	FALSE
8	status	DWord	16#0	16#0000_0000
9	lengthRec	Int	0	34
10	record	Struct		
11	Mode_C0_Bool	Array[0..7] of Bool		

Der Datensatz 128 enthält die Parameterinformationen über alle Kanäle der Baugruppe. Der Wert "2" der Variable "ByteOfDataRec" bezieht sich auf die Länge der Daten pro Kanal. Zu jedem Kanal gibt es Informationen zum Zustand und zum Betriebsmodus.

Auszug des gelesenen Datensatzes bis Kanal 1:

record	Struct		
Version	USInt	0	16
ByteOfDataRec	USInt	0	2
Diag_Channel0	Byte	16#0	16#00
Mode_Channel0	Byte	16#0	16#06
Diag_Channel1	Byte	16#0	16#00
Mode_Channel1	Byte	16#0	16#06

Zur näheren Erläuterung der Daten des Datensatz 128 wandeln Sie die BYTE-Werte in eine Bitfolge um. Hier ein Beispiel für den Betriebsmodus des Kanals 0: Da Bit 1 und 2 auf "TRUE" sind, besitzt der Kanal 0 eine Eingangsverzögerung von 3,2 ms. Die Erklärung der Struktur des Parameterdatensatzes bzw. des Kanals finden Sie im Gerätehandbuch der Baugruppe.

11	Mode_C0_Bool	Array[0..7] o...		
12	Mode_C0_Bool[0]	Bool	false	FALSE
13	Mode_C0_Bool[1]	Bool	false	TRUE
14	Mode_C0_Bool[2]	Bool	false	TRUE
15	Mode_C0_Bool[3]	Bool	false	FALSE
16	Mode_C0_Bool[4]	Bool	false	FALSE
17	Mode_C0_Bool[5]	Bool	false	FALSE
18	Mode_C0_Bool[6]	Bool	false	FALSE
19	Mode_C0_Bool[7]	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

RD_DPARM: Datensatz aus projektierten Systemdaten lesen (Seite 2530)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Übersicht über die Typen von Datensätzen (Seite 2298)

RD_DPARA: Datensatz einer Baugruppe asynchron lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM einer ausgewählten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD definierten Zielbereich eingetragen.

Hinweis

Informationen zu den Datensätzen

Die Informationen zu den Datensatz-Nummern und zur Datensatz-Struktur entnehmen Sie dem Gerätehandbuch der jeweiligen Baugruppe.

Arbeitsweise

Die Anweisung "RD_DPARA" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Lesevorgang, indem Sie die Anweisung mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPARA":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe (z. B. Eingabebaugruppe). Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Baugruppe unter "HW-Kennung" (z. B. "Eingang 0 -15 > HW-Kennung") hinterlegt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode. Falls bei der Übertragung kein Fehler auftrat, sind die folgenden Fälle zu unterscheiden: • RET_VAL enthält die Länge des tatsächlich gelesenen Datensatzes in Bytes, falls der Zielbereich größer ist als der gelesene Datensatz. • RET_VAL enthält "0", falls die Länge des gelesenen Datensatzes gleich der Länge des Zielbereichs ist.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Auftrag ist noch nicht beendet.
RECORD	Output	VARIANT	E, A, M, D, L	Zielbereich für den gelesenen Datensatz.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
0000	Kein Fehler	-
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.	-
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".	Dezentrale Peripherie
8092	Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.	-
8093	Angegebene Adresse am Parameter LADDR ist ungültig.	-
80B0	Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.	-
80B1	Die Länge des zu übertragenden Datensatzes ist falsch. Bei RD_DPARM (Seite 2530): Die Länge des durch RECORD aufgespannten Zielbereichs ist zu klein.	-
80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.	
allgemeine Fehlerinformationen	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RD_DPARM: Datensatz aus projektierten Systemdaten lesen

Nutzung

Hinweis

Nutzungsbeschränkung

Die Anweisung "RD_DPARM" ist nur zur Migration von S7-300/400-CPUs zu S7-1500-CPUs vorgesehen. In allen anderen Fällen benutzen Sie die Anweisung "RD_DPARA".

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM der adressierten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD aufgespannten Zielbereich eingetragen.

Hinweis

Informationen zu den Datensätzen

Die Informationen zu den Datensatz-Nummern und zur Datensatz-Struktur entnehmen Sie dem Gerätehandbuch der jeweiligen Baugruppe.

Arbeitsweise

Die Anweisung "RD_DPARM" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe.

Über den Ausgangsparameter RET_VAL wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Wenn es sich um eine Mischbaugruppe handelt, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe (z. B. Eingabebaugruppe). Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Baugruppe unter "HW-Kennung" (z. B. "Eingang 0 -15 > HW-Kennung") hinterlegt.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 0 bis 240)
RET_VAL	Return	INT	E, A, M, D, L	Länge des gelesenen Datensatzes in Bytes, falls der gelesene Datensatz in den Zielbereich passt und bei der Übertragung kein Fehler auftrat. Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
RECORD	Output	VARIANT	E, A, M, D, L	Zielbereich für den gelesenen Datensatz.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
0000	Kein Fehler	-
7001	Erstaufruf: Datenübertragung angestoßen	Dezentrale Peripherie
7002	Zwischenaufruf: Datenübertragung bereits aktiv	Dezentrale Peripherie
8090	Angegebene logische Basisadresse ungültig.	-
8092	In der VARIANT-Referenz am Parameter RECORD ist eine Typangabe ungleich BYTE angegeben.	-
8093	Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.	-
80B0	Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht	-
80B1	Die Länge des durch RECORD aufgespannten Zielbereichs ist zu klein.	-
80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.	-

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

WR_DPARM: Datensatz übertragen

Beschreibung

Mit der Anweisung "WR_DPARM" übertragen Sie den Datensatz mit der Nummer RECNUM aus den Projektierungsdaten zur adressierten Baugruppe. Der Datensatz kann statisch oder dynamisch sein.

Hinweis

Informationen zu den Datensätzen

Die Informationen zu den Datensatz-Nummern und zur Datensatz-Struktur entnehmen Sie dem Gerätehandbuch der jeweiligen Baugruppe.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_DPARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zu Schreiben
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe (z. B. Eingabebaugruppe). Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Baugruppe unter "HW-Kennung" (z. B. "Eingang 0 -15 > HW-Kennung") hinterlegt.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung	Einschränkung
0000	Kein Fehler	-
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.	-
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".	Dezentrale Peripherie
8093	Angegebene Adresse am Parameter LADDR ist ungültig.	-
80A1	Negative Quittung beim Senden des Datensatzes zur Baugruppe (Baugruppe während des Sendens gezogen oder Baugruppe defekt).	-
80A2	DP-Protokollfehler bei Layer 2, evtl. Hardware-/Schnittstellenfehler im DP-Slave.	Dezentrale Peripherie
80A3	DP-Protokollfehler bei User Interface/User.	Dezentrale Peripherie
80A9	Funktion wird nicht unterstützt.	-
80B0	Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.	-
80B1	Die Länge des zu übertragenden Datensatzes ist falsch.	-
80B2	Der projektierte Steckplatz ist nicht belegt.	-
80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.	-
80B6	Zugriff verweigert	-
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.	-
80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.	
80C4	Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte ihren Aufbau auf elektrische Störquellen.	-
80C5	Dezentrale Peripherie nicht verfügbar oder deaktiviert.	Dezentrale Peripherie
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund).	Dezentrale Peripherie
80D0	Für die Baugruppe ist kein Eintrag vorhanden.	-
80D1	Die Datensatznummer ist für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.).	-
80D2	Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.	-
80D5	Der Datensatz ist statisch.	-
allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)	-
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

4.1.4.7 Alarme

ATTACH: OB zu Alarmereignis zuordnen

Beschreibung

Mit der Anweisung "ATTACH" ordnen Sie einem Prozessalarmereignis einen Organisationsbaustein (OB) zu.

- Am Parameter OB_NR geben Sie die symbolische oder numerische Bezeichnung des Organisationsbausteins an, der dem am Parameter EVENT angegebenen Ereignis zugeordnet wird.
- An Parameter EVENT wählen sie das Prozessalarmereignis aus. Bereits erstellte Prozessalarmereignisse werden bei den PLC-Variablen unter "Systemkonstanten" aufgelistet.

Wenn das Ereignis am Parameter EVENT nach der fehlerfreien Ausführung der Anweisung "ATTACH" auftritt, wird der Organisationsbaustein am Parameter OB_NR aufgerufen und dessen Programm ausgeführt.

Mit dem Parameter ADD legen Sie fest, ob bisherige Zuordnungen des Organisationsbausteins zu anderen Ereignissen aufgehoben oder beibehalten werden. Bei einem Wert "0" am Parameter ADD werden die vorhandenen Zuordnungen durch die aktuelle Zuordnung ersetzt.

Prozessalarmereignisse

Wenn Ereignisse nicht lang genug anstehen, können Prozessalarme verwendet werden. Sie reagieren auf diese zur Laufzeit. Den Prozessalarmen können jeweils Prozessalarm-OBs zugeordnet werden, welche die Reaktion auf ein bestimmtes Ereignis enthalten.

Prozessalarme können für verschieden Ereignisse erstellt werden. Beispiele hierfür sind:

- Bei Digitaleigängen die Erkennung steigender oder fallender Flanken.
- Bei Analogeingängen die Unter- und Überschreitung eines definierten Grenzwertes.
- Bei schnellen Zählern bei einer Richtungsumkehr, externem Rücksetzen, Überlauf/ Unterlaut, usw.

Funktionsweise

Jeder Prozessalarm kann einem Prozessalarm-OB zugeordnet werden, der bei Auftreten des Prozessalarmereignisses in die Warteschlange für die Bearbeitung gestellt wird. Die Zuordnung von OB und Ereignis kann während der Konfiguration oder zur Laufzeit erfolgen:

- Um während der Konfiguration einem Ereignis einen OB zuzuordnen, wählen Sie in der Hardware-Konfiguration unter "Prozessalarme" für ein Ereignis einen Prozessalarm-OB aus.
- Um die Zuordnung während der Laufzeit durchzuführen, verwenden Sie die Anweisung ATTACH. Die Zuordnung von Ereignis und Prozessalarm-OB erfolgt über die Parameter EVENT und OB_NR.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATTACH":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_ATT	E, A, M, D, L oder Konstante	Organisationsbaustein (Nummern bis 32767 werden unterstützt.)
EVENT	Input	EVENT_ATT	E, A, M, D, L oder Konstante	Prozessalarmereignis, das dem OB zugeordnet werden soll. Das Prozessalarmereignis muss zuerst in der Hardware-Gerätekonfiguration für Eingänge oder schnelle Zähler aktiviert werden.
ADD	Input	BOOL	E, A, M, D, L oder Konstante	Auswirkungen auf die bisherigen Zuordnungen: • ADD=0 (Vorbelegung): Dieses Ereignis ersetzt alle bisherigen Ereigniszuordnungen für diesen OB. • ADD=1: Dieses Ereignis wird den bisherigen Ereigniszuordnungen für diesen OB hinzugefügt.
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0	Kein Fehler
8090	OB ist nicht vorhanden
8091	OB hat den falschen Typ
8093	Ereignis ist nicht vorhanden
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ordnen Sie einem erzeugten Prozessalarmereignis einen HW Interupt-OB (OB40) zu.

Voraussetzung:

Sie legen eine Eingabebaugruppe (DI) an und projektieren unter "Eigenschaften > Eingange 0-15 > ... > Kanal 0" ein Prozessalarmereignis fur eine steigende Flanke.

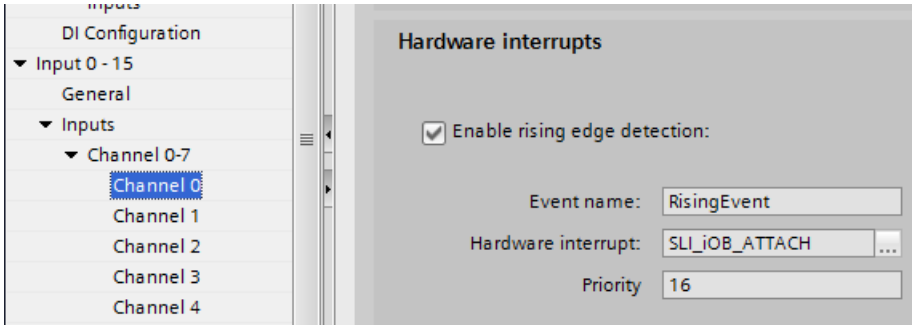

Am Eingangsparameter EVENT ("event") wird die folgende Hardware-ID des projektierten Prozessalarmereignisses hinterlegt.

General | IO tags | System constants | Texts

Name	Type	Hardware identifier
RisingEvent	Event_HwInt	16#C0000101
Local~DI_16x24VDC_HF_1	Hw_SubModule	257

Datenablage

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein neun Variablen an. Uber den Eingangsparameter OB_NR ("obNbr") der Anweisung "ATTACH" hinterlegen Sie die Nummer des zuzuordnenden OBs (OB40).

SLI_gDB_ATTACH

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	obNbr	OB_ATT	40
4	add	Bool	0
5	event	EVENT_ATT	16#C0000101
6	returnValue	Int	0
7	testValue1	Int	123
8	testValue2	Int	0
9	signal	Bool	0
10	scannedSignal	Bool	false

Für den Aufruf des Prozessalarmereignisses legen Sie eine Variablentabelle mit einer PLC-Variable (Adresse "I0.0") an.

SLI_Var_ATTACH

	Name	Data type	Address
1	SLI_ATTACH_force	Bool	%I0.0

Verschaltung

Die Parameter der Anweisung verschalten Sie wie folgt.

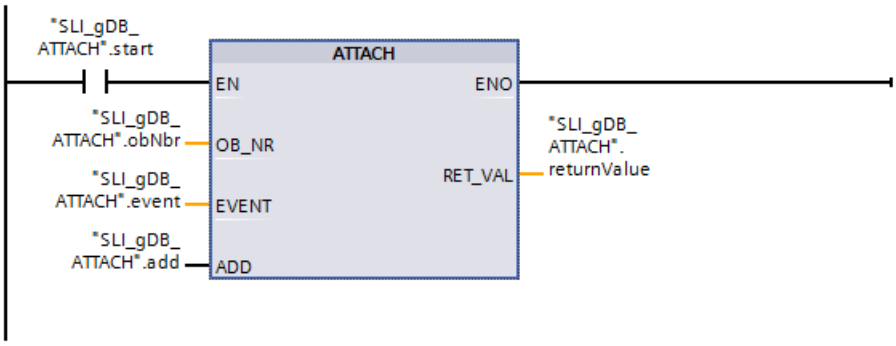

Für den Aufruf des Prozessalarmereignisses verschalten Sie die PLC-Variable wie folgt.

Netzwerk 2:

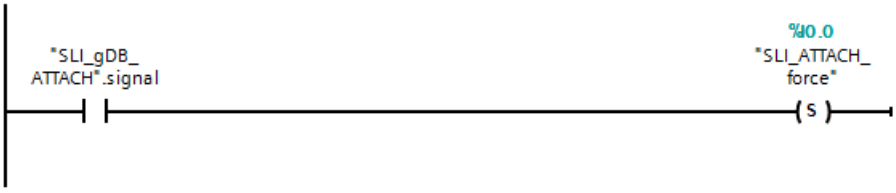

Netzwerk 3:

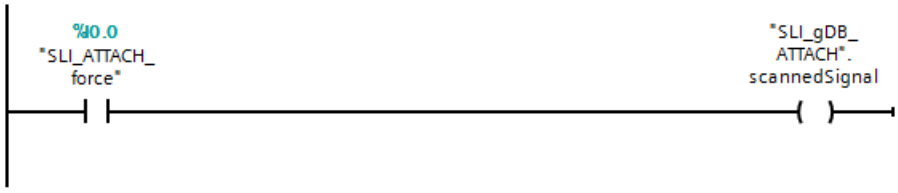

Zum Rücksetzen verschalten Sie "testValue2" und "SLI_ATTACH_force" wie folgt.

Netzwerk 4:

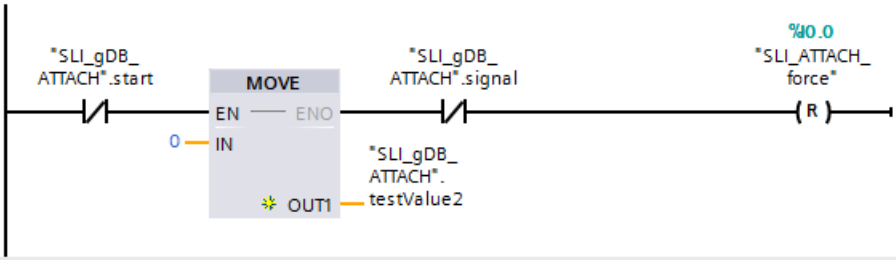

Im OB40 legen Sie die folgende Verschaltung an:

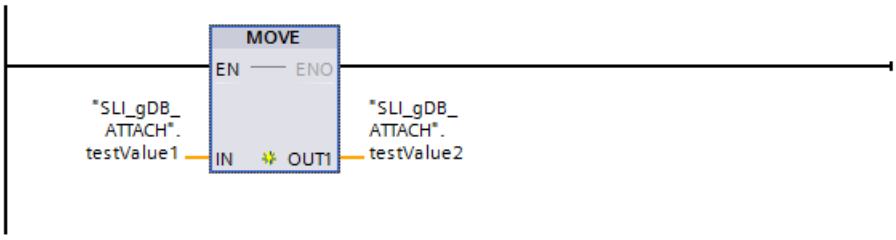

Verarbeitung

Wenn der Schließerkontakt ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "ATTACH" ausgefuhrt und das Ereignis dem OB zugeordnet. Durch den Wert "FALSE" des Eingangsparameters ADD ("add") ersetzt das Ereignis alle bisherigen Ereigniszuordnungen fur den OB40. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Wenn im Netzwerk 2 der Schließerkontakt ("signal") den Signalzustand "TRUE" liefert, wird die PLC-Variable "SLI_ATTACH_force" in den Signalzustand "TRUE" gesetzt und der OB40 ausgefuhrt. Im OB40 wird während dessen Aufrufs der Wert von "testValue1" an "testValue2" übergeben.

Wenn die Schließerkontakte ("start" und "signal") den Signalzustand "FALSE" liefern, werden im Netzwerk 4 die Werte von "testValue2" und "SLI_ATTACH_force" rückgesetzt.

SLI_gDB_ATTACH

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	obNbr	OB_ATT	40	40
4	add	Bool	0	FALSE
5	event	EVENT_ATT	16#C0000101	16#C000_0101
6	returnValue	Int	0	0
7	testValue1	Int	123	123
8	testValue2	Int	0	123
9	signal	Bool	0	TRUE
10	scannedSignal	Bool	false	TRUE

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DETACH: OB vom Alarmereignis trennen

Beschreibung

Mit der Anweisung machen Sie zur Laufzeit die bestehende Zuordnung eines Organisationsbausteins zu einem oder mehreren Prozessalarmereignissen rückgängig.

Am Parameter OB_NR geben Sie die symbolische oder numerische Bezeichnung des Organisationsbausteins ein. Die Zuordnung zu dem am Parameter EVENT angegebenen Ereignis wird rückgängig gemacht.

- Wenn Sie am Parameter EVENT ein einzelnes Prozessalarmereignis ausgewählt haben, wird die Zuordnung des OBs zu diesem Prozessalarmereignis rückgängig gemacht. Alle anderen momentan bestehenden Zuordnungen bleiben erhalten. Ein einzelnes Prozessalarmereignis können Sie über die Klappliste des Operandenplatzhalters auswählen.
- Wenn Sie kein Prozessalarmereignis ausgewählt haben, werden alle derzeit zu diesem Organisationsbaustein OB_NR zugeordneten Ereignisse getrennt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DETACH":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_ATT	E, A, M, D, L oder Konstante	Organisationsbaustein (Nummern bis 32767 werden unterstützt.)
EVENT	Input	EVENT_ATT	E, A, M, D, L oder Konstante	Prozessalarmereignis
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0	Kein Fehler
1	Keine Zuordnung vorhanden (Warnung)
8090	OB ist nicht vorhanden
8091	OB ist vom falschen Typ
8093	Ereignis nicht vorhanden
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ordnen Sie ein Prozessalarmereignis einem HW Interupt-OB (OB41) zu und lösen die Zuordnung wieder.

Voraussetzung:

Ein HW Interupt-OB (OB41 mit dem Namen "SLI_iOB_DETACH") ist bereits angelegt. Sie legen eine Eingabebaugruppe (DI) an und projektieren unter "Eigenschaften > Eingange 0-15 > ... > Kanal 1" ein Prozessalarmereignis fur eine steigende Flanke.

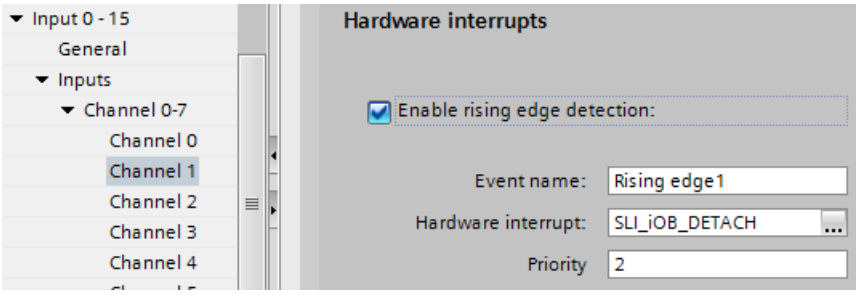

Am Eingangsparameter EVENT ("event") wird die folgende Hardware-ID des projektierten Prozessalarmereignisses ("Rising edge 1") hinterlegt.

DI 16x24VDC HF_1 [DI 16x24VDC HF]

General | IO tags | System constants | Texts

Name	Type	Hardware identi.
Rising edge0	Event_HwInt	16#C0000101
Rising edge1	Event_HwInt	16#C0010101
Local~DI_16x24VDC_HF_1	Hw_SubModule	257

Datenablage:

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zehn Variablen an. Uber den Eingangsparameter OB_NR ("obNbr") der Anweisungen "ATTACH" und "DETACH" hinterlegen Sie die Nummer des zuzuordnenden OBs (OB41).

SLI_gDB_DETACH

	Name	Data type	Start value
1	Static		
2	startAttach	Bool	false
3	add	Bool	0
4	retValAtt	Int	0
5	obNbr	OB_ATT	41
6	event	EVENT_ATT	16#C0010101
7	retValDet	Int	0
8	testValue1	Int	125
9	testValue2	Int	0
10	signal	Bool	0
11	scannedSignal	Bool	false

Für den Aufruf des Prozessalarmereignisses legen Sie eine Variablentabelle mit einer PLC-Variable (Adresse "I1.0") an.

SLI_Var_DETACH

	Name	Data type	Address
1	SLI_DETACH_event	Bool	%I1.0

Verschaltung:

Zum Aufruf der Anweisungen erstellen Sie einen FB ("SLI_FB_DETACH").

Netzwerk 1: Die Parameter der Anweisung "ATTACH" verschalten Sie wie folgt.

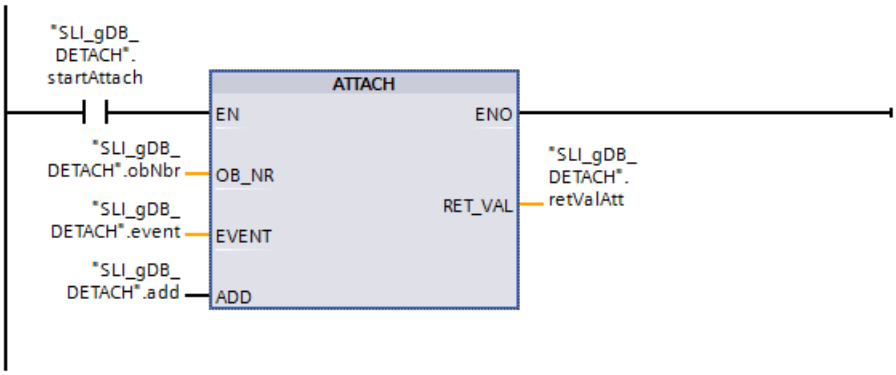

Netzwerk 2: Für den Aufruf des Prozessalarmereignisses verschalten Sie die PLC-Variable wie folgt.

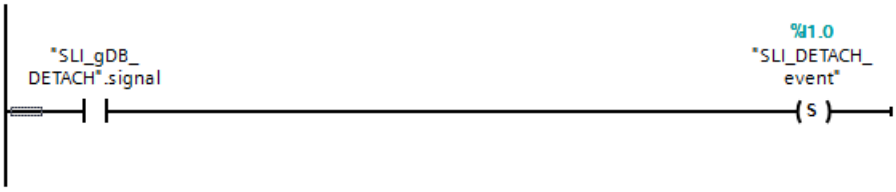

Netzwerk 3: Den Aufruf des Prozessalarmereignisses geben Sie wie folgt weiter.

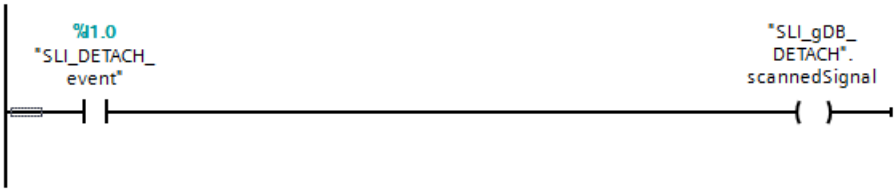

Netzwerk 4: Die Parameter der Anweisung "DETACH" verschalten Sie wie folgt.

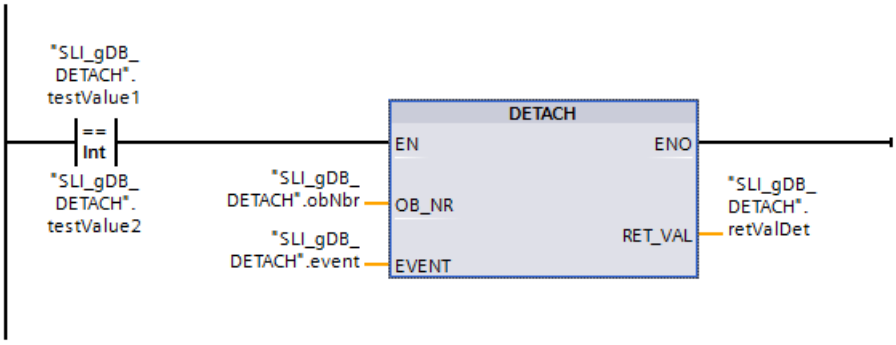

Netzwerk 5: Zum Rücksetzen verschalten Sie "testValue2" und "SLI_DETACH_event" wie folgt.

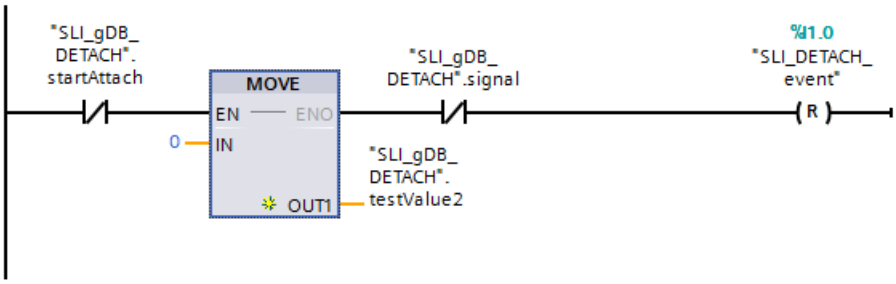

Im OB41 legen Sie die folgende Verschaltung an:

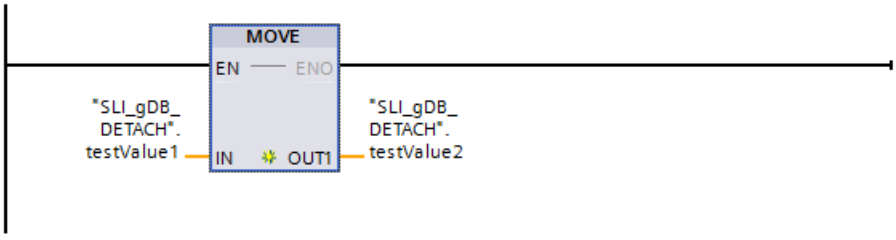

Verarbeitung

Wenn der Schließerkontakt ("startAttach") den Signalzustand "TRUE" liefert, wird die Anweisung "ATTACH" ausgefuhrt und das Ereignis dem OB41 zugeordnet. Durch den Wert "FALSE" des Eingangsparameters ADD ("add") ersetzt das Ereignis alle bisherigen Ereigniszuordnungen fur den OB41. Am Ausgangsparameter RET_VAL ("retValAtt") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

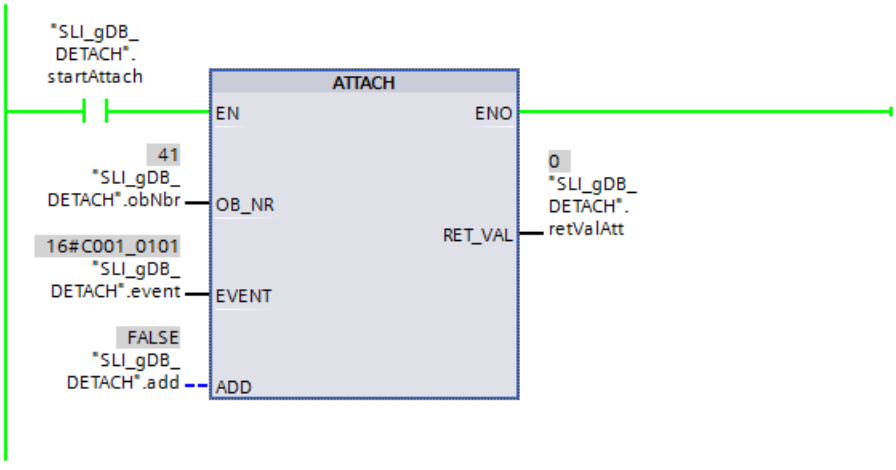

Wenn im Netzwerk 2 der Schließerkontakt ("signal") den Signalzustand "TRUE" liefert, wird das Prozessalarmereignis ("SLI_DETACH_event") auf "TRUE" gesetzt. Der OB41 wird aufgerufen und im OB41 wird der Wert von "testValue1" an "testValue2" übergeben. Solange das Prozessalarmereignis den Signalzustand "TRUE" liefert, wird der OB41 aufgerufen und im Netzwerk 3 durch die Spule ("scannedSignal") angezeigt.

SLI_gDB_DETACH				
	Name	Data type	Start value	Monitor value
1	Static			
2	startAttach	Bool	false	TRUE
3	add	Bool	0	FALSE
4	retValAtt	Int	0	0
5	obNbr	OB_ATT	41	41
6	event	EVENT_ATT	16#C0010101	16#C001_0101
7	retValDet	Int	0	0
8	testValue1	Int	125	125
9	testValue2	Int	0	125
10	signal	Bool	0	TRUE
11	scannedSignal	Bool	false	TRUE

Wenn "testValue1" und "testValue2" den gleichen Wert liefern, wird die Anweisung "DETACH" ausgefuhrt und das Ereignis vom OB41 getrennt. Am Ausgangsparameter RET_VAL ("retValDet") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

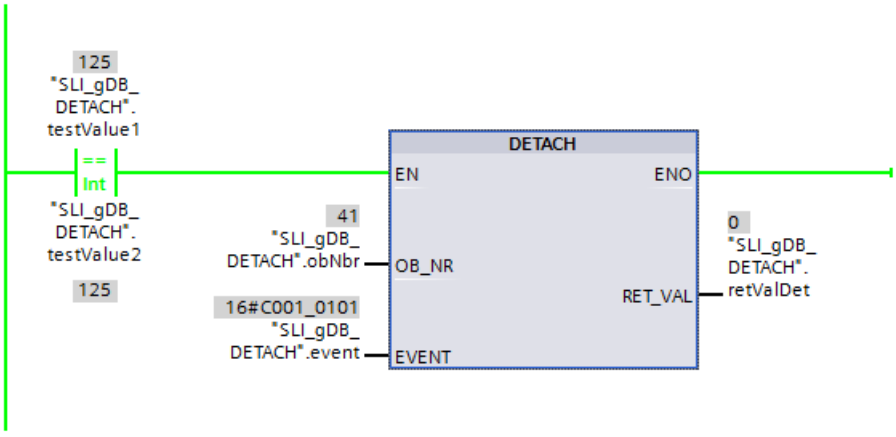

Wenn die Variable "startAttach" den Signalzustand "FALSE" liefert, werden im Netzwerk 5 die Werte von "testValue2" und der PLC-Variable zurückgesetzt.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier:Sample Library for Instructions (Seite 615)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Weckalarm

SET_CINT: Weckalarm-Parameter setzen

Beschreibung

Mit der Anweisung setzen Sie die Parameter für einen Weckalarm-OB. Der Startzeitpunkt für einen Weckalarm-OB ergibt sich aus dem jeweiligen Zeitintervall des OBs und der Phasenverschiebung.

- Der Zeitintervall eines OBs ist der Zeitabstand, in dem der OB regelmäßig aufgerufen wird. Wenn das Zeitintervall z. B. 100µs beträgt, wird der OB während der Programmausführung alle 100µs aufgerufen.
- Die Phasenverschiebung ist ein Zeitintervall, um das der Aufruf eines Weckalarm-OBs verschoben wird. Um niederpriore Organisationsbausteine in einem genauen Zeitraster zu bearbeiten, können Sie die Phasenverschiebung verwenden.

Wenn der OB nicht existiert oder der verwendete Zeitintervall nicht unterstützt wird, wird eine entsprechende Fehlermeldung am Parameter RET_VAL ausgegeben.

Ein Zeitintervall am Parameter CYCLE von "0" bedeutet, dass der OB nicht aufgerufen wird.

Arbeitsweise

Werden ein niederpriorer OB und ein höherpriorer OB im gleichen Zeitintervall aufgerufen, wird der niederpriore OB erst aufgerufen, nachdem der höherpriore OB abgearbeitet wurde. Der Aufrufzeitpunkt für den niederprioren OB kann sich je nach Länge des Bearbeitungszeitraums des höherprioren OBs verschieben.

OB-Aufruf ohne Phasenverschiebung

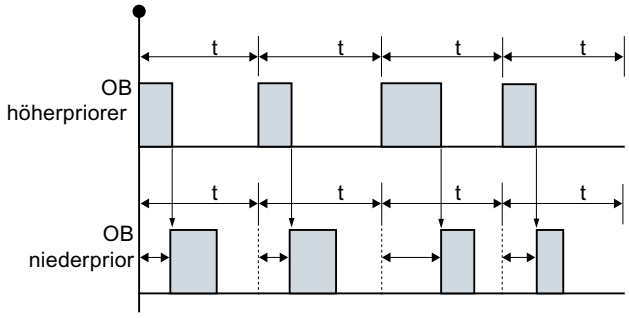

Wenn für den niederprioren OB eine Phasenverschiebung projektiert wird und die Phasenverschiebung größer als die jeweils aktuelle Bearbeitungszeit des höherprioren OBs ist, wird der Baustein in einem festen Zeitraster aufgerufen.

OB-Aufruf mit Phasenverschiebung

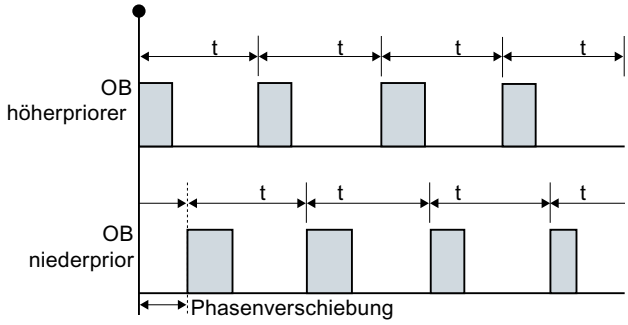

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_CINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_CYCLIC	E, A, M, D, L oder Konstante	OB-Nummer (<32768)
CYCLE	Input	UDINT	E, A, M, D, L oder Konstante	Zeitintervall in Mikrosekunden
PHASE	Input	UDINT	E, A, M, D, L oder Konstante	Phasenverschiebung in Mikrosekunden
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#....)	Beschreibung
0	Kein Fehler
8090	OB ist nicht vorhanden oder OB ist vom falschen Typ
8091	Falscher Zeitintervall
8092	Falsche Phasenverschiebung
80B2	Kein Ereignis zu OB zugeordnet
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Weckalarm-Funktionen (Seite 2547).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

QRY_CINT: Weckalarm-Parameter abfragen

Beschreibung

Die Anweisung QRY_CINT gibt die folgenden Parameter eines Weckalarm-OBs aus:

- Zeittakt
- Phasenverschiebung
- Status des Weckalarms (freigegeben, verzögert, abgelaufen, ...)

Der Weckalarm-OB wird über den Parameter OB_NR identifiziert.

Die Werte der abgefragten Parameter des Weckalarms entsprechen denen zum Zeitpunkt der Ausführung der Anweisung "QRY_CINT".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY_CINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_CYCLIC (INT)	E, A, M, D, L oder Konstante	OB-Nummer (<32768) oder symbolische Adressierung über den Namen des OBs (z. B. OB_MyOB)
CYCLE	Output	UDINT	E, A, M, D, L	Zeittakt in Microsekunden
PHASE	Output	UDINT	E, A, M, D, L	Phasenverschiebung
STATUS	Output	WORD	E, A, M, D, L	Status des Weckalarms: • Bit 0 bis Bit 4: siehe Parameter STATUS • Weitere Bits: immer "0"
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Bit	Wert	Bedeutung
0	0	Nicht verwendet (immer "0").
1	0	Der Weckalarm ist freigegeben.
	1	Der Weckalarm ist verzögert.
2	0	Der Weckalarm ist nicht aktiviert oder abgelaufen.
	1	Der Weckalarm ist aktiviert.
3	0	Nicht verwendet (immer "0").

Bit	Wert	Bedeutung
4	0	Ein OB mit der angegebenen Nummer existiert nicht.
	1	Ein OB mit der angegebenen Nummer existiert.
Weitere Bits		Nicht verwendet (immer "0").

Parameter RET_VAL

Wenn ein Fehler auftritt, wird am Parameter RET_VAL der entsprechende Fehlercode angezeigt und der Parameter STATUS auf "0" gesetzt.

Fehlercode* (W#16#....)	Beschreibung
0	Kein Fehler.
8090	OB ist nicht vorhanden oder OB ist vom falschen Typ.
80B2	Kein Ereignis zu OB zugeordnet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Weckalarm-Funktionen (Seite 2547).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Programmbeispiel zu Weckalarm-Funktionen

Einleitung

In dem folgenden Beispiel lesen Sie von einem Weckalarm-OB (OB31) die Phase und die Taktzeit aus und stellen eine neue Taktzeit ein.

Der OB31 besitzt die folgenden Voreinstellungen:

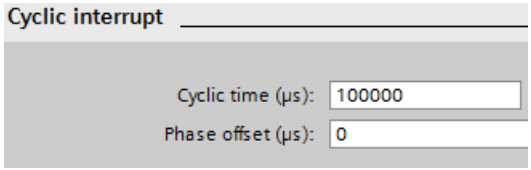

Voraussetzung

- Zum Aufrufen erstellen Sie einen OB31.
- Zur Ablage der Daten legen Sie in einem globalen Datenbaustein zwölf Variablen an.

SLI_gDB_SET_CINT

	Name	Data type	Start value
1	Static		
2	executeSet	Bool	false
3	obNbr	OB_CYCLIC	31
4	setCycle	UDInt	1000
5	setPhase	UDInt	0
6	returnValueSet	Int	0
7	testTime	Time	T#0ms
8	testTimeLimit	Time	T#2m
9	executeQry	Bool	false
10	returnValueQry	Int	0
11	itemiseStatus	Struct	
12	Cyclic_interrupt_enabled	Bool	false
13	Cyclic_interrupt_not_expired	Bool	false
14	OB_with_obNbr_exists	Bool	false
15	qryCycle	UDInt	0
16	qryPhase	UDInt	0

Parameter verschalten: FC "SLI_itemiseStatus_CINT"

Zum Auswerten der Statusinformationen legen Sie einen FC "SLI_itemiseStatus_CINT" an.

Darin legen Sie die folgende lokale Variable an.

SLI_itemiseStatus_CINT

	Name	Data type
1	Input	
2	inWord	Word

Für alle relevanten Bits legen Sie Verschaltungen an.

Netzwerk 1:

Netzwerk 2:

Netzwerk 3:

Parameter verschalten: FB "SLI_FB_SET_CINT"

Zum Aufruf der Anweisungen erstellen Sie einen FB ("SLI_FB_SET_CINT"). In dem FB "SLI_FB_SET_CINT" legen Sie die folgende lokale Variable an.

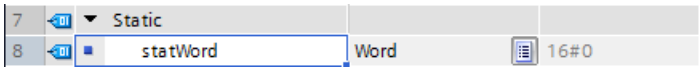

Netzwerk 1: Die Parameter der Anweisung "QRY_CINT" verschalten Sie wie folgt.

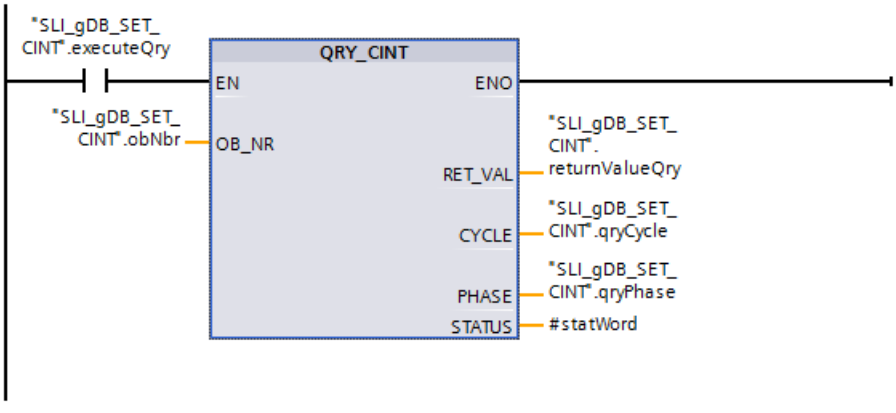

Netzwerk 2: Die Parameter der Anweisung "SET_CINT" verschalten Sie wie folgt.

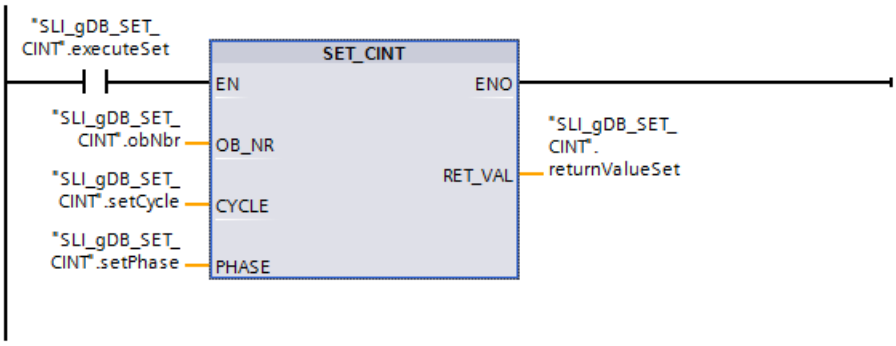

Netzwerk 3: Den Parameter des FC "SLI_itemiseStatus_CINT" verschalten Sie wie folgt.

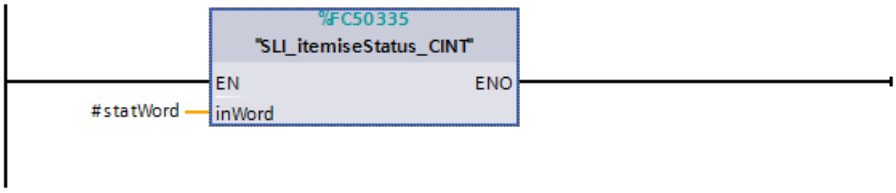

Parameter verschalten: OB31

Im OB31 legen Sie folgende Verschaltungen an.

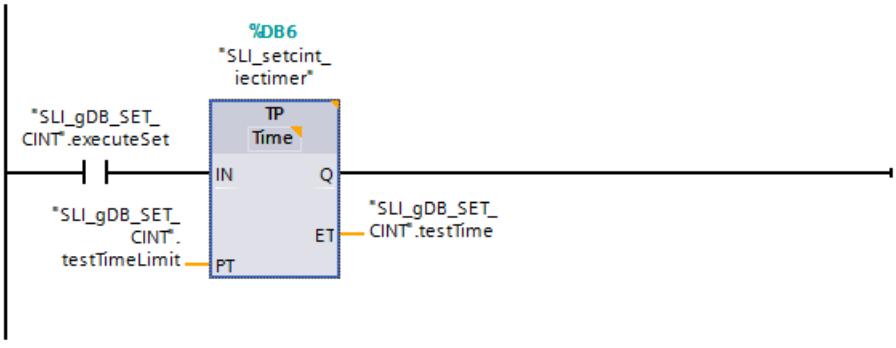

Ergebnis von "QRY_CINT"

Über den Eingangsparameter OB_NR ("obNbr") der Anweisungen "QRY_CINT" und "SET_CINT" ist die Nummer des aufzurufenden OBs (OB31) hinterlegt.

Wenn der Schließerkontakt ("executeQry") den Signalzustand "TRUE" liefert, wird die Anweisung "QRY_CINT" ausgefuhrt und die Daten des OB31 ausgelesen.

Am Ausgangsparameter CYCLE ("qryCycle") wird die Taktzeit ausgegeben. Am Ausgangsparameter PHASE ("qryPhase") wird die Phasenverschiebung ausgegeben. Der Ausgangsparameter STATUS ("#statWord") zeigt den Status des OB31 an. Am Ausgangsparameter RET_VAL ("returnValueQry") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft.

Die aktuellen Statusinformationen ("#statWord") des OB31 werden im Netzwerk 3 über den FC "SLI_itemiseStatus_CINT" interpretiert und über die Struktur "itemiseStatus" ausgegeben.

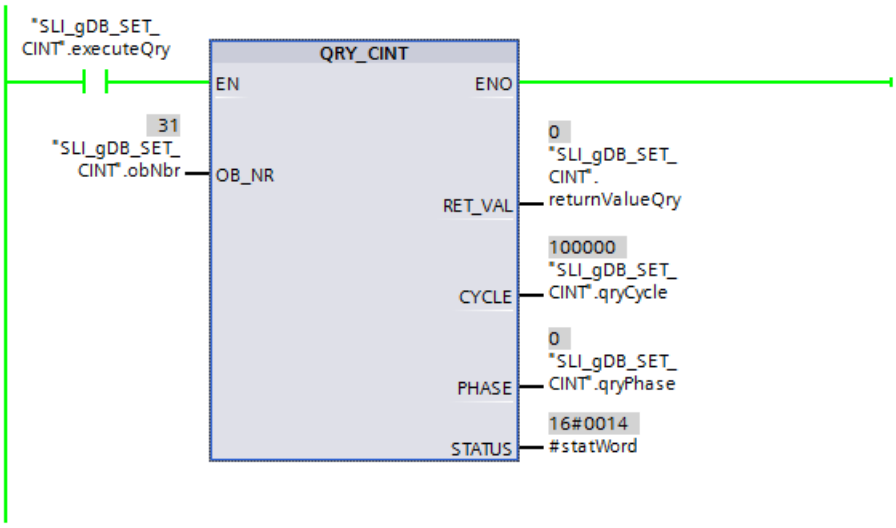

Der Wert der Variable "#statWord" wird in der Struktur "itemiseStatus" abgebildet:

12	itemiseStatus	Struct		
13	Cyclic_interrupt_enabled	Bool	false	TRUE
14	Cyclic_interrupt_not_expired	Bool	false	TRUE
15	OB_with_obNbr_exists	Bool	false	TRUE

Ergebnis von "SET_CINT"

Wenn der Schließerkontakt ("executeSet") den Signalzustand "TRUE" liefert, wird die Anweisung "SET_CINT" ausgefuhrt. Über den Eingangsparameter CYCLE ("setCycle") der Anweisung "SET_CINT" wird dem OB31 eine neue Taktzeit übergeben. Am Eingangsparameter PHASE ("setPhase") ist als Phasenverschiebung der Wert "0" hinterlegt.

Am Ausgangsparameter RET_VAL ("returnValueSet") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_SET_CINT

	Name	Data type	Start value	Monitor value
1	Static			
2	executeSet	Bool	false	TRUE
3	obNbr	OB_CYCLIC	31	31
4	setCycle	UDInt	1000	1000
5	setPhase	UDInt	0	0
6	returnValueSet	Int	0	0
7	testTime	Time	T#0ms	T#14S_487MS
8	testTimeLimit	Time	T#2m	T#2M
9	executeQry	Bool	false	TRUE
10	returnValueQry	Int	0	0
11	itemiseStatus	Struct		
12	Cyclic_interrupt_enabled	Bool	false	TRUE
13	Cyclic_interrupt_not_expired	Bool	false	TRUE
14	OB_with_obNbr_exists	Bool	false	TRUE
15	qryCycle	UDInt	0	1000
16	qryPhase	UDInt	0	0

Nach der Ausführung der Anweisung "SET_CINT" behält der OB31 die neuen Weckalarm-Daten bei bis zum STOPP der CPU. Sie können den Schließerkontakt ("executeSet") nach der Ausführung wieder auf den Signalzustand "FALSE" setzen. Die voreingestellten Eigenschaften des OB31 bleiben von der Ausführung des "SET_CINT" unberührt.

Ergebnis des OB31

Wenn der Schließerkontakt ("executeSet") den Signalzustand "TRUE" liefert, wird im OB31 ein IEC-Timer "TP" gestartet. Der IEC-Timer stoppt nach 2 Minuten.

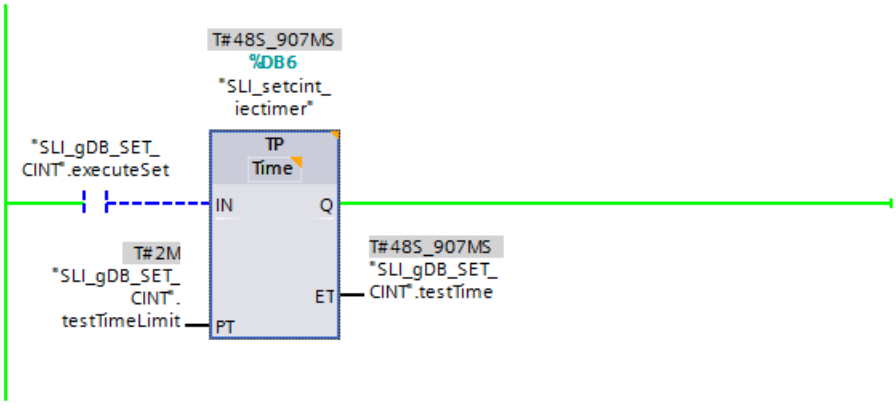

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Uhrzeitalarm

SET_TINT: Uhrzeitalarm einstellen

Beschreibung

Mit der Anweisung "SET_TINT" können Sie Startdatum und -uhrzeit der Uhrzeitalarm-Organisationsbausteine aus dem Anwenderprogramm heraus einstellen, ohne dass in der Hardware-Konfiguration Einstellungen vorgenommen wurden.

- An dem Parameter OB_NR geben Sie die Nummer des Uhrzeitalarm-OBs an, für den Sie Startdatum und -uhrzeit einstellen möchten.
- Über die Parameter SDT und PERIOD geben Sie an, wann und wie häufig der Uhrzeitalarm-OB aufgerufen werden soll:
  - Einmaliger Aufruf: Geben Sie am Parameter SDT das Datum und die Uhrzeit an. Verwenden Sie am Parameter PERIOD den Wert "0".
  - Mehrmaliger Aufruf: Geben Sie am Parameter SDT das Datum und die Uhrzeit des ersten Aufrufs an. Stellen Sie über den Parameter PERIOD ein, in welchem Zeitintervall die Folgeaufrufe des OBs erfolgen sollen.

Beachten Sie beim Einstellen des Startdatums und der Startuhrzeit Folgendes:

- Die Datums- und Uhrzeitangabe am Parameter SDT bezieht sich auf die Systemzeit.
- Bei der Startuhrzeit wird Ihre Sekunden- und Millisekundenangabe ignoriert und auf "0" gesetzt.
- Falls Sie einen monatlichen Start eines Uhrzeitalarm-OBs vorgeben wollen, sind beim Startdatum nur die Tage 1, 2, ... 28 möglich. Diese Einschränkung verhindert ein Überspringen des monatlichen Aufrufs (z. B. bei Monaten mit 30 Tagen oder im Februar). Alternativ zu dem 29., 30. und 31. eines Monats können Sie am Parameter PERIOD die Einstellung "Monatsende" (W#16#2001) verwenden.

Nach dem Stellen des Uhrzeitalarms mit "SET_TINT" müssen Sie diesen noch mit der Anweisung "ACT_TINT" aktivieren.

Hinweis

Weitere Informationen zu Uhrzeitalarm-OBs

Weitere Besonderheiten für die Verwendung von Uhrzeitalarm-OBs finden Sie in den Beschreibungen der Organisationsbausteine der jeweiligen CPU:

Für die S7-1200: AUTOHOTSPOT

Für die S7-1500: AUTOHOTSPOT

Die Einstellungen am Parameter SDT und PERIOD entsprechen den Einstellungen zum Uhrzeitalarm in den Eigenschaften des Uhrzeitalarm-OBs.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_TOD	E, A, M, D, L oder Konstante	Nummer des Uhrzeitalarm-OBs • Für die Uhrzeitalarm-OBs stehen die Nummern 10 bis 17 zur Verfügung. • Alternativ kann auch eine OB-Nummer ab 123 vergeben werden. Die OB-Nummer wird im Ordner Programmbausteine und in den Systemkonstanten angezeigt.
SDT	Input	DT	D, L oder Konstante	Startdatum und Startuhrzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PERIOD	Input	WORD	E, A, M, D, L oder Konstante	Ausführungsintervalle vom Ausgangspunkt SDT an: • W#16#0000 = Einmalige Ausführung • W#16#0201 = Minütlich • W#16#0401 = Stündlich • W#16#1001 = Täglich • W#16#1201 = Wöchentlich • W#16#1401 = Monatlich • W#16#1801 = Jährlich • W#16#2001 = Monatsende
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Fehler am Parameter OB_NR (kein Uhrzeitalarm-OB adressiert).
8091	Fehler am Parameter SDT (ungültige Angabe zu Datum und Uhrzeit).
8092	Fehlerhafte Eingabe am Parameter PERIOD.
80A1	Der eingestellte Startzeitpunkt liegt in der Vergangenheit. Dieser Fehlercode tritt nur bei PERIOD = W#16#0000 auf.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Uhrzeitalarm-Funktionen (Seite 2561).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ACT_TINT: Uhrzeitalarm aktivieren (Seite 2558)

SET_TINTL: Uhrzeitalarm einstellen

Beschreibung

Mit der Anweisung "SET_TINTL" können Sie Startdatum und -uhrzeit der Uhrzeitalarm-Organisationsbausteine aus dem Anwenderprogramm heraus einstellen, ohne dass in der Hardware-Konfiguration Einstellungen vorgenommen wurden.

- An dem Parameter OB_NR geben Sie die Nummer des Uhrzeitalarm-OBs an, für den Sie Startdatum und -uhrzeit einstellen möchten.
- Über die Parameter SDT und PERIOD geben Sie an, wann und wie häufig der Uhrzeitalarm-OB aufgerufen werden soll:
  - Einmaliger Aufruf: Geben Sie am Parameter SDT das Datum und die Uhrzeit an. Verwenden Sie am Parameter PERIOD den Wert "0".
  - Mehrmaliger Aufruf: Geben Sie am Parameter SDT das Datum und die Uhrzeit des ersten Aufrufs an. Stellen Sie über den Parameter PERIOD ein, in welchem Zeitintervall die Folgeaufrufe des OBs erfolgen sollen.
- Über den Parameter LOCAL wählen Sie aus, ob sich die Zeitangabe am Parameter SDT auf die Lokalzeit oder die Systemzeit bezieht.
- Über den Parameter ACTIVATE geben Sie vor, wann die vorgenommenen Einstellungen für den Organisationsbaustein übernommen werden:
  - ACTIVATE = true: Die vorgenommenen Einstellungen werden direkt übernommen.
  - ACTIVATE = false: Die Einstellungen werden erst nach dem Aufruf von "ACT_TINT (Seite 2558)" übernommen.

Beachten Sie beim Einstellen des Startdatums und der Startuhrzeit Folgendes:

- Bei der Startuhrzeit wird Ihre Sekunden- und Millisekundenangabe ignoriert und auf "0" gesetzt.
- Falls Sie einen monatlichen Start eines Uhrzeitalarm-OBs vorgeben wollen, sind beim Startdatum nur die Tage 1, 2, ... 28 möglich. Diese Einschränkung verhindert ein Überspringen des monatlichen Aufrufs (z. B. bei Monaten mit 30 Tagen oder im Februar). Alternativ zu dem 29., 30. und 31. eines Monats können Sie am Parameter PERIOD die Einstellung "Monatsende" (W#16#2001) verwenden.

Beachten Sie bei der Verwendung der Lokalzeit Folgendes:

- Wechsel von Sommer- auf Winterzeit: Verwenden Sie beim Aufruf der Uhrzeitalarm-Organisationsbausteine mit einer Startzeit innerhalb der zweiten Stunde beim Wechsel von Sommer- auf Winterzeit zusätzlich einen Verzögerungsalarm während der ersten Stunde der Zeitumstellung.
- Wechsel von Winter- auf Sommerzeit: Wenn Sie für den Tag der Umstellung auf die Sommerzeit als Uhrzeit die übersprungene Stunde angeben, wird bei einer einmaligen Ausführung (PERIOD = W#16#0000) der Fehlercode 16#8091 ausgegeben.

Hinweis

Weitere Informationen zu Uhrzeitalarm-OBs

Weitere Besonderheiten für die Verwendung von Uhrzeitalarm-OBs finden Sie in den Beschreibungen der Organisationsbausteine der jeweiligen CPU:

Für die S7-1200: AUTOHOTSPOT

Für die S7-1500: AUTOHOTSPOT

Die Einstellungen am Parameter SDT und PERIOD entsprechen den Einstellungen zum Uhrzeitalarm in den Eigenschaften des Uhrzeitalarm-OBs.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_TINTL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_TOD	E, A, M, D, L oder Konstante	Nummer des Uhrzeitalarm-OBs • Für die Uhrzeitalarm-OBs stehen die Nummern 10 bis 17 zur Verfügung. • Alternativ kann auch eine OB-Nummer ab 123 vergeben werden. Die OB-Nummer wird im Ordner Programmbausteine und in den Systemkonstanten angezeigt.
SDT	Input	DTL	D, L oder Konstante	Startdatum und Startuhrzeit
LOCAL	Input	BOOL	E, A, M, D, L oder Konstante	• true: Lokalzeit verwenden • false: Systemzeit verwenden
PERIOD	Input	WORD	E, A, M, D, L oder Konstante	Ausführungsintervalle vom Ausgangspunkt SDT an: • W#16#0000 = Einmalige Ausführung • W#16#0201 = Minütlich • W#16#0401 = Stündlich • W#16#1001 = Täglich • W#16#1201 = Wöchentlich • W#16#1401 = Monatlich • W#16#1801 = Jährlich • W#16#2001 = Monatsende
ACTIVATE	Input	BOOL	E, A, M, D, L oder Konstante	• true: Uhrzeitalarm stellen und aktivieren • false: Uhrzeitalarm stellen und nur bei Aufruf von "ACT_TINT (Seite 2558)" aktivieren
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Fehler am Parameter OB_NR (kein Uhrzeitalarm-OB adressiert).
8091	Fehler am Parameter SDT (ungültige Angabe zu Datum und Uhrzeit).
8092	Fehlerhafte Eingabe am Parameter PERIOD.
80A1	Der eingestellte Startzeitpunkt liegt in der Vergangenheit. Dieser Fehlercode tritt nur bei PERIOD = W#16#0000 auf.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

CAN_TINT: Uhrzeitalarm stornieren

Beschreibung

Löschen Sie mit der Anweisung "CAN_TINT" das Startdatum und -uhrzeit eines angegebenen Uhrzeitalarm-Organisationsbausteins. Der Uhrzeitalarm wird dadurch deaktiviert und der Organisationsbaustein wird nicht mehr aufgerufen.

Möchten Sie den Uhrzeitalarm wieder nutzen, müssen Sie zuerst den Startzeitpunkt neu stellen (Anweisung "SET_TINTL (Seite 2555)" oder "SET_TINT (Seite 2552)").

Danach müssen Sie den Uhrzeitalarm wieder aktivieren:

- Wenn Sie zum stellen des Uhrzeitalarms die Anweisung "SET_TINT (Seite 2552)" oder "SET_TINTL (Seite 2555)" mit Parameter ACTIVE=false verwendet haben, rufen Sie "ACT_TINT (Seite 2558)" auf.
- Bei der Anweisung "SET_TINTL (Seite 2555)" können Sie den Uhrzeitalarm auch direkt über den Parameter ACTIVE=true aktivieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_TOD	E, A, M, D, L oder Konstante	Nummer des Uhrzeitalarm-OBs, dessen Startdatum und -uhrzeit gelöscht werden soll.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Fehler am Parameter OB_NR.
80A0	Kein Startdatum/-uhrzeit festgelegt für den betreffenden Uhrzeitalarm-OB.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Uhrzeitalarm-Funktionen (Seite 2561).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ACT_TINT: Uhrzeitalarm aktivieren

Beschreibung

Mit der Anweisung "ACT_TINT" aktivieren Sie einen Uhrzeitalarm-Organisationsbaustein aus dem Anwenderprogramm. Voraussetzung zur Ausführung der Anweisung ist, dass Startdatum und -uhrzeit für den Uhrzeitalarm-OB bereits eingestellt sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ACT_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_TOD	E, A, M, D, L oder Konstante	Nummer des Uhrzeitalarm-OBs • Für die Uhrzeitalarm-OBs stehen die Nummern 10 bis 17 zur Verfügung. • Alternativ kann auch eine OB-Nummer ab 123 vergeben werden. Die OB-Nummer wird im Ordner Programmbausteine und in den Systemkonstanten angezeigt.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Fehler am Parameter OB_NR (kein Uhrzeitalarm-OB adressiert).
80A0	Startdatum und -uhrzeit sind nicht gestellt für den betreffenden Uhrzeitalarm-OB.
80A1	Die aktivierte Zeit liegt in der Vergangenheit. Wenn der Uhrzeitalarm nur einmalig ausgeführt werden soll, tritt nur der Fehler auf.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Uhrzeitalarm-Funktionen (Seite 2561).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

QRY_TINT: Status des Uhrzeitalarms abfragen

Beschreibung

Mit der Anweisung können Sie sich den Status eines Uhrzeitalarm-Organisationsbausteins am Ausgangsparameter STATUS anzeigen lassen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_TOD	E, A, M, D, L oder Konstante	Nummer des Uhrzeitalarm-OBs, dessen Status abgefragt wird Die OB-Nummer wird im Ordner Programmbausteine und in den Systemkonstanten angezeigt.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode.
STATUS	Output	WORD	E, A, M, D, L	Status des Uhrzeitalarms (siehe unten)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten. Ausnahme: Statusmeldung "0" am Bit 4 (kein OB mit dieser Nummer erstellt).
8090	Fehler am Parameter OB_NR. Mögliche Ursachen: • Der Wert am Parameter OB_NR ist keine durch die CPU unterstützte OB-Nummer (<1 oder >32767). • Die OB-Nummer adressiert kein Uhrzeitalarm-OB.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS

Wenn ein Fehler auftritt (siehe Parameter RET_VAL), wird am Parameter STATUS "0" ausgegeben.

Bit	Wert	Bedeutung
0	Immer "0"	Irrelevant
1	0	Der Uhrzeitalarm ist freigegeben.

Bit	Wert	Bedeutung
	1	Der Uhrzeitalarm ist gesperrt.
2	0	Der Uhrzeitalarm ist nicht aktiviert oder abgelaufen.
	1	Der Uhrzeitalarm ist aktiviert.
4	0	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben existiert nicht.
	1	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben ist vorhanden.
6	0	Grundlage für den Uhrzeitalarm ist die Systemzeit
	1	Grundlage für den Uhrzeitalarm ist die Lokalzeit
Andere		Immer "0"

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Uhrzeitalarm-Funktionen (Seite 2561).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Programmbeispiel zu Uhrzeitalarm-Funktionen

Einleitung

In dem folgenden Beispiel lesen Sie von einem Uhrzeitalarm-OB (OB10) den Status aus und stellen einen Startzeitpunkt sowie die Aufrufhäufigkeit ein. Nachdem der OB10 gestartet wurde, deaktivieren Sie den OB10 wieder.

Der OB10 besitzt die folgenden Voreinstellungen:

Voraussetzung

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein 13 Variablen und eine Struktur an.

SLI_gDB_SET_TINT

	Name	Data type	Start value
1	Static		
2	obNbr	OB_TOD	10
3	executeQry	Bool	false
4	itemiseStatus	Struct	
5	T-o-d_interrupt_enabled	Bool	false
6	T-o-d_interrupt_activated	Bool	false
7	OB_with_obNbr_exists	Bool	false
8	T-o-d_interrupt_on_system_time	Bool	false
9	T-o-d_interrupt_on_local_time	Bool	false
10	returnValueQry	Int	0
11	executeSet	Bool	false
12	startDateTime	Date_And_Time	DT#1990-01-01-0
13	presetTime	Time	T#5M
14	periodOfExecute	Word	16#0201
15	returnValueSet	Int	0
16	executeAct	Bool	false
17	returnValueAct	Int	0
18	executeCan	Bool	false
19	returnValueCan	Int	0
20	testCurrCount	Int	0

Parameter verschalten: FC "SLI_itemiseStatus_TINT"

Zum Auswerten der Statusinformationen legen Sie einen FC "SLI_itemiseStatus_TINT" an.

Darin legen Sie die folgende lokale Variable an.

SLI_itemiseStatus_TINT

	Name	Data type
1	Input	
2	inWord	Word

Für alle relevanten Bits legen Sie Verschaltungen an.

Netzwerk 1:

Netzwerk 2:

Netzwerk 3:

Netzwerk 4:

Netzwerk 5:

Parameter verschalten: FC "SLI_retrievetime_TINT"

Zum Auslesen des Datums (Systemzeit) und Definieren des Startzeitpunkts des OB10 legen Sie einen FC "SLI_retrievetime_TINT" an. In dem FC legen Sie die folgenden lokalen Variablen an.

7		Temp	
8		statTIME	Date_And_Time
9		statRetVal	Int

Netzwerk 1: Die Parameter verschalten Sie wie folgt. "RD_SYS_T" liesst die Systemzeit aus. Über "T_ADD" wird der Systemzeit 5 Minuten Zeit dazu addiert.

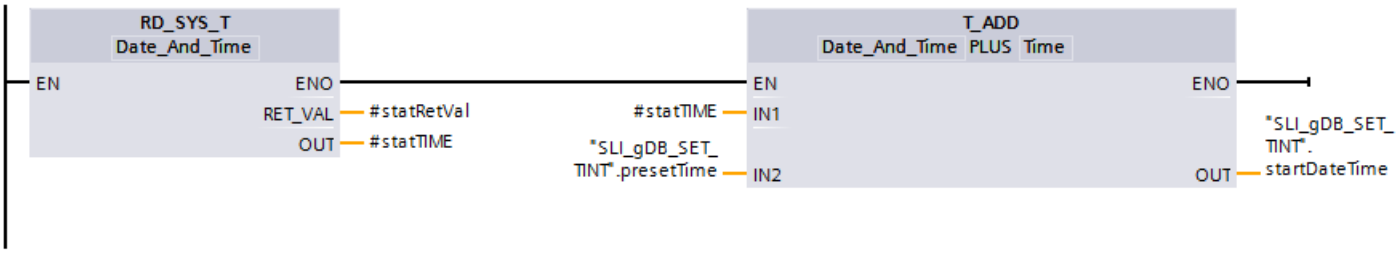

Parameter verschalten: FB "SLI_FB_SET_TINT"

Zum Aufruf der Anweisungen erstellen Sie einen FB ("SLI_FB_SET_TINT"). In dem FB "SLI_FB_SET_TINT" legen Sie die folgenden lokalen Variablen an.

7		Static		
8		statMem	Bool	false
9		statWord	Word	16#0

Netzwerk 1: Die Parameter der Anweisung "QRY_TINT" verschalten Sie wie folgt.

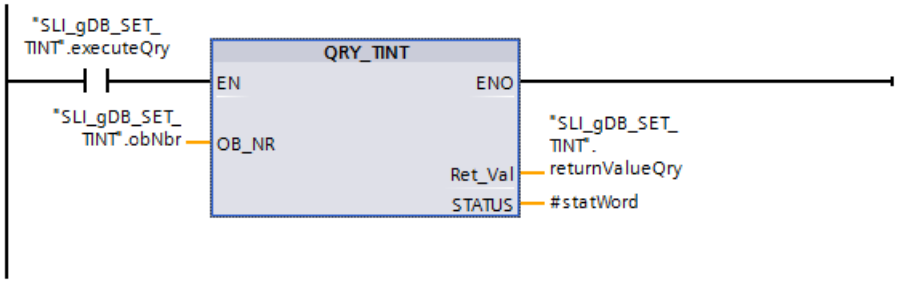

Netzwerk 2: Den FC "SLI_retrievetime_TINT" rufen Sie im Netzwerk 2 des FBs auf.

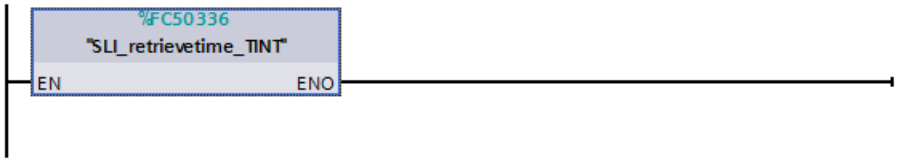

Netzwerk 3: Die Parameter der Anweisung "SET_TINT" verschalten Sie wie folgt.

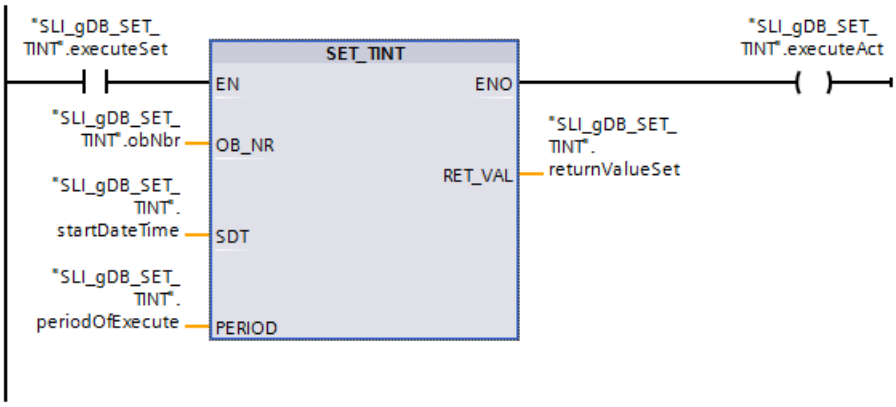

Netzwerk 4: Die Parameter der Anweisung "ACT_TINT" verschalten Sie wie folgt.

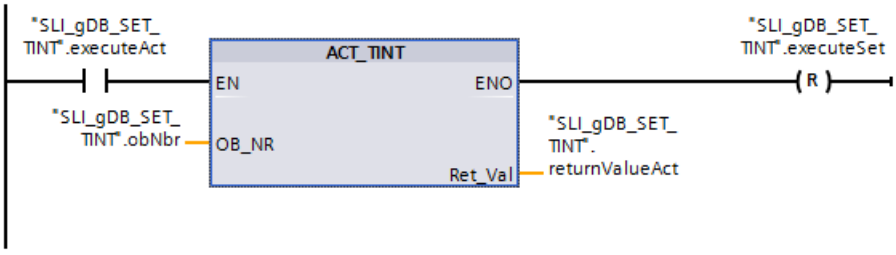

Netzwerk 5: Die Parameter der Anweisung "CAN_TINT" verschalten Sie wie folgt.

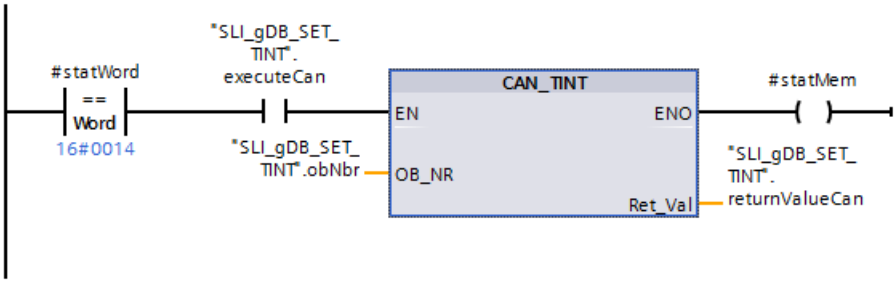

Netzwerk 6: Die mit der Ausführung von "CAN_TINT" rückzusetzenden Parameter verschalten Sie wie folgt.

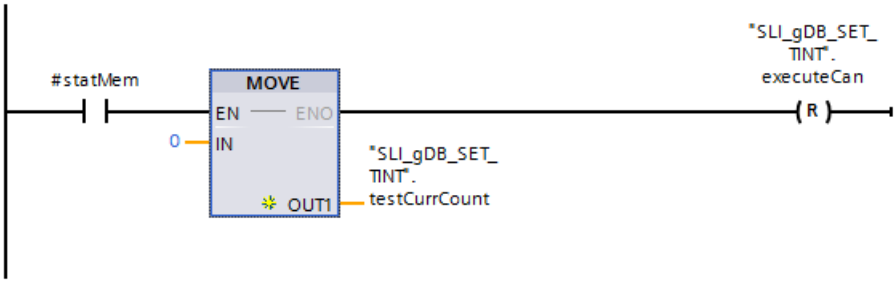

Netzwerk 7: Den Parameter des FC "SLI_itemiseStatus_TINT" verschalten Sie wie folgt.

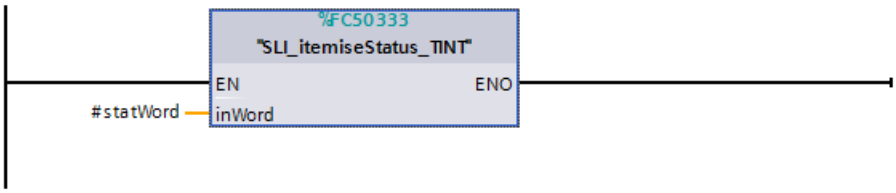

Parameter verschalten: OB10

Im OB10 legen Sie folgende Verschaltungen an.

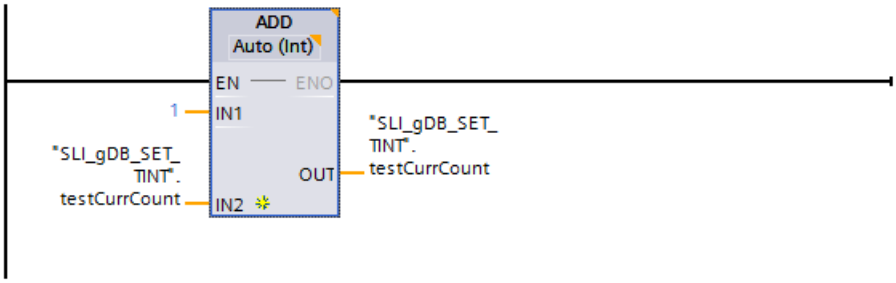

Ergebnis von QRY_TINT

Über den Eingangsparameter OB_NR ("obNbr") wird den Anweisungen die Nummer des aufzurufenden OBs (OB10) mitgeteilt.

Netzwerk1:

Wenn der Schließerkontakt ("executeQry") den Signalzustand "TRUE" liefert, wird die Anweisung "QRY_TINT" ausgefuhrt und die Daten des OB10 ausgelesen. Der Ausgangsparameter STATUS ("#statWord") zeigt den Status des OB10 an. Am Ausgangsparameter RET_VAL ("returnValueQry") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft.

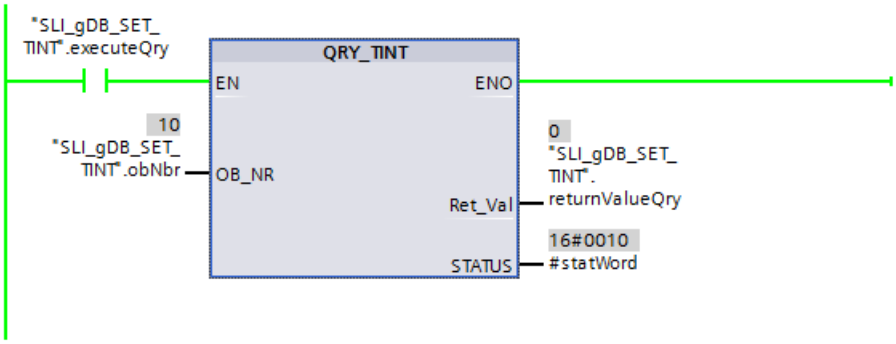

Der Wert der Variable "#statWord" wird in der Struktur "itemiseStatus" abgebildet:

4		itemiseStatus	Struct		
5		T-o-d_interrupt_enabled	Bool	false	TRUE
6		T-o-d_interrupt_activated	Bool	false	FALSE
7		OB_with_obNbr_exists	Bool	false	TRUE
8		T-o-d_interrupt_on_system_time	Bool	false	TRUE
9		T-o-d_interrupt_on_local_time	Bool	false	FALSE

Ergebnis von SET_TINT

Netzwerk 3: Wenn der Schließerkontakt ("executeSet") den Signalzustand "TRUE" liefert, wird die Anweisung "SET_TINT" ausgefuhrt. Über den Eingangsparameter SDT ("startDateTime") wird dem OB10 der definierte Startzeitpunkt (siehe Netzwerk 2) übergeben. Am Eingangsparameter PERIOD ("periodOfExecute") ist als Aufrufhäufigkeit "jede Minute" (Wert "16#0201") hinterlegt.

Am Ausgangsparameter RET_VAL ("returnValueSet") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Durch die Ausführung von SET_TINT wird die Spule ("executeAct") auf "TRUE" gesetzt und im Netzwerk 4 die Anweisung "ACT_TINT" ausgeführt.

Ergebnis von ACT_TINT

Netzwerk 4: Die Anweisung "ACT_TINT" aktiviert den OB10. Am Ausgangsparameter RET_VAL ("returnValueAct") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Durch den Rücksetzer ("executeSet") im Signalzustand "TRUE" wird die Ausführung der Anweisung SET_TINT und damit auch der Anweisung ACT_TINT automatisch beendet.

Hinweis

Nach der Ausführung der Anweisung "SET_TINT" behält der OB10 die neuen Aufruf-Daten bei bis zum STOPP der CPU. Die voreingestellten Eigenschaften des OB10 bleiben von der Ausführung der Anweisung "SET_TINT" unberührt.

Ergebnis des OB10

Bei jedem Aufruf des OB10 wird die Anweisung "ADD" ausgeführt und die Variable "testCurrCount" um "1" erhöht. Der Aufruf des OB10 erfolgt (gemäß dem Wert von "periodOfExecute") einmal jede Minute.

SLI_gDB_SET_TINT

	Name	Data type	Start value	Monitor value
1	Static			
2	obNbr	OB_TOD	10	10
3	executeQry	Bool	false	TRUE
4	itemiseStatus	Struct		
5	T-o-d_interrupt_enabled	Bool	false	TRUE
6	T-o-d_interrupt_activated	Bool	false	TRUE
7	OB_with_obNbr_exists	Bool	false	TRUE
8	T-o-d_interrupt_on_system_time	Bool	false	TRUE
9	T-o-d_interrupt_on_local_time	Bool	false	FALSE
10	returnValueQry	Int	0	0
11	executeSet	Bool	false	FALSE
12	startDateTime	Date_And_Time	DT#1990-01-01-0	DT#2016-11-30-17...
13	presetTime	Time	T#5M	T#5M
14	periodOfExecute	Word	16#0201	16#0201
15	returnValueSet	Int	0	0
16	executeAct	Bool	false	FALSE
17	returnValueAct	Int	0	0
18	executeCan	Bool	false	FALSE
19	returnValueCan	Int	0	0
20	testCurrCount	Int	0	5

Ergebnis von CAN_TINT

Netzwerk 5: Wenn der OB10 eingestellt und aktiviert ist ("#statWord" hat den Wert "16#0014"), liefert der Vergleicher den Signalzustand "TRUE" und CAN_TINT kann ausgeführt werden.

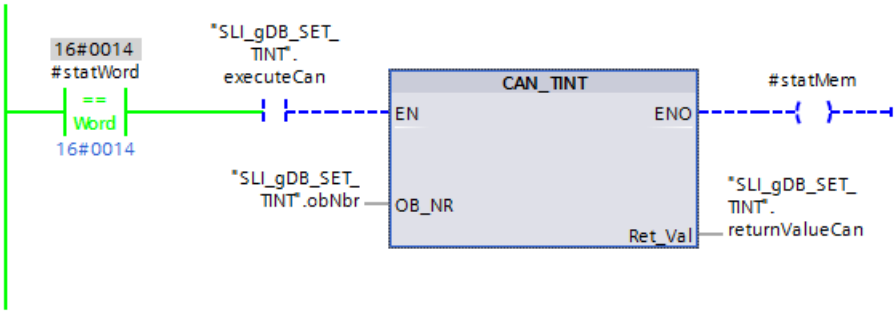

Wenn der Schließerkontakt ("executeCan") den Signalzustand "TRUE" liefert, wird die Anweisung "CAN_TINT" ausgefuhrt. Die Anweisung deaktiviert den OB10 und löscht die per SET_TINT gesetzten Aufruf-Daten des OB10. Am Ausgangsparameter RET_VAL ("returnValueCan") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Durch die Ausführung von CAN_TINT wird die Spule ("#statMem") auf "TRUE" gesetzt und das Netzwerk 6 ausgeführt.

Ergebnis des Rücksetzens

Netzwerk 6: Wenn der Schließerkontakt ("#statMem") den Signalzustand "TRUE" liefert, wird über die Anweisung "MOVE" die Variable "testCurrCount" auf den Wert "0" rückgesetzt. Zudem wird durch den Rücksetzer ("executeCan") im Signalzustand "TRUE" die Ausführung der Anweisung "CAN_TINT" automatisch beendet.

Ergebnis der Statusinformation "#statWord"

Im Netzwerk 7 wird der FC "SLI_itemiseStatus_TINT" aufgerufen. Die bedeutenden Bits der Variable "#statWord" werden ausgewertet und als boolsche Variablen mit Klarnamen ausgegeben. Wenn eine Variable der Struktur "itemiseStatus" den Signalzustand "TRUE" liefert, ist deren Aussage zutreffend.

Weitere Informationen

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Verzögerungsalarm

Verwendung von Verzögerungsalarmen

Definition

Nachdem Sie die Anweisung "SRT_DINT (Seite 2570)" aufgerufen haben, wird vom Betriebssystem nach Ablauf der parametrierten Verzögerungszeit ein Alarm generiert. Der parametrierte Verzögerungsalarm-OB wird aufgerufen.

Voraussetzungen für Aufruf

Damit ein Verzögerungsalarm vom Betriebssystem aufgerufen werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Verzögerungsalarm-OB muss gestartet sein durch die Anweisung "SRT_DINT (Seite 2570)".
- Der Verzögerungsalarm-OB darf nicht per Projektierung abgewählt sein.
- Der Verzögerungsalarm-OB muss in der CPU vorhanden sein.

Zweck der Anweisungen "SRT_DINT", "CAN_DINT" und "QRY_DINT"

Die Anweisungen verwenden Sie, um Verzögerungsalarme

- zu starten ("SRT_DINT (Seite 2570)")
- zu stornieren ("CAN_DINT (Seite 2572)")
- abzufragen ("QRY_DINT (Seite 2573)").

Auswirkungen auf den Verzögerungsalarm

Die folgende Tabelle zeigt, welche Situationen welche Auswirkungen auf den Verzögerungsalarm haben.

Wenn ...	und ...	dann ...
ein Verzögerungsalarm gestartet wird (Aufruf von "SRT_DINT (Seite 2570)")	der Verzögerungsalarm bereits gestartet ist,	wird die Verzögerungszeit überschrieben; der Verzögerungsalarm wird erneut gestartet.
	der Verzögerungsalarm-OB nicht vorhanden ist zum Zeitpunkt seines Aufrufs,	erfolgt ein Diagnosepuffereintrag und die CPU läuft weiter.
	der Start in einem Anlauf-OB erfolgte und die Verzögerungszeit abgelaufen ist, bevor die CPU im RUN ist,	verzögert sich der Aufruf des Verzögerungsalarm-OBs, bis die CPU im RUN ist.
die Verzögerungszeit abgelaufen ist	ein zuvor gestarteter Verzögerungsalarm-OB noch in Bearbeitung ist,	erzeugt das Betriebssystem einen Zeitfehler (Aufruf des OB 80; falls der OB 80 nicht vorhanden ist, läuft die CPU weiter.).

Verhalten bei Neustart (Warmstart) und Kaltstart

Bei Neustart (Warmstart) und Kaltstart werden alle im Anwenderprogramm über Anweisungen eingestellte Verzögerungsalarm-Einstellungen gelöscht.

Start in einem Anlauf-OB

Ein Verzögerungsalarm kann in einem Anlauf-OB gestartet werden. Für den Aufruf des Verzögerungsalarm-OBs müssen zwei Bedingungen erfüllt sein:

- Die Verzögerungszeit ist abgelaufen.
- Die CPU ist im Betriebszustand RUN.

Wenn die Verzögerungszeit abgelaufen ist und die CPU sich noch nicht im Betriebszustand RUN befindet, dann verzögert sich der Aufruf des Verzögerungsalarm-OB, bis die CPU in RUN ist. Der Verzögerungsalarm-OB wird dann noch vor der ersten Anweisung im OB Main[OB 1] aufgerufen.

SRT_DINT: Verzögerungsalarm starten

Beschreibung

Mit der Anweisung "SRT_DINT" starten Sie einen Verzögerungsalarm, der nach Ablauf der am Parameter DTIME angegebenen Verzögerungszeit einen Verzögerungsalarm-OB aufruft. Der Verzögerungsalarm-OB wird einmalig für den Verzögerungsalarm ausgeführt.

Wenn der Verzögerungsalarm noch nicht ausgeführt wurde und die Anweisung "SRT_DINT" von Neuem aufgerufen wird, wird der bestehende Verzögerungsalarm gelöscht und ein neuer Verzögerungsalarm gestartet.

Hinweis

Verzögerungszeit

Wenn die verwendete Verzögerungszeit kleiner gleich der aktuell verwendeten Zykluszeit der CPU ist und SRT_DINT zyklisch aufgerufen wird, wird der Verzögerungsalarm-OB mit jedem CPU-Zyklus ausgeführt.

Achten Sie darauf, eine Verzögerungszeit zu wählen, welche größer als die Zykluszeit der CPU ist.

Am Parameter SIGN können Sie ein anwenderspezifisches Kennzeichen angeben, um den Start des Verzögerungsalarms zu kennzeichnen. Die Werte der Parameter DTIME und SIGN werden in der Startinformation des aufgerufenen Organisationsbausteins angezeigt.

Genauigkeit

Die Zeit zwischen Aufruf der Anweisung "SRT_DINT" und dem Start des Verzögerungsalarm-OBs ist maximal eine Millisekunde größer als die parametrierte Verzögerungszeit, sofern keine Unterbrechungsereignisse den Aufruf verzögern.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SRT_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_DELAY (INT)	E, A, M, D, L oder Konstante	Nummer des OBs, der nach einer Verzögerungszeit ausgeführt wird
DTIME	Input	TIME	E, A, M, D, L oder Konstante	Verzögerungszeit (1 bis 60000 ms) Größere Zeiten können Sie realisieren, indem Sie z. B. in einem Verzögerungsalarm-OB einen Zähler benutzen.
SIGN	Input	WORD	E, A, M, D, L oder Konstante	Kennzeichen, das beim Aufruf des Verzögerungsalarm-OBs in der Startereignisinformation des OBs erscheint.
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler
8090	Fehlerhafter Parameter OB_NR
8091	Fehlerhafter Parameter DTIME
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Verzögerungsalarm-Funktionen (Seite 2574).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

CAN_DINT: Verzögerungsalarm stornieren

Beschreibung

Mit dieser Anweisung stornieren Sie einen gestarteten Verzögerungsalarm und dadurch auch den Aufruf des Verzögerungsalarm-OBs, der nach der parametrierten Verzögerungszeit ausgeführt wird. Die Nummer des Organisationsbausteins, dessen Aufruf storniert wird, geben Sie am Parameter OB_NR an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_DELAY (INT)	E, A, M, D, L oder Konstante	Nummer des OBs, dessen Aufruf storniert wird
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler
8090	Fehlerhafter Parameter OB_NR
80A0	Verzögerungsalarm ist nicht gestartet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Verzögerungsalarm-Funktionen (Seite 2574).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

QRY_DINT: Zustand eines Verzögerungsalarms abfragen

Beschreibung

Mit der Anweisung "QRY_DINT" fragen Sie den Zustand eines Verzögerungsalarms ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	OB_DELAY (INT)	E, A, M, D, L oder Konstante	Nummer des OBs, dessen Zustand abgefragt wird.
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Aktualparameter von RET_VAL einen Fehlercode. Am Parameter STATUS wird als Wert "0" angezeigt.
STATUS	Output	WORD	E, A, M, D, L	Zustand des Verzögerungsalarms; siehe folgende Tabelle.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Bit	Wert	Bedeutung
0	0	Irrelevant
1	0	Verzögerungsalarm ist vom Betriebssystem freigegeben.

Bit	Wert	Bedeutung
	1	Verzögerungsalarm ist gesperrt.
2	0	Verzögerungsalarm ist nicht aktiviert oder abgelaufen.
	1	Verzögerungsalarm ist aktiviert.
3	-	-
4	0	Verzögerungsalarm-OB mit der angegebenen Nummer existiert nicht.
	1	Verzögerungsalarm-OB mit der angegebenen Nummer vorhanden.
Weitere Bits		Immer "0"

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Fehlerhafte Angabe am Parameter OB_NR
Allgemeine Fehler-information	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu Verzögerungsalarm-Funktionen (Seite 2574).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Programmbeispiel zu Verzögerungsalarm-Funktionen

Einleitung

In dem folgenden Beispiel lesen Sie von einem Verzögerungsalarm-OB (OB20) den Status aus, stellen eine Verzögerungszeit ein und starten oder deaktivieren den OB20.

Voraussetzung

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein 10 Variablen und eine Struktur an.

SLI_gDB_SRT_DINT			
	Name	Data type	Start value
1	Static		
2	obNbr	OB_DELAY	20
3	executeQry	Bool	false
4	itemiseStatus	Struct	
5	Time-delay_interrupt_enabled	Bool	false
6	Time-delay_interrupt_activated	Bool	false
7	OB_with_obNbr_exists	Bool	false
8	returnValueQry	Int	0
9	executeSrt	Bool	false
10	delayTime	Time	T#6000ms
11	sign	Word	16#0001
12	returnValueSrt	Int	0
13	executeCan	Bool	false
14	returnValueCan	Int	0
15	testCurrCount	Int	0

Parameter verschalten: FC "SLI_itemiseStatus_DINT"

Zum Auswerten der Statusinformationen legen Sie einen FC "SLI_itemiseStatus_DINT" an.

Darin legen Sie die folgende lokale Variable an.

SLI_itemiseStatus_DINT		
	Name	Data type
1	Input	
2	inWord	Word

Für alle relevanten Bits legen Sie Verschaltungen an.

Netzwerk 1:

Netzwerk 2:

Netzwerk 3:

Parameter verschalten: FB "SLI_FB_SRT_DINT"

Zum Aufruf der Anweisungen erstellen Sie einen FB "SLI_FB_SRT_DINT". In dem FB "SLI_FB_SRT_DINT" legen Sie die folgenden lokalen Variablen an.

7		Static		
8		statWord	Word	16#0
9		statMem	Bool	false

Netzwerk 1: Die Parameter der Anweisung "QRY_DINT" verschalten Sie wie folgt.

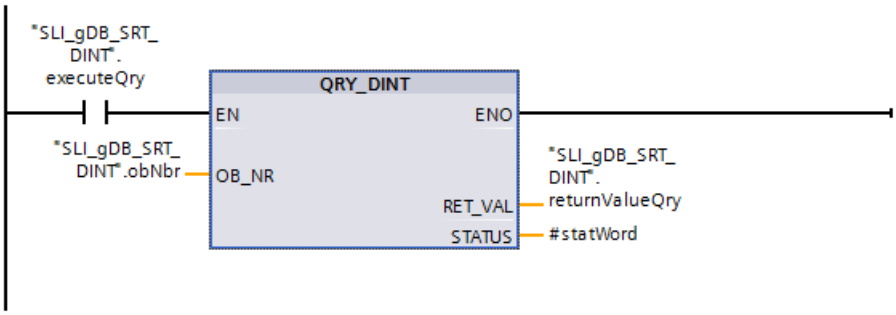

Netzwerk 2: Die Parameter der Anweisung "SRT_DINT" verschalten Sie wie folgt.

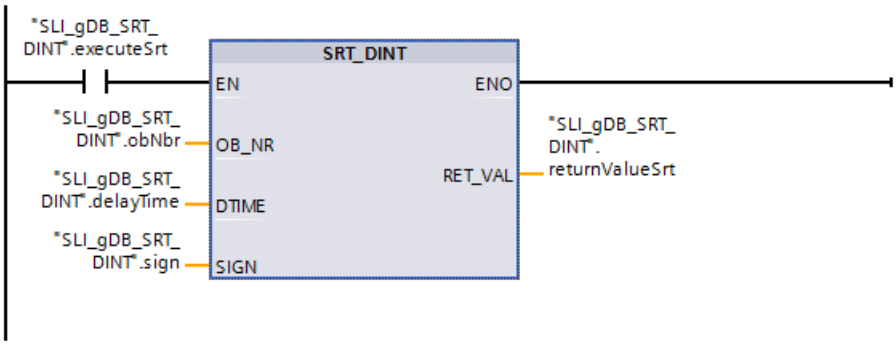

Netzwerk 3: Die Parameter der Anweisung "CAN_DINT" verschalten Sie wie folgt.

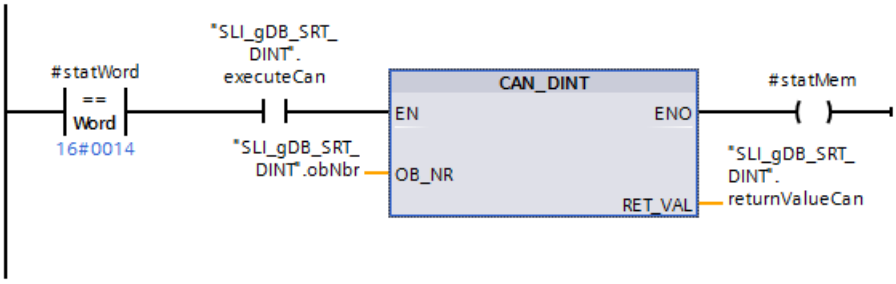

Netzwerk 4: Die mit der Ausführung von "CAN_DINT" rückzusetzenden Parameter verschalten Sie wie folgt.

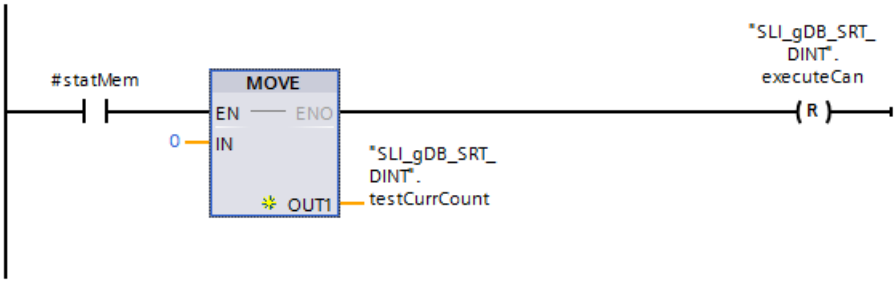

Netzwerk 5: Den Parameter des FC "SLI_itemiseStatus_DINT" verschalten Sie wie folgt.

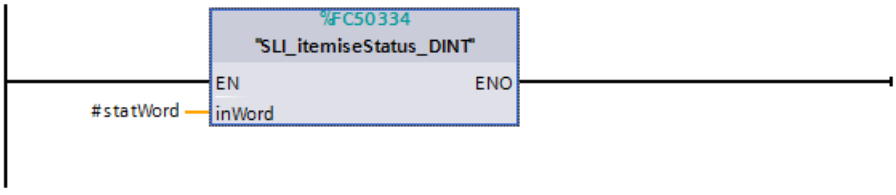

Den FB "SLI_FB_SRT_DINT" rufen Sie in einem Zyklus-OB (OB1) auf.

Parameter verschalten: OB20

Im OB20 legen Sie folgende Verschaltungen an.

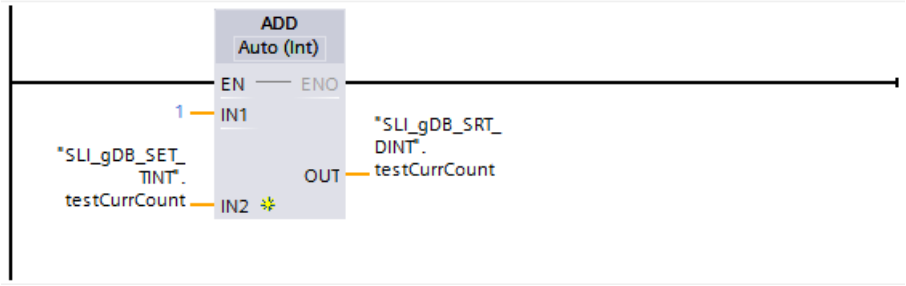

Ergebnis von QRY_DINT

Über den Eingangsparameter OB_NR ("obNbr") wird den Anweisungen die Nummer des aufzurufenden OBs (OB20) mitgeteilt.

Netzwerk 1:

Wenn der Schließerkontakt ("executeQry") den Signalzustand "TRUE" liefert, wird die Anweisung "QRY_DINT" ausgefuhrt und die Daten des OB20 ausgelesen. Der Ausgangsparameter STATUS ("#statWord") zeigt den Status des OB20 an. Am Ausgangsparameter RET_VAL ("returnValueQry") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft.

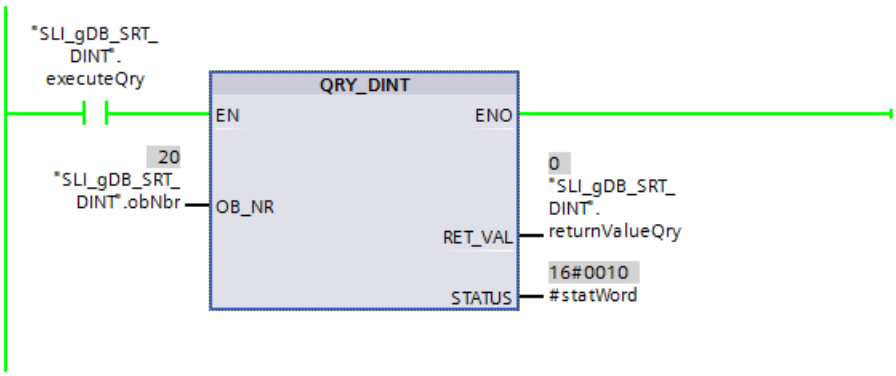

Der Wert der Variable "#statWord" wird in der Struktur "itemiseStatus" abgebildet:

4	itemiseStatus	Struct		
5	Time-delay_interrupt_enabled	Bool	false	TRUE
6	Time-delay_interrupt_activated	Bool	false	FALSE
7	OB_with_obNbr_exists	Bool	false	TRUE

Ergebnis von SRT_DINT

Netzwerk 2: Wenn der Schließerkontakt ("executeSrt") den Signalzustand "TRUE" liefert, wird die Anweisung "SRT_DINT" ausgefuhrt. Über den Eingangsparameter DTIME ("delayTime") wird dem OB20 die Verzögerungszeit (6 s) übergeben und der OB20 aktiviert.

Hinweis

Ab dem Zeitpunkt der Aktivierung des OB20 läuft dessen Verzögerungszeit ab. Danach wird der OB20 aufgerufen, sofern diesem nicht zuvor in einem neuen CPU-Zyklus erneut die Verzögerungszeit übergeben wurde.

Am Eingangsparameter SIGN ("sign") ist das bei Ausführung des OB20 auszugebende, gewählte Kennzeichen (Wert "16#0001") hinterlegt. Das Kennzeichen erscheint in der Startereignisinformation des OB20. Am Ausgangsparameter RET_VAL ("returnValueSrt") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Wenn der Schließerkontakt ("executeSrt") nach der erfolgreichen Verarbeitung von SRT_DINT den Signalzustand "FALSE" liefert, wird dem OB20 die Verzögerungszeit nicht neu übergeben. Sobald die Verzögerungszeit abgelaufen ist, wird der OB20 einmal ausgeführt.

Hinweis

Das erläuterte Verhalten des OB20 in Kombination mit dem Parameter EN gilt nur, wenn die aktuell verwendete Zykluszeit der CPU kleiner ist als die Verzögerungszeit des OB20.

Ergebnis des OB20

Beim Aufruf des OB20 wird die Anweisung "ADD" ausgeführt und die Variable "testCurrCount" um "1" erhöht. Der Aufruf des OB20 erfolgt nur einmal nach Ablauf der über SRT_DINT definierten Verzögerungszeit. Nach seinem Aufruf ist der OB20 wieder im Zustand "deaktiviert".

SLI_gDB_SRT_DINT

	Name	Data type	Start value	Monitor value
1	Static			
2	obNbr	OB_DELAY	20	20
3	executeQry	Bool	false	TRUE
4	itemiseStatus	Struct		
5	Time-delay_interrupt_enabled	Bool	false	TRUE
6	Time-delay_interrupt_activated	Bool	false	FALSE
7	OB_with_obNbr_exists	Bool	false	TRUE
8	returnValueQry	Int	0	0
9	executeSrt	Bool	false	FALSE
10	delayTime	Time	T#6000ms	T#6S
11	sign	Word	16#0001	16#0001
12	returnValueSrt	Int	0	0
13	executeCan	Bool	false	FALSE
14	returnValueCan	Int	0	0
15	testCurrCount	Int	0	1

Ergebnis von CAN_DINT

Netzwerk 3: Solange der OB20 eingestellt und aktiviert ist ("#statWord" hat den Wert "16#0014"), liefert der Vergleicher den Signalzustand "TRUE" und CAN_DINT kann ausgeführt werden.

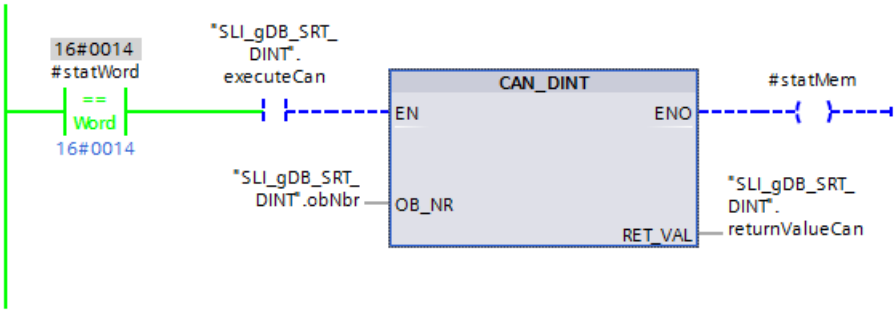

Wenn der Schließerkontakt ("executeCan") den Signalzustand "TRUE" liefert, wird die Anweisung "CAN_DINT" ausgefuhrt. Die Anweisung deaktiviert den OB20 und löscht die gesetzten Aufruf-Daten des OB20.
Am Ausgangsparameter RET_VAL ("returnValueCan") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Durch die Ausführung von CAN_DINT wird die Spule ("#statMem") auf "TRUE" gesetzt und das Netzwerk 4 ausgeführt.

Ergebnis des Rücksetzens

Netzwerk 4: Wenn der Schließerkontakt ("#statMem") den Signalzustand "TRUE" liefert, wird über die Anweisung "MOVE" die Variable "testCurrCount" auf den Wert "0" rückgesetzt. Zudem wird durch den Rücksetzer ("executeCan") im Signalzustand "TRUE" die Ausführung der Anweisung "CAN_DINT" automatisch beendet.

Ergebnis der Statusinformation "#statWord"

Im Netzwerk 7 wird der FC "SLI_itemiseStatus_DINT" aufgerufen. Die bedeutenden Bits der Variable "#statWord" werden ausgewertet und als boolsche Variablen mit Klarnamen ausgegeben. Wenn eine Variable der Struktur "itemiseStatus" den Signalzustand "TRUE" liefert, ist deren Aussage zutreffend.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Synchronfehlerereignisse

Synchronfehlerereignisse maskieren

Einleitung

Synchronfehlerereignisse sind Programmier- und Zugriffsfehlerereignisse. Solche Fehlerereignisse treten auf durch die Programmierung mit falschen Operandenbereichen, -nummern oder falschen Adressen. Diese Fehlerereignisse **maskieren** heißt,

- maskierte Fehlerereignisse lösen keinen Fehler-OB aus und führen nicht zu einer programmierten Ersatzreaktion.
- von den maskierten Fehlern "merkt sich" die CPU die Fehler, die aufgetreten sind, in einem Ereignisstatusregister.

Die Maskierung erfolgt durch Aufruf der Anweisung "MSK_FLT (Seite 2587)".

Fehlerereignisse **demaskieren** heißt, eine vorher vorgenommene Maskierung wieder rückgängig zu machen und das zugehörige Bit im Ereignisstatusregister der aktuellen Prioritätsklasse zu löschen. Die Maskierung wird aufgehoben

- durch einen Aufruf der Anweisung "DMSK_FLT (Seite 2588)"
- wenn die aktuelle Prioritätsklasse beendet ist.

Wenn ein Fehlerereignis nach seiner Demaskierung auftritt, dann wird vom Betriebssystem der zugehörige Fehler-OB gestartet.

Maskierte und aufgetretene Fehlerereignisse können Sie mit der Anweisung "READ_ERR (Seite 2589)" auslesen.

Hinweis

Unabhängig von einer Maskierung oder Demaskierung der Fehlerereignisse erfolgt bei der S7-1500 ein Eintragen des Fehlerereignisses in den Diagnosepuffer, und die Sammelfehler-LED der CPU leuchtet.

Fehlerbehandlung allgemein

Wenn in einem Anwenderprogramm Programmier- und Peripheriezugriffsfehler auftreten, dann können Sie verschieden darauf reagieren:

- Sie können einen Fehler-OB programmieren und das Betriebssystem ruft diesen Fehler-OB auf, wenn ein entsprechendes Fehlerereignis auftritt.
- Sie können den Aufruf des Fehler-OB für jede Prioritätsklasse einzeln sperren. Wenn in der Prioritätsklasse ein entsprechender Fehler auftritt, geht die CPU in diesem Fall nicht in STOP. Die CPU trägt die aufgetretenen Fehler in ein Ereignisstatusregister ein. Sie erkennem an diesem Eintrag aber nicht, wann und wie oft dieser Fehler aufgetreten ist.

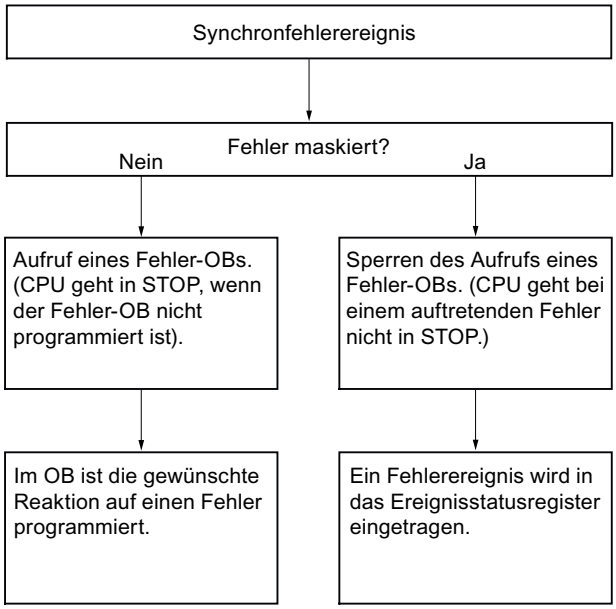

Die **Synchronfehlerereignisse** sind einem bestimmten Bitmuster zugeordnet, der **Fehlermaske**. Diese Fehlermaske finden Sie in den Eingangs- und Ausgangsparametern der Anweisungen "MSK_FLT (Seite 2587)", "DMSK_FLT (Seite 2588)", "READ_ERR (Seite 2589)" wieder.

Bei den Synchronfehlerereignissen wird zwischen **Programmier- und Zugriffsfehlern** unterschieden, die Sie in zwei Fehlermasken maskieren können. Die jeweiligen Fehlermasken finden Sie in den folgenden Bildern.

Programmierfehlermaske

Das folgende Bild zeigt Ihnen das Bitmuster der Fehlermaske für Programmierfehler. Die Fehlermaske für Programmierfehler steht in den Parametern "PRGFLT_..." (siehe im Folgenden "Programmierfehler, Low-Word" bzw. "Programmierfehler, High-Word").

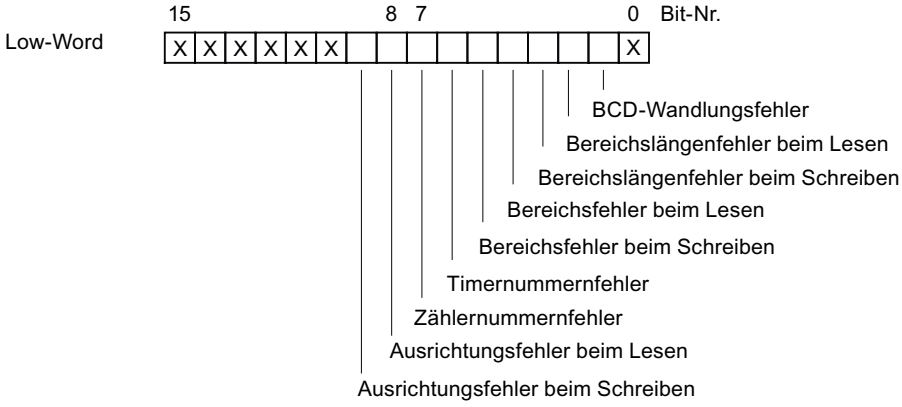

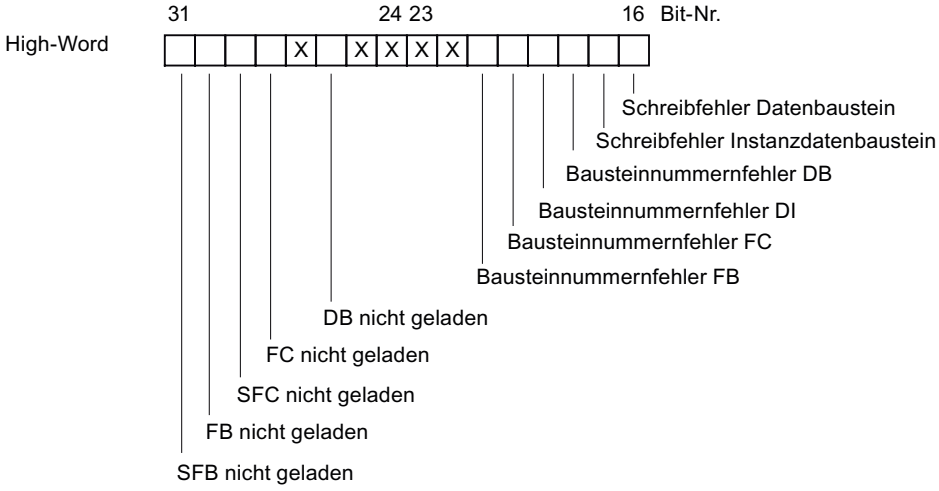

Legende: X nicht relevant

Die nicht relevanten Bits "x" werden für die Ein- und Ausgangsparameter der Anweisungen "MSK_FLT (Seite 2587)", "DMSK_FLT (Seite 2588)", "READ_ERR (Seite 2589)" auf "0" gesetzt.

Zugriffsfehlermaske

Das folgende Bild zeigt Ihnen das Bitmuster der Fehlermaske für Zugriffsfehler. Die Fehlermaske für Zugriffsfehler steht in den Parametern ACCFLT_...

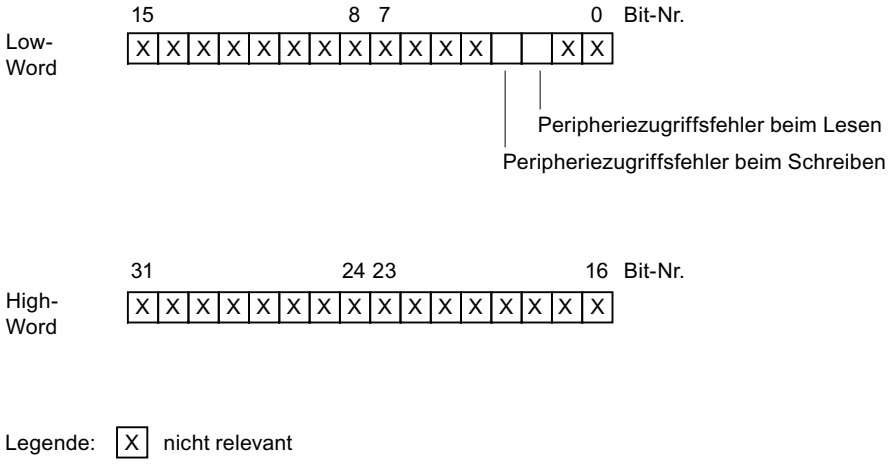

Beispiel: Das folgende Bild zeigt Ihnen, wie das Low-Word der Fehlermaske für Zugriffsfehler mit allen maskierten Fehlern aussieht.

- Als Eingangsparameter für "MSK_FLT (Seite 2587)"
- Als Ausgangsparameter für "MSK_FLT (Seite 2587)"

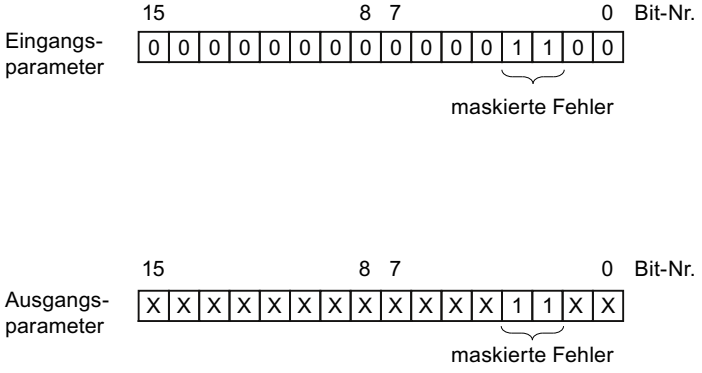

Programmierfehler Low-Word

Die folgende Tabelle enthält die Fehler, die dem Low-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Fehler	Ereignis-ID (W#16#...)	Mögliche Fehlerursache
BCD-Wandlungsfehler	2521	Der zu wandelnde Wert ist keine BCD-Zahl (z. B. 5E8).
Bereichslängenfehler beim Lesen	2522	Ein adressierter Operand liegt nicht vollständig innerhalb des möglichen Operandenbereichs. Beispiel: MW 320 soll gelesen werden, obwohl der Merkerbereich nur 256 Byte groß ist.
Bereichslängenfehler beim Schreiben	2523	Ein adressierter Operand liegt nicht vollständig innerhalb des möglichen Operandenbereichs. Beispiel: Ein Wert soll auf MW 320 geschrieben werden, obwohl der Merkerbereich nur 256 Byte groß ist.
Bereichsfehler beim Lesen	2524	Bei indirekter, bereichsübergreifender Adressierung wurde eine falsche Bereichskennung für den Operanden angegeben. Beispiel: • Richtig: LAR1 P#E 12.0 L W[AR1, P#0.0] • Falsch: LAR1 P#12.0 L W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet.
Bereichsfehler beim Schreiben	2525	Bei indirekter, bereichsübergreifender Adressierung wurde eine falsche Bereichskennung für den Operanden angegeben. Beispiel: • Richtig: LAR1 P#E 12.0 T W[AR1, P#0.0] • Falsch: LAR1 P#12.0 T W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet.
Timernummernfehler (bei S7-1500 nicht implementiert)	2526	Es wird auf einen nicht vorhandenen Timer zugegriffen. Beispiel: • SI T [MW 0] mit MW 0 = 129; der Timer 129 soll gestartet werden, obwohl nur 128 Timer vorhanden sind.
Zählernummernfehler (bei S7-1500 nicht implementiert)	2527	Es wird auf einen nicht vorhandenen Zähler zugegriffen. • Beispiel: ZV Z [MW 0] mit MW 0 = 600; es wird auf den Zähler 600 zugegriffen, obwohl nur 512 Zähler vorhanden sind.

Fehler	Ereignis-ID (W#16#...)	Mögliche Fehlerursache
Ausrichtungsfehler beim Lesen	2528	Es wird ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert. Beispiel: • Richtig: LAR1 P#M12.0 L B[AR1, P#0.0] • Falsch: LAR1 P#M12.4 L B[AR1, P#0.0]
Ausrichtungsfehler beim Schreiben	2529	Es wird ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert. Beispiel: • Richtig: LAR1 P#M12.0 T B[AR1, P#0.0] • Falsch: LAR1 P#M12.4 T B[AR1, P#0.0]

Programmierfehler High-Word

Die folgende Tabelle enthält die Fehler, die dem High-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Fehler	Ereignis-ID (W#16#...)	Mögliche Fehlerursache
Schreibfehler Datenbaustein	2530	Der Datenbaustein, auf den geschrieben wird, ist schreibgeschützt.
Schreibfehler Instanzdatenbaustein	2531	Der Instanzdatenbaustein, auf den geschrieben wird, ist schreibgeschützt.
Bausteinnummernfehler DB	2532	Die Nummer des Datenbausteins ist größer als die größte zulässige Nummer.
Bausteinnummernfehler DI	2533	Die Nummer des Instanzdatenbausteins ist größer als die größte zulässige Nummer.
Bausteinnummernfehler FC	2534	Die Nummer einer aufgerufenen Funktion (FC) ist größer als die größte zulässige Nummer.
Bausteinnummernfehler FB	2535	Die Nummer eines aufgerufenen Funktionsbausteins (FB) ist größer als die größte zulässige Nummer.
DB nicht geladen	253A	Der Datenbaustein ist nicht geladen.
Anweisung nicht geladen	253C bis 253F	Die aufzurufende Anweisung ist nicht geladen.

Zugriffsfehler

Die folgende Tabelle enthält die Fehler, die der Fehlermaske für Zugriffsfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Fehler	Ereignis-ID (W#16#...)	Mögliche Fehlerursache
Peripheriezugriffsfehler beim Lesen	2942	• Der Adresse im Peripheriebereich ist keine Signalbaugruppe zugeordnet. • Der Zugriff auf diesen Peripheriebereich wurde nicht innerhalb der eingestellten Baugruppenüberwachungszeit quittiert (Quittungsverzug).
Peripheriezugriffsfehler beim Schreiben	2943	

MSK_FLT: Synchronfehlerereignisse maskieren

Beschreibung

Die Anweisung nutzen Sie, um die Reaktion der CPU auf Synchronfehlerereignisse zu steuern. Dazu maskieren Sie die jeweiligen Synchronfehlerereignisse (Fehlermasken siehe Synchronfehlerereignisse maskieren (Seite 2581)). Mit dem Aufruf von "MSK_FLT" maskieren Sie die Synchronfehlerereignisse in der aktuellen Prioritätsklasse.

Wenn Sie in den Eingangsparametern einzelne Bits der Synchronfehlermasken auf "1" setzen, dann behalten andere, vorher bereits gesetzte Bits ihren Wert "1". Sie erhalten neue Fehlermasken, die Sie über die Ausgangsparameter auslesen. Die von Ihnen maskierten Synchronfehlerereignisse rufen keinen OB auf, sondern werden nur in einem Ereignisstatusregister eingetragen. Das Ereignisstatusregister lesen Sie mit der Anweisung "READ_ERR (Seite 2589)" aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MSK_FLT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_SET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu maskierende Programmierfehler
ACCFLT_SET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu maskierende Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PRGFLT_MASKED	Output	DWORD	E, A, M, D, L	Maskierte Programmierfehler
ACCFLT_MASKED	Output	DWORD	E, A, M, D, L	Maskierte Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Keiner der Fehler war bereits maskiert.
0001	Mindestens einer der Fehler war bereits maskiert. Trotzdem werden die anderen Fehler maskiert.
-	Allgemeine Fehlerinformation Siehe auch: Fehler-ID lokal abfragen über GetErrorID (Seite 1935)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Synchronfehlerereignisse (Seite 2590).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DMSK_FLT: Synchronfehlerereignisse demaskieren

Beschreibung

Die Anweisung nutzen Sie, um die mit "MSK_FLT (Seite 2587)" maskierten Fehlerereignisse zu demaskieren. Dazu müssen Sie in den Eingangsparametern die entsprechenden Bits der Fehlermasken auf "1" setzen. Mit dem Aufruf von "DMSK_FLT" demaskieren Sie die entsprechenden Synchronfehlerereignisse der aktuellen Prioritätsklasse. Gleichzeitig werden die abgefragten Einträge im Ereignisstatusregister gelöscht. Die neuen Fehlermasken lesen Sie über die Ausgangsparameter aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DMSK_FLT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_RESET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu demaskierende Programmierfehler
ACCFLT_RESET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu demaskierende Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_MAS-KED	Output	DWORD	E, A, M, D, L	Noch maskierte Programmierfehler
ACCFLT_MAS-KED	Output	DWORD	E, A, M, D, L	Noch maskierte Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Alle angegebenen Fehler wurden demaskiert.
0001	Mindestens einer der Fehler war nicht maskiert. Trotzdem werden die anderen Fehler demaskiert.
-	Allgemeine Fehlerinformation Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Synchronfehlerereignisse (Seite 2590).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

READ_ERR: Ereignisstatusregister auslesen

Beschreibung

Mit der Anweisung lesen Sie das Ereignisstatusregister aus. Das Ereignisstatusregister entspricht in seinem Aufbau den Programmier- und Zugriffsfehlermasken. Diesen können Sie mit "MSK_FLT (Seite 2587)" und "DMSK_FLT (Seite 2588)" als Eingangsparameter programmieren.

In die Eingangsparameter tragen Sie ein, welche Synchronfehlerereignisse Sie aus dem Ereignisstatusregister abfragen wollen. Mit dem Aufruf von "READ_ERR" lesen Sie die gewünschten Einträge aus dem Ereignisstatusregister aus und löschen gleichzeitig diese Einträge.

Aus dem Ereignisstatusregister entnehmen Sie die Information, welche der maskierten Synchronfehler in der aktuellen Prioritätsklasse mindestens einmal aufgetreten sind. Ein gesetztes Bit bedeutet, dass der zugehörige maskierte Synchronfehler mindestens einmal aufgetreten ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_ERR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_QUERY	Input	DWORD	E, A, M, D, L oder Konstante	Abfrage der Programmierfehler
ACCFLT_QUERY	Input	DWORD	E, A, M, D, L oder Konstante	Abfrage der Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PRGFLT_CLR	Output	DWORD	E, A, M, D, L	Aufgetretene Programmierfehler
ACCFLT_CLR	Output	DWORD	E, A, M, D, L	Aufgetretene Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Alle abgefragten Fehler sind maskiert.
0001	Mindestens einer der abgefragten Fehler ist nicht maskiert.
-	Allgemeine Fehlerinformation Siehe auch: Fehler-ID lokal abfragen über GetErrorID (Seite 1935)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Synchronfehlerereignisse (Seite 2590).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Programmbeispiel für Synchronfehlerereignisse

Einleitung

In dem folgenden Beispiel maskieren Sie Synchronfehlerereignisse (Programmfehler) mit Hilfe der Anweisung "MSK_FLT", demaskieren einen Teil der Synchronfehlerereignisse mit Hilfe der Anweisung "DMSK_FLT" und lesen das Ereignisstatusregister mit Hilfe der Anweisung "READ_ERR" aus. Wenn der Programmfehler unmaskiert ist, rufen Sie einen Programm-Fehler-OB auf.

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Strukturen und Variablen an.

- Variablen für MSK_FLT:

SLI_gDB_MSK_FLT			
	Name	Data type	Start value
1	Static		
2	maskedProgErrs	DWord	16#0
3	maskedAccessErrs	DWord	16#0
4	MSK_FLT	Struct	
5	execute	Bool	false
6	progErrMask_Details	Struct	
7	accessErrMask_Details	Struct	
8	progErrSetMask	DWord	16#0
9	accessErrSetMask	DWord	16#0
10	returnValue	Int	0

- MSK_FLT / Inhalt der Struktur "progErrMask_Details":

6	progErrMask_Details	Struct	
7	BCD_conversion_error	Bool	false
8	Area_length_error_when_reading	Bool	1
9	Area_length_error_when_writing	Bool	1
10	Range_error_when_reading	Bool	1
11	Range_error_when_writing	Bool	1

Den vollständigen Aufbau der Struktur finden Sie hier: Synchronfehlerereignisse maskieren (Seite 2581).

- MSK_FLT / Inhalt der Struktur "accessErrMask_Details":

7	accessErrMask_Details	Struct	
8	IO_access_error_when_reading	Bool	false
9	IO_access_error_when_writing	Bool	false

- Variablen für DMSK_FLT:

11	DMSK_FLT	Struct	
12	execute	Bool	false
13	progErrResetMask	DWord	16#0000_000C
14	accessErrResetMask	DWord	16#0
15	returnValue	Int	0

- Variablen für READ_ERR:

16	READ_ERR	Struct	
17	busyReading	Bool	false
18	peOB_unmasked_active	Bool	false
19	returnValue	Int	0
20	occuredMskProgErrs	DWord	16#0
21	occuredMskAccessErrs	DWord	16#0
22	occProgErrMask_Details	Struct	
23	occAccErrMask_Details	Struct	

Der Aufbau der Strukturen "occProgErrMask_Details" und "occAccErrMask_Details" gleicht dem Aufbau der Strukturen "progErrMask_Details" und "accessErrMask_Details".

- Variablen für das Testereignis:

24	testErrorEvent	Struct	
25	index	Int	0
26	field	Array[0..10] of Real	
27	result	Real	0.0
28	startOut	Bool	false

Parameter verschalten: FB "SLI_FB_genErr_SyncErrEv"

Für die Generierung eines Programmfehlers erstellen sie einen FB "SLI_FB_genErr_SyncErrEv".

Darin legen Sie die folgende lokale Variable an.

SLI_FB_genErr_SyncErrEv

	Name	Data type	Default value
1	Input		
2	index[0..10]	Int	0

In dem FB legen Sie die folgenden Verschaltungen an.

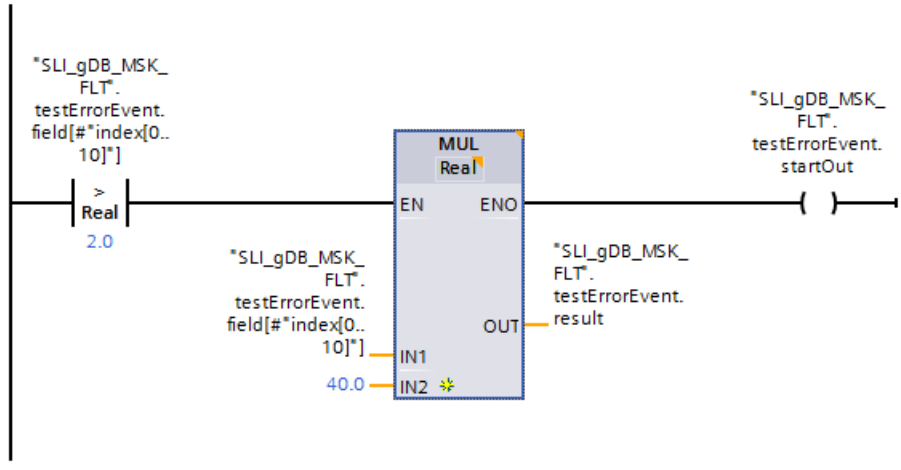

Parameter verschalten: FC "SLI_itemiseDW_SyncErrEv"

Für die Weitergabe einer Bitfolge in der Größe eines DWORD legen Sie den FC "SLI_itemiseDW_SyncErrEv" an.

Darin legen Sie die folgenden lokalen Variablen (Ein- und Ausgänge von 1 bis 31) an.

- Auszug der Eingänge:

SLI_itemiseDW_SyncErrEv		
	Name	Data type
1	Input	
2	in1	Bool
3	in2	Bool
4	in3	Bool
5	in4	Bool
6	in5	Bool
7	in6	Bool
8	in7	Bool
9	in8	Bool

- Auszug der Ausgänge:

21	in31	Bool
22	Output	
23	out1	Bool
24	out2	Bool
25	out3	Bool
26	out4	Bool
27	out5	Bool
28	out6	Bool
29	out7	Bool
30	out8	Bool
39	out31	Bool

In dem FC legen Sie nach Vorbild des Netzwerk 1 für die Ein- und Ausgänge folgende Verschaltungen an.

Netzwerk 1:

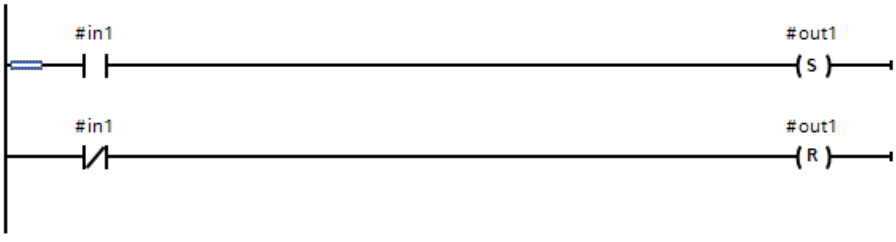

Parameter verschalten: FC "SLI_mask_SyncErrEv"

Für die Konvertierung der zu setzenden Masken für Programm- und Zugriffsfehler legen Sie den FC "SLI_mask_SyncErrEv" an.

Darin legen Sie die folgenden lokalen Variablen an.

3	Output	
4	outSetMaskPE	DWord
5	outSetMaskAE	DWord

Netzwerk 1:

Die Parameter des FC "SLI_itemiseDW_SyncErrEv" verschalten Sie nach folgendem Vorbild.

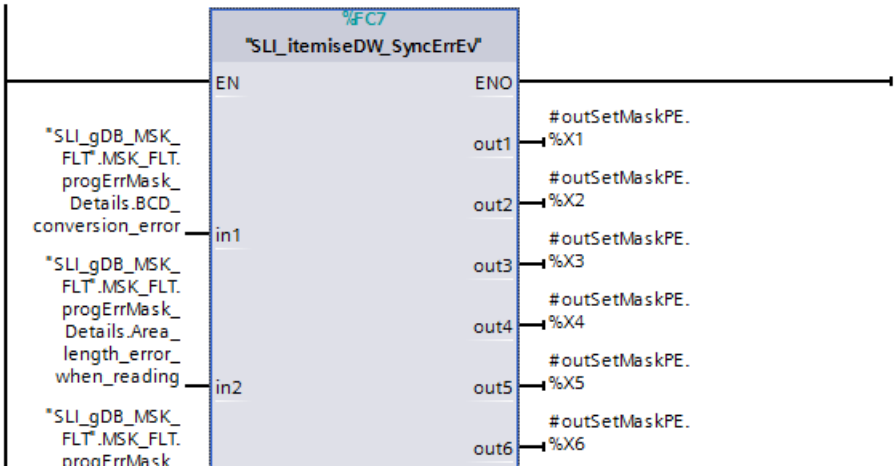

Den vollständigen Aufbau der Struktur der Programmfehler ("progErrMask_Details") finden Sie hier: Synchronfehlerereignisse maskieren (Seite 2581).

Netzwerk 2:

Das Bit 2 des DWORD der Zugriffsfehler verschalten Sie wie folgt.

Netzwerk 3:

Das Bit 3 des DWORD der Zugriffsfehler verschalten Sie wie folgt.

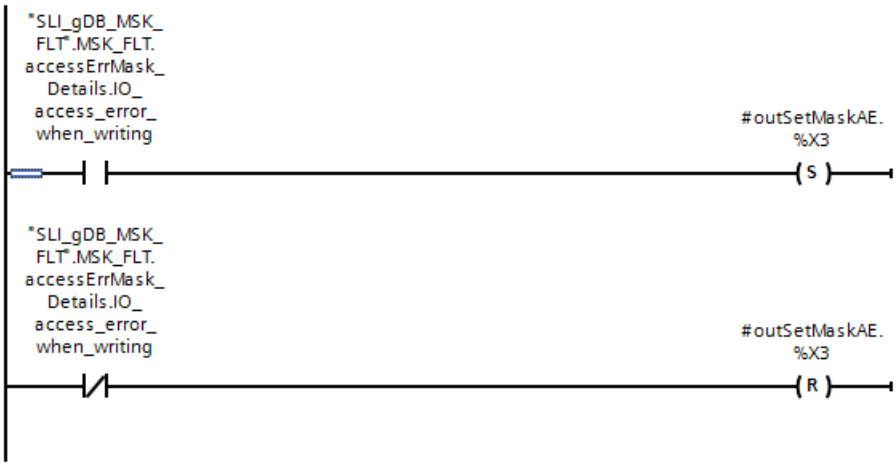

Parameter verschalten: FC "SLI_occured_SyncErrEv"

Für die Konvertierung gelesener Programm- oder Zugriffsfehler legen Sie den FC "SLI_occured_SyncErrEv" an.

Darin legen Sie die folgenden lokalen Variablen an.

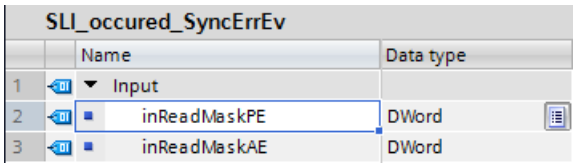

SLI_occured_SyncErrEv

		Name	Data type
1		Input	
2		inReadMaskPE	DWord
3		inReadMaskAE	DWord

Netzwerk 1:

Die Parameter des FC "SLI_itemiseDW_SyncErrEv" verschalten Sie nach folgendem Vorbild.

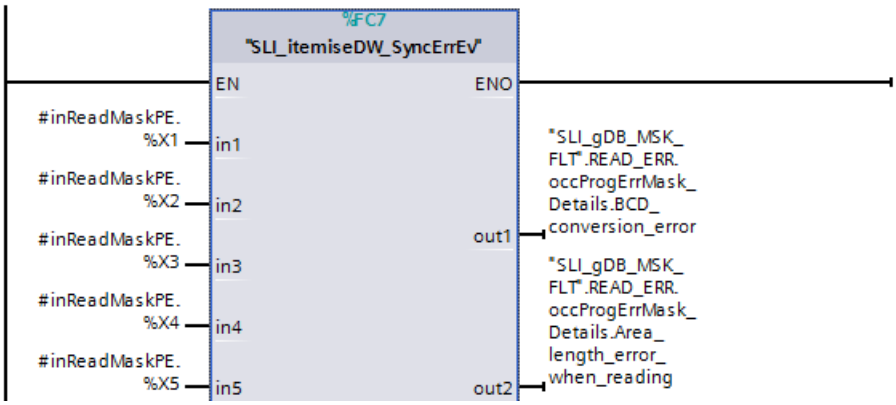

Den vollständigen Aufbau der Struktur der Programmfehler ("occProgErrMask_Details") finden Sie hier: Synchronfehlerereignisse maskieren (Seite 2581).

Netzwerk 2:

Das Bit 2 des DWORD der Zugriffsfehler verschalten Sie wie folgt.

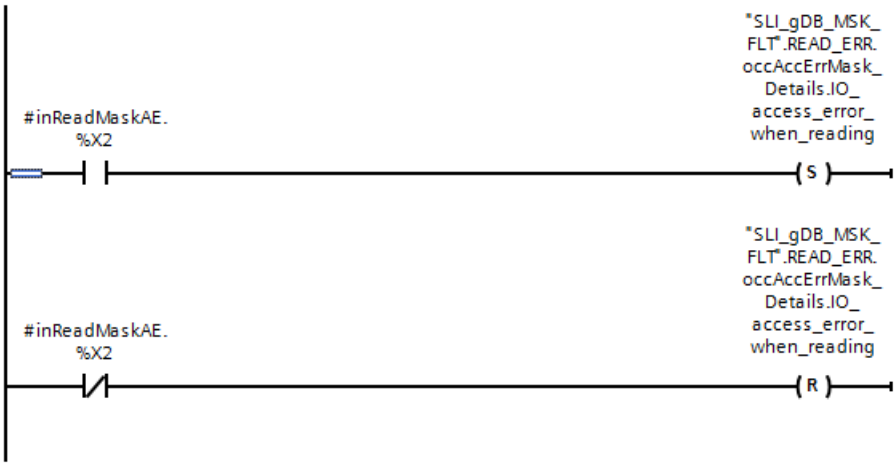

Netzwerk 3:

Das Bit 3 des DWORD der Zugriffsfehler verschalten Sie wie folgt.

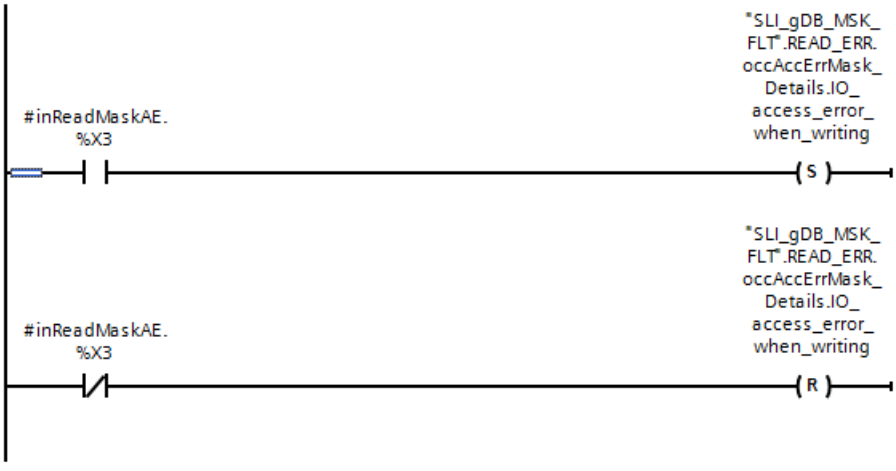

Parameter verschalten: OB "SLI_peOB_SyncErrEV"

Für den Fall eines Programmfehlers im unmaskierten Zustand legen Sie den OB "SLI_peOB_SyncErrEV" an.

Netzwerk 1:

In dem OB legen Sie die folgende Verschaltung an.

Parameter verschalten: FC "SLI_FC_MSK_FLT"

Zum Aufruf der Programmteile legen Sie den FC "SLI_FC_MSK_FLT" an.

Netzwerk 1:

Die Parameter des FC "SLI_mask_SyncErrEv" sowie die Variablen "peOB_unmased_active", "MSK_FLT.execute" und "maskedProgErrs" verschalten Sie wie folgt.

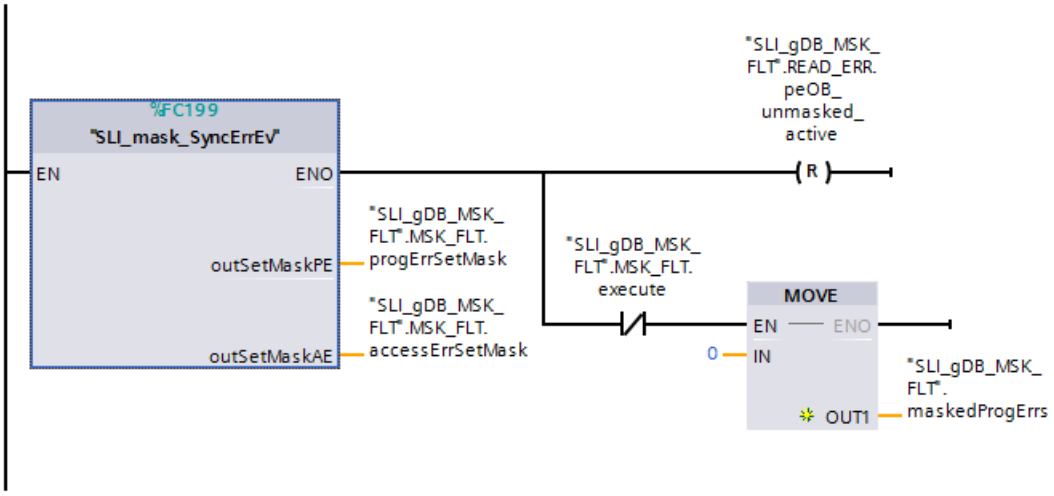

Netzwerk 2:

Die Parameter der Anweisung "MSK_FLT" verschalten Sie wie folgt.

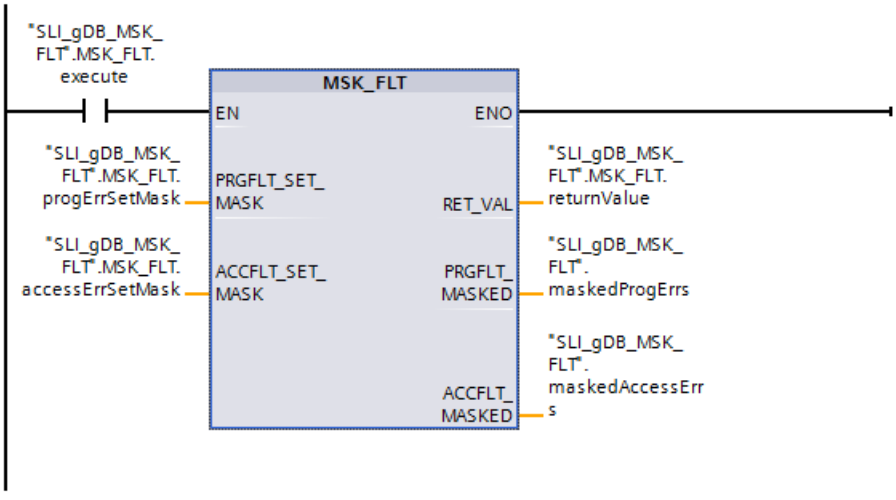

Netzwerk 3:

Die Parameter der Anweisung "DMSK_FLT" verschalten Sie wie folgt.

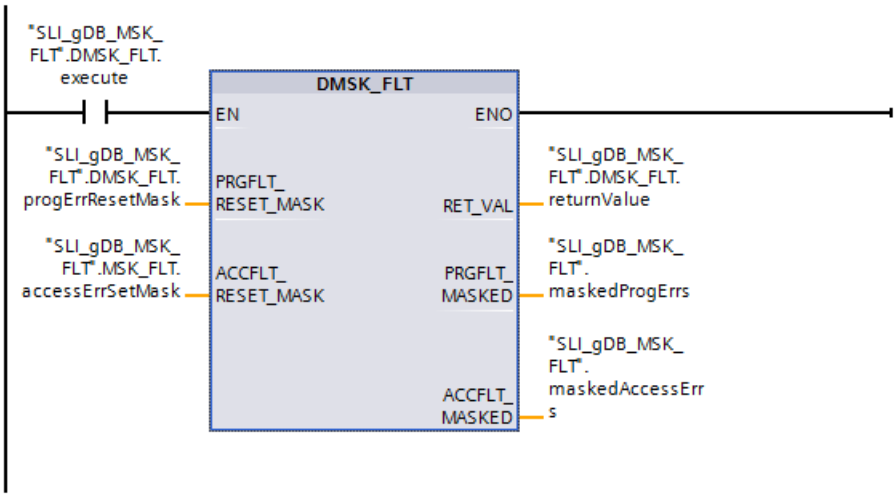

Netzwerk 4:

Die Parameter des FC "SLI_FB_genErr_SyncErrEv" verschalten Sie wie folgt.

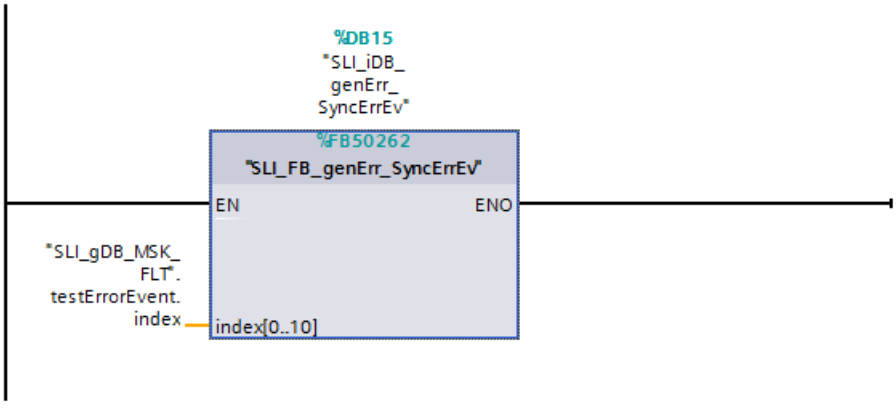

Netzwerk 5:

Die Parameter der Anweisung "READ_ERR" verschalten Sie wie folgt.

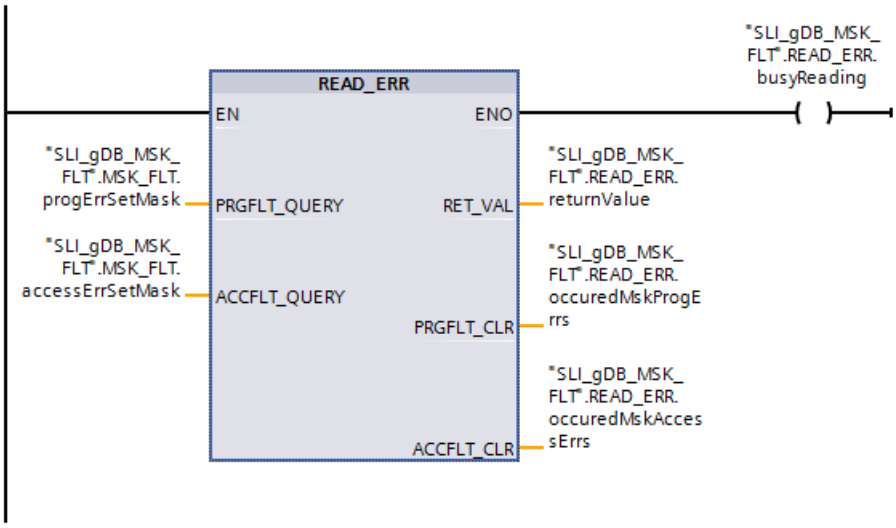

Netzwerk 6:

Die Parameter des FC "SLI_occured_SyncErrEv" verschalten Sie wie folgt.

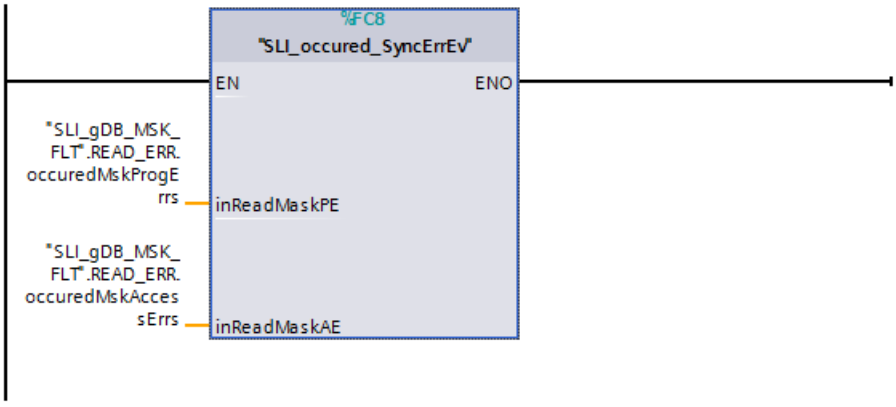

Ergebnis von OB "SLI_peOB_SyncErrEv"

Wenn ein Programmfehler ausgelöst wurde und dieser nicht durch einen Filter maskiert ist, wird der Programmfehler-OB "SLI_peOB_SyncErrEv" ausgeführt.

Im Netzwerk 1 des OBs wird die Variable "READ_ERR.peOB_unmasked_active" auf "TRUE" gesetzt.

Ergebnis von FC "SLI_FC_MSK_FLT"

Netzwerk 1 (FC "SLI_mask_SyncErrEv"):

Mit dem Aufruf des FC "SLI_mask_SyncErrEv" werden die Strukturen für Programmfehler und Zugriffsfehler in den Datentyp DWORD ("progErrSetMask" und "accessErrSetMask") konvertiert.

Da der Wert "TRUE" der Variable "READ_ERR.peOB_unmasked_active" auch noch nach dem Aufruf des Programfehler-OBs angezeigt wird, setzen Sie den Wert zurück. Wenn die

Anweisung "MSK_FLT" nicht ausgeführt wird ("MSK_FLT.execute" hat den Wert "FALSE"), wird die Variable "maskedProgErrs" zurückgesetzt.

Netzwerk 2 ("MSK_FLT"):

Wenn der Schließerkontakt ("MSK_FLT.execute") den Signalzustand "TRUE" liefert, wird die Anweisung "MSK_FLT" ausgefuhrt.

Entsprechend der Variablen "progErrMask_Details" und "accessErrMask_Details" sind die zu steuernden Synchronfehlerereignisse vorgegeben:

- Mit "progErrMask_Details" geben Sie die zu maskierenden Programmfehler an.
- Im Beispiel werden keine Zugriffsfehler ("accessErrMask_Details") maskiert.

Über die Eingangsparameter PRGFLT_SET_MASK ("MSK_FLT.progErrSetMask") und ACCFLT_SET_MASK ("MSK_FLT.accessErrSetMask") sind die zu steuernden Synchronfehlerereignisse im Datentyp DWORD hinterlegt.

Am Ausgangsparameter RET_VAL ("MSK_FLT.returnValue") wird angezeigt, dass keiner der Fehler bereits maskiert war. An den Ausgangsparametern PRGFLT_MASKED ("maskedProgErrs") und ACCFLT_MASKED ("maskedAccessErrs") werden die aktuell maskierten Synchronfehlerereignisse angezeigt.

2	maskedProgErrs	DWord	16#0	16#0000_003C
3	maskedAccessErrs	DWord	16#0	16#0000_0000
4	MSK_FLT	Struct		
5	execute	Bool	false	TRUE
6	progErrMask_Details	Struct		
7	accessErrMask_Details	Struct		
8	progErrSetMask	DWord	16#0	16#0000_003C
9	accessErrSetMask	DWord	16#0	16#0000_0000
10	returnValue	Int	0	0

Netzwerk 3 ("DMSK_FLT"):

Wenn der Schließerkontakt ("DMSK_FLT.execute") den Signalzustand "TRUE" liefert, wird die Anweisung "DMSK_FLT" ausgefuhrt. Über die Eingangsparameter PRGFLT_RESET_MASK ("DMSK_FLT.progErrResetMask") und ACCFLT_RESET_MASK ("DMSK_FLT.accessErrResetMask") sind die zu demaskierenden Synchronfehlerereignisse im Datentyp DWORD hinterlegt.

Am Ausgangsparameter RET_VAL ("DMSK_FLT.returnValue") wird angezeigt, dass alle angegebenen Fehler demaskiert wurden.

11	DMSK_FLT	Struct		
12	execute	Bool	false	TRUE
13	progErrResetMask	DWord	16#0000_000C	16#0000_000C
14	accessErrResetMask	DWord	16#0	16#0000_0000
15	returnValue	Int	0	0

An den Ausgangsparametern PRGFLT_MASKED ("maskedProgErrs") und ACCFLT_MASKED ("maskedAccessErrs") werden die aktuell maskierten Synchronfehlerereignisse angezeigt.

2	maskedProgErrs	DWord	16#0	16#0000_0030
3	maskedAccessErrs	DWord	16#0	16#0000_0000

Netzwerk 4 (FB "SLI_FB_genErr_SyncErrEv"):

Wenn die Variable "testErrorEvent.index" einen Wert größer als "10" liefert, wird ein Programmfehler ausgelöst.

Netzwerk 5 ("READ_ERR"):

Die Anweisung "READ_ERR" liesst das Ereignisstatusregister aus. Über die Eingangsparameter PRGFLT_QUERY ("READ_ERR.progErrSetMask") und ACCFLT_QUERY ("READ_ERR.accessErrSetMask") sind die zu lesenden Synchronfehlerereignisse im Datentyp DWORD hinterlegt.

- Fall 1 - Der Programmfehler wurde nicht ausgelöst. Keine Masken sind gesetzt:
  Am Ausgangsparameter RET_VAL ("READ_ERR.returnValue") wird angezeigt, dass mindestens einer der abgefragten Fehler nicht maskiert ist.
  An den Ausgangsparametern PRGFLT_CLR ("READ_ERR.occuredMskProgErrs") und ACCFLT_CLR ("READ_ERR.occuredMskAccessErrs") werden die aktuell aufgetretenen Fehlerereignisse angezeigt.

16	READ_ERR	Struct		
17	busyReading	Bool	false	TRUE
18	peOB_unmasked_active	Bool	false	FALSE
19	returnValue	Int	0	1
20	occuredMskProgErrs	DWord	16#0	16#0000_0000
21	occuredMskAccessErrs	DWord	16#0	16#0000_0000
22	occProgErrMask_Details	Struct		
23	occAccErrMask_Details	Struct		

- Fall 2 - Der Programmfehler wurde ausgelöst und ist nicht maskiert:
  MSK_FLT, DMSK_FLT sind inaktiv. Der Programmfehler-OB wird ausgeführt.

16	READ_ERR	Struct		
17	busyReading	Bool	false	TRUE
18	peOB_unmasked_active	Bool	false	TRUE
19	returnValue	Int	0	1
20	occuredMskProgErrs	DWord	16#0	16#0000_0000
21	occuredMskAccessErrs	DWord	16#0	16#0000_0000
22	occProgErrMask_Details	Struct		
23	occAccErrMask_Details	Struct		
24	testErrorEvent	Struct		
25	index	Int	0	11
26	field	Array[0..10] of Real		
27	result	Real	0.0	0.0
28	startOut	Bool	false	FALSE

- Fall 3 - Der ausgelöste Programmfehler wird maskiert:
  MSK_FLT ist aktiv. DMSK_FLT und der Programmfehler-OB sind inaktiv.

16	READ_ERR	Struct		
17	busyReading	Bool	false	TRUE
18	peOB_unmasked_active	Bool	false	FALSE
19	returnValue	Int	0	0
20	occuredMskProgErrs	DWord	16#0	16#0000_0004
21	occuredMskAccessErrs	DWord	16#0	16#0000_0000
22	occProgErrMask_Details	Struct		
23	occAccErrMask_Details	Struct		
24	testErrorEvent	Struct		
25	index	Int	0	11
26	field	Array[0..10] of Real		
27	result	Real	0.0	0.0
28	startOut	Bool	false	FALSE

  Gemäß der Struktur "occProgErrMask_Details" ist der folgende Programmfehler aufgetreten:

22	occProgErrMask_Details	Struct		
23	BCD_conversion_error	Bool	false	FALSE
24	Area_length_error_when_reading	Bool	false	TRUE
25	Area_length_error_when_writing	Bool	false	FALSE
26	Range_error_when_reading	Bool	false	FALSE
27	Range_error_when_writing	Bool	false	FALSE

- Fall 4 - Der ausgelöste Programmfehler wird demaskiert:
  MSK_FLT, DMSK_FLT und der Programmfehler-OB sind aktiv.

16	READ_ERR	Struct		
17	busyReading	Bool	false	TRUE
18	peOB_unmasked_active	Bool	false	TRUE
19	returnValue	Int	0	1
20	occuredMskProgErrs	DWord	16#0	16#0000_0000
21	occuredMskAccessErrs	DWord	16#0	16#0000_0000
22	occProgErrMask_Details	Struct		
23	occAccErrMask_Details	Struct		
24	testErrorEvent	Struct		
25	index	Int	0	11
26	field	Array[0..10] of Real		
27	result	Real	0.0	0.0
28	startOut	Bool	false	FALSE

Netzwerk 6 (FC "SLI_occured_SyncErrEv"):

Mit dem Aufruf des FC "SLI_occured_SyncErrEv" werden die gelesenen Programmfehler ("occuredMskProgErrs") und Zugriffsfehler ("occuredMskAccessErrs") in die Strukturen "occProgErrMask_Details" und "occAccErrMask_Details" konvertiert.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

MSK_FLT: Synchronfehlerereignisse maskieren (Seite 2587)

DMSK_FLT: Synchronfehlerereignisse demaskieren (Seite 2588)

READ_ERR: Ereignisstatusregister auslesen (Seite 2589)

Asynchronfehlerereignis

DIS_IRT: Unterbrechungsereignis sperren

Beschreibung

Mit der Anweisung "DIS_IRT" sperren Sie die Bearbeitung neuer Alarm- und Asynchronfehlerereignisse. Sperren heißt, dass das Betriebssystem der CPU bei einem Unterbrechungsereignis

- **weder** einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft,
- **noch** die festgelegte Reaktion bei nicht programmiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

Die Sperre der Bearbeitung von Alarm- und Asynchronfehlerereignissen bleibt über alle Prioritätsklassen gültig. Die Sperre wird nur mit der Anweisung "EN_IRT (Seite 2605)" aufgehoben bzw. auch bei Neustart (Warmstart) und bei Kaltstart.

Hinweis

Beachten Sie, dass bei Programmierung der Anweisung "DIS_IRT" alle eintreffenden Alarme verworfen werden!

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS_IRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Gibt an, welche Alarm- und Asynchronfehlerereignisse gesperrt werden.
OB_NR	Input	INT	E, A, M, D, L oder Konstante	OB-Nummer
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

MODE (B#16#...)	Bedeutung
00	Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden gesperrt (mit Ausnahme des Programmierfehlers und des Fehlers beim direkten Zugriff auf Peripheriedaten und der Motion Control-OBs). Dem Parameter OB_NR weisen Sie den Wert "0" zu. Die Einträge in den Diagnosepuffer erfolgen weiterhin.
01	Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden gesperrt. Die Alarmklasse kennzeichnen Sie wie folgt: • Uhrzeitalarme: 10 • Verzögerungsalarme: 20 • Weckalarme: 30 • Prozessalarme: 40 • Alarme für DPV1: 50 • Taktsynchronalarm: 60 • Redundanzfehleralarme: 70 • Asynchrone Fehleralarme: 80 Die Einträge in den Diagnosepuffer erfolgen weiterhin.
02	Alle neu auftretenden Ereignisse eines angegebenen Alarms werden gesperrt. Den Alarm kennzeichnen Sie durch die OB-Nummer. Die Einträge in den Diagnosepuffer erfolgen weiterhin.

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für DIS_IRT & EN_IRT (Seite 2606).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

EN_IRT: Unterbrechungsereignis freigeben

Beschreibung

Mit der Anweisung geben Sie die mit der Anweisung "DIS_IRT (Seite 2603)" gesperrte Bearbeitung neuer Alarm- und Asynchronfehlerereignisse wieder frei. Freigeben heißt, dass das Betriebssystem der CPU bei einem Unterbrechungsereignis

- einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft
  oder
- die festgelegte Reaktion bei nicht programmiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN_IRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Gibt an, welche Alarm- und Asynchronfehlerereignisse freigegeben werden (siehe unten).
OB_NR	Input	INT	E, A, M, D, L oder Konstante	OB-Nummer
RET_VAL	Return	INT	E, A, M, D, L	Wenn während der Bearbeitung der Anweisung ein Fehler auftritt, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

MODE	Bedeutung
0	Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden freigegeben, mit Ausnahme des Programmierfehlers und des Fehlers beim direkten Zugriff auf Peripheriedaten und der Motion Control-OBs.
1	Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden freigegeben. Die Alarmklasse kennzeichnen Sie wie folgt: • Uhrzeitalarme: 10 • Verzögerungsalarme: 20 • Weckalarme: 30 • Prozessalarme: 40 • Alarme für DPV1: 50 • Taktsynchronalarm: 60 • Redundanzfehleralarme: 70 • Asynchrone Fehleralarme: 80
2	Alle neu auftretenden Ereignisse eines angegebenen Alarms werden freigegeben. Den Alarm kennzeichnen Sie durch die OB-Nummer.

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler aufgetreten.
8090	Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für DIS_IRT & EN_IRT (Seite 2606).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Programmbeispiel für DIS_IRT & EN_IRT

Einleitung

In dem folgenden Beispiel sperren Sie die Bearbeitung eines Weckalarm-OBs (OB33) mit Hilfe der Anweisung "DIS_IRT" und geben den OB33 mit Hilfe der Anweisung "EN_IRT" wieder frei.

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Strukturen und Variablen an.

SLI_gDB_AsyncEvent

	Name	Data type	Start value
1	Static		
2	DIS_IRT	Struct	
3	execute	Bool	false
4	done	Bool	false
5	mode	Byte	16#02
6	obNbr	Int	33
7	returnValue	Int	0
8	EN_IRT	Struct	
9	execute	Bool	false
10	done	Bool	false
11	mode	Byte	16#02
12	obNbr	Int	33
13	returnValue	Int	0
14	testOB	Struct	
15	execute	Bool	false
16	testTime	Time	T#0ms
17	testTimeLimit	Time	T#2m
18	setOut	Bool	false

OB "SLI_ciOB_AsyncEvent"

Sie legen einen OB "SLI_ciOB_AsyncEvent" an und beobachten dessen Verhalten beim Sperren/Entsprerren.

Netzwerk 1: In dem OB rufen Sie die Anweisung "TP" auf und legen die folgenden Verschaltungen an.

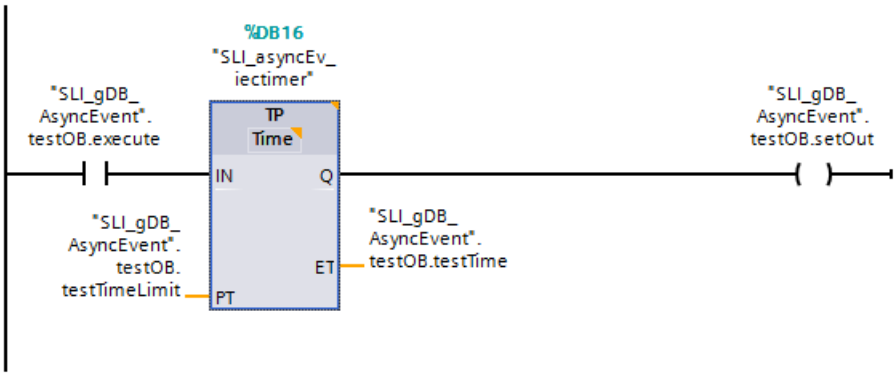

Netzwerk 2: Die Variable "testOB.execute" setzen Sie wie folgt zurück.

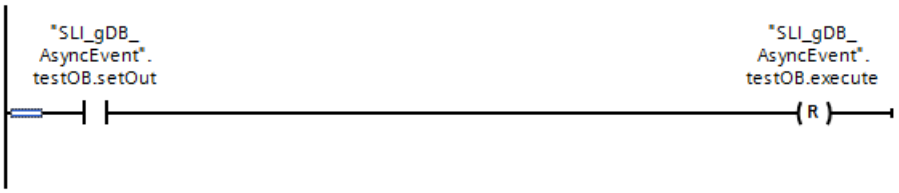

FC "SLI_FC_AsyncEvent"

Zum Aufruf der Programmteile legen Sie den FC "SLI_FC_AsyncEvent" an.

In dem FC legen Sie die folgenden lokalen Variablen an.

7		Temp	
8		tempDIS	Bool
9		tempEN	Bool

Netzwerk 1: Die Parameter der Anweisung "DIS_IRT" verschalten Sie wie folgt.

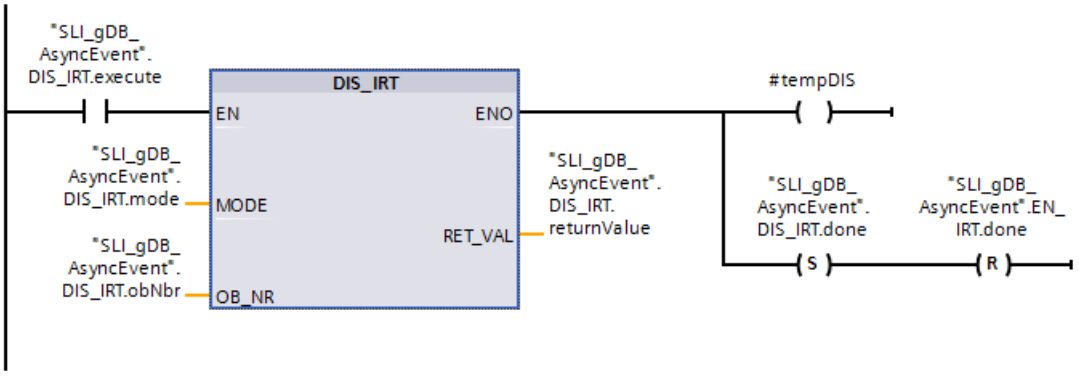

Netzwerk 2: Die Variable "DIS_IRT.execute" setzen Sie wie folgt zurück.

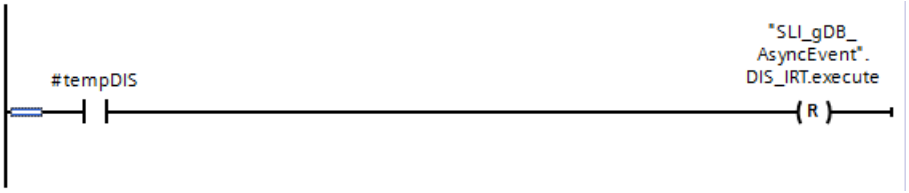

Netzwerk 3: Die Parameter der Anweisung "EN_IRT" verschalten Sie wie folgt.

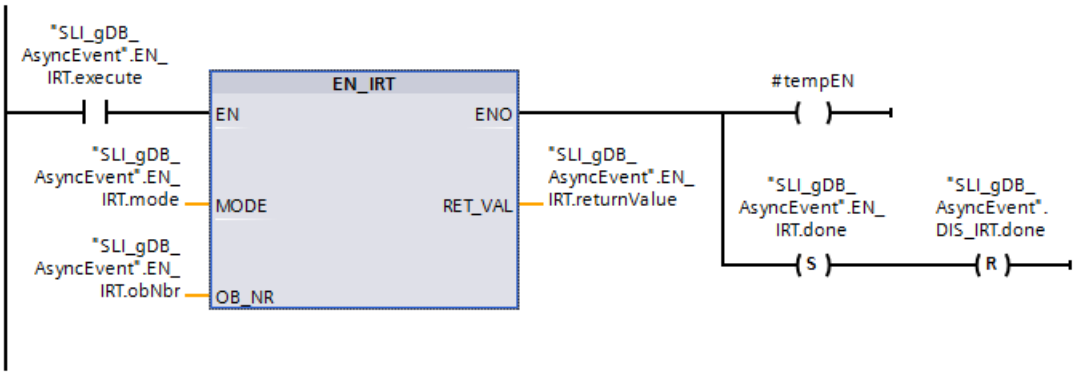

Netzwerk 4: Die Variable "EN_IRT.execute" setzen Sie wie folgt zurück.

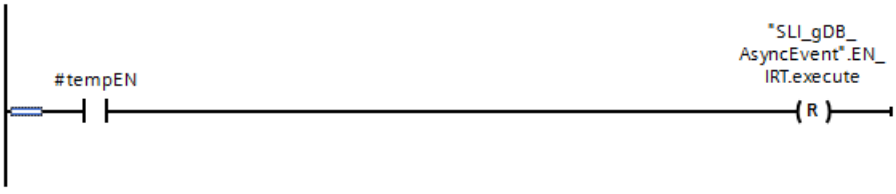

Ergebnis von OB "SLI_ciOB_AsyncEvent"

Wenn der Schließerkontakt ("testOB.execute") einmalig den Signalzustand "TRUE" liefert, wird die Anweisung "TP" ausgeführt. Die Variable "testOB.execute" wird automatisch zurückgesetzt auf Signalzustand "FALSE". Der Wert der Variable "testTime" wird hoch gezählt, bis dieser dem Wert der Variable "testTimeLimit" entspricht.

SLI_gDB_AsyncEvent

	Name	Data type	Start value	Monitor value
1	Static			
2	DIS_IRT	Struct		
3	execute	Bool	false	FALSE
4	done	Bool	false	FALSE
5	mode	Byte	16#02	16#02
6	obNbr	Int	33	33
7	returnValue	Int	0	0
8	EN_IRT	Struct		
9	execute	Bool	false	FALSE
10	done	Bool	false	FALSE
11	mode	Byte	16#02	16#02
12	obNbr	Int	33	33
13	returnValue	Int	0	0
14	testOB	Struct		
15	execute	Bool	false	FALSE
16	testTime	Time	T#0ms	T#19S_100MS
17	testTimeLimit	Time	T#2m	T#2M
18	setOut	Bool	false	TRUE

Ergebnis von FC "SLI_FC_AsyncEvent"

Netzwerk 1:

Wenn der Schließerkontakt ("DIS_IRT.execute") den Signalzustand "TRUE" liefert, wird die Anweisung "DIS_IRT" ausgeführt. Entsprechend dem Eingangsparameter MODE ("DIS_IRT.mode") ist der zu verwendende Betriebsmodus "2" vorgegeben. Gemäß dem Eingangsparameter OB_NR ("DIS_IRT.obNbr") wird der OB33 addressiert und dessen Bearbeitung gesperrt.

Am Ausgangsparameter RET_VAL ("DIS_IRT.returnValue") wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft. Die erfolgte Verarbeitung von "DIS_IRT" speichern Sie in der Variable "DIS_IRT.done". Zudem setzen Sie die Variable "EN_IRT.done" zurück.

Der Wert der Variable "testOB.testTime" wird nicht länger hochgezählt. Nichtsdestotrotz zählt der IEC-Timer der Anweisung "TP" im Hintergrund weiter.

SLI_gDB_AsyncEvent

	Name	Data type	Start value	Monitor value
1	Static			
2	DIS_IRT	Struct		
3	execute	Bool	false	FALSE
4	done	Bool	false	TRUE
5	mode	Byte	16#02	16#02
6	obNbr	Int	33	33
7	returnValue	Int	0	0
8	EN_IRT	Struct		
9	execute	Bool	false	FALSE
10	done	Bool	false	FALSE
11	mode	Byte	16#02	16#02
12	obNbr	Int	33	33
13	returnValue	Int	0	0
14	testOB	Struct		
15	execute	Bool	false	FALSE
16	testTime	Time	T#0ms	T#28S_500MS
17	testTimeLimit	Time	T#2m	T#2M
18	setOut	Bool	false	TRUE

Netzwerk 2:

Wenn der Schließerkontakt ("#tempDIS") den Signalzustand "TRUE" liefert, wird die Variable "DIS_IRT.execute" zurückgesetzt.

Netzwerk 3:

Wenn der Schließerkontakt ("EN_IRT.execute") den Signalzustand "TRUE" liefert, wird die Anweisung "EN_IRT" ausgeführt. Entsprechend dem Eingangsparameter MODE ("EN_IRT.mode") ist der zuverwendende Betriebsmodus "2" vorgegeben. Gemäß dem Eingangsparameter OB_NR ("EN_IRT.obNbr") wird der OB33 addressiert und dessen Bearbeitung wieder entsperrt.

Am Ausgangsparameter RET_VAL ("EN_IRT.returnValue") wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft. Die erfolgte Verarbeitung von "EN_IRT" speichern Sie in der Variable "EN_IRT.done". Zudem setzen Sie die Variable "DIS_IRT.done" zurück.

Der Wert der Variable "testOB.testTime" wird entsprechend dem IEC-Timer der Anweisung "TP" weiter hochgezählt bzw. geändert.

SLI_gDB_AsyncEvent

	Name	Data type	Start value	Monitor value
1	Static			
2	DIS_IRT	Struct		
3	execute	Bool	false	FALSE
4	done	Bool	false	FALSE
5	mode	Byte	16#02	16#02
6	obNbr	Int	33	33
7	returnValue	Int	0	0
8	EN_IRT	Struct		
9	execute	Bool	false	FALSE
10	done	Bool	false	TRUE
11	mode	Byte	16#02	16#02
12	obNbr	Int	33	33
13	returnValue	Int	0	0
14	testOB	Struct		
15	execute	Bool	false	FALSE
16	testTime	Time	T#0ms	T#1M_40S_800MS
17	testTimeLimit	Time	T#2m	T#2M
18	setOut	Bool	false	TRUE

Netzwerk 4:

Wenn der Schließerkontakt ("#tempEN") den Signalzustand "TRUE" liefert, wird die Variable "EN_IRT.execute" zurückgesetzt.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

DIS_IRT: Unterbrechungsereignis sperren (Seite 2603)

EN_IRT: Unterbrechungsereignis freigeben (Seite 2605)

DIS_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen verzögern

Beschreibung

Mit "DIS_AIRT" verzögern Sie die Bearbeitung von Alarm-OBs, deren Priorität höher als die Priorität des aktuellen Organisationsbausteins ist.

Sie können "DIS_AIRT" in einem Organisationsbaustein mehrmals aufrufen. Die Aufrufe von "DIS_AIRT" werden vom Betriebssystem gezählt. Bei jeder Ausführung von "DIS_AIRT" wird die Bearbeitung weiter verzögert. Um eine Verzögerung aufzuheben, müssen Sie die Anweisung "EN_AIRT (Seite 2612)" ausführen. Die Bearbeitungsverzögerung gilt solange, bis

die Anzahl der Ausführungen von "EN_AIRT (Seite 2612)" gleich der Anzahl der Aufrufe von "DIS_AIRT" ist oder der aktuelle OB abgearbeitet ist.

Jeder Aufruf von "DIS_AIRT" erhöht die Anzahl der Verzögerungen; der aktuelle Wert wird Ihnen am Parameter RET_VAL zur Verfügung gestellt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS_AIRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Anzahl der Verzögerungen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Rückgabe-wert	Beschreibung
n	"n" zeigt nach Ablauf der Anweisung die Anzahl der Bearbeitungsverzögerungen, also der Aufrufe von DIS_AIRT an. Benutzen Sie EN_AIRT (Seite 2612), um die Verzögerungen aufzuheben.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für DIS_AIRT & EN_AIRT (Seite 2613).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

EN_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen freigeben

Beschreibung

Mit "EN_AIRT" geben Sie die Bearbeitung von Organisationsbausteinen beim Auftreten von Alarmen frei, die durch die Anweisung "DIS_AIRT (Seite 2611)" verzögert wurde.

Mit einer Ausführung von "EN_AIRT" heben Sie eine Bearbeitungsverzögerung auf, die bei einem Aufruf von "DIS_AIRT (Seite 2611)" vom Betriebssystem registriert wurde. Zum Aufheben aller Verzögerungen muss die Anzahl der Ausführungen von "EN_AIRT" gleich der Anzahl der Aufrufe von "DIS_AIRT (Seite 2611)" sein. Beispiel: Sie haben "DIS_AIRT (Seite 2611)" fünf Mal aufgerufen. Dadurch wird die Bearbeitung auch fünf Mal verzögert. Dann müssen Sie die Anweisung "EN_AIRT" fünf Mal ausführen, um alle fünf Verzögerungen aufzuheben.

Am Parameter RET_VAL der Anweisung "EN_AIRT" fragen Sie die Anzahl der Alarmverzögerungen ab, die nach der Ausführung von "EN_AIRT" noch nicht frei gegeben sind. Der Wert "0" am Parameter RET_VAL bedeutet, dass alle durch "DIS_AIRT (Seite 2611)" aktivierten Verzögerungen aufgehoben sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN_AIRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Anzahl der noch programmierten Verzögerungen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Rückgabewert/ Fehlercode* (W#16#...)	Beschreibung
n	"n" zeigt nach Ablauf der Anweisung die Anzahl der noch nicht freigegebenen Bearbeitungsverzögerungen an. Wenn n = 0 ist, ist die Alarmbearbeitung erst wieder freigegeben.
8080	Obwohl die Alarmbearbeitung bereits freigegeben war, wurde die Anweisung aufgerufen.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für DIS_AIRT & EN_AIRT (Seite 2613).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Programmbeispiel für DIS_AIRT & EN_AIRT

Einleitung

In dem folgenden Beispiel verzögern Sie die Verarbeitung eines Weckalarm-OBs bzw. von höher-prioren Alarm-OBs mit Hilfe der Anweisungen "DIS_AIRT" und "EN_AIRT". Durch den Aufruf der Anweisungen wird ein Testprogramm nicht durch einen Alarm unterbrochen.

Über das Testprogramm speichern Sie Zeit- und Diagnosedaten in einem Datenspeicher.

- In einem Weckalarm-OB lesen Sie dazu die CPU-Zeit über die Anweisung "TIME_TCK" und Diagnosedaten über die Anweisung "GET_DIAG" aus.
- In einem Zyklus-OB (OB1) lesen Sie zudem die Systemzeit der CPU-Uhr über die Anweisung "RD_SYS_T" aus.

Ablage der Daten

Für den Datenspeicher legen Sie den PLC-Datentyp "testAIRT" an:

testAIRT

		Name	Data type	Default value
1		errorHW	Bool	false
2		outCPUtimer	Time	T#0ms
3		outputTIME	Date_And_Time	DT#1990-01-01-0

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Strukturen und Variablen an.

- Für "DIS_AIRT", "EN_AIRT" und die Zeiterfassung:

SLI_gDB_delayAIRT			
	Name	Data type	Start value
1	Static		
2	startOB1	Bool	false
3	startOB_ci	Bool	false
4	delay	Struct	
5	executeDel	Bool	false
6	executeEn	Bool	false
7	nbrOfDelays	Int	0
8	timeData	Struct	
9	retValRDsys	Int	0
10	outCPUtimer	Time	T#0ms

- Für die Diagnose:

Hinweis

HW_Kennung ("laddr") finden

Die HW-Kennung finden Sie über "PLC-Variablen > Systemkonstanten". Für eine CPU S7-1500: Suchen Sie nach "Local~Common", mit dem Datentyp "HW_Submodule". Die zugehörige Zelle "Wert" enthält die HW-Kennung.

11	diag	Struct	
12	mode	UInt	2
13	laddr	HW_ANY	50
14	returnValue	Int	0
15	countDiagDetails	UInt	0
16	getDIAG	DNN	
17	SubordinateState	UInt	0
18	SubordinateIOState	Word	16#0
19	DNNmode	Word	16#0
20	itemiseIOState	Struct	
21	No_maintenance_required	Bool	false
22	The module_or_device_is_disabled	Bool	false
23	Maintenance_required	Bool	false
24	Maintenance_demanded	Bool	false
25	Error	Bool	false
26	The_module_or_the_device_can_not_be_reached_from_the_CPU	Bool	false
27	Inputs_or_outputs_are_not_available	Bool	false

- Für den Datenspeicher:

28	dataStorage	Array[0..3] of "testAIRT"	
29	pos	Int	0

Parameter verschalten: im FC "SLI_FC_itemise_IOState_AIRT"

Zum weiteren Auswerten der Diagnosedaten der Anweisung "GET_DIAG" legen Sie einen FC "SLI_FC_itemise_IOState_AIRT" an.

In dem FC legen Sie die folgenden lokalen Variablen an.

SLI_FC_itemise_IOState_AIRT		
	Name	Data type
1	Input	
2	ComponentStateDetail	Word
3	Output	
4	_____No_maintenance_required	Bool
5	_____The module_or_device_is_disabled	Bool
6	_____Maintenance_required	Bool
7	_____Maintenance_demanded	Bool
8	Error	Bool
9	_____The_module_or_the_device_can_not_be_reached_from_the_CPU	Bool
10	_____Inputs_or_outputs_are_not_available	Bool

Netzwerk 1: Sie legen die folgenden Verschaltungen an.

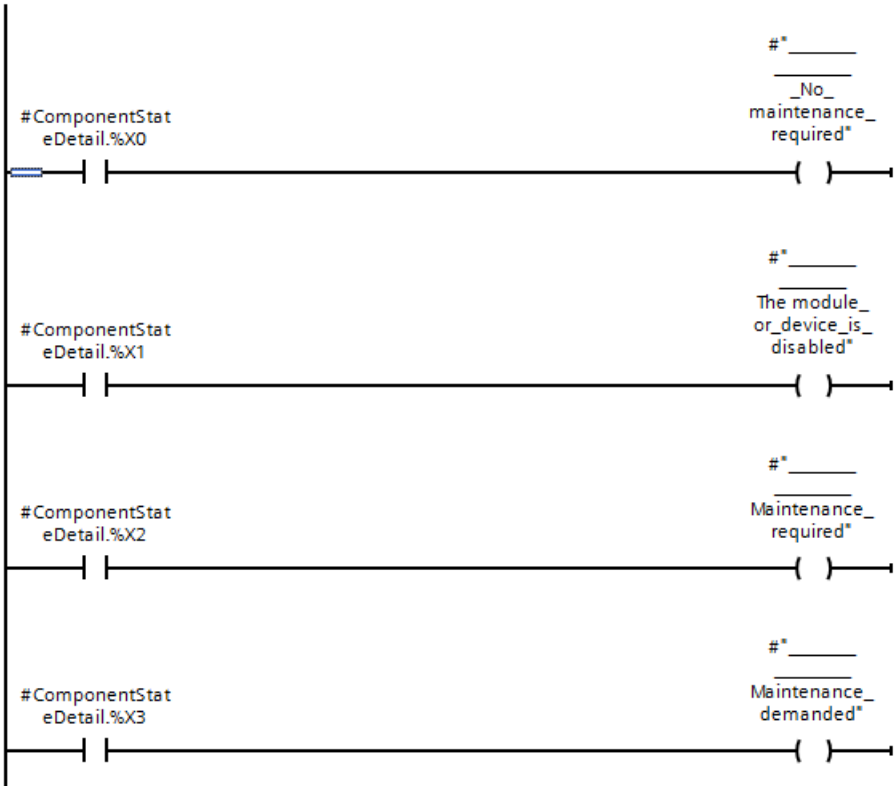

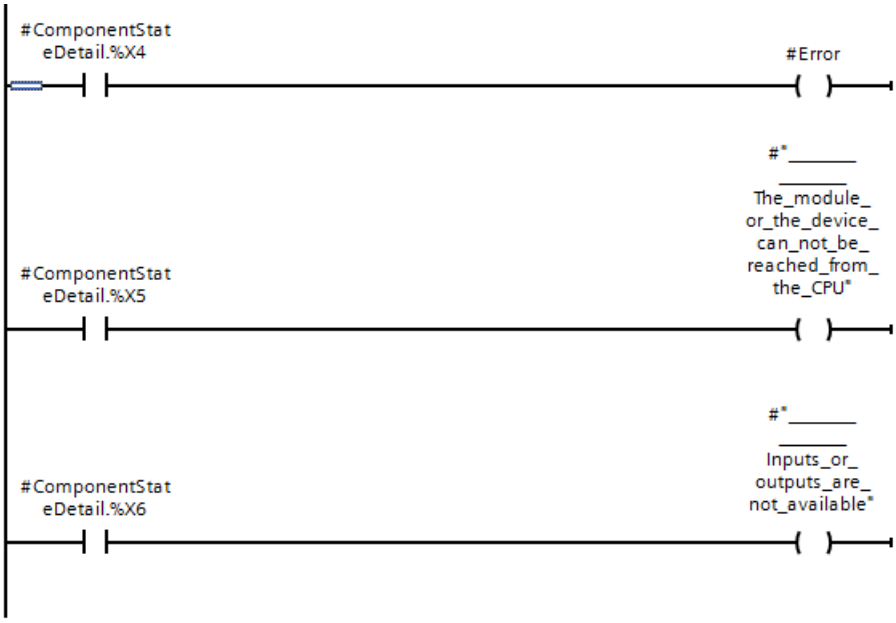

Parameter verschalten: im OB "SLI_ciOB_diagProg_AIRT"

Sie legen einen Weckalarm-OB "SLI_ciOB_diagProg_AIRT" an und beobachten dessen Verhalten beim Verzögern.

In dem OB legen Sie die folgende lokale Variable an.

Netzwerk 1: In dem OB rufen Sie die Anweisung "GET_DIAG" auf und legen die folgenden Verschaltungen an.

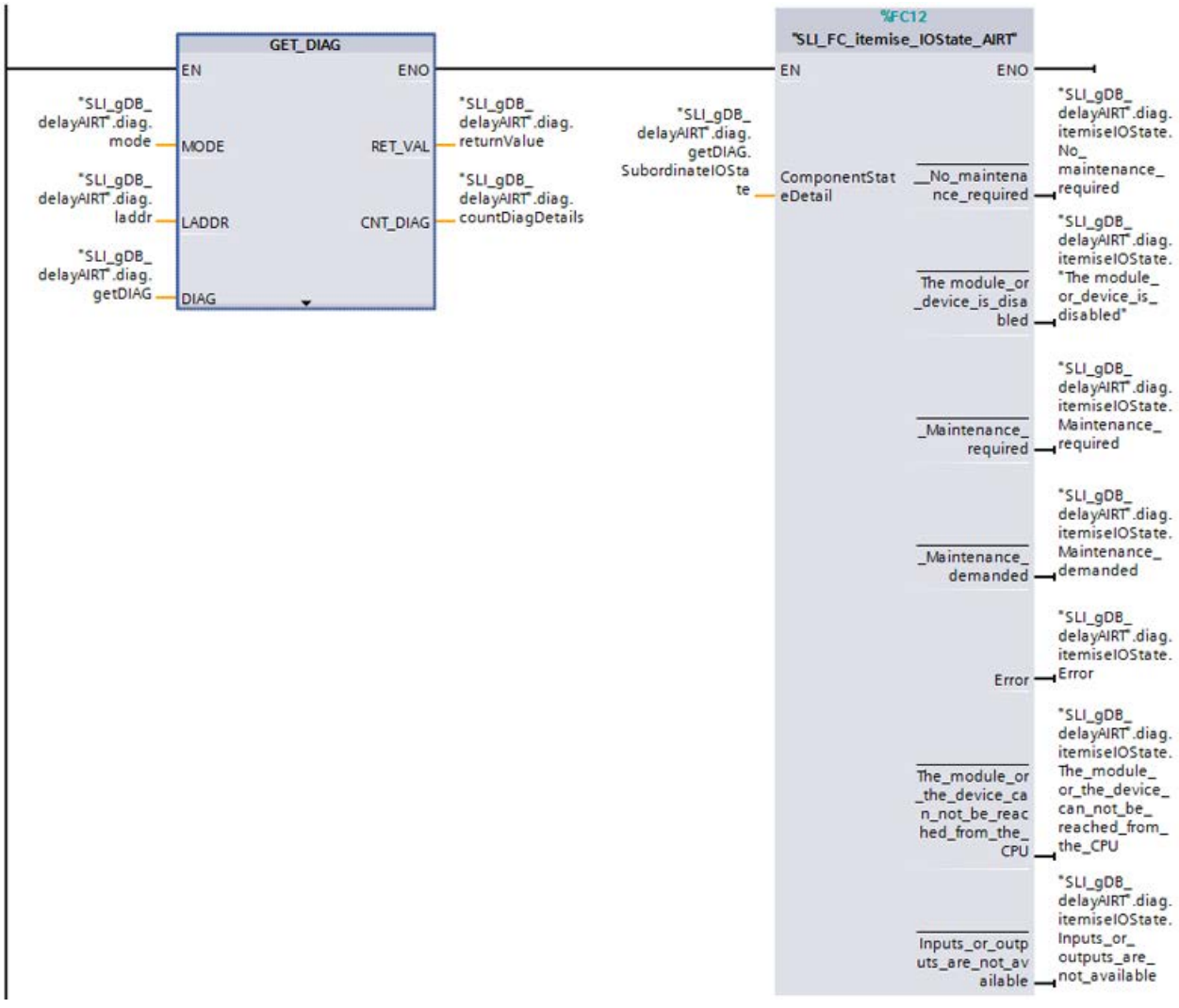

Netzwerk 2: Sie erfassen die CPU-Zeit wie folgt und inkrementieren die Variable "pos" nach der folgenden Bedingung.

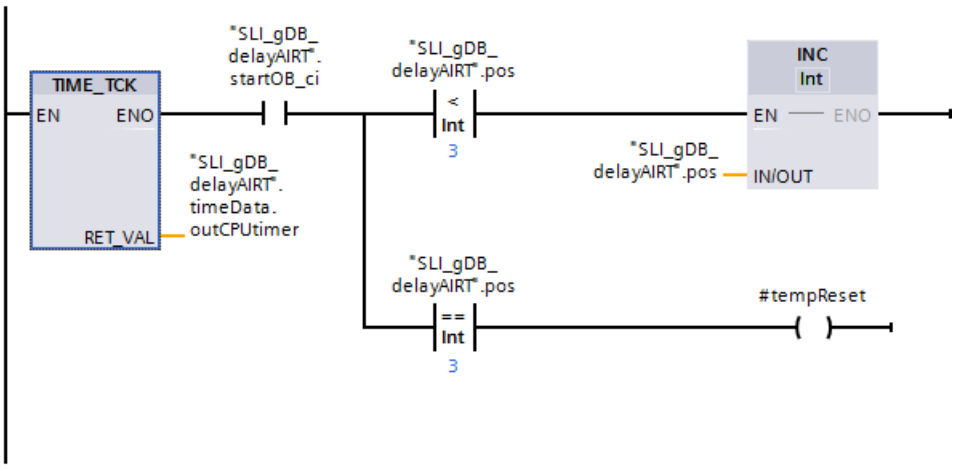

Netzwerk 3: Die Variable "startOB_ci" setzen Sie wie folgt zurück.

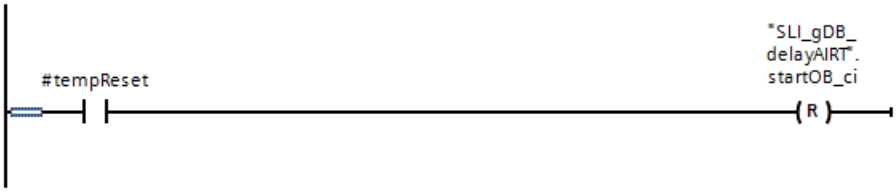

Parameter verschalten: im FC "SLI_FC_saveDiag_AIRT"

Zum Auslesen der Systemzeit und zum Zusammenführen von Zeit- und Diagnosedaten legen Sie den FC "SLI_FC_saveDiag_AIRT" an.

In dem FC legen Sie die folgenden lokalen Variablen an.

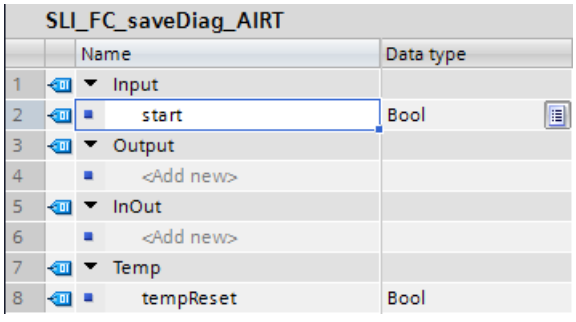

SLI_FC_saveDiag_AIRT

		Name	Data type
1		Input	
2		start	Bool
3		Output	
4		<Add new>	
5		InOut	
6		<Add new>	
7		Temp	
8		tempReset	Bool

Netzwerk 1: Sie legen die folgenden Verschaltungen an.

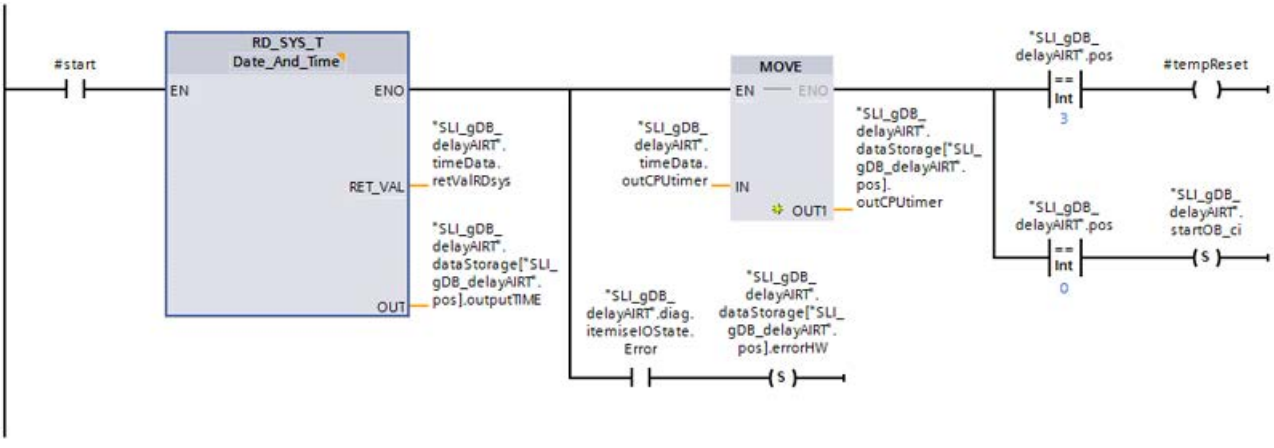

Netzwerk 2: Die Variable "startOB1" setzen Sie wie folgt zurück.

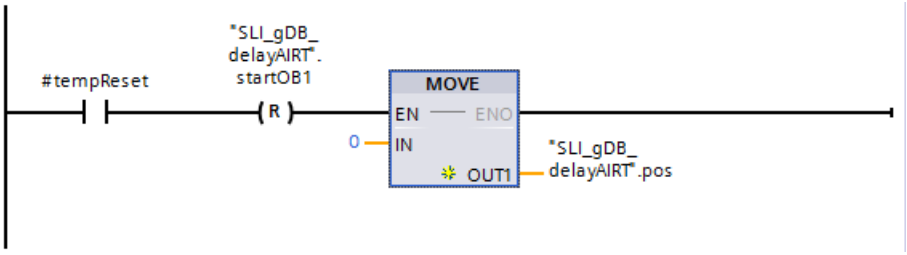

Parameter verschalten: im FC "SLI_FC_delayAIRT"

Zum Aufruf des Testprogramms legen Sie den FC "SLI_FC_delayAIRT" an.

Netzwerk 1: Die Parameter der Anweisung "DIS_AIRT" verschalten Sie wie folgt.

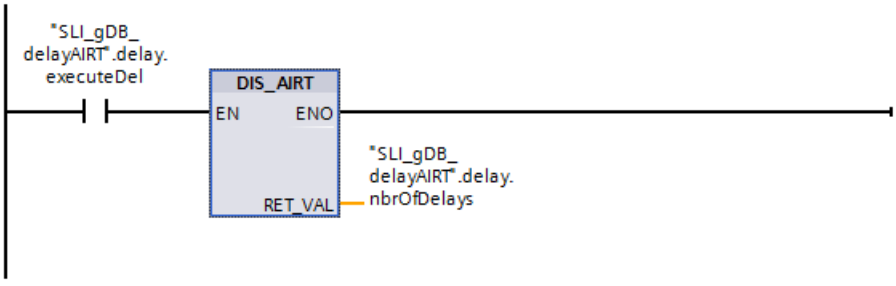

Netzwerk 2: Den FC "SLI_FC_saveDiag_AIRT" verschalten Sie wie folgt.

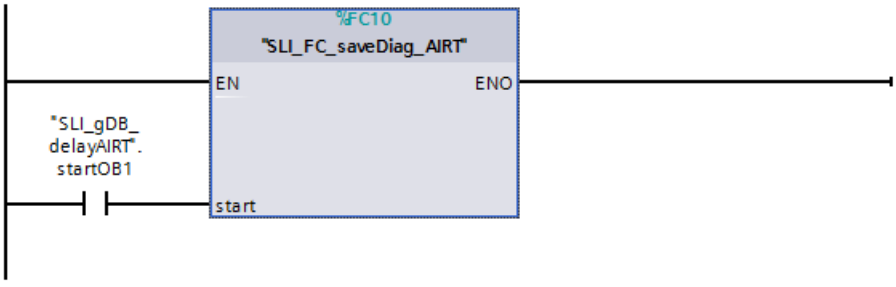

Netzwerk 3: Die Parameter der Anweisung "EN_AIRT" verschalten Sie wie folgt.

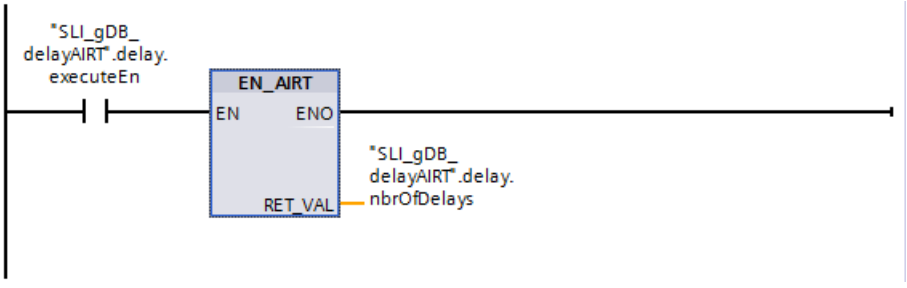

Vorgehen zum Starten des Testszenarios

Um das Testprogramm zu starten, gehen Sie wie folgt vor:

Hinweis

Wenn Sie die Bearbeitung des Weckalarm-OB ("SLI_ciOB_diagProg_AIRT") nicht verzögert ausführen wollen, überspringen Sie Schritt 1 und 2.

1. Setzen Sie die Variable "delay.executeDel" auf "TRUE".
2. Setzen Sie die Variable "delay.executeEn" auf "TRUE".
3. Setzen Sie die Variable "startOB1" auf "TRUE". Nachfolgend wird die Variable "startOB_ci" automatisch gesetzt.
   Sowohl die Variable "startOB1" als auch die Variable "startOB_ci" wird nach der Befüllung des Datenspeichers automatisch zurückgesetzt.

Vorgehen zum Beenden der Bearbeitung von DIS_AIRT und EN_AIRT

Um die verzögerte Bearbeitung des Weckalarm-OBs ("SLI_ciOB_diagProg_AIRT") zu beenden, gehen Sie wie folgt vor:

1. Setzen Sie die Variable "delay.executeEn" auf "FALSE".
2. Setzen Sie die Variable "delay.executeDel" auf "FALSE".

Ergebnis von OB "SLI_ciOB_diagProg_AIRT"

Der Weckalarm-OB ("SLI_ciOB_diagProg_AIRT") wird alle 100 ms aufgerufen und liest die über die Anweisung "GET_DIAG" Diagnosedaten aus und gibt die Zeit der CPU aus.

Wenn der Schließerkontakt ("startOB_ci") den Signalzustand "TRUE" und die Variable "pos" einen Wert größer als "3" liefert, wird die Variable "pos" inkrementiert. Wenn die Variable "pos" den Wert "3" liefert, wird die Variable "startOB_ci" zurückgesetzt.

SLI_gDB_delayAIRT

	Name	Data type	Start value	Monitor value
1	Static			
2	startOB1	Bool	false	FALSE
3	startOB_ci	Bool	false	FALSE
4	delay	Struct		
5	executeDel	Bool	false	FALSE
6	executeEn	Bool	false	FALSE
7	nbrOfDelays	Int	0	0
8	timeData	Struct		
9	retValRDsys	Int	0	0
10	outCPUtimer	Time	T#0ms	T#5M_26S_588MS

Wenn der Weckalarm-OB ("SLI_ciOB_diagProg_AIRT") nicht verzögert ausgeführt wird, sehen die Werte des Datenspeichers etwa wie folgt aus.

28	dataStorage	Array[0...3] o...		
29	dataStorage[0]	"testAIRT"		
30	errorHW	Bool	false	FALSE
31	outCPUtimer	Time	T#0ms	T#2M_9S_388MS
32	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-09:58:24.384
33	dataStorage[1]	"testAIRT"		
34	errorHW	Bool	false	FALSE
35	outCPUtimer	Time	T#0ms	T#2M_9S_488MS
36	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-09:58:24.484
37	dataStorage[2]	"testAIRT"		
38	errorHW	Bool	false	FALSE
39	outCPUtimer	Time	T#0ms	T#2M_9S_588MS
40	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-09:58:24.585
41	dataStorage[3]	"testAIRT"		
42	errorHW	Bool	false	FALSE
43	outCPUtimer	Time	T#0ms	T#2M_9S_688MS
44	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-09:58:24.587
45	pos	Int	0	0

Ergebnis von FC "SLI_FC_delayAIRT"

Netzwerk 1:

Wenn der Schließerkontakt ("delay.executeDel") den Signalzustand "TRUE" liefert, wird die Anweisung "DIS_AIRT" ausgeführt. Die Bearbeitung des Weckalarm-OBs ("SLI_ciOB_diagProg_AIRT") bzw. eines höher-prioren Alarm-OBs wird verzögert. Am Ausgangsparameter RET_VAL ("delay.nbrOfDelays") wird angezeigt, dass eine Verzögerung im aufrufenden OB (OB1) aktiv ist.

Netzwerk 2:

Wenn der Schließerkontakt ("startOB1") den Signalzustand "TRUE" liefert, wird das Testprogramm des FCs "SLI_FC_saveDiag_AIRT" ausgeführt.

Im FC "SLI_FC_saveDiag_AIRT":

Über die Anweisung "RD_SYS_T" wird die Systemzeit ausgelesen und in dem Datenspeicher "dataStorage" abgelegt. Die zuverwendende Zelle des Datenspeichers ist durch die Variable "pos" bestimmt.

Ebenso wird dem Datenspeicher die CPU-Zeit ("outCPUtimer") übergeben. Wenn die Variable "diag.itemiseIOState.Error" den Signalzustand "TRUE" liefert, wird in dem Datenspeicher die Variable "errorHW" auf "TRUE" gesetzt.

Nachfolgend wird Folgendes entschieden:

- Wenn "pos" den Wert "0" hat: Der Wert der Variable "startOB_ci" wird auf "TRUE" gesetzt. Dadurch kann die Variable "pos" in dem Weckalarm-OB ("SLI_ciOB_diagProg_AIRT") inkrementiert werden.
- Wenn "pos" den Wert "3" hat: Die Variablen "startOB1" und "pos" werden zurückgesetzt.

Wenn der Weckalarm-OB ("SLI_ciOB_diagProg_AIRT") verzögert ausgeführt wird, sehen die Werte des Datenspeichers etwa wie folgt aus.

28	dataStorage	Array[0..3] o...		
29	dataStorage[0]	"testAIRT"		
30	errorHW	Bool	false	FALSE
31	outCPUtimer	Time	T#0ms	T#25M_52S_688MS
32	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-10:22:07.684
33	dataStorage[1]	"testAIRT"		
34	errorHW	Bool	false	FALSE
35	outCPUtimer	Time	T#0ms	T#25M_52S_788MS
36	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-10:22:07.784
37	dataStorage[2]	"testAIRT"		
38	errorHW	Bool	false	FALSE
39	outCPUtimer	Time	T#0ms	T#25M_52S_888MS
40	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-10:22:07.884
41	dataStorage[3]	"testAIRT"		
42	errorHW	Bool	false	FALSE
43	outCPUtimer	Time	T#0ms	T#25M_52S_988MS
44	outputTIME	Date_And_Time	DT#1990-01-01-0	DT#2017-01-20-10:22:07.887
45	pos	Int	0	0

Netzwerk 3:

Wenn der Schließerkontakt ("delay.executeEn") den Signalzustand "TRUE" liefert, wird die Anweisung "EN_AIRT" ausgeführt. Die Bearbeitung des Weckalarm-OBs ("SLI_ciOB_diagProg_AIRT") bzw. eines höher-prioren Alarm-OBs wird wieder freigegeben. Am Ausgangsparameter RET_VAL ("delay.nbrOfDelays") wird angezeigt, dass nachfolgend keine Verzögerung im aufrufenden OB (OB1) aktiv ist. Die Verarbeitung im Beispiel verläuft ohne Fehler.

SLI_gDB_delayAIRT

	Name	Data type	Start value	Monitor value
1	Static			
2	startOB1	Bool	false	FALSE
3	startOB_ci	Bool	false	FALSE
4	delay	Struct		
5	executeDel	Bool	false	TRUE
6	executeEn	Bool	false	TRUE
7	nbrOfDelays	Int	0	0
8	timeData	Struct		
9	retValRDsys	Int	0	0
10	outCPUtimer	Time	T#0ms	T#27M_36S_988MS

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

DIS_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen verzögern (Seite 2611)

EN_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen freigeben (Seite 2612)

4.1.4.8 Meldungen

Program_Alarm: Programmmeldung mit Begleitwerten erzeugen

Beschreibung

Die Anweisung "Programmmeldung mit Begleitwerten erzeugen" überwacht ein Signal und generiert bei einem Signalwechsel am Parameter SIG eine Programmmeldung (zur Definition siehe auch: Einführung in die Meldungsprojektierung (Seite 7815)). Bei einem Signalwechsel von 0 auf 1 wird eine kommende Programmmeldung generiert und bei einem Signalwechsel von 1 auf 0 eine gehende Programmmeldung. Die Programmmeldung wird jeweils synchron zum Programmablauf ausgelöst.

An die Programmmeldung können Sie an den Parametern SD_i (0 ≤ i ≤10) bis zu zehn Begleitwerte anhängen. Die Begleitwerte werden zum Zeitpunkt des Signalwechsels am Parameter SIG erfasst und der Programmmeldung zugeordnet. Weitere Informationen zum Aufbau von Begleitwerten finden Sie in den folgenden Abschnitten: Begleitwerte in Meldungen einfügen (Seite 7838), Beispiele für Begleitwerte (Seite 7842).

Jede Meldung, sowohl kommende als auch gehende, werden mit einem Zeitstempel versehen:

- Standardmäßig wird hierzu die aktuelle Systemzeit der PLC beim Signalwechsel benutzt (Standardwert am Parameter TIMESTAMP).
- Wenn Sie abweichend hiervon einen Zeitstempel vorgeben wollen, können Sie diesen an den Parameter TIMESTAMP anlegen.
  Der Zeitwert muss hierbei immer in Systemzeit (d. h. UTC) angegeben werden, da diese Zeit bei der anlagenweiten Uhrzeitsynchronisation verwendet wird.
- Falls eine Meldung mit einer Lokalzeit gestempelt werden soll, so muss ein Konvertierungsbaustein vorgeschaltet werden, der die Lokalzeit auf die Systemzeit umrechnet. Nur dadurch ist garantiert, dass die Zeitstempel in der Meldungsanzeige richtig dargestellt werden.

Um wieder die aktuelle Systemzeit der CPU zu verwenden, stellen Sie den Parameter TIMESTAMP auf seinen Standardwert (LDT#1970-01-01-00:00:00.0).

Gesamtgröße der Begleitwerte für Programmmeldungen

Das Meldeverfahren der S7-1500-CPU stellt maximal 512 Bytes für Begleitwerte (SD-Parameter) der Anweisung "Program_Alarm" zur Verfügung.

Auf diese Größe wurde im TIA Portal <= V14 beim Übersetzen lediglich mit einer Warnung hingewiesen. Ab TIA Portal V14 SP1 erfolgt eine strengere Überprüfung. Beim Überschreiten dieser Größe wird bereits beim Übersetzen ein Fehler ausgegeben, um während der Laufzeit Fehlerszenarien auszuschließen. Diese Überprüfung bezieht sich fortan nicht mehr auf die tatsächliche Größe der SD-Parameter zum Zeitpunkt des Aufrufs der Anweisung "Program_Alarm", sondern auf die maximal mögliche Größe der SD-Parameter. Sie müssen sich also im Vorfeld überlegen, mit welcher Länge Sie Ihre zu übergebenden Zeichenfolgen-Variablen deklarieren, da diese besonders viele Bytes in Anspruch nehmen.

Dies ist bei der Migration von Projekten von der Version <= V14 zur Version V14 SP1 Update 1 entsprechend zu berücksichtigen.

Aufruf der Anweisung "Programmmeldung mit Begleitwerten erzeugen"

Rufen Sie die Anweisung nur in einem Funktionsbaustein (FB) auf. Der Baustein wird synchron abgearbeitet. Ein Alarm wird ausgelöst, sobald der Baustein beendet wird. Wenn es während der Bearbeitung zu einem Fehler gekommen ist, dann wird ein Fehlercode ausgegeben.

Nach dem Einfügen der Anweisung in den FB wird im Abschnitt "Static" der Bausteinschnittstelle eine Multiinstanz vom Datentyp "Program_Alarm" angelegt. Wählen Sie den Namen dieser Multiinstanz frei im aufgeblendeten Dialog. Er ist gleichzeitig der Name der Programmmeldung.

Ergänzen Sie anschließend die Parameter der Anweisung entsprechend Ihren Anforderungen (siehe Tabelle "Parameter").

Projektieren der Programmmeldung

Wenn sie den Namen der Programmmeldung im Abschnitt "Static" oder im Netzwerk des FBs anwählen, werden im Fenster "Eigenschaften" die Einstellungen der Programmmeldung angezeigt. Nehmen Sie die Auswahl der Meldeklasse, Priorität, usw. vor und editieren den Meldetext.

Die hier vorgenommenen Einstellungen können auch in der Projektnavigation bearbeitet werden. Öffnen Sie hierzu unter "PLC-Überwachungen & -Meldungen" das Register "Meldungen" und darin das Register "Programmmeldungen". In der Tabelle "Meldungstypen" werden alle bereits angelegten Programmmeldungen angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Programmmeldung mit Begleitwerten erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Das zu überwachende Signal. • Positive Signalflanke: Eine kommende Programmmeldung wird generiert • Negative Signalflanke: Eine gehende Programmmeldung wird generiert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TIMESTAMP	Input	LDT	M, D, L oder Konstante	Dieser Parameter wird dazu verwendet, einen Alarm mit einem Zeitstempel zu versehen, der z. B. von einem dezentral gestempelten Eingangssignal stammt. Der Zeitwert ist immer in Systemzeit (d. h. UTC) anzugeben, da diese Zeit bei der anlagenweiten Uhrzeitsynchronisation verwendet wird. • "Nicht verschaltet" bedeutet, dass die Systemzeit der CPU beim Signalwechsel als Zeitstempel für den Alarm verwendet wird (Voreinstellung). • Die Übergabe einer Systemzeit bedeutet, dass diese beim Signalwechsel als Zeitstempel für den Alarm verwendet wird. Hinweis: Falls ein Alarm mit einer Lokalzeit gestempelt werden soll, so muss ein Konvertierungsbaustein vorgeschaltet werden, der die Lokalzeit auf die Systemzeit umrechnet. Nur dadurch ist garantiert, dass der Zeitstempel in der Alarmanzeige richtig dargestellt wird. Hinweis: Beim Quittieren der Meldung wird als Zeitstempel stets die Systemzeit der PLC verwendet.
SD_i	Input	VARIANT	E, A, M, D, L	i-ter Begleitwert (1 ≤ i ≤ 10 Als Begleitwerte können Sie Binärzahlen, Ganzzahlen, Gleitpunktzahlen oder Zeichenfolgen verwenden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Error	Output	BOOL	E, A, M, D, L	Zustandsparameter Error Error = TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Die mögliche Fehlerursache wird am Parameter Status angezeigt.
Status	Output	WORD	E, A, M, D, L	Zustandsparameter Status Anzeige der Fehlerinformation (siehe "Parameter Error und Status").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter Error und Status

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über die Parameter Error und Status ausgegeben werden.

Error	Status*	Erläuterung
0	0000	Kein Fehler oder die Anweisung wurde nicht bearbeitet, da kein Flankenwechsel am Parameter SIG vorlag.
1	0085	Meldung vom Typ "Nur Information"
1	8001	Ungültige statische Meldungsinformation
1	8002	Keine gültige statische Meldungsinformation
1	8004	Maximale Größe von 512 Bytes für die Begleitwerte der Meldung erreicht.
1	8005	Am Parameter SIG liegt eine positive Signalflanke an und eine Meldung liegt bereits vor, die noch nicht quittiert wurde.
1	8007	Gehende Meldung, der keine kommende Meldung vorausgegangen ist
1	8087	Statische Meldungen deaktiviert
1	8089	Länge der Meldung zu groß
1	80Ax	Ungültiger Wert am Parameter SD_i.
1	80C1	Die CPU generiert derzeit keine Meldungen, da Initialisierungsroutinen ablaufen (Das ist z. B. nach Download im RUN der Fall.). Versuchen Sie es später noch einmal.
1	80C2	Die maximale Anzahl von Meldungen pro Zeiteinheit wurde gesendet. Versuchen Sie es später noch einmal.
1	80C3	Alle dynamischen Meldungs-Instanzen werden verwendet. Versuchen Sie es später noch einmal.
1	80C4	Meldung wird ausgegeben und ist nicht überschreibbar. Versuchen Sie es später noch einmal.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

In dem folgenden Beispiel erzeugen Sie eine Programmmeldung mit Begleitwert zu einem Signalwechsel.

Zur Ablage des zu überwachenden Signalwerts legen Sie in einem globalen Datenbaustein eine Variable an.

Program_Alarm_Tags			
	Name	Data type	Start value
1	Static		
2	CreateAlarm	Bool	false

Die Anweisung wird in einem Funktionsbaustein aufgerufen. Zur Verschaltung der Anweisung legen Sie für den Funktionsbaustein vier Parameter an.

ProgramAlarm			
	Name	Data type	Default value
1	Input		
2	<Add new>		
3	Output		
4	<Add new>		
5	InOut		
6	<Add new>		
7	Static		
8	MyStaticText	String	'MyStaticText'
9	Program_Alarm_1	Program_Alarm	
10	Temp		
11	OUTPUT_Error	Bool	
12	OUTPUT_Status	Word	

Die Parameter der Anweisung verschalten Sie wie folgt.

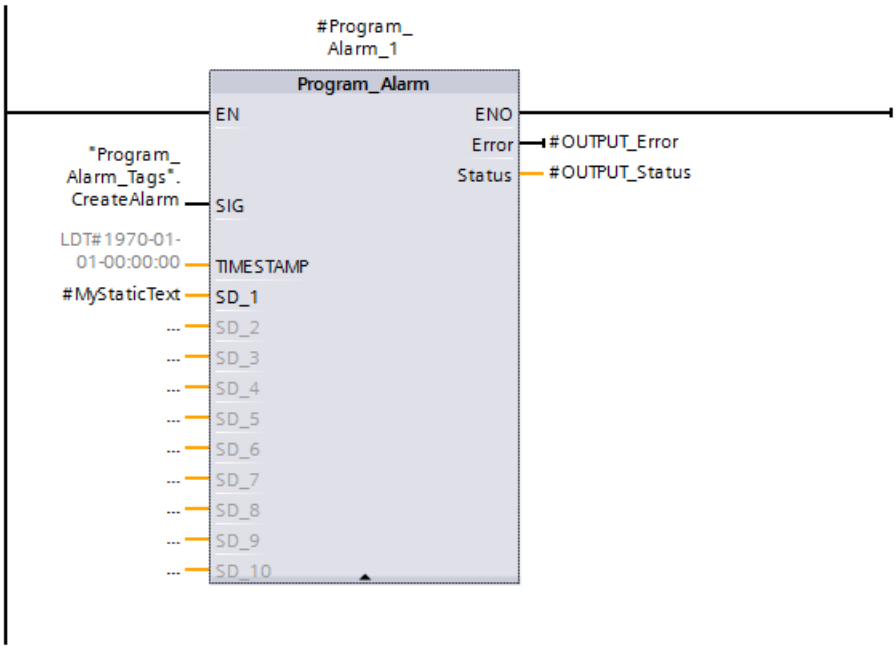

Den Funktionsbaustein rufen Sie in einem OB auf.

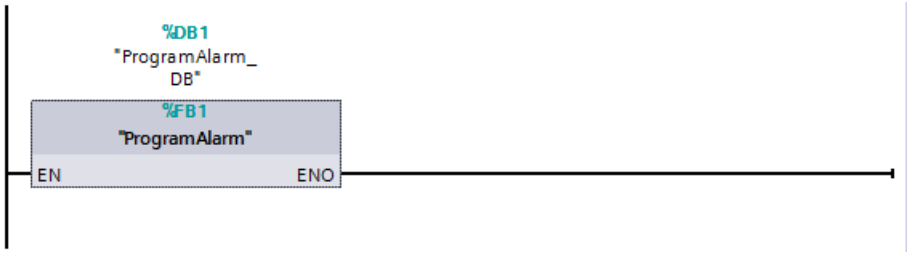

Mit dem Anlegen der Anweisung wird automatisch eine PLC-Meldung erstellt. Sie öffnen den Dialog PLC-Überwachungen & -Meldungen und wählen im Register Meldungen > Programmmeldungen die zu bearbeitende Meldung aus. In den Eigenschaften > Grundeinstellungen der Meldung legen Sie den Meldetext, einschließlich zwei Schlüsselworte, an.

Hinweis: Wenn Sie mit der rechten Maustaste in das Textfeld klicken, können Sie ein Schlüsselwort, eine Variable oder eine Textliste einfügen.

Durch die Zeichenfolge "@1%s@" wird der Wert des Parameters SD_1 ("#MyStaticText") ausgelesen und als Zeichenkette ausgegeben.

Siehe auch:

- Begleitwerte in Meldungen einfügen (Seite 7838)
- Beispiele für Begleitwerte (Seite 7842)
- Schlüsselwörter für Meldetexte (Seite 7844)

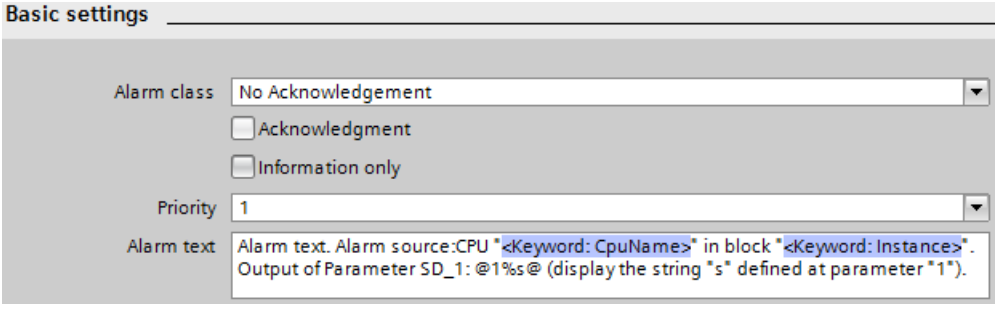

Wenn der Parameter SIG ("CreateAlarm") den Signalzustand "TRUE" liefert, wird die PLC-Meldung ausgegeben. Am Ausgangsparameter STATUS ("OUTPUT_Status") wird durch den Wert "0001" angezeigt , dass ein Signalwechsel erfolgt ist. Nachfolgend wird angezeigt, dass keine weitere Verarbeitung erfolgt (der Wert ist "0000"). Am Parameter SD_1 ("#MyStaticText") wird der Begleitwert der PLC-Meldung ausgegeben.

Dem Parameter TIMESTAMP kann ein Zeitstempel übergeben werden. Unverschaltet gibt der Parameter TIMESTAMP die Lokalzeit der CPU-Uhr aus. Am Ausgangsparameter ERROR ("OUTPUT_Error") wird angezeigt, dass die Verarbeitung der Anweisung ohne Fehler verlief.

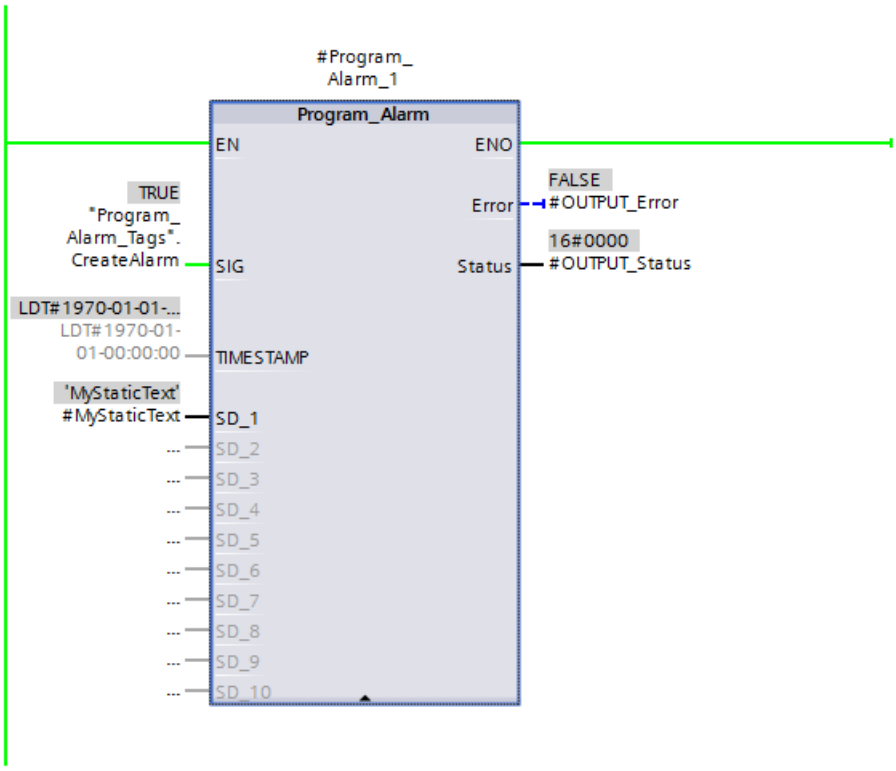

Zur Ausgabe der PLC-Meldung verwenden Sie z. B. den Webserver der CPU. Um den Webserver zu nutzen, ist Folgendes notwendig:

- Der Webserver muss in den CPU-Einstellungen aktviert sein.

Über den Internet-Browser öffnen Sie den Webserver (per IP-Adresse der CPU) und melden sich im Menü des Webservers an. Die CPU gibt den Meldetext solange aus, wie das zu überwachende Signal ("CreateAlarm") den Wert "TRUE" liefert.

Time	Message text
02:32:09.226 pm	Alarm text. Alarm source:CPU "PLC_2" in block "ProgramAlarm_DB". Output of Par

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Beispiel

Ein ausführliches Applikationsbeispiel finden Sie im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/98210758).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Get_AlarmState: Meldungszustand ausgeben

Beschreibung

Mit der Anweisung "Meldungszustand ausgeben" geben Sie den Meldungszustand einer Programmmeldung aus.

Es gibt drei mögliche Meldungszustände:

- Kommend
- Gehend
- Quittiert

Die Ausgabe des Meldungszustands bezieht sich jeweils auf eine Programmmeldung, die über die Anweisung "Programmmeldung mit Begleitwerten erzeugen" erstellt wurde.

Die Auswahl der Programmmeldung erfolgt über den Eingangsparameter Alarm. Am Parameter Alarm geben Sie den Instanz-DB der Anweisung "Programmmeldung mit Begleitwerten erzeugen" an.

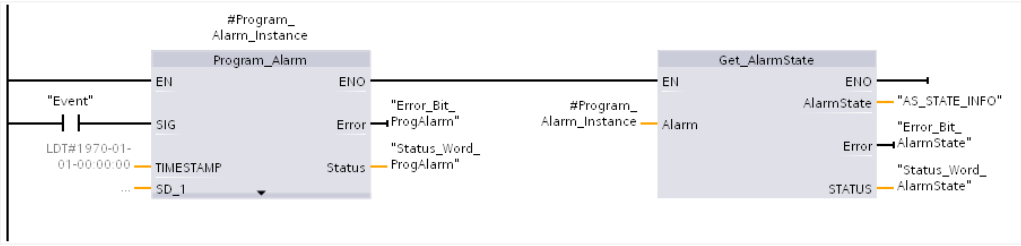

Über den Ausgangsparameter AlarmState wird der Meldungszustand in einem Byte ausgegeben. Die Bedeutung der einzelnen Bits ist in der folgenden Abbildung dargestellt:

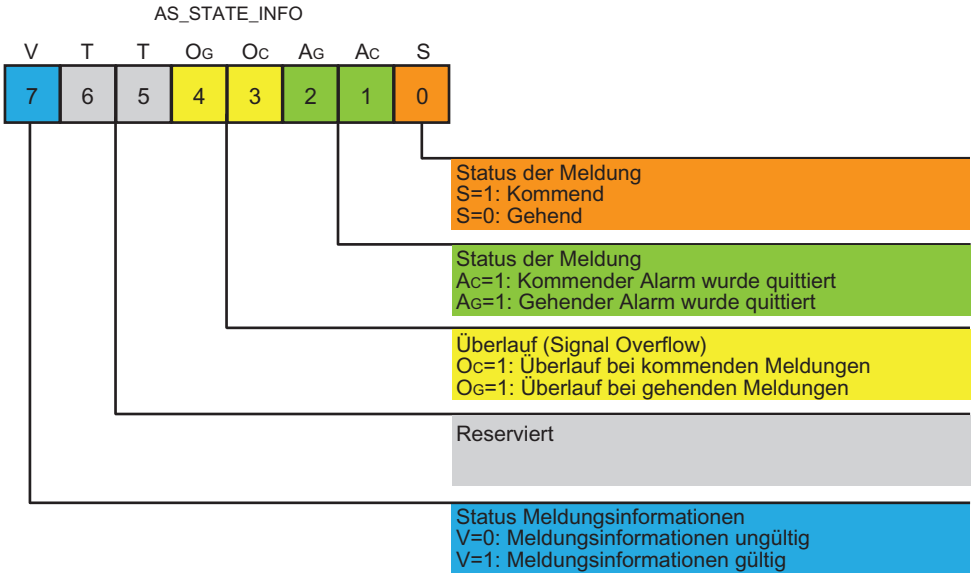

Über die Ausgangsparameter Error und STATUS wird der Bearbeitungszustand der Anweisung angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Meldungszustand ausgeben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Alarm	Input	ALARM_BASE	D	Instanz der Anweisung "Programmmeldung mit Begleitwerten erzeugen". • Alarm.Messagetype = Alarm_AP, dann hat das Bit Ac entweder den Signalzustand 0 oder 1 und das Bit Ag hat den Signalzustand 1 – nicht aktiv: 0x86 (1000 0110) – aktiv/nicht quittiert: 0x85 (1000 0101) – aktiv/quittiert: 0x87 (1000 0111) – gegangen/nicht quittiert: 0x84 (1000 0100) • Alarm.Messagetype = Notify_AP, dann haben die Bits Ac und Ag beide den Signalzustand 1. – nicht aktiv: 0x86 (1000 0110) – aktiv: 0x87 (1000 0111) • Alarm.Messagetype = Inforeport_AP, dann haben die Bits Ac und Ag beide den Signalzustand 1 – nicht aktiv: 0x86 (1000 0110) – (temporär) aktiv:0x87 (1000 0111) Wenn ein Alarm nicht aktiv ist, hat das Bit S grundsätzlich den Signalzustand 0.
AlarmState	Output	BYTE	E, A, M, D, L	Status der Meldung als Bitfeld

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Error	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Kein Fehler • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
8001	Ungültige statische Meldungsinstanz
8002	ID der Meldung ungültig
8003	Keine aktiven Meldungen innerhalb der Meldeklasse • Alarm_AP: Der Alarm ist gegangen und quittiert. • Notify_AP: Der Alarm ist gegangen. • Inforeport Das Bit V wird auf den Signalzustand 0 gesetzt.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel geben Sie den Meldungszustand einer Programmmeldung aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

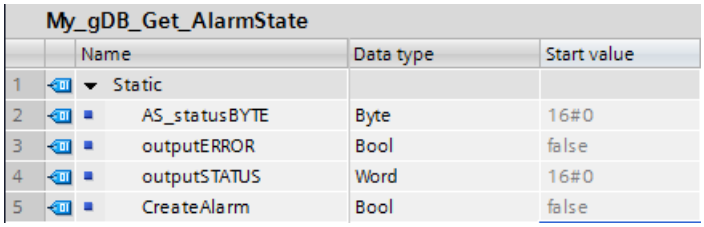

Die Anweisung "Get_AlarmState" wird zusammen mit der Anweisung "Program_Alarm" in einem Funktionsbaustein aufgerufen. Zur Verschaltung der Anweisungen legen Sie für den Funktionsbaustein vier Parameter an.

7	Static		
8	MyStaticText	String	'MyStaticText'
9	Program_Alarm_2	Program_Alarm	
10	Temp		
11	OUTPUT_Error	Bool	
12	OUTPUT_Status	Word	

Die Parameter der Anweisungen verschalten Sie wie folgt.

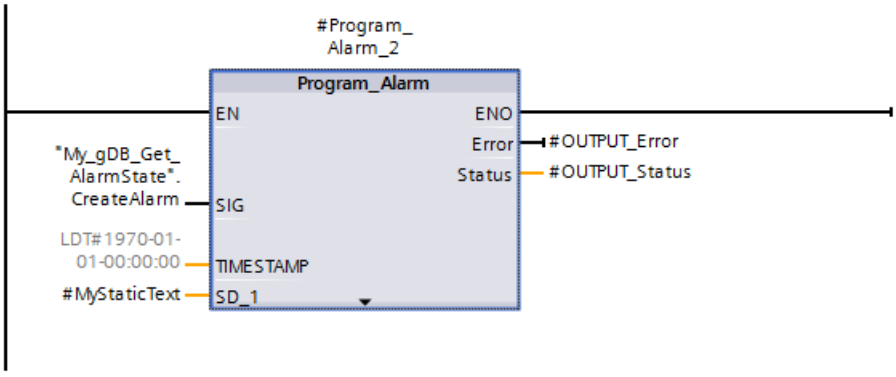

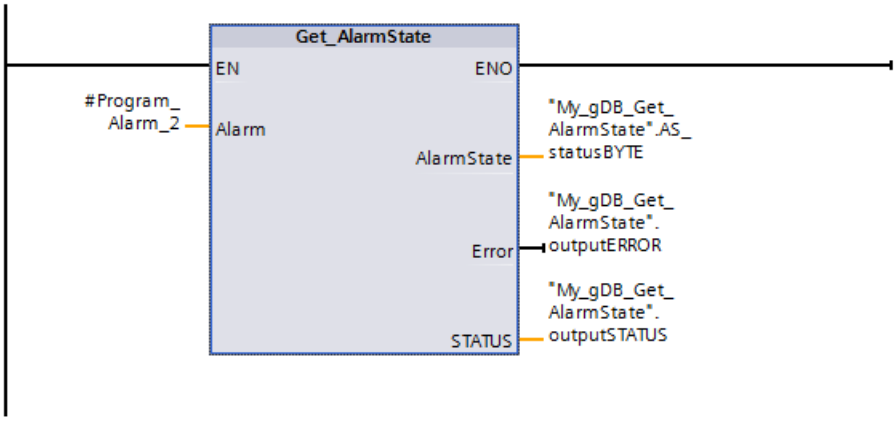

Den Funktionsbaustein rufen Sie in einem OB auf.

Mit dem Anlegen der Anweisung "Program_Alarm" wird automatisch eine PLC-Meldung erstellt. Sie öffnen den Dialog PLC-Überwachungen & -Meldungen und wählen im Register Meldungen > Programmmeldungen die zu bearbeitende Meldung aus. Sie legen den Meldetext, einschließlich zwei Schlüsselworte, an.

Für die Meldung wählen Sie folgende Einstellungen.

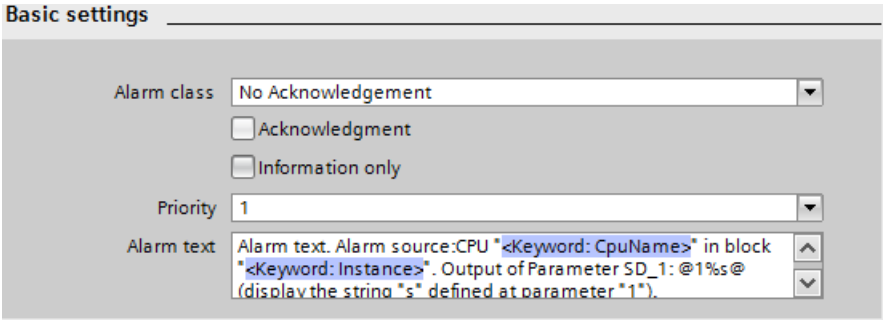

Wenn der Eingangsparameter SIG ("CreateAlarm") der Anweisung "Program_Alarm" den Signalzustand "TRUE" liefert, wird die PLC-Meldung ausgegeben.

Bei der Anweisung "Get_AlarmState" geschieht Folgendes: Über den Eingangsparameter ALARM wird der Anweisung "Get_AlarmState" die PLC-Meldung übermittelt. Am Ausgangsparameter AlarmState ("AS_statusBYTE") wird angezeigt, dass die Meldung entsprechend der Meldeklasse "No Acknowledgement" aktiv ist.

An den Ausgangsparametern ERROR ("outputERROR") und STATUS ("outputSTATUS") wird angezeigt, dass die Verarbeitung der Anweisung ohne Fehler verlief.

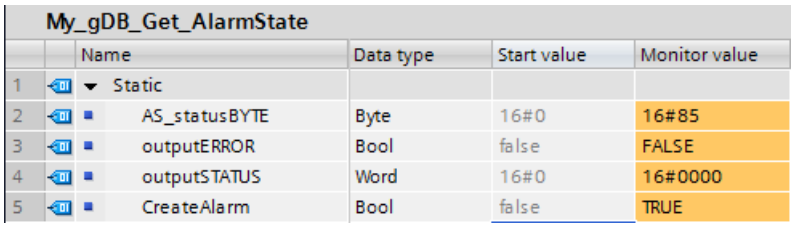

My_gDB_Get_AlarmState

	Name	Data type	Start value	Monitor value
1	Static			
2	AS_statusBYTE	Byte	16#0	16#85
3	outputERROR	Bool	false	FALSE
4	outputSTATUS	Word	16#0	16#0000
5	CreateAlarm	Bool	false	TRUE

Zur Ausgabe der PLC-Meldung verwenden Sie den Webserver der CPU.

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615)

Beispiel

Ein ausführliches Applikationsbeispiel finden Sie im Siemens Industry Online Support. (https://support.industry.siemens.com/cs/ww/de/view/98210758)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Program_Alarm: Programmmeldung mit Begleitwerten erzeugen (Seite 2624)

Meldungen anlegen und bearbeiten (Seite 7835)

Gen_UsrMsg: Anwenderdiagnosemeldungen erzeugen

Beschreibung

Mit der Anweisung "Gen_UsrMsg" erzeugen Sie eine Meldung, die in den Diagnosepuffer eingetragen wird.

Über den Parameter Mode wählen Sie aus, ob eine kommende oder eine gehende Meldung erzeugt werden soll:

- Bei Mode = 1: Erstellung einer kommenden Meldung.
- Bei Mode = 2: Erstellung einer gehenden Meldung.
- Unabhängig davon, ob eine kommende oder gehende Meldung erzeugt wird, hat die Meldung immer das Attribut "Nur Information".

Die Erstellung des Eintrags in den Diagnosepuffer erfolgt synchron. Das Übermitteln der Meldung erfolgt asynchron.

Wenn bei der Ausführung der Anweisung ein Fehler auftritt, wird dieser über den Parameter RET_VAL ausgegeben.

Inhalt der Meldung

Den Inhalt der Meldung definieren Sie über eine Textliste:

- Über den Parameter TextListID geben Sie die Textliste an, die Sie verwenden wollen. Öffnen Sie hierzu den Dialog "PLC-Meldetextlisten" in der Projektnavigation. Blenden Sie die Spalte "ID" im Dialog "Textlisten" ein. Übernehmen Sie die ID am Parameter TextListID.
- Über den Parameter TextID wählen Sie den Textlisten-Eintrag aus, den Sie in den Diagnosepuffer schreiben möchten. Wählen Sie hierzu aus dem Dialog "Textlisten-Einträge" einen Eintrag aus, indem Sie die Zahl aus den Spalten "Bereich von / Bereich bis" am Parameter TextID übernehmen. Hierzu muss für den Textlisten-Eintrag für "Bereich von" und "Bereich bis" jeweils die gleiche Zahl verwendet werden.
- Weitere Informationen zu Textlisten finden Sie hier: AUTOHOTSPOT

Begleitwerte definieren

In dem Textlisten-Eintrag definieren Sie zusätzlich Begleitwerte, um welche die Meldung ergänzt wird:

- Die Begleitwerte definieren Sie, indem Sie den Textlisten-Eintrag um die folgende Angabe ergänzen:
  @<Nummer des Begleitwertes><Elementtyp><Formatangabe>@
- Über den Systemdatentyp AssocValues geben Sie an, welcher Begleitwert beim Erzeugen der Meldung ergänzt wird.
- Weitere Informationen zum Aufbau von Begleitwerten finden Sie hier: Aufbau von Begleitwerten (Seite 7831)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gen_UsrMsg":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Mode	Input	UInt	E, A, M, D, L oder Konstante	Parameter zur Auswahl des Status der Meldung: • 1: kommende Meldung • 2: gehende Meldung
TextID	Input	UInt	E, A, M, D, L oder Konstante	ID des Textlisten-Eintrags, welcher für den Meldetext verwendet werden soll.
TextListID	Input	UInt	E, A, M, D, L oder Konstante	ID der Textliste, die Textlisten-Eintrag enthält.
Ret_Val	Return	Int	E, A, M, D, L	Fehlercode der Anweisung.
AssocValues	InOut	VARIANT	D, L	Zeiger auf den Systemdatentyp AssocValues, über den Sie die Begleitwerte definieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter AssocValues

Über den Systemdatentyp AssocValues definieren Sie, welche Begleitwerte mitgesendet werden. Maximal sind 8 Begleitwerte möglich. Die Struktur des Systemdatentyps erstellen Sie, indem Sie in einem Datenbaustein als Datentyp "AssocValues" eingeben.

Die Begleitwerte wählen Sie aus, indem Sie für die Parameter Value[x] die Nummern der Begleitwerte eintragen. Beachten Sie hierbei Folgendes:

- Die Werte für TextID und TextListID werden von der Anweisung wie die zu sendenden Begleitwerte behandelt. Dadurch sind "1" und "2" als Nummern für die Adressierung von Begleitwerten bereits vergeben. Verwenden Sie nicht die Nummern "1" oder "2", um Begleitwerte zu adressieren.
- Adressieren Sie den Begleitwert am Parameter Value [1] als Nummer "3", am Parameter Value [2] als Nummer "4" usw.

Byte	Parameter	Datentyp	Startwert	Beschreibung	Nummer des Begleitwerts
0..1	Value[1]	UINT	0	Erster Begleitwert der Meldung.	3
2..3	Value[2]	UINT	0	Zweiter Begleitwert der Meldung.	4
4..5	Value[3]	UINT	0	...	5
6..7	Value[4]	UINT	0	...	6
8..9	Value[5]	UINT	0	...	7

Byte	Parameter	Datentyp	Startwert	Beschreibung	Nummer des Begleitwerts
10..11	Value[6]	UINT	0	...	8
12..13	Value[7]	UINT	0	...	9
14..15	Value[8]	UINT	0	Achter Begleitwert der Meldung.	10

Parameter RET_VAL

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über den Parameter RET_VAL ausgegeben werden.

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Wert am Parameter Mode nicht unterstützt.
80C1	Ressourcenengpass aufgrund zu vieler paralleler Aufrufe.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel erzeugen Sie eine Meldung, die in den Diagnosepuffer eingetragen wird.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein fünf Variablen und eine Struktur "AssocValues" (mit dem Datentyp AssocValues) an.

My_gDB_Gen_UsrMsg

	Name	Data type	Start value
1	Static		
2	myMODE	UInt	1
3	myTEXT	UInt	3
4	myLIST	UInt	512
5	returnValue	Int	0
6	execute	Bool	false
7	AssocValues	AssocValues	
8	Value	Array[1..8] of UInt	
9	Value[1]	UInt	33
10	Value[2]	UInt	411
11	Value[3]	UInt	578
12	Value[4]	UInt	6122
13	Value[5]	UInt	722
14	Value[6]	UInt	8111
15	Value[7]	UInt	9829
16	Value[8]	UInt	10457

Die Parameter der Anweisung verschalten Sie wie folgt.

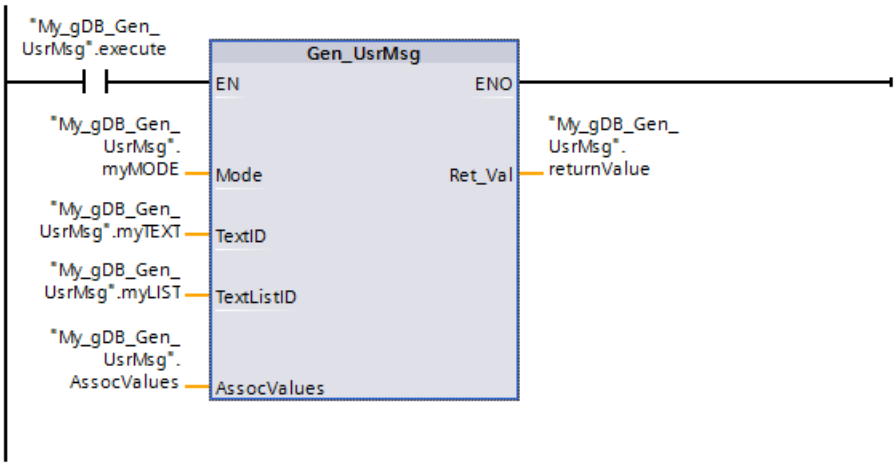

Für die Meldung erstellen Sie über den Eintrag "Textlisten" eine Textliste sowie einen Textlisten-Eintrag. Die ID der Textliste übernehmen Sie am Parameter TextListID ("myLIST"). Die ID (Bereich ...) des Textlisten-Eintrags übernehmen Sie am Parameter TextID ("myTEXT"). Die Meldung parametrieren Sie wie folgt.

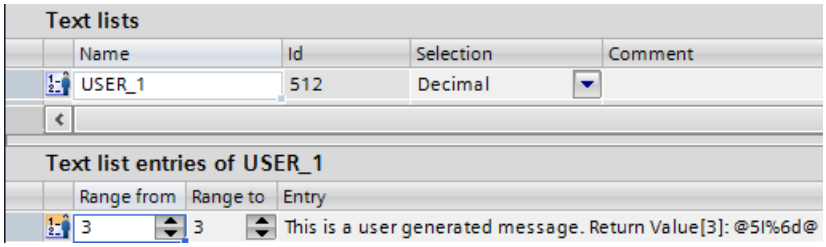

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "Gen_UsrMsg" ausgeführt. Entsprechend dem Wert des Parameters Mode ("myMODE") wird eine kommende Meldung erzeugt. Über die Parameter TextListID ("myLIST") und TextID ("myTEXT") wird die auszugebende Meldung der Anweisung bekannt gegeben. Über den Parameter AssocValues ("AssocValues") werden die Begleitwerte der Meldung übergeben.

Beim Erzeugen der Meldung wird die im Meldetext enthaltene Zeichenfolge "@5I%6d@" wie folgt interpretiert:

- Der Begleitwert mit der Nummer "5" wird im Datentyp INT ausgelesen. Die Nummer entspricht dem Parameter Value[3] der Struktur "AssocValues".
- Der Begleitwert wird als Dezimalzahl ausgegeben. Die Dezimalzahl ist auf sechs Stellen begrenzt.

Am Ausgangsparameter Ret_Val ("returnValue") wird angezeigt, dass die Verarbeitung der Anweisung ohne Fehler verlief.

My_gDB_Gen_UsrMsg

	Name	Data type	Start value	Monitor value
1	Static			
2	myMODE	UInt	1	1
3	myTEXT	UInt	3	3
4	myLIST	UInt	512	512
5	returnValue	Int	0	0
6	execute	Bool	false	FALSE
7	AssocValues	AssocValues		
8	Value	Array[1..8] of UInt		
9	Value[1]	UInt	33	33
10	Value[2]	UInt	411	411
11	Value[3]	UInt	578	578
12	Value[4]	UInt	6122	6122
13	Value[5]	UInt	722	722
14	Value[6]	UInt	8111	8111
15	Value[7]	UInt	9829	9829
16	Value[8]	UInt	10457	10457

Zur Ausgabe der Meldung öffnen Sie bei einer CPU der Baureihe S7-1500 den Eintrag "Online & Diagnose > Diagnosepuffer".

Details on event: 1 of 1000 Event ID: 16# 2D2:0003

Description: User message: This is a user generated message. Return Value[3]: 578

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel

Ein ausführliches Applikationsbeispiel finden Sie im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/view/98210758).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Meldungen anlegen und bearbeiten (Seite 7835)

Get_Alarm: Anstehende Meldung lesen

Beschreibung

Mithilfe der Anweisung "Anstehende Meldung lesen" können Sie sich im Anwenderprogramm für erzeugte Meldungen anmelden, um jeweils eine kommende oder gehende Meldung in einen referenzierten Datenbaustein zu kopieren. Dadurch können Sie z. B. Meldungen als STRING an einen Leitrechner weitergeben.

Wenn Sie die Anweisung mit dem "OperateMode = 1" aufrufen und sich damit anmelden, dann müssen Sie mindestens eine oder mehrere Anzeigeklassen am Eingang "DispClassNr" angeben. Nach dem Anmelden gehen keine Meldungen mehr verloren, weder kommende noch gehende.

Die Meldungen werden sequenziell, beginnend mit der ältesten Meldung, abgearbeitet. Sie werden ausgelesen und in der CPU in einem Datenbaustein gespeichert. Anschließend können Sie die Daten, z. B. mithilfe der Anweisung "TSEND_C", an einen Leitrechner weiterleiten.

Die Anweisung arbeitet asynchron und vermeidet dadurch lange Laufzeiten des Funktionsbausteins "Get_Alarm [FB702]", die zu einem Zyklusüberlauf führen könnten. Die Anweisung muss wiederkehrend in einem Zyklus-Organisationsbaustein aufgerufen werden.

Da der Funktionsbaustein "Get_Alarm [FB702]" pro abgeholter Meldung bis zu 11 Aufrufe pro abgefragter Sprache benötigt, kann am Parameter Busy bis zu 33 Aufrufe lang der Signalzustand "1" bestehen bleiben. Erst nachdem der Parameter Busy wieder den Signalzustand "0" hat, ist das Abholen einer Meldung abgeschlossen.

Daher ergibt sich die empfohlene Aufrufzeit des "Get_Alarm [FB702]" zwischen 20ms und 200ms. Sollte die Aufrufzeit > 1,8 Sekunden betragen, kann es zu einer automatischen Abmeldung des "Get_Alarm [FB702]" kommen. Die Abmeldung führt dazu, dass keine Meldungen mehr ausgelesen werden.

In den folgenden Fällen ist es erforderlich, dass Sie die Anweisung mithilfe des Eingangs "OperateMode = 1" erneut anmelden:

- Nach automatischem Abmelden aufgrund eines zu seltenen Aufrufs des Funktionsbausteins "Get_Alarm [FB702]".
- Bei einem Wechsel des Betriebszustands von "STOP" > "RUN".
- Sobald Sie bereits bestehende Meldungsprojektierungen ändern, z. B. den Program_Alarm oder ProDiag-Überwachungen, meldet sich die Anweisung automatisch ab.

Die Anweisung können Sie für Meldungen aus den folgenden Bereichen verwenden:

- ProDiag
- Systemdiagnose
- GRAPH
- Program_Alarm
- Motion Control

Hinweis

CPU-Eigenschaft "Zentrale Meldungsverwaltung in der PLC"

Wenn Sie die CPU-Eigenschaft "Zentrale Meldungsverwaltung in der PLC" in der Gerätekonfiguration nicht aktivieren, dann werden statt der Meldetexte (Meldetext, Informationstext und die zusätzlichen Meldetexte 1 - 9) nur die IDs der Meldetextfelder ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anstehende Meldung lesen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN *)	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO *)	Output	BOOL	E, A, M, D, L	Freigabeausgang
Operate-Mode	Input	BYTE	E, A, M, D, L	Es gibt drei verschiedene mögliche Werte: • 1: Anmelden • 2: Daten auslesen • 3: Abmelden
DataMode	Input	BYTE	E, A, M, D, L	Es gibt vier verschiedene mögliche Werte: • 0: Es werden nur die Spaltenüberschriften ausgelesen (Producer-ID, ID_1, ID_2, Prio, Timestamp, State) • 1: Es werden die Spaltenüberschriften und der Meldetext inkl. Begleitwerte ausgelesen. • 2: Es werden die Spaltenüberschriften, der Meldetext inkl. Begleitwerte und der Informationstext ausgelesen. • 3: Es werden die Spaltenüberschriften, der Meldetext inkl. Begleitwerte, der Informationstext und die zusätzlichen Meldetexte ausgelesen.
DispC-lassNr (Display class number)	Input	AR-RAY[0..31] of UINT	E, A, M, D, L	Sie können bis zu 32 verschiedene Anzeigeklassen definieren: • 0 ... 16: Anwenderdefinierte Anzeigeklassen, die im HMI-Gerät eingestellt werden können • 257: Systemdiagnose • 259: Security Wenn Sie den Parameter nicht verwenden, dann bekommen Sie alle Anzeigeklassen gemeldet. Eine Änderung der Anzeigeklasse wird nur berücksichtigt, wenn am Parameter Mode der Modus 1 eingestellt ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Lcid (Locale ID)	Input	UDINT	E, A, M, D, L	Es gibt verschiedene mögliche Werte: • 0: Alle Sprachen • 1: Erste Sprache mit der kleinsten LC Nummer • 2: Erste Sprache plus zweite Sprache mit der zweit kleinsten LC Nummer • 3: Alle Sprachen • oder: Geben Sie die genaue LC Nummer um eine spezifische Sprache auszuwählen.
Data	InOut	"AlarmData"	E, A, M, D, L	Zielbereich für die ausgelesenen Meldungsdaten. Meldetextstruktur vom PLC-Datentyp "AlarmData" entsprechend des Werts am Parameter "DataMode".
DataReady	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Es stehen keine Daten zur Verfügung. • 1: Die Daten stehen zur Verfügung. Die Daten sind nur bis zum nächsten Aufruf der Anweisung gültig. Deshalb müssen sie im gleichen Programmzyklus z. B. mithilfe einer Move-Anweisung verarbeitet werden.
Busy	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Die Ausführung der Anweisung wurde noch nicht gestartet oder die asynchrone Bearbeitung wurde abgeschlossen. Weitere Informationen zum Bearbeitungszustand können Sie dem Parameter STATUS entnehmen. • 1: Ausführung der Anweisung noch nicht abgeschlossen Weitere Einträge sind vorhanden
Error	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Kein Fehler • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Init	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Keine Handlung notwendig. • 1: Eine erneute Anmeldung ist erforderlich. Die bisher erhaltenen Meldedaten sind nicht mehr gültig. Das kann in folgenden Fällen notwendig sein: • Sie haben sich angemeldet. • Die CPU wechselt von "STOP" in den Betriebszustand "RUN". • Meldungen wurden umprojektiert und die Änderungen wurden mittels "Laden in Gerät" im Betriebszustand "RUN" verändert, z. B. eine Änderung an der Meldetextstruktur.
Status	Output	WORD	E, A, M, D, L	Fehler- und Statusinformationen
*): Der Freigabeeingang EN und der Freigabeausgang ENO sind nur in den Programmiersprachen KOP und FUP sichtbar.				

Parameter Status

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters Status:

Statuscode* (W#16#...)	Erläuterung
0000	Kein Fehler
0001	Es sind neue Daten verfügbar.
8001	Der Wert am Parameter OperateMode ist ungültig.
8002	Der Parameter Data hat den falschen Datentyp.
8003	Der Parameter Lcid ist ungültig oder eine am Parameter Lcid gewählte Sprache ist nicht geladen.
8004	Sie hatten sich bereits über den Parameter OperateMode = 1 im Anwenderprogramm angemeldet, um anstehende Meldungen zu lesen. Ein erneutes Anmelden ist erst dann wieder zulässig und notwendig, wenn Sie sich mithilfe des OperateMode = 3 abmelden, eine Umprojektierung von Meldungen vorgenommen oder einen STOP > RUN Übergang ausgeführt haben.
8005	Der Wert am Parameter DataMode ist ungültig.
8085	• Sie haben· bei der Anmeldung keine Anzeigeklasse angegeben. • Die Anweisung wurde bei der Umprojektierung von Überwachungen (ProDiag) abgemeldet. • Automatisches Abmelden aufgrund eines zu seltenen Aufrufs des Funktionsbausteins "Get_Alarm [FB702]".
80C5	Die Bearbeitung der Anweisung ist noch nicht abgeschlossen. Versuchen Sie es zu einem späteren Zeitpunkt noch einmal.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

PLC-Datentyp "AlarmData"

Der PLC-Datentyp "AlarmData" gibt die Struktur einer Meldung wieder.

Name	Datentyp	Defaultwert	Beschreibung
ProducerID	UINT	0	Es gibt verschiedene mögliche Werte: • 0: Program_Alarm • 4: Systemdiagnose • 5: Standard Motion Control (T-CPU) • 6: Security (z. B. Falsches Login) • 7: SINUMERIK • 8: GRAPH • 9: ProDiag
ID_1	UINT	0	Meldungs-ID Diese ID wird auch im Meldungseditor angezeigt wird.
ID_2	UINT	0	ID der Meldungs-Runtime-Instanz Diese ID ist eine interne Range-ID, die bei der Pro-Diag und bei Schrittketten eine Bedeutung hat: • Bei ProDiag ist es die Überwachungs-ID. • Bei einer Schrittkette ist es die Schrittnummer. Bei der Anweisung "Program_Alarm" ist sie 0 und bei der Systemdiagnose ist sie ohne Bedeutung.
PRIO	UINT	0	Prioritäten: 0 ... 16
TimeStamp	LDT	DT#1970-01-01-0:0:0.000000000	Zeitstempel Der Zeitstempel basiert auf der UTC-Zeit (Universal Time Coordinated) bzw. der Systemzeit.
State	BYTE	16#0	Zustandsparameter: • 0: Gehend • 1: Kommend
LC_ID	ARRAY[0..2] of UINT	-	Liefert den Wert, der am Eingang "Lcid" angegeben ist: • 0: Alle Sprachen • 1: Erste Sprache mit der kleinsten LC Nummer • 2: Erste Sprache plus zweite Sprache mit der zweit kleinsten LC Nummer • 3: Alle Sprachen oder: Geben Sie die genaue LC Nummer um eine spezifische Sprache auszuwählen.
AlarmText	ARRAY[0..2] of WSTRING	-	Meldetext mit bis zu 254 Zeichen (508 Byte). Ist der Text länger, wird er bei 251 Zeichen abgeschnitten, z. B. WSTRING#'Meld... .
InfoText	ARRAY[0..2] of WSTRING	-	Informationstext
AddText_1 - 9	ARRAY[0..2] of WSTRING	-	Zusätzlicher Text 1 - 9

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 1 (Seite 2649).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Begleitwerte in Meldungen einfügen (Seite 7838)

Beispiele für Begleitwerte (Seite 7842)

Einführung in die Meldungsprojektierung (Seite 7815)

Ack_Alarms: Meldungen quittieren

Beschreibung

Mithilfe der Anweisung "Meldungen quittieren" können Sie alle kommenden und gehenden Meldungen innerhalb Ihres Programmcodes mit einem einzigen Befehl quittieren.

Sobald eine positive Signalflanke vorliegt, entweder am Freigabeeingang EN (KOP / FUP) oder als Vorverknüpfung (AWL / SCL), und am Parameter MODE der Wert = 1 eingestellt ist, wird die Anweisung aufgerufen. Die Anweisung arbeitet asynchron und vermeidet dadurch lange Laufzeiten des Funktionsbausteins "Ack_Alarms [FB701]", die zu einem Zyklusüberlauf führen könnten. Die CPU bestätigt alle kommenden und gehenden Meldungen und informiert alle HMI-Geräte über diese Bestätigung. Eventuell anstehende Meldungen auf den Displays verschwinden, sobald sie "gegangen" sind.

Es kann jeweils nur eine Bestätigung (max. 100 quittierpflichtige Meldungen) zur gleichen Zeit durchgeführt werden. Wenn weitere Meldungen anstehen, dann werden automatisch die nächsten 100 quittierpflichtigen Meldungen bestätigt.

Die Anweisung können Sie auch für ProDiag-Überwachungsmeldungen verwenden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Meldungen quittieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN *)	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO *)	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L	Es gibt zwei verschiedene mögliche Werte: • 0: Die Anweisung wird nicht ausgeführt. • 1: Alle Meldungen bestätigen. Alle aktiven Meldungen, die bestätigt werden können, werden bestätigt.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Kein Fehler • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Fehler- und Statusinformationen
*): Der Freigabeeingang EN und der Freigabeausgang ENO sind nur in den Programmiersprachen KOP und FUP sichtbar.				

Parameter STATUS

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters STATUS:

Statuscode* (W#16#...)	Erläuterung
0000	Kein Fehler
7001	Es ist kein Fehler aufgetreten und die nächsten 100 quittierpflichtigen Meldungen werden bestätigt.
8001	Der Wert am Parameter MODE ist ungültig.
80C5	Aktuell läuft bereits ein Bestätigungsvorgang. Bitte versuchen Sie es später nochmal.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 1 (Seite 2649).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Begleitwerte in Meldungen einfügen (Seite 7838)

Beispiele für Begleitwerte (Seite 7842)

Einführung in die Meldungsprojektierung (Seite 7815)

Programmbeispiel für Get_Alarm & Ack_Alarms

Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 1

Einleitung

In dem folgenden Beispiel erfassen Sie alle Meldungen der CPU 1 (S7-1500 ab Firmware v2.0) und speichern die Meldungen ab. Anschließend nutzen Sie eine programmierte Verbindung zur CPU 2 (S7-1500), um die Meldungen (in gekürzter Form; im PLC-Datentyp "rcvGetAlarm") von der CPU 1 zur CPU 2 zu senden. Auf der CPU 2 speichern Sie die Meldungen ebenso ab.

Im ersten Teil des Programmbeispiels zu Get_Alarm und Ack_Alarms beschäftigen Sie sich mit den Voraussetzungen und dem Verschalten der Parameter.

Die Ergebnisse und das Vorgehen für die Verbindung finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 2 (Seite 2667).

Hinweis

Größe der Speicher-Arrays für die Meldungen

Die Speicher-Arrays für die Meldungen ("alarmStorage" und "rcvdStorage") können nach beschriebener Projektierung maximal 16 Meldungen aufnehmen.

Wenn Sie die Speichergröße ändern wollen, müssen Sie jeweils die Array-Größe und den Maximalwert "maxPosition" ändern.

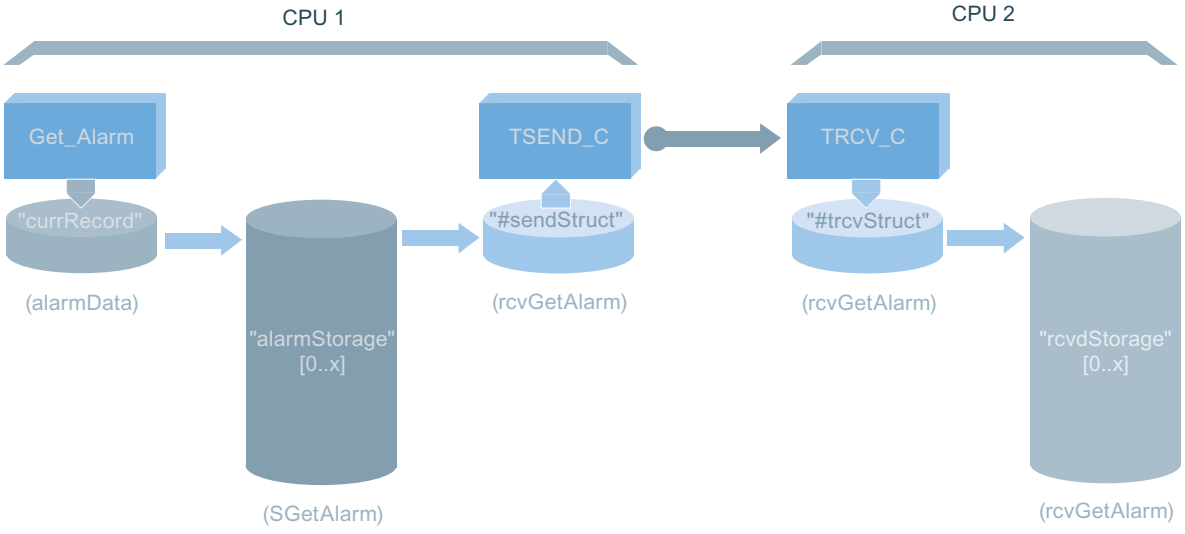

Voraussetzung für eine programmierte Verbindung

Sie legen zwei CPUs (S7-1500) an und verbinden die CPUs per PROFINET miteinander. Die Verbindung lassen Sie unparametriert.

Hinweis

Get_Alarm und Ack_Alarms nutzen

Die CPU 1 (für den Aufruf von Get_Alarm und Ack_Alarms) muss mindestens die Firmware v2.0 besitzen. Zudem muss für die CPU 1 die CPU-Eigenschaft "Zentrale Meldungsverwaltung in der PLC (Seite 2642)" aktiviert sein.

Hinweis

Lese- und Schreibzugriffe erlauben

Bei den Eigenschaften > Schutz der CPU stellen Sie mit einer niedrigen Schutzstufe sicher, dass Lese- und Schreibzugriffe erlaubt sind.

CPU 1: Programm-Voraussetzungen

PLC-Datentypen erstellen

Sie benötigen die folgenden PLC-Datentypen für die Weitergabe und Ablage der Meldungen:

- "SGetAlarm" benötigen Sie für die Ablage der Meldungen, inklusive Information über den aktuellen Leseverlust

SGetAlarm

	Name	Data type	Default value
1	countReadLoss	UDInt	0
2	alarm	AlarmData	

- "rcvGetAlarm" benötigen Sie für die Übermittlung der Meldungen über TSEND_C.

rcvGetAlarm

	Name	Data type	Default value
1	countReadLoss	UDInt	0
2	ProducerID	UInt	0
3	ID_1	UInt	0
4	ID_2	UInt	0
5	PRIO	UInt	0
6	TimeStamp	LDT	LDT#1970-01-01-
7	State	Byte	16#0
8	LCID	Array[0..2] of UInt	
9	AlarmText	WString[254]	WSTRING#''

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein ("SLI_gDB_GAalarm") folgende Strukturen und Variablen an.

- Für die Parametrierung von TSEND_C:

SLI_gDB_GAalarm			
	Name	Data type	Start value
1	Static		
2	TSEND_C	Struct	
3	start	Bool	false
4	stop	Bool	false
5	comControl	Bool	false
6	done	Bool	false
7	busy	Bool	false
8	error	Bool	false
9	status	Word	16#0
10	memErrStatus	Word	16#0

- Für die Parametrierung von Get_Alarm:

SLI_gDB_GAalarm			
	Name	Data type	Start value
11	GetAlarm	Struct	
12	start	Bool	false
13	stop	Bool	false
14	check	Struct	
15	anew	Bool	false
16	operateMode	Struct	
17	value	Byte	16#0
18	Logon_done	Bool	false
19	ReadData_ready	Bool	false
20	ReadData_active	Bool	false
21	Logoff_done	Bool	false
22	outputMode	Byte	16#3
23	languageCodeID	UDInt	0
24	currRecord	AlarmData	
25	currReadLoss	UDInt	0
26	dataReady	Bool	false
27	initiateLogin	Bool	false
28	busy	Bool	false
29	status	Word	16#0
30	memError	Bool	false
31	memErrStatus	Word	16#0

- Als Teil der Struktur "GetAlarm", zum Speichern der Meldungen in dem Speicher-Array:

32	maxPosition	UInt	15
33	alarmStorage	Array[0..15] of "SGetAlarm"	
34	limitReach	Bool	false
35	savingDone	Bool	false
36	initiateSending	Bool	false
37	countAlarms	UInt	0
38	posForAlarm	UInt	0

- Für die Parametrierung von Ack_Alarms:

40	AckAlarm	Struct	
41	executeAck	Bool	false
42	error	Bool	false
43	status	Word	16#0

ProDiag-Meldung

Für ein Test-Szenario legen Sie in einem globalen Datenbaustein ("SLI_gDB_testAlarm_GAA") drei Variablen an.

SLI_gDB_testAlarm_GAA

	Name	Data type	Start value
1	Static		
2	startPump	Bool	false
3	closeValve	Bool	false
4	valveOut	Bool	false

Zum Auslösen einer ProDiag-Meldung legen Sie in dem globalen Datenbaustein ("SLI_gDB_testAlarm_GAA") die folgende Regel an. Öffnen Sie dazu den globalen Datenbaustein > die Eigenschaften der Variable "startPump" > Überwachungen > Neue Überwachung hinzufügen.

General

Type of supervision:	Operand		
Supervised tag:	"SLI_gDB_testAlarm_GAA".startPump	Trigger:	True
Delay time:	T#0ms		
Condition 1:	"SLI_gDB_testAlarm_GAA".closeValve	C1 trigger:	True
Condition 2:		C2 trigger:	True
Condition 3:		C3 trigger:	True
Category:	2: Warning		
Subcategory 1:			
Subcategory 2:			

CPU 2: Programm-Voraussetzungen

PLC-Datentyp erstellen

Legen Sie den PLC-Datentyp "rcvGetAlarm" an wie für die CPU 1 beschrieben.

Ablage der Daten

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein ("SLI_gDB_trcv_GAA") folgende Strukturen und Variablen an.

SLI_gDB_trcv_GAA

	Name	Data type	Start value
1	Static		
2	TRCV_C	Struct	
3	start	Bool	false
4	comControl	Bool	false
5	done	Bool	false
6	busy	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStatus	Word	16#0
10	rcvLen	UDInt	0
11	dataReceived	Bool	false
12	GetAlarm	Struct	
13	maxPosition	UInt	15
14	rcvdStorage	Array[0..15] of "rcvGetAlarm"	
15	limitReach	Bool	false
16	savingDone	Bool	false
17	countAlarms	UInt	0
18	posForAlarm	UInt	0

CPU 1: FB "SLI_FB_send_GAA" - Parameter verschalten

Für den Verbindungsaufbau zwischen CPU 1 und CPU 2 und der Datenübermittlung erstellen Sie den FB "SLI_FB_send_GAA".

Im FB legen Sie die folgenden lokalen Variablen an.

SLI_FB_send_GAA

	Name	Data type	Default value	Retain
1	Input			
2	start	Bool	false	Non-ret...
3	connect	Bool	false	Non-retain
4	Output			
5	<Add new>			
6	InOut			
7	done	Bool	false	Non-retain
8	Static			
9	posForAlarm	UInt	0	Set in IDB
10	sendStruct	"rcvGetAlarm"		Non-retain
11	initiateSending	Bool	false	Set in IDB
12	do	Bool	false	Non-retain
13	do2	Bool	false	Non-retain
14	Temp			
15	tempDone	Bool		
16	Constant			
17	<Add new>			

Im FB verwenden Sie:

- Verschaltungen zum Abbrechen ("TSEND_C.stop") der Datenübermittlung.
- Verschaltungen zum Starten ("#do2") der Datenübermittlung.

- Verschaltungen, um eine neue Datenübermittlung zu initiieren ("#initiateSending", "#do").
- die Anweisung "TSEND_C" für den Verbindungsaufbau und die Datenübermittlung.

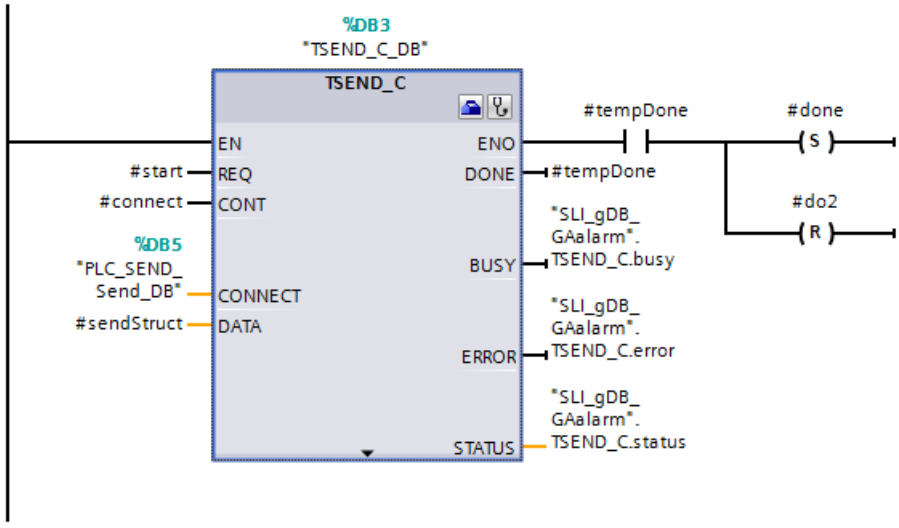

Hinweis

Verschaltung des Eingangsparameters CONNECT der Anweisung "TSEND_C"

Zur Verschaltung des Eingangsparameters CONNECT öffnen Sie den Wizard der Anweisung "TSEND_C" über deren "Eigenschaften > Konfiguration".

Nehmen Sie die folgenden Einstellungen für die Verbindung von TSEND_C vor:

- Endpunkt: Sie wählen jeweils über die Klappliste die CPUs aus. Schnittstelle, Subnetz und Adresse werden automatisch eingetragen.
- Verbindungsdaten: Über die Klappliste und die Auswahl "Neu" erstellen Sie je CPU einen Datenbaustein. Dieser Datenbaustein wird für die Ablage der Verbindungsdaten benötigt. Der Name des Datenbausteins ist je CPU (Kommunikationspartner) frei wählbar.
- Sie wählen den Verbindungstyp "TCP" und die Konfigurationsart "Programmbaustein verwenden". Diese Einstellungen sorgen für eine Ethernet-Verbindung mit dem Protokoll "TCP", und als programmierte Verbindung.
- Sie tragen eine selbst gewählte Verbindungs-ID ein. Verwenden Sie eine Verbindungs-ID, die nicht bereits vergeben ist.
- Sie tragen einen Wert für den Partner Port ein. Der Wert muss >=2000 sein.

Weitere Informationen zur Konfiguration von TSEND_C (Seite 3631) finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

Weitere Informationen zum Aufbau und den Verschaltungen des FBs "SLI_FB_send_GAA" finden Sie im Programmcode des Beispiels in der Sample Library for Instructions (Seite 615).

CPU 1: FC "SLI_FC_switchMode_GAA" - Parameter verschalten

Zum Steuern der Anweisung "Get_Alarm" erstellen Sie den FC "SLI_FC_switchMode_GAA".

Im FC legen Sie die folgenden lokalen Variablen an.

SLI_FC_switchMode_GAA		
	Name	Data type
1	Input	
2	start	Bool
3	stop	Bool
4	Output	
5	<Add new>	
6	InOut	
7	mode2Reinit	Bool
8	Temp	
9	tempStart	Bool
10	tempStop	Bool
11	Constant	
12	<Add new>	
13	Return	
14	SLI_FC_switchMode_GAA	Void

Im FC legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Zum Starten der Verarbeitung von Get_Alarm legen Sie die folgenden Verschaltungen an.

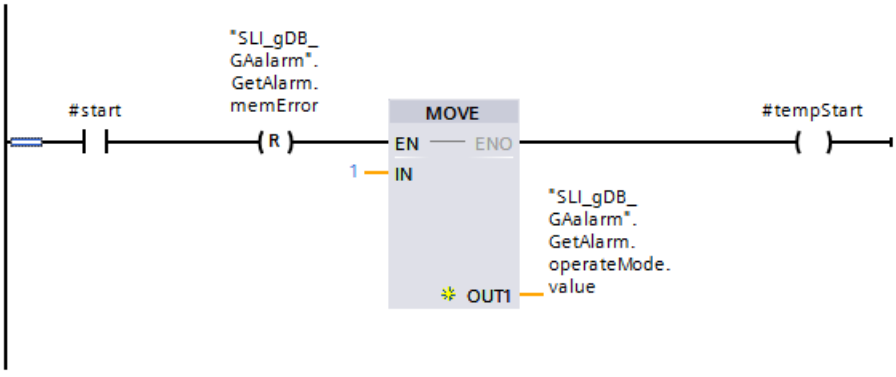

Netzwerk 2: Zum Starten der Verarbeitung von Get_Alarm legen Sie die folgenden Verschaltungen an (Teil 2).

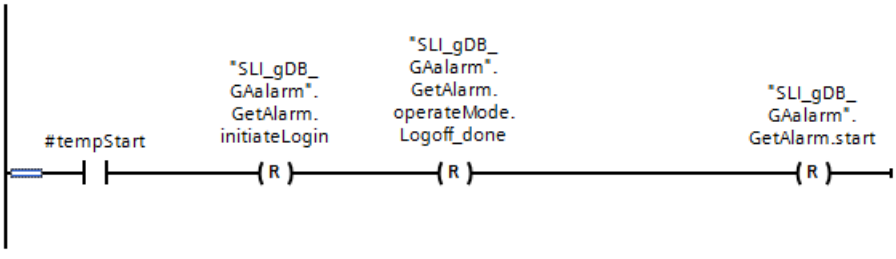

Netzwerk 3: Für den Login von Get_Alarm legen Sie die folgenden Verschaltungen an.

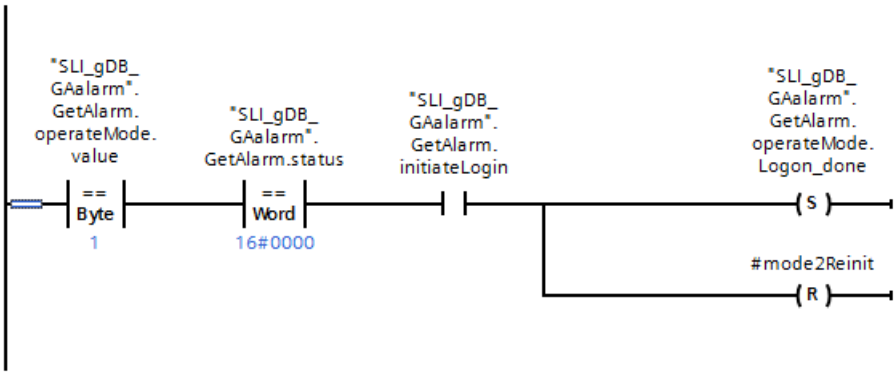

Netzwerk 4: Zum Umschalten in den Modus "Lesen" von Get_Alarm legen Sie die folgenden Verschaltungen an.

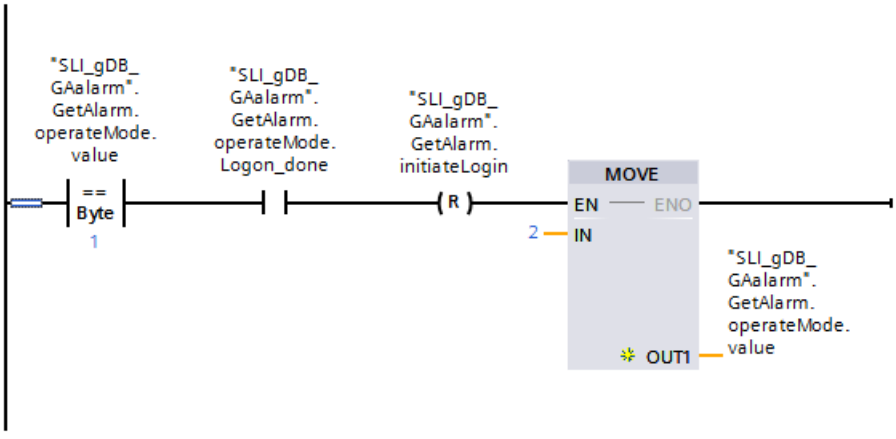

Netzwerk 5: Zur Verarbeitung des Modus "Lesen" von Get_Alarm legen Sie die folgenden Verschaltungen an.

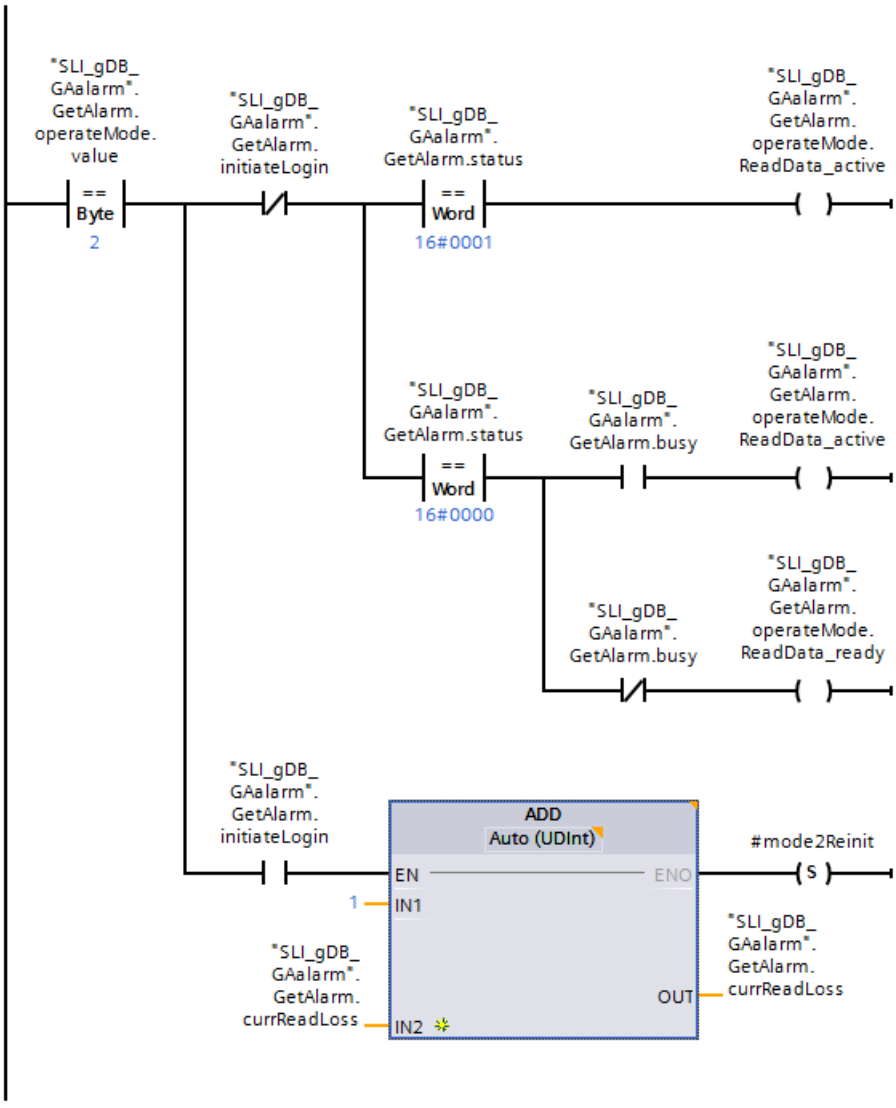

Netzwerk 6: Zur Verarbeitung des Modus "Lesen" von Get_Alarm legen Sie die folgenden Verschaltungen an (Teil 2).

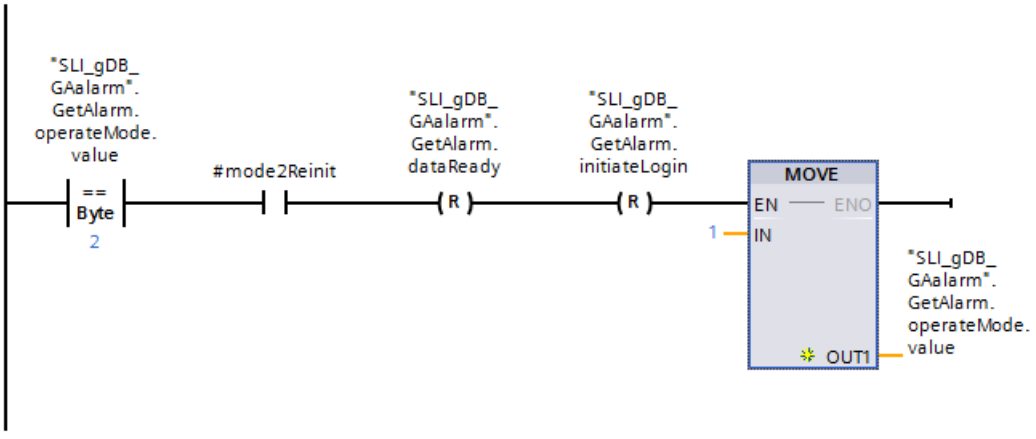

Netzwerk 7: Zum Beenden der Verarbeitung von Get_Alarm legen Sie die folgenden Verschaltungen an.

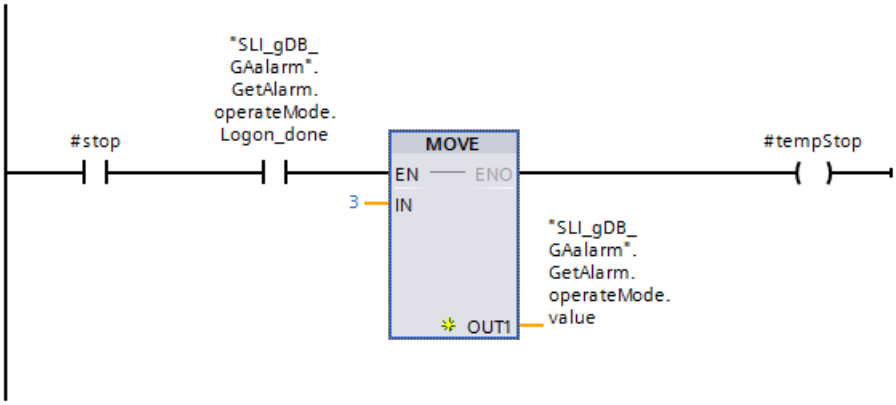

Netzwerk 8: Zum Beenden der Verarbeitung von Get_Alarm legen Sie die folgenden Verschaltungen an (Teil 2).

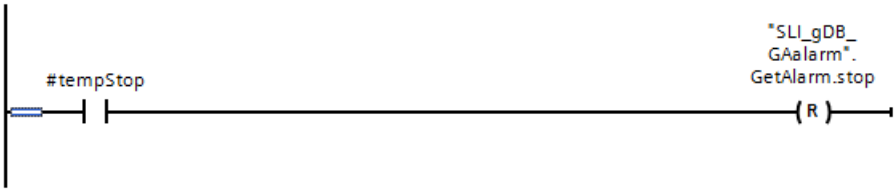

Netzwerk 9: Zum Beenden der Verarbeitung von Get_Alarm legen Sie die folgenden Verschaltungen an (Teil 3).

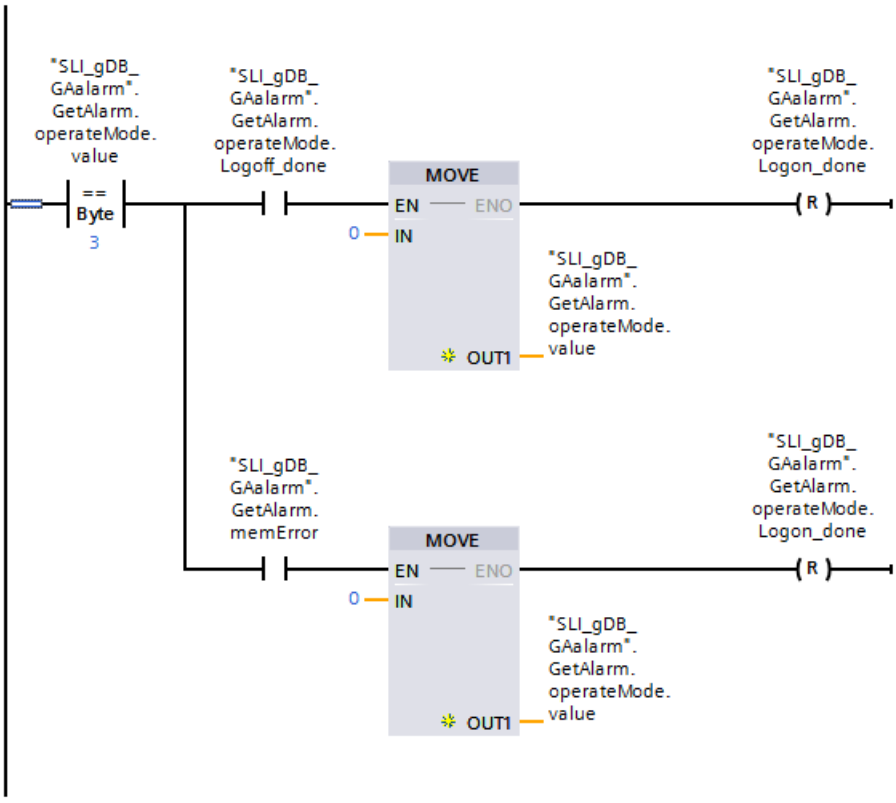

CPU 1: FB "SLI_FB_saveAlarms_GAA" - Parameter verschalten

Den FB "SLI_FB_saveAlarms_GAA" erstellen Sie zum Kopieren der Meldungen in das Speicher-Array "GetAlarm.alarmStorage".

Im FB legen Sie die folgenden lokalen Variablen an.

7	Static			
8	posForAlarm	UInt	0	Set in IDB
9	do	Bool	false	Set in IDB
10	do2	Bool	false	Set in IDB
11	nothing	Bool	false	Set in IDB
12	newRound	Bool	false	Set in IDB
13	dbComplete	Bool	false	Set in IDB
14	allDone	Bool	false	Set in IDB

Im FB legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Das Kopieren der Meldungen in das Speicher-Array "GetAlarm.alarmStorage" wird wie folgt entschieden:

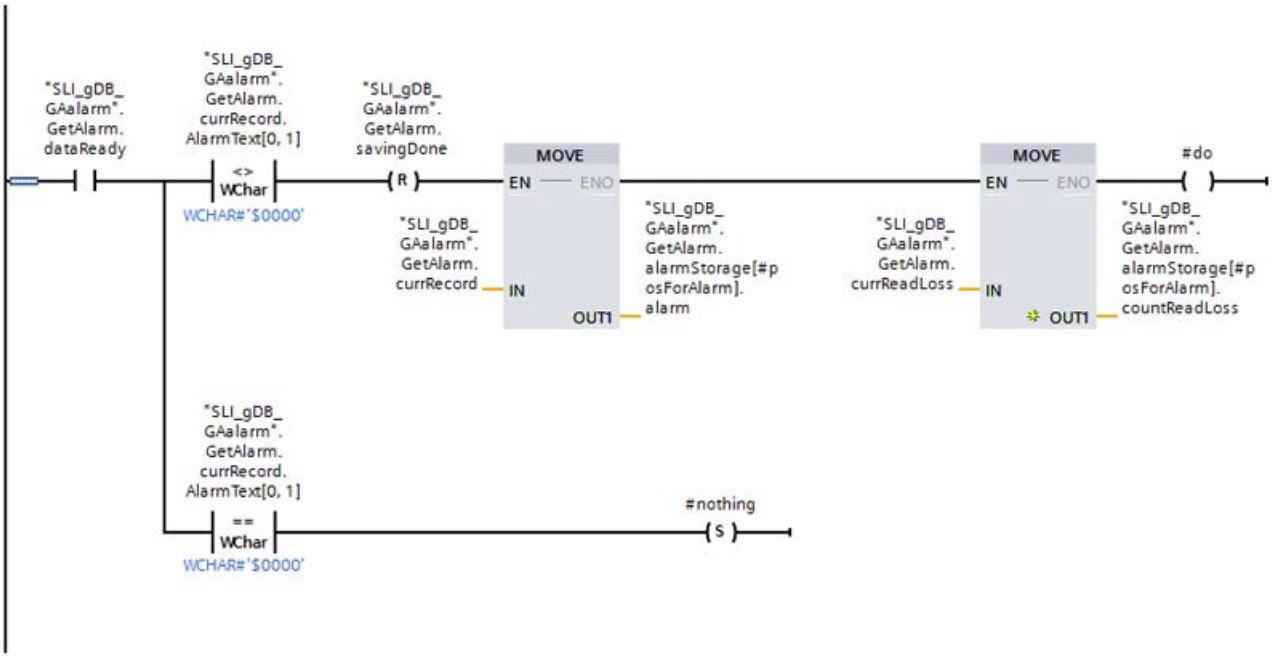

Netzwerk 2: Zum Speichern der zu letzt verwendeten Zellnummer des Speicher-Arrays und zum Zählen der Meldungen legen Sie die folgenden Verschaltungen an.

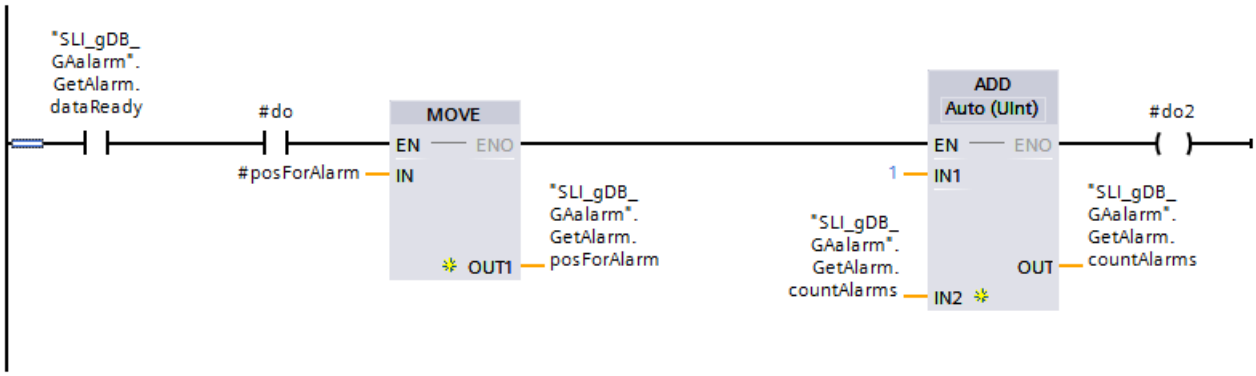

Netzwerk 3: Das Inkrementieren oder Rücksetzen der Zellnummer des Speicher-Arrays wird wie folgt entschieden:

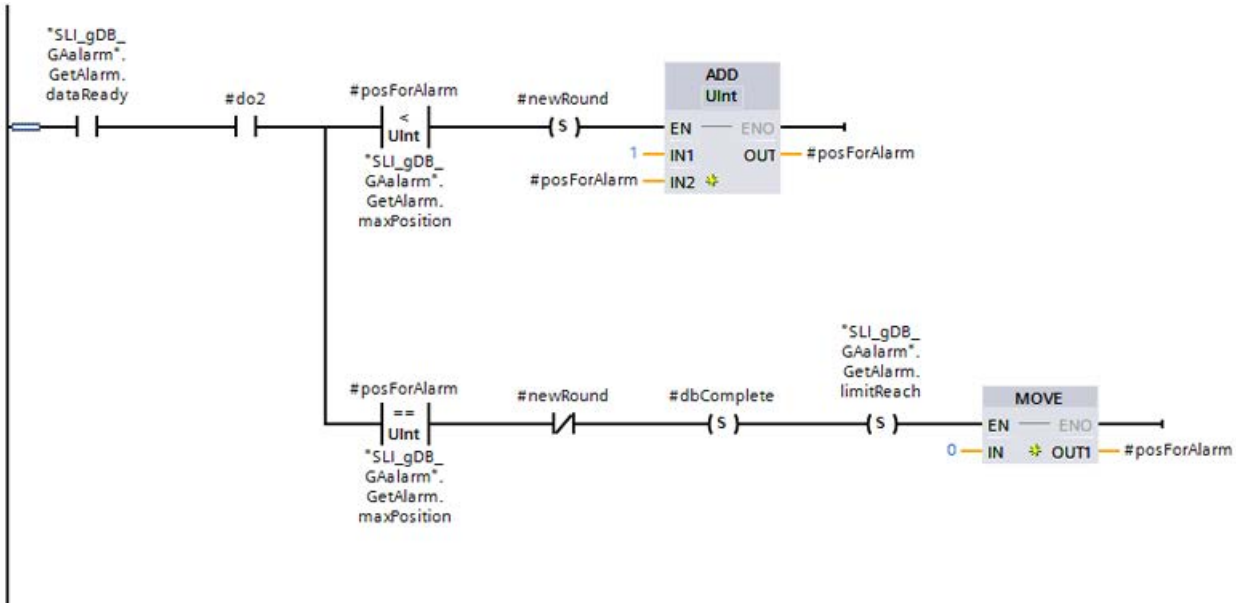

Netzwerk 4: Zum Speichern des Erfolgs beim Kopieren in das Speicher-Array legen Sie die folgenden Verschaltungen an.

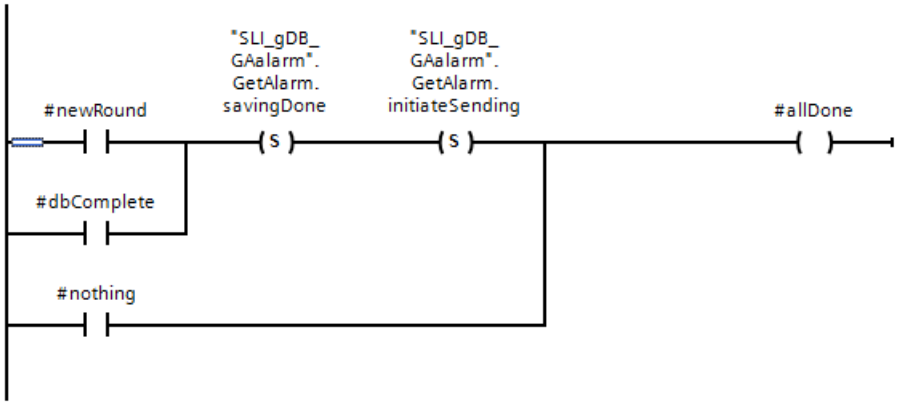

Netzwerk 5: Zum Beenden der Verarbeitung von "SLI_FB_saveAlarms_GAA" legen Sie die folgenden Verschaltungen an.

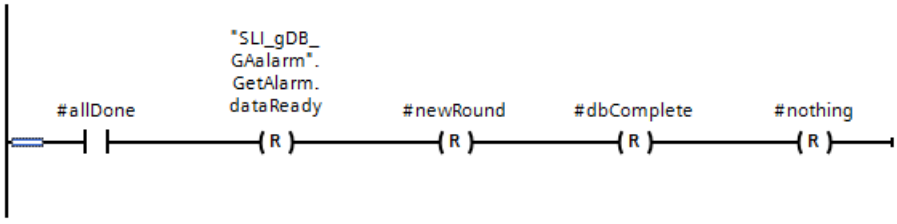

CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Parameter verschalten

Den FB "SLI_FBCall_GetAlarm_AckA" erstellen Sie für folgende Zwecke:

- Zum Aufrufen und Steuern von Get_Alarm.
- Zum Aufrufen der Bausteine für Kopier- und Sendeprozesse.
- Zum Aufrufen von Ack_Alarms.

Im FB legen Sie die folgenden lokalen Variablen an.

7	Static			
8	acknowledge	UInt	1	Non-ret...
9	dataReady	Bool	false	Non-retain
10	initiate	Bool	false	Non-retain
11	error	Bool	false	Non-retain
12	SLI_FB_send_GAA_Instance	"SLI_FB_send_GAA"		

Im FB legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Die Parameter des FBs "SLI_FB_send_GAA" verschalten Sie wie folgt.

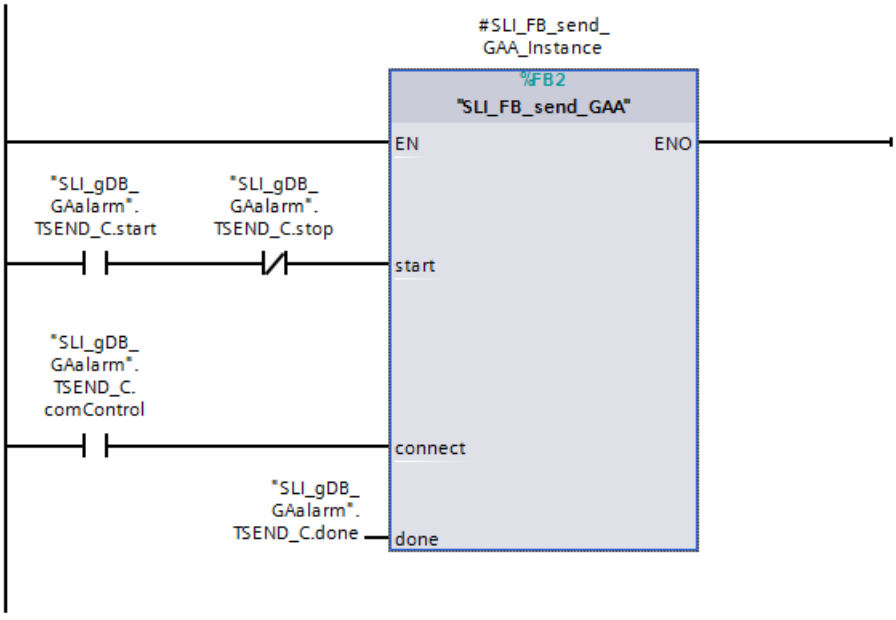

Netzwerk 2: Die Parameter des FCs "SLI_FC_switchMode_GAA" verschalten Sie wie folgt.

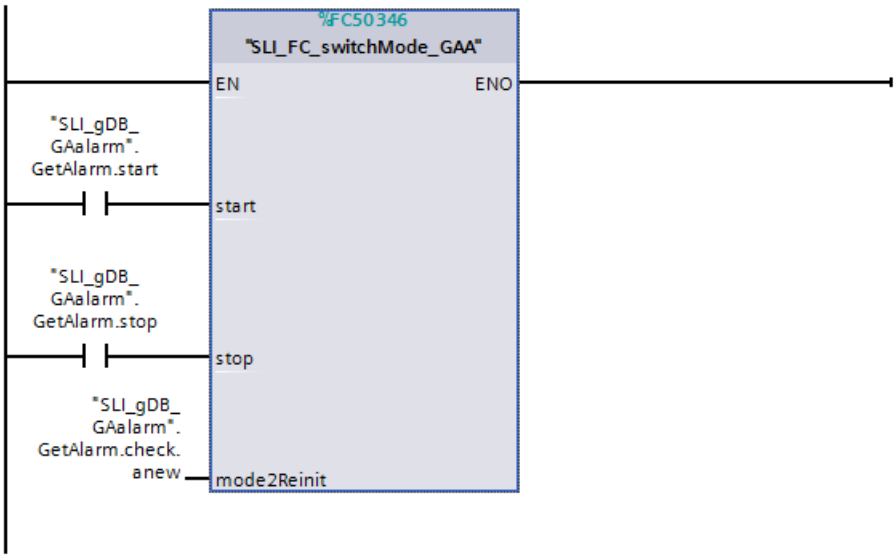

Netzwerk 3: Die Parameter der Anweisung "Get_Alarm" verschalten Sie wie folgt.

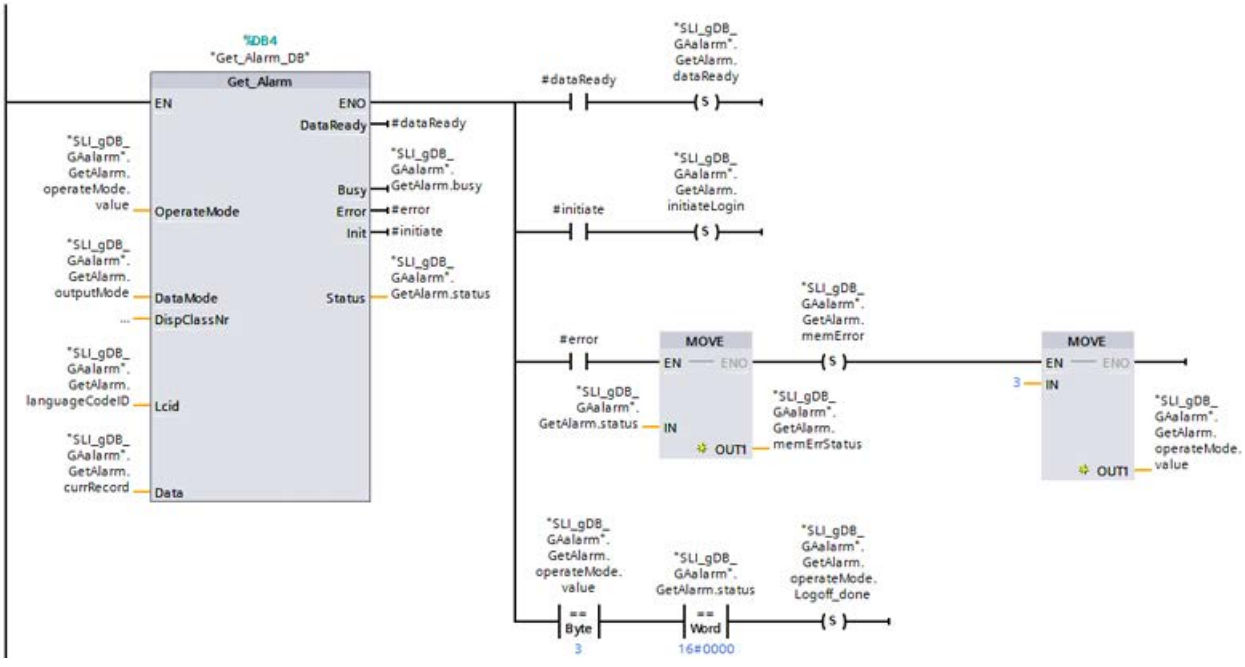

Netzwerk 4: Die Parameter des FBs "SLI_FB_saveAlarms_GAA" verschalten Sie wie folgt.

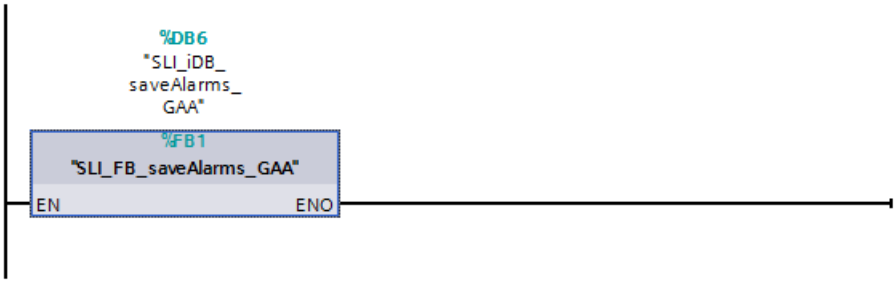

Netzwerk 5: Zum Testen der ProDiag-Meldung legen Sie die folgenden Verschaltungen an.

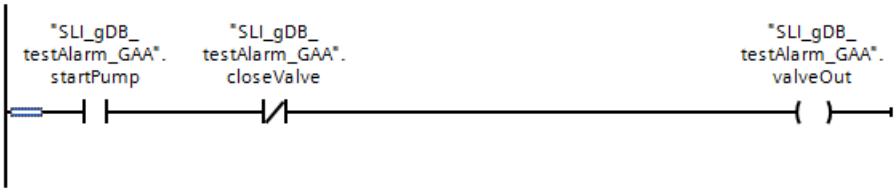

Netzwerk 6: Die Parameter der Anweisung "Ack_Alarms" verschalten Sie wie folgt.

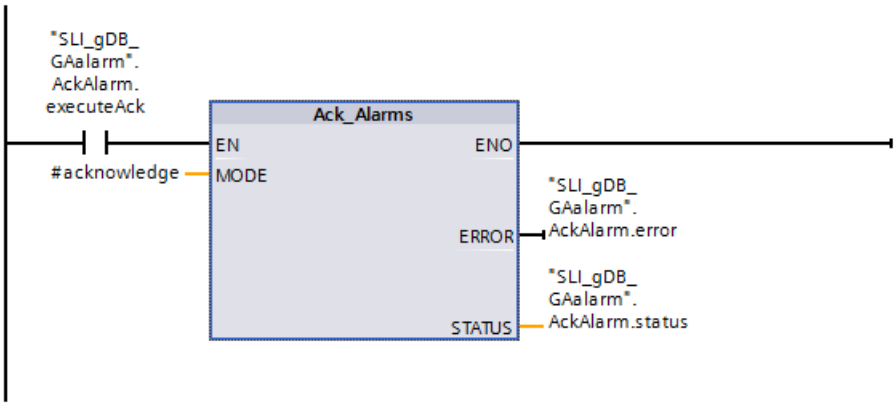

CPU 2: FB "SLI_FB_rcvdAlarms_GAA" - Parameter verschalten

Den FB "SLI_FB_rcvdAlarms_GAA" erstellen Sie zum Kopieren der Meldungen in das Speicher-Array "GetAlarm.rcvdStorage". Der Aufbau und die Verschaltungen gleichen denen des FBs "SLI_FB_saveAlarms_GAA".

Weitere Informationen zum Aufbau und den Verschaltungen des FBs "SLI_FB_rcvdAlarms_GAA" finden Sie im Programmcode des Beispiels in der Sample Library for Instructions (Seite 615).

CPU 2: FB "SLI_FBCall_trcv_GAA" - Parameter verschalten

Den FB "SLI_FBCall_trcv_GAA" erstellen Sie zum Aufrufen und Verarbeiten von TRCV_C sowie aller nachfolgenden Prozesse.

Im FB legen Sie die folgenden lokalen Variablen an.

7	Static			
8	TRCV_C_Instance	TRCV_C		
9	trcvStruct	"rcvGetAlarm"		Non-retain

Im FB legen Sie die folgenden Verschaltungen an.

Netzwerk 1: Die Parameter der Anweisung "TRCV_C" verschalten Sie wie folgt. Konfigurieren Sie den Parameter CONNECT von TRCV_C über den Wizard von TRCV_C.

Weitere Informationen zur Konfiguration von TRCV_C (Seite 3646) finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

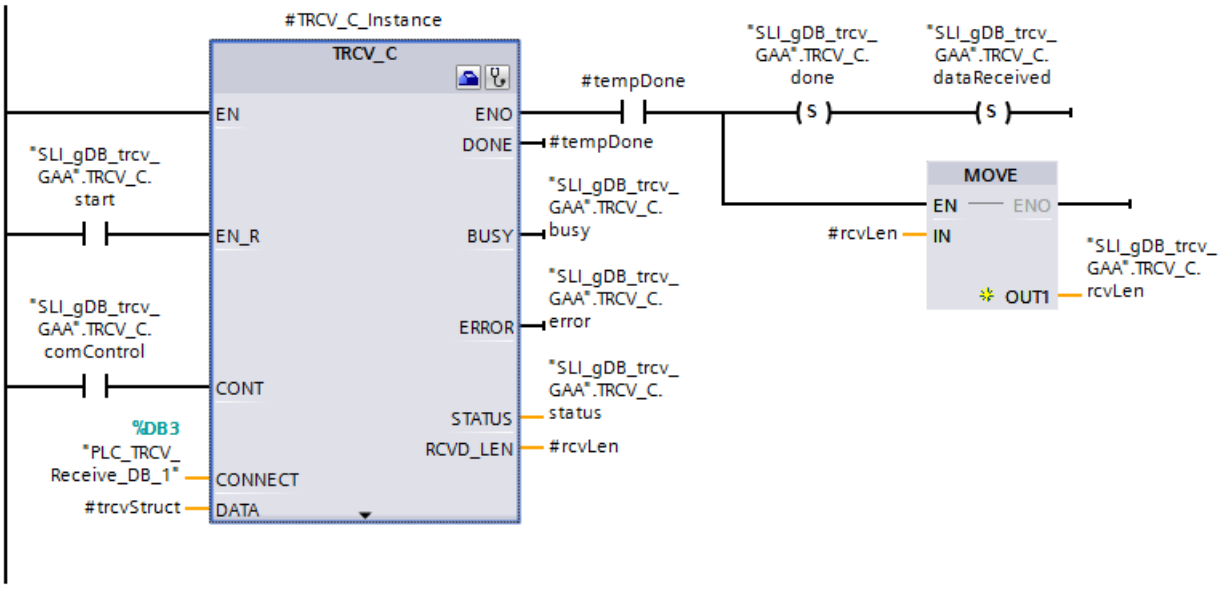

Netzwerk 2: Zum Speichern des Status im Fehlerfall von TRCV_C verschalten Sie den Parameter wie folgt.

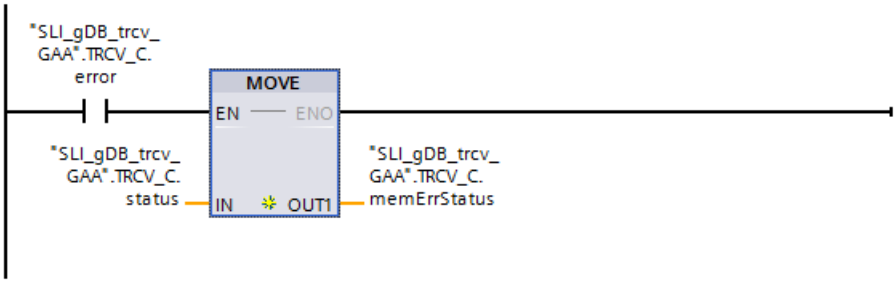

Netzwerk 3: Die Parameter des FBs "SLI_FB_rcvdAlarms_GAA" verschalten Sie wie folgt.

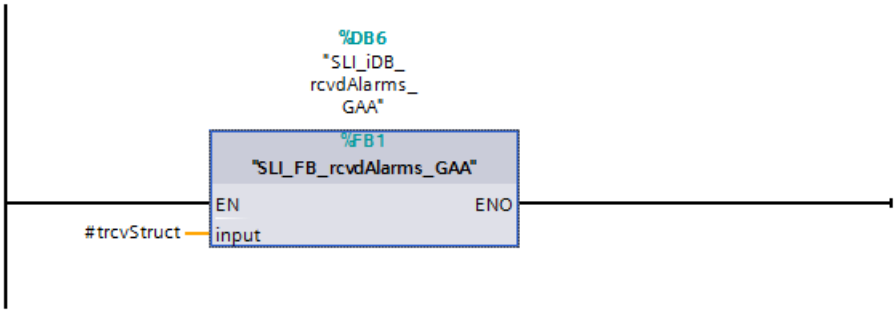

Ergebnisse und Vorgehen

Die Ergebnisse und das Vorgehen für die Kommunikationsverbindung finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 2 (Seite 2667).

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Ack_Alarms: Meldungen quittieren (Seite 2647)

Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 2

Einleitung

Im zweiten Teil des Programmbeispiels zu Get_Alarm und Ack_Alarms machen Sie sich mit den Ergebnissen der Projektierung und der ProDiag-Meldung vertraut.

Die Voraussetzungen und die Verschaltung der Parameter finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 1 (Seite 2649).

Vorgehen für Verbindungsaufbau zwischen CPU 1 und CPU 2

Beachten Sie die Reihenfolge der Handungsschritte:

1. Ändern Sie für CPU 2 die Variable "TRCV_C.start" auf "TRUE".
2. Ändern Sie für CPU 2 die Variable "TRCV_C.comControl" auf "TRUE".
3. Ändern Sie für CPU 1 die Variable "TSEND_C.start" auf "TRUE".
4. Ändern Sie für CPU 1 die Variable "TSEND_C.comControl" auf "TRUE".

Vorgehen für Verbindungsabbbau zwischen CPU 1 und CPU 2

Beachten Sie die Reihenfolge der Handungsschritte:

1. Ändern Sie für CPU 1 die Variable "TSEND_C.stop" auf "TRUE".
2. Ändern Sie für CPU 2 die Variable "TRCV_C.start" auf "FALSE".
3. Ändern Sie für CPU 2 die Variable "TRCV_C.comControl" auf "FALSE".
4. Ändern Sie für CPU 1 die Variable "TSEND_C.start" auf "FALSE" und die Variable "TSEND_C.stop" auf "FALSE".
5. Ändern Sie für CPU 1 die Variable "TSEND_C.comControl" auf "FALSE".

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 1

Wenn der Schließerkontakt ("TSEND_C.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TSEND_C" gestartet. Wenn der Eingangsparameter CONNECT ("TSEND_C.comControl") den Signalzustand "TRUE" liefert, erstellt die Anweisung "TSEND_C" eine Kommunikationsverbindung zwischen CPU 1 und CPU 2.

Über den Ausgangsparameter DONE ("TSEND_C.done") mit dem Signalzustand "TRUE" ("TSEND_C.status" hat den Wert "0000") wird die erfolgreiche Übermittlung des Datensatzes

(#sendStruct) angezeigt. Nach der Übermittlung des Datensatzes (#sendStruct) an die CPU 2 wird die Kommunikationsverbindung weiter überwacht (Status "7004").

Wenn der Schließerkontakt ("TSEND_C.stop") den Signalzustand "TRUE" liefert, wird die Anweisung "TSEND_C" gestoppt.

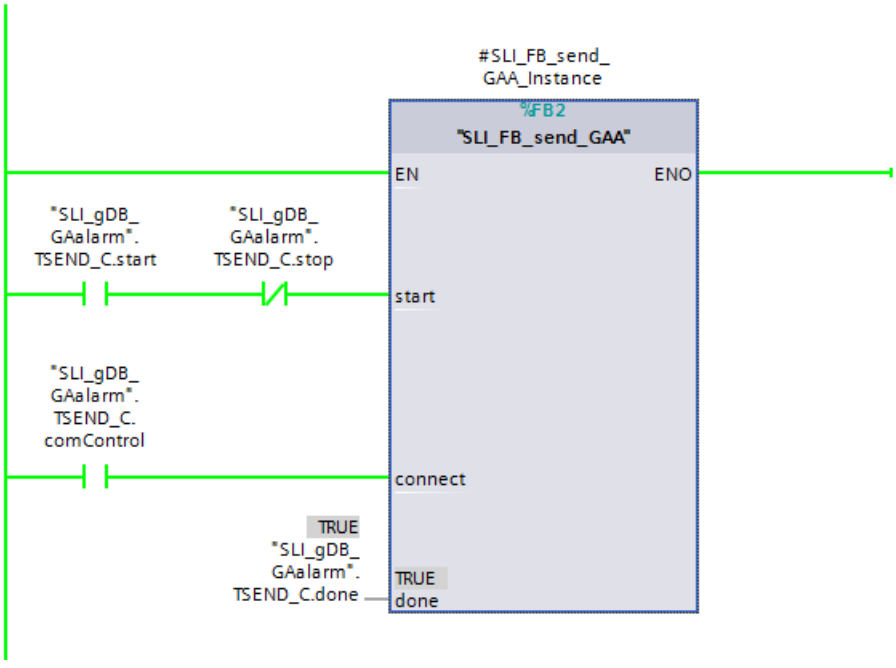

Weitere Informationen zur Funktionsweise von TSEND_C (Seite 3631) und TRCV_C (Seite 3646) finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 2

Wenn der Schließerkontakt ("GetAlarm.start") den Signalzustand "TRUE" liefert, wird die Anweisung "Get_Alarm" gestartet. Dazu wird die Anweisung "Get_Alarm" im Modus 1 ("GetAlarm.operateMode.value") aufgerufen und wechselt anschließend in den Modus 2. Nach ihrer Verarbeitung wird die Variable "GetAlarm.start" automatisch rückgesetzt auf "FALSE".

Wenn der Schließerkontakt ("GetAlarm.stop") den Signalzustand "TRUE" liefert, wird die Anweisung "Get_Alarm" gestoppt. Dazu wird die Anweisung "Get_Alarm" im Modus 3 ("GetAlarm.operateMode.value") aufgerufen und wechselt anschließend in die Inaktivität (Modus 0). Voraussetzung für den Stoppbefehl ist eine erfolgreiche Anmeldung von Get_Alarm

("GetAlarm.operateMode.Logon_done" ist "TRUE"). Nach ihrer Verarbeitung wird die Variable "GetAlarm.stop" automatisch rückgesetzt auf "FALSE".

Hinweis

Modus 0

Die Anweisung "Get_Alarm" gibt bei der Verwendung des Wert "0" am Parameter OPERATEMODE ("GetAlarm.operateMode.value") als Status einen Parametrierfehler aus. Der Grund: Die Anweisung "Get_Alarm" kennt keinen Modus 0. Der Wert "0" am Parameter OPERATEMODE ist aber notwendig, um in dem Beispiel die Anweisung "Get_Alarm" in einem Zustand vollständiger Inaktivität halten zu können. Andernfalls müsste die Anweisung "Get_Alarm" stets einen der Modi (Anmelden oder Abmelden) durchlaufen.

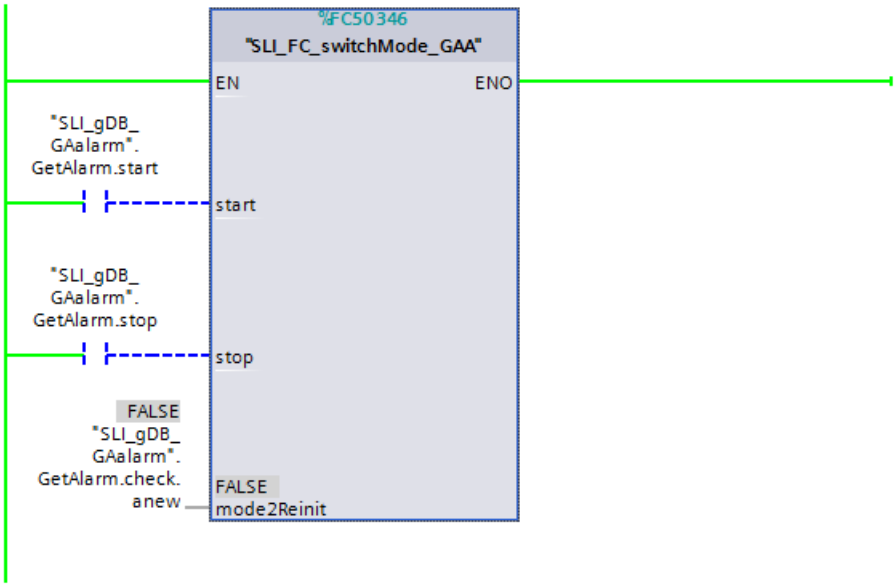

Der Erfolgszustand des verarbeiteten Modus von Get_Alarm wird jeweils festgehalten:

- "GetAlarm.operateMode.Logon_done" ist "TRUE" = der Modus 1 wurde erfolgreich beendet.
- "GetAlarm.operateMode.ReadData_ready" ist "TRUE" = der Modus 2 ist gesetzt. Get_Alarm ist lesebereit.
- "GetAlarm.operateMode.ReadData_active" ist "TRUE" = Get_Alarm ist im Modus 2 und beschäftigt mit dem Daten lesen.
- "GetAlarm.operateMode.Logoff_done" ist "TRUE" = der Modus 3 wurde erfolgreich beendet.

Hinweis

Nicht vorgesehenes Abmelden von Get_Alarm

Wenn Get_Alarm im Modus 2 abgemeldet wird, liefert der Ausgangsparameter INIT ("GetAlarm.initiateLogin") den Signalzustand "TRUE". Im FC "SLI_FC_switchMode_GAA" wird dieses Ereignis als Leseverlust ("GetAlarm.currReadLoss") gezählt und der (INOUT-)Parameter MODE2REINIT ("GetAlarm.check.anew") von "SLI_FC_switchMode_GAA" auf "TRUE" gesetzt. Dadurch werden die Variablen "GetAlarm.dataReady" und "GetAlarm.initiateLogin" rückgesetzt und die Anweisung "Get_Alarm" automatisch im Modus 1 neugestartet. Anschließend befindet sich Get_Alarm wieder lesebereit im Modus 2.

18		currReadLoss	UDInt	0	0
19		dataReady	Bool	false	FALSE
20		initiateLogin	Bool	false	FALSE
21		busy	Bool	false	FALSE
22		status	Word	16#0	16#0000
23		memError	Bool	false	FALSE
24		memErrStatus	Word	16#0	16#0000

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 3

Wenn der Eingangsparameter OPERATEMODE ("GetAlarm.operateMode.value") den Wert "2" zeigt, liest die Anweisung "Get_Alarm" alle Meldungen der CPU 1 aus. Dazu wird am Eingangsparameter DATA ("GetAlarm.currRecord") die jeweilige Meldung erfasst.

SLI_gDB_GAalarm

	Name	Data type	Start value	Monitor value
1	Static			
2	TSEND_C	Struct		
3	GetAlarm	Struct		
4	start	Bool	false	FALSE
5	stop	Bool	false	FALSE
6	check	Struct		
7	anew	Bool	false	FALSE
8	operateMode	Struct		
9	value	Byte	16#0	16#02
10	Logon_done	Bool	false	TRUE
11	ReadData_ready	Bool	false	TRUE
12	ReadData_active	Bool	false	FALSE
13	Logoff_done	Bool	false	FALSE
14	outputMode	Byte	16#3	16#03
15	languageCodeID	UDInt	0	0
16	currRecord	AlarmData		

Am Ausgangsparameter DATAREADY ("#dataReady") wird mit dem Signalzustand "TRUE" auf eine neue Meldung hingewiesen. Da der Wert von "#dataReady" (wie auch von "#initiate) nur einen Zyklus lang gesetzt ist, wird der Zustand über die Variable "GetAlarm.dataReady" gespeichert.

Wenn der Ausgangsparameter INIT ("#initiate") im Modus 2 den Signalzustand "TRUE" liefert, wurde Get_Alarm abgemeldet und muss neu angemeldet werden. Siehe dazu den Hinweis zum Netzwerk 2. Im Modus 1 zeigt "#initiate" / "GetAlarm.initiateLogin" die erfolreiche Anmeldung an.

Im Fehlerfall wird der Status von Get_Alarm ("GetAlarm.status") über die Variablen "GetAlarm.memErrStatus" und "GetAlarm.memErr" gespeichert und Get_Alarm zum Abmelden in den Modus 3 versetzt. Wenn Get_Alarm abgemeldet werden soll und kein Fehler auftritt ("GetAlarm.status" hat den Wert "16#0000"), wird das Abmelden durch "GetAlarm.operateMode.Logoff_done" mit dem Wert "TRUE" gespeichert.

3	GetAlarm	Struct		
4	start	Bool	false	FALSE
5	stop	Bool	false	FALSE
6	check	Struct		
7	anew	Bool	false	FALSE
8	operateMode	Struct		
9	value	Byte	16#0	16#00
10	Logon_done	Bool	false	FALSE
11	ReadData_ready	Bool	false	FALSE
12	ReadData_active	Bool	false	FALSE
13	Logoff_done	Bool	false	TRUE
14	outputMode	Byte	16#3	16#03

Über den Eingangsparameter DATAMODE ("GetAlarm.outputMode") mit dem Wert "16#03" wird vorgegeben, dass die Meldungen komplett (mit allen Informationen) ausgelesen werden sollen. Der Eingangsparameter DISPCLASSNR ist unverschalten. Dadurch liest Get_Alarm alle Anzeigeklassen aus. Über den Eingangsparameter LCID ("GetAlarm.languageCodeID") mit dem Wert "0" wird vorgegeben, dass alle Sprachversionen von Meldungen gelesen werden sollen.

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 4

Unter den folgenden Bedingungen werden die aktuell gelesene Meldung ("GetAlarm.currRecord") sowie die bis dahin gezählten Leseverluste ("GetAlarm.currReadLoss") als ein zusammengesetzter Datensatz in dem Speicher-Array "GetAlarm.alarmStorage" gespeichert:

- "GetAlarm.dataReady" liefert den Signalzustand "TRUE". Dementsprechend hat Get_Alarm eine neue Meldung gelesen.
- Zudem ist der Datensatz der gelesenen Meldung ("GetAlarm.currRecord") nicht leer. Dazu wird dessen Meldetext geprüft.

26	alarmStorage	Array[0..15] of "SG...		
27	alarmStorage[0]	"SGetAlarm"		
28	countReadL...	UDInt	0	0
29	alarm	AlarmData		
30	Produce...	UInt	0	9
31	ID_1	UInt	0	53
32	ID_2	UInt	0	1
33	PRIO	UInt	0	0
34	TimeSta...	LDT	LDT#1970-01-01-(	LDT#2016-06-22-0...
35	State	Byte	16#0	16#01
36	LCID	Array[0..2] of UInt		
37	LCID...	UInt	0	0
38	LCID...	UInt	0	0
39	LCID...	UInt	0	0
40	AlarmText	Array[0..2] of WStri...		
41	Alar...	WString[254]	WSTRING#''	WSTRING#'Warning ...

Nachfolgend wird die verwendete Position ("#posForAlarm") im Speicher-Array "GetAlarm.alarmStorage" in der globalen Variable "GetAlarm.posForAlarm" gespeichert. Außerdem werden die Meldungen gezählt und der aktuelle Wert über die globale Variable "GetAlarm.countAlarms" festgehalten.

Die Positionsangabe ("#posForAlarm") wird inkrementiert, sodass eine neue Meldung in einer anderen Zelle im Speicher-Array gespeichert werden kann. Voraussetzung für das Inkrementieren ist, dass der Grenzbereich des Speicher-Arrays ("GetAlarm.maxPositon") nicht überschritten wird. Bei Überschreiten des Grenzbereichs wird dieser Umstand vermerkt ("GetAlarm.limitReach") und die Positionsangabe ("#posForAlarm") auf den Wert "0" gesetzt. Dadurch würden nachfolgende Meldungen alte Meldeeinträge im Speicher-Array überschreiben.

Der Erfolg beim Kopieren wird über die Variable "GetAlarm.savingDone" vermerkt. Die Variable "GetAlarm.initiateSending" wird gesetzt, um nachfolgend die Übermittlung der Meldung zu initiieren.

Zum Abschluss der Verarbeitung von "SLI_FB_saveAlarms_GAA" wird die Variable "GetAlarm.dataReady" (und lokale Variablen) auf "FALSE" rückgesetzt. Das Programm von "SLI_FB_saveAlarms_GAA" wird nur dann wieder abgearbeitet, wenn "GetAlarm.dataReady" den Wert "TRUE" liefert.

25	maxPosition	UInt	15	15
26	alarmStorage	Array[0..15] of "SG...		
27	limitReach	Bool	false	FALSE
28	savingDone	Bool	false	TRUE
29	initiateSending	Bool	false	FALSE
30	countAlarms	UInt	0	1
31	posForAlarm	UInt	0	0

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 5

Wenn sowohl Schließerkontakt ("startPump") als auch Öffnerkontakt ("closeValve") den Signalzustand "TRUE" liefern, wird die projektierte ProDiag-Meldung ausgelöst.

Get_Alarm liest die Meldung aus und speichert diese in der zugewiesenen Variable ab ("GetAlarm.currRecord"). Im Netzwerk 4 wird die Meldung nachfolgend in das Speicher-Array ("GetAlarm.alarmStorage") kopiert.

17	currRecord	AlarmData		
18	ProducerID	UInt	0	9
19	ID_1	UInt	0	53
20	ID_2	UInt	0	1
21	PRIO	UInt	0	0
22	TimeStamp	LDT	LDT#1970-01-01-4	LDT#2016-06-22-0...
23	State	Byte	16#0	16#01
24	LCID	Array[0..2] of UInt		
25	LCID[0]	UInt	0	0
26	LCID[1]	UInt	0	0
27	LCID[2]	UInt	0	0
28	AlarmText	Array[0..2] of WStri...		
29	AlarmText[0]	WString[254]	WSTRING#''	WSTRING#'Warning ...
30	AlarmText[1]	WString[254]	WSTRING#''	WSTRING#''
31	AlarmText[2]	WString[254]	WSTRING#''	WSTRING#''

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 1 (Datenübermittlung)

Wenn die Variable "GetAlarm.initiateSending" den Signalzustand "TRUE" liefert, wird die gespeicherte Position ("GetAlarm.posForAlarm") im Speicher-Array einer lokalen Variablen gleichen Namens ("#posForAlarm") übergeben und der Befehl für eine neue Datenübermittlung weitergeleitet. Die Variable "#initiateSending" wird auf "TRUE" gesetzt.

Wenn zudem kein Fehler und TSEND_C als wachend gemeldet wird ("TSEND_C.error" und "TSEND_C.busy" sind "FALSE"), wird die jeweilige Meldung ("GetAlarm.alarmStorage[..]") in gekürzter Form der Struktur "#sendStruct" übergeben. Sofern kein Stoppbefehl ("TSEND_C.stop" auf "TRUE") gegeben wurde, wird die Variable "TSEND_C.start" auf "TRUE" gesetzt.

Durch den Startbefehl zur Datenübermittlung ("TSEND_C.start" ist "TRUE") wird die gekürzte Meldung ("#sendStruct") über mehrere Zyklen an die CPU 2 gesandt.

Ergebnis der CPU 1: FB "SLI_FBCall_GetAlarm_AckA" - Netzwerk 6

Wenn der Schließerkontakt ("AckAlarm.executeAck") den Signalzustand "TRUE" liefert, wird die Anweisung "Ack_Alarms" gestartet. Entsprechend dem Wert "1" des Eingangsparameters MODE ("#acknowledge") bestätigt Ack_Alarms alle aktiven Meldungen. Die Verarbeitung im Beispiel verläuft ohne Fehler ("AckAlarm.error" ist "FALSE" und "AckAlarm.status" hat den Wert "16#0000").

Um sich die Bestätigungen anzuschauen, verwenden Sie ein HMI-Device oder den Webserver der CPU.

Ergebnis der CPU 2: FB "SLI_FBCall_trcv_GAA" - Netzwerk 1

Wenn der Schließerkontakt ("TRCV_C.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TRCV_C" gestartet. Wenn der Eingangsparameter CONNECT ("TRCV_C.comControl") dem Signalzustand "TRUE" liefert, erstellt die Anweisung "TRCV_C" eine Kommunikationsverbindung zwischen CPU 2 und CPU 1.

Über den Ausgangsparameter DONE ("TRCV_C.done" und "TRCV_C.dataReceived") mit dem Signalzustand "TRUE" ("TRCV_C.status" hat den Wert "0000") wird die erfolgreiche Übermittlung des Datensatzes (#trcvStruct) angezeigt. Nach der Übermittlung des Datensatzes (#trcvStruct) an die CPU 2 wird die Kommunikationsverbindung weiter überwacht und TRCV_C ist empfangsbereit (Status "7006"). Die Länge in Byte des tatsächlich übermittelten Datensatzes wird über den Ausgangsparameter RCVD_LEN ("TRCV_C.rcvLen") erfasst.

SLI_gDB_trcv_GAA

	Name	Data type	Start value	Monitor value
1	Static			
2	TRCV_C	Struct		
3	start	Bool	false	TRUE
4	comControl	Bool	false	TRUE
5	done	Bool	false	TRUE
6	busy	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7006
9	memErrStatus	Word	16#0	16#0000
10	rcvLen	UDInt	0	540
11	dataReceived	Bool	false	FALSE
12	GetAlarm	Struct		
13	maxPosition	UInt	15	15
14	rcvdStorage	Array[0..15] of "rcv...		
15	limitReach	Bool	false	FALSE
16	savingDone	Bool	false	FALSE
17	countAlarms	UInt	0	0
18	posForAlarm	UInt	0	0

Weitere Informationen zur Funktionsweise von TSEND_C (Seite 3631) und TRCV_C (Seite 3646) finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

Ergebnis der CPU 2: FB "SLI_FBCall_trcv_GAA" - Netzwerk 2

Wenn der Schließerkontakt ("TRCV_C.error") den Signalzustand "TRUE" liefert, wird der Status ("TRCV_C.status") des Fehlerfalls gespeichert ("TRCV_C.memErrStatus").

Ergebnis der CPU 2: FB "SLI_FBCall_trcv_GAA" - Netzwerk 3

Unter den folgenden Bedingungen wird eine Meldung in dem Speicher-Array "GetAlarm.rcvdStorage" gespeichert:

- "TRCV_C.dataReceived" liefert den Signalzustand "TRUE". Dementsprechend wurde ein neuer Datensatz an TRCV_C gesandt.
- Zudem ist der neu empfangene Datensatz (#trcvStruct) nicht leer.

12	GetAlarm	Struct		
13	maxPosition	UInt	15	15
14	rcvdStorage	Array[0..15] of "rcv...		
15	rcvdStorage[0]	"rcvGetAlarm"		
16	countReadL...	UDInt	0	0
17	ProducerID	UInt	0	9
18	ID_1	UInt	0	53
19	ID_2	UInt	0	1
20	PRIO	UInt	0	0
21	TimeStamp	LDT	LDT#1970-01-01-(	LDT#2016-06-22-0...
22	State	Byte	16#0	16#01
23	LCID	Array[0..2] of UInt		
24	AlarmText	WString[254]	WSTRING#''	WSTRING#'Warning ...

Nachfolgend wird die verwendete Position ("#posForAlarm") im Speicher-Array "GetAlarm.rcvdStorage" in der globalen Variable "GetAlarm.posForAlarm" gespeichert. Außerdem werden die Meldungen gezählt und der aktuelle Wert über die globale Variable "GetAlarm.countAlarms" festgehalten.

Die Positionsangabe ("#posForAlarm") wird inkrementiert, sodass eine neue Meldung in einer anderen Zelle im Speicher-Array gespeichert werden kann. Voraussetzung für das Inkrementieren ist, dass der Grenzbereich des Speicher-Arrays ("GetAlarm.maxPositon") nicht überschritten wird. Bei Überschreiten des Grenzbereichs wird dieser Umstand vermerkt ("GetAlarm.limitReach") und die Positionsangabe ("#posForAlarm") auf den Wert "0" gesetzt. Dadurch würden nachfolgende Meldungen alte Meldeeinträge im Speicher-Array überschreiben.

Der Erfolg beim Kopieren wird über die Variable "GetAlarm.savingDone" vermerkt.

Zum Abschluss der Verarbeitung von "SLI_FB_rcvdAlarms_GAA" wird die Variable "TRCV_C.dataReceived" (und lokale Variablen) auf "FALSE" rückgesetzt. Das Programm von "SLI_FB_rcvdAlarms_GAA" wird nur dann wieder abgearbeitet, wenn "TRCV_C.dataReceived" den Wert "TRUE" liefert.

SLI_gDB_trcv_GAA

	Name	Data type	Start value	Monitor value
1	Static			
2	TRCV_C	Struct		
3	start	Bool	false	TRUE
4	comControl	Bool	false	TRUE
5	done	Bool	false	TRUE
6	busy	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7006
9	memErrStatus	Word	16#0	16#0000
10	rcvLen	UDInt	0	540
11	dataReceived	Bool	false	FALSE
12	GetAlarm	Struct		
13	maxPosition	UInt	15	15
14	rcvdStorage	Array[0..15] of "rcv...		
15	limitReach	Bool	false	FALSE
16	savingDone	Bool	false	TRUE
17	countAlarms	UInt	0	1
18	posForAlarm	UInt	0	0

Teil 1 des Programmbeispiels

Die Voraussetzungen und die Verschaltung der Parameter finden Sie hier: Programmbeispiel für Get_Alarm & Ack_Alarms - Teil 1 (Seite 2649).

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

4.1.4.9 Diagnose

RD_SINFO: Startinformation des aktuellen OBs auslesen

Beschreibung

Mit der Anweisung "RD_SINFO" lesen Sie die Startinformation

- des zuletzt aufgerufenen OBs, der noch nicht vollständig abgearbeitet wurde oder
- des zuletzt gestarteten Anlauf-OBs.

Beide Startinformationen enthalten keinen Zeitstempel. Erfolgt der Aufruf im OB 100, OB 101 oder OB 102, werden zwei identische Startinformationen zurückgeliefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_SINFO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
TOP_SI	Output	VARIANT	D, L	Startinformation des aktuellen OBs
START_UP_SI	Output	VARIANT	D, L	Startinformation des zuletzt gestarteten Anlauf-OBs

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

SDTs des Parameters TOP_SI

Die folgende Tabelle zeigt die möglichen SDTs des Parameters TOP_SI:

Organisationsbausteine (OB)	Systemdatentypen (SDT)	Nummern der Systemdatentypen
Alle	SI_classic	592
	SI_none	593
ProgramCycleOB	SI_ProgramCycle	594
TimeOfDayOB	SI_TimeOfDay	595
TimeDelayOB	SI_Delay	596
CyclicOB	SI_Cyclic	597
ProcessEventOB	SI_HWInterrupt	598
	SI_HWInterrupt_Extended	616
ProfileEventOB StatusEventOB UpdateEventOB	SI_Submodule	601
SynchronousCycleOB	SI_SynchCycle	602
IOredundancyErrorOB	SI_IORedundancyError	604
CPUredundancyErrorOB	SI_CPURedundancyError	605
TimeErrorOB	SI_TimeError	606
DiagnosticErrorOB	SI_DiagnosticInterrupt	607
PullPlugEventOB	SI_PlugPullModule	608
PeripheralAccessErrorOB	SI_AccessError	609
RackStationFailureOB	SI_StationFailure	610
ServoOB	SI_Servo	611
IpoOB	SI_Ipo	612
StartupOB	SI_Startup	613
ProgrammingErrorOB IOaccessErrorOB	SI_ProgIOAccessError	614
CPURedundancyErrorOB	SI_CPURedundancyError_V2	617

SDTs des Parameters START_UP_SI

Die folgende Tabelle zeigt die möglichen SDTs des Parameters START_UP_SI:

Systemdatentypen (SDT)	Nummern der Systemdatentypen
SI_classic	592
SI_none	593
SI_Startup	613

Strukturen

Die folgenden Tabellen zeigen die Bedeutung der Strukturelemente der einzelnen Strukturen:

Tabelle 4-36 Struktur SI_classic

Strukturelement	Datentyp	Beschreibung
EV_CLASS	BYTE	• Bits 0 bis 3: Ereigniskennung • Bits 4 bis 7: Ereignisklasse
EV_NUM	BYTE	Ereignisnummer
PRIORITY	BYTE	Nummer der Ablaufebene (Bedeutung von B#16#FE: OB nicht vorhanden oder gesperrt oder im aktuellen Betriebszustand nicht startbar.)
NUM	BYTE	OB-Nummer
TYP2_3	BYTE	Datenkennung 2_3: Kennzeichnet die in ZI2_3 eingetragene Information
TYP1	BYTE	Datenkennung 1: Kennzeichnet die in ZI1 eingetragene Information
ZI1	WORD	Zusatzinformation 1
ZI2_3	DWORD	Zusatzinformation 2_3

Tabelle 4-37 Struktur SI_none

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)

Tabelle 4-38 Struktur SI_ProgramCycle

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 1	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)

Strukturelement	Datentyp	Beschreibung
Initial_Call	BOOL	Für OB_Class = 1, 30, 52, 61, 65
Remanence	BOOL	Für OB_Class = 1

Tabelle 4-39 Struktur SI_TimeOfDay

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 10	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
CaughtUp	BOOL	Für OB_Class = 10
SecondTime	BOOL	Für OB_Class = 10

Tabelle 4-40 Struktur SI_Delay

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 20	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Sign	WORD	Für OB_Class = 20

Tabelle 4-41 Struktur SI_Cyclic

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 30	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Initial_Call	BOOL	Für OB_Class = 1, 30, 52, 61, 65
Event_Count	INT	Für OB_Class = 30, 51, 52, 61, 65, 91, 92

Tabelle 4-42 Struktur SI_HWInterrupt

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 40	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_IO	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
USI	WORD	Für OB_Class = 40

Strukturelement	Datentyp	Beschreibung
IChannel	USINT	Für OB_Class = 40
EventType	BYTE	Für OB_Class = 40

Tabelle 4-43 Struktur SI_HWInterrupt_Extended

Strukturelement	Datentyp	Byte	Beschreibung
SI_Format	USINT	0	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 40	1	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	2	OB-Nummer (1 ... 32767)
LADDR	HW_IO	4	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
USI	WORD	6	Für OB_Class = 40
IChannel	USINT	8	Für OB_Class = 40
EventType	BYTE	9	Für OB_Class = 40
PointAddr	DWORD	12	Für OB_Class = 40

Tabelle 4-44 Struktur SI_Submodule

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_IO	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Slot	UINT	Für OB_Class = 55, 56, 57
Specifier	WORD	Für OB_Class = 55, 56, 57

Tabelle 4-45 Struktur SI_SynchCycle

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 61	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Initial_Call	BOOL	Für OB_Class = 1, 30, 52, 61, 65
PIP_Input	BOOL	Für OB_Class = 61, 91, 92
PIP_Output	BOOL	Für OB_Class = 61, 91, 92
IO_System	USINT	Für OB_Class = 61, 91, 92
Event_Count	INT	Für OB_Class = 30, 51, 52, 61, 65, 91, 92
SyncCycleTime	LTIME	Berechnete Zykluszeit

Tabelle 4-46 Struktur SI_IORedundancyError

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 70	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_ANY	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Für OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86

Tabelle 4-47 Struktur SI_CPURedundancyError

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 72	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Switch_Over	BOOL	Für OB_Class = 72

Tabelle 4-48 Struktur SI_TimeError

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 80	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86
Csg_OBnr	OB_ANY	Für OB_Class = 80
Csg_Prio	UINT	Für OB_Class = 80

Tabelle 4-49 Struktur SI_DiagnosticInterrupt

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 82	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
IO_State	WORD	Für OB_Class = 82
LADDR	HW_ANY	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Channel	UINT	Für OB_Class = 82
MultiError	BOOL	Für OB_Class = 82

Tabelle 4-50 Struktur SI_PlugPullModule

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 83	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_IO	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Für OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86

Tabelle 4-51 Struktur SI_AccessError

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 85	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_IO	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Für OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86
IO_Addr	UINT	Für OB_Class = 85
IO_LEN	UINT	Für OB_Class = 85

Tabelle 4-52 Struktur SI_StationFailure

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 86	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_IO	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Für OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86

Tabelle 4-53 Struktur SI_Servo

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 91	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Initial_Call	BOOL	Für OB_Class = 1, 30, 52, 61, 65

Strukturelement	Datentyp	Beschreibung
PIP_Input	BOOL	Für OB_Class = 61, 91, 92
PIP_Output	BOOL	Für OB_Class = 61, 91, 92
IO_System	USINT	Für OB_Class = 61, 91, 92
Event_Count	INT	Für OB_Class = 30, 51, 52, 61, 65, 91, 92
Synchronous	BOOL	

Tabelle 4-54 Struktur SI_Ipo

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 92	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
Initial_Call	BOOL	Für OB_Class = 1, 30, 52, 61, 65
PIP_Input	BOOL	Für OB_Class = 61, 91, 92
PIP_Output	BOOL	Für OB_Class = 61, 91, 92
IO_System	USINT	Für OB_Class = 61, 91, 92
Event_Count	INT	Für OB_Class = 30, 51, 52, 61, 65, 91, 92
Reduction	UINT	Für OB_Class = 92

Tabelle 4-55 Struktur SI_Startup

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 100	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LostRetentive	BOOL	Für OB_Class = 100
LostRTC	BOOL	Für OB_Class = 100

Tabelle 4-56 Struktur SI_ProgIOAcessError

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
BlockNr	UINT	Für OB_Class = 121, 122
Reaction	USINT	Für OB_Class = 121, 122
Fault_ID	BYTE	Für OB_Class = 121, 122
BlockType	USINT	Für OB_Class = 121, 122
Area	USINT	Für OB_Class = 121, 122

Strukturelement	Datentyp	Beschreibung
DBNr	DB_ANY	Für OB_Class = 121, 122
Csg_OBNr	OB_ANY	Für OB_Class = 121, 122
Csg_Prio	USINT	Für OB_Class = 121, 122
Width	USINT	Für OB_Class = 121, 122

Tabelle 4-57 Struktur SI_CPURedundancyError_V2

Strukturelement	Datentyp	Beschreibung
SI_Format	USINT	• 16#FF = Keine Angabe • 16#FE = Optimierte Startinformation
OB_Class	USINT := 72	OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"
OB_Nr	UINT	OB-Nummer (1 ... 32767)
LADDR	HW_ANY	Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Für OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Für OB_Class = 70, 80, 83, 85, 86

Hinweis

Wenn dieser mit der Bausteineigenschaft "Standard" erstellt wurde, entsprechen die angegebenen Strukturelemente der Struktur SI_classic inhaltlich genau den temporären Variablen eines OBs.

Beachten Sie jedoch, dass die temporären Variablen in den einzelnen OBs andere Namen und andere Datentypen haben können. Beachten Sie weiterhin, dass die Aufrufschnittstelle der OBs zusätzlich Datum und Uhrzeit, zu denen der OB angefordert wurde, enthält.

Die Bits 4 bis 7 des Strukturelements EV_CLASS enthalten die Ereignisklasse. Hier sind folgende Werte möglich:

- 1: Startereignisse von Standard-OBs
- 2: Startereignisse von Synchronfehler-OBs
- 3: Startereignisse von Asynchronfehler-OBs

Das Strukturelement PRIORITY liefert die zum aktuellen OB gehörige Prioritätsklasse.

Neben diesen beiden Elementen ist noch NUM von Bedeutung. NUM enthält die Nummer des aktuellen OBs bzw. des zuletzt gestarteten Anlauf-OBs.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
8081	Startinformation des aktuellen OBs passt nicht zum angegebenen Systemdatentyp
8083	Startinformation des zuletzt gestarteten Anlauf-OBs passt nicht zum angegebenen Systemdatentyp

Beispiel

Der zuletzt aufgerufene OB, der noch nicht vollständig abgearbeitet wurde, sei der OB 80, der zuletzt gestartete Anlauf-OB sei der OB 100.

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters TOP_SI von der Anweisung "RD_SINFO" und den zugehörigen lokalen Variablen des OB 80.

TOP_SI Strukturelement	Datentyp	OB 80 - Zugehörige lokale Variable	Datentyp
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters START_UP_SI von der Anweisung "RD_SINFO" und den zugehörigen lokalen Variablen des OB 100.

START_UP_SI Strukturelement	Datentyp	OB 100 - Lokale Variable	Datentyp
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

Beispiel

In dem folgenden Beispiel lesen Sie die Startinformation des zuletzt aufgerufenen Programmzyklus-OBs und des zuletzt gestarteten Anlauf-OBs der CPU aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

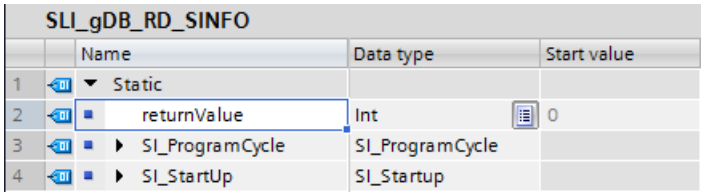

Rufen Sie die Anweisung "RD_SINFO" in einem OB1 auf. Die Parameter der Anweisung "RD_SINFO" verschalten Sie wie folgt.

Verwenden Sie für die CPU, für das Beispiel lediglich den einen OB.

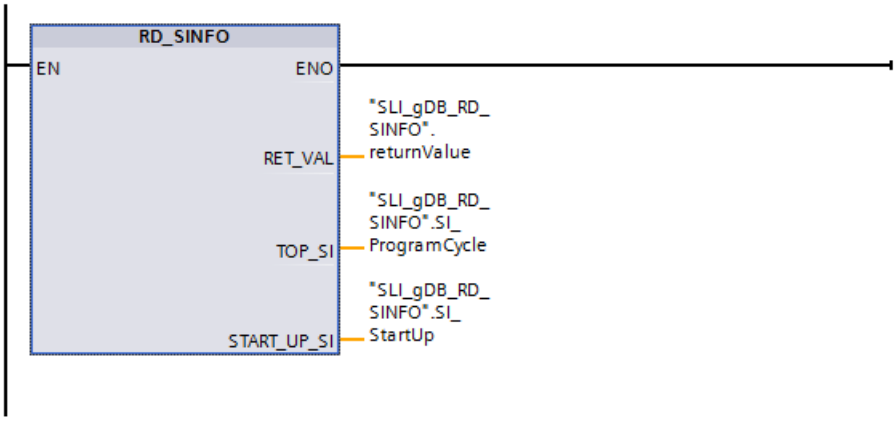

Wenn der Eingangsparameter EN den Signalzustand "TRUE" liefert, wird die Anweisung "RD_SINFO" abgearbeitet. Die Anweisung "RD_SINFO" liesst die Startinformation des zuletzt aufgerufenen Programmzyklus-OBs (OB1) auf und gibt diese am Ausgangsparameter TOP_SI ("SI_ProgrammCycle") aus. Am Ausgangsparameter START_UP_SI ("SI_Startup") wird die Startinformation des zuletzt gestarteten Anlauf-OBs ausgegeben.

Entsprechend dem Wert des Parameters "SI_Format" verwendet der OB1 optimierte Startinformationen. Zudem wird die Klasse des OBs ("OB_Class") und die vergebene Nummer des OBs ("OB_Nr") ausgelesen. Für den Anlauf-OB ("SI_Startup") wird keine Nummer ("OB_Nr" mit dem Wert "0") ausgelesen, da von der CPU kein Anlauf-OB verwendet wird. Zudem werden für den Anlauf-OB Standardwerte ("SI_Format", "OB_Class") angezeigt.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung für das Beispiel ohne Fehler verläuft (der Werte ist "0").

SLI_gDB_RD_SINFO

	Name	Data type	Start value	Monitor value
1	Static			
2	returnValue	Int	0	0
3	SI_ProgramCycle	SI_ProgramCycle		
4	SI_Format	USInt	0	254
5	OB_Class	USInt	1	1
6	OB_Nr	UInt	0	1
7	Initial_Call	Bool	false	FALSE
8	Remanence	Bool	false	TRUE
9	SI_StartUp	SI_Startup		
10	SI_Format	USInt	0	254
11	OB_Class	USInt	100	100
12	OB_Nr	UInt	0	0
13	LostRetentive	Bool	false	FALSE
14	LostRTC	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

RT_INFO: Runtime-Statistiken auslesen

Beschreibung

Mit der Anweisung "RT_INFO" generieren Sie Statistiken zur Laufzeit von bestimmten Organisationsbausteinen der Kommunikation oder dem Anwenderprogramm.

Welche Information ausgegeben wird, wählen Sie am Parameter MODE aus:

- MODE 1 bis 3 liefern die Daten zur Laufzeit eines bestimmten Organisationsbausteins zurück, dessen Nummer am Parameter OB angegeben wurde.
- MODE 10 und 11 messen über einen Zeitraum von 1 s und liefern den Mittelwert für den Anteil der höherprioren OBs (MODE 10) bzw. den Mittelwert für den Anteil der Kommunikation (MODE 11) an der Gesamtlaufzeit zurück.
- MODE 20 und 21 liefern für den letzten Programmzyklus zurück, welche Anteile der Laufzeit auf die Kommunikation und das Anwenderprogramm entfallen.
- MODE 23 bis 25 geben die kürzeste, längste und die aktuelle Zykluszeit des Anwenderprogramms aus.
- MODE 30 bis 32 liefern Daten zu konfigurierten Einstellungen des Anwenderprogramms zurück.
- MODE 50 bis 55 liefern Daten zum Systemzustand SYNCUP eines S7-1500-R/H-Systems

Beim Übergang der CPU von Anlauf in RUN werden alle Messungen erneut gestartet.

Die Laufzeit eines OBs steht erst zur Verfügung, wenn die erste Bearbeitung des OBs abgeschlosen ist; die Zykluszeit steht erst zur Verfügung, wenn der erste Zyklus abgeschlossen ist.
Wenn Sie RT_INFO aufrufen, bevor die erste Messung abgeschlossen ist, wird der Vorbelegungswert des zugehörigen Messwerts in INFO eingetragen und RET_VAL erhält den Wert 0. In der folgenden Tabelle sind die Vorbelegungswerte für MODE = 1, 2, 3, 23, 24 und 25 angegeben.

MODE	Vorbelegungswert
1	0
2	0
3	0x FFFF FFFF FFFF FFFF
23	0
24	0x FFFF FFFF FFFF FFFF
25	Parametrierte Mindestzykluszeit

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RT_INFO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	Über den Parameter MODE wählen Sie aus, welche Informationen Sie auslesen wollen (siehe Tabelle "Parameter MODE").
OB	Input	OB_ANY	E, A, M, D, L oder Konstante	Über den Parameter OB wählen Sie aus, zu welchem OB Sie Informationen auslesen möchten.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation (siehe "Parameter RET_VAL").
INFO	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Bereich, in den die gelesenen Daten geschrieben werden sollen. Der für INFO benötigte Datentyp ist abhängig von dem Parameter MODE (siehe Tabelle "Parameter MODE").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

Die folgende Tabelle zeigt, welche Informationen für die Werte am Parameter MODE zurückgeliefert werden.

MODE (dezimal)	Beschreibung	Hinweis	Wert am Parameter OB	Datentyp von INFO	Verfügbar ab CPU Version
1	Laufzeit eines bestimmten OBs	Unter der Laufzeit eines OBs versteht man hier diejenige Zeitdauer, in der die CPU die Befehle dieses OBs bearbeitet. Die Zeitdauer für die Bearbeitung von höher prioren OBs und von Kommunikationsaufgaben, die den OB unterbrechen, gehört also nicht dazu. Wenn Sie hingegen die gesamte Zeitdauer ermitteln wollen, die zwischen der Bearbeitung des ersten und des letzten Befehls des OBs vergeht - also einschließlich der Bearbeitung von höher prioren OBs und von Kommunikationsaufgaben, die den OB unterbrechen - verwenden Sie die Anweisung "RUNTIME".	OB-Nummer	LTIME	S7-1500 V1.5
2	Maximale Laufzeit eines bestimmten OBs		OB-Nummer	LTIME	S7-1500 V1.5
3	Minimale Laufzeit eines bestimmten OBs		OB-Nummer	LTIME	S7-1500 V1.5

MODE (dezimal)	Beschreibung	Hinweis	Wert am Parameter OB	Datentyp von INFO	Verfügbar ab CPU Version
10	Mittelwert des Anteils der höherprioren OBs an der Gesamtlaufzeit in Prozent Ausgegeben wird die Laufzeit von allen im Anwenderprogramm verwendeten OBs mit höherer Priorität als zyklische Programm-OBs (ProgramCycle). Hierunter fallen in der Regel alle OB-Typen, mit Ausnahme von Anlauf-OBs. Welche OBs verwendet werden können und welche Priorität diese jeweils haben ist CPU-abhängig und in dem Grundlagenkapitel zur Programmierung beschrieben.	Die Mittelwertbildung erfolgt über den Zeitraum von 1 s.	Nicht relevant	UINT	S7-1500 V1.5
11	Mittelwert des Anteils der Kommunikation an der Gesamtlaufzeit in Prozent Ausgegeben wird, welchen Anteil Kommunikationsprozesse an der Gesamtlaufzeit des Anwenderprogramms haben.	Die Mittelwertbildung erfolgt über den Zeitraum von 1 s. Der zurückgelieferte Kommunikationsanteil kann größer sein, als Sie durch den CPU-Parameter "Zyklusbelastung durch Kommunikation" vorgegeben haben. Dies kann dann der Fall sein, wenn Sie eine minimale Zykluszeit vorgegeben haben und diese vom Anwenderprogramm nicht vollständig benötigt wird. Dann kann die restliche zur Verfügung stehende Zeit nämlich erneut für Kommunikationsprozesse benutzt werden.	Nicht relevant	UINT	S7-1500 V1.5

MODE (dezimal)	Beschreibung	Hinweis	Wert am Parameter OB	Datentyp von INFO	Verfügbar ab CPU Version
20	Identisch zu MODE 10 mit dem Unterschied, dass die Auswertung auf dem zuletzt ausgeführten Programmzyklus basiert.	• Wenn die Zykluszeit zu hoch (> 1 Sekunde) ist, wird die Berechnung nicht durchgeführt. Die Anweisung gibt den Wert 65535 (0xFFFF) aus. • Ist die Zykluszeit zu niedrig (< 1 Millisekunde) kann die Berechnung nicht durchgeführt werden. In diesem Fall wird der letzte Zyklus ausgewertet, der mindestens die Dauer von einer Millisekunde hatte. Durch die Parametrierung einer Mindestzykluszeit in den CPU-Eigenschaften können Sie eine Unterschreitung der Zykluszeit von 1 ms verhindern. • Für MODE 21: Der zurückgelieferte Kommunikationsanteil kann größer sein, als Sie durch den CPU-Parameter "Zyklusbelastung durch Kommunikation" vorgegeben haben. Dies kann dann der Fall sein, wenn Sie eine minimale Zykluszeit vorgegeben haben und diese vom Anwenderprogramm nicht vollständig benötigt wird. Dann kann die restliche zur Verfügung stehende Zeit nämlich erneut für Kommunikationsprozesse benutzt werden.	Nicht relevant	UINT	S7-1500 V1.7
21	Identisch zu MODE 11 mit dem Unterschied, dass die Auswertung auf dem zuletzt ausgeführten Programmzyklus basiert.		Nicht relevant	UINT	S7-1500 V1.7
23	Längste Zykluszeit Dauer des längsten Zyklus seit dem letzten Übergang von STOP nach RUN.	Die Zeiten entsprechen den Werten "Zykluszeiten gemessen" im Dialog "Zykluszeit" im TIA Portal. Den Dialog öffnen Sie über Online & Diagnose > Diagnose > Zykluszeit.	Nicht relevant	LTIME	S7-1500 V1.7
24	Kürzeste Zykluszeit Dauer des kürzesten Zyklus seit dem letzten Übergang von STOP nach RUN.		Nicht relevant	LTIME	S7-1500 V1.7
25	Aktuelle / letzte Zykluszeit Dauer des letzten Zyklus.		Nicht relevant	LTIME	S7-1500 V1.7

MODE (dezimal)	Beschreibung	Hinweis	Wert am Parameter OB	Datentyp von INFO	Verfügbar ab CPU Version
30	Zyklusüberwachungszeit Maximal zulässige Zeitdauer des CPU-Programms. Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, geht die CPU in den Betriebszustand STOP oder ruft einen Zeitfehler-OB auf.	Die Zeiten entsprechen den Werten "Zykluszeit parametriert" im Dialog "Zykluszeit" im TIA Portal. Den Dialog öffnen Sie über Online & Diagnose > Diagnose > Zykluszeit.	Nicht relevant	LTIME	S7-1500 V1.5
31	Ausgabe der konfigurierten Mindestzykluszeit für das Anwenderprogramm. Wurde eine Mindestzykluszeit in den CPU-Eigenschaften parametriert, verzögert das Betriebssystem den Beginn eines neuen Zyklus so lange, bis die minimale Zykluszeit erreicht ist.		Nicht relevant	LTIME	S7-1500 V1.5
32	Ausgabe der konfigurierten maximalen Kommunikationslast in Prozent	Die Festlegung, welcher Anteil der Zyklusbelastung auf die Kommunikation entfallen darf, wird in den Eigenschaften der CPU unter "Kommunikationslast" festgelegt.	Nicht relevant	UINT	S7-1500 V1.5
50	Anzahl der im Systemzustand SYNCUP zu übertragenden statischen Daten	-	Nicht relevant	LINT	S7-1500 V2.6
51	Anzahl der im Systemzustand SYNCUP bereits übertragenen statischen Daten	-	Nicht relevant	LINT	S7-1500 V2.6
52	Anzahl der im Systemzustand SYNCUP zu übertragenden dynamischen Daten	-	Nicht relevant	LINT	S7-1500 V2.6
53	Anzahl der im SYNCUP bereits übertragenen dynamischen Daten	-	Nicht relevant	LINT	S7-1500 V2.6
54	Maximale Verzögerung zwischen Primary- und Backup-CPU, die die Backup-CPU aufholen kann	-	Nicht relevant	LTIME	S7-1500 V2.6
55	Aktuelle Verzögerung zwischen Primary- und Backup-CPU in der Aufholphase der Backup-CPU	-	Nicht relevant	LTIME	S7-1500 V2.6

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0	Kein Fehler
8080	Wert am Parameter MODE wird nicht unterstützt.
8081	Am Parameter OB ausgewählter Organisationsbaustein existiert nicht im Anwenderprogramm.
8092	Ausgewählter Parameter MODE wird von dieser CPU-Version nicht unterstützt.
80C3	Unzureichende Ressourcen. Versuchen Sie, die Anweisung zu einem späteren Zeitpunkt erneut aufzurufen.
8452	Datentyp am Parameter INFO falsch. Kontrollieren Sie, ob der richtige Datentyp in Abhängigkeit von dem Parameter MODE gewählt worden ist.

Beispiel

In dem folgenden Beispiel lesen Sie die Laufzeit eines Zyklus-Organisationsbausteins aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

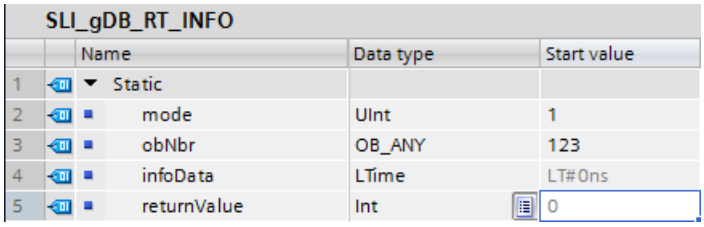

Die Parameter der Anweisung verschalten Sie wie folgt.

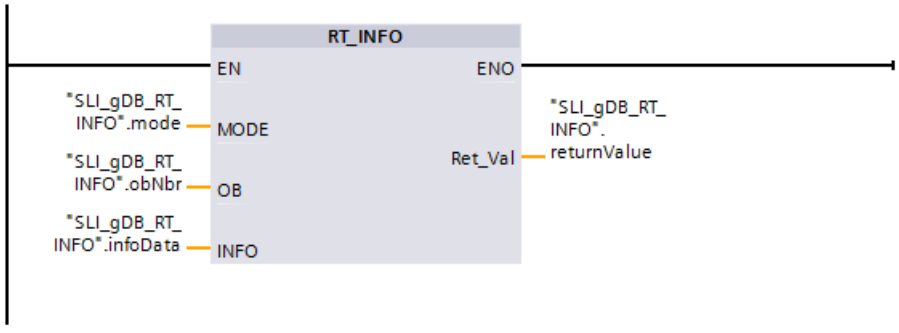

- Legen Sie einen neuen Baustein vom Typ "Program cycle" an. Die OB-Nummer geben Sie am Parameter OB an.
- Am Parameter MODE geben Sie "1" ein (Laufzeit eines bestimmten OBs auslesen).
- Am Parameter INFO geben Sie eine Variable mit dem Datentyp LTIME an (hier "infoData").
- Am Parameter RET_VAL geben Sie eine Variable mit dem Datentyp INT an, um die Fehlermeldungen der Anweisung auszugeben.

Nach dem Aufruf der Anweisung wird die aktuell gemessene Laufzeit in die Variable "infoData" geschrieben.

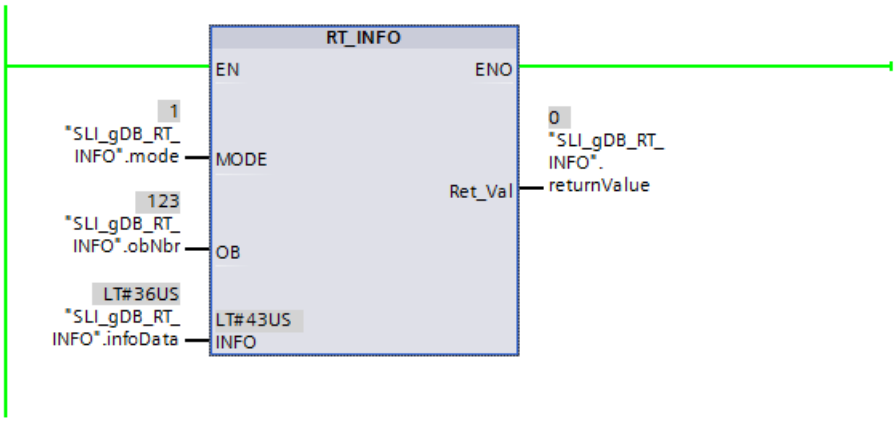

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

LED: LED-Status lesen

Beschreibung

Mit der Anweisung "LED" können Sie den Status (z. B. "Ein" oder "Aus") einer bestimmten Baugruppen-LED auslesen.

- Über den Parameter LADDR adressieren Sie die CPU oder das Interface.
- Über den Parameter LED wählen Sie die LED der Baugruppe aus, deren aktueller Status über die Anweisung ausgelesen wird.
- Der Parameter RET_VAL gibt bei Aufruf der Anweisung den Status der ausgewählten LED aus. Je nach ausgewählter LED können nur bestimmte Statusinformationen angezeigt werden. Beispielsweise haben bestimmte LEDs nur eine Farbe. Die Information, welcher Status für eine bestimmte LED möglich ist, entnehmen Sie der Hardware-Dokumentation der jeweiligen Baugruppe.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LED":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, L oder Konstante	HW-Kennung der CPU oder der Schnittstelle Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder der Schnittstelle in der Hardware-Konfiguration hinterlegt (*CPU-Name* + *~Common*).
LED	Input	UINT	E, A, M, D, L oder Konstante	Identifikationsnummer der LED: • 1: STOP/RUN • 2: ERROR • 3: MAINT (Wartung) • 4: Redundant • 5: Link (grün) • 6: Rx/Tx (gelb)
RET_VAL	Return	INT	E, A, M, D, L	Status der LED

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

RET_VAL	Beschreibung
0 bis 9	LED Status: • 0 = LED existiert nicht oder Statusinformation nicht verfügbar • 1 = Permanent ausgeschaltet • 2 = Farbe 1 (z. B. bei LED STOP/RUN: grün) permanent ein • 3 = Farbe 2 (z. B. bei LED STOP/RUN: orange) permanent ein • 4 = Farbe 1 blinkt mit 2 Hz • 5 = Farbe 2 blinkt mit 2 Hz • 6 = Farbe 1 und 2 blinken im Wechsel mit 2 Hz • 7 = LED ist aktiv, Farbe 1 • 8 = LED ist aktiv, Farbe 2 • 9 = LED existiert nicht oder Statusinformation nicht verfügbar Aus Kompatibilitätsgründen wird für diesen Wert von RET_VAL ENO auf FALSE gesetzt.
8086	Das R/H-System ist nicht im Systemzustand RUN-Redundant. Daher sind die Daten der Partner-CPU nicht verfügbar.
8091	Die über den Parameter LADDR adressierte Hardware-Komponente ist nicht vorhanden.
8092	Über den Parameter LADDR wurde eine Hardware-Komponente adressiert, die die gewünschte Information nicht liefert.
8093	Die am Parameter LED angegebene Identifikationsnummer ist nicht definiert.
80Bx	Die am Parameter LADDR angegebene CPU unterstützt die Anweisung "LED" nicht.

Beispiel

In dem folgenden Beispiel lesen Sie den Status der CPU-LED aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

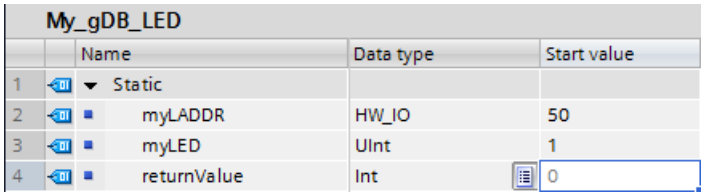

My_gDB_LED			
	Name	Data type	Start value
1	Static		
2	myLADDR	HW_IO	50
3	myLED	UInt	1
4	returnValue	Int	0

Die Parameter der Anweisung verschalten Sie wie folgt.

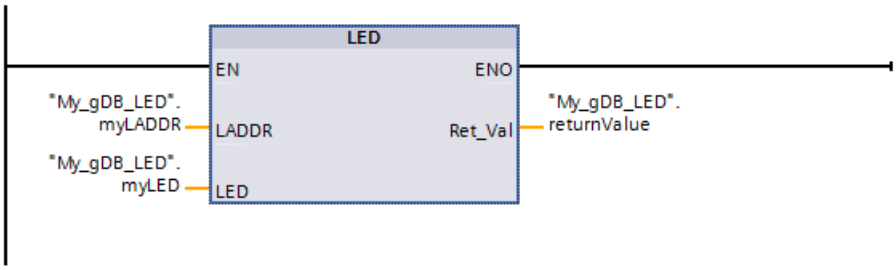

Durch den Parameter LADDR ("myLADDR") wird der Anweisung "LED" die HW-Kennung der CPU bekannt gegeben. Die zu beobachtende CPU-LED wird durch den Parameter LED ("myLED") bekannt gegeben. Der Status der CPU-LED (STOP/RUN) wird abgefragt. Wenn die CPU von STOP in den Betriebszustand RUN gesetzt wird, wird am Ausgangsparameter RET_VAL ("returnValue") der Wert "6" (grün und orange im Wechsel) angezeigt. Nachfolgend wird als Status der LED ("returnValue") der Wert "2" (permanent grün) angezeigt.

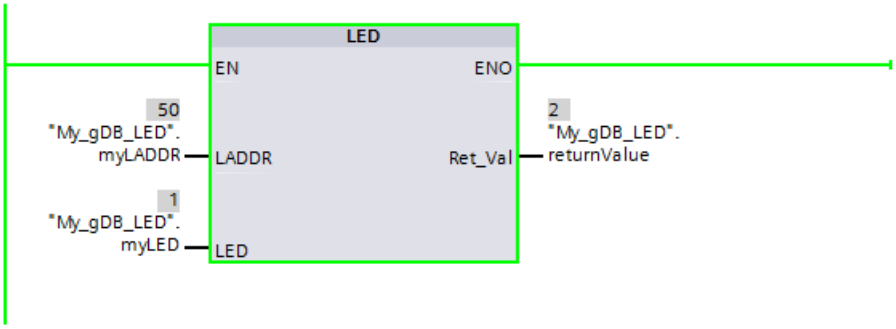

Hinweis: Die Bedeutung der Farben für die LED STOP/RUN ist wie folgt:

Farbe	Bedeutung
Rot	STOP
Grün	RUN
Grün und Orange im Wechsel	Die CPU lädt
Rot und Grün im Wechsel	Programmbearbeitung läuft

Weitere Informationen zur Bedeutung der LED-Farben finden Sie in der Hardwarebeschreibung der CPU.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Get_IM_Data: Identifikations- und Maintenancedaten lesen

Beschreibung

Die Anweisung "Get_IM_Data" liest die Identifikations & Maintenance Daten (I&M-Daten) eines Geräts aus. Über den Parameter LADDR wählen Sie über die Hardware-Kennung das Gerät aus, dessen I&M-Daten Sie auslesen möchten.

Über den Parameter IM_TYPE definieren Sie, welche Daten über die Anweisung ausgelesen werden:

- IM_TYPE = 0: I&M 0-Daten
  Die I&M 0-Daten sind die gerätespezifischen Basisinformationen eines Geräts und enthalten Informationen wie die Herstellerkennung, die Bestellnummer, die Seriennummer oder die Hardware- und Firmware-Version. Auf die I&M 0-Daten kann nur lesend zugegriffen werden. Die Informationen werden im TIA-Portal auch in der Ansicht "Online & Diagnose" eines Geräts angezeigt.
- IM_TYPE = 11: I&M 1-Daten aus den Parametrierungsdaten der CPU
  Die I&M 1-Daten enthalten eine Funktionsbeschreibung des Geräts, sowie das Ortskennzeichen. Die Information, wie das Gerät innerhalb der Anlage bezeichnet wurde.
- IM_TYPE = 12: I&M 2-Daten aus den Parametrierungsdaten der CPU
  Die I&M 2-Daten enthalten das Einbaudatum, d. h. die Information, wann das Gerät in der Anlage installiert wurde.
- IM_TYPE = 13: I&M 3-Daten aus den Parametrierungsdaten der CPU
  Die I&M 3-Daten enthalten Zusatzinformationen des installierten Geräts. Die Zusatzinformationen sind ein Freitext und werden beliebig vergeben.

Die gelesenen I&M-Daten werden in den am Parameter DATA adressierten Bereich geschrieben.

Über die Ausgangsparameter BUSY, DONE, ERROR und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Bearbeitungszustand des Leseauftrags angezeigt.

Definition: Identifikations- und Maintenance-Daten (I&M)

Identifikations- und Maintenance-Daten (I&M) sind in einer Baugruppe gespeicherte Informationen, welche das Überprüfen der Anlagenkonfiguration, das Auffinden von Hardware-Änderungen einer Anlage oder das Beheben von Fehlern erleichtern.

- Identifikationsdaten (I-Daten) sind gleich bleibende Informationen zu einem Gerät, auf die nur lesend zugegriffen werden kann.
- Maintenance-Daten (M-Daten) sind anlagenabhängige Informationen, wie z. B. Einbauort oder Einbaudatum. Maintenance-Daten werden während der Projektierung erstellt und auf die Baugruppe geschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Get_IM_Data":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung des Geräts. Die Nummer wird automatisch vergeben und ist in den Eigenschaften des Geräts in der Hardware-Konfiguration hinterlegt.
IM_TYPE	Input	UINT	E, A, M, D, L oder Konstante	Identifikations- und Maintenance-Datennummer Mögliche Werte: • 0: I&M 0-Daten • 11: I&M 1-Daten aus den Parametrierungsdaten der CPU • 12: I&M 2-Daten aus den Parametrierungsdaten der CPU • 13: I&M 3-Daten aus den Parametrierungsdaten der CPU
DATA	InOut	VARIANT	E, A, M, D, L	Bereich zur Ablage der gelesenen Identifikations- und Maintenance-Daten (siehe unten).
DONE	Output	BOOL	E, A, M, D, L	Die Anweisung wurde erfolgreich ausgeführt. Die I&M-Daten wurden an den Parameter DATA übertragen.
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen oder noch nicht gestartet. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige kopieren Sie deshalb den Wert von STATUS in einen freien Datenbereich.

Weitere Informationen zu den Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter DATA für I&M 0-Daten

Zur Ablage der I&M 0-Daten verwenden Sie ein Array (ARRAY of BYTE) oder eine spezielle Datenstruktur:

- Wenn Sie am Parameter DATA ein Array (ARRAY of BYTE) adressieren, werden die gelesenen I&M 0-Daten als Byte-Folge nach DATA kopiert. Ist das adressierte Array länger als die gelesenen Daten, werden die nicht benötigten Bytes mit Nullen aufgefüllt.
- Für die I&M 0-Daten kann am Parameter DATA auch die folgende Struktur "IM0_Data" verwendet werden:

Parameter		Datentyp	Bytes	Beschreibung
Manufacturer_ID		UINT	2	Herstellerkennung (z. B. "42" für SIEMENS)
Order_ID		CHAR[20]	20	Bestellnummer
Serial_Number		CHAR[16]	16	Seriennummer
Hardware_Revision		UINT	2	Hardware Ausgabestand
Software_Revision		STRUCT	4	Firmware Ausgabestand
	Type	CHAR	1	-
	Functional	USInt	1	-
	Bugfix	USInt	1	-
	Internal	USInt	1	-
Revision_Counter		UINT	2	Revisionszähler
Profile_ID		UINT	2	Profil
Profile_Specific_Type		UINT	2	Geräteklasse
IM_Version		WORD	2	I&M-Version
I&M_Supported		WORD	2	Geräteseitig unterstützte I&M-Daten (I&M 0-I&M 4)

Wird am Parameter DATA ein anderer Datentyp verwendet, wird am Parameter STATUS der Fehlercode 8093 ausgegeben.

Parameter DATA für I&M 1-, I&M 2- und I&M 3-Daten

Zur Ablage der I&M-Daten verwenden Sie eine Zeichenkette (STRING), ein Array (ARRAY of CHAR/BYTE) oder eine Datenstruktur (STRUCT):

- Wenn Sie am Parameter DATA eine Zeichenkette (Datentyp STRING) adressieren, passt sich die Länge der Zeichenkette automatisch der Länge der gelesenen I&M-Daten an (maximal 254 Zeichen).
- Wenn Sie am Parameter DATA eine Datenstruktur (ARRAY of CHAR/BYTE oder STRUCT) adressieren, werden die gelesenen I&M-Daten in die einzelnen Komponenten des verwendeten Datentyps geschrieben. Wenn die adressierte Datenstruktur länger ist als die gelesenen Daten, werden die übrigen Komponenten mit Nullen aufgefüllt.
- Wenn Sie am Parameter DATA eine Datenstruktur vom Datentyp STRUCT erstellen, verwenden Sie einen Datenbaustein ohne optimierten Bausteinzugriff (siehe Kategorie "Attribute" der Bausteineigenschaften).

Wenn am Parameter DATA ein anderer Datentyp als STRING, ARRAY of BYTE/CHAR oder STRUCT verwendet wird, wird am Parameter STATUS der Fehlercode 8093 ausgegeben.

Hinweis

Weitere Informationen zu I&M Daten

Weitere Informationen zu I&M Daten finden Sie unter anderem auf den Internetseiten der PROFIBUS & PROFINET International (Link: http://www.profibus.com (http://www.profibus.com)).

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Kein Fehler
7001	Erster Aufruf der asynchronen Anweisung "Get_IM_Data". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
7002	Weiterer Aufruf der asynchronen Anweisung "Get_IM_Data". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
8086	Das R/H-System ist nicht im Systemzustand RUN-Redundant. Daher sind die Daten der Partner-CPU nicht verfügbar.
8091	Das am Parameter LADDR adressierte Gerät ist nicht vorhanden.
8092	LADDR adressiert ein Gerät, das die Ausgabe von I&M-Daten nicht unterstützt.
8093	Datentyp am Parameter DATA wird nicht unterstützt.
80A1	LADDR adressiert ein Gerät, das die Ausgabe von I&M 1-, I&M 2- oder I&M 3-Daten nicht unterstützt.
80B1	Keine gültigen Daten vorhanden. (Die I&M-Daten werden entweder nicht vom (Sub)Modul unterstützt oder Sie haben für dieses Gerät keine I&M-Daten konfiguriert.)
80B2	Ungültiger Wert am Parameter IM_TYPE oder der gewählte IM_TYPE wird von der CPU oder dem adressierten Gerät nicht unterstützt.
80C3	Maximum von 10 parallelen Ausführungen der Instanzen überschritten: • Vermeiden Sie mehr als 10 parallele Aufrufe. • Wiederholen Sie den Aufruf zu einem späteren Zeitpunkt, da es sich um einen temporären Fehler handelt.
8752	Um alle I&M-Daten abzulegen, ist der am Parameter DATA angegebene Speicherbereich ist zu klein. Die gelesenen I&M-Daten werden nur bis zur maximalen Länge des angegebenen Speicherbereichs abgelegt.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie die IM0-Daten einer S7-1500 CPU aus. Die IM0-Daten sind die Basisinformationen eines Geräts und enthalten Informationen wie die Herstellerkennung, die Bestellnummer, die Seriennummer oder die Hardware- und Firmware-Version.

Zur Ablage der IM0-Daten legen Sie in einem globalen Datenbaustein vier Variablen und eine Struktur mit dem Datentyp IM0_Data an. Den Namen für die Struktur (hier "MyPLC_IM_Data") können Sie beliebig vergeben.

SLI_gDB_Get_IM_Data

		Name	Data type	Start value
1		Static		
2		imData	IM0_Data	
3		done	Bool	false
4		busy	Bool	false
5		error	Bool	false
6		status	Word	16#0

Am Parameter LADDR geben Sie die Hardware-Kennung der CPU ein. Über die Hardware-Kennung wird das Gerät eindeutig identifiziert. Um die Hardware-Kennung zu Ihrer CPU zu ermitteln, öffnen Sie die PLC-Variablentabelle und das Register Systemkonstanten. Dann suchen Sie in der Spalte Name nach der CPU. Der dazugehörige Wert ist die Hardware-Kennung, die Sie am Parameter LADDR eintragen

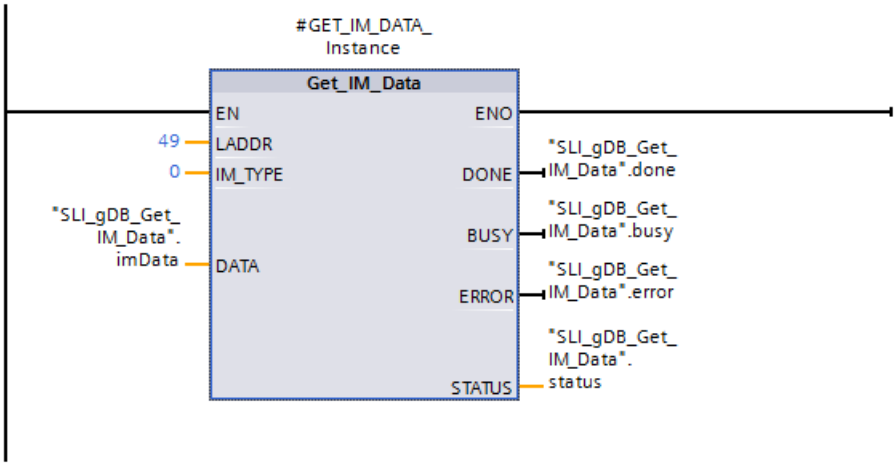

Nach erfolgreicher Ausführung der Anweisung werden die IM0-Daten in den Datenbaustein geschrieben.

SLI_gDB_Get_IM_Data

	Name	Data type	Start value	Monitor value
1	Static			
2	imData	IM0_Data		
3	Manufacturer_ID	UInt	0	42
4	Order_ID	String[20]	''	'6ES7 511-1AK00-0...
5	Serial_Number	String[16]	''	'S C-DOS710132013'
6	Hardware_Revision	UInt	0	3
7	Software_Revision	IM0_Version		
8	Type	Char	' '	'V'
9	Functional	USInt	0	1
10	Bugfix	USInt	0	5
11	Internal	USInt	0	0
12	Revision_Counter	UInt	0	0
13	Profile_ID	UInt	0	0
14	Profile_Specific_Ty...	UInt	0	0
15	IM_Version	Word	16#0	16#0101
16	IM_Supported	Word	16#0	16#001E
17	done	Bool	false	TRUE
18	busy	Bool	false	FALSE
19	error	Bool	false	FALSE
20	status	Word	16#0	16#0000

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

GET_NAME: Name eines IO-Device oder eines DP-Slaves auslesen

Beschreibung

Die Anweisung "GET_NAME" liest den Namen eines IO-Device oder eines DP-Slaves aus. Der Name wird in der Netzsicht und in den Eigenschaften des IO-Device bzw. des DP-Slaves angezeigt. Der folgende Screenshot zeigt dies am Bespiel eines IO-Device.

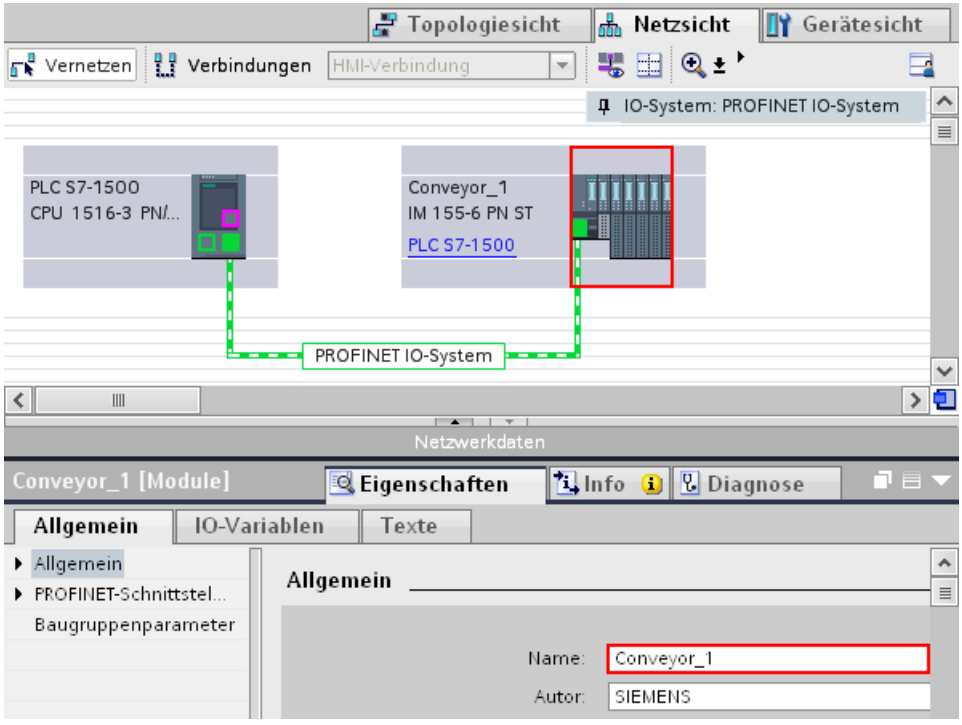

Die Auswahl des IO-Device bzw. des DP-Slaves erfolgt über die Hardware-Kennung des PROFINET IO- bzw. des DP-Master-Systems (am Parameter LADDR) und der Gerätenummer des IO-Device bzw. der PROFIBUS-Adresse des DP-Slaves (Parameter STATION_NR).

Nach erfolgreicher Ausführung der Anweisung wird der Name des IO-Device bzw. des DP-Slaves in den über den Parameter DATA adressierten Bereich geschrieben.

Der gelesene Name ist abhängig von dem Typ des Peripheriegeräts:

- Bei einem DP-Slave oder einem IO-Device wird der Name der Kopfbaugruppe ausgegeben.
- Bei einem I-Slave oder I-Device wird der Name des Interfacemoduls ausgegeben.
- Bei einem HMI-Panel wird der Name der Schnittstelle ausgegeben.
- Bei einer PC-Station wird der Name des Interfacemoduls ausgegeben.
- Bei GSD-Geräten wird der Name des Device Access Point (DAP) angezeigt (Name des Interface oder Kopfbaugruppe).

Die Länge des Namens wird am Parameter LEN ausgegeben. Wenn der Name länger ist als der am Parameter DATA angegebene Bereich, wird dieser nur bis zur maximalen Länge des adressierten Bereichs geschrieben.

Die maximale Länge für einen Namen beträgt 128 Zeichen.

Hinweis

Name der CPU auslesen (ab Version 1.1)

Wenn Sie an den Parametern LADDR und STATION_NR jeweils eine "0" übergeben, gibt die Anweisung den Namen der CPU aus.

Beim R/H-System im Betriebszustand RUN-Redundant liefert die Anweisung den Namen der Primary-CPU.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET_NAME":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IOSYSTEM	E, A, M, D, L oder Konstante	Hardware-Kennung des PROFINET-IO- oder DP-Master-Systems. Die Nummer wird aus den Systemkonstanten oder den Eigenschaften des PROFINET-IO- oder DP-Master-Systems übernommen.
STATION_NR	Input	UINT	E, A, M, D, L oder Konstante	Gerätenummer des PROFINET IO-Device bzw. PROFIBUS-Adresse des DP-Slaves. Diese wird Ihnen in der Netzsicht in den Eigenschaften des IO-Device bzw. des DP-Slaves angezeigt.
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Bereich, in den der Name geschrieben wird.
DONE	Output	BOOL	E, A, M, D, L	Die Anweisung wurde erfolgreich ausgeführt. Der Name des IO-Device bzw. des DP-Slaves wurde an den Bereich am Parameter DATA übertragen.
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
LEN	Output	DINT	E, A, M, D, L	Länge des Namens des IO-Device bzw. des DP-Slaves (Anzahl der Zeichen).
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status kopieren Sie deshalb den STATUS in einen freien Datenbereich.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
7001	Erster Aufruf der asynchronen Anweisung "GET_NAME". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
7002	Weiterer Aufruf der asynchronen Anweisung "GET_NAME". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
8090	Die am Parameter LADDR angegebene Hardware-Kennung existiert in dem Projekt nicht.
8092	Der Wert am Parameter LADDR adressiert kein PROFINET-IO-System bzw. kein DP-Master-System.
8093	Datentyp am Parameter DATA wird durch die Anweisung nicht unterstützt.
8095	• Gerätenummer (Parameter STATION_NR) existiert nicht in dem ausgewählten PROFINET-IO-System oder adressiert kein IO-Device. • PROFIBUS-Adresse (Parameter STATION_NR) existiert nicht in dem ausgewählten DP-Master-System.
80B1	Anweisung wird durch die verwendete CPU nicht unterstützt.
80C3	Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an gleichzeitigen Bausteinaufrufen. Vor der Beendigung von mindestens einem Bausteinaufruf wird "GET_NAME" nicht ausgeführt.
8852	Der an dem Parameter DATA angegebene Bereich ist zu kurz für den vollständigen Namen des IO-Device bzw. des DP-Slaves. Der Name wird bis zur maximal möglichen Länge geschrieben. Um den vollständigen Namen auszulesen, verwenden Sie am Parameter DATA einen längeren Datenbereich. Der Bereich muss hierzu mindestens die Anzahl der Zeichen am Parameter LEN umfassen.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt, wie Sie den Stationsnamen eines HMI-Panels auslesen.

Konfiguration des HMI-Panels:

- Das HMI-Panel mit dem Stationsnamen "HMI_IO-Device" ist in der Netzsicht erstellt und dem gleichen PROFINET-IO System zugeordnet wie die CPU.
- Für das HMI-Panel wurde in den Eigenschaften der Hardware-Konfiguration die Betriebsart "IO-Device" aktiviert und als IO-Controller die CPU zugewiesen.
- In den Eigenschaften wurde unter "Ethernet-Adressen" die Gerätenummer "20" vergeben.

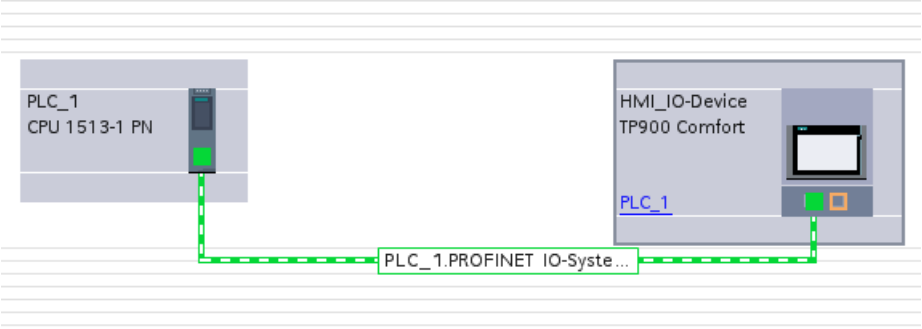

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_GET_NAME

	Name	Data type	Start value
1	Static		
2	moduleName	String	''
3	done	Bool	false
4	busy	Bool	false
5	error	Bool	false
6	lengthOfData	DInt	0
7	status	Word	16#0

Parametrierung der Anweisung "GET_NAME":

- Am Eingangsparameter LADDR geben Sie die Hardware-Kennung des IO-Systems (262) an.
- Am Eingangsparameter STATION_NR geben Sie die Gerätenummer des HMI-Panels (2) an.
- Die restlichen Parameter der Anweisung "GET_NAME" verschalten Sie wie folgt.

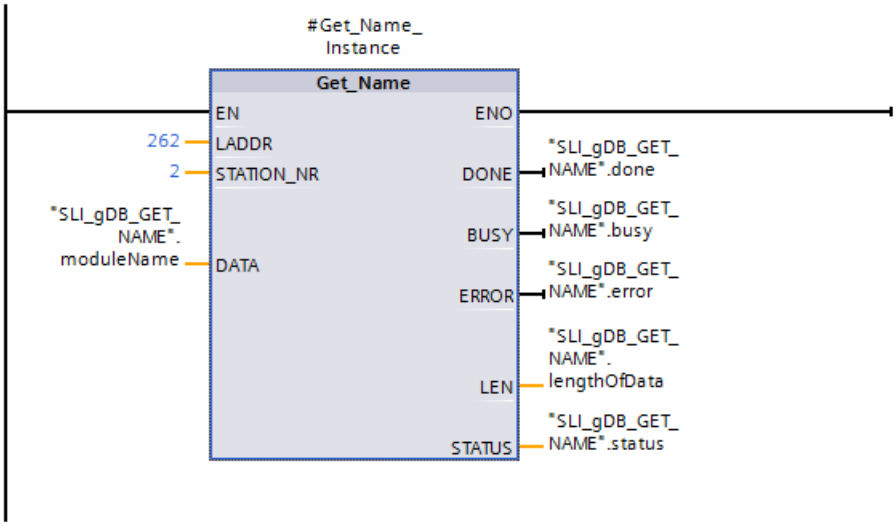

Ausführung der Anweisung:

- Nach der Ausführung der Anweisung wird der Name der Station des HMI-Panels (HMI_IO-Device) am Parameter DATA ("moduleName") gespeichert.
- Am Parameter LEN wird die Anzahl der Zeichen des Namens ausgegeben (13).
- Da GET_NAME nach dem erfolgreichen Lesen des IO-Device nicht gestoppt wurde, zeigt der Ausgangsparameter BUSY ("busy") den Wert "TRUE" und der Ausgangsparameter STATUS ("status") den Wert "16#7002".

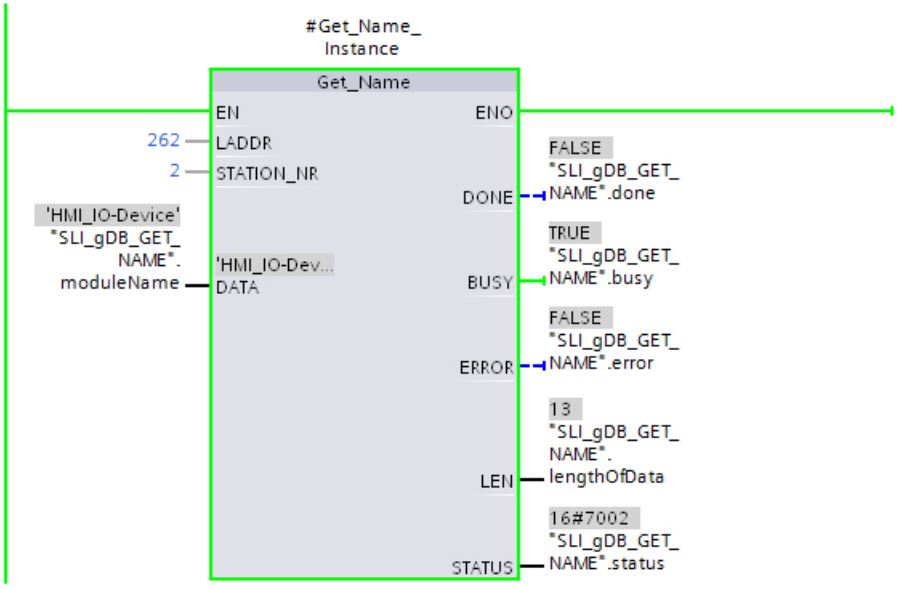

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetStationInfo: Informationen eines IO-Device auslesen

Beschreibung

Mit der Anweisung "GetStationInfo" lesen Sie die Informationen eines PROFINET IO-Device aus. Mit der Anweisung können auch die Informationen eines IO-Device gelesen werden, welches sich in einem unterlagerten IO-System befindet (Anbindung über CP/CM).

Das IO-Device wird über die Hardware-Kennung der Station am Parameter LADDR adressiert. Die Hardware-Kennung wird in der Ansicht "Geräte & Netze" in den Eigenschaften der Station angezeigt.

Über den Parameter MODE wählen Sie aus, welche Informationen gelesen werden sollen.

Am Parameter DATA geben Sie den Datenbereich an, in den die gelesenen Adressdaten geschrieben werden. Zur Ablage der IP-Adresse verwenden Sie die Struktur "IF_CONF_v4", zur Ablage der MAC-Adresse die Struktur "IF_CONF_MAC".

Über den Steuerparameter REQ aktivieren Sie das Auslesen der Adressdaten. Hierzu muss das IO-Device erreichbar sein.

Über die Ausgangsparameter BUSY, DONE, ERROR und den Ausgangsparameter STATUS wird der Bearbeitungszustand des Leseauftrags angezeigt.

Hinweis

Adressieren Sie das IO-Device ausschließlich über die Hardware-Kennung der Station

Die Station, das IO-Device und die PROFINET-Schnittstelle haben jeweils eine eigene Hardware-Kennung. Verwenden Sie für die Anweisung "GetStationInfo" ausschließlich die Hardware-Kennung der Station.

Wenn über den Parameter LADDR z. B. eine PROFINET-Schnittstelle adressiert wird, werden die Adressdaten nicht gelesen und der Fehlercode 8092 ausgegeben.

Um die Adressdaten einer integrierten PROFINET-Schnittstelle oder eines CM/CP im zentralen Aufbau auszulesen, verwenden Sie die Anweisung "RDREC".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetStationInfo":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter Request Aktiviert das Auslesen der Informationen bei REQ = "1".
LADDR	Input	HW_DEVICE	E, A, M, D, L oder Konstante	Hardware-Kennung der Station des IO-Device Die Nummer wird in der Netzsicht aus den Eigenschaften der Station oder dem Register "Systemkonstanten" der Standard-Variablentabelle übernommen.
DETAIL	Input	HW_SUB-MODULE	E, A, M, D, L oder Konstante	Der Parameter DETAIL wird nicht verwendet. Belassen Sie den Parameter unverschaltet.
MODE	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der zu lesenden Adressdaten: • MODE = 1: Adressparameter nach IPv4 (S7-1500-CPUs ab Firmware-Stand V1.1) • MODE = 2: MAC-Adresse (S7-1500-CPUs ab Firmware-Stand V1.5)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	InOut	VARIANT	D, L	Zeiger auf den Bereich, in den die Adressdaten des IO-Device geschrieben werden. Verwenden Sie bei MODE = 1 die Struktur "IF_CONF_v4" und bei MODE = 2 die Struktur "IF_CONF_MAC".
DONE	Output	BOOL	E, A, M, D, L	Die Anweisung wurde erfolgreich ausgeführt. Die Adressdaten wurden an den Parameter DATA übertragen.
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter DATA

- Verwenden Sie am Parameter DATA die Struktur "IF_CONF_v4" zur Ablage der Adressparameter nach IPv4.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	Id	UINT	30	ID der Struktur "IF_CONF_v4".
2 … 3	Length	UNIT	18	Länge der gelesenen Daten in BYTE.
4 ... 5	Mode	UNIT	0	Für die Anweisung "GetStationInfo" nicht relevant (wird auf "0" belassen).
6 ... 9	InterfaceAddress	ARRAY [1..4] of BYTE	-	IP-Adresse des IO-Device im Format IP_V4, z. B. für 192.168.3.10: • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 10

Byte	Parameter	Datentyp	Startwert	Beschreibung
10 … 13	SubnetMask	ARRAY [1..4] of BYTE	-	Subnetzmaske des IO-Device im Format IP_V4, z. B. für 255.255.255.0: • addr[1] = 255 • addr[2] = 255 • addr[3] = 255 • addr[4] = 0
14 ... 17	DefaultRouter	ARRAY [1..4] of BYTE	-	IP-Adresse des Routers im Format IP_V4, z. B. für 192.168.3.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 1

- Verwenden Sie am Parameter DATA die Struktur "IF_CONF_MAC" zur Ablage der MAC-Adresse.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	Id	UINT	3	ID der Struktur "IF_CONF_MAC".
2 … 3	Length	UNIT	12	Länge der gelesenen Daten in BYTE.
4 ... 5	Mode	UNIT	0	Für die Anweisung "GetStationInfo" nicht relevant (wird auf "0" belassen).
6 ... 11	MACAddress	ARRAY [1..6] of BYTE	-	MAC-Adresse des IO-Device, z. B. für 08-00-06-12-34-56 • Mac[1] = 8 • Mac[2] = 0 • Mac[3] = 6 • Mac[4] = 12 • Mac[5] = 34 • Mac[6] = 56

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
7000	Kein Auftrag in Bearbeitung.
7001	Erster Aufruf der asynchronen Anweisung "GetStationInfo". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
7002	Weiterer Aufruf der asynchronen Anweisung "GetStationInfo". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).
8080	Wert am Parameter MODE wird nicht unterstützt.
8090	Die am Parameter LADDR angegebene Hardware-Kennung ist nicht konfiguriert.
8092	Der Parameter LADDR adressiert kein PROFINET IO-Device.
8093	Ungültiger Datentyp am Parameter DATA.

Fehlercode* (W#16#...)	Erläuterung
80A0	Angeforderte Information wird nicht gelesen.
80C0	Adressiertes IO-Device nicht erreichbar.
80C3	Die maximale Anzahl simultaner Aufrufe der Anweisung "GetStationInfo" (10 Instanzen) ist zur Zeit erreicht.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im Folgenden lesen Sie mit der Anweisung „GetStationInfo“ die IP-Adressdaten eines IO-Device aus und schreiben die Informationen in einen Datenbaustein. Die IP-Adressdaten umfassen die IP-Adresse, die Subnetzmaske und (falls verwendet) die Adressdaten des Routers.

Die Anweisung wird auf dem IO-Controller ausgeführt und liest die IP-Adressdaten eines unterlagerten IO-Device (hier als Beispiel eine ET200MP).

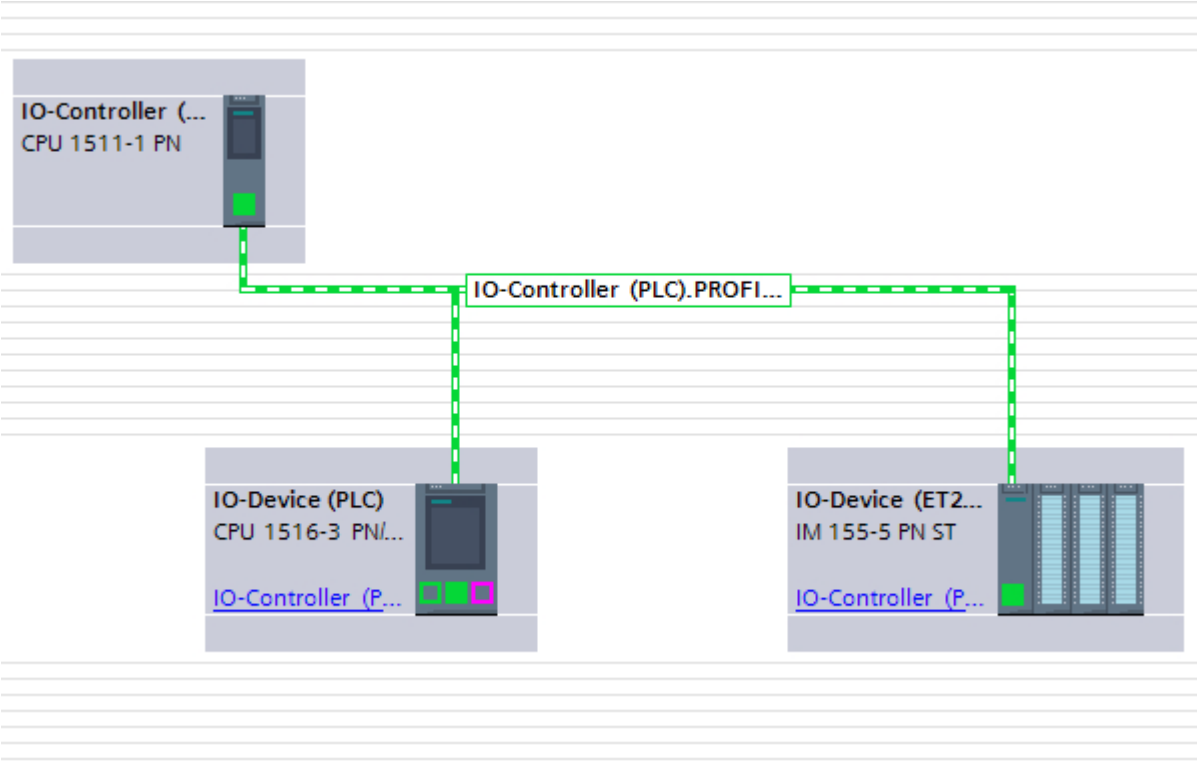

Zur Ablage der Adressdaten legen Sie in einem globalen Datenbaustein fünf Variablen und eine Struktur mit dem Datentyp IF_CONF_v4 an. Vergeben Sie den Namen für die Struktur (hier „IP_Address“) beliebig.

SLI_gDB_GetStationInfo			
	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	IP_address	IF_CONF_v4	
4	done	Bool	false
5	busy	Bool	false
6	error	Bool	false
7	status	Word	16#0

Danach rufen Sie die Anweisung "GetStationInfo" auf:

- Die Struktur IF_CONF_v4 verwenden Sie am Parameter DATA.
- Am Parameter LADDR geben Sie die Hardware-Kennung des IO-Device ein. Über die Hardware-Kennung wird das Gerät eindeutig identifiziert. Um die Die Hardware-Kennung zu Ihrem IO-Device zu ermitteln, öffnen Sie die PLC-Variablentabelle und das Register Systemkonstanten. Dann suchen Sie in der Spalte Name nach dem Gerät und in der Spalte Datentyp nach „Hw_Device“. Der dazugehörige Wert ist die Hardware-Kennung, die Sie am Parameter LADDR eintragen
- Für den Parameter MODE wählen Sie „1“ (Auslesen der Adressparameter nach IPv4).

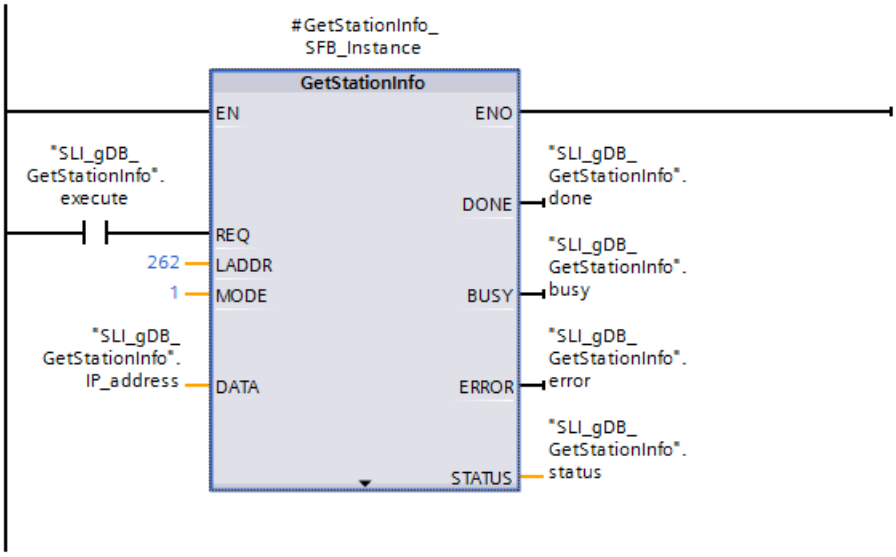

Mit REQ= 1 (TRUE) starten Sie das Auslesen der Adressdaten. Nach erfolgreicher Ausführung werden die IP-Adressdaten in den Datenbaustein geschrieben.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

GetChecksum: Prüfsumme auslesen

Beschreibung

Die Anweisung "GetChecksum" liest die Prüfsumme einer Gruppe von Objekten aus. Die Auswahl der Objekte erfolgt über den Parameter "Scope".

Hinweis

Bildung der Prüfsumme bei Bausteinen

Bei der Bildung der Prüfsumme gibt es Unterschiede zwischen Standard- und Safety-Bausteinen. Genaueres hierzu finden Sie in der Hilfe zum Optionspaket STEP 7 Safety.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetChecksum":

Parameter	Deklaration	Datentyp	Beschreibung
Scope	Input	UINT	Über den Parameter "Scope" wählen Sie aus, zu welchen Objekten Sie die Prüfsumme auslesen wollen: • 1: Standard-Bausteine (d. h. keine Safety-Bausteine) • 2: Safety-Bausteine • 3: Textlisten
Range (versteckt)	Input	UINT	"Range" muss den Wert 0 haben.
Subrange (versteckt)	Input	UINT	"Subrange" muss den Wert 0 haben.
Done	Output	BOOL	Die Anweisung wurde erfolgreich ausgeführt. Die Prüfsumme wurde an den Parameter "Checksum" übertragen.
Busy	Output	BOOL	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
Error	Output	BOOL	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter "Status" ausgegeben.

Parameter	Deklaration	Datentyp	Beschreibung
Status	Output	WORD	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.
Checksum	InOut	VARIANT	Prüfsumme Der Parameter "Checksum" muss auf eine Variable vom Datentyp ARRAY[0..7] of BYTE zeigen; unbenutzte Bytes werden auf 0 gesetzt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter "Status"

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
7001	Erstaufruf: Bildung der Prüfsumme angestoßen
7002	Zwischenaufruf: Bildung der Prüfsumme bereits aktiv
8080	Ungültiger Wert des Parameters "Scope"
8081	Der Parameter "Range" wird in dieser Firmware-Version nicht unterstützt und muss den Wert 0 haben.
8082	Der Parameter "Subrange" wird in dieser Firmware-Version nicht unterstützt und muss den Wert 0 haben.
8091	Die Anweisung "GetChecksum" wird von der CPU nicht unterstützt, oder die adressierten Objekte sind nicht verfügbar, z. B. weil sich kein fehlersicheres Programm auf der CPU befindet.
8852	Die Variable, auf die der Parameter „Checksum“ zeigt, hat einen falschen Datentyp.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie die Prüfsumme der Standard-Bausteine einer CPU S7-1500 aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_GetChecksum

	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	scope	UInt	1
4	checksum	Array[0..7] of Byte	
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	memErrStatus	Word	16#0

Sie legen einen FB an. In dem FB legen Sie die folgenden lokalen Variablen an.

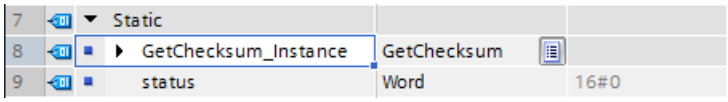

Netzwerk 1: Die Parameter der Anweisung "GetChecksum" verschalten Sie wie folgt.

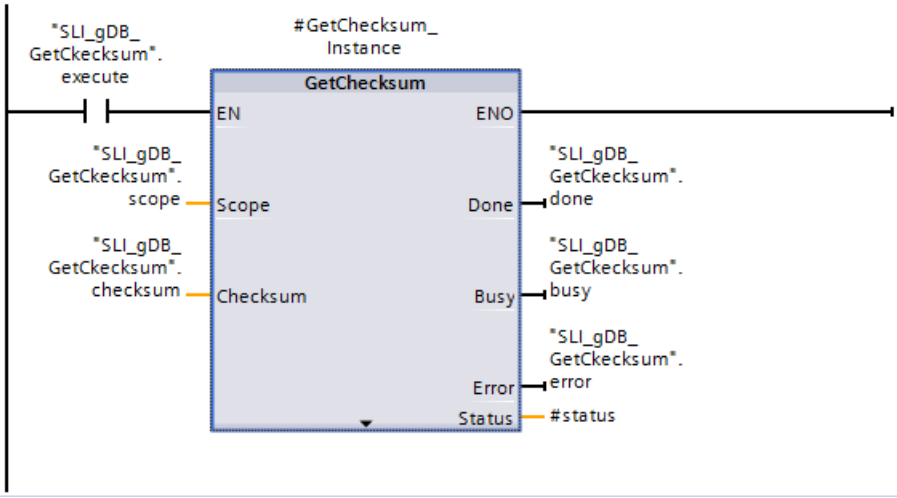

Netzwerk 2: Den Status im Fehlerfall speichern Sie wie folgt.

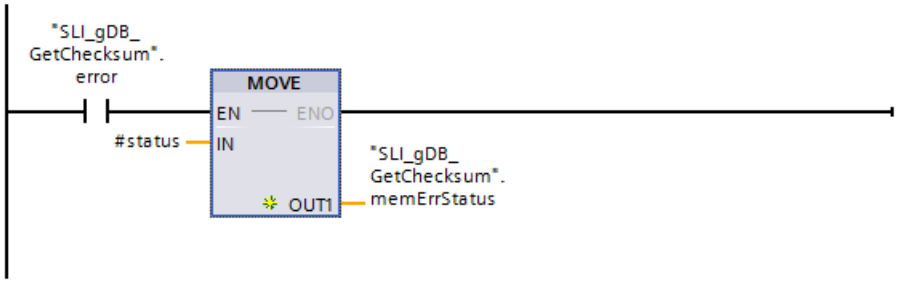

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "GetChecksum" ausgeführt. Entsprechend dem Wert "1" des Eingangsparameters SCOPE ("scope") wird die Prüfsumme der Standard-Bausteine ausgelesen. Am Parameter CHECKSUM ("checksum") wird die Prüfsumme ausgegeben. Der Erfolgstatus von GetChecksum wird am Ausgangsparameter DONE ("done") angezeigt.

An den Ausgangsparametern STATUS ("#status" und "memErrStatus") und ERROR ("error") wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verlief.

SLI_gDB_GetCkecksum

	Name	Data type	Start value	Monitor value
1	Static			
2	execute	Bool	false	TRUE
3	scope	UInt	1	1
4	checksum	Array[0..7] of Byte		
5	checksum[0]	Byte	16#0	16#F3
6	checksum[1]	Byte	16#0	16#7A
7	checksum[2]	Byte	16#0	16#5F
8	checksum[3]	Byte	16#0	16#9F
9	checksum[4]	Byte	16#0	16#F3
10	checksum[5]	Byte	16#0	16#35
11	checksum[6]	Byte	16#0	16#22
12	checksum[7]	Byte	16#0	16#CC
13	busy	Bool	false	FALSE
14	done	Bool	false	TRUE
15	error	Bool	false	FALSE
16	memErrStatus	Word	16#0	16#0000

Die Angabe der Prüfsumme finden Sie ebenso in den CPU-Eigenschaften.

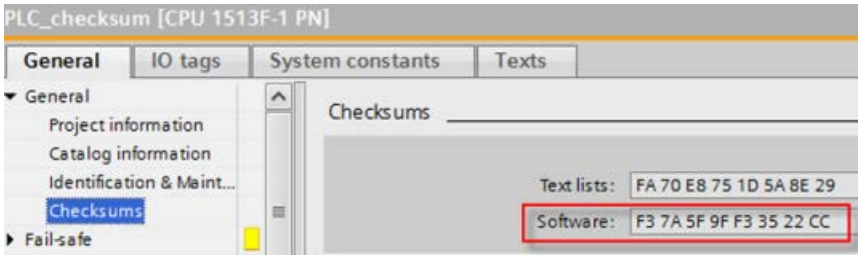

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Vergleich von PLC-Programmen anhand von Prüfsummen (Seite 7921)

GetSMCinfo: Informationen über die SIMATIC Memory Card auslesen

Beschreibung

Mit der Anweisung „GetSMCinfo“ lesen Sie Informationen über die gesteckte SIMATIC Memory Card aus. Die Auswahl der auszulesenden Information erfolgt über den Parameter „Mode“:

- Mode = 0, 1, 2, 3: Zulässig bei Standard-CPUs
- Mode = 10, ... 13 und 20, ... 23: Zulässig bei S7-1500-R/H-CPUs
  Die Werte 10, ... 13 von Mode beziehen sich auf die CPU mit Redundanz-ID 1.
  Die Werte 20, ... 23 von Mode beziehen sich auf die CPU mit Redundanz-ID 2.

Über den Steuerparameter REQ aktivieren Sie das Auslesen der Memory Card. Über die Ausgangsparameter Done, Busy, Error und Status wird der Bearbeitungszustand des Auftrags angezeigt.

Abhängigkeit der auslesbaren Information vom Systemzustand des R/H-Systems

- Speichergröße

Systemzustand und ggf. Rolle der CPU	Pairing-Zustand der CPUs	Zulässige Werte von Mode
RUN-Solo, Primary-CPU	Pairing besteht nicht	• Redundanz-ID=1: Mode=10 (eigene SMC) • Redundanz-ID=2: Mode=20 (eigene SMC)
	Pairing besteht	• Redundanz-ID=1: – Mode=10 (eigene SMC) oder – Mode=20 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=10 (SMC der Partner-CPU) oder – Mode=20 (eigene SMC)
SYNCUP, Primary-CPU	-	• Redundanz-ID=1: – Mode=10 (eigene SMC) oder – Mode=20 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=10 (SMC der Partner-CPU) oder – Mode=20 (eigene SMC)
RUN-Redundant	-	• Redundanz-ID=1: – Mode=10 (eigene SMC) oder – Mode=20 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=10 (SMC der Partner-CPU) oder – Mode=20 (eigene SMC)

- Belegter Speicherplatz

Systemzustand und ggf. Rolle der CPU	Pairing-Zustand der CPUs	Zulässige Werte von Mode
RUN-Solo, Primary-CPU	irrelevant	• Redundanz-ID=1: Mode=11 (eigene SMC) • Redundanz-ID=2: Mode=21 (eigene SMC)
SYNCUP, Primary-CPU	-	Abhängig vom Startzeitpunkt von GetSMCinfo und der Synchronisation der beiden CPUs ist die Information lesbar oder nicht lesbar. Falls sie lesbar ist: • Redundanz-ID=1: – Mode=11 (eigene SMC) oder – Mode=21 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=11 (SMC der Partner-CPU) oder – Mode=21 (eigene SMC)
RUN-Redundant	-	• Redundanz-ID=1: – Mode=11 (eigene SMC) oder – Mode=21 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=11 (SMC der Partner-CPU) oder – Mode=21 (eigene SMC)

Hinweis

Ermittlung des belegten Speicherplatzes von Speicherkarten unterschiedlicher Speichergröße

Falls Sie in den beiden CPUs eines R/H-Systems Speicherkarten unterschiedlicher Speichergröße einsetzen, kann die Ermittlung des belegten Speicherplatzes trotz identischen Speicherinhalts zu unterschiedlichen Ergebnissen führen. Ursache dafür ist die unterschiedliche Cluster-Größe der Speicherkarten.

- Verbrauchter Anteil der Lebensdauer

Systemzustand und ggf. Rolle der CPU	Pairing-Zustand der CPUs	Zulässige Werte von Mode
RUN-Solo, Primary-CPU	irrelevant	• Redundanz-ID=1: Mode=12 (eigene SMC) • Redundanz-ID=2: Mode=22 (eigene SMC)
SYNCUP, Primary-CPU	-	Abhängig vom Startzeitpunkt von GetSMCinfo und der Synchronisation der beiden CPUs ist die Information lesbar oder nicht lesbar. Falls sie lesbar ist: • Redundanz-ID=1: – Mode=12 (eigene SMC) oder – Mode=22 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=12 (SMC der Partner-CPU) oder – Mode=22 (eigene SMC)
RUN-Redundant	-	• Redundanz-ID=1: – Mode=12 (eigene SMC) oder – Mode=22 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=12 (SMC der Partner-CPU) oder – Mode=22 (eigene SMC)

- Parametrierte Schwelle der Lebensdauer

Systemzustand und ggf. Rolle der CPU	Pairing-Zustand der CPUs	Zulässige Werte von Mode
RUN-Solo, Primary-CPU	irrelevant	• Redundanz-ID=1: – Mode=13 (eigene SMC) oder – Mode=23 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=13 (SMC der Partner-CPU) oder – Mode=23 (eigene SMC)
SYNCUP, Primary-CPU	-	• Redundanz-ID=1: – Mode=13 (eigene SMC) oder – Mode=23 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=13 (SMC der Partner-CPU) oder – Mode=23 (eigene SMC)
RUN-Redundant	-	• Redundanz-ID=1: – Mode=13 (eigene SMC) oder – Mode=23 (SMC der Partner-CPU) • Redundanz-ID=2: – Mode=13 (SMC der Partner-CPU) oder – Mode=23 (eigene SMC)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetSMCinfo":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerparameter Request Aktiviert das Auslesen der Memory Card bei REQ = "1".
Mode	Input	UINT	Über den Parameter "Mode" wählen Sie aus, welche Information über die SIMATIC Memory Card Sie auslesen wollen: • 0: Speichergröße in KByte (1 KByte = 1024 Byte) • 1: Belegter Speicherplatz in KByte • 2: Wartungsinformation: Bisher verbrauchter Anteil der Lebensdauer in % Hinweis: – Bei den S7-1500 Software Controllern und S7-PLCSIM wird bei Mode=2 stets 0x00 in Info eingetragen (keine Wartungsinformation verfügbar). – Bei SIMATIC Memory Cards, die diese Information generell nicht unterstützen oder die mit der aktuellen Firmware Ihrer CPU diese Information nicht liefern können, wird 0xFF in Info eingetragen. • 3: Parametrierter Anteil der Lebensdauer in %, nach dessen Ablauf die CPU einen Diagnosepuffereintrag erstellt und die Maintenance-LED einschaltet Hinweis: – Diese Information wird nicht aus der Memory Card ausgelesen, sondern aus den Projektierdaten. Falls Sie die Erzeugung des Diagnosealarms gesperrt haben, wird 0xFF in Info eingetragen. – Bei den S7-1500 Software Controllern und S7-PLCSIM wird bei Mode=3 stets 0xFF in Info eingetragen. • 10 oder 20: entspricht Mode 0 bei Standard-CPUs – Mode 10: Speichergröße der SMC in CPU mit Redundanz-ID 1 – Mode 20: Speichergröße der SMC in CPU mit Redundanz-ID 2 • 11 oder 21: entspricht Mode 1 bei Standard-CPUs – Mode 11: Belegter Speicherplatz der SMC in CPU mit Redundanz-ID 1 – Mode 21: Belegter Speicherplatz der SMC in CPU mit Redundanz-ID 2 • 12 oder 22: entspricht Mode 2 bei Standard-CPUs – Mode 12: Verbrauchter Anteil der Lebensdauer bei CPU mit Redundanz-ID 1 – Mode 22: Verbrauchter Anteil der Lebensdauer bei CPU mit Redundanz-ID 2 • 13 oder 23: entspricht Mode 3 bei Standard-CPUs – Mode 13: Parametrierte Schwelle der Lebensdauer bei CPU mit Redundanz-ID 1 – Mode 23: Parametrierte Schwelle der Lebensdauer bei CPU mit Redundanz-ID 2

Parameter	Deklaration	Datentyp	Beschreibung
Done	Output	BOOL	1: Die Anweisung wurde erfolgreich ausgeführt. Die gelesenen Informationen wurden an den Parameter Info übertragen.
Busy	Output	BOOL	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
Error	Output	BOOL	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter Status ausgegeben.
Status	Output	WORD	Fehlercode
Info	InOut	UDINT	Puffer für die ausgelesene Information

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Ermitteln der I&M 0-Daten

Die I&M 0-Daten der SIMATIC Memory Card können mit der Anweisung GetSMCinfo nicht ermittelt werden. Verwenden Sie dazu die Anweisung „Get_IM_Data (Seite 2696)“.

Parameter Status

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler.
7000	Kein Auftrag in Bearbeitung.
7001	Erstaufruf: Auftrag angestoßen (Busy = 1, Done = 0).
7002	Zwischenaufruf: Auftrag ist bereits aktiv (Busy = 1, Done = 0).
8080	Ungültiger Wert des Parameters "Mode" .
8081 (nur S7-1200)	Keine SIMATIC Memory Card gesteckt.
8086	Das R/H-System ist nicht im Systemzustand RUN-Redundant. Daher sind die Daten der Partner-CPU nicht verfügbar.
8092	Keine Daten verfügbar, z. B. weil "GetSMCinfo" von der CPU nicht unterstützt wird.
80C3	Die maximale Anzahl gleichzeitiger Aufrufe der Anweisung "GetSMCinfo" ist zur Zeit erreicht.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ermitteln Sie die Speichergröße der verwendeten SIMATIC Memory Card.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_GetSMCinfo			
	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	mode	UInt	0
4	getInfo	UDInt	0
5	done	Bool	false
6	busy	Bool	false
7	error	Bool	false
8	status	Word	16#0

Sie legen einen FB an. Im Statusbereich des FBs legen Sie eine lokale Variable "#done" mit dem Datentyp Bool an.

Die Parameter der Anweisung "GetSMCinfo" verschalten Sie wie folgt.

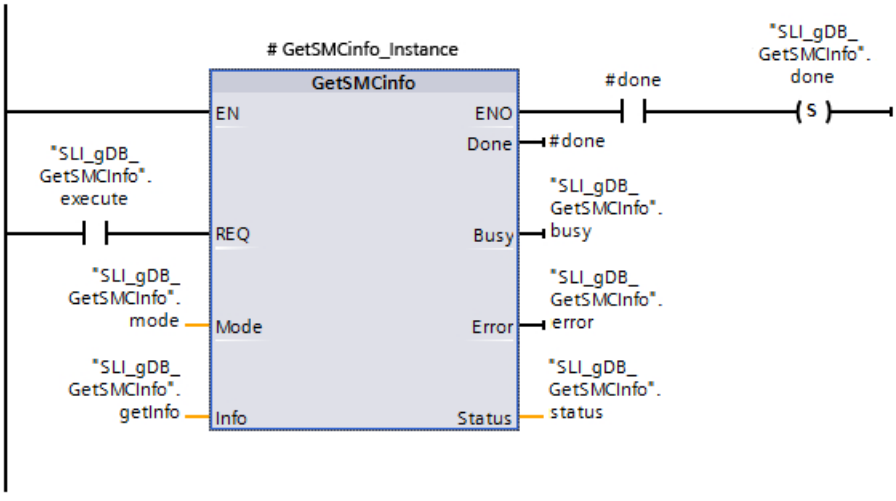

Wenn der Eingangsparameter REQ ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "GetSMCinfo" ausgeführt. Über den Eingangsparameter MODE ("mode") ist der zu verwendende Modus zum Auslesen der SIMATIC Memory Card hinterlegt. Entsprechend dem Wert "0" wird die Speichergröße der SIMATIC Memory Card gelesen und in KB am Parameter INFO ("getInfo") angezeigt. Der Erfolgstatus von GetSMCinfo wird am Ausgangsparameter DONE ("#done") angezeigt und in der Variable "done" gespeichert.

An Ausgangsparametern STATUS ("status") und ERROR ("error") wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verlief.

SLI_gDB_GetSMCinfo				
	Name	Data type	Start value	Monitor value
1	Static			
2	execute	Bool	false	TRUE
3	mode	UInt	0	0
4	getInfo	UDInt	0	2025008
5	done	Bool	false	TRUE
6	busy	Bool	false	FALSE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#0000

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GetClockStatus: Status der CPU-Uhr auslesen

Beschreibung

Mit der Anweisung "GetClockStatus" lesen Sie die folgenden Informationen über die CPU-interne Uhr aus:

- Information, ob Sie bei den Eigenschaften Ihrer CPU die Uhrzeitsynchronisation über NTP-Server aktiviert haben (Bit1 des Parameters ClockStatus).
- Information, ob Uhrzeitsynchronisierungen verpasst wurden (Bit0 des Parameters ClockStatus).
- Information, ob Sommerzeit oder Standardzeit aktiv ist (Bit2 des Parameters ClockStatus).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GetClockStatus":

Parameter	Deklaration	Datentyp	Beschreibung
ClockStatus	Output	WORD	Status der CPU-internen Uhr: • Bit0: Information, ob Uhrzeitsynchronisierungen verpasst wurden – Bit0 = 1: Eine oder mehrere Uhrzeitsynchronisierungen wurden verpasst. – Bit0 = 0: Keine Uhrzeitsynchronisierung wurde verpasst. Hinweis zur Auswertung von Bit0: Bit0 dürfen Sie nur bei aktivierter Uhrzeitsynchronisation auswerten, also wenn Bit1 = 1 ist. Bit0 wird in den folgenden Fällen auf 0 zurückgesetzt: – Nach einem Neustart der CPU – Nach dem Stellen der Uhrzeit Beachten Sie, dass bei einer S7-1500-CPU der früheste Umschlag auf 1 ca. drei Minuten nach einer verpassten Uhrzeitsynchronisation erfolgt. • Bit1 = 1: Uhrzeitsynchronisation ist aktiviert • Bit2 – = 1: Sommerzeit ist aktiv – = 0: Standardzeit ist aktiv • Bits 3 bis 15: 0 (reserviert)
Ret_Val	Output	INT	Rückgabewert

Weitere Informationen zu den gültigen Datentypen finden Sie unter: Übersicht über die gültigen Datentypen (Seite 243)

Parameter Ret_Val

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
8092	Keine Daten verfügbar, z. B. weil "GetClockStatus" von Ihrer CPU nicht unterstützt wird.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DeviceStates: Baugruppenzustandsinformation in einem IO-System lesen

Beschreibung

Über die Anweisung "DeviceStates" fragen Sie eine bestimmte Statusinformation für alle Baugruppen in einem IO-System ab, d. h.:

- Entweder zu allen IO-Devices innerhalb eines PROFINET-IO Systems
- Oder zu allen DP-Slaves innerhalb eines DP-Mastersystems

Ausgegeben wird als boolscher Wert, für welche Baugruppen der gewählte Status zutrifft. Zum Beispiel lesen Sie aus, welche IO-Devices in einem PROFINET-IO System gerade deaktiviert sind.

Darüber hinaus wird angezeigt, ob die zu lesende Statusinformation für mindestens eines der IO-Devices bzw. DP-Slaves zutrifft.

Die Anweisung kann sowohl im zyklischen OB, als auch in einem Alarm-OB (z. B. OB82 - Diagnosealarm) aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DeviceStates":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IOSYSTEM	E, A, M, L oder Konstante	Hardware-Kennung des PROFINET-IO- oder DP-Master-Systems (Beschreibung siehe unten)
MODE	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der zu lesenden Statusinformationen (Beschreibung siehe unten)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung (Beschreibung siehe unten)
STATE	InOut	VARIANT	E, A, M, D, L	Puffer für den Status der IO-Devices bzw. der DP-Slaves (Beschreibung siehe unten)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LADDR

Das PROFINET-IO- oder DP-Master-System wählen Sie an dem Parameter LADDR über die Hardware-Kennung aus.

Die Hardware-Kennung entnehmen Sie:

- Entweder in der Netzsicht den Eigenschaften des PROFINET-IO- oder DP-Master-Systems.
- Oder In der PLC-Variablentabelle den aufgelisteten Systemkonstanten mit dem Datentyp HW_IOSYSTEM.

Parameter MODE

Die zu lesende Statusinformation wählen Sie über den Parameter MODE aus. Es kann jeweils eine der folgenden Statusinformationen für das gesamte PROFINET-IO- oder DP-Master-System gelesen werden:

- 1: IO-Devices/DP-Slaves sind konfiguriert
- 2: IO-Devices/DP-Slaves gestört
- 3: IO-Devices/DP-Slaves deaktiviert
- 4: IO-Devices/DP-Slaves vorhanden
- 5: IO-Devices/DP-Slaves, bei denen ein Problem aufgetreten ist. Zum Beispiel:
  - Wartung erforderlich oder empfohlen
  - Nicht erreichbar
  - Nicht verfügbar
  - Fehler aufgetreten

Parameter STATE

Über den Parameter STATE wird der über den Parameter MODE ausgewählte Status der IO-Devices/DP-Slaves ausgegeben.

Wenn für ein IO-Device/DP-Slave der über MODE ausgewählte Status zutrifft, werden am Parameter STATE die folgenden Bits auf "1" gesetzt:

- Bit 0 = 1: Sammelanzeige. Das Bit n von mindestens einem IO-Device/DP-Slave wurde auf "1" gesetzt.
- Bit n = 1: Der über MODE ausgewählte Status trifft für den IO-Device/DP-Slave zu.
  - Bei einem PROFINET-IO System entspricht das Bit n der Gerätenummer des jeweiligen IO-Device (siehe Eigenschaften der PROFINET-Schnittstelle in der Geräte- und Netzsicht)
  - Bei einem PROFIBUS-DP System entspricht das Bit n der PROFIBUS-Adressse des DP-Slaves (siehe Eigenschaften des DP-Slaves in der Geräte- und Netzsicht)

Als Datentyp verwenden Sie BOOL oder Array of BOOL:

- Um ausschließlich das Bit zur Sammelanzeige der Statusinformationen auszugeben, können Sie am Parameter STATE den Datentyp BOOL verwenden.
- Um die Statusinformation zu allen IO-Devices/DP-Slaves auszugeben, verwenden Sie Array of BOOL mit folgender Länge:
  - Bei PROFINET-IO System: 1024 Bit
  - Bei DP-Master System: 128 Bit

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0	Kein Fehler
8091	Hardware-Kennung des Parameters LADDR nicht vorhanden. Überprüfen Sie (z. B. in den Systemkonstanten), ob der Wert für LADDR im Projekt existiert.
8092	LADDR adressiert kein PROFINET-IO- oder DP-Master-System.
8093	Ungültiger Datentyp am Parameter STATE.
80B1	Anweisung "DeviceStates" wird durch die CPU nicht unterstützt.
80B2	Der ausgewählte Parameter MODE wird von der verwendeten CPU für das am Parameter LADDR angegebene IO-System nicht unterstützt.
8452	Die vollständigen Status-Informationen passen nicht in die am Parameter STATE projektierte Variable. Hinweis: Bei der Prüfung der an STATE projektierten Variable auf ausreichende Feldlänge wird die Anweisung CountOfElements aufgerufen. Diese zählt die Füllelemente mit, wenn der Datentyp VARIANT auf ein Array of BOOL zeigt; bei einem Array [0...120] of BOOL z. B. wird als Feldlänge 128 ermittelt. Dies hat zur Folge, dass der Fehlercode W#16#8452 erst dann von DeviceStates zurückgeliefert wird, wenn die von Ihnen vorgegebene Anzahl der Feldelemente plus der Anzahl der von der CPU angelegten Füllelemente den Wert 1024 bzw. 128 unterschreitet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel - Auslesen des Vorhandenseins im PROFINET IO Mastersystem

In dem folgenden Beispiel fragen Sie die Existenz von IO-Devices in einem IO-System ab. Das IO-System besteht aus zwei CPUs der Baureihe S7-1500. Die CPU "PLC_14" soll das Programm, inklusive der Anweisung "DeviceStates", beinhalten. Die CPU "PLC_13" soll als IO-Device konfiguriert sein.

In der CPU "PLC_14": Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen und eine Struktur "mySTATE" (mit dem Datentyp Array of BOOL) an.

My_gDB_DeviceStates

	Name	Data type	Start value
1	Static		
2	myLADDR	HW_IOSYSTEM	258
3	myMODE	UInt	4
4	returnValue	Int	0
5	mySTATE	Array[0..1023] ...	
6	mySTATE[0]	Bool	false
7	mySTATE[1]	Bool	false
8	mySTATE[2]	Bool	false
9	mySTATE[3]	Bool	false
10	mySTATE[4]	Bool	false
11	mySTATE[5]	Bool	false
12	mySTATE[6]	Bool	false

In der CPU "PLC_14": Die Anweisung wird in einem zyklischen OB aufgerufen. Die Parameter der Anweisung verschalten Sie wie folgt.

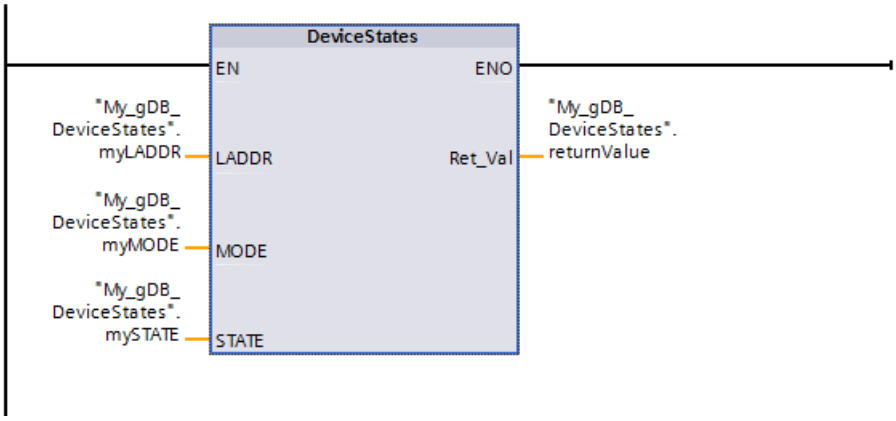

In der CPU "PLC_13": Über die CPU-Eigenschaften der CPU "PLC_13" richten Sie diese als IO-Device ein. Das IO-Device erhält die Gerätenummer 1.

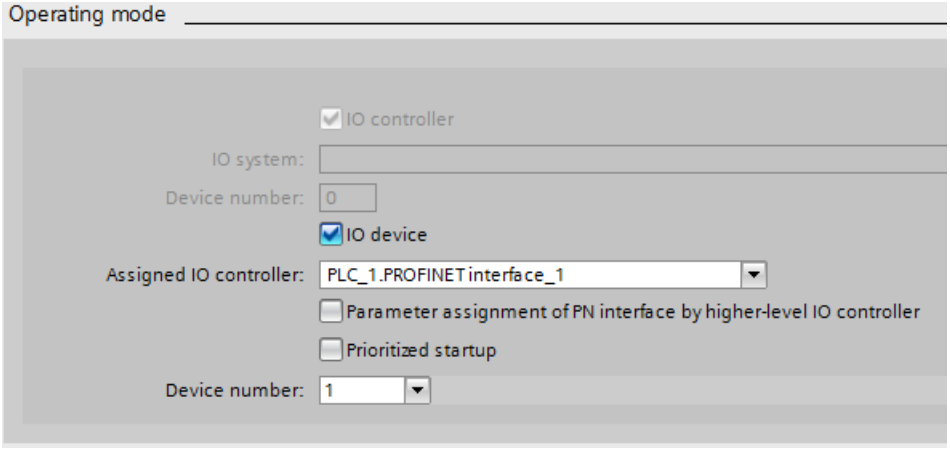

In der Netzansicht wird ein IO-System angezeigt.

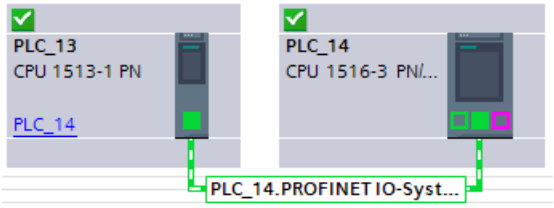

In der CPU "PLC_14": Durch den Parameter LADDR ("myLADDR") wird der Anweisung "DeviceStates" die HW-Kennung des IO-Systems bekannt gegeben. Entsprechend dem Wert "4" des Parameter MODE ("myMODE") wird das IO-System nach IO-Devices durchsucht.

Am Parameter STATE ("mySTATE") wird (gemäß dem Wert des Parameters MODE) für die IO-Devices das Vorhandensein ausgegeben. Das Bit 0 dient als Sammelwert und zeigt an, dass IO-Devices vorhanden sind. Das Bit 1 zeigt an, dass das IO-Device mit der Gerätenummer 1 vorhanden ist.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

My_gDB_DeviceStates

	Name	Data type	Start value	Monitor value
1	Static			
2	myLADDR	HW_IOSYSTEM	258	16#0102
3	myMODE	UInt	4	4
4	returnValue	Int	0	0
5	mySTATE	Array[0..1023] ...		
6	mySTATE[0]	Bool	false	TRUE
7	mySTATE[1]	Bool	false	TRUE
8	mySTATE[2]	Bool	false	FALSE
9	mySTATE[3]	Bool	false	FALSE
10	mySTATE[4]	Bool	false	FALSE
11	mySTATE[5]	Bool	false	FALSE
12	mySTATE[6]	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel - Auslesen der gestörten Stationen eines PROFINET IO Mastersystems

Ein PROFINET IO-System enthält 4 IO-Devices mit den Gerätenummern 1, 2, 3 und 4. Das IO-Device mit der Nummer 2 ist gestört.

Die Anweisung "DeviceStates" wird für das PROFINET IO-System mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem der IO-Devices liegt eine Störung vor.
- Bit 1 = 0: IO-Device mit der Gerätenummer 1 nicht gestört.
- Bit 2 = 1: IO-Device mit der Gerätenummer 2 gestört.
- Bit 3 = 0: IO-Device mit der Gerätenummer 3 nicht gestört.
- Bit 4 = 0: IO-Device mit der Gerätenummer 4 nicht gestört.
- Bit 5 = 0: irrelevant
- Bit 6 = 0: irrelevant
- ...

Beispiel - Auslesen der gestörten Stationen eines PROFIBUS DP Mastersystems

Ein DP-Mastersystem enthält 4 DP-Slaves mit den PROFIBUS-Adressen 3, 4, 5 und 6. Der DP-Slave mit der Adresse 4 ist gestört.

Die Anweisung "DeviceStates" wird für das DP-Mastersystem mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem der DP-Slaves liegt eine Störung vor.
- Bit 1 = 0: irrelevant
- Bit 2 = 0: irrelevant
- Bit 3 = 0: DP-Slave mit der Adresse 3 nicht gestört.
- Bit 4 = 1: DP-Slave mit der Adresse 4 gestört.
- Bit 5 = 0: DP-Slave mit der Adresse 5 nicht gestört.
- Bit 6 = 0: DP-Slave mit der Adresse 6 nicht gestört.
- Bit 7 = 0: irrelevant
- Bit 8 = 0: irrelevant
- ...

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ModuleStates: Modulzustandsinformation einer Baugruppe lesen

Beschreibung

Über die Anweisung "ModuleStates" lesen Sie Statusinformationen der Module eines PROFINET IO-Device oder PROFIBUS DP-Slave aus.

Ausgegeben wird als boolscher Wert, für welche Module der gewählte Status zutrifft. Zum Beispiel können Sie auslesen, welche Module eines PROFINET IO-Device gerade deaktiviert sind.

Darüber hinaus wird angezeigt, ob die zu lesende Statusinformation für mindestens eines der Module zutrifft.

Die Anweisung kann sowohl im zyklischen OB, als auch in einem Alarm-OB (z. B. OB82 - Diagnosealarm) aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ModuleStates":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_DEVICE	E, A, M, D, L oder Konstante	Hardware-Kennung der Station (Beschreibung siehe unten)
MODE	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der zu lesenden Statusinformationen der Module (Beschreibung siehe unten)
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung (Beschreibung siehe unten)
STATE	InOut	VARIANT	E, A, M, D, L	Puffer für den Status der Module (Beschreibung siehe unten)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LADDR

Das IO-Device oder den DP-Slave wählen Sie an dem Parameter LADDR über die Hardware-Kennung der Station aus.

Die Hardware-Kennung entnehmen Sie:

- Entweder in der Netzsicht, den Eigenschaften der Station des IO-Device oder des DP-Slave.
- Oder in der PLC-Variablentabelle den aufgelisteten Systemkonstanten mit dem Datentyp HW_DEVICE (für ein IO-Device) oder mit dem Datentyp HW_DPSLAVE (für ein DP-Slave).

Parameter MODE

Die zu lesende Statusinformation wählen Sie über den Parameter MODE aus. Es kann jeweils eine der folgenden Statusinformationen für die Module gelesen werden:

- 1: Module sind konfiguriert
- 2: Module gestört
- 3: Module deaktiviert
- 4: Module vorhanden
- 5: In den Modulen ist ein Problem aufgetreten. Zum Beispiel:
  - Wartung erforderlich oder empfohlen
  - Nicht erreichbar
  - Nicht verfügbar
  - Fehler aufgetreten

Parameter STATE

Über den Parameter STATE wird der über den Parameter MODE ausgewählte Status der Module ausgegeben.

Wenn für eine Baugruppe der über MODE ausgewählte Status zutrifft, werden die folgenden Bits auf "1" gesetzt:

- Bit 0 = 1: Sammelanzeige. Das Bit n von mindestens einem Modul wurde auf "1" gesetzt.
- Bit n = 1: Der über MODE ausgewählte Status trifft für das Modul im Slot n-1 zu (Beispiel: Bit 3 = Slot 2).

Als Datentyp verwenden Sie BOOL oder Array of BOOL:

- Um ausschließlich das Bit zur Sammelanzeige der Statusinformationen auszugeben, verwenden Sie am Parameter STATE den Datentyp BOOL.
- Um die Statusinformation zu allen Modulen auszugeben, verwenden Sie Array of BOOL mit einer Länge von 128 Bit.

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0	Kein Fehler
8091	Hardware-Kennung des Parameters LADDR nicht vorhanden. Überprüfen Sie (z. B. in den Systemkonstanten), ob der Wert für LADDR im Projekt existiert.
8092	LADDR adressiert kein IO-Device oder DP-Slave.
8093	Ungültiger Datentyp am Parameter STATE.
80B1	Die Anweisung "ModuleStates" wird durch die CPU nicht unterstützt.
80B2	Der ausgewählte Parameter MODE wird von der verwendeten CPU für das IO-Device / den DP-Slave am Parameter LADDR nicht unterstützt.

Fehlercode* (W#16#...)	Beschreibung
8452	Die vollständigen Status-Informationen passen nicht in die am Parameter STATE projektierte Variable. Hinweis: Bei der Prüfung der an STATE projektierten Variable auf ausreichende Feldlänge wird die Anweisung CountOfElements aufgerufen. Diese zählt die Füllelemente mit, wenn der Datentyp VARIANT auf ein Array of BOOL zeigt; bei einem Array [0...120] of BOOL z. B. wird als Feldlänge 128 ermittelt. Dies hat zur Folge, dass der Fehlercode W#16#8452 erst dann von ModuleStates zurückgeliefert wird, wenn die von Ihnen vorgegebene Anzahl der Feldelemente plus der Anzahl der von der CPU angelegten Füllelemente den Wert 128 unterschreitet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel fragen Sie die Existenz von Modulen eines PROFINET IO-Device ab. Das IO-System besteht aus zwei CPUs der Baureihe S7-1500. Die CPU "PLC_14" soll das Programm, inklusive der Anweisung "ModuleStates", beinhalten. DIe CPU "PLC_13" soll als IO-Device konfiguriert sein.

In der CPU "PLC_14": Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen und eine Struktur "mySTATE" (mit dem Datentyp Array of BOOL) an.

My_gDB_ModuleStates

	Name	Data type	Start value
1	Static		
2	myLADDR	HW_DEVICE	260
3	myMODE	UInt	4
4	returnValue	Int	0
5	mySTATE	Array[0..127] of Bool	
6	mySTATE[0]	Bool	false
7	mySTATE[1]	Bool	false
8	mySTATE[2]	Bool	false
9	mySTATE[3]	Bool	false
10	mySTATE[4]	Bool	false
11	mySTATE[5]	Bool	false
12	mySTATE[6]	Bool	false

In der CPU "PLC_14": Die Anweisung wird in einem zyklischen OB aufgerufen. Die Parameter der Anweisung verschalten Sie wie folgt.

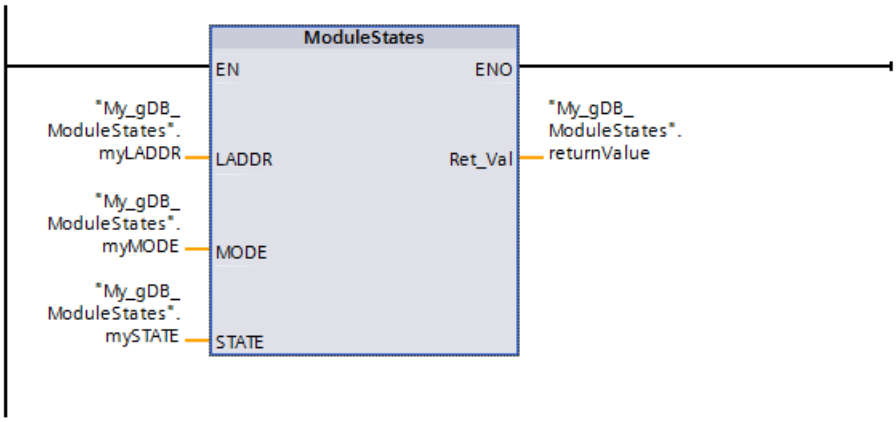

In der CPU "PLC_13": Über die CPU-Eigenschaften der CPU "PLC_13" richten Sie diese als IO-Device ein.

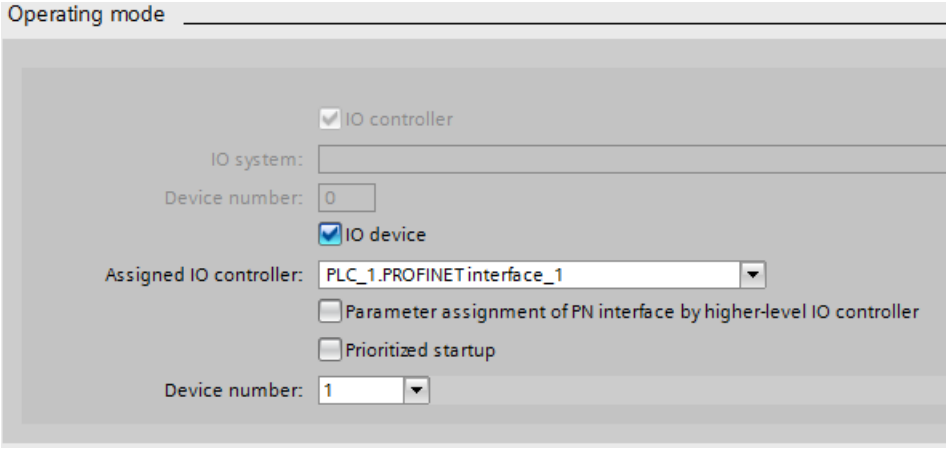

In der Netzansicht wird ein IO-System angezeigt.

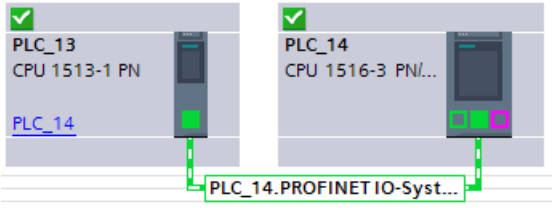

In der CPU "PLC_14": Ein Modul befindet sich im Slot 1 des IO-Device.

In der CPU "PLC_14": Durch den Parameter LADDR ("myLADDR") wird der Anweisung "ModuleStates" die HW-Kennung des IO-Device bekannt gegeben. Entsprechend dem Wert "4" des Parameter MODE ("myMODE") wird das IO-Device nach Modulen durchsucht.

Am Parameter STATE ("mySTATE") wird (gemäß dem Wert des Parameters MODE) für die Module das Vorhandensein ausgegeben. Das Bit 0 dient als Sammelwert und zeigt an, dass Module vorhanden sind. Das Bit 2 zeigt an, dass das im Slot 1 ein Modul vorhanden ist.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

My_gDB_ModuleStates

	Name	Data type	Start value	Monitor value
1	Static			
2	myLADDR	HW_DEVICE	260	16#0104
3	myMODE	UInt	4	4
4	returnValue	Int	0	0
5	mySTATE	Array[0..127] of Bool		
6	mySTATE[0]	Bool	false	TRUE
7	mySTATE[1]	Bool	false	FALSE
8	mySTATE[2]	Bool	false	TRUE
9	mySTATE[3]	Bool	false	FALSE
10	mySTATE[4]	Bool	false	FALSE
11	mySTATE[5]	Bool	false	FALSE
12	mySTATE[6]	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel

Ein IO-Device enthält 4 Module in den Slots 1 bis 4. Das Modul in Slot 2 ist gestört.

Die Anweisung "ModuleStates" wird für das IO-Device mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem Modul liegt eine Störung vor.
- Bit 1 = 0: Slot-Nummer 0 (durch IO-Device belegt)
- Bit 2 = 0: Modul in Slot Nummer 1 nicht gestört.
- Bit 3 = 1: Modul in Slot Nummer 2 gestört.
- Bit 4 = 0: Modul in Slot Nummer 3 nicht gestört.
- Bit 5 = 0: Modul in Slot Nummer 4 nicht gestört.
- Bit 6 = 0: irrelevant
- Bit 7 = 0: irrelevant

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GEN_DIAG: Diagnoseinformation generieren

Beschreibung

Die Anweisung "GEN_DIAG" generiert Diagnoseinformationen für Hardware-Komponenten anderer Hersteller zur Verwendung in der Diagnose des TIA-Portals. Zur Verwendung der Anweisung muss die vom Hersteller mitgelieferte GSD(GSDL/GSDML)-Datei zuvor installiert werden.

Die Anweisung generiert alle Diagnose-Ereignisse (auch für Wartungsbedarf).

- Über den Parameter LADDR wählen Sie die Hardware-Komponente aus, für die Sie ein Diagnose-Ereignis generieren möchten.
- Über den Parameter MODE spezifizieren Sie, ob es sich um ein kommendes oder gehendes Ereignis handelt.
- Über den Parameter DiagEvent definieren Sie das Diagnose-Ereignis in der Struktur DiagnosticDetail. Wenn Sie am Parameter DiagEvent eine Variable definieren, wird die Struktur wird automatisch in der lokalen Schnittstelle des Bausteins angelegt.

Die Bereitstellung der Diagnoseinformationen erfolgt synchron. Das Übermitteln der Diagnoseinformationen und die Ausgabe von Meldungen erfolgen asynchron.

ACHTUNG
Failsafe-spezifische Fehlermeldungen nicht zulässig Wenn Sie am Parameter DiagEvent Failsafe-spezifischen Diagnoseinformationen definieren, wird dies durch die Anweisung geprüft und der Fehlercode 80A1 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEN_DIAG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_ANY	E, A, M, D, L oder Konstante	Identifikationsnummer der Hardware-Komponente
MODE	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der Kommend-/Gehend-Information: • 1: Das spezifizierte Diagnose-Ereignis ist ein kommendes Ereignis • 2: Das spezifizierte Diagnose-Ereignis ist ein gehendes Ereignis • 3: Alle Diagnose-Ereignisse sind gehend. Hiernach liegt bei der Hardware-Komponente keine Störung vor (grünes Diagnosesymbol). Der Parameter DiagEvent wird bei MODE = 3 nicht ausgewertet.
DiagEvent	InOut	DiagnosticDetail	L, D	Spezifiziert das Diagnose-Ereignis (siehe "Parameter DiagEvent").
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung / Fehlermeldung (siehe "Parameter RET_VAL").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter DiagEvent

Die Struktur DiagnosticDetail ist ein Systemdatentyp zur Spezifizierung des Diagnose-Ereignises mit folgendem Aufbau:

Parameter		Datentyp	Beschreibung
DiagnosticDetail		Struct	
	ChannelInfo	WORD	Kanaleigenschaften (0...7)
	ALID	UINT	Lokale ID der Meldung. Anhand der ID wird die Meldung eindeutig identifiziert.
	TextID	UINT	ID eines Meldungstexts in einer Textliste.
	ChannelNumber	UINT	Herstellerspezifische Kanalnummer (0x0000 — 0x7FFF)
	Addval_0	DWORD	Platzhalter für zusätzliche Informationen. Der Wert / die Liste der Werte ist abhängig von dem Verbindungsfehler.
	TextID2	UINT	Texte für die Reaktion der CPU (Betriebszustand, OB-Aufrufe, etc.).
	LADDR	HW_ANY	Identisch zu dem Parameter LADDR.
	TextListId	UINT	• 0: Keine Textliste • ≠0: ID der Textliste
	ChannelDirection	UINT	• 0000: irrelevant • FFF1: Input • FFF2: Output • FFF3: In-/Output
	Addval_1	DWORD	Platzhalter für zusätzliche Informationen zu dem Kanalfehler (abhängig von der GSD-Datei). Zu den Kanalfehlertypen siehe auch: IEC-Norm 61158 (PROFINET IO Type 10 and PROFIBUS DP Type 3).

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
8080	Wert am Parameter MODE nicht unterstützt.
8090	Identifikationsnummer für Hardware-Komponente am Parameter LADDR nicht vorhanden.
8091	Die Generierung von Diagnoseinformationen ist für die am Parameter LADDR adressierte Hardware-Komponente nicht möglich.
80A1	• Inhalte der Struktur DiagnosticsDetail am Parameter DiagEvent ungültig oder inkonsistent. • Failsafe-spezifische Diagnoseinformationen am Parameter DiagEvent definiert (nicht zulässig).
80A4	Adressierte Hardware-Komponente nicht erreichbar.
80C1	Unzureichende Ressourcen für parallele Ausführung.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_DIAG: Diagnoseinformation lesen

Beschreibung

Über die Anweisung "GET_DIAG" lesen Sie die Diagnoseinformationen einer Hardware-Komponente aus. Die Auswahl der Hardware-Komponente erfolgt über den Parameter LADDR. Über den Parameter MODE wählen Sie aus, welche Diagnoseinformationen ausgelesen werden. Die S7-1200-CPU ignoriert bei MODE=0 den Parameter LADDR.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET_DIAG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	UINT	E, A, M, D, L oder Konstante	Über den Parameter MODE wählen Sie aus, welche Diagnosedaten ausgegeben werden.
LADDR	Input	HW_ANY (UINT)	E, A, M, L oder Konstante	Hardwarekennung der Hardware-Komponente
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung
CNT_DIAG	Output	UINT	E, A, M, D, L	Reserviert (immer "0").
DIAG	InOut	VARIANT	E, A, M, D, L	Diagnoseinformationen entsprechend dem ausgewählten Modus, siehe folgende Tabelle
DETAIL	InOut	VARIANT	E, A, M, D, L	Parameter ist ausgeblendet. Benutzen Sie diesen Parameter nicht!

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

Abhängig von dem Wert am Parameter MODE, werden an den Ausgangsparametern DIAG, CNT_DIAG und DETAIL unterschiedliche Diagnosedaten ausgegeben.

MODE	Beschreibung	DIAG	CNT_DIAG
0	Ausgabe aller unterstützten Diagnoseinformationen für eine Hardware-Komponente als DWORD, wobei Bit X=1 anzeigt, dass der Modus X unterstützt wird.	Bits des Datentyps DWORD: • Bit 0 = 1: MODE 0 wird unterstützt • Bit 1 = 1: MODE 1 wird unterstützt • Bit 2 = 1: MODE 2 wird unterstützt • Bit 3 bis 31: nicht relevant Bei MODE=0 ignoriert die S7-1200-CPU den Parameter LADDR.	0
1	Ausgabe des Diagnosestatus der adressierten Hardware-Komponente.	Struktur DIS (Beschreibung siehe unten): • MaintenanceState • ComponentStateDetail • OwnState • IOState • OperatingState	0
2	Ausgabe des Status aller untergeordneten Komponenten der adressierten Hardware-Komponente.	Struktur DNN (Beschreibung siehe unten): • SubordinateState • SubordinateIOState • DNNmode	0

Struktur DIS

Bei Parameter MODE = 1 werden die Diagnoseinformationen entsprechend der Struktur DIS ausgegeben. In diesem Fall geben Sie bei der Variablendeklaration den Systemdatentyp "DIS" als Datentyp an.

Hinweis

Ermittlung des OperatingState

Bei der Ermittlung des OperatingState einer Standard-CPU müssen Sie LADDR = 52 wählen (Hardwarekennung von CPUExecUnit).

Bei der Ermittlung des OperatingState einer CPU eines R/H-Systems müssen Sie LADDR = 65147 (Hardwarekennung von Local1~HCPUredCtrl) bzw. LADDR = 65347 (Hardwarekennung von Local2~HCPUredCtrl) wählen.

Bei der Ermittlung des Systemzustands eines R/H-Systems müssen Sie LADDR = 34 wählen (Hardwarekennung von Local1~RHSystem).

Die folgende Tabelle zeigt die Bedeutung der einzelnen Parameterwerte.

Parameter	Datentyp	Wert	Beschreibung	Regel zur Bildung des Parameters
MaintenanceState	DWORD	Enum		MaintenanceState hängt von ComponentStateDetail ab.
		0	Gut	• Bit 15 von ComponentStateDetail ist stets 1. • Bits 11 bis 14 von ComponentStateDetail können den Wert 0 oder 1 haben. • Bit 3 von ComponentStateDetail kann 1 sein. • Alle anderen Bits von ComponentStateDetail sind 0.
		1	Die Hardware-Komponente ist deaktiviert.	• Wenn Bit 16 von ComponentStateDetail 1 ist • Wenn Bit 17 von ComponentStateDetail 1 ist
		2	-	-
		3	-	-
		4	-	-
		5	Wartungsbedarf	Wenn Bit 4 von ComponentStateDetail 1 ist
		6	Wartungsanforderung	Wenn Bit 5 von ComponentStateDetail 1 ist
		7	Fehler	Wenn Bit 6 von ComponentStateDetail 1 ist
		8	Status unbekannt / Fehler in übergeordneter Hardware-Komponente	• Wenn die Bits 11 bis 14 von ComponentStateDetail den Wert 3 haben • Wenn die Bits 7 bis 10 von ComponentStateDetail den Wert 3 oder 4 haben
		9	-	-
		10	Eingänge/Ausgänge stehen nicht zur Verfügung.	• Wenn die Bits 11 bis 14 von ComponentStateDetail den Wert 2 haben • Wenn die Bits 7 bis 10 von ComponentStateDetail den Wert 1 oder 2 haben • Wenn Bit 18 von ComponentStateDetail 1 ist • Wenn Bit 19 von ComponentStateDetail 1 ist

Parameter	Daten-typ	Wert	Beschreibung	Regel zur Bildung des Parameters
ComponentS-tateDetail	DWORD	Bitar-ray	Status: • Bit 0 bis 15: Status der Hardware-Komponente • Bit 16 bis 31: Status der CPU	-
		0 bis 2 (Enum)	Zusatzinformation: • 0: Keine Zusatzinformationen • 1: Keine Übernahme zulässig	-
		3	Bit 3 = 1, wenn mindestens eine qualifizierte Diagnose vorhanden ist	-
		4	Bit 4 = 1: Wartungsbedarf bei mindestens einem Kanal oder der Komponente	-
		5	Bit 5 = 1: Wartungsanforderung bei mindestens einem Kanal oder der Komponente	-
		6	Bit 6 = 1: Fehler bei mindestens einem Kanal oder der Komponente	-
		7 bis 10 (Enum)	• 0: Own - Hardware-Komponente ist verfügbar • 1: Application Ready Pending - kein IO-Zugriff • 2: Locked by Superordinated - kein Zugriff auf die Hardware-Komponente • 3: Locked by IOController - kein Zugriff auf die Hardware-Komponente • 4: Locked by IOSupervisor - kein Zugriff auf die Hardware-Komponente	-
		11 bis 14 (Enum)	• 0 (Bit 11 = 0, Bit 12 = 0, Bit 13 = 0, Bit 14 = 0): OK - Das korrekte Element ist vorhanden. • 1 (Bit 11 = 1, Bit 12 = 0, Bit 13 = 0, Bit 14 = 0): Substitute - Das Element wurde durch ein kompatibles Element ersetzt. • 2 (Bit 11 = 0, Bit 12 = 1, Bit 13 = 0, Bit 14 = 0): Wrong - Das Element wurde durch ein nicht kompatibles Element ersetzt (Das Element ist erreichbar, aber nicht betriebsbereit.). • 3 (Bit 11 = 1, Bit 12 = 1, Bit 13 = 0, Bit 14 = 0): No Submodule - Das Element ist nicht vorhanden.	-
		15	Reserviert (immer "1")	-

Parameter	Daten-typ	Wert	Beschreibung	Regel zur Bildung des Parameters
		16 bis 31	Durch die CPU generierte Statusinformationen für Module: • Bit 16 = 1: Submodul oder Modul deaktiviert • Bit 17 = 1: CiR-Operation aktiv • Bit 18 = 1: Eingang nicht verfügbar • Bit 19 = 1: Ausgang nicht verfügbar • Bit 20 = 1: Überlauf Diagnosepuffer • Bit 21 = 1: Diagnose nicht verfügbar • Bit 22 = 1: Partial device failure (bei PNIO für Shared iDevices) • Bit 23 - 31: reserviert (immer 0)	-
OwnState	UINT	Enum	Wartungszustand	OwnState hängt von MaintenanceState ab.
		0	Gut	Wenn MaintenanceState den Wert 0 hat.
		1	Deaktiviert	Wenn MaintenanceState den Wert 1 hat.
		2	Wartungsbedarf	Wenn MaintenanceState den Wert 5 hat.
		3	Wartungsanforderung	Wenn MaintenanceState den Wert 6 hat.
		4	Fehler	Wenn MaintenanceState den Wert 7 hat.
		5	Nicht erreichbar	Wenn MaintenanceState den Wert 8 hat.
		6	Diagnosezustand unbekannt	Vorbelegungswert für Komponenten, die keine Diagnoseereignisse liefern.
		7	IO not available - Eingänge/Ausgänge stehen nicht zur Verfügung.	Wenn MaintenanceState den Wert 10 hat.

Parameter	Daten-typ	Wert	Beschreibung	Regel zur Bildung des Parameters
IOState	WORD	Bitar-ray	E/A-Status der Hardware-Komponente	Ein oder mehrere Bits können gesetzt sein. IOState hängt von ComponentStateDetail ab.
		Bit 0 = 1	Gut	• Bit 15 von ComponentStateDetail ist stets 1. • Bits 11 bis 14 von ComponentStateDetail können den Wert 0 oder 1 haben. • Bit 3 von ComponentStateDetail kann 1 sein. • Alle anderen Bits von ComponentStateDetail sind 0. Wenn Bit 0 = 1 ist, sind die Bits 1 bis 5 und das Bit 7 0.
		Bit 1 = 1	Deaktiviert	• Wenn Bit 16 von ComponentStateDetail 1 ist • Wenn Bit 17 von ComponentStateDetail 1 ist Wenn Bit 1 = 1 ist, sind alle anderen Bits 0.
		Bit 2 = 1	Wartungsbedarf	Wenn Bit 4 von ComponentStateDetail 1 ist
		Bit 3 = 1	Wartungsanforderung	Wenn Bit 5 von ComponentStateDetail 1 ist
		Bit 4 = 1	Fehler	Wenn Bit 6 von ComponentStateDetail 1 ist
		Bit 5 = 1	Hardware component not reachable - Hardware-Komponente nicht erreichbar	• Wenn die Bits 11 bis 14 von ComponentStateDetail den Wert 3 haben • Wenn die Bits 7 bis 10 von ComponentStateDetail den Wert 3 oder 4 haben Wenn Bit 5 = 1 ist, sind alle anderen Bits 0.
		Bit 6 = 1	Qualified: Bit 6 = 1, wenn mindestens eine qualifizierte Diagnose vorhanden ist	Wenn Bit 3 von ComponentStateDetail 1 ist In diesem Fall wird Bit 0, 2 oder 3 gesetzt.
		Bit 7 = 1	I/O data not available - E/A-Daten nicht verfügbar	• Wenn die Bits 11 bis 14 von ComponentStateDetail den Wert 2 haben • Wenn die Bits 7 bis 10 von ComponentStateDetail den Wert 1 oder 2 haben • Wenn Bit 18 von ComponentStateDetail 1 ist • Wenn Bit 19 von ComponentStateDetail 1 ist
		8 bis 14	reserviert (immer "0")	-
		Bit 15	Netzwerk- / Hardware-Fehler: • S7-1200: reserviert (immer "0")	-

Parameter	Daten-typ	Wert	Beschreibung	Regel zur Bildung des Parameters
			• S7-1500: Wenn Bit 4 = 1 oder Bit 5 = 1 ist: – Bit 15 = 0: Netzwerk-Fehler – Bit 15 = 1: Hardware-Fehler	

Parameter	Daten-typ	Wert	Beschreibung	Regel zur Bildung des Parameters
OperatingState	UINT	Enum		-
		0	Not Supported - OperatingState hat für E/A-Module immer den Wert 0.	-
		1	STOP / Firmware-Update	-
		2	STOP / Urlöschen	-
		3	STOP / Selbststart	-
		4	STOP	-
		5	Urlöschen	-
		6	Anlauf	-
		7	-	-
		8	RUN	-
		9	RUN-Redundant	-
		10	HALT	-
		11	-	-
		12	-	-
		13	DEFEKT (Hinweis: sieht man nur in Diagnosepuffer-Einträgen)	-
		14	-	-
		15	Spannungslos (Hinweis: sieht man nur in Diagnosepuffer-Einträgen)	-
		16	CiR	-
		17	STOP ohne ODIS	-
		18	RUN ODIS	-
		19	PgmTest	-
		20	RunPgmTest (Zustand der Primary-CPU, wenn die Backup-CPU sich im Testmode befindet)	-
		21	Run-Syncup	-
		22	SYNCUP (nur Backup-CPU im Systemzustand SYNCUP)	-
		31	Zustand der Partner-CPU unbekannt (z. B. wenn die Partner-CPU nicht verfügbar ist)	-
		32	-	-
		33	Systemzustand STOP	-
		34	reserviert	-
		35	Systemzustand ANLAUF	-
		36	reserviert	-
		37	Systemzustand RUN-Solo	-
		38	Systemzustand SYNCUP	-
		39	reserviert	-
		40	Systemzustand RUN-Redundant	-

Struktur DNN

Bei Parameter MODE = 2 werden die Details der Diagnoseinformationen entsprechend der Struktur DNN ausgegeben. In diesem Fall geben Sie bei der Variablendeklaration den Systemdatentyp "DNN" als Datentyp an.

Die folgende Tabelle zeigt die Bedeutung der einzelnen Parameterwerte.

Parameter	Datentyp	Wert	Beschreibung
SubordinateState	UINT	Enum	Status der untergeordneten Komponente (siehe Parameter OwnState der Struktur DIS)
SubordinateIOState	WORD	Bitarray	Status der Ein- und Ausgänge der untergeordneten Komponente (siehe Parameter IO State der Struktur DIS)
DNNmode	WORD	Bitarray	• Bit 0 = 0: Diagnose aktiviert • Bit 0 = 1: Diagnose deaktiviert • Bit 1 bis 15: reserviert

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0	Kein Fehler
n	Datenbereich am Parameter DETAIL ist zu klein. Nicht alle Details der Diagnosedaten werden ausgegeben.
8080	Wert am Parameter MODE nicht unterstützt.
8081	Typ am Parameter DIAG wird mit dem ausgewählten Modus (Parameter MODE) nicht unterstützt.
8082	Typ am Parameter DETAIL wird mit dem ausgewählten Modus (Parameter MODE) nicht unterstützt.
8090	LADDR nicht vorhanden
80C1	Unzureichende Ressourcen für parallele Ausführung.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie die Diagnoseinformationen einer CPU aus.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen und eine Struktur "myDIAG" (mit dem Datentyp DIS) an.

My_gDB_GET_DIAG			
	Name	Data type	Start value
1	Static		
2	diagMODE	UInt	1
3	myLADDR	HW_ANY	50
4	returnValue	Int	0
5	CountDiagDetails	UInt	0
6	myDIAG	DIS	
7	MaintainanceState	DWord	16#0
8	ComponentStateDetail	DWord	16#0
9	OwnState	UInt	0
10	IOState	Word	16#0
11	OperatingState	UInt	0

Die Parameter der Anweisung verschalten Sie wie folgt.

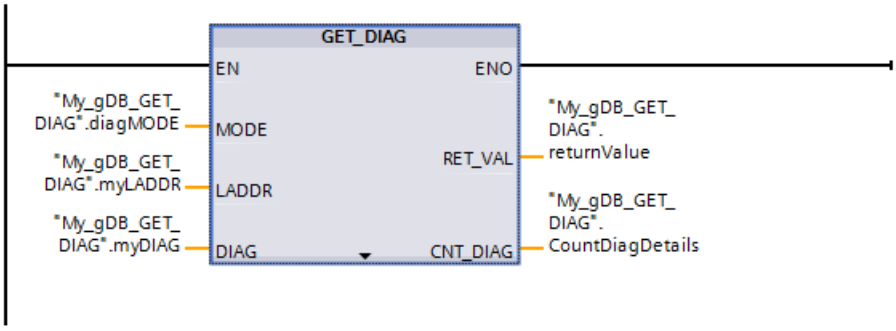

Durch den Parameter LADDR ("myLADDR") wird der Anweisung "GET_DIAG" die HW-Kennung der CPU bekannt gegeben. Entsprechend dem Wert "1" des Parameter MODE ("diagMODE") gilt Folgendes:

- Die Anweisung liest den Status des adressierten Hardware-Objekts (der CPU) aus.
- Am Parameter DIAG ("myDIAG") werden die Diagnoseinformationen in einer Struktur (Datentyp DIS) ausgegeben.

Zum Verständnis der Diagnoseinformationen müssen die Hex-Werte in Binärcode umgerechnet werden. Am Parameter DIAG ("myDIAG") wird Folgendes angezeigt:

- MaintenanceState: Entsprechend dem Wert "0" bedarf die CPU keiner Wartung.
- ComponentStateDetail: Entsprechend dem Hex-Wert "0000_8000" ist das Bit 15 aktiv.
- OwnState: Entsprechend dem Wert "0" ist keine Störung aufgetreten.
- IOState: Entsprechend dem Hex-Wert "0001" besteht kein Wartungsbedarf.
- OperatingState: Gibt "0" aus.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Am Ausgangsparameter CNT_DIAG ("CountDiagDetails") wird angezeigt, dass "0" Diagnosedetails des Parameter DETAIL ausgegeben wurden.

My_gDB_GET_DIAG

	Name	Data type	Start value	Monitor value
1	Static			
2	diagMODE	UInt	1	1
3	myLADDR	HW_ANY	50	50
4	returnValue	Int	0	0
5	CountDiagDetails	UInt	0	0
6	myDIAG	DIS		
7	MaintainanceState	DWord	16#0	16#0000_0000
8	ComponentStateDetail	DWord	16#0	16#0000_8000
9	OwnState	UInt	0	0
10	IOState	Word	16#0	16#0001
11	OperatingState	UInt	0	0

Hinweis: Sie könnten zum Beispiel aus der Variable ComponentStateDetail das Bit 3 (Kanaldiagnose ja/nein) einzeln auslesen.

- Adressieren Sie das Bit wie folgt: ComponentStateDetail.%X3.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Beispiel - Diagnose eines IO-Device

In dem folgenden Beispiel lesen Sie die Diagnoseinformationen eines IO-Device aus. Zur Diagnose verwenden Sie zum einen die Gerätesicht des IO-Device und zum anderen die Anweisung GET_DIAG.

Diagnose in der Gerätesicht

Eines der Module des IO-Device meldet einen Fehler. Der Gerätezustand ändert sich und wird wie folgt angezeigt:

Das Modul und die Kopfbaugruppe des IO-Device zeigen in der Gerätesicht einen Sammelzustand an. Deswegen wird bei beiden ein Fehler angezeigt.

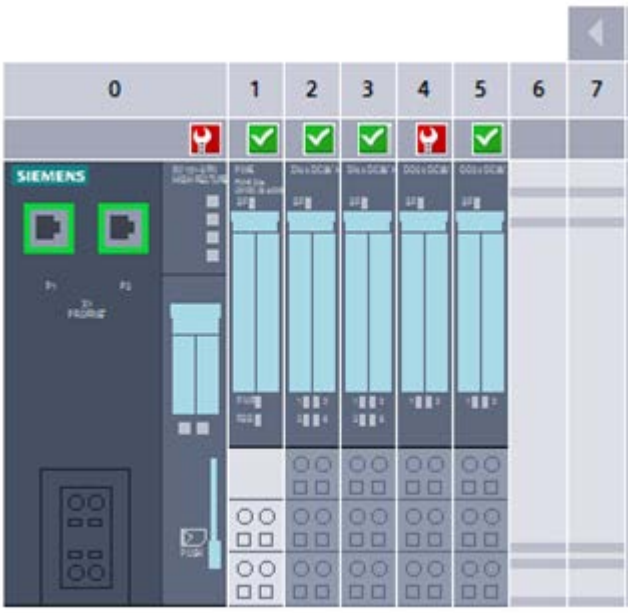

Diagnose über die Anweisung GET_DIAG

Zum Auslesen der Diagnoseinformationen des IO-Device addressieren Sie (via LADDR) die Kopfbaugruppe des IO-Device.

- Im Modus 1 (MODE hat den Wert 1) lesen Sie den Zustand des IO-Device aus.
- Im Modus 2 (MODE hat den Wert 2) lesen Sie den Zustand der unterlagerten Module aus.

Aufruf MODE 1: Eigene Diagnose	Aufruf MODE 2: Zusammenfassung der unterlagerten Diagnosen
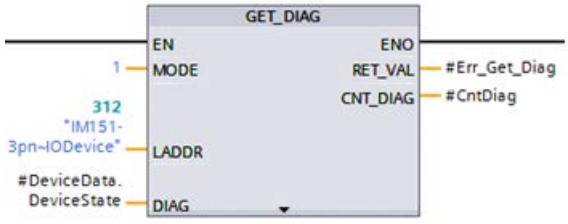	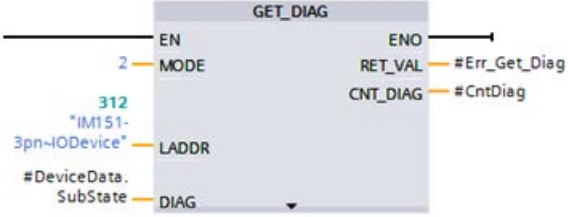

LADDR (HW-ID) = IM151-3pn~IODevice
DIAG (Struktur DIS) = #DeviceData.DeviceState
DIS.OwnState = 0 (Kein Fehler)
DIS.IOState = Bitfolge 0000 0000 0000 0001
	LADDR (HW-ID) = IM151-3pn~IODevice
	DIAG (Struktur DNN) = #DeviceData.SubState
	DNN.SubordinateState = 4 (Fehler)
	DNN.SubordianteIOState = Bitfolge 1000 0000 0001 0001

Ergebnis

Im Gegensatz zur Gerätesicht liefert die Anweisung GET_DIAG nicht den Sammelzustand des IO-Device. Der Fehler im unterlagerten Modul wird erst im Modus 2 ausgelesen.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.4.10 Puls

CTRL_PWM: Pulsweitenmodulation

Beschreibung

Mit der Anweisung "CTRL_PWM" aktivieren und deaktivieren Sie einen von der CPU unterstützten Impulsgenerator über die Software.

- Die Hardware-Kennung des Impulsgenerators, den Sie über die Anweisung steuern wollen, geben Sie am Eingang PWM ein.
- Wenn das Bit am Eingang ENABLE der Anweisung gesetzt ist, wird der Impulsgenerator aktiviert.
  - Wenn ENABLE den Wert TRUE hat, erzeugt der Impulsgenerator Impulse mit den in der Gerätekonfiguration festgelegten Eigenschaften.
  - Wenn das Bit am Eingang ENABLE zurückgesetzt wird oder die CPU in STOP geht, wird der Impulsgenerator deaktiviert und es werden keine Impulse mehr erzeugt.

Da die S7-1200 den Impulsgenerator aktiviert, wenn die Anweisung "CTRL_PWM" ausgeführt wird, hat BUSY bei der S7-1200 stets den Wert FALSE.

Wenn der Freigabeeingang EN den Signalzustand "1" liefert und keine Fehler während der Bearbeitung der Anweisung aufgetreten sind, wird der Freigabeausgang ENO gesetzt

Hinweis

Verwendung der Forcetabelle bei PWM und PTO

Digitale Ein- und Ausgänge, die für PWM und PTO verwendet werden, können nicht geforct werden. Digitale Ein- und Ausgänge, die über die Gerätekonfiguration zugewiesen wurden, werden weder durch die Forcetabelle noch durch die Beobachtungstabelle gesteuert.

Hinweis

Prozessabbild von PTO-/PWM-Ausgängen

Benutzen Sie die Ausgänge von PTO/PWM nicht im Prozessabbild (z. B. bei Zugriffen im Anwenderprogramm, bei Onlinefunktionen oder in HMI). Die Aktualisierungsrate des Prozessabbilds ist wesentlich geringer als die Geschwindigkeit der Signaländerungen. Die Anzeige im Prozessabbild gibt daher nicht den Signalverlauf wieder.

Voraussetzung

Eine Voraussetzung für die fehlerfreie Ausführung der Anweisung ist, dass der angegebene Impulsgenerator in der Hardware-Konfiguration aktiviert ist.

Öffnen Sie hierzu in der Gerätesicht die Eigenschaften der Baugruppe. Unter "Impulsgeneratoren (PTO/PWM)" öffnen Sie den gewünschten PTO/PWM und aktivieren unter "Allgemein" die Funktion "Diesen Impulsgenerator aktivieren".

Unter "Parametrierung" stellen Sie die Impulsoptionen ein.

Hinweis

Die Parametrierung eines Impulsgenerators erfolgt ausschließlich in der Gerätekonfiguration und nicht über die Anweisung "CTRL_PWM". Eine Parameteränderung, die auf der CPU wirksam werden soll, ist daher nur im Betriebszustand STOP der CPU möglich. Eine Ausnahme ist die Änderung der Impulsdauer.

Ändern der Impulsdauer aus dem Anwenderprogramm

Die in dem Dialog "Impulsoptionen" vorgenommene Einstellung zur Impulsdauer können Sie aus dem Anwenderprogramm ändern.

Der eingestellte Wert für "Anfängliche Impulsdauer" wird in die Ausgangsbytes des Impulsgenerators geschrieben. Die Anfangsadresse und die Endadresse werden in den Eigenschaften des Impulsgenerators unter "E/A-Adressen" angezeigt.

Um die Impulsdauer zu ändern, schreiben Sie die gewünschten Werte in die in der Gerätekonfiguration angegebene Ausgangswortadresse.

Beispiel:

- Für "Anfängliche Impulsdauer" wird als Wert 500 (dezimal) verwendet. Die Anfangsadresse des PTO/PWM ist "1000", die Endadresse "1001".
- In die beiden Ausgangsbytes wird der Binär-Wert "0000000111110100" (=500 dezimal) geschrieben.
  - Anfangsadresse (AB1000): 0000_0001 (BIN)
  - Endadresse (AB1001): 1111_0100 (BIN)

Beachten Sie, dass die Impulsdauer immer abhängig ist von dem parametrierten Impulsdauerformat (Hundertstel, Tausendstel, ...).

Ändern der Zykluszeit aus dem Anwenderprogramm

Aktivieren Sie das Optionskästchen "Änderung der Zykluszeit zur Laufzeit erlauben" in den "Impulsoptionen" des zugehörigen Impulsgenerators. Bei aktiviertem Optionskästchen enthalten die ersten beiden Ausgangsbytes die Impulsdauer und das dritte bis sechste Ausgangsbyte die Zykluszeit.

Während der Laufzeit des Impulsgenerators können Sie den Wert des Doppelworts am Ende des zugeordneten Speichers der Ausgänge verändern. Dies ändert die Zykluszeit des PWM-Signals.

Beispiel: Sie aktivieren das Optionskästchen, so dass die CPU PWM1 sechs Ausgangsbytes zuordnet, und Sie wählen dafür AB1008 bis AB1013. Nach dem Laden des Programms in die CPU und dem Start des Impulsgenerators können Sie die Impulsdauer durch Beschreiben von AW1008 und die Zykluszeit durch Beschreiben von AD1010 verändern.

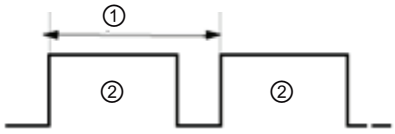

Bild 4-1 PWM_Beispiel

① Zykluszeit
② Impulsdauer

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CTRL_PWM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PWM	Input	HW_PWM	E, A, M, D, L oder Konstante	Hardware-Kennung des Impulsgenerators Die Hardware-Kennung können Sie in der Gerätesicht den Eigenschaften des Impulsgenerators entnehmen. Die Hardware-Kennungen der Impulsgeneratoren werden auch in den Systemkonstanten aufgelistet.
ENABLE	Input	BOOL	E, A, M, D, L oder Konstante	Der Impulsgenerator wird bei ENABLE = TRUE aktiviert und bei ENABLE = FALSE deaktiviert.
BUSY	Output	BOOL	E, A, M, D, L	Bearbeitungsstatus
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung (siehe unten).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler
80A1	Hardware-Kennung des Impulsgenerators ungültig.
80D0	Impulsgenerator mit der angegebenen Hardware-ID ist nicht aktiviert. Aktivieren Sie den Impulsgenerator in den Eigenschaften der CPU unter "Impulsgeneratoren (PTO/PWM)".
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

CTRL_PTO: Eine Impulsfolge mit vorgegebener Frequenz ausgeben

Beschreibung

Die Anweisung "CTRL_PTO" stellt Ihnen eine Impulsfolge mit vorgegebener Frequenz zur Verfügung. Sie benötigen dazu keinen Achsen-DB eines Technologie-Objekts.

Voraussetzung

Die Benutzung der Anweisung "CTRL_PTO" setzt einen aktivierten Impulsgenerator voraus. Diese Aktivierung nehmen Sie in der Hardware-Konfiguration vor, ebenso wie die Wahl des Signaltyps. Sie müssen den Impulsgenerator am Parameter "PTO" angeben und den Parameter "REQ" auf TRUE setzen.

Hinweis

Verwendung eines Impulsgenerators

Ein Impulsgenerator kann zu jedem Zeitpunkt nur von einer Anweisung benutzt werden.

Hinweis

Verwendung der Forcetabelle bei PWM und PTO

Digitale Ein- und Ausgänge, die für PWM und PTO verwendet werden, können nicht geforct werden. Digitale Ein- und Ausgänge, die über die Gerätekonfiguration zugewiesen wurden, werden weder durch die Forcetabelle noch durch die Beobachtungstabelle gesteuert.

Hinweis

Prozessabbild von PTO-/PWM-Ausgängen

Benutzen Sie die Ausgänge von PTO/PWM nicht im Prozessabbild (z. B. bei Zugriffen im Anwenderprogramm, bei Onlinefunktionen oder in HMI). Die Aktualisierungsrate des Prozessabbilds ist wesentlich geringer als die Geschwindigkeit der Signaländerungen. Die Anzeige im Prozessabbild gibt daher nicht den Signalverlauf wieder.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CTRL_PTO":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	• REQ=1: Setzt die Frequenz des Impulsgenerators auf den Wert von FREQUENCY REQ=1 und FREQUENCY=0: Deaktiviert den Impulsgenerator • REQ=0: Keine Änderung am Impulsgenerator
PTO	Input	HW_PTO	Hardware-Kennung des Impulsgenerators Diese entnehmen Sie den Eigenschaften des Impulsgenerators in der Gerätesicht oder der Liste der Systemkonstanten.
FREQUENCY	Input	UDINT	Frequenz der auszugebenden Impulsfolge in Hz

Parameter	Deklaration	Datentyp	Beschreibung
DONE	Output	BOOL	Zustandsparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	Bearbeitungsstatus Da die S7-1200 den Impulsgenerator aktiviert, wenn die Anweisung "CTRL_PTO" ausgeführt wird, hat BUSY bei der S7-1200 stets den Wert FALSE.
ERROR	Output	BOOL	Zustandsparameter • 0: Kein Fehler • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.
STATUS	Output	WORD	Status der Anweisung (siehe unten)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter Status

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
8090	Der Impulsgenerator mit der angegebenen Hardware-ID wird bereits benutzt.
8091	Bereichsüberschreitung beim Parameter "FREQUENCY", Genaueres siehe Handbuch.
80A1	Der Parameter "PTO" adressiert keine Hardware-ID eins Impulsgenerators.
80D0	Der Impulsgenerator mit der angegebenen Hardware-ID ist nicht aktiviert oder hat nicht die Eigenschaft "PTO“. Aktivieren Sie den Impulsgenerator in den Eigenschaften der CPU unter "Impulsgeneratoren (PTO/PWM)" und wählen Sie eine Signalart "PTO".
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.4.11 Rezepturen und Data Logging

Rezeptfunktionen

Rezeptfunktionen - Übersicht

Beschreibung

Mit den Anweisungen "RecipeExport" und "RecipeImport" übertragen Sie Rezeptdaten

- aus einem Rezept-Datenbaustein in eine CSV-Datei auf der Memory Card der CPU ("RecipeExport")
- aus einer CSV-Datei auf der Memory Card der CPU in einen Rezept-Datenbaustein ("RecipeImport")

Hinweis

Lebensdauer von Memory Cards

Die Lebensdauer von Memory Cards ist begrenzt. Nähere Informationen finden Sie hier: AUTOHOTSPOT

Schutz der Rezeptdaten

Die Anweisungen "RecipeExport" und "RecipeImport" sind nicht für Anwendungen geeignet, bei denen für die Rezeptdaten ein besonderer Schutz notwendig ist. Für höheren Schutz der Daten verwenden Sie die Rezeptfunktion von HMI.

Lage des Rezept-Datenbausteins bei "RecipeExport"

Der Datenbaustein mit den Rezeptdaten, welche über "RecipeExport" exportiert werden, kann sich im Arbeits- oder im Ladespeicher befinden. Wenn der Datenbaustein sowohl im Arbeits- als auch im Ladespeicher vorhanden ist, werden die Werte aus dem Arbeitsspeicher verwendet.

Trennzeichen

Um die Zeilenwerte des Rezepts voneinander zu trennen, können Sie unterschiedliche Trennzeichen verwenden. Beachten Sie hierbei, dass wenn Sie die CSV-Datei mit den exportierten Rezeptdaten zum Beispiel in Excel öffnen, Kommas durch Punkte ersetzt werden können und umgekehrt. Die Ursache hierfür liegt darin, dass unterschiedliche Sprachen verschiedene Dezimaltrennzeichen verwenden. Dies kann sich auf den Import von CSV-Dateien auswirken. Als Konvention gilt für "RecipeImport", dass das erste erkannte mögliche Trennzeichen als Trennzeichen für sämtliche Werte einer Tabellenzeile angenommen wird.

- Beispiel 1: Wenn als erstes Trennzeichen ein Semikolon erkannt wird, wird beim Import davon ausgegangen, dass das Semikolon als Trennzeichen für alle weiteren Werte verwendet wird. Als Dezimaltrennzeichen bei einem Wert vom Typ REAL wird dann ein Komma verwendet.
- Beispiel 2: Wenn als erstes Trennzeichen ein Komma erkannt wird, wird beim Import davon ausgegangen, dass das Komma als Trennzeichen für alle weiteren Werte verwendet wird. Als Dezimaltrennzeichen bei einem Wert vom Typ REAL wird dann ein Punkt verwendet.

Wenn der Rezeptdatensatz eine Zeichenkette enthält, achten Sie darauf, dass die Zeichenkette nicht das verwendete Trennzeichen enthält. Achten Sie bei Zeichenketten auch darauf, dass diese keine Steuerzeichen enthalten, die einen Zeilenumbruch angeben (bei ASCII: LF, CR LF, CR).

Hinweis: Leerzeichen und Tabulatoren werden nicht als Trennzeichen verwendet. Diese führen beim Import zum Fehlercode 80B2.

Hinweis

Asynchrone Bearbeitung der Rezeptfunktionen

Wegen der asynchronen Bearbeitung der Rezeptfunktionen müssen Sie die Daten im Quellbereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

RecipeExport: Rezept exportieren

Beschreibung

Die Anweisung "RecipeExport" exportiert die Rezept-Daten aus einem Datenbaustein in eine CSV-Datei im Ladespeicher.

Bei S7-1500-CPUs kann der zu exportierende DB sowohl im Arbeitsspeicher als auch im Ladespeicher vorhanden sein. Falls er in beiden Speichern vorhanden ist, werden die Werte aus dem Arbeitsspeicher exportiert. Bei S7-1200-CPUs muss der zu exportierende DB im Ladespeicher vorhanden sein.

Die CSV-Datei wird im Ladespeicher erstellt. Dies ist bei S7-1500-CPUs die Memory Card, bei S7-1200-CPUs der interne oder der externe Ladespeicher (falls eine Memory Card vorhanden ist).

Beim Exportieren wird die CSV-Datei in dem Ordner "Recipes" im Hauptverzeichnis des Ladespeichers erstellt. Als Dateiname der erstellten CSV-Datei wird der Name des Datenbausteins verwendet. Wenn bereits eine CSV-Datei mit dem gleichen Namen vorhanden ist, wird diese beim Export überschrieben.

Jeder Rezept-Datensatz erzeugt eine Zeile der CSV-Datei. Diese Zeilen sind wie folgt aufgebaut:

- Die erste Zeile enthält den Eintrag „Index“ sowie die Namen der Rezeptbestandteile.
- Jede weitere Zeile enthält die Nummer des Rezept-Datensatzes (Index) sowie seine zugehörigen Bestandteile.

Wenn man die Datentypen des Rezept-DB zugrundelegt, beträgt die maximale Größe eines Datensatzes 5000 Byte. Wenn sie überschritten wird, wird der Fehlercode W#16#8092 zurückgegeben. Diese Beschränkung auf 5000 Byte gilt entsprechend für die Größe der Kopfzeile. Wenn die Kopfzeile größer als 5000 Byte ist, wird sie bei einer S7-1500-CPU bis zur 5000-Byte-Grenze exportiert und der Export wird mit dem ersten Rezept-Datensatz weitergeführt. Eine S7-1200-CPU lehnt in diesem Fall den Export gänzlich ab und liefert den Fehlercode W#16#8092.

Die S7-1200-CPUs überwachen zusätzlich die Anzahl der Spalten. Sie darf 255 nicht überschreiten (Dabei wird die Index-Spalte nicht mitgezählt.).

Wenn der Rezept-DB ein DB mit Standardzugriff ist und aus mehr Komponenten besteht als erlaubt (z. B. aus zwei ARRAYs und einer Struktur), wird nur das erste ARRAY exportiert und STATUS wird auf W#16#0000 gesetzt.

Arbeitsweise

Der Export wird durch den Parameter REQ angestoßen. Während des Exports wird der Parameter BUSY auf "1" gesetzt. Nach Ausführung der Anweisung wird BUSY zurück auf "0" gesetzt und der Abschluss der Ausführung am Parameter DONE mit "1" angezeigt.

Wenn bei der Ausführung ein Fehler auftreten sollte, wird dieser über die Parameter ERROR und STATUS gemeldet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RecipeExport":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Steuerparameter REQUEST: Aktiviert den Export bei steigender Flanke.
RECIPE_DB	InOut	VARIANT	D	Zeiger auf den Rezept-Datenbaustein. Zum Aufbau des Datenbausteins siehe: Aufbau eines Rezept-DBs (Seite 2764)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Die Anweisung wird nicht ausgeführt. • 1: Die Anweisung wird ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler. • 1: Ein Fehler liegt vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten
7000	Keine Auftragsbearbeitung aktiv
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8090	Dateiname der CSV-Datei enthält ungültige Zeichen. Der Dateiname für die CSV-Datei ist identisch mit der des Datenbausteins.
8091	Die durch RECIPE_DB referenzierte Datenstruktur kann nicht verarbeitet werden, oder die Anzahl der Datenspalten in einer S7-1200-CPU ist größer als 255. Die CSV-Datei wurde möglicherweise teilweise beschrieben.
8092	Datenstruktur am Parameter RECIPE_DB überschreitet 5000 Byte. Bei S7-1200-CPUs zusätzlich: Die Größe der Kopfzeile ist größer als 5000 Byte. Der Export wird zurückgewiesen.
80B3	Kein ausreichender Speicherplatz auf der Memory-Card oder dem internen Ladespeicher.
80B4	Die Memory-Card ist schreibgeschützt.
80B6	(Nur bei S7-1200-CPUs) Attribut "Nur in Ladespeicher abspeichern" des Rezept-Datenbausteins ist nicht aktiviert.
80C0	CSV-Datei vorübergehend gesperrt.
80C1	Rezept-Datenbaustein vorübergehend gesperrt.
80C3	Zu wenig Ressourcen
8624	RECIPE_DB zeigt auf einen unzulässigen Bereich, z. B. auf Lokaldaten.
8652	Die Variable an RECIPE_DB ist unzureichend. Die CSV-Datei wurde nicht beschrieben..
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm für Rezeptfunktionen (Seite 2767).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

RecipeImport: Rezept importieren

Beschreibung

Die Anweisung "RecipeImport" importiert die Rezept-Daten aus einer CSV-Datei des Ladespeichers in den Datenbaustein am Parameter RECIPE_DB.

Die CSV-Datei liegt bei S7-1500-CPUs auf der Memory Card, bei S7-1200-CPUs im internen oder im externen Ladespeicher (falls eine Memory Card vorhanden ist).

Die Werte in dem Datenbaustein werden dabei überschrieben. Bei S7-1500-CPUs kann der DB, in den die Rezept-Daten importiert werden, sowohl im Arbeitsspeicher als auch im Ladespeicher vorhanden sein. Falls er in beiden Speichern vorhanden ist, wird in den DB im Arbeitsspeicher importiert. Bei S7-1200-CPUs muss der DB, in den die Rezeptdaten importiert werden, im Ladespeicher vorhanden sein.

Das Rezept bleibt nur dann dauerhaft erhalten, wenn es in einen Ladespeicher-DB importiert wird. Falls es in einen Arbeitsspeicher-DB importiert wird und Sie es aber dauerhaft benötigen, müssen Sie es nach dem Import mit Hilfe der Anweisung WRIT_DBL in einen Ladespeicher-DB übertragen.

Beachten Sie für den Import der CSV-Datei Folgendes:

- Die CSV-Datei muss sich in dem Verzeichnis "Recipes" des Ladespeichers befinden.
- Der Name der CSV-Datei muss mit dem Namen des Datenbausteins am Parameter RECIPE_DB identisch sein.
- Jede Zeile der CSV-Datei entspricht einem Rezept-Datensatz. Die erste Zeile (Kopfzeile) der CSV-Datei enthält die Namen der Rezeptbestandteile (siehe auch: Aufbau eines Rezept-DBs (Seite 2764)). Die erste Zeile wird beim Import völlig ignoriert. Ein Abgleich der Namen der Rezeptbestandteile in der CSV-Datei und dem Datenbaustein wird beim Import nicht durchgeführt, und auch die Anzahl der Einträge ist irrelevant.
- Der jeweils erste Wert in jeder Zeile der CSV-Datei ist die Indexnummer des Rezepts. Die einzelnen Rezepte werden in Reihenfolge des Index importiert. Der Index in der CSV-Datei muss hierzu aufsteigend sein und darf keine Lücken enthalten (ist dies der Fall, wird am Parameter STATUS die Fehlermeldung 80B0 ausgegeben).
- Die CSV-Datei darf nicht mehr Rezepturdatensätze enthalten, als im Datenbaustein vorgesehen. Die maximale Anzahl an Datensätzen ist im Datenbaustein durch die Array-Grenzen vorgegeben.

- Wenn ein Wert der CSV-Datei nicht in den zugehörigen Datentyp des DB passt, wird der Import abgewiesen. Zeichenketten werden ggf. nach ihrer maximal zulässigen Länge abgeschnitten.
  Wenn der Datentyp im DB vom Typ BOOL ist, akzeptiert eine S7-1500-CPU alle Integer-Werte, wobei der Wert 1 auf TRUE und alle anderen Werte auf FALSE abgebildet werden. Eine S7-1200-CPU akzeptiert nur die Werte 0 und 1, alle anderen Werte werden abgewiesen.
- Wenn der Rezept-DB ein DB mit Standardzugriff ist und aus mehr Komponenten besteht als erlaubt (z. B. aus zwei ARRAYs und einer Struktur), werden die Rezeptdaten nur in das erste ARRAY importiert und STATUS wird auf W#16#0000 gesetzt.

Arbeitsweise

Der Import wird durch den Parameter REQ angestoßen. Während des Imports wird der Parameter BUSY auf "1" gesetzt. Nach Ausführung der Anweisung wird BUSY zurück auf "0" gesetzt und der Abschluss der Ausführung am Parameter DONE mit "1" angezeigt.

Wenn bei der Ausführung ein Fehler auftreten sollte, wird dieser über die Parameter ERROR und STATUS gemeldet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RecipeImport":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Steuerparameter REQUEST: Aktiviert den Import bei steigender Flanke
RECIPE_DB	InOut	VARIANT	D	Zeiger auf den Rezept-Datenbaustein. Zum Aufbau des Datenbausteins siehe: Aufbau eines Rezept-DBs (Seite 2764)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Die Anweisung wird nicht ausgeführt. • 1: Die Anweisung wird ausgeführt.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler. • 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten
7000	Keine Auftragsbearbeitung aktiv
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8090	Der Dateiname enthält ungültige Zeichen.
8091	Die durch RECIPE_DB referenzierte Datenstruktur kann nicht verarbeitet werden, z. B. weil der Datenbaustein nicht existiert.
8092	Keine passende CSV-Datei für den Import gefunden. Möglich Ursache: Der Name der CSV-Datei stimmt nicht mit dem Namen des Rezept-DBs überein.
80C0	CSV-Datei ist vorübergehend gesperrt.
80C1	Rezept-Datenbaustein ist vorübergehend gesperrt.
80B0	Nummerierung im Index der CSV-Datei ist nicht durchgängig, nicht aufsteigend oder übersteigt die maximale Anzahl (Array-Grenze) im Datenbaustein.
80B1	Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Die CSV-Datei enthält zu viele Felder.
80B2	Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Die CSV-Datei enthält zu wenige Felder. Hinweis: Die Zeilen der CSV-Datei werden eine nach der anderen gelesen. Falls eine Zeile der CSV-Datei mehr Zeichen enthält als von der CPU erwartet, wird der Fehlecode W#16#80B2 zurückgegeben, egal wie viele Felder erkannt wurden.
80B6	(Nur bei S7-1200-CPUs) Attribut "Nur in Ladespeicher abspeichern" des Rezept-Datenbausteins ist nicht aktiviert.
80C3	Zu wenig Ressourcen
80D0 +n	Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Datentyp im Feld n stimmt nicht überein (n<=46).
80FF	Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Datentyp im Feld n stimmt nicht überein (n>46).
8624	RECIPE_DB zeigt auf einen unzulässigen Bereich, z. B. auf Lokaldaten.
8630	Ziel ist schreibgeschützt, z. B. ein schreibgeschützter DB.
8652	Die Variable an RECIPE_DB ist unzureichend. Der Datenbaustein wurde nicht beschrieben.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm für Rezeptfunktionen (Seite 2767).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Aufbau eines Rezept-DBs

Einführung

Im Folgenden wird der Aufbau eines Rezept-DBs anhand eines einfachen Beispiels beschrieben. Der Rezept-DB besteht aus fünf Datensätzen, von denen drei verwendet werden. Der vierte und der fünfte Datensatz werden für spätere Erweiterungen freigelassen. Ein Datensatz enthält jeweils ein Rezept, welches sich aus dem Rezeptnamen und acht Zutaten zusammensetzt.

product-name	water	barley	wheat	hops	yeast	wa-terTmp	mashTmp	mash-Time	QTest
Pils	10	9	3	280	39	40	30	100	0
Lager	10	9	3	150	33	50	30	120	0
Black-Beer	10	9	3	410	47	60	30	90	1
Not_used	0	0	0	0	0	0	0	0	0
Not_used	0	0	0	0	0	0	0	0	0

Aufbau des Rezept-Datenbausteins

In einem globalen Datenbaustein werden die Rezeptdaten folgendermaßen umgesetzt:

- Die Vorlage für alle Rezepte ist der PLC-Datentyp "Beer_Recipe" mit den Rezeptbestandteilen "procutname", "water", usw. mit den entsprechenden Datentypen.
- In einem globalen Datenbaustein wird der PLC-Datentyp als Array [1.. 5] of "Beer_Recipe" verwendet. Über die Array-Grenzen (hier 1 bis 5) wird die maximale Anzahl von Rezepten vorgegeben, die der DB enthalten darf.
- Die Werte für die Rezeptbestandteile werden als Startwert in dem Datenbaustein ergänzt.
- Über den InOut-Parameter RECIPE_DB wird der globale DB mit der Anweisung verschaltet.

Recipe_DB				
	Name	Data type	Offset	Start value
1	Static			
2	Products	Array [1 .. 5] of "Beer_Recipe"	...	
3	Products[1]	"Beer_Recipe"	...	
4	Products[2]	"Beer_Recipe"	...	
5	Products[3]	"Beer_Recipe"	...	
6	productname	String[20]	...	'BlackBeer'
7	water	UInt	...	10
8	barley	UInt	...	9
9	wheat	UInt	...	3
10	hops	UInt	...	410
11	yeast	UInt	...	47
12	waterTmp	UInt	...	60
13	mashTmp	UInt	...	30
14	mashTime	UInt	...	90
15	QTest	UInt	...	1
16	Products[4]	"Beer_Recipe"	...	
17	Products[5]	"Beer_Recipe"	...	

Export in CSV-Datei

Nach Ausführung der Anweisung "RecipeExport (Seite 2758)" werden die Daten des DBs in eine CSV-Datei mit dem folgenden Aufbau geschrieben:

Recipe_DB.csv

```
index,productname,water,barley,wheat,hops,yeast,waterTmp,mashTmp,mashTime,QTest
1,”Pils”,10,9,3,280,39,40,30,100,0
2,”Lager”,10,9,3,150,33,50,30,120,0
3,”BlackBeer”,10,9,3,410,47,60,30,90,1
4,”Not_used”,0,0,0,0,0,0,0,0,0
5,”Not_used”,0,0,0,0,0,0,0,0,0
```

Darstellung in Excel

Die CSV-Datei kann zur übersichtlichen Darstellung und Bearbeitung mit Excel geöffnet werden. Sollte beim Öffnen die Kommas als Trennzeichen nicht erkannt werden, nutzen Sie die Importfunktion von Excel, um die Daten strukturiert auszugeben:

	A	B	C	D	E	F	G	H	I	J	K
1	index	product	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
2	1	"Pils"	10	9	3	280	39	40	30	100	0
3	2	"Lager"	10	9	3	150	33	50	30	120	0
4	3	"BlackBeer"	10	9	3	410	47	60	30	90	1
5	4	"Not_used"	0	0	0	0	0	0	0	0	0
6	5	"Not_used"	0	0	0	0	0	0	0	0	0

Editieren der CSV-Datei

Über den Web-Server wird die CSV-Datei auf den PC/PG geladen und editiert. Nach der Bearbeitung kann die geänderte Datei wieder in die CPU geladen werden. Hierzu müssen Sie die bereits vorhandene CSV-Datei löschen.

Über die Anweisung "RecipeImport (Seite 2761)" können die geänderten Daten aus der CSV-Datei wieder in den Datenbaustein importiert werden.

Beachten Sie, dass die geänderten Daten hierfür noch mit dem Datenbaustein kompatibel sein müssen. Das heißt:

- In der Tabelle dürfen keine Änderungen an der Struktur vorgenommen werden (zum Beispiel durch Hinzufügen von Zutaten in einer neuen Spalte).
- Wenn Sie die Datei um zusätzliche Datensätze ergänzen, achten Sie beim Import in den Datenbaustein darauf, dass die Array-Grenzen, über die Sie die maximale Anzahl von Datensätzen vorgeben, mindestens der Anzahl der Datensätze entspricht.
- Beim Export in die CSV-Datei wird automatisch ein Index erzeugt. Wenn Sie zusätzliche Datensätze erstellen, ergänzen Sie die fortlaufende Index-Nummer entsprechend.
- Die Werte in den Tabellenzellen müssen den verwendeten Datentypen im Datenbaustein entsprechen, sowohl in Bezug auf das Format als auch die Länge.
  - Beispiel 1: Wenn im Datenbaustein der Datentyp INT verwendet wurde, verwenden Sie in der Tabelle nur Ganzzahlen verwenden.
  - Beispiel 2: Wenn im Datenbaustein der Datentyp SINT verwendet wurde, verwenden Sie in der Tabelle nur Ganzzahlen mit den Werten -128 bis +127 verwenden.

  Beachten Sie bei Änderungen in der Tabelle generell die zulässigen Datentypen und Datenbereiche, wie sie in der folgenden Tabelle aufgelistet sind.

Datentyp		Format	Hinweis
Gleitpunkt-zahlen	LReal	+9.9999999999999E+999	Immer in Exponentialschreibweise
	Real	+9.9999999E+99	Immer in Exponentialschreibweise
Ganzzahlen mit Vorzei-chen	LInt	+9999999999999999999	Wertebereich für Ganzzahlen mit Vorzeichen: -9223372036854775808 .. +9223372036854775807
	DInt	+9999999999	Wertebereich Ganzzahlen: -2147483648 bis +2147483647
	Int	+99999	Wertebereich Ganzzahlen: -32768 bis +32767
	SInt	+999	Wertebereich Ganzzahlen: -128 bis +127
Ganzzahlen ohne Vorzei-chen	ULInt	+9999999999999999999	Wertebereich Ganzzahlen: 0 bis +18446744073709551615
	UDInt	+9999999999	Wertebereich Ganzzahlen: 0 bis +4294967295
	UInt	+99999	Wertebereich Ganzzahlen: 0 bis +65535
	USInt	+999	Wertebereich Ganzzahlen: 0 bis +255
Binärzahlen	LWord	+9999999999999999999	Wertebereich Ganzzahlen: 0 bis +18446744073709551615
	DWord	+9999999999	Wertebereich Ganzzahlen: 0 bis +4294967295
	Word	+99999	Wertebereich Ganzzahlen: 0 bis +65535
	Byte	+999	Wertebereich Ganzzahlen: 0 bis +255
	Bool	9	Wertebereich: 0 oder 1

Datentyp		Format	Hinweis
Datum und Uhrzeit	LTIME	dddddd:hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	TIME	hhh:mm:ss.999	ISO-Format mit Millisekunden
	S5TIME	hhh:mm:ss.999	ISO-Format mit Millisekunden
	LDT	YYYY-MM-DD hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	DTL	YYYY-MM-DD hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	DT	YYYY-MM-DD hh:mm:ss.999	ISO-Format mit Millisekunden
	DATE	YYYY-MM-DD	ISO-Format
	LTime_Of_Day	hh:mm:ss.999_999_999	ISO-Format
	TOD	hh:mm:ss.999	ISO-Format
Zeichen	WString	"abcd"	• Zeichenfolge in doppelten Anführungszeichen. Beim Import eines Rezept-DBs können die doppelten Anführungszeichen entfallen. • aktuelle Länge • Eine Zeichenkette vom Datentyp WString besteht aus Elementen des Datentyps WChar. Beim Export eines Rezept-DBs werden die Inhalte von Zeichen des Datentyps WChar auf 16#FF beschränkt (Bsp.: 16#1255 wird zu 16#55 geändert.).
	String	"abcd"	• Zeichenfolge in doppelten Anführungszeichen. Beim Import eines Rezept-DBs können die doppelten Anführungszeichen entfallen. • aktuelle Länge
	WChar	"a"	Einzelnes Zeichen in doppelten Anführungszeichen. Beim Import eines Rezept-DBs können die doppelten Anführungszeichen entfallen. Beim Export eines Rezept-DBs werden die Inhalte von Zeichen des Datentyps WChar auf 16#FF beschränkt (Bsp.: 16#1255 wird zu 16#55 geändert.).
	Char	"a"	Einzelnes Zeichen in doppelten Anführungszeichen. Beim Import eines Rezept-DBs können die doppelten Anführungszeichen entfallen.

Beispielprogramm für Rezeptfunktionen

In dem folgenden Beispiel exportieren Sie die Rezept-Daten aus einem Datenbaustein (Rezept-DB) in eine CSV-Datei und importieren neue Rezept-Daten in den Datenbaustein.

Voraussetzung

Als Vorlage für die Rezept-Daten legen Sie einen PLC-Datentyp "Beer_Recipe" an.

Beer_Recipe

	Name	Data type	Default value
1	productname	String[20]	''
2	water	UInt	0
3	barley	UInt	0
4	wheat	UInt	0
5	hops	UInt	0
6	yeast	UInt	0
7	waterTmp	UInt	0
8	mashTmp	UInt	0
9	mashTime	UInt	0
10	QTest	UInt	0

Zur Ablage aller Rezeptdatensätze legen Sie einen Rezept-DB an. Weitere Informationen zur Erstellung eines Rezept-DBs finden Sie unter "Aufbau eines Rezept-DBs (Seite 2764)".

Recipe_DB

	Name	Data type	Start value
1	Static		
2	Products	Array[1..5] of "Beer_Recipe"	
3	Products[1]	"Beer_Recipe"	
4	Products[2]	"Beer_Recipe"	
5	Products[3]	"Beer_Recipe"	
6	productname	String[20]	'Blackbeer'
7	water	UInt	10
8	barley	UInt	9
9	wheat	UInt	3
10	hops	UInt	410
11	yeast	UInt	47
12	waterTmp	UInt	60
13	mashTmp	UInt	30
14	mashTime	UInt	90
15	QTest	UInt	1
16	Products[4]	"Beer_Recipe"	
17	Products[5]	"Beer_Recipe"	

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein zwölf Variablen an.

My_gDB_Recipefunctions			
	Name	Data type	Start value
1	Static		
2	expREQ	Bool	false
3	expDONE	Bool	false
4	expBUSY	Bool	false
5	expERROR	Bool	false
6	expSTATUS	Word	16#0
7	impREQ	Bool	false
8	impDONE	Bool	false
9	impBUSY	Bool	false
10	impERROR	Bool	false
11	impSTATUS	Word	16#0
12	outputErr_EXP	Byte	0
13	outputErr_IMP	Byte	0

Zum Anzeigen und Abspeichern einer CSV-Datei verwenden Sie den Webserver der CPU. Um den Webserver nutzen sowie die CSV-Datei bearbeiten zu können, ist Folgendes notwendig:

- Der Webserver muss in den CPU-Einstellungen aktviert sein.
- Ein Nutzerprofil in den Webserver-Eigenschaften der CPU mit den folgenden Einstellungen muss eingerichtet sein:
  - Der Nutzer muss Dateien lesen können.
  - Der Nutzer muss Dateien schreiben/löschen können.

Verschaltung der Parameter

Die Anweisungen werden in einem Funktionsbaustein aufgerufen.

Netzwerk 1: Die Parameter der Anweisung "RecipeExport" verschalten Sie wie folgt.

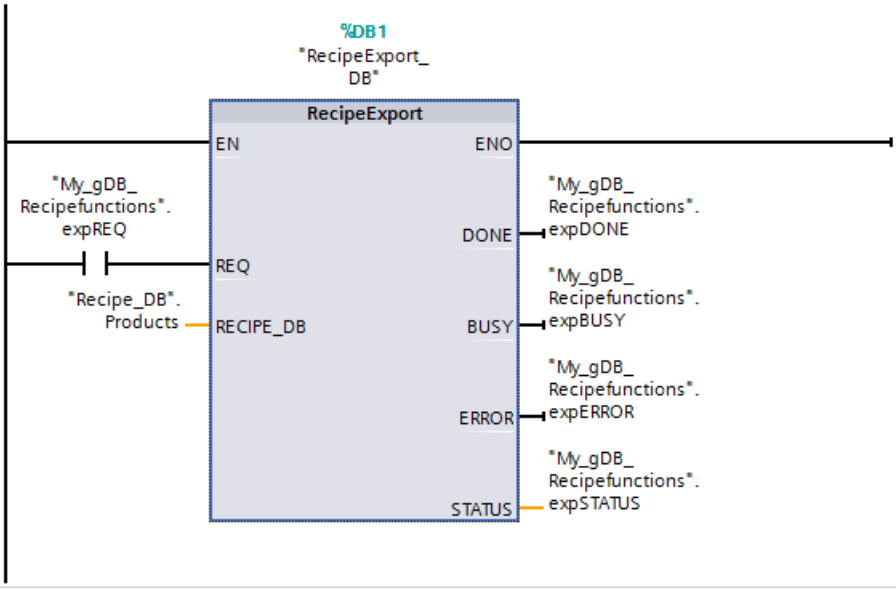

Netzwerk 3: Die Parameter der Anweisung "RecipeImport" verschalten Sie wie folgt.

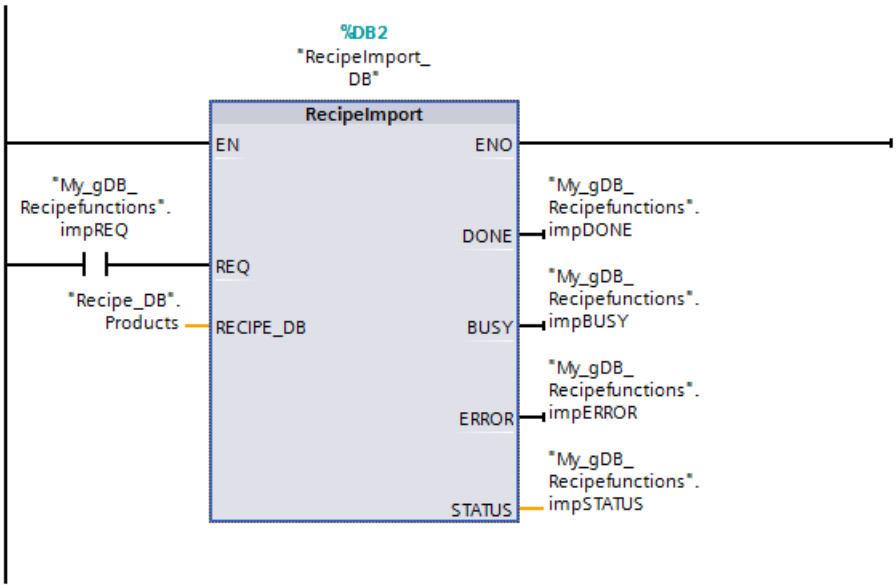

Netzwerk 2 und 4: Da die Parameter STATUS ("expSTATUS" und "impSTATUS") nur einen Zyklus lang gültig sind, erfassen Sie die Parameter mit einer Anweisung "MOVE". An den Ausgangsparametern OUT1 ("outputErr_EXP" bzw. "outputErr_IMP") geben Sie den Status im Falle eines Fehlers aus.

Die Parameter im Netzwerk 2 verschalten Sie wie folgt.

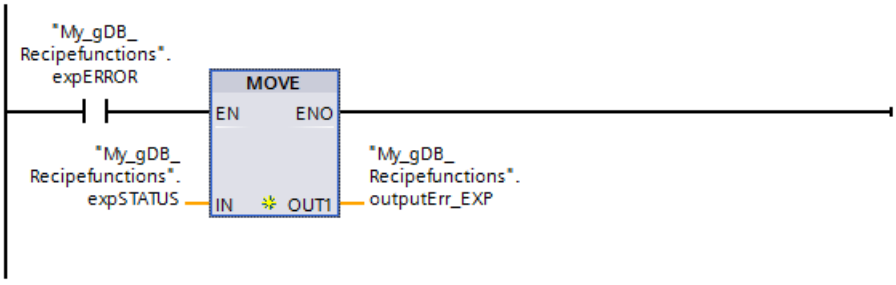

Die Parameter im Netzwerk 4 verschalten Sie wie folgt.

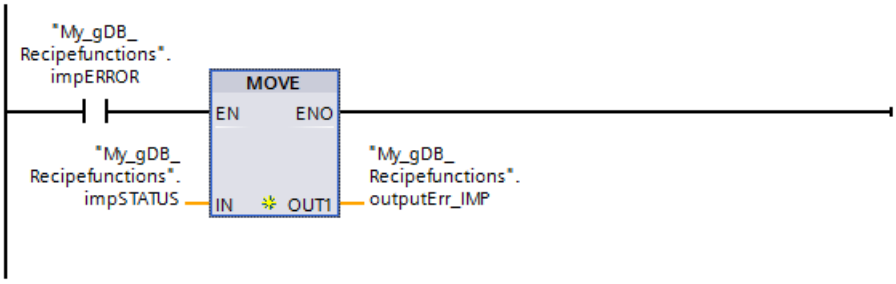

Anweisung "RecipeExport"

Am Parameter RECIPE_DB ist der zu exportierende Rezept-DB verschalten. Wenn der Parameter REQ ("expREQ") den Signalzustand "TRUE" liefert, wird der Exportvorgang gestartet. Aus den Daten des Rezept-DBs wird eine CSV-Datei generiert und im Rezeptordner im CPU-Speicher abgelegt. Am Ausgangsparameter STATUS ("expSTATUS") wird angezeigt, dass ein Signalwechsel erfolgt ist. Nachfolgend wird angezeigt, dass keine weitere Verarbeitung erfolgt (der Wert ist "7000").

Am Ausgangsparameter ERROR ("expERROR") wird angezeigt, dass die Verarbeitung der Anweisung ohne Fehler verlief.

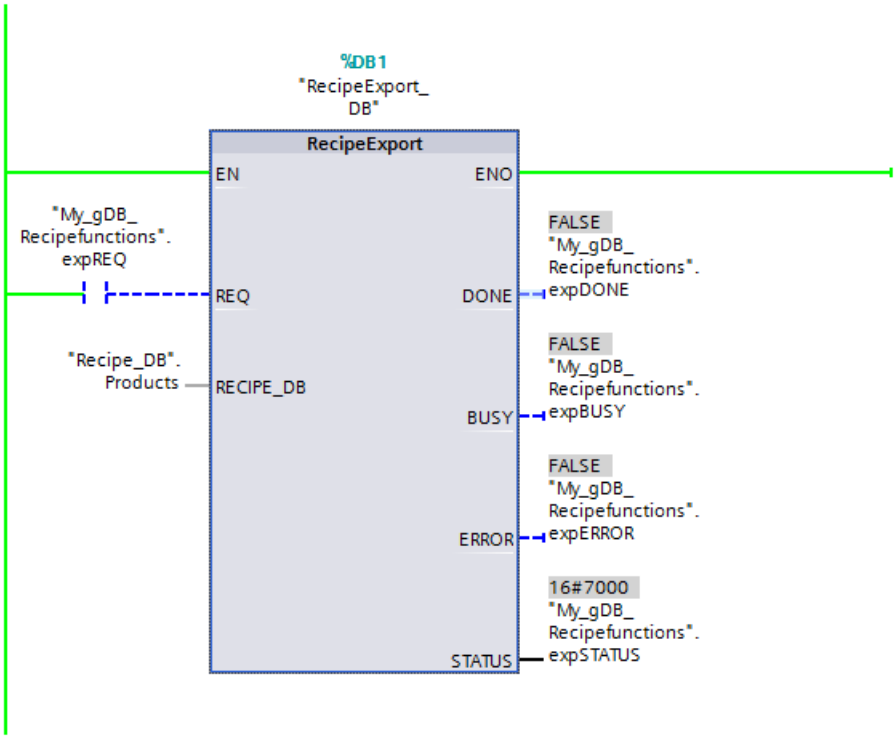

Anweisung "RecipeImport"

Am Parameter RECIPE_DB ist der zu importierende Rezept-DB verschalten. Wenn der Parameter REQ ("impREQ") den Signalzustand "TRUE" liefert, wird der Exportvorgang gestartet. In den vorhandenen Rezept-DB werden alle Rezeptdaten aus der entsprechenden CSV-Datei geladen, die sich im Rezeptordner des CPU-Speichers befindet. Am Ausgangsparameter STATUS ("impSTATUS") wird angezeigt, dass ein Signalwechsel erfolgt ist. Nachfolgend wird angezeigt, dass keine weitere Verarbeitung erfolgt (der Wert ist "7000").

Am Ausgangsparameter ERROR ("impERROR") wird angezeigt, dass die Verarbeitung der Anweisung ohne Fehler verlief.

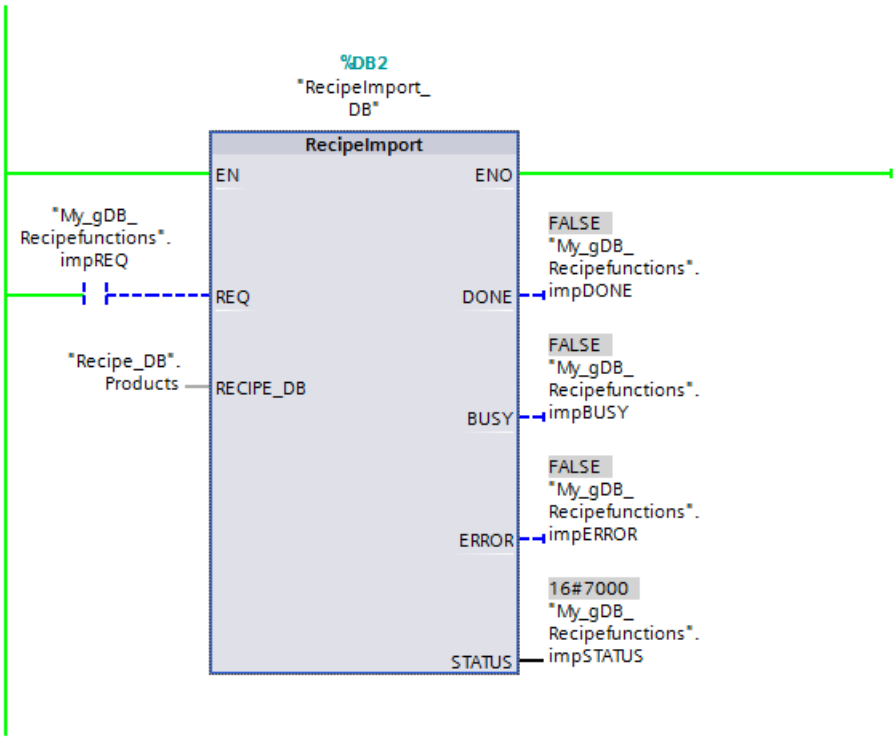

Anzeigen und Abspeichern der CSV-Datei

Webserver öffnen

Über den Internet-Browser öffnen Sie den Webserver (per IP-Adresse der CPU) und melden sich im Menü des Webservers an.

Über das Register "Filebrowser" öffnen Sie den Ordner "Rezepte" im CPU-Speicher.

Die CPU gibt die CSV-Datei aus, sobald die Anweisung "RecipeExport" ausgeführt wird. Um den jeweils aktuellen Stand der CSV-Datei angezeigt zu bekommen, klicken Sie auf die Schaltfläche "Automatisch aktualisieren". Nachfolgend wird die Funktion als "Ein" angezeigt.

CSV-Datei bearbeiten

Um die CSV-Datei zu bearbeiten, haben Sie folgende Möglichkeiten:

- Sie öffnen die CSV-Datei im Webserver.
- Sie speichern die CSV-Datei auf dem PG/PC.
  - **Hinweis**: Um die bearbeitete CSV-Datei vom PC in den CPU-Speicher laden zu können: Löschen Sie zuvor die gleichnamige CSV-Datei im CPU-Speicher.

/Recipes/

Name	Size	Changed	Delete	Rename
..				
Recipe_DB.csv	511	03:09:32 pm 10/06/2014		

  - Nach der Bearbeitung laden Sie die CSV-Datei über den Webserver in den CPU-Speicher.

Directory operations:

Browse... No file selected. Upload file

Während der Laufzeit die Komponentenwerte ändern

Um während der Laufzeit die Komponentenwerte des Rezept-DBs (z. B. Recipe_DB.Product[4]) zu ändern, verwenden Sie eine Beobachtungstabelle.

	i	Name	...	Display format	Monitor value	Modify value
1		"Recipe_DB".Products[4].productname		String	'Not_used'	'Mix'
2		"Recipe_DB".Products[4].water		DEC	0	10
3		"Recipe_DB".Products[4].barley		DEC	0	2
4		"Recipe_DB".Products[4].wheat		DEC	0	2
5		"Recipe_DB".Products[4].hops		DEC	0	4
6		"Recipe_DB".Products[4].yeast		DEC	0	30
7		"Recipe_DB".Products[4].waterTmp		DEC	0	60
8		"Recipe_DB".Products[4].mashTmp		DEC	0	60
9		"Recipe_DB".Products[4].mashTime		DEC	0	10
10		"Recipe_DB".Products[4].QTest		DEC	0	0

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Data Logging

Data Logging - Übersicht

Speichern von Prozesswerten

Die Data Logging-Anweisungen werden im Anwenderprogramm genutzt, um Prozesswerte in Data Logs zu speichern. Data Logs können auf der Memory Card (MC) oder im internen Ladespeicher gespeichert werden. Das Speichern der Data Logs erfolgt im CSV-Format (Comma Separated Value).

ACHTUNG
Besondere Zeichen in Zeichenketten Wenn Sie innerhalb des Datalogs Zeichenketten verwenden, wird dringend empfohlen, das Trennzeichen der CSV-Datei nicht innerhalb dieser Zeichenketten zu verwenden. Wenn Sie diese Empfehlung außer Acht lassen, kann das Öffnen der CSV-Datei z. B. mit MS Excel zu Problemen führen. Verwenden Sie anstatt des Trennzeichens der CSV-Datei ein anderes Zeichen, wie z. B. "-", "/" oder das Leerzeichen " ". Auch das Zeichen, welches das Ende einer Zeichenkette angibt, sowie CR und LF sind besondere Zeichen in einer Zeichenkette. Sie werden von den Anweisungen nicht korrekt behandelt und führen zu Problemen.

Die Datentypen werden gemäß den folgenden Regeln in eine Zeichenkette umgewandelt:

Datentyp		Format	Hinweis
Gleitpunkt-zahlen	LReal	+9.9999999999999E+999	Immer in Exponentialschreibweise
	Real	+9.9999999E+99	Immer in Exponentialschreibweise
Ganzzahlen mit Vorzei-chen	LInt	+9999999999999999999	Wertebereich für Ganzzahlen mit Vorzeichen: -9223372036854775808 .. +9223372036854775807
	DInt	+9999999999	Wertebereich Ganzzahlen: -2147483648 bis +2147483647
	Int	+99999	Wertebereich Ganzzahlen: -32768 bis +32767
	SInt	+999	Wertebereich Ganzzahlen: -128 bis +127
Ganzzahlen ohne Vorzei-chen	ULInt	+99999999999999999999	Wertebereich Ganzzahlen: 0 bis +18446744073709551615
	UDInt	+9999999999	Wertebereich Ganzzahlen: 0 bis +4294967295
	UInt	+99999	Wertebereich Ganzzahlen: 0 bis +65535
	USInt	+999	Wertebereich Ganzzahlen: 0 bis +255
Binärzahlen	LWord	+99999999999999999999	Wertebereich Ganzzahlen: 0 bis +18446744073709551615
	DWord	+9999999999	Wertebereich Ganzzahlen: 0 bis +4294967295
	Word	+99999	Wertebereich Ganzzahlen: 0 bis +65535
	Byte	+999	Wertebereich Ganzzahlen: 0 bis +255
	Bool	9	Wertebereich: 0 oder 1

Datentyp		Format	Hinweis
Datum und Uhrzeit	LTIME	dddddd:hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	TIME	hhh:mm:ss.999	ISO-Format mit Millisekunden
	S5TIME	hhh:mm:ss.999	ISO-Format mit Millisekunden
	LDT	YYYY-MM-DD hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	DTL	YYYY-MM-DD hh:mm:ss. 999_999_999	ISO-Format mit Millisekunden, Mikrosekunden und Nanosekunden
	DT	YYYY-MM-DD hh:mm:ss.999	ISO-Format mit Millisekunden
	DATE	YYYY-MM-DD	ISO-Format
	LTime_Of_Day	hh:mm:ss.999_999_999	ISO-Format
	TOD	hh:mm:ss.999	ISO-Format
Zeichen	WString	"abcd"	• Zeichenfolge in doppelten Anführungszeichen • Die aktuelle Länge wird mit Leerzeichen bis zur maximalen Länge aufgefüllt. • Eine Zeichenkette vom Datentyp WString besteht aus Elementen des Datentyps WChar. Beim Speichern eines Data Logs werden die Inhalte von Zeichen des Datentyps WChar auf 16#FF beschränkt (Bsp.: 16#1255 wird zu 16#55 geändert.).
	String	"abcd"	• Zeichenfolge in doppelten Anführungszeichen • Die aktuelle Länge wird mit Leerzeichen bis zur maximalen Länge aufgefüllt.
	WChar	"a"	Einzelnes Zeichen in doppelten Anführungszeichen. Beim Speichern eines Data Logs werden die Inhalte von Zeichen des Datentyps WChar auf 16#FF beschränkt (Bsp.: 16#1255 wird zu 16#55 geändert.).
	Char	"a"	Einzelnes Zeichen in doppelten Anführungszeichen

Die Data Logging-Anweisungen verwenden Sie in Ihrem Programm zum Erstellen oder Öffnen eines Data Logs, zum Schreiben eines Eintrags und zum Schließen der Data Log-Datei.

⚠ WARNUNG

Laufzeit der Data Logging-Anweisungen

Die nachfolgend angegebenen Bedingungen verlangsamen die Data Logging-Anweisungen. Dieser Effekt verstärkt sich mit zunehmender Größe und Anzahl der Data Log-Dateien.

- Große Dateigröße der Data Log-Datei
- Große Anzahl der Data Log-Dateien auf der Memory Card
- Gleichzeitiger Zugriff auf eine Data Log-Datei durch den Webserver

Die Laufzeit einiger Data Logging-Anweisungen wird von der Größe und Anzahl der Data Logs typischerweise nicht beeinträchtigt (z. B. "DataLogWrite"). Die Anweisungen "DataLogCreate", "DataLogNewFile", "DataLogClear" und "DataLogOpen" hingegen zeigen das oben beschriebene Verhalten.

Hinweis

Data Logging-Anweisungen, bei den Daten übertragen werden

Wegen der asynchronen Bearbeitung der Data Logging-Anweisungen müssen Sie die Daten im Quellbereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

Welche Programmwerte im Data Log beim Erstellen des Datenpuffers abgelegt werden entscheiden Sie durch das Erstellen des Datenpuffers. Der Datenpuffer wird als Speicher für neue Data Log-Einträge genutzt. Neue Werte müssen vor dem Aufruf von "DataLogWrite (Seite 2793)" in den Puffer geschrieben werden. Bei der Ausführung der Anweisung "DataLogWrite (Seite 2793)" werden die Daten vom Puffer in einen Data Log-Datensatz geschrieben.

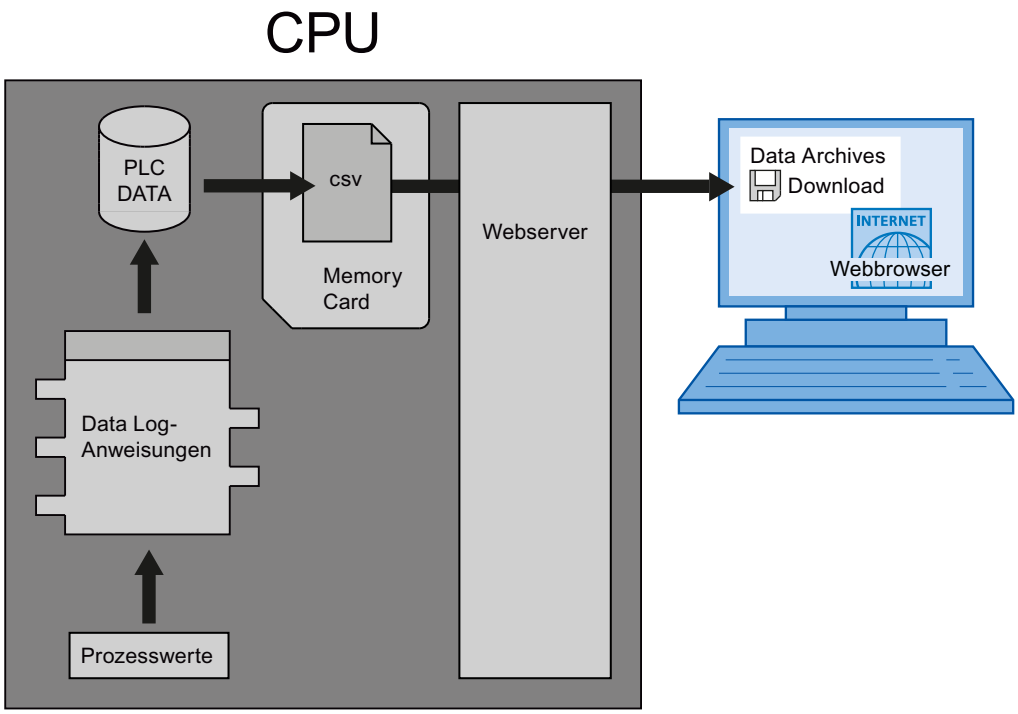

Data Log-Dateien können folgendermaßen in den PC kopiert werden:

- Wenn die PROFINET-Schnittstelle mit dem PC verbunden ist, benutzen Sie einen Webbrowser, um auf die Data Logs über den Webserver zuzugreifen. Die CPU kann dabei im "RUN"- oder im "STOP"-Betrieb sein. Wenn die CPU in der Betriebsart "RUN" ist, läuft das Programm weiter, während der Webserver einen Datentransfer durchführt.
- Befindet sich in der CPU eine Memory Card, dann können Sie diese Karte herausnehmen und diese in einen Standardeinschub für SD-Cards (Secure Digital) oder MMC-Cards (MulitMediaCard) an einem PC oder Programmiergerät stecken. Nutzen Sie den Dateimanager, um die Data Log-Dateien von der Memory Card in den PC zu übertragen. Wenn Sie die Memory Card herausnehmen, geht die CPU in "STOP".

Hinweis

Lebensdauer von Memory Cards

Die Lebensdauer von Memory Cards ist begrenzt. Nähere Informationen finden Sie hier: AUTOHOTSPOT

Eigenschaften des Data Logs

Das Schreiben der Datensätze eines Data Logs erfolgt nach dem Prinzip eines Ringpuffers. Neue Datensätze werden angehängt, bis die maximale Anzahl Datensätze erreicht ist (Parameter RECORD). Der nächste Datensatz überschreibt dann den "ältesten" Datensatz des Data Logs.

Wenn Sie verhindern wollen, dass Datensätze überschrieben werden, dann nutzen Sie die Anweisung "DataLogNewFile (Seite 2799)", um eine neue Data Log-Datei basierend auf dem aktuellen Data Log zu erstellen. Neue Datensätze werden dann in den neuen Data Log geschrieben.

Erstellen von Data Logs

Mit der Anweisung "DataLogCreate (Seite 2779)" erzeugen Sie eine neue Data Log-Datei im Verzeichnis "\DataLogs" des Ladespeichers.

- Der am Parameter NAME vergebene Name ist die Benennung für das Data Log. Dieser wird auch als Dateiname für die CSV-Datei verwendet. Die Datei wird im Verzeichnis "DataLogs" abgelegt.
- Der Bausteinparameter DATA bestimmt den Datenpuffer für das neue Data Log-Objekt und die Spalten und Datentypen in dem Data Log. Die Spalten und Datentypen eines Datensatzes in dem Data Log entstehen durch die Elemente der Strukturdeklaration oder der Arraydeklaration dieses Datenpuffers. Jedes Element einer Struktur oder eines Arrays entspricht einer Spalte in einer Zeile im Data Log.
- Mithilfe des Bausteinparameters HEADER weisen Sie den Spalten jeweils eine Überschrift in der Kopfzeile zu.
- Die Anweisung "DataLogCreate (Seite 2779)" gibt eine ID zurück. Diese ID wird von den anderen Data Logging -Anweisungen als Referenz für das erzeugte Data Log verwendet.

Öffnen von Data Logs

Über die Anweisung "DataLogOpen (S7-1200 und S7-1500) öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Ein Data Log muss geöffnet sein, um neue Datensätze in das Data Log zu schreiben.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2779)" und "DataLogNewFile (Seite 2799)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder den Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und als ID 0 angeben, wird die zum Data Log gehörende ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden.

Für "DataLogOpen" (S7-1500) ist eine Konsistenzprüfung möglich.

Schreiben in Data Logs

Voraussetzung, um in einen Data Log einen Datensatz zu schreiben, ist ein geöffnetes Data Log (Anweisung "DataLogOpen (Seite 2787)"). Die Anweisung "DataLogWrite (Seite 2793)" schreibt einen Datensatz in das Data Log.

Hinweis

Verändern des Inhalts von Data Log-Dateien

Da die Größe von Data Log-Dateien enorm sein kann (500.000 kByte), kann der Inhalt dieser Dateien nicht überprüft werden, nicht beim Einschalten und nicht vor der Ausführung von "DataLogWrite". Sie dürfen den Inhalt außerhalb der CPU nicht verändern, da dies zu einer beschädigten Datei führen kann.

Schließen von Data Logs

Mit der Anweisung "DataLogClose (Seite 2795)" schließen Sie ein geöffnetes Data Log. Das Data Log wählen Sie über den Parameter ID aus.

Beim Übergang der CPU in STOP oder bei einem Neustart wird das Data Log automatisch geschlossen.

Löschen von Data Logs

Mit der Anweisung "DataLogDelete (Seite 2797)" (S7-1500) löschen Sie eine Data Log-Datei auf der Memory Card. Das Data Log und die enthaltenen Datensätze können nur gelöscht werden, wenn es über die Anweisung "DataLogCreate (Seite 2779)" erstellt wurde.

Über die Parameter NAME und ID wählen Sie das zu löschende Data Log aus. Der Parameter ID wird zuerst ausgewertet. Ist ein Data Log mit der entsprechenden ID vorhanden, wird der

Parameter NAME nicht ausgewertet. Wenn am Parameter ID der Wert "0" verwendet wird, muss am Parameter NAME ein Wert mit dem Datentyp STRING verwendet werden.

Hinweis

Löschen von Data Logs auf der Memory Card

Sie können Data Logs von einer Memory Card mit Hilfe eines PC-Kartenleser kopieren, aber Sie dürfen sie über einen PC-Kartenleser nicht ändern oder löschen.

Zusätzlich zu den Data Log-Anweisungen wird die Benutzung des Webservers empfohlen, um Data Logs anzuzeigen, herunterzuladen (zu kopieren) und zu löschen.

Das direkte Aufrufen des Dateisystems der Memory Card im Datei-Explorer birgt das Risiko, dass Sie versehentlich Data Log- oder andere Systemdateien löschen oder ändern, so dass möglicherweise eine Datei beschädigt oder die Memory Card unbrauchbar wird. Änderungen von Data Logs auf der Memory Card, die Sie außerhalb der PLC vorgenommen haben, bewirken, dass die NETZ EIN-Zeit der PLC vergrößert wird.

Leeren von Data Logs

Mit der Anweisung "DataLogClear (Seite 2791)" (S7-1500) löschen Sie alle Datensätze eines vorhandenen Data Logs. Nicht gelöscht wird die optionale Kopfzeile der CSV-Datei (siehe Beschreibung des Parameters HEADER der Anweisung "DataLogCreate (Seite 2779)").

Über den Parameter ID wählen Sie das Data Log aus, deren Datensätze gelöscht werden sollen. Um die Datensätze zu löschen, muss das Data Log geöffnet sein.

Neue Datei für Data Logs

Über die Anweisung "DataLogNewFile (Seite 2799)" (S7-1200) und "DataLogTypedNewFile (Seite 2801)" (S7-1500) erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

Für "DataLogTypedNewFile (Seite 2801)" (S7-1500) ist eine Konsistenzprüfung möglich.

DataLogCreate: Data Log erstellen

Beschreibung

Mit der Anweisung "DataLogCreate" erstellen Sie ein Data Log.

Das Data Log wird auf der Memory Card bzw. im internen Ladespeicher im Verzeichnis "\DataLogs" gespeichert. Die Datenmenge, die in einem Data Log abgelegt wird, ist dabei abhängig von dem zur Verfügung stehenden Speicherplatz auf der Memory Card bzw. des Speicherplatzes im internen Ladespeicher der eingesetzten CPU. Die maximale Größe der erzeugten CSV-Datei beträgt bei S7-1200-CPUs 500.000.000 Byte, bei S7-1500-CPUs 1.000.000.000 Byte.

Die maximale Anzahl der Datensätze, die in einem Data Log abgelegt wird, geben Sie am Parameter RECORDS vor. Wenn die vorgegebene maximale Anzahl von Datensätzen in dem Data Log erreicht wird, wird der jeweils älteste Datensatz überschrieben. Um ein Überschreiben der vorhandenen Datensätze zu vermeiden, können Sie die Anweisung "DataLogNewFile (Seite 2799)" nutzen. Die Anweisung wird bei Erreichen der am Parameter RECORDS vorgegebenen Anzahl (Returnwert 1 am Parameter STATUS der Anweisung "DataLogWrite (Seite 2793)") zum Erstellen eines neues Data Log mit der gleichen Struktur genutzt. Die Datensätze werden danach in dem neuen Data Log gespeichert.

Den Namen für das Data Log vergeben Sie am Parameter NAME. Das Data Log wird im CSV-Format (Comma Separated Values) erstellt. Über den Parameter HEADER erstellen Sie eine (optionale) Kopfzeile für das Data Log.

Nach der Erstellung wird das Data Log automatisch geöffnet. Das heißt, es können Daten geschrieben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogCreate":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführung der Anweisung Bei einer steigenden Flanke am Parameter REQ wird das Data Log erstellt.
RECORDS	Input	UDInt	E, A, M, L, D oder Konstante	Maximale Anzahl der Datensätze im Data Log Wenn über die Anweisung "DataLogWrite (Seite 2793)" mehr Datensätze geschrieben werden als an diesem Parameter vorgegeben, wird der älteste Datensatz überschrieben.
FORMAT	Input	UInt	E, A, M, L, D oder Konstante	Datenformat: • 0: Intern (wird nicht unterstützt) • 1: CSV (Comma separated values)
TIMESTAMP	Input	UInt	E, A, M, L, D oder Konstante	Zeitstempelung; Details siehe unten Bei aktivierter Zeitstempelung werden die zusätzlichen Spalten im Header automatisch hinzugefügt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NAME	Input	VARIANT	L, D	Name des Data Logs Der vergebene Name wird auch als Dateiname der CSV-Datei verwendet. Bei S7-1200-CPUs bestehen für den Namen des Data Logs die folgenden Einschränkungen: • Der Name darf nicht länger als 35 Zeichen sein. • Erlaubt sind alle ASCII-Zeichen zwischen 0x20 und 0x7E mit Ausnahme von \ ' / " : ; [ ] | = . * ? < > Bei S7-1500-CPUs bestehen für den Namen des Data Logs die folgenden Einschränkungen: • Der Name darf nicht länger als 55 Zeichen sein. • Erlaubt sind die folgenden Zeichen: 0 ... 9, a ... z, A ... Z sowie - und _
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs (nur Output) Die ID des Data Logs wird für die weiteren Data Logging -Anweisungen benötigt, um das erstellte Data Log zu adressieren.
HEADER	InOut	VARIANT	D	Kopfzeile des Data Logs (optional) Der Parameter ist nach dem Einfügen der Anweisung ausgeblendet. In der CSV-Datei wird der Header als erste Zeile geschrieben.
DATA	InOut	VARIANT	D	Zeiger auf die Datenstruktur, die beim Ausführen der Anweisung "DataLogWrite (Seite 2793)" als Datensatz geschrieben werden.
DONE	Output	BOOL	E, A, M, L, D	Zustandsparameter: • 0: Bearbeitung noch nicht abgeschlossen • 1: Bearbeitung der Anweisung erfolgreich beendet.
BUSY	Output	BOOL	E, A, M, L, D	Zustandsparameter: • 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen. • 1: Bearbeitung der Anweisung läuft.
ERROR	Output	BOOL	E, A, M, L, D	Zustandsparameter: • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Detaillierte Statusinformation: Am Parameter STATUS werden detaillierte Fehler- und Statusinformationen ausgegeben. Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter HEADER

Der Parameter HEADER ist ein VARIANT-Pointer auf einen Datenbaustein, der eine Kopfzeile für die CSV-Datei (Header) definiert. Die Kopfzeile ist in der Darstellung einer CSV-Datei jeweils die erste Zeile.

- Die durch HEADER vorgegebene Kopfzeile wird in die CSV-Datei übertragen. Dabei wird nicht überprüft, ob sie zum restlichen Data Log konsistent ist. Insbesondere wird nicht überprüft, ob die Anzahl der Spalten, die durch Kommas getrennt sind, mit der Anzahl der Datenspalten übereinstimmt.
- Als Datentyp kann für die einzelne Spaltenbenennung ein STRING, Array of BYTE oder Array of CHAR verwendet werden. Bei Verwendung von STRING ist die Länge auf 254 Bytes beschränkt. Bei Verwendung von Array of BYTE und Array of CHAR sind 5000 Elemente möglich.

Wenn Sie am Parameter HEADER keinen Wert angeben oder HEADER im Ladespeicher liegt, wird keine Kopfzeile erstellt.

Bei den Werten 1, 3 und 5 von TIMESTAMP wird der folgende Text am Anfang der Kopfzeile eingefügt: "Record, Date, UTC Time,".

Bei den Werten 2 und 4 von TIMESTAMP wird der folgende Text am Anfang der Kopfzeile eingefügt: "Record, Date, Loc Time,".

Parameter TIMESTAMP

Die folgende Tabelle zeigt alle möglichen Werte von TIMESTAMP.

Wert	Format von DATE_AND_TIME	Zeitquelle	Bemerkung
0	-	keine Zeitstempelung	obligatorisch
1	• S7 1200: mm/dd/yyyy, hh:mm:ss • S7 1500: yyyy-mm-dd, hh:mm:ss.999	Systemzeit	obligatorisch
2	• S7-1200-CPUs: mm/dd/yyyy, hh:mm:ss • S7-1500-CPUs ab Firmwarestand V2.6: mm/dd/yyyy, hh:mm:ss[.999] (24-Stunden-Uhrzeit, [Millisekunden optional])	Lokalzeit	optional

Wert	Format von DATE_AND_TIME	Zeitquelle	Bemerkung
3	S7-1500-CPUs ab Firmwarestand V2.6: mm/dd/yyyy, hh:mm:ss[.999] (24-Stunden-Uhrzeit, [Millisekunden optional])	Systemzeit	optional
4	S7-1500-CPUs ab Firmwarestand V2.6: yyyy-mm-dd, hh:mm:ss.[999] (24-Stunden-Uhrzeit, [Millisekunden optional])	Lokalzeit	optional
5	S7-1500-CPUs ab Firmwarestand V2.6: yyyy-mm-dd, hh:mm:ss.[999] (24-Stunden-Uhrzeit, [Millisekunden optional])	Systemzeit	optional

Parameter DATA

Der Parameter DATA ist ein VARIANT-Pointer auf einen elementaren Datentyp, eine Struktur oder ein Array in einem Datenbaustein. Ein Element einer Struktur oder eines Arrays entspricht einer Spalte im Data Log mit einem bestimmten Datentyp.

Beachten Sie bei der Erstellung des Datenbausteins Folgendes:

- Üblicherweise entspricht die Anzahl der Spalten der Spaltenzahl, die am Parameter HEADER definiert ist. Es ist jedoch nicht vorgeschrieben.
- Jedes Element der Struktur oder des Arrays entspricht einem Spalteneintrag in der CSV-Datei. Bei Verwendung des Datentyps STRUCT dürfen daher keine Struktur-Verschachtelungen (STRUCT in STRUCT) verwendet werden. Erlaubt ist jedoch die Verwendung von Arrays innerhalb eines Datentyps STRUCT mit folgender Ausnahme: Arrays, deren Elemente vom Datentyp "PLC-Datentyp (UDT)" sind, sind nicht erlaubt.
- Die Datenstruktur kann bis zu 256 Elemente enthalten. Bei mehr als 256 Elementen, wird am Parameter STATUS der Fehlercode 8C52 ausgegeben.
- Die Variablen des Datenbausteins können als remanent oder nicht remanent eingestellt sein. Die Remanenzeinstellung für alle Variablen des Datenbausteins muss jedoch gleich sein.

Parameter STATUS (S7-1200)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogCreate-Anweisungen wurde bereits erreicht.

Fehlercode* (W#16#...)	Beschreibung
8090	Unzulässiger Dateiname (siehe Beschreibung des Parameters NAME).
8093	Data Log existiert bereits.
8097	Länge der Datei überschreitet Begrenzung des Dateisystems oder CPU-spezifische Begrenzung.
80A2	Schreibfehler vom Dateisystem zurückgegeben.
80B2	Maximale Anzahl von DataLogs überschritten
80B3	Speicherplatz auf der Memory Card nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich.
80C1	Zu viele Data Logs geöffnet.
8253	Ungültiger Wert am Parameter RECORDS.
8353	Ungültige Formatauswahl
8453	Ungültige Zeitstempelung
8B24	Zuweisung am Parameter HEADER ungültig (zum Beispiel: Zeigt auf lokalen Speicher).
8B51	Unzulässiger Datentyp am Parameter HEADER oder Länge überschreitet die maximale Größe.
8C24	Zuweisung am Parameter DATA ungültig (zum Beispiel: Zeigt auf lokalen Speicher).
8C51	Unzulässiger Datentyp am Parameter DATA / Datenstruktur kann nicht verwendet werden.
8C52	Struktur am Parameter DATA enthält mehr als 256 Elemente.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (S7-1500)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
8070	Die maximale Anzahl gleichzeitig aktiver DataLogCreate-Anweisungen wurde bereits erreicht.
8090	Unzulässiger Dateiname (siehe Beschreibung des Parameters NAME).
8091	Parameter "NAME" ist kein String.
8093	Data Log existiert bereits.
8097	Länge der Datei überschreitet Begrenzung des Dateisystems oder CPU-spezifische Begrenzung.
80A2	Schreibfehler vom Dateisystem zurückgegeben.
80B2	Maximale Anzahl von DataLogs überschritten
80B3	Speicherplatz auf der Memory Card nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich.
80C1	Zu viele Data Logs geöffnet.
8253	Ungültiger Wert am Parameter RECORDS.
8353	Ungültige Formatauswahl.
8453	Ungültige Zeitstempelung.
8B24	Zuweisung am Parameter HEADER ungültig (zum Beispiel: Zeigt auf lokalen Speicher).

Fehlercode* (W#16#...)	Beschreibung
8B51	Unzulässiger Datentyp am Parameter HEADER oder Länge überschreitet die maximale Größe.
8C24	Zuweisung am Parameter DATA ungültig (zum Beispiel: Zeigt auf lokalen Speicher).
8C51	Unzulässiger Datentyp am Parameter DATA / Datenstruktur kann nicht verwendet werden.
8C52	Struktur am Parameter DATA enthält mehr als 256 Elemente.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel wird ein einfaches Data Log mit einer Zeitstempelung und drei Prozesswerten erstellt.

Variablen im globalen Datenbaustein

Die Werte für die Eingangsparameter des Data Log sind in dem globalen Datenbaustein "DataLogDB" hinterlegt:

- DataLogName (String): Die Variable enthält den Namen des Data Log, der auch als Dateiname für die CSV-Datei verwendet wird.
- DataLogID (DInt): In diese Variable wird bei Aufruf der Anweisung die ID des Data Log geschrieben.
  - Die ID wird durch die Anweisung automatisch vergeben.
  - Die Variable "DataLogID" verwenden Sie in weiteren DataLog-Anweisungen, um dieses Data Log zu adressieren.
- MyHeader (String): Die Variable enthält die Kopfzeile des DataLogs, d. h. die Spaltenüberschriften für die Prozesswerte.
- MyData (Struct): Die Variable enthält die drei Prozesswerte, die in das Data Log geschrieben werden. Jedes Mal, wenn ein Datensatz geschrieben wird (Anweisung "DataLogWrite (Seite 2793)"), werden die aktuellen Werte in einen neuen Datensatz geschrieben.

DataLogDB

	Name	Datentyp	Startwert	Remanenz
1	Static			
2	DataLogName	String	'MyDataLog'	
3	DataLogID	DInt	0	
4	MyHeader	String	'Value1,Value2,Value3'	
5	MyData	Struct		
6	ProcessValue1	Int	2	
7	ProcessValue2	Int	3	
8	ProcessValue3	Int	4	

Aufruf der Anweisung "DataLogCreate"

Die Anweisung wird mit den folgenden Eingangs-Parametern aufgerufen:

- REQ (BOOL): Bei REQ = "TRUE" wird das Data Log erstellt.
- RECORD (3): In das Data Log können maximal drei Datensätze geschrieben werden. Danach wird der älteste Datensatz überschrieben.
- FORMAT (1): Das Data Log wird als CSV-Datei erstellt.
- TIMESTAMP (1): Aktiviert. Für das Data Log werden automatisch zwei zusätzliche Spalten (Datum und Uhrzeit) erstellt. Der aktuelle Zeitstempel wird bei jeder Ausführung von "DataLogWrite (Seite 2793)" in den Datensatz geschrieben.
- NAME (VARIANT): Zeiger auf die Variable "DataLogName" im Datenbaustein "DataLogDB".
- ID (DWORD): Zeiger auf die Variable "DataLogID" im Datenbaustein "DataLogDB" (nur Output).
- HEADER (VARIANT): Zeiger auf die Variable "MyHeader" im Datenbaustein "DataLogDB".
- DATA (VARIANT): Zeiger auf die Variable "MyData" im Datenbaustein "DataLogDB".

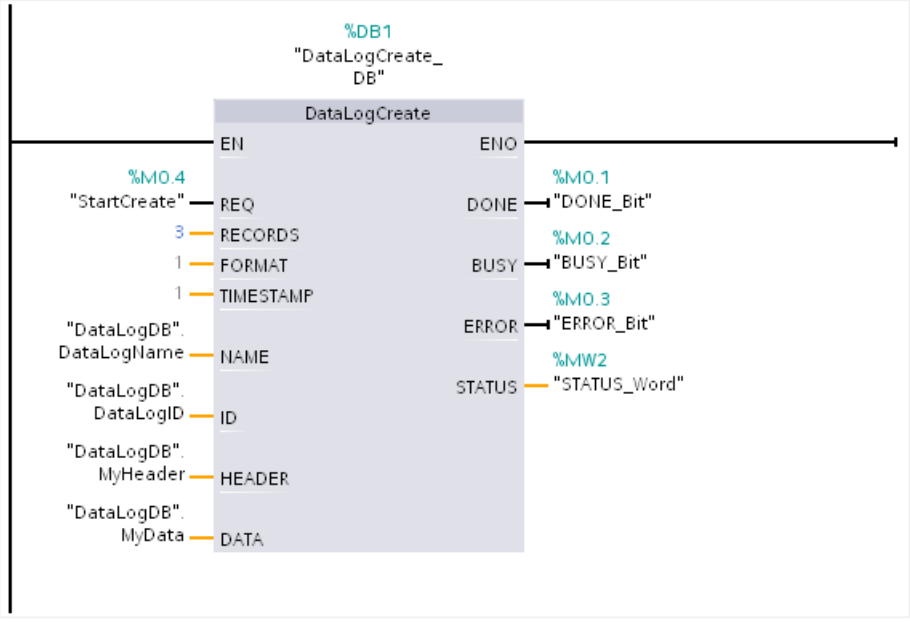

Auslesen des Data Log über den Webserver

Über den Webserver lesen Sie das erstelle Data Log aus:

- Aktivieren Sie den Webserver in den Eigenschaften der CPU. Um das Data Log auszulesen, muss die Berechtigung "...Dateien zu lesen" für den Webserver aktiviert sein.
- Über den Internet.-Browser können Sie auf den Webserver zugreifen, indem Sie die IP-Adresse der CPU als URL eingeben.

- Unter "Filebrowser" wurde automatisch das Verzeichnis "\DataLogs" erstellt, welches das Data Log enthält. Alternativ können Sie auch im Menü unter "DataLogs" nachschauen.
- Wurde die Anweisung "DataLogWrite (Seite 2793)" noch nicht ausgeführt, enthält das DataLog nur den Eintrag "//END". Nach erstmaliger Ausführung von "DataLogWrite (Seite 2793)" wird der erste Datensatz geschrieben.

	A	B	C	D	E	F
1	SeqNo	Date	Time	ProcessValue1	ProcessValue2	ProcessValue3
2	1	01.07.2013	12:33:42.917	2	3	4
3						
4						
5						

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Data Logging - Übersicht (Seite 2774)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogOpen: Data Log öffnen

DataLogOpen: Data Log öffnen

Beschreibung

Über die Anweisung "DataLogOpen" öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Um neue Datensätze in das Data Log zu schreiben, muss ein Data Log geöffnet sein.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2779)" und "DataLogNewFile (Seite 2799)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder den Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und als ID 0 angeben, wird die zum Data Log gehörende ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogOpen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, L oder Konstante	Ausführen der Anweisung bei steigender Flanke.
MODE	Input	UInt	E, A, M, L, D oder Konstante	Modus zum Öffnen des Data Logs: • MODE= "0" Datensätze des Data Logs bleiben erhalten • MODE= "1" Datensätze des Data Logs werden gelöscht, die Kopfzeile bleibt erhalten
NAME	Input	VARIANT	L, D	(Datei-)name des Data Logs.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs.
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht beendet.
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
2	Warnung: Data Log-Datei wurde bereits durch diese Anwendung geöffnet.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogOpen-Anweisungen wurde bereits erreicht.
8090	Zwischen der Data Log-Definition und bestehenden Data Log-Daten bestehen Inkonsistenzen.
8091	Am Parameter NAME wurde ein anderer Datentyp als String verwendet.
8092	Data Log existiert nicht.
80B4	Memory Card oder Data Log-Datei ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich. Dies kann z. B. beim Laden im RUN der Fall sein.

Fehlercode* (W#16#...)	Beschreibung
80C1	Zu viele Dateien geöffnet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Data Logging - Übersicht (Seite 2774)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogOpen: Data Log öffnen

Beschreibung

Über die Anweisung "DataLogOpen" öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Ein Data Log muss geöffnet sein, um neue Datensätze in das Data Log schreiben zu können.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2779)" und "DataLogNewFile (Seite 2799)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder den Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und als ID 0 angeben, wird die zum Data Log gehörende ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden.

Der Parameter DATA ermöglicht eine Konsistenzprüfung zwischen dem zu öffnenden Data Log und der Definition des Data Logs der Anweisung "DataLogCreate (Seite 2779)". Es wird überprüft, ob der Datentyp des Parameters DATA identisch zum Datentyp bei "DataLogCreate" ist. Wenn dies nicht der Fall ist, wird am Parameter STATUS der Fehlercode W#16#8090 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogOpen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführen der Anweisung bei steigender Flanke.
MODE	Input	UInt	E, A, M, L, D oder Konstante	Modus zum Öffnen des Data Logs: • MODE= "0" Datensätze des Data Logs bleiben erhalten • MODE= "1" Datensätze des Data Logs werden gelöscht, die Kopfzeile bleibt erhalten
NAME	Input	VARIANT	L, D	(Datei-)name des Data Logs.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs.
DATA	InOut	VARIANT	D	Bei Konsistenzprüfung: Zeiger auf den Datenbereich am Parameter DATA der Anweisung "DataLogCreate (Seite 2779)".
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht beendet.
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
2	Warnung: Data Log-Datei wurde bereits durch diese Anwendung geöffnet.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogOpen-Anweisungen wurde bereits erreicht.

Fehlercode* (W#16#...)	Beschreibung
8090	Datentypen inkonsistent. Das Data Log am Parameter ID verwendet andere Datentypen als am Parameter DATA angegeben.
8091	Am Parameter NAME wurde ein anderer Datentyp als String verwendet.
8092	Data Log existiert nicht.
80B4	Memory Card oder Data Log-Datei ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich. Dies kann z. B. beim Laden im RUN der Fall sein.
80C1	Zu viele Dateien geöffnet.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Data Logging - Übersicht (Seite 2774)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogClear: Data Log leeren

Beschreibung

Mit der Anweisung "DataLogClear" löschen Sie alle Datensätze eines vorhandenen Data Logs. Nicht gelöscht wird die optionale Kopfzeile der CSV-Datei (siehe Beschreibung des Parameters HEADER der Anweisung "DataLogCreate (Seite 2779)").

Über den Parameter ID wählen Sie das Data Log aus, dessen Datensätze gelöscht werden.

Voraussetzung

Um die Datensätze zu löschen, muss das Data Log geöffnet sein (siehe Anweisung "DataLogOpen (Seite 2787)").

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogClear":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführung der Anweisung bei steigender Flanke.
ID	InOut	DWORD	E, A, M, D, L	Objekt-ID des Data Logs
DONE	Output	BOOL	E, A, M, D, L	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	Ausführung der Anweisung noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden an dem Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogClear-Anweisungen wurde bereits erreicht.
8080	Die über den Parameter ID ausgewählte Data Log-Datei wird nicht durch die Anweisung "DataLogClear" bearbeitet.
8092	Data Log existiert nicht.
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B0	Data Log ist nicht geöffnet.
80B4	Memory-Card ist schreibgeschützt.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Data Logging - Übersicht (Seite 2774)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogWrite: Data Log schreiben

Beschreibung

Mit der Anweisung "DataLogWrite" schreiben Sie einen Datensatz in ein vorhandenes Data Log. Über den Parameter ID wählen Sie das Data Log aus, in welches der Datensatz geschrieben werden soll. Um einen neuen Datensatz zu erstellen, muss das Data Log geöffnet sein. Die Anweisung erstellt einen neuen Datensatz in dem Format, das am Parameter DATA bei der Erstellung des Data Logs angegeben wurde.

Übertragen Sie vor dem Aufruf der Anweisung "DataLogWrite" die Daten an die Variable, welche Sie am Parameter DATA der Anweisung "DataLogCreate" verschaltet haben. Bei der Ausführung der Anweisung "DataLogWrite" werden die übertragenen Daten in das Data Log kopiert.

ACHTUNG
Data Log-Datenverlust bei Unterbrechung der Stromversorgung der CPU Wenn die Stromversorgung während der Ausführung der Anweisung "DataLogWrite" unterbrochen wird, geht der zu übertragende Datensatz verloren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogWrite":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführung der Anweisung bei steigender Flanke.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS (S7-1200)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler
0001	Letzter möglicher Datensatz am Ende der Datei erstellt. Beim Erstellen eines weiteren Datensatzes wird ein älterer überschrieben.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogWrite-Anweisungen wurde bereits erreicht.
8090	Data-Log-Definition stimmt nicht mit dem vorhandenen Data Log überein.
8092	Data Log existiert nicht.
8093	Die Quelladresse für das Datalog wurde geändert (z. B. durch Laden).
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B0	Data Log ist nicht geöffnet.
80B3	Speicherplatz auf der Memory Card nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (S7-1500)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler
0001	Letzter möglicher Datensatz am Ende der Datei erstellt. Beim Erstellen eines weiteren Datensatzes wird ein Älterer überschrieben.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
8070	Die maximale Anzahl gleichzeitig aktiver DataLogWrite-Anweisungen wurde bereits erreicht.

Fehlercode* (W#16#...)	Beschreibung
8090	Data-Log-Definition stimmt nicht mit dem vorhandenen Data Log überein.
8092	Data Log existiert nicht.
8093	Die Quelladresse für das Datalog wurde geändert (z. B. durch Laden).
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B0	Data Log ist nicht geöffnet.
80B3	Speicherplatz auf der Memory Card nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Data Logging - Übersicht (Seite 2774)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogClose: Data Log schließen

Beschreibung

Mit der Anweisung "DataLogClose" schließen Sie ein geöffnetes Data Log. Das Data Log wählen Sie über den Parameter ID aus.

Hinweis

Automatisches Schließen von Data Logs

Beim Übergang der CPU in STOP oder einem Neustart wird das Data Log automatisch geschlossen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogClose":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Funktion bei steigender Flanke ausführen.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS (S7-1200)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler
1	Data Log nicht geöffnet
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogClose-Anweisungen wurde bereits erreicht.
8092	Data Log existiert nicht.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (S7-1500)

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler
1	Data Log nicht geöffnet
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
8070	Die maximale Anzahl gleichzeitig aktiver DataLogClose-Anweisungen wurde bereits erreicht.
8092	Data Log existiert nicht.
80B4	Memory Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogDelete: Data Log löschen

Beschreibung

Mit der Anweisung "DataLogDelete" löschen Sie eine Data Log-Datei auf der Memory Card. Wenn es über die Anweisungen "DataLogCreate" oder "DataLogNewFile" erstellt wurde, werden das Data Log und die enthaltenen Datensätze gelöscht.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogDelete":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführung der Anweisung bei steigender Flanke.
NAME	Input	VARIANT	L, D	Dateiname des Data Logs
DELFILE	Input	BOOL	E, A, M, D, L oder Konstante	• 0: Data Log bleibt erhalten. • 1: Data Log wird gelöscht.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	InOut	DWORD	E, A, M, D, L	Objekt-ID des Data Logs
DONE	Output	BOOL	E, A, M, D, L	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	Löschen des Data Logs noch nicht abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden an dem Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter NAME und ID

Über die Parameter NAME und ID wählen Sie das zu löschende Data Log aus. Der Parameter ID wird zuerst ausgewertet. Ist ein Data Log mit der entsprechenden ID vorhanden, wird der Parameter NAME nicht ausgewertet. Wird am Parameter ID der Wert "0" verwendet, muss am Parameter NAME ein Wert mit dem Datentyp STRING verwendet werden.

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogDelete-Anweisungen wurde bereits erreicht.
8091	Am Parameter NAME wird ein anderer Datentyp als STRING verwendet.
8092	Data Log existiert nicht.
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B4	Memory-Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DataLogCreate: Data Log erstellen (Seite 2779)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogNewFile: Data Log in neuer Datei

DataLogNewFile: Data Log in neuer Datei

Beschreibung

Über die Anweisung "DataLogNewFile" erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

Nach der Erstellung wird das neue Data Log automatisch geöffnet. Das heißt, es können Daten geschrieben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogNewFile":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D oder Konstante	Ausführung der Anweisung bei steigender Flanke.
RECORDS	Input	UDInt	E, A, M, L, D oder Konstante	Anzahl der Datensätze im neuen Data Log.
NAME	Input	VARIANT	L, D	Dateiname des neuen Data Logs.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs • In: ID des bereits vorhandenen Data Logs • Out: ID des neuen Data Logs
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogNewFile-Anweisungen wurde bereits erreicht.
8090	Ungültiger Dateiname.
8091	Datentyp am Parameter NAME ist kein STRING.
8092	Quellen- Data Log existiert nicht.
8093	Neues Data Log existiert bereits.
8097	Länge der Datei überschreitet Begrenzung des Dateisystems.
80A0	Datentypen inkonsistent. Das Data Log am Parameter ID verwendet andere Datentypen als am Parameter DATA angegeben.
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B2	Maximale Anzahl von DataLogs überschritten oder DataLog-Ressourcen aufgebraucht
80B3	Ladespeicher nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich
80C1	Zu viele Dateien geöffnet.
8253	Ungültiger Wert am Parameter RECORDS.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

DataLogNewFile: Data Log in neuer Datei

Beschreibung

Über die Anweisung "DataLogNewFile" erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

Der Parameter DATA ermöglicht eine Konsistenzprüfung zwischen dem neu zu erstellenden Data Log und der Definition des Data Logs der Anweisung "DataLogCreate (Seite 2779)". Es wird überprüft, ob der Datentyp des Parameters DATA identisch zum Datentyp bei "DataLogCreate" ist. Wenn dies nicht der Fall ist, wird am Parameter STATUS der Fehlercode W#16#80A0 ausgegeben.

Nach der Erstellung wird das neue Data Log automatisch geöffnet. Das heißt, es können Daten geschrieben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogNewFile":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, L, D, T, Z oder Konstante (T und Z sind nur bei S7-1500 in KOP und FUP verfügbar)	Ausführung der Anweisung bei steigender Flanke.
RECORDS	Input	UDInt	E, A, M, L, D oder Konstante	Anzahl der Datensätze im neuen Data Log.
NAME	Input	VARIANT	L, D	Dateiname des neuen Data Logs.
ID	InOut	DWORD	E, A, M, L, D	Objekt-ID des Data Logs • In: ID des bereits vorhandenen Data Logs • Out: ID des neuen Data Logs
DATA	InOut	VARIANT	D	Datentyp für Konsistenzprüfung
DONE	Output	BOOL	E, A, M, L, D	Anweisung wurde erfolgreich ausgeführt.
BUSY	Output	BOOL	E, A, M, L, D	Ausführung der Anweisung noch nicht abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, L, D	• 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, L, D	Zustandsparameter Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
0	Keine Fehler.
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8070	Die maximale Anzahl gleichzeitig aktiver DataLogNewFile-Anweisungen wurde bereits erreicht.
8090	Ungültiger Dateiname
8091	Datentyp am Parameter NAME ist kein STRING.
8092	Quellen- Data Log existiert nicht.
8093	Neues Data Log existiert bereits.
8097	Länge der Datei überschreitet Begrenzung des Dateisystems oder CPU-spezifische Begrenzungen.
80A0	Datentypen inkonsistent. Das Data Log am Parameter ID verwendet andere Datentypen als am Parameter DATA angegeben.
80A2	Schreibfehler durch das Dateisystem zurückgemeldet.
80B2	Maximale Anzahl von DataLogs überschritten oder DataLog-Ressourcen aufgebraucht
80B3	Ladespeicher nicht ausreichend.
80B4	Memory Card ist schreibgeschützt.
80C0	Zugriff momentan nicht möglich.
80C1	Zu viele Dateien geöffnet.
8253	Ungültiger Wert am Parameter RECORDS.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Beispielprogramm zum Arbeiten mit DataLogs (Seite 2803).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Beispielprogramm zum Arbeiten mit DataLogs

Einleitung

In dem folgenden Beispielprogramm erstellen Sie einen Data Log, schreiben einen oder mehrere Datensätze in den Data Log, schließen und öffnen den Data Log und erstellen eine neue Datei basierend auf dem Data Log. Eine Möglichkeit zum Löschen von Data Logs ist in dem Beispiel ebenso gegeben. Das Beispiel bietet außerdem die Möglichkeit zum Bearbeiten von bis zu acht Data Logs.

Für Detailinformationen zu einzelnen Anweisungen, öffnen Sie die entsprechende Beschreibung der Hilfe über die entsprechenden Links.

Allgemeine Hinweise zur Verwendung von Data Logs

- Nach Ausführung der Anweisungen "DataLogCreate" und "DataLogNew" wird das erstellte Data Log automatisch geöffnet.
- Nach einem Wechsel der CPU von RUN in STOP oder nach einem Neustart der CPU werden Data Logs automatisch geschlossen.
- Damit die Anweisung "DataLogWrite" ausgeführt wird, muss das Data Log geöffnet sein.
- Bei S7-1200-CPUs dürfen maximal acht, bei S7-1500-CPUs maximal zehn Data Logs gleichzeitig geöffnet sein.

Voraussetzungen

Sie nehmen die folgenden Einstellungen in den CPU-Eigenschaften vor:

- Unter "PROFINET-Schnittstelle > Zugriff auf den Webserver" und aktivieren Sie den Zugriff auf den Webserver.
- Unter dem Eintrag "Webserver" aktivieren den Webserver für das Modul.
- Im Abschnitt "Benutzerverwaltung" erstellen Sie einen Benutzer mit Lese-, Schreib- und Löschrechten für Dateien.

Ablage der Daten

Um einen Data Log aufrufen und bearbeiten zu können, werden der Name und die ID des Data Logs benötigt. Damit diese Daten nicht verloren gehen, erstellen Sie einen Speicher für Data Log-Einträge. Die Struktur eines Data Log-Eintrags definieren Sie über einen PLC-Datentyp "stackDataLog".

stackDataLog

	Name	Data type	Default value
1	name	String	''
2	ID	DWord	16#0
3	DLclosed	Bool	false

Zur Ablage der Daten erstellen Sie in einem globalen Datenbaustein ("SLI_gDB_Datalogging") die folgenden Variablen. Darunter befinden sich auch der Speicher für Data Log-Einträge und die Variablen zum Steuern des Speichers.

Hinweis: Wenn Sie die Data Logs auch physisch aus dem Ladespeicher der CPU löschen wollen, verwenden Sie die Variable "DLdelete.deleteMode" mit dem Wert "TRUE".

SLI_gDB_DataLogging

	Name	Data type	Start value	Retain
1	Static			
2	DLcreate	Struct		
3	execute	Bool	false	
4	busy	Bool	false	
5	error	Bool	false	
6	memErrStatus	Word	16#0	
7	dlogCreated	Bool	false	
8	DLdelete	Struct		
9	execute	Bool	false	
10	deleteMode	Bool	false	
11	busy	Bool	false	
12	error	Bool	false	
13	memErrStatus	Word	16#0	
14	dlogDeleted	Bool	false	
15	name	String	'myDataLogEE1'	
16	newName	String	'myNewDataLogEE1'	
17	logID	DWord	16#0	
18	logHeader	String	'Count,Temperature,Pressure'	
19	myData	Struct		
20	count	Int	0	
21	temperature	Real	0.0	
22	pressure	Real	0.0	
23	maxPosEntry	UInt	7	
24	dataLogEntries	Array[0..7] of "stackDataLog"		✓
25	nextPosInStack	UInt	0	✓
26	callEntry	UInt	0	

Der DB ("SLI_gDB_Datalogging") wird verwendet, um ein Data Log zu erstellen (DataLogCreate (Seite 2779)) oder zu löschen (DataLogDelete (Seite 2797)). Als Prozesswerte werden die drei Einträge der Struktur "myData" verwendet: count, temerature und pressure. In dem Datenbaustein werden diese drei Werte temporär gespeichert, um dann mit der Anweisung "DataLogWrite (Seite 2793)" als Datensatz in ein Data Log übertragen zu werden.

Ein Datensatz setzt sich später aus den folgenden Einträgen zusammen:

- Die Datensatznummer (wird automatisch vergeben)
- Dem Datum (wird automatisch vergeben, wenn bei DataLogCreate am Parameter TIMESTAMP "1" verwendet wird).
- Der Uhrzeit (wird automatisch vergeben, wenn bei DataLogCreate am Parameter TIMESTAMP "1" verwendet wird).
- Dem aktuellen Wert von "count" aus der Struktur "myData".
- Dem aktuellen Wert von "temperature" aus der Struktur "myData".
- Dem aktuellen Wert von "pressure" aus der Struktur "myData".

Die Prozesswerte für das Schreiben eines Datensatzes (DataLogWrite (Seite 2793)) stellen Sie über den den globalen Datenbaustein "SLI_gDB_DataLogW" bereit.

SLI_gDB_DataLogW

	Name	Data type	Start value	Retain
1	Static			
2	DLwrite	Struct		
3	execute	Bool	false	
4	preset	Struct		
5	count	Int	5	
6	scaledTemp	Real	25.7	
7	scaledPressure	Real	45.0	
8	busy	Bool	false	
9	error	Bool	false	
10	status	Word	16#0	
11	memErrStatus	Word	16#0	
12	memDone	Bool	false	
13	DLclose	Struct		
14	execute	Bool	false	
15	busy	Bool	false	
16	error	Bool	false	
17	status	Word	16#0	
18	memDone	Bool	false	
19	DLopen	Struct		
20	execute	Bool	false	
21	busy	Bool	false	
22	error	Bool	false	
23	status	Word	16#0	
24	memDone	Bool	false	
25	DLnewfile	Struct		
26	execute	Bool	false	
27	busy	Bool	false	
28	error	Bool	false	
29	status	Word	16#0	
30	newFileCreated	Bool	false	

FC "SLI_FC_saveEntry_DataLog": Parameter verschalten

Zum Abspeichern von Name und ID eines Data Logs verwenden Sie einen FC "SLI_FC_saveEntry_DataLog". Darin erstellen Sie die folgenden lokalen Variablen.

SLI_FC_saveEntry_DataLog		
	Name	Data type
1	Input	
2	name	String
3	logID	DWord
4	Output	
5	<Add new>	
6	InOut	
7	<Add new>	
8	Temp	
9	nextPosInStack	UInt
10	maxPosEntry	UInt
11	tempName	String
12	tempID	DWord

Netzwerk 1: Über die Eingangsparameter "#name" und "#logID" übergeben Sie die Werte der Variablen "name" und "logID".

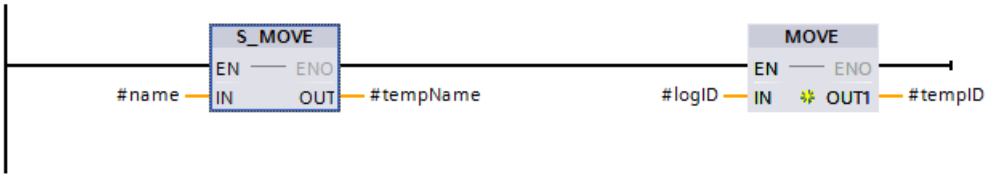

Netzwerk 2: Die Variable "nextPosInstack" nennt den zu verwendenden Speicherort für einen Data Log-Eintrag in dem Array "dataLogEntries". Die Variable "maxPosEntry" nennt die letzte Zelle in dem dem Array "dataLogEntries". Zur Weiterverarbeitung verschalten Sie die Werte der Variablen "nextPosInstack" und "maxPosEntry" wie folgt.

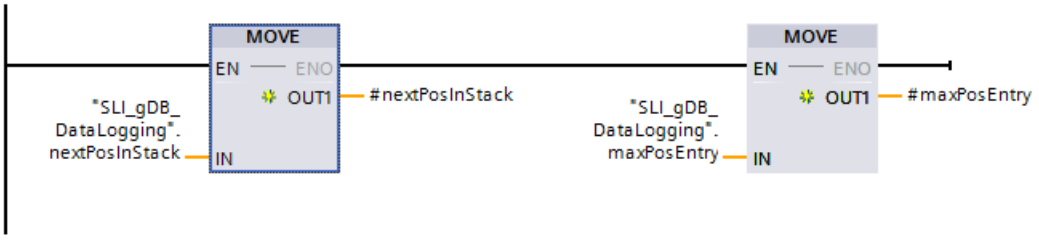

Netzwerk 3: Den Namen und die ID speichern Sie wie folgt in dem Array "dataLogEntries".

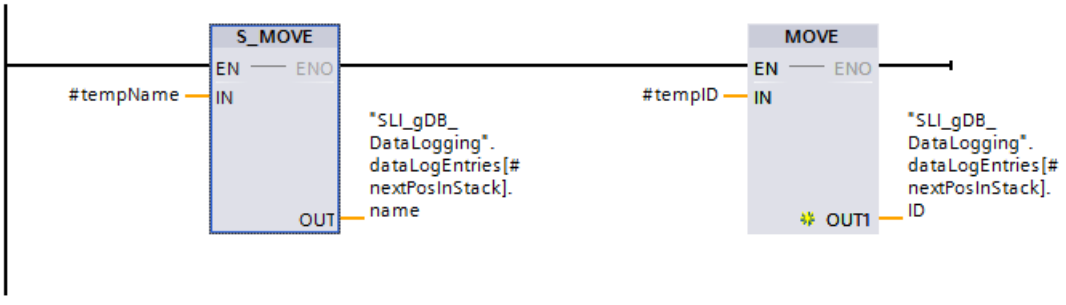

Netzwerk 4: Um die Ablage von Daten eines neuen Data Logs in einem anderen Eintrag des Speichers zu gewährleisten, inkrementieren Sie die Variable "nextPosInStack".

Hinweis: Wenn das Speicherlimit ("#maxPosEntry") erreicht ist, wird die Variable "nextPosInStack" nicht inkrementiert, sondern rückgesetzt auf den Wert "0". Das hat zur Folge, dass die Daten eines neuen Data Logs die Daten eines alten Data Logs überschreiben würden.

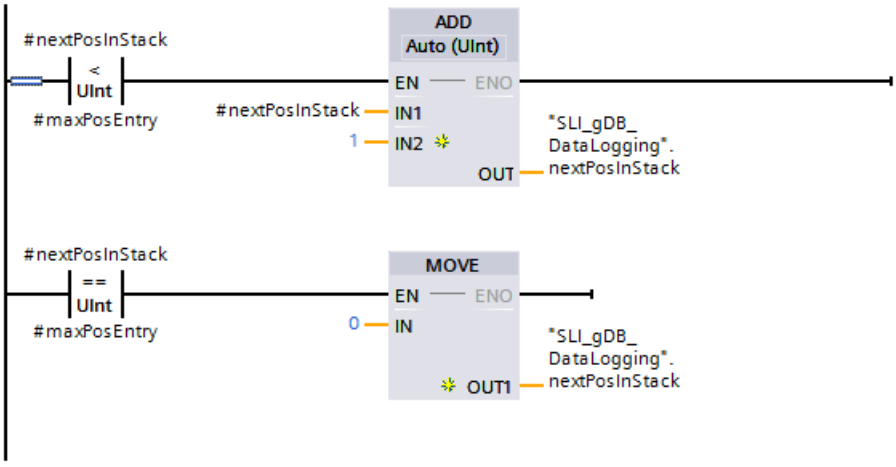

FC "SLI_FC_callEntry_DataLog": Parameter verschalten

Zum Aufrufen von Name und ID eines ausgewählten Data Log-Eintrags erstellen Sie einen FC "SLI_FC_callEntry_DataLog". Darin erstellen Sie die folgenden lokalen Variablen. Die Auswahl des Data Log-Eintrags spezfizieren Sie über die Variable "callEntry".

SLI_FC_callEntry_DataLog

		Name	Data type
1		Input	
2		callEntry	UInt
3		Output	
4		name	String
5		logID	DWord
6		InOut	
7		<Add new>	
8		Temp	
9		tempEntry	UInt
10		tempName	String
11		tempID	DWord

Netzwerk 1: Zur Weiterverarbeitung verschalten Sie den Wert der Variablen "callEntry" wie folgt.

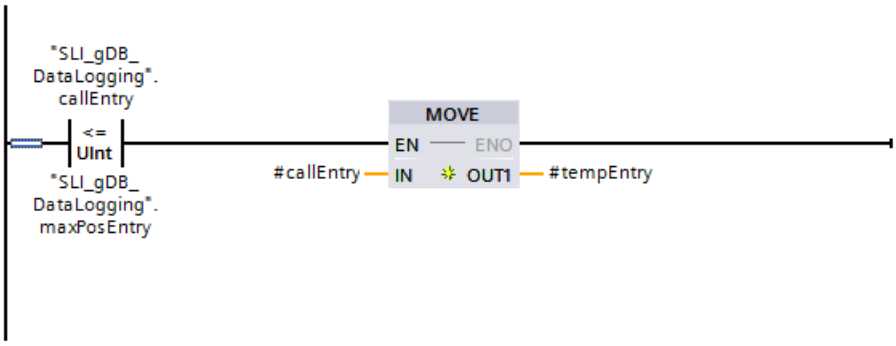

Netzwerk 2: Den Aufruf des Namens und der ID des erstellten Data Logs erzielen Sie wie folgt.

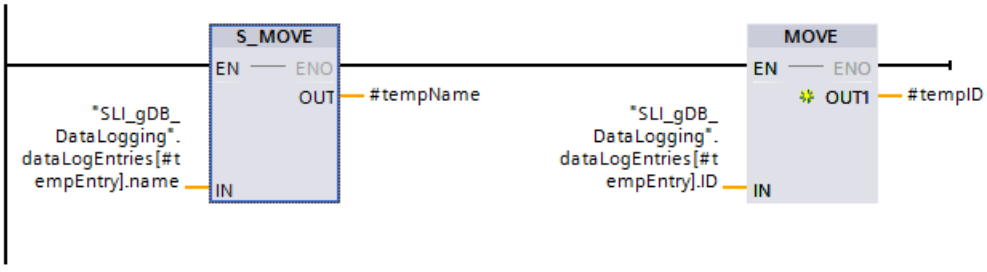

Netzwerk 3: Über die Ausgangsparameter "#name" und "#logID" übergeben Sie die Werte der Variablen "dataLogEntries.name" und "dataLogEntries.ID".

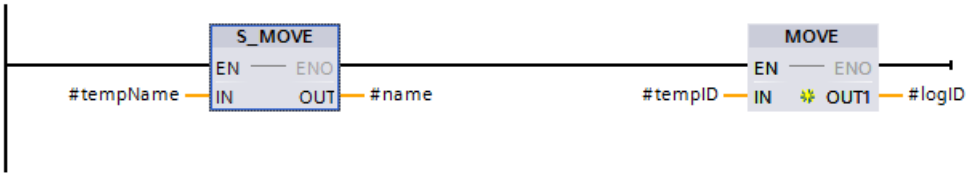

FC "SLI_FC_deleteEntry_DataLog": Parameter verschalten

Zum Löschen der Daten eines Data Log-Eintrags verwenden Sie einen FC "SLI_FC_deleteEntry_DataLog". Darin erstellen Sie die folgenden lokalen Variablen.

SLI_FC_deleteEntry_DataLog

		Name	Data type	Default value
1		Input		
2		deleteEntry	UInt	
3		Output		
4		<Add new>		
5		InOut		
6		<Add new>		
7		Temp		
8		tempEntry	UInt	
9		Constant		
10		EMPTYSTRING	String	''

Netzwerk 1: Im Folgenden stellen Sie sicher, dass die temporäre Variable "#tempEntry" einen Wert besitzt.

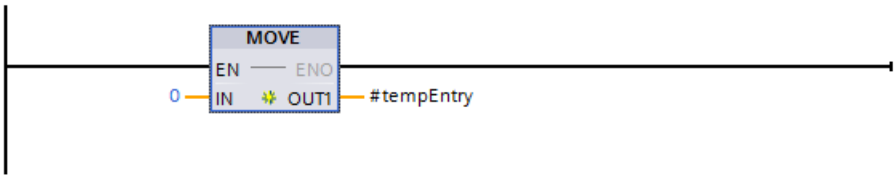

Netzwerk 2: Wenn der Wert der Variablen "callEntry" den Wert der Variablen "maxPosEntry" nicht überschreitet, soll folgendes passieren:

- Am Eingangsparameter #deleteEntry wird der Wert der Variablen "callEntry" übergeben. Der zu löschende Data Log-Eintrag wird über die Variable "callEntry" gewählt.
- Die Variablen "dataLogEntries.name", "dataLogEntries.ID" und "dataLogEntries.DLclosed" des gewählten Data Log-Eintrags werden rückgesetzt.

Die Verschaltungen sehen wie folgt aus.

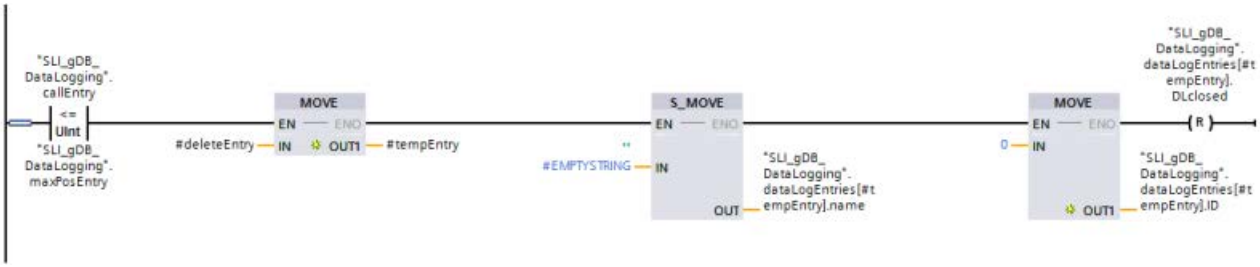

FC "SLI_FC_memCloseEntry_DataLog": Parameter verschalten

Zum Abspeichern des Zustands (offen oder geschlossen) des Data Logs verwenden Sie einen FC "SLI_FC_memCloseEntry_DataLog". Darin erstellen Sie die folgenden lokalen Variablen.

SLI_FC_memCloseEntry_DataLog		
	Name	Data type
1	Input	
2	callEntry	UInt
3	memClose	Bool
4	Output	
5	<Add new>	
6	InOut	
7	<Add new>	
8	Temp	
9	tempEntry	UInt
10	tempClose	Bool

Netzwerk 1: Zur Auswahl des Data Log-Eintrags übergeben Sie den Wert des Eingangsparameters "#callEntry". Zudem übergeben Sie über den Eingangsparameter "#memClose" den Zustand des Data Logs.

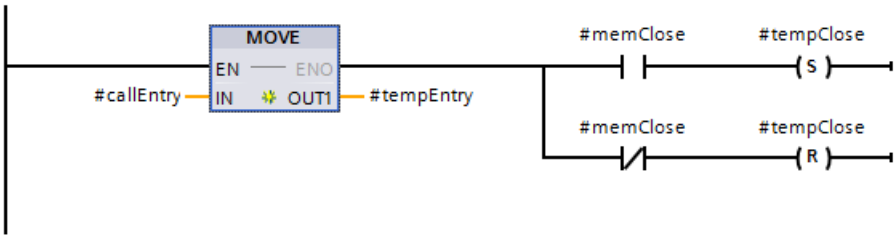

Netzwerk 2: Den Zustand des Data Logs speichern Sie in dem passenden Data Log-Eintrag. Die Verschaltungen sehen wie folgt aus.

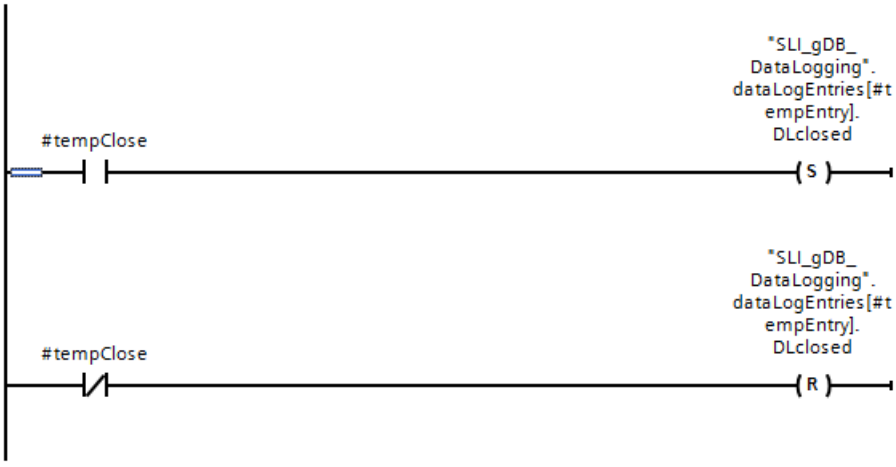

FB "SLI_FB_DataLogging": Parameter verschalten

Als zentralen Baustein des Beispielprogramms erstellen Sie einen FB "SLI_FB_DataLogging". Darin erstellen Sie die folgenden lokalen Variablen.

7	Static			
8	createDone	Bool	false	Non-retain
9	createStatus	Word	16#0	Non-retain
10	deleteDone	Bool	false	Non-retain
11	deleteStatus	Word	16#0	Non-retain
12	writeDone	Bool	false	Non-retain
13	executeClose	Bool	false	Non-retain
14	closeDone	Bool	false	Non-retain
15	openDone	Bool	false	Non-retain
16	memOpenDone	Bool	false	Non-retain
17	newFileDone	Bool	false	Non-retain
18	newFileStatus	Word	16#0	Non-retain
19	stateMem	Bool	false	Non-retain
20	stateMem2	Bool	false	Non-retain
21	stateMem3	Bool	false	Non-retain
22	stateMem4	Bool	false	Non-retain
23	stateMem5	Bool	false	Non-retain
24	stateMem6	Bool	false	Non-retain
25	DataLogCreate_Instan..	DataLogCreate		
26	DataLogWrite_Instance	DataLogWrite		
27	DataLogClose_Instance	DataLogClose		
28	DataLogOpen_Instance	DataLogOpen		
29	DataLogNewFile_Insta..	DataLogNewFile		
30	DataLogDelete_Instan..	DataLogDelete		
31	Temp			
32	<Add new>			
33	Constant			
34	EMPTYSTRING	String	''	

In dem FB legen Sie die folgenden Verschaltungen an. Den FB rufen Sie in einem Zyklus-OB (OB1) auf.

Netzwerk 1: Eine steigende Flanke am Eingangsparameter REQ ("DLcreate.execute") der Anweisung "DataLogCreate (Seite 2779)" startet die Data Log-Erstellung.

Der.Ausgangsparameter DONE ("#createDone") von DataLogCreate ist nur einen Zyklus lang gültig. Deswegen speichern Sie dessen Wert in der Variable "DLcreate.dlogCreated". Die Variable "DLcreate.execute" wird dabei zurückgesetzt.

Im Fehlerfall wird der Status ("DLcreate.status") in der Variable "DLcreate.memErrStatus" gespeichert.

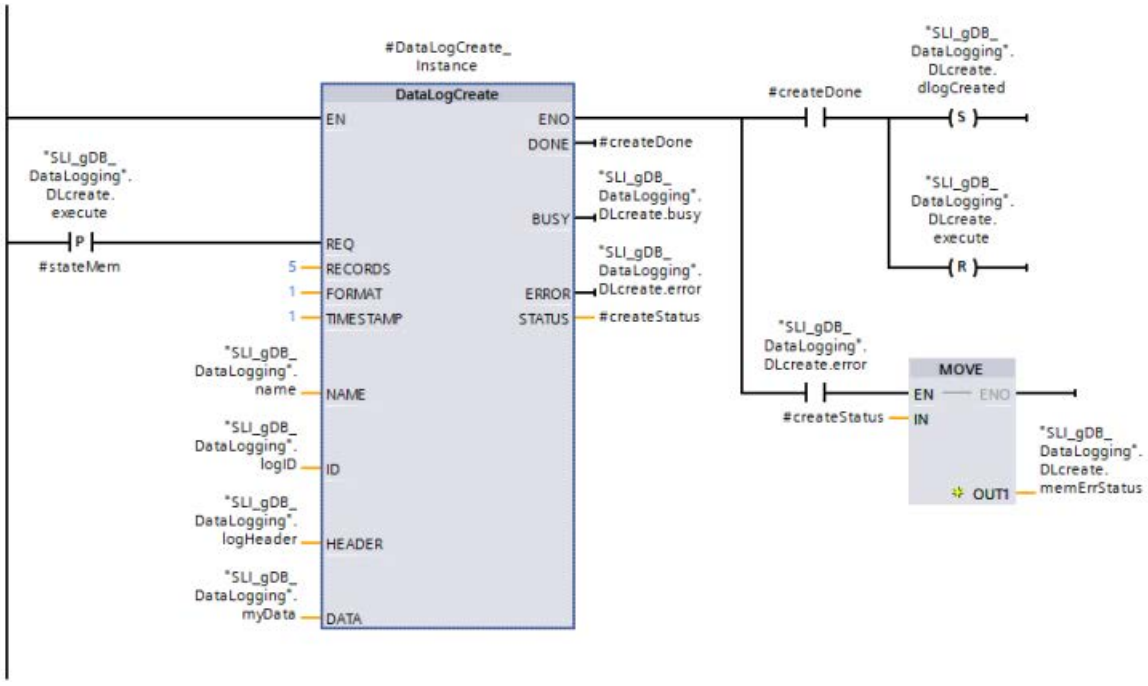

Netzwerk 2: Den Namen und die ID des Data Logs speichern Sie mit Hilfe des FC "SLI_FC_saveEntry_DataLog". Das Speichern erfolgt, wenn der Parameter DONE ("#createDone") den Signalzustand "TRUE" liefert.

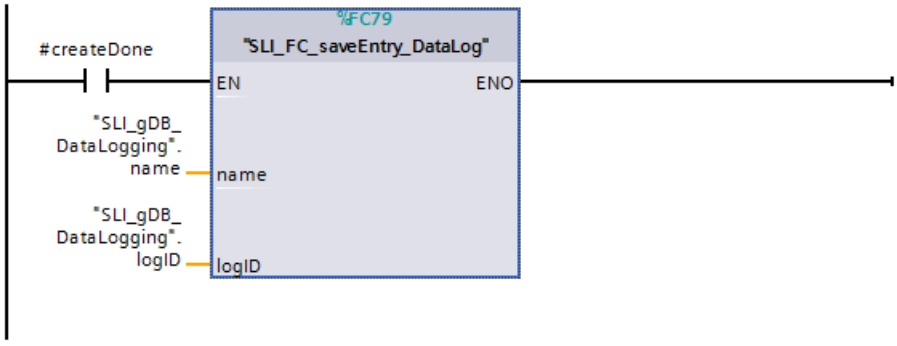

Netzwerk 3: Eine positive Flanke löst den Zeitpunkt aus, an dem neue Prozesswerte in der Struktur myData gespeichert werden. Dieser Schritt dient dazu, die gewünschten Prozesswerte temporär in dem Datenbaustein "SLI_gDB_DataLogging" zu speichern.

Des Weiteren wird mit Hilfe des FC "SLI_FC_callEntry_DataLog" der gewünschte Data Log aufgerufen. Über die Variable "callEntry" wird der Data Log anhand seines Data Log-Eintrags in dem Array "dataLogEntries" ausgewählt.

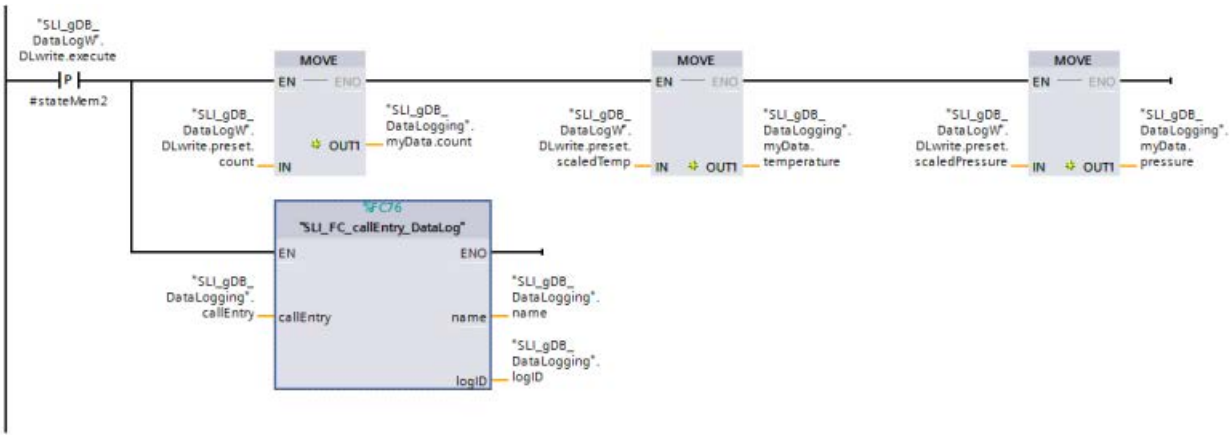

Netzwerk 4: Wenn ein Data Log erstellt wurde und der Wert der Variable "logID" nicht "0" ist, wird der Eingang EN von DataLogWrite (Seite 2793) gesetzt. Der Grund für die Abfrage ist, dass sich ein Erstellungsvorgang über mehrere Zyklen erstreckt und beendet sein muss, damit ein Schreibvorgang durchgeführt wird. Durch die positive Flanke am Eingangsparameter REQ ("DLwrite.execute") wird der Schreibvorgang für einen Datensatz ausgelöst.

Der Erfolgstatus von DataLogWrite ("#writeDone" ist "TRUE") wird über die Variable "DLwrite.memDone" gespeichert. Die Variable "DLwrite.execute" wird dabei zurückgesetzt.

Im Fehlerfall wird der Status ("DLwrite.status") in der Variable "DLwrite.memErrStatus" gespeichert. Beim Status "16#0001" löst die Anweisung "DataLogWrite" automatisch eine Schließung des Data Logs ("DLclose.execute" ist "TRUE") aus.

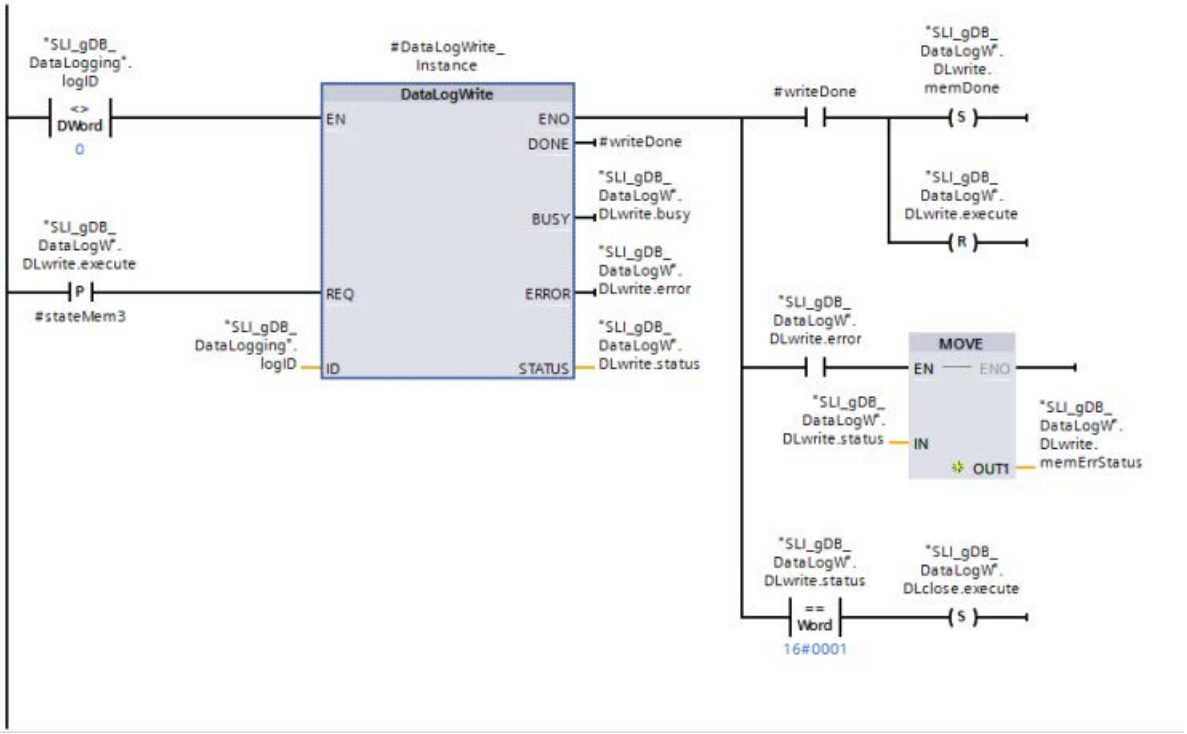

Hinweis

Menge der Datensätze im Data Log

Das Data Log wird für fünf Datensätze erstellt (siehe Netzwerk 1).

Das bedeutet:

- Nach fünf Datensätzen wird an der Anweisung "DataLogWrite (Seite 2793)" am Parameter STATUS ("DLwrite.status") der Wert "0001" ausgegeben.
- Der letzte, möglicher Datensatz am Ende der Datei ist erstellt. Beim Erstellen eines weiteren Datensatzes wird ein Älterer überschrieben.

Netzwerk 5: Vor der Schließung eines Data Logs wählen Sie den zu schließenden Data Log aus (via "callEntry"). Wenn der Schließerkontakt ("DLclose.execute") den Signalzustand "TRUE" liefert, wird der gewünschte Data Log anhand seines Eintrags im Array "dataLogEntries" aufgerufen. Zudem wird "#executeClose" auf "TRUE" gesetzt. Legen Sie dazu die folgenden Verschaltungen an.

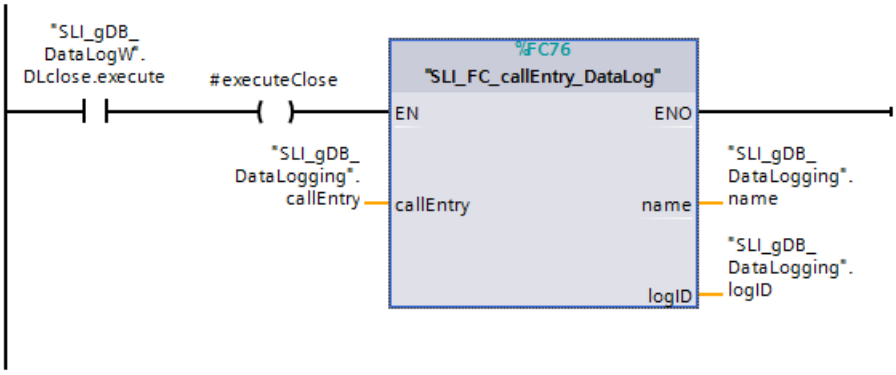

Netzwerk 6: Schließen Sie das Data Log, nachdem der letzte Datensatz geschrieben wurde oder die Bearbeitung des Data Logs abgeschlossen ist.

Dazu wird der Eingang REQ ("#executeClose") gesetzt, der die Anweisung "DataLogClose (Seite 2795)" ausführt. Wenn das Data Log geschlossen ist, werden keine Datensätze mehr geschrieben.

Der Erfolgstatus von DataLogClose ("#closeDone" ist "TRUE") wird über die Variable "DLclose.memDone" gespeichert.

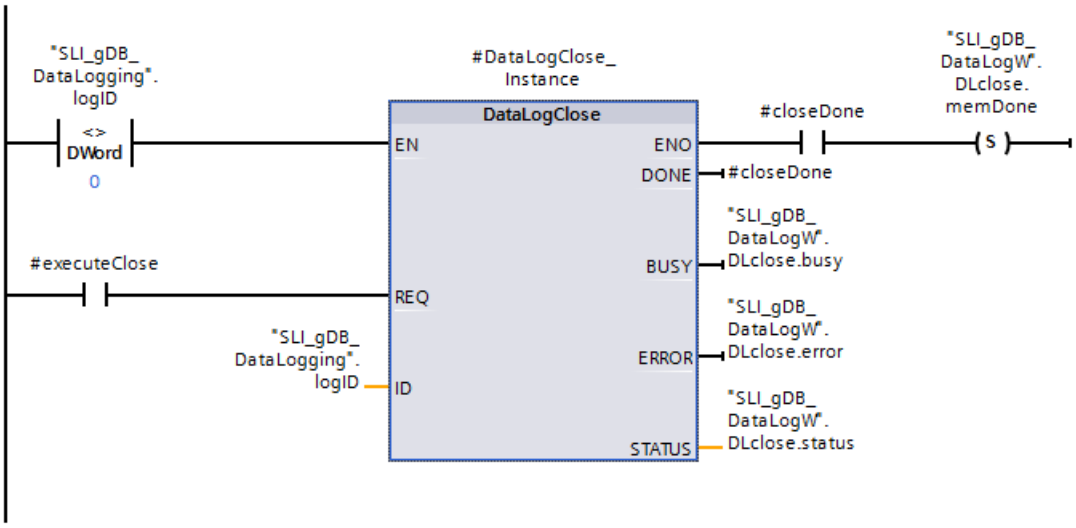

Netzwerk 7: Lassen Sie die Variable "DLclose.execute" wie folgt automatisch zurücksetzen.

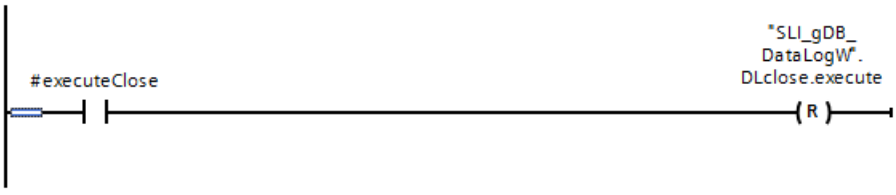

Netzwerk 8: Anhand der folgenden Verschaltung speichern Sie den Status "geschlossen" ("DLclose.memDone" ist "TRUE") des Data Logs in dem Data Log-Eintrag ("dataLogEntries.DLclosed" ist "TRUE").

Netzwerk 9: Vor dem Öffnen eines Data Logs wählen Sie den zu öffnenden Data Log aus (via "callEntry"). Wenn der Schließerkontakt ("DLopen.execute") den Signalzustand "TRUE" liefert, wird der gewünschte Data Log anhand seines Eintrags im Array "dataLogEntries" aufgerufen. Legen Sie dazu die folgenden Verschaltungen an.

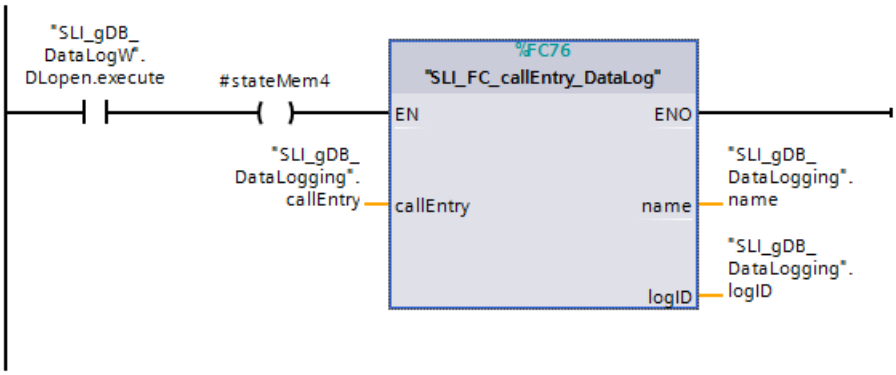

Netzwerk 10: Um zu einem späteren Zeitpunkt wieder einen Datensatz schreiben zu könnnen, muss das Data Log mit der Anweisung "DataLogOpen (Seite 2789)" wieder geöffnet werden. Wenn jetzt ein weiterer Datensatz mit DataLogWrite (Seite 2793) geschrieben wird, wird immer der älteste Datensatz überschrieben.

Der Erfolgstatus von DataLogOpen ("#openDone" ist "TRUE") wird über die Variable "DLopen.memDone" gespeichert. Das negierte Ergebnis von "#openDone" wird über die Variable "#memOpenDone" gespeichert.

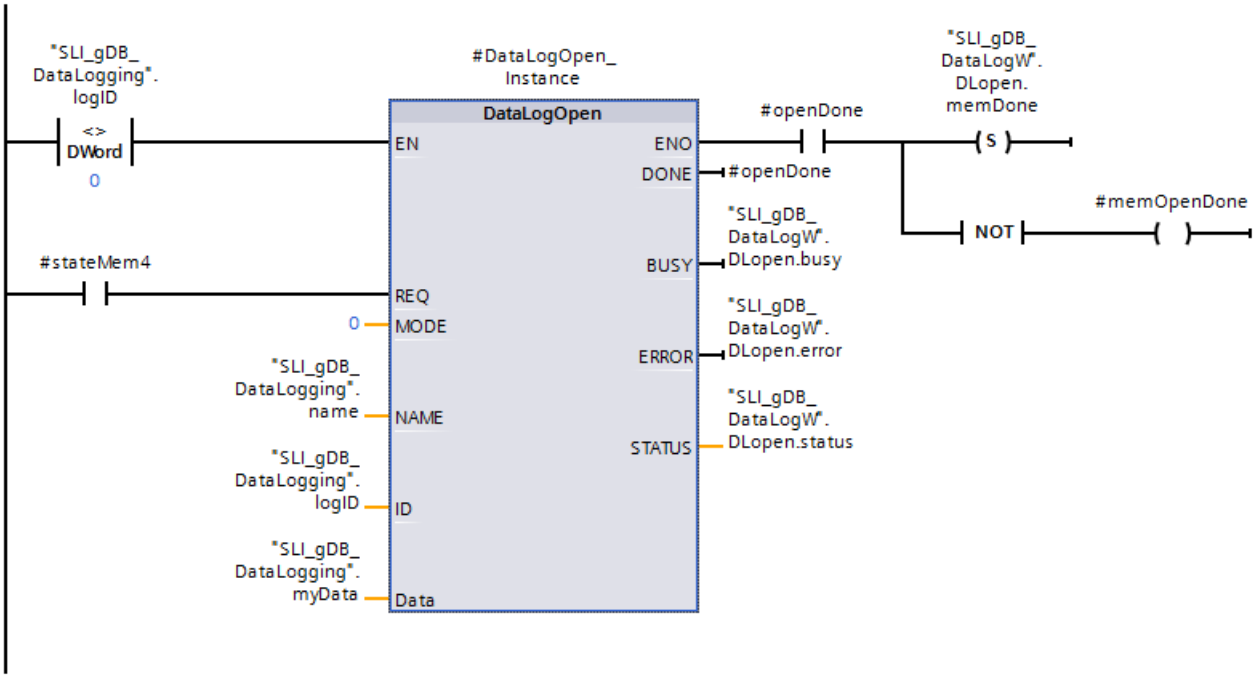

Netzwerk 11: Lassen Sie die Variable "DLopen.execute" wie folgt automatisch zurücksetzen.

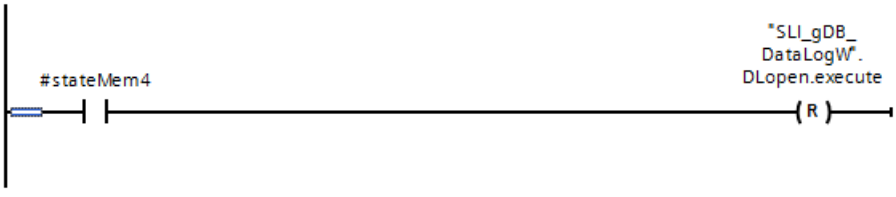

Netzwerk 12: Anhand der folgenden Verschaltung speichern Sie den Status "offen" des Data Logs in dem Data Log-Eintrag ("#memClose" bzw. "dataLogEntries.DLclosed" ist "FALSE").

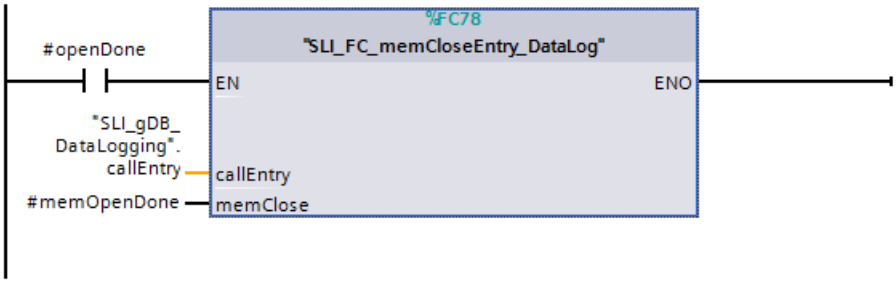

Netzwerk 13: Um eine neue Datei auf Basis eines Data Logs erstellen zu können, müssen Sie zuerst den zu verwendenden Data Log auswählen (via "callEntry"). Wenn der Schließerkontakt ("DLnewfile.execute") den Signalzustand "TRUE" liefert, wird der gewünschte Data Log anhand seines Eintrags im Array "dataLogEntries" aufgerufen. Legen Sie dazu die folgenden Verschaltungen an.

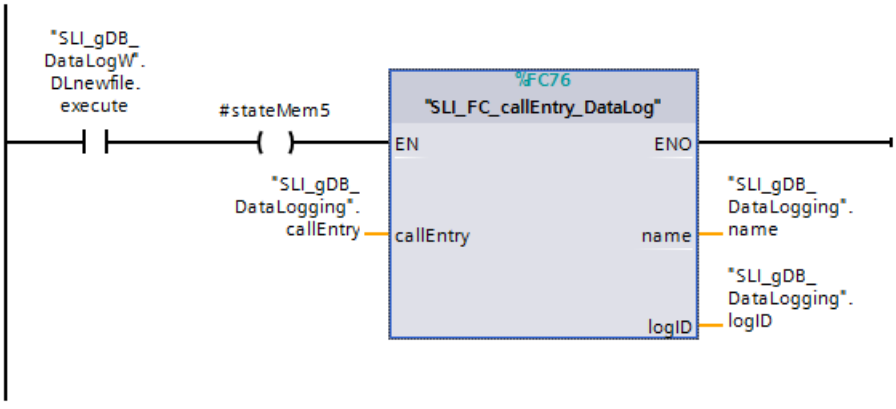

Netzwerk 14: Falls Sie die älteren Datensätze nicht überschreiben möchten, haben Sie mit der Anweisung "DataLogNewFile (Seite 2801)" die Möglichkeit, ein neues Data Log mit dem gleichen Aufbau zu erstellen. Hierzu geben Sie am Parameter ID der Anweisung die ID des vorhandenen Data Logs an, dessen Struktur Sie kopieren möchten. Nachdem die Anweisung "DataLogNewFile" ausgeführt wurde, wird ein neuer und eindeutiger ID-Wert für das neue Data Log vergeben.

Durch die positive Flanke am Eingangsparameter REQ ("#stateMem5") wird der Erstellungsvorgang ausgelöst. Der Erfolgstatus von DataLogNewFile ("#newFileDone" ist "TRUE") wird über die Variable "DLnewfile.memDone" gespeichert. Im Fehlerfall wird der Status ("#newFileStatus") in der Variable "DLnewfile.memErrStatus" gespeichert.

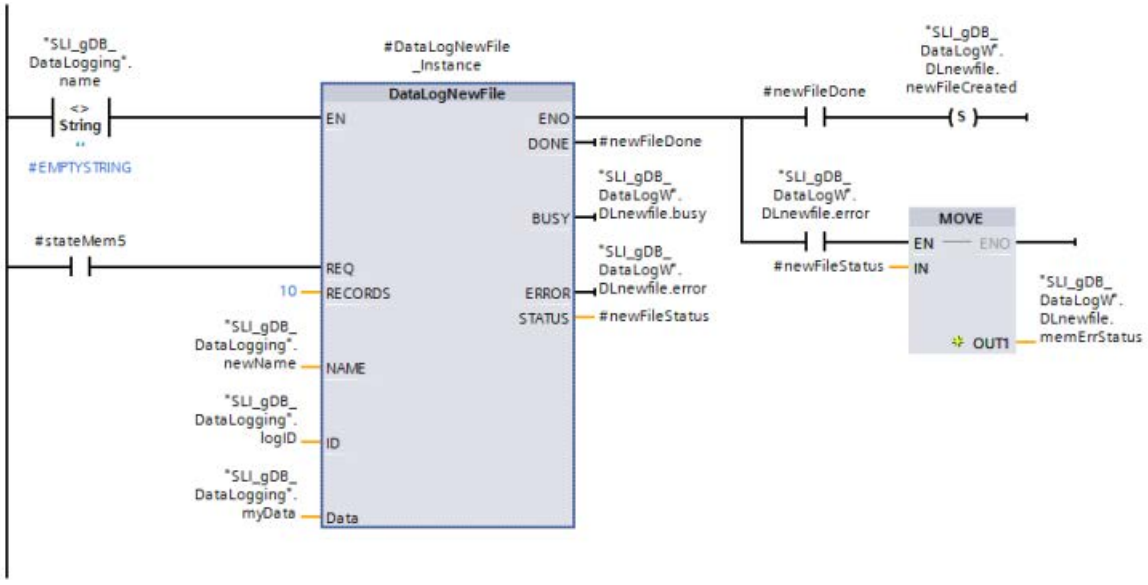

Hinweis

Aufruf über mehrere Zyklen

Beachten Sie, dass sich auch bei DataLogNewFile (Seite 2801) der Aufruf über mehrere Zyklen erstreckt. Zum Beispiel könnten Sie eine zu frühe Ausführung der Anweisung "DataLogNewFile" durch die Abfrage des Parameters BUSY der Anweisung DataLogWrite (Seite 2793) verhindern.

Netzwerk 15: Lassen Sie die Variable "DLnewfile.execute" wie folgt automatisch zurücksetzen.

Netzwerk 16: Den Namen und die ID des Data Logs speichern Sie mit Hilfe des FC "SLI_FC_saveEntry_DataLog". Das Speichern erfolgt, wenn der Parameter DONE ("#newFileDone") den Signalzustand "TRUE" liefert.

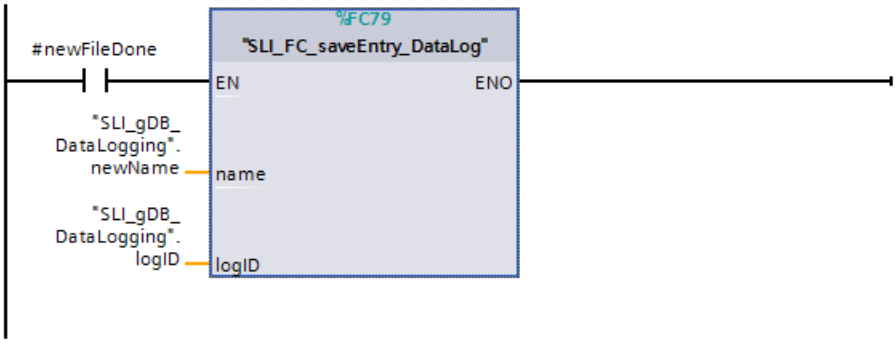

Netzwerk 17: Um einen Data Log löschen zu können, müssen Sie zuerst den zu löschenden Data Log auswählen (via "callEntry"). Wenn der Schließerkontakt ("DLdelete.execute") den Signalzustand "TRUE" liefert, wird der gewünschte Data Log anhand seines Eintrags im Array "dataLogEntries" aufgerufen. Legen Sie dazu die folgenden Verschaltungen an.

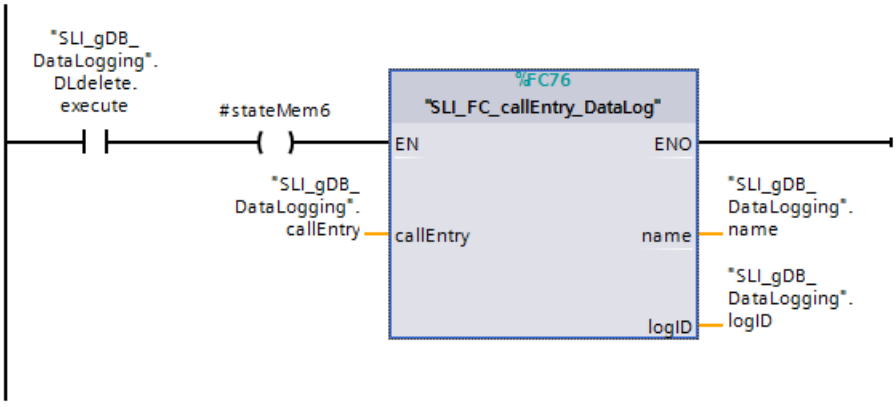

Netzwerk 18: Löschen Sie nicht mehr verwendete Data Logs mit Hilfe der Anweisung "DataLogDelete (Seite 2797)".

Durch die positive Flanke am Eingangsparameter REQ ("#stateMem6") wird der Löschvorgang für einen Data Log ausgelöst. Der Erfolgstatus von DataLogDelete (DONE ist "TRUE") wird über die Variable "DLdelete.dlogDeleted" gespeichert. Im Fehlerfall wird der Status ("#deleteStatus") in der Variable "DLdelete.memErrStatus" gespeichert.

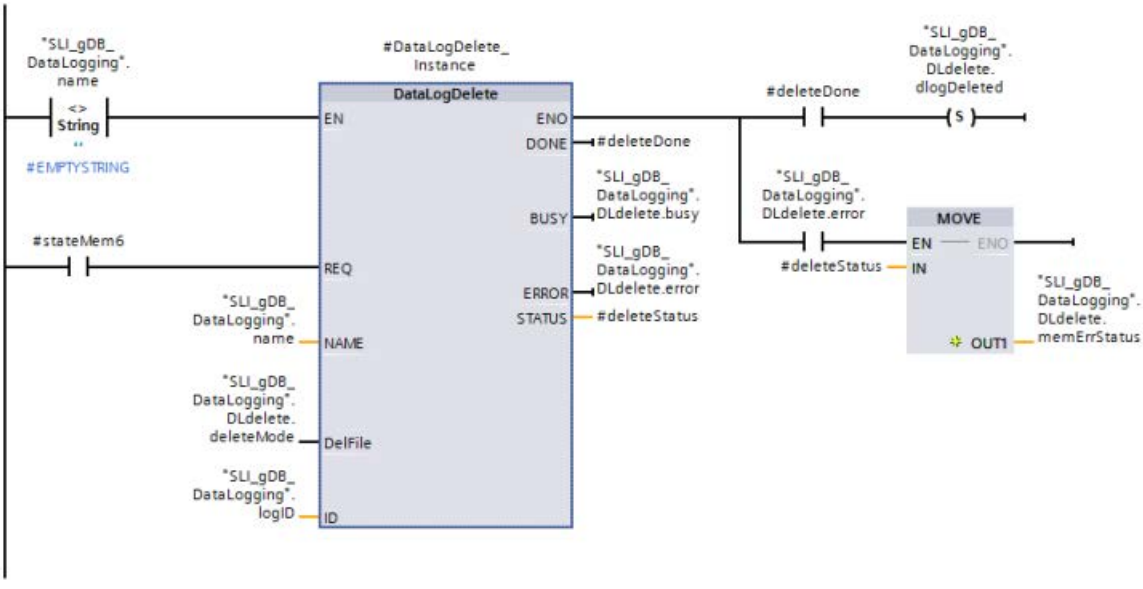

Netzwerk 19: Lassen Sie die Variable "DLdelete.execute" wie folgt automatisch zurücksetzen.

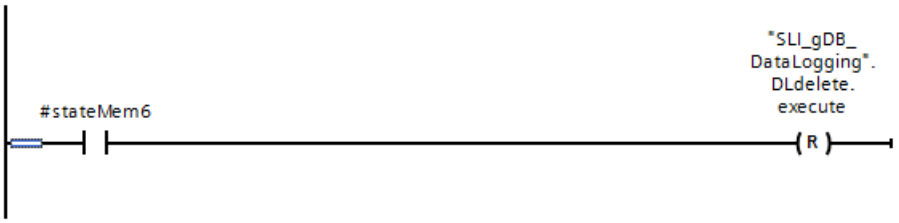

Netzwerk 20: Den Eintrag des Data Logs löschen Sie mit Hilfe des FC "SLI_FC_deleteEntry_DataLog". Das Löschen erfolgt, wenn der Parameter DONE ("#deleteDone") den Signalzustand "TRUE" liefert.

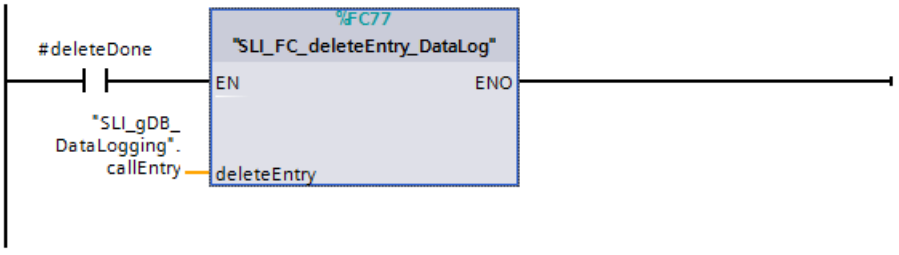

Ergebnis

Öffnen der geschriebenen Daten über den Webserver

Die mit dem Beispielprogramm angelegten Data Logs werden über den Webserver angezeigt. Öffnen Sie hierzu den Webserver über den Internet-Browser und öffnen Sie das Verzeichnis "\DataLogs".

Inhalte der CSV-Dateien

- Beim Erstellen des Data Logs mit der Anweisung DataLogCreate wurde als maximale Anzahl von Datensätzen "5" eingestellt. Wenn diese Anzahl noch nicht überschritten ist, sind in dem Data Log alle geschriebenen Datensätze enthalten.

	A	B	C	D	E	F
1	Record	Date	UTC Time	Count	Temperature	Pressure
2	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01
7						

- Wenn ein weiterer Datensatz hinzu kommt, wird der älteste Datensatz (Record 1) überschrieben.

	A	B	C	D	E	F
1	Record	Date	UTC Time	Count	Temperature	Pressure
2	6	9/30/2010	20:32:03	6	9.86E+01	3.58E+01
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01
7						

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

4.1.4.12 Datenbausteinfunktionen

CREATE_DB: Datenbaustein erzeugen

Beschreibung

Mit der Anweisung "CREATE_DB" erstellen Sie im Lade- und/oder Arbeitsspeicher einen neuen Datenbaustein.

Die Anweisung "CREATE_DB" verändert nicht die Prüfsumme des Anwenderprogramms.

Ein Datenbaustein, der nur im Arbeitsspeicher erzeugt wurde, hat die folgenden Eigenschaften:

- Nach Memory Reset oder NETZ AUS / NETZ EIN ist dieser Baustein nicht mehr vorhanden.
- Beim Laden oder bei einem STOP-RUN-Übergang bleibt sein Inhalt unverändert.

Nummer des Datenbausteins

Der erzeugte Datenbaustein erhält die kleinste freie Nummer aus dem Bereich, der durch die Parameter LOW_LIMIT (Untergrenze) und UP_LIMIT (Obergrenze) definiert wird. Diese Nummer wird am Parameter DB_NUM ausgegeben. Wenn im zulässigen Bereich der CPU eine Nummer gefunden wird, wird der DB erzeugt.

Ein DB mit einer bestimmten Nummer erzeugen Sie, indem Sie für die Ober- und Untergrenze des anzugebenden Bereichs die gleiche Nummer vergeben. Wenn bereits ein DB mit gleicher Nummer im Arbeitsspeicher und/oder im Ladespeicher vorhanden ist oder liegt der DB als kopierte Variante vor, wird die Anweisung beendet und eine Fehlermeldung am Parameter RET_VAL ausgegeben.

Startwerte des Datenbausteins

Über den Parameter SRCBLK definieren Sie für den zu erzeugenden DB Startwerte. Der Parameter SRCBLK ist ein Zeiger auf ein DB oder ein DB-Bereich, aus dem Sie die Startwerte übernehmen. Der am Parameter SRCBLK adressierte DB muss mit Standardzugriff erstellt sein (Attribut "Optimierter Bausteinzugriff" deaktiviert).

- Wenn der am Parameter SRCBLK angegebene Bereich größer ist als der erzeugte DB werden die Werte bis zur Länge des erzeugten DBs als Startwerte übernommen.
- Wenn der am Parameter SRCBLK angegebene Bereich kleiner ist als der erzeugte DB werden die übrigen Werte mit "0" aufgefüllt.

Um Datenkonsistenz zu gewährleisten, dürfen Sie diesen Datenbereich während der Bearbeitung von "CREATE_DB" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Hinweis

SRCBLK zeigt auf eine TEMP-Variable

TEMP-Variablen von Standard-Bausteinen befinden sich im temporären Lokaldatenbereich. Daher hat ein Verweis in SRCBLK auf eine TEMP-Variable zur Folge, dass als RET_VAL der Wert 80A3 (Der Datenbaustein am Parameter SRCBLK wurde nicht mit Standardzugriff erstellt.) ausgegeben wird.

Arbeitsweise

Die Anweisung "CREATE_DB" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "CREATE_DB" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CREATE_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Pegelgetriggerter Steuerparameter "request to activate" REQ = 1: Anforderung zum Erzeugen des Datenbausteins
LOW_LIMIT	Input	UINT	E, A, M, D, L oder Konstante	Untergrenze des Bereichs zur Vergabe einer DB-Nummer. Die kleinste mögliche DB-Nummer ist 60000.
UP_LIMIT	Input	UINT	E, A, M, D, L oder Konstante	Obergrenze des Bereichs, aus dem "CREATE_DB" Ihrem DB eine Nummer vergibt (Größt mögliche DB-Nummer: 60999)
COUNT	Input	UDINT	E, A, M, D, L oder Konstante	Der Zählwert gibt die Anzahl der Bytes an, die Sie für den erzeugten DB reservieren wollen. Für die Anzahl der Bytes muss eine gerade Nummer vergeben werden. Die maximale Länge beträgt 65534 Bytes.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ATTRIB	Input	BYTE	E, A, M, D, L oder Konstante	Über die ersten 4 Bit des Bytes am Parameter ATTRIB definieren Sie die Eigenschaften des Datenbausteins *:
				• Bit 0 = 0: Attribut "Nur im Ladespeicher ablegen" nicht gesetzt. • Bit 0 = 1: Attribut "Nur im Ladespeicher ablegen" gesetzt. Der DB belegt bei dieser Einstellung keinen Platz im Arbeitsspeicher und wird nicht in das Programm eingebunden. Auf den DB kann mit Bitbefehlen nicht zugegríffen werden. Bei Bit 0 = 1 ist die Auswahl für Bit 2 irrelevant. Aus Kompatibilitätsgründen zu STEP 7 V5.x müssen die Bits 0 uns 3 zusammen betrachtet werden (siehe unten).
				• Bit 1 = 0: Attribut "Datenbaustein im Gerät schreibgeschützt" nicht gesetzt. • Bit 1 = 1: Attribut "Datenbaustein im Gerät schreibgeschützt" gesetzt.
				• Bit 2 = 0: DB remanent (nur bei DBs, die im Lade- und im Arbeitsspeicher erzeugt werden). Wenn mindestens ein Wert remanent gesetzt wurde, gilt der DB als remanent. • Bit 2 = 1: DB nicht remanent Remanenz wird nicht unterstützt bei DBs, die nur im Ladespeicher oder nur im Arbeitsspeicher abgelegt werden. Wenn SIe die Anweisung "CREATE_DB" mit einer der beiden Kombinationen "remanent und nur Ladespeicher" oder "remanent und nur Arbeitsspeicher" aufrufen, wird der zu erzeugende DB als nicht remanent gekennzeichnet.
				• Bit 3 = 0: Erstellung des DBs entweder im Lade- oder im Arbeitsspeicher (Auswahl durch Bit 0, siehe oben) • Bit 3 = 1: Erstellung des DBs sowohl im Lade- als auch im Arbeitsspeicher (Bit 0 irrelvant) Aus Kompatibilitätsgründen zu STEP7 V5.x müssen die Bits 0 und 3 in Kombination verwendet werden. Wenn Bit 3 = 1 ist, ist Bit 0 irrelevant.

Bit 0	Bit 3	DB Erstellung
0	0	Nur im Arbeitsspeicher
1	0	Nur im Ladespeicher
Irrelevant	1	Arbeits- und Ladespeicher

				• Bit 4 = 0 - Keine Startwerte vorgegeben (Eingangswerte am Parameter SRCBLK werden ignoriert). • Bit 4 = 1 - Startwerte vorgeben (Werte entsprechen den über Parameter SRCBLK abdressierten DB).
SRCBLK	Input	VARIANT	D	Zeiger auf den Datenbaustein, mit dessen Werten der zu erzeugende Datenbaustein initialisiert wird.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Vorgang ist noch nicht beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB_NUM	Output	DB_DYN (UINT)	E, A, M, D, L	Nummer des erzeugten DBs.
* Die hier ausgewählten Eigenschaften entsprechen den Attributen in den Eigenschaften eines Datenbausteins.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs bleiben unverändert.
7000	Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1"
8081	Der Quellbereich ist größer als der Zielbereich. Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.
8092	Die Funktion "Erzeugen eines DB" ist momentan nicht durchführbar, weil • die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist • die maximale Anzahl an Bausteinen auf Ihrer CPU bereits erreicht ist
8093	Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
8094	Beim Parameter ATTRIB wurde ein ungültiger Wert angegeben.
80A1	Fehler bei der DB-Nummer: • Die Nummern im vorgegebenen Bereich (LOW_LIMIT bis UP_LIMIT) sind belegt oder liegen nicht im zulässigen Bereich der CPU • Untergrenze > Obergrenze
80A2	Fehler bei der DB-Länge: • die Länge ist "0" • die Länge ist eine ungerade Zahl • die Länge ist größer, als die CPU zulässt
80A3	Der Datenbaustein am Parameter SRCBLK wurde nicht mit Standardzugriff erstellt.
80B1	Es ist keine DB-Nummer frei.
80B2	Nicht genügend freier Speicher verfügbar.
80B4	Die Memory-Card ist schreibgeschützt.
80BB	Nicht genügend Ladespeicher frei.
80C3	Die maximale Anzahl gleichzeitig aktiver "CREATE_DB"-Anweisungen ist momentan bereits erreicht.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu CREATE Funktionen (Seite 2841).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

READ_DBL: Aus Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung kopieren Sie einen DB oder einen Teil eines DBs, der im Ladespeicher (Micro Memory Card) liegt, in den Datenbereich eines Ziel-DBs. Der Ziel-DB muss ablaufrelevant sein. Er darf nicht mit dem Attribut "Nur im Ladespeicher ablegen" erzeugt worden sein. Der Inhalt des Ladespeichers wird beim Kopiervorgang nicht verändert.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Zielbereich während der Bearbeitung von "READ_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Die Parameter SRCBLK und DSTBLK (Quellbaustein und Zielbaustein) müssen beide vom Datentyp VARIANT sein. Dabei gelten folgende Einschränkungen:

- Gültig für Standard-Speicherbereiche: Wenn Sie einen Parameter vom Datentyp VARIANT (Quell- oder Zielbereich) mit einer Variablen vom Datentyp BOOL oder einem ARRAY of BOOL verschalten möchten, dann haben Sie folgende Möglichkeiten:
  - Sie können ihn symbolisch adressieren
    Beispiel: Parameter SRCBLK: "Data_block".myArray
  - Sie können ihn absolut mit Hilfe des ANY-Pointers adressieren. Dabei müssen Sie allerdings beachten, dass die angegebene Länge des Bereichs durch 8 teilbar sein muss, da sonst die Anweisung nicht ausgeführt wird.
    Beispiel: Parameter SRCBLK: P#DB123.DBX456.0 BOOL 1000
- Bei einem VARIANT-Pointer vom Typ STRING muss die Länge gleich 1 sein.
- Quell- und Zielbaustein müssen mit dem gleichen Bausteinzugriff erstellt worden sein. Beide müssen entweder die Zugriffsart "Optimiert" oder "Standard" verwenden.
- Wenn der Quell- und der Ziel-DB Bausteine mit optimiertem Zugriff sind, dürfen Sie den Datentyp STRUCT nicht verwenden.

Hinweis

"READ_DBL" wird asynchron bearbeitet und eignet sich deshalb nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu lesen.

Ein einmal begonnener Auftrag wird stets zu Ende geführt. Wenn die maximale Anzahl gleichzeitig aktiver "READ_DBL"-Anweisungen erreicht ist und Sie "READ_DBL" zu diesem Zeitpunkt in einer Prioritätsklasse hoher Priorität erneut aufrufen, dann wird der Fehlercode W#16#80C3 zurückgeliefert. Es ist daher nicht sinnvoll, den hoch prioren Auftrag sofort wieder zu starten.

Arbeitsweise

Die Anweisung "READ_DBL" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "READ_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_DBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen
SRCBLK	Input	VARIANT	D	Zeiger auf den Datenbaustein im Ladespeicher, aus dem gelesen wird
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.
DSTBLK	Output	VARIANT	D	Zeiger auf den Datenbaustein im Arbeitsspeicher, in den geschrieben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert.
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".

Fehlercode* (W#16#...)	Beschreibung
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
8x51	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Datentypfehler im Datenbaustein.
8081	Der Quellbereich ist größer als der Zielbereich. Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.
8082	• Ziel-DB und Quell-DB sind von unterschiedlichem Typ (optimierter Zugriff/Standardzugriff). • Ziel- und Quell-DB sind beide mit optimiertem Zugriff erstellt worden. Sie haben bei Ziel- und Quell-DB jedoch Strukturen verwendet, die beim Übersetzen zu einer unterschiedlichen Anordnung der Daten führen. (Hinweis: Verwenden Sie stattdessen UDTs).
8093	Hinweis: Diesen Fehlercode gibt es nur bei S7-1500-CPUs. Beim Parameter DSTBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
80B1	Hinweis: Diesen Fehlercode gibt es nur bei S7-1500-CPUs. Am Parameter SRCBLK sind nur Datenbausteine zulässig, die sich im Ladespeicher befinden.
8xB1	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Beim Parameter SRCBLK ist kein Datenbaustein angegeben, oder der dort angegebene Datenbaustein ist kein Ladespeicherobjekt.
80B4	DSTBLK zeigt auf einen DB mit F-Attribut (der nicht beschrieben werden darf)
8xC0	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Der Ziel-DB wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.
80C3	Die maximale Anzahl gleichzeitig aktiver "READ_DBL"-Anweisungen ist momentan bereits erreicht.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie den Inhalt von einem DB im Ladespeicher in einem DB im Arbeitsspeicher aus.

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein drei Variablen an.

SLI_gDB_READ_DBL

		Name	Data type	Start value
1		Static		
2		execute	Bool	false
3		busy	Bool	false
4		returnValue	Int	0

Zur Ablage der Sendedaten legen Sie zwei globale Datenbausteine mit je drei Variablen vom Datentyp INT an.

- Für Empfangsdaten im Ladespeicher:

SourceDB_LoadMemory

	Name	Data type	Offset	Start value
1	Static			
2	MySourceValue1	Int	0.0	111
3	MySourceValue2	Int	2.0	222
4	MySourceValue3	Int	4.0	333

- Für Empfangsdaten im Arbeitsspeicher:

TargetDB

	Name	Data type	Offset	Start value
1	Static			
2	MyTargetValue1	Int	0.0	0
3	MyTargetValue2	Int	2.0	0
4	MyTargetValue3	Int	4.0	0

Sie legen die nachfolgenden Verschaltungen in einem FB ("SLI_FB_READ_DBL") an.

Netzwerk 1: Die Parameter der Anweisung "READ_DBL" verschalten Sie wie folgt.

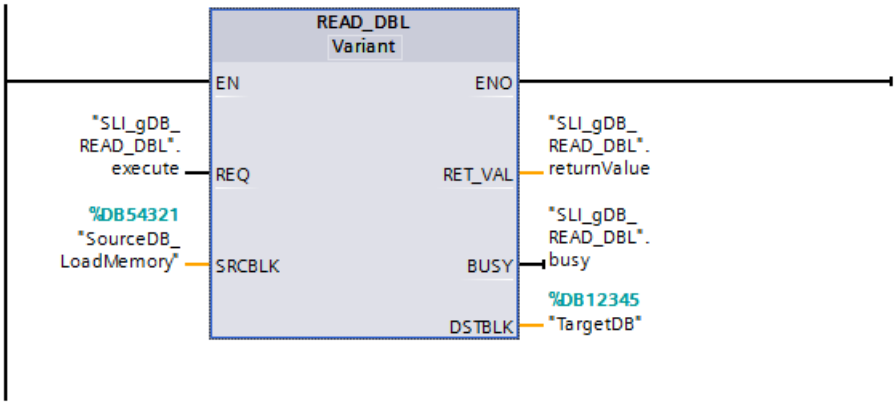

Netzwerk 2: Die Verarbeitung von READ_DBL stoppen Sie wie folgt.

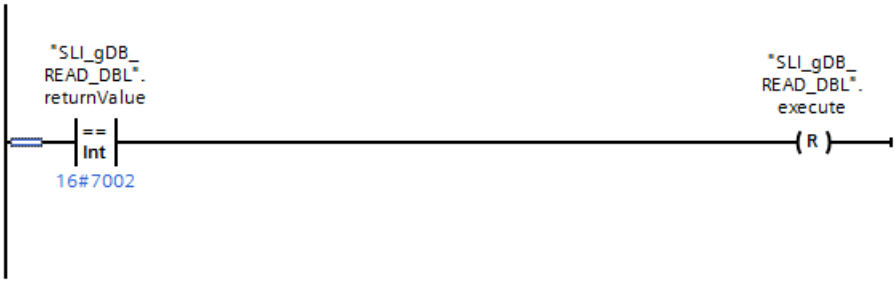

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "READ_DBL" ausgeführt. Über den Eingangsparameter SRCBLK ("SourceDB_LoadMemory") ist der DB im Ladespeicher hinterlegt.
Über den Ausgangsparameter DSTBLK ("TargetDB") ist der DB im Arbeitsspeicher hinterlegt.

Die Anweisung "READ_DBL" ruft die zu übertragenden Daten im Ladespeicher (SRCBLK) auf und liest die Daten im Arbeitsspeicher (DSTBLK) aus.

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1935)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung "WRIT_DBL" übertragen Sie den Inhalt eines DBs oder eines DB-Bereichs aus dem Arbeitsspeicher in einen DB oder einen DB-Bereich im Ladespeicher (Micro-Memory-Card). Der Quell-DB muss ablaufrelevant sein. Er darf nicht mit dem Attribut "Nur im Ladespeicher ablegen" erzeugt worden sein.

Hinweis

Startwerteverhalten beim "Laden von Gerät"

Die Startwerte, die Sie mithilfe der Anweisung "WRIT_DBL" änderten, gehen beim Ausführen der Aktion "Laden von Gerät" wieder verloren.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Quellbereich während der Bearbeitung von "WRIT_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Die Parameter SRCBLK und DSTBLK (Quellbaustein und Zielbaustein) müssen beide vom Datentyp VARIANT sein. Dabei gelten folgende Einschränkungen:

- Bei einem VARIANT-Pointer vom Typ BOOL muss die Länge durch 8 teilbar sein.
- Bei einem VARIANT-Pointer vom Typ STRING muss die Länge gleich 1 sein.
- Quell- und Zielbaustein müssen mit dem gleichen Bausteinzugriff erstellt worden sein. Beide müssen entweder die Zugriffsart "Optimierter Bausteinzugriff" verwenden oder bei beiden muss der optimierte Zugriff deaktiviert sein.
- Wenn der Quell- und der Ziel-DB Bausteine mit optimiertem Zugriff sind, dürfen Sie den Datentyp STRUCT nicht verwenden.

Die Anweisung "WRIT_DBL" verändert nicht die Prüfsumme des Anwenderprogramms, wenn Sie einen mittels Anweisung erzeugten DB beschreiben. Beim Beschreiben eines geladenen

DB hingegen verändert das erste Schreiben auf diesen DB die Prüfsumme des Anwenderprogramms.

Hinweis

"WRIT_DBL" eignet sich nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu schreiben, da auf eine Memory-Card technologiebedingt nur eine bestimmte Anzahl von Schreibzugriffen möglich ist.

Arbeitsweise

Die Anweisung "WRIT_DBL" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "WRIT_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRIT_DBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Schreiben
SRCBLK	Input	VARIANT	D	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.
DSTBLK	Output	VARIANT	D	Zeiger auf den Datenbaustein im Ladespeicher, in den geschrieben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Beschreibung
0000	Kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert.
7000	Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".

Fehlercode* (W#16#...)	Beschreibung
8051	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Typfehler Datenbaustein.
8251	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Datentypfehler im Quell-DB
8551	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Datentypfehler im Ziel-DB
8081	Der Quellbereich ist größer als der Zielbereich. Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.
8082	Hinweis: Diesen Fehlercode gibt es nur bei S7-1500-CPUs. • Ziel-DB und Quell-DB sind von unterschiedlichem Typ (optimierter/nicht optimierter Zugriff). • Ziel- und Quell-DB sind beide mit optimiertem Zugriff erstellt worden. Sie haben bei Ziel- und Quell-DB jedoch Strukturen verwendet, die beim Übersetzen zu einer unterschiedlichen Anordnung der Daten führen. (Hinweis: Verwenden Sie stattdessen UDTs).
8093	Hinweis: Diesen Fehlercode gibt es nur bei S7-1500-CPUs. Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
80B1	Hinweis: Diesen Fehlercode gibt es nur bei S7-1500-CPUs. Am Parameter DSTBLK sind nur Datenbausteine zulässig, die sich im Ladespeicher befinden.
82B1	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Beim Parameter SRCBLK ist kein Datenbaustein angegeben.
85B1	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Beim Parameter DSTBLK ist kein Datenbaustein angegeben oder der dort angegebene Datenbaustein befindet sich nicht im Ladespeicher.
80B4	• Memory Card ist schreibgeschützt. • DB mit F-Attribut darf nicht gelesen werden.
80BB	Nicht ausreichend Ladespeicher verfügbar.
82C0	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Der Quell-DB wird gerade von einer anderern Anweisung oder einer Kommunikations-Anweisung bearbeitet.
85C0	Hinweis: Diesen Fehlercode gibt es nur bei S7-1200-CPUs. Der Ziel-DB wird gerade von einer anderern Anweisung oder einer Kommunikations-Anweisung bearbeitet.
80C3	Die maximale Anzahl gleichzeitig aktiver "WRIT_DBL"-Anweisungen ist momentan bereits erreicht.
allgemeine Fehlercodes	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel übertragen Sie den Inhalt eines DBs im Arbeitsspeichers in einen DB im Ladespeicher. Um die erfolgreiche Übertragung in den DB nachzuweisen, lesen Sie den Inhalt in einem anderen DB im Arbeitsspeicher aus.

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein fünf Variablen an.

SLI_gDB_WRIT_DBL			
	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	returnValue	Int	0
4	busy	Bool	false
5	readRetVal	Int	0
6	readBusy	Bool	false

Zur Ablage der zu übermittelnden Daten legen Sie drei globale Datenbausteine mit jeweils einer Variable "myData" (Datentyp Array of INT) an:

- Einen DB für Sendedaten im Arbeitsspeicher (SLI_srcDB_WRIT_DBL).

SLI_srcDB_WRIT_DBL			
	Name	Data type	Start value
1	Static		
2	myData	Array[0..3] of Int	
3	myData[0]	Int	1
4	myData[1]	Int	2
5	myData[2]	Int	3
6	myData[3]	Int	4

- Einen DB für Empfangsdaten im Ladespeicher (SLI_dstDB_WRIT_DBL).
- Einen DB für Empfangsdaten im Arbeitsspeicher (SLI_finDB_WRIT_DBL).

Netzwerk 1:

Die Parameter der Anweisung "WRIT_DBL" verschalten Sie wie folgt.

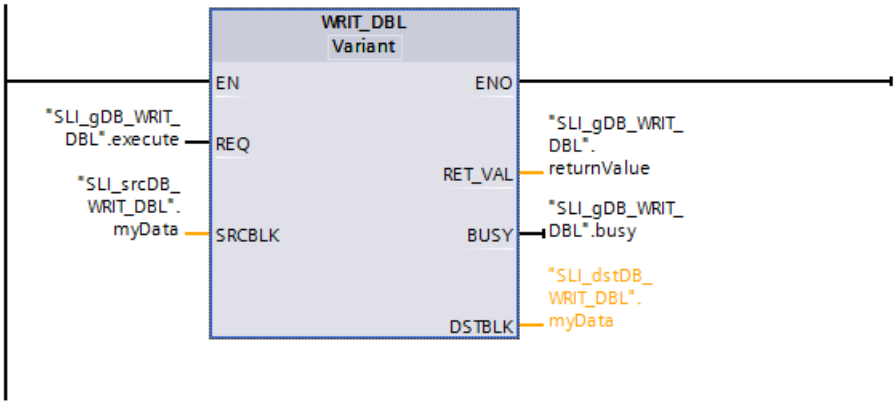

Netzwerk 2:

Die Parameter der Anweisung "READ_DBL" verschalten Sie wie folgt.

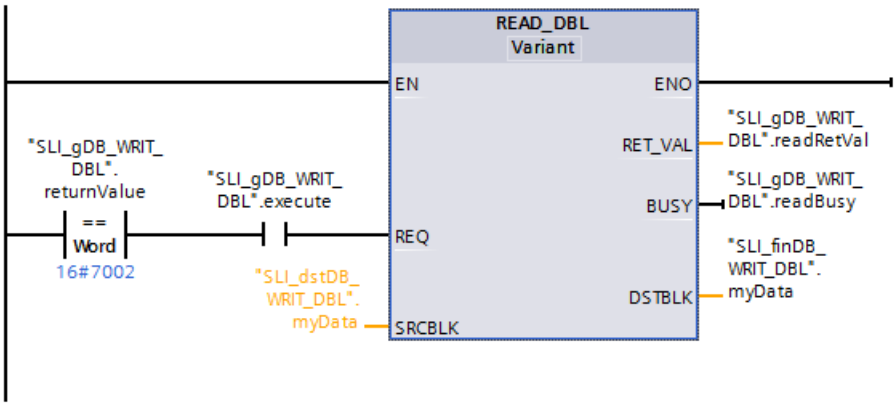

Netzwerk 3:

Die Variable "execute" setzen Sie wie folgt zurück.

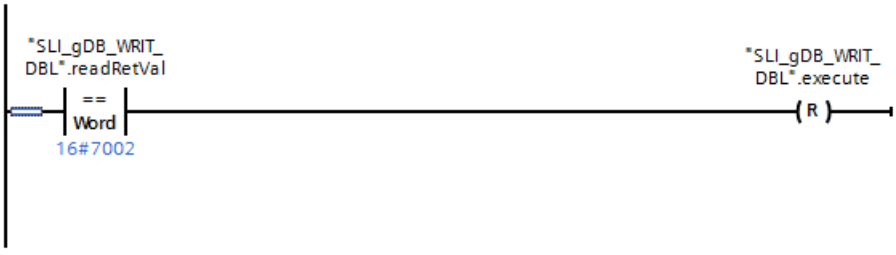

Anweisung "WRIT_DBL":

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "WRIT_DBL" ausgeführt. Über den Eingangsparameter SRCBLK ("SLI_srcDB_WRIT_DBL.myData") ist der DB im Arbeitsspeicher hinterlegt. Über den Ausgangsparameter DSTBLK ("SLI_dstDB_WRIT_DBL.myData") ist der DB im Ladespeicher hinterlegt.

Die Anweisung "WRIT_DBL" ruft die zu übertragenden Daten im Arbeitsspeicher (SRCBLK) auf und schreibt die Daten an gewünschter Stelle in den Ladespeicher (DSTBLK). Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

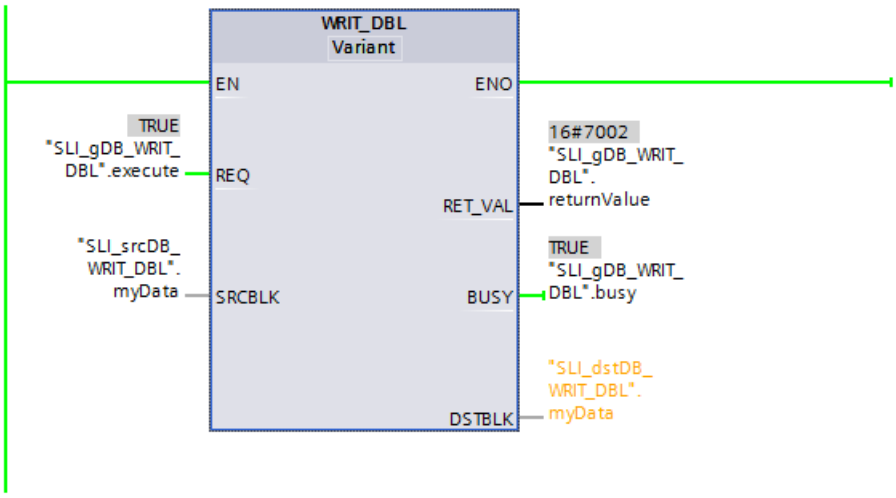

Anweisung "READ_DBL":

Wenn die Verarbeitung von WRIT_DBL ohne Fehler verlief und der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "READ_DBL" ausgeführt. Über den Eingangsparameter SRCBLK ("SLI_dstDB_WRIT_DBL.myData") ist der DB im Ladespeicher hinterlegt. Über den Ausgangsparameter DSTBLK ("SLI_finDB_WRIT_DBL.myData") ist der DB für die Empfangsdaten hinterlegt.

Die Anweisung "READ_DBL"ruft die zu übertragenden Daten im Ladespeicher (SRCBLK) auf und liest die Daten an gewünschter Stelle im Arbeitsspeicher (DSTBLK) aus. Am Ausgangsparameter RET_VAL ("readRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

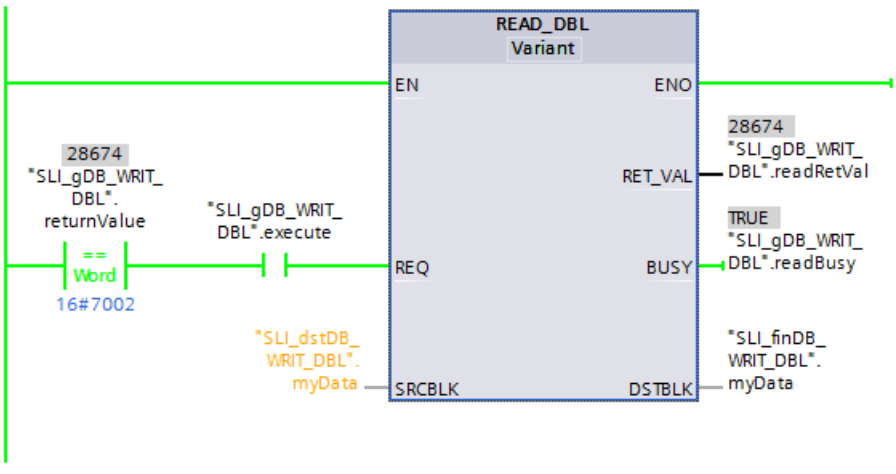

Der DB für die Empfangsdaten im Arbeitsspeicher (SLI_finDB_WRIT_DBL) zeigt die übermittelten Daten an.

SLI_finDB_WRIT_DBL

	Name	Data type	Start value	Monitor value
1	Static			
2	myData	Array[0..3] o...		
3	myData[0]	Int	0	1
4	myData[1]	Int	0	2
5	myData[2]	Int	0	3
6	myData[3]	Int	0	4

Nach erfolgreicher Verarbeitung von READ_DBL wird die Variable "execute" zurückgesetzt (siehe Netzwerk 3). Die Verarbeitungen von READ_DBL und WRIT_DBL werden gestoppt.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

ATTR_DB: Attribute eines Datenbausteins lesen

Beschreibung

Mit der Anweisung "ATTR_DB" erhalten Sie Informationen über einen im Arbeitsspeicher der CPU liegenden Datenbaustein (DB). Die Anweisung ermittelt für den ausgewählten DB die gesetzten Attribute am Parameter ATTRIB.

Für Datenbausteine mit optimiertem Zugriff und Datenbausteine, die nur im Ladespeicher liegen, kann die Länge nicht ausgelesen werden. In diesen Fällen enthält der Parameter DB_LENGTH den Wert "0".

Wenden Sie ATTR_DB nicht auf Datenbausteine mit optimiertem Zugriff und aktivierter Speicherreserve an.

Lesen Sie die Datenbausteine für Motion Control nicht mit der Anweisung "ATTR_DB" aus. Für diese wird der Fehlercode 80B2 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATTR_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ = 1: Anforderung zum Lesen der Bausteinattribute
DB_NUMBER	Input	DB_ANY	E, A, M, D, L oder Konstante	Nummer des zu überprüfenden DB
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB_LENGTH	Output	UDINT	E, A, M, D, L	• Anzahl der Datenbytes, die der ausgewählte DB enthält • "0" für Datenbausteine mit optimiertem Zugriff und Datenbausteine, die nur im Ladespeicher liegen
ATTRIB	Output	BYTE	E, A, M, D, L	DB-Eigenschaften:
				• Bit 0* = 0: Attribut "Nur im Ladespeicher ablegen" nicht gesetzt. • Bit 0* = 1: Attribut "Nur im Ladespeicher ablegen" gesetzt.
				• Bit 1 = 0: Attribut "Datenbaustein im Gerät schreibgeschützt" nicht gesetzt. • Bit 1 = 1: Attribut "Datenbaustein im Gerät schreibgeschützt" gesetzt.
				Wenn Bit 0 = 1, dann ist Bit 2 irrelevant und erhält den Wert 1. • Bit 2 = 0: Remanent - Wenn mindestens ein Wert remanent gesetzt wurde, gilt der DB gilt als remanent. • Bit 2 = 1: Nicht remanent - Der komplette DB ist nicht remanent.
				• Bit 3*= 0: Der DB ist entweder im Ladespeicher (Bit 0 =1) oder im Arbeitsspeicher (Bit 0 = 0). • Bit 3*= 1: Der DB ist sowohl im Lade- als auch im Arbeitsspeicher.
* Der Zusammenhang zwischen Bit 0 und Bit 3 ist in den Parametern der Anweisung "CREATE_DB: Datenbaustein erzeugen (Seite 2821)"erläutert.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
80A1	Fehler beim Eingangsparameter DB_NUMBER: Der gewählte Aktualparameter • hat den Wert "0" • ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.
80B1	Der DB mit der angegebenen Nummer ist auf der CPU nicht vorhanden.
80B2	Auf DBs der Technologieobjekte Motion Control kann mit der Anweisung "ATTR_DB" nicht zugegriffen werden.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel lesen Sie den Status eines Datenbausteins aus.

Sie legen einen DB mit einer Variable "myData" (Datentyp Array[0..3] of INT) an.

SLI_srcDB_ATTR_DB

	Name	Data type	Offset	Start value
1	Static			
2	myData	Array[0..3] of Int	0.0	
3	myData[0]	Int	0.0	1
4	myData[1]	Int	2.0	2
5	myData[2]	Int	4.0	3
6	myData[3]	Int	6.0	4

Für Standardzugriff deaktivieren Sie in den Eigenschaften des DBs den optimierten Datenzugriff.

Attributes

- [ ] Only store in load memory
- [ ] Data block write-protected in the device
- [ ] Optimized block access

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein vier Variablen an.

SLI_gDB_ATTR_DB

	Name	Data type	Start value
1	Static		
2	startAttr	Bool	false
3	attrRetVal	Int	0
4	readLenByte	UDInt	0
5	readAttrBool	Array[0..7] of Bool	

Parameter verschalten: FC "SLI_itemiseByte_ATTR_DB"

Um den Wert des Ausgangsparameters ATTRIB besser auswerten zu können, legen Sie den FC "SLI_itemiseByte_ATTR_DB" an.

Darin legen Sie die folgende lokale Variable an.

SLI_itemiseByte_ATTR_DB

	Name	Data type
1	Input	
2	inByte	Byte

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) der lokalen Variable Verschaltungen an.

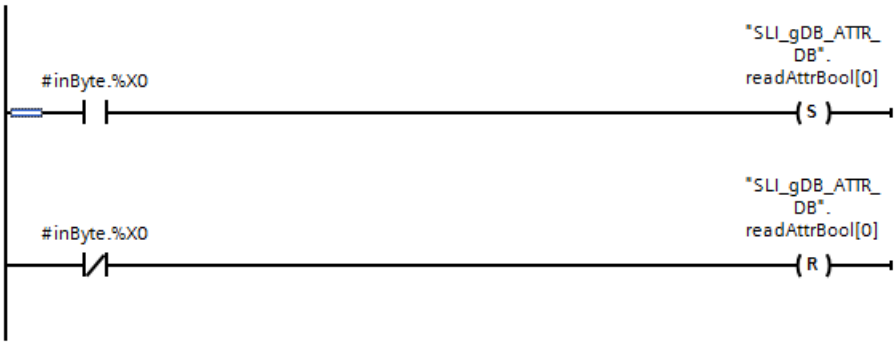

Parameter verschalten: FB "SLI_FB_ATTR_DB"

Sie legen einen FB "SLI_FB_ATTR_DB" für den Aufruf von ATTR_DB an. Darin legen Sie die folgende lokale Variable an.

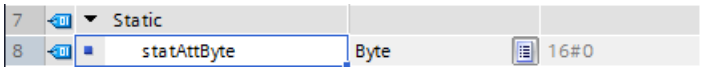

Netzwerk 1: Die Parameter der Anweisung "ATTR_DB" verschalten Sie wie folgt.

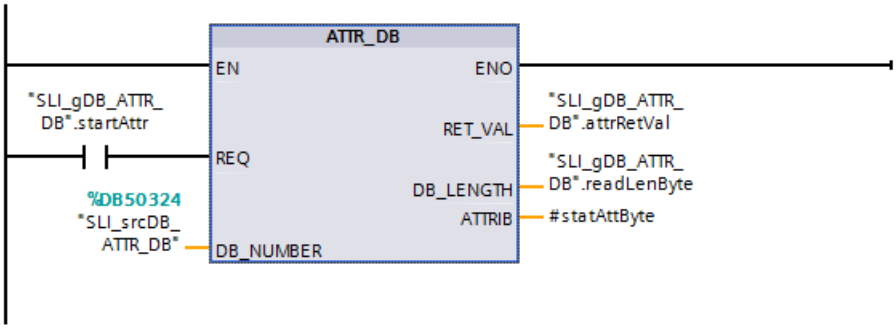

Netzwerk 2: Den FC "SLI_itemiseByte_ATTR_DB" verschalten Sie wie folgt.

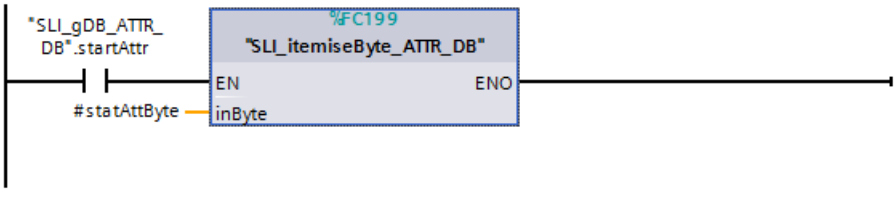

Ergebnis von FB "SLI_FB_ATTR_DB"

Wenn der Schließerkontakt ("startAttr") den Signalzustand "TRUE" liefert, wird die Anweisung "ATTR_DB" ausgeführt. Über den Eingangsparameter DB_NUMBER ("SLI_srcDB_ATTR_DB") ist die Nummer des auszuwertenden DBs hinterlegt. Die Anweisung "ATTR_DB" ruft den DB anhand der Nummer auf und gibt dessen Status am Ausgangsparameter ATTRIB ("#statAttrByte") aus. Am Ausgangsparameter DB_LENGTH ("readLenByte") wird die Länge der ausgelesenen Daten (4 INT) des Datenbausteins in BYTE (Wert "8") ausgegeben.

Der hexadezimale Wert des Ausgangsparameters ATTRIB wird über den FC "SLI_itemiseByte_ATTR_DB" in eine Bitfolge umgerechnet. Die Bitfolge "00110000" (Bit 0..7) bedeutet:

- Der DB befindet sich nicht nur im Ladespeicher.
- Der DB ist nicht schreibgeschützt.
- Der DB ist nicht remanent.
- Der DB ist sowohl im Lade- als auch im Arbeitsspeicher generiert.

Am Ausgangsparameter RET_VAL ("attrRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_ATTR_DB

	Name	Data type	Start value	Monitor value
1	Static			
2	startAttr	Bool	false	TRUE
3	attrRetVal	Int	0	0
4	readLenByte	UDInt	0	8
5	readAttrBool	Array[0..7] of Bool		
6	readAttrBool[0]	Bool	false	FALSE
7	readAttrBool[1]	Bool	false	FALSE
8	readAttrBool[2]	Bool	false	TRUE
9	readAttrBool[3]	Bool	false	TRUE
10	readAttrBool[4]	Bool	false	FALSE
11	readAttrBool[5]	Bool	false	FALSE
12	readAttrBool[6]	Bool	false	FALSE
13	readAttrBool[7]	Bool	false	FALSE

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

DELETE_DB: Datenbaustein löschen

Beschreibung

Mit der Anweisung "DELETE_DB" löschen Sie einen Datenbaustein (DB), der aus dem Anwenderprogramm durch Aufruf der Anweisung "CREATE_DB (Seite 2821)" erzeugt wurde.

Wenn der Datenbaustein nicht durch "CREATE_DB" erzeugt wurde, wird am Parameter RET_VAL der Fehlercode W#16#80B5 ausgegeben.

Der ausgewählte Datenbaustein wird nicht sofort gelöscht, sondern erst im Zykluskontrollpunkt nach Ausführung des Zyklus-OBs.

Arbeitsweise

Die Anweisung "DELETE_DB" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Alarmübertragung, indem Sie Anweisung mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die Bytes 2 und 3 des Ausgangsparameters RET_VAL wird der Zustand des Auftrags angezeigt.

Wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat, ist das Löschen des Datenbausteins abgeschlossen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DELETE_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	REQ =1: Anforderung zum Löschen des DB mit der Nummer am Parameter DB_NUMBER
DB_NUMBER	Input	UINT	E, A, M, D, L oder Konstante	Nummer des zu löschenden DBs
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation (siehe "Parameter RET_VAL")
BUSY	Output	BOOL	E, A, M, D, L	BUSY =1: Der Vorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
7000	Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
80A1	Fehler beim Eingangsparameter DB_NUMBER: • Der Wert am Parameter ist "0". • Der Wert am Parameter ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.
80B1	Der DB mit der angegebenen Nummer ist auf der CPU nicht vorhanden.
80B4	Der DB kann nicht gelöscht werden, da die Memory-Card der CPU schreibgeschützt ist.
80B5	Der DB wurde nicht über "CREATE_DB" erzeugt.
80BB	Nicht genügend Ladespeicher frei.
80C3	Die Funktion "Löschen eines DB" ist wegen temporären Ressourcenengpasses momentan nicht durchführbar.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel zu CREATE Funktionen (Seite 2841).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel zu CREATE Funktionen

Einleitung

In dem folgenden Beispiel erstellen Sie einen DB im Ladespeicher, lesen dessen Inhalt aus und löschen den DB anschließend wieder.

Voraussetzung

Zur Vorlage für den neuen DB legen Sie einen globalen DB ("SLI_srcDB_CREATE_DB") an.

SLI_srcDB_CREATE_DB

	Name	Data type	Offset	Start value
1	Static			
2	myData	Array[0..3] of Int	0.0	
3	myData[0]	Int	0.0	1
4	myData[1]	Int	2.0	2
5	myData[2]	Int	4.0	3
6	myData[3]	Int	6.0	4
7	readData	Array[0..3] of Int	8.0	
8	readData[0]	Int	0.0	0
9	readData[1]	Int	2.0	0
10	readData[2]	Int	4.0	0
11	readData[3]	Int	6.0	0

Die Eigenschaften des DBs sind wie folgt.

Attributes

- [ ] Only store in load memory
- [ ] Data block write-protected in the device
- [ ] Optimized block access

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein ("SLI_gDB_CREATE_DB") 14 Variablen an.

SLI_gDB_CREATE_DB			
	Name	Data type	Start value
1	Static		
2	startCreate	Bool	false
3	lowLim	UInt	60003
4	hiLim	UInt	60003
5	setLenByte	UDInt	8
6	setAttrBool	Array[0..7] of Bool	
7	setAttrBool[0]	Bool	1
8	setAttrBool[1]	Bool	0
9	setAttrBool[2]	Bool	1
10	setAttrBool[3]	Bool	0
11	setAttrBool[4]	Bool	1
12	setAttrBool[5]	Bool	false
13	setAttrBool[6]	Bool	false
14	setAttrBool[7]	Bool	false
15	createRetVal	Int	0
16	createBusy	Bool	false
17	memNbrNewDB	UInt	0
18	startRead	Bool	false
19	rdblRetVal	Int	0
20	rdblBusy	Bool	false
21	startDelete	Bool	false
22	delRetVal	Int	0
23	delBusy	Bool	false

Zur Ablage der Daten legen Sie zudem folgende lokale Variablen in dem FB "SLI_FB_CREATE_DB" an.

7	Static			
8	statAttByte	Byte	16#0	Non-ret...
9	nbrNewDB	UInt	0	Non-retain
10	Temp			
11	tempCreate	Bool		
12	tempRead	Bool		
13	tempDelete	Bool		

Parameter verschalten: FC "SLI_itemiseByte_CRDfunctions"

Zur Umwandlung der Werte der Variable "setAttrBool" in die Variable "#statAttByte", legen Sie den FC "SLI_itemiseByte_CRDfunctions" an.

Darin legen Sie die folgende lokale Variable an.

3	Output	
4	outByte	Byte

Netzwerk 1: Nach Vorbild des Netzwerk 1 legen Sie für alle Bits (0..7) der lokalen Variable Verschaltungen an.

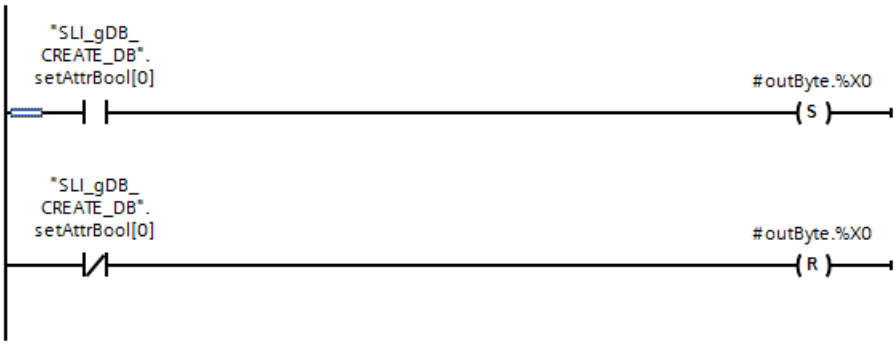

Parameter verschalten: FB "SLI_FB_CREATE_DB"

Netzwerk 1: Die Parameter des FC "SLI_itemiseByte_CRDfunctions" verschalten Sie wie folgt.

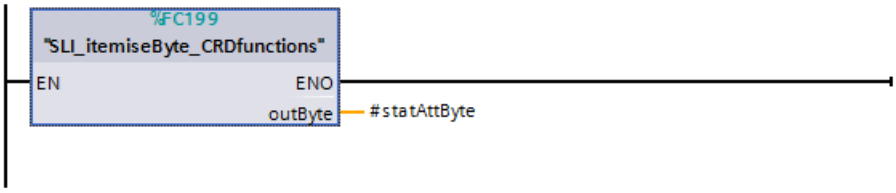

Netzwerk 2: Die Parameter der Anweisung "CREATE_DB" verschalten Sie wie folgt.

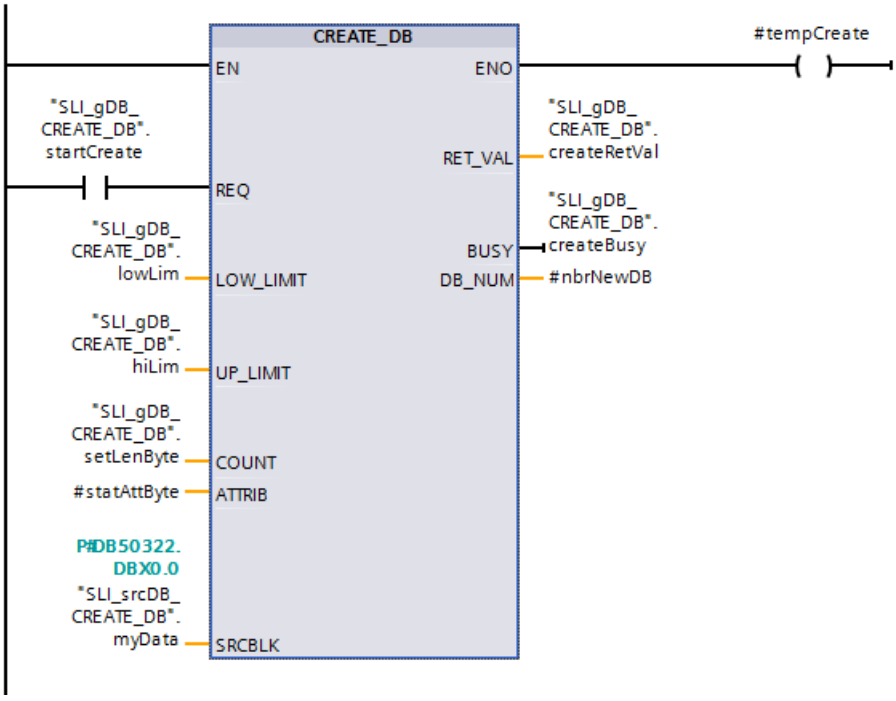

Netzwerk 3: Um die Nummer des neu erstellten DBs zu speichern, legen Sie folgende Verschaltungen an.

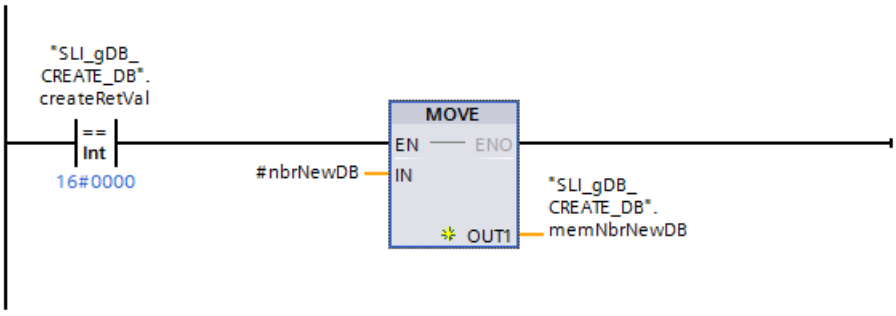

Netzwerk 4: Um die Ausführung von CREATE_DB wieder zu beenden, legen Sie folgende Verschaltungen an.

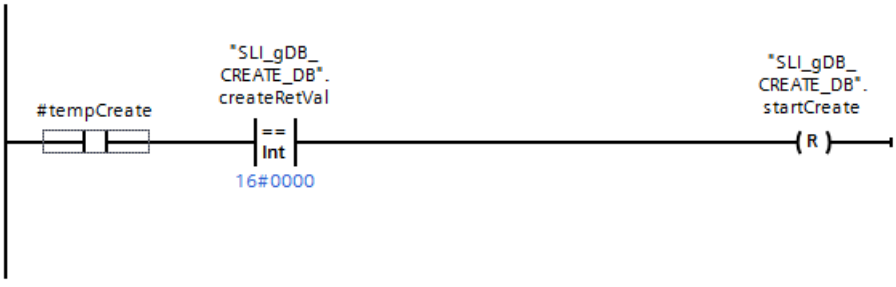

Netzwerk 5: Die Parameter der Anweisung "READ_DBL" verschalten Sie wie folgt.

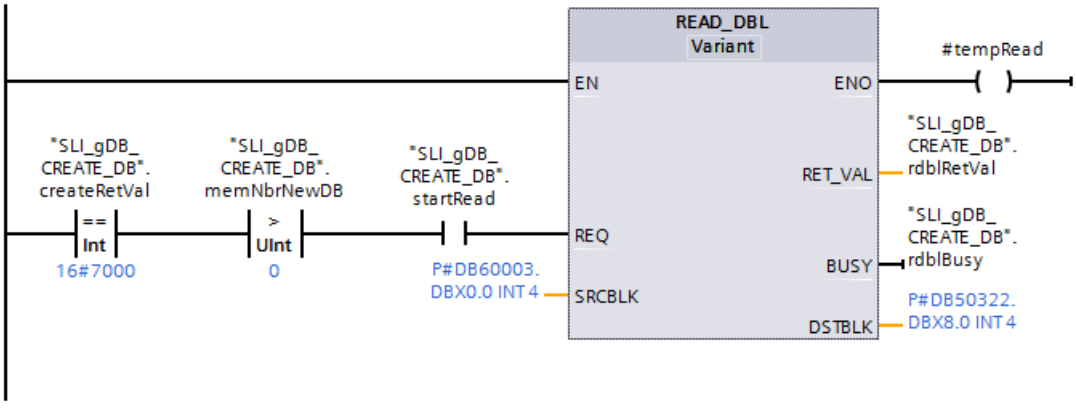

Netzwerk 6: Um die Ausführung von READ_DBL wieder zu beenden, legen Sie folgende Verschaltungen an.

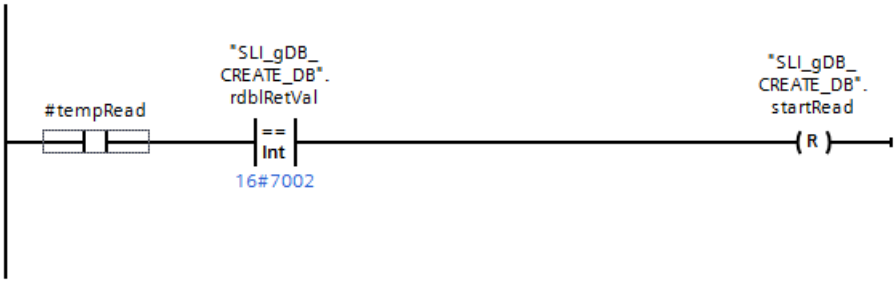

Netzwerk 7: Die Parameter der Anweisung "DELETE_DB" verschalten Sie wie folgt.

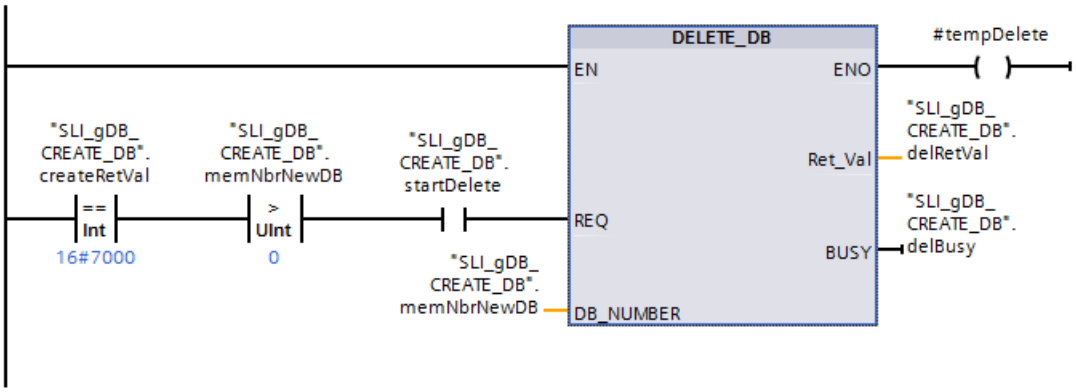

Netzwerk 8: Um die Ausführung von DELETE_DB wieder zu beenden, legen Sie folgende Verschaltungen an.

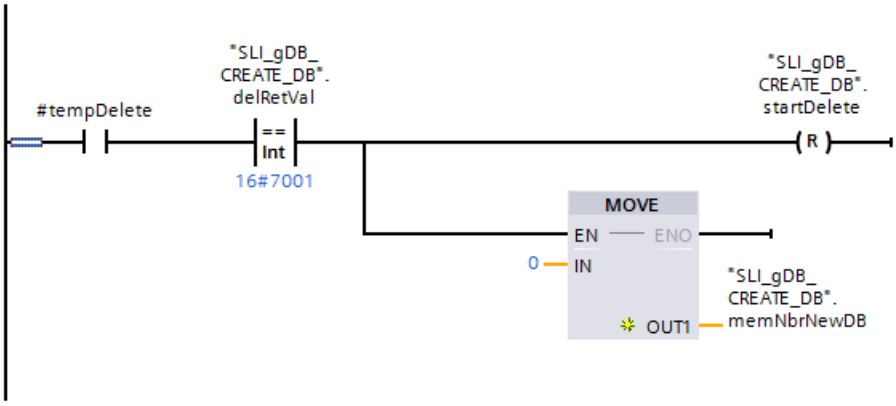

Ergebnis von Netzwerk 1

Durch ein AT-Konstrukt werden die Eigenschaften ("setAttrBool") im Datentyp Array of BOOL (#statAttBool) in BYTE (#statAttByte) umgewandelt, abgelegt.

Ergebnis von Netzwerk 2 - Anweisung "CREATE_DB"

Wenn der Schließerkontakt ("startCreate") den Signalzustand "TRUE" liefert, wird die Anweisung "CREATE_DB" ausgeführt. Gemäß dem Eingangsparameter COUNT ("setLenByte") wird dabei ein neuer DB mit einer Länge von 8 Byte erstellt. Durch den Eingangsparameter LOW_LIMIT ("lowLim") bzw. UP_LIMIT ("hiLim") ist die Nummer des zu erstellenden DBs vorgegeben. Am Eingangsparameter ATTR ("#statAttByte") sind die Eigenschaften (in BYTE) hinterlegt, mit denen der DB erstellt werden soll. So soll der DB nur im Ladespeicher erstellt und mit einem definierten Inhalt befüllt werden. Als Vorlage für den Inhalt des neuen DBs ist am Eingangsparameter SRCBLK ("SLI_srcDB_CREATE_DB.myData") ein DB hinterlegt.

Am Ausgangsparameter RET_VAL ("CreateRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Die Anweisung "CREATE_DB" verbleibt nach seiner Ausführung inaktiv ("createRetVal" hat den Wert 16#7000"). Am Ausgangsparameter DB_NUM ("#nbrNewDB") wird zum Ende der Ausführung die Nummer des neuen DBs kurze Zeit ausgegeben.

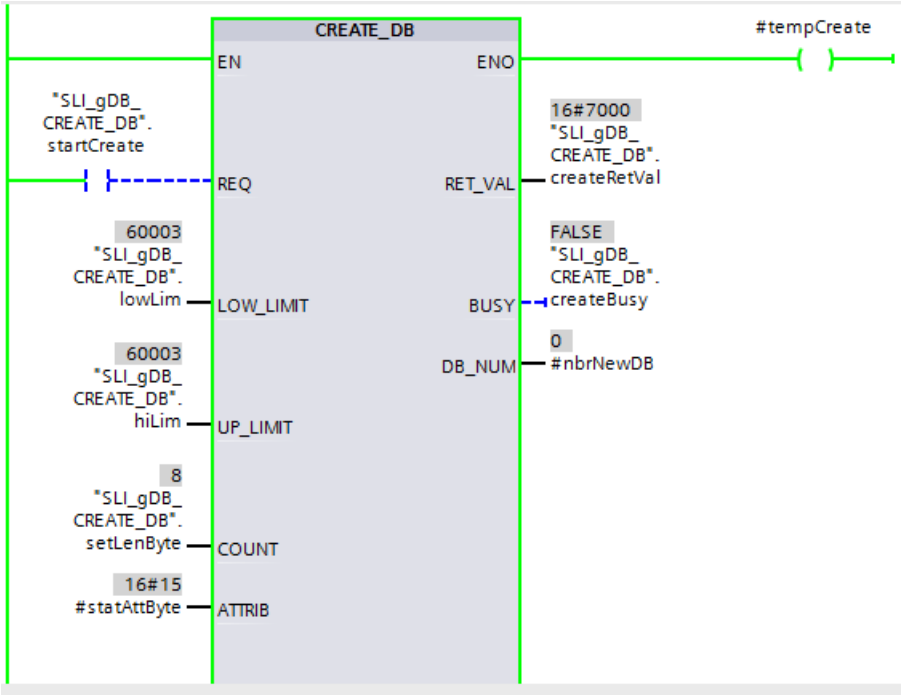

Ergebnis von Netzwerk 3

Wenn die Verarbeitung von CREATE_DB ohne Fehler verlief, hat "createRetVal" den Wert "16#0000". Als Folge wird die Nummer ("#nbrNewDB") des DBs in der Variable "memNbrNewDB" gespeichert.

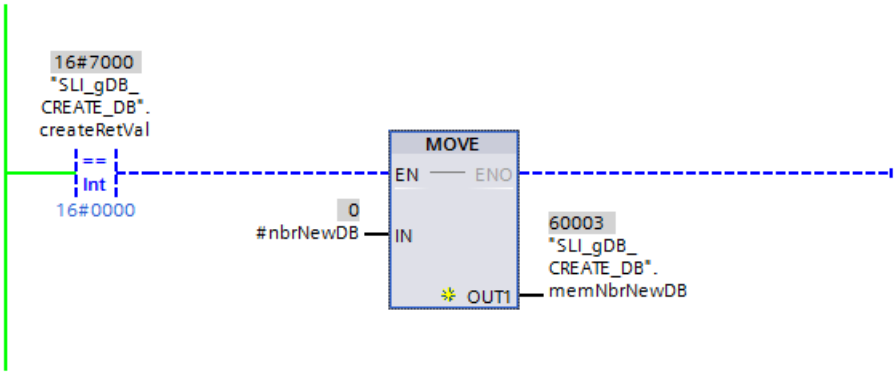

Ergebnis von Netzwerk 4

Wenn CREATE_DB am Parameter ENO ("#tempCreate") den Signalzustand "TRUE" liefert, ist die Verschaltung des Netzwerk 4 aktiv. Wenn zusätzlich die Verarbeitung von CREATE_DB ohne Fehler verlief, wird der Rücksetzer ("startCreate") auf "TRUE" gesetzt und die Ausführung von CREATE_DB beendet.

SLI_gDB_CREATE_DB

	Name	Data type	Start value	Monitor value
1	Static			
2	startCreate	Bool	false	FALSE
3	lowLim	UInt	60003	60003
4	hiLim	UInt	60003	60003
5	setLenByte	UDInt	8	8
6	setAttrBool	Array[0..7] of Bool		
7	createRetVal	Int	0	28672
8	createBusy	Bool	false	FALSE
9	memNbrNewDB	UInt	0	60003
10	startRead	Bool	false	FALSE
11	rdblRetVal	Int	0	28672
12	rdblBusy	Bool	false	FALSE
13	startDelete	Bool	false	FALSE
14	delRetVal	Int	0	28672
15	delBusy	Bool	false	FALSE

Ergebnis von Netzwerk 5 - Anweisung "READ_DBL"

Zur Ausführung von READ_DBL gibt es folgende Vorbedingungen:

- CREATE_DB ist inaktiv ("createRetVal" hat den Wert "16#7000").
- CREATE_DB hat einen neuen DB erzeugt, dessen Nummer gespeichert wurde ("memNbrNewDB" hat einen Wert > 0).

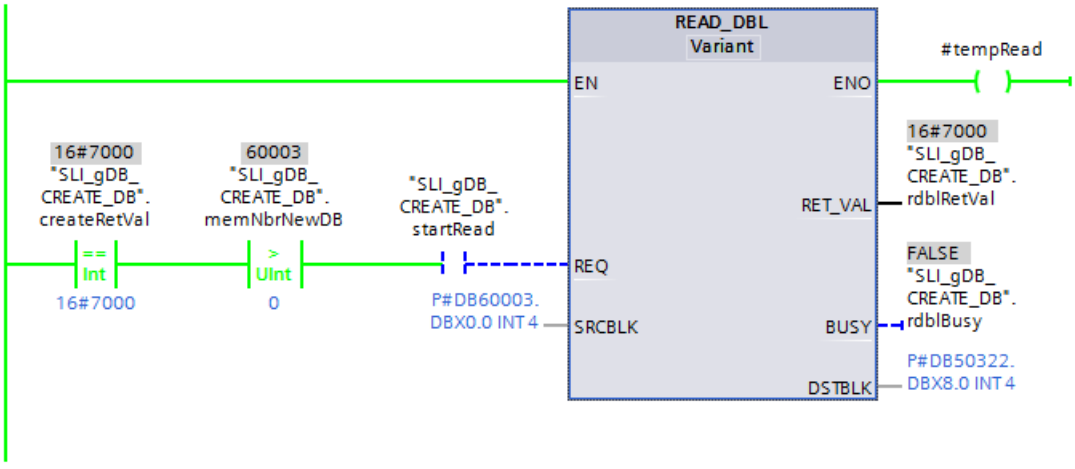

Wenn der Schließerkontakt ("startRead") den Signalzustand "TRUE" liefert, wird die Anweisung "READ_DBL" ausgeführt. Über den Eingangsparameter SCRBLK ist der neu erstellte DB im Ladespeicher referenziert. Über den Ausgangsparameter DSTBLK ist der DB im Arbeitsspeicher ("SLI_srcDB_CREATE_DB.readData") referenziert. Die Anweisung

"READ_DBL" ruft die zu übertragenden Daten im Ladespeicher (SCRBLK) auf und liest die Daten an gewünschter Stelle im Arbeitsspeicher (DSTBLK) aus.

Am Ausgangsparameter RET_VAL ("rdblRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief. Beim DB im Arbeitsspeicher ("SLI_srcDB_CREATE_DB.readData") werden die übertragenen Daten angezeigt.

SLI_srcDB_CREATE_DB

	Name	Data type	Offset	Start value	Monitor value
1	Static				
2	myData	Array[0..3] of Int	0.0		
3	myData[0]	Int	0.0	1	1
4	myData[1]	Int	2.0	2	2
5	myData[2]	Int	4.0	3	3
6	myData[3]	Int	6.0	4	4
7	readData	Array[0..3] of Int	8.0		
8	readData[0]	Int	0.0	0	1
9	readData[1]	Int	2.0	0	2
10	readData[2]	Int	4.0	0	3
11	readData[3]	Int	6.0	0	4

Ergebnis von Netzwerk 6

Wenn READ_DBL am Parameter ENO ("#tempRead") den Signalzustand "TRUE" liefert, ist die Verschaltung des Netzwerk 6 aktiv. Wenn zusätzlich die Verarbeitung von READ_DBL abgeschlossen ist, wird der Rücksetzer ("startRead") auf "TRUE" gesetzt und die Ausführung von READ_DBL beendet.

Ergebnis von Netzwerk 7 - Anweisung "DELETE_DB"

Zur Ausführung von DELETE_DB gibt es folgende Vorbedingungen:

- CREATE_DB ist inaktiv ("createRetVal" hat den Wert "16#7000").
- CREATE_DB hat einen neuen DB erzeugt, dessen Nummer gespeichert wurde ("memNbrNewDB" hat einen Wert > 0).

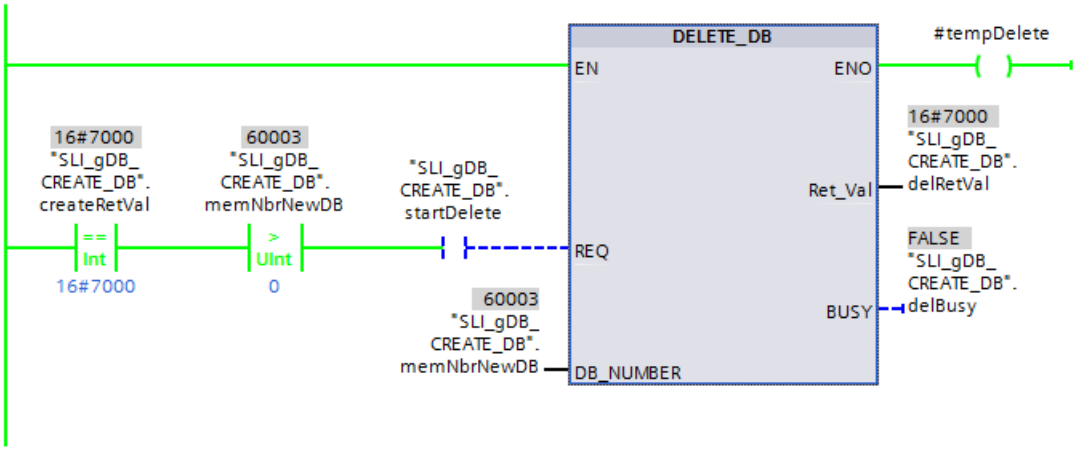

Wenn der Schließerkontakt ("startDelete") den Signalzustand "TRUE" liefert, wird die Anweisung "DELETE_DB" ausgeführt. Über den Eingangsparameter DB_NUMBER ("memNbrNewDB") ist der neu erstellte DB anhand seiner Nummer referenziert. Die Anweisung "DELETE_DB" löscht den neu erstellten DB.

Am Ausgangsparameter RET_VAL ("deleteRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

Ergebnis von Netzwerk 8

Wenn DELETE_DB am Parameter ENO ("#tempRead") den Signalzustand "TRUE" liefert, ist die Verschaltung des Netzwerk 8 aktiv. Wenn zusätzlich die Verarbeitung von DELETE_DB abgeschlossen ist, wird der Rücksetzer ("startDelete") auf "TRUE" gesetzt und die Ausführung von DELETE_DB beendet.

Nach der Ausführung von DELETE_DB ist der zuvor erstellte Baustein gelöscht. Um zu verhindern, dass die Löschung eines bereits gelöschten Bausteins befohlen werden kann, wird zudem die Variable "memNbrNewDB" auf "0" gesetzt.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

4.1.4.13 Adressierung

Anweisungen zur Adressumwandlung

Beschreibung

Es bestehen unterschiedliche Möglichkeiten zur Adressierung einer Baugruppe (EA-Adresse, Hardware-Kennung, Steckplatz).

Die Adressdaten können Sie mit den folgenden Anweisungen umwandeln:

- GEO2LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln (Seite 2850)
- LOG2GEO: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2853)
- LOG2MOD: Aus der Adressierung von STEP 7 V5.5 SPx die Hardware-Kennung ermitteln (Seite 2855)
- IO2MOD: Aus einer EA-Adresse die Hardware-Kennung ermitteln (Seite 2856)
- RD_ADDR: Aus der Hardware-Kennung die EA-Adressen ermitteln (Seite 2859)

Für migrierte Projekte werden darüber hinaus die folgenden Anweisungen unterstützt:

- GEO_LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln (Seite 2862)
- LOG_GEO: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2864)
- RD_LGADR: Aus der Hardware-Kennung die EA-Adressen ermitteln (Seite 2865)

- GADR_LGC: Aus dem Steckplatz und dem Offset im Nutzdatenadressraum die Hardware-Kennung ermitteln (Seite 2867)
- LGC_GADR: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2869)

Art der Adressumwandlung

Das folgende Bild zeigt, welche Anweisung welche Adressumwandlung durchführt.

Name	Typ	EA-Adresse(n)	HW-Kennung	Steckplatz
GEO2LOG	SFC			
LOG2GEO	SFC			
LOG2MOD	SFC			
IO2MOD	SFC			
RD_ADDR	SFC			
GEO_LOG	FC			
LOG_GEO	FC			
RD_LGADR	FC			
GADR_LGC	FC			
LGC_GADR	FC			

GEO2LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln

Beschreibung

Mit der Anweisung "GEO2LOG" ermitteln Sie die Hardware-Kennung anhand der Steckplatzinformationen, welche Sie über den Systemdatentyp GEOADDR definieren.

Abhängig von dem Hardwaretyp, den Sie am Parameter HWTYPE definieren, werden folgende Informationen aus den übrigen Parametern von GEOADDR ausgewertet:

- Bei HWTYPE = 1 (IO-System):
  - Es wird nur IOSYSTEM ausgewertet. Die übrigen Parameter von GEOADDR werden nicht berücksichtigt.
  - Die Hardware-Kennung des IO-Systems wird ausgegeben.
- Bei HWTYPE = 2 (IO-Device):
  - Es werden IOSYSTEM und STATION ausgewertet. Die übrigen Parameter von GEOADDR werden nicht berücksichtigt.
  - Die Hardware-Kennung des IO-Device wird ausgegeben.

- Bei HWTYPE = 4 (Modul):
  - Es werden IOSYSTEM, STATION und SLOT ausgewertet. Der Parameter SUBSLOT von GEOADDR wird nicht berücksichtigt.
  - Die Hardware-Kennung des Moduls wird ausgegeben.
- Bei HWTYPE = 5 (Submodul):
  - Es werden alle Parameter von GEOADDR ausgewertet.
  - Die Hardware-Kennung des Submoduls wird ausgegeben.

Der Parameter AREA des Systemdatentyps GEOADDR wird nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEO2LOG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
GEOADDR	Input	VARIANT	D, L	Zeiger auf die Struktur des Systemdatentyps GEOADDR. Der Systemdatentyp enthält die Steckplatzinformationen, aus denen die Hardware-Kennung ermittelt wird. Siehe auch: Systemdatentyp GEOADDR (Seite 2861)
RET_VAL	Return	INT	E, A, M, D, L	Ausgabe der Fehlerinformation.
LADDR	Output	HW_ANY	E, A, M, D, L	Hardware-Kennung der Baugruppe oder des Moduls. Die Nummer wird automatisch vergeben und ist in den Eigenschaften in der Hardware-Konfiguration hinterlegt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten.
8091	Ungültiger Wert in GEOADDR für HWTYPE.
8094	Ungültiger Wert in GEOADDR für IOSYSTEM.
8095	Ungültiger Wert in GEOADDR für STATION.
8096	Ungültiger Wert in GEOADDR für SLOT.
8097	Ungültiger Wert in GEOADDR für SUBSLOT.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ermitteln Sie die Hardware-Kennung der CPU anhand der Steckplatzinformationen.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

SLI_gDB_GEO2LOG

	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	geoAddr	GEOADDR	
4	HWTYPE	UInt	4
5	AREA	UInt	0
6	IOSYSTEM	UInt	0
7	STATION	UInt	0
8	SLOT	UInt	1
9	SUBSLOT	UInt	0
10	returnValue	Int	0
11	laddr	HW_ANY	16#0

Die Parameter der Anweisung verschalten Sie wie folgt.

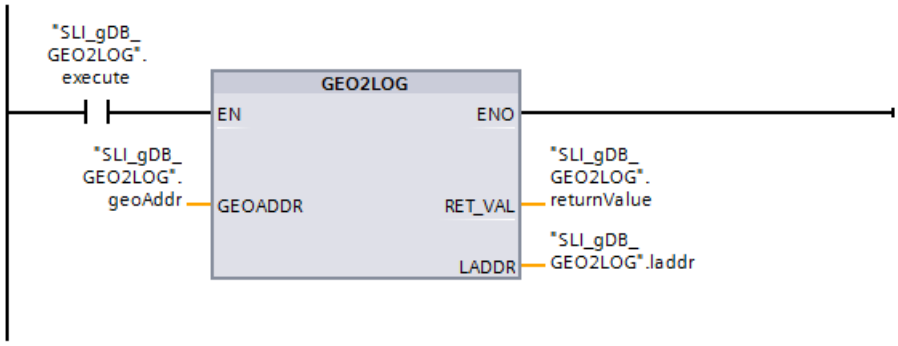

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "GEO2LOG" ausgeführt. Über den Eingangsparameter GEOADDR ("geoAddr") sind die Steckplatzinformationen hinterlegt. Die Anweisung "GEO2LOG" ruft die Hardware-Kennung für ein Modul (HWTYPE mit dem Wert "4") des Zentralgeräts (IOSYSTEM mit dem Wert "0") im Baugruppenträger "0" (Wert von STATION), im Steckplatz "1" (Wert von SLOT) auf.

Die ermittelte Hardware-Kennung der lokalen Komponente der CPU wird am Ausgangsparameter LADDR ("laddr") angezeigt. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

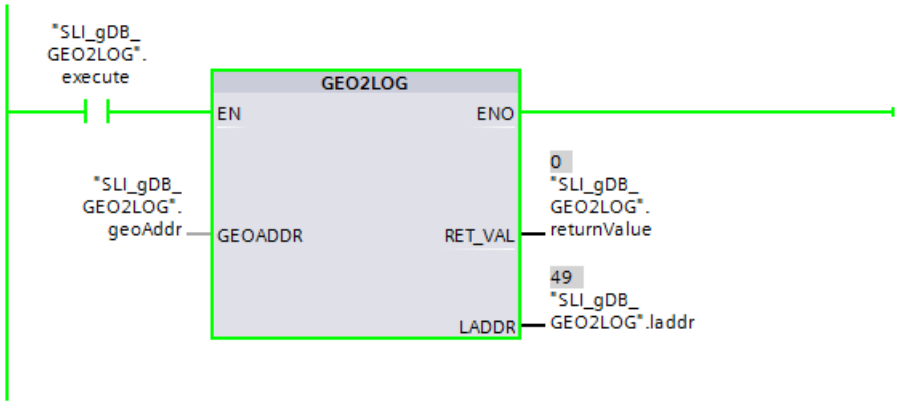

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

LOG2GEO: Aus der Hardware-Kennung den Steckplatz ermitteln

Beschreibung

Mit der Anweisung "LOG2GEO" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG2GEO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_ANY	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe, deren Steckplatz Sie ermitteln möchten. Die Hardware-Kennung wird automatisch vergeben und ist in den Eigenschaften der Baugruppe in der Hardware-Konfiguration und in den Systemkonstanten hinterlegt.
RET_VAL	Output	INT	E, A, M, D, L	Ausgabe der Fehlerinformation.
GEOADDR	InOut	VARIANT	D, L	Zeiger auf den Systemdatentyp GEOADDR. In dem Systemdatentyp GEOADDR werden die Steckplatzinformationen geschrieben. Siehe auch: Systemdatentyp GEOADDR (Seite 2861)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten.
8090	Die am Parameter LADDR angegeben Adresse ist ungültig.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ermitteln Sie die Steckplatzinformationen einer Eingabebaugruppe anhand der Hardware-Kennung.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein vier Variablen an.

SLI_gDB_LOG2GEO

	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	laddr	HW_ANY	257
4	geoAddr	GEOADDR	
5	HWTYPE	UInt	0
6	AREA	UInt	0
7	IOSYSTEM	UInt	0
8	STATION	UInt	0
9	SLOT	UInt	0
10	SUBSLOT	UInt	0
11	returnValue	Int	0

Die Parameter der Anweisung verschalten Sie wie folgt.

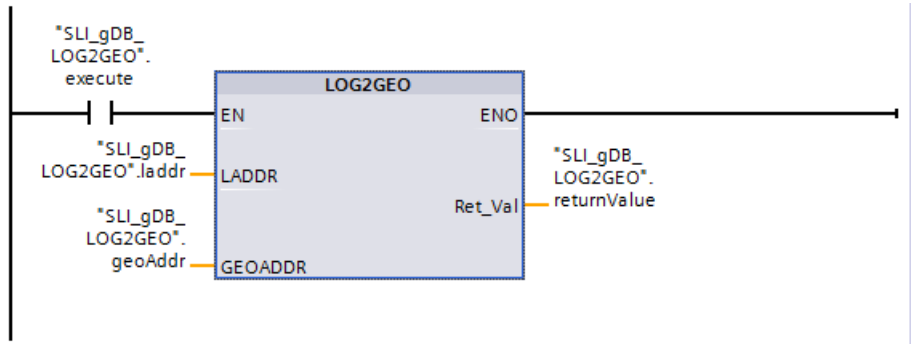

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "LOG2GEO" ausgeführt. Über den Eingangsparameter LADDR ("laddr") ist die Hardware-Kennung der Eingabebaugruppe hinterlegt. Die Anweisung "LOG2GEO" ruft über die Hardware-Kennung die Steckplatzinformationen der Eingabebaugruppe auf.

Die ermittelten Steckplatzinformationen der Eingabebaugruppe werden am Ausgangsparameter GEOADDR ("geoAddr") angezeigt. Ermittelt wird ein Modul (HWTYPE mit dem Wert "4") des Zentralgeräts (IOSYSTEM mit dem Wert "0") im Baugruppenträger "0" (Wert von STATION), im Steckplatz "2" (Wert von SLOT).

Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_LOG2GEO

	Name	Data type	Start value	Monitor value
1	Static			
2	execute	Bool	false	TRUE
3	laddr	HW_ANY	257	16#0101
4	geoAddr	GEOADDR		
5	HWTYPE	UInt	0	4
6	AREA	UInt	0	0
7	IOSYSTEM	UInt	0	0
8	STATION	UInt	0	0
9	SLOT	UInt	0	2
10	SUBSLOT	UInt	0	0
11	returnValue	Int	0	0

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

LOG2MOD: Aus der Adressierung von STEP 7 V5.5 SPx die Hardware-Kennung ermitteln

Beschreibung

Mit der Anweisung "LOG2MOD" ermitteln Sie für ein EA-(Sub)Modul aus der Adressierung von STEP 7 5.5 SPx (EA-Datenadresse oder Diagnoseadresse) die Hardware-Kennung.

Die Hardware-Kennung wird an dem Eingangsparameter LADDR von diversen Anweisungen zur Adressierung verwendet. Durch den vorhergehenden Aufruf von "LOG2MOD" konvertieren Sie die Adressierungsparameter aus STEP 7 5.5 SPx.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG2MOD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs wie in STEP 7 5.5 SPx: • B#16#00: Bit15 von ADDR gibt an, ob Ein- (Bit15=0) oder Ausgangsadresse (Bit15=1) vorliegt. • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA)
ADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Adresse der EA-Daten des Moduls als Offset (entsprechend Adressierung in STEP 7 5.5 SPx) oder Diagnoseadresse.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Fehlercode der Anweisung.
HWID	Output	HW_IO	E, A, M, D, L	Ermittelte Hardware-Kennung des EA-(Sub)Moduls.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten.
8093	• Angegebene Adresse wird von keiner Hardware-Komponente verwendet. • Angegebener Wert am Parameter IOID ist ungültig.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

IO2MOD: Aus einer EA-Adresse die Hardware-Kennung ermitteln

Beschreibung

Die Anweisung "IO2MOD" ermittelt aus einer EA-Adresse (E, A, PE, PA) eines Moduls die Hardware-Kennung des Moduls.

Am Parameter ADDR geben Sie die EA-Adresse an:

- Wenn an diesem Parameter eine Reihe von EA-Adressen verwendet wird, wird nur die erste Adresse ausgewertet, um die Hardware-Kennung zu ermitteln.

- Wenn die erste Adresse korrekt angegeben wird, spielt die Länge für die Adressangabe am Parameter ADDR keine Rolle.

- Wenn ein Adressenbereich verwendet wird, der mehrere Module umfasst oder nicht verwendete Adressen enthält, kann die Hardware-Kennung des ersten Moduls ermitteln werden.

- Wenn an dem Parameter ADDR keine EA-Adresse eines Moduls angegeben wird, wird am Parameter RET_VAL der Fehlercode 8090 ausgegeben.

Hinweis

Eingabe der EA-Adresse bei SCL

In SCL programmieren Sie nicht mithilfe der Peripheriezugriffskennung "%QWx:P". Verwenden Sie in diesem Fall den symbolischen Variablennamen oder die absolute Adresse im Prozessabbild.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "IO2MOD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ADDR	Input	VARIANT	E, A	EA-Adresse (E, A, PE, PA) innerhalb eines Moduls. Achten Sie darauf, dass für den Parameter ADDR keine Slice-Zugriffe verwendet werden. Wenn dies der Fall ist, werden am Parameter LADDR falsche Werte ausgegeben.
RET_VAL	Return	INT	E, A, M, D, L	Fehlercode der Anweisung.
LADDR	Output	HW_IO	E, A, M, D, L	Ermittelte Hardware-Kennung (logische Adresse) des EA-Moduls.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten.
8090	Angegebene EA-Adresse am Parameter ADDR wird von keiner Hardware-Komponente verwendet.
8092	Ungültiger Datentyp am Parameter ADDR verwendet (zu Beispiel WCHAR oder WSTRING).
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ermitteln Sie die Hardware-Kennung einer Eingabebaugruppe anhand einer IO-Adresse.

Zum Auslesen legen Sie in einer Variablentabelle eine PLC-Variable an.

SLI_tag_IO2MOD

		Name	Data type	Address
1		SLI_IOtest	Bool	%I0.0

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein drei Variablen an.

SLI_gDB_IO2MOD

		Name	Data type	Start value
1		Static		
2		execute	Bool	false
3		returnValue	Int	0
4		ioLaddr	HW_IO	16#0

Die Parameter der Anweisung verschalten Sie wie folgt.

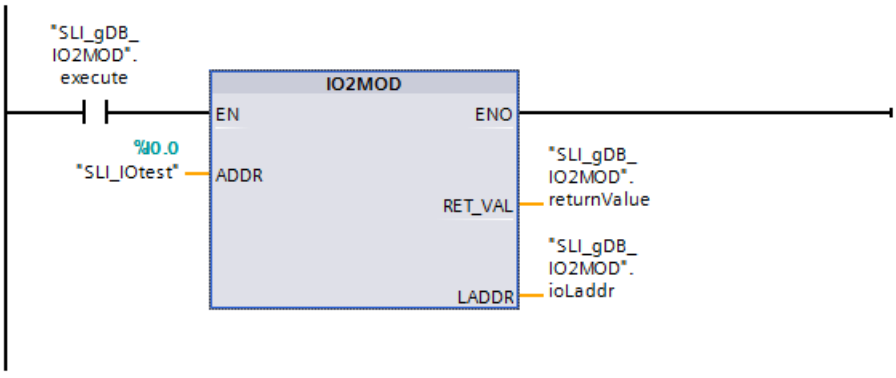

Wenn der Schließerkontakt ("execute") den Signalzustand "TRUE" liefert, wird die Anweisung "IO2MOD" ausgeführt. Über den Eingangsparameter ADDR ("SLI_IOtest") wird die zu lokalisierende IO-Adresse übermittelt. Die Anweisung "IO2MOD" ruft die IO-Adresse auf und liesst die Eingabebaugruppe aus.

Die ermittelte Hardware-Kennung der Eingabebaugruppe wird am Ausgangsparameter LADDR ("ioLaddr") angezeigt. Am Ausgangsparameter RET_VAL ("returnValue") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

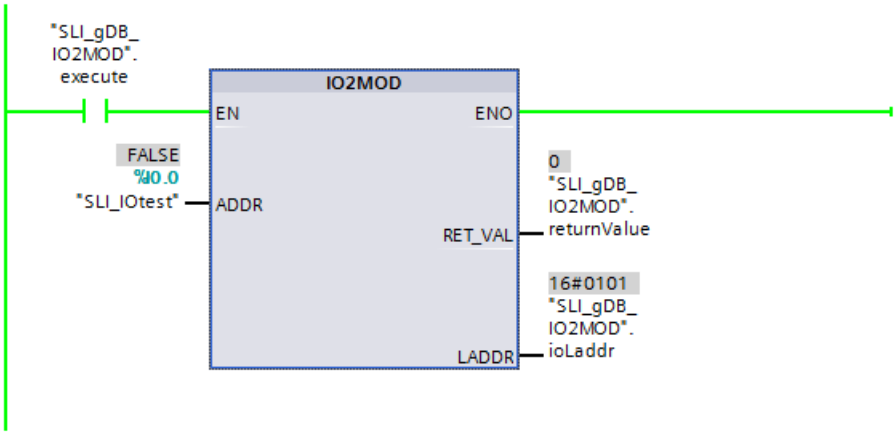

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RD_ADDR: Aus der Hardware-Kennung die EA-Adressen ermitteln

Beschreibung

Die Anweisung "RD_ADDR" ermittelt anhand der Hardware-Kennung eines (Sub-)Moduls die Länge und die Anfangsadresse der Ein- bzw. Ausgänge.

- Über den Parameter LADDR wählen Sie anhand der Hardware-Kennung das Ein- oder Ausgangsmodul aus.
- Je nachdem, ob es sich um ein Ein- oder Ausgangsmodul handelt, werden die folgenden Ausgangsparameter verwendet:
  - Bei einem Eingabemodul werden die ermittelten Werte an den Parametern PIADDR und PICOUNT ausgegeben.
  - Bei einem Ausgabemodul werden die ermittelten Werte an den Parametern PQADDR und PQCOUNT ausgegeben.
- Die Parameter PIADDR und PQADDR enthalten jeweils die Anfangsadresse der E/A-Adressen des Moduls.
- Die Parameter PICOUNT und PQCOUNT enthalten jeweils die Anzahl Bytes der Ein- bzw. Ausgänge (1 Byte bei 8 Ein-/Ausgängen, 2 Byte bei 16 Ein-/Ausgängen).

Hinweis

Ermittlung der EA-Adressen bei gepackten Adressen einer ET200

Bei gepackten Adressen einer ET200 liefert das erste Modul der Packgruppe alle Adressen zurück.

Bei den übrigen Modulen wird an den Parametern PIADDR und PQADDR das Folgende ausgegeben:

- bei PROFINET als Adresse "0"
- bei PROFIBUS als Adresse "0". Zusätzlich wird der Fehlercode W#16#8090 ausgegeben.

Für die Anzahl Bytes der Ein- und Ausgänge (Parameter PICOUNT und PQCOUNT) wird jeweils "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_ADDR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung des (Sub-)Moduls.
RET_VAL	Return	INT	E, A, M, D, L	Fehlercode der Anweisung.
PIADDR	Output	UDINT	E, A, M, D, L	Anfangsadresse des Eingabemoduls.
PICOUNT	Output	UINT	E, A, M, D, L	Anzahl Bytes der Eingänge.
PQADDR	Output	UDINT	E, A, M, D, L	Anfangsadresse des Ausgabemoduls.
PQCOUNT	Output	UINT	E, A, M, D, L	Anzahl Bytes der Ausgänge.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler aufgetreten.
8090	Hardware-Kennung des Moduls am Parameter LADDR ungültig.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

In dem folgenden Beispiel ermitteln Sie die Länge und Startadresse der Eingänge einer Eingabebaugruppe anhand der Hardware-Kennung.

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein sieben Variablen an.

SLI_gDB_RD_ADDR

	Name	Data type	Start value
1	Static		
2	executeRead	Bool	false
3	readLaddr	HW_IO	257
4	readRetVal	Int	0
5	piAddr	UDInt	0
6	piCount	UInt	0
7	pqAddr	UDInt	0
8	pqCount	UInt	0

Die Parameter der Anweisung verschalten Sie wie folgt.

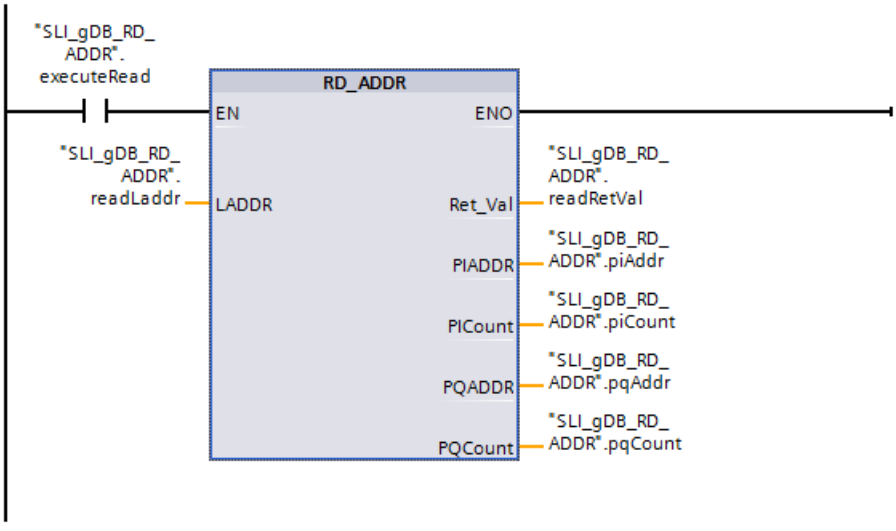

Wenn der Schließerkontakt ("executeRead") den Signalzustand "TRUE" liefert, wird die Anweisung "RD_ADDR" ausgeführt. Über den Eingangsparameter LADDR ("readLaddr") ist

die Hardware-Kennung der Eingabebaugruppe hinterlegt. Die Anweisung "RD_ADDR" ermittelt über die Hardware-Kennung die Eingabebaugruppe und liesst deren Länge der Eingänge und Startadresse aus.

Die ermittelte Startadresse wird am Ausgangsparameter PIADDR ("piAddr") angezeigt. Die Anzahl an Bytes der Eingänge wird am Ausgangsparameter PICOUNT ("piCount") angezeigt.

Am Ausgangsparameter RET_VAL ("readRetVal") wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_RD_ADDR

	Name	Data type	Start value	Monitor value
1	Static			
2	executeRead	Bool	false	TRUE
3	readLaddr	HW_IO	257	16#0101
4	readRetVal	Int	0	0
5	piAddr	UDInt	0	0
6	piCount	UInt	0	2
7	pqAddr	UDInt	0	0
8	pqCount	UInt	0	0

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Systemdatentyp GEOADDR

Geografische Adresse

Der Systemdatentyp GEOADDR enthält die geografische Adresse einer Baugruppe, das heißt die Steckplatzinformation.

- Geografische Adresse bei PROFINET IO
  Bei PROFINET IO setzte sich die geografische Adresse zusammen aus der ID des PROFINET-IO Systems, der Gerätenummer, der Steckplatz-Nummer sowie der Submodulnummer (falls ein Submodul verwendet wird).
- Geografische Adresse bei PROFIBUS DP
  Bei PROFIBUS DP setzt sich die geografische Adresse zusammen aus der ID des DP-Mastersystems, der Stationsnummer und Steckplatz-Nummer.

Die Steckplatzinformationen der Baugruppen entnehmen Sie der Hardware-Konfiguration der jeweiligen Baugruppe.

Systemdatentyp GEOADDR

Wenn Sie in einem Datenbaustein als Datentyp "GEOADDR" eingeben, wird die Struktur GEOADDR automatisch angelegt.

Der Systemdatentyp GEOADDR hat den folgenden Aufbau:

Parametername		Datentyp	Beschreibung
GEOADDR		STRUCT	
	HWTYPE	UINT	Hardwaretyp: • 1: IO-System (PROFINET/PROFIBUS) • 2: IO-Device / DP-Slave • 3: Baugruppenträger • 4: Modul • 5: Submodul Wenn der Hardwaretyp von der Anweisung nicht unterstützt wird, wird an HWTYPE "0" ausgegeben.
	AREA	UINT	Bereichskennung: • 0 = Zentralbaugruppe • 1 = PROFINET IO • 2 = PROFIBUS DP • 3 = AS-i
	IOSYSTEM	UINT	PROFINET IO-System (0=Zentralgerät im Baugruppenträger)
	STATION	UINT	• Nummer des Baugruppenträgers, falls Bereichskennung AREA = 0 (Zentralbaugruppe). • Stationsnummer, falls Bereichskennung AREA > 0.
	SLOT	UINT	Steckplatz-Nummer
	SUBSLOT	UINT	Nummer des Submoduls. Falls kein Submodul vorhanden ist oder gesteckt werden kann, hat dieser Parameter den Wert "0".

Legacy

GEO_LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln

Beschreibung

Von einer Signalbaugruppe ist der zugehörige Baugruppensteckplatz bekannt. Mit der Anweisung "GEO_LOG" ermitteln Sie daraus die zugehörige Hardware-Kennung der Baugruppe.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEO_LOG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MASTER	Input	INT	E, A, M, D, L oder Konstante	Bereichskennung: • 0, falls sich der Steckplatz im zentralen Aufbau befindet. • 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet • 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet
STATION	Input	INT	E, A, M, D, L oder Konstante	• Wenn MASTER = 0: Nummer des Baugruppenträgers • Wenn MASTER > 0: Stationsnummer des Feldgeräts
SLOT	Input	INT	E, A, M, D, L oder Konstante	Steckplatz-Nummer
SUBSLOT	Input	INT	E, A, M, D, L oder Konstante	Nummer des Submoduls. Falls kein Submodul vorhanden ist oder gesteckt werden kann, muss SUBSLOT den Wert "0" haben.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
LADDR	Output	HW_IO	E, A, M, D, L	Hardware-Kennung der Baugruppe

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8094	Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.
8095	Unzulässiger Wert beim Parameter STATION
8096	Unzulässiger Wert beim Parameter SLOT
8097	Unzuläsiger Wert beim Parameter SUBSLOT
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

LOG_GEO: Aus der Hardware-Kennung den Steckplatz ermitteln

Beschreibung

Mit der Anweisung "LOG_GEO" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG_GEO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	HW_IO	E, A, M, D, L oder Konstante	Hardware-Kennung des Moduls, zu dem der Steckplatz ermittelt werden soll.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
AREA	Output	INT	E, A, M, D, L	Die Bereichskennung gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind: • 0: Zentralgerät • 2: PROFIBUS DP / PROFINET IO
MASTER	Output	INT	E, A, M, D, L	Bei AREA = 0: • 0: Falls sich der Steckplatz in einem der Baugruppenträger (Zentralgerät) befindet. Bei AREA = 2: • 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet • 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet
STATION	Output	INT	E, A, M, D, L	• Bei MASTER = 0: Nummer des Baugruppenträgers • Bei MASTER > 0: Stationsnummer des Feldgeräts
SLOT	Output	INT	E, A, M, D, L	Steckplatz-Nummer
SUBSLOT	Output	INT	E, A, M, D, L	Nummer des Submoduls
OFFSET	Output	INT	E, A, M, D, L	Der Parameter OFFSET wird von der Anweisung nicht ausgegeben (immer "0").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8090	Angegebene logische Adresse ungültig
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

RD_LGADR: Aus der Hardware-Kennung die EA-Adressen ermitteln

Beschreibung

Mit der Anweisung "RD_LGADR" ermitteln Sie anhand der Hardware-Kennung die EA-Adressen einer Baugruppe, eines zentralen Submoduls oder eines Submoduls bei PNIO und DP-Slaves.

- Die Hardware-Kennung des Submoduls geben Sie am Parameter LADDR an.
- Die Adressen werden an den Parametern PEADDR und PAADDR in aufsteigender Reihenfolge geschrieben.
  - Bei einer Eingangsbaugruppe wird nur der Parameter PEADDR beschrieben. Bei einer Ausgangsbaugruppe der Parameter PAADDR.
  - Zur Ablage der Adressen verwenden Sie jeweils ein Array of WORD.
- Die Anzahl der Adressen wird an den Parametern PECOUNT (bei einer Eingangsbaugruppe) und PACOUNT (bei einer Ausgangsbaugruppe) ausgegeben.

Hinweis

Beschreibung der Felder PEADDR und PAADDR

In PEADDR bzw. PAADDR werden nur so viele Adressen eingetragen, wie zurückgeliefert werden (Diese Anzahl steht in PECOUNT bzw. PACOUNT). Adressen von früheren RD_LGADR-Aufrufen, die hinter den aktuell gelieferten Adressen liegen, bleiben also erhalten; sie dürfen im aktuellen Aufruf nicht ausgewertet werden.

Hinweis

Ermittlung der EA-Adressen bei gepackten Adressen einer ET200

Bei gepackten Adressen einer ET200 liefert das erste Modul der Packgruppe alle Adressen zurück.

Bei den übrigen Modulen wird an den Parametern PEADDR und PAADDR das Folgende ausgegeben:

- bei PROFINET als Adresse "0"
- bei PROFIBUS als Adresse "0". Zusätzlich wird der Fehlercode W#16#8090 ausgegeben.

Für die Anzahl Bytes der Ein- und Ausgänge (Parameter PECOUNT und PACOUNT) wird jeweils "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_LGADR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Wird nicht ausgewertet
LADDR	Input	HW_ANY	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe bzw. des Submoduls.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PEADDR	Output	ANY	E, A, M, D, L	Feld für die PE-Adressen mit Datentyp Array of WORD Hinweis: Das Feld muss im nicht optimierten Speicherbereich liegen.
PECOUNT	Output	INT	E, A, M, D, L	Anzahl der zurückgelieferten PE-Adressen
PAADDR	Output	ANY	E, A, M, D, L	Feld für die PA-Adressen mit Datentyp Array of WORD Hinweis: Das Feld muss im nicht optimierten Speicherbereich liegen.
PACOUNT	Output	INT	E, A, M, D, L	Anzahl der zurückgelieferten PA-Adressen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8090	Angegebene Hardware-Kennung ungültig
80A0	Fehler beim Ausgangsparameter PEADDR: Der Datentyp der Feldelemente ist nicht WORD.
80A1	Fehler beim Ausgangsparameter PAADDR: Der Datentyp der Feldelemente ist nicht WORD.
80A2	Fehler beim Ausgangsparameter PEADDR: Das angegebene Feld nimmt nicht alle logischen Adressen auf.

Fehlercode (W#16#...)	Erläuterung
80A3	Fehler beim Ausgangsparameter PAADDR: Das angegebene Feld nimmt nicht alle logischen Adressen auf.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

GADR_LGC: Aus dem Steckplatz und dem Offset im Nutzdatenadressraum die Hardware-Kennung ermitteln

Beschreibung

Mit der Anweisung "GADR_LGC" ermitteln Sie die Hardware-Kennung einer Signalbaugruppe. Die Hardware-Kennung wird aus dem Baugruppensteckplatz und dem Offset im Nutzdatenadressraum der Baugruppe ermittelt.

Hinweis

Nutzungseinschränkung

Die Anweisung "GADR_LGC" kann für Baugruppen hinter Netzübergängen (z. B. IE/PB Link) nicht benutzt werden. Verwenden Sie stattdessen die Anweisung "GEO2LOG".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GADR_LGC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SUBNETID	Input	BYTE	E, A, M, D, L oder Konstante	Bereichskennung: • 0: Falls sich der Steckplatz in der Zentralbaugruppe befindet • 1 bis 32: DP-Mastersystem-ID des zugehörigen dezentralen Peripheriesystems, falls sich der Steckplatz in einem dezentralen Peripheriegerät befindet • 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet
RACK	Input	WORD	E, A, M, D, L oder Konstante	• Nummer des Baugruppenträgers, falls Bereichskennung 0 • Stationsnummer des dezentralen Peripheriegeräts, falls Bereichskennung > 0
SLOT	Input	WORD	E, A, M, D, L oder Konstante	Steckplatz-Nr.
SUBSLOT	Input	BYTE	E, A, M, D, L oder Konstante	Irrelevant
SUBADDR	Input	WORD	E, A, M, D, L oder Konstante	Offset im Nutzdatenadressraum der Baugruppe
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
IOID	Output	BYTE	E, A, M, D, L	Der Ausgangsparameter IOID wird nicht beschrieben (immer "0").
LADDR	Output	HW_MODULE	E, A, M, D, L	Hardware-Kennung der Baugruppe

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8094	Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.
8095	Unzulässiger Wert beim Parameter RACK.
8096	Unzulässiger Wert beim Parameter SLOT.
8098	Unzulässiger Wert beim Parameter SUBADDR.

Fehlercode* (W#16#...)	Erläuterung
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

LGC_GADR: Aus der Hardware-Kennung den Steckplatz ermitteln

Beschreibung

Mit der Anweisung "LGC_GADR" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz einer Baugruppe.

Hinweis

Nutzungseinschränkung

Die Anweisung "LGC_GADR" kann für Baugruppen hinter Netzübergängen (z. B. IE/PB Link) nicht benutzt werden. Verwenden Sie stattdessen die Anweisung "LOG2GEO".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LGC_GADR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Wird nicht ausgewertet.
LADDR	Input	HW_MODULE	E, A, M, D, L oder Konstante	Hardware-Kennung der Baugruppe
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
AREA	Output	BYTE	E, A, M, D, L	Die Bereichskennung gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind: • 0: Zentralbaugruppe • 2: PROFIBUS DP

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RACK	Output	WORD	E, A, M, D, L	Baugruppenträger-Nummer: • Bei Zentralbaugruppe (AREA=0): – Baugruppenträger-Nummer • Bei PROFIBUS DP (AREA=2): – Low Byte: Stationsnummer – High Byte: DP-Mastersystem-ID
SLOT	Output	WORD	E, A, M, D, L	Steckplatz-Nummer: • Bei Zentralbaugruppe (AREA=0): – Steckplatz-Nummer • Bei PROFIBUS DP (AREA=2): – Steckplatz-Nr. in der Station
SUBADDR	Output	WORD	E, A, M, D, L	Wird nicht ausgegeben (immer "0").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler aufgetreten.
8090	Angegebene logische Adresse ungültig
8093	Für die über die Parameter IOID und LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.
Allgemeine Fehlerinformation	Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Anweisungen zur Adressumwandlung (Seite 2849)

4.1.4.14 File Handling

FileReadC: Datei von der Memory Card lesen

Beschreibung

Mit der Anweisung "FileReadC" lesen Sie Daten aus einer Datei, die sich auf der Memory Card befindet, und schreiben sie in einen Zielbereich auf der CPU. Die Datei identifizieren Sie über ihren Namen und ihre vollständige Pfadangabe.

Über den Steuerparameter REQ aktivieren Sie das Lesen aus der Datei. Über die Ausgangsparameter Done, Busy, Error und Status wird der Bearbeitungszustand des Auftrags angezeigt.

Für den Dateinamen und die Pfadangabe gelten die folgenden Regeln:

- Der Dateiname darf nicht länger als 55 Zeichen sein.
- Für Ordner und Dateiname sind die folgenden Zeichen zulässig: "0" … "9", "a" … "z", "A" … "Z", "-", "_"
- Der Dateiname darf maximal einen Punkt (".") enthalten; der Punkt trennt den Namen von der Dateierweiterung. Der Dateiname muss mindestens ein Zeichen enthalten. Eine Dateierweiterung ist nicht erforderlich.
- Der Pfadname darf nicht mit "/", "\" oder "." beginnen.
- Der Pfadname darf kein ".." enthalten.

Beispiele für gültige Pfadangaben bzw. Dateinamen sind: "UserFiles/Lift16_DataBase.txt", "UserFiles/2017-04-13_ErrorLog.bin"

Mit einer Ausführung von "FileReadC" können Sie bis zu 16 MByte (16.777.216 Byte) lesen, wobei die Daten segmentweise gelesen werden. Die Segmentgröße ist gerätespezifisch; eine S7-1500-CPU z. B. benutzt 32 KByte (32.768 Byte) große Blöcke. Wenn die Anzahl der zu lesenden Daten größer als die Segmentgröße ist, müssen Sie die Anweisung in Ihrem Programm mehrfach aufrufen. Das hat zur Folge, dass Sie aus Konsistenzgründen erst nach dem letzten Aufruf auf die gelesenen Daten zugreifen dürfen.

Die Parameter "Offset" und "Length" legen die zu lesenden Daten fest. Nach Abschluss des Lesevorgangs wird die belegte Ressource wieder freigegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FileReadC":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerparameter Request Aktiviert das Lesen einer Datei von der Memory Card bei steigender Flanke.
Name	Input	STRING	Name der zu lesenden Datei inkl. vollständigem Pfad
Offset	Input	UDINT	Byte-Offset, ab dem in der Datei gelesen werden soll
Length	Input	UDINT	Länge des zu lesenden Bereichs in Byte Length = 0 bedeutet, dass die maximal mögliche Anzahl von Daten pro Aufruf gelesen wird (bei einer S7-1500-CPU 32 KByte bzw. die Größe des Parameters "Data")
Done	Output	BOOL	1: Die Anweisung wurde erfolgreich ausgeführt. Die gelesenen Informationen wurden an den Parameter "Data" übertragen.
Busy	Output	BOOL	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen oder noch nicht gestartet. • 1: Ausführung der Anweisung noch nicht abgeschlossen.

Parameter	Deklaration	Datentyp	Beschreibung
Error	Output	BOOL	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter "Status" ausgegeben.
Status	Output	WORD	Fehlercode
ResultLength	Output	UDINT	Länge der gelesenen Daten in Byte
Data	InOut	VARIANT	Zielbereich für die gelesenen Daten Zulässige Datentypen: BYTE und Array of BYTE

Weitere Informationen zu den gültigen Datentypen finden Sie unter Übersicht über die gültigen Datentypen (Seite 243)

Parameter Status

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
0081	Warnung: Offset + Length ist größer als die Länge der Datei. Daten werden gelesen von "Offset" bis zum Ende der Datei. "Data" enthält weniger Daten als angefordert wurden; "ResultLength" ist kleiner als "Length"; Daten jenseits von "ResultLength" in "Data" bleiben unverändert.
8091	Pfad existiert nicht oder ist ungültig.
8092	Parameter "Name" ist nicht vom Datentyp "STRING", ist zu lang oder enthält unzulässige Zeichen.
8093	Parameter "Offset" zeigt über die zu lesende Datei hinaus.
8094	Parameter "Length" ist größer als der maximal zulässige Wert. Bei einer S7-1500-CPU z. B. ist der maximal zulässige Wert 16 MByte, also 16.777.216 Byte.
80A1	Lesefehler; der durch den Parameter "Data" vorgegebene Zielbereich kann teilweise überschrieben sein.
80B1	Der durch den Parameter "Data" vorgegebene Zielbereich ist kürzer als die am Parameter "Length" geforderte Länge.
80C0	Auf die Datei kann nicht zugegriffen werden (schreibgeschützt oder durch einen anderen Prozess gesperrt).
80C3	Die maximale Anzahl von gleichzeitig aktiven FileReadC-Anweisungen ist bereits erreicht.
8A24	Der Parameter "Data" zeigt auf einen unzulässigen Bereich, z. B. auf den Ladespeicher oder die Lokaldaten.
8A30	Der Zielbereich ist schreibgeschützt, z. B. ein schreibgeschützter DB.
8A51	Ungültiger Datentyp des Parameters "Data"
8A52	Die Variable am Parameter Data ist unzureichend. Möglicherweise wurde ein Teil der Quellbereich-Daten geschrieben.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FileWriteC: Datei auf der Memory Card beschreiben

Beschreibung

Mit der Anweisung "FileWriteC" schreiben Sie Daten, die sich in einem Quellbereich auf der CPU befinden, in eine im Ordner "UserFiles" auf der Memory Card liegende Datei. Diese Datei identifizieren Sie über ihren Namen und ihre vollständige Pfadangabe. Wenn die Datei nicht existiert, wird sie von der CPU im Ordner "UserFiles" angelegt. Der Parameter "Offset" muss in diesem Fall den Wert Null haben, sonst wird W#16#8093 am Parameter "Status" ausgegeben. Falls auch der Ordner "UserFiles" nicht existiert, wird dieser von der CPU ebenfalls angelegt; Unterverzeichnisse werden nicht angelegt – in diesem Fall wird W#16#8091 am Parameter "Status" ausgegeben.

Über den Steuerparameter "REQ" aktivieren Sie das Schreiben in die Datei. Über die Ausgangsparameter "Done", "Busy", "Error" und "Status" wird der Bearbeitungszustand des Auftrags angezeigt.

Für den Dateinamen und die Pfadangabe gelten die folgenden Regeln:

- Der Dateiname darf nicht länger als 55 Zeichen sein.
- Für Ordner und Dateiname sind die folgenden Zeichen zulässig: "0" … "9", "a" … "z", "A" … "Z", "-", "_"
- Der Dateiname darf maximal einen Punkt (".") enthalten; der Punkt trennt den Namen von der Dateierweiterung. Der Dateiname muss mindestens ein Zeichen enthalten. Eine Dateierweiterung ist nicht erforderlich.
- Der Pfadname darf nicht mit "/", "\" oder "." beginnen.
- Der Pfadname darf kein ".." enthalten.

Beispiele für gültige Pfadangaben bzw. Dateinamen sind: "UserFiles/Lift16_DataBase.txt", "UserFiles/2017-04-13_ErrorLog.bin"

Mit einer Ausführung von "FileWriteC" können Sie bis zu 16 MByte (16.777.216 Byte) schreiben, wobei die Daten segmentweise geschrieben werden. Die Segmentgröße ist gerätespezifisch; eine S7-1500-CPU z. B. benutzt 32 KByte (32.768 Byte) große Blöcke. Wenn die Anzahl der zu schreibenden Daten größer als die Segmentgröße ist, müssen Sie die Anweisung in Ihrem Programm mehrfach aufrufen. Das hat zur Folge, dass Sie aus Konsistenzgründen erst nach dem letzten Aufruf auf die geschriebenen Daten zugreifen dürfen. Wenn die vorhandene Datei zu kurz ist, wird sie auf die notwendige Größe verlängert.

Die Parameter "Offset" und "Length" legen fest, wohin die Daten in der Datei geschrieben werden. Nach Abschluss des Schreibvorgangs wird die belegte Ressource wieder freigegeben.

Die Anweisung "FileWriteC" startet den Schreibvorgang nur dann, wenn die folgende Voraussetzung erfüllt ist: "Offset" + "Length" <= 16 MByte

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FileWriteC":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerparameter Request Aktiviert das Beschreiben einer Datei auf der Memory Card bei steigender Flanke.
Name	Input	STRING	Name der zu beschreibenden Datei inkl. vollständigem Pfad
Offset	Input	UDINT	Byte-Offset, ab dem in der Datei geschrieben werden soll
Length	Input	UDINT	Länge des zu beschreibenden Bereichs in Byte "Length" = 0 bedeutet, dass der gesamte Quellbereich, der durch den Parameter "Data" vorgegeben wird, geschrieben wird.
Done	Output	BOOL	1: Die Anweisung wurde erfolgreich ausgeführt.
Busy	Output	BOOL	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen oder noch nicht gestartet. • 1: Ausführung der Anweisung noch nicht abgeschlossen.
Error	Output	BOOL	Zustandsparameter • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter "Status" ausgegeben.
Status	Output	WORD	Fehlercode
ResultLength	Output	UDINT	Länge der geschriebenen Daten in Byte
Data	InOut	VARIANT	Quellbereich Zulässige Datentypen: BYTE und Array of BYTE

Weitere Informationen zu den gültigen Datentypen finden Sie unter Übersicht über die gültigen Datentypen (Seite 243)

Parameter Status

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler
8091	Pfad existiert nicht oder ist ungültig.
8092	Parameter "Name" ist nicht vom Datentyp "STRING", ist zu lang oder enthält unzulässige Zeichen.
8093	• Parameter "Offset" zeigt über die zu beschreibende Datei hinaus • Das Anlegen der Datei wird abgewiesen, weil "Offset" größer als Null ist.
8094	• "Length" ist größer als der maximal zulässige Wert. Bei einer S7-1500-CPU z. B. ist der maximal zulässige Wert 16 MByte, also 16.777.216 Byte. • "Length" + "Offset" ist größer als der maximal zulässige Wert.
80A1	Schreibfehler; die Daten in der Datei auf der Memory Card können teilweise überschrieben sein.
80B1	Der durch den Parameter "Data" vorgegebene Quellbereich ist kürzer als die am Parameter "Length" geforderte Länge.
80B3	Auf der Memory Card oder dem internen Ladespeicher ist nicht genügend Platz.

Fehlercode* (W#16#...)	Erläuterung
80B4	Die Memory Card oder die Datei ist schreibgeschützt.
80C0	Auf die Datei kann nicht zugegriffen werden.
80C3	Die maximale Anzahl von gleichzeitig aktiven FileWriteC-Anweisungen ist bereits erreicht.
8A24	Der Parameter "Data" zeigt auf einen unzulässigen Bereich, z. B. auf den Ladespeicher oder die Lokaldaten.
8A51	Ungültiger Datentyp des Parameters "Data"
8A52	Die Variable am Parameter Data ist unzureichend. Möglicherweise wurde ein Teil der Quellbereich-Daten geschrieben.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.4.15 R/H-System

RH_CTRL: Abläufe bei R/H-Systemen beeinflussen

Beschreibung

Mit der Anweisung "RH_CTRL" können Sie folgendermaßen auf R/H-Systeme einwirken:

- Sie können den Systemzustand "SYNCUP" sperren. Diese Sperre gilt so lange, bis Sie diese mit der Anweisung "RH_CTRL" wieder aufheben oder bis das R/H-System in den Systemzustand "STOP" geht.
- Sie können den Systemzustand "SYNCUP" wieder freigeben.

Arbeitsweise

Die Anweisung "RH_CTRL" ist eine asynchron arbeitende Anweisung. Die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "RH_CTRL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RH_CTRL":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerparameter Request Aktiviert die Ausführung der Anweisung bei REQ = "1".
Mode	Input	BYTE	Über den Parameter "Mode" wählen Sie aus, wie Sie auf das R/H-System einwirken wollen: • 3: Systemzustand SYNCUP sperren • 4: Systemzustand SYNCUP wieder freigeben
Submode	Input	WORD	Verborgener Parameter. Er muss den Wert 0 haben.
Ret_Val	Output	INT	Fehlercode
Busy	Output	BOOL	Zustandsparameter • 0: Ausführung der Anweisung abgeschlossen. • 1: Ausführung der Anweisung noch nicht abgeschlossen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter Ret_Val

Fehlercode* (W#16#...)	Erläuterung
0	Kein Fehler.
1	• bei Mode = 3: Der Systemzustand SYNCUP ist bereits gesperrt. • bei Mode = 4: Der Systemzustand SYNCUP ist bereits freigegeben.
7000	Kein Auftrag in Bearbeitung (REQ = 0).
7001	Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
8082	Das Sperren des Systemzustands "SYNCUP" ist nicht mehr möglich, weil das R/H-System sich bereits in diesem Systemzustand befindet.
8090	Ungültiger Wert des Parameters "Mode" .
8091	Ungültiger Wert des Parameters "Submode". Er muss den Wert 0 haben.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RH_GetPrimaryID: Redundanz-ID der Primary-CPU bestimmen

Beschreibung

Mit der Anweisung "RH_GetPrimaryID" ermitteln Sie die Redundanz-ID der Primary-CPU.

Sie können diese Anweisung in allen Systemzuständen verwenden, also auch dann, wenn Sie das Pairing der beiden R- bzw. H-CPUs noch nicht durchgeführt haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RH_GetPrimaryID":

Parameter	Deklaration	Datentyp	Beschreibung
Ret_Val	Output	INT	Rückgabewert

Weitere Informationen zu den gültigen Datentypen finden Sie unter: Übersicht über die gültigen Datentypen (Seite 243)

Parameter Ret_Val

Fehlercode* (W#16#...)	Erläuterung
1	Die CPU mit Redundanz-ID 1 ist die Primary-CPU.
2	Die CPU mit Redundanz-ID 2 ist die Primary-CPU.
8091	Keine Daten verfügbar, z. B. weil "RH_GetPrimaryID" von Ihrer CPU nicht unterstützt wird.
* Die Fehlercodes im Programmeditor werden als Integer- oder Hexadezimalwert dargestellt. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.1.5 Technologie

4.1.5.1 Schnelle Zähler

CTRL_HSC: Schnelle Zähler steuern

Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	INPUT	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	OUTPUT	BOOL	E, A, M, D, L	Freigabeausgang
HSC	INPUT	HW_HSC	E, A, M oder Konstante	Hardware-Kennung des Schnellen Zählers (HW-ID)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DIR	INPUT	BOOL	E, A, M, D, L oder Konstante	Freigabe der neuen Zählrichtung (siehe NEW_DIR)
CV	INPUT	BOOL	E, A, M, D, L oder Konstante	Freigabe des neuen Zählerwerts (siehe NEW_CV)
RV	INPUT	BOOL	E, A, M, D, L oder Konstante	Freigabe des neuen Referenzwerts (siehe NEW_RV)
PERIOD	INPUT	BOOL	E, A, M, D, L oder Konstante	Freigabe der neuen Zeitdauer einer Frequenzmessung (siehe NEW_PERIOD)
NEW_DIR	INPUT	INT	E, A, M, D, L oder Konstante	Zählrichtung, die bei DIR = TRUE geladen wird.
NEW_CV	INPUT	DINT	E, A, M, D, L oder Konstante	Zählerwert, der bei CV = TRUE geladen wird.
NEW_RV	INPUT	DINT	E, A, M, D, L oder Konstante	Referenzwert, der bei RV = TRUE geladen wird.
NEW_PERIOD	INPUT	INT	E, A, M, D, L oder Konstante	Zeitdauer der Frequenzmessung, die bei PERIOD = TRUE geladen wird.
BUSY	OUTPUT	BOOL	E, A, M, D, L	Bearbeitungsstatus *
STATUS	OUTPUT	WORD	E, A, M, D, L	Status der Operation

* Bei einem Schnellen Zähler in der CPU oder im Signal Board hat der Parameter BUSY immer den Wert 0.

Beschreibung

Mit der Anweisung "Schnelle Zähler steuern" können Sie die von der CPU unterstützten Schnellen Zähler über die Software parametrieren und steuern, indem Sie neue Werte in die Zähler laden. Eine Voraussetzung für die Ausführung der Anweisung ist, dass ein zu steuernder Schneller Zähler aktiviert ist. Pro Schnellen Zähler können Sie im Programm nicht mehrere Anweisungen "Schnelle Zähler steuern" gleichzeitig ausführen.

Die folgenden Parameterwerte können Sie mithilfe der Anweisung "Schnelle Zähler steuern" in einen Schnellen Zähler laden:

- Zählrichtung (NEW_DIR): Die Zählrichtung definiert, ob ein Schneller Zähler vorwärts oder rückwärts zählt. Die Zählrichtung wird durch die folgenden Werte am Eingang NEW_DIR definiert: 1 = Vorwärts, -1=Rückwärts.
  Ein Wechsel der Zählrichtung über die Anweisung "Schnelle Zähler steuern" ist nur dann möglich, wenn für den angegebenen Zähler eine Richtungssteuerung über das Programm parametriert ist. Die am Eingang NEW_DIR angegebene Zählrichtung wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang DIR gesetzt ist.
- Zählerwert (NEW_CV): Der Zählwert ist der Anfangswert, mit dem ein Schneller Zähler neu zum Zählen beginnt. Der Zählwert kann in einem Bereich von -2147483648 bis 2147483647 liegen.
  Der am Eingang NEW_CV angegebene Zählwert wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang CV gesetzt ist.
- Referenzwert (NEW_RV): Der Referenzwert kann mit dem aktuellen Zählerwert verglichen werden, um einen Alarm auszulösen. Der Referenzwert kann wie der Zählerwert in einem Bereich von -2147483648 bis 2147483647 liegen.
  Der am Eingang NEW_RV angegebene Referenzwert wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang RV gesetzt ist.
- Zeitdauer der Frequenzmessung (NEW_PERIOD): Die Zeitdauer der Frequenzmessung wird durch die folgenden Werte am Eingang NEW_PERIOD festgelegt: 10 = 0,01s, 100 = 0,1s, 1000 = 1s.
  Eine Aktualisierung der Zeitdauer ist dann möglich, wenn die Funktion "Frequenzmessen" für den angegebenen Schnellen Zähler parametriert ist. Die am Eingang NEW_PERIOD angegebene Zeitdauer wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang PERIOD gesetzt ist.

Die Anweisung "Schnelle Zähler steuern" wird nur ausgeführt, wenn der Signalzustand am Eingang EN "1" ist.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und keine Fehler während der Bearbeitung der Operation auftreten.

Beim Einfügen der Anweisung "Schnelle Zähler steuern" wird ein Instanz-Datenbaustein angelegt, in dem die Daten der Operation gespeichert werden.

Parameter STATUS

Am Ausgang STATUS kann abgefragt werden, ob Fehler während der Bearbeitung der Anweisung "Schnelle Zähler steuern" aufgetreten sind. Die folgende Tabelle zeigt die Bedeutung der Werte, die am Ausgang STATUS ausgegeben werden:

Fehlercode (Hexadezimal)	Beschreibung
0	Keine Fehler
80A1	Hardware-Kennung der Schnellen Zähler ungültig
80B1	Zählrichtung (NEW_DIR) ungültig
80B2	Zählwert (NEW_CV) ungültig
80B3	Referenzwert (NEW_RV) ungültig
80B4	Zeitdauer der Frequenzmessung (NEW_PERIOD) ungültig

Fehlercode (Hexadezimal)	Beschreibung
80C0	Mehrfacher Zugriff auf den Schnellen Zähler
80D0	Der Schnelle Zähler (HSC) ist in der CPU Hardware-Konfiguration nicht aktiviert.

CTRL_HSC_EXT: Schnelle Zähler steuern (erweitert)

Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	INPUT	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	OUTPUT	BOOL	E, A, M, D, L	Freigabeausgang
HSC	INPUT	HW_HSC	E, A, M oder Konstante	Hardware-Kennung des Schnellen Zählers (HW-ID)
CTRL	INOUT	VARIANT	M, D	Verwendung eines Systemdatentyps (SDT)
DONE	OUTPUT	BOOL	E, A, M, D, L	Rückmeldung nach erfolgreicher Bearbeitung Anweisung
BUSY	OUTPUT	BOOL	E, A, M, D, L	Bearbeitungsstatus
ERROR	OUTPUT	BOOL	E, A, M, D, L	Rückmeldung bei fehlerhafter Bearbeitung der Anweisung
STATUS	OUTPUT	WORD	E, A, M, D, L	Status der Operation

Beschreibung

Mit der Anweisung "Schnelle Zähler steuern (erweitert)" können Sie die von der CPU unterstützten Schnellen Zähler über die Software parametrieren und steuern, indem Sie neue Werte in die Zähler laden. Eine Voraussetzung für die Ausführung der Anweisung ist, dass ein zu steuernder Schneller Zähler aktiviert ist. Pro Schnellen Zähler können Sie im Programm nicht mehrere Anweisungen "Schnelle Zähler steuern (erweitert)" gleichzeitig ausführen.

Die Anweisung "Schnelle Zähler steuern (erweitert)" wird nur ausgeführt, wenn der Signalzustand am Eingang EN "1" ist. Solange die Operation bearbeitet wird, bleibt das Bit am Ausgang BUSY gesetzt. Wenn die Operation vollständig ausgeführt ist, wird das Bit am Ausgang BUSY zurückgesetzt.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und keine Fehler während der Bearbeitung der Operation auftreten.

Beim Einfügen der Anweisung "Schnelle Zähler steuern (erweitert)" wird ein Instanz-Datenbaustein angelegt, in dem die Daten der Operation gespeichert werden.

Verwendung des Systemdatentyps HSC_Period

Die Anweisung "Schnelle Zähler steuern (erweitert)" unterstützt für die Periodendauermessung den Systemdatentyp SDT 381 "HSC_Period".

Der Datentyp "HSC_Period" entspricht einem HSC, der für die Betriebsart "Periode" konfiguriert ist. Die Anweisung CTRL_HSC_EXT bietet Programmzugriff auf die Anzahl von Eingangsimpulsen über ein angegebenes Messintervall. Diese Anweisung ermöglicht die Berechnung des Zeitraums zwischen den Eingangsimpulsen mit einer Auflösung im Nanosekundenbereich.

Byte	Parameter	Deklaration	Datentyp	Beschreibung
0 ... 3	Elapsed-Time	OUT	UDINT	Zeit zwischen den steigenden Flanken aus Edge_Count
4 ... 7	EdgeCount	OUT	UDINT	Anzahl der steigenden Flanken innerhalb der vergangenen Zeit aus Elapsed_Time. Wenn Edge_Count = 0 ist, dann ist Elapsed_Time die Zeit seit der letzten steigenden Flanke.
8.0	EnHSC	IN	BOOL	Verwendung als Freigabeeingang über Torsteuerung: • FALSE: Messung gestoppt • TRUE: Messung freigegeben
8.6	EnPeriod	IN	BOOL	Aktualisierung der Periode • FALSE: Keine Aktualisierung • TRUE: Periode aktualisieren
10 ... 11	NewPeriod	IN	INT	Intervall der Periodenmessung in Millisekunden. Erlaubte Werte sind 10, 100 and 1000.

ElapsedTime gibt die Zeit in Nanosekunden zwischen den letzten Zählereignissen von aufeinander folgenden Messintervallen an. Traten während eines Messintervalls keine Zählereignisse auf, gibt ElapsedTime die kumulierte Zeit seit dem letzten Zählereignis aus. ElapsedTime hat einen Bereich von 0 bis 4.294.967.280 Nanosekunden (0x0000 0000 bis 0xFFFF FFF0). Der Rückgabewert 4.294.967.295 (0xFFFF FFFF) meldet, dass ein Periodenüberlauf aufgetreten ist. Ein Überlauf deutet darauf hin, dass die Zeit zwischen Impulsflanken größer als 4,295 Sekunden ist und die Periode mit dieser Anweisung nicht berechnet werden kann. Die Werte von 0xFFFF FFF1 bis 0xFFFF FFFE sind reserviert.

EdgeCount gibt die Anzahl der während des Messintervalls empfangenen Zählereignisse aus. Die Periode kann nur berechnet werden, wenn der Wert von EdgeCount größer als Null ist. Wenn ElapsedTime entweder 0 (keine Eingangsimpulse empfangen) oder 0xFFFF FFFF (Periodenüberlauf) ist, dann ist EdgeCount nicht gültig.

Wenn EdgeCount gültig ist, berechnen Sie die Periode in Nanosekunden mit Hilfe der folgenden Formel: Periode = ElapsedTime/EdgeCount

Der berechnete Periodenwert ist ein Mittelwert aus den Zeitperioden sämtlicher Impulse, die während des Messintervalls auftreten. Wenn die Periode eines kommenden Impulses größer als das Messintervall ist (10, 100 oder 1000 ms), benötigt die Periodenberechnung mehrere Messintervalle.

Die folgenden Bilder zeigen Beispiele für die Periodendauermessung mit der Anweisung:

Beispiel 1: Mehrere Zählereignisse in einem Messintervall

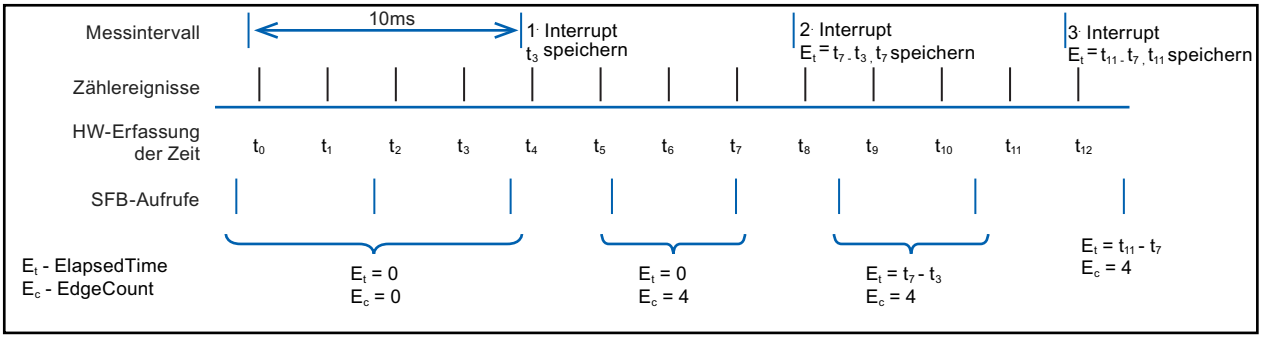

Beispiel 2: Null und ein Zählereignis in mehreren Messintervallen

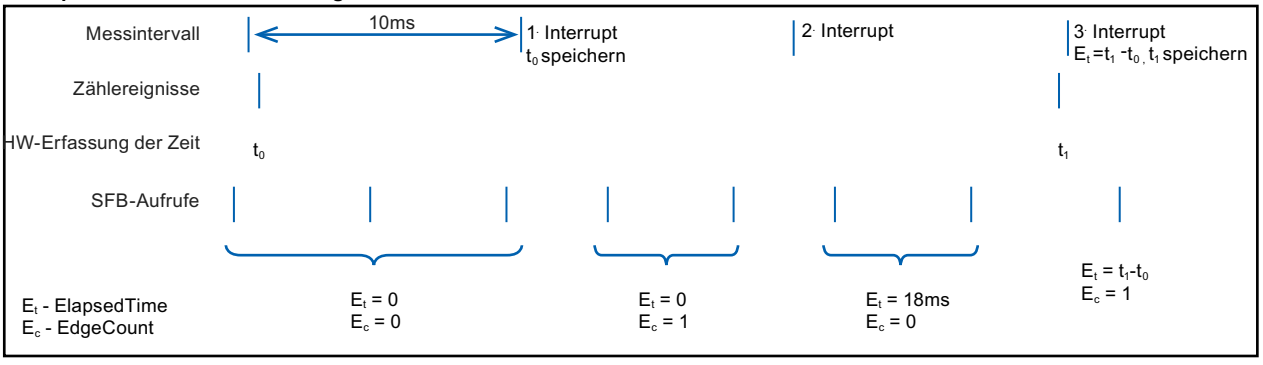

Regeln:
1. Wenn $E1_t = 0$, ist der Zeitraum ungültig
2. Andernfalls, Zeitraum = E_t / E_c

Parameter STATUS

Am Ausgang STATUS kann abgefragt werden, ob Fehler während der Bearbeitung der Anweisung "Schnelle Zähler steuern (erweitert)" aufgetreten sind. Die folgende Tabelle zeigt die Bedeutung der Werte, die am Ausgang STATUS ausgegeben werden:

Fehlercode (Hexadezimal)	Beschreibung
0	Keine Fehler
80A1	Hardware-Kennung der Schnellen Zähler ungültig
80C0	Mehrfacher Zugriff auf den Schnellen Zähler
80D0	Der Schnelle Zähler (HSC) ist in der CPU Hardware-Konfiguration nicht aktiviert.

4.1.5.2 Zählen und Messen (S7-1500)

High_Speed_Counter

High_Speed_Counter

Beschreibung High_Speed_Counter

Beschreibung

Mit der Anweisung High_Speed_Counter steuern Sie über das Anwenderprogramm die Zähl- und Messfunktionen des Technologiemoduls.

Aufruf

Die Anweisung High_Speed_Counter muss im Zyklus oder alternativ in einem zeitgesteuerten Programm einmal pro Zähler aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Arbeitsweise

Zählwert: Der Zählwert ist am Ausgangsparameter CountValue verfügbar. Der Zählwert wird bei jedem Aufruf der Anweisung High_Speed_Counter aktualisiert.

Messwert: Das Technologiemodul aktualisiert den Messwert mit der parametrierten Aktualisierungszeit asynchron zum Aufruf der Anweisung. Bei jedem Aufruf der Anweisung wird der zuletzt vom Technologiemodul ermittelte Messwert am Ausgangsparameter MeasuredValue aktualisiert.

Messwert und Zählwert stehen in der Rückmeldeschnittstelle parallel zur Verfügung.

Capture: Der Ausgangsparameter CaptureStatus = TRUE zeigt einen gültigen Capture-Wert am Ausgangsparameter CapturedValue an.

- Ein Capture-Wert wird unter folgenden Bedingungen erfasst:
  - Ein Digitaleingang hat die Parametrierung "Capture"
  - CaptureEnable = TRUE
  - Flanke am Digitaleingang mit der Capture-Funktion
- Der Ausgangsparameter CaptureStatus wird rückgesetzt durch eine fallende Flanke am Eingangsparameter CaptureEnable.

Synchronisation: Der Ausgangsparameter SyncStatus = TRUE zeigt an, dass eine Synchronisation aufgetreten ist.

- Der Zählwert wird unter folgenden Bedingungen synchronisiert:
  - Ein Digitaleingang hat die Parametrierung "Synchronisation" **oder** der Inkrementalgeber hat die Parametrierung "Synchronisation bei Signal N"
  - SyncEnable = TRUE
  - Die statische Variable SyncUpDirection (bzw. SyncDownDirection) = TRUE
  - Flanke am Digitaleingang mit der Synchronisationsfunktion **oder** steigende Flanke des Signals N am Gebereingang
- Der Ausgangsparameter SyncStatus wird rückgesetzt durch eine fallende Flanke an
  - dem Eingangsparameter SyncEnable oder
  - der statischen Variable SyncDownDirection oder
  - der statischen Variable SyncUpDirection

Parameteränderung über das Anwenderprogramm

Wenn Sie Parameter über das Anwenderprogramm ändern wollen, gehen Sie folgendermaßen vor:

1. Prüfen Sie anhand der jeweiligen Set-Variable, ob das Technologieobjekt für die Parameteränderung bereit ist (Set-Variable = FALSE) oder ob noch ein Änderungsauftrag läuft (Set-Variable = TRUE).
   In den statischen Variablen des Technologieobjekt-Instanz-DB stehen die folgenden Set-Variablen in UserCmdFlags dafür zur Verfügung:
   - SetReferenceValue0
   - SetReferenceValue1
   - SetUpperLimit
   - SetLowerLimit
   - SetCountValue
   - SetStartValue
   - SetNewDirection
2. Wenn das Technologieobjekt für die Parameteränderung bereit ist, ändern Sie die jeweilige statische Variable.
   Die folgenden statischen Variablen des Technologieobjekt-Instanz-DB stehen dafür zur Verfügung:
   - NewReferenceValue0 / NewReferenceValue0_M (für SetReferenceValue0)
   - NewReferenceValue1 / NewReferenceValue1_M (für SetReferenceValue1)
   - NewUpperLimit
   - NewLowerLimit
   - NewCountValue
   - NewStartValue
   - NewDirection

3. Setzen Sie die jeweilige Set-Variable zur Durchführung des Änderungsauftrags.
4. Prüfen Sie mit Hilfe des Ausgangsparameters Error, ob ein Fehler aufgetreten ist. Wenn kein Fehler aufgetreten ist und die Set-Variable durch das Technologieobjekt automatisch rückgesetzt wurde, war die Parameteränderung erfolgreich.

Hinweis

Geänderte Zählgrenze

Wenn die neue obere Zählgrenze kleiner ist als der aktuelle Zählwert, wird der Zählwert abhängig von der Parametrierung auf die untere Zählgrenze oder den Startwert gesetzt. Wenn die neue untere Zählgrenze größer ist als der aktuelle Zählwert, wird der Zählwert abhängig von der Parametrierung auf die obere Zählgrenze oder den Startwert gesetzt.

Betriebsart (High_Speed_Counter ab V3.0)

Die Betriebsart parametrieren Sie im Technologieobjekt unter "Verhalten DQ0".

Die Betriebsart wird durch den Ausgangsparameter CompareMeasuredValue angezeigt:

Zustand	Beschreibung
FALSE	Betriebsart "Zählwert als Bezug verwenden": Die Vergleichsfunktionen arbeiten mit dem Zählwert. Folgende statische Variablen werden spezifisch in dieser Betriebsart verwendet: • NewReferenceValue0 • NewReferenceValue1 • CurReferenceValue0 • CurReferenceValue1 Die vier spezifischen statischen Variablen der Betriebsart "Messwert als Bezug verwenden" werden ignoriert.
TRUE	Betriebsart "Messwert als Bezug verwenden": Die Vergleichsfunktionen arbeiten mit dem Messwert. Folgende statische Variablen werden spezifisch in dieser Betriebsart verwendet: • NewReferenceValue0_M • NewReferenceValue1_M • CurReferenceValue0_M • CurReferenceValue1_M Die vier spezifischen statischen Variablen der Betriebsart "Zählwert als Bezug verwenden" werden ignoriert.

Quittierung von Ereignissen

Die Quittierung von gemeldeten Ereignissen nehmen Sie über die steigende Flanke des Eingangsparameters EventAck vor. EventAck muss gesetzt bleiben, bis das Technologieobjekt die Statusbits der folgenden Ereignisse des Zählkanals zurückgesetzt hat:

- CompResult0
- CompResult1

- ZeroStatus
- PosOverflow
- NegOverflow

Status der Digitaleingänge (TM Count und TM PosInput)

Den Status der Digitaleingänge erhalten Sie über die statischen Variablen StatusDI0, StatusDI1 bzw. StatusDI2.

Status der Digitaleingänge (Kompakt-CPU)

Den Status der Digitaleingänge erhalten Sie über die statischen Variablen StatusDI0 und StatusDI1. Wenn ein Digitaleingang der Kompakt-CPU nicht für einen Zähler verwendet wird, können Sie den Digitaleingang über das Anwenderprogramm nutzen.

Digitalausgänge durch Anwenderprogramm nutzen (TM Count und TM PosInput)

Sie können über die Anweisung High_Speed_Counter die Digitalausgänge setzen,

- wenn für "Ausgang setzen" die Einstellung "Nutzung durch Anwenderprogramm" parametriert ist.
- wenn für "Ausgang setzen" die Einstellung "Nach Setzbefehl aus CPU bis Vergleichswert" parametriert ist.
- wenn Sie die jeweilige statische Variable ManualCtrlDQm setzen (temporäres Überschreiben).

Nur in diesen Fällen sind die statischen Variablen SetDQ0 und SetDQ1 wirksam. Im ersten und dritten Fall folgt DQm dem Wert von SetDQm. Im zweiten Fall wird DQm mit einer steigenden Flanke von SetDQm gesetzt. DQm wird zurückgesetzt, wenn der Zählwert dem Vergleichswert entspricht oder bei einer fallenden Flanke von SetDQm..

Digitalausgänge durch Anwenderprogramm nutzen (Kompakt-CPU)

Sie können über die Anweisung High_Speed_Counter den Digitalausgang DQ1 setzen,

- wenn für "Ausgang setzen" die Einstellung "Nutzung durch Anwenderprogramm" parametriert ist.
- wenn für "Ausgang setzen" die Einstellung "Nach Setzbefehl aus CPU bis Vergleichswert" parametriert ist.
- wenn Sie die statische Variable ManualCtrlDQ1 setzen (temporäres Überschreiben).

Nur in diesen Fällen ist die statische Variable SetDQ1 wirksam. Im ersten und dritten Fall folgt DQ1 dem Wert von SetDQ1. Im zweiten Fall wird DQ1 mit einer steigenden Flanke von SetDQ1

gesetzt und zurückgesetzt, wenn der Zählwert dem Vergleichswert entspricht oder bei einer fallenden Flanke von SetDQ1.

Hinweis

Bevor Sie über die Anweisung High_Speed_Counter einen physikalischen Digitalausgang der Kompakt-CPU setzen können, müssen Sie das Signal DQ1 dem gewünschten Digitalausgang zuordnen.

Sie können über die Anweisung High_Speed_Counter mit der statischen Variable StatusDQ0 das Signal DQ0 setzen,

- wenn für "Ausgang setzen" die Einstellung "Nutzung durch Anwenderprogramm" parametriert ist.
- wenn für "Ausgang setzen" die Einstellung "Nach Setzbefehl aus CPU bis Vergleichswert" parametriert ist.
- wenn Sie die statische Variable ManualCtrlDQ0 setzen (temporäres Überschreiben).

Nur in diesen Fällen ist die statische Variable SetDQ0 wirksam. Im ersten und dritten Fall folgt StatusDQ0 dem Wert von SetDQ0. Im zweiten Fall wird StatusDQ0 mit einer Flanke (steigend oder fallend) von SetDQ0 gesetzt und zurückgesetzt, wenn der Zählwert dem Vergleichswert entspricht.

Hinweis

Der Digitalausgang DQ0 ist nicht als physikalischer Ausgang verfügbar.

Verhalten im Fehlerfall

Wenn beim Aufruf der Anweisung oder im Technologiemodul ein Fehler aufgetreten ist, wird der Ausgangsparameter Error gesetzt. Die weiterführende Fehlerinformation kann dann am Ausgangsparameter ErrorID (Seite 2892) ausgelesen werden.

Beheben Sie die Ursache des Fehlers und quittieren Sie die Fehlermeldung durch Setzen des Eingangsparameters ErrorAck. Wenn kein Fehler mehr ansteht, setzt das Technologieobjekt den Ausgangsparameter Error zurück. Es wird kein neuer Fehler gemeldet, solange Sie den vorherigen Fehler nicht quittiert haben.

Zählrichtung ändern

Die Zählrichtung kann vom Anwenderprogramm nur geändert werden, wenn als Signalart "Impuls (A)" parametriert ist. In allen anderen Fällen wird die Zählrichtung von den Eingangssignalen des Technologiemoduls bestimmt. Die Zählrichtung kann durch die statische Variable NewDirection gesteuert werden:

- +1: Zählrichtung vorwärts
- -1: Zählrichtung rückwärts

Zur Durchführung des Änderungsauftrags ist ein Setzen der statischen Variable SetNewDirection = TRUE erforderlich.

Eingangsparameter High_Speed_Counter

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
SwGate	INPUT	BOOL	FALSE	SW-Tor steuern: • Steigende Flanke: SW-Tor wird geöffnet • Fallende Flanke: SW-Tor wird geschlossen SwGate gibt in Verbindung mit dem HW-Tor das interne Tor frei.
CaptureEnable	INPUT	BOOL	FALSE	Capture-Funktion freigeben Nach der Freigabe erfolgt ein Capture-Ereignis bei der nächsten parametrierten Flanke am jeweiligen Digitaleingang. Eine fallende Flanke an CaptureEnable setzt den Ausgangsparameter Capture-Status zurück. Eine fallende Flanke an CaptureEnable setzt die Freigabe auch dann zurück, wenn kein Capture-Ereignis erfolgt ist. Unabhängig von CaptureEnable bleibt der letzte Wert am Ausgangsparameter CapturedValue erhalten bis zum nächsten Capture-Ereignis.
SyncEnable	INPUT	BOOL	FALSE	Synchronisation freigeben Die freigegebene Richtung für die Synchronisation wird in den statischen Variablen SyncUpDirection und SyncDownDirection angegeben. Eine fallende Flanke an SyncEnable setzt den Ausgangsparameter SyncStatus zurück.
ErrorAck	INPUT	BOOL	FALSE	High_Speed_Counter bis V3.0: Eine steigende Flanke quittiert den gemeldeten Fehlerzustand. High_Speed_Counter ab V3.1: Ein High-Pegel quittiert den gemeldeten Fehlerzustand.
EventAck	INPUT	BOOL	FALSE	Eine steigende Flanke setzt folgende Ausgangsparameter zurück: • CompResult0 • CompResult1 • ZeroStatus • PosOverflow • NegOverflow
SetCountValue	INOUT	BOOL	FALSE	Eine steigende Flanke startet die Übertragung des neuen Zählwerts in der statischen Variable NewCountValue zum Technologiemodul. Der Zählwert wird sofort nach der Übertragung wirksam.

Ausgangsparameter High_Speed_Counter

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
StatusHW	OUTPUT	BOOL	FALSE	Statusbit Technologiemodul: Das Modul ist parametriert und betriebsbereit. Die Daten des Moduls sind gültig.
StatusGate	OUTPUT	BOOL	FALSE	Statusbit: Internes Tor ist freigegeben, wenn Parameter gesetzt
StatusUp	OUTPUT	BOOL	FALSE	Statusbit: Letzter Zählimpuls hat Zähler inkrementiert und liegt maximal 0,5 s zurück
StatusDown	OUTPUT	BOOL	FALSE	Statusbit: Letzter Zählimpuls hat Zähler dekrementiert und liegt maximal 0,5 s zurück

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
CompResult0	OUTPUT	BOOL	FALSE	Statusbit: Vergleichsereignis für DQ0 (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten Wenn in der Betriebsart "Zählwert als Bezug verwenden" der Zählwert auf den Startwert gesetzt wird, wird CompResult0 nicht gesetzt. Über die steigende Flanke des Eingangsparameters EventAck setzen Sie CompResult0 zurück.
CompResult1	OUTPUT	BOOL	FALSE	Statusbit: Vergleichsereignis für DQ1 (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten Wenn in der Betriebsart "Zählwert als Bezug verwenden" der Zählwert auf den Startwert gesetzt wird, wird CompResult1 nicht gesetzt. Über die steigende Flanke des Eingangsparameters EventAck setzen Sie CompResult1 zurück.
SyncStatus	OUTPUT	BOOL	FALSE	Statusbit: Synchronisation aufgetreten Wenn der Eingangsparameter SyncEnable gesetzt ist, setzt die parametrierte Flanke am jeweiligen Digitaleingang das Statusbit SyncStatus. SyncStatus wird rückgesetzt durch fallende Flanke an • SyncEnable (Eingangsparameter) oder • SyncUpDirection (Statische Variable) oder • SyncDownDirection (Statische Variable)
CaptureStatus	OUTPUT	BOOL	FALSE	Statusbit: Capture-Ereignis aufgetreten, ein gültiger Capture-Wert liegt vor im Ausgangsparameter CapturedValue Wenn der Eingangsparameter CaptureEnable gesetzt ist, setzt die parametrierte Flanke am jeweiligen Digitaleingang das Statusbit CaptureStatus. Über die fallende Flanke des Eingangsparameters CaptureEnable setzen Sie CaptureStatus zurück.
ZeroStatus	OUTPUT	BOOL	FALSE	Statusbit: CountValue hat unabhängig von der Zählrichtung Wert "0" erreicht Über die steigende Flanke des Eingangsparameters EventAck setzen Sie ZeroStatus zurück.
PosOverflow	OUTPUT	BOOL	FALSE	Statusbit: CountValue hat obere Zählgrenze in positiver Richtung überschritten Über die steigende Flanke des Eingangsparameters EventAck setzen Sie PosOverflow zurück.
NegOverflow	OUTPUT	BOOL	FALSE	Statusbit: CountValue hat untere Zählgrenze in negativer Richtung unterschritten Über die steigende Flanke des Eingangsparameters EventAck setzen Sie NegOverflow zurück.
Error	OUTPUT	BOOL	FALSE	Ein Fehler ist aufgetreten. Die Fehlerursache entnehmen Sie dem Ausgangsparameter ErrorID.
ErrorID	OUTPUT	WORD	0	Parameter ErrorID (Seite 2892) zeigt die Nummer der Fehlermeldung an. ErrorID = 0000_H: Es steht kein Fehler an.
CountValue	OUTPUT	DINT	0	Aktueller Zählwert

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
CapturedValue	OUTPUT	DINT	0	Letzter erfasster Capture-Wert. Der Wert bleibt erhalten bis zum nächsten Capture-Ereignis, unabhängig vom Eingangsparameter CaptureEnable. Wenn ein neues Capture-Ereignis aufgetreten ist, wird Capture-Status gesetzt und von Ihnen über die fallende Flanke des Eingangsparameters CaptureEnable rückgesetzt.
MeasuredValue	OUTPUT	REAL	0.0	Aktueller Messwert für Frequenz, Periodendauer oder Geschwindigkeit (abhängig von der Parametrierung)
CompareMeasuredValue[1]	OUTPUT	BOOL	FALSE	Statusbit: FALSE: Betriebsart "Zählwert als Bezug verwenden"; Vergleichsfunktionen arbeiten mit Zählwert TRUE: Betriebsart "Messwert als Bezug verwenden"; Vergleichsfunktionen arbeiten mit Messwert

[1] Verfügbar für High_Speed_Counter ab Version V3.0

Statische Variablen High_Speed_Counter

Variable	Datentyp	Vorbelegung	Zugriff	Beschreibung
NewCountValue	DINT	L#0	Schreiben	Neuer Zählwert
NewReferenceValue0	DINT	L#0	Schreiben	Neuer Vergleichswert 0 in Betriebsart "Zählwert als Bezug verwenden"
NewReferenceValue1	DINT	L#10	Schreiben	Neuer Vergleichswert 1 in Betriebsart "Zählwert als Bezug verwenden"
NewReferenceValue0_M[1]	REAL	L#0.0	Schreiben	Neuer Vergleichswert 0 in Betriebsart "Messwert als Bezug verwenden"
NewReferenceValue1_M[1]	REAL	L#10.0	Schreiben	Neuer Vergleichswert 1 in Betriebsart "Messwert als Bezug verwenden"
NewUpperLimit	DINT	L#2147483647	Schreiben	Neue obere Zählgrenze
NewLowerLimit	DINT	L#-2147483648	Schreiben	Neue untere Zählgrenze
NewStartValue	DINT	L#0	Schreiben	Neuer Startwert
NewDirection	INT	0	Schreiben	Neue Zählrichtung: +1: Zählrichtung vorwärts -1: Zählrichtung rückwärts
CurReferenceValue0	DINT	L#0	Lesen	Aktueller Vergleichswert 0 in Betriebsart "Zählwert als Bezug verwenden"
CurReferenceValue1	DINT	L#10	Lesen	Aktueller Vergleichswert 1 in Betriebsart "Zählwert als Bezug verwenden"
CurReferenceValue0_M[1]	REAL	L#0.0	Lesen	Aktueller Vergleichswert 0 in Betriebsart "Messwert als Bezug verwenden"
CurReferenceValue1_M[1]	REAL	L#10.0	Lesen	Aktueller Vergleichswert 1 in Betriebsart "Messwert als Bezug verwenden"
CurUpperLimit	DINT	L#2147483647	Lesen	Aktuelle obere Zählgrenze
CurLowerLimit	DINT	L#-2147483648	Lesen	Aktuelle untere Zählgrenze
CurStartValue	DINT	L#0	Lesen	Aktueller Startwert

Variable		Datentyp	Vorbelegung	Zugriff	Beschreibung
AdditionalErrorID		DWORD	W#16#0000	Lesen	Fehlerinformation einer internen Anweisung, z. B. RDREC
UserCmdFlags		STRUCT	-		
	SetNewDirection	BOOL	FALSE	Schreiben	Neue Zählrichtung setzen
	SetUpperLimit	BOOL	FALSE	Schreiben	Obere Zählgrenze setzen
	SetLowerLimit	BOOL	FALSE	Schreiben	Untere Zählgrenze setzen
	SetReferenceValue0	BOOL	FALSE	Schreiben	Vergleichswert 0 setzen
	SetReferenceValue1	BOOL	FALSE	Schreiben	Vergleichswert 1 setzen
	SetStartValue	BOOL	FALSE	Schreiben	Startwert setzen
	SyncDownDirection	BOOL	TRUE	Schreiben	Synchronisation in Zählrichtung rückwärts freigeben
	SyncUpDirection	BOOL	TRUE	Schreiben	Synchronisation in Zählrichtung vorwärts freigeben
	SetDQ0	BOOL	FALSE	Schreiben	Digitalausgang DQ0 setzen
	SetDQ1	BOOL	FALSE	Schreiben	Digitalausgang DQ1 setzen
	ManualCtrlDQ0	BOOL	FALSE	Schreiben	Setzen des Digitalausgangs DQ0 freigeben: TRUE: • SetDQ0 setzt DQ0 • Steuerbit TM_CTRL_DQ0 = FALSE FALSE: • Setzen nicht freigegeben • Steuerbit TM_CTRL_DQ0 = TRUE
	ManualCtrlDQ1	BOOL	FALSE	Schreiben	Setzen des Digitalausgangs DQ1 freigeben: TRUE: • SetDQ1 setzt DQ1 • Steuerbit TM_CTRL_DQ1 = FALSE FALSE: • Setzen nicht freigegeben • Steuerbit TM_CTRL_DQ1 = TRUE
UserStatusFlags		STRUCT	-		
	StatusDI0	BOOL	FALSE	Lesen	Aktueller Status Digitaleingang DI0
	StatusDI1	BOOL	FALSE	Lesen	Aktueller Status Digitaleingang DI1
	StatusDI2	BOOL	FALSE	Lesen	Aktueller Status Digitaleingang DI2
	StatusDQ0	BOOL	FALSE	Lesen	Aktueller Status Digitalausgang DQ0
	StatusDQ1	BOOL	FALSE	Lesen	Aktueller Status Digitalausgang DQ1

[1] Verfügbar für High_Speed_Counter ab Version V3.0

Fehlercodes des Parameters ErrorID

Fehlercode (W#16#...)	Beschreibung
0000	Kein Fehler
Fehlermeldungen vom Technologiemodul	
80A1	POWER_ERROR aus Rückmeldeschnittstelle: Fehlerhafte Versorgungsspannung L+
80A2	ENC_ERROR aus Rückmeldeschnittstelle: Fehlerhaftes Gebersignal
80A3	LD_ERROR aus Rückmeldeschnittstelle: Fehler beim Laden über Steuerschnittstelle
Fehlermeldungen der Anweisung High_Speed_Counter	
80B1	Ungültige Zählrichtung
80B4	Für Betriebsart "Zählwert als Bezug verwenden" gilt: Neue untere Zählgrenze erfüllt folgende Bedingungen nicht: • Untere Zählgrenze < Obere Zählgrenze • Untere Zählgrenze <= Vergleichswert/Startwert Für Betriebsart "Messwert als Bezug verwenden" gilt: Neue untere Zählgrenze erfüllt folgende Bedingungen nicht: • Untere Zählgrenze < Obere Zählgrenze • Untere Zählgrenze <= Startwert
80B5	Für Betriebsart "Zählwert als Bezug verwenden" gilt: Neue obere Zählgrenze erfüllt folgende Bedingungen nicht: • Untere Zählgrenze < Obere Zählgrenze • Obere Zählgrenze >= Vergleichswert/Startwert Für Betriebsart "Messwert als Bezug verwenden" gilt: Neue obere Zählgrenze erfüllt folgende Bedingungen nicht: • Untere Zählgrenze < Obere Zählgrenze • Obere Zählgrenze >= Startwert
80B6	Neuer Startwert erfüllt folgende Bedingung nicht: • Untere Zählgrenze <= Startwert <= Obere Zählgrenze
80B7	Neuer Zählwert erfüllt folgende Bedingung nicht: • Untere Zählgrenze <= Zählwert <= Obere Zählgrenze
80B8	Für Betriebsart "Zählwert als Bezug verwenden" gilt: Neuer Vergleichswert 0 erfüllt folgende Bedingungen nicht: • Untere Zählgrenze <= Vergleichswert 0 <= Obere Zählgrenze • Vergleichswert 0 < Vergleichswert 1 Für Betriebsart "Messwert als Bezug verwenden" gilt: Neuer Vergleichswert 0 erfüllt folgende Bedingung nicht: • Vergleichswert 0 < Vergleichswert 1

Fehlercode (W#16#...)	Beschreibung
80B9	Für Betriebsart "Zählwert als Bezug verwenden" gilt: Neuer Vergleichswert 1 erfüllt folgende Bedingungen nicht: • Untere Zählgrenze <= Vergleichswert 1 <= Obere Zählgrenze • Vergleichswert 0 < Vergleichswert 1 Für Betriebsart "Messwert als Bezug verwenden" gilt: Neuer Vergleichswert 1 erfüllt folgende Bedingung nicht: • Vergleichswert 0 < Vergleichswert 1
80C0	Anweisung High_Speed_Counter wurde mit der gleichen Instanz (DB) mehrfach aufgerufen
80C1	Kommunikation mit dem Technologiemodul fehlgeschlagen (Lese-Datensätze): Fehlerinformation der internen Anweisung RDREC in statischer Variable AdditionalErrorID hinterlegt
80C2	Kommunikation mit dem Technologiemodul fehlgeschlagen (Schreib-Datensätze): Fehlerinformation der internen Anweisung WRREC in statischer Variable AdditionalErrorID hinterlegt
80C3	Zugriff auf Eingangsdaten (Rückmeldeschnittstelle) fehlgeschlagen: Fehlerinformation der internen Anweisung GETIO_PART in statischer Variable AdditionalErrorID hinterlegt
80C4	Zugriff auf Ausgangsdaten (Steuerschnittstelle) fehlgeschlagen: Fehlerinformation der internen Anweisung SETIO_PART in statischer Variable AdditionalErrorID hinterlegt
80C5	Lesen der aktuellen Startinformation des OB fehlgeschlagen: Fehlerinformation der internen Anweisung RD_SINFO in statischer Variable AdditionalErrorID hinterlegt
80C6	Ermitteln der EA-Adressen des Technologiemoduls fehlgeschlagen: Fehlerinformation der internen Anweisung RD_ADDR in statischer Variable AdditionalErrorID hinterlegt
80C7	Modul nicht gesteckt oder Versorgungsspannung L+ fehlt

SSI_Absolute_Encoder

SSI_Absolute_Encoder

Beschreibung SSI_Absolute_Encoder

Beschreibung

Mit der Anweisung SSI_Absolute_Encoder steuern Sie über das Anwenderprogramm die Positionserfassungs- und Messfunktionen des Technologiemoduls TM PosInput.

Aufruf

Die Anweisung muss im Zyklus oder alternativ in einem zeitgesteuerten Programm einmal pro Kanal aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Arbeitsweise

Positionswert: Der Positionswert ist am Ausgangsparameter PositionValue verfügbar. Der Positionswert wird bei jedem Aufruf der Anweisung aktualisiert.

Messwert: Das Technologiemodul aktualisiert den Messwert mit der parametrierten Aktualisierungszeit asynchron zum Aufruf der Anweisung. Bei jedem Aufruf der Anweisung wird der zuletzt vom Technologiemodul ermittelte Messwert am Ausgangsparameter MeasuredValue aktualisiert.

Messwert und Positionswert stehen als Ausgangsparameter parallel zur Verfügung.

Anstelle eines Messwerts kann das vollständige SSI-Telegramm am Ausgangsparameter CompleteSSIFrame zurückgeliefert werden. Je nach Parametrierung ist entweder MeasuredValue oder CompleteSSIFrame gültig.

Capture: Der Ausgangsparameter CaptureStatus = TRUE zeigt einen gültigen Capture-Wert am Ausgangsparameter CapturedValue an.

- Ein Capture-Wert wird unter folgenden Bedingungen erfasst:
  - Ein Digitaleingang hat die Parametrierung "Capture"
  - CaptureEnable = TRUE
  - Flanke am Digitaleingang mit der Capture-Funktion
- Der Ausgangsparameter CaptureStatus wird rückgesetzt durch eine fallende Flanke am Eingangsparameter CaptureEnable.

Parameteränderung über das Anwenderprogramm

Wenn Sie Parameter über das Anwenderprogramm ändern wollen, gehen Sie folgendermaßen vor:

1. Prüfen Sie anhand der jeweiligen Set-Variable, ob das Technologieobjekt für die Parameteränderung bereit ist (Set-Variable = FALSE) oder ob noch ein Änderungsauftrag läuft (Set-Variable = TRUE).
   In den statischen Variablen des Technologieobjekt-Instanz-DB stehen die folgenden Set-Variablen in UserCmdFlags dafür zur Verfügung:
   - SetReferenceValue0
   - SetReferenceValue1
2. Wenn das Technologieobjekt für die Parameteränderung bereit ist, ändern Sie die jeweilige statische Variable.
   Die folgenden statischen Variablen des Technologieobjekt-Instanz-DB stehen dafür zur Verfügung:
   - NewReferenceValue0 / NewReferenceValue0_M (für SetReferenceValue0)
   - NewReferenceValue1 / NewReferenceValue1_M (für SetReferenceValue1)
3. Setzen Sie die jeweilige Set-Variable zur Durchführung des Änderungsauftrags.
4. Prüfen Sie mit Hilfe des Ausgangsparameters Error, ob ein Fehler aufgetreten ist.
   Wenn kein Fehler aufgetreten ist und die Set-Variable durch das Technologieobjekt automatisch rückgesetzt wurde, war die Parameteränderung erfolgreich.

Betriebsart

Die Betriebsart parametrieren Sie im Technologieobjekt unter "Verhalten DQ0".

Die Betriebsart wird durch den Ausgangsparameter CompareMeasuredValue angezeigt:

Zustand	Beschreibung
FALSE	Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden": Die Vergleichsfunktionen arbeiten mit dem Positionswert. Folgende statische Variablen werden spezifisch in dieser Betriebsart verwendet: • NewReferenceValue0 • NewReferenceValue1 • CurReferenceValue0 • CurReferenceValue1 Die vier spezifischen statischen Variablen der Betriebsart "Messwert als Bezug verwenden" werden ignoriert.
TRUE	Betriebsart "Messwert als Bezug verwenden": Die Vergleichsfunktionen arbeiten mit dem Messwert. Folgende statische Variablen werden spezifisch in dieser Betriebsart verwendet: • NewReferenceValue0_M • NewReferenceValue1_M • CurReferenceValue0_M • CurReferenceValue1_M Die vier spezifischen statischen Variablen der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" werden ignoriert.

Quittierung von Ereignissen

Die Quittierung von gemeldeten Ereignissen nehmen Sie über die steigende Flanke des Eingangsparameters EventAck vor. EventAck muss gesetzt bleiben, bis das Technologieobjekt die Statusbits der folgenden Ereignisse des Zählkanals zurückgesetzt hat:

- CompResult0
- CompResult1
- ZeroStatus
- PosOverflow
- NegOverflow

Status der Digitaleingänge

Den Status der Digitaleingänge erhalten Sie über die statischen Variablen StatusDI0 und StatusDI1.

Digitalausgänge durch Anwenderprogramm nutzen

Sie können in folgenden Fällen die Digitalausgänge über die Anweisung setzen:

Fall	Beschreibung
Für "Ausgang setzen" ist die Einstellung "Nutzung durch Anwenderprogramm" parametriert.	Der jeweilige Digitalausgang DQm folgt dem Wert von SetDQm.
Für "Ausgang setzen" ist die Einstellung "Nach Setzbefehl aus CPU bis Vergleichswert" parametriert.	Der jeweilige Digitalausgang DQm wird mit einer steigenden Flanke von SetDQm gesetzt. DQm wird zurückgesetzt, wenn der Positionswert dem Vergleichswert entspricht oder bei einer fallenden Flanke von SetDQm.
Sie setzen die jeweilige statische Variable ManualCtrlDQm (temporäres Überschreiben).	Der jeweilige Digitalausgang DQm folgt dem Wert von SetDQm.

Verhalten im Fehlerfall

Wenn beim Aufruf der Anweisung oder im Technologiemodul ein Fehler aufgetreten ist, wird der Ausgangsparameter Error gesetzt. Die weiterführende Fehlerinformation kann dann am Ausgangsparameter ErrorID (Seite 2899) ausgelesen werden.

Beheben Sie die Ursache des Fehlers und quittieren Sie die Fehlermeldung durch Setzen des Eingangsparameters ErrorAck. Wenn kein Fehler mehr ansteht, setzt das Technologieobjekt den Ausgangsparameter Error zurück. Es wird kein neuer Fehler gemeldet, solange Sie den vorherigen Fehler nicht quittiert haben.

Eingangsparameter SSI_Absolute_Encoder

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
CaptureEnable	INPUT	BOOL	FALSE	Capture-Funktion freigeben Nach der Freigabe erfolgt ein Capture-Ereignis bei der nächsten parametrierten Flanke am jeweiligen Digitaleingang. Eine fallende Flanke an CaptureEnable setzt den Ausgangsparameter Capture-Status zurück. Eine fallende Flanke an CaptureEnable setzt die Freigabe auch dann zurück, wenn kein Capture-Ereignis erfolgt ist. Unabhängig von CaptureEnable bleibt der letzte Wert am Ausgangsparameter CapturedValue erhalten bis zum nächsten Capture-Ereignis.
ErrorAck	INPUT	BOOL	FALSE	SSI_Absolute_Encoder V1.0: Eine steigende Flanke quittiert den gemeldeten Fehlerzustand. SSI_Absolute_Encoder ab V2.0: Ein High-Pegel quittiert den gemeldeten Fehlerzustand.
EventAck	INPUT	BOOL	FALSE	Eine steigende Flanke setzt folgende Ausgangsparameter zurück: • CompResult0 • CompResult1 • ZeroStatus • PosOverflow • NegOverflow

Ausgangsparameter SSI_Absolute_Encoder

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
StatusHW	OUTPUT	BOOL	FALSE	Statusbit Technologiemodul: Das Modul ist parametriert und betriebsbereit. Die Daten des Moduls sind gültig.
StatusUp	OUTPUT	BOOL	FALSE	Statusbit: Letzter Positionswert hat sich in positive Richtung geändert und liegt maximal 0,5 s zurück
StatusDown	OUTPUT	BOOL	FALSE	Statusbit: Letzter Positionswert hat sich in negative Richtung geändert und liegt maximal 0,5 s zurück
CompResult0	OUTPUT	BOOL	FALSE	Statusbit: Vergleichsereignis für DQ0 (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten Über die steigende Flanke des Eingangsparameters EventAck setzen Sie CompResult0 zurück.
CompResult1	OUTPUT	BOOL	FALSE	Statusbit: Vergleichsereignis für DQ1 (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten Über die steigende Flanke des Eingangsparameters EventAck setzen Sie CompResult1 zurück.
CaptureStatus	OUTPUT	BOOL	FALSE	Statusbit: Capture-Ereignis aufgetreten, ein gültiger Capture-Wert liegt vor im Ausgangsparameter CapturedValue Wenn der Eingangsparameter CaptureEnable gesetzt ist, setzt die parametrierte Flanke am jeweiligen Digitaleingang das Statusbit CaptureStatus. Über die fallende Flanke des Eingangsparameters CaptureEnable setzen Sie CaptureStatus zurück.
CapturedValue	OUTPUT	DINT	0	Letzter erfasster Capture-Wert. Der Wert bleibt erhalten bis zum nächsten Capture-Ereignis, unabhängig vom Eingangsparameter CaptureEnable. Wenn ein neues Capture-Ereignis aufgetreten ist, wird CaptureStatus gesetzt und von Ihnen über die fallende Flanke des Eingangsparameters CaptureEnable rückgesetzt.
ZeroStatus	OUTPUT	BOOL	FALSE	Statusbit: PositionValue hat unabhängig von der Zählrichtung den Wert "0" erreicht oder überschritten. Über die steigende Flanke des Eingangsparameters EventAck setzen Sie ZeroStatus zurück.
PosOverflow	OUTPUT	BOOL	FALSE	Statusbit: PositionValue hat die obere Grenze des Positionswertebereichs des Gebers in positiver Richtung überschritten. Über die steigende Flanke des Eingangsparameters EventAck setzen Sie PosOverflow zurück.
NegOverflow	OUTPUT	BOOL	FALSE	Statusbit: PositionValue hat die untere Grenze des Positionswertebereichs des Gebers in negativer Richtung überschritten. Über die steigende Flanke des Eingangsparameters EventAck setzen Sie NegOverflow zurück.
Error	OUTPUT	BOOL	FALSE	Ein Fehler ist aufgetreten. Die Fehlerursache entnehmen Sie dem Ausgangsparameter ErrorID.
ErrorID	OUTPUT	WORD	0	Parameter ErrorID (Seite 2899) zeigt die Nummer der Fehlermeldung an. ErrorID = 0000_H: Es steht kein Fehler an.
PositionValue	OUTPUT	DINT	0	Aktueller Positionswert

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
MeasuredValue	OUTPUT	REAL	0.0	Aktueller Messwert für Frequenz, Periodendauer oder Geschwindigkeit (abhängig von der Parametrierung) Je nach Parametrierung im Technologieobjekt unter "Messwert" ist entweder MeasuredValue oder CompleteSSIFrame gültig.
CompleteSSIFrame	OUTPUT	DWORD	0	Letztes erhaltenes vollständiges SSI-Telegramm (niederwertigste 32 Bit) Je nach Parametrierung im Technologieobjekt unter "Messwert" ist entweder MeasuredValue oder CompleteSSIFrame gültig.
CompareMeasuredValue	OUTPUT	BOOL	FALSE	Statusbit: FALSE: Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"; als Bezug wird Positionswert verwendet TRUE: Betriebsart "Messwert als Bezug verwenden"; als Bezug wird Messwert verwendet

Statische Variablen SSI_Absolute_Encoder

Variable		Datentyp	Vorbelegung	Zugriff	Beschreibung
NewReferenceValue0		DINT	L#0	Schreiben	Neuer Vergleichswert 0 in Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"
NewReferenceValue1		DINT	L#10	Schreiben	Neuer Vergleichswert 1 in Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"
NewReferenceValue0_M		REAL	L#0.0	Schreiben	Neuer Vergleichswert 0 in Betriebsart "Messwert als Bezug verwenden"
NewReferenceValue1_M		REAL	L#10.0	Schreiben	Neuer Vergleichswert 1 in Betriebsart "Messwert als Bezug verwenden"
CurReferenceValue0		DINT	L#0	Lesen	Aktueller Vergleichswert 0 in Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"
CurReferenceValue1		DINT	L#10	Lesen	Aktueller Vergleichswert 1 in Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"
CurReferenceValue0_M		REAL	L#0.0	Lesen	Aktueller Vergleichswert 0 in Betriebsart "Messwert als Bezug verwenden"
CurReferenceValue1_M		REAL	L#10.0	Lesen	Aktueller Vergleichswert 1 in Betriebsart "Messwert als Bezug verwenden"
AdditionalErrorID		DWORD	W#16#0000	Lesen	Fehlerinformation einer internen Anweisung, z. B. RDREC
UserCmdFlags		STRUCT	-		
	SetReferenceValue0	BOOL	FALSE	Schreiben	Vergleichswert 0 setzen
	SetReferenceValue1	BOOL	FALSE	Schreiben	Vergleichswert 1 setzen
	SetDQ0	BOOL	FALSE	Schreiben	Digitalausgang DQ0 setzen
	SetDQ1	BOOL	FALSE	Schreiben	Digitalausgang DQ1 setzen
	ManualCtrlDQ0	BOOL	FALSE	Schreiben	Setzen des Digitalausgangs DQ0 freigeben. TRUE: SetDQ0 setzt DQ0[1] FALSE: Setzen nicht freigegeben

Variable		Datentyp	Vorbelegung	Zugriff	Beschreibung
	ManualCtrlDQ1	BOOL	FALSE	Schreiben	Setzen des Digitalausgangs DQ1 freigeben: TRUE: SetDQ1 setzt DQ1[1] FALSE: Setzen nicht freigegeben
UserStatusFlags		STRUCT	-		
	StatusDI0	BOOL	FALSE	Lesen	Aktueller Status Digitaleingang DI0
	StatusDI1	BOOL	FALSE	Lesen	Aktueller Status Digitaleingang DI1
	StatusDQ0	BOOL	FALSE	Lesen	Aktueller Status Digitalausgang DQ0
	StatusDQ1	BOOL	FALSE	Lesen	Aktueller Status Digitalausgang DQ1

[1] Die Anweisung setzt in der Steuerschnittstelle des Moduls das Bit TM_CTRL_DQm auf FALSE. Die statische Variable SetDQm wirkt auf das Steuerbit SET_DQm.

Fehlercodes des Parameters ErrorID

Fehlercode (W#16#...)	Beschreibung
0000	Kein Fehler
Fehlermeldungen vom Technologiemodul	
80A1	POWER_ERROR aus Rückmeldeschnittstelle: Fehlerhafte Versorgungsspannung L+
80A2	ENC_ERROR aus Rückmeldeschnittstelle: Fehlerhaftes Gebersignal
80A3	LD_ERROR aus Rückmeldeschnittstelle: Fehler beim Laden über Steuerschnittstelle
Fehlermeldungen der Anweisung SSI_Absolute_Encoder	
80B8	Neuer Vergleichswert 0 erfüllt folgende Bedingungen nicht: • Untere Zählgrenze <= Vergleichswert 0 <= Obere Zählgrenze • Vergleichswert 0 < Vergleichswert 1
80B9	Neuer Vergleichswert 1 erfüllt folgende Bedingungen nicht: • Untere Zählgrenze <= Vergleichswert 1 <= Obere Zählgrenze • Vergleichswert 0 < Vergleichswert 1
80C0	Anweisung wurde mit der gleichen Instanz (DB) mehrfach aufgerufen
80C1	Kommunikation mit dem Technologiemodul fehlgeschlagen (Lese-Datensätze): Fehlerinformation der internen Anweisung RDREC in statischer Variable AdditionalErrorID hinterlegt
80C2	Kommunikation mit dem Technologiemodul fehlgeschlagen (Schreib-Datensätze): Fehlerinformation der internen Anweisung WRREC in statischer Variable AdditionalErrorID hinterlegt
80C5	Lesen der aktuellen Startinformation des OB fehlgeschlagen: Fehlerinformation der internen Anweisung RD_SINFO in statischer Variable AdditionalErrorID hinterlegt
80C6	Ermitteln der EA-Adressen des Technologiemoduls fehlgeschlagen: Fehlerinformation der internen Anweisung RD_ADDR in statischer Variable AdditionalErrorID hinterlegt
80C7	Modul nicht gesteckt oder Versorgungsspannung L+ fehlt

4.1.5.3 PID Control

PID_Compact

Neuerungen PID_Compact

PID_Compact V2.4

- **Initialisierung des Integral-Anteils**
  PID_Compact initialisiert den Integral-Anteil nun korrekt, wenn Sie OverwriteInitialOutputValue zusammen mit invertiertem Regelsinn verwenden.
  Falls Sie bisher OverwriteInitialOutputValue zusammen mit invertiertem Regelsinn verwenden, beachten Sie, dass sich mit PID_Compact V2.4 das Vorzeichen des Ausgangswerts ändert.

PID_Compact V2.3

- **Reaktion des Ausgangswerts beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb"**
  Die neue Option IntegralResetMode = 4 wurde hinzugefügt und als Vorbelegung festgelegt. Mit IntegralResetMode = 4 wird der I-Anteil, beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb", automatisch so vorbelegt, dass eine Regeldifferenz zu einem Sprung des Ausgangswerts mit identischem Vorzeichen führt.
- **Initialisierung des Integral-Anteils im Automatikbetrieb**
  Der Integral-Anteil kann im Automatikbetrieb mithilfe der Variablen OverwriteInitialOutputValue und PIDCtrl.PIDInit initialisiert werden. Dies vereinfacht den Einsatz von PID_Compact für Ablöseregelungen.

PID_Compact V2.2

- **Einsatz mit S7-1200**
  Ab PID_Compact V2.2 ist die Anweisung mit V2-Funktionalität auch auf einer S7-1200 ab der Firmware-Version 4.0 einsetzbar.

PID_Compact V2.0

- **Verhalten im Fehlerfall**
  Das Verhalten im Fehlerfall wurde grundlegend überarbeitet. PID_Compact verhält sich in der Default-Einstellung Fehler toleranter. Dieses Verhalten wird beim Kopieren von PID_Compact V1.X von einer S7-1200 CPU auf eine S7-1500 CPU eingestellt.

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn Sie die Default-Einstellung verwenden, bleibt PID_Compact beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

  Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. Mit ErrorAck quittieren Sie die Fehler und Warnungen, ohne den Regler neu zu starten oder den I-Anteil zu löschen. Durch einen Wechsel der Betriebsart werden nicht mehr anstehende Fehler nicht mehr gelöscht.
  Sie konfigurieren das Verhalten im Fehlerfall mit SetSubstituteOutput und ActivateRecoverMode.

- **Ersatzausgangswert**
  Sie können einen Ersatzausgangswert konfigurieren, der im Fehlerfall ausgegeben wird.

- **Wechsel der Betriebsarten**
  Die Betriebsart geben Sie am Durchgangsparameter Mode vor und starten diese über eine steigende Flanke an ModeActivate. Die Variable sRet.i_Mode ist entfallen.

- **Multiinstanzfähigkeit**
  Sie können PID_Compact als Multiinstanz-DB aufrufen. Dann wird kein Technologieobjekt angelegt und es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_Compact direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

- **Anlaufverhalten**
  Die an Mode vorgegebene Betriebsart wird auch gestartet bei einer fallenden Flanke an Reset und bei Kaltstart der CPU, wenn RunModeByStartup = TRUE ist.

- **ENO-Verhalten**
  ENO wird in Abhängigkeit von der Betriebsart gesetzt.
  Wenn State = 0 ist, dann ist ENO = FALSE.
  Wenn State ≠ 0 ist, dann ist ENO = TRUE.

- **Sollwertvorgabe während der Optimierung**
  Die erlaubte Schwankung des Sollwerts während der Optimierung konfigurieren Sie an der Variable CancelTuningLevel.

- **Wertebereich für Ausgangswertgrenzen**
  Der Wert 0.0 muss nicht mehr innerhalb der Ausgangswertgrenzen liegen.

- **I-Anteil vorbelegen**
  Mit den Variablen IntegralResetMode und OverwriteInitialOutputValue bestimmen Sie, wie der I-Anteil beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb" vorbelegt wird.

- **Aufschalten einer Störgröße**
  Sie können am Parameter Disturbance eine Störgröße aufschalten.
- **Vorbelegung der PID-Parameter**
  Folgende Vorbelegungen wurden geändert:
  - Gewichtung des P-Anteils (PWeighting) von 0.0 auf 1.0
  - Gewichtung des D-Anteils (DWeighting) von 0.0 auf 1.0
  - Koeffizient für den Differenzierverzug (TdFiltRatio) von 0.0 auf 0.2
- **Umbenennung der Variablen**
  Die statischen Variablen wurden kompatibel zu PID_3Step neu benannt.

PID_Compact V1.2

- **Handbetrieb beim Anlauf der CPU**
  Wenn beim Start der CPU ManualEnable = TRUE, startet PID_Compact im Handbetrieb. Eine steigende Flanke an ManualEnable ist nicht notwendig.
- **Erstoptimierung**
  Wenn die CPU während der Erstoptimierung ausgeschaltet wird, wird die Erstoptimierung beim Einschalten der CPU erneut gestartet.

PID_Compact V1.1

- **Handbetrieb beim Anlauf der CPU**
  Beim Start der CPU schaltet PID_Compact nur bei einer steigenden Flanke an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID_Compact in der letzten Betriebsart bei der ManualEnable FALSE war.
- **Verhalten bei Reset**
  Durch eine steigende Flanke an Reset werden Fehler und Warnungen zurückgesetzt und der I-Anteil gelöscht. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt.
- **Vorbelegung der Obergrenze Istwert**
  Die Vorbelegung von r_Pv_Hlm wurde zu 120.0 geändert.
- **Überwachung der Abtastzeit**
  - Es wird kein Fehler mehr ausgegeben, wenn aktuelle Abtastzeit ≥ 1,5 x aktueller Mittelwert oder Aktuelle Abtastzeit ≤ 0,5 x aktueller Mittelwert. Im Automatikbetrieb darf die Abtastzeit stärker abweichen.
  - PID_Compact ist kompatibel mit FW ab V2.0.

- **Zugriff auf Variablen**
  Die folgenden Variablen können jetzt im Anwenderprogramm verwendet werden.
  - i_Event_SUT
  - i_Event_TIR
  - r_Ctrl_Ioutv
- **Fehlerbehebung**
  PID_Compact gibt jetzt korrekte Impulse aus, wenn die kleinste Einschaltzeit ungleich der kleinsten Ausschaltzeit ist.

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von PID_Compact einsetzen können.

CPU	FW	PID_Compact
S7-1200	ab V4.2	V2.3 V2.2 V1.2
	V4.0 bis V4.1	V2.2 V1.2
	V3.x	V1.2 V1.1
	V2.x	V1.2 V1.1
	V1.x	V1.0
S7-1500	ab V2.5	V2.4 V2.3 V2.2 V2.1 V2.0
	V2.0 und V2.1	V2.3 V2.2 V2.1 V2.0
	V1.5 bis V1.8	V2.2 V2.1 V2.0
	V1.1	V2.1 V2.0
	V1.0	V2.0

CPU-Bearbeitungszeit und Speicherbedarf PID_Compact V2.x

CPU-Bearbeitungszeit

Typische CPU-Bearbeitungszeiten des Technologieobjekts PID_Compact ab der Version V2.0 in Abhängigkeit vom CPU-Typ.

CPU	typ. CPU-Bearbeitungszeit PID_Compact V2.x
CPU 1211C ≥ V4.0	300 µs
CPU 1215C ≥ V4.0	300 µs
CPU 1217C ≥ V4.0	300 µs
CPU 1505S ≥ V1.0	45 µs
CPU 1510SP-1 PN ≥ V1.6	85 µs
CPU 1511-1 PN ≥ V1.5	85 µs
CPU 1512SP-1 PN ≥ V1.6	85 µs
CPU 1516-3 PN/DP ≥ V1.5	50 µs
CPU 1518-4 PN/DP ≥ V1.5	4 µs

Speicherbedarf

Speicherbedarf eines Instanz-DB des Technologieobjekts PID_Compact ab der Version V2.0.

Speicherbedarf	Speicherbedarf des Instanz-DB von PID_Compact V2.x
Ladespeicherbedarf	ca. 12000 Bytes
Gesamter Arbeitsspeicherbedarf	788 Bytes
Remanenter Arbeitsspeicherbedarf	44 Bytes

PID_Compact V2

Beschreibung PID_Compact V2

Beschreibung

Die Anweisung PID_Compact stellt einen PID-Regler mit integrierter Optimierung für proportional wirkende Stellglieder zur Verfügung.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert mit Fehlerüberwachung

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

PID-Algorithmus

PID_Compact ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der PID-Algorithmus arbeitet nach folgender Formel:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Symbol	Beschreibung
y	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
T_D	Differenzierzeit
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)
c	Gewichtung des D-Anteils

Blockschaltbild PID_Compact

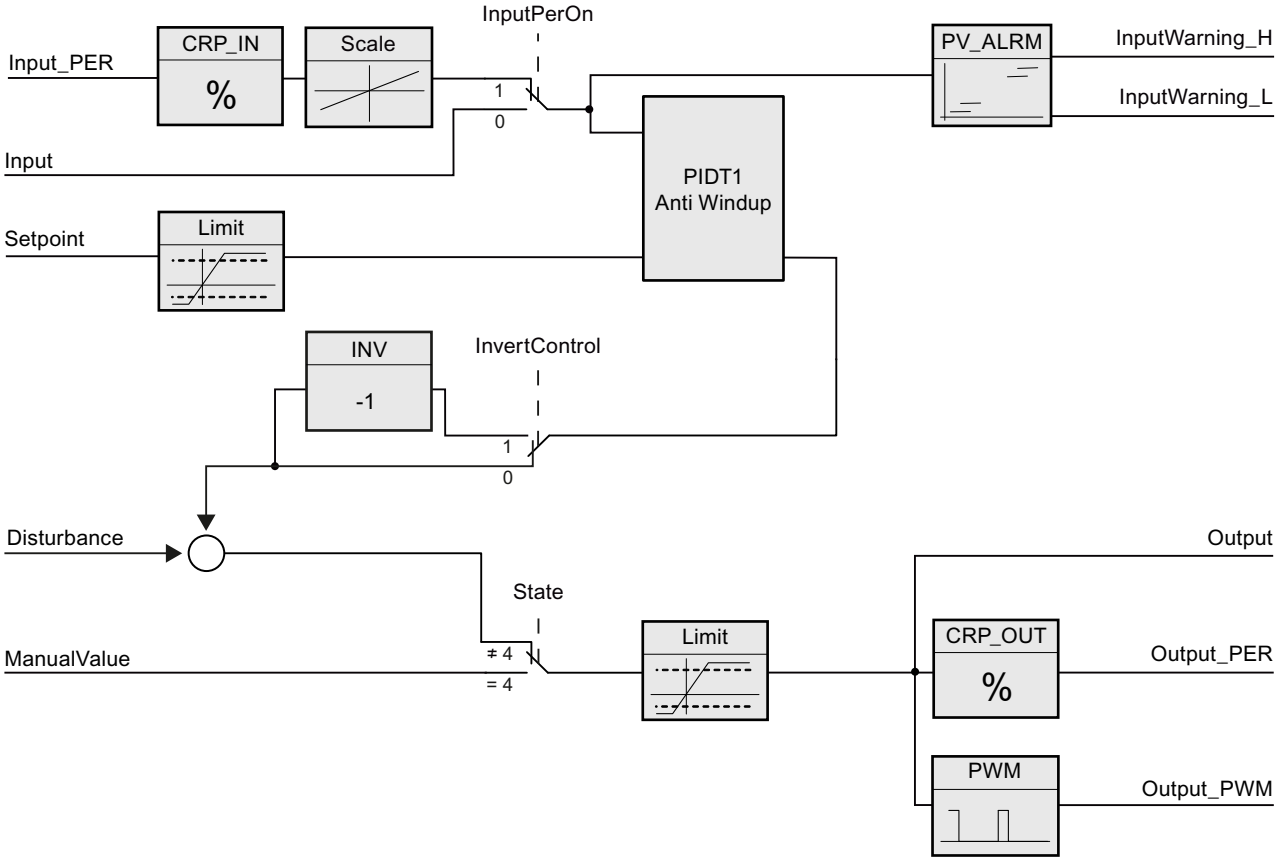

Blockschaltbild PIDT1 mit Anti-Windup

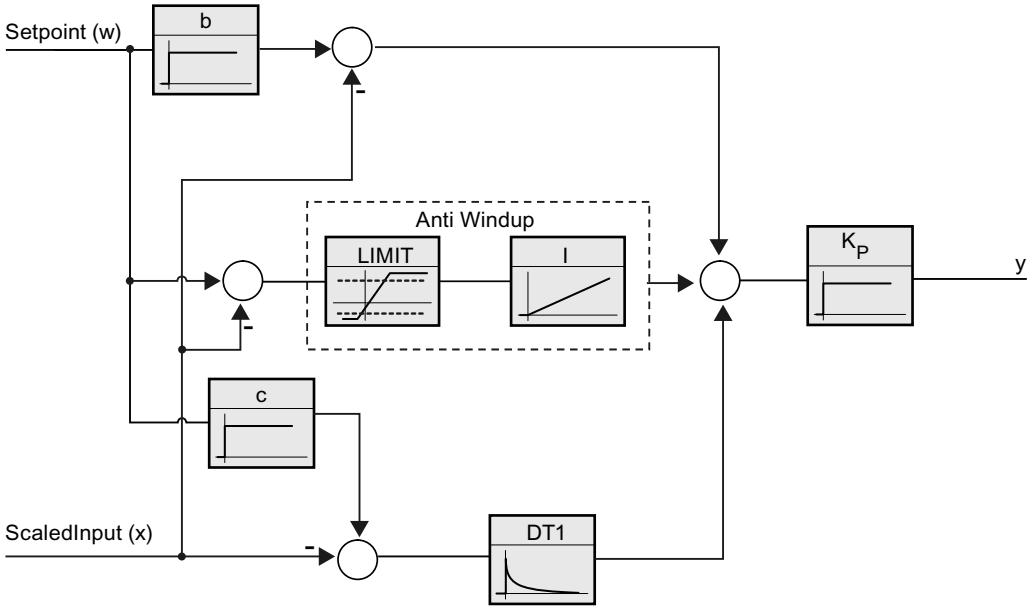

Aufruf

PID_Compact wird im konstanten Zeitraster eines Weckalarm-OBs aufgerufen.

Wenn Sie PID_Compact als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_Compact direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID_Compact vollständig laden.

AUTOHOTSPOT

Anlauf

Bei Anlauf der CPU startet PID_Compact in der Betriebsart, die am Durchgangsparameter Mode gespeichert ist. Um beim Anlauf in die Betriebsart "Inaktiv" zu wechseln, setzen Sie RunModeByStartup = FALSE.

Verhalten im Fehlerfall

Im Automatikbetrieb und während der Inbetriebnahme hängt das Verhalten im Fehlerfall von den Variablen SetSubstituteOutput und ActivateRecoverMode ab. Im Handbetrieb ist das Verhalten unabhängig von SetSubstituteOutput und ActivateRecoverMode. Wenn ActivateRecoverMode = TRUE ist, hängt das Verhalten zusätzlich vom aufgetretenen Fehler ab.

SetSubstituteOutput	ActivateRecoverMode	Konfigurationseditor > Ausgangswert > Output setzen auf	Verhalten
nicht relevant	FALSE	Null (Inaktiv)	Wechsel in Betriebsart "Inaktiv" (State = 0) Der Wert 0.0 0 wird an das Stellglied übertragen.
FALSE	TRUE	Aktuellen Ausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5) Der aktuelle Ausgangswert wird für die Fehlerdauer an das Stellglied übertragen.
TRUE	TRUE	Ersatzausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5) Der Wert an SubstituteOutput wird für die Fehlerdauer an das Stellglied übertragen.

PID_Compact verwendet im Handbetrieb ManualValue als Ausgangswert, außer ManualValue ist ungültig. Wenn ManualValue ungültig ist, wird SubstituteOutput verwendet. Wenn ManualValue und SubstituteOutput ungültig sind, wird Config.OutputLowerLimit verwendet.

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits wird durch eine steigende Flanke an Reset oder ErrorAck zurückgesetzt.

Arbeitsweise PID_Compact V2

Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001h).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warning = 0040h) und der Ausgangsparameter InputWarning_H oder InputWarning_L wird TRUE.

Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID_Compact begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID_Compact prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit_H oder SetpointLimit_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Output, ManualValue und SubstituteOutput werden auf diese Werte begrenzt. Die Ausgangswertgrenzen müssen zum Regelsinn passen.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

Output	-100.0 bis 100.0 %
Output_PER	-100.0 bis 100.0 %
Output_PWM	0.0 bis 100.0 %

Es muss gelten:

OutputUpperLimit > OutputLowerLimit

Hinweis

Einsatz mit zwei oder mehr Stellgliedern

PID_Compact ist nicht für den Einsatz mit zwei oder mehr Stellgliedern (z. B. in Heizen-/Kühlen-Anwendungen) geeignet, da unterschiedliche Stellglieder auch unterschiedliche PID-Parameter benötigen, um ein gutes Regelverhalten zu erzielen. Verwenden Sie für Anwendungen mit zwei entgegengesetzt wirkenden Stellgliedern PID_Temp.

Ersatzausgangswert

PID_Compact kann im Fehlerfall einen Ersatzausgangswert ausgeben, den Sie an der Variablen SubstituteOutput vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden bei Verwendung auf Gültigkeit überwacht:

- Setpoint
- Input
- Input_PER
- Disturbance
- ManualValue
- SubstituteOutput
- Output
- Output_PER
- Output_PWM

Überwachung der Abtastzeit PID_Compact

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID_Compact misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (Error = 0800h).

Während der Optimierung tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert

Im Automatikbetrieb tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,5 x alter Mittelwert
- Neuer Mittelwert <= 0,5 x alter Mittelwert

Wenn Sie die Überwachung der Abtastzeit ausschalten (CycleTime.EnMonitoring = FALSE), können Sie PID_Compact auch im OB1 aufrufen. Sie müssen dann aufgrund der schwankenden Abtastzeit eine schlechtere Regelqualität akzeptieren.

Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID_Compact arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

Eingangsparameter PID_Compact V2

Tabelle 4-58

Parameter	Datentyp	Vorbelegung	Beschreibung
Setpoint	REAL	0.0	Sollwert des PID-Reglers im Automatikbetrieb
Input	REAL	0.0	Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.
Input_PER	INT	0	Ein Analogeingang wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.
Disturbance	REAL	0.0	Störgröße oder Vorsteuerungswert
ManualEnable	BOOL	FALSE	• Flanke FALSE -> TRUE aktiviert die Betriebsart "Handbetrieb", State = 4, Mode bleibt unverändert. Solange ManualEnable = TRUE ist, können Sie die Betriebsart nicht über eine steigende Flanke an ModeActivate ändern und den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE aktiviert die Betriebsart, die durch Mode vorgegeben wird. Es wird empfohlen, die Betriebsart nur über ModeActivate zu ändern.
ManualValue	REAL	0.0	Handwert Dieser Wert wird im Handbetrieb als Ausgangswert verwendet. Zulässig sind Werte von Config.OutputLowerLimit bis Config.OutputUpperLimit
ErrorAck	BOOL	FALSE	• Flanke FALSE -> TRUE ErrorBits und Warning werden zurückgesetzt.

Parameter	Datentyp	Vorbelegung	Beschreibung
Reset	BOOL	FALSE	Führt einen Neustart des Reglers durch. • Flanke FALSE -> TRUE – Wechsel in Betriebsart "Inaktiv" – ErrorBits und Warnings werden zurückgesetzt. • Solange Reset = TRUE ist, – bleibt PID_Compact in der Betriebsart "Inaktiv" (State = 0). – können Sie die Betriebsart nicht über Mode und ModeActivate oder ManualEnable ändern. – können Sie den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE – Falls ManualEnable = FALSE, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist. – Falls Mode = 3, wird der I-Anteil behandelt wie durch die Variable IntegralResetMode konfiguriert.
ModeActivate	BOOL	FALSE	• Flanke FALSE -> TRUE PID_Compact wechselt in die Betriebsart, die an Mode gespeichert ist.

Ausgangsparameter PID_Compact V2

Tabelle 4-59

Parameter	Datentyp	Vorbelegung	Beschreibung
ScaledInput	REAL	0.0	Skalierter Istwert
Die Ausgänge "Output", "Output_PER" und "Output_PWM" können parallel genutzt werden.			
Output	REAL	0.0	Ausgangswert im REAL-Format
Output_PER	INT	0	Analoger Ausgangswert
Output_PWM	BOOL	FALSE	Pulsweitenmodulierter Ausgangswert Der Ausgangswert wird über variable Ein- und Ausschaltzeiten gebildet.
SetpointLimit_H	BOOL	FALSE	Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht (Setpoint ≥ Config.SetpointUpperLimit). Der Sollwert wird auf Config.SetpointUpperLimit begrenzt.
SetpointLimit_L	BOOL	FALSE	Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht (Setpoint ≤ Config.SetpointLowerLimit). Der Sollwert wird auf Config.SetpointLowerLimit begrenzt.
InputWarning_H	BOOL	FALSE	Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.

Parameter	Datentyp	Vorbelegung	Beschreibung
InputWarning_L	BOOL	FALSE	Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.
State	INT	0	Der Parameter State (Seite 2922) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit dem Eingangsparameter Mode und einer steigenden Flanke an ModeActivate. • State = 0: Inaktiv • State = 1: Erstoptimierung • State = 2: Nachoptimierung • State = 3: Automatikbetrieb • State = 4: Handbetrieb • State = 5: Ersatzausgangswert mit Fehlerüberwachung
Error	BOOL	FALSE	Wenn Error = TRUE, liegt in diesem Takt mindestens eine Fehlermeldungen vor.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 2926) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.

Durchgangsparameter PID_Compact V2

Tabelle 4-60

Parameter	Datentyp	Vorbelegung	Beschreibung
Mode	INT	4	An Mode geben Sie die Betriebsart vor, in die PID_Compact wechseln soll. Möglich sind: • Mode = 0: Inaktiv • Mode = 1: Erstoptimierung • Mode = 2: Nachoptimierung • Mode = 3: Automatikbetrieb • Mode = 4: Handbetrieb Die Betriebsart wird aktiviert durch: • Steigende Flanke an ModeActivate • Fallende Flanke an Reset • Fallende Flanke an ManualEnable • Kaltstart der CPU, wenn RunModeByStartup = TRUE Mode ist remanent. Eine detaillierte Beschreibung der Betriebsarten finden Sie unter Parameter State und Mode V2 (Seite 2922) .

Siehe auch

Parameter State und Mode V2 (Seite 2922)

Statische Variablen PID_Compact V2

Hinweis

Ändern Sie die mit (1) gekennzeichneten Variablen nur in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden.

Variable	Datentyp	Vorbelegung	Beschreibung
IntegralResetMode	INT	bis V2.2: 1, ab V2.3: 4	Die Variable IntegralResetMode V2 (Seite 2930) bestimmt, wie der I-Anteil PIDCtrl.IntegralSum beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb" vorbelegt wird. Diese Einstellung wirkt nur für einen Zyklus. Möglich sind: • IntegralResetMode = 0: Glätten • IntegralResetMode = 1: Löschen • IntegralResetMode = 2: Halten • IntegralResetMode = 3: Vorbelegen • IntegralResetMode = 4: Wie Sollwertänderung (nur für PID_Compact mit Version ≥ 2.3)
OverwriteInitialOutputValue	REAL	0.0	Falls eine der folgenden Bedingungen erfüllt ist, wird der Integral-Anteil PIDCtrl.IntegralSum automatisch so vorbelegt, als ob im vorherigen Zyklus Output = OverwriteInitialOutputValue gewesen wäre: • IntegralResetMode = 3 beim Wechsel von Betriebsart "Inaktiv" in "Automatikbetrieb • Flanke TRUE -> FALSE an Parameter Reset und Parameter Mode = 3 • PIDCtrl.PIDInit = TRUE im “Automatikbetrieb” (ab PID_Compact Version 2.3 verfügbar)
RunModeByStartup	BOOL	TRUE	Nach CPU Neustart Betriebsart an Mode aktivieren Wenn RunModeByStartup = TRUE, startet PID_Compact nach CPU-Anlauf in der Betriebsart, die an Mode gespeichert ist. Wenn RunModeByStartup = FALSE, bleibt PID_Compact nach CPU-Anlauf in der Betriebsart "Inaktiv".
LoadBackUp	BOOL	FALSE	Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.
PhysicalUnit	INT	0	Physikalische Einheit des Ist- und Sollwerts, z. B. ºC oder ºF.

Variable	Datentyp	Vorbelegung	Beschreibung
PhysicalQuantity	INT	0	Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur.
ActivateRecoverMode	BOOL	TRUE	Die Variable ActivateRecoverMode V2 (Seite 2928) bestimmt das Verhalten im Fehlerfall.
Warning	DWORD	0	Die Variable Warning V2 (Seite 2930) zeigt die Warnungen seit Reset = TRUE oder ErrorAck =TRUE. Warning ist remanent.
Progress	REAL	0.0	Fortschritt der Optimierung in Prozent (0.0 - 100.0)
CurrentSetpoint	REAL	0.0	CurrentSetpoint zeigt immer den aktuellen Sollwert an. Dieser Wert wird während der Optimierung eingefroren.
CancelTuningLevel	REAL	10.0	Zulässige Schwankung des Sollwerts während der Optimierung. Die Optimierung wird erst abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel
SubstituteOutput	REAL	0.0	Ersatzausgangswert Wenn die folgenden Bedingungen erfüllt sind, wird der Ersatzausgangswert verwendet: • Im Automatikbetrieb ist ein Fehler aufgetreten. • SetSubstituteOutput = TRUE • ActivateRecoverMode = TRUE
SetSubstituteOutput	BOOL	TRUE	Wenn SetSubstituteOutput = TRUE und ActivateRecoverMode = TRUE sind, wird der konfigurierte Ersatzausgangswert ausgegeben, solange ein Fehler ansteht. Wenn SetSubstituteOutput = FALSE und ActivateRecoverMode = TRUE sind, bleibt das Stellglied auf dem aktuellen Ausgangswert, solange ein Fehler ansteht. Wenn ActivateRecoverMode = FALSE ist, wirkt sich SetSubstituteOutput nicht aus. Wenn SubstituteOutput ungültig ist (ErrorBits = 20000h), kann der Ersatzausgangswert nicht ausgegeben werden.
Config.InputPerOn(1)	BOOL	TRUE	Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.
Config.InvertControl(1)	BOOL	FALSE	Invertieren des Regelsinns Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.InputUpperLimit(1)	REAL	120.0	Obergrenze des Istwert Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird mit der Voreinstellung kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_Compact verhält sich wie unter Verhalten im Fehlerfall konfiguriert. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit(1)	REAL	0.0	Untergrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. InputLowerLimit < InputUpperLimit
Config.InputUpperWarning(1)	REAL	3.402822e+38	Obere Warngrenze des Istwerts Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als obere Warngrenze verwendet. Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning(1)	REAL	-3.402822e+38	Untere Warngrenze des Istwerts Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als untere Warngrenze verwendet. Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit
Config.OutputUpperLimit(1)	REAL	100.0	Obergrenze des Ausgangswerts Details siehe OutputLowerLimit OutputUpperLimit > OutputLowerLimit
Config.OutputLowerLimit(1)	REAL	0.0	Untergrenze des Ausgangswerts Für Output und Output_PER gilt der Wertebereich von -100.0 bis +100.0 einschließlich Null. Bei -100.0 beträgt Output_PER = -27648; bei +100.0 beträgt Output_PER = 27648. Für Output_PWM gilt der Wertebereich 0.0 bis +100.0. Die Ausgangswertgrenzen müssen zum Regelsinn passen. OutputLowerLimit < OutputUpperLimit

Variable	Datentyp	Vorbelegung	Beschreibung
Config.SetpointUpperLimit(1)	REAL	3.402822e+38	Obergrenze des Sollwerts Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert verwendet. Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.
Config.SetpointLowerLimit(1)	REAL	-3.402822e +38	Untergrenze des Sollwerts Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert verwendet. Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
Config.MinimumOnTime(1)	REAL	0.0	Minimale Einschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf MinimumOnTime = n×CycleTime.Value
Config.MinimumOffTime(1)	REAL	0.0	Minimale Ausschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf MinimumOffTime = n×CycleTime.Value
Config.InputScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Input_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointIn(1)	REAL	0.0	Skalierung Input_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.
Config.InputScaling.UpperPointOut(1)	REAL	100.0	Skalierter oberer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointOut(1)	REAL	0.0	Skalierter unterer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.
CycleTime.StartEstimation	BOOL	TRUE	Wenn CycleTime.StartEstimation = TRUE, wird die automatische Ermittlung der Zykluszeit gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Wenn CycleTime.EnEstimation = TRUE, wird die Abtastzeit PID_Compact berechnet. Wenn CycleTime.EnEstimation = FALSE, wird die Abtastzeit PID_Compact nicht berechnet und Sie müssen CycleTime.Value manuell korrekt konfigurieren.

Variable	Datentyp	Vorbelegung	Beschreibung
CycleTime.EnMonitoring	BOOL	TRUE	Wenn CycleTime.EnMonitoring = FALSE, wird die Abtastzeit PID_Compact nicht überwacht. Wenn PID_Compact nicht innerhalb der Abtastzeit ausgeführt werden kann, wird kein Fehler (ErrorBits=0800h) ausgegeben und PID_Compact wechselt nicht in die Betriebsart "Inaktiv".
CycleTime.Value(1)	REAL	0.1	Abtastzeit PID_Compact in Sekunden CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.
CtrlParamsBackUp.Gain	REAL	1.0	gespeicherte Proportionalverstärkung Werte aus der Struktur CtrlParamsBackUp können mit LoadBackUp = TRUE wieder geladen werden.
CtrlParamsBackUp.Ti	REAL	20.0	gespeicherte Integrationszeit [s]
CtrlParamsBackUp.Td	REAL	0.0	gespeicherte Differenzierzeit [s]
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	gespeicherter Koeffizient Differenzierverzug
CtrlParamsBackUp.PWeighting	REAL	1.0	gespeicherter Gewichtungsfaktor des P-Anteils
CtrlParamsBackUp.DWeighting	REAL	1.0	gespeicherter Gewichtungsfaktor des D-Anteils
CtrlParamsBackUp.Cycle	REAL	1.0	gespeicherte Abtastzeit PID-Algorithmus
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn SUT.CalculateParams = TRUE, werden anhand dieser Eigenschaften die Parameter für die Erstoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen. SUT.CalculateParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.SUT.TuneRule	INT	0	Parameter während Erstoptimierung berechnen nach Methode: • SUT.TuneRule = 0: PID nach Chien, Hrones und Reswick • SUT.TuneRule = 1: PI nach Chien, Hrones und Reswick
PIDSelfTune.SUT.State	INT	0	Die Variable SUT.State zeigt die aktuelle Phase der Erstoptimierung: • State = 0: Erstoptimierung initialisieren • State = 100: Standardabweichung berechnen • State = 200: Wendepunkt ermitteln • State = 9900: Erstoptimierung erfolgreich • State = 1: Erstoptimierung nicht erfolgreich

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Mit der Variable RunIn können Sie festlegen, dass eine Nachoptimierung auch ohne Erstoptimierung durchgeführt wird. • RunIn = FALSE Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstoptimierung gestartet. Wenn die Voraussetzungen für Erstoptimierung nicht erfüllt sind, verhält sich PID_Compact wie bei RunIn = TRUE. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_Compact in die Betriebsart, aus der die Optimierung gestartet wurde. • RunIn = TRUE Die Erstoptimierung wird übersprungen. PID_Compact versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet anschließend automatisch. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.
PIDSelfTune.TIR.CalculateParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn TIR.CalculateParams = TRUE, werden anhand dieser Eigenschaften die Parameter für die Nachoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen. TIR.CalculateParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.TIR.TuneRule	INT	0	Parameter während Nachoptimierung berechnen nach Methode: • TIR.TuneRule = 0: PID automatisch • TIR.TuneRule = 1: PID schnell • TIR.TuneRule = 2: PID langsam • TIR.TuneRule = 3: Ziegler-Nichols PID • TIR.TuneRule = 4: Ziegler-Nichols PI • TIR.TuneRule = 5: Ziegler-Nichols P

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.TIR.State	INT	0	Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung": • State = -100 Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt. • State = 0: Nachoptimierung initialisieren • State = 200: Standardabweichung berechnen • State = 300: Versuchen Sollwert zu erreichen • State = 400: Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war) • State = 500: Oszillation ermitteln und Parameter berechnen • State = 9900: Nachoptimierung erfolgreich • State = 1: Nachoptimierung nicht erfolgreich
PIDCtrl.IntegralSum(1)	REAL	0.0	Aktueller I-Anteil
PIDCtrl.PIDInit	BOOL	FALSE	PIDCtrl.PIDInit ist ab PID_Compact Version 2.3 verfügbar. Falls im "Automatikbetrieb" PIDCtrl.PIDInit = TRUE, wird der Integral-Anteil PIDCtrl.IntegralSum automatisch so vorbelegt, als ob im vorherigen Zyklus Output = OverwriteInitialOutputValue gewesen wäre. Dies kann für eine AUTOHOTSPOT genutzt werden.
Retain.CtrlParams.Gain(1)	REAL	1.0	Aktive Proportionalverstärkung Verwenden Sie zum Invertieren des Regelsinns die Variable Config.InvertControl. Negative Werte an Gain invertieren den Regelsinn ebenfalls. Es wird empfohlen den Regelsinn nur über InvertControl einzustellen. Wenn InvertControl = TRUE und Gain < 0.0 sind, ist der Regelsinn auch invertiert. Gain ist remanent.
Retain.CtrlParams.Ti(1)	REAL	20.0	• CtrlParams.Ti > 0.0: Aktive Integrationszeit • CtrlParams.Ti = 0.0: I-Anteil ist ausgeschaltet Ti ist remanent.
Retain.CtrlParams.Td(1)	REAL	0.0	• CtrlParams.Td > 0.0: Aktive Differenzierzeit • CtrlParams.Td = 0.0: D-Anteil ist ausgeschaltet Td ist remanent.

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.CtrlParams.TdFiltRatio(1)	REAL	0.2	Aktiver Koeffizient Differenzierverzug Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierverzug verzögert. Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug • 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam. • 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt. • > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert. TdFiltRatio ist remanent.
Retain.CtrlParams.PWeighting(1)	REAL	1.0	Aktive Gewichtung des P-Anteils Sie können bei Sollwertänderungen den P-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: P-Anteil bei Sollwertänderung voll wirksam • 0.0: P-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam. PWeighting ist remanent.
Retain.CtrlParams.DWeighting(1)	REAL	1.0	Aktive Gewichtung des D-Anteils Sie können bei Sollwertänderungen den D-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: D-Anteil bei Sollwertänderung voll wirksam • 0.0: D-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam. DWeighting ist remanent.
Retain.CtrlParams.Cycle(1)	REAL	1.0	Aktive Abtastzeit des PID-Algorithmus CtrlParams.Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von CycleTime.Value aufgerundet. CtrlParams.Cycle wird als Periodendauer der Pulsweitenmodulation verwendet. Cycle ist remanent.

Siehe auch

Variable ActivateRecoverMode V2 (Seite 2928)

Variable Warning V2 (Seite 2930)

Änderungen der Schnittstelle PID_Compact V2

Die folgende Tabelle zeigt, was sich an der Schnittstelle der Anweisung PID_Compact geändert hat.

PID_Compact V1	PID_Compact V2	Änderung
Input_PER	Input_PER	Datentyp von Word zu Int
	Disturbance	Neu
	ErrorAck	Neu
	ModeActivate	Neu
Output_PER	Output_PER	Datentyp von Word zu Int
Error	ErrorBits	Umbenannt
	Error	Neu
	Mode	Neu
sb_RunModeByStartup	RunModeByStartup	Funktion
	IntegralResetMode	
	OverwriteInitialOutputValue	Neu
	SetSubstituteOutput	Neu
	CancelTuningLevel	Neu
	SubstituteOutput	Neu

Die folgende Tabelle zeigt, welche Variablen umbenannt wurden.

PID_Compact V1.x	PID_Compact V2
sb_GetCycleTime	CycleTime.StartEstimation
sb_EnCyclEstimation	CycleTime.EnEstimation
sb_EnCyclMonitoring	CycleTime.EnMonitoring
sb_RunModeByStartup	RunModeByStartup
si_Unit	PhysicalUnit
si_Type	PhysicalQuantity
sd_Warning	Warning
sBackUp.r_Gain	CtrlParamsBackUp.Gain
sBackUp.r_Ti	CtrlParamsBackUp.Ti
sBackUp.r_Td	CtrlParamsBackUp.Td
sBackUp.r_A	CtrlParamsBackUp.TdFiltRatio
sBackUp.r_B	CtrlParamsBackUp.PWeighting
sBackUp.r_C	CtrlParamsBackUp.DWeighting
sBackUp.r_Cycle	CtrlParamsBackUp.Cycle
sPid_Calc.r_Cycle	CycleTime.Value
sPid_Calc.b_RunIn	PIDSelfTune.TIR.RunIn
sPid_Calc.b_CalcParamSUT	PIDSelfTune.SUT.CalculateParams
sPid_Calc.b_CalcParamTIR	PIDSelfTune.TIR.CalculateParams
sPid_Calc.i_CtrlTypeSUT	PIDSelfTune.SUT.TuneRule
sPid_Calc.i_CtrlTypeTIR	PIDSelfTune.TIR.TuneRule
sPid_Calc.r_Progress	Progress

PID_Compact V1.x	PID_Compact V2
sPid_Cmpt.r_Sp_Hlm	Config.SetpointUpperLimit
sPid_Cmpt.r_Sp_Llm	Config.SetpointLowerLimit
sPid_Cmpt.r_Pv_Norm_IN_1	Config.InputScaling.LowerPointIn
sPid_Cmpt.r_Pv_Norm_IN_2	Config.InputScaling.UpperPointIn
sPid_Cmpt.r_Pv_Norm_OUT_1	Config.InputScaling.LowerPointOut
sPid_Cmpt.r_Pv_Norm_OUT_2	Config.InputScaling.UpperPointOut
sPid_Cmpt.r_Lmn_Hlm	Config.OutputUpperLimit
sPid_Cmpt.r_Lmn_Llm	Config.OutputLowerLimit
sPid_Cmpt.b_Input_PER_On	Config.InputPerOn
sPid_Cmpt.b_LoadBackUp	LoadBackUp
sPid_Cmpt.b_InvCtrl	Config.InvertControl
sPid_Cmpt.r_Lmn_Pwm_PPTm	Config.MinimumOnTime
sPid_Cmpt.r_Lmn_Pwm_PBTm	Config.MinimumOffTime
sPid_Cmpt.r_Pv_Hlm	Config.InputUpperLimit
sPid_Cmpt.r_Pv_Llm	Config.InputLowerLimit
sPid_Cmpt.r_Pv_HWrn	Config.InputUpperWarning
sPid_Cmpt.r_Pv_LWrn	Config.InputLowerWarning
sParamCalc.i_Event_SUT	PIDSelfTune.SUT.State
sParamCalc.i_Event_TIR	PIDSelfTune.TIR.State
sRet.i_Mode	sRet.i_Mode ist entfallen. Die Betriebsart wird durch Mode und ModeActivate gewechselt.
sRet.r_Ctrl_Gain	Retain.CtrlParams.Gain
sRet.r_Ctrl_Ti	Retain.CtrlParams.Ti
sRet.r_Ctrl_Td	Retain.CtrlParams.Td
sRet.r_Ctrl_A	Retain.CtrlParams.TdFiltRatio
sRet.r_Ctrl_B	Retain.CtrlParams.PWeighting
sRet.r_Ctrl_C	Retain.CtrlParams.DWeighting
sRet.r_Ctrl_Cycle	Retain.CtrlParams.Cycle

Parameter State und Mode V2

Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Mit einer steigenden Flanke an ModeActivate wechselt PID_Compact in die Betriebsart, die am Durchgangsparameter Mode gespeichert ist.

Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_Compact in der Betriebsart, die an Mode gespeichert ist. Um PID_Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

Bedeutung der Werte

State / Mode	Beschreibung der Betriebsart
0	Inaktiv In der Betriebsart "Inaktiv" wird immer der Ausgangswert 0.0 ausgegeben, unabhängig von Config.OutputUpperLimit und Config.OutputLowerLimit. Die Pulsweitenmodulation ist ausgeschaltet.
1	Erstoptimierung Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. Voraussetzungen für die Erstoptimierung: • Betriebsart Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3) • ManualEnable = FALSE • Reset = FALSE • Der Istwert darf nicht zu nah am Sollwert sein. |Setpoint - Input| > 0.3 * | Config.InputUpperLimit - Config.InputLowerLimit| und |Setpoint - Input| > 0.5 * |Setpoint| • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden. Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode. Die Phase der Erstoptimierung wird mit PIDSelfTune.SUT.State angezeigt.

State / Mode	Beschreibung der Betriebsart
2	Nachoptimierung Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden. Voraussetzungen für Nachoptimierung: • Es werden keine Störungen erwartet. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen • ManualEnable = FALSE • Reset = FALSE • Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4) Die Nachoptimierung verläuft folgendermaßen beim Start aus: • Automatikbetrieb (State = 3) Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb. PID_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. • Inaktiv (State = 0) oder Handbetrieb (State = 4) Wenn die Voraussetzungen für eine Erstoptimierung erfüllt sind, wird eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Wenn sich der Istwert für eine Erstoptimierung bereits zu nah am Sollwert befindet oder PIDSelfTune.TIR.RunIn = TRUE, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Erst dann startet die Nachoptimierung. Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Nachoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode. Die Phase der "Nachoptimierung" wird mit PIDSelfTune.TIR.State angezeigt.
3	Automatikbetrieb Im Automatikbetrieb regelt PID_Compact die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt: • Erstoptimierung erfolgreich abgeschlossen • Nachoptimierung erfolgreich abgeschlossen • Ändern des Durchgangsparameters Mode auf den Wert 3 und eine steigende Flanke an ModeActivate. Der Wechsel vom Automatikbetrieb in den Handbetrieb erfolgt nur im Inbetriebnahmeeditor stoßfrei. Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.

State / Mode	Beschreibung der Betriebsart
4	Handbetrieb Im Handbetrieb geben Sie einen manuellen Ausgangswert am Parameter ManualValue vor. Diese Betriebsart können Sie zusätzlich über ManualEnable = TRUE aktivieren. Es wird empfohlen, die Betriebsarten nur über Mode und ModeActivate zu wechseln. Der Wechsel vom Handbetrieb in den Automatikbetrieb erfolgt stoßfrei. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.
5	Ersatzausgangswert mit Fehlerüberwachung Der Regelalgorithmus ist ausgeschaltet. Die Variable SetSubstituteOutput bestimmt, welcher Ausgangswert während dieser Betriebsart ausgegeben wird. • SetSubstituteOutput = FALSE: letzter gültiger Ausgangswert • SetSubstituteOutput = TRUE: Ersatzausgangswert Diese Betriebsart können Sie nicht mit Mode = 5 aktivieren. Sie wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert, wenn alle folgenden Bedingungen erfüllt sind: • Automatikbetrieb (Mode = 3) • ActivateRecoverMode = TRUE • Ein oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_Compact wieder in den Automatikbetrieb.

ENO-Verhalten

Wenn State = 0 ist, dann ist ENO = FALSE.

Wenn State ≠ 0 ist, dann ist ENO = TRUE.

Automatische Betriebsartwechsel während der Inbetriebnahme

Nach erfolgreicher Erst- oder Nachoptimierung wird der Automatikbetrieb aktiviert. Die folgende Tabelle zeigt, wie sich Mode und State während einer erfolgreichen Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet
n	4	1	Erstoptimierung erfolgreich beendet
n	3	3	Automatikbetrieb wird gestartet

Im Fehlerfall wechselt PID_Compact automatisch die Betriebsart. Die folgende Tabelle zeigt, wie sich Mode und State während einer fehlerhaften Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet

Zyklus-Nr.	Mode	State	Aktion
n	4	1	Erstoptimierung abgebrochen
n	4	4	Handbetrieb wird gestartet

Wenn ActivateRecoverMode = TRUE ist, wird die Betriebsart aktiviert, die an Mode gespeichert ist. Beim Start der Erst- oder Nachoptimierung hat PID_Compact den Wert von State am Durchgangsparameter Mode gespeichert. PID_Compact wechselt also in die Betriebsart, aus der die Optimierung gestartet wurde.

Wenn ActivateRecoverMode = FALSE ist, wird in die Betriebsart "Inaktiv" gewechselt.

Siehe auch

Ausgangsparameter PID_Compact V2 (Seite 2911)

Parameter ErrorBits V2

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der ErrorBits binär addiert angezeigt. Wird z. B. ErrorBits = 0003h angezeigt, so stehen gleichzeitig die Fehler 0001h und 0002h an.

PID_Compact verwendet im Handbetrieb ManualValue als Ausgangswert. Ausnahme ist Errorbits = 10000h.

ErrorBits (DW#16#...)	Beschreibung
0000	Es steht kein Fehler an.
0001	Der Parameter "Input" ist außerhalb der Istwertgrenzen. • Input > Config.InputUpperLimit oder • Input < Config.InputLowerLimit Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Compact im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.
0002	Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.
0004	Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0008	Fehler beim Starten der Erstoptimierung. Der Istwert ist zu nahe am Sollwert. Starten Sie die Nachoptimierung. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.

ErrorBits (DW#16#...)	Beschreibung
0010	Der Sollwert wurde während der Optimierung verändert. An der Variable CancelTuningLevel können Sie die zulässige Schwankung des Sollwerts einstellen. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0020	Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_Compact in der Betriebsart Nachoptimierung.
0080	Fehler bei der Erstoptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert oder der Istwert reagiert nicht wie erwartet. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen. Stellen Sie außerdem sicher, dass der Istwert vor Start einer Erstoptimierung nicht stark schwingt. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0100	Fehler während der Nachoptimierung führte zu ungültigen Parametern. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0200	Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.
0400	Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.
0800	Abtastzeitfehler: PID_Compact wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Compact im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist. Falls dieser Fehler bei der Simulation mit PLCSIM aufgetreten ist, beachten Sie die Hinweise unter AUTOHOTSPOT.
1000	Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv und ActivateRecoverMode = TRUE waren, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.
10000	Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, verwendet PID_Compact SubstituteOutput als Ausgangswert. Sobald Sie einen gültigen Wert an ManualValue vorgeben, verwendet PID_Compact diesen als Ausgangswert.

ErrorBits (DW#16#...)	Beschreibung
20000	Ungültiger Wert an der Variablen SubstituteOutput. Wert hat kein gültiges Zahlenformat. PID_Compact verwendet die Untergrenze Ausgangswert als Ausgangswert. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.
40000	Ungültiger Wert am Parameter Disturbance. Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, wird Disturbance auf Null gesetzt. PID_Compact bleibt im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erst- oder Nachoptimierung aktiv und ActivateRecover-Mode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist. Wenn Disturbance in der aktuellen Phase keinen Einfluss auf den Ausgangswert hat, wird die Optimierung nicht abgebrochen.

Variable ActivateRecoverMode V2

Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall. Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

Automatikbetrieb

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn ActivateRecoverMode = TRUE ist, bleibt PID_Compact im Fehlerfall auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_Compact in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.
TRUE	Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert, da PID_Compact bei jedem Fehlerfall zwischen berechnetem Ausgangswert und dem Ersatzausgangswert wechselt. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_Compact im Automatikbetrieb: • 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen. • 0800h: Abtastzeitfehler • 40000h: Ungültiger Wert am Parameter Disturbance. Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_Compact in die Betriebsart " Ersatzausgangswert mit Fehlerüberwachung": • 0002h: Ungültiger Wert am Parameter Input_PER. • 0200h: Ungültiger Wert am Parameter Input. • 0400h: Berechnung des Ausgangswerts fehlgeschlagen. • 1000h: Ungültiger Wert am Parameter Setpoint. Wenn der folgende Fehler auftritt, wechselt PID_Compact in die Betriebsart " Ersatzausgangswert mit Fehlerüberwachung" und fährt das Stellglied auf Config.OutputLowerLimit: • 20000h: Ungültiger Wert an der Variable SubstituteOutput. Wert hat kein gültiges Zahlenformat. Dieses Verhalten ist unabhängig von SetSubstituteOutput. Sobald die Fehler nicht mehr anstehen, wechselt PID_Compact wieder in den Automatikbetrieb.

Erstoptimierung und Nachoptimierung

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_Compact in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.
TRUE	Wenn der folgende Fehler auftritt, bleibt PID_Compact in der aktiven Betriebsart: • 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Die folgenden Fehler werden ignoriert: • 10000h: Ungültiger Wert am Parameter ManualValue. • 20000h: Ungültiger Wert an der Variable SubstituteOutput. Bei allen anderen Fehlern bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.

Handbetrieb

Im Handbetrieb wirkt ActivateRecoverMode nicht.

Variable Warning V2

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Variable Warning binär addiert angezeigt. Wenn z. B. die Warnung 0003h angezeigt, so stehen gleichzeitig die Warnungen 0001h und 0002h an.

Warning (DW#16#....)	Beschreibung
0000	Es liegt keine Warnung vor.
0001	Während der Erstoptimierung wurde der Wendepunkt nicht gefunden.
0004	Der Sollwert wurde begrenzt auf die eingestellten Grenzen.
0008	Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TIR.TuneRule = 3 berechnet.
0010	Die Betriebsart konnte nicht geändert werden, da Reset = TRUE oder ManualEnable = TRUE.
0020	Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.
0040	Der Istwert hat eine seiner Warngrenzen überschritten.
0080	Ungültiger Wert an Mode. Die Betriebsart wird nicht gewechselt.
0100	Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.
0200	Die angegebene Regel zur Optimierung wird nicht unterstützt. Es werden keine PID-Parameter berechnet.
1000	Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h

Alle anderen Warnungen werden bei steigender Flanke an Reset oder ErrorAck gelöscht.

Variable IntegralResetMode V2

Die Variable IntegralResetMode bestimmt, wie der I-Anteil PIDCtrl.IntegralSum vorbelegt wird:

- beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb"
- bei Flanke TRUE -> FALSE an Parameter Reset und Parameter Mode = 3

Diese Einstellung wirkt nur für einen Zyklus und ist nur wirksam, falls der I-Anteil aktiviert ist (Variable Retain.CtrlParams.Ti > 0.0)

IntegralReset-Mode	Beschreibung
0	Glätten Der Wert von PIDCtrl.IntegralSum wird so vorbelegt, dass die Umschaltung stoßfrei erfolgt d. h. der "Automatikbetrieb" startet ausgehend von Ausgangswert = 0.0 (Parameter Output) und es erfolgt kein Sprung des Ausgangswerts unabhängig von der Regeldifferenz (Sollwert – Istwert).
1	Löschen Es wird empfohlen die Gewichtung des P-Anteils (Retain.CtrlParams.PWeighting) auf 1.0 zu setzen, falls diese Option verwendet wird. Der Wert von PIDCtrl.IntegralSum wird gelöscht. Wenn eine Regeldifferenz vorhanden ist, kommt es zu einem Sprung des Ausgangswerts. Die Richtung des Ausgangswertsprungs hängt von der konfigurierten Gewichtung des P-Anteils (Variable Retain.CtrlParams.PWeighting) und der Regeldifferenz ab: • Gewichtung des P-Anteils = 1.0: Ausgangswertsprung und Regeldifferenz haben identische Vorzeichen. Beispiel: Liegt der Istwert unterhalb des Sollwerts (positive Regeldifferenz), springt der Ausgangswert auf einen positiven Wert. • Gewichtung des P-Anteils < 1.0: Für große Regeldifferenzen haben Ausgangswertsprung und Regeldifferenz identische Vorzeichen. Beispiel: Liegt der Istwert weit unterhalb des Sollwerts (positive Regeldifferenz), springt der Ausgangswert auf einen positiven Wert. Für kleine Regeldifferenzen haben Ausgangswertsprung und Regeldifferenz unterschiedliche Vorzeichen. Beispiel: Liegt der Istwert knapp unterhalb des Sollwerts (positive Regeldifferenz), springt der Ausgangswert auf einen negativen Wert. Dies ist im Normalfall nicht erwünscht, da es zu einer vorübergehenden Vergrößerung der Regeldifferenz führt. Je kleiner dabei die Gewichtung des P-Anteils konfiguriert ist, desto größer muss die Regeldifferenz sein; um einen Ausgangswertsprung mit identischem Vorzeichen zu erhalten. Es wird empfohlen die Gewichtung des P-Anteils (Retain.CtrlParams.PWeighting) auf 1.0 zu setzen, wenn diese Option verwendet wird. Andernfalls kann dies zum beschriebenen unerwünschten Verhalten bei kleinen Regeldifferenzen führen. Alternativ können Sie auch IntegralResetMode = 4 verwenden. Diese Option garantiert identische Vorzeichen von Ausgangswertsprung und Regeldifferenz unabhängig von der konfigurierten Gewichtung des P-Anteils und der Regeldifferenz.
2	Halten Der Wert von PIDCtrl.IntegralSum wird nicht verändert. Sie können über das Anwenderprogramm einen neuen Wert vorgeben.
3	Vorbelegen Der Wert von PIDCtrl.IntegralSum wird automatisch so vorbelegt, als ob im letzten Zyklus Output = OverwriteInitialOutputValue gewesen wäre.
4	Wie Sollwertänderung (nur für PID_Compact mit Version ≥ 2.3) Der Wert von PIDCtrl.IntegralSum wird automatisch so vorbelegt, dass sich ein ähnlicher Ausgangswertsprung ergibt, wie für einen PI Regler im Automatikbetrieb bei einer Sollwertänderung vom aktuellen Istwert zum aktuellen Sollwert. Falls eine Regeldifferenz vorhanden ist, kommt es zu einem Sprung des Ausgangswerts. Ausgangswertsprung und Regeldifferenz haben identische Vorzeichen. Beispiel: Liegt der Istwert unterhalb des Sollwerts (positive Regeldifferenz), springt der Ausgangswert auf einen positiven Wert. Dies ist unabhängig von der konfigurierten Gewichtung des P-Anteils und der Regeldifferenz.

Wird IntegralResetMode mit einem Wert außerhalb des gültigen Wertebereichs belegt, verhält sich PID_Compact wie mit der Vorbelegung von IntegralResetMode:

- PID_Compact bis V2.2: IntegralResetMode = 1
- PID_Compact ab V2.3: IntegralResetMode = 4

Alle obigen Aussagen zum Vorzeichen des Ausgangswertsprungs gehen von einem normalen Regelsinn (Variable Config.InvertControl = FALSE) aus. Bei invertiertem Regelsinn (Config.InvertControl = TRUE) ergibt sich ein umgekehrtes Vorzeichen des Ausgangswertsprungs.

Beispielprogramm für PID_Compact

In dem folgenden Beispiel regeln Sie Temperaturwerte mit dem Technologieobjekt der Anweisung "PID_Compact". Die Temperaturwerte werden anhand eines Bausteins simuliert, der ein Verzögerungsglied dritter Ordnung (PT3-Glied) nachbildet. Die PID-Parameter des Technologieobjekts lassen Sie automatisch über die Erstoptimierung einstellen.

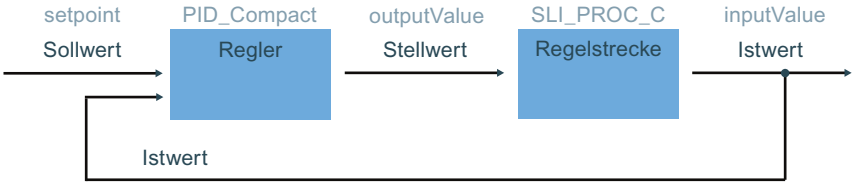

Ablage der Daten

Zur Ablage der Daten für die Verschaltung legen Sie in einem globalen Datenbaustein sieben Variablen an.

SLI_gDB_PID_Compact

		Name	Data type	Start value
1		Static		
2		setpoint	Real	75.0
3		inputValue	Real	0.0
4		outputValue	Real	0.0
5		state	Int	0
6		error	Bool	false
7		errorBits	DWord	16#0
8		resetAll	Bool	false

Verschaltung der Parameter

Die folgenden Verschaltungen rufen Sie in einem Weckalarm-OB auf.

Netzwerk 1: Die Parameter der Anweisung "PID_Compact" verschalten Sie wie folgt.

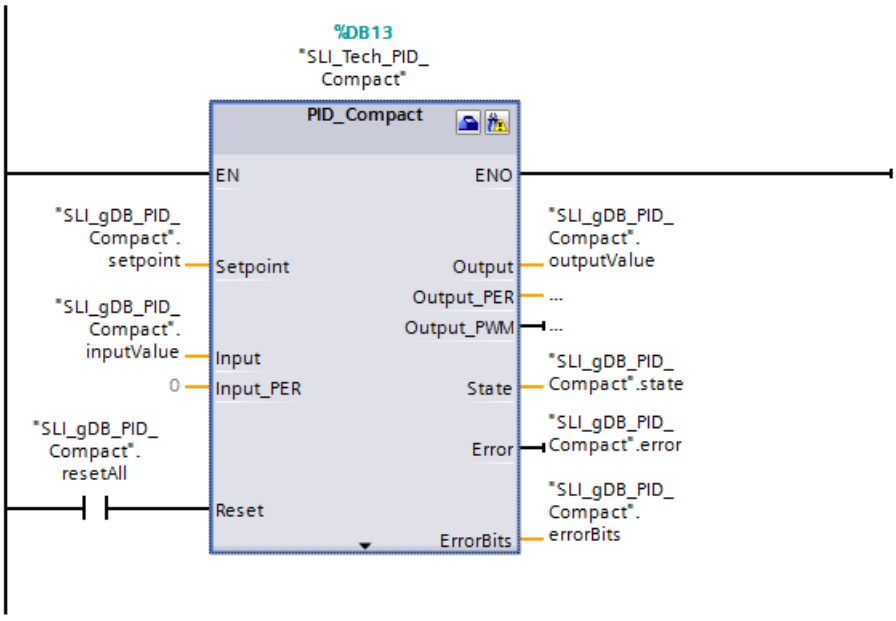

Netzwerk 2: Die Parameter des Temperaturwerte simulierenden Bausteins "SLI_PROC_C" verschalten Sie wie folgt.

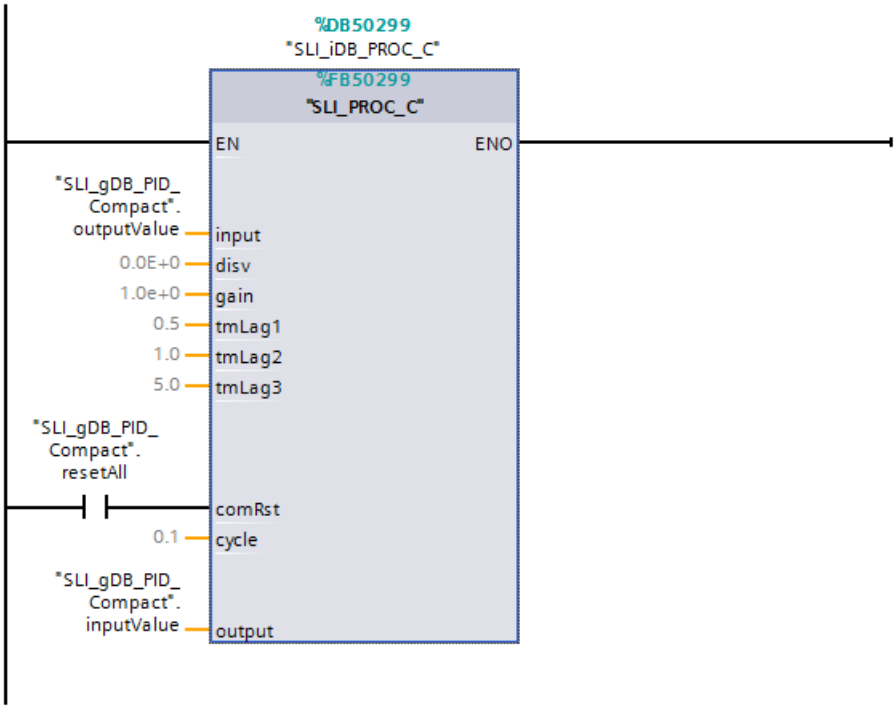

Technologieobjekt

Das Technologieobjekt lässt sich über die Eigenschaften der Anweisung "PID_Compact" oder über den Pfad Technologieobjekt > Konfiguration konfigurieren. Für das Beispiel sind die Regelungsart und die Ein- und Ausgangsparameter von Bedeutung. Über die Regelungsart wird eine Vorauswahl für die Einheit des zu regelnden Werts festgelegt. In diesem Beispiel wird als Regelungsart "Temperatur" mit der Einheit "°C" verwendet. Die Parameter des "PID_Compact" sind bereits mit globalen Variablen verschalten. Dementsprechend genügt die Angabe zur Verwendung der Parameter Input und Output.

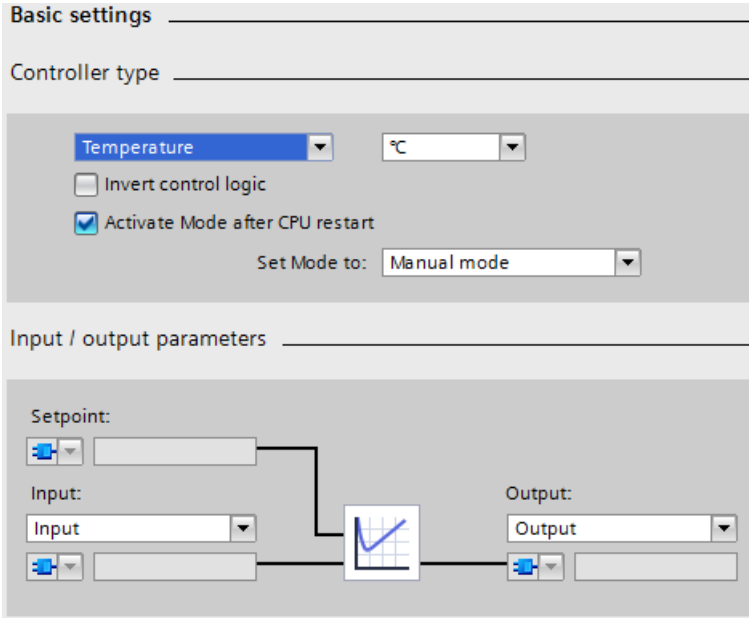

Vorgehen zum Starten der Regelung

Nach dem Laden in die CPU befindet sich PID_Compact im Handbetrieb mit Handwert 0.0. Um die Regelung zu beginnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Inbetriebnahme des Technologieobjekts "SLI_Tech_PID_Compact".
2. Klicken Sie im Bereich "Messung" auf die Schaltfläche "Start".

   Die Messung startet und PID_Compact kann aktiviert werden.

3. Die Erstoptimierung ist ausgewählt.
   Klicken Sie im Bereich "Optimierungsart" auf die Schaltfläche "Start".
   Eine Erstoptimierung wird durchgeführt. Die PID-Parameter werden dabei automatisch auf die Regelstrecke angepasst. Nach Abschluss der Erstoptimierung wechselt PID_Compact in den Automatikbetrieb.

Hinweis

Alternative zum Starten von PID_Compact

Alternativ können Sie PID_Compact im Bereich "Online-Zustand des Reglers" über die Schaltfläche "Stop PID_Compact" / "Start PID_Compact" ohne Erstoptimierung in den Automatikbetrieb versetzen. Dadurch verwendet der Regler standardmäßige Werte für die PID-Parameter und zeigt ein schlechteres Regelverhalten für den Anwendungsfall.

Vorgehen zum Stoppen der Regelung

Um PID_Compact und das Programm zu stoppen und zu beenden gehen Sie folgendermaßen vor:

1. Klicken Sie im Technologieobjekt "SLI_Tech_PID_Compact", im Bereich "Online-Zustand des Reglers" auf die Schaltfläche "Stop PID_Compact".

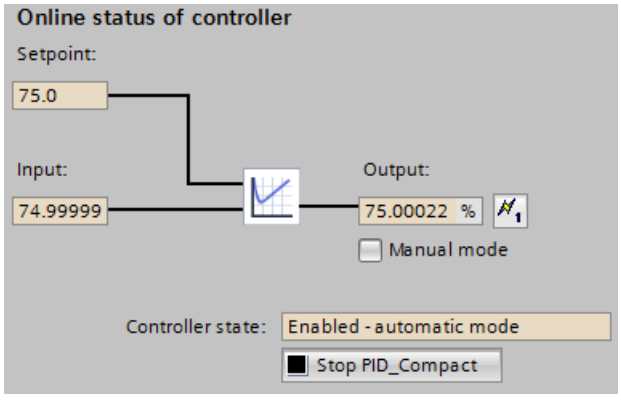

   Die Anweisung "PID_Compact" beendet die Regelung und gibt als Stellwert den Wert "0.0" aus.

2. Klicken Sie im Bereich "Messung" auf die Schaltfläche "Stop".
3. Um den Istwert sofort auf den Wert "0.0" zu setzen, gehen Sie folgendermaßen vor: Setzen Sie im Baustein "SLI_OB_PID_Compact" die Variable "resetAll" auf den Wert "TRUE", und anschließend auf den Wert "FALSE".

Anweisung "PID_Compact"

Am Parameter Setpoint ("setpoint") ist der Sollwert für die zu regelnde Temperatur vorgegeben. Wenn die Anweisung "PID_Compact" über das Technologieobjekt gestartet wurde, wird die Regelung gestartet. Die Anweisung "PID_Compact" gibt am Ausgangsparameter Output ("outputValue") einen Stellwert aus. Über den Eingangsparameter Input ("inputValue") wird der Anweisung "PID_Compact" der Istwert der Temperatur übermittelt.

Abhängig vom Verlauf der Differenz zwischen Sollwert ("setpoint") und Istwert ("inputValue") passt die Anweisung "PID_Compact" den Stellwert ("outputValue") an. Dieser Vorgang wird wiederholt, so dass sich der Istwert ("inputValue") durch den Stellwert ("outputValue") dem Sollwert ("setpoint") annähert.

Am Ausgangsparameter State ("state") wird die aktuelle Betriebsart der Anweisung "PID_Compact" angezeigt. Nach einer Erstoptimierung (der Wert von "state" ist "1") schaltet der PID_Compact in den Automatikbetrieb (der Wert ist "3").

Der Ausgangsparameter Error ("error") zeigt aktuell an, dass kein Fehler ansteht. Der Ausgangsparameter ErrorBits ("errorBits") stellt im Fehlerfall Informationen zur Fehlerart bereit. Wenn ein Fehler auftritt, kann dieser im Technologieobjekt, im Bereich Optimierungsstatus über die Schaltfläche "ErrorAck" quittiert werden.

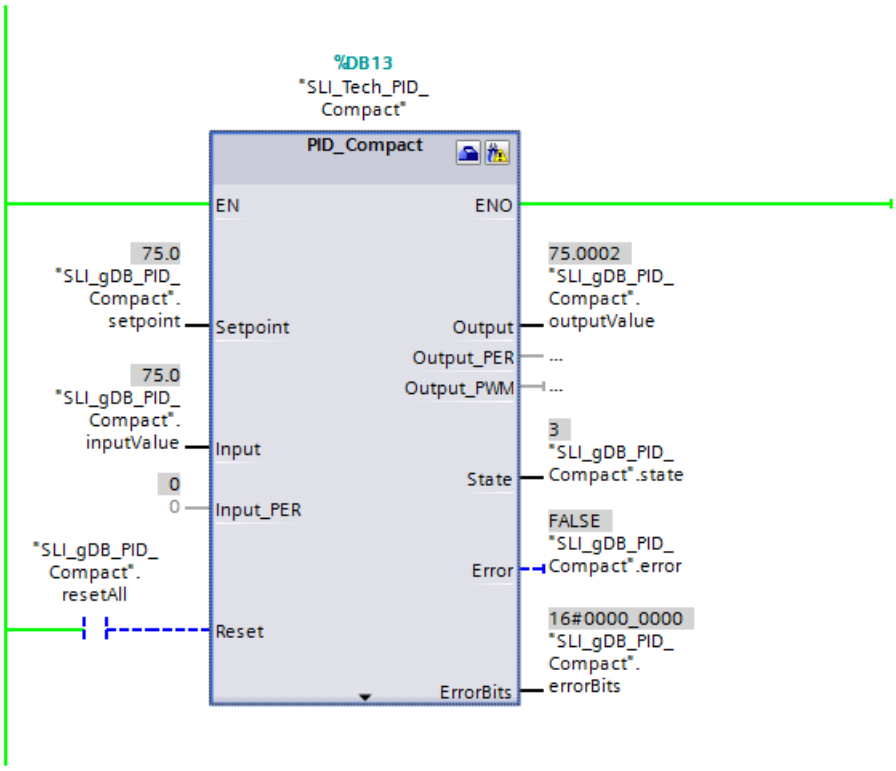

Baustein "SLI_PROC_C"

Der Baustein "SLI_PROC_C" simuliert den Istwert ("inputValue") der steigenden Temperatur einer Anlage. Der Baustein "SLI_PROC_C" erhält den Stellwert des Reglers ("outputValue) und simuliert das Temperaturverhalten der Regelstrecke. Diese Temperatur wird als Istwert ("inputValue") in den Regler eingespeist.

Eine Änderung der Werte der Variable "resetAll" (des Parameters comRst) hat folgende Auswirkungen:

Parameter comRst ("resetAll")	Die Anweisung "PID_Compact" läuft	Die Anweisung "PID_Compact" wurde gestoppt
comRst ("resetAll") bleibt auf dem Wert "FALSE"	Der Baustein "SLI_PROC_C" gibt auf Basis eines Stellwerts ("outputValue") einen neuen Istwert ("inputValue") aus.	Der Baustein "SLI_PROC_C" erhält zwar keinen Stellwert > "0.0" übermittelt, gibt aber weiterhin einen Istwert > "0.0" aus.
comRst ("resetAll"): Änderung von "FALSE" auf den Wert "TRUE"	Sowohl Stellwert ("outputValue") als auch ausgegebener Istwert ("inputValue") werden auf "0.0" zurückgesetzt.	Der ausgegebene Istwert ("inputValue") / die Temperatur des Bausteins "SLI_PROC_C" wird auf "0.0" zurückgesetzt.
comRst ("resetAll"): Änderung von "TRUE" auf den Wert "FALSE"	Die Regelung der Temperatur beginnt von Neuem.	Der ausgegebene Istwert / die Temperatur ("inputValue") bleibt "0.0".

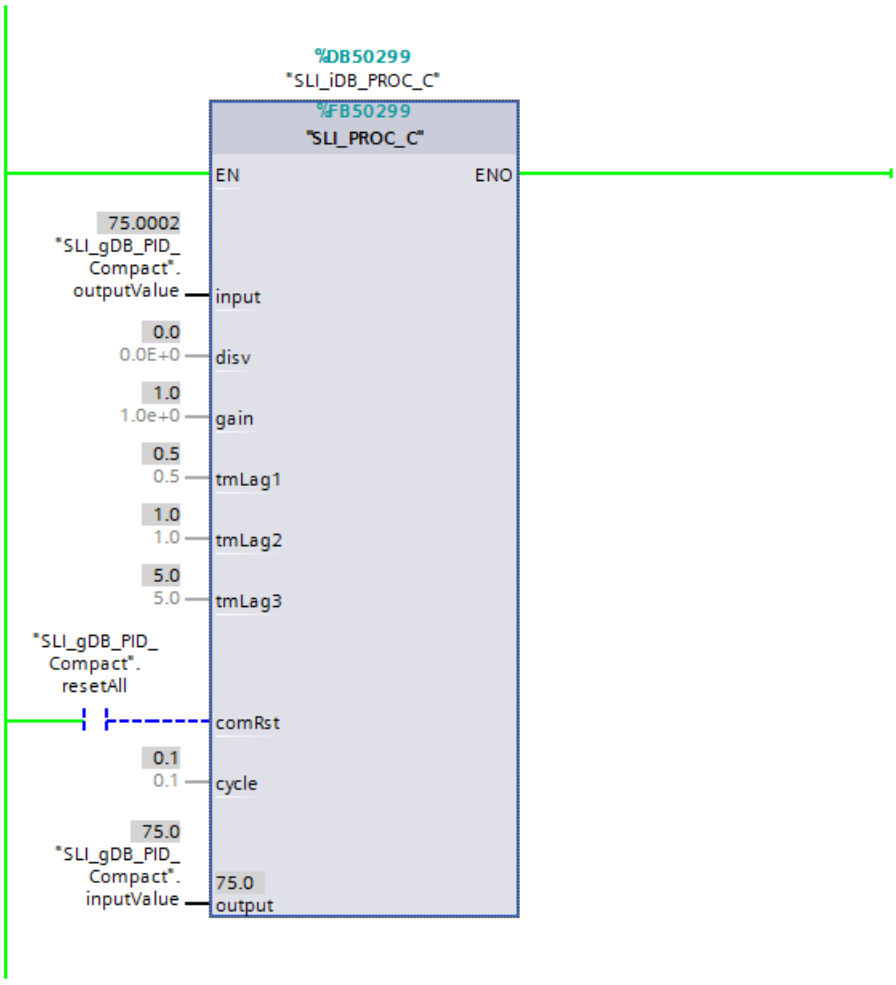

Programmcode

Weitere Informationen zum Programmcode zu oben genanntem Beispiel finden Sie unter dem Stichwort "Sample Library for Instructions".

PID_Compact V1

Beschreibung PID_Compact V1

Beschreibung

Die Anweisung PID_Compact stellt einen PID-Regler mit integrierter Optimierung für Automatik- und Handbetrieb zur Verfügung.

Aufruf

PID_Compact wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID_Compact vollständig laden.

AUTOHOTSPOT

Anlauf

Bei Anlauf der CPU startet PID_Compact in der zuletzt aktiven Betriebsart. Um PID_ Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie sb_RunModeByStartup = FALSE.

Überwachung der Abtastzeit PID_Compact

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID_Compact misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (Error = 0800 hex) und PID_Compact wechselt in die Betriebsart "Inaktiv"

Folgende Bedingungen versetzen PID_Compact ab Version 1.1 während der Optimierung in die Betriebsart "Inaktiv":

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert

Folgende Bedingungen versetzen PID_Compact ab Version 1.1 bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert >= 1,5 x alter Mittelwert
- Neuer Mittelwert <= 0,5 x alter Mittelwert

Folgende Bedingungen versetzen PID_Compact 1.0 während der Optimierung und bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert
- Aktuelle Abtastzeit >= 1,5 x aktueller Mittelwert
- Aktuelle Abtastzeit <= 0,5 x aktueller Mittelwert

Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_Compact werden bei jedem Aufruf durchgeführt.

PID-Algorithmus

PID_Compact ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der Ausgangswert wird anhand folgender Formel berechnet.

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Symbol	Beschreibung
y	Ausgangswert
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient für den Differenzierverzug (T1 = a × T_D)
	Differenzierzeit
c	Gewichtung des D-Anteils

Blockschaltbild PID_Compact

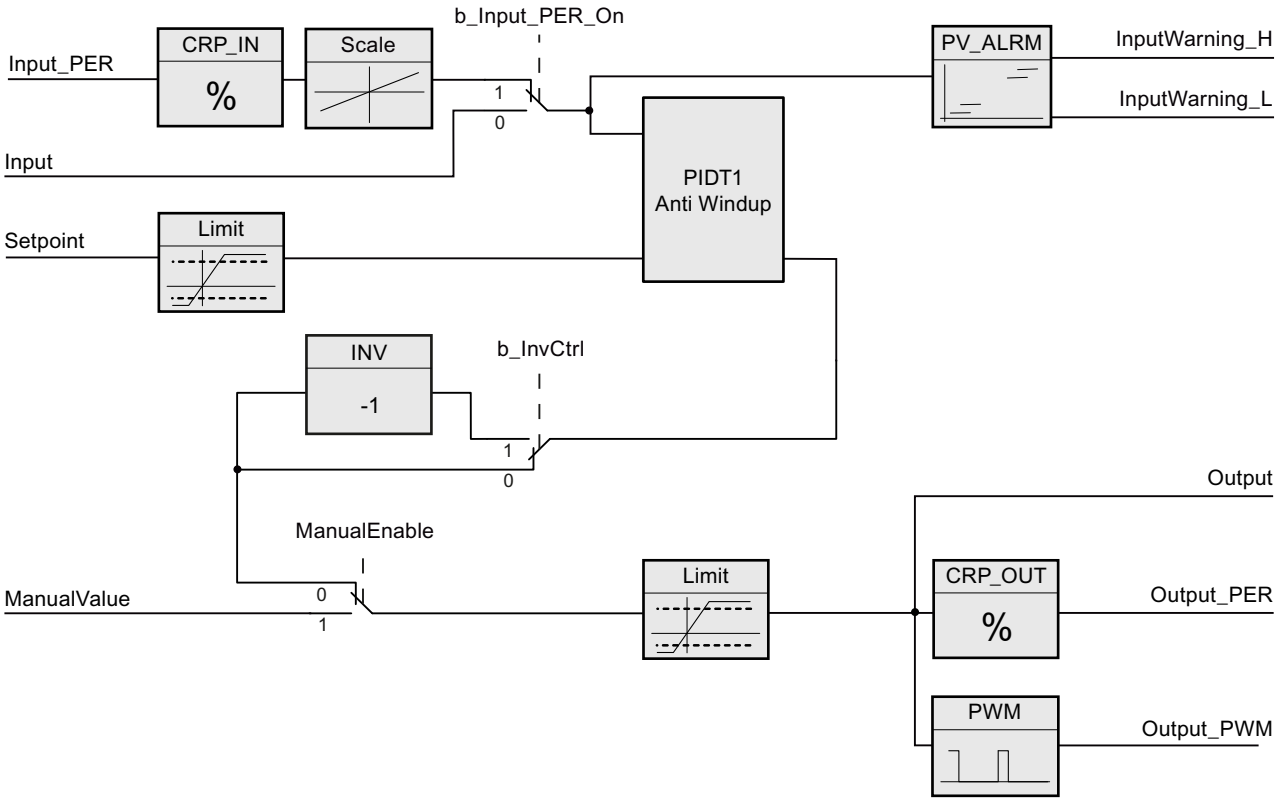

Blockschaltbild PIDT1 mit Anti-Windup

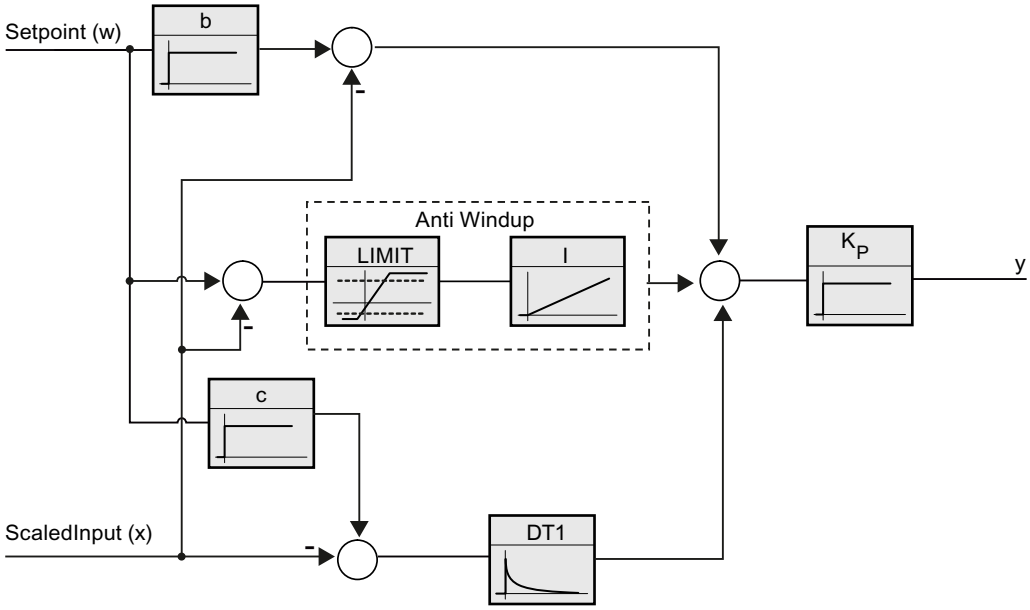

Verhalten im Fehlerfall

Wenn Fehler auftreten, werden diese am Parameter Error ausgegeben und PID_Compact wechselt in die Betriebsart "Inaktiv". Mit dem Parameter Reset setzen Sie die Fehler wieder zurück.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID_Compact arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

Eingangsparameter PID_Compact V1

Tabelle 4-61

Parameter	Datentyp	Vorbelegung	Beschreibung
Setpoint	REAL	0.0	Sollwert des PID-Reglers im Automatikbetrieb
Input	REAL	0.0	Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input verwenden, muss sPid_Cmpt.b_Input_PER_On = FALSE sein.

Parameter	Datentyp	Vorbelegung	Beschreibung
Input_PER	WORD	W#16#0	Analogeingang als Quelle des Istwerts Wenn Sie den Parameter Input_PER verwenden, muss sPid_Cmpt.b_Input_PER_On = TRUE sein.
ManualEnable	BOOL	FALSE	• Flanke FALSE -> TRUE wählt die Betriebsart "Handbetrieb", State = 4, sRet.i_Mode bleibt unverändert. • Flanke TRUE -> FALSE wählt die zuletzt aktive Betriebsart , State =sRet.i_Mode Während ManualEnable = TRUE wirkt sich eine Änderung von sRet.i_Mode nicht aus. Erst bei der Flanke TRUE -> FALSE an ManualEnable wird die Änderung von sRet.i_Mode berücksichtigt. **PID_Compact V1.2 und PID_Compact V1.0** Wenn beim Start der CPU ManualEnable = TRUE, startet PID_Compact im Handbetrieb. Eine steigende Flanke (FALSE > TRUE) an ManualEnable ist nicht notwendig. **PID_Compact V1.1** Beim Start der CPU schaltet PID_Compact nur bei einer steigenden Flanke (FALSE->TRUE) an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID_Compact in der letzten Betriebsart bei der ManualEnable FALSE war.
ManualValue	REAL	0.0	Handwert Dieser Wert wird im Handbetrieb als Ausgangswert verwendet.
Reset	BOOL	FALSE	Der Parameter Reset (Seite 2953) führt einen Neustart des Reglers durch.

Ausgangsparameter PID_Compact V1

Tabelle 4-62

Parameter	Datentyp	Vorbelegung	Beschreibung
ScaledInput	REAL	0.0	Ausgabe des skalierten Istwerts
Die Ausgänge "Output", "Output_PER" und "Output_PWM" können parallel genutzt werden.			
Output	REAL	0.0	Ausgangswert im REAL-Format
Output_PER	WORD	W#16#0	Analoger Ausgangswert
Output_PWM	BOOL	FALSE	Pulsweitenmodulierter Ausgangswert Der Ausgangswert wird über variable Ein- und Ausschaltzeiten gebildet.
SetpointLimit_H	BOOL	FALSE	Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Obergrenze des Sollwerts begrenzt. Als Obergrenze Sollwert ist die konfigurierte absolute Obergrenze Istwert vorbelegt. Wenn Sie sPid_Cmpt.r_Sp_Hlm auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.

Parameter	Datentyp	Vorbelegung	Beschreibung
SetpointLimit_L	BOOL	FALSE	Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Untergrenze des Sollwerts begrenzt. Als Untergrenze Sollwert ist die konfigurierte absolute Untergrenze Istwert vorbelegt. Wenn Sie sPid_Cmpt.r_Sp_Llm auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
InputWarning_H	BOOL	FALSE	Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.
InputWarning_L	BOOL	FALSE	Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.
State	INT	0	Der Parameter State (Seite 2948) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit der Variable sRet.i_Mode. • State = 0: Inaktiv • State = 1: Erstoptimierung • State = 2: Nachoptimierung • State = 3: Automatikbetrieb • State = 4: Handbetrieb
Error	DWORD	W#16#0	Der Parameter Error (Seite 2952) zeigt die Fehlermeldungen. Error = 0000: Es steht kein Fehler an.

Statische Variablen PID_Compact V1

Hinweis

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Verändern Sie die mit (1) gekennzeichneten Variablen nur in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden. Die Betriebsart "Inaktiv" erzwingen Sie durch den Wert "0" der Variable "sRet.i_Mode".

Tabelle 4-63

Variable	Datentyp	Vorbelegung	Beschreibung
sb_GetCycleTime	BOOL	TRUE	Wenn sb_GetCycleTime = TRUE, wird die automatische Ermittlung der Zykluszeit gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.
sb_EnCyclEstimation	BOOL	TRUE	Wenn sb_EnCyclEstimation = TRUE, wird die Abtastzeit PID_Compact berechnet.
sb_EnCyclMonitoring	BOOL	TRUE	Wenn sb_EnCyclMonitoring = FALSE, wird die Abtastzeit PID_Compact nicht überwacht. Wenn PID_Compact nicht innerhalb der Abtastzeit ausgeführt werden kann, wird kein Fehler 0800 ausgegeben und PID_Compact wechselt nicht in die Betriebsart "Inaktiv".

Variable	Datentyp	Vorbelegung	Beschreibung
sb_RunModeByStartup	BOOL	TRUE	Nach CPU Neustart letzte Betriebsart aktivieren Wenn sb_RunModeByStartup = FALSE, bleibt der Regler nach CPU-Anlauf inaktiv. Wenn sb_RunModeByStartup = TRUE, kehrt der Regler nach CPU-Anlauf in die zuletzt aktive Betriebsart zurück.
si_Unit	INT	0	Physikalische Einheit des Ist- und Sollwerts, z. B. ºC oder ºF.
si_Type	INT	0	Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur
sd_Warning	DWORD	DW#16#0	Die Variable sd_warning (Seite 2955) zeigt die Warnungen seit Reset oder dem letzen Wechsel der Betriebsart.
sBackUp.r_Gain	REAL	1.0	gespeicherte Proportionalverstärkung Werte aus der Struktur sBackUp können mit sPid_Cmpt.b_LoadBackUp = TRUE wieder geladen werden.
sBackUp.r_Ti	REAL	20.0	gespeicherte Integrationszeit [s]
sBackUp.r_Td	REAL	0.0	gespeicherte Differenzierzeit [s]
sBackUp.r_A	REAL	0.0	gespeicherter Koeffizient Differenzierverzug
sBackUp.r_B	REAL	0.0	gespeicherter Gewichtungsfaktor des P-Anteils
sBackUp.r_C	REAL	0.0	gespeicherter Gewichtungsfaktor des D-Anteils
sBackUp.r_Cycle	REAL	1.0	gespeicherte Abtastzeit PID-Algorithmus
sPid_Calc.r_Cycle(1)	REAL	0.1	Abtastzeit der Anweisung PID_Compact r_Cycle wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.

Variable	Datentyp	Vorbelegung	Beschreibung
sPid_Calc.b_RunIn	BOOL	FALSE	• b_RunIn = FALSE Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstoptimierung gestartet. Wenn die Voraussetzungen für Erstoptimierung nicht erfüllt sind, verhält sich PID_Compact wie bei b_RunIn = TRUE. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_Compact in die Betriebsart "Inaktiv". • b_RunIn = TRUE Die Erstoptimierung wird übersprungen. PID_3Compact versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet anschließend automatisch. b_RunIn wird nach der Nachoptimierung auf FALSE gesetzt.
sPid_Calc.b_CalcParamSUT	BOOL	FALSE	Wenn b_CalcParamSUT = TRUE, werden die Parameter für die Erstoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen. b_CalcParamSUT wird nach der Berechnung auf FALSE gesetzt.
sPid_Calc.b_CalcParamTIR	BOOL	FALSE	Wenn b_CalcParamTIR = TRUE, werden die Parameter für die Nachoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen. b_CalcParamTIR wird nach der Berechnung auf FALSE gesetzt.
sPid_Calc.i_CtrlTypeSUT	INT	0	Parameter während Erstoptimierung berechnen nach Methode: • i_CtrlTypeSUT = 0: PID nach Chien, Hrones und Reswick • i_CtrlTypeSUT = 1: PI nach Chien, Hrones und Reswick

Variable	Datentyp	Vorbelegung	Beschreibung
sPid_Calc.i_CtrlTypeTIR	INT	0	Parameter während Nachoptimierung berechnen nach Methode: • i_CtrlTypeTIR = 0: PID automatisch • i_CtrlTypeTIR = 1: PID schnell • i_CtrlTypeTIR = 2: PID langsam • i_CtrlTypeTIR = 3: Ziegler-Nichols PID • i_CtrlTypeTIR = 4: Ziegler-Nichols PI • i_CtrlTypeTIR = 5: Ziegler-Nichols P
sPid_Calc.r_Progress	REAL	0.0	Fortschritt der Optimierung in Prozent (0.0 - 100.0)
sPid_Cmpt.r_Sp_Hlm(1)	REAL	+3.402822e +38	Obergrenze des Sollwerts Wenn Sie sPid_Cmpt.r_Sp_Hlm außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert verwendet. Wenn Sie sPid_Cmpt.r_Sp_Hlm innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.
sPid_Cmpt.r_Sp_Llm(1)	REAL	-3.402822e +38	Untergrenze des Sollwerts Wenn Sie sPid_Cmpt.r_Sp_Llm außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert verwendet. Wenn Sie sPid_Cmpt.r_Sp_Llm innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
sPid_Cmpt.r_Pv_Norm_IN_1(1)	REAL	0.0	Skalierung Input_PER Unten Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.
sPid_Cmpt.r_Pv_Norm_IN_2(1)	REAL	27648.0	Skalierung Input_PER Oben Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.
sPid_Cmpt.r_Pv_Norm_OUT_1(1)	REAL	0.0	Skalierter unterer Istwert Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.
sPid_Cmpt.r_Pv_Norm_OUT_2(1)	REAL	100.0	Skalierter oberer Istwert Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.
sPid_Cmpt.r_Lmn_Hlm(1)	REAL	100.0	Obere Ausgangswertgrenze für den Ausgangsparameter "Output"

Variable	Datentyp	Vorbelegung	Beschreibung
sPid_Cmpt.r_Lmn_Llm(1)	REAL	0.0	Untere Ausgangswertgrenze für den Ausgangsparameter "Output"
sPid_Cmpt.b_Input_PER_On(1)	BOOL	TRUE	Wenn b_Input_PER_On = TRUE, wird der Parameter Input_PER verwendet. Wenn b_Input_PER_On = FALSE, wird der Parameter Input verwendet.
sPid_Cmpt.b_LoadBackUp	BOOL	FALSE	Aktivieren des Backup Parametersatzes. Sollte eine Optimierung fehlgeschlagen sein, können Sie durch Setzen dieses Bits die vorigen PID Parameter wieder reaktivieren.
sPid_Cmpt.b_InvCtrl(1)	BOOL	FALSE	Invertieren des Regelsinns Wenn b_InvCtrl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.
sPid_Cmpt.r_Lmn_Pwm_PPTm(1)	REAL	0.0	Minimale Einschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf r_Lmn_Pwm_PPTm = r_Cycle oder r_Lmn_Pwm_PPTm = n*r_Cycle
sPid_Cmpt.r_Lmn_Pwm_PBTm(1)	REAL	0.0	Minimale Ausschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf r_Lmn_Pwm_PBTm = r_Cycle oder r_Lmn_Pwm_PBTm = n*r_Cycle
sPid_Cmpt.r_Pv_Hlm(1)	REAL	120.0	Obergrenze des Istwerts Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird keine Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_Compact wechselt in die Betriebsart "Inaktiv". r_Pv_Hlm > r_Pv_Llm
sPid_Cmpt.r_Pv_Llm(1)	REAL	0.0	Untergrenze des Istwerts r_Pv_Llm < r_Pv_Hlm
sPid_Cmpt.r_Pv_HWrn(1)	REAL	+3.402822e +38	Obere Warngrenze des Istwerts Wenn Sie r_Pv_HWrn außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obere Warngrenze verwendet. Wenn Sie r_Pv_HWrn innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet. r_Pv_HWrn > r_Pv_LWrn r_Pv_HWrn ≤ r_Pv_Hlm
sPid_Cmpt.r_Pv_LWrn(1)	REAL	-3.402822e +38	Untere Warngrenze des Istwerts Wenn Sie r_Pv_LWrn außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untere Warngrenze verwendet. Wenn Sie r_Pv_LWrn innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet. r_Pv_LWrn < r_Pv_HWrn r_Pv_LWrn ≥ r_Pv_LWrn

Variable	Datentyp	Vorbelegung	Beschreibung
sPidCalc.i_Ctrl_IOutv(1)	REAL	0.0	Aktueller I-Anteil
sParamCalc.i_Event_SUT	INT	0	Die Variable i_Event_SUT (Seite 2956) zeigt die aktuelle Phase der "Erstoptimierung":
sParamCalc.i_Event_TIR	INT	0	Die Variable i_Event_TIR (Seite 2956) zeigt die aktuelle Phase der "Nachoptimierung":
sRet.i_Mode	INT	0	Die Betriebsart wird flankengesteuert gewechselt. Folgende Betriebsart wird aktiviert beim Wechsel zu • i_Mode = 0: Betriebsart "Inaktiv" (Stopp des Reglers) • i_Mode = 1: Betriebsart "Erstoptimierung" • i_Mode = 2: Betriebsart "Nachoptimierung" • i_Mode = 3: Betriebsart "Automatikbetrieb" • i_Mode = 4: Betriebsart "Handbetrieb" i_Mode ist remanent.
sRet.r_Ctrl_Gain(1)	REAL	1.0	Aktive Proportionalverstärkung Gain ist remanent.
sRet.r_Ctrl_Ti(1)	REAL	20.0	• r_Ctrl_Ti > 0.0: Aktive Integrationszeit • r_Ctrl_Ti = 0.0: I-Anteil ist ausgeschaltet r_Ctrl_Ti ist remanent.
sRet.r_Ctrl_Td(1)	REAL	0.0	• r_Ctrl_Td > 0.0: Aktive Differenzierzeit • r_Ctrl_Td = 0.0: D-Anteil ist ausgeschaltet r_Ctrl_Td ist remanent.
sRet.r_Ctrl_A(1)	REAL	0.0	Aktiver Koeffizient Differenzierverzug r_Ctrl_A ist remanent.
sRet.r_Ctrl_B(1)	REAL	0.0	Aktive Gewichtung des P-Anteils r_Ctrl_B ist remanent.
sRet.r_Ctrl_C(1)	REAL	0.0	Aktive Gewichtung des D-Anteils r_Ctrl_C ist remanent.
sRet.r_Ctrl_Cycle(1)	REAL	1.0	Aktive Abtastzeit des PID-Algorithmus r_Ctrl_Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von r_Cycle aufgerundet. r_Ctrl_Cycle wird als Periodendauer der Pulsweitenmodulation verwendet. r_Ctrl_Cycle ist remanent.

Parameter State und sRet.i_Mode V1

Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Um die Betriebsart zu wechseln, müssen Sie die Variable sRet.i_Mode ändern. Das gilt auch, wenn in sRet.i_Mode bereits der Wert für die neue Betriebsart steht. Setzen Sie dann z. B. erst sRet.i_Mode = 0 und anschließend sRet.i_Mode = 3. Wenn die aktuelle Betriebsart des Reglers diesen Wechsel erlaubt, wird State auf den Wert von sRet.i_Mode gesetzt.

Wenn PID_Compact automatisch die Betriebsart wechselt gilt: State != sRet.i_Mode.

Beispiele:

- Erfolgreiche Erstoptimierung
  State = 3 und sRet.i_Mode = 1
- Im Fehlerfall
  State = 0 und sRet.i_Mode bleibt auf bisherigem Wert, z. B. sRet.i_Mode = 3
- ManualEnalbe = TRUE
  State = 4 und sRet.i_Mode bleibt auf bisherigem Wert, z. B. sRet.i_Mode = 3

Hinweis

Sie wollen z. B. eine erfolgreiche Nachoptimierung wiederholen ohne den Automatikbetrieb mit i_Mode = 0 zu beenden.

Wenn Sie sRet.i_Mode für einen Takt auf einen ungültigen Wert setzen, z. B. 9999, hat das keinen Einfluß auf State. Im nächsten Takt setzen Sie Mode = 2. So können Sie eine Änderung an sRet.i_Mode erzeugen, ohne erst in die Betriebsart "Inaktiv" zu wechseln.

Bedeutung der Werte

State / sRet.i_Mode	Beschreibung der Betriebsart
0	Inaktiv Der Regler ist ausgeschaltet. Bevor eine Erstoptimierung durchgeführt wurde, befindet sich der Regler in der Betriebsart "Inaktiv". Im laufenden Betrieb wechselt der PID-Regler in die Betriebsart "Inaktiv", wenn ein Fehler eintritt oder wenn im Inbetriebnahmefenster das Symbol "Regler deaktivieren" geklickt wird.
1	Erstoptimierung Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet. Voraussetzungen für die Erstoptimierung: • Der Regler befindet sich in der Betriebsart Inaktiv oder Handbetrieb • ManualEnable = FALSE • Der Istwert darf nicht zu nah am Sollwert sein. |Setpoint - Input| > 0.3 * | sPid_Cmpt.r_Pv_Hlm - sPid_Cmpt.r_Pv_Llm| und |Setpoint - Input| > 0.5 * |Setpoint| • Der Sollwert darf während der Erstoptimierung nicht geändert werden. Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit sPid_Cmpt.b_LoadBackUp reaktiviert werden. Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstoptimierung in die Betriebsart "Inaktiv". Die Phase der Erstoptimierung wird mit Variable i_Event_SUT V1 (Seite 2956) angezeigt.

State / sRet.i_Mode	Beschreibung der Betriebsart
2	Nachoptimierung Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter optimiert. Die Unterschiede zwischen der Prozessantwort während der Erst- und Nachoptimierung werden analysiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit sPid_Cmpt.b_LoadBackUp reaktiviert werden. Voraussetzungen für Nachoptimierung: • Es werden keine Störungen erwartet. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen • Der Sollwert darf während der Nachoptimierung nicht geändert werden. • ManualEnable = FALSE • Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4) Die Nachoptimierung verläuft folgendermaßen beim Start aus: • Automatikbetrieb (State = 3) Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb. PID_Comact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. • Inaktiv (State = 0) oder Handbetrieb (State = 4) Wenn die Voraussetzungen für eine Erstoptimierung erfüllt sind, wird eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_Compact in die Betriebsart "Inaktiv". Wenn sich der Istwert für eine Erstoptimierung bereits zu nah am Sollwert befindet oder sPid_Calc.b_RunIn = TRUE, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Nach erfolgreicher "Nachoptimierung" wechselt der Regler in die Betriebsart "Automatikbetrieb"; nach erfolgloser "Nachoptimierung" in die Betriebsart "Inaktiv". Die Phase der "Nachoptimierung" wird mit Variable i_Event_TIR V1 (Seite 2956) angezeigt.

State / sRet.i_Mode	Beschreibung der Betriebsart
3	Automatikbetrieb Im Automatikbetrieb regelt PID_Compact die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt: • Erstoptimierung erfolgreich abgeschlossen • Nachoptimierung erfolgreich abgeschlossen • Ändern der Variable sRet.i_Mode auf den Wert 3. Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_Compact in der zuletzt aktiven Betriebsart. Um PID_Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie sb_RunModeByStart-up = FALSE.
4	Handbetrieb Im Handbetrieb geben Sie einen manuellen Ausgangswert am Parameter ManualValue vor. Diese Betriebsart wird aktiviert, wenn sRet.i_Mode = 4 oder bei steigender Flanke an ManualEnable. Wenn ManualEnable TRUE wird, ändert sich nur State. sRet.i_Mode bleibt auf dem bisherigen Wert. Bei einer fallenden Flanke an ManualEnable kehrt PID_Compact in die vorhergehende Betriebsart zurück. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

Siehe auch

Ausgangsparameter PID_Compact V1 (Seite 2942)

Variable i_Event_SUT V1 (Seite 2956)

Variable i_Event_TIR V1 (Seite 2956)

Parameter Error V1

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der Errorcodes binär addiert angezeigt. Wird z. B. der Errorcode 0003 angezeigt, so stehen gleichzeitig die Fehler 0001 und 0002 an.

Error (DW#16#...)	Beschreibung
0000	Es steht kein Fehler an.
0001	Der Parameter "Input" ist außerhalb der Istwertgrenzen. • Input > sPid_Cmpt.r_Pv_Hlm oder • Input < sPid_Cmpt.r_Pv_Llm Sie können das Stellglied erst wieder verfahren, wenn der Fehler behoben ist.
0002	Am Parameter "Input_PER" liegt ein ungültiger Wert an. Überprüfen Sie, ob am Analogeingang ein Fehler ansteht.
0004	Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden.
0008	Fehler beim Starten der Erstoptimierung. Der Istwert ist zu nahe am Sollwert. Starten Sie die Nachoptimierung.
0010	Der Sollwert wurde während der Optimierung verändert.
0020	Die Erstoptimierung ist im Automatikbetrieb und während der Nachoptimierung nicht erlaubt.

Error (DW#16#...)	Beschreibung
0080	Fehler bei der Erstoptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert oder der Istwert reagiert nicht wie erwartet. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen. Stellen Sie außerdem sicher, dass der Istwert vor Start der Erstoptimierung nicht stark schwingt.
0100	Fehler während der Optimierung führte zu ungültigen Parametern.
0200	Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat.
0400	Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.
0800	Abtastzeitfehler: PID_Compact wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Falls dieser Fehler bei der Simulation mit PLCSIM aufgetreten ist, beachten Sie die Hinweise unter AUTOHOTSPOT.
1000	Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat.

Siehe auch

Ausgangsparameter PID_Compact V1 (Seite 2942)

Parameter Reset V1

Das Verhalten bei Reset = TRUE ist abhängig von der Version der Anweisung PID_Compact.

Reset-Verhalten PID_Compact ab V.1.1

Durch eine steigende Flanke an Reset wird in die Betriebsart "Inaktiv" gewechselt, Fehler und Warnungen werden zurückgesetzt und der I-Anteil wird gelöscht. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt. Falls zuvor Automatikbetrieb aktiv war, wird der I-Anteil so vorbelegt, dass die Umschaltung stoßfrei erfolgt.

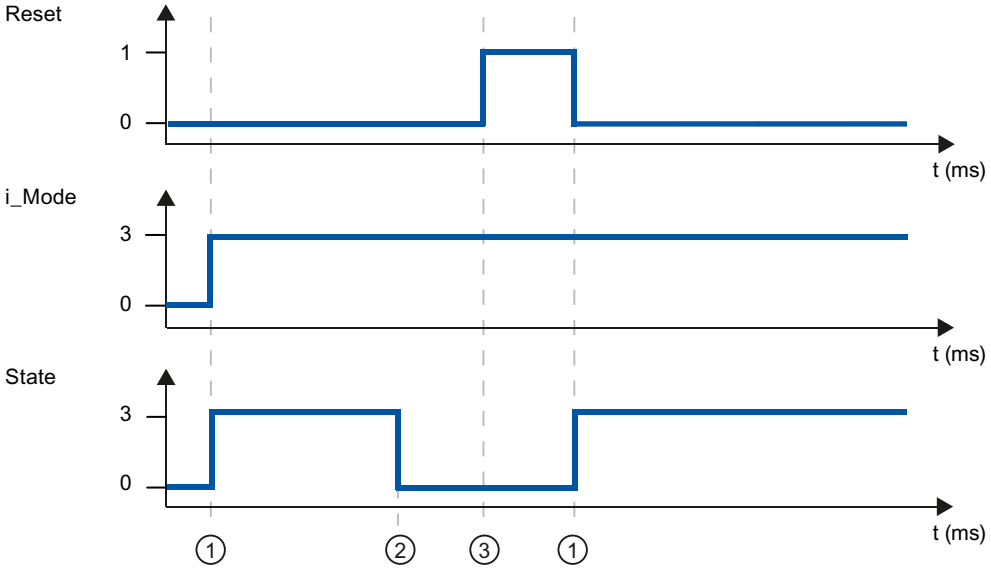

① Einschalten
② Fehler
③ Reset

Reset-Verhalten PID_Compact V.1.0

Durch eine steigende Flanke an Reset wird in die Betriebsart "Inaktiv" gewechselt, Fehler und Warnungen werden zurückgesetzt und der I-Anteil wird gelöscht. Erst durch eine Flanke an i_Mode wird der Regler wieder eingeschaltet.

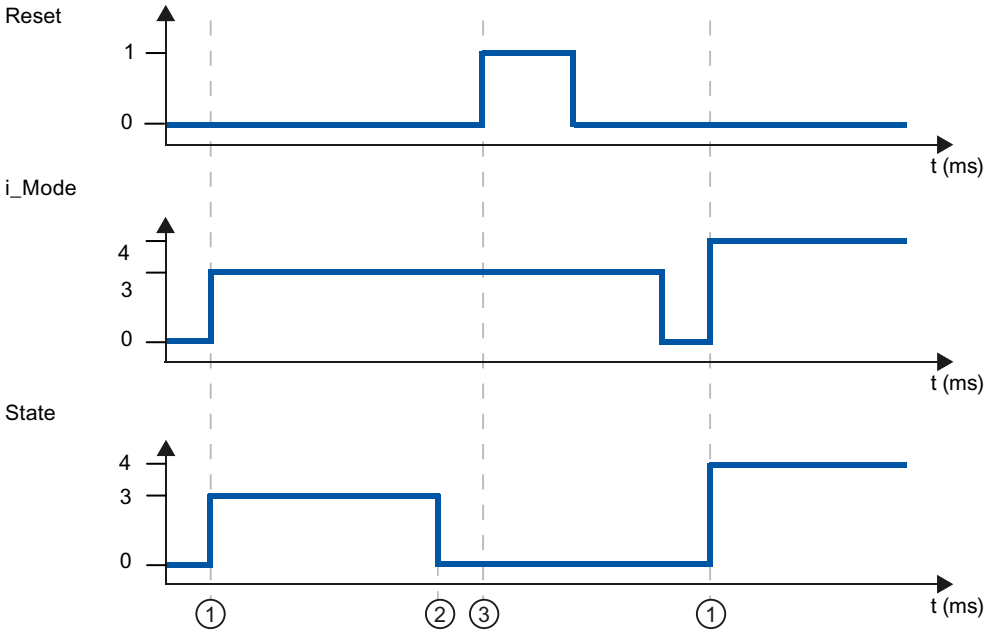

① Einschalten
② Fehler
③ Reset

Variable sd_warning V1

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Variable sd_warning binär addiert angezeigt. Wenn z. B. die Warnung 0003 angezeigt, so stehen gleichzeitig die Warnungen 0001 und 0002 an.

sd_warning (DW#16#....)	Beschreibung
0000	Es liegt keine Warnung vor.
0001	Während der Erstoptimierung wurde der Wendepunkt nicht gefunden.
0002	Während der Nachoptimierung war die Oszillation verstärkt.
0004	Der Sollwert lag außerhalb der eingestellten Grenzen.
0008	Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode "i_CtrlTypeTIR = 3" berechnet.
0010	Die Betriebsart konnte nicht geändert werden, da ManualEnable = TRUE.
0020	Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.
0040	Der Istwert hat eine seiner Warngrenze überschritten.

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0004
- 0020
- 0040

Alle anderen Warnungen werden bei steigender Flanke an Reset gelöscht.

Variable i_Event_SUT V1

i_Event_SUT	Name	Beschreibung
0	SUT_INIT	Erstoptimierung initialisieren
100	SUT_STDABW	Standardabweichung berechnen
200	SUT_GET_POI	Wendepunkt ermitteln
9900	SUT_IO	Erstoptimierung erfolgreich
1	SUT_NIO	Erstoptimierung nicht erfolgreich

Siehe auch

Statische Variablen PID_Compact V1 (Seite 2943)

Parameter State und sRet.i_Mode V1 (Seite 2948)

Variable i_Event_TIR V1

i_Event_TIR	Name	Beschreibung
-100	TIR_FIRST_SUT	Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt.
0	TIR_INIT	Nachoptimierung initialisieren
200	TIR_STDABW	Standardabweichung berechnen
300	TIR_RUN_IN	Versuchen Sollwert zu erreichen
400	TIR_CTRLN	Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war)
500	TIR_OSZIL	Oszillation ermitteln und Parameter berechnen
9900	TIR_IO	Nachoptimierung erfolgreich
1	TIR_NIO	Nachoptimierung nicht erfolgreich

Siehe auch

Statische Variablen PID_Compact V1 (Seite 2943)

Parameter State und sRet.i_Mode V1 (Seite 2948)

PID_3Step

Neuerungen PID_3Step

PID_3Step V2.3

- Ab PID_3Step Version 2.3 kann die Überwachung und Begrenzung der Verfahrzeit mit Config.VirtualActuatorLimit = 0.0 deaktiviert werden.

PID_3Step V2.2

- **Einsatz mit S7-1200**
  Ab PID_3Step V2.2 ist die Anweisung mit V2-Funktionalität auch auf einer S7-1200 ab der Firmware-Version 4.0 einsetzbar.

PID_3Step V2.0

- **Verhalten im Fehlerfall**
  Das Verhalten bei ActivateRecoverMode = TRUE wurde grundlegend überarbeitet. PID_3Step verhält sich in der Default-Einstellung Fehler toleranter.

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn Sie die Default-Einstellung verwenden, bleibt PID_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

  Mit dem Eingangsparameter ErrorAck quittieren Sie die Fehler und Warnungen, ohne den Regler neu zu starten oder den I-Anteil zu löschen.
  Durch einen Wechsel der Betriebsart werden nicht mehr anstehende Fehler nicht quittiert.

- **Wechsel der Betriebsarten**
  Die Betriebsart geben Sie am Durchgangsparameter Mode vor und starten diese über eine steigende Flanke an ModeActivate. Die Variable Retain.Mode ist entfallen.
  Die Stellzeitmessung kann nicht mehr mit GetTransitTime.Start gestartet werden, sondern nur noch mit Mode = 6 und einer steigende Flanke an ModeActivate.

- **Multiinstanzfähigkeit**
  Sie können PID_3Step als Multiinstanz-DB aufrufen. Dann wird kein Technologieobjekt angelegt und es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_3Step direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

- **Anlaufverhalten**
  Die an Mode vorgegebene Betriebsart wird auch gestartet bei einer fallenden Flanke an Reset und bei Kaltstart der CPU, wenn RunModeByStartup = TRUE ist.

- **ENO-Verhalten**
  ENO wird in Abhängigkeit von der Betriebsart gesetzt.
  Wenn State = 0 ist, dann ist ENO = FALSE.
  Wenn State ≠ 0 ist, dann ist ENO = TRUE.
- **Handbetrieb**
  Die Eingangsparameter Manual_UP und Manual_DN arbeiten nicht mehr Flanken gesteuert. Der Flanken gesteuerte Handbetrieb ist weiterhin über die Variablen ManualUpInternal und ManualDnInternal möglich.
  Im "Handbetrieb ohne Anschlagsignale" (Mode = 10) werden die Anschlagsignale Actuator_H und Actuator_L ignoriert, obwohl sie eingeschaltet sind.
- **Vorbelegung der PID-Parameter**
  Folgende Vorbelegungen wurden geändert:
  - Gewichtung des P-Anteils (PWeighting) von 0.0 auf 1.0
  - Gewichtung des D-Anteils (DWeighting) von 0.0 auf 1.0
  - Koeffizient für den Differenzierverzug (TdFiltRatio) von 0.0 auf 0.2
- **Begrenzung der Motorstellzeit**
  Sie konfigurieren an der Variable Config.VirtualActuatorLimit, um wie viel % der Motorstellzeit das Stellglied maximal in eine Richtung verfahren wird.
- **Sollwertvorgabe während der Optimierung**
  Die erlaubte Schwankung des Sollwerts während der Optimierung konfigurieren Sie an der Variable CancelTuningLevel.
- **Aufschalten einer Störgröße**
  Sie können am Parameter Disturbance eine Störgröße aufschalten.
- **Fehlerbehebung**
  Wenn die Anschlagsignale nicht aktiviert sind (ActuatorEndStopOn = FALSE), werden Actuator_H und Actuator_L nicht mehr für die Ermittlung von ScaledFeedback berücksichtigt.

PID_3Step V1.1

- **Handbetrieb beim Anlauf der CPU**
  Wenn beim Start der CPU ManualEnable = TRUE, startet PID_3Step im Handbetrieb. Eine steigende Flanke an ManualEnable ist nicht notwendig.
- **Verhalten im Fehlerfall**
  Die Variable ActivateRecoverMode wirkt im Handbetrieb nicht mehr.
- **Fehlerbehebung**
  Die Variable Progress wird nach erfolgreicher Optimierung oder Stellzeitmessung zurückgesetzt.

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von PID_3Step einsetzen können.

CPU	FW	PID_3Step
S7-1200	ab V4.2	V2.3 V2.2 V1.1
	V4.0 bis V4.1	V2.2 V1.1
	V3.x	V1.1 V1.0
	V2.x	V1.1 V1.0
	V1.x	-
S7-1500	ab V2.0	V2.3 V2.2 V2.1 V2.0
	V1.5 bis V1.8	V2.2 V2.1 V2.0
	V1.1	V2.1 V2.0
	V1.0	V2.0

CPU-Bearbeitungszeit und Speicherbedarf PID_3Step V2.x

CPU-Bearbeitungszeit

Typische CPU-Bearbeitungszeiten des Technologieobjekts PID_3Step ab der Version V2.0 in Abhängigkeit vom CPU-Typ.

CPU	Typ. CPU-Bearbeitungszeit PID_3Step V2.x
CPU 1211C ≥ V4.0	410 µs
CPU 1215C ≥ V4.0	410 µs
CPU 1217C ≥ V4.0	410 µs
CPU 1505S ≥ V1.0	50 µs
CPU 1510SP-1 PN ≥ V1.6	120 µs
CPU 1511-1 PN ≥ V1.5	120 µs
CPU 1512SP-1 PN ≥ V1.6	120 µs
CPU 1516-3 PN/DP ≥ V1.5	65 µs
CPU 1518-4 PN/DP ≥ V1.5	5 µs

Speicherbedarf

Speicherbedarf eines Instanz-DB des Technologieobjekts PID_3Step ab der Version V2.0.

Speicherbedarf	Speicherbedarf des Instanz-DB von PID_3Step V2.x
Ladespeicherbedarf	ca. 15000 Bytes
Gesamter Arbeitsspeicherbedarf	1040 Bytes
Remanenter Arbeitsspeicherbedarf	60 Bytes

PID_3Step V2

Beschreibung PID_3Step V2

Beschreibung

Mit der Anweisung PID_3Step konfigurieren Sie einen PID-Regler mit Selbstoptimierung für Ventile oder Stellglieder mit integrierendem Verhalten.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert anfahren
- Stellzeitmessung
- Fehlerüberwachung
- Ersatzausgangswert anfahren mit Fehlerüberwachung
- Handbetrieb ohne Anschlagsignale

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

PID-Algorithmus

PID_3Step ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der PID-Algorithmus arbeitet nach folgender Formel:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Δy	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
T_D	Differenzierzeit
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)
c	Gewichtung des D-Anteils

Blockschaltbild ohne Stellungsrückmeldung

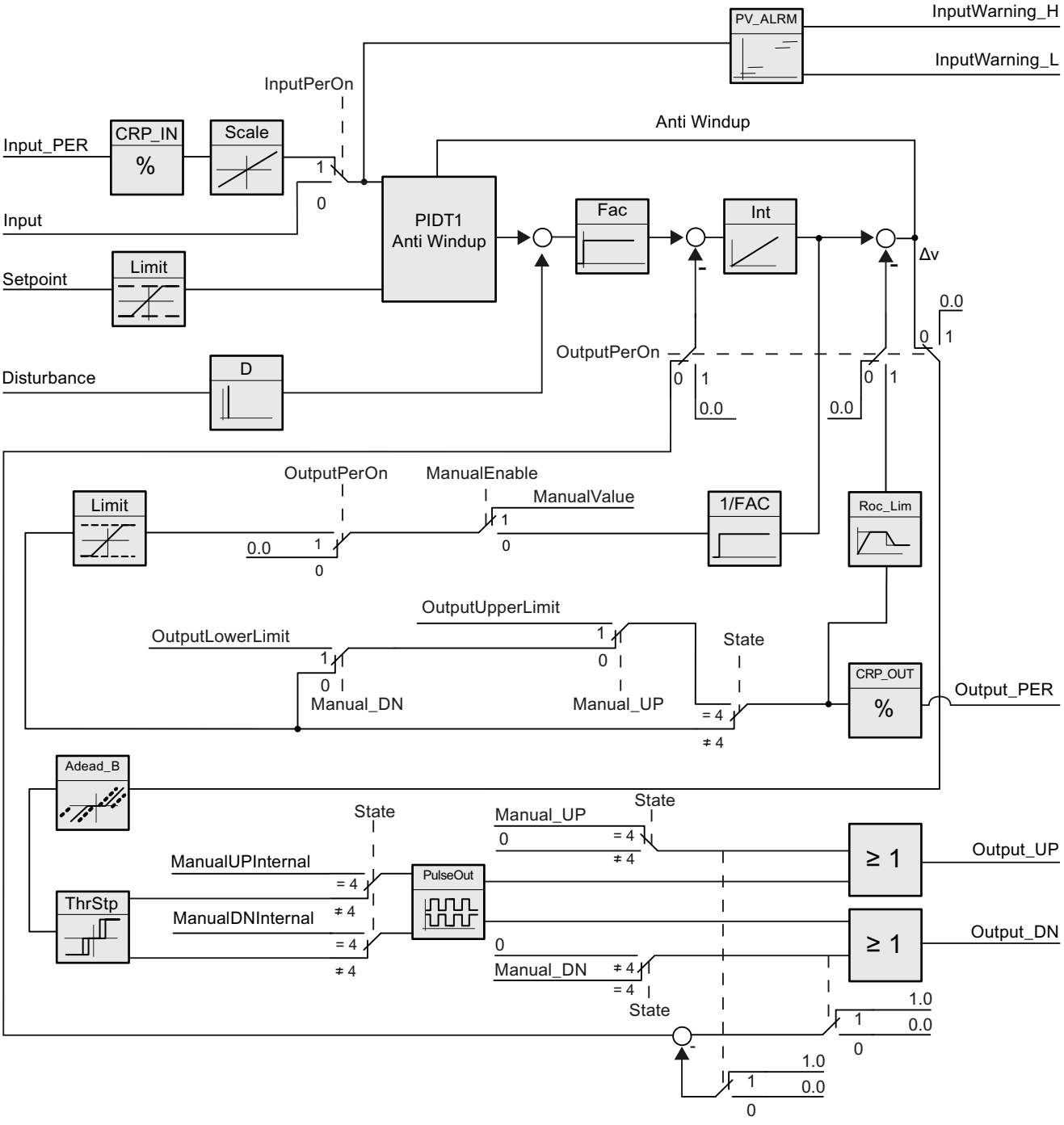

Blockschaltbild mit Stellungsrückmeldung

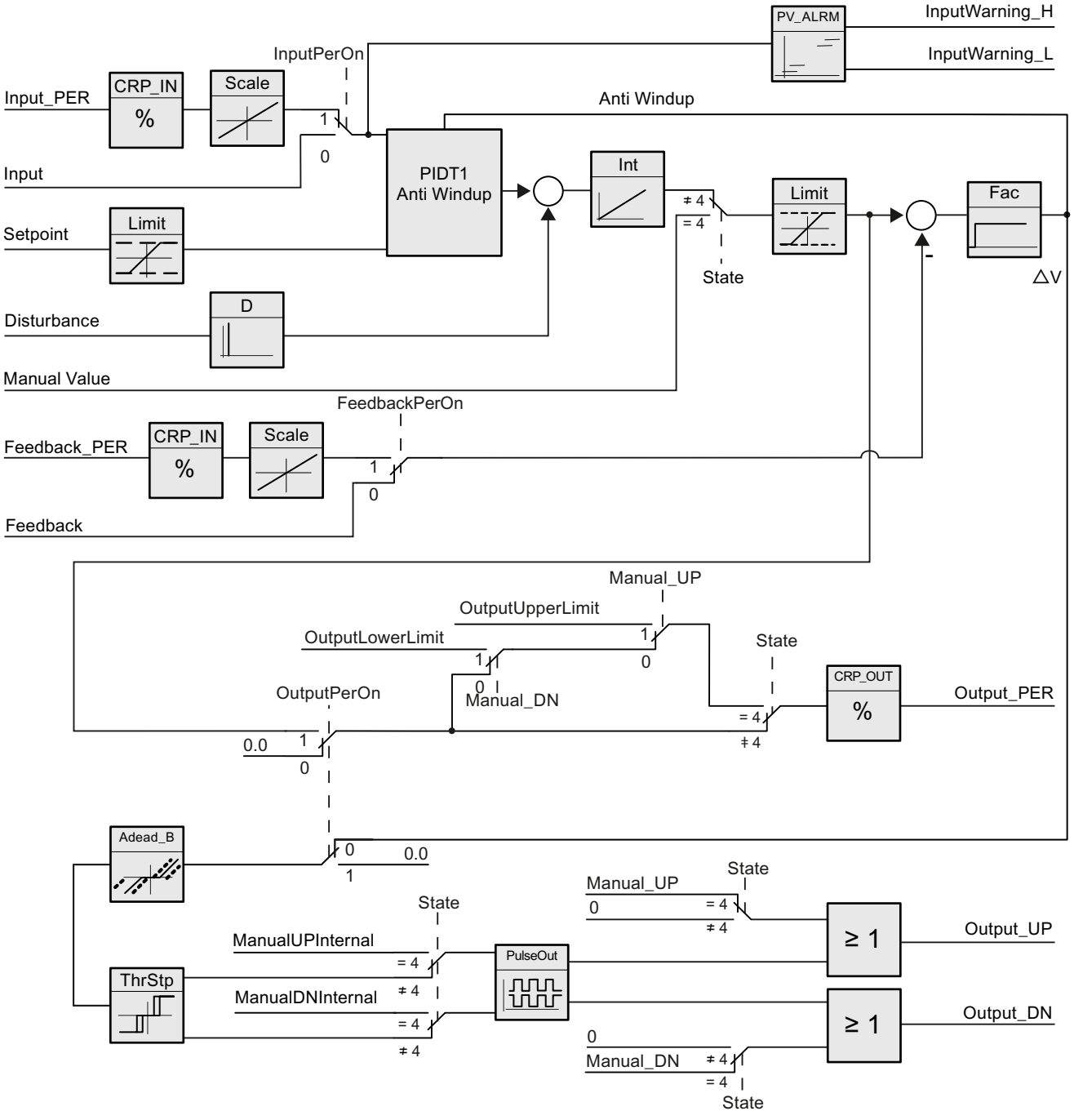

Blockschaltbild PIDT1 mit Anti-Windup

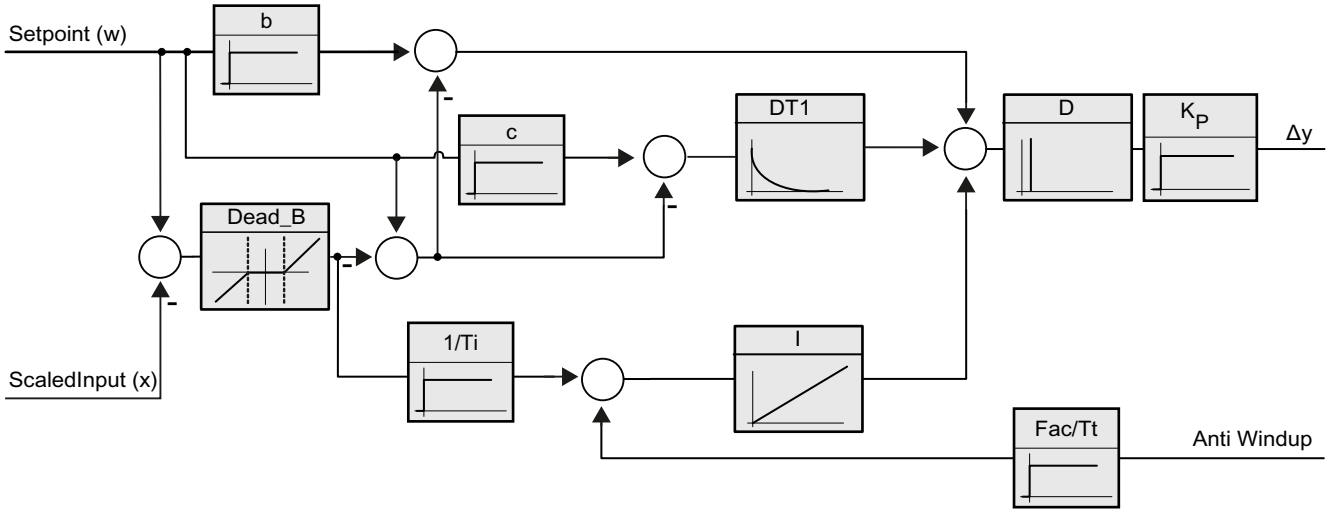

Aufruf

PID_3Step wird im konstanten Zeitraster eines Weckalarm-OBs aufgerufen.

Wenn Sie PID_3Step als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_3Step direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID_3Step vollständig laden.

AUTOHOTSPOT

Anlauf

Bei Anlauf der CPU startet PID_3Step in der Betriebsart, die am Durchgangsparameter Mode gespeichert ist. Um PID_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

Verhalten im Fehlerfall

Im Automatikbetrieb und während der Inbetriebnahme hängt das Verhalten im Fehlerfall von den Variablen ErrorBehaviour und ActivateRecoverMode ab. Im Handbetrieb ist das Verhalten unabhängig von ErrorBehaviour und ActivateRecoverMode. Wenn ActivateRecoverMode = TRUE ist, hängt das Verhalten zusätzlich vom aufgetretenen Fehler ab.

ErrorBe-haviour	ActivateReco-verMode	Konfigurationseditor > Stellgliedeinstellung > Output setzen auf	Verhalten
FALSE	FALSE	Aktuellen Ausgangswert	Wechsel in Betriebsart "Inaktiv" (State = 0) Das Stellglied bleibt in der aktuellen Position.
FALSE	TRUE	Aktuellen Ausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Fehlerüberwachung" (State = 7) Das Stellglied bleibt für die Fehlerdauer in der aktuellen Position.
TRUE	FALSE	Ersatzausgangswert	Wechsel in Betriebsart "Ersatzausgangswert anfahren" (State = 5) Das Stellglied wird auf den konfigurierten Ersatzausgangswert gefahren. Wechsel in Betriebsart "Inaktiv" (State = 0) Das Stellglied bleibt in der aktuellen Position.
TRUE	TRUE	Ersatzausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" (State = 8) Das Stellglied wird auf den konfigurierten Ersatzausgangswert gefahren. Wechsel in Betriebsart "Fehlerüberwachung" (State = 7)

PID_3Step verwendet im Handbetrieb ManualValue als Ausgangswert, außer bei den folgenden Fehlern:

- 2000h: Ungültiger Wert am Parameter Feedback_PER.
- 4000h: Ungültiger Wert am Parameter Feedback.
- 8000h: Fehler bei der digitalen Stellungsrückmeldung.

Sie können die Position des Stellglieds dann nur mit Manual_UP und Manual_DN verändern, nicht mit ManualValue:

Der Parameter Error zeigt, ob in diesem Takt ein Fehler aufgetreten ist. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits wird durch eine steigende Flanke an Reset oder ErrorAck zurückgesetzt.

Siehe auch

Parameter State und Mode V2 (Seite 2983)

Parameter ErrorBits V2 (Seite 2988)

Arbeitsweise PID_3Step V2

Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001h).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warning = 0040h) und der Ausgangsparameter InputWarning_H oder InputWarning_L wird TRUE.

Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID_3Step begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID_3Step prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit_H oder SetpointLimit_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

- Oberer Anschlag: Config.FeedbackScaling.UpperPointOut
- Unter Anschlag: Config.FeedbackScaling.LowerPointOut

Es muss gelten:

UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
FALSE	TRUE	FALSE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
FALSE	TRUE	TRUE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
TRUE	FALSE	FALSE	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
TRUE	TRUE	FALSE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
TRUE	TRUE	TRUE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %

Falls OutputPerOn = FALSE und FeedbackOn = FALSE, können Sie den Ausgangswert nicht begrenzen. Output_UP und Output_DN werden dann bei Actuator_H = TRUE oder Actuator_L = TRUE zurück gesetzt. Wenn auch keine Anschlagsignale vorhanden sind, werden Output_UP und Output_DN nach einer Verfahrzeit von Config.VirtualActuatorLimit × Retain.TransitTime/100 zurück gesetzt. Ab PID_3Step Version 2.3 kann die Überwachung und Begrenzung der Verfahrzeit mit Config.VirtualActuatorLimit = 0.0 deaktiviert werden.

Der Ausgangswert beträgt 27648 bei 100 % und -27648 bei -100 %. PID_3Step muss das Ventil vollständig schließen können.

Hinweis

Einsatz mit zwei oder mehr Stellgliedern

PID_3 Step ist nicht für den Einsatz mit zwei oder mehr Stellgliedern (z. B. in Heizen-/Kühlen-Anwendungen) geeignet, da unterschiedliche Stellglieder auch unterschiedliche PID-Parameter benötigen, um ein gutes Regelverhalten zu erzielen.

Ersatzausgangswert

PID_3Step kann im Fehlerfall einen Ersatzausgangswert ausgeben und das Stellglied in eine sichere Position stellen, die Sie an der Variablen SavePosition vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden bei Verwendung auf Gültigkeit überwacht:

- Setpoint
- Input
- Input_PER
- Input_PER
- Feedback
- Feedback_PER
- Disturbance
- ManualValue
- SavePosition
- Output_PER

Abtastzeit PID_3Step überwachen

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID_3Step misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (ErrorBits = 0800h).

Während der Optimierung tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert

Im Automatikbetrieb tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,5 x alter Mittelwert
- Neuer Mittelwert <= 0,5 x alter Mittelwert

Wenn Sie die Überwachung der Abtastzeit ausschalten (CycleTime.EnMonitoring = FALSE), können Sie PID_3Step auch im OB1 aufrufen. Sie müssen dann aufgrund der schwankenden Abtastzeit eine schlechtere Regelqualität akzeptieren.

Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_3Step werden bei jedem Aufruf durchgeführt.

Motorstellzeit messen

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal für Config.VirtualActuatorLimit × Retain.TransitTime/100 in eine Richtung bewegt. PID_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren. Die Motorstellzeit können Sie während der Inbetriebnahme messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der Selbstoptimierung und dem Anti-Windup-Verhalten für eine korrekte Arbeitsweise notwendig. Konfigurieren Sie deshalb die Motorstellzeit mit dem Wert, den der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen.

Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID_3Step arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

Änderungen der Schnittstelle PID_3Step V2

Die folgende Tabelle zeigt, was sich an der Schnittstelle der Anweisung PID_3Step geändert hat.

PID_3Step V1	PID_3Step V2	Änderung
Input_PER	Input_PER	Datentyp von Word zu Int
Feedback_PER	Feedback_PER	Datentyp von Word zu Int
	Disturbance	Neu
Manual_UP	Manual_UP	Funktion
Manual_DN	Manual_DN	Funktion
	ErrorAck	Neu
	ModeActivate	Neu
Output_PER	Output_PER	Datentyp von Word zu Int
	ManualUPInternal	Neu
	ManualDNInternal	Neu
	CancelTuningLevel	Neu
	VirtualActuatorLImit	Neu
Config.Loadbackup	Loadbackup	Umbenannt
Config.TransitTime	Retain.TransitTime	Umbenannt und Remanenz hinzugefügt
GetTransitTime.Start		Ersetzt durch Mode und ModeActivate.
SUT.CalculateSUTPa-rams	SUT.CalculateParams	Umbenannt
SUT.TuneRuleSUT	SUT.TuneRule	Umbenannt
TIR.CalculateTIRParams	TIR.CalculateParams	Umbenannt
TIR.TuneRuleTIR	TIR.TuneRule	Umbenannt
Retain.Mode	Mode	Funktion Deklaration von Static zu Durchgangsparameter

Eingangsparameter PID_3Step V2

Tabelle 4-64

Parameter	Datentyp	Vorbelegung	Beschreibung
Setpoint	REAL	0.0	Sollwert des PID-Reglers im Automatikbetrieb
Input	REAL	0.0	Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.
Input_PER	INT	0	Ein Analogeingang wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.
Actuator_H	BOOL	FALSE	Digitale Stellungsrückmeldung des Ventils für den oberen Anschlag Wenn Actuator_H = TRUE ist die Stellung des Ventils am oberen Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.
Actuator_L	BOOL	FALSE	Digitale Stellungsrückmeldung des Ventils für den unteren Anschlag Wenn Actuator_L = TRUE ist die Stellung des Ventils am unteren Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.
Feedback	REAL	0.0	Stellungsrückmeldung des Ventils Wenn Sie den Parameter Feedback verwenden, muss Config.FeedbackPerOn = FALSE sein.
Feedback_PER	INT	0	Analoge Stellungsrückmeldung eines Ventils Wenn Sie den Parameter Feedback_PER verwenden, muss Config.FeedbackPerOn = TRUE sein. Feedback_PER wird skaliert anhand der Variablen: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut
Disturbance	REAL	0.0	Störgröße oder Vorsteuerungswert
ManualEnable	BOOL	FALSE	• Flanke FALSE -> TRUE aktiviert die Betriebsart "Handbetrieb", State = 4, Mode bleibt unverändert. Solange ManualEnable = TRUE ist, können Sie die Betriebsart nicht über eine steigende Flanke an ModeActivate ändern und den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE aktiviert die Betriebsart, die durch Mode vorgegeben wird. Es wird empfohlen, die Betriebsart nur über ModeActivate zu ändern.
ManualValue	REAL	0.0	Im Handbetrieb wird die absolute Position des Ventils vorgegeben. ManualValue wird nur ausgewertet, wenn Sie Output_PER verwenden **oder** eine Stellungsrückmeldung verfügbar ist.

Parameter	Datentyp	Vorbelegung	Beschreibung
Manual_UP	BOOL	FALSE	• Manual_UP = TRUE Das Ventil wird geöffnet, auch wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden. Das Ventil wird nicht mehr bewegt, wenn der obere Anschlag erreicht ist. Siehe auch Config.VirtualActuatorLimit • Manual_UP = FALSE Wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden, wird das Ventil auf ManualValue gefahren. Andernfalls wird das Ventil nicht mehr bewegt. Wenn Manual_UP und Manual_DN gleichzeitig TRUE gesetzt werden, wird das Ventil nicht bewegt.
Manual_DN	BOOL	FALSE	• Manual_DN = TRUE Das Ventil wird geschlossen, auch wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden. Das Ventil wird nicht mehr bewegt, wenn der untere Anschlag erreicht ist. Siehe auch Config.VirtualActuatorLimit • Manual_DN = FALSE Wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden, wird das Ventil auf ManualValue gefahren. Andernfalls wird das Ventil nicht mehr bewegt.
ErrorAck	BOOL	FALSE	• Flanke FALSE -> TRUE ErrorBits und Warning werden zurückgesetzt.
Reset	BOOL	FALSE	Führt einen Neustart des Reglers durch. • Flanke FALSE -> TRUE – Wechsel in Betriebsart "Inaktiv" – ErrorBits und Warnings werden zurückgesetzt. • Solange Reset = TRUE ist, – bleibt PID_3Step in der Betriebsart "Inaktiv" (State = 0). – können Sie die Betriebsart nicht über Mode und ModeActivate oder ManualEnable ändern. – können Sie den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE – Falls ManualEnable = FALSE, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist. – Falls Mode = 3, wird der I-Anteil behandelt, wie durch die Variable IntegralResetMode konfiguriert.
ModeActivate	BOOL	FALSE	• Flanke FALSE -> TRUE PID_3Step wechselt in die Betriebsart, die an Mode gespeichert ist.

Ausgangsparameter PID_3Step V2

Tabelle 4-65

Parameter	Datentyp	Vorbelegung	Beschreibung
ScaledInput	REAL	0.0	Skalierter Istwert
ScaledFeedback	REAL	0.0	Skalierte Stellungsrückmeldung Bei einem Stellglied ohne Stellungsrückmeldung zeigt ScaledFeedback die Position des Stellglieds sehr ungenau an. Dann darf ScaledFeedback nur zur groben Abschätzung der aktuellen Position genutzt werden.
Output_UP	BOOL	FALSE	Digitaler Ausgangswert zum Öffnen des Ventils Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_UP verwendet.
Output_DN	BOOL	FALSE	Digitaler Ausgangswert zum Schließen des Ventils Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_DN verwendet.
Output_PER	INT	0	Analoger Ausgangswert Wenn Config.OutputPerOn = TRUE, wird Output_PER verwendet. Verwenden Sie Output_PER, falls Sie als Stellglied ein Ventil nutzen, welches über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert wird, z. B. 0...10V oder 4...20mA. Der Wert an Output_PER entspricht der Zielstellung des Ventils z. B. Output_PER = 13824, wenn das Ventil zu 50% geöffnet werden soll.
SetpointLimit_H	BOOL	FALSE	Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht (Setpoint ≥ Config.SetpointUpperLimit). Der Sollwert wird auf Config.SetpointUpperLimit begrenzt.
SetpointLimit_L	BOOL	FALSE	Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht (Setpoint ≤ Config.SetpointLowerLimit). Der Sollwert wird auf Config.SetpointLowerLimit begrenzt.
InputWarning_H	BOOL	FALSE	Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.
InputWarning_L	BOOL	FALSE	Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.

Parameter	Datentyp	Vorbelegung	Beschreibung
State	INT	0	Der Parameter State (Seite 2983) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit dem Eingangsparameter Mode und einer steigenden Flanke an Mode-Activate. • State = 0: Inaktiv • State = 1: Erstoptimierung • State = 2: Nachoptimierung • State = 3: Automatikbetrieb • State = 4: Handbetrieb • State = 5: Ersatzausgangswert anfahren • State = 6: Stellzeitmessung • State = 7: Fehlerüberwachung • State = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung • State = 10: Handbetrieb ohne Anschlagsignale
Error	BOOL	FALSE	Wenn Error = TRUE, liegt in diesem Takt mindestens eine Fehlermeldungen vor.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 2988) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.

Siehe auch

Parameter State und Mode V2 (Seite 2983)

Parameter ErrorBits V2 (Seite 2988)

Durchgangsparameter PID_3Step V2

Tabelle 4-66

Parameter	Datentyp	Vorbelegung	Beschreibung
Mode	INT	4	An Mode geben Sie die Betriebsart vor, in die PID_3Step wechseln soll. Möglich sind: • Mode = 0: Inaktiv • Mode = 1: Erstoptimierung • Mode = 2: Nachoptimierung • Mode = 3: Automatikbetrieb • Mode = 4: Handbetrieb • Mode = 6: Stellzeitmessung • Mode = 10: Handbetrieb ohne Anschlagsignale Die Betriebsart wird aktiviert durch: • Steigende Flanke an ModeActivate • Fallende Flanke an Reset • Fallende Flanke an ManualEnable • Kaltstart der CPU, wenn RunModeByStartup = TRUE Mode ist remanent. Eine detaillierte Beschreibung der Betriebsarten finden Sie unter Parameter State und Mode V2 (Seite 2983) .

Statische Variablen PID_3Step V2

Hinweis

Ändern Sie die mit (1) gekennzeichneten Variablen nur in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden.

Variable	Datentyp	Vorbelegung	Beschreibung
ManualUpInternal	BOOL	FALSE	Im Handbetrieb öffnet jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. ManualUpInternal wird nur ausgewertet, wenn Sie weder Output_PER noch eine Stellungsrückmeldung verwenden. Diese Variable wird im Inbetriebnahmedialog verwendet.
ManualDnInternal	BOOL	FALSE	Im Handbetrieb schließt jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. ManualDnInternal wird nur ausgewertet, wenn Sie weder Output_PER noch eine Stellungsrückmeldung verwenden. Diese Variable wird im Inbetriebnahmedialog verwendet.
ActivateRecoverMode	BOOL	TRUE	Die Variable ActivateRecoverMode V2 (Seite 2992) bestimmt das Verhalten im Fehlerfall.

Variable	Datentyp	Vorbelegung	Beschreibung
RunModeByStartup	BOOL	TRUE	Nach CPU Neustart Betriebsart an Mode aktivieren Wenn RunModeByStartup = TRUE, startet PID_3Step nach CPU-Anlauf in der Betriebsart, die an Mode gespeichert ist. Wenn RunModeByStartup = FALSE, bleibt PID_3Step nach CPU-Anlauf in der Betriebsart "Inaktiv".
LoadBackUp	BOOL	FALSE	Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.
PhysicalUnit	INT	0	Physikalische Einheit des Ist- und Sollwerts, z. B. ºC oder ºF.
PhysicalQuantity	INT	0	Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur
ErrorBehaviour	BOOL	FALSE	Wenn ErrorBehaviour = FALSE, bleibt das Ventil im Fehlerfall in der aktuellen Position und der Regler wechselt direkt in die Betriebsart "Inaktiv" oder "Fehlerüberwachung". Wenn ErrorBehaviour = TRUE, wird im Fehlerfall das Stellglied auf den Ersatzausgangswert gefahren und erst dann in die Betriebsart "Inaktiv" oder "Fehlerüberwachung" gewechselt. Wenn die folgenden Fehler auftreten, kann das Ventil nicht mehr auf einen konfigurierten Ersatzausgangswert gefahren werden. • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung. • 20000h: Ungültiger Wert an der Variablen SavePosition.
Warning	DWORD	DW#16#0	Die Variable Warning (Seite 2983) zeigt die Warnungen seit Reset = TRUE oder ErrorAck =TRUE. Warning ist remanent. Zyklische Warnungen (z. B. Istwertwarnung) werden solange angezeigt wie die Ursache der Warnung anliegt. Entfällt der Grund, werden sie automatisch gelöscht. Nichtzyklische Warnungen (z. B. Wendepunkt nicht gefunden) bleiben bestehen und werden wie Fehler gelöscht.
SavePosition	REAL	0.0	Ersatzausgangswert Wenn ErrorBehaviour = TRUE, wird das Stellglied im Fehlerfall in eine für die Anlage sichere Position gefahren. Sobald der Ersatzausgangswert erreicht wurde, wechselt PID_3Step in Abhängigkeit von ActivateRecoverMode die Betriebsart.
CurrentSetpoint	REAL	0.0	Aktuell aktiver Sollwert. Dieser Wert wird beim Start der Optimierung eingefroren.
CancelTuningLevel	REAL	10.0	Zulässige Schwankung des Sollwerts während der Optimierung. Die Optimierung wird erst abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel
Progress	REAL	0.0	Fortschritt der Optimierung in Prozent (0.0 - 100.0)
Config.InputPerOn(1)	BOOL	TRUE	Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.OutputPerOn(1)	BOOL	FALSE	Wenn OutputPerOn = TRUE, wird der Parameter Output_PER verwendet. Wenn OutputPerOn = FALSE, werden die Parameter Ouput_UP und Output_DN verwendet.
Config.InvertControl(1)	BOOL	FALSE	Invertieren des Regelsinns Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.
Config.FeedbackOn(1)	BOOL	FALSE	Wenn FeedbackOn = FALSE, wird eine Stellungsrückmeldung simuliert. Wenn FeedbackOn = TRUE, wird generell die Stellungsrückmeldung aktiviert.
Config.FeedbackPerOn(1)	BOOL	FALSE	FeedbackPerOn ist nur wirksam, wenn FeedbackOn = TRUE. Wenn FeedbackPerOn = TRUE, wird der Analogeingang für die Stellungsrückmeldung verwendet (Parameter Feedback_PER). Wenn FeedbackPerOn = FALSE, wird der Parameter Feedback für die Stellungsrückmeldung verwendet.
Config.ActuatorEndStopOn(1)	BOOL	FALSE	Wenn ActuatorEndStopOn = TRUE, wird die digitale Stellungsrückmeldung Actuator_L und Actuator_H berücksichtigt.
Config.InputUpperLimit(1)	REAL	120.0	Obergrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit(1)	REAL	0.0	Untergrenze des Istwerts InputLowerLimit < InputUpperLimit
Config.InputUpperWarning(1)	REAL	+3.402822e+38	Obere Warngrenze des Istwerts Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als obere Warngrenze verwendet. Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als obere Warngrenze verwendet. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning(1)	REAL	-3.402822e+38	Untere Warngrenze des Istwerts Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als Untere Warngrenze verwendet. Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit

Variable	Datentyp	Vorbelegung	Beschreibung
Config.OutputUpperLimit(1)	REAL	100.0	Obergrenze des Ausgangswerts Folgender Wertebereich ist zulässig: UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit Weitere Details siehe OutputLowerLimit
Config.OutputLowerLimit(1)	REAL	0.0	Untergrenze des Ausgangswerts Folgender Wertebereich ist zulässig: OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut Bei Verwendung von Output_PER entspricht eine Ausgangswertgrenze von -100 % dem Wert Output_PER = -27648; 100 % entsprechen dem Wert Output_PER = 27648. Wenn OutputPerOn = FALSE und FeedbackOn = FALSE, werden OutputLowerLimit und OutputUpperLimit nicht ausgewertet. Output_UP und Output_DN werden dann bei Actuator_H = TRUE oder Actuator_L = TRUE zurück gesetzt (wenn ActuatorEndStopOn = TRUE) oder nach einer Verfahrzeit von Config.VirtualActuatorLimit * Retain.TransitTime/100 (wenn ActuatorEndStopOn = FALSE).
Config.SetpointUpperLimit(1)	REAL	+3.402822e+38	Obergrenze des Sollwerts Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert vorbelegt. Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.
Config.SetpointLowerLimit(1)	REAL	-3.402822e+38	Untergrenze des Sollwerts Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert vorbelegt. Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
Config.MinimumOnTime(1)	REAL	0.0	Minimale Einschaltzeit Zeit in Sekunden, die der Stellantrieb mindestens eingeschaltet sein muss. Config.MinimumOnTime ist nur wirksam, falls Output_UP und Output_DN verwendet werden (Config.OutputPerOn = FALSE).
Config.MinimumOffTime(1)	REAL	0.0	Minimale Ausschaltzeit Zeit in Sekunden, die der Stellantrieb mindestens ausgeschaltet sein muss. Config.MinimumOffTime ist nur wirksam, falls Output_UP und Output_DN verwendet werden (Config.OutputPerOn = FALSE).

Variable	Datentyp	Vorbelegung	Beschreibung
Config.VirtualActuatorLimit(1)	REAL	150.0	Wenn alle folgenden Bedingungen erfüllt sind, wird das Stellglied maximal für die Dauer von VirtualActuatorLimit × Retain.TransitTime/100 in eine Richtung bewegt und die Warnung 2000h ausgegeben: • Config.OutputPerOn = FALSE • Config.ActuatorEndStopOn = FALSE • Config.FeedbackOn = FALSE Wenn Config.OutputPerOn = FALSE und Config.ActuatorEndStopOn = TRUE oder Config.FeedbackOn = TRUE sind, wird nur die Warnung 2000h ausgegeben. Wenn Config.OutputPerOn = TRUE ist, wird VirtualActuatorLimit nicht berücksichtigt. Ab PID_3Step Version 2.3 kann die Überwachung und Begrenzung der Verfahrzeit mit Config.VirtualActuatorLimit = 0.0 deaktiviert werden.
Config.InputScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Input_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointIn(1)	REAL	0.0	Skalierung Input_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.InputScaling.UpperPointOut(1)	REAL	100.0	Skalierter oberer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointOut(1)	REAL	0.0	Skalierter unterer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.FeedbackScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Feedback_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.
Config.FeedbackScaling.LowerPointIn(1)	REAL	0.0	Skalierung Feedback_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.FeedbackScaling.UpperPointOut(1)	REAL	100.0	Oberer Anschlag Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet. Der zulässige Wertebereich ist abhängig von der Konfiguration. • FeedbackOn = FALSE: UpperPointOut = 100.0 • FeedbackOn = TRUE: UpperPointOut = 100.0 oder 0.0 UpperPointOut ≠ LowerPointOut
Config.FeedbackScaling.LowerPointOut(1)	REAL	0.0	Unterer Anschlag Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet. Der zulässige Wertebereich ist abhängig von der Konfiguration. • FeedbackOn = FALSE: LowerPointOut = 0.0 • FeedbackOn = TRUE: LowerPointOut = 0.0 oder -100.0 LowerPointOut ≠ UpperPointOut
GetTransitTime.InvertDirection	BOOL	FALSE	Wenn InvertDirection = FALSE, wird das Ventil zum Ermitteln der Stellzeit vollständig geöffnet, geschlossen und wieder geöffnet. Wenn InvertDirection = TRUE, wird das Ventil vollständig geschlossen, geöffnet und wieder geschlossen.
GetTransitTime.SelectFeedback	BOOL	FALSE	Wenn SelectFeedback = TRUE, wird Feedback_PER oder Feedback bei der Stellzeitmessung berücksichtigt. Wenn SelectFeedback = FALSE, wird Actuator_H und Actuator_L bei der Stellzeitmessung berücksichtigt.
GetTransitTime.State	INT	0	Aktuelle Phase der Stellzeitmessung • State = 0: Inaktiv • State = 1: Ventil vollständig öffnen • State = 2: Ventil vollständig schließen • State = 3: Ventil auf Zielstellung (NewOutput) stellen • State = 4: Stellzeitmessung erfolgreich beendet • State = 5: Stellzeitmessung abgebrochen
GetTransitTime.NewOutput	REAL	0.0	Zielstellung für die Stellzeitmessung mit Stellungsrückmeldung Die Zielstellung muss innerhalb "Oberer Anschlag" und "Unterer Anschlag" liegen. Die Differenz zwischen NewOutput und ScaledFeedback muss mindestens 50 % des zulässigen Stellbereichs sein.
CycleTime.StartEstimation	BOOL	TRUE	Wenn StartEstimation = TRUE, wird die Messung der Abtastzeit PID_3Step gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.

Variable	Datentyp	Vorbelegung	Beschreibung
CycleTime.EnEstimation	BOOL	TRUE	Wenn EnEstimation = TRUE, wird die Abtastzeit PID_3Step berechnet. Wenn CycleTime.EnEstimation = FALSE, wird die Abtastzeit PID_3Step nicht berechnet und Sie müssen CycleTime.Value manuell korrekt konfigurieren.
CycleTime.EnMonitoring	BOOL	TRUE	Wenn EnMonitoring = TRUE, wird die Abtastzeit PID_3Step überwacht. Wenn PID_3Step nicht innerhalb der Abtastzeit ausgeführt werden kann, wird der Fehler 0800h ausgegeben und die Betriebsart gewechselt. In welche Betriebsart gewechselt wird, hängt ab von ActivateRecoverMode und ErrorBehaviour. Wenn EnMonitoring = FALSE, wird die Abtastzeit PID_3Step nicht überwacht, der Fehler 0800h nicht ausgegeben und die Betriebsart nicht gewechselt.
CycleTime.Value(1)	REAL	0.1	Abtastzeit PID_3Step in Sekunden CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Gespeicherter Wert von Retain.CtrlParams.SetByUser Werte aus der Struktur CtrlParamsBackUp können mit LoadBackUp = TRUE wieder geladen werden.
CtrlParamsBackUp.Gain	REAL	1.0	Gespeicherte Proportionalverstärkung
CtrlParamsBackUp.Ti	REAL	20.0	Gespeicherte Integrationszeit in Sekunden
CtrlParamsBackUp.Td	REAL	0.0	Gespeicherte Differenzierzeit in Sekunden
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	Gespeicherter Koeffizient für den Differenzierverzug
CtrlParamsBackUp.PWeighting	REAL	1.0	Gespeicherte Gewichtung des P-Anteils
CtrlParamsBackUp.DWeighting	REAL	1.0	Gespeicherte Gewichtung des D-Anteils
CtrlParamsBackUp.Cycle	REAL	1.0	Gespeicherte Abtastzeit PID-Algorithmus in Sekunden
CtrlParamsBackUp.InputDeadBand	REAL	0.0	Gespeicherte Totzonenbreite der Regeldifferenz
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRule eingestellt ist. CalculateParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.SUT.TuneRule	INT	1	Parameter während Erstoptimierung berechnen nach Methode: • SUT.TuneRule = 0: PID schnell I • SUT.TuneRule = 1: PID langsam I • SUT.TuneRule = 2: Chien, Hrones, Reswick PID • SUT.TuneRule = 3: Chien, Hrones, Reswick PI • SUT.TuneRule = 4: PID schnell II • SUT.TuneRule = 5: PID langsam II

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.SUT.State	INT	0	Die Variable SUT.State zeigt die aktuelle Phase der Erstoptimierung: • State = 0: Erstoptimierung initialisieren • State = 50: Startposition ohne Stellungsrückmeldung ermitteln • State = 100: Standardabweichung berechnen • State = 200: Wendepunkt ermitteln • State = 300: Anregelzeit ermitteln • State = 9900: Erstoptimierung erfolgreich • State = 1: Erstoptimierung nicht erfolgreich
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Mit der Variable RunIn können Sie festlegen, dass eine Nachoptimierung auch ohne Erstoptimierung durchgeführt wird. • RunIn = FALSE Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstoptimierung gestartet. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_3Step in die Betriebsart, aus der die Optimierung gestartet wurde. • RunIn = TRUE Die Erstoptimierung wird übersprungen. PID_3Step versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Erst dann startet die Nachoptimierung. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.
PIDSelfTune.TIR.CalculateParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRule eingestellt ist. CalculateParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.TIR.TuneRule	INT	0	Parameter während Nachoptimierung berechnen nach Methode: • TIR.TuneRule = 0: PID automatisch • TIR.TuneRule = 1: PID schnell • TIR.TuneRule = 2: PID langsam • TIR.TuneRule = 3: Ziegler-Nichols PID • TIR.TuneRule = 4: Ziegler-Nichols PI • TIR.TuneRule = 5: Ziegler-Nichols P

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.TIR.State	INT	0	Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung": • State = -100 Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt. • State = 0: Nachoptimierung initialisieren • State = 200: Standardabweichung berechnen • State = 300: Versuchen Sollwert mit dem maximalen oder minimalen Ausgangswert zu erreichen • State = 400: Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war) • State = 500: Oszillation ermitteln und Parameter berechnen • State = 9900: Nachoptimierung erfolgreich • State = 1: Nachoptimierung nicht erfolgreich
Retain.TransitTime(1)	REAL	30.0	Motorstellzeit in Sekunden Zeit in Sekunden, die der Stellantrieb benötigt, um das Ventil vom geschlossenen in den geöffneten Zustand zu bewegen. TransitTime ist remanent.
Retain.CtrlParams.SetByUser(1)	BOOL	FALSE	Wenn SetByUser = FALSE, sind die PID-Parameter automatisch ermittelt und PID_3Step arbeitet mit einer Totzone am Ausgangswert. Die Totzonenbreite wird während der Optimierung anhand der Standardabweichung des Ausgangswerts berechnet und in Retain.CtrlParams.OutputDeadBand gespeichert. Wenn SetByUser = TRUE, sind die PID-Parameter manuell eingegeben und PID_3 Step arbeitet ohne Totzone am Ausgangswert. Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser ist remanent.
Retain.CtrlParams.Gain(1)	REAL	1.0	Aktive Proportionalverstärkung Verwenden Sie zum Invertieren des Regelsinns die Variable Config.InvertControl. Negative Werte an Gain invertieren den Regelsinn ebenfalls. Es wird empfohlen den Regelsinn nur über InvertControl einzustellen. Wenn InvertControl = TRUE und Gain < 0.0 sind, ist der Regelsinn auch invertiert. Gain ist remanent.
Retain.CtrlParams.Ti(1)	REAL	20.0	• Ti > 0.0: Aktive Integrationszeit in Sekunden • Ti = 0.0: I-Anteil ist ausgeschaltet Ti ist remanent.
Retain.CtrlParams.Td(1)	REAL	0.0	• Td > 0.0: Aktive Differenzierzeit in Sekunden • Td = 0.0: D-Anteil ist ausgeschaltet Td ist remanent.

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.CtrlParams.TdFiltRatio(1)	REAL	0.2	Aktiver Koeffizient für den Differenzierverzug Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierverzug verzögert. Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug • 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam. • 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt. • > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert. TdFiltRatio ist remanent.
Retain.CtrlParams.PWeighting(1)	REAL	1.0	Aktive Gewichtung des P-Anteils Sie können bei Sollwertänderungen den P-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: P-Anteil bei Sollwertänderung voll wirksam • 0.0: P-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam. PWeighting ist remanent.
Retain.CtrlParams.DWeighting(1)	REAL	1.0	Aktive Gewichtung des D- Anteils Sie können bei Sollwertänderungen den D-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: D-Anteil bei Sollwertänderung voll wirksam • 0.0: D-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam. DWeighting ist remanent.
Retain.CtrlParams.Cycle(1)	REAL	1.0	Aktive Abtastzeit PID-Algorithmus in Sekunden, die zu einem ganzzahligen Vielfachen der Zykluszeit des aufrufenden OB aufgerundet wird. Cycle ist remanent.
Retain.CtrlParams.InputDeadBand(1)	REAL	0.0	Totzonenbreite der Regeldifferenz InputDeadBand ist remanent.

Siehe auch

Parameter State und Mode V2 (Seite 2983)

Variable ActivateRecoverMode V2 (Seite 2992)

Parameter State und Mode V2

Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Mit einer steigenden Flanke an ModeActivate wechselt PID_3Step in die Betriebsart, die am Durchgangsparameter Mode gespeichert ist.

Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_3Step in der Betriebsart, die an Mode gespeichert ist. Um PID_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

Bedeutung der Werte

State	Beschreibung der Betriebsart
0	Inaktiv Der Regler ist ausgeschaltet und verändert die Position des Ventils nicht mehr.
1	Erstoptimierung Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. Voraussetzungen für die Erstoptimierung: • Die Motorstellzeit ist konfiguriert oder gemessen. • Betriebsart Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3) • ManualEnable = FALSE • Reset = FALSE • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden. Nach erfolgreicher Erstoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Erstoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode und ErrorBehaviour. Die Phase der Erstoptimierung wird mit Variable SUT.State angezeigt.

State	Beschreibung der Betriebsart
2	Nachoptimierung Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Vor der Nachoptimierung werden die PID-Parameter gesichert. Sie können mit LoadBackUp reaktiviert werden. Voraussetzungen für die Nachoptimierung: • Die Motorstellzeit ist konfiguriert oder gemessen. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. • ManualEnable = FALSE • Reset = FALSE • Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4) Die Nachoptimierung verläuft folgendermaßen beim Start aus: • Automatikbetrieb (State = 3) Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb. PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. • Inaktiv (State = 0) oder Handbetrieb (State = 4) Wenn die Voraussetzungen für eine Erstoptimierung erfüllt sind, wird eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Wenn PIDSelfTune.TIR.RunIn = TRUE, wird die Erstoptimierung übersprungen und versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet dann automatisch. Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Nachoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode und ErrorBehaviour. Die Phase der Nachoptimierung wird mit Variable TIR.State angezeigt.
3	Automatikbetrieb Im Automatikbetrieb regelt PID_3Step die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt: • Erstoptimierung erfolgreich abgeschlossen • Nachoptimierung erfolgreich abgeschlossen • Ändern des Durchgangsparameters Mode auf den Wert 3 und eine steigende Flanke an ModeActivate. Der Wechsel vom Automatikbetrieb in den Handbetrieb erfolgt nur im Inbetriebnahmeeditor stoßfrei. Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.

State	Beschreibung der Betriebsart
4	Handbetrieb Im Handbetrieb geben Sie manuelle Ausgangswerte an den Parametern Manual_UP und Manual_DN oder ManualValue vor. Ob das Stellglied im Fehlerfall auf den Ausgangswert verfahren werden kann, ist beim Parameter ErrorBits beschrieben. Diese Betriebsart können Sie zusätzlich über ManualEnable = TRUE aktivieren. Es wird empfohlen, die Betriebsarten nur über Mode und ModeActivate zu wechseln. Der Wechsel vom Handbetrieb in den Automatikbetrieb erfolgt stoßfrei. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.
5	Ersatzausgangswert anfahren Diese Betriebsart wird im Fehlerfall aktiviert, wenn Errorbehaviour = TRUE und ActivateRecoverMode = FALSE. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Inaktiv".
6	Stellzeitmessung Es wird die Zeit ermittelt, die der Motor benötigt, um das Ventil aus dem geschlossenen Zustand vollständig zu öffnen. Diese Betriebsart wird aktiviert, wenn Mode = 6 und ModeActivate = TRUE gesetzt wird. Wenn zur Stellzeitmessung Anschlagsignale verwendet werden, wird das Ventil von der aktuellen Position aus vollständig geöffnet, vollständig geschlossen und erneut vollständig geöffnet. Wenn GetTransitTime.InvertDirection = TRUE, wird dieses Verhalten umgekehrt. Wenn zur Stellzeitmessung eine Stellungsrückmeldung verwendet wird, wird das Stellglied von der aktuellen Stellung in eine Zielstellung gefahren. Die Ausgangswertgrenzen werden während der Stellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.
7	Fehlerüberwachung Der Regelalgorithmus ist ausgeschaltet und verändert die Position des Ventils nicht mehr. Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert. Alle folgenden Bedingungen müssen erfüllt sein: • Automatikbetrieb (Mode = 3) • Errorbehaviour = FALSE • ActivateRecoverMode = TRUE • Einer oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode (Seite 2992) wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.
8	Ersatzausgangswert anfahren mit Fehlerüberwachung Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Ersatzausgangswert anfahren" aktiviert. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Fehlerüberwachung". Alle folgenden Bedingungen müssen erfüllt sein: • Automatikbetrieb (Mode = 3) • Errorbehaviour = TRUE • ActivateRecoverMode = TRUE • Ein oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode (Seite 2992) wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.
10	Handbetrieb ohne Anschlagsignale Die Anschlagsignale werden **nicht** berücksichtigt, obwohl Config.ActuatorEndStopOn = TRUE ist. Die Ausgangswertgrenzen werden nicht berücksichtigt. Sonst verhält sich PID_3Step genauso wie im Handbetrieb.

ENO-Verhalten

Wenn State = 0 ist, dann ist ENO = FALSE.

Wenn State ≠ 0 ist, dann ist ENO = TRUE.

Automatische Betriebsartwechsel während der Inbetriebnahme

Nach erfolgreicher Erst- oder Nachoptimierung wird der Automatikbetrieb aktiviert. Die folgende Tabelle zeigt, wie sich Mode und State während einer erfolgreichen Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet
n	4	1	Erstoptimierung erfolgreich beendet
n	3	3	Automatikbetrieb wird gestartet

Im Fehlerfall wechselt PID_3Step automatisch die Betriebsart. Die folgende Tabelle zeigt, wie sich Mode und State während einer fehlerhaften Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet
n	4	1	Erstoptimierung abgebrochen
n	4	4	Handbetrieb wird gestartet

Wenn ActivateRecoverMode = TRUE ist, wird die Betriebsart aktiviert, die an Mode gespeichert ist. Beim Start der Stellzeitmessung, Erst- oder Nachoptimierung hat PID_3Step den Wert von State am Durchgangsparameter Mode gespeichert. PID_3Step wechselt also in die Betriebsart, aus der die Stellzeitmessung oder Optimierung gestartet wurde.

Wenn ActivateRecoverMode = FALSE ist, wird die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren" aktiviert.

Automatischer Betriebsartwechsel nach der Stellzeitmessung

Wenn ActivateRecoverMode = TRUE ist, wird nach erfolgreicher Stellzeitmessung die Betriebsart aktiviert, die an Mode gespeichert ist.

Wenn ActivateRecoverMode = FALSE ist, wird nach erfolgreicher Stellzeitmessung in die Betriebsart "Inaktiv" gewechselt.

Automatische Betriebsartwechsel im Automatikbetrieb

Im Fehlerfall wechselt PID_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.

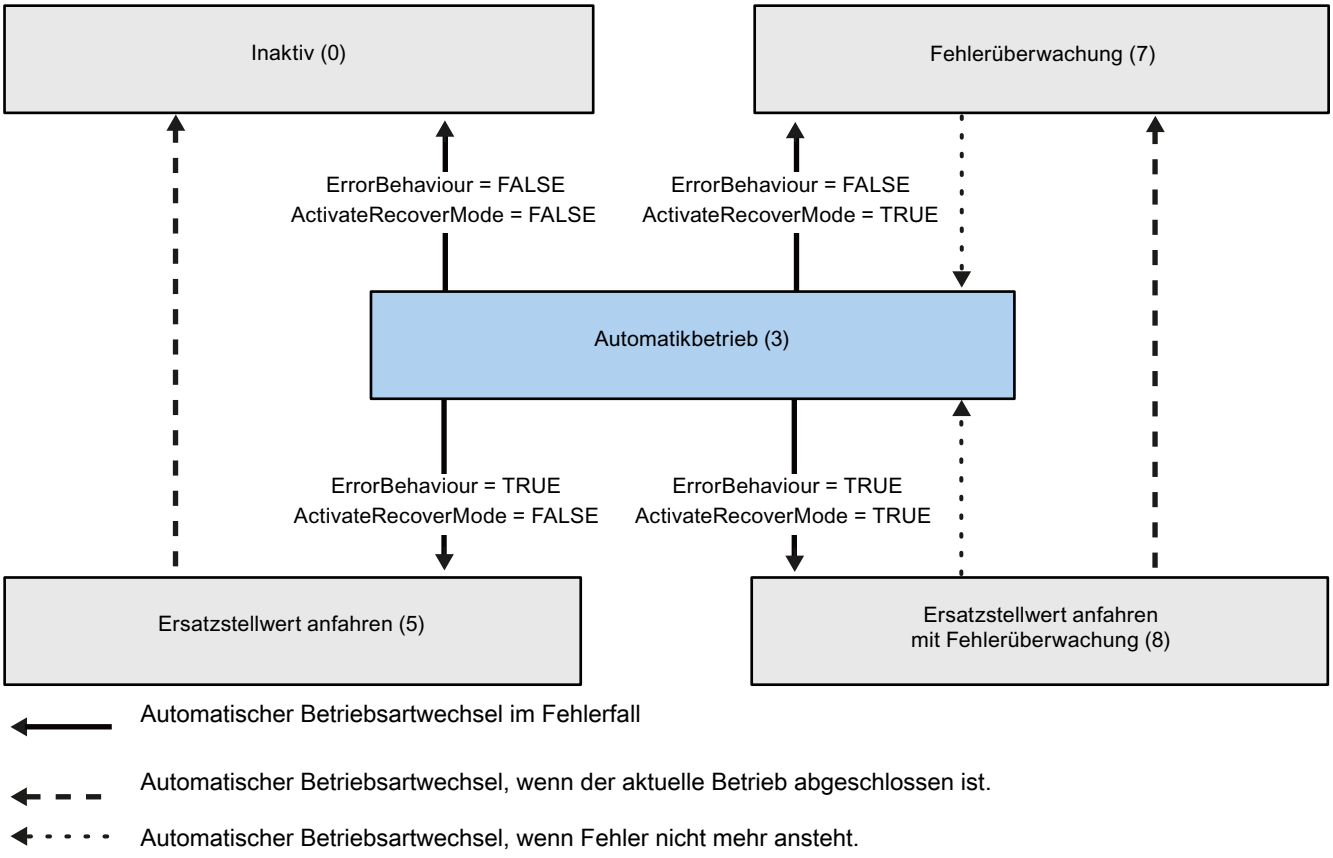

Siehe auch

Variable ActivateRecoverMode V2 (Seite 2992)

Parameter ErrorBits V2 (Seite 2988)

Parameter ErrorBits V2

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der ErrorBits binär addiert angezeigt. Wird z. B. ErrorBits = 0003h angezeigt, so stehen gleichzeitig die Fehler 0001h und 0002h an.

Wenn eine Stellungsrückmeldung vorhanden ist, verwendet PID_3Step im Handbetrieb ManualValue als Ausgangswert. Ausnahme ist Errorbits = 10000h.

ErrorBits (DW#16#...)	Beschreibung
0000	Es steht kein Fehler an.
0001	Der Parameter "Input" ist außerhalb der Istwertgrenzen. • Input > Config.InputUpperLimit oder • Input < Config.InputLowerLimit Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_3Step im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
0002	Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
0004	Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0010	Der Sollwert wurde während der Optimierung verändert. An der Variable CancelTuningLevel können Sie die zulässige Schwankung des Sollwerts einstellen. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0020	Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_3Step in der Betriebsart Nachoptimierung.
0080	Fehler bei der Erstoptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert oder der Istwert reagiert nicht wie erwartet. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen. Stellen Sie außerdem sicher, dass der Istwert vor Start der Erstoptimierung nicht stark schwingt. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0100	Fehler während der Nachoptimierung führte zu ungültigen Parametern. Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.

ErrorBits (DW#16#...)	Beschreibung
0200	Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
0400	Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
0800	Abtastzeitfehler: PID_3Step wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_3Step im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist. Falls dieser Fehler bei der Simulation mit PLCSIM aufgetreten ist, beachten Sie die Hinweise unter AUTOHOTSPOT.
1000	Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
2000	Ungültiger Wert am Parameter Feedback_PER. Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Im Handbetrieb können Sie die Position des Stellglieds nur mit Manual_UP und Manual_DN verändern, nicht mit ManualValue. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.

ErrorBits (DW#16#...)	Beschreibung
4000	Ungültiger Wert am Parameter Feedback. Wert hat kein gültiges Zahlenformat. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Im Handbetrieb können Sie die Position des Stellglieds nur mit Manual_UP und Manual_DN verändern, nicht mit ManualValue. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
8000	Fehler bei der digitalen Stellungsrückmeldung. Actuator_H = TRUE und Actuator_L = TRUE. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die "Anschlagsignale Stellglied" deaktivieren (Config.ActuatorEndStopOn = FALSE) oder in die Betriebsart "Handbetrieb ohne Anschlagsignale" (Mode = 10) wechseln. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.
10000	Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat. Das Stellglied kann nicht auf den Handwert verfahren werden und bleibt in der aktuellen Position stehen. Geben Sie einen gültigen Wert an ManualValue vor oder verfahren Sie das Stellglied im Handbetrieb mit Manual_UP und Manual_DN.
20000	Ungültiger Wert an der Variablen SavePosition. Wert hat kein gültiges Zahlenformat. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen.
40000	Ungültiger Wert am Parameter Disturbance. Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, wird Disturbance auf Null gesetzt. PID_3Step bleibt im Automatikbetrieb. Wenn vor Auftreten des Fehlers die Betriebsart Erst- oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist. Wenn Disturbance in der aktuellen Phase keinen Einfluss auf den Ausgangswert hat, wird die Optimierung nicht abgebrochen. Während der Stellzeitmessung hat der Fehler keinen Einfluss.

Variable ActivateRecoverMode V2

Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall. Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn ActivateRecoverMode = TRUE ist, bleibt PID_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

Automatikbetrieb

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.
TRUE	Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert, da PID_3Step bei jedem Fehlerfall zwischen berechnetem Ausgangswert und dem Ersatzausgangswert wechselt. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_3Step im Automatikbetrieb: • 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen. • 0800h: Abtastzeitfehler • 40000h: Ungültiger Wert am Parameter Disturbance. Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung": • 0002h: Ungültiger Wert am Parameter Input_PER. • 0200h: Ungültiger Wert am Parameter Input. • 0400h: Berechnung des Ausgangswerts fehlgeschlagen. • 1000h: Ungültiger Wert am Parameter Setpoint. Wenn einer oder mehrere der folgenden Fehler auftreten, kann PID_3Step das Stellglied nicht mehr bewegen: • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung. • 20000h: Ungültiger Wert an der Variablen SavePosition. Wert hat kein gültiges Zahlenformat. Dieses Verhalten ist unabhängig von ErrorBehaviour. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.

Erstoptimierung, Nachoptimierung und Stellzeitmessung

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert. Nach erfolgreicher Stellzeitmessung wechselt der Regler in die Betriebsart Inaktiv.
TRUE	Wenn der folgende Fehler auftritt, bleibt PID_3Step in der aktiven Betriebsart: • 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Die folgenden Fehler werden ignoriert: • 10000h: Ungültiger Wert am Parameter ManualValue. • 20000h: Ungültiger Wert an der Variable SavePosition. Bei allen anderen Fehlern bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.

Handbetrieb

Im Handbetrieb wirkt ActivateRecoverMode nicht.

Siehe auch

Statische Variablen PID_3Step V2 (Seite 2974)

Parameter State und Mode V2 (Seite 2983)

Variable Warning V2

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Warnungen binär addiert angezeigt. Wird z. B. die Warnung 0005h angezeigt, so stehen gleichzeitig die Warnungen 0001h und 0004h an.

Warning (DW#16#...)	Beschreibung
0000	Es steht keine Warnung an.
0001	Während der Erstoptimierung wurde der Wendepunkt nicht gefunden.
0004	Der Sollwert wurde begrenzt auf die eingestellten Grenzen.
0008	Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TIR.TuneRule = 3 berechnet.
0010	Die Betriebsart konnte nicht geändert werden, da Reset = TRUE oder ManualEnable = TRUE.
0020	Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.
0040	Der Istwert hat eine seiner Warngrenzen überschritten.
0080	Ungültiger Wert an Mode. Die Betriebsart wird nicht gewechselt.
0100	Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.
0200	Die angegebene Regel zur Optimierung wird nicht unterstützt. Es werden keine PID-Parameter berechnet.

Warning (DW#16#...)	Beschreibung
0400	Die Stellzeit kann nicht gemessen werden, da die Einstellungen des Stellglieds nicht zur gewählten Messmethode passen.
0800	Bei der Stellzeitmessung ist die Differenz zwischen der aktuellen Position und dem neuen Ausgangswert zu klein. Dies kann zu falschen Ergebnissen führen. Die Differenz zwischen aktuellem Ausgangswert und neuem Ausgangswert muss mindestens 50% des gesamten Stellbereichs betragen.
1000	Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.
2000	Das Stellglied wurde länger als Config.VirtualActuatorLimit × Retain.TransitTime in eine Richtung bewegt. Kontrollieren Sie, ob das Stellglied ein Anschlagsignal erreicht hat.

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h
- 2000h

Alle anderen Warnungen werden bei steigender Flanke an Reset oder ErrorAck gelöscht.

PID_3Step V1

Beschreibung PID_3Step V1

Beschreibung

Mit der Anweisung PID_3Step konfigurieren Sie einen PID-Regler mit Selbstoptimierung für Ventile oder Stellglieder mit integrierendem Verhalten.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert anfahren
- Stellzeitmessung
- Ersatzausgangswert anfahren mit Fehlerüberwachung
- Fehlerüberwachung

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

PID-Algorithmus

PID_3Step ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der Ausgangswert wird anhand folgender Formel berechnet.

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Δy	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient für den Differenzierverzug ($T1 = a \times T_D$)
T_D	Differenzierzeit
c	Gewichtung des D-Anteils

Blockschaltbild ohne Stellungsrückmeldung

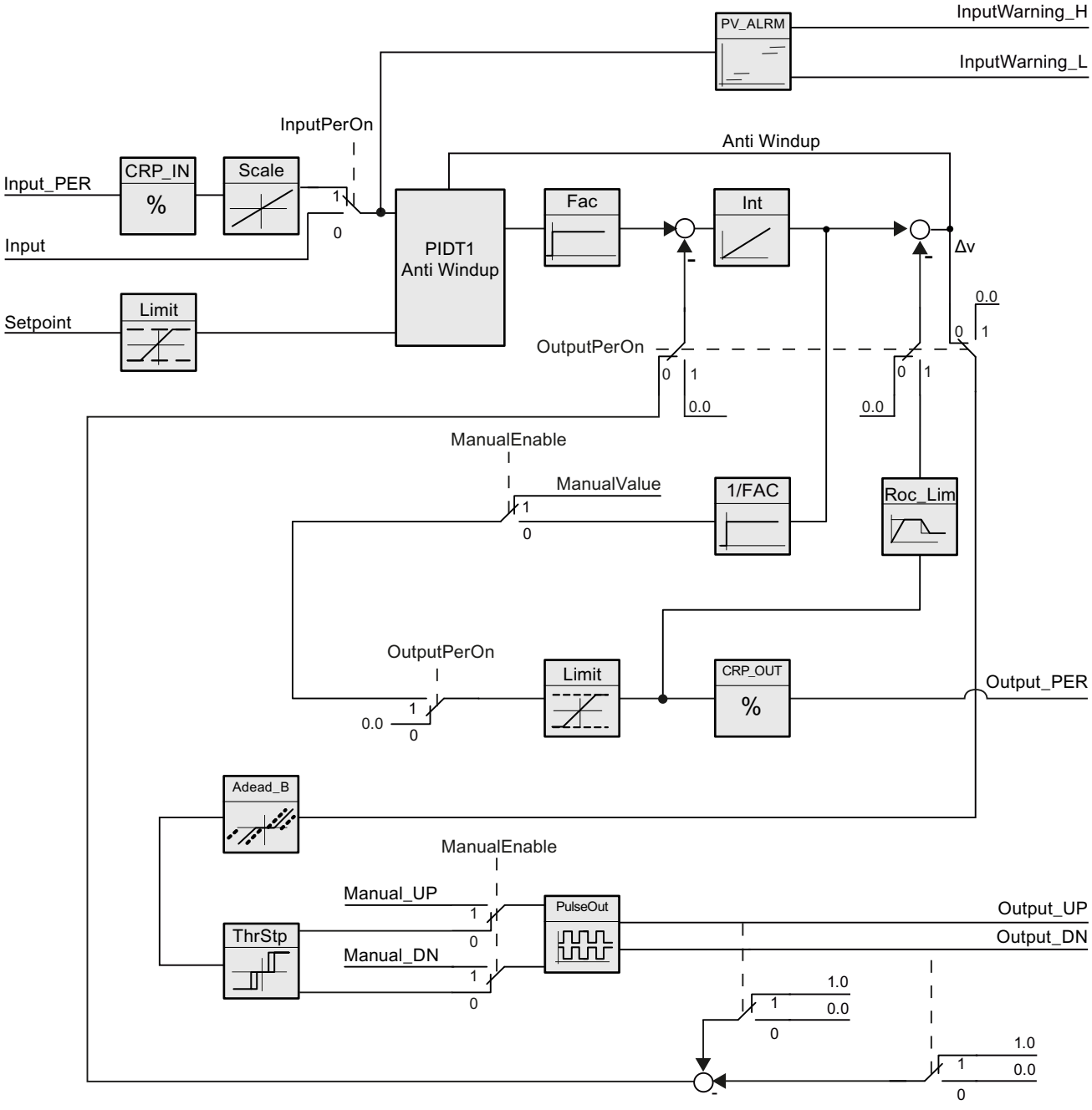

Blockschaltbild mit Stellungsrückmeldung

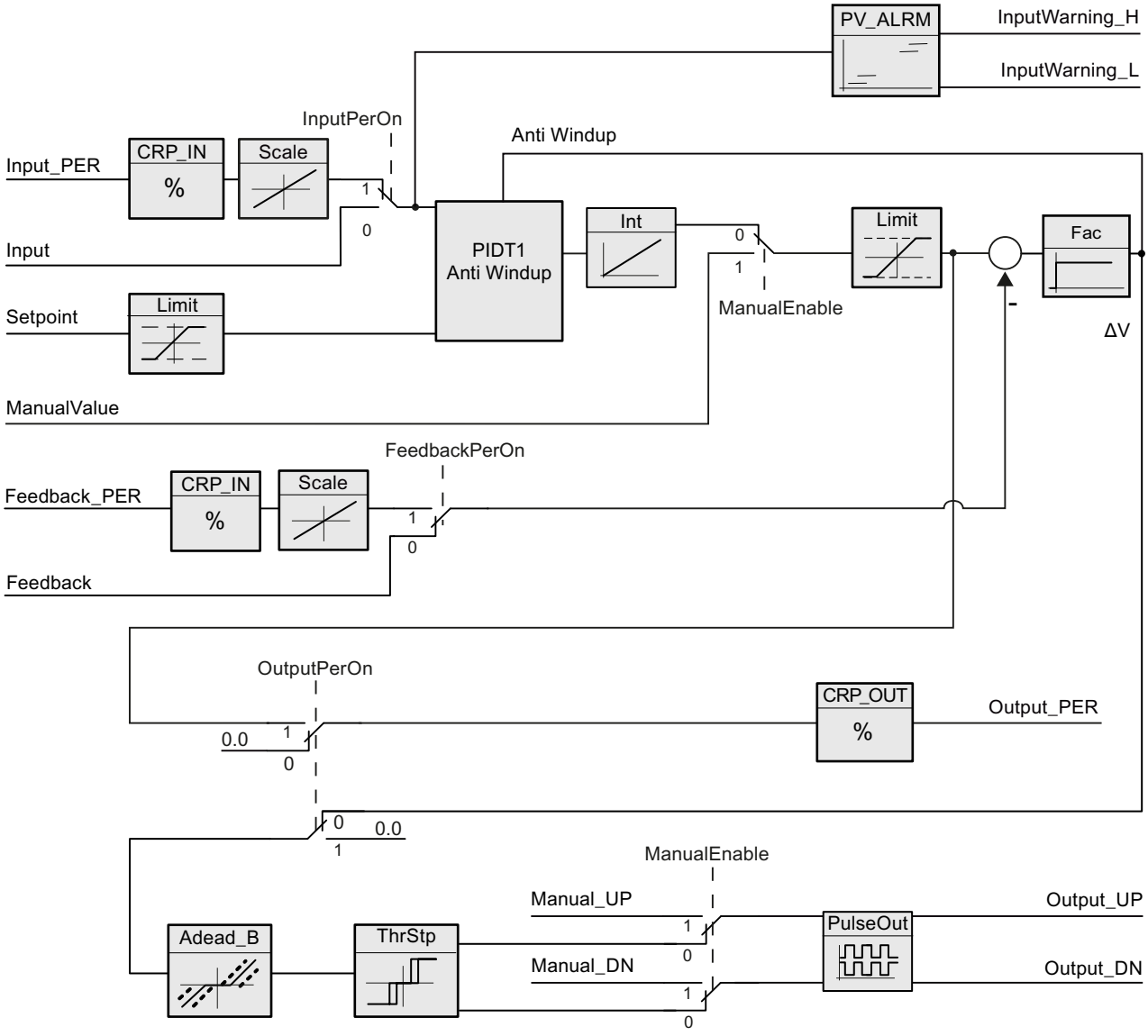

Blockschaltbild PIDT1 mit Anti-Windup

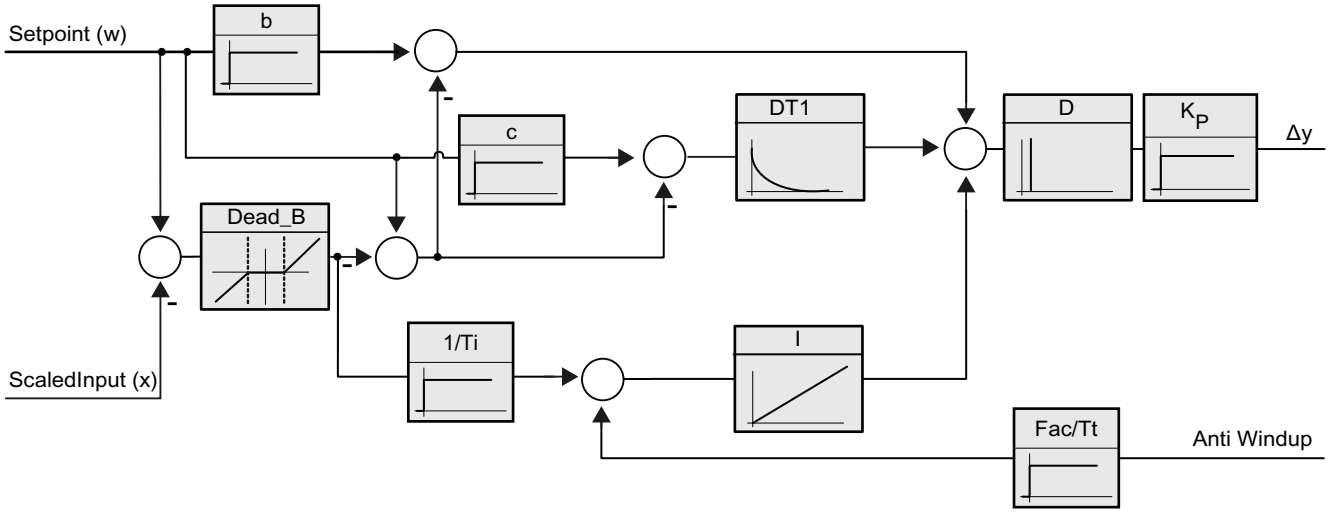

Aufruf

PID_3Step wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID_3Step vollständig laden.

AUTOHOTSPOT

Anlauf

Bei Anlauf der CPU startet PID_3Step in der zuletzt aktiven Betriebsart. Um PID_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

Verhalten im Fehlerfall

Wenn Fehler auftreten, werden diese am Parameter Error ausgegeben. Das Verhalten von PID_3Step konfigurieren Sie über die Variablen ErrorBehaviour und ActivateRecoverMode.

ErrorBe-haviour	ActivateReco-verMode	Konfiguration Stellgliedein-stellung Output setzen auf	Verhalten
0	FALSE	Aktuellen Ausgangswert	Wechsel in Betriebsart "Inaktiv" (Mo-de = 0)
0	TRUE	Aktuellen Ausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Fehlerüberwa-chung" (Mode = 7)

ErrorBe-haviour	ActivateReco-verMode	Konfiguration Stellgliedein-stellung Output setzen auf	Verhalten
1	FALSE	Ersatzausgangswert	Wechsel in Betriebsart "Ersatzausgangs-wert anfahren" (Mode = 5) Wechsel in Betriebsart "Inaktiv" (Mo-de = 0)
1	TRUE	Ersatzausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangs-wert anfahren mit Fehlerüberwachung" (Mode = 8) Wechsel in Betriebsart "Fehlerüberwa-chung" (Mode = 7)

Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

Siehe auch

Parameter State und Retain.Mode V1 (Seite 3013)

Parameter ErrorBits V1 (Seite 3021)

Arbeitsweise PID_3Step V1

Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001hex).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warnings = 0040hex) und der Ausgangsparameter InputWarning_H oder InputWarning_L wird TRUE.

Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID_3Step begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID_3Step prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit_H oder SetpointLimit_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

- Oberer Anschlag: Config.FeedbackScaling.UpperPointOut
- Unter Anschlag: Config.FeedbackScaling.LowerPointOut

Es muss gelten:

UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
FALSE	TRUE	FALSE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
FALSE	TRUE	TRUE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
TRUE	FALSE	FALSE	nicht einstellbar (100.0 %)	nicht einstellbar (100.0 %)
TRUE	TRUE	FALSE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
TRUE	TRUE	TRUE	-100.0 % oder 0.0 %	0.0 % oder +100.0 %

Wenn OutputPerOn = FALSE und FeedbackOn = FALSE, können Sie den Ausgangswert nicht begrenzen. Die Digitalausgänge entweder bei Actuator_H = TRUE oder Actuator_L = TRUE oder nach einer Verfahrzeit von 110% der Motorstellzeit zurück gesetzt.

Der Ausgangswert beträgt 27648 bei 100 % und -27648 bei -100 %. PID_3Step muss das Ventil vollständig schließen können. Daher muss die Null in den Ausgangswertgrenzen enthalten sein.

Hinweis

Einsatz mit zwei oder mehr Stellgliedern

PID_3 Step ist nicht für den Einsatz mit zwei oder mehr Stellgliedern (z. B. in Heizen-/Kühlen-Anwendungen) geeignet, da unterschiedliche Stellglieder auch unterschiedliche PID-Parameter benötigen, um ein gutes Regelverhalten zu erzielen.

Ersatzausgangswert

PID_3Step kann im Fehlerfall einen Ersatzausgangswert ausgeben und das Stellglied in eine sichere Position stellen, die Sie an der Variablen SavePosition vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden auf Gültigkeit überwacht:

- Setpoint
- Input
- Input_PER
- Feedback
- Feedback_PER
- Output

Abtastzeit PID_3Step überwachen

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID_3Step misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (ErrorBits = 0800 hex).

Folgende Bedingungen versetzen PID_3Step während der Optimierung in die Betriebsart "Inaktiv":

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert

Folgende Bedingungen versetzen PID_3Step bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert >= 1,5 x alter Mittelwert
- Neuer Mittelwert <= 0,5 x alter Mittelwert

Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_3Step werden bei jedem Aufruf durchgeführt.

Motorstellzeit messen

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal 110% der Motorstellzeit in eine Richtung bewegt. PID_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren. Die Motorstellzeit können Sie während der Inbetriebnahme messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der

Selbstoptimierung und dem Anti-Windup-Verhalten für eine korrekte Arbeitsweise notwendig. Konfigurieren Sie deshalb die Motorstellzeit mit dem Wert, den der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen.

Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID_3Step arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

Eingangsparameter PID_3Step V1

Tabelle 4-67

Parameter	Datentyp	Vorbelegung	Beschreibung
Setpoint	REAL	0.0	Sollwert des PID-Reglers im Automatikbetrieb
Input	REAL	0.0	Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.
Input_PER	WORD	W#16#0	Ein Analogeingang wird als Quelle des Istwerts verwendet. Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.
Actuator_H	BOOL	FALSE	Digitale Stellungsrückmeldung des Ventils für den Oberen Anschlag Wenn Actuator_H = TRUE ist die Stellung des Ventils am oberen Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.
Actuator_L	BOOL	FALSE	Digitale Stellungsrückmeldung des Ventils für den Unteren Anschlag Wenn Actuator_L = TRUE ist die Stellung des Ventils am unteren Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.
Feedback	REAL	0.0	Stellungsrückmeldung des Ventils Wenn Sie den Parameter Feedback verwenden, muss Config.FeedbackPerOn = FALSE sein.

Parameter	Datentyp	Vorbelegung	Beschreibung
Feedback_PER	WORD	W#16#0	Analoge Stellungsrückmeldung eines Ventils Wenn Sie den Parameter Feedback_PER verwenden, muss Config.FeedbackPerOn = TRUE sein. Feedback_PER wird skaliert anhand der Variablen: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut
ManualEnable	BOOL	FALSE	• Flanke FALSE -> TRUE wählt die Betriebsart "Handbetrieb", State = 4, Retain.Mode bleibt unverändert. • Flanke TRUE -> FALSE wählt die zuletzt aktive Betriebsart Während ManualEnable = TRUE wirkt sich eine Änderung von Retain.Mode nicht aus. Erst bei der Flanke TRUE -> FALSE an ManualEnable wird die Änderung von Retain.Mode berücksichtigt. **PID_3Step V1.1**Wenn beim Start der CPU ManualEnable = TRUE, startet PID_3Step im Handbetrieb. Eine steigende Flanke (FALSE > TRUE) an ManualEnable ist nicht notwendig. **PID_3Step V1.0** Beim Start der CPU schaltet PID_3Step nur bei einer steigenden Flanke (FALSE->TRUE) an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID_3Step in der letzten Betriebsart bei der ManualEnable FALSE war.
ManualValue	REAL	0.0	Im Handbetrieb wird die absolute Position des Ventils vorgegeben. ManualValue wird nur ausgewertet, wenn Sie OutputPer verwenden **oder** eine Stellungsrückmeldung verfügbar ist.
Manual_UP	BOOL	FALSE	Im Handbetrieb öffnet jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. Manual_UP wird nur ausgewertet, wenn Sie weder Output_PER verwenden noch eine Stellungsrückmeldung verfügbar ist.
Manual_DN	BOOL	FALSE	Im Handbetrieb schließt jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. Manual_DN wird nur ausgewertet, wenn Sie weder Output_PER verwenden noch eine Stellungsrückmeldung verfügbar ist.
Reset	BOOL	FALSE	Führt einen Neustart des Reglers durch. • Flanke FALSE -> TRUE – Wechsel in Betriebsart "Inaktiv" – ErrorBits und Warning werden zurückgesetzt – Zwischenwerte der Regelung werden zurückgesetzt (PID-Parameter bleiben erhalten) • Flanke TRUE -> FALSE – Wechsel in letzte aktive Betriebsart – Falls zuvor Automatikbetrieb aktiv war, erfolgt die Umschaltung in den Automatikbetrieb stoßfrei.

Ausgangsparameter PID_3Step V1

Tabelle 4-68

Parameter	Datentyp	Vorbelegung	Beschreibung
ScaledInput	REAL	0.0	Skalierter Istwert
ScaledFeedback	REAL	0.0	Skalierte Stellungsrückmeldung Bei einem Stellglied ohne Stellungsrückmeldung zeigt ScaledFeedback die Position des Stellglieds sehr ungenau an. Dann darf ScaledFeedback nur zur groben Abschätzung der aktuellen Position genutzt werden.
Output_UP	BOOL	FALSE	Digitaler Ausgangswert zum Öffnen des Ventils Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_UP verwendet.
Output_DN	BOOL	FALSE	Digitaler Ausgangswert zum Schließen des Ventils Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_DN verwendet.
Output_PER	WORD	W#16#0	Analoger Ausgangswert Wenn Config.OutputPerOn = TRUE, wird Output_PER verwendet. Verwenden Sie Output_PER, falls Sie als Stellglied ein Ventil nutzen, welches über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert wird, z. B. 0...10V oder 4...20mA. Der Wert an Output_PER entspricht der Zielstellung des Ventils z. B. Output_PER = 13824, wenn das Ventil zu 50% geöffnet werden soll.
SetpointLimit_H	BOOL	FALSE	Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Obergrenze des Sollwerts begrenzt. Als Obergrenze Sollwert wird die konfigurierte absolute Obergrenze Istwert vorbelegt. Wenn Sie Config.SetpointUpperLimit auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.
SetpointLimit_L	BOOL	FALSE	Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Untergrenze des Sollwerts begrenzt. Als Untergrenze Sollwert ist die konfigurierte absolute Untergrenze Istwert vorbelegt. Wenn Sie Config.SetpointLowerLimit auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
InputWarning_H	BOOL	FALSE	Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.
InputWarning_L	BOOL	FALSE	Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.

Parameter	Datentyp	Vorbelegung	Beschreibung
State	INT	0	Der Parameter State (Seite 3013) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit der Variable Retain.Mode. • State = 0: Inaktiv • State = 1: Erstoptimierung • State = 2: Nachoptimierung • State = 3: Automatikbetrieb • State = 4: Handbetrieb • State = 5: Ersatzausgangswert anfahren • State = 6: Stellzeitmessung • State = 7: Fehlerüberwachung • State = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung
Error	BOOL	FALSE	Wenn Error = TRUE, liegt mindestens eine Fehlermeldungen vor.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 3021) zeigt die Fehlermeldungen.

Siehe auch

Parameter State und Retain.Mode V1 (Seite 3013)

Parameter ErrorBits V1 (Seite 3021)

Statische Variablen PID_3Step V1

Hinweis

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Verändern Sie die mit (1) gekennzeichneten Variablen nur in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden. Die Betriebsart "Inaktiv" erzwingen Sie durch den Wert "0" der Variable "Retain.Mode".

Tabelle 4-69

Variable	Datentyp	Vorbelegung	Beschreibung
ActivateRecoverMode	BOOL	TRUE	Die Variable ActivateRecoverMode (Seite 3023) bestimmt das Verhalten im Fehlerfall.
RunModeByStartup	BOOL	TRUE	Nach CPU Neustart letzte Betriebsart aktivieren Wenn RunModeByStartup = TRUE, kehrt der Regler nach CPU-Anlauf in die zuletzt aktive Betriebsart zurück. Wenn RunModeByStartup = FALSE, bleibt der Regler nach CPU-Anlauf inaktiv.

Variable	Datentyp	Vorbelegung	Beschreibung
PhysicalUnit	INT	0	Physikalische Einheit des Ist- und Sollwerts, z. B. ºC oder ºF.
PhysicalQuantity	INT	0	Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur
ErrorBehaviour	INT	0	Wenn ErrorBehaviour = 0, bleibt das Ventil im Fehlerfall in der aktuellen Position und der Regler wechselt direkt in die Betriebsart "Inaktiv" oder "Fehlerüberwachung". Wenn ErrorBehaviour = 1, wird im Fehlerfall das Stellglied auf den Ersatzausgangswert gefahren und erst dann in die Betriebsart "Inaktiv" oder "Fehlerüberwachung" gewechselt. Wenn die folgenden Fehler auftreten, kann das Ventil nicht mehr auf einen konfigurierten Ersatzausgangswert gefahren werden. • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung.
Warning	DWORD	DW#16#0	Die Variable Warning (Seite 3013) zeigt die Warnungen seit Reset oder dem letzten Wechsel der Betriebsart. Zyklische Warnungen (z. B. Istwertwarnung) werden solange angezeigt wie die Ursache der Warnung anliegt. Fällt der Grund Weg, werden sie automatisch gelöscht. Nichtzyklische Warnungen (z. B. Wendepunkt nicht gefunden) bleiben bestehen und werden wie Fehler gelöscht.
SavePosition	REAL	0.0	Ersatzausgangswert Wenn ErrorBehaviour = 1, wird das Stellglied im Fehlerfall in eine für die Anlage sichere Position gefahren und erst dann in die Betriebsart "Inaktiv" gewechselt.
CurrentSetpoint	REAL	0.0	Aktuell aktiver Sollwert. Dieser Wert wird beim Start der Optimierung eingefroren.
Progress	REAL	0.0	Fortschritt der Optimierung in Prozent (0.0 - 100.0)
Config.InputPerOn(1)	BOOL	TRUE	Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.
Config.OutputPerOn(1)	BOOL	FALSE	Wenn OutputPerOn = TRUE, wird der Parameter Output_PER verwendet. Wenn OutputPerOn = FALSE, werden die Parameter Ouput_UP und Output_DN verwendet.
Config.LoadBackUp	BOOL	FALSE	Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.
Config.InvertControl(1)	BOOL	FALSE	Invertieren des Regelsinns Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.
Config.FeedbackOn(1)	BOOL	FALSE	Wenn FeedbackOn = FALSE, wird eine Stellungsrückmeldung simuliert. Wenn FeedbackOn = TRUE, wird generell die Stellungsrückmeldung aktiviert.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.FeedbackPerOn(1)	BOOL	FALSE	FeedbackPerOn ist nur wirksam, wenn FeedbackOn = TRUE. Wenn FeedbackPerOn = TRUE, wird der Analogeingang für die Stellungsrückmeldung verwendet (Parameter Feedback_PER). Wenn FeedbackPerOn = FALSE, wird der Parameter Feedback für die Stellungsrückmeldung verwendet.
Config.ActuatorEndStopOn(1)	BOOL	FALSE	Wenn ActuatorEndStopOn = TRUE, wird die digitale Stellungsrückmeldung Actuator_L und Actuator_H berücksichtigt.
Config.InputUpperLimit(1)	REAL	120.0	Obergrenze des Istwerts Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit(1)	REAL	0.0	Untergrenze des Istwerts InputLowerLimit < InputUpperLimit
Config.InputUpperWarning(1)	REAL	+3.402822e+38	Obere Warngrenze des Istwerts Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als Obere Warngrenze verwendet. Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning(1)	REAL	-3.402822e+38	Untere Warngrenze des Istwerts Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als Untere Warngrenze verwendet. Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit
Config.OutputUpperLimit(1)	REAL	100.0	Obergrenze des Ausgangswerts Folgender Wertebereich ist zulässig: UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit Weitere Details siehe OutputLowerLimit

Variable	Datentyp	Vorbelegung	Beschreibung
Config.OutputLowerLimit(1)	REAL	0.0	Untergrenze des Ausgangswerts Folgender Wertebereich ist zulässig: OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut Bei Verwendung von Output_PER entspricht eine Ausgangswertgrenze von -100 % dem Wert Output_PER = -27648; 100 % entsprechen dem Wert Output_PER = 27648. Wenn OutputPerOn = FALSE und FeedbackOn = FALSE, werden OutputLowerLimit und OutputUpperLimit nicht ausgewertet. Output_UP und Output_DN werden dann bei Actuator_H = TRUE oder Actuator_L = TRUE zurück gesetzt (wenn ActuatorEndStopOn = TRUE) oder nach einer Verfahrzeit von 110 % * Config.TransitTime (wenn ActuatorEndStopOn = FALSE).
Config.SetpointUpperLimit(1)	REAL	+3.402822e+38	Obergrenze des Sollwerts Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert vorbelegt. Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.
Config.SetpointLowerLimit(1)	REAL	-3.402822e+38	Untergrenze des Sollwerts Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert vorbelegt. Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.
Config.MinimumOnTime(1)	REAL	0.0	Minimale Einschaltzeit Zeit in Sekunden, die der Stellantrieb mindestens eingeschaltet sein muss. Config.MinimumOnTime ist nur wirksam, falls Output_UP und Output_DN verwendet werden (Config.OutputPerOn = FALSE).
Config.MinimumOffTime(1)	REAL	0.0	Minimale Ausschaltzeit Zeit in Sekunden, die der Stellantrieb mindestens ausgeschaltet sein muss. Config.MinimumOffTime ist nur wirksam, falls Output_UP und Output_DN verwendet werden (Config.OutputPerOn = FALSE).
Config.TransitTime(1)	REAL	30.0	Motorstellzeit Zeit in Sekunden, die der Stellantrieb benötigt, um das Ventil vom geschlossenen in den geöffneten Zustand zu bewegen.
Config.InputScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Input_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointIn(1)	REAL	0.0	Skalierung Input_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.InputScaling.UpperPointOut(1)	REAL	100.0	Skalierter oberer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.InputScaling.LowerPointOut(1)	REAL	0.0	Skalierter unterer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.
Config.FeedbackScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Feedback_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.
Config.FeedbackScaling.LowerPointIn(1)	REAL	0.0	Skalierung Feedback_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.
Config.FeedbackScaling.UpperPointOut(1)	REAL	100.0	Oberer Anschlag Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet. Der zulässige Wertebereich ist abhängig von der Konfiguration. • FeedbackOn = FALSE: UpperPointOut = 100.0 • FeedbackOn = TRUE: UpperPointOut = 100.0 oder 0.0 UpperPointOut ≠ LowerPointOut
Config.FeedbackScaling.LowerPointOut(1)	REAL	0.0	Unterer Anschlag Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet. Der zulässige Wertebereich ist abhängig von der Konfiguration. • FeedbackOn = FALSE: LowerPointOut = 0.0 • FeedbackOn = TRUE: LowerPointOut = 0.0 oder -100.0 LowerPointOut ≠ UpperPointOut
GetTransitTime.InvertDirection	BOOL	FALSE	Wenn InvertDirection = FALSE, wird das Ventil zum Ermitteln der Stellzeit vollständig geöffnet, geschlossen und wieder geöffnet. Wenn InvertDirection = TRUE, wird das Ventil vollständig geschlossen, geöffnet und wieder geschlossen.
GetTransitTime.SelectFeedback	BOOL	FALSE	Wenn SelectFeedback = TRUE, wird Feedback_PER oder Feedback bei der Stellzeitmessung berücksichtigt. Wenn SelectFeedback = FALSE, wird Actuator_H und Actuator_L bei der Stellzeitmessung berücksichtigt.
GetTransitTime.Start	BOOL	FALSE	Wenn Start = TRUE, wird die Stellzeitmessung gestartet.

Variable	Datentyp	Vorbelegung	Beschreibung
GetTransitTime.State	INT	0	Aktuelle Phase der Stellzeitmessung • State = 0: Inaktiv • State = 1: Ventil vollständig öffnen • State = 2: Ventil vollständig schließen • State = 3: Ventil auf Zielstellung (NewOutput) stellen • State = 4: Stellzeitmessung erfolgreich beendet • State = 5: Stellzeitmessung abgebrochen
GetTransitTime.NewOutput	REAL	0.0	Zielstellung für die Stellzeitmessung mit Stellungsrückmeldung Die Zielstellung muss innerhalb "Oberer Anschlag" und "Unterer Anschlag" liegen. Die Differenz zwischen NewOutput und ScaledFeedback muss mindestens 50 % des zulässigen Stellbereichs sein.
CycleTime.StartEstimation	BOOL	TRUE	Wenn StartEstimation = TRUE, wird die Messung der Abtastzeit PID_3Step gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Wenn EnEstimation = TRUE, wird die Abtastzeit PID_3Step berechnet.
CycleTime.EnMonitoring	BOOL	TRUE	Wenn EnMonitoring = TRUE, wird die Abtastzeit PID_3Step überwacht. Wenn PID_3Step nicht innerhalb der Abtastzeit ausgeführt werden kann, wird der Fehler 0800h ausgegeben und die Betriebsart gewechselt. In welche Betriebsart gewechselt wird, hängt ab von ActivateRecoverMode und ErrorBehaviour. Wenn EnMonitoring = FALSE, wird die Abtastzeit PID_3Step nicht überwacht, der Fehler 0800h nicht ausgegeben und die Betriebsart nicht gewechselt.
CycleTime.Value(1)	REAL	0.1	Abtastzeit PID_3Step in Sekunden CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Gespeicherter Wert von Retain.CtrlParams.SetByUser Werte aus der Struktur CtrlParamsBackUp können mit Config.LoadBackUp = TRUE wieder geladen werden.
CtrlParamsBackUp.Gain	REAL	1.0	Gespeicherte Proportionalverstärkung
CtrlParamsBackUp.Ti	REAL	20.0	Gespeicherte Integrationszeit
CtrlParamsBackUp.Td	REAL	0.0	Gespeicherte Differenzierzeit
CtrlParamsBackUp.TdFiltRatio	REAL	0.0	Gespeicherter Koeffizient für den Differenzierverzug
CtrlParamsBackUp.PWeighting	REAL	0.0	Gespeicherte Gewichtung des P-Anteils
CtrlParamsBackUp.DWeighting	REAL	0.0	Gespeicherte Gewichtung des D-Anteils
CtrlParamsBackUp.Cycle	REAL	1.0	Gespeicherte Abtastzeit PID-Algorithmus
CtrlParamsBackUp.InputDeadBand	REAL	0.0	Gespeicherte Totzonenbreite der Regeldifferenz

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.SUT.CalculateSUTParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateSUTParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRuleSUT eingestellt ist. CalculateSUTParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.SUT.TuneRuleSUT	INT	1	Parameter während Erstoptimierung berechnen nach Methode: • TuneRuleSUT = 0: PID schnell I • TuneRuleSUT = 1: PID langsam I • TuneRuleSUT = 2: Chien, Hrones, Reswick PID • TuneRuleSUT = 3: Chien, Hrones, Reswick PI • TuneRuleSUT = 4: PID schnell II • TuneRuleSUT = 5: PID langsam II
PIDSelfTune.SUT.State	INT	0	Die Variable SUT.State zeigt die aktuelle Phase der Erstoptimierung:
PIDSelfTune.TIR.RunIn	BOOL	FALSE	• RunIn = FALSE Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstoptimierung gestartet. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_3Step in die Betriebsart "Inaktiv". • RunIn = TRUE Die Erstoptimierung wird übersprungen. PID_3Step versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Erst dann startet die Nachoptimierung. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.
PIDSelfTune.TIR.CalculateTIRParams	BOOL	FALSE	Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateTIRParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRuleTIR eingestellt ist. CalculateTIRParams wird nach der Berechnung auf FALSE gesetzt.
PIDSelfTune.TIR.TuneRuleTIR	INT	0	Parameter während Nachoptimierung berechnen nach Methode: • TuneRuleTIR = 0: PID automatisch • TuneRuleTIR = 1: PID schnell • TuneRuleTIR = 2: PID langsam • TuneRuleTIR = 3: Ziegler-Nichols PID • TuneRuleTIR = 4: Ziegler-Nichols PI • TuneRuleTIR = 5: Ziegler-Nichols P
PIDSelfTune.TIR.State	INT	0	Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung":

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.Mode	INT	0	Wenn sich der Wert von Retain.Mode ändert, wird in eine andere Betriebsart gewechselt. Folgende Betriebsart wird aktiviert bei Änderung von Mode zu: • Mode = 0: Inaktiv • Mode = 1: Erstoptimierung • Mode = 2: Nachoptimierung • Mode = 3: Automatikbetrieb • Mode = 4: Handbetrieb • Mode = 5: Ersatzausgangswert anfahren • Mode = 6: Stellzeitmessung • Mode = 7: Fehlerüberwachung • Mode = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung Mode ist remanent.
Retain.CtrlParams.SetByUser(1)	BOOL	FALSE	Wenn SetByUser = FALSE, sind die PID-Parameter automatisch ermittelt und PID_3Step arbeitet mit einer Totzone am Ausgangswert. Die Totzonenbreite wird während der Optimierung anhand der Standardabweichung des Ausgangswerts berechnet und in Retain.CtrlParams.OutputDeadBand gespeichert. Wenn SetByUser = TRUE, sind die PID-Parameter manuell eingegeben und PID_3 Step arbeitet ohne Totzone am Ausgangswert. Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser ist remanent.
Retain.CtrlParams.Gain(1)	REAL	1.0	Aktive Proportionalverstärkung Gain ist remanent.
Retain.CtrlParams.Ti(1)	REAL	20.0	• Ti > 0.0: Aktive Integrationszeit • Ti = 0.0: I-Anteil ist ausgeschaltet Ti ist remanent.
Retain.CtrlParams.Td(1)	REAL	0.0	• Td > 0.0: Aktive Differenzierzeit • Td = 0.0: D-Anteil ist ausgeschaltet Td ist remanent.
Retain.CtrlParams.TdFiltRatio(1)	REAL	0.0	Aktiver Koeffizient für den Differenzierverzug TdFiltRatio ist remanent.
Retain.CtrlParams.PWeighting(1)	REAL	0.0	Aktive Gewichtung des P-Anteils PWeighting ist remanent.
Retain.CtrlParams.DWeighting(1)	REAL	0.0	Aktive Gewichtung des D- Anteils DWeighting ist remanent.
Retain.CtrlParams.Cycle(1)	REAL	1.0	Aktive Abtastzeit PID-Algorithmus in Sekunden, die zu einem ganzzahligen Vielfachen der Zykluszeit des aufrufenden OB aufgerundet wird. Cycle ist remanent.
Retain.CtrlParams.InputDeadBand(1)	REAL	0.0	Totzonenbreite der Regeldifferenz InputDeadBand ist remanent.

Siehe auch

Parameter State und Retain.Mode V1 (Seite 3013)

Variable ActivateRecoverMode V1 (Seite 3023)

Parameter State und Retain.Mode V1

Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Um die Betriebsart zu wechseln, müssen Sie die Variable Retain.Mode ändern. Das gilt auch, wenn in Retain.Mode bereits der Wert für die neue Betriebsart steht. Setzen Sie dann z. B. erst Retain.Mode = 0 und anschließend Retain.Mode = 3. Wenn die aktuelle Betriebsart des Reglers diesen Wechsel erlaubt, wird State auf den Wert von Retain.Mode gesetzt.

Wenn PID_3Step automatisch die Betriebsart wechselt gilt: State != Retain.Mode.

Beispiele:

- Nach erfolgreicher Erstoptimierung
  State = 3 und Retain.Mode = 1
- Im Fehlerfall
  State = 0 und Retain.Mode bleibt auf bisherigem Wert, z. B. Retain.Mode = 3
- ManualEnalbe = TRUE
  State = 4 und Retain.Mode bleibt auf bisherigem Wert, z. B. Retain.Mode = 3

Hinweis

Sie wollen z. B. eine erfolgreiche Nachoptimierung wiederholen ohne den Automatikbetrieb mit Mode = 0 zu beenden.

Wenn Sie Retain.Mode für einen Takt auf einen ungültigen Wert setzen, z. B. 9999, hat das keinen Einfluss auf State. Im nächsten Takt setzen Sie Mode = 2. So können Sie eine Änderung an Retain.Mode erzeugen, ohne erst in die Betriebsart "Inaktiv" zu wechseln.

Bedeutung der Werte

State / Retain.Mode	Beschreibung
0	Inaktiv Der Regler ist ausgeschaltet und verändert die Position des Ventils nicht mehr.
1	Erstoptimierung Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet. Voraussetzungen für die Erstoptimierung: • State = 0 oder State = 4 • ManualEnable = FALSE • Die Motorstellzeit ist konfiguriert oder gemessen. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit Config.LoadBackUp reaktiviert werden. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstoptimierung in die Betriebsart "Inaktiv". Die Phase der Erstoptimierung wird mit Variable SUT.State angezeigt.

State / Retain.Mode	Beschreibung
2	Nachoptimierung Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter optimiert. Die Unterschiede zwischen der Prozessantwort während Erst- und Nachoptimierung werden analysiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Vor der Nachoptimierung werden die PID-Parameter gesichert. Sie können mit Config.LoadBackUp reaktiviert werden. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Voraussetzungen für die Nachoptimierung: • Die Motorstellzeit ist konfiguriert oder gemessen. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. • ManualEnable = FALSE • Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4) Die Nachoptimierung verläuft folgendermaßen beim Start aus: • Automatikbetrieb (State = 3) Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb. PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. • Inaktiv (State = 0) oder Handbetrieb (State = 4) Zuerst wird immer eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Wenn PIDSelfTune.TIR.RunIn = TRUE, wird die Erstoptimierung übersprungen und versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet dann automatisch. Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; bei erfolgloser Nachoptimierung in die Betriebsart "Inaktiv". Die Phase der Nachoptimierung wird mit Variable TIR.State angezeigt.
3	Automatikbetrieb Im Automatikbetrieb regelt PID_3Step die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt: • Erstoptimierung erfolgreich abgeschlossen • Nachoptimierung erfolgreich abgeschlossen • Ändern der Variable Retain.Mode auf den Wert 3. Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_3Step in der zuletzt aktiven Betriebsart. Um PID_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE. Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.

State / Retain.Mode	Beschreibung
4	Handbetrieb Im Handbetrieb geben Sie manuelle Ausgangswerte an den Parametern Manual_UP und Manual_DN oder ManualValue vor. Ob das Stellglied im Fehlerfall auf den Ausgangswert verfahren werden kann, ist beim Parameter ErrorBits beschrieben. Diese Betriebsart wird aktiviert, wenn Retain.Mode = 4 oder bei steigender Flanke an ManualEnable. Wenn ManualEnable TRUE wird, ändert sich nur State. Retain.Mode bleibt auf dem bisherigen Wert. Bei einer fallenden Flanke an ManualEnable kehrt PID_3Step in die vorhergehende Betriebsart zurück. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei. **PID_3Step V1.1** Im Fehlerfall ist der Handbetrieb immer möglich. **PID_3Step V1.0** Im Fehlerfall ist der Handbetrieb abhängig von der Variablen ActivateRecoverMode.
5	Ersatzausgangswert anfahren Diese Betriebsart wird im Fehlerfall oder bei Reset = TRUE aktiviert, wenn Errorbehaviour = 1 und ActivateRecoverMode = FALSE. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Inaktiv".
6	Stellzeitmessung Es wird die Zeit ermittelt, die der Motor benötigt, um das Ventil aus dem geschlossenen Zustand vollständig zu öffnen. Diese Betriebsart wird aktiviert, wenn GetTransitTime.Start = TRUE gesetzt wird. Wenn zur Stellzeitmessung Anschlagsignale verwendet werden, wird das Ventil von der aktuellen Position aus vollständig geöffnet, vollständig geschlossen und erneut vollständig geöffnet. Wenn GetTransitTime.InvertDirection = TRUE, wird dieses Verhalten umgekehrt. Wenn zur Stellzeitmessung eine Stellungsrückmeldung verwendet wird, wird das Stellglied von der aktuellen Stellung in eine Zielstellung gefahren. Die Ausgangswertgrenzen werden während der Stellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

State / Retain.Mode	Beschreibung
7	Fehlerüberwachung Der Regelalgorithmus ist ausgeschaltet und verändert die Position des Ventils nicht mehr. Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert. Alle folgenden Bedingungen müssen erfüllt sein: • Mode = 3 (Automatikbetrieb) • Errorbehaviour = 0 • ActivateRecoverMode = TRUE • Einer oder mehrere Fehler sind auftreten, bei denen ActivateRecoverMode (Seite 3023) wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.
8	Ersatzausgangswert anfahren mit Fehlerüberwachung Diese Betriebsart wird im Fehlerfall statt der Betriebsart " Ersatzausgangswert anfahren" aktiviert. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Fehlerüberwachung". Alle folgenden Bedingungen müssen erfüllt sein: • Mode = 3 (Automatikbetrieb) • Errorbehaviour = 1 • ActivateRecoverMode = TRUE • Einer oder mehrere Fehler sind auftreten, bei denen ActivateRecoverMode (Seite 3023) wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.

Automatische Betriebsartwechsel während der Inbetriebnahme

Im Fehlerfall wechselt PID_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour auf den Betriebsartwechsel aus den Betriebsarten Stellzeitmessung, Erstoptimierung und Nachoptimierung.

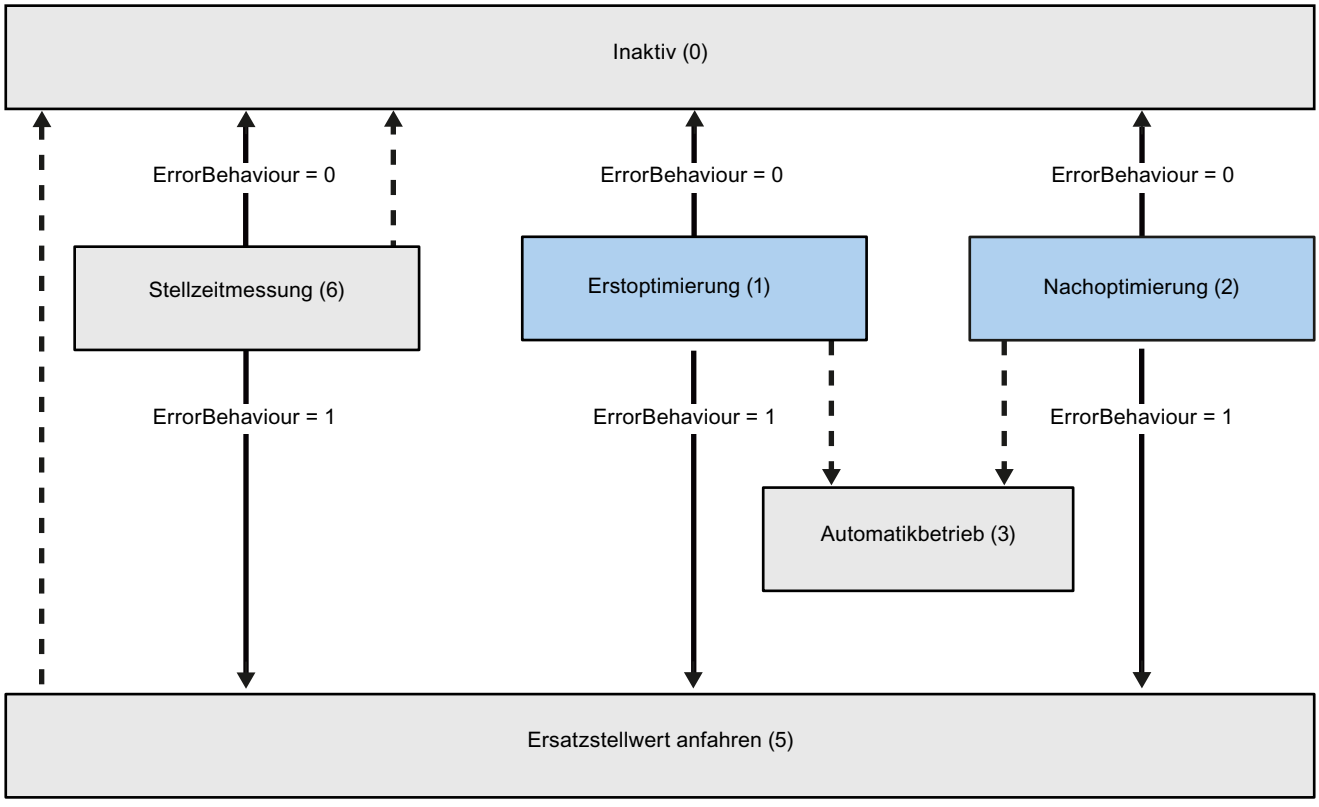

Automatischer Betriebsartwechsel im Fehlerfall

Automatischer Betriebsartwechsel, wenn der aktuelle Betrieb abgeschlossen ist.

Automatische Betriebsartwechsel im Automatikbetrieb (PID_3Step V1.1)

Im Fehlerfall wechselt PID_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.

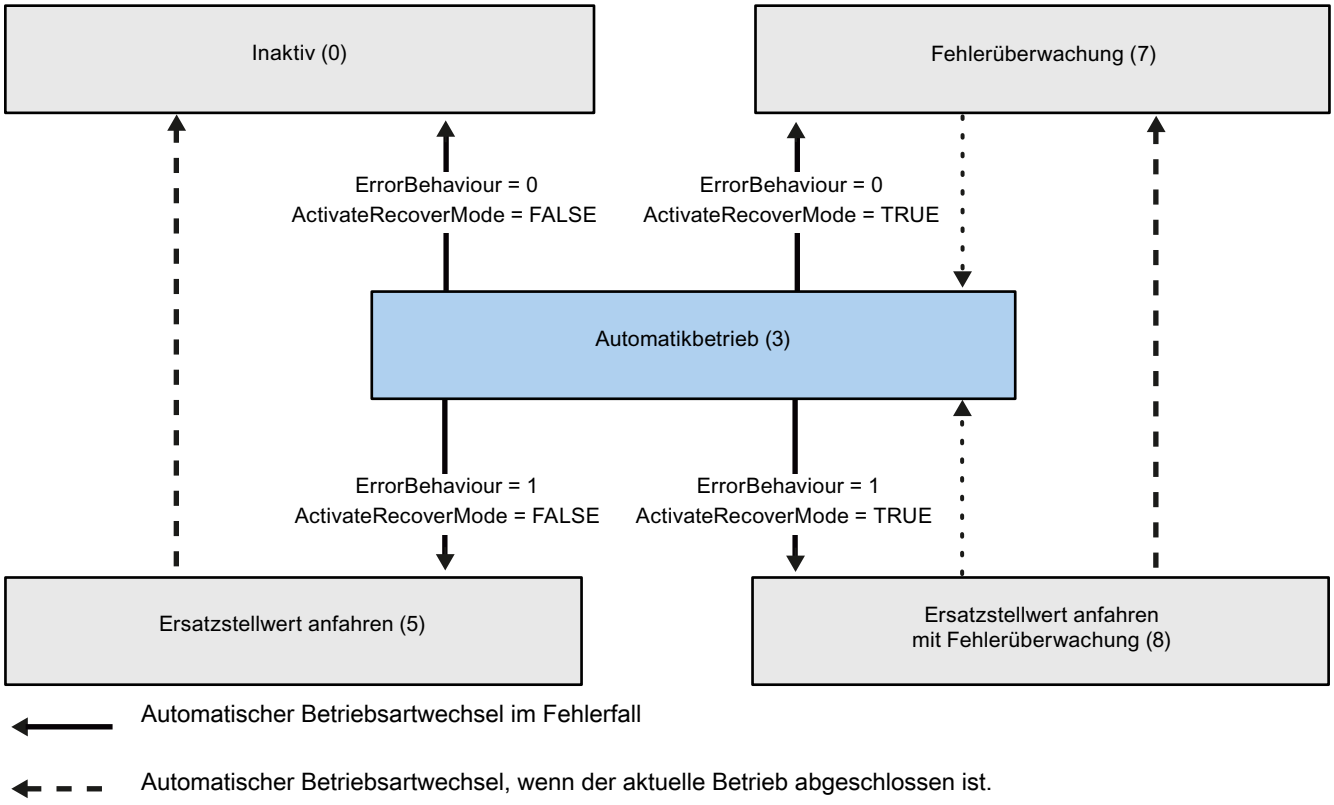

Automatischer Betriebsartwechsel im Fehlerfall

Automatischer Betriebsartwechsel, wenn der aktuelle Betrieb abgeschlossen ist.

Automatische Betriebsartwechsel im Automatik- und Handbetrieb (PID_3Step V1.0)

Im Fehlerfall wechselt PID_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.

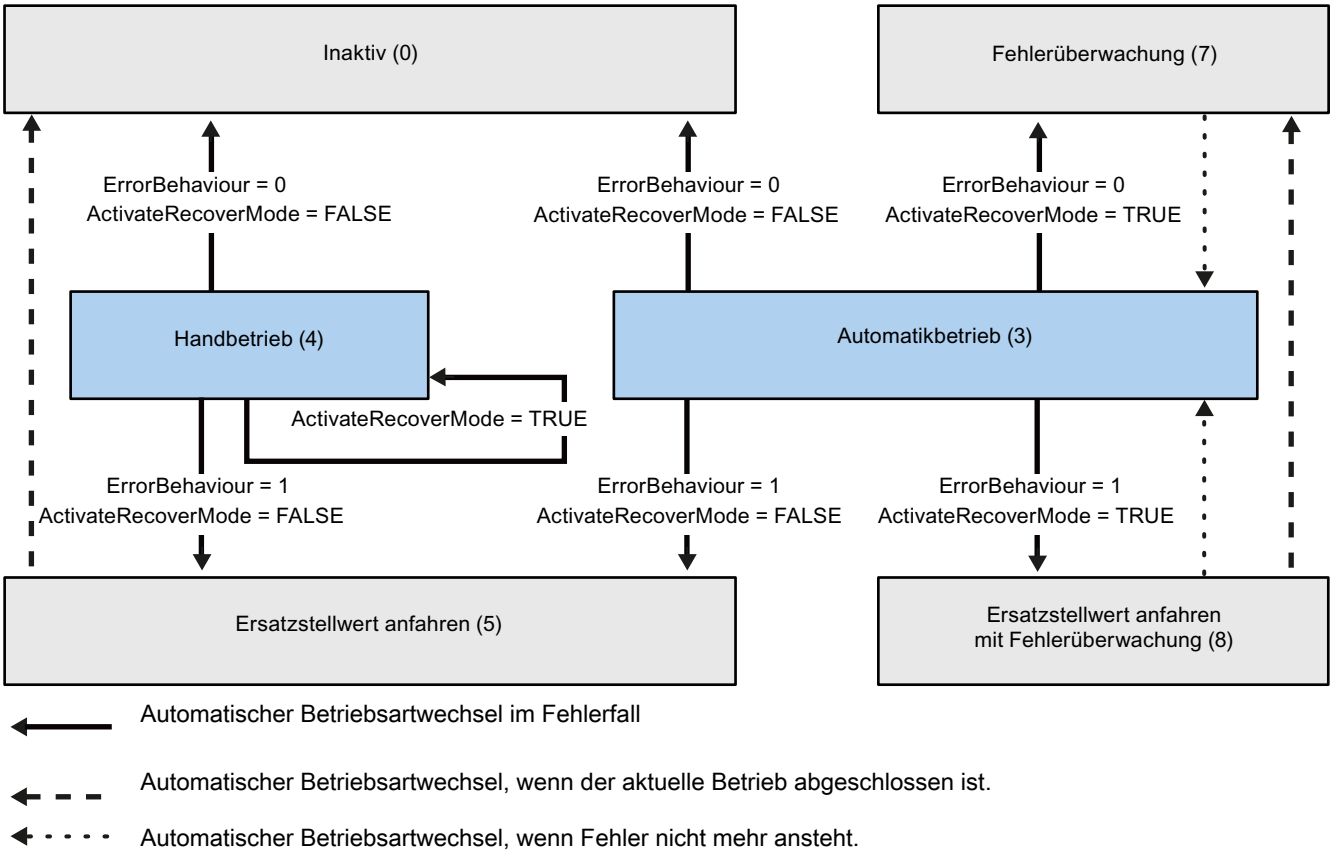

Siehe auch

Variable ActivateRecoverMode V1 (Seite 3023)

Parameter ErrorBits V1 (Seite 3021)

Parameter ErrorBits V1

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der Errorcodes binär addiert angezeigt. Wird z. B. der Errorcode 0003 angezeigt, so stehen gleichzeitig die Fehler 0001 und 0002 an.

ErrorBits (DW#16#...)	Beschreibung
0000	Es steht kein Fehler an.
0001	Der Parameter "Input" ist außerhalb der Istwertgrenzen. • Input > Config.InputUpperLimit oder • Input < Config.InputLowerLimit Wenn ActivateRecoverMode = TRUE und ErrorBehaviour = 1, wird das Stellglied auf den Ersatzausgangswert verfahren. Wenn ActivateRecoverMode = TRUE und ErrorBehaviour = 0, bleibt das Stellglied in der aktuellen Position stehen. Wenn ActivateRecoverMode = FALSE, bleibt das Stellglied in der aktuellen Position stehen. PID_3Step V1.1 Im Handbetrieb können Sie das Stellglied verfahren. PID_3Step V1.0 Handbetrieb ist in diesem Zustand nicht möglich. Erst wenn der Fehler behoben ist, können Sie das Stellglied wieder verfahren.
0002	Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.
0004	Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden.
0020	Die Erstoptimierung ist im Automatikbetrieb und während der Nachoptimierung nicht erlaubt.
0080	Fehler bei der Erstoptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert oder der Istwert reagiert nicht wie erwartet. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen. Stellen Sie außerdem sicher, dass der Istwert vor Start der Erstoptimierung nicht stark schwingt.
0100	Fehler während der Nachoptimierung führte zu ungültigen Parametern.
0200	Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.
0400	Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.
0800	Abtastzeitfehler: PID_3Step wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. Falls dieser Fehler bei der Simulation mit PLCSIM aufgetreten ist, beachten Sie die Hinweise unter AUTOHOTSPOT.
1000	Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.

ErrorBits (DW#16#...)	Beschreibung
2000	Ungültiger Wert am Parameter Feedback_PER. Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die Stellungsrückmeldung deaktivieren (Config. FeedbackOn = FALSE). Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.
4000	Ungültiger Wert am Parameter Feedback. Wert hat kein gültiges Zahlenformat. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die Stellungsrückmeldung deaktivieren (Config. FeedbackOn = FALSE). Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.
8000	Fehler bei der digitalen Stellungsrückmeldung. Actuator_H = TRUE und Actuator_L = TRUE. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen die "Anschlagsignale Stellglied" deaktiviert werden (Config.ActuatorEndStopOn = FALSE). Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.

Parameter Reset V1

Durch eine steigende Flanke an Reset wird in die Betriebsart "Inaktiv" gewechselt und Fehler und Warnungen werden zurückgesetzt. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt. Falls zuvor Automatikbetrieb aktiv war, erfolgt die Umschaltung in den Automatikbetrieb stoßfrei.

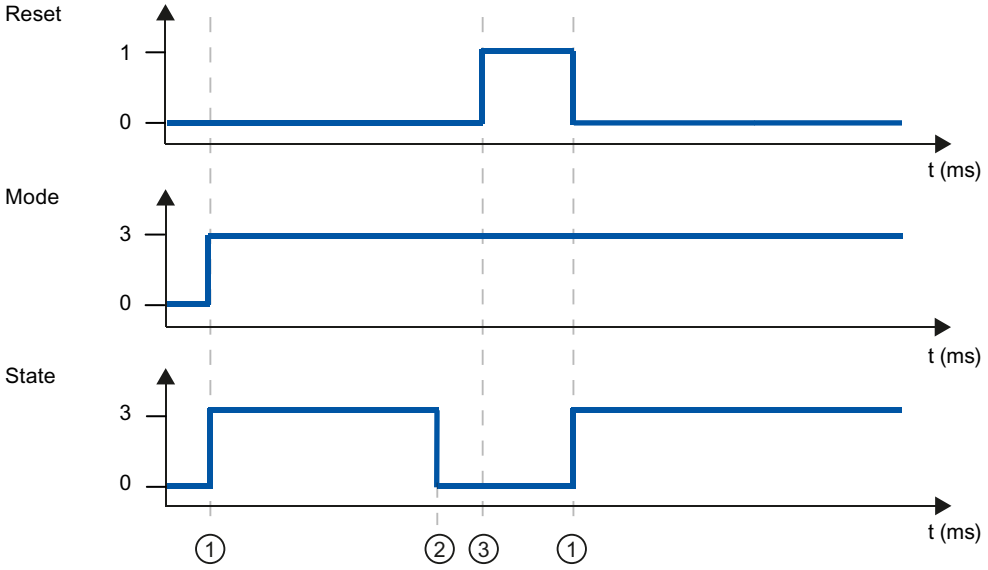

① Einschalten
② Fehler
③ Reset

Variable ActivateRecoverMode V1

Der Einfluss der Variablen ActivateRecoverMode ist abhängig von der Version der Anweisung PID_3Step.

Verhalten in Version 1.1

Die Variable ActivateRecoverMode bestimmt im Automatikbetrieb das Verhalten im Fehlerfall. Während der Erstoptimierung, Nachoptimierung und Stellzeitmessung wirkt ActivateRecoverMode nicht.

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch Reset oder eine Änderung an Retain.Mode aktiviert.
TRUE	Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung": • 0002h: Ungültiger Wert am Parameter Input_PER. • 0200h: Ungültiger Wert am Parameter Input. • 0800h: Abtastzeitfehler • 1000h: Ungültiger Wert am Parameter Setpoint. • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung. Bei den Fehlern 2000h, 4000h und 8000h kann PID_3Step den konfigurierten Ersatzausgangswert **nicht** anfahren. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.

Verhalten in Version 1.0

Die Variable ActivateRecoverMode bestimmt im Automatik- und Handbetrieb das Verhalten im Fehlerfall. Während der Erstoptimierung, Nachoptimierung und Stellzeitmessung wirkt ActivateRecoverMode nicht.

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch Reset oder eine Änderung an Retain.Mode aktiviert.
TRUE	**Fehler im Automatikbetrieb** Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung": • 0002h: Ungültiger Wert am Parameter Input_PER. • 0200h: Ungültiger Wert am Parameter Input. • 0800h: Abtastzeitfehler • 1000h: Ungültiger Wert am Parameter Setpoint. • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung. Bei den Fehlern 2000h, 4000h und 8000h kann PID_3Step den konfigurierten Ersatzausgangswert **nicht** anfahren. Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb. **Fehler im Handbetrieb** Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_3Step im Handbetrieb: • 0002h: Ungültiger Wert am Parameter Input_PER. • 0200h: Ungültiger Wert am Parameter Input. • 0800h: Abtastzeitfehler • 1000h: Ungültiger Wert am Parameter Setpoint. • 2000h: Ungültiger Wert am Parameter Feedback_PER. • 4000h: Ungültiger Wert am Parameter Feedback. • 8000h: Fehler bei der digitalen Stellungsrückmeldung. Bei den Fehlern 2000h, 4000h und 8000h können Sie das Ventil **nicht** in eine geeignete Position fahren.

Siehe auch

Statische Variablen PID_3Step V1 (Seite 3005)

Parameter State und Retain.Mode V1 (Seite 3013)

Variable Warning V1

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Warnungen binär addiert angezeigt. Wird z. B. die Warnung 0003 angezeigt, so stehen gleichzeitig die Warnungen 0001 und 0002 an.

Warning (DW#16#...)	Beschreibung
0000	Es steht keine Warnung an.
0001	Während der Erstoptimierung wurde der Wendepunkt nicht gefunden.
0002	Während der Nachoptimierung war die Oszillation verstärkt.
0004	Der Sollwert wurde begrenzt auf die eingestellten Grenzen.
0008	Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TuneRuleTIR = 3 berechnet.
0010	Die Betriebsart konnte nicht geändert werden, da ManualEnable = TRUE.
0020	Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.
0040	Der Istwert hat eine seiner Warngrenze überschritten.
0080	Ungültiger Wert an Retain.Mode. Die Betriebsart wird nicht gewechselt.
0100	Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.
0200	Die verwendete Regel zur Optimierung führt zu keinem richtigen Ergebnis oder wird nicht unterstützt.
0400	Bei Stellzeitmessung Methode ausgewählt nicht zum Stellglied passt. Die Stellzeit kann nicht gemessen werden, da die Einstellungen des Stellglieds nicht zur gewählten Messmethode passen.
0800	Bei der Stellzeitmessung ist die Differenz zwischen der aktuellen Position und dem neuen Ausgangswert zu klein. Dies kann zu falschen Ergebnissen führen. Die Differenz zwischen aktuellem Ausgangswert und neuem Ausgangswert muss mindestens 50% des gesamten Stellbereichs betragen.
1000	Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0004
- 0020
- 0040
- 0100

Alle anderen Warnungen werden bei steigender Flanke an Reset gelöscht.

Variable SUT.State V1

SUT.State	Name	Beschreibung
0	SUT_INIT	Erstoptimierung initialisieren
50	SUT_TPDN	Startposition ohne Stellungsrückmeldung ermitteln
100	SUT_STDABW	Standardabweichung berechnen
200	SUT_GET_POI	Wendepunkt ermitteln
300	SUT_GET_RISETM	Anregelzeit ermitteln

SUT.State	Name	Beschreibung
9900	SUT_IO	Erstoptimierung erfolgreich
1	SUT_NIO	Erstoptimierung nicht erfolgreich

Variable TIR.State V1

TIR.State	Name	Beschreibung
-100	TIR_FIRST_SUT	Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt.
0	TIR_INIT	Nachoptimierung initialisieren
200	TIR_STDABW	Standardabweichung berechnen
300	TIR_RUN_IN	Versuchen Sollwert mit dem maximalen oder minimalen Ausgangswert zu erreichen
400	TIR_CTRLN	Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war)
500	TIR_OSZIL	Oszillation ermitteln und Parameter berechnen
9900	TIR_IO	Nachoptimierung erfolgreich
1	TIR_NIO	Nachoptimierung nicht erfolgreich

PID_Temp

Neuerungen PID_Temp

PID_Temp V1.1

- **Reaktion des Ausgangswerts beim Wechsel von der Betriebsart “Inaktiv” in "Automatikbetrieb"**
  Die neue Option IntegralResetMode = 4 wurde hinzugefügt und als Vorbelegung festgelegt. Mit IntegralResetMode = 4 wird der I-Anteil, beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb", automatisch so vorbelegt, dass eine Regeldifferenz zu einem Sprung des PID Ausgangswerts mit identischem Vorzeichen führt.
- **Initialisierung des Integral-Anteils im Automatikbetrieb**
  Der Integral-Anteil kann im Automatikbetrieb mit Hilfe der Variablen OverwriteInitialOutputValue und PIDCtrl.PIDInit initialisiert werden. Dies vereinfacht den Einsatz von PID_Temp für Ablöseregelungen.

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von PID_Temp einsetzen können.

CPU	FW	PID_Temp
S7-1200	ab V4.2	V1.1 V1.0
	V4.1	V1.0

CPU	FW	PID_Temp
S7-1500	ab V2.0	V1.1 V1.0
	V1.7 bis V1.8	V1.0

CPU-Bearbeitungszeit und Speicherbedarf PID_Temp V1

CPU-Bearbeitungszeit

Typische CPU-Bearbeitungszeiten des Technologieobjekts PID_Temp ab der Version V1.0 in Abhängigkeit vom CPU-Typ.

CPU	Typ. CPU-Bearbeitungszeit PID_Temp V1
CPU 1211C ≥ V4.1	580 µs
CPU 1215C ≥ V4.1	580 µs
CPU 1217C ≥ V4.1	580 µs
CPU 1505S ≥ V1.0	50 µs
CPU 1510SP-1 PN ≥ V1.7	130 µs
CPU 1511-1 PN ≥ V1.7	130 µs
CPU 1512SP-1 PN ≥ V1.7	130 µs
CPU 1516-3 PN/DP ≥ V1.7	75 µs
CPU 1518-4 PN/DP ≥ V1.7	6 µs

Speicherbedarf

Speicherbedarf eines Instanz-DB des Technologieobjekts PID_Temp ab der Version V1.0.

	Speicherbedarf des Instanz-DB von PID_Temp V1
Ladespeicherbedarf	ca. 17000 Bytes
Gesamter Arbeitsspeicherbedarf	1280 Bytes
Remanenter Arbeitsspeicherbedarf	100 Bytes

PID_Temp

Beschreibung PID_Temp

Beschreibung

Die Anweisung PID_Temp stellt einen PID-Regler mit integrierter Optimierung für Temperaturprozesse zur Verfügung. PID_Temp eignet sich für reine Heizen- oder Heizen/Kühlen-Anwendungen.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert mit Fehlerüberwachung

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

PID-Algorithmus

PID_Temp ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der PID-Algorithmus arbeitet nach folgender Formel (Regelzone und Totzone deaktiviert):

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Die folgende Tabelle zeigt die Bedeutung der in der Formel und in den nachfolgenden Bildern verwendeten Symbole.

Symbol	Beschreibung	Zugehörige Parameter der Anweisung PID_Temp
y	Ausgangswert des PID-Algorithmus	-
K_p	Proportionalverstärkung	Retain.CtrlParams.Heat.Gain Retain.CtrlParams.Cool.Gain CoolFactor
s	Laplace-Operator	-
b	Gewichtung des P-Anteils	Retain.CtrlParams.Heat.PWeighting Retain.CtrlParams.Cool.PWeighting
w	Sollwert	CurrentSetpoint
x	Istwert	ScaledInput
T_I	Integrationszeit	Retain.CtrlParams.Heat.Ti Retain.CtrlParams.Cool.Ti
T_D	Differenzierzeit	Retain.CtrlParams.Heat.Td Retain.CtrlParams.Cool.Td
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)	Retain.CtrlParams.Heat.TdFiltRatio Retain.CtrlParams.Cool.TdFiltRatio
c	Gewichtung des D-Anteils	Retain.CtrlParams.Heat.DWeighting Retain.CtrlParams.Cool.DWeighting
DeadZone	Totzonenbreite	Retain.CtrlParams.Heat.DeadZone Retain.CtrlParams.Cool.DeadZone
ControlZone	Regelzonenbreite	Retain.CtrlParams.Heat.ControlZone Retain.CtrlParams.Cool.ControlZone

Blockschaltbild PID_Temp

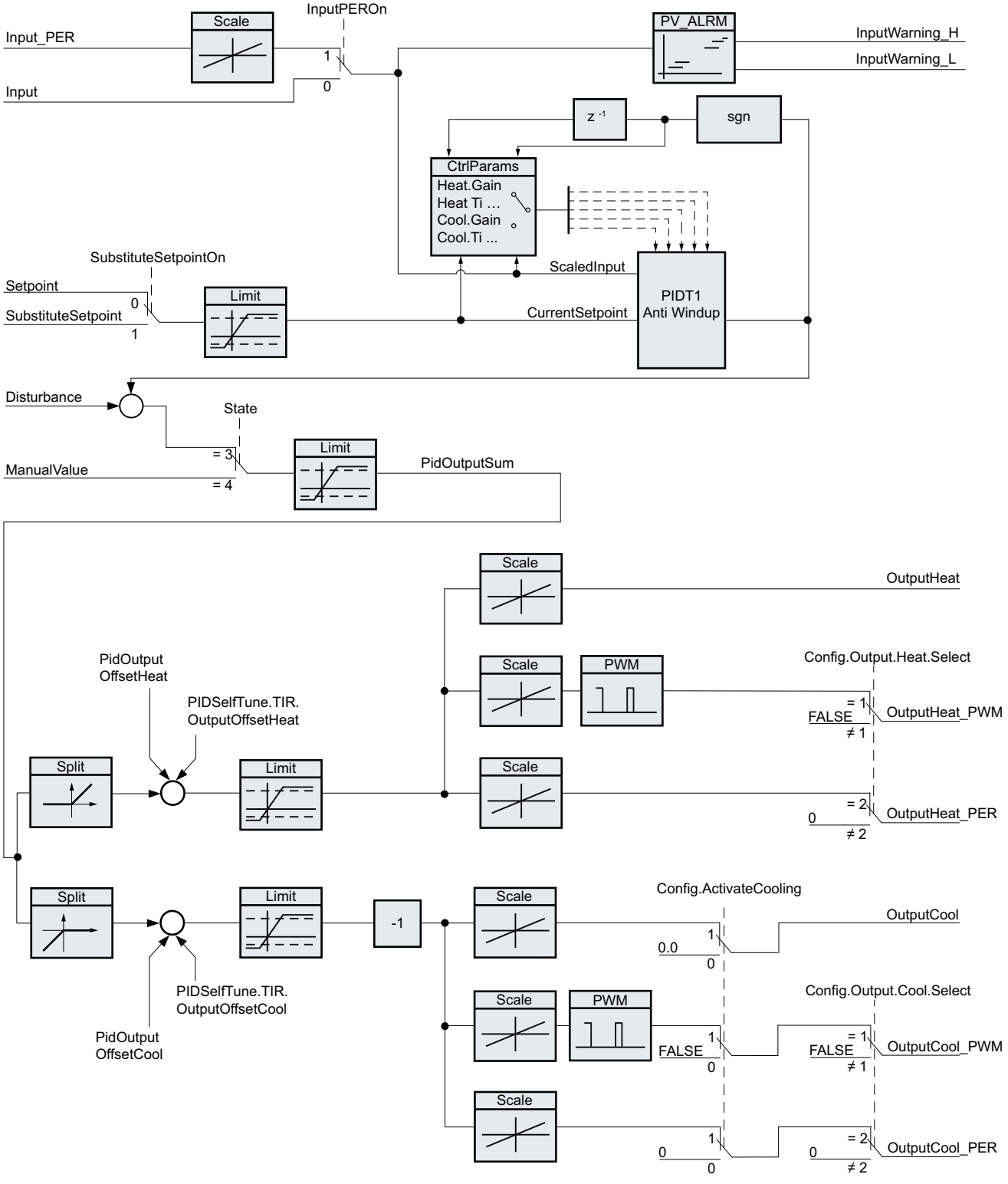

Blockschaltbild PIDT1 mit Anti-Windup

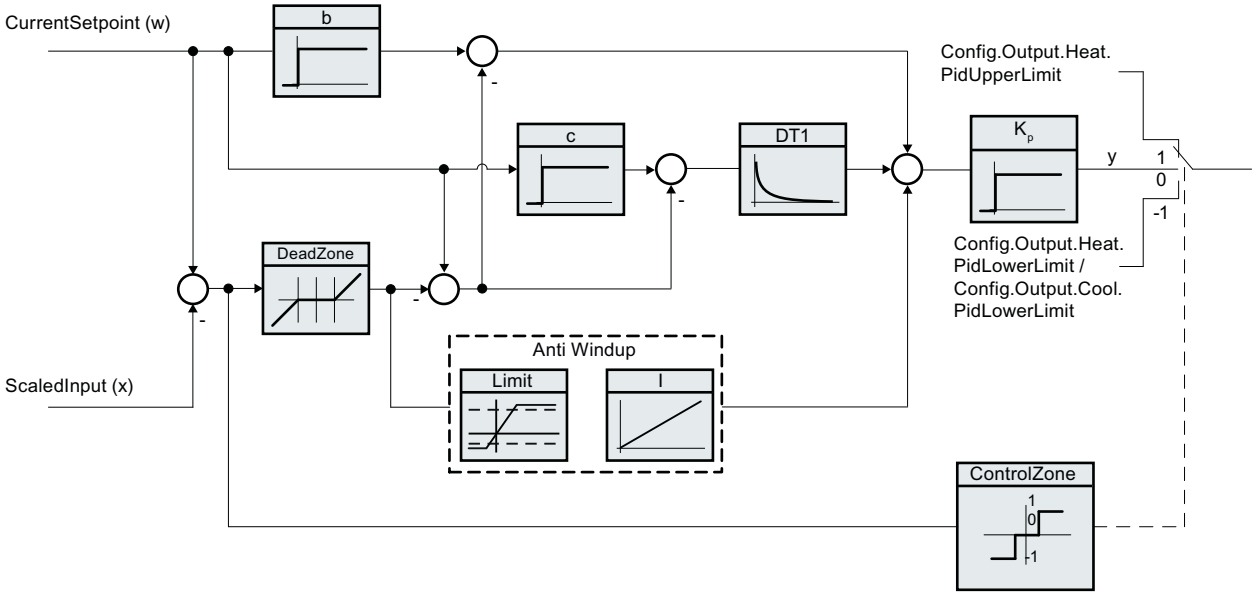

Aufruf

PID_Temp wird im konstanten Zeitraster eines Weckalarm-OBs aufgerufen.

Wenn Sie PID_Temp als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_Temp direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID_Temp vollständig laden.

Technologieobjekt in Gerät laden

Anlauf

Bei Anlauf der CPU startet PID_Temp in der Betriebsart, die am Durchgangsparameter Mode gespeichert ist. Um beim Anlauf in die Betriebsart "Inaktiv" zu wechseln, setzen Sie RunModeByStartup = FALSE.

Verhalten im Fehlerfall

Das Verhalten im Fehlerfall hängt von den Variablen SetSubstituteOutput und ActivateRecoverMode ab. Falls ActivateRecoverMode = TRUE, ist das Verhalten zudem abhängig vom aufgetretenen Fehler.

SetSubstituteOutput	ActivateRecoverMode	Konfigurationseditor > Grundeinstellungen Ausgang > PidOutputSum setzen auf	Verhalten
nicht relevant	FALSE	Null (Inaktiv)	Wechsel in Betriebsart "Inaktiv" (State = 0) Der Ausgangswert des PID-Algorithmus und alle Ausgänge für Heizen und Kühlen werden auf 0 gesetzt. Die Skalierung der Ausgänge für Heizen und Kühlen ist nicht aktiv.
FALSE	TRUE	Aktueller Wert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5) Der aktuelle Ausgangswert wird für die Fehlerdauer an das Stellglied übertragen.
TRUE	TRUE	Ersatzausgangswert für die Fehlerdauer	Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5) Der Wert an SubstituteOutput wird für die Fehlerdauer an das Stellglied übertragen.

PID_Temp verwendet im Handbetrieb ManualValue als Ausgangswert, außer ManualValue ist ungültig.

- Falls ManualValue ungültig ist, wird SubstituteOutput verwendet.
- Falls ManualValue und SubstituteOutput ungültig sind, wird Config.Output.Heat.PidLowerLimit verwendet..

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits wird durch eine steigende Flanke an Reset oder ErrorAck zurückgesetzt.

Arbeitsweise PID_Temp

Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0000001h).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warning = 0000040h) und der Ausgangsparameter InputWarning_H oder InputWarning_L wird TRUE.

Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID_Temp begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID_Temp prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Falls der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit_H oder SetpointLimit_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

Ersatzsollwert

An der Variablen SubstituteSetpoint können Sie einen Ersatzsollwert vorgeben und diesen mit SubstituteSetpointOn = TRUE aktivieren. Damit können Sie z. B. für einen Slave-Regler in einer Kaskade vorübergehend den Sollwert direkt vorgeben ohne das Anwenderprogramm zu ändern. Die für den Sollwert festgelegten Grenzen gelten auch für den Ersatzsollwert.

Heizen und Kühlen

Mit der Voreinstellung verwendet PID_Temp nur die Ausgänge für Heizen (OutputHeat, OutputHeat_PWM, OutputHeat_PER). Der Ausgangswert des PID-Algorithmus (PidOutputSum) wird skaliert und an den Ausgängen für Heizen ausgegeben. Ob OutputHeat_PWM oder OutputHeat_PER berechnet werden, legen Sie mit Config.Output.Heat.Select fest. OutputHeat wird immer berechnet.

Mit Config.ActivateCooling = TRUE können Sie auch die Ausgänge für Kühlen (OutputCool, OutputCool_PWM, OutputCool_PER) aktivieren. Positive Ausgangswerte des PID-Algorithmus (PidOutputSum) werden skaliert und an den Ausgängen für Heizen ausgegeben. Negative Ausgangswerte des PID-Algorithmus werden skaliert und an den Ausgängen für Kühlen ausgegeben. Ob OutputCool_PWM oder OutputCool_PER berechnet werden, legen Sie mit Config.Output.Cool.Select fest. OutputCool wird immer berechnet.

Für die Berechnung des PID Ausgangswerts bei aktivierter Kühlung stehen zwei Methoden zur Verfügung:

- Kühlfaktor (Config.AdvancedCooling = FALSE):
  Die Ausgangswertberechnung für Kühlen erfolgt mit den PID-Parametern für Heizen unter Berücksichtigung des konfigurierbaren Kühlfaktors Config.CoolFactor. Diese Methode ist geeignet, falls Heiz- und Kühlstellglied ähnliches Zeitverhalten aber unterschiedliche Verstärkungen aufweisen. Bei Wahl dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen sowie der PID-Parametersatz für Kühlen nicht zur Verfügung. Es können nur die Optimierungen für Heizen ausgeführt werden.
- PID-Parameterumschaltung (Config.AdvancedCooling = TRUE):
  Die Ausgangswertberechnung für Kühlen erfolgt über einen eigenen PID-Parametersatz. Der PID-Algorithmus entscheidet anhand des berechneten Ausgangswerts und der Regeldifferenz, ob die PID-Parameter für Heizen oder Kühlen verwendet werden. Diese Methode ist geeignet, falls Heiz- und Kühlstellglied unterschiedliches Zeitverhalten und Verstärkungen aufweisen. Nur bei Wahl dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen zur Verfügung.

Ausgangswertgrenzen und –skalierung

Der PID Ausgangswert (PidOutputSum) wird je nach Betriebsart durch den PID-Algorithmus automatisch berechnet oder durch den Handwert (ManualValue) oder den konfigurierten Ersatzausgangswert (SubstituteOutput) vorgegeben.

Der PID Ausgangswert wird abhängig von der Konfiguration begrenzt:

- Ist die Kühlung deaktiviert (Config.ActivateCooling = FALSE), gilt als Obergrenze Config.Output.Heat.PidUpperLimit und als Untergrenze Config.Output.Heat.PidLowerLimit.
- Ist die Kühlung aktiviert (Config.ActivateCooling = TRUE), gilt als Obergrenze Config.Output.Heat.PidUpperLimit und als Untergrenze Config.Output.Cool.PidLowerLimit.

Der PID Ausgangswert wird skaliert und an den Ausgängen für Heizen und Kühlen ausgegeben. Die Skalierung kann für jeden Ausgang getrennt vorgegeben werden und wird über je 2 Wertepaare in den Strukturen Config.Output.Heat bzw. Config.Output.Cool festgelegt:

Ausgang	Wertepaar	Parameter
OutputHeat	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer Ausgangswert (Heizen) Config.Output.Heat.UpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer Ausgangswert (Heizen) Config.Output.Heat.LowerScaling
OutputHeat_PWM	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer PWM Ausgangswert (Heizen) Config.Output.Heat.PwmUpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer PWM Ausgangswert (Heizen) Config.Output.Heat.PwmLowerScaling
OutputHeat_PER	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer analoger Ausgangswert (Heizen) Config.Output.Heat.PerUpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer analoger Ausgangswert (Heizen) Config.Output.Heat.PerLowerScaling

Ausgang	Wertepaar	Parameter
OutputCool	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer Ausgangswert (Kühlen) Config.Output.Cool.UpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer Ausgangswert (Kühlen) Config.Output.Cool.LowerScaling
OutputCool_PWM	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer PWM Ausgangswert (Kühlen) Config.Output.Cool.PwmUpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer PWM Ausgangswert (Kühlen) Config.Output.Cool.PwmLowerScaling
OutputCool_PER	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer analoger Ausgangswert (Kühlen) Config.Output.Cool.PerUpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer analoger Ausgangswert (Kühlen) Config.Output.Cool.PerLowerScaling

Falls die Kühlung aktiviert ist (Config.ActivateCooling = TRUE), muss Config.Output.Heat.PidLowerLimit den Wert 0.0 haben.

Config.Output.Cool.PidUpperLimit muss immer den Wert 0.0 haben.

Beispiel:

Ausgangsskalierung bei Verwendung von Ausgang OutputHeat (Kühlung deaktiviert; Config.Output.Heat.PidLowerLimit darf ungleich 0.0 sein):

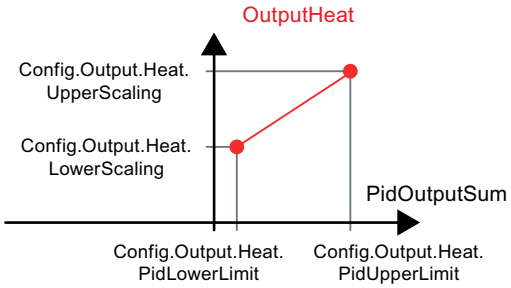

Beispiel:

Ausgangsskalierung bei Verwendung von Ausgang OutputHeat_PWM und OutputCool_PER (Kühlung aktiviert; Config.Output.Heat.PidLowerLimit muss 0.0 sein):

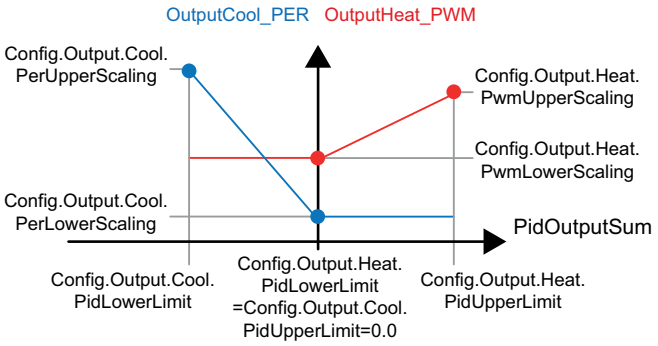

Mit Ausnahme der Betriebsart "Inaktiv", liegt der Wert an einem Ausgang immer zwischen seinem skalierten oberen Ausgangswert und skalierten unteren Ausgangswert z. B. für OutputHeat immer zwischen Config.Output.Heat.UpperScaling und Config.Output.Heat.LowerScaling.

Falls Sie den Wert an dem zugehörigen Ausgang begrenzen wollen, müssen Sie daher auch diese Skalierungswerte anpassen.

Kaskadierung

PID_Temp bietet Ihnen Unterstützung bei der Verwendung in einer Kaskadenregelung (siehe: AUTOHOTSPOT).

Ersatzausgangswert

PID_Temp kann im Fehlerfall einen Ersatzausgangswert ausgeben, den Sie an der Variablen SubstituteOutput vorgeben. Der Ersatzausgangswert muss innerhalb der Grenzen für den PID Ausgangswert liegen. Die aus dem Ersatzausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung.

Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden bei Verwendung auf Gültigkeit überwacht:

- Setpoint
- SubstituteSetpoint
- Input
- Input_PER
- Disturbance
- ManualValue
- SubstituteOutput
- PID-Parameter in den Strukturen Retain.CtrlParams.Heat und Retain.CtrlParams.Cool.

Überwachung der Abtastzeit PID_Temp

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden Weckalarm-OB. Die Anweisung PID_Temp misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (Error = 0000800h).

Während der Optimierung tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,1 x alter Mittelwert
- Neuer Mittelwert <= 0,9 x alter Mittelwert

Im Automatikbetrieb tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert >= 1,5 x alter Mittelwert
- Neuer Mittelwert <= 0,5 x alter Mittelwert

Wenn Sie die Überwachung der Abtastzeit ausschalten (CycleTime.EnMonitoring = FALSE), können Sie PID_Temp auch im OB1 aufrufen. Sie müssen dann aufgrund der schwankenden Abtastzeit eine schlechtere Regelqualität akzeptieren.

Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit des PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit des Weckalarm-OB (Abtastzeit PID_Temp) gerundet. Alle anderen Funktionen des PID_Temp werden bei jedem Aufruf durchgeführt.

Sind Kühlung und PID-Parameterumschaltung aktiviert, verwendet PID_Temp jeweils eine eigene Abtastzeit des PID-Algorithmus für Heizen und Kühlen. In allen anderen Konfigurationen wird nur die Abtastzeit des PID-Algorithmus für Heizen verwendet.

Falls Sie OutputHeat_PWM bzw. OutputCool_PWM verwenden, wird die Abtastzeit des PID-Algorithmus als Periodendauer der Pulsweitenmodulation verwendet. Die Genauigkeit des Ausgangssignals wird bestimmt durch das Verhältnis von Abtastzeit des PID-Algorithmus zu Zykluszeit des OB. Die Zykluszeit sollte höchstens ein Zehntel der Abtastzeit des PID-Algorithmus betragen.

Ist bei Verwendung von OutputHeat_PWM bzw. OutputCool_PWM die Abtastzeit des PID-Algorithmus und damit die Periodendauer der Pulsweitenmodulation sehr groß, können Sie, um die Glattheit des Istwerts zu verbessern, an den Parametern Config.Output.Heat.PwmPeriode bzw. Config.Output.Cool.PwmPeriode eine abweichende kürzere Periodendauer vorgeben.

Regelsinn

PID_Temp kann für Heizen- oder Heizen/Kühlen-Anwendungen eingesetzt werden und arbeitet fest mit normalem Regelsinn.

Mit einer Erhöhung des PID-Ausgangswerts (PidOutputSum) soll eine Erhöhung des Istwerts erreicht werden. Die aus dem PID-Ausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung.

Invertierung des Regelsinns oder negative Proportionalverstärkung werden nicht unterstützt.

Benötigen Sie für Ihre Anwendung nur einen Ausgangswert, bei dem eine Erhöhung eine Verringerung des Istwerts bewirken soll (z. B. Abflussregelung), können Sie PID_Compact mit invertiertem Regelsinn einsetzen.

Eingangsparameter PID_Temp

Parameter	Datentyp	Vorbelegung	Beschreibung
Setpoint	REAL	0.0	Sollwert des PID-Reglers im Automatikbetrieb Zulässiger Wertebereich: Config.SetpointUpperLimit ≥ Setpoint ≥ Config.SetpointLowerLimit Config.InputUpperLimit ≥ Setpoint ≥ Config.InputLowerLimit
Input	REAL	0.0	Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet. Falls Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.
Input_PER	INT	0	Ein Analogeingang wird als Quelle des Istwerts verwendet. Falls Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.
Disturbance	REAL	0.0	Störgröße oder Vorsteuerungswert
ManualEnable	BOOL	FALSE	• Flanke FALSE -> TRUE aktiviert die Betriebsart "Handbetrieb", State = 4, Mode bleibt unverändert. Solange ManualEnable = TRUE ist, können Sie die Betriebsart nicht über eine steigende Flanke an ModeActivate ändern und den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE aktiviert die Betriebsart, die durch Mode vorgegeben wird. Es wird empfohlen, die Betriebsart nur über Mode und ModeActivate zu ändern.

Parameter	Datentyp	Vorbelegung	Beschreibung
ManualValue	REAL	0.0	Handwert Dieser Wert wird im Handbetrieb als PID Ausgangswert (PidOutputSum) verwendet. Die aus diesem Handwert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung (Strukturen Config.Output.Heat und Config.Output.Cool). Für Regler mit aktiviertem Kühlausgang (Config.ActivateCooling = TRUE) geben Sie an: • einen positiven Handwert, um den Wert an den Ausgängen für Heizen auszugeben • einen negativen Handwert, um den Wert an den Ausgängen für Kühlen auszugeben Der zulässige Wertebereich ist abhängig von der Konfiguration. • Kühlausgang deaktiviert (Config.ActivateCooling = FALSE): Config.Output.Heat.PidUpperLimit ≥ ManualValue ≥ Config.Output.Heat.PidLowerLimit • Kühlausgang aktiviert (Config.ActivateCooling = TRUE): Config.Output.Heat.PidUpperLimit ≥ ManualValue ≥ Config.Output.Cool.PidLowerLimit
ErrorAck	BOOL	FALSE	• Flanke FALSE -> TRUE ErrorBits und Warning werden zurückgesetzt.
Reset	BOOL	FALSE	Führt einen Neustart des Reglers durch. • Flanke FALSE -> TRUE – Wechsel in Betriebsart "Inaktiv" – ErrorBits und Warning werden zurückgesetzt. • Solange Reset = TRUE ist, – bleibt PID_Temp in der Betriebsart "Inaktiv" (State = 0). – können Sie die Betriebsart nicht über Mode und ModeActivate oder ManualEnable ändern – können Sie den Inbetriebnahmedialog nicht nutzen. • Flanke TRUE -> FALSE – Falls ManualEnable = FALSE, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist. – Falls Mode = 3 (Automatikbetrieb), wird der I-Anteil behandelt wie durch die Variable IntegralResetMode konfiguriert.
ModeActivate	BOOL	FALSE	• Flanke FALSE -> TRUE PID_Temp wechselt in die Betriebsart, die am Eingang Mode anliegt.

Ausgangsparameter PID_Temp

Parameter	Datentyp	Vorbelegung	Beschreibung
ScaledInput	REAL	0.0	Skalierter Istwert
OutputHeat	REAL	0.0	Ausgangswert Heizen im REAL-Format Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Heat.PidUpperLimit, Config.Output.Heat.UpperScaling und Config.Output.Heat.PidLowerLimit, Config.Output.Heat.LowerScaling skaliert und im REAL-Format an OutputHeat ausgegeben. OutputHeat wird immer berechnet.
OutputCool	REAL	0.0	Ausgangswert Kühlen im REAL-Format Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Cool.PidUpperLimit, Config.Output.Cool.LowerScaling und Config.Output.Cool.PidLowerLimit, Config.Output.Cool.UpperScaling skaliert und im REAL-Format an OutputCool ausgegeben. OutputCool wird nur berechnet, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE).
OutputHeat_PER	INT	0	Analoger Ausgangswert Heizen Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Heat.PidUpperLimit, Config.Output.Heat.PerUpperScaling und Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PerLowerScaling skaliert und als Analogwert an OutputHeat_PER ausgegeben. OutputHeat_PER wird nur berechnet, falls Config.Output.Heat.Select = 2.
OutputCool_PER	INT	0	Analoger Ausgangswert Kühlen Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Cool.PidUpperLimit, Config.Output.Cool.PerLowerScaling und Config.Output.Cool.PidLowerLimit, Config.Output.Cool.PerUpperScaling skaliert und als Analogwert an OutputCool_PER ausgegeben. OutputCool_PER wird nur berechnet, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Config.Output.Cool.Select = 2.
OutputHeat_PWM	BOOL	FALSE	Pulsweitenmodulierter Ausgangswert Heizen Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Heat.PidUpperLimit, Config.Output.Heat.PwmUpperScaling und Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PwmLowerScaling skaliert und als pulsweitenmodulierter Wert (variable Ein- und Ausschaltzeiten) an OutputHeat_PWM ausgegeben. OutputHeat_PWM wird nur berechnet, falls Config.Output.Heat.Select = 1.
OutputCool_PWM	BOOL	FALSE	Pulsweitenmodulierter Ausgangswert Kühlen Der PID Ausgangswert (PidOutputSum) wird über die beiden Wertepaare Config.Output.Cool.PidUpperLimit, Config.Output.Cool.PwmLowerScaling und Config.Output.Cool.PidLowerLimit, Config.Output.Cool.PwmUpperScaling skaliert und als pulsweitenmodulierter Wert (variable Ein- und Ausschaltzeiten) an OutputCool_PWM ausgegeben. OutputCool_PWM wird nur berechnet, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Config.Output.Cool.Select = 1.
SetpointLimit_H	BOOL	FALSE	Falls SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht (Setpoint ≥ Config.SetpointUpperLimit oder Setpoint ≥ Config.InputUpperLimit). Der Sollwert wird nach oben auf das Minimum aus Config.SetpointUpperLimit und Config.InputUpperLimit begrenzt.

Parameter	Datentyp	Vorbelegung	Beschreibung
SetpointLimit_L	BOOL	FALSE	Falls SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht (Setpoint ≤ Config.SetpointLowerLimit oder Setpoint ≤ Config.InputLowerLimit). Der Sollwert wird nach unten auf das Maximum aus Config.SetpointLowerLimit und Config.InputLowerLimit begrenzt.
InputWarning_H	BOOL	FALSE	Falls InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten (ScaledInput ≥ Config.InputUpperWarning).
InputWarning_L	BOOL	FALSE	Falls InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten (ScaledInput ≤ Config.InputLowerWarning).
State	INT	0	Der Parameter State und Mode PID_Temp (Seite 3077) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit dem Eingangsparameter Mode und einer steigenden Flanke an ModeActivate. Für Erstoptimierung und Nachoptimierung legen Sie mit Heat.EnableTuning und Cool.EnableTuning fest, ob die Optimierung für Heizen oder Kühlen ausgeführt wird. • State = 0: Inaktiv • State = 1: Erstoptimierung • State = 2: Nachoptimierung • State = 3: Automatikbetrieb • State = 4: Handbetrieb • State = 5: Ersatzausgangswert mit Fehlerüberwachung
Error	BOOL	FALSE	Falls Error = TRUE, liegt in diesem Takt mindestens eine Fehlermeldung vor.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits PID_Temp (Seite 3086) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.

Durchgangsparameter PID_Temp

Parameter	Datentyp	Vorbele-gung	Beschreibung
Mode	INT	4	An Mode geben Sie die Betriebsart vor, in die PID_Temp wechseln soll. Möglich sind: • Mode = 0: Inaktiv • Mode = 1: Erstoptimierung • Mode = 2: Nachoptimierung • Mode = 3: Automatikbetrieb • Mode = 4: Handbetrieb Die Betriebsart wird aktiviert durch: • Steigende Flanke an ModeActivate • Fallende Flanke an Reset • Fallende Flanke an ManualEnable • Kaltstart der CPU, falls RunModeByStartup = TRUE Für Erstoptimierung und Nachoptimierung legen Sie mit Heat.EnableTuning und Cool.EnableTuning fest, ob die Optimierung für Heizen oder Kühlen ausgeführt wird. Mode ist remanent. Eine detaillierte Beschreibung der Betriebsarten finden Sie unter Parameter State und Mode (Seite 3077).

Parameter	Datentyp	Vorbelegung	Beschreibung
Master	DWORD	DW#16#0	Schnittstelle für Kaskadenregelung Falls diese PID_Temp Instanz als Slave-Regler in einer Kaskade verwendet wird (Config.Cascade.IsSlave = TRUE), belegen Sie den Parameter Master am Anweisungsaufruf mit dem Parameter Slave des Master-Reglers. Beispiel: Aufruf eines Slave-Reglers "PID_Temp_2" mit Master-Regler "PID_Temp_1" in SCL: ---------- `"PID_Temp_2"(Master := "PID_Temp_1".Slave, Setpoint := "PID_Temp_1".OutputHeat);` ---------- Über diese Schnittstelle tauschen Slave-Regler Informationen über Betriebsart, Begrenzung und Ersatzsollwert mit ihrem Master-Regler aus. Beachten Sie, dass der Aufruf des Master-Reglers vor dem Aufruf des Slave-Reglers und im selben Weckalarm-OB erfolgen muss. Belegung: • Bit 0 bis 15: nicht belegt • Bit 16 bis 23 – Begrenzungszähler: Ein Slave-Regler, dessen Ausgangswert sich in Begrenzung befindet, erhöht diesen Zähler. In Abhängigkeit der konfigurierten Anzahl der Slaves (Config.Cascade.CountSlaves) und des Anti-Windup-Modus (Config.Cascade.AntiWindUpMode) reagiert der Master-Regler entsprechend. • Bit 24 – Automatikbetrieb der Slave-Regler: TRUE, falls sich alle Slave-Regler im Automatikbetrieb befinden • Bit 25 – Ersatzsollwert der Slave-Regler: TRUE, falls ein Slave-Regler den Ersatzsollwert aktiviert hat (SubstituteSetpointOn = TRUE)
Slave	DWORD	DW#16#0	Schnittstelle für Kaskadenregelung Über diese Schnittstelle tauschen Slave-Regler Informationen über Betriebsart, Begrenzung und Ersatzsollwert mit ihrem Master-Regler aus. Siehe Beschreibung zu Parameter Master

Siehe auch

Parameter State und Mode PID_Temp (Seite 3077)

Statische Variablen PID_Temp

Hinweis

Verändern Sie die mit (1) gekennzeichneten Variablen nur in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden.

Variable	Daten-typ	Vorbele-gung	Beschreibung
IntegralResetMode	Int	V1.0: 1, ab V1.1: 4	Die Variable IntegralResetMode (Seite 3094) bestimmt, wie der I-Anteil PIDCtrl.IOutputOld beim Wechsel der Betriebsart von "Inaktiv" in "Automatikbetrieb" vorbelegt wird. Diese Einstellung wirkt nur für einen Zyklus. • IntegralResetMode = 0: Glätten • IntegralResetMode = 1: Löschen • IntegralResetMode = 2: Halten • IntegralResetMode = 3: Vorbelegen • IntegralResetMode = 4: wie Sollwertänderung (nur für PID_Temp mit Version ≥ 1.1)
OverwriteInitialOutputValue	REAL	0.0	Falls eine der folgenden Bedingungen erfüllt ist, wird der Integral-Anteil PIDCtrl.IOutputOld automatisch so vorbelegt, als ob im vorherigen Zyklus PIDOutputSum = OverwriteInitialOutputValue gewesen wäre: • IntegralResetMode = 3 beim Wechsel von Betriebsart "Inaktiv" in "Automatikbetrieb • Flanke TRUE -> FALSE an Parameter Reset und Parameter Mode = 3 • PIDCtrl.PIDInit = TRUE im “Automatikbetrieb” (ab PID_Temp Version 1.1 verfügbar)
RunModeByStartup	BOOL	TRUE	Nach CPU Neustart Betriebsart an Mode aktivieren • Falls RunModeByStartup = TRUE, startet PID_Temp nach CPU-Anlauf in der Betriebsart, die an Mode gespeichert ist. • Falls RunModeByStartup = FALSE, bleibt PID_Temp nach CPU-Anlauf in der Betriebsart "Inaktiv".
LoadBackUp	BOOL	FALSE	Falls LoadBackUp = TRUE, wird der letzte Satz PID-Parameter aus der Struktur CtrlParamsBackUp wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt. Die Übernahme erfolgt stoßfrei.

Variable	Daten-typ	Vorbele-gung	Beschreibung
SetSubstituteOutput	BOOL	TRUE	Auswahl des Ausgangswerts, solange ein Fehler ansteht (State = 5): • Falls SetSubstituteOutput = TRUE und ActivateRecoverMode = TRUE sind, wird der konfigurierte Ersatzausgangswert SubstituteOutput als PID Ausgangswert ausgegeben, solange ein Fehler ansteht. • Falls SetSubstituteOutput = FALSE und ActivateRecoverMode = TRUE sind, bleibt das Stellglied auf dem aktuellen PID Ausgangswert, solange ein Fehler ansteht. • Falls ActivateRecoverMode = FALSE ist, wirkt sich SetSubstituteOutput nicht aus. • Falls SubstituteOutput ungültig ist (ErrorBits = 0020000h), kann der Ersatzausgangswert nicht ausgegeben werden. In diesem Fall wird die Untergrenze des PID Ausgangswerts für Heizen (Config.Output.Heat.PidLowerLimit) als PID Ausgangswert verwendet.
PhysicalUnit	INT	0	Physikalische Einheit des Ist- und Sollwerts, z. B. ºC oder ºF. Dieser Parameter dient der Anzeige in den Editoren und be-einflusst den Regelalgorithmus nicht.
PhysicalQuantity	INT	0	Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur. Dieser Parameter dient der Anzeige in den Editoren und be-einflusst den Regelalgorithmus nicht.
ActivateRecoverMode	BOOL	TRUE	Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall.
Warning	DWORD	0	Die Variable Warning zeigt die Warnungen seit Reset = TRUE oder ErrorAck =TRUE. Warning ist remanent.
Progress	REAL	0.0	Fortschritt der aktuellen Phase der Optimierung in Prozent (0.0 - 100.0)
CurrentSetpoint	REAL	0.0	CurrentSetpoint zeigt immer den aktuell wirksamen Sollwert an. Dieser Wert wird während der Optimierung eingefroren.
CancelTuningLevel	REAL	10.0	Zulässige Schwankung des Sollwerts während der Optimie-rung. Die Optimierung wird erst abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel

Variable	Datentyp	Vorbelegung	Beschreibung
SubstituteOutput	REAL	0.0	Der Ersatzausgangswert wird als PID Ausgangswert verwendet, solange folgende Bedingungen erfüllt sind: • im Automatikbetrieb stehen ein oder mehrere Fehler an, bei denen ActivateRecoverMode wirkt • SetSubstituteOutput = TRUE • ActivateRecoverMode = TRUE Die aus dem Ersatzausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung (Strukturen Config.Output.Heat und Config.Output.Cool). Für Regler mit aktiviertem Kühlausgang (Config.ActivateCooling = TRUE) geben Sie an: • einen positiven Ersatzausgangswert, um den Wert an den Ausgängen für Heizen auszugeben • einen negativen Ersatzausgangswert, um den Wert an den Ausgängen für Kühlen auszugeben Der zulässige Wertebereich ist abhängig von der Konfiguration. • Kühlausgang deaktiviert (Config.ActivateCooling = FALSE): Config.Output.Heat.PidUpperLimit ≥ SubstituteOutput ≥ Config.Output.Heat.PidLowerLimit • Kühlausgang aktiviert (Config.ActivateCooling = TRUE): Config.Output.Heat.PidUpperLimit ≥ SubstituteOutput ≥ Config.Output.Cool.PidLowerLimit
PidOutputSum	REAL	0.0	PID Ausgangswert PidOutputSum zeigt den Ausgangswert des PID-Algorithmus. Er wird je nach Betriebsart automatisch berechnet oder durch den Handwert oder den konfigurierten Ersatzausgangswert vorgegeben. Die aus dem PID Ausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung (Strukturen Config.Output.Heat und Config.Output.Cool). PidOutputSum wird abhängig von der Konfiguration begrenzt. • Kühlausgang deaktiviert (Config.ActivateCooling = FALSE): Config.Output.Heat.PidUpperLimit ≥ PidOutputSum ≥ Config.Output.Heat.PidLowerLimit • Kühlausgang aktiviert (ConfigActivateCooling = TRUE): Config.Output.Heat.PidUpperLimit ≥ PidOutputSum ≥ Config.Output.Cool.PidLowerLimit

Variable	Daten-typ	Vorbele-gung	Beschreibung
PidOutputOffsetHeat	REAL	0.0	Offset des PID Ausgangswerts Heizen PidOutputOffsetHeat wird zum Wert, der sich aus PidOutput-Sum für den Heizzweig ergibt, addiert. Geben Sie einen positiven Wert für PidOutputOffsetHeat vor, um einen positiven Offset an den Ausgängen für Heizen zu erhalten. Die resultierenden Werte an den Ausgängen für Heizen ergeben sich aus der konfigurierten Ausgangsskalierung (Struktur Config.Output.Heat). Dieser Offset kann für Stellglieder genutzt werden, die einen festen Mindestwert benötigen z. B. Lüfter mit einer Mindestdrehzahl.
PidOutputOffsetCool	REAL	0.0	Offset des PID Ausgangswerts Kühlen PidOutputOffsetCool wird zum Wert, der sich aus PidOutput-Sum für den Kühlzweig ergibt, addiert. Geben Sie einen negativen Wert für PidOutputOffsetCool vor, um einen positiven Offset an den Ausgängen für Kühlen zu erhalten. Die resultierenden Werte an den Ausgängen für Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung (Struktur Config.Output.Cool). Dieser Offset kann für Stellglieder genutzt werden, die einen festen Mindestwert benötigen z. B. Lüfter mit einer Mindestdrehzahl.
SubstituteSetpointOn	BOOL	FALSE	Aktiviert den Ersatzsollwert als Sollwert des Reglers. • FALSE = der Parameter Setpoint wird verwendet. • TRUE = der Parameter SubstituteSetpoint wird als Sollwert verwendet SubstituteSetpointOn kann genutzt werden, um den Sollwert eines Slave-Reglers in einer Kaskade direkt vorzugeben, ohne das Anwenderprogramm ändern zu müssen.
SubstituteSetpoint	REAL	0.0	Ersatzsollwert Falls SubstituteSetpointOn = TRUE, wird der Ersatzsollwert SubstituteSetpoint als Sollwert verwendet. Zulässiger Wertebereich: Config.SetpointUpperLimit ≥ SubstituteSetpoint ≥ Config.SetpointLowerLimit, Config.InputUpperLimit ≥ SubstituteSetpoint ≥ Config.InputLowerLimit
DisableCooling	BOOL	FALSE	DisableCooling = TRUE deaktiviert für Heizkühlregler (Config.ActivateCooling = TRUE) im Automatikbetrieb den Kühlzweig, indem PidOutputSum auf 0.0 als Untergrenze begrenzt wird. PidOutputOffsetCool und die Ausgangsskalierung für die Ausgänge für Kühlen bleiben aktiv. DisableCooling kann für die Optimierung von Mehrzonenanwendungen genutzt werden, um den Kühlzweig vorübergehend zu deaktivieren, solange noch nicht alle Regler ihre Optimierung abgeschlossen haben. Dieser Parameter wird vom Anwender manuell gesetzt/rückgesetzt und wird von der Anweisung PID_Temp nicht automatisch zurückgesetzt.

Variable	Datentyp	Vorbelegung	Beschreibung
AllSlaveAutomaticState	BOOL	FALSE	Wird diese PID_Temp Instanz als Master-Regler einer Kaskade genutzt (Config.Cascade.IsMaster = TRUE), zeigt AllSlaveAutomaticState = TRUE, dass sich alle Slave-Regler im Automatikbetrieb befinden. Optimierung, Handbetrieb oder Automatikbetrieb des Master-Reglers können nur richtig ausgeführt werden, falls sich alle Slave-Regler im Automatikbetrieb befinden. AllSlaveAutomaticState wird nur ermittelt, falls Sie Master-Regler und Slave-Regler über die Parameter Master und Slave verschalten. Details siehe Parameter Master.
NoSlaveSubstituteSetpoint	BOOL	FALSE	Wird diese PID_Temp Instanz als Master-Regler einer Kaskade genutzt (Config.Cascade.IsMaster = TRUE), zeigt NoSlaveSubstituteSetpoint = TRUE, dass kein Slave-Regler seinen Ersatzsollwert aktiviert hat. Optimierung, Handbetrieb oder Automatikbetrieb des Master-Reglers können nur richtig ausgeführt werden,falls kein Slave-Regler seinen Ersatzsollwert aktiviert hat. NoSlaveSubstituteSetpoint wird nur ermittelt, falls Sie Master-Regler und Slave-Regler über die Parameter Master und Slave verschalten. Details siehe Parameter Master.
Heat.EnableTuning	BOOL	TRUE	Freigabe der Optimierung für Heizen Heat.EnableTuning muss für folgende Optimierungen gesetzt werden (gleichzeitig mit oder vor dem Start mit Mode und ModeActivate): • Erstoptimierung Heizen • Erstoptimierung Heizen und Kühlen • Nachoptimierung Heizen Dieser Parameter wird von der Anweisung PID_Temp nicht automatisch zurückgesetzt.
Cool.EnableTuning	BOOL	FALSE	Freigabe der Optimierung für Kühlen Cool.EnableTuning muss für folgende Optimierungen gesetzt werden (gleichzeitig mit oder vor dem Start mit Mode und ModeActivate): • Erstoptimierung Kühlen • Erstoptimierung Heizen und Kühlen • Nachoptimierung Kühlen Nur wirksam, falls der Kühlausgang und die PID-Parameterumschalung aktiviert sind ("Config.ActivateCooling" = TRUE und "Config.AdvancedCooling" = TRUE). Dieser Parameter wird von der Anweisung PID_Temp nicht automatisch zurückgesetzt.
Config.InputPerOn(1)	BOOL	TRUE	Falls InputPerOn = TRUE, wird der Parameter Input_PER für die Erfassung des Istwerts verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.InputUpperLimit(1)	REAL	120.0	Obergrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze überschritten, wird ein Fehler ausgegeben und abhängig von ActivateRecoverMode reagiert. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Nennbereich (Übersteuerungsbereich) liegen. Bei Verwendung des Peripherie-Eingangs mit der Voreinstellung für Obergrenze und Istwertskalierung ist ein Überschreiten der Grenze daher nicht möglich. Bei Start einer Erstoptimierung wird anhand der Differenz zwischen Ober- und Untergrenze des Istwerts überprüft, ob der Abstand zwischen Sollwert und Istwert die nötigen Voraussetzungen erfüllt. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit(1)	REAL	0.0	Untergrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze unterschritten, wird ein Fehler ausgegeben und abhängig von ActivateRecoverMode reagiert. InputLowerLimit < InputUpperLimit
Config.InputUpperWarning(1)	REAL	3.402822e +38	Obere Warngrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze überschritten, wird eine Warnung an Parameter Warning ausgegeben. • Falls Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als obere Warngrenze verwendet. • Falls Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet. InputUpperWarning > InputLowerWarning
Config.InputLowerWarning(1)	REAL	-3.402822e +38	Untere Warngrenze des Istwerts Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze unterschritten, wird eine Warnung an Parameter Warning ausgegeben. • Falls Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als untere Warngrenze verwendet. • Falls Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet. InputLowerWarning < InputUpperWarning

Variable	Daten-typ	Vorbele-gung	Beschreibung
Config.SetpointUpperLimit(1)	REAL	3.402822e +38	Obergrenze des Sollwerts Setpoint und SubstituteSetpoint werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze überschritten, wird eine Warnung an Parameter Warning ausgegeben. • Falls Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert verwendet. • Falls Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet. SetpointUpperLimit > SetpointLowerLimit
Config.SetpointLowerLimit(1)	REAL	-3.402822e +38	Untergrenze des Sollwerts Setpoint und SubstituteSetpoint werden auf die Einhaltung dieser Grenze überwacht. Wird die Grenze unterschritten, wird eine Warnung an Parameter Warning ausgegeben. • Falls Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert verwendet. • Falls Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet. SetpointLowerLimit < SetpointUpperLimit
Config.ActivateCooling(1)	BOOL	FALSE	Kühlausgang aktivieren • Config.ActivateCooling = FALSE Es werden nur die Ausgänge für Heizen verwendet. • Config.ActivateCooling = TRUE Es werden die Ausgänge für Heizen und Kühlen verwendet. Falls Sie den Kühlausgang verwenden, darf der Regler nicht als Master-Regler konfiguriert sein (Config.Cascade.IsMaster muss FALSE sein) .

Variable	Daten-typ	Vorbele-gung	Beschreibung
Config.AdvancedCooling(1)	BOOL	TRUE	Methode für Heizen/Kühlen • Kühlfaktor (Config.AdvancedCooling = FALSE) Die Ausgangswertberechnung für Kühlen erfolgt mit dem PID-Parameter für Heizen (Struktur Retain.CtrlParams.Heat) unter Berücksichtigung des konfigurierbaren Kühlfaktors Config.CoolFactor. Diese Methode ist geeignet, falls Heiz- und Kühlstellglied ähnliches Zeitverhalten aber unterschiedliche Verstärkungen aufweisen. Bei dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen nicht zur Verfügung. Es können nur die Optimierungen für Heizen ausgeführt werden. • PID-Parameterumschaltung (Config.AdvancedCooling = TRUE) Die Ausgangswertberechnung für Kühlen erfolgt über einen eigenen PID-Parametersatz (Struktur Retain.CtrlParams.Cool). Diese Methode ist geeignet, falls Heiz- und Kühlstellglied unterschiedliches Zeitverhalten und Verstärkungen aufweisen. Nur bei dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen zur Verfügung (Mode = 1 oder 2, Cool.EnableTuning = TRUE). Config.AdvancedCooling ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE).
Config.CoolFactor(1)	REAL	1.0	Kühlfaktor Falls Config.AdvancedCooling = FALSE, wird Config.CoolFactor als Faktor in der Berechnung des Ausgangswerts für Kühlen berücksichtigt. Auf diese Weise können unterschiedliche Verstärkungen von Heiz- und Kühlstellglied berücksichtigt werden. Config.CoolFactor wird nicht automatisch eingestellt oder während der Optimierung angepasst. Sie müssen Config.CoolFactor manuell mit dem Verhältnis "Verstärkung Heizstellglied / Verstärkung Kühlstellglied" korrekt konfigurieren. Beispiel: Config.CoolFactor = 2.0 bedeutet, dass die Verstärkung des Heizstellglieds doppelt so stark wie die Verstärkung des Kühlstellglieds ist. Config.CoolFactor ist nur wirksam, falls der Kühlausgang aktiviert (Config.ActivateCooling = TRUE) und Kühlfaktor als Methode für Heizen/Kühlen gewählt ist (Config.AdvancedCooling = FALSE). Config.CoolFactor > 0.0

Variable	Datentyp	Vorbelegung	Beschreibung
Config.InputScaling.UpperPointIn(1)	REAL	27648.0	Skalierung Input_PER Oben Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER skaliert. Nur wirksam, falls Input_PER für die Istwerterfasung verwendet wird (Config.InputPerOn = TRUE). UpperPointIn > LowerPointIn
Config.InputScaling.LowerPointIn(1)	REAL	0.0	Skalierung Input_PER Unten Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER skaliert. Nur wirksam, falls Input_PER für die Istwerterfasung verwendet wird (Config.InputPerOn = TRUE). LowerPointIn < UpperPointIn
Config.InputScaling.UpperPointOut(1)	REAL	100.0	Skalierter oberer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER skaliert. Nur wirksam, falls Input_PER für die Istwerterfasung verwendet wird (Config.InputPerOn = TRUE). UpperPointOut > LowerPointOut
Config.InputScaling.LowerPointOut(1)	REAL	0.0	Skalierter unterer Istwert Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER skaliert. Nur wirksam, falls Input_PER für die Istwerterfasung verwendet wird (Config.InputPerOn = TRUE). LowerPointOut < UpperPointOut
Config.Output.Heat.Select(1)	INT	1	Auswahl des Ausgangswerts für Heizen Config.Output.Heat.Select gibt vor, welche Ausgänge für Heizen verwendet werden: • Heat.Select = 0 - OutputHeat wird verwendet • Heat.Select = 1 - OutputHeat und OutputHeat_PWM werden verwendet • Heat.Select = 2 - OutputHeat und OutputHeat_PER werden verwendet Nicht verwendete Ausgänge werden nicht berechnet und verbleiben auf dem Default-Wert.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Heat.PwmPeriode(1)	REAL	0.0	Periodendauer der Pulsweitenmodulation (PWM) für Heizen (Ausgang OutputHeat_PWM) in Sekunden: • Heat.PwmPeriode = 0.0 Die Abtastzeit des PID-Algorithmus für Heizen (Retain.CtrlParams.Heat.Cycle) wird als Periodendauer der PWM verwendet. • Heat.PwmPeriode > 0.0 Der Wert wird auf ein ganzzahliges Vielfaches der Abtastzeit PID_Temp (CycleTime.Value) gerundet und als Periodendauer der PWM verwendet. Mit dieser Einstellung kann bei großer Abtastzeit des PID-Algorithmus die Glattheit des Istwerts verbessert werden. Der Wert muss folgende Bedingungen erfüllen: – Heat.PwmPeriode ≤ Retain.CtrlParams.Heat.Cycle, – Heat.PwmPeriode > Config.Output.Heat.MinimumOnTime – Heat.PwmPeriode > Config.Output.Heat.MinimumOffTime
Config.Output.Heat.PidUpperLimit(1)	REAL	100.0	Obergrenze des PID Ausgangswerts für Heizen Der PID Ausgangswert (PidOutputSum) wird auf diese Obergrenze begrenzt. Heat.PidUpperLimit bildet zusammen mit folgenden Parametern ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf die Ausgänge für Heizen: • Heat.UpperScaling für OutputHeat • Heat.PwmUpperScaling für OutputHeat_PWM • Heat.PerUpperScaling für OutputHeat_PER Falls Sie den Wert am zugehörigen Ausgang begrenzen wollen, müssen Sie auch diese Skalierungswerte anpassen. Heat.PidUpperLimit > Heat.PidLowerLimit

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Heat.PidLowerLimit(1)	REAL	0.0	Untergrenze des PID Ausgangswerts für Heizen Bei Reglern mit **deaktiviertem** Kühlausgang (Config.ActivateCooling = FALSE) wird der PID Ausgangswert (PidOutputSum) auf diese Untergrenze begrenzt. Bei Reglern mit **aktiviertem** Kühlausgang (Config.ActivateCooling = TRUE) muss der Wert 0.0 sein. Heat.PidLowerLimit bildet zusammen mit folgenden Parametern ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf die Ausgänge für Heizen: • Heat.LowerScaling für OutputHeat • Heat.PwmLowerScaling für OutputHeat_PWM • Heat.PerLowerScaling für OutputHeat_PER Falls Sie den Wert am zugehörigen Ausgang begrenzen wollen, müssen Sie auch diese Skalierungswerte anpassen. Der zulässige Wertebereich ist abhängig von der Konfiguration. • Kühlausgang deaktiviert (Config.ActivateCooling = FALSE): Heat.PidLowerLimit < Heat.PidUpperLimit • Kühlausgang aktiviert (Config.ActivateCooling = TRUE): Heat.PidLowerLimit = 0.0
Config.Output.Heat.UpperScaling(1)	REAL	100.0	Skalierter oberer Ausgangswert für Heizen Heat.UpperScaling und Heat.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den Ausgangswert für Heizen (OutputHeat). Der Wert von OutputHeat liegt immer zwischen Heat.UpperScaling und Heat.LowerScaling. Heat.UpperScaling ≠ Heat.LowerScaling
Config.Output.Heat.LowerScaling(1)	REAL	0.0	Skalierter unterer Ausgangswert für Heizen Heat.LowerScaling und Heat.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den Ausgangswert für Heizen (OutputHeat). Der Wert von OutputHeat liegt immer zwischen Heat.UpperScaling und Heat.LowerScaling. Heat.UpperScaling ≠ Heat.LowerScaling
Config.Output.Heat.PwmUpperScaling(1)	REAL	100.0	Skalierter oberer PWM Ausgangswert für Heizen Heat.PwmUpperScaling und Heat.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den pulsweitenmodulierten Ausgangswert für Heizen (OutputHeat_PWM). Der Wert von OutputHeat_PWM liegt immer zwischen Heat.PwmUpperScaling und Heat.PWMLowerScaling. Heat.PwmUpperScaling ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PWM gewählt ist (Heat.Select = 1) 100.0 ≥ Heat.PwmUpperScaling ≥ 0.0 Heat.PwmUpperScaling ≠ Heat.PwmLowerScaling

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Heat.PwmLowerScaling(1)	REAL	0.0	Skalierter unterer PWM Ausgangswert für Heizen Heat.PwmLowerScaling und Heat.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den pulsweitenmodulierten Ausgangswert für Heizen (OutputHeat_PWM). Der Wert von OutputHeat_PWM liegt immer zwischen Heat.PwmUpperScaling und Heat.PwmLowerScaling. Heat.PwmLowerScaling ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PWM gewählt ist (Heat.Select = 1) 100.0 ≥ Heat.PwmLowerScaling ≥ 0.0 Heat.PwmUpperScaling ≠ Heat.PwmLowerScaling
Config.Output.Heat.PerUpperScaling(1)	REAL	27648.0	Skalierter oberer analoger Ausgangswert für Heizen Heat.PerUpperScaling und Heat.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den analogen Ausgangswert für Heizen (OutputHeat_PER). Der Wert von OutputHeat_PER liegt immer zwischen Heat.PerUpperScaling und Heat.PerLowerScaling. Heat.PerUpperScaling ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PER gewählt ist (Heat.Select = 2) 32511.0 ≥ Heat.PerUpperScaling ≥ -32512.0 Heat.PerUpperScaling ≠ Heat.PerLowerScaling
Config.Output.Heat.PerLowerScaling(1)	REAL	0.0	Skalierter unterer analoger Ausgangswert für Heizen Heat.PerLowerScaling und Heat.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den analogen Ausgangswert für Heizen (OutputHeat_PER). Der Wert von OutputHeat_PER liegt immer zwischen Heat.PerUpperScaling und Heat.PerLowerScaling. Heat.PerLowerScaling ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PER gewählt ist (Heat.Select = 2) 32511.0 ≥ Heat.PerLowerScaling ≥ -32512.0 Heat.PerUpperScaling ≠ Heat.PerLowerScaling
Config.Output.Heat.MinimumOnTime(1)	REAL	0.0	Minimale Einschaltzeit der Pulsweitenmodulation für Heizen (Ausgang OutputHeat_PWM) Ein PWM-Impuls ist nie kürzer als dieser Wert. Der Wert wird gerundet auf: Heat.MinimumOnTime = n × CycleTime.Value Heat.MinimumOnTime ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PWM gewählt ist (Heat.Select = 1)". 100000.0 ≥ Heat.MinimumOnTime ≥ 0.0

Variable	Daten-typ	Vorbele-gung	Beschreibung
Config.Output.Heat.MinimumOff-Time(1)	REAL	0.0	Minimale Ausschaltzeit der Pulsweitenmodulation für Heizen (Ausgang OutputHeat_PWM) Eine PWM-Pause ist nie kürzer als dieser Wert. Der Wert wird gerundet auf: Heat.MinimumOffTime = n × CycleTime.Value Heat.MinimumOffTime ist nur wirksam, falls als Ausgang für Heizen OutputHeat_PWM gewählt ist (Heat.Select = 1)". 100000.0 ≥ Heat.MinimumOffTime ≥ 0.0
Config.Output.Cool.Select(1)	INT	1	Auswahl des Ausgangswerts für Kühlen Config.Output.Cool.Select gibt vor, welche Ausgänge für Kühlen verwendet werden: • Cool.Select = 0 - OutputCool wird verwendet • Cool.Select = 1 - OutputCool und OutputCool_PWM werden verwendet • Cool.Select = 2 - OutputCool und OutputCool_PER werden verwendet Nicht verwendete Ausgänge werden nicht berechnet und verbleiben auf dem Default-Wert. Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE).

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Cool.PwmPeriode(1)	REAL	0.0	Periodendauer der Pulsweitenmodulation für Kühlen (Ausgang OutputCool_PWM) in Sekunden: • Cool.PwmPeriode = 0.0 und Config.AdvancedCooling = FALSE: Die Abtastzeit des PID-Algorithmus für Heizen (Retain.CtrlParams.Heat.Cycle) wird als Periodendauer der PWM verwendet. • Cool.PwmPeriode = 0.0 und Config.AdvancedCooling = TRUE: Die Abtastzeit des PID-Algorithmus für Kühlen (Retain.CtrlParams.Cool.Cycle) wird als Periodendauer der PWM verwendet. • Cool.PwmPeriode > 0.0: Der Wert wird auf ein ganzzahliges Vielfaches der Abtastzeit PID_Temp (CycleTime.Value) gerundet und als Periodendauer der PWM verwendet. Mit dieser Einstellung kann bei großer Abtastzeit des PID-Algorithmus die Glattheit des Istwerts verbessert werden. Der Wert muss folgende Bedingungen erfüllen: – Cool.PwmPeriode ≤ Retain.CtrlParams.Cool.Cycle bzw. Retain.CtrlParams.Heat.Cycle – Cool.PwmPeriode > Config.Output.Cool.MinimumOnTime – Cool.PwmPeriode > Config.Output.Cool.MinimumOffTime Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE).
Config.Output.Cool.PidUpperLimit(1)	REAL	0.0	Obergrenze des PID Ausgangswerts für Kühlen Der Wert muss 0.0 sein. Cool.PidUpperLimit bildet zusammen mit folgenden Parametern ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf die Ausgänge für Kühlen: • Cool.LowerScaling für OutputCool • Cool.PwmLowerScaling für OutputCool_PWM • Cool.PerLowerScaling für OutputCool_PER Sie müssen auch diese Skalierungswerte anpassen, falls Sie den Wert an dem zugehörigen Ausgang begrenzen wollen. Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). Cool.PidUpperLimit = 0.0

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Cool.PidLowerLimit(1)	REAL	-100.0	Untergrenze des PID Ausgangswerts für Kühlen Bei Reglern mit aktiviertem Kühlausgang (Config.ActivateCooling = TRUE) wird der PID Ausgangswert (PidOutputSum) auf diese Untergrenze begrenzt. Cool.PidLowerLimit bildet zusammen mit folgenden Parametern ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf die Ausgänge für Kühlen: • Cool.UpperScaling für OutputCool • Cool.PwmUpperScaling für OutputCool_PWM • Cool.PerUpperScaling für OutputCool_PER Sie müssen auch diese Skalierungswerte anpassen, falls Sie den Wert an dem zugehörigen Ausgang begrenzen wollen. Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). Cool.PidLowerLimit < Cool.PidUpperLimit
Config.Output.Cool.UpperScaling(1)	REAL	100.0	Skalierter oberer Ausgangswert für Kühlen Cool.UpperScaling und Cool.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den Ausgangswert Kühlen (OutputCool). Der Wert von OutputCool liegt immer zwischen Cool.UpperScaling und Cool.LowerScaling. Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). Cool.UpperScaling ≠ Cool.LowerScaling
Config.Output.Cool.LowerScaling(1)	REAL	0.0	Skalierter unterer Ausgangswert für Kühlen Cool.LowerScaling und Cool.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den Ausgangswert Kühlen (OutputCool). Der Wert von OutputCool liegt immer zwischen Cool.UpperScaling und Cool.LowerScaling. Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). Cool.UpperScaling ≠ Cool.LowerScaling
Config.Output.Cool.PwmUpperScaling(1)	REAL	100.0	Skalierter oberer PWM Ausgangswert für Kühlen Cool.PwmUpperScaling und Cool.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den pulsweitenmodulierten Ausgangswert Kühlen (OutputCool_PWM). Der Wert von OutputCool_PWM liegt immer zwischen Cool.PwmUpperScaling und Cool.PwmLowerScaling. Cool.PwmUpperScaling ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Sie als Ausgang für Kühlen OutputCool_PWM gewählt haben (Cool.Select = 1). 100.0 ≥ Cool.PwmUpperScaling ≥ 0.0 Cool.PwmUpperScaling ≠ Cool.PwmLowerScaling

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Cool.PwmLowerScaling(1)	REAL	0.0	Skalierter unterer PWM Ausgangswert für Kühlen Cool.PwmLowerScaling und Cool.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den pulsweitenmodulierten Ausgangswert Kühlen (OutputCool_PWM). Der Wert von OutputCool_PWM liegt immer zwischen Cool.PwmUpperScaling und CoolPwm.LowerScaling. Cool.PwmLowerScaling ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Sie als Ausgang für Kühlen OutputCool_PWM gewählt haben (Cool.Select = 1). 100.0 ≥ Cool.PwmLowerScaling ≥ 0.0 Cool.PwmUpperScaling ≠ Cool.PwmLowerScaling
Config.Output.Cool.PerUpperScaling(1)	REAL	27648.0	Skalierter oberer analoger Ausgangswert für Kühlen Cool.PerUpperScaling und Cool.PidLowerLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den analogen Ausgangswert Kühlen (OutputCool_PER). Der Wert von OutputCool_PER liegt immer zwischen Cool.PerUpperScaling und Cool.PerLowerScaling. Cool.PerUpperScaling ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Sie als Ausgang für Kühlen OutputCool_PER gewählt haben (Cool.Select = 2). 32511.0 ≥ Cool.PerUpperScaling ≥ -32512.0 Cool.PerUpperScaling ≠ Cool.PerLowerScaling
Config.Output.Cool.PerLowerScaling(1)	REAL	0.0	Skalierter unterer analoger Ausgangswert für Kühlen Cool.PerLowerScaling und Cool.PidUpperLimit bilden ein Wertepaar für die Skalierung des PID Ausgangswerts (PidOutputSum) auf den analogen Ausgangswert Kühlen (OutputCool_PER). Der Wert von OutputCool_PER liegt immer zwischen Cool.PerUpperScaling und Cool.PerLowerScaling. Cool.PerLowerScaling ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE) und Sie als Ausgang für Kühlen OutputCool_PER gewählt haben (Cool.Select = 2). 32511.0 ≥ Cool.PerLowerScaling ≥ -32512.0 Cool.PerUpperScaling ≠ Cool.PerLowerScaling

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Output.Cool.MinimumOnTime[1]	REAL	0.0	Minimale Einschaltzeit der Pulsweitenmodulation für Kühlen (Ausgang OutputCool_PWM) Ein PWM-Impuls ist nie kürzer als dieser Wert. Der Wert wird gerundet auf: Cool.MinimumOnTime = n × CycleTime.Value Cool.MinimumOnTime ist nur wirksam, falls als Ausgang für Kühlen OutputCool_PWM gewählt ist (Cool.Select = 1). Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). 100000.0 ≥ Cool.MinimumOnTime ≥ 0.0
Config.Output.Cool.MinimumOffTime[1]	REAL	0.0	Minimale Ausschaltzeit der Pulsweitenmodulation für Kühlen (Ausgang OutputCool_PWM) Eine PWM-Pause ist nie kürzer als dieser Wert. Der Wert wird gerundet auf: Cool.MinimumOffTime = n × CycleTime.Value Cool.MinimumOffTime ist nur wirksam, falls als Ausgang für Kühlen OutputCool_PWM gewählt ist (Cool.Select = 1). Nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling = TRUE). 100000.0 ≥ Cool.MinimumOffTime ≥ 0.0
Falls Sie PID_Temp in einer Kaskade verwenden, tauschen Master-Regler und Slave-Regler über die Parameter Master und Slave Informationen aus. Die Verschaltung muss von Ihnen vorgenommen werden. Details siehe Parameter Master.			
Config.Cascade.IsMaster[1]	BOOL	FALSE	Der Regler ist Master in einer Kaskade und stellt den Sollwert für den Slave bereit. Setzen Sie IsMaster = TRUE, falls Sie diese PID_Temp Instanz als Master-Regler in einer Kaskade verwenden. Ein Master-Regler gibt mit seinem Ausgang den Sollwert eines Slave-Reglers vor. Eine PID_Temp Instanz kann gleichzeitig Master-Regler und Slave-Regler sein. Falls der Regler als Master-Regler verwendet wird, muss der Kühlausgang deaktiviert sein (Config.ActivateCooling = FALSE).
Config.Cascade.IsSlave[1]	BOOL	FALSE	Der Regler ist Slave in einer Kaskade und erhält seinen Sollwert vom Master. Setzen Sie IsSlave = TRUE, falls Sie diese PID_Temp Instanz als Slave-Regler in einer Kaskade verwenden. Ein Slave-Regler erhält seinen Sollwert (Parameter Setpoint) vom Ausgang seines Master-Reglers (Parameter OutputHeat). Eine PID_Temp Instanz kann gleichzeitig Master-Regler und Slave-Regler sein.

Variable	Datentyp	Vorbelegung	Beschreibung
Config.Cascade.AntiWindUpMode(1)	INT	1	Anti-Wind-Up-Verhalten in der Kaskade Möglich sind: • Anti-Windup = 0 Die AntiWindUp Funktionalität ist deaktiviert. Der Master-Regler reagiert nicht auf die Begrenzung seiner Slave-Regler. • Anti-Windup = 1 Der I-Anteil des Master-Reglers wird im Verhältnis "Slaves in Begrenzung" zu "Anzahl der Slaves" reduziert (Parameter “CountSlaves”). Dadurch werden die Auswirkungen der Begrenzung auf das Regelverhalten verringert. • Anti-Windup = 2 Der I-Anteil des Master-Reglers wird gehalten, sobald sich ein Slave-Regler in der Begrenzung befindet. Nur wirksam, falls der Regler als Master-Regler konfiguriert ist (Config.Cascade.IsMaster = TRUE).
Config.Cascade.CountSlaves(1)	INT	1	Anzahl der untergeordneten Slaves Geben Sie hier die Anzahl direkt untergeordneter Slave-Regler an, die ihren Sollwert von diesem Master-Regler erhalten. Nur wirksam, falls der Regler als Master-Regler konfiguriert ist (Config.Cascade.IsMaster = TRUE). 255 ≥ CountSlaves ≥ 1
CycleTime.StartEstimation	BOOL	TRUE	Falls CycleTime.EnEstimation = TRUE, startet CycleTime.StartEstimation = TRUE die automatische Ermittlung der Abtastzeit PID_Temp (Zykluszeit des aufrufenden OBs). Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE gesetzt.
CycleTime.EnEstimation	BOOL	TRUE	Falls CycleTime.EnEstimation = TRUE, wird die Abtastzeit PID_Temp automatisch ermittelt. Falls CycleTime.EnEstimation = FALSE, wird die Abtastzeit PID_Temp nicht automatisch ermittelt und Sie müssen CycleTime.Value manuell korrekt konfigurieren.
CycleTime.EnMonitoring	BOOL	TRUE	Falls CycleTime.EnMonitoring = FALSE, wird die Abtastzeit PID_Temp nicht überwacht. Falls PID_Temp nicht innerhalb der Abtastzeit ausgeführt werden kann, wird kein Fehler (ErrorBits=0000800h) ausgegeben und PID_Temp reagiert nicht wie mit ActivateRecoverMode konfiguriert.
CycleTime.Value(1)	REAL	0.1	Abtastzeit PID_Temp (Zykluszeit des aufrufenden OBs) in Sekunden CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.
Werte aus der Struktur CtrlParamsBackUp können mit LoadBackUp = TRUE wieder geladen werden.			
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Gespeicherter Wert von Retain.CtrlParams.SetByUser
CtrlParamsBackUp.Heat.Gain	REAL	1.0	Gespeicherte Proportionalverstärkung für Heizen
CtrlParamsBackUp.Heat.Ti	REAL	20.0	Gespeicherte Integrationszeit für Heizen in Sekunden
CtrlParamsBackUp.Heat.Td	REAL	0.0	Gespeicherte Differenzierzeit für Heizen in Sekunden

Variable	Datentyp	Vorbelegung	Beschreibung
CtrlParamsBackUp.Heat.TdFiltRatio	REAL	0.2	Gespeicherter Koeffizient Differenzierverzug für Heizen
CtrlParamsBackUp.Heat.PWeighting	REAL	1.0	Gespeicherte Gewichtung des P-Anteils für Heizen
CtrlParamsBackUp.Heat.DWeighting	REAL	1.0	Gespeicherte Gewichtung des D-Anteils für Heizen
CtrlParamsBackUp.Heat.Cycle	REAL	1.0	Gespeicherte Abtastzeit des PID-Algorithmus für Heizen in Sekunden
CtrlParamsBackUp.Heat.ControlZone	REAL	3.402822e+38	Gespeicherte Regelzonenbreite für Heizen
CtrlParamsBackUp.Heat.DeadZone	REAL	0.0	Gespeicherte Totzonenbreite für Heizen
CtrlParamsBackUp.Cool.Gain	REAL	1.0	Gespeicherte Proportionalverstärkung für Kühlen
CtrlParamsBackUp.Cool.Ti	REAL	20.0	Gespeicherte Integrationszeit für Kühlen in Sekunden
CtrlParamsBackUp.Cool.Td	REAL	0.0	Gespeicherte Differenzierzeit für Kühlen in Sekunden
CtrlParamsBackUp.Cool.TdFiltRatio	REAL	0.2	Gespeicherter Koeffizient Differenzierverzug für Kühlen
CtrlParamsBackUp.Cool.PWeighting	REAL	1.0	Gespeicherter Gewichtungsfaktor des P-Anteils für Kühlen
CtrlParamsBackUp.Cool.DWeighting	REAL	1.0	Gespeicherter Gewichtungsfaktor des D-Anteils für Kühlen
CtrlParamsBackUp.Cool.Cycle	REAL	1.0	Gespeicherte Abtastzeit des PID-Algorithmus für Kühlen in Sekunden
CtrlParamsBackUp.Cool.ControlZone	REAL	3.402822e+38	Gespeicherte Regelzonenbreite für Kühlen
CtrlParamsBackUp.Cool.DeadZone	REAL	0.0	Gespeicherte Totzonenbreite für Kühlen
PIDSelfTune.SUT.CalculateParamsHeat	BOOL	FALSE	Die Eigenschaften des Heizzweigs der Regelstrecke werden bei der Erstoptimierung Heizen gespeichert. Falls SUT.CalculateParamsHeat = TRUE, werden anhand dieser Eigenschaften die PID Parameter für Heizen (Struktur Retain.CtrlParams.Heat) neu berechnet. Dadurch kann die Methode für die Parameterberechnung (Parameter PIDSelfTune.SUT.TuneRuleHeat) geändert werden, ohne die Optimierung zu wiederholen. SUT.CalculateParamsHeat wird nach der Berechnung auf FALSE gesetzt. Nur möglich, falls die Erstoptimierung erfolgreich war (SUT.ProcParHeatOk = TRUE).

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.SUT.CalculateParamsCool	BOOL	FALSE	Die Eigenschaften des Kühlzweigs der Regelstrecke werden bei der Optimierung Kühlen gespeichert. Falls SUT.CalculateParamsCool = TRUE, werden anhand dieser Eigenschaften die PID Parameter für Kühlen (Struktur Retain.CtrlParams.Cool) neu berechnet. Dadurch kann die Methode für die Parameterberechnung (Parameter PIDSelfTune.SUT.TuneRuleCool) geändert werden, ohne die Optimierung zu wiederholen. SUT.CalculateParamsCool wird nach der Berechnung auf FALSE gesetzt. Nur möglich, falls die Erstoptimierung erfolgreich war (SUT.ProcParCoolOk = TRUE). Nur wirksam, falls Config.ActivateCooling = TRUE und Config.AdvancedCooling = TRUE.
PIDSelfTune.SUT.TuneRuleHeat	INT	2	Methode für die PID Parameterberechnung bei Erstoptimierung Heizen Möglich sind: • SUT.TuneRuleHeat = 0: PID nach CHR • SUT.TuneRuleHeat = 1: PI nach CHR • SUT.TuneRuleHeat = 2: PID für Temperaturprozesse nach CHR (führt zu einem langsameren und eher asymptotischen Regelverhalten mit geringerem Überschwingen als SUT.TuneRuleHeat = 0) (CHR = Chien, Hrones und Reswick) Nur bei SUT.TuneRuleHeat = 2 wird die Regelzone Retain.CtrlParams.Heat.ControlZone automatisch bei der Erstoptimierung Heizen eingestellt.
PIDSelfTune.SUT.TuneRuleCool	INT	2	Methode für die PID Parameterberechnung bei Erstoptimierung Kühlen Möglich sind: • SUT.TuneRuleCool = 0: PID nach CHR • SUT.TuneRuleCool = 1: PI nach CHR • SUT.TuneRuleCool = 2: PID für Temperaturprozesse nach CHR (führt zu einem langsameren und eher asymptotischen Regelverhalten mit geringerem Überschwingen als SUT.TuneRuleCool = 0) (CHR = Chien, Hrones und Reswick) Nur bei SUT.TuneRuleCool = 2 wird die Regelzone Retain.CtrlParams.Cool.ControlZone automatisch bei der Erstoptimierung Kühlen eingestellt. SUT.TuneRuleCool ist nur wirksam, falls der Kühlausgang und die PID-Parameterumschalung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE).

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.SUT.State	INT	0	Die Variable SUT.State zeigt die aktuelle Phase der Erstoptimierung: • State = 0: Erstoptimierung initialisieren • State = 100: Standardabweichung Heizen berechnen • State = 200: Standardabweichung Kühlen berechnen • State = 300: Wendepunkt Heizen ermitteln • State = 400: Wendepunkt Kühlen ermitteln • State = 500: Nach Erreichen des Wendepunkts Heizen auf Sollwert heizen • State = 600: Nach Erreichen des Wendepunkts Kühlen auf Sollwert heizen • State = 700: Wirkung von Heizstellglied und Kühlstellglied vergleichen • State = 800: Heizen und Kühlen aktiviert • State = 900: Kühlen aktiviert • State = 1000: Verzugszeit nach Abschaltung des Heizens ermitteln • State = 9900: Erstoptimierung erfolgreich • State = 1: Erstoptimierung nicht erfolgreich
PIDSelfTune.SUT.ProcParHeatOk	BOOL	FALSE	TRUE: Die Berechnung der Prozessparameter für die Erstoptimierung Heizen war erfolgreich. Diese Variable wird während der Optimierung gesetzt. Sie muss für die Berechnung der PID-Parameter für Heizen TRUE sein.
PIDSelfTune.SUT.ProcParCoolOk	BOOL	FALSE	TRUE: Die Berechnung der Prozessparameter für die Erstoptimierung Kühlen war erfolgreich. Diese Variable wird während der Optimierung gesetzt. Sie muss für die Berechnung der PID-Parameter für Kühlen TRUE sein.

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.SUT.AdaptDelay-Time	INT	0	Die Variable AdaptDelayTime bestimmt die Anpassung der Verzugszeit für Heizen am Arbeitspunkt (für "Erstoptimierung Heizen" und "Erstoptimierung Heizen und Kühlen"). Möglich sind: • SUT.AdaptDelayTime = 0: Keine Anpassung der Verzugszeit. Die Phase SUT.State = 1000 wird übersprungen. Diese Option führt zu einer kürzeren Dauer der Optimierung als mit SUT.AdaptDelayTime = 1. • SUT.AdaptDelayTime = 1: Anpassung der Verzögerungszeit an den Sollwert in Phase SUT.State = 1000 durch vorübergehendes Ausschalten des Heizens. Diese Option führt zu einer längeren Dauer der Optimierung als mit SUT.AdaptDelayTime = 0. Sie kann das Regelverhalten verbessern, falls das Prozessverhalten stark vom Arbeitspunkt abhängig ist (Nichtlinearität). Diese Option sollte für Mehrzonenanwendungen mit starken thermischen Kopplungen nicht verwendet werden.
PIDSelfTune.SUT.CoolingMode	INT	0	Die Variable CoolingMode bestimmt die Stellwertausgabe zur Ermittlung der Kühlparameter (bei Erstoptimierung Heizen und Kühlen). Möglich sind: • SUT.CoolingMode = 0: Heizen ausschalten und Kühlen einschalten nach Erreichen des Sollwerts. Phase SUT.State = 700 wird übersprungen. Auf Phase SUT.State = 500 folgt Phase SUT.State = 900. Diese Option kann das Regelverhalten verbessern, falls die Verstärkung des Kühlstellglieds gegenüber der Verstärkung des Heizstellgliedes gering ist. Sie führt zu einer kürzeren Dauer der Optimierung als mit SUT.CoolingMode = 1 oder 2. • SUT.CoolingMode = 1: Kühlen zusätzlich zum Heizen einschalten nach Erreichen des Sollwerts. Phase SUT.State = 700 wird übersprungen. Auf Phase SUT.State = 500 folgt Phase SUT.State = 800. Diese Option kann das Regelverhalten verbessern, falls die Verstärkung des Kühlstellglieds gegenüber der Verstärkung des Heizstellgliedes groß ist. • SUT.CoolingMode = 2: Nach Aufheizen zum Sollwert wird in Phase SUT.State = 700 automatisch entschieden, ob die Heizung ausgeschaltet wird. Auf Phase SUT.State = 500 folgt Phase SUT.State = 700 und anschließend SUT.State = 800 oder SUT.State = 900. Diese Option erfordert mehr Zeit, als die Optionen 0 und 1.

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Mit der Variable RunIn können Sie den Ablauf der Nachoptimierung beim Start aus dem Automatikbetrieb festlegen. • RunIn = FALSE Falls die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt (TIR.State = 500 oder 600). Erst dann startet die Nachoptimierung. • RunIn = TRUE PID_Temp versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen (TIR.State = 300 oder 400). Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet anschließend automatisch. RunIn wird nach der Nachoptimierung auf FALSE gesetzt. Beim Start der Nachoptimierung aus Inaktiv oder Handbetrieb verhält sich PID_Temp wie unter RunIn = TRUE beschrieben.
PIDSelfTune.TIR.CalculateParamsHeat	BOOL	FALSE	Die Eigenschaften des Heizzweigs der Regelstrecke werden bei Nachoptimierung Heizen gespeichert. Falls TIR.CalculateParamsHeat = TRUE, werden anhand dieser Eigenschaften die PID Parameter für Heizen (Struktur Retain.CtrlParams.Heat) neu berechnet. Dadurch kann die Methode für die Parameterberechnung (Parameter PIDSelfTune.TIR.TuneRuleHeat) geändert werden, ohne die Optimierung zu wiederholen. TIR.CalculateParamsHeat wird nach der Berechnung auf FALSE gesetzt. Nur möglich, falls vorher die Nachoptimierung Heizen erfolgreich war (TIR.ProcParHeatOk = TRUE).
PIDSelfTune.TIR.CalculateParamsCool	BOOL	FALSE	Die Eigenschaften des Kühlzweigs der Regelstrecke werden bei Nachoptimierung Kühlen gespeichert. Falls TIR.CalculateParamsCool = TRUE, werden anhand dieser Eigenschaften die PID Parameter für Kühlen (Struktur Retain.CtrlParams.Cool) neu berechnet. Dadurch kann die Methode für die Parameterberechnung (Parameter PIDSelfTune.TIR.TuneRuleCool) geändert werden, ohne die Optimierung zu wiederholen. TIR.CalculateParamsCool wird nach der Berechnung auf FALSE gesetzt. Nur möglich, falls vorher die Nachoptimierung Kühlen erfolgreich war (TIR.ProcParCoolOk = TRUE). Nur wirksam, falls Config.ActivateCooling = TRUE und Config.AdvancedCooling = TRUE

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.TIR.TuneRuleHeat	INT	0	Methode für die Parameterberechnung während Nachoptimierung Heizen Möglich sind: • TIR.TuneRuleHeat = 0: PID automatisch • TIR.TuneRuleHeat = 1: PID schnell (schnelleres Regelverhalten mit höheren Amplituden des Ausgangswerts als mit TIR.TuneRuleHeat = 2) • TIR.TuneRuleHeat = 2: PID langsam (langsameres Regelverhalten mit geringeren Amplituden des Ausgangswerts als mit TIR.TuneRuleHeat = 1) • TIR.TuneRuleHeat = 3: ZN PID • TIR.TuneRuleHeat = 4: ZN PI • TIR.TuneRuleHeat = 5: ZN P (ZN=Ziegler-Nichols) Um die Berechnung der PID-Parameter für Heizen mit TIR.CalculateParamsHeat und TIR.TuneRuleHeat = 0, 1 oder 2 wiederholen zu können, muss auch die vorhergehende Nachoptimierung mit TIR.TuneRuleHeat = 0, 1 oder 2 ausgeführt worden sein. Ist dies nicht der Fall, wird TIR.TuneRuleHeat = 3 verwendet. Die erneute Berechnung der PID-Parameter für Heizen mit TIR.CalculateParamsHeat und TIR.TuneRuleHeat = 3, 4 oder 5 ist immer möglich.

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.TIR.TuneRuleCool	INT	0	Methode für die Parameterberechnung während Nachoptimierung Kühlen Möglich sind: • TIR.TuneRuleCool = 0: PID automatisch • TIR.TuneRuleCool = 1: PID schnell (schnelleres Regelverhalten mit höheren Amplituden des Ausgangswerts als mit TIR.TuneRuleCool = 2) • TIR.TuneRuleCool = 2: PID langsam (langsameres Regelverhalten mit geringeren Amplituden des Ausgangswerts als mit TIR.TuneRuleCool = 1) • TIR.TuneRuleCool = 3: ZN PID • TIR.TuneRuleCool = 4: ZN PI • TIR.TuneRuleCool = 5: ZN P (ZN=Ziegler-Nichols) Um die Berechnung der PID-Parameter für Kühlen mit TIR.CalculateParamsCool und TIR.TuneRuleCool = 0, 1 oder 2 wiederholen zu können, muss auch die vorhergehende Nachoptimierung mit TIR.TuneRuleCool = 0, 1 oder 2 ausgeführt worden sein. Ist dies nicht der Fall, wird TIR.TuneRuleCool = 3 verwendet. Die erneute Berechnung der PID-Parameter für Kühlen mit TIR.CalculateParamsCool und TIR.TuneRuleCool = 3, 4 oder 5 ist immer möglich. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschalung aktiviert sind (ConfigActivateCooling = TRUE und Config.AdvancedCooling = TRUE).

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.TIR.State	INT	0	Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung": • State = 0: Nachoptimierung initialisieren • State = 100: Standardabweichung Heizen berechnen • State = 200: Standardabweichung Kühlen berechnen • State = 300: Versuchen Sollwert Heizen mit 2-Punkt-Regelung Heizen zu erreichen • State = 400: Versuchen Sollwert Kühlen mit 2-Punkt-Regelung Kühlen zu erreichen • State = 500: Versuchen Sollwert Heizen mit PID-Regelung zu erreichen • State = 600: Versuchen Sollwert Kühlen mit PID-Regelung zu erreichen • State = 700: Standardabweichung Heizen berechnen • State = 800: Standardabweichung Kühlen berechnen • State = 900: Oszillation ermitteln und Parameter für Heizen berechnen • State = 1000: Oszillation ermitteln und Parameter für Kühlen berechnen • State = 9900: Nachoptimierung erfolgreich • State = 1: Nachoptimierung nicht erfolgreich
PIDSelfTune.TIR.ProcParHeatOk	BOOL	FALSE	TRUE: Die Berechnung der Prozessparameter für die Nachoptimierung Heizen war erfolgreich. Diese Variable wird während der Optimierung gesetzt. Sie muss für die Berechnung der PID-Parameter für Heizen erfüllt sein.
PIDSelfTune.TIR.ProcParCoolOk	BOOL	FALSE	TRUE: Die Berechnung der Prozessparameter für die Nachoptimierung Kühlen war erfolgreich. Diese Variable wird während der Optimierung gesetzt. Sie muss für die Berechnung der PID-Parameter für Kühlen erfüllt sein.

Variable	Daten-typ	Vorbele-gung	Beschreibung
PIDSelfTune.TIR.OutputOffse-tHeat	REAL	0.0	Optimierungsoffset Heizen des PID Ausgangswerts TIR.OutputOffsetHeat wird zum Wert addiert, der sich aus PidOutputSum für den Heizzweig ergibt. Geben Sie, um einen positiven Offset an den Ausgängen für Heizen zu erhalten, einen positiven Wert für TIR.OutputOffsetHeat vor. Die resultierenden Werte an den Ausgängen für Heizen ergeben sich aus der konfigurierten Ausgangsskalierung (Struktur Config.Output.Heat). Dieser Optimierungsoffset kann bei Reglern mit aktiviertem Kühlausgang und PID-Parameterumschaltung (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) für die Nachoptimierung Kühlen genutzt werden. Falls die Ausgänge für Kühlen am Sollwert, an dem die Optimierung durchgeführt werden soll, nicht aktiv sind (PidOutputSum > 0.0), ist die Nachoptimierung Kühlen nicht möglich. Geben Sie in diesem Fall vor dem Start der Optimierung einen positiven Optimierungsoffset Heizen vor, der größer ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand. Dadurch werden die Werte an den Ausgängen für Heizen vergrößert und die Ausgänge für Kühlen aktiv (PidOutputSum < 0.0). Damit ist die Nachoptimierung Kühlen möglich. Wenn die Nachoptimierung beendet ist, wird TIR.OutputOffsetHeat auf 0.0 zurückgesetzt. Große Änderungen an TIR.OutputOffsetHeat in einem Schritt können zu vorübergehenden Überschwingern führen. Config.Output.Heat.PidUpperLimit ≥ PIDSelfTune.TIR.OutputOffsetHeat ≥ Config.Output.Heat.PidLowerLimit

Variable	Datentyp	Vorbelegung	Beschreibung
PIDSelfTune.TIR.OutputOffset-Cool	REAL	0.0	Optimierungsoffset Kühlen des PID Ausgangswerts TIR.OutputOffsetCool wird zum Wert addiert, der sich aus PidOutputSum für den Kühlzweig ergibt. Geben Sie, um einen positiven Offset an den Ausgängen für Kühlen zu erhalten, einen negativen Wert für TIR.OutputOffsetCool vor. Die resultierenden Werte an den Ausgängen für Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung (Struktur Config.Output.Coool). Dieser Optimierungsoffset kann bei Reglern mit aktiviertem Kühlausgang (Config.ActivateCooling = TRUE) für die Nachoptimierung Heizen genutzt werden. Falls die Ausgänge für Heizen am Sollwert, an dem die Optimierung durchgeführt werden soll, nicht aktiv sind (PidOutputSum < 0.0), ist die Nachoptimierung Heizen nicht möglich. Geben Sie in diesem Fall vor dem Start der Optimierung einen negativen Optimierungsoffset Kühlen vor, der kleiner ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand. Dadurch werden die Werte an den Ausgängen für Kühlen vergrößert und die Ausgänge für Heizen aktiv (PidOutputSum > 0.0). Damit ist die Nachoptimierung Heizen möglich. Wenn die Nachoptimierung beendet ist, wird TIR.OutputOffsetCool auf 0.0 zurückgesetzt. Große Änderungen an TIR.OutputOffsetCool in einem Schritt können zu vorübergehenden Überschwingern führen. Config.Output.Cool.PidUpperLimit ≥ PIDSelfTune.TIR.OutputOffsetCool ≥ Config.Output.Cool.PidLowerLimit
PIDSelfTune.TIR.WaitForControlIn	BOOL	FALSE	Warten bei Nachoptimierung nach Erreichen des Sollwerts Falls TIR.WaitForControlIn = TRUE, wird bei Nachoptimierung zwischen Erreichen des Sollwerts (TIR.State = 500 oder 600) und Berechnung der Standardabweichung (TIR.State = 700 oder 800) gewartet, bis an TIR.FinishControlIn eine Flanke FALSE -> TRUE vorgegeben wird. TIR.WaitForControlIn kann bei gleichzeitiger Nachoptimierung mehrerer Regler in Mehrzonenanwendungen genutzt werden, um die Optimierungen der einzelnen Zonen zu synchronisieren. Damit kann sichergestellt werden, dass alle Zonen ihre Sollwerte erreicht haben, bevor die eigentliche Optimierung startet. Auf diese Weise wird der Einfluss thermischer Kopplungen zwischen den Zonen auf die Optimierung verringert. TIR.WaitForControlIn ist nur wirksam, falls die Nachoptimierung aus dem Automatikbetrieb heraus mit PIDSelfTune.TIR.RunIn = FALSE gestartet wird.
PIDSelfTune.TIR.ControlInReady	BOOL	FALSE	Falls TIR.WaitForControlIn = TRUE, setzt PID_Temp TIR.ControlInReady = TRUE, sobald der Sollwert erreicht ist und wartet mit weiteren Optimierungsschritten bis eine Flanke FALSE -> TRUE an TIR.FinishControlIn vorgegeben wird.
PIDSelfTune.TIR.FinishControlIn	BOOL	FALSE	Falls TIR.ControlInReady = TRUE, beendet eine Flanke FALSE -> TRUE an TIR.FinishControlIn das Warten und die Nachoptimierung wird fortgeführt.

Variable	Datentyp	Vorbelegung	Beschreibung
PIDCtrl.IOutputOld(1)	REAL	0.0	I-Anteil im letzten Zyklus
PIDCtrl.PIDInit	BOOL	FALSE	PIDCtrl.PIDInit ist ab PID_Temp Version 1.1 verfügbar. Falls im "Automatikbetrieb" PIDCtrl.PIDInit = TRUE, wird der Integral-Anteil PIDCtrl.IOutputOld automatisch so vorbelegt, als ob im vorherigen Zyklus PidOutputSum = OverwriteInitialOutputValue gewesen wäre. Dies kann für eine AUTOHOTSPOT genutzt werden.
Retain.CtrlParams.SetByUser(1)	BOOL	FALSE	Werden die PID Parameter im Konfigurationseditor von Hand eingegeben, wird SetByUser = TRUE gesetzt. Dieser Parameter dient der Anzeige in den Editoren und beeinflusst den Regelalgorithmus nicht. SetByUser ist remanent.
Retain.CtrlParams.Heat.Gain(1)	REAL	1.0	Aktive Proportionalverstärkung für Heizen Heat.Gain ist remanent. Heat.Gain ≥ 0.0
Retain..CtrlParams.Heat.Ti(1)	REAL	20.0	Aktive Integrationszeit für Heizen in Sekunden Mit Heat.CtrlParams.Ti = 0.0 ist der I-Anteil für Heizen ausgeschaltet. Heat.Ti ist remanent. 100000.0 ≥ Heat.Ti ≥ 0.0
Retain.CtrlParams.Heat.Td(1)	REAL	0.0	Aktive Differenzierzeit für Heizen in Sekunden Mit Heat.CtrlParams.Td = 0.0 ist der D-Anteil für Heizen ausgeschaltet. Heat.Td ist remanent. 100000.0 ≥ Heat.Td ≥ 0.0
Retain.CtrlParams.Heat.TdFiltRatio(1)	REAL	0.2	Aktiver Koeffizient Differenzierverzug für Heizen Die Wirkung des D-Anteils wird durch den Koeffizienten Differenzierverzug verzögert. Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug • 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam. • 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt. • > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert. Heat.TdFiltRatio ist remanent. Heat.TdFiltRatio ≥ 0.0

Variable	Daten-typ	Vorbele-gung	Beschreibung
Retain.CtrlParams.Heat.PWeighting(1)	REAL	1.0	Aktive Gewichtung des P-Anteils für Heizen Sie können bei Sollwertänderungen den P-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: P-Anteil bei Sollwertänderung voll wirksam • 0.0: P-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam. Heat.PWeighting ist remanent. 1.0 ≥ Heat.PWeighting ≥ 0.0
Retain.CtrlParams.Heat.DWeighting(1)	REAL	1.0	Aktive Gewichtung des D-Anteils für Heizen Sie können bei Sollwertänderungen den D-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: D-Anteil bei Sollwertänderung voll wirksam • 0.0: D-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam. Heat.DWeighting ist remanent. 1.0 ≥ Heat.DWeighting ≥ 0.0
Retain.CtrlParams.Heat.Cycle(1)	REAL	1.0	Aktive Abtastzeit des PID-Algorithmus für Heizen in Sekunden CtrlParams.Heat.Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von CycleTime.Value aufgerundet. Falls Config.Output.Heat.PwmPeriode = 0.0, wird Heat.Cycle als Periodendauer der Pulsweitenmodulation für Heizen verwendet. Falls Config.Output.Cool.PwmPeriode = 0.0 und Config.AdvancedCooling = FALSE, wird Heat.Cycle als Periodendauer der Pulsweitenmodulation für Kühlen verwendet. Heat.Cycle ist remanent. 100000.0 ≥ Heat.Cycle > 0.0

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.CtrlParams.Heat.ControlZone(1)	REAL	3.402822e+38	Aktive Regelzonenbreite für Heizen Mit Heat.ControlZone = 3.402822e+38 ist die Regelzone für Heizen ausgeschaltet. Heat.ControlZone wird nur während Erstoptimierung Heizen oder Erstoptimierung Heizen und Kühlen automatisch eingestellt, falls als Methode der Parameterberechnung PIDSelfTune.SUT.TuneRuleHeat = 2 gewählt ist. Für Regler mit deaktiviertem Kühlausgang (Config.ActivateCooling = FALSE) oder Regler mit aktiviertem Kühlausgang und Kühlfaktor (Config.AdvancedCooling = FALSE) liegt die Regelzone symmetrisch zwischen Setpoint – Heat.ControlZone und Setpoint + Heat.ControlZone. Für Regler mit aktiviertem Kühlausgang und PID-Parameterumschaltung (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) liegt die Regelzone zwischen Setpoint – Heat.ControlZone und Setpoint + Cool.ControlZone. Heat.ControlZone ist remanent. Heat.ControlZone > 0.0
Retain.CtrlParams.Heat.DeadZone(1)	REAL	0.0	Aktive Totzonenbreite für Heizen (siehe AUTOHOTSPOT) Mit Heat.DeadZone = 0.0 ist die Totzone für Heizen ausgeschaltet. Heat.DeadZone wird nicht automatisch eingestellt oder während der Optimierung angepasst. Sie müssen Heat.DeadZone manuell korrekt konfigurieren. Bei eingeschalteter Totzone kann sich eine dauerhafte Regeldifferenz (Abweichung zwischen Sollwert und Istwert) einstellen. Dies kann sich negativ auf die Durchführung einer Nachoptimierung auswirken. Für Regler mit deaktiviertem Kühlausgang (Config.ActivateCooling = FALSE) oder Regler mit aktiviertem Kühlausgang und Kühlfaktor (Config.AdvancedCooling = FALSE) liegt die Totzone symmetrisch zwischen Setpoint – Heat.DeadZone und Setpoint + Heat.DeadZone. Für Regler mit aktiviertem Kühlausgang und PID-Parameterumschaltung (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) liegt die Totzone zwischen Setpoint – Heat.DeadZone und Setpoint + Cool.DeadZone. Heat.DeadZone ist remanent. Heat.DeadZone ≥ 0.0
Retain.CtrlParams.Cool.Gain(1)	REAL	1.0	Aktive Proportionalverstärkung für Kühlen Cool.Gain ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.Gain ≥ 0.0

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.CtrlParams.Cool.Ti(1)	REAL	20.0	Aktive Integrationszeit für Kühlen in Sekunden Mit Cool.CtrlParams.Ti = 0.0 ist der I-Anteil für Kühlen ausgeschaltet. Cool.Ti ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 100000.0 ≥ Cool.Ti ≥ 0.0
Retain.CtrlParams.Cool.Td(1)	REAL	0.0	Aktive Differenzierzeit für Kühlen in Sekunden Mit Cool.CtrlParams.Td = 0.0 ist der D-Anteil für Kühlen ausgeschaltet. Cool.Td ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 100000.0 ≥ Cool.Td ≥ 0.0
Retain.CtrlParams.Cool.TdFiltRatio(1)	REAL	0.2	Aktiver Koeffizient Differenzierverzug für Kühlen Die Wirkung des D-Anteils wird durch den Koeffizienten Differenzierverzug verzögert. Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug • 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam. • 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit einer dominierenden Zeitkonstanten bewährt. • > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert. Cool.TdFiltRatio ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.TdFiltRatio ≥ 0.0
Retain.CtrlParams.Cool.PWeighting(1)	REAL	1.0	Aktive Gewichtung des P-Anteils für Kühlen Sie können bei Sollwertänderungen den P-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: P-Anteil bei Sollwertänderung voll wirksam • 0.0: P-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam. Cool.PWeighting ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 1.0 ≥ Cool.PWeighting ≥ 0.0

Variable	Daten-typ	Vorbele-gung	Beschreibung
Retain.CtrlParams.Cool.DWeighting[(1)]	REAL	1.0	Aktive Gewichtung des D-Anteils für Kühlen Sie können bei Sollwertänderungen den D-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0. • 1.0: D-Anteil bei Sollwertänderung voll wirksam • 0.0: D-Anteil bei Sollwertänderung nicht wirksam Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam. Cool.DWeighting ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 1.0 ≥ Cool.DWeighting ≥ 0.0
Retain.CtrlParams.Cool.Cycle[(1)]	REAL	1.0	Aktive Abtastzeit des PID-Algorithmus für Kühlen in Sekunden CtrlParams.Cool.Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von CycleTime.Value aufgerundet. Falls Config.Output.Cool.PwmPeriode = 0.0 und Config.AdvancedCooling = TRUE, wird Cool.Cycle als Periodendauer der Pulsweitenmodulation für Kühlen verwendet. Falls Config.Output.Cool.PwmPeriode = 0.0 und Config.AdvancedCooling = FALSE, wird Heat.Cycle als Periodendauer der Pulsweitenmodulation für Kühlen verwendet. Cool.Cycle ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). 100000.0 ≥ Cool.Cycle > 0.0

Variable	Datentyp	Vorbelegung	Beschreibung
Retain.CtrlParams.Cool.ControlZone(1)	REAL	3.402822e+38	Aktive Regelzonenbreite für Kühlen Mit Cool.ControlZone = 3.402822e+38 ist die Regelzone für Kühlen ausgeschaltet. Cool.ControlZone wird nur während Erstoptimierung Kühlen oder Erstoptimierung Heizen und Kühlen automatisch eingestellt, falls als Methode der Parameterberechnung PIDSelfTune.SUT.TuneRuleCool = 2 gewählt ist. Cool.ControlZone ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.ControlZone > 0.0
Retain.CtrlParams.Cool.DeadZone(1)	REAL	0.0	Aktive Totzonenbreite für Kühlen (siehe AUTOHOTSPOT) Mit Cool.DeadZone = 0.0 ist die Totzone für Kühlen ausgeschaltet. Cool.DeadZone wird nicht automatisch eingestellt oder während der Optimierung angepasst. Sie müssen Cool.DeadZone manuell korrekt konfigurieren. Bei eingeschalteter Totzone kann sich eine dauerhafte Regeldifferenz (Abweichung zwischen Sollwert und Istwert) einstellen. Dies kann sich negativ auf die Durchführung einer Nachoptimierung auswirken. Cool.DeadZone ist remanent. Nur wirksam, falls der Kühlausgang und die PID-Parameterumschaltung aktiviert sind (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE). Cool.DeadZone ≥ 0.0

Siehe auch

Variable ActivateRecoverMode PID_Temp (Seite 3088)

Variable Warning PID_Temp (Seite 3091)

Parameter State und Mode PID_Temp

Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Mit einer steigenden Flanke an ModeActivate wechselt PID_Temp in die Betriebsart, die am Durchgangsparameter Mode gespeichert ist.

Heat.EnableTuning und Cool.EnableTuning legen für Erstoptimierung und Nachoptimierung fest, ob die Optimierung für Heizen oder Kühlen durchgeführt wird.

Falls die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_Temp in der Betriebsart, die an Mode gespeichert ist. Um PID_Temp in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

Bedeutung der Werte

State / Mode	Beschreibung der Betriebsart
0	**Inaktiv** In der Betriebsart "Inaktiv" werden folgende Ausgangswerte ausgegeben: • 0.0 als PID Ausgangswert (PidOutputSum) • 0.0 als Ausgangswert für Heizen (OutputHeat) und Ausgangswert für Kühlen (OutputCool) • 0 als analoger Ausgangswert für Heizen (OutputHeat_PER) und analoger Ausgangswert für Kühlen (OutputCool_PER) • FALSE als PWM Ausgangswert für Heizen (OutputHeat_PWM) und PWM Ausgangswert für Kühlen (OutputCool_PWM) Dies ist unabhängig von den konfigurierten Ausgangswertgrenzen und –skalierung in den Strukturen Config.Output.Heat und Config.Output.Cool.

State / Mode	Beschreibung der Betriebsart
1	**Erstoptimierung** Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_Temp bietet abhängig von der Konfiguration verschiedene Erstoptimierungsarten: • Erstoptimierung Heizen: Es wird ein Sprung am Ausgangswert Heizen ausgegeben, die PID-Parameter für Heizen werden berechnet (Struktur Retain.CtrlParams.Heat) und anschließend wird im Automatikbetrieb auf den Sollwert geregelt. Ist das Prozessverhalten stark vom Arbeitspunkt abhängig, kann mit PIDSelfTune.SUT.AdaptDelayTime eine Anpassung der Verzugszeit am Sollwert aktiviert werden. • Erstoptimierung Heizen und Kühlen: Es wird ein Sprung am Ausgangswert Heizen ausgegeben. Sobald sich der Istwert in der Nähe des Sollwerts befindet, wird ein Sprung am Ausgangswert Kühlen ausgegeben. Die PID-Parameter für Heizen (Struktur Retain.CtrlParams.Heat) und Kühlen (Struktur Retain.CtrlParams.Cool) werden berechnet. Anschließend wird im Automatikbetrieb auf den Sollwert geregelt. Ist das Prozessverhalten stark vom Arbeitspunkt abhängig, kann mit PIDSelfTune.SUT.AdaptDelayTime eine Anpassung der Verzugszeit am Sollwert aktiviert werden. Abhängig von der Wirkung des Kühlstellgliedes im Vergleich zum Heizstellglied kann die Qualität der Optimierung beeinflusst werden, indem während der Optimierung Heiz- und Kühlausgang gleichzeitig betrieben werden oder nicht. Dies können Sie mit PIDSelfTune.SUT.CoolingMode festlegen. • Erstoptimierung Kühlen: Es wird ein Sprung am Ausgangswert Kühlen ausgegeben und die PID-Parameter für Kühlen werden berechnet (Struktur Retain.CtrlParams.Cool). Anschließend wird im Automatikbetrieb auf den Sollwert geregelt. Falls Sie die PID-Parameter für Heizen und Kühlen optimieren wollen, können Sie bei Durchführung einer "Erstoptimierung Heizen" und anschließender "Erstoptimierung Kühlen" ein besseres Regelverhalten erwarten als bei Durchführung einer "Erstoptimierung Heizen und Kühlen". Die Durchführung der Erstoptimierung in zwei Schritten erfordert jedoch mehr Zeit. **Allgemeine Voraussetzungen für die Erstoptimierung:** • Die Anweisung PID_Temp wird in einem Weckalarm-OB aufgerufen. • Betriebsart Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3) • ManualEnable = FALSE • Reset = FALSE • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. **Voraussetzungen für Erstoptimierung Heizen:** • Heat.EnableTuning = TRUE • Cool.EnableTuning = FALSE • Der Istwert darf nicht zu nah am Sollwert sein. |Setpoint - Input| > 0.3 * |Config.InputUpperLimit - Config.InputLowerLimit| und |Setpoint - Input| > 0.5 * |Setpoint| • Der Sollwert ist größer als der Istwert.

State / Mode	Beschreibung der Betriebsart
	Setpoint > Input **Voraussetzungen für Erstoptimierung Heizen und Kühlen:** • Heat.EnableTuning = TRUE· • Cool.EnableTuning = TRUE • Der Kühlausgang ist aktiviert (Config.ActivateCooling = TRUE).· • Die PID-Parameterumschaltung ist aktiviert (Config.AdvancedCooling = TRUE). • Der Istwert darf nicht zu nah am Sollwert sein. |Setpoint - Input| > 0.3 * |Config.InputUpperLimit - Config.InputLowerLimit| und |Setpoint - Input| > 0.5 * |Setpoint| • Der Sollwert ist größer als der Istwert. Setpoint > Input **Voraussetzungen für Erstoptimierung Kühlen:** • Heat.EnableTuning = FALSE· • Cool.EnableTuning = TRUE· • Der Kühlausgang ist aktiviert (Config.ActivateCooling = TRUE). • Die PID-Parameterumschaltung ist aktiviert (Config.AdvancedCooling = TRUE). • Eine "Erstoptimierung Heizen" oder "Erstoptimierung Heizen und Kühlen" wurde erfolgreich durchgeführt (PIDSelfTune.SUT.ProcParHeatOk = TRUE), möglichst am gleichen Sollwert. • Der Istwert muss nah am Sollwert sein. |Setpoint - Input| < 0.05 * |Config.InputUpperLimit - Config.InputLowerLimit| Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Die Methode zur Berechnung der PID-Parameter kann mit PIDSelfTune.SUT.TuneRuleHeat und PIDSelfTune.SUT.TuneRuleCool für Heizen und Kühlen getrennt festgelegt werden. Bevor die PID-Parameter neu berechnet werden, werden die PID-Parameter in der Struktur CtrlParamsBackUp gesichert und können mit LoadBackUp reaktiviert werden. Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet. Nach erfolgloser Erstoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode. Die Phase der Erstoptimierung wird mit PIDSelfTune.SUT.State angezeigt.

State / Mode	Beschreibung der Betriebsart
2	**Nachoptimierung** Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwerts. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_Temp versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. PID_Temp bietet abhängig von der Konfiguration verschiedene Nachoptimierungsarten: • Nachoptimierung Heizen: PID_Temp erzeugt mit periodischen Änderungen am Ausgangswert Heizen eine Schwingung des Istwerts und berechnet die PID-Parameter für Heizen (Struktur Retain.CtrlParams.Heat). • Nachoptimierung Kühlen: PID_Temp erzeugt mit periodischen Änderungen am Ausgangswert Kühlen eine Schwingung des Istwerts und berechnet die PID-Parameter für Kühlen (Struktur Retain.CtrlParams.Cool). **Vorübergehender Optimierungsoffset für Heizkühlregler** Wird PID_Temp als Heizkühlregler eingesetzt (Config.ActivateCooling = TRUE) muss der PID Ausgangswert (PidOutputSum) am Sollwert folgende Voraussetzung erfüllen, damit eine Istwertschwingung erzeugt und die Nachoptimierung erfolgreich durchgeführt werden kann: • Positiver PID Ausgangswert für Nachoptimierung Heizen • Negativer PID Ausgangswert für Nachoptimierung Kühlen Ist diese Voraussetzung nicht erfüllt, können Sie einen vorübergehenden Offset für die Nachoptimierung vorgeben, der am entgegengesetzt wirkenden Ausgang ausgegeben wird: • Offset für Kühlausgang (PIDSelfTune.TIR.OutputOffsetCool) bei Nachoptimierung Heizen. Geben Sie vor dem Start der Optimierung einen negativen Optimierungsoffset Kühlen vor, der kleiner ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand. • Offset für Heizausgang (PIDSelfTune.TIR.OutputOffsetHeat) bei Nachoptimierung Kühlen. Geben Sie vor dem Start der Optimierung einen positiven Optimierungsoffset Heizen vor, der größer ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand. Der vorgegebene Offset wird dann vom PID-Algorithmus ausgeglichen, sodass der Istwert am Sollwert verbleibt. Durch die Höhe des Offsets kann somit der PID Ausgangswert entsprechend angepasst werden, damit er oben genannte Voraussetzung erfüllt. Um größere Überschwinger des Istwerts bei Vorgabe des Offsets zu vermeiden, kann dieser auch in mehreren Schritten erhöht werden. Verlässt PID_Temp die Betriebsart Nachoptimierung, wird der Optimierungsoffset zurückgesetzt. Beispiel für Vorgabe eines Offsets für Nachoptimierung Kühlen: • Ohne Offset: – Sollwert (Setpoint) = Istwert (ScaledInput) = 80°C – PID Ausgangswert (PidOutputSum) = 30.0 – Ausgangswert Heizen (OutputHeat) = 30.0 – Ausgangswert Kühlen (OutputCool) = 0.0 Eine Schwingung des Istwerts um den Sollwert kann mit dem Kühlausgang alleine nicht erzeugt werden. Die Nachoptimierung würde hier fehlschlagen. • Mit Vorgabe eines Offsets für Heizausgang (PIDSelfTune.TIR.OutputOffsetHeat) = 80.0 – Sollwert (Setpoint) = Istwert (ScaledInput) = 80°C

State / Mode	Beschreibung der Betriebsart
	– PID Ausgangswert (PidOutputSum) = -50.0 – Ausgangswert Heizen (OutputHeat) = 80.0 – Ausgangswert Kühlen (OutputCool) = -50.0 Durch die Vorgabe eines Offsets für den Heizausgang kann der Kühlausgang nun eine Schwingung des Istwerts um den Sollwert erzeugen. Die Nachoptimierung kann somit erfolgreich durchgeführt werden. **Allgemeine Voraussetzungen für Nachoptimierung:** • Die Anweisung PID_Temp wird in einem Weckalarm-OB aufgerufen. • Es werden keine Störungen erwartet. • Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen. • Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht. Bei eingeschalteter Totzone kann sich eine dauerhafte Regeldifferenz (Abweichung zwischen Sollwert und Istwert) einstellen. Dies kann sich negativ auf die Durchführung der Nachoptimierung auswirken. • ManualEnable = FALSE • Reset = FALSE • Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4) **Voraussetzungen für Nachoptimierung Heizen:** • Heat.EnableTuning = TRUE • Cool.EnableTuning = FALSE • Wenn PID_Temp als Heizkühlregler konfiguriert ist (Config.ActivateCooling = TRUE), muss am Arbeitspunkt, an dem die Optimierung durchgeführt werden soll, der Heizausgang aktiv sein (PidOutputSum > 0.0 (siehe Optimierungsoffset)). **Voraussetzungen für Nachoptimierung Kühlen:** • Heat.EnableTuning = FALSE • Cool.EnableTuning = TRUE • Der Kühlausgang ist aktiviert (Config.ActivateCooling = TRUE). • Die PID-Parameterumschaltung ist aktiviert (Config.AdvancedCooling = TRUE) • Der Kühlausgang muss am Arbeitspunkt, an dem die Optimierung durchgeführt werden soll, aktiv sein (PidOutputSum < 0.0 (siehe Optimierungsoffset)). **Der Ablauf der Nachoptimierung ist abhängig von der Betriebsart, aus der sie gestartet wird:** • Automatikbetrieb (State = 3) mit PIDSelfTune.TIR.RunIn = FALSE (Vorbelegung) Falls Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb. PID_Temp regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. • Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3) mit PIDSelfTune.TIR.RunIn = TRUE Es wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen: – mit minimalem oder maximalem Ausgangswert Heizen bei Nachoptimierung Heizen

State / Mode	Beschreibung der Betriebsart
	– mit minimalem oder maximalem Ausgangswert Kühlen bei Nachoptimierung Kühlen. Das kann ein erhöhtes Überschwingen verursachen. Ist der Sollwert erreicht startet die Nachoptimierung. Falls der Sollwert nicht erreicht werden kann, bricht PID_Temp die Optimierung nicht automatisch ab. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt: • Setpoint > CurrentSetpoint + CancelTuningLevel oder • Setpoint < CurrentSetpoint - CancelTuningLevel Die Methode zur Berechnung der PID-Parameter kann mit PIDSelfTune.TIR.TuneRuleHeat und PIDSelfTune.TIR.TuneRuleCool für Heizen und Kühlen getrennt festgelegt werden. Bevor die PID-Parameter neu berechnet werden, werden die PID-Parameter in der Struktur CtrlParamsBackUp gesichert und können mit LoadBackUp reaktiviert werden. Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb. Nach erfolgloser Nachoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode. Die Phase der "Nachoptimierung" wird mit PIDSelfTune.TIR.State angezeigt.
3	Automatikbetrieb Im Automatikbetrieb regelt PID_Temp die Regelstrecke entsprechend der vorgegebenen Parameter aus. Falls eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt: • Erstoptimierung erfolgreich abgeschlossen • Nachoptimierung erfolgreich abgeschlossen • Ändern des Durchgangsparameters Mode auf den Wert 3 und eine steigende Flanke an ModeActivate. Der Wechsel vom Automatikbetrieb in den Handbetrieb erfolgt nur im Inbetriebnahmeeditor stoßfrei. Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.
4	Handbetrieb Im Handbetrieb geben Sie einen manuellen PID Ausgangswert am Parameter ManualValue vor. Die aus diesem Handwert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung. Diese Betriebsart können Sie zusätzlich über ManualEnable = TRUE aktivieren. Es wird empfohlen, die Betriebsarten nur über Mode und ModeActivate zu wechseln. Der Wechsel vom Handbetrieb in den Automatikbetrieb erfolgt stoßfrei. Im Handbetrieb wird die Variable ActivateRecoverMode berücksichtigt.
5	Ersatzausgangswert mit Fehlerüberwachung Der Regelalgorithmus ist ausgeschaltet. Die Variable SetSubstituteOutput bestimmt, welcher PID Ausgangswert (PidOutputSum) während dieser Betriebsart ausgegeben wird. • SetSubstituteOutput = FALSE: letzter gültiger PID Ausgangswert • SetSubstituteOutput = TRUE: Ersatzausgangswert (SubstituteOutput) Diese Betriebsart können Sie nicht mit Mode = 5 aktivieren. Sie wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert, wenn alle folgenden Bedingungen erfüllt sind: • Automatikbetrieb (State = 3) • ActivateRecoverMode = TRUE • Ein oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode wirkt. Sobald die Fehler nicht mehr anstehen, wechselt PID_Temp wieder in den Automatikbetrieb.

ENO-Verhalten

Falls State = 0 ist, dann ist ENO = FALSE.

Falls State ≠ 0 ist, dann ist ENO = TRUE.

Automatische Betriebsartwechsel während der Inbetriebnahme

Nach erfolgreicher Erst- oder Nachoptimierung wird der Automatikbetrieb aktiviert. Die folgende Tabelle zeigt, wie sich Mode und State während einer erfolgreichen Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet
n	4	1	Erstoptimierung erfolgreich beendet
n	3	3	Automatikbetrieb wird gestartet

Im Fehlerfall wechselt PID_Temp automatisch die Betriebsart.

Die folgende Tabelle zeigt, wie sich Mode und State während einer fehlerhaften Erstoptimierung ändern.

Zyklus-Nr.	Mode	State	Aktion
0	4	4	Mode = 1 setzen
1	1	4	ModeActivate = TRUE setzen
1	4	1	Wert von State wird an Mode gespeichert Erstoptimierung wird gestartet
n	4	1	Erstoptimierung abgebrochen
n	4	4	Handbetrieb wird gestartet

Falls ActivateRecoverMode = TRUE ist, wird die Betriebsart aktiviert, die an Mode gespeichert ist. Beim Start der Erst- oder Nachoptimierung hat PID_Temp den Wert von State am Durchgangsparameter Mode gespeichert. PID_Temp wechselt also in die Betriebsart, aus der die Optimierung gestartet wurde.

Falls ActivateRecoverMode = FALSE ist, wird in die Betriebsart "Inaktiv" gewechselt.

Siehe auch

Ausgangsparameter PID_Temp (Seite 3040)

Durchgangsparameter PID_Temp (Seite 3042)

Parameter ErrorBits PID_Temp

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der ErrorBits binär addiert angezeigt. Wird z. B. ErrorBits = 0000003h angezeigt, so stehen gleichzeitig die Fehler 0000001h und 0000002h an.

ErrorBits (DW#16#...)	Beschreibung
0000000	Es steht kein Fehler an.
0000001	Der Parameter "Input" ist außerhalb der Istwertgrenzen. • Input > Config.InputUpperLimit oder • Input < Config.InputLowerLimit Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Automatikbetrieb. Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.
0000002	Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Temp den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Temp wieder in den Automatikbetrieb. Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.
0000004	Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden. Wird PID_Temp als Heizkühlregler eingesetzt (Config.ActivateCooling = TRUE), so muss, um eine Istwertschwingung zu erzeugen und die Nachoptimierung erfolgreich durchzuführen, der PID Ausgangswert (PidOutputSum) am Sollwert • positiv sein für Nachoptimierung Heizen • negativ sein für Nachoptimierung Kühlen Ist diese Voraussetzung nicht erfüllt, nutzen Sie die Optimierungsoffsets (Variablen PIDSelfTune.TIR.OutputOffsetCool und PIDSelfTune.TIR.OutputOffsetHeat), siehe AUTOHOTSPOT. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0000008	Fehler beim Starten der Erstoptimierung. Der Istwert ist zu nahe am Sollwert oder größer als der Sollwert. Starten Sie die Nachoptimierung. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0000010	Der Sollwert wurde während der Optimierung verändert. An der Variable CancelTuningLevel können Sie die zulässige Schwankung des Sollwerts einstellen. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0000020	Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_Temp in der Betriebsart Nachoptimierung.

ErrorBits (DW#16#...)	Beschreibung
0000040	Fehler während der Erstoptimierung. Die Kühlung konnte den Istwert nicht verringern. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0000100	Fehler während der Nachoptimierung führte zu ungültigen Parametern. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0000200	Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Temp den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Temp wieder in den Automatikbetrieb. Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.
0000400	Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Temp den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Temp wieder in den Automatikbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.
0000800	Abtastzeitfehler: PID_Temp wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Automatikbetrieb. Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist. Falls dieser Fehler bei der Simulation mit PLCSIM aufgetreten ist, beachten Sie die Hinweise unter AUTOHOTSPOT.
0001000	Ungültiger Wert am Parameter "Setpoint" oder "SubstituteSetpoint": Wert hat kein gültiges Zahlenformat. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv und ActivateRecoverMode = TRUE waren, gibt PID_Temp den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Temp wieder in den Automatikbetrieb. Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.
0010000	Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_Temp im Handbetrieb und verwendet SubstituteOutput als PID Ausgangswert. Sobald Sie einen gültigen Wert an ManualValue vorgeben, verwendet PID_Temp diesen als PID Ausgangswert.
0020000	Ungültiger Wert an der Variablen SubstituteOutput. Wert hat kein gültiges Zahlenformat. PID_Temp bleibt in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" bzw. Handbetrieb und verwendet die Untergrenze des PID Ausgangswerts für Heizen (Config.Output.Heat.PidLowerLimit) als PID Ausgangswert. Sobald Sie einen gültigen Wert an SubstituteOutput vorgeben, verwendet PID_Temp diesen als PID Ausgangswert.

ErrorBits (DW#16#...)	Beschreibung
0040000	Ungültiger Wert am Parameter Disturbance. Wert hat kein gültiges Zahlenformat. Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, wird Disturbance auf Null gesetzt. PID_Temp bleibt im Automatikbetrieb. Falls vor Auftreten des Fehlers die Betriebsart Erst- oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist. Wenn Disturbance in der aktuellen Phase keinen Einfluss auf den Ausgangswert hat, wird die Optimierung nicht abgebrochen.
0200000	Fehler beim Master in der Kaskade: Slaves sind nicht im Automatikbetrieb oder haben Ersatzsollwert aktiviert und verhindern Optimierung des Masters. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0400000	Die Erstoptimierung Heizen ist nicht erlaubt, solange die Kühlung aktiv ist. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
0800000	Der Istwert muss nahe am Sollwert sein, um Erstoptimierung Kühlen zu starten. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
1000000	Fehler beim Starten der Optimierung: Heat.EnableTuning und Cool.EnableTuning sind nicht gesetzt oder passen nicht zur Konfiguration. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
2000000	Erstoptimierung Kühlen setzt erfolgreiche Erstoptimierung Heizen voraus. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
4000000	Fehler beim Starten der Nachoptimierung: Heat.EnableTuning und Cool.EnableTuning dürfen nicht gleichzeitig gesetzt sein. Falls vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.
8000000	Fehler während der Berechnung der PID Parameter führte zu ungültigen Parametern. Die ungültigen Parameter werden verworfen und die ursprünglichen PID Parameter bleiben unverändert erhalten. Es werden folge Fälle unterschieden: • Falls vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Automatikbetrieb. • Falls vor Auftreten des Fehlers der Handbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Temp im Handbetrieb. • Falls vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE war, wechselt PID_Temp in die Betriebsart, die an Mode gespeichert ist.

Variable ActivateRecoverMode PID_Temp

Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall. Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Falls der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

Automatikbetrieb und Handbetrieb

ACHTUNG
Ihre Anlage kann beschädigt werden. Falls ActivateRecoverMode = TRUE ist, bleibt PID_Temp im Fehlerfall auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb bzw. im Handbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_Temp in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.
TRUE	**Automatikbetrieb** Falls im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert, da PID_Temp bei jedem Fehlerfall zwischen berechnetem PID Ausgangswert und dem Ersatzausgangswert wechselt. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Falls einer oder mehrere der folgenden Fehler auftreten und vor Auftreten des Fehlers der Automatikbetrieb aktiv war, bleibt PID_Temp im Automatikbetrieb: • 0000001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen. • 0000800h: Abtastzeitfehler • 0040000h: Ungültiger Wert am Parameter Disturbance. • 8000000h: Fehler während der Berechnung der PID Parameter Falls einer oder mehrere der folgenden Fehler auftreten und vor Auftreten des Fehlers der Automatikbetrieb aktiv war, wechselt PID_Temp in die Betriebsart "Ersatzausgangswert mit Fehlerüberwachung": • 0000002h: Ungültiger Wert am Parameter Input_PER. • 0000200h: Ungültiger Wert am Parameter Input. • 0000400h: Berechnung des Ausgangswerts fehlgeschlagen. • 0001000h: Ungültiger Wert am Parameter Setpoint oder SubstituteSetpoint. Sobald die Fehler nicht mehr anstehen, wechselt PID_Temp wieder in den Automatikbetrieb. Falls in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" folgender Fehler auftritt, setzt PID_Temp den PID Ausgangswert solange wie dieser Fehler ansteht auf Config.Output.Heat.PidLowerLimit: • 0020000h: Ungültiger Wert an der Variable SubstituteOutput. Wert hat kein gültiges Zahlenformat. Dieses Verhalten ist unabhängig von SetSubstituteOutput.
	Handbetrieb Falls einer oder mehrere Fehler auftreten und vor Auftreten des Fehlers der Handbetrieb aktiv war, bleibt PID_Temp im Handbetrieb. Falls im Handbetrieb der folgende Fehler auftritt, setzt PID_Temp den PID Ausgangswert solange wie dieser Fehler ansteht auf SubstituteOutput: • 0010000h: Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat. Falls im Handbetrieb der Fehler 0010000h ansteht und folgender Fehler auftritt, setzt PID_Temp den PID Ausgangswert solange wie dieser Fehler ansteht auf Config.Output.Heat.PidLowerLimit: • 0020000h: Ungültiger Wert an der Variable SubstituteOutput. Wert hat kein gültiges Zahlenformat. Dieses Verhalten ist unabhängig von SetSubstituteOutput.

Erstoptimierung und Nachoptimierung

ActivateRecoverMode	Beschreibung
FALSE	Im Fehlerfall wechselt PID_Temp in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.
TRUE	Falls der folgende Fehler auftritt, bleibt PID_Temp in der aktiven Betriebsart: • 0000020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt. Die folgenden Fehler werden ignoriert: • 0010000h: Ungültiger Wert am Parameter ManualValue. • 0020000h: Ungültiger Wert an der Variable SubstituteOutput. Bei allen anderen Fehlern bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.

Variable Warning PID_Temp

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Variable Warning binär addiert angezeigt. Wird z. B. die Warnung 0000003h angezeigt, so stehen gleichzeitig die Warnungen 0000001h und 0000002h an.

Warning (DW#16#....)	Beschreibung
0000000	Es liegt keine Warnung vor.
0000001	Während der Erstoptimierung wurde der Wendepunkt nicht gefunden.
0000004	Der Sollwert wurde begrenzt auf die eingestellten Grenzen.
0000008	Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TIR.TuneRuleHeat = 3 bzw. TIR.TuneRuleCool = 3 berechnet.
0000010	Die Betriebsart konnte nicht geändert werden, da Reset = TRUE oder ManualEnable = TRUE.
0000020	Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.
0000040	Der Istwert hat eine seiner Warngrenzen überschritten.
0000080	Ungültiger Wert an Mode. Die Betriebsart wird nicht gewechselt.
0000100	Der Handwert wurde begrenzt auf die Grenzen des PID Ausgangswerts.
0000200	Die angegebene Regel zur Optimierung wird nicht unterstützt. Es werden keine PID-Parameter berechnet.
0001000	Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.
0004000	Die angegebene Auswahl des Ausgangswerts für Heizen und / oder Kühlen wird nicht unterstützt. Es wird nur der Ausgang OutputHeat bzw. OutputCool verwendet.
0008000	Ungültiger Wert an PIDSelfTune.SUT.AdaptDelayTime. Der Defaultwert 0 wird verwendet.
0010000	Ungültiger Wert an PIDSelfTune.SUT.CoolingMode. Der Defaultwert 0 wird verwendet.
0020000	Die Aktivierung der Kühlung (Variable Config.ActivateCooling) wird bei einem Regler, der als Master verwendet wird (Variable Config.Cascade.IsMaster), nicht unterstützt. PID_Temp arbeitet als Heizregler. Setzen Sie die Variable Config.ActivateCooling auf FALSE.
0040000	Ungültiger Wert an Retain.CtrlParams.Heat.Gain, Retain.CtrlParams.Cool.Gain oder Config.CoolFactor. PID_Temp unterstützt nur positive Werte für Proportionalverstärkung (Heizen und Kühlen) und Kühlfaktor. Der Automatikbetrieb bleibt aktiv mit PID Ausgangswert 0.0. Der Integralanteil wird angehalten.

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist oder Sie die Aktion mit gültigen Parametern wiederholen:

- 0000001h
- 0000004h
- 0000008h
- 0000040h
- 0000100h

Alle anderen Warnungen werden bei steigender Flanke an Reset oder ErrorAck gelöscht.

Variable PwmPeriode

Ist bei Verwendung von OutputHeat_PWM bzw. OutputCool_PWM die Abtastzeit PID-Algorithmus (Retain.CtrlParams.Heat.Cycle bzw. Retain.CtrlParams.Heat.Cycle) und damit die Periodendauer der Pulsweitenmodulation sehr groß, können Sie, um die Glattheit des Istwerts zu verbessern, an den Parametern Config.Output.Heat.PwmPeriode bzw. Config.Output.Cool.PwmPeriode eine abweichende kürzere Periodendauer vorgeben.

Periodendauer der Pulsweitenmodulation an OutputHeat_PWM

Periodendauer der PWM am Ausgang OutputHeat_PWM in Abhängigkeit von Config.Output.Heat.PwmPeriode:

- Heat.PwmPeriode = 0.0 (default)
  Die Abtastzeit des PID-Algorithmus für Heizen (Retain.CtrlParams.Heat.Cycle) wird als Periodendauer der PWM verwendet.
- Heat.PwmPeriode > 0.0
  Der Wert wird auf ein ganzzahliges Vielfaches der Abtastzeit PID_Temp (CycleTime.Value) gerundet und als Periodendauer der PWM verwendet.
  Der Wert muss folgende Bedingungen erfüllen:
  - Heat.PwmPeriode ≤ Retain.CtrlParams.Heat.Cycle
  - Heat.PwmPeriode > Config.Output.Heat.MinimumOnTime
  - Heat.PwmPeriode > Config.Output.Heat.MinimumOffTime

Periodendauer der Pulsweitenmodulation an OutputCool_PWM

Periodendauer der PWM am Ausgang OutputCool_PWM in Abhängigkeit von Config.Output.Cool.PwmPeriode und der Methode für Heizen/Kühlen:

- Cool.PwmPeriode = 0.0 und Kühlfaktor (Config.AdvancedCooling = FALSE):
  Die Abtastzeit des PID-Algorithmus für Heizen (Retain.CtrlParams.Heat.Cycle) wird als Periodendauer der PWM verwendet.
- Cool.PwmPeriode = 0.0 und PID-Parameterumschaltung (Config.AdvancedCooling = TRUE):
  Die Abtastzeit des PID-Algorithmus für Kühlen (Retain.CtrlParams.Cool.Cycle) wird als Periodendauer der PWM verwendet.
- Cool.PwmPeriode > 0.0:
  Der Wert wird auf ein ganzzahliges Vielfaches der Abtastzeit PID_Temp (CycleTime.Value) gerundet und als Periodendauer der PWM verwendet.
  Der Wert muss folgende Bedingungen erfüllen:
  - Cool.PwmPeriode ≤ Retain.CtrlParams.Cool.Cycle bzw. Retain.CtrlParams.Heat.Cycle
  - Cool.PwmPeriode > Config.Output.Cool.MinimumOnTime
  - Cool.PwmPeriode > Config.Output.Cool.MinimumOffTime

Config.Output.Cool.PwmPeriode ist nur wirksam, falls der Kühlausgang aktiviert ist (Config.ActivateCooling =TRUE).

Bei Verwendung von PwmPeriode wird die Genauigkeit des PWM-Ausgangssignals bestimmt durch das Verhältnis von PwmPeriode zur Abtastzeit PID_Temp (Zykluszeit des OB). PwmPeriode sollte mindestens das 10-fache der Abtastzeit PID_Temp betragen.

Falls die Abtastzeit des PID-Algorithmus nicht ein ganzzahliges Vielfaches von PwmPeriode ist, wird jede letzte Periode der PWM innerhalb der Abtastzeit des PID-Algorithmus entsprechend verlängert.

Beispiel für OutputHeat_PWM

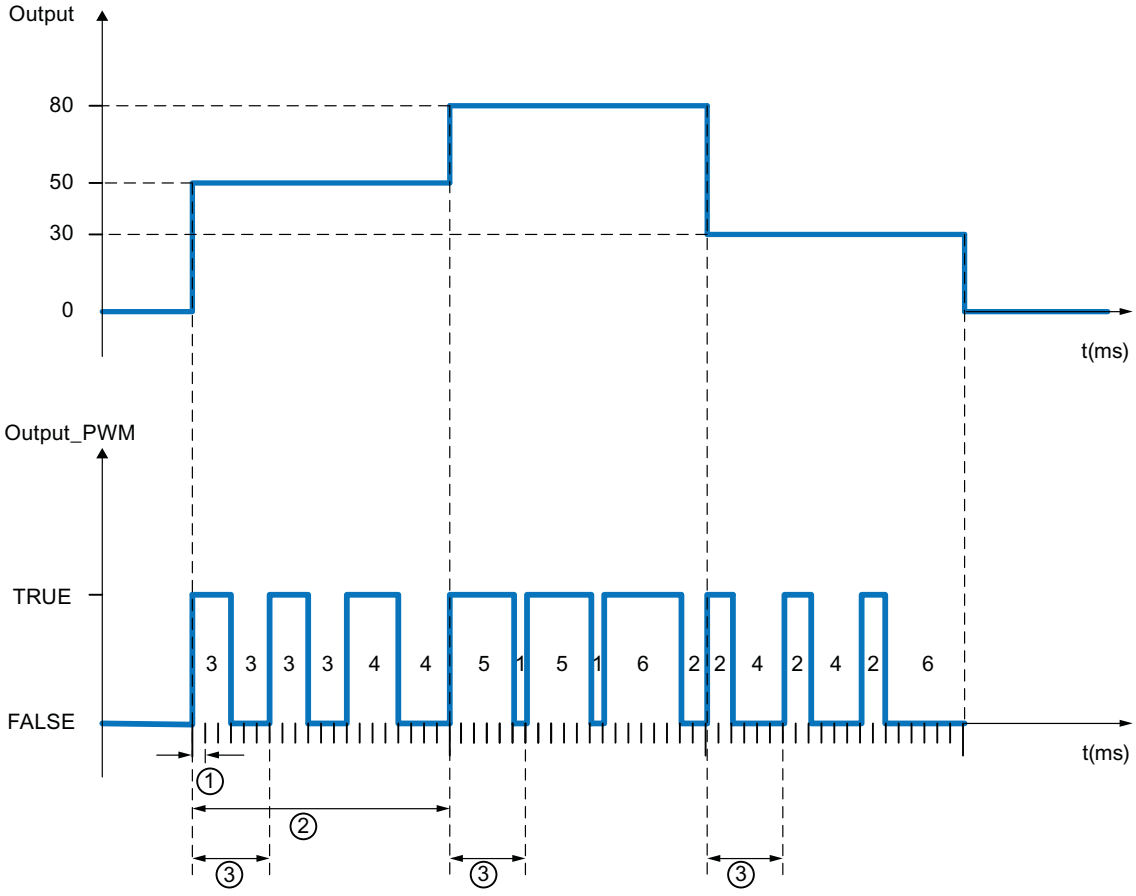

① Abtastzeit PID_Temp = 100.0 ms (Zykluszeit des aufrufenden Weckalarm-OB, Variable CycleTime.Value)

② Abtastzeit PID-Algorithmus = 2000.0 ms (Variable Retain.CtrlParams.Heat.Cycle)

③ Periodendauer der PWM für Heizen = 600.0 ms (Variable Config.Output.Heat.PwmPeriode)

Variable IntegralResetMode

Die Variable IntegralResetMode bestimmt, wie der I-Anteil PIDCtrl.IOutputOld vorbelegt wird:

- beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb"
- bei Flanke TRUE -> FALSE an Parameter Reset und Parameter Mode = 3

Diese Einstellung wirkt nur für einen Zyklus und ist nur wirksam, wenn der I-Anteil aktiviert ist (Variablen Retain.CtrlParams.Heat.Ti und Retain.CtrlParams.Cool.Ti > 0.0)

IntegralReset-Mode	Beschreibung
0	Glätten Der Wert von PIDCtrl.IOutputOld wird so vorbelegt, dass die Umschaltung stoßfrei erfolgt d. h. der "Automatikbetrieb" startet ausgehend von Ausgangswert = 0.0 (Parameter PidOutputSum) und es erfolgt kein Sprung des Ausgangswerts unabhängig von der Regeldifferenz (Sollwert – Istwert).
1	Löschen Es wird empfohlen die Gewichtung des P-Anteils (Variablen Retain.CtrlParams.Heat.PWeighting und Retain.CtrlParams.Cool.PWeighting) auf 1.0 zu setzen, falls diese Option verwendet wird. Der Wert von PIDCtrl.IOutputOld wird gelöscht. Wenn eine Regeldifferenz vorhanden ist, kommt es zu einem Sprung des PID Ausgangswerts. Die Richtung des Ausgangswertsprungs hängt von der aktiven Gewichtung des P-Anteils (Variablen Retain.CtrlParams.Heat.PWeighting und Retain.CtrlParams.Cool.PWeighting) und der Regeldifferenz ab: • aktive Gewichtung des P-Anteils = 1.0: Ausgangswertsprung und Regeldifferenz haben identische Vorzeichen. Beispiel: Liegt der Istwert unterhalb des Sollwerts (positive Regeldifferenz), springt der PID Ausgangswert auf einen positiven Wert. • aktive Gewichtung des P-Anteils < 1.0: Für große Regeldifferenzen haben der Sprung des PID Ausgangswerts und die Regeldifferenz identische Vorzeichen. Beispiel: Liegt der Istwert weit unterhalb des Sollwerts (positive Regeldifferenz), springt der PID Ausgangswert auf einen positiven Wert. Für kleine Regeldifferenzen haben der Sprung des PID Ausgangswerts und die Regeldifferenz unterschiedliche Vorzeichen. Beispiel: Liegt der Istwert knapp unterhalb des Sollwerts (positive Regeldifferenz), springt der PID Ausgangswert auf einen negativen Wert. Dies ist im Normalfall nicht erwünscht, da es zu einer vorübergehenden Vergrößerung der Regeldifferenz führt. Je kleiner dabei die Gewichtung des P-Anteils konfiguriert ist, desto größer muss die Regeldifferenz sein; um einen Sprung des PID Ausgangswertsprungs mit identischem Vorzeichen zu erhalten. Es wird empfohlen die Gewichtung des P-Anteils (Variablen Retain.CtrlParams.Heat.PWeighting und Retain.CtrlParams.Cool.PWeighting) auf 1.0 zu setzen, wenn diese Option verwendet wird. Andernfalls kann dies zum beschriebenen unerwünschten Verhalten bei kleinen Regeldifferenzen führen. Alternativ können Sie auch IntegralResetMode = 4 verwenden. Diese Option garantiert identische Vorzeichen für den Sprung des PID Ausgangs und für die Regeldifferenz unabhängig von der konfigurierten Gewichtung des P-Anteils und der Regeldifferenz.
2	Halten Der Wert von PIDCtrl.IOutputOld wird nicht verändert. Sie können über das Anwenderprogramm einen neuen Wert vorgeben.

IntegralReset-Mode	Beschreibung
3	Vorbelegen Der Wert von PIDCtrl.IOutputOld wird automatisch so vorbelegt, als ob im letzten Zyklus PidOutputSum = OverwriteInitialOutputValue gewesen wäre.
4	Wie Sollwertänderung (nur für PID_Temp mit Version ≥ 1.1) Der Wert von PIDCtrl.IOutputOld wird automatisch so vorbelegt, dass sich ein ähnlicher Sprung des PID Ausgangswerts ergibt, wie für einen PI Regler im Automatikbetrieb bei einer Sollwertänderung vom aktuellen Istwert zum aktuellen Sollwert. Falls eine Regeldifferenz vorhanden ist, kommt es zu einem Sprung des PID Ausgangswerts. Der Sprung des PID Ausgangswerts und Regeldifferenz haben identische Vorzeichen. Beispiel: Liegt der Istwert unterhalb des Sollwerts (positive Regeldifferenz), springt der PID Ausgangswert auf einen positiven Wert. Dies ist unabhängig von der konfigurierten Gewichtung des P-Anteils und der Regeldifferenz.

Wird IntegralResetMode mit einem Wert außerhalb des gültigen Wertebereichs belegt, verhält sich PID_Temp wie mit der Vorbelegung von IntegralResetMode:

- PID_Temp bis V1.0: IntegralResetMode = 1
- PID_Temp ab V1.1: IntegralResetMode = 4

PID Basisfunktionen

Unterschiede zu CONT_C S7-300/400

Die Anweisung CONT_C ist mit optimiertem Bausteinzugriff erstellt.

Alle Parameter sind remanent. Die Remanenz kann nicht geändert werden. Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie CONT_C vollständig laden.

AUTOHOTSPOT

Sonst verhält sich CONT_C auf einer CPU S7-1500 genauso wie auf einer CPU S7-300, S7-400.

Beschreibung CONT_C (Seite 5800)

Arbeitsweise CONT_C (Seite 5801)

Blockschaltbild CONT_C (Seite 5803)

Eingangsparameter CONT_C (Seite 5804)

Ausgangsparameter CONT_C (Seite 5805)

Unterschiede zu CONT_S S7-300/400

Die Anweisung CONT_S ist mit optimiertem Bausteinzugriff erstellt. S

Alle Parameter sind remanent. Die Remanenz kann nicht geändert werden. Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie CONT_S vollständig laden.

AUTOHOTSPOT

Sonst verhält sich CONT_S auf einer CPU S7-1500 genauso wie auf einer CPU S7-300, S7-400.

Beschreibung CONT_S (Seite 5806)

Arbeitsweise CONT_S (Seite 5807)

Blockschaltbild CONT_S (Seite 5808)

Eingangsparameter CONT_S (Seite 5809)

Ausgangsparameter CONT_S (Seite 5810)

Unterschiede zu PULSEGEN S7-300/400

Die Anweisung PULSEGEN ist mit optimiertem Bausteinzugriff erstellt.

Alle Parameter sind remanent. Die Remanenz kann nicht geändert werden. Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PULSEGEN vollständig laden.

AUTOHOTSPOT

Auf einer CPU S7-1500 verhält sich PULSEGEN genauso wie auf einer CPU S7-300, S7-400.

Beschreibung PULSEGEN (Seite 5810)

Arbeitsweise PULSEGEN (Seite 5812)

Betriebsarten PULSEGEN (Seite 5815)

Dreipunktregelung (Seite 5815)

Zweipunktregelung (Seite 5817)

Eingangsparameter PULSEGEN (Seite 5819)

Ausgangsparameter PULSEGEN (Seite 5820)

Unterschiede zu TCONT_CP S7-300/400

Die Anweisung TCONT_CP ist mit optimiertem Bausteinzugriff erstellt.

Alle Parameter sind remanent. Die Remanenz kann nicht geändert werden. Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie TCONT_CP vollständig laden.

AUTOHOTSPOT

Sonst verhält sich TCONT_CP auf einer CPU S7-1500 genauso wie auf einer CPU S7-300, S7-400.

Beschreibung TCONT_CP (Seite 5820)

Arbeitsweise TCONT_CP (Seite 5821)

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Eingangsparameter TCONT_CP (Seite 5834)

Ausgangsparameter TCONT_CP (Seite 5835)

Durchgangsparameter TCONT_CP (Seite 5836)

Statische Variablen TCONT_CP (Seite 5836)

Parameter STATUS_H (Seite 5841)

Parameter STATUS_D (Seite 5842)

Unterschiede zu TCONT_S S7-300/400

Die Anweisung TCONT_S ist mit optimiertem Bausteinzugriff erstellt.

Alle Parameter sind remanent. Die Remanenz kann nicht geändert werden. Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie TCONT_S vollständig laden.

AUTOHOTSPOT

Sonst verhält sich TCONT_S auf einer CPU S7-1500 genauso wie auf einer CPU S7-300, S7-400.

Beschreibung TCONT_S (Seite 5842)

Arbeitsweise TCONT_S (Seite 5844)

Blockschaltbild TCONT_S (Seite 5848)

Eingangsparameter TCONT_S (Seite 5849)

Ausgangsparameter TCONT_S (Seite 5850)

Durchgangsparameter TCONT_S (Seite 5850)

Statische Variablen TCONT_S (Seite 5850)

Polyline

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von Polyline einsetzen können:

CPU	FW	Polyline
S7-1200	ab V4.2	V1.0
S7-1500	ab V2.0	V1.0

Beschreibung Polyline

Beschreibung

Die Anweisung Polyline bildet den Eingangswert Input über eine Kennline auf den Ausgangswert Output ab. Die Kennlinie wird als Polygonzug mit maximal 50 Stützpunkten definiert. Zwischen diesen Stützpunkten wird linear interpoliert. Über die Anzahl und die Konfiguration der Stützpunkte können Sie den Polygonzug an die gewünschte Kennlinie anpassen.

Die Anweisung Polyline kann z.B. verwendet werden um ein nichtlineares Verhalten von Sensoren oder Aktoren zu linearisieren.

Interpolationsberechnung

Polyline berechnet den Ausgangswert am Parameter Output für den Eingangswert am Parameter Input, der zwischen den Stützpunktwerten x_i und x_{i+1} liegt mit einer linearen Interpolation. Die lineare Interpolation wird nach folgender Formel berechnet:

$$Output = \frac{(Input - x_i)}{(x_{i+1} - x_i)} (y_{i+1} - y_i) + y_i$$

Mit dem Parameter Reset = TRUE kann auch ein alternativer Ausgangswert über den Parameter SubstituteOutput vorgegeben werden.

Polygonzug-Daten

Die Wertepaare für den Polygonzug sind in dem Bereich Static der Anweisung enthalten.

Hinweis

- Die minimale Anzahl der zu konfigurierenden Wertepaare beträgt 2.
- Die maximale Anzahl der zu konfigurierenden Wertepaare beträgt 50.
- Für eine gültige Konfiguration müssen die x-Werte in aufsteigender Reihenfolge vorgegeben werden.

Um die Polygonzug-Daten ändern zu können, ohne dass die Änderungen sofort wirksam werden, sind die Wertepaare des Polygonzugs dupliziert und in folgenden Strukturen enthalten:

- UserData
  Die Polygonzug-Daten in dieser Struktur sind editierbar.
  Nutzen Sie diese Struktur um die Polygonzug-Daten festzulegen oder zu ändern. Änderungen in dieser Struktur wirken sich auf die Interpolationsberechnung erst aus, wenn die Prüfung und das Kopieren der Daten in die Struktur WorkingData angestoßen wird. Dies erfolgt durch Setzen von Validate = TRUE oder automatisch während der ersten Bearbeitung von Polyline nach dem Betriebszustandwechsel der CPU von STOP auf RUN. Die Vorbelegung der Werte in dieser Struktur stellt keine gültige Konfiguration dar. Um die Werte für die Interpolationsberechnung zu verwenden, ändern Sie die Variablen auf gültige Werte.
- WorkingData
  Die Polygonzug-Daten in dieser Struktur sind nicht editierbar. Diese Daten werden für die Interpolationsberechnung verwendet. Ändern Sie die Daten in dieser Struktur nicht manuell.

Beide Strukturen haben den gleichen Datentyp und somit den gleichen Inhalt:

- NumberOfUsedPoints
  Anzahl der verwendeten Stützpunkte für die Interpolationsberechnung.
- Point
  Das Array mit 50 Elementen enthält Wertepaare der Stützpunkte Point[i].x und Point[i].y mit dem Index "i" von 1 bis 50.

Das folgende Bild zeigt ein Polygonzug mit vier Stützpunkten:

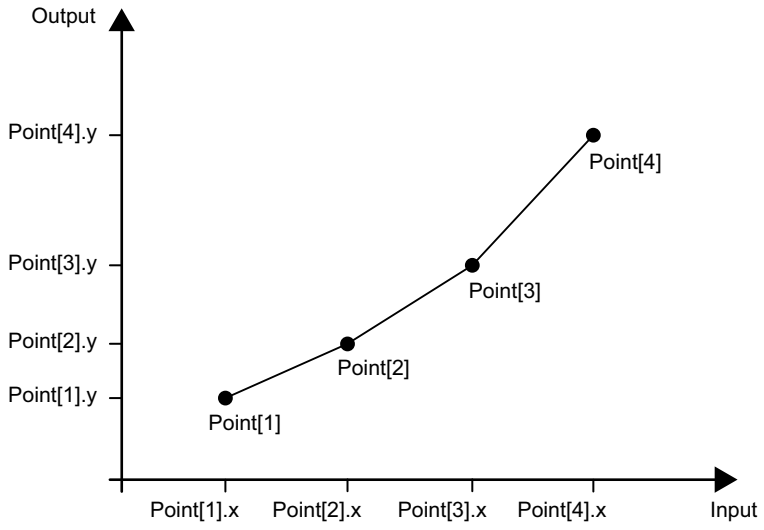

Aufruf

In einem OB wird Polyline als Einzelinstanz-DB aufgerufen. In einem FB kann Polyline sowohl als Einzelinstanz-DB, als auch als ein Multiinstanz-DB und als ein Parameterinstanz-DB aufgerufen werden.

Bei dem Aufruf der Anweisung wird kein Technologieobjekt angelegt. Parametrier- und Inbetriebnahmeoberfläche steht Ihnen nicht zur Verfügung. Sie parametrieren Polyline direkt in dem Instanz-DB und nehmen Polyline über eine Beobachtungstabelle des Anwenderprogramms in der CPU oder HMI in Betrieb.

Anlauf

Die Variablen in den Strukturen UserData und WorkingData sind nicht remanent. Diese Variablen werden nach jedem Betriebszustandwechsel der CPU von STOP auf RUN mit den Startwerten initialisiert.

Wenn Sie die Aktualwerte in der Struktur UserData im Online-Betrieb ändern und diese Werte nach dem Betriebszustandwechsel der CPU von STOP auf RUN erhalten bleiben sollen, dann sichern Sie diese Werte in den Startwerten des Datenbausteins.

Bei dem ersten Aufruf der Anweisung Polyline nach dem Betriebszustandwechsel der CPU von STOP auf RUN werden die Daten in der Struktur UserData automatisch auf Gültigkeit überprüft. Bei erfolgreicher Überprüfung werden die Daten an die Struktur WorkingData übergeben.

Verhalten im Fehlerfall

Die Anweisung Polyline erkennt unterschiedliche Fehler, die bei Interpolationsberechnung auftreten können. Das Ergebnis der Interpolationsberechnung kann trotz eines anstehenden Fehlers am Ausgang ausgegeben werden. Wenn keine korrekte Berechnung des Interpolationsergebnisses wegen eines Fehlers möglich ist, dann wird ein Ersatzausgangswert am Ausgang ausgegeben.

An der Variable ErrorMode legen Sie im Fall eines Fehlers, bei dem keine korrekte Berechnung des Interpolationsergebnisses möglich ist, den Ersatzausgangswert folgendermaßen fest:

- Wenn die Variable ErrorMode auf den Wert 0 gesetzt ist, dann gibt Polyline den Eingangswert am Parameter Input aus.
- Wenn die Variable ErrorMode auf den Wert 1 gesetzt ist, dann gibt Polyline den Ersatzausgangswert aus. Den Ersatzausgangswert geben Sie am Parameter SubstituteOutput vor.
- Wenn die Variable ErrorMode auf den Wert 2 gesetzt ist, dann gibt Polyline den letzten gültigen Ausgangswert am Parameter Output aus.

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error auf FALSE gesetzt. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits ist remanent und wird nur durch eine steigende Flanke am Parameter Reset oder ErrorAck zurückgesetzt.

Arbeitsweise Polyline

Polygonzug-Daten

Um die Polygonzug-Daten zu ändern, editieren Sie die Werte in der Struktur UserData. Anschließend werden die Werte auf Gültigkeit überprüft und an die Struktur WorkingData übergeben. Erst in der Stuktur WorkingData werden die Werte für die Interpolationsberechnung verwendet.

Die Werte werden überprüft und übertragen, wenn

- Sie den Parameter Validate auf TRUE setzen, während der Parameter Reset auf FALSE gesetzt ist.
- Polyline nach dem Betriebszustandwechsel der CPU von STOP nach RUN zum ersten mal aufgerufen wird, während der Parameter Reset auf FALSE gesetzt ist.
  Wenn Polyline bereits z. B. in dem OB100 aufgerufen wurde, dann wird keine erneute automatische Überprüfung der Werte bei nachfolgenden Aufrufen durchgeführt.

Wenn die Polygonzug-Daten in der Struktur UserData ungültig sind, dann bleiben die vorherigen Polygonzug-Daten in der Struktur WorkingData unverändert und eine entsprechende Fehlermeldung wird ausgegeben. Wenn die Überprüfung zum ersten mal durchgeführt wurde, dann sind keine gültigen Werte in der Struktur WorkingData verfügbar und eine entsprechende Fehlermeldung wird ausgegeben. Der Parameter Output wird in diesem Fall mit dem Ersatzausgangswert festgelegt, den Sie mit der Variable ErrorMode konfigurieren.

Die Überprüfung und Übertragung der Werte von der Struktur UserData erfordert mehr CPU-Bearbeitungszeit als die Interpolationsberechnung. In zeitkritischen Anwendungen kann die erste Ausführung von Polyline im Anlauf-OB 100 erfolgen. Somit wird die einmalige zeitaufwändige Überprüfung und Übertragung der Polygonzug-Daten vor den zyklischen Anwendungsprogrammteilen bereits abgeschlossen.

Gültigkeit der Polygonzug-Daten

Bei der Prüfung der Werte in der Struktur UserData müssen die Werte folgende Bedingungen erfüllen, damit ein gültiger Polygonzug für die Interpolationsberechnung verfügbar ist:

- 2 ≤ UserData.NumberOfUsedPoints ≤ 50
- UserData.Point[j].x < UserData.Point[j+1].x mit Index j = 1.. (UserData.NumberOfUsedPoints – 1)
- -3.402823e+38 ≤ UserData.Point[i].x ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints
- -3.402823e+38 ≤ UserData.Point[i].y ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints
- UserData.Point[i].x and UserData.Point[i].y sind gültige REAL-Werte (≠ NaN) mit Index i = 1..UserData.NumberOfUsedPoints

Wenn eine oder mehrere Bedingungen während der Überprüfung nicht erfüllt sind, dann werden die Werte in der Struktur UserData nicht in die Struktur WorkingData übernommen. Eine entsprechende Fehlermeldung wird am Parameter ErrorBits (Seite 3106) ausgegeben.

Die Vorbelegung der Werte in der Struktur UserData stellt keine gültige Konfiguration dar. Ändern Sie die Variablen auf gültige Werte, damit die Variablen für die Interpolationsberechnung verwendet werden können.

Hinweis

Wenn für Ihre Anwendung mehr als die maximale Anzahl von 50 Stützpunkten erforderlich sind, dann verwenden Sie zwei oder mehr Instanzen von Polyline.

Ausgangswert berechnen

Wenn der Eingangswert am Parameter Input unterhalb des ersten x-Werts oder oberhalb des letzten x-Werts der verwendeten Stützpunkte liegt, dann konfigurieren Sie die Belegung des Parameters Output mit folgenden Einstellungen an der Variable OutOfRangeMode:

- OutOfRangeMode = 0
  Der Ausgangswert wird mit der Steigung der ersten oder letzten zwei Stützpunkte extrapoliert.

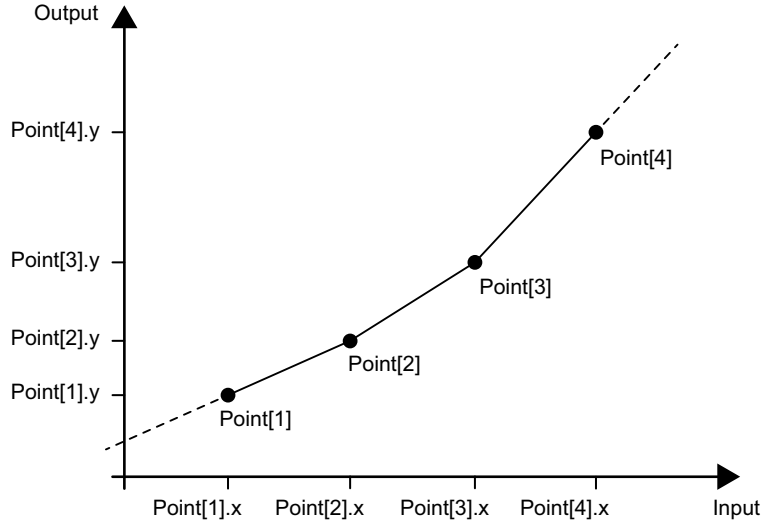

  Wenn die Variable OutOfRangeMode außerhalb des zulässigen Wertebereichs von 0 bis 1 liegt, dann wird die standardmäßige Vorbelegung 0 wirksam.

- OutOfRangeMode = 1
  Der Ausgangswert wird auf den y-Wert des ersten oder letzten Stützpunkts begrenzt.

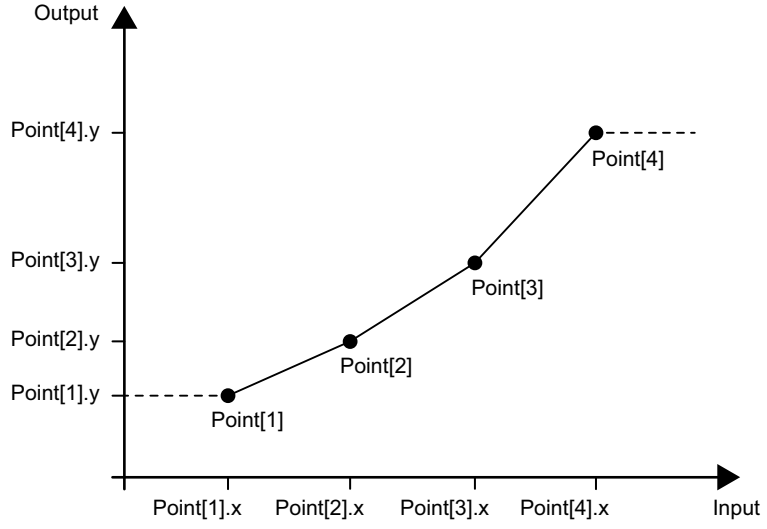

Der Parameter Output hat einen zulässigen Wertebereich eines REAL-Datentyps von -3.402823e+38 bis 3.402823e+38. Der Ausgangswert am Parameter Output wird bei jeder Ausführung der Anweisung Polyline auf Gültigkeit überprüft. Wenn die Interpolationsberechnung einen ungültigen REAL-Wert ergibt, dann wird der Ausgangswert mit der Einstellung an der Variable ErrorMode ersetzt.

Freigabeverhalten EN/ENO

Wenn eine der folgenden Bedingungen erfüllt ist, dann wird der Freigabeausgang ENO auf FALSE gesetzt:

- Freigabeeingang EN ist auf TRUE gesetzt und der Parameter Output ist durch einen Ersatzausgangswert festgelegt bei Fehlermeldungen ErrorBits ≥ 16#0001_0000.
- Freigabeeingang EN ist auf FALSE gesetzt.

Wenn der Freigabeeingang EN auf TRUE gesetzt ist und der Parameter Output folgendermaßen ermittelt werden kann, dann wird der Freigabeausgang ENO auf TRUE gesetzt:

- Mit der gültigen Interpolationsberechnung bei Reset = FALSE.

oder

- Mit dem gültigen Parameter SubstituteOutput bei Reset = TRUE.

Aktuell verwendete Stützpunkte

Die Variable NextXIndex gibt den Index des nächsthöheren x-Werts für den aktuellen Eingangswert aus. Damit können Sie die für die aktuelle Interpolationsberechnung verwendeten Stützpunkte ermitteln.

WorkingData.Point[NextXIndex-1].x < Input ≤ WorkingData.Point[NextXIndex].x

Beispiel:

- Wenn der Wert des Parameters Input zwischen WorkingData.Point[3].x und WorkingData.Point[4].x liegt, dann hat die Variable NextXIndex den Wert 4.
- Wenn der Wert des Parameters Input kleiner als WorkingData.Point[1].x ist, dann hat die Variable NextXIndex den Wert 1.
- Wenn der Wert des Parameters Input größer als WorkingData.Point[WorkingData.NumberOfUsedPoints].x und somit größer als der letzte x-Wert des Polygonzugs ist, dann hat die Variable NextXIndex den Wert der Variable WorkingData.NumberOfUsedPoints + 1. Somit beträgt der zulässige maximal-Wert der Variable NextXIndex 51.

Eingangsparameter Polyline

Parameter	Datentyp	Vorbelegung	Beschreibung
Input	REAL	0.0	Eingangswert
Substitu-teOutput	REAL	0.0	SubstituteOutput wird als Ersatzausgangswert verwendet, wenn • Reset = TRUE oder • keine korrekte Berechnung des Interpolationsergebnisses wegen eines Fehlers mit Fehlermeldung ErrorBits ≥ 16#0001_0000 möglich ist und ErrorMode auf den Wert 1 konfiguriert ist.
Validate	BOOL	FALSE	Wenn Validate auf TRUE gesetzt ist, dann werden die Polygonzug-Daten in UserData auf Gültigkeit geprüft und nach WorkingData übernommen.

Parameter	Datentyp	Vorbelegung	Beschreibung
ErrorAck	BOOL	FALSE	Löscht die Fehlermeldungen • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt
Reset	BOOL	FALSE	Führt einen Neustart der Anweisung durch • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt. • Solange Reset auf TRUE gesetzt ist, wird der Ersatzausgangswert SubstituteOutput am Ausgang ausgegeben. • Solange Reset auf FALSE gesetzt ist, wird die Interpolationsberechnung ausgeführt.

Ausgangsparameter Polyline

Parameter	Datentyp	Vorbelegung	Beschreibung
Output	REAL	0.0	Ausgangswert
Error	BOOL	FALSE	Wenn Error auf TRUE gesetzt ist, steht aktuell mindestens ein Fehler an.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 3106) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.

Statische Variablen Polyline

Variable	Datentyp	Vorbelegung	Beschreibung
UserData	AuxFct_PointTable	-	Eingabebereich für Polygonzug-Daten Die Polygonzug-Daten in der Struktur UserData sind editierbar. Änderungen in dieser Struktur wirken sich auf die Interpolationsberechnung erst aus, wenn die Überprüfung und das Kopieren der Daten in die Struktur WorkingData angestoßen wird.
UserData.NumberOfUsedPoints	INT	0	Anzahl der verwendeten Stützpunkte für die Interpolationsberechnung Zulässiger Wertebereich: 2 bis 50
UserData.Point	Array[1..50] of AuxFct_Point	-	Stützpunkte für die Interpolationsberechnung Das Array mit 50 Elementen vom Datentyp AuxFct_Point enthält die Wertepaare der Stützpunkte.
UserData.Point[i]	AuxFct_Point	-	Stützpunkt für die Interpolationsberechnung Ein Element mit dem Index "i" aus dem Array "Point".
UserData.Point[i].x	REAL	0.0	x-Wert des Stützpunkts Zulässiger Wertebereich: Point[i].x < Point[i+1].x
UserData.Point[i].y	REAL	0.0	y-Wert des Stützpunkts

Variable	Datentyp	Vorbelegung	Beschreibung
WorkingData	AuxFct_PointTable	-	Anzeigebereich der aktuell wirksamen Polygonzug-Daten Die Polygonzug-Daten in der der Struktur WorkingData sind nicht editierbar. Sie werden für die Interpolationsberechnung verwendet.
WorkingData.NumberOfUsedPoints	INT	0	Anzahl der verwendeten Stützpunkte für die Interpolationsberechnung Zulässiger Wertebereich: 2 bis 50
WorkingData.Point	Array[1..50] of AuxFct_Point	-	Stützpunkte für die Interpolationsberechnung Das Array mit 50 Elementen vom Typ AuxFct_Point enthält die Wertepaare der Stützpunkte.
WorkingData.Point[i]	AuxFct_Point	-	Stützpunkt für die Interpolationsberechnung Ein Element mit dem Index "i" aus dem Array "Point".
WorkingData.Point[i].x	REAL	0.0	x-Wert des Stützpunkts Zulässiger Wertebereich: Point[i].x < Point[i+1].x
WorkingData.Point[i].y	REAL	0.0	y-Wert des Stützpunkts
ErrorMode	INT	0	Auswahl des Ersatzausgangswerts im Fehlerfall • 0 = Input • 1 = SubstituteOutput • 2 = letzter gültiger Ausgangswert Zulässiger Wertebereich: 0 bis 2
OutOfRangeMode	INT	0	Auswahl des Ausgangswerts, wenn der Eingangswert außerhalb der definierten x-Werte liegt • 0 = Steigung halten • 1 = y-Wert des ersten / letzten Stützpunkts Zulässiger Wertebereich: 0 bis 1
NextXIndex	INT	2	Index des nächsten x-Werts Dient zur Beobachtung des Indexes der Stützpunkte, die bei der aktuellen interpolationsberechnung verwendet werden. Folgende Bedingung gilt: WorkingData.Point[NextXIndex-1].x < Input ≤ WorkingData.Point[NextXIndex].x Ändern Sie diesen Wert nicht manuell.

Parameter ErrorBits

Wenn gleichzeitig mehrere Fehler anstehen, dann werden die Werte der ErrorBits binär addiert angezeigt. Wenn z. B. ErrorBits = 16#0000_0003 angezeigt wird, dann stehen gleichzeitig die Fehler 16#0000_0001 und 16#0000_0002 an.

Bei Polyline werden die Fehler, die am Parameter ErrorBits ausgegeben werden, in zwei Kategorien unterschieden:

- Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000
- Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000 anstehen, dann wird der Freigabeausgang ENO nicht geändert. Der Ausgangswert kann trotz dieses Fehlers folgendermaßen ermittelt werden:

- Interpolationsberechnung wenn Reset = FALSE.

oder

- Ausgabe von SubstituteOutput wenn Reset = TRUE.

ErrorBits (DW#16#...)	Beschreibung
0000_0000	Kein Fehler steht an.
0000_0001	Der Parameter Output wurde begrenzt auf -3.402823e+38 oder +3.402823e+38. Wenn am Ausgang der Interpolationswert ausgegeben wird (Reset = FALSE und ErrorBits < 16#0001_0000), dann überprüfen Sie folgende, in der Interpolationsberechnung verwendete Variablen: • Input • WorkingData.Point[i].x • WorkingData.Point[i].y Wenn ein Ersatzausgangswert am Parameter Output ausgegeben wird bei Reset = FALSE und ErrorBits ≥ 16#0001_0000, dann überprüfen Sie die Parameter Input oder SubstituteOutput, abhängig von dem eingestellten Wert an der Variable ErrorMode. Wenn Reset = TRUE, dann überprüfen Sie den Parameter SubstituteOutput. Wenn der Parameter Output nicht mehr begrenzt ist, dann wird der Fehler gelöscht, der am Parameter Error gespeichert ist.
0000_0002	Eine oder mehrere Variablen in der Struktur UserData haben ungültige Werte, jedoch sind gültige Polygonzug-Daten in der Struktur WorkingData verfügbar. Die Polygonzug-Daten in der Struktur UserData wurden überprüft, weil der Parameter Validate auf TRUE gesetzt ist und gleichzeitig der Parameter Reset auf FALSE gesetzt ist. Die Polygonzug-Daten in der Struktur UserData werden nicht in die Struktur WorkingData übernommen, sodass die vorgenommenen Änderungen in der Struktur UserData nicht wirksam werden. Der FB Polyline setzt die Interpolationsberechnung mit den unveränderten Werten in der Struktur WorkingData fort. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind, wenn der Parameter Validate auf TRUE gesetzt ist: 1. 2 ≤ UserData.NumberOfUsedPoints ≤ 50 2. UserData.Point[j].x < UserData.Point[j+1].x mit Index j = 1..(UserData.NumberOfUsedPoints - 1) 3. -3.402823e+38 ≤ UserData.Point[i].x ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints 4. -3.402823e+38 ≤ UserData.Point[i].y ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints 5. UserData.Point[i].x und UserData.Point[i].y sind gültige REAL-Werte (≠ NaN) mit Index i = 1..UserData.NumberOfUsedPoints Das Fehlerbit am Parameter Error wird gelöscht, wenn • der Parameter Validate auf FALSE gesetzt wird. oder • der Parameter Reset auf TRUE gesetzt wird. oder • der Parameter Validate auf TRUE gesetzt ist, während der Parameter Reset auf FALSE gesetzt ist und die fünf oben benannten Bedingungen für UserData erfüllt sind.

Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000 anstehen, dann kann der Ausgangswert nicht wie erwartet ermittelt werden. Stattdessen wird der Ersatzausgangswert ausgegeben. Der Freigabeausgang ENO wird auf FALSE gesetzt.

ErrorBits (DW#16#...)	Beschreibung
0001_0000	Der Parameter SubstituteOutput oder Input, der als Ausgangswert verwendet wird, hat keinen gültigen REAL-Wert. Der Ausgang wird auf 0.0 gesetzt. Mögliche Fehlerszenarios: • Der Parameter Reset ist auf FALSE gesetzt und ein Fehler mit ErrorBits ≥ 16#0002_0000 steht an, der die Interpolationsberechnung verhindert und – der Parameter Input hat keinen gültigen REAL-Wert, die Variable ErrorMode = 0. – der Parameter SubstituteOutput hat keinen gültigen REAL-Wert, die Variable ErrorMode = 1. • Der Parameter Reset ist auf TRUE gesetzt, während der Parameter SubstituteOutput keinen gültigen REAL-Wert hat. Das Fehlerbit ma Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter SubstituteOutput oder Input auf einen gültigen REAL-Wert geändert wird. oder • der Parameter SubstituteOutput oder Input nicht mehr als Ausgangswert verwendet wird. Wenn der Parameter SubstituteOutput oder Input auf einen gültigen REAL-Wert geändert wird, dann verwendet FB Polyline den Parameter SubstituteOutput oder Input wieder als Ausgangswert.
0002_0000	Der Parameter Input hat keinen gültigen REAL-Wert, während die Interpolationsberechnung ausgeführt werden soll (Reset = FALSE). Der Parameter Output wird mit dem Ersatzausgangswert belegt, den Sie an der Variable ErrorMode konfigurieren. Beachten Sie, dass der Ausgangswert auf 0.0 gesetzt wird, wenn die Variable ErrorMode = 0 gesetzt ist, da der Wert am Parameter Input in diesem Fall nicht als Ersatzausgangswert verwendet wird. Die Variable NextXIndex wird nicht aktualisiert, solange der Parameter Input einen ungültigen REAL-Wert hat. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter Input auf einen gültigen REAL-Wert geändert wird. oder • der Parameter Reset auf TRUE gesetzt wird. Wenn der Parameter Input auf einen gültigen REAL-Wert geändert wird und der Parameter Reset auf FALSE gesetzt ist, dann gibt der FB Polyline am Ausgang wieder den Interpolationswert aus.

ErrorBits (DW#16#...)	Beschreibung
0004_0000	Die Interpolationsberechnung ergibt einen ungültigen REAL-Wert für Parameter Output. Überprüfen Sie die REAL-Werte in der Struktur WorkingData auf Gültigkeit. Der Parameter Output wird mit dem Ersatzausgangswert belegt, den Sie an der Variable ErrorMode konfigurieren. Wenn Sie die Polygonzug-Daten ändern möchten, dann editieren Sie zuerst die Struktur UserData und setzen Sie anschließend den Parameter Validate = TRUE. Ändern Sie die Daten in der Struktur Working-Data nicht manuell. Die Variable NextXIndex kann einen ungültigen Wert haben, solange dieser Fehler ansteht. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • die Interpolationsberechnung wieder gültige Werte für Parameter Output ermittelt. oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die Interpolationsberechnung wieder gültige REAL-Werte ermittelt und der Parameter Reset = FALSE gesetzt ist, dann gibt der FB Polyline am Ausgang wieder den Interpolationswert aus.

ErrorBits (DW#16#...)	Beschreibung
0008_0000	Eine oder mehrere Variablen in der Struktur UserData haben ungültige Werte, während keine gültigen Polygonzug-Daten in der Struktur WorkingData verfügbar sind. Die Polygonzug-Daten in der Struktur UserData wurden geprüft, da • der Parameter Validate auf TRUE gesetzt ist, während der Parameter Reset auf FALSE gesetzt ist. oder • während der ersten Ausführung von Polyline nach dem Betriebszustandwechsel der CPU von RUN auf STOP der Parameter Reset auf FALSE gesetzt ist. Die Polygonzug-Daten in der Struktur UserData werden nicht in die Struktur WorkingData übernommen, sodass die Werte in der Struktur UserData nicht wirksam werden. Der FB Polyline gibt am Parameter Output nicht den Interpolationswert aus, da keine gültigen Polygonzug-Daten in der Struktur WorkingData enthalten sind. Der Parameter Output wird mit dem Ersatzausgangswert belegt, der an der Variable ErrorMode konfiguriert ist. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind, wenn der Parameter Validate auf TRUE gesetzt ist oder während der ersten Ausführung von Polyline nach dem Betriebszustandwechsel der CPU von STOP auf RUN der Parameter Reset auf FALSE gesetzt ist: 1. 2 ≤ UserData.NumberOfUsedPoints ≤ 50 2. UserData.Point[j].x < UserData.Point[j+1].x mit Index j = 1..(UserData.NumberOfUsedPoints – 1) 3. -3.402823e+38 ≤ UserData.Point[i].x ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints 4. -3.402823e+38 ≤ UserData.Point[i].y ≤ 3.402823e+38 mit Index i = 1..UserData.NumberOfUsedPoints 5. UserData.Point[i].x und UserData.Point[i].y sind gültige REAL-Werte (≠ NaN) mit Index i = 1..UserData.NumberOfUsedPoints Beachten Sie, dass alle Variablen in den Strukturen UserData und WorkingData nicht remanent sind. Diese Variablen werden nach jedem Betriebszustandwechsel der CPU von STOP auf RUN mit den Startwerten initialisiert. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter Validate auf TRUE gesetzt wird, während der Parameter Reset auf FALSE gesetzt ist und die fünf oben benannten Bedingungen für UserData erfüllt sind. oder • während der ersten Ausführung von Polyline nach dem Betriebszustandwechsel der CPU von STOP auf RUN der Parameter Reset auf FALSE gesetzt ist und die fünf oben benannten Bedingungen für UserData erfüllt sind. oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die fünf oben benannten Bedingungen für UserData erfüllt sind und folgende Bedingungen erfüllt sind, dann gibt der FB Polyline am Ausgang wieder den Interpolationswert aus: • Der Parameter Validate auf TRUE gesetzt wird, während der Parameter Reset auf FALSE gesetzt ist. oder • Während der ersten Ausführung von Polyline nach dem Betriebszustandwechsel der CPU von RUN auf STOP der Parameter Reset auf FALSE gesetzt ist.

SplitRange

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von SplitRange einsetzen können:

CPU	FW	SplitRange
S7-1200	ab V4.2	V1.0
S7-1500	ab V2.0	V1.0

Beschreibung SplitRange

Beschreibung

Die Anweisung SplitRange konvertiert den Eingangswert in einen Ausgangswert. Der Eingangswert liegt dabei in dem Wertebereich, der durch Points.x1 und Points.x2 begrenzt ist. Der Ausgangswert liegt in dem Wertebereich, der durch Points.y1 und Points.y2 begrenzt ist.

Das folgende Bild zeigt die Kennlinie einer beispielhaften Konfiguration der Anweisung SplitRange:

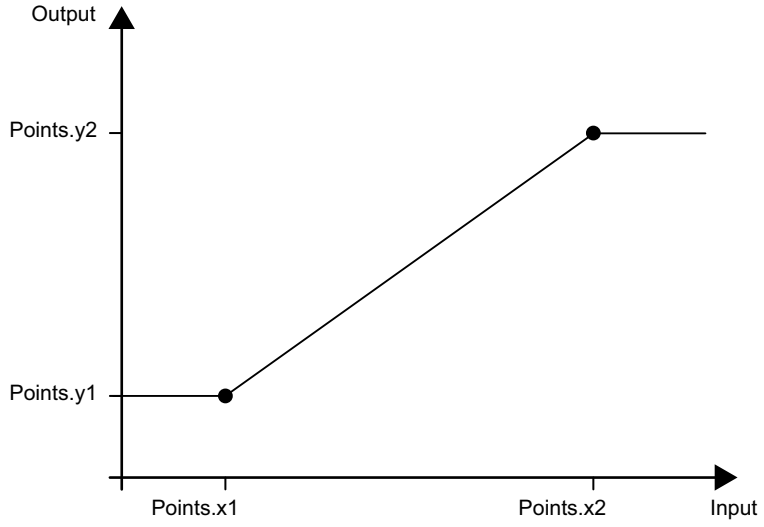

Setzen Sie SplitRange ein, wenn Sie eine Regelung eines Prozesses benötigen, den mehrere Aktoren beeinflussen. SplitRange teilt dabei den Ausgangswertebereich des PID-Reglers in mehrere Unterbereiche auf. Weisen Sie jedem Aktor einen Unterbereich zu. Das Anwenderprogramm ruft den Baustein ein mal pro Unterbereich auf. Der Eingangswert jeder SplitRange-Instanz ist mit dem Ausgangswert des PID-Reglers verbunden.

Das folgende Bild zeigt beispielhaft ein Regelkreis mit zwei SplitRange-Instanzen und zwei Aktoren:

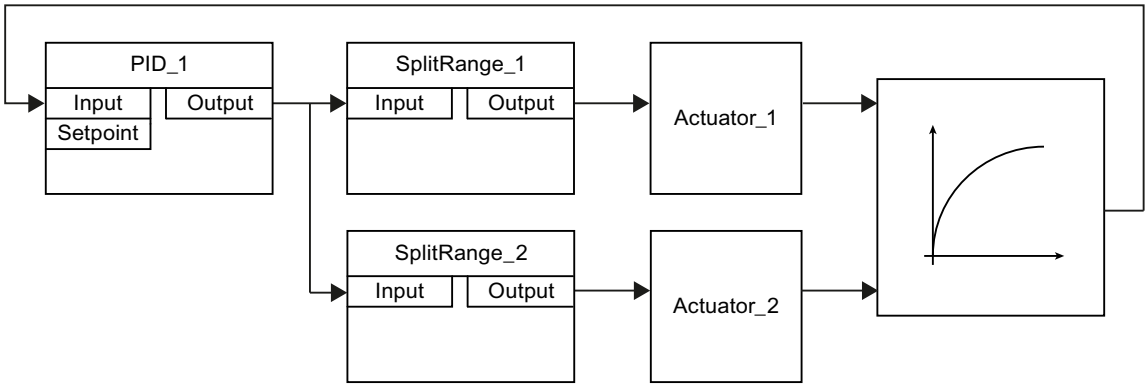

Gültigkeit der SplitRange-Daten

Die Wertepaare in der Struktur Points definieren den Eingangs- und Ausgangswertebereich von SplitRange. Die beiden Wertepaare liegen im statischen Bereich des Bausteins SplitRange.

SplitRange prüft bei jedem Aufruf, ob folgende Bedingungen erfüllt sind, damit gültige Werte für die Berechnung des Ausgangswerts verfügbar sind:

- Points.x1 < Points.x2
- Points.x1, Points.y1, Points.x2 und Points.y2 liegen innerhalb des zulässigen Wertebereichs -3.402823e+38 bis 3.402823e+38
- Points.x1, Points.y1, Points.x2 und Points.y2 sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF)

Wenn eine oder mehrere Bedingungen nicht erfüllt sind, dann ist keine korrekte Berechnung des Ausgangswerts möglich. Eine entsprechende Fehlermeldung wird am Parameter ErrorBits ausgegeben.

Die Vorbelegung der x- und y-Werte mit 0.0 stellt keine gültige Konfiguration dar. Ändern Sie die Variablen auf gültige Werte, damit die Variablen für die Berechnung des Ausgangswerts verwendet werden.

Freigabeverhalten EN/ENO

Wenn eine der folgenden Bedingungen erfüllt ist, dann wird der Freigabeausgang ENO auf FALSE gesetzt:

- Freigabeeingang EN ist auf TRUE gesetzt und der Parameter Output ist durch einen Ersatzausgangswert festgelegt bei Fehlermeldungen ErrorBits ≥ 16#0001_0000.
- Freigabeeingang EN ist auf FALSE gesetzt.

Wenn der Freigabeeingang EN auf TRUE gesetzt ist und der Parameter Output folgendermaßen ermittelt werden kann, dann wird der Freigabeausgang ENO auf TRUE gesetzt:

- Mit der gültigen Berechnung des Ausgangswerts bei Reset = FALSE.

oder

- Mit dem gültigen Parameter SubstituteOutput bei Reset = TRUE.

Aufruf

In einem OB wird SplitRange als Einzelinstanz-DB aufgerufen. In einem FB kann SplitRange sowohl als Einzelinstanz-DB, als auch als ein Multiinstanz-DB und als ein Parameterinstanz-DB aufgerufen werden.

Bei dem Aufruf der Anweisung wird kein Technologieobjekt angelegt. Parametrier- und Inbetriebnahmeoberfläche steht Ihnen nicht zur Verfügung. Sie parametrieren SplitRange direkt in dem Instanz-DB und nehmen SplitRange über eine Beobachtungstabelle des Anwenderprogramms in der CPU oder HMI in Betrieb.

Anlauf

Die Variablen im statischen Bereich von SplitRange sind nicht remanent. Diese Variablen werden nach jedem Betriebszustandswechsel der CPU von STOP auf RUN mit den Startwerten initialisiert.

Wenn Sie die Aktualwerte in der Struktur Points im Online-Betrieb ändern und diese Werte nach dem Betriebszustandswechsel der CPU von STOP auf RUN erhalten bleiben sollen, dann sichern Sie diese Werte in den Startwerten des Datenbausteins.

Verhalten im Fehlerfall

Die Anweisung SplitRange erkennt unterschiedliche Fehler, die bei der Berechnung des Ausgangswerts auftreten können. Das Ergebnis der Berechnung kann trotz eines anstehenden Fehlers am Ausgang ausgegeben werden. Wenn keine korrekte Berechnung des Ausgangswerts wegen eines Fehlers möglich ist, dann wird ein Ersatzausgangswert am Ausgang ausgegeben.

An der Variable ErrorMode legen Sie im Fall eines Fehlers, bei dem keine korrekte Berechnung des Ausgangswerts möglich ist, den Ersatzausgangswert folgendermaßen fest:

- Wenn die Variable ErrorMode auf den Wert 0 gesetzt ist, dann gibt SplitRange den Eingangswert am Parameter Input aus.
- Wenn die Variable ErrorMode auf den Wert 1 gesetzt ist, dann gibt SplitRange den Ersatzausgangswert aus. Den Ersatzausgangswert geben Sie am Parameter SubstituteOutput vor.
- Wenn die Variable ErrorMode auf den Wert 2 gesetzt ist, dann gibt SplitRange den letzten gültigen Ausgangswert am Parameter Output aus.

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error auf FALSE gesetzt. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits ist remanent und wird nur durch eine steigende Flanke am Parameter Reset oder ErrorAck zurückgesetzt.

Eingangsparameter SplitRange

Parameter	Datentyp	Vorbelegung	Beschreibung
Input	REAL	0.0	Eingangswert
Substitu-teOutput	REAL	0.0	SubstituteOutput wird als Ersatzausgangswert verwendet, wenn • Reset = TRUE oder • keine korrekte Berechnung des Ausgangswerts wegen eines Fehlers mit Fehlermeldung ErrorBits ≥ 16#0001_0000 möglich ist und ErrorMode auf den Wert 1 konfiguriert ist.
ErrorAck	BOOL	FALSE	Löscht die Fehlermeldungen • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt
Reset	BOOL	FALSE	Führt einen Neustart der Anweisung durch • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt. • Solange Reset auf TRUE gesetzt ist, wird der Ersatzausgangswert SubstituteOutput am Ausgang ausgegeben. • Solange Reset auf FALSE gesetzt ist, wird die Berechnung des Ausgangswerts ausgeführt.

Ausgangsparameter SplitRange

Parameter	Datentyp	Vorbelegung	Beschreibung
Output	REAL	0.0	Ausgangswert
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 3116) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.
Error	BOOL	FALSE	Wenn Error auf TRUE gesetzt ist, steht aktuell mindestens ein Fehler an.

Statische Variablen SplitRange

Variable	Datentyp	Vorbelegung	Beschreibung
Points	AuxFct_Split-Range_Points	-	Stützpunktdaten
Points.x1	REAL	0.0	x-Wert des Stützpunkts 1 Zulässiger Wertebereich: Points.x1 < Points.x2
Points.y1	REAL	0.0	y-Wert des Stützpunkts 1
Points.x2	REAL	0.0	x-Wert des Stützpunkts 2 Zulässiger Wertebereich: Points.x1 < Points.x2

Variable	Datentyp	Vorbelegung	Beschreibung
Points.y2	REAL	0.0	y-Wert des Stützpunkts 2
ErrorMode	INT	0	Auswahl des Ersatzausgangswerts im Fehlerfall • 0 = Input • 1 = SubstituteOutput • 2 = letzter gültiger Ausgangswert Zulässiger Wertebereich: 0 bis 2

Parameter ErrorBits

Wenn gleichzeitig mehrere Fehler anstehen, dann werden die Werte der ErrorBits binär addiert angezeigt. Wenn z. B. ErrorBits = 16#0000_0003 angezeigt wird, dann stehen gleichzeitig die Fehler 16#0000_0001 und 16#0000_0002 an.

Bei SplitRange werden die Fehler, die am Parameter ErrorBits ausgegeben werden, in zwei Kategorien unterschieden:

- Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000
- Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000 anstehen, dann wird der Freigabeausgang ENO nicht geändert. Der Ausgangswert kann trotz dieses Fehlers folgendermaßen ermittelt werden:

- Ausgangswertberechnung auf Basis des Eingangswerts und der Wertepaare in der Struktur Points wenn Reset = FALSE.

Oder

- Ausgabe von SubstituteOutput wenn Reset = TRUE.

ErrorBits (DW#16#...)	Beschreibung
0000_0000	Kein Fehler steht an.
0000_0001	Der Parameter Output wurde begrenzt auf -3.402823e+38 oder +3.402823e+38. Wenn ein Ersatzausgangswert am Parameter Output ausgegeben wird bei Reset = FALSE und ErrorBits ≥ 16#0001_0000, dann überprüfen Sie die Parameter Input oder SubstituteOutput, abhängig von dem eingestellten Wert an der Variable ErrorMode. Wenn Reset = TRUE, dann überprüfen Sie den Parameter SubstituteOutput. Wenn der Parameter Output nicht mehr begrenzt ist, dann wird der Fehler gelöscht, der am Parameter Error gespeichert ist.

Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000 anstehen, dann kann der Ausgangswert nicht wie erwartet ermittelt werden. Stattdessen wird der Ersatzausgangswert ausgegeben. Der Freigabeausgang ENO wird auf FALSE gesetzt.

ErrorBits (DW#16#...)	Beschreibung
0001_0000	Der Parameter SubstituteOutput oder Input, der als Ausgangswert verwendet wird, hat keinen gültigen REAL-Wert. Der Ausgang wird auf 0.0 gesetzt. Mögliche Fehlerszenarios: • Der Parameter Reset ist auf FALSE gesetzt und ein Fehler mit ErrorBits ≥ 16#0002_0000 steht an, der die Berechnung des Ausgangswerts verhindert und – der Parameter Input hat keinen gültigen REAL-Wert, die Variable ErrorMode = 0. – der Parameter SubstituteOutput hat keinen gültigen REAL-Wert, die Variable ErrorMode = 1. • Der Parameter Reset ist auf TRUE gesetzt, während der Parameter SubstituteOutput keinen gültigen REAL-Wert hat. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter SubstituteOutput oder Input auf einen gültigen REAL-Wert geändert wird. oder • der Parameter SubstituteOutput oder Input nicht mehr als Ausgangswert verwendet wird. Wenn der Parameter SubstituteOutput oder Input auf einen gültigen REAL-Wert geändert wird, dann verwendet FB SplitRange den Parameter SubstituteOutput oder Input wieder als Ausgangswert.
0002_0000	Der Parameter Input hat keinen gültigen REAL-Wert, während die Berechnung des Ausgangswerts ausgeführt wird (Reset = FALSE). Der Parameter Output wird mit dem Ersatzausgangswert belegt, den Sie an der Variable ErrorMode konfigurieren. Beachten Sie, dass der Ausgangswert auf 0.0 gesetzt wird, wenn die Variable ErrorMode = 0 gesetzt ist, da der Wert am Parameter Input in diesem Fall nicht als Ersatzausgangswert verwendet wird. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter Input auf einen gültigen REAL-Wert geändert wird. oder • der Parameter Reset auf TRUE gesetzt wird. Wenn der Parameter Input auf einen gültigen REAL-Wert geändert wird und der Parameter Reset auf FALSE gesetzt ist, dann gibt der FB SplitRange am Ausgang wieder den berechneten Ausgangswert aus.

ErrorBits (DW#16#...)	Beschreibung
0004_0000	Eine oder mehrere Variablen in der Struktur Points haben ungültige Werte. Oder die Berechnung des Ausgangswerts ergibt einen ungültigen REAL-Wert für Parameter Output. Der Parameter Output wird mit dem Ersatzausgangswert belegt, den Sie an der Variable ErrorMode konfigurieren. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind: 1. Points.x1 < Points.x2 2. Points.x1, Points.y1, Points.x2 und Points.y2 liegen im zulässigen Wertebereich von -3.402823e+38 bis 3.402823e+38 3. Points.x1, Points.y1, Points.x2 und Points.y2 sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF) Beachten Sie, dass alle Variablen in der Struktur Points nicht remanent sind. Diese Variablen werden nach jedem Betriebszustandswechsel der CPU von STOP auf RUN mit den Startwerten initialisiert. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • die drei oben benannten Bedingungen für die Werte in der Struktur Points erfüllt sind. oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die drei oben benannten Bedingungen für die Werte in der Struktur Points erfüllt sind und der Parameter Reset = FALSE gesetzt ist, dann gibt der FB SplitRange am Ausgang wieder den berechneten Ausgangswert aus.

RampFunction

Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von RampFunction einsetzen können:

CPU	FW	RampFunction
S7-1200	ab V4.2	V1.0
S7-1500	ab V2.0	V1.0

Beschreibung RampFunction

Beschreibung

Die Anweisung RampFunction begrenzt die Änderungsgeschwindigkeit eines Signals. RampFunction gibt einen Signalsprung am Eingang als Rampenfunktion des Ausgangswerts aus.

Setzen Sie RampFunction ein, um Signalsprünge z. B. in folgenden Fällen zu vermeiden:

- Zwischen Sollwertquelle und Sollwerteingang des Reglers, um ein sanfteres Führungsverhalten zu erreichen, ohne Beeinflussung des Störverhaltens.
- Zwischen dem Reglerausgang und Aktoreingang, um den Aktor z. B. einen Motor mit Getriebe, oder den Prozess zu schonen.

Folgende Grenzwerte für die Änderungsgeschwindigkeit sind einstellbar:

- Steigende Änderungsgeschwindigkeit im positiven Wertebereich
- Fallende Änderungsgeschwindigkeit im positiven Wertebereich
- Steigende Änderungsgeschwindigkeit im negativen Wertebereich
- Fallende Änderungsgeschwindigkeit im negativen Wertebereich

Zusätzlich begrenzt die Anweisung RampFunction den Ausgangswert auf den unteren und oberen Grenzwert.

Wenn die Grenze der Änderungsgeschwindigkeit oder die untere oder obere Grenze erreicht wird, dann setzt RampFunction das dazugehörige Ausgangsbit auf TRUE.

Funktionsdiagramm

Das folgende Bild zeigt die Anweisung RampFunction und beispielhaft ein Funktionsdiagramm:

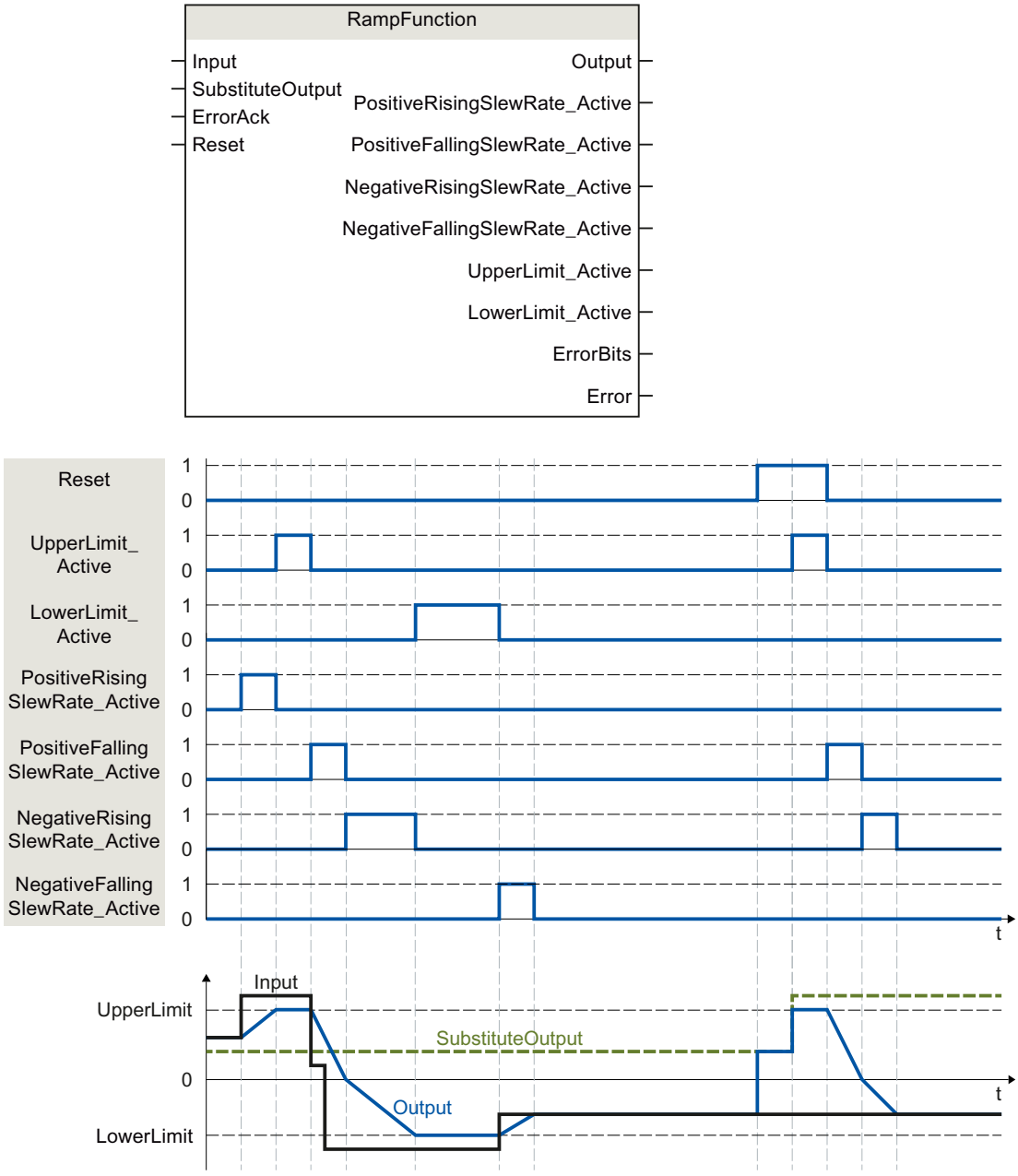

Aufruf

In einem OB wird RampFunction als Einzelinstanz-DB aufgerufen. In einem FB kann RampFunction sowohl als Einzelinstanz-DB, als auch als ein Multiinstanz-DB und als ein Parameterinstanz-DB aufgerufen werden.

Bei dem Aufruf der Anweisung wird kein Technologieobjekt angelegt. Parametrier- und Inbetriebnahmeoberfläche steht Ihnen nicht zur Verfügung. Sie parametrieren RampFunction

direkt in dem Instanz-DB und nehmen RampFunction über eine Beobachtungstabelle des Anwenderprogramms in der CPU oder HMI in Betrieb.

Anlauf

Die Variablen im statischen Bereich von RampFunction sind nicht remanent. Diese Variablen werden nach jedem Betriebszustandswechsel der CPU von STOP auf RUN mit den Startwerten initialisiert.

Wenn Sie die Aktualwerte der Grenzwerte im Online-Betrieb ändern und diese Werte nach dem Betriebszustandswechsel der CPU erhalten bleiben sollen, dann sichern Sie diese Werte in den Startwerten des Datenbausteins.

An der Variable StartMode legen Sie den Initialisierungswert für den Parameter Output fest. Der Initialisierungswert wird beim ersten Aufruf von RampFunction nach dem Betriebszustandswechsel der CPU am Parameter Output ausgegeben. Bei den nachfolgenden Aufrufen berechnet RampFunction den Ausgangswert, ausgehend von diesem Initialisierungswert, auf Basis des Eingangswerts und der Grenzwerte für die Änderungsgeschwindigkeit.

Die folgende Tabelle zeigt die Abhängigkeit zwischen der Variable StartMode und dem Parameter Output. Die Werte in der Spalte Output werden nach dem Betriebszustandswechsel der CPU am Parameter Output ausgegeben:

StartMode	Output	Beispiel
0	Wert des Parameters Input	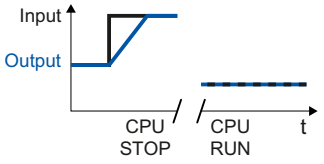
1	Wert des Parameters SubstituteOutput	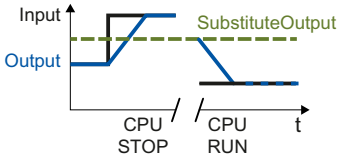
2	Bleibt unverändert. Parameter Output ist remanent.	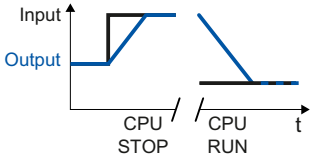
3	0.0	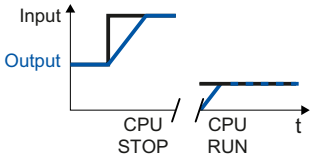

StartMode	Output	Beispiel
4	Wert der Variable LowerLimit	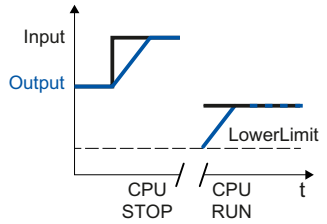
5	Wert der Variable UpperLimit	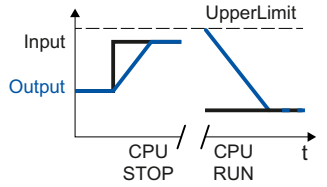

Folgendes gilt zusätzlich für alle Werte der Variable StartMode:

- Wenn die Werte der Variablen UpperLimit und LowerLimit gültig sind, wird der Initialisierungswert auf den Wertebereich dieser Variablen begrenzt. Erst dann wird der Initialisierungswert am Parameter Output ausgegeben.
- Wenn der Initialisierungswert ungültig ist, wird bei dem ersten Aufruf der Anweisung nach dem Betriebszustandswechsel der CPU der Ersatzausgangswert am Parameter Output ausgegeben. Den Ersatzausgangswert konfigurierten Sie an der Variable ErrorMode. Der Ersatzausgangswert wird durch den Wertebereich der Variablen UpperLimit und LowerLimit begrenzt. Bei den nachfolgenden Aufrufen berechnet die Anweisung den Ausgangswert ausgehend von diesem Ersatzausgangswert.
- Die Variable StartMode ist nur dann wirksam, wenn bei dem ersten Aufruf der Anweisung der Parameter Reset = FALSE gesetzt ist. Wenn der Parameter Reset = TRUE gesetzt ist, wird der Wert von Parameter SubstituteOutput am Parameter Output ausgegeben.

Verhalten im Fehlerfall

Die Anweisung RampFunction erkennt unterschiedliche Fehler, die bei der Berechnung des Ausgangswerts auftreten können. Das Ergebnis dieser Berechnung kann trotz eines anstehenden Fehlers am Ausgang ausgegeben werden. Wenn keine korrekte Berechnung des Ausgangswerts wegen eines Fehlers möglich ist, dann wird ein Ersatzausgangswert am Ausgang ausgegeben.

An der Variable ErrorMode legen Sie im Fall eines Fehlers, bei dem keine korrekte Berechnung des Ausgangswerts möglich ist, den Ersatzausgangswert fest.

Die folgende Tabelle zeigt die Abhängigkeit zwischen dem Wert der Variable ErrorMode und dem Ersatzausgangswert, den RampFunction am Parameter Output ausgibt:

ErrorMode	Output
0	Wert des Parameters Input
1	Wert des Parameters SubstituteOutput
2	Der letzte gültige Ausgangswert am Parameter Output
3	0.0

ErrorMode	Output
4	Wert der Variable LowerLimit
5	Wert der Variable UpperLimit

Folgendes gilt zusätzlich für alle Werte der Variable ErrorMode:

- Wenn der Ersatzausgangswert kein gültiger REAL-Wert ist, wird 0.0 als Ausgangswert ausgegeben.
- Wenn die Werte der Variablen UpperLimit und LowerLimit gültig sind, wird der Ersatzausgangswert auf den Wertebereich dieser Variablen begrenzt. Erst dann wird der Ersatzausgangswert am Parameter Output ausgegeben.
- Wenn ein Fehler ansteht, der eine korrekte Berechnung des Ausgangswerts verhindert, dann wechselt RampFunction am Parameter Output von dem berechneten Ausgangswert auf den Ersatzausgangswert. Dabei kann je nach Wert der Variable ErrorMode ein Sprung des Ausgangswerts auftreten.

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error auf FALSE gesetzt. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits ist remanent und wird nur durch eine steigende Flanke am Parameter Reset oder ErrorAck zurückgesetzt.

Arbeitsweise RampFunction

Änderungsgeschwindigkeit begrenzen

Vier Grenzwerte für die Änderungsgeschwindigkeit des Eingangssignals sind konfigurierbar. Welcher Grenzwert momentan wirksam ist, hängt von den folgenden Faktoren ab:

- Vorzeichen des Ausgangswerts am Parameter Output
- Änderungsrichtung des Betrags des Ausgangswerts am Parameter Output

Die folgende Tabelle zeigt die wirksamen Variablen für den Grenzwert der Änderungsgeschwindigkeit in Abhängigkeit des Parameters Output:

Output	Wirksame Variable
Output ≥ 0 und |Output| steigend	PositiveRisingSlewRate
Output ≥ 0 und |Output| fallend	PositiveFallingSlewRate
Output < 0 und |Output| steigend	NegativeRisingSlewRate
Output < 0 und |Output| fallend	NegativeFallingSlewRate

Der Betrag der Grenzwerte der Änderungsgeschwindigkeit definiert die maximale Änderung des Ausgangswerts pro Sekunde.

Beispiel:

Folgendes Szenario gilt für das Beispiel:

- PositiveRisingSlewRate = 10.0
- Aufrufzeit von RampFunction = 0.1 s
- Input > Output ≥ 0.0

Ergebnis:

Der Ausgangswert Output steigt um 1.0 pro Aufruf (10.0 pro Sekunde), bis der Wert am Parameter Input erreicht ist.

Um die Begrenzung der Änderungsgeschwindigkeit für einen oder mehrere Bereiche zu deaktivieren, setzen Sie die entsprechende Variable auf den Wert 3.402823e+38.

Wenn der Ausgangswert Output aktuell durch einen Grenzwert der Änderungsgeschwindigkeit begrenzt ist, dann setzt RampFunction das zugehörige Ausgangsbit auf TRUE:

- PositiveRisingSlewRate_Active
- PositiveFallingSlewRate_Active
- NegativeRisingSlewRate_Active
- NegativeFallingSlewRate_Active

Wenn der Parameter Reset auf TRUE gesetzt ist, dann sind die Grenzwerte der Änderungsgeschwindigkeit nicht wirksam. Dadurch führen Sprünge am Parameter SubstituteOutput zu Sprüngen am Parameter Output.

RampFunction prüft bei jedem Aufruf, ob folgende Bedingungen für die Variablen PositiveRisingSlewRate, PositiveFallingSlewRate, NegativeRisingSlewRate und NegativeFallingSlewRate erfüllt sind:

- Werte liegen innerhalb des zulässigen Wertebereichs größer 0.0 bis 3.402823e+38
- Werte sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF)

Wenn eine oder mehrere Bedingungen nicht erfüllt sind, dann wird der Ersatzausgangswert am Parameter Output ausgegeben. Die entsprechende Fehlermeldung wird am Parameter ErrorBits ausgegeben.

Ausgangswert begrenzen

Der Ausgangswert Output ist immer auf den Wertebereich der Variablen UpperLimit und LowerLimit begrenzt, solange diese Variablen gültige Werte haben.

Wenn der Ausgangswert Output aktuell durch diesen Wertebereich begrenzt ist, dann setzt RampFunction das zugehörige Ausgangsbit auf TRUE:

- UpperLimit_Active
- LowerLimit_Active

Die Begrenzung des Ausgangswerts hat eine höhere Priorität als die Begrenzung der Änderungsgeschwindigkeit. Dadurch führen Änderungen der Variablen UpperLimit und LowerLimit zu Sprüngen des Ausgangswerts Output, wenn dies für die Einhaltung der Grenzwerte der Variablen UpperLimit und LowerLimit erforderlich ist. Die Begrenzung der Änderungsgeschwindigkeit wird in diesem Fall nicht berücksichtigt.

Beispiel:

Wenn der Grenzwert UpperLimit von 100.0 auf 80.0 reduziert wird, während die Werte der Parameter Input und Output bei 90.0 liegen, dann springt der Ausgangswert Output auf 80.0. Der Ausgangswert Output springt auf 80.0 unabhängig davon, ob dadurch der konfigurierte Grenzwert für die Änderungsgeschwindigkeit verletzt wird.

RampFunction prüft bei jedem Aufruf, ob folgende Bedingungen erfüllt sind:

- LowerLimit < UpperLimit
- LowerLimit und UpperLimit liegen innerhalb des zulässigen Wertebereichs -3.402823e+38 bis 3.402823e+38
- LowerLimit und UpperLimit sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF)

Wenn eine oder mehrere Bedingungen nicht erfüllt sind, dann wird der Ersatzausgangswert am Parameter Output ausgegeben. Die entsprechende Fehlermeldung wird am Parameter ErrorBits ausgegeben.

Zusätzlich prüft RampFunction bei jedem Aufruf, ob der Ausgangswert Output den zulässigen Wertebereich eines REAL-Datentyps von -3.402823e+38 bis 3.402823e+38 hat. Wenn die Berechnung des Ausgangswerts einen ungültigen REAL-Wert ergibt, dann wird am Parameter Output der Ersatzausgangswert ausgegeben. Den Ersatzausgangswert konfigurieren Sie an der Variable ErrorMode.

Freigabeverhalten EN/ENO

Wenn eine der folgenden Bedingungen erfüllt ist, dann wird der Freigabeausgang ENO auf FALSE gesetzt:

- Freigabeeingang EN ist auf TRUE gesetzt und der Parameter Output ist durch einen Ersatzausgangswert festgelegt bei Fehlermeldungen ErrorBits ≥ 16#0001_0000.
- Freigabeeingang EN ist auf FALSE gesetzt.

Wenn der Freigabeeingang EN auf TRUE gesetzt ist und der Parameter Output folgendermaßen ermittelt werden kann, dann wird der Freigabeausgang ENO auf TRUE gesetzt:

- Mit dem änderungsgeschwindigkeitsbegrenzten Signal bei Reset = FALSE.

oder

- Mit dem gültigen Parameter SubstituteOutput bei Reset = TRUE.

Zykluszeit automatisch messen

Für die Berechnung der Änderungsgeschwindigkeit des Ausgangswerts benötigt RampFunction die Zeit, die seit dem letzten Aufruf von RampFunction vergangen ist.

Standardmäßig wird die Zykluszeit automatisch gemessen und ab dem zweiten Aufruf an der Variable CycleTime.Value ausgegeben. RampFunction misst die Zykluszeit bei jedem Aufruf der Anweisung und ist somit in nicht äquidistanten Aufrufzyklen einsetzbar, z. B. im OB1.

Wenn die Messung der Zykluszeit kein gültiges Ergebnis liefert, dann berechnet RampFunction den aktuellen Ausgangswert mit der letzten gültigen Zykluszeit. Zusätzlich gibt RampFunction eine Fehlermeldung am Parameter ErrorBits aus.

Wenn Sie die automatische Messung der Zykluszeit mit Setzen der Variable CycleTime.EnableMeasurement = FALSE deaktivieren, dann müssen Sie die Zykluszeit manuell an der Variable CycleTime.Value vorgeben. RampFunction prüft die Variable CycleTime.Value bei jedem Aufruf auf Gültigkeit.

Automatische Messung der Zykluszeit mit Haltepunkten

Wenn Haltepunkte zwischen zwei Aufrufen von RampFunction aktiv sind, ergibt die automatische Messung der Zykluszeit die tatsächliche Zeit, die zwischen zwei Aufrufen vergangen ist. Wenn ein Haltepunkt aktiv ist, dann befindet sich die CPU im Betriebszustand HALT.

Hinweis

Die aktiven Haltepunkte verlängern den Zeitabstand zwischen zwei Aufrufen von RampFunction.

Mit einem längeren Zeitabstand zwischen zwei Aufrufen vergrößert sich auch die maximal zulässige Änderung des Ausgangswerts am Parameter Output.

Beispiel:

Folgendes Szenario gilt für das Beispiel:

- PositiveRisingSlewRate = 10.0
- Aufrufzeit von RampFunction = 0.1 s
- Input > Output ≥ 0.0

Ergebnis ohne Haltepunkte:

Der Ausgangswert Output steigt um 1.0 pro Aufruf, bis der Wert am Parameter Input erreicht ist.

Ergebnis bei einem zehn Sekunden lang aktiven Haltepunkt:

Bei dem nächsten Aufruf steigt der Ausgangswert Output um 100.0.

Wenn Sie die Berechnung des Ausgangswerts basierend auf der tatsächlichen Zeit mit aktiven Haltepunkten nicht benötigen, dann führen Sie folgende Schritte durch:

- Deaktivieren Sie die automatische Messung der Zykluszeit mit Setzen der Variable CycleTime.EnableMeasurement = FALSE.
- Geben Sie die Zykluszeit manuell an der Variable CycleTime.Value ein.

Eingangsparameter RampFunction

Parameter	Datentyp	Vorbelegung	Beschreibung
Input	REAL	0.0	Eingangswert
Substitu-teOutput	REAL	0.0	SubstituteOutput wird als Ersatzausgangswert verwendet, wenn • Reset = TRUE oder • keine korrekte Berechnung des Ausgangswerts wegen eines Fehlers mit Fehlermeldung ErrorBits ≥ 16#0001_0000 möglich ist und ErrorMode auf den Wert 1 konfiguriert ist.

Parameter	Datentyp	Vorbelegung	Beschreibung
ErrorAck	BOOL	FALSE	Löscht die Fehlermeldungen • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt
Reset	BOOL	FALSE	Führt einen Neustart der Anweisung durch • Flanke FALSE -> TRUE ErrorBits wird zurückgesetzt. • Solange Reset auf TRUE gesetzt ist, wird der Ersatzausgangswert SubstituteOutput am Ausgang ausgegeben. • Solange Reset auf FALSE gesetzt ist, wird die Berechnung des Ausgangswerts ausgeführt.

Ausgangsparameter RampFunction

Parameter	Datentyp	Vorbelegung	Beschreibung
Output	REAL	0.0	Ausgangswert
PositiveRisingSlew-Rate_Active	BOOL	FALSE	Wenn PositiveRisingSlewRate_Active = TRUE, dann ist der Ausgangswert aktuell durch PositiveRisingSlewRate begrenzt.
PositiveFallingSlew-Rate_Active	BOOL	FALSE	Wenn PositiveFallingSlewRate_Active TRUE, dann ist der Ausgangswert aktuell durch PositiveFallingSlewRate begrenzt.
NegativeRisingS-lewRate_Active	BOOL	FALSE	Wenn NegativeRisingSlewRate_Active = TRUE, dann ist der Ausgangswert aktuell durch NegativeRisingSlewRate begrenzt.
NegativeFallingS-lewRate_Active	BOOL	FALSE	Wenn NegativeFallingSlewRate_Active = TRUE, dann ist der Ausgangswert aktuell durch NegativeFallingSlewRate begrenzt.
UpperLimit_Active	BOOL	FALSE	Wenn UpperLimit_Active = TRUE, dann ist der Ausgangswert aktuell durch UpperLimit begrenzt.
LowerLimit_Active	BOOL	FALSE	Wenn LowerLimit_Active = TRUE, dann ist der Ausgangswert aktuell durch LowerLimit begrenzt.
ErrorBits	DWORD	DW#16#0	Der Parameter ErrorBits (Seite 3129) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.
Error	BOOL	FALSE	Wenn Error = TRUE, dann steht aktuell mindestens ein Fehler an.

Statische Variablen RampFunction

Variable	Datentyp	Vorbelegung	Beschreibung
PositiveRisingSlewRate	REAL	10.0	Grenzwert für die Änderungsgeschwindigkeit des Ausgangswerts pro Sekunde im positiven Bereich bei steigendem Betrag Mit PositiveRisingSlewRate = 3.402823e+38 ist diese Begrenzung der Änderungsgeschwindigkeit deaktiviert. Zulässiger Wertebereich: > 0.0
PositiveFallingSlewRate	REAL	10.0	Grenzwert für die Änderungsgeschwindigkeit des Ausgangswerts pro Sekunde im positiven Bereich bei fallendem Betrag Mit PositiveFallingSlewRate = 3.402823e+38 ist diese Begrenzung der Änderungsgeschwindigkeit deaktiviert. Zulässiger Wertebereich: > 0.0
NegativeRisingSlewRate	REAL	10.0	Grenzwert für die Änderungsgeschwindigkeit des Ausgangswerts pro Sekunde im negativen Bereich bei steigendem Betrag Mit NegativeRisingSlewRate = 3.402823e+38 ist diese Begrenzung der Änderungsgeschwindigkeit deaktiviert Zulässiger Wertebereich: > 0.0
NegativeFallingSlewRate	REAL	10.0	Grenzwert für die Änderungsgeschwindigkeit des Ausgangswerts pro Sekunde im negativen Bereich bei fallendem Betrag Mit NegativeFallingSlewRate = 3.402823e+38 ist diese Begrenzung der Änderungsgeschwindigkeit deaktiviert. Zulässiger Wertebereich: > 0.0
UpperLimit	REAL	100.0	Obergrenze des Ausgangswerts Zulässiger Wertebereich: > LowerLimit
LowerLimit	REAL	0.0	Untergrenze des Ausgangswerts Zulässiger Wertebereich: < UpperLimit
ErrorMode	INT	2	Auswahl des Ersatzausgangswerts im Fehlerfall • 0 = Input • 1 = SubstituteOutput • 2 = letzter gültiger Ausgangswert • 3 = 0.0 • 4 = LowerLimit • 5 = UpperLimit Zulässiger Wertebereich: 0 bis 5
StartMode	INT	2	Auswahl des Ausgangswerts für den ersten Aufruf der Anweisung • 0 = Input • 1 = SubstituteOutput • 2 = letzter Ausgangswert • 3 = 0.0 • 4 = LowerLimit • 5 = UpperLimit Zulässiger Wertebereich: 0 bis 5

Variable	Datentyp	Vorbelegung	Beschreibung
CycleTime	AuxFct_Cycle-Time	-	Daten der Zykluszeit
CycleTime.Value	REAL	0.1	Zeitabstand zwischen zwei Aufrufen der Anweisung in Sekunden Zulässiger Wertebereich: > 0.0
CycleTime.Enable-Measurement	BOOL	TRUE	Automatische Messung der Zykluszeit aktivieren

Parameter ErrorBits

Wenn gleichzeitig mehrere Fehler anstehen, dann werden die Werte der ErrorBits binär addiert angezeigt. Wenn z. B. ErrorBits = 16#0000_0003 angezeigt wird, dann stehen gleichzeitig die Fehler 16#0000_0001 und 16#0000_0002 an.

Die Anweisung RampFunction unterscheidet Fehler, die am Parameter ErrorBits ausgegeben werden, in zwei Kategorien:

- Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000
- Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits < 16#0001_0000 anstehen, dann wird der Freigabeausgang ENO nicht geändert. Der Ausgangswert kann trotz dieses Fehlers folgendermaßen ermittelt werden:

- Ausgangswertberechnung auf Basis des Eingangswerts und der Grenzwerte für die Änderungsgeschwindigkeit wenn Reset = FALSE.

oder

- Ausgabe von SubstituteOutput wenn Reset = TRUE.

ErrorBits (DW#16#...)	Beschreibung
0000_0000	Kein Fehler steht an.
0000_0002	Die Messung der Zykluszeit ergibt einen ungültigen Wert, während die Berechnung des Ausgangswerts ausgeführt wird (Reset = FALSE). Wenn bereits ein gültiger Wert der Zykluszeit gemessen wurde, dann berechnet RampFunction den Ausgangswert ausgehend von diesem letzten Wert der Variable CycleTime.Value. Wenn zuvor kein gültiger Wert der Zykluszeit gemessen wurde, dann gibt RampFunction weiterhin den Ausgangswert am Parameter Output aus, der mit der Variable StartMode konfiguriert ist. Das Fehlerbit am Parameter Error wird gelöscht, wenn • die Messung der Zykluszeit einen gültigen Wert ergibt oder • der Parameter Reset auf TRUE gesetzt wird.

Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000

Wenn ein oder mehrere Fehler mit Fehlermeldungen ErrorBits ≥ 16#0001_0000 anstehen, dann kann der Ausgangswert nicht wie erwartet berechnet werden. Stattdessen wird der Ersatzausgangswert ausgegeben. Der Freigabeausgang ENO wird auf FALSE gesetzt.

Die Begrenzung des Ausgangswerts bleibt wirksam, solange die Variablen LowerLimit und UpperLimit gültige Werte haben.

Die Begrenzung der Änderungsgeschwindigkeit ist nicht mehr wirksam. In einem der folgenden Szenarios können Sprünge am Ausgangswert auftreten:

- In dem Moment, in dem der Fehler erkannt wird, schaltet RampFunction von dem berechneten Ausgangswert auf den Ersatzausgangswert um. Ob dabei ein Sprung auftritt, ist abhängig von dem Wert der Variable ErrorMode.
- Der Ersatzausgangswert wird geändert, während dieser wirksam ist.

ErrorBits (DW#16#...)	Beschreibung
0001_0000	Der Parameter SubstituteOutput oder eine andere Variable, die als Ausgangswert verwendet wird, hat keinen gültigen REAL-Wert. Der Ausgangswert am Parameter Output wird deshalb auf 0.0 gesetzt und durch die Variablen LowerLimit und UpperLimit begrenzt. Mögliche Fehlerszenarios: • Der Parameter Reset ist auf FALSE gesetzt und ein Fehler mit ErrorBits ≥ 16#0002_0000 steht an, der die Berechnung des Ausgangswerts verhindert und – der Parameter Input hat keinen gültigen REAL-Wert, die Variable ErrorMode = 0. – der Parameter SubstituteOutput hat keinen gültigen REAL-Wert, die Variable ErrorMode = 1. – die Variable LowerLimit hat keinen gültigen REAL-Wert, die Variable ErrorMode = 4. – die Variable UpperLimit hat keinen gültigen REAL-Wert, die Variable ErrorMode = 5. • Der Parameter Reset ist auf TRUE gesetzt, während der Parameter SubstituteOutput keinen gültigen REAL-Wert hat. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter SubstituteOutput, Input, LowerLimit oder UpperLimit auf einen gültigen REAL-Wert geändert wird oder • der Parameter SubstituteOutput, Input, LowerLimit oder UpperLimit nicht mehr als Ausgangswert verwendet wird. Wenn der Parameter SubstituteOutput, Input, LowerLimit oder UpperLimit auf einen gültigen REAL-Wert geändert wird, dann gibt RampFunction den Wert dieses Parameters wieder als Ausgangswert aus.
0002_0000	Der Parameter Input hat keinen gültigen REAL-Wert, während die Berechnung des Ausgangswerts ausgeführt wird (Reset = FALSE). Am Parameter Output wird der Ersatzausgangswert ausgegeben, der an der Variable ErrorMode konfiguriert ist und durch die Variablen UpperLimit und LowerLimit begrenzt ist. Beachten Sie, dass der Ausgangswert auf 0.0 gesetzt wird, wenn die Variable ErrorMode = 0 gesetzt ist, da der Wert am Parameter Input in diesem Fall nicht als Ersatzausgangswert verwendet wird. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • der Parameter Input auf einen gültigen REAL-Wert geändert wird oder • der Parameter Reset auf TRUE gesetzt wird. Wenn der Parameter Input auf einen gültigen REAL-Wert geändert wird und der Parameter Reset auf FALSE gesetzt ist, dann gibt RampFunction am Parameter Output wieder den berechneten Ausgangswert aus.

ErrorBits (DW#16#...)	Beschreibung
0004_0000	Die Berechnung des Ausgangswerts ergibt einen ungültigen REAL-Wert am Parameter Output. Am Parameter Output wird der Ersatzausgangswert ausgegeben, der an der Variable ErrorMode konfiguriert ist und durch die Variablen UpperLimit und LowerLimit begrenzt ist. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • die Berechnung des Ausgangswerts wieder gültige Werte am Parameter Output ausgibt oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die Berechnung des Ausgangswerts wieder gültige REAL-Werte ergibt und der Parameter Reset auf FALSE gesetzt ist, dann gibt RampFunction am Parameter Output wieder den berechneten Ausgangswert aus.
0008_0000	Die Variable LowerLimit oder UpperLimit hat einen ungültigen Wert. Am Parameter Output wird folgender Wert ausgegeben, abhängig von dem Parameter Reset: • Reset = FALSE Am Parameter Output wird der Ersatzausgangswert ausgegeben, der mit der Variable ErrorMode konfiguriert ist. • Reset = TRUE Am Parameter Output wird der Wert des Parameters SubstituteOutput ausgegeben. In beiden Fällen wird der Parameter Ouput auf den Wertebereich des REAL-Datentyps von -3.402823e+38 bis 3.402823e+38 begrenzt. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind: 1. LowerLimit < UpperLimit 2. LowerLimit und UpperLimit liegen innerhalb des zulässigen Wertebereichs -3.402823e+38 bis 3.402823e+38 3. LowerLimit und UpperLimit sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF) Beachten Sie, dass der Ausgangswert am Parameter Output auf 0.0 gesetzt wird, wenn Reset = FALSE und • die Variable LowerLimit hat keinen gültigen REAL-Wert, die Variable ErrorMode = 4 oder • die Variable UpperLimit hat keinen gültigen REAL-Wert, die Variable ErrorMode = 5. In diesem Fall werden die Werte der Variablen LowerLimit und UpperLimit nicht als Ersatzausgangswerte am Parameter Output ausgegeben, da diese ungültig sind. Wenn die drei oben benannten Bedingungen für Variablen LowerLimit und UpperLimit erfüllt sind, dann gilt Folgendes: • Das Fehlerbit am Parameter Error wird gelöscht. • Der Freigabeausgang ENO wird wieder auf TRUE gesetzt. • RampFunction gibt am Parameter Output wieder den Ausgangswert aus, bei – Reset = FALSE, durch die Berechnung und Begrenzung des Ausgangswerts. – Reset = TRUE, durch den begrenzten Wert am Parameter SubstituteOutput.

ErrorBits (DW#16#...)	Beschreibung
0010_0000	Mindestens einer der folgenden Parameter hat ungültige Werte, während die Berechnung des Ausgangswerts ausgeführt wird (Reset = FALSE): 1. PositiveRisingSlewRate 2. PositiveFallingSlewRate 3. NegativeRisingSlewRate 4. NegativeFallingSlewRate Am Parameter Output wird der Ersatzausgangswert ausgegeben, der an der Variable ErrorMode konfiguriert ist und durch die Variablen UpperLimit und LowerLimit begrenzt ist. Stellen Sie sicher, dass folgende Bedingungen für alle vier oben benannten Parameter erfüllt sind: • Die Werte liegen innerhalb des zulässigen Wertebereichs größer 0.0 bis 3.402823e+38 • Die Werte sind gültige REAL-Werte (≠ NaN z. B. 16#7FFF_FFFF) Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • die oben benannten Bedingungen für die vier Parameter erfüllt sind oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die oben benannten Bedingungen für die vier Parameter erfüllt sind und Reset auf FALSE gesetzt ist, dann gibt RampFunction am Parameter Output wieder den berechneten Ausgangswert aus.

ErrorBits (DW#16#...)	Beschreibung
0020_0000	Die Variable (konfiguriert mit StartMode) für die Initialisierung des Parameters Output bei dem ersten Aufruf der Anweisung hat keinen gültigen REAL-Wert. Bei dem ersten Aufruf der Anweisung wird der Ersatzausgangswert am Parameter Output ausgegeben, der an der Variable ErrorMode konfiguriert ist und durch die Variablen LowerLimit und UpperLimit begrenzt ist. Bei den nachfolgenden Aufrufen berechnet RampFunction den Ausgangswert ausgehend von diesem Ersatzausgangswert. Dieser Fehler tritt bei dem ersten Aufruf der Anweisung nach dem Betriebszustandswechsel der CPU von STOP auf RUN auf, wenn Reset = FALSE gesetzt ist und • der Parameter SubstituteOutput hat keinen gültigen REAL-Wert, bei StartMode = 1. • der Parameter Output hat keinen gültigen REAL-Wert, bei StartMode = 2. Bei den nachfolgenden Aufrufen gilt Folgendes: • Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt. • RampFunction gibt am Parameter Output wieder den berechneten Ausgangswert aus.
0040_0000	Die Variable CycleTime.Value hat einen ungültigen Wert, während die Berechnung des Ausgangswerts ausgeführt wird (Reset = FALSE). Am Parameter Output wird der Ersatzausgangswert ausgegeben, der an der Variable ErrorMode konfiguriert ist und durch die Variablen UpperLimit und LowerLimit begrenzt ist. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind: • 0.0 < CycleTime.Value < 3.402823e+38 • CycleTime.Value ist ein gültiger REAL-Wert (≠ NaN z. B. 16#7FFF_FFFF) Für eine automatische Berechnung des Werts der Variable CycleTime.Value, setzen Sie die Variable CycleTime.EnableMeasurement auf TRUE. Das Fehlerbit am Parameter Error wird gelöscht und der Freigabeausgang ENO wieder auf TRUE gesetzt, wenn • die oben benannten Bedingungen für die Variable CycleTime.Value erfüllt sind oder • der Parameter Reset auf TRUE gesetzt wird. Wenn die oben benannten Bedingungen für die Variable CycleTime.Value erfüllt sind und Reset auf FALSE gesetzt ist, dann gibt RampFunction am Parameter Output wieder den berechneten Ausgangswert aus.

4.1.5.4 Motion Control

S7-1200 Motion Control

S7-1200 Motion Control ab V6

MC_Power

MC_Power: Achse freigeben, sperren ab V6

Beschreibung

Die Motion Control-Anweisung "MC_Power" gibt eine Achse frei, bzw. sperrt diese.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Es steht kein freigabeverhindernder Fehler an.

Ablöseverhalten

Die Bearbeitung des "MC_Power" kann durch keinen Motion Control-Auftrag abgebrochen werden.

Mit dem Sperren der Achse (Eingangsparameter "Enable" = FALSE) werden alle Motion Control-Aufträge entsprechend dem gewählten "StopMode" am zugehörigen Technologieobjekt abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse	
Enable	INPUT	BOOL	FALSE	TRUE	Die Achse wird freigegeben.
				FALSE	Alle laufenden Aufträge werden entsprechend dem parametrierten "StopMode" abgebrochen. Die Achse wird gestoppt und gesperrt.
StartMode	INPUT	INT	1	0	Positionierachse nicht lagegeregelt freigeben *)
				1	Positionierachse lagegeregelt freigeben *)
				*) Beim Einsatz einer Positionierachse mit PTO (Pulse Train Output) -Antrieb wird der Parameter ignoriert. Der Parameter wirkt initial beim Freigeben der Positionierachse (Enable wechselt von FALSE nach TRUE) und beim Freigeben nach dem erfolgreichen Quittieren eines Alarms, der zum Sperren der Achse geführt hat.	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
StopMode	INPUT	INT	0	0	Notstopp Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Beim Erreichen des Stillstands wird die Achse gesperrt.
				1	Sofort-Aus Steht eine Anforderung zum Sperren der Achse an, so wird der Sollwert Null ausgegeben und die Achse gesperrt. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Bei Antriebsanbindung über PTO (Pulse Train Output): Beim Sperren der Achse wird die Impulsausgabe betriebsbedingt mit einer frequenzabhängigen Verzögerung gestoppt: • Ausgangsfrequenz ≥ 100 Hz Verzögerung: max. 30 ms • Ausgangsfrequenz < 100 Hz Verzögerung: 30 ms bis max. 1,5 s bei 2 Hz
				2	Notstopp mit Ruckbegrenzung Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Ist die Ruckbegrenzung aktiviert, so wird der konfigurierte Ruck berücksichtigt. Beim Erreichen des Stillstands wird die Achse gesperrt.
Status	OUTPUT	BOOL	FALSE	Status der Achsfreigabe	
				FALSE	Die Achse ist gesperrt Die Achse führt keine Motion Control-Aufträge aus und nimmt keine neuen Aufträge an (Ausnahme: MC_Reset-Auftrag). Bei Antriebsanbindung über PTO (Pulse Train Output): Die Achse ist nicht referenziert. Beim Sperren wechselt der Status erst mit dem Erreichen des Stillstands der Achse auf FALSE.
				TRUE	Die Achse ist freigegeben Die Achse ist bereit, Motion Control-Aufträge auszuführen. Beim Freigeben der Achse wechselt der Status erst mit dem Anstehen des Signals "Antrieb bereit" auf TRUE. Wurde das Antriebssignal "Antrieb bereit" in der Achskonfiguration nicht konfiguriert, so wechselt der Status sofort nach TRUE.
Busy	OUTPUT	BOOL	FALSE	TRUE	"MC_Power" ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" oder am zugehörigen Technologieobjekt ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"

Hinweis

Wird die Achse durch einen Fehler ausgeschaltet, so wird sie nach dem Beseitigen und Quittieren des Fehlers automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Eingangsparameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Freigeben einer Achse mit konfigurierten Antriebssignalen

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StartMode" und "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE.
   Der Freigabe-Ausgang für die "Antriebsfreigabe" wechselt auf TRUE, um dem Antrieb die Leistungsfreigabe zu erteilen. Die CPU wartet auf das Signal "Antrieb bereit" des Antriebs. Mit dem Anstehen des Signals "Antrieb bereit" am konfigurierten Bereit-Eingang der CPU wird die Achse freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Freigeben einer Achse ohne konfigurierte Antriebssignale

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StartMode" und "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE. Die Achse wird freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Sperren einer Achse

Zum Sperren einer Achse können Sie wie nachfolgend beschrieben vorgehen:

1. Bringen Sie die Achse in den Stillstand.
   Wann sich die Achse im Stillstand befindet, erkennen Sie an der Variable des Technologieobjekts <Achsname>.StatusBits.StandStill.
2. Setzen Sie den Eingangsparameters "Enable" nach dem Erreichen des Stillstandes auf FALSE.
3. Zeigen die Ausgangsparameter "Busy" und "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable den Wert FALSE, so ist das Sperren der Achse abgeschlossen.

Siehe auch

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_Power: Funktionsdiagramm ab V6

Funktionsdiagramm

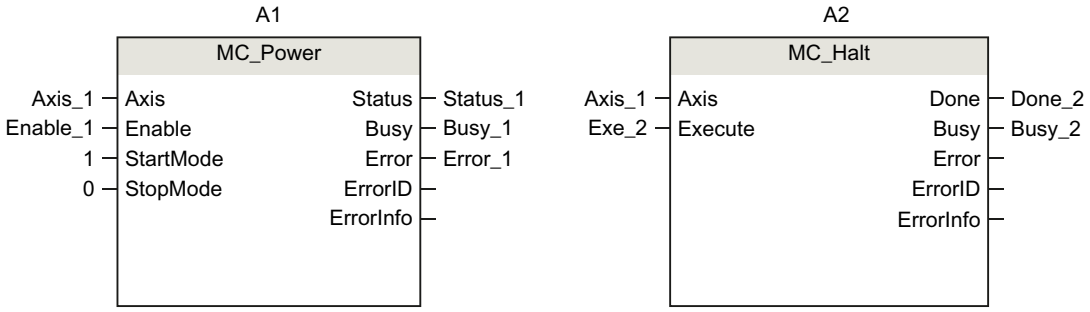

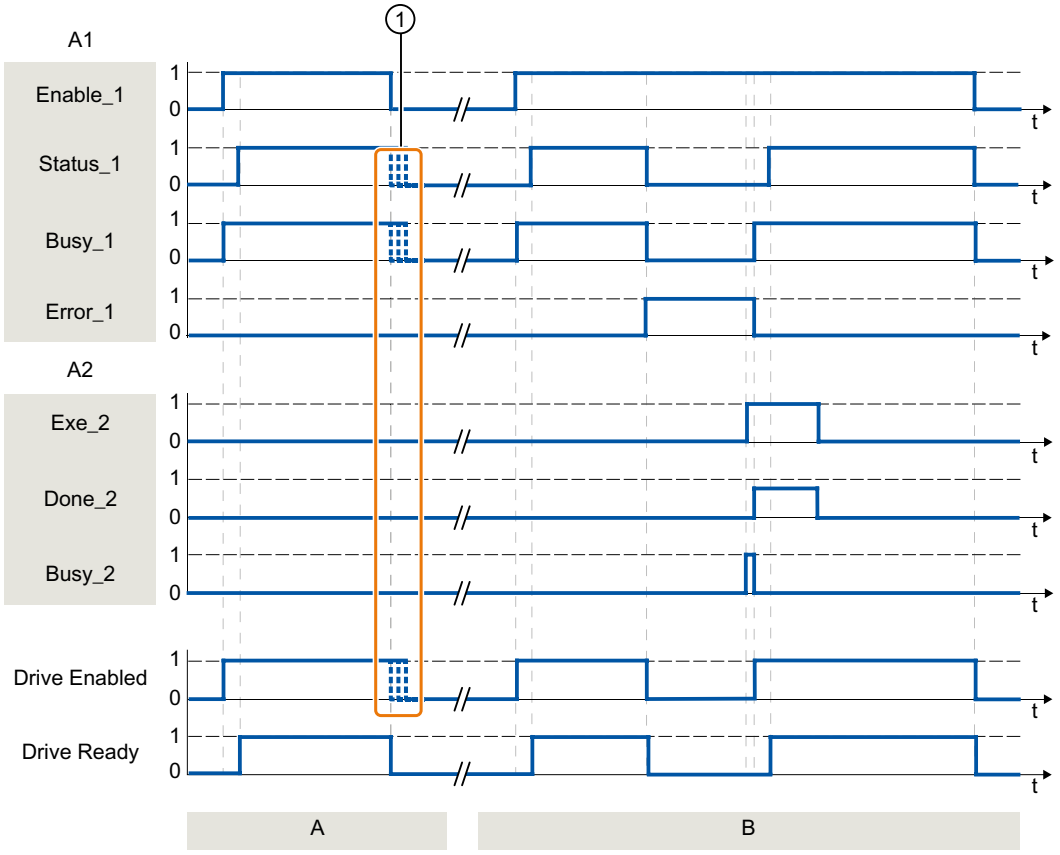

Abschnitt A	Eine Achse wird freigegeben und anschließend wieder gesperrt. Nachdem der Antrieb das Signal "Antrieb bereit" an die CPU zurück gemeldet hat, kann die erfolgreiche Freigabe über "Status_1" abgelesen werden.
Abschnitt B	Nach dem Freigeben einer Achse tritt ein Fehler auf, der das Sperren der Achse zur Folge hat. Der Fehler wird beseitigt und mit "MC_Reset" quittiert. Daraufhin wird die Achse wieder freigegeben.
①	Das exakte Ende der Signale hängt vom gewählten Antrieb und vom StopMode ab.

MC_Reset

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" können "Betriebsfehler mit Stopp der Achse" und "Konfigurationsfehler" quittiert werden. Welche Fehler quittiert werden müssen können Sie der "Liste der ErrorIDs und ErrorInfos" unter "Abhilfe" entnehmen.

Nach einem Laden im Betriebszustand RUN kann die Konfiguration der Achse in den Arbeitsspeicher geladen werden.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Bei einem anstehenden quittierbaren Konfigurationsfehler wurde die Fehlerursache beseitigt (z. B. die Beschleunigung im Technologieobjekt Positionierachse auf einen gültigen Wert geändert).

Ablöseverhalten

Der MC_Reset-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der neue MC_Reset-Auftrag bricht keine anderen laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Restart	INPUT	BOOL	FALSE	TRUE	Lädt die Konfiguration der Achse aus dem Ladespeicher in den Arbeitsspeicher. Der Auftrag ist nur bei gesperrter Achse ausführbar. Beachten Sie hierzu die Hinweise zum Laden in die CPU.
				FALSE	Quittiert anstehende Fehler
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Einen mit MC_Reset zu quittierenden Fehler quittieren

Gehen Sie zum Quittieren eines Fehlers folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" den Wert TRUE sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Error den Wert FALSE, so wurde der Fehler quittiert.

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_Home

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6

Beschreibung

Mit Hilfe der Motion Control-Anweisung "MC_Home" gleichen Sie die Achskoordinate auf die reale, physikalische Position des Antriebs ab. Soll die Achse absolut positioniert werden, so ist eine Referenzierung nötig. Folgende Referenzierarten können durchgeführt werden:

- Aktives Referenzieren (Mode = 3)
  Die Referenzpunktfahrt wird automatisch durchgeführt.
- Passives Referenzieren (Mode = 2)
  Beim passiven Referenzieren führt die Motion Control-Anweisung "MC_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse referenziert.

- Direktes Referenzieren Absolut (Mode = 0)
  Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.
- Direktes Referenzieren Relativ (Mode = 1)
  Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.
- Absolutwertgeberjustage Relativ (Mode = 6)
  Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.
- Absolutwertgeberjustage Absolut (Mode = 7)
  Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.

Mode 6 und 7 sind nur bei Antrieben mit anloger Antriebschnittstelle und PROFIdrive-Antrieb einsetzbar.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben. (Gilt nicht für PROFIdrive-Antrieb/Analoge Antriebsanbindung Mode = 0 oder 1)
- Beim Start mit Mode = 0, 1 und 2 darf kein MC_CommandTable-Auftrag aktiv sein.

Ablöseverhalten

Das Ablöseverhalten ist vom gewählten Mode abhängig:

Mode = 0, 1, 6, 7

Der MC_Home-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der MC_Home-Auftrag bricht keine laufenden Motion Control-Aufträge ab. Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 2

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 2, 3

Der neue MC_Home-Auftrag bricht folgenden laufenden Motion Control-Auftrag ab:

- MC_Home-Auftrag Mode = 2

Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 3

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag

- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Home-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 2, 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	REAL	0.0	• Mode = 0, 2 und 3 Absolute Position der Achse nach dem Abschluss des Referenziervorgangs • Mode = 1 Korrekturwert zur aktuellen Achsposition Grenzwerte: -1.0E12 ≤ Position ≤ 1.0E12

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	INT	0	Referenziermodus	
				0	Direktes Referenzieren (Absolut) Neue Achsposition ist der Positionswert des Parameters "Position".
				1	Direktes Referenzieren (Relativ) Neue Achsposition ist die aktuelle Achsposition + Positionswert des Parameters "Position".
				2	Passives Referenzieren Referenzieren gemäß der Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
				3	Aktives Referenzieren Referenzpunktfahrt gemäß Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<Achsname>.StatusSensor.AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<Achsname>.StatusSensor.AbsEncoderOffset)
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	
ReferenceMarkPosition	OUTPUT	REAL	0.0	Anzeige der Position, an der das Technologieobjekt referenziert wurde (gültig bei "Done" = TRUE)	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjekts (<Achsname>.StatusBits.HomingDone) wird unter folgenden Bedingungen rückgesetzt:

- **Antriebsanbindung über PTO (Pulse Train Output):**
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Sperren der Achse durch die Motion Control-Anweisung "MC_Power"
  - Wechsel zwischen Automatikbetrieb und Handsteuerung
  - Nach NETZ-AUS → NETZ-EIN der CPU
  - Nach Neustart der CPU (RUN-STOP → STOP-RUN)
- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS → NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Eine Achse referenzieren

Gehen Sie beim Referenzieren der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die notwendigen Eingangsparameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" sowie die Variable des Technologieobjekts <Achsname>.StatusBits.HomingDone den Wert TRUE, so ist das Referenzieren abgeschlossen. Die Referenzposition kann der Variable <Achsname>.ReferenceMarkPosition entnommen werden.

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_Halt

MC_Halt: Achse anhalten ab V6

Beschreibung

Die Motion Control-Anweisung "MC_Halt" bricht alle Bewegungsvorgänge ab und bremst die Achse mit der konfigurierten Verzögerung bis zum Stillstand. Die Stillstandsposition ist nicht definiert.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_Halt-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Halt-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit Null erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennnung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_Halt: Funktionsdiagramm ab V6

Funktionsdiagramm

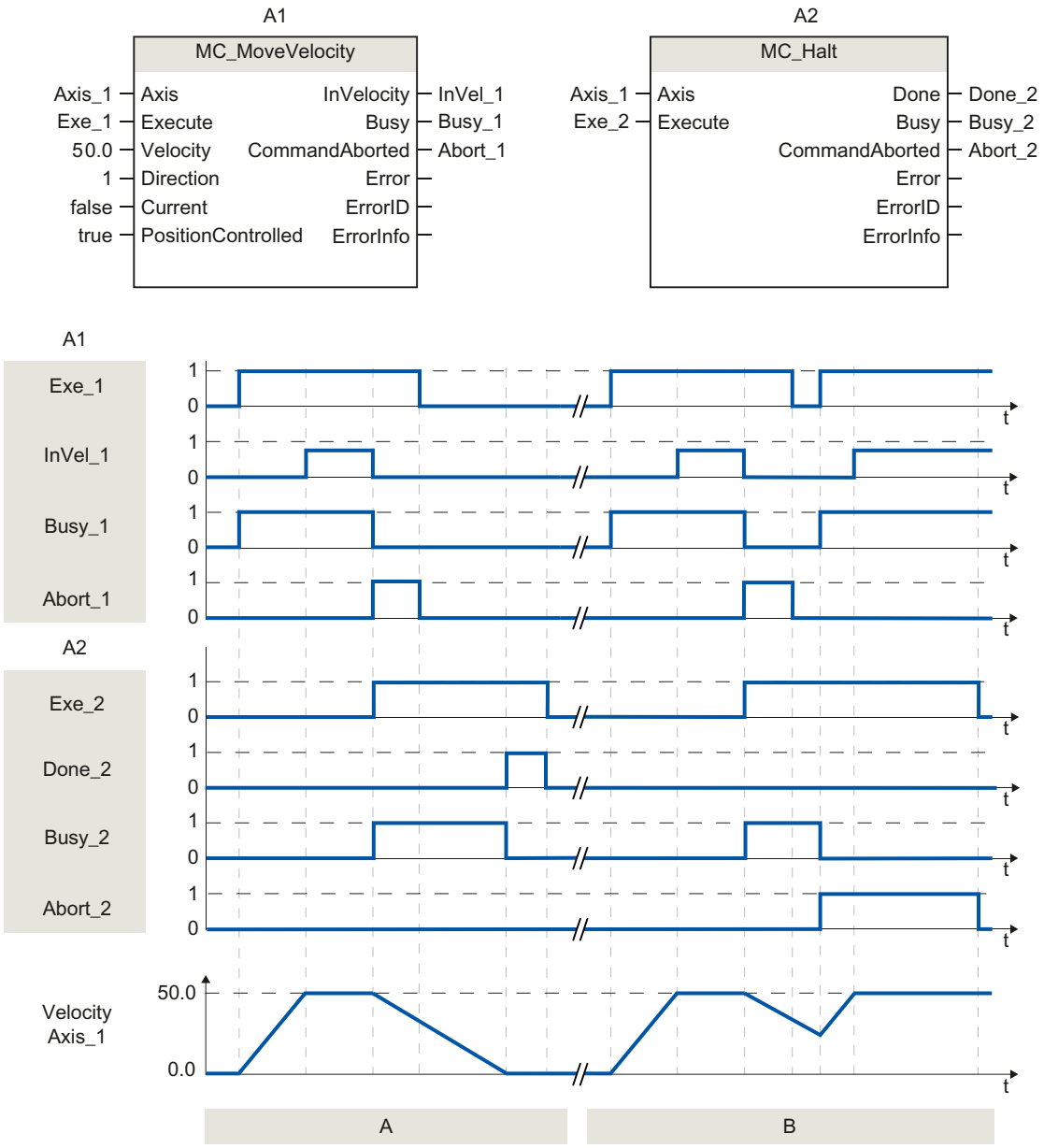

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

Abschnitt A	Die Achse wird durch einen MC_Halt-Auftrag bis zum Stillstand abgebremst. Das Erreichen des Stillstandes wird über "Done_2" gemeldet.
Abschnitt B	Während ein MC_Halt-Auftrag die Achse abbremst, wird dieser Auftrag durch einen anderen Verfahrauftrag abgebrochen. Der Abbruch wird über "Abort_2" gemeldet.

MC_MoveAbsolute

MC_MoveAbsolute: Achse absolut positionieren ab V6

Beschreibung

Die Motion Control-Anweisung "MC_MoveAbsolute" startet eine Positionierbewegung der Achse zu einer absoluten Position.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Die Achse ist referenziert.

Ablöseverhalten

Der MC_MoveAbsolute-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveAbsolute-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag

- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_PositioningAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Position	INPUT	REAL	0.0	Absolute Zielposition Grenzwerte: -1.0E12 ≤ Position ≤ 1.0E12	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der anzufahrenden Zielposition nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Direction	INPUT	INT	1	Bewegungsrichtung der Achse Wird nur bei aktiviertem "Modulo" ausgewertet. "Technologieobjekt > Konfiguration > Erweiterte Parameter > Modulo > Modulo aktivieren" Parameter wird bei PTO-Achsen ignoriert.	
				0	Das Vorzeichen der Geschwindigkeit (Parameter "Velocity") bestimmt die Bewegungsrichtung.
				1	Positive Richtung (Zielposition wird in positiver Richtung angefahren)
				2	Negative Richtung (Zielposition wird in negativer Richtung angefahren)
				3	Kürzester Weg (Die Technologie wählt, ausgegehend von der aktuellen Position, den kürzesten Weg zur Zielposition)
Done	OUTPUT	BOOL	FALSE	TRUE	Absolute Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_MoveAbsolute: Funktionsdiagramm ab V6

Funktionsdiagramm

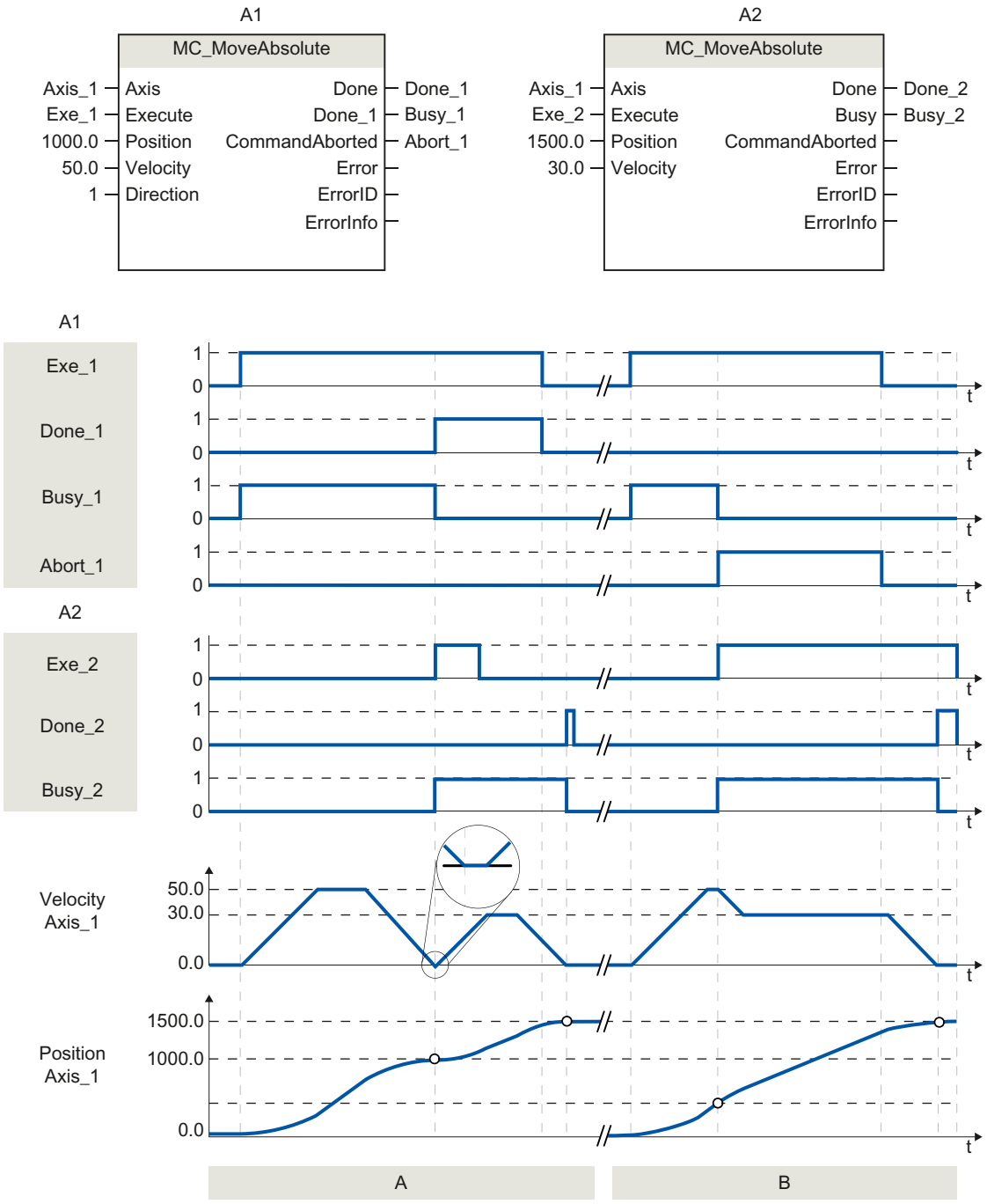

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

Abschnitt A	Eine Achse wird durch einen MC_MoveAbsolute-Auftrag auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveAbsolute-Auftrag, mit Zielposition 1500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
Abschnitt B	Ein laufender MC_MoveAbsolute-Auftrag wird durch einen weiteren MC_MoveAbsolute-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveRelative

MC_MoveRelative: Achse relativ positionieren ab V6

Beschreibung

Die Motion Control-Anweisung "MC_MoveRelative" startet eine Positionierbewegung relativ zur Startposition.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveRelative-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveRelative-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag

- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Distance	INPUT	REAL	0.0	Wegstrecke für den Positioniervorgang Grenzwerte: -1.0E12 ≤ Distance ≤ 1.0E12	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der zu fahrenden Wegstrecke nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_MoveRelative: Funktionsdiagramm ab V6

Funktionsdiagramm

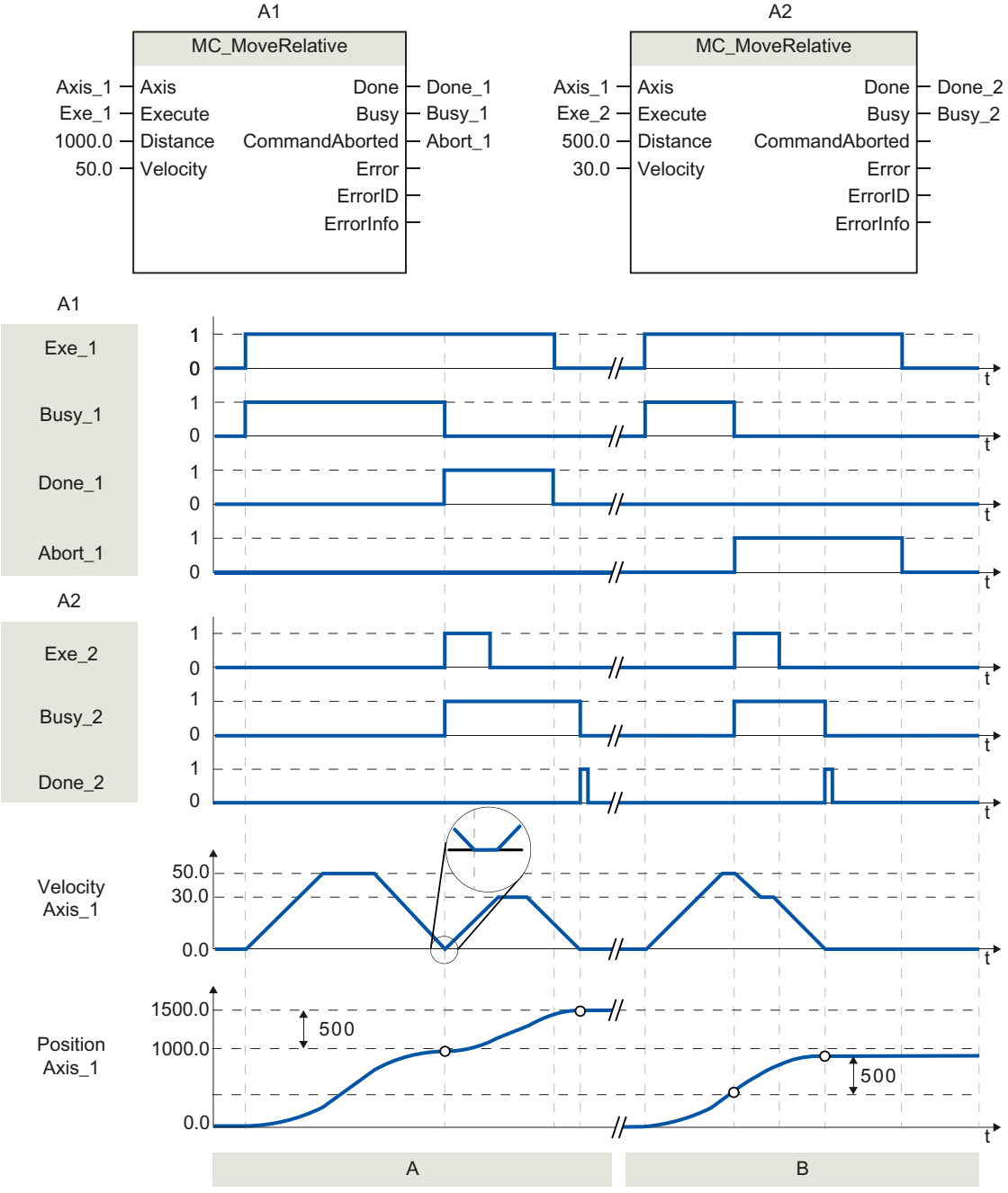

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

Abschnitt A	Die Achse wird durch einen MC_MoveRelative-Auftrag um die Wegstrecke ("Distance") 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveRelative-Auftrag, mit der Wegstrecke 500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
Abschnitt B	Ein laufender MC_MoveRelative-Auftrag wird durch einen weiteren MC_MoveRelative-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die neue Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveVelocity

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6

Beschreibung

Die Motion Control-Anweisung "MC_MoveVelocity" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveVelocity kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveVelocity-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag

- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für das Verfahren der Achse Grenzwerte: Start/Stopp-Geschwindigkeit ≤ |Velocity| ≤ Maximale Geschwindigkeit (Velocity = 0.0 ist erlaubt)	
Direction	INPUT	INT	0	Richtungsvorgabe	
				0	Drehrichtung entsprechend dem Vorzeichen des Werts am Parameter "Velocity"
				1	Drehrichtung positiv (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
				2	Drehrichtung negativ (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	"Aktuelle Geschwindigkeit beibehalten" deaktiviert. Es werden die Werte der Parameter "Velocity" und "Direction" verwendet.
				TRUE	"Aktuelle Geschwindigkeit beibehalten" aktiviert. Die Werte an den Parametern "Velocity" und "Direction" bleiben unberücksichtigt. Sobald die Achse mit der aktuellen Geschwindigkeit weiterfährt liefert der Parameter "InVelocity" den Wert TRUE.
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der MC_MoveVelocity-Auftrag ausgeführt wird. Danach gilt wieder die Einstellung des MC_Power. Beim Einsatz einer PTO-Achse wird der Parameter ignoriert.	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	• "Current" = FALSE: Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht. • "Current" = TRUE: Die Achse verfährt mit der zum Startzeitpunkt aktuellen Geschwindigkeit.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

PLCopen Version 2.0

Die Motion Control-Anweisung "MC_MoveVelocity" ist ab V4 konform mit der PLCopen Version 2.0.

Die Parameter "InVelocity" und "Busy" zeigen ihren Status unabhängig vom Parameter "Execute" an, bis der Auftrag abgelöst oder mit Fehler abgebrochen wird. Beachten Sie dazu auch das Kapitel Verfolgung laufender Aufträge.

Verhalten bei Sollgeschwindigkeit Null (Velocity = 0.0)

Ein MC_MoveVelocity-Auftrag mit "Velocity" = 0.0 bricht (wie ein MC_Halt-Auftrag) aktive Bewegungsaufträge ab und stoppt die Achse mit der konfigurierten Verzögerung.

Mit dem Erreichen des Stillstands wird am Ausgangsparameter "InVelocity" für mindestens einen Programmzyklus TRUE angezeigt.

"Busy" zeigt während des Verzögerungsvorgangs den Wert TRUE und wechselt diesen zusammen mit "InVelocity" zu FALSE. Ist der Parameter "Execute" = TRUE gesetzt, so wird "InVelocity" und "Busy" speichernd angezeigt.

Mit dem Start des MC_MoveVelocity-Auftrags wird das Statusbit "SpeedCommand" im Technologieobjekt gesetzt. Das Statusbit "ConstantVelocity" mit dem Stillstand der Achse. Beide Bits werden mit dem Start eines neuen Verfahrauftrags an die neue Situation angepasst.

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_MoveVelocity: Funktionsdiagramm ab V6

Funktionsdiagramm

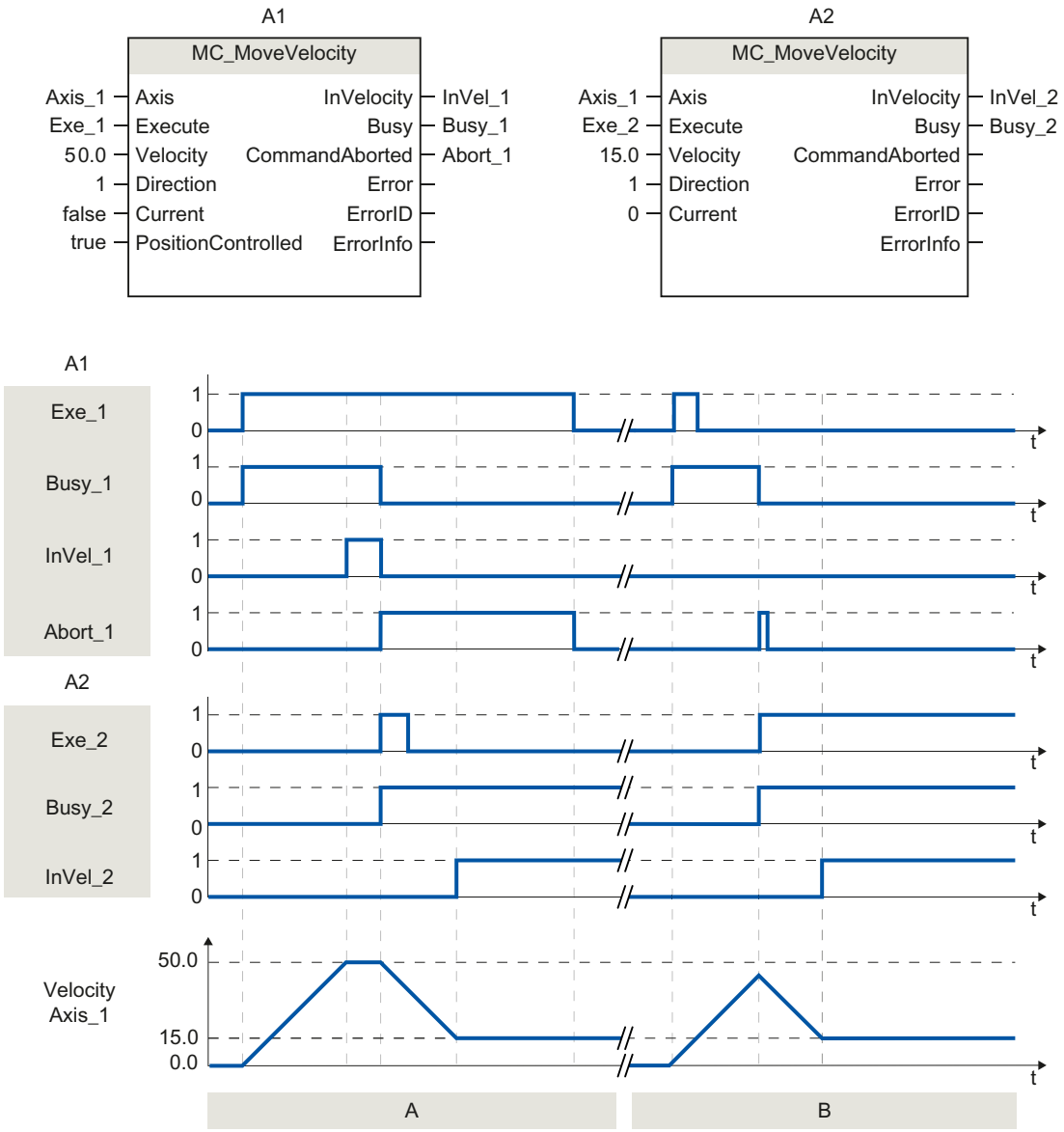

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

Abschnitt A	Ein laufender MC_MoveVelocity-Auftrag meldet über "InVel_1" das Erreichen seiner Zielgeschwindigkeit. Anschließend wird er durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.
Abschnitt B	Ein laufender MC_MoveVelocity-Auftrag wird vor dem Erreichen seiner Zielgeschwindigkeit durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.

MC_MoveJog

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6

Beschreibung

Die Motion Control-Anweisung "MC_MoveJog" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit im Tippbetrieb. Verwenden Sie diese Motion Control Anweisung z. B. für Test- und Inbetriebnahmezwecke.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveJog-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveJog-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag

- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.	
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.	
Sind beide Parameter gleichzeitig TRUE, so stoppt die Achse mit der konfigurierten Verzögerung. An den Parametern "Error", "ErrorID" und "ErrorInfo" wird ein Fehler angezeigt.					
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für den Tippbetrieb. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der MC_MoveJog-Auftrag ausgeführt wird. Danach gilt wieder die Einstellung des MC_Power. Beim Einsatz einer PTO-Achse wird der Parameter ignoriert.	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_MoveJog: Funktionsdiagramm ab V6

Funktionsdiagramm

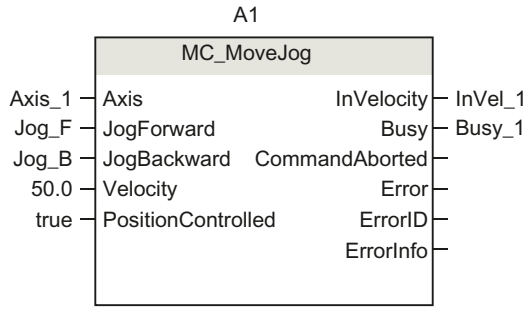

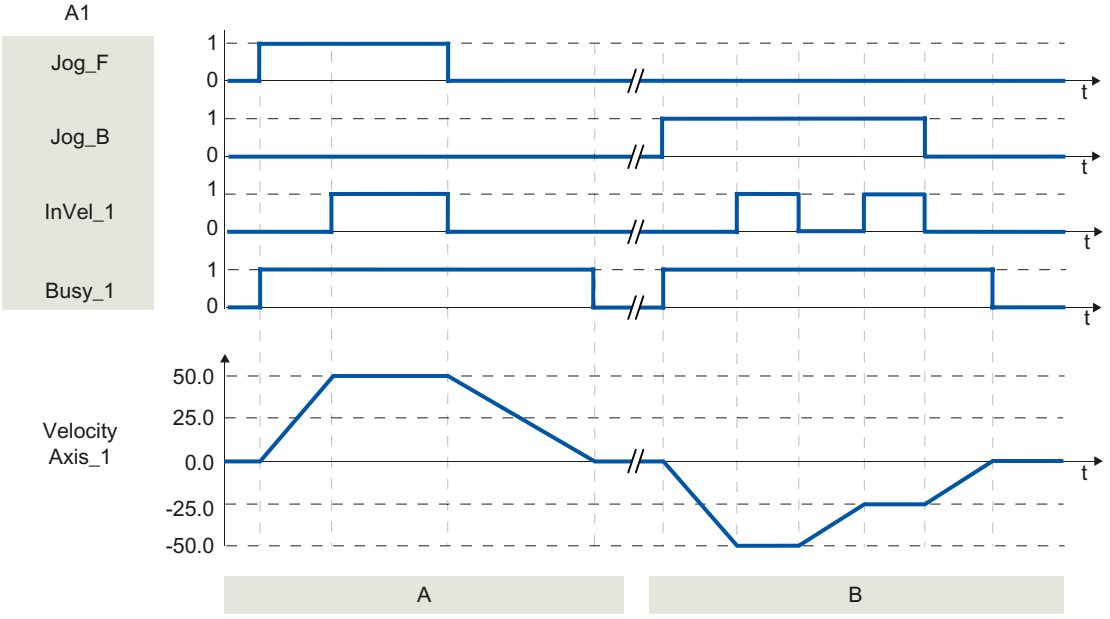

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

Abschnitt A	Über "Jog_F" wird die Achse im Tippbetrieb in positiver Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Nach dem Rücksetzen von "Jog_F" wird die Achse bis zum Stillstand abgebremst.
Abschnitt B	Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Zielgeschwindigkeit -50.0 wird über "InVel_1" gemeldet. Bei gesetztem "Jog_B" wird der Wert am Parameter "Velocity" auf 25.0 geändert. "InVel_1" wird rückgesetzt und die Achse wird abgebremst. Das Erreichen der neuen Zielgeschwindigkeit -25.0 wird über "InVel_1" gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse bis zum Stillstand abgebremst.

MC_CommandTable

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6

Beschreibung

Die Motion Control-Anweisung "MC_CommandTable" fasst mehrere Einzelaufträge zur Steuerung einer Achse zu einer Bewegungssequenz zusammen. "MC_CommandTable" ist für Achsen mit Antriebsanbindung über PTO (Pulse Train Output) verfügbar.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde eingefügt und korrekt konfiguriert.
- Der Antrieb ist über PTO (Pulse Train Output) angebunden.
- Das Technologieobjekt Auftragstabelle wurde eingefügt und korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_CommandTable-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_CommandTable-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag

- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der laufende Motion Control-Auftrag wird mit dem Start des ersten "Positioning Relative", "Positioning Absolute", "Velocity set point" oder "Halt" Auftrag abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Command-Table	INPUT	TO_Command-Table	-	Technologieobjekt der Auftragstabelle	
Execute	INPUT	BOOL	FALSE	Start der Auftragstabelle mit steigender Flanke	
StartStep	INPUT	INT	1	Vorgabe ab welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: 1 ≤ StartStep ≤ EndStep	
EndStep	INPUT	INT	32	Vorgabe bis zu welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: StartStep ≤ EndStep ≤ 32	
Done	OUTPUT	BOOL	FALSE	TRUE	Auftragstabelle wurde erfolgreich abgearbeitet
Busy	OUTPU	BOOL	FALSE	TRUE	Auftragstabelle ist in Bearbeitung
CommandA-borted	OUTPUT	BOOL	FALSE	TRUE	Die Auftragstabelle wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung der Auftragestabelle ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	
CurrentStep	OUTPUT	INT	0	Aktuell in Bearbeitung befindlicher Schritt der Auftragstabelle	
StepCode	OUTPUT	WORD	16#0000	Benutzerdefinierte Zahlenwert / Bitmuster des aktuell in Bearbeitung befindlichen Schritts	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_ChangeDynamic

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6

Beschreibung

Die Motion Control-Anweisung "MC_ChangeDynamic" ermöglicht die Änderung folgender Einstellungen der Achse:

- Wert für Hochlaufzeit (Beschleunigung) ändern
- Wert für Rücklaufzeit (Verzögerung) ändern
- Wert für Notstopp-Rücklaufzeit (Notstopp-Verzögerung) ändern
- Wert für Verrundungszeit (Ruck) ändern

Die Wirksamkeit der Änderung entnehmen Sie der Beschreibung der Variablen.

Voraussetzungen

Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.

Ablöseverhalten

Ein MC_ChangeDynamic-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_ChangeDynamic-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Change-RampUp	INPUT	BOOL	FALSE	TRUE	Hochlaufzeit entsprechend Eingangsparameter "RampUpTime" ändern

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
RampUp-Time	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung vom Stillstand auf die konfigurierte maximale Geschwindigkeit zu beschleunigen Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Acceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-RampDown	INPUT	BOOL	FALSE	TRUE	Rücklaufzeit entsprechend Eingangsparameter "RampDownTime" geändert
Ramp-DownTime	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Deceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
ChangeE-mergency	INPUT	BOOL	FALSE	TRUE	Notstopp-Rücklaufzeit entsprechend Eingangsparameter "EmergencyRampTime" ändern
Emergen-cyRamp-Time	INPUT	REAL	2.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung im Notstopp-Betrieb von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.EmergencyDeceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-JerkTime	INPUT	BOOL	FALSE	TRUE	Verrundungszeit entsprechend Eingangsparameter "JerkTime" ändern
JerkTime	INPUT	REAL	0.25	Verrundungszeit (in Sekunden), die auf die Beschleunigungs- und Verzögerungsrampe der Achse angewendet wird Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Jerk beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Done	OUTPUT	BOOL	FALSE	TRUE	Die geänderten Werte wurden in den Technologie-DB geschrieben. Wann die Änderung wirksam wird, können Sie der Beschreibung der Variablen entnehmen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

An den Eingangsparametern "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" können Werte angegeben werden, welche die zulässigen Grenzwerte der resultierenden Parameter: "Beschleunigung", "Verzögerung", "Notstopp-Verzögerung" und "Ruck" überschreiten.

Bitte beachten Sie unter Berücksichtigung der Gleichungen und Grenzwerte im Kapitel "Dynamik", dass Ihre Eingaben innerhalb des gültigen Bereichs liegen.

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_ReadParam

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6

Beschreibung

Die Motion Control-Anweisung "MC_ReadParam" ermöglicht das kontinuierliche Lesen von Bewegungsdaten und Statusmeldungen einer Achse. Der aktuelle Wert der entsprechenden Variablen wird beim Start des Auftrags ermittelt.

Folgende Bewegungsdaten und Statusmeldungen können gelesen werden:

- Ab Technologieversion V4:
  - Sollposition der Achse
  - Soll- und Istgeschwindigkeit der Achse
  - Akuteller Abstand der Achse zur Zielposition
  - Zielposition der Achse
- Zusätzlich ab Technologieversion V5:
  - Istposition der Achse
  - Istgeschwindigkeit der Achse
  - Aktueller Schleppfehler
  - Status des Antriebs
  - Status des Gebers
  - Statusbits
  - Errorbits

Voraussetzungen

Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.

Ablöseverhalten

Ein MC_ReadParam-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_ReadParam-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Enable	INPUT	BOOL	FALSE	TRUE	Die über "Parameter" festgelegte Variable lesen und den Wert in der über "Value" festgelegten Zieladresse speichern.
				FALSE	Parametrierte Bewegungsdaten nicht aktualisieren

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Parameter	INPUT	VARIANT (REAL)	-	VARIANT-Zeiger auf den zu lesenden Wert. Folgende Variablen sind zulässig: • <Achsname>.Position • <Achsname>.Velocity • <Achsname>.ActualPosition • <Achsname>.ActualVelocity • <Achsname>.StatusPositioning.<Variablenname> • <Achsname>.StatusDrive.<Variablenname> • <Achsname>.StatusSensor.<Variablenname> • <Achsname>.StatusBits.<Variablenname> • <Achsname>.ErrorBits.<Variablenname> Die Beschreibung der genannten Variablen und Variablenstrukturen finden Sie im Anhang AUTOHOTSPOT.	
Value	INOUT	VARIANT (REAL)	-	VARIANT-Zeiger auf die Ziel-Variable oder Ziel-Adresse in die der gelesene Wert geschrieben werden soll.	
Valid	OUTPUT	BOOL	FALSE	TRUE	Der gelesene Wert ist gültig.
				FALSE	Der gelesene Wert ist ungültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_WriteParam: Variable der Positionierachse schreiben ab V6 (Seite 3171)

MC_WriteParam

MC_WriteParam: Variable der Positionierachse schreiben ab V6

Beschreibung

Die Motion Control-Anweisung "MC_WriteParam" ermöglicht das Schreiben von Variablen des Technologieobjekts Positionierachse im Anwenderprogramm. Im Gegensatz zur Wertzuweisung der Variablen im Anwenderprogramm kann "MC_WriteParam" auch Werte von nur lesbaren Variablen ändern.

Welche Variablen Sie unter welchen Bedingungen schreiben können sowie den Zeitpunkt, zu dem eine Änderung wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Um Variablen zu schreiben, die im Anwenderprogramm nur lesbar sind, muss die Achse gesperrt sein.
- Variablen, deren Änderung einen Restart erfordern, sind mit "MC_WriteParam" nicht schreibbar.

Ablöseverhalten

Ein MC_WriteParam-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_WriteParam-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Parameter	INPUT	VARIANT (BOOL, INT, DINT, UDINT, REAL)	-	VARIANT-Zeiger auf die zu beschreibende Variablen des Technologieobjekts Positionierachse (Zieladresse)	
Value	INPUT	VARIANT (BOOL, INT, DINT, UDINT, REAL)	-	VARIANT-Zeiger auf den zu schreibenden Wert (Quelladresse)	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Wert wurde geschrieben
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"

Siehe auch

MC_Power: Achse freigeben, sperren ab V6 (Seite 3135)

MC_Reset: Fehler quittieren, Restart Technologieobjekt ab V6 (Seite 3140)

MC_Home: Achse referenzieren, Referenzpunkt setzen ab V6 (Seite 3141)

MC_Halt: Achse anhalten ab V6 (Seite 3146)

MC_MoveAbsolute: Achse absolut positionieren ab V6 (Seite 3149)

MC_MoveRelative: Achse relativ positionieren ab V6 (Seite 3153)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen ab V6 (Seite 3156)

MC_MoveJog: Achse im Tippbetrieb bewegen ab V6 (Seite 3161)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen ab V6 (Seite 3164)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern ab V6 (Seite 3166)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen ab V6 (Seite 3168)

S7-1200 Motion Control V4...5

MC_Power

MC_Power: Achse freigeben, sperren V4...5

Beschreibung

Die Motion Control-Anweisung "MC_Power" gibt eine Achse frei, bzw. sperrt diese.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Es steht kein freigabeverhindernder Fehler an.

Ablöseverhalten

Die Bearbeitung des "MC_Power" kann durch keinen Motion Control-Auftrag abgebrochen werden.

Mit dem Sperren der Achse (Eingangsparameter "Enable" = FALSE) werden alle Motion Control-Aufträge entsprechend dem gewählten "StopMode" am zugehörigen Technologieobjekt abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse	
Enable	INPUT	BOOL	FALSE	TRUE	Die Achse wird freigegeben.
				FALSE	Alle laufenden Aufträge werden entsprechend dem parametrierten "StopMode" abgebrochen. Die Achse wird gestoppt und gesperrt.
StopMode	INPUT	INT	0	0	Notstopp Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Beim Erreichen des Stillstands wird die Achse gesperrt.
				1	Sofort-Aus Steht eine Anforderung zum Sperren der Achse an, so wird der Sollwert Null ausgegeben und die Achse gesperrt. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Bei Antriebsanbindung über PTO (Pulse Train Output): Beim Sperren der Achse wird die Impulsausgabe betriebsbedingt mit einer frequenzabhängigen Verzögerung gestoppt: • Ausgangsfrequenz ≥ 100 Hz Verzögerung: max. 30 ms • Ausgangsfrequenz < 100 Hz Verzögerung: 30 ms bis max. 1,5 s bei 2 Hz
				2	Notstopp mit Ruckbegrenzung Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Ist die Ruckbegrenzung aktiviert, so wird der konfigurierte Ruck berücksichtigt. Beim Erreichen des Stillstands wird die Achse gesperrt.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Status	OUTPUT	BOOL	FALSE	Status der Achsfreigabe	
				FALSE	Die Achse ist gesperrt Die Achse führt keine Motion Control-Aufträge aus und nimmt keine neuen Aufträge an (Ausnahme: MC_Reset-Auftrag). Bei Antriebsanbindung über PTO (Pulse Train Output): Die Achse ist nicht referenziert. Beim Sperren wechselt der Status erst mit dem Erreichen des Stillstands der Achse auf FALSE.
				TRUE	Die Achse ist freigegeben Die Achse ist bereit, Motion Control-Aufträge auszuführen. Beim Freigeben der Achse wechselt der Status erst mit dem Anstehen des Signals "Antrieb bereit" auf TRUE. Wurde das Antriebssignal "Antrieb bereit" in der Achskonfiguration nicht konfiguriert, so wechselt der Status sofort nach TRUE.
Busy	OUTPUT	BOOL	FALSE	TRUE	"MC_Power" ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" oder am zugehörigen Technologieobjekt ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

Wird die Achse durch einen Fehler ausgeschaltet, so wird sie nach dem Beseitigen und Quittieren des Fehlers automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Eingangsparameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Freigeben einer Achse mit konfigurierten Antriebssignalen

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE.
   Der Freigabe-Ausgang für die "Antriebsfreigabe" wechselt auf TRUE, um dem Antrieb die Leistungsfreigabe zu erteilen. Die CPU wartet auf das Signal "Antrieb bereit" des Antriebs. Mit dem Anstehen des Signals "Antrieb bereit" am konfigurierten Bereit-Eingang der CPU wird die Achse freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Freigeben einer Achse ohne konfigurierte Antriebssignale

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE. Die Achse wird freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Sperren einer Achse

Zum Sperren einer Achse können Sie wie nachfolgend beschrieben vorgehen:

1. Bringen Sie die Achse in den Stillstand.
   Wann sich die Achse im Stillstand befindet, erkennen Sie an der Variable des Technologieobjekts <Achsname>.StatusBits.StandStill.
2. Setzen Sie den Eingangsparameters "Enable" nach dem Erreichen des Stillstandes auf FALSE.
3. Zeigen die Ausgangsparameter "Busy" und "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable den Wert FALSE, so ist das Sperren der Achse abgeschlossen.

Siehe auch

MC_Power: Funktionsdiagramm V4...5 (Seite 3176)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_Power: Funktionsdiagramm V4...5

Funktionsdiagramm

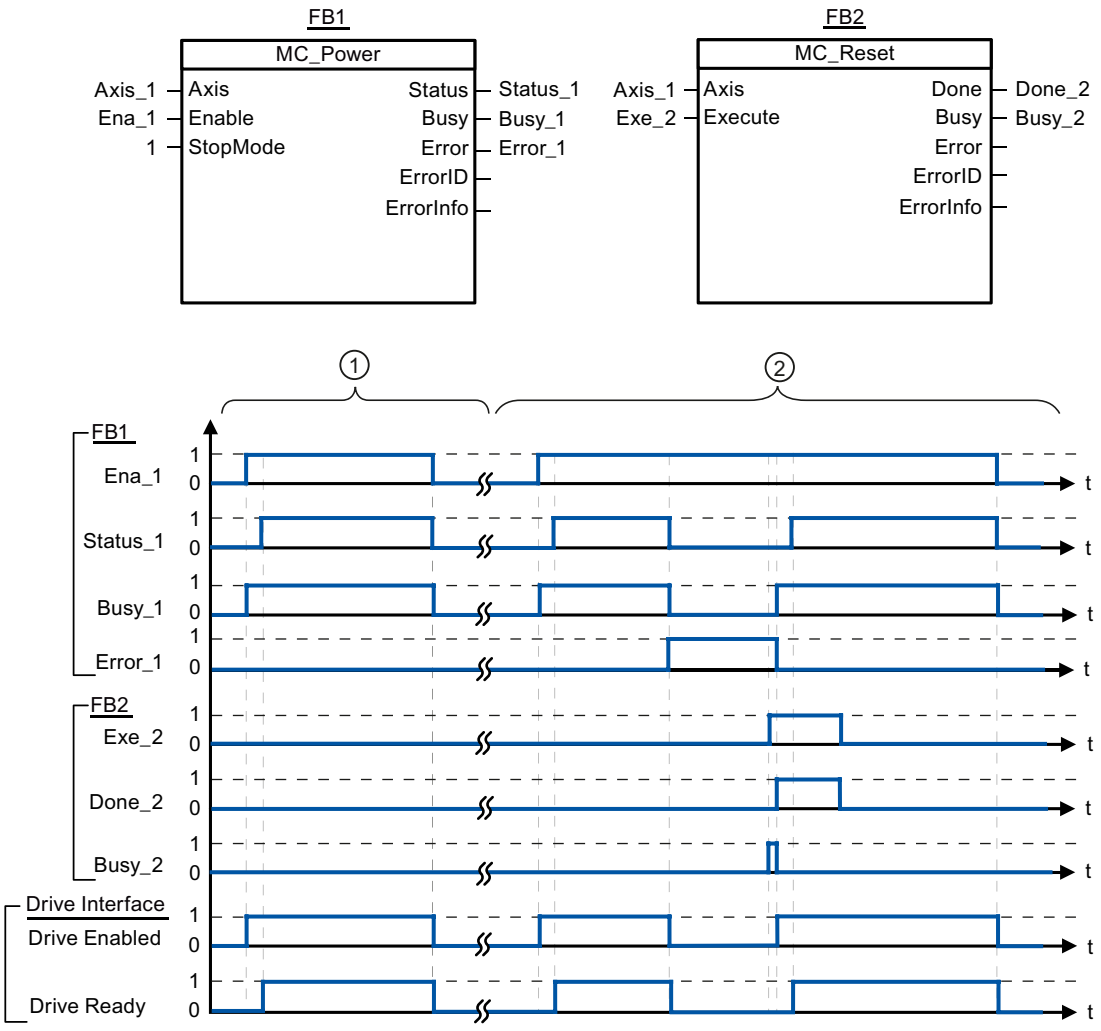

①	Eine Achse wird freigegeben und anschließend wieder gesperrt. Nachdem der Antrieb das Signal "Antrieb bereit" an die CPU zurück gemeldet hat, kann die erfolgreiche Freigabe über "Status_1" abgelesen werden.
②	Nach dem Freigeben einer Achse tritt ein Fehler auf, der das Sperren der Achse zur Folge hat. Der Fehler wird beseitigt und mit "MC_Reset" quittiert. Daraufhin wird die Achse wieder freigegeben.

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" können "Betriebsfehler mit Stopp der Achse" und "Konfigurationsfehler" quittiert werden. Welche Fehler quittiert werden müssen können Sie der "Liste der ErrorIDs und ErrorInfos" unter "Abhilfe" entnehmen.

Nach einem Laden im Betriebszustand RUN kann die Konfiguration der Achse in den Arbeitsspeicher geladen werden.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Bei einem anstehenden quittierbaren Konfigurationsfehler wurde die Fehlerursache beseitigt (z. B. die Beschleunigung im Technologieobjekt Positionierachse auf einen gültigen Wert geändert).

Ablöseverhalten

Der MC_Reset-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der neue MC_Reset-Auftrag bricht keine anderen laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Restart	INPUT	BOOL	FALSE	TRUE	Lädt die Konfiguration der Achse aus dem Ladespeicher in den Arbeitsspeicher. Der Auftrag ist nur bei gesperrter Achse ausführbar. Beachten Sie hierzu die Hinweise zum Laden in die CPU.
				FALSE	Quittiert anstehende Fehler
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Einen mit MC_Reset zu quittierenden Fehler quittieren

Gehen Sie zum Quittieren eines Fehlers folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" den Wert TRUE sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Error den Wert FALSE, so wurde der Fehler quittiert.

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_Home

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5

Beschreibung

Mit Hilfe der Motion Control-Anweisung "MC_Home" gleichen Sie die Achskoordinate auf die reale, physikalische Position des Antriebs ab. Soll die Achse absolut positioniert werden, so ist eine Referenzierung nötig. Folgende Referenzierarten können durchgeführt werden:

- Aktives Referenzieren (Mode = 3)
  Die Referenzpunktfahrt wird automatisch durchgeführt.
- Passives Referenzieren (Mode = 2)
  Beim passiven Referenzieren führt die Motion Control-Anweisung "MC_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse referenziert.

- Direktes Referenzieren Absolut (Mode = 0)
  Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.
- Direktes Referenzieren Relativ (Mode = 1)
  Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.
- Absolutwertgeberjustage Relativ (Mode = 6)
  Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.
- Absolutwertgeberjustage Absolut (Mode = 7)
  Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Beim Start mit Mode = 0, 1 und 2 darf kein MC_CommandTable-Auftrag aktiv sein.

Ablöseverhalten

Das Ablöseverhalten ist vom gewählten Mode abhängig:

Mode = 0, 1, 6, 7

Der MC_Home-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der MC_Home-Auftrag bricht keine laufenden Motion Control-Aufträge ab. Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 2

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 2, 3

Der neue MC_Home-Auftrag bricht folgenden laufenden Motion Control-Auftrag ab:

- MC_Home-Auftrag Mode = 2

Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 3

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Home-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 2, 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_Axis	-	Technologieobjekt der Achse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	REAL	0.0	• Mode = 0, 2 und 3 Absolute Position der Achse nach dem Abschluss des Referenziervorgangs • Mode = 1 Korrekturwert zur aktuellen Achsposition Grenzwerte: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	INT	0	Referenziermodus	
				0	Direktes Referenzieren (Absolut) Neue Achsposition ist der Positionswert des Parameters "Position".
				1	Direktes Referenzieren (Relativ) Neue Achsposition ist die aktuelle Achsposition + Positionswert des Parameters "Position".
				2	Passives Referenzieren Referenzieren gemäß der Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
				3	Aktives Referenzieren Referenzpunktfahrt gemäß Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<Achsname>.StatusSensor.AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<Achsname>.StatusSensor.AbsEncoderOffset)
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjekts (<Achsname>.StatusBits.HomingDone) wird unter folgenden Bedingungen rückgesetzt:

- **Antriebsanbindung über PTO (Pulse Train Output):**
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Sperren der Achse durch die Motion Control-Anweisung "MC_Power"
  - Wechsel zwischen Automatikbetrieb und Handsteuerung
  - Nach NETZ-AUS -> NETZ-EIN der CPU
  - Nach Neustart der CPU (RUN-STOP -> STOP-RUN)
- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS → NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Eine Achse referenzieren

Gehen Sie beim Referenzieren der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die notwendigen Eingangsparameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" sowie die Variable des Technologieobjekts <Achsname>.StatusBits.HomingDone den Wert TRUE, so ist das Referenzieren abgeschlossen.

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_Halt

MC_Halt: Achse anhalten V4...5

Beschreibung

Die Motion Control-Anweisung "MC_Halt" bricht alle Bewegungsvorgänge ab und bremst die Achse mit der konfigurierten Verzögerung bis zum Stillstand. Die Stillstandsposition ist nicht definiert.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_Halt-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag

- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Halt-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit Null erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennnung zum Parameter "ErrorID"	

Siehe auch

MC_Halt: Funktionsdiagramm V4...5 (Seite 3185)

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_Halt: Funktionsdiagramm V4...5

Funktionsdiagramm

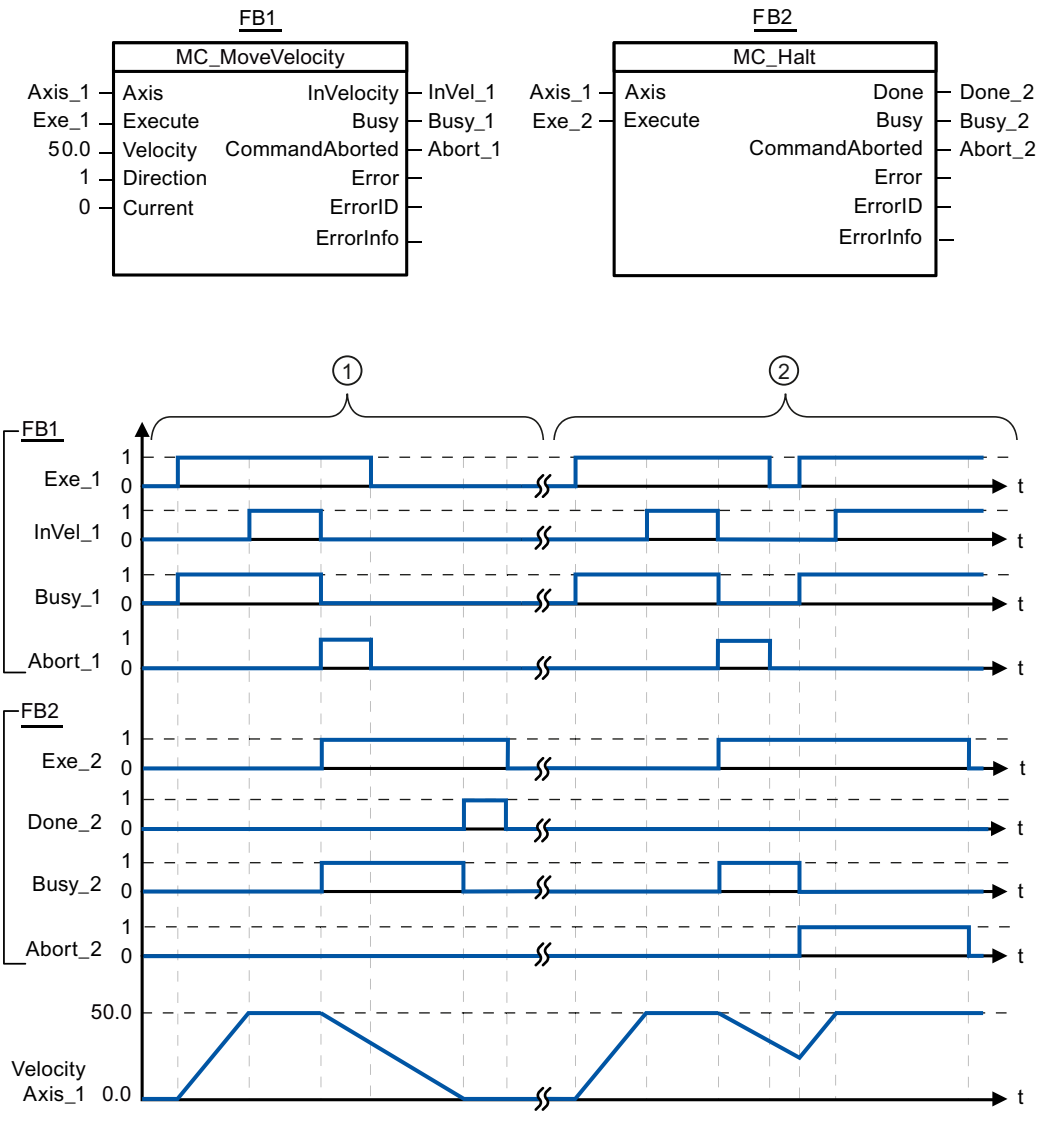

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

①	Die Achse wird durch einen MC_Halt-Auftrag bis zum Stillstand abgebremst. Das Erreichen des Stillstandes wird über "Done_2" gemeldet.
②	Während ein MC_Halt-Auftrag die Achse abbremst wird dieser Auftrag durch einen anderen Verfahrauftrag abgebrochen. Der Abbruch wird über "Abort_2" gemeldet.

Siehe auch

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute

MC_MoveAbsolute: Achse absolut positionieren V4...5

Beschreibung

Die Motion Control-Anweisung "MC_MoveAbsolute" startet eine Positionierbewegung der Achse zu einer absoluten Position.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Die Achse ist referenziert.

Ablöseverhalten

Der MC_MoveAbsolute-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveAbsolute-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag

- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Position	INPUT	REAL	0.0	Absolute Zielposition Grenzwerte: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der anzufahrenden Zielposition nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Done	OUTPUT	BOOL	FALSE	TRUE	Absolute Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_MoveAbsolute: Funktionsdiagramm V4...5 (Seite 3188)

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_MoveAbsolute: Funktionsdiagramm V4...5

Funktionsdiagramm

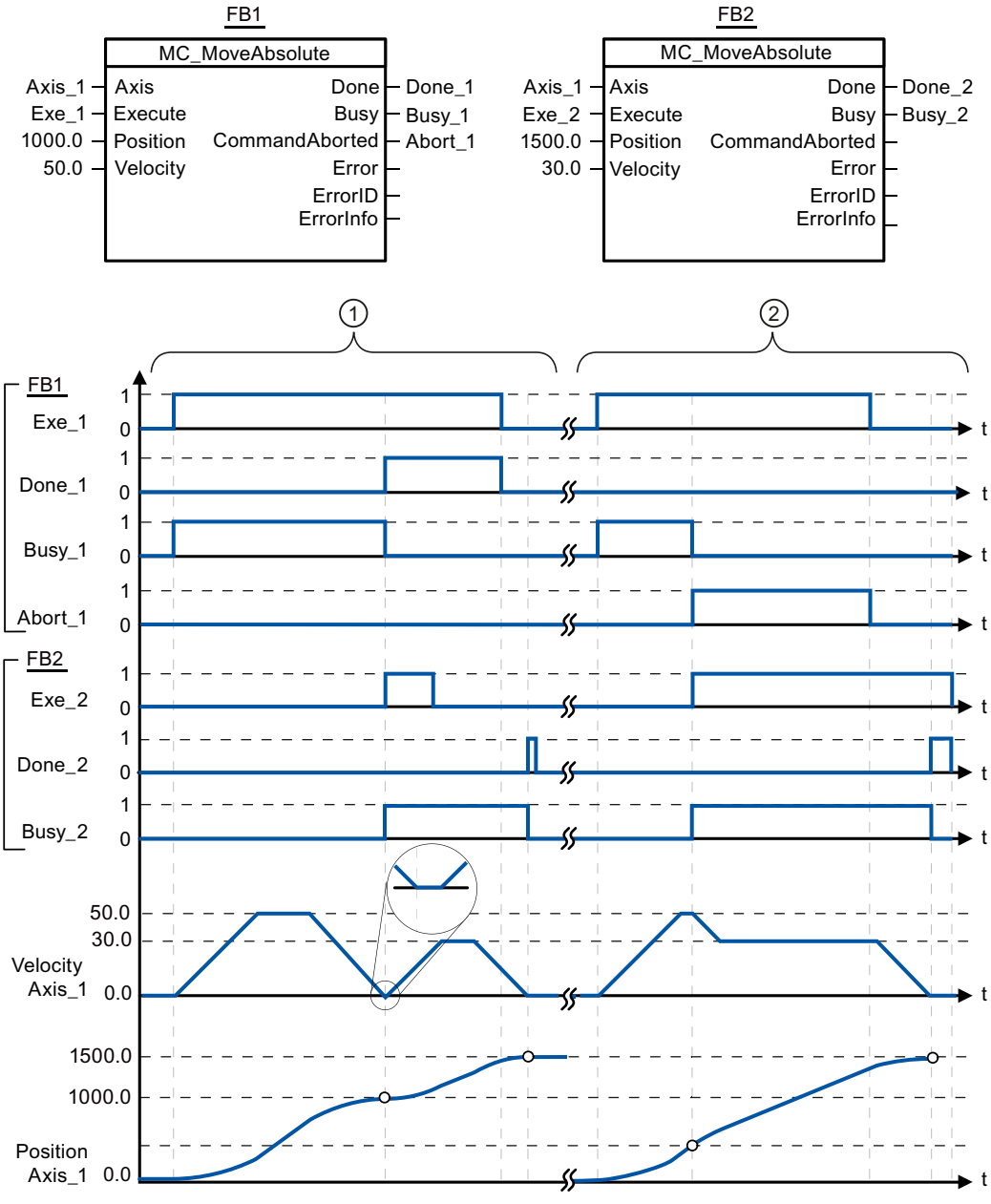

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Eine Achse wird durch einen MC_MoveAbsolute-Auftrag auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveAbsolute-Auftrag, mit Zielposition 1500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
②	Ein laufender MC_MoveAbsolute-Auftrag wird durch einen weiteren MC_MoveAbsolute-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

Siehe auch

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative

MC_MoveRelative: Achse relativ positionieren V4...5

Beschreibung

Die Motion Control-Anweisung "MC_MoveRelative" startet eine Positionierbewegung relativ zur Startposition.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveRelative-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveRelative-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Distance	INPUT	REAL	0.0	Wegstrecke für den Positioniervorgang Grenzwerte: $-1.0e^{12} \leq$ Distance $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der zu fahrenden Wegstrecke nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_MoveRelative: Funktionsdiagramm V4...5 (Seite 3192)

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_MoveRelative: Funktionsdiagramm V4...5

Funktionsdiagramm

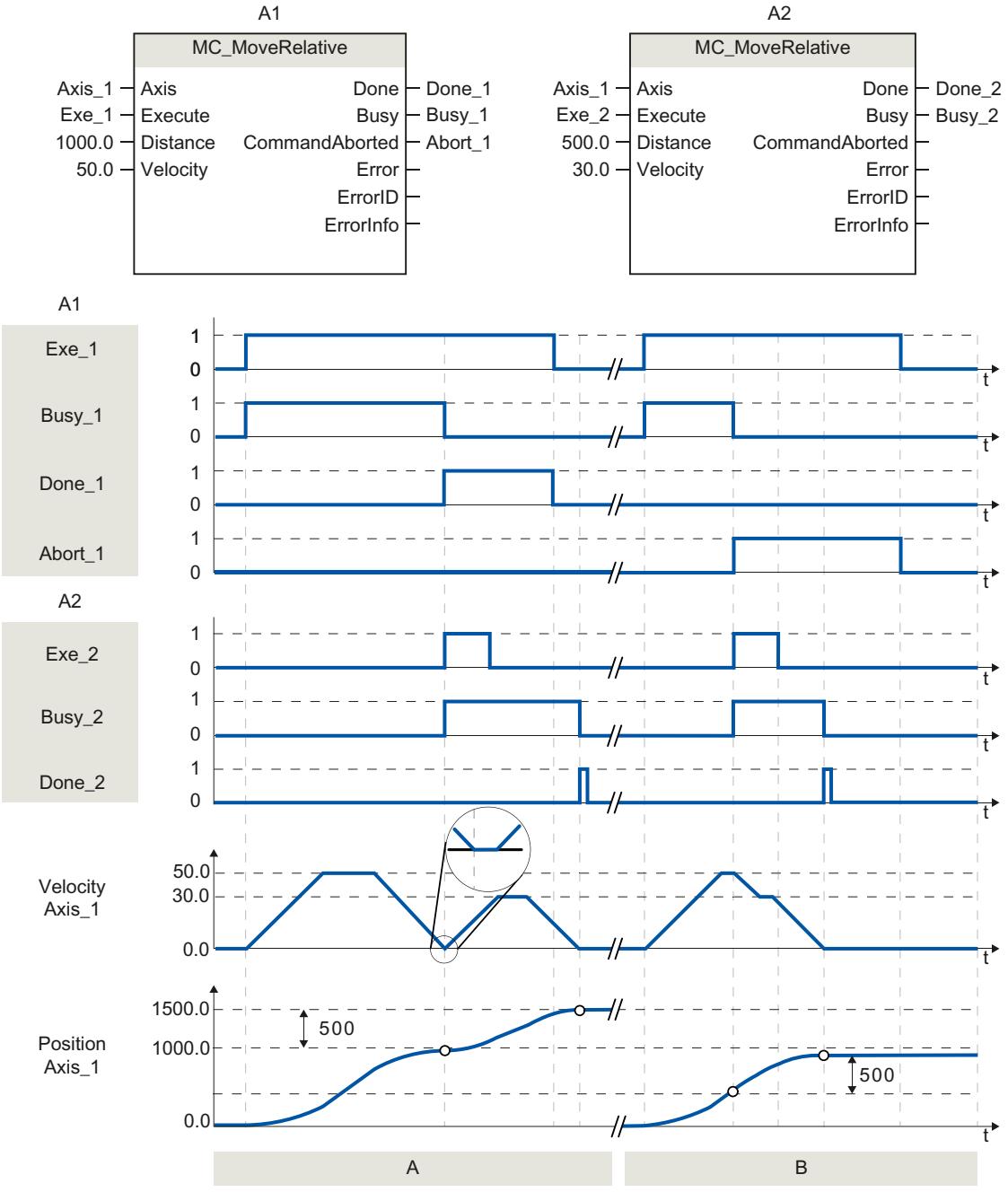

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Die Achse wird durch einen MC_MoveRelative-Auftrag um die Wegstrecke ("Distance") 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveRelative-Auftrag, mit der Wegstrecke 500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
②	Ein laufender MC_MoveRelative-Auftrag wird durch einen weiteren MC_MoveRelative-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die neue Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

Siehe auch

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5

Beschreibung

Die Motion Control-Anweisung "MC_MoveVelocity" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveVelocity kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveVelocity-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag

- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für das Verfahren der Achse Grenzwerte: Start/Stopp-Geschwindigkeit ≤ |Velocity| ≤ Maximale Geschwindigkeit (Velocity = 0.0 ist erlaubt)	
Direction	INPUT	INT	0	Richtungsvorgabe	
				0	Drehrichtung entsprechend dem Vorzeichen des Werts am Parameter "Velocity"
				1	Drehrichtung positiv (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
				2	Drehrichtung negativ (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	"Aktuelle Geschwindigkeit beibehalten" deaktiviert. Es werden die Werte der Parameter "Velocity" und "Direction" verwendet.
				TRUE	"Aktuelle Geschwindigkeit beibehalten" aktiviert. Die Werte an den Parametern "Velocity" und "Direction" bleiben unberücksichtigt. Sobald die Achse mit der aktuellen Geschwindigkeit weiterfährt liefert der Parameter "InVelocity" den Wert TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	• "Current" = FALSE: Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht. • "Current" = TRUE: Die Achse verfährt mit der zum Startzeitpunkt aktuellen Geschwindigkeit.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

PLCopen Version 2.0

Die Motion Control-Anweisung "MC_MoveVelocity" ist ab V4 konform mit der PLCopen Version 2.0.

Die Parameter "InVelocity" und "Busy" zeigen ihren Status unabhängig vom Parameter "Execute" an, bis der Auftrag abgelöst oder mit Fehler abgebrochen wird. Beachten Sie dazu auch das Kapitel Verfolgung laufender Aufträge.

Verhalten bei Sollgeschwindigkeit Null (Velocity = 0.0)

Ein MC_MoveVelocity-Auftrag mit "Velocity" = 0.0 bricht (wie ein MC_Halt-Auftrag) aktive Bewegungsaufträge ab und stoppt die Achse mit der konfigurierten Verzögerung.

Mit dem Erreichen des Stillstands wird am Ausgangsparameter "InVelocity" für mindestens einen Programmzyklus TRUE angezeigt.

"Busy" zeigt während des Verzögerungsvorgangs den Wert TRUE und wechselt diesen zusammen mit "InVelocity" zu FALSE. Ist der Parameter "Execute" = TRUE gesetzt, so wird "InVelocity" und "Busy" speichernd angezeigt.

Mit dem Start des MC_MoveVelocity-Auftrags wird das Statusbit "SpeedCommand" im Technologieobjekt gesetzt. Das Statusbit "ConstantVelocity" mit dem Stillstand der Achse. Beide Bits werden mit dem Start eines neuen Verfahrauftrags an die neue Situation angepasst.

Siehe auch

MC_MoveVelocity: Funktionsdiagramm V4...5 (Seite 3196)

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_MoveVelocity: Funktionsdiagramm V4...5

Funktionsdiagramm

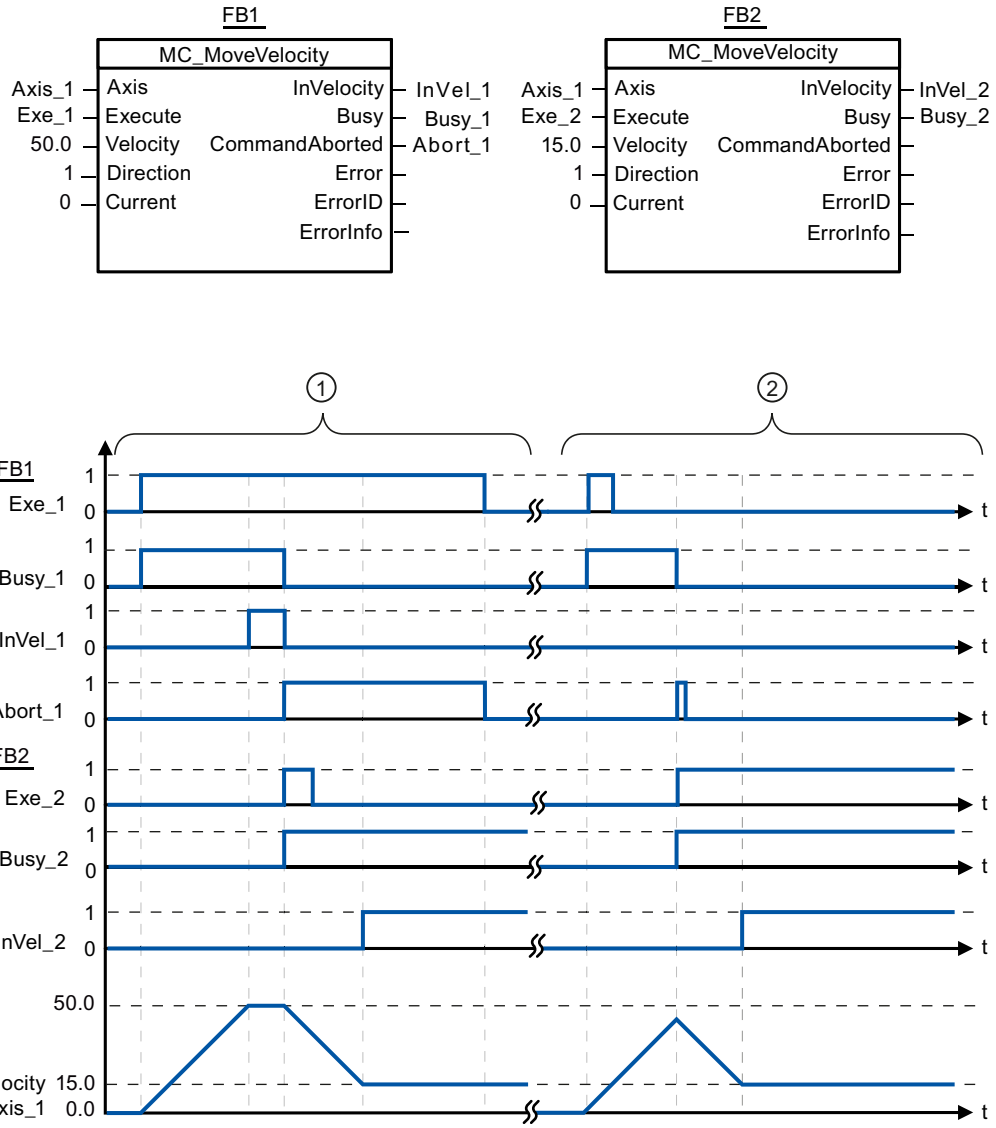

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Ein laufender MC_MoveVelocity-Auftrag meldet über "InVel_1" das Erreichen seiner Zielgeschwindigkeit. Anschließend wird er durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.
②	Ein laufender MC_MoveVelocity-Auftrag wird vor dem Erreichen seiner Zielgeschwindigkeit durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.

Siehe auch

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5

Beschreibung

Die Motion Control-Anweisung "MC_MoveJog" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit im Tippbetrieb. Verwenden Sie diese Motion Control Anweisung z. B. für Test- und Inbetriebnahmezwecke.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveJog-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveJog-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag

- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.	
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.	
Sind beide Parameter gleichzeitig TRUE, so stoppt die Achse mit der konfigurierten Verzögerung. An den Parametern "Error", "ErrorID" und "ErrorInfo" wird ein Fehler angezeigt.					
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für den Tippbetrieb. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_MoveJog: Funktionsdiagramm V4...5 (Seite 3199)

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_MoveJog: Funktionsdiagramm V4...5

Funktionsdiagramm

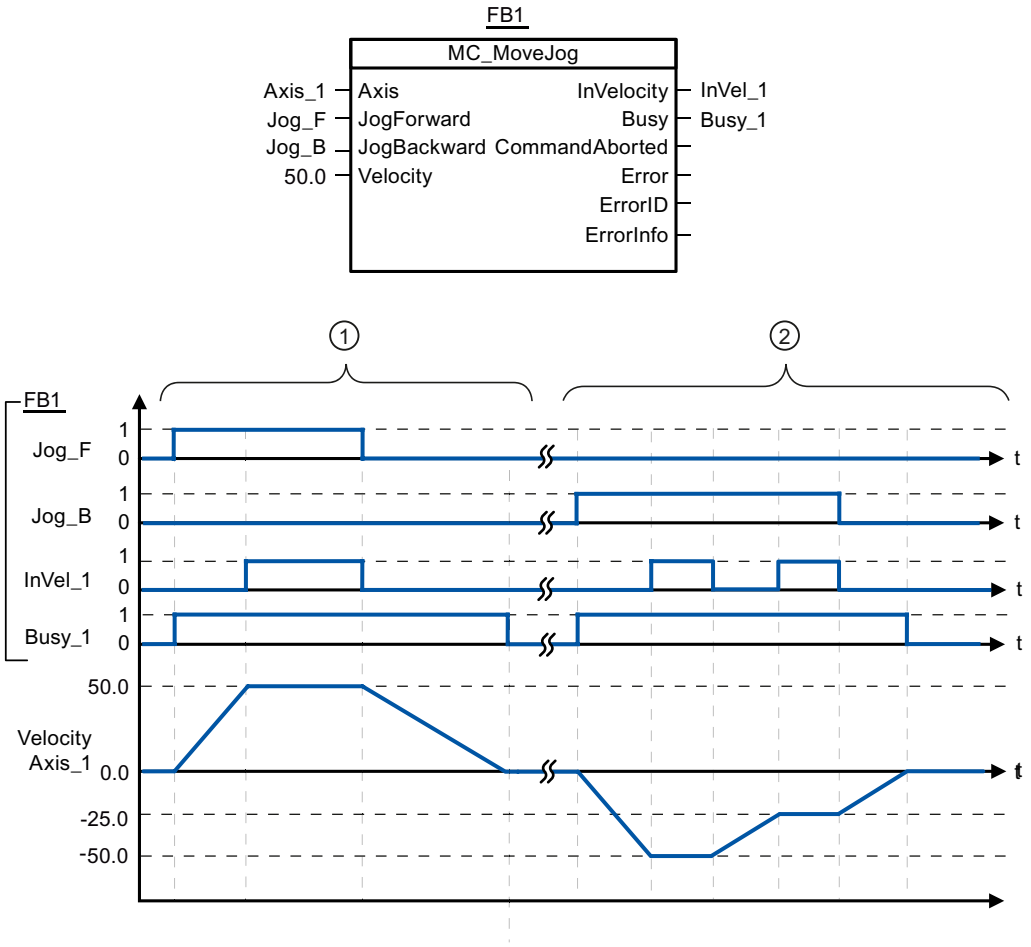

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

①	Über "Jog_F" wird die Achse im Tippbetrieb in positiver Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Nach dem Rücksetzen von "Jog_F" wird die Achse bis zum Stillstand abgebremst.
②	Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Zielgeschwindigkeit -50.0 wird über "InVel_1" gemeldet. Bei gesetztem "Jog_B" wird der Wert am Parameter "Velocity" auf 25.0 geändert. "InVel_1" wird rückgesetzt und die Achse wird abgebremst. Das Erreichen der neuen Zielgeschwindigkeit -25.0 wird über "InVel_1" gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse bis zum Stillstand abgebremst.

Siehe auch

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5

Beschreibung

Die Motion Control-Anweisung "MC_CommandTable" fasst mehrere Einzelaufträge zur Steuerung einer Achse zu einer Bewegungssequenz zusammen. "MC_CommandTable" ist für Achsen mit Antriebsanbindung über PTO (Pulse Train Output) verfügbar.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde eingefügt und korrekt konfiguriert.
- Der Antrieb ist über PTO (Pulse Train Output) angebunden.
- Das Technologieobjekt Auftragstabelle wurde eingefügt und korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_CommandTable-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_CommandTable-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der laufende Motion Control-Auftrag wird mit dem Start des ersten "Positioning Relative", "Positioning Absolute", "Velocity set point" oder "Halt" Auftrag abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Command-Table	INPUT	TO_Command-Table	-	Technologieobjekt der Auftragstabelle	
Execute	INPUT	BOOL	FALSE	Start der Auftragstabelle mit steigender Flanke	
StartStep	INPUT	INT	1	Vorgabe ab welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: 1 ≤ StartStep ≤ EndStep	
EndStep	INPUT	INT	32	Vorgabe bis zu welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: StartStep ≤ EndStep ≤ 32	
Done	OUTPUT	BOOL	FALSE	TRUE	Auftragstabelle wurde erfolgreich abgearbeitet
Busy	OUTPU	BOOL	FALSE	TRUE	Auftragstabelle ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Die Auftragstabelle wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung der Auftragestabelle ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	
CurrentStep	OUTPUT	INT	0	Aktuell in Bearbeitung befindlicher Schritt der Auftragstabelle	
StepCode	OUTPUT	WORD	16#0000	Benutzerdefinierte Zahlenwert / Bitmuster des aktuell in Bearbeitung befindlichen Schritts	

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_ChangeDynamic

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5

Beschreibung

Die Motion Control-Anweisung "MC_ChangeDynamic" ermöglicht die Änderung folgender Einstellungen der Achse:

- Wert für Hochlaufzeit (Beschleunigung) ändern
- Wert für Rücklaufzeit (Verzögerung) ändern
- Wert für Notstopp-Rücklaufzeit (Notstopp-Verzögerung) ändern
- Wert für Verrundungszeit (Ruck) ändern

Die Wirksamkeit der Änderung entnehmen Sie der Beschreibung der Variablen.

Voraussetzungen

Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.

Ablöseverhalten

Ein MC_ChangeDynamic-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_ChangeDynamic-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Change-RampUp	INPUT	BOOL	FALSE	TRUE	Hochlaufzeit entsprechend Eingangsparameter "RampUpTime" ändern
RampUp-Time	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung vom Stillstand auf die konfigurierte maximale Geschwindigkeit zu beschleunigen Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Acceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-RampDown	INPUT	BOOL	FALSE	TRUE	Rücklaufzeit entsprechend Eingangsparameter "RampDownTime" geändert
Ramp-DownTime	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Deceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
ChangeE-mergency	INPUT	BOOL	FALSE	TRUE	Notstopp-Rücklaufzeit entsprechend Eingangsparameter "EmergencyRampTime" ändern
Emergen-cyRamp-Time	INPUT	REAL	2.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung im Notstopp-Betrieb von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.EmergencyDeceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-JerkTime	INPUT	BOOL	FALSE	TRUE	Verrundungszeit entsprechend Eingangsparameter "JerkTime" ändern
JerkTime	INPUT	REAL	0.25	Verrundungszeit (in Sekunden), die auf die Beschleunigungs- und Verzögerungsrampe der Achse angewendet wird Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Jerk beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Done	OUTPUT	BOOL	FALSE	TRUE	Die geänderten Werte wurden in den Technologie-DB geschrieben. Wann die Änderung wirksam wird, können Sie der Beschreibung der Variablen entnehmen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

An den Eingangsparametern "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" können Werte angegeben werden, welche die zulässigen Grenzwerte der resultierenden Parameter: "Beschleunigung", "Verzögerung", "Notstopp-Verzögerung" und "Ruck" überschreiten.

Bitte beachten Sie unter Berücksichtigung der Gleichungen und Grenzwerte im Kapitel "Dynamik", dass Ihre Eingaben innerhalb des gültigen Bereichs liegen.

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_ReadParam

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5

Beschreibung

Die Motion Control-Anweisung "MC_ReadParam" ermöglicht das kontinuierliche Lesen von Bewegungsdaten und Statusmeldungen einer Achse. Der aktuelle Wert der entsprechenden Variablen wird beim Start des Auftrags ermittelt.

Folgende Bewegungsdaten und Statusmeldungen können gelesen werden:

- Ab Technologieversion V4:
  - Sollposition der Achse
  - Soll- und Istgeschwindigkeit der Achse
  - Akuteller Abstand der Achse zur Zielposition
  - Zielposition der Achse
- Zusätzlich ab Technologieversion V5:
  - Istposition der Achse
  - Istgeschwindigkeit der Achse
  - Aktueller Schleppfehler
  - Status des Antriebs
  - Status des Gebers
  - Statusbits
  - Errorbits

Voraussetzungen

Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.

Ablöseverhalten

Ein MC_ReadParam-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_ReadParam-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Enable	INPUT	BOOL	FALSE	TRUE	Die über "Parameter" festgelegte Variable lesen und den Wert in der über "Value" festgelegten Zieladresse speichern.
				FALSE	Parametrierte Bewegungsdaten nicht aktualisieren

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Parameter	INPUT	VARIANT (REAL)	-	VARIANT-Zeiger auf den zu lesenden Wert. Folgende Variablen sind zulässig: • <Achsname>.Position • <Achsname>.Velocity • <Achsname>.ActualPosition • <Achsname>.ActualVelocity • <Achsname>.StatusPositioning.<Variablenname> • <Achsname>.StatusDrive.<Variablenname> • <Achsname>.StatusSensor.<Variablenname> • <Achsname>.StatusBits.<Variablenname> • <Achsname>.ErrorBits.<Variablenname> Die Beschreibung der genannten Variablen und Variablenstrukturen finden Sie im Anhang AUTOHOTSPOT.	
Value	INOUT	VARIANT (REAL)	-	VARIANT-Zeiger auf die Ziel-Variable oder Ziel-Adresse in die der gelesene Wert geschrieben werden soll.	
Valid	OUTPUT	BOOL	FALSE	TRUE	Der gelesene Wert ist gültig.
				FALSE	Der gelesene Wert ist ungültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_WriteParam: Variable der Positionierachse schreiben V4...5 (Seite 3207)

S7-1200 Motion Control V1...3 (Seite 3208)

MC_WriteParam

MC_WriteParam: Variable der Positionierachse schreiben V4...5

Beschreibung

Die Motion Control-Anweisung "MC_WriteParam" ermöglicht das Schreiben von Variablen des Technologieobjekts Positionierachse im Anwenderprogramm. Im Gegensatz zur Wertzuweisung der Variablen im Anwenderprogramm kann "MC_WriteParam" auch Werte von nur lesbaren Variablen ändern.

Welche Variablen Sie unter welchen Bedingungen schreiben können sowie den Zeitpunkt, zu dem eine Änderung wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts.

Voraussetzungen

- Das Technologieobjekt Positionierachse wurde korrekt konfiguriert.
- Um Variablen zu schreiben, die im Anwenderprogramm nur lesbar sind, muss die Achse gesperrt sein.

Ablöseverhalten

Ein MC_WriteParam-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_WriteParam-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Parameter	INPUT	VARIANT (BOOL, INT, DINT, UDINT*, REAL) *) ab V5	-	VARIANT-Zeiger auf die zu beschreibende Variable des Technologieobjekts Positionierachse (Zieladresse)	
Value	INPUT	VARIANT (BOOL, INT, DINT, UDINT*, REAL) *) ab V5	-	VARIANT-Zeiger auf den zu schreibenden Wert (Quelladresse)	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Wert wurde geschrieben
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"

Siehe auch

MC_Power: Achse freigeben, sperren V4...5 (Seite 3172)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V4...5 (Seite 3177)

MC_Home: Achse referenzieren, Referenzpunkt setzen V4...5 (Seite 3178)

MC_Halt: Achse anhalten V4...5 (Seite 3183)

MC_MoveAbsolute: Achse absolut positionieren V4...5 (Seite 3186)

MC_MoveRelative: Achse relativ positionieren V4...5 (Seite 3189)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V4...5 (Seite 3193)

MC_MoveJog: Achse im Tippbetrieb bewegen V4...5 (Seite 3197)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V4...5 (Seite 3200)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V4...5 (Seite 3202)

MC_ReadParam: Bewegungsdaten einer Positionierachse kontinuierlich lesen V4...5 (Seite 3204)

S7-1200 Motion Control V1...3 (Seite 3208)

S7-1200 Motion Control V1...3

MC_Power

MC_Power: Achse freigeben, sperren V1...3

Beschreibung

Die Motion Control-Anweisung "MC_Power" gibt eine Achse frei, bzw. sperrt diese.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Es steht kein freigabeverhindernder Fehler an.

Ablöseverhalten

Die Bearbeitung des "MC_Power" kann durch keinen Motion Control-Auftrag abgebrochen werden.

Mit dem Sperren der Achse (Eingangsparameter "Enable" = FALSE) werden alle Motion Control-Aufträge entsprechend dem gewählten "StopMode" am zugehörigen Technologieobjekt abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Enable	INPUT	BOOL	FALSE	TRUE	Motion Control versucht die Achse freizugeben.
				FALSE	Alle laufenden Aufträge werden entsprechend dem parametrierten "StopMode" abgebrochen. Die Achse wird gestoppt und gesperrt.
StopMode	INPUT	INT	0	0	Notstopp Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Beim Erreichen des Stillstands wird die Achse gesperrt.
				1	Sofort-Aus Steht eine Anforderung zum Sperren der Achse an, so wird diese ohne Verzögerung gesperrt. Die Impulsausgabe wird sofort gestoppt.
				2	Notstopp mit Ruckbegrenzung Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Ist die Ruckbegrenzung aktiviert, so wird der konfigurierte Ruck berücksichtigt. Beim Erreichen des Stillstands wird die Achse gesperrt.
Status	OUTPUT	BOOL	FALSE	Status der Achsfreigabe	
				FALSE	Die Achse ist gesperrt Die Achse führt keine Motion Control-Aufträge aus und nimmt keine neuen Aufträge an (Ausnahme: MC_Reset-Auftrag). Die Achse ist nicht referenziert. Beim Sperren wechselt der Status erst mit dem Erreichen des Stillstandes der Achse auf FALSE.
				TRUE	Die Achse ist freigegeben Die Achse ist bereit, Motion Control-Aufträge auszuführen. Beim Freigeben der Achse wechselt der Status erst mit dem Anstehen des Signals "Antrieb bereit" auf TRUE. Wurde das Antriebssignal "Antrieb bereit" in der Achskonfiguration nicht konfiguriert, so wechselt der Status sofort nach TRUE.
Busy	OUTPUT	BOOL	FALSE	TRUE	"MC_Power" ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" oder am zugehörigen Technologieobjekt ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"

Hinweis

Wird die Achse durch einen Fehler ausgeschaltet, so wird sie nach dem Beseitigen und Quittieren des Fehlers automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Eingangsparameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Freigeben einer Achse mit konfigurierten Antriebssignalen

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE.
   Der Freigabe-Ausgang für die "Antriebsfreigabe" wechselt auf TRUE, um dem Antrieb die Leistungsfreigabe zu erteilen. Die CPU wartet auf das Signal "Antrieb bereit" des Antriebs. Mit dem Anstehen des Signals "Antrieb bereit" am konfigurierten Bereit-Eingang der CPU wird die Achse freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Freigeben einer Achse ohne konfigurierte Antriebssignale

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE. Die Achse wird freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable zeigen den Wert TRUE.

Sperren einer Achse

Zum Sperren einer Achse können Sie wie nachfolgend beschrieben vorgehen:

1. Bringen Sie die Achse in den Stillstand.
   Wann sich die Achse im Stillstand befindet, erkennen Sie an der Variable des Technologieobjekts <Achsname>.StatusBits.StandStill.
2. Setzen Sie den Eingangsparameters "Enable" nach dem Erreichen des Stillstandes auf FALSE.
3. Zeigen die Ausgangsparameter "Busy" und "Status", sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Enable den Wert FALSE, so ist das Sperren der Achse abgeschlossen.

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)
MC_Power: Funktionsdiagramm V1...3 (Seite 3212)
MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)
MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)
MC_Halt: Achse anhalten V1...3 (Seite 3218)
MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)
MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)
MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)
MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)
MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)
MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_Power: Funktionsdiagramm V1...3

Funktionsdiagramm

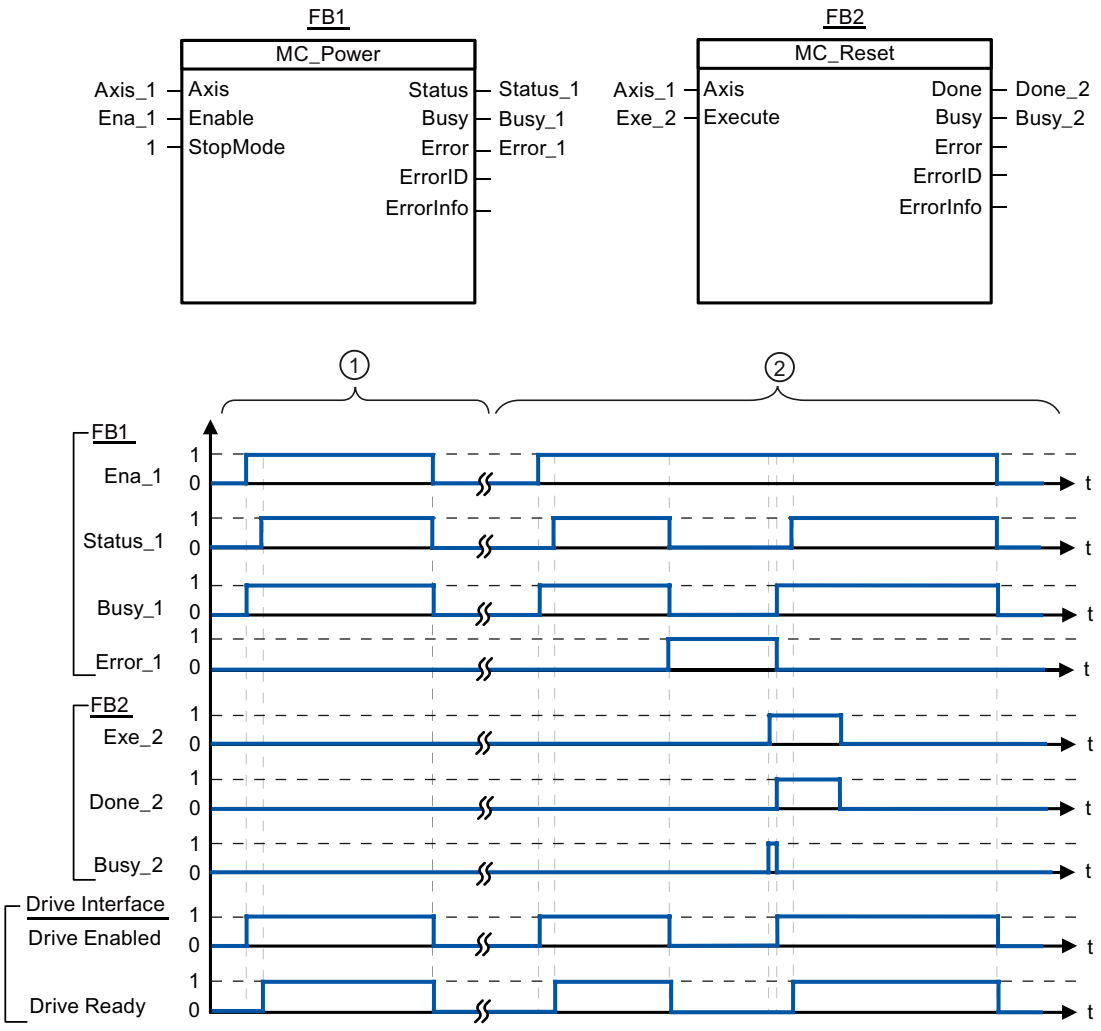

①	Eine Achse wird freigegeben und anschließend wieder gesperrt. Nachdem der Antrieb das Signal "Antrieb bereit" an die CPU zurück gemeldet hat, kann die erfolgreiche Freigabe über "Status_1" abgelesen werden.
②	Nach dem Freigeben einer Achse tritt ein Fehler auf, der das Sperren der Achse zur Folge hat. Der Fehler wird beseitigt und mit "MC_Reset" quittiert. Daraufhin wird die Achse wieder freigegeben.

Siehe auch

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" können "Betriebsfehler mit Stopp der Achse" und "Konfigurationsfehler" quittiert werden. Welche Fehler quittiert werden müssen können Sie der "Liste der ErrorIDs und ErrorInfos" unter "Abhilfe" entnehmen.

Ab Version V3.0 kann nach einem Laden im Betriebszustand RUN die Konfiguration der Achse in den Arbeitsspeicher geladen werden.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Bei einem anstehenden quittierbaren Konfigurationsfehler wurde die Fehlerursache beseitigt (z. B. die Beschleunigung im Technologieobjekt Positionierachse auf einen gültigen Wert geändert).

Ablöseverhalten

Der MC_Reset-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der neue MC_Reset-Auftrag bricht keine anderen laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Restart	INPUT	BOOL	FALSE	(Ab Version V3.0)	
				TRUE	Lädt die Konfiguration der Achse aus dem Ladespeicher in den Arbeitsspeicher. Der Auftrag ist nur bei gesperrter Achse ausführbar. Beachten Sie hierzu die Hinweise zum Laden in die CPU.
				FALSE	Quittiert anstehende Fehler
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Einen mit MC_Reset zu quittierenden Fehler quittieren

Gehen Sie zum Quittieren eines Fehlers folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" den Wert TRUE sowie die Variable des Technologieobjekts <Achsname>.StatusBits.Error den Wert FALSE, so wurde der Fehler quittiert.

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_Home

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3

Beschreibung

Mit Hilfe der Motion Control-Anweisung "MC_Home" gleichen Sie die Achskoordinate auf die reale, physikalische Position des Antriebs ab. Soll die Achse absolut positioniert werden, so ist eine Referenzierung nötig. Folgende Referenzierarten können durchgeführt werden:

- Aktives Referenzieren (Mode = 3)
  Die Referenzpunktfahrt wird automatisch durchgeführt.
- Passives Referenzieren (Mode = 2)
  Beim passiven Referenzieren führt die Motion Control-Anweisung "MC_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse referenziert.

- Direktes Referenzieren Absolut (Mode = 0)
  Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.
- Direktes Referenzieren Relativ(Mode = 1)
  Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Beim Start mit Mode = 0, 1 und 2 darf kein MC_CommandTable-Auftrag aktiv sein.

Ablöseverhalten

Das Ablöseverhalten ist vom gewählten Mode abhängig:

Mode = 0, 1

Der MC_Home-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der MC_Home-Auftrag bricht keine laufenden Motion Control-Aufträge ab. Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 2

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 2, 3

Der neue MC_Home-Auftrag bricht folgenden laufenden Motion Control-Auftrag ab:

- MC_Home-Auftrag Mode = 2

Positionsbezogene Verfahraufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

Mode = 3

Der MC_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Home-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 2, 3
- MC_Halt-Auftrag

- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Position	INPUT	REAL	0.0	• Mode = 0, 2 und 3 Absolute Position der Achse nach dem Abschluss des Referenziervorgangs • Mode = 1 Korrekturwert zur aktuellen Achsposition Grenzwerte: -1.0E+12 ≤ Position ≤ 1.0E+12	
Mode	INPUT	INT	0	Referenziermodus	
				0	Direktes Referenzieren Absolut Neue Achsposition ist der Positionswert des Parameters "Position".
				1	Direktes Referenzieren Relativ Neue Achsposition ist die aktuelle Achsposition + Positionswert des Parameters "Position".
				2	Passives Referenzieren Referenzieren gemäß der Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
				3	Aktives Referenzieren Referenzpunktfahrt gemäß Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt.
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"

Hinweis

Die Referenzierung einer Achse geht unter folgenden Bedingungen verloren:

- Sperren der Achse durch die Motion Control-Anweisung "MC_Power"
- Wechsel zwischen Automatikbetrieb und Handsteuerung
- Mit dem Starten des aktiven Referenzierens. Nach dem erfolgreichen Abschluss des Referenziervorgangs, ist die Referenzierung der Achse wieder vorhanden.
- Nach NETZ-AUS -> NETZ-EIN der CPU
- Nach Neustart der CPU (RUN-STOP -> STOP-RUN)

Eine Achse referenzieren

Gehen Sie beim Referenzieren der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die notwendigen Eingangsparameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Eingangsparameter "Execute"
3. Zeigt der Ausgangsparameter "Done" sowie die Variable des Technologieobjekts <Achsname>.StatusBits.HomingDone den Wert TRUE, so ist das Referenzieren abgeschlossen.

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_Halt

MC_Halt: Achse anhalten V1...3

Beschreibung

Die Motion Control-Anweisung "MC_Halt" bricht alle Bewegungsvorgänge ab und bremst die Achse mit der konfigurierten Verzögerung bis zum Stillstand. Die Stillstandsposition ist nicht definiert.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_Halt-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_Halt-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit Null erreicht

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennnung zum Parameter "ErrorID"	

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_Halt: Funktionsdiagramm V1...3 (Seite 3220)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_Halt: Funktionsdiagramm V1...3

Funktionsdiagramm

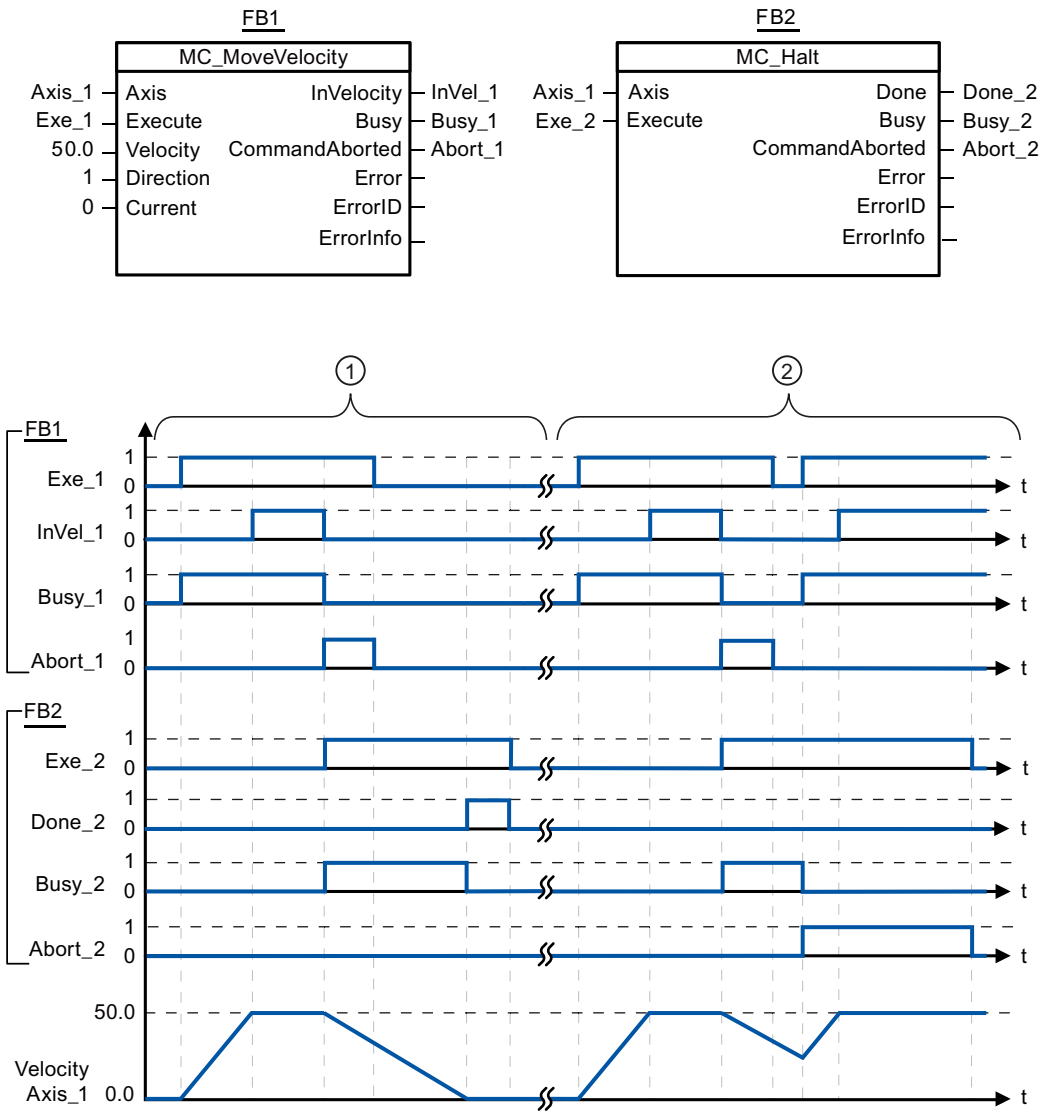

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

①	Die Achse wird durch einen MC_Halt-Auftrag bis zum Stillstand abgebremst. Das Erreichen des Stillstandes wird über "Done_2" gemeldet.
②	Während ein MC_Halt-Auftrag die Achse abbremst wird dieser Auftrag durch einen anderen Verfahrauftrag abgebrochen. Der Abbruch wird über "Abort_2" gemeldet.

Siehe auch

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute

MC_MoveAbsolute: Achse absolut positionieren V1...3

Beschreibung

Die Motion Control-Anweisung "MC_MoveAbsolute" startet eine Positionierbewegung der Achse zu einer absoluten Position.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Die Achse ist referenziert.

Ablöseverhalten

Der MC_MoveAbsolute-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveAbsolute-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Position	INPUT	REAL	0.0	Absolute Zielposition Grenzwerte: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der anzufahrenden Zielposition nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Done	OUTPUT	BOOL	FALSE	TRUE	Absolute Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_MoveAbsolute: Funktionsdiagramm V1...3 (Seite 3223)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_MoveAbsolute: Funktionsdiagramm V1...3

Funktionsdiagramm

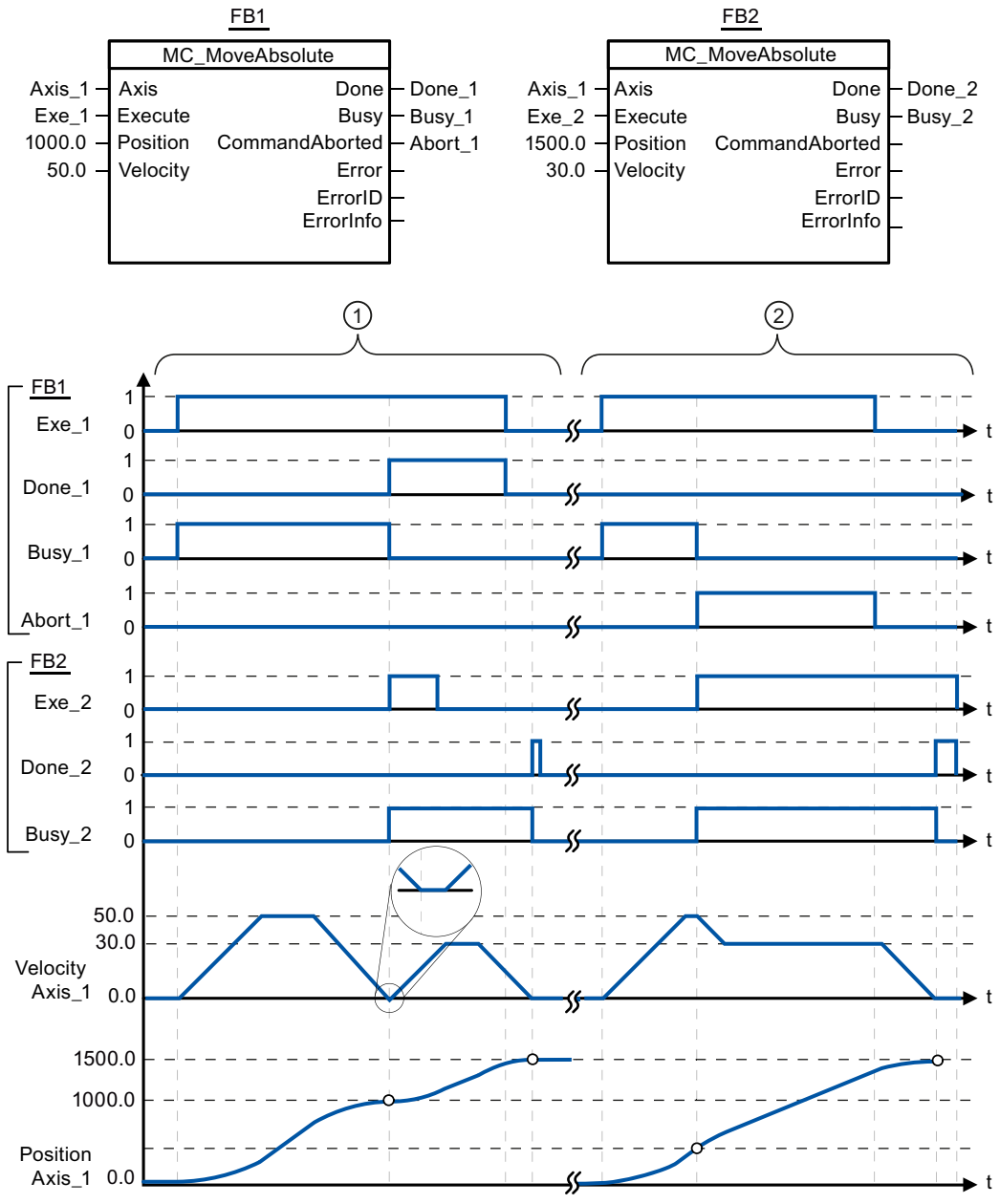

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Eine Achse wird durch einen MC_MoveAbsolute-Auftrag auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveAbsolute-Auftrag, mit Zielposition 1500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
②	Ein laufender MC_MoveAbsolute-Auftrag wird durch einen weiteren MC_MoveAbsolute-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

Siehe auch

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative

MC_MoveRelative: Achse relativ positionieren V1...3

Beschreibung

Die Motion Control-Anweisung "MC_MoveRelative" startet eine Positionierbewegung relativ zur Startposition.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveRelative-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveRelative-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag

- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Distance	INPUT	REAL	0.0	Wegstrecke für den Positioniervorgang Grenzwerte: $-1.0e^{12} \leq$ Distance $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Geschwindigkeit der Achse Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der zu fahrenden Wegstrecke nicht immer erreicht. Grenzwerte: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_MoveRelative: Funktionsdiagramm V1...3 (Seite 3226)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)
MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)
MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_MoveRelative: Funktionsdiagramm V1...3

Funktionsdiagramm

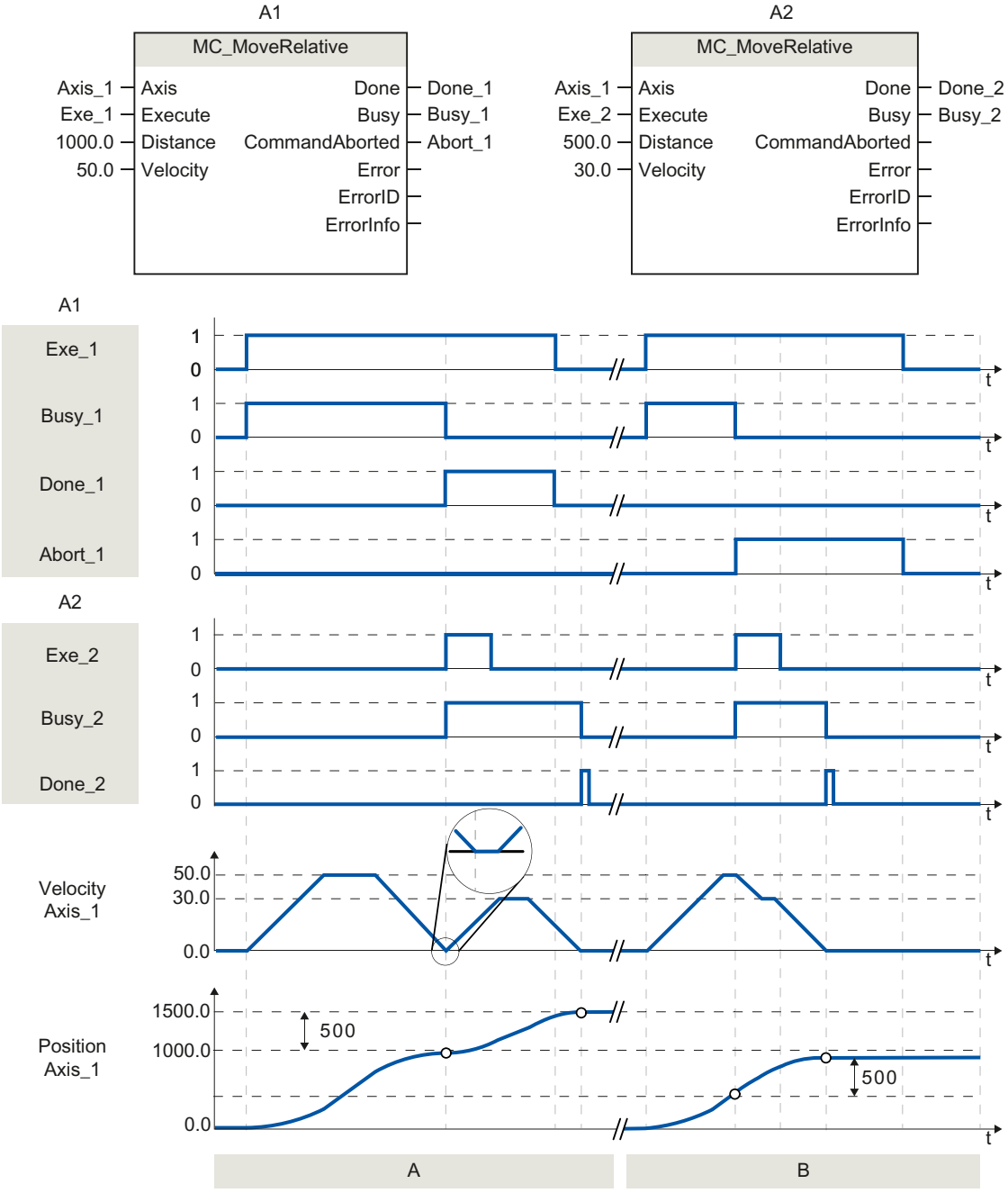

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Die Achse wird durch einen MC_MoveRelative-Auftrag um die Wegstrecke ("Distance") 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveRelative-Auftrag, mit der Wegstrecke 500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet.
②	Ein laufender MC_MoveRelative-Auftrag wird durch einen weiteren MC_MoveRelative-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die neue Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

Siehe auch

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3

Beschreibung

Die Motion Control-Anweisung "MC_MoveVelocity" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveVelocity kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveVelocity-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für das Verfahren der Achse Grenzwerte: Start/Stopp-Geschwindigkeit ≤ |Velocity| ≤ Maximale Geschwindigkeit (Velocity = 0.0 ist erlaubt)	
Direction	INPUT	INT	0	Richtungsvorgabe	
				0	Drehrichtung entsprechend dem Vorzeichen des Werts am Parameter "Velocity"
				1	Drehrichtung positiv (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
				2	Drehrichtung negativ (Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	"Aktuelle Geschwindigkeit beibehalten" deaktiviert. Es werden die Werte der Parameter "Velocity" und "Direction" verwendet.
				TRUE	"Aktuelle Geschwindigkeit beibehalten" aktiviert. Die Werte an den Parametern "Velocity" und "Direction" bleiben unberücksichtigt. Sobald die Achse mit der aktuellen Geschwindigkeit weiterfährt liefert der Parameter "InVelocity" den Wert TRUE.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	• "Current" = FALSE: Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht. • "Current" = TRUE: Die Achse verfährt mit der zum Startzeitpunkt aktuellen Geschwindigkeit.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit Null (Velocity = 0.0)

Ein MC_MoveVelocity-Auftrag mit "Velocity" = 0.0 bricht (wie ein MC_Halt-Auftrag) aktive Bewegungsaufträge ab und stoppt die Achse mit der konfigurierten Verzögerung.

Mit dem Erreichen des Stillstands wird am Ausgangsparameter "InVelocity" für mindestens einen Programmzyklus TRUE angezeigt.

"Busy" zeigt während des Verzögerungsvorgangs den Wert TRUE und wechselt diesen zusammen mit "InVelocity" zu FALSE. Ist der Parameter "Execute" = TRUE gesetzt, so wird "InVelocity" und "Busy" speichernd angezeigt.

Mit dem Start des MC_MoveVelocity-Auftrags wird das Statusbit "SpeedCommand" im Technologieobjekt gesetzt. Das Statusbit "ConstantVelocity" mit dem Stillstand der Achse. Beide Bits werden mit dem Start eines neuen Verfahrauftrags an die neue Situation angepasst.

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_MoveVelocity: Funktionsdiagramm V1...3 (Seite 3230)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)
MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_MoveVelocity: Funktionsdiagramm V1...3

Funktionsdiagramm

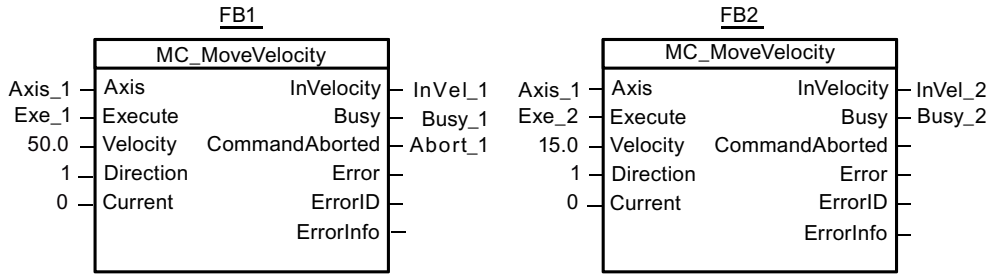

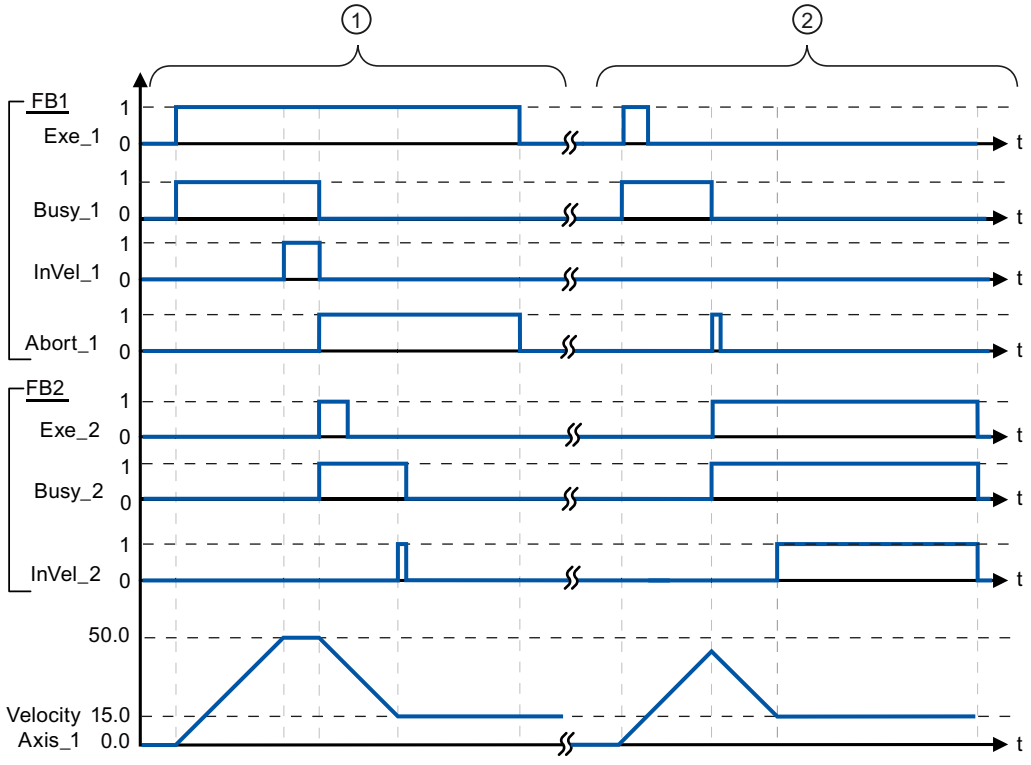

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

①	Ein laufender MC_MoveVelocity-Auftrag meldet über "InVel_1" das Erreichen seiner Zielgeschwindigkeit. Anschließend wird er durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.
②	Ein laufender MC_MoveVelocity-Auftrag wird vor dem Erreichen seiner Zielgeschwindigkeit durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.

Siehe auch

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3

Beschreibung

Die Motion Control-Anweisung "MC_MoveJog" bewegt die Achse konstant mit der vorgegebenen Geschwindigkeit im Tippbetrieb. Verwenden Sie diese Motion Control Anweisung z. B. für Test- und Inbetriebnahmezwecke.

Voraussetzungen

- Das Technologieobjekt Achse wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_MoveJog-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_MoveJog-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag

- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.	
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.	
Sind beide Parameter gleichzeitig TRUE, so stoppt die Achse mit der konfigurierten Verzögerung. An den Parametern "Error", "ErrorID" und "ErrorInfo" wird ein Fehler angezeigt.					
Velocity	INPUT	REAL	10.0	Geschwindigkeitsvorgabe für den Tippbetrieb.	
				Grenzwerte Anweisungsversion V1.0: Start/Stopp-Geschwindigkeit ≤ |Velocity| ≤ Maximale Geschwindigkeit	
				Grenzwerte Anweisungsversion V2.0: Start/Stopp-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_MoveJog: Funktionsdiagramm V1...3 (Seite 3233)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_MoveJog: Funktionsdiagramm V1...3

Funktionsdiagramm

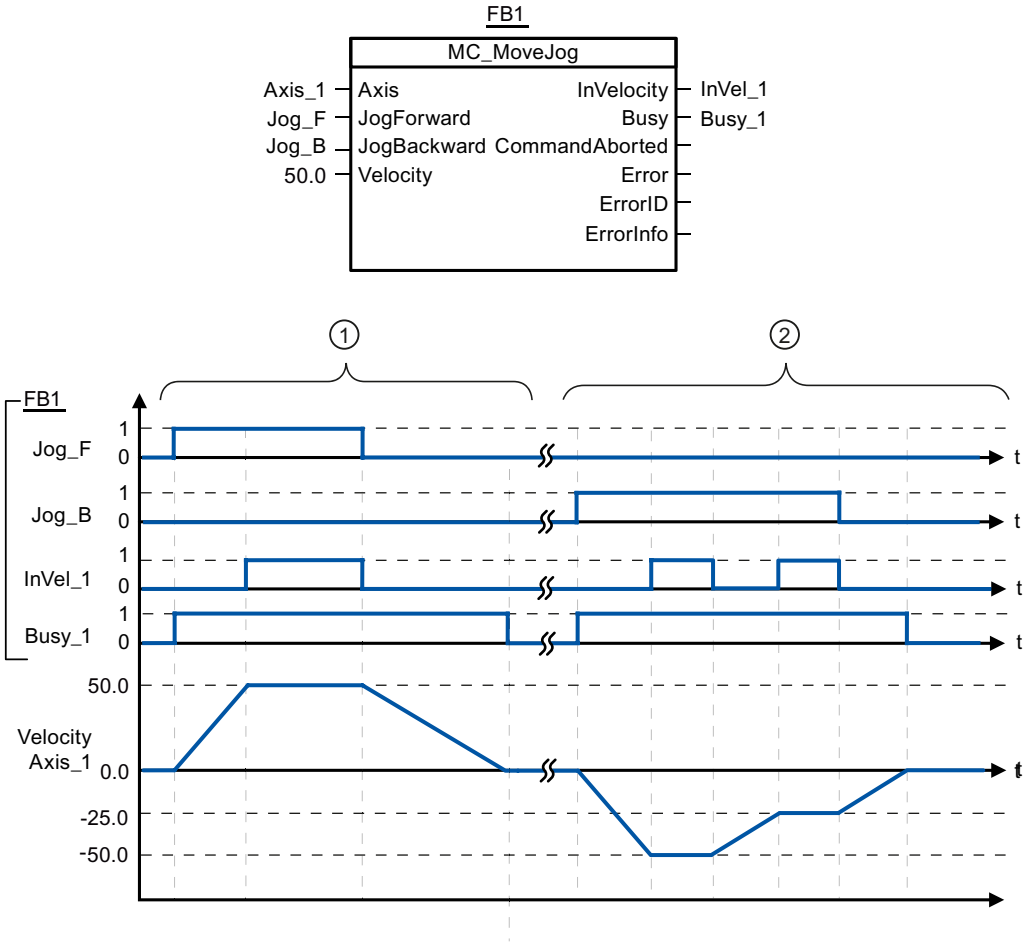

Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

①	Über "Jog_F" wird die Achse im Tippbetrieb in positiver Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVelo_1" gemeldet. Nach dem Rücksetzen von " Jog_F" bremst die Achse wieder bis zum Stillstand ab.
②	Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVelo_1" gemeldet. Nach dem Rücksetzen von " Jog_B" bremst die Achse wieder bis zum Stillstand ab.

Siehe auch

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3

Beschreibung

Die Motion Control-Anweisung "MC_CommandTable" fasst mehrere Einzelaufträge zur Steuerung einer Achse zu einer Bewegungssequenz zusammen.

Voraussetzungen

- Das Technologieobjekt Achse wurde in der Version V2 eingefügt und korrekt konfiguriert.
- Das Technologieobjekt Auftragstabelle wurde eingefügt und korrekt konfiguriert.
- Die Achse ist freigegeben.

Ablöseverhalten

Der MC_CommandTable-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag
- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der neue MC_CommandTable-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC_Home-Auftrag Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute-Auftrag

- MC_MoveRelative-Auftrag
- MC_MoveVelocity-Auftrag
- MC_MoveJog-Auftrag
- MC_CommandTable-Auftrag

Der laufende Motion Control-Auftrag wird mit dem Start des ersten "Positioning Relative", "Positioning Absolute", "Velocity set point" oder "Halt" Auftrag abgebrochen.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Command-Table	INPUT	TO_Command-Table_1	-	Technologieobjekt der Auftragstabelle	
Execute	INPUT	BOOL	FALSE	Start der Auftragstabelle mit steigender Flanke	
StartStep	INPUT	INT	1	Vorgabe ab welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: 1 ≤ StartStep ≤ EndStep	
EndStep	INPUT	INT	32	Vorgabe bis zu welchem Schritt die Auftragstabelle abgearbeitet werden soll Grenzwerte: StartStep ≤ EndStep ≤ 32	
Done	OUTPUT	BOOL	FALSE	TRUE	Auftragstabelle wurde erfolgreich abgearbeitet
Busy	OUTPU	BOOL	FALSE	TRUE	Auftragstabelle ist in Bearbeitung
CommandA-borted	OUTPUT	BOOL	FALSE	TRUE	Die Auftragstabelle wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung der Auftragestabelle ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	
CurrentStep	OUTPUT	INT	0	Aktuell in Bearbeitung befindlicher Schritt der Auftragstabelle	
StepCode	OUTPUT	WORD	16#0000	Benutzerdefinierte Zahlenwert / Bitmuster des aktuell in Bearbeitung befindlichen Schritts	

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3 (Seite 3236)

MC_ChangeDynamic

MC_ChangeDynamic: Dynamikeinstellungen der Achse ändern V2...3

Beschreibung

Die Motion Control-Anweisung "MC_ChangeDynamic" ermöglicht die Änderung folgender Einstellungen der Achse:

- Wert für Hochlaufzeit (Beschleunigung) ändern
- Wert für Rücklaufzeit (Verzögerung) ändern
- Wert für Notstopp-Rücklaufzeit (Notstopp-Verzögerung) ändern
- Wert für Verrundungszeit (Ruck) ändern

Die Wirksamkeit der Änderung entnehmen Sie der Beschreibung der Variablen.

Voraussetzungen

- Das Technologieobjekt Achse wurde in der Version V2 eingefügt.
- Das Technologieobjekt Achse wurde korrekt konfiguriert.

Ablöseverhalten

Ein MC_ChangeDynamic-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC_ChangeDynamic-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis_1	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Change-RampUp	INPUT	BOOL	FALSE	TRUE	Hochlaufzeit entsprechend Eingangsparameter "RampUpTime" ändern

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
RampUp-Time	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung vom Stillstand auf die konfigurierte maximale Geschwindigkeit zu beschleunigen Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Acceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-RampDown	INPUT	BOOL	FALSE	TRUE	Rücklaufzeit entsprechend Eingangsparameter "RampDownTime" geändern
Ramp-DownTime	INPUT	REAL	5.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Deceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
ChangeE-mergency	INPUT	BOOL	FALSE	TRUE	Notstopp-Rücklaufzeit entsprechend Eingangsparameter "EmergencyRampTime" ändern
Emergen-cyRamp-Time	INPUT	REAL	2.00	Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung im Notstopp-Betrieb von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.EmergencyDeceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Change-JerkTime	INPUT	BOOL	FALSE	TRUE	Verrundungszeit entsprechend Eingangsparameter "JerkTime" ändern
JerkTime	INPUT	REAL	0.25	Verrundungszeit (in Sekunden), die auf die Beschleunigungs- und Verzögerungsrampe der Achse angewendet wird Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Jerk beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.	
Done	OUTPUT	BOOL	FALSE	TRUE	Die geänderten Werte wurden in den Technologie-DB geschrieben. Wann die Änderung wirksam wird, können Sie der Beschreibung der Variablen entnehmen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Fehlerinfokennung zum Parameter "ErrorID"	

Hinweis

An den Eingangsparametern "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" können Werte angegeben werden, welche die zulässigen Grenzwerte der resultierenden Parameter: "Beschleunigung", "Verzögerung", "Notstopp-Verzögerung" und "Ruck" überschreiten.

Bitte beachten Sie unter Berücksichtigung der Gleichungen und Grenzwerte in Kapitel: "Technologieobjekt Achse" -> "Konfigurieren des Technologieobjekts" -> "Dynamik" dass Ihre Eingaben innerhalb des gültigen Bereichs liegen.

Siehe auch

S7-1200 Motion Control V4...5 (Seite 3172)

MC_Power: Achse freigeben, sperren V1...3 (Seite 3208)

MC_Reset: Fehler quittieren, Restart Technologieobjekt V1...3 (Seite 3213)

MC_Home: Achse referenzieren, Referenzpunkt setzen V1...3 (Seite 3214)

MC_Halt: Achse anhalten V1...3 (Seite 3218)

MC_MoveAbsolute: Achse absolut positionieren V1...3 (Seite 3221)

MC_MoveRelative: Achse relativ positionieren V1...3 (Seite 3224)

MC_MoveVelocity: Achse mit Geschwindigkeitsvorgabe bewegen V1...3 (Seite 3227)

MC_MoveJog: Achse im Tippbetrieb bewegen V1...3 (Seite 3231)

MC_CommandTable: Achsaufträge als Bewegungssequenz ausführen V2...3 (Seite 3234)

S7-1500 Motion Control

S7-1500 Motion Control V4

MC_Power V4

MC_Power: Technologieobjekt freigeben, sperren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_Power" wird ein Technologieobjekt freigegeben bzw. gesperrt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die zyklische BUS-Kommunikation zwischen Steuerung und Geber ist aufgebaut ("<TO>.StatusSensor[n].CommunicationOK" = TRUE).
- Die zyklische BUS-Kommunikation zwischen Steuerung und Antrieb ist aufgebaut ("<TO>.StatusDrive.CommunicationOK" = TRUE).

Ablöseverhalten

- Ein "MC_Power"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein "MC_Power"-Auftrag mit dem Parameter "Enable" = TRUE gibt ein Technologieobjekt frei und bricht somit keine anderen Motion Control-Anweisungen ab.
- Mit dem Sperren des Technologieobjekts (Parameter "Enable" = FALSE) werden alle Bewegungsaufträge am zugehörigen Technologieobjekt entsprechend dem ausgewählten "StopMode" abgebrochen. Dieser Vorgang kann vom Anwender nicht abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Power":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Das Technologieobjekt wird freigegeben.
				FALSE	Das Technologieobjekt wird gesperrt. Alle laufenden Aufträge am Technologieobjekt werden entsprechend dem parametrierten "StopMode" abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
StartMode	INPUT	DINT	1	0	Positionierachse/Gleichlaufachse nicht lagegeregelt freigeben
				1	Positionierachse/Gleichlaufachse lagegeregelt freigeben
				Der Parameter wirkt initial beim Freigeben der Positionierachse ("Enable" wechselt von "FALSE" nach "TRUE") und beim Freigeben nach dem Quittieren eines Alarms, der zum Sperren der Achse geführt hat. Beim Einsatz einer Drehzahlachse oder eines Externen Gebers wird der Parameter ignoriert.	
StopMode	INPUT	INT	0	Für Technologieobjekt Externer Geber nicht relevant. Wenn Sie ein Technologieobjekt mit einer fallenden Flanke am Parameter "Enable" sperren, bremst die Achse gemäß dem gewählten "StopMode" ab.	
				0	Notstopp Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Notstopp-Rampe" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicDefaults.EmergencyDeceleration)
				1	Sofort-Aus Beim Sperren eines Technologieobjekts wird der Sollwert Null ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
				2	Stopp mit maximalen Dynamikwerten Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicLimits.MaxDeceleration; <TO>.DynamicLimits.MaxJerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Status	OUTPUT	BOOL	FALSE	Freigabestatus des Technologieobjekts	
				FALSE	Gesperrt • Das Technologieobjekt nimmt keine Bewegungsaufträge an. • Drehzahlsteuerung und Positionsregelung sind nicht aktiv. • Die Istwerte des Technologieobjekts werden nicht auf Gültigkeit überprüft.
				TRUE	Freigegeben • Das freigegebene Technologieobjekt nimmt Bewegungsaufträge an. • Drehzahlsteuerung und Positionsregelung sind aktiv. • Die Istwerte des Technologieobjekts sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Freigeben von Technologieobjekten

Setzen Sie zum Freigeben eines Technologieobjekts den Parameter "Enable" auf "TRUE".

Die folgenden zwei Fälle werden unterschieden:

- Freigeben im Stillstand
- Freigeben in Bewegung der Achse

Freigeben im Stillstand

Abhängig vom Parameter "StartMode" wird die Position gehalten ("StartMode" = 1) oder der Geschwindigkeitssollwert null ausgegeben ("StartMode" = 0). Wenn der Parameter "Status" den Wert "TRUE" zeigt, ist das Technologieobjekt freigegeben.

Freigeben in Bewegung der Achse

Wenn "StartMode" = 1 ist, wird die Position zum Zeitpunkt des Setzens des "Enable"-Eingangs als Sollposition für den Lageregler wirksam. Abhängig der unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung wird die Achse auf null abgebremst und auf die Sollposition ausgeregelt. Wenn dabei Überwachungen oder dynamische Grenzen überschritten werden, führt dies zu entsprechenden Alarmreaktionen.

Wenn "StartMode" = 0 ist, wird die Achse durch die Vorgabe des Geschwindigkeitssollwerts null so weit wie möglich abgebremst. Überwachungen und dynamische Grenzen sind in diesem Fall nicht aktiv.

Wenn der Parameter "Status" den Wert "TRUE" zeigt, ist das Technologieobjekt freigegeben.

Hinweis

Automatische Freigabe nach Quittierung eines Technologiealarms

Wenn das Technologieobjekt wegen eines Technologiealarms gesperrt wird, wird das Technologieobjekt nach dem Beseitigen der Ursache und dem Quittieren des Alarms automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Parameter "Enable" während dieses Vorgangs den Wert "TRUE" beibehalten hat.

Sperren von Technologieobjekten

Setzen Sie zum Sperren eines Technologieobjekts den Parameter "Enable" auf "FALSE".

Wenn eine Achse in Bewegung ist, wird diese gemäß dem gewählten "StopMode" abgebremst und zum Stillstand gebracht.

Wenn die Parameter "Busy" und "Status" den Wert "FALSE" zeigen, ist das Sperren des Technologieobjekts abgeschlossen.

Antriebsanbindung über PROFIdrive

Bei der Ankopplung eines Antriebs über PROFIdrive werden Sollwert, Freigabe und Antriebsstatus über das PROFIdrive-Telegramm übertragen.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird das Technologieobjekt freigegeben. Der Antrieb wird gemäß PROFIdrive-Norm aktiviert.
  Wenn die Variable "<TO>.StatusDrive.InOperation" den Wert "TRUE" zeigt, ist der Antrieb bereit, Sollwerte auszuführen. Der Parameter "Status" wird auf den Wert "TRUE" gesetzt.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert "FALSE" gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Der Antrieb wird gemäß PROFIdrive-Norm deaktiviert.

Analoge Antriebsanbindung

Der Sollwert wird über einen Analogausgang ausgegeben. Optional können ein Freigabesignal über Digitalausgang (<TO>.Actor.Interface.EnableDriveOutput) und ein Bereitschaftssignal über Digitaleingang (<TO>.Actor.Interface.DriveReadyInput) konfiguriert werden.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird der Freigabe-Ausgang ("Enable drive output") gesetzt.
  Wenn der Antrieb das Bereitschaftssignal über den Bereit-Eingang ("Drive ready input") zurückmeldet, wird der Parameter "Status" und die Variable "<TO>.StatusDrive.InOperation" des Technologieobjekts auf "TRUE" gesetzt und der Sollwert auf den Analogausgang geschaltet.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert "FALSE" gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Mit dem Erreichen von Sollwert null wird der Freigabe-Ausgang auf "FALSE" gesetzt.

Weitere Informationen

Weiterführende Informationen zum Freigeben und Sperren von Technologieobjekten und Antrieben finden Sie im Anhang ""MC_Power"-Funktionsdiagramme".

MC_Power: Funktionsdiagramm V4

Funktionsdiagramm: Freigeben eines Technologieobjekts und Beispiel für Alarmreaktion

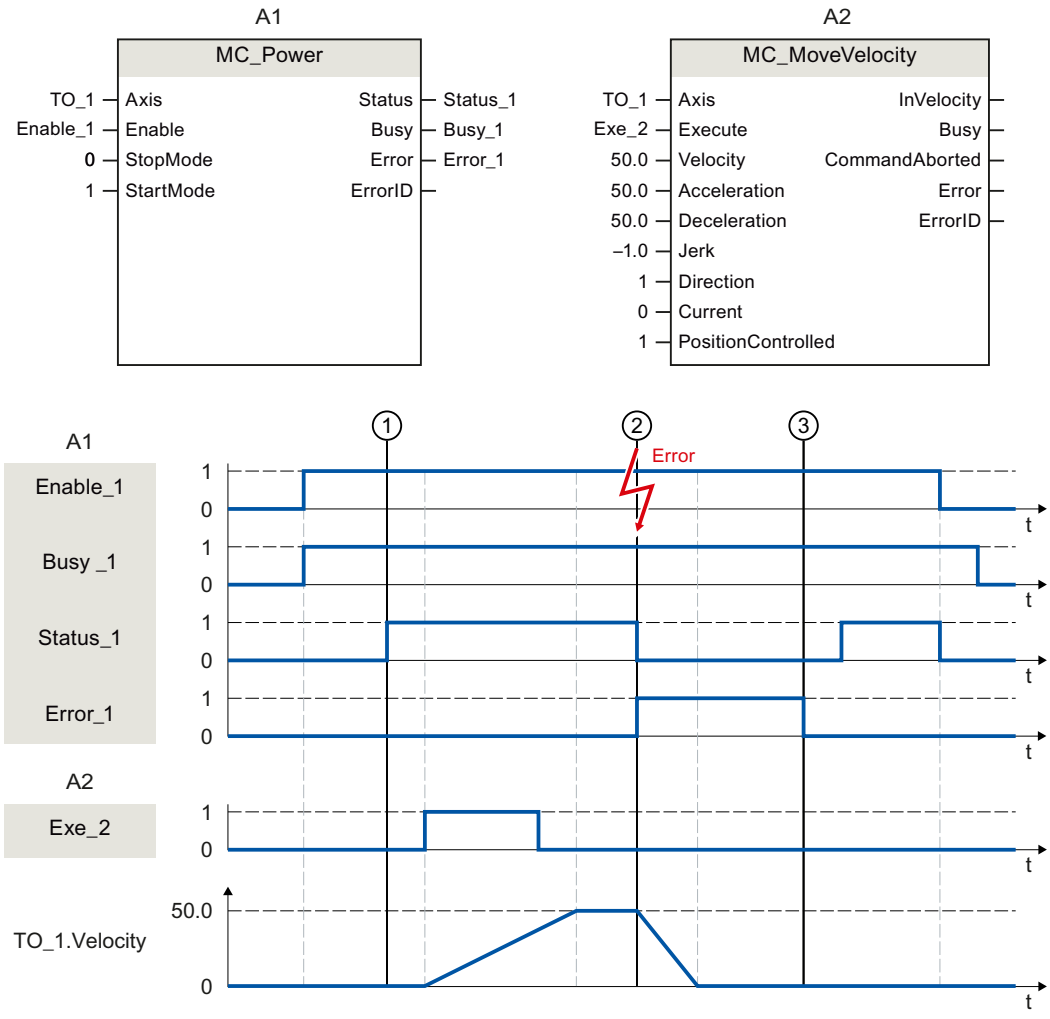

Ein Technologieobjekt wird mit "Enable_1" = TRUE freigegeben. Die erfolgreiche Freigabe kann an "Status_1" zum Zeitpunkt ① abgelesen werden. Anschließend wird die Achse mit einem "MC_MoveVelocity"-Auftrag (A2) verfahren. Der Geschwindigkeitsverlauf der Achse kann an "TO_1.Velocity" abgelesen werden.

Zum Zeitpunkt ② tritt ein Fehler am Technologieobjekt auf, der das Sperren des Technologieobjekts zur Folge hat (Alarmreaktion: Freigabe wegnehmen). Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Mit dem Sperren des Technologieobjekts wird "Status_1" rückgesetzt. Da die Achse nicht über "Enable_1" = FALSE gesperrt wurde, ist der gewählte "StopMode" nicht relevant. Die Fehlerursache wird beseitigt und der Alarm wird zum Zeitpunkt ③ quittiert.

Da "Enable_1" weiterhin gesetzt ist, wird das Technologieobjekt wieder freigegeben. Die erfolgreiche Freigabe kann an "Status_1" abgelesen werden. Abschließend wird das Technologieobjekt über "Enable_1" = FALSE gesperrt.

MC_Reset V4

MC_Reset: Alarme quittieren, Restart Technologieobjekt V4

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" quittieren Sie alle Technologie-Alarme, die im Anwenderprogramm quittierbar sind. Mit der Quittierung werden auch die "Error"- und "Warning"-Bits im Technologie-Datenbaustein zurückgesetzt.

Mit der Motion Control-Anweisung "MC_Reset" mit "Restart" = TRUE starten Sie das Neuinitialisieren (Restart) von Technologieobjekten. Mit dem Restart des Technologieobjekts werden neue Konfigurationsdaten im Technologie-Datenbaustein übernommen.

Anwendbar auf

- Alle Technologieobjekte

Voraussetzung

- Technologieobjekte Positionier-/Gleichlaufachse, Drehzahlachse und Externer Geber:
  Für einen Restart muss das Technologieobjekt gesperrt sein.
  ("MC_Power.Status" = FALSE und "MC_Power.Busy" = FALSE)

Ablöseverhalten

- Parameter "Restart" = FALSE:
  Die Bearbeitung der Anweisung "MC_Reset" mit Parameter "Restart" = FALSE kann durch andere Motion Control-Aufträge abgebrochen werden. Der "MC_Reset"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.
- Parameter "Restart" = TRUE:
  Die Bearbeitung der Anweisung "MC_Reset" mit Parameter "Restart" = TRUE kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Reset":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Object	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Restart	INPUT	BOOL	FALSE	TRUE	Neuinitialisierung des Technologieobjekts und Quittierung anstehender Technologie-Alarme. Das Technologieobjekt wird mit den konfigurierten Startwerten neu initialisiert.
				FALSE	Quittierung anstehender Technologie-Alarme
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurden quittiert. Der Restart wurde ausgeführt.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Technologie-Alarme quittieren

Gehen Sie zum Quittieren von Technologie-Alarmen folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen Sie den Parameter "Restart" = FALSE.
3. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Parameter "Execute".
   Wenn der Parameter "Done" den Wert "TRUE" zeigt, wurde der Fehler quittiert.

Hinweis

Quittieren mit "Restart" = FALSE

Wenn nur die Technologie-Alarme quittiert werden sollen, setzen Sie "Restart" = FALSE. Während eines Restarts kann das Technologieobjekt nicht verwendet werden.

Restart eines Technologieobjekts

Gehen Sie zum Restart eines Technologieobjekts wie folgt vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen Sie den Parameter "Restart" = TRUE.
3. Führen Sie den Restart durch eine steigende Flanke am Parameter "Execute" aus.
   Wenn der Parameter "Done" den Wert "TRUE" zeigt, ist der Restart des Technologieobjekts abgeschlossen.

Weitere Informationen zum Restart finden Sie im Kapitel "Restart von Technologieobjekten".

MC_Home V4

MC_Home: Technologieobjekt referenzieren, Referenzpunkt setzen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_Home" stellen Sie den Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung her. Der Positionsistwert am Technologieobjekt wird dabei einer Referenzmarke zugeordnet. Diese Referenzmarke repräsentiert eine bekannte mechanische Position.

Der Referenziervorgang erfolgt gemäß der am Parameter "Mode" gewählten Betriebsart und der Konfiguration unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren".

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und ≥ V2.0 finden Sie im Kapitel "Versionsübersicht".

Für die Dynamikwerte Beschleunigung, Verzögerung und Ruck werden die voreingestellten Werte unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik-Voreinstellung" verwendet.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Externer Geber

Die folgende Tabelle zeigt, welche Betriebsarten mit den jeweiligen Technologieobjekten möglich sind:

Betriebsart	Positionierachse/ Gleichlaufachse mit Inkrementalgeber	Positionierachse/ Gleichlaufachse mit Absolutwertgeber	Externer Geber inkrementell	Externer Geber absolut
Aktives Referenzieren ("Mode" = 3, 5)	X	-	-	-
Passives Referenzieren ("Mode" = 2, 8,10)	X	-	X	-
Setzen der Istposition ("Mode" = 0)	X	X	X	X

Betriebsart	Positionierachse/ Gleichlaufachse mit Inkrementalgeber	Positionierachse/ Gleichlaufachse mit Absolutwertgeber	Externer Geber inkrementell	Externer Geber absolut
Relative Verschiebung der Istposition ("Mode" = 1)	X	X	X	X
Absolutwertgeberjustage ("Mode" = 6, 7)	-	X	-	X

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- "Mode" = 2, 3, 5, 8, 10
  Das Technologieobjekt ist freigegeben.
- "Mode" = 0, 1, 6, 7, 8
  Die Geberistwerte sind gültig (<TO>.StatusSensor[n].State = 2).
- "Mode" = 6, 7
  Die Achse befindet sich im lagegeregelten Betrieb.

Ablöseverhalten

Das Ablöseverhalten für "MC_Home"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Home":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Der angegebene Wert wird dem gewählten "Mode" entsprechend verwendet.	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	INT	0	Betriebsart	
				0	Direktes Referenzieren (Absolut) Die aktuelle Position des Technologieobjekts wird auf den Wert des Parameters "Position" gesetzt.
				1	Direktes Referenzieren (Relativ) Die aktuelle Position des Technologieobjekts wird um den Wert des Parameters "Position" verschoben.
				2	Passives Referenzieren (ohne Rücksetzen) Funktion wie "Mode" 8 mit dem Unterschied, dass der Status "referenziert" mit dem Aktivieren der Funktion **nicht** zurückgesetzt wird.
				3	Aktives Referenzieren Das Technologieobjekt Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem Wert des Parameters "Position".
				4	Reserviert
				5	Aktives Referenzieren (Parameter "Position" unwirksam) Das Technologieobjekt Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Aktives Referenzieren" konfigurierten Referenzpunkt. (<TO>.Homing.HomePosition)
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Position wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n].AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Position wird auf den Wert des Parameters "Position" gesetzt. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
					(<TO>.StatusSensor[n].AbsEncoderOffset)
				8	Passives Referenzieren Beim Erkennen der Referenzmarke wird der Istwert auf den Wert des Parameters "Position" gesetzt.
				9	Abbruch Passives Referenzieren Ein laufender Auftrag zum passiven Referenzieren wird abgebrochen.
				10	Passives Referenzieren (Parameter "Position" unwirksam) Beim Erkennen der Referenzmarke wird der Istwert auf den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Passives Referenzieren" konfigurierten Referenzpunkt gesetzt. (<TO>.Homing.HomePosition)
ReferenceMarkPosition	OUTPUT	LREAL	0.0	Anzeige der Position, an der das Technologieobjekt referenziert wurde (gültig bei "Done" = TRUE)	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjekts wird unter folgenden Bedingungen rückgesetzt (<TO>.StatusWord.X5 (HomingDone)):

- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags mit "Mode" = 3, 5, 8, 10
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS → NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Referenzieren eines Technologieobjekts mit "Mode" = 1 … 8, 10

Gehen Sie zum Referenzieren eines Technologieobjekts folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Mode" die gewünschte Referenzierfunktion an.
3. Versorgen Sie die notwendigen Parameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Parameter "Execute".
   Wenn der Parameter "Done" den Wert "TRUE" zeigt, ist der "MC_Home"-Auftrag gemäß dem gewählten "Mode" abgeschlossen. Der Status "Referenziert" des Technologieobjekts wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Referenziert" angezeigt (<TO>.StatusWord.X5 (HomingDone)).

Abbruch eines passiven Referenziervorgangs mit "Mode" = 9

Das Technologieobjekt wird mit "Mode" = 9 nicht referenziert. Wenn ein laufender "MC_Home"-Auftrag zum passiven Referenzieren ("Mode" = 2, 8, 10) durch einen anderen "MC_Home"-Auftrag mit "Mode" = 9 abgelöst wird, wird der laufende Auftrag mit dem Parameter "CommandAborted" = TRUE abgebrochen. Der ablösende Auftrag mit "Mode" = 9 meldet die erfolgreiche Durchführung mit Parameter "Done" = TRUE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_Halt V4

MC_Halt: Achse anhalten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_Halt" bremsen Sie eine Achse bis zum Stillstand ab.

Mit den Parametern "Jerk" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bremsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_Halt"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Halt":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
AbortAcceleration	INPUT	BOOL	FALSE	FALSE	Die aktuelle Beschleunigung beim Start des Auftrags wird über den konfigurierten Ruck abgebaut. Danach wird die Verzögerung aufgebaut.
				TRUE	Die Beschleunigung wird beim Start des Auftrags auf 0.0 gesetzt und die Verzögerung unmittelbar aufgebaut.
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit null ist erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Abbremsen einer Achse mit "MC_Halt"

Gehen Sie zum Abbremsen einer Achse bis zum Stillstand folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die Parameter "Deceleration", "Jerk" und "AbortAcceleration" mit den gewünschten Werten.
3. Starten Sie den "MC_Halt"-Auftrag mit einer steigenden Flanke am Parameter "Execute". An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt. Der Stillstand der Achse wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Stillstand" angezeigt (<TO>.StatusWord.X7 (Standstill)).

Abbremsen einer Achse mit aktiver Kraft-/Momentenbegrenzung

Bei aktiver Kraft-/Momentenbegrenzung kann das zulässige Moment unzureichend sein, um die berechnete Bremsrampe zu verfahren. Der "MC_Halt"-Auftrag meldet dann nicht "Done" = TRUE.

Verwenden Sie zum Abbremsen einer Achse mit aktiver Kraft-/Momentenbegrenzung die Motion Control-Anweisung "MC_MoveVelocity" mit "MC_MoveVelocity.Velocity" = 0.0 und "MC_MoveVelocity.PositionControlled" = FALSE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V4 (Seite 3264)

MC_Halt: Funktionsdiagramm V4

Funktionsdiagramm: Anhalten einer Achse und ablösendes Auftragsverhalten

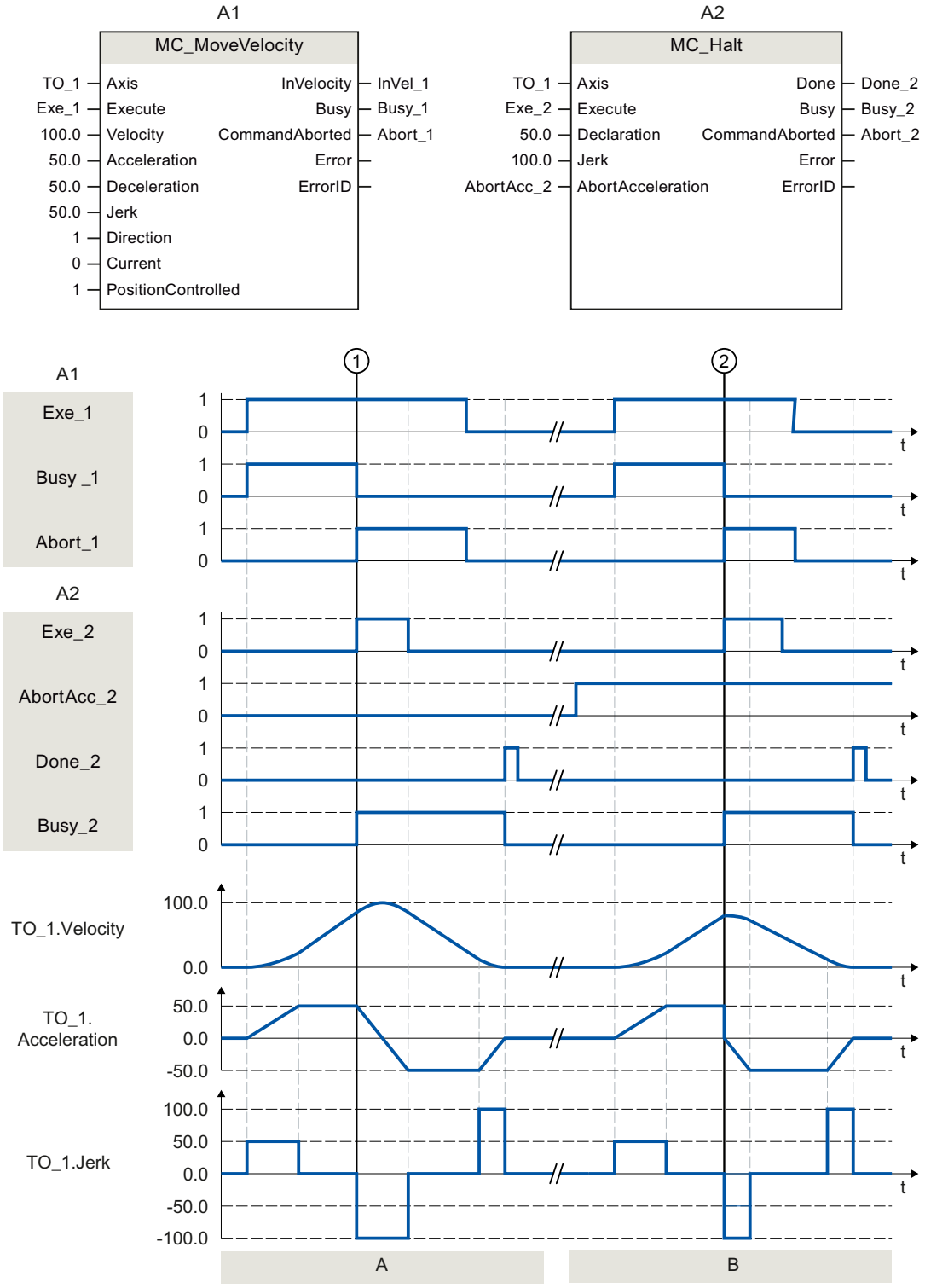

Abschnitt A	Eine Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Zum Zeitpunkt ① wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Mit "AbortAcc_2" = FALSE wird die aktuelle Beschleunigung mit dem angegebenen Ruck abgebaut. Danach wird die Verzögerung aufgebaut und die Achse bis zum Stillstand abgebremst. Der Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.
Abschnitt B	Die Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Zum Zeitpunkt ② wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Mit "AbortAcc_2" = TRUE wird die aktuelle Beschleunigung unmittelbar auf null gesetzt und die Verzögerung aufgebaut. Die Achse wird bis zum Stillstand abgebremst. Der Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.

MC_MoveAbsolute V4

MC_MoveAbsolute: Achse absolut positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveAbsolute" verfahren Sie eine Achse auf eine absolute Position.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.
- Das Technologieobjekt ist referenziert.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveAbsolute"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Absolute Zielposition	
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Direction	INPUT	INT	1	Bewegungsrichtung der Achse Dieser Parameter wird nur bei aktivierter Modulo-Funktion ausgewertet. "Technologieobjekt > Konfiguration > Grundparameter > Modulo aktivieren"	
				1	Positive Richtung
				2	Negative Richtung
				3	Kürzester Weg
Done	OUTPUT	BOOL	FALSE	TRUE	Die Zielposition ist erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse auf eine absolute Position

Gehen Sie zum Verfahren einer Achse auf eine absolute Position folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Position" die gewünschte Zielposition an.
3. Starten Sie den "MC_MoveAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveAbsolute: Funktionsdiagramm V4

Funktionsdiagramm: Absolutes Positionieren einer Achse und ablösendes Auftragsverhalten

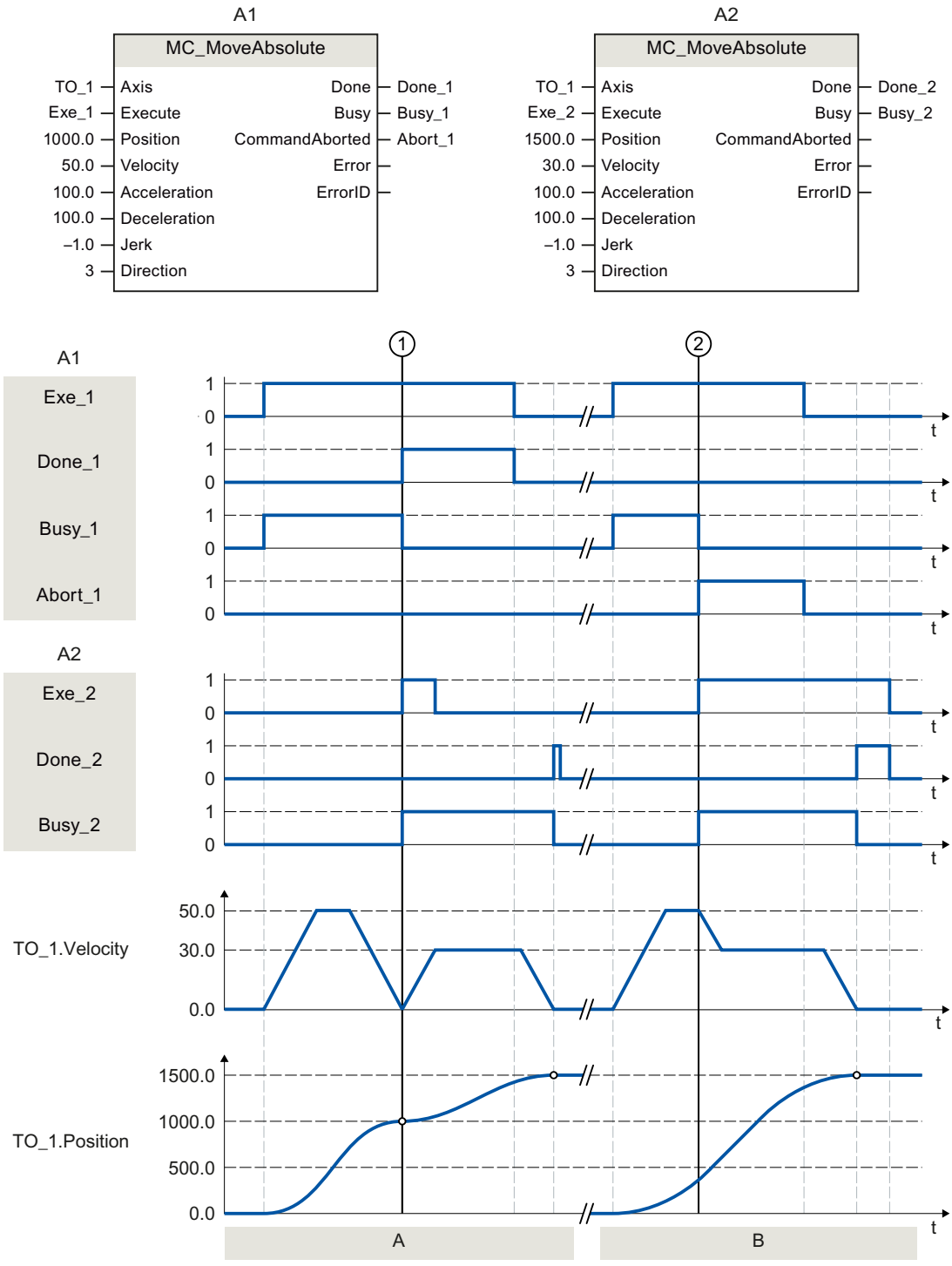

Abschnitt A	Eine Achse wird durch einen "MC_MoveAbsolute"-Auftrag (A1) auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveAbsolute"-Auftrag (A2) mit Zielposition 1500.0 gestartet. Das Erreichen der Zielposition 1500.0 wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveAbsolute"-Auftrag (A1) wird zum Zeitpunkt ② durch einen weiteren "MC_MoveAbsolute"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird auf die geänderte Geschwindigkeit abgebremst und auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveRelative V4

MC_MoveRelative: Achse relativ positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveRelative" verfahren Sie eine Achse relativ zu der Position, die bei Beginn der Auftragsbearbeitung vorliegt.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveRelative"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Distance	INPUT	LREAL	0.0	Wegstrecke für den Positioniervorgang (negativ oder positiv)	
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Done	OUTPUT	BOOL	FALSE	TRUE	Die Zielposition ist erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse relativ zur Startposition

Gehen Sie zum Verfahren einer Achse relativ zur Startposition folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveRelative: Funktionsdiagramm V4

Funktionsdiagramm: Relatives Positionieren einer Achse und ablösendes Auftragsverhalten

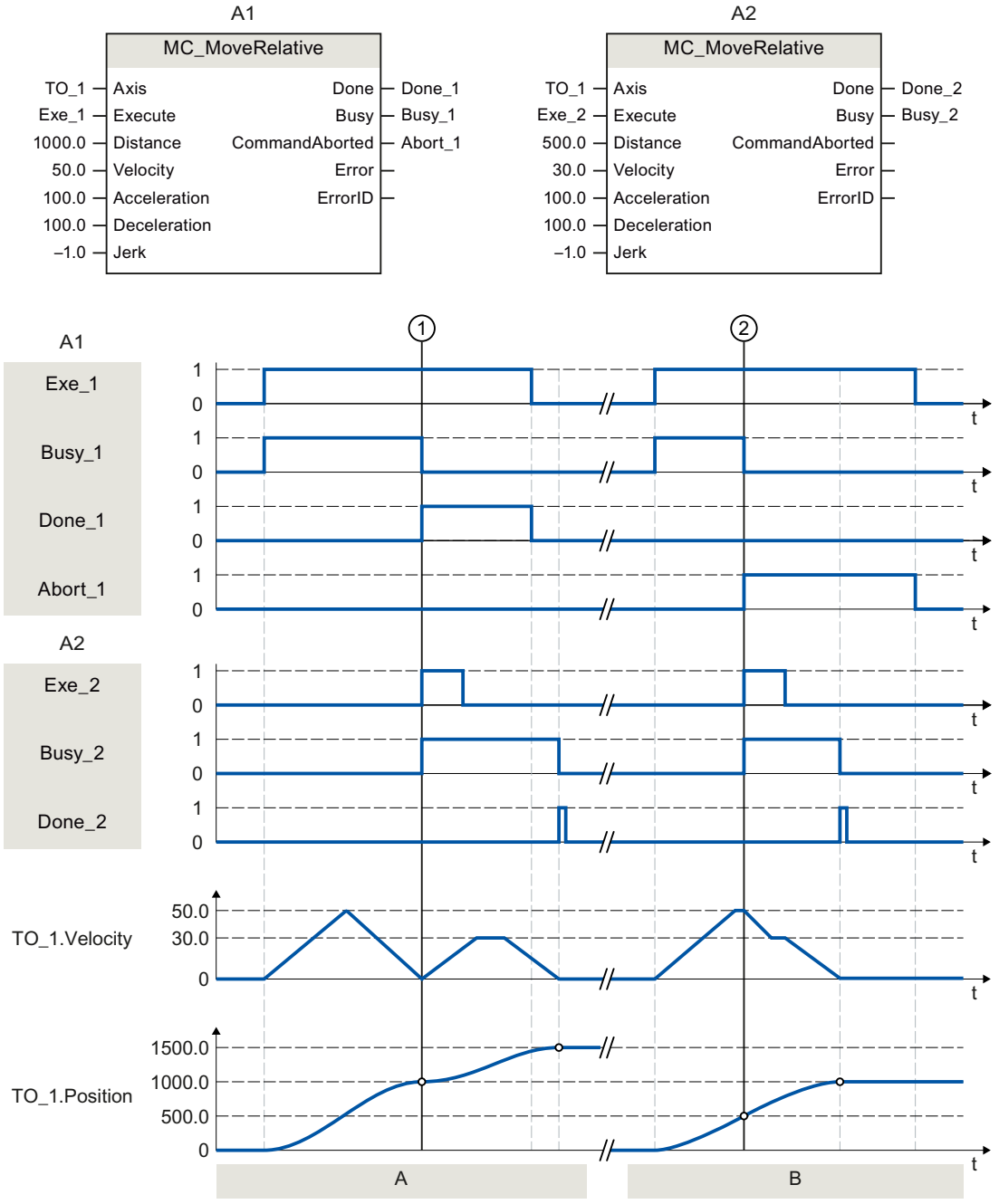

Abschnitt A	Die Achse wird durch einen "MC_MoveRelative"-Auftrag (A1) um die Wegstrecke ("Distance") 1000.0 verfahren (Startposition ist hier die Position 0.0). Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveRelative"-Auftrag (A2) mit der Wegstrecke 500.0 gestartet. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveRelative"-Auftrag (A1) wird durch einen weiteren "MC_MoveRelative"-Auftrag (A2) abgelöst. Der Abbruch wird zum Zeitpunkt ② über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveVelocity V4

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveVelocity" bewegen Sie eine Achse mit konstanter Geschwindigkeit/Drehzahl.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveVelocity"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveVelocity":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit/Solldrehzahl für den Bewegungsvorgang ("Velocity" = 0.0 ist erlaubt.)	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Direction	INPUT	INT	0	Drehrichtung der Achse	
				0	Das Vorzeichen der am Parameter "Velocity" angegebenen Geschwindigkeit definiert die Drehrichtung.
				1	Positive Drehrichtung Der Betrag von "Velocity" wird verwendet.
				2	Negative Drehrichtung Der Betrag von "Velocity" wird verwendet.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	Deaktiviert Die Werte der Parameter "Velocity" und "Direction" werden berücksichtigt.
				TRUE	Aktiviert Die Werte an den Parametern "Velocity" und "Direction" werden nicht berücksichtigt. Die zum Funktionsstart aktuelle Geschwindigkeit und Richtung werden beibehalten. Sobald die Achse mit der zum Funktionsstart aktuellen Geschwindigkeit weiter verfährt, liefert der Parameter "InVelocity" den Wert "TRUE".
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der MC_MoveVelocity-Auftrag ausgeführt wird. Danach gilt die Einstellung des folgenden Auftrags. Beim Einsatz einer Drehzahlachse wird der Parameter ignoriert.	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit/Solldrehzahl ist erreicht. Eine konstante Sollgeschwindigkeit/Solldrehzahl wird ausgegeben.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit/Solldrehzahl null ("Velocity" = 0.0)

Ein "MC_MoveVelocity"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit/Solldrehzahl null wird am Parameter "InVelocity" der Wert "TRUE" angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Die Parameter "InVelocity" und "Busy" zeigen solange den Wert "TRUE", bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Verfahren einer Achse mit konstanter Geschwindigkeit/Drehzahl

Gehen Sie zum Verfahren einer Achse mit konstanter Geschwindigkeit/Drehzahl folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Velocity" die Geschwindigkeit/Drehzahl an, mit welcher die Achse verfahren werden soll.
3. Starten Sie den "MC_MoveVelocity"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.
   Wenn der Parameter "InVelocity" den Wert "TRUE" zeigt, wurde die Sollgeschwindigkeit/Solldrehzahl erreicht. Die Achse wird mit dieser Geschwindigkeit konstant weiter verfahren. Die Parameter "InVelocity" und "Busy" zeigen solange den Wert "TRUE", bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit/Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit dem Erreichen der neu errechneten Geschwindigkeit/Drehzahl ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveVelocity: Funktionsdiagramm V4

Funktionsdiagramm: Verfahren einer Achse mit Geschwindigkeitsvorgabe und ablösendes Auftragsverhalten

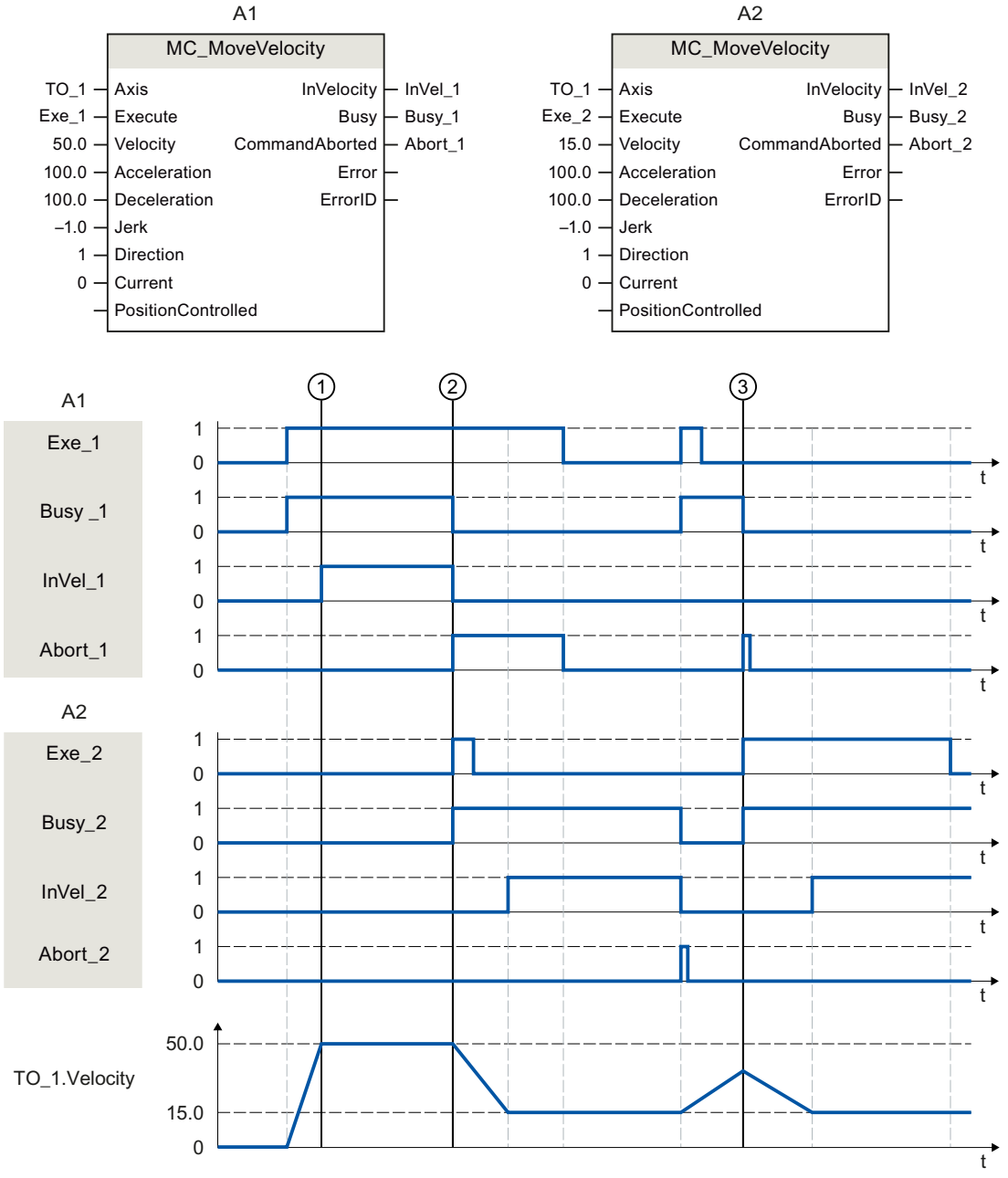

Ein über "Exe_1"angestoßener "MC_MoveVelocity"-Auftrag (A1) beschleunigt die Achse und meldet zum Zeitpunkt ① über "InVel_1" das Erreichen der Sollgeschwindigkeit 50.0.

Zum Zeitpunkt ② wird der Auftrag durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen

Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

Der laufende "MC_MoveVelocity"-Auftrag (A2) wird durch einen weiteren "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Abbruch wird über "Abort_2" gemeldet. Die Achse wird auf die neue Sollgeschwindigkeit 50.0 beschleunigt. Vor dem Erreichen der Sollgeschwindigkeit wird der aktuelle "MC_MoveVelocity"-Auftrag (A1) zum Zeitpunkt ③ durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

MC_MoveJog V4

MC_MoveJog: Achse im Tippbetrieb bewegen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveJog" bewegen Sie eine Achse im Tippbetrieb.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveJog"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveJog":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
JogForward	INPUT	BOOL	FALSE	TRUE	Solange der Parameter "TRUE" ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.
JogBackward	INPUT	BOOL	FALSE	TRUE	Solange der Parameter "TRUE" ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit/Solldrehzahl für den Bewegungsvorgang	
				≥ 0.0	Der angegebene Wert wird verwendet.
				< 0.0	Der Betrag des angegebenen Werts wird verwendet.
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der "MC_MoveJog"-Auftrag ausgeführt wird. Danach gilt die Einstellung des folgenden Auftrags. Beim Einsatz einer Drehzahlachse wird der Parameter ignoriert.	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit/Solldrehzahl ist erreicht. Eine konstante Sollgeschwindigkeit/Solldrehzahl wird ausgegeben.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit/Solldrehzahl null ("Velocity" = 0.0)

Ein "MC_MoveJog"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit/Solldrehzahl null wird am Parameter "InVelocity" der Wert "TRUE" angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Verfahren einer Achse im Tippbetrieb

Gehen Sie zum Verfahren einer Achse im Tippbetrieb folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Verfahren Sie die Achse mit "JogForward" in positiver Richtung oder mit "JogBackward" in negativer Richtung.
   An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.
   Wenn sowohl "JogForward" als auch "JogBackward" auf TRUE gesetzt ist, wird die Achse mit der zuletzt gültigen Verzögerung abgebremst. Der Fehler 16#8007 (falsche Richtungsangabe) wird ausgegeben.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit/Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveJog: Funktionsdiagramm V4

Funktionsdiagramm: Verfahren einer Achse im Tippbetrieb

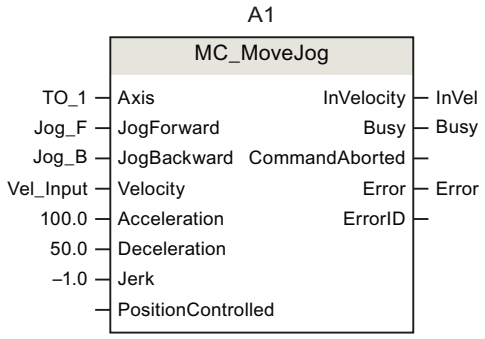

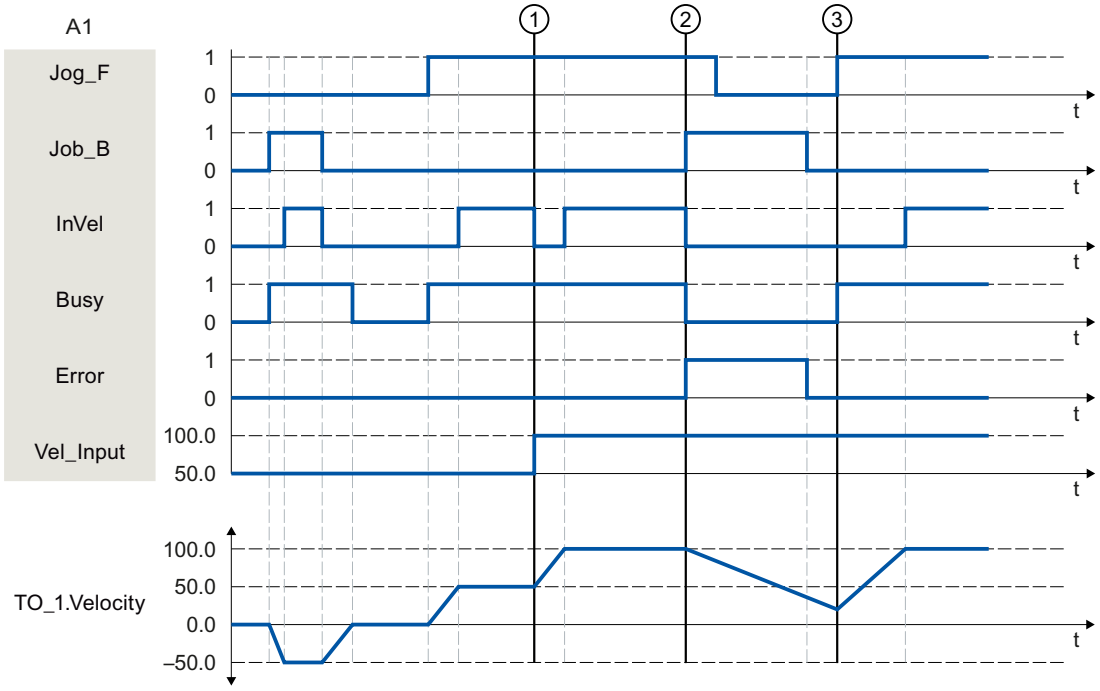

Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Sollgeschwindigkeit -50.0 wird über "InVel" = TRUE gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse abgebremst und zum Stillstand gebracht. Anschließend wird die Achse über "Jog_F" in positiver Richtung verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel" = TRUE gemeldet.

Zum Zeitpunkt ① wird bei gesetztem "Jog_F" die Sollgeschwindigkeit über "Vel_Input" auf 100.0 geändert. Alternativ können Sie die Sollgeschwindigkeit auch über den Geschwindigkeits-Override ändern. "InVel" wird rückgesetzt. Die Achse wird beschleunigt. Das Erreichen der neuen Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

Bei gesetztem "Jog_F" wird zum Zeitpunkt ② ebenfalls "Jog_B" gesetzt. Wenn sowohl "Jog_F" als auch "Jog_B" gesetzt sind, wird die Achse mit der zuletzt gültigen Verzögerung

abgebremst. Über "Error" wird ein Fehler angezeigt und am Ausgang "ErrorID" der Fehler 16#8007 (falsche Richtungsangabe) ausgegeben.

Dieser Fehler wird durch Rücksetzen beider Eingänge "Jog_F" und "Jog_B" behoben.

Noch während der Bremsrampe wird zum Zeitpunkt ③ "Jog_F" gesetzt. Die Achse wird auf die zuletzt konfigurierte Geschwindigkeit beschleunigt. Das Erreichen der Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

MC_MoveSuperimposed V4

MC_MoveSuperimposed: Achse überlagernd positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveSuperimposed" starten Sie eine relative Positionierbewegung, die einer laufenden Basisbewegung überlagert wird.

Mit den Parametern "VelocityDiff", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung. Die Dynamikwerte werden zu den Werten der Basisbewegung addiert. Die Dauer der Basisbewegung wird durch eine überlagernde Bewegung nicht verlängert.

Die Dynamik der Gesamtbewegung der Achse ergibt sich durch die Addition der Dynamikwerte der Basisbewegung und der überlagerten Bewegung.

Das Verhalten der Gesamtbewegung unterscheidet sich je nach Art der Basisbewegung:

- Die Basisbewegung ist eine Einzelachsbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gesamtbewegung ist auf die konfigurierten Dynamikgrenzen begrenzt.
- Die Basisbewegung ist eine Gleichlaufbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gleichlaufbewegung der Folgeachse wird nicht auf die Dynamikgrenzen der Folgeachse begrenzt.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Leitachse im Gleichlauf wirkt sich auf die Leitachse und auf die Folgeachse aus.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Folgeachse im Gleichlauf wirkt sich nur auf die Folgeachse aus.

Im Technologie-Datenbaustein und im TIA Portal wird immer die Dynamik der Gesamtbewegung angezeigt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveSuperimposed"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveSuperimposed":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt der Achse	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Distance	INPUT	LREAL	0.0	Zusätzliche Wegstrecke für den überlagernden Positioniervorgang (negativ oder positiv)	
VelocityDiff	INPUT	LREAL	-1.0	Maximale Geschwindigkeitsabweichung gegenüber der laufenden Bewegung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Done	OUTPUT	BOOL	FALSE	TRUE	Überlagernde Positionierung abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Überlagerte Positionierbewegung starten

Um mit der Motion Control-Anweisung "MC_MoveSuperimposed" eine überlagerte Positionierbewegung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zusätzlich zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveSuperimposed"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MoveSuperimposed: Funktionsdiagramm V4

Funktionsdiagramm: Achsen überlagernd positionieren

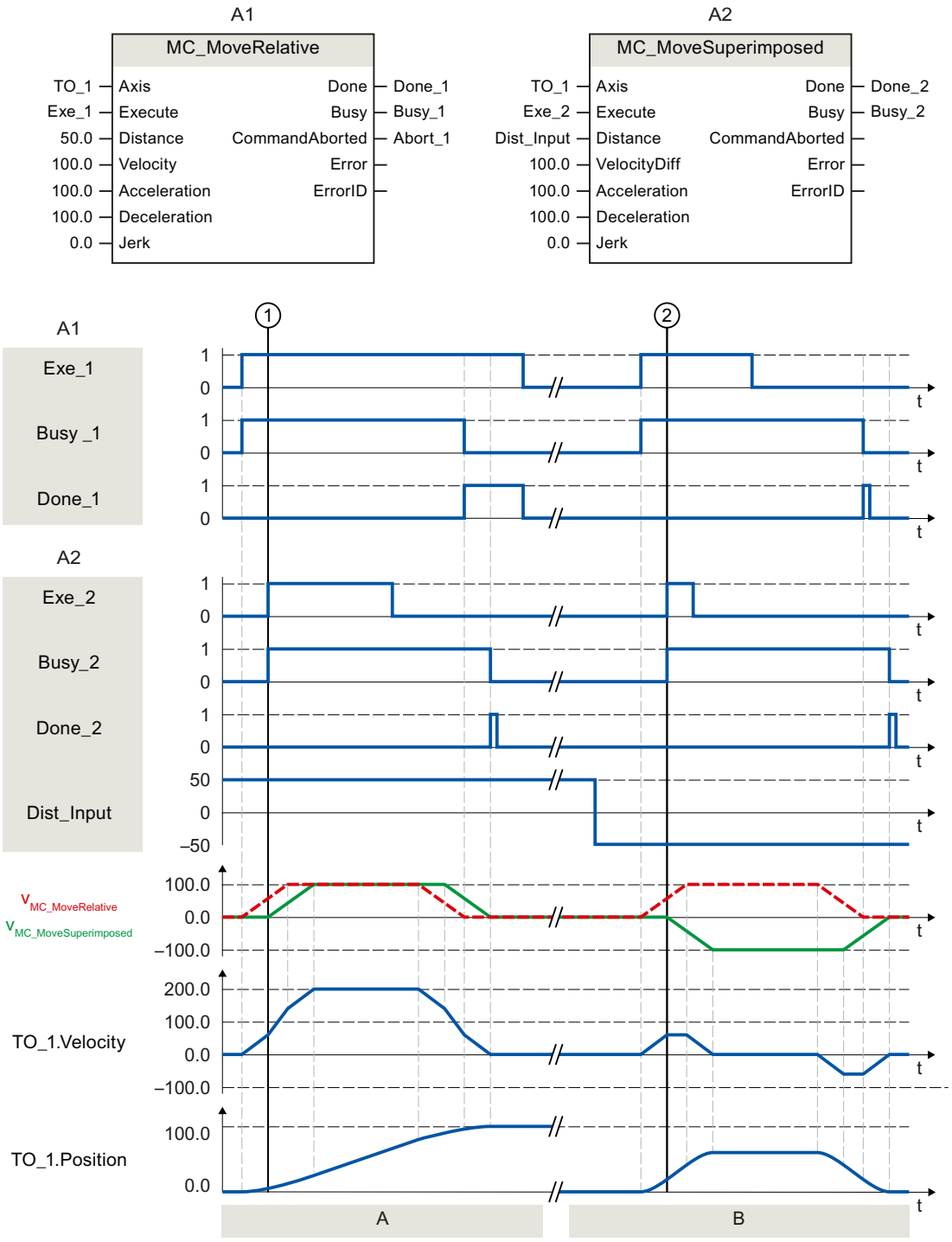

Abschnitt A	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ① wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke 50.0 angestoßen. Die Achse wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50 + 50 = 100.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.
Abschnitt B	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ② wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke -50.0 angestoßen. Die Achse reversiert und wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50.0 - 50.0 = 0.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.

MC_SetSensor V4

MC_SetSensor: Alternativen Geber als operativ wirksamen Geber umschalten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetSensor" schalten Sie den Geber für die Lageregelung der Achse um.

Über den Parameter "Mode" 2 und 3 kann der Istwert des adressierten Gebers ohne Umschaltung angepasst werden.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt und der alternative Geber wurden korrekt konfiguriert.
- Kein Restart- und kein "MC-Home"-Auftrag aktiv.

Ablöseverhalten

- Ein "MC_SetSensor"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_SetSensor"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetSensor":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Sensor	INPUT	INT	1	Nummer des neuen Gebers (1 bis 4).	
Mode	INPUT	DINT	0	Der Mode bestimmt die Positionsangleichung zwischen dem alten und dem neuen Geber.	
				0	Geber umschalten und aktuelle Istposition auf den neuen Geber übertragen Bei dieser Geberumschaltung werden Sprünge in der Positionsregelung verhindert. Eine stoßfreie Umschaltung der Geber ist möglich.
				1	Geber umschalten, ohne die Istposition abzugleichen **Hinweis** Bei aktiver Lageregelung wirkt eine zusätzliche Differenz der beiden Geber als zusätzliche Regelabweichung und kann zu einer Ausgleichbewegung führen.
				2	Istwert übertragen Die aktuelle Istposition wird auf den am Parameter "Sensor" angegebenen Geber übertragen.
				3	Istwert übertragen Die Istposition des "Referenzgebers" (Parameter "ReferenceSensor") wird auf den am Parameter "Sensor" angegebenen Geber übertragen.
ReferenceSensor	INPUT	INT	1	Nummer des Referenzgebers (siehe Parameter "Mode" = 3)	
Done	OUTPUT	BOOL	FALSE	TRUE	Geber für Lageregelung der Achse wurde umgeschaltet.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Umschalten auf Absolutwertgeber

Wenn Sie den Geber auf einen Absolutwertgeber umschalten und den Istwert übertragen ("Mode" = 2, 3), wird der Istwert mit dem Wert des Absolutwertgebers und dem Absolutwert-Offset verrechnet. Beim Umschalten auf einen anderen Geber wird die Verrechnung des Istwerts verworfen. Der Absolutwertgeber liefert wieder den Absolutwert + Absolutwert-Offset (<TO>.StatusSensor[n].AbsEncoderOffset) ohne die Verrechnung durch den "MC_SetSensor"-Auftrag.

Messtaster, Nocken, Nockenspur

MC_MeasuringInput V4

MC_MeasuringInput: Einmaliges Messen starten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MeasuringInput" wird ein einmaliges Messen gestartet.

Mit dem einmaligen Messen können mit einem Messauftrag eine oder zwei Flanken erfasst werden. Dem Messereignis wird die Position vom jeweiligen Technologieobjekt Achse oder Externer Geber zugeordnet. Das Messergebnis wird am Funktionsbaustein und im Technologie-Datenbaustein angezeigt und kann im Anwenderprogramm weiterverarbeitet werden. Damit ist der Messauftrag abgeschlossen.

Ein erneuter Messauftrag muss wieder neu über "MC_MeasuringInput.Execute" = TRUE gestartet werden.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Der Geber der Achse muss im Status "valid" (StatusSensor[1..4].State = valid) sein. Anderenfalls wird der Messauftrag am Funktionsbaustein mit einer Fehleranzeige abgelehnt.
- Das Messen über PROFIdrive ist während eines aktiven oder passiven Referenzierens nicht möglich.

Ablöseverhalten

Das Ablöseverhalten für "MC_MeasuringInput"-Aufträge ist im Kapitel "Ablöseverhalten V4: Messtasteraufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MeasuringInput":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
MeasuringInput	INPUT	TO_MeasuringInput	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Mode	INPUT	DINT	0	Art der Messung	
				0	Messung der nächsten steigenden Flanke
				1	Messung der nächsten fallenden Flanke
				2	Messung der beiden nächsten Flanken
				3	Messung an beiden Flanken, beginnend mit der steigenden Flanke • Steigende Flanke = "MeasuredValue1" (Messwert 1) • Fallende Flanke = "MeasuredValue2" (Messwert 2)
				4	Messung an beiden Flanken, beginnend mit der fallenden Flanke • Fallende Flanke = "MeasuredValue1" • Steigende Flanke = "MeasuredValue2"
MeasuringRange	INPUT	BOOL	FALSE	Erfassung der Messwerte Beachten Sie die zeitlichen Randbedingungen.	
				FALSE	Messwerte immer erfassen
				TRUE	Messwerte nur innerhalb des Messbereichs erfassen
StartPosition	INPUT	LREAL	0.0	Anfangsposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
EndPosition	INPUT	LREAL	0.0	Endposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Baustein wurde fertig bearbeitet. Messwerte sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Messauftrag wurde abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
MeasuredValue1	OUTPUT	LREAL	0.0	Erster Messwert	
MeasuredValue2	OUTPUT	LREAL	0.0	Zweiter Messwert (bei Messung an beiden Flanken)	

Siehe auch

Ablöseverhalten V4: Messtasteraufträge (Seite 3396)

MC_MeasuringInput: Funktionsdiagramm V4

Funktionsdiagramm: Einmaligen Messauftrag starten

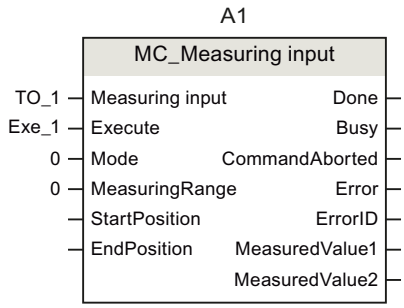

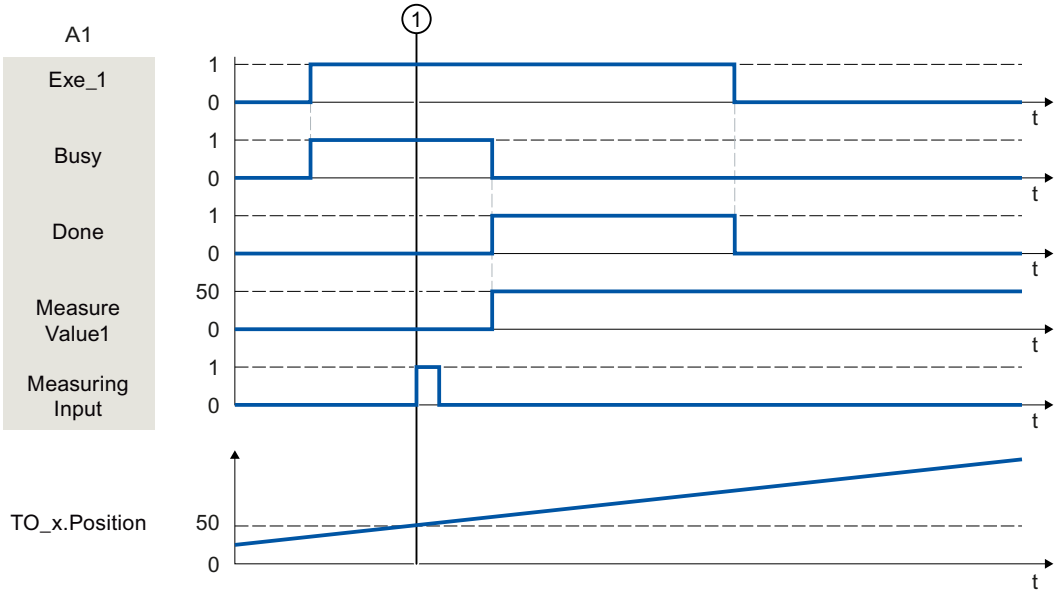

Ein über "Execute" angestoßener "MC_MeasuringInput"-Auftrag mit eingestelltem "Mode" = 0 führt eine Messung der nächsten steigenden Flanke durch. Eine erfolgreich durchgeführte Messung wird mit "Done" = TRUE signalisiert. Der ermittelte Messwert ① (im Beispiel 50) wird über "MeasuredValue1" ausgegeben.

MC_MeasuringInputCyclic V4

MC_MeasuringInputCyclic: Zyklisches Messen starten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MeasuringInputCyclic" wird eine zyklische Messung gestartet.

Mit dem zyklischen Messen werden in jedem Servotakt bis zu zwei Messereignisse vom System erfasst und die dazugehörigen Messpositionen angezeigt. Die Messungen werden zyklisch fortgesetzt, bis sie per Befehl beendet werden.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das zyklische Messen ist nur beim Messen über TM Timer DIDQ möglich.
- Der operativ wirksame Geber der Achse muss im Status "valid" ("StatusSensor[1..4].State" = valid) sein. Anderenfalls wird der Messauftrag am Funktionsbaustein mit einer Fehleranzeige abgelehnt.

Ablöseverhalten

Das Ablöseverhalten für "MC_MeasuringInputCyclic"-Aufträge ist im Kapitel "Ablöseverhalten V4: Messtasteraufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MeasuringInputCyclic":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
MeasuringInput	INPUT	TO_MeasuringInput	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	DINT	0	Art der Messung	
				0	Messung der steigenden Flanken
				1	Messung der fallenden Flanken
				2	Messung der beiden Flanken
MeasuringRange	INPUT	BOOL	FALSE	Erfassung der Messwerte Beachten Sie die zeitlichen Randbedingungen.	
				FALSE	Messwerte immer erfassen
				TRUE	Messwerte nur innerhalb des Messbereichs erfassen
StartPosition	INPUT	LREAL	0.0	Anfangsposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
EndPosition	INPUT	LREAL	0.0	Endposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Messauftrag wurde abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
MeasuredValue1	OUTPUT	LREAL		Erster Messwert	
MeasuredValue2	OUTPUT	LREAL		Zweiter Messwert (bei Messung von mehreren Flanken in einem Servotakt)	
MeasuredValue1Counter	OUTPUT	UDINT	0	Zählwert für den ersten Messwert	
MeasuredValue2Counter	OUTPUT	UDINT	0	Zählwert für den zweiten Messwert	
LostEdgeCounter1	OUTPUT	UDINT	0	Zählwert für die verloren gegangenen Flanken im Takt der Erfassung vom ersten Messwert	
LostEdgeCounter2	OUTPUT	UDINT	0	Zählwert für die verloren gegangenen Flanken im Takt der Erfassung vom zweiten Messwert	

Siehe auch

Ablöseverhalten V4: Messtasteraufträge (Seite 3396)

MC_MeasuringInputCyclic: Funktionsdiagramm V4

Funktionsdiagramm: Zyklischen Messauftrag starten

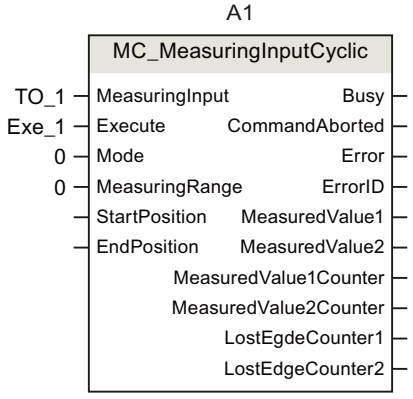

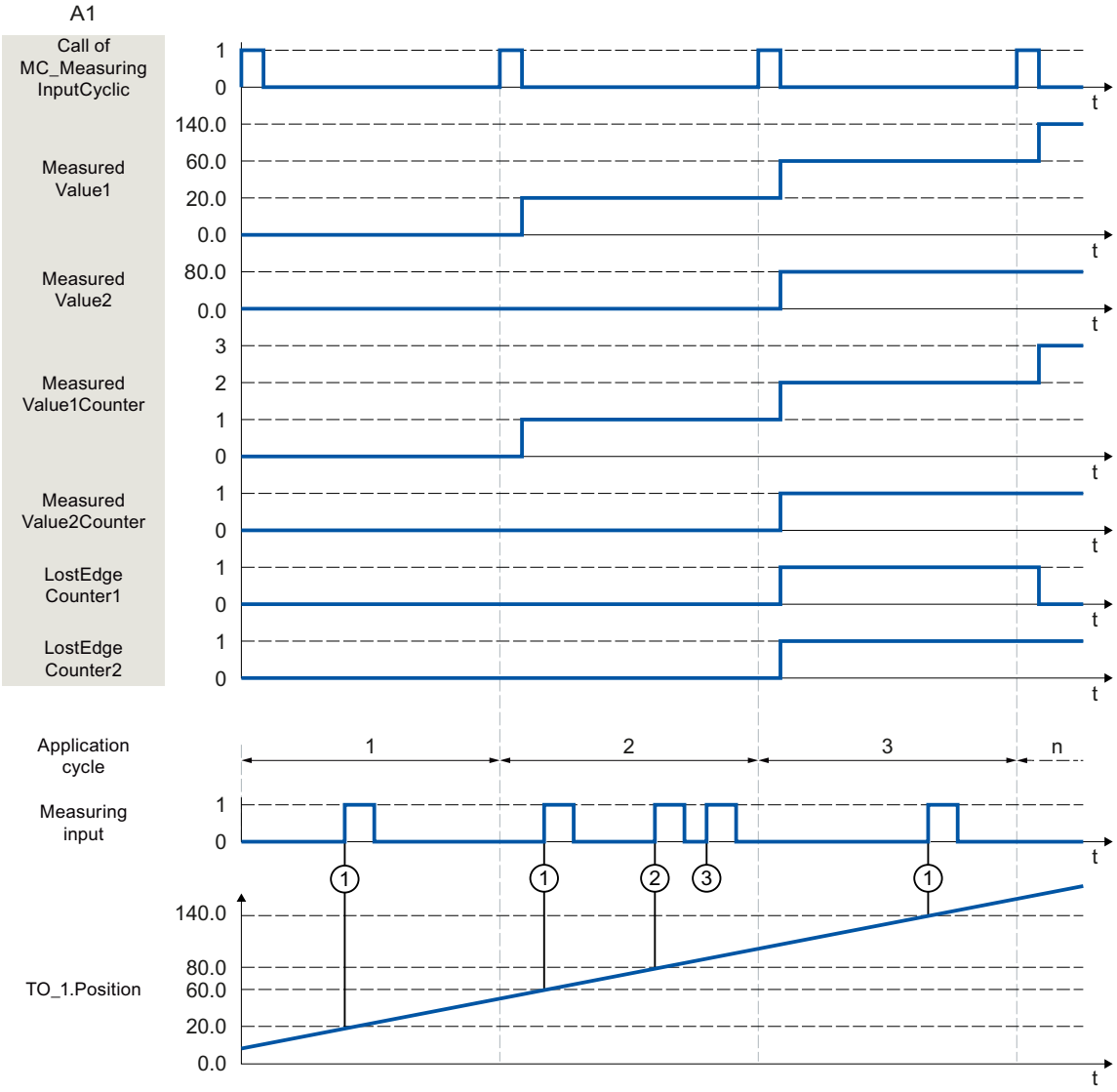

Ein über "Execute" (Signal und Startpunkt nicht dargestellt) angestoßener "MC_MeasuringInputCyclic"-Auftrag (z. B. Aufruf im MC-PreServo [OB67]) mit eingestelltem "Mode" = 0 und ohne vorgegebenen Messbereich "MeasuringRange = FALSE führt eine Messung der steigenden Flanken durch. Durch den Aufruf der Motion Control-Anweisung "MC_MeasuringInputCyclic" im MC-PreServo [OB67] oder MC-PreServo [OB67] erfolgt sowohl der Aufruf des Messauftrags als auch die Messung selbst im gleichen Applikationszyklus.

Die ermittelte Istposition zum Zeitpunkt der ersten steigenden Flanke ① im Servotakt wird über "MeasuredValue1" ausgegeben und die Zählvariable "MeasuredValue1Counter" um "1" hochgezählt.

Die ermittelte Istposition zum Zeitpunkt der zweiten steigenden Flanke ② im Servotakt wird über "MeasuredValue2" ausgegeben und die Zählvariable "MeasuredValue2Counter" um "1" hochgezählt.

Treten im gleichen Servotakt weitere steigende Flanken ③ auf, werden diese im LostEdgeCounter1 und LostEdgeCounter2 festgehalten.

MC_AbortMeasuringInput V4

MC_AbortMeasuringInput: Aktives Messen abbrechen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_AbortMeasuringInput" wird ein aktiver einmaliger oder zyklischer Messauftrag abgebrochen.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.

Ablöseverhalten

Das Ablöseverhalten für "MC_AbortMeasuringInput"-Aufträge ist im Kapitel "Ablöseverhalten V4: Messtasteraufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_AbortMeasuringInput":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
MeasuringInput	INPUT	TO_MeasuringInput	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Done	OUTPUT	BOOL	0	Der Funktionsbaustein wurde bearbeitet. Der Messauftrag wurde deaktiviert.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Messtasteraufträge (Seite 3396)

MC_OutputCam V4

MC_OutputCam: Nocken aktivieren/deaktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_OutputCam" aktivieren Sie den eingestellten Nocken.

Abhängig vom Nockentyp sind die folgenden Eingangsparameter wirksam:

- Wegnocken
  - "OnPosition"
  - "OffPosition"
- Zeitnocken
  - "OnPosition"
  - "Duration"

Mit den Parametern "Mode" und "Direction" bestimmen Sie die Betriebsart und die Wirkrichtung des Nockens.

Bei "MC_OutputCam.Enable" = TRUE werden die Eingangsparameter immer gelesen und sind mit dem nächsten Servotakt wirksam.

Bei einem technologischen Alarm wird der Nocken nach erfolgter Fehlerquittierung wieder bearbeitet.

Anwendbar auf

- Nocken

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das übergeordnete Technologieobjekt muss eine gültige Position haben.
- Sollwertnocken werden bei einer Achse im nicht lagegeregelten Betrieb nicht ausgegeben.
- Ein Sollwertnocken mit der Einstellung "MC_OutputCam.OnPosition" = 0 schaltet bei Sollposition = 0.

Ablöseverhalten

Der "MC_OutputCam" wird abgebrochen durch:

- Das Sperren des Nockens mit "MC_OutputCam.Enable" = FALSE
- Im Anwenderprogramm kann nur eine Instanz des Funktionsbausteins "MC_OutputCam" auf einen Nocken aktiv sein. Eine zweite Instanz des Funktionsbausteins "MC_OutputCam" auf einen bereits aktiven Nocken wird mit Fehler abgelehnt.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_OutputCam":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
OutputCam	INPUT	TO_OutputCam	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	FALSE	Nocken wird gesperrt.
				TRUE	Nocken wird bearbeitet.
OnPosition	INPUT	LREAL	0.0	Anfangsposition des Nockens [Einheit des verschalteten Technologieobjekts] Positionsangaben, die bei aktiver Modulofunktion an der Achse oder des Externen Gebers außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
OffPosition	INPUT	LREAL	0.0	Endposition des Wegnockens [Einheit des verschalteten Technologieobjekts] Positionsangaben, die bei aktiver Modulofunktion an der Achse oder des Externen Gebers außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Duration	INPUT	LREAL	0.0	Einschaltdauer des Zeitnockens [Einheit: ms] Der Wert für die Einschaltdauer eines Zeitnockens muss größer als 0.0 sein.	
Mode	INPUT	DINT	1	Betriebsart	
				1	Standard Nockenfunktionalität (Ausgabe nicht invertiert)
				2	Nockenfunktionalität mit Ausgabe invertiert
				3	Nocken immer ein (Während "Enable" = TRUE)
Direction	INPUT	DINT	1	Wirkrichtung des Nockens	
				1	Positive Richtung
				2	Negative Richtung
				3	Beide Richtungen
CamOutput	OUTPUT	BOOL	-	Statusanzeige bezogen auf den letzten Aufruf der Motion Control-Anweisung "MC_OutputCam"	
Busy	OUTPUT	BOOL	FALSE	TRUE	Die Bearbeitung des Nockens ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_OutputCam: Funktionsdiagramm V4

Funktionsdiagramm: Aktivieren des eingestellten Nockentyps

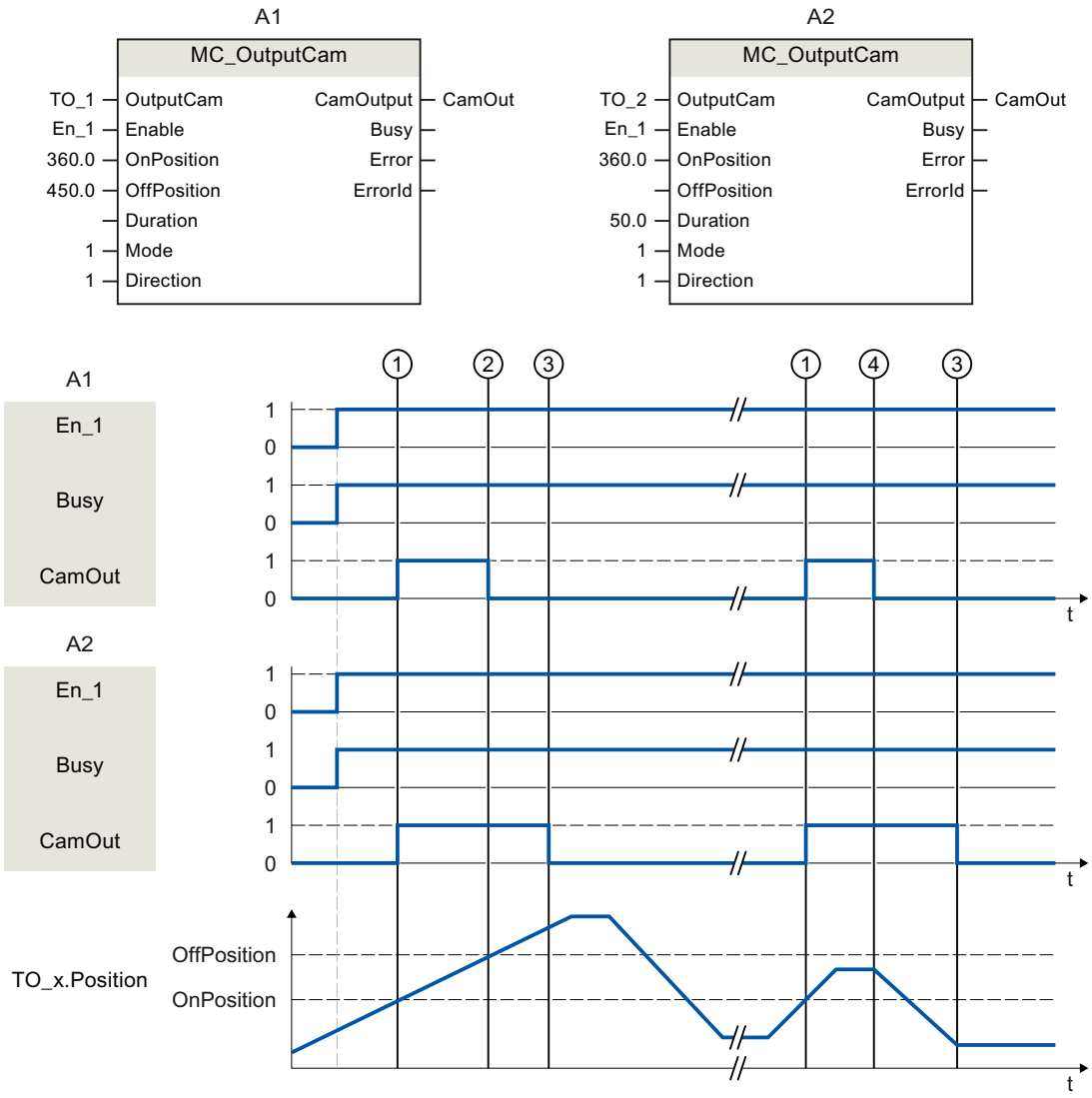

A1 Wegnocken

A2 Zeitnocken

Ein Nocken wird mit "Enable" = TRUE aktiviert. Der Nocken wird entsprechend den Parametern ausgegeben.

Die Bearbeitung des Nockens wird mit "Busy" = TRUE angezeigt.

Bei eingestellter Wirkrichtung "Direction" = 1 (positiv) verhält sich der Nocken abhängig vom eingestellten Nockentyp wie folgt:

- Der Wegnocken schaltet ein bei "OnPosition" ① und schaltet wieder aus bei "OffPosition" ②. Bei einer Richtungsumkehr wird der Nocken ausgeschaltet ④.
- Der Zeitnocken schaltet ein bei "OnPosition" ① und schaltet wieder aus nach Ablauf der eingestellten Zeit "Duration" ③. Ein eingeschalteter Zeitnocken bleibt für die eingestellte Einschaltdauer "Duration" aktiv, auch wenn die Anfangsposition wieder in umgekehrter Richtung überfahren wird.

Über "CamOutput" wird der Schaltzustand des Nockens ausgegeben.

Mit "Enable" = FALSE wird der eingestellte Nockentyp deaktiviert. "Busy" = FALSE zeigt, dass der Nocken nicht bearbeitet wird.

MC_CamTrack V4

MC_CamTrack: Nockenspur aktivieren/deaktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_CamTrack" wird die Bearbeitung einer Nockenspur freigegeben.

Anwendbar auf

- Nockenspur

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das übergeordnete Technologieobjekt muss eine gültige Position haben.
- Sollwertnocken werden bei einer Achse im nicht lagegeregelten Betrieb nicht ausgegeben.

Ablöseverhalten

- Der "MC_CamTrack" wird abgebrochen durch das Sperren der Nockenspur mit "MC_CamTrack.Enable" = FALSE.
- Im Anwenderprogramm kann nur eine Instanz des Funktionsbausteins "MC_CamTrack" auf einer Nockenspur aktiv sein. Eine zweite Instanz des Funktionsbausteins "MC_CamTrack" auf eine bereits aktive Nockenspur wird mit Fehler abgelehnt.
- Bei Änderung der Parameter der Motion Control-Anweisung "MC_CamTrack" oder am Technologie-Datenbaustein wird die Nockenspur neu berechnet. Die Nockenspurbearbeitung wird entsprechend aller Parametervorgaben bearbeitet.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_CamTrack":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CamTrack	INPUT	TO_CamTrack	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Die Nockenspur wird bearbeitet.
				FALSE	Die Nockenspur wird gesperrt.
Mode	INPUT	INT	1	0	Bei "Enable" = TRUE: Die Nockenspurbearbeitung wird sofort aktiviert. Die Nockenspurdaten sind sofort wirksam. Bereits angesteuerte Wegnocken werden abgebrochen, wenn deren Spursignal durch die geänderten Nockenspurdaten nicht weiterhin angesteuert wird. Bereits angesteuerte Zeitnocken werden in jedem Fall abgebrochen.
					Bei "Enable" = FALSE: Die Nockenspurbearbeitung wird sofort beendet. Wegnocken/Zeitnocken werden unmittelbar abgebrochen.
				1	Bei "Enable" = TRUE: Die Nockenspurbearbeitung wird sofort/mit nächstem Spurzyklus aktiviert: • Beim erstmaligen Aktivieren einer Nockenspur beginnt die Nockenspurbearbeitung sofort. • Bei bereits aktiver Nockenspurbearbeitung wird die aktuelle Nockenspur bis zum Ende des Spurzyklus ausgegeben. Anschließend werden die neuen Nockenspurdaten wirksam.
					Bei "Enable" = FALSE: Die Nockenspurbearbeitung wird am Nockenspurende beendet.
				2	Bei "Enable" = TRUE: Der Nockenspurausgang wird sofort eingeschaltet und bleibt eingeschaltet.
					Bei "Enable" = FALSE: Der Nockenspurausgang wird sofort ausgeschaltet.
InvertOutput	INPUT	BOOL	FALSE	Invertierte Ausgabe	
				TRUE	Der Spurausgang wird invertiert ausgegeben.
				FALSE	Der Spurausgang wird nicht invertiert ausgegeben.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
TrackOutput	OUTPUT	BOOL	-	Zeigt den Schaltzustand der Nockenspur an.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Die Bearbeitung der Nockenspur ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_CamTrack: Funktionsdiagramm V4

Funktionsdiagramm: Nockenspur aktivieren

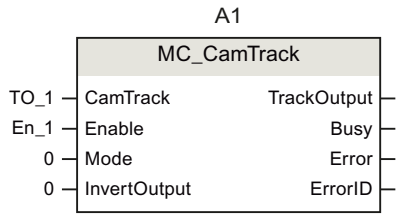

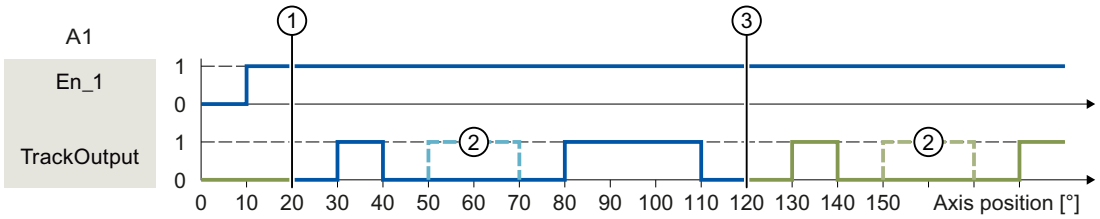

① Achsbezugsposition
② Als nicht gültig konfigurierter Nocken
③ Beginn zyklisch fortgesetzte Nockenspur

Eine Nockenspur wird mit "Enable" = TRUE aktiviert. Die Nockenspur wird entsprechend den im Technologie-Datenbaustein eingestellten Parametern ausgegeben:

Variable		Wert	Beschreibung
<TO>Parameter.			
	CamTrackType	0	Festgelegter Nockentyp für die Nockenspur ist Wegnocken
	ReferencePosition	20.0	Festgelegte Achsbezugsposition für die Nockenspur
	CamTrackLength	100.0	Festgelegte Länge der Nockenspur
	CamTrack[1].Existent	TRUE	Der erste Wegnocken der Nockenspur wird als gültig festgelegt
	CamTrack[1].OnPosition	10.0	Anfangsposition für den ersten Nocken der Nockenspur
	CamTrack[1].OffPosition	20.0	Endposition für den ersten Nocken der Nockenspur
	CamTrack[2].Existent	FALSE	Der zweite Wegnocken der Nockenspur wird als nicht gültig festgelegt
	CamTrack[2].OnPosition	30.0	Anfangsposition für den zweiten Nocken der Nockenspur
	CamTrack[2].OffPosition	50.0	Endposition für den zweiten Nocken der Nockenspur
	CamTrack[3].Existent	TRUE	Der dritte Wegnocken der Nockenspur wird als gültig festgelegt
	CamTrack[3].OnPosition	60.0	Anfangsposition für den dritten Nocken der Nockenspur
	CamTrack[3].OffPosition	90.0	Endposition für den dritten Nocken der Nockenspur

Synchrone Bewegung

MC_GearIn V4

MC_GearIn: Getriebegleichlauf starten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GearIn" starten Sie einen Getriebegleichlauf zwischen einer Leitachse und einer Folgeachse.

Mit den Parametern "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Folgeachse beim Aufsynchronisieren.

Dauer und Wegstrecke des Aufsynchronisierens sind von folgenden Parametern abhängig:

- Startzeitpunkt des "MC_GearIn"-Auftrags
- Dynamik der Folgeachse zum Startzeitpunkt
- Dynamikvorgaben für das Aufsynchronisieren
- Dynamik der Leitachse

Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen an den Parametern "RatioNumerator" und "RatioDenominator" vor (Zähler/Nenner).

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- Positiver Getriebefaktor:
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- Negativer Getriebefaktor:
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, eine Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_GearIn"-Aufträge ist im Kapitel "Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)" beschrieben.

Das Sperren der Folgeachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf in jedem Status ab.

Das Sperren der Leitachse mit "MC_Power" bricht den Gleichlauf hingegen nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GearIn":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse	
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
RatioNumerator	INPUT	DINT	1	Getriebefaktor Zähler Zulässige ganzzahlige Werte: -2147483648 bis 2147483647 (Wert 0 nicht zulässig)	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
RatioDenominator	INPUT	DINT	1	Getriebefaktor Nenner Zulässige ganzzahlige Werte: 1 bis 2147483647	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
InGear	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_GearIn" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse und den Getriebefaktor an.
3. Starten Sie den "MC_GearIn"-Auftrag mit einer steigenden Flanke am Parameter "Execute". Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InGear" den Wert "TRUE" zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Die Parameter "InGear" und "Busy" zeigen so lange den Wert "TRUE", bis der "MC_GearIn"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Siehe auch

Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)

MC_GearIn: Funktionsdiagramm V4

Funktionsdiagramm: Aufsynchronisieren und Umschalten des Leitwerts

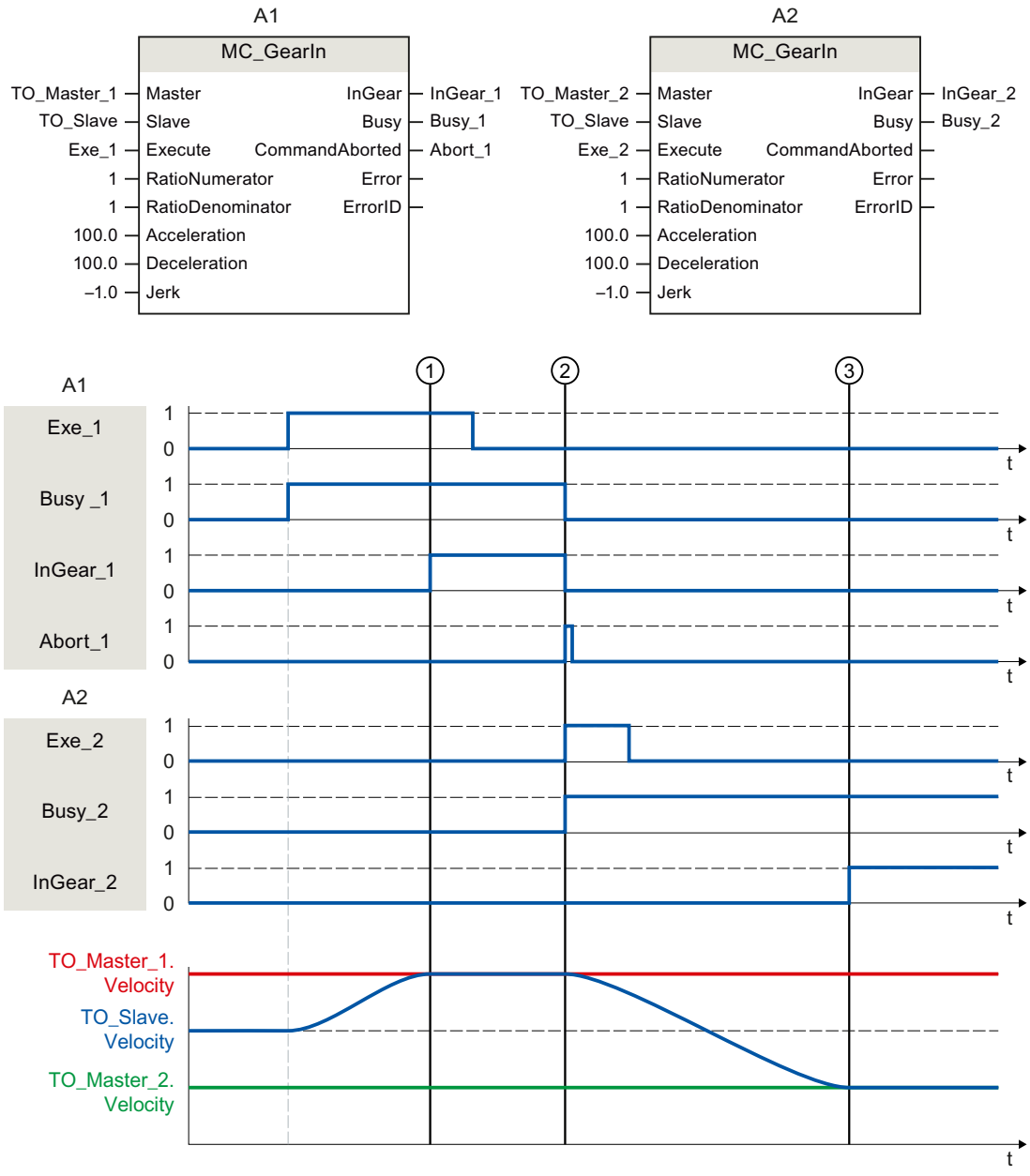

Über "Exe_1" wird ein "MC_GearIn"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird auf die Leitachse (TO_Master_1) aufsynchronisiert. Zum Zeitpunkt ① wird über "InGear_1" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

Zum Zeitpunkt ② wird der Gleichlauf durch einen weiteren "MC_GearIn"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Folgeachse wird auf die neue Leitachse (TO_Master_2) aufsynchronisiert. Zum Zeitpunkt ③ wird über "InGear_2" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

MC_GearInPos V4

MC_GearInPos: Getriebegleichlauf mit vorgegebenen Synchronpositionen starten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GearInPos" starten Sie einen Getriebegleichlauf zwischen einer Leitachse und einer Folgeachse. Der Gleichlauf wird vorlaufend zu den vorgegebenen Synchronpositionen für die Leit- und Folgeachse aufsynchronisiert.

Folgende Arten des Aufsynchronisierens sind möglich:

- Aufsynchronisieren über Dynamikparameter ("SyncProfileReference" = 0)
- Aufsynchronisieren über Leitwertweg ("SyncProfileReference" = 1)

Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen an den Parametern "RatioNumerator" und "RatioDenominator" vor (Zähler/Nenner).

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- Positiver Getriebefaktor:
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- Negativer Getriebefaktor:
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.
- Bei Verwendung der "MasterStartDistance" muss die Leitachse beim Start des Auftrags mindestens um die angegebene Distanz von der "MasterSyncPosition" entfernt sein.

Ablöseverhalten

Das Ablöseverhalten für "MC_GearInPos"-Aufträge ist im Kapitel "Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)" beschrieben.

Das Sperren der Folgeachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf in jedem Status ab.

Das Sperren der Leitachse mit "MC_Power" bricht den Gleichlauf hingegen nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GearInPos":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse	
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
RatioNumerator	INPUT	DINT	1	Getriebefaktor Zähler Zulässige ganzzahlige Werte: -2147483648 bis 2147483647 (Wert 0 nicht zulässig)	
RatioDenominator	INPUT	DINT	1	Getriebefaktor Nenner Zulässige ganzzahlige Werte: 1 bis 2147483647	
MasterSyncPosition	INPUT	LREAL	0.0	Synchronposition Leitachse Position der Leitachse, ab der Leit- und Folgeachse synchron verfahren	
SlaveSyncPosition	INPUT	LREAL	0.0	Synchronposition Folgeachse Position der Folgeachse, ab der Leit- und Folgeachse synchron verfahren	
SyncProfileReference	INPUT	DINT	1	Art des Aufsynchronisierens	
				0	Aufsynchronisieren über Dynamikparameter
				1	Aufsynchronisieren über Leitwertweg
MasterStartDistance	INPUT	LREAL	1.0	Leitwertweg (bei "SyncProfileReference" = 1)	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck (bei "SyncProfileReference" = 0)	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
SyncDirection	INPUT	DINT	3	Richtung des Aufsynchronisierens (Wirksam bei Achsen mit aktivierter Modulo-Einstellung.)	
				1	Positive Richtung Beim Aufsynchronisieren darf die Folgeachse nur in positive Richtung fahren.
				2	Negative Richtung Beim Aufsynchronisieren darf die Folgeachse nur in negative Richtung fahren.
				3	Kürzester Weg Beim Aufsynchronisieren sind Richtungsänderungen der Folgeachse erlaubt.
StartSync	OUTPUT	BOOL	FALSE	TRUE	Die Folgeachse wird auf die Leitachse aufsynchronisiert.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
InSync	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_GearInPos" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse, den Getriebefaktor und die Synchronpositionen an.
3. Starten Sie den "MC_GearInPos"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InSync" den Wert "TRUE" zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Die Parameter "InSync" und "Busy" zeigen so lange den Wert "TRUE", bis der "MC_GearInPos"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Siehe auch

Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)

MC_GearInPos: Funktionsdiagramm V4

Funktionsdiagramm: Aufsynchronisieren über Dynamikparameter/Leitwertweg

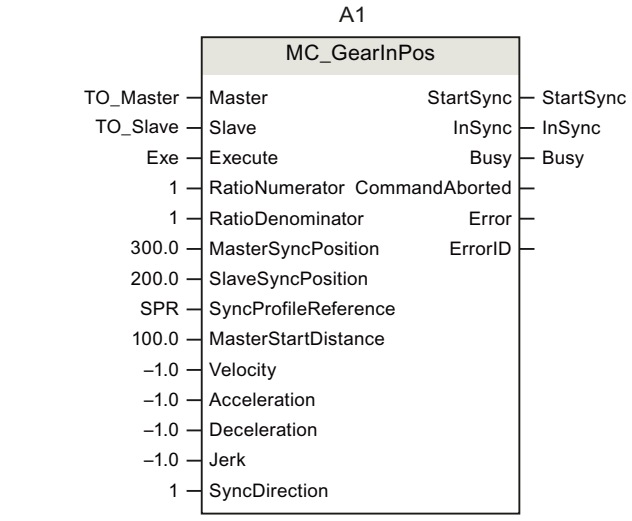

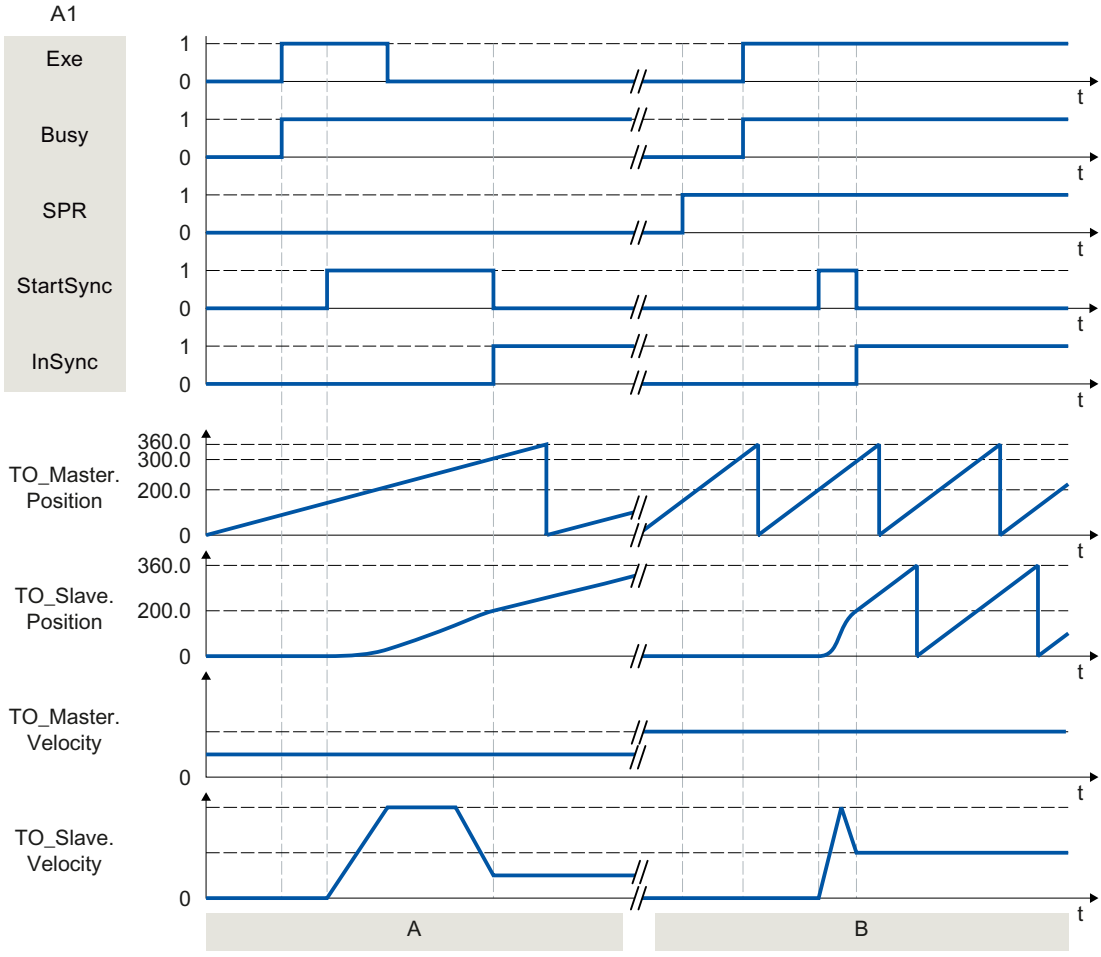

Abschnitt A	Über "Exe" wird ein "MC_GearInPos"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird über die vorgegebenen Dynamikparameter auf die Leitachse (TO_Master) aufsynchronisiert. Der zum Aufsynchronisieren benötigte Weg wird vom System berechnet. Über "StartSync" wird der Start des Aufsynchronisierens angezeigt. Beim Erreichen der vorgegebenen Synchronpositionen wird über "InSync" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.
Abschnitt B	Über "Exe" wird ein "MC_GearInPos"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird über den vorgegebenen Leitwertweg auf die Leitachse (TO_Master) aufsynchronisiert. Die zum Aufsynchronisieren benötigte Dynamik wird vom System berechnet. Über "StartSync" wird der Start des Aufsynchronisierens angezeigt. Über "InSync" wird gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

MC_PhasingAbsolute V4

MC_PhasingAbsolute: Leitwert an der Folgeachse absolut verschieben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_PhasingAbsolute" verschieben Sie den Leitwert an einer Folgeachse im Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos" absolut. Die Position der Leitachse wird dadurch nicht beeinflusst.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung der Folgeachse. Die Dynamikwerte werden zu den Werten der Gleichlaufbewegung addiert.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist über die Motion Control-Anweisung "MC_GearIn" oder "MC_GearInPos" auf die Leitachse aufsynchronisiert ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE).
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_PhasingAbsolute"-Aufträge ist im Kapitel "Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)" beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht die Leitwertverschiebung nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_PhasingAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse	
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
PhaseShift	INPUT	LREAL	0.0	Absolute Leitwertverschiebung	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Done	OUTPUT	BOOL	FALSE	TRUE	Leitwertverschiebung ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
AbsolutePhaseShift	OUTPUT	LREAL	0.0	So lange "Busy" = TRUE: Anzeige der bisher ausgefahrenen Leitwertverschiebung	

Absolute Leitwertverschiebung starten

Um mit der Motion Control-Anweisung "MC_PhasingAbsolute" eine absolute Leitwertverschiebung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "PhaseShift" die absolute Leitwertverschiebung an.
3. Starten Sie den "MC_PhasingAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "AbsolutePhaseShift", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)

MC_GearIn V4 (Seite 3294)

MC_GearInPos V4 (Seite 3299)

MC_PhasingAbsolute: Funktionsdiagramm V4

Funktionsdiagramm: Leitwert absolut verschieben

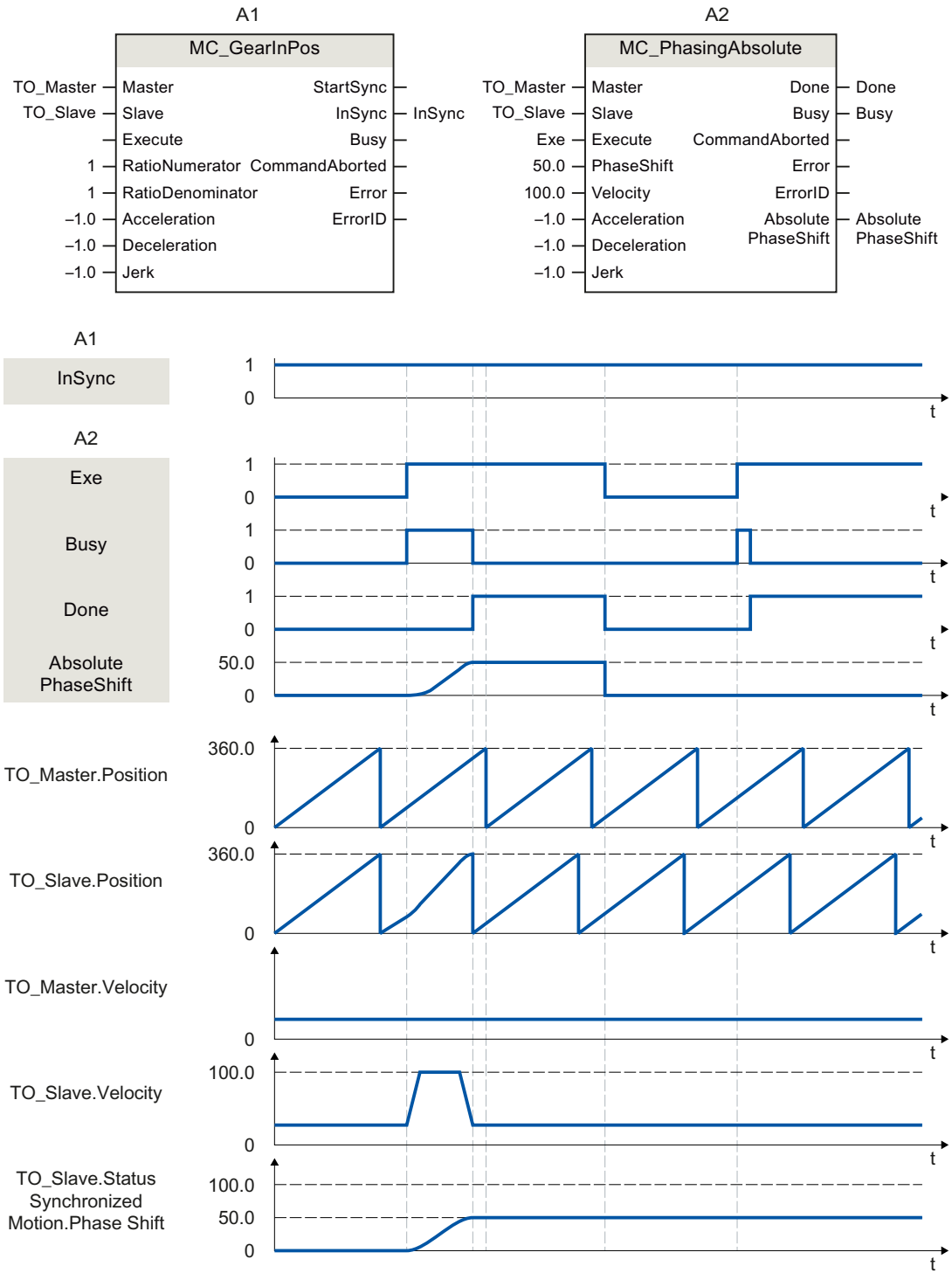

Während eines aktiven Getriebegleichlaufs mit "MC_GearInPos" (A1) wird über "Exe" ein "MC_PhasingAbsolute"-Auftrag (A2) angestoßen. Die Leitwertverschiebung wird mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "AbsolutePhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 50.0 angezeigt. Die Bewegung der Leitachse wird nicht beeinflusst.

Nach der Leitwertverschiebung wird der "MC_PhasingAbsolute"-Auftrag (A2) über "Exe" erneut angestoßen. Da die Leitwertverschiebung (<TO>.StatusSynchronizedMotion.PhaseShift) bereits 50.0 beträgt, wird der Leitwert nicht verschoben.

MC_PhasingRelative V4

MC_PhasingRelative: Leitwert an der Folgeachse relativ verschieben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_PhasingRelative" verschieben Sie den Leitwert an einer Folgeachse im Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos" relativ zur vorhandenen Leitwertverschiebung. Die Position der Leitachse wird dadurch nicht beeinflusst.

Mit den Parametern "Velocity", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung der Folgeachse. Die Dynamikwerte werden zu den Werten der Gleichlaufbewegung addiert.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist über die Motion Control-Anweisung "MC_GearIn" oder "MC_GearInPos" auf die Leitachse aufsynchronisiert ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE).
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_PhasingRelative"-Aufträge ist im Kapitel "Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)" beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht die Leitwertverschiebung nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_PhasingRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse	
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
PhaseShift	INPUT	LREAL	0.0	Relative Leitwertverschiebung	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung)	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Done	OUTPUT	BOOL	FALSE	TRUE	Leitwertverschiebung ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
CoveredPhaseShift	OUTPUT	LREAL	0.0	Solange "Busy" = TRUE: Anzeige der bisher ausgefahrenen Leitwertverschiebung	

Relative Leitwertverschiebung starten

Um mit der Motion Control-Anweisung "MC_PhasingRelative" eine relative Leitwertverschiebung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "PhaseShift" die relative Leitwertverschiebung an.
3. Starten Sie den "MC_PhasingRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   An den Parametern "Busy", "CoveredPhaseShift", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)

MC_GearIn V4 (Seite 3294)

MC_GearInPos V4 (Seite 3299)

MC_PhasingRelative: Funktionsdiagramm V4

Funktionsdiagramm: Leitwert relativ verschieben

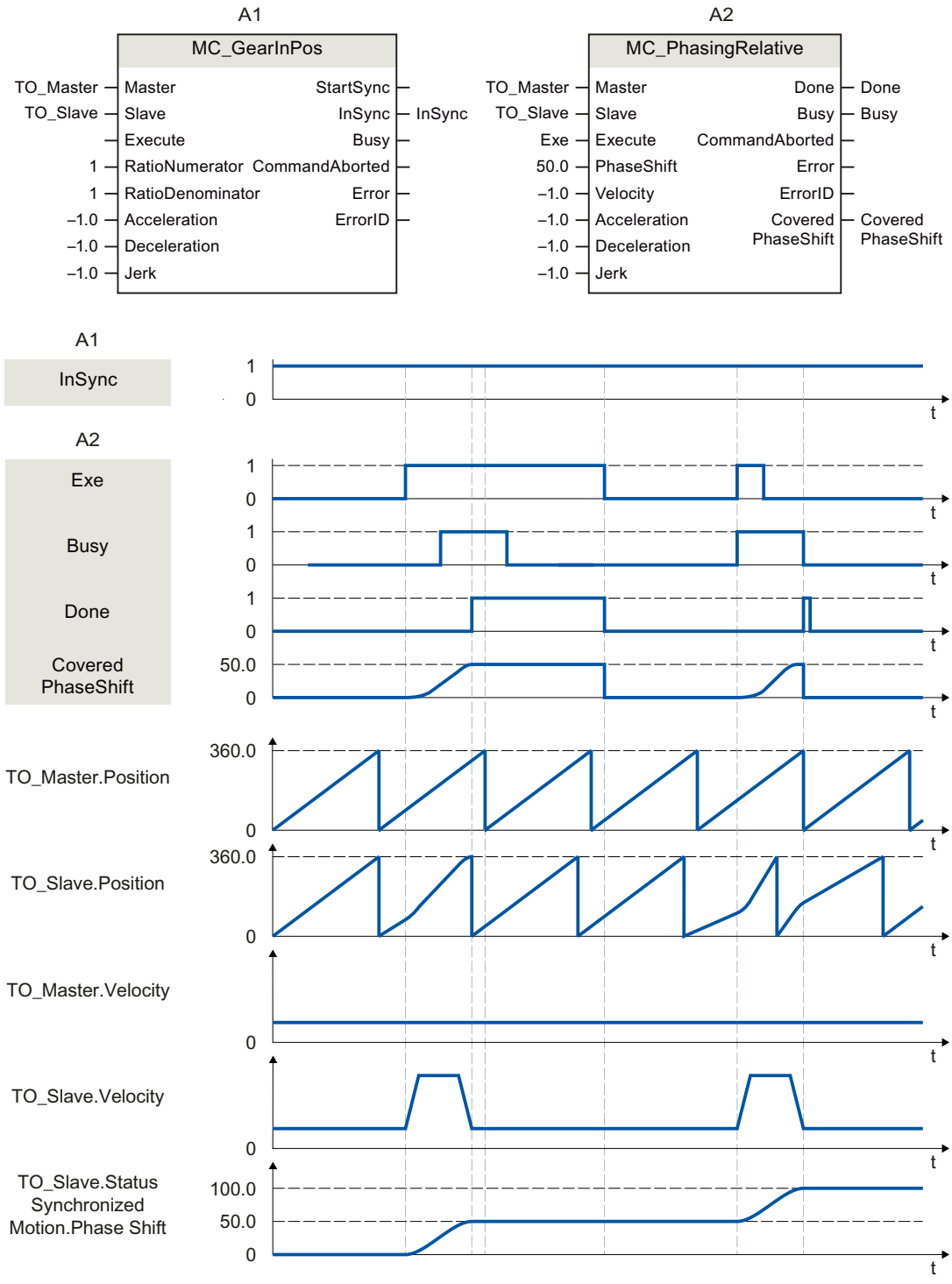

Während eines aktiven Getriebegleichlaufs mit "MC_GearInPos" (A1) wird über "Exe" ein "MC_PhasingRelative"-Auftrag (A2) angestoßen. Die Leitwertverschiebung wird mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "CoveredPhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 50.0 angezeigt. Die Bewegung der Leitachse wird nicht beeinflusst.

Nach der Leitwertverschiebung wird der "MC_PhasingRelative"-Auftrag (A2) über "Exe" erneut angestoßen. Die Leitwertverschiebung wird erneut mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "CoveredPhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 100.0 angezeigt.

MC_CamIn V4

MC_CamIn: Kurvenscheibengleichlauf starten V4

Beschreibung

Mit der Motion Control-Anweisung "MC_CamIn" starten Sie einen Kurvenscheibengleichlauf zwischen einer Leitachse und einer Folgeachse.

Mit dem Parameter "SyncProfileReference" geben Sie das Synchronisierprofil vor:

- **"SyncProfileReference" = 0 oder 1**
  Der Gleichlauf wird vorlaufend zur vorgegebenen Synchronposition der Leitachse aufsynchronisiert.
  Die Synchronposition der Leitachse, ab der Leit- und Folgeachse synchron verfahren, ergibt sich aus der Anfangsposition der Kurvenscheibe sowie den Parametern "MasterSyncPosition" und "MasterOffset".
  Mit dem Parameter "MasterOffset" verschieben Sie die Leitwerte der Kurvenscheibe (bei "SyncProfileReference" = 0 oder 1). Damit legen Sie die Lage der Kurvenscheibe bezogen auf den Leitwert der Gleichlauffunktion fest. So verschieben die Kurvenscheibe in den benötigten Positionsbereich.
- **"SyncProfileReference" = 2**
  Der Gleichlauf wird sofort synchron gesetzt. Über den Parameter "MasterSyncPosition" legen Sie die exakte Synchronposition in der Kurvenscheibe fest. Diese Einstellung eignet sich vor allem für das Synchronisieren im Stillstand.

Eine Kurvenscheibe ist nach der Interpolation zwischen der Anfangsposition (<TO>.StatusCam.StartLeadingValue) und der Endposition (<TO>.StatusCam.EndLeadingValue) definiert. Die Angaben zum Leit- und Folgewertebereich in der Konfiguration des Technologieobjekts wirken sich nur auf die Anzeige im grafischen Editor aus.

Mit dem Parameter "MasterSyncPosition" geben Sie die Synchronposition in der Kurvenscheibe bezogen auf die Anfangsposition der Kurvenscheibe vor. Mit Erreichen der Synchronposition wird das Aufsynchronisieren abgeschlossen. Um die gesamte Kurvenscheibe zu durchfahren, geben Sie in "MasterSyncPosition" den Wert 0.0

(Standardwert) an. Mit "MasterSyncPosition" ≠ 0.0 verschieben Sie die Synchronposition innerhalb der Kurvenscheibe, ohne die Lage der Kurvenscheibe zu verändern.

Folgendes Bild zeigt prinzipiell die Wirkung der Leit- und Folgewertverschiebung sowie die Definition der Kurvenscheibe bei folgenden Parameterwerten:

- Anfangsposition der Kurvenscheibe > 0
- "MasterSyncPosition" > 0

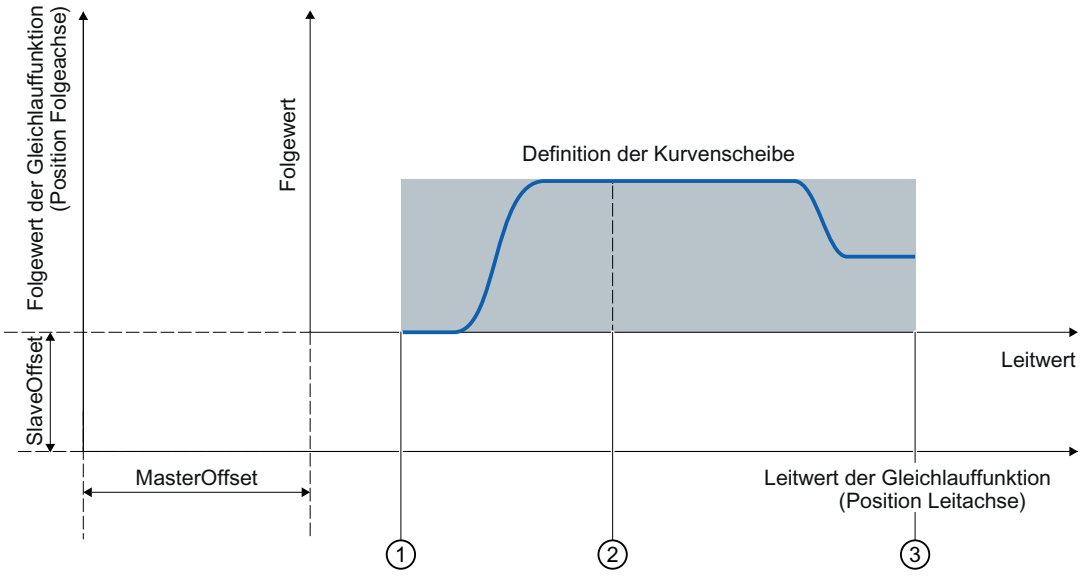

① Anfangsposition der Kurvenscheibe

Erster definierter Stützpunkt/Beginn des ersten Segments der Kurvenscheibe (<TO>.StatusCam.StartLeadingValue)

② Abschluss des Aufsynchronisierens ("MasterSyncPosition")

③ Endposition der Kurvenscheibe

Letzter definierter Stützpunkt/Ende des letzten Segments der Kurvenscheibe (<TO>.StatusCam.EndLeadingValue)

Folgendes Bild zeigt prinzipiell die Wirkung der Skalierung der Kurvenscheibe mit den Parametern "MasterScaling" und "SlaveScaling":

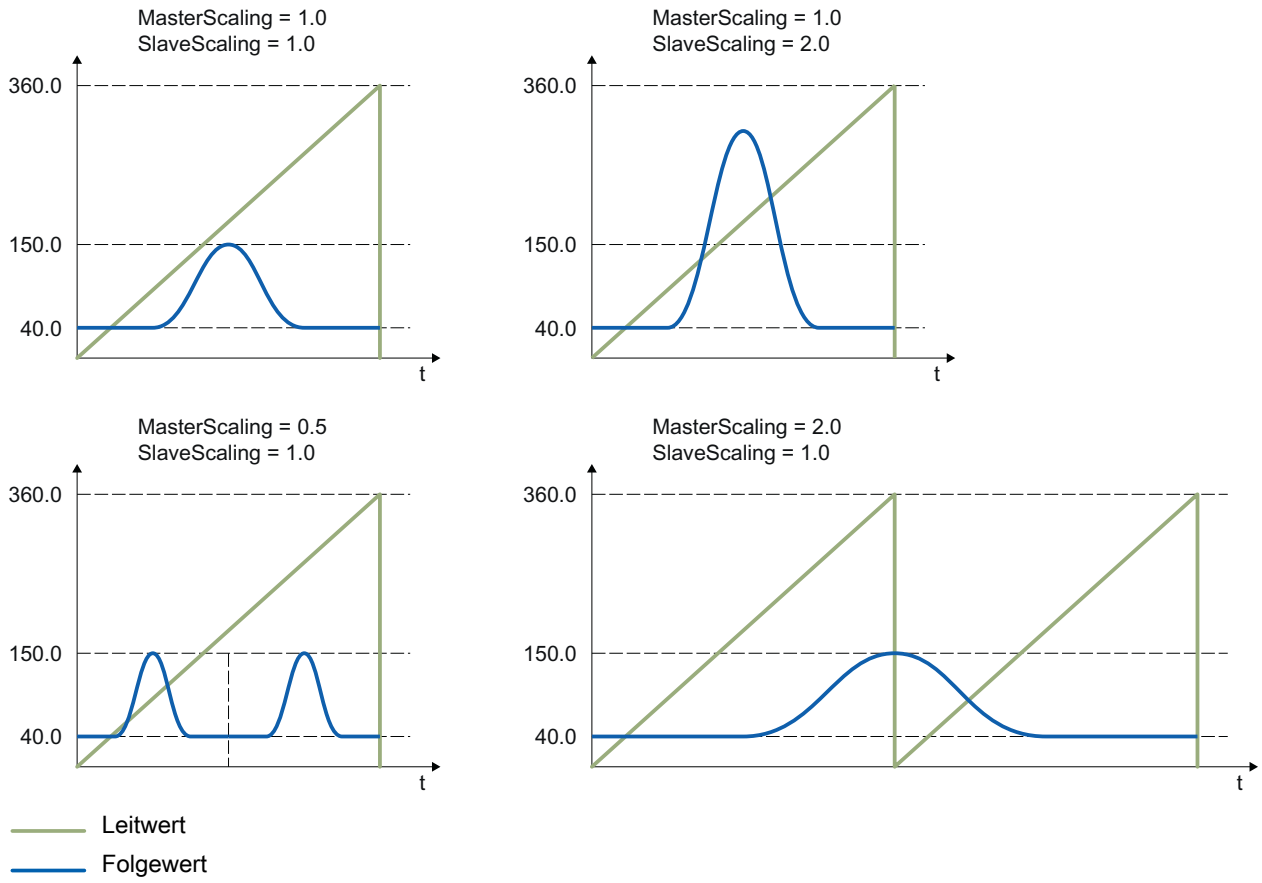

Folgende Arten des Aufsynchronisierens sind möglich:

- Aufsynchronisieren über Dynamikparameter ("SyncProfileReference" = 0)
- Aufsynchronisieren über Leitwertweg ("SyncProfileReference" = 1)
- Direkt synchron setzen ("SyncProfileReference" = 2)

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse oder Folgeachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse, der Folgeachse und der Kurvenscheibe wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.

- Die Folgeachse ist freigegeben.
- Die Kurvenscheibe ist mit "MC_InterpolateCam" interpoliert.
- Bei Verwendung der "MasterStartDistance" muss die Leitachse beim Start des Auftrags mindestens um die angegebene Distanz von der "MasterSyncPosition" entfernt sein.

Ablöseverhalten

Das Ablöseverhalten für "MC_CamIn"-Aufträge ist im Kapitel "Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)" beschrieben.

Das Sperren der Folgeachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf in jedem Status ab.

Das Sperren der Leitachse mit "MC_Power" bricht den Gleichlauf hingegen nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_CamIn":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe
Execute	INPUT	BOOL	FALSE	TRUE: Start des Auftrags mit steigender Flanke
MasterOffset	INPUT	LREAL	0.0	Verschiebung der Leitwerte der Kurvenscheibe (bei "SyncProfileReference" = 0 oder 1) Das Technologieobjekt Kurvenscheibe wird nicht verändert.
SlaveOffset	INPUT	LREAL	0.0	Verschiebung der Folgewerte der Kurvenscheibe (bei "SyncProfileReference" = 0 oder 1) Das Technologieobjekt Kurvenscheibe wird nicht verändert.
MasterScaling	INPUT	LREAL	1.0	Skalierung der Leitwerte der Kurvenscheibe Das Technologieobjekt Kurvenscheibe wird nicht verändert.
SlaveScaling	INPUT	LREAL	1.0	Skalierung der Folgewerte der Kurvenscheibe Das Technologieobjekt Kurvenscheibe wird nicht verändert.
MasterSyncPosition	INPUT	LREAL	0.0	Synchronposition Leitwertseitige Abschlussposition des Aufsynchronisierens bezogen auf die Anfangsposition der Kurvenscheibe (<TO>.StatusCam.StartLeadingValue) Der Wert muss innerhalb der Definition der Kurvenscheibe liegen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
SyncProfileReference	INPUT	DINT	1	Synchronisierprofil	
				0	Aufsynchronisieren über Dynamikparameter
				1	Aufsynchronisieren über Leitwertweg
				2	Direkt synchron setzen
MasterStartDistance	INPUT	LREAL	0.0	Leitwertweg (Weg der Leitachse während des Aufsynchronisierens) (bei "SyncProfileReference" = 1)	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung (bei "SyncProfileReference" = 0)	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck (bei "SyncProfileReference" = 0)	
				> 0.0	Beschleunigungsstetiges Geschwindigkeitsprofil Der angegebene Wert wird verwendet.
				= 0.0	Trapezförmiges Geschwindigkeitsprofil
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
ApplicationMode	INPUT	DINT	0	Anwendung der Kurvenscheibe	
				0	Einmalig/nicht zyklisch
				1	Zyklisch (folgewertseitig absolut)
				2	Zyklisch anhängend (folgewertseitig stetig anhängend)
SyncDirection	INPUT	DINT	3	Richtung des Aufsynchronisierens (wirksam bei Achsen mit aktivierter Modulo-Einstellung)	
				1	Positive Richtung Beim Aufsynchronisieren darf die Folgeachse nur in positive Richtung fahren.
				2	Negative Richtung Beim Aufsynchronisieren darf die Folgeachse nur in negative Richtung fahren.
				3	Kürzester Weg Beim Aufsynchronisieren sind Richtungsänderungen der Folgeachse erlaubt.
StartSync	OUTPUT	BOOL	FALSE	TRUE	Die Folgeachse wird auf die Leitachse aufsynchronisiert.
InSync	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
EndOfProfile	OUTPUT	BOOL	FALSE	TRUE	Das Ende der Kurvenscheibe wurde erreicht. Wird bei zyklischer Verwendung der Kurvenscheibe für mindestens einen Aufruf von "MC_CamIn" im Anwenderprogramm angezeigt.

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_CamIn" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse, die zu verwendende Kurvenscheibe und die Synchronposition an.
3. Starten Sie den "MC_CamIn"-Auftrag mit einer steigenden Flanke am Parameter "Execute". Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InSync" den Wert "TRUE" zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Bei zyklischer Anwendung der Kurvenscheibe zeigen "InSync" und "Busy" so lange den Wert "TRUE", bis der "MC_CamIn"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird. Bei nichtzyklischer Anwendung der Kurvenscheibe werden "InSync" und "Busy" auf den Wert "FALSE" gesetzt, wenn der Parameter "EndOfProfile" auf den Wert "TRUE" gesetzt wird.

Siehe auch

Ablöseverhalten V4: Gleichlaufaufträge (Seite 3395)

MC_CamIn: Funktionsdiagramm V4

Funktionsdiagramm: Aufsynchronisieren und Umschalten der Kurvenscheibe

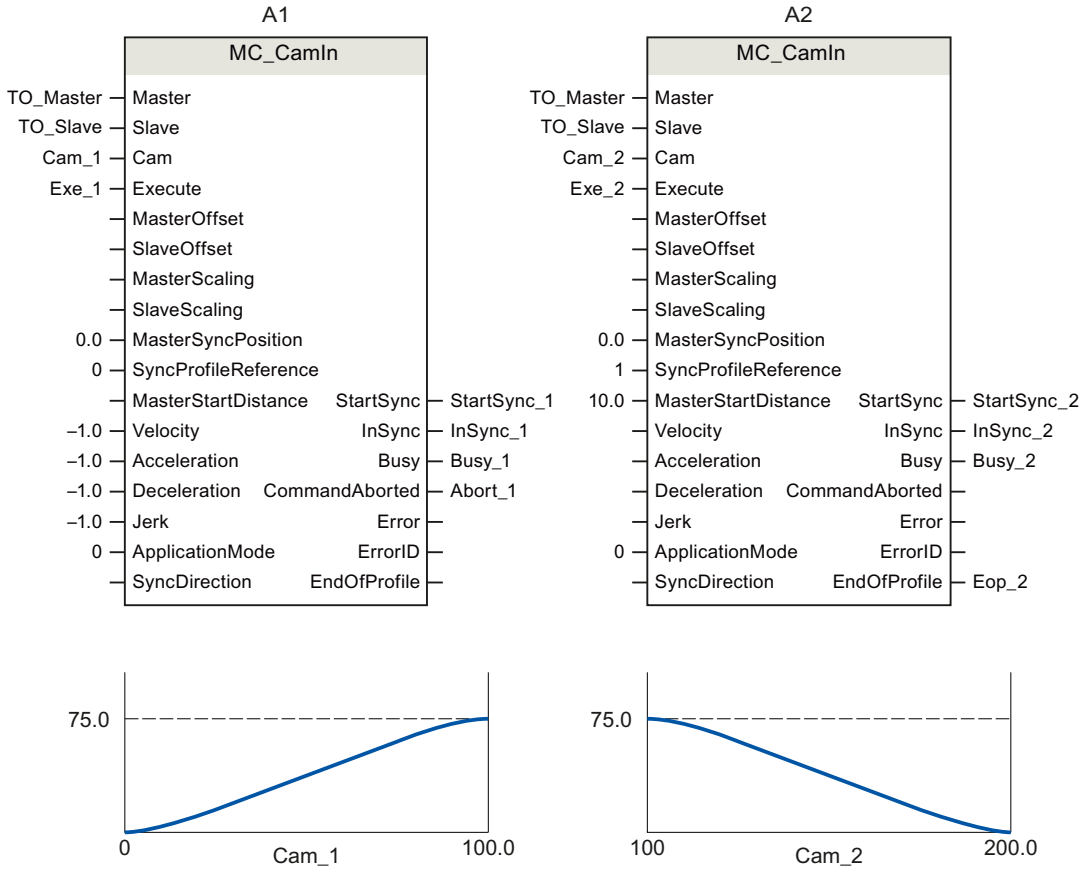

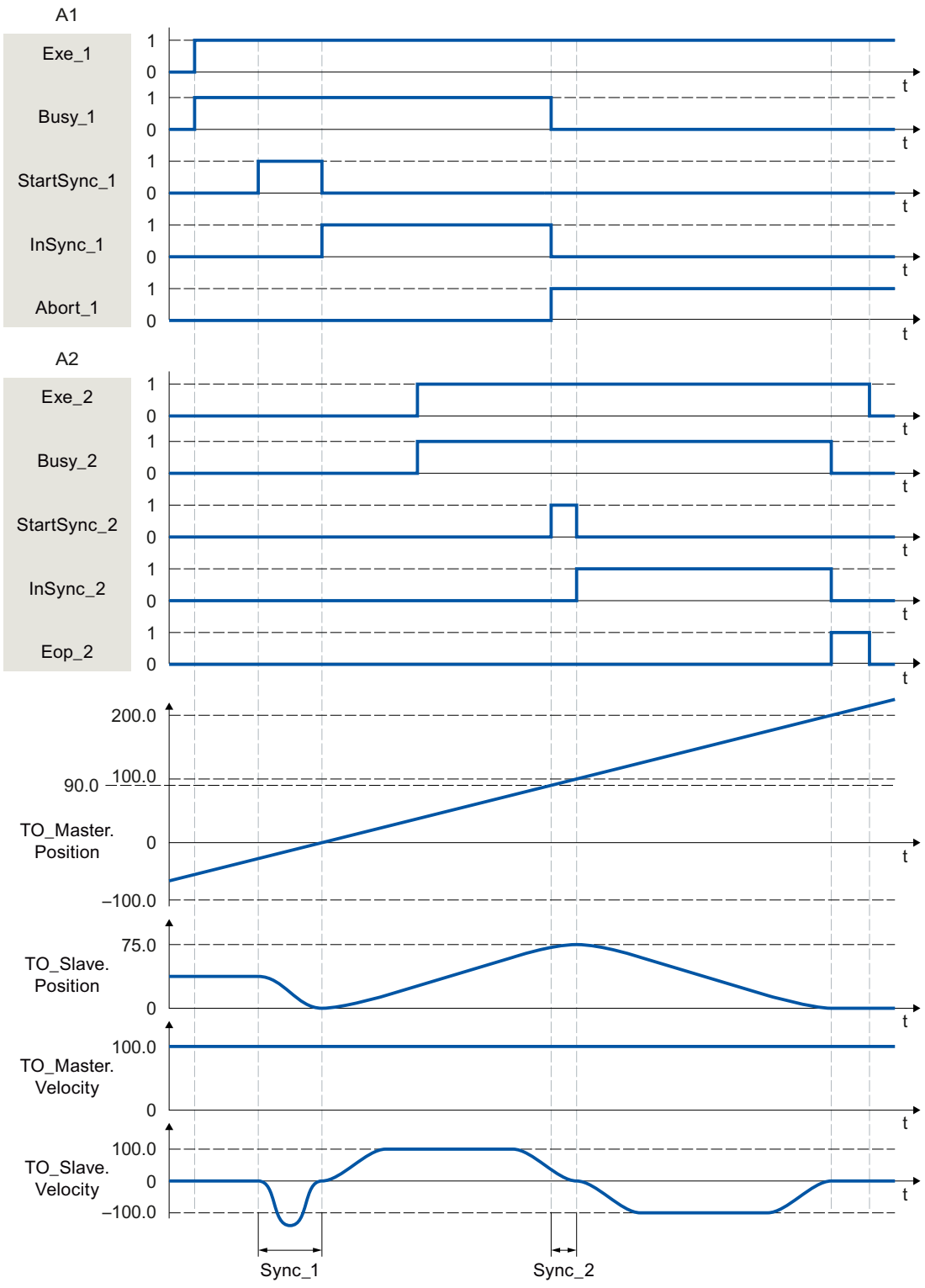

Über "Exe_1" wird ein "MC_CamIn"-Auftrag (A1) angestoßen. Über "StartSync_1" wird der Start des Aufsynchronisierens angezeigt. Die Folgeachse (TO_Slave) wird im Bereich "Sync_1" über die vorgegebenen Dynamikparameter auf die Kurvenscheibe (Cam_1)

aufsynchronisiert. Über "InSync_1" wird gemeldet, dass die Folgeachse aufsynchronisiert ist und synchron zur Leitachse fährt.

Der Gleichlauf wird durch einen weiteren "MC_CamIn"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Über "StartSync_2" wird der Start des Aufsynchronisierens angezeigt. Die Folgeachse wird im Bereich "Sync_2" über den vorgegebenen Leitwertweg auf die neue Kurvenscheibe (Cam_2) aufsynchronisiert. Im Bereich "Sync_2" folgt die Achse nicht der Kurvenscheibe "Cam_1". Die Achse wird mit der vom System berechneten Dynamik zur Synchronposition von A2 verfahren. Über "InSync_2" gemeldet, dass die Folgeachse aufsynchronisiert ist und synchron zur Leitachse fährt.

MC_SynchronizedMotionSimulation V4

MC_SynchronizedMotionSimulation: Gleichlauf in Simulation setzen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SynchronizedMotionSimulation" setzen Sie einen aktiven Gleichlauf an einer Folgeachse in Simulation. Damit bleibt ein Gleichlauf beim Sperren der Folgeachse mit einem "MC_Power"-Auftrag aktiv. Die Folgeachse muss nach der erneuten Freigabe nicht neu aufsynchronisieren.

Mit dem Start eines "MC_SynchronizedMotionSimulation"-Auftrags wird die Sollgeschwindigkeit aus dem Gleichlauf auf null gesetzt. Wenn beim Start der Simulation eine überlagerte Bewegung an der Folgeachse aktiv ist, werden die Sollwerte dieser Überlagerung weiterhin ausgegeben.

Sollwerte von Bewegungsaufträgen, die während der Gleichlaufsimulation gestartet werden, werden an den Antrieb ausgegeben.

Wenn die Position der Folgeachse beim Beenden der Simulation gegenüber der Position beim Start der Simulation abweicht, führt dies zu einem Sollwertsprung.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Folgeachse ist eine Gleichlaufachse.
- Am Technologieobjekt ist ein Gleichlauf aktiv im Status "Synchron" (<TO>.StatusWord.X22 = TRUE).

Ablöseverhalten

Ein "MC_SynchronizedMotionSimulation"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen. Der simulierte Gleichlauf bleibt auch beim Sperren der Folgeachse mit "MC_Power.Enable" = FALSE aktiv.

Ein Restart des Technologieobjekts beendet die Simulation und bricht den Gleichlauf ab.

Ein neuer "MC_SynchronizedMotionSimulation"-Auftrag bricht keinen anderen Motion Control-Auftrag ab. Mit "MC_SynchronizedMotionSimulation.Enable" = TRUE werden Gleichlaufaufträge abgelehnt.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SynchronizedMotionSimulation":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Slave	INPUT	TO_Synchronous-Axis	-	Technologieobjekt der Folgeachse	
Enable	INPUT	BOOL	FALSE	TRUE	Der Gleichlauf wird in Simulation gesetzt.
				FALSE	Die Simulation des Gleichlaufs wird beendet.
InSimulation	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf in Simulation
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf beim Sperren der Folgeachse nicht beenden

Um mit der Motion Control-Anweisung "MC_SynchronizedMotionSimulation" einen Gleichlauf beim Sperren der Folgeachse nicht zu beenden, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Slave" die Folgeachse an.
3. Setzen Sie die Leitachse in den Stillstand (z. B. mit "MC_Halt").
4. Setzen Sie den Gleichlauf an der angegebenen Folgeachse mit "MC_SynchronizedMotionSimulation.Enable" = TRUE in Simulation.
5. Wenn der Parameter "InSimulation" den Wert TRUE zeigt, sperren Sie die Folgeachse. Der Gleichlauf bleibt in der Simulation aktiv.
6. Um den Gleichlauf nach der Freigabe der Folgeachse wieder aus der Simulation zurückzusetzen, beenden Sie die Gleichlaufsimulation mit "MC_SynchronizedMotionSimulation.Enable" = FALSE.
   Die Gleichlaufsimulation wird beendet. Die Folgeachse folgt der Leitachse, ohne neu aufzusynchronisieren.

Kurvenscheibe

MC_InterpolateCam V4

MC_InterpolateCam: Kurvenscheibe interpolieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_InterpolateCam" interpolieren Sie eine Kurvenscheibe.

Die Interpolation schließt die Lücken zwischen den definierten Stützpunkten und Segmenten der Kurvenscheibe. Die Kurvenscheibe wird im Definitionsbereich zwischen folgenden Werten interpoliert:

- Erster definierter Stützpunkt/Beginn des ersten Segments der Kurvenscheibe (<TO>.StatusCam.StartLeadingValue)
- Letzter definierter Stützpunkt/Ende des letzten Segments der Kurvenscheibe (<TO>.StatusCam.EndLeadingValue)

Nach der Interpolation ist jedem Wert im Definitionsbereich der Kurvenscheibe ein eindeutiger Wert im Wertebereich zugeordnet.

Die Interpolationsart bestimmt, wie fehlende Bereiche interpoliert werden. Die Interpolationsart stellen Sie in der Konfiguration des Technologieobjekts ein. Folgende Interpolationsarten sind möglich:

- Lineare Interpolation
- Interpolation mit kubischen Splines
- Interpolation mit Bézier-Splines

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe wird aktuell nicht verwendet, z. B. für einen Kurvenscheibengleichlauf.

Ablöseverhalten

- Ein "MC_InterpolateCam"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_InterpolateCam"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_InterpolateCam":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Done	OUTPUT	BOOL	FALSE	TRUE	Die Kurvenscheibe ist interpoliert.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Kurvenscheibe interpolieren

Um mit der Motion Control-Anweisung "MC_InterpolateCam" eine Kurvenscheibe zu interpolieren, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Cam" die zu interpolierende Kurvenscheibe an.
3. Starten Sie den "MC_InterpolateCam"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   Die Kurvenscheibe wird interpoliert. Wenn der Parameter "Done" den Wert "TRUE" zeigt, ist die Interpolation abgeschlossen.

MC_GetCamFollowingValue V4

MC_GetCamFollowingValue: Folgewert einer Kurvenscheibe auslesen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GetCamFollowingValue" lesen Sie aus einer Kurvenscheibe zu einem Leitwert den Folgewert sowie die erste und zweite Ableitung des Folgewerts aus.

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe ist interpoliert.

Ablöseverhalten

- Ein "MC_GetCamFollowingValue"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_GetCamFollowingValue"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GetCamFollowingValue":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
LeadingValue	INPUT	LREAL	0.0	Leitwert, zu dem der Folgewert ausgelesen wird	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Folgewert wurde ausgelesen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
Value	OUTPUT	LREAL	-	Ausgelesener Folgewert (Position) (gültig bei "Done" = TRUE)	
FirstDerivative	OUTPUT	LREAL	-	Erste Ableitung am ausgelesenen Folgewert (gültig bei "Done" = TRUE)	
SecondDerivative	OUTPUT	LREAL	-	Zweite Ableitung am ausgelesenen Folgewert (gültig bei "Done" = TRUE)	

Folgewert auslesen

Um mit der Motion Control-Anweisung "MC_GetCamFollowingValue" einen Folgewert aus einer Kurvenscheibe auszulesen, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Kurvenscheibe und den Leitwert an.
3. Starten Sie den "MC_GetCamFollowingValue"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   Wenn der Parameter "Done" den Wert "TRUE" zeigt, wurde der Folgewert ausgelesen. Der Folgewert und die Ableitungen werden an den Parametern "Value", "FirstDerivative" und "SecondDerivative" ausgegeben.

MC_GetCamLeadingValue V4

MC_GetCamLeadingValue: Leitwert einer Kurvenscheibe auslesen V4 (S7-1500T)

Beschreibung

Mit der Motion Control-Anweisung "MC_GetCamLeadingValue" lesen Sie aus einer Kurvenscheibe den Leitwert aus, der zu einem Folgewert definiert ist.

Da für verschiedene Leitwerte gleiche Folgewerte definiert sein können, lässt sich am Parameter "ApproachValue" eine Näherung des Leitwerts angeben.

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe ist interpoliert.

Ablöseverhalten

- Ein "MC_GetCamLeadingValue"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_GetCamLeadingValue"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GetCamLeadingValue":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
FollowingValue	INPUT	LREAL	0.0	Folgewert, zu dem der Leitwert ausgelesen wird	
ApproachLeadingValue	INPUT	LREAL	0.0	Näherungswert zum gesuchten Leitwert Wenn der Folgewert mehrfach in der Kurvenscheibe verwendet ist, lässt sich damit der gesuchte Leitwert eingrenzen.	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Leitwert wurde ausgelesen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
Value	OUTPUT	LREAL	-	Ausgelesener Leitwert (Position) (gültig bei "Done" = TRUE)	

Leitwert auslesen

Um mit der Motion Control-Anweisung "MC_GetCamLeadingValue" einen Leitwert aus einer Kurvenscheibe auszulesen, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Kurvenscheibe, den Folgewert und den Näherungswert zum gesuchten Leitwert an.
3. Starten Sie den "MC_GetCamLeadingValue"-Auftrag mit einer steigenden Flanke am Parameter "Execute".
   Wenn der Parameter "Done" den Wert "TRUE" zeigt, wurde der Leitwert ermittelt. Die Berechnung des Leitwerts kann mehrere Takte dauern. Der Leitwert wird am Parameter "Value" ausgegeben.

MotionIn

MC_MotionInVelocity V4

MC_MotionInVelocity: Bewegungssollwerte vorgeben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MotionInVelocity" geben Sie der Achse zyklisch applikativ berechnete Bewegungssollwerte für Geschwindigkeit und Beschleunigung als Basisbewegung vor. Dabei wird kein Geschwindigkeitsprofil berechnet, die Werte sind am Technologieobjekt direkt wirksam. Die Dynamikgrenzen sind nicht wirksam. Die Bewegungsvorgabe über MotionIn kann mit einem "MC_MoveSuperimposed"-Auftrag (Seite 3274) überlagert werden.

Mit dem Parameter "Velocity" geben Sie die Sollgeschwindigkeit und mit "Acceleration" die Sollbeschleunigung vor. Die Sollbeschleunigung wirkt nur, wenn der "MC_MotionInVelocity"-Auftrag durch einen anderen Motion Control-Auftrag abgelöst wird. Dabei wird für die Beschleunigung auf die Sollbeschleunigung des "MC_MotionInVelocity"-Auftrags aufgesetzt. Wenn die Sollbeschleunigung null ist, wird beim Ablösen auf die Beschleunigung null aufgesetzt.

Anwendbar auf

- Positionierachse
- Drehzahlachse
- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MotionInVelocity"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MotionInVelocity":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke Solange der Parameter TRUE ist, werden die angegebenen Sollwerte verwendet.
				FALSE	Ende des Auftrags mit fallender Flanke Wenn der Parameter von TRUE auf FALSE gesetzt wird, werden die Sollwerte auf 0.0 gesetzt.
Velocity	INPUT	LREAL	0.0	Sollgeschwindigkeit Beachten Sie die Dynamikgrenzen.	
Acceleration	INPUT	LREAL	0.0	Sollbeschleunigung Beachten Sie die Dynamikgrenzen.	
PositionControlled	INPUT	BOOL	TRUE	TRUE	Lagegeregelter Betrieb
				FALSE	Gesteuerter Betrieb
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MotionInVelocity: Funktionsdiagramm V4

Funktionsdiagramm: Bewegungssollwerte vorgeben

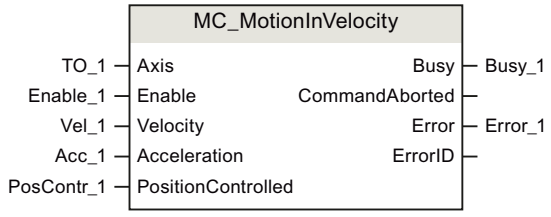

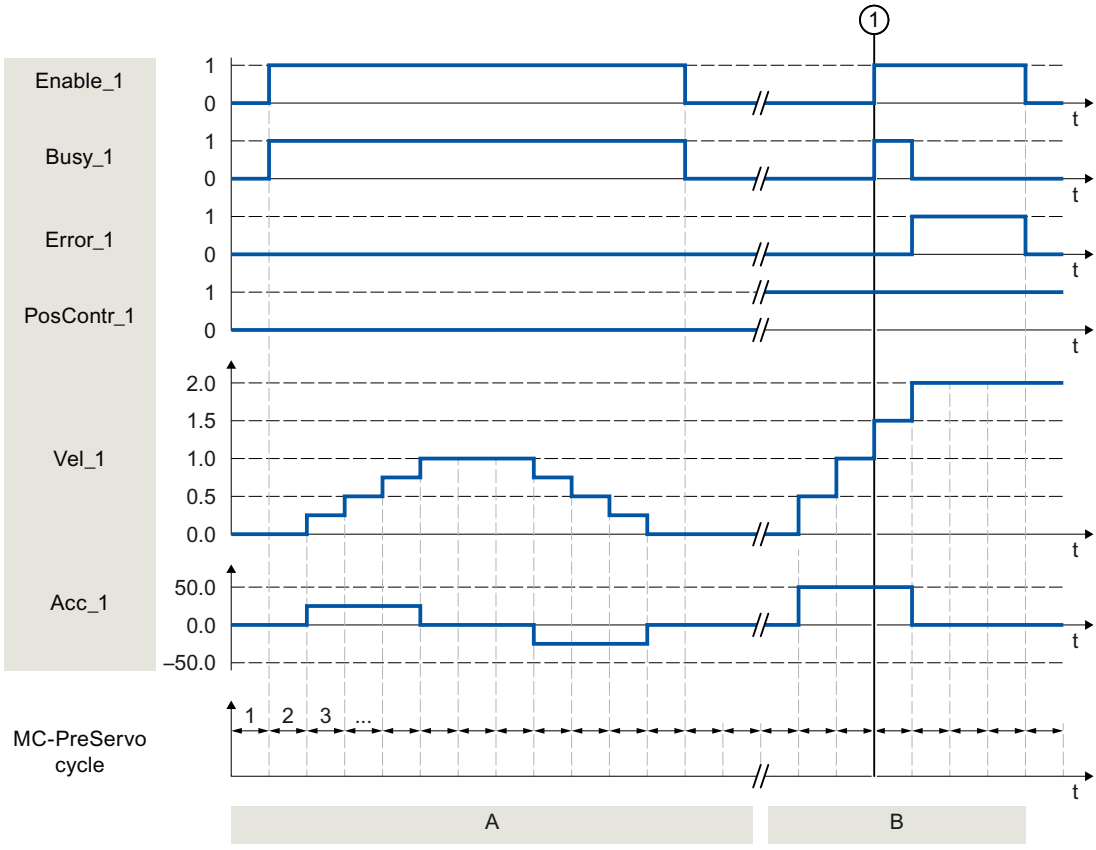

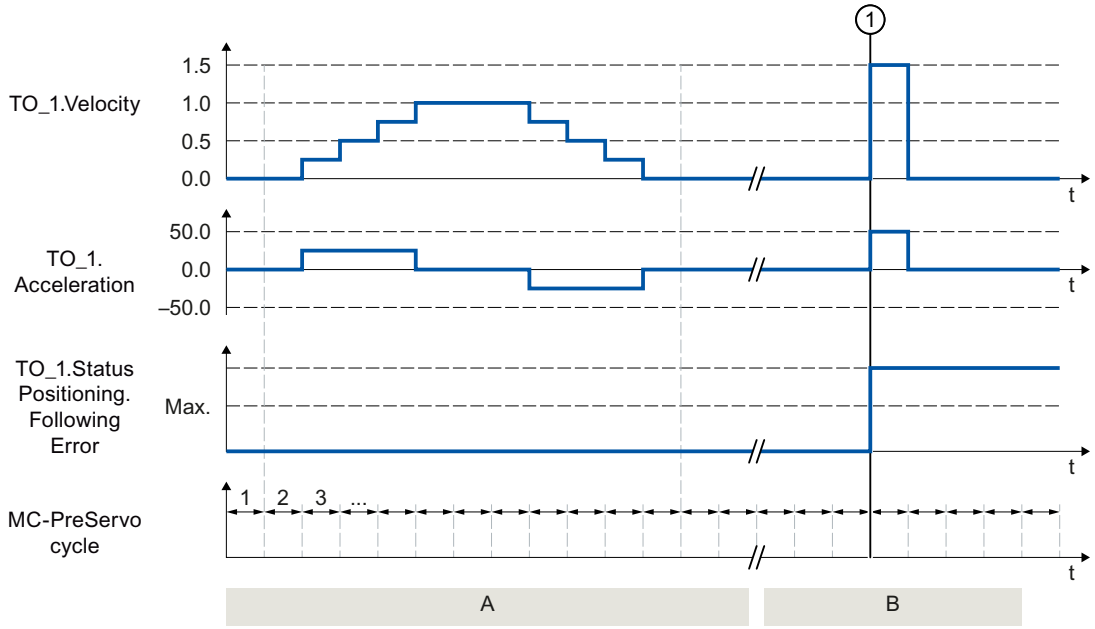

Abschnitt A	Mit "Enable_1 = TRUE" wird dem Technologieobjekt zyklisch im MC-PreServo-Takt Geschwindigkeit "Vel_1" und Beschleunigung "Acc_1" vorgegeben. Diese Vorgaben werden direkt als Sollgeschwindigkeit "TO_1.Velocity" und Sollbeschleunigung "TO_1.Acceleration" übernommen, ohne dass dabei ein Geschwindigkeitsprofil berechnet wird. Da die Positionsüberwachung "PosContr_1" auf FALSE gesetzt ist, wird kein Schleppfehler "TO_1.StatusPositioning.FollowingError" ermittelt.
Abschnitt B	Solange "Enable_1" auf FALSE gesetzt ist, sind die Geschwindigkeits- und Beschleunigungsvorgaben nicht wirksam. Zum Zeitpunkt ① wird "Enable_1" auf TRUE gesetzt. Da die Positionsüberwachung "PosContr_1" auf TRUE gesetzt ist, wird ein Schleppfehler "TO_1.StatusPositioning.FollowingError" ermittelt. Die Geschwindigkeitsvorgabe "Vel_1" und die Beschleunigungsvorgabe "Acc_1" bewirken einen Sollwertsprung, der den maximal zulässigen Schleppfehler überschreitet. Bei aktiver Schleppfehlerüberwachung wird der Technologie-Alarm 521 ausgegeben und das Technologieobjekt gesperrt. Bei deaktivierter Schleppfehlerüberwachung wird der Sollwertsprung mit maximaler Dynamik ausgeführt.

MC_MotionInPosition V4

MC_MotionInPosition: Bewegungssollwerte vorgeben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MotionInPosition" geben Sie der Achse zyklisch applikative Bewegungssollwerte für Position, Geschwindigkeit und Beschleunigung als Basisbewegung vor. Dabei wird kein Geschwindigkeitsprofil berechnet, die Werte sind am Technologieobjekt direkt wirksam. Die Dynamikgrenzen sind nicht wirksam. Die Bewegungsvorgabe über "MotionIn" kann mit einem "MC_MoveSuperimposed"-Auftrag (Seite 3274) überlagert werden.

Mit dem Parameter "Position" geben Sie die Sollposition vor.

Mit dem Parameter "Velocity" geben Sie die Sollgeschwindigkeit vor. Die Sollgeschwindigkeit wird bei aktivierter Geschwindigkeitsvorsteuerung als Vorsteuerwert verwendet. Wenn der "MotionInPosition"-Auftrag durch einen anderen Motion Control-Auftrag abgelöst wird, wird für die Geschwindigkeit auf die Sollgeschwindigkeit des "MC_MotionInPosition"-Auftrags aufgesetzt.

Mit dem Parameter "Acceleration" geben Sie die Sollbeschleunigung vor. Die Sollbeschleunigung wirkt nur, wenn der "MC_MotionInPosition"-Auftrag durch einen anderen Motion Control-Auftrag abgelöst wird. Dabei wird für die Beschleunigung auf die Sollbeschleunigung des "MC_MotionInPosition"-Auftrags aufgesetzt. Wenn die Sollbeschleunigung null ist, wird beim Ablösen auf die Beschleunigung null aufgesetzt.

Anwendbar auf

- Positionierachse
- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MotionInPosition"-Aufträge ist im Kapitel "Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MotionInPosition":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke Solange der Parameter TRUE ist, werden die angegebenen Sollwerte verwendet.
				FALSE	Ende des Auftrags mit fallender Flanke Wenn der Parameter von TRUE auf FALSE gesetzt wird, werden die Sollwerte auf 0.0 gesetzt. Für die Sollposition bleibt der zuletzt angegebene Wert bestehen.
Position	INPUT	LREAL	0.0	Sollposition	
Velocity	INPUT	LREAL	0.0	Sollgeschwindigkeit Beachten Sie die Dynamikgrenzen.	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Acceleration	INPUT	LREAL	0.0	Sollbeschleunigung Beachten Sie die Dynamikgrenzen.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlererkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge (Seite 3394)

MC_MotionInPosition: Funktionsdiagramm V4

Funktionsdiagramm: Bewegungssollwerte vorgeben

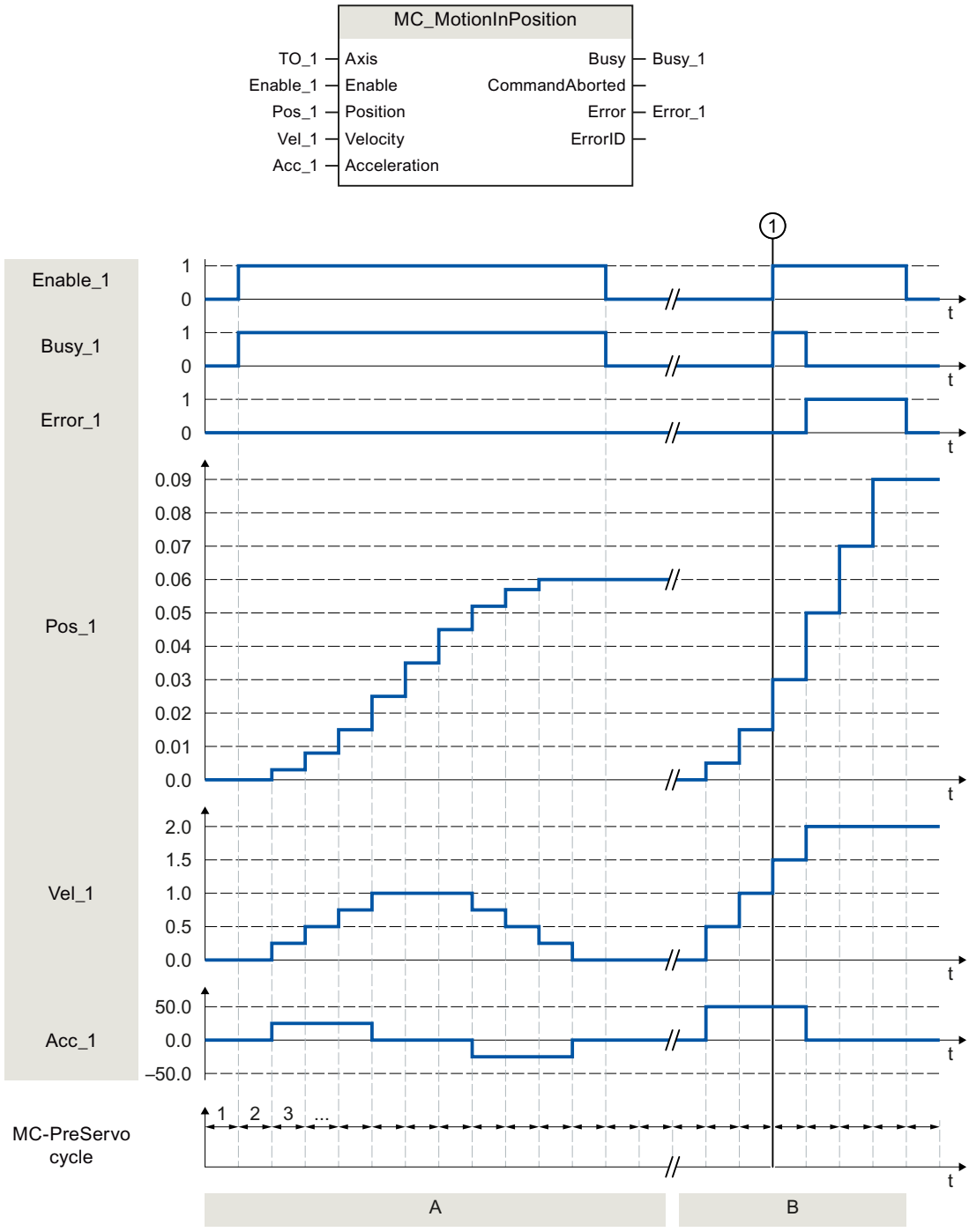

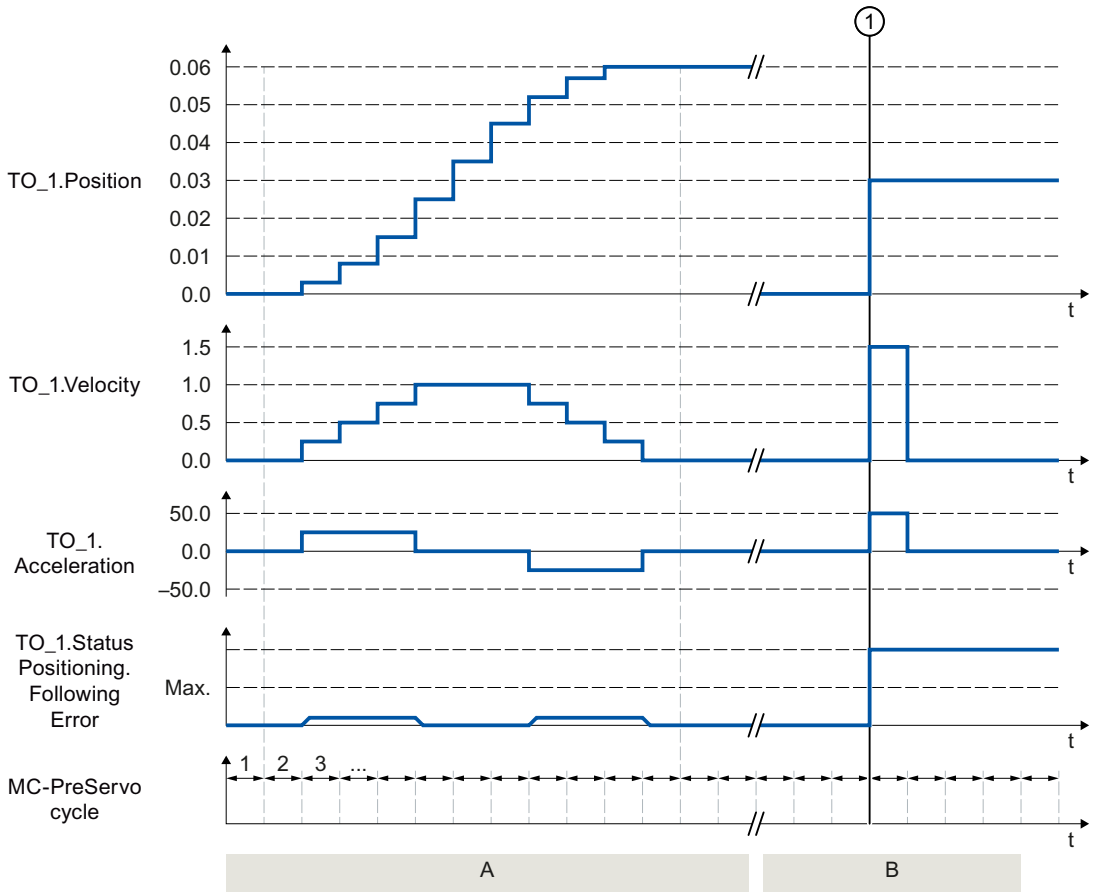

Abschnitt A	Mit "Enable_1 = TRUE" wird dem Technologieobjekt zyklisch im MC-PreServo-Takt Position "Pos_1", Geschwindigkeit "Vel_1" und Beschleunigung "Acc_1" vorgegeben. Diese Vorgaben werden direkt als Sollposition "TO_1.Position", Sollgeschwindigkeit "TO_1.Velocity" und Sollbeschleunigung "TO_1.Acceleration" übernommen, ohne dass dabei ein Geschwindigkeitsprofil berechnet wird.
Abschnitt B	Solange "Enable_1" auf FALSE gesetzt ist, sind die Positions-, Geschwindigkeits- und Beschleunigungsvorgaben nicht wirksam. Zum Zeitpunkt ① wird "Enable_1" auf TRUE gesetzt. Die Positionsvorgabe "Pos_1" bewirkt einen Sollwertsprung, der den maximal zulässigen Schleppfehler überschreitet. Bei aktiver Schleppfehlerüberwachung wird der Technologie-Alarm 521 ausgegeben und das Technologieobjekt gesperrt. Bei deaktivierter Schleppfehlerüberwachung wird der Sollwertsprung mit maximaler Dynamik ausgeführt.

Momentendaten

MC_TorqueLimiting V4

MC_TorqueLimiting: Kraft-/Momentenbegrenzung / Festanschlagserkennung aktivieren und deaktivieren V4

Beschreibung

Aktivieren und parametrieren Sie mit der Motion Control-Anweisung "MC_TorqueLimiting" eine Kraft-/Momentenbegrenzung bzw. eine Festanschlagserkennung. Zusammen mit einem lagegeregelten Bewegungsauftrag kann mit der Festanschlagserkennung ein "Fahren auf Festanschlag" realisiert werden. In der Konfiguration der Achse können Sie konfigurieren, ob die Kraft-/Momentenbegrenzung auf die Antriebsseite oder auf die Lastseite bezogen werden soll.

Die Funktionen der Motion Control-Anweisung "MC_TorqueLimiting" können vor und während eines Bewegungsauftrags aktiviert und deaktiviert werden.

Kraft-/Momentenbegrenzung anwendbar auf

- Drehzahlachse
- Positionierachse
- Gleichlaufachse

Voraussetzung für die Kraft-/Momentenbegrenzung

- Das Technologieobjekt und das Bezugsmoment des Antriebs wurden korrekt konfiguriert.
- Am Technologieobjekt stehen keine Freigabe verhindernden Fehler an (das Technologieobjekt muss nicht freigegeben sein).
- Der Antrieb muss die Kraft-/Momentenreduzierung unterstützen. Nur PROFIdrive-Antriebe mit SIEMENS-Telegramm 10x unterstützen die Kraft-/Momentenbegrenzung.
- Verschaltung im SINAMICS-Antrieb:
  - P1522 auf einen Festwert von 100 %
  - P1523 auf einen Festwert von -100 % (z. B. durch Verschaltung auf Festwertparameter P2902[i])

Festanschlagserkennung anwendbar auf

- Positionierachse
- Gleichlaufachse

Voraussetzung für Festanschlagserkennung

- Die Festanschlagserkennung kann nur auf lagegeregelte Achsen angewandt werden. Zur Festanschlagserkennung muss die Achse lagegeregelt freigegeben sein. Bewegungsaufträge müssen lagegeregelt ausgeführt werden.
- Das Technologieobjekt wurde korrekt konfiguriert.
- Beim Einsatz eines Antriebs und Telegramms, welche die Kraft-/Momentenbegrenzung unterstützten, muss das Bezugsmoment des Antriebs korrekt am Technologieobjekt konfiguriert sein.
- Am Technologieobjekt stehen keine Freigabe verhindernden Fehler an (das Technologieobjekt muss freigegeben sein).

Ablöseverhalten

- Ein "MC_TorqueLimiting"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein neuer "MC_TorqueLimiting"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.
- Wenn die obere und untere Momentenbegrenzung über den "MC_TorqueRange"-Auftrag aktiv ist, wird der "MC_TorqueLimiting"-Auftrag mit einer Fehlermeldung abgelehnt und umgekehrt. Die Funktionen wirken nicht ablösend aufeinander.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_TorqueLimiting":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Funktion entsprechend Eingangsparameter "Mode" aktivieren
Limit	INPUT	LREAL	-1.0	Wert der Kraft-/Momentenbegrenzung (in der konfigurierten Einheit) Unterstützen Antrieb und Telegramm keine Kraft-/Momentenbegrenzung, so ist der angegebene Wert belanglos.	
				≥ 0.0	Verwende den am Parameter angegebenen Wert
				< 0.0	Verwende den im Konfigurationsfenster "Momentenbegrenzung" konfigurierten Wert Variable Grenzwert Moment: <TO>.TorqueLimiting.LimitDefaults.Torque Variable Grenzwert Kraft: <TO>.TorqueLimiting.LimitDefaults.Force

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	DINT	0	0	Kraft/Momentenbegrenzung
				1	Festanschlagserkennung Unterstützen Antrieb und Telegramm die Kraft/Momentenbegrenzung, so wird diese angewandt.
InClamping	OUTPUT	BOOL	FALSE	TRUE	"Mode" = 1: Der Antrieb wird am Festanschlag gehalten (Klemmung), die Achsposition befindet sich innerhalb der Positioniertoleranz.
InLimitation	OUTPUT	BOOL	FALSE	TRUE	"Mode" = 0 und 1: Der Antrieb arbeitet an der Kraft-/Momentengrenze.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Änderungen an den Eingangsparametern "Limit" und "Mode" werden auch bei "Enable" = TRUE beim zyklischen Aufruf der Motion Control-Anweisung übernommen.

MC_TorqueLimiting: Funktionsdiagramm V4

Funktionsdiagramm: Anhalten einer Achse, wenn die Drehmomentgrenze erreicht wird

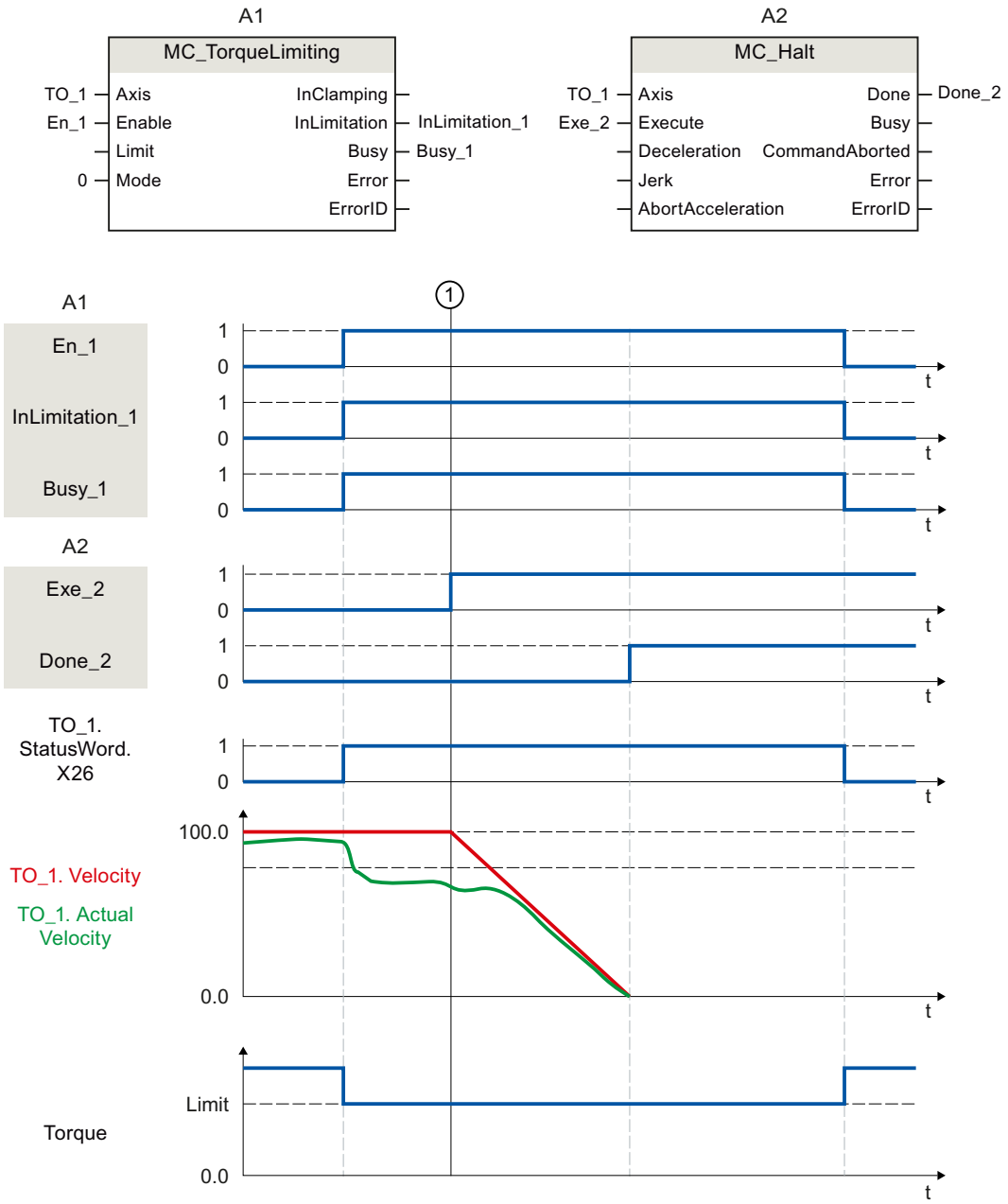

Zum Zeitpunkt ① wird ein "MC_Halt"-Auftrag (A2) auf eine Achse mit aktiver Drehmomentbegrenzung "MC_TorqueLimiting" (A1) ausgeführt. Die Drehmomentbegrenzung ist weiterhin aktiv ("MC_TorqueLimiting.Enable" = TRUE) und ein gegebenenfalls aufgebauter Schleppabstand bleibt erhalten und wird mit der Zeit abgebaut. Wenn die Istgeschwindigkeit "0.0" beträgt und die minimale Verweildauer im Stillstandsfenster abgelaufen ist, zeigt die Variable "MC_Halt.Done" = TRUE. Bei eingeschalteter Positionierüberwachung wird auch das Erreichen der Zielposition überwacht.

Funktionsdiagramm: Drehmomentbegrenzung mit Festanschlagserkennung (Mode = 1)

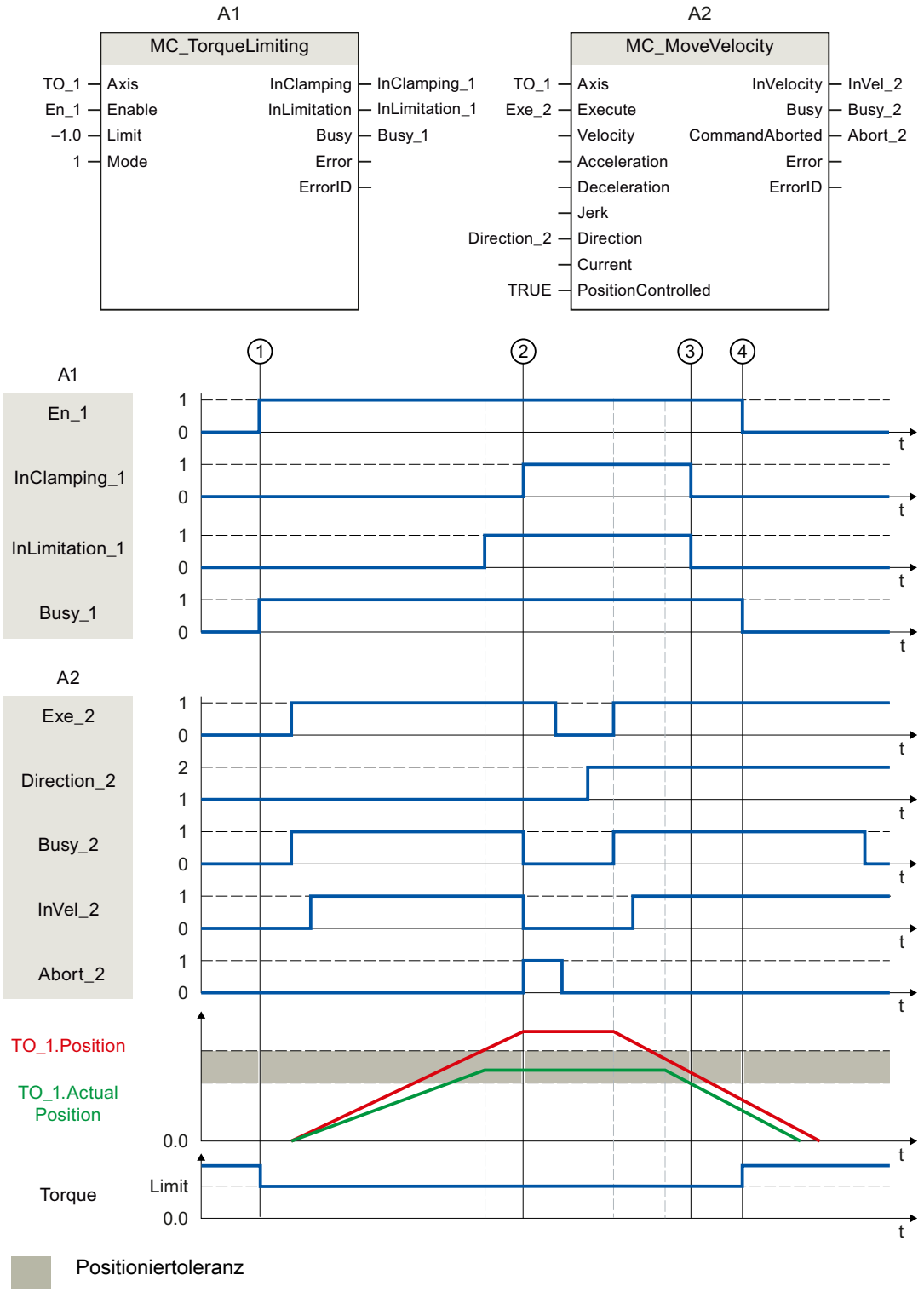

Über "En_1" wird zum Zeitpunkt ① ein "MC_TorqueLimiting"-Auftrag (A1) angestoßen. Auf die Achse mit aktiver Drehmomentbegrenzung wird ein "MC_MoveVelocity"-Auftrag (A2)

ausgeführt. Die Drehmomentbegrenzung ist weiterhin aktiv ("MC_TorqueLimiting.Enable" = TRUE). Beim Erreichen der Schleppabstandsgrenze ② wird der "MC_MoveVelocity"-Auftrag mit "Abort" = TRUE abgebrochen. Der Antrieb wird am Festanschlag gehalten (Klemmung). Die Istposition der Achse befindet sich innerhalb der Positioniertoleranz. Über die beiden Variablen "Execute" = TRUE und "Direction_2" = TRUE wird erneut ein "MC_MoveVelocity"-Auftrag aufgerufen und die Achse bewegt sich mit konstanter Geschwindigkeit in die Gegenrichtung. Dabei wird die Klemmung beim Verlassen der Positioniertoleranz ③ abgebaut. Zum Zeitpunkt ④ wird die Drehmomentbegrenzung aufgehoben.

MC_TorqueAdditive V4

MC_TorqueAdditive: Additives Moment vorgeben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_TorqueAdditive" geben Sie dem Antrieb, der dem Technologieobjekt zugeordnet ist, ein additives Moment vor. Die Momentendaten werden über das Telegramm 750 übertragen.

Mit dem Parameter "Value" geben Sie das additive Sollmoment vor. Die Vorgabe des additiven Sollmoments wirkt überlagernd. Ein Zusatzmoment kann positiv oder negativ sein. Wenn Sie den Sollwert am Technologieobjekt invertieren, wird auch der Wert für das additive Moment invertiert und umgekehrt zum Antrieb übertragen.

Anwendbar auf

- Positionierachse
- Drehzahlachse
- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.
- Der Antrieb ist über ein PROFIdrive-Telegramm angebunden.
- Das Telegramm 750 ist konfiguriert.
  Das Telegramm 750 ist für SINAMICS-Antriebe ab V4.9 verfügbar.

Ablöseverhalten

- Ein "MC_TorqueAdditive"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_TorqueAdditive"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_TorqueAdditive":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Solange der Parameter TRUE ist, wird der angegebene Sollwert verwendet.
				FALSE	Das zum Antrieb übertragene additive Moment ist null.
Value	INPUT	LREAL	0.0	Additives Sollmoment Zulässige Werte: -1.0E12 bis 1.0E12	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_TorqueAdditive: Funktionsdiagramm V4

Funktionsdiagramm: Additives Sollmoment aktivieren/deaktivieren

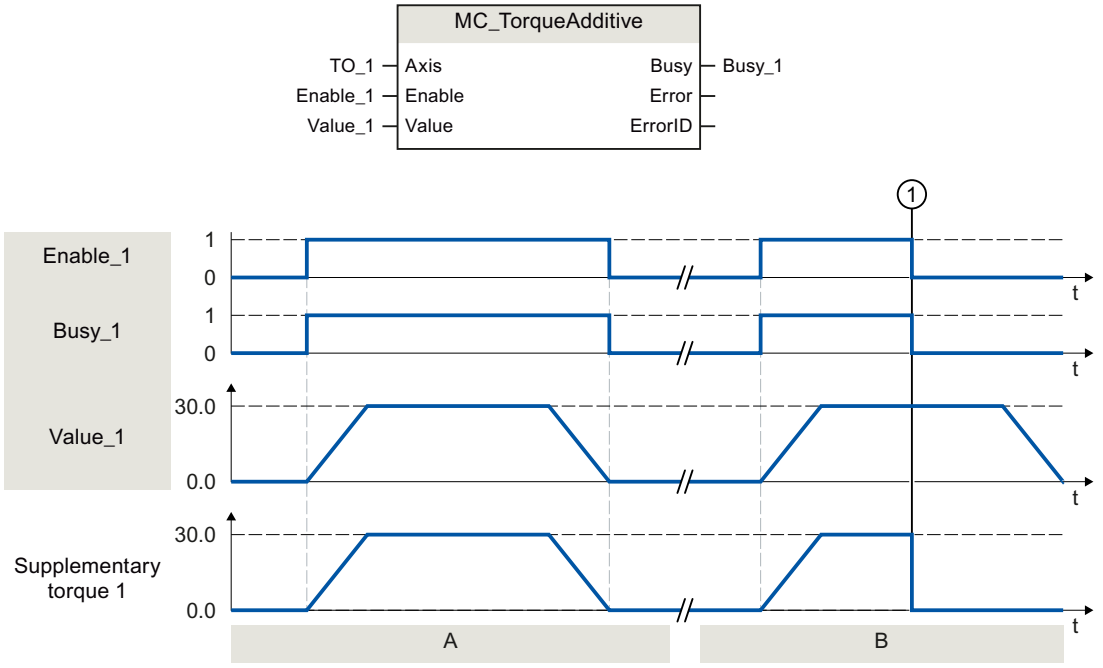

Abschnitt A	Mit "Enable_1 = TRUE" wird dem Antrieb, der dem Technologieobjekt zugeordnet ist, ein additives Sollmoment "Value_1" vorgegeben. Diese Vorgabe wird über das Telegramm 750 an den Antriebsparameter "p1511 - Supplementary torque 1" übertragen.
Abschnitt B	Mit "Enable_1 = TRUE" wird dem Technologieobjekt zugeordneten Antrieb ein additives Sollmoment "Value_1" vorgegeben. Diese Vorgabe wird über das Telegramm 750 an den Antriebsparameter "p1511 - Supplementary torque 1" übertragen. Das additive Sollmoment wird zunächst aufgebaut. Zum Zeitpunkt ① wird "Enable_1" bereits auf FALSE gesetzt, bevor das additive Sollmoment wieder abgebaut ist. Die Verringerung des Sollmoments wird direkt an den Antrieb übertragen.

MC_TorqueRange V4

MC_TorqueRange: Obere und untere Momentengrenze vorgeben V4

Beschreibung

Mit der Motion Control-Anweisung "MC_TorqueRange" können Sie dem Technologieobjekt zugeordneten Antrieb eine obere und untere Momentengrenze vorgeben. Die Momentendaten werden über das Telegramm 750 übertragen.

Mit dem Parameter "UpperLimit" geben Sie die obere und mit "LowerLimit" die untere Momentengrenze vor. Die Vorgabe der Momentengrenzen wirkt zu den Bewegungen überlagernd. Wenn Sie die Sollwerte am Technologieobjekt invertieren, werden auch die Werte für die obere und untere Momentengrenze invertiert und umgekehrt zum Antrieb übertragen.

Wenn die obere und untere Momentengrenze aktiv ist, werden folgende Überwachungen und Begrenzungen standardmäßig deaktiviert:

- Schleppfehlerüberwachung
- Zeitbegrenzungen bei Positionier- und Stillstandsüberwachung

Die Überwachungen bleiben weiterhin wirksam, wenn Sie unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Begrenzungen > Momentenbegrenzung" die Option "Positionsbezogene Überwachungen aktiv lassen" ausgewählt haben.

Anwendbar auf

- Positionierachse
- Drehzahlachse
- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Der Antrieb ist über ein PROFIdrive-Telegramm angebunden.
- Das Telegramm 750 ist konfiguriert.
  Das Telegramm 750 ist für SINAMICS-Antriebe ab V4.9 verfügbar.

Ablöseverhalten

- Ein "MC_TorqueRange"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_TorqueRange"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.
- Wenn die Momentenbegrenzung über den "MC_TorqueLimiting"-Auftrag aktiv ist, wird der "MC_TorqueRange"-Auftrag mit einer Fehlermeldung abgelehnt und umgekehrt. Die Funktionen wirken nicht ablösend aufeinander.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_TorqueRange":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Solange der Parameter TRUE ist, werden die angegebenen Werte verwendet.
				FALSE	Zum Antrieb werden keine Werte für die obere und untere Momentengrenze übertragen.
UpperLimit	INPUT	LREAL	1.0 E12	Obere Momentengrenze (in der konfigurierten Einheit)	
				Zulässiger Wertebereich: -1.0 E12 bis 1.0 E12 Der Wert des Parameters "UpperLimit" muss größer als der Wert des Parameters "LowerLimit" sein.	
LowerLimit	INPUT	LREAL	-1.0 E12	Untere Momentengrenze (in der konfigurierten Einheit)	
				Zulässiger Wertebereich: -1.0 E12 bis 1.0 E12 Der Wert des Parameters "LowerLimit" muss kleiner als der Wert des Parameters "UpperLimit" sein.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_TorqueRange: Funktionsdiagramm V4

Funktionsdiagramm: Obere und untere Momentengrenze vorgeben

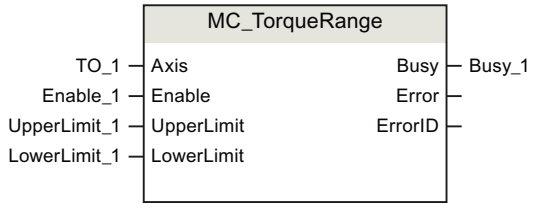

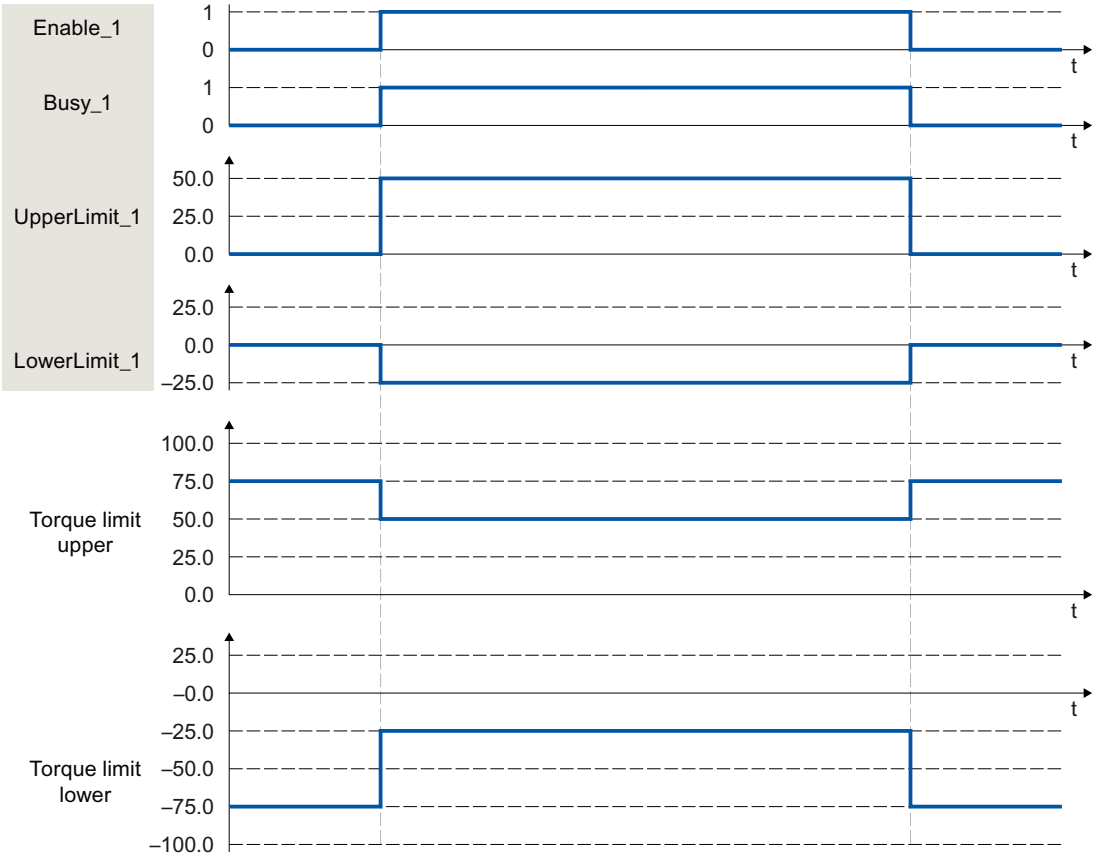

Mit "Enable_1 = TRUE" wird dem Technologieobjekt zugeordneten Antrieb eine obere Momentengrenze "UpperLimit_1" und eine untere Momentengrenze "LowerLimit_1" vorgegeben. Diese Vorgaben werden über das Telegramm 750 an die Antriebsparameter "p1522 - Torque limit upper" und "p1523 - Torque limit lower" übertragen. Wenn "Enable_1" wieder auf FALSE gesetzt wird, sind die oberen und unteren Momentengrenzen nicht mehr wirksam.

Kinematikbewegungen

MC_GroupInterrupt V4

MC_GroupInterrupt: Bewegungsausführung unterbrechen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GroupInterrupt" unterbrechen Sie die Bewegungsausführung am Technologieobjekt Kinematik. Die unterbrochene Kinematikbewegung lässt sich mit einem "MC_GroupContinue"-Auftrag wieder fortsetzen.

Wenn sich die Kinematik bereits im Stillstand befindet, ist die Bewegungsausführung ebenso für folgende Bewegungsaufträge unterbrochen. Neue Bewegungsaufträge reihen sich dann als anstehende Aufträge in die Auftragskette ein.

Mit dem Parameter "Mode" bestimmen Sie das dynamische Verhalten des Anhaltens.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_GroupInterrupt"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GroupInterrupt":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Mode	INPUT	DINT	0	Modus für das dynamische Verhalten	
				0	Mit der Dynamik des zu unterbrechenden Bewegungsauftrags anhalten
				1	Mit maximaler Dynamik der Kinematikbewegung anhalten
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_GroupContinue V4

MC_GroupContinue: Bewegungsausführung fortsetzen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GroupContinue" setzen Sie eine Kinematikbewegung fort, die zuvor mit einem "MC_GroupInterrupt"-Auftrag unterbrochen wurde. Die Kinematikbewegung lässt sich auch fortsetzen, wenn die Kinematik durch den "MC_GroupInterrupt"-Auftrag den Stillstand noch nicht erreicht hat.

Der "MC_GroupContinue"-Auftrag hat nur eine Wirkung, wenn sich das Technologieobjekt im Zustand "Interrupted" befindet.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.

Ablöseverhalten

- Ein "MC_GroupContinue"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_GroupContinue"-Auftrag bricht einen laufenden "MC_GroupInterrupt"-Auftrag ab.
- Das Ablöseverhalten für "MC_GroupContinue"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GroupContinue":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_GroupContinue: Funktionsdiagramm V4

Funktionsdiagramm: Bewegungsausführung fortsetzen

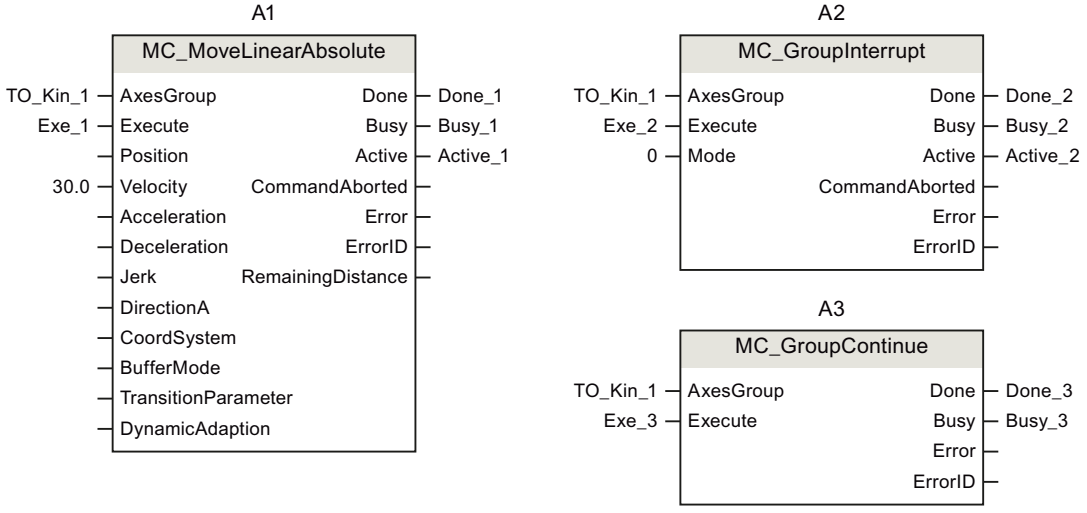

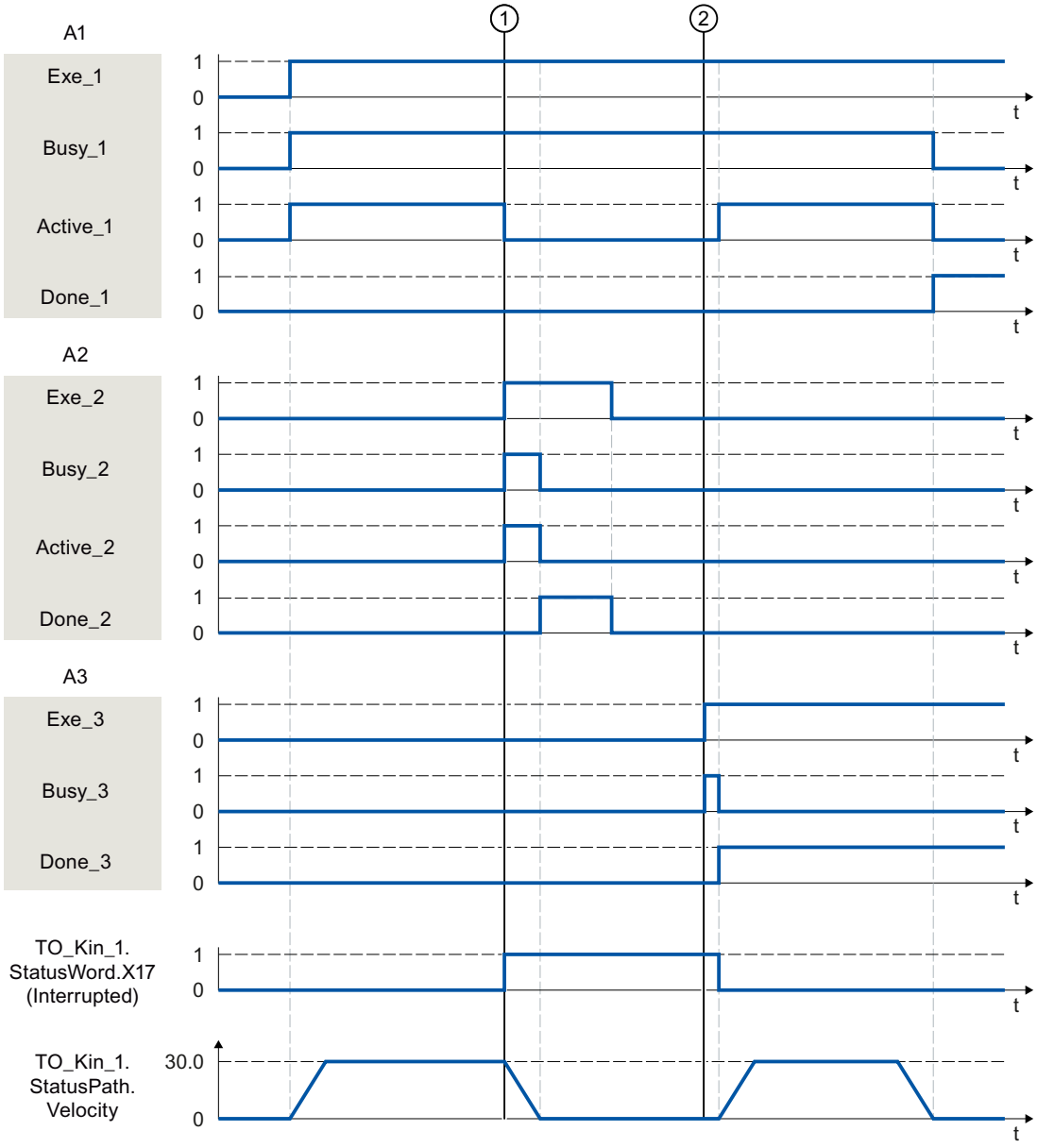

Eine Kinematik wird über einen "MC_MoveLinearAbsolute"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird der "MC_MoveLinearAbsolute"-Auftrag durch einen "MC_GroupInterrupt"-Auftrag (A2) unterbrochen. Die Kinematik befindet sich im Zustand "Interrupted". Die Bewegung wird bei "Mode = 0" mit der Dynamik des "MC_MoveLinearAbsolute"-Auftrags angehalten. Der Abschluss des "MC_GroupInterrupt"-Auftrags wird über "Done_2" gemeldet.

Zum Zeitpunkt ② wird der "MC_MoveLinearAbsolute"-Auftrag durch einen "MC_GroupContinue"-Auftrag (A3) fortgesetzt.

MC_GroupStop V4

MC_GroupStop: Bewegung stoppen V4

Beschreibung

Mit der Motion Control-Anweisung "MC_GroupStop" stoppen und brechen Sie eine aktive Bewegung am Technologieobjekt Kinematik ab. Wenn die Bewegung bereits mit einem "MC_GroupInterrupt"-Auftrag unterbrochen wurde, wird diese abgebrochen. Auch alle noch anstehenden Aufträge in der Auftragskette werden durch einen "MC_GroupStop"-Auftrag abgebrochen. Solange der Parameter "Execute" auf TRUE gesetzt ist, werden Kinematikaufträge abgelehnt ("ErrorID" = 16#80CD).

Mit dem Parameter "Mode" bestimmen Sie das dynamische Verhalten der Stopp-Bewegung.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_GroupStop"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GroupStop":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Mode	INPUT	DINT	0	Modus für das dynamische Verhalten	
				0	Mit der Dynamik des zu unterbrechenden Bewegungsauftrags anhalten
				1	Mit maximaler Dynamik der Kinematikbewegung anhalten Bei aktivierter Dynamikadaption wird die Dynamik gegebenenfalls reduziert, sodass die Dynamikgrenzen der Achsen nicht überschritten werden.
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_GroupStop: Funktionsdiagramm V4

Funktionsdiagramm: Bewegung stoppen

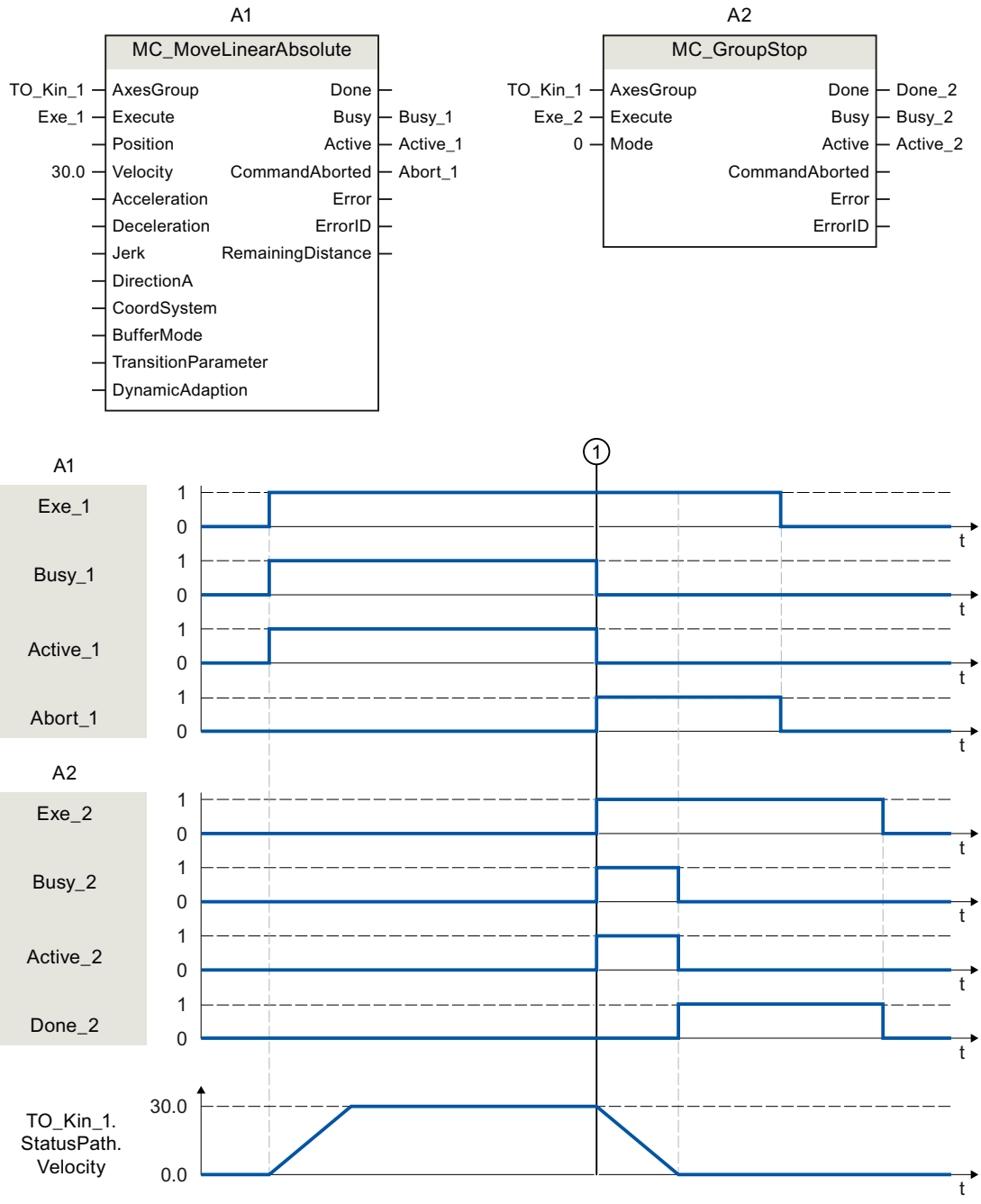

Eine Kinematik wird über einen "MC_MoveLinearAbsolute"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird der "MC_MoveLinearAbsolute"-Auftrag durch einen "MC_GroupStop"-Auftrag (A2) abgebrochen. Der Auftragsabbruch wird über "Abort_1" gemeldet. Die Bewegung

wird bei "Mode = 0" mit der Dynamik des "MC_MoveLinearAbsolute"-Auftrags angehalten. Der Abschluss des "MC_GroupStop"-Auftrags wird über "Done_2" gemeldet.

MC_MoveLinearAbsolute V4

MC_MoveLinearAbsolute: Kinematik mit linearer Bahnbewegung positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveLinearAbsolute" verfahren Sie eine Kinematik mit einer linearen Bewegung auf eine absolute Position. Ebenso wird die kartesische Orientierung absolut verfahren.

Mit den Parametern "Velocity", "Acceleration", "Deceleration" und "Jerk" bestimmen Sie das dynamische Verhalten der Bewegung. Für die Dynamik der Orientierungsbewegung werden die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierten Standardwerte verwendet:

- <TO>.DynamicDefaults.Orientation.Velocity
- <TO>.DynamicDefaults.Orientation.Acceleration
- <TO>.DynamicDefaults.Orientation.Deceleration
- <TO>.DynamicDefaults.Orientation.Jerk

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- Die verschalteten Achsen sind referenziert.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

Mit dem Parameter "TransitionParameter[1]" bestimmen Sie, in welchem Abstand zum Zielpunkt des vorherigen Bewegungsauftrags in den neuen Bewegungsaufrag überschliffen wird.

Das Ablöseverhalten für "MC_MoveLinearAbsolute"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveLinearAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Position	INPUT	ARRAY [1..4] OF LREAL	-	Absolute Zielkoordinate im angegebenen Bezugssystem	
Position[1]	INPUT	LREAL	0.0	x-Koordinate	
Position[2]	INPUT	LREAL	0.0	y-Koordinate	
Position[3]	INPUT	LREAL	0.0	z-Koordinate	
Position[4]	INPUT	LREAL	0.0	A-Koordinate	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Keine Ruckbegrenzung
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
DirectionA	INPUT	DINT	3	Bewegungsrichtung der kartesischen Orientierung	
				1	Positive Richtung
				2	Negative Richtung
				3	Kürzester Weg
CoordSystem	INPUT	DINT	0	Bezugssystem der angegebenen Zielposition und Zielorientierung	
				0	Weltkoordinatensystem (WCS)
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])
BufferMode	INPUT	DINT	1	Bewegungsübergang	
				1	Bewegung anhängen Die laufende Bewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Bewegung dieses Auftrags ausgeführt.
				2	Mit der niedrigeren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die niedrigere Geschwindigkeit beider Aufträge verwendet.
				5	Mit der höheren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die höhere Geschwindigkeit beider Aufträge verwendet.
TransitionParameter	INPUT	ARRAY [1..5] OF LREAL		Übergangsparameter	
TransitionParameter[1]	INPUT	LREAL	-1.0	Überschleifabstand	
				≥ 0.0	Der angegebene Wert wird verwendet.
				< 0.0	Der maximal mögliche Überschleifabstand wird verwendet.
TransitionParameter[2]	INPUT	LREAL	-	Reserviert	
TransitionParameter[3]	INPUT	LREAL	-	Reserviert	
TransitionParameter[4]	INPUT	LREAL	-	Reserviert	
TransitionParameter[5]	INPUT	LREAL	-	Reserviert	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
DynamicAdaption	INPUT	DINT	-1	Dynamikadaption	
				< 0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierte Einstellung wird verwendet. (<TO>.DynamicDefaults.DynamicAdaption)
				0	Ohne Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
				FALSE	Bei "Busy" = TRUE: Der Auftrag wartet. (Typisch: Ein vorhergehender Auftrag ist noch aktiv.)
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
RemainingDistance	OUTPUT	LREAL	-1.0	Restwegstrecke des aktuellen Auftrags	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_MoveLinearAbsolute: Funktionsdiagramm V4

Funktionsdiagramm: Kinematik mit linearer Bahnbewegung positionieren

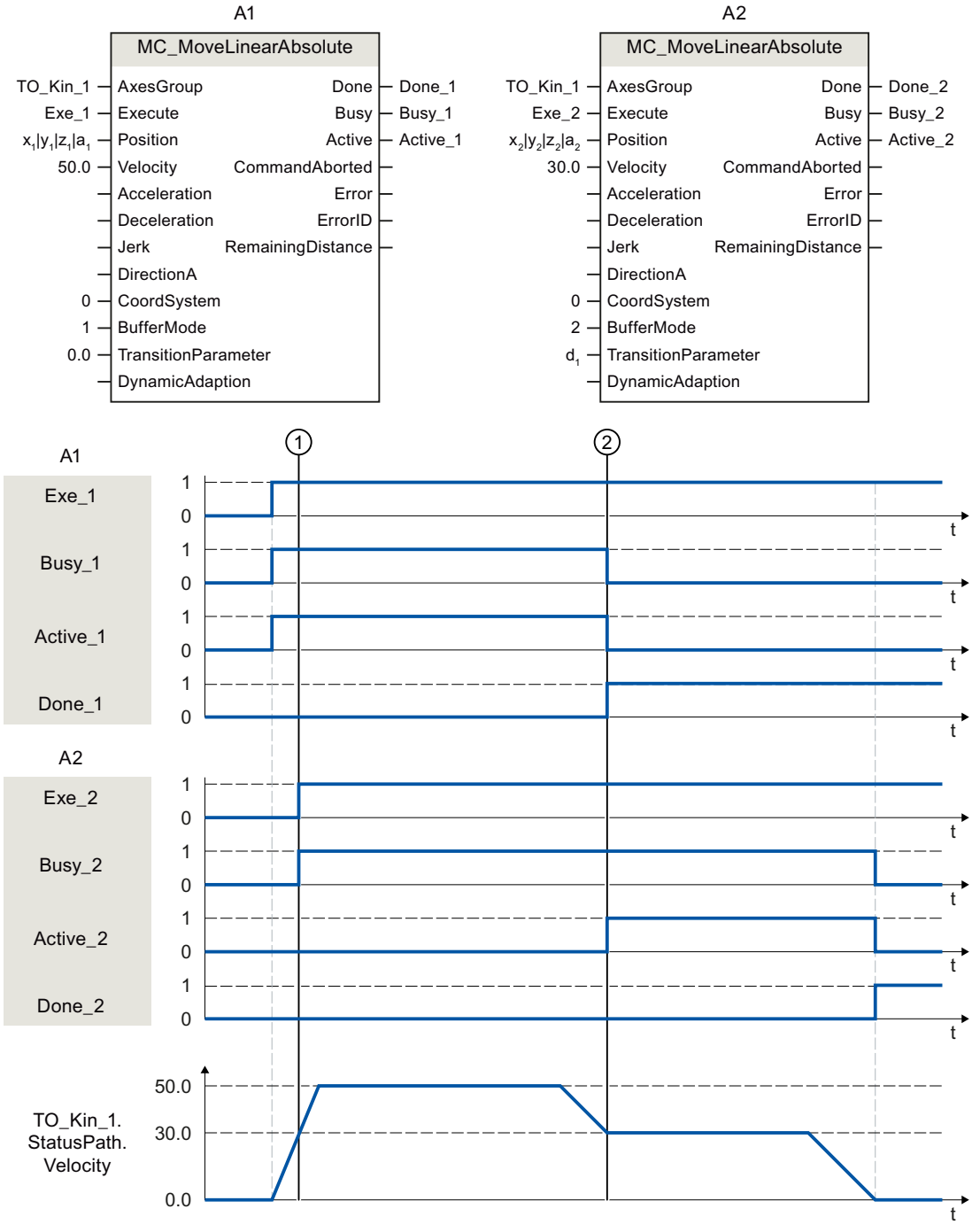

Eine Kinematik wird über einen "MC_MoveLinearAbsolute"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird ein weiterer "MC_MoveLinearAbsolute"-Auftrag (A2) angestoßen. Da "MC_MoveLinearAbsolute"-Aufträge nicht ablösend aufeinander wirken, reiht sich der Auftrag A2 in die Auftragskette ein.

Zum Zeitpunkt ② wird der Abschluss des Auftrags A1 über "Done_1" gemeldet und der Auftrag A2 wird gestartet. Da für den Auftrag A2 "BufferMode = 2" gesetzt ist, wird der Bewegungsübergang mit der niedrigeren Geschwindigkeit beider Aufträge überschliffen. Sobald die Zielposition erreicht ist, wird der Abschluss des Auftrags A2 über "Done_2" gemeldet.

MC_MoveLinearRelative V4

MC_MoveLinearRelative: Kinematik mit linearer Bahnbewegung relativ positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveLinearRelative" verfahren Sie eine Kinematik mit einer linearen Bewegung relativ zu der Position, die zu Beginn der Auftragsbearbeitung vorliegt. Ebenso wird die kartesische Orientierung relativ verfahren.

Mit den Parametern "Velocity", "Acceleration", "Deceleration" und "Jerk" bestimmen Sie das dynamische Verhalten der Bewegung. Für die Dynamik der Orientierungsbewegung werden die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierten Standardwerte verwendet:

- <TO>.DynamicDefaults.Orientation.Velocity
- <TO>.DynamicDefaults.Orientation.Acceleration
- <TO>.DynamicDefaults.Orientation.Deceleration
- <TO>.DynamicDefaults.Orientation.Jerk

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

Mit dem Parameter "TransitionParameter[1]" bestimmen Sie, in welchem Abstand zum Zielpunkt des vorherigen Bewegungsauftrags in den neuen Bewegungsaufrag überschliffen wird.

Das Ablöseverhalten für "MC_MoveLinearRelative"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveLinearRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
Distance	INPUT	ARRAY [1..4] OF LREAL	-	Relative Zielkoordinaten im angegebenen Bezugssystem	
Distance[1]	INPUT	LREAL	0.0	x-Koordinate	
Distance[2]	INPUT	LREAL	0.0	y-Koordinate	
Distance[3]	INPUT	LREAL	0.0	z-Koordinate	
Distance[4]	INPUT	LREAL	0.0	A-Koordinate	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Keine Ruckbegrenzung
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Jerk)
CoordSystem	INPUT	DINT	0	Bezugssystem der angegebenen Zielposition und Zielorientierung	
				0	Weltkoordinatensystem (WCS)
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])
BufferMode	INPUT	DINT	1	Bewegungsübergang	
				1	Bewegung anhängen Die laufende Bewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Bewegung dieses Auftrags ausgeführt.
				2	Mit der niedrigeren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die niedrigere Geschwindigkeit beider Aufträge verwendet.
				5	Mit der höheren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die höhere Geschwindigkeit beider Aufträge verwendet.
TransitionParameter	INPUT	ARRAY [1..5] OF LREAL	-	Übergangsparameter	
TransitionParameter[1]	INPUT	LREAL	-1.0	Überschleifabstand	
				≥ 0.0	Der angegebene Wert wird verwendet.
				< 0.0	Der maximal mögliche Überschleifabstand wird verwendet.
TransitionParameter[2]	INPUT	LREAL	-	Reserviert	
TransitionParameter[3]	INPUT	LREAL	-	Reserviert	
TransitionParameter[4]	INPUT	LREAL	-	Reserviert	
TransitionParameter[5]	INPUT	LREAL	-	Reserviert	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
DynamicAdaption	INPUT	DINT	-1.0	Dynamikadaption	
				< 0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierte Einstellung wird verwendet. (<TO>.DynamicDefaults.DynamicAdaption)
				0	Ohne Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
				FALSE	Bei "Busy" = TRUE: Der Auftrag wartet. (Typisch: Ein vorhergehender Auftrag ist noch aktiv.)
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
RemainingDistance	OUTPUT	LREAL	-1.0	Restwegstrecke des aktuellen Auftrags	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_MoveLinearRelative: Funktionsdiagramm V4

Funktionsdiagramm: Kinematik mit linearer Bahnbewegung relativ positionieren

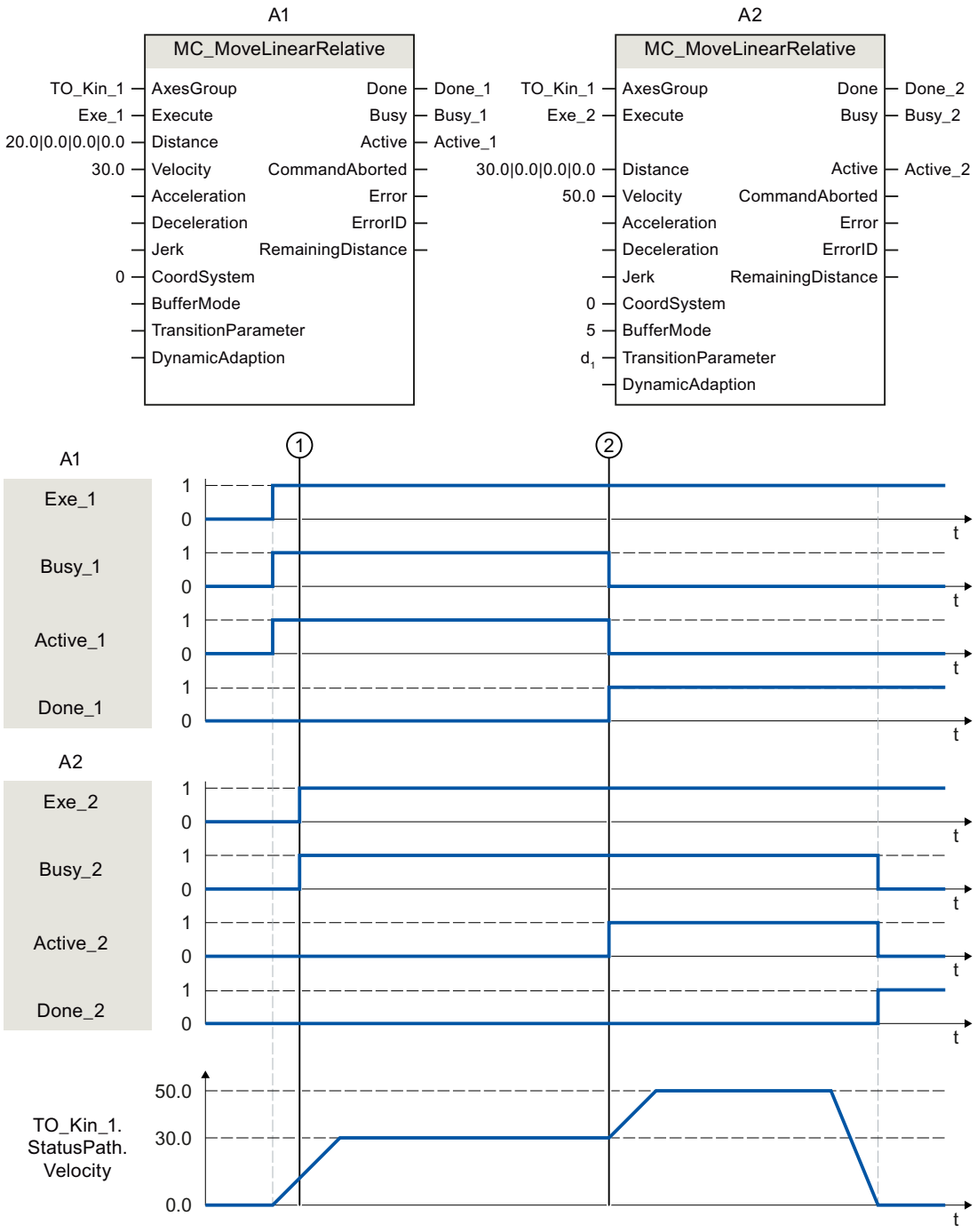

Eine Kinematik wird über einen "MC_MoveLinearRelative"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird ein weiterer "MC_MoveLinearRelative"-Auftrag (A2) angestoßen. Da "MC_MoveLinearRelative"-Aufträge nicht ablösend aufeinander wirken, reiht sich der Auftrag A2 in die Auftragskette ein.

Zum Zeitpunkt ② wird der Abschluss des Auftrags A1 über "Done_1" gemeldet und der Auftrag A2 wird gestartet. Da für den Auftrag A2 "BufferMode = 5" gesetzt ist, wird der Bewegungsübergang mit der höheren Geschwindigkeit beider Aufträge überschliffen. Sobald die Zielposition erreicht ist, wird der Abschluss des Auftrags A2 über "Done_2" gemeldet.

MC_MoveCircularAbsolute V4

MC_MoveCircularAbsolute: Kinematik mit zirkularer Bahnbewegung positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveCircularAbsolute" verfahren Sie eine Kinematik mit einer zirkularen Bewegung auf eine absolute Position. Ebenso wird die kartesische Orientierung absolut verfahren.

Mit dem Parameter "CircMode" legen Sie die Definition der Kreisbahn fest:

- Über einen Zwischen- und den Endpunkt
  Mit dem Parameter "AuxPoint" geben Sie einen Zwischenpunkt auf der Kreisbahn an, über den der im Parameter "EndPoint" festgelegte Endpunkt angefahren werden soll. Über den Start-, Zwischen- und Endpunkt wird die Kreisbahn berechnet. Dabei können nur Kreisbahnen unter 360° verfahren werden.
- Über den Kreismittelpunkt und Winkel in einer Hauptebene
  Mit dem Parameter "AuxPoint" definieren Sie den Mittelpunkt des Kreises. Über den im Parameter "Arc" angegebenen Winkel wird der Endpunkt der Kreisbahn berechnet. Mit dem Parameter "PathChoice" legen Sie fest, ob die Kreisbahn in positiver oder negativer Drehrichtung verfahren werden soll. Mit dem Parameter "CirclePlane" geben Sie die Hauptebene an, in der die Kreisbahn verfahren werden soll.
- Über den Kreisradius und den Endpunkt in einer Hauptebene
  Mit dem Parameter "EndPoint" geben Sie den Endpunkt und mit "Radius" den Radius der Kreisbahn an. Abhängig vom Radius können sich dabei in der mit dem Parameter "CirclePlane" definierten Ebene bis zu vier mögliche Kreisbahnen ergeben. Mit dem Parameter "PathChoice" legen Sie fest, welche dieser Kreisbahnen verfahren werden soll.

Mit den Parametern "Velocity", "Acceleration", "Deceleration" und "Jerk" bestimmen Sie das dynamische Verhalten der Kinematikbewegung. Für die Dynamik der Orientierungsbewegung werden die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierten Standardwerte verwendet:

- <TO>.DynamicDefaults.Orientation.Velocity
- <TO>.DynamicDefaults.Orientation.Acceleration
- <TO>.DynamicDefaults.Orientation.Deceleration
- <TO>.DynamicDefaults.Orientation.Jerk

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- Die verschalteten Achsen sind referenziert.
- An den verschalteten Achsen des Technologieobjekts ist kein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

Mit dem Parameter "TransitionParameter[1]" bestimmen Sie, in welchem Abstand zum Zielpunkt des vorherigen Bewegungsauftrags in den neuen Bewegungsaufrag überschliffen wird.

Das Ablöseverhalten für "MC_MoveCircularAbsolute"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveCircularAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
CircMode	INPUT	DINT	0	Definition der Kreisbahn	
				0	Der am Parameter "AuxPoint" angegebene Positionsvektor definiert einen Punkt auf der Kreisbahn.
				1	Die am Parameter "AuxPoint" angegebene Position definiert den Kreismittelpunkt.
				2	Die Parameter "Radius" und "EndPoint" definieren das Kreissegment.
AuxPoint	INPUT	ARRAY [1..3] OF LREAL	-	Kreisbahn-Hilfspunkt • Bei "CircMode" = 0: Punkt auf der Kreisbahn • Bei "CircMode" = 1: Mittelpunkt der Kreisbahn • Bei "CircMode" = 2: Nicht relevant	
AuxPoint[1]	INPUT	LREAL	0.0	x-Koordinate	
AuxPoint[2]	INPUT	LREAL	0.0	y-Koordinate	
AuxPoint[3]	INPUT	LREAL	0.0	z-Koordinate	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
EndPoint	INPUT	ARRAY [1..4] OF LREAL	-	Zielposition im angegebenen Bezugssystem	
				Bei "CircMode" = 1: Nur EndPoint[4] relevant (Orientierungsachse)	
EndPoint[1]	INPUT	LREAL	0.0	x-Koordinate	
EndPoint[2]	INPUT	LREAL	0.0	y-Koordinate	
EndPoint[3]	INPUT	LREAL	0.0	z-Koordinate	
EndPoint[4]	INPUT	LREAL	0.0	A-Koordinate	
PathChoice	INPUT	DINT	0	Orientierung der Kreisbahn	
				Bei "CircMode" = 0: Nicht relevant	
				Bei "CircMode" = 1:	
				0	Positive Drehrichtung
				1	Negative Drehrichtung
				Bei "CircMode" = 2:	
				0	Kürzeres positives Kreissegment
				1	Kürzeres negatives Kreissegment
				2	Längeres positives Kreissegment
				3	Längeres negatives Kreissegment
CirclePlane	INPUT	DINT	0	Hauptebene der Kreisbahn	
				Bei "CircMode" = 0: Nicht relevant	
				Bei "CircMode" = 1 und 2:	
				0	x-z-Ebene
				1	y-z-Ebene
				2	x-y-Ebene
Radius	INPUT	LREAL	0.0	Bei "CircMode" = 2: Radius der Kreisbewegung	
Arc	INPUT	LREAL	0.0	Bei "CircMode" = 1: Winkel der Kreisbewegung	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Acceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Keine Ruckbegrenzung
				< 0.0	Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Wert wird verwendet. (<TO>.DynamicDefaults.Path.Jerk)
DirectionA	INPUT	DINT	3	Bewegungsrichtung der kartesischen Orientierung	
				1	Positive Richtung
				2	Negative Richtung
				3	Kürzester Weg
CoordSystem	INPUT	DINT	0	Bezugssystem	
				0	Weltkoordinatensystem (WCS)
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
BufferMode	INPUT	DINT	1	Bewegungsübergang	
				1	Bewegung anhängen Die laufende Bewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Bewegung dieses Auftrags ausgeführt.
				2	Mit der niedrigeren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die niedrigere Geschwindigkeit beider Aufträge verwendet.
				5	Mit der höheren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die höhere Geschwindigkeit beider Aufträge verwendet.
TransitionParameter	INPUT	ARRAY [1..5] OF LREAL		Übergangsparameter	
TransitionParameter[1]	INPUT	LREAL	-1.0	Überschleifabstand	
				≥ 0.0	Der angegebene Wert wird verwendet.
				< 0.0	Der maximal mögliche Überschleifabstand wird verwendet.
TransitionParameter[2]	INPUT	LREAL	-	Reserviert	
TransitionParameter[3]	INPUT	LREAL	-	Reserviert	
TransitionParameter[4]	INPUT	LREAL	-	Reserviert	
TransitionParameter[5]	INPUT	LREAL	-	Reserviert	
DynamicAdaption	INPUT	DINT	-1.0	Dynamikadaption	
				< 0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierte Einstellung wird verwendet. (<TO>.DynamicDefaults.DynamicAdaption)
				0	Ohne Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
				FALSE	Bei "Busy" = TRUE: Der Auftrag wartet. (Typisch: Ein vorhergehender Auftrag ist noch aktiv.)
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
RemainingDistance	OUTPUT	LREAL	-1.0	Restwegstrecke des aktuellen Auftrags	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_MoveCircularAbsolute: Funktionsdiagramm V4

Funktionsdiagramm: Kinematik mit zirkularer Bahnbewegung positionieren

Im Funktionsdiagramm werden die Abschnitte A1 und A2 der folgenden Kinematikbewegung betrachtet:

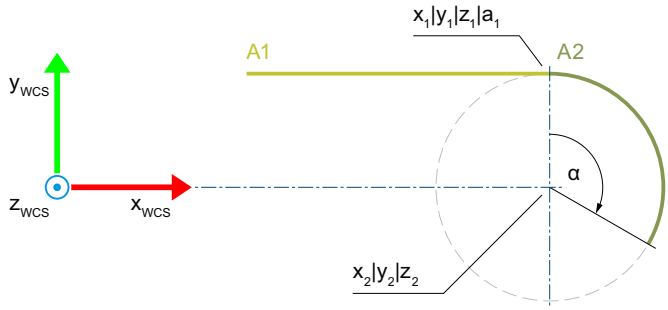

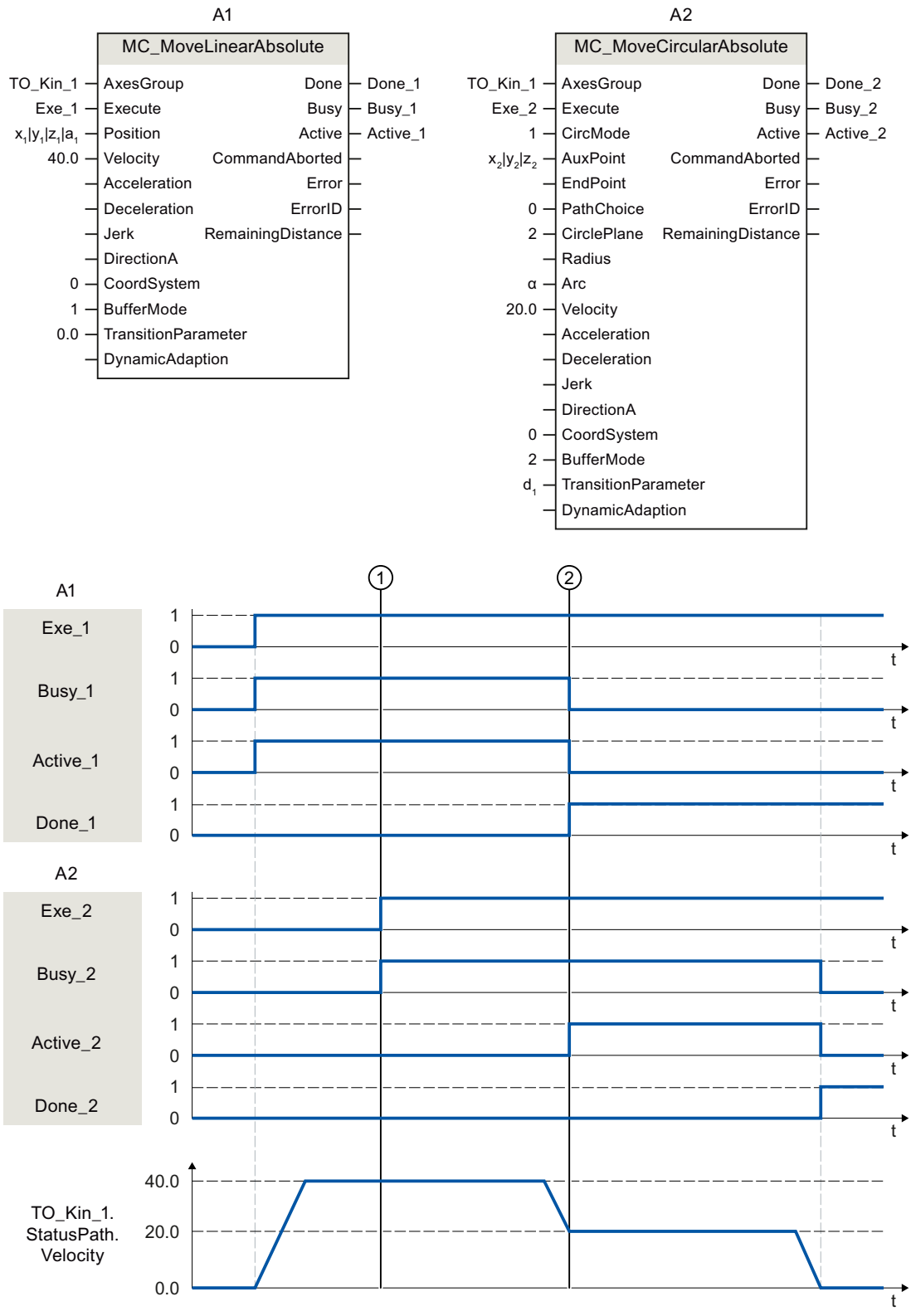

Eine Kinematik wird über einen "MC_MoveLinearAbsolute"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird ein "MC_MoveCircularAbsolute"-Auftrag (A2) angestoßen. Da der "MC_MoveCircularAbsolute"-Auftrag nicht ablösend wirkt, reiht sich der Auftrag A2 in die Auftragskette ein.

Zum Zeitpunkt ② wird der Abschluss des Auftrags A1 über "Done_1" gemeldet und der Auftrag A2 gestartet. Da für den Auftrag A2 "BufferMode = 2" gesetzt ist, wird der Bewegungsübergang mit der niedrigeren Geschwindigkeit beider Aufträge und dem Überschleifabstand d_1 überschliffen. Die zirkulare Bewegung wird mit "CircMode = 1" über den Kreismittelpunkt (x_2|y_2|z_2) und den Winkel α bestimmt. Sobald die Zielposition erreicht ist, wird der Abschluss des Auftrags A2 über "Done_2" gemeldet.

MC_MoveCircularRelative V4

MC_MoveCircularRelative: Kinematik mit zirkularer Bahnbewegung relativ positionieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveCircularRelative" verfahren Sie eine Kinematik mit einer zirkularen Bewegung auf eine relative Position. Ebenso wird die kartesische Orientierung relativ verfahren.

Mit dem Parameter "CircMode" legen Sie die Definition der Kreisbahn fest:

- Über einen Zwischen- und den Endpunkt
  Mit dem Parameter "AuxPoint" geben Sie einen Zwischenpunkt auf der Kreisbahn an, über den der im Parameter "EndPoint" festgelegte Endpunkt angefahren werden soll. Über den Start-, Zwischen- und Endpunkt wird die Kreisbahn berechnet. Dabei können nur Kreisbahnen unter 360° verfahren werden.
- Über den Kreismittelpunkt und Winkel in einer Hauptebene
  Mit dem Parameter "AuxPoint" definieren Sie den Mittelpunkt des Kreises. Über den im Parameter "Arc" angegebenen Winkel wird der Endpunkt der Kreisbahn berechnet. Mit dem Parameter "PathChoice" legen Sie fest, ob die Kreisbahn in positiver oder negativer Drehrichtung verfahren werden soll. Mit dem Parameter "CirclePlane" geben Sie die Hauptebene an, in der die Kreisbahn verfahren werden soll.
- Über den Kreisradius und den Endpunkt in einer Hauptebene
  Mit dem Parameter "EndPoint" geben Sie den Endpunkt und mit "Radius" den Radius der Kreisbahn an. Abhängig vom Radius können sich dabei in der mit dem Parameter "CirclePlane" definierten Ebene bis zu vier mögliche Kreisbahnen ergeben. Mit dem Parameter "PathChoice" legen Sie fest, welche dieser Kreisbahnen verfahren werden soll.

Mit den Parametern "Velocity", "Acceleration", "Deceleration" und "Jerk" bestimmen Sie das dynamische Verhalten der Kinematikbewegung. Für die Dynamik der Orientierungsbewegung werden die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierten Standardwerte verwendet:

- <TO>.DynamicDefaults.Orientation.Velocity
- <TO>.DynamicDefaults.Orientation.Acceleration
- <TO>.DynamicDefaults.Orientation.Deceleration
- <TO>.DynamicDefaults.Orientation.Jerk

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An den verschalteten Achsen des Technologieobjekts ist kein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

Mit dem Parameter "TransitionParameter[1]" bestimmen Sie, in welchem Abstand zum Zielpunkt des vorherigen Bewegungsauftrags in den neuen Bewegungsaufrag überschliffen wird.

Das Ablöseverhalten für "MC_MoveCircularAbsolute"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveCircularRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
CircMode	INPUT	DINT	0	Definition der Kreisbahn	
				0	Der am Parameter "AuxPoint" angegebene Positionsvektor definiert einen Punkt auf der Kreisbahn.
				1	Die am Parameter "AuxPoint" angegebene Position definiert den Kreismittelpunkt.
				2	Die Parameter "Radius" und "EndPoint" definieren das Kreissegment.
AuxPoint	INPUT	ARRAY [1..3] OF LREAL	-	Kreisbahn-Hilfspunkt • Bei "CircMode" = 0: Punkt auf der Kreisbahn • Bei "CircMode" = 1: Mittelpunkt der Kreisbahn • Bei "CircMode" = 2: Nicht relevant Relative Koordinate zum Startpunkt	
AuxPoint[1]	INPUT	LREAL	0.0	x-Koordinate	
AuxPoint[2]	INPUT	LREAL	0.0	y-Koordinate	
AuxPoint[3]	INPUT	LREAL	0.0	z-Koordinate	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
EndPoint	INPUT	ARRAY [1..4] OF LREAL	-	Zielposition relativ zum Startpunkt im angegebenen Bezugssystem	
				Bei "CircMode" = 1: Nur EndPoint[4] relevant (Orientierungsachse)	
EndPoint[1]	INPUT	LREAL	0.0	x-Koordinate	
EndPoint[2]	INPUT	LREAL	0.0	y-Koordinate	
EndPoint[3]	INPUT	LREAL	0.0	z-Koordinate	
EndPoint[4]	INPUT	LREAL	0.0	A-Koordinate	
PathChoice	INPUT	DINT	0	Orientierung der Kreisbahn	
				Bei "CircMode" = 0: Nicht relevant	
				Bei "CircMode" = 1:	
				0	Positive Drehrichtung
				1	Negative Drehrichtung
				Bei "CircMode" = 2:	
				0	Kürzere positive Kreisbahn
				1	Kürzere negative Kreisbahn
				2	Längere positive Kreisbahn
				3	Längere negative Kreisbahn
CirclePlane	INPUT	DINT	0	Hauptebene der Kreisbahn	
				Bei "CircMode" = 0: Nicht relevant	
				Bei "CircMode" = 1 und 2:	
				0	x-z-Ebene
				1	y-z-Ebene
				2	x-y-Ebene
Radius	INPUT	LREAL	0.0	Bei "CircMode" = 2: Radius der Kreisbewegung	
Arc	INPUT	LREAL	0.0	Bei "CircMode" = 1: Winkel der Kreisbewegung	
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Path.Velocity)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Path.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Nicht zulässig
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Path.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck	
				> 0.0	Der angegebene Wert wird verwendet.
				= 0.0	Keine Ruckbegrenzung
				< 0.0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Path.Jerk)
CoordSystem	INPUT	DINT	0	Bezugssystem	
				0	Weltkoordinatensystem (WCS)
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
BufferMode	INPUT	DINT	1	Bewegungsübergang	
				1	Bewegung anhängen Die laufende Bewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Bewegung dieses Auftrags ausgeführt.
				2	Mit der niedrigeren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die niedrigere Geschwindigkeit beider Aufträge verwendet.
				5	Mit der höheren Geschwindigkeit überschleifen Mit dem Erreichen des Überschleifabstands wird die laufende Bewegung mit der Bewegung dieses Auftrags überschliffen. Dabei wird die höhere Geschwindigkeit beider Aufträge verwendet.
TransitionParameter	INPUT	ARRAY [1..5] OF LREAL		Übergangsparameter	
TransitionParameter[1]	INPUT	LREAL	-1.0	Überschleifabstand	
				≥ 0.0	Der angegebene Wert wird verwendet.
				< 0.0	Der maximal mögliche Überschleifabstand wird verwendet.
TransitionParameter[2]	INPUT	LREAL	-	Reserviert	
TransitionParameter[3]	INPUT	LREAL	-	Reserviert	
TransitionParameter[4]	INPUT	LREAL	-	Reserviert	
TransitionParameter[5]	INPUT	LREAL	-	Reserviert	
DynamicAdaption	INPUT	DINT	-1	Dynamikadaption	
				< 0	Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierte Einstellung wird verwendet. (<TO>.DynamicDefaults.DynamicAdaption)
				0	Ohne Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Active	OUTPUT	BOOL	FALSE	TRUE	Die Sollwerte werden berechnet.
				FALSE	Bei "Busy" = TRUE: Der Auftrag wartet. (Typisch: Ein vorhergehender Auftrag ist noch aktiv.)
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
RemainingDistance	OUTPUT	LREAL	-1.0	Restwegstrecke des aktuellen Auftrags	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

MC_MoveCircularRelative: Funktionsdiagramm V4

Funktionsdiagramm: Kinematik mit zirkularer Bahnbewegung relativ positionieren

Im Funktionsdiagramm werden die Abschnitte A1 und A2 der folgenden Kinematikbewegung betrachtet:

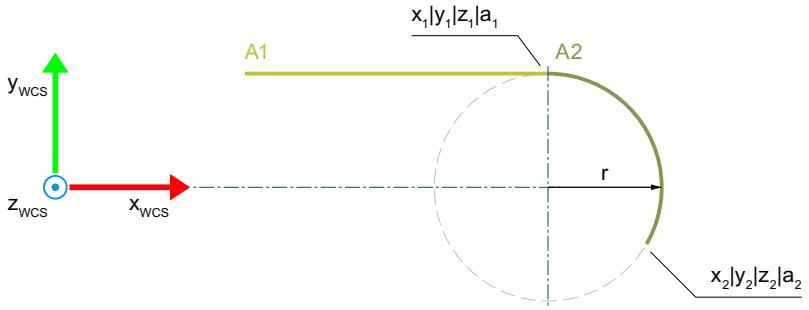

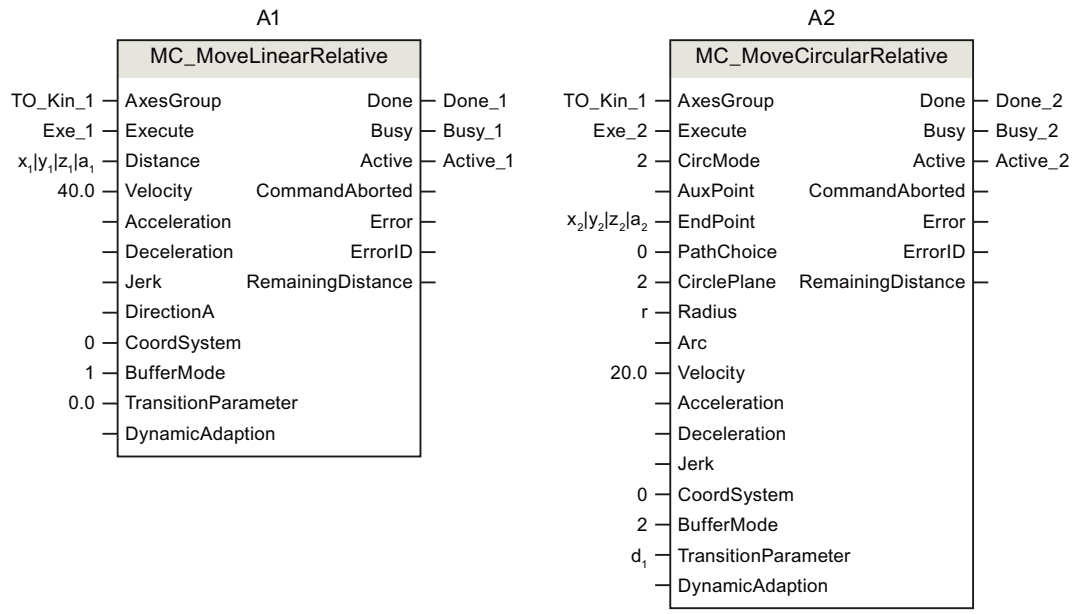

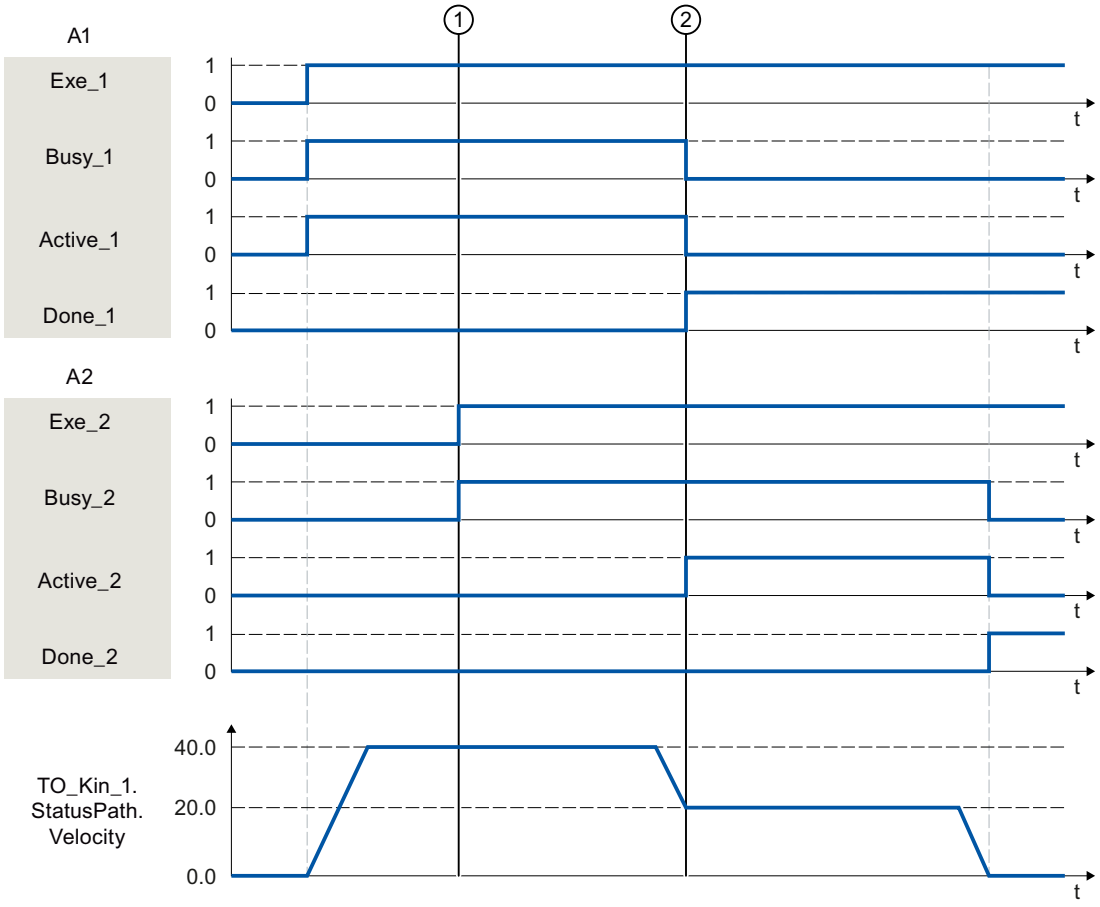

Eine Kinematik wird über einen "MC_MoveLinearRelative"-Auftrag (A1) verfahren.

Zum Zeitpunkt ① wird ein "MC_MoveCircularRelative"-Auftrag (A2) angestoßen. Da der "MC_MoveCircularRelative"-Auftrag nicht ablösend wirkt, reiht sich der Auftrag A2 in die Auftragskette ein.

Zum Zeitpunkt ② wird der Abschluss des Auftrags A1 wird über "Done_1" gemeldet und der Auftrag A2 gestartet. Da für den Auftrag A2 "BufferMode = 2" gesetzt ist, wird der Bewegungsübergang mit der niedrigeren Geschwindigkeit beider Aufträge und dem Überschleifabstand d_1 überschliffen. Die zirkulare Bewegung wird mit "CircMode = 2" über den Radius r und den Endpunkt ($x_2|y_2|z_2$) bestimmt. Sobald die Zielposition erreicht ist, wird der Abschluss des Auftrags A2 über "Done_2" gemeldet.

Zonen

MC_DefineWorkspaceZone V4

MC_DefineWorkspaceZone: Arbeitsraumzone definieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_DefineWorkspaceZone" definieren Sie eine Arbeitsraumzone bezogen auf das Welt- oder ein Objektkoordinatensystem. Dabei werden die unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Zonen" definierten Zonen (<TO>.WorkspaceZone[1..10]) nicht verändert und stehen nach einem Restart des Technologieobjekts wieder zur Verfügung. Die Variable "<TO>.StatusWorkspaceZone" des Technologieobjekt-Datenbausteins enthält die aktuell wirksamen Arbeitsraumzonen.

Der "MC_DefineWorkspaceZone"-Auftrag reiht sich in die Auftragskette am Technologieobjekt Kinematik ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Mit den Parametern "GeometryType" und "GeometryParameter" legen Sie den Zonengeometrietyp und die Zonengröße fest. Über den Parameter "ZoneType" definieren Sie eine Arbeitsraumzone als Arbeitszone, Sperrzone oder Meldezone. Sie können bis zu zehn Arbeitsraumzonen definieren. Während von den definierten Sperrzonen und Meldezonen mehrere gleichzeitig aktiv sein können, kann von den definierten Arbeitszonen immer nur eine aktiv sein.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

- Ein "MC_DefineWorkspaceZone"-Auftrag wird durch einen "MC_GroupStop"-Auftrag abgebrochen.
- Ein neuer "MC_DefineWorkspaceZone"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_DefineWorkspaceZone":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Der Auftrag wird in die Auftragskette übernommen.
ZoneType	INPUT	DINT	0	Zonentyp	
				0	Sperrzone
				1	Arbeitszone
				2	Meldezone
ZoneNumber	INPUT	DINT	1	Zonennummer	
				1	Zone 1
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10
ReferenceSystem	INPUT	DINT	0	Bezugssystem	
				0	Weltkoordinatensystem (WCS)
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])
Frame	INPUT	TO_Struct_Kinematics_Frame	-	Verschiebung des Zonennullpunkts zum Bezugssystem	
GeometryType	INPUT	DINT	0	Zonengeometrietyp	
				0	Quader
				1	Kugel
				2	Zylinder
GeometryParameter	INPUT	ARRAY [1..3] OF LREAL	-	Geometrische Parameter	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
GeometryParameter[1]	INPUT	LREAL	0.0	Länge x	
				Bei "GeometryType" = 1 oder 2: Radius	
GeometryParameter[2]	INPUT	LREAL	0.0	Länge y	
				Bei "GeometryType" = 2: Höhe	
GeometryParameter[3]	INPUT	LREAL	0.0	Länge z	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_SetWorkspaceZoneActive: Arbeitsraumzone aktivieren V4 (Seite 3382)

MC_SetWorkspaceZoneInactive: Arbeitsraumzone deaktivieren V4 (Seite 3384)

MC_DefineKinematicsZone V4

MC_DefineKinematicsZone: Kinematikzone definieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_DefineKinematicsZone" definieren Sie eine Kinematikzone bezogen auf das Werkzeug- oder Flanschkoordinatensystem. Dabei werden die unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Zonen" definierten Zonen (<TO>.KinematicsZone[2..10]) nicht verändert und stehen nach einem Restart des Technologieobjekts wieder zur Verfügung. Die Variable "<TO>.StatusKinematicsZone" des Technologieobjekt-Datenbausteins enthält die aktuell wirksamen Kinematikzonen.

Der "MC_DefineKinematicsZone"-Auftrag reiht sich in die Auftragskette ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Mit den Parametern "GeometryType" und "GeometryParameter" legen Sie den Zonengeometrietyp und die Größe fest. Sie können bis zu neun Kinematikzonen definieren. Die Kinematikzone 1 ist der Werkzeugarbeitspunkt (TCP) und kann nicht geändert werden.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

- Ein "MC_DefineKinematicsZone"-Auftrag wird durch einen "MC_GroupStop"-Auftrag abgebrochen.
- Ein neuer "MC_DefineKinematicsZone"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_DefineKinematicsZone":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Der Auftrag wird in die Auftragskette übernommen.
ZoneNumber	INPUT	DINT	2	Zonennummer	
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10
ReferenceSystem	INPUT	DINT	0	Bezugssystem	
				0	Flanschkoordinatensystem (FCS)
				1	Werkzeugkoordinatensystem (TCS)
Frame	INPUT	TO_Struct_Kinematics_Frame	-	Verschiebung des Zonennullpunkts zum Bezugssystem	

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
GeometryType	INPUT	DINT	0	Zonengeometrietyp	
				0	Quader
				1	Kugel
				2	Zylinder
GeometryParameter	INPUT	ARRAY [1..3] OF LREAL	-	Geometrische Parameter	
GeometryParameter[1]	INPUT	LREAL	0.0	Länge x	
				Bei "GeometryType" = 1 oder 2: Radius	
GeometryParameter[2]	INPUT	LREAL	0.0	Länge y	
				Bei "GeometryType" = 2: Höhe	
GeometryParameter[3]	INPUT	LREAL	0.0	Länge z	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_SetKinematicsZoneActive: Kinematikzone aktivieren V4 (Seite 3386)

MC_SetKinematicsZoneInactive: Kinematikzone deaktivieren V4 (Seite 3387)

MC_SetWorkspaceZoneActive V4

MC_SetWorkspaceZoneActive: Arbeitsraumzone aktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetWorkspaceZoneActive" aktivieren Sie eine Arbeitsraumzone, die Sie unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Zonen" oder über einen "MC_DefineWorkspaceZone"-Auftrag definiert haben. Mit dem Parameter "ZoneNumber" geben Sie die Nummer der zu aktivierenden Zone an.

Der "MC_SetWorkspaceZoneActive"-Auftrag reiht sich in die Auftragskette ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Die Variablen "<TO>.StatusWorkspaceZone[1..10].Active" des Technologieobjekt-Datenbausteins enthalten den aktuellen Aktivierungsstatus der Zonen. Während von den definierten Sperrzonen und Meldezonen mehrere gleichzeitig aktiv sein können, kann von den definierten Arbeitszonen immer nur eine aktiv sein.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die zu aktivierende Zone ist definiert.

Ablöseverhalten

- Ein "MC_SetWorkspaceZoneActive"-Auftrag wird durch einen "MC_GroupStop"-Auftrag abgebrochen.
- Ein neuer "MC_SetWorkspaceZoneActive"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetWorkspaceZoneActive":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
ZoneNumber	INPUT	DINT	1	Zonennummer	
				1	Zone 1
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_DefineWorkspaceZone: Arbeitsraumzone definieren V4 (Seite 3378)

MC_SetWorkspaceZoneInactive: Arbeitsraumzone deaktivieren V4 (Seite 3384)

MC_SetWorkspaceZoneInactive V4

MC_SetWorkspaceZoneInactive: Arbeitsraumzone deaktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetWorkspaceZoneInactive" deaktivieren Sie eine aktive Arbeitsraumzone. Über den Parameter "Mode" können Sie eine bestimmte Zone, alle Zonen eines Typs oder alle Zonen deaktivieren.

Der "MC_SetWorkspaceZoneInactive"-Auftrag reiht sich in die Auftragskette ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Die Variablen "<TO>.StatusWorkspaceZone[1..10].Active" des Technologieobjekt-Datenbausteins enthalten den aktuellen Aktivierungsstatus der Zonen.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die zu deaktivierende Zone ist definiert.

Ablöseverhalten

- Ein "MC_SetWorkspaceZoneInactive"-Auftrag wird durch einen "MC_GroupStop"-Auftrag abgebrochen.
- Ein neuer "MC_SetWorkspaceZoneInactive"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetWorkspaceZoneInactive":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
ZoneNumber	INPUT	DINT	1	Zonennummer	
				1	Zone 1
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10
Mode	INPUT	DINT	0	Zonen eines Zonentyps inaktiv setzen	
				0	Eine bestimmte Zone inaktiv setzen
				1	Alle Arbeitsraumzonen inaktiv setzen
				2	Alle Sperrzonen inaktiv setzen
				3	Alle Meldezonen inaktiv setzen
				4	Die Arbeitszone inaktiv setzen
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_DefineWorkspaceZone: Arbeitsraumzone definieren V4 (Seite 3378)

MC_SetWorkspaceZoneActive: Arbeitsraumzone aktivieren V4 (Seite 3382)

MC_SetKinematicsZoneActive V4

MC_SetKinematicsZoneActive: Kinematikzone aktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetKinematicsZoneActive" aktivieren Sie die Zonenüberwachung für eine Kinematikzone, die Sie unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Zonen" oder über einen "MC_DefineKinematicsZone"-Auftrag definiert haben. Mit dem Parameter "ZoneNumber" geben Sie die Nummer der zu aktivierenden Kinematikzone an.

Der "MC_SetKinematicsZoneActive"-Auftrag reiht sich in die Auftragskette ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Die Variablen "<TO>.StatusKinematicsZone[2..10].Active" des Technologieobjekt-Datenbausteins enthalten den aktuellen Aktivierungsstatus der Kinematikzonen.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die zu aktivierende Kinematikzone ist definiert.

Ablöseverhalten

- Ein "MC_SetKinematicsZoneActive"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_SetKinematicsZoneActive"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetKinematicsZoneActive":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
ZoneNumber	INPUT	DINT	2	Zonennummer	
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_DefineKinematicsZone: Kinematikzone definieren V4 (Seite 3380)

MC_SetKinematicsZoneInactive: Kinematikzone deaktivieren V4 (Seite 3387)

MC_SetKinematicsZoneInactive V4

MC_SetKinematicsZoneInactive: Kinematikzone deaktivieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetKinematicsZoneInactive" deaktivieren Sie eine aktive Kinematikzone. Über den Parameter "Mode" können Sie eine bestimmte Kinematikzone oder alle Kinematikzonen deaktivieren.

Der "MC_SetKinematicsZoneInactive"-Auftrag reiht sich in die Auftragskette ein und ist somit für nachfolgende Bewegungsaufträge wirksam.

Die Variablen "<TO>.StatusKinematicsZone[2..10].Active" des Technologieobjekt-Datenbausteins enthalten den aktuellen Aktivierungsstatus der Kinematikzonen.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die zu deaktivierende Kinematikzone ist definiert.

Ablöseverhalten

- Ein "MC_SetKinematicsZoneInactive"-Auftrag wird durch keinen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_SetKinematicsZoneInactive"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetKinematicsZoneInactive":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
ZoneNumber	INPUT	DINT	2	Zonennummer	
				2	Zone 2
				3	Zone 3
				4	Zone 4
				5	Zone 5
				6	Zone 6
				7	Zone 7
				8	Zone 8
				9	Zone 9
				10	Zone 10

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	DINT	0	Zonen inaktiv setzen	
				0	Eine bestimmte Zone inaktiv setzen
				1	Alle Zonen inaktiv setzen
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

MC_DefineKinematicsZone: Kinematikzone definieren V4 (Seite 3380)

MC_SetKinematicsZoneActive: Kinematikzone aktivieren V4 (Seite 3386)

Werkzeuge

MC_DefineTool V4

MC_DefineTool: Werkzeug neu definieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_DefineTool" definieren Sie den Tool-Frame des Werkzeugs 1 neu. Dabei werden die im System hinterlegten Startwerte nicht überschrieben. Das Werkzeug 1 ist standardmäßig aktiv.

Der "MC_DefineTool"-Auftrag reiht sich nicht in die Auftragskette am Technologieobjekt Kinematik ein und ist somit direkt wirksam. Der "MC_DefineTool"-Auftrag kann nur ausgeführt werden, wenn sich die Kinematik im Stillstand befindet.

Die parametrierbaren Koordinaten sind abhängig vom eingesetzten Kinematiktyp:

Kinematiktyp		Parametrierbare Koordinaten
2D	ohne Orientierung	x, z[1)]
	mit Orientierung	z, a[2)]

Kinematiktyp		Parametrierbare Koordinaten
3D	ohne Orientierung	x, y, z[3)]
	mit Orientierung	x, y, z, a

1) Die Parameter "y" und "a" sind mit dem Wert "0.0" vordefiniert.

2) Die Parameter "x" und "y" sind mit dem Wert "0.0" vordefiniert.

3) Der Parameter "a" sind mit dem Wert "0.0" vordefiniert.

Die folgenden Variablen des Technologieobjekt-Datenbausteins enthalten die aktuellen Koordinaten des Tool-Frame 1:

- <TO>.StatusTool.Frame[1].x
- <TO>.StatusTool.Frame[1].y
- <TO>.StatusTool.Frame[1].z
- <TO>.StatusTool.Frame[1].a

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die Kinematik befindet sich im Stillstand.
- Die Kinematik befindet sich nicht im Zustand "Interrupted".
- Keine Kinematikbewegung ist aktiv.

Ablöseverhalten

- Ein "MC_DefineTool"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein neuer "MC_DefineTool"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_DefineTool":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Frame	INPUT	TO_Struct_Kinematics_KinematicsFrame	-	Koordinaten in Bezug auf das FCS	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_SetTool V4

MC_SetTool: Aktives Werkzeug wechseln V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetTool" setzen Sie ein Werkzeug aktiv. Mit dem Parameter "ToolNumber" geben Sie die Werkzeugnummer an. Der "MC_SetTool"-Auftrag kann nur ausgeführt werden, wenn sich die Kinematik im Stillstand befindet. Das Werkzeug 1 ist standardmäßig aktiv.

Die Variable "<TO>.StatusTool.ActiveTool" des Technologieobjekt-Datenbausteins enthält die Werkzeugnummer des aktuell aktiven Werkzeugs.

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.
- Die Kinematik befindet sich im Stillstand.
- Die Kinematik befindet sich nicht im Zustand "Interrupted".
- Keine Kinematikbewegung ist aktiv.

Ablöseverhalten

- Ein "MC_SetTool"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein neuer "MC_SetTool"-Auftrag bricht keinen laufenden Motion Control-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetTool":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Start des Auftrags mit steigender Flanke
ToolNumber	INPUT	DINT	1	Nummer des wirksam zu setzenden Werkzeugs	
				1	Werkzeug 1
				2	Werkzeug 2
				3	Werkzeug 3
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Koordinatensysteme

MC_SetOcsFrame V4

MC_SetOcsFrame: Objektkoordinatensysteme neu definieren V4

Beschreibung

Mit der Motion Control-Anweisung "MC_SetOcsFrame" definieren Sie die Lage eines Objektkoordinatensystems (OCS) in Bezug auf das Weltkoordinatensystem (WCS). Dabei werden die im Technologieobjekt-Datenbaustein hinterlegten Startwerte nicht überschrieben.

Der "MC_SetOcsFrame"-Auftrag reiht sich in die Auftragskette ein und ist somit nur für nachfolgende Bewegungsaufträge wirksam.

Die folgenden Variablen des Technologieobjekt-Datenbausteins enthalten die aktuellen Koordinaten der Objektkoordinatensysteme:

- <TO>.StatusOcsFrame[1..3].x
- <TO>.StatusOcsFrame[1..3].y
- <TO>.StatusOcsFrame[1..3].z
- <TO>.StatusOcsFrame[1..3].a
- <TO>.StatusOcsFrame[1..3].b
- <TO>.StatusOcsFrame[1..3].c

Anwendbar auf

- Kinematik

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die verschalteten Achsen sind freigegeben.
- An keiner der verschalteten Achsen ist ein Einzelachsauftrag (z. B. "MC_MoveVelocity") aktiv.

Ablöseverhalten

Das Ablöseverhalten für "MC_SetOcsFrame"-Aufträge ist im Kapitel "Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)" beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetOcsFrame":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
AxesGroup	INPUT	TO_Kinematics	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	TRUE	Der Auftrag wird in die Auftragskette übernommen.
Frame	INPUT	TO_Struct_Kinematics_Frame	-	Koordinaten in Bezug auf das WCS	
OcsNumber	INPUT	DINT	1	Objektkoordinatensystem	
				1	Objektkoordinatensystem 1 (OCS[1])
				2	Objektkoordinatensystem 2 (OCS[2])
				3	Objektkoordinatensystem 3 (OCS[3])
Done	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist abgeschlossen. Die neuen Werte sind für die Anzeige und die Bewegung wirksam.
Busy	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten V4: Kinematikbewegungsaufträge (Seite 3396)

Ablöseverhalten von Motion Control-Aufträgen V4

Ablöseverhalten V4: Referenzier- und Bewegungsaufträge

Folgende Tabelle zeigt, wie ein neuer Motion Control-Auftrag auf laufende Referenzier- und Bewegungsaufträge wirkt:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_Home "Mode" = 2, 8, 10	MC_Home "Mode" = 3, 5	MC_Halt MC_MoveAbsolute MC_MoveRelative MC_MoveVelocity MC_MoveJog	MC_MoveSuperimposed	MC_MotionInVelocity MC_MotionInPosition
MC_Home "Mode" = 3, 5	A	A	A	A	A
MC_Home "Mode" = 9	A	-	-	-	-
MC_Halt MC_MoveAbsolute MC_MoveRelative MC_MoveVelocity MC_MoveJog MC_MotionInVelocity MC_MotionInPosition	-	A	A	A	A
MC_MoveSuperimposed	-	-	-	A	-
MC_GearIn	-	A	A	A	-
MC_GearInPos wartend[1)]	-	-	-	-	-
MC_GearInPos aktiv[2)]	-	A	A	A	-

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_Home "Mode" = 2, 8, 10	MC_Home "Mode" = 3, 5	MC_Halt MC_MoveAbsolute MC_MoveRelative MC_MoveVelocity MC_MoveJog	MC_MoveSuper-imposed	MC_MotionIn-Velocity MC_MotionIn-Position
MC_CamIn wartend[1)]	-	-	-	-	-
MC_CamIn aktiv[2)]	-	A	A	A	-

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen.

- Keine Auswirkung. Der laufende Auftrag wird weiterhin ausgeführt.

1) "Busy" = TRUE, "StartSync" = FALSE, "InSync" = FALSE

2) "Busy" = TRUE, "StartSync" oder "InSync" = TRUE

Ablöseverhalten V4: Gleichlaufaufträge

Folgende Tabelle zeigt, wie ein neuer Motion Control-Auftrag zur Bewegung der Achse auf laufende Gleichlaufaufträge wirkt:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_GearIn	MC_GearIn-Pos wartend[1)]	MC_GearIn-Pos aktiv[2)]	MC_Phasing-Absolute MC_Phasing-Relative	MC_CamIn wartend[1)]	MC_CamIn aktiv[2)]
MC_Home "Mode" = 3, 5	A	-	-	-	-	-
MC_Halt	A	-	A	A	-	A
MC_MoveAbsolute MC_MoveRelative MC_MoveVelocity MC_MoveJog	A	-	A	A	-	A
MC_MotionInVelocity MC_MotionInPosition	A	A	A	-	A	A
MC_MoveSuperimposed	-	-	-	-	-	-
MC_GearIn	A	A	A	A	A	A
MC_GearInPos wartend[1)]	-	A	-	-	A	-
MC_GearInPos aktiv[2)]	A	A	A	A	A	A
MC_PhasingAbsolute MC_PhasingRelative	-	-	-	A	-	-
MC_CamIn wartend[1)]	-	A	-	-	A	-
MC_CamIn aktiv[2)]	A	A	A	A	A	A

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen.

- Keine Auswirkung. Der laufende Auftrag wird weiterhin ausgeführt.

1) Ein wartender Gleichlaufauftrag ("Busy" = TRUE, "StartSync" = FALSE, "InSync" = FALSE) bricht keine laufenden Aufträge ab. Ein wartender Gleichlaufauftrag kann nur durch einen weiteren Gleichlaufauftrag auf die gleiche Folgeachse abgelöst werden. Ein Abbruch ist durch den "MC_Power" möglich.

2) Der Status "Busy" = TRUE, "StartSync" oder "InSync" = TRUE entspricht einem aktiven Gleichlauf.

Ablöseverhalten V4: Messtasteraufträge

Folgende Tabelle zeigt, durch welche neuen Motion Control-Aufträge laufende Messtasteraufträge abgelöst werden:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_MeasuringInput	MC_MeasuringInputCyclic
MC_Home "Mode" = 2, 3, 5, 8, 9, 10	A	A
MC_Home "Mode" = 0, 1, 6, 7	-	-
MC_MeasuringInput MC_MeasuringInputCyclic MC_AbortMeasuringInput	A	A

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen.

- Keine Auswirkung. Der laufende Auftrag wird weiterhin ausgeführt.

Ablöseverhalten V4: Kinematikbewegungsaufträge

Einzelachsaufträge werden durch Kinematikaufträge nicht abgelöst.

Folgende Tabelle zeigt, wie ein neuer Motion Control-Auftrag auf laufende Kinematikbewegungsaufträge wirkt:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircularAbsolute MC_MoveCircularRelative MC_DefineWorkspaceZone MC_DefineKinematikZone MC_SetWorkspaceZoneActive MC_SetWorkspaceZoneInactive MC_SetKinematicsZoneActive MC_SetKinematicsZoneInactive MC_SetOcsFrame	MC_GroupInterrupt	MC_GroupStop
MC_Home MC_MoveSuperimposed	N	N	N
MC_Halt MC_MoveAbsolute MC_MoveRelative MC_MoveVelocity MC_MoveJog MC_GearIn MC_GearInPos MC_CamIn MC_MotionInVelocity MC_MotionInPosition MC_GroupStop	A	A	A
MC_GroupInterrupt MC_GroupContinue	B	A	N
MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircularAbsolute MC_MoveCircularRelative MC_DefineWorkspaceZone MC_DefineKinematikZone MC_SetWorkspaceZoneActive MC_SetWorkspaceZoneInactive MC_SetKinematicsZoneActive MC_SetKinematicsZoneInactive MC_SetOcsFrame	-	-	N

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen.

B Der laufende Auftrag wird unterbrochen bzw. fortgesetzt.

N Nicht erlaubt. Der laufende Auftrag wird weiterhin ausgeführt. Der neue Auftrag wird abgelehnt.

- Keine Auswirkung. Der laufende Auftrag wird weiterhin ausgeführt. Der neue Auftrag reiht sich in die Auftragskette ein.

S7-1500 Motion Control V3

MC_Power V3

MC_Power: Technologieobjekt freigeben, sperren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_Power" wird ein Technologieobjekt freigegeben bzw. gesperrt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.

Ablöseverhalten

- Ein "MC_Power"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein "MC_Power"-Auftrag mit dem Parameter "Enable" = TRUE gibt ein Technologieobjekt frei und bricht somit keine anderen Motion Control-Anweisungen ab.
- Mit dem Sperren des Technologieobjekts (Parameter "Enable" = FALSE) werden alle Bewegungsaufträge am zugehörigen Technologieobjekt entsprechend dem gewählten "StopMode" abgebrochen. Dieser Vorgang kann vom Anwender nicht abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Power":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Axis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Das Technologieobjekt wird freigegeben.
				FALSE	Das Technologieobjekt wird gesperrt. Alle laufenden Aufträge am Technologieobjekt werden entsprechend dem parametrierten "StopMode" abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
StartMode	INPUT	DINT	1	0	Positionierachse/Gleichlaufachse nicht lagegeregelt freigeben
				1	Positionierachse/Gleichlaufachse lagegeregelt freigeben
				Der Parameter wirkt initial beim Freigeben der Positionierachse (Enable wechselt von FALSE nach TRUE) und beim Freigeben nach dem erfolgreichen Quittieren eines Alarms, der zum Sperren der Achse geführt hat. Beim Einsatz einer Drehzahlachse oder eines Externen Gebers wird der Parameter ignoriert.	
StopMode	INPUT	INT	0	Für Technologieobjekt Externer Geber nicht relevant. Wenn Sie ein Technologieobjekt mit einer fallenden Flanke am Parameter "Enable" sperren, bremst die Achse gemäß dem gewählten "StopMode" ab.	
				0	Notstopp Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Notstopp-Rampe" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicDefaults.EmergencyDeceleration)
				1	Sofort-Aus Beim Sperren eines Technologieobjekts wird der Sollwert Null ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
				2	Stopp mit maximalen Dynamikwerten Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicLimits.MaxDeceleration; <TO>.DynamicLimits.MaxJerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Status	OUTPUT	BOOL	FALSE	Freigabestatus des Technologieobjekts	
				FALSE	Gesperrt - Eine Positionier-, Gleichlauf- oder Drehzahlachse nimmt keine Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind nicht aktiv. - Die Istwerte des Technologieobjekts werden nicht auf Gültigkeit überprüft.
				TRUE	Freigegeben - Eine freigegebene Positionier-, Gleichlauf- oder Drehzahlachse nimmt Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind aktiv. - Die Istwerte des Technologieobjekts sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Freigeben von Technologieobjekten

Setzen Sie zum Freigeben eines Technologieobjekts den Parameter "Enable" auf TRUE.

Wenn der Parameter "Status" den Wert TRUE zeigt, ist das Technologieobjekt freigegeben.

Wenn eine Achse beim Freigeben des Technologieobjekts in Bewegung ist (Istgeschwindigkeit vorhanden), wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung auf Sollwert Null abgebremst (<TO>.DynamicLimits.MaxDeceleration). Diese Bremsrampe ist von Bewegungsaufträgen ablösbar.

Hinweis

Automatische Freigabe nach Quittierung eines Technologiealarms

Wenn das Technologieobjekt aufgrund eines Technologiealarms gesperrt wird, so wird es nach dem Beseitigen der Ursache und Quittieren des Alarms automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Parameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Sperren von Technologieobjekten

Setzen Sie zum Sperren eines Technologieobjekts den Parameter "Enable" auf FALSE.

Wenn eine Achse in Bewegung ist, wird Sie gemäß dem gewählten "StopMode" abgebremst und zum Stillstand gebracht.

Wenn die Parameter "Busy" und "Status" den Wert FALSE zeigen, ist das Sperren des Technologieobjekts abgeschlossen.

Antriebsanbindung über PROFIdrive

Bei der Ankopplung eines Antriebs über PROFIdrive werden Sollwert, Freigabe und Antriebsstatus über das PROFIdrive-Telegramm übertragen.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird das Technologieobjekt freigegeben. Der Antrieb wird gemäß PROFIdrive-Norm aktiviert.
  Wenn die Variable <TO>.StatusDrive.InOperation den Wert TRUE zeigt, ist der Antrieb bereit, Sollwerte auszuführen. Der Parameter "Status" wird auf den Wert TRUE gesetzt.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Der Antrieb wird gemäß PROFIdrive-Norm deaktiviert.

Analoge Antriebsanbindung

Der Sollwert wird über einen Analogausgang ausgegeben. Optional kann ein Freigabesignal über Digitalausgang (<TO>.Actor.Interface.EnableDriveOutput) und ein Bereitschaftssignal über Digitaleingang (<TO>.Actor.Interface.DriveReadyInput) konfiguriert werden.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird der Freigabe-Ausgang ("Enable drive output") gesetzt.
  Wenn der Antrieb das Bereitschaftssignal über den Bereit-Eingang ("Drive ready input") zurückmeldet, wird der Parameter "Status" und die Variable des Technologieobjekts <TO>.StatusDrive.InOperation auf TRUE gesetzt und der Sollwert auf den Analogausgang geschaltet.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Mit dem Erreichen von Sollwert Null wird der Freigabe-Ausgang auf FALSE gesetzt.

Weitere Informationen

Weiterführende Informationen zum Freigeben und Sperren von Technologieobjekten und Antrieben finden Sie im Anhang Funktionsdiagramme MC_Power.

MC_Power: Funktionsdiagramm V3

Funktionsdiagramm: Freigeben eines Technologieobjekts und Beispiel für Alarmreaktion

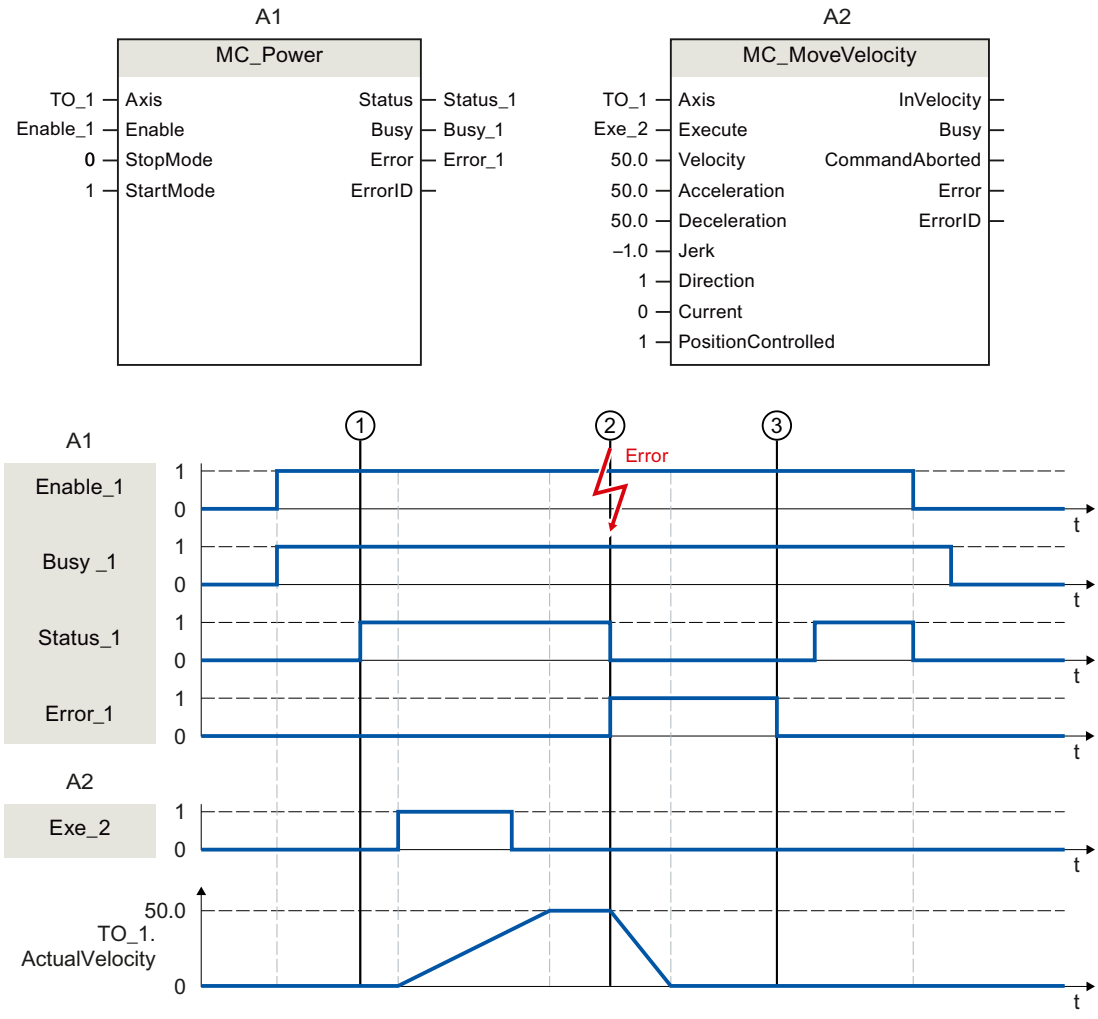

Ein Technologieobjekt wird mit "Enable_1" = TRUE freigegeben. Die erfolgreiche Freigabe kann an "Status_1" zum Zeitpunkt ① abgelesen werden. Anschließend wird die Achse mit einem "MC_MoveVelocity"-Auftrag (A2) verfahren. Der Geschwindigkeitsverlauf der Achse kann an "Velocity Axis_1" abgelesen werden.

Zum Zeitpunkt ② tritt ein Fehler am Technologieobjekt auf, der das Sperren des Technologieobjekts zur Folge hat (Alarmreaktion: Freigabe wegnehmen). Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Mit dem Sperren des Technologieobjekts wird "Status_1" rückgesetzt. Da die Achse nicht über "Enable_1" = FALSE gesperrt wurde, ist der gewählte "StopMode" nicht relevant. Die Fehlerursache wird beseitigt und der Alarm wird zum Zeitpunkt ③ quittiert.

Da "Enable_1" weiterhin gesetzt ist, wird das Technologieobjekt wieder freigegeben. Die erfolgreiche Freigabe kann an "Status_1" abgelesen werden. Abschließend wird das Technologieobjekt über "Enable_1" = FALSE gesperrt.

MC_Reset V3

MC_Reset: Alarme quittieren, Restart Technologieobjekt V3

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" quittieren Sie alle Technologie-Alarme, die im Anwenderprogramm quittierbar sind. Mit der Quittierung werden auch die "Error"- und "Warning"-Bits im Technologie-Datenbaustein zurückgesetzt.

Mit der Motion Control-Anweisung "MC_Reset" mit "Restart" = TRUE starten Sie das Neuinitialisieren (Restart) von Technologieobjekten. Mit dem Restart des Technologieobjekts werden neue Konfigurationsdaten im Technologie-Datenbaustein übernommen.

Anwendbar auf

- Alle Technologieobjekte

Voraussetzung

- Technologieobjekte Positionier-/Gleichlaufachse, Drehzahlachse und Externer Geber:
  Für einen Restart muss das Technologieobjekt gesperrt sein.
  ("MC_Power.Status" = FALSE und "MC_Power.Busy" = FALSE)

Ablöseverhalten

- Parameter "Restart" = FALSE:
  Der "MC_Reset"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.
- Parameter "Restart" = TRUE:
  Die Bearbeitung der Anweisung "MC_Reset" mit Parameter "Restart" = TRUE kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Reset":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_Object	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Restart	INPUT	BOOL	FALSE	TRUE	"Restart" Neuinitialisierung des Technologieobjektes und Quittierung anstehender Technologie-Alarme. Das Technologieobjekt wird mit den konfigurierten Startwerten neu initialisiert.
				FALSE	Quittierung anstehender Technologie-Alarme.
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert. Restart wurde ausgeführt.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Comman-dAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "Er-rorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Technologie-Alarme quittieren

Gehen Sie zum Quittieren von Technologie-Alarmen folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen Sie den Parameter "Restart" = FALSE.
3. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so wurde der Fehler quittiert.

Hinweis

Quittieren mit "Restart" = FALSE

Setzen Sie "Restart" = FALSE, wenn nur die Technologie-Alarme quittiert werden sollen. Während eines Restarts kann das Technologieobjekt nicht verwendet werden.

Restart eines Technologieobjektes

Gehen Sie zum Restart eines Technologieobjektes wie folgt vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen Sie den Parameter "Restart" = TRUE.
3. Führen Sie den Restart durch eine steigende Flanke am Parameter "Execute" aus.

Wenn der Parameter "Done" den Wert TRUE zeigt, ist der Restart des Technologieobjektes abgeschlossen.

Weitere Informationen zum Restart finden Sie im Kapitel Restart von Technologieobjekten.

MC_Home V3

MC_Home: Technologieobjekt referenzieren, Referenzpunkt setzen V3

Beschreibung

Mit der Motion Control-Anweisung "MC_Home" stellen Sie den Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung her. Der Positionsistwert am Technologieobjekt wird dabei einer Referenzmarke zugeordnet. Diese Referenzmarke repräsentiert eine bekannte mechanische Position.

Der Referenziervorgang erfolgt gemäß der am Parameter "Mode" gewählten Betriebsart und der Konfiguration unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren".

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und ≥ V2.0 finden Sie im Kapitel Versionsübersicht.

Für die Dynamikwerte Beschleunigung, Verzögerung und Ruck werden die voreingestellten Werte unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik-Voreinstellung" verwendet.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Externer Geber

Die folgende Tabelle zeigt, welche Betriebsarten mit den jeweiligen Technologieobjekten möglich sind:

Betriebsart	Positionierachse / Gleichlaufachse mit Inkrementalgeber	Positionierachse / Gleichlaufachse mit Absolutwertgeber	Externer Geber inkrementell	Externer Geber absolut
Aktives Referenzieren ("Mode" = 3, 5)	X	-	-	-
Passives Referenzieren ("Mode" = 2, 8,10)	X	-	X	-
Setzen der Istposition ("Mode" = 0)	X	X	X	X
Relative Verschiebung der Istposition ("Mode" = 1)	X	X	X	X
Absolutwertgeberjustage ("Mode" = 6, 7)	-	X	-	X

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- "Mode" = 2, 3, 5, 8, 10
  Das Technologieobjekt ist freigegeben.
- "Mode" = 0, 1, 6, 7, 8
  Die Geberistwerte sind gültig (<TO>.StatusSensor[n].State = 2).
- "Mode" = 6, 7
  Die Achse befindet sich im lagegeregelten Betrieb.

Ablöseverhalten

Das Ablöseverhalten für "MC_Home"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Home":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Der angegebene Wert wird dem gewähltem "Mode" entsprechend verwendet.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	INT	0	Betriebsart	
				0	Direktes Referenzieren (Absolut) Die aktuelle Position des Technologieobjektes wird auf den Wert des Parameters "Position" gesetzt.
				1	Direktes Referenzieren (Relativ) Die aktuelle Position des Technologieobjektes wird um den Wert des Parameters "Position" verschoben.
				2	Passives Referenzieren (ohne Rücksetzen) Funktion wie "Mode" 8 mit dem Unterschied, dass der Status "referenziert" mit dem Aktivieren der Funktion **nicht** zurückgesetzt wird.
				3	Aktives Referenzieren Das TO Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem Wert des Parameters "Position".
				4	Reserviert
				5	Aktives Referenzieren (Parameter "Position" unwirksam) Das TO Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Aktives Referenzieren" konfigurierten Referenzpunkt. (<TO>.Homing.HomePosition)
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Position wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n]. AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Position wird auf den Wert des Parameters "Position" gesetzt. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n]. AbsEncoderOffset)
				8	Passives Referenzieren

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
					Beim Erkennen der Referenzmarke wird der Istwert auf den Wert des Parameters "Position" gesetzt.
				9	Abbruch Passives Referenzieren Ein laufender Auftrag zum passiven Referenzieren wird abgebrochen.
				10	Passives Referenzieren (Parameter "Position" unwirksam) Beim Erkennen der Referenzmarke wird der Istwert auf den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Passives Referenzieren" konfigurierten Referenzpunkt gesetzt. (<TO>.Homing.HomePosition)
ReferenceMarkPosition	OUTPUT	LREAL	0.0	Anzeige der Position, an der das Technologieobjekt referenziert wurde (gültig bei "Done" = TRUE)	
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjektes wird unter folgenden Bedingungen rückgesetzt (<TO>.StatusWord.X5 (HomingDone)):

- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags mit "Mode" = 3, 5, 8, 10
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS -> NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Referenzieren eines Technologieobjekts mit "Mode" = 1..8, 10

Gehen Sie zum Referenzieren eines Technologieobjektes folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Mode" die gewünschten Referenzierfunktion an.
3. Versorgen Sie die notwendigen Parameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so ist der "MC_Home"-Auftrag gemäß dem gewählten "Mode" abgeschlossen. Der Status "Referenziert" des Technologieobjekts wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Referenziert" angezeigt (<TO>.StatusWord.X5 (HomingDone)).

Abbruch eines passiven Refenziervorgangs mit "Mode" = 9

Das Technologieobjekt wird mit "Mode" = 9 nicht referenziert. Wenn ein laufender "MC_Home"-Auftrag zum passiven Referenzieren ("Mode" = 2, 8, 10) durch einen anderen "MC_Home"-Auftrag mit "Mode" = 9 abgelöst wird, so wird der laufende Auftrag mit dem Parameter "CommandAborted" = TRUE abgebrochen. Der ablösende Auftrag mit "Mode" = 9 meldet die erfolgreiche Durchführung mit Parameter "Done" = TRUE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_Halt V3

MC_Halt: Achse anhalten V3

Beschreibung

Mit der Motion Control-Anweisung "MC_Halt" bremsen Sie eine Achse bis zum Stillstand ab.

Mit den Parametern "Jerk" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bremsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_Halt"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Halt":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
AbortAcceleration	INPUT	BOOL	FALSE	FALSE	Die aktuelle Beschleunigung beim Start des Auftrags wird über den konfigurierten Ruck abgebaut. Danach wird die Verzögerung aufgebaut.
				TRUE	Die Beschleunigung wird beim Start des Auftrags auf 0.0 gesetzt und die Verzögerung unmittelbar aufgebaut.
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit null erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Abbremsen einer Achse mit "MC_Halt"

Gehen Sie zum Abbremsen einer Achse bis zum Stillstand folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die Parameter "Deceleration", "Jerk" und "AbortAcceleration" mit den gewünschten Werten.
3. Starten Sie den "MC_Halt"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt. Der Stillstand der Achse wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Stillstand" angezeigt (<TO>.StatusWord.X7 (Standstill)).

Abbremsen einer Achse mit aktiver Kraft-/Momentenbegrenzung

Bei aktiver Kraft-/Momentenbegrenzung kann das zulässige Moment unzureichend sein, um die berechnete Bremsrampe zu verfahren. Der "MC_Halt"-Auftrag meldet dann nicht "Done" = TRUE.

Verwenden Sie zum Abbremsen einer Achse mit aktiver Kraft-/Momentenbegrenzung die Motion Control-Anweisung "MC_MoveVelocity" mit "MC_MoveVelocity.Velocity" = 0.0 und "MC_MoveVelocity.PositionControlled" = FALSE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V3 (Seite 3422)

MC_Halt: Funktionsdiagramm V3

Funktionsdiagramm: Anhalten einer Achse und ablösendes Auftragsverhalten

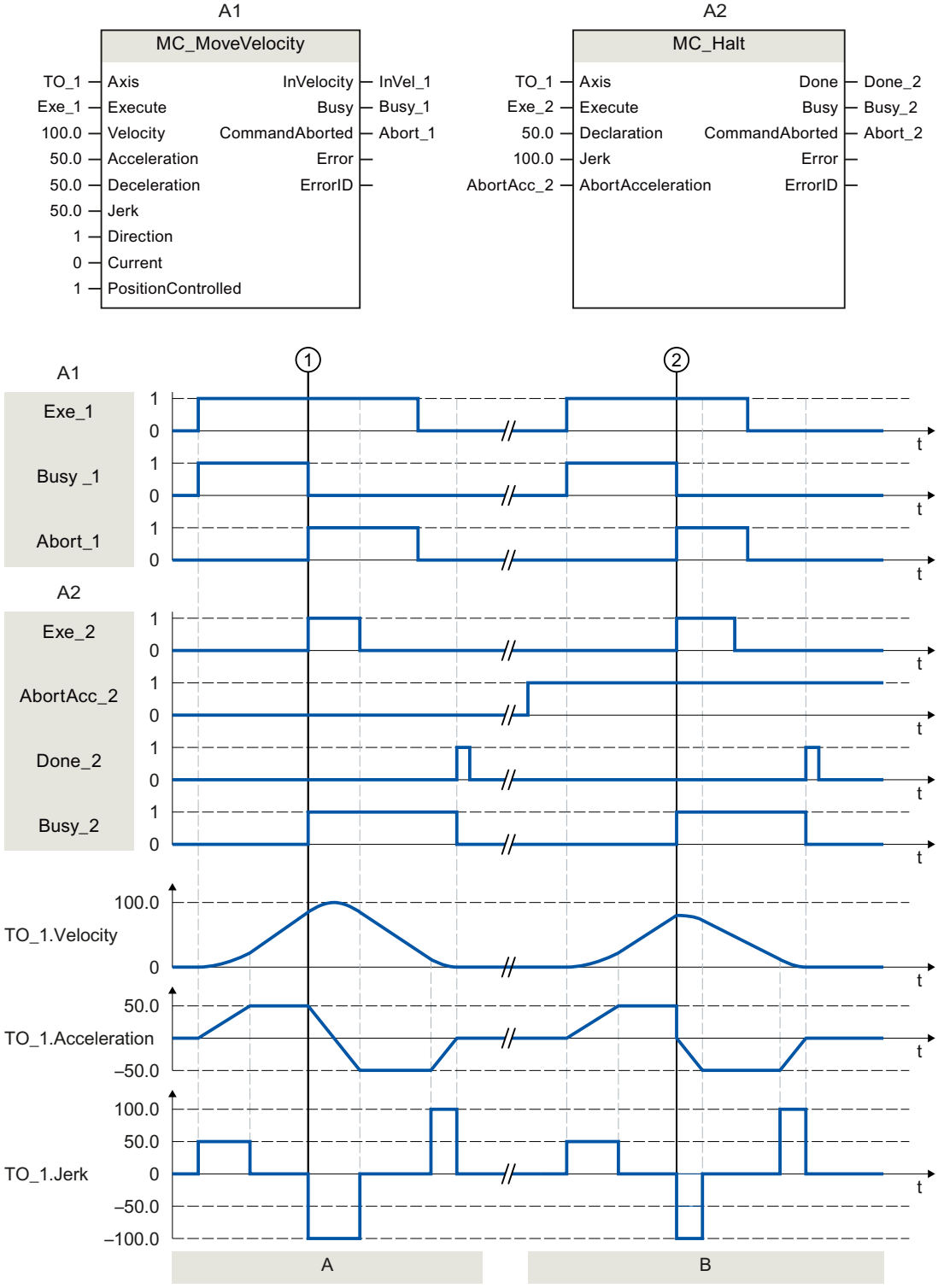

Abschnitt A	Eine Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Zum Zeitpunkt ① wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Mit "AbortAcc_2" = FALSE wird die aktuelle Beschleunigung mit dem angegebenen Ruck abgebaut. Danach wird die Verzögerung aufgebaut und die Achse bis zum Stillstand abgebremst. Der erfolgreiche Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.
Abschnitt B	Die Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Zum Zeitpunkt ② wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Mit "AbortAcc_2" = TRUE wird die aktuelle Beschleunigung unmittelbar auf null gesetzt und die Verzögerung aufgebaut. Die Achse wird bis zum Stillstand abgebremst. Der erfolgreiche Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.

MC_MoveAbsolute V3

MC_MoveAbsolute: Achse absolut positionieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveAbsolute" verfahren Sie eine Achse auf eine absolute Position.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.
- Das Technologieobjekt ist referenziert.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveAbsolute"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_Positioning-Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Absolute Zielposition
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Direction	INPUT	INT	1	Bewegungsrichtung der Achse Wird nur bei aktiviertem "Modulo" ausgewertet. "Technologieobjekt > Konfiguration > Grundparameter > Modulo aktivieren" 1 Positive Richtung 2 Negative Richtung 3 Kürzester Weg
Done	OUTPUT	BOOL	FALSE	TRUE Zielposition erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse auf eine absolute Position

Gehen Sie zum Verfahren einer Achse auf eine absolute Position folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Position" die gewünschte Zielposition an.
3. Starten Sie den "MC_MoveAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveAbsolute: Funktionsdiagramm V3

Funktionsdiagramm: Absolutes Positionieren einer Achse und ablösendes Auftragsverhalten

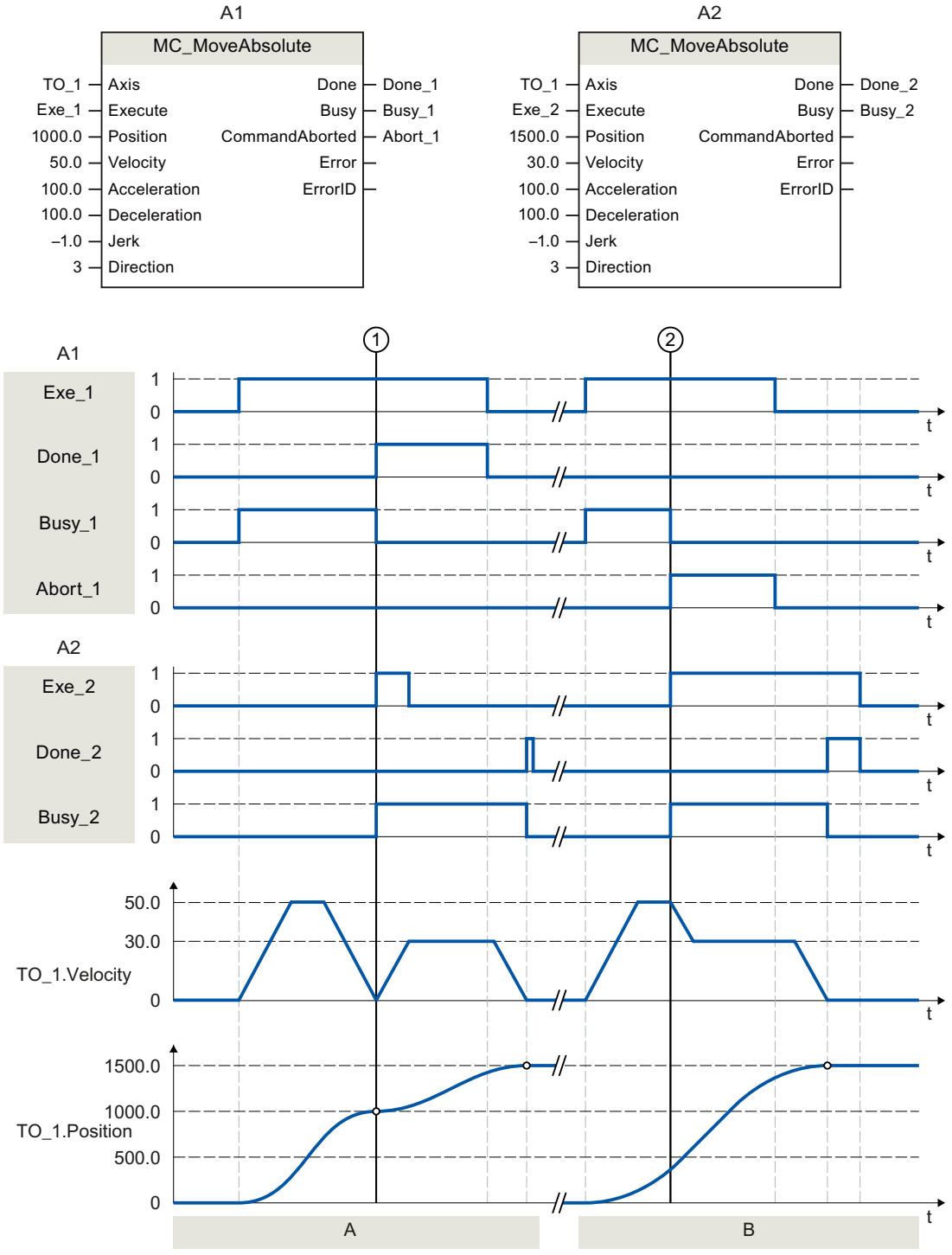

Abschnitt A	Eine Achse wird durch einen "MC_MoveAbsolute"-Auftrag (A1) auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveAbsolute"-Auftrag (A2) mit Zielposition 1500.0 gestartet. Das Erreichen der Zielposition 1500.0 wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveAbsolute"-Auftrag (A1) wird zum Zeitpunkt ② durch einen weiteren "MC_MoveAbsolute"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird auf die geänderte Geschwindigkeit abgebremst und auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveRelative V3

MC_MoveRelative: Achse relativ positionieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveRelative" verfahren Sie eine Achse relativ zu der Position, die bei Beginn der Auftragsbearbeitung vorliegt.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveRelative"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_PositioningAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Distance	INPUT	LREAL	0.0	Wegstrecke für den Positioniervorgang (negativ oder positiv)	
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse relativ zur Startposition

Gehen Sie zum Verfahren einer Achse relativ zur Startposition folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveRelative: Funktionsdiagramm V3

Funktionsdiagramm: Relatives Positionieren einer Achse und ablösendes Auftragsverhalten

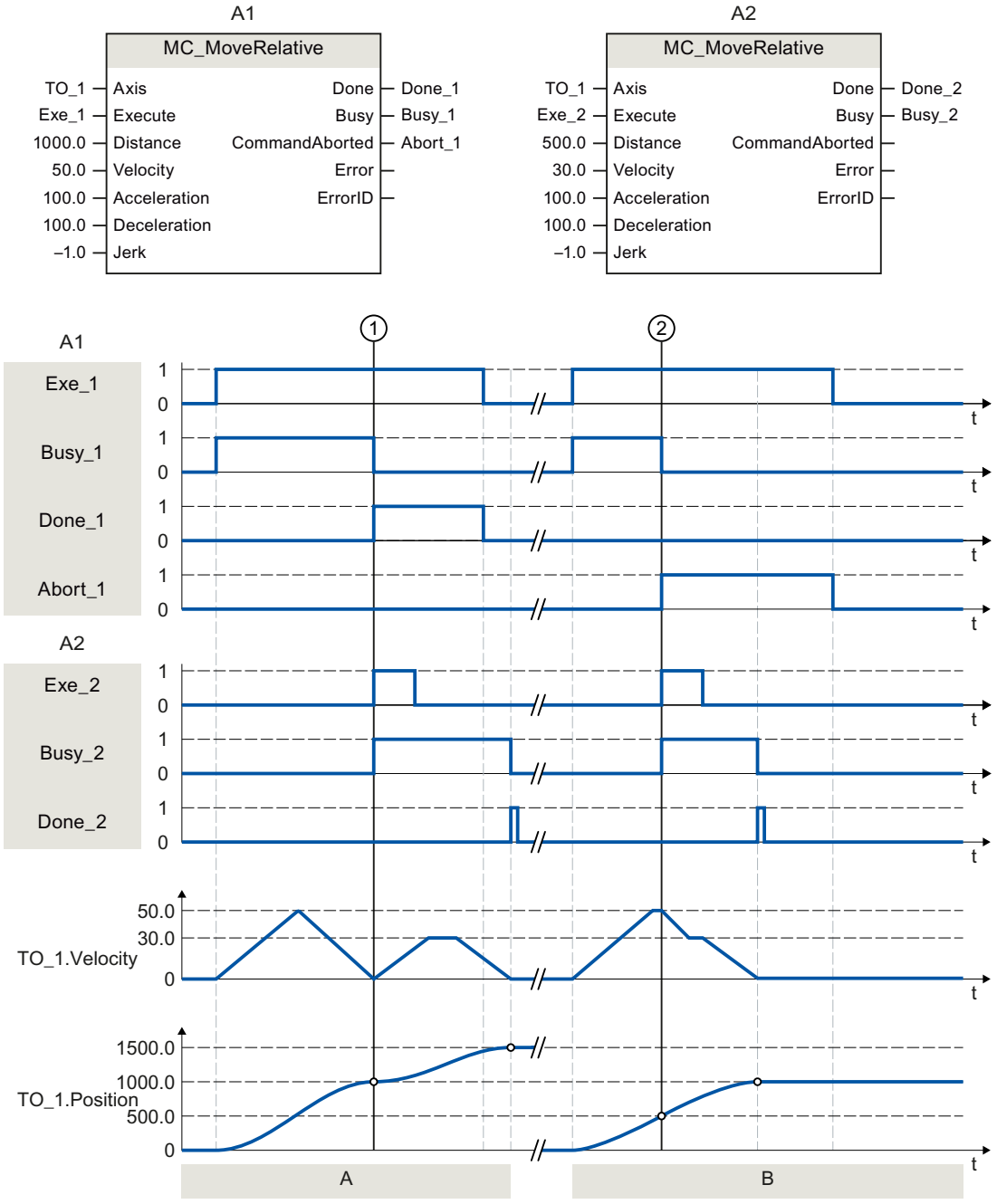

Abschnitt A	Die Achse wird durch einen "MC_MoveRelative"-Auftrag (A1) um die Wegstrecke ("Distance") 1000.0 verfahren (Startposition ist hier die Position 0.0). Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveRelative"-Auftrag (A2) mit der Wegstrecke 500.0 gestartet. Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveRelative"-Auftrag (A1) wird durch einen weiteren "MC_MoveRelative"-Auftrag (A2) abgelöst. Der Abbruch wird zum Zeitpunkt ② über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveVelocity V3

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V3

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveVelocity" bewegen Sie eine Achse mit konstanter Geschwindigkeit/Drehzahl.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveVelocity"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveVelocity":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang ("Velocity" = 0.0 ist erlaubt)	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Direction	INPUT	INT	0	Drehrichtung der Achse	
				0	Das Vorzeichen der am Parameter "Velocity" angegebenen Geschwindigkeit definiert die Drehrichtung.
				1	Positive Drehrichtung Betrag von "Velocity" wird verwendet.
				2	Negative Drehrichtung Betrag von "Velocity" wird verwendet.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	Deaktiviert Die Werte der Parameter "Velocity" und "Direction" werden berücksichtigt.
				TRUE	Aktiviert Die Werte an den Parametern "Velocity" und "Direction" werden **nicht** berücksichtigt. Die zum Funktionsstart aktuelle Geschwindigkeit und Richtung werden beibehalten. Sobald die Achse mit der zum Funktionsstart aktuellen Geschwindigkeit weiter verfährt, liefert der Parameter "InVelocity" den Wert TRUE.
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der MC_MoveVelocity-Auftrag ausgeführt wird. Danach gilt die Einstellung des folgenden Auftrags. Beim Einsatz einer Drehzahlachse wird der Parameter ignoriert.	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveVelocity"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl

Gehen Sie zum Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Velocity" die Geschwindigkeit / Drehzahl an, mit der die Achse verfahren werden soll.
3. Starten Sie den "MC_MoveVelocity"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Wenn der Parameter "InVelocity" den Wert TRUE zeigt, wurde die Sollgeschwindigkeit / Solldrehzahl erreicht. Die Achse wird mit dieser Geschwindigkeit konstant weiter verfahren. Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit / Drehzahl ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveVelocity: Funktionsdiagramm V3

Funktionsdiagramm: Verfahren einer Achse mit Geschwindigkeitsvorgabe und ablösendes Auftragsverhalten

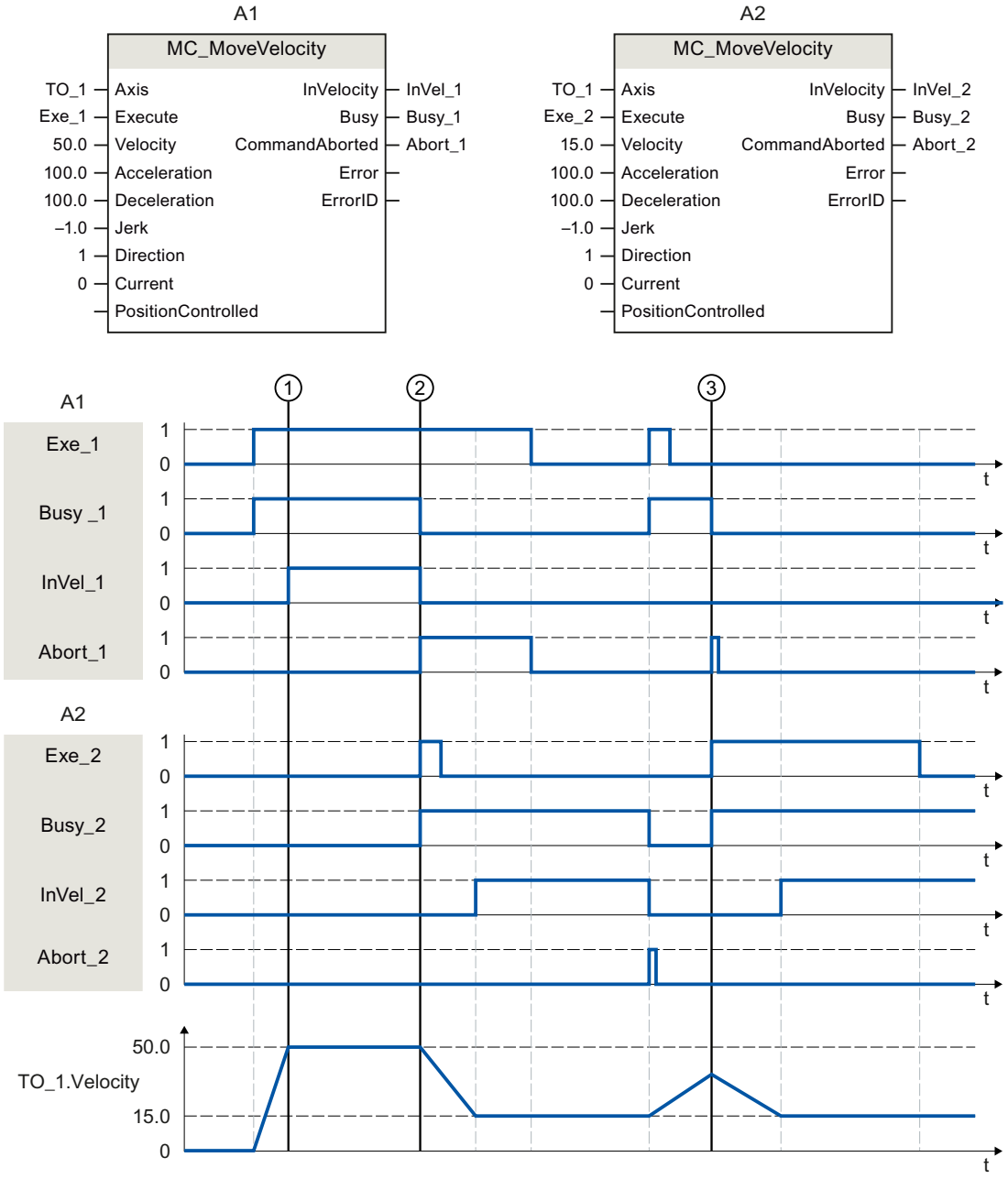

Ein über "Exe_1"angestoßener "MC_MoveVelocity"-Auftrag (A1) beschleunigt die Achse und meldet zum Zeitpunkt ① über "InVel_1" das Erreichen der Sollgeschwindigkeit 50.0.

Zum Zeitpunkt ② wird der Auftrag durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen

Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

Der laufende "MC_MoveVelocity"-Auftrag (A2) wird durch einen weiteren "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Abbruch wird über "Abort_2" gemeldet. Die Achse wird auf die neue Sollgeschwindigkeit 50.0 beschleunigt. Vor dem Erreichen der Sollgeschwindigkeit wird der aktuelle "MC_MoveVelocity"-Auftrag (A1) zum Zeitpunkt ③ durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

MC_MoveJog V3

MC_MoveJog: Achse im Tippbetrieb bewegen V3

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveJog" bewegen Sie eine Achse im Tippbetrieb.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveJog"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveJog":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.	
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.	
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang Wert > 0.0: Der angegebene Wert wird verwendet. Wert < 0.0: Der Betrag des angegebenen Werts wird verwendet. ("Velocity" = 0.0 ist erlaubt)	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
PositionControlled	INPUT	BOOL	TRUE	FALSE	Nicht lagegeregelter Betrieb
				TRUE	Lagegeregelter Betrieb
				Der Parameter gilt solange, wie der MC_MoveJog-Auftrag ausgeführt wird. Danach gilt die Einstellung des folgenden Auftrags. Beim Einsatz einer Drehzahlachse wird der Parameter ignoriert.	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveJog"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Verfahren einer Achse im Tippbetrieb

Gehen Sie zum Verfahren einer Achse im Tippbetrieb folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Verfahren Sie die Achse mit "JogForward" in positiver Richtung oder mit "JogBackward" in negativer Richtung.

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveJog: Funktionsdiagramm V3

Funktionsdiagramm: Verfahren einer Achse im Tippbetrieb

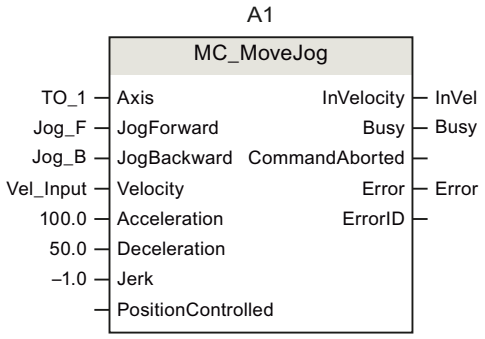

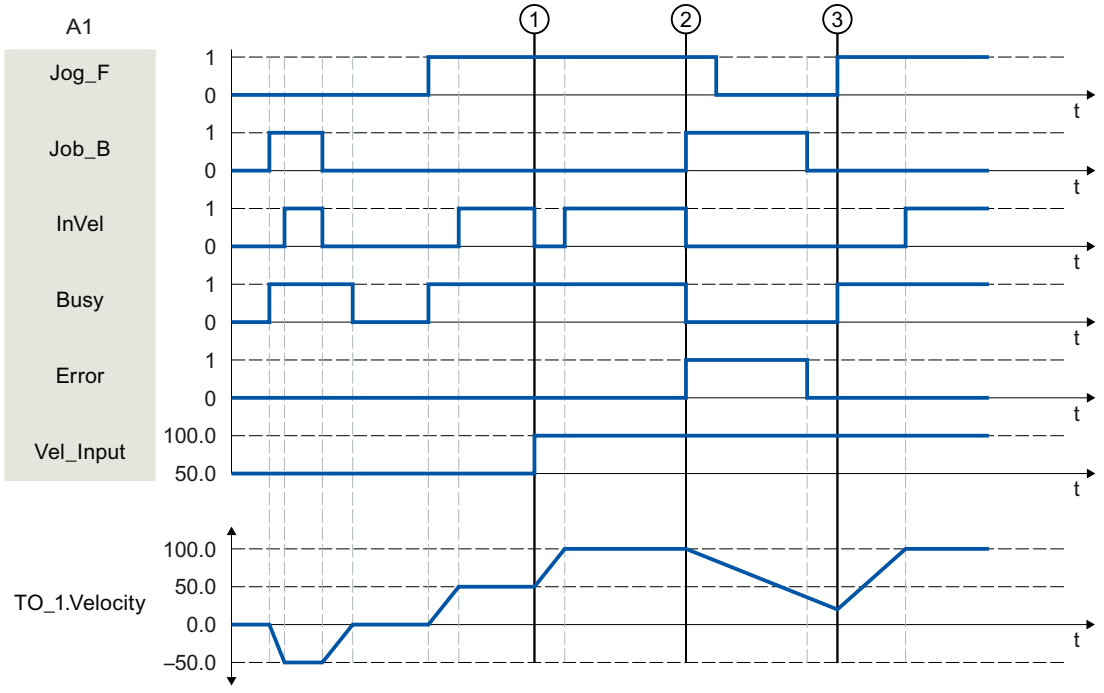

Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Sollgeschwindigkeit -50.0 wird über "InVel" = TRUE gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse abgebremst und zum Stillstand gebracht. Anschließend wird die Achse über "Jog_F" in positiver Richtung verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel" = TRUE gemeldet.

Zum Zeitpunktpunkt ① wird bei gesetztem "Jog_F" die Sollgeschwindigkeit über "Vel_Input" auf 100.0 geändert. Alternativ können Sie die Sollgeschwindigkeit auch über den Geschwindigkeits-Override ändern. "InVel" wird rückgesetzt. Die Achse wird beschleunigt. Das Erreichen der neuen Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

Bei gesetztem "Jog_F" wird zum Zeitpunkt ② ebenfalls "Jog_B" gesetzt. Wenn sowohl "Jog_F" als auch "Jog_B" gesetzt sind, wird die Achse mit der zuletzt gültigen Verzögerung

abgebremst. Über "Error" wird ein Fehler angezeigt und am Ausgang "ErrorID" der Fehler 16#8007 (falsche Richtungsangabe) ausgegeben.

Dieser Fehler wird durch Rücksetzen beider Eingänge "Jog_F" und "Jog_B" behoben.

Noch während der Bremsrampe wird zum Zeitpunkt ③ "Jog_F" gesetzt. Die Achse wird auf die zuletzt konfigurierte Geschwindigkeit beschleunigt. Das Erreichen der Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

MC_MoveSuperimposed V3

MC_MoveSuperimposed: Achse überlagernd positionieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveSuperimposed" starten Sie eine relative Positionierbewegung, die einer laufenden Basisbewegung überlagert wird.

Mit den Parametern "VelocityDiff", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung. Die Dynamikwerte werden zu den Werten der Basisbewegung addiert. Die Dauer der Basisbewegung wird durch eine überlagernde Bewegung nicht verlängert.

Die Dynamik der Gesamtbewegung der Achse ergibt sich durch die Addition der Dynamikwerte der Basisbewegung und der überlagerten Bewegung.

Das Verhalten der Gesamtbewegung unterscheidet sich je nach Art der Basisbewegung:

- Die Basisbewegung ist eine Einzelachsbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gesamtbewegung ist auf die konfigurierten Dynamikgrenzen begrenzt.
- Die Basisbewegung ist eine Gleichlaufbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gleichlaufbewegung der Folgeachse wird nicht auf die Dynamikgrenzen der Folgeachse begrenzt.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Leitachse im Gleichlauf wirkt sich auf die Leitachse und auf die Folgeachse aus.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Folgeachse im Gleichlauf wirkt sich nur auf die Folgeachse aus.

Im Technologie-Datenbaustein und im TIA Portal wird immer die Dynamik der Gesamtbewegung angezeigt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_MoveSuperimposed"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveSuperimposed":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_PositioningAxis	-	Technologieobjekt der Achse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Distance	INPUT	LREAL	0.0	Zusätzliche Wegstrecke für den überlagernden Positioniervorgang (negativ oder positiv)
VelocityDiff	INPUT	LREAL	-1.0	Maximale Geschwindigkeitsabweichung gegenüber der laufenden Bewegung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Überlagernde Positionierung abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Überlagerte Positionierbewegung starten

Um mit der Motion Control-Anweisung "MC_MoveSuperimposed" eine überlagerte Positionierbewegung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zusätzlich zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveSuperimposed"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MoveSuperimposed: Funktionsdiagramm V3

Funktionsdiagramm: Achsen überlagernd positionieren

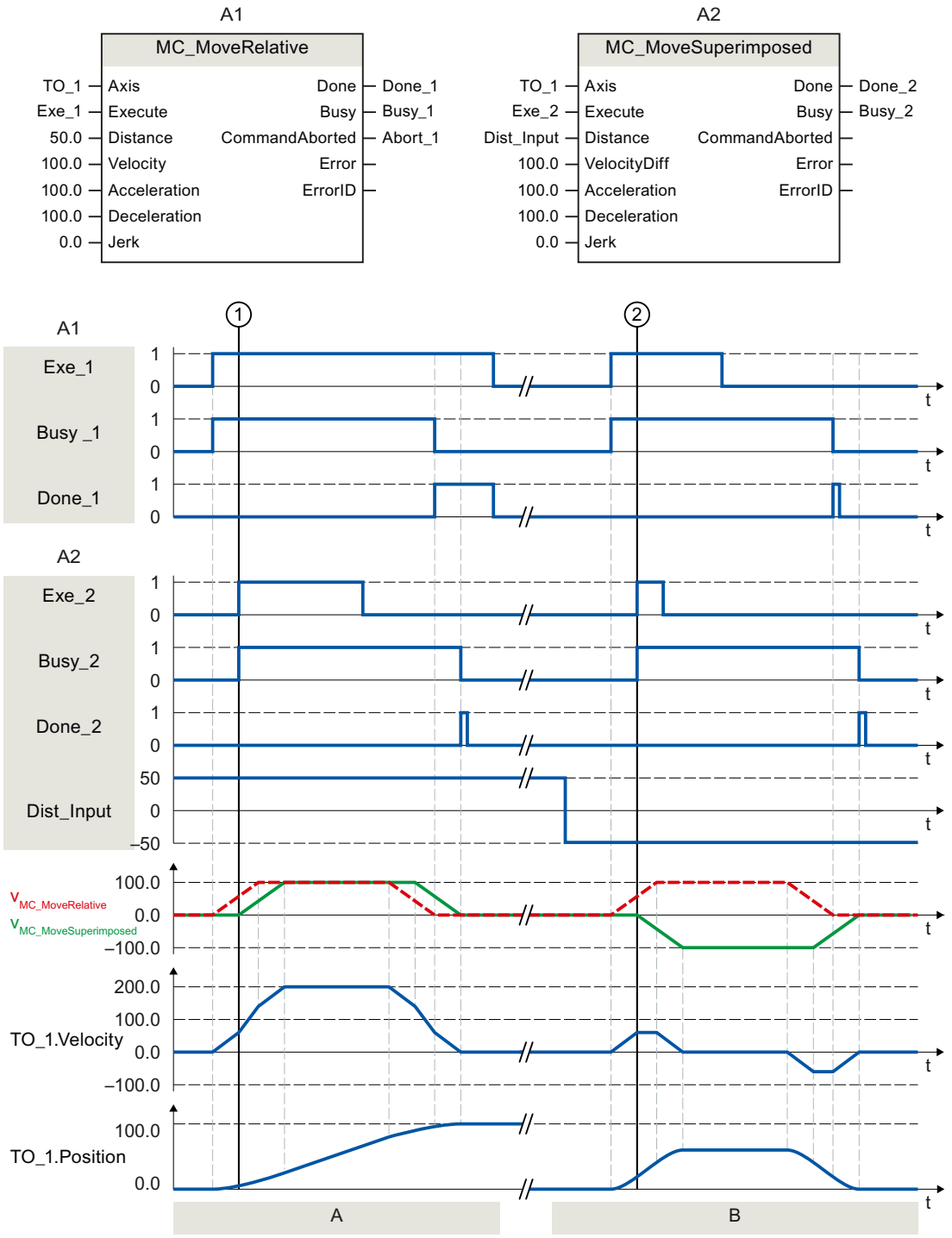

Abschnitt A	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ① wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke 50.0 angestoßen. Die Achse wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50 + 50 = 100.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.
Abschnitt B	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ② wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke -50.0 angestoßen. Die Achse reversiert und wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50.0 - 50.0 = 0.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.

MC_SetSensor V3

MC_SetSensor: Alternativen Geber als operativ wirksamen Geber umschalten V3

Beschreibung

Mit der Motion Control-Anweisung "MC_SetSensor" schalten Sie den Geber für die Lageregelung der Achse um.

Über den Parameter "Mode" 2 und 3 kann der Istwert des adressierten Gebers ohne Umschaltung angepasst werden.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt und der alternative Geber wurden korrekt konfiguriert.
- Kein Restart- und kein "MC-Home"-Auftrag aktiv.

Ablöseverhalten

- Ein "MC_SetSensor"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.
- Ein neuer "MC_SetSensor"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SetSensor":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	INPUT	TO_PositioningAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Sensor	INPUT	INT	1	Nummer des neuen Gebers (1 bis 4).	
Mode	INPUT	DINT	0	Der Mode bestimmt die Positionsangleichung zwischen dem alten und den neuen Geber.	
				0	Geber umschalten und aktuelle Istposition auf den neuen Geber übertragen Bei dieser Geberumschaltung werden Sprünge in der Positionsregelung verhindert. Eine stoßfreie Umschaltung der Geber ist möglich.
				1	Geber umschalten ohne die Istposition abzugleichen **Hinweis:** Bei aktiver Lageregelung wirkt eine zusätzliche Differenz der beiden Geber als zusätzliche Regelabweichung und kann zu einer Ausgleichbewegung führen.
				2	Istwert übertragen Die aktuelle Istposition wird auf den am Parameter "Sensor" angegebenen Geber übertragen.
				3	Istwert übertragen Die Istposition des "Referenzgebers" (Parameter "ReferenceSensor") wird auf den am Parameter "Sensor" angegebenen Geber übertragen.
ReferenceSensor	INPUT	INT	1	Nummer des Referenzgebers (siehe Parameter "Mode" 3)	
Done	OUTPUT	BOOL	FALSE	TRUE	Geber für Lageregelung der Achse wurde umgeschaltet.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Umschalten auf Absolutwertgeber

Wenn Sie den Geber auf einen Absolutwertgeber umschalten und den Istwert übertragen (Mode = 2, 3), wird der Istwert mit dem Wert des Absolutwertgebers und dem Absolutwert-Offset verrechnet. Beim Umschalten auf einen anderen Geber wird die Verrechnung des Istwerts verworfen. Der Absolutwertgeber liefert wieder den Absolutwert + Absolutwert-Offset (<TO>.StatusSensor[n].AbsEncoderOffset) ohne die Verrechnung durch den MC_SetSensor.

Messtaster, Nocken, Nockenspur

MC_MeasuringInput V3

MC_MeasuringInput: Einmaliges Messen starten V3

Mit der Motion Control-Anweisung "MC_MeasuringInput" wird ein einmaliges Messen gestartet.

Mit dem einmaligen Messen können mit einem Messauftrag eine oder zwei Flanken erfasst werden. Dem Messereignis wird die Position vom jeweiligen Technologieobjekt Achse oder Externer Geber zugeordnet. Das Messergebnis wird am Funktionsbaustein und im Technologie-Datenbaustein angezeigt und kann im Anwenderprogramm weiterverarbeitet werden. Damit ist der Messauftrag abgeschlossen.

Ein erneuter Messauftrag muss wieder neu über "MC_MeasuringInput.Execute" = TRUE gestartet werden.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Der Geber der Achse muss im Status "valid" (StatusSensor[1..4].State = valid) sein. Anderenfalls wird der Messauftrag am Funktionsbaustein mit einer Fehleranzeige abgelehnt.
- Das Messen über PROFIdrive ist während eines aktiven oder passiven Referenzierens nicht möglich.

Ablöseverhalten

Das Ablöseverhalten für "MC_MeasuringInput"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MeasuringInput":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
MeasuringInput	INPUT	TO_Measuring-Input	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	DINT	0	Art der Messung	
				0	Messung der nächsten steigenden Flanke
				1	Messung der nächsten fallenden Flanke
				2	Messung der beiden nächsten Flanken
				3	Messung an beiden Flanken, beginnend mit der steigenden Flanke • Steigende Flanke = "MeasuredValue1" (Messwert 1) • Fallende Flanke = "MeasuredValue2" (Messwert 2)
				4	Messung an beiden Flanken, beginnend mit der fallenden Flanke • Fallende Flanke = "MeasuredValue1" • Steigende Flanke = "MeasuredValue2"
MeasuringRange	INPUT	BOOL	FALSE	Erfassung der Messwerte Beachten Sie die zeitlichen Randbedingungen.	
				FALSE	Messwerte immer erfassen
				TRUE	Messwerte nur innerhalb des Messbereichs erfassen
StartPosition	INPUT	LREAL	0.0	Anfangsposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
EndPosition	INPUT	LREAL	0.0	Endposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Baustein wurde fertig bearbeitet. Messwerte sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Messauftrag wurde abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
MeasuredValue1	OUTPUT	LREAL	0.0	Erster Messwert	
MeasuredValue2	OUTPUT	LREAL	0.0	Zweiter Messwert (bei Messung an beiden Flanken)	

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MeasuringInput: Funktionsdiagramm V3

Funktionsdiagramm: Einmaligen Messauftrag starten

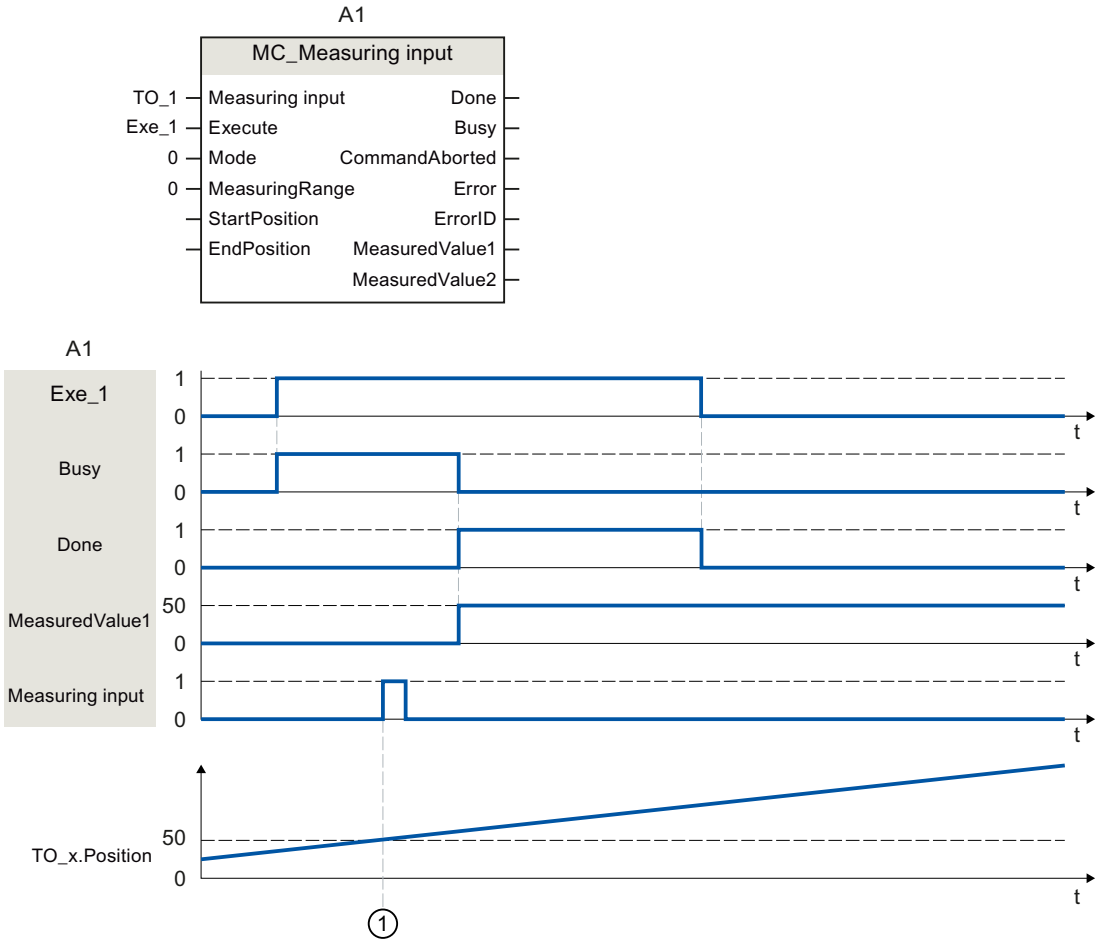

Ein über "Execute" angestoßener "MC_MeasuringInput"-Auftrag mit eingestellter "Mode" = 0 führt eine Messung der nächsten steigenden Flanke durch. Eine erfolgreich durchgeführte Messung wird mit "Done" = TRUE signalisiert. Der ermittelte Messwert ① (im Beispiel 50) wird über "MeasuredValue1" ausgegeben.

MC_MeasuringInputCyclic V3

MC_MeasuringInputCyclic: Zyklisches Messen starten V3

Mit der Motion Control-Anweisung "MC_MeasuringInputCyclic" wird eine zyklische Messung gestartet.

Mit dem zyklischen Messen werden in jedem Servotakt bis zu zwei Messereignisse vom System erfasst und die dazugehörigen Messpositionen angezeigt. Die Messungen werden zyklisch fortgesetzt, bis sie per Befehl beendet werden.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das zyklische Messen ist nur beim Messen über TM Timer DIDQ möglich.
- Der operativ wirksame Geber der Achse muss im Status "valid" (StatusSensor[1..4].State = valid) sein. Anderenfalls wird der Messauftrag am Funktionsbaustein mit einer Fehleranzeige abgelehnt.

Ablöseverhalten

Das Ablöseverhalten für "MC_MeasuringInputCyclic"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MeasuringInputCyclic":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
MeasuringInput	INPUT	TO_Measuring-Input	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Mode	INPUT	DINT	0	Art der Messung	
				0	Messung der steigenden Flanken
				1	Messung der fallenden Flanken
				2	Messung der beiden Flanken
MeasuringRange	INPUT	BOOL	FALSE	Erfassung der Messwerte Beachten Sie die zeitlichen Randbedingungen.	
				FALSE	Messwerte immer erfassen
				TRUE	Messwerte nur innerhalb des Messbereichs erfassen
StartPosition	INPUT	LREAL	0.0	Anfangsposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
EndPosition	INPUT	LREAL	0.0	Endposition des Messbereichs Positionsangaben, die bei aktiver Modulofunktion an der Achse oder dem externen Geber außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Messauftrag wurde abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	
MeasuredValue1	OUTPUT	LREAL		Erster Messwert	
MeasuredValue2	OUTPUT	LREAL		Zweiter Messwert (bei Messung von mehreren Flanken in einem Servotakt)	
MeasuredValue1Counter	OUTPUT	UDINT	0	Zählwert für den ersten Messwert	
MeasuredValue2Counter	OUTPUT	UDINT	0	Zählwert für den zweiten Messwert	
LostEdgeCounter1	OUTPUT	UDINT	0	Zählwert für die verloren gegangenen Flanken im Takt der Erfassung vom ersten Messwert	
LostEdgeCounter2	OUTPUT	UDINT	0	Zählwert für die verloren gegangenen Flanken im Takt der Erfassung vom zweiten Messwert	

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_MeasuringInputCyclic: Funktionsdiagramm V3

Funktionsdiagramm: Zyklischen Messauftrag starten

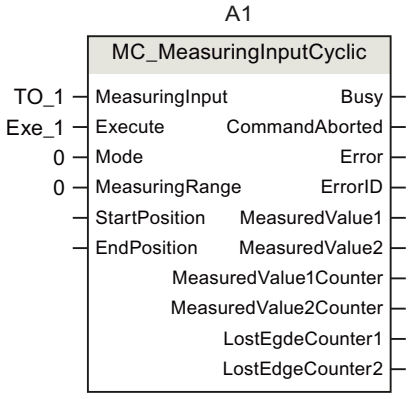

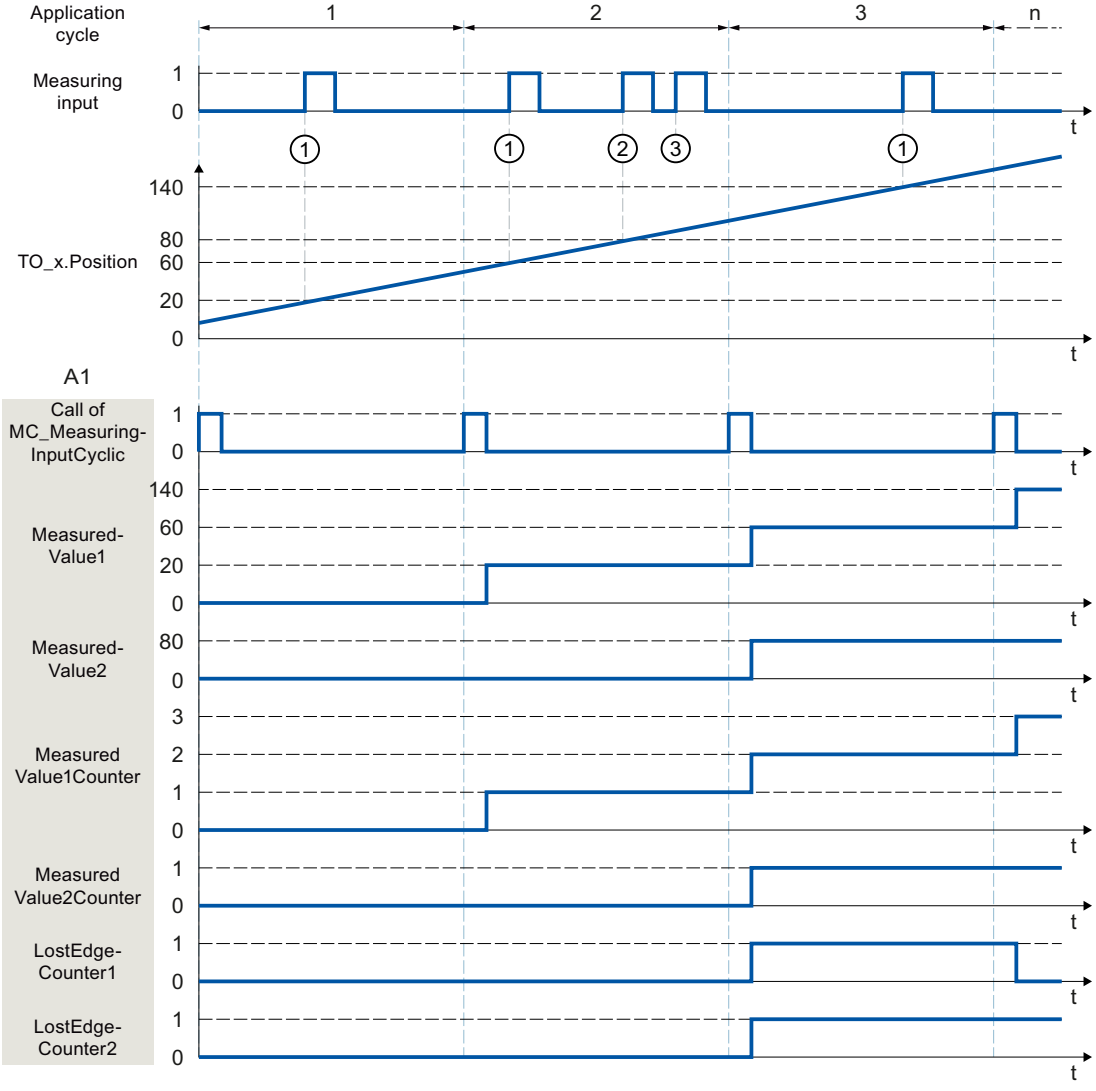

Ein über "Execute" (Signal und Startpunkt nicht dargestellt) angestoßener "MC_MeasuringInputCyclic"-Auftrag (z. B. Aufruf im MC-PreServo [OB67]) mit eingestelltem "Mode" = 0 und ohne vorgegebenen Messbereich "MeasuringRange = FALSE führt eine

Messung der steigenden Flanken durch. Durch den Aufruf der Motion Control-Anweisung "MC_MeasuringInputCyclic" im MC-PreServo [OB67] oder MC-PreServo [OB67] erfolgt der Aufruf des Messauftrags als auch die Messung selbst im gleichen in Applikationszyklus statt.

Die ermittelte Istposition zum Zeitpunkt der ersten steigenden Flanke ① im Servotakt wird über "MeasuredValue1" ausgegeben und die Zählvariable "MeasuredValue1Counter" um "1" hochgezählt.

Die ermittelte Istposition zum Zeitpunkt der zweiten steigenden Flanke ② im Servotakt wird über "MeasuredValue2" ausgegeben und die Zählvariable "MeasuredValue2Counter" um "1" hochgezählt.

Treten im gleichen Servotakt weitere steigende Flanken ③ auf, werden diese im LostEdgeCounter1 und LostEdgeCounter2 festgehalten.

MC_AbortMeasuringInput V3

MC_AbortMeasuringInput: Aktives Messen abbrechen V3

Mit der Motion Control-Anweisung "MC_AbortMeasuringInput" wird ein aktiver einmaliger oder zyklischer Messauftrag abgebrochen.

Anwendbar auf

- Messtaster

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.

Ablöseverhalten

Das Ablöseverhalten für "MC_AbortMeasuringInput"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_AbortMeasuringInput":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
MeasuringInput	INPUT	TO_Measuring-Input	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	0	Der Funktionsbaustein wurde bearbeitet. Der Mess-auftrag wurde deaktiviert.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_OutputCam V3

MC_OutputCam: Nocken aktivieren/deaktivieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_OutputCam" aktivieren Sie den eingestellten Nocken.

Je nach Nockentyp sind die folgenden Eingangsparameter wirksam:

- Wegnocken
  - "OnPosition"
  - "OffPosition"
- Zeitnocken
  - "OnPosition"
  - "Duration"

Mit den Parametern "Mode" und "Direction" bestimmen Sie die Betriebsart und die Wirkrichtung des Nockens.

Bei "MC_OutputCam.Enable" = TRUE werden die Eingangsparameter immer gelesen und sind mit dem nächsten Servotakt wirksam.

Bei einem technologischen Alarm wird der Nocken nach erfolgter Fehlerquittierung wieder bearbeitet.

Anwendbar auf

- Nocken

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das übergeordnete Technologieobjekt muss eine gültige Position haben.

- Sollwertnocken werden bei einer Achse im nicht lagegeregelten Betrieb nicht ausgegeben.
- Ein Sollwertnocken mit der Einstellung "MC_OutputCam.OnPosition" = 0 schaltet bei Sollposition = 0.

Ablöseverhalten

Der "MC_OutputCam" wird abgebrochen durch:

- Das Sperren des Nockens mit "MC_OutputCam.Enable" = FALSE
- Im Anwenderprogramm kann nur eine Instanz des Funktionsbausteins "MC_OutputCam" auf einen Nocken aktiv sein. Eine zweite Instanz des Funktionsbausteins "MC_OutputCam" auf einen bereits aktiven Nocken wird mit Fehler abgelehnt.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_OutputCam":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
OutputCam	INPUT	TO_OutputCam	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	FALSE	Nocken wird gesperrt
				TRUE	Nocken wird bearbeitet
OnPosition	INPUT	LREAL	0.0	Anfangsposition des Nockens [Einheit des verschalteten Technologieobjekts] Positionsangaben, die bei aktiver Modulofunktion an der Achse oder des Externen Gebers außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
OffPosition	INPUT	LREAL	0.0	Endposition des Wegnockens [Einheit des verschalteten Technologieobjekts] Positionsangaben, die bei aktiver Modulofunktion an der Achse oder des Externen Gebers außerhalb des Modulobereichs liegen, werden in den Modulobereich gespiegelt.	
Duration	INPUT	LREAL	0.0	Einschaltdauer des Zeitnockens [Einheit: ms] Der Wert für die Einschaltdauer eines Zeitnockens muss größer 0.0 sein.	
Mode	INPUT	DINT	1	Betriebsart	
				1	Standard Nockenfunktionalität (Ausgabe nicht invertiert)
				2	Nockenfunktionalität mit Ausgabe invertiert
				3	Nocken immer ein (Während "Enable" = TRUE)
Direction	INPUT	DINT	1	Wirkrichtung des Nockens	
				1	Positive Richtung
				2	Negative Richtung
				3	Beide Richtungen

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CamOutput	OUTPUT	BOOL	-	Statusanzeige bezogen auf den letzten Aufruf der Motion Control-Anweisung "MC_OutputCam"	
Busy	OUTPUT	BOOL	FALSE	TRUE	Bearbeitung des Nockens ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_OutputCam: Funktionsdiagramm V3

Funktionsdiagramm: Aktivieren des eingestellten Nockentyps

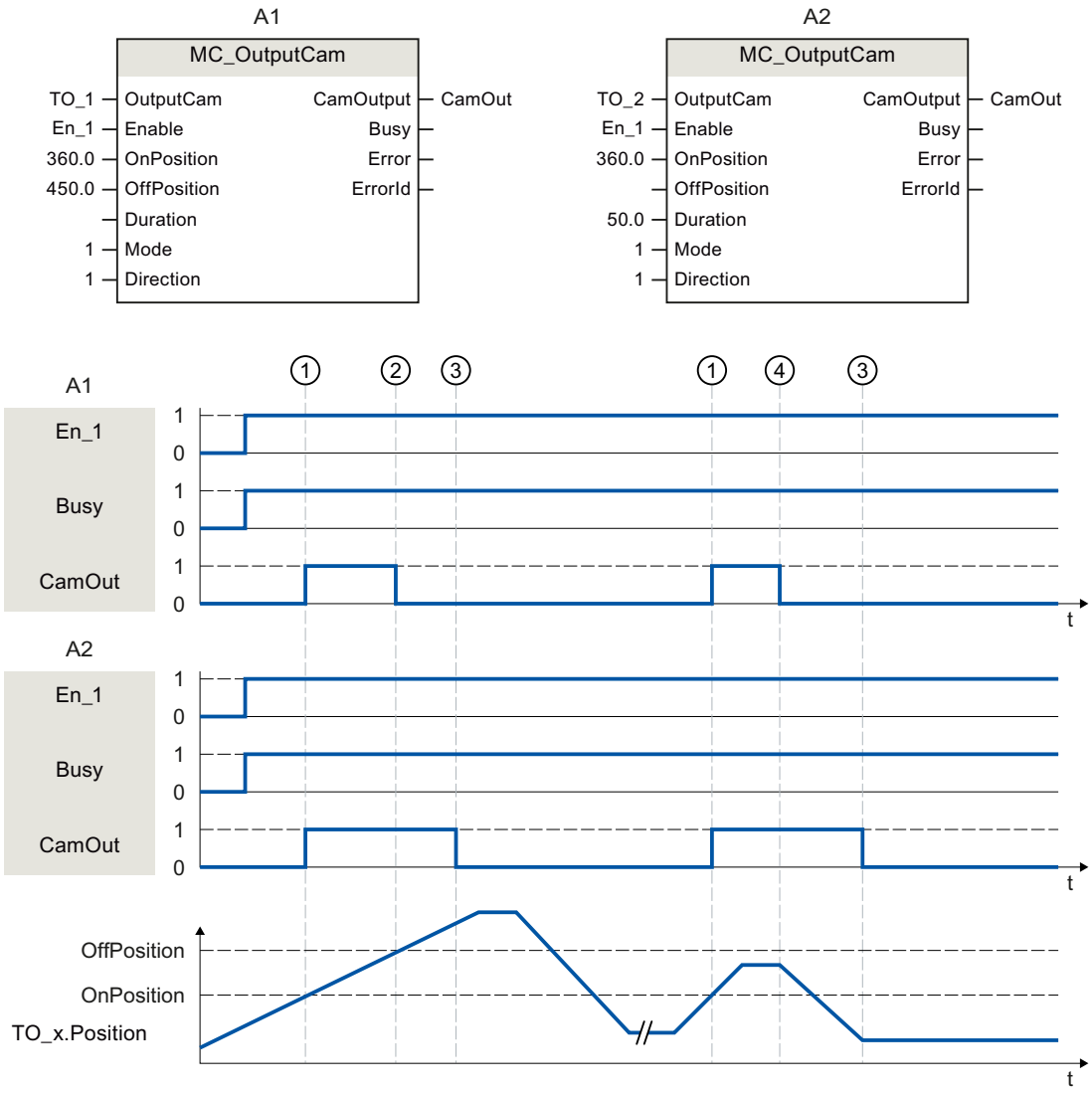

A1 Wegnocken

A2 Zeitnocken

Ein Nocken wird mit "Enable" = TRUE aktiviert. Der Nocken wird entsprechend den Parametern ausgegeben.

Die Bearbeitung des Nockens wird mit "Busy" = TRUE angezeigt.

Bei eingestellter Wirkrichtung "Direction" = 1 (positiv) verhält sich der Nocken abhängig vom eingestellten Nockentyp wie folgt:

- Der Wegnocken schaltet ein bei "OnPosition" ① und schaltet wieder aus bei "OffPosition" ②. Bei einer Richtungsumkehr wird der Nocken ausgeschaltet ④.
- Der Zeitnocken schaltet ein bei "OnPosition" ① und schaltet wieder aus nach Ablauf der eingestellten Zeit "Duration" ③. Ein eingeschalteter Zeitnocken bleibt für die eingestellte Einschaltdauer "Duration" aktiv, auch wenn die Anfangsposition wieder in umgekehrter Richtung überfahren wird.

Über "CamOutput" wird der Schaltzustand des Nockens ausgegeben.

Mit "Enable" = FALSE wird der eingestellte Nockentyp deaktiviert. "Busy" = FALSE zeigt, dass der Nocken nicht bearbeitet wird.

MC_CamTrack V3

MC_CamTrack: Nockenspur aktivieren/deaktivieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_CamTrack" wird die Bearbeitung einer Nockenspur freigegeben.

Anwendbar auf

- Nockenspur

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das übergeordnete Technologieobjekt muss eine gültige Position haben.
- Sollwertnocken werden bei einer Achse im nicht lagegeregelten Betrieb nicht ausgegeben.

Ablöseverhalten

- Der "MC_CamTrack" wird abgebrochen durch das Sperren der Nockenspur mit "MC_CamTrack.Enable" = FALSE.
- Im Anwenderprogramm kann nur eine Instanz des Funktionsbausteins "MC_CamTrack" auf einer Nockenspur aktiv sein. Eine zweite Instanz des Funktionsbausteins "MC_CamTrack" auf eine bereits aktive Nockenspur wird mit Fehler abgelehnt.
- Bei Änderung der Parameter der Motion Control-Anweisung "MC_CamTrack" oder am Technologie-Datenbaustein wird die Nockenspur neu berechnet. Die Nockenspurbearbeitung wird entsprechend aller Parametervorgaben bearbeitet.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_OutputCam":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CamTrack	INPUT	TO_CamTrack	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	FALSE	Nockenspur wird gesperrt
				TRUE	Nockenspur wird bearbeitet

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung		
Mode	INPUT	INT	1	0	"Enable" = TRUE	Nockenspurbearbeitung wird sofort aktiviert Die Nockenspurdaten sind sofort wirksam. Bereits angesteuerte Wegnocken werden abgebrochen, wenn dessen Spursignal durch die geänderten Nockenspurdaten nicht weiterhin angesteuert wird. Bereits angesteuerte Zeitnocken werden in jedem Fall abgebrochen.
					"Enable" = FALSE	Nockenspurbearbeitung wird sofort beendet Wegnocken/Zeitnocken werden unmittelbar abgebrochen
				1	"Enable" = TRUE	Nockenspurbearbeitung wird sofort/mit nächstem Spurzyklus aktiviert • Beim erstmaligen Aktivieren einer Nockenspur beginnt die Nockenspurbearbeitung sofort. • Bei bereits aktiver Nockenspurbearbeitung wird die aktuelle Nockenspur bis zum Ende des Spurzyklus ausgegeben. Anschließend werden die neuen Nockenspurdaten wirksam.
					"Enable" = FALSE	Nockenspurbearbeitung wird am Nockenspurende beendet
				2	"Enable" = TRUE	Nockenspurausgang wird sofort eingeschaltet und bleibt eingeschaltet
					"Enable" = FALSE	Nockenspurausgang wird sofort ausgeschaltet

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
InvertOutput	INPUT	BOOL	FALSE	Invertierte Ausgabe	
				TRUE	Spurausgang wird invertiert ausgegeben
				FALSE	Spurausgang wird nicht invertiert ausgegeben
TrackOutput	OUTPUT	BOOL	-	Zeigt den Schaltzustand der Nockenspur an.	
Busy	OUTPUT	BOOL	FALSE	TRUE	Bearbeitung der Nockenspur ist aktiv.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

MC_CamTrack: Funktionsdiagramm V3

Funktionsdiagramm: Nockenspur aktivieren

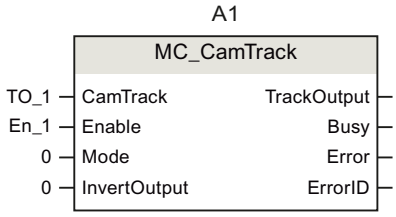

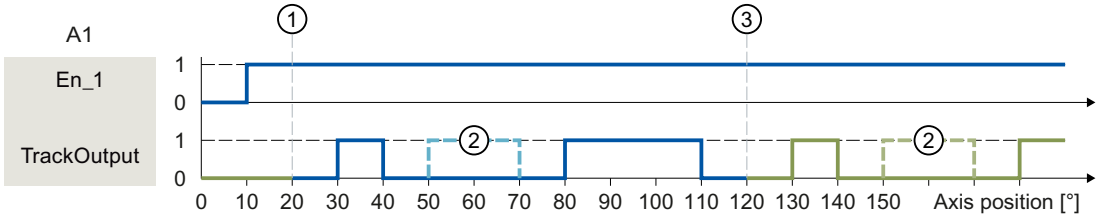

① Achsbezugsposition
② Als nicht gültig konfigurierter Nocken
③ Beginn zyklisch fortgesetzte Nockenspur

Eine Nockenspur wird mit "Enable" = TRUE aktiviert. Die Nockenspur wird entsprechend den im Technologie-Datenbaustein eingestellten Parametern ausgegeben:

Variable		Wert	Beschreibung
<TO>Parameter.			
	CamTrackType	0	Festgelegter Nockentyp für die Nockenspur ist Wegnocken
	ReferencePosition	20.0	Festgelegte Achsbezugsposition für die Nockenspur
	CamTrackLength	100.0	Festgelegte Länge der Nockenspur
	CamTrack[1].Existent	TRUE	Der erste Wegnocken der Nockenspur wird als gültig festgelegt
	CamTrack[1].OnPosition	10.0	Anfangsposition für den ersten Nocken der Nockenspur
	CamTrack[1].OffPosition	20.0	Endposition für den ersten Nocken der Nockenspur

Variable		Wert	Beschreibung
	CamTrack[2].Existent	FALSE	Der zweite Wegnocken der Nockenspur wird als nicht gültig festgelegt
	CamTrack[2].OnPosition	30.0	Anfangsposition für den zweiten Nocken der Nockenspur
	CamTrack[2].OffPosition	50.0	Endposition für den zweiten Nocken der Nockenspur
	CamTrack[3].Existent	TRUE	Der dritte Wegnocken der Nockenspur wird als gültig festgelegt
	CamTrack[3].OnPosition	60.0	Anfangsposition für den dritten Nocken der Nockenspur
	CamTrack[3].OffPosition	90.0	Endposition für den dritten Nocken der Nockenspur

Synchrone Bewegung

MC_GearIn V3

MC_GearIn: Getriebegleichlauf starten V3

Beschreibung

Mit der Motion Control-Anweisung "MC_GearIn" starten Sie einen Getriebegleichlauf zwischen einer Leitachse und einer Folgeachse.

Mit den Parametern "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Folgeachse beim Aufsynchronisieren.

Dauer und Wegstrecke des Aufsynchronisierens sind von folgenden Parametern abhängig:

- Startzeitpunkt des "MC_GearIn"-Auftrags
- Dynamik der Folgeachse zum Startzeitpunkt
- Dynamikvorgaben für das Aufsynchronisieren
- Dynamik der Leitachse

Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen an den Parametern "RatioNumerator" und "RatioDenominator" vor (Zähler/Nenner).

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- **Positiver Getriebefaktor:**
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- **Negativer Getriebefaktor**:
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, eine Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_GearIn"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GearIn":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
RatioNumerator	INPUT	DINT	1	Getriebefaktor Zähler Zulässige ganzzahlige Werte: -2147483648 bis 2147483647 (Wert 0 nicht zulässig)
RatioDenominator	INPUT	DINT	1	Getriebefaktor Nenner Zulässige ganzzahlige Werte: 1 bis 2147483647
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
InGear	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_GearIn" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse und den Getriebefaktor an.
3. Starten Sie den "MC_GearIn"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InGear" den Wert TRUE zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Die Parameter "InGear" und "Busy" zeigen so lange den Wert TRUE, bis der "MC_GearIn"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_GearIn: Funktionsdiagramm V3

Funktionsdiagramm: Aufsynchronisieren und Umschalten des Leitwerts

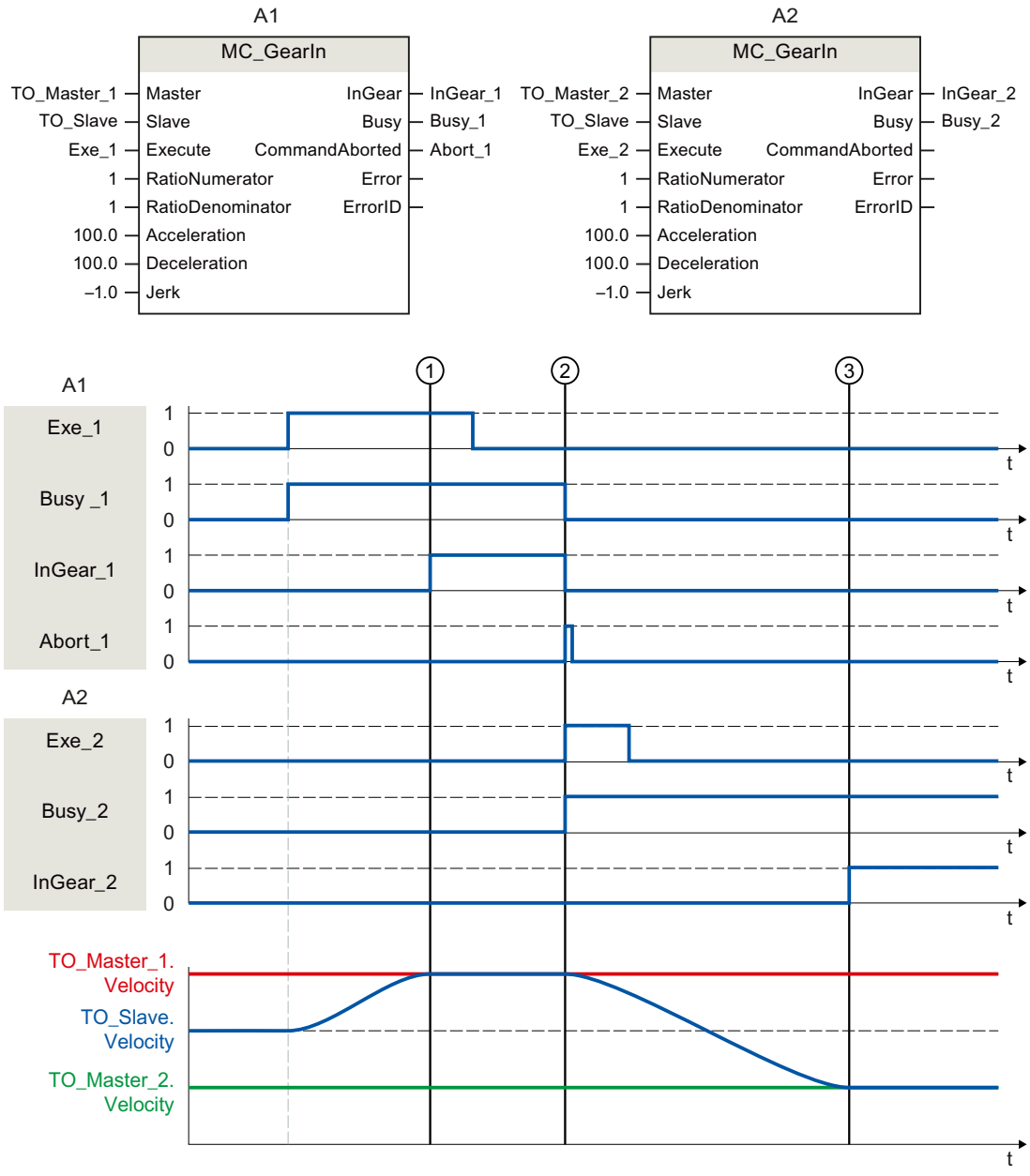

Über "Exe_1" wird ein "MC_GearIn"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird auf die Leitachse (TO_Master_1) aufsynchronisiert. Zum Zeitpunkt ① wird über "InGear_1" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

Zum Zeitpunkt ② wird der Gleichlauf durch einen weiteren "MC_GearIn"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Folgeachse wird auf die neue Leitachse (TO_Master_2) aufsynchronisiert. Zum Zeitpunkt ③ wird über "InGear_2" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

MC_GearInPos V3

MC_GearInPos: Getriebegleichlauf mit vorgegebenen Synchronpositionen starten V3

Beschreibung

Mit der Motion Control-Anweisung "MC_GearInPos" starten Sie einen Getriebegleichlauf zwischen einer Leitachse und einer Folgeachse.

Der Gleichlauf wird vorlaufend zu den vorgegebenen Synchronpositionen für die Leit- und Folgeachse aufsynchronisiert. Folgende Arten des Aufsynchronisierens sind möglich:

- **Aufsynchronisieren über Dynamikparameter**
  (SyncProfileReference = 0)
- **Aufsynchronisieren über Leitwertweg**
  (SyncProfileReference = 1)

Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen an den Parametern "RatioNumerator" und "RatioDenominator" vor (Zähler/Nenner).

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- **Positiver Getriebefaktor:**
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- **Negativer Getriebefaktor**:
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_GearInPos"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GearInPos":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
RatioNumerator	INPUT	DINT	1	Getriebefaktor Zähler Zulässige ganzzahlige Werte: -2147483648 bis 2147483647 (Wert 0 nicht zulässig)
RatioDenominator	INPUT	DINT	1	Getriebefaktor Nenner Zulässige ganzzahlige Werte: 1 bis 2147483647
MasterSyncPosition	INPUT	LREAL	0.0	Synchronposition Leitachse Position der Leitachse, ab der Leit- und Folgeachse synchron verfahren
SlaveSyncPosition	INPUT	LREAL	0.0	Synchronposition Folgeachse Position der Folgeachse, ab der Leit- und Folgeachse synchron verfahren
SyncProfileReference	INPUT	DINT	1	Art des Aufsynchronisierens 0 – Aufsynchronisieren über Dynamikparameter 1 – Aufsynchronisieren über Leitwertweg
MasterStartDistance	INPUT	LREAL	1.0	Leitwertweg (bei "SyncProfileReference" = 1)
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck (bei "SyncProfileReference" = 0) Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
SyncDirection	INPUT	DINT	3	Richtung des Aufsynchronisierens (wirksam bei Achsen mit aktivierter Modulo-Einstellung)	
				1	Positive Richtung Beim Aufsynchronisieren darf die Folgeachse nur in positive Richtung fahren.
				2	Negative Richtung Beim Aufsynchronisieren darf die Folgeachse nur in negative Richtung fahren.
				3	Kürzester Weg Beim Aufsynchronisieren sind Richtungsänderungen der Folgeachse erlaubt.
StartSync	OUTPUT	BOOL	FALSE	TRUE	Die Folgeachse wird auf die Leitachse aufsynchronisiert.
InSync	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_GearInPos" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse, den Getriebefaktor und die Synchronpositionen an.
3. Starten Sie den "MC_GearInPos"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InSync" den Wert TRUE zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Die Parameter "InSync" und "Busy" zeigen so lange den Wert TRUE, bis der "MC_GearInPos"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_GearInPos: Funktionsdiagramm V3

Funktionsdiagramm: Aufsynchronisieren über Dynamikparameter/Leitwertweg

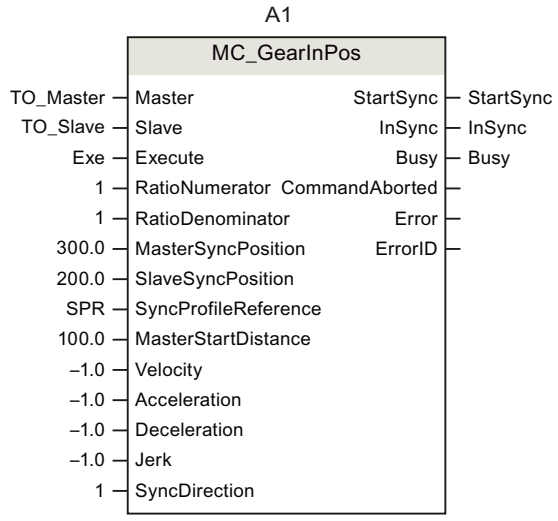

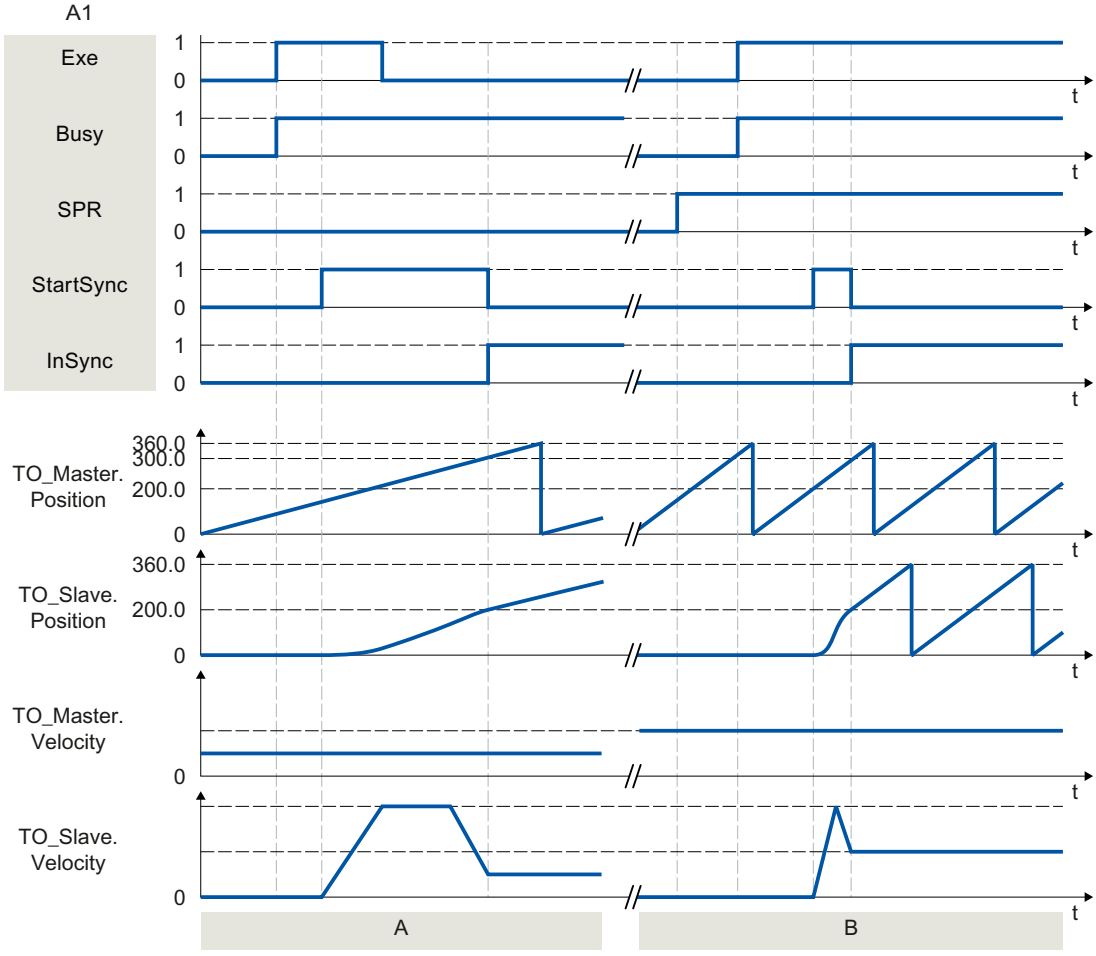

Abschnitt A	Über "Exe" wird ein "MC_GearInPos"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird über die vorgegebenen Dynamikparameter auf die Leitachse (TO_Master) aufsynchronisiert. Der zum Aufsynchronisieren benötigte Weg wird vom System berechnet. Über "StartSync" wird der Start des Aufsynchronisierens angezeigt. Beim Erreichen der vorgegebenen Synchronpositionen wird über "InSync" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.
Abschnitt B	Über "Exe" wird ein "MC_GearInPos"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird über den vorgegebenen Leitwertweg auf die Leitachse (TO_Master) aufsynchronisiert. Die zum Aufsynchronisieren benötigte Dynamik wird vom System berechnet. Über "StartSync" wird der Start des Aufsynchronisierens angezeigt. Über "InSync" wird gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

MC_PhasingAbsolute V3

MC_PhasingAbsolute: Leitwert an der Folgeachse absolut verschieben V3

Beschreibung

Mit der Motion Control-Anweisung "MC_PhasingAbsolute" verschieben Sie den Leitwert an einer Folgeachse im Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos" absolut. Die Position der Leitachse wird dadurch nicht beeinflusst.

Mit den Parametern "Velocity", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung der Folgeachse. Die Dynamikwerte werden zu den Werten der Gleichlaufbewegung addiert.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist über die Motion Control-Anweisung "MC_GearIn" oder "MC_GearInPos" auf die Leitachse aufsynchronisiert ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE).
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_PhasingAbsolute"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht die Leitwertverschiebung nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_PhasingAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
PhaseShift	INPUT	LREAL	0.0	Absolute Leitwertverschiebung
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Leitwertverschiebung ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
AbsolutePhaseShift	OUTPUT	LREAL	0.0	So lange "Busy" = TRUE: Anzeige der bisher ausgefahrenen Leitwertverschiebung	

Absolute Leitwertverschiebung starten

Um mit der Motion Control-Anweisung "MC_PhasingAbsolute" eine absolute Leitwertverschiebung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "PhaseShift" die absolute Leitwertverschiebung an.
3. Starten Sie den "MC_PhasingAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "AbsolutePhaseShift", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_GearInPos V3 (Seite 3457)

MC_GearIn V3 (Seite 3453)

MC_PhasingAbsolute: Funktionsdiagramm V3

Funktionsdiagramm: Leitwert absolut verschieben

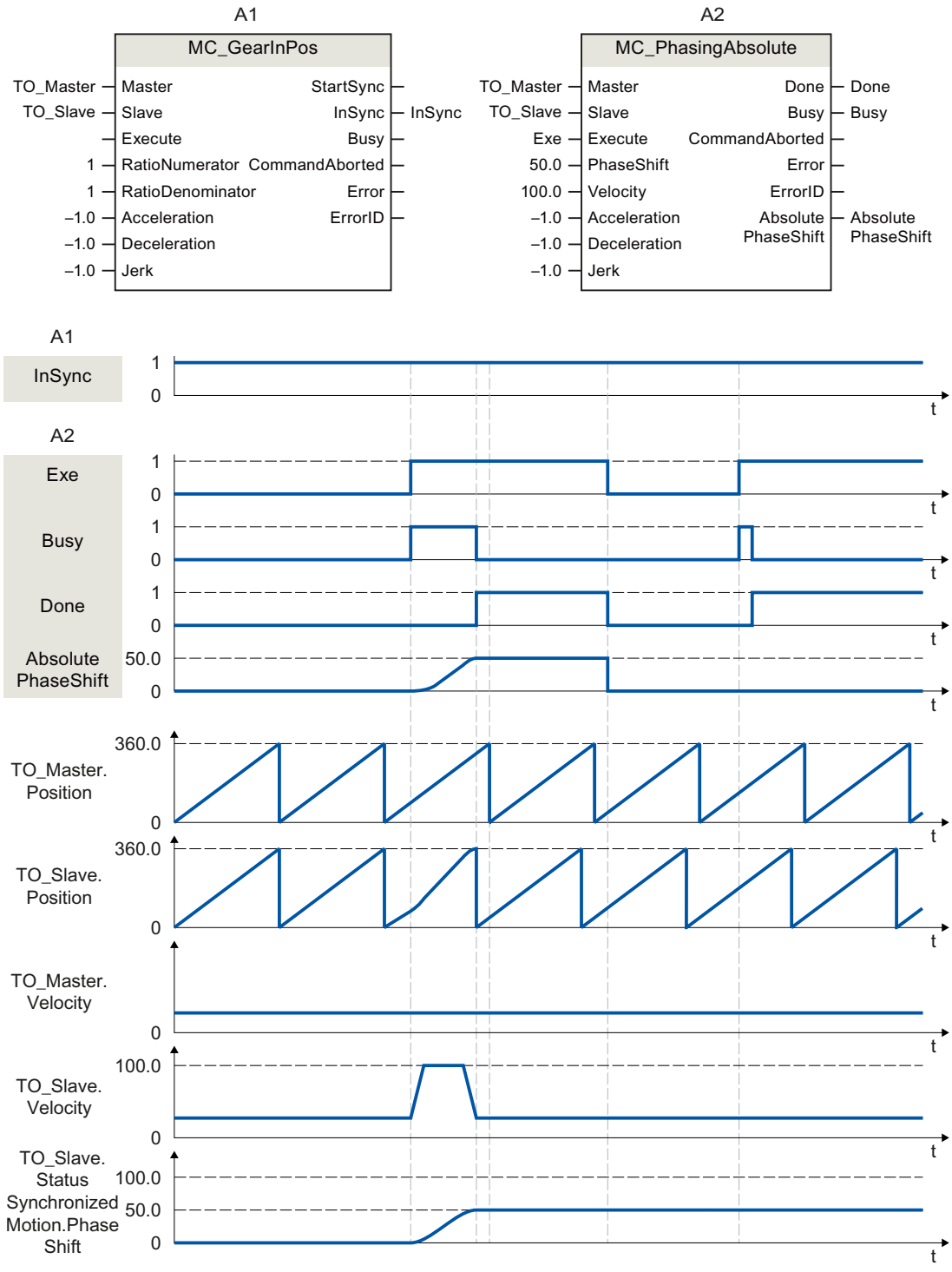

Während eines aktiven Getriebegleichlaufs mit "MC_GearInPos" (A1) wird über "Exe" ein "MC_PhasingAbsolute"-Auftrag (A2) angestoßen. Die Leitwertverschiebung wird mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "AbsolutePhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 50.0 angezeigt. Die Bewegung der Leitachse wird nicht beeinflusst.

Nach der Leitwertverschiebung wird der "MC_PhasingAbsolute"-Auftrag (A2) über "Exe" erneut angestoßen. Da die Leitwertverschiebung (<TO>.StatusSynchronizedMotion.PhaseShift) bereits 50.0 beträgt, wird der Leitwert nicht verschoben.

MC_PhasingRelative V3

MC_PhasingRelative: Leitwert an der Folgeachse relativ verschieben V3

Beschreibung

Mit der Motion Control-Anweisung "MC_PhasingRelative" verschieben Sie den Leitwert an einer Folgeachse im Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos" relativ zur vorhandenen Leitwertverschiebung. Die Position der Leitachse wird dadurch nicht beeinflusst.

Mit den Parametern "Velocity", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung der Folgeachse. Die Dynamikwerte werden zu den Werten der Gleichlaufbewegung addiert.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse und der Folgeachse wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist über die Motion Control-Anweisung "MC_GearIn" oder "MC_GearInPos" auf die Leitachse aufsynchronisiert ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE).
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Das Ablöseverhalten für "MC_PhasingRelative"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht die Leitwertverschiebung nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_PhasingRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
PhaseShift	INPUT	LREAL	0.0	Relative Leitwertverschiebung
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck der Folgeachse bei der Leitwertverschiebung (additiv zur Gleichlaufbewegung) Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Leitwertverschiebung ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
CoveredPhaseShift	OUTPUT	LREAL	0.0	So lange "Busy" = TRUE: Anzeige der bisher ausgefahrenen Leitwertverschiebung	

Relative Leitwertverschiebung starten

Um mit der Motion Control-Anweisung "MC_PhasingRelative" eine relative Leitwertverschiebung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "PhaseShift" die relative Leitwertverschiebung an.
3. Starten Sie den "MC_PhasingRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "CoveredPhaseShift", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_GearInPos V3 (Seite 3457)

MC_GearIn V3 (Seite 3453)

MC_PhasingRelative: Funktionsdiagramm V3

Funktionsdiagramm: Leitwert relativ verschieben

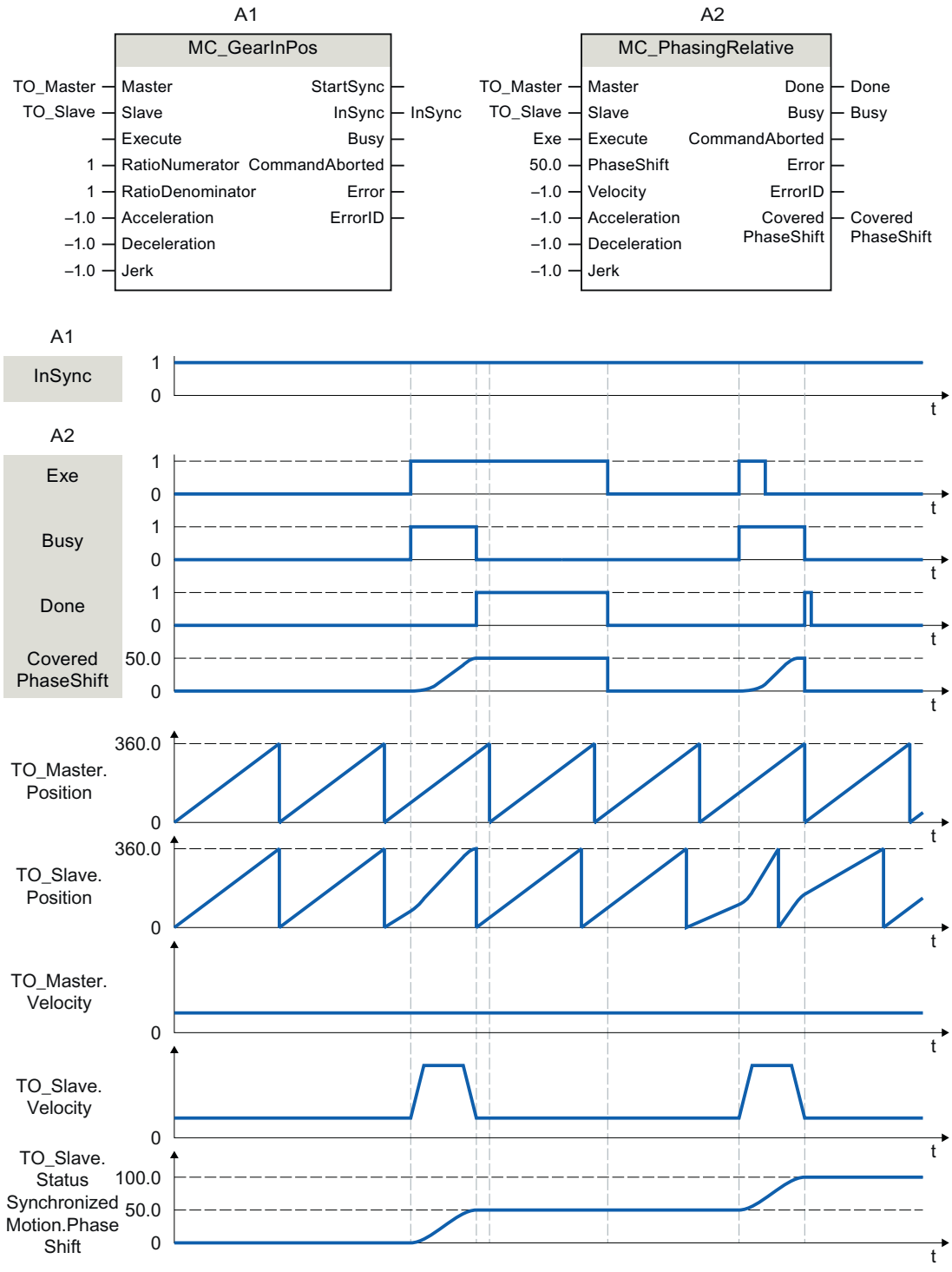

Während eines aktiven Getriebegleichlaufs mit "MC_GearInPos" (A1) wird über "Exe" ein "MC_PhasingRelative"-Auftrag (A2) angestoßen. Die Leitwertverschiebung wird mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "CoveredPhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 50.0 angezeigt. Die Bewegung der Leitachse wird nicht beeinflusst.

Nach der Leitwertverschiebung wird der "MC_PhasingRelative"-Auftrag (A2) über "Exe" erneut angestoßen. Die Leitwertverschiebung wird erneut mit der additiv zur Gleichlaufbewegung vorgegebenen Dynamik herausgefahren. Über "Done" wird angezeigt, dass der Leitwert erfolgreich verschoben wurde. In "CoveredPhaseShift" wird die über den Auftrag herausgefahrene Leitwertverschiebung 50.0 angezeigt. In der Variable des Technologieobjekts <TO>.StatusSynchronizedMotion.PhaseShift wird die absolute Leitwertverschiebung 100.0 angezeigt.

MC_CamIn V3

MC_CamIn: Kurvenscheibengleichlauf starten V3

Beschreibung

Mit der Motion Control-Anweisung "MC_CamIn" starten Sie einen Kurvenscheibengleichlauf zwischen einer Leitachse und einer Folgeachse.

Der Gleichlauf wird vorlaufend zur vorgegebenen Synchronposition der Leitachse aufsynchronisiert. Die Synchronposition der Leitachse, ab der Leit- und Folgeachse synchron verfahren, ergibt sich aus der Anfangsposition der Kurvenscheibe sowie den Parametern "MasterSyncPosition" und "MasterOffset".

Mit dem Parameter "MasterSyncPosition" geben Sie die Position in der Kurvenscheibe bezogen auf den Kurvenscheibenanfang vor, ab der die Kurvenscheibe durchfahren wird. Wenn die gesamte Kurvenscheibe durchfahren werden soll, geben Sie in "MasterSyncPosition" den Wert 0.0 (Defaultwert) an.

Mit dem Parameter "MasterOffset" verschieben Sie die Leitwerte der Kurvenscheibe. Damit verschieben Sie die absolut definierte Kurve in den applikativ benötigten Positionsbereich.

Folgendes Bild zeigt prinzipiell die Wirkung der Leit- und Folgewertverschiebung sowie von "MasterSyncPosition":

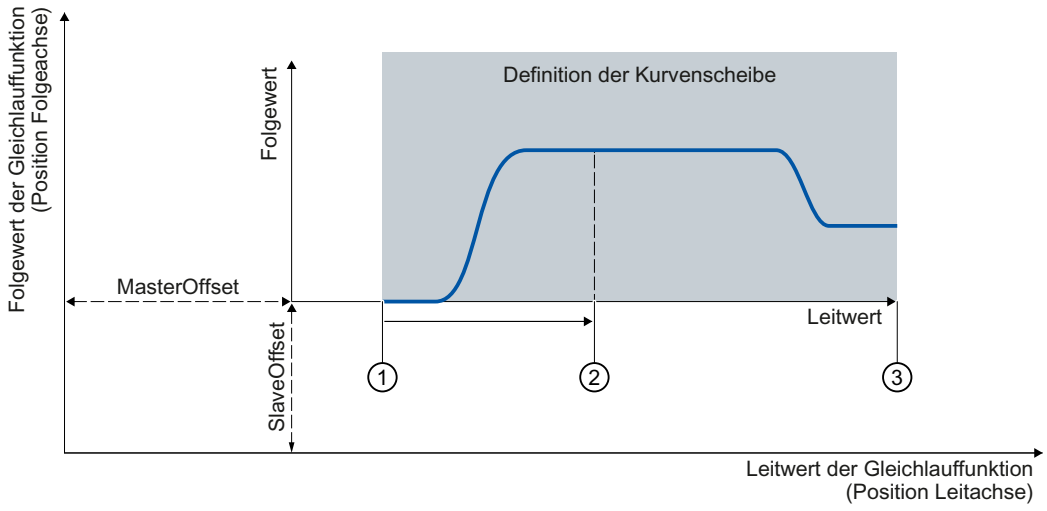

Folgendes Bild zeigt prinzipiell die Wirkung der Skalierung der Kurvenscheibe mit den Parametern "MasterScaling" und "SlaveScaling":

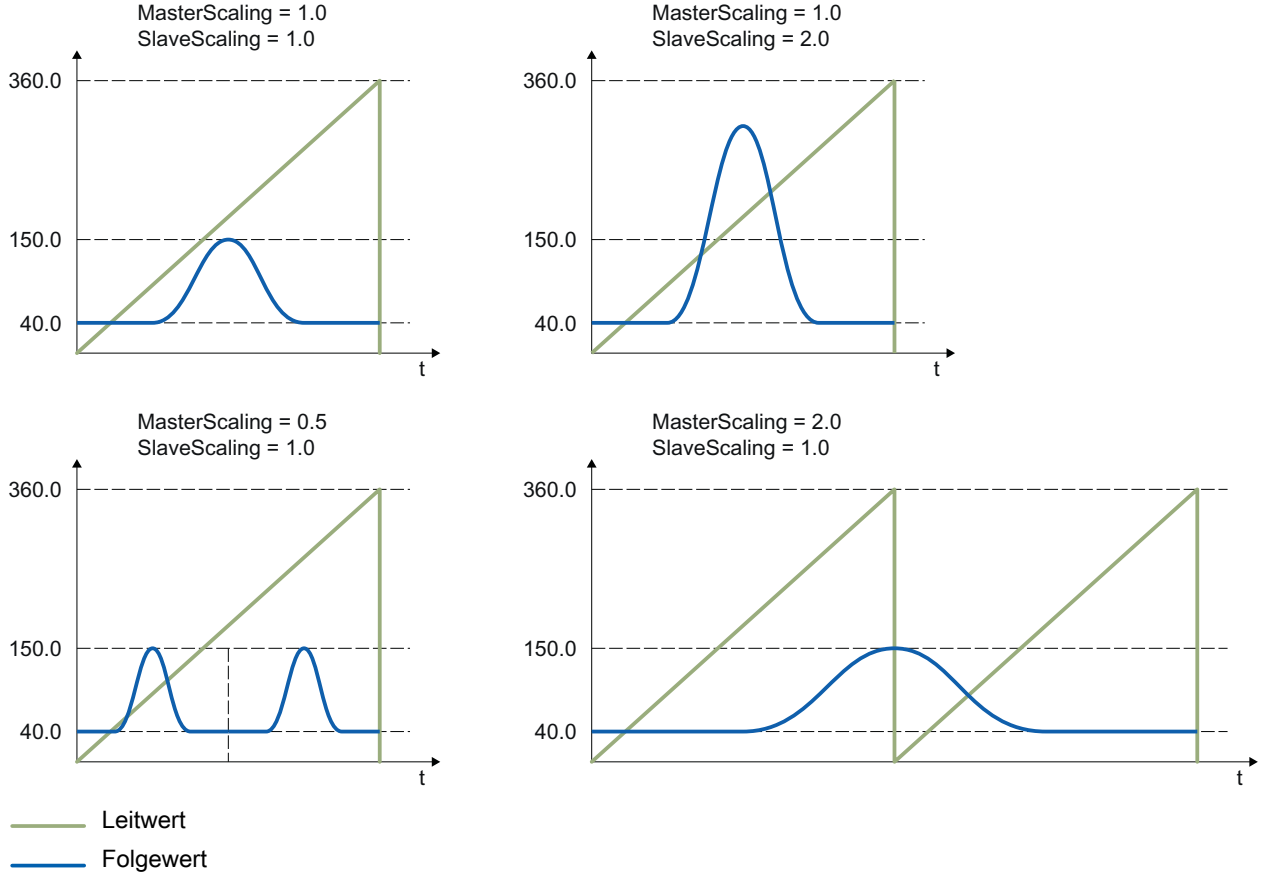

Folgende Arten des Aufsynchronisierens sind möglich:

- **Aufsynchronisieren über Dynamikparameter**
  (SyncProfileReference = 0)
- **Aufsynchronisieren über Leitwertweg**
  (SyncProfileReference = 1)

Der Gleichlauf lässt sich sowohl bei Stillstand als auch bei Bewegung der Leitachse oder Folgeachse starten.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Die Technologieobjekte der Leitachse, der Folgeachse und der Kurvenscheibe wurden korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse, Gleichlaufachse oder ein Externer Geber.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.
- Die Kurvenscheibe ist mit "MC_InterpolateCam" interpoliert.

Ablöseverhalten

Das Ablöseverhalten für "MC_CamIn"-Aufträge ist im Kapitel Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490) beschrieben.

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_CamIn":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Master	INPUT	TO_Axis	-	Technologieobjekt der Leitachse
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
MasterOffset	INPUT	LREAL	0.0	Verschiebung der Leitwerte der Kurvenscheibe (Das Technologieobjekt Kurvenscheibe wird nicht verändert.)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
SlaveOffset	INPUT	LREAL	0.0	Verschiebung der Folgewerte der Kurvenscheibe (Das Technologieobjekt Kurvenscheibe wird nicht verändert.)
MasterScaling	INPUT	LREAL	1.0	Skalierung der Leitwerte der Kurvenscheibe (Das Technologieobjekt Kurvenscheibe wird nicht verändert.)
SlaveScaling	INPUT	LREAL	1.0	Skalierung der Folgewerte der Kurvenscheibe (Das Technologieobjekt Kurvenscheibe wird nicht verändert.)
MasterSyncPosition	INPUT	LREAL	0.0	Startpunkt in der Kurvenscheibe/Abstand zum Kurvenscheibenanfang mit Bezug auf die skalierte Kurvenscheibe Der Wert muss innerhalb des Leitwertbereichs der Kurvenscheibe liegen.
SyncProfileReference	INPUT	DINT	1	Synchronisierprofil 0 Aufsynchronisieren über Dynamikparameter 1 Aufsynchronisieren über Leitwertweg
MasterStartDistance	INPUT	LREAL	0.0	Leitwertweg (bei "SyncProfileReference" = 1)
Velocity	INPUT	LREAL	-1.0	Geschwindigkeit (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung (bei "SyncProfileReference" = 0) Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck (bei "SyncProfileReference" = 0) Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
ApplicationMode	INPUT	DINT	0	Anwendung der Kurvenscheibe	
				0	einmalig/nicht zyklisch
				1	zyklisch (folgewertseitig absolut)
				2	zyklisch anhängend (folgewertseitig stetig anhängend)
SyncDirection	INPUT	DINT	3	Richtung des Aufsynchronisierens (wirksam bei Achsen mit aktivierter Modulo-Einstellung)	
				1	Positive Richtung Beim Aufsynchronisieren darf die Folgeachse nur in positive Richtung fahren.
				2	Negative Richtung Beim Aufsynchronisieren darf die Folgeachse nur in negative Richtung fahren.
				3	Kürzester Weg Beim Aufsynchronisieren sind Richtungsänderungen der Folgeachse erlaubt.
StartSync	OUTPUT	BOOL	FALSE	TRUE	Die Folgeachse wird auf die Leitachse aufsynchronisiert.
InSync	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
EndOfProfile	OUTPUT	BOOL	FALSE	TRUE	Das Ende der Kurvenscheibe wurde erreicht. Wird bei zyklischer Verwendung der Kurvenscheibe für mindestens einen Aufruf von "MC_CamIn" im Anwenderprogramm angezeigt.

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_CamIn" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse, die zu verwendende Kurvenscheibe und die Synchronposition an.
3. Starten Sie den "MC_CamIn"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InSync" den Wert TRUE zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Bei zyklischer Anwendung der Kurvenscheibe zeigen "InSync" und "Busy" so lange den Wert TRUE, bis der "MC_CamIn"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird. Bei nicht zyklischer Anwendung der Kurvenscheibe werden "InSync" und "Busy" auf den Wert FALSE gesetzt, wenn der Parameter "EndOfProfile" auf den Wert TRUE gesetzt wird.

Siehe auch

Ablöseverhalten von Motion Control-Aufträgen V3 (Seite 3490)

MC_CamIn: Funktionsdiagramm V3

Funktionsdiagramm: Aufsynchronisieren und Umschalten der Kurvenscheibe

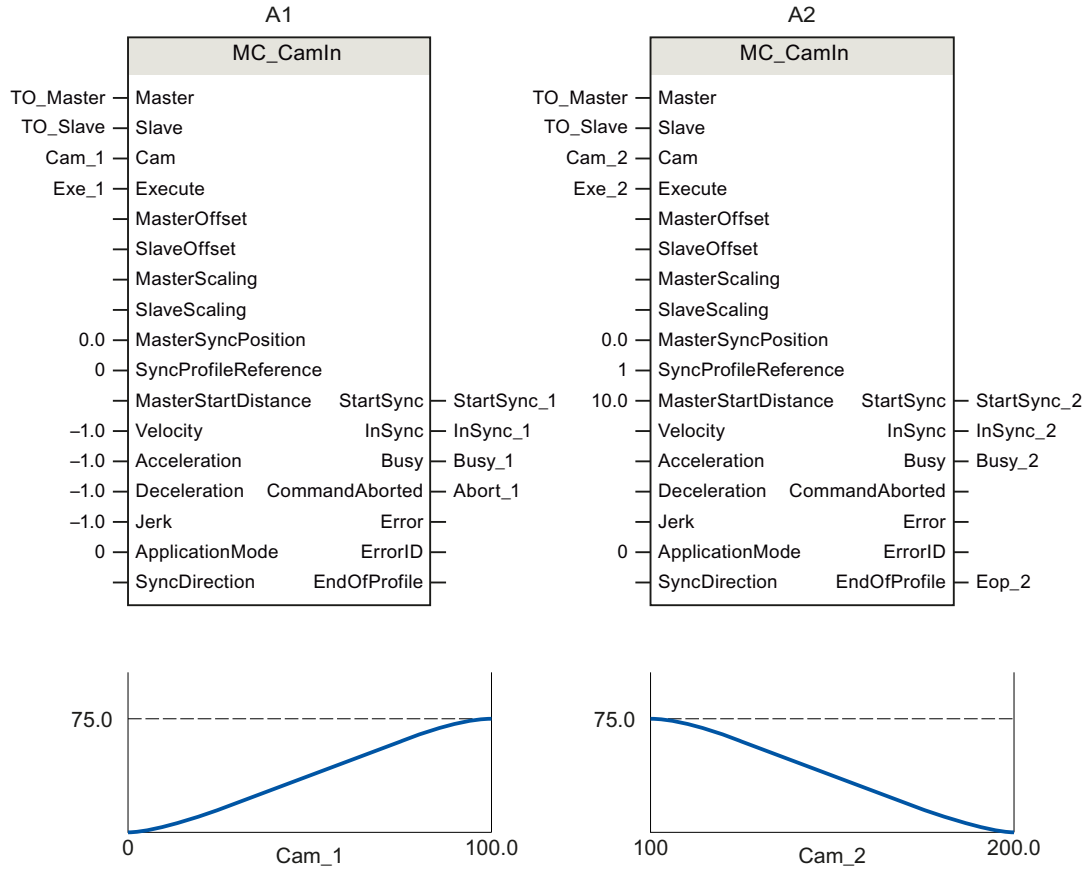

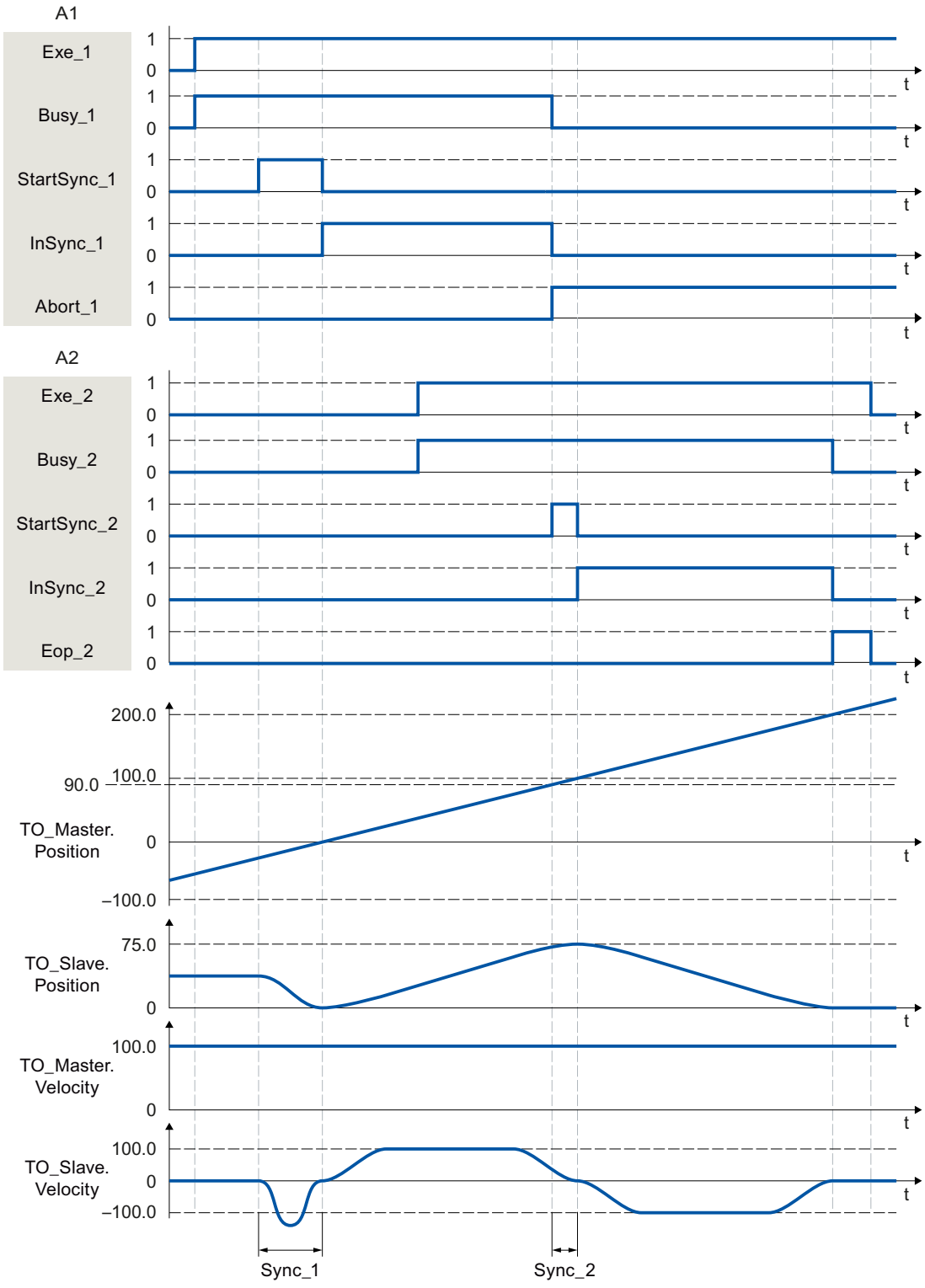

Über "Exe_1" wird ein "MC_CamIn"-Auftrag (A1) angestoßen. Über "StartSync_1" wird der Start des Aufsynchronisierens angezeigt. Die Folgeachse (TO_Slave) wird im Bereich "Sync_1" über die vorgegebenen Dynamikparameter auf die Kurvenscheibe (Cam_1)

aufsynchronisiert. Über "InSync_1" gemeldet, dass die Folgeachse aufsynchronisiert ist und synchron zur Leitachse fährt.

Der Gleichlauf wird durch einen weiteren "MC_CamIn"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Über "StartSync_2" wird der Start des Aufsynchronisierens angezeigt. Die Folgeachse wird im Bereich "Sync_2" über den vorgegebenen Leitwertweg auf die neue Kurvenscheibe (Cam_2) aufsynchronisiert. Im Bereich "Sync_2" folgt die Achse nicht der Kurvenscheibe "Cam_1". Die Achse wird mit der vom System berechneten Dynamik zur Synchronposition von A2 verfahren. Über "InSync_2" gemeldet, dass die Folgeachse aufsynchronisiert ist und synchron zur Leitachse fährt.

MC_SynchronizedMotionSimulation V3

MC_SynchronizedMotionSimulation: Gleichlauf in Simulation setzen V3

Beschreibung

Mit der Motion Control-Anweisung "MC_SynchronizedMotionSimulation" setzen Sie einen aktiven Gleichlauf an einer Folgeachse in Simulation. Damit bleibt ein Gleichlauf beim Sperren der Folgeachse mit "MC_Power" aktiv. Die Folgeachse muss nach der erneuten Freigabe nicht neu aufsynchronisieren.

Mit dem Start eines "MC_SynchronizedMotionSimulation"-Auftrags wird die Sollgeschwindigkeit aus dem Gleichlauf auf null gesetzt. Wenn beim Start der Simulation eine überlagerte Bewegung an der Folgeachse aktiv ist, werden die Sollwerte dieser Überlagerung weiterhin ausgegeben.

Sollwerte von Bewegungsaufträgen, die während der Gleichlaufsimulation gestartet werden, werden an den Antrieb ausgegeben.

Wenn die Position der Folgeachse beim Beenden der Simulation gegenüber der Position beim Start der Simulation abweicht, führt dies zu einem Sollwertsprung.

Anwendbar auf

- Gleichlaufachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Folgeachse ist eine Gleichlaufachse.
- Am Technologieobjekt ist ein Gleichlauf aktiv im Status "Synchron" (<TO>.StatusWord.X22 = TRUE).

Ablöseverhalten

Der "MC_SynchronizedMotionSimulation"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen. Der simulierte Gleichlauf bleibt auch beim Sperren der Folgeachse mit "MC_Power.Enable" = FALSE aktiv.

Ein Restart des Technologieobjekts beendet die Simulation und bricht den Gleichlauf ab.

Der Start eines "MC_SynchronizedMotionSimulation"-Auftrags bricht keine anderen Motion Control-Aufträge ab. Mit "MC_SynchronizedMotionSimulation.Enable" = TRUE werden Gleichlaufaufträge abgelehnt.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_SynchronizedMotionSimulation":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Slave	INPUT	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse	
Enable	INPUT	BOOL	FALSE	TRUE	Der Gleichlauf wird in Simulation gesetzt.
				FALSE	Die Simulation des Gleichlaufs wird beendet.
InSimulation	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf in Simulation
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf beim Sperren der Folgeachse nicht beenden

Um mit der Motion Control-Anweisung "MC_SynchronizedMotionSimulation" einen Gleichlauf beim Sperren der Folgeachse nicht zu beenden, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Slave" die Folgeachse an.
3. Setzen Sie die Leitachse in den Stillstand (z. B. mit "MC_Halt").
4. Setzen Sie den Gleichlauf an der angegebenen Folgeachse mit "MC_SynchronizedMotionSimulation.Enable" = TRUE in Simulation.
5. Wenn der Parameter "InSimulation" den Wert TRUE zeigt, sperren Sie die Folgeachse. Der Gleichlauf bleibt in der Simulation aktiv.
6. Um den Gleichlauf nach der Freigabe der Folgeachse wieder aus der Simulation zurückzusetzen, beenden Sie die Gleichlaufsimulation mit "MC_SynchronizedMotionSimulation.Enable" = FALSE.
   Die Gleichlaufsimulation wird beendet. Die Folgeachse folgt der Leitachse, ohne neu aufzusynchronisieren.

Kurvenscheibe

MC_InterpolateCam V3

MC_InterpolateCam: Kurvenscheibe interpolieren V3

Beschreibung

Mit der Motion Control-Anweisung "MC_InterpolateCam" interpolieren Sie eine Kurvenscheibe.

Bei der Interpolation werden die Lücken zwischen den definierten Stützpunkten und Segmenten der Kurvenscheibe geschlossen. Die Kurvenscheibe wird vom minimalen Wert im Definitionsbereich bis zum maximalen Wert interpoliert. Nach der Interpolation ist jedem Wert im Definitionsbereich der Kurvenscheibe ein eindeutiger Wert im Wertebereich zugeordnet.

Die Interpolationsart bestimmt, wie fehlende Bereiche interpoliert werden. Die Interpolationsart stellen Sie in der Konfiguration des Technologieobjekts ein. Folgende Interpolationsarten sind möglich:

- Lineare Interpolation
- Interpolation mit kubischen Splines
- Interpolation mit Bézier-Splines

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe wird aktuell nicht verwendet, z. B. für einen Kurvenscheibengleichlauf.

Ablöseverhalten

Der "MC_InterpolateCam"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.

Der Start eines "MC_InterpolateCam"-Auftrags bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_InterpolateCam":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Done	OUTPUT	BOOL	FALSE	TRUE	Die Kurvenscheibe ist interpoliert.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Kurvenscheibe interpolieren

Um mit der Motion Control-Anweisung "MC_InterpolateCam" eine Kurvenscheibe zu interpolieren, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parametern "Cam" die zu interpolierende Kurvenscheibe an.
3. Starten Sie den "MC_InterpolateCam"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Die Kurvenscheibe wird interpoliert. Wenn der Parameter "Done" den Wert TRUE zeigt, ist die Interpolation abgeschlossen.

MC_GetCamFollowingValue V3

MC_GetCamFollowingValue: Folgewert einer Kurvenscheibe auslesen V3

Beschreibung

Mit der Motion Control-Anweisung "MC_GetCamFollowingValue" lesen Sie aus einer Kurvenscheibe zu einem Leitwert den Folgewert sowie die erste und zweite Ableitung des Folgewerts aus.

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe ist interpoliert.

Ablöseverhalten

Der "MC_GetCamFollowingValue"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.

Der Start eines "MC_GetCamFollowingValue"-Auftrags bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GetCamFollowingValue":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
LeadingValue	INPUT	LREAL	0.0	Leitwert, zu dem der Folgewert ausgelesen wird	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Folgewert wurde ausgelesen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
Value	OUTPUT	LREAL	-	Ausgelesener Folgewert (Position) (gültig bei "Done" = TRUE)	
FirstDerivative	OUTPUT	LREAL	-	Erste Ableitung am ausgelesenen Folgewert (gültig bei "Done" = TRUE)	
SecondDerivative	OUTPUT	LREAL	-	Zweite Ableitung am ausgelesenen Folgewert (gültig bei "Done" = TRUE)	

Folgewert auslesen

Um mit der Motion Control-Anweisung "MC_GetCamFollowingValue" einen Folgewert aus einer Kurvenscheibe auszulesen, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Kurvenscheibe und den Leitwert an.
3. Starten Sie den "MC_GetCamFollowingValue"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Wenn der Parameter "Done" den Wert TRUE zeigt, wurde der Folgewert ausgelesen. Der Folgewert und die Ableitungen werden an den Parametern "Value", "FirstDerivative" und "SecondDerivative" ausgegeben.

MC_GetCamLeadingValue V3

MC_GetCamLeadingValue: Leitwert einer Kurvenscheibe auslesen V3 (S7-1500T)

Beschreibung

Mit der Motion Control-Anweisung "MC_GetCamLeadingValue" lesen Sie aus einer Kurvenscheibe den Leitwert aus, der zu einem Folgewert definiert ist.

Da für verschiedene Leitwerte gleiche Folgewerte definiert sein können, lässt sich am Parameter "ApproachValue" eine Näherung des Leitwerts angeben.

Anwendbar auf

- Kurvenscheibe

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Kurvenscheibe ist interpoliert.

Ablöseverhalten

Der "MC_GetCamLeadingValue"-Auftrag wird durch keinen anderen Motion Control-Auftrag abgebrochen.

Der Start eines "MC_GetCamLeadingValue"-Auftrags bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GetCamLeadingValue":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Cam	INPUT	TO_Cam	-	Technologieobjekt der Kurvenscheibe
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
FollowingValue	INPUT	LREAL	0.0	Folgewert, zu dem der Leitwert ausgelesen wird
ApproachLeadingValue	INPUT	LREAL	0.0	Näherungswert zum gesuchten Leitwert Wenn der Folgewert mehrfach in der Kurvenscheibe verwendet ist, lässt sich damit der gesuchte Leitwert eingrenzen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Done	OUTPUT	BOOL	FALSE	TRUE	Der Leitwert wurde ausgelesen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	
Value	OUTPUT	LREAL	-	Ausgelesener Leitwert (Position) (gültig bei "Done" = TRUE)	

Leitwert auslesen

Um mit der Motion Control-Anweisung "MC_GetCamLeadingValue" einen Leitwert aus einer Kurvenscheibe auszulesen, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Kurvenscheibe, den Folgewert und den Näherungswert zum gesuchten Leitwert an.
3. Starten Sie den "MC_GetCamLeadingValue"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Wenn der Parameter "Done" den Wert TRUE zeigt, wurde der Folgewert ermittelt. Der Leitwert wird am Parameter "Value" ausgegeben.

Momentendaten

MC_TorqueLimiting V3

MC_TorqueLimiting: Kraft-/Momentenbegrenzung / Festanschlagserkennung aktivieren und deaktivieren

Beschreibung

Aktivieren und parametrieren Sie mit der Motion Control-Anweisung "MC_TorqueLimiting" eine Kraft-/Momentenbegrenzung bzw. eine Festanschlagserkennung.

Zusammen mit einem lagegeregelten Bewegungsauftrag kann mit der Festanschlagserkennung ein "Fahren auf Festanschlag" realisiert werden.

In der Konfiguration der Achse können Sie konfigurieren, ob die Kraft-/Momentenbegrenzung auf die Antriebsseite oder auf die Lastseite bezogen werden soll.

Die Funktionen der Motion Control-Anweisung "MC-TorqueLimiting" können vor und während eines Bewegungsauftrages aktiviert und deaktiviert werden.

Kraft-/Momentenbegrenzung anwendbar auf

- Drehzahlachse
- Positionierachse
- Gleichlaufachse

Voraussetzung für die Kraft-/Momentenbegrenzung

- Das Technologieobjekt und das Bezugsmoment des Antriebs wurden korrekt konfiguriert.
- Der Antrieb muss die Kraft-/Momentenreduzierung unterstützen. Nur PROFIdrive-Antriebe mit SIEMENS Telegramm 10x unterstützen die Kraft-/Momentenbegrenzung.
- Am Technologieobjekt stehen keine Freigabe verhindernde Fehler an (das Technologieobjekt muss nicht freigegeben sein).

Festanschlagserkennung anwendbar auf

- Positionierachse
- Gleichlaufachse

Voraussetzung für Festanschlagserkennung

- Die Festanschlagserkennung kann nur auf lagegeregelte Achsen angewandt werden. Zur Festanschlagserkennung muss die Achse lagegeregelt freigegeben sein; Bewegungsaufträge müssen lagegeregelt ausgeführt werden.
- Das Technologieobjekt wurde korrekt konfiguriert.
- Beim Einsatz eines Antriebs und Telegramms, welche die Kraft-/Momentenbegrenzung unterstützten, muss das Bezugsmoment des Antriebs korrekt am Technologieobjekt konfiguriert sein.
- Am Technologieobjekt stehen keine Freigabe verhindernde Fehler an (das Technologieobjekt muss nicht freigegeben sein).

Ablöseverhalten

- Ein MC_TorqueLimiting-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein neuer MC_TorqueLimiting-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_TorqueLimiting":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	INPUT	TO_SpeedAxis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Funktion entsprechend Eingangsparameter Mode aktivieren

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Limit	INPUT	LREAL	-1.0	Wert der Kraft-/Momentenbegrenzung (in der konfigurierten Einheit) Unterstützen Antrieb und Telegramm keine Kraft-/Momentenbegrenzung, so ist der angegebene Wert belanglos.	
				≥ 0	Verwende den am Parameter angegebenen Wert (Wert "0" ist zulässig)
				< 0	Verwende den im Konfigurationsfenster "Momentenbegrenzung" konfigurierten Wert Variable Grenzwert Moment: <TO>.TorqueLimiting.LimitDefaults.Torque Variable Grenzwert Kraft: <TO>.TorqueLimiting.LimitDefaults.Force
Mode	INPUT	DINT	0	0	Kraft/Momentenbegrenzung
				1	Festanschlagserkennung Unterstützen Antrieb und Telegramm die Kraft/Momentenbegrenzung, so wird diese angewandt.
InClamping	OUTPUT	BOOL	FALSE	TRUE	Mode = 1: Der Antrieb wird am Festanschlag gehalten (Klemmung), die Achsposition befindet sich innerhalb der Positioniertoleranz.
InLimitation	OUTPUT	BOOL	FALSE	TRUE	Mode = 0 und 1: Der Antrieb arbeitet an der Kraft-/Momentengrenze.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Änderungen an den Eingangsparametern "Limit" und "Mode" werden auch bei "Enable = TRUE" beim zyklischen Aufruf der Motion Control-Anweisung übernommen.

MC_TorqueLimiting: Funktionsdiagramm V3

Funktionsdiagramm: Anhalten einer Achse, wenn die Drehmomentgrenze erreicht wird

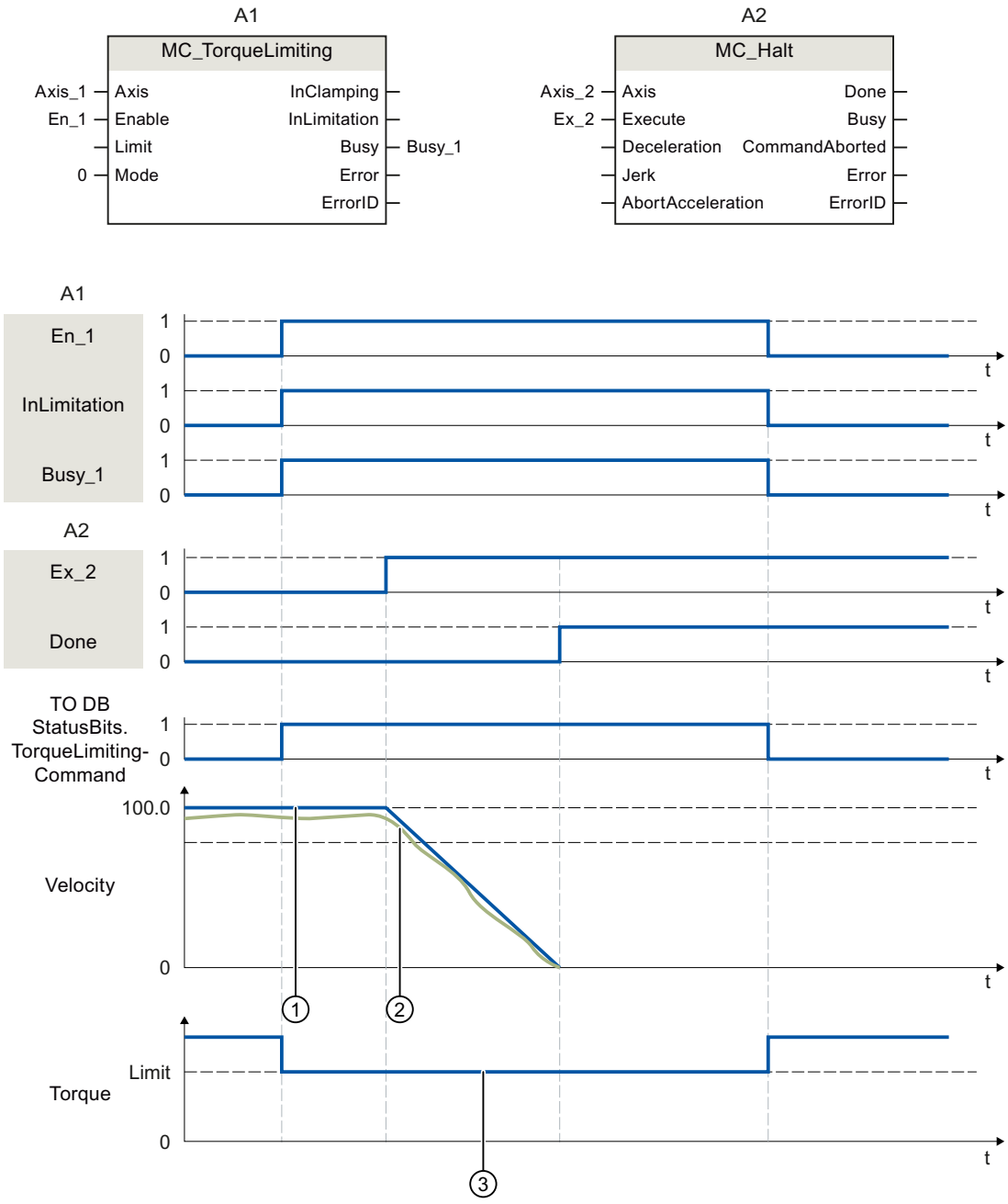

① Sollgeschwindigkeit
② Istgeschwindigkeit
③ Drehmoment

Eine Motion Control-Anweisung "MC_Halt" wird auf eine Achse mit aktiver Drehmomentbegrenzung "MC_TorqueLimiting" ausgeführt. Die Drehmomentbegrenzung

"MC_TorqueLimiting.Enable" ist weiterhin TRUE (aktiv) und ein evtl. aufgebauter Schleppabstand bleibt erhalten und wird abgebaut. Wenn die Istgeschwindigkeit "0.0" und die Mindestverweildauer im Stillstandsfenster abgelaufen ist, zeigt "MC_Halt.Done" = TRUE. Bei eingeschalteter Positionierüberwachung wird auch das Erreichen der Zielposition überwacht.

Funktionsdiagramm: Drehmomentbegrenzung mit Festanschlagserkennung (Mode = 1)

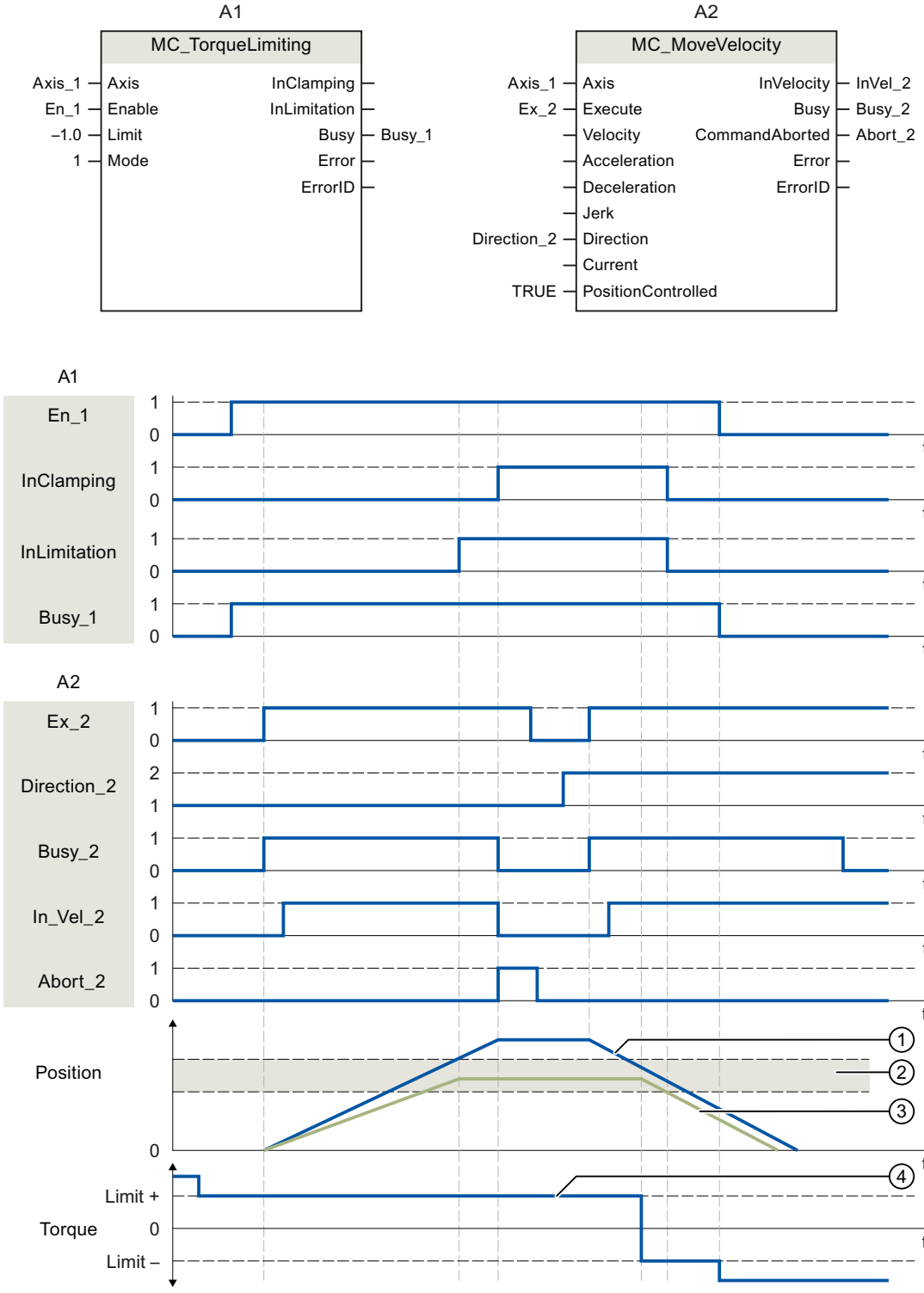

① Istposition bleibt stehen

② Schleppabstandsgrenze erreicht (Sollposition)

③ Positioniertoleranz
④ Drehmoment

Eine Motion Control-Anweisung "MC_MoveVelocity" wird auf eine Achse mit aktiver Drehmomentbegrenzung "MC_TorqueLimiting" ausgeführt. Die Drehmomentbegrenzung "MC_TorqueLimiting.Enable" ist weiterhin TRUE (aktiv) und der konfigurierte Schleppabstand wird abgebaut. Bei Erreichen der Schleppabstandsgrenze wird Motion Control-Anweisung "MC_MoveVelocity" mit "Abort" = TRUE abgebrochen. Der Antrieb wird am Festanschlag gehalten (Klemmung), die Achsposition befindet sich innerhalb der Positioniertoleranz. Die Motion Control-Anweisung "MC_MoveVelocity" wird über "Execute" = TRUE erneut aufgerufen und die Achse mit konstanter Geschwindigkeit bewegt.

Ablöseverhalten von Motion Control-Aufträgen V3

Referenzier- und Bewegungsaufträge

Folgende Tabelle zeigt, wie ein neuer Motion Control-Auftrag auf laufende Referenzier- und Bewegungsaufträge wirkt:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_Home Mode = 2, 8, 10	MC_Home Mode = 3, 5	MC_Halt	MC_Move Absolute	MC_Move Relative	MC_Move Velocity	MC_Move Jog	MC_Move Super imposed
MC_Home Mode = 3, 5	A	A	A	A	A	A	A	A
MC_Home Mode = 9	A	-	-	-	-	-	-	-
MC_Halt	-	A	A	A	A	A	A	A
MC_Move Absolute	-	A	A	A	A	A	A	A
MC_Move Relative	-	A	A	A	A	A	A	A
MC_Move Velocity	-	A	A	A	A	A	A	A
MC_MoveJog	-	A	A	A	A	A	A	A
MC_Move Superimposed	-	-	-	-	-	-	-	A
MC_GearIn	-	A	A	A	A	A	A	A
MC_GearInPos wartend (Busy = TRUE, StartSync = FALSE, InSync = FALSE)	-	-	-	-	-	-	-	-
MC_GearInPos aktiv (Busy = TRUE, StartSync oder InSync = TRUE)	-	A	A	A	A	A	A	A

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_Home Mode = 2, 8, 10	MC_Home Mode = 3, 5	MC_Halt	MC_Move Absolute	MC_Move Relative	MC_Move Velocity	MC_Move Jog	MC_Move Super imposed
MC_CamIn wartend (Busy = TRUE, StartSync = FALSE, InSync = FALSE)	-	-	-	-	-	-	-	-
MC_CamIn aktiv (Busy = TRUE, StartSync oder In-Sync = TRUE)	-	A	A	A	A	A	A	A

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen. Die aktuelle Dynamik wird auf die Dynamikwerte des neuen Auftrags überschliffen.

- Keine Auswirkung, laufender Auftrag wird weiterhin ausgeführt

Gleichlaufaufträge

Folgende Tabelle zeigt, wie ein neuer Motion Control-Auftrag zur Bewegung der Achse auf laufende Gleichlaufaufträge wirkt:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_GearIn	MC_GearInPos wartend (Busy = TRUE, InSync = FALSE, StartSync = FALSE)	MC_GearInPos aktiv (Busy = TRUE, StartSync oder In-Sync = TRUE)	MC_Phasing Absolute	MC_Phasing Relative	MC_CamIn wartend (Busy = TRUE, InSync = FALSE, StartSync = FALSE)	MC_CamIn aktiv (Busy = TRUE, StartSync oder In-Sync = TRUE)
MC_Home Mode = 3, 5	A	-	-	-	-	-	-
MC_Halt	A	-	A	A	A	-	A
MC_MoveAbsolute	A	-	A	A	A	-	A
MC_MoveRelative	A	-	A	A	A	-	A
MC_MoveVelocity	A	-	A	A	A	-	A
MC_MoveJog	A	-	A	A	A	-	A
MC_MoveSuperimposed	-	-	-	-	-	-	-
MC_GearIn	A	A	A	A	A	A	A
MC_GearInPos wartend (Busy = TRUE, StartSync = FALSE, InSync = FALSE)	-	A	-	-	-	A	-
MC_GearInPos aktiv (Busy = TRUE, StartSync oder InSync = TRUE)	A	A	A	A	A	A	A
MC_PhasingAbsolute	-	-	-	A	A	-	-
MC_PhasingRelative	-	-	-	A	A	-	-

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_GearIn	MC_GearInPos wartend (Busy = TRUE, InSync = FALSE, StartSync = FALSE)	MC_GearInPos aktiv (Busy = TRUE, StartSync oder InSync = TRUE)	MC_Phasing Absolute	MC_Phasing Relative	MC_CamIn wartend (Busy = TRUE, InSync = FALSE, StartSync = FALSE)	MC_CamIn aktiv (Busy = TRUE, StartSync oder InSync = TRUE)
MC_CamIn wartend (Busy = TRUE, StartSync = FALSE, InSync = FALSE)	-	A	-	-	-	A	-
MC_CamIn aktiv (Busy = TRUE, StartSync oder InSync = TRUE)	A	A	A	A	A	A	A

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen. Die aktuelle Dynamik wird auf die Dynamikwerte des neuen Auftrags überschliffen.

- Keine Auswirkung, laufender Auftrag wird weiterhin ausgeführt

Ein wartender Gleichlaufauftrag (Busy = TRUE, StartSync = FALSE, InSync = FALSE) bricht keine laufenden Aufträge ab. Nur ein wartender Gleichlaufauftrag ist zulässig. Ein zweiter wartender Gleichlaufauftrag bricht den ersten wartenden Gleichlaufauftrag ab.

Wenn ein Gleichlaufauftrag aktiv ist und ein Gleichlaufauftrag wartend ist, bricht ein neuer Bewegungsauftrag sowohl den aktiven als auch den wartenden Gleichlaufauftrag ab.

Messtaster

Folgende Tabelle zeigt, durch welche neuen Motion Control-Aufträge laufende Messtasteraufträge abgelöst werden:

⇒ Laufender Auftrag ⇓ Neuer Auftrag	MC_MeasuringInput	MC_MeasuringInputCyclic
MC_Home Mode = 2, 3, 5, 8, 9, 10	A	A
MC_Home Mode = 0, 1, 6, 7	-	-
MC_MeasuringInput	A	A
MC_MeasuringInputCyclic	A	A
MC_AbortMeasuringInput	A	A

A Der laufende Auftrag wird mit "CommandAborted" = TRUE abgebrochen. Der neue Auftrag wird ausgeführt.

- Keine Auswirkung, laufender Auftrag wird weiterhin ausgeführt

S7-1500 Motion Control V2

MC_Power V2

MC_Power: Technologieobjekt freigeben, sperren V2

Beschreibung

Mit der Motion Control-Anweisung "MC_Power" wird ein Technologieobjekt freigegeben bzw. gesperrt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.

Ablöseverhalten

- Ein "MC_Power"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein "MC_Power"-Auftrag mit dem Parameter "Enable" = TRUE gibt ein Technologieobjekt frei und bricht somit keine anderen Motion Control-Anweisungen ab.
- Mit dem Sperren des Technologieobjekts (Parameter "Enable" = FALSE) werden alle Bewegungsaufträge am zugehörigen Technologieobjekt entsprechend dem gewählten "StopMode" abgebrochen. Dieser Vorgang kann vom Anwender nicht abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Power":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	InOut	TO_Axis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Das Technologieobjekt wird freigegeben.
				FALSE	Das Technologieobjekt wird gesperrt. Alle laufenden Aufträge am Technologieobjekt werden entsprechend dem parametrierten "StopMode" abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
StopMode	INPUT	INT	0	Für Technologieobjekt Externer Geber nicht relevant. Wenn Sie ein Technologieobjekt mit einer fallenden Flanke am Parameter "Enable" sperren, bremst die Achse gemäß dem gewählten "StopMode" ab.	
				0	Notstopp Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Notstopp-Rampe" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicDefaults.EmergencyDeceleration)
				1	Sofort-Aus Beim Sperren eines Technologieobjekts wird der Sollwert Null ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
				2	Stopp mit maximalen Dynamikwerten Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicLimits.MaxDeceleration; <TO>.DynamicLimits.MaxJerk)
Status	OUTPUT	BOOL	FALSE	Freigabestatus des Technologieobjekts	
				FALSE	Gesperrt - Eine Positionier- oder Drehzahlachse nimmt keine Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind nicht aktiv. - Die Istwerte des Technologieobjekts werden nicht auf Gültigkeit überprüft.
				TRUE	Freigegeben - Eine freigegebene Positionier- oder Drehzahlachse nimmt Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind aktiv. - Die Istwerte des Technologieobjekts sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Freigeben von Technologieobjekten

Setzen Sie zum Freigeben eines Technologieobjekts den Parameter "Enable" auf TRUE.

Wenn der Parameter "Status" den Wert TRUE zeigt, ist das Technologieobjekt freigegeben.

Wenn eine Achse beim Freigeben des Technologieobjekts in Bewegung ist (Istgeschwindigkeit vorhanden), wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung auf Sollwert Null abgebremst (<TO>.DynamicLimits.MaxDeceleration). Diese Bremsrampe ist von Bewegungsaufträgen ablösbar.

Hinweis

Automatische Freigabe nach Quittierung eines Technologiealarms

Wenn das Technologieobjekt aufgrund eines Technologiealarms gesperrt wird, so wird es nach dem Beseitigen der Ursache und Quittieren des Alarms automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Parameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Sperren von Technologieobjekten

Setzen Sie zum Sperren eines Technologieobjekts den Parameter "Enable" auf FALSE.

Wenn eine Achse in Bewegung ist, wird Sie gemäß dem gewählten "StopMode" abgebremst und zum Stillstand gebracht.

Wenn die Parameter "Busy" und "Status" den Wert FALSE zeigen, ist das Sperren des Technologieobjekts abgeschlossen.

Antriebsanbindung über PROFIdrive

Bei der Ankopplung eines Antriebs über PROFIdrive werden Sollwert, Freigabe und Antriebsstatus über das PROFIdrive-Telegramm übertragen.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird das Technologieobjekt freigegeben. Der Antrieb wird gemäß PROFIdrive-Norm aktiviert.
  Wenn die Variable <TO>.StatusDrive.InOperation den Wert TRUE zeigt, ist der Antrieb bereit, Sollwerte auszuführen. Der Parameter "Status" wird auf den Wert TRUE gesetzt.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Der Antrieb wird gemäß PROFIdrive-Norm deaktiviert.

Analoge Antriebsanbindung

Der Sollwert wird über einen Analogausgang ausgegeben. Optional kann ein Freigabesignal über Digitalausgang (<TO>.Actor.Interface.EnableDriveOutput) und ein Bereitschaftssignal über Digitaleingang (<TO>.Actor.Interface.DriveReadyInput) konfiguriert werden.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird der Freigabe-Ausgang ("Enable drive output") gesetzt.
  Wenn der Antrieb das Bereitschaftssignal über den Bereit-Eingang ("Drive ready input") zurückmeldet, wird der Parameter "Status" und die Variable des Technologieobjekts <TO>.StatusDrive.InOperation auf TRUE gesetzt und der Sollwert auf den Analogausgang geschaltet.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Mit dem Erreichen von Sollwert Null wird der Freigabe-Ausgang auf FALSE gesetzt.

Weitere Informationen

Weiterführende Informationen zum Freigeben und Sperren von Technologieobjekten und Antrieben finden Sie im Anhang Funktionsdiagramme MC_Power.

MC_Power: Funktionsdiagramm V2

Funktionsdiagramm: Freigeben eines Technologieobjekts und Beispiel für Alarmreaktion

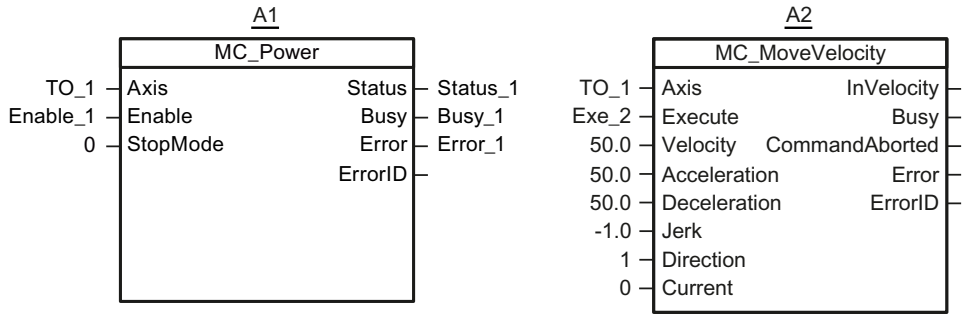

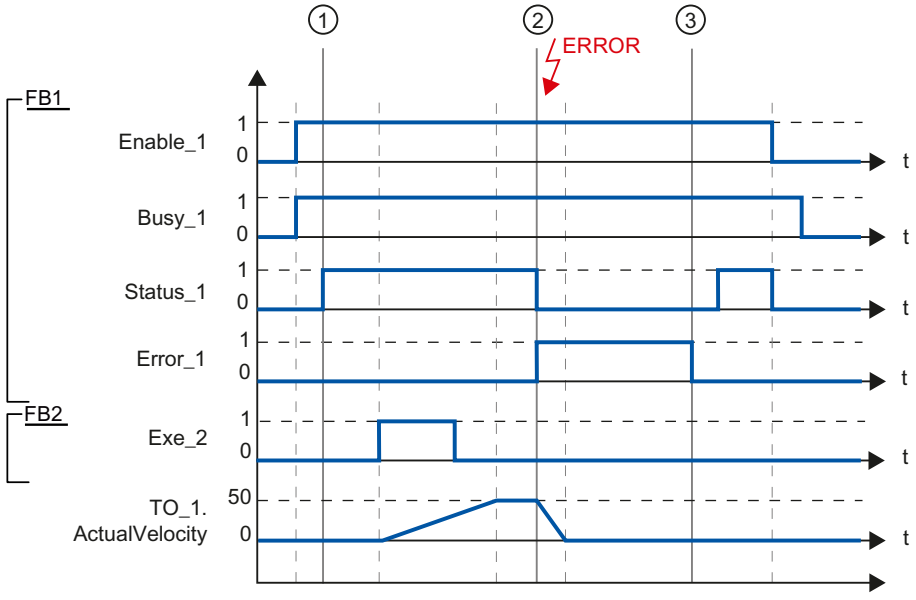

Ein Technologieobjekt wird mit "Enable_1" = TRUE freigegeben. Die erfolgreiche Freigabe kann an "Status_1" zum Zeitpunkt ① abgelesen werden. Anschließend wird die Achse mit einem "MC_MoveVelocity"-Auftrag (A2) verfahren. Der Geschwindigkeitsverlauf der Achse kann an "Velocity Axis_1" abgelesen werden.

Zum Zeitpunkt ② tritt ein Fehler am Technologieobjekt auf, der das Sperren des Technologieobjekts zur Folge hat (Alarmreaktion: Freigabe wegnehmen). Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Mit dem Sperren des Technologieobjekts wird "Status_1" rückgesetzt. Da die Achse nicht über "Enable_1" = FALSE gesperrt wurde, ist der gewählte "StopMode" nicht relevant. Die Fehlerursache wird beseitigt und der Alarm wird zum Zeitpunkt ③ quittiert.

Da "Enable_1" weiterhin gesetzt ist, wird das Technologieobjekt wieder freigegeben. Die erfolgreiche Freigabe kann an "Status_1" abgelesen werden. Abschließend wird das Technologieobjekt über "Enable_1" = FALSE gesperrt.

MC_Reset V2

MC_Reset: Alarme quittieren, Restart Technologieobjekt V2

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" quittieren Sie alle Technologie-Alarme, die im Anwenderprogramm quittierbar sind. Mit der Quittierung werden auch die "Error"- und "Warning"-Bits im Technologie-Datenbaustein zurückgesetzt.

Mit der Motion Control-Anweisung "MC_Reset" mit "Restart" = TRUE starten Sie das Neuinitialisieren (Restart) von Technologieobjekten. Mit dem Restart des Technologieobjekts werden neue Konfigurationsdaten im Technologie-Datenbaustein übernommen.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Bei einem anstehenden Technologie-Alarm wurde die Fehlerursache beseitigt.
- Für einen Restart muss das Technologieobjekt gesperrt sein.
  ("MC_Power.Status" = FALSE und "MC_Power.Busy" = FALSE)

Ablöseverhalten

- Parameter "Restart" = FALSE:
  Die Bearbeitung der Anweisung "MC_Reset" kann durch andere Motion Control-Aufträge abgebrochen werden. Der "MC_Reset"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.
- Parameter "Restart" = TRUE:
  Die Bearbeitung der Anweisung "MC_Reset" mit Parameter "Restart" = TRUE kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Reset":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Restart	INPUT	BOOL	FALSE	TRUE	"Restart" Neuinitialisierung des Technologieobjektes und Quittierung anstehender Technologie-Alarme. Das Technologieobjekt wird mit den konfigurierten Startwerten neu initialisiert.
				FALSE	Quittierung anstehender Technologie-Alarme.
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert. Restart wurde ausgeführt.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Technologie-Alarme quittieren

Gehen Sie zum Quittieren von Technologie-Alarmen folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen sie den Parameter "Restart" = FALSE.
3. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so wurde der Fehler quittiert.

Hinweis

Quittieren mit "Restart" = FALSE

Setzen Sie "Restart" = FALSE, wenn nur die Technologie-Alarmen quittiert werden sollen. Während eines Restarts kann das Technologieobjekt nicht verwendet werden.

Restart eines Technologieobjektes

Gehen Sie zum Restart eines Technologieobjektes wie folgt vor:

1. Prüfen Sie die oben genannten Vorraussetzungen.
2. Setzen sie den Parameter "Restart" = TRUE.
3. Führen Sie den Restart durch eine steigende Flanke am Parameter "Execute" aus.

Wenn der Parameter "Done" den Wert TRUE zeigt, ist der Restart des Technologieobjektes abgeschlossen.

Weitere Informationen zum Restart finden Sie im Kapitel Restart von Technologieobjekten.

MC_Home V2

MC_Home: Technologieobjekt referenzieren, Referenzpunkt setzen V2

Beschreibung

Mit der Motion Control-Anweisung "MC_Home" stellen Sie den Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung her. Der Positionsistwert am Technologieobjekt wird dabei einer Referenzmarke zugeordnet. Diese Referenzmarke repräsentiert eine bekannte mechanische Position.

Der Referenziervorgang erfolgt gemäß der am Parameter "Mode" gewählten Betriebsart und der Konfiguration unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren".

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel Versionsübersicht.

Für die Dynamikwerte Beschleunigung, Verzögerung und Ruck werden die voreingestellten Werte unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik-Voreinstellung" verwendet.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Externer Geber

Die folgende Tabelle zeigt, welche Betriebsarten mit den jeweiligen Technologieobjekten möglich sind:

Betriebsart	Positionierachse / Gleichlaufachse mit Inkrementalgeber	Positionierachse / Gleichlaufachse mit Absolutwertgeber	Externer Geber inkrementell	Externer Geber absolut
Aktives Referenzieren ("Mode" = 3, 5)	X	-	-	-
Passives Referenzieren ("Mode" = 2, 8,10)	X	-	X	-
Setzen der Istposition ("Mode" = 0)	X	X	X	X
Relative Verschiebung der Istposition ("Mode" = 1)	X	X	X	X
Absolutwertgeber referenzieren ("Mode" = 6, 7)	-	X	-	X

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- "Mode" = 2, 3, 5, 8, 10
  Das Technologieobjekt muss freigegeben sein.
- "Mode" = 0, 1, 6, 7, 8
  Die Geberwerte müssen gültig sein. (<TO>.StatusSensor[n].State = 2)

Ablöseverhalten

- Ein "MC_Home"-Auftrag zum passiven Referenzieren wird abgebrochen durch:
  - Sperren des Technologieobjekts mit "MC_Power.Enable" = FALSE
  - "MC_Home"-Auftrag mit Parameter "Mode" = 3, 5, 9
- Ein "MC_Home"-Auftrag zum passiven Referenzieren bricht keine anderen Motion Control-Aufträge ab.
- Ein "MC_Home"-Auftrag zum aktiven Referenzieren wird abgebrochen durch:
  - Sperren des Technologieobjekts "MC_Power.Enable" = FALSE
  - "MC_Home"-Auftrag "Mode" = 3, 5
  - "MC_Halt"-Auftrag
  - "MC_MoveAbsolute"-Auftrag
  - "MC_MoveRelative"-Auftrag
  - "MC_MoveVelocity"-Auftrag
  - "MC_MoveJog"-Auftrag
  - "MC_GearIn"-Auftrag
- Ein "MC_Home"-Auftrag zum aktiven Referenzieren bricht folgende laufende Motion Control-Aufträge ab:
  - "MC_Home"-Auftrag mit Parameter "Mode" = 3, 5, 8, 10
  - "MC_Halt"-Auftrag
  - "MC_MoveAbsolute"-Auftrag
  - "MC_MoveRelative"-Auftrag
  - "MC_MoveVelocity"-Auftrag
  - "MC_MoveJog"-Auftrag
  - "MC_GearIn"-Auftrag
  - "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Home":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Der angegebene Wert wird dem gewähltem "Mode" entsprechend verwendet.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Mode	INPUT	INT	0	Betriebsart	
				0	Direktes Referenzieren (Absolut) Die aktuelle Position des Technologieobjektes wird auf den Wert des Parameters "Position" gesetzt.
				1	Direktes Referenzieren (Relativ) Die aktuelle Position des Technologieobjektes wird um den Wert des Parameters "Position" verschoben.
				2	Passives Referenzieren (ohne Rücksetzen) Funktion wie "Mode" 8 mit dem Unterschied, dass der Status "referenziert" mit dem Aktivieren der Funktion **nicht** zurückgesetzt wird.
				3	Aktives Referenzieren Das TO Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem Wert des Parameters "Position".
				4	Reserviert
				5	Aktives Referenzieren (Parameter "Position" unwirksam) Das TO Positionierachse/Gleichlaufachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Aktives Referenzieren" konfigurierten Referenzpunkt. (<TO>.Homing.HomePosition)
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Position wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n]. AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Position wird auf den Wert des Parameters "Position" gesetzt. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n]. AbsEncoderOffset)
				8	Passives Referenzieren

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
					Beim Erkennen der Referenzmarke wird der Istwert auf den Wert des Parameters "Position" gesetzt.
				9	Abbruch Passives Referenzieren Ein laufender Auftrag zum passiven Referenzieren wird abgebrochen.
				10	Passives Referenzieren (Parameter "Position" unwirksam) Beim Erkennen der Referenzmarke wird der Istwert auf den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Passives Referenzieren" konfigurierten Referenzpunkt gesetzt. (<TO>.Homing.HomePosition)
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjektes wird unter folgenden Bedingungen rückgesetzt (<TO>.StatusWord.X5 (HomingDone)):

- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags mit "Mode" = 3, 5, 8, 10
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS -> NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Referenzieren eines Technologieobjekts mit "Mode" = 1..8, 10

Gehen Sie zum Referenzieren eines Technologieobjektes folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Mode" die gewünschten Referenzierfunktion an.
3. Versorgen Sie die notwendigen Parameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so ist der "MC_Home"-Auftrag gemäß dem gewählten "Mode" abgeschlossen. Der Status "Referenziert" des Technologieobjekts wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Referenziert" angezeigt (<TO>.StatusWord.X5 (HomingDone)).

Abbruch eines passiven Refenziervorgangs mit "Mode" = 9

Das Technologieobjekt wird mit "Mode" = 9 nicht referenziert. Wenn ein laufender "MC_Home"-Auftrag zum passiven Referenzieren ("Mode" = 2, 8, 10) durch einen anderen "MC_Home"-Auftrag mit "Mode" = 9 abgelöst wird, so wird der laufende Auftrag mit dem Parameter "CommandAborted" = TRUE abgebrochen. Der ablösende Auftrag mit "Mode" = 9 meldet die erfolgreiche Durchführung mit Parameter "Done" = TRUE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_Halt V2

MC_Halt: Achse anhalten V2

Beschreibung

Mit der Motion Control-Anweisung "MC_Halt" bremsen Sie eine Achse bis zum Stillstand ab.

Mit den Parametern "Jerk" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bremsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_Halt"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_Halt"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Halt":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit Null erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Abbremsen einer Achse mit "MC_Halt"

Gehen Sie zum Abbremsen einer Achse bis zum Stillstand folgendermaßen vor:

1. Prüfen Sie die oben genannten Vorraussetzungen.
2. Versorgen Sie die Parameter "Deceleration" und "Jerk" mit den gewünschten Werten.
3. Starten Sie den "MC_Halt"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt. Der Stillstand der Achse wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Stillstand" angezeigt (<TO>.StatusWord.X7 (Standstill)).

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_Halt: Funktionsdiagramm V2

Funktionsdiagramm: Anhalten einer Achse und ablösendes Auftragsverhalten

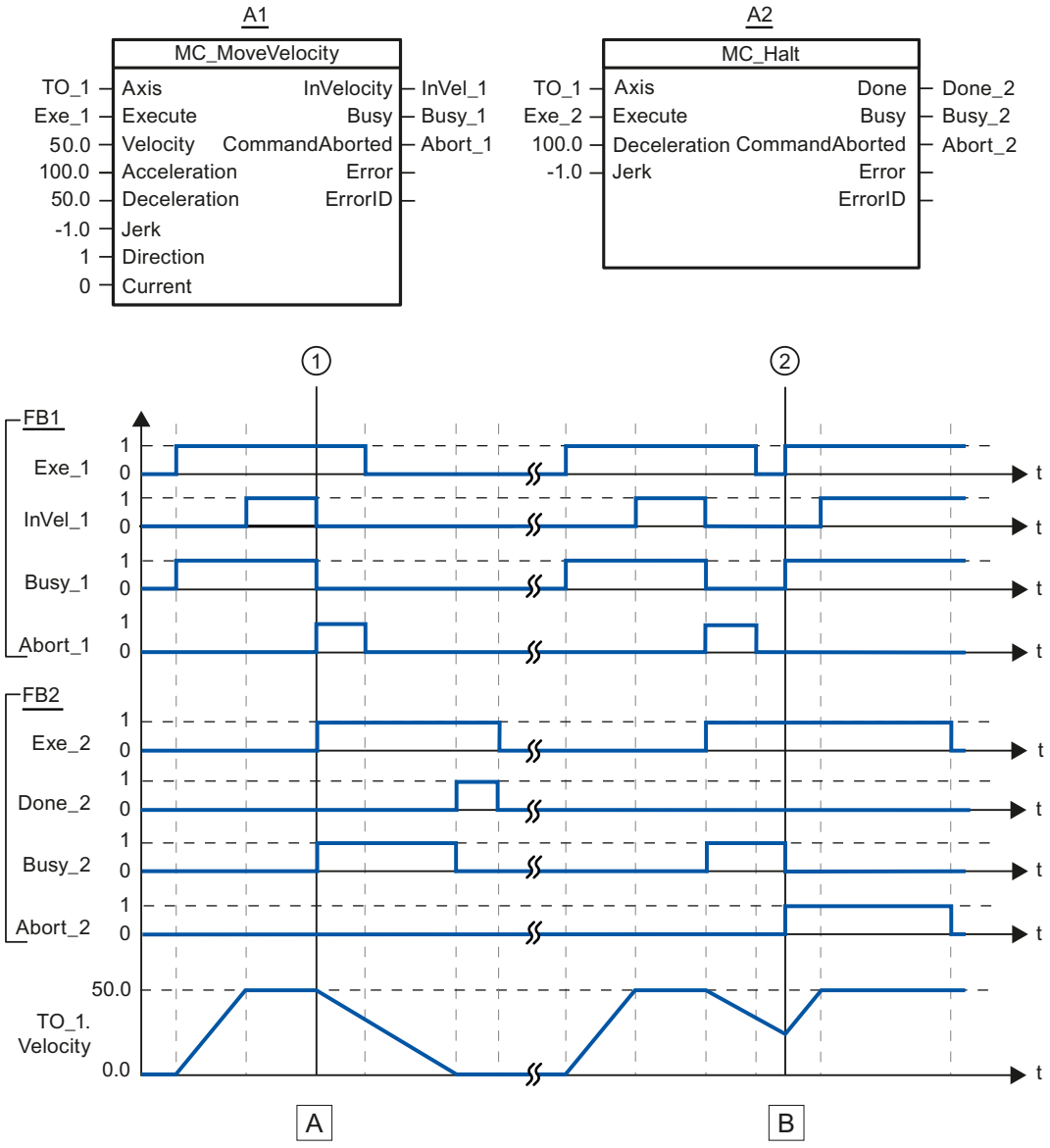

Abschnitt A	Eine Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Zum Zeitpunkt ① wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Die Achse wird bis zum Stillstand abgebremst. Der erfolgreiche Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.
Abschnitt B	Die Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Anschließend wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Während des Bremsvorgangs wird der "MC_Halt"-Auftrag zum Zeitpunkt ② durch einen "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Auftragsabbruch wird über "Abort_2" gemeldet. Das Erreichen der Solllgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Anschließend wird die Achse mit konstanter Geschwindigkeit verfahren.

MC_MoveAbsolute V2

MC_MoveAbsolute: Achse absolut positionieren V2

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveAbsolute" verfahren Sie eine Achse auf eine absolute Position.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.
- Das Technologieobjekt ist referenziert.

Ablöseverhalten

Der "MC_MoveAbsolute"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_MoveAbsolute"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag

- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Positioning-Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	REAL	0.0	Absolute Zielposition
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Direction	INPUT	INT	1	Drehrichtung der Achse Wird nur bei aktiviertem "Modulo" ausgewertet. "Technologieobjekt > Konfiguration > Grundparameter > Modulo aktivieren"	
				1	Positive Drehrichtung
				2	Negative Drehrichtung
				3	Kürzester Weg
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse auf eine absolute Position

Gehen Sie zum Verfahren einer Achse auf eine absolute Position folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Position" die gewünschte Zielposition an.
3. Starten Sie den "MC_MoveAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

MC_MoveAbsolute: Funktionsdiagramm V2

Funktionsdiagramm: Absolutes Positionieren einer Achse und ablösendes Auftragsverhalten

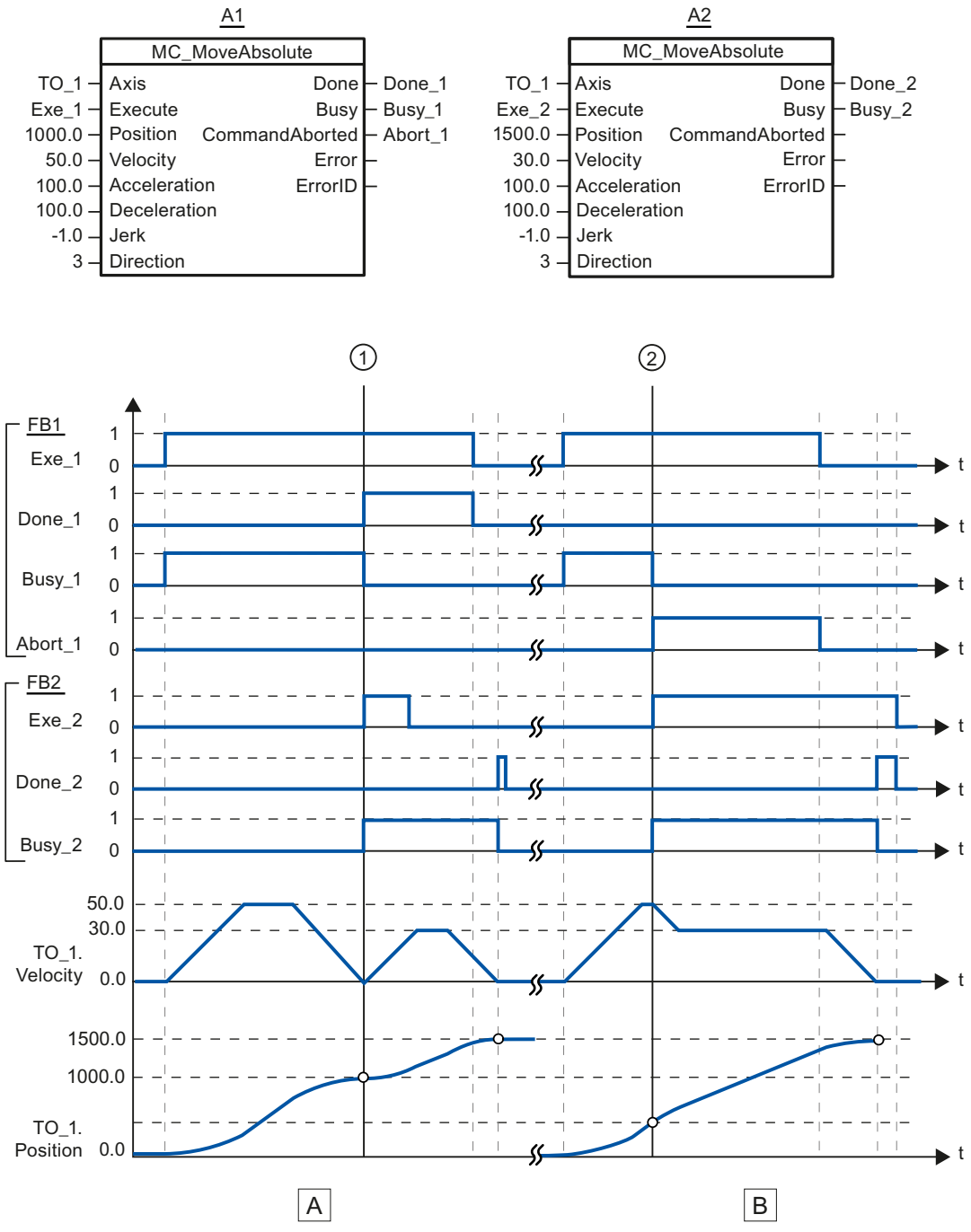

Abschnitt A	Eine Achse wird durch einen "MC_MoveAbsolute"-Auftrag (A1) auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveAbsolute"-Auftrag (A2) mit Zielposition 1500.0 gestartet. Das Erreichen der Zielposition 1500.0 wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveAbsolute"-Auftrag (A1) wird zum Zeitpunkt ② durch einen weiteren "MC_MoveAbsolute"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird auf die geänderte Geschwindigkeit abgebremst und auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveRelative V2

MC_MoveRelative: Achse relativ positionieren V2

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveRelative" verfahren Sie eine Achse relativ zu der Position, die bei Beginn der Auftragsbearbeitung vorliegt.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveRelative"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_MoveRelative"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Positioning-Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Distance	INPUT	LREAL	0.0	Wegstrecke für den Positioniervorgang (negativ oder positiv)
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse relativ zur Startposition

Gehen Sie zum Verfahren einer Achse relativ zur Startposition folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

MC_MoveRelative: Funktionsdiagramm V2

Funktionsdiagramm: Relatives Positionieren einer Achse und ablösendes Auftragsverhalten

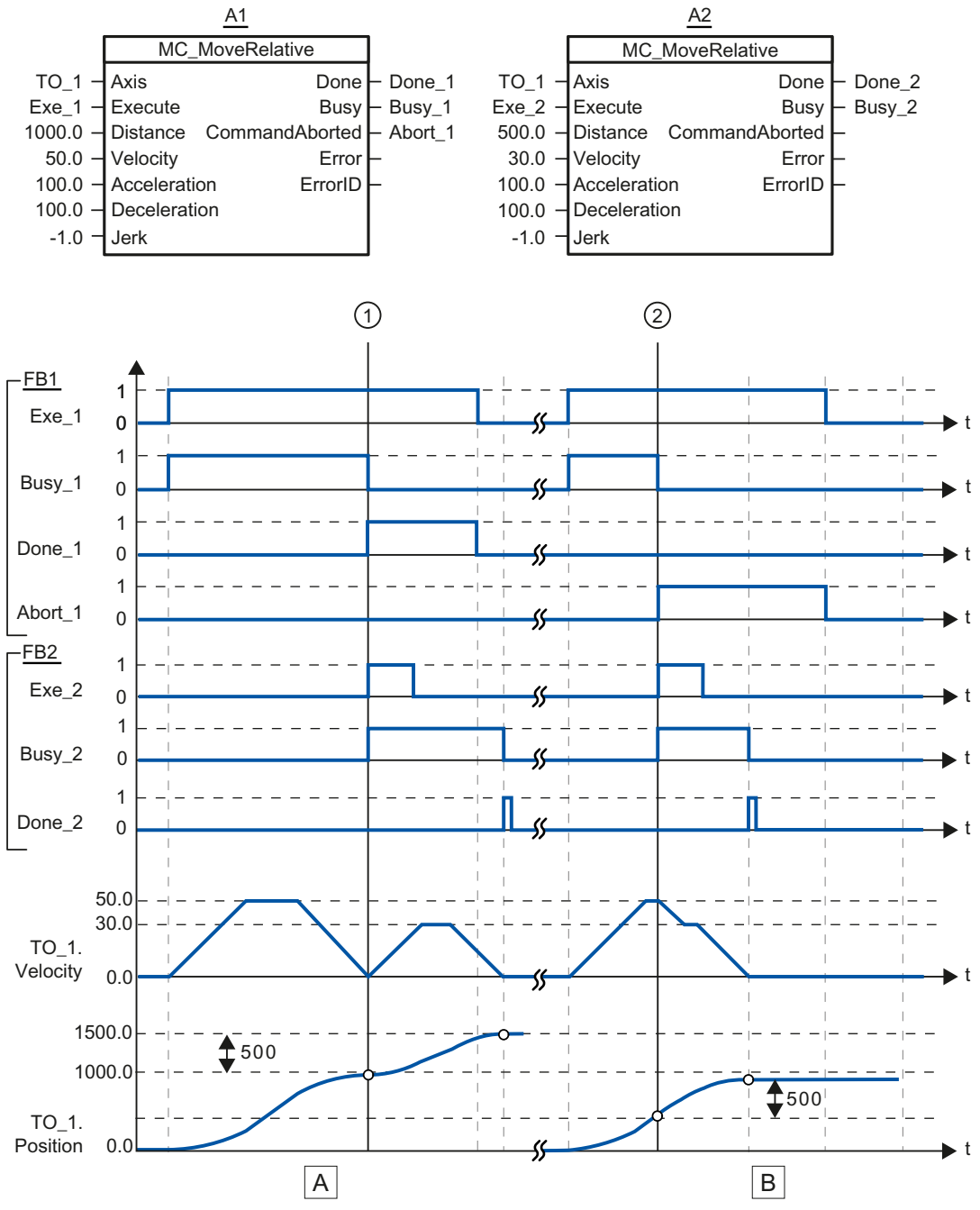

Abschnitt A	Die Achse wird durch einen "MC_MoveRelative"-Auftrag (A1) um die Wegstrecke ("Distance") 1000.0 verfahren (Startposition ist hier die Position 0.0). Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveRelative"-Auftrag (A2) mit der Wegstrecke 500.0 gestartet. Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveRelative"-Auftrag (A1) wird durch einen weiteren "MC_MoveRelative"-Auftrag (A2) abgelöst. Der Abbruch wird zum Zeitpunkt ② über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveVelocity V2

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V2

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveVelocity" bewegen sie eine Achse mit konstanter Geschwindigkeit/Drehzahl.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveVelocity" wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag

- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_MoveVelocity"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveVelocity":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang ("Velocity" = 0.0 ist erlaubt)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Direction	INPUT	INT	0	Drehrichtung der Achse	
				0	Das Vorzeichen der am Parameter "Velocity" angegebenen Geschwindigkeit definiert die Drehrichtung.
				1	Positive Drehrichtung Betrag von "Velocity" wird verwendet.
				2	Negative Drehrichtung Betrag von "Velocity" wird verwendet.
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	Deaktiviert Die Werte der Parameter "Velocity" und "Direction" werden berücksichtigt.
				TRUE	Aktiviert Die Werte an den Parametern "Velocity" und "Direction" werden **nicht** berücksichtigt. Die zum Funktionsstart aktuelle Geschwindigkeit und Richtung werden beibehalten. Sobald die Achse mit der zum Funktionsstart aktuellen Geschwindigkeit weiter verfährt, liefert der Parameter "InVelocity" den Wert TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveVelocity"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl

Gehen Sie zum Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Velocity" die Geschwindigkeit / Drehzahl an, mit der die Achse verfahren werden soll.
3. Starten Sie den "MC_MoveVelocity"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Wenn der Parameter "InVelocity" den Wert TRUE zeigt, wurde die Sollgeschwindigkeit / Solldrehzahl erreicht. Die Achse wird mit dieser Geschwindigkeit konstant weiter verfahren. Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit / Drehzahl ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_MoveVelocity: Funktionsdiagramm V2

Funktionsdiagramm: Verfahren einer Achse mit Geschwindigkeitsvorgabe und ablösendes Auftragsverhalten

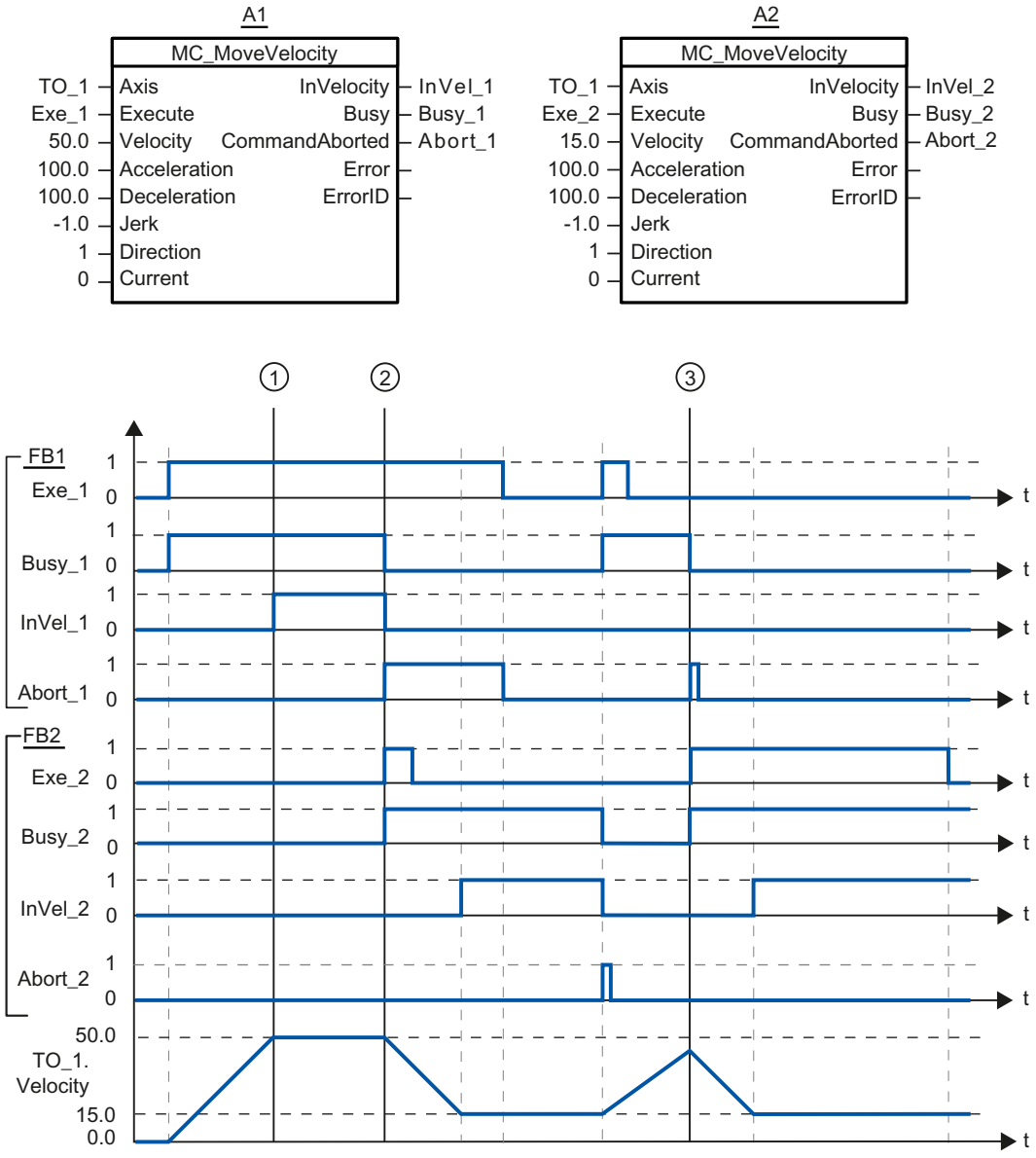

Ein über "Exe_1"angestoßener "MC_MoveVelocity"-Auftrag (A1) beschleunigt die Achse und meldet zum Zeitpunkt ① über "InVel_1" das Erreichen der Sollgeschwindigkeit 50.0.

Zum Zeitpunkt ② wird der Auftrag durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

Der laufende "MC_MoveVelocity"-Auftrag (A2) wird durch einen weiteren "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Abbruch wird über "Abort_2" gemeldet. Die Achse wird auf die neue Sollgeschwindigkeit 50.0 beschleunigt. Vor dem Erreichen der Sollgeschwindigkeit wird der aktuelle "MC_MoveVelocity"-Auftrag (A1) zum Zeitpunkt ③ durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

MC_MoveJog V2

MC_MoveJog: Achse im Tippbetrieb bewegen V2

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveJog" bewegen Sie eine Achse im Tippbetrieb.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse/Gleichlaufachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Gleichlaufachse
- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveJog"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag

- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_MoveJog"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveJog":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang Wert > 0.0: Der angegebene Wert wird verwendet. Wert < 0.0: Der Betrag des angegebenen Werts wird verwendet. ("Velocity" = 0.0 ist erlaubt)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveJog"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Verfahren einer Achse im Tippbetrieb

Gehen Sie zum Verfahren einer Achse im Tippbetrieb folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Verfahren Sie die Achse mit "JogForward" in positiver Richtung oder mit "JogBackward" in negativer Richtung.

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_MoveJog: Funktionsdiagramm V2

Funktionsdiagramm: Verfahren einer Achse im Tippbetrieb

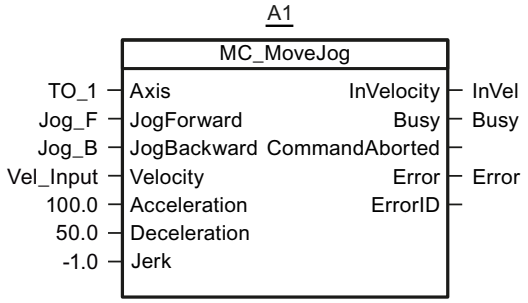

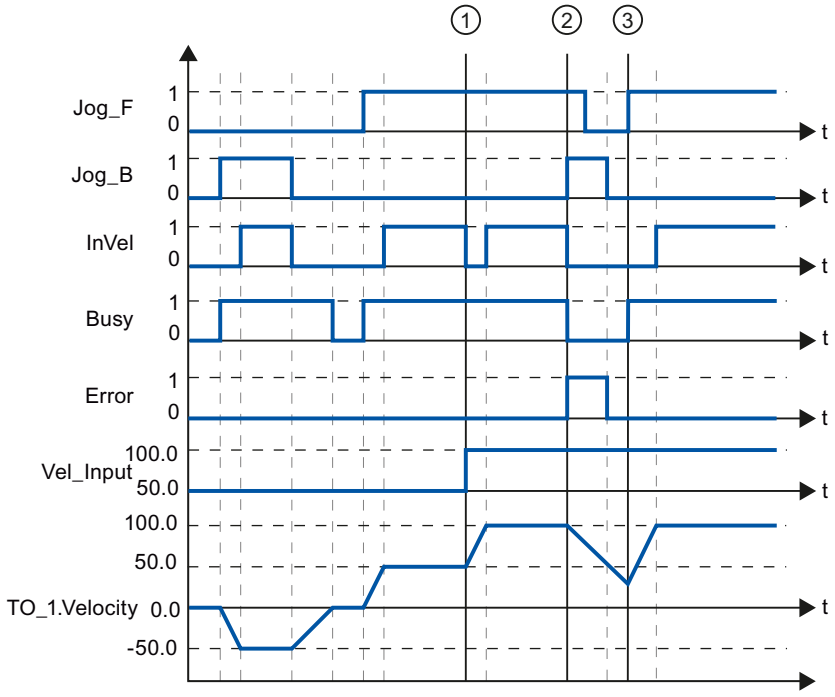

Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Sollgeschwindigkeit -50.0 wird über "InVel" = TRUE gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse abgebremst und zum Stillstand gebracht. Anschließend wird die Achse über "Jog_F" in positiver Richtung verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel" = TRUE gemeldet.

Zum Zeitpunktpunkt ① wird bei gesetztem "Jog_F" die Sollgeschwindigkeit über "Vel_Input" auf 100.0 geändert. Alternativ können Sie die Sollgeschwindigkeit auch über den Geschwindigkeits-Override ändern. "InVel" wird rückgesetzt. Die Achse wird beschleunigt. Das Erreichen der neuen Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

Bei gesetztem "Jog_F" wird zum Zeitpunkt ② ebenfalls "Jog_B" gesetzt. Wenn sowohl "Jog_F" als auch "Jog_B" gesetzt sind, wird die Achse mit der zuletzt gültigen Verzögerung abgebremst. Über "Error" wird ein Fehler angezeigt und am Ausgang "ErrorID" der Fehler 16#8007 (falsche Richtungsangabe) ausgegeben.

Dieser Fehler wird durch Rücksetzen beider Eingänge "Jog_F" und "Jog_B" behoben.

Noch während der Bremsrampe wird zum Zeitpunkt ③ "Jog_F" gesetzt. Die Achse wird auf die zuletzt konfigurierte Geschwindigkeit beschleunigt. Das Erreichen der Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

MC_MoveSuperimposed V2

MC_MoveSuperimposed: Achsen überlagernd positionieren V2

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveSuperimposed" starten Sie eine relative Positionierbewegung, die einer laufenden Basisbewegung überlagert wird.

Mit den Parametern "VelocityDiff", "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Bewegung. Die Dynamikwerte werden zu den Werten der Basisbewegung addiert. Die Dauer der Basisbewegung wird durch eine überlagernde Bewegung nicht verlängert.

Die Dynamik der Gesamtbewegung der Achse ergibt sich durch die Addition der Dynamikwerte der Basisbewegung und der überlagerten Bewegung.

Das Verhalten der Gesamtbewegung unterscheidet sich je nach Art der Basisbewegung:

- Die Basisbewegung ist eine Einzelachsbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gesamtbewegung ist auf die konfigurierten Dynamikgrenzen begrenzt.
- Die Basisbewegung ist eine Gleichlaufbewegung:
  - Die Dynamik der überlagerten Bewegung beträgt maximal die Differenz zwischen den aktuellen Dynamikwerten der Basisbewegung und den Dynamikgrenzen.
  - Die Gleichlaufbewegung der Folgeachse wird nicht auf die Dynamikgrenzen der Folgeachse begrenzt.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Leitachse im Gleichlauf wirkt sich auf die Leitachse und auf die Folgeachse aus.
  - Ein "MC_MoveSuperimposed"-Auftrag auf eine Folgeachse im Gleichlauf wirkt sich nur auf die Folgeachse aus.

Im Technologie-Datenbaustein und im TIA Portal wird immer die Dynamik der Gesamtbewegung angezeigt.

Anwendbar auf

- Gleichlaufachse
- Positionierachse

Voraussetzungen

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveSuperimposed"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Der Start eines "MC_MoveSuperimposed"-Auftrags bricht einen laufenden "MC_MoveSuperimposed"-Auftrag ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveSuperimposed":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_PositioningAxis	-	Technologieobjekt der Achse
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Distance	INPUT	LREAL	0.0	Zusätzliche Wegstrecke für den überlagernden Positioniervorgang (negativ oder positiv)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
VelocityDiff	INPUT	LREAL	-1.0	Maximale Geschwindigkeitsabweichung gegenüber der laufenden Bewegung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Überlagernde Positionierung abgeschlossen
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Überlagerte Positionierbewegung starten

Um mit der Motion Control-Anweisung "MC_MoveSuperimposed" eine überlagerte Positionierbewegung zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zusätzlich zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveSuperimposed"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

MC_MoveSuperimposed: Funktionsdiagramm V2

Funktionsdiagramm: Achsen überlagernd positionieren

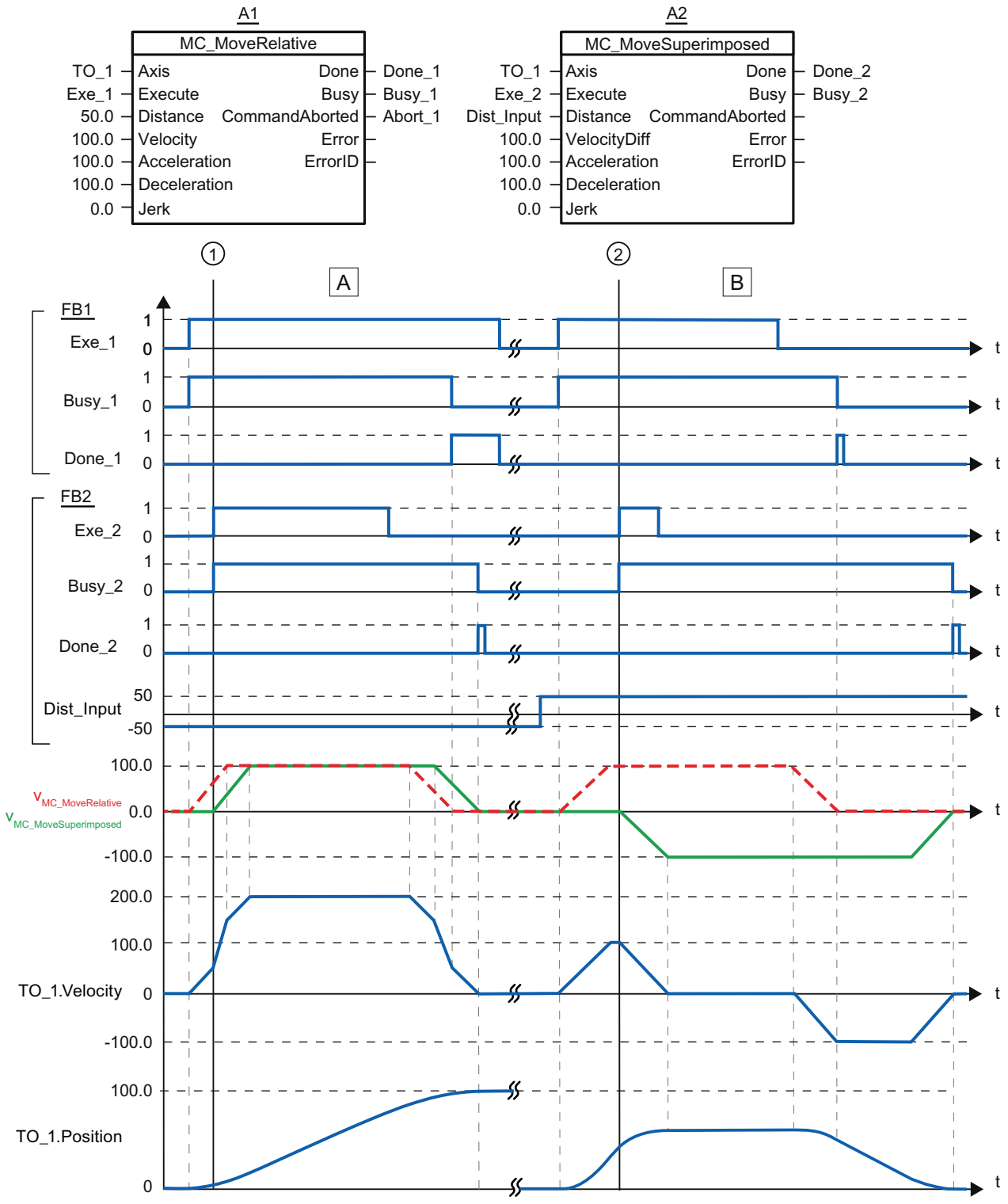

Abschnitt A	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ① wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke 50.0 angestoßen. Die Achse wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50 + 50 = 100.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.
Abschnitt B	Über "Exe_1" wird ein "MC_MoveRelative"-Auftrag mit der Wegstrecke 50.0 angestoßen. Zum Zeitpunkt ② wird über "Exe_2" ein "MC_MoveSuperimposed"-Auftrag mit der Wegstrecke -50.0 angestoßen. Die Achse reversiert und wird mit den addierten Dynamikwerten beider Aufträge um die Wegstrecke 50.0 - 50.0 = 0.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.

Synchrone Bewegung

MC_GearIn V2

MC_GearIn: Relativen Getriebegleichlauf starten V2

Beschreibung

Mit der Motion Control-Anweisung "MC_GearIn" starten Sie einen relativen Getriebegleichlauf zwischen einer Leitachse und einer Folgeachse.

Mit den Parametern "Jerk", "Acceleration", und "Deceleration" bestimmen Sie das dynamische Verhalten der Folgeachse beim Aufsynchronisieren.

Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen an den Parametern "RatioNumerator" und "RatioDenominator" vor (Zähler/Nenner). Den Getriebefaktor können Sie im laufenden Betrieb ändern, indem Sie einen neuen "MC_GearIn"-Auftrag absetzen. Übergangsvorgänge werden mit den angegebenen Dynamikwerten ausgeführt.

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- **Positiver Getriebefaktor:**
  Die Leitachse und die Folgeachse drehen in die gleiche Richtung.
- **Negativer Getriebefaktor**:
  Die Folgeachse dreht in entgegengesetzte Richtung zur Leitachse.

Sie können den Gleichlauf sowohl bei Stillstand als auch bei Bewegung der Leitachse starten.

Beim Aufsynchronisieren kann ein Positionsversatz zwischen Leitachse und Folgeachse entstehen. Dieser Positionsversatz ist vom Startzeitpunkt des "MC_GearIn" und der Dauer des Aufsynchronisierens abhängig. Der Positionsversatz wird nicht ausgeglichen.

Anwendbar auf

- Gleichlaufachse

Voraussetzungen

- Das Technologieobjekt wurde korrekt konfiguriert.
- Die Leitachse ist eine Positionierachse oder Gleichlaufachse.
- Die Folgeachse ist eine Gleichlaufachse.
- Die Leitachse ist in der Konfiguration der Folgeachse in "Technologieobjekt > Konfiguration > Leitwertverschaltungen" als mögliche Leitachse angegeben.
- Die Folgeachse ist freigegeben.

Ablöseverhalten

Der "MC_GearIn"-Auftrag wird abgebrochen durch:

- Sperren der Folgeachse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag

Der Start eines "MC_GearIn"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "Mode" = 3, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag
- "MC_GearIn"-Auftrag
- "MC_MoveSuperimposed"-Auftrag

Das Sperren der Leitachse mit "MC_Power.Enable" = FALSE bricht den Gleichlauf nicht ab. Die Folgeachse folgt der Leitachse auch während der Bremsrampe sowie nach dem erneuten Freigeben der Leitachse.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_GearIn":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Master	InOut	TO_PositioningAxis	-	Technologieobjekt der Leitachse	
Slave	InOut	TO_SynchronousAxis	-	Technologieobjekt der Folgeachse	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
RatioNumerator	INPUT	DINT	1	Getriebefaktor Zähler Zulässige ganzzahlige Werte: -2147483648 bis 2147483648 (Wert 0 nicht zulässig)	
RatioDenominator	INPUT	DINT	1	Getriebefaktor Nenner Zulässige ganzzahlige Werte: 1 bis 2147483648	
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
InGear	OUTPUT	BOOL	FALSE	TRUE	Gleichlauf erreicht Die Folgeachse ist aufsynchronisiert und fährt synchron zur Leitachse.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrags ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	0	Fehlerkennung zum Parameter "ErrorID"	

Gleichlauf starten

Um mit der Motion Control-Anweisung "MC_GearIn" einen Gleichlauf zu starten, gehen Sie folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie an den entsprechenden Parametern die Leitachse, die Folgeachse und den Getriebefaktor an.
3. Starten Sie den "MC_GearIn"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

Die Folgeachse wird auf den Leitwert der Leitachse aufsynchronisiert. Wenn der Parameter "InGear" den Wert TRUE zeigt, ist die Folgeachse aufsynchronisiert und fährt synchron zur Leitachse. Die Parameter "InGear" und "Busy" zeigen so lange den Wert TRUE, bis der "MC_GearIn"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

MC_GearIn: Funktionsdiagramm V2

Funktionsdiagramm: Aufsynchronisieren und Umschalten des Leitwerts

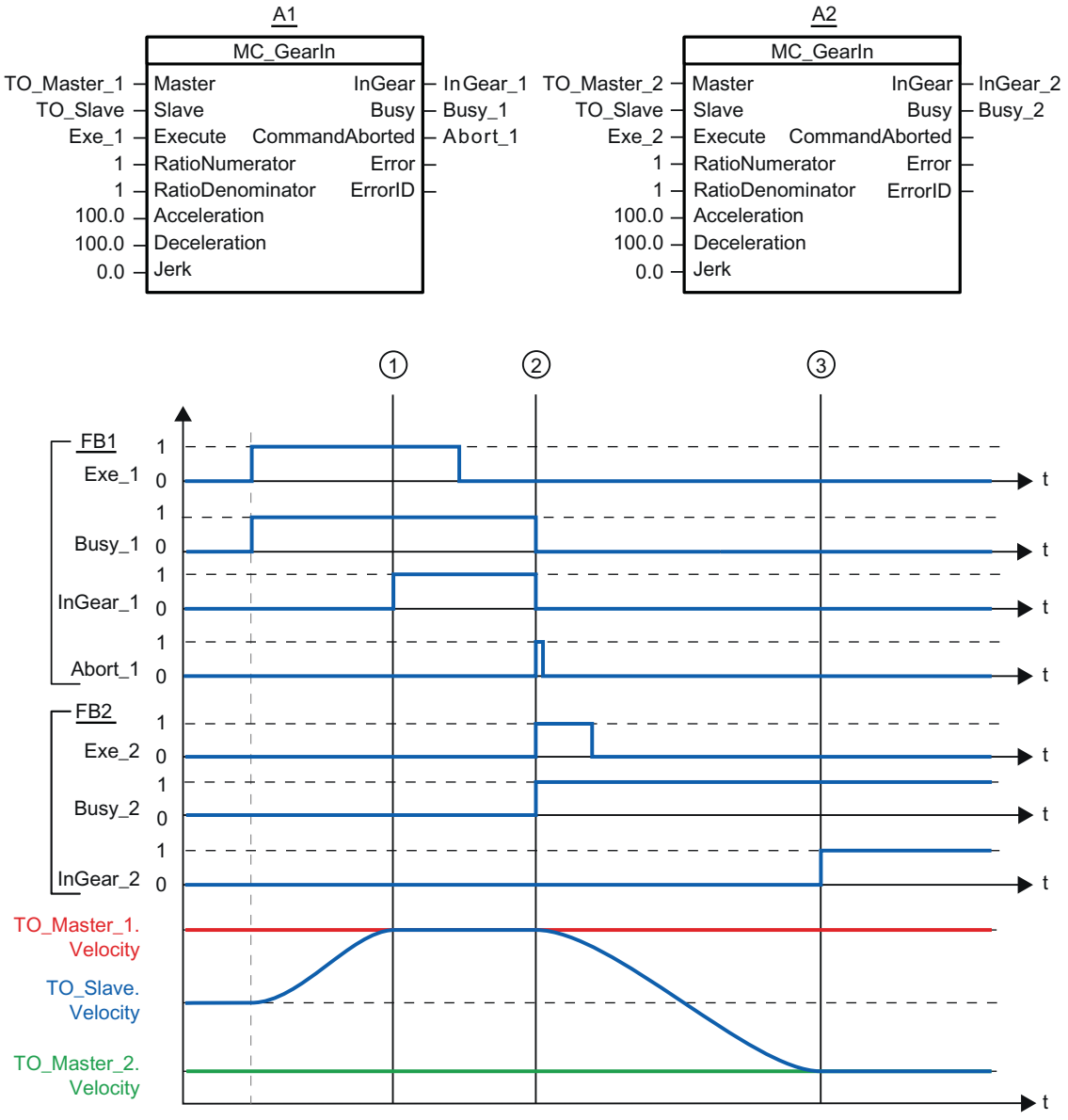

Über "Exe_1" wird ein "MC_GearIn"-Auftrag (A1) angestoßen. Die Folgeachse (TO_Slave) wird auf die Leitachse (TO_Master_1) aufsynchronisiert. Zum Zeitpunkt ① wird über "InGear_1" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

Zum Zeitpunkt ② wird der Gleichlauf durch einen weiteren "MC_GearIn"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Folgeachse wird auf die neue Leitachse (TO_Master_2) aufsynchronisiert. Zum Zeitpunkt ③ wird über "InGear_2" gemeldet, dass die Folgeachse synchronisiert ist und synchron zur Leitachse fährt.

S7-1500 Motion Control V1

MC_Power V1

MC_Power: Technologieobjekt freigeben, sperren V1

Beschreibung

Mit der Motion Control-Anweisung "MC_Power" wird ein Technologieobjekt freigegeben bzw. gesperrt.

Anwendbar auf

- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.

Ablöseverhalten

- Ein "MC_Power"-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.
- Ein "MC_Power"-Auftrag mit dem Parameter "Enable" = TRUE gibt ein Technologieobjekt frei und bricht somit keine anderen Motion Control-Anweisungen ab.
- Mit dem Sperren des Technologieobjekts (Parameter "Enable" = FALSE) werden alle Bewegungsaufträge am zugehörigen Technologieobjekt entsprechend dem gewählten "StopMode" abgebrochen. Dieser Vorgang kann vom Anwender nicht abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Power":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	InOut	TO_Axis	-	Technologieobjekt	
Enable	INPUT	BOOL	FALSE	TRUE	Das Technologieobjekt wird freigegeben.
				FALSE	Das Technologieobjekt wird gesperrt. Alle laufenden Aufträge am Technologieobjekt werden entsprechend dem parametrierten "StopMode" abgebrochen.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
StopMode	INPUT	INT	0	Für Technologieobjekt Externer Geber nicht relevant. Wenn Sie ein Technologieobjekt mit einer fallenden Flanke am Parameter "Enable" sperren, bremst die Achse gemäß dem gewählten "StopMode" ab.	
				0	Notstopp Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Notstopp-Rampe" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicDefaults. EmergencyDeceleration)
				1	Sofort-Aus Beim Sperren eines Technologieobjekts wird der Sollwert Null ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
				2	Stopp mit maximalen Dynamikwerten Beim Sperren des Technologieobjekts wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt. Anschließend wird die Freigabe weggenommen. (<TO>.DynamicLimits.MaxDeceleration; <TO>.DynamicLimits.MaxJerk)
Status	OUTPUT	BOOL	FALSE	Freigabestatus des Technologieobjekts	
				FALSE	Gesperrt - Eine Positionier- oder Drehzahlachse nimmt keine Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind nicht aktiv. - Die Istwerte des Technologieobjekts werden nicht auf Gültigkeit überprüft.
				TRUE	Freigegeben - Eine freigegebene Positionier- oder Drehzahlachse nimmt Bewegungsaufträge an. - Drehzahlsteuerung und Positionsregelung sind aktiv. - Die Istwerte des Technologieobjekts sind gültig.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
Error	OUTPUT	BOOL	FALSE	TRUE	An der Motion Control-Anweisung "MC_Power" ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Freigeben von Technologieobjekten

Setzen Sie zum Freigeben eines Technologieobjekts den Parameter "Enable" auf TRUE.

Wenn der Parameter "Status" den Wert TRUE zeigt, ist das Technologieobjekt freigegeben.

Wenn eine Achse beim Freigeben des Technologieobjekts in Bewegung ist (Istgeschwindigkeit vorhanden), wird die Achse mit der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Verzögerung auf Sollwert Null abgebremst (<TO>.DynamicLimits.MaxDeceleration). Diese Bremsrampe ist von Bewegungsaufträgen ablösbar.

Hinweis

Automatische Freigabe nach Quittierung eines Technologiealarms

Wenn das Technologieobjekt aufgrund eines Technologiealarms gesperrt wird, so wird es nach dem Beseitigen der Ursache und Quittieren des Alarms automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Parameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

Sperren von Technologieobjekten

Setzen Sie zum Sperren eines Technologieobjekts den Parameter "Enable" auf FALSE.

Wenn eine Achse in Bewegung ist, wird Sie gemäß dem gewählten "StopMode" abgebremst und zum Stillstand gebracht.

Wenn die Parameter "Busy" und "Status" den Wert FALSE zeigen, ist das Sperren des Technologieobjekts abgeschlossen.

Antriebsanbindung über PROFIdrive

Bei der Ankopplung eines Antriebs über PROFIdrive werden Sollwert, Freigabe und Antriebsstatus über das PROFIdrive-Telegramm übertragen.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird das Technologieobjekt freigegeben. Der Antrieb wird gemäß PROFIdrive-Norm aktiviert.
  Wenn die Variable <TO>.StatusDrive.InOperation den Wert TRUE zeigt, ist der Antrieb bereit, Sollwerte auszuführen. Der Parameter "Status" wird auf den Wert TRUE gesetzt.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Der Antrieb wird gemäß PROFIdrive-Norm deaktiviert.

Analoge Antriebsanbindung

Der Sollwert wird über einen Analogausgang ausgegeben. Optional kann ein Freigabesignal über Digitalausgang (<TO>.Actor.Interface.EnableDriveOutput) und ein Bereitschaftssignal über Digitaleingang (<TO>.Actor.Interface.DriveReadyInput) konfiguriert werden.

- **Technologieobjekt freigeben und Antrieb aktivieren**
  Mit dem Parameter "Enable" = TRUE wird der Freigabe-Ausgang ("Enable drive output") gesetzt.
  Wenn der Antrieb das Bereitschaftssignal über den Bereit-Eingang ("Drive ready input") zurückmeldet, wird der Parameter "Status" und die Variable des Technologieobjekts <TO>.StatusDrive.InOperation auf TRUE gesetzt und der Sollwert auf den Analogausgang geschaltet.
- **Technologieobjekt sperren und Antrieb deaktivieren**
  Mit dem Parameter "Enable" = FALSE wird der Parameter "Status" auf den Wert FALSE gesetzt und die Achse gemäß dem gewählten "StopMode" abgebremst. Mit dem Erreichen von Sollwert Null wird der Freigabe-Ausgang auf FALSE gesetzt.

Weitere Informationen

Weiterführende Informationen zum Freigeben und Sperren von Technologieobjekten und Antrieben finden Sie im Anhang Funktionsdiagramme MC_Power.

MC_Power: Funktionsdiagramm V1

Funktionsdiagramm: Freigeben eines Technologieobjekts und Beispiel für Alarmreaktion

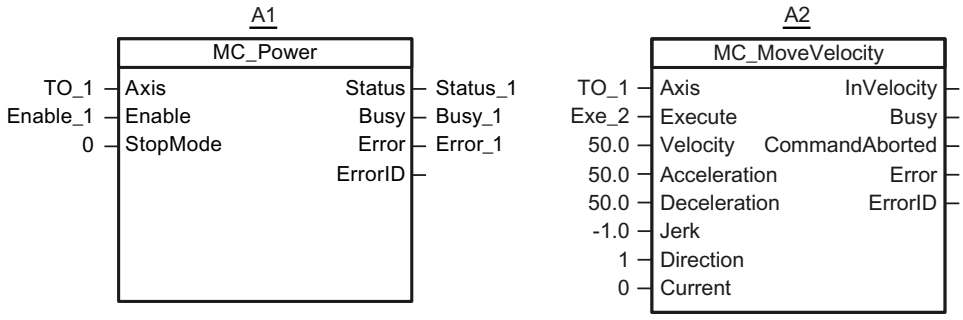

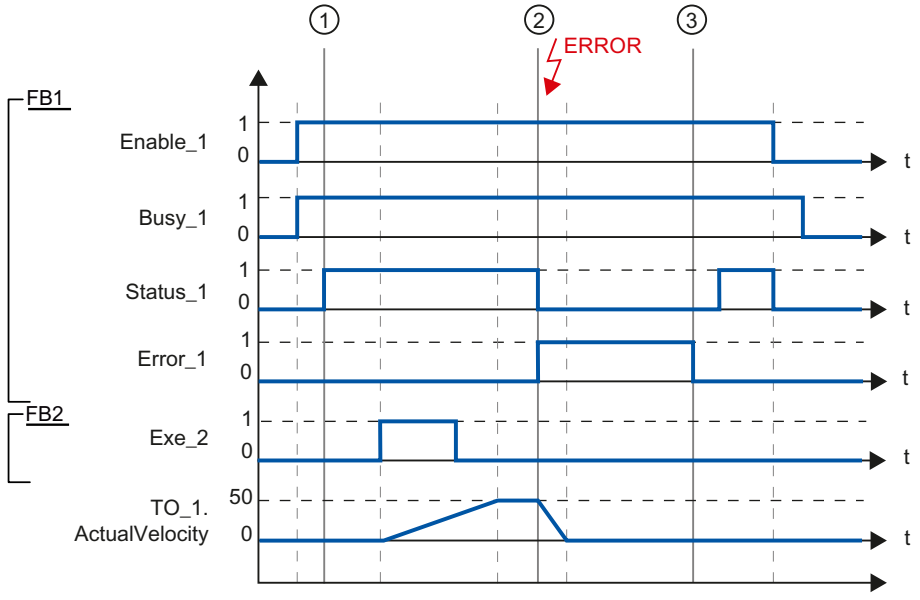

Ein Technologieobjekt wird mit "Enable_1" = TRUE freigegeben. Die erfolgreiche Freigabe kann an "Status_1" zum Zeitpunkt ① abgelesen werden. Anschließend wird die Achse mit einem "MC_MoveVelocity"-Auftrag (A2) verfahren. Der Geschwindigkeitsverlauf der Achse kann an "Velocity Axis_1" abgelesen werden.

Zum Zeitpunkt ② tritt ein Fehler am Technologieobjekt auf, der das Sperren des Technologieobjekts zur Folge hat (Alarmreaktion: Freigabe wegnehmen). Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht. Mit dem Sperren des Technologieobjekts wird "Status_1" rückgesetzt. Da die Achse nicht über "Enable_1" = FALSE gesperrt wurde, ist der gewählte "StopMode" nicht relevant. Die Fehlerursache wird beseitigt und der Alarm wird zum Zeitpunkt ③ quittiert.

Da "Enable_1" weiterhin gesetzt ist, wird das Technologieobjekt wieder freigegeben. Die erfolgreiche Freigabe kann an "Status_1" abgelesen werden. Abschließend wird das Technologieobjekt über "Enable_1" = FALSE gesperrt.

MC_Reset V1

MC_Reset: Alarme quittieren, Restart Technologieobjekt V1

Beschreibung

Mit der Motion Control-Anweisung "MC_Reset" quittieren Sie alle Technologie-Alarme, die im Anwenderprogramm quittierbar sind. Mit der Quittierung werden auch die "Error"- und "Warning"-Bits im Technologie-Datenbaustein zurückgesetzt.

Mit der Motion Control-Anweisung "MC_Reset" mit "Restart" = TRUE starten Sie das Neuinitialisieren (Restart) von Technologieobjekten. Mit dem Restart des Technologieobjekts werden neue Konfigurationsdaten im Technologie-Datenbaustein übernommen.

Anwendbar auf

- Positionierachse
- Drehzahlachse
- Externer Geber

Voraussetzung

- Bei einem anstehenden Technologie-Alarm wurde die Fehlerursache beseitigt.
- Für einen Restart muss das Technologieobjekt gesperrt sein.
  ("MC_Power.Status" = FALSE und "MC_Power.Busy" = FALSE)

Ablöseverhalten

- Parameter "Restart" = FALSE:
  Die Bearbeitung der Anweisung "MC_Reset" kann durch andere Motion Control-Aufträge abgebrochen werden. Der "MC_Reset"-Auftrag bricht keine laufenden Motion Control-Aufträge ab.
- Parameter "Restart" = TRUE:
  Die Bearbeitung der Anweisung "MC_Reset" mit Parameter "Restart" = TRUE kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Reset":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Restart	INPUT	BOOL	FALSE	TRUE	"Restart" Neuinitialisierung des Technologieobjektes und Quittierung anstehender Technologie-Alarme. Das Technologieobjekt wird mit den konfigurierten Startwerten neu initialisiert.
				FALSE	Quittierung anstehender Technologie-Alarme.
Done	OUTPUT	BOOL	FALSE	TRUE	Fehler wurde quittiert. Restart wurde ausgeführt.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Technologie-Alarme quittieren

Gehen Sie zum Quittieren von Technologie-Alarmen folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Setzen sie den Parameter "Restart" = FALSE.
3. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so wurde der Fehler quittiert.

Hinweis

Quittieren mit "Restart" = FALSE

Setzen Sie "Restart" = FALSE, wenn nur die Technologie-Alarmen quittiert werden sollen. Während eines Restarts kann das Technologieobjekt nicht verwendet werden.

Restart eines Technologieobjektes

Gehen Sie zum Restart eines Technologieobjektes wie folgt vor:

1. Prüfen Sie die oben genannten Vorraussetzungen.
2. Setzen sie den Parameter "Restart" = TRUE.
3. Führen Sie den Restart durch eine steigende Flanke am Parameter "Execute" aus.

Wenn der Parameter "Done" den Wert TRUE zeigt, ist der Restart des Technologieobjektes abgeschlossen.

Weitere Informationen zum Restart finden Sie im Kapitel Restart von Technologieobjekten.

MC_Home V1

MC_Home: Technologieobjekt referenzieren, Referenzpunkt setzen V1

Beschreibung

Mit der Motion Control-Anweisung "MC_Home" stellen Sie den Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung her. Der Positionsistwert am Technologieobjekt wird dabei einer Referenzmarke zugeordnet. Diese Referenzmarke repräsentiert eine bekannte mechanische Position.

Der Referenziervorgang erfolgt gemäß der am Parameter "HomingMode" gewählten Betriebsart und der Konfiguration unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren".

Für die Dynamikwerte Beschleunigung, Verzögerung und Ruck werden die voreingestellten Werte unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik-Voreinstellung" verwendet.

Anwendbar auf

- Positionierachse
- Externer Geber

Die folgende Tabelle zeigt, welche Betriebsarten mit den jeweiligen Technologieobjekten möglich sind:

Betriebsart	Positionierachse mit Inkrementalgeber	Positionierachse mit Absolutwertgeber	Externer Geber inkrementell	Externer Geber absolut
Aktives Referenzieren ("HomingMode" = 4, 5)	X	-	-	-
Passives Referenzieren ("HomingMode" = 2, 3, 8)	X	-	X	-
Setzen eines Positionswertes ("HomingMode" = 0)	X	X	X	X
Relative Verschiebung des Istwertes ("HomingMode" = 1)	X	X	X	X
Absolutwertgeber referenzieren ("HomingMode" = 6, 7)	-	X	-	X

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- "HomingMode" = 2, 3, 4, 5, 8
  Das Technologieobjekt muss freigegeben sein.
- "HomingMode" = 0, 1, 2, 6, 7
  Die Geberwerte müssen gültig sein. (<TO>.StatusSensor[n].State = 2)

Ablöseverhalten

- Ein "MC_Home"-Auftrag zum passiven Referenzieren wird abgebrochen durch:
  - Sperren des Technologieobjekts mit "MC_Power.Enable" = FALSE
  - "MC_Home"-Auftrag mit Parameter "HomingMode" = 4, 5, 9
- Ein "MC_Home"-Auftrag zum passiven Referenzieren bricht keine anderen Motion Control-Aufträge ab.
- Ein "MC_Home"-Auftrag zum aktiven Referenzieren wird abgebrochen durch:
  - Sperren des Technologieobjekts "MC_Power.Enable" = FALSE
  - "MC_Home"-Auftrag "HomingMode" = 4, 5
  - "MC_Halt"-Auftrag
  - "MC_MoveAbsolute"-Auftrag
  - "MC_MoveRelative"-Auftrag
  - "MC_MoveVelocity"-Auftrag
  - "MC_MoveJog"-Auftrag
- Ein "MC_Home"-Auftrag zum aktiven Referenzieren bricht folgende laufende Motion Control-Aufträge ab:
  - "MC_Home"-Auftrag mit Parameter "HomingMode" = 2, 3, 4, 5
  - "MC_Halt"-Auftrag
  - "MC_MoveAbsolute"-Auftrag
  - "MC_MoveRelative"-Auftrag
  - "MC_MoveVelocity"-Auftrag
  - "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Home":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	LREAL	0.0	Der angegebene Wert wird dem gewähltem "Homing-Mode" entsprechend verwendet.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
HomingMode	INPUT	INT	4	Betriebsart	
				0	Direktes Referenzieren (Absolut) Die aktuelle Position des Technologieobjektes wird auf den Wert des Parameters "Position" gesetzt.
				1	Direktes Referenzieren (Relativ) Die aktuelle Position des Technologieobjektes wird um den Wert des Parameters "Position" verschoben.
				2	Passives Referenzieren Beim Erkennen der Referenzmarke wird der Istwert auf den Wert des Parameters "Position" gesetzt.
				3	Passives Referenzieren (Parameter "Position" unwirksam) Beim Erkennen der Referenzmarke wird der Istwert auf den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Passives Referenzieren" konfigurierten Referenzpunkt gesetzt. (<TO>.Homing.HomePosition)
				4	Aktives Referenzieren Das TO Positionierachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem Wert des Parameters "Position".
				5	Aktives Referenzieren (Parameter "Position" unwirksam) Das TO Positionierachse führt eine Referenzierbewegung gemäß der Konfiguration aus. Nach Abschluss der Bewegung steht die Achse auf dem unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Referenzieren > Aktives Referenzieren" konfigurierten Referenzpunkt. (<TO>.Homing.HomePosition)
				6	Absolutwertgeberjustage (Relativ) Die aktuelle Position wird um den Wert des Parameters "Position" verschoben. Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n].AbsEncoderOffset)
				7	Absolutwertgeberjustage (Absolut) Die aktuelle Position wird auf den Wert des Parameters "Position" gesetzt.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
					Der berechnete Absolutwert-Offset wird remanent in der CPU gespeichert. (<TO>.StatusSensor[n].AbsEncoderOffset)
				8	Passives Referenzieren (ohne Rücksetzen) Funktion wie "HomingMode" 2 mit dem Unterschied, dass der Status "referenziert" mit dem Aktivieren der Funktion **nicht** zurückgesetzt wird.
				9	Abbruch Passives Referenzieren Ein laufender Auftrag zum passiven Referenzieren wird abgebrochen.
Done	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist abgeschlossen.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjektes wird unter folgenden Bedingungen rückgesetzt (<TO>.StatusWord.X5 (HomingDone)):

- **Technologieobjekte mit inkrementellen Istwerten:**
  - Starten eines "MC_Home"-Auftrags mit "HomingMode" = 2, 3, 4, 5
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS -> NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- **Technologieobjekte mit absoluten Istwerten:**
  - Wiederherstellen der CPU-Werkseinstellung
  - Veränderung der Geberkonfiguration
  - Austausch der CPU

Referenzieren eines Technologieobjekts mit "HomingMode" = 1..8

Gehen Sie zum Referenzieren eines Technologieobjektes folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "HomingMode" die gewünschten Referenzierfunktion an.
3. Versorgen Sie die notwendigen Parameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Parameter "Execute".

Zeigt der Parameter "Done" den Wert TRUE, so ist der "MC_Home"-Auftrag gemäß dem gewählten "HomingMode" abgeschlossen. Der Status "Referenziert" des Technologieobjekts wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Referenziert" angezeigt (<TO>.StatusWord.X5 (HomingDone)).

Abbruch eines passiven Refenziervorgangs mit "HomingMode" = 9

Das Technologieobjekt wird mit "HomingMode" = 9 nicht referenziert. Wenn ein laufender "MC_Home"-Auftrag zum passiven Referenzieren ("HomingMode" = 2, 3, 8) durch einen anderen "MC_Home"-Auftrag mit "HomingMode" = 9 abgelöst wird, so wird der laufende Auftrag mit dem Parameter "CommandAborted" = TRUE abgebrochen. Der ablösende Auftrag mit "HomingMode" = 9 meldet die erfolgreiche Durchführung mit Parameter "Done" = TRUE.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_Halt V1

MC_Halt: Achse anhalten V1

Beschreibung

Mit der Motion Control-Anweisung "MC_Halt" bremsen Sie eine Achse bis zum Stillstand ab.

Mit den Parametern "Jerk" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bremsvorgang.

Anwendbar auf

- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_Halt"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Der Start eines "MC_Halt"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_Halt":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt	
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke	
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Geschwindigkeit Null erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Abbremsen einer Achse mit "MC_Halt"

Gehen Sie zum Abbremsen einer Achse bis zum Stillstand folgendermaßen vor:

1. Prüfen Sie die oben genannten Vorraussetzungen.
2. Versorgen Sie die Parameter "Deceleration" und "Jerk" mit den gewünschten Werten.
3. Starten Sie den "MC_Halt"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt. Der Stillstand der Achse wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung > Stillstand" angezeigt (<TO>.StatusWord.X7 (Standstill)).

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_Halt: Funktionsdiagramm V1

Funktionsdiagramm: Anhalten einer Achse und ablösendes Auftragsverhalten

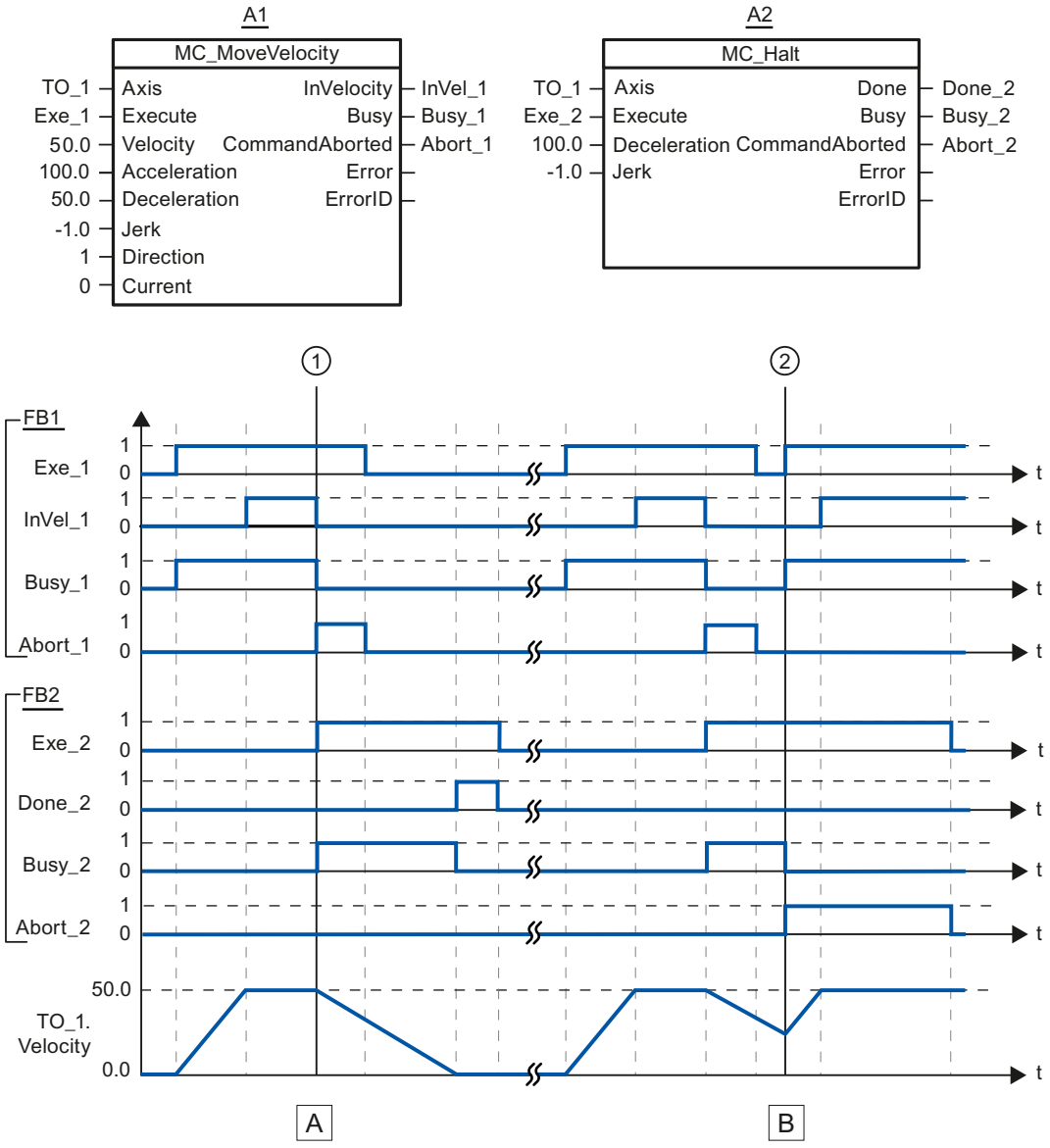

Abschnitt A	Eine Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Zum Zeitpunkt ① wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Die Achse wird bis zum Stillstand abgebremst. Der erfolgreiche Abschluss des "MC_Halt"-Auftrags wird über "Done_2" gemeldet.
Abschnitt B	Die Achse wird über einen "MC_MoveVelocity"-Auftrag (A1) verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Anschließend wird der "MC_MoveVelocity"-Auftrag durch einen "MC_Halt"-Auftrag (A2) abgelöst. Der Auftragsabbruch wird über "Abort_1" gemeldet. Während des Bremsvorgangs wird der "MC_Halt"-Auftrag zum Zeitpunkt ② durch einen "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Auftragsabbruch wird über "Abort_2" gemeldet. Das Erreichen der Solllgeschwindigkeit 50.0 wird über "InVel_1" gemeldet. Anschließend wird die Achse mit konstanter Geschwindigkeit verfahren.

MC_MoveAbsolute V1

MC_MoveAbsolute: Achse absolut positionieren V1

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveAbsolute" verfahren Sie eine Achse auf eine absolute Position.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.
- Das Technologieobjekt ist referenziert.

Ablöseverhalten

Der "MC_MoveAbsolute"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Der Start eines "MC_MoveAbsolute"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveAbsolute":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Positioning-Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Position	INPUT	REAL	0.0	Absolute Zielposition
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)
Direction	INPUT	INT	1	Drehrichtung der Achse Wird nur bei aktiviertem "Modulo" ausgewertet. "Technologieobjekt > Konfiguration > Grundparameter > Modulo aktivieren" 1 Positive Drehrichtung 2 Negative Drehrichtung 3 Kürzester Weg
Done	OUTPUT	BOOL	FALSE	TRUE Zielposition erreicht.
Busy	OUTPUT	BOOL	FALSE	TRUE Auftrag ist in Bearbeitung.

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse auf eine absolute Position

Gehen Sie zum Verfahren einer Achse auf eine absolute Position folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Position" die gewünschte Zielposition an.
3. Starten Sie den "MC_MoveAbsolute"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

MC_MoveAbsolute: Funktionsdiagramm V1

Funktionsdiagramm: Absolutes Positionieren einer Achse und ablösendes Auftragsverhalten

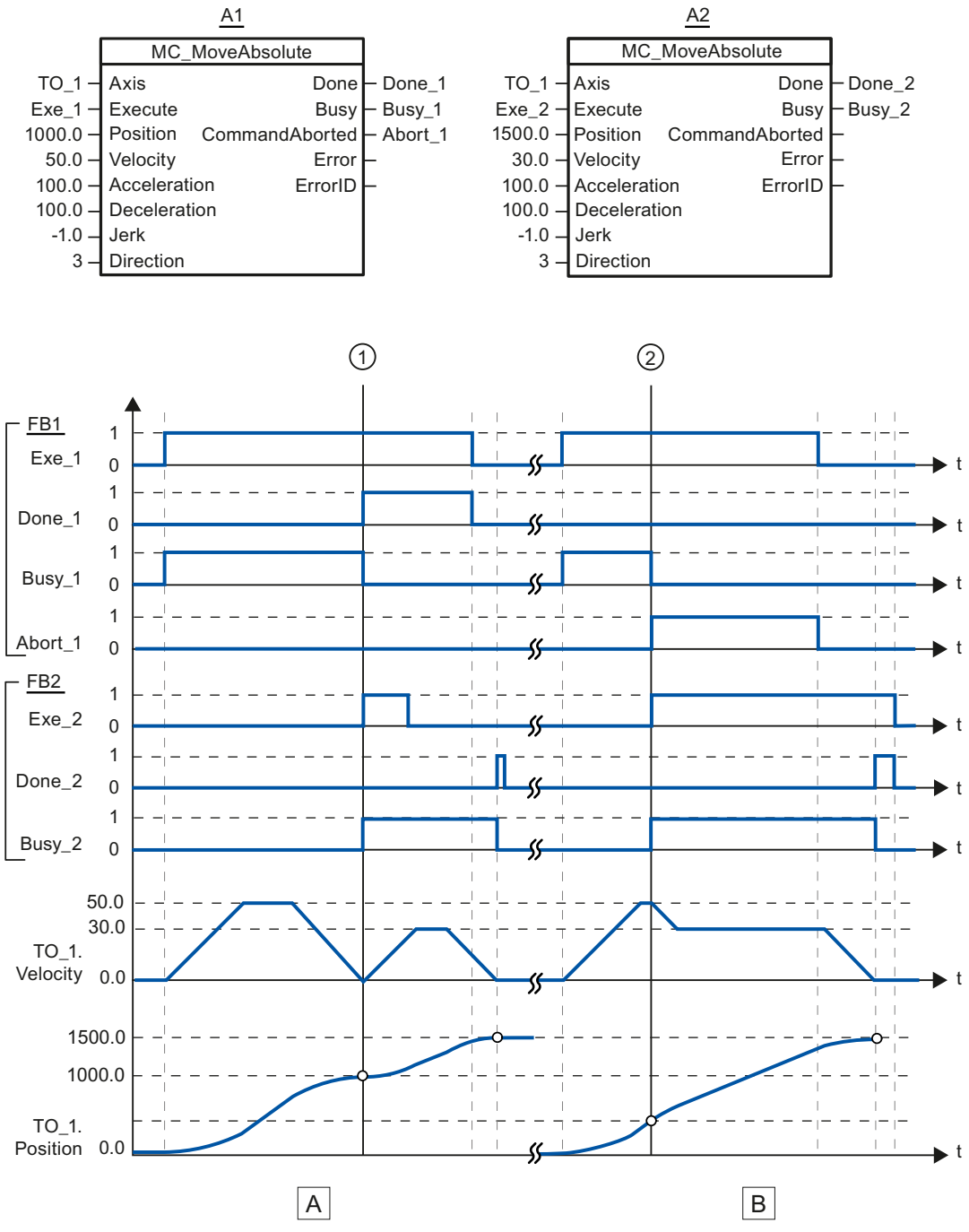

Abschnitt A	Eine Achse wird durch einen "MC_MoveAbsolute"-Auftrag (A1) auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveAbsolute"-Auftrag (A2) mit Zielposition 1500.0 gestartet. Das Erreichen der Zielposition 1500.0 wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveAbsolute"-Auftrag (A1) wird zum Zeitpunkt ② durch einen weiteren "MC_MoveAbsolute"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird auf die geänderte Geschwindigkeit abgebremst und auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveRelative V1

MC_MoveRelative: Achse relativ positionieren V1

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveRelative" verfahren Sie eine Achse relativ zu der Position, die bei Beginn der Auftragsbearbeitung vorliegt.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

Anwendbar auf

- Positionierachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveRelative"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Der Start eines "MC_MoveRelative"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveRelative":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_Positioning-Axis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Distance	INPUT	LREAL	0.0	Wegstrecke für den Positioniervorgang (negativ oder positiv)
Velocity	INPUT	LREAL	-1.0	Sollgeschwindigkeit für die Positionierung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Geschwindigkeit wird verwendet. (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Zielposition erreicht
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verfahren einer Achse relativ zur Startposition

Gehen Sie zum Verfahren einer Achse relativ zur Startposition folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Distance" die zu verfahrende Wegstrecke an.
3. Starten Sie den "MC_MoveRelative"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "Done" und "Error" wird der aktuelle Bewegungszustand angezeigt.

MC_MoveRelative: Funktionsdiagramm V1

Funktionsdiagramm: Relatives Positionieren einer Achse und ablösendes Auftragsverhalten

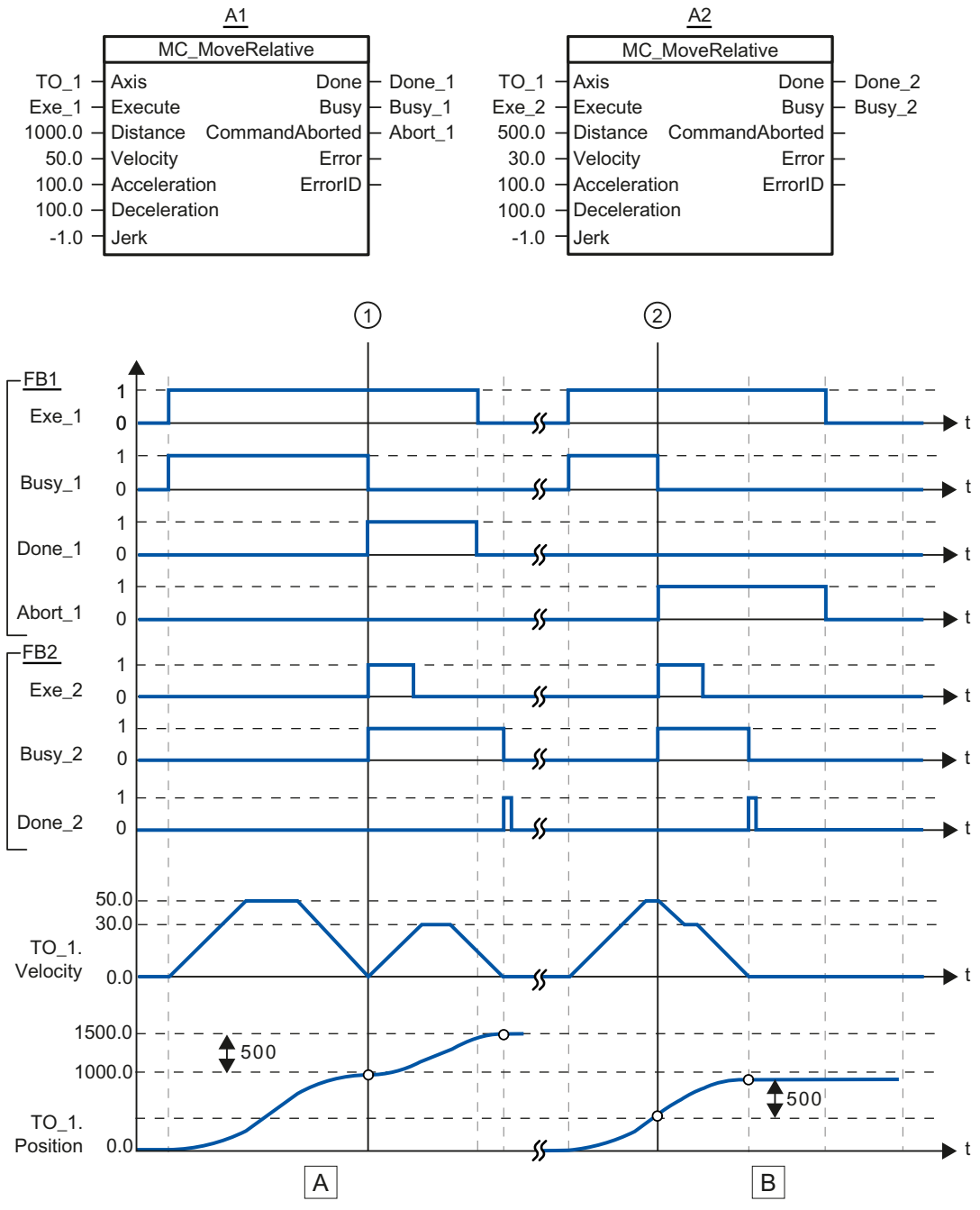

Abschnitt A	Die Achse wird durch einen "MC_MoveRelative"-Auftrag (A1) um die Wegstrecke ("Distance") 1000.0 verfahren (Startposition ist hier die Position 0.0). Das Erreichen der Zielposition wird zum Zeitpunkt ① über "Done_1" gemeldet. Zu diesem Zeitpunkt ① wird ein weiterer "MC_MoveRelative"-Auftrag (A2) mit der Wegstrecke 500.0 gestartet. Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet. Da "Exe_2" vorher zurückgesetzt wurde, steht "Done_2" nur für einen Zyklus an.
Abschnitt B	Ein laufender "MC_MoveRelative"-Auftrag (A1) wird durch einen weiteren "MC_MoveRelative"-Auftrag (A2) abgelöst. Der Abbruch wird zum Zeitpunkt ② über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.

MC_MoveVelocity V1

MC_MoveVelocity: Achse mit Geschwindigkeits-/Drehzahlvorgabe bewegen V1

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveVelocity" bewegen sie eine Achse mit konstanter Geschwindigkeit/Drehzahl.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveVelocity" wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag

- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Der Start eines "MC_MoveVelocity"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveVelocity":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt
Execute	INPUT	BOOL	FALSE	Start des Auftrags mit steigender Flanke
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang ("Velocity" = 0.0 ist erlaubt)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Direction	INPUT	INT	0	Drehrichtung der Achse	
				0	Das Vorzeichen der am Parameter "Velocity" angegebenen Geschwindigkeit definiert die Drehrichtung.
				1	Positive Drehrichtung Betrag von "Velocity" wird verwendet.
				2	Negative Drehrichtung Betrag von "Velocity" wird verwendet.
Current	INPUT	BOOL	FALSE	Aktuelle Geschwindigkeit beibehalten	
				FALSE	Deaktiviert Die Werte der Parameter "Velocity" und "Direction" werden berücksichtigt.
				TRUE	Aktiviert Die Werte an den Parametern "Velocity" und "Direction" werden **nicht** berücksichtigt. Die zum Funktionsstart aktuelle Geschwindigkeit und Richtung werden beibehalten. Sobald die Achse mit der zum Funktionsstart aktuellen Geschwindigkeit weiter verfährt, liefert der Parameter "InVelocity" den Wert TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveVelocity"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl

Gehen Sie zum Verfahren einer Achse mit konstanter Geschwindigkeit / Drehzahl folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Geben Sie am Parameter "Velocity" die Geschwindigkeit / Drehzahl an, mit der die Achse verfahren werden soll.
3. Starten Sie den "MC_MoveVelocity"-Auftrag mit einer steigenden Flanke am Parameter "Execute".

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Wenn der Parameter "InVelocity" den Wert TRUE zeigt, wurde die Sollgeschwindigkeit / Solldrehzahl erreicht. Die Achse wird mit dieser Geschwindigkeit konstant weiter verfahren. Die Parameter "InVelocity" und "Busy" zeigen solange den Wert TRUE, bis der "MC_MoveVelocity"-Auftrag von einem anderen Motion Control-Auftrag abgelöst wird.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit / Drehzahl ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_MoveVelocity: Funktionsdiagramm V1

Funktionsdiagramm: Verfahren einer Achse mit Geschwindigkeitsvorgabe und ablösendes Auftragsverhalten

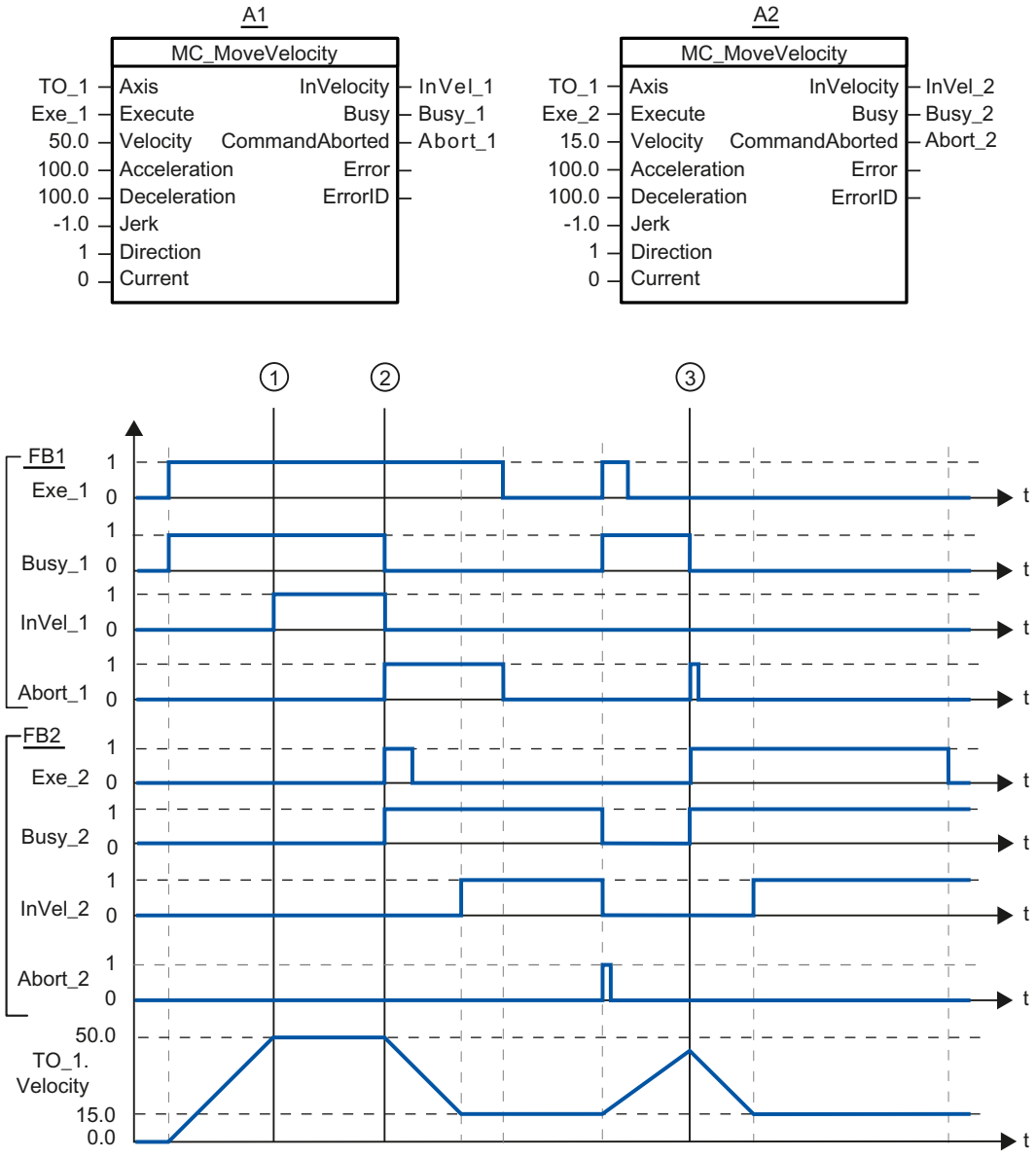

Ein über "Exe_1"angestoßener "MC_MoveVelocity"-Auftrag (A1) beschleunigt die Achse und meldet zum Zeitpunkt ① über "InVel_1" das Erreichen der Sollgeschwindigkeit 50.0.

Zum Zeitpunkt ② wird der Auftrag durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

Der laufende "MC_MoveVelocity"-Auftrag (A2) wird durch einen weiteren "MC_MoveVelocity"-Auftrag (A1) abgelöst. Der Abbruch wird über "Abort_2" gemeldet. Die Achse wird auf die neue Sollgeschwindigkeit 50.0 beschleunigt. Vor dem Erreichen der Sollgeschwindigkeit wird der aktuelle "MC_MoveVelocity"-Auftrag (A1) zum Zeitpunkt ③ durch einen weiteren "MC_MoveVelocity"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Sollgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der Geschwindigkeit 15.0 konstant weiter verfahren.

MC_MoveJog V1

MC_MoveJog: Achse im Tippbetrieb bewegen V1

Beschreibung

Mit der Motion Control-Anweisung "MC_MoveJog" bewegen Sie eine Achse im Tippbetrieb.

Mit den Parametern "Velocity", "Jerk", "Acceleration" und "Deceleration" bestimmen Sie das dynamische Verhalten beim Bewegungsvorgang.

- Positionierachse:
  Am Parameter "Velocity" wird eine Geschwindigkeit vorgegeben.
- Drehzahlachse:
  Am Parameter "Velocity" wird eine Drehzahl vorgegeben.

Anwendbar auf

- Positionierachse
- Drehzahlachse

Voraussetzung

- Das Technologieobjekt wurde korrekt konfiguriert.
- Das Technologieobjekt ist freigegeben.

Ablöseverhalten

Der "MC_MoveJog"-Auftrag wird abgebrochen durch:

- Sperren der Achse mit "MC_Power.Enable" = FALSE
- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Der Start eines "MC_MoveJog"-Auftrags bricht folgende laufende Motion Control-Aufträge ab:

- "MC_Home"-Auftrag "HomingMode" = 4, 5
- "MC_Halt"-Auftrag
- "MC_MoveAbsolute"-Auftrag
- "MC_MoveRelative"-Auftrag
- "MC_MoveVelocity"-Auftrag
- "MC_MoveJog"-Auftrag

Parameter

Die folgende Tabelle zeigt die Parameter der Motion Control-Anweisung "MC_MoveJog":

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung
Axis	InOut	TO_SpeedAxis	-	Technologieobjekt
JogForward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.
JogBackward	INPUT	BOOL	FALSE	Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.
Velocity	INPUT	LREAL	100.0	Sollgeschwindigkeit / Solldrehzahl für den Bewegungsvorgang Wert > 0.0: Der angegebene Wert wird verwendet. Wert < 0.0: Der Betrag des angegebenen Werts wird verwendet. ("Velocity" = 0.0 ist erlaubt)
Acceleration	INPUT	LREAL	-1.0	Beschleunigung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Beschleunigung wird verwendet. (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Verzögerung Wert > 0.0: Der angegebene Wert wird verwendet. Wert = 0.0: nicht zulässig Wert < 0.0: Die in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Verzögerung wird verwendet. (<TO>.DynamicDefaults.Deceleration)

Parameter	Deklaration	Datentyp	Defaultwert	Beschreibung	
Jerk	INPUT	LREAL	-1.0	Ruck Wert > 0.0: Beschleunigungsstetiges Geschwindigkeitsprofil, der angegebene Ruck wird verwendet. Wert = 0.0: trapezförmiges Geschwindigkeitsprofil Wert < 0.0: Der in "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik Voreinstellung" konfigurierte Ruck wird verwendet. (<TO>.DynamicDefaults.Jerk)	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Die Sollgeschwindigkeit / Solldrehzahl wurde erreicht und wird beibehalten.
Busy	OUTPUT	BOOL	FALSE	TRUE	Auftrag ist in Bearbeitung.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	FALSE	TRUE	Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Der Auftrag wird abgewiesen. Die Fehlerursache können Sie dem Parameter "ErrorID" entnehmen.
ErrorID	OUTPUT	WORD	16#0000	Fehlerkennung zum Parameter "ErrorID"	

Verhalten bei Sollgeschwindigkeit / Solldrehzahl Null ("Velocity" = 0.0)

Ein "MC_MoveJog"-Auftrag mit "Velocity" = 0.0 stoppt die Achse mit der konfigurierten Verzögerung. Mit dem Erreichen der Sollgeschwindigkeit / Solldrehzahl Null wird am Parameter "InVelocity" der Wert TRUE angezeigt.

Unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Bewegung" wird "konstante Geschwindigkeit" und "Stillstand" angezeigt (<TO>.StatusWord.X12 (ConstantVelocity); <TO>.StatusWord.X7 (Standstill)).

Verfahren einer Achse im Tippbetrieb

Gehen Sie zum Verfahren einer Achse im Tippbetrieb folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Verfahren Sie die Achse mit "JogForward" in positiver Richtung oder mit "JogBackward" in negativer Richtung.

An den Parametern "Busy", "InVelocity" und "Error" wird der aktuelle Bewegungszustand angezeigt.

Hinweis

Verhalten bei Änderung des Override

Wenn die Geschwindigkeit / Drehzahl während der konstanten Bewegung durch eine Änderung des Override beeinflusst wird (<TO>.Override.Velocity), wird der Parameter "InVelocity" während der Beschleunigung bzw. Verzögerung zurückgesetzt. Mit Erreichen der neu errechneten Geschwindigkeit ("Velocity" × "Override" %) wird "InVelocity" wieder gesetzt.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten.

MC_MoveJog: Funktionsdiagramm V1

Funktionsdiagramm: Verfahren einer Achse im Tippbetrieb

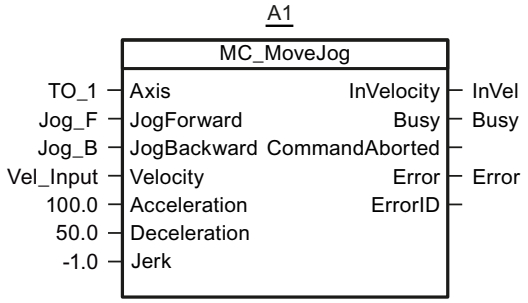

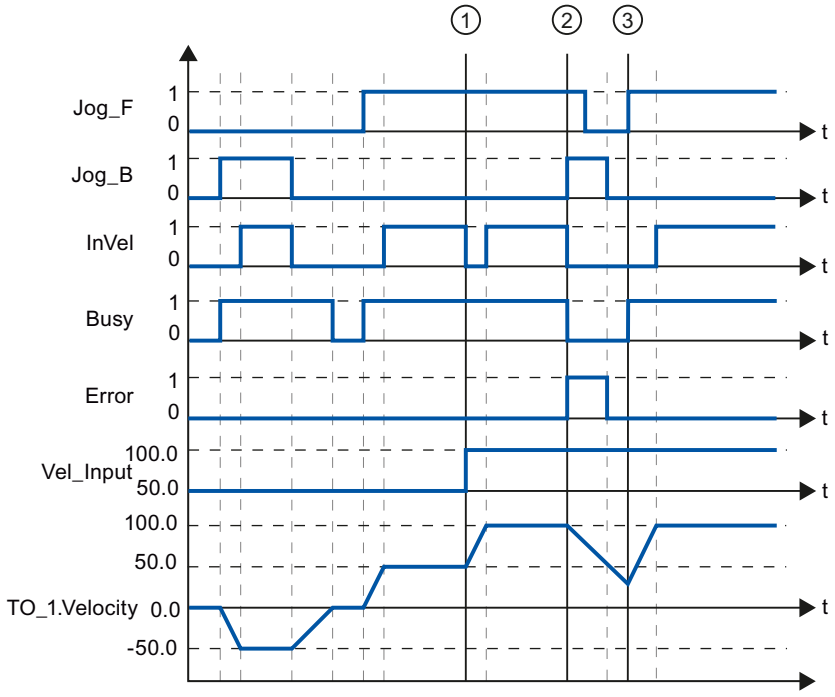

Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Sollgeschwindigkeit -50.0 wird über "InVel" = TRUE gemeldet. Nach dem Rücksetzen von "Jog_B" wird die Achse abgebremst und zum Stillstand gebracht. Anschließend wird die Achse über "Jog_F" in positiver Richtung verfahren. Das Erreichen der Sollgeschwindigkeit 50.0 wird über "InVel" = TRUE gemeldet.

Zum Zeitpunktpunkt ① wird bei gesetztem "Jog_F" die Sollgeschwindigkeit über "Vel_Input" auf 100.0 geändert. Alternativ können Sie die Sollgeschwindigkeit auch über den Geschwindigkeits-Override ändern. "InVel" wird rückgesetzt. Die Achse wird beschleunigt. Das Erreichen der neuen Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

Bei gesetztem "Jog_F" wird zum Zeitpunkt ② ebenfalls "Jog_B" gesetzt. Wenn sowohl "Jog_F" als auch "Jog_B" gesetzt sind, wird die Achse mit der zuletzt gültigen Verzögerung abgebremst. Über "Error" wird ein Fehler angezeigt und am Ausgang "ErrorID" der Fehler 16#8007 (falsche Richtungsangabe) ausgegeben.

Dieser Fehler wird durch Rücksetzen beider Eingänge "Jog_F" und "Jog_B" behoben.

Noch während der Bremsrampe wird zum Zeitpunkt ③ "Jog_F" gesetzt. Die Achse wird auf die zuletzt konfigurierte Geschwindigkeit beschleunigt. Das Erreichen der Sollgeschwindigkeit 100.0 wird über "InVel" = TRUE gemeldet.

4.1.5.5 Time-based IO

Time-based IO

TIO_SYNC: TIO-Module synchronisieren

Beschreibung

Mit der Anweisung TIO_SYNC können Sie Time-based IO einsetzen. Mit TIO_SYNC erreichen Sie eine Synchronisierung von TIO-Modulen auf eine gemeinsame Zeitbasis TIO_Time.

Mit TIO_SYNC können maximal 8 TIO-Module synchronisiert werden. Alle TIO-Module müssen dem gleichen Teilprozessabbild (PIP) zugeordnet sein. Wenn Sie für den Eingangsparameter PIP_Mode "0" wählen, parametrieren Sie die Nummer des Teilprozessabbilds am Eingangsparameter PIP_PART.

Anlaufverhalten

Beim Anlauf der CPU übernimmt und prüft die Anweisung TIO_SYNC einmalig die Eingangsparameter und initialisiert die TIO_Time. Nach einem fehlerfreien Anlauf wechselt die Anweisung in den Normalbetrieb. Im Falle eines Fehlers wechselt die Anweisung nicht in den Normalbetrieb und generiert eine Fehlermeldung.

Arbeitsweise

Im Normalbetrieb stellt die Anweisung die applikative Synchronisierung aller am Eingang HWID projektierten TIO-Module sicher.

Am Ausgang TIO_SYNC_Data wird die berechnete TIO_Time für die Anweisungen der TIO-Module bereitgestellt.

Verhalten im Fehlerfall

Am Ausgang Error wird angezeigt, ob die Anweisung korrekt parametriert und ob ein Fehler aufgetreten ist. Am Ausgang Status werden die Fehlerursachen bei nicht erfolgreicher Bearbeitung der Anweisung angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung TIO_SYNC.

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
		S7-1500		
HWID_1 ... HWID_8*	Input	HW_IO	-1	Hardware-Kennung für TIO-Modul aus HWCN
SendClock*	Input	LTime	LT#0ns	Sendetakt der Sync-Domain. Übernehmen Sie den Sendetakt aus der PROFINET-Konfiguration.
PIP_Mode*	Input	USInt	0	Modus für die Datenaktualisierung**: • 0: AEV-Modell mit interner Aktualisierung des Prozessabbilds durch SYNC_PO und SYNC_PI. • 1: AEV-Modell ohne interne Aktualisierung des Prozessabbilds • 2: EVA-Modell ohne interne Aktualisierung des Prozessabbilds
PIP_PART*	Input	USInt	1	Nur relevant, wenn PIP_Mode = 0 Nummer des Teilprozessabbildes, welches taktsynchron aktualisiert werden soll.
TIO_SYNC_Data	Output	"TIO_SYNC_Data"		Berechnete Systemzeit für die TIO-Anweisungen der TIO-Module und interne Daten, die zur Synchronisierung verwendet werden. Siehe UDT "TIO_SYNC_Data" (Seite 3589).
Status	Output	DWord	16#0	Status der Anweisung: siehe Beschreibung Parameter Status
Error	Output	Bool	False	Falls True, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten ist. Detailinformationen siehe Parameter Status. Error wird zurückgesetzt, sobald der Fehler behoben ist.

* Einmalig beim Anlauf der CPU geprüft

** Das EVA-Modell (PIP_Mode = 2) lässt die geringsten Reaktionszeiten zu, stellt jedoch die höchsten Anforderungen an die Systemleistung. Die Bearbeitung aller TIO-Anweisungen und der weiteren Programmteile muss innerhalb eines Sendetakts abgeschlossen sein.

Parameter Status

Am Ausgang Status werden Fehlercodes bzw. Statusinformationen als Doppelwort ausgegeben.

Das Doppelwort hat folgende Aufteilung:

Segment des Fehlercodes	Bedeutung
z0yywwww	Fehler einer Systemfunktion: • im Anlauf (z = 0) • im Normalbetrieb (z = 1) In yy sind unterlagert genutzte Systemfunktionen codiert: Siehe Tabelle mit Fehlercodes. wwww gibt den RET_VAL der Systemfunktion an. Sie finden die Fehlerinformation in der Hilfe zu der Systemfunktion.
z0yy0000	Ein Fehler, der nicht von einer Systemfunktion stammt. Dieser erhält eine laufende Fehlernummer yy. Der Fehler kann auftreten: • im Anlauf (z = 0) • im Normalbetrieb (z = 1)

Tabelle mit Fehlercodes

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00000000	Kein Fehler vorhanden.	—
1001xxxx	Es ist ein Fehler bei der Systemfunktion RD_SINFO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_SINFO an.	• Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_SINFO. • Stellen Sie sicher, dass TIO_SYNC in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10020000	Die ausgelesene Zykluszeit des taktsynchronen OBs ist LT#0ms oder negativ, und damit ungültig. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	• Korrigieren Sie den Wert der Zykluszeit. • Stellen Sie sicher, dass TIO_SYNC in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10030000	Die Anweisung TIO_SYNC wird nicht in einem taktsynchronen OB aufgerufen. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden. Mögliche Ursache: Die Anweisung TIO_SYNC wurde in einem OB des Typs "MC-Servo" oder "MC-Interpolator" aufgerufen.	Stellen Sie sicher, dass TIO_SYNC in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10040000	Der parametrierte Wert am Eingangsparameter PIP_Mode ist außerhalb des gültigen Bereichs von 0 ... 2.	Korrigieren Sie den Wert am Eingangsparameter PIP_Mode.
10050000	Der parametrierte Sendetakt ist außerhalb des zulässigen Bereichs von 0 < SendClock <= 4 ms, und damit ungültig.	Korrigieren Sie den Wert des Sendetakts.
0006xxxx*	Es ist ein Fehler bei der Ausführung der Systemfunktion SYNC_PI aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von SYNC_PI an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des SYNC_PI.
0007xxxx*	Es ist ein Fehler bei der Ausführung der Systemfunktion SYNC_PO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von SYNC_PO an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des SYNC_PO.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
10080000	Taktsynchronität fehlt. Time-based IO kann nicht genutzt werden. Mögliche Ursache: Interner Fehler bei der Adressberechnung	Prüfen Sie den Werte der Eingangsparameter HWID_1 bis HWID_8. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an.
10090000	HWID_1 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	Prüfen Sie den Wert des jeweiligen Eingangsparameters. Geben Sie jeweils die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
100A0000	HWID_2 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
100B0000	HWID_3 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
100C0000	HWID_4 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
100D0000	HWID_5 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
100E0000	HWID_6 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
100F0000	HWID_7 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
10100000	HWID_8 ist keine gültige Hardware-Kennung. Mögliche Ursache: Kein Modul mit dieser Hardware-Kennung vorhanden.	
10110000	Das projektierte Teilprozessabbild am Eingangsparameter PIP_PART ist außerhalb des gültigen Bereichs von 1 ... 31.	Korrigieren Sie den Wert am Eingangsparameter PIP_PART.
x0FF0000	Allgemeiner interner Fehler.	—

* Nur möglich, wenn für den Eingangsparameter PIP_Mode "0" gewählt ist.

TIO_IOLink_IN: Prozesseingangssignale mit Zeitstempeln einlesen

Beschreibung

Mit der Anweisung TIO_IOLink_IN können Sie Time-based IO einsetzen. TIO_IOLink_IN erfasst ein Ereignis am IO-Link Device und liefert den Prozesswert, inklusive zugehörigem Zeitstempel.

Das IO-Link Device muss mit der Funktion zum Zeitstempeln ausgestattet sein und der Port muss im Modus "IO-Link, Time based IN" sein.

Anlaufverhalten

Beim Anlauf der CPU übernimmt die Anweisung TIO_IOLink_IN einmalig die Eingangsparameter und prüft Folgendes:

- HWID prüfen
- Prüfen, ob die Port-Nummer im gültigen Bereich (1 bis 4) liegt
- TIO_SYNC_Data.Error prüfen: Liegt bereits am TIO_SYNC ein Fehler vor?
- T_o auf einen positiven Wert prüfen
- Einstellung auf den Typ IO-Link prüfen
- Port-Modus auf die Konfiguration für IO-Link Time based IN prüfen
- Prüfen, ob der OB vom Typ "Synchronous Cycle" ist
- PortQualifier prüfen

Nach einem fehlerfreien Anlauf wechselt die Anweisung in den Normalbetrieb. Im Falle eines Fehlers wechselt die Anweisung nicht in den Normalbetrieb und generiert eine Fehlermeldung.

Arbeitsweise

Im Normalbetrieb erfasst die Anweisung die Prozessdaten (SA_Data) eines IO-Link Device und den zugehörigen Zeitstempel (Zeit = TIO_Time) der letzten, gültigen Änderung. Jede gültige Änderung der SA_Data, bei einem fehlerfrei funktionierenden Port, wird von einem gültigen Zeitstempel begleitet.

Den Eingang TIO_SYNC_Data verschalten Sie mit dem gleichnamigen Ausgang der Anweisung TIO_SYNC. Dadurch stellen Sie eine gemeinsame Zeitbasis sicher.

Verhalten im Fehlerfall

Am Ausgang Error wird angezeigt, ob die Anweisung fehlerfrei bearbeitet wurde. Am Ausgang Status werden die Fehlerursachen im Fehlerfall angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung TIO_IOLink_IN.

Parameter	Deklaration	Datentyp S7-1500	Vorbelegung	Beschreibung
HWID*	Input	HW_IO	0	Hardware-Kennung für TIO-Modul aus HWCN
Port*	Input	USInt	0	Port-Nummer (1 ... 4) des verbundenen IO-Link Device
TIO_SYNC_Data*	Input	"TIO_SYNC_Data"		Von Anweisung TIO_SYNC bereitgestellte Systemzeit für die TIO-Anweisungen der TIO-Module. Siehe UDT "TIO_SYNC_Data" (Seite 3589). Dieser Eingangsparameter ist mit dem Ausgangsparameter TIO_SYNC_Data der Anweisung TIO_SYNC zu verschalten.
TO*	Input	LTime	LT#0ns	T_o: Zeit zum Ausgeben der taktsynchronen Ausgangsdaten. Übernehmen Sie T_o von der ET 200-Station (Eigenschaften der PROFINET-Schnittstelle).

Parameter	Deklaration	Datentyp S7-1500	Vorbelegung	Beschreibung
SA_Bit0	Output	Bool	False	Anzeige einer Prozessdatenänderung (Bit 0 von SA_Data). SA_Bit0 ist ereignisorientiert. Solange kein gültiger Zeitstempel erfasst wurde (EventCount = 0), sind die Prozessdaten ungültig.
SA_Bit1	Output	Bool	False	Anzeige einer Prozessdatenänderung (Bit 1 von SA_Data). SA_Bit1 ist ereignisorientiert. Solange kein gültiger Zeitstempel erfasst wurde (EventCount = 0), sind die Prozessdaten ungültig.
SA_Data	Output	DWord	16#0	Prozessdaten SA_Data (Sensor Application Data)
TimeStamp	Output	LTime	LT#0ns	Zeitstempel: Zeit, zu der eine Änderung des Prozesssignals SA-Data am IO-Link Device stattgefunden hat.
EventCount	Output	UInt	0	Zähler: Wird mit jedem neuen, gültigen Zeitstempel inkrementiert. Mit jedem CPU-Anlauf wird der Zähler zurückgesetzt.
Status	Output	DWord	16#0	Status der Anweisung: siehe Beschreibung Parameter Status
Error	Output	Bool	False	Falls True, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten ist. Detailinformationen siehe Parameter Status. Error wird zurückgesetzt, sobald der Fehler behoben ist.

* Einmalig beim Anlauf der CPU geprüft

Parameter Status

Am Ausgang Status werden Fehlercodes bzw. Statusinformationen als Doppelwort ausgegeben.

Das Doppelwort hat folgende Aufteilung:

Segment des Fehlercodes	Bedeutung
z0yywwww	Fehler einer Systemfunktion: • im Anlauf (z = 0) • im Normalbetrieb (z = 1) In yy sind unterlagert genutzte Systemfunktionen codiert: Siehe Tabelle mit Fehlercodes. wwww gibt den RET_VAL der Systemfunktion an. Sie finden die Systemfunktion in der Hilfe zu der Systemfunktion.
z0yy0000	Ein Fehler, der nicht von einer Systemfunktion stammt. Dieser erhält eine laufende Fehlernummer yy. Der Fehler kann auftreten: • im Anlauf (z = 0) • im Normalbetrieb (z = 1)

Tabelle mit Fehlercodes

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00000000	Kein Fehler vorhanden.	—
10010000	Die am Eingangsparameter Port parametrierte Nummer ist außerhalb des gültigen Bereichs von 1 ... 4.	Korrigieren Sie den Wert am Eingangsparameter Port.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
10020000	Die Anweisung TIO_IOLink_IN wird nicht in einem taktsynchronen OB aufgerufen. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	Stellen Sie sicher, dass TIO_IOLink_IN in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10030000	Es ist ein Fehler beim Lesen des Eingangsparameters HWID aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
x0040000	Die Daten in TIO_SYNC_Data sind ungültig/fehlerhaft.	Prüfen Sie die Anweisung TIO_SYNC sowie die Verschaltung von dessen Ausgang TIO_SYNC_Data.
1005xxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_SINFO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_SINFO an.	• Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_SINFO. • Stellen Sie sicher, dass TIO_IOLink_IN in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10060000	Keine IO-Link Device gefunden. Mögliche Ursache: Das über die Hardware-Kennung projektierte Modul ist kein IO-Link Master für Time-based IO.	• Stellen Sie sicher, dass das projektierte Modul ein IO-Link Master für Time-based IO ist. • Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an.
10070000	Es ist ein interner Fehler bei der Adressberechnung aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
00080000	TIO-Modul ist nicht über die Anweisung TIO_SYNC synchronisiert.	Prüfen Sie die Anweisung TIO_SYNC.
10090000	Die ausgelesene Zykluszeit des taktsynchronen OB ist außerhalb des zulässigen Bereichs von $0 < T_{APP} <= 16$ ms, und damit ungültig. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	• Korrigieren Sie die Zykluszeit. • Stellen Sie sicher, dass TIO_IOLink_IN in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
100A0000	Die parametrierte Zeit am Eingangsparameter TO ist außerhalb des zulässigen Bereichs von $0 < T_o <= 4$ ms.	Korrigieren Sie den Wert am Eingangsparameter TO.
100Bxxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_ADDR aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_ADDR an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_ADDR.
000C0000	Der umgerechnete Zeitstempel ist ungültig.	Prüfen Sie den angeschlossenen Sensor und das Zusammenspiel zwischen IO-Link Master und Sensor (z. B. Konfiguration).

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
000D0000	Der Wertestatus PortQualifier des IO-Link zeigt an, dass die Prozessdaten ungültig sind.	Prüfen Sie den angeschlossenen Sensor und dessen Konfiguration.
100E0000	Der konfigurierte Port-Modus des IO-Link ist fehlerhaft.	Prüfen Sie mit S7-PCT die Konfiguration des angeschlossenen Sensors.
x0FF0000	Allgemeiner interner Fehler.	—

TIO_DI: Flanken am Digitaleingang und zugehörige Zeitstempel einlesen

Beschreibung

Mit der Anweisung TIO_DI können Sie Time-based IO einsetzen. TIO_DI erfasst kontinuierlich die Flanken an einem Digitaleingang eines TM Timer DIDQ und liefert die zugehörigen Zeitstempel.

Anlaufverhalten

Beim Anlauf der CPU übernimmt die Anweisung TIO_DI einmalig die Eingangsparameter und prüft Folgendes:

- HWID prüfen
- Prüfen, ob Nummer des Digitaleingangs (Channel) im erlaubten Bereich liegt (abhängig von adressiertem Modul und Kanalkonfiguration)
- TIO_SYNC_Data.Error prüfen: Liegt bereits am TIO_SYNC ein Fehler vor?
- T_o auf Plausibiltiät prüfen (0 ms bis 4 ms)
- Prüfen, ob der OB vom Typ "Synchronous Cycle" ist

Nach einem fehlerfreien Anlauf wechselt die Anweisung in den Normalbetrieb. Der Eingangsparameter EdgeSel kann während des Normalbetriebs geändert werden. Im Falle eines Fehlers wechselt die Anweisung nicht in den Normalbetrieb und generiert eine Fehlermeldung.

Arbeitsweise

Im Normalbetrieb erfasst die Anweisung die Flanken an einem Digitaleingang und die zugehörigen Zeitstempel (Zeit = TIO_Time) des letzten gültigen, definierten Flankenpaars. Mit dem Eingangsparameter EdgeSel bestimmen Sie, für welche Flanken Zeitstempel erfasst werden.

Den Eingang TIO_SYNC_Data verschalten Sie mit dem gleichnamigen Ausgang der Anweisung TIO_SYNC. Dadurch stellen Sie eine gemeinsame Zeitbasis sicher.

Verhalten im Fehlerfall

Am Ausgang Error wird angezeigt, ob die Anweisung fehlerfrei bearbeitet wurde. Am Ausgang Status werden die Fehlerursachen im Fehlerfall angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung TIO_DI.

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
		S7-1500		
HWID*	Input	HW_IO	0	Hardware-Kennung für TIO-Modul aus HWCN
Channel*	Input	UInt	0	Nummer (0 ... m) des Digitaleingangs des verbundenen TM Timer DIDQ
TIO_SYNC_Data*	Input	"TIO_SYNC_Data"		Von Anweisung TIO_SYNC bereitgestellte Systemzeit für die TIO-Anweisungen der TIO-Module. Siehe UDT "TIO_SYNC_Data" (Seite 3589). Dieser Eingangsparameter ist mit dem Ausgangsparameter TIO_SYNC_Data der Anweisung TIO_SYNC zu verschalten.
EdgeSel	Input	UInt	3	Festlegung, zu welchen Flanken Zeitstempel erfasst werden: 0_D: Rearm / Wiederholung Zeitstempelerfassung 1_D: Zwei steigende Flanken 2_D: Zwei fallende Flanken 3_D: Steigende und fallende Flanke (Reihenfolge je nach Auftreten) 4_D: Erst steigende, dann fallende Flanke 5_D: Erst fallende, dann steigende Flanke 6 bis 255_D: Unzulässig
TO*	Input	LTime	LT#0ns	T_o: Zeit zum Ausgeben der taktsynchronen Ausgangsdaten. Übernehmen Sie T_o von der ET 200-Station (Eigenschaften der PROFINET-Schnittstelle).
DI	Output	Bool	False	Zustand des Digitaleingangs. Wenn eine Invertierung des Digitaleingangs parametriert ist, ist dieser Parameter auch invertiert.
TimeStamp-RE	Output	LTime	LT#0ns	Zeitstempel: Zeit, zu der eine steigende Flanke erkannt wurde. Ausnahme: EdgeSel = 2_D: Zeit, zu der eine fallende Flanke erkannt wurde (falls *mehrere* fallende Flanken im Applikationszyklus aufgetreten sind).
TimeStamp-FE	Output	LTime	LT#0ns	Zeitstempel: Zeit, zu der eine fallende Flanke erkannt wurde. Ausnahme: EdgeSel = 1_D: Zeit, zu der eine steigende Flanke erkannt wurde (falls *mehrere* steigende Flanken im Applikationszyklus aufgetreten sind).
EventCount-RE	Output	UInt	0	Zähler: Wird mit jedem neuen, gültigen Zeitstempel einer steigenden Flanke inkrementiert. Mit jedem CPU-Anlauf wird der Zähler zurückgesetzt.
EventCount-FE	Output	UInt	0	Zähler: Wird mit jedem neuen, gültigen Zeitstempel einer fallenden Flanke inkrementiert. Mit jedem CPU-Anlauf wird der Zähler zurückgesetzt.
LEC	Output	UInt	0	Zähler: Anzahl der Flanken, zu denen kein Zeitstempel gespeichert werden konnte. Das Modul kann pro Applikationszyklus maximal sieben Flanken zählen. Mit jedem neuen Applikationszyklus wird der Zähler zurückgesetzt.

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
		S7-1500		
Status	Output	DWord	16#0	Status der Anweisung: siehe Beschreibung Parameter Status
Error	Output	Bool	False	Falls True, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten ist. Detailinformationen siehe Parameter Status. Error wird zurückgesetzt, sobald der Fehler behoben ist.

* Einmalig beim Anlauf der CPU geprüft

Parameter Status

Am Ausgang Status werden Fehlercodes bzw. Statusinformationen als Doppelwort ausgegeben.

Das Doppelwort hat folgende Aufteilung:

Segment des Fehlercodes	Bedeutung
z0yywwww	Fehler einer Systemfunktion: • im Anlauf (z = 0) • im Normalbetrieb (z = 1) In yy sind unterlagert genutzte Systemfunktionen codiert: Siehe Tabelle mit Fehlercodes. wwww gibt den RET_VAL der Systemfunktion an. Sie finden die Systemfunktion in der Hilfe zu der Systemfunktion.
z0yy0000	Ein Fehler, der nicht von einer Systemfunktion stammt. Dieser erhält eine laufende Fehlernummer yy. Der Fehler kann auftreten: • im Anlauf (z = 0) • im Normalbetrieb (z = 1)

Tabelle mit Fehlercodes

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00000000	Kein Fehler vorhanden.	—
10010000	Die am Eingangsparameter Channel parametrierte Nummer des Digitaleingangs ist außerhalb des gültigen Bereichs (abhängig von adressiertem Modul und Kanalkonfiguration).	Korrigieren Sie den Wert am Eingangsparameter Channel.
10020000	Die Anweisung TIO_DI wird nicht in einem taktsynchronen OB aufgerufen. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	Stellen Sie sicher, dass TIO_DI in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10030000	Es ist ein Fehler beim Lesen des Eingangsparameters HWID aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
x0040000	Die Daten in TIO_SYNC_Data sind ungültig/fehlerhaft.	Prüfen Sie die Anweisung TIO_SYNC sowie die Verschaltung von dessen Ausgang TIO_SYNC_Data.
1005xxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_SINFO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_SINFO an.	• Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_SINFO. • Stellen Sie sicher, dass TIO_DI in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10060000	Kein TM Timer DIDQ gefunden. Mögliche Ursache: Das über die Hardware-Kennung projektierte Modul ist kein TM Timer DIDQ.	• Stellen Sie sicher, dass das projektierte Modul ein TM Timer DIDQ ist. • Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an.
10070000	Es ist ein interner Fehler bei der Adressberechnung aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
00080000	TIO-Modul ist nicht über die Anweisung TIO_SYNC synchronisiert.	Prüfen Sie die Anweisung TIO_SYNC.
10090000	Die ausgelesene Zykluszeit des taktsynchronen OBs ist außerhalb des zulässigen Bereichs von $0 < T_{APP} <= 16$ ms, und damit ungültig. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	• Korrigieren Sie die Zykluszeit. • Stellen Sie sicher, dass TIO_DI in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
100A0000	Die parametrierte Zeit am Eingangsparameter TO ist außerhalb des zulässigen Bereichs von $0 < T_o <= 4$ ms.	Korrigieren Sie den Wert am Eingangsparameter TO.
100Bxxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_ADDR aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_ADDR an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_ADDR.
000C0000	Der umgerechnete Zeitstempel ist ungültig. Mögliche Ursache: Störung in der Kommunikation	Prüfen Sie die PROFINET-Kommunikation zum TIO-Modul.
000D0000	Die Quality Information des Digitaleingangs zeigt an, dass am Digitaleingang ein Fehler aufgetreten ist.	Prüfen Sie die Verdrahtung des Digitaleingangs.
000E0000	Die parametrierte Nummer am Eingangsparameter Channel ist kein als Timer-DI parametrierter Digitaleingang.	• Prüfen Sie die Kanalkonfiguration (nur bei TM Timer DIDQ 16x24V). • Prüfen Sie die Betriebsart des Digitaleingangs.
000F0000	Der parametrierte Wert am Eingangsparameter EdgeSel ist außerhalb des gültigen Bereichs von 1 ... 5.	Korrigieren Sie den Wert am Eingangsparameter EdgeSel.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
10100000	Der Sendetakt ist außerhalb des zulässigen Bereichs von 0 < SendClock <= 4 ms, und damit ungültig. Der Fehlercode kann auch darauf hinweisen, dass • die Daten in TIO_SYNC_Data ungültig oder nicht vorhanden sind. • die Anweisung TIO_DI nicht in einem taktsynchronen OB aufgerufen wird.	Korrigieren Sie den Sendetakt.
x0FF0000	Allgemeiner interner Fehler.	—

TIO_IOLink_OUT: Prozessausgangssignale zeitgesteuert ausgeben

Beschreibung

Mit der Anweisung TIO_IOLink_OUT können Sie Time-based IO einsetzen. TIO_IOLink_OUT ermöglicht die Aktivierung der Ausgabedaten eines IO-Link Device zu einer vordefinierten Zeit.

Das IO-Link Device muss mit der Funktion zum Zeitstempeln ausgestattet sein und der Port muss im Modus "IO-Link, Time based OUT" sein.

Anlaufverhalten

Beim Anlauf der CPU übernimmt die Anweisung TIO_IOLink_OUT einmalig die Eingangsparameter und prüft Folgendes:

- HWID prüfen
- Prüfen, ob Port-Nummer im gültigen Bereich (1 bis 4) liegt
- TIO_SYNC_Data.Error prüfen: Liegt bereits am TIO_SYNC ein Fehler vor?
- T_o auf positiven Wert prüfen
- Einstellung auf den Typ IO-Link prüfen
- Port-Modus auf die Konfiguration für IO-Link Time based OUT prüfen
- Prüfen, ob der OB vom Typ "Synchronous Cycle" ist
- PortQualifier prüfen

Nach einem fehlerfreien Anlauf wechselt die Anweisung in den Normalbetrieb. Die Eingangsparameter REQ, Out_Mode, TimeStamp und AA_Data können während des Normalbetriebs geändert werden. Im Falle eines Fehlers wechselt die Anweisung nicht in den Normalbetrieb und generiert eine Fehlermeldung.

Arbeitsweise

Im Normalbetrieb sendet die Anweisung die Prozessdaten (AA_Data) an ein IO-Link Device. Die Ausgangsdaten AA_Data werden zu dem am Eingangsparameter TimeStamp definierten Zeitpunkt aktiviert.

Den Eingang TIO_SYNC_Data verschalten Sie mit dem gleichnamigen Ausgang der Anweisung TIO_SYNC. Dadurch stellen Sie eine gemeinsame Zeitbasis sicher.

Einen Ausgabeauftrag starten Sie mit einer steigenden Flanke am Parameter "REQ". Es kann nur dann ein neuer Auftrag gestartet werden, wenn kein Fehler anliegt und kein Auftrag aktiv ist. Beim Start des Ausgabeauftrags werden die AA_Data (Bit 0, 1) zu einem durch TimeStamp definierten Zeitpunkt auf dem IO-Link Device aktiviert. Mit Ablauf des letzten Applikationszyklus vor Erreichen der Ausgabezeit wird der Auftrag beendet (Done). Status und Error werden während der Laufzeit des Auftrags ständig aktualisiert.

Hinweis

Nach dem Starten des Auftrags über eine steigende Flanke können Sie den Ausgabezeitpunkt durch eine neue Angabe von TimeStamp adaptieren, ohne den Auftrag neu zu starten.

Randbedingung:

Wenn der adaptierte Zeitstempel weniger als 16 ms vor dem Ausgabezeitpunkt liegt (TimeStamp - TIO_Time < 16), wird der zuletzt gültige Zeitstempel verwendet.

Wenn Sie als TimeStamp den Wert 0 vorgeben, wird der Ausgang direkt mit den am Eingang AA_Data vorgegebenen Daten beschrieben. Damit haben Sie die Möglichkeit, im Handbetrieb eine direkte Ansteuerung vom TIO-Modul ohne Zeitstempel zu implementieren. Sie können die direkte Ansteuerung nutzen, um einen laufenden Auftrag zu unterbrechen.

Verhalten im Fehlerfall

Am Ausgang Error wird angezeigt, ob die Anweisung fehlerfrei bearbeitet wurde. Am Ausgang Status werden die Fehlerursachen im Fehlerfall angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung TIO_IOLink_OUT.

Parameter	Deklaration	Datentyp S7-1500	Vorbelegung	Beschreibung
REQ	Input	Bool	False	Startet den Auftrag bei einer steigenden Flanke.
HWID*	Input	HW_IO	0	Hardware-Kennung für TIO-Modul aus HWCN.
TO*	Input	LTime	LT#0ns	T_o: Zeit zum Ausgeben der taktsynchronen Ausgangsdaten. Übernehmen Sie T_o von der ET 200-Station (Eigenschaften der PROFINET-Schnittstelle).
Port*	Input	USInt	0	Port-Nummer (1 ... 4) des bestimmten IO-Link Device
TIO_SYNC_Data*	Input	"TIO_SYNC_Data"		Von Anweisung TIO_SYNC bereitgestellte Systemzeit für die TIO-Anweisungen der TIO-Module. Siehe UDT "TIO_SYNC_Data" (Seite 3589). Dieser Eingangsparameter ist mit dem Ausgangsparameter TIO_SYNC_Data der Anweisung TIO_SYNC zu verschalten.
TimeStamp	Input	LTime	LT#0ns	Zeitstempel: Zeit, zu der die Prozessdaten (AAE1, AAE2) ausgegeben werden sollen.
AA_Data	Input	Word	16#0	Prozessausgangsdaten: Auszugebendes Datum (Wort). Enthält AAE1 und AAE2 auf Bit 0, 1.

Parameter	Deklaration	Datentyp S7-1500	Vorbelegung	Beschreibung
Busy	Output	Bool	False	Busy = True: Der Auftrag ist noch nicht abgeschlossen.
Done	Output	Bool	False	Done = True wird einen Zyklus lang angezeigt: Der Auftrag wird als "fertig, ohne Fehler" signalisiert.
Error	Output	Bool	False	Falls True, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten ist. In diesem Fall sind BUSY und DONE auf False gesetzt. Detailinformationen siehe Parameter Status. Error wird zurückgesetzt, sobald der Fehler behoben ist.
Status	Output	DWord	16#0	Status der Anweisung: siehe Beschreibung Parameter Status

* Einmalig beim Anlauf der CPU geprüft

Parameter Status

Am Ausgang Status werden Fehlercodes bzw. Statusinformationen als Doppelwort ausgegeben.

Das Doppelwort hat folgende Aufteilung:

Segment des Fehlercodes	Bedeutung
z0yywwww	Fehler einer Systemfunktion: • im Anlauf (z = 0) • im Normalbetrieb (z = 1) In yy sind unterlagert genutzte Systemfunktionen codiert: Siehe Tabelle mit Fehlercodes. wwww gibt den RET_VAL der Systemfunktion an. Sie finden die Fehlerinformation in der Hilfe zu der Systemfunktion.
z0yy0000	Ein Fehler, der nicht von einer Systemfunktion stammt. Dieser erhält eine laufende Fehlernummer yy. Der Fehler kann auftreten: • im Anlauf (z = 0) • im Normalbetrieb (z = 1)

Tabelle mit Fehlercodes

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00000000	Kein Fehler vorhanden.	—
10010000	Die am Eingangsparameter Port parametrierte Nummer ist außerhalb des gültigen Bereichs von 1 ... 4.	Korrigieren Sie den Wert am Eingangsparameter Port.
10020000	Die Anweisung TIO_IOLink_OUT wird nicht in einem taktsynchronen OB aufgerufen. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	Stellen Sie sicher, dass TIO_IOLink_OUT in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10030000	Es ist ein Fehler beim Lesen des Eingangsparameters HWID aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
x0040000	Die Daten in TIO_SYNC_Data sind ungültig/fehlerhaft.	Prüfen Sie die Anweisung TIO_SYNC sowie die Verschaltung von dessen Ausgang TIO_SYNC_Data.
1005xxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_SINFO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_SINFO an.	• Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_SINFO. • Stellen Sie sicher, dass TIO_IOLink_OUT in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10060000	Keine IO-Link Device gefunden. Mögliche Ursache: Das über die Hardware-Kennung projektierte Modul ist kein IO-Link Master für Time-based IO.	• Stellen Sie sicher, dass das projektierte Modul ein IO-Link Master für Time-based IO ist. • Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an.
10070000	Es ist ein interner Fehler bei der Adressberechnung aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in der Hardware-Konfiguration an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
00080000	TIO-Modul ist nicht über die Anweisung TIO_SYNC synchronisiert.	Prüfen Sie die Anweisung TIO_SYNC.
10090000	Die ausgelesene Zykluszeit des taktsynchronen OB ist außerhalb des zulässigen Bereichs von $0 < T_{APP} <= 16$ ms, und damit ungültig. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	• Korrigieren Sie die Zykluszeit. • Stellen Sie sicher, dass TIO_IOLink_OUT in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
100A0000	Die parametrierte Zeit am Eingangsparameter TO ist außerhalb des zulässigen Bereichs von $0 < T_o <= 4$ ms.	Korrigieren Sie den Wert am Eingangsparameter TO.
100Bxxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_ADDR aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_ADDR an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_ADDR.
000C0000	Der Zeitstempel am Eingangsparameter TimeStamp ist ungültig.	Prüfen Sie den Eingangsparameter TimeStamp.
000D0000	Der Wertestatus PortQualifier des IO-Link zeigt an, dass die Prozessdaten ungültig sind.	Prüfen Sie den angeschlossenen Sensor und dessen Konfiguration.
100E0000	Der konfigurierte Port-Modus des IO-Link ist fehlerhaft.	Prüfen Sie mit S7-PCT die Konfiguration des angeschlossenen Sensors.
100F0000	Die ausgelesene Zykluszeit des OB des Typs "Synchronous Cycle" ist zu lang: $T_{APP} > 16$ ms.	Projektieren Sie als Zykluszeit ein kleineres Vielfaches des Sendetakts.
x0FF0000	Allgemeiner interner Fehler.	—

TIO_DQ: Flanken am Digitalausgang zeitgesteuert ausgeben

Beschreibung

Mit der Anweisung TIO_DQ können Sie Time-based IO einsetzen. TIO_DQ ermöglicht das Schalten eines Digitalausgangs eines TM Timer DIDQ zu definierten Zeiten.

Anlaufverhalten

Beim Anlauf der CPU übernimmt die Anweisung TIO_DQ einmalig die Eingangsparameter und prüft Folgendes:

- HWID prüfen
- Prüfen, ob Nummer des Digitalausgangs (Channel) im erlaubten Bereich liegt (abhängig von adressiertem Modul und Kanalkonfiguration)
- TIO_SYNC_Data.Error prüfen: Liegt bereits am TIO_SYNC ein Fehler vor?
- T_o auf Plausibiltiät prüfen (0 ms bis 4 ms)
- Prüfen, ob der OB vom Typ "Synchronous Cycle" ist

Nach einem fehlerfreien Anlauf wechselt die Anweisung in den Normalbetrieb. Die Eingangsparameter REQ, Out_Mode, TimeStampRE und TimeStampFE können während des Normalbetriebs geändert werden. Im Falle eines Fehlers wechselt die Anweisung nicht in den Normalbetrieb und generiert eine Fehlermeldung.

Arbeitsweise

Im Normalbetrieb gibt die Anweisung an einem Digitalausgang zeitgesteuert Flanken aus.

- Zu dem am Eingangsparameter TimeStampRE definierten Zeitpunkt wird am Digitalausgang eine steigende Flanke ausgegeben.
- Zu dem am Eingangsparameter TimeStampFE definierten Zeitpunkt wird am Digitalausgang eine fallende Flanke ausgegeben.

Abhängig vom Modell der Programmbearbeitung, muss ein Zeitstempel den folgenden Wert übersteigen:

Modell der Programmbearbeitung	TimeStampRE > ... TimeStampFE > ...
EVA-Modell	TIO_Time + T_{APP} + T_O
AEV-Modell	TIO_Time + T_{APP} + SendClock + T_O

Mit dem Eingangsparameter Out_Mode bestimmen Sie, ob nur eine oder beide Flanken ausgegeben werden.

Den Eingang TIO_SYNC_Data verschalten Sie mit dem gleichnamigen Ausgang der Anweisung TIO_SYNC. Dadurch stellen Sie eine gemeinsame Zeitbasis sicher.

Einen Ausgabeauftrag starten Sie mit einer steigenden Flanke am Parameter "REQ". Es kann nur dann ein neuer Auftrag gestartet werden, wenn kein Fehler anliegt und kein Auftrag aktiv

ist. Beim Start des Ausgabeauftrags wird der Digitalausgang zu den durch TimeStampRE und TimeStampFE definierten Zeitpunkten geschaltet.

- Wenn der Digitalausgang zum Zeitpunkt TimeStampRE bereits gesetzt ist, wird der Ausgabeauftrag für die steigende Flanke nicht zum Modul übertragen.
- Wenn der Digitalausgang zum Zeitpunkt TimeStampFE nicht gesetzt ist, wird der Ausgabeauftrag für die fallende Flanke nicht zum Modul übertragen.

In beiden Fällen wird somit der Digitalausgang nicht geschaltet.

Mit Ablauf des letzten Applikationszyklus vor Erreichen der zweiten Ausgabezeit wird der Auftrag beendet (DONE). Status und Error werden während der Laufzeit des Auftrags ständig aktualisiert. Einen aktiven Ausgabeauftrag können Sie abbrechen, indem Sie TimeStampRE oder TimeStampFE mit einem ungültigen Zeitstempel adaptieren.

Das folgende Bild zeigt ein Beispiel für die Reaktionen der Bits DONE und BUSY bei Start eines Ausgabeauftrags unter folgenden Bedingungen:

- Out_Mode = 2
- Die beiden Zeitstempel werden zwischen Start des Auftrags und der Ausgabe nicht adaptiert.

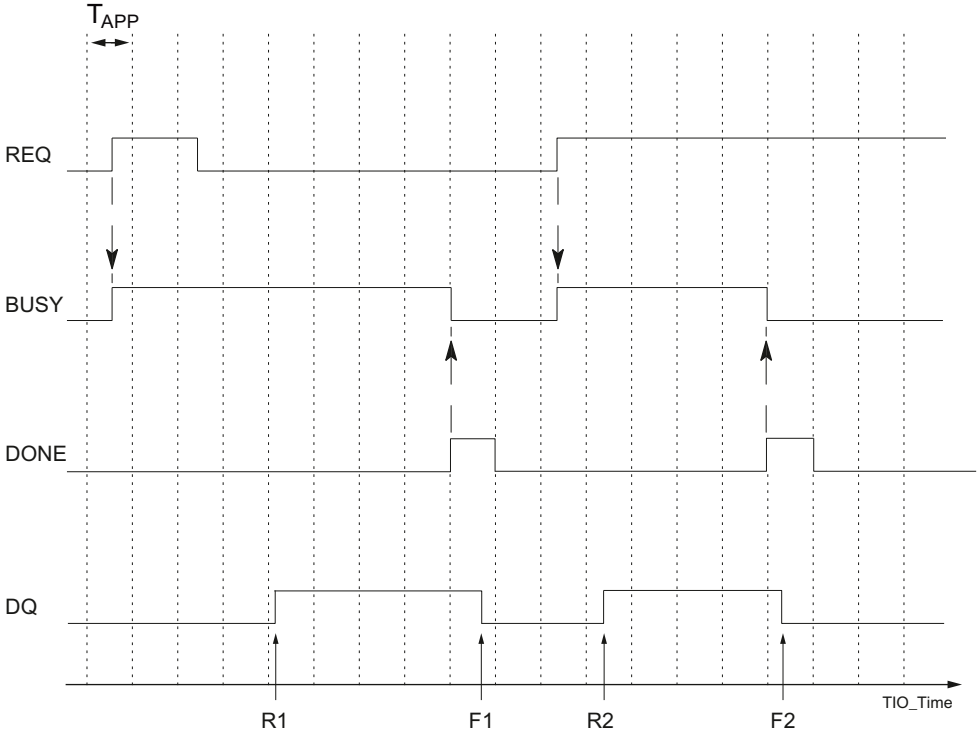

T_{APP}	Applikationszyklus
R1, R2	Vorgegebene Zeitpunkte einer steigenden DQ-Flanke
F1, F2	Vorgegebene Zeitpunkte einer fallenden DQ-Flanke

Hinweis

Nach dem Starten des Auftrags über eine steigende Flanke können Sie die Ausgabezeitpunkte durch eine neue Angabe von TimeStampRE und TimeStampFE adaptieren, ohne den Auftrag neu zu starten.

Randbedingung:

Wenn ein adaptierter Zeitstempel weniger als zwei Applikationszyklen vor dem Ausgabezeitpunkt liegt (TimeStampRE - TIO_Time < $2*T_{APP}$ oder TimeStampFE - TIO_Time < $2*T_{APP}$), wird der zuletzt gültige Zeitstempel verwendet.

Wenn Sie als TimeStampRE oder TimeStampFE den Wert 0 vorgeben, können Sie mit dem Eingangsparameter Out_Mode = 3 die jeweilige Flanke am Digitalausgang direkt ausgeben. Damit haben Sie die Möglichkeit, im Handbetrieb eine direkte Ansteuerung vom TIO-Modul ohne Zeitstempel zu implementieren. Sie können die direkte Ansteuerung nutzen, um einen laufenden Auftrag zu unterbrechen.

Änderungen an den Eingangsparametern werden erst übernommen, sobald die Anweisung TIO_DQ einen Anlauf der CPU erkennt.

Verhalten im Fehlerfall

Am Ausgang Error wird angezeigt, ob die Anweisung fehlerfrei bearbeitet wurde. Am Ausgang Status werden die Fehlerursachen im Fehlerfall angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung TIO_DQ.

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
		S7-1500		
REQ	Input	Bool	False	Startet den Auftrag bei einer steigenden Flanke.
HWID*	Input	HW_IO	0	Hardware-Kennung für TIO-Modul aus HWCN.
Channel*	Input	UInt	0	Nummer (0 ... m) des Digitalausgangs des verbundenen TM Timer DIDQ
TIO_SYNC_Data*	Input	"TIO_SYNC_Data"		Von Anweisung TIO_SYNC bereitgestellte Systemzeit für die TIO-Anweisungen der TIO-Module. Siehe UDT "TIO_SYNC_Data" (Seite 3589). Dieser Eingangsparameter ist mit dem Ausgangsparameter TIO_SYNC_Data der Anweisung TIO_SYNC zu verschalten.

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
		S7-1500		
Out_Mode	Input	UInt	2	Festlegung des Ausgabemodus für die Flanken am Digitalausgang: 0_D: Nur steigende Flanke wird ausgegeben (TimeStampRE). 1_D: Nur fallende Flanke wird ausgegeben (TimeStampFE). 2_D: Beide Flanken werden ausgegeben (TimeStampRE und TimeStampFE). 3_D: Jeweilige Flanke wird direkt ausgegeben, wenn TimeStampRE = 0 oder TimeStampFE = 0. Wenn beide Zeitstempel mit "0" belegt sind oder kein Zeitstempel mit "0" belegt ist, wird keine Flanke ausgegeben. 4 bis 255_D: Unzulässig
TO*	Input	LTime	LT#0ns	T_o: Zeit zum Ausgeben der taktsynchronen Ausgangsdaten. Übernehmen Sie T_o von der ET 200-Station (Eigenschaften der PROFINET-Schnittstelle).
TimeStampRE	Input	LTime	LT#0ns	Zeitstempel: Zeit, zu der eine steigende Flanke ausgegeben werden soll.
TimeStampFE	Input	LTime	LT#0ns	Zeitstempel: Zeit, zu der eine fallende Flanke ausgegeben werden soll.
StatusDQ	Output	Bool	False	Istzustand des Digitalausgangs. Wenn eine Invertierung des Digitalausgangs parametriert ist, ist StatusDQ auch invertiert. TIO_DQ kann die Parametrierung des jeweiligen Digitalausgangs und dessen tatsächlichen Zustand nicht prüfen. Wenn eine HW-Freigabe parametriert und der Digitalausgang während eines Ausgabeauftrag nicht freigegeben ist, kann StatusDQ einen falschen Wert liefern.
BUSY	Output	Bool	False	BUSY = True: Der Auftrag ist noch nicht abgeschlossen.
DONE	Output	Bool	False	DONE = True wird einen Zyklus lang angezeigt: Der Auftrag wird als "fertig, ohne Fehler" signalisiert.
Status	Output	DWord	16#0	Status der Anweisung: siehe Beschreibung Parameter Status
Error	Output	Bool	False	Falls True, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten ist. In diesem Fall sind BUSY und DONE auf False gesetzt. Detailinformationen siehe Parameter Status. Error wird zurückgesetzt, sobald der Fehler behoben ist.

* Einmalig beim Anlauf der CPU geprüft

Parameter Status

Am Ausgang Status werden Fehlercodes bzw. Statusinformationen als Doppelwort ausgegeben.

Das Doppelwort hat folgende Aufteilung:

Segment des Fehlercodes	Bedeutung
z0yywwww	Fehler einer Systemfunktion: • im Anlauf (z = 0) • im Normalbetrieb (z = 1) In yy sind unterlagert genutzte Systemfunktionen codiert: Siehe Tabelle mit Fehlercodes. wwww gibt den RET_VAL der Systemfunktion an. Sie finden die Fehlerinformation in der Hilfe zu der Systemfunktion.
z0yy0000	Ein Fehler, der nicht von einer Systemfunktion stammt. Dieser erhält eine laufende Fehlernummer yy. Der Fehler kann auftreten: • im Anlauf (z = 0) • im Normalbetrieb (z = 1)

Tabelle mit Fehlercodes

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00000000	Kein Fehler vorhanden.	—
10010000	Die am Eingangsparameter Channel parametrierte Nummer des Digitalausgangs ist außerhalb des gültigen Bereichs (abhängig von adressiertem Modul und Kanalkonfiguration).	Korrigieren Sie den Wert am Eingangsparameter Channel.
10020000	Die Anweisung TIO_DQ wird nicht in einem taktsynchronen OB aufgerufen. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	Stellen Sie sicher, dass TIO_DQ in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10030000	Es ist ein Fehler beim Lesen des Eingangsparameters HWID aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in HWCN an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
x0040000	Die Daten in TIO_SYNC_Data sind ungültig/fehlerhaft.	Prüfen Sie die Anweisung TIO_SYNC sowie die Verschaltung von dessen Ausgang TIO_SYNC_Data.
1005xxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_SINFO aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_SINFO an.	• Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_SINFO. • Stellen Sie sicher, dass TIO_SYNC in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
10060000	Kein TM Timer DIDQ gefunden. Mögliche Ursache: Das über die Hardware-Kennung projektierte Modul ist kein TM Timer DIDQ.	• Stellen Sie sicher, dass das projektierte Modul ein TM Timer DIDQ ist. • Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in HWCN an.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
10070000	Es ist ein interner Fehler bei der Adressberechnung aufgetreten.	Prüfen Sie den Wert am Eingangsparameter HWID. Geben Sie die Hardware-Kennung des TIO-Moduls aus dessen Moduleigenschaften in HWCN an. Für die symbolische Adressierung steht eine interne Systemkonstante vom Datentyp Hw_SubModule zur Verfügung.
x0080000	TIO-Modul ist nicht über die Anweisung TIO_SYNC synchronisiert. Der Fehlercode kann auch darauf hinweisen, dass: • vor dem ersten Durchlauf der Anweisung schon ein Auftrag ansteht. • die parametrierte Nummer am Eingangsparameter Channel kein Digitalausgang ist.	Prüfen Sie die Anweisung TIO_SYNC.
10090000	Die ausgelesene Zykluszeit des taktsynchronen OBs ist LT#0ms oder negativ, und damit ungültig. Korrigieren Sie den Wert. Die Anweisung kann nur in einem taktsynchronen OB fehlerfrei verwendet werden.	• Korrigieren Sie die Zykluszeit. • Stellen Sie sicher, dass TIO_DQ in einem OB des Typs "Synchronous Cycle" aufgerufen wird.
100A0000	Die parametrierte Zeit am Eingangsparameter TO ist außerhalb des zulässigen Bereichs von $0 < T_o <= 4$ ms.	Korrigieren Sie den Wert am Eingangsparameter TO.
100Bxxxx	Es ist ein Fehler bei der Ausführung der Systemfunktion RD_ADDR aufgetreten. Das Low-Wort xxxx zeigt die Fehlerinformation des Rückgabewertes RET_VAL von RD_ADDR an.	Lesen Sie im TIA Portal Infosystem die Beschreibung des RD_ADDR.
000C0000	Ein oder beide Zeitstempel an den Eingangsparametern TimeStampRE und TimeStampFE sind ungültig. Der Fehler wird nur für die Dauer eines Applikationszyklus gemeldet.	Prüfen Sie die Eingangsparameter TimeStampRE und TimeStampFE.
000D0000	Die Quality Information des Digitalausgangs zeigt an, dass am Digitalausgang ein Fehler aufgetreten ist.	• Prüfen Sie die Parametrierung des Digitalausgangs. • Prüfen Sie die Verdrahtung des Digitalausgangs bezüglich Kurzschluss, Überlast und Übertemperatur.
000E0000	Die parametrierte Nummer am Eingangsparameter Channel ist kein als Timer-DQ parametrierter Digitalausgang.	• Prüfen Sie die Kanalkonfiguration (nur bei TM Timer DIDQ 16x24V). • Prüfen Sie die Betriebsart des Digitalausgangs.
100F0000	Die ausgelesene Zykluszeit des OB des Typs "Synchronous Cycle" ist zu lang: $T_{APP} > 16$ ms.	Projektieren Sie als Zykluszeit ein kleineres Vielfaches des Sendetakts.
10100000	Der Sendetakt ist außerhalb des zulässigen Bereichs von 0 < SEND_CLOCK <= 4 ms, und damit ungültig. Der Fehlercode kann auch darauf hinweisen, dass • die Daten in TIO_SYNC_Data ungültig oder nicht vorhanden sind. • die Anweisung TIO_DQ nicht in einem taktsynchronen OB aufgerufen wird.	Korrigieren Sie den Sendetakt.

Fehlercode (DW#16#...)	Bedeutung	Abhilfe
00110000	Der parametrierte Wert am Eingangsparameter Out_Mode ist außerhalb des gültigen Bereichs von 0 ... 3.	Korrigieren Sie den Wert am Eingangsparameter Out_Mode.
x0FF0000	Allgemeiner interner Fehler.	—

UDT "TIO_SYNC_Data"

Beschreibung

Der Datentyp UDT "TIO_SYNC_Data" enthält die zentrale Struktur und die Daten zur Synchronisierung der Module und Weitergabe der TIO_Time.

Parameter

Parameter	Datentyp S7-1500	Beschreibung
TIO_TIME_BASE	LTime	Interne Verwendung
TIO_TIME	LTime	Gemeinsame Zeitbasis (relative Zeit) der TIO-Module.
PIP_MODE	USInt	Modus für die Datenaktualisierung (wird vom Eingangsparameter PIP_Mode der Anweisung TIO_SYNC weitergereicht)
APP_CYC	LTime	Applikationszyklus des OB "Synchronous Cycle"
SEND_CLOCK	LTime	Sendetakt der Sync-Domain (wird vom Eingangsparameter SendClock der Anweisung TIO_SYNC weitergereicht)
TBase	LTime	Interne Verwendung
ERROR	Bool	Interne Verwendung

4.1.6 Kommunikation

4.1.6.1 S7-Kommunikation

Datenkonsistenz

Definition

Ein Datenblock, der nicht durch konkurrierende Prozesse verändert werden kann, wird als konsistenter Datenbereich bezeichnet. Ein Datenblock, der größer als der konsistente Datenbereich ist, kann somit in seiner Gesamtheit bei der Übertragung verfälscht werden. Das heißt, ein in sich zusammengehöriger Datenblock, der größer als der konsistente Datenbereich ist, kann zu einem Zeitpunkt teilweise aus neuen und aus alten konsistenten Daten bestehen.

Beispiel

Eine Inkonsistenz kann entstehen, wenn eine Anweisung für Kommunikation z. B. durch einen Prozessalarm-OB mit höherer Priorität unterbrochen wird. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die teilweise bereits von der Anweisung verarbeitet wurden, stammen die übertragenen Daten:

- zum einen Teil aus der Zeit vor der Prozessalarm-Bearbeitung
- und zum anderen Teil aus der Zeit nach der Prozessalarm-Bearbeitung.

Das bedeutet, dass diese Daten inkonsistent (nicht zueinander passend) sind.

Sicherstellung der Datenkonsistenz

Wenn der Kommunikationsvorgang durch ein Alarm-OB unterbrochen werden kann, müssen Sie sicherstellen, dass die Daten konsistent übertragen werden. Sorgen Sie dafür, dass die zu übertragenden Daten nicht direkt durch den Alarm-OB geändert werden, sondern nur ein Abbild der Daten. Kopieren Sie vor dem nächsten Daten-Transfer das Abbild der Daten in den Transfer-Bereich der Kommunikationsanweisung.

- Existiert im Anwenderprogramm eine Kommunikationsanweisung, welche auf gemeinsame Daten zugreift, kann der Zugriff auf diesen Datenbereich z. B. über den Parameter DONE selbst koordiniert werden. Die Datenkonsistenz der Kommunikationsbereiche, die lokal mit einer Kommunikationsanweisung übertragen wird, kann deshalb im Anwenderprogramm sichergestellt werden.
- Bei S7-Kommunikationsanweisungen "PUT (Seite 3600)" / "GET (Seite 3594)" muss bereits bei der Programmierung bzw. Projektierung die Größe der konsistenten Datenbereiche berücksichtigt werden, da im Anwenderprogramm des Zielgerätes (Server) kein Kommunikationsbaustein vorhanden ist, der die Kommunikationsdaten in das Anwenderprogramm einsynchronisiert:
- Bei der S7-300 und C7-300 (Ausnahme: CPU 318-2 DP) werden die Kommunikationsdaten in Blöcken zu 32 Bytes im Zykluskontrollpunkt des Betriebssystems, konsistent in den Anwenderspeicher kopiert. Für alle größeren Datenbereiche wird keine Datenkonsistenz garantiert. Ist eine definierte
  Datenkonsistenz gefordert, so dürfen die Kommunikationsdaten im Anwenderprogramm nicht größer als 32 Bytes sein (je nach Ausgabestand maximal 8 Byte).
- Bei der S7-400 und S7-1500 werden im Gegensatz dazu die Kommunikationsdaten in Blöcken zu 462 Bytes nicht am Zykluskontrollpunkt, sondern in festen Zeitscheiben während des Programmzyklus bearbeitet. Systemseitig wird die Konsistenz einer Variable garantiert. Auf diese Kommunikationsbereiche kann dann, z. B. von einem OP oder von einer OS, mit den "PUT (Seite 3600)" / "GET (Seite 3594)"-Anweisungen bzw. Lesen/ Schreiben von Variablen konsistent zugegriffen werden.

Hinweis

Weitere Hinweise zu Datenkonsistenz finden Sie bei der Beschreibung der einzelnen Anweisungen.

In der Baugruppenfamilie S7-1500 ist es bei den Anweisungen der S7-Kommunikation möglich, Parameter während des laufenden Betriebs zu ändern. Geänderte Parameter werden sofort verwendet, auch wenn noch ein Auftrag aktiv ist. Beides kann zu Dateninkonsistenz

führen! Um Dateninkonsistenzen zu vermeiden, sollten Sie auf keinen Fall während eines laufenden Auftrags Parameter verändern.

Auswirkung auf die Alarmreaktionszeiten

Durch das Kopieren der Daten werden die Alarmreaktionszeiten der CPU minimal verlängert. Je größer die Menge der garantiert konsistent zu übertragenen Daten, desto länger die Alarmreaktionszeit eines Systems.

Gemeinsame Parameter der Anweisungen zur S7-Kommunikation

Klassifikation

Die Parameter der Anweisungen zur S7-Kommunikation lassen sich von ihrer Funktion her in die folgenden fünf Klassen einteilen:

1. Steuerparameter dienen der Aktivierung einer Anweisung.
2. Adressierungsparameter dienen der Adressierung des remoten Kommunikationspartners.
3. Sendeparameter zeigen auf diejenigen Datenbereiche, die zum remoten Partner gesendet werden sollen.
4. Empfangsparameter zeigen auf diejenigen Datenbereiche, in welche die vom remoten Partner empfangenen Daten eingetragen werden.
5. Zustandsparameter dienen zur Überwachung, ob die Anweisung ihre Aufgabe fehlerfrei beendet hat, bzw. zur Analyse der aufgetretenen Fehler.

Steuerparameter

Die Aktivierung des Datenaustauschs erfolgt nur dann, wenn beim Aufruf der Anweisung die zugehörigen Steuerparameter einen definierten Wert haben (z. B. gesetzt sind) bzw. wenn sich der Wert gegenüber dem letzten Aufruf in definierter Weise geändert hat (z. B. positive Flanke).

Adressierungsparameter

Parameter	Beschreibung
ID	Referenz auf die lokale Verbindungsbeschreibung (wird durch die Verbindungsprojektierung vorgegeben).
R_ID	Mit dem Parameter R_ID legen Sie die Zusammengehörigkeit einer Sende- und einer Empfangs-Anweisung fest: Der Parameter R_ID muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite übereinstimmen. Damit wird die Kommunikation mehrerer Anweisungspaare über dieselbe logische Verbindung ermöglicht. • R_ID muss in der Form DW#16#wxyzWXYZ angegeben werden. • Die über R_ID festgelegten Anweisungspaare einer logischen Verbindung müssen für diese Verbindung eindeutig sein.

Hinweis

Adressierungsparameter ID und R_ID

Sie können die Adressierungsparameter ID und R_ID zur Laufzeit umparametrieren. Die neuen Parameter werden mit jedem neuen Auftrag nach Abschluss des vorangegangenen Auftrags wirksam.

Mit folgenden Möglichkeiten können Sie Instanz-DBs und somit Arbeitsspeicher sparen:

1. Durch variable IDs können Sie mehrere Verbindungen über einen Dateninstanzbaustein nutzen.
2. Durch variable R_IDs können Sie mit einer Instanz mehrere Zusammengehörigkeiten von Sende- und Empfangs-Anweisungspaaren für einen Auftrag festlegen.
3. Sie können die Fälle 1 und 2 miteinander kombinieren.

Beachten Sie, dass die neuen Parameter erst nach Abschluss des vorangegangenen Auftrags gültig werden. Wenn Sie den Sendevorgang aktivieren, muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite der Parameter R_ID übereinstimmen.

Zustandsparameter

Mit den Zustandsparametern überwachen Sie, ob die Anweisung ihre Aufgabe ordnungsgemäß beendet hat oder ob sie noch tätig ist. Darüber hinaus zeigen diese aufgetretene Fehler an.

Hinweis

Die Zustandsparameter sind nur einen Zyklus lang gültig, nämlich vom ersten Befehl, der dem Aufruf folgt, bis zum nächsten Aufruf. Daraus folgt, dass Sie diese Parameter nach jedem Durchlauf der Anweisung auswerten müssen.

Sende- und Empfangsparameter

Bei den Anweisungen zur zweiseitigen Kommunikation

- muss die Anzahl der verwendeten SD_i-und RD_i-Parameter auf Sende- und Empfangsseite übereinstimmen
- müssen die Datentypen bei zusammengehörigen SD_i- und RD_i-Parametern auf Sende- und Empfangsseite übereinstimmen
- darf die Anzahl der über den Parameter SD_i zu sendenden Daten nicht größer sein als der Bereich, der durch den zugehörigen Parameter RD_i aufgespannt wird (gilt nicht für "BSEND (Seite 3615)" / "BRCV (Seite 3617)"). Die Parameter RD_i müssen (mit Ausnahme von "BSEND"/"BRCV") die gleich Datengröße adressieren.

Wenn Sie gegen die genannten Regeln verstoßen, so wird Ihnen dies mit ERROR = 1 und STATUS = 4 angezeigt.

Hinweis

Versorgung der Sende- und Empfangsparameter

Sende- und Empfangsparameter mit dem Datentyp VARIANT müssen beim Aufruf aller Kommunikations-Anweisung immer versorgt werden. Es ist nicht möglich die Sendeparameter der Kommunikations-Anweisungen im Anlauf zu versorgen und im zyklischen Betrieb nur den Sendeauftrag anzustoßen.

Nutzdatengröße

Bei den Anweisungen "USEND (Seite 3606)", "URCV (Seite 3608)", "GET (Seite 3594)" und "PUT (Seite 3600)" darf die Anzahl der zu übertragenden Daten eine bestimmte Nutzdatengröße nicht überschreiten. Die maximale Nutzdatengröße ist abhängig von:

- verwendeter Anweisung und
- Kommunikationspartner.

Die garantierte Mindestgröße der Nutzdaten für eine Anweisung mit 1-4 Variablen können Sie folgender Tabelle entnehmen:

Anweisung	Partner: S7-300	Partner: S7-400	Partner: S7-1200	Partner: S7-1500
PUT / GET	160 Byte	400 Byte	160 Byte	880 Byte
USEND / URCV	160 Byte	440 Byte	-	920 Byte
BSEND / BRCV	32768/65534 Byte	65534 Byte	-	• 65534 Byte bei Standardzugriff • 65535 Byte bei optimiertem Zugriff

Beachten Sie die Einschränkungen zur Nutzdatengröße in den Technischen Daten der jeweiligen CPU.

Exakte Nutzdatengröße

Wenn die oben angegebene Nutzdatengröße nicht ausreicht, können Sie die exakte maximale Nutzdatenlänge in Byte wie folgt ermitteln:

Zunächst lesen Sie aus der folgenden Tabelle die bei der Kommunikation gültige Datenblockgröße ab:

Eigene CPU	Remote CPU	Datenblockgröße in Byte
S7-1200	beliebig	240
S7-1500	S7-300	240
	S7-400	480
	S7-1200	240
	S7-1500	960

Mit diesem Wert gehen Sie in die folgende Tabelle und lesen daraus die maximal mögliche Nutzdatenlänge in Byte als Summe der benutzten Parameter ab. Sie gilt für gerade Längen der Bereiche SD_i, RD_i, ADDR_i.

Für jeden Bereich ungerader Länge reduziert sich die maximal mögliche Nutzdatenlänge um ein Byte.

		Anzahl benutzter Parameter SD_i, RD_i, ADDR_i			
Datenblockgröße	Anweisung	1	2	3	4
240 (S7-300)	PUT/GET/ USEND	160	-	-	-
240 (S7-300 via integrierte Schnittstelle)	PUT	212	-	-	-
	GET	222	-	-	-
	USEND	212	-	-	-
240 (S7-400)	PUT	212	196	180	164
	GET	222	218	214	210
	USEND	212	-	-	-
480 (S7-400)	PUT	452	436	420	404
	GET	462	458	454	450
	USEND	452	448	444	440
240 (S7-1200)	PUT	212	196	180	164
	GET	222	218	214	210
960 (S7-1500)	PUT	932	916	900	884
	GET	942	938	934	930
	USEND	932	928	924	920

GET: Daten aus einer remoten CPU lesen

Beschreibung

Mit der Anweisung "GET" können Sie Daten aus einer remoten CPU auslesen.

Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet:

- Dabei werden die relevanten Zeiger auf die auszulesenden Bereiche (ADDR_i) an die Partner-CPU gesendet. Die Partner-CPU kann sich im Betriebszustand RUN oder STOP befinden.
- Die Partner-CPU sendet die Dateninhalte zurück:
  - Überschreitet die Antwort die maximale Nutzdatenlänge, wird dies am Parameter STATUS mit dem Fehlercode "2" angezeigt.
  - Die empfangenen Daten werden beim nächsten Anweisungs-Aufruf in die projektierten Empfangsbereiche (RD_i) kopiert.
- Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt.

Eine erneute Aktivierung eines Lesevorgangs ist erst nach dem Abschluss des letzten möglich. Wenn beim Lesen der Daten Zugriffsprobleme auftraten, oder die Datentypprüfung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Änderungen in den adressierten Datenbereichen auf der Partner-CPU werden durch die Anweisung "GET" nicht registriert.

Voraussetzungen zur Verwendung der Anweisung

- In den Eigenschaften der Partner-CPU wurde unter "Schutz" die Funktion "Zugriff über PUT/GET-Kommunikation durch entfernten Partner erlauben" aktiviert.
- Die Bausteine, auf die Sie mit der Anweisung "GET" zugreifen, wurden mit der Zugriffsart "standard" erstellt.
- Sie müssen darauf achten, dass die über die Parameter ADDR_i und SD_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinanderpassen.
- Der zu lesende Bereich (Parameter ADDR_i) darf nicht größer sein als der Bereich zur Datenablage (Parameter RD_i).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: – 0000H: weder Warnung noch Fehler – <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	
ADDR_1	InOut	REMOTE	E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen. Wenn der REMOTE-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren. Beispiel: P#DB10.DBX5.0 Byte 10.
ADDR_2	InOut	REMOTE		
ADDR_3	InOut	REMOTE		
ADDR_4	InOut	REMOTE		
RD_1	InOut	VARIANT	E, A, M, D, L	Zeiger auf diejenigen Bereiche in der eigenen CPU, in der die gelesenen Daten abgelegt werden.
RD_2	InOut	VARIANT		
RD_3	InOut	VARIANT		
RD_4	InOut	VARIANT		

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "GET" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut
1	2	• Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar. • Antwort der remoten Station überschreitet die maximale Nutzdatenlänge (siehe: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)). • Zugriffsschutz bei der Partner-CPU ist aktiviert. Deaktivieren Sie den Zugriffsschutz in den CPU-Einstellungen.

ERROR	STATUS (dezimal)	Erläuterung
1	4	Fehler in den Zeigern zur Datenablage RD_i: • Datentypen der Parameter RD_i und ADDR_i passen nicht zueinander. • Die Länge des Bereichs RD_i ist kleiner als die Länge der zu lesenden Daten der Parameter ADDR_i.
1	8	Zugriffsfehler bei der Partner-CPU.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Hinweis

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie den aktuell benutzten Teil des Empfangsbereichs RD_i vollständig auslesen, bevor Sie einen erneuten Auftrag aktivieren.

Beispiel

In dem folgenden Beispiel verwenden Sie eine S7-Verbindung und lesen in der CPU 1 einen Datensatz der CPU 2 aus. Der zu übermittelnde Datensatz ist vom Datentyp INT.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Die Verbindung ist noch nicht konfiguriert.
- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.
- Der Zugriff per PUT/GET-Anweisung ist erlaubt.

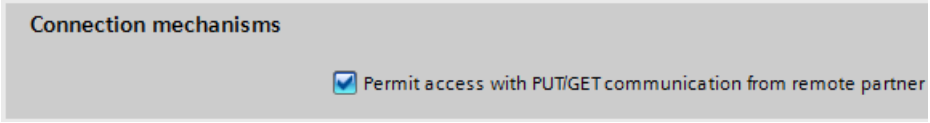

Programm der CPU 1

Zur Ablage der Daten von "GET" legen Sie in einem globalen Datenbaustein sechs Variablen an.

SLI_gDB_GET			
	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	done	Bool	false
4	memDoneStat	Word	16#0
5	error	Bool	false
6	status	Word	16#0
7	memErrStatus	Word	16#0

Zur Ablage des zu empfangenen Datensatzes legen Sie einen globalen Datenbaustein mit einer Variable an.

SLI_gDB_getData				
	Name	Data type	Offset	Start value
1	Static			
2	readValue	Int	0.0	0

Netzwerk 1: Die Parameter der Anweisung "GET" verschalten Sie wie folgt.

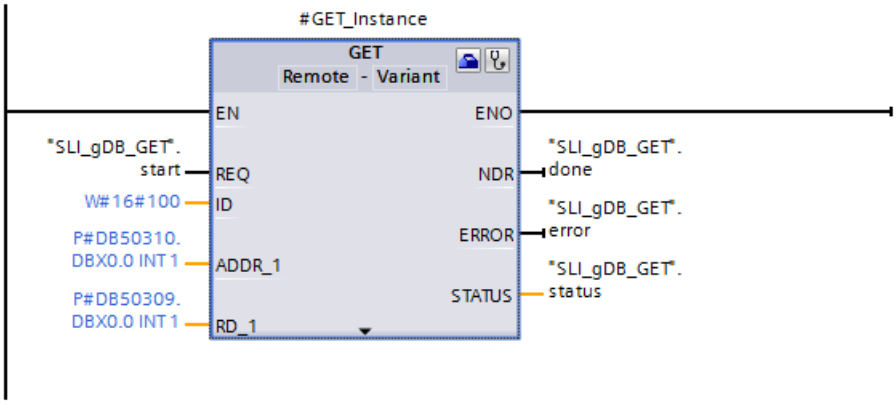

Netzwerk 2: Den Status bei einem Fehler von GET speichern Sie wie folgt.

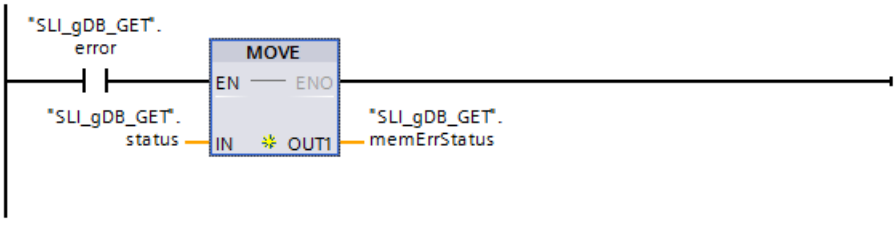

Netzwerk 3: Den Erfolgstatus von GET speichern Sie wie folgt.

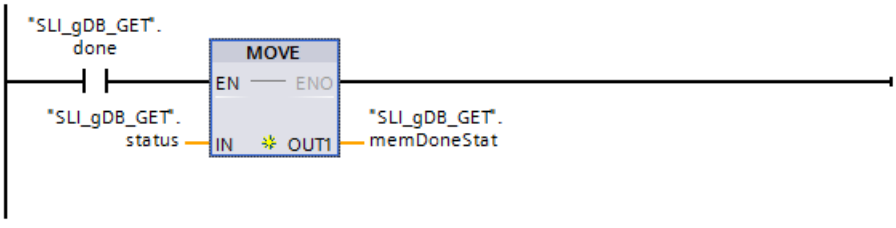

Konfiguration von GET

Zur Verschaltung des Eingangsparameters ID öffnen Sie den Wizard der Anweisung "GET" über "Eigenschaften > Konfiguration".

Folgende Einstellungen nehmen Sie für die Verbindung vor:

Abschnitt	Einstellung
Endpunkt	Sie wählen den Kommunikationspartner aus. Die restlichen Verbindungsdaten werden automatisch eingetragen. Eine S7-Verbindung wird automatisch angelegt und deren Kennung am Eingangsparameter ID eingetragen.
Aktiv	Sie stellen sicher, dass die aktive Verbindung von der CPU 1 ausgeht.

Programm der CPU 2

Zur Ablage des zu sendenden Datensatzes legen Sie einen globalen Datenbaustein mit einer Variable an.

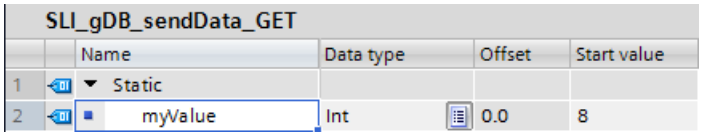

Unter "Eigenschaften > Attribute" des Datenbausteins deaktivieren Sie den optimierten Bausteinzugriff.

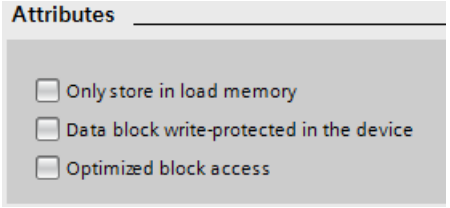

Verhalten von GET

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "GET" gestartet. Die Anweisung ruft die Verbindungsdaten ab und nimmt Kontakt auf zum Kommunikationspartner der verwendeten S7-Verbindung. Dazu ist am Eingangsparameter ID die Kennung der S7-Verbindung hinterlegt.

Über den Parameter ADDR_1 wird der Ablageort des zu sendenden Datensatzes ("myValue") der CPU 2 erfasst. Der Datensatz ("myValue") wird ausgelesen und am Parameter RD_1 erfassten Ablageort ("readValue") eingetragen.

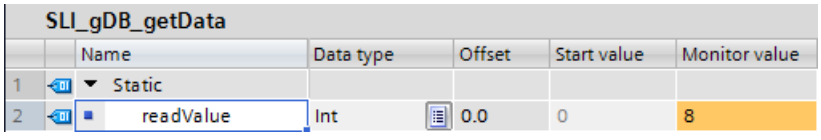

Über den Ausgangsparameter NDR ("done") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("status") mit dem Wert "16#0000" wird eine erfolgreiche Verarbeitung angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer

Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "memDoneStat". Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_GET

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	done	Bool	false	FALSE
4	memDoneStat	Word	16#0	16#0000
5	error	Bool	false	FALSE
6	status	Word	16#0	16#0000
7	memErrStatus	Word	16#0	16#0000

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

PUT: Daten in eine remote CPU schreiben

Beschreibung

Mit der Anweisung "PUT" können Sie Daten in eine remote CPU schreiben.

Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet:

- Dabei werden die Zeiger auf die zu schreibenden Bereiche (ADDR_i) und die Daten (SD_i) an die Partner-CPU gesendet. Die Partner-CPU kann sich im Betriebszustand RUN oder STOP befinden.
- Die zu sendenden Daten werden aus den projektierten Sendebereichen (SD_i) kopiert. Die Partner-CPU legt die gesendeten Daten unter den mitgeführten Adressen ab und sendet eine Ausführungsquittung zurück.
- Falls keine Fehler auftraten, wird dies beim nächsten Anweisungs-Aufruf am Zustandsparameter DONE mit "1" angezeigt. Eine erneute Aktivierung eines Schreibvorgangs ist erst nach dem Abschluss des letzten möglich.

Wenn beim Schreiben der Daten Zugriffsprobleme auftraten, oder die Prüfung der Ausführungsquittung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Voraussetzungen zur Verwendung der Anweisung

- In den Eigenschaften der Partner-CPU wurde unter "Schutz" die Funktion "Zugriff über PUT/GET-Kommunikation durch entfernten Partner erlauben" aktiviert.
- Die Bausteine, auf die Sie mit der Anweisung "PUT" zugreifen, wurden mit der Zugriffsart "standard" erstellt.

- Sie müssen darauf achten, dass die über die Parameter ADDR_i und SD_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinanderpassen.
- Der zu schreibende Bereich (Parameter ADDR_i) muss so groß sein, wie der Sendebereich (Parameter SD_i).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PUT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: – 0000H: weder Warnung noch Fehler – <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	
ADDR_1	InOut	REMOTE	E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll. Wenn der REMOTE-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren. Beispiel: P#DB10.DBX5.0 Byte 10. Bei der Übertragung von Datenstrukturen (z. B. Struct) muss an den Parametern ADDR_i der Datentyp CHAR verwendet werden.
ADDR_2	InOut	REMOTE		
ADDR_3	InOut	REMOTE		
ADDR_4	InOut	REMOTE		
SD_1	InOut	VARIANT	E, A, M, D, L	Zeiger auf diejenigen Bereiche in der eigenen CPU, die die zu versendenden Daten enthalten. Zulässig sind nur die Datentypen BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL. Bei der Übertragung von Datenstrukturen (z. B. Struct) muss an den Parametern SD_i der Datentyp CHAR verwendet werden.
SD_2	InOut	VARIANT		
SD_3	InOut	VARIANT		
SD_4	InOut	VARIANT		

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "PUT" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP). • Verbindung zum Partner noch nicht aufgebaut.
1	2	• Negative Quittung von Partner-CPU. Die Funktion ist nicht ausführbar. • Zugriff bei der Partner-CPU wurde nicht erlaubt. Aktivieren Sie den Zugriff in den CPU-Einstellungen.
1	4	Fehler in den Zeigern zur Datenablage: • Datentypen der Parameter SD_i und ADDR_i passen nicht zueinander. • Die Länge des Bereichs SD_i ist größer als die Länge der zu schreibenden Daten der Parameter ADDR_i. • Auf SD_i kann nicht zugegriffen werden. • Maximale Nutzdatengröße überschritten. • Anzahl der Parameter SD_i und ADDR_i stimmen nicht überein.
1	8	Zugriffsfehler bei der Partner-CPU (z. B. DB nicht geladen oder schreibgeschützt).
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Datenkonsistenz

Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) sind die zu sendenden Daten der Sendebereiche SD_i aus dem Anwenderprogramm kopiert. Sie können diese Bereiche nach dem Bausteinaufruf neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

Hinweis

Der gesamte Sendevorgang ist erst dann abgeschlossen, wenn der Zustandsparameter DONE den Wert "1" annimmt.

Beispiel

In dem folgenden Beispiel verwenden Sie eine S7-Verbindung und übertragen einen Datensatz von der CPU 1 zur CPU 2. Der zu übermittelnde Datensatz ist vom Datentyp INT.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Die Verbindung ist noch nicht konfiguriert.
- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.
- Der Zugriff per PUT/GET-Anweisung ist erlaubt.

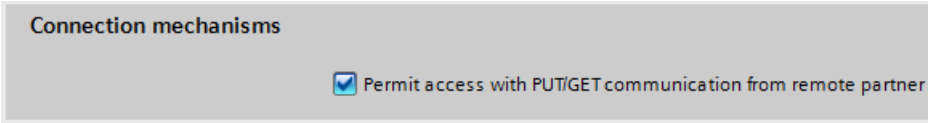

Programm der CPU 1

Zur Ablage der Daten von "PUT" legen Sie in einem globalen Datenbaustein sechs Variablen an.

SLI_gDB_PUT

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	done	Bool	false
4	memDoneStat	Word	16#0
5	error	Bool	false
6	status	Word	16#0
7	memErrStatus	Word	16#0

Zur Ablage des zu sendenden Datensatzes legen Sie einen globalen Datenbaustein mit einer Variable an.

SLI_gDB_putData

	Name	Data type	Offset	Start value
1	Static			
2	writeValue	Int	0.0	8

Netzwerk 1: Die Parameter der Anweisung "PUT" verschalten Sie wie folgt.

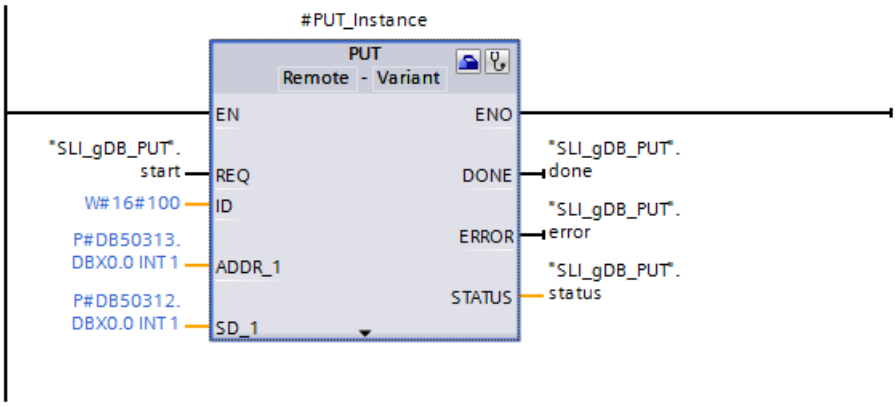

Netzwerk 2: Den Status bei einem Fehler von PUT speichern Sie wie folgt.

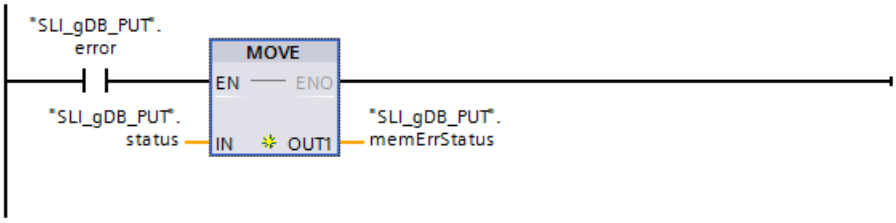

Netzwerk 3: Den Erfolgstatus von PUT speichern Sie wie folgt.

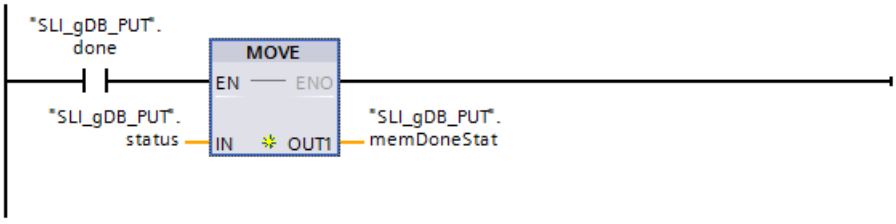

Konfiguration von PUT

Zur Verschaltung des Eingangsparameters ID öffnen Sie den Wizard der Anweisung "PUT" über "Eigenschaften > Konfiguration".

Sie nehmen folgende Einstellungen für die Verbindung vor:

Abschnitt	Einstellung
Endpunkt	Sie wählen den Kommunikationspartner aus. Die restlichen Verbindungsdaten werden automatisch eingetragen. Eine S7-Verbindung wird automatisch angelegt und deren Kennung am Eingangsparameter ID eingetragen.
Aktiv	Sie stellen sicher, dass die aktive Verbindung von der CPU 1 ausgeht.

Programm der CPU 2

Zur Ablage des zu empfangenden Datensatzes legen Sie einen globalen Datenbaustein mit einer Variable an.

SLI_gDB_rcvData_PUT

	Name	Data type	Offset	Start value
1	Static			
2	myValue	Int	0.0	0

Unter "Eigenschaften > Attribute" des Datenbausteins deaktivieren Sie den optimierten Bausteinzugriff.

Attributes

- [ ] Only store in load memory
- [ ] Data block write-protected in the device
- [ ] Optimized block access

Verhalten von PUT

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "PUT" gestartet. Die Anweisung ruft die Verbindungsdaten ab und nimmt Kontakt auf zum Kommunikationspartner der verwendeten S7-Verbindung. Dazu ist am Eingangsparameter ID die Kennung der S7-Verbindung hinterlegt.

Am Parameter SD_1 ist der Ablageort des zu sendenden Datensatzes ("writeValue") der CPU 1 erfasst. Am Parameter ADDR_1 ist der Ablageort ("myValue") des Datensatzes in der CPU 2 erfasst. Der Datensatz ("writeValue") wird übertragen und am Ablageort der CPU 2 ("myValue") angezeigt.

SLI_gDB_rcvData_PUT

	Name	Data type	Offset	Start value	Monitor value
1	Static				
2	myValue	Int	0.0	0	8

Über den Ausgangsparameter DONE ("done") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("status") mit dem Wert "16#0000" wird eine erfolgreiche Verarbeitung angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "memDoneStat". Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung ohne Fehler verlief.

SLI_gDB_PUT

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	done	Bool	false	FALSE
4	memDoneStat	Word	16#0	16#0000
5	error	Bool	false	FALSE
6	status	Word	16#0	16#0000
7	memErrStatus	Word	16#0	16#0000

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Weitere

USEND: Daten unkoordiniert senden

Beschreibung

Die Anweisung "USEND" sendet Daten an eine remote Partner-Anweisung vom Typ "URCV (Seite 3608)". Der Sendevorgang verläuft ohne Koordination mit der Partner-Anweisung, d. h. die Datenübertragung erfolgt ohne Quittierung durch die Partner-Anweisung.

Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) werden die zu sendenden Daten der Sendebereiche SD_i aus dem Anwenderprogramm kopiert. Sie können diese Sendebereiche nach dem Aufruf der Anweisung neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit "1" angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	CONN_PRG	E, A, M, D, L, P oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
R_ID	Input	CONN_R_ID	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID zur Festlegung der Anweisungs-Paare "USEND" und "URCV". Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler. • 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter ERROR und STATUS".
SD_i (1≤ i ≤4)	InOut	VARIANT	E, A, M, D, L	Zeiger auf i-ten Sendebereich. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL oder STRUCT. Die maximale Nutzdatengröße für die Parameter SD_i ist abhängig von der Partner-CPU (Anweisung "URCV") sowie der Anzahl der verwendeten Parameter. Weitere Informationen finden Sie unter: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsproblem aufgetreten. Mögliche Gründe: • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut
1	4	• Fehler in den Sendebereichszeigern SD_i bezüglich der Datenlänge oder des Datentyps. • Maximale Nutzdatenlänge wurde überschritten.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	18	Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für USEND & URCV (Seite 3610).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

URCV: Daten unkoordiniert empfangen

Beschreibung

Die Anweisung "URCV" empfängt asynchron Daten von einer remoten Partner-Anweisung vom Typ "USEND (Seite 3606)" und kopiert sie in die projektierten Empfangsbereiche.

Die Anweisung ist empfangsbereit, wenn am Eingang EN_R eine logische "1" anliegt. Mit EN_R=0 kann ein laufender Auftrag abgebrochen werden.

Die Empfangsdatenbereiche werden durch die Parameter RD_1, ... RD_4 referenziert. Sie müssen beachten, dass die über die Parameter RD_i / RD_1 und SD_i / SD_1 (bei der zugehörigen Partner-Anweisung "USEND (Seite 3606)") definierten Bereiche übereinstimmen in Anzahl und Länge.

Der erfolgreiche Abschluss des Kopiervorgangs wird am Zustandsparameter NDR mit einer logischen "1" angezeigt. Nachdem der Zustandsparameter NDR den Wert "1" angenommen hat, befinden sich neue Empfangsdaten in Ihren Empfangsbereichen (RD_i). Ein erneuter Bausteinaufruf kann diese Daten mit neuen Empfangsdaten überschreiben. Wenn Sie dies verhindern wollen, müssen Sie "URCV" mit dem Wert "0" an EN_R so lange aufrufen (z. B. bei zyklischer Bausteinbearbeitung), bis Sie die Bearbeitung der Empfangsdaten beendet haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "URCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	CONN_PRG	E, A, M, D, L, P oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
R_ID	Input	CONN_R_ID	E, A, M, D, L oder Konstante	Adressierungsparameter zur Festlegung der Anweisungs-Paare "USEND" und "URCV". Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler • 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter ERROR und STATUS".
RD_i (1≤ i ≤4)	InOut	VARIANT	E, A, M, D, L	Zeiger auf den i-ten Empfangsbereich: Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL oder STRUCT. Weitere Informationen finden Sie unter: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	9	Warnung: Ältere Empfangsdaten wurden von neueren Empfangsdaten überschrieben.
0	11	Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet (Fehler kann beim Kopieren der Daten in den Empfangsbereich auftreten).
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsproblem aufgetreten. Mögliche Gründe: • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut
1	4	Fehler in den Empfangsbereichszeigern RD_i bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	18	Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).
1	19	Die zugehörige Anweisung "USEND (Seite 3606)" sendet schneller Daten, als diese von "URCV" in die Empfangsbereiche kopiert werden können.
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für USEND & URCV (Seite 3610).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für USEND & URCV

Einleitung

In dem folgenden Beispiel nutzen Sie eine S7-Verbindung zwischen zwei CPUs der Baureihe S7-1500. Sie senden eine Zeichenkette (Datentyp STRING) von CPU 1 zu CPU 2. Die Datenübertragung erfolgt ohne Quittierung durch den Kommunikationspartner.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Eine S7-Verbindung ist konfiguriert.

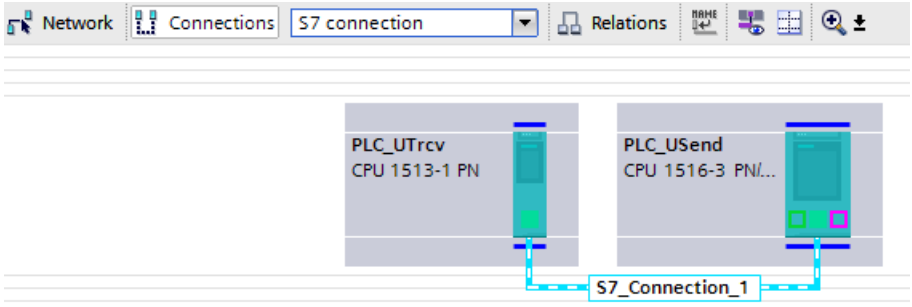

- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Programm der CPU 1

Zur Ablage der Daten von USEND legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_USEND

		Name	Data type	Start value
1		Static		
2		start	Bool	false
3		connectionID	CONN_PRG	16#0100
4		instructionPair	CONN_R_ID	16#01
5		done	Bool	false
6		error	Bool	false
7		status	Word	16#0
8		memErrStatus	Word	16#0
9		sendData	String	'HelloData'

Sie legen einen FB "SLI_FB_USEND" an. Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	USEND_Instance	USEND		
9	done	Bool	false	Non-ret...

Netzwerk 1: Die Parameter der Anweisung "USEND" verschalten Sie wie folgt.

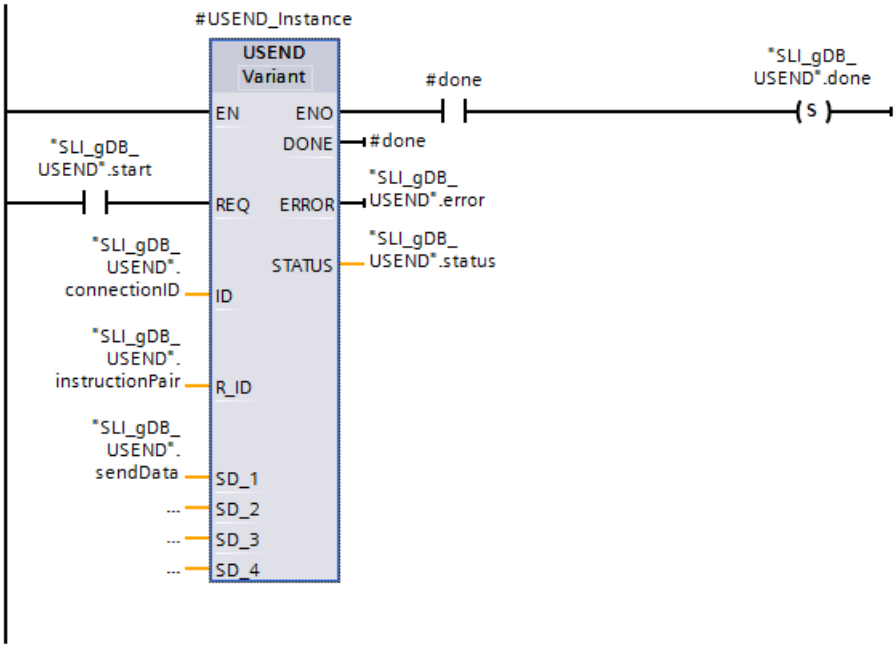

Netzwerk 2: Den Status im Fehlerfall von USEND speichern Sie wie folgt.

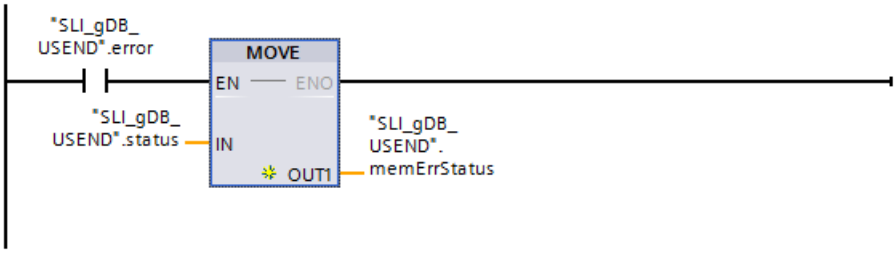

Programm der CPU 2

Zur Ablage der Daten von URCV legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_URCV			
	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	connectionID	CONN_PRG	16#0100
4	instructionPair	CONN_R_ID	16#01
5	done	Bool	false
6	error	Bool	false
7	status	Word	16#0
8	memErrStatus	Word	16#0
9	rcvData	String	''

Sie legen einen FB "SLI_FB_URCV" an. Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	URCV_Instance	URCV		
9	done	Bool	false	Non-ret...

Netzwerk 1: Die Parameter der Anweisung "URCV" verschalten Sie wie folgt.

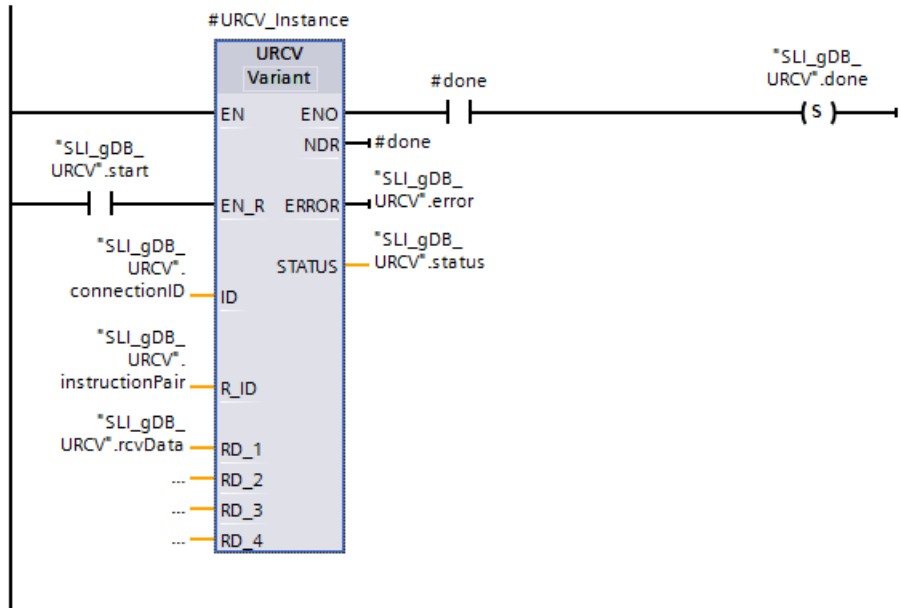

Netzwerk 2: Den Status im Fehlerfall von URCV speichern Sie wie folgt.

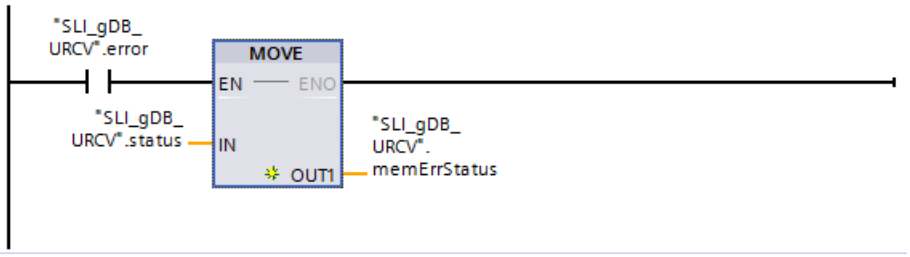

Kommunikationsverbindung zuweisen

Die Adressierungsparameter für die Verbindung der beiden CPUs und zur Festlegung des Anweisungs-Paars müssen einheitlich sein.

- Hinterlegen Sie jeweils am Eingangsparameter ID ("connectionID") den hexadezimalen Wert der Hardware-Kennung der konfigurierten S7-Verbindung.
  Die Hardware-Kennung finden Sie in der "Netzsicht" unter "Verbindungen".

Network overview | Connections | Relations | I/O communication | VPN

Local connection n...	Local end point	Local ID (hex)	Partner ID (hex)	Partner	Connection type
S7_Connection_1	PLC_UTrcv	100	100	PLC_USend	S7 connection
S7_Connection_1	PLC_USend	100	100	PLC_UTrcv	S7 connection

- Hinterlegen Sie jeweils am Eingangsparameter R_ID ("instructionPair") eine selbstgewählte Kennung (als Hexwert, in Byte-Größe) für das Anweisungs-Paar. Die Kennung darf nicht bereits für ein anderes Anweisungs-Paar vergeben sein.

Verhalten der CPU 1

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "USEND" gestartet. Dadurch kopiert die Anweisung "USEND" den am Eingangsparameter SD_1 ("sendData") erfassten Datensatz in den Arbeitsspeicher der CPU. Über den Ausgangsparameter DONE ("#done") mit dem Signalzustand "TRUE" wird das erfolgreiche Kopieren des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "done".

Entsprechend der Eingangsparameter ID ("connectionID") und R_ID ("instructionPair") ist die Anweisung "URCV" der CPU 2 als Empfänger des Datensatzes ("sendData" mit dem Wert "HelloData") addressiert. Die Fortsetzung der Datenübertragung über die S7-Verbindung erfolgt ohne weitere Beteiligung von USEND ("start" darf auf "FALSE" gesetzt sein).

Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung von USEND im Beispiel ohne Fehler verläuft.

SLI_gDB_USEND				
	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	connectionID	CONN_PRG	16#0100	16#0100
4	instructionPair	CONN_R_ID	16#01	16#0000_0001
5	done	Bool	false	TRUE
6	error	Bool	false	FALSE
7	status	Word	16#0	16#0000
8	memErrStatus	Word	16#0	16#0000
9	sendData	String	'HelloData'	'HelloData'

Verhalten der CPU 2

Die CPU 2 empfängt über die S7-Verbindung einen Datensatz der CPU 1. Entsprechend der Eingangsparameter ID ("connectionID") und R_ID ("instructionPair") ist der Anweisung "URCV" der Empfänger des Datensatzes bekannt.

Wenn der Eingangsparameter EN_R ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "URCV" gestartet und ist empfangsbereit. Die Anweisung "URCV" liesst den übermittelten Datensatz aus und kopiert den Datensatz in den Wertebereich am Eingangsparameter RD_1 ("rcvData").

Über den Ausgangsparameter NDR ("#done") mit dem Signalzustand "TRUE" wird das erfolgreiche Kopieren des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "done".

Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung von URCV im Beispiel ohne Fehler verläuft.

SLI_gDB_URCV				
	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	connectionID	CONN_PRG	16#0100	16#0100
4	instructionPair	CONN_R_ID	16#01	16#0000_0001
5	done	Bool	false	TRUE
6	error	Bool	false	FALSE
7	status	Word	16#0	16#0000
8	memErrStatus	Word	16#0	16#0000
9	rcvData	String	''	'HelloData'

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

USEND: Daten unkoordiniert senden (Seite 3606)

URCV: Daten unkoordiniert empfangen (Seite 3608)

BSEND: Daten in Segmenten senden

Beschreibung

Die Anweisung "BSEND" sendet Daten an eine remote Partner-Anweisung vom Typ "BRCV (Seite 3617)". Bei diesem Datentransfer kann eine größere Datenmenge zwischen den Kommunikationspartnern transportiert werden, als dies mit allen anderen Kommunikations-Anweisungen für projektierte S7-Verbindungen möglich ist. Die maximale Datenmenge beträgt jeweils 65534 Byte (Standard-Zugriff) bzw. 65535 Byte (optimierter Zugriff) bei der integrierten Schnittstelle als auch bei SIMATIC-Net-CP.

Arbeitsweise

Das Anweisungspaar "BSEND" und "BRCV" legen Sie über den Parameter R_ID fest. Der Parameter R_ID muss bei den zusammengehörenden Anweisungen identisch sein.

Die Aktivierung des Sendevorgangs erfolgt nach Aufruf der Anweisung und positiver Flanke am Steuereingang REQ. Die Bearbeitung von "BSEND" erfolgt nach dem Aufruf nicht im Hintergrund, d.h. die Daten können nur innerhalb des Anwenderprogramms gelesen werden.

Der zu sendende Datenbereich wird segmentiert. Jedes Segment wird einzeln an den Partner gesendet. Jedes Segment wird vom Partner nach der Übernahme dieses Segments durch "BRCV (Seite 3617)" quittiert. Bei einer Segmentierung der Daten muss "BSEND" mehrfach aufgerufen werden, bis alle Segmente übertragen worden sind.

Der Datenbereich der zu sendenden Daten wird durch SD_1 vorgegeben. Um Datenkonsistenz zu gewährleisten, dürfen Sie den aktuell benutzten Teil des Sendebereichs SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

Die Länge der Sendedaten legen Sie auftragsbezogen durch LEN fest. Bei LEN = "0" werden alle Daten gesendet, die über den Parameter SD_1 adressiert sind.

Bei einer positiven Flanke am Steuereingang R wird ein laufender Sendevorgang abgebrochen.

Aufgrund der asynchronen Datenübertragung kann ein erneutes Senden von Daten erst gestartet werden, wenn die vorhergehenden Daten durch Aufruf der Partner-Anweisung abgeholt wurden. Wenn die Daten abgeholt wurden, wird in der Partner-Anweisung "BRCV" der Zustandsparameter "NDR" gesetzt.

Hinweis

Migration von S7-400 Anwenderprogrammen

Eine S7-400 CPU interpretiert den Parameter SD_1 als Zeiger, nicht als Datenbereich.

Bei der S7-1500 darf LEN den Bereich von SD1 nicht überschreiten. Bei der S7-400 war dies zulässig. Empfehlung: Verwenden Sie als Größe des Datenbereichs am Parameter SD_1 die maximale Größe für den Parameter LEN (65534 Byte bei der integrierten Schnittstelle).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke
R	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter reset, aktiviert Abbruch eines noch laufenden Datenaustauschs bei steigender Flanke
ID	Input	CONN_PRG	E, A, M, D, L, P oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
R_ID	Input	CONN_R_ID	E, A, M, D, L oder Konstante	Adressierungsparameter zur Festlegung der Anweisungs-Paare "BSEND" und "BRCV (Seite 3617)". Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)
SD_1	InOut	VARIANT	E, A, M, D, L	Zeiger auf Sendebereich Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
LEN	InOut	WORD	E, A, M, D, L	Länge des zu sendenden Datenblocks in Byte. Bei LEN = "0" werden alle Daten von SD_1 gesendet.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler. • 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter ERROR und STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BSEND" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsproblem aufgetreten. Mögliche Gründe: • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut

ERROR	STATUS (dezimal)	Erläuterung
1	2	Negative Quittung von Partner-Anweisung. Die Anweisung ist nicht ausführbar.
1	3	R_ID ist auf der durch ID vorgegebenen Kommunikationsverbindung nicht bekannt, oder der Empfangsbaustein wurde noch nie aufgerufen.
1	4	• Fehler im Sendebereichszeiger SD_1 bezüglich der Datenlänge oder des Datentyps. • Wert für LEN größer als der Bereich SD_1.
1	5	Resetanforderung wurde ausgeführt.
1	6	Partner-Anweisung befindet sich im Zustand DISABLED (EN_R hat den Wert "0"). Überprüfen Sie zusätzlich die Eingangsparameter von "BRCV (Seite 3617)" auf Konsistenz zu "BSEND".
1	7	Partner-Anweisung "BRCV (Seite 3617)" wurde nach der letzten Datenübertragung nicht mehr aufgerufen.
1	8	Zugriff auf remotes Objekt im Anwenderspeicher wurde abgelehnt: Der Zielbereich beim zugehörigen "BRCV (Seite 3617)" ist zu klein. An den Ausgangsparametern von "BRCV (Seite 3617)" wird ERROR = 1, STATUS = 4 oder ERROR = 1, STATUS = 10 gemeldet.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	18	R_ID existiert bereits in der Verbindung.
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für BSEND & BRCV (Seite 3619).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

BRCV: Daten in Segmenten empfangen

Beschreibung

Die Anweisung "BRCV" empfängt Daten von einer remoten Partner-Anweisung vom Typ "BSEND (Seite 3615)". Der Parameter R_ID muss bei den zusammengehörenden Anweisungen identisch sein.

Die Anweisung ist nach Aufruf mit dem Wert "1" am Steuereingang EN_R empfangsbereit (STATUS = 25). Mit EN_R=0 kann ein laufender Auftrag abgebrochen werden.

Der maximale Empfangsbereich wird durch RD_1 vorgegeben. Die Daten werden konsistent empfangen, wenn Sie den aktuell benutzten Teil des Empfangsbereichs RD_1 vollständig auswerten, bevor Sie den Baustein wieder mit dem Wert 1 am Steuereingang EN_R aufrufen.

Nach jedem empfangenen Datensegment wird eine Quittung an die Partner-Anweisung geschickt. Bei mehreren Segmenten ist es nötig, "BRCV" mehrfach aufzurufen, bis alle Segmente empfangen worden sind. Der asynchrone Datenempfang wird durch STATUS = 17 angezeigt. Die Anzahl der derzeit empfangenen Daten wird am Parameter LEN angezeigt. Der Parameter RD_1 muss während der Operation konstant bleiben.

Der fehlerfreie Empfang sämtlicher Datensegmente wird am Zustandsparameter NDR mit dem Wert "1" angezeigt. Die empfangenen Daten bleiben unverändert bis zum nächsten Aufruf mit EN_R=1.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BRCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	CONN_PRG	E, A, M, D, L, P oder Konstante	Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.
R_ID	Input	CONN_R_ID	E, A, M, D, L oder Konstante	Adressierungsparameter zur Festlegung der Anweisungs-Paare "BSEND (Seite 3615)" und "BRCV". Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 3591)
RD_1	InOut	VARIANT	E, A, M, D, L	Zeiger auf Empfangsbereich. Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
LEN	InOut	WORD	E, A, M, D, L	Länge der bisher empfangenen Daten in Byte.
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter • 0: Weder Warnung noch Fehler • 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter Siehe Tabelle "Parameter ERROR und STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BRCV" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	17	Warnung: Anweisung empfängt asynchron Daten. Der Parameter LEN zeigt die Anzahl der bisher empfangenen Daten in Byte an.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsproblem aufgetreten. Mögliche Gründe: • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut
1	2	Funktion nicht ausführbar (Protokollfehler)
1	4	Fehler im Empfangsbereichszeiger RD_1 bezüglich der Datenlänge oder des Datentyps. Der gesendete Datenblock ist länger als der Empfangsbereich.
1	5	Resetanforderung eingetroffen, unvollständige Übertragung.
1	8	Zugriffsfehler beim zugehörigen "BSEND (Seite 3615)": Nach dem Versenden des letzten gültigen Datensegments wird ERROR = 1 und STATUS = 4 oder ERROR = 1 und STATUS = 10 gemeldet.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	18	Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).
1	20	• Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).
1	W#16#80C3	(nur bei S7-1500) • Maximale Anzahl an parallelen Aufträgen überschritten. • Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für BSEND & BRCV (Seite 3619).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für BSEND & BRCV

Einleitung

In dem folgenden Beispiel nutzen Sie eine S7-Verbindung zwischen zwei CPUs der Baureihe S7-1500. Sie senden mit Hilfe von BSEND und BRCV einen Datensatz von CPU 1 zur CPU 2. Die Datenübertragung erfolgt mit Quittierung durch BRCV.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Eine S7-Verbindung ist konfiguriert.

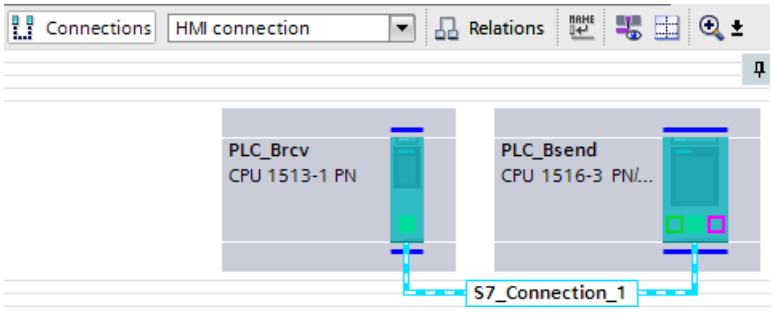

- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Programm der CPU 1

Zur Ablage der Daten von BSEND legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_BSEND

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	reset	Bool	false
4	connectionID	CONN_PRG	16#0100
5	instructionPair	CONN_R_ID	16#02
6	maxLength	UInt	0
7	done	Bool	false
8	error	Bool	false
9	status	Word	16#0
10	memErrStatus	Word	16#0

Für den Datensatz legen Sie den folgenden PLC-Datentyp an.

BSEND_User

	Name	Data type	Default value
1	prename	String	''
2	lastname	String	''
3	userId	UInt	0

Für die Datenübertragung legen Sie auf Basis des erstellten PLC-Datentyps "BSEND_User" den folgenden Datenbaustein an.

SLI_plcDB_sendData_BSEND

	Name	Data type	Start value
1	Static		
2	prename	String	'Simon'
3	lastname	String	'Sample'
4	userId	UInt	1

Sie legen einen FB "SLI_FB_BSEND" an. Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	done	Bool	false	Non-retain
9	BSEND_Instance	BSEND		

Netzwerk 1: Die Parameter der Anweisung "BSEND" verschalten Sie wie folgt.

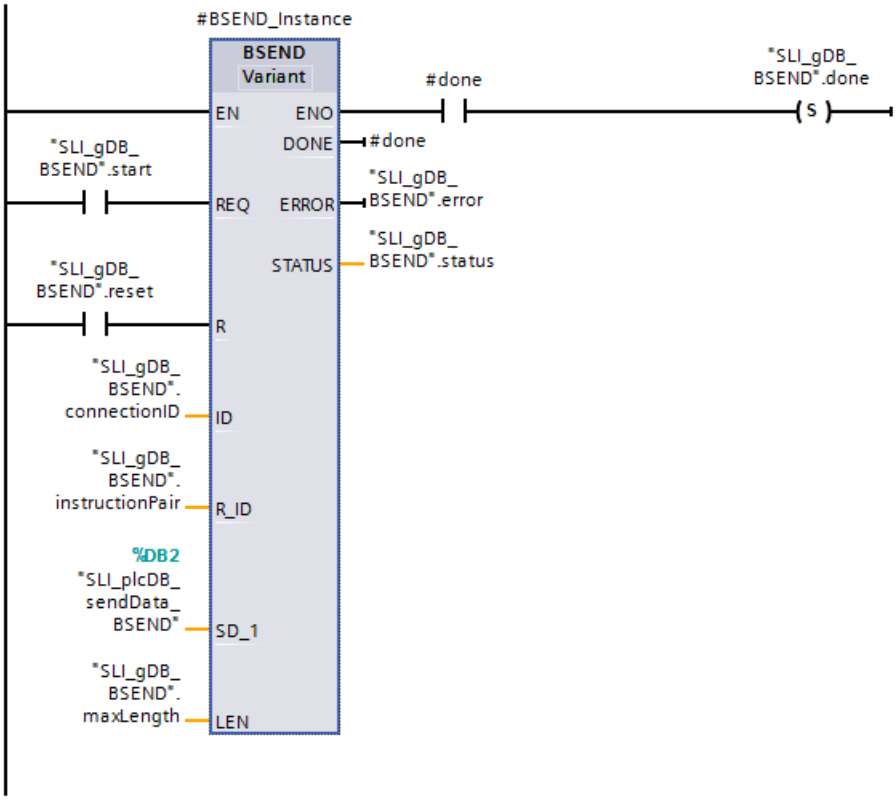

Netzwerk 2: Den Status im Fehlerfall von BSEND speichern Sie wie folgt.

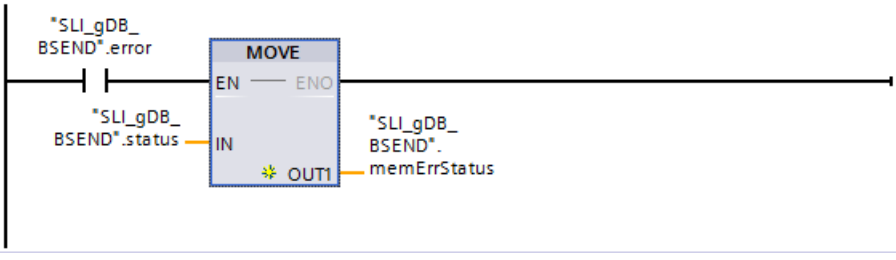

Programm der CPU 2

Zur Ablage der Daten von BRCV legen Sie in einem globalen Datenbaustein die folgenden Variablen an.

SLI_gDB_BRCV

	Name	Data type	Start value
1	Static		
2	start	Bool	false
3	connectionID	CONN_PRG	16#0100
4	instructionPair	CONN_R_ID	16#02
5	done	Bool	false
6	error	Bool	false
7	status	Word	16#0
8	memErrStatus	Word	16#0
9	readLength	UInt	0

Für den Datensatz legen Sie den folgenden PLC-Datentyp an.

BSEND_User

	Name	Data type	Default value
1	prename	String	''
2	lastname	String	''
3	userId	UInt	0

Für den Datenempfang legen Sie auf Basis des erstellten PLC-Datentyps "BSEND_User" den folgenden Datenbaustein an.

SLI_plcDB_rcvData_BRCV

	Name	Data type	Start value
1	Static		
2	prename	String	''
3	lastname	String	''
4	userId	UInt	0

Sie legen einen FB "SLI_FB_BSEND" an. Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	done	Bool	false	Non-ret...
9	length	UInt	0	Non-retain
10	BRCV_Instance	BRCV		

Netzwerk 1: Die Parameter der Anweisung "BRCV" verschalten Sie wie folgt.

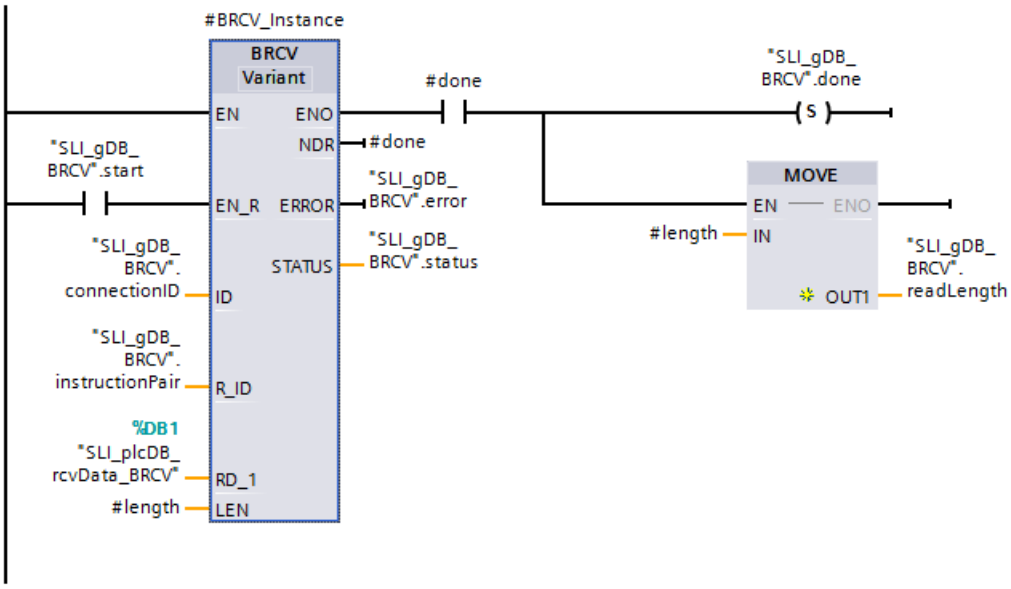

Netzwerk 2: Den Status im Fehlerfall von BRCV speichern Sie wie folgt.

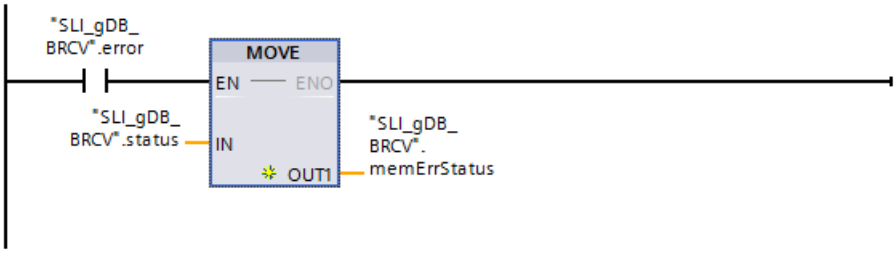

Kommunikationsverbindung zuweisen

Die Adressierungsparameter für die Verbindung der beiden CPUs und zur Festlegung des Anweisungs-Paars müssen einheitlich sein.

- Hinterlegen Sie jeweils am Eingangsparameter ID ("connectionID") den hexadezimalen Wert der Hardware-Kennung der konfigurierten S7-Verbindung.
  Die Hardware-Kennung finden Sie in der "Netzsicht" unter "Verbindungen".

Network overview | Connections | Relations | I/O communication

Local connection name	Local end point	Local ID (hex)	Partner ID (hex)
S7_Connection_1	PLC_Bsend	100	100
S7_Connection_1	PLC_Brcv	100	100

- Hinterlegen Sie jeweils am Eingangsparameter R_ID ("instructionPair") eine selbstgewählte Kennung (als Hexwert, in Byte-Größe) für das Anweisungs-Paar. Die Kennung darf nicht bereits für ein anderes Anweisungs-Paar vergeben sein.

Verhalten der CPU 1

Entsprechend der Eingangsparameter ID ("connectionID") und R_ID ("instructionPair") ist die Anweisung "BRCV" der CPU 2 als Empfänger des Datensatzes ("SLI_plcDB_sendData_BSEND") addressiert.

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "BSEND" gestartet. Über mehrere Aufrufe sendet die Anweisung "BSEND" den am Eingangsparameter SD_1 ("SLI_plcDB_sendData_BSEND") erfassten Datensatz in Segmenten an die CPU 2. Entsprechend dem Wert "0" des Eingangsparameter LEN ("maxLength") ist die Länge des Datensatzes unbeschränkt.

Über den Ausgangsparameter DONE ("#done") mit dem Signalzustand "TRUE" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "done".

Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_BSEND

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	reset	Bool	false	FALSE
4	connectionID	CONN_PRG	16#0100	16#0100
5	instructionPair	CONN_R_ID	16#02	16#0000_0002
6	maxLength	UInt	0	0
7	done	Bool	false	TRUE
8	error	Bool	false	FALSE
9	status	Word	16#0	16#0000
10	memErrStatus	Word	16#0	16#0000

Ein erneutes Senden von Daten kann erst gestartet werden, wenn die vorhergehenden Daten durch Aufruf von BRCV abgeholt wurden.

Verhalten der CPU 2

Die CPU 2 empfängt über die S7-Verbindung einen Datensatz (in Segmenten) der CPU 1. Entsprechend der Eingangsparameter ID ("connectionID") und R_ID ("instructionPair") ist der Anweisung "BRCV" der Empfänger des Datensatzes bekannt.

Wenn der Eingangsparameter EN_R ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "BRCV" gestartet und ist empfangsbereit. Über mehrere Aufrufe empfängt die Anweisung "BRCV" den Datensatz in Segmenten und speichert den Datensatz am Eingangsparameter RD_1 ("SLI_plcDB_rcvData_BRCV") ab.

SLI_plcDB_rcvData_BRCV

	Name	Data type	Start value	Monitor value
1	Static			
2	prename	String	''	'Simon'
3	lastname	String	''	'Sample'
4	userId	UInt	0	1

Über den Ausgangsparameter NDR ("#done") mit dem Signalzustand "TRUE" wird das erfolgreiche Empfangen des gesamten Datensatzes angezeigt. Die Länge in BYTE des tatsächlich übermittelten Datensatzes wird über den Ausgangsparameter LEN ("#length") erfasst. Dessen Wert wird nur während des Erfolgsstatus angezeigt. Danach wird "0" erfasst. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus von #done in der Variable "done" und die Länge in BYTE in der Variable "readLength".

Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_BRCV

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	connectionID	CONN_PRG	16#0100	16#0100
4	instructionPair	CONN_R_ID	16#02	16#0000_0002
5	done	Bool	false	TRUE
6	error	Bool	false	FALSE
7	status	Word	16#0	16#0019
8	memErrStatus	Word	16#0	16#0000
9	readLength	UInt	0	514

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

BSEND: Daten in Segmenten senden (Seite 3615)

BRCV: Daten in Segmenten empfangen (Seite 3617)

4.1.6.2 Open User Communication

Datenkonsistenz

Sicherstellung der Datenkonsistenz

In der Baugruppenfamilie S7-1500 ist es bei den Anweisungen der Open User Communication möglich, Parameter während des laufenden Betriebs zu ändern. Geänderte Parameter werden sofort verwendet, auch wenn noch ein Auftrag aktiv ist. Beides kann zu Dateninkonsistenz führen! Um Dateninkonsistenzen zu vermeiden, sollten Sie auf keinen Fall während eines laufenden Auftrags Parameter verändern.

TSEND_C: Verbindung aufbauen und Daten senden

TSEND_C: Daten über Ethernet senden

Gültigkeit

Die folgende Beschreibung der Anweisung "TSEND_C" gilt für S7-1200-CPUs mit Firmware-Version < V4.0.

Beschreibung

Die Anweisung "TSEND_C" richtet eine TCP- oder ISO-on-TCP-Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet.

Die Anweisung wird asynchron ausgeführt und hat die folgenden Funktionen:

- Eine Kommunikationsverbindung einrichten und aufbauen:
  Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE für einen Zyklus auf "1" gesetzt. Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TSEND_C" erneut ausgeführt werden. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.
- Daten über die bestehende Kommunikationsverbindung senden:
  Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Verwenden Sie am Parameter DATA keinen Datenbereich mit dem Datentyp BOOL oder Array of BOOL. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.
- Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird. Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden. Beim Absenden von Daten (steigende Flanke am Parameter REQ) muss der Parameter CONT den Wert "1" haben, um eine Verbindung aufzubauen oder zu halten. Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits gelesen wurden.
- Die Kommunikationsverbindung abbauen:
  Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie für "TSEND_C" eine bereits projektierte Verbindung verwenden.

Beim Setzen des Parameters COM_RST auf "1" kann der laufende Verbindungsaufbau oder eine laufende Datenübertragung jederzeit zurücksetzen werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut.

Wenn zum Zeitpunkt der Neuausführung Daten übertragen werden, kann es zum Datenverlust kommen.

Um "TSEND_C" nach der Ausführung (DONE = 1) wieder freizugeben, rufen Sie die Anweisung einmal mit REQ = 0 auf.

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP oder ISO-on-TCP anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND_C":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Startet den Sendeauftrag bei einer steigenden Flanke.
CONT	Input	BOOL	E, A, M, D, L	Steuert die Kommunikationsverbindung: • 0: Kommunikationsverbindung trennen • 1: Kommunikationsverbindung aufbauen und halten Beim Absenden von Daten (steigende Flanke am Parameter REQ) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten.
LEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.
CONNECT	InOut	TCON_Param	D	Zeiger auf die Verbindungsbeschreibung Siehe auch: AUTOHOTSPOT
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält (maximale Länge: 8192 Bytes). Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
COM_RST	InOut	BOOL	E, A, M, D, L	Bewirkt einen Neustart der Anweisung: • 0: Irrelevant • 1: Vollständiger Neustart der Anweisung, wodurch die vorhandene Verbindung abgebaut und eine neue Verbindung aufgebaut wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Beschreibung
0	0000	Auftrag fehlerfrei ausgeführt.
0	0001	Der Verbindungsaufbau ist abgeschlossen.
0	0003	Der Verbindungsabbau ist abgeschlossen.
0	7000	Keine Auftragsbearbeitung aktiv.

ERROR	STATUS* (W#16#...)	Beschreibung
0	7001	• Auftragsbearbeitung starten • Verbindung aufbauen • Auf Verbindungspartner warten
0	7002	Daten wurden gesendet.
0	7003	Verbindung wird abgebaut.
0	7004	Verbindung aufgebaut und überwacht, keine Auftragsverarbeitung aktiv.
1	80A0	Sammelfehler für Fehlercodes 80A1 und 80A2.
1	80A1	• Verbindung oder Port wird vom Anwender bereits verwendet. • Kommunikationsfehler: – Die angegebene Verbindung wurde noch nicht aufgebaut. – Die angegebene Verbindung wird gerade beendet. Eine Übertragung über diese Verbindung ist nicht möglich. – Die Schnittstelle wird neu initialisiert.
1	80A2	Lokaler oder remoter Port wird vom System verwendet.
1	80A3	Es wird versucht, eine nicht vorhandene Verbindung zu beenden.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.
1	80A7	Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.
1	80AA	Es läuft gerade ein Verbindungsaufbau mit der gleichen Verbindungs-ID durch einen anderen Baustein. Bitte wiederholen Sie den Auftrag durch einen erneuten Flankenanstieg am Parameter REQ.
1	80B2	Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.
1	80B3	Inkonsistente Parametrierung: Sammelfehler für Fehlercodes 80A0 bis 80A2, 80A4, 80B4 bis 80B9.
1	80B4	Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.
1	80B6	Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.
1	80B7	Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	8085	Parameter LEN ist größer als der größte zulässige Wert.
1	8086	Der Parameter ID innerhalb des Parameters CONNECT liegt außerhalb des zulässigen Bereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich.
1	8088	Der Wert am Parameter LEN passt nicht zu den am Parameter DATA angegebene Empfangsbereich.
1	8089	Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.
1	8091	Maximale Schachtelungstiefe überschritten.
1	809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.

ERROR	STATUS* (W#16#...)	Beschreibung
1	809B	Die ID des lokalen Geräts in der Verbindungsbeschreibung entspricht nicht der CPU.
1	80C3	• Alle Verbindungsressourcen sind belegt. • Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt. • Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST= 1 beendet
1	80C6	Remoter Netzwerkfehler. Remoter Partner kann nicht erreicht werden.
1	8722	Parameter CONNECT: Der Quellbereich ist ungültig. Der Bereich ist im DB nicht vorhanden.
1	873A	Parameter CONNECT: Zugriff auf Verbindungsbeschreibung nicht möglich (z.B. DB nicht verfügbar).
1	877F	Parameter CONNECT: Interner Fehler.
1	8822	Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden.
1	8824	Parameter DATA: Bereichsfehler im VARIANT-Pointer.
1	8832	Parameter DATA: DB-Nummer ist zu groß.
1	883A	Parameter DATA: Kein Zugriff auf den Datenbereich, da z. B. der Datenbaustein nicht existiert
1	887F	Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.
1	893A	Parameter ADDR: Zugriff auf den Sendebereich nicht möglich (z.B. weil der DB nicht vorhanden ist).
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Hinweis

Fehlermeldungen der Anweisungen "TCON", "TSEND", "T_DIAG" und "TDISCON"

Die Anweisung "TSEND_C" verwendet intern die Anweisungen "TCON (Seite 3689)", "TSEND (Seite 3705)", "T_DIAG (Seite 3758)", "T_RESET (Seite 3753)" und "TDISCON (Seite 3700)". Die Fehlermeldungen dieser Anweisungen können ebenfalls am Parameter STATUS ausgegeben werden. Die Bedeutung der Fehlercodes ist bei den entsprechenden Anweisungen beschrieben. Bei gleichen Fehlercodes intern verwendeter Anweisungen mit unterschiedlicher Bedeutung kann über den Instanzdatenbaustein von "TSEND_C" ermittelt werden, durch welche Anweisung der Fehler ausgegeben wurde.

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TSEND_C: Verbindung aufbauen und Daten senden

Gültigkeit

Die folgende Beschreibung der Anweisung "TSEND_C" ist gültig für:

- Ethernet
  - CPU S7-1200 mit Firmware-Stand ≥ V4.0 und CPU S7-1500
  - CPU S7-1500 ab Firmware-Stand V2.1: UDP-Multicast-Kommunikation
  - CPU S7-1500 ab Firmware-Stand V2.0 und TSEND_C ab Anweisungs-Version V3.2: Secure Communication
- PROFIBUS
  FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL

Beschreibung

Die Anweisung "TSEND_C" richtet eine Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht.

Die Anweisung wird asynchron ausgeführt und hat die folgenden Funktionen:

- Eine Kommunikationsverbindung einrichten und aufbauen
- Daten über die bestehende Kommunikationsverbindung senden
- Die Kommunikationsverbindung abbauen oder zurücksetzen

Die Anweisung "TSEND_C" verwendet dabei intern die Kommunikations-Anweisungen "TCON", "TSEND", "T_DIAG", "T_RESET" und "TDISCON".

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP, ISO-on-TCP oder FDL anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen. UDP-Multicast-Verbindungen über die integrierten PROFINET-Schnittstellen sind bei S7-1500-CPUs ab Firmware-Stand V2.1 möglich.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Kommunikationsverbindung einrichten und aufbauen

Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Die folgenden Verbindungstypen können verwendet werden:

- Programmierte Verbindungen (Aufbau der Verbindung über "TCON"):
  - TCP / UDP: Verbindungsbeschreibung über den Systemdatentyp TCON_IP_v4
  - TCP / UDP mittels secure communication: Verbindungsbeschreibung über die Systemdatentypen TCON_IP_V4_SEC oder TCON_QDN_SEC
  - ISO-on-TCP: Verbindungsbeschreibung über den Systemdatentyp TCON_IP_RFC
  - ISO: Verbindungsbeschreibung über den Systemdatentyp TCON_ISOnative (nur bei CP 1543-1)
  - Telecontrol-Verbindungen zu SMS-Clients: Verbindungsbeschreibung über den Systemdatentyp TCON_PHONE
    Für diesen Verbindungstyp benötigt die Station Zugang zum Mobilfunknetz über einen Mobilfunk-CP.
  - FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL
- Konfigurierte Verbindungen
  - Angabe einer bestehenden Verbindung in dem Systemdatentyp TCON_Configured.

Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TSEND_C" erneut ausgeführt werden.

Daten über die bestehende Kommunikationsverbindung senden

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird. Wie oben beschrieben wird zuvor die Kommunikationsverbindung aufgebaut.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Verwenden Sie am Parameter DATA keinen Datenbereich mit dem Datentyp BOOL oder Array of BOOL. Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden. Wenn Sie einen Sendebereich mit optimiertem Zugriff am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist.

Die Kommunikationsverbindung abbauen und zurücksetzten

Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie für "TSEND_C" eine konfigurierte Verbindung verwenden.

Durch Setzen des Parameters COM_RST auf "1" kann die Verbindung jederzeit zurückgesetzt werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zu diesem Zeitpunkt Daten übertragen werden, kann es zum Datenverlust kommen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND_C":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Startet den Sendeauftrag bei einer steigenden Flanke.
CONT	Input	BOOL	E, A, M, D, L	Steuert die Kommunikationsverbindung: • 0: Kommunikationsverbindung trennen. • 1: Kommunikationsverbindung aufbauen und halten.
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Optionaler Parameter (ausgeblendet) Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden. Wenn Sie einen Sendebereich mit optimiertem Zugriff am Parameter DATA verwenden, muss am Parameter LEN der Wert "0" verwendet werden. Bei FDL-Verbindungen des CM 1542-5 beträgt die maximale Länge 240 Byte. Beachten Sie dabei die maximale verarbeitbaren Längen beim Verbindungspartner.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CONNECT	InOut	VARIANT	D	Zeiger auf die Struktur zur Verbindungsbeschreibung: • Programmierte Verbindung: – Bei TCP oder UDP verwenden Sie den Systemdatentyp TCON_IP_v4. Zur Beschreibung sieheAUTOHOTSPOT. – Bei TCP oder UDP mittels secure communication verwenden Sie die Struktur TCON_IP_V4_SEC oder TCON_QDN_SEC. Zur Beschreibung siehe: Verbindungsparameter nach TCON_IP_V4_SEC bzw. Verbindungsparameter nach TCON_QDN_SEC – Bei ISO-on-TCP verwenden Sie den Systemdatentyp TCON_IP_RFC. Zur Beschreibung siehe AUTOHOTSPOT. – Bei ISO verwenden Sie den Systemdatentyp TCON_ISOnative (nur bei CP 1543-1). Zur Beschreibung siehe Anweisung "TCON (Seite 3692)". – Bei Verbindungen zu SMS-Clients verwenden Sie den Systemdatentyp TCON_PHONE. Zur Beschreibung siehe AUTOHOTSPOT". – Für FDL-Verbindungen des CM 1542-5 verwenden Sie den Systemdatentyp TCON_FDL, siehe AUTOHOTSPOT. • Konfigurierte Verbindung: – Für bestehende Verbindungen verwenden Sie den Systemdatentyp TCON_Configured. Zur Beschreibung siehe unten "Systemdatentyp für konfigurierte Verbindungen".
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält. Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
ADDR	InOut	VARIANT	D	Ausgeblendeter Parameter, der aber bei UDP verwendet werden muss. Er enthält in diesem Fall einen Zeiger auf den Systemdatentyp TADDR_Param. In einem Datenbaustein mit dem Systemdatentyp TADDR_Param hinterlegen Sie die Adressinformation des Empfängers (IP-Adresse und Portnummer). Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 3742)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
COM_RST	InOut	BOOL	E, A, M, D, L	Optionaler Parameter (ausgeblendet) Bewirkt ein Zurücksetzen der Verbindung: • 0: Irrelevant • 1: Die vorhandene Verbindung wird zurückgesetzt. Der Parameter COM_RST wird nach Auswertung durch die Anweisung "TSEND_C" rückgesetzt und sollte daher nicht statisch verschaltet werden.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Sendeauftrag noch nicht gestartet oder noch in Bearbeitung. • 1: Sendeauftrag fehlerfrei ausgeführt. Dieser Zustand wird nur für einen Zyklus angezeigt. Der Ausgangsparameter DONE wird gesetzt, wenn ein Zwischenschritt bei der Bearbeitung erfolgreich abgeschlossen wurde (Verbindungsaufbau, Senden, Verbindungsabbau) und wenn die Ausführung von "TSEND_C" erfolgreich abgeschlossen ist.
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Sendeauftrag noch nicht gestartet oder bereits beendet. • 1: Sendeauftrag noch nicht beendet. Ein neuer Sendeauftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten beim Verbindungsaufbau, beim Senden der Daten oder beim Verbindungsabbau. Der Ausgangsparameter ERROR kann auf Grund eines Fehlers bei der Anweisung "TSEND_C" oder den intern verwendeten Kommunikations-Anweisungen gesetzt werden.
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung (siehe Beschreibung "Parameter ERROR und STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter REQ, CONT und COM_RST

Der Parameter CONT steuert den Verbindungsaufbau der Anweisung "TSEND_C" unabhängig von dem Parameter REQ. Das Verhalten des Parameters CONT hängt zum Teil davon ab, ob eine programmierte oder konfigurierte Verbindung verwendet wird:

- Bei CONT = "0": Es werden keine Daten gesendet (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Beim Wechsel von CONT = "0" auf "1":
  - Bei einer programmierten Verbindung wird diese mit "TCON" aufgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T_DIAG" überprüft.
- Bei CONT = "1":
  - Solange keine Daten gesendet werden (REQ="0"), wird die Verbindung mit "T_DIAG" überprüft.
  - Wird durch die intern verwendeten Kommunikations-Anweisungen gemeldet, dass kein Verbindungsendpunkt vorhanden ist, wird die Verbindung über "TCON" automatisch wieder aufgebaut.
- Beim Wechsel von CONT = "1" auf "0":
  - Bei einer programmierten Verbindung wird diese mit "TDISCON" abgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T_RESET" zurückgesetzt.

Der Parameter COM_RST setzt die Verbindung beim Wechsel von "0" auf "1" zurück:

- Ist eine Verbindung aufgebaut, wird diese über "T_RESET" zurückgesetzt (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Ist keine Verbindung aufgebaut, hat das Setzten des Parameters keine Auswirkungen.

Die Parameter REQ und COM_RST sind nur wirksam, wenn CONT auf "1" gesetzt wurde. Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern REQ, CONT und COM_RST:

REQ	CONT	COM_RST	Status der Anweisung	Beschreibung
irrelevant	0	irrelevant	Noch nicht ausgeführt	Kein Auftrag aktiv (STATUS = 7000).
irrelevant	0	irrelevant	Initialisierung	Verbindung wird abgebaut. Die Anweisung wird zurückgesetzt.
irrelevant	0 > 1	irrelevant	Verbindungsaufbau	Verbindung wird aufgebaut. Es werden noch keine Daten übertragen.
0	1	0	Verbindung hergestellt	Die Verbindung ist hergestellt und wird mit der Anweisung "T_DIAG" überwacht.
irrelevant	1	0 > 1	Verbindung hergestellt	Die Verbindung wird durch "T_RESET" kurzfristig unterbrochen und zurückgesetzt.
0 > 1	1	0	Verbindung hergestellt	Anweisung beginnt zu senden.
irrelevant	1	0 > 1	Daten werden gesendet	Datenübertragung wird unterbrochen. Die Verbindung wird zurückgesetzt.

Systemdatentyp für konfigurierte Verbindungen

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_Configured:

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceID	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	-	Referenz auf die Verbindung (Wertebereich: 1 bis 4095). Geben Sie die Verbindungs-ID der bestehenden Verbindung an.
4	ConnectionType	BYTE	-	Verbindungstyp Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.

Hinweis

Die Parameter InterfaceID und ConnectionType sind aus Kompatibilitätsgründen Bestandteil der Struktur zur Verbindungsbeschreibung nach TCON_Configured. Diese Parameter haben hier keine Auswirkung auf die Verbindungsparametrierung, lediglich der Parameter ID für die Verbindungs-ID wird in der Verbindungsparametrierung ausgewertet.

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Sendeauftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

DONE	BUSY	ERROR	Beschreibung
0	0	0	Die Anweisung wurde noch nicht ausgeführt (keine steigende Flanke am Parameter REQ).
0	1	0	Die Anweisung wird ausgeführt und ruft die intern verwendeten Kommunikations-Anweisungen auf.
1	0	0	Der Sendeauftrag wurde erfolgreich durchgeführt. Am Parameter STATUS wird "0000" ausgegeben. DONE = "1" wird nur für einen Zyklus angezeigt.
0	0	1	Die Ausführung der Anweisung oder ein Zwischenschritt bei der Bearbeitung wurde mit einem Fehler beendet. Kommt es durch eine intern verwendete Kommunikations-Anweisung zu einem Folgefehler, wird der Fehler angezeigt, welcher bei der Bearbeitung zuerst aufgetreten ist. Dieser Zustand wird nur für einen Zyklus angezeigt.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Beschreibung
0	0000	Sendeauftrag wurde fehlerfrei ausgeführt.
0	0001	Kommunikationsverbindung wurde aufgebaut.
0	0003	Kommunikationsverbindung wurde abgebaut.
0	7000	Keine Sendeauftragsbearbeitung aktiv, keine Kommunikationsverbindung aufgebaut.
0	7001	Erstaufruf beim Aufbau einer Verbindung.
0	7002	Zweitaufruf beim Aufbau einer Verbindung
0	7003	Kommunikationsverbindung wird abgebaut.
0	7004	Kommunikationsverbindung ist aufgebaut und wird überwacht. Keine Sendeauftragsverarbeitung aktiv.
0	7005	Senden der Daten läuft.
1	80A1	• Verbindung oder Port wird vom Anwender bereits verwendet. • Kommunikationsfehler: – Die angegebene Verbindung wurde noch nicht aufgebaut. – Die angegebene Verbindung wird gerade beendet. Eine Übertragung über diese Verbindung ist nicht möglich. – Die Schnittstelle wird neu initialisiert.
1	80A3	Die unterlagerte Anweisung "T_DIAG" hat den Abbau der Verbindung gemeldet.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, oder sie passt zur IP-Adresse des lokalen Partners.
1	80A7	Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.
1	80AA	Es läuft gerade ein Verbindungsaufbau mit der gleichen Verbindungs-ID durch einen anderen Baustein. Bitte wiederholen Sie den Auftrag durch einen erneuten Flankenanstieg am Parameter REQ.
1	80B3	• Bei Verwendung der Protokollvariante UDP enthält der Parameter ADDR keine Daten. • Fehler in der Verbindungsbeschreibung • Der lokale Port wird bereits in einer anderen Verbindungsbeschreibung verwendet.
1	80B4	Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.
1	80B6	Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.
1	80B7	• Für Systemdatentyp TCON_Param: Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len. • Für Systemdatentypen TCON_IP_V4 und TCON_IP_RFC: IP-Adresse des Partner-Endpunkts wurde auf 0.0.0.0 gesetzt.
1	8085	Parameter LEN ist größer als der größte zulässige Wert.

ERROR	STATUS* (W#16#...)	Beschreibung
1	8086	Der Parameter ID innerhalb des Parameters CONNECT liegt außerhalb des zulässigen Bereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich.
1	8088	Der Wert am Parameter LEN passt nicht zu den am Parameter DATA angegebenen Empfangsbereich.
1	8089	• Der Parameter CONNECT zeigt nicht auf einen Datenbaustein. • Der Parameter CONNECT zeigt nicht auf eine Verbindungsbeschreibung. • Die manuell erstellte Verbindungsbeschreibung hat für den gewählten Verbindungstyp eine falsche Struktur.
1	8091	Maximale Schachtelungstiefe überschritten.
1	809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.
1	809B	InterfaceID ist ungültig: • Sie zeigt nicht auf eine lokale CPU-Schnittstelle oder einen CP. • Sie darf bei Benutzung der Verbindungsparametrierung nicht den Wert 0 haben. • Sie darf in der verwendeten TCON_xxx-Struktur nicht den Wert 0 haben. Siehe TCON: Kommunikationsverbindung aufbauen (Seite 3692)
1	80C3	• Alle Verbindungsressourcen sind belegt. • Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 3700)" entfernt. • Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST = 1 beendet. • Temporär ist keine Empfangsressourcen beim Verbindungspartner verfügbar. Der Verbindungspartner ist nicht empfangsbereit.
1	80C5	• Verbindung durch den Kommunikationspartner abgebaut. • LSAP des remoten Verbindungspartners nicht freigegeben
1	80C6	Netzwerkfehler: • Remoter Partner kann nicht erreicht werden. • Physikalische Unterbrechung an PROFIBUS
1	8722	Parameter CONNECT: Der Quellbereich ist ungültig. Der Bereich ist im DB nicht vorhanden.
1	873A	Parameter CONNECT: Zugriff auf Verbindungsbeschreibung nicht möglich (z. B. weil der DB nicht verfügbar ist).
1	877F	Parameter CONNECT: Interner Fehler.
1	8822	Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden.
1	8824	Parameter DATA: Bereichsfehler im VARIANT-Zeiger.
1	8832	Parameter DATA: DB-Nummer ist zu groß.
1	883A	Parameter DATA: Kein Zugriff auf den Datenbereich, da z. B. der Datenbaustein nicht existiert

ERROR	STATUS* (W#16#...)	Beschreibung
1	887F	Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.
1	893A	Parameter ADDR: Zugriff auf den Sendebereich nicht möglich (z.B. weil der DB nicht vorhanden ist).
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Hinweis

Fehlermeldungen der Anweisungen "TCON", "TSEND", "T_DIAG", "T_RESET" und "TDISCON"

Die Anweisung "TSEND_C" verwendet intern die Anweisungen "TCON", "TSEND (Seite 3705)", "T_DIAG (Seite 3758)", "T_RESET (Seite 3753)" und "TDISCON (Seite 3700)". Die Fehlermeldungen dieser Anweisungen können ebenfalls am Parameter STATUS ausgegeben werden. Die Bedeutung der Fehlercodes ist bei den entsprechenden Anweisungen beschrieben. Bei gleichen Fehlercodes intern verwendeter Anweisungen mit unterschiedlicher Bedeutung kann über den Instanzdatenbaustein von "TSEND_C" ermittelt werden, durch welche Anweisung der Fehler ausgegeben wurde.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TRCV_C: Verbindung aufbauen und Daten empfangen

TRCV_C: Daten über Ethernet empfangen

Gültigkeit

Die folgende Beschreibung der Anweisung "TRCV_C" gilt für S7-1200-CPUs mit Firmware-Version < V4.0.

Beschreibung

Die Anweisung "TRCV_C" wird asynchron ausgeführt und hat die folgenden Funktionen:

1. **Eine Kommunikationsverbindung einrichten und aufbauen:**
   "TRCV_C" richtet eine TCP- oder ISO-on-TCP-Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht.
   Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Um die Verbindung aufzubauen, muss der Parameter CONT auf den Wert "1" gesetzt werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.
   Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TRCV_C" erneut ausgeführt werden.
   Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.

2. **Daten über die bestehende Kommunikationsverbindung empfangen:**
   Wenn der Parameter EN_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Beim Empfangen von Daten (steigende Flanke am Parameter EN_R) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten.
   Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.
   Nach einem erfolgreichen Datenempfang ist der Signalzustand am Parameter DONE "1". Wenn bei der Datenübertragung Fehler auftreten, wird der Parameter DONE auf "0" zurückgesetzt.

3. **Die Kommunikationsverbindung abbauen:**
   Die Kommunikationsverbindung wird sofort abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird.

Beim Setzen des Parameters COM_RST wird "TRCV_C" neu ausgeführt. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zum Zeitpunkt der Neuausführung Daten empfangen werden, kann es zum Datenverlust kommen.

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP oder ISO-on-TCP anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Empfangsmodi von TRCV_C

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

Protokollvariante	Verfügbarkeit der Daten im Empfangs-bereich	Parameter connection_type der Verbindungs-beschreibung	Parameter LEN
TCP (Ad-hoc-Modus)	Die Daten sind sofort verfügbar.	Hexadezimalwert: B#16#11 Integerwert: 17	0
TCP (Datenempfang mit angegebener Länge)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Da-tenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#11 Integerwert: 17	1 bis 8192
ISO on TCP (Nachrichtenorientiert Datenübertragung)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Da-tenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#12 Integerwert: 18	• 1 bis 1452, wenn ein CP verwendet wird. • 1 bis 8192, wenn kein CP verwendet wird.

TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter LEN den Wert "0" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Die tatsächlich empfangene Datenlänge wird am Parameter RCVD_LEN ausgegeben.

TCP (Datenempfang mit angegebener Länge)

Die Länge für den Datenempfang geben Sie durch den Wert des Parameters LEN vor. Der Datenempfang wird erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV_C" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV_C" einen Fehler. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV_C":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Empfangsfreigabe
CONT	Input	BOOL	E, A, M, D, L	Steuert die Kommunikationsverbindung: • 0: Kommunikationsverbindung trennen • 1: Kommunikationsverbindung aufbauen und halten Beim Empfangen von Daten (steigende Flanke am Parameter EN_R) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten.
LEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Länge der zu empfangenden Daten (größter Wert: 8192 Bytes). Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.
CONNECT	InOut	TCON_Param	D	Zeiger auf die Verbindungsbeschreibung Siehe auch: AUTOHOTSPOT
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Empfangsbereich Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
COM_RST	InOut	BOOL	E, A, M, D, L	Bewirkt einen Neustart der Anweisung: • 0: Irrelevant • 1: Vollständiger Neustart der Anweisung, wodurch die vorhandene Verbindung abgebaut wird
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter ERROR: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung
RCVD_LEN	Output	UINT	E, A, M, D, L	Tatsächlich empfangene Datenmenge in Byte

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TRCV_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Beschreibung
0	0000	Auftrag fehlerfrei ausgeführt.
0	0001	Der Verbindungsaufbau ist abgeschlossen.
0	0003	Der Verbindungsabbau ist abgeschlossen.
0	7000	Keine Auftragsbearbeitung aktiv.
0	7001	• Auftragsbearbeitung starten • Verbindung aufbauen • Auf Verbindungspartner warten
0	7002	Daten werden empfangen.
0	7003	Verbindung wird beendet.
0	7004	• Verbindung aufgebaut und überwacht • Keine Auftragsbearbeitung aktiv
0	7006	Daten werden gerade empfangen.
1	8085	• Parameter LEN ist größer als der größte zulässige Wert. • Der Wert am Parameter LEN oder DATA wurde nach dem ersten Aufruf geändert.
1	8086	Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
1	8088	Der Wert am Parameter LEN passt nicht zu dem am Parameter DATA angegebenen Empfangsbereich.
1	8089	Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.
1	8091	Maximale Schachtelungstiefe überschritten.
1	809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.
1	809B	Die ID des lokalen Geräts (local_device_id) in der Verbindungsbeschreibung entspricht nicht der CPU.

ERROR	STATUS* (W#16#...)	Beschreibung
1	80A0	Sammelfehler für Fehlercodes W#16#80A1 und W#16#80A2.
1	80A1	• Verbindung oder Port wird vom Anwender bereits verwendet. • Kommunikationsfehler: – Die angegebene Verbindung wurde noch nicht aufgebaut. – Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich. – Die Schnittstelle wird neu initialisiert.
1	80A2	Lokaler oder remoter Port wird vom System verwendet.
1	80A3	• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen. • Es wird versucht, eine nicht vorhandene Verbindung zu beenden.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.
1	80A7	Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.
1	80B2	Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.
1	80B3	Inkonsistente Parametrierung: Sammelfehler für Fehlercodes W#16#80A0 bis W#16#80A2, W#16#80A4, W#16#80B4 bis W#16#80B9.
1	80B4	Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.
1	80B6	Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.
1	80B7	Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	80C3	• Alle Verbindungsressourcen sind belegt. • Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt. • Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST= 1 beendet
1	80C6	Der remote Partner kann nicht erreicht werden (Netzwerkfehler).
1	8722	Fehler im Parameter CONNECT: Ungültiger Quellbereich (der Bereich ist im Datenbaustein nicht deklariert).
1	873A	Fehler im Parameter CONNECT: Zugriff auf die Verbindungsbeschreibung nicht möglich (kein Zugriff auf den Datenbaustein).
1	877F	Fehler im Parameter CONNECT: Interner Fehler
1	8922	Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden.
1	8924	Parameter DATA: Bereichsfehler im VARIANT-Pointer.

ERROR	STATUS* (W#16#...)	Beschreibung
1	8932	Parameter DATA: DB-Nummer ist zu groß.
1	893A	Parameter DATA: Kein Zugriff auf den Datenbereich, da z.B. der Datenbaustein nicht existiert.
1	897F	Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.
1	8A3A	Parameter ADDR: Kein Zugriff auf den Adressbereich, da z.B. der Datenbaustein nicht existiert.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Hinweis

Fehlermeldungen der Anweisungen "TCON", "TRCV" und "TDISCON"

Die Anweisung "TRV_C" verwendet intern die Anweisungen "TCON (Seite 3689)", "TRCV (Seite 3709)" und "TDISCON (Seite 3700)". Die Fehlermeldungen dieser Anweisungen sind in den jeweiligen Beschreibungen enthalten.

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TRCV_C: Verbindung aufbauen und Daten empfangen

Gültigkeit

Die folgende Beschreibung der Anweisung "TRCV_C" ist gültig für:

- Ethernet:
  - CPU S7-1200 mit Firmware-Stand ab V4.0 und CPU S7-1500
  - CPU S7-1500 ab Firmware-Stand V2.1: UDP-Multicast-Kommunikation
  - CPU S7-1500 ab Firmware-Stand V2.0 und TRCV_C ab Anweisungs-Version V3.2: Secure Communication
- PROFIBUS: FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL

Beschreibung

Die Anweisung "TRCV_C" wird asynchron ausgeführt und führt der Reihe nach die folgenden Funktionen aus:

- Eine Kommunikationsverbindung einrichten und aufbauen
- Daten über die bestehende Kommunikationsverbindung empfangen
- Die Kommunikationsverbindung abbauen oder zurücksetzen

Die Anweisung "TRCV_C" verwendet dabei intern die Kommunikations-Anweisungen "TCON", "TRCV", "T_DIAG", "T_RESET" und "TDISCON".

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP, ISO-on-TCP oder FDL anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen. UDP-Multicast-Verbindungen über die integrierten PROFINET-Schnittstellen sind bei S7-1500-CPUs ab Firmware-Stand V2.1 möglich.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Eine Kommunikationsverbindung einrichten und aufbauen

Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Die folgenden Verbindungstypen können verwendet werden:

- Programmierte Verbindungen (Aufbau der Verbindung über "TCON"):
  - TCP / UDP: Verbindungsbeschreibung über den Systemdatentyp TCON_IP_v4
  - TCP / UDP mittels secure communication: Verbindungsbeschreibung über die Systemdatentypen TCON_IP_V4_SEC oder TCON_QDN_SEC
  - ISO-on-TCP: Verbindungsbeschreibung über den Systemdatentyp TCON_IP_RFC
  - ISO: Verbindungsbeschreibung über den Systemdatentyp TCON_ISOnative (nur bei CP 1543-1)
  - Telecontrol-Verbindungen zu SMS-Clients: Verbindungsbeschreibung über den Systemdatentyp TCON_PHONE
    Für diesen Verbindungstyp benötigt die Station Zugang zum Mobilfunknetz über einen Mobilfunk-CP.
  - FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL
- Konfigurierte Verbindungen
  - Angabe einer bestehenden Verbindung in dem Systemdatentyp TCON_Configured.

Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TRCV_C" erneut ausgeführt werden.

Daten über die bestehende Kommunikationsverbindung empfangen

Wenn der Parameter EN_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss für den Parameter LEN als Wert "0" verwendet werden.

Empfangsmodi von TRCV_C:

- **TCP (Ad-hoc-Modus)**
  Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV_C" Daten mit dynamischen Längen zu empfangen.
  Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter ADHOC den Wert "1" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Die tatsächlich empfangene Datenlänge wird am Parameter RCVD_LEN ausgegeben.
- **TCP (Datenempfang mit angegebener Länge)**
  Für einen Datenempfang mit angegebener Länge weisen Sie dem Parameter ADHOC den Wert "0" zu. Ist der Ad-hoc-Modus deaktiviert, wird der Datenempfang erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.
- **ISO-on-TCP (Nachrichtenorientiert Datenübertragung)**
  Bei der Protokollvariante ISO-on-TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV_C" einen Fehler. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

Protokollvariante	Verfügbarkeit der Daten im Empfangsbereich	Parameter connection_type der Verbindungs-beschreibung	Parameter LEN
TCP (Ad-hoc-Modus)	Die Daten sind sofort verfügbar.	Hexadezimalwert: B#16#11 Integerwert: 17	1 bis maximale Länge (CPU-abhängig)
TCP (Datenempfang mit angegebener Länge)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#11 Integerwert: 17	1 bis 8192
ISO-on-TCP (Nachrichtenorientierte Datenübertragung)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#12 Integerwert: 18	1 bis 8192
FDL	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#15 Dezimal: 21	1 bis 240

Die Kommunikationsverbindung abbauen

Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie eine konfigurierte Verbindung verwenden.

Durch Setzen des Parameters COM_RST auf "1" kann die Verbindung jederzeit zurückgesetzt werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zu diesem Zeitpunkt Daten übertragen werden, kann es zum Datenverlust kommen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV_C":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Empfangsfreigabe
CONT	Input	BOOL	E, A, M, D, L	Steuert die Kommunikationsverbindung: • 0: Kommunikationsverbindung trennen. • 1: Kommunikationsverbindung aufbauen und nach Empfangen der Daten halten.
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Maximale Länge der zu empfangenen Daten. Wenn Sie einen Empfangsbereich mit optimiertem Zugriff am Parameter DATA verwenden, muss am Parameter LEN der Wert "0" verwendet werden.
ADHOC	Input	BOOL	E, A, M, D, L oder Konstante	Optionaler Parameter (ausgeblendet) Ad-hoc Modus für die Protokollvariante TCP verwenden. ADHOC muss den Wert FALSE haben, wenn kein TCP-Protokoll benutzt wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CONNECT	InOut	VARIANT	D	Zeiger auf die Verbindungsbeschreibung • Programmierte Verbindung: – Bei TCP oder UDP verwenden Sie die Struktur TCON_IP_v4 Zur Beschreibung siehe:AUTOHOTSPOT – Bei TCP oder UDP mittels secure communication verwenden Sie die Struktur TCON_IP_V4_SEC oder TCON_QDN_SEC. Zur Beschreibung siehe: Verbindungsparameter nach TCON_IP_V4_SEC bzw. Verbindungsparameter nach TCON_QDN_SEC – Bei ISO-on-TCP verwenden Sie die Struktur TCON_IP_RFC Zur Beschreibung siehe: AUTOHOTSPOT – Bei ISO verwenden Sie die Struktur TCON_ISOnative (nur bei CP 1543-1) Zur Beschreibung siehe Anweisung "TCON (Seite 3692)": "Aufbau der Verbindungsbeschreibung nach TCON_ISOnative" – Bei Verbindungen zu SMS-Clients verwenden Sie den Systemdatentyp TCON_PHONE. Zur Beschreibung siehe AUTOHOTSPOT. – Für FDL-Verbindungen des CM 1542-5 verwenden Sie den Systemdatentyp TCON_FDL, siehe AUTOHOTSPOT. • Konfigurierte Verbindung: – Für bestehende Verbindungen verwenden Sie den Systemdatentyp TCON_Configured. Zur Beschreibung siehe unten "Systemdatentyp für konfigurierte Verbindungen".
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Empfangsbereich. Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
ADDR	InOut	VARIANT	D	Ausgeblendeter Parameter, der aber bei UDP verwendet werden muss. Er enthält in diesem Fall einen Zeiger auf den Systemdatentyp TADDR_Param. In einem Datenbaustein mit dem Systemdatentyp TADDR_Param hinterlegen Sie die Adressinformation des Senders (IP-Adresse und Portnummer). Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 3742)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
COM_RST	InOut	BOOL	E, A, M, D, L	Optionaler Parameter (ausgeblendet) Bewirkt ein Zurücksetzen der Verbindung: • 0: Irrelevant • 1: Die vorhandene Verbindung wird zurückgesetzt. Der Parameter COM_RST wird nach Auswertung durch die Anweisung "TRCV_C" rückgesetzt und sollte daher nicht statisch verschaltet werden.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Empfang noch nicht gestartet oder noch in Bearbeitung. • 1: Empfang fehlerfrei ausgeführt. Dieser Zustand wird nur für einen Zyklus angezeigt. Der Ausgangsparameter DONE wird gesetzt, wenn ein Zwischenschritt bei der Bearbeitung erfolgreich abgeschlossen wurde (Verbindungsaufbau, Empfang, Verbindungsabbau) und wenn die Ausführung von "TRCV_C" erfolgreich abgeschlossen ist.
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Empfang noch nicht gestartet oder bereits beendet. • 1: Empfang noch nicht beendet. Ein neuer Sendeauftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten beim Verbindungsaufbau, beim Empfang der Daten oder beim Verbindungsabbau. Der Ausgangsparameter ERROR kann auf Grund eines Fehlers bei der Anweisung "TRCV_C" oder den intern verwendeten Kommunikations-Anweisungen gesetzt werden.
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung
RCVD_LEN	Output	UDINT	E, A, M, D, L	Tatsächlich empfangene Datenmenge in Byte

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter EN_R, CONT und COM_RST

Der Parameter CONT steuert den Verbindungsaufbau der Anweisung "TRCV_C" unabhängig von dem Parameter EN_R. Das Verhalten des Parameters CONT hängt zum Teil davon ab, ob eine programmierte oder konfigurierte Verbindung verwendet wird:

- Bei CONT = "0": Es werden keine Daten empfangen (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Beim Wechsel von CONT = "0" auf "1":
  - Bei einer programmierten Verbindung wird diese mit "TCON" aufgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T_DIAG" überprüft.
- Bei CONT = "1":
  - Solange keine Daten empfangen werden (EN_R ="0"), wird die Verbindung mit "T_DIAG" überprüft.
  - Wird durch die intern verwendeten Kommunikations-Anweisungen gemeldet, dass kein Verbindungsendpunkt vorhanden ist, wird die Verbindung über "TCON" automatisch wieder aufgebaut.
- Beim Wechsel von CONT = "1" auf "0":
  - Bei einer programmierten Verbindung wird diese mit "TDISCON" abgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T_RESET" zurückgesetzt.

Der Parameter COM_RST setzt die Verbindung beim Wechsel von "0" auf "1" zurück:

- Ist eine Verbindung aufgebaut, wird diese über "T_RESET" zurückgesetzt (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Ist keine Verbindung aufgebaut, hat das Setzten des Parameters keine Auswirkungen.

Die Parameter EN_R und COM_RST sind nur wirksam, wenn CONT auf "1" gesetzt wurde. Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern EN_R, CONT und COM_RST:

EN_R	CONT	COM_RST	Status der Anweisung	Beschreibung
irrelevant	0	irrelevant	Noch nicht ausgeführt	Kein Auftrag aktiv (STATUS = 7000).
irrelevant	0	irrelevant	Initialisierung	Verbindung wird abgebaut. Die Anweisung wird zurückgesetzt.
irrelevant	0 > 1	irrelevant	Verbindungsaufbau	Verbindung wird aufgebaut. Es werden noch keine Daten übertragen.
0	1	0	Verbindung hergestellt	Die Verbindung ist hergestellt und wird mit der Anweisung "T_DIAG" überwacht.
irrelevant	1	0 > 1	Verbindung hergestellt	Die Verbindung wird durch "T_RESET" kurzfristig unterbrochen und zurückgesetzt.
0 > 1	1	0	Verbindung hergestellt	Anweisung beginnt zu empfangen.
irrelevant	1	0 > 1	Daten werden empfangen	Datenübertragung wird unterbrochen. Die Verbindung wird zurückgesetzt.

Systemdatentyp für konfigurierte Verbindungen

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_Configured:

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceID	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	-	Referenz auf die Verbindung (Wertebereich: 1 bis 4095). Geben Sie die Verbindungs-ID der bestehenden Verbindung an.
4	ConnectionType	BYTE	-	Verbindungstyp Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TRCV_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

DONE	BUSY	ERROR	Beschreibung
0	0	0	Die Anweisung wurde noch nicht ausgeführt (keine steigende Flanke am Parameter EN_R).
0	1	0	Die Anweisung wird ausgeführt und ruft die intern verwendeten Kommunikations-Anweisungen auf.
1	0	0	Der Empfang wurde erfolgreich durchgeführt. Am Parameter STATUS wird "0000" ausgegeben. DONE = "1" wird nur für einen Zyklus angezeigt.
0	0	1	Die Ausführung der Anweisung oder ein Zwischenschritt bei der Bearbeitung wurde mit einem Fehler beendet. Kommt es durch eine intern verwendete Kommunikations-Anweisung zu einem Folgefehler, wird der Fehler angezeigt, welcher bei der Bearbeitung zuerst aufgetreten ist. Dieser Zustand wird nur für einen Zyklus angezeigt.

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Beschreibung
0	0000	Empfangsauftrag fehlerfrei ausgeführt.
0	0001	Kommunikationsverbindung wurde aufgebaut.
0	0003	Kommunikationsverbindung wurde abgebaut.
0	7000	Keine Auftragsbearbeitung aktiv.
0	7001	Erstaufruf beim Aufbau einer Verbindung.
0	7002	Zweitaufruf beim Aufbau einer Verbindung.
0	7003	Kommunikationsverbindung wird abgebaut.

ERROR	STATUS (W#16#...)	Beschreibung
0	7004	Kommunikationsverbindung ist aufgebaut und wird überwacht. Keine Empfangsauftragsbearbeitung aktiv.
0	7006	Daten werden gerade empfangen.
1	8085	• Parameter LEN ist größer als der größte zulässige Wert. • Der Wert am Parameter LEN oder DATA wurde nach dem ersten Aufruf geändert.
1	8086	Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
1	8088	Der Wert am Parameter LEN passt nicht zu dem am Parameter DATA angegebenen Empfangsbereich.
1	8089	• Der Parameter CONNECT zeigt nicht auf einen Datenbaustein. • Der Parameter CONNECT zeigt nicht auf eine Verbindungsbeschreibung. • Die manuell erstellte Verbindungsbeschreibung hat für den gewählten Verbindungstyp eine falsche Struktur.
1	8091	Maximale Schachtelungstiefe überschritten.
1	809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.
1	809B	InterfaceID ist ungültig: • Sie zeigt nicht auf eine lokale CPU-Schnittstelle oder einen CP. • Sie darf bei Benutzung der Verbindungsparametrierung nicht den Wert 0 haben. • Sie darf in der verwendeten TCON_xxx-Struktur nicht den Wert 0 haben. Siehe TCON: Kommunikationsverbindung aufbauen (Seite 3692)
1	80A1	• Verbindung oder Port wird vom Anwender bereits verwendet. • Kommunikationsfehler: – Die angegebene Verbindung wurde noch nicht aufgebaut. – Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich. – Die Schnittstelle wird neu initialisiert.
1	80A3	Die unterlagerte Anweisung "T_DIAG" hat den Abbau der Verbindung gemeldet.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, oder sie passt zur IP-Adresse des lokalen Partners.
1	80A7	Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.
1	80AA	Es läuft gerade ein Verbindungsaufbau mit der gleichen Verbindungs-ID durch einen anderen Baustein. Bitte wiederholen Sie den Auftrag durch einen erneuten Flankenanstieg am Parameter REQ.
1	80B3	• Bei Verwendung der Protokollvariante UDP enthält der Parameter ADDR keine Daten. • Fehler in der Verbindungsbeschreibung • Der lokale Port wird bereits in einer anderen Verbindungsbeschreibung verwendet.
1	80B4	Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.
1	80B6	Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.

ERROR	STATUS (W#16#...)	Beschreibung
1	80B7	• Für Systemdatentyp TCON_Param: Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len. • Für Systemdatentypen TCON_IP_V4 und TCON_IP_RFC: IP-Adresse des Partner-Endpunkts wurde auf 0.0.0.0 gesetzt.
1	80C3	• Alle Verbindungsressourcen sind belegt. • Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt. • Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST = 1 beendet
1	80C6	Der remote Partner kann nicht erreicht werden (Netzwerkfehler).
1	8722	Fehler im Parameter CONNECT: Ungültiger Quellbereich (der Bereich ist im Datenbaustein nicht deklariert).
1	873A	Fehler im Parameter CONNECT: Zugriff auf die Verbindungsbeschreibung nicht möglich (kein Zugriff auf den Datenbaustein).
1	877F	Fehler im Parameter CONNECT: Interner Fehler
1	8922	Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden.
1	8924	Parameter DATA: Bereichsfehler im VARIANT-Zeiger.
1	8932	Parameter DATA: DB-Nummer ist zu groß.
1	893A	Parameter DATA: Kein Zugriff auf den Datenbereich, da z.B. der Datenbaustein nicht existiert.
1	897F	Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.
1	8A3A	Parameter ADDR: Kein Zugriff auf den Adressbereich, da z.B. der Datenbaustein nicht existiert.

Hinweis

Fehlermeldungen der Anweisungen "TCON", "TRCV" und "TDISCON"

Die Anweisung "TRV_C" verwendet intern die Anweisungen "TCON (Seite 3692)", "TRCV (Seite 3709)" und "TDISCON (Seite 3700)". Die Fehlermeldungen dieser Anweisungen sind in den jeweiligen Beschreibungen enthalten.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für Sendefunktionen (Seite 3656).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

TSEND_C: Verbindung aufbauen und Daten senden (Seite 3631)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für Sendefunktionen

Einleitung

In dem folgenden Beispiel stellen Sie eine programmierte Verbindung zwischen zwei CPUs der Baurreihe S7-1500 her und senden eine Zeichenkette von der CPU 1 zur CPU 2. Die zu übermittelnde Zeichenkette ist vom Datentyp STRING.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Deren Verbindung ist nicht konfiguriert.

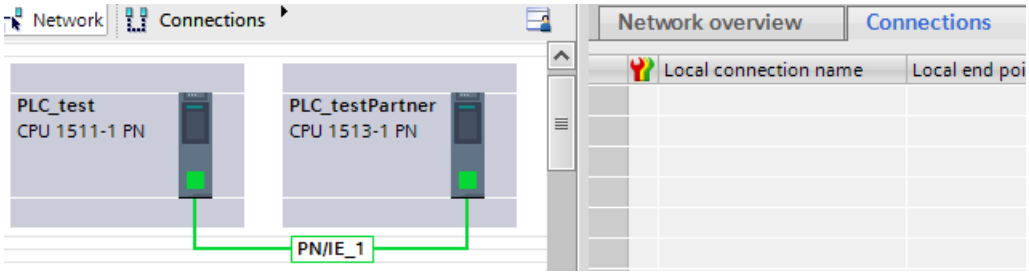

- Je CPU ist unter Eigenschaften > Schutz mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Programm der CPU 1

Zur Ablage der Daten von "TSEND_C" legen Sie in einem globalen Datenbaustein neun Variablen an.

SLI_gDB_TSEND_C

		Name	Data type	Start value
1		Static		
2		start	Bool	false
3		comControl	Bool	false
4		done	Bool	false
5		busy	Bool	false
6		error	Bool	false
7		status	Word	16#0
8		memErrStatus	Word	16#0
9		sendData	String	'ProcessXYZ'

Sie erstellen einen FB "SLI_FB_TSEND_C" und legen darin die folgenden lokalen Variablen an.

7	Static			
8	TSEND_C_Instance	TSEND_C		
9	done	Bool	false	Non-retain

Netzwerk 1: Die Parameter der Anweisung "TSEND_C" verschalten Sie wie folgt. Den Parameter CONNECT verschalten Sie noch nicht.

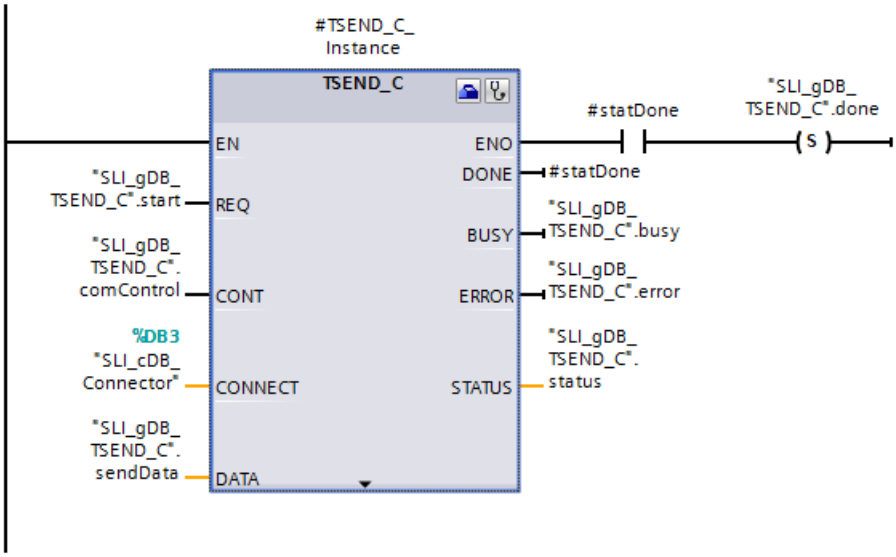

Netzwerk 2: Den Status bei einem Fehler von TSEND_C speichern Sie wie folgt.

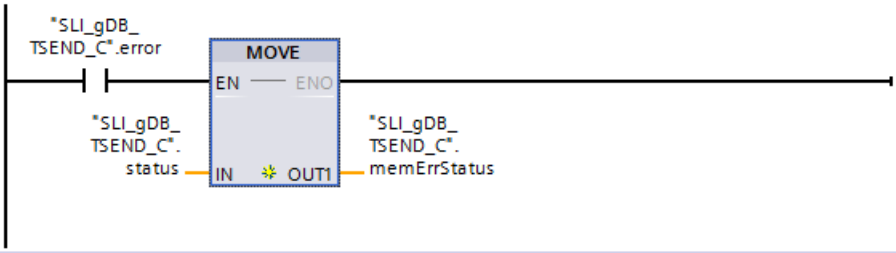

Konfiguration von TSEND_C

Zur Verschaltung des Parameters CONNECT öffnen Sie den Wizard der Anweisung "TSEND_C" über Eigenschaften > Konfiguration.

Sie nehmen folgende Einstellungen für die Konfiguration von TSEND_C vor:

Eingabefeld	Eintrag
Endpunkt	Sie wählen für Sender und Empfänger über die Klappliste eine CPU aus. Schnittstelle, Subnetz und Adresse werden automatisch eingetragen.
Verbindungsdaten	Über die Klappliste und die Auswahl "Neu" erstellen Sie je CPU einen Datenbaustein. Dieser Datenbaustein wird für die Ablage der Verbindungsdaten benötigt. Der Name des Datenbausteins ist frei wählbar. Für die CPU 1 (lokale CPU) ist der aktive Verbindungsaufbau aktiviert.

Eingabefeld	Eintrag
Verbindungstyp	Sie wählen den Eintrag "TCP". Dadurch wird für den Verbindungsaufbau eine Ethernet-Verbindung mit dem Protokoll "TCP" verwendet.
Konfigurationsmodus	Sie wählen den Eintrag "Programmbaustein verwenden". Dadurch wird der Verbindungsaufbau über eine programmierte Verbindung erfolgen.
Verbindungs-ID	Sie tragen eine selbst gewählte Verbindungs-ID für die Kommunikationsverbindung ein. Die Verbindungs-ID darf nicht bereits im Projekt vergeben sein.
Partner Port	Sie tragen einen Wert für den Partner Port ein. Der Wert muss >=2000 sein.

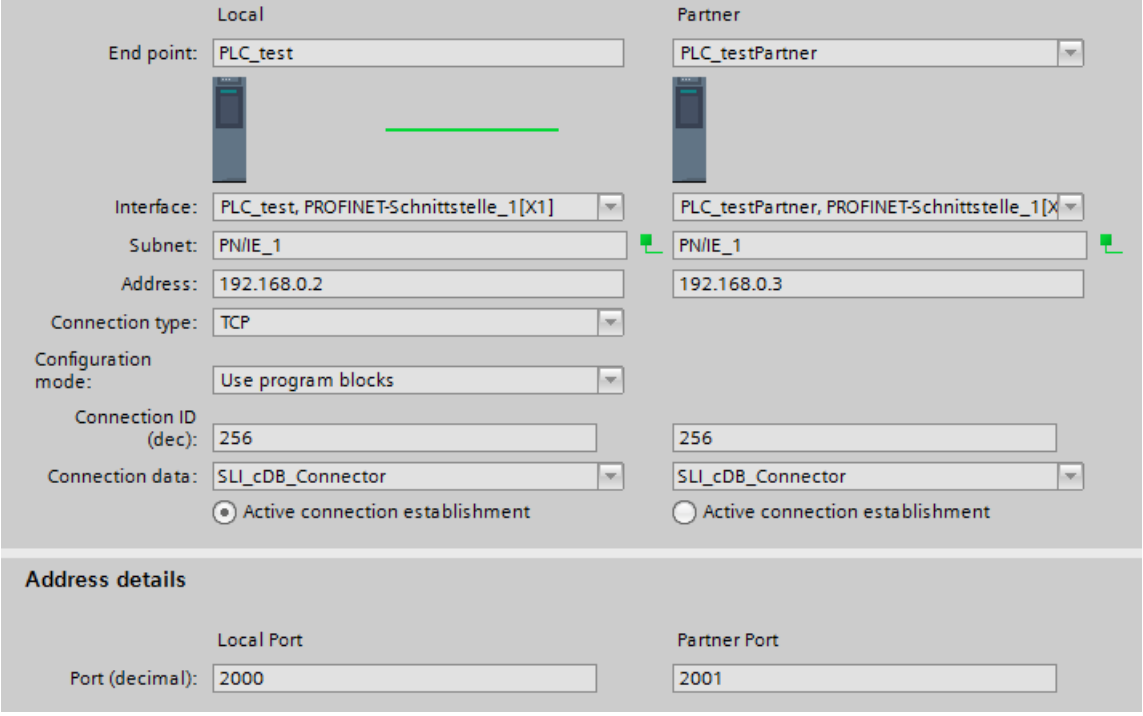

Programm der CPU 2

Zur Ablage der Daten von "TRCV_C" legen Sie in einem globalen Datenbaustein zehn Variablen an.

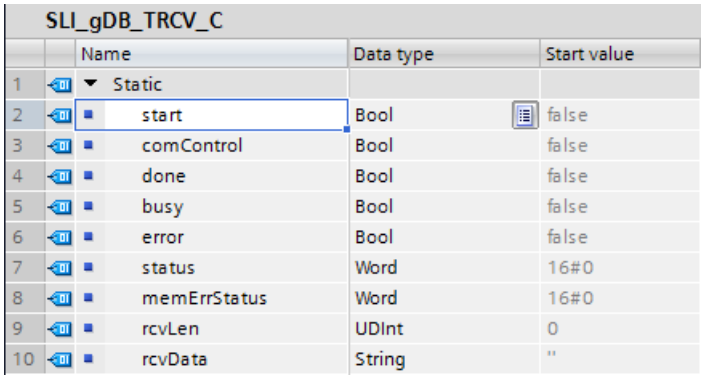

SLI_gDB_TRCV_C

		Name	Data type	Start value
1		Static		
2		start	Bool	false
3		comControl	Bool	false
4		done	Bool	false
5		busy	Bool	false
6		error	Bool	false
7		status	Word	16#0
8		memErrStatus	Word	16#0
9		rcvLen	UDInt	0
10		rcvData	String	''

Sie erstellen einen FB "SLI_FB_TRCV_C" und legen darin die folgenden lokalen Variablen an.

7	Static			
8	TRCV_C_Instance	TRCV_C		
9	statDone	Bool	false	Non-retain
10	statRcvLen	UDInt	0	Non-retain

Netzwerk 1: Die Parameter der Anweisung "TRCV_C" verschalten Sie wie folgt. Den Parameter CONNECT verschalten Sie noch nicht.

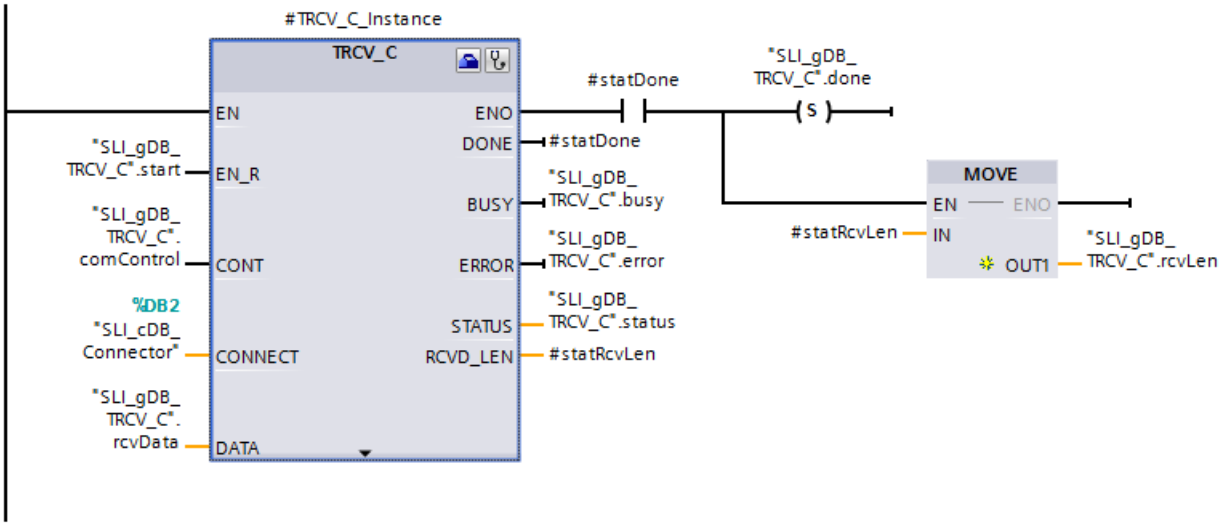

Netzwerk 2: Den Status bei einem Fehler von TRCV_C speichern Sie wie folgt.

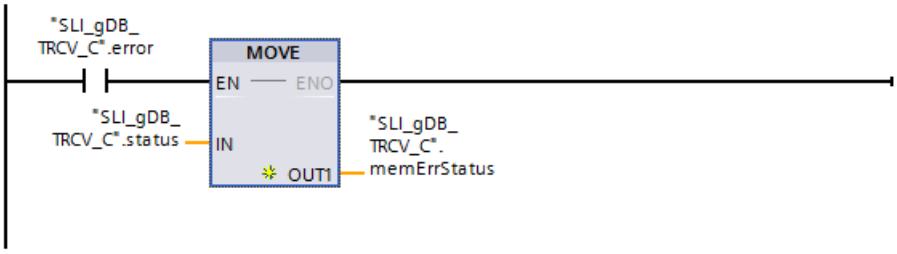

Konfiguration von TRCV_C

Folgende Einstellungen nehmen Sie für die Konfiguration von TRCV_C vor:

Hinweis

Verbindungsdaten

Mit dem Konfigurieren von TSEND_C ist bereits je CPU ein Datenbaustein ("SLI_cDB_Connector") mit den hinterlegten Verbindungsdaten erstellt. Anstatt neue Datenbausteine zu erstellen und die Verbindungsdaten einzutragen, können Sie einfach die erstellten Datenbausteine ("SLI_cDB_Connector") verwenden.

Damit die erstellten Datenbausteine auswählbar sind, müssen diese im Projektbaum unter Programbausteine > Systembausteine weiterhin zu finden sein.

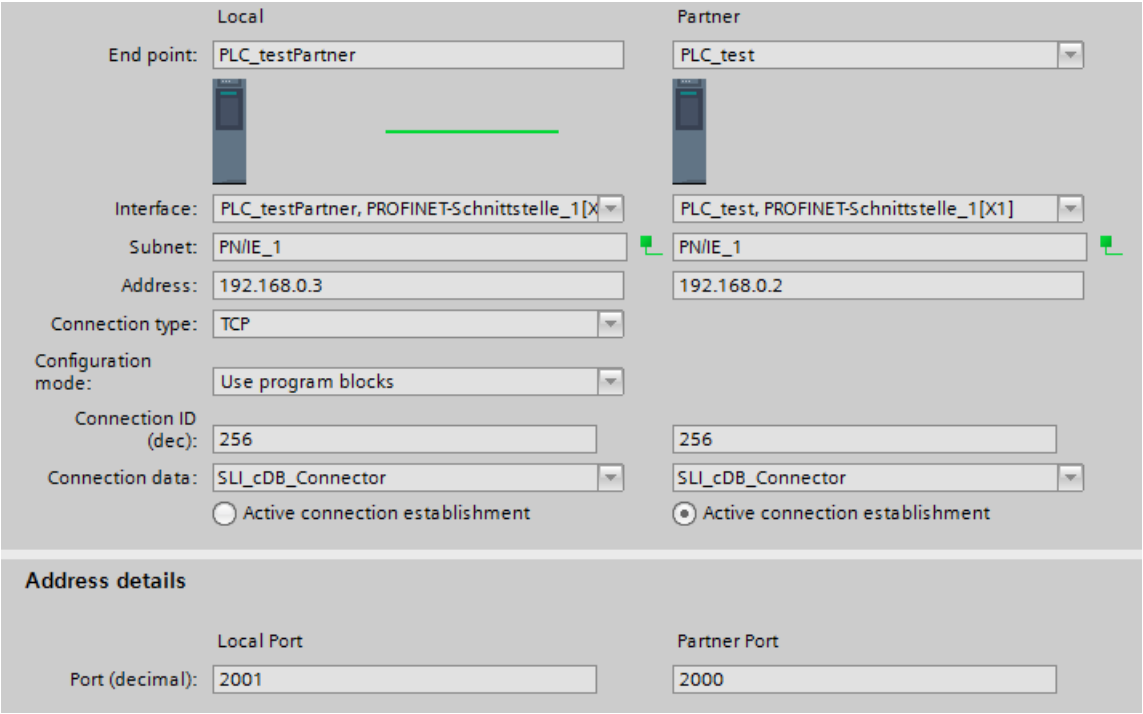

Vorgehen für Verbindungsaufbau

Um im Modus RUN die Kommunikationsverbindung aufzubauen, gehen Sie wie folgt vor:

1. Ändern Sie für TRCV_C den Parameter EN_R auf "1".
2. Ändern Sie für TRCV_C den Parameter CONT auf "1".
3. Ändern Sie für TSEND_C den Parameter REQ auf "1".
4. Ändern Sie für TSEND_C den Parameter CONT auf "1".

Vorgehen für Verbindungsabbbau

Um im Modus RUN die Kommunikationsverbindung abzubauen, gehen Sie wie folgt vor:

1. Ändern Sie für TSEND_C den Parameter CONT auf "0".
2. Ändern Sie für TSEND_C den Parameter REQ auf "0".
3. Ändern Sie für TRCV_C den Parameter CONT auf "0".
4. Ändern Sie für TRCV_C den Parameter EN_R auf "0".

Verhalten der CPU 1

Wenn der Eingangsparameter REQ ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "TSEND_C" gestartet. Wenn der Eingangsparameter CONT ("comControl") dem Signalzustand "TRUE" liefert, erstellt die Anweisung "TSEND_C" eine Kommunikationsverbindung zwischen CPU 1 und CPU 2. Dazu werden die Verbindungsdaten über den Eingangsparameter CONNECT (bzw. den Datenbaustein "SLI_cDB_Connector") abgerufen.

Über mehrere Aufrufe hinweg übermittelt die Anweisung "TSEND_C" den am Eingangsparameter DATA ("sendData") erfassten Datensatz. Über den Ausgangsparameter DONE ("#statDone") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus von DONE ("#statDone") in der Variable "done".

Nach der Übermittelung des Datensatzes an die CPU 2 wird die Kommunikationsverbindung weiter überwacht (Status "7004"). Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_TSEND_C

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	comControl	Bool	false	TRUE
4	done	Bool	false	TRUE
5	busy	Bool	false	FALSE
6	error	Bool	false	FALSE
7	status	Word	16#0	16#7004
8	memErrStatus	Word	16#0	16#0000
9	sendData	String	'ProcessXYZ'	'ProcessXYZ'

Verhalten der CPU 2

Wenn der Eingangsparameter EN_R ("start") den Signalzustand "TRUE" liefert, wird die Anweisung "TRCV_C" gestartet. Wenn der Eingangsparameter CONT ("comControl") den Signalzustand "TRUE" liefert, erstellt die Anweisung "TRCV_C" eine Kommunikationsverbindung zwischen CPU 2 und CPU 1. Dazu werden die Verbindungsdaten über den Eingangsparameter CONNECT (bzw. Datenbaustein "SLI_cDB_Connector") abgerufen.

Über mehrere Aufrufe hinweg empfängt die Anweisung "TRCV_C" den übermittelten Datensatz. Am Parameter DATA ("rcvData") wird der Datensatz erfasst. Über den Ausgangsparameter DONE ("done") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Die Länge in BYTE des tatsächlich übermittelten Datensatzes wird uber den Ausgangsparameter RCVD_LEN ("#statRcvLen") erfasst. Dessen Wert wird nur während des Erfolgsstatus angezeigt. Danach wird "0" erfasst.

Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, machen Sie Folgendes:

- Sie speichern den Wert von "#statRcvLen" in "rcvLen".
- Sie speichern den Wert von "#statDone" in "done".
- Sie speichern den Wert von "#status" in "memErrStatus".

Nach dem Empfang des Datensatzes durch die CPU 2 wird die Kommunikationsverbindung weiter überwacht (Status "7006"). Am Ausgangsparameter ERROR ("error") bzw. der Variable "memErrStatus" wird angezeigt, dass die Verarbeitung des Beispiels ohne Fehler verläuft.

SLI_gDB_TRCV_C

	Name	Data type	Start value	Monitor value
1	Static			
2	start	Bool	false	TRUE
3	comControl	Bool	false	TRUE
4	done	Bool	false	TRUE
5	busy	Bool	false	TRUE
6	error	Bool	false	FALSE
7	status	Word	16#0	16#7006
8	memErrStatus	Word	16#0	16#0000
9	rcvLen	UDInt	0	256
10	rcvData	String	''	'ProcessXYZ'

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

TMAIL_C: E-Mail übertragen

Beschreibung TMAIL_C

Beschreibung

Über die Anweisung "TMAIL_C" versenden Sie eine E-Mail über die Ethernet-Schnittstelle der S7-1500 oder S7-1200 > V4.0, eines Kommunikationsmoduls (CM) oder eines Kommunikationsprozessors (CP).

Voraussetzung zur Verwendung der Anweisung ist, dass die Hardware bereits konfiguriert ist und die Infrastruktur des Netzwerks eine Kommunikationsverbindung zu dem Mailserver zulässt.

Die Inhalte der E-Mail sowie die Verbindungsdaten definieren Sie über die folgenden Parameter:

- Die Empfängeradressen definieren Sie über die Parameter TO_S und CC.
- Den Inhalt der E-Mail definieren über die Parameter SUBJECT und TEXT.
- Einen Anhang können Sie über VARIANT-Zeiger an den Parametern ATTACHMENT und ATTACHMENT_NAME definieren.

- Die Definition der Verbindungsdaten sowie die Adressierung und Authentisierung für den Mailserver erfolgt über einen der Systemdatentypen TMail_V4, TMail_V6 oder TMail_FQDN am Parameter MAIL_ADDR_PARAM.
  - Bei Verwendung der Schnittstelle der S7-1500 CPU verwenden Sie ausschließlich den Systemdatentyp TMail_V4. Der Versand der Mail ist in diesem Fall nur über SMTP möglich.
  - Bei Verwendung der Schnittstelle eines CMs/CPs können Sie alle Systemdatentypen verwenden. Der Versand der Mail ist auch über SMTPS möglich.
- Sie starten das Versenden einer E-Mail mit einem Flankenwechsel von "0" auf "1" am Parameter REQ.
- Über die Ausgangsparameter "BUSY", "DONE", "ERROR" und "STATUS" wird der Zustand des Auftrags angezeigt.

Das direkte Versenden einer SMS über die Anweisung "TMAIL_C" ist nicht möglich. Ob eine Weiterleitung der E-Mail als SMS durch den Mailserver erfolgen kann, ist abhängig von dem Telekommunikationsanbieter.

Hinweis

Anzahl zu versendender E-Mails

Über eine PLC können Sie gleichzeitig mehr als eine E-Mail versenden. Bei Verwendung eines CP 1243-8 oder eines CP 1543-1 können Sie für jeden CP immer nur jeweils eine E-Mail versenden. Bei Verwendung von zwei CPs ist also das parallele Versenden von zwei E-Mails möglich.

Arbeitsweise der Anweisung

Die Anweisung "TMAIL_C" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Beim Aufruf der Anweisung "TMAIL_C" ist die Angabe einer Instanz zwingend erforderlich.

In den folgenden Fällen wird die Verbindung zum Mailserver abgebrochen:

- Beim Wechsel der CPU in den Betriebszustand STOP, während "TMAIL_C" aktiv ist.
- Wenn am Industrial Ethernet-Bus Kommunikationsprobleme auftreten.

In diesen Fällen wird das Versenden der E-Mail unterbrochen und sie erreicht ihre Empfänger nicht. Nach erfolgreicher Ausführung der Anweisung und dem Versenden der Mail wird die Verbindung ebenfalls abgebaut.

ACHTUNG
Ändern von Anwenderprogrammen Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf die Aufrufe von "TMAIL_C" auswirken, nur ändern: • Wenn die CPU im Betriebszustand "STOP" ist. • Wenn keine Mail gesendet wird (REQ = 0 und BUSY = 0). Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "TMAIL_C" oder Aufrufe der Instanz von "TMAIL_C" enthalten. Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den TCP/IP-Kommunikationsfunktionen über Industrial Ethernet geraten. Nach dem Übertragen der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

Datenkonsistenz

Die Parameter TO_S, CC, SUBJECT, TEXT, ATTACHMENT und MAIL_ADDR_PARAM werden während des Betriebs der Anweisung "TMAIL_C" von diesem übernommen und dürfen somit erst geändert werden, wenn der Auftrag beendet wurde (BUSY = 0).

SMTP-Authentisierung

Unter Authentisierung versteht man ein Verfahren zur Sicherstellung einer Identität z. B. durch eine Passwortabfrage.

Die Anweisung "TMAIL_C" unterstützt bei Verwendung der Schnittstelle der S7-1500 CPU das SMTP-Authentisierungsverfahren AUTH-LOGIN, das von den meisten Mailservern gefordert wird. Informationen über die Authentisierungsverfahren Ihres Mail-Servers entnehmen Sie bitte dem Handbuch des Mail-Servers oder der Website Ihres Internet Service Providers.

- Um das Authentisierungsverfahren AUTH-LOGIN zu nutzen, benötigt die Anweisung "TMAIL_C" den Benutzernamen, mit dem er sich beim Mailserver anmelden kann. Dieser Benutzername entspricht dem Benutzernamen mit dem Sie ein Mailkonto auf Ihrem Mailserver eingerichtet haben. Er wird über den Parameter UserName der Struktur am Parameter MAIL_ADDR_PARAM übergeben.
  Wird in der Struktur am Parameter MAIL_ADDR_PARAM kein Benutzername angegeben, so wird das Authentisierungsverfahren AUTH-LOGIN auch nicht verwendet. Die E-Mail wird dann ohne Authentisierung versendet.
- Des Weiteren benötigt die Anweisung "TMAIL_C" zur Anmeldung das dazugehörige Passwort. Dieses Passwort entspricht dem Passwort, das Sie beim Einrichten Ihres Mailkontos vergeben haben. Es wird über den Parameter PassWord der Struktur am Parameter MAIL_ADDR_PARAM übergeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TMAIL_C":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z oder Konstante	Steuerparameter REQUEST: Aktiviert das Versenden einer E-Mail bei steigender Flanke.
TO_S (Seite 3671)	Input	STRING	D, L oder Konstante	Empfängeradressen STRING mit einer maximalen Länge von 240 Zeichen (Byte). Zur Schreibweise der Mailadresse siehe Beispiel in der Beschreibung des Parameters.
CC (Seite 3671)	Input	STRING	D, L oder Konstante	CC-Empfängeradressen (optional) STRING mit einer maximalen Länge von 240 Zeichen (Byte). Gleiche Schreibweise der Mailadresse wie bei dem Parameter TO_S. Wird hier ein Leerstring zugewiesen, wird die E-Mail nicht an einen CC-Empfänger gesendet.
SUBJECT	Input	STRING	D, L oder Konstante	Betreff der E-Mail STRING mit einer maximalen Länge von 240 Zeichen (Byte).
TEXT	Input	STRING	D, L oder Konstante	Text der E-Mail (optional) STRING mit einer maximalen Länge von 240 Zeichen (Byte). Wird an diesem Parameter ein Leerstring zugewiesen, wird die E-Mail ohne Text versendet.
ATTACHMENT	Input	VARIANT	D	Anhang der E-Mail (optional) Referenz auf ein Byte-/Wort-/Doppelwort-Feld (ArrayOfByte, ArrayOfWord oder ArrayOfDWord) mit einer maximalen Länge von 64 KByte. Wird kein Wert zugewiesen, wird die E-Mail ohne Anhang versendet.
ATTACHMENT_NAME	Input	STRING	D, L oder Konstante	Name des Anhangs der E-Mail (optional) Referenz auf eine Zeichenkette mit einer maximalen Länge von 50 Zeichen (Byte) zur Definition des Dateinamens des Anhangs. Wird an diesem Parameter ein Leerstring zugewiesen, wird der Anhang der E-Mail mit einem Dateinamen empfangen, den das E-Mail-Empfangsprogramm vergibt. Es wird daher empfohlen, einen definierten Dateinamen zu verwenden.
MAIL_ADDR_PARAM (Seite 3672)	Input	VARIANT	D	Parameter der Verbindung und Adresse des E-Mail-Servers Verwenden Sie zur Definition der Verbindungsparameter die Struktur TMail_V4, TMail_V6 oder TMail_FQDN (siehe Beschreibung des Parameters).

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DONE (Seite 3681)	Output	BOOL	E, A, M, D, L	Zustandsparameter • DONE = 0: Auftrag ist noch nicht gestartet oder wird noch ausgeführt. • DONE = 1: Auftrag wurde fehlerfrei ausgeführt.
BUSY (Seite 3681)	Output	BOOL	E, A, M, D, L	Zustandsparameter • BUSY = 0: Die Bearbeitung von "TMAIL_C" wurde beendet. • BUSY = 1: Das Senden der E-Mail ist noch nicht abgeschlossen.
ERROR (Seite 3681)	Output	BOOL	E, A, M, D, L	Zustandsparameter • ERROR = 0: Kein Fehler aufgetreten. • ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefern detaillierte Auskunft über die Art des Fehlers.
STATUS (Seite 3682)	Output	WORD	E, A, M, D, L	Zustandsparameter Rückgabewert bzw. Fehlerinformation der Anweisung "TMAIL_C" (siehe Beschreibung des Parameters).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Optionale Parameter

Die optionalen Parameter CC, TEXT und ATTACHMENT werden nur mit der E-Mail versendet, wenn die entsprechenden Parameter einen String der Länge > 0 enthalten.

Beispiel

Unter dem folgenden Link finden Sie ein Beispiel zum Versenden von E-Mails mit der Anweisung TMAIL_C: Beispiel: E-Mail versenden mit TMAIL_C (Seite 3685)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Beschreibung TMAIL_C ab Version V4.0 (Seite 3667)

Beschreibung TMAIL_C ab Version V5.0 (Seite 3668)

Beschreibung TMAIL_C ab Version V4.0

Einleitung

Im Folgenden wird die Anweisung TMAIL_C ab Version V4.0 beschrieben. Dabei wird nicht auf alle Einzelheiten eingegangen, sondern es werden nur die Unterschiede zu den Anweisungs-Versionen < V4.0 erläutert.

Auf die Handhabung der benötigten Zertifikate wird hier ebenfalls nicht eingegangen.

CPs mit Nutzung von Secure Communication

Über die Anweisung TMAIL_C ab Version V4.0 versenden die folgenden CPs gesicherte E-Mails:

- CP 1543-1 - V2.0
- CP 1542SP-1 IRC - V1.0, CP 1543SP-1 - V1.0
- CP 1243-1 - V2.1
- CP 1242-7 GPRS V2 - V2.1
- CP 1243-7 LTE - V2.1
- CP 1243-8 IRC - V2.1

Die zu sendenden Daten und den TCP-Port des E-Mail-Servers legen Sie am Parameter MAIL_ADDR_PARAM der folgenden Systemdatentypen fest:

TMAIL_V4_SEC, TMAIL_V6_SEC, TMAIL_QDN_SEC

Hinweis

Kompatibilität zu den TMAIL_C-Versionen < V4.0

Aus Kompatibilitätsgründen können Sie auch die bisherigen Systemdatentypen TMAIL_V4, TMAIL_V6 und TMAIL_FQDN wie folgt verwenden:

- CPUs:
  TMAIL_V4, TMAIL_FQDN
- Alle oben genannten CPs und CP 1542SP-1:
  TMAIL_V4, TMAIL_V6, TMAIL_FQDN

Regeln für CPs

Beachten Sie folgende Regeln:

- Für jede TMAIL_C-Instanz einer CPU dürfen Sie nur einen CP als E-Mail-Client verwenden.
- Die E-Mail-Server müssen von den CPs (E-Mail-Clients) über ein IP-Netzwerk erreichbar sein.
- Falls ein E-Mail Server über seinen voll qualifizierten Domänen-Namen erreicht werden soll, müssen auch auch die DNS Server, die im CP konfiguriert werden, im IP-Netzwerk erreichbar sein.

Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
...				
MAIL_ADDR_PARAM (Seite 3675)	Input	VARIANT	D	Für den Sendevorgang notwendige Daten (u. a. die Verbindungsparameter, die Adresse des E-Mail-Servers und der verwendete TCP Port) Verwenden Sie dafür einen der folgenden Systemdatentypen: • TMail_V4_SEC • TMail_V6_SEC • TMail_QDN_SEC
...				

Diagnosemöglichkeiten beim Auftreten eines Fehlers

Beim Auftreten eines Fehlers haben Sie die folgenden beiden Diagnosemöglichkeiten:

- Auswertung des Parametes STATUS
  Es können dieselben Werte auftreten wie bei den Anweisungs-Versionen < V4.0. Lediglich der Wert W#16#8015 erfährt eine Erweiterung. Siehe Parameter STATUS (Seite 3682)
- Auswertung der Variable MS_STATUS in den Instanz-Daten von TMAIL_C. Diese Variable ist in der Version V4.0 von TMAIL_C hinzugekommen. Sie enthält sowohl die Protokoll-Codes als auch ihre zugehörigen textuellen Beschreibungen. Die Protokoll- Codes sind in RFC 5321 "Simple Mail Transfer Protocol" definiert. Ihre zugehörigen textuellen Beschreibungen finden Sie ebenda oder in den zugehörigen Erweiterungen oder über andere Medien.
  Beispiele für Diagnoseinformationen in MS_STATUS:
  - 550 SMTP AUTH required before submission
  - 550 StartTLS required
  - 550 Requested action not taken

Beschreibung TMAIL_C ab Version V5.0

Einleitung

Im Folgenden wird die Anweisung TMAIL_C ab Version V5.0 beschrieben. Dabei wird nicht auf alle Einzelheiten eingegangen, sondern es werden nur die Unterschiede zu den Anweisungs-Versionen < V5.0 erläutert.

Beschreibung

Über die Anweisung TMAIL_C können Sie ab Version V5.0 zusätzlich mittels Secure Communication eine E-Mail über eine integrierte Ethernet-Schnittstelle Ihrer S7-1500-CPU versenden. Ihre S7-1500-CPU benötigt dafür mindestens den Firmware-Stand V2.5. Die Festlegung der für den Sendevorgang notwendigen Daten erfolgt dabei über einen der Systemdatentypen TMAIL_V4_SEC oder TMAIL_QDN_SEC am Parameter MAIL_ADDR_PARAM.

Bei Verwenden des Systemdatentyps TMAIL_V4 können Sie über eine integrierte Ethernet-Schnittstelle Ihrer S7-1500-CPU E-Mails nur ungesichert versenden.

Die Systemdatentypen TMAIL_FQDN und TMAIL_V6 werden beim Versenden von E-Mails über eine integrierte Ethernet-Schnittstelle Ihrer S7-1500-CPU nicht unterstützt.

Hinweis

S7-1500 Software Controller

Die Verwendung der Systemdatentypen TMAIL_V4_SEC und TMAIL_QDN_SEC ist beim S7-1500 Software Controller über ID=59 nicht möglich.

Parameter STATUS

Ab Anweisungs-Version V5.0 von TMAIL_C gibt es zusätzlich den folgenden Rückgabewert am Parameter STATUS:

STATUS (W#16#...)	Erläuterung	Hinweise
8009	Fehler in interner Funktion	Eine interne Funktion hat einen Fehler zurückgeliefert. Detaillierte Information dazu finden Sie im Parameter SFB_STATUS des Instanz-DB. Dessen mögliche Werte sind im Folgenden beschrieben.

Die folgende Tabelle zeigt die möglichen Werte am Parameter SFB_STATUS des Instanz-DB:

Parameter SFB_STATUS des Instanz-DB(W#16#...)	Erläuterung
8090	WatchDog-Zeit muss größer als Null sein.
8092	Die Parameter TO_S und CC sind leer oder der Subparameter From ist leer oder unvollständig.
8093	Der Parameter MAIL_ADDR_PARAM gibt vor, die Verbindung zu einer sicheren Verbindung hochzurüsten, aber der Mailserver unterstützt das STARTTLS-Kommando nicht.
8085	Die Verbindungs-ID (Parameter ID) wird bereits von einer konfigurierten Verbindung verwendet.
8086	Der Parameter ID liegt außerhalb des zulässigen Bereichs.
8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
8095	Ungültige Antwort vom Mailserver. Möglicherweise ist der Mailserver nicht RFC-konform.
809A	Die SDT-Struktur am Parameter MAIL_ADDR_PARAM wird an einer integrierten Schnittstelle nicht unterstützt.
809B	Ungültige Interface-ID im SDT an MAIL_ADDR_PARAM.
80A1	Die angegebene Verbindung oder der remote Port sind bereits belegt.

Parameter SFB_STATUS des Instanz-DB(W#16#...)	Erläuterung
80A3	ID wird von einer durch das Anwenderprogramm erstellten Verbindung genutzt.
80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig oder sie entspricht der IP-Adresse des lokalen Partners.
80A7	Kommunikationsfehler: Sie haben TDISCON ausgeführt, bevor TMAIL_C beendet war.
80B7	Remoter Port ist 0 oder IP-Adresse des Partner-Endpunkts wurde auf 0.0.0.0 gesetzt.
80C3	Zu wenig Ressourcen in der CPU
80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter. • Die projektierte Verbindung wird gerade von einer Anweisung TDISCON entfernt.
80C5	Der Mailserver verweigert den Verbindungsaufbau, hat die Verbindung abgebaut oder aktiv beendet.
80C6	Der Verbindungspartner kann nicht erreicht werden (Netzwerkfehler).
80C7	Zeitüberschreitung der Ausführung
80C8	Es wird versucht, eine bestehende Verbindung erneut einzurichten.
80C9	Validierung des Verbindungspartners fehlgeschlagen. Der Mailserver entspricht nicht dem definierten Partner am Parameter MailServerAddress.
80CE	Die IP-Adresse des lokalen Interfaces ist 0.0.0.0.
80D0	Der Parameter MailServerAddress enthält eine leere Zeichenkette beim Versuch, DNS zu benutzen.
80D1	Der Parameter MailServerAddress ist kein voll qualifizierter Domainname. Möglicherweise fehlt der Punkt am Ende.
80D2	Es ist keine DNS-Server-Adresse konfiguriert.
80D3	Der voll qualifizierte Domänen-Name konnte nicht aufgelöst werden. Mögliche Ursachen: • Der DNS-Server ist nicht erreichbar z. B. weil er heruntergefahren ist oder der remote Port nicht erreichbar ist. • Während der Kommunikation mit dem DNS-Server ist ein Fehler aufgetreten. • Der DNS-Server schickte eine gültige DNS-Antwort zurück, aber die Antwort enthielt keine IPv4-Adresse.
80E0	Kommunikation mit dem MailServer aufgrund einer fehlerhaften Nachricht fehlgeschlagen. Mögliche Ursachen: • Ungültiger Nachrichtenauthentifizierungscode • Decodierung einer Nachricht fehlgeschlagen. • Fehler beim Dekomprimieren einer Nachricht • Interner Aufzeichnungsüberlauf
80E1	Fehler während des Handshakes. Mögliche Ursachen: • Abbruch durch Benutzer • Security nicht hoch genug • Erneute Verhandlung wird nicht unterstützt • SSL/TLS-Version wird nicht unterstützt. • Entschlüsselungsfehler

Parameter SFB_STATUS des Instanz-DB(W#16#...)	Erläuterung
80E2	Nicht unterstütztes / Ungültiges Zertifikat Mögliche Ursache: Die Uhrzeit der betroffenen Baugruppe ist nicht gestellt bzw. die Baugruppe wird nicht synchronisiert. Beispiel: Das Datum der Baugruppe ist auf den 1.1.2012 voreingestellt und wurde bei der Inbetriebnahme nicht eingestellt. Der Gültigkeitszeitraum des Zertifikats beginnt am 20.08.2016 und endet am 20.08.2024. In diesem Fall liegt das Datum der Baugruppe außerhalb des Gültigkeitszeitraums des Zertifikats; für die Baugruppe ist das Zertifikat ungültig
80E3	Mailserver-Zertifikat wurde verworfen
80E4	Für das Mailserver-Zertifikat wurde keine gültige Zertifizierungsstelle gefunden.
80E5	Mailserver-Zertifikat abgelaufen
80E6	Integritätsfehler im Transport-Layer-Security-Protokoll
80E7	Nicht unterstützte Erweiterung im Mailserver-Zertifikat
80E9	TLS-Server ohne Server-Zertifikat wird nicht unterstützt

Erweiterter Wertebereich der Verbindungsparameter InterfaceID und ID

Beim gesicherten Versenden einer E-Mail über eine integrierte Schnittstelle Ihrer S7-1500-CPU gilt für die Verbindungsparameter InterfaceID und ID in den SDTs TMAIL_V4_SEC und TMAIL_QDN_SEC:

Parameter	Beschreibung
InterfaceID	Hardware-Kennung der Schnittstelle. Wertebereich: • 0 (neu): Das Betriebssystem wählt selbstständig die geeignete integrierte Schnittstelle aus. • Hardware-Kennung der zu verwendenden integrierten Schnittstelle
ID	Referenz auf die Verbindung: Wertebereich: • 0 (neu): Das Betriebssystem wählt selbstständig eine freie Verbindungs-ID aus dem internen Bereich aus. • 1 bis 4095: die zu verwendende Verbindungs-ID

Parameter TO_S und CC

Beschreibung

Die Parameter TO_S und CC sind Strings mit z. B. folgendem Inhalt:

- <wenna@mydomain.com>, <ruby@mydomain.com>
- <admin@mydomain.com>, <judy@mydomain.com>

Folgende Regeln sind bei der Eingabe der Parameter zu beachten:

- Vor jeder Adresse muss ein Leerzeichen und eine spitze Klammer auf "<" eingegeben werden.
- Nach jeder Adresse muss eine spitze Klammer zu ">" eingegeben werden.
- Zwischen den Adressen in TO und CC muss ein Komma eingegeben werden.

Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "TMAIL_C" keine Syntaxprüfung der Parameter TO_S und CC durch.

Parameter MAIL_ADDR_PARAM

Beschreibung

Am Parameter MAIL_ADDR_PARAM definieren Sie in der Struktur TMail_V4, TMail_V6 oder Tmail_FQDN, über welche Verbindung die E-Mail gesendet werden soll und hinterlegen die Adresse des E-Mail-Servers sowie die Login-Daten.

Je nachdem, in welchem Format Sie den E-Mail-Server adressieren möchten verwenden Sie am Parameter MAIL_ADDR_PARAM die Struktur:

- TMail_V4: Adressierung über die IP-Adresse nach IPv4.
- TMail_V6: Adressierung über die IP-Adresse nach IPv6.
- TMail_FQDN: Adressierung über voll qualifizierten Domainnamen (FQDN).

Welche Struktur Sie verwenden können, ist abhängig von der am Parameter InterfaceId adressierten Schnittstelle:

- Zur Verwendung der Anweisung "TMAIL_C" mit der internen Schnittstelle muss am Parameter MAIL_ADDR_PARAM die Struktur TMail_V4 verwendet werden.
- Bei der Verwendung eines Kommunikationsprozessors (CP) oder Kommunikationsmoduls (CM) können Sie alle drei Adressierungsmöglichkeiten (IPv4, IPv6 und FQDN) verwendet werden.

Tabelle 4-70 **TMail_V4: Adressieren des Mailservers über die IP-Adresse nach IPv4**

Parameter	Datentyp	Beschreibung
TMail_V4	Struct	
InterfaceId	LADDR	Hardware-Kennung der Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für IPv4 als Verbindungstyp 16#20.
ActiveEstablished	BOOL	Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.
CertIndex	BYTE	• =0: Verwendung von SMTP (**S**imple **M**ail **T**ransfer **P**rotocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden. • ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs).
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerAddress	IP_V4	IP Adresse des Mailservers. Nach IPv4 im Format XXX.XXX.XXX.XXX (dezimal). Beispiel: 192.142.131.237.
UserName	STRING[254]	Loginname des Mailservers
PassWord	STRING[254]	Passwort des Mailservers
From	EMAIL_ADDR	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com".

Tabelle 4-71 **TMail_V6: Adressieren des Mailservers über die IP-Adresse nach IPv6**

Parameter	Datentyp	Beschreibung
TMail_V6	Struct	
InterfaceId	LADDR	Hardware-Kennung der Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für IPv6 als Verbindungstyp 16#21.
ActiveEstablished	BOOL	Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.
CertIndex	BYTE	• =0: Verwendung von SMTP (**S**imple **M**ail **T**ransfer **P**rotocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden. • ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs). Über den Parameter CertIndex geben Sie das zu verwendende Zertifikat an (siehe "Projektnavigation" > "Globale Security-Einstellungen" > "Zertifikatsmanager").
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerAddress	IP_V6	IP Adresse des Mailservers nach IPv6 im Format XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX (hexadezimal). Die Adresse ist in 8 Blöcke von jeweils 2 Byte unterteilt (insgesamt 16 Byte). Beispiel: 2001:db8:1f11:08d3:290:27ff:0370:2093
UserName	STRING[254]	Loginname des Mailservers
PassWord	STRING[254]	Passwort des Mailservers
From	EMAIL_ADDR	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com".

Tabelle 4-72 **TMail_FQDN: Adressieren des Mailservers über den FQDN**

Parameter	Datentyp	Beschreibung
TMail_FQDN	Struct	
InterfaceId	LADDR	Hardware-Kennung der Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für FQDN als Verbindungstyp 16#22.
ActiveEstablished	BOOL	Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.
CertIndex	BYTE	• =0: Verwendung von SMTP (Simple Mail Transfer Protocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden. • ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs). Über den Parameter CertIndex geben Sie das zu verwendende Zertifikat an (siehe "Projektnavigation" > "Globale Security-Einstellungen" > "Zertifikatsmanager").
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerAddress	STRING[254]	FQDN (Fully Qualified Domain Name) des Mailservers. Die Adressierung des Mailservers erfolgt über den vollständigen Namen der Domain. Beispiel: "www.mymailserver.com.".
UserName	STRING[254]	Loginname des Mailservers
PassWord	STRING[254]	Passwort des Mailservers
From	Struct	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com".

Parameter MAIL_ADDR_PARAM ab Version 4.0 von TMAIL_C

Beschreibung

Über den Parameter MAIL_ADDR_PARAM geben Sie die für den Sendevorgang notwendigen Daten vor, z. B. die Verbindungsparameter, die Adresse des E-Mail-Servers und den verwendeten TCP-Port.

Je nachdem, in welchem Format Sie den E-Mail-Server adressieren möchten, verwenden Sie am Parameter MAIL_ADDR_PARAM eine der folgenden Strukturen:

- TMail_V4_SEC: Adressierung über die IP-Adresse im Format IPv4
- TMail_V6_SEC: Adressierung über die IP-Adresse im Format IPv6
- TMail_QDN_SEC: Adressierung über den voll qualifizierten Host-Namen (FQDN)

TMail_V4_SEC: Adressieren des Mail Servers über die IP-Adresse im Format IPv4

Parameter	Datentyp	Beschreibung
TMail_V4_SEC	Struct	
InterfaceId	LADDR	Hardware-Kennung der Ethernet-Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für IPv4 als Verbindungstyp 16#20.
ActiveEstablishment	BOOL	Aktiver / passiver Verbindungsaufbau. Da der CP immer der SMTP Client ist, muss dieser Parameter auf "1" gesetzt werden.
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WatchDogTime berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerAddress	IP_V4	IP-Adresse des Mailservers im Format IPv4: XXX.XXX.XXX.XXX (dezimal). Beispiel: 192.142.131.237.
UserName	STRING[254]	Benutzername. Dieser wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
PassWord	STRING[254]	Benutzer-Passwort. Dies wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
From	EMAIL_ADDR	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymail-server.com".
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".

Parameter	Datentyp	Beschreibung
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailser-ver.com".
RemotePort	UINT	TCP Port des Mail Servers
ActivateSecureConn	BOOL	• 1: secure SMTP-Verbindung • 0: SMTP-Verbindung (non secure). In diesem Fall sind die nachfolgenden Parameter irrelevant.
ExtTLSCapabilities	BYTE	derzeit nicht benutzt
TLSServerCertRef	UDINT	Referenz auf das X.509 V3 (CA)-Zertifikat des Mail Servers, welches vom TLS Client benutzt wird, um die Authentifizierung des TLS Servers zu validieren.

TMail_V6_SEC: Adressieren des Mail Servers über die IP-Adresse im Format IPv6

Parameter	Datentyp	Beschreibung
TMail_V6_SEC	Struct	
InterfaceId	LADDR	Hardware-Kennung der Ethernet-Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für IPv6 als Verbindungstyp 16#21.
ActiveEstablishment	BOOL	Aktiver / passiver Verbindungsaufbau. Da der CP immer der SMTP Client ist, muss dieser Parameter auf "1" gesetzt werden.
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WatchDogTime berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerAddress	IP_V6	IP-Adresse des Mailservers im Format IPv6: XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX (hexadezimal). Die Adresse ist in 8 Blöcke von jeweils 2 Byte unterteilt (insgesamt 16 Byte) Beispiel: 2001:db8:1f11:08d3:290:27ff:0370:2093
UserName	STRING[254]	Benutzername. Dieser wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
PassWord	STRING[254]	Benutzer-Passwort. Dies wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
From	EMAIL_ADDR	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com".

Parameter	Datentyp	Beschreibung
RemotePort	UINT	TCP Port des Mail Servers
ActivateSecureConn	BOOL	• 1: secure SMTP-Verbindung • 0: SMTP-Verbindung (non secure). In diesem Fall sind die nachfolgenden Parameter irrelevant.
ExtTLSCapabilities	BYTE	derzeit nicht benutzt
TLSServerCertRef	UDINT	Referenz auf das X.509 V3 (CA)-Zertifikat des Mail Servers, welches vom TLS Client benutzt wird, um die Authentifizierung des TLS Servers zu validieren.

TMail_QDN_SEC: Adressieren des Mail Servers über FQDN

Parameter	Datentyp	Beschreibung
TMail_QDN_SEC	Struct	
InterfaceId	LADDR	Hardware-Kennung der Ethernet-Schnittstelle
ID	CONN_OUC	Verbindungs-ID
ConnectionType	BYTE	Verbindungstyp. Wählen Sie für FQDN als Verbindungstyp 16#22.
ActiveEstablishment	BOOL	Aktiver / passiver Verbindungsaufbau. Da der S7-CP immer der SMTP Client ist, muss dieser Parameter auf "1" gesetzt werden.
WatchDogTime	TIME	Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs. Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WatchDogTime berücksichtigen. Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut.
MailServerQDN	STRING[254]	FQDN (Fully Qualified Domain Name) des Mail Servers. Die Adressierung des Mail Servers erfolgt über den vollständigen Namen der Domäne, der mit "." abgeschlossen sein muss. Beispiel: "www.mymailserver.com"
UserName	STRING[254]	Benutzername Dieser wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
PassWord	STRING[254]	Benutzer-Passwort Dies wird benötigt, wenn der Benutzer auf sein E-Mail-Postfach zugreifen möchte, um sich gegenüber dem E-Mail-Anbieter als Besitzer des E-Mail-Postfachs zu identifizieren.
From	EMAIL_ADDR	Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com"
LocalPartPlusAtSign	STRING[64]	Lokaler Teil der Absenderadresse inklusive @-Zeichen. Beispiel: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com"

Parameter	Datentyp	Beschreibung
RemotePort	UINT	TCP Port des Mail Servers
ActivateSecureConn	BOOL	• 1: Secure SMTP-Verbindung • 0: Ungesicherte SMTP-Verbindung. In diesem Fall sind die nachfolgenden Parameter irrelevant.
ExtTLSCapabilities	BYTE	derzeit nicht benutzt
TLSServerCertRef	UDINT	Referenz auf das X.509 V3 (CA)-Zertifikat des Mail Servers, welches vom TLS-Client benutzt wird, um die Authentifizierung des TLS-Servers zu validieren.

Parameter DONE, BUSY und ERROR

Beschreibung

Die Ausgangsparameter DONE, BUSY und ERROR werden jeweils nur einen Zyklus lang angezeigt, wenn der Zustand des Ausgangsparameters BUSY von "1" auf "0" wechselt.

In der folgenden Tabelle ist der Zusammenhang zwischen DONE, BUSY und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung "TMAIL_C" aktuell befindet bzw. wann das Senden der E-Mail beendet ist.

DONE	BUSY	ER-ROR	Beschreibung
0	1	0	Der Auftrag ist in Bearbeitung.
1	0	0	Der Auftrag wurde erfolgreich abgeschlossen.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache entnehmen Sie dem Parameter STATUS (Seite 3682).
0	0	0	Der Anweisung "TMAIL_C" wurde kein (neuer) Auftrag erteilt.

Parameter STATUS

Beschreibung

Die folgende Tabelle zeigt die Rückgabewerte am Parameter STATUS von "TMAIL_C":

Rückgabewert STATUS* (W#16#...):	Erläuterung	Hinweise
0000	Die Bearbeitung von "TMAIL_C" wurde fehlerfrei abgeschlossen.	Ein fehlerfreies Beenden von "TMAIL_C" bedeutet nicht, dass die versendete E-Mail ankommt. Eine fehlerhafte Eingabe der Empfängeradressen erzeugt keinen Statusfehler der Anweisung "TMAIL_C". Das Versenden der E-Mail an weitere Empfänger, auch wenn diese richtig eingegeben wurden, ist in diesem Fall nicht sichergestellt.
7001	"TMAIL_C" ist aktiv (BUSY = 1).	Erstaufruf: Auftrag angestoßen.
7002	"TMAIL_C" ist aktiv (BUSY = 1).	Zwischenaufruf: Auftrag ist bereits aktiv.
8xxx	Die Bearbeitung von "TMAIL_C" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen abgeschlossen.	Ausführliche Informationen finden Sie in den Beschreibungen des Parameters STATUS der Kommunikations-Anweisungen "TCON (Seite 3692)", "TDISCON (Seite 3700)", "TSEND (Seite 3705)" und "TRCV (Seite 3709)".
8010	Fehler beim Verbindungsaufbau	Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TCON (Seite 3692)" enthalten.
8011	Fehler beim Senden der Daten	Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TSEND (Seite 3705)" enthalten.
8012	Fehler beim Empfangen der Daten	Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TRCV (Seite 3709)" enthalten.
8013	Fehler beim Verbindungsaufbau	Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisungen "TCON (Seite 3692)" und "TDISCON (Seite 3700)" enthalten.

Rückgabewert STATUS* (W#16#...):	Erläuterung	Hinweise
8014	Aufbau einer Verbindung nicht möglich.	Möglicherweise haben Sie eine fehlerhafte Mailserver-IP-Adresse (MailServerAddress (Seite 3672)) oder eine zu kleine Zeitspanne (WatchDogTime (Seite 3672)) für den Verbindungsaufbau eingegeben. Es besteht auch die Möglichkeit, dass die CPU keine Verbindung zum Netzwerk hat oder die CPU-Konfiguration fehlerhaft ist.
8015	Falscher Datentyp für MAIL_ADDR_PARAM	Erlaubte Datentypen sind die Systemdatentypen (Strukturen) TMail_V4, TMail_V6 und TMail_FQDN. Ab der Anweisungs-Version V4.0 sind zusätzlich die folgenden Systemdatentypen erlaubt: TMail_V4_SEC, TMail_V6_SEC und TMail_QDN_SEC
8016	Falscher Datentyp für den Parameter ATTACHMENT	Erlaubte Datentypen sind ArrayOfByte, ArrayOfWord und ArrayOfDWord.
8017	Falsche Datenlänge für den Parameter ATTACHMENT	Erlaubte Datenlänge beträgt <= 65534 Byte.
8401	Keine Kanäle verfügbar. Mögliche Ursache: Es besteht bereits eine E-Mail-Verbindung über den CP. Eine zweite Verbindung kann nicht parallel eingerichtet werden.	Spezifischer Fehler für CP 1543
8403	Es konnte keine TCP/IP-Verbindung zum Mailserver aufgebaut werden.	Spezifischer Fehler für CP 1543
8405	Server hat Login-Anfrage verweigert.	Spezifischer Fehler für CP 1543
8406	Ein interner SSL-Fehler oder ein Problem mit der Struktur des Zertifikats wurde durch den SMTP-Client festgestellt.	Spezifischer Fehler für CP 1543
8407	Anfrage zur Verwendung von SSL wurde verweigert.	Spezifischer Fehler für CP 1543
8408	Der Client konnte kein Socket zur Erstellung einer TCP/IP-Verbindung zum Mailserver ermitteln.	Spezifischer Fehler für CP 1543
8409	Über die Verbindung kann nicht geschrieben werden. Mögliche Ursache: Durch den Kommunikationspartner wurde ein Reset der Verbindung durchgeführt oder die Verbindung wurde abgebrochen.	Spezifischer Fehler für CP 1543
8410	Über die Verbindung kann nicht gelesen werden. Mögliche Ursache: Durch den Kommunikationspartner wurde ein Reset der Verbindung durchgeführt oder die Verbindung wurde abgebrochen.	Spezifischer Fehler für CP 1543

Rückgabewert STATUS* (W#16#...):	Erläuterung	Hinweise
8411	Senden der E-Mail fehlgeschlagen. Ursache: Speicherplatz war nicht ausreichend, um den Sendevorgang durchzuführen.	Spezifischer Fehler für CP 1543
8412	Konfigurierter DNS-Server konnte den angegebenen Domainnamen nicht auflösen.	Spezifischer Fehler für CP 1543
8413	Aufgrund eines internen Fehlers im DNS-Subsystem konnte der Domainname nicht aufgelöst werden.	Spezifischer Fehler für CP 1543
8414	Als Domainname wurde eine leere Zeichenkette angegeben.	Spezifischer Fehler für CP 1543
8415	Ein interner Fehler ist im Curl-Modul aufgetreten. Ausführung wurde gestoppt.	Spezifischer Fehler für CP 1543
8416	Ein interner Fehler ist im SMTP-Modul aufgetreten. Ausführung wurde gestoppt.	Spezifischer Fehler für CP 1543
8417	Anfrage an SMTP auf bereits verwendetem Kanal oder ungültige Kanal-ID. Ausführung wurde gestoppt.	Spezifischer Fehler für CP 1543
8418	Senden der E-Mail wurde abgebrochen. Mögliche Ursachen: Überschreitung der Ausführungszeit (Parameter WatchDogTime) oder Start/Stopp-Übergang des CPs.	Spezifischer Fehler für CP 1543
8419	Der Kanal wurde unterbrochen und kann nicht verwendet werden, bevor die Verbindung geschlossen wird.	Spezifischer Fehler für CP 1543
8420	Zertifikatskette von dem Server konnte mit dem Root-Zertifikat des CPs nicht verifiziert werden.	Spezifischer Fehler für CP 1543
8421	Interner Fehler aufgetreten. Ausführung wurde gestoppt.	Spezifischer Fehler für CP 1543
82xx, 84xx, bzw. 85xx	Die Fehlermeldung stammt vom Mailserver und entspricht bis auf die "8" der Fehlernummer des SMTP-Protokolls. In den nachfolgenden Zeilen sind einige Fehlercodes aufgeführt, die auftreten können.	Weitere Informationen zum SMTP-Fehlercode und weitere Fehlercodes finden Sie im SMTP-Protokoll im Internet bzw. in der Fehlerdokumentation des Mailservers. Darüber hinaus können Sie die letzte, vom Mailserver gesendete Fehlermeldung in Textform in ihrem Instanz-DB im Parameter BUFFER1 einsehen. In dem Instanz-DB finden Sie unter DATEN die zuletzt von der Anweisung "TMAIL_C" versendeten Daten.
8450	Aktion nicht ausgeführt: Mailbox nicht verfügbar/nicht erreichbar	Versuchen Sie es später noch einmal.
8451	Aktion abgebrochen: Lokaler Fehler in der Verarbeitung	Versuchen Sie es später noch einmal.

Rückgabewert STATUS* (W#16#...):	Erläuterung	Hinweise
8500	Syntax-Fehler: Fehler nicht erkannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann auch sein, dass der E-Mail-Server das Authentisierungsverfahren LOGIN nicht unterstützt.	Überprüfen Sie die Parameter von "TMAIL_C". Versuchen Sie, eine E-Mail ohne Authentisierung zu versenden. Ersetzen Sie dazu den Inhalt des Parameters UserName durch einen leeren String. Wird kein Benutzername angegeben, wird das Authentisierungsverfahren LOGIN auch nicht verwendet.
8501	Syntax-Fehler: Falsche Eingabe an einem Parameter	Mögliche Ursache: Fehlerhafte Adresse an den Parametern TO_S oder CC (siehe auch: Parameter TO_S und CC (Seite 3671)).
8502	Befehl unbekannt bzw. nicht implementiert	Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" oder "." vergessen (siehe auch: Parameter TO_S und CC (Seite 3671)).
8504	Befehlsparameter nicht implementiert	Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" vergessen (siehe auch: Parameter TO_S und CC (Seite 3671)).
8535	SMTP-Authentisierung unvollständig	Eventuell haben Sie einen fehlerhaften Benutzernamen oder ein fehlerhaftes Passwort eingegeben.
8550	Mailserver kann nicht erreicht werden. Sie haben keine Zugriffsrechte.	Möglicherweise haben Sie einen fehlerhaften Benutzernamen bzw. Passwort eingegeben oder der Mailserver unterstützt Ihr Login nicht. Eine weitere Fehlerursache könnte eine fehlerhafte Eingabe des Domainnamens nach dem "@" oder ein fehlender "." an den Parametern TO_S, CC und FROM sein (siehe auch: Parameter TO_S und CC (Seite 3671)).
8552	Aktion abgebrochen: Überschreitung der zugewiesenen Speichergröße	Versuchen Sie es später noch einmal.
8554	Übertragung fehlgeschlagen	Versuchen Sie es später noch einmal.
8555	Fehler durch falsche Eingabe der E-Mail Adresse.	Wenn mehrere Adressen eingetragen wurden, müssen diese durch ein Komma getrennt werden.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel: E-Mail versenden mit TMAIL_C

Einleitung

Das folgende Aufrufbeispiel zeigt die Anwendung der Anweisung TMAIL_C.

Voraussetzung

- Für den Versand einer E-Mail muss eine Hardware bereits konfiguriert sein.
- Die Infrastruktur des Netzwerks muss eine Kommunikationsverbindung zu dem Mailserver zulassen.

E-Mail Anhang

Die E-Mail soll mit einem Anhang versendet werden.

Als Quelle für den Anhang legen Sie in einem globalen Datenbaustein ("SLI_gDB_MailAttach_scl") ein Array of Byte mit dem Namen "Data" an. Das Array wird später an dem Eingangsparameter ATTACHMENT verschaltet.

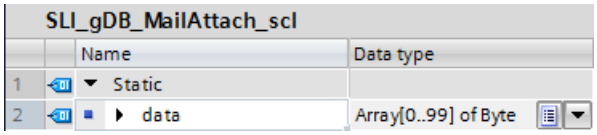

SLI_gDB_MailAttach_scl

		Name	Data type
1		Static	
2		data	Array[0..99] of Byte

Ablage der Daten

Die Verbindungsinformationen der CPU und Adressdaten des Mailservers erfassen Sie in dem Systemdatentyp TMail_V4.

- Erstellen Sie einen globalen Datenbaustein ("SLI_gDB_TMAIL_C_scl").
- In dem Datenbaustein legen Sie eine Variable "Par1" mit dem Datentyp "TMail_V4" an.
- Geben Sie in der folgenden Struktur die Parameter entsprechend der Konfiguration ihrer CPU und den Verbindungsdaten ihres Mailservers ein. Weitere Informationen finden Sie in der Beschreibung des Eingangsparameters MAIL_ADDR_PARAM (Seite 3672).

Die Variable "Par1" wird später an dem Eingangsparameter MAIL_ADDR_PARAM verschaltet.

SLI_gDB_TMAIL_C_scl			
	Name	Data type	Start value
1	Static		
2	par1	TMail_V4	
3	InterfaceId	HW_ANY	64
4	ID	CONN_OUC	100
5	ConnectionType	Byte	16#20
6	ActiveEstablished	Bool	false
7	CertIndex	Byte	16#0
8	WatchDogTime	Time	T#5s
9	MailServerAddress	IP_V4	
10	ADDR	Array[1..4] of Byte	
11	ADDR[1]	Byte	192
12	ADDR[2]	Byte	168
13	ADDR[3]	Byte	100
14	ADDR[4]	Byte	10
15	UserName	String[254]	'myusername'
16	PassWord	String[254]	'mypassword'
17	From	EMAIL_ADDR	
18	LocalPartPlusAtSign	String[64]	'station1@'
19	FullQualifiedDomainName	String[254]	'mycpu.com'

Legen Sie zudem in dem globalen Datenbaustein die folgenden Variablen an.

3	sendMail	Bool	false
4	target	String[240]	' <wenna@mydomain.com>,'
5	subject	String[240]	'Testmail'
6	text	String[240]	'The cpu switched to run'
7	busy	Bool	false
8	done	Bool	false
9	error	Bool	false
10	memErrStatus	Word	16#0

Aufruf der Anweisung

Erstellen Sie einen neuen Funktionsbaustein in der Sprache SCL mit folgenden Schnittstellen:

7	Static		
8	status	Word	16#0
9	TMAIL_C_Instance	TMAIL_C	
10	Temp		
11	done	Bool	

Kopieren Sie den folgenden Quellcode mit dem Aufruf von TMAIL_C in das Programmierfenster:

Aufruf TMAIL_C

```
//Set "sendMail" to "TRUE" to start TMAIL_C.
#TMAIL_C_Instance(REQ:="SLI_gDB_TMAIL_C_scl".sendMail AND NOT("SLI_gDB_TMAIL_C_scl".busy),
            TO_S:="SLI_gDB_TMAIL_C_scl".target,
            SUBJECT:="SLI_gDB_TMAIL_C_scl".subject,
            TEXT:="SLI_gDB_TMAIL_C_scl".text,
            ATTACHMENT:="SLI_gDB_MailAttach_scl".data,
            MAIL_ADDR_PARAM:="SLI_gDB_TMAIL_C_scl".par1,
            DONE=>#done,
            BUSY=>"SLI_gDB_TMAIL_C_scl".busy,
            ERROR=>"SLI_gDB_TMAIL_C_scl".error,
            STATUS=>#status);
//INFO: Saves the success status of TMAIL_C.
IF #done THEN
   "SLI_gDB_TMAIL_C_scl".done := TRUE;
   "SLI_gDB_TMAIL_C_scl".sendMail := FALSE;
END_IF;
//INFO: In case of an error, saves the error status of TMAIL_C and resets "sendMail".
IF "SLI_gDB_TMAIL_C_scl".error THEN
   "SLI_gDB_TMAIL_C_scl".memErrStatus := #status;
   "SLI_gDB_TMAIL_C_scl".sendMail := FALSE;
END_IF;
```

Ergebnis

Wenn die Variable "sendMail" den Signalzustand "TRUE" liefert (Flankenwechsel von "0" auf "1" am Parameter REQ), wird die Anweisung "TMAIL_C" ausgeführt. Sobald die Variable "#done" den Signalzustand "TRUE" liefert, wird die Variable "sendMail" automatisch zurückgesetzt in den Signalzustand "FALSE". Außerdem wird der Erfolg beim Versenden der E-Mail ("#done" ist "TRUE") in der Variable "done" gespeichert.

Im Fehlerfall der Anweisung "TMAIL_C" wird der Wert des Parameters STATUS ("#status") in der Variable "memErrStatus" gespeichert und die Variable "sendMail" zurückgesetzt.

Siehe auch

Beschreibung TMAIL_C (Seite 3662)

Weitere

TCON: Kommunikationsverbindung aufbauen

TCON: Kommunikationsverbindung aufbauen

Gültigkeit

Die folgende Beschreibung der Anweisung "TCON" gilt für S7-1200-CPUs mit Firmware-Version < V4.0.

Beschreibung

Mit der Anweisung "TCON" richten Sie eine Kommunikationsverbindung ein und bauen diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. "TCON" wird asynchron ausgeführt.

Zum Einrichten der Kommunikationsverbindung werden die am Parameter CONNECT und ID angegebenen Verbindungsdaten verwendet. Um die Verbindung aufzubauen, muss am Parameter REQ eine steigende Flanke erfasst werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP oder ISO-on-TCP anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Anzahl möglicher Verbindungen

Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.

TCP- und ISO-on-TCP-Verbindungen

Beide Kommunikationspartner rufen die Anweisung "TCON" zum Einrichten und Aufbauen der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationsendpunkt ist.

Bei Verbindungsabbruch durch z. B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner, die eingerichtete Verbindung wieder aufzubauen. Sie müssen "TCON" nicht erneut aufrufen. Dies gilt jedoch nur, wenn "TCON" einmal erfolgreich durchlaufen wurde (DONE = 1).

Mit der Ausführung der Anweisung "TDISCON (Seite 3700)" oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung müssen Sie "TCON" erneut ausführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TCON":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Startet den Auftrag zum Aufbauen der in der ID angegebenen Verbindung bei einer steigenden Flanke.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
CONNECT	InOut	TCON_Param	D	Zeiger auf die Verbindungsbeschreibung Siehe auch: AUTOHOTSPOT
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter ERROR: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob einen Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TCON" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	0	0	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.

BUSY	DONE	ERROR	Beschreibung
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Verbindung wurde erfolgreich aufgebaut.
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Auftragsbearbeitung starten, Verbindung aufbauen.
0	7002	Verbindung wird aufgebaut (REQ irrelevant).
1	8085	Die Verbindungs-ID (Parameter ID) wird bereits von einer konfigurierten Verbindung verwendet.
1	8086	Der Parameter ID liegt außerhalb des zulässigen Bereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
1	8089	Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.
1	809A	Die Struktur am Parameter CONNECT wird an einer integrierten Schnittstelle nicht unterstützt oder die Länge ist ungültig oder es wird eine falsche "InterfaceID" in der Verbindungsbeschreibung (SDT) angegeben.
1	809B	Das Element InterfaceId innerhalb der TCON_xxx Struktur referenziert keine Hardware-Kennung einer CPU- oder CM/CP Schnittstelle oder hat den Wert "0".
1	80A0	Sammelfehler für Fehlercodes W#16#80A1 und W#16#80A2
1	80A1	Die angegebene Verbindung oder der Port sind bereits belegt.
1	80A2	Lokaler oder remoter Port wird vom System verwendet.
1	80A3	Es wird versucht, eine vorhandene Verbindung erneut aufzubauen.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.
1	80A5	Verbindungs-ID wird bereits verwendet.
1	80A7	Kommunikationsfehler: Sie haben "TDISCON (Seite 3700)" ausgeführt, bevor "TCON" beendet war.
1	80B2	Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.
1	80B4	Sie haben beim passiven Aufbauen einer Verbindung mit der Protokollvariante ISO on TCP (connection_type = B#16#12) eine oder mehrere der folgenden Bedingungen verletzt: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0 • Bei local_tsap_id_len >= B#16#03 ist local_tsap_id[1] ein ASCII-Zeichen. • local_tsap_id[1] ist ein ASCII-Zeichen und local_tsap_id_len >= B#16#03.
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.
1	80B6	Parametrierungsfehler im Parameter connection_type des SDTs TCON_Param.
1	80B7	Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len. Hinweis: Wenn Sie TCON bei TCP für die passive Seite aufrufen, muss local_tsap_id_len den Wert 2 und rem_tsap_id_len den Wert 0 haben.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80B8	Verbindungsbeschreibung des Strukturelements ID und der Bausteinparameter ID sind unterschiedlich.
1	80C3	Alle Verbindungsressourcen sind belegt.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 3700)" entfernt.
* Die Fehlercodes im Programmiereditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

TCON: Kommunikationsverbindung aufbauen

Beschreibung

Mit der Anweisung "TCON" richten Sie eine Kommunikationsverbindung ein und bauen diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. "TCON" wird asynchron ausgeführt.

Bei S7-1500-CPUs ab Firmware-Stand V2.0 können Sie mit der Anweisung "TCON" über die integrierten PROFINET-Schnittstellen auch einen Zugangspunkt für IPv4-Multicast-Kommunikation einrichten.

Zum Einrichten der Kommunikationsverbindung werden die am Parameter CONNECT und ID angegebenen Verbindungsdaten verwendet. Nutzen Sie am Parameter CONNECT nach Möglichkeit nur vordefinierte Strukturen, wie sie durch die Verbindungsparametrierung im Inspektorfenster des Programmiereditors erstellt werden.

Ab dem Firmware-Stand V2.0 der S7-1500-CPUs können Sie für bestimmte TCON_xxx-Strukturen am Parameter InterfaceId den Wert 0 angeben. Damit überlassen Sie der Firmware die Auswahl der zu verwendenden Industrial-Ethernet-Schnittstelle. Diese Vorgabe des Werts 0 ist bei folgenden SDTs möglich: TCON_IP_V4_SEC, TCON_QDN, TCON_QDN_SEC, TMail_QDN_SEC, TMail_V4 und TMAIL_V4_SEC. Bei allen anderen SDTs müssen Sie InterfaceId einen festen Wert zuweisen.

Um die Verbindung aufzubauen, muss am Parameter REQ eine steigende Flanke erfasst werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.

Hinweis

Unterstützung bei Verbindungsprogrammierung

Wenn Sie eine Anweisung zur Kommunikation TCON, TSEND_C oder TRCV_C in einem Programmbaustein auswählen und Verbindungen vom Typ TCP, UDP, ISO-on-TCP oder FDL anlegen und parametrieren möchten, können Sie die Unterstützung der Verbindungsparametrierung nutzen.

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors.

Anzahl möglicher Verbindungen

Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.

Verbindung bei TCP und ISO-on-TCP

Beide Kommunikationspartner rufen die Anweisung "TCON" zum Einrichten und Aufbauen der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationsendpunkt ist.

Bei Verbindungsabbruch durch z. B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner, die eingerichtete Verbindung wieder aufzubauen. Sie müssen "TCON" nicht erneut aufrufen. Dies gilt jedoch nur, wenn "TCON" einmal erfolgreich durchlaufen wurde (DONE = 1).

Mit der Ausführung der Anweisung "TDISCON (Seite 3700)" oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung müssen Sie "TCON" erneut ausführen.

Telecontrol-Verbindungen zwischen CP und SMS-Client

Verwenden Sie am Parameter CONNECT für die Verbindungsbeschreibung den Systemdatentyp TCON_PHONE.

Für diesen Verbindungstyp benötigt die Station Zugang zum Mobilfunknetz über einen Mobilfunk-CP.

FDL-Verbindungen

Zum Einrichten von FDL-Verbindungen des CM 1542-5 siehe AUTOHOTSPOT.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TCON":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, oder Konstante	Startet den Auftrag zum Aufbauen der Verbindung bei einer steigenden Flanke.
ID	Input	CONN_OUC	E, A, M, D, L, oder Konstante	Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
CONNECT	InOut	VARIANT	D	Zeiger auf die Verbindungsbeschreibung • Bei TCP oder UDP verwenden Sie die Struktur TCON_IP_v4 oder TCON_QDN Zur Beschreibung siehe: AUTOHOTSPOT bzw. AUTOHOTSPOT • Bei TCP oder UDP mittels secure communication verwenden Sie die Struktur TCON_IP_V4_SEC oder TCON_QDN_SEC Zur Beschreibung siehe: AUTOHOTSPOT bzw. AUTOHOTSPOT • Bei ISO-on-TCP verwenden Sie die Struktur TCON_IP_RFC Zur Beschreibung siehe: AUTOHOTSPOT • Bei ISO-Verbindungen des CP 1543-1 verwenden Sie die Struktur TCON_ISOnative. Zur Beschreibung siehe: "Aufbau der Verbindungsbeschreibung nach TCON_ISOnative" • Bei Verbindungen zu SMS-Clients verwenden Sie den Systemdatentyp TCON_PHONE. Zur Beschreibung siehe AUTOHOTSPOT. • Für FDL-Verbindungen des CM 1542-5 verwenden Sie den Systemdatentyp TCON_FDL, siehe AUTOHOTSPOT.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter ERROR: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob einen Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TCON" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die Anweisung "TCON" erzeugt in Version 3.0 eine Fehlermeldung, wenn ein aktiver Verbindungsaufbau zu einem remoten Partner scheitert. Um einen erneuten Verbindungsaufbau durchzuführen, erzeugen Sie am Parameter REQ eine steigende Flanke.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	0	0	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Aufbau der Verbindungsbeschreibung nach TCON_ISOnative

Um die Kommunikationsverbindungen bei ISO zu parametrieren, wird ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_ISOnative verwendet. Die feste Datenstruktur des TCON_ISOnative enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceId	HW_ANY	0	Hardware-Kennung der Schnittstelle des CP
2 … 3	ID	CONN_OUC	1	Referenz auf diese Verbindung (eindeutige ID im Wertebereich: 1 bis 4095).
4	ConnectionType	BYTE	16#16	Verbindungstyp: ISO
5	ActiveEstablished	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau

Byte	Parameter	Datentyp	Startwert	Beschreibung
8 … 13	RemoteMacAddress	ARRAY [1..6] of BYTE	-	MAC-Adresse des Partner-Endpunkts, z. B. für 00-74-41-FD-AE-84: • MacAddr[1] = 00 • MacAddr[2] = 74 • MacAddr[3] = 41 • MacAddr[4] = FD • MacAddr[5] = AE • MacAddr[6] = 84
14 … 19	LocalMacAddress	ARRAY [1..6] of BYTE	-	MAC-Adresse des lokalen Endpunkts, z. B. für 00-74-41-FD-AE-84: • MacAddr[1] = 00 • MacAddr[2] = 74 • MacAddr[3] = 41 • MacAddr[4] = FD • MacAddr[5] = AE • MacAddr[6] = 84
20 … 53	RemoteTSelector	Struct	-	TSelector des entfernten Verbindungspartners: • Byte 20 bis 21 = TSelLength • Byte 22 bis 53: = TSel[1-32]
	TSelLength	UINT	-	Wertebereich 0 bis 32
	TSel	ARRAY [1..32] of BYTE	-	Wertebereich jeweils 0 bis 255 in Bytes
54 … 87	LocalTSelector	Struct	-	TSelector des entfernten Verbindungspartners: • Byte 20 bis 21 = TSelLength • Byte 22 bis 53: = TSel[1-32]
	TSelLength	UINT		Wertebereich 0 bis 32
	TSel	ARRAY [1..32] of BYTE	-	Wertebereich jeweils 0 bis 255 in Bytes
88 … 89	CrRetransmitionTime	UINT	-	Zeitdauer bis zur Wiederholung des Verbindungsaufbaus in Sekunden.
90 … 91	DataRetransmition-Time	UINT	100 ms	Zeitdauer bis zur Wiederholung des Datentransfers in Millisekunden.
92 … 93	MaxRetransmition-Count	UINT	-	Maximale Anzahl der Wiederholungen.
94 … 95	InactivityTime	UINT	-	in Sekunden
96 … 97	WindowTime	UINT		in Sekunden

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Verbindung wurde erfolgreich aufgebaut.
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Auftragsbearbeitung starten, Verbindung aufbauen.
0	7002	Verbindung wird aufgebaut (REQ irrelevant).
1	8085	Die Verbindungs-ID (Parameter ID) wird bereits von einer konfigurierten Verbindung verwendet.
1	8086	Der Parameter ID liegt außerhalb des zulässigen Bereichs.
1	8087	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
1	8089	Der Parameter CONNECT zeigt nicht auf eine Verbindungsbeschreibung oder die Verbindungsbeschreibung wurde manuell erstellt.
1	809A	Die Struktur am Parameter CONNECT wird an einer integrierten Schnittstelle nicht unterstützt oder die Länge ist ungültig.
1	809B	• Das Element InterfaceId innerhalb der TCON_xxx-Struktur referenziert keine Hardware-Kennung einer CPU- oder CM/CP Schnittstelle. • Das Element InterfaceId darf in der verwendeten TCON_xxx-Struktur nicht den Wert "0" haben, siehe Abschnitt "Beschreibung".
1	80A1	Die angegebene Verbindung oder der Port sind bereits belegt.
1	80A2	Lokaler oder remoter Port wird vom System verwendet. Lokal sind folgende Ports reserviert: 20, 21, 80, 102, 135, 161, 162, 443, 34962, 34963, 34964 sowie der Bereich 49152 bis 65535.
1	80A3	ID wird von einer durch das Anwenderprogramm erstellten Verbindung genutzt, die auch die gleiche Verbindungsbeschreibung am Parameter CONNECT verwendet.
1	80A4	IP-Adresse des remoten Endpunkts der Verbindung ist ungültig oder sie entspricht der IP-Adresse des lokalen Partners.
1	80A7	Kommunikationsfehler: Sie haben "TDISCON (Seite 3700)" ausgeführt, bevor "TCON" beendet war.
1	80B4	Nur bei TCON_IP_RFC: Der lokale T-Selektor wurde nicht angegeben oder das erste Byte enthält nicht den Wert 0x0E (nur bei einer Länge von T-Selector = 2) oder der lokale T-Selektor beginnt mit "SIMATIC-".
1	80B5	Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig (Parameter ActiveEstablished der Struktur TCON_IP_v4 / TCON_PARAM hat den Wert TRUE).
1	80B6	Parametrierungsfehler im Parameter ConnectionType des Datenbausteins für Verbindungsbeschreibung. • Nur gültig bei TCON_IP_v4: 0x11, 0x0B und 0x13. • Nur gültig bei TCON_IP_RFC: 0x0C und 0x12

ERROR	STATUS* (W#16#...)	Erläuterung
1	80B7	Bei TCON_IP_v4: • TCP (aktiver Verbindungsaufbau): Entfernter Port ist "0". • TCP (passiver Verbindungsaufbau): Lokaler Port ist "0". • UDP: Lokaler Port ist "0". • IP-Adresse des Partner-Endpunkts wurde auf 0.0.0.0 gesetzt. Bei TCON_IP_RFC: • Lokaler (LocalTSelector) oder entfernter (RemoteTSelector) T-Selektor wurde mit einer Länge von mehr als 32 Byte angegeben. • Bei TSelLength des T-Selektors (lokal oder entfernt) wurde eine Länge größer als 32 eingegeben. • Fehler bei der Länge der IP-Adresse des jeweiligen Verbindungspartners. • IP-Adresse des Partner-Endpunkts wurde auf 0.0.0.0 gesetzt.
1	80B8	Parameter ID in der lokalen Verbindungsbeschreibung (Struktur am Parameter CONNECT) und Parameter ID der Anweisung sind unterschiedlich.
1	80C3	Alle Verbindungsressourcen sind belegt, oder Ports werden möglicherweise von anderen Anwendungen oder Verbindungen dynamisch verwendet.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung kann derzeit nicht aufgebaut werden. • Die Verbindung kann nicht aufgebaut werden, weil auf dem Verbindungsweg liegende Firewalls für die benötigten Ports nicht freigeschaltet sind. • Die Schnittstelle empfängt gerade neue Parameter. • Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 3700)" entfernt.
1	80C5	Der Verbindungspartner verweigert den Verbindungsaufbau, hat die Verbindung abgebaut oder aktiv beendet.
1	80C6	Der Verbindungspartner kann nicht erreicht werden (Netzwerkfehler).
1	80C7	Zeitüberschreitung der Ausführung.
1	80C8	Wert am Parameter ID wird bereits von einer Verbindung genutzt, die über das Anwenderprogramm erstellt wurde. Die Verbindung verwendet die gleiche ID, aber unterschiedliche Verbindungseinstellungen am Parameter CONNECT.
1	80C9	Validierung des Verbindungspartners fehlgeschlagen. Der Verbindungspartner, der eine Verbindung aufbauen will, entspricht nicht dem definierten Partner der Struktur am Parameter CONNECT.
1	80CE	Die IP-Adresse des lokalen Interfaces ist 0.0.0.0.
1	80D0	Im Zusammenhang mit TCP und dem aktiven Verbindungs-Endpunkt: Der Parameter remote_qdn ist eine leere Zeichenkette. In diesem Fall kann keine Verbindung aufgebaut werden.
1	80D1	Der Parameter remote_qdn ist kein voll qualifizierter Domainname. Möglicherweise fehlt der Punkt am Ende.
1	80D2	Es ist keine DNS-Server-Adresse konfiguriert.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80D3	Der voll qualifizierte Domänen-Name konnte nicht aufgelöst werden. Mögliche Ursachen: • Der DNS Server ist nicht erreichbar z. B. weil er heruntergefahren ist oder der remote Port nicht erreichbar ist. • Während der Kommunikation mit dem DNS Server ist ein Fehler aufgetreten. • Der DNS Server schickte eine gültige DNS-Antwort zurück, aber die Antwort enthielt keine IPv4-Adresse.
1	80E0	Unpassende oder schlechte Nachricht wurde empfangen.
1	80E1	Fehler während des Handshakes. Mögliche Ursachen: • Abbruch durch Benutzer • Security nicht hoch genug • Erneute Verhandlung wird nicht unterstützt • SSL/TLS-Version wird nicht unterstützt • Validierung des Hostname fehlgeschlagen
1	80E2	Nicht unterstütztes / Ungültiges / Fehlendes Zertifikat Mögliche Ursache: Die Uhrzeit der betroffenen Baugruppe ist nicht gestellt bzw. die Baugruppe wird nicht synchronisiert. Beispiel: Das Datum der Baugruppe ist auf den 1.1.2012 voreingestellt und wurde bei der Inbetriebnahme nicht eingestellt. Der Gültigkeitszeitraum des Zertifikats beginnt am 20.08.2016 und endet am 20.08.2024. In diesem Fall liegt das Datum der Baugruppe außerhalb des Gültigkeitszeitraums des Zertifikats; für die Baugruppe ist das Zertifikat ungültig.
1	80E3	Zertifikat wurde verworfen.
1	80E4	Es wurde keine gültige Zertifizierungsstelle gefunden.
1	80E5	Zertifikat abgelaufen
1	80E6	Integritätsfehler im Transport-Layer-Security-Protokoll
1	80E7	Nicht unterstützte Erweiterung in X.509-V3-Zertifikat
1	80E9	TLS-Server ohneServer-Zertifikat wird nicht unterstützt.
1	80EA	DTLS- (UDP-) Protokoll wird nicht unterstützt.
1	80EB	Ein Client kann kein Client-Zertifikat anfordern.
1	80EC	Der Server kann nicht gegen den subjectAlternateName validieren (Nur Clients können das.).
1	80ED	TLSServerCertRef_m-ID ungültig
1	8xyy	Allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
* Die Fehlercodes im Programmiereditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TCON, TDISCON, TSEND & TRCV (Seite 3720).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

TDISCON: Kommunikationsverbindung abbauen

Beschreibung

Die Anweisung "TDISCON" baut eine Kommunikationsverbindung der CPU zu einem Verbindungspartner ab.

Arbeitsweise

"TDISCON" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag zum Verbindungsabbau, indem Sie "TDISCON" mit REQ = 1 aufrufen.

Nach dem erfolgreichen Durchlauf von "TDISCON" ist die bei "TCON" angegebene ID nicht mehr gültig und kann damit weder zum Senden noch zum Empfangen verwendet werden.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen (siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TDISCON" aktuell befindet bzw. wann der Verbindungsaufbau beendet ist.

BUSY	DONE	ERROR	Beschreibung
1	0	0	Der Auftrag ist in Bearbeitung.
0	1	0	Der Auftrag wurde erfolgreich abgeschlossen.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
0	0	0	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TDISCON":

Parameter	Deklaration	Datentyp	Speicherbereich		Beschreibung
			S7-1200	S7-1500	
REQ	Input	BOOL	E, A, M, D, L oder Konstante	E, A, M, D, L oder Konstante	Steuerparameter REQUEST, startet den Auftrag zum Abbau der durch ID gegebenen Verbindung. Der Auftragsstart erfolgt bei steigender Flanke.
ID	Input	CONN_OUC (WORD)	D, L oder Konstante	E, A, M, D, L oder Konstante	Referenz auf die abzubauende Verbindung (Verbindungs-ID) Wertebereich: W#16#0001 bis W#16#0FFF
DONE	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Statusparameter: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Statusparameter: • BUSY = 1: Der Auftrag ist noch nicht beendet. • BUSY = 0: Der Auftrag ist beendet oder noch nicht gestartet.
ERROR	Output	BOOL	E, A, M, D, L	E, A, M, D, L	Statusparameter: • ERROR=0: Kein Fehler. • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	E, A, M, D, L	E, A, M, D, L	Statusparameter: Fehlerinformation (siehe "Parameter ERROR und STATUS")

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Verbindung konnte abgebaut werden.
0	7000	Keine Auftragsbearbeitung aktiv.
0	7001	Start der Auftragsbearbeitung, Verbindung wird abgebaut.
0	7002	Zwischenaufruf (REQ irrelevant), Verbindung wird abgebaut.
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich.
1	80A3	Es wird versucht, eine nicht bestehende Verbindung abzubauen oder die Verbindung ist bereits abgebaut.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80C4	Temporärer Kommunikationsfehler: Schnittstelle wird neu parametriert bzw. Verbindung wird gerade eingerichtet.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TCON, TDISCON, TSEND & TRCV (Seite 3720).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

TSEND: Daten über Kommunikationsverbindung senden

TSEND: Daten über Kommunikationsverbindung senden

Beschreibung

Die folgende Beschreibung der Anweisung "TSEND" ist gültig für die CPU S7-1200 bis Version 3.0.

Mit der Anweisung "TSEND" senden Sie Daten über eine bestehende Kommunikationsverbindung. "TSEND" wird asynchron ausgeführt.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Für die zu sendenden Daten können alle Datentypen bis auf BOOL und Array of BOOL verwendet werden.

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird.

Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden.

- Bei der Datenübertragung über TCP (Streaming-Protokoll) werden von der Anweisung "TSEND" keine Information über die Länge der gesendeten Daten an "TRCV (Seite 3709)" übermittelt.
- Bei der Datenübertragung über ISO-on-TCP (Nachrichtenorientiertes Protokoll) wird die Länge der gesendeten Daten an "TRCV (Seite 3709)" übermittelt. Die Menge von Daten, die über "TSEND" als Paket gesendet werden, müssen auf der Empfängerseite ("TRCV (Seite 3709)") auch wieder empfangen werden:
  - Ist der Empfangspuffer zu klein für die gesendeten Daten, gibt es auf Empfängerseite einen Fehler.
  - Ist der Empfangspuffer ausreichend groß, kehrt "TRCV" mit DONE=1 dann zurück, sobald das Datenpaket empfangen wurde.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter

DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits ausgelesen wurden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Startet den Sendeauftrag bei einer steigenden Flanke.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die mit "TCON" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	UINT	E, A, M, D, L oder Konstante	Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden.
DATA	InOut	VARIANT	E, A, M, D	Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält. Die Adresse verweist auf: • Das Prozessabbild der Eingänge • Das Prozessabbild der Ausgänge • Einen Merker • Einen Datenbaustein Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LEN und DATA

- Bei LEN = 0 werden alle über den Parameter DATA angegebenen Daten versendet.
- Ist die Anzahl der Bytes am Parameter LEN größer als die Länge der zu sendenden Daten, die über den Parameter DATA definiert wurden, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).

- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Bei den Datentypen STRING und WSTRING werden alle Daten übertragen, wenn der Parameter LEN = 0 ist. Bei LEN > 0 muss die Länge mindestens die maximale Anzahl der Bytes umfassen sowie zwei Byte zusätzlich, welche die Längeninformation enthalten. Weitere Informationen zum Aufbau der Datentypen finden Sie unter: "Übersicht über die gültigen Datentypen (Seite 243)".
- Die maximale Anzahl von Bytes, die übertragen werden können, beträgt 8192.
- Bei der Verwendung von strukturierten Variablen aus optimierten DBs sollte die Adresse der strukturierten Variable am Parameter DATA verschaltet und der Parameter LEN = 0 gesetzt werden. Damit kann eine typsichere Übertragung der gesamten Struktur gewährleistet werden, sofern auf Empfängerseite dieselbe Struktur verwendet wird.

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	0	0	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Hinweis

Wegen der asynchronen Bearbeitung von "TSEND" müssen Sie die Daten im Sendebereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Beschreibung
0	0000	Sendeauftrag fehlerfrei beendet.
0	7000	Keine Auftragsbearbeitung aktiv.
0	7001	Beginn der Auftragsbearbeitung, Daten werden gesendet. Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
0	7002	Auftrag wird bearbeitet (REQ irrelevant). Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.

ERROR	STATUS* (W#16#...)	Beschreibung
1	8085	• Parameter LEN ist größer als der größte zulässige Wert (65536). • Parameter DATA und LEN haben beide den Wert "0".
1	8086	Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1..0xFFF).
1	8088	Parameter LEN ist größer als der in DATA angegebene Bereich.
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu initialisiert.
1	80B3	Die parametrierte Protokollvariante (Parameter ConnectionType in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TUSEND".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet. • Interner Mangel an Ressourcen.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Partner kann derzeit nicht aufgebaut werden. • Die Schnittstelle empfängt gerade neue Parametereinstellungen oder die Verbindung wird gerade aufgebaut.
1	80C5	Verbindung durch den Kommunikationspartner abgebaut.
1	80C6	Netzwerkfehler. Kommunikationspartner kann nicht erreicht werden.
1	80C7	Zeitüberschreitung bei Ausführung.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TSEND: Daten über Kommunikationsverbindung senden

Beschreibung

Die folgende Beschreibung der Anweisung "TSEND" ist gültig für:

- Ethernet
  CPU S7-1200 mit Firmware-Stand ≥ V4.0 und CPU S7-1500
- PROFIBUS
  FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL

Mit der Anweisung "TSEND" senden Sie Daten über eine bestehende Kommunikationsverbindung. "TSEND" wird asynchron ausgeführt.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Für die zu sendenden Daten können alle Datentypen bis auf BOOL und Array of BOOL verwendet werden.

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird.

Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden.

- Bei der Datenübertragung über TCP (Streaming-Protokoll) werden von der Anweisung "TSEND" keine Information über die Länge der gesendeten Daten an "TRCV (Seite 3709)" übermittelt.
- Bei der Datenübertragung über ISO-on-TCP (Nachrichtenorientiertes Protokoll) wird die Länge der gesendeten Daten an "TRCV (Seite 3709)" übermittelt. Die Menge von Daten, die über "TSEND" als Paket gesendet werden, müssen auf der Empfängerseite ("TRCV (Seite 3709)") auch wieder empfangen werden:
  - Ist der Empfangspuffer zu klein für die gesendeten Daten, gibt es auf Empfängerseite einen Fehler.
  - Ist der Empfangspuffer ausreichend groß, kehrt "TRCV" mit DONE=1 dann zurück, sobald das Datenpaket empfangen wurde.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits ausgelesen wurden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Startet den Sendeauftrag bei einer steigenden Flanke.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die mit "TCON" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden (größter zulässiger Wert bei S7-1200: 8192, größter zulässiger Wert bei S7-1500: 65536). Bei FDL-Verbindungen des CM 1542-5 beträgt die maximale Länge 240 Byte. Beachten Sie dabei die maximale verarbeitbaren Längen beim Verbindungspartner.
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält. Die Adresse verweist auf: • Das Prozessabbild der Eingänge • Das Prozessabbild der Ausgänge • Einen Merker • Einen Datenbaustein • Lokaldaten Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LEN und DATA

- Bei LEN = 0 werden alle über den Parameter DATA angegebenen Daten versendet.
- Ist die Anzahl der Bytes am Parameter LEN größer als die Länge der zu sendenden Daten, die über den Parameter DATA definiert wurden, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Bei den Datentypen STRING und WSTRING werden alle Daten übertragen, wenn der Parameter LEN = 0 ist. Bei LEN > 0 muss die Länge mindestens die maximale Anzahl der Bytes umfassen sowie zwei Byte zusätzlich, welche die Längeninformation enthalten. Weitere Informationen zum Aufbau der Datentypen finden Sie unter: "Übersicht über die gültigen Datentypen (Seite 243)".
- Die maximale Anzahl von Bytes, die übertragen werden können, ist geräteabhängig.
- Bei der Verwendung von strukturierten Variablen aus optimierten DBs sollte die Adresse der strukturierten Variable am Parameter DATA verschaltet und der Parameter LEN = 0 gesetzt werden. Damit kann eine typsichere Übertragung der gesamten Struktur gewährleistet werden, sofern auf Empfängerseite dieselbe Struktur verwendet wird.

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	0	0	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Hinweis

Wegen der asynchronen Bearbeitung von "TSEND" müssen Sie die Daten im Sendebereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Beschreibung
0	0000	Sendeauftrag fehlerfrei beendet.
0	7000	Keine Auftragsbearbeitung aktiv.
0	7001	Beginn der Auftragsbearbeitung, Daten werden gesendet. Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
0	7002	Auftrag wird bearbeitet (REQ irrelevant). Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
1	8085	• Parameter LEN ist größer als der größte zulässige Wert Bei S7-1200: 8192; bei S7-1500 (TCP): 65536; bei S7-1500 (FDL): 240 / 236 • Parameter DATA und LEN haben beide den Wert "0".
1	8086	Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1..0xFFF).
1	8088	Parameter LEN ist größer als der in DATA angegebene Bereich.
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu initialisiert.
1	80B1	Sie haben den Parameter DATA geändert, bevor der laufende Auftrag beendet wurde.
1	80B3	Die parametrierte Protokollvariante (Parameter ConnectionType in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TUSEND".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet. • Interner Mangel an Ressourcen.

ERROR	STATUS* (W#16#...)	Beschreibung
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Partner kann derzeit nicht aufgebaut werden. • Die Schnittstelle empfängt gerade neue Parametereinstellungen oder die Verbindung wird gerade aufgebaut. • Temporär ist keine Empfangsressourcen beim Verbindungspartner verfügbar. Der Verbindungspartner ist nicht empfangsbereit. • Das R/H-System befindet sich im Systemzustand SYNCUP oder es findet eine Primary-Backup-Umschaltung statt. Die Verbindung wird geschlossen. Die Bearbeitung von TSEND wird gestoppt.
1	80C5	• Verbindung durch den Kommunikationspartner abgebaut. • LSAP des remoten Verbindungspartners nicht freigegeben
1	80C6	Netzwerkfehler: • Remoter Partner kann nicht erreicht werden. • Physikalische Unterbrechung an PROFIBUS
1	80C7	Zeitüberschreitung bei Ausführung.
1	80EE	Handshake noch nicht beendet
1	8xyy	Allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TCON, TDISCON, TSEND & TRCV (Seite 3720).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TSEND: Daten über Kommunikationsverbindung senden (Seite 3702)

TRCV: Daten über Kommunikationsverbindung empfangen

TRCV: Daten über Kommunikationsverbindung empfangen

Beschreibung

Die folgende Beschreibung der Anweisung "TRCV" ist gültig für die CPU S7-1200 bis Version 3.0.

Mit der Anweisung "TRCV" empfangen Sie Daten über eine bestehende Kommunikationsverbindung. "TRCV" wird asynchron ausgeführt.

Wenn der Parameter EN_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest.

Während Daten empfangen werden, dürfen an dem Parameter DATA und dem definierten Empfangsbereich keine Änderungen vorgenommen werden, um die Konsistenz der empfangenen Daten sicherzustellen.

Nach einem erfolgreichen Datenempfang wird der Parameter NDR auf den Wert "1" gesetzt. Die tatsächlich empfangene Datenmenge können Sie am Parameter RCVD_LEN abfragen.

Empfangsmodi von "TRCV"

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

Protokollvariante	Verfügbarkeit der Daten im Empfangsbereich	Parameter connection_type* der Verbindungsbeschreibung	Parameter LEN
TCP (Ad-hoc-Modus)	Die Daten sind sofort verfügbar.	Hexadezimalwert: B#16#11 Integerwert: 17	0
TCP (Datenempfang mit angegebener Länge)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#11 Integerwert: 17	1 bis 8192
ISO on TCP (Nachrichtenorientiert Datenübertragung)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#12 Integerwert: 18	• 1 bis 1452, wenn ein CP verwendet wird. • 1 bis 8192, wenn kein CP verwendet wird.
* Siehe "AUTOHOTSPOT".			

TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter LEN den Wert "0" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Ist der Ad-hoc-Modus aktiviert, wird der Datenempfang schon nach einem empfangenen Byte am Parameter NDR gemeldet.

TCP (Datenempfang mit angegebener Länge)

Für einen Datenempfang mit angegebener Länge geben Sie die Länge der Daten am Parameter LEN an. Der Datenempfang wird erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Der Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV" einen Fehler. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Empfangsfreigabe
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die mit "TCON (Seite 3689)" aufgebaute Verbindung. Wertebereich: W#16#0001 (1) bis W#16#0FFF (4095)
LEN	Input	UINT	E, A, M, D, L oder Konstante	Länge des Empfangsbereichs in Bytes (ausgeblendet). Wenn Sie einen Speicherbereich mit optimiertem Zugriff am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.
DATA	InOut	VARIANT	E, A, M, D	Zeiger auf den Empfangsbereich Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
NDR	Output	BOOL	E, A, M, D, L	Statusparameter (New Data Received): • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter: • 0: Kein Fehler. • 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten. Detailinformationen werden über den Parameter STATUS ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Statusparameter: Ausgabe der Status- und Fehlerinformation.
RCVD_LEN	Output	UINT	E, A, M, D, L	Tatsächlich empfangene Datenmenge in Bytes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LEN, DATA und RCVD_LEN

- Ist LEN = 0, werden die empfangenen Daten in dem am Parameter DATA angegebenen Empfangsbereich gespeichert. Die Anzahl der empfangenen Bytes werden am Parameter RCVD_LEN angezeigt.
  Bei LEN = 0 (Voreinstellung des Parameters LEN) wird die Länge der zu empfangenden Daten über den Parameter DATA festgelegt. Es wird empfohlen, dass der Empfangsbereich (Parameter DATA) dieselbe Größe hat wie die von TSEND übertragenen Daten.
  Falls LEN den Wert 0 hat und die gesendeten Daten in Segmenten übertragen werden, die kleiner sind als der Empfangsbereich DATA, dann gilt das Folgende. Es wird empfohlen, EN_R so lange gesetzt zu halten, bis die zugehörige TSEND-Anweisung alle Daten gesendet hat. Solange die Größe der von TSEND gesendeten Daten ungleich der Größe des Empfangsbereichs DATA ist, zeigt STATUS den Wert 7002. EN_R muss gesetzt sein, bis die Anzahl der empfangenen Daten gleich der Größe des Empfangsbereichs DATA ist. Wenn Sie EN_R pulsen, müssen Sie dies so lange tun, bis BUSY=0 bzw. ERROR <> 0. Die Daten im Empfangsbereich DATA sind erst dann gültig, wenn BUSY den Wert 0 annimmt.
- Ist die am Parameter LEN angegebene Länge größer als die Länge der am Parameter DATA empfangenen Daten, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Verweist der Parameter DATA auf einen Datenbaustein mit optimiertem Zugriff, muss der Parameter LEN auf "0" gesetzt werden.
- Wird über den Parameter DATA ein Datentyp STRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder >=2 sein (LEN = 1 ist nicht erlaubt).
- Wird über den Parameter DATA ein Datentyp WSTRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder >=5 sein.

Parameter BUSY, NDR und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, NDR, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter NDR kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von TRCV auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, NDR und ERROR:

BUSY	NDR	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Hinweis

Aufgrund der asynchronen Bearbeitung von "TRCV" sind die Daten im Empfangsbereich nur dann konsistent, wenn der Parameter NDR auf den Wert "1" gesetzt wird.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Auftrag erfolgreich ausgeführt. Die aktuelle Länge der empfangenen Daten wird am Parameter RCVD_LEN ausgegeben.
0	7000	Baustein nicht für den Empfang bereit.
0	7001	Baustein ist für den Empfang bereit, Empfangsauftrag wurde aktiviert.
0	7002	Zwischenaufruf, Empfangsauftrag wird bearbeitet. Hinweis: Während dieser Bearbeitung werden Daten in den Empfangsbereich geschrieben. Währenddessen kann der Zugriff auf den Empfangsbereich inkonsistente Daten liefern.
1	8085	• Parameter LEN ist größer als der größte zulässige Wert. • Der Wert des Parameters LEN oder DATA wurde nach dem ersten Aufruf geändert. • Beide LEN-Parameter und der Parameter DATA haben den Wert "0" oder LEN ist länger als der maximal zulässige Wert (65536).
1	8086	Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1 .. 0x0FFF).
1	8088	• Empfangsbereich ist zu klein. • Der Wert am Parameter LEN ist größer als der am Parameter DATA angegebene Empfangsbereich.
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird gerade beendet. Empfangsauftrag über diese Verbindung ist nicht möglich. • Die Verbindung wird gerade neu initialisiert.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80B3	Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TURCV".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet. • Interner Mangel an Ressourcen.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Partner kann derzeit nicht aufgebaut werden. • Die Schnittstelle empfängt neue Parametereinstellungen oder die Verbindung wird aufgebaut.
1	80C5	Der remote Partner hat die Verbindung abgebaut.
1	80C6	Der remote Partner kann nicht erreicht werden (Netzwerkfehler).
1	80C7	Zeitüberschreitung der Ausführung.
1	80C9	Die Länge des Empfangsbereichs ist kleiner als dieLänge der gesendeten Daten.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

TRCV: Daten über Kommunikationsverbindung empfangen

Beschreibung

Die folgende Beschreibung der Anweisung "TRCV" ist gültig für:

- Ethernet
  CPU S7-1200 mit Firmware-Stand ≥ V4.0 und CPU S7-1500
- PROFIBUS
  FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL

Mit der Anweisung "TRCV" empfangen Sie Daten über eine bestehende Kommunikationsverbindung. "TRCV" wird asynchron ausgeführt.

Wenn der Parameter EN_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest.

Während Daten empfangen werden, dürfen an dem Parameter DATA und dem definierten Empfangsbereich keine Änderungen vorgenommen werden, um die Konsistenz der empfangenen Daten sicherzustellen.

Nach einem erfolgreichen Datenempfang wird der Parameter NDR auf den Wert "1" gesetzt. Die tatsächlich empfangene Datenmenge können Sie am Parameter RCVD_LEN abfragen.

Empfangsmodi von "TRCV"

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

Protokollvariante	Parameter ADHOC	Verfügbarkeit der Daten im Empfangsbereich	Parameter connection_type der Verbindungsbeschreibung	Parameter LEN
TCP (Ad-hoc-Modus)	1 (Ad-hoc aktiviert)	Die Daten sind sofort verfügbar.	Hexadezimalwert: B#16#11 Integerwert: 17	Es werden so viele Daten gelesen, wie gerade verfügbar sind; maximal bis zur Datenlänge, die durch LEN festgelegt wird. LEN = 0 (empfohlen): Länge des Empfangsbereichs, der durch DATA refernziert wird
TCP (Datenempfang mit angegebener Länge)	0 (Ad-hoc deaktiviert)	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#11 Integerwert: 17	• S7-1200: 1 bis 8192 • S7-1500: 1 bis 65536
ISO on TCP (Nachrichtenorientierte Datenübertragung)	-	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#12 Integerwert: 18	• S7-1200: 1 bis 8192 • S7-1500: 1 bis 65536
FDL	-	Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde.	Hexadezimalwert: B#16#15 Dezimal: 21	1 bis 240

TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter ADHOC den Wert "1" zuweisen. Ist der Ad-hoc-Modus aktiviert, wird der Datenempfang schon nach einem empfangenen Byte am Parameter NDR gemeldet. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.).

TCP (Datenempfang mit angegebener Länge)

Für einen Datenempfang mit angegebener Länge weisen Sie dem Parameter ADHOC den Wert "0" zu. Ist der Ad-hoc-Modus deaktiviert, wird der Datenempfang erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV" einen Fehler. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Empfangsfreigabe
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die mit "TCON" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Länge des Empfangsbereichs in Byte (ausgeblendet) Größter Wert bei S7-1200: 8192; größter Wert bei S7-1500: 65536; größter Wert beim CM 1542-5 (FDL): 240. Wenn Sie am Parameter DATA einen Empfangsbereich mit optimiertem Zugriff verwenden, muss der Parameter LEN den Wert "0" haben.
ADHOC	Input	BOOL	E, A, M, D, L oder Konstante	Ad-hoc Modus für die Protokollvariante TCP verwenden (ausgeblendet). ADHOC muss den Wert FALSE haben, wenn kein TCP-Protokoll benutzt wird.
DATA	InOut	VARIANT	E, A, M, D, L	Zeiger auf den Empfangsbereich Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
NDR	Output	BOOL	E, A, M, D, L	Statusparameter (New Data Received): • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Neue Daten empfangen
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter ERROR: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung
RCVD_LEN	Output	UDINT	E, A, M, D, L	Tatsächlich empfangene Datenmenge in Bytes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LEN, DATA und RCVD_LEN

- Ist LEN = 0, werden die empfangenen Daten in dem am Parameter DATA angegebenen Empfangsbereich gespeichert. Die Anzahl der empfangenen Bytes werden am Parameter RCVD_LEN angezeigt.
  Bei LEN = 0 (Voreinstellung des Parameters LEN) wird die Länge der zu empfangenden Daten über den Parameter DATA festgelegt. Es wird empfohlen, dass der Empfangsbereich (Parameter DATA) dieselbe Größe hat wie die von TSEND übertragenen Daten.
  Falls LEN den Wert 0 hat und die gesendeten Daten in Segmenten übertragen werden, die kleiner sind als der Empfangsbereich DATA, dann gilt das Folgende. Es wird empfohlen, EN_R so lange gesetzt zu halten, bis die zugehörige TSEND-Anweisung alle Daten gesendet hat. Solange die Größe der von TSEND gesendeten Daten ungleich der Größe des Empfangsbereichs DATA ist, zeigt STATUS den Wert 7002. EN_R muss gesetzt sein, bis die Anzahl der empfangenen Daten gleich der Größe des Empfangsbereichs DATA ist. Wenn Sie EN_R pulsen, müssen Sie dies so lange tun, bis BUSY=0 bzw. ERROR <> 0. Die Daten im Empfangsbereich DATA sind erst dann gültig, wenn BUSY den Wert 0 annimmt.
- Ist die am Parameter LEN angegebene Länge größer als die Länge der am Parameter DATA empfangenen Daten, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Verweist der Parameter DATA auf einen Datenbaustein mit optimiertem Zugriff, muss der Parameter LEN auf "0" gesetzt werden. Stimmt die Länge der Daten bei elementaren Datentypen nicht überein, werden die Daten nicht empfangen und der Fehlercode 8088 am Parameter STATUS ausgegeben.
- Wird über den Parameter DATA ein Datentyp STRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder >=2 sein (LEN = 1 ist nicht erlaubt).
- Wird über den Parameter DATA ein Datentyp WSTRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder >=5 sein.

Parameter BUSY, NDR und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, NDR, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter NDR kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von TRCV auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, NDR und ERROR:

BUSY	NDR	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Hinweis

Aufgrund der asynchronen Bearbeitung von "TRCV" sind die Daten im Empfangsbereich nur dann konsistent, wenn der Parameter NDR auf den Wert "1" gesetzt wird.

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Auftrag erfolgreich ausgeführt. Die aktuelle Länge der empfangenen Daten wird am Parameter RCVD_LEN ausgegeben.
0	7000	Baustein nicht für den Empfang bereit.
0	7001	Baustein ist für den Empfang bereit, Empfangsauftrag wurde aktiviert.
0	7002	Zwischenaufruf, Empfangsauftrag wird bearbeitet. Hinweis: Während dieser Bearbeitung werden Daten in den Empfangsbereich geschrieben. Währenddessen kann der Zugriff auf den Empfangsbereich inkonsistente Daten liefern.
1	8085	• Parameter LEN ist größer als der größte zulässige Wert (bei S7-1200: 8192 Bytes, bei S7-1500: 65536 Bytes). • Der Wert des Parameters LEN oder DATA wurde nach dem ersten Aufruf geändert. • Beide LEN-Parameter und der Parameter DATA haben den Wert "0" oder LEN ist länger als der maximal zulässige Wert (bei S7-1200: 8192 Bytes, bei S7-1500: 65536 Bytes).
1	8086	Der Parameter ID liegt außerhalb des zulässigen Wertebereichs (1 .. 0x0FFF).
1	8088	• Empfangsbereich ist zu klein. • Der Wert am Parameter LEN ist größer als der am Parameter DATA angegebene Empfangsbereich.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird gerade beendet. Empfangsauftrag über diese Verbindung ist nicht möglich. • Die Verbindung wird gerade neu initialisiert.
1	80B1	Sie haben den Parameter DATA geändert, bevor der laufende Auftrag beendet wurde.
1	80B3	Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TURCV".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet. • Interner Mangel an Ressourcen.
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Partner kann derzeit nicht aufgebaut werden. • Die Schnittstelle empfängt neue Parametereinstellungen oder die Verbindung wird aufgebaut. • Das R/H-System befindet sich im Systemzustand SYNCUP oder es findet eine Primary-Backup-Umschaltung statt. Die Verbindung wird geschlossen. Die Bearbeitung von TRCV wird gestoppt.
1	80C5	Der remote Partner hat die Verbindung abgebaut.
1	80C6	Der remote Partner kann nicht erreicht werden (Netzwerkfehler).
1	80C7	Zeitüberschreitung der Ausführung.
1	80C9	Die Länge des Empfangsbereichs ist kleiner als die Länge der gesendeten Daten.
1	80EE	Handshake noch nicht beendet
1	8xyy	Allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TCON, TDISCON, TSEND & TRCV (Seite 3720).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

TRCV: Daten über Kommunikationsverbindung empfangen (Seite 3709)

TCON: Kommunikationsverbindung aufbauen (Seite 3692)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für TCON, TDISCON, TSEND & TRCV

Einleitung

In dem folgenden Beispiel stellen Sie eine programmierte Verbindung zwischen zwei CPUs der Baurreihe S7-1500 her und senden einen Datensatz von der CPU 1 zur CPU 2.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Deren Verbindung ist nicht konfiguriert.

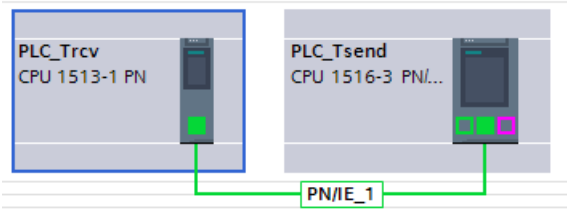

- Je CPU ist unter Eigenschaften > Schutz mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Programm der CPU 1: Ablage der Daten

Für den Datensatz legen Sie den folgenden PLC-Datentyp an.

TSEND_User

	Name	Data type	Default value
1	prename	String	''
2	lastname	String	''
3	userId	UInt	0

Für die Datenübermittlung legen Sie auf Basis des erstellten PLC-Datentyps den folgenden Datenbaustein ("SLI_plcDB_sendData_TSEND") an.

SLI_plcDB_sendData_TSEND

	Name	Data type	Start value
1	Static		
2	prename	String	'Simon'
3	lastname	String	'Sample'
4	userId	UInt	1

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_TSEND") mit den folgenden Strukturen und Variablen an.

SLI_gDB_TSEND			
	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	1
3	TCON	Struct	
4	start	Bool	false
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStatus	Word	16#0
10	TSEND	Struct	
11	start	Bool	false
12	maxLength	UInt	0
13	busy	Bool	false
14	done	Bool	false
15	error	Bool	false
16	status	Word	16#0
17	memErrStatus	Word	16#0
18	TDISCON	Struct	
19	start	Bool	false
20	busy	Bool	false
21	done	Bool	false
22	error	Bool	false
23	status	Word	16#0
24	memErrStatus	Word	16#0

Programm der CPU 1: Parameter verschalten

Sie erstellen den FB "SLI_FB_TSEND". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	TCON_Instance	TCON		
9	TSEND_Instance	TSEND		
10	TDISCON_Instance	TDISCON		
11	doneCON	Bool	false	Non-retain
12	doneSEND	Bool	false	Non-retain
13	doneDISC	Bool	false	Non-retain

Netzwerk 1: Die Parameter der Anweisung "TCON" verschalten Sie wie folgt.

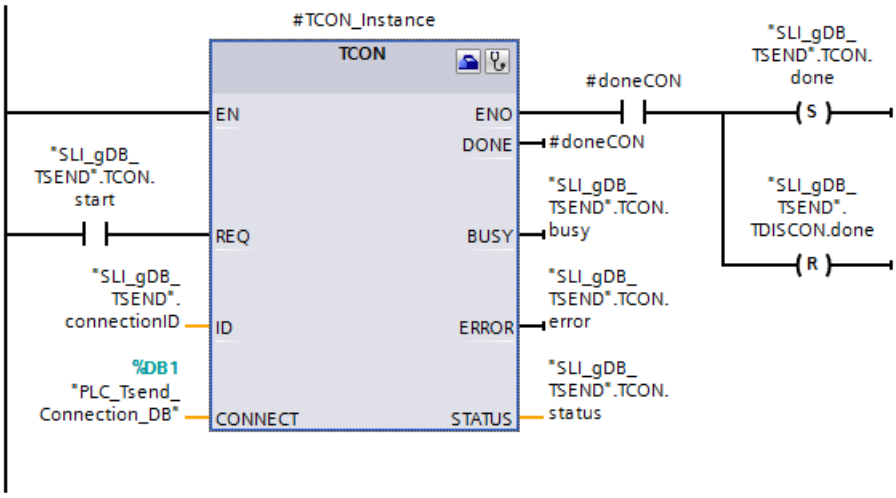

Netzwerk 2: Den Status im Fehlerfall von TCON speichern Sie wie folgt.

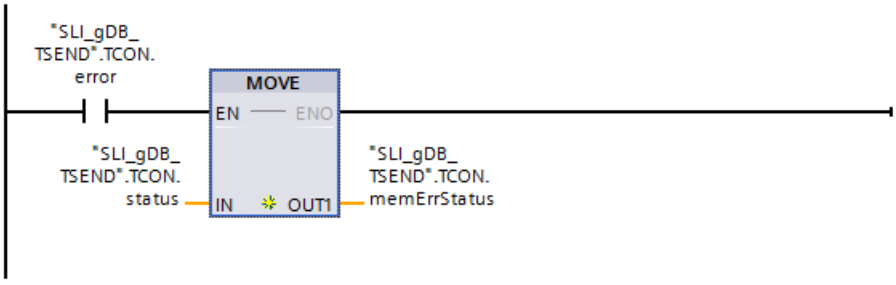

Netzwerk 3: Die Parameter der Anweisung "TSEND" verschalten Sie wie folgt.

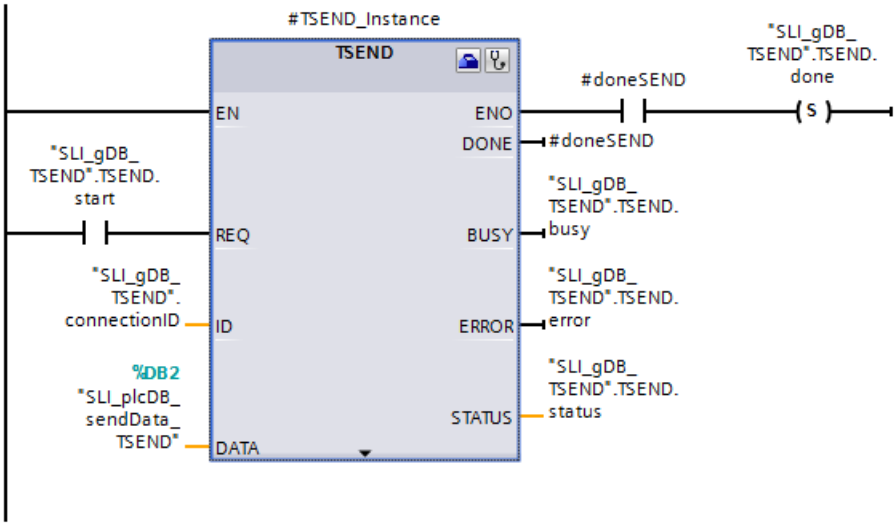

Netzwerk 4: Den Status im Fehlerfall von TSEND speichern Sie wie folgt.

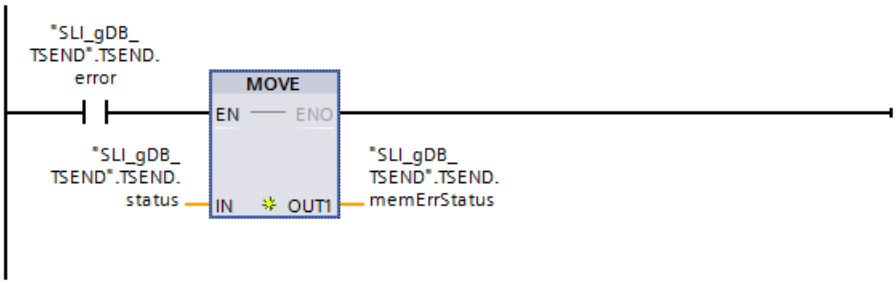

Netzwerk 5: Die Parameter der Anweisung "TDISCON" verschalten Sie wie folgt.

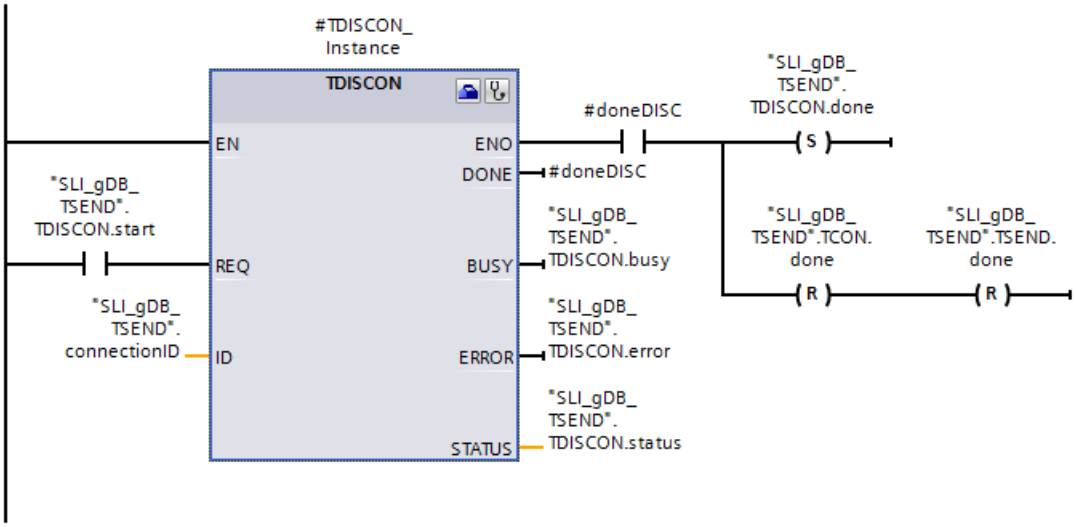

Netzwerk 6: Den Status im Fehlerfall von TDISCON speichern Sie wie folgt.

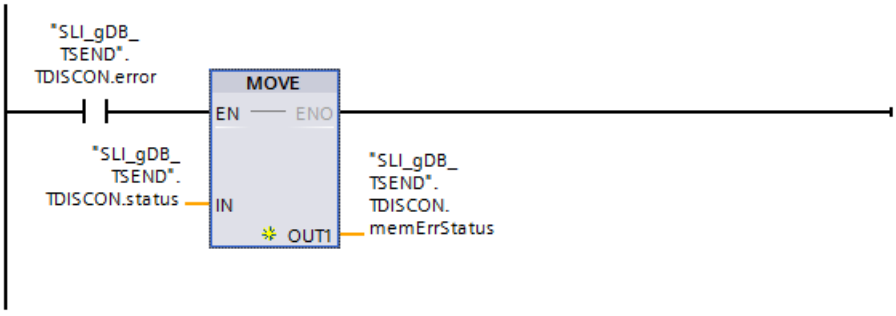

Programm der CPU 1: Konfiguration von TCON

Zur Verschaltung des Eingangsparameters CONNECT öffnen Sie den Wizard der Anweisung "TCON" über deren Eigenschaften > Konfiguration.

Sie nehmen folgende Einstellungen für die Konfiguration von TCON vor:

Eingabefeld	Eintrag
Endpunkt	Sie wählen für Sender und Empfänger über die Klappliste eine CPU aus. Schnittstelle, Subnetz und Adresse werden automatisch eingetragen.
Verbindungsdaten	Über die Klappliste und die Auswahl "Neu" erstellen Sie je CPU einen Datenbaustein. Dieser Datenbaustein wird für die Ablage der Verbindungsdaten benötigt. Der Name des Datenbausteins ist frei wählbar. Für die CPU 1 (lokale CPU) ist der aktive Verbindungsaufbau aktiviert.
Verbindungstyp	Sie wählen den Eintrag "ISO-on-TCP". Dadurch wird für den Verbindungsaufbau eine Ethernet-Verbindung mit dem Protokoll "ISO-on-TCP" verwendet.
Verbindungs-ID	Sie tragen eine selbst gewählte Verbindungs-ID für die Kommunikationsverbindung ein. Die Verbindungs-ID darf nicht bereits im Projekt vergeben sein. Achten Sie darauf, dass der Wert mit dem verwendeten Wert der Variable "connectionID" übereinstimmt.
TSAP ID	Die Adressdetails werden bei der Auswahl des Verbindungstyps "ISO-on-TCP" automatisch eingetragen.

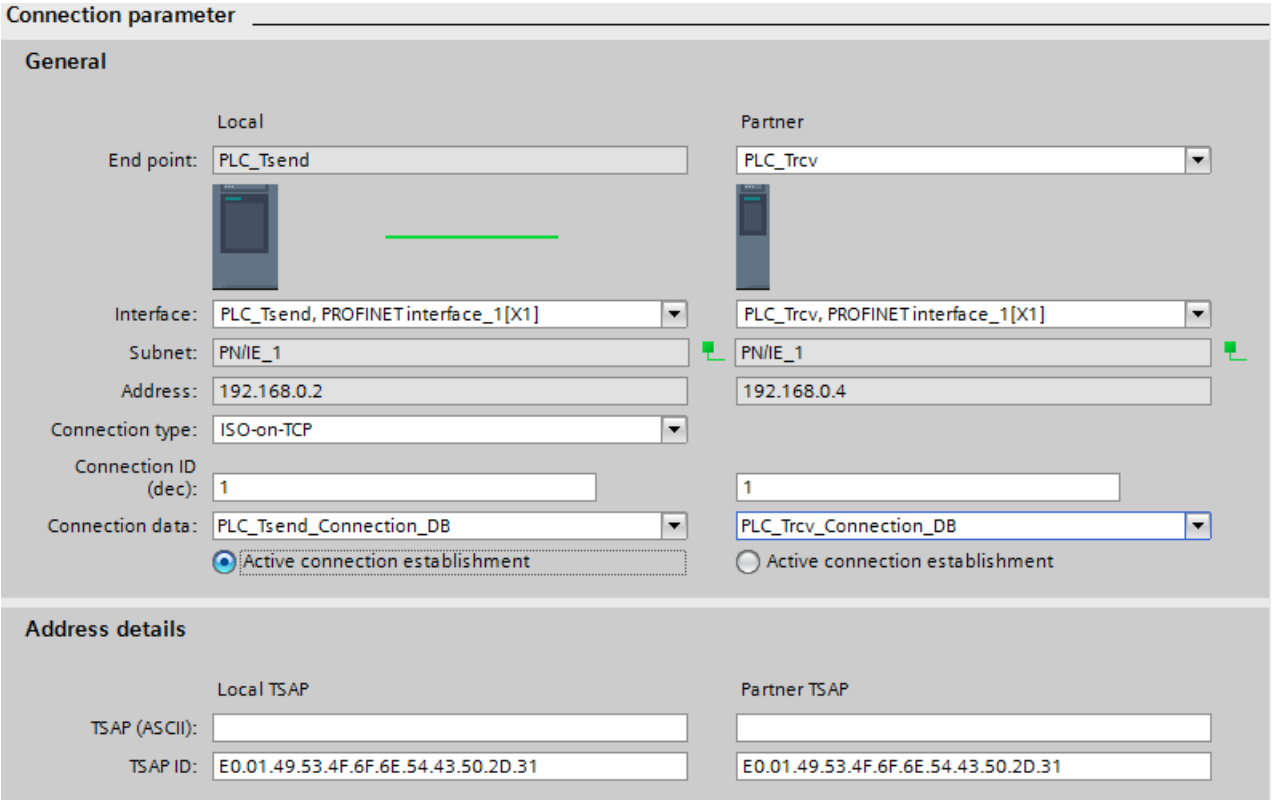

Programm der CPU 2: Ablage der Daten

Für den Datensatz legen Sie den folgenden PLC-Datentyp an.

TSEND_User

	Name	Data type	Default value
1	prename	String	''
2	lastname	String	''
3	userId	UInt	0

Für die Datenübermittlung legen Sie auf Basis des erstellten PLC-Datentyps den folgenden Datenbaustein ("SLI_plcDB_rcvData_TRCV") an.

SLI_plcDB_rcvData_TRCV

	Name	Data type	Start value
1	Static		
2	prename	String	''
3	lastname	String	''
4	userId	UInt	0

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_TRCV") mit den folgenden Strukturen und Variablen an.

SLI_gDB_TRCV

	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	1
3	TCON	Struct	
4	start	Bool	false
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStatus	Word	16#0
10	TRCV	Struct	
11	start	Bool	false
12	busy	Bool	false
13	done	Bool	false
14	error	Bool	false
15	status	Word	16#0
16	memErrStatus	Word	16#0
17	readLength	UInt	0
18	TDISCON	Struct	
19	start	Bool	false
20	busy	Bool	false
21	done	Bool	false
22	error	Bool	false
23	status	Word	16#0
24	memErrStatus	Word	16#0

Programm der CPU 2: Parameter verschalten

Sie erstellen den FB "SLI_FB_TRCV". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	doneCON	Bool	false	Non-ret...
9	doneRCV	Bool	false	Non-retain
10	doneDISC	Bool	false	Non-retain
11	length	UInt	0	Non-retain
12	TCON_Instance	TCON		
13	TRCV_Instance	TRCV		
14	TDISCON_Instance	TDISCON		

Netzwerk 1: Die Parameter der Anweisung "TCON" verschalten Sie wie folgt.

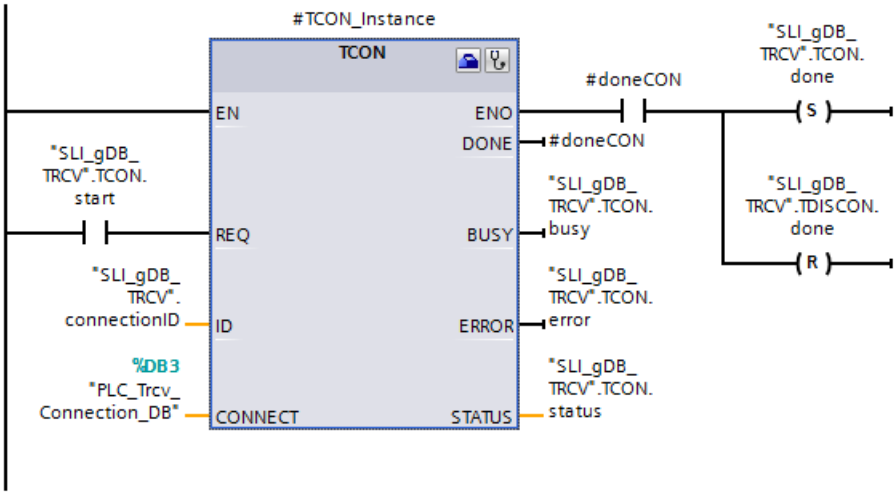

Netzwerk 2: Den Status im Fehlerfall von TCON speichern Sie wie folgt.

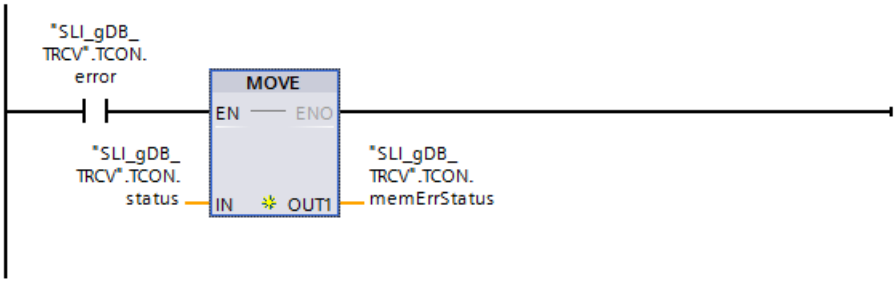

Netzwerk 3: Die Parameter der Anweisung "TRCV" verschalten Sie wie folgt.

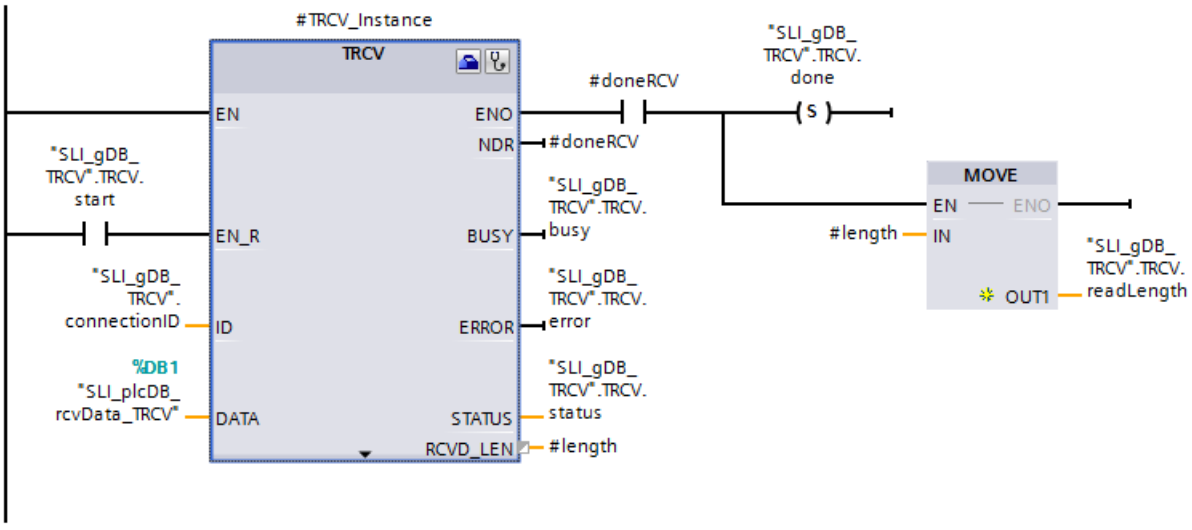

Netzwerk 4: Den Status im Fehlerfall von TRCV speichern Sie wie folgt.

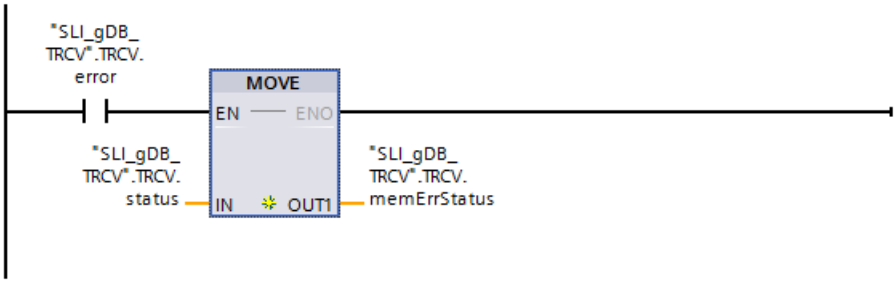

Netzwerk 5: Die Parameter der Anweisung "TDISCON" verschalten Sie wie folgt.

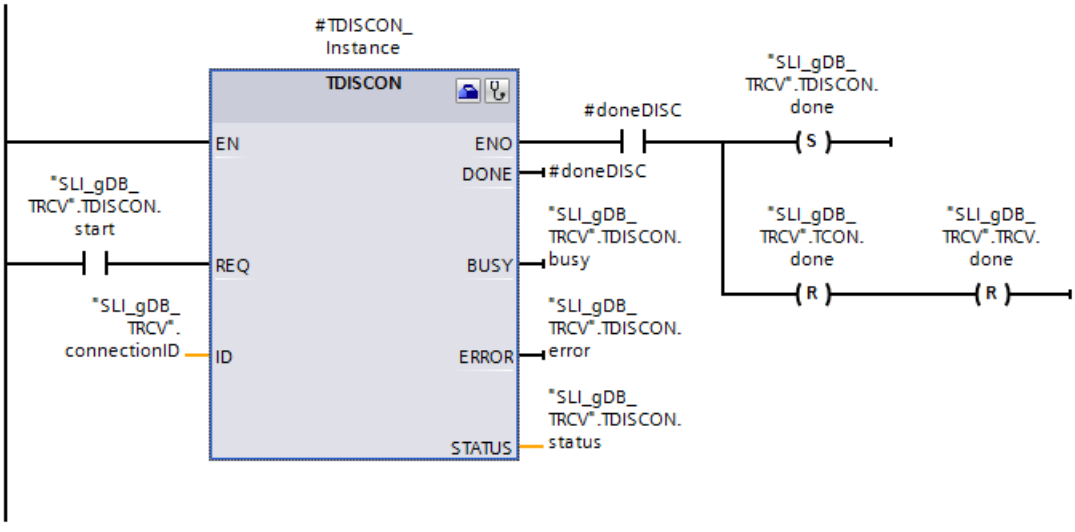

Netzwerk 6: Den Status im Fehlerfall von TDISCON speichern Sie wie folgt.

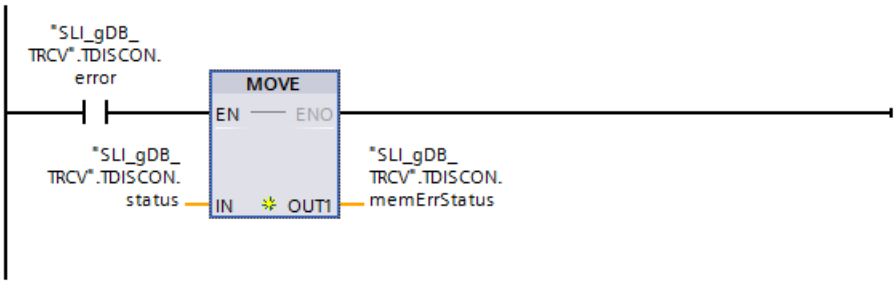

Programm der CPU 2: Konfiguration von TCON

Sie nehmen folgende Einstellungen für die Verbindung von CPU 2 / TCON nach dem Vorbild von CPU 1 / TCON vor:

Hinweis

Verbindungsdaten

Mit dem Konfigurieren von CPU 1 / TCON ist bereits je CPU ein Datenbaustein ("PLC_Tsend_Connection_DB", "PLC_Trcv_Connection_DB") mit den hinterlegten Verbindungsdaten erstellt. Anstatt neue Datenbausteine zu erstellen und die Verbindungsdaten einzutragen, können Sie einfach die erstellten Datenbausteine ("PLC_Tsend_Connection_DB", "PLC_Trcv_Connection_DB") verwenden.

Damit die erstellten Datenbausteine auswählbar sind, müssen diese im Projektbaum unter Programbausteine > Systembausteine weiterhin zu finden sein.

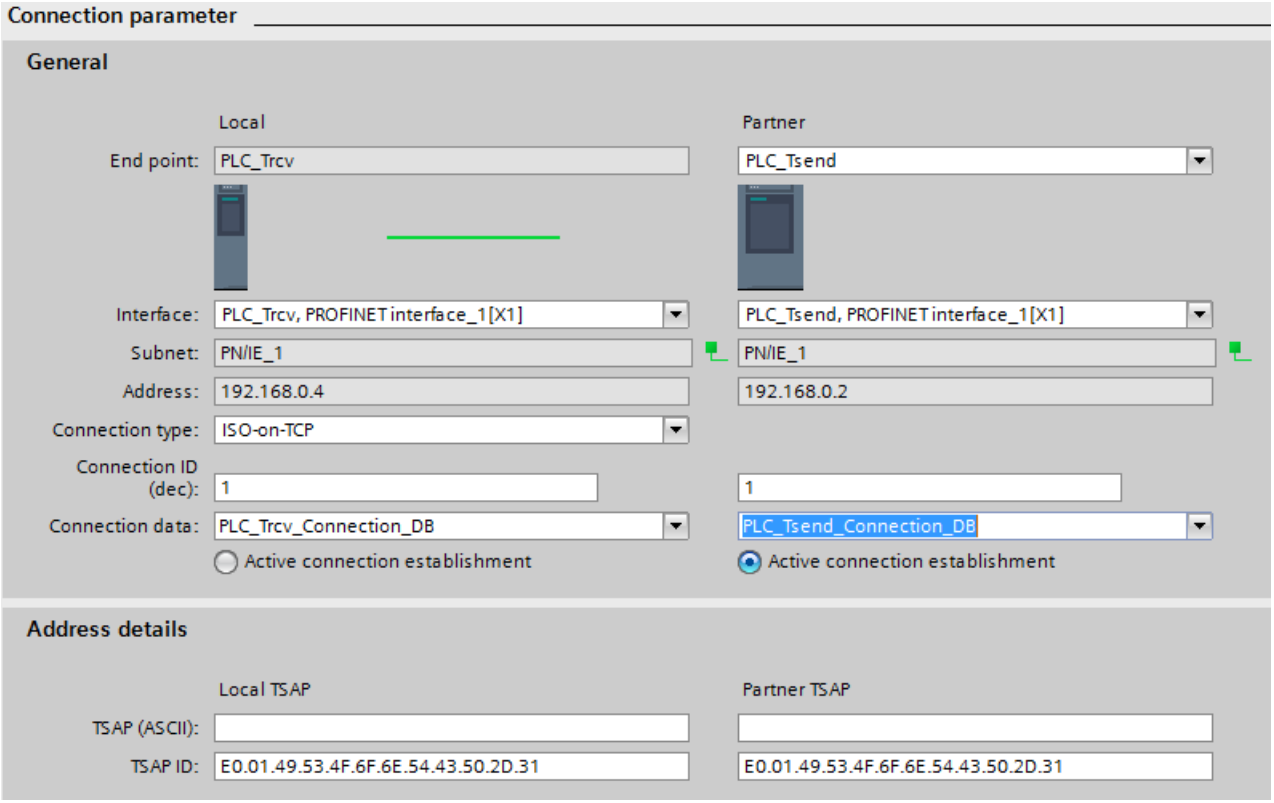

Vorgehen für den Verbindungsaufbau

Beachten Sie die Reihenfolge der Handungsschritte:

1. Ändern Sie für CPU 2 / TCON den Parameter REQ auf "TRUE".
2. Ändern Sie für CPU 1 / TCON den Parameter REQ auf "TRUE".
3. Ändern Sie für CPU 2 / TCON den Parameter REQ auf "FALSE".
4. Ändern Sie für CPU 1 / TCON den Parameter REQ auf "FALSE".

Vorgehen für den Verbindungsabbbau

Beachten Sie die Reihenfolge der Handungsschritte:

1. Ändern Sie für CPU 1 / TDISCON den Parameter REQ auf "TRUE".
2. Ändern Sie für CPU 2 / TDISCON den Parameter REQ auf "TRUE".
3. Ändern Sie für CPU 1 / TDISCON den Parameter REQ auf "FALSE".
4. Ändern Sie für CPU 2 / TDISCON den Parameter REQ auf "FALSE".

Verhalten der CPU 1

Netzwerk 1 (TCON):

Wenn der Eingangsparameter REQ ("TCON.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TCON" gestartet. Über mehrere Aufrufe hinweg erstellt die Anweisung "TCON" eine Kommunikationsverbindung zur Partner-CPU. Dazu werden die Verbindungsdaten über den Eingangsparameter CONNECT (Datenbaustein "PLC_Trcv_Connection_DB") abgerufen.

Über den Ausgangsparameter DONE ("#doneCON") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TCON.status") mit dem Wert "0000" wird der erfolgreiche Verbindungsaufbau angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "TCON.done". Den Erfolgsstatus eines vorherigen Verbindungsabbaus ("TDISCON.done") setzen Sie zurück.

Am Ausgangsparameter ERROR ("TCON.error") bzw. der Variable "TCON.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

3	TCON	Struct		
4	start	Bool	false	TRUE
5	busy	Bool	false	FALSE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7000
9	memErrStatus	Word	16#0	16#0000

Netzwerk 2 (TCON):

Wenn TCON einen Fehler meldet ("TCON.error" ist "TRUE"), wird der gemeldete Status ("TCON.status") dauerhaft gespeichert ("TCON.memErrStatus").

Netzwerk 3 (TSEND):

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "TSEND" die zu verwendende Kommunikationsverbindung bekannt.

Wenn der Eingangsparameter REQ ("TSEND.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TSEND" gestartet. Über mehrere Aufrufe hinweg übermittelt die Anweisung "TSEND" den am Eingangsparameter DATA ("SLI_plcDB_sendData_TSEND") erfassten Datensatz.

Über den Ausgangsparameter DONE ("#doneSEND") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TSEND.status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "TSEND.done".

Am Ausgangsparameter ERROR ("TSEND.error") bzw. der Variable "TSEND.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

10	TSEND	Struct		
11	start	Bool	false	TRUE
12	maxLength	UInt	0	0
13	busy	Bool	false	FALSE
14	done	Bool	false	TRUE
15	error	Bool	false	FALSE
16	status	Word	16#0	16#7000
17	memErrStatus	Word	16#0	16#0000

Netzwerk 4 (TSEND):

Wenn TSEND einen Fehler meldet ("TSEND.error" ist "TRUE"), wird der gemeldete Status ("TSEND.status") dauerhaft gespeichert ("TSEND.memErrStatus").

Netzwerk 5 (TDISCON):

Wenn der Eingangsparameter REQ ("TDISCON.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TDISCON" gestartet. Über mehrere Aufrufe hinweg baut die Anweisung "TDISCON" die Kommunikationsverbindung ab. Dazu wird die verwendende Kommunikationsverbindung am Eingangsparameters ID ("connectionID") abgerufen.

Über den Ausgangsparameter DONE ("#doneDISC") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TDISCON.status") mit dem Wert "0000" wird der erfolgreiche Verbindungsabbau angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "TDISCON.done". Die Erfolgsmeldungen zum Verbindungsaufbau ("TCON.done") und zur Datenübermittlung ("TSEND.done") setzen Sie zurück.

Am Ausgangsparameter ERROR ("TDISCON.error") bzw. der Variable "TDISCON.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

18	TDISCON	Struct		
19	start	Bool	false	TRUE
20	busy	Bool	false	FALSE
21	done	Bool	false	TRUE
22	error	Bool	false	FALSE
23	status	Word	16#0	16#7000
24	memErrStatus	Word	16#0	16#0000

Netzwerk 6 (TDISCON):

Wenn TDISCON einen Fehler meldet ("TDISCON.error" ist "TRUE"), wird der gemeldete Status ("TDISCON.status") dauerhaft gespeichert ("TDISCON.memErrStatus").

Verhalten der CPU 2

Für TCON und TDISCON ist das Verhalten gleich dem Verhalten bei der CPU 1.

3	TCON	Struct		
4	start	Bool	false	TRUE
5	busy	Bool	false	FALSE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7000
9	memErrStatus	Word	16#0	16#0000

18	TDISCON	Struct		
19	start	Bool	false	TRUE
20	busy	Bool	false	FALSE
21	done	Bool	false	TRUE
22	error	Bool	false	FALSE
23	status	Word	16#0	16#7000
24	memErrStatus	Word	16#0	16#0000

Netzwerk 3 (TRCV):

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "TRCV" die zu verwendende Kommunikationsverbindung bekannt.

Wenn der Eingangsparameter EN_R ("TRCV.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TRCV" gestartet. Über mehrere Aufrufe hinweg empfängt die Anweisung "TRCV" den übermittelten Datensatz. Am Eingangsparameter DATA ("SLI_plcDB_rcvData_TRCV") wird der Datensatz erfasst.

SLI_plcDB_rcvData_TRCV

	Name	Data type	Start value	Monitor value
1	Static			
2	prename	String	''	'Simon'
3	lastname	String	''	'Sample'
4	userId	UInt	0	1

Die Länge in BYTE des tatsächlich übermittelten Datensatzes wird über den Ausgangsparameter LEN ("#length") erfasst. Dessen Wert wird nur während des Erfolgsstatus angezeigt. Danach wird "0" erfasst. Über den Ausgangsparameter DONE ("#doneRCV") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TRCV.status") mit dem Wert "0000" wird das erfolgreiche Empfangen des Datensatzes angezeigt.

Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, machen Sie Folgendes:

- Sie speichern den Erfolgsstatus in der Variable "TRCV.done".
- Sie speichern die Länge in BYTE in der Variable "TRCV.readLength".

Am Ausgangsparameter ERROR ("TRCV.error") bzw. der Variable "TRCV.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

10	TRCV	Struct		
11	start	Bool	false	TRUE
12	busy	Bool	false	TRUE
13	done	Bool	false	TRUE
14	error	Bool	false	FALSE
15	status	Word	16#0	16#7002
16	memErrStatus	Word	16#0	16#0000
17	readLength	UInt	0	514

Netzwerk 4 (TRCV):

Wenn TRCV einen Fehler meldet ("TRCV.error" ist "TRUE"), wird der gemeldete Status ("TRCV.status") dauerhaft gespeichert ("TRCV.memErrStatus").

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

TCON: Kommunikationsverbindung aufbauen (Seite 3692)

TDISCON: Kommunikationsverbindung abbauen (Seite 3700)

TSEND: Daten über Kommunikationsverbindung senden (Seite 3705)

TRCV: Daten über Kommunikationsverbindung empfangen (Seite 3714)

Daten über Ethernet (UDP) oder FDL senden und empfangen

TUSEND: Daten senden

Beschreibung

TUSEND unterstützt die Datenübertragung über UDP (bei S7-1500-CPUs ab Firmware-Stand V2.0 über die integrierten PROFINET-Schnittstellen auch als Multicast-Kommunikation) und über FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0 mit dem Systemdatentyp TCON_FDL.

Die Anweisung "TUSEND" sendet Daten an den über den Parameter ADDR adressierten Kommunikationspartner.

⚠ WARNUNG

Datenübertragung über UDP

Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird.

Das Risiko von nicht erkennbaren Übertragungsfehlern erhöht sich deutlich, falls mehr als 1472 Bytes übertragen werden.

Hinweis

Bei aufeinander folgenden Sendevorgängen an verschiedene Partner müssen Sie bei den Aufrufen von "TUSEND" lediglich den Parameter ADDR anpassen. Der erneute Aufruf der Anweisungen "TCON (Seite 3689)" und "TDISCON (Seite 3700)" hingegen entfällt. Damit der jeweilige Partner die Daten empfangen kann, muss dort der adressierte UDP-Port eingerichtet sein.

Senden von mehr als 1472 Bytes

Ab Firmware-Stand V2.5 der S7-1500-CPUs können Sie bei Datenübertragung über UDP nicht nur bis zu 1472 Bytes, sondern bis zu 2048 Bytes senden, falls Sie Unicast oder Multicast verwenden.

Beim Senden von mehr als 1472 Bytes müssen Sie darauf achten, dass Ihr Empfänger mehr als 1472 Bytes unterstützt. Wenn dies nicht der Fall ist, bekommen Sie den fehlgeschlagenen Empfang auf Senderseite nicht mit.

Arbeitsweise

"TUSEND" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Sendevorgang, indem Sie am Parameter REQ eine steigende Flanke erzeugen.

Über die Ausgangsparameter BUSY, DONE, ERROR und STATUS wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TUSEND" aktuell befindet bzw. wann der Sendevorgang beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	FALSE	FALSE	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.

BUSY	DONE	ERROR	Beschreibung
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TUSEND" müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TUSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter REQUEST, startet den Sendeauftrag bei steigender Flanke. Die Daten werden aus dem mit DATA und LEN angegebenen Bereich übergeben.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung bei der Anweisung "TCON". Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen Wertebereich bei Ethernet/UDP: 1 bis 1472 (ab Firmware-Stand V2.5 der S7-1500-CPUs bei Unicast oder Multicast: 1 bis 2048) Bei FDL-Verbindungen des CM 1542-5 beträgt die maximale Länge 240 Byte. Beachten Sie dabei die maximale verarbeitbaren Längen beim Verbindungspartner.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt. Der Wert wird nur für einen Zyklus angezeigt.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	InOut	VARIANT	E, A, M, D	Sendebereich, enthält Adresse und Länge Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
ADDR	InOut	VARIANT	D	Adressinformationen des Kommunikationspartners Siehe auch: Adressierung des remoten Partners über TADDR_Param (Seite 3742) und Adressierung des remoten Partners über TADDR_SEND_QDN und TADDR_RCV_IP (Seite 3744) und AUTOHOTSPOT

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Sendeauftrag wurde ohne Fehler abgeschlossen
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Start der Auftragsbearbeitung, Daten werden versendet Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
0	7002	Zwischenaufruf (REQ irrelevant), Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
1	8085	Parameter LEN hat den Wert "0" oder ist größer als der größte zulässige Wert.
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich.
0	8088	Parameter LEN ist größer als der in DATA angegebene Speicherbereich.
1	8089	Parameter ADDR zeigt nicht auf einen Datenbaustein mit der Struktur TADDR_Param oder TADDR_SEND_QDN.
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut. • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich. • Schnittstelle wird neu initialisiert
1	80B1	Sie haben den Parameter DATA geändert, bevor der laufende Auftrag beendet wurde.
1	80A4	IP-Adresse (am Parameter ADDR) des remoten Verbindungsendpunkts ist ungültig, evtl. stimmt sie mit der eigenen IP-Adresse überein.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80B3	• Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TSEND (Seite 3705)". • Parameter ADDR: ungültige Angaben für Port-Nr.
1	80B7	Die Länge der durch den Parameter ADDR referenzierten Struktur ist nicht 8 Byte.
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert. • Temporär ist keine Empfangsressourcen beim Verbindungspartner verfügbar. Der Verbindungspartner ist nicht empfangsbereit. • Das R/H-System befindet sich im Systemzustand SYNCUP oder es findet eine Primary-Backup-Umschaltung statt. Die Verbindung wird geschlossen. Die Bearbeitung von TUSEND wird gestoppt.
1	80C5	• Verbindung durch den Kommunikationspartner abgebaut. • LSAP des remoten Verbindungspartners nicht freigegeben
1	80C6	Netzwerkfehler: • Remoter Partner kann nicht erreicht werden. • Physikalische Unterbrechung an PROFIBUS
1	80C7	Zeitüberschreitung der Ausführung.
1	80D3	Der voll qualifizierte Domänen-Name konnte nicht aufgelöst werden. Mögliche Ursachen: • Der DNS Server ist nicht erreichbar z. B. weil er heruntergefahren ist oder der remote Port nicht erreichbar ist. • Während der Kommunikation mit dem DNS Server ist ein Fehler aufgetreten. • Der DNS Server schickte eine gültige DNS-Antwort zurück, aber die Antwort enthielt keine IPv4-Adresse.
1	80EA	DTLS- (UDP-) Protokoll wird nicht unterstützt
1	8xyy	Allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TUSEND & TURCV (Seite 3745).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

TURCV: Daten empfangen

Beschreibung

Die Anweisung "TURCV" empfängt Daten über UDP (bei S7-1500-CPUs ab Firmware-Stand V2.0 über die integrierten PROFINET-Schnittstellen auch als Multicast-Kommunikation) und über FDL-Verbindungen der S7-1500 mit CM 1542-5 ab V2.0. Nach erfolgreichem Abschluss von "TURCV" wird Ihnen am Parameter ADDR die Adresse des remoten Partners, also des Senders, zur Verfügung gestellt.

⚠ WARNUNG
Ungesicherte Datenübertragung Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum remoten Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird.

Empfang von mehr als 1472 Bytes

Ab Firmware-Stand V2.5 der S7-1500-CPUs können Sie bei Datenübertragung über UDP nicht mehr bis zu 1472 Bytes, sondern bis zu 2048 Bytes empfangen.

Falls Ihre empfangende Baugruppe allerdings max. 1472 Bytes unterstützt und Ihre sendende Baugruppe mehr als 1472 Bytes überträgt, liefert TURCV im Parameter STATUS den Wert W#16#8085 oder W#16#80C9.

Arbeitsweise

"TURCV" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Empfangsauftrag, indem Sie "TURCV" mit EN_R = 1 aufrufen.

Über die Ausgangsparameter BUSY, DONE, ERROR und STATUS wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, NDR und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich TURCV aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

BUSY	NDR	ERROR	Beschreibung
TRUE	FALSE	FALSE	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen. Neue Daten wurden empfangen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TURCV" sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter NDR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TURCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter enabled to receive Mit EN_R = 1 wird "TURCV" empfangsbereit. Der Empfangsauftrag wird bearbeitet.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	UDINT	E, A, M, D, L oder Konstante	Länge des Empfangsbereichs in Byte: 0 (empfohlen) bzw. 1 bis 1472 (ab Firmware-Stand V2.5 der S7-1500-CPUs bei Unicast oder Multicast: 1 bis 2048) Größter Wert beim CM 1542-5 (FDL): 240
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch • NDR = 1: Auftrag wurde erfolgreich abgeschlossen
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Fehlerinformation
RCVD_LEN	Output	UDINT	E, A, M, D, L	Anzahl der tatsächlich empfangenen Daten in Bytes

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	InOut	VARIANT	E, A, M, D, L	Empfangsbereich Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Bei der Übertragung von Strukturen müssen die Strukturen auf der Sende- und der Empfangsseite identisch sein.
ADDR	InOut	VARIANT	D	Adressinformationen des remoten Partners Siehe auch: Adressierung des remoten Partners über TADDR_Param (Seite 3742) und Adressierung des remoten Partners über TADDR_SEND_QDN und TADDR_RCV_IP (Seite 3744) und AUTOHOTSPOT

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS* (W#16#...)	Erläuterung
0	0000	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.
0	7000	Baustein nicht empfangsbereit
0	7001	Baustein ist empfangsbereit, Empfangsauftrag wurde aktiviert
0	7002	Zwischenaufruf, Empfangsauftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase schreibt "TURCV" Daten in den Empfangsbereich. Deshalb können im Fehlerfall inkonsistente Daten im Empfangsbereich stehen.
1	8085	Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben einen der Parameter LEN oder DATA gegenüber dem Erstaufruf geändert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	• Empfangsbereich ist zu klein • Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich
1	8089	Parameter ADDR zeigt nicht auf einen Datenbaustein mit der Struktur TADDR_Param oder TADDR_RCV_IP.
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut. • Die angegebene Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Empfangsauftrag über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu parametriert.
1	80B1	Sie haben den Parameter DATA geändert, bevor der laufende Auftrag beendet wurde.
1	80B3	Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TRCV (Seite 3709)".
1	80B7	Die Länge am Parameter ADDR entspricht nicht 8 Byte.

ERROR	STATUS* (W#16#...)	Erläuterung
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel
1	80C4	Temporärer Kommunikationsfehler: • Die Schnittstelle wird neu parametriert. • Das R/H-System befindet sich im Systemzustand SYNCUP oder es findet eine Primary-Backup-Umschaltung statt. Die Verbindung wird geschlossen. Die Bearbeitung von TURCV wird gestoppt.
1	80C5	Der remote Partner hat die Verbindung abgebaut.
1	80C7	Zeitüberschreitung der Ausführung.
1	80C9	Bei RFC1006 / UDP: Die empfangenen Daten sind länger als erwartet (Größe des Empfangspuffers überschritten).
1	80EA	DTLS- (UDP-) Protokoll wird nicht unterstützt
1	8xyy	Allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für TUSEND & TURCV (Seite 3745).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

TCON: Kommunikationsverbindung aufbauen (Seite 3689)

TDISCON: Kommunikationsverbindung abbauen (Seite 3700)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Adressierung des remoten Partners über TADDR_Param

Übersicht

Bei Verwendung einer UDP-Verbindung können Sie die Adressinformationen des remoten Partners in dem Systemdatentyp TADDR_Param hinterlegen:

- Bei der Anweisung "TUSEND (Seite 3733)" übergeben Sie am Parameter ADDR über TADDR_Param die Adressinformation des Empfängers.
  Die hinterlegten Adressdaten des remoten Partners werden von der Anweisung aus dem Systemdatentyp gelesen.

  Hinweis

  Unterstützung eines Limited Broadcast

  Für S7-1500-CPUs ab Firmware-Version V2.0 gilt für die Anweisung TUSEND: Bei Adressierung des remoten Partners über den Systemdatentyp TADDR_Param wird ein Limited Broadcast (255.255.255.255 als IP-Adresse des remoten Partners) unter konsequenter Einhaltung von IP-Routing-Regeln unterstützt. Demzufolge werden Limited Broadcasts nur über diejenige Schnittstelle versendet, über die laut Projektierung ein Default-Router erreichbar ist.

  Es wird die Verwendung von Directed-Broadcast-Adressen empfohlen, da diese - den IP-Standards entsprechend - nicht dieser Einschränkung unterliegen.

- Bei der Anweisung "TURCV (Seite 3738)" erhalten Sie am Parameter ADDR über TADDR_Param die Adresse des Absenders.
  Die Adressdaten werden von der Anweisung in den Systemdatentyp geschrieben.

Aufbau der Adressinformation nach TADDR_Param

Der Systemdatentyp TADDR_Param enthält die Adressinformationen des remoten Partners, bestehend aus der IP-Adresse und der Port-Nummer.

Der Systemdatentyp TADDR_Param ist folgendermaßen aufgebaut:

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 bis 3	rem_ip_addr	ARRAY [1..4] of USINT	B#16#00 ...	• IP-Adresse des remoten Partners, z. B. 192.168.002.003: – rem_ip_addr[1] = B#16#C0 (192) – rem_ip_addr[2] = B#16#A8 (168) – rem_ip_addr[3] = B#16#02 (002) – rem_ip_addr[4] = B#16#03 (003) Die IP-Adresse entnehmen Sie in der Sicht Geräte & Netze den Eigenschaften der Schnittstelle des remoten Partners. Alternativ wird diese auch in den Eigenschaften der UDP-Verbindung unter Adressdetails angezeigt. • Multicast-Adresse einer IPv4-Multicast-Gruppe (für die Anweisung "TUSEND" bei S7-1500-CPUs ab Firmware-Stand V2.0. Hinweis: Für die Anweisung "TURCV" ist auch bei Multicast-Kommunikation die IP-Adresse des Senders anzugeben.)
4 bis 5	rem_port_nr	UINT	B#16#00 ...	Remote Port-Nummer (mögliche Werte siehe: AUTOHOT-SPOT): • rem_port_nr[1] = high byte der Port-Nr. in hexadezimaler Darstellung • rem_port_nr[2] = low byte der Port-Nr. in hexadezimaler Darstellung Die Port-Nummer entnehmen Sie in der Sicht Geräte & Netze den Eigenschaften der UDP-Verbindung. Unter Adressdetails wird die Port-Nummer als dezimaler Wert angezeigt. Beispiel: Port-Nummer = 2000 (dezimal) / W#16#07D0 (hexadezimal) • rem_port_nr[1] = 07 (high byte) • rem_port_nr[2] = D0 (low byte)
6 bis 7	reserved	WORD	B#16#00 ...	Nicht verwendet. Belassen Sie diesen Parameter auf dem Wert "0".

Anlegen von TADDR_Param in einem Datenbaustein

Zum Erstellen von TADDR_Param haben Sie folgende Möglichkeiten:

- Erstellen Sie einen neuen Datenbaustein, und wählen Sie im Dialog "Neuen Baustein hinzufügen" als Typ TADDR_Param aus.
- Öffnen Sie einen vorhandenen Datenbaustein, erstellen Sie eine neue Variable und geben Sie in der Spalte Datentyp TADDR_Param ein.

Ein Datenbaustein kann mehrere Systemdatentypen TADDR_Param enthalten.

Adressierung des remoten Partners über TADDR_SEND_QDN und TADDR_RCV_IP

Übersicht

Beim Senden von Daten über UDP können Sie für S7-1500-CPUs ab Firmware-Version V2.0 den Empfänger mit seinem voll qualifizierten Domainnamen adressieren. Dabei verweisen Sie bei der Anweisung TUSEND am Parameter ADDR auf eine Struktur vom Typ TADDR_SEND_QDN.

Der Empfänger kann eine IPv4- oder eine IPv6-Adresse zurückliefern. Verweisen Sie bei der Anweisung TURCV am Parameter ADDR daher auf eine Struktur vom Typ TADDR_RCV_IP. Nur diese kann beide IP-Adresstypen aufnehmen.

Aufbau der Adressinformation nach TADDR_SEND_QDN

Der Systemdatentyp TADDR_SEND_QDN enthält die Adressinformationen des remoten Partners (des Empfängers), bestehend aus dem voll qualifizierten Domainnamen und der Port-Nummer.

Er ist folgendermaßen aufgebaut:

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 bis 255	RemoteQDN	ARRAY of STRING[1..254]	-	Fully qualified domain name des Partner-Endpunkts, der mit "." abgeschlossen sein muss. Beachten Sie, dass in einer SIMATIC-Umgebung der Name einschließlich des abschließenden Punkts maximal 254 Zeichen lang sein darf.
256 bis 257	RemotePort	UINT	B#16#00 ...	Remote Port-Nummer (mögliche Werte siehe: AUTOHOT-SPOT): • RemotePort[1] = high byte der Port-Nr. in hexadezimaler Darstellung • RemotePort[2] = low byte der Port-Nr. in hexadezimaler Darstellung Beispiel: Port-Nummer = 2000 (dezimal) / W#16#07D0 (hexadezimal) • RemotePort[1] = 07 (high byte) • RemotePort[2] = D0 (low byte)
258 bis 259	reserved	WORD	W#16#00	Nicht verwendet. Belassen Sie diesen Parameter auf dem Wert "0".

Hinweis

Laufzeit der Anweisung TUSEND

Bei jedem Aufruf von TUSEND wird der voll qualifizierte Domainname in die IP-Adresse aufgelöst. Das hat zur Folge, dass die Ausführungszeit von TUSEND länger ist, als wenn sie den Empfänger direkt über seine IPv4-Adresse ansprechen.

Aufbau der Adressinformation nach TADDR_RCV_IP

Der Systemdatentyp TADDR_RCV_IP enthält die Adressinformationen des remoten Partners (des Senders), bestehend aus der IP-Adresse, der Port-Nr. und der Information, ob die IP-Adresse eine IPv4- oder eine IPv6-Adresse ist.

Er ist folgendermaßen aufgebaut:

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 bis 15	RemoteAddress	ARRAY [1..16] of BYTE	B#16#00 ...	IP-Adresse des remoten Partners, z. B. für die IPv4-Adresse 192.168.002.003: • RemoteAddress[1] = B#16#C0 (192) • RemoteAddress[2] = B#16#A8 (168) • RemoteAddress[3] = B#16#02 (002) • RemoteAddress[4] = B#16#03 (003) Für eine IPv6-Adresse gilt Entsprechendes.
16 bis 17	RemotePort	UINT	B#16#00 ...	Remote Port-Nummer (mögliche Werte siehe: AUTOHOTSPOT): • RemotePort[1] = high byte der Port-Nr. in hexadezimaler Darstellung • RemotePort[2] = low byte der Port-Nr. in hexadezimaler Darstellung Beispiel: Port-Nummer = 2000 (dezimal) / W#16#07D0 (hexadezimal) • RemotePort[1] = 07 (high byte) • RemotePort[2] = D0 (low byte)
18	RemAddrType	BYTE	B#16#00	Typ der remoten Adresse: • B#16#03: IPv4 • B#16#04: IPv6
19	reserved	BYTE	B#16#00	Nicht verwendet. Belassen Sie diesen Parameter auf dem Wert "0".

Programmbeispiel für TUSEND & TURCV

Einleitung

In dem folgenden Beispiel nutzen Sie eine konfigurierte UDP-Verbindung zwischen zwei CPUs der Baurreihe S7-1500. Mit Hilfe der Anweisungen "TUSEND" und "TURCV" senden Sie einen Datensatz von der CPU1 zur CPU2.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Eine UDP-Verbindung ist konfiguriert.

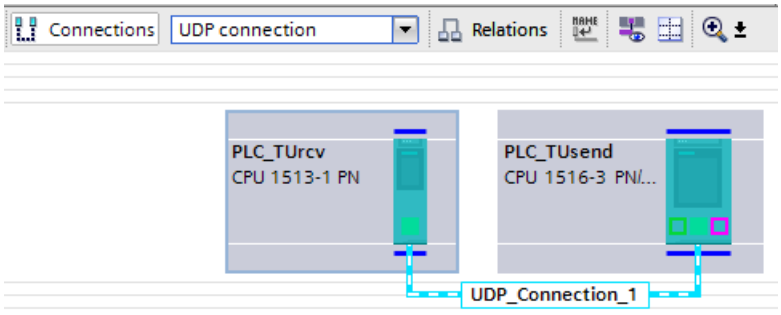

- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Programm der CPU 1

Für den Datensatz legen Sie den PLC-Datentyp "TUSEND_User" an.

TUSEND_User

	Name	Data type	Default value
1	prename	String	''
2	lastname	String	''
3	userId	UInt	0

Für die Datenübermittlung legen Sie auf Basis des PLC-Datentyps "TUSEND_User" den folgenden Datenbaustein ("SLI_plcDB_sendData_TUSEND") an.

SLI_plcDB_sendData_TUSEND

	Name	Data type	Start value
1	Static		
2	prename	String	'Simon'
3	lastname	String	'Sample'
4	userId	UInt	1

Für die Adressierung des Kommunikationspartners legen Sie auf Basis des Systemdatentyps "TADDR_Param" den folgenden Datenbaustein ("SLI_plcDB_taddr_param_TUSEND") an.

SLI_plcDB_taddr_param_TUSEND

	Name	Data type	Start value
1	Static		
2	REM_IP_ADDR	Array[1..4] of USInt	
3	REM_IP_ADDR[1]	USInt	192
4	REM_IP_ADDR[2]	USInt	168
5	REM_IP_ADDR[3]	USInt	0
6	REM_IP_ADDR[4]	USInt	3
7	REM_PORT_NR	UInt	2000
8	RESERVED	Word	16#0

Die Adressdaten finden Sie in der "Netzsicht" unter "Verbindungen > Eigenschaften > Adressdetails". In den Adressdetails werden unter "Lokal" die Daten der CPU angegeben, welche zuvor ausgewählt worden ist. Im folgenden Bild ist die CPU 1 ("TUsend") ausgewählt.

	Local	Partner
IP (dec):	192.168.0.2	192.168.0.3
Port (dec):	2000	2000
	Remote address assignment in block	Remote address assignment in block

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_TUSEND") mit den folgenden Strukturen und Variablen an.

SLI_gDB_TUSEND

	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	16#0100
3	TUSEND	Struct	
4	start	Bool	false
5	maxLength	UDInt	0
6	busy	Bool	false
7	done	Bool	false
8	error	Bool	false
9	status	Word	16#0
10	memErrStat	Word	16#0

Sie erstellen den FB "SLI_FB_TUSEND". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	doneSEND	Bool	false	Non-ret...
9	TUSEND_Instance	TUSEND		

Netzwerk 1: Die Parameter der Anweisung "TUSEND" verschalten Sie wie folgt.

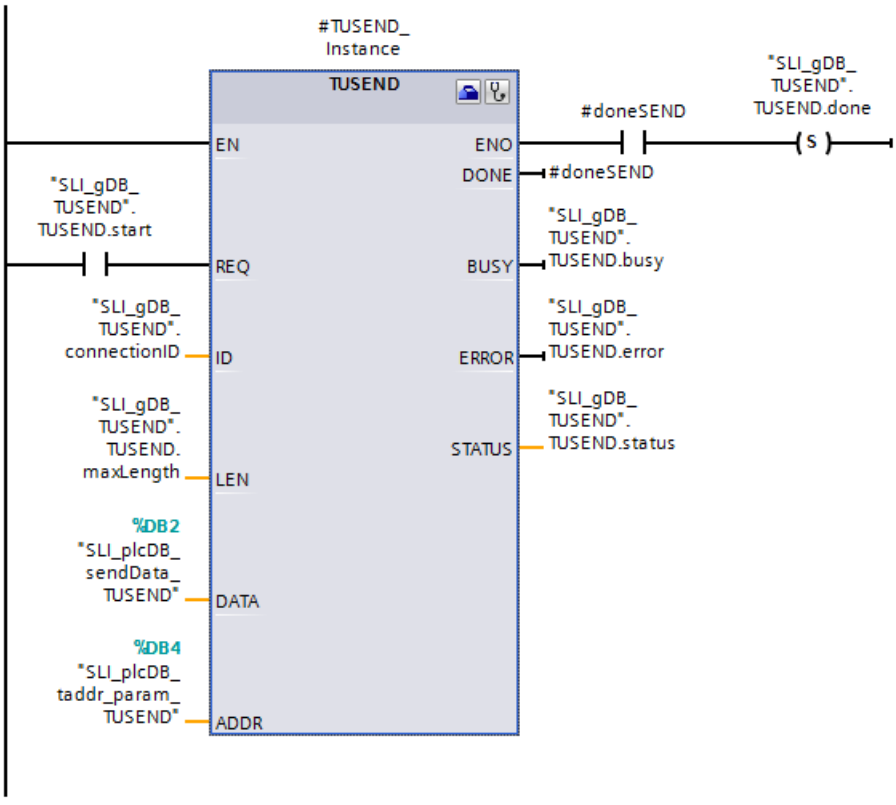

Netzwerk 2: Den Status im Fehlerfall von TUSEND speichern Sie wie folgt.

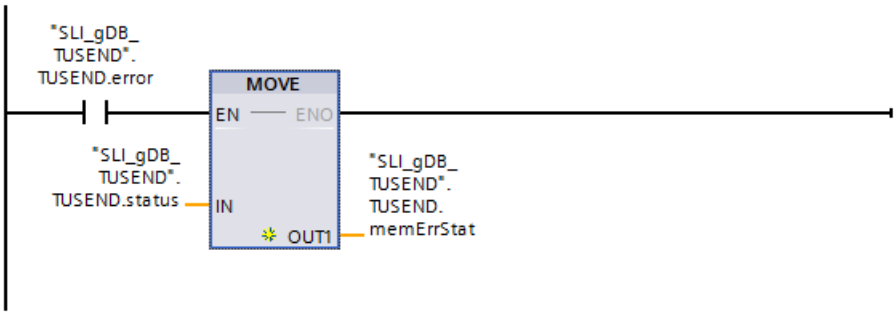

Programm der CPU 2

Für den Datensatz legen Sie nach Vorbild der CPU 1 den PLC-Datentyp "TUSEND_User" an.

Für den Datenempfang legen Sie auf Basis des PLC-Datentyps "TUSEND_User" den folgenden Datenbaustein ("SLI_plcDB_rcvData_TURCV") an.

SLI_plcDB_rcvData_TURCV

		Name	Data type	Start value
1		Static		
2		prename	String	''
3		lastname	String	''
4		userId	UInt	0

Für die Adressierung des Kommunikationspartners legen Sie auf Basis des Systemdatentyps "TADDR_Param" den folgenden Datenbaustein ("SLI_plcDB_taddr_param_TURCV") an.

SLI_plcDB_taddr_param_TURCV			
	Name	Data type	Start value
1	Static		
2	REM_IP_ADDR	Array[1..4] of USInt	
3	REM_IP_ADDR[1]	USInt	192
4	REM_IP_ADDR[2]	USInt	168
5	REM_IP_ADDR[3]	USInt	0
6	REM_IP_ADDR[4]	USInt	2
7	REM_PORT_NR	UInt	2000
8	RESERVED	Word	16#0

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_TURCV") mit den folgenden Strukturen und Variablen an.

SLI_gDB_TURCV			
	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	16#0100
3	TURCV	Struct	
4	start	Bool	false
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStat	Word	16#0
10	readLength	UDInt	0

Sie erstellen den FB "SLI_FB_TURCV". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	doneRCV	Bool	false	Non-ret...
9	length	UDInt	0	Non-retain
10	TURCV_Instance	TURCV		

Netzwerk 1: Die Parameter der Anweisung "TURCV" verschalten Sie wie folgt.

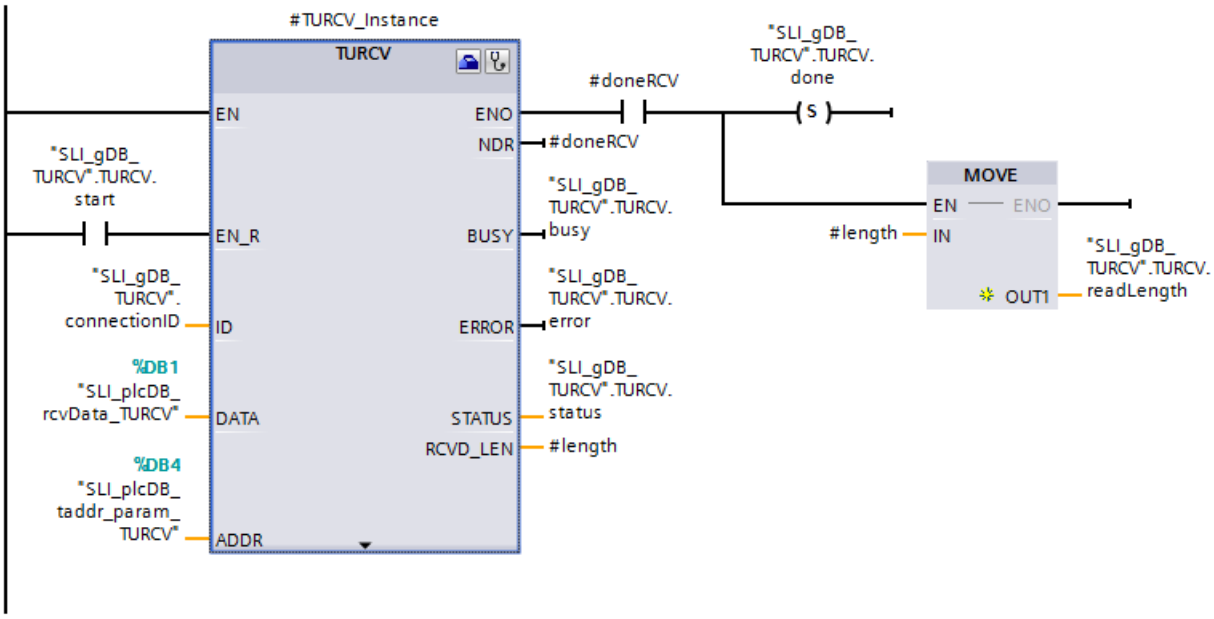

Netzwerk 2: Den Status im Fehlerfall von TURCV speichern Sie wie folgt.

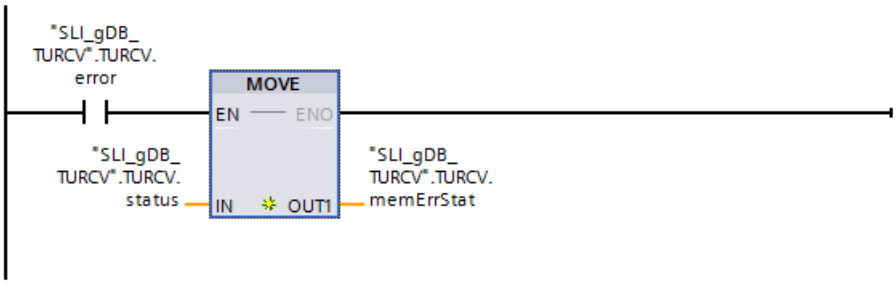

Kommunikationsverbindung zuweisen

Die Adressierungsparameter für die Verbindung der beiden CPUs müssen einheitlich sein.

- Hinterlegen Sie jeweils am Eingangsparameter ID ("connectionID") den hexadezimalen Wert der Hardware-Kennung der konfigurierten UDP-Verbindung.
  Die Hardware-Kennung finden Sie in der "Netzsicht" unter "Verbindungen".

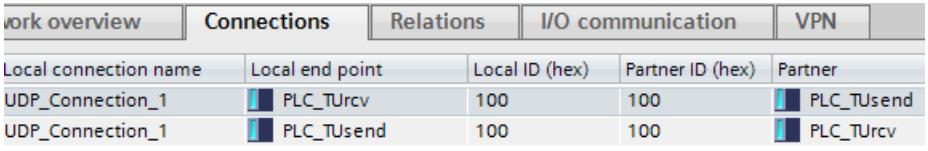

ork overview | Connections | Relations | I/O communication | VPN

Local connection name	Local end point	Local ID (hex)	Partner ID (hex)	Partner
UDP_Connection_1	PLC_TUrcv	100	100	PLC_TUsend
UDP_Connection_1	PLC_TUsend	100	100	PLC_TUrcv

Ergebnis der CPU 1

Netzwerk 1 (TUSEND):

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "TUSEND" die zu verwendende Kommunikationsverbindung bekannt. Durch den Eingangsparameter ADDR

("SLI_plcDB_taddr_param_TUSEND") kann der Kommunikationspartner (CPU 2) adressiert werden.

Wenn der Eingangsparameter REQ ("TUSEND.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TUSEND" gestartet. Über mehrere Aufrufe hinweg übermittelt die Anweisung "TUSEND" den am Eingangsparameter DATA ("SLI_plcDB_sendData_TUSEND") erfassten Datensatz. Entsprechend dem Wert des Eingangsparameters LEN ("TUSEND.maxLength") ist die Größe zu übermittelnden Datensatzes unbeschränkt.

Über den Ausgangsparameter DONE ("#doneSEND") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TUSEND.status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "TUSEND.done".

Am Ausgangsparameter ERROR ("TUSEND.error") bzw. der Variable "TUSEND.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_TUSEND

	Name	Data type	Start value	Monitor value
1	Static			
2	connectionID	CONN_OUC	16#0100	16#0100
3	TUSEND	Struct		
4	start	Bool	false	TRUE
5	maxLength	UDInt	0	0
6	busy	Bool	false	FALSE
7	done	Bool	false	TRUE
8	error	Bool	false	FALSE
9	status	Word	16#0	16#7000
10	memErrStat	Word	16#0	16#0000

Netzwerk 2 (TUSEND):

Wenn TUSEND einen Fehler meldet ("TUSEND.error" ist "TRUE"), wird der gemeldete Status ("TUSEND.status") dauerhaft gespeichert ("TUSEND.memErrStatus").

Ergebnis der CPU 2

Netzwerk 1 (TURCV):

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "TURCV" die zu verwendende Kommunikationsverbindung bekannt. Durch den Eingangsparameter ADDR ("SLI_plcDB_taddr_param_TURCV") kann der Kommunikationspartner (CPU 1) adressiert werden.

Wenn der Eingangsparameter EN_R ("TURCV.start") den Signalzustand "TRUE" liefert, wird die Anweisung "TURCV" gestartet. Über mehrere Aufrufe hinweg empfängt die Anweisung "TURCV" den übermittelten Datensatz. Am Eingangsparameter DATA ("SLI_plcDB_rcvData_TURCV") wird der Datensatz erfasst.

SLI_plcDB_rcvData_TURCV				
	Name	Data type	Start value	Monitor value
1	Static			
2	prename	String	''	'Simon'
3	lastname	String	''	'Sample'
4	userId	UInt	0	1

Die Länge in BYTE des tatsächlich übermittelten Datensatzes wird über den Ausgangsparameter LEN ("#length") erfasst. Dessen Wert wird nur während des Erfolgsstatus angezeigt. Danach wird "0" erfasst. Über den Ausgangsparameter DONE ("#doneRCV") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("TURCV.status") mit dem Wert "0000" wird das erfolgreiche Empfangen des Datensatzes angezeigt.

Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, machen Sie folgendes:

- Sie speichern den Erfolgsstatus in der Variable "TURCV.done".
- Sie speichern die Länge in BYTE in der Variable "TURCV.readLength".

Am Ausgangsparameter ERROR ("TURCV.error") bzw. der Variable "TURCV.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_TURCV				
	Name	Data type	Start value	Monitor value
1	Static			
2	connectionID	CONN_OUC	16#0100	16#0100
3	TURCV	Struct		
4	start	Bool	false	TRUE
5	busy	Bool	false	TRUE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7002
9	memErrStat	Word	16#0	16#0000
10	readLength	UDInt	0	514

Netzwerk 2 (TURCV):

Wenn TURCV einen Fehler meldet ("TURCV.error" ist "TRUE"), wird der gemeldete Status ("TURCV.status") dauerhaft gespeichert ("TURCV.memErrStatus").

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

TUSEND: Daten senden (Seite 3733)

TURCV: Daten empfangen (Seite 3738)

T_RESET: Verbindung zurücksetzen

Beschreibung

Die Anweisung "T_RESET" baut eine bestehende Verbindung ab und wieder auf.

Die lokalen Endpunkte der Verbindung bleiben dabei erhalten. Diese werden automatisch erzeugt:

- Wenn eine Verbindung konfiguriert und in die CPU geladen wurde.
- Wenn eine Verbindung durch das Anwenderprogramm erzeugt wurde, beispielsweise durch den Aufruf der Anweisung "TCON (Seite 3689)".

Die Anweisung "T_RESET" lässt sich für alle Verbindungsarten (TCP, UDP, ISO-on-TCP,...etc.) durchführen. Hierbei spielt es keine Rolle, ob für die Verbindung die lokale Schnittstelle der CPU oder die Schnittstelle eines CMs/CPs verwendet wurde.

Nach Aufruf der Anweisung "T_RESET" über den Parameter REQ wird die über den Parameter ID angegebene Verbindung abgebaut und ggf. die Puffer zum Senden und Empfangen von Daten geleert. Mit der Unterbrechung der Verbindung werden auch laufende Datenübertragungen unterbrochen. Dabei besteht die Gefahr, dass bei einer laufenden Übertragung Daten verloren gehen können. Danach versucht die als aktiver Verbindungspartner definierte CPU automatisch, die unterbrochene Kommunikationsverbindung wieder herzustellen. Der Aufruf der Anweisung "TCON (Seite 3689)" zum erneuten Aufbauen der Kommunikationsverbindung ist deshalb nicht notwendig.

Über die Ausgangsparameter DONE, BUSY und STATUS wird der Zustand des Auftrags angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_RESET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter REQUEST, startet den Auftrag zum Abbau der durch ID gegebenen Verbindung. Der Auftragsstart erfolgt bei steigender Flanke.
ID	Input	CONN_OUC	E, A, M, D, L oder Konstante	Referenz auf die abzubauende Verbindung zum passiven Partner. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Zustandsparameter BUSY • 0: Der Auftrag ist beendet. • 1: Der Auftrag ist noch nicht beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR • 0: Kein Fehler aufgetreten. • 1: Bei der Bearbeitung ist ein Fehler aufgetreten. Der Parameter STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS Fehlerinformation (siehe Tabelle "Parameter STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

STATUS* (W#16#...)	Erläuterung
0000	Kein Fehler.
0001	Verbindung ist nicht aufgebaut.
7000	Keine Auftragsbearbeitung aktiv.
7001	Unterbrechung der Verbindung wurde eingeleitet.
7002	Unterbrechung der Verbindung wird durchgeführt.
8081	Angegebene Verbindung am Parameter ID ist unbekannt.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für T_RESET (Seite 3754).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für T_RESET

Einleitung

In dem folgenden Beispiel nutzen Sie eine konfigurierte Verbindung (z. B. per UDP) zwischen zwei CPUs (S7-1500). Mit Hilfe der Anweisung "T_RESET" setzen Sie die Verbindung zurück.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Eine UDP-Verbindung ist konfiguriert.

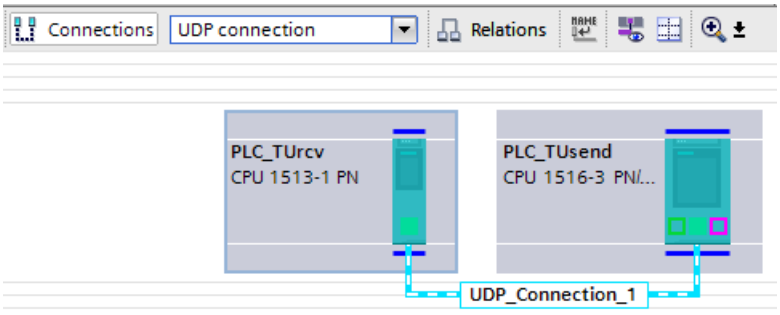

- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Variablen anlegen und Parameter verschalten (Programm der CPU 1)

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_T_RESET") mit den folgenden Strukturen und Variablen an.

SLI_gDB_T_RESET

	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	16#0100
3	T_RESET	Struct	
4	start	Bool	false
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStatus	Word	16#0

Sie erstellen den FB "SLI_FB_T_RESET". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	done	Bool	false	Non-ret...
9	T_RESET_Instance	T_RESET		

Netzwerk 1: Die Parameter der Anweisung "T_RESET" verschalten Sie wie folgt.

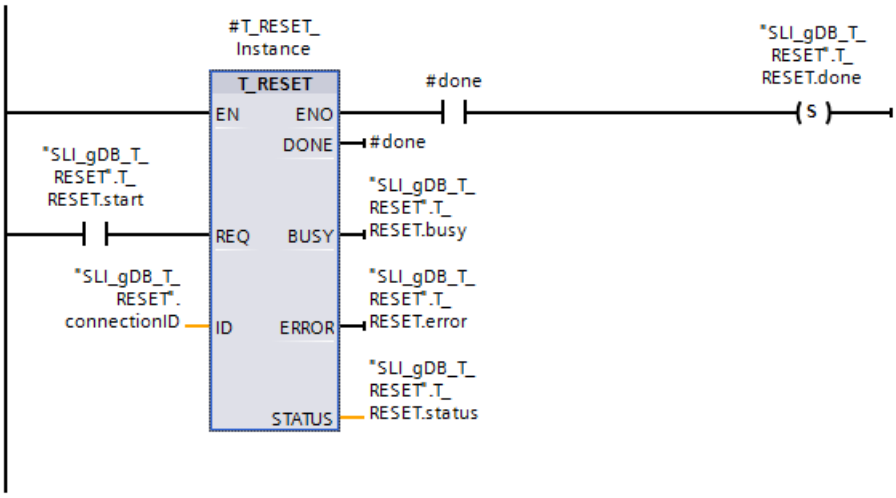

Netzwerk 2: Den Status im Fehlerfall von T_RESET speichern Sie wie folgt.

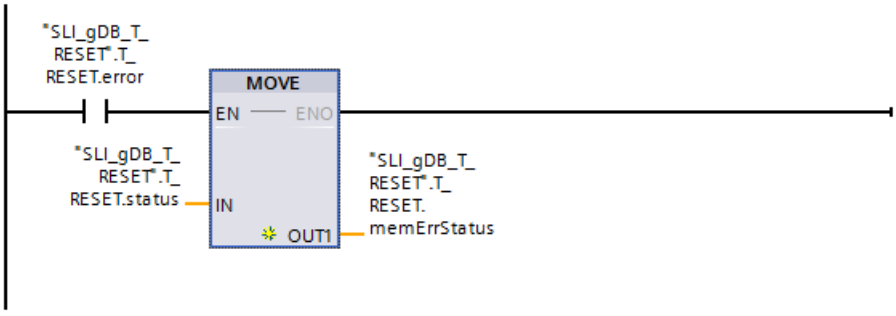

Netzwerk 3: Zum Rücksetzen des Parameters REQ legen Sie folgende Verschaltungen an.

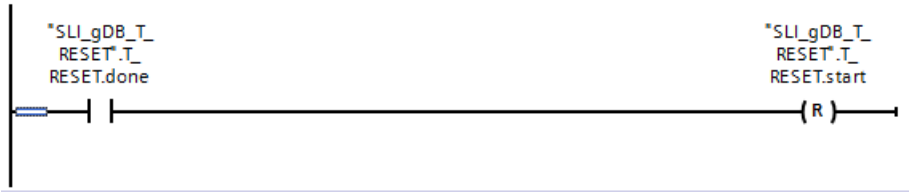

Kommunikationsverbindung zuweisen

Der Adressierungsparameter für die Verbindung muss angepasst werden.

- Hinterlegen Sie am Eingangsparameter ID ("connectionID") den hexadezimalen Wert der Hardware-Kennung der konfigurierten UDP-Verbindung.
  Die Hardware-Kennung finden Sie in der "Netzsicht" unter "Verbindungen".

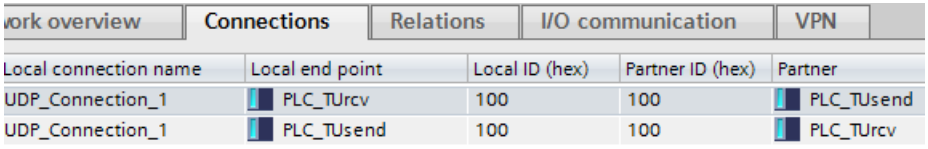

ork overview | Connections | Relations | I/O communication | VPN

Local connection name	Local end point	Local ID (hex)	Partner ID (hex)	Partner
UDP_Connection_1	PLC_TUrcv	100	100	PLC_TUsend
UDP_Connection_1	PLC_TUsend	100	100	PLC_TUrcv

Ergebnis

Netzwerk 1:

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "T_RESET" die zu verwendende Kommunikationsverbindung bekannt.

Wenn der Eingangsparameter REQ ("T_RESET.start") den Signalzustand "TRUE" liefert, wird die Anweisung "T_RESET" gestartet. Die Anweisung "T_RESET" baut die angegebene Kommunikationsverbindung ab. Alle Aufträge, welche die Verbindung nutzen, werden abgebrochen. Der aktive Verbindungspartner (CPU 1) stellt anschließend die Verbindung automatisch wieder her.

Über den Ausgangsparameter DONE ("#done") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("T_RESET.status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt.
Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "T_RESET.done".

Am Ausgangsparameter ERROR ("T_RESET.error") bzw. der Variable "T_RESET.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_T_RESET

	Name	Data type	Start value	Monitor value
1	Static			
2	connectionID	CONN_OUC	16#0100	16#0100
3	T_RESET	Struct		
4	start	Bool	false	FALSE
5	busy	Bool	false	FALSE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7000
9	memErrStatus	Word	16#0	16#0000

Netzwerk 2:

Wenn T_RESET einen Fehler meldet ("T_RESET.error" ist "TRUE"), wird der gemeldete Status ("T_RESET.status") dauerhaft gespeichert ("T_RESET.memErrStatus").

Netzwerk 3:

Nach erfolgreicher Verarbeitung von T_RESET ("T_RESET.done") wird T_RESET gestoppt.

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

T_RESET: Verbindung zurücksetzen (Seite 3753)

T_DIAG: Verbindung überprüfen

Beschreibung

Mit der Anweisung "T_DIAG" überprüfen Sie den Zustand einer Verbindung und lesen weitere Informationen zum lokalen Endpunkt dieser Verbindung.

- Die Verbindung wird referenziert über den Parameter ID. Es können sowohl im Verbindungseditor projektierte, als auch programmierte Verbindungsendpunkten (z. B. mit der Anweisung "TCON") gelesen werden.
  Temporäre Verbindungsendpunkte (die z. B. beim Verbinden einer Engineering-Station entstehen) können nicht diagnostiziert werden, da hierbei auch keine Verbindungs-ID erstellt wird.
- Die gelesenen Verbindungsinformationen werden in einer Struktur abgelegt, die über den Parameter RESULT referenziert ist.
- Der Ausgangsparameter STATUS zeigt an, ob die Verbindungsinformationen gelesen werden konnten. Nur wenn die Anweisung "T_DIAG" mit STATUS = W#16#0000 und ERROR = FALSE beendet wurde, sind die Verbindungsinformationen in der Struktur am Parameter RESULT gültig.
  Im Fehlerfall können die Verbindungsinformationen nicht ausgewertet werden.

Mögliche Verbindungsinformationen

Zum Lesen der Verbindungsinformationen können am Parameter RESULT zwei unterschiedliche Strukturen verwendet werden:

- Die Struktur "TDiag_Status" enthält nur die wichtigsten Informationen zu einem Verbindungsendpunkt, wie das verwendete Protokoll, der Verbindungszustand oder die Anzahl der gesendeten oder empfangenen Daten.
- Die Struktur "TDiag_StatusExt" (nur bei S7-1500-CPUs) liefert neben den wichtigsten Informationen auch die Anzahl der Verbindungsaufbauversuche, den Grund für einen möglichen Verbindungsabbruch, usw.

Der Aufbau und die Parameter der beiden Strukturen sind weiter unten beschrieben (siehe Tabelle "Strukturen TDIAG_Status und TDIAG_StatusExt").

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_DIAG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Startet bei einer positiven Flanke die Anweisung zum Überprüfen der im Parameter ID angegebenen Verbindung.
ID	Input	CONN_OUC (WORD)	E, A, M, D, L oder Konstante	Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESULT	InOut	VARIANT	D	Zeiger auf die Struktur, in der die Verbindungsinformationen abgelegt werden. An dem Parameter RESULT können die Strukturen TDiag_Status oder TDiag_StatusExt (nur bei S7-1500-CPUs) verwendet werden (zur Beschreibung siehe Tabelle "Strukturen TDIAG_Status und TDIAG_StatusExt").
DONE	Output	BOOL	E, A, M, D, L	Statusparameter: • 0: Anweisung noch nicht gestartet oder noch in Bearbeitung. • 1: Anweisung fehlerfrei ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter: • 0: Anweisung noch nicht gestartet oder bereits beendet. • 1: Anweisung noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter: • 0: Kein Fehler. • 1: Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus der Anweisung "T_DIAG" kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob die Anweisung erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "T_DIAG" auftreten.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Die Anweisung wird bearbeitet.
0	1	0	Die Anweisung wurde erfolgreich durchgeführt. Nur in diesem Fall sind die Daten in der mit RESULT referenzierten Struktur gültig.
0	0	1	Die Anweisung wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Es wurde keine neue Anweisung zugewiesen.

Parameter STATUS

Die folgende Tabelle zeigt die Bedeutung der Werte am Parameter STATUS:

STATUS* (W#16#...)	Erläuterung
0000	Die Anweisung "T_DIAG" wurde erfolgreich ausgeführt. Die Daten innerhalb der am Parameter RESULT referenzierten Struktur können ausgewertet werden.
7000	Keine Anweisungsbearbeitung aktiv.
7001	Anweisungsbearbeitung gestartet.
7002	Die Verbindungsinformationen werden gelesen (Parameter REQ irrelevant).
8086	Der Wert am Parameter ID ist nicht im erlaubten Bereich (W#16#0001 ... W#16#0FFF).
8089	Der Parameter RESULT zeigt auf einen ungültigen Datentyp (nur Strukturen TDIAG_Status und TDIAG_StatusExt erlaubt).
80A3	Der Parameter ID referenziert einen Verbindungsendpunkt, der nicht existiert. Dieser Fehler tritt bei programmierten Verbindungen auch nach dem Aufruf der Anweisung "TDISCON" auf.
80B1	Der Parameter RESULT wurde verändert, bevor die Ausführung von T_DIAG beendet wurde. Eine Veränderung von RESULT während der Ausführung von T_DIAG ist nicht erlaubt.
80C4	Interner Fehler. Der Zugriff auf den Verbindungsendpunkt ist temporär nicht möglich.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Strukturen TDIAG_Status und TDIAG_StatusExt

Hinweis

Struktur TDIAG_StatusExt

Die Struktur TDIAG_StatusExt gibt es nur bei S7-1500-CPUs.

Die folgende Tabelle beschreibt den Aufbau der Strukturen TDIAG_Status und TDIAG_StatusExt:

- Die Struktur TDIAG_StatusExt ist von dem Parameter InterfaceID bis zum Parameter ReceivedBytes identisch zu der Struktur TDIAG_Status.
- Die Struktur TDIAG_StatusExt enthält darüber hinaus die Parameter ConnTrials bis LastDisconnTimeStamp.

Alle Elemente haben nur dann einen gültigen Wert, wenn die Anweisung ohne Fehler ausgeführt werden konnte. Im Fehlerfall bleibt der Inhalt der Parameter unverändert.

Name	Datentyp	Beschreibung
Die nachfolgenden Parameter existieren sowohl bei der Struktur TDIAG_Status als auch bei der Struktur TDIAG_StatusExt:		
InterfaceID	HW_ANY	Schnittstellen-ID (LADDR) der CPU oder des CM/CP.
ID	CONN_OUC	ID der Verbindung, die diagnostiziert wurde. Nach einem erfolgreichen Aufruf ist der Wert dieses Elements mit dem Parameter ID der Anweisung "T_DIAG" identisch.

Name	Datentyp	Beschreibung
ConnectionType	BYTE	Bei der Verbindung verwendeter Protokolltyp: • 0x01: Nicht verwendet. • ... • 0x0B: TCP-Protokoll (IP_v4) • 0x0C: ISO-on-TCP-Protokoll (RFC1006) • 0x0D: TCP-Protokoll (DNS) • 0x0E: Dial-in-Protokoll • 0x0F: WDC-Protokoll • 0x10: SMTP-Protokoll • 0x11: TCP-Protokoll • 0x12: TCP und ISO-on-TCP Protokoll (RFC1006) • 0x13: UDP-Protokoll • 0x14: reserviert • 0x15: PROFIBUS-Buszugriffsprotokoll (FDL) • 0x16: ISO 8073-Transportprotokoll (ISOnative) • ... • 0x20: SMTP- oder SMTPS-Protokoll - basierend auf IPv4 • 0x21: SMTP- oder SMTPS-Protokoll - basierend auf IPv6 • 0x22: SMTP- oder SMTPS-Protokoll - basierend auf FQDN (Fully Qualified Domain Name) • ... • 0x70: S7-Verbindung • Weitere: reserviert
ActiveEstablished	BOOL	• FALSE: Es handelt sich lokal um den passiven Verbindungsendpunkt • TRUE: Es handelt sich lokal um den aktiven Verbindungsendpunkt
State	BYTE	Aktueller Zustand des Verbindungsendpunkts: • 0x00: Nicht verwendet. • 0x01: Verbindung ist abgebaut. Temporärer Zustand, der z. B. nach Aufruf der Anweisung "T_RESET" eingenommen wird. Danach versucht das System selbstständig wieder einen erneuten Verbindungsaufbau. • 0x02: Der aktive Verbindungsendpunkt versucht, die Verbindung zum remoten Kommunikationspartner aufzubauen. • 0x03: Der passive Verbindungsendpunkt wartet auf den Verbindungsaufbau des remoten Kommunikationspartners. • 0x04: Verbindung ist aufgebaut. • 0x05: Die Verbindung wird gerade abgebaut. Grund für den Abbau kann der Aufruf der Anweisung "T_RESET" oder "T_DISCON" sein. Weitere mögliche Gründe können auch ein Protokollfehler oder ein Leitungsbruch sein. • 0x06..0xFF: Nicht verwendet.

Name	Datentyp	Beschreibung
Kind	BYTE	Betriebsart des Verbindungsendpunkts: • 0x00: Nicht verwendet. • 0x01: Konfigurierte, statische Verbindung, die projektiert und anschließend auf die CPU geladen wurde. • 0x02: Konfigurierte, dynamische Verbindung, die projektiert und anschließend auf die CPU geladen wurde (wird derzeit nicht unterstützt). • 0x03: Programmierte Verbindung, die aus dem Anwenderprogramm mit der Anweisung "TCON" erzeugt wurde. Durch einen Aufruf der Anweisung "TDISCON" oder ein Übergang in den CPU-Zustand STOPP wurde der Verbindungsendpunkt wieder zerstört. • 0x04: Temporäre, dynamische Verbindung, die z. B. durch die Engineering-Station (ES) oder Operator Station (OS) aufgebaut wird. (diese Verbindungsart kann derzeit wegen der fehlenden ID nicht diagnostiziert werden). • 0x05..0xFF: Nicht verwendet.
SentBytes	UDINT	Anzahl der gesendeten Daten.
ReceivedBytes	UDINT	Anzahl der empfangenen Daten.
Die nachfolgenden Parameter existieren nur bei der Struktur TDiag_StatusExt:		
ConnTrials	UDINT	Anzahl der Verbindungsaufbauversuche. Nach einem erfolgreichen Verbindungsaufbau enthält ConnTrials den Wert 0. Ist das Element ungleich 0, deutet dies auf Verbindungsprobleme hin. Hinweis: Bei einem passiven Verbindungsendpunkt ist dieser Wert nie größer als 1.
ConnTrialsSuccess	UDINT	Anzahl der erfolgreichen Verbindungsaufbauversuche. Dieses Element wird während des Lebenszyklus eines Verbindungsendpunkts nie zurückgesetzt und bricht nach dem Wert 0xFFFF FFFF wieder auf 0 um. Hinweis: Wenn es noch nie ein Problem auf dieser Verbindung gab, hat dieser Parameter den Wert 1.

Name	Datentyp	Beschreibung
LastConnErrReason	UDINT	Fehlerkennung, die beim letzten fehlerhaften Versuch eines Verbindungsaufbaus auftrat (die Fehlermeldungen sind identisch zu denen am Parameter LastDisconnReason): • 0x4F01: Remoter Verbindungsendpunkt nicht erreichbar (dieser Fehler tritt normalerweise während der Phase des Verbindungsaufbaus auf). • 0x4F02: Verbindung wurde lokal abgebaut. • 0x4F03: Die Verbindung wurde durch den remoten Kommunikationspartner abgebaut. • 0x4F04: Die Verbindung wurde durch einen Protokollfehler abgebaut. • 0x4F05: Die Verbindung wurde durch ein lokal erkanntes Netzwerkproblem abgebaut. • 0x4F06: Die Verbindung wurde durch ein remote erkanntes Netzwerkproblem abgebaut. • 0x4F07: Die Verbindung wurde auf Grund einer Zeitüberschreitung im Protokoll abgebaut. • 0x4F08: Falsche Parametrierung: Der Verbindungsaufbau soll zur eigenen Adresse erfolgen. • 0x4F09: Die Verbindung wurde temporär durch den Aufruf der Anweisung "T_RESET" zurückgesetzt. • 0x4F0A: Zu wenig Verbindungsressourcen verfügbar (Mengengerüst überschritten) • 0x4F0B: Interner Fehler: falsche Adressierungsparameter • 0x4F0C: CPU-interner Kommunikationsfehler • 0x4F0D: AS-interner Kommunikationsfehler zwischen CPU und CM/CP • 0x4F0E: Der angegebene lokale TCP/UDP–Port (bzw. RFC1006-TSelector) ist bereits verwendet.
LastConnErrTimeStamp	LDT	Zeitpunkt des letzten fehlerhaften Verbindungsaufbauversuchs.

Name	Datentyp	Beschreibung
LastDisconnRea-son	UDINT	Fehlerkennung, die zum letzten Verbindungsabbau geführt hat (die Fehlermeldungen sind identisch zu denen am Parameter LastConnErrReason): • 0x4F01: Remoter Verbindungsendpunkt nicht erreichbar (dieser Fehler tritt normalerweise während der Phase des Verbindungsaufbaus auf). • 0x4F02: Verbindung wurde lokal abgebaut. • 0x4F03: Die Verbindung wurde durch den remoten Kommunikationspartner abgebaut. • 0x4F04: Die Verbindung wurde durch einen Protokollfehler abgebaut. • 0x4F05: Die Verbindung wurde durch ein lokal erkanntes Netzwerkproblem abgebaut. • 0x4F06: Die Verbindung wurde durch ein remote erkanntes Netzwerkproblem abgebaut. • 0x4F07: Die Verbindung wurde auf Grund einer Zeitüberschreitung im Protokoll abgebaut. • 0x4F08: Falsche Parametrierung: Der Verbindungsaufbau soll zur eigenen Adresse erfolgen. • 0x4F09: Die Verbindung wurde temporär durch den Aufruf der Anweisung "T_RESET" zurückgesetzt. • 0x4F0A: Zu wenig Verbindungsressourcen verfügbar (Mengengerüst überschritten) • 0x4F0B: Interner Fehler: falsche Adressierungsparameter • 0x4F0C: CPU-interner Kommunikationsfehler • 0x4F0D: AS-interner Kommunikationsfehler zwischen CPU und CM/CP • 0x4F0E: Der angegebene lokale TCP/UDP–Port (bzw. RFC1006-TSelector) ist bereits verwendet.
LastDisconnTi-meStamp	LDT	Zeitpunkt des letzten Verbindungsabbaus.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für T_DIAG (Seite 3764).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Programmbeispiel für T_DIAG

Einleitung

In dem folgenden Beispiel nutzen Sie eine konfigurierte Verbindung (z. B. per UDP) zwischen zwei CPUs (S7-1500). Mit Hilfe der Anweisung "T_DIAG" diagnostizieren Sie die Verbindung und lesen Informationen über den lokalen Endpunkt der Verbindung aus.

Voraussetzung

- Zwei CPUs der Baurreihe S7-1500 sind angelegt und per PROFINET miteinander verbunden. Eine UDP-Verbindung ist konfiguriert.

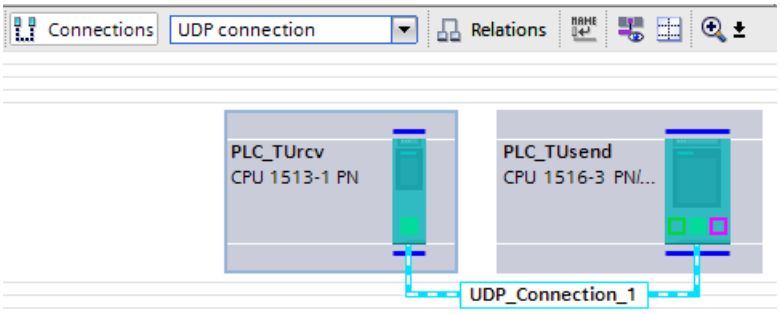

- Für die CPUs ist unter "<CPU> > Eigenschaften > Schutz" mit einer niedrigen Schutzstufe sichergestellt, dass Lese- und Schreibzugriffe erlaubt sind.

Variablen anlegen und Parameter verschalten (Programm der CPU 1)

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_T_DIAG") mit den folgenden Strukturen und Variablen an.

Hinweis: "TDiag_StatusExt" ist ein Systemdatentyp zum Erfassen der Diagnosedaten.

SLI_gDB_T_DIAG

	Name	Data type	Start value
1	Static		
2	connectionID	CONN_OUC	16#0100
3	T_DIAG	Struct	
4	start	Bool	false
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	status	Word	16#0
9	memErrStatus	Word	16#0
10	connectionDiag	TDiag_StatusExt	

Sie erstellen den FB "SLI_FB_T_DIAG". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	T_DIAG_Instance	T_DIAG		
9	done	Bool	false	Non-retain

Netzwerk 1: Die Parameter der Anweisung "T_DIAG" verschalten Sie wie folgt.

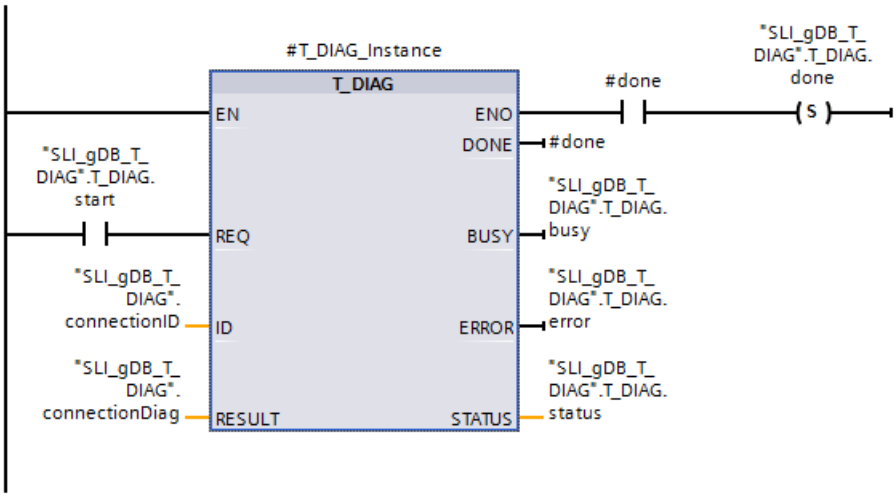

Netzwerk 2: Den Status im Fehlerfall von T_DIAG speichern Sie wie folgt.

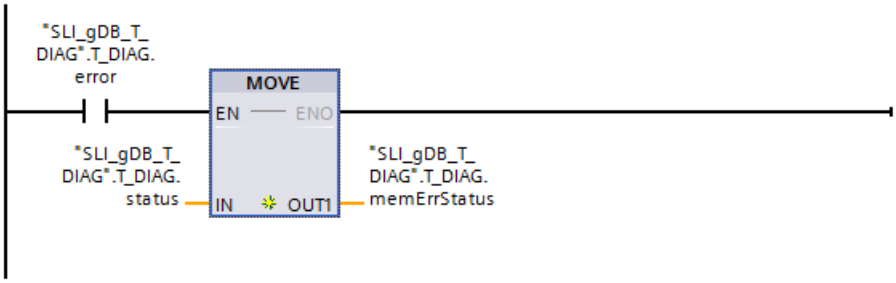

Kommunikationsverbindung zuweisen

Der Adressierungsparameter für die Verbindung muss angepasst werden.

- Hinterlegen Sie am Eingangsparameter ID ("connectionID") den hexadezimalen Wert der Hardware-Kennung der konfigurierten UDP-Verbindung.
  Die Hardware-Kennung finden Sie in der "Netzsicht" unter "Verbindungen".

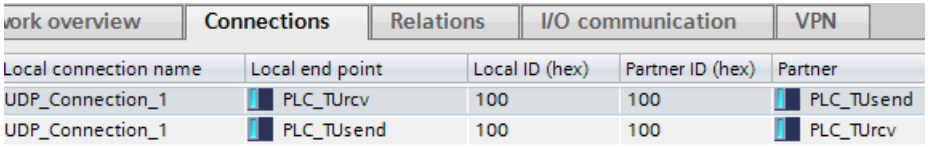

ork overview | Connections | Relations | I/O communication | VPN

Local connection name	Local end point	Local ID (hex)	Partner ID (hex)	Partner
UDP_Connection_1	PLC_TUrcv	100	100	PLC_TUsend
UDP_Connection_1	PLC_TUsend	100	100	PLC_TUrcv

Ergebnis

Netzwerk 1:

Entsprechend dem Eingangsparameter ID ("connectionID") ist der Anweisung "T_DIAG" die zu verwendende Kommunikationsverbindung bekannt.

Wenn der Eingangsparameter REQ ("T_DIAG.start") den Signalzustand "TRUE" liefert, wird die Anweisung "T_DIAG" gestartet. Die Anweisung "T_DIAG" liesst den Zustand der Verbindung und den Zustand des lokalen Endpunkts der Verbindung aus.

Über den Ausgangsparameter DONE ("#done") mit dem Signalzustand "TRUE" und den Ausgangsparameter STATUS ("T_DIAG.status") mit dem Wert "0000" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie den Erfolgsstatus in der Variable "T_DIAG.done".

Am Ausgangsparameter ERROR ("T_DIAG.error") bzw. der Variable "T_DIAG.memErrStatus" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_T_DIAG

	Name	Data type	Start value	Monitor value
1	Static			
2	connectionID	CONN_OUC	16#0100	16#0100
3	T_DIAG	Struct		
4	start	Bool	false	TRUE
5	busy	Bool	false	FALSE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#7000
9	memErrStatus	Word	16#0	16#0000

Am Parameter RESULT ("TDiag_StatusExt") werden die Diagnosedaten erfasst.

10	connectionDiag	TDiag_StatusExt		
11	InterfaceId	HW_ANY	16#0	16#0040
12	ID	CONN_OUC	16#0	16#0100
13	ConnectionType	Byte	0	16#13
14	ActiveEstablished	Bool	false	FALSE
15	State	Byte	0	16#04
16	Kind	Byte	0	16#01
17	SentBytes	UDInt	0	0
18	ReceivedBytes	UDInt	0	0
19	ConnTrials	UDInt	0	0
20	ConnTrialsSuccess	UDInt	0	2
21	LastConnErrReason	UDInt	0	0
22	LastConnErrTimeSt...	LDT	LDT#1970-01-01-(	LDT#1970-01-01-0...
23	LastDisconnReason	UDInt	0	20226
24	LastDisconnTimeS...	LDT	LDT#1970-01-01-(	LDT#2012-01-21-0...

Netzwerk 2:

Wenn T_DIAG einen Fehler meldet ("T_DIAG.error" ist "TRUE"), wird der gemeldete Status ("T_DIAG.status") dauerhaft gespeichert ("T_DIAG.memErrStatus").

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

T_DIAG: Verbindung überprüfen (Seite 3758)

T_CONFIG: Interface konfigurieren

Beschreibung T_CONFIG

Beschreibung

Die Anweisung "T_CONFIG" dient zur programmgesteuerten Konfiguration der integrierten PROFINET-Schnittstellen der CPU oder der Schnittstelle eines CPs/CMs.

Konfiguration der integrierten PROFINET-Schnittstellen der CPU

Über die Anweisung "T_CONFIG" können Sie aus dem Anwenderprogramm heraus die Ethernet-Adresse, den PROFINET-Gerätenamen oder die IP-Adressen der NTP-Server für die Uhrzeitsynchronisation ändern. Die bisher gültigen Konfigurationsdaten werden dabei überschrieben.

Die folgenden Änderungen können Sie vornehmen:

- Einstellungen für das IP-Protokoll
  - IP-Adresse
  - Subnetzmaske
  - Router-Adresse
- Einstellungen für PROFINET
  - Vergabe des PROFINET-Gerätenamens

    Hinweis
    Abhängigkeit zwischen Gerätenamen und IP-Protokoll

    Bei Vergabe eines neuen Gerätenamens muss auch das IP-Protokoll eingestellt werden.
- Einstellungen für die Uhrzeitsynchronisation
  - Vergabe der IP-Adressen der NTP-Server für die Uhrzeitsynchronisation (nur bei S7-1500 und dort nur für die PROFINET-Schnittstelle [X1] einer Modular-CPU)

Die Einstellungen entsprechen den Konfigurationsoptionen unter "IP-Protokoll" und "PROFINET" in dem Dialog "Ethernet-Adressen" bzw. unter "NTP-Verfahren" im Dialog

"Uhrzeitsynchronisation". Dieser wird in der Sicht "Geräte & Netze" unter den Eigenschaften der PROFINET-Schnittstelle angezeigt.

⚠ WARNUNG

Neustart der CPU nach Ausführung der Anweisung "T_CONFIG" (nur S7-1200-CPUs der Firmware-Versionen V2.0 bis V4.1.2)

Nachdem Sie die Anweisung ausgeführt haben, um einen IP-Parameter zu ändern, wird die CPU neu gestartet. Die CPU geht in den Betriebszustand STOP, ein Warmstart wird ausgeführt und die CPU läuft erneut an (Betriebszustand RUN).

Stellen Sie sicher, dass sich der Steuerungsprozess in einem sicheren Betriebszustand befindet, nachdem die CPU nach der Ausführung der Anweisung neu gestartet wurde. Ein unkontrollierter Betrieb, verursacht durch z. B. Funktionsstörungen oder Programmfehler, kann zu schweren Sach- und Personenschäden führen. Nicht remanente Daten können verloren gehen.

Konfiguration der PROFINET-Schnittstelle eines CP 1543-1

Über die Anweisung "T_CONFIG" können Sie aus dem Anwenderprogramm heraus die IPv6-Adresse, die MAC-Adresse oder die IP-Adressen von bis zu zwei DNS-Servern der PROFINET-Schnittstelle eines CP 1543-1 ändern. Außerdem können die NTP-Server beim CP konfiguriert werden. Die bisher gültigen Konfigurationsdaten werden dabei überschrieben.

Voraussetzung

- Um die Anweisung zu verwenden, müssen Sie bestimmte Eigenschaften der PROFINET-Schnittstelle einstellen. Öffnen Sie hierzu in der Gerätesicht die Eigenschaften der PROFINET-Schnittstelle. Im Dialog "Ethernet-Adressen" bzw. "Uhrzeitsynchronisation" aktivieren Sie die folgenden Optionen:
  - Zum Ändern der IP-Adressparameter über "T_CONFIG": Wählen Sie unter "IP-Protokoll" die Einstellung "Anpassen der IP-Adresse direkt am Gerät erlauben".
  - Zum Ändern des PROFINET-Gerätenamens über "T_CONFIG":
    Wählen Sie unter "PROFINET" die Einstellung "Anpassen des PROFINET-Gerätenamens direkt am Gerät erlauben".
    Wählen Sie zudem unter "IP-Protokoll" die Einstellung "Anpassen der IP-Adresse direkt am Gerät erlauben".
  - Zum Ändern der IP-Adressen der NTP-Server über "T_CONFIG":
    Wählen Sie unter "IP-Protokoll" die Einstellung "Anpassen der IP-Adresse direkt am Gerät erlauben".
    Wählen Sie "Uhrzeitsynchronisation über NTP-Server aktivieren" und geben Sie die IP-Adresse mindestens eines NTP-Servers vor.
- Die Konfigurationsdaten müssen in den folgenden Systemdatentypen hinterlegt und am Parameter CONF_DATA (Seite 3772) übergeben werden:
  - Die IP-Adresse, die Subnetz-Maske und die Router-Adresse hinterlegen Sie in dem Systemdatentyp IF_CONF_V4.
  - Den Gerätenamen hinterlegen Sie in dem Systemdatentyp IF_CONF_NOS. Beachten Sie die Einschränkungen, die für die Vergabe des Gerätenamens bestehen (siehe Parameter CONF_DATA (Seite 3772)).
  - Die IP-Adressen für die NTP-Uhrzeitsynchronisation hinterlegen Sie im Systemdatentyp IF_CONF_NTP.
  - Die IPv6-Adresse der PROFINET-Schnittstelle eines CP 1543-1 hinterlegen Sie im Systemdatentyp IF_CONF_IPV6.
  - Die MAC-Adresse der PROFINET-Schnittstelle eines CP 1543-1 hinterlegen Sie im Systemdatentyp IF_CONF_MAC.
  - Die IP-Adressen von bis zu zwei DNS-Servern für die PROFINET-Schnittstelle eines CP 1543-1 hinterlegen Sie im Systemdatentyp IF_CONF_DNS.

Arbeitsweise

Die Anweisung "T_CONFIG" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Konfigurationsvorgang, indem Sie "T_CONFIG" mit REQ = 1 aufrufen. Zu jedem Zeitpunkt kann nur ein Auftrag aktiv sein.

Der Baustein ist flankengetriggert, d. h. dass nach BUSY= FALSE der Baustein nochmals mit REQ=FALSE aufgerufen werden muss, damit die Instanz freigegeben wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_CONFIG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Beim Aufruf der Anweisung mit REQ = 1 wird die Bearbeitung der Anweisung gestartet.
INTERFACE	Input	HW_INTER-FACE	E, A, M, D, L oder Konstante	Hardware-Kennung der Schnittstelle Die Hardware-Kennung wird in den Eigenschaften der Schnittstelle in der Gerätesicht sowie in den Systemkonstanten der PLC-Variablen angezeigt.
CONF_DATA (Seite 3772)	Input	VARIANT	D, L	Zeiger auf die übergeordnete Struktur, welche die Systemdatentypen IF_CONF_HEADER, IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC und IF_CONF_DNS enthält (siehe Beschreibung des Parameters CONF_DATA).
DONE (Seite 3777)	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Bearbeitung noch nicht abgeschlossen • 1: Bearbeitung der Anweisung erfolgreich beendet.
BUSY (Seite 3777)	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen • 1: Bearbeitung der Anweisung läuft
ERROR (Seite 3777)	Output	BOOL	E, A, M, D, L	Zustandsparameter: • 0: kein Fehler • 1: Fehler
STATUS (Seite 3777)	Output	DWORD	E, A, M, D, L	Detaillierte Statusinformation: Am Parameter STATUS werden detaillierte Fehler- und Statusinformationen in Form eines Fehlercodes ausgegeben.
ERR_LOC (Seite 3777)	Output	DWORD	E, A, M, D, L	Fehlerort: • 0: Fehler bei Ausführung der Anweisung oder der Parametrierung. • > 0: Fehler bei Aufbau oder Inhalt der Konfigurationsdaten am Parameter CONF_DATA.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für T_CONFIG (Seite 3779).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter CONF_DATA

Aufbau der Konfigurationsdaten

Die Konfigurationsdaten am Parameter CONF_DATA können in einem globalen Datenbaustein oder im Abschnitt "Static" der Bausteinschnittstelle hinterlegt werden.

Die Konfigurationsdaten müssen entsprechend der folgenden Struktur hinterlegt werden:

Name		Datentyp	Beschreibung
ConfData		Struct	Übergeordnete Struktur, die am Parameter CONF_DATA übergeben wird.
	Header	IF_CONF_HEADER	Über den Header definieren Sie die Anzahl der folgenden Systemdatentypen. Der Systemdatentyp IF_CONF_HEADER muss immer enthalten sein.
	IPData	IF_CONF_V4	In diesem Systemdatentyp hinterlegen Sie die IP-Adresse, die Subnetz-Maske und die Router-Adresse.
	NoS	IF_CONF_NOS	In diesem Systemdatentyp hinterlegen Sie den PROFINET-Gerätenamen. Legen Sie IF_CONF_NOS nur an, wenn Sie über "T_CONFIG" auch den Gerätenamen ändern möchten.
	NTP	IF_CONF_NTP	In diesem Systemdatentyp hinterlegen Sie die IP-Adressen der NTP-Server für die Uhrzeitsynchronisation.
	IP_V6	IF_CONF_V6	In diesem Systemdatentyp hinterlegen Sie die IPv6-Adresse eines CP 1543-1.
	MAC	IF_CONF_MAC	In diesem Systemdatentyp hinterlegen Sie die MAC-Adresse eines CP 1543-1.
	DNS	IF_CONF_DNS	In diesem Systemdatentyp hinterlegen Sie die IP-Adressen von bis zu zwei DNS-Servern für einen CP 1543-1.

Die Systemdatentypen IF_CONF_HEADER, IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC und IF_CONF_DNS erstellen Sie, indem Sie in der Spalte "Datentyp" des Datenbausteins oder der Bausteinschnittstelle den Namen des Systemdatentyps eingeben. Den Namen für die Systemdatentypen können frei vergeben werden. Die Systemdatentypen können in beliebiger Reihenfolge kombiniert werden.

Systemdatentyp IF_CONF_Header

Über Systemdatentyp IF_CONF_Header geben Sie an, wie viele der Systemdatentypen IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC und IF_CONF_DNS bei der Ausführung von "T_CONFIG" verwendet werden.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	FieldType	UINT	0	Feldtyp: Muss immer den Wert "0" haben.
2 ... 3	FieldId	UINT	0	Feld-ID: Muss immer den Wert "0" haben.
4 ... 5	SubfieldCount	UINT	0	Anzahl der verwendeten Systemdatentypen IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC und IF_CONF_DNS

Systemdatentyp IF_CONF_V4

Über den Systemdatentyp IF_CONF_V4 definieren Sie die IP-Adresse, die Subnetz-Maske und die Router-Adresse.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	30	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	18	Länge des Systemdatentyps IF_CONF_V4 Da die Parameter von IF_CONF_V4 eine feste Länge und Struktur haben, muss für die Längenangabe der Startwert verwendet werden.
4 ... 5	Mode	UINT	0	Gültigkeit der Adressierung: • 1: Permanente Gültigkeit der Konfigurationsdaten • 2: Temporäre Gültigkeit der Konfigurationsdaten einschließlich Löschen vorhandener permanenter Konfigurationsdaten
6 ... 9	InterfaceAddress	IP_V4 *	0.0.0.0	IP-Adresse
10 ... 12	SubnetMask	IP_V4 *	0.0.0.0	Subnetzmaske
14 ... 16	DefaultRouter	IP_V4 *	0.0.0.0	Routeradresse
* Der Datentyp IP_V4 ist eine Struktur von 4 BYTE, welche die jeweilige Adresse des entsprechenden Parameters enthält (z.B. am Parameter SubnetMask die vierstellige Adresse der Subnetzmaske des IP-Protokolls).				

Systemdatentyp IF_CONF_NOS

Über den Systemdatentyp IF_CONF_NOS legen Sie den Stationsnamen fest, der bei der Ausführung der Anweisung "T_CONFIG" vergeben werden soll.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	40	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	246	Länge des Systemdatentyps IF_CONF_NOS in Byte. • Für eine absolute Längenangabe ergibt sich der Wert für dem Parameter Lenght aus: – 6 Byte für die Parameter Id, Length und Mode. – Bis zu 240 Byte für den Gerätenamen (Parameter NOS). Beispiel: Bei dem Gerätenamen "plc1" mit einer Länge von 4 Zeichen (=4 Byte) ergibt sich eine Gesamtlänge von 10. • Für eine dynamische Länge vernwenden Sie am Parameter Lenght den Default-Startwert 246. Achten Sie darauf, nach dem Namen den Wert "0" einzutragen (siehe Beschreibung für Parameter NOS).

Byte	Parameter	Datentyp	Startwert	Beschreibung
4 ... 5	Mode	UINT	0	Gültigkeit der Änderung des Gerätenamens: • 1: Permanente Gültigkeit des Gerätenamens. • 2: Temporäre Gültigkeit des Gerätenamens.
6 ... 244	NOS	ARRAY [1...240] of Byte	0	Gerätename (Name of Station) • Sie müssen das ARRAY ab dem ersten Byte belegen. Wenn Sie das erste Byte mit "0" belegen, wird der Stationsname gelöscht. • Die Mindestlänge für den Namen beträgt 1 Byte. Die maximale Länge für den Namen beträgt 240 Byte. • Ist der Gerätename kürzer als am Parameter Length angegeben, müssen Sie nach dem eigentlichen Stationsnamen ein Nullbyte (16#0 hex) eintragen (konform zu IEC 61185-6-10). Andernfalls wird NOS zurückgewiesen, und die Anweisung "T_CONFIG" gibt den Fehlercode DW#16#C0809400 am Parameter STATUS aus. • Ist der Gerätename länger als am Parameter Length angegeben, wird der Gerätename nur bis zur vorgegeben Länge geschrieben. Für den Gerätenamen gelten folgende Einschränkungen: • Die Namensangabe muss im ASCII-Code erfolgen. • Für den Namen dürfen nur Kleinbuchstaben, Ziffern, Bindestriche oder Punkte verwendet werden. – Der Name darf nicht mit einem Bindestrich beginnen oder enden. – Der Name darf nicht die Form n.n.n.n haben (n = 0, ... 999). – Der Name darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9). • Ein Namensbestandteil zwischen zwei Punkten darf maximal 63 Zeichen lang sein. • Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Wird ein ungültiges Zeichen verwendet, wird am Parameter STATUS der Fehlercode C080_9400 ausgegeben.

Systemdatentyp IF_CONF_NTP

Über den Systemdatentyp IF_CONF_NTP legen Sie die IP-Adressen der NTP-Server für die Uhrzeitsynchronisation fest.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	17	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	22	Länge des Systemdatentyps IF_CONF_NTP in Byte Da die Parameter von IF_CONF_NTP eine feste Länge und Struktur haben, muss für die Längenangabe der Startwert verwendet werden.
4 ... 5	Mode	UINT	2	Gültigkeit: • 1: Permanente Gültigkeit (nicht zulässig) • 2: Temporäre Gültigkeit
6 ... 9	NTP_IP[1]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 1
...	...			IP-Adresse von NTP-Server 2 ... 3
18 ... 21	NTP_IP[4]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 4

Systemdatentyp IF_CONF_V6 (nur bei Verwendung eines CP 1543-1)

Über den Systemdatentyp IF_CONF_V6 definieren Sie eine IPv6-Adresse für die mittels INTERFACE ausgewählte Schnittstelle.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	22	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	22	Länge des Systemdatentyps IF_CONF_V6 in Byte Da die Parameter von IF_CONF_V6 eine feste Länge und Struktur haben, muss für die Längenangabe der Startwert verwendet werden.
4 ... 5	Mode	UINT	0	Gültigkeit der IPv6-Adresse: • 1: Permanente Gültigkeit • 2: Temporäre Gültigkeit
6 ... 21	InterfaceAddress	IP_V6 *		IPv6-Adresse
* Der Datentyp IP_V6 ist eine Struktur von 16 Bytes.				

Systemdatentyp IF_CONF_MAC (nur bei Verwendung eines CP 1543-1)

Über den Systemdatentyp IF_CONF_MAC geben Sie die MAC-Adresse für die mittels INTERFACE ausgewählte Schnittstelle vor.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	3	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	12	Länge des Systemdatentyps IF_CONF_MAC in Byte Da die Parameter von IF_CONF_MAC eine feste Länge und Struktur haben, muss für die Längenangabe der Startwert verwendet werden.
4 ... 5	Mode	UINT	0	Gültigkeit der MAC-Adresse: • 1: Permanente Gültigkeit • 2: Temporäre Gültigkeit
6 ... 11	Mac	ARRAY [1...6] of Byte		MAC-Adresse Beachten Sie, dass durch das Ändern der MAC-Adresse Verbindungen abgebrochen werden können.

Systemdatentyp IF_CONF_DNS (nur bei Verwendung eines CP 1543-1)

Über den Systemdatentyp IF_CONF_DNS geben Sie die IP-Adressen von bis zu zwei DNS-Servern für die mittels INTERFACE ausgewählte Schnittstelle fest.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 ... 1	Id	UINT	16	Kennung des Systemdatentyps. Der Startwert dieses Parameters darf nicht verändert werden.
2 ... 3	Length	UINT	14	Länge des Systemdatentyps IF_CONF_DNS in Byte Da die Parameter von IF_CONF_DNS eine feste Länge und Struktur haben, muss für die Längenangabe der Startwert verwendet werden.
4 ... 5	Mode	UINT	0	Gültigkeit: • 1: Permanente Gültigkeit • 2: Temporäre Gültigkeit
6 ... 9	DNS_IP1	IP_V4		IP-Adresse von DNS-Server 1
10 ... 13	DNS_IP2	IP_V4		IP-Adresse von DNS-Server 2 Wenn nur eine Adresse vorgegeben werden soll, ist die IP-Adresse von DNS-Server 2 mit 0.0.0.0 zu belegen.

Parameter DONE, BUSY und ERROR

Beschreibung

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung aktuell befindet bzw. wann die Übertragung der Konfigurationsdaten beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	FALSE	FALSE	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS (Seite 3777) entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Parameter STATUS und ERR_LOC

Beschreibung

An den Parametern STATUS und ERR_LOC werden die Status- und Fehlermeldungen der Anweisung "T_CONFIG" ausgegeben:

- Am Parameter STATUS wird die Fehlerursache ausgegeben.
- Am Parameter ERR_LOC wird ausgegeben, wo der Fehler aufgetreten ist. Hierfür bestehen folgende Möglichkeiten:
  - 16#0000_0000: Fehler bei Aufruf der Anweisung (z. B. Fehler bei Parametrierung der Anweisung oder bei Kommunikation mit der PROFINET-Schnittstelle).
  - 16#0001_0000: Fehler bei Konfigurationsdaten in den Parametern des Systemdatentyps IF_CONF_HEADER.
  - 16#0001_000x: Fehler bei Konfigurationsdaten in den Parametern des Systemdatentyps IF_CONF_V4 oder IF_CONF_NOS oder IF_CONF_NTP oder IF_CONF_V6 oder IF_CONF_MAC oder IF_CONF_DNS (x gibt die Position des fehlerhaften Teilblocks in der T_CONFIG-Struktur an. Wenn die T_CONFIG-Struktur z. B. einen Teilblock für die IP-Adresse und einen Teilblock für den Stationsnamen enthält und der Fehler im Teilblock für den Stationsnamen liegt, hat ERR_LOC den Wert 0001_0002.)

Die folgende Tabelle zeigt die möglichen Werte für die Parameter STATUS und ERR_LOC:

STATUS*	ERR_LOC*	Erläuterung
0000_0000	0000_0000	Auftragsbearbeitung ohne Fehler beendet.
0070_0000	0000_0000	Keine Auftragsbearbeitung aktiv.
0070_0100	0000_0000	Start der Auftragsbearbeitung.
0070_0200	0000_0000	Zwischenaufruf (REQ irrelevant).
C08x_yy00	0000_0000	Allgemeine Fehlerinformation. Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

STATUS*	ERR_LOC*	Erläuterung
C080_8000	0000_0000	Fehler bei Aufruf der Anweisung: Die Hardware-Kennung am Parameter Interface ist ungültig.
C080_8100	0000_0000	Fehler bei Aufruf der Anweisung: Die Hardware-Kennung am Parameter Interface adressiert keine PROFINET-Schnittstelle.
C080_8700	0000_0000	Fehler bei Aufruf der Anweisung: Falsche Länge des Datenbausteins am Parameter CONF_DATA.
C080_8800	0001_0000	Fehler im Systemdatentyp IF_CONF_HEADER: Der Parameter FieldType hat einen unzulässigen Wert. Verwenden Sie für FieldType den Wert "0".
C080_8900	0001_0000	Fehler im Systemdatentyp IF_CONF_HEADER: Der Parameter FieldId hat einen unzulässigen Wert. Verwenden Sie für FieldId den Wert "0".
C080_8A00	0001_0000	Fehler im Systemdatentyp IF_CONF_HEADER: Falsche Anzahl am Parameter SubfieldCount. Geben Sie die korrekte Anzahl der verwendeten Systemdatentypen IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS an.
C080_8B00	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Der Parameter Id hat einen unzulässigen Wert. Verwenden Sie bei IF_CONF_V4 "30", bei IF_CONF_NOS "40", bei IF_CONF_NTP "17", bei IF_CONF_V6 "22", bei IF_CONF_MAC "3", bei IF_CONF_DNS "16". Hinweis: IF_CONF_NTP ist nur zulässig, wenn T_CONFIG auf die erste Schnittstelle [X1] einer S7-1500-Modular-CPU angewendet wird.
C080_8C00	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Falscher Systemdatentyp verwendet, falsche Reihenfolge oder mehrfache Verwendung eines Systemdatentyps.
C080_8D00	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Der Parameter Length hat einen falschen oder unzulässigen Wert.
C080_8E00	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Der Parameter Mode hat einen falschen oder unzulässigen Wert. • Bei IF_CONF_V4 und IF_CONF_NOS sind nur die Werte "1" (permanent) oder "2" (temporär) zulässig. • Bei IF_CONF_NTP ist nur der Wert "2" (temporär) zulässig.
C080_9000	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Konfigurationsdaten können nicht übernommen werden. Mögliche Ursache: • Bei IF_CONF_V4: In der Hardware-Konfiguration wurde die Einstellung "IP-Adresse am Gerät einstellen" nicht ausgewählt. • Bei IF_CONF_NOS: In der Hardware-Konfiguration wurde die Einstellung "PROFINET-Gerätenamen am Gerät einstellen" nicht ausgewählt. • Bei IF_CONF_NTP: In der Hardware-Konfiguration wurde die Einstellung "Uhrzeitsynchronisation über NTP-Server aktivieren" nicht ausgewählt und keine IP-Adresse für NTP-Server vorgegeben.

STATUS*	ERR_LOC*	Erläuterung
C080_9400	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Ein Parameterwert ist nicht definiert oder unzulässig.
C080_9500	0001_000x	Fehler im Systemdatentyp IF_CONF_V4, IF_CONF_NOS, IF_CONF_NTP, IF_CONF_V6, IF_CONF_MAC oder IF_CONF_DNS: Die Werte von zwei Parametern sind inkonsistent.
C080_C200	0000_0000	Fehler bei Aufruf der Anweisung: Die Übertragung der Konfigurationsdaten kann nicht durchgeführt werden. Mögliche Ursache: Die PROFINET-Schnittstelle ist nicht erreichbar.
C080_C300	0000_0000	Fehler bei Aufruf der Anweisung: Ungenügende Ressourcen (z. B. mehrfacher Aufruf von "T_CONFIG" mit unterschiedlichen Parametern).
C080_C400	0000_0000	Fehler bei Aufruf der Anweisung: Temporärer Kommunikationsfehler. Versuchen Sie, die Anweisung zu einem späteren Zeitpunkt erneut aufzurufen.
C080_D200	0000_0000	Fehler bei Aufruf der Anweisung: Aufruf nicht möglich. Anweisung wird durch die ausgewählte PROFINET-Schnittstelle nicht unterstützt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Programmbeispiel für T_CONFIG

Einleitung

In dem folgenden Beispiel nutzen Sie eine konfigurierte Verbindung (z. B. per UDP) zwischen zwei CPUs. Mit Hilfe der Anweisung "T_CONFIG" konfigurieren Sie die IP-Adresse und den PROFINET Gerätenamen der CPU 1.

Voraussetzungen

Verbindung setzen:

- Zwei CPUs (z. B. S7-1513-1 PN) sind angelegt und per PROFINET miteinander verbunden.
- Eine konfigurierte Verbindung ist nicht notwendig.

PROFINET der CPU 1 einrichten:

1. Öffnen Sie die Gerätesicht der CPU 1 "> Eigenschaften > PROFINET-Schnittstelle > Ethernet-Adressen > PROFINET".
2. Um den Gerätenamen einzustellen, wählen Sie die folgenden Optionen:
   – "Anpassen der IP-Adresse direkt am Gerät erlauben".
   – "Anpassen des PROFINET-Gerätenamens direkt am Gerät erlauben".

Den Wert von "hwid" anpassen:

- Ändern Sie den Wert von "hwid" entsprechend der Hardware-Kennung der PROFINET-Schnittstelle Ihres lokalen Gerätes (CPU 1).

Hinweis

Öffnen Sie "PLC-Variablen > Alle Variablen anzeigen > Systemkonstanten". Suchen Sie den Eintrag "<Local~PROFINET_interface_1>", mit dem Datentyp "Hw_Interface". Die Zelle "Wert" enthält die Hardware-Kennung.

Variablen anlegen und Parameter verschalten (Programm der CPU 1)

Zur Ablage der Daten legen Sie einen globalen Datenbaustein ("SLI_gDB_T_CONFIG") mit den folgenden Strukturen und Variablen an.

SLI_gDB_T_CONFIG

	Name	Data type	Start value
1	Static		
2	T_CONFIG	Struct	
3	start	Bool	false
4	hwid	HW_INTERFACE	64
5	busy	Bool	false
6	done	Bool	false
7	error	Bool	false
8	errorLocation	DWord	16#0
9	status	DWord	16#0
10	memErrStat	DWord	16#0
11	configData	Struct	
12	header	IF_CONF_Header	
13	deviceIP	IF_CONF_v4	
14	deviceName	IF_CONF_NOS	

In der Struktur "configData": Den Systemdatentyp "IF_CONF_Header" zur Angabe der Größe der PROFINET-Daten parametrieren Sie wie folgt:

12	header	IF_CONF_Header	
13	FieldType	UInt	0
14	FieldId	UInt	0
15	SubfieldCount	UInt	2

In der Struktur "configData": Den Systemdatentyp "IF_CONF_v4" zur Definition der IP-Adresse parametrieren Sie wie folgt:

16	deviceIP	IF_CONF_v4	
17	Id	UInt	30
18	Length	UInt	18
19	Mode	UInt	1
20	InterfaceAddress	IP_V4	
21	ADDR	Array[1..4] of Byte	
22	ADDR[1]	Byte	192
23	ADDR[2]	Byte	168
24	ADDR[3]	Byte	0
25	ADDR[4]	Byte	3
26	SubnetMask	IP_V4	
27	ADDR	Array[1..4] of Byte	
28	ADDR[1]	Byte	255
29	ADDR[2]	Byte	255
30	ADDR[3]	Byte	255
31	ADDR[4]	Byte	0
32	DefaultRouter	IP_V4	
33	ADDR	Array[1..4] of Byte	
34	ADDR[1]	Byte	16#0
35	ADDR[2]	Byte	16#0
36	ADDR[3]	Byte	16#0
37	ADDR[4]	Byte	16#0

In der Struktur "configData": Den Systemdatentyp "IF_CONF_NOS" zur Definition des PROFINET-Gerätenamens parametrieren Sie wie folgt:

38	deviceName	IF_CONF_NOS	
39	Id	UInt	40
40	Length	UInt	11
41	Mode	UInt	1
42	NOS	Array[1..240] of Byte	
43	NOS[1]	Byte	'm'
44	NOS[2]	Byte	'y'
45	NOS[3]	Byte	'p'
46	NOS[4]	Byte	'l'
47	NOS[5]	Byte	'c'

Hinweis

Aufbau von "configData"

Die Informationen der Struktur "configData" lassen sich wie folgt interpretieren:

- Im Header (IF_CONF_Header):
  SubfieldCount = 2. Der Wert bedeutet: Im Folgenden werden zwei weitere Strukturen ("deviceIP", "deviceName") (*) verwendet.
  *Achten Sie auf die Einhaltung der Reihenfolge der beiden Strukturen.
- Z. B. in der Struktur "deviceName" (Systemdatentyp IF_CONF_NOS):
  - Lenght = 11 (*). Der Wert enspricht der Gesamtlänge der Struktur NOS (5 Byte für den Gerätenamen "myplc" + 6 Byte für die Parameter Id, Length und Mode).
    *Anstelle einer absoluten Längenangabe, können Sie für eine dynamische Länge auch den Default-Startwert (Lenght = 0) verwenden.
  - Mode = 1. Der Wert bewirkt eine permanente Änderung des Gerätenamens in "myplc".
  - NOS[1] ... NOS[5]. Das Array NOS enthält den neuen Gerätenamen (1 Zeichen / Byte).

Sie erstellen den FB "SLI_FB_T_CONFIG". Darin legen Sie die folgenden lokalen Variablen an.

7	Static			
8	T_CONFIG_Instance	T_CONFIG		
9	done	Bool	false	Non-retain
10	errorLocation	DWord	16#0	Non-retain

Netzwerk 1: Die Parameter der Anweisung "T_CONFIG" verschalten Sie wie folgt.

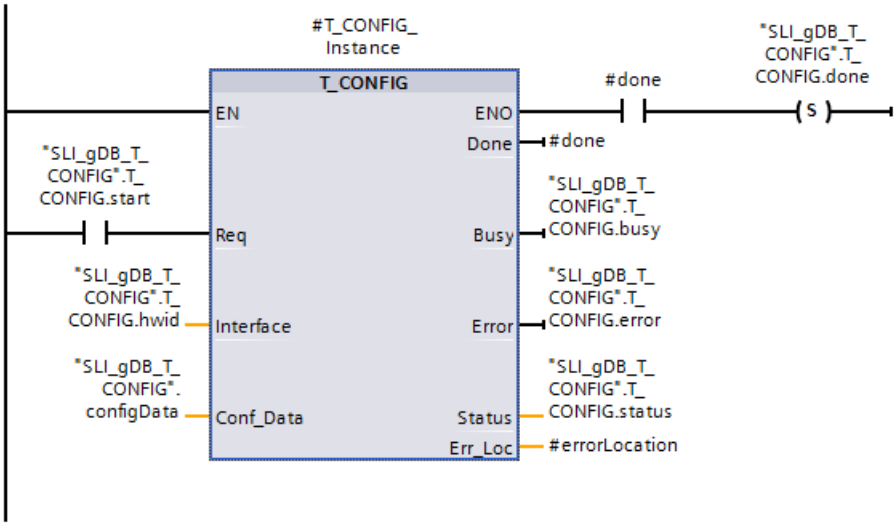

Netzwerk 2: Den Status im Fehlerfall von T_CONFIG speichern Sie wie folgt.

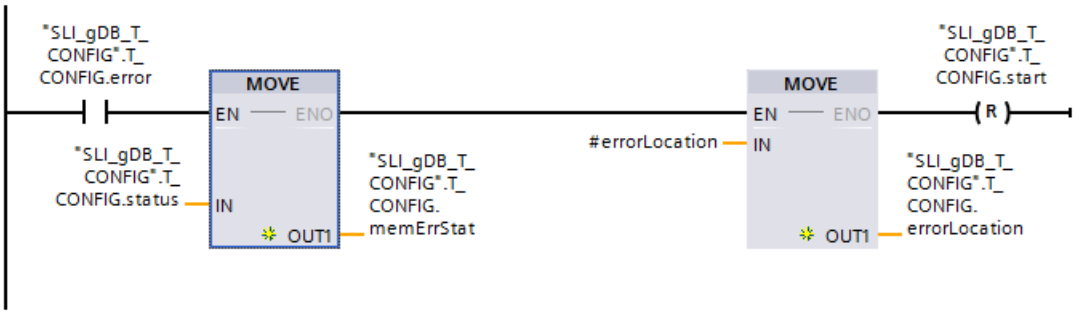

Netzwerk 3: Zum Beenden von T_CONFIG legen Sie die folgenden Verschaltungen an.

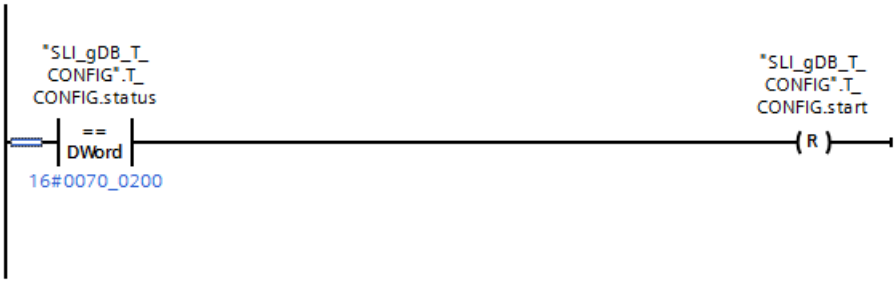

Ergebnis

Netzwerk 1:

Wenn der Eingangsparameter REQ ("T_CONFIG.start") den Signalzustand "TRUE" liefert, wird die Anweisung "T_CONFIG" gestartet. Über mehrere Aufrufe hinweg konfiguriert die Anweisung "T_CONFIG" die integrierte PROFINET-Schnittstelle der CPU. Entsprechend dem Eingangsparameter INTERFACE ("T_CONFIG.hwid") ist der Anweisung "T_CONFIG" die zu verwendende Schnittstelle bekannt.

Über den Ausgangsparameter DONE ("#done") mit dem Signalzustand "TRUE" wird die erfolgreiche Übermittelung des Datensatzes angezeigt. Zugleich wird am Ausgangsparameter STATUS ("T_CONFIG.status") der Wert "0000_0000" angezeigt. Da die Werte der Ausgangsparameter nur für den Moment ihrer Gültigkeit angezeigt werden, speichern Sie die Erfolgsmeldung in der Variable "T_CONFIG.done".

Am Ausgangsparameter ERROR ("T_CONFIG.error") bzw. der Variable "T_CONFIG.memErrStat" wird angezeigt, dass die Verarbeitung im Beispiel ohne Fehler verläuft.

SLI_gDB_T_CONFIG				
	Name	Data type	Start value	Monitor value
1	Static			
2	T_CONFIG	Struct		
3	start	Bool	false	FALSE
4	hwid	HW_INTERFACE	64	16#0040
5	busy	Bool	false	FALSE
6	done	Bool	false	TRUE
7	error	Bool	false	FALSE
8	errorLocation	DWord	16#0	16#0000_0000
9	status	DWord	16#0	16#0070_0000
10	memErrStat	DWord	16#0	16#0000_0000
11	configData	Struct		

Netzwerk 2:

Wenn T_CONFIG einen Fehler meldet ("T_CONFIG.error" ist "TRUE"), speichern Sie die Meldung wie folgt:

- Sie speichern den Status ("T_CONFIG.status") in der Variable "T_CONFIG.memErrStat".
- Sie speichern den Fehlerort, ausgegeben am Ausgangsparameter ERR_LOC ("#errorLocation") von T_CONFIG, in der Variable "T_CONFIG.errorLocation".

Online & Diagnose

Zum Überprüfen, ob die PROFINET-Daten geändert wurden, öffnen Sie im Projektbaum der CPU den Pfad "Online & Diagnose > Funktionen > Name zuweisen".

Accessible devices in the network:			
IP address	MAC address	Device	PROFINET device name
192.168.0.2	28-63-36-8D-BB-49	S7-1500	simaticplc
192.168.0.3	00-1B-1B-22-63-D0	S7-1500	myplc

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Beschreibung T_CONFIG (Seite 3768)

Änderungen der Kommunikationsanweisungen

Unterschiede zwischen den Versionen <= V3.x und >= V4.1 der OUC-Bibliothek

Einleitung

Die CPU S7-1200 unterstützt ab der Firmware-Version 4.1 neue Anweisungsversionen zur Open User Communication (OUC). Die neuen Anweisungen sind in der Bibliothek mit der Version 4.1 enthalten.

Wenn Sie diese Bibliothek nutzen möchten, müssen an dem Anwenderprogramm Änderungen vorgenommen werden, da sich die neuen Anweisungen zur Open User Communication zum Teil anders verhalten.

In diesem Kapitel werden die Unterschiede, insbesondere im Aufrufverhalten der Anweisungen detailliert beschrieben.

Hinweis

Beschreibung nur relevant beim Umstieg von der CPU S7-1200 ≤ V4.0 auf S7-1200 ≥ V4.1

Wenn Sie eine CPU S7-1500 CPU verwenden, sind Änderungen zwischen den Bibliotheksversionen nicht relevant. Das Gleiche gilt, wenn Sie eine S7-1200 ≥ Version 4.1 einsetzen, und die Bibliothek zur Open User Communication nicht auf Version <4.1 umstellen.

Unterschiede zwischen den Versionen <= V3.x und >= V4.1 der Bibliothek Open User Communication

Die folgende Tabelle zeigt, bei welchen Anweisungen der Bibliothek Open User Communication Unterschiede zwischen den Versionen <= V3.x und >= V4.1 bestehen. Klicken Sie für Detailinformationen auf den Namen der Anweisung.

Anweisung	Version in Bibliothek <= V3.x (CPU-FW ≤ V4.0)	Version in Bibliothek >= V4.1 (CPU-FW ≥ V4.1)
TSEND_C (Seite 3786)	V2.1	V3.0
TRCV_C (Seite 3787)	V2.1	V3.0
TMAIL_C *	V2.1	V3.0
TCON (Seite 3789)	V3.0	V4.0
TDISCON	V2.1	V2.1 (identisch zu Bibliothek <= V3.x)
TSEND (Seite 3791)	V3.0	V4.0
TRCV (Seite 3793)	V3.0	V4.0
TUSEND (Seite 3791)	V3.0	V4.0
TURCV (Seite 3793)	V3.0	V4.0
T_RESET *	V1.1	V1.2
T_DIAG *	V1.1	V1.2
T_CONFIG	V1.0	V1.0 (identisch zu Bibliothek <= V3.x)
* Es bestehen keine Unterschiede in den Versionen, die sich auf das Anwenderprogramm auswirken.		

Hinweis

Wechsel der Open-User-Communication-Bibliothek von einer Version <= V3.x zu einer Version >= V4.1

Beim Austausch der Open-User-Communication-Bibliothek von einer Version <= V3.x zu einer Version >= V4.1 müssen Sie auch die Modbus-TCP-Bibliothek austauschen. Überprüfen Sie danach alle für Ihr Programm relevanten Anweisungen.

Siehe auch

Unterschiede zwischen den Versionen <= V3.x und >= V4.0 der Modbus-TCP-Bibliothek (Seite 4320)

Änderungen bei der Anweisung TSEND_C

Aufrufverhalten von TSEND_C (V<3.0)

Bis zur Version 2.1 der Anweisung TSEND_C wird Ausgangsparameter DONE zweimal gesetzt: Einmal nach dem Verbindungsaufbau durch die intern verwendete Anweisung TCON und danach nach Abschluss eines Sendevorgangs durch die intern verwendete Anweisung TSEND.

Die folgende Grafik zeigt den Verbindungsaufbau und das Senden von Daten mit TSEND_C V2.1:

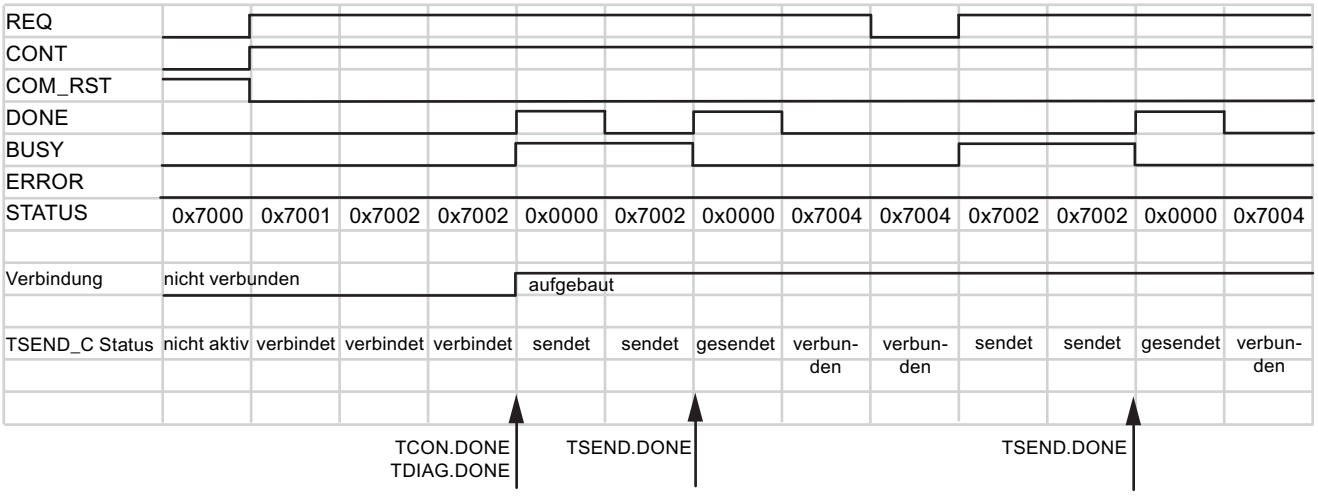

Beim Setzen des Parameters COM_RST auf "1" kann der laufende Verbindungsaufbau oder eine laufende Datenübertragung jederzeit zurücksetzen werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut.

Aufrufverhalten von TSEND_C (V≥3.0)

Ab der Version TSEND_C wird der Parameter DONE nur gesetzt, wenn ein Datentransfer durch die intern verwendete Anweisung TSEND beendet wurde (STATUS = 0000).

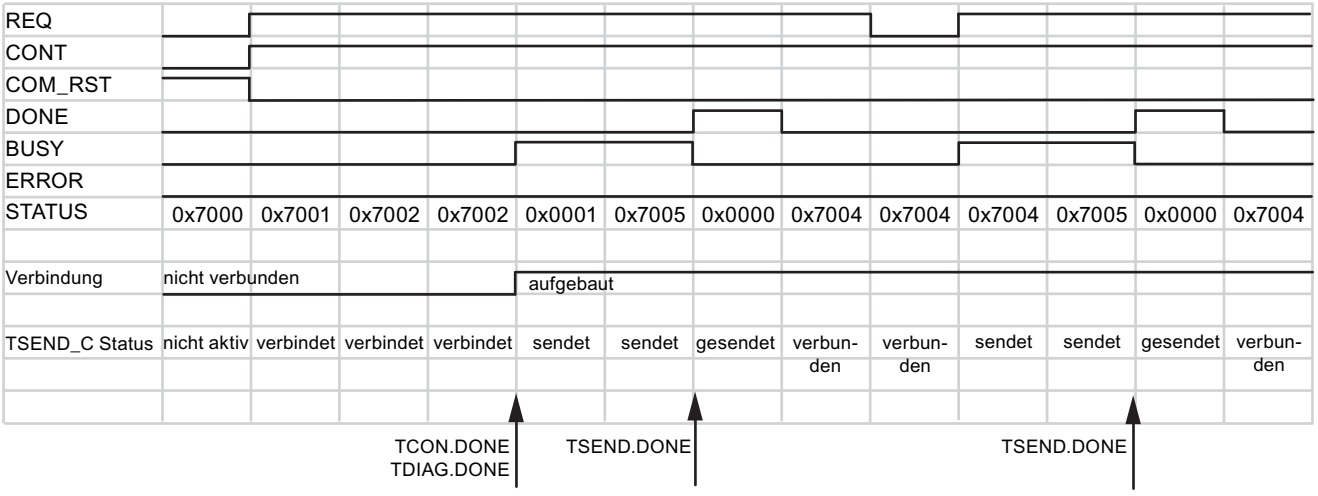

Beim Setzen des Parameters COM_RST auf "1" wird die bestehende Verbindung kurzfristig unterbrochen und zurückgesetzt. Im Gegensatz zu TSEND_C V2.1 bleibt der Verbindungsendpunkt jedoch bestehen.

Hinweis

Zusätzliche Protokolle bei TSEND_C ab Version 3.0

In der Version 3.0 unterstützt die Anweisung TSEND_C auch UDP und UDP Broadcast über das Interface der CPU und über CM/CPs.

Siehe auch

TSEND_C: Daten über Ethernet senden (Seite 3626)

TSEND_C: Verbindung aufbauen und Daten senden (Seite 3631)

Änderungen bei der Anweisung TRCV_C

Aufrufverhalten von TRCV_C (V<3.0)

Bis zur Version 2.1 der Anweisung TRCV_C wird Ausgangsparameter DONE gesetzt, nachdem die Verbindung aufgebaut ist. Bei dem Ausgangsparameter STATUS wird nicht unterschieden, ob der Verbindungsaufbau oder der Datentransfer abgeschlossen wurde.

Die folgende Grafik zeigt den Verbindungsaufbau und das Senden von Daten mit TRCV_C V2.1:

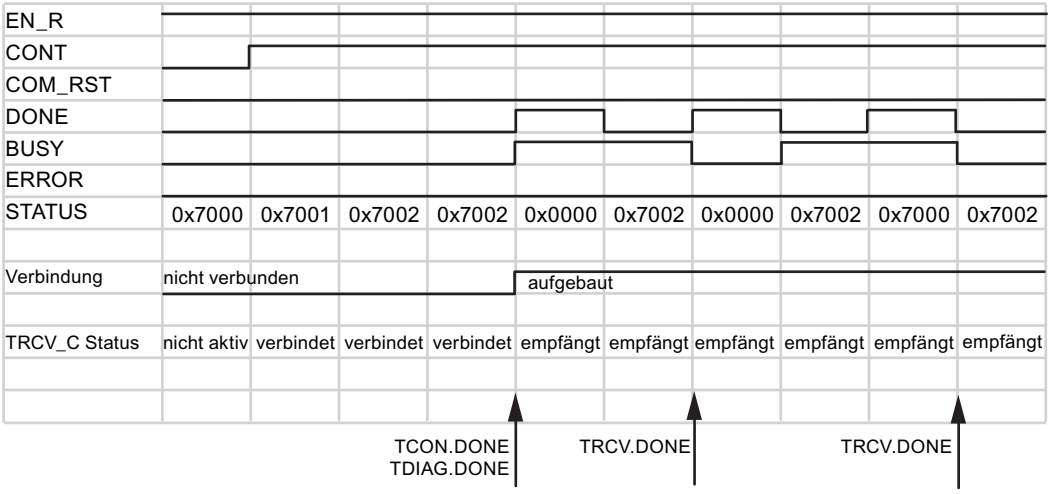

Beim Setzen des Parameters COM_RST auf "1" kann der laufende Verbindungsaufbau oder eine laufende Datenübertragung jederzeit zurückgesetzt werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut.

Aufrufverhalten von TRCV_C (V≥3.0)

Ab der Version TRCV_C wird der Parameter DONE nur gesetzt, wenn ein Datentransfer durch die intern verwendete Anweisung TRCV beendet wurde (STATUS = 0000). Der Abschluss des Verbindungsaufbaus durch die intern verwendete Anweisung TCON wird am Parameter STATUS durch den Ausgabewert 0x0001 angezeigt.

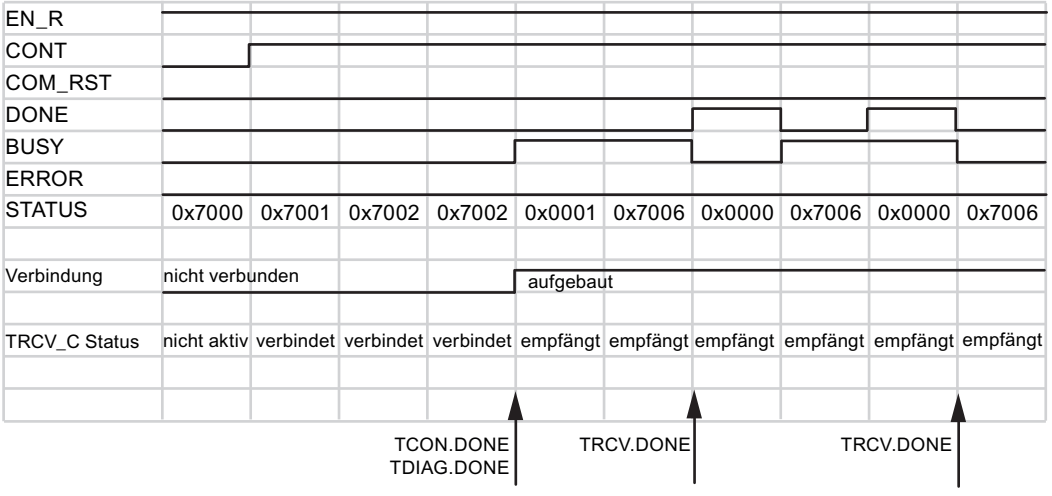

Beim Setzen des Parameters COM_RST auf "1" wird die bestehende Verbindung kurzfristig unterbrochen und zurückgesetzt. Im Gegensatz zu TSEND_C V2.1 bleibt der Verbindungsendpunkt jedoch bestehen.

Hinweis

Zusätzliche Protokolle bei TRCV_C ab Version 3.0

In der Version 3.0 unterstützt die Anweisung TRCV_C auch UDP und UDP Broadcast über das Interface der CPU und über CM/CPs.

Hinweis

ADHOC-Modus bei der Protokollvariante TCP

Entsprechend der Anweisung TRCV wird bei der TRCV_C bis zur Version 2.1 der ADHOC-Modus aktiviert, indem Sie dem Parameter LEN den Wert "0" zuweisen. Ab der Version 3.0 der Anweisung verwenden Sie hierzu den Parameter ADHOC. Detaillierte Informationen hierzu finden Sie in den Beschreibungen der Anweisungen.

Siehe auch

Änderungen bei den Anweisungen TRCV / TURCV (Seite 3793)

TRCV_C: Daten über Ethernet empfangen (Seite 3640)

TRCV_C: Verbindung aufbauen und Daten empfangen (Seite 3646)

Änderungen bei der Anweisung TCON

Geändertes Aufrufverhalten bei Verbindungsfehler

Bisheriges Aufrufverhalten von TCON (V<4.0)

- Die bisherige Anweisung TCON (V<4.0) wird mit einer steigenden Flanke am Eingangsparameter REQ aktiviert.
- Wenn der entfernte Kommunikationspartner nicht erreichbar ist, setzt die Anweisung den Ausgangsparameter BUSY.
- Eine Fehlermeldung wird nicht ausgegeben.

Die folgende Grafik zeigt das Verhalten von TCON aufseiten des aktiven Verbindungspartners. Kommt keine Verbindung zu Stande, wird die Anweisung nicht erneut aufgerufen.

Aufruf	1	2	3	4	5	6
REQ						
DONE						
BUSY						
ERROR						
STATUS	0x7000	0x7001	0x7002	0x7002	0x70020	0x7002
entfernter Partner		nicht verfügbar				
verbunden						
TCON Status	nicht aktiv	verbindet	verbindet	verbindet	verbindet	verbindet

Neues Aufrufverhalten von TCON (V≥4.0)

- Die neue Anweisung TCON wird ebenfalls einer steigenden Flanke am Eingangsparameter REQ aktiviert.
- Anders als bisher wird eine Fehlermeldung ausgegeben, wenn der entfernte Kommunikationspartner nicht erreichbar ist. Der Fehler kann abgefragt und der Aufruf durch eine erneute Flanke am Parameter REQ erneut gestartet werden.
- Für das Anwenderprogramm ergeben sich folgende Änderungen:
  - Zusätzliche Fehlermeldungen von TCON, die ausgewertet werden können (siehe Beschreibung TCON (Seite 3692)).
  - Der Verbindungsaufbau kann über eine erneute steigende Flanke reinitialisiert werden. Werden Sie hierzu auch die Parameter DONE und ERROR aus, um sicherzugehen, dass ein Fehler vorliegt.

Die folgende Grafik zeigt das Verhalten von TCON aufseiten des aktiven Verbindungspartners. Nach einem Netzwerkfehler (Fehlercode 80C6) wird die Anweisung erneut aufgerufen.

Aufruf	1	2	3	4	5	6	7	8	9
REQ									
DONE									
BUSY									
ERROR									
STATUS	0x7000	0x7001	0x7002	0x7002	0x80C6	0x7000	0x7001	0x7002	0x7002
entfernter Partner			nicht verfügbar						
verbunden									
TCON Status	nicht aktiv	verbindet	verbindet	verbindet	Fehler	nicht aktiv	verbindet	verbindet	verbindet

Aufrufverhalten ohne Verbindungsfehler

Liegt kein Verbindungsfehler vor, verhalten sich die Versionen von TCON gleich.

Die folgende Grafik zeigt das Verhalten von TCON aufseiten des aktiven Verbindungspartners.

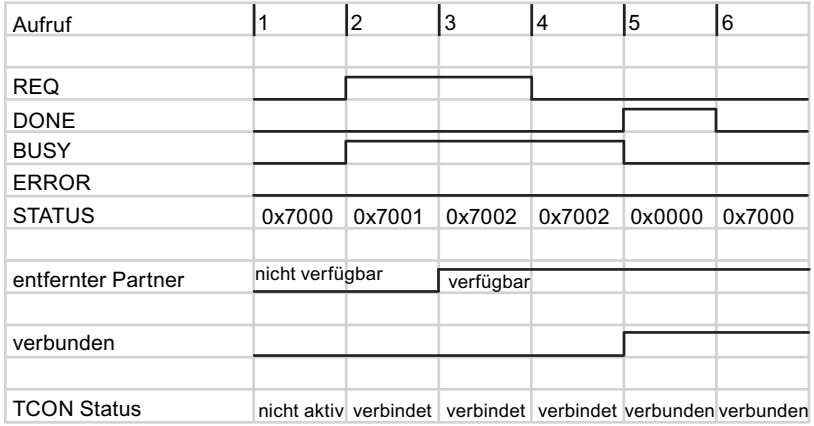

Siehe auch

TCON: Kommunikationsverbindung aufbauen (Seite 3689)

Änderungen bei den Anweisungen TSEND / TUSEND

Sendevorgang bei bestehender Verbindung

Tritt bei dem Sendevorgang keine Verzögerung oder Unterbrechung auf, verhalten sich die Version 3.0 und 4.0 der Anweisungen TSEND / TUSEND gleich:

- Die Anweisung wird mit einer positiven Flanke am Parameter REQ aufgerufen. Die Anweisung wird asynchron abgearbeitet, d. h. sie muss so lange aufgerufen werden, bis durch den Parameter DONE die vollständige Ausführung angezeigt wird.
- Maximal können bei TSEND 8 kByte Daten übertragen werden. Bei TUSEND maximal 1472 Bytes (ab Firmware-Stand V2.5 der S7-1500-CPUs bei Unicast oder Multicast: 2048 Bytes). Die Angaben beziehen sich jeweils auf die vollständige Abarbeitung der Anweisung, d. h. alle nötigen Aufrufe, bis der Parameter DONE gesetzt wird.
- Die Datenübertragung erfolgt in drei Schritten:
  - Die Daten werden aus dem Operandenbereich in einen internen Puffer geschrieben (wird angezeigt durch STATUS=7001).
  - Danach erfolgt die Übertragung an den entfernten Kommunikationspartner.
  - War die Übertragung erfolgreich, wird DONE auf "1" und STATUS zurück auf "0" gesetzt wird (siehe Aufruf 5 in folgender Grafik).

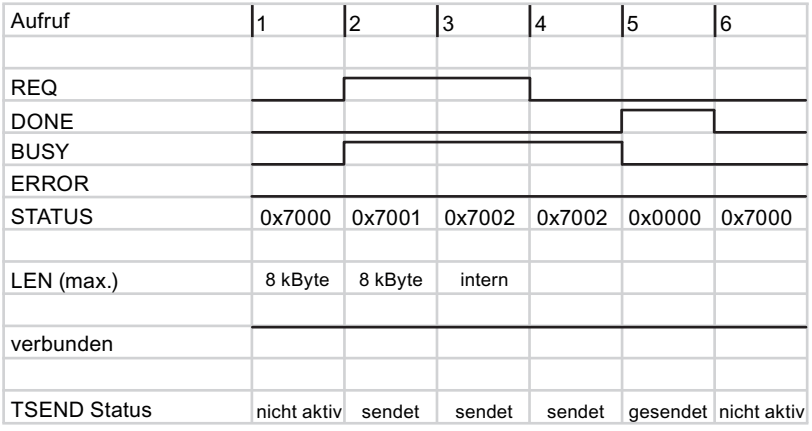

Sendevorgang bei zeitverzögertem Aufruf

Erfolgt der Sendevorgang nicht über ein Kommunikationsmodul (CM) oder einen Kommunikationsprozessor (CP) verhalten sich die Versionen 3.0 und 4.0 der Anweisungen TSEND / TUSEND gleich:

- In dem folgenden Beispiel wird TSEND oder TUSEND nur einmal mit einer positiven Flanke am Parameter REQ aufgerufen (siehe Aufruf 2 in folgender Grafik).
- Ist die Zeitverzögerung ausreichend lang und konnten die Daten übertragen werden, wird im nächsten Aufruf (hier Aufruf 3) der Parameter DONE direkt auf "1" gesetzt.

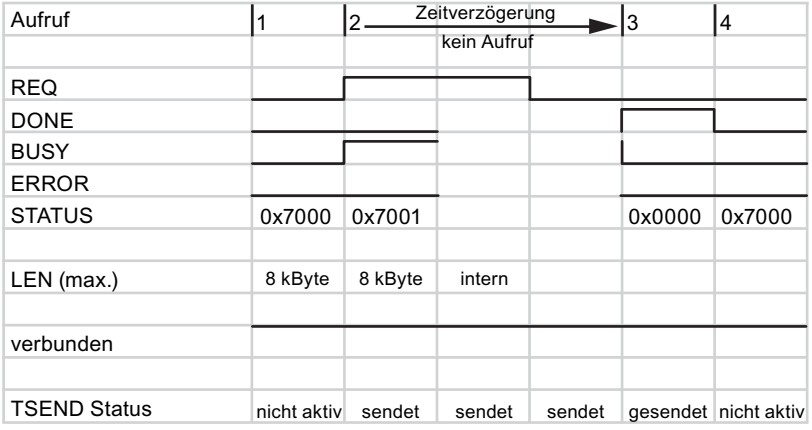

Erfolgt der Sendevorgang über ein CM/CP, verhält sich die Version 4.0 der der Anweisungen TSEND / TUSEND unterschiedlich. In diesem Fall muss die Anweisung mehrfach aufgerufen werden, bis durch den Parameter NDR der Anweisungen TRCV / TURCV der Empfang der Daten bestätigt wird.

Sendevorgang bei Verbindungsunterbrechung

Wird die Verbindung während des Sendevorgangs unterbrochen, zeigen die Parameter ERROR und STATUS den Fehler und die Fehlerursache an. In dem folgenden Beispiel kam des beim Aufruf Nr.5 zu einer Verbindungsunterbrechung, die durch den Parameter STATUS entsprechend angezeigt wird.

Ab der Version 4.0 der Anweisungen TSEND / TUSEND stehen ihnen mehr STATUS-Meldungen zur Verfügung, die entsprechend ausgewertet werden können (siehe Beschreibung TSEND (Seite 3705) / TUSEND (Seite 3733)) .

Aufruf	1	2	3	4	5	6	7	8	9
REQ									
DONE									
BUSY									
ERROR									
STATUS	0x7000	0x7001	0x7002	0x7002	0x80C4	0x7000	0x7001	0x7002	0x7002
LEN (max.)	8 kByte	8 kByte	intern			8 kByte	8 kByte	intern	
Verbindung		aufgebaut		unterbrochen				aufgebaut	
TSEND Status	nicht aktiv	sendet	sendet	sendet	Fehler	nicht aktiv	sendet	sendet	sendet

Siehe auch

TSEND: Daten über Kommunikationsverbindung senden (Seite 3702)

Änderungen bei den Anweisungen TRCV / TURCV

Empfangsvorgang bei bestehender Verbindung

Tritt bei dem Sendevorgang keine Verzögerung oder Unterbrechung auf, verhalten sich die Version 3.0 und 4.0 der Anweisungen TRCV / TURCV gleich:

- Maximal können bei TRCV 8 kByte Daten übertragen werden. Bei TURCV maximal 1472 Bytes (ab Firmware-Stand V2.5 der S7-1500-CPUs bei Unicast oder Multicast: 2048 Bytes). Die Angaben beziehen sich jeweils auf die vollständige Abarbeitung der Anweisung, d. h. alle nötigen Aufrufe, bis der Parameter DONE gesetzt wird.
- Die Anweisung empfängt Daten, wenn der Parameter EN_R auf "1" gesetzt ist.
- Der Datenempfang ist erst abgeschlossen, wenn die am Parameter LEN definierte Datenlänge empfangen wurde. Die Länge der empfangenen Daten wird am Ausgangsparameter RCVD_LEN ausgegeben. Nur dann sind die Daten in dem Bereich verfügbar, der am Parameter DATA definiert wurde.

Aufruf	1	2	3	4	5	6
EN_R						
LEN (max.)	8 kByte	8 kByte	intern			
NDR						
BUSY						
ERROR						
STATUS	0x7000	0x7001	0x7002	0x7002	0x0000	0x7000
RCVD_LEN	0	0	0	0	max. 8 kB	0
Verbindung				aufgebaut		
TRCV Status	nicht aktiv	empfängt	empfängt	empfängt	empfängt	nicht aktiv

Empfangsvorgang bei zeitverzögertem Aufruf

Erfolgt der Sendevorgang nicht über ein Kommunikationsmodul (CM) oder einen Kommunikationsprozessor (CP) verhalten sich die Versionen 3.0 und 4.0 der Anweisungen TRCV / TURCV gleich:

- In dem folgenden Beispiel wird TRCV oder TURCV nur einmal mit einer positiven Flanke am Parameter EN_R aufgerufen (siehe Aufruf 2 in folgender Grafik).
- Ist die Zeitverzögerung ausreichend lang und konnten die Daten übertragen werden, wird im nächsten Aufruf (hier Aufruf 3) der Parameter NDR direkt auf "1" gesetzt.

Aufruf	1	2	Verzögerung kein Aufruf		3	4
EN_R						
LEN (max.)	8 kByte	8 kByte	intern			
NDR						
BUSY						
ERROR						
STATUS	0x7000	0x7001			0x0000	0x7000
RCVD_LEN	0	0	0	0	max. 8 kB	0
Verbindung				aufgebaut		
TRCV Status	nicht aktiv	empfängt	empfängt	empfängt	empfängt	nicht aktiv

Erfolgt der Sendevorgang über ein CM/CP, verhält sich die Version 4.0 der Anweisungen TSEND / TUSEND unterschiedlich. In diesem Fall muss die Anweisung mehrfach aufgerufen werden, bis durch den Parameter NDR der Empfang der Daten bestätigt wird.

Empfangsvorgang bei Verbindungsunterbrechung

Wird die Verbindung während des Empfangsvorgangs unterbrochen, zeigen die Parameter ERROR und STATUS den Fehler und die Fehlerursache an. In dem folgenden Beispiel kam des beim Aufruf Nr.5 zu einem temporären Kommunikationsfehler, der durch den Parameter STATUS entsprechend angezeigt wird.

Aufruf	1	2	3	4	5	6	7	8	9
EN_R									
LEN									
NDR									
BUSY									
ERROR	0x7000	0x7001	0x7002	0x7002	0x80C4	0x7000	0x7001	0x7002	0x7002
STATUS	8 kByte	8 kByte	intern			8 kByte	8 kByte	intern	
RCVD_LEN								aufgebaut	
Verbindung				aufgebaut					
TRCV State	nicht aktiv	sendet	sendet	sendet	Fehler	nicht aktiv	sendet	sendet	sendet

Ab der Version 4.0 der Anweisungen TRCV / TURCV stehen ihnen mehr STATUS-Meldungen zur Verfügung, die entsprechend ausgewertet werden können (siehe Beschreibung TRCV (Seite 3714) / TURCV (Seite 3738)).

Empfangsvorgang bei Verwendung des ADHOC-Modus

Der ADHOC-Modus steht nur bei der Protokoll-Variante TCP zur Verfügung. Sie könnne den ADHOC-Modus verwenden, um mit der Anweisung TRCV / TURCV Daten mit variabler Länge zu emf angen. Ist der ADHOC-Modus aktiv, wird der Empfang von Daten am Parameter NDR bestätigt, sobald mindestens ein Byte übertragen wurde.

Datenempfang im ADHOC-Modus bei TRCV < 3.0 (S7-1200 < V4.0)

In der älteren Version von TRCV (Seite 3709) erfolgt die Aktivierung des ADHOC-Modus, indem der Parameter LEN auf "0" gesetzt wird. In dem folgenden Beispiel werden beim 5. Aufruf 10 Byte Daten übertragen.

Aufruf	1	2	3	4	5	6
EN_R						
LEN (max.)	0 = Aktivierung des ADHOC-Modus					
NDR						
BUSY						
ERROR						
STATUS	0x7000	0x7001	0x7002	0x7002	0x0000	0x7000
RCVD_LEN	0	0	0	0	10 Byte	0
Verbindung				aufgebaut		
TRCV Status	nicht aktiv	empfängt	empfängt	empfängt	empfängt	nicht aktiv

Datenempfang im ADHOC-Modus bei TRCV ≥ 3.0 (S7-1200 ≥ V4.0 oder S7-1500)

Ab der Version 3.0 der Anweisung TRCV wird der ADHOC-Modus über einen eigenen Parameter (ADHOC) aktiviert.

Aufruf	1	2	3	4	5	6
EN_R						
LEN	max.8 kB	max.8 kB	intern			
ADHOC						
NDR						
BUSY						
ERROR						
STATUS	0x7000	0x7001	0x7002	0x7002	0x0000	0x7000
RCVD_LEN	0	0	0	0	10 Byte	0
Verbindung				aufgebaut		
TRCV Status	nicht aktiv	empfängt	empfängt	empfängt	empfängt	nicht aktiv

Tritt bei der Bearbeitung ein Fehler auf, stehen ihnen ab der Version 4.0 der Anweisungen TRCV mehr STATUS-Meldungen zur Verfügung, die entsprechend ausgewertet werden können.

Erfolgt der Sendevorgang über ein CM/CP, verhält sich die Version 4.0 der der Anweisungen TRCV / TURCV unterschiedlich. In diesem Fall muss die Anweisung mehrfach aufgerufen werden, bis durch den Parameter NDR der Empfang der Daten bestätigt wird.

4.1.6.3 OPC UA

Client-Anweisungen

OPC UA-Anweisungen für Client-Programme

Einleitung

Die folgenden Kapitel zeigen anhand von Programmbeispielen, wie Sie die OPC UA-Anweisungen für Client-Programme verwenden.

Die Programmbeispiele basieren auf einer Anlage, bei der eine S7-1500 CPU als Server und eine andere S7-1500 CPU als Client fungiert:

- Der Server stellt PLC-Variablen zum Lesen und Schreiben sowie eine Server-Methode bereit.
- Der Client greift auf die bereitgestellten Werte zu und ruft die Server-Methode auf.

STEP 7 hilft Ihnen bei der Versorgung der Parameter der Anweisungen, wenn Sie eine Client-Schnittstelle anlegen und die Verbindungsparametrierung für OPC UA verwenden. Dadurch müssen Sie viele Variablen für die Versorgung der Parameter nicht manuell anlegen. Wie Sie manuell arbeiten, beschreibt beispielhaft das Kapitel zur Anweisung "OPC_UA_TranslatePathList (Seite 3831)".

Wie Sie die komfortable Parametrierung für OPC UA nutzen, ist in den folgenden Kapiteln für die OPC UA-Anweisungen beschrieben.

Programmbeispiele in SCL

Die Programmbeispiele verwenden die Programmiersprache Structured Control Language (SCL).

Zu jedem Programmbeispiel ist der komplette Sourcecode verfügbar.

Standardisierte Anweisungen nach PLCopen

Die OPC UA-Anweisungen für Clients sind weitgehend gemäß der Spezifikation "PLCopen OPC-UA-Client for IEC 61131-3" umgesetzt.

Durch die Standardisierung wird es leichter, ein Anwenderprogramm auf eine Steuerung eines anderen Herstellers zu portieren, der ebenfalls die OPC UA-Anweisungen gemäß der PLCopen zur Verfügung stellt.

Wenn Sie die Grundstrukturen und Grundabläufe bei OPC UA Clients bereits kennen, finden Sie sich daher leichter zurecht.

Viele Client-Anweisungen arbeiten mit Listen

Viele OPC UA-Client-Anweisungen verarbeiten nicht nur einen Wert, sondern Listen.

So erhält zum Beispiel die Anweisung "OPC_UA_NamespaceGetIndexList" eine Liste mit Namensräumen und gibt eine Liste mit den Indizes für diese Namensräume zurück.

Leistungsfähigkeit eines OPC UA-Servers

OPC UA-Server geben selbst Auskunft über ihre Leistungsfähigkeit.

Das folgende Bild zeigt den Adressraum einer S7-CPU des Typs 1516-3 PN/DP.

Im Knoten "MaxNodesPerRead" ist zum Beispiel angegeben, wie viele PLC-Variablen mit einem Aufruf der Anweisung "OPC_UA_ReadList" gleichzeitig gelesen werden können.

Begrenzt wird die obere Feldgrenze des Arrays, mit dem Sie den Parameter "NodeHdls" versorgen können, durch die Eigenschaften des verwendeten OPC UA-Clients (siehe Technische Daten des Clients).

Über alle Client-Schnittstellen (d. h. über alle Listen hinweg) gelten folgende Grenzen:

- 1000 Knoten für CPU 1510SP (F), CPU 1511 (C/F/T/TF), CPU 1512C, CPU 1512SP (F), CPU 1513 (F)
- 2000 Knoten für CPU 1505 (S/SP/SP F/SP T/SP TF), CPU 1515 (F/T/TF), CPU 1515 SP PC (F/T/TF), CPU 1516 (F/T/TF)
- 5000 Knoten für CPU 1507S (F), CPU 1517 (F/T/TF), CPU 1518 (F)

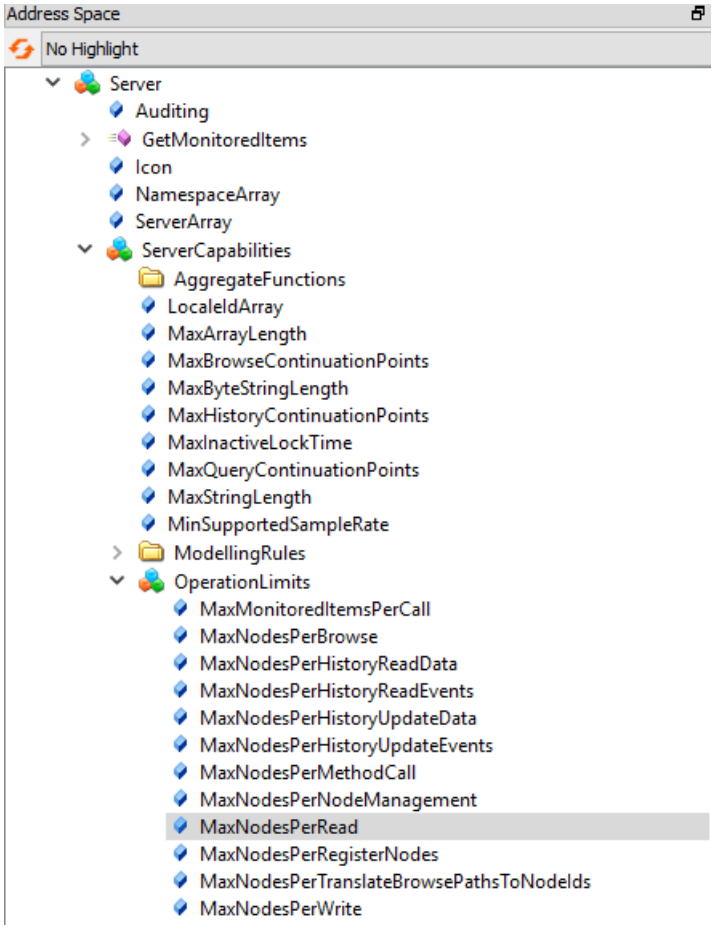

Das Bild zeigt einen Ausschnitt aus dem Adressraum eines OPC UA-Servers, dargestellt mit dem Programm "UaExpert" von Unified Automation.

Fehlernummern

Die Fehlernummern für die Anweisungen sind folgendermaßen unterteilt:

- Wenn die Fehlernummer mit "16#B0" oder "16#C0" beginnt, dann wurde die Nummer von Siemens vergeben, siehe Fehlercodes von Siemens (Seite 3931).
  Beispiel: Folgende Fehlernummern sind übergreifend und können prinzipiell bei jeder der Client-Anweisungen vorkommen:

B080_C400	ClientNotEnabled	Client deaktiviert. Aktivieren Sie den Client im TIA Portal.
B080_C500	ClientNotAvailable	Fehler beim Initialisieren des Clients
C080_C300	Simatic_OutOfResources	Fehler bei Speicherallokation oder zu viele Instanzen der Anweisung

- Wenn die Fehlernummer mit "16#80" beginnt, dann wurde die Nummer von der OPC Foundation vergeben, siehe Fehlercodes der OPC Foundation (Seite 3933).
- Wenn die Fehlernummer mit "16#A0" beginnt, dann wurde die Nummer von PLCopen vergeben, siehe Fehlercodes von PLCopen (Seite 3943).
  Eine Liste mit Fehlernummern der PLCopen finden Sie auch im Kapitel 4 der Spezifikation "PLCopen OPC UA Client for IEC 61131-3". Die Spezifikation ist bei PLCopen erhältlich.

Datenaustausch vorbereiten

OPC_UA_Connect: Verbindung erzeugen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_Connect" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Die Anweisung "OPC_UA_Connect" baut eine Verbindung zu einem OPC UA-Server auf.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

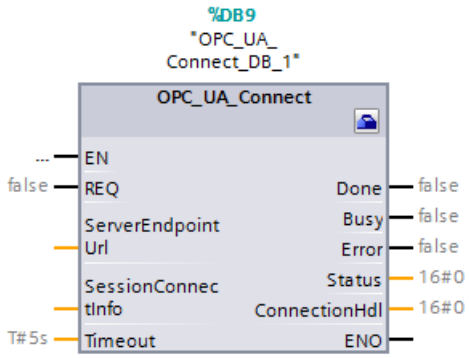

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung dient zur Vorbereitung des Datenaustauschs zwischen einem OPC UA-Server und -Client, siehe Erläuterung zu ① in der folgenden Grafik:

Die Anweisung liefert ein Verbindungshandle (eine numerische Referenz) zurück, das die nachfolgenden Anweisungen verwenden, um sich auf die Verbindung zu beziehen.

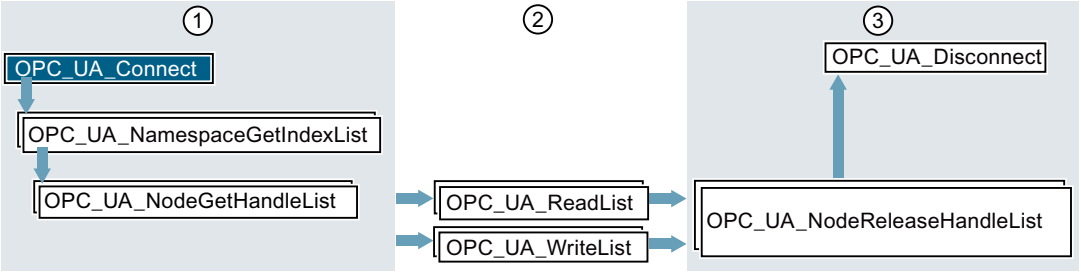

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen (je nach Anwendung rufen Sie die Anweisungen wiederholt auf)

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_Connect"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ServerEndpointUrl	InOut	VARIANT	Zeiger auf eine Variable des Typs STRING oder WSTRING mit der Adresse (URL) des OPC UA-Servers. Nur IPv4-Adressen sind zugelassen.
SessionConnectInfo	InOut	VARIANT	Zeiger auf die Verbindungsbeschreibung in einer Variablen mit dem Systemdatentyp "SessionConnectInfo (Seite 3922)".

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Timeout	Input	TIME	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Der Parameter "Timeout" definiert den Service-Timeout für unterlagerte OPC UA Service Requests, die der Client zum Server sendet. Timeout-Werte kleiner als 100 ms bzw. negative Werte werden auf 100 ms gesetzt, da kleinere Werte nicht sinnvoll sind.
Done	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status".
ConnectionHdl	Output	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Dieses Handle benötigen andere OPC UA-Anweisungen als Eingangsparameter.

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten. • Allgemein: Reduzieren Sie die Anzahl der über OPC UA angemeldeten Elemente. Stellen Sie mit einem OPC UA Server Diagnose Tool (wie z.B. UaExpert von Unified Automation) sicher, dass keine inaktiven Sessions und Subscriptions vorhanden sind. Als Gegenmaßnahme können Sie entsprechende Timeouts (Subscription Timeout <= Session Timeout) im Client wählen.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
8005_0000	OpcUa_BadCommunicationError	Verbindungsaufbau gescheitert. Mögliche Ursachen: • Die Server-Adresse (ServerEndpointURL) ist falsch oder unvollständig. • Der Server akzeptiert die vom Client beim Verbindungsaufbau übermittelte Puffergröße nicht. • Die für den Verbindungsaufbau benötigten Verbindungsressourcen des Clients sind erschöpft. Abhilfen: • Prüfen und korrigieren Sie die ServerEndpointURL • Überprüfen Sie die zulässige Puffergröße. Der Client der S7-1500 CPU stellt 8192 Bytes zur Verfügung. Diesen Wert muss der Server akzeptieren. • Bei einem Auf- und Abbau der Verbindung in schneller Folge ist die Frequenz zu verringern.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Ursachen: • Verbindung zum OPC UA-Server ist zu langsam. • Netzwerklast ist zu hoch. • Der OPC UA-Server ist nicht verfügbar. • Der Server akzeptiert möglicherweise die vom Client beim Verbindungsaufbau übermittelte Puffergröße nicht. Abhilfen: • Überprüfen Sie die URL des OPC UA-Servers. • Erhöhen Sie den Timeout-Wert der Anweisung OPC_UA_Connect. • Überprüfen Sie die zulässige Puffergröße. Der Client der S7-1500 CPU stellt 8192 Bytes zur Verfügung. Diesen Wert muss der Server unterstützen.
8010_0000	OpcUa_BadTooManyOperations	Die Anforderung (Request) konnte nicht ausgeführt werden, weil die Anzahl zu lesender bzw. zu schreibender Knoten zu groß ist. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der Knoten, auf die Sie - mit einem Aufruf - zugreifen wollen.
803C_0000	OpcUa_BadOutOfRange	Bereichsgrenzen verletzt. Mögliche Abhilfe bei Arrays: • Kontrollieren Sie, ob Sie auf Elemente eines Arrays zugreifen, die nicht definiert sind.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
803D_0000	OpcUa_BadNotSupported	Mögliche Ursachen: • Die Server-Adresse (ServerEndpointURL) ist falsch oder unvollständig. • Server unterstützt die angeforderte Operation nicht.
8054_0000	OpcUa_BadSecurityModeInsufficient	Der Server stellt höhere Sicherheitsanforderungen (Security Policy). Mögliche Abhilfe: • Verwenden Sie eine höhere Sicherheitseinstellung für die Verbindung zum Server.
8055_0000	OpcUa_BadSecurityPolicyRejected	Server unterstützt angeforderte Security Policy oder Transport Profile nicht
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_2200	Simatic_BadValue_VariantInput2	Der Wert der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert. Abhilfe: Aktivieren Sie den OPC UA-Client in den Eigenschaften der CPU (Bereich OPC UA > Client).
B080_C500	Simatic_NothingToDo	Fehler beim Initialisieren des Clients.
B080_C600	Simatic_ClientNotAvailable	Fehler beim Initialisieren des Clients.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Fehler bei Speicherallokation oder zu viele Instanzen der Anweisung. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Die folgenden Abschnitte zeigen, wie Sie die Anweisung "OPC_UA_Connect" in einem Programm verwenden, das Daten mit einem OPC UA-Server austauscht.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben in den Eigenschaften dieser Client-Schnittstelle eine Verbindung zu einem OPC UA-Server parametriert.

Sichere Verbindungen parametrieren

Mit der Anweisung "OPC_UA_Connect" ist es zwar möglich, eine unsichere Verbindung zu einem OPC UA-Server aufzubauen. Sie sollten aber stets eine sichere Verbindung verwenden.

So verwenden Sie eine Client-Schnittstelle

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Fügen Sie dem Ordner "Programmbausteine" einen neuen Funktionsbaustein hinzu.
   Im Beispiel heißt der Funktionsbaustein "ReadFromProductionline".
   Verwendete Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_Connect" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_Connect_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_Connect_Instance".

   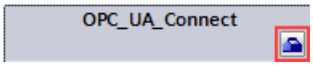

   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.

6. Unter "Client-Schnittstelle für die OPC UA-Verbindung auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline".

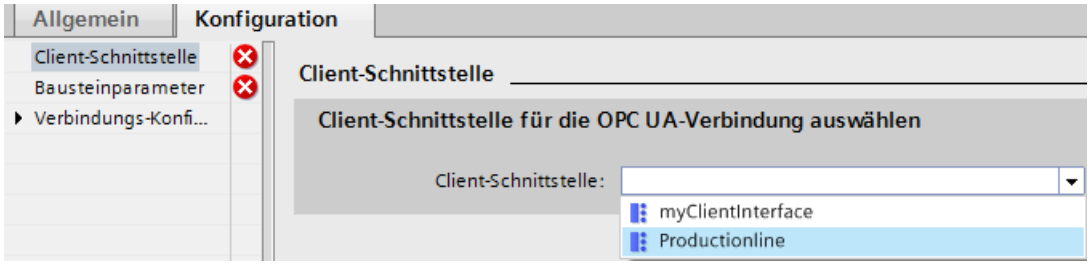

   STEP 7 versorgt automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.

7. Klicken Sie auf "Bausteinparameter" und weisen Sie den restlichen Parametern (REQ, Busy, Done, Error, Status) manuell Variablen zu.
   STEP 7 fügt die gewählten Variablen in den Funktionsaufruf ein.

Aufruf der Anweisung

Der folgende Quellcode zeigt, wie Sie die Anweisung "OPC_UA_Connect" verwenden, um eine Verbindung zu einem OPC UA-Server herzustellen.

Das vollständige Programm finden Sie im Kapitel "AUTOHOTSPOT"

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Programmabschnitte (Cases) unterteilt. Die Variable #State steuert, welcher Programmabschnitt ausgeführt wird.

Im 1. Programmabschnitt wird eine Verbindung aufgebaut:

SCL

```
1:  // case 1, connect to server
            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                     SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                     ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                     Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                     REQ := #Req,
                                     Done => #Done,
                                     Busy => #Busy,
                                     Error => #Error,
                                     Status => #Status);


            IF #Done = TRUE THEN
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                // Did we get a connection handle?
                IF "Productionline_Configuration".Connection.ConnectionHdl <> 0 THEN
                    // We have to release all resources in the server and disconnect
                    #State := 100;
                    //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                ELSE
                    #State := 99;
                END_IF;

                #Mem_Status := #Status;
                // set parameter REQ of OPC_UA_Connect to FALSE
                #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                         SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                         REQ := FALSE,
                                         ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                         Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;
```

Das Beispielprogramm ruft die Anweisung "OPC_UA_Connect" aus den folgenden Gründen auf:

- Um die Verbindung zu einem OPC UA-Server aufzubauen.
- Um im Fehlerfall den Parameter REQ auf den Wert "FALSE" zu setzen.

Asynchrone Ausführung

Die Anweisung "OPC_UA_Connect" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung

Dieser Programmteil baut eine Verbindung zu dem OPC UA-Server auf, den Sie bei der Parametrierung der Client-Schnittstelle eingetragen haben.

Die Variable "#State" enthält beim Start des Programms den Wert "1".

Wenn die Anweisung "OPC_UA_Connect" noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass #Req auf TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Wenn der Ausgangsparameter "Done" den Wert TRUE besitzt, dann konnte eine Verbindung aufgebaut werden. Durch den Erfolgszustand wird der Wert der Variable #State um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt (2:) ausgeführt, siehe OPC_UA_NamespaceGetIndexList: Namensraum-Indexe lesen (Seite 3810) . Durch den Erfolgszustand wird außerdem am Ausgangsparameter "ConnectionHdl" die numerische Referenz für die aufgebaute Verbindung angezeigt.

Falls ein Fehler aufgetreten ist, hat der Ausgangsparameter "Error" den Wert TRUE.

- Wenn im Fehlerfall ein Verbindungshandle zurückgeliefert wird, müssen Ressourcen im Server freigeben werden. Dazu wird der Variablen #State den Wert 100 zugewiesen.
- Wenn im Fehlerfall kein Verbindungshandle zurückgeliefert wird, wird der Variablen #State der Wert 99 zugewiesen.

Die Programmabschnitte 99 und 100 sind für Fehlerbehandlungen vorgesehen.

Siehe auch

Beispielprogramm für das Lesen von PLC-Variablen (Seite 3950)

OPC_UA_TranslatePathList: Knotenparameter lesen (Seite 3831)

Fehlercodes von Siemens (Seite 3931)

OPC UA-Anweisungen für Client-Programme (Seite 3797)

OPC_UA_NamespaceGetIndexList: Namensraum-Indexe lesen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_NamespaceGetIndexList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_NamespaceGetIndexList" erfragen Sie die aktuellen Indizes der Namensräume in einem OPC UA-Server.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

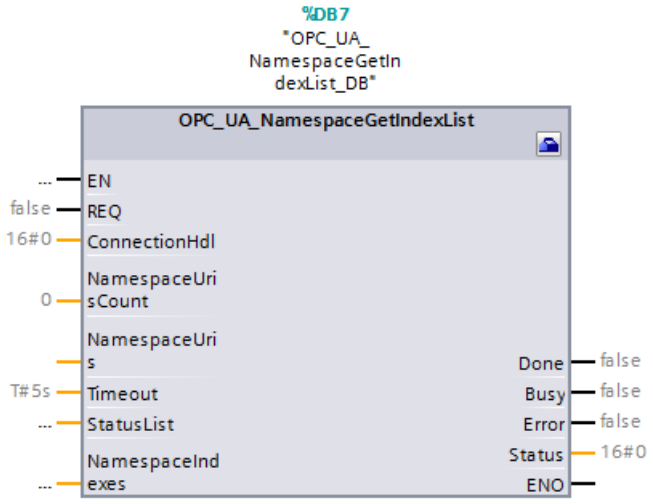

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_NamespaceGetIndexList" dient zur Vorbereitung des Datenaustauschs mit einem OPC UA-Server, siehe ① in der folgenden Grafik.

Um eine PLC-Variable zu lesen, zu schreiben oder um eine Methode aufzurufen müssen Sie den Index des Namensraums kennen, in dem die PLC-Variable liegt. Der Index ist ein Teil der Adresse (NodeID) dieser PLC-Variablen im Adressraum des OPC UA-Servers.

Die Anweisung liefert eine Liste mit den Indizes für die Namensräume des OPC UA-Servers zurück.

Diese Liste wird benötigt, um über die Anweisung "OPC_UA_NodeGetHandleList" die NodeIDs zu erfragen, mit denen die Lese- und Schreibeaufträge abgesetzt werden können.

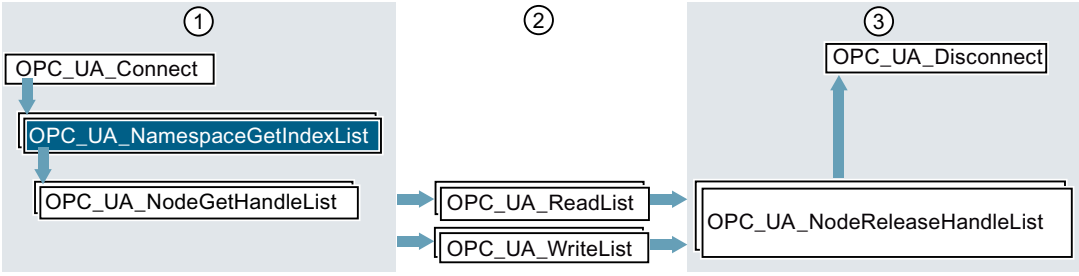

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_NamespaceGetIndexList"

Tabelle 4-73 Die Parameter der Anweisung "OPC_UA_NamespaceGetIndexList"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung OPC_UA_Connect: Verbindung erzeugen (Seite 3799).
NamespaceUrisCount	Input	UINT	Anzahl der Namensräume im Parameter NamespaceUris
NamespaceUris	InOut	VARIANT	Zeiger auf ein Array des Typs STRING oder WSTRING. Das Array enthält die URIs der einzelnen Namensräume, deren Indizes erfragt werden sollen.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei OPC_UA_Connect: Verbindung erzeugen (Seite 3799).
StatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD (optional). Das Array enthält die Fehlercodes für die einzelnen Namensräume. Für jeden Namensraum ist angegeben, ob ihm ein Index zugeordnet werden konnte.
NamespaceIndexes	InOut	VARIANT	Zeiger auf ein Array des Typs UINT. Das Array enthält die Indizes der einzelnen Namensräume, die vom OPC UA-Server erfragt wurden.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Server antwortet nicht mit der erwarteten Anzahl an Namespace-Array-Elementen

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	NamespaceUrisCount ist 0
8010_0000	OpcUa_BadTooManyOperations	Die Anforderung (Request) kann nicht ausgeführt werden, weil zu viele Namensräume gelesen werden sollen. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der Namensräume, auf die Sie - mit einem Aufruf - zugreifen wollen.
809E_0000	OpcUa_BadDataUnavailable	Die Indizes der Namensräume sind zurzeit nicht verfügbar.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
B080_0500	Simatic_BadType_VariantInput5	Der Datentyp der Variablen stimmt nicht, auf die der 5. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	NamespaceUrisCount > MAX_ELEMENTS_NAMESPACES
B080_3100	BadNumElements_VariantInput1	Anzahl Array-Elemente im Variant 1 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist (z. B. NodeIDCount bei OPC_UA_NodeGetHandleList oder NamespaceUrisCount bei OPC_UA_NamespaceGetIndexList etc.).
B080_3200 B080_3300 B080_3400 B080_3500	BadNumElements_VariantInput2 ... BadNumElements_VariantInput5	Anzahl Array-Elemente im Variant 2 (... 5) ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente (Namensräume) enthält.
C080_C300	Simatic_InsufficientResources	Keine ausreichenden Ressourcen. Entweder (a) Fehler bei Speicherallokation oder (b) zu viele SFB-Instanzen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "StatusList"

Der Parameter enthält für jeden einzelnen Namensraum einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Der Namensraum wird gefunden. Der Index des Namensraums wird in das Array eingetragen, auf das der Parameter "NamespaceIndexes" zeigt.
806F_0000	OpcUA_BadNoMatch	Namensraum wird nicht gefunden.
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_NamespaceGetIndexList" in einem Programm verwenden, das Daten mit einem OPC UA-Server austauscht.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben in den Eigenschaften dieser Client-Schnittstelle eine Verbindung zu einem OPC UA-Server parametriert, siehe "AUTOHOTSPOT".

Zudem muss für die Anweisung "OPC_UA_NamespaceGetIndexList" die folgende Voraussetzung erfüllt sein:

- Ein Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungshandle durch den Aufruf der Anweisung OPC_UA_Connect (Seite 3799).

Funktion der Anweisung

Für den Datenaustausch benötigen Sie die Indizes der Namensräume, in denen die Variablen liegen, die Sie über Ihr Programm lesen oder schreiben wollen.

Auch für den Aufruf von Methoden benötigen Sie die aktuellen Indizes der Namensräume, in denen die Methoden liegen.

Bei einem OPC UA-Server ist nach der OPC UA-Spezifikation die Zuordnung zwischen Namensraum und zugehöriger Index nicht fest und kann sich ändern, z. B. wenn neue Namensräume hinzukommen oder Namensräume entfallen.

Deshalb ist es erforderlich, dass Sie die aktuellen Indizes immer beim OPC UA-Server erfragen, bevor Sie Knoten-Handles mit der Anweisung OPC_UA_NodeGetHandleList erfassen. Dazu dient die Anweisung "OPC_UA_NamespaceGetIndexList".

Welche Informationen gibt die Anweisung zurück?

Die Anweisung liefert die folgenden Angaben:

- Eine Liste mit den Indizes für die einzelnen Namensräume (Parameter "NamespaceIndexes"). Die Reihenfolge der Indizes entspricht der Reihenfolge der Namensräume im Parameter "NamespaceUris".
- Eine Liste mit Fehlermeldungen (Parameter "StatusList").
  Jede Fehlermeldung in dieser Liste bezieht sich auf den entsprechenden Namensraum im Parameter "NamespaceUris".
  Sie müssen für jeden einzelnen Namensraum prüfen, ob der Index dieses Namensraums zurückgeliefert werden kann.

Namensräume und deren Indizes mit UaExpert ermitteln

Um die Anweisung verwenden zu können, muss die Namensraum-URI (Uniform Resource Indentifier) bekannt sein. Namesraum-URIs und Indizes der Namensräume (Namespace URI und Node Namespace Index) können Sie mit UaExpert ermitteln, wie folgendes Beispiel zeigt:

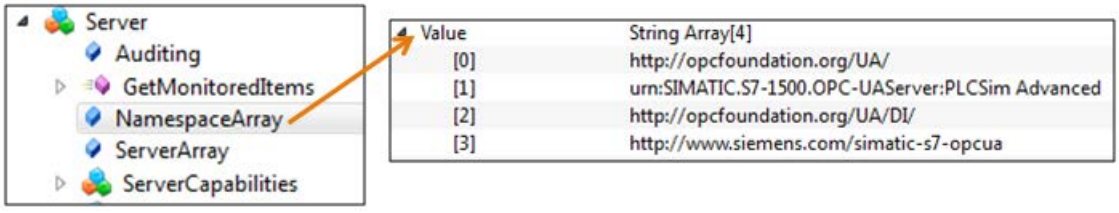

So verwenden Sie eine Client-Schnittstelle

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung enthalten soll.
   Im Beispiel heisst der Funktionsbaustein "ReadFromProductionline".
   Gewählte Sprache: SCL
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_NamespaceGetIndexList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_NamespaceGetIndexList_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_NamespaceGetIndexList_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", siehe "AUTOHOTSPOT".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.
7. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsaufruf ein.

Aufruf der Anweisung

Der folgende Quellcode zeigt, wie Sie die Anweisung "OPC_UA_NamespaceGetIndexList" verwenden, um die Indizes für die Namensräume zu erfragen, die der verbundene OPC UA-Server aktuell verwendet.

Das vollständige Programm finden Sie im Kapitel "Beispielprogramm für das Lesen von PLC-Variablen (Seite 3950)".

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Im 2. Programmabschnitt werden die Indizes der Namensräume erfragt:

SCL

```
2:  // case 2, get index for namespace

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                   StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                   NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   REQ := #Req,
                                                   Done => #Done,
                                                   Busy => #Busy,
                                                   Error => #Error,
                                                   Status => #Status);


            IF #Done = TRUE THEN
               IF "Productionline_Configuration".Namespaces.NamespaceStatusList[0] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NamespaceGetIndexList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                       StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                       NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                       REQ := FALSE);
            END_IF;
```

Das Beispielprogramm ruft die Anweisung "OPC_UA_NamespaceGetIndexList" aus den folgenden Gründen auf:

- Um den Index für einen Namensraum zu erfragen.
- Um im Fehlerfall den Parameter REQ auf den Wert FALSE zu setzen.

Asynchrone Ausführung

Die Anweisung "OPC_UA_NamespaceGetIndexList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung

Im Bild oben ist Fall 2 dargestellt. Dieser Programmteil erfragt die Indizes für die Namensräume.

Wenn die Variable #State den Wert 2 besitzt, dann wird der Fall 2 ausgeführt.

Wenn die Anweisung "OPC_UA_NamespaceGetIndexList" noch nicht ausgeführt wird, ist #Busy gleich FALSE und die Variable "#Req" wird auf TRUE gesetzt. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Wenn im Ausgangsparameter "Error" der Wert TRUE steht, dann trat bei der Ausführung der Anweisung ein Fehler auf. Dadurch wird die Variable #State auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt. Durch den Erfolgszustand wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 3 (Handles für Lese- und Schreibzugrife holen) - ausgeführt. Der Fall 3 ist in OPC_UA_NodeGetHandleList (Seite 3819) beschrieben.

Hinweis

Ausführung der Anweisung

Es ist möglich, dass die Anweisung erfolgreich ausgeführt wird (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber für einen bestimmten Namensraum kein Index zurückgeliefert werden kann.

Deshalb wird im Beispiel zusätzlich geprüft, ob für den zu erfragenden Namensraum ein Index gefunden wird. Wenn ein Index gefunden wird, dann enthält das 1. Element des Arrays, auf das der Parameter "StatusList" zeigt, den Wert 0.

Siehe auch

OPC_UA_TranslatePathList: Knotenparameter lesen (Seite 3831)

OPC_UA_NodeGetHandleList:Handles für Lese- und Schreibzugriffe holen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_NodeGetHandleList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_NodeGetHandleList" registrieren Sie bei einem OPC UA-Server PLC-Variablen.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

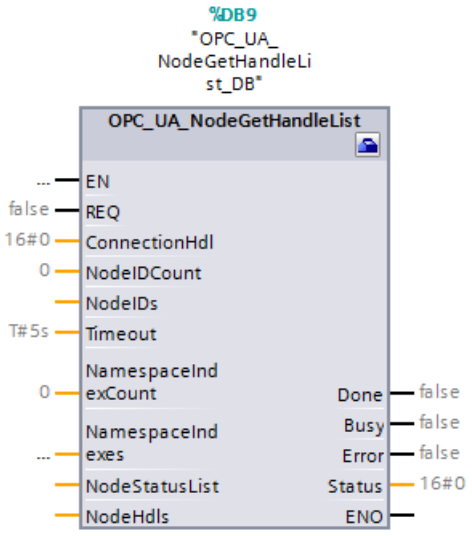

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_NodeGetHandleList" dient zur Vorbereitung des Datenaustauschs mit einem OPC UA-Server, siehe ① in der folgenden Grafik:

Mit dieser Anweisung registrieren Sie beim Server PLC-Variablen, die Sie lesen oder schreiben wollen.

Dadurch wird der spätere Zugriff mit den Anweisungen "OPC_UA_ReadList" und "OPC_UA_WriteList" optimiert.

Die Anweisung liefert eine Liste mit numerischen Referenzen (Handles) für die registrierten PLC-Variablen zurück.

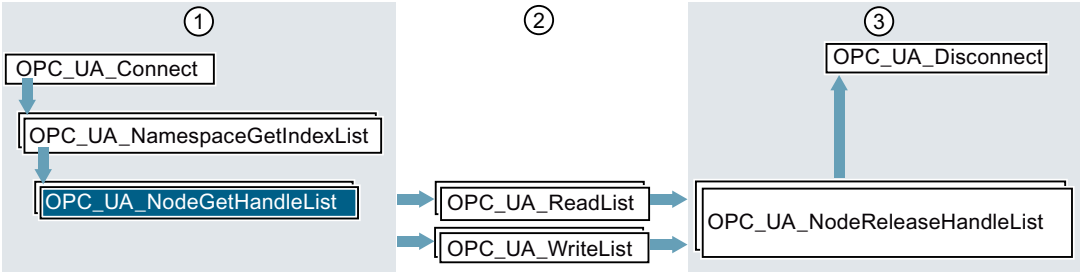

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_NodeGetHandleList"

Tabelle 4-74 Die Parameter der Anweisung "OPC_UA_NodeGetHandleList"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung OPC_UA_Connect.
NodeIDCount	Input	UINT	Anzahl der Elemente im Parameter NodeIDs
NodeIDs	InOut	VARIANT	Zeiger auf ein Array mit Elementen des Typs OPC_UA_NodeId. Das Array enthält die NodeIds der Variablen, für die der OPC UA-Server numerische Referenzen (Handles) bereitstellen soll (siehe oben Funktion des Bausteins).
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter beiAUTOHOTSPOT
NamespaceIndexCount	Input	UINT	Anzahl der Elemente im Parameter NamespaceIndexes.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
NamespaceIndexes	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Dieser Parameter ist nicht erforderlich, wenn es keine Elemente im Parameter NamespaceIndexCount gibt (NamespaceIndexCount = 0). Das Array wird verwendet, um den Namensraum-Indizes im Parameter NodeIDs neue Indizes zuzuordnen. Beispiel: Wenn im NamespaceIndexes[4] der Wert 10 steht, dann wird allen NodeIds im Parameter "NodeIDs", die bislang den Namensraumindex 4 hatten, der Namensraumindex 10 zugeordnet.
NodeStatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die NodeIds der einzelnen Variablen, siehe unten "Fehlernummern für NodeStatusList". Für jede NodeId ist angegeben, ob ihm ein Handle zugeordnet werden konnte. Es kann durchaus sein, dass der Funktionsbaustein erfolgreich ausgeführt wurde (Parameter Error nicht gesetzt), aber für eine bestimmte NodeId (Variable) kein Handle gefunden wurde. Eine NodeId bestehen aus einem Namensraumindex, einem Variablennamen (Identifier) und einem Identifier-Typ. Siehe OPC_UA_NodeID.
NodeHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Handles der einzelnen NodeIds, die vom OPC UA-Server erfragt wurden (siehe oben Funktion des Bausteins).
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Der Server sendete eine Antwort, die nicht erkannt werden konnte.
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
8033_0000	OpcUa_BadNodeIdInvalid	Die Syntax des Knotens ist ungültig.
8034_0000	OpcUa_BadNodeIdUnknown	Die ID des Knotens (NodeId) verweist auf einen Knoten, den es im Adressraum des Servers nicht gibt.
8035_0000	OpcUa_BadAttributeIdInvalid	Das Attribute gibt es nicht in diesem Knoten.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.
B080_0500	Simatic_BadType_VariantInput5	Der Datentyp der Variablen stimmt nicht, auf die der 5. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	NodeIDCount > MAX_ELEMENTS_NODELIST
B080_3100	BadNumElements_VariantInput1	Anzahl Array-Elemente im Variant 1 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist (z. B. NodeIDCount bei OPC_UA_NodeGetHandleList oder NamespaceUrisCount bei OPC_UA_NamespaceGetIndexList etc.).
B080_3200 B080_3300 B080_3400 B080_3500	BadNumElements_VariantInput2 ... BadNumElements_VariantInput5	Anzahl Array-Elemente im Variant 2 (... 5) ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Abhilfe: Reduzieren Sie die Anzahl der offenen Verbindungen.
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "NodeStatusList"

Der Parameter "NodeStatusList" enthält für jede einzelne NodeId (Variable) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Die PLC-Variable (NodeId) wurde gefunden. Das Handle der Variablen wurde in das Array eingetragen, auf das der Parameter "NodeHdls" zeigt.
8033_0000	OpcUa_BadNodeIdInvalid	Die Syntax der NodeId ist nicht korrekt.
8034_0000	OpcUa_BadNodeIdUnknown	Die NodeId bezieht sich auf einen Knoten (eine Variable), die es im OPC UA-Server nicht gibt.
A000_0309	PLCopenUA_Bad_OutOfHandles	Die NodeId wurde nicht registriert, weil die maximale Anzahl an Knoten (PLCOPEN_MAXREGISTEREDNODES) erreicht wurde.
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_NodeGetHandleList" in einem Programm verwenden, das Daten mit einem OPC UA-Server austauscht.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben in den Eigenschaften dieser Client-Schnittstelle eine Verbindung zu einem OPC UA-Server parametriert, siehe "AUTOHOTSPOT".

Zusätzlich müssen für die Anweisung "OPC_UA_NodeGetHandleList" die folgenden Voraussetzungen erfüllt sein:

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle von der Anweisung OPC_UA_Connect.
- Indizes der Namenräume, in denen die Variablen liegen, die Sie mit dem Client lesen oder schreiben wollen.
  Sie erhalten die Indizes von der Anweisung OPC_UA_NamespaceGetIndexList.

Funktion der Anweisung

Um die Zugriffe auf Variablen zu optimieren, bietet OPC UA den Service "RegisterNodes" für wiederholte Zugriffe.

Mit Hilfe dieses Services können Server den optimierten Zugriff auf Variablen vorbereiten. Die Anweisung "OPC_UA_NodeGetHandleList" ruft implizit diesen Service auf, um den Server auf optimierte Zugriffe (im Sprachgebrauch von OPC UA "Registered Read/Write") vorzubereiten.

Daher erhalten Sie numerische Referenzen (Handles) vom OPC UA-Server durch den Aufruf der Anweisung "OPC_UA_NodeGetHandleList". Diese Handles verwenden Sie bei den Anweisungen "OPC_UA_ReadList" oder "OPC_UA_WriteList", was einem optimierten Zugriff mit "Registered Read/Write" entspricht.

Welche Informationen sind erforderlich?

Zum Erfragen der numerischen Referenzen (Handles) benötigt die Anweisung die folgenden Angaben:

- Eine Liste mit den Variablen, die gelesen oder gesetzt werden sollen (Parameter "NodeIds", siehe Systemdatentyp "OPC_UA_NodeId").
  Diese Liste enthält für jede Variable folgende Angaben:
  - Index des Namensraums, in dem die Variable liegt
  - Identifier der Variablen (Variablenname)
  - Typ des Identifiers

Welche Informationen gibt die Anweisung zurück?

Die Anweisung liefert die folgenden Angaben:

- Eine Liste mit den Handles für die Variablen ("NodeHdls").
  Die Reihenfolge der Handles entspricht der Reihenfolge der Variablen im Parameter NodeIDs.
- Eine Liste mit Fehlermeldungen (Parameter "NodeStatusList").
  Jede Fehlermeldung in dieser Liste bezieht sich auf die entsprechende Variable im Parameter NodeIds.
  Sie müssen für jede einzelne Variable prüfen, ob der OPC UA-Server ein Handle für diese Variable zurückliefern konnte.

So verwenden Sie eine Client-Schnittstelle

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung enthalten soll.
   Im Beispiel heisst der Funktionsbaustein "ReadFromProductionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_NodeGetHandleList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_NodeGetHandleList_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_NodeGetHandleList_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel wird die Client-Schnittstelle "Productionline" gewählt, die für die Beispielanlage angelegt wurde, siehe "AUTOHOTSPOT".

7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Leseliste.
   Im Beispiel: Leseliste "ReadListProduct".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.
8. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Das vollständige Programm finden Sie im Kapitel "AUTOHOTSPOT"

Das Beispielprogramm verwendet die Leseliste "ReadListProduct", die die folgenden PLC-Variablen enthält:

- NewProduct
- ProductNumber

Für diese Variablen erhält das Programm numerische Referenzen (Handles) vom Server. Das Kapitel "AUTOHOTSPOT" zeigt, wie Sie eine Client-Schnittstelle anlegen und ihr eine Leseliste mit PLC-Variablen hinzufügen.

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Im 3. Programmabschnitt werden die Handles für die zu lesenden oder schreibenden Knoten (PLC-Variablen) erfragt:

SCL

```
3:  // case 3, get handles for nodes

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".ReadLists."ReadListProduct".Nodes,
                                               NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                               NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                               REQ := #Req,
                                               ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                               Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                               NodeIDCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                               NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                               NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                               Done => #Done,
                                               Error => #Error,
                                               Busy => #Busy,
                                               Status => #Status);
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_NodeGetHandleList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Der Programmteil im Bild oben ruft die Anweisung "OPC_UA_NodeGetHandleList" auf, um Handles für PLC-Variablen zu erfragen.

Wenn die Anweisung noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
            IF #Done = TRUE THEN
                IF "Productionline_Data"."ReadListProduct".NodeStatusList[0] = 0 AND
"Productionline_Data"."ReadListProduct".NodeStatusList[1] = 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NodeGetHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".ReadLists."ReadListProduct".Nodes,
                                                   NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                   NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                   REQ := FALSE,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   NodeIDCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                                   NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall wird der Wert des Ausgangsparameters "Error" auf TRUE gesetzt. Dadurch wird der Wert der Variable "#State" auf 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen. Zudem ruft das Beispielprogramm die Anweisung "OPC_UA_NodeGetHandleList" auf, um den Parameter REQ auf FALSE zu setzen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt.

Achtung: Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber für eine bestimmte PLC-Variable kein Handle zurückgeliefert werden konnte.

Im Beispielprogramm wird deshalb geprüft, ob für jede PLC-Variable ein Handle zurückgeliefert wird.

- Kommt für die 1. Variable ein Handle zurück, steht im 1. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
- Kommt für die 2. Variable ein Handle zurück, steht im 2. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.

Wenn beide Bedingungen erfüllt sind, zeigt der Ausgangsparameter "NodeHdls" auf eine gültige Liste mit den Handles für die registrierten PLC-Variablen.

Der Wert der Variable "#State" wird um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 4 - ausgeführt.

Fall 4 ist in OPC_UA_ReadList (Seite 3844) beschrieben.

Hinweis

Wenn Sie in Ihrer Anwendung sehr viele Variablen mit OPC_UA_NodeGetHandleList registrieren, dann kann es sinnvoll sein, #State immer bei #Done = TRUE um eins zu erhöhen. Überprüfen Sie dann bei der Verwendung der abgefragten Werte, ob die einzelnen Werte gültig sind.

Wozu dient der Parameter "NamespacesIndexes"?

Der Parameter "NamespaceIndexes" dient zur Umsetzung der Indizes für die Namensräume.

Das Umsetzen der Indizes ist zum Beispiel notwendig, wenn Sie in Ihrem Anwenderprogramm die Bezeichner (Identifier) von Knoten aus einer externen Liste lesen, zum Beispiel aus einer exportierten OPC UA-XML-Datei.

In diesem Fall müssen Sie in der Liste zu allen Identifiern den Index nachtragen, den der OPC UA-Server aktuell für den Namensraum verwendet.

Zudem können sich die Indizes der Namensräume ändern, sodass Sie die Liste jedes Mal anpassen müssen.

Automatische Änderung der Indizes

Sie können die Indizes der Namensräume automatisch umsetzen lassen: Dazu nutzen Sie die Parameter "NamespaceIndexes" und "NamespacesIndexCount" der Anweisung "OPC_UA_NodeGetHandleList".

Gehen Sie folgendermaßen vor:

1. Prüfen Sie, welcher Namensraum den NodeIds zugewiesen ist, die Sie aus einer Liste einlesen.
   Im Beispiel wird den NodeIds der Namensraumindex 0 zugewiesen.
   Im Beispiel liegen die NodeIds im Siemens-Namensraum "http://www.siemens.com/simatic-s7-opcua".
2. Legen Sie ein Array des Typs "WString" an.
   Im Beispiel heisst das Array "myNamespaces".
3. Tragen Sie den Siemens-Namensraum "http://www.siemens.com/simatic-s7-opcua" unter "myNamespaces[0]" ein (also im Arrayelement 0).

4. Versorgen Sie den Parameter "NamespaceUris" der Anweisung "OPC_UA_NamespaceGetIndexList" mit dem neuen Array "myNamespaces". Die Anweisung stellt das Ergebnis (den aktuellen Index) am Parameter "NamespaceIndexes" bereit.
5. Legen Sie für das Ergebnis ein Array des Typs "UInt" an. Im Beispiel heisst das Array "myNamespaceIndexes".
6. Versorgen Sie den Parameter "NamespaceIndexes" der Anweisung "OPC_UA_NamespaceGetIndexList" mit dem neuen Array "myNamespaceIndexes".
7. Weisen Sie dieses Array "myNamespaceIndexes" auch dem Parameter "NamespacesIndexes" der Anweisung "OPC_UA_NodeGetIndexList" zu. Diese Anweisung liest dann aus "myNamespaceIndexes[0]" den aktuellen Index des Siemens-Namensraums und verwendet diesen (anstatt des ursprünglichen Namensraumindex 0). Damit ist der Index umgesetzt.
8. Weisen Sie dem Parameter "NamespacesIndexCount" der Anweisung "OPC_UA_NodeGetIndexList" den Wert 1 zu (im Beispiel verwenden wir nur einen Namensraum).

Der folgende Programmcode zeigt eine Umsetzung des Beispiels:

SCL

```
#OPC_UA_NodeGetHandleList_Instance(NodeIDs := #TargetNodeIds,
                                   NodeStatusList := #NodeStatusList,
                                   NodeHdls := #NodeHdls,
                                   ConnectionHdl := #ConnectionHdl,
                                   Timeout := T#6S,
                                   NodeIDCount := 2,
                                   REQ := #Req,
                                   Busy => #Busy,
                                   Done => #Done,
                                   Error => #Error,
                                   Status => #Status,
                                   NamespaceIndexCount := 1,
                                   NamespaceIndexes := #myNamespaceIndexes);
```

Parameter "NamespaceIndexes" und "NamespaceIndexCount" sind optional

Die Parameter sind optional.

Daher müssen sie nicht versorgt werden, wenn keine Index-Umsetzung erforderlich ist.

Eine Index-Umsetzung ist nicht erforderlich, wenn Sie eine Client-Schnittstelle verwenden oder wenn Sie die NodeIds durch Browsen (Navigieren) auf dem Server ermittelt haben (mit der Anweisung "OPC_UA_TranslatePathList".)

Siehe auch

OPC_UA_TranslatePathList: Knotenparameter lesen (Seite 3831)

OPC UA-Anweisungen für Client-Programme (Seite 3797)

OPC_UA_TranslatePathList: Knotenparameter lesen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_TranslatePathList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Sie benötigen diese Anweisung, wenn Sie nur den symbolische Zugriffsweg kennen und die NodeIds dazu dynamisch ermitteln wollen. Dazu navigieren (browsen) Sie mit der Anweisung "OPC_UA_TranslatePathList" im Adressraum des OPC UA-Servers.

Sie benötigen diese Anweisung nicht, wenn Sie mit einer Client-Schnittstelle arbeiten.

Sie benötigen diese Anweisung auch nicht, wenn Sie die NodeIds der zu lesenden, schreibenden oder aufzurufenden OPC UA-Knoten kennen.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

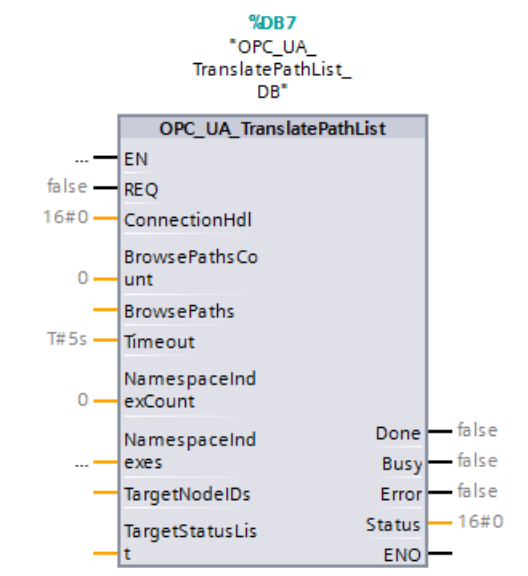

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

OPC_UA_TranslatePathList dient dazu, den Datenaustausch vorzubereiten, siehe ① in der folgenden Grafik.

Sie verwenden die Anweisung "OPC_UA_TranslatePathList", um vom Server die NodeIds von Knoten zu erfragen.

Dazu definieren Sie Browse-Pfade.

Die Anweisung liefert die NodeIds der gesuchten Knoten zurück.

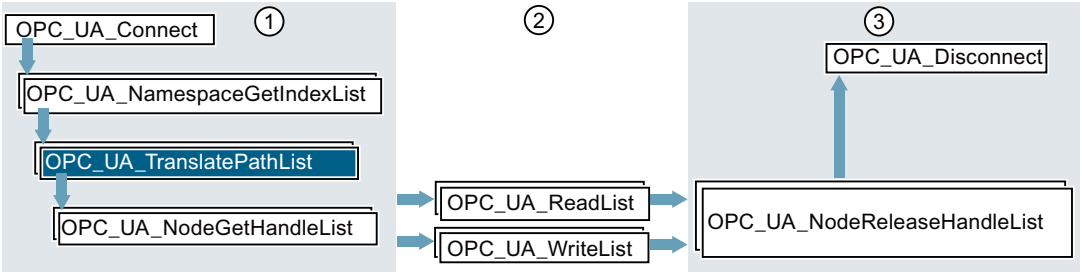

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_TranslatePathList"

Tabelle 4-75 Die Parameter der Anweisung "OPC_UA_TranslatePathList"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung OPC_UA_Connect.
BrowsePathsCount	Input	UINT	Anzahl der Elemente in BrowsePaths (max. 10).
BrowsePaths	InOut	VARIANT	Zeiger auf ein Array des Typs OPC_UA_BrowsePath. Das Array enthält die einzelnen Browse-Pfade, denen gefolgt werden soll.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
NamespaceIndexCount	Input	UINT	Anzahl der Elemente in NamespaceIndexes.
NamespaceIndexes	InOut	VARIANT	Zeiger auf ein Array des Typs UINT. Das Array dient zur Umsetzung der Indizes für die Namensräume. Beispiel: Wenn in NamespaceIndexes[4] der Wert 10 steht, dann wird allen NodeIds im Parameter "NodeIDs", die bislang den Namensraumindex 4 hatten, der Namensraumindex 10 zugeordnet.
TargetNodeIDs	InOut	VARIANT	Zeiger auf ein Array des Typs OPC_UA_NodeId. Das Array enthält die ermittelten Knoten-IDs (NodeIds).

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
TargetStatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlernummern für die einzelnen Browse-Pfade. Das erste Element des Arrays bezieht auf das erste Element im Parameter BrowsePaths, usw., siehe unten "Fehlernummern für TargetStatusList"
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Parameter "NamespacesIndexes"

Der Parameter "NamespaceIndexes" ist optional.

Der Parameter wird nur gebraucht, wenn man in den BrowsePaths in "OPC_UA_RelativePathElement" in ReferenceTypeId eine NodeId aus einem Namensraum ungleich CORE (also Index ungleich 0) verwenden möchte und diese NodeId offline bekommen hat.

Eine Beschreibung des Parameters finden Sie im Kapitel "OPC_UA_NodeGetHandleList".

Der Programmcode unten nutzt diesen Parameter nicht: Der Parameter "NamespaceIndexCount" wird auf 0 gesetzt. Dadurch wertet die Anweisung "OPC_UA_TranslatePathList" den Parameter "NamespaceIndexes" nicht aus.

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8002_0000	OpcUa_BadInternalError	Ein Fehler ist aufgetreten aufgrund eines internen Programmierfehlers des Funktionsbausteins. Bitte wenden Sie sich an den Support.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA-Client verfügbar. Da sich OPC UA-Client und OPC UA-Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Server antwortete nicht mit der erwarteten Anzahl an Ergebnissen
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect)
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	BrowsePathscount > MAX_ELEMENTS_RELATIVEPATH
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Parameter "BrowsePathsCount" ist nicht versorgt oder "BrowsePathsCount" = 0.
B080_C600	Simatic_ClientNotAvailable	Fehler beim Initialisieren des Clients
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "TargetStatusList"

Der Parameter "TargetStatusList" enthält für jede einzelne NodeId (Variable) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Wert erfolgreich gelesen.
8033_0000	OpcUa_BadNodeIdInvalid	Die Syntax der NodeId ist nicht korrekt.
8034_0000	OpcUa_BadNodeUnknown	Das übergebene Node-Handle ist nicht bekannt.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
8035_0000	OpcUa_BadAttributeInvalid	Das geforderte Attribut wird für den angegebenen Knoten nicht unterstützt.
8037_0000	OpcUa_BadIndexRangeNoD	Es gibt keine Daten in den Index-Bereich.
8039_0000	OpcUa_BadDataEncodingUnsupported	Der OPC UA-Server unterstützt für diesen Knoten nicht die geforderte Daten-Dekodierung.
803A_0000	OpcUa_BadNotReadable	Fehlende Rechte zum Lesen oder Abonnieren (Subscribing) dieses Knotens.
803C_0000	OpcUa_BadOutOfRange	Der Index-Wert, angegeben im Parameter NodeAddInfos, ist außerhalb des zulässigen Bereichs.
803D_0000	OpcUa_BadNotSupported	Der OPC UA-Server unterstützt eine der angeforderten Funktionen nicht. Manche OPC UA-Server erlauben keinen Zugriff auf Index-Bereiche eines Arrays.
8060_0000	OpcUa_BadBrowseNameInvalid	Der NamespaceIndex des QualifiedNames existiert nicht
806F_0000	OpcUa_BadNoMatch	Der angegebene Pfad führt zu keinem Knoten.
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Beispiels, wie Sie die Anweisung "OPC_UA_TranslatePathList" in einem Programm verwenden.

Beispiel

Angenommen, in einer Anlage erfassen RFID-Reader an den Toren die ausgehenden Waren.

Im Beispiel heisst der erste Reader "RfidReader_Door_1".

Die RFID-Reader sind OPC UA-Server und stellen ihre Daten und Methoden für OPC UA-Clients zur Verfügung.

Spezifikation AutoID

Die RFID-Reader sind mit einer standardisierten Schnittstelle gemäß der Companion Specification "AutoID" ausgestattet.

Durch die Standardisierung sind die Namen der Eigenschaften und Methoden der RFID-Reader festgelegt.

Das folgende Bild zeigt einen Ausschnitt aus dem Adressraum des Readers "FfidReaderDoor_1" in UAExpert, einem Tool von Unified Automation.

Unterhalb von "DeviceSet" befindet sich der Knoten "RfidReader_Door_1".

Darunter wiederum stehen die freigegebenen Eigenschaften und Methoden des Readers gemäß der AutoId:

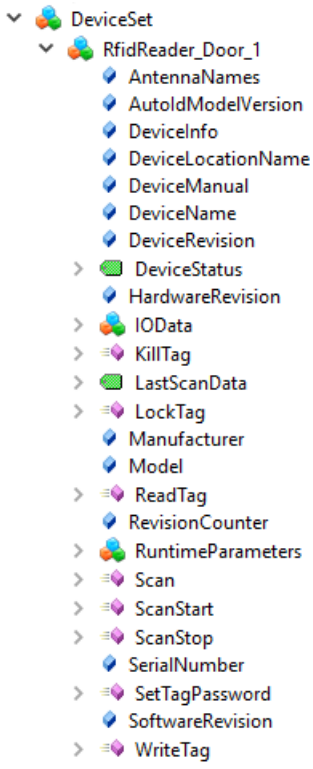

Die Eigenschaften und Methoden des Readers sind bekannt, weil der Reader die Spezifikation AutoID einhält.

Nicht bekannt sind folgende Elemente:

- Der Name "RfidReader_Door_1". Er ist frei wählbar, bzw. in der jeweiligen Anlage vorgegeben.
- Die NodeId des Knotens "RfidReader_Door_1" im Adressraum des Readers
- Die NodeIds der Eigenschaften und Methoden des Readers, d. h. aller Knoten unterhalb von "RfidReader_Door_1", siehe Bild oben.

Um die NodeIds zu ermitteln, gibt es die folgenden Vorgehensweisen:

- Sie nutzen ein Tool wie UAExpert und lesen die NodeId von "RfidReader_Door_1" sowie die NodeIds aller unterlagerten Eigenschaften und Methoden dieses Readers aus, die Sie in Ihrem Anwenderprogramm nutzen wollen.
  Der nächste Schritt in Ihrem Programm ist dann der Aufruf von "OPC_UA_NodeGetHandleList".
- Sie arbeiten mit der Anweisung "OPC_UA_TranslatePathList".
  Ausgehend vom Knoten "DeviceSet", dessen NodeId bekannt ist, definieren Sie in Ihrem Anwenderprogramm einen Pfad (BrowsePath).
  Anschließend ermitteln Sie mit Anweisung "OPC_UA_TranslatePathList" die NodeIds aller unterlagerten Eigenschaften und Methoden des Readers, die Sie in Ihrem Anwenderprogramm nutzen.
  Der nächste Schritt ist dann der Aufruf von "OPC_UA_NodeGetHandleList".
  Diese Vorgehensweise zeigt das folgende Kapitel.

Voraussetzung

Für die Anweisung "OPC_UA_TranslatePathList" muss die folgende Voraussetzungen erfüllt sein:

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle von der Anweisung OPC_UA_Connect.

Aufruf der Anweisung (Erstaufruf)

Dieses Kapitel beschreibt den Funktionsbaustein "Read_from_RfidReader_Door_1". Der Funktionsbaustein zeigt, wie Sie die Anweisung "OPC_UA_TranslatePathList" verwenden können.

Das vollständige Beispiel finden Sie im Kapitel "AUTOHOTSPOT".

Deklaration der Variablen

Deklarieren Sie eine Instanz der Anweisung "OPC_UA_TranslatePathList" sowie die Variablen, mit denen Sie die Parameter der Anweisung versorgen.

Das Programmbeispiel verwendet die folgende Deklaration:

OPC_UA_TranslatePathList_Instance	OPC_UA_TranslatePathList
BrowsePaths	Array[0..1] of OPC_UA_BrowsePath
TagetNodeIds	Array[0..1] of OPC_UA_NodeId
TargetStatusList	Array[0..1] of DWord

Anwenderprogramm

Der folgende Auszug aus dem Anwenderprogramm definiert den Browse-Pfad für die PLC-Variable "DeviceInfo". Die Definition für "DeviceStatus" ist nicht dargestellt.

Zuerst werden die Startknoten (StartingNode) definiert, von dem aus der Browse-Pfad zum gesuchten Knoten führt. Nachfolgend wird der Pfad zum gesuchten Knoten (RelativePath) definiert.

SCL

```
IF #FirstCall = TRUE THEN
                //set BrowsePath[0] for DeviceInfo
                //set BrowsePath.StartingNode
                #BrowsePaths[0].StartingNode.Identifier := WString#'85';
                #BrowsePaths[0].StartingNode.IdentifierType := 0;
                #BrowsePaths[0].StartingNode.NamespaceIndex := 0;

                //set BrowsePath.RelativePath
                #BrowsePaths[0].RelativePath.NoOfElements := 4;
                #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.Identifier :=
WString#'33';
               #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.IdentifierType :=
0;
               #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[0].RelativePath.Elements[1].TargetName.Name :=
WString#'productionline';
                #BrowsePaths[0].RelativePath.Elements[1].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.Identifier :=
WString#'33';
               #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.IdentifierType :=
0;
               #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[0].RelativePath.Elements[2].TargetName.Name :=
WString#'DataBlocksGlobal';
                #BrowsePaths[0].RelativePath.Elements[2].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.Identifier :=
WString#'33';
               #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.IdentifierType :=
0;
               #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[0].RelativePath.Elements[3].TargetName.Name :=
WString#'RfidReader_Door_1';
                #BrowsePaths[0].RelativePath.Elements[3].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.Identifier :=
WString#'33';
               #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.IdentifierType :=
0;
               #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[0].RelativePath.Elements[4].TargetName.Name :=
WString#'DeviceInfo';
                #BrowsePaths[0].RelativePath.Elements[4].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];
```

Programmbeispiel Teil 2

Der folgende Ausschnitt aus dem Programmbeispiel konfiguriert die Verbindung zu einem OPC UA-Server.

Für die Variable "#ServerEndpointUrl" wird der ServerEndpoint gesetzt. Im Beispiel ist der ServerEndpoint "opc.tcp://192.168.1.1:4840", weil das Protokoll "opc.tcp" bei OPC UA genutzt wird und der Server die IP-Adresse "192.168.1.1" besitzt.

Der Security-Mode "2" (SecurityMsgMode) wird gesetzt, sodass Nachrichten signiert übertragen werden. Als Verfahren für die Signierung (SecurityPolicy) wird der Wert "Basic128Rsa15" festgelegt. Des Weiteren wird die Nummer des Client-Zertifikats (CerificateID) angegeben. In unserem Beispiel ist das die Nummer 6.

SCL

```
CASE #State OF
        1:  // case 1, connect to server


            IF #FirstCall = TRUE THEN
                #FirstCall := FALSE;

                //set ServerEndPointUrl of the server we want to connect to
                #SeverEndpointUrl := WString#'opc.tcp://192.168.1.1:4840';

                //set Securtiy Message Mode
                // 1 = None, 2 = Sign, 3 = Sign & Encrypt
                #SessionConnectInfo.SecurityMsgMode := 1;

                //set Security Policy
                // 1 = None, 2 = Basic128Rsa15,, 3 = Basic256, 4 = Basic256Sha256
                #SessionConnectInfo.SecurityPolicy := 2;

                //set the number of the client certificate you are using
                //in our case, this is 6
                //#SessionConnectInfo.CertificateID := 6;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Connect_Instance(REQ:=#Req,
                                     ServerEndpointUrl:=#SeverEndpointUrl,
                                     SessionConnectInfo:=#SessionConnectInfo,
                                     Timeout:=T#8S,
                                     Done=>#Done,
                                     Busy=>#Busy,
                                     Error=>#Error,
                                     Status=>#Status,
                                     ConnectionHdl=>#ConnectionHdl);
```

Programmbeispiel Teil 3

Der folgende Ausschnitt aus dem Beispielprogramm "Read_From_RfidReader_Door_1" erfragt beim OPC UA-Server die NodeIds der Knoten mit BrowseName "DeviceInfo" und "DeviceStatus".

Das vollständige Beispiel finden Sie im Kapitel "AUTOHOTSPOT".

SCL

```
IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_TranslatePathList_Instance(BrowsePaths := #BrowsePaths,
                                               TargetNodeIDs := #TagetNodeIds,
                                               TargetStatusList := #TargetStatusList,
                                               ConnectionHdl := #ConnectionHdl,
                                               NamespaceIndexCount := 0,
                                               BrowsePathsCount := 2,
                                               Timeout := T#6S,
                                               REQ := #Req,
                                               Busy => #Busy,
                                               Error => #Error,
                                               Done => #Done,
                                               Status => #Status);
            IF #Done = TRUE THEN
                IF #TargetStatusList[0] = 0 AND #TargetStatusList[1] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at TranslatePathList';
                END_IF;
                #FirstCall := TRUE;
            END_IF;
            IF #Error = TRUE THEN
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #State := 100;
                #Mem_Status := #Status;
                #OPC_UA_TranslatePathList_Instance(BrowsePaths := #BrowsePaths,
                                                   TargetNodeIDs := #TagetNodeIds,
                                                   REQ := FALSE,
                                                   TargetStatusList := #TargetStatusList,
                                                   ConnectionHdl := #ConnectionHdl);
                #FirstCall := TRUE;
            END_IF;
```

Das Beispielprogramm ruft die Anweisung "OPC_UA_TranslatePathList" aus den folgenden Gründen auf:

- Um die NodeId für die Knoten mit BrowseName "DeviceInfo" und "DeviceStatus" zu erfragen.
- Um den Parameter REQ auf FALSE zu setzen, falls ein Fehler aufgetreten ist.

Asynchrone Ausführung

Die Anweisung "OPC_UA_TranslatePathList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Programmbeispiels Teil 3

Im Bild oben ist ein Ausschnitt aus dem Funktionsbaustein "Read_From_RfidReader_Door_1" dargestellt.

Wenn die Anweisung "OPC_UA_TranslatePathList" noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req " auf TRUE gesetzt wird. Dadurch startet die Anweisung.

Im nächsten Zyklus besitzt #Req den Wert FALSE.

Wenn im Ausgangsparameter "Error" der Wert TRUE steht, dann trat bei der Ausführung der Anweisung ein Fehler auf. Dadurch wird die Variable "#State"auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt.

Achtung: Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber für eine PLC-Variable keine NodeId zurückgeliefert werden konnte.

Deshalb wird im Beispiel geprüft, ob das Programm für jede PLC-Variable eine NodeId erhalten hat:

- Kommt für die 1. Variable eine NodeId zurück, steht im 1. Element des Arrays, auf das der Parameter "TargetStatusList" zeigt, der Wert 0.
- Kommt für die 2. Variable eine NodeId zurück, steht im 2. Element des Arrays, auf das der Parameter "TargeteStatusList" zeigt, der Wert 0.

Wenn beide Bedingungen erfüllt sind, dann zeigt der Ausgangsparameter "TargetNodeHdls" auf eine gültige Liste mit den NodeIds. Dadurch wird der Wert der Variable "#State" um eins erhöht. Nachfolgend wird im nächsten Zyklus der nächste Programmabschnitt - Fall 4 - ausgeführt.

Fall 4 ist in OPC_UA_NodeGetHandleList beschrieben.

Datenaustausch

OPC_UA_ReadList: Variablen lesen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_ReadList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_ReadList" lesen Sie die Werte von PLC-Variablen.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

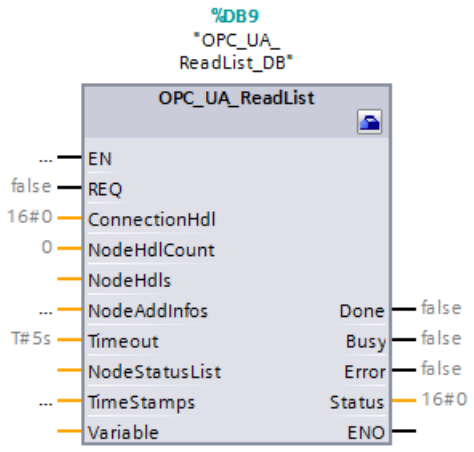

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_ReadList" dient zum Lesen von Variablen von einem OPC UA-Server, siehe ② in der folgenden Grafik.

Die Anweisung liefert eine Liste mit den Werten von Variablen zurück.

Neben den Werten von Variablen können auch andere Attribute der Variablen gelesen werden. Dies wird über die AttributeID in NodeAdditionalInfo vorgegeben.

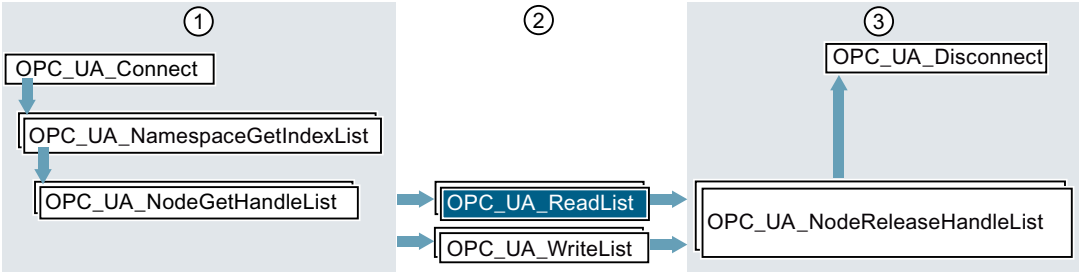

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_ReadList"

Tabelle 4-76 Die Parameter der Anweisung "OPC_UA_ReadList"

Parameter	Deklaration	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung AUTOHOTSPOT.
NodeHdlCount	Input	UINT	Anzahl der Elemente in dem Array, auf das der Parameter NodeHdls zeigt.
NodeHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Knoten-Handles (Node-Handlles) der Variablen, deren Werte gelesen werden sollen.
NodeAddInfos	InOut	VARIANT	Zeiger auf ein Array des Typs AUTOHOTSPOT oder AUTOHOTSPOT. Das Array bestimmt, welches Attribut jeweils im Knoten (in einer Variablen) gelesen werden soll. Das erste Element dieses Arrays bezieht sich auf das erste Element im Array, auf den der Parameter NodeHdls zeigt. Dieser Parameter ist optional. Fehlt er, so wird von allen Knoten (Variablen) der Wert gelesen. Beim Lesen von Arrays kann eingeschränkt werden, welche Elemente des Arrays gelesen werden sollen, siehe Array-Bereich lesen mit OPC_UA_ReadList (Seite 3854).

Parameter	Deklaration	Datentyp	Bedeutung
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter beiAUTOHOTSPOT
NodeStatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die einzelnen Variablen (siehe unten "Fehlernummern für NodeStatusList"). Für jede Variable ist angegeben, ob ihr Wert gelesen werden konnte Es kann durchaus sein, dass die Anweisung erfolgreich ausgeführt wurde (Parameter Error nicht gesetzt), aber für eine bestimmte NodeId (Variable) kein Wert gelesen werden konnte.
TimeStamps	InOut	VARIANT	Zeiger auf ein Array des Typs LDT. Das erste Element dieses Arrays bezieht auf das erste Element des Arrays, auf das der Parameter NodeHdls zeigt. Der Parameter ist optional. Fehlt er, dann liefert der OPC UA-Server keine Zeitstempel.
Variable	InOut	VARIANT	Zeiger auf eine Variable, die die zu lesenden Werte aufnimmt. Für diese Variable müssen Sie einen PLC-Datentyp (UDT) oder eine Struktur (STRUCT) anlegen.
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Funktionsbaustein erfolgreich ausgeführt.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Server antwortete nicht mit der erwarteten Anzahl an Ergebnissen
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
8074_0000	OpcUa_BadTypeMismatch	Sie verwenden einen Datentyp, der nicht zum Datentyp im Server passt.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.
B080_0500	Simatic_BadType_VariantInput5	Der Datentyp der Variablen stimmt nicht, auf die der 5. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	NodeHdlCount > MAX_ELEMENTS_NODELIST NamespaceIndexCount > MAX_ELEMENTS_NAMESPACES
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT.		

Fehlernummern für "NodeStatusList"

Der Parameter "NodeStatusList" enthält für jede einzelne Node-Handle (Variable) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Wert erfolgreich gelesen.
8034_0000	OpcUa_BadNodeIdUnknown	Das übergebene Node-Handle ist nicht bekannt.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
8035_0000	OpcUa_BadAttributeInvalid	Das geforderte Attribut wird für den angegebenen Knoten nicht unterstützt.
8037_0000	OpcUa_BadIndexRangeNoD	Es gibt keine Daten im Index-Bereich.
8039_0000	OpcUa_BadDataEncodingUnsupported	Der OPC UA-Server unterstützt für diesen Knoten nicht die geforderte Daten-Dekodierung.
8074_0000	OpcUa_BadTypeMismatch	Double / Float Wert in ReadResponse ist NaN Variable UDT Member haben falschen Typ
803A_0000	OpcUa_BadNotReadable	Fehlende Rechte zum Lesen oder Abonnieren (Subscribing) dieses Knotens.
803C_0000	OpcUa_BadOutOfRange	Der Index-Wert, angegeben im Parameter NodeAddInfos, ist außerhalb des zulässigen Bereichs.
803D_0000	OpcUa_BadNotSupported	Der OPC UA-Server unterstützt eine der angeforderten Funktionen nicht. Manche OPC UA-Server erlauben keinen Zugriff auf Index-Bereiche eines Arrays.
80AB_0000	OpcUa_BadInvalidArgument	StartIndex > EndIndex
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_ReadList" in einem Anwenderprogramm verwenden, das die Werte von PLC-Variablen liest.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben ein Client-Interface angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "Verbindungen anlegen und parametrieren".

Zudem müssen für die Anweisung "OPC_UA_ReadList" die folgenden Voraussetzungen erfüllt sein:

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle von der Anweisung AUTOHOTSPOT.
- Indizes der Namenräume, in denen die Variablen liegen, die Sie mit dem Client lesen wollen.
  Sie erhalten die Indizes von der Anweisung AUTOHOTSPOT
- Eine Liste mit den Handles für die einzelnen Variablen, deren Werte Sie vom OPC UA-Server lesen wollen.
  Sie erhalten diese Handle-Liste von der Anweisung AUTOHOTSPOT

Funktion der Anweisung

Sie verwenden die Anweisung "OPC_UA_ReadList", um z. B. aus einem OPC UA-Server die Werte von Variablen zu lesen (siehe oben).

Welche Informationen sind erforderlich?

Die Anweisung benötigt die folgende Angabe:

- Die Handles für die Variablen, deren Werte Sie lesen wollen (im Parameter "NodeHdls").

Welche Informationen gibt die Anweisung zurück?

Die Anweisung liefert die folgenden Angaben:

- Die aktuellen Werten der Variablen (im Parameter "Variable").
  Die Reihenfolge der Werte entspricht der Reihenfolge der Handles im Parameter "NodeHdls".
  Beachten Sie hierzu die folgenden Fälle:
  - Verwenden Sie eine Client-Schnittstelle mit einer Leseliste, legt STEP 7 automatisch einen Systemdatentyp mit den zu lesenden Variablen an.
  - Verwenden Sie keine Client-Schnittstelle, müssen Sie einen PLC-Datentyp (UDT) oder Struktur (STRUCT) für den Parameter "Variable" anlegen.
    Die Komponenten dieses UDTs definieren Sie entsprechend der zu lesenden Variablen.
    Verwenden Sie den kompatiblen SIMATIC-Datentyp zum jeweiligen OPC UA-Datentyp, siehe AUTOHOTSPOT.
- Eine Liste mit Fehlermeldungen (im Parameter "NodeStatusList").
  Jede Fehlermeldung in dieser Liste bezieht sich auf das entsprechende Handle im Parameter "NodeHdls", Beispiel: NodeStatusList[0] bezieht sich auf NodeHdls[0].
  Sie müssen für jede einzelne Variable (jedes einzelne "NodeHdl") prüfen, ob der OPC UA-Server einen gültigen Wert zurückliefern konnte.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung enthalten soll.
   Im Beispiel heisst der Funktionsbaustein "ReadFromProductionline".
   Gewählte Sprache: SCL.

3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_ReadList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_ReadList_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_ReadList_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline".
7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Leseliste.
   Im Beispiel heisst die Leseliste "ReadListProduct".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.
8. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Das vollständige Programm finden Sie im Kapitel "AUTOHOTSPOT"

Das Beispielprogramm verwendet die Leseliste "ReadListProduct", die die folgenden PLC-Variablen enthält:

- NewProduct
- ProductNumber

Die Werte dieser Variablen liest das Programm vom Server.

Das Kapitel "AUTOHOTSPOT" zeigt, wie Sie eine Client-Schnittstelle anlegen und ihr eine Leseliste mit PLC-Variablen hinzufügen.

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Im 4. Programmabschnitt werden die Werte von Variablen gelesen:

SCL

```
4:  // case 4, read from nodes

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_ReadList_Instance(
                                    NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                    NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                    Variable :=
"Productionline_Data"."ReadListProduct".Variable,
                                    ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                    Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                    NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                    TimeStamps :=
"Productionline_Data"."ReadListProduct".TimeStamps,
                                    REQ := #Req,
                                    Done => #Done,
                                    Busy => #Busy,
                                    Error => #Error,
                                    Status => #Status);
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_ReadList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 4 dargestellt.

Dieser Programmteil ruft die Anweisung "OPC_UA_ReadList" auf, um die Werte von PLC-Variablen zu lesen.

Wenn die Anweisung noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
            IF #Done = TRUE THEN
                FOR #i := 0 TO
UINT_TO_INT("Productionline_Configuration".ReadLists."ReadListProduct".NodeCount) - 1 DO
                  IF NOT ("Productionline_Data"."ReadListProduct".NodeStatusList[#i] = 0)
THEN
                       #Output_Error_Message := CONCAT_WSTRING(IN1 := WSTRING#'Error at
Readlist "ReadListProduct", Index: ',
                                                                    IN2 := INT_TO_WSTRING(#i));
                   END_IF;
                END_FOR;
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_ReadList_Instance(
                                          NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                          NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                          Variable :=
"Productionline_Data"."ReadListProduct".Variable,
                                          REQ := FALSE,
                                          ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                          Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                          NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                          TimeStamps :=
"Productionline_Data"."ReadListProduct".TimeStamps);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall wird der Ausgangsparameter "Error" auf den Wert TRUE gesetzt. Dadurch wird der Wert der Variable "#State" auf 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen. Zudem ruft das Beispielprogramm die Anweisung "OPC_UA_ReadList" auf, um den Parameter REQ auf FALSE zu setzen.

Steht im Ausgangsparameter "Done" der Wert TRUE, wird die Anweisung erfolgreich ausgeführt.

Achtung: Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber eine PLC-Variable nicht gelesen werden konnte.

Deshalb wird im Beispiel für jede PLC-Variable geprüft, ob das Programm einen gültigen Wert erhalten hat:

- Wenn der Server für die 1. PLC-Varialbe einen gültigen Wert sendet, dann steht im 1. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
  Im Beispiel steht der Statuscode in "Productionline_Data"."ReadListProduct".NodeStatusList[0].
- Wenn der Server für die 2. PLC-Varialbe einen gültigen Wert sendet, dann steht im 2. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
  Im Beispiel steht der Statuscode in "Productionline_Data"."ReadListProduct".NodeStatusList[1].

Das Programm gibt eine Fehlermeldung aus, wenn ein ungültiger Wert erkannt wurde.

Auch wenn ein oder zwei ungültige Werte erkannt wurden, wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 5 - ausgeführt.

Fall 5 ist in AUTOHOTSPOT beschrieben.

Siehe auch

OPC_UA_NodeReleaseHandleList: Handles für Lese- und Schreibzugriffe freigeben (Seite 3875)

OPC_UA_TranslatePathList: Knotenparameter lesen (Seite 3831)

OPC UA-Anweisungen für Client-Programme (Seite 3797)

Array-Bereich lesen mit OPC_UA_ReadList

Einleitung

Dieses Kapitel basiert auf dem Kapitel "OPC_UA_ReadList", siehe OPC_UA_ReadList (Seite 3844).

Das Kapitel beschreibt, wie Sie mit der Anweisung "OPC_UA_ReadList" einen Teilbereich eines Arrays lesen.

Prinzipiell funktioniert das Lesen eines Array-Teilbereichs von einem OPC UA-Server folgendermaßen:

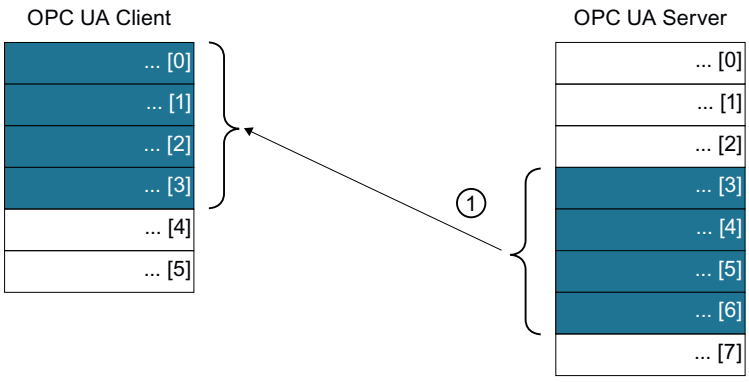

① Lesen eines Array-Teilbereichs von einem OPC UA-Server

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine CPU als **OPC UA-Server** konfiguriert.
  Der Server stellt ein Array bereit, das von OPC UA-Clients gelesen werden kann.
  Im Beispiel heißt die CPU "Productionline", die für OPC UA-Clients den Datenbaustein "Data_for_OPC_UA_Clients" bereitstellt:

Static	
NewProduct	Bool
ProductNumber	Int
Temperature	Array[0..9] of Real
Temperature[0]	Real
Temperature[1]	Real
Temperature[2]	Real
Temperature[3]	Real
Temperature[4]	Real
Temperature[5]	Real
Temperature[6]	Real
Temperature[7]	Real
Temperature[8]	Real
Temperature[9]	Real

- Sie haben eine CPU als **OPC UA-Client** konfiguriert.
  Sie haben im Client eine Client-Schnittstelle angelegt.
  Die Client-Schnittstelle enthält eine Leseliste mit einem Array, das Sie teilweise auslesen wollen.
  Im Beispiel heißt die Client-Schnittstelle "Productionline" mit der Leseliste "ReadListProduct":

NewProduct	BOOL
ProductNumber	INT
Temperature	ARRAY[0..9] of REAL

  Aus dem Array "Temperature" des Servers werden die Werte in "Temperature[5]" bis "Temperature[9]" gelesen.

Zusätzliche Informationen bereitstellen

Das Kapitel "OPC_UA_ReadList" zeigt, wie Sie mit der Anweisung "OPC_UA_ReadList" Skalare (Variablen mit einem Wert) oder Arrays (Variablen mit mehreren Werten) im Ganzen auslesen.

Im folgenden Abschnitt sind die Erweiterungen zum Kapitel "OPC_UA_ReadList" dargestellt, die notwendig sind, um mit "OPC_UA_ReadList" einen Teilbereich eines Arrays zu lesen.

OPC_UA_NodeAdditionalInfo

Zunächst legen Sie eine lokale Variable des Typs "OPC_UA_NodeAdditionalInfo (Seite 3927)" an.

Diese Variable dient dazu, der Anweisung "OPC_UA_ReadList" mitzuteilen, welcher Array-Bereich gelesen wird.

Gehen Sie folgendermaßen vor:

1. Legen Sie eine Array-Variable des Typs "OPC_UA_NodeAdditionalInfo" an.
   Im Beispiel wird die Variable im Deklarationsteil des Funktionsbausteins angelegt, der die Anweisung "OPC_UA_ReadList" aufruft.
   Die neue lokale Variable heißt "NodeAdditionalInformation":

NodeAdditonalInformation	Array[0..2] of OPC_UA_NodeAdditionalInfo
NodeAdditonalInformation[0]	OPC_UA_NodeAdditionalInfo
AttributeID	UDInt
StartIndex	UDInt
EndIndex	UDInt
NodeAdditonalInformation[1]	OPC_UA_NodeAdditionalInfo
AttributeID	UDInt
StartIndex	UDInt
EndIndex	UDInt
NodeAdditonalInformation[2]	OPC_UA_NodeAdditionalInfo
AttributeID	UDInt
StartIndex	UDInt
EndIndex	UDInt

   Die neue lokale Variable "NodeAdditionalInformation" enthält drei Komponenten des Typs OPC_UA_AdditionalInfo, weil in der Leseliste "ReadListProduct" drei Knoten enthalten sind.
   NodeAdditionalInformation[0] stellt zusätzliche Informationen für die erste zu lesende Variable bereit (im Beispiel "NewProduct")
   NodeAdditionalInformation[1] stellt zusätzliche Informationen für die zweite zu lesende Variable bereit (im Beispiel "ProductNumber")
   NodeAdditionalInformation[2] stellt zusätzliche Informationen für die dritte zu lesende Variable bereit (im Beispiel "Temperature")

2. Weisen Sie der neuen Variablen Werte zu (in der Spalte "Defaultwerte" des Deklarationsteils).
   Im Beispiel wird die dritte Variable (Temperature) teilweise gelesen.
   Deshalb wird NodeAdditionalInformation[2].AttributeId 13 zugewiesen (d. h. die Werte des Arrays werden gelesen).
   NodeAdditionalInformation[2].StartIndex erhält den Wert 5, da ab Index 5 (einschließlich) gelesen wird.
   NodeAdditionalInformation[2].EndIndex erhält den Wert 9, da bis Index 9 (einschließlich) gelesen wird.
   NodeAdditionalInformation[0] und NodeAdditionalInformation[1] werden Standardwerte gesetzt.
   Die Konstante "#ignore" enthält den Wert 4_294_967_295. Der Server ignoriert diesen Wert für StartIndex und EndIndex und liest Skalare und Arrays vollständig.

NodeAdditonalInformation	Array[0..2] of OPC_UA_NodeAdditionalInfo	
NodeAdditonalInformation[0]	OPC_UA_NodeAdditionalInfo	
AttributeID	UDInt	13
StartIndex	UDInt	#ignore
EndIndex	UDInt	#ignore
NodeAdditonalInformation[1]	OPC_UA_NodeAdditionalInfo	
AttributeID	UDInt	13
StartIndex	UDInt	#ignore
EndIndex	UDInt	#ignore
NodeAdditonalInformation[2]	OPC_UA_NodeAdditionalInfo	
AttributeID	UDInt	13
StartIndex	UDInt	5
EndIndex	UDInt	9

Alternative:
Sie können die Werte für NodeAdditionalInformation auch im Anwenderprogramm setzen:

```
                // set additonal information to Array-Index 0
                // we set default values here
                #NodeAdditionalInformation[0].AttributeID := 13;
                #NodeAdditionalInformation[0].StartIndex :=
4_294_967_295;
                #NodeAdditionalInformation[0].EndIndex :=
4_294_967_295;

                // set additonal information to Array-Index 1
                // we set default values here
                #NodeAdditonalInformation[1].AttributeID := 13;
                #NodeAdditonalInformation[1].StartIndex :=
4_294_967_295;
                #NodeAdditonalInformation[1].EndIndex :=
4_294_967_295;

                // set additonal information to Array-Index 2
                // because we want to read only five values from
an array in the OPC UA server
                #NodeAdditonalInformation[2].AttributeID := 13;
                #NodeAdditonalInformation[2].StartIndex := 5;
                #NodeAdditonalInformation[2].EndIndex := 9;
```

Hinweis:
Das Array NodeAdditionalInformation des Typs "OPC_UA_NodeAdditionalInfo" muss so viele Komponenten enthalten wie die Leseliste (im Beispiel 3).

3. Weisen Sie NodeAdditionalInformation dem Parameter "NodeAddInfos" der Anweisung "OPC_UA_ReadList" zu.
   Das folgende Bild zeigt den Aufruf der OPC_UA_ReadList mit der neuen Versorgung des Parameters "NodeAddInfos":

```
     #OPC_UA_ReadList_Instance(
                              NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdl
s,
                              NodeStatusList :=
"myProductionline_Data"."ReadListProduct".NodeStatusList,
```

```
                                Variable :=
"myProductionline_Data"."ReadListProduct".Variable,
                                ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeo
ut,
                                NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCou
nt,
                                NodeAddInfos :=
#NodeAddtionalInformation,
                                TimeStamps :=
"myProductionline_Data".ReadListProduct.TimeStamps,
                                REQ := #Req,
                                Done => #Done,
                                Busy => #Busy,
                                Error => #Error,
                                Status => #Status);
```

Ergebnis im Client

Die Anweisung "OP_UA_ReadList" liest jetzt nur einen Teilbereich des Arrays "Temperature" aus dem OPC UA-Server.

Das folgende Bild zeigt das Array "Temperature". Der Client weist die gelesenen Werte ab Index 0 zu:

Temperature	Array[0..9] of Real		
Temperature[0]	Real	0.0	206.6
Temperature[1]	Real	0.0	207.7
Temperature[2]	Real	0.0	208.8
Temperature[3]	Real	0.0	209.9
Temperature[4]	Real	0.0	210.0
Temperature[5]	Real	0.0	0.0
Temperature[6]	Real	0.0	0.0
Temperature[7]	Real	0.0	0.0
Temperature[8]	Real	0.0	0.0
Temperature[9]	Real	0.0	0.0

Hinweis

Um einen Teilbereich eines mehrdimensionalen Arrays zu lesen (bis zu sechs Dimensionen), verwenden Sie den Datentyp "OPC_UA_NodeAdditionalInfoExt".

OPC_UA_WriteList:Variablen schreiben

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_WriteList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_WriteList" schreiben Sie neue Werte in PLC-Variablen.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

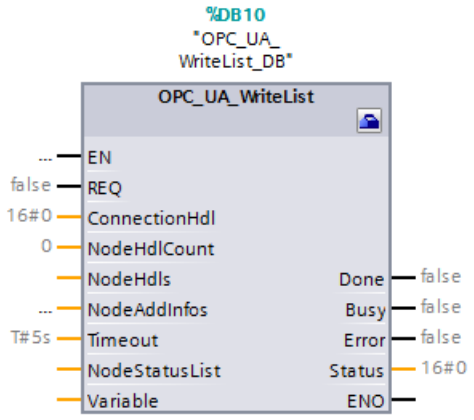

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_WriteList" dient dazu, PLC-Variablen neue Werte zuzuweisen, siehe ② in der folgenden Grafik:

Die Anweisung liefert eine Liste mit Statusangaben für jede PLC-Variable zurück.

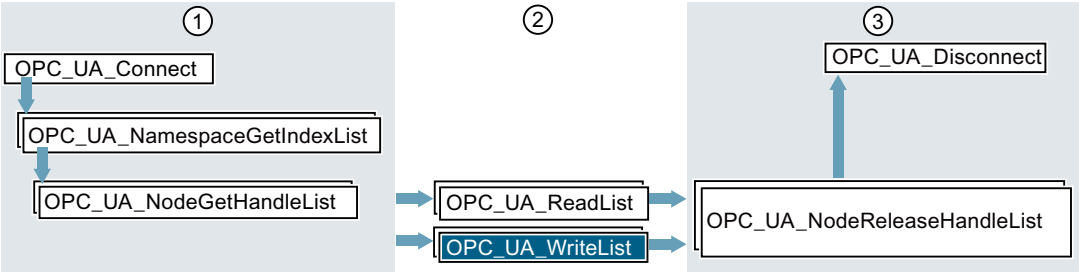

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für OPC_UA_WriteList

Tabelle 4-77 Die Parameter der Anweisung OPC_UA_WriteList

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung AUTOHOTSPOT.
NodeHdlCount	Input	UINT	Anzahl der Elemente in dem Array, auf das der Parameter NodeHdls zeigt.
NodeHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Knoten-Handles (Node-Handles) der Variablen, deren Werte geschrieben werden sollen.
NodeAddInfos	InOut	VARIANT	Zeiger auf ein Array des Typs AUTOHOTSPOT oder AUTOHOTSPOT. Das Array bestimmt, welches Attribut jeweils in einem Knoten (einer Variablen) gesetzt werden soll. Das erste Element dieses Arrays bezieht sich auf das erste Element im Array, auf den der Parameter NodeHdls zeigt. Dieser Parameter ist optional. Fehlt er, so wird in allen Knoten (Variablen) der Wert gesetzt. Beim Schreiben von Arrays kann eingeschränkt werden, welche Elemente des Arrays geschrieben werden sollen, siehe Array-Bereich schreiben mit OPC_UA_WriteList (Seite 3869).
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
NodeStatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die einzelnen Variablen, siehe unten "Fehlernummern für NodeStatusList". Für jede Variable ist angegeben, ob ihr Wert gesetzt werden konnte. Es kann durchaus sein, dass die Anweisung erfolgreich ausgeführt wurde (Parameter Error nicht gesetzt), aber einer bestimmten Variablen kein Wert zugewiesen werden konnte. NodeStatusList[0] bezieht sich auf NodeHdls[0], usw.
Variable	InOut	VARIANT	Zeiger auf eine Variable, die die zu schreibenden Werte enthält. Für diese Variable müssen Sie einen PLC-Datentyp (UDT) anlegen.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Funktionsbaustein erfolgreich ausgeführt.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Der Server sendet eine Antwort, die nicht erkannt werden kann.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	Übergreifender Fehlercode. Tritt auf, wenn ein Array zu viele Elemente hat.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "NodeStatusList"

Der Parameter "NodeStatusList" enthält für jedes einzelne Node-Handle (Variable) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Wert erfolgreich gelesen.
8034_0000	OpcUa_BadNodeUnknown	Das übergebene Node-Handle ist nicht bekannt.
8035_0000	OpcUa_BadAttributeInvalid	Das geforderte Attribut wird für den angegebenen Knoten nicht unterstützt.
8037_0000	OpcUa_BadIndexRangeNoD	Es gibt keine Daten in den Index-Bereich.
8039_0000	OpcUa_BadDataEncodingUnsupported	Der OPC UA-Server unterstützt für diesen Knoten nicht die geforderte Daten-Dekodierung.
803A_0000	OpcUa_BadNotReadable	Fehlende Rechte zum Lesen oder Abonnieren (Subscribing) dieses Knotens.
803C_0000	OpcUa_BadOutOfRange	Der Index-Wert, angegeben im Parameter NodeAddInfos, ist außerhalb des zulässigen Bereichs.
803D_0000	OpcUa_BadNotSupported	Der OPC UA-Server unterstützt eine der angeforderten Funktionen nicht. Manche OPC UA-Server erlauben keinen Zugriff auf Index-Bereiche eines Arrays.
80AB_0000	OpcUa_BadInvalidArgument	Ein Argument ist ungültig oder mehrere Argumente sind ungültig.
Weitere Fehlercodes sieheAUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_WriteList" in einem Anwenderprogramm verwenden, das neue Werte in PLC-Variablen setzt.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_WriteList" die folgenden Voraussetzungen erfüllt sein:

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle von der Anweisung AUTOHOTSPOT.
- Indizes der Namenräume, in denen die Variablen liegen, die Sie mit dem Client schreiben wollen.
  Sie erhalten die Indizes von der Anweisung AUTOHOTSPOT
- Eine Liste mit den Handles für die einzelnen Variablen, deren Werte Sie im OPC UA-Server setzen wollen.
  Sie erhalten diese Handle-Liste von der Anweisung AUTOHOTSPOT

Im Programmbeispiel unten wird die Anweisung einmal aufgerufen, um ein Freigabesignal an den OPC UA-Server zu übertragen.

Funktion des Bausteins

Sie verwenden die Anweisung "OPC_UA_WriteList", um die Werte von Variablen in einem OPC UA-Server zu schreiben.

Welche Informationen sind erforderlich?

Die Funktion benötigt die folgenden Angaben:

- Die Handles für die Variablen, deren Werte Sie schreiben wollen (Parameter "NodeHdls").
- Die zu übertragenden Werten für die einzelnen Variablen (Parameter "Variable").
  Beachten Sie hierzu die folgenden Fälle:
  - Verwenden Sie eine Client-Schnittstelle mit einer Schreibliste, legt STEP 7 automatisch einen Systemdatentyp mit den zu übertragenden Variablen an.
  - Verwenden Sie keine Client-Schnittstelle, müssen Sie einen PLC-Datentyp (UDT) oder eine Struktur (STRUCT) für den Parameter "Variable" anlegen.
    Die Komponenten dieses UDTs definieren Sie entsprechend der zu übertragenden Variablen.
    Verwenden Sie den kompatiblen SIMATIC-Datentyp zum jeweiligen OPC UA-Datentyp, siehe AUTOHOTSPOT.

Welche Informationen gibt die Anweisung zurück?

Die Anweisung liefert die folgende Angabe:

- Eine Liste mit Fehlermeldungen (Parameter "NodeStatusList").
  Jede Fehlermeldung in dieser Liste bezieht sich auf das entsprechende Handle im Parameter "NodeHdls".
  Sie müssen für jede einzelne Variable ("NodeHdl") prüfen, ob der OPC UA-Server den neuen Wert übernehmen konnte.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "WriteToProductionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_WriteList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_WriteList_Instance".
5. Klicken Sie auf "OPC_UA_WriteList_Instance".
   STEP 7 öffnet die "Konfiguration".
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", die wir für unsere AUTOHOTSPOT angelegt hatten, siehe "AUTOHOTSPOT".
7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Leseliste.
   Im Beispiel heisst die Leseliste "WriteListStatus".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben..
8. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Das vollständige Programm finden Sie im Kapitel AUTOHOTSPOT.

Das Beispielprogramm verwendet die Schreibliste "WriteListStatus", die die folgende PLC-Variable enthält:

- ProductionEnabled

Das Programm schreibt den neuen Wert dieser Variablen zum Server.

Das Kapitel AUTOHOTSPOT zeigt, wie Sie eine Client-Schnittstelle anlegen und ihr eine Schreibliste mit PLC-Variablen hinzufügen.

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Im 4. Programmabschnitt werden die Werte von Variablen geschrieben:

SCL

```
4:  // case 4;  write value to PLC variable "ProductionEnabled"

            IF #SetProductionEnabled = TRUE THEN
                #SetProductionEnabled := FALSE;
                //set new value to true
               "Productionline_Data"."WriteListStatus".Variable.ProductionEnabled := TRUE;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_WriteList_Instance(
                                        NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                        NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                        Variable :=
"Productionline_Data"."WriteListStatus".Variable,
                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount,
                                        REQ := #Req,
                                        Busy => #Busy,
                                        Error => #Error,
                                        Status => #Status,
                                        Done => #Done);
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_WriteList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 4 dargestellt.

Dieser Fall weist im Beispielprogramm nur einer PLC-Variablen einen neuen Wert zu. In Ihrer Anwendung können Sie sehr vielen PLC-Variablen neue Werte zuschreiben.

Der neuen Wert wird auf TRUE gesetzt, wenn die Variable "#SetProductionEnabled" den Wert "TRUE" enthält.

Anschließend setzen wir "#SetProductionEnabled" auf FALSE, weil die Initialisierung nur einmal erfolgen soll (im ersten Zyklus).

Die Anweisung "OPC_UA_WriteList" schreibt den neuen Wert in die PLC-Variable "ProductionEnabled" im OPC UA-Server.

Wenn die Anweisung noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
             IF #Done = TRUE THEN
                FOR #i := 0 TO
UINT_TO_INT("Productionline_Configuration".WriteLists.WriteListStatus.NodeCount) - 1 DO
                   IF NOT ("Productionline_Data"."WriteListStatus".NodeStatusList[#i] = 0)
THEN
                        #Output_Error_Message := CONCAT_WSTRING(IN1 := WSTRING#'Error at
Writelist "WriteListStauts", Index: ',
                                                                    IN2 := INT_TO_WSTRING(#i));
                    END_IF;
                END_FOR;
                #State := #State + 1;
            END_IF;
            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_WriteList_Instance(
                                            NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                            NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                            Variable :=
"Productionline_Data"."WriteListStatus".Variable,
                                            REQ := FALSE,
                                            ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                            NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall wird der Ausgangsparameter "Error" auf den Wert TRUE gesetzt. Dadurch wird die Variable "#State" auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen. Zudem ruft das Beispielprogramm die Anweisung "OPC_UA_WriteList" auf, um den Parameter REQ auf FALSE zu setzen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt.

Achtung: Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber eine PLC-Variable nicht gesetzt werden konnte.

Deshalb wird im Beispiel für die PLC-Variable geprüft, ob das Programm einen gültigen Wert setzen konnte:

- Kann der Server der 1. PLC-Variablen den neuen Wert zuschreiben, steht im 1. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
  Im Beispiel ist nur eine Variable verwendet. Ihr Statuscode steht in "Productionline_Data"."WriteListStatus".NodeStatusList[0]".

Das Programm gibt eine Fehlermeldung aus, wenn eine PLC-Variable nicht gesetzt werden konnte.

Der Wert der Variable "#State" wird um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 5 - (Release Handle List) ausgeführt.

Siehe auch

OPC UA-Anweisungen für Client-Programme (Seite 3797)

OPC_UA_NodeReleaseHandleList: Handles für Lese- und Schreibzugriffe freigeben (Seite 3875)

Array-Bereich schreiben mit OPC_UA_WriteList

Einleitung

Dieses Kapitel basiert auf dem Kapitel "OPC_UA_WriteList", siehe OPC_UA_WriteList (Seite 3860).

Das Kapitel beschreibt, wie Sie mit der Anweisung "OPC_UA_WriteList" einem Teilbereich eines Arrays neue Werte zuweisen (schreiben).

Prinzipiell funktioniert das Schreiben in einen Array-Teilbereich eines OPC UA-Servers folgendermaßen:

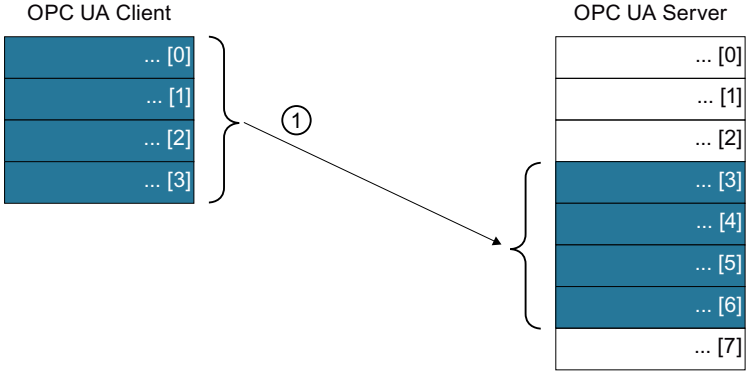

① Schreiben von Array-Elementen in ein Array-Teilbereich eines OPC UA-Servers

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine CPU als **OPC UA-Server** konfiguriert.
  Der Server stellt ein Array bereit, das von OPC UA-Clients beschrieben werden kann.
  Im Beispiel verwenden Sie die CPU "Productionline", die für OPC UA-Clients den Datenbaustein "Data_from_OPC_UA_Clients" bereitstellt:

Static	
ProductionEnabled	Bool
Serial_Number	Array[0..9] of LInt
Serial_Number[0]	LInt
Serial_Number[1]	LInt
Serial_Number[2]	LInt
Serial_Number[3]	LInt
Serial_Number[4]	LInt
Serial_Number[5]	LInt
Serial_Number[6]	LInt
Serial_Number[7]	LInt
Serial_Number[8]	LInt
Serial_Number[9]	LInt

- Sie haben eine CPU als **OPC UA-Client** konfiguriert.
  Sie haben im Client eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT".
  Die Client-Schnittstelle enthält eine Schreibliste mit einem Array.
  Nur ein Teilbereich dieses Arrays soll für das Beschreiben des Arrays im Server genutzt werden.
  Das Beispiel verwendet die Client-Schnittstelle "Productionline" mit der Schreibliste "WriteListStatus":

ProductionEnabled	BOOL
Serial_Number	ARRAY[0..9] of LINT

  In das Array "Serial_Number" des Servers sollen neue Werte in "Serial_Number[5]" bis "Serial_Number[9]" geschrieben werden.

Schreiblisten-Datentyp und Schreiblisten-DB anpassen

Das Kapitel "OPC_UA_WriteList" zeigt, wie Sie mit der Anweisung "OPC_UA_WriteList" neue Werte in Skalare (Variablen mit einem Wert) oder in Arrays (Variablen mit mehreren Werten) im Ganzen schreiben.

Im folgenden Abschnitt sind die Erweiterungen zum Kapitel "OPC_UA_WriteList" dargestellt, die notwendig sind, um mit OPC_UA_WriteList neue Werte in einen Teilbereich eines Arrays zu schreiben.

Schritt 1: Schreiblisten-Datentyp kopieren und anpassen

Zunächst ändern Sie den Datentyp der Variablen, die die Werte bereitstellt, die Sie zum OPC UA-Server schreiben.

Die Änderung ist notwendig, weil der von STEP 7 erzeugte Datentyp zu lang ist.

Der Datentyp ist zu lang, weil Sie nur einem Teilbereich des Arrays neue Werte zuweisen wollen (nicht dem ganzen Array).

Im Beispiel greifen Sie auf das Array "Serial_Number" zu, das zehn Elemente enthält.

Deshalb erzeugt STEP 7 einen Datentyp mit zehn Elementen.

Um diesen Datentyp zu verkürzen, gehen Sie folgendermaßen vor:

1. Im Bereich "Projektnavigation" öffnen Sie den Ordner "PLC-Datentypen"
2. Öffnen Sie den Ordner "Systemdatentypen".
3. Kopieren Sie den Datentyp "<Client-Schnittstelle>.<Schreibliste>".
   Im Beispiel heisst der Datentyp "Productionline.WriteListStatus".
4. Fügen Sie den kopierten Datentyp in den Ordner "PLC-Datentypen" ein.
5. Ändern Sie den Namen des eingefügten Datentyps, sodass er in Ihrem Projekt aussagekräftig ist.
   Im Beispiel ändern wir den Namen in "myUDTProductionline.WriteListStatus".
6. Ändern Sie den Datentyp des Arrays.
   Im Beispiel wollen wir die Werte von Serial_Number[5] bis Serial_Number[9] zum OPC UA-Server schreiben.
   Deshalb verkürzen wir den Datentyp von "Array[0..9] of Lint" auf "Array[0..4] of Lintl".
   Im Beispiel:

ProductionEnabled	Bool
Serial_Number	Array[0..4] of LInt
Serial_Number[0]	LInt
Serial_Number[1]	LInt
Serial_Number[2]	LInt
Serial_Number[3]	LInt
Serial_Number[4]	LInt

Schritt. 2: Schreiblisten-DB kopieren und neuen Schreiblisten-Datentyp darauf anwenden

In diesem Schritt verwenden Sie den neuen Datentyp in einem neuen Datenbaustein.

Dieser Schritt ist notwendig, weil der von STEP 7 erzeugte Datenbaustein den ursprünglichen Datentyp für das Array enthält.

Es ist nicht möglich, den Datentyp zu ändern: Bei jedem Übersetzen des Datenbausteins wird der ursprüngliche Datentyp wieder hergestellt. Dabei gehen die Änderungen verloren.

Um einen neuen Datenbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungiert.
2. Öffnen Sie den Ordner "Programmbausteine"
3. Kopieren Sie den Datenbaustein <Client-Schnittstelle>_Data
   Im Beispiel heisst der Datenbaustein "Productionline_Data"
4. Fügen Sie den kopierten Datenbaustein in den Ordner "Programmbausteine" ein.
5. Ändern Sie den Namen des eingefügten Datenbausteins, sodass er in Ihrem Projekt aussagekräftig ist.
   Im Beispiel ändern wir den Namen in "myProductionline_Data"

6. Verwenden Sie nun den neuen Datentyp, den wir im Schritt 1 angelegt haben.
   Im Beipiel weisen wir der Variablen "Variable" in der Schreibliste "WriteListStatus" den neuen Datentyp zu:

WriteListStatus	Struct
Variable	"myUDT_Productionline.WriteListStatus"
ProductionEnabled	Bool
Serial_Number	Array[0..4] of LInt
Serial_Number[0]	LInt
Serial_Number[1]	LInt
Serial_Number[2]	LInt
Serial_Number[3]	LInt
Serial_Number[4]	LInt

7. Versorgen Sie den Parameter "Variable" der Anweisung "OPC_UA_WriteList" mit dem neuen Datenbaustein (im Beispiel "myProductionline_Data").
   `Variable := "myProductionline_Data"."WriteListStatus".Variable,`

Schritt 3: Variable für OPC_UA_ReadList-Parameter "NodeAddInfos" erzeugen

In diesem Schritt erzeugen Sie eine lokale Variable des Typs "OPC_UA_NodeAdditionalInfo".

Diese Variable dient dazu, der Anweisung "OPC_UA_WriteList" mitzuteilen, welchem Array-Bereich neue Werte zugewiesen werden.

Gehen Sie folgendermaßen vor:

1. Legen Sie eine Array-Variable des Typs "OPC_UA_NodeAdditionalInfo" an.
   Im Beispiel legen Sie die Variable im Deklarationsteil des Funktionsbausteins an, der die Anweisung "OPC_UA_WriteList" aufruft.
   Die neue lokale Variable nennen Sie "NodeAdditionalInformation":

NodeAdditonalInformation	Array[0..1] of OPC_UA_NodeAdditionalInfo
NodeAdditonalInformation[0]	OPC_UA_NodeAdditionalInfo
AttributeID	UDInt
StartIndex	UDInt
EndIndex	UDInt
NodeAdditonalInformation[1]	OPC_UA_NodeAdditionalInfo
AttributeID	UDInt
StartIndex	UDInt
EndIndex	UDInt

   Die neue lokale Variable "NodeAdditionalInformation" enthält zwei Elemente des Typs "OPC_UA_AdditonalInfo".
   NodeAdditionalInformation[0] stellt zusätzliche Informationen für die erste zu schreibende Variable bereit (im Beispiel "ProductionEnabled")
   NodeAdditionalInformation[1] stellt zusätzliche Informationen für die zweite zu schreibende Variable bereit (im Beispiel "Serial_Number")

2. Weisen Sie der neuen Variablen Werte zu (in der Spalte "Defaultwerte" des Deklarationsteils).
   Im Beispiel schreiben Sie nicht alle Elemente der zweiten Variablen (Serial_Number). Deshalb weisen Sie NodeAdditionalInformation[1].AttributeId den Wert 13 zu (d. h., die Werte des Arrays werden geschrieben).
   NodeAdditionalInformation[1].StartIndex erhält den Wert 5, da Sie ab Index 5 (einschließlich) schreiben.
   NodeAdditionalInformation[1].EndIndex erhält den Wert 9, da Sie bis Index 9 (einschließlich) schreiben.
   Für NodeAdditionalInformation[0] setzen Sie folgende Werte, die dem Server signalisieren, dass das erste Element der Schreibliste kein Array-Teilbereich ist, sondern ein Skalar oder ein Array, das vollständig geschrieben werden soll.
   Die Konstante "#ignore" enthält den Wert 4_294_967_295. Der Server ignoriert diesen Wert für StartIndex und EndIndex und schreibt Skalare und Arrays vollständig.

NodeAdditonalInformation	Array[0..1] of OPC_UA_NodeAdditionalInfo	
NodeAdditonalInformation[0]	OPC_UA_NodeAdditionalInfo	
AttributeID	UDInt	13
StartIndex	UDInt	#ignore
EndIndex	UDInt	#ignore
NodeAdditonalInformation[1]	OPC_UA_NodeAdditionalInfo	
AttributeID	UDInt	13
StartIndex	UDInt	5
EndIndex	UDInt	9

   Alternative:
   Sie können Sie die Werte für NodeAdditionalInformation auch im Anwenderprogramm setzen:

```
// set additonal information to Array-Index 0
// we set default values here
#NodeAdditonalInformation[0].AttributeID := 13;
```

```
                #NodeAdditonalInformation[0].StartIndex :=
4_294_967_295;
                #NodeAdditonalInformation[0].EndIndex :=
4_294_967_295;

                // set additonal information to Array-Index 1
               // because we want to write only five values to the
array in the OPC UA server
                #NodeAdditonalInformation[1].AttributeID := 13;
                #NodeAdditonalInformation[1].StartIndex := 5;
                #NodeAdditonalInformation[1].EndIndex := 9;
```

Hinweis:
Das Array NodeAdditionalInformation des Typs "OPC_UA_NodeAdditionalInfo" muss so viele Elemente enthalten wie die Schreibliste (im Beispiel 2).

3. Weisen Sie die lokale Variable NodeAdditionalInformation dem Parameter "NodeAddInfos" der Anweisung "OPC_UA_WriteList" zu.
   Das folgende Bild zeigt den Aufruf der OPC_UA_WriteList mit der neuen Versorgung der Parameter "Variable" und "NodeAddInfos":

```
#OPC_UA_WriteList_Instance(
                           NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHd
ls,
                           NodeStatusList :=
"myProductionline_Data"."WriteListStatus".NodeStatusList,
                           Variable :=
"myProductionline_Data"."WriteListStatus".Variable,
                           ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                           Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeo
ut,
                           NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCo
unt,
                        NodeAddInfos := #NodeAdditonalInformation,
                           REQ := #Req,
                           Busy => #Busy,
                           Error => #Error,
                           Status => #Status,
                           Done => #Done);
```

Ergebnis im Server

Die Anweisung "OP_UA_WriteList" schreibt jetzt nur einen Teilbereich des Arrays "Serial_Number" zum OPC UA-Server.

Das folgende Bild zeigt das Array "Serial_Number". Der Server weist die geschriebenen Werte ab dem Index zu, den Sie durch NodeAdditionalInformation bestimmen (Im Beispiel ab Index 5):

Serial_Number	Array[0..9] of LInt		
Serial_Number[0]	LInt	0	0
Serial_Number[1]	LInt	0	0
Serial_Number[2]	LInt	0	0
Serial_Number[3]	LInt	0	0
Serial_Number[4]	LInt	0	0
Serial_Number[5]	LInt	0	100
Serial_Number[6]	LInt	0	101
Serial_Number[7]	LInt	0	102
Serial_Number[8]	LInt	0	103
Serial_Number[9]	LInt	0	104

Hinweis

Verwenden Sie den Datentyp "OPC_UA_NodeAdditionalInfoExt", um in einen Teilbereich eines mehrdimensionalen Arrays zu schreiben (bis zu sechs Dimensionen).

Datenaustausch beenden

OPC_UA_NodeReleaseHandleList: Handles für Lese- und Schreibzugriffe freigeben

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_WriteList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_WriteList" geben Sie Handles frei.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

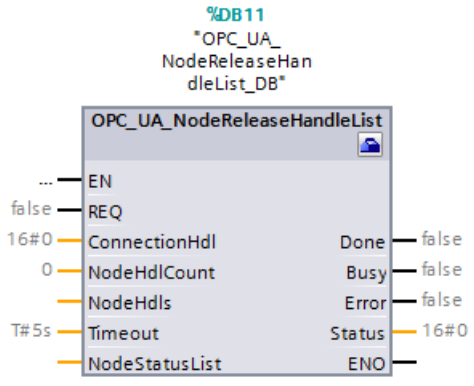

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_NodeReleaseHandleList" dient dazu, Handles für PLC-Variablen frei zu geben, die zuvor mit der Anweisung "OPC_UA_NodeGetHandleList" registriert worden sind, siehe ③ in der folgenden Grafik.

Die Anweisung "OPC_UA_NodeReleaseHandleList" ist nicht erforderlich, wenn Sie die Verbindung zum OPC UA-Server trennen wollen. In diesem Fall rufen Sie gleich die Anweisung "OPC_UA_Disconnect" auf. Diese Anweisung gibt alle Handles automatisch frei.

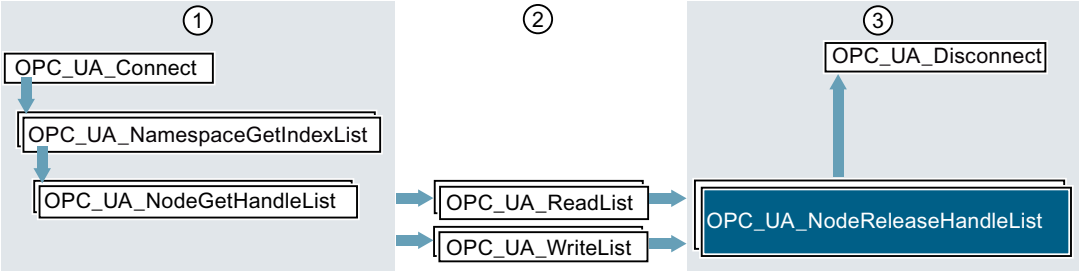

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_NodeReleaseHandleList"

Tabelle 4-78 Die Parameter des Funktionsbausteins "OPC_UA_NodeReleaseHandleList"

Parameter	Deklaration im Bereich	S7-Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung: Verbindungs-Handle. Das Verbindungs-Handle erhalten Sie durch den Aufruf des Funktionsbausteins OPC_UA_Connect.
NodeHdlCount	Input	UINT	Anzahl der Elemente im Parameter NodeHdls
NodeHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Handles für die Variablen, die Sie in Ihrem Programm nicht länger lesen oder setzen wollen.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung des Funktionsbausteins. Siehe auch Erläuterung zu diesem Parameter bei AUTO-HOTSPOT
NodeStatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die einzelnen Node-Handles (Variablen), siehe "Fehlernummern im Parameter NodeStatusList". Für jedes Node-Handle ist angegeben, ob es frei gegeben werden konnte.

Parameter	Deklaration im Bereich	S7-Datentyp	Bedeutung
Done	Output	BOOL	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	Output	BOOL	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	siehe unten "Fehlernummern im Parameter Status"

Fehlernummern für Status

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Funktionsbaustein erfolgreich ausgeführt.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	NodeHdlCount > MAX_ELEMENTS_NODELIST
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "NodeStatusList"

Der Parameter "NodeStatusList" enthält für jede einzelne Node-Handle (Variable) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Handle erfolgreich frei geben.
8034_0000	OpcUa_BadNodeUnknown	Das übergebene Node-Handle ist nicht bekannt.
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_NodeReleaseHandleList" in einem Programm verwenden, das Daten mit einem OPC UA-Server austauscht.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben ein Client-Interface angelegt.
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_NodeReleaseHandleList" die folgenden Voraussetzungen erfüllt sein:

- Ein Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle durch den Aufruf des Funktionsbausteins AUTOHOTSPOT.
- Eine Liste mit den Handles für die einzelnen Variablen, deren Werte Sie in Ihrem Programm nicht länger lesen oder schreiben wollen.
  Sie erhalten diese Handle-Liste durch den Aufruf des Funktionsbausteins AUTOHOTSPOT

Funktion der Anweisung

Sie rufen die Anweisung "OPC_UA_NodeReleaseHandleList" auf, um Node-Handles frei zu geben, wenn Sie diese Handles nicht mehr in Ihrem Programm benötigen.

Node-Handles sind numerische Referenzen auf Knoten (zum Beispiel PLC-Variablen) im OPC UA-Server.

Welche Informationen sind für den Funktionsbaustein erforderlich?

Der Funktionsbaustein benötigt die folgenden Angaben:

- Eine Liste mit den Handles für die Variablen (oder andere Knoten), deren Werte Sie nicht mehr lesen oder schreiben wollen (im Parameter "NodeHdls").

Welche Informationen gibt der Funktionsbaustein zurück?

Der Funktionsbaustein liefert die folgenden Angaben:

- Eine Liste mit Fehlermeldungen (im Parameter "NodeStatusList").
  Jede Fehlermeldung in dieser Liste bezieht sich auf das entsprechende Handle im Parameter "NodeHdls".
  Sie müssen für jedes einzelne Node-Handle prüfen, ob es der OPC UA-Server frei geben konnte.
  Wenn ein Node-Handle nicht freigegeben werden konnte, dann muss Ihr Anwenderprogramm diesen Fehler abfangen: Rufen Sie zum Beispiel in einer Fehlerbehandlung diese Anweisung nochmals auf.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungieren soll.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "ReadFromProductionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_NodeReleaseHandleList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_NodeReleaseHandleList_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_NodeReleaseHandleList_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", die wir für unsere AUTOHOTSPOT angelegt hatten.
7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Leseliste.
   Im Beispiel heisst die Leseliste "ReadListProduct".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.
8. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu..
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Das vollständige Programm finden Sie im Kapitel "Beispielprogramm für das Lesen von PLC-Variablen (Seite 3950)"

Das Beispielprogramm verwendet die Leseliste "ReadListProduct", die die folgenden PLC-Variablen enthält:

- NewProduct
- ProductNumber

Die Handles für diese PLC-Variablen gibt das folgende Programmbeispiel frei.

CASE-Anweisung

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Hier ist der Fall 5 dargestellt:

SCL

```
5:  // case 5, release node handles

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NodeReleaseHandleList_Instance(
                                                    NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                    NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                    REQ := #Req,
                                                    ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                    Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                    NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                                    Done => #Done,
                                                    Error => #Error,
                                                    Busy => #Busy,
                                                    Status => #Status);
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_NodeReleaseHandleList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 5 dargestellt.

Dieser Programmteil ruft die Anweisung "OPC_UA_NodeReleaseHandleList" auf. Damit werden Handles für PLC-Variablen freigegeben. Die Variablen sind in der Leseliste "ReadListProduct" zusammengefasst.

Wenn die Anweisung noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
            IF #Done = TRUE THEN
              IF "Productionline_Data"."ReadListProduct".NodeStatusList[0]
= 0 AND
                 "Productionline_Data"."ReadListProduct".NodeStatusList[1]
= 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction
"OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at
NodeReleaseHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
               //In case 100, to set REG of instruction "OPC_UA_Disconnect"
to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_NodeReleaseHandleList_Instance(
                                                        NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                        NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                        REQ := FALSE,
                                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                        NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall wird der Ausgangsparameter "Error" auf den Wert TRUE gesetzt. Dadurch wird die Variable #State auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen. Zudem ruft das Beispielprogramm die Anweisung "OPC_UA_NodeReleaseHandleList" auf, um den Parameter REQ auf FALSE zu setzen.

Steht im Ausgangsparameter "Done" der Wert TRUE, wurde die Anweisung erfolgreich ausgeführt.

Achtung: Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber ein Handle für eine PLC-Variablen nicht frei gegeben werden konnte.

Deshalb wird im Beispiel geprüft, ob die Handles für beide Variablen frei gegeben wurde:

- Hat der Server das Handle für die 1. PLC-Variable frei gegeben, steht im 1. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
  Im Beispiel steht der Statuscode in "Productionline_Data"."ReadListProduct".NodeStatusList[0].
- Hat der Server das Handle für die 2. PLC-Variable frei gegeben, steht im 2. Element des Arrays, auf das der Parameter "NodeStatusList" zeigt, der Wert 0.
  Im Beispiel steht der Statuscode in "Productionline_Data"."ReadListProduct".NodeStatusList[1].

Wenn beide Bedingungen erfüllt sind, dann wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschritt - Fall 6 (Verbindung schließen) - ausgeführt.

Fall 6 ist in AUTOHOTSPOT beschrieben.

Hinweis

Fall 6 dient dazu, die Verbindung zum OPC UA-Server freizugeben. Damit sind auch alle anderen Ressourcen freigegeben, die das Client-Programm im Server belegt hat. Wenn Sie die Verbindung zum Server beenden wollten, dann kann Ihr Anwenderprogramm unmittelbar Fall 6 ausführen und Fall 5 überspringen.

Siehe auch

OPC UA-Anweisungen für Client-Programme (Seite 3797)

OPC_UA_Disconnect: Verbindung schließen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_Disconnect" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Mit der Anweisung "OPC_UA_Disconnect" beenden Sie die Verbindung zu einem OPC UA-Server.

Das folgende Bild zeigt das Symbol der Anweisung im Editor (FUP).

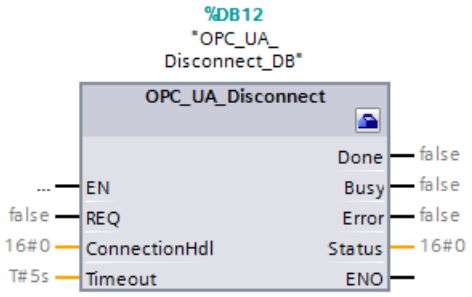

Im Bild oben sind die Parameter der Anweisung noch nicht versorgt.

Die Anweisung "OPC_UA_Disconnect" beendet die Verbindung und Session mit einem OPC UA-Server, siehe ③ in der folgenden Grafik.

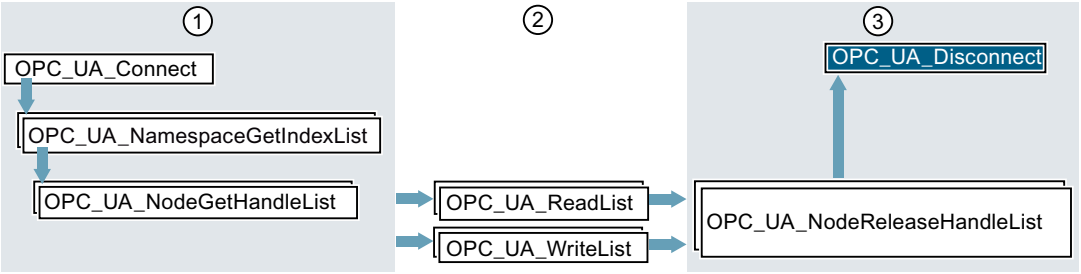

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum Freigeben von Ressourcen nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_Disconnect"

Tabelle 4-79 Die Parameter des Funktionsbausteins "OPC_UA_Disconnect"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung: Verbindungs-Handle. Das Verbindungs-Handle erhalten Sie durch den Aufruf des Funktionsbausteins OPC_UA_Connect.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung des Funktionsbausteins. Siehe auch Erläuterung zu diesem Parameter bei OPC_UA_Connect: Verbindung erzeugen (Seite 3799)
Done	Output	BOOL	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	Output	BOOL	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	siehe unten "Fehlernummern im Parameter Status"

Fehlernummern im Parameter Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Funktionsbaustein erfolgreich ausgeführt.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) kann nicht durchgeführt werden, weil zu viele Anweisungen enthalten sind.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C600	Simatic_ClientNotAvailable	Fehler beim Initialisieren des Clients
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

So verwenden Sie diese Anweisung

Dieses Kapitel zeigt anhand eines Programmbeispiels, wie Sie die Anweisung "OPC_UA_Disconnect" in einem Programm verwenden, das Daten mit einem OPC UA-Server austauscht.

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben ein Client-Interface angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_Disconnect" die folgende Voraussetzung erfüllt sein:

- Ein Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle durch den Aufruf der Anweisung OPC_UA_Connect: Verbindung erzeugen (Seite 3799).

Funktion der Anweisung

Sie rufen die Anweisung "OPC_UA_Disconnect" auf, um die Verbindung zu einem OPC UA-Server zu beenden.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungieren soll.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "ReadFromProductionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_Connect" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_Disconnect_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_Disconnect_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", die für die AUTOHOTSPOT angelegt wurde, siehe "AUTOHOTSPOT".
   STEP 7 versorgt nun automatisch die Parameter der Anweisung mit den Werten, die Sie bei der Client-Schnittstelle konfiguriert haben.
7. Klicken Sie auf "Bausteinparameter" und weisen Sie den Parametern REQ, Busy, Done, Error, Status manuell Variablen zu.
   STEP 7 fügt jeweils die gewählte Variable in den Aufruf der Anweisung ein.

Aufruf der Anweisung

Das vollständige Programm finden Sie im Kapitel "AUTOHOTSPOT"

Das Beispielprogramm beendet die Verbindung und die Session mit einem OPC UA-Server:

SCL

```
6:  // case 6, disconnect from server

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Done => #Done,
                                        Error => #Error,
                                        Busy => #Busy,
                                        Status => #Status);
```

Das Beispielprogramm ruft die Anweisung "OPC_UA_Disconnect" aus den folgenden Gründen auf:

- Um die Verbindung zum OPC UA-Server zu beenden.
- Um den Parameter REQ auf FALSE zu setzen, falls ein Fehler aufgetreten ist.

Asynchrone Ausführung

Die Anweisung "OPC_UA_Disconnect" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung

Im Bild oben ist Fall 6 dargestellt. Dieser Programmteil beendet die Verbindung und Session mit dem verbundenen OPC UA-Server.

Wenn die Variable #State den Wert 6 besitzt, dann wird Fall 6 ausgeführt.

Wenn die Anweisung "OPC_UA_Disconnect" noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Wenn der Ausgangsparameter "Done" den Wert TRUE besitzt, dann konnte das Handle für die Verbindung freigegeben sowie die Verbindung abgebaut werden. Dadurch wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 7 - ausgeführt. Dieser Fall setzt die Statusangaben für den Funktionsbaustein (Output_Busy = FALSE, Output_Done = TRUE).

Kann die Anweisung nicht fehlerfrei ausgeführt werden, steht im Ausgangsparameter "Error" der Wert TRUE und die Variable #State erhält den Wert 100. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

Diagnose

OPC_UA_ConnectionGetStatus: Verbindungsstatus lesen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_ConnectionGetStatus" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Sie verwenden die Anweisung "OPC_UA_ConnectionGetStatus", um Informationen über die Qualität der Verbindung zu einem OPC UA-Server zu erhalten.

OPC_UA_ConnectionGetStatus dient dazu, den Datenaustausch vorzubereiten, siehe ① in der folgenden Grafik.

Die Anweisung liefert Informationen zu Verbindung, Server und Service-Level zurück.

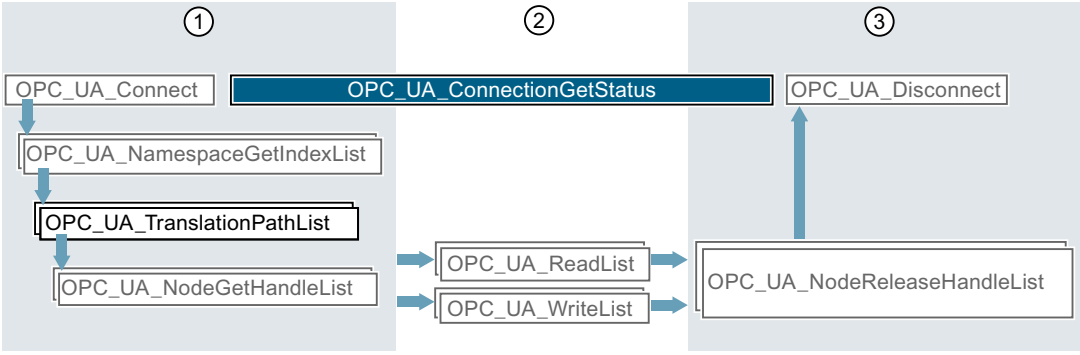

① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum "Clean-up" nach durchgeführten Lese- oder Schreibvorgängen

Parameter für "OPC_UA_ConnectionGetStatus"

Tabelle 4-80 Die Parameter des Funktionsbausteins "OPC_UA_ConnectionGetStatus"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung AUTOHOTSPOT.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"
ConnectionStatus	Output	UDINT	Status der Verbindung, die im Parameter ConnectionHdl angegeben ist. Bedeutung: • 0: Verbindung zum OPC UA-Server ist aufgebaut. • 1: Fehlerhafte Verbindung zum OPC UA-Server. • 2: Die Verbindung zum OPC UA-Server wurde durch den Server getrennt.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
ServerState	Output	UDINT	Informationen zum OPC UA-Server Bedeutung der Werte: • 0: Der Server läuft normal. • 1: Ein Fehler hat sich im Server ereignet. Der Server funktioniert nicht länger zuverlässig. • 2: Der Server läuft. Es wurde aber keine Konfiguration geladen. Deshalb findet kein Datenaustausch statt. • 3: Der Server steht vorübergehend nicht zur Verfügung. • 4: Der Server wurde abgeschaltet oder wird gerade abgeschaltet. • 5: Der Server läuft im Testmodus. Ausgänge sind nicht geschaltet. • 6: Der Server läuft normal. Es gibt jedoch Probleme beim Lesen der PLC-Variablen. • 7: Zustand des Servers ist nicht bekannt. Dieser Wert wird gesetzt, wenn keine Verbindung zum Server besteht.
ServiceLevel	Output	Byte	Eine Angabe zur Leistungsfähigkeit des verbundenen OPC UA-Servers. Der Server liefert eine Zahl zwischen 0 und 255 zurück, um seine Leistungsfähigkeit anzuzeigen: Je größer die Zahl, desto leistungsfähiger ist der Server.

Fehlernummern für Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hexadezimale Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Der Server sendet eine unbekannte oder ungültge Antwort.
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
8074_0000	OpcUa_BadTypeMismatch	ServerState / ServiceLevel Value(s) haben unerwarteten Typ
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Voraussetzung

Für die Anweisung "OPC_UA_ConnectionGetStatus" muss die folgende Voraussetzung erfüllt sein:

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungshandle von der Anweisung AUTOHOTSPOT.

Aufruf der Anweisung

Dieses Kapitel beschreibt den Funktionsbaustein "Analyze_Connection". Der Funktionsbaustein zeigt, wie Sie die Anweisung "OPC_UA_ConnectionGetStatus" verwenden können.

Das vollständige Beispiel finden Sie im Kapitel Beispielprogramm für OPC_UA_ConnectionGetStatus (Seite 3944).

Deklaration der Variablen

Deklarieren Sie eine Instanz der Anweisung "OPC_UA_ConnectionGetStatus" sowie die Variablen, mit denen Sie die Parameter der Anweisung versorgen.

Das Programmbeispiel verwendet die folgende Deklaration:

ConnectionHdl	DWord
ConnectionStatus	UDInt
ServerState	UDInt
ServiceLevel	Byte
▸ OPC_UA_ConnectionGetStatus_Instance	OPC_UA_ConnectionGetStatus

Anwenderprogramm

Das folgende Bild zeigt den Abschnitt aus dem Beispielprogramm "Analyze_Connection", das die Qualität der Verbindung zum Server ermittelt.

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Hier ist der Fall 2 (Analyse der Verbindung) dargestellt:

SCL

```
2: // case 2, analyse connection
   IF #Busy = FALSE THEN
       #Req := TRUE;
   ELSE
       #Req := FALSE;
   END_IF;

   #OPC_UA_ConnectionGetStatus_Instance(ConnectionHdl := #ConnectionHdl,
                                        ConnectionStatus => #ConnectionStatus,
                                        ServerState => #ServerState,
                                        ServiceLevel => #ServiceLevel,
                                        Timeout := T#6S,
                                        Done => #Done,
                                        Error => #Error,
                                        Status => #Status,
                                        Busy => #Busy,
                                        REQ := #Req);
   IF #Done = TRUE THEN
       #State := #State + 1;
   END_IF;

   IF #Error = TRUE THEN
       #State := 100;
       #Mem_Status := #Status;
       #OPC_UA_ConnectionGetStatus_Instance(REQ := FALSE, ConnectionHdl := #ConnectionHdl);
   END_IF;
```

Im Fehlerfall der Anweisung "OPC_UA_ConnectionGetStatus" (#Error hat den Wert "TRUE") wird der Parameter REQ auf den Wert FALSE gesetzt.

Asynchrone Ausführung

Die Anweisung "OPC_UA_NodeGetHandleList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung

Das Bild oben zeigt den Fall 2 des Beispielprogramms "Analyze_Connection".

Dieser Teil des Programms erfragt die aktuellen Werte für "ConnectionStatus", "ServerState" und "ServiceLevel" vom verbundenen OPC UA-Server.

Wenn die Variable #State den Wert 2 besitzt, dann wird Fall 2 ausgeführt.

Wenn die Anweisung "OPC_UA_ConnectionGetStatus" noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Wenn im Ausgangsparameter "Error" der Wert TRUE steht, dann trat bei der Ausführung der Anweisung ein Fehler auf. Dadurch wird die Variable "#State" auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt. Dadurch wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 3 - ausgeführt. Fall 3 ist in OPC_UA_Disconnect: Verbindung schließen (Seite 3883) beschrieben.

Methoden

OPC_UA_MethodGetHandleList: Handles für Methodenaufrufe holen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_MethodGetHandleList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Ein Programmierbeispiel zeigt Ihnen, wie Sie die Anweisung in der Programmiersprache SCL verwenden.

Dieses Kapitel geht davon aus, dass Sie für die OPC UA-Kommunikation eine Client-Schnittstelle anlegen und parametrieren, siehe: AUTOHOTSPOT. Diese effiziente Vorgehensweise sorgt dafür, dass die meisten Parameter der Client-Anweisung automatisch versorgt werden.

Sie verwenden die Anweisung "OPC_UA_MethodGetHandleList" in Ihrem Anwenderprogramm, um vom Server numerische Referenzen (Handles) für Methoden zu erhalten, die der Server für OPC UA-Clients zur Verfügung stellt. OPC_UA_MethodGetHandleList dient dazu, Methodenaufrufe vorzubereiten, siehe ① in der folgenden Grafik.

Die Anweisung liefert Handles für die Server-Methoden zurück.

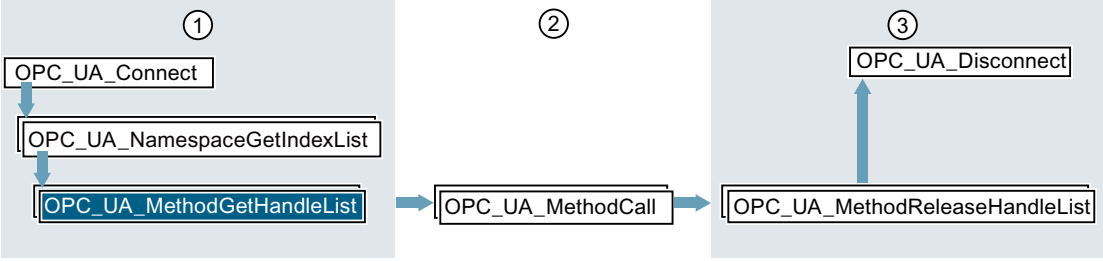

① Anweisungen zur Vorbereitung der Methodenaufrufe
② Methodenaufrufe
③ Anweisungen zum "Clean-up" nach durchgeführten Methodenaufrufen

Parameter für "OPC_UA_MethodGetHandleList"

Tabelle 4-81 Die Parameter der Anweisung "OPC_UA_MethodGetHandleList"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung OPC_UA_Connect.
NodeIDCount	Input	UINT	Anzahl der Elemente in den Parametern ObjectNodeIDs bzw. MethodeNodeIDs.
ObjectNodeIDs	InOut	VARIANT	Zeiger auf ein Array mit Elementen des Typs OPC_UA_NodeId. Das Array enthält die Knoten-IDs (NodeIDs) der Objekte (Ordner), in denen die jeweiligen Methoden liegen.
MethodNodeIDs	InOut	VARIANT	Zeiger auf ein Array mit Elementen des Typs OPC_UA_NodeId. Das Array enthält die Knoten-IDs (NodeIDs) der Methoden, für die vom Server Handles erfragt werden.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden.Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
NamespaceIndexCount	Input	UINT	Anzahl der Array-Elemente im Parameter NamespaceIndexes.
NamespaceIndexes	InOut	VARIANT	Zeiger auf ein Array des Typs UINT. Dieser Parameter ist nicht erforderlich, wenn es keine Elemente im Parameter NamespaceIndexCount gibt (NamespaceIndexCount = 0). Das Array wird verwendet, um den Namensraumindizes im Parameter NodeIDs neue Indizes zuzuordnen. Beispiel: Wenn im NamespaceIndexes[4] der Wert 10 steht, dann wird allen NodeIDs im Parameter "NodeIDs", die bislang den Namensraumindex 4 hatten, der Namensraumindex 10 zugeordnet.

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"
StatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die einzelnen Methoden, siehe unten "Fehlernummern für NodeStatusList". Für jede Methode ist angegeben, ob ihr ein Handle zugeordnet werden konnte.
MethodHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Handles für die einzelnen Methoden (NodeIDs), die der OPC UA-Server zurückgegeben hat.

Parameter "ObjectNodeIDs" und "MethodNodeIDs" ermitteln

ObjectNodeIDs

Die OPC_UA_NodeId, die am Parameter "ObjectNodeIDs" einzutragen ist, können Sie z. B. mit UaExpert ermitteln:

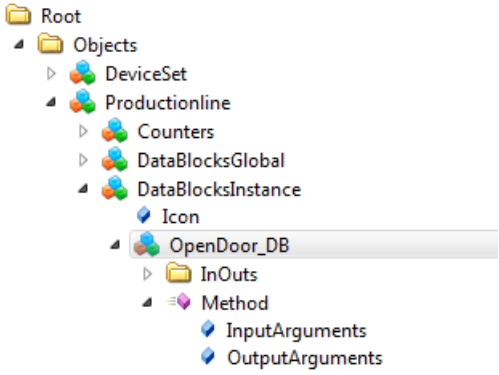

NodeId	NodeId
NamespaceIndex	3
IdentifierType	String
Identifier	"OpenDoor_DB"

MethodNodeIDs

Die OPC_UA_NodeId am Parameter "MethodNodeIDs" finden Sie hier:

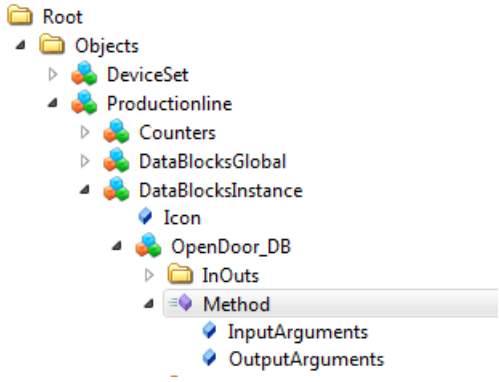

NodeId	NodeId
NamespaceIndex	3
IdentifierType	String
Identifier	"OpenDoor_DB".Method

Fehlernummern für Status

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8009_0000	OpcUa_BadUnknownResponse	Server antwortet nicht mit der erwarteten Anzahl an Ergebnissen
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.
8074_0000	OpcUa_BadTypeMismatch	Sie verwenden einen Datentyp, der nicht zum Datentyp im Server passt.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	Simatic_BadType_VariantInput3	Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	Simatic_BadType_VariantInput4	Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
B080_0500	Simatic_BadType_VariantInput5	Der Datentyp der Variablen stimmt nicht, auf die der 5. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	Maximal erlaubte Anzahl für Methodenliste (Max_ELEMENTS_METHODLIST) überschritten.
B080_3100 B080_3200 B080_3300 B080_3400 B080_3500 Allgemein: B080_3N00(N=N-ter Variant der Anweisung)	Beispiel für B080_3300: Simatic_BadNumElements_VariantInput3 Allgemein: Simatic_BadNumElements_VariantInputN (N=N-ter Variant der Anweisung)	Beispiel für falsche Anzahl Array-Elemente am 3. VARIANT-Parameter: • NamespaceIndexCount > Anzahl der Array-Elemente von NamespaceIndexes (3. VARIANT-Parameter). Beispiel für falsche Anzahl Array-Elemente am 1. VARIANT-Parameter: • NodeIDCount > Anzahl der Array-Elemente von ObjectNodeIDs (1. VARIANT-Parameter). Allgemein: Falsche Anzahl an Elementen beim N-ten VARIANT-Parameter.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für StatusList

Der Parameter StatusList enthält für jede einzelne Knoten-Id (Methode) einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	kein Fehler
8033_0000	OpcUa_BadNodeIdInvalid	Die Syntax der Knoten-Id (NodeId) ist nicht korrekt.
8034_0000	OpcUa_BadNodeIdUnknown	Die NodeId bezieht sich auf einen Knoten (eine Methode), den es im OPC UA-Server nicht gibt.
8074_0000	OpcUa_BadTypeMismatch	InputArguments / OutputArguments haben falschen Typ
Weitere Fehlercodes siehe AUTOHOTSPOT		

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_MethodGetHandleList" die folgenden Voraussetzungen erfüllt sein (vergleiche Grafik oben):

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungs-Handle von der Anweisung "OPC_UA_Connect".
- Indizes der Namenräume, in denen die Methoden liegen, die Sie aufrufen wollen.
  Sie erhalten die Indizes mit der Anweisung AUTOHOTSPOT

Funktion der Anweisung

Um Methoden-Aufrufe zu beschleunigen, verwenden OPC UA-Server numerische Referenzen (Handles) für Methoden.

Diese Handles erhalten Sie vom OPC UA-Server durch die Anweisung "OPC_UA_MethodGetHandleList".

Sie benötigen diese Handles für die OPC UA-Anweisung zum Aufruf von Methoden, für "OPC_UA_MethodCall".

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungieren soll.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "Call_OpenDoor_On_Productionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_MethodGetHandleList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_MethodGetHandleList_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_MethodGetHandleList_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", siehe "AUTOHOTSPOT".

7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Leseliste.
   Im Beispiel heisst die Methodenliste "MethodListOpenDoor".
   STEP 7 versorgt nun automatisch die meisten Parameter der Anweisung mit den richtigen Variablen.
8. Klicken Sie auf "Bausteinparameter" und weisen Sie den restlichen Parametern manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Der folgende Ausschnitt aus dem Programm "Call_OpenDoor_On_Productionine" zeigt, wie Sie die Anweisung "OPC_UA_MethodGetHandleList" verwenden.

Das vollständige Beispielprogramm finden Sie im Kapitel AUTOHOTSPOT

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Hier ist der Fall 3 dargestellt:

SCL

```
3:  // case 3, get an handle for server method OpenDoor

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodGetHandleList_Instance(
                                                REQ := #Req,
                                                ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                ObjectNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".ObjectNode
s,
                                                MethodNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodNode
s,
                                                NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                Done => #Done,
                                                Busy => #Busy,
                                                Error => #Error,
                                                Status => #Status,
                                                StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
,
                                                Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                NodeIDCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCoun
t,
                                                NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
```

```
111  3:  // case 3, get an handle for server method OpenDoor
112
113  IF #Busy = FALSE THEN
114      #Req := TRUE;
115  ELSE
116      #Req := FALSE;
117  END_IF;
118  #OPC_UA_MethodGetHandleList_Instance(
119                  REQ := #Req,
120                  ConnectionHdl := "Productionline_Configuration".Connection.ConnectionHdl,
121                  ObjectNodeIDs := "Productionline_Configuration".MethodLists."MethodListOpenDoor".ObjectNodes,
122                  MethodNodeIDs := "Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodNodes,
123                  NamespaceIndexCount := "Productionline_Configuration".Namespaces.NamespaceCount,
124                  Done => #Done,
125                  Busy => #Busy,
126                  Error => #Error,
127                  Status => #Status,
128                  StatusList := "Productionline_Data"."MethodListOpenDoor".MethodStatusList,
129                  MethodHdls := "Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
130                  Timeout := "Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
131                  NodeIDCount := "Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
132                  NamespaceIndexes := "Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
133
134
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_MethodGetHandleList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 3 dargestellt. Dieser Programmteil registriert die Methode "OpenDoor" im OPC UA-Server. Dazu wird die Methode "OPC_UA_MethodGetHandleList" aufgerufen.

Wenn die Anweisung noch nicht ausgeführt wird, ist #Busy gleich FALSE, sodass die Variable "#Req" auf den Wert TRUE gesetzt wird. Dadurch startet die Anweisung. Im nächsten Zyklus besitzt #Req den Wert FALSE.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
             IF #Done = TRUE THEN
                IF
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0] = 0
THEN      //one method
                    #State := #State + 1;
                ELSE
                    #State := 100;
                  //In case 100, set REQ of instruction "OPC_UA_Disconnect"
to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at
MethodGetHandleList';
                END_IF;
            END_IF;
            IF #Error = TRUE THEN
                #State := 100;
               //In case 100, to set REG of instruction "OPC_UA_Disconnect"
to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_MethodGetHandleList_Instance(
                                                        ObjectNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".ObjectNode
s,
                                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                        MethodNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodNode
s,
                                                        StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                        MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
,
                                                        REQ := FALSE,
                                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                        NodeIDCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCoun
t,
                                                       NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                        NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall wird am Ausgangsparameter "Error" der Wert TRUE gesetzt. Dadurch wird die "#State" auf den Wert 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung erfolgreich ausgeführt.

Es ist möglich, dass die Anweisung erfolgreich ausgeführt wurde (Parameter "Error" nicht gesetzt, Parameter "Done" gesetzt), aber für eine Methode kein Handle zurückgeliefert werden konnte.

Deshalb wird im Beispiel geprüft, ob ein Handle für die Methode "OpenDoor" zurückgeliefert wurde:

- Wenn die Bedingung erfüllt ist, dann wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 4 (Methode aufrufen) - ausgeführt. Fall 4 ist in OPC_UA_MethodCall: Methode aufrufen (Seite 3906) beschrieben.
- Wenn die Bedingung nicht erfüllt ist, dann wird dies als Fehler gewertet und #State erhält den Wert 100. Im nächsten Zyklus wird der Fehlerfall durchlaufen.

Hinweis

Wenn Sie in Ihrer Anwendung mehrere Methoden mit OPC_UA_MethodGetHandleList registrieren, dann kann es sinnvoll sein, #State immer bei #Done = TRUE um eins zu erhöhen und erst vor dem Aufruf der einzelnen Methoden durch "OPC_UA_MethodCall" (im Fall 4) zu prüfen, ob für die jeweilige Methode ein gültiges Handle zurückgeliefert wurde.

OPC_UA_MethodCall: Methode aufrufen

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_MethodCall" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Ein Programmierbeispiel zeigt Ihnen, wie Sie die Anweisung in der Programmiersprache SCL verwenden.

Dieses Kapitel geht davon aus, dass Sie für die OPC UA-Kommunikation eine Client-Schnittstelle anlegen und parametrieren, siehe: AUTOHOTSPOT. Diese effiziente Vorgehensweise sorgt dafür, dass die meisten Parameter der Client-Anweisung automatisch versorgt werden.

Sie verwenden die Anweisung "OPC_UA_MethodCall" in Ihrem Anwenderprogramm, um eine Methode auf dem Server aufzurufen.

Die folgende Grafik zeigt die Struktur eines Anwenderprogramms. Die Anweisung "OPC_UA_MethodCall" befindet sich im Bereich 2:

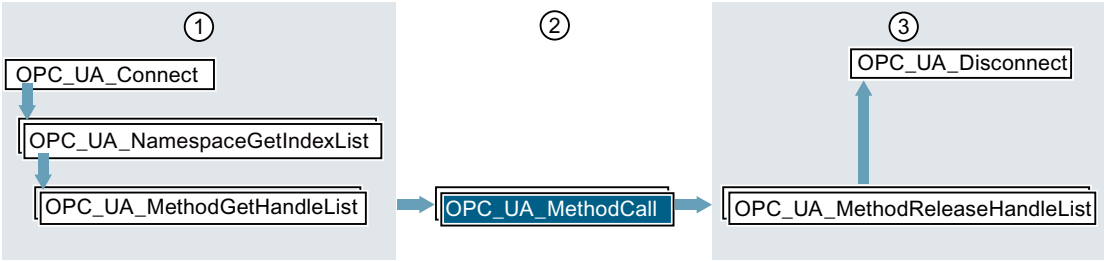

① Anweisungen zur Vorbereitung der Methodenaufrufe

② Methodenaufrufe

③ Anweisungen zum "Clean-up" nach durchgeführten Methodenaufrufen

Parameter für "OPC_UA_MethodCall"

Tabelle 4-82 Die Parameter der Anweisung "OPC_UA_MethodCall"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung AUTOHOTSPOT.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
MethodHdl	Input	DWORD	Handle für die Methode, die aufgerufen werden soll.
InputArguments	InOut	VARIANT	Zeiger auf eine Variable mit PLC-Datentyp (UDT) oder STRUCT, der in Anzahl, Reihenfolge und Datentyp den erwarteten Eingangsparametern entspricht.
OutputArguments	InOut	VARIANT	Zeiger auf eine Variable mit PLC-Datentyp (UDT) oder STRUCT, der in Anzahl, Reihenfolge und Datentyp den erwarteten Ausgangsparametern entspricht.
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"
MethodResult	InOut	VARIANT	Zeiger auf eine Variable des Typs DWORD, die den Fehlercode (Result) der aufgerufenen Server-Methode enthält. Dieser Fehlercode wird vom Anwenderprogramm des OPC UA-Servers bereitgestellt. Im Fall einer S7-1500 CPU als OPC UA-Server ist es der Parameter "UAMethod_Result" der Server-Anweisung "OPC_UA_ServerMethodPost", den Sie über die Client-Anweisung "OPC_UA_MethodCall" auslesen können. Wenn die Anweisung "OPC_UA_MethodCall" nicht erfolgreich aufgerufen wurde (Status & 0x8000_0000 ≠ 0), dann wird kein Wert für MethodResult gesetzt.

Fehlernummern für Status

Der Parameter Status informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
8006_0000	OpcUa_BadEncodingError	Fehler beim Dekodieren des Output-Arguments
8009_0000	OpcUa_BadUnknownResponse	Server antwortet nicht mit der erwarteten Anzahl an Ergebnissen
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist nicht verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) kann nicht durchgeführt werden, weil zu viele Anweisungen enthalten sind.

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
8034_0000	BadNodeIdUnknown	The node id refers to a node that does not exist in the server address space. Möglicherweise existiert der Namespace-Index der gegebenen NodeId nicht. Falls der Fehlercode beim Aufruf der Anweisung OPC_UA_MethodCall angezeigt wird: Eine weitere mögliche Ursache ist die Verwendung ungültiger ObjectIds beim vorhergehenden Aufruf von OPC_UA_MethodGetHandleList. Die Anweisung OPC_UA_MethodGetHandleList wertet die ObjectIds nicht aus.
8074_0000	OpcUa_BadTypeMismatch	Argument-Variant hat falschen Typ.
8075_0000	OpcUa_BadMethodInvalid	Zum gegebenen MethodHdl existiert keine registrierte Methode.
80AB_0000	OpcUa_BadInvalidArgument	Input-Arguments haben falschen Typ.
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	Übergreifender Fehlercode. Tritt auf, wenn ein Array zu viele Elemente hat.
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben ein Client-Interface angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_MethodCall" die folgenden Voraussetzungen erfüllt sein (vergleiche Grafik oben):

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungshandle von der Anweisung AUTOHOTSPOT.
- Indizes der Namenräume, in denen die Methoden liegen, die Sie mit dem Client aufrufen wollen.
  Sie erhalten die Indizes von der Anweisung "OPC_UA_NamespaceGetIndexList".
  Ein Handle für die Methode, die Sie in Ihrem Anwenderprogramm aufrufen wollen.
  Sie erhalten das Handle von der Anweisung AUTOHOTSPOT

Funktion der Anweisung

Sie nutzen die Anweisung "OPC_UA_MethodCall", um eine Methode aufzurufen, die in einem OPC UA-Server freigegeben ist.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungieren soll.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "Call_OpenDoor_On_Productionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_MethodCall" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_MethodCall_Instance".
5. Klicken Sie auf das Symbol für "Konfiguration starten" an der Anweisung "OPC_UA_MethodCall_Instance".
   STEP 7 öffnet das Register "Konfiguration" im Inspektorfenster.
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline".
7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Methodenliste.
   Im Beispiel heisst die Methodenliste "MethodListOpenDoor".
   STEP 7 versorgt automatisch die meisten Parameter der Anweisung mit Variablen, nicht aber die Parameter Busy, Done, Error.
8. Legen Sie lokale Variablen für die noch unversorgten Parameter an.
9. Klicken Sie auf "Bausteinparameter" und weisen Sie die lokalen Variablen den restlichen Parametern zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Der folgende Ausschnitt aus dem Programm "Call_OpenDoor_On_Productionine" zeigt, wie Sie die Anweisung "OPC_UA_MethodGetHandleList" verwenden.

Das vollständige Beispielprogramm finden Sie im Kapitel AUTOHOTSPOT

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Hier ist der Fall 4 dargestellt:

SCL

```
        4:  // case 4, call method OpenDoor

            IF #Init_Method_InputParameter = TRUE THEN
                #Init_Method_InputParameter := FALSE;
                // for our server method at Productionline, we set input
parameters to 1

"Productionline_Data".MethodListOpenDoor.Method.Inputs.Number := 1;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodCall_Instance(
                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        MethodHdl :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
[0],
                                        Status =>
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0],
                                        InputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Inputs,
                                        OutputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Outputs,
                                        REQ := #Req,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        MethodResult := #MethodResult);

        #Output_MethodResult := #MethodResult;
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_MethodCall" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 4 dargestellt. Dieser Programmteil ruft die Server-Methode "OpenDoor" im OPC UA-Server auf.

Dazu erhält die Anweisung "OPC_UA_MethodCall" das Handle für die Server-Methode sowie die Eingangsparameter.

Im Beispiel wird nur der Eingangsparameter "Number" verwendet. Der Wert wird auf 1 gesetzt, damit die Server-Methode vollständig ausgeführt wird.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
            IF #Done = TRUE THEN
                IF #MethodResult < 16#8000_0000 THEN
                    #State := #State + 1;
                ELSE
                    //server method did not run successfully
                    #State := 100;
                  //In case 100, set REQ of instruction "OPC_UA_Disconnect"
to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#' Error at server
method';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
               //In case 100, to set REQ of instruction "OPC_UA_Disconnect"
to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status :=
"Productionline_Data".MethodListOpenDoor.MethodStatusList[0];
                #OPC_UA_MethodCall_Instance(
                                            ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            MethodHdl :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
[0],
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                            Status =>
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0],
                                            InputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Inputs,
                                            OutputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Outputs,
MethodResult := #MethodResult);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung "OPC_UA_MethodCall" erfolgreich ausgeführt.

Überprüfen Sie zudem, ob auch die aufgerufene Methode im OPC UA-Server "Productionline" erfolgreich ausgeführt wurde. Das ist dann der Fall, wenn der Parameter "MethodResult" den Wert 0 enthält.

Bei erfolgreicher Ausführung wird der Wert der Variablen "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 5 - ausgeführt.

Fall 5 ist in OPC_UA_MethodReleaseHandleList: Handles für Methodenaufrufe freigeben (Seite 3914) beschrieben.

Im Fehlerfall der Anweisung "OPC_UA_MethodCall" wird am Ausgangsparameter "Error" der Wert TRUE angezeigt. Dadurch wird der Wert der Variable "#State" auf 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen.

OPC_UA_MethodReleaseHandleList: Handles für Methodenaufrufe freigeben

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_MethodReleaseHandleList" gilt für S7-1500-CPUs ab Firmware-Version V2.6.

Beschreibung

Ein Programmierbeispiel zeigt Ihnen, wie Sie die Anweisung in der Programmiersprache SCL manuell verwenden.

Dieses Kapitel geht davon aus, dass Sie für die OPC UA-Kommunikation eine Client-Schnittstelle anlegen und parametrieren, siehe: AUTOHOTSPOT. Diese effiziente Vorgehensweise sorgt dafür, dass die meisten Parameter der Client-Anweisung automatisch versorgt werden.

Sie verwenden die Anweisung "OPC_UA_MethodReleaseHandleList" in Ihrem Anwenderprogramm, um numerische Referenzen (Handles) für Methoden frei zu geben.

OPC_UA_MethodReleaseHandleList dient dazu, Methoden-Handles frei zu geben, siehe ③ in der folgenden Grafik.

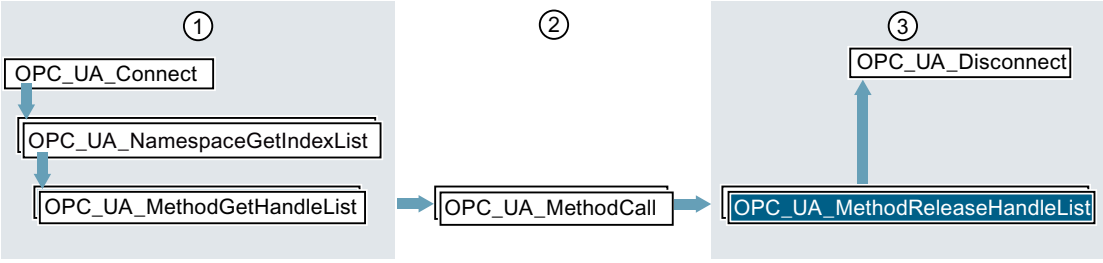

① Anweisungen zur Vorbereitung der Methodenaufrufe

② Methodenaufrufe

③ Anweisungen zum "Clean-up" nach durchgeführten Methodenaufrufen

Parameter für "OPC_UA_MethodReleaseHandleList"

Tabelle 4-83 Die Parameter der Anweisung "OPC_UA_MethodReleaseHandleList"

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
REQ	Input	BOOL	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung der Anweisung aus.
ConnectionHdl	Input	DWORD	Eindeutige Kennzeichnung einer aufgebauten Verbindung. Das Handle erhalten Sie von der Anweisung OPC_UA_Connect.
Timeout	Input	TIME	Maximale Zeitspanne für die Ausführung der Anweisung in Millisekunden. Siehe auch Erläuterung zu diesem Parameter bei AUTOHOTSPOT
MethodHdlCount	Input	UINT	Anzahl der Elemente im Parameter MethodHdls
MethodHdls	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Handles für die einzelnen Methoden (NodeIds), die freigegeben werden sollen.
StatusList	InOut	VARIANT	Zeiger auf ein Array des Typs DWORD. Das Array enthält die Fehlercodes für die einzelnen Methoden, siehe unten "Fehlernummern für NodeStatusList". Für jede Methode ist angegeben, ob ihr ein Handle zugeordnet werden konnte.
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung

Parameter	Deklaration im Bereich	Datentyp	Bedeutung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlernummern für Status"

Fehlernummern für Status

In der folgenden Tabelle finden Sie eine Zusammenstellung von Fehlercodes für diese Anweisung:

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	Anweisung erfolgreich beendet.
8003_0000	OpcUa_BadOutOfMemory	Kein Speicher für den OPC UA Client verfügbar. Da sich OPC UA Client und OPC UA Server einen Speicherbereich teilen, sollten Sie den Speicherbedarf des Servers reduzieren. Dazu gibt es die folgenden Möglichkeiten: • Weniger PLC-Variablen für OPC UA frei geben. • Anzahl der OPC UA-Clients reduzieren, die aktuell mit dem Server verbunden sind. • Weniger Subscriptions einrichten.
800A_0000	OpcUa_BadTimeout	Ein Netzwerk-Timeout ist aufgetreten. Mögliche Gründe sind: • Verbindung zum OPC UA-Server ist zu langsam (zu geringe Kapazität) • Netzwerklast ist zu hoch • Der OPC UA-Server ist verfügbar Mögliche Abhilfe: • Überprüfen Sie die URL des OPC UA-Servers • Erhöhen Sie die Timeout-Zeitspanne (höherer Wert für den Parameter Timeout des Funktionsbausteins OPC_UA_Connect).
800D_0000	OpcUa_BadServerNotConnected	Der Server ist nicht verbunden oder das Verbindungshandle ist falsch oder ungültig.
800F_0000	OpcUa_BadNothingToDo	Nichts zu tun: Der OPC UA-Server erhielt vom OPC UA-Client eine leere Liste mit keinen Anweisungen.
8010_0000	OpcUa_BadTooManyOperations	Die Anfrage (Request) konnte nicht durchgeführt werden, weil zu viele Anweisungen enthalten waren.

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
B080_0100	Simatic_BadType_VariantInput1	Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt.
B080_0200	Simatic_BadType_VariantInput2	Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_1100	Simatic_ArrayElements_TooMany	MethodHdlCount > MAX_ELEMENTS_METHOD
B080_C400	Simatic_ClientNotEnabled	Der OPC UA-Client ist deaktiviert.
B080_C500	Simatic_NothingToDo	Nichts zu tun: Die Anweisung verwendet eine Liste, die keine Elemente enthält.
C080_C300	Simatic_OutOfResources	Die Ressourcen der CPU sind erschöpft. Mögliche Abhilfe: • Reduzieren Sie die Anzahl der offenen Verbindungen
Weitere Fehlercodes siehe AUTOHOTSPOT		

Fehlernummern für "StatusList"

Der Parameter "StatusList" enthält für jedes einzelne Handle einen Fehlercode.

Die folgende Tabelle erläutert die Fehlercodes:

Fehlercode (hex. Werte)	Name des Fehlers	Erläuterung
0000_0000	OpcUa_Good	kein Fehler
8034_0000	OpcUa_BadNodeIdUnknown	Das übergebene Method-Handle ist nicht bekannt.
Weitere Fehlercodes siehe AUTOHOTSPOT		

Voraussetzungen

Die folgende Beschreibung setzt voraus:

- Sie haben eine Client-Schnittstelle angelegt, siehe "AUTOHOTSPOT"
- Sie haben eine Verbindung zu einem OPC UA-Server angelegt und parametriert, siehe "AUTOHOTSPOT".

Zudem müssen für die Anweisung "OPC_UA_MethodReleaseHandleList" die folgenden Voraussetzungen erfüllt sein (vergleiche Grafik oben):

- Handle für eine Verbindung zu einem OPC UA-Server ist vorhanden.
  Sie erhalten das Verbindungshandle von der Anweisung OPC_UA_Connect.
- Eine Liste mit Methoden-Handles, die Sie in Ihrem Anwenderprogramm nicht länger aufrufen wollen.
  Sie erhalten diese Handle-Liste von der Anweisung OPC_UA_MethodGetHandleList.
- Optional (hier nicht benötigt): Indizes der Namenräume, in denen die Methoden liegen.
  Sie erhalten die Indizes von der Anweisung OPC_UA_NamespaceGetIndexList..

Funktion der Anweisung

Sie rufen die Anweisung "OPC_UA_MethodReleaseHandleList" auf, um Handles für Methoden frei zu geben, die Sie nicht mehr in Ihrem Programm benötigen.

So verwenden Sie eine parametrierte Verbindung

1. Wählen Sie im Bereich "Projektnavigation" die CPU aus, die als Client fungieren soll.
2. Wählen Sie im Ordner "Programmbausteine" den Funktionsbaustein aus, der die Client-Anweisung ausführen soll.
   Im Beispiel heisst der Funktionsbaustein "Call_OpenDoor_On_Productionline".
   Gewählte Sprache: SCL.
3. Ziehen Sie per Drag & Drop aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Client" die Anweisung "OPC_UA_MethodReleaseHandleList" in den Editor.
4. Wählen Sie den Aufruf als Multi-Instanz.
   STEP 7 erzeugt eine Instanz dieser Anweisung und nennt sie "OPC_UA_MethodReleaseHandleList_Instance".
5. Klicken Sie auf "OPC_UA_MethodReleaseHandleList_Instance".
   STEP 7 öffnet die "Konfiguration".
6. Unter "Client-Schnittstelle für OPC UA-Schnittstelle auswählen" wählen Sie die Client-Schnittstelle aus, die Sie für die Anweisung verwenden wollen.
   Im Beispiel heisst die Client-Schnittstelle "Productionline", die für die Beispielanlage angelegt wurde, siehe "AUTOHOTSPOT".
7. Klicken Sie auf "Dateizugriff" und wählen Sie eine Methodenliste.
   Im Beispiel heisst die Methodenliste "MethodListOpenDoor".
   STEP 7 versorgt automatisch die meisten Parameter der Anweisung mit Variablen, nicht aber die Parameter Busy, Done, Error.
8. Legen Sie lokale Variablen für die noch unversorgten Parameter an.
9. Klicken Sie auf "Bausteinparameter" und weisen Sie den restlichen Parametern manuell Variablen zu.
   STEP 7 fügt die gewählte Variable in den Funktionsausruf ein.

Aufruf der Anweisung (Erstaufruf)

Der folgende Ausschnitt aus dem Programm "Call_OpenDoor_On_Productionine" zeigt, wie Sie die Anweisung "OPC_UA_MethodReleaseHandleList" verwenden.

Das vollständige Beispielprogramm finden Sie im Kapitel "Programmbeispiel zum Aufruf von Server-Methoden".

Das Beispielprogramm ist durch eine CASE-Anweisung in mehrere Teilbereiche (Cases) unterteilt.

Hier ist der Fall 5 dargestellt:

SCL

```
 5:  // case 5, release method handle

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodReleaseHandleList_Instance(
                                                    REQ := #Req,
                                                    ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                    MethodHdlCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCoun
t,
                                                    MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
,
                                                    Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                    Done => #Done,
                                                    Busy => #Busy,
                                                    Error => #Error,
                                                    Status => #Status,
                                                    StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList);
```

```
221  5:  // case 5, release method handle
222
223  IF #Busy = FALSE THEN
224      #Req := TRUE;
225  ELSE
226      #Req := FALSE;
227  END_IF;
228  #OPC_UA_MethodReleaseHandleList_Instance(
229              REQ := #Req,
230              ConnectionHdl := "Productionline_Configuration".Connection.ConnectionHdl,
231              MethodHdlCount := "Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
232              MethodHdls := "Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
233              Timeout := "Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
234              Done => #Done,
235              Busy => #Busy,
236              Error => #Error,
237              Status => #Status,
238              StatusList := "Productionline_Data"."MethodListOpenDoor".MethodStatusList);
239
```

Asynchrone Ausführung

Die Anweisung "OPC_UA_MethodReleaseHandleList" läuft asynchron zum Anwenderprogramm und benötigt mehrere Programmzyklen.

Den aktuellen Stand der Ausführung kontrollieren Sie über die Parameter "Busy", "Done", "Error" und "Status".

Die asynchrone Programmausführung ist im Kapitel "Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)" beschrieben.

Erläuterung des Aufrufs der Anweisung (Erstaufruf)

Im Bild oben ist Fall 5 dargestellt. Dieser Programmausschnitt gibt das Handle für eine Server-Methode wieder frei.

Im Beispiel wird dem Parameter "MethodHdls" das Handle für die Server-Methode "OpenDoor" übergeben.

Wenn dieses Handle nicht freigegeben werden kann, dann enthält der Parameter "StatusList" einen Fehlercode.

Aufruf der Anweisung (Fehlerbehandlung)

Das folgende Bild zeigt die Auswertung der Parameter "Done" und "Error".

SCL

```
            IF #Done = TRUE THEN
                #Mem_Status := #Status;
                IF
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction
"OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at
MethodGetHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
               //In case 100, to set REG of instruction "OPC_UA_Disconnect"
to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_MethodReleaseHandleList_Instance(
                                                        MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls
,
                                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                        StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                       MethodHdlCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCoun
t,
                                                        REQ := FALSE,
                                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;
```

Erläuterung des Aufrufs der Anweisung (Fehlerbehandlung)

Im Fehlerfall der Anweisung "OPC_UA_MethodReleaseHandleList" wird am Ausgangsparameter "Error" der Wert TRUE angezeigt. Dadurch wird der Wert der Variable "#State" auf 100 gesetzt. Dieser Fall ist für die Fehlerbehandlung vorgesehen. Zudem ruft das Beispielprogramm die Anweisung "OPC_UA_MethodReleaseHandleList" auf, um den Parameter REQ auf FALSE zu setzen.

Wenn im Ausgangsparameter "Done" der Wert TRUE steht, dann wurde die Anweisung "OPC_UA_MethodReleaseHandleList" erfolgreich ausgeführt.

Im Beispiel wird zusätzlich überprüft, ob auch das Handle für die Server-Methode "OpenDoor" freigegeben werden konnte.

Bei erfolgreicher Ausführung erhöht wird der Wert der Variable "#State" um eins erhöht. Dadurch wird im nächsten Zyklus der nächste Programmabschnitt - Fall 6 - ausgeführt.

Fall 6 ist in OPC_UA_Disconnect (Seite 3883) beschrieben.

Hinweis

Fall 6 dient dazu, die Verbindung zum OPC UA-Server freizugeben. Damit sind auch alle anderen Ressourcen freigegeben, die das Client-Programm im Server belegt hat. Wenn Sie die Verbindung zum Server beenden wollten, dann kann Ihr Anwenderprogramm unmittelbar Fall 6 ausführen und Fall 5 überspringen.

Systemdatentypen für OPC UA Client-Anweisungen

OPC_UA_SessionConnectInfo

OPC_UA_SessionConnectInfo

Entnehmen Sie der folgenden Tabelle die Bedeutung der Verbindungsinformationen für den Parameter "SessionConnectInfo" der Anweisung "OPC_UA_Connect".

Tabelle 4-84 Parameter des Systemdatentyps "OPC_UA_SessionConnectInfo"

Parameter	S7-Datentyp	Bedeutung
SessionName	WSTRING[64]	Name der Sitzung, den Sie hier eintragen können. Der Parameter kann leer sein. Falls Sie keinen Namen eintragen, vergibt der OPC UA-Server einen Namen. Der Sitzungsname wird z. B. für die Identifikation der Verbindung zu Diagnosezwecken verwendet.
ApplicationName	WSTRING[64]	Name des OPC UA-Clients. Der Applikationsname ist eine Komponente der ClientDescription; welche wiederum eine Komponente der Sitzung ist (siehe SessionName). Es ist nicht möglich, einen eindeutigen ApplicationName für jede Sitzung zu setzen. Der STRING **muss** leer sein, andernfalls wird der Fehlercode "Simatic_BadValue_VariantInput2 (0xB080_0200)" zurückgeliefert. Als ApplicationName wird bei SIMATIC der projektierte Applikationsname verwendet (CPU-Parameter im Bereich OPC UA).
SecurityMsgMode	UDINT	Security-Verfahren • 0 = Bestmögliches Verfahren. Achtung: Diese Einstellung ist nicht zulässig! • 1 = None • 2 = Sign • 3 = Sign&Encrypt

Parameter	S7-Datentyp	Bedeutung
SecurityPolicy	UDINT	Security-Profil • 0 = Bestmögliches Security-Profil. Achtung: Diese Einstellung ist nicht zulässig! • 1 = Kein Security-Profil • 2 = Basic128Rsa15 • 3 = Basic256 • 4 = Basic256Sha256
ServerUri	STRING[254]	Nur relevant für Verbindungen zu einem OPC UA-Server über einen Gateway-Server. Die aktuelle Firmware der CPU unterstützt keine Gateway-Server. Deshalb muss dieser Parameter mit einem Leerstring versorgt sein.
CheckServerCertificate	BOOL	Wenn das Bit gesetzt ist, prüft der Client das vom Server übermittelte Zertifikat gegen die projektierten vertrauenswürdigen Zertifikate. Zur Liste der vertrauenswürdigen Zertifikate gelangen Sie über die Eigenschaften der Client-CPU > Bereich "Schutz & Security" > Zertifikatsmanager. Dort befindet sich die Tabelle "Zertifikate der Partnergeräte", die Sie befüllen müssen. Voraussetzung für eine Befüllung ist, dass Sie die erforderlichen Zertifikate über den globalen Zertifikatsmanager importiert haben (Projektnavigation > Zertifikatsmanager > Register "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen)"). Wenn das Bit nicht gesetzt ist, prüft der Client das Server-Zertifikat nicht.
TransportProfile	UDINT	1 = UATP_UATcp Nur dieses Transportprofil gemäß der PLCopen-Spezifikation wird unterstützt.
UserIdentityTokenType	UDINT	Daten der Benutzerauthentifizierung siehe Erläuterung unterhalb dieser Tabelle.
UserTokenParam1	WSTRING[64]	Die Bedeutung ist abhängig vom UserIdentityTokenType (z. B. Benutzername).
UserTokenParam2	WSTRING[64]	Die Bedeutung ist abhängig vom UserIdentityTokenType (z. B. Passwort).

Parameter	S7-Datentyp	Bedeutung
CertificateID	UDINT	Die ID des Zertifikats, das Sie für den OPC UA-Client verwenden. Die ID finden Sie in STEP 7 (TIA Portal) unter "Globale Security-Einstellungen" > "Zertifikatsmanager" > "Gerätezertifikate" und auch in den CPU-Eigenschaften (Bereich "Schutz & Security" > "Zertifikatsmanager"). Dort befindet sich die Tabelle "Gerätezertifikate" mit den IDs der Zertifikate. Um ein Zertifikat für den Client zu erzeugen, gehen Sie folgendermaßen vor: 1. Wählen Sie im TIA Portal die Eigenschaften der CPU aus, die als Client genutzt wird. 2. Wenn für Sie ein selbstsigniertes Zertifikat ausreichend ist, dann überspringen Sie diesen Schritt. Wenn Sie ein Zertifikat signiert von einer CA nutzen wollen, dann aktivieren Sie unter "Schutz & Security" > "Zertifikatsmanager" > "Globale Security-Einstellungen" die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". 3. Navigieren Sie zur Tabelle "Gerätezertifikate" und doppelklicken Sie auf ein leeres Feld der Spalte "Zertifikatsinhaber". Ein Zertifikatssymbol erscheint am linken Rand der Zeile. 4. Klicken Sie erneut auf das Feld "Zertifikatsinhaber". Eine Klappliste öffnet sich und zeigt die bereits vorhandenen Gerätezertifikate. 5. Klicken Sie auf die Schaltfläche "Hinzufügen". Ein Dialog mit Einstellungsmöglichkeiten zum Zertifikat wird angezeigt. 6. Wählen Sie "OPC UA-Client" oder "OPC UA-Client & Server" als Verwendungszweck und dann auf "OK".
SessionTimeout	TIME	Maximale Zeitspanne, die eine Sitzung aufrechterhalten bleibt, nachdem die Verbindung zum OPC UA-Server unterbrochen wurde (in Millisekunden). Voreinstellung: 20000 (20 Sekunden)
MonitorConnection	TIME	Verbindungsüberwachungszeit (in Millisekunden) Zeitintervall zur Überprüfung einer Verbindung, über die aktuell keine Daten übertragen werden. Voreinstellung: 5000 (5 Sekunden)
LocaleIDs	ARRAY[1..5] of STRING[6]	Optionale Sprach- und Regionenkennung gemäß RFC 3066. 0 = keine oder unbekannte LocaleId.

Parameter "UserIdentityTokenType"

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern "UserIdentityTokenType", "UserTokenParam1" und "UserTokenParam2" des Systemdatentyps "UASessionConnectInfo":

UserIdentityTokenType	UserTokenParam1	UserTokenParam2	Beschreibung
0: anonymer Zugang	ignoriert	ignoriert	Gast-Account, ohne Authentifizierung
1: Zugang mit Anwendername und Passwort	Benutzername	Passwort	Das Passwort wird im entsprechenden Globaldatenbaustein oder Instanzdatenbaustein im Klartext gespeichert und kann ausgelesen werden. Empfehlung: Verwenden Sie stattdessen den Typ SIMATIC-User.
42: SIMATIC-User	Name eines Benutzers, der mit STEP 7 projektiert wurde.	ignoriert	Das Passwort laden Sie bei einem SIMATIC-User zusammen mit den übrigen Benutzerdaten und den Konfigurationsdaten in die Client-CPU. Auf diese Weise ist das Passwort vor Zugriffen von außen geschützt. Dem Benutzer (SIMATIC-User) müssen Sie in der Projektnavigation (Security-Einstellungen > Benutzer und Rollen) das Funktionsrecht "Benutzer-Authentifizierung des OPC UA-Clients" zugewiesen haben! Wenn Sie die Einstellungen für die Benutzer-Authentifizierung in der OPC UA-Client-Schnittstelle vornehmen, dann wählen Sie die Option „Benutzer (TIA Portal – Security-Einstellungen)“ und tragen einen bereits projektierten Benutzernamen ein.

Siehe auch

OPC_UA_Connect: Verbindung erzeugen (Seite 3799)

OPC_UA_NodeId

OPC_UA_NodeId

Entnehmen Sie der folgenden Tabelle die Bedeutung der Parameter von "OPC_UA_NodeId" zur Identifizierung des Ziel-Knotens im OPC UA-Server. "OPC_UA_NodeId" versorgt den Parameter "NodeIDs" der Anweisung "OPC_UA_NodeGetHandleList".

Tabelle 4-85 Parameter des Systemdatentyps "OPC_UA_NodeId"

Parameter	S7-Datentyp	Bedeutung
NamespaceIndex	UINT	Namensraumindex des Knotens im OPC UA-Server. Ein Knoten kann zum Beispiel ein Objekt oder eine Variable sein.
Identifier	WSTRING[254]	Die Bezeichnung für den Knoten (Objekt oder Variable) ist abhängig vom Identifier-Typ: • Numerischer Identifier: Der Knoten wird mit einer Zahl bezeichnet, zum Beispiel "12345678" (STRING mit konvertiertem DWORD). • String-Identifier: Der Knoten wird mit einem Namen bezeichnet, zum Beispiel "MeineVariable". Groß- und Kleinschreibung wird unterschieden. • Zu den Bezeichnungen von Knoten vom Typ "GUID" und "Opaque" siehe unten.
IdentifierType	UDINT	Typ des Identifiers • 0: Numerischer Identifier • 1: String-Identifier • 2: GUID • 3: Opaque

Identifier vom IdentifierType "GUID"

Format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}

Beispiel für die Eingabe des Formats von IdentifierType GUID (2) in STEP 7:
`WSTRING#'5ce9dbce-5d79-434c-9ac3-1cfba9a6e92c'`

Identifier vom IdentifierType "Opaque"

Opaque Identifier sind Identifier mit scheinbar zufällig angeordneter Reihenfolge von Zeichen ohne unmittelbar erkennbare Semantik.

Bei OPC UA können Sie einen Identifier vom Typ Opaque erzeugen, indem Sie ein beliebiges Byte-Array definierter Länge (z. B. ASCII-Zeichenfolge 'abcd' bzw. das HEX-Pendant (0x61626364) Base64 kodieren. Für die Base64-Kodierung werden zahlreiche Tools angeboten.

Beispiel für das Format von IdentifierType Opaque (3) in STEP 7 nach Kodierung des HEX-Wertes:
`WSTRING#'YWJjZA=='`

Der Server zeigt den unkodierten Wert des Identifiers, hier also den HEX-Wert an.

Siehe auch

OPC_UA_NodeGetHandleList:Handles für Lese- und Schreibzugriffe holen (Seite 3819)

OPC_UA_NodeAdditionalInfo

OPC_UA_NodeAdditionalInfo

Spezifiziert Attribut-Typen und Indexbereich von Knoten für den Parameter "NodeAddInfos" der Anweisungen "OPC_UA_ReadList" und "OPC_UA_WriteList".

- Der Datentyp wird für skalare Werte und eindimensionale Arrays verwendet.
  Um einen Teilbereich eines mehrdimensionalen Arrays zu lesen (bis zu sechs Dimensionen, (Matrix)), verwenden Sie den Datentyp "OPC_UA_NodeAdditionalInfoExt". Beachten Sie die Hinweise zum Mapping von Datentypen.
- Der Startindex muss größer oder gleich "0" sein und kleiner oder gleich Endindex.
- Einstellungen für das vollständiges Lesen von Leselisten bestehend aus Skalaren und Arrays:
  Wenn sowohl Startindex als auch Endindex den Wert 4_294_967_295 haben (UDINT_MAX), dann ignoriert der Server den Wert für Startindex und Endindex und liest Skalare und Arrays vollständig, siehe Array-Bereich lesen mit OPC_UA_ReadList (Seite 3854).

Tabelle 4-86 Parameter des Systemdatentyps "OPC_UA_NodeAdditionalInfo"

Parameter	Datentyp	Bedeutung
AttributeId	UDINT	Gibt an, welches Attribut von einem Knoten (einer Variable) gelesen werden soll. Die IDs der Attribute sind im OPC UA-Aufzählungstyp "UAAttributeId" festgelegt. Beispiele: • 5: Description (Beschreibung) • 13: Value (Wert) • 14: DataType (Datentyp) • 15: ValueRank (Anzahl der Dimensionen im Attribute Value) • 16: ArrayDimensions (Länge einer Dimension, Anzahl der Elemente) Vorgabewert ist 13 (Value).
StartIndex	UDINT	Bei AttributeId = 13: Der Startindex eines Arrays. Ab diesem Index werden Werte gelesen.
EndIndex	UDINT	Bei AttributeId = 13: Der Endindex eines Arrays. Bis zu diesem Index werden Werte gelesen.

Siehe auch

OPC_UA_NodeAdditionalInfoExt (Seite 3928)

OPC_UA_NodeAdditionalInfoExt

OPC_UA_NodeAdditionalInfoExt

Spezifiziert das Attribut des Knotens und den Indexbereich für den Parameter "NodeAddInfos" (Anweisungen "OPC_UA_ReadList" und "OPC_UA_WriteList").

Tabelle 4-87 Parameter des Systemdatentyps "OPC_UA_NodeAdditionalInfoExt"

Parameter	S7-Datentyp	Bedeutung
AttributeID	UDINT	Gibt an, welches Attribut eines Knotens (einer Variable) gelesen oder geschrieben werden soll. Vorgabewert ist 13 (Value).
IndexRangeCount	UDINT	Anzahl der folgenden Index-Bereiche. Es müssen nicht alle sechs Index-Bereiche angegeben werden.
StartIndex1	UDINT	Der Index, ab dem Werte für Array-Dimension 1 gelesen oder geschrieben werden sollen.
EndIndex1	UDINT	Der Index, bis zu dem Werte für Array-Dimension 1 gelesen oder geschrieben werden sollen.
StartIndex2	UDINT	Der Index, ab dem Werte für Array-Dimension 2 gelesen oder geschrieben werden sollen.
EndIndex2	UDINT	Der Index, bis zu dem Werte für Array-Dimension 2 gelesen oder geschrieben werden sollen.
StartIndex3	UDINT	Der Index, ab dem Werte für Array-Dimension 3 gelesen oder geschrieben werden sollen.
EndIndex3	UDINT	Der Index, bis zu dem Werte für Array-Dimension 3 gelesen oder geschrieben werden sollen.
StartIndex4	UDINT	Der Index, ab dem Werte für Array-Dimension 4 gelesen oder geschrieben werden sollen.
EndIndex4	UDINT	Der Index, bis zu dem Werte für Array-Dimension 4 gelesen oder geschrieben werden sollen..
StartIndex5	UDINT	Der Index, ab dem Werte für Array-Dimension 5 gelesen oder geschrieben werden sollen.
EndIndex5	UDINT	Der Index, bis zu dem Werte für Array-Dimension 5 gelesen oder geschrieben werden sollen.
StartIndex6	UDINT	Der Index, ab dem Werte für Array-Dimension 6 gelesen oder geschrieben werden sollen.
EndIndex6	UDINT	Der Index, bis zu dem Werte für Array-Dimension 6 gelesen oder geschrieben werden sollen.

Siehe auch

OPC_UA_NodeAdditionalInfo (Seite 3927)

OPC_UA_BrowsePath

OPC_UA_BrowsePath

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_BrowsePath":

Tabelle 4-88 Parameter des Systemdatentyps "OPC_UA_BrowsePath"

Name	Datentyp	Bedeutung
StartingNode	OPC_UA_NodeId	NodeId des Knotens (oder der Variablen), an dem dieser Pfad beginnt. Von diesem Knoten aus soll gebrowst werden.
RelativePath	OPC_UA_RelativePath	Der relative Pfad, an dem entlang gebrowst werden soll, siehe OPC_UA_RelativePath.

OPC_UA_RelativePath

OPC_UA_RelativePath

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_RelativePath":

Tabelle 4-89 Parameter des Systemdatentyps "OPC_UA_RelativePath"

Name	Datentyp	Bedeutung
NoOfElements	UINT	Anzahl der Elemente in Elements
Elements	Array [4] of OPC_UA_RelativePathElement	Array mit bis zu vier Elementen des Typs OPC_UA_RelativePathElement (Seite 3930)

OPC_UA_RelativePathElement

OPC_UA_RelativePathElement

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_RelativePathElement":

Tabelle 4-90 Parameter des Systemdatentyps "OPC_UA_RelativePathElement"

Name	Datentyp	Bedeutung
ReferenceTypeId	OPC_UA_NodeId	Der Referenz-Typ, dem gefolgt werden soll in diesem RelativePathElement. Voreingestellt ist die ReferenceTypeId für hierarchische Referenzen.
IsInverse	BOOL	Wenn TRUE, dann wird der inversen Richtung gefolgt. Vorgabewert ist FALSE.
IncludeSubtypes	BOOL	Wenn TRUE, dann wird auch den Reference-Typen gefolgt, die vom Typ abgeleitet sind, der unter ReferenceTypeId angegeben ist. Vorgabewert ist TRUE.
TargetName	OPC_UA_QualifiedName	Siehe Systemdatentyp OPC_UA_Qualified-Name

OPC_UA_QualifiedName

OPC_UA_QualifiedName

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_QualifiedName":

Tabelle 4-91 Parameter des Systemdatentyps "OPC_UA_QualifiedName"

Name	Datentyp	Bedeutung
NamespaceIndex	UINT	Der Namespaceindex des Namens.
Name	WSTRING[64]	BrowseName des Knotens (oder der Variablen).

Fehlercodes

Fehlercodes von Siemens

SIMATIC-Statuscodes für OPC UA-Anweisungen

In der folgenden Tabelle finden Sie die Fehlercodes von Siemens für die OPC UA-Client-Anweisungen:

Statuscode	Bedeutung
B080_0100	BadType_VariantInput1: Der Datentyp der Variablen stimmt nicht, auf die der 1. Variant-Parameter der Anweisung zeigt. **Ein Beispiel:** Bei der Anweisung "OPC_UA_Connect" ist der Parameter "ServerEndpointUrl" der 1. Variant-Parameter. Die Reihenfolge von VariantInput1 bis VariantInput5 entspricht der Reihenfolge der Variant-Parameter in der Dokumentation zur jeweiligen OPC UA-Anweisung. Wenn der 1. Variant-Parameter auf eine Variable zeigt, die den falschen Datentyp besitzt, dann wird der Statuscode "B080_0100" zurückgeliefert. Der richtige Datentyp ist in der Tabelle "Parameter" der jeweiligen Anweisung aufgeführt. Im Beispiel entnehmen Sie den Datentyp der Tabelle 'Parameter für "OPC_UA_Connect"' im Kapitel zur Anweisung "OPC_UA_Connect (Seite 3799)".
B080_0200	BadType_VariantInput2. Der Datentyp der Variablen stimmt nicht, auf die der 2. Variant-Parameter der Anweisung zeigt.
B080_0300	BadType_VariantInput3: Der Datentyp der Variablen stimmt nicht, auf die der 3. Variant-Parameter der Anweisung zeigt.
B080_0400	BadType_VariantInput4: Der Datentyp der Variablen stimmt nicht, auf die der 4. Variant-Parameter der Anweisung zeigt.
B080_0500	BadType_VariantInput5: Der Datentyp der Variablen stimmt nicht, auf die der 5. Variant-Parameter der Anweisung zeigt.
B080_3100	BadNumElements_VariantInput1: Anzahl Array-Elemente im Variant 1 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist (z. B. NodeID-Count bei OPC_UA_NodeGetHandleList oder NamespaceUris-Count bei OPC_UA_NamespaceGetIndexList etc.).
B080_3200	BadNumElements_VariantInput2: Anzahl Array-Elemente im Variant 2 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.

Statuscode	Bedeutung
B080_3300	BadNumElements_VariantInput3: Anzahl Array-Elemente im Variant 3 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.
B080_3400	BadNumElements_VariantInput4: Anzahl Array-Elemente im Variant 4 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.
B080_3500	BadNumElements_VariantInput5: Anzahl Array-Elemente im Variant 5 ist kleiner als die Anzahl, die im Count-Parameter der Anweisung hinterlegt ist.
B080_C400	Simatic_ClientNotEnabled: Der OPC UA-Client ist deaktiviert.
B080_C500	SimaticNothingToDo: Fehler beim Initialisieren des Clients
B080_C600	ClientNotAvailable Fehler beim Initialisieren des Clients
B080_0B00	ArrayElements_TooMany: Das Array besitzt zu viele Elemente.
B080_1100	ArrayElements_TooMany: Übergreifender Fehlercode. Tritt auf, wenn ein Array zu viele Elemente hat: NamespaceUrisCount > MAX_ELEMENTS_NAMESPACES NodeIDCount > MAX_ELEMENTS_NODELIST BrowsePathscount > MAX_ELEMENTS_RELATIVEPATH MethodHdlCount > MAX_ELEMENTS_METHODLIST MAX_ELEMENTS_NAMESPACES = 20 MAX_ELEMENTS_NODELIST = 300 MAX_ELEMENTS_METHODLIST = 100 MAX_ELEMENTS_BROWSEPATH = 10
B080_1200	ArrayElements_TooFew
B080_2100	BadValue_VariantInput1: Der Wert der Variablen stimmt nicht, auf die der 1. Variant der Anweisung zeigt.
B080_2200	BadValue_VariantInput2: Der Wert der Variablen stimmt nicht, auf die der 2. Variant der Anweisung zeigt.
B080_2300	BadValue_VariantInput3: Der Wert der Variablen stimmt nicht, auf die der 3. Variant der Anweisung zeigt.
B080_2400	BadValue_VariantInput4: Der Wert der Variablen stimmt nicht, auf die der 4. Variant der Anweisung zeigt.
B080_2500	BadValue_VariantInput5: Der Wert der Variablen stimmt nicht, auf die der 5. Variant der Anweisung zeigt.

Statuscode	Bedeutung
B080_B000	TooManyMethods: Max. Anzahl Server-Methoden bzw. max. Anzahl Server-Methoden-Instanzen überschritten. (Aufrufe der Anweisungen OPC_UA_ServerMethodPre, OPC_UA_ServerMethodPost): Für S7-1510 bis S7-1513 ...: max. 20 Für S7-1515, S7-1516...: max. 50 Für S7-1517, S7-1518, S7-1507S...: max. 100
C080_C300	InsufficientResources: Keine ausreichenden Ressourcen.Entweder (a) Fehler bei Speicherallokation oder (b) zu viele SFB-Instanzen

Fehlercodes der OPC Foundation

Fehlercodes

Die folgende Tabelle enthält Fehlercodes der OPC Foundation.

Die Namen und Erläuterungen der Fehler sind im Original (Englisch) aufgeführt und punktuell mit weiteren Informationen ergänzt.

Fehlercodes

Status "Good" (0000_0000 - 3FFF_FFFF)		
Fehlernummer (hex)	Name	Bedeutung
0000_0000	Good	Success, no error.
002D_0000	GoodSubscriptionTransferred	The subscription was transferred to another session.
002E_0000	GoodCompletesAsynchronously	The processing will complete asynchronously.
002F_0000	GoodOverload	Sampling has slowed down due to resource limitations.
0030_0000	GoodClamped	The value written was accepted but was clamped.
0096_0000	GoodLocalOverride	The value has been overridden.
00A2_0000	GoodEntryInserted	The data or event was successfully inserted into the historical database.
00A3_0000	GoodEntryReplaced	The data or event field was successfully replaced in the historical database.
00A5_0000	GoodNoData	No data exists for the requested time range or event filter.
00A6_0000	GoodMoreData	The data or event field was successfully replaced in the historical database.
00A7_0000	GoodCommunicationEvent	The communication layer has raised an event.
00A8_0000	GoodShutdownEvent	The system is shutting down.
00A9_0000	GoodCallAgain	The operation is not finished and needs to be called again.
00AA_0000	GoodNonCriticalTimeout	A non-critical timeout occurred.
00BA_0000	GoodResultsMayBeIncomplete	The server should have followed a reference to a node in a remote server but did not. The result set may be incomplete.

Status "Good" (0000_0000 - 3FFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
00D9_0000	GoodDataIgnored	The request pecifies fields which are not valid for the EventType or cannot be saved by the historian.
00DC_0000	GoodEdited	The value does not come from the real source and has been edited by the server.
00DD_0000	GoodPostActionFailed	There was an error in execution of these post-actions.
00E0_0000	GoodDependentValueChanged	A dependent value has been changed but the change has not been applied to the device.

Status "Uncertain" (4000_0000 - 7FFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
406C_0000	UncertainReferenceOutOfServer	One of the references to follow in the relative path references to a node in the address space in another server.
408F_0000	UncertainNoCommunicationLastUsableValue	Communication to the data source has failed. The variable value is the last value that had a good quality.
4090_0000	UncertainLastUsableValue	Whatever was updating this value has stopped doing so.
4091_0000	UncertainSubstituteValue	The value is an operational value that was manually overwritten.
4092_0000	UncertainInitialValue	The value is an initial value for a variable that normally receives its value from another variable.
4093_0000	UncertainSensorNotAccurate	The value is at one of the sensor limits.
4094_0000	UncertainEngineeringUnitsExceeded	The value is outside of the range of values defined for this parameter.
4095_0000	UncertainSubNormal	The value is derived from multiple sources and has less than the required number of Good sources.
40A4_0000	UncertainDataSubNormal	The value is derived from multiple values and has less than the required number of Good values.
40BC_0000	UncertainReferenceNotDeleted	The server was not able to delete all target references.
40C0_0000	UncertainNotAllNodesAvailable	The list of references may not be complete because the underlying system is not available.
40DE_0000	UncertainDominantValueChanged	The related EngineeringUnit has been changed but the Variable Value is still provided based on the previous unit.
40E2_0000	UncertainDependentValueChanged	A dependent value has been changed but the change has not been applied to the device. The quality of the dominant variable is uncertain.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
8001_0000	BadUnexpectedError	An unexpected error occurred
8002_0000	BadInternalError	An internal error occurred as a result of a programming or configuration error. **Mögliche Ursache**: Sie haben auf ein einzelnes Element einer Variable vom Datentyp DTL schreibend zugegriffen. Schreibzugriffe auf Variablen vom Datentyp DTL sind nur auf die komplette Struktur möglich.
8003_0000	BadOutOfMemory	Not enough memory to complete the operation. **Mögliche Ursache:** Überschreitung des maximalen Mengengerüstes des OPC UA Servers **Abhilfe:** Reduzieren Sie die Anzahl der über OPC UA angemeldeten Elemente. Stellen Sie mit einem OPC UA Server Diagnose Tool (wie z.B. UaExpert von Unified Automation) sicher, dass keine inaktiven Sessions und Subscriptions vorhanden sind. Als Gegenmaßnahme können Sie entsprechende Timeouts (Subscription Timeout <= Session Timeout) im Client wählen. Siehe auch Beschreibung zu Fehlermeldungen OPC_UA_Connect: Verbindung erzeugen (Seite 3799).
8004_0000	BadResourceUnavailable	An operating system resource is not available
8005_0000	BadCommunicationError	A low level communication error occurred. **Mögliche Ursache:** Die Server-Adresse (ServerEndpointURL) ist falsch oder unvollständig. Siehe auch Erläuterung zu diesem Fehlercode bei OPC_UA_Connect: Verbindung erzeugen (Seite 3799)
8006_0000	BadEncodingError	Encoding halted because of invalid data in the objects being serialized.
8007_0000	BadDecodingError	Decoding halted because of invalid data in the stream.
8008_0000	BadEncodingLimitsExceeded	The message encoding/decoding limits imposed by the stack have been exceeded.
8009_0000	BadUnknownResponse	An unrecognized response was received from the server.
80B8_0000	BadRequestTooLarge	The request message size exceeds limits set by the server.
80B9_0000	BadResponseTooLarge	The response message size exceeds limits set by the client.
800A_0000	BadTimeout	The operation timed out. Siehe auch Fehlermeldung bei OPC_UA_Connect: Verbindung erzeugen (Seite 3799).
800B_0000	BadServiceUnsupported	The server does not support the requested service.
800C_0000	BadShutdown	The operation was cancelled because the application is shutting down.
800D_0000	BadServerNotConnected	The operation could not complete because the client is not connected to the server.
800E_0000	BadServerHalted	The server has stopped and cannot process any requests.
800F_0000	BadNothingToDo	There was nothing to do because the client passed a list of operations with no elements.
8010_0000	BadTooManyOperations	The request could not be processed because it specified too many operations. Siehe auch Fehlermeldung bei OPC_UA_Connect: Verbindung erzeugen (Seite 3799).

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
8011_0000	BadDataTypeIdUnknown	The extension object cannot be (de)serialized because the data type id is not recognized.
8012_0000	BadCertificateInvalid	The certificate provided as a parameter is not valid.
8013_0000	BadSecurityChecksFailed	An error occurred verifying security. The certificate provided as a parameter is not valid.
8014_0000	BadCertificateTimeInvalid	The Certificate has expired or is not yet valid.
8015_0000	BadCertificateIssuer TimeInvalid	An Issuer Certificate has expired or is not yet valid.
8016_0000	BadCertificateHost NameInvalid	The HostName used to connect to a Server does not match a HostName in the Certificate.
8017_0000	BadCertificateUriInvalid	The URI specified in the Application Description does not match the URI in the Certificate.
8018_0000	BadCertificateUseNotAllowed	The Certificate may not be used for the requested operation.
8019_0000	BadCertificateIssuerUse NotAllowed	The Issuer Certificate may not be used for the requested operation.
801A_0000	BadCertificateUntrusted	The Certificate is not trusted.
801B_0000	BadCertificateRevocation Unknown	It was not possible to determine if the Certificate has been revoked.
801C_0000	BadCertificateIssuer RevocationUnknown	It was not possible to determine if the Issuer Certificate has been revoked.
801D_0000	BadCertificateRevoked	The Certificate has been revoked.
801E_0000	BadCertificateIssuerRevoked	The Issuer Certificate has been revoked.
801F_0000	BadUserAccessDenied	User does not have permission to perform the requested operation.
8020_0000	BadIdentityTokenInvalid	The user identity token is not valid.
8021_0000	BadIdentityTokenRejected	The user identity token is valid but the server has rejected it.
8022_0000	BadSecureChannelIdInvalid	The specified secure channel is no longer valid.
8023_0000	BadInvalidTimestamp	The timestamp is outside the range allowed by the server.
8024_0000	BadNonceInvalid	The nonce does appear to be not a random value or it is not the correct length.
8025_0000	BadSessionIdInvalid	The session id is not valid.
8026_0000	BadSessionClosed	The session was closed by the client.
8027_0000	BadSessionNotActivated	The session cannot be used because ActivateSession has not been called.
8028_0000	BadSubscriptionIdInvalid	The subscription id is not valid.
802A_0000	BadRequestHeaderInvalid	The header for the request is missing or invalid.
802B_0000	BadTimestampsTo ReturnInvalid	The timestamps to return parameter is invalid.
802C_0000	BadRequestCancelled ByClient	The request was cancelled by the client.
8031_0000	BadNoCommunication	Communication with the data source is defined, but not established, and there is no last known value available.
8032_0000	BadWaitingForInitialData	Waiting for the server to obtain values from the underlying data source.
8033_0000	BadNodeIdInvalid	The syntax of the node id is not valid.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
8034_0000	BadNodeIdUnknown	The node id refers to a node that does not exist in the server address space. Möglicherweise existiert der NamespaceIndex der gegebenen NodeId nicht. Falls der Fehlercode beim Aufruf der Anweisung OPC_UA_MethodCall angezeigt wird: Eine weitere mögliche Ursache ist die Verwendung ungültiger ObjectIds beim vorhergehenden Aufruf von OPC_UA_MethodGetHandleList. Die Anweisung OPC_UA_MethodGetHandleList wertet die ObjectIds nicht aus.
8035_0000	BadAttributeIdInvalid	The attribute is not supported for the specified Node.
8036_0000	BadIndexRangeInvalid	The syntax of the index range parameter is invalid.
8037_0000	BadIndexRangeNoData	No data exists within the range of indexes specified.
8038_0000	BadDataEncodingInvalid	The data encoding is invalid.
8039_0000	BadDataEncoding Unsupported	The server does not support the requested data encoding for the node.
803A_0000	BadNotReadable	The access level does not allow reading or subscribing to the Node.
803B_0000	BadNotWritable	The access level does not allow writing to the Node.
803C_0000	BadOutOfRange	The value was out of range. Mögliche Abhilfe bei Arrays: Kontrollieren Sie, ob Sie auf Elemente eines Arrays zugreifen, die nicht definiert sind.
803D_0000	BadNotSupported	The requested operation is not supported. **Mögliche Ursache:** Die Server-Adresse (ServerEndpointURL) ist falsch oder unvollständig.
803E_0000	BadNotFound	A requested item was not found or a search operation ended without success.
803F_0000	BadObjectDeleted	The object cannot be used because it has been deleted.
8040_0000	BadNotImplemented	Requested operation is not implemented.
8041_0000	BadMonitoringModeInvalid	The monitoring mode is invalid.
8042_0000	BadMonitoredItemIdInvalid	The monitoring item id does not refer to a valid monitored item.
8043_0000	BadMonitoredItem FilterInvalid	The monitored item filter parameter is not valid.
8044_0000	BadMonitoredItem FilterUnsupported	The server does not support the requested monitored item filter.
8045_0000	BadFilterNotAllowed	A monitoring filter cannot be used in combination with the attribute specified.
8046_0000	BadStructureMissing	A mandatory structured parameter was missing or null.
8047_0000	BadEventFilterInvalid	The event filter is not valid.
8048_0000	BadContentFilterInvalid	The content filter is not valid.
8049_0000	BadFilterOperandInvalid	The operand used in a content filter is not valid.
804A_0000	BadContinuation PointInvalid	The continuation point provide is longer valid.
804B_0000	BadNoContinuationPoints	The operation could not be processed because all continuation points have been allocated.
804C_0000	BadReferenceTypeIdInvalid	The operation could not be processed because all continuation points have been allocated.
804D_0000	BadBrowseDirectionInvalid	The browse direction is not valid.
804E_0000	BadNodeNotInView	The node is not part of the view.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
804F_0000	BadServerUriInvalid	The ServerUri is not a valid URI.
8050_0000	BadServerNameMissing	No ServerName was specified
8051_0000	BadDiscoveryUrlMissing	No DiscoveryUrl was specified.
8052_0000	BadSempahoreFileMissing	The semaphore file specified by the client is not valid.
8053_0000	BadRequestTypeInvalid	The security token request type is not valid.
8054_0000	BadSecurityModeRejected	The security mode does not meet the requirements set by the Server. Der Server stellt höhere Sicherheitsanforderungen. Mögliche Abhilfe: Verwenden Sie eine höhere Sicherheitseinstellung für die Verbindung zum Server.
8055_0000	BadSecurityPolicyRejected	The security policy does not meet the requirements set by the Server.
8056_0000	BadTooManySessions	The server has reached its maximum number of sessions.
8057_0000	BadUserSignatureInvalid	The user token signature is missing or invalid.
8058_0000	BadApplicationSignature Invalid	The signature generated with the client certificate is missing or invalid.
8059_0000	BadNoValidCertificates	The client did not provide at least one software certificate that is valid and meets the profile requirements for the server.
805A_0000	BadRequestCancelled ByRequest	The request was cancelled by the client with the Cancel service.
805B_0000	BadParentNodeIdInvalid	The parent node id does not to refer to a valid node.
805C_0000	BadReferenceNotAllowed	The reference could not be created because it violates constraints imposed by the data model.
805D_0000	BadNodeIdRejected	The requested node id was reject because it was either invalid or server does not allow node ids to be specified by the client.
805E_0000	BadNodeIdExists	The requested node id is already used by another node.
805F_0000	BadNodeClassInvalid	The node class is not valid.
8060_0000	BadBrowseNameInvalid	The browse name is invalid.
8061_0000	BadBrowseNameDuplicated	The browse name is not unique among nodes that share the same relationship with the parent.
8062_0000	BadNodeAttributesInvalid	The node attributes are not valid for the node class.
8063_0000	BadTypeDefinitionInvalid	The type definition node id does not reference an appropriate type node.
8064_0000	BadSourceNodeIdInvalid	The source node id does not reference a valid node.
8065_0000	BadTargetNodeIdInvalid	The target node id does not reference a valid node.
8066_0000	BadDuplicateReference NotAllowed	The reference type between the nodes is already defined.
8067_0000	BadInvalidSelfReference	The server does not allow this type of selfreference on this node.
8068_0000	BadReferenceLocalOnly	The reference type is not valid for a reference to a remote server.
8069_0000	BadNoDeleteRights	The server will not allow the node to be deleted.
806A_0000	BadServerIndexInvalid	The server index is not valid.
806B_0000	BadViewIdUnknown	The view id does not refer to a valid view node.
806D_0000	BadTooManyMatches	The requested operation has too many matches to return.
806E_0000	BadQueryTooComplex	The requested operation requires too many resources in the server.
806F_0000	BadNoMatch	The requested operation has no match to return.
8070_0000	BadMaxAgeInvalid	The max age parameter is invalid.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
8071_0000	BadHistoryOperationInvalid	The history details parameter is not valid.
8072_0000	BadHistoryOperation Unsupported	The server does not support the requested operation.
8073_0000	BadWriteNotSupported	The server not does support writing the combination of value, status and timestamps provided.
8074_0000	BadTypeMismatch	The value supplied for the attribute is not of the same type as the attribute's value.
8075_0000	BadMethodInvalid	The method id does not refer to a method for the specified object.
8076_0000	BadArgumentsMissing	The client did not specify all of the input arguments for the method.
8077_0000	BadTooManySubscriptions	The server has reached its maximum number of subscriptions. **Mögliche Ursache:** Überschreitung der maximalen Anzahl von Subscriptions **Abhilfe:** Legen Sie eine Subscription in einer neuen Session an.
8078_0000	BadTooManyPublish Requests	The server has reached the maximum number of queued publish requests.
8079_0000	BadNoSubscription	There is no subscription available for this session.
807A_0000	BadSequenceNumber Unknown	The sequence number is unknown to the server.
807B_0000	BadMessageNotAvailable	The requested notification message is no longer available.
807C_0000	BadInsufficientClientProfile	The Client of the current Session does not support one or more Profiles that are necessary for the Subscription.
80BF_0000	BadStateNotActive	The sub-state machine is not currently active.
807D_0000	BadTcpServerTooBusy	The server cannot process the request because it is too busy.
807E_0000	BadTcpMessageTypeInvalid	The type of the message specified in the header invalid.
807F_0000	BadTcpSecureChannel Unknown	The SecureChannelId and/or TokenId are not currently in use.
8080_0000	BadTcpMessageTooLarge	The size of the message specified in the header is too large.
8081_0000	BadTcpNotEnough Resources	There are not enough resources to process the request.
8082_0000	BadTcpInternalError	An internal error occurred.
8083_0000	BadTcpEndpointUrlInvalid	The Server does not recognize the QueryString specified.
8084_0000	BadRequestInterrupted	The request could not be sent because of a network interruption.
8085_0000	BadRequestTimeout	Timeout occurred while processing the request.
8086_0000	BadSecureChannelClosed	The secure channel has been closed.
8087_0000	BadSecureChannelToken Unknown	The token has expired or is not recognized.
8088_0000	BadSequenceNumberInvalid	The sequence number is not valid.
8089_0000	BadConfigurationError	There is a problem with the configuration that affects the usefulness of the value.
808A_0000	BadNotConnected	The variable should receive its value from another variable, but has never been configured to do so.
808B_0000	BadDeviceFailure	There has been a failure in the device/data source that generates the value that has affected the value.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
808C_0000	BadSensorFailure	There has been a failure in the sensor from which the value is derived by the device/data source.
808D_0000	BadOutOfService	The source of the data is not operational.
808E_0000	BadDeadbandFilterInvalid	The dead band filter is not valid.
8097_0000	BadRefreshInProgress	This Condition refresh failed, a Condition refresh operation is already in progress.
8098_0000	BadConditionAlreadyDisabled	This condition has already been disabled.
8099_0000	BadConditionDisabled	Property not available, this condition is disabled.
809A_0000	BadEventIdUnknown	The specified event id is not recognized.
809B_0000	BadNoData	No data exists for the requested time range or event filter.
809D_0000	BadDataLost	Data is missing due to collection started/stopped/lost.
809E_0000	BadDataUnavailable	Expected data is unavailable for the requested time range due to an unmounted volume an off-line archive or tape or similar reason for temporary unavailability.
809F_0000	BadEntryExists	The data or event was not successfully inserted because a matching entry exists.
80A0_0000	BadNoEntryExists	The data or event was not successfully updated because no matching entry exists.
80A1_0000	BadTimestampNotSupported	The client requested history using a timestamp format the server does not support (i. e. requested ServerTimestamp when server only supports SourceTimestamp).
80AB_0000	BadInvalidArgument	One or more arguments are invalid. **Mögliche Ursache:** Die Server-Adresse (ServerEndpointURL) ist falsch oder unvollständig.
80AC_0000	BadConnectionRejected	Could not establish a network connection to remote server.
80AD_0000	BadDisconnect	The server has disconnected from the client.
80AE_0000	BadConnectionClosed	The network connection has been closed.
80AF_0000	BadInvalidState	The operation cannot be completed because the object is closed uninitialized or in some other invalid state.
80B0_0000	BadEndOfStream	Cannot move beyond end of the stream.
80B1_0000	BadNoDataAvailable	No data is currently available for reading from a non-blocking stream.
80B2_0000	BadWaitingForResponse	The asynchronous operation is waiting for a response.
80B3_0000	BadOperationAbandoned	The asynchronous operation was abandoned by the caller.
80B4_0000	BadExpectedStreamToBlock	The stream did not return all data requested (possibly because it is a non-blocking stream).
80B5_0000	BadWouldBlock	Non-blocking behavior is required and the operation would block.
80B6_0000	BadSyntaxError	A value had an invalid syntax.
80B7_0000	BadMaxConnections Reached	The operation could not be finished because all available connections are in use.
80BB_0000	BadEventNot Acknowledgeable	The event cannot be acknowledged.
80BD_0000	BadInvalidTimestamp Argument	The defined timestamp to return was invalid.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
80BE_0000	BadProtocolVersion Unsupported	The applications do not have compatible protocol versions.
80C1_0000	BadFilterOperatorInvalid	An unrecognized operator was provided in a filter.
80C2_0000	BadFilterOperator Unsupported	A valid operator was provided, but the server does not provide support for this filter operator.
80C3_0000	BadFilterOperandCount Mismatch	The number of operands provided for the filter operator was less than expected for the operand provided.
80C4_0000	BadFilterElementInvalid	The referenced element is not a valid element in the content filter.
80C5_0000	BadFilterLiteralInvalid	The referenced literal is not a valid value.
80C6_0000	BadIdentityChangeNotSupported	The Server does not support changing the user identity assigned to the session.
80C8_0000	BadNotTypeDefinition	The provided Nodeid was not a type definition nodeid.
80C9_0000	BadViewTimestampInvalid	The view timestamp is not available or not supported.
80CA_0000	BadViewParameterMismatch	The view parameters are not consistent with each other.
80CB_0000	BadViewVersionInvalid	The view version is not available or not supported.
80CC_0000	BadConditionAlready Enabled	This condition has already been enabled.
80CD_0000	BadDialogNotActive	The dialog condition is not active.
80CE_0000	BadDialogResponseInvalid	The response is not valid for the dialog.
80CF_0000	BadConditionBranch AlreadyAcked	The condition branch has already been acknowledged.
80D0_0000	BadConditionBranch AlreadyConfirmed	The condition branch has already been confirmed.
80D1_0000	BadConditionAlreadyShelved	The condition has already been shelved.
80D2_0000	BadConditionNotShelved	The condition is not currently shelved.
80D3_0000	BadShelvingTimeOutOfRange	The shelving time not within an acceptable range.
80D4_0000	BadAggregateListMismatch	The requested number of Aggregates does not match the requested number of NodeIds.
80D5_0000	BadAggregateNotSupported	The requested Aggregate is not support by the server.
80D6_0000	BadAggregateInvalidInputs	The aggregate value could not be derived due to invalid data inputs.

Status "Bad" (8000_0000 - FFFF_FFFF)		
Fehlernummer (hex)	**Name**	**Bedeutung**
80DB_0000	BadTooManyMonitoredItems	The request could not be processed because there are too many monitored items in the subscription. **Mögliche Ursache:** Es wurde die maximale Anzahl überwachter Elemente in einer Subscription erreicht. **Abhilfe:** Zur Anmeldung weiterer Elemente müssen Sie eine neue Subscription anlegen. Bei Projekten ab TIA Portal V15 können Sie zusätzlich die maximale Anzahl der überwachten Elemente über die CPU Konfiguration einstellen. **Weitere mögliche Ursache:** Request mit zu vielen Elementen **Abhilfe:** Beachten Sie auf dem Server im Objekt "ServerCapabilities" die angegebenen Maxima.
80D7_0000	BadBoundNotFound	No data found to provide upper or lower bound value.
80D8_0000	BadBoundNotSupported	The server cannot retrieve a bound for the variable.
80DA_0000	BadAggregateConfiguration Rejected	The aggregate configuration is not valid for specified node.
80E1_0000	BadDominantValueChanged	The related EngineeringUnit has been changed but this change has not been applied to the device. The Variable Value is still dependent on the previous unit but its status is currently bad.
80E3_0000	BadDependentValueChanged	A dependent value has been changed but the change has not been applied to the device. The quality of the dominant variable is bad.
80E4_0000	BadRequestNotAllowed	The request was rejected by the server because it did not meet the criteria set by the server.
80E5_0000	BadTooManyArguments	Too many arguments were provided.
80E6_0000	BadSecurityModeInsufficient	The operation is not permitted over the current secure channel.
810D_0000	BadCertificateChainIncomplete	The certificate chain is incomplete.
810E_0000	BadLicenseExpired	The server requires a license to operate in general or to perform a service or operation, but existing license is expired.
810F_0000	BadLicenseLimitsExceeded	The server has limits on number of allowed operations / objects, based on installed licenses, and these limits where exceeded.
8110_0000	BadLicenseNotAvailable	The server does not have a license which is required to operate in general or to perform a service or operation.
8111_0000	BadNotExecutable	The executable attribute does not allow the execution of the method.
8112_0000	BadNumericOverflow	The number was not accepted because of a numeric overflow.
8113_0000	BadRequestNotComplete	The request has not been processed by the server yet.

Fehlercodes von PLCopen

Fehlercodes

Die folgende Tabelle enthält Fehlernummern von PLCopen.

Die Namen und Erläuterungen der Fehler sind im Original aufgeführt. Sie sind nur in Englisch verfügbar.

Fehlernummern von PLCopen

ErrorID (hex)	Name	Bedeutung
A000_0001	PLCopenUA_Bad_FW_ PermanentError	Internal, permanent error.
A000_0002	PLCopenUA_Bad_FW_ TempError	Temp. Error; FB could retry to reach FW.
A000_0003	PLCopenUA_Bad_ ConnectionError	Connection could not be established.
A000_0004	PLCopenUA_Bad_ HostNotFound	The requested hostname could not be found.
A000_0005	PLCopenUA_Bad_ AlreadyConnected	Connection was already established.
A000_0006	PLCopenUA_Bad_ SecurityFailed	Connection failed due to security setup.
A000_0007	PLCopenUA_Bad_ Suspended	Connection is suspended.
A000_0008	PLCopenUA_Bad_ ConnectionInvalidHdl	Provided ConnectionHdl is not known.
A000_0009	PLCopenUA_Bad_ NSNotFound	A namespace with the requested name cannot be found on server.
A000_000A	PLCopenUA_Bad_ ResultTooLong	Target PLC variable is too short for retrieved data.
A000_000B	PLCopenUA_Bad_ InvalidType	Invalid or unsupported Type.
A000_000C	PLCopenUA_Bad_ NodeInvalidHdl	Provided NodeHdl is not known.
A000_000D	PLCopenUA_Bad_ MethodInvalidHdl	Provided MethodHdl is not known.
A000_000E	PLCopenUA_Bad_ ReadFailed	Read failed for unknown reason.
A000_000F	PLCopenUA_Bad_ WriteFailed	Write failed for unknown reason.
A000_0010	PLCopenUA_Bad_ CallFailed	Method Call failed for unknown reason.
A000_0011	PLCopenUA_Bad_ InParamFailed	Method Call Input parameter conversion failed.
A000_0012	PLCopenUA_Bad_ OutParamFailed	Method Call Output parameter conversion failed. ATTENTION: this means the MethodCall was executed successfully but the returned values could not be converted.

ErrorID (hex)	Name	Bedeutung
A000_0013	PLCopenUA_Bad_ SubscriptionInvalidHdl	Provided SubscriptionHdl is not known.
A000_0014	PLCopenUA_Bad_ MonitoredItemInvalidHdl	Provided MonitoredItemHdl is not known.

Beispielprogramme für OPC UA-Clients

Vorbemerkungen

Die Beispiel-Funktionsbausteine "Call_OpenDoor_On_Productionline", "ReadFromProductionline" und "WriteToProductionline" veranschaulichen jeweils nur die einzelne Funktion und sind nicht für eine gemeinsame Nutzung ausgelegt. Sie können daher nur einzeln ablaufen.

Die FBs bauen jeweils eine eigene Verbindung mit der Anweisung OPC_UA_Connect auf und greifen dabei auf die gleiche Client-Schnittstelle zu und damit auf den DB "Productionline_Configuration".

Die Verbindungs-Handles der drei Verbindungen werden bei gemeinsamer Nutzung in der derselben Variable abgelegt und überschreiben sich gegenseitig.

Berücksichtigen Sie diesen Sachverhalt, wenn Sie alle beschriebenen Funktionen in Ihrem Projekt nutzen.

Beispielprogramm für OPC_UA_ConnectionGetStatus

Beispielprogramm zum Analysieren von Verbindungen

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "Analyze_Connection".

Das Beispiel zeigt, wie ein Anwenderprogramm die Anweisung "OPC_UA_ConnectionGetStatus" verwendet, um Informationen über eine Verbindung zu einem OPC UA-Server einzuholen.

Die Beschreibung der Anweisung "OPC_UA_ConnectionGetStatus" finden Sie hier (Seite 3889).

Programmstruktur

Das Programm arbeitet als OPC UA-Client und gliedert sich in die folgenden Abschritte:

1. Aufbau einer Verbindung zu einem OPC UA-Server.
2. Analysieren der Verbindung
3. Abbau der Verbindung zum OPC UA-Server.

Sie starten das Programm mit einer positiven Flanke an "Input_REQ".

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen eines Funktionsbausteins, der die Anweisung "OPC_UA_ConnectionGetStatus" nutzt:

Analyze_Connection

Name	Data type	Default value
Input		
Input_REQ	Bool	false
Output		
Output_Busy	Bool	false
Output_Error	Bool	false
Output_Done	Bool	false
Output_Status	DWord	16#0
Output_Error_Message	WString	WSTRING#''
ConnectionStatus	UDInt	0
ServerState	UDInt	0
ServiceLevel	Byte	16#0
InOut		
<Hinzufügen>		
Static		
Mem_Input_REQ	Bool	false
Run	Bool	false
Started	Bool	false
State	Int	0
Mem_Status	DWord	16#0
Req	Bool	false
Busy	Bool	false
Error	Bool	false
Done	Bool	false
Status	DWord	16#0
SetServerEndpoint	Bool	TRUE
ServerEndpointUrl	WString	WSTRING#''
OPC_UA_SessionConnectInfo_Instance	OPC_UA_SessionConnectInfo	
ConnectionHdl	DWord	16#0
OPC_UA_ConnectionGetStatus_Instance	OPC_UA_ConnectionGetStatus	
OPC_UA_Disconnect_Instance	OPC_UA_Disconnect	
ResetREQ_For_OPC_UA_Disconnect	Bool	false
OPC_UA_Connect_Instance	OPC_UA_Connect	

Programm

SCL

```
IF #Run = FALSE THEN
    #Started := #Input_REQ AND NOT #Mem_Input_REQ;
    #Mem_Input_REQ := #Input_REQ;
    IF #Started THEN
        #Output_Busy := TRUE;
        #Output_Done := FALSE;
        #Output_Error := FALSE;
        #State := 1;
        #Run := TRUE;
        #ResetREQ_For_OPC_UA_Disconnect := TRUE;
        #SetServerEndpoint := TRUE;
    END_IF;
END_IF;

IF #Run = TRUE THEN


     CASE #State OF

        1:  // case 1, connect to server

             IF #SetServerEndpoint = TRUE THEN
                #SetServerEndpoint := FALSE;

                //set ServerSendpoint
                #ServerEndpointUrl := WSTRING#'opc.tcp://192.168.1.1:4840';

                //setting for SecurityMsgMode
                //1 = None
                //2 = Sign
                //3 = Sign & Encrypt
                #OPC_UA_SessionConnectInfo_Instance.SecurityMsgMode := 3;

                //setting for SecurityPolicy
                //1 = None
                //2 = Basic128Rsa15
                //3 = Basic256
                //4 = Basic256Sha256
                #OPC_UA_SessionConnectInfo_Instance.SecurityPolicy := 4;

                //set TransportProfile to 1
                #OPC_UA_SessionConnectInfo_Instance.TransportProfile := 1;

                //we are using our Client-Certificate Nr. 10
                //because Nr. 10 is the lastest Certificate we created for our client
application
                //please look into Certificate manager for the number of your Client
certificate
                //Certificate manager is located at
                // "Project tree > Security settings > Security features"
                #OPC_UA_SessionConnectInfo_Instance.CertificateID := 10;

            END_IF;
```

SCL

```
            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Connect_Instance(ServerEndpointUrl := #ServerEndpointUrl,
                                     SessionConnectInfo := #OPC_UA_SessionConnectInfo_Instance,
                                         REQ := #Req,
                                         Error => #Error,
                                         Done => #Done,
                                         Busy => #Busy,
                                         ConnectionHdl => #ConnectionHdl,
                                         Status => #Status);
            IF #Done = TRUE THEN
                #State := #State + 1;
                #ResetREQ_For_OPC_UA_Disconnect := TRUE;
            END_IF;

            IF #Error = TRUE THEN
                // Did we get a connection handle?
                IF #ConnectionHdl <> 0 THEN
                    // We have to release all resources in the server and disconnect
                    #State := 100;
                ELSE
                    #State := 99;
                END_IF;
                #Mem_Status := #Status;
                #OPC_UA_Connect_Instance(ServerEndpointUrl := #ServerEndpointUrl,
                                         SessionConnectInfo :=
#OPC_UA_SessionConnectInfo_Instance,
                                         REQ := FALSE);
                #ResetREQ_For_OPC_UA_Disconnect := TRUE;
            END_IF;


       2:  // case 2, analyse connection

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_ConnectionGetStatus_Instance(ConnectionHdl := #ConnectionHdl,
                                                 ConnectionStatus => #ConnectionStatus,
                                                 ServerState => #ServerState,
                                                 ServiceLevel => #ServiceLevel,
                                                 Timeout := T#6S,
                                                 Done => #Done,
                                                 Error => #Error,
                                                 Status => #Status,
                                                 Busy => #Busy,
                                                 REQ := #Req);
            IF #Done = TRUE THEN
                #State := #State + 1;
```

SCL

```
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                #Mem_Status := #Status;
                #OPC_UA_ConnectionGetStatus_Instance(REQ := FALSE,
                                                     ConnectionHdl := #ConnectionHdl);
            END_IF;


        3:  // case 3, disconnect from server
            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
                                        REQ := #Req,
                                        Timeout := T#6S,
                                        Done => #Done,
                                        Error => #Error,
                                        Busy => #Busy,
                                        Status => #Status);

            IF #Done = TRUE THEN
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                #Mem_Status := #Status;
                #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
                                            REQ := FALSE);
            END_IF;


         4: //case = 4, function block has run successfully

            #Output_Done := TRUE;
            #Output_Busy := FALSE;
            #State := 0;
            #Mem_Input_REQ := FALSE;
            #Run := FALSE;
            #Started := FALSE;


        99: // ERROR handling

            #Output_Error := TRUE;
            #Output_Status := #Mem_Status;
            #State := 0;
            #Output_Busy := FALSE;
            #Run := FALSE;
            #Mem_Input_REQ := FALSE;
            #Started := FALSE;
```

SCL

```
    100:    // ERROR handling, disconnect from server to release resources (handles)

            IF #ResetREQ_For_OPC_UA_Disconnect = TRUE THEN
                #ResetREQ_For_OPC_UA_Disconnect := FALSE;

            //set REQ to FALSE
                #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            REQ := FALSE);
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout := T#6S,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        Status => #Status);

            IF #Done = TRUE THEN
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

            IF #Error = TRUE THEN
                #Output_Error_Message := WString#'Error handling: Resources cannot be
released!';
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;


    END_CASE;

END_IF;
```

SCL

Beispielprogramm für das Lesen von PLC-Variablen

Beispielprogramm zum Lesen von PLC-Variablen

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "ReadFromProductionline".

Das Beispiel zeigt, wie ein Anwenderprogramm die Werte von PLC-Variablen liest.

Programmstruktur

Das Programm arbeitet als OPC UA-Client und gliedert sich in die folgenden Schritte:

1. Aufbau einer Verbindung zum OPC UA-Server der CPU, von der Werte gelesen werden sollen.
2. Lesen der Werte.
3. Abbau der Verbindung zum OPC UA-Server.

Sie starten das Programm mit einer positiven Flanke an "Input_REQ".

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen für den Funktionsbausteins "ReadFromProductionline".

Input	
Input_REQ	Bool
Output	
Output_Busy	Bool
Output_Error	Bool
Output_Done	Bool
Output_Status	DWord
Output_Error_Message	WString
InOut	
<Hinzufügen>	
Static	
Mem_Input_REQ	Bool
Run	Bool
Started	Bool
State	Int
Mem_Status	DWord
Busy	Bool
Error	Bool
Done	Bool
Status	DWord
Req	Bool
Set_REQ_To_FALSE	Bool
i	Int
▶ OPC_UA_Connect_Instance	OPC_UA_Connect
▶ OPC_UA_NamespaceGetIndexList_Instance	OPC_UA_NamespaceGetIndexList
▶ OPC_UA_NodeGetHandleList_Instance	OPC_UA_NodeGetHandleList
▶ OPC_UA_ReadList_Instance	OPC_UA_ReadList
▶ OPC_UA_NodeReleaseHandleList_Instance	OPC_UA_NodeReleaseHandleList
▶ OPC_UA_Disconnect_Instance	OPC_UA_Disconnect

Zudem verwendet das Beispielprogramm die folgenden Datenbausteine:

- Productionline_Configuration
- Productionline_Data

Diese Datenbausteine legt STEP 7 (TIA Portal) automatisch an, da das Beispielprogramm die Parametrierung für OPC UA verwendet.

Programm

SCL

```
IF #Run = FALSE THEN
    #Started := #Input_REQ AND NOT #Mem_Input_REQ;
    #Mem_Input_REQ := #Input_REQ;
    IF #Started THEN
        #Output_Busy := TRUE;
        #Output_Done := FALSE;
        #Output_Error := FALSE;
        #Output_Status := 0;
        #State := 1;
        #Run := TRUE;
        #Set_REQ_To_FALSE := TRUE;      //In case 100, to set REQ of instruction
"OPC_UA_Disconnect" to FALSE
    END_IF;
END_IF;

IF #Run = TRUE THEN


    CASE #State OF
        1:  // case 1, connect to server
            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                     SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                     ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                     Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                     REQ := #Req,
                                     Done => #Done,
                                     Busy => #Busy,
                                     Error => #Error,
                                     Status => #Status);


            IF #Done = TRUE THEN

                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                // Did we get a connection handle?
                IF "Productionline_Configuration".Connection.ConnectionHdl <> 0 THEN
                    // We have to release all resources in the server and disconnect
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                ELSE
                    #State := 99;
```

SCL

```
                END_IF;

                #Mem_Status := #Status;
                // set parameter REQ of OPC_UA_Connect to FALSE
                #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                         SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                         REQ := FALSE,
                                         ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                         Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

        2:  // case 2, get index for namespace

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                   StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                   NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   REQ := #Req,
                                                   Done => #Done,
                                                   Busy => #Busy,
                                                   Error => #Error,
                                                   Status => #Status);


            IF #Done = TRUE THEN
               IF "Productionline_Configuration".Namespaces.NamespaceStatusList[0] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NamespaceGetIndexList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
```

SCL

```
                #Mem_Status := #Status;
                #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                  StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                  NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                  EQ := FALSE);
            END_IF;

        3:  // case 3, get handles for nodes

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".ReadLists."ReadListProduct".Nodes,
                                               NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                               NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                               REQ := #Req,
                                               ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                               Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                               NodeIDCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                               NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                               NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                               Done => #Done,
                                               Error => #Error,
                                               Busy => #Busy,
                                               Status => #Status);

            IF #Done = TRUE THEN
                IF "Productionline_Data"."ReadListProduct".NodeStatusList[0] = 0 AND
"Productionline_Data"."ReadListProduct".NodeStatusList[1] = 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NodeGetHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
```

SCL

```
                #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".ReadLists."ReadListProduct".Nodes,
                                                   NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                   NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                   REQ := FALSE,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   NodeIDCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                                   NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
            END_IF;


        4:  // case 4, read from nodes

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_ReadList_Instance(
                                        NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                        NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                        Variable :=
"Productionline_Data"."ReadListProduct".Variable,
                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                        TimeStamps :=
"Productionline_Data"."ReadListProduct".TimeStamps,
                                        REQ := #Req,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        Status => #Status);


            IF #Done = TRUE THEN
                FOR #i := 0 TO
UINT_TO_INT("Productionline_Configuration".ReadLists."ReadListProduct".NodeCount) - 1 DO
                  IF NOT ("Productionline_Data"."ReadListProduct".NodeStatusList[#i] = 0)
THEN
                       #Output_Error_Message := CONCAT_WSTRING(IN1 := WSTRING#'Error at
Readlist "ReadListProduct", Index: ',
```

SCL

```
                                                                    IN2 := INT_TO_WSTRING(#i));
                    END_IF;
                END_FOR;
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_ReadList_Instance(
                                          NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                          NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                          Variable :=
"Productionline_Data"."ReadListProduct".Variable,
                                          REQ := FALSE,
                                          ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                          Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                          NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                          TimeStamps :=
"Productionline_Data"."ReadListProduct".TimeStamps);
            END_IF;


        5:  // case 5, release node handles

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NodeReleaseHandleList_Instance(
                                                   NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                   NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                   REQ := #Req,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount,
                                                   Done => #Done,
                                                   Error => #Error,
                                                   Busy => #Busy,
                                                   Status => #Status);

            IF #Done = TRUE THEN
```

SCL

```
                IF "Productionline_Data"."ReadListProduct".NodeStatusList[0] = 0 AND
                    "Productionline_Data"."ReadListProduct".NodeStatusList[1] = 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NodeReleaseHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_NodeReleaseHandleList_Instance(
                                                        NodeHdls :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeHdls,
                                                        NodeStatusList :=
"Productionline_Data"."ReadListProduct".NodeStatusList,
                                                        REQ := FALSE,
                                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                        NodeHdlCount :=
"Productionline_Configuration".ReadLists."ReadListProduct".NodeCount);
            END_IF;


        6:  // case 6, disconnect from server

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Done => #Done,
                                        Error => #Error,
                                        Busy => #Busy,
                                        Status => #Status);

            IF #Done = TRUE THEN
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
```

SCL

```
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
            END_IF;

        7: //case = 7, function block has run successfully

            #Output_Done := TRUE;
            #Output_Busy := FALSE;
            #State := 0;
            #Mem_Input_REQ := FALSE;
            #Run := FALSE;
            #Started := FALSE;


        99: // ERROR handling
            #Output_Error := TRUE;
            #Output_Status := #Mem_Status;
            #State := 0;
            #Output_Busy := FALSE;
            #Run := FALSE;
            #Mem_Input_REQ := FALSE;
            #Started := FALSE;


        100:    // ERROR handling, disconnect from server to release resources (handles)

            IF #Set_REQ_To_FALSE = TRUE THEN
                #Set_REQ_To_FALSE := FALSE;

                //set REQ to FALSE
                #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Done => #Done,
                                        Error => #Error,
                                        Busy => #Busy,
                                        Status => #Status);
```

SCL

```
            IF #Done = TRUE THEN
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

            IF #Error = TRUE THEN
                #Output_Error_Message := WString#'Error handling: Resources cannot be
released!';
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;
    END_CASE;
END_IF;
```

Beispielprogramm zum Schreiben von PLC-Variablen

Beispielprogramm zum Schreiben von PLC-Variablen

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "WriteToProductionline".

Das Beispiel zeigt, wie ein Anwenderprogramm die Werte von PLC-Variablen schreibt.

Programmstruktur

Das Programm arbeitet als OPC UA-Client und gliedert sich in die folgenden Schritte:

1. Aufbau einer Verbindung zum OPC UA-Server der CPU, in der Werte geschrieben werden sollen.
2. Schreiben der Werte
3. Abbau der Verbindung zum OPC UA-Server

Sie starten das Programm mit einer positiven Flanke an "Input_REQ".

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen für den Funktionsbaustein "WriteToProductionline":

Input	
Input_REQ	Bool
Output	
Output_Busy	Bool
Output_Error	Bool
Output_Done	Bool
Output_Status	DWord
Output_Error_Message	WString
InOut	
<Hinzufügen>	
Static	
Mem_Input_REQ	Bool
Run	Bool
Started	Bool
State	Int
Mem_Status	DWord
Req	Bool
Busy	Bool
Error	Bool
Done	Bool
Status	DWord
Set_REQ_To_FALSE	Bool
i	Int
SetProductionEnabled	Bool
▸ OPC_UA_Connect_Instance	OPC_UA_Connect
▸ OPC_UA_NamespaceGetIndexList_Instance	OPC_UA_NamespaceGetIndexList
▸ OPC_UA_NodeGetHandleList_Instance	OPC_UA_NodeGetHandleList
▸ OPC_UA_WriteList_Instance	OPC_UA_WriteList
▸ OPC_UA_NodeReleaseHandleList_Instance	OPC_UA_NodeReleaseHandleList
▸ OPC_UA_Disconnect_Instance	OPC_UA_Disconnect

Zudem verwendet das Beispielprogramm die folgenden Datenbausteine:

- Productionline_Configuration
- Productionline_Data

Diese Datenbausteine legt STEP 7 (TIA Portal) automatisch an, da das Beispielprogramm die Parametrierung für OPC UA verwendet.

Programm

SCL

```
IF #Run = FALSE THEN
    #Started := #Input_REQ AND NOT #Mem_Input_REQ;
    #Mem_Input_REQ := #Input_REQ;
    IF #Started THEN
        #Output_Busy := TRUE;
        #Output_Done := FALSE;
        #Output_Error := FALSE;
        #Output_Status := 0;
        #State := 1;
        #Run := TRUE;
        #SetProductionEnabled := TRUE;
        #Set_REQ_To_FALSE := TRUE;      //In case 100, to set REQ of instruction
"OPC_UA_Disvonnect" to FALSE
    END_IF;
END_IF;

IF #Run = TRUE THEN


    CASE #State OF
        1:  // case 1, connect to server

            IF #Busy = FALSE THEN

                #Req := TRUE;
            ELSE
                #Req := FALSE;

            END_IF;
            #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                     SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                     ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                     Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                     REQ := #Req,
                                     Done => #Done,
                                     Busy => #Busy,
                                     Error => #Error,
                                     Status => #Status);


            IF #Done = TRUE THEN
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN

                // Did we get a connection handle?
                IF "Productionline_Configuration".Connection.ConnectionHdl <> 0 THEN
                    // We have to release all resources in the server and disconnect
                    #State := 100;
```

SCL

```
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                ELSE
                    #State := 99;
                END_IF;

                #Mem_Status := #Status;
                #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                         SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                         REQ := FALSE,
                                         ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                         Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

        2:  // case 2, get index for namespace

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                   StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                   REQ := #Req,
                                                   NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   Busy => #Busy,
                                                   Done => #Done,
                                                   Error => #Error);

            IF #Done = TRUE THEN
               IF "Productionline_Configuration".Namespaces.NamespaceStatusList[0] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NamespaceGetIndexList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
```

SCL

```
                #Mem_Status := #Status;
                #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                       REQ := FALSE,
                                                       ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                       NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                       StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                       NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                       Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

        3:  // case 3, get handles for nodes

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".WriteLists."WriteListStatus".Nodes,
                                               NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                               NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                               REQ := #Req,
                                               ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                               Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                               NodeIDCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount,
                                               NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                               NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                               Busy => #Busy,
                                               Done => #Done,
                                               Error => #Error,
                                               Status =>#Status);

            IF #Done = TRUE THEN
                IF "Productionline_Data"."WriteListStatus".NodeStatusList[0] = 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NodeGetHandleList';
                END_IF;
            END_IF;
```

SCL

```
            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_NodeGetHandleList_Instance(NodeIDs :=
"Productionline_Configuration".WriteLists."WriteListStatus".Nodes,
                                                   NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                                   NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                                   REQ := FALSE,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   NodeIDCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount,
                                                   NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
            END_IF;

        4:  // case 4;  write value to PLC variable "ProductionEnabled"

            IF #SetProductionEnabled = TRUE THEN
                #SetProductionEnabled := FALSE;
                //set new value to true
               "Productionline_Data"."WriteListStatus".Variable.ProductionEnabled := TRUE;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_WriteList_Instance(
                                       NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                       NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                       Variable :=
"Productionline_Data"."WriteListStatus".Variable,
                                       ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                       Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                       NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount,
                                       REQ := #Req,
                                       Busy => #Busy,
                                       Error => #Error,
                                       Status => #Status,
```

SCL

```
                                        Done => #Done);

            IF #Done = TRUE THEN
                FOR #i := 0 TO
UINT_TO_INT("Productionline_Configuration".WriteLists.WriteListStatus.NodeCount) - 1 DO
                    IF NOT ("Productionline_Data"."WriteListStatus".NodeStatusList[#i] = 0)
THEN
                        #Output_Error_Message := CONCAT_WSTRING(IN1 := WSTRING#'Error at
Writelist "WriteListStauts", Index: ',
                                                                    IN2 := INT_TO_WSTRING(#i));
                    END_IF;
                END_FOR;
                #State := #State + 1;
            END_IF;
            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_WriteList_Instance(
                                            NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                            NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                            Variable :=
"Productionline_Data"."WriteListStatus".Variable,
                                            REQ := FALSE,
                                            ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                            NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount);
            END_IF;

        5:  // case 5, release node handles

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_NodeReleaseHandleList_Instance(NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                                   NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                                   REQ := #Req,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount,
                                                   Busy => #Busy,
                                                   Done => #Done,
                                                   Error => #Error,
```

SCL

```
                                                    Status => #Status);

            IF #Done = TRUE THEN
                IF "Productionline_Data"."WriteListStatus".NodeStatusList[0] = 0
                THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at NodeReleaseHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;

                #OPC_UA_NodeReleaseHandleList_Instance(NodeHdls :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeHdls,
                                                       NodeStatusList :=
"Productionline_Data"."WriteListStatus".NodeStatusList,
                                                       REQ := FALSE,
                                                       ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                       Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                       NodeHdlCount :=
"Productionline_Configuration".WriteLists."WriteListStatus".NodeCount);

                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
            END_IF;


        6:  // case 6, disconnect from server

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Busy => #Busy,
                                        Done => #Done,
                                        Error => #Error,
                                        Status => #Status);


            IF #Done = TRUE THEN
                #State := #State + 1;
```

SCL

```
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
            END_IF;

        7: //case = 7, function block has run successfully

            #Output_Done := TRUE;
            #Output_Busy := FALSE;
            #State := 0;
            #Mem_Input_REQ := FALSE;
            #Run := FALSE;
            #Started := FALSE;

        99: // ERROR handling
            #Output_Error := TRUE;
            #Output_Status := #Mem_Status;
            #State := 0;
            #Output_Busy := FALSE;
            #Run := FALSE;
            #Mem_Input_REQ := FALSE;
            #Started := FALSE;

        100:    // ERROR handling, disconnect from server to release resources (handles)

            IF #Set_REQ_To_FALSE = TRUE THEN
                #Set_REQ_To_FALSE := FALSE;

                //set REQ to FALSE
                #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        Status => #Status);
```

SCL

```
            IF #Done = TRUE THEN
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

            IF #Error = TRUE THEN
                #Output_Error_Message := WString#'Error handling: Resources cannot be
released!';
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

    END_CASE;


END_IF;
```

Beispielprogramm für OPC_UA_TranslatePathList

Beispielprogramm zur Anweisung "OPC_UA_TranslatePathList"

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "Read_From_RfidReader_Door_1".

Das Beispiel zeigt, wie ein Anwenderprogramm die Anweisung "OPC_UA_TranslatePathList" verwendet, um die NodeIds von Knoten zu erhalten und anschließend die Werte dieser Knoten zu lesen.

Die Beschreibung der Anweisung "OPC_UA_TranslatePathList" finden Sie hier (Seite 3831).

Programmstruktur

Das Programm arbeitet als OPC UA-Client und gliedert sich in die folgenden Schritte:

1. Aufbau einer Verbindung zum OPC UA-Server der CPU, von der Werte gelesen werden sollen.
2. Lesen der Werte.
3. Abbau der Verbindung zum OPC UA-Server.

Sie starten das Programm mit einer positiven Flanke an "Input_REQ".

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen des Funktionsbausteins "Read_From_RfidReader_Door_1", der die Anweisung "OPC_UA_TanslatePathList" nutzt:

Read_From_RfidReader_Door_1		
	Name	Data type
1	Input	
2	Input_REQ	Bool
3	Output	
4	Output_Busy	Bool
5	Output_Error	Bool
6	Output_Done	Bool
7	Output_Status	DWord
8	Output_Error_Message	WString
11	Static	
12	Mem_Input_REQ	Bool
13	Run	Bool
14	Started	Bool
15	State	Int
16	Mem_Status	DWord
17	Req	Bool
18	Busy	Bool
19	Error	Bool
20	Done	Bool
21	Status	DWord
22	i	Int
23	FirstCall	Bool
24	SeverEndpointUrl	WString
25	SessionConnectInfo	OPC_UA_SessionConnectInfo
26	OPC_UA_Connect_Instance	OPC_UA_Connect
27	NamespaceUris	Array[0..0] of WString
28	NamespaceIndexes	Array[0..0] of UInt
29	NamespaceStatusList	Array[0..0] of DWord
30	OPC_UA_NamespaceGetIndexList_Instance	OPC_UA_NamespaceGetIndexList
31	OPC_UA_NodeGetHandleList_Instance	OPC_UA_NodeGetHandleList
32	OPC_UA_ReadList_Instance	OPC_UA_ReadList
33	OPC_UA_NodeReleaseHandleList_Instance	OPC_UA_NodeReleaseHandleList
34	OPC_UA_Disconnect_Instance	OPC_UA_Disconnect
35	OPC_UA_TranslatePathList_Instance	OPC_UA_TranslatePathList
36	BrowsePaths	Array[0..1] of OPC_UA_BrowsePath
37	TagetNodeIds	Array[0..1] of OPC_UA_NodeId
38	TargetStatusList	Array[0..1] of DWord
39	NodeStatusList	Array[0..1] of DWord
40	NodeHdls	Array[0..1] of DWord
41	ConnectionHdl	DWord
42	Set_REQ_To_FALSE	Bool
43	Variable	"UDT_Variable"

Programm

Das folgende Beispiel zeigt, wie Sie die Anweisung "OPC UA-TranslatePathList" nutzen können:

SCL

```
IF #Run = FALSE THEN
    #Started := #Input_REQ AND NOT #Mem_Input_REQ;
    #Mem_Input_REQ := #Input_REQ;
    IF #Started THEN
        #Output_Busy := TRUE;
        #Output_Done := FALSE;
        #Output_Error := FALSE;
        #Output_Status := 0;
        #State := 1;
        #Run := TRUE;
        #FirstCall := TRUE;
        #Set_REQ_To_FALSE := TRUE;      //In case 100, to set REG of instruction
"OPC_UA_Disconnect" to FALSE
    END_IF;
END_IF;

IF #Run = TRUE THEN


    CASE #State OF
        1:  // case 1, connect to server


            IF #FirstCall = TRUE THEN
                #FirstCall := FALSE;

                //set ServerEndPointUrl of the server we want to connect to
                #SeverEndpointUrl := WString#'opc.tcp://192.168.1.1:4840';

                //set Securtiy Message Mode
                // 1 = None, 2 = Sign, 3 = Sign & Encrypt
                #SessionConnectInfo.SecurityMsgMode := 1;

                //set Security Policy
                // 1 = None, 2 = Basic128Rsa15,, 3 = Basic256, 4 = Basic256Sha256
                #SessionConnectInfo.SecurityPolicy := 1;

                //set the number of the client certificate you are using
                //in our case, this is 6
                //#SessionConnectInfo.CertificateID := 6;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Connect_Instance(REQ:=#Req,
                                     ServerEndpointUrl:=#SeverEndpointUrl,
                                     SessionConnectInfo:=#SessionConnectInfo,
                                     Timeout:=T#8S,
                                     Done=>#Done,
                                     Busy=>#Busy,
                                     Error=>#Error,
```

SCL

```
                                Status=>#Status,
                                ConnectionHdl=>#ConnectionHdl);

        IF #Done = TRUE THEN
            #State := #State + 1;
            #FirstCall := TRUE;
        END_IF;

        IF #Error = TRUE THEN
            // Did we get a connection handle?
            IF #ConnectionHdl <> 0 THEN
                // We got a connection handle
                // We have to release all resources in the server and disconnect
                #State := 100;
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
            ELSE
                // We did not get a connection handle
                #State := 99;
            END_IF;

            #Mem_Status := #Status;
            // set parameter REQ of OPC_UA_Connect to FALSE
            #OPC_UA_Connect_Instance(ServerEndpointUrl := #SeverEndpointUrl,
                                     SessionConnectInfo := #SessionConnectInfo,
                                     REQ := FALSE);
            #FirstCall := TRUE;
        END_IF;

    2:  // case 2, get index of Siemens namespace


        IF #FirstCall = TRUE THEN
            #FirstCall := FALSE;

            //set Simatic namespace
            #NamespaceUris[0] := WString#'http://www.siemens.com/simatic-s7-opcua';
        END_IF;

        IF #Busy = FALSE THEN
            #Req := TRUE;
        ELSE
            #Req := FALSE;
        END_IF;

        #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris := #NamespaceUris,
                                               ConnectionHdl := #ConnectionHdl,
                                               NamespaceIndexes := #NamespaceIndexes,
                                               StatusList := #NamespaceStatusList,
                                               REQ := #Req,
                                               NamespaceUrisCount := 1,
                                               Timeout := T#6S,
                                               Busy => #Busy,
                                               Done => #Done,
                                               Error => #Error,
                                               Status => #Status);
```

SCL

```
            IF #Done = TRUE THEN
                IF #NamespaceStatusList[0] = 0 THEN  //one namespace
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                  #Output_Error_Message := WString#' Error at NamespaceGetIndexList, Index
0';
                END_IF;
                #FirstCall := TRUE;
            END_IF;

            IF #Error = TRUE THEN
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #State := 100;
                #Mem_Status := #Status;
                #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris := #NamespaceUris,
                                                       REQ := FALSE,
                                                       StatusList := #NamespaceStatusList);
                #FirstCall := TRUE;
            END_IF;



        3:  // case 3, get NodeIds from server




            IF #FirstCall = TRUE THEN
                //set BrowsePath[0] for DeviceInfo
                //set BrowsePath.StartingNode
                #BrowsePaths[0].StartingNode.Identifier := WString#'85';
                #BrowsePaths[0].StartingNode.IdentifierType := 0;
                #BrowsePaths[0].StartingNode.NamespaceIndex := 0;

                //set BrowsePath.RelativePath
                #BrowsePaths[0].RelativePath.NoOfElements := 4;
                #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[0].RelativePath.Elements[1].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[0].RelativePath.Elements[1].TargetName.Name :=
WString#'productionline';
                #BrowsePaths[0].RelativePath.Elements[1].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.IdentifierType :=
0;
```

SCL

```
                #BrowsePaths[0].RelativePath.Elements[2].ReferenceTypeId.NamespaceIndex :=
0;
                 #BrowsePaths[0].RelativePath.Elements[2].TargetName.Name :=
WString#'DataBlocksGlobal';
                 #BrowsePaths[0].RelativePath.Elements[2].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                 #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[0].RelativePath.Elements[3].ReferenceTypeId.NamespaceIndex :=
0;
                 #BrowsePaths[0].RelativePath.Elements[3].TargetName.Name :=
WString#'RfidReader_Door_1';
                 #BrowsePaths[0].RelativePath.Elements[3].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                 #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[0].RelativePath.Elements[4].ReferenceTypeId.NamespaceIndex :=
0;
                 #BrowsePaths[0].RelativePath.Elements[4].TargetName.Name :=
WString#'DeviceInfo';
                 #BrowsePaths[0].RelativePath.Elements[4].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                 //set BrowsePath[1] for DeviceStatus
                 //set BrowsePath.StartingNode
                 #BrowsePaths[1].StartingNode.Identifier := WString#'85';
                 #BrowsePaths[1].StartingNode.IdentifierType := 0;
                 #BrowsePaths[1].StartingNode.NamespaceIndex := 0;

                 //set BrowsePath.RelativePath
                 #BrowsePaths[1].RelativePath.NoOfElements := 4;
                 #BrowsePaths[1].RelativePath.Elements[1].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[1].RelativePath.Elements[1].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[1].RelativePath.Elements[1].ReferenceTypeId.NamespaceIndex :=
0;
                 #BrowsePaths[1].RelativePath.Elements[1].TargetName.Name :=
WString#'productionline';
                 #BrowsePaths[1].RelativePath.Elements[1].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                 #BrowsePaths[1].RelativePath.Elements[2].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[1].RelativePath.Elements[2].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[1].RelativePath.Elements[2].ReferenceTypeId.NamespaceIndex :=
0;
                 #BrowsePaths[1].RelativePath.Elements[2].TargetName.Name :=
WString#'DataBlocksGlobal';
```

SCL

```
                #BrowsePaths[1].RelativePath.Elements[2].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[1].RelativePath.Elements[3].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[1].RelativePath.Elements[3].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[1].RelativePath.Elements[3].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[1].RelativePath.Elements[3].TargetName.Name :=
WString#'RfidReader_Door_1';
                #BrowsePaths[1].RelativePath.Elements[3].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

                #BrowsePaths[1].RelativePath.Elements[4].ReferenceTypeId.Identifier :=
WString#'33';
                #BrowsePaths[1].RelativePath.Elements[4].ReferenceTypeId.IdentifierType :=
0;
                #BrowsePaths[1].RelativePath.Elements[4].ReferenceTypeId.NamespaceIndex :=
0;
                #BrowsePaths[1].RelativePath.Elements[4].TargetName.Name :=
WString#'DeviceStatus';
                #BrowsePaths[1].RelativePath.Elements[4].TargetName.NamespaceIndex :=
#NamespaceIndexes[0];

            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_TranslatePathList_Instance(BrowsePaths := #BrowsePaths,
                                               TargetNodeIDs := #TagetNodeIds,
                                               TargetStatusList := #TargetStatusList,
                                               ConnectionHdl := #ConnectionHdl,
                                               NamespaceIndexCount := 0,
                                               BrowsePathsCount := 2,
                                               Timeout := T#6S,
                                               REQ := #Req,
                                               Busy => #Busy,
                                               Error => #Error,
                                               Done => #Done,
                                               Status => #Status);
            IF #Done = TRUE THEN
                IF #TargetStatusList[0] = 0 AND #TargetStatusList[1] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at TranslatePathList';
                END_IF;
                #FirstCall := TRUE;
            END_IF;
            IF #Error = TRUE THEN
```

SCL

```
            //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
            #Set_REQ_To_FALSE := TRUE;
            #State := 100;
            #Mem_Status := #Status;
            #OPC_UA_TranslatePathList_Instance(BrowsePaths := #BrowsePaths,
                                               TargetNodeIDs := #TagetNodeIds,
                                               REQ := FALSE,
                                               TargetStatusList := #TargetStatusList,
                                               ConnectionHdl := #ConnectionHdl);
            #FirstCall := TRUE;
        END_IF;


    4:  // case 4, get handles for the NodeIds we have got in case 3




        IF #Busy = FALSE THEN
            #Req := TRUE;
        ELSE
            #Req := FALSE;
        END_IF;

        #OPC_UA_NodeGetHandleList_Instance(NodeIDs := #TagetNodeIds,
                                           NodeStatusList := #NodeStatusList,
                                           NodeHdls := #NodeHdls,
                                           ConnectionHdl := #ConnectionHdl,
                                           Timeout := T#6S,
                                           NodeIDCount := 2,
                                           REQ := #Req,
                                           Busy => #Busy,
                                           Done => #Done,
                                           Error => #Error,
                                           Status => #Status);
        IF #Done = TRUE THEN
            IF #NodeStatusList[0] = 0 AND #NodeStatusList[1] = 0
            THEN
                #State := #State + 1;
            ELSE
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #State := 100;
                #Mem_Status := #Status;
                #Output_Error_Message := WString#'Error at NodeGetHandleList';
            END_IF;
        END_IF;

        IF #Error = TRUE THEN
            //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
            #Set_REQ_To_FALSE := TRUE;
            #State := 100;
            #Mem_Status := #Status;
            #OPC_UA_NodeGetHandleList_Instance(NodeIDs := #TagetNodeIds,
                                               NodeStatusList := #NodeStatusList,
                                               NodeHdls := #NodeHdls,
```

SCL

```
                                                REQ := FALSE);
            END_IF;

        5:  // case 5, read the values of the NodeIds



            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_ReadList_Instance(NodeHdls := #NodeHdls,
                                      NodeStatusList := #NodeStatusList,
                                      Variable := #Variable,
                                      REQ := #Req,
                                      ConnectionHdl := #ConnectionHdl,
                                      Timeout := T#6S,
                                      NodeHdlCount := 2,
                                      Busy => #Busy,
                                      Done => #Done,
                                      Error => #Error,
                                      Status => #Status);


            IF #Done = TRUE THEN
                FOR #i := 0 TO 1 DO
                    IF NOT (#NodeStatusList[#i] = 0) THEN
                        #Output_Error_Message := CONCAT_WSTRING(IN1 := WSTRING#'Error at
Readlist, Index: ',
                                                                IN2 := INT_TO_WSTRING(#i));
                    END_IF;
                END_FOR;
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #State := 100;
                #Mem_Status := #Status;
                #OPC_UA_ReadList_Instance(NodeHdls := #NodeHdls,
                                          NodeStatusList := #NodeStatusList,
                                          Variable := #Variable,
                                          REQ := FALSE);
            END_IF;


        6:  // case 6, release the node handles


            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
```

SCL

```
        END_IF;

        #OPC_UA_NodeReleaseHandleList_Instance(NodeHdls := #NodeHdls,
                                               NodeStatusList := #NodeStatusList,
                                               REQ := #Req,
                                               ConnectionHdl := #ConnectionHdl,
                                               Timeout := T#6S,
                                               NodeHdlCount := 1,
                                               Busy => #Busy,
                                               Done => #Done,
                                               Error => #Error,
                                               Status => #Status);

        IF #NodeStatusList[0] = 0 AND #NodeStatusList[1] = 0
        THEN
            #State := #State + 1;
        ELSE
            //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
            #Set_REQ_To_FALSE := TRUE;
            #State := 100;
            #Mem_Status := #Status;
            #Output_Error_Message := WString#'Error at NodeReleasetHandleList';
        END_IF;

        IF #Error = TRUE THEN
            //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
            #Set_REQ_To_FALSE := TRUE;
            #State := 100;
            #Mem_Status := #Status;
            #OPC_UA_NodeReleaseHandleList_Instance(NodeHdls := #NodeHdls,
                                                   NodeStatusList := #NodeStatusList,
                                                   REQ := FALSE);
        END_IF;


    7:  // case 7; disconnect form server


        IF #Busy = FALSE THEN
            #Req := TRUE;
        ELSE
            #Req := FALSE;
        END_IF;

        #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
                                    REQ := #Req,
                                    Timeout := T#6S,
                                    Busy => #Busy,
                                    Done => #Done,
                                    Error => #Error,
                                    Status => #Status);

        IF #Done = TRUE THEN
            #State := #State + 1;
        END_IF;
```

SCL

```
        IF #Error = TRUE THEN
            //In case 100, to set REG of instruction "OPC_UA_Disconnect" to FALSE
            #Set_REQ_To_FALSE := TRUE;
            #State := 100;
            #Mem_Status := #Status;
            #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
                                        REQ := FALSE);

        END_IF;


    8: //case = 8, function block has run successfully



        #Output_Done := TRUE;
        #Output_Busy := FALSE;
        #State := 0;
        #Mem_Input_REQ := FALSE;
        #Run := FALSE;
        #Started := FALSE;


    99: // ERROR handling



        #Output_Error := TRUE;
        #Output_Status := #Mem_Status;
        #State := 0;
        #Output_Busy := FALSE;
        #Run := FALSE;
        #Mem_Input_REQ := FALSE;
        #Started := FALSE;


    100:    // ERROR handling, disconnect from server to release resources (handles)



        IF #Set_REQ_To_FALSE = TRUE THEN
            #Set_REQ_To_FALSE := FALSE;

            //set REQ to FALSE
            #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
                                        REQ := FALSE);
        END_IF;

        IF #Busy = FALSE THEN
            #Req := TRUE;
        ELSE
            #Req := FALSE;
        END_IF;

        #OPC_UA_Disconnect_Instance(ConnectionHdl := #ConnectionHdl,
```

SCL

```
                                        REQ := #Req,
                                        Timeout := T#6S,
                                        Busy => #Busy,
                                        Done => #Done,
                                        Error => #Error,
                                        Status => #Status);


            IF #Done = TRUE THEN
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

            IF #Error = TRUE THEN
                #Output_Error_Message := WString#'Error handling: Resources cannot be
released!';
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

    END_CASE;

END_IF;
```

Beispielprogramm zum Aufruf einer Methode eines OPC UA-Servers

Beispielprogramm zum Aufruf einer Server-Methode

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "Call_OpenDoor_On_Productionline".

Das Beispiel zeigt, wie ein Anwenderprogramm die Anweisung "OPC_UA_MethodCall" verwendet, um eine Server-Methode aufzurufen.

Die Beschreibung der Anweisung "OPC_UA_MethodCall" finden Sie hier (Seite 3906).

Programmstruktur

Das Programm arbeitet als OPC UA-Client und gliedert sich in die folgenden Schritte:

1. Aufbau einer Verbindung zu einem OPC UA-Server.
2. Aufruf der Server-Methode
3. Abbau der Verbindung zum OPC UA-Server.

Sie starten das Programm mit einer positiven Flanke an "Input_REQ".

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen für den Funktionsbaustein "Call_OpenDoor_On_Productionline":

Input	
Input_REQ	Bool
Output	
Output_Busy	Bool
Output_Error	Bool
Output_Done	Bool
Output_Status	DWord
Output_Error_Message	WString
Output_MethodResult	DWord
InOut	
<Hinzufügen>	
Static	
Mem_Input_REQ	Bool
Run	Bool
Started	Bool
State	Int
Mem_Status	DWord
MethodResult	DWord
Req	Bool
Busy	Bool
Error	Bool
Done	Bool
Status	DWord
Init_Method_InputParameter	Bool
Set_REQ_To_FALSE	Bool
▸ OPC_UA_Connect_Instance	OPC_UA_Connect
▸ OPC_UA_NamespaceGetIndexList_Instance	OPC_UA_NamespaceGetIndexList
▸ OPC_UA_MethodGetHandleList_Instance	OPC_UA_MethodGetHandleList
▸ OPC_UA_MethodCall_Instance	OPC_UA_MethodCall
▸ OPC_UA_MethodReleaseHandleList_Instance	OPC_UA_MethodReleaseHandleList
▸ OPC_UA_Disconnect_Instance	OPC_UA_Disconnect

Programm

SCL

```
IF #Run = FALSE THEN
    #Started := #Input_REQ AND NOT #Mem_Input_REQ;
    #Mem_Input_REQ := #Input_REQ;
    IF #Started THEN
        #Output_Busy := TRUE;
        #Output_Done := FALSE;
        #Output_Error := FALSE;
        #Output_Status := 0;
        #Output_Error_Message := WSTRING#'';
        #State := 1;
        #Run := TRUE;
        #Init_Method_InputParameter := TRUE;
        #Set_REQ_To_FALSE := TRUE;      //In case 100, to set REQ of instruction
"OPC_UA_Disconnect" to FALSE
    END_IF;
END_IF;


IF #Run = TRUE THEN


    CASE #State OF
        1:  // case 1, connect to server

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_Connect_Instance(REQ := #Req,
                                     ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                     SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                     Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                     Done => #Done,
                                     Busy => #Busy,
                                     Error => #Error,
                                     Status => #Status,
                                     ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl);


            IF #Done = TRUE THEN
                #State := #State + 1;
            END_IF;

            IF #Error = TRUE THEN
                // Did we get a connection handle?
                IF "Productionline_Configuration".Connection.ConnectionHdl <> 0 THEN
                    // We have to release all resources in the server and disconnect
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
```

SCL

```
                    #Set_REQ_To_FALSE := TRUE;
                ELSE
                    #State := 99;
                END_IF;
                #Mem_Status := #Status;
                #OPC_UA_Connect_Instance(ServerEndpointUrl :=
"Productionline_Configuration".Connection.ServerEndpointUrl,
                                         SessionConnectInfo :=
"Productionline_Configuration".Connection.ConnectInfo,
                                         REQ := FALSE,
                                         ConnectionHdl =>
"Productionline_Configuration".Connection.ConnectionHdl,
                                         Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);

            END_IF;

        2:  // case 2, get index of Siemens namespace

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_NamespaceGetIndexList_Instance(REQ := #Req,
                                                   ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                   NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                   NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                   Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                   Done => #Done,
                                                   Busy => #Busy,
                                                   Error => #Error,
                                                   Status => #Status,
                                                   StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                   NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);


            IF #Done = TRUE THEN
                IF "Productionline_Configuration".Namespaces.NamespaceStatusList[0] = 0
THEN  //one namespace
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                  #Output_Error_Message := WString#' Error at NamespaceGetIndexList, Index
0';
                END_IF;
```

SCL

```
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_NamespaceGetIndexList_Instance(NamespaceUris :=
"Productionline_Configuration".Namespaces.NamespaceURIs,
                                                       REQ := FALSE,
                                                       ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                       NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes,
                                                       StatusList :=
"Productionline_Configuration".Namespaces.NamespaceStatusList,
                                                       NamespaceUrisCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                       Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;


        3:  // case 3, get an handle for server method OpenDoor

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodGetHandleList_Instance(
                                                 REQ := #Req,
                                                 ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                 ObjectNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".ObjectNodes,
                                                 MethodNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodNodes,
                                                 NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                 Done => #Done,
                                                 Busy => #Busy,
                                                 Error => #Error,
                                                 Status => #Status,
                                                 StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                 MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
                                                 Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                 NodeIDCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
                                                 NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
```

SCL

```
            IF #Done = TRUE THEN
                IF "Productionline_Data"."MethodListOpenDoor".MethodStatusList[0] = 0
THEN      //one method
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at MethodGetHandleList';
                END_IF;
            END_IF;
            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_MethodGetHandleList_Instance(
                                                    ObjectNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".ObjectNodes,
                                                    ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                    MethodNodeIDs :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodNodes,
                                                    StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                    MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
                                                    REQ := FALSE,
                                                    Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                    NodeIDCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
                                                    NamespaceIndexCount :=
"Productionline_Configuration".Namespaces.NamespaceCount,
                                                    NamespaceIndexes :=
"Productionline_Configuration".Namespaces.ServerNamespaceIndexes);
            END_IF;

        4:  // case 4, call method OpenDoor

            IF #Init_Method_InputParameter = TRUE THEN
                #Init_Method_InputParameter := FALSE;
                // for our server method at Productionline, we set input parameters to 1
                "Productionline_Data".MethodListOpenDoor.Method.Inputs.Number := 1;
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodCall_Instance(
                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
```

SCL

```
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        MethodHdl :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls[0],
                                        Status =>
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0],
                                        InputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Inputs,
                                        OutputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Outputs,
                                        REQ := #Req,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        MethodResult := #MethodResult);

            #Output_MethodResult := #MethodResult;

            IF #Done = TRUE THEN
                IF #MethodResult < 16#8000_0000 THEN
                    #State := #State + 1;
                ELSE
                    //server method did not run successfully
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#' Error at server method';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
               #Mem_Status := "Productionline_Data".MethodListOpenDoor.MethodStatusList[0];
                #OPC_UA_MethodCall_Instance(
                                            ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            MethodHdl :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls[0],
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                            Status =>
"Productionline_Data"."MethodListOpenDoor".MethodStatusList[0],
                                            InputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Inputs,
                                            OutputArguments :=
"Productionline_Data"."MethodListOpenDoor"."Method".Outputs,
                                            MethodResult := #MethodResult);
            END_IF;


        5:  // case 5, release method handle
```

SCL

```
            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_MethodReleaseHandleList_Instance(
                                                    REQ := #Req,
                                                    ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                    MethodHdlCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
                                                    MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
                                                    Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                                    Done => #Done,
                                                    Busy => #Busy,
                                                    Error => #Error,
                                                    Status => #Status,
                                                    StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList);


            IF #Done = TRUE THEN
                #Mem_Status := #Status;
                IF "Productionline_Data"."MethodListOpenDoor".MethodStatusList[0] = 0 THEN
                    #State := #State + 1;
                ELSE
                    #State := 100;
                    //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                    #Set_REQ_To_FALSE := TRUE;
                    #Mem_Status := #Status;
                    #Output_Error_Message := WString#'Error at MethodGetHandleList';
                END_IF;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_MethodReleaseHandleList_Instance(
                                                        MethodHdls :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodHdls,
                                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                                        StatusList :=
"Productionline_Data"."MethodListOpenDoor".MethodStatusList,
                                                        MethodHdlCount :=
"Productionline_Configuration".MethodLists."MethodListOpenDoor".MethodCount,
                                                        REQ := FALSE,
                                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;
```

SCL

```
        6:  // case 6, disconnect from server

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;
            #OPC_UA_Disconnect_Instance(REQ := #Req,
                                        ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Done => #Done,
                                        Busy => #Busy,
                                        Error => #Error,
                                        Status => #Status);


            IF #Done = TRUE THEN
                #State := #State + 1;
                #Mem_Status := #Status;
            END_IF;

            IF #Error = TRUE THEN
                #State := 100;
                //In case 100, to set REQ of instruction "OPC_UA_Disconnect" to FALSE
                #Set_REQ_To_FALSE := TRUE;
                #Mem_Status := #Status;
                #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);

            END_IF;


        7: //case = 7, function block has run successfully

            #Output_Done := TRUE;
            #Output_Busy := FALSE;
            #State := 0;
            #Mem_Input_REQ := FALSE;
            #Run := FALSE;
            #Started := FALSE;


        99: // ERROR handling
            #Output_Error := TRUE;
            #Output_Status := #Mem_Status;
            #State := 0;
            #Output_Busy := FALSE;
            #Run := FALSE;
            #Mem_Input_REQ := FALSE;
            #Started := FALSE;
```

SCL

```
        100:    // ERROR handling, disconnect from server to release resources (handles)
            IF #Set_REQ_To_FALSE = TRUE THEN
                #Set_REQ_To_FALSE := FALSE;

                //set REQ to definitely to FALSE
                #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                            REQ := FALSE,
                                            Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout);
            END_IF;

            IF #Busy = FALSE THEN
                #Req := TRUE;
            ELSE
                #Req := FALSE;
            END_IF;

            #OPC_UA_Disconnect_Instance(ConnectionHdl :=
"Productionline_Configuration".Connection.ConnectionHdl,
                                        REQ := #Req,
                                        Timeout :=
"Productionline_Configuration".Connection.ConnectInfo.SessionTimeout,
                                        Busy => #Busy,
                                        Done => #Done,
                                        Error => #Error,
                                        Status => #Status);


            IF #Done = TRUE THEN
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

            IF #Error = TRUE THEN
                #Output_Error_Message := WString#'Error handling: Resources cannot be
released!';
                #Output_Error := TRUE;
                #Output_Status := #Mem_Status;
                #State := 0;
                #Output_Busy := FALSE;
                #Run := FALSE;
                #Mem_Input_REQ := FALSE;
                #Started := FALSE;
            END_IF;

    END_CASE;

END_IF;
```

Server-Anweisungen

OPC_UA_ServerMethodPre: Vorbereitung des Server-Methodenaufrufs

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_ServerMethodPre" gilt für S7-1500-CPUs ab Firmware-Version V2.5.

Beschreibung

Dieses Kapitel beschreibt die Anweisung "OPC_UA_ServerMethodPre".

Weil die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" immer pärchenweise im Anwenderprogramm aufzurufen sind, beachten Sie auch das Kapitel zur Anweisung "OPC_UA_ServerMethodPost".

Parameter für "OPC_UA_ServerMethodPre"

Tabelle 4-92 Die Parameter der Anweisung "OPC_UA_ServerMethodPre"

Parameter	Deklaration	Datentyp	Bedeutung
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlercodes für den Status".
UAMethodCalled	Output	BOOL	Die bereitgestellte Methode ist von einem OPC UA-Client aufgerufen worden.
UAMethod_InParameters	InOut	VARIANT	Zeiger auf eine Variable, die die Eingangsparameter für die bereitgestellte Methode enthält.

Fehlercodes für den Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

Die folgende Tabelle beschreibt die unterschiedlichen Kategorien von Fehlercodes.

Tabelle 4-93 Fehlercodes für den Status

Fehlercode (hexadezimale Werte)	Erläuterung
0000_0000	Anweisung erfolgreich beendet.
8xxx_xxxx	OPC UA spezifischer Fehler
Axxx_xxxx	PLCopen spezifischer Fehler
B080_C300	Unzureichende Ressourcen
B08x_yz00	SIMATIC-spezifischer Fehler
Weitere Fehlercodes siehe AUTOHOTSPOT	

Deklaration der Variablen

Deklarieren Sie eine Instanz der Anweisung "OPC_UA_ServerMethodPre" sowie die Variablen, mit denen Sie die Parameter der Anweisung versorgen, siehe auch Beispielprogramm zum Bereitstellen einer Methode für OPC UA-Clients (Seite 3997).

Für die Deklaration sind die folgenden Punkte wichtig:

- Legen Sie die Anweisung "OPC_UA_ServerMethodPre" als Multiinstanz im aufrufenden Funktionsbaustein an.

  Hinweis

  Name der Multiinstanz

  Die Multiinstanz muss zwingend den Namen "OPC_UA_ServerMethodPre_Instance" haben, sonst wird keine Methode auf dem Server angelegt.

  Dazu ziehen Sie per Drag & Drop die Anweisung aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Server" in den Editor.
  Klicken Sie dann auf "Multiinstanz".

- Wenn die Server-Methode einen oder mehrere Eingangsparameter besitzt, dann müssen Sie eine Variable mit dem Namen "**UAMethod_InParameters**" deklarieren.
  Legen Sie zunächst einen PLC-Datentyp (UDT) für die Eingangsparameter der Server-Methode an.
  Verwenden Sie dann diesen UDT für die Variable "UAMethod_InParameters".
  Im Beispiel wird der Datentyp "UDT_OpenDoorInArguments" genannt und enthält das Element Number.
  Alternative:
  Sie können der Variablen "**UAMethod_InParameters**" auch den Datentyp "Struct" zuweisen. Legen Sie dann die Komponenten dieses Datentyps entsprechend der Eingangsparameter der Server-Methode an (gleiche Namen und Datentypen).

OPC_UA_ServerMethodPre_Instance	OPC_UA_ServerMethodPre
Input	
Output	
Done	Bool
Busy	Bool
Error	Bool
Status	DWord
UAMethod_Called	Bool
InOut	
UAMethod_InParameters	Variant
Static	
UAMethod_InParameters	"UDT_OpenDoorInArguments"
Number	Int

Aufruf der Anweisung

Die Anweisung "OPC_UA_ServerMethodPre" fragt beim Betriebssystem nach, ob die Server-Methode aufgerufen wurde.

Wenn die Server-Methode vom Client aufgerufen wurde, dann stellt die Anweisung "OPC_UA_ServerMethodPre" die Eingangsparameter für die Server-Methode bereit.

Zuordnung von Datentypen (SIMATIC - OPC UA)

Für die Eingangs- und Ausgangsparameter von Methoden beachten Sie die Erläuterungen zu den Regeln verwendbarer Datentypen (AUTOHOTSPOT).

Hinweis

Versorgung von strukturierten Datentypen mit geschachtelten Arrays

Wenn ein strukturierter Datentyp (Struct/UDT) ein Array enthält, stellt der OPC UA-Server keine Information über die Länge dieses Arrays bereit.

Falls Sie eine solche Struktur z. B. als Eingangs- oder Ausgangsparameter einer Server-Methode verwenden, dann müssen Sie sicherstellen, dass das geschachtelte Array beim Aufruf der Methode mit der richtigen Länge versorgt wird.

Wenn Sie diese Regel nicht berücksichtigen, schlägt die Methode fehl mit dem Fehlercode "BadInvalidArgument".

OPC_UA_ServerMethodPost: Nachbereitung des Server-Methodenaufrufs

Gültigkeit

Die folgende Beschreibung der Anweisung "OPC_UA_ServerMethodPost" gilt für S7-1500-CPUs ab Firmware-Version V2.5.

Beschreibung

Dieses Kapitel beschreibt die Anweisung "OPC_UA_ServerMethodPost".

Weil die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" immer pärchenweise im Anwenderprogramm aufzurufen sind, beachten Sie auch das Kapitel zur Anweisung "OPC_UA_ServerMethodPre".

Parameter für "OPC_UA_ServerMethodPost"

Tabelle 4-94 Die Parameter der Anweisung "OPC_UA_ServerMethodPost"

Parameter	Deklaration	Datentyp	Bedeutung
Done	Output	BOOL	Stand der Bearbeitung: • 0: Ausführung der Anweisung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Ausführung der Anweisung ohne Fehler abgeschlossen
Busy	Output	BOOL	Parameter zum Stand der Bearbeitung: • 0: Anweisung nicht in Bearbeitung • 1: Anweisung momentan in Bearbeitung
Error	Output	BOOL	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "Status".
Status	Output	DWORD	Ursache für den Fehler, siehe unten "Fehlercodes für den Status"
UAMethod_Result	Input	DWORD	Fehlercodes für den OPC UA-Server, bereitgestellt vom Anwenderprogramm. Empfehlung: Verwenden Sie für die Rückmeldung von Fehlern Codes, die mit 0xFF beginnen. Für OPC UA sind folgende Bereiche definiert: • Good: 0x0000_0000 bis 0x3FFF_ FFFF • Uncertain: 0x4000_0000 bis 0x7FFF_FFFF • Bad: 0x8000_0000 bis 0xFFFF_FFFF Je nach Client kann es sein, dass die Codes der Bereiche "Good" und "Uncertain" nicht ausgegeben werden.
UAMethod_Finished	Input	BOOL	Setzen Sie den Wert des Parameters auf TRUE, wenn die bereitgestellte Methode ausgeführt wurde.
UAMethod_OutParameters	InOut	VARIANT	Zeiger auf eine Variable, die die Ausgangsparameter der bereitgestellten Methode enthält.

Fehlercodes für den Status

Der Parameter "Status" informiert Sie über Fehler, die bei der Ausführung der Anweisung auftreten können.

Die folgende Tabelle beschreibt die unterschiedlichen Kategorien von Fehlercodes.

Tabelle 4-95 Fehlercodes für den Status

Fehlercode (hexadezimale Werte)	Erläuterung
0000_0000	Anweisung erfolgreich beendet.
8xxx_xxxx	OPC UA spezifischer Fehler
Axxx_xxxx	PLCopen spezifischer Fehler
B080_C300	Unzureichende Ressourcen
B08x_yz00	SIMATIC-spezifischer Fehler
Weitere Fehlercodes siehe AUTOHOTSPOT	

Deklaration der Variablen

Deklarieren Sie eine Instanz der Anweisung "OPC_UA_ServerMethodPost" sowie die Variablen, mit denen Sie die Parameter der Anweisung versorgen, siehe auch Beispielprogramm zum Bereitstellen einer Methode für OPC UA-Clients (Seite 3997).

Für die Deklaration sind die folgenden Punkte wichtig:

- Legen Sie die Anweisung "OPC_UA_ServerMethodPost" als Multiinstanz im aufrufenden Funktionsbaustein an.

  Hinweis

  Name der Multiinstanz

  Die Multiinstanz muss zwingend den Namen "OPC_UA_ServerMethodPost_Instance" haben, sonst wird keine Methode auf dem Server angelegt.

  Dazu ziehen Sie per Drag & Drop die Anweisung aus dem Ordner "Anweisungen > Kommunikation > OPC UA > OPC UA Server" in den Editor. Klicken Sie dann auf "Multiinstanz".

- Wenn die Server-Methode einen oder mehrere Ausgangsparameter besitzt, dann müssen Sie eine Variable mit dem Namen "**UAMethod_OutParameters**" deklarieren.
  Legen Sie zunächst einen PLC-Datentyp (UDT) für die Ausgangsparameter der Server-Methode an.
  Verwenden Sie dann diesen UDT für die Variable "UAMethod_OutParameters".
  Im Beispiel wird der Datentyp "UDT_OpenDoorOutArguments" genannt; einziger Ausgangsparameter ist Result.
  Alternative:
  Sie können der Variablen "**UAMethod_OutParameters**" auch den Datentyp "Struct" zuweisen. Legen Sie dann die Komponenten dieses Datentyps entsprechend der Ausgangsparameter der Server-Methode an (gleiche Namen und Datentypen).

OPC_UA_ServerMethodPost_Instance	OPC_UA_ServerMethodPost
Input	
UAMethod_Result	DWord
UAMethod_Finished	Bool
Output	
Done	Bool
Busy	Bool
Error	Bool
Status	DWord
InOut	
UAMethod_OutParameters	Variant
Static	
UAMethod_OutParameters	"UDT_OpenDoorOutArguments"
Result	Int

Bild 4-2 Deklaration der Variablen

Aufruf der Anweisung

Mit der Anweisung "OPC_UA_ServerMethodPost" informieren Sie das Betriebssystem darüber, dass die Server-Methode ausgeführt wurde und die Werte der Ausgangsparameter gültig sind.

Zuordnung von Datentypen (SIMATIC - OPC UA)

Für die Eingangs- und Ausgangsparameter von Methoden beachten Sie die Erläuterungen zu den Regeln verwendbarer Datentypen im Abschnitt "Mapping von Datentypen".

Hinweis

Versorgung von strukturierten Datentypen mit geschachtelten Arrays

Wenn ein strukturierter Datentyp (Struct/UDT) ein Array enthält, stellt der OPC UA-Server keine Information über die Länge dieses Arrays bereit.

Falls Sie eine solche Struktur z. B. als Eingangs- oder Ausgangsparameter einer Server-Methode verwenden, dann müssen Sie sicherstellen, dass das geschachtelte Array beim Aufruf der Methode mit der richtigen Länge versorgt wird.

Wenn Sie diese Regel nicht berücksichtigen, schlägt die Methode fehl mit dem Fehlercode "BadInvalidArgument".

Beispielprogramm zum Bereitstellen einer Methode für OPC UA-Clients

Beispielprogramm für eine Server-Methode

Dieses Kapitel enthält den vollständigen Programmcode für das Beispielprogramm "OpenDoor".

Das Beispiel zeigt, wie ein Anwenderprogramm die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" verwendet.

Die Anweisungen sind in den Kapiteln OPC_UA_ServerMethodPre: Vorbereitung des Server-Methodenaufrufs (Seite 3990) und OPC_UA_ServerMethodPost: Nachbereitung des Server-Methodenaufrufs (Seite 3993) beschrieben.

Das Programm stellt eine Server-Methode für OPC UA-Clients bereit: Das Programm setzt den Ausgangsparameter "Result" auf den Wert 1, wenn der Eingangsparameter "Number" den Wert "1" hat.

Um das Beispiel einfach und übersichtlich zu halten, wurde auf eine detaillierte Fehlerauswertung (Parameter "Status") verzichtet.

Programmstruktur

Das Programm gliedert sich in die folgenden Abschnitte:

1. Aufruf der Anweisung "OPC_UA_ServerMethodPre", um festzustellen, ob die Server-Methode von einem Client aufgerufen wurde.
2. Wurde die Server-Methode aufgerufen, dann wird die Server-Methode ausgeführt. Sie definiert die eigentliche Funktionalität bei einem Aufruf der Methode durch einen OPC UA-Client.
3. Wenn die Server-Methode beendet ist, dann wird die Anweisung "OPC_UA_ServerMethodPost" aufgerufen. Dieser Abschnitt informiert das Betriebssystem darüber, dass die Server-Methode ausgeführt wurde.

Deklaration

Das folgende Bild zeigt die Deklaration der lokalen Variablen für das Beispielprogramm:

OpenDoor

	Name	Datentyp	Offset	Defaultwert
1	Input			
2	<Hinzufügen>			
3	Output			
4	<Hinzufügen>			
5	InOut			
6	doorLocked	Bool	0.0	TRUE
7	Static			
8	UAMethod_OutParameters	"UDT_OpenDoorOutArguments"	2.0	
9	UAMethod_InParameters	"UDT_OpenDoorInArguments"	4.0	
10	Method_Result	DWord	6.0	16#0
11	Method_Finished	Bool	10.0	false
12	Started	Bool	10.1	false
13	Error_Message	WString	12.0	WSTRING#''
14	OPC_UA_ServerMethodPre_Instance	OPC_UA_ServerMethodPre		
15	OPC_UA_ServerMethodPost_Instance	OPC_UA_ServerMethodPost		
16	Temp			
17	Method_Called	Bool	0.0	
18	Pre_Done	Bool	0.1	
19	Pre_Error	Bool	0.2	
20	Post_Done	Bool	0.3	
21	Post_Error	Bool	0.4	

Programm

Das folgende Programm zeigt, wie Sie OPC UA-Anweisungen nutzen, um eine Methode für OPC UA-Clients bereitzustellen, die in Ihrem Anwenderprogramm ausgeführt wird (Server-Methode).

Aufruf der Anweisung "OPC_UA_ServerMethodPre"

Zuerst wird die Anweisung "OPC_UA_ServerMethodPre" aufgerufen, um beim Betriebssystem zu erfragen, ob die Server-Methode von einen OPC UA-Client aufgerufen wurde.

Wurde die Server-Methode aufgerufen, dann ist die Variable "#Method_Called" gleich "TRUE".

Wurde zudem die Anweisung "OPC_UA_ServerMethodPre" erfolgreich ausgeführt (#Pre_Done = TRUE), dann wird die Variable "#Started" auf den Wert "TRUE" gesetzt.

Server-Methode

Wenn die Variable "#Started" den Wert "TRUE" hat, wird nun das eigentliche Anwendungsprogramm ausgeführt.

In diesem Abschnitt haben Sie alle programmtechnischen Möglichkeiten: Sie können Prozesswerte einlesen oder Prozesswerte ausgeben, auf globale Datenbausteine zugreifen, Funktionen und Funktionsbausteine aufrufen, usw.

Das Anwenderprogramm kann mehrere Zyklen dauern.

Um das Ende des Anwenderprogramms zu signalisieren, setzen Sie die Variable "#Method_Finished" auf "TRUE".

Mit "#Method_Result" übermitteln Sie einen selbst definierten Fehlercode, der am Bausteinparameter "UAMethod_Result" der Anweisung "OPC_UA_ServerMethodPost" verwendet wird.

Aufruf der Anweisung "OPC_UA_ServerMethodPost"

Die Variable "#Method_Finished" wurde (Server-Methode) gesetzt, um den Zustand zu speichern, dass das Anwenderprogramm ausgeführt wurde.

Diese Variable wird in der Anweisung "OPC_UA_ServerMethodPost" genutzt, um beim Betriebssystem mitzuteilen, ob die Server-Methode ausgeführt wurde oder nicht.

Das Betriebssystem übernimmt dann die Quittierung zum Client, der die Methode aufgerufen hat.

SCL

```
REGION PRE

    IF #Started = FALSE AND #Method_Finished = FALSE THEN

        #OPC_UA_ServerMethodPre_Instance(Done => #Pre_Done,
                                         Error => #Pre_Error,
                                         UAMethod_Called => #Method_Called,
                                        UAMethod_InParameters := #UAMethod_InParameters);
      IF #Pre_Error THEN
            #Error_Message := WString#'Error at OPC_UA_ServerMethodPre';
        END_IF;
        IF #Pre_Done AND #Method_Called THEN
            #Started := TRUE;
            #Error_Message := WString#'';
        END_IF;
    END_IF;

END_REGION

REGION FUNCTIONALITY

    // The purpose of this demo method is to demonstrate the basic structure of a server
method
    IF #Started = TRUE THEN

        // This demo server method just checks if input parameter
#UAMethod_InParameters.Number is 1
        // If this is the case, the server method sets the ouput parameter.
        IF #UAMethod_InParameters.Number = 1 THEN
            //set doorLocked to false, so that the door can be opened
            #doorLocked := FALSE;

            //set values for output parameters
            //this is just an example
            #UAMethod_OutParameters.Result := 1;
            #Method_Result := 16#0000_0000;
            #Method_Finished := TRUE;
            #Started := FALSE;
        ELSE
           #Error_Message := WString#'Input-Parameter Number must be 1 to run this example
server method';
            //we send back an error code to the calling OPC UA client
            //for our example, we define 16#FFFF_FFFF = "Input-Parameter Number must be 1"
            #Method_Result := 16#FFFF_FFFF;
            #Started := FALSE;
            #Method_Finished := TRUE;
        END_IF;
    END_IF;

END_REGION

REGION POST

    IF #Method_Finished = TRUE THEN
        #OPC_UA_ServerMethodPost_Instance(UAMethod_Result := #Method_Result,
```

SCL

```
                                        UAMethod_Finished := #Method_Finished,
                                        Done => #Post_Done,
                                        Error => #Post_Error,
                                    UAMethod_OutParameters := #UAMethod_OutParameters);

        IF #Post_Error = TRUE THEN
            #Error_Message := WString#'Error at OPC_UA_ServerMethodPost';
            #Method_Finished := FALSE;
        END_IF;
        IF #Post_Done = TRUE THEN
            #Method_Finished := FALSE;
        END_IF;
    END_IF;

END_REGION
```

Methode löschen

Wenn Sie eine Server-Methode nicht mehr zur Verfügung stellen und löschen wollen, dann müssen Sie nicht nur den Aufruf des entsprechenden FBs entfernen, sondern auch den Instanz-Datenbaustein löschen.

Wird der Instanz-Datenbaustein nicht gelöscht, dann ist die Methode im Adressraum sichtbar, die Variablen werden nicht versorgt.

4.1.6.4 Webserver

WWW: Anwenderseiten synchronisieren

WWW: Anwenderseiten synchronisieren

Beschreibung

Die Anweisung WWW initialisiert den Webserver der CPU bzw. synchronisiert anwenderdefinierte Webseiten, kurz: "Anwenderseiten", mit dem Anwenderprogramm in der CPU.

Anwenderseiten bilden zusammen mit dem Webserver der CPU die Möglichkeit, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen.

Mit Hilfe von Scriptanweisungen (z. B. Javascript) und HTML-Code in Anwenderseiten können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an die CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen. Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung "WWW" im Anwenderprogramm aufrufen.

Initialisierung

Anwenderseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Die Datenbausteine mit codierten Webseiten werden Fragment-DBs genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin Anwenderseiten codiert sind. Durch die Anweisung "WWW" z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die Anwenderseiten über einen Webbrowser erreichbar.

Synchronisierung

Wenn das Anwenderprogramm mit den Anwenderseiten interagieren soll, muss die Anweisung "WWW" im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragendenden Webbrowser zusammenstellen und zurücksenden

In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer vom Webbrowser angeforderten Webseite dem Webserver mitgeteilt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WWW":

Parameter	Deklaration	Datentyp	Speicherbe-reich	Beschreibung
CTRL_DB	Input	DB_WWW	E, A, M, D, L oder Konstante	Datenbaustein, der die Anwenderseiten beschreibt (Web-Control-DB)
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter Übersicht über die gültigen Datentypen (Seite 243).

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten. Es stehen keine Webseitenanforderungen an, die vom Anwenderprogramm freigegeben werden müssen.
00xy	x: zeigt an, ob bei der Initialisierung des Web-Control-DBs (CTRL_DB) ein Fehler aufgetreten ist: x=0: keine Fehler aufgetreten. x=1: Fehler aufgetreten. Der Fehler ist im Byte "CTRL_DB.last_error" des Web-Control-DBs kodiert, siehe Beschreibung zum Web-Control-DB. y: Nummer des anstehenden Requests. Mehrere Requests sind möglich (z. B. Request "0" und "1" stehen an: y="3". y="1": Request "0" y="2": Request "1" y="4": Request "2" y="8": Request "3"
803A	Der angegebene Web-Control-DB ist auf der CPU nicht vorhanden.
8081	Falsche Version oder falsches Format des Web-Control-DBs.
80C1	Es sind keine Ressourcen vorhanden, um die Web-Applikation zu initialisieren, z. B. weil nur zwei oder vier Web-Applikationen laufen dürfen.

Beispiel

Das Beispiel finden Sie hier: Programmbeispiel für WWW (Seite 4004).

Weitere Informationen und den Programmcode zu dem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmbeispiel für WWW

Einleitung

In dem folgenden Beispiel synchronisieren Sie eine Anwender definierte Website mit dem Programmbeispiel in einer CPU S7-1500 und testen das Programmbeispiel über den Webserver.

Voraussetzung

Zur Ablage der Daten legen Sie in einem globalen Datenbaustein neunzehn Variablen an.

SLI_gDB_www

	Name	Data type	Start value
1	Static		
2	executeWWW	Bool	false
3	returnValueWWW	Word	16#0
4	executeSetOB	Bool	false
5	setDone	Bool	false
6	start	Bool	false
7	stop	Bool	false
8	engineStarted	Bool	false
9	reset	Bool	false
10	tempReset	Bool	false
11	openValve	Bool	false
12	closeValve	Bool	false
13	valveOutput	Bool	false
14	flowrate	Int	1
15	tankLevel	Int	0
16	alarm	Int	0
17	tankLevelOverflow	Int	160
18	tankLevelMax	Int	150
19	tankLevelMidth	Int	75
20	tankLevelMin	Int	5

Parameter verschalten - im FB "SLI_FB_www"

Die folgenden Verschaltungen legen Sie in einem FB "SLI_FB_www" an. Den FB rufen Sie in einem OB1 auf.

Netzwerk 1: Die Parameter der Anweisung "WWW" verschalten Sie wie folgt.

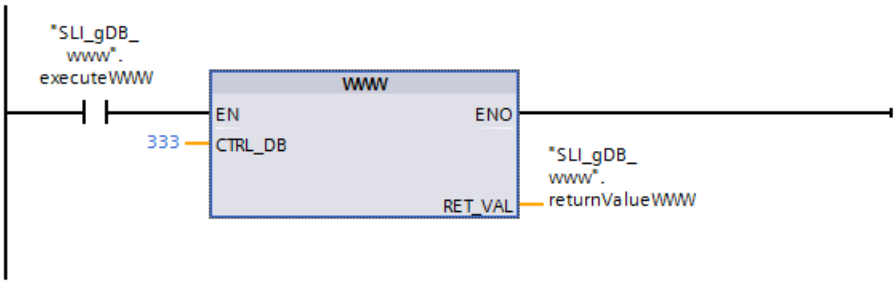

Netzwerk 2: Das Programm für die Website legen Sie in dem FC "SLI_FC_MainWebProg_www" an. Danach rufen Sie den FC "SLI_FC_MainWebProg_www" im Netzwerk 2 des FBs auf.

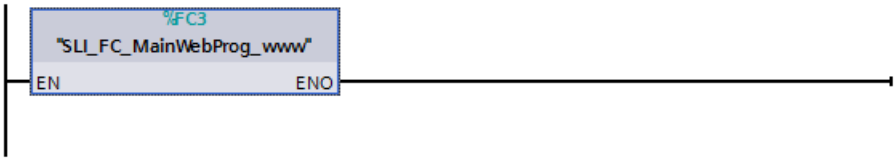

Parameter verschalten - im FC "SLI_FC_SetTOD_www"

In einem FC "SLI_FC_SetTOD_www" erstellen Sie ein Programm zur Verwendung eines Uhrzeitalarm-OBs (OB11). Sie legen folgende Verschaltungen an:

Netzwerk 1: In dem FC verschalten Sie die folgenden lokalen Variablen mit den Anweisungen "RD_SYS_T" und "T_ADD".

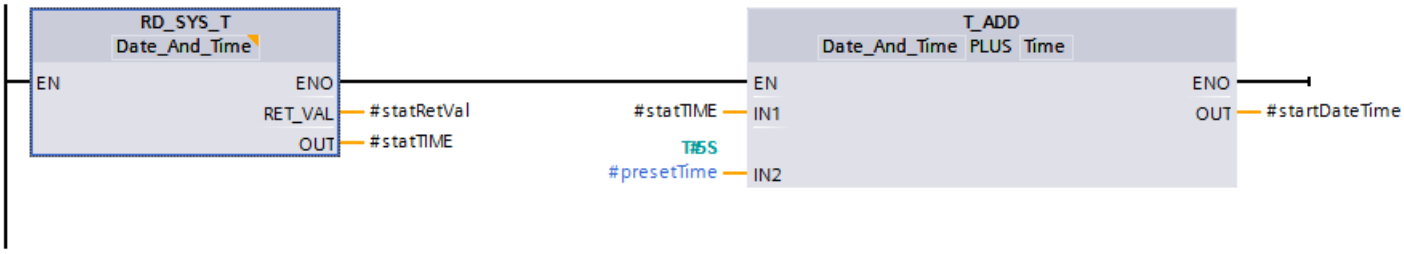

Netzwerk 2: Im Folgenden verschalten Sie die Variablen zum Einstellen des Uhrzeitalarm-OBs.

Die lokale Variable "#startDateTime" (Datentyp "Date_And_Time") übergibt dem Uhrzeitalarm-OB den Startzeitpunkt.

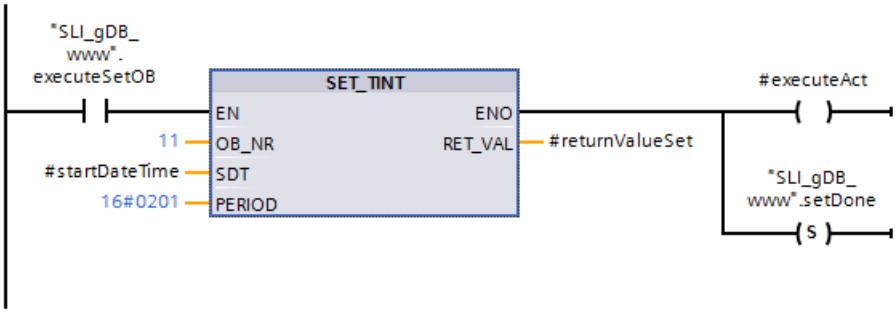

Netzwerk 3: Im Folgenden verschalten Sie die Variablen zum Aktivieren des Uhrzeitalarm-OBs.

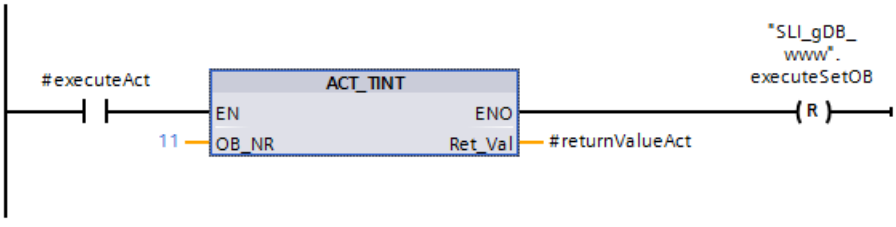

Parameter verschalten - im OB "SLI_todOB_www"

In dem Uhrzeitalarm-OB "SLI_todOB_www" (OB11) legen Sie die folgenden Verschaltungen zur zyklischen Erhöhung der Variable "tankLevel" an.

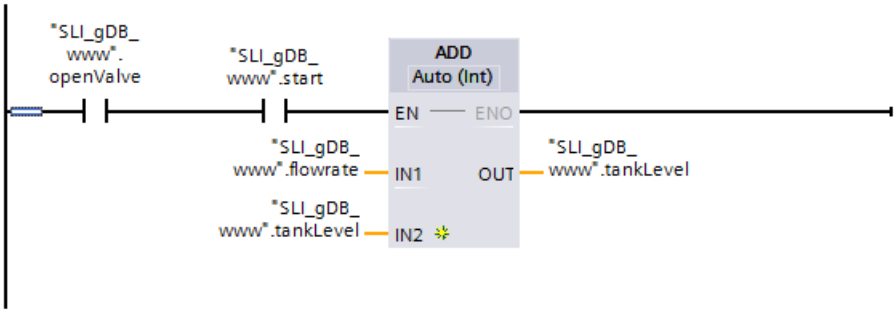

Parameter verschalten - im FC "SLI_FC_MainWebProg_www"

Die folgenden Verschaltungen legen Sie in dem FC "SLI_FC_MainWebProg_www" an.

Das Programm im "SLI_FC_MainWebProg_www" ermöglicht die Bedienung eines Motors, eines Ventils und der Tankbefüllung. Zudem wird die Ausgabe verschiendener Alarmzustände ermöglicht. Die Bedienung des Programms erfolgt über den Webserver.

Netzwerk 1: Den FC "SLI_FC_SetTOD_www" rufen Sie in dem FC "SLI_FC_MainWebProg_www" auf.

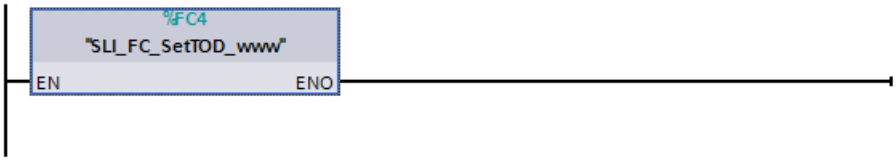

Netzwerk 2: Im Folgenden verschalten Sie die Variablen zum Öffnen des Ventils.

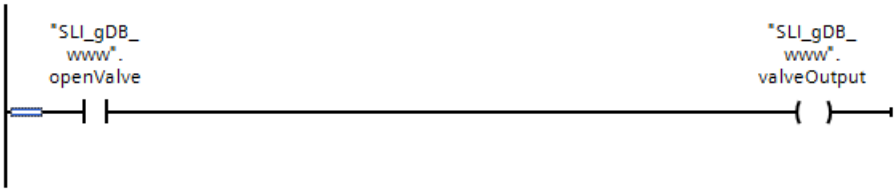

Netzwerk 3: Im Folgenden verschalten Sie die Variablen zum Starten des Motors.

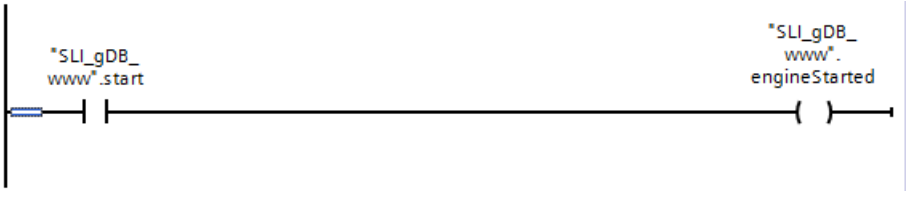

Netzwerk 4: Im Folgenden verschalten Sie die Variablen zur Abfrage des Alarmzustand "1".

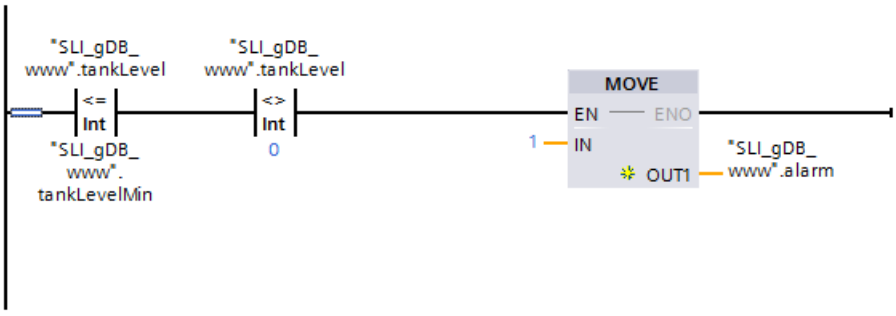

Netzwerk 5 bis 8: Im Folgenden verschalten Sie die Variablen zur Abfrage des Alarmzustand "2".

Verschalten Sie Variablen zur Abfrage der weiteren Alarmzustände nach gleichem Schema.

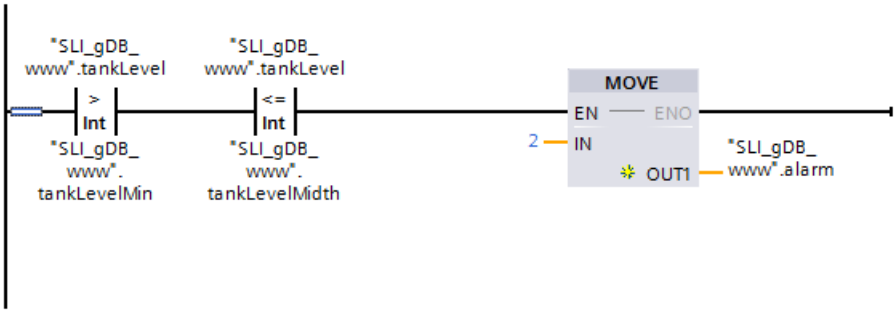

Netzwerk 9: Im Folgenden verschalten Sie die Variablen zum Rücksetzen des Motors und der Tankbefüllung.

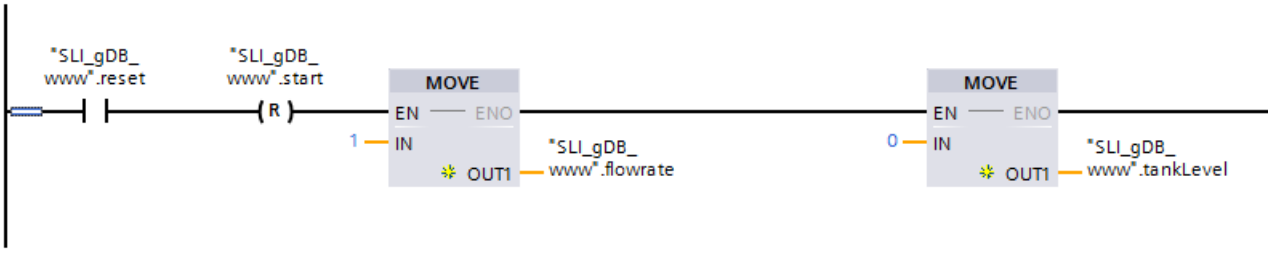

Netzwerk 10: Im Folgenden verschalten Sie die Variablen zum Rücksetzen aller restlichen Werte (BOOL).

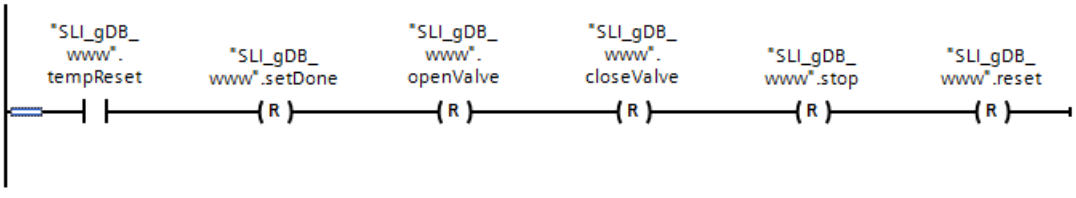

CPU Eigenschaften einstellen

Bei den CPU Eigenschaften nehmen Sie folgende Einstellungen vor:

- Unter "... > PROFINET Schnittstelle > Zugriff auf den Webserver" aktivieren Sie den Webserver-Zugang.
- Unter "CPU Eigenschaften > Webserver" aktivieren Sie den Webserver für das Modul.
- Im Abschnitt "Benutzerverwaltung" erstellen Sie einen neuen Nutzer mit Lese- und Schreibrechten.
- Im Abschnitt "Anwenderseiten" nehmen Sie folgende Einstellungen vor:
  - Sie stellen den "HTML-Dateipfad" ein.

    Hinweis

    Der Ablagepfad sollte der gleiche wie der Pfad der Sample Library for Instructions (SLI) sein. Der Ordnername lautet "SLI_html". Zum Beispiel: "C:\TIA_library\SLI_html".

  - Sie geben für die "Start HTML-Seite" den Text "index.html" ein.
  - Sie geben für die Anwendung den Namen "WWW sample" ein.
  - Sie klicken auf die Schaltfläche "Baustein erzeugen". Der Web-DB (333) und der Fragment-DB (334) werden daraufhin erstellt.

Ergebnis für WWW

Wenn der Schließerkontakt ("executeWWW") den Signalzustand "TRUE" liefert, wird die Anweisung "WWW" ausgeführt. Über den Eingangsparameter CTRL_DB ("333") ist die Nummer des Datenbausteins hinterlegt, welcher die Anwender definierte Website beschreibt. Die Anweisung "WWW" initialisiert die Anwender definierte Website im Webserver der CPU und synchronisiert absofort den Inhalt mit dem Programmbeispiel.

Am Ausgangsparameter RET_VAL ("returnValueWWW") wird angezeigt, dass die Verarbeitung ohne Fehler verläuft.

SLI_gDB_www

	Name	Data type	Start value	Monitor value
1	Static			
2	executeWWW	Bool	false	TRUE
3	returnValueWWW	Word	16#0	16#0000
4	executeSetOB	Bool	false	FALSE
5	setDone	Bool	false	TRUE
6	start	Bool	false	TRUE
7	stop	Bool	false	FALSE
8	engineStarted	Bool	false	TRUE
9	reset	Bool	false	FALSE
10	tempReset	Bool	false	FALSE
11	openValve	Bool	false	TRUE
12	closeValve	Bool	false	FALSE
13	valveOutput	Bool	false	TRUE
14	flowrate	Int	1	3
15	tankLevel	Int	0	3
16	alarm	Int	0	1
17	tankLevelOverflow	Int	160	160
18	tankLevelMax	Int	150	150
19	tankLevelMidth	Int	75	75
20	tankLevelMin	Int	5	5

Ergebnis für den Webserver

Den Webserver rufen Sie über den "Internet Browser > IP des Webservers" auf. Sie melden sich mit Ihrem Benutzernamen und -Passwort an.

Im Webserver wird unter "Anwenderseiten" ein Link auf die Anwender definierte Website angezeigt.

Customer pages

Homepage of the application WWW sample:

Auf der Website des Programmbeispiels wird Ihnen ermöglicht, Variablen auszugeben als auch neue Werte den Variablen zu übergeben.

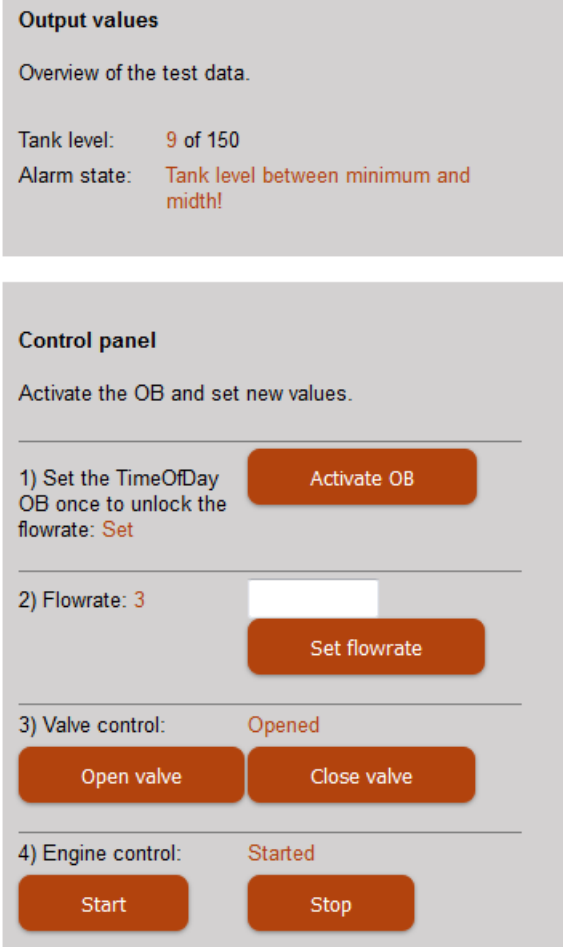

Genutzte Funktionen im HTML-Dokument

Im Folgenden sehen Sie Auszüge zu den im Programmbeispiel genutzten Funktionen, über die im HTML-Dokument die Variablen dem Quellcode übergeben werden. Für jede Funktion wird im HTML-Dokument eine Variable über einen Aufruf implementiert und an entsprechender Stelle im HTML-Dokument über einen weiteren Aufruf verwendet.

Funktion	Zum Aufruf im HTML-Dokument	Zur Ausgabe im HTML-Element
Datenausgabe:	`<!-- AWP_In_Variable Name='"SLI_gDB_www".tankLevel' -->`	`:="SLI_gDB_www".tankLevel:`
Enumeration (Wertersetzung):	`<!-- AWP_Enum_Def Name="OpValvValue" Values='0:"Closed", 1:"Opened"' -->`	`<!-- AWP_Enum_Ref Name='"SLI_gDB_www".valveOutput' Enum="OpValvValue" -->:="SLI_gDB_www".valveOutput:`

Funktion	Zum Aufruf im HTML-Dokument	Zur Ausgabe im HTML-Element

Wertänderung (von INT):	`<!-- AWP_In_Variable Name='"SLI_gDB_www".flowrate' -->`	`<form method="post" action="" onsubmit="return check();">`
		`<input type="text" name='"SLI_gDB_www".flowrate' size="10px"/>`
		`<input class="button1" type="submit" value="Set flowrate"/>`
		`</form>`

Wertänderung (von BOOL):	`<!-- AWP_Enum_Def Name="ResetValue" Values='0:"Off", 1:"On"' -->`	`<form method="post" action="">`
		`<input class="button1" type="submit" value="Reset"/>`
		`<input type="hidden" name='"SLI_gDB_www".reset' size="34px" value="1"/>`
		`</form>`

Programmcode

Weitere Informationen und den Programmcode zu oben genanntem Beispiel finden Sie hier: Sample Library for Instructions (Seite 615).

4.1.6.5 Kommunikationsprozessor

SIMATIC NET-CM/CP

S7-1500-CM/CP

Industrial Ethernet

Anweisungen für FTP-Dienste

FTP_CMD für FTP-Dienste

Übersicht FTP_CMD

Bedeutung

Mit der Anweisung FTP_CMD können Sie FTP-Verbindungen aufbauen und Dateien von und zu einem FTP-Server übertragen.

Hinweis

Bausteinversionen

Die Version V2.x des FTP_CMD können Sie in einer Station nur zusammen mit einer CPU V2.x und einem CP V2.x verwenden.

Sobald die Station eine CPU V1.x oder einen CP V1.x enthält, müssen Sie den FTP_CMD in der älteren Version V1.x verwenden (bspw. V1.4). Schalten Sie hierzu die Version der Bibliothek "SIMATIC NET CP" auf V3.4. Sie können dann eine ältere Version des Bausteins auswählen.

Der Datentransfer ist über FTP oder FTPS (gesichterte SSL-Verbindungen) möglich.

Hinweis

FTPS: Zertifikate abgleichen

FTPS erfordert den Abgleich der Zertifikate zwischen FTP-Server und FTP-Client. Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich. Importieren Sie im Zertifikatsmanager das Zertifikat des FTP-Servers als vertrauenswürdiges Zertifikat.

Funktionsweise

Die Anweisung FTP_CMD verweist auf einen Auftragsblock (ARG), in dem das FTP-Kommando spezifiziert wird. Je nach Typ des FTP-Kommandos (CMD) verwendet dieser Auftragsblock unterschiedliche Datenstrukturen zur Parametrierung. Für diese unterschiedlichen Strukturen stehen jeweils passende Datentypen (UDTs) zur Verfügung.

Die folgende Darstellung zeigt die Aufrufstruktur:

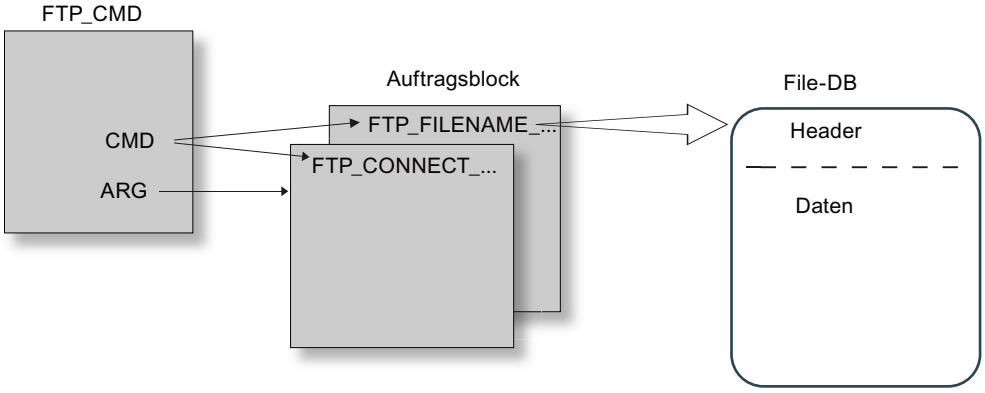

Auftragsblöcke

Für die Auftragsblöcke werden folgende Datenstrukturen verwendet:

- Verbindungsaufbau
  Für den Verbindungsaufbau stehen unterschiedliche Datenstrukturen für folgende Zugriffsarten zur Verfügung:
  - FTP_CONNECT_IPV4: Verbindungsaufbau mit IP-Adressen gemäß IPv4
  - FTP_CONNECT_IPV6: Verbindungsaufbau mit IP-Adressen gemäß IPv6
  - FTP_CONNECT_NAME: Verbindungsaufbau mit Server-Namen (DNS)
- Datentransfer
  Für den Datentransfer stehen zwei unterschiedliche Datenstrukturen zur Verfügung:
  - FTP_FILENAME: Datenstruktur für den Zugriff auf eine vollständige Datei
  - FTP_FILENAME_PART: Datenstruktur für den lesenden Zugriff auf einen Datenbereich

Datenübertragung im File_DB

Der Datentransfer erfolgt über Datenbausteine, die einen Header für Auftragsdaten sowie den Bereich für die Nutzdaten enthalten. Der Datenbaustein wird im Auftragspuffer angegeben.

Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen, um den FTP-Zugriff zu ermöglichen:

- Deaktivieren Sie bei allen als File-DB verwendeten Datenbausteinen das Attribut "Optimierter Bausteinzugriff".
- Nur bei Verwendunng einer CPU V1.x und eines CP V1.1.x:
  Deaktivieren Sie in den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz" die Option "PUT/GET-Kommunikation deaktivieren" (PUT/GET muss freigegeben sein).

Gültigkeit

Die Anweisung FTP_CMD kann mit folgenden Baugruppentypen eingesetzt werden:

- CP 1543-1

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

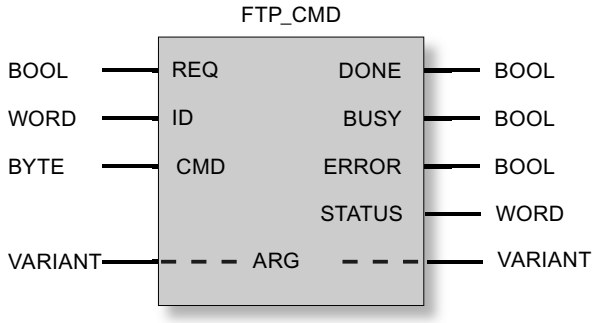

Siehe auch

Ausgangsparameter und Statusinformationen FTP_CMD (Seite 4023)

Eingangsparameter FTP_CMD (Seite 4016)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb (Seite 4026)

Eingangsparameter FTP_CMD

Erläuterung der Eingangsparameter

Sie versorgen die Anweisung FTP_CMD mit folgenden Eingangsparametern:

Tabelle 4-96 Formalparameter der Anweisung FTP_CMD - Eingangsparameter

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung / Bemerkung
REQ	Input	BOOL	E, A, M, D, L	Startet den Sendeauftrag bei einer steigenden Flanke.
ID *	INPUT	INT	1, 2 ... 64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
CMD *	INPUT	BYTE	Siehe nachfolgende Tabelle "Kommandos".	FTP-Kommando, das beim Aufruf der Anweisung ausgeführt werden soll. Wertebereiche für die FTP-Kommandotypen finden Sie im Anschluss an die Tabelle. Hinweis: Das hier angegebene FTP-Kommando muss identisch im Auftragsblock (Parameter ARG) angegeben werden. Wenn ein Kommando nicht von der CP-Firmware unterstützt wird, dann wird eine Fehlermeldung mit STATUS = $8F6B_H$ ausgegeben.
ARG *	INPUT	VARIANT	Siehe nachfolgende Tabelle "Kommandos".	Auftragsblock Verweist auf den Datenbereich mit den zum FTP-Kommando passenden Ausführungsparametern. Abhängig vom FTP-Kommando werden spezifische Datentypen (UDT) verwendet. Die UDTs werden nachfolgend angegeben. Für den hier anzugebenden Zeiger ist der Datentyp ANY nicht zulässig!

* Die Werte der Eingangsparameter "ID" und "CMD" überschreiben den Wert des Eingangsparameters "ARG".

FTP-Kommandos im Parameter "CMD"

Entnehmen Sie der folgenden Tabelle, welche Bedeutung die Kommandos des Parameters "CMD" haben und welche UDTs Sie zur Versorung der Auftragsblöcke verwenden.

Tabelle 4-97 Kommandotypen

CMD (Kommandotyp)	Relevante Auftragsblöcke / UDT	Bedeutung / Handhabung
0 (NOOP)	*	Der aufgerufene FC führt keine Aktionen aus. Die Statusanzeigen werden bei dieser Parameterversorgung wie folgt gesetzt: DONE=1; ERROR=0; STATUS=0
1 (CONNECT)	FTP_CONNECT_IPV4 FTP_CONNECT_IPV6 FTP_CONNECT_NAME	FTP-Verbindungsaufbau Mit diesem Kommando baut der FTP-Client eine FTP-Verbindung zu einem FTP-Server auf (Port 21). Die Verbindung steht unter der hier zugewiesenen Verbindungs-ID für alle weiteren FTP-Kommandos zur Verfügung. Daten werden dann mit dem für diesen Benutzer angegebenen FTP-Server ausgetauscht.
2 (STORE)	FTP_FILENAME	Mit diesem Funktionsaufruf wird ein Datenbaustein (File-DB) vom FTP-Client (S7-CPU) zum FTP-Server übertragen. Achtung: Falls die Datei (File-DB) auf dem FTP-Server schon vorhanden ist, wird diese überschrieben.
3 (RETRIEVE)	FTP_FILENAME	Mit diesem Funktionsaufruf wird eine Datei vom FTP-Server zum FTP-Client (S7-CPU) übertragen. Achtung: Falls der Datenbaustein (File-DB) beim FTP-Client schon eine Datei enthält, wird diese überschrieben.
4 (DELETE)	FTP_FILENAME	Mit diesem Funktionsaufruf löschen Sie eine Datei auf dem FTP-Server.
5 (QUIT)	*	Mit diesem Funktionsaufruf bauen Sie die über die ID benannte FTP-Verbindung ab.
6 (APPEND)	FTP_FILENAME	Ähnlich wie "STORE" speichert das Kommando "APPEND" (anhängen) eine Datei auf dem FTP-Server. Bei "APPEND" wird die Datei auf dem FTP-Server aber nicht überschrieben, sondern der neue zu speichernde Inhalt wird an die Datei angehängt. Falls die Datei auf dem FTP-Server nicht vorhanden ist, wird sie angelegt.
7 (RETR_PART)	FTP_FILENAME_PART	Mit dem Kommando "RETR_PART" (Teillänge lesen) können Sie einen Ausschnitt einer Datei vom FTP-Server anfordern. Bei sehr großen Dateien können Sie damit das Lesen auf den Teil beschränken, den Sie gerade benötigen. Dazu müssen Sie die Struktur dieser Datei kennen. Geben Sie den gewünschten Ausschnitt der Datei mithilfe der zwei Parameter "OFFSET" und "LEN" am FB 40 an.

***** Bei den Kommandotypen 0 (NOOP) und 5 (QUIT) muss ein beliebiger Auftragsblock (UDT) angegeben werden. Dieser wird nicht ausgewertet.

Siehe auch

Übersicht FTP_CMD (Seite 4013)

Auftragsblöcke für FTP_CMD

Bedeutung

Sie versorgen die Anweisung FTP_CMD über den Parameter ARG mit einem Auftragsblock. Die Struktur ist abhängig vom FTP-Kommandotyp. Indem Sie die vorgegebenen Datentypen (UDT) verwenden, erkennt die Anweisung den Typ des Auftragsblocks. Nachfolgend werden die jeweiligen Datentypen (UDT) für folgende Auftragsblöcke angegeben:

- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4
- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6
- FTP-Verbindungsaufbau mit Server-Name
- Schreib- und Lesezugriff sowie übrige FTP-Kommandos
- FTP-Kommando RETR_PART

Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4 wird die folgende Datenstruktur verwendet.

Tabelle 4-98 FTP_CONNECT_IPV4

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPCmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 4016). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	'benutzer'	Benutzername für das Login auf dem FTP-Server

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverIPaddr	IP_V4	ADDR(1) ... ADDR(4)	IP-Adresse des FTP-Servers als Array[1..4] of Byte, wobei jeweils 1 Byte einen Block der Adresse spezifiziert. Bsp.: ADDR(1) spezifizieren den ersten Adressblock (das erste Byte der Adresse).

Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6 wird die folgende Datenstruktur verwendet.

Tabelle 4-99 FTP_CONNECT_IPV6

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPCmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 4016). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverIPaddr	IP_V6	ADDR(1) ... ADDR(16)	IP-Adresse des FTP-Servers als Array[1..16] of Byte, wobei jeweils 2 Byte einen Block der Adresse spezifizieren. Bsp.: ADDR(1) + ADDR(2) spezifizieren den ersten Adressblock.

Auftragsblock für FTP-Verbindungsaufbau mit Server-Name

Für den FTP-Verbindungsaufbau mit Angabe eines Server-Namens wird die folgende Datenstruktur verwendet. Der Server-Name wird über DNS einer IP-Adresse zugewiesen.

Tabelle 4-100 FTP_CONNECT_NAME

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 4016). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverName	STRING[254]		IP-Adresse des FTP-Servers

Auftragsblock für Schreib- und Lesezugriff sowie übrige FTP-Kommandos

Für die FTP-Kommandos store, retrieve, delete und append wird die folgende Datenstruktur verwendet.

Tabelle 4-101 FTP_FILENAME

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	2, 3, 4, 6	FTP-Kommando "STORE / RETRIEVE / DELETE / APPEND" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 4016). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
DataBlockNumber	UINT		Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB.
LenFilename	UINT	0...1000	Der Parameter "LenFilename" zur Angabe der Gesamtlänge des Dateinamens wird nicht ausgewertet. Statt dessen wird die Längenangabe im String des Parameters "Filename" ausgewertet.
Filename	ARRAY[0..3] OF STRING[220]		Dateiname der Ziel- bzw. Quelldatei.

Auftragsblock für das FTP-Kommando RETR_PART

Für das FTP-Kommando RETR_PART wird die folgende Datenstruktur verwendet.

Tabelle 4-102 FTP_FILENAME_PART

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	7	FTP-Kommando "RETR_PART" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird.I. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 4016). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
Offset	DWORD		Offset in Byte, ab dem die Datei gelesen werden soll.
Length	DWORD		Teillänge in Byte, die ab dem in "OFFSET" angegebenen Wert gelesen werden. Besonderheiten: • Bei Angabe von "DW#16#FFFFFFFF" wird der verfügbare Rest der Datei gelesen. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET > Länge der Original-Datei: Länge der Ziel-Datei (ACT_LENGTH im File-DB): 0 Byte in der CPU. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET + LEN > Länge der Original-Datei (und LEN ≠ 0xFFFFFFFF): Länge der Ziel-Datei (ACT_LENGTH im File-DB): Verfügbare Bytes ab "OFFSET". Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt.

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
DataBlockNumber	UINT		Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB.
LenFilename	UINT	0...1000	Der Parameter "LenFilename" zur Angabe der Gesamtlänge des Dateinamens wird nicht ausgewertet. Statt dessen wird die Längenangabe im String des Parameters "Filename" ausgewertet.
Filename	ARRAY[0..3] OF STRING[220]		Dateiname der Ziel- bzw. Quelldatei.

Siehe auch

Übersicht FTP_CMD (Seite 4013)

Ausgangsparameter und Statusinformationen FTP_CMD

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "FTP_CMD" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Statusanzeigen auswerten

Hinweis

Beachten Sie für die Einträge mit der Codierung 8FxxH unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Tabelle 4-103 FTP_CMD: Bedeutung des Parameters STATUS in Zusammenhang mit DONE und ERROR

DONE	ERROR	STATUS	Bedeutung
0	0	0000_H	Es ist kein Auftrag in Bearbeitung.
1	0	0000_H	Der Auftrag ist fertig ohne Fehler.
0	0	7001_H	Der Auftrag wurde erstmalig angestoßen (BUSY=1)
0	0	7002_H	Der Auftrag läuft noch (BUSY=1)
0	0	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)
0	0	8183_H	Die Projektierung entspricht nicht den Auftragsparametern.
0	1	8401_H	unbekannter Fehler mögliche Ursachen: • Auf der Verbindung wurde ein Timeout erkannt. • Der FTP-Server hat die Verbindung abgebrochen. Abhilfe: • QUIT und CONNECT-Kommandos erneut senden um die Verbindung wieder einzurichten.
0	1	8402_H	Verbindung befindet sich in fehlerhaftem Zustand. Das Timeout der Verbindung kann überschritten sein oder der FTP-Server kann die Verbindung abgebaut haben. Abhilfe: Senden Sie erneut die Kommandos QUIT und CONNECT um die Verbindung wieder aufzubauen.
0	1	8403_H	Login ist fehlgeschlagen.
0	1	8404_H	FTP-Server ist nicht erreichbar.
0	1	8405_H	Übertragung ist fehlgeschlagen.
0	1	8406_H	Timeout bei aktuellem Vorgang
0	1	8407_H	Datei im FTP-Server wurde nicht gefunden.
0	1	8408_H	Übertragung nicht möglich.
0	1	8409_H	Datei konnte nicht geholt werden.
0	1	8410_H	Das Setzen des TCP Port für die Datenverbindung ist fehlgeschlagen.
0	1	8411_H	Offset-Angabe passt nicht.
0	1	8412_H	Fehler beim Wechsel der Verzeichnisangabe
0	1	8413_H	Fehler beim Daten Empfangen
0	1	8414_H	Fehler beim Daten Senden.
0	1	8415_H	CMD-Angabe (Kommandotyp) wurde vom Client abgewiesen.
0	1	8416_H	Verbindung wurde vom FTP-Server geschlossen.
0	1	8418_H	Fehler in den Nutzdaten. Mögliche Ursachen: • Dateiname ist leer. • Datenlänge ist "0" • usw.
0	1	8419_H	Es ist keine Socket Ressource zum Öffnen einer Datenverbindung vorhanden.
0	1	8420_H	Es ist keine Socket Ressource zum Öffenen einer Controlverbindung vorhanden.

DONE	ERROR	STATUS	Bedeutung
0	1	8421_H	Fehler beim Öffnen des zu lesenden File-DB.
0	1	8422_H	Fehler beim Öffnen des zu beschreibenden File-DB.
0	1	8423_H	Der Verbindungsaufbau zum FTP Server ist fehlgeschlagen.
0	1	8424_H	interner Fehler
0	1	8425_H	Formatfehler beim Domain-Namen
0	1	8426_H	Es stehen zu viele DNS-Anfragen an.
0	1	8427_H	Der angegebene DNS-Server konnte den spezifizierten Domain-Namen niocht zuordnen.
0	1	8428_H	Es ist keine Verbindungs-Ressource verfügbar.
0	1	8429_H	Unbekannte Kanal-ID
0	1	8430_H	Der File-DB ist zu kurz.
0	1	8431_H	Fehler beim Schreiben in den File-DB.
0	1	8432_H	Fehler beim Lesen aus dem File-DB.
0	1	8433_H	Fehler beim Zugriff auf den File-DB.
0	1	8434_H	Aktion wurde abgebrochen.
0	1	8435_H	Kanal wird zurückgesetzt.
0	1	8436_H	Unerwartete Server Antwort
0	1	8437_H	Zertifikat konnte nicht verifiziert werden.
0	1	8438_H	Unbekannter Fehler aufgetreten
0	1	8439_H	Das FTP-Kommando führt zu einem Fehler. Die Ursache ist beim FTP-Server zu suchen (REST-Kommando).
0	1	8440_H	Der FTP-Server unterstützt das geforderte SSL-Protokoll nicht.
0	1	8446_H	Nachdem das FTP-Passwort an den FTP-Server gesendet wurde,wurde ein unerwarteter Code vom FTP-Server zurückgegeben.
0	1	8451_H	Beim Versuch, den Übertragungsmodus von binär auf ASCII umzustellen, wurde ein Fehler gemeldet.
0	1	8455_H	Eine Speicheranforderung im CM/CP ist fehlgeschlagen.
0	1	8460_H	Bei der SSL/TLS-Abwicklung ist ein Problem aufgetreten.
0	1	8469_H	Schnittstellen-Fehler Die angegebene Ausgangs-Schnittstelle konnte nicht genutzt werden. Abhilfe: Stellen Sie ein, welche Schnittstelle für ausgehende Verbindungen genutzt werden soll.
0	1	8475_H	Das SSL-Zertifikat oder der SSH md5 fingerprint wurde als nicht vertrauenswürdig erachtet.
0	1	8476_H	Vom FTP-Server wurde nichts empfangen. Im aktuellen Zustand muss von einem fehlerhaften Verhalten ausgegangen werden.
0	1	8477_H	Die angegebene "Crypto engine" (kryptografisches Modul) wurde nicht gefunden.
0	1	8478_H	Der Vorgang schlug fehl, die gewählte SSL-"Crypto engine" (kryptografisches Modul) als Standard zu setzen.
0	1	8480_H	Es ist ein Problem mit dem Zertifikat des FTP-Client aufgetreten.
0	1	8481_H	Die angegebene Ziffer konnte nicht verwendet werden.
0	1	8482_H	Der FTP-Server verwendet eine Codierung die nicht unterstützt wird.
0	1	8484_H	Die maximale Dateigröße wurde überschritten.

DONE	ERROR	STATUS	Bedeutung
0	1	8485_H	Der File-DB wurde während der Sendebearbeitung verändert oder der File-DB ist nicht korrekt aufgebaut.
0	1	8489_H	Daten konnten nicht gesendet werden. Es ist nicht genügend Speicherkapazität für den Vorgang auf dem FTP-Server verfügbar.
0	1	8492_H	Die Datei existiert bereits. Die Datei wird nicht überschrieben.
0	1	8496_H	Beim Lesen des SSL CA-Zertifikats ist ein Problem aufgetreten.
0	1	8497_H	Ein unerwarteter Fehler ist bei der SSH-Sitzung aufgetreten.
0	1	8498_H	Es war nicht möglich, die SSL-Verbindung abzubauen.
0	1	8499_H	Der Socket ist nicht bereit zum Senden/Empfangen. Warten Sie, bis die Bereitschaft besteht und versuchen Sie es erneut.
0	1	8501_H	Die Überprüfung des SSL-Zertifikates vom FTP-Server ist fehlgeschlagen.
0	1	8507_H	Während der aktiven FTP-Sitzung ist im Rahmen des Verbindungsaufbaus beim Warten auf den FTP-Server ein timeout aufgetreten.
0	1	$8F55_H$	Header-Status-Bit: Locked
0	1	$8F56_H$	Das NEW-Bit im File-DB-Header wurde nicht zurückgesetzt
0	1	$8F6B_H$	Mögliche Ursachen: • Falscher Wert für den Parameter CMD Werte von 0 bis 15 sind zugelassen. • Ein Kommando des FB 40 wird nicht unterstützt. Mögliche Ursache: Falsche Firmware des CP Abhilfe: Firmware-Update (bei älteren CPs statt des FB 40 die Funktionen FC 40...FC 44 benutzen.)
0	1	$8F7F_H$	Interner Fehler; beispielsweise unzulässige ANY-Referenz

Siehe auch

Übersicht FTP_CMD (Seite 4013)

Aufbau und Verwendung des File-DB - FTP-Client-Betrieb

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten als "Array [..] of Byte" oder "Array [..] of Char" (besitzt variable Länge und Struktur)

Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen, um den FTP-Zugriff zu ermöglichen:

- In den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Deaktivieren Sie bei allen als File-DB verwendeten Datenbausteinen das Attribut "Optimierter Bausteinzugriff".

File-DB-Header für FTP-Client-Betrieb

Der hier beschriebene File-DB Header ist identisch zu dem für den Server-Betrieb beschriebenen File-DB-Header.

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten (Datei existiert nicht). • 1: Der File-DB enthält gültige Nutzdaten (Datei existiert).	Das FTP-Kommando "DELETE" setzt EXIST=0. Das FTP-Kommando "STORE" setzt EXIST=1.
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos "STORE" und "RETRIEVE" setzen während der Bearbeitung LOCKED=1, wenn das Bit zuvor auf 0 stand. Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes ebenfalls LOCKED setzen bzw. zurücksetzen. Dadurch wird zur Konsistenzsicherung eine gegenseitige Verriegelung zwischen Anwenderprogramm und FTP-Abwicklung bewirkt. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen (wenn = 0) 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen (wenn = 0) 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen

Parameter	Typ	Wert / Bedeutung	Versorgung
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muss nach dem Lesen der Daten NEW=0 setzen, um ein erneutes Kommando "RETRIEVE" zu ermöglichen.
WRITE_ACCESS	BOOL	• 0: Das Anwenderprogramm hat Schreibrecht für die File-DBs in der S7-CPU. • 1: Das Anwenderprogramm hat kein Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungwert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereichs (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Vorzeichenlose Zahl (16 Bit), die den letzten Reply-Code von FTP als Binärwert enthält. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Wird vom FTP-Protokoll-Abwickler bei der FTP-Kommandobearbeitung des Servers aktualisiert.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung des Files. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wenn die Funktion "Uhrzeitweiterleitung" genutzt wird, dann entspricht der Eintrag der weitergeleiteten Zeit. Wenn die Funktion "Uhrzeitweiterleitung" nicht genutzt wird, dann wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des CP (Initialisierungswert: 01.01.1994 00:00h).

Siehe auch

Datenbaustein FILE_DB_HEADER als Vorlage - FTP-Client-Betrieb (Seite 4029)

Datenbaustein FILE_DB_HEADER als Vorlage - FTP-Client-Betrieb

Bedeutung

Der hier beschriebene File-DB-Header ist in den Betriebsarten für FTP-Client-Betrieb und FTP-Server-Betrieb identisch.

Für das Anlegen eines File-DB-Header ist der Datentyp FILE_DB_HEADER vordefiniert.

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

Gehen Sie wie folgt vor:

1. Legen Sie in der CPU, in der Sie das Anwenderprogramm mit den FTP-Anweisungen erstellen, einen Datenbaustein vom Typ "Global-DB" an.
2. Öffnen Sie den Eigenschaftendialog des neu angelegten Datenbausteins (Kontextmenü), deaktivieren Sie das Attribut "Optimierter Bausteinzugriff" und schließen Sie den Eigenschaftendialog wieder.
3. Selektieren Sie im Bausteineditor des DB die Zeile, die Sie als Startzeile für den File-DB verwenden werden.
4. Wählen Sie in der Spalte "Datentyp" ein Strukturelement vom Typ "FILE_DB_HEADER" oder bzw. geben Sie den Typ "FILE_DB_HEADER" über die Tastatur ein.
   Ergebnis: Es wird eine Datenstruktur mit der für den File-DB benötigte Header-Struktur angelegt.
5. Setzen Sie den Parameter "WRITE_ACCESS" auf "true", um den Zugriff zu ermöglichen.
6. Geben Sie einen Wert für die Länge der Nutzdaten am Parameter "MAX_LENGTH" ein, vgl. nächster Punkt.
7. Legen Sie darunter für die zu übertragenden Nutzdaten ein Datenfeld vom Typ "Array [..] of Byte" oder "Array [..] of Char" an.
   Die Größe des Feldes muss der Angabe von MAX_LENGTH im Header entsprechen.

Hinweis

Funktion "Neuen Baustein hinzufügen" - Typauswahl

Beim Anlegen neuer Datenbausteine wird unter dem Eintrag "Typ" in der Klappliste auch der Bausteintyp "FILE_DB_HEADER" angeboten. Verwenden Sie diese Auswahlmöglichkeit nicht! Der so erzeugte Datenbaustein enthält lediglich die Header-Struktur und kann nicht um den benötigten Bereich zur Ablage von Nutzdaten erweitert werden.

Datenbaustein FILE_DB_HEADER - Beispiel und Vorlage für den File-DB-Header

In der Deklarationsansicht erkennen Sie folgende Struktur:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	bit08	BOOL	FALSE	reserviert
+0.1	bit09	BOOL	FALSE	reserviert
+0.2	bit10	BOOL	FALSE	reserviert
+0.3	bit11	BOOL	FALSE	reserviert
+0.4	bit12	BOOL	FALSE	reserviert
+0.5	bit13	BOOL	FALSE	reserviert
+0.6	bit14	BOOL	FALSE	reserviert
+0.7	bit15	BOOL	FALSE	reserviert
+1.0	EXIST	BOOL	FALSE	wenn TRUE: File-DB enthält gültige Daten
+1.1	LOCKED	BOOL	FALSE	wenn TRUE: Der File-DB ist blockiert aufgrund einer inhaltlichen Änderung.
+1.2	NEW	BOOL	FALSE	wenn TRUE: Der Inhalt des File-DB ist neu und darf nicht überschrieben werden.
+1.3	WRITE_ACCESS	BOOL	FALSE	wenn TRUE: Der FTP-Server hat Schreib-Zugriff.
+1.4	bit04	BOOL	FALSE	reserviert
+1.5	bit05	BOOL	FALSE	reserviert
+1.6	bit06	BOOL	FALSE	reserviert
+1.7	bit07	BOOL	FALSE	reserviert
+2.0	ACT_LENGTH	DINT	L#0	Aktuelle Länge des Inhalts in Byte (nicht eingeschlossen 20 Byte für den Header)
+6.0	MAX_LENGTH	DINT	L#0	Maximale Länge des Inhalts in Byte (nicht eingeschlossen 20 Byte für den Header)
+10.0	FTP_REPLY_CODE	INT	0	Wird vom FTP-Protokoll-Abwickler bei der FTP-Kommandobearbeitung aktualisiert, letzte Antwortinformation vom FTP-Server.
+12.0	DATE_TIME	DATE_AND_TIME	DT#00-1-1-0:0:0.000	Datum und Zeitangabe für die letzte Änderung am Inhalt des File-DB.
=20.0		END_STRUCT		

S7-1200-CM/CP

Telecontrol

Telecontrol-Anweisungen

TC_CON: Verbindung über das GSM-Netz aufbauen

Bedeutung

Die Anweisung TC_CON ermöglicht einer S7-1200 mit Telecontrol-CP (bspw. CP 1242-7 - 6GK7242-7KX30-0XE0) den Aufbau einer Verbindungen der folgenden Typen:

- ISO-ON-TCP
  Verbindungspartner ist ein CP in einer S7-Station.
  ISO-ON-TCP-Verbindungen werden beim CP 1242-7 nur in der Betriebsart "GPRS direkt" verwendet.
- UDP
  Der Verbindungspartner ist beliebig.
- SMS
  Verbindungspartner ist ein SMS-Client.
- Telecontrol-Verbindung (Protokoll: TeleControl Basic)
  Verbindungspartner ist entweder der Telecontrol-Server oder eine andere Station, die über den Telecontrol-Server erreichbar ist.

Ein TC_CON baut genau eine Verbindung auf. Abhängig von der Betriebsart des CP 1242-7 und dem verwendeten Protokoll werden pro CP maximal 3 bis 5 parallele Verbindungen mit eindeutigen IDs (siehe unten) unterstützt. Die maximal mögliche Anzahl paralleler Verbindungen entnehmen Sie den Leistungsdaten des CP.

Der Parameter CONNECT verwendet zur Verbindungsbeschreibung einen Datenbaustein (DB) mit der Struktur eines Systemdatentyps (SDT).

Der gewünschte Verbindungstyp wird über einen jeweils verbindungsspezifischen SDT "TCON_..." (siehe unten) definiert. Für jeden der oben genannten Verbindungstypen muss einer der folgenden SDTs parametriert werden:

- TCON_IP_RFC für ISO-ON-TCP-Verbindungen
- TCON_IP_V4 für UDP-Verbindungen
- TCON_PHONE für SMS-Verbindungen
- TCON_WDC für Telecontrol-Verbindungen

Im Parameter "ActiveEstablished" dieser SDTs wird unter anderem auch festgelegt, ob es sich um einen aktiven oder passiven Verbindungaufbau handelt.

Zur Parametrierung dieser SDTs siehe TCON_...: SDTs für den Telecontrol-Verbindungsaufbau (Seite 4048).

Der Parameter ID referenziert die Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.

Der Parameter INTERFACE referenziert die Schnittstelle des gewünschten lokalen CP. Diese muss aus STEP 7 übernommen werden.

Aufrufschnittstelle in FUP-Darstellung

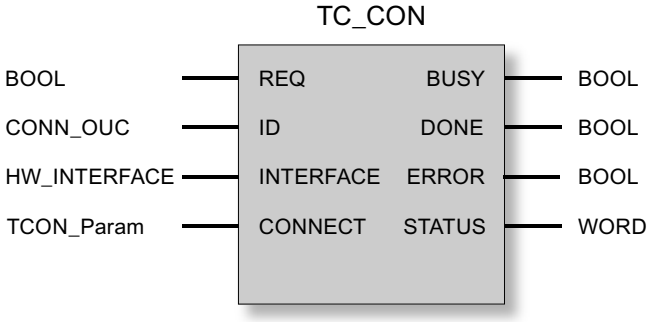

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_CON.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	$1...07FF_h$	Referenz auf die jeweilige Verbindung. Die ID wird vergeben. Der Wert von ID wird auch vom jeweiligen Systemdatentyp (SDT) des Parameters CONNECT benötigt.
INTERFACE	INPUT	HW_INTERFACE		Referenz auf die Schnittstelle des CP (HW-Kennung) *
CONNECT	INOUT	TCON_Param	Siehe auch "TCON_...: SDTs für den Telecontrol-Verbindungsaufbau"	Referenz auf einen Datenbaustein für den Verbindungsaufbau. Die SDTs vom Typ TCON_IP_RFC, TCON_IP_V4, TCON_PHONE oder TCON_WDC geben die Struktur des passenden Datenbausteins für die jeweilige Verbindung vor. Beachten Sie bei den SDTs den Parameter "ActiveEstablished" (aktiver / passiver Verbindungsaufbau).
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

***** Abhängig vom CP-Typ ist die "HW-Kennung" der Telecontrol-Schnittstelle oder der Ethernet-Schnittstelle zu referenzieren. Nach Selektion des CP in der Netz- oder Gerätesicht von STEP 7 finden Sie die "HW-Kennung" an folgender Stelle im Inspektorfenster:

- Bis STEP 7 V14 SP1
  Register "Eigenschaften" > "Allgemein" > "Betreffende Schnittstelle" > Parametergruppe "HW-Kennung"
- Ab STEP 7 V15
  Register "Eigenschaften" > "Systemkonstanten" > "HW-Kennung" (Systemkonstante vom Datentyp "Hw_Interface")
  Abhängig vom CP-Typ ist folgende Systemkonstante relevant:
  - CP 1242-7 (6GK7242-7KX30-0XE0)
    "Local~<CP-Name>~<Name der Telecontrol-Schnittstelle>"
  - Alle anderen Telecontrol-CPs
    "Local~<CP-Name>~<Name der Ethernet-Schnittstelle>"

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1. In folgenden Fällen kann der Zustand BUSY = 1 länger andauern:

- Bei aktiven ISO-ON-TCP-Verbindungen, wenn der Partner nicht erreichbar ist.
- Bei passiven Verbindungen, wenn kein Telegramm empfangen wird.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	8087_H	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich
0	1	$80E3_H$	Die ID wird bereits für eine andere Verbindung verwendet. Das heißt, bei TC_SEND, TC_RECV oder TC_DISCON ist BUSY derzeit TRUE. Die Statusanzeige wird auch ausgegeben, wenn EN_R bei TC_RECV dauerhaft TRUE ist. Dies bewirkt, dass TC_RECV größtenteils aufgerufen ist. Abhilfe für diesen Fall: Schalten Sie EN_R aus, bevor TC_CON oder TC_DISCON aufgerufen wird. EN_R darf erst wieder eingeschaltet werden, wenn TC_CON fehlerfrei ausgeführt wurde.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80E8_H$	Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter. In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.
0	1	$80EB_H$	Anfrage vorübergehend zurückgewiesen (TC_CON wurde bereits mit der selben Zieladresse aufgerufen.)
0	1	$80EC_H$	Öffnen des Listener Port konnte nicht durchgeführt werden: Prüfen Sie die Verbindungsparameter.
0	1	$80F2_H$	Der CP befindet sich in der falschen Betriebsart: • Telecontrol-Verbindungen sind nur in der Betriebsart "Telecontrol" erlaubt. • ISO-ON-TCP-Verbindungen sind nur in der Betriebsart "GPRS direkt" erlaubt.
0	1	$80F3_H$	Kein freier Verbindungsendpunkt zum Senden von Daten: • Nutzen Sie weniger Verbindungen oder • Nutzen Sie weniger passive Verbindungen oder • Schalten Sie NTP aus. Beachten Sie die maximale Anzahl paralleler Verbindungen des CP 1242-7.
0	1	$80F4_H$	Verbindungsendpunkt kann nicht erzeugt werden: Wiederholen Sie den Aufruf. Prüfen Sie ggf. die Verbindungsparameter.
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: Verbindungsaufbau durch TC_CON fehlgeschlagen. Wiederholen Sie den Bausteinaufruf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format, ungültiger Wert oder Rufnummer im TCON_Phone größer als 20 Zeichen) Prüfen Sie die Projektierung des SDT "TC_CON...".

TC_DISCON: Verbindung über das GSM-Netz abbauen

Bedeutung

Die Anweisung TC_DISCON bewirkt bei einer S7-1200 mit CP 1242-7 den Abbau einer ISO-ON-TCP-, UDP, SMS- oder Telecontrol-Verbindung, die mit der Anweisung TC_CON aufgebaut wurde.

Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

TC_DISCON baut die Verbindung zum Telecontrol-Server nur logisch ab. Auf TCP/IP-Ebene bleibt die Verbindung erhalten.

Wenn die Verbindung zum Telecontrol-Server physikalisch abgebaut werden soll, dann projektieren Sie die Verbindung in STEP 7 unter der Parametergruppe "Telecontrol-Server" als "Temporäre Verbindung". Temporäre Stationen bauen die Verbindung nach dem Senden der Daten automatisch ab.

Hinweis

Abbruch der Bearbeitung weiterer Programmbausteine durch TC_DISCON

Ein Aufruf von TC_DISCON beendet die Bearbeitung von TC_CON-, TC_SEND- und TC_RECV-Bausteinen, die mit derselben Verbindungs-ID (Parameter "ID") und Schnittstelle (Parameter "INTERFACE") aufgerufen wurden. Diese Bausteine melden anschließend ERROR.

Kein Aufruf von TC_DISCON, wenn TC_CON "Error = 1" anzeigt.

Wenn TC_CON "ERROR" anzeigt, dann ist die Verbindung nicht aufgebaut. TC_DISCON darf in diesem Fall nicht aufgerufen werden.

Wenn TC_DISCON in diesem Fall doch aufgerufen wird, dann wird die Verbindungs-ID ("ID") kurrzeitig noch reserviert und ein unmittelbar daraufhin aufgerufener TC_CON würde ERROR und STATUS 80E3 anzeigen.

Der Parameter ID referenziert die GPRS-Verbindung. Die ID muss innerhalb der CPU eindeutig sein und gleich derjenigen ID sein, die bei TC_CON verwendet wird.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Aufrufschnittstelle in FUP-Darstellung

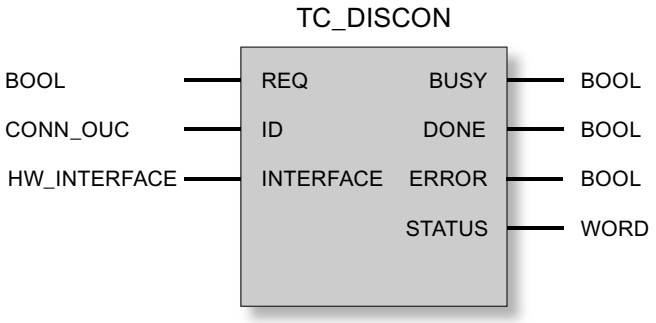

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_DISCON

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	1...07FF$_h$	Referenz auf die jeweilige Verbindung
INTERFACE	INPUT	HW_INTERFACE		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Die Anweisung wurde noch nicht aufgerufen.

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Hinweis

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E4_H$	Unbekannte ID: Keine Verbindung mit dieser ID ist durch TC_CON aufgebaut.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen oder • Verbindung vom entfernten Partner abgebaut.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie die Projektierung des SDT "TC_CON...".

TC_SEND: Daten über das GSM-Netz senden

Bedeutung

Die Anweisung TC_SEND ermöglicht das Senden von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- UDP-Verbindungen
- SMS-Verbindungen
  Das Versenden von SMS wird nur unterstützt, wenn dies in der STEP 7-Projektierung des CP eingerichtet wurde.
- Telecontrol-Verbindungen

Hinweis

SMS an mehrere Empfänger senden

Wenn Sie eine identische SMS an mehrere Empfänger senden wollen, dann müssen Sie für jeden Empfänger eine Verbindung aufbauen.

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC_CON verwendeten Wert für ID entsprechen.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Die Anzahl zu sendender Daten wird mit dem Parameter LEN angegeben.

Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projektierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL.

Die Zieladresse (Verbindungspartner) für die zu sendenden Daten wird in der Anweisung TC_CON projektiert.

Aufrufschnittstelle in FUP-Darstellung

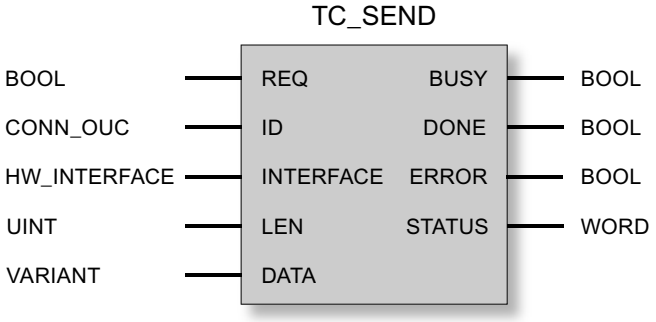

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_SEND.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	1...07FF$_h$	Referenz auf die jeweilige Verbindung
INTERFACE	INPUT	HW_INTERFACE		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
LEN	INPUT	UINT	1...2048	Anzahl der Bytes der zu sendenden Daten. Der Wert muss ≥ 1 und ≤ 2048 sein. Der Wert sollte der Größe des Bereichs von DATA entsprechen.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
DATA	INOUT	VARIANT		Adressverweis auf den zu übertragenden Datenbereich der CPU *
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. ** Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

** Nach dem Senden eines Telegramms setzt TC_SEND DONE = 1. Beachten Sie folgendes Verhalten:
Der Ausfall einer ISO-on-TCP-Verbindungen wird beim Sender erst nach 1 bis 2 Minuten festgestellt. Die übertragenen Daten können verloren sein, obwohl TC_SEND beim Sender DONE = 1 gesetzt hat.
Wenn eine ISO-on-TCP-Verbindung nach dem Empfang eines Telegramms abgebrochen wird, bevor TC_RECV gestartet wurde, dann können die übertragenen Daten verloren sein, auch wenn TC_SEND beim Sender DONE = 1 gesetzt hat.

Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung sendet diejenigen Daten als SMS-Text, auf welche der Zeiger vom Typ VARIANT des Parameters DATA referenziert.

Wenn von DATA auf einen Operanden des Datentyps STRING für SMS-Texte referenziert wird, dann werden die ersten zwei Bytes mit Längenangaben des Strings mit übertragen.

Eine Möglichkeit für die korrekte Textdarstellung von zu sendenden SMS ist die Umwandlung des Text-Strings in ein Array of BYTE oder Array of CHAR mithilfe der Konvertierungsfunktion Strg_TO_Chars. Strg_TO_Chars wird am Parameter EN mit dem Ausgangsparameter ENO von TC_SEND verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS *	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E0_H$	Interner Fehler Wenn Sie Telegramme direkt an den Telecontrol-Server schicken (Betriebsart "Telecontrol"), dann stellen Sie sicher, dass die Sendezykluszeit ≥ 1 Sekunde ist.
0	1	$80E1_H$	Zeitüberschreitung: • Erhöhen Sie in der Projektierung des CP 1242-7 den Wert der "Verbindungsüberwachungszeit" oder • Überprüfen Sie den Verbindungspartner.
0	1	$80E4_H$	Unbekannte ID: Rufen Sie zuerst TC_CON auf.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80E7_H$	Zu sendende Daten nicht vollständig übertragen: Wiederholen Sie den Auftrag.
0	1	$80E8_H$	Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter. In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.
0	1	$80E9_H$	Verbindungsabbau durch entfernten Partner: Überprüfen Sie den Verbindungspartner. Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.
0	1	$80EA_H$	Fehlermeldung vom entfernten Partner: • Überprüfen Sie den Verbindungspartner. Aktivieren Sie beim Kommunikationspartner die Anweisung "TC_RECV". • Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.

DONE	ERROR	STATUS *	Bedeutung
0	1	$80EF_H$	Senden der SMS konnte nicht durchgeführt werden: • Überprüfen Sie, ob die Zieladresse (Rufnummer des Zielteilnehmers) existiert. • Überprüfen Sie, ob die gesteckte SIM-Karte das Senden von SMS unterstützt. • Überprüfen Sie die Länge des versendeten SMS-Texts. SMS-Texte > 160 Zeichen werden nicht gesendet. • Stellen Sie sicher, dass beim Anlegen des Datenbausteins TCON_PHONE die Option "Standard" für den Bausteinzugriff gewählt wurde.
0	1	$80F1_H$	Das Senden von SMS ist in der STEP 7-Projektierung des CP nicht aktiviert: Aktivieren Sie in der Projektierung des CP die Option "SMS ermöglichen".
0	1	$80F4_H$	Verbindungsendpunkt kann nicht erzeugt werden: Überprüfen Sie den Verbindungspartner.
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen. oder • Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert): Prüfen Sie die Projektierung des SDT "TC_CON...".

* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

TC_RECV: Daten über das GSM-Netz empfangen

Bedeutung

Die Anweisung TC_RECV ermöglicht das Empfangen von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- SMS-Verbindungen
  Zum Empfangen von SMS muss die Rufnummer des Absenders in der STEP 7-Projektierung des empfangenden CP projektiert sein (Autorisierte Rufnummern). Der Absender muss die CLIP-Funktion unterstützen.
  Die Rufnummer des Verbindungspartners muss im SDT "TCON_PHONE" eingetragen sein. Weck-SMS werden ausgefiltert.
- Telecontrol-Verbindungen

Hinweis

SMS von verschiedenen Sendern empfangen

Wenn Sie SMS von verschiedenen Sendern empfangen wollen, dann stehen Ihnen hierfür zwei Alternativen zur Wahl:

- Sie müssen mehrere Verbindungen (TC_CON, TC_RECV, TC_DISCON) projektieren.
  oder
- Sie dürfen bei nur einer einzigen projektierten Verbindung in dem erforderlichen Datenbaustein "TCON_PHONE" im Parameter "PhoneNumber" keine Rufnummer eingeben. Dies wird beim Empfangen von Nachrichten als Platzhalter für alle autorisierten Verbindungspartner interpretiert.

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC_CON verwendeten Wert für ID entsprechen.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Die maximale Anzahl der Empfangsdaten wird mit dem Parameter LEN angegeben.

Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projektierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL. Die empfangenen Daten werden so interpretiert, als hätte der sendende Partner die gleichen Datentypen verwendet.

Der für die Verbindungsbeschreibung des TC_RECV verwendete DB (Systemdatentyp) muss sich von einem für TC_SEND verwendeten DB unterscheiden.

Speichern von SMS

Empfangene SMS werden im CP 1242-7 (25 Speicherplätze) und auf der SIM-Karte (variable Anzahl an Speicherplätze) remanent gespeichert.

- Nach dem Lesen der SMS durch TC_RECV wird die SMS aus ihrem Speicherplatz gelöscht.
- Wenn alle Speicherplätze belegt sind und eine neue SMS empfangen wird, wird die älteste SMS gelöscht.

Aufrufschnittstelle in FUP-Darstellung

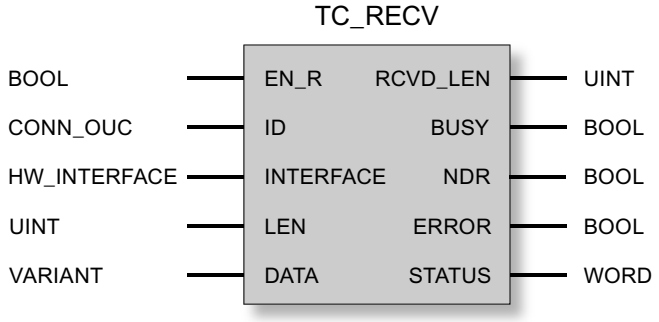

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_RECV

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
EN_R	INPUT	BOOL	0: Datenempfang gesperrt 1: Datenempfang ermöglicht	Ermöglichen / Sperren des Empfangs von Daten. • Bausteinversion 1.1: Nach dem Setzen von 1 auf 0 ist der Baustein inaktiv. • Bausteinversion 1.0: Nach dem Setzen von 1 auf 0 empfängt der Programmbaustein noch einmal Daten (bis DONE = 0 und ERROR = 0). Beachten Sie die Hinweise zur Statusanzeige 80E3 beim TC_CON.
ID	INPUT	CONN_OUC	1...07FF$_h$	Referenz auf die jeweilige Verbindung
INTERFACE	INPUT	HW_INTER-FACE		Referenz auf die Schnittstelle des CP 1242-7 Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
LEN	INPUT	UINT	1...2048	(Mindest-) Azahl der Bytes der zu empfangenden Daten, maximal 2048
DATA	INOUT	VARIANT		Adressverweis auf Empfangs-Datenbereich der CPU *

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
RCVD_LEN	OUTPUT	UINT		Byte-Anzahl der empfangenen Daten
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung referenziert den empfangenen SMS-Text mit dem Zeiger vom Typ VARIANT des Parameters DATA auf den Datenbereich der CPU.

Wenn von DATA auf einen Operanden des Datentyps STRING für den SMS-Text referenziert wird, dann werden die ersten zwei Bytes des SMS-Texts als Längenangaben des Datentyps STRING und nicht als SMS-Text interpretiert.

Eine Möglichkeit für die korrekte Textdarstellung von zu empfangenden SMS ist die Umwandlung eines Array of BYTE oder Array of CHAR in einen Text-String mithilfe der Konvertierungsfunktion Chars_TO_Strg. Chars_TO_Strg wird am Parameter EN mit dem Ausgangsparameter ENO von TC_RECV verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS *	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	$80A3_H$	• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen. • Es wird versucht, eine nicht vorhandene Verbindung zu beenden.
0	1	$80E0_H$	Interner Fehler
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E4_H$	Unbekannte ID: Rufen Sie zuerst TC_CON auf.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen. oder • Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie die Projektierung des SDT "TC_CON...".

* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

TC_CONFIG: Projektierungsdaten auf CP übertragen

Bedeutung

Mit der Anweisung TC_CONFIG können die in STEP 7 projektierten Parameter eines S7-1200-Mobilfunk-CP geändert werden. Die projektierten Werte werden nicht remanent überschrieben. Die überschriebenen Werte bleiben gültig bis zum erneuten Aufruf von TC_CONFIG oder bis zum nächsten Anlauf der Station (Kaltstart durch Spannung AUS → EIN).

Um die Funktion nutzen zu können, müssen in der STEP 7-Basisprojektierung des CP bereits projektierte Werte vorhanden sein.

Wenn die STEP 7-Projektierungsdaten des CP dauerhaft geändert werden sollen, dann muss die Anweisung nach jedem Anlauf der Station (Kaltstart) neu aufgerufen werden oder ein geändertes Projekt muss in die Station geladen werden.

Der Parameter CONFIG zeigt auf den Speicherbereich mit den Projektierungsdaten. Die Projektierungsdaten werden in einem Datenbaustein (DB) gespeichert. Die Struktur des DB wird durch den Systemdatentyp (SDT) IF_CONF vorgegeben.

Diejenigen Projektierungsdaten, die im CP geändert werden sollen, werden im IF_CONF als Blöcke "IF_CONF_..." für die einzelnen Parameter nach Bedarf zusammengestellt.

Parameter, die sich durch die Anweisung nicht ändern sollten, werden im IF_CONF nicht eingetragen. Sie behalten dann den in STEP 7 projektierten Wert.

Details zur Parametrierung von IF_CONF enthält der Abschnitt IF_CONF: SDT für Telecontrol-Projektierungsdaten (Seite 4053).

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP.

Aufrufschnittstelle in FUP-Darstellung

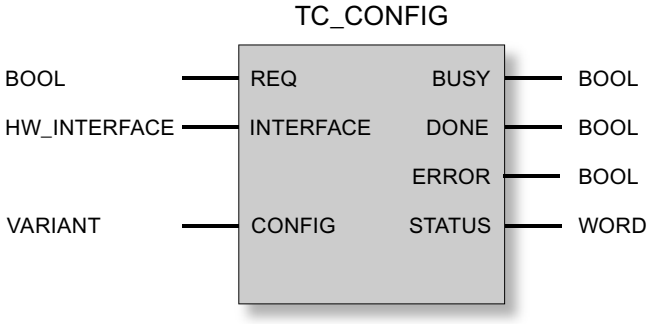

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_CONFIG

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
INTERFACE	INPUT	HW_INTERFACE (WORD)		Referenz auf die Schnittstelle des CP 1242-7 Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
CONFIG	INOUT	VARIANT	Siehe auch "IF_CONF: SDT für Telecontrol-Projektierungsdaten"	Referenz auf den Speicherbereich mit der Zusammenstellung der zu ändernden Projektierungsdaten
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	$80E0_H$	Interner Fehler (bspw. kein Wert in der Basisprojektierung eines Parameters vorhanden)
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80EB_H$	Anfrage vorübergehend zurückgewiesen (der CP wird momentan von STEP 7 konfiguriert.)

DONE	ERROR	STATUS	Bedeutung
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie den SDT "IF_CONF".
0	1	$80F7_H$	Falsche ID in den Parameterblöcken der Projektierungsdaten: Prüfen Sie den SDT "IF_CONF".

TCON_...: SDTs für den Telecontrol-Verbindungsaufbau

Systemdatentypen TCON_... für Telecontrol-Verbindungen

Die vier nachfolgend beschriebenen Systemdatentypen TCON_... werden unter anderem für die Verbindungsbeschreibung von Telecontrol-Verbindungen eines Telecontrol-CP verwendet.

TCON_PHONE wird auch bei Mobilfunk-CPs der S7-1200 für die Beschreibung einer Verbindung der Open User Communication zur Übertragung von SMS verwendet (TCON / TDISCON / TSEND oder TSEND_C).

Um für den Telecontrol-CP den Aufbau einer Telecontrol-Verbindung mithilfe der Anweisung TC_CON zu projektieren, wird der Parameter CONNECT der Anweisung für die Verbindungsbeschreibung verwendet.

Die Verbindungsbeschreibung ist durch die Struktur eines Systemdatentyps (SDT) vorgegeben. Die Struktur des jeweiligen SDT enthält die notwendigen Parameter, die zum Aufbau der Verbindung mit dem entfernten Kommunikationspartner erforderlich sind.

Für unterschiedliche Verbindungstypen, die vom jeweiligen entfernten Kommunikationspartner abhängen, werden folgende SDTs verwendet:

- TCON_IP_RFC für ISO-on-TCP-Verbindungen zu IPv4-Stationen mit Telecontrol-CP
- TCON_IP_V4 für UDP-Verbindungen zu IPv4-Stationen (nur Senden)
- TCON_PHONE für Verbindungen zu SMS-Clients
- TCON_WDC für Verbindungen zu Telecontrol-Servern oder Stationen, die über den Telecontrol-Server erreichbar sind.

Die Parametrierung der Verbindungsbeschreibung wird in einem Datenbaustein des gleichen Typs wie der SDT vorgenommen.

Anlegen eines DB vom Typ TCON_...

Die Datentypen der jeweiligen DBs müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen eines TCON_...-DB folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Standard-Bausteinzugriff (nicht optimiert) an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB einen SDT an, indem Sie den Namen vergeben und in die Zelle des Datentyps den gewünschten Typ (bspw. "TCON_IP_RFC") eintippen.
   Der SDT mit seinen Parametern (siehe unten) wird angelegt.
3. Projektieren Sie die Parameter, die nachfolgend für jeden SDT-Typ beschrieben sind.

Reservierte Bits werden nicht angezeigt.

Systemdatentyp TCON_IP_RFC für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur bei ISO-on-TCP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt. Der CP muss für die Betriebsart "GPRS direkt" projektiert sein.

Tabelle 4-104 Parameter von TCON_IP_RFC

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0C	Protokoll-Variante 12 (C$_h$): ISO-on-TCP-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau • 1: Aktiver Verbindungsaufbau
6 ... 7	-	-	-	- reserviert -
8 ... 11	RemoteAddress	IP_V4		IP-Adresse der Verbindungspartner
	ADDR	Array [1...4] of Byte		IP-Adresse des jeweiligen Verbindungspartners
12 ... 13	RemoteTSelector	TSelector		Entfernter T-Selektor
	TSelLen	UINT		Länge des entfernten T-Selektors "RemoteTSelector"
14 ... 45	TSel	Array [1...32] of Byte	beliebig	Entfernter Transport-Selektor der Verbindung • Bei "ActiveEstablished" = 1: Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner). • Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
46 ... 47	LocalTSelector	TSelector		Lokaler T-Selektor
	TSelLen	UINT		Länge des lokalen T-Selektors "LOCAL_TSel"
48 ... 79	TSel	Array [1...32] of Byte	beliebig	Lokaler Transport-Selektor der Verbindung • Bei "ActiveEstablished" = 1: Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner). • Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)

Systemdatentyp TCON_IP_V4 für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur zum Senden über UDP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt.

Wenn empfangen werden soll, dann muss ActiveEstablished = 0 gesetzt werden.

Tabelle 4-105 Parameter von TCON_IP_V4

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0B	Protokoll-Variante 11 (B$_h$): UDP-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau Einstellung zum Senden und Empfangen von Daten. • 1: Aktiver Verbindungsaufbau Einstellung nur zum Senden von Daten.
6 ... 9	RemoteAddress	IP_V4		IP-Adresse des Verbindungspartners
	ADDR	Array [1...4] of Byte		Die vier Bytes (ADDR[1] ... ADDR[4]) spezifizieren die vier Blöcke der IP-Adresse.
10 ... 11	RemotePort	UINT	1...65535	IP-Port des Verbindungspartners Nicht relevant, wenn ActiveEstablished = 0.
12 ... 13	LocalPort	UINT	1...65535	Lokaler IP-Port ("0" ist nicht erlaubt.) Nicht relevant, wenn ActiveEstablished = 1.

Systemdatentyp TCON_PHONE für SMS-Verbindungen

Hinweis

Autorisierte Rufnummern

Voraussetzung dafür, dass der CP eine SMS akzeptiert, ist die Autorisierung des sendenden Kommunikationspartners mithilfe seiner Rufnummer. Diese Rufnummern werden in STEP 7 beim CP in der Liste "Autorisierte Rufnummern" projektiert.

SMS-Text

- Auf programmierte SMS-Texte für zu sendende SMS wird über den Parameter DATA der Anweisung TC_SEND zugegriffen.
- Der Text einer empfangenen SMS wird dem Adressbereich der CPU über den Parameter DATA der Anweisung TC_RECV zugewiesen.

Tabelle 4-106 Parameter von TCON_PHONE

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale Mobilfunk-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0E	Protokoll-Variante 14 (E$_h$): SMS-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus (nicht relevant für den CP 1242-7): • 0: Passiver Verbindungsaufbau (hier nicht relevant) • 1: Aktiver Verbindungsaufbau
6...7	-	-	-	- reserviert -
8 ... 31	PhoneNumber	STRING[22]		Rufnummer des Verbindungspartners Erlaubte Werte: Plus-Zeichen (+) und Ziffern Achten Sie auf die genaue Zeichenfolge der vom Netzbetreiber zugewiesenen Ländervorwahl der betreffenden Rufnummer ("+"-Zeichen oder Nullen). Ohne Eintrag des Parameters PhoneNumber wird kein Verbindungspartner spezifiziert und SMS können von allen autorisierten Verbindungspartnern empfangen werden. Beachten Sie beim Anlauf: Ohne Eintrag liefert TC_RECV zunächst die älteste empfangene SMS.

Systemdatentyp TCON_WDC für Verbindungen zu Telecontrol-Servern oder entfernten Stationen

Sie können die Verbindung mit dem Telecontrol-Server, welcher der S7-1200 zugeordnet ist, oder zu einer entfernten Station, die über den Telecontrol-Server erreichbar ist, mit dem TCON_WDC projektieren. Die Adressdaten des Telecontrol-Servers, der dem CP zugeordnet ist, finden Sie in STEP 7 im Register "Telecontrol-Schnittstelle > Betriebsart" des CP. Der Telecontrol-Server oder eine entfernte Station wird über den Host-Namen oder die IP-Adresse adressiert.

Im Parameter "RemoteWdcAddress" des TCON_WDC wird die Zugangs-ID des Verbindungspartners angegeben.

Tabelle 4-107 Parameter von TCON_WDC

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des CP Zur "HW-Kennung" siehe Kapitel TC_CON: Verbindung über das GSM-Netz aufbauen (Seite 4031).
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0F	Protokoll-Variante 15 (F$_h$): Telecontrol-Verbindung über IP-Adressse
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau • 1: Aktiver Verbindungsaufbau
6 ... 7	-	-	-	- reserviert -
8 ... 11	RemoteWdcAddress	DWORD		Angabe der Zugangs-ID (hex). Die Zugangs-ID ist abhängig vom Verbindungspartner. • Verbindung zu einem entfernten CP: Die Zugangs-ID wird gebildet aus: – STEP 7-Projektnummer – Stationsnummer – Steckplatz Wenn die entfernte Station mehrere GPRS-CPs hat und der Weg nicht festgelegt werden soll, dann muss das letzte Byte für den Steckplatz gleich 0 gesetzt werden. Die Zugangs-ID finden Sie im STEP 7-Projekt unter der Parametergruppe "CP-Authentifizierung" des CP. • Verbindung zum Telecontrol-Server: Zugangs-ID = 0 • Um nur in das Prozessabbild des CP zu schreiben: Zugangs-ID = DW#16#FEEDDADA

IF_CONF: SDT für Telecontrol-Projektierungsdaten

Aufbau des Systemdatentyps IF_CONF für die Anweisung TC_CONFIG

Der Parameter CONFIG der Anweisung TC_CONFIG referenziert den Speicherbereich mit den zu ändernden Projektierungsdaten des CP. Die in einem Datenbaustein abgelegten Projektierungsdaten werden als Struktur vom Systemdatentyp (SDT) IF_CONF beschrieben.

Um die Funktion nutzen zu können, müssen in der STEP 7-Basisprojektierung des CP bereits projektierte Werte vorhanden sein.

IF_CONF setzt sich aus einem Header und nachfolgenden Blöcken zusammen, die den Parametern oder Parameterbereichen des CP in den Geräteeigenschaften des STEP 7-Projekts entsprechen.

Die zu ändernden Projektierungsdaten des CP werden als IF_CONF-Blöcke zusammengestellt. Nicht zu ändernde Parameter werden in der IF_CONF-Struktur nicht berücksichtigt und bleiben so, wie sie im STEP 7-Projekt konfiguriert wurden.

Anlegen des DB und der IF_CONF-Strukturen

Die Parameter des CP können Sie innerhalb des IF_CONF-DB in einer oder in mehreren Strukturen mit jeweils einem oder mehreren Blöcken anlegen.

Die Datentypen der jeweiligen Blöcke müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen von IF_CONF folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB eine Struktur an (Datentyp "Struct").
   Den Name können Sie frei festlegen.
3. Fügen Sie unter dieser Struktur einen Header ein, indem Sie den Namen des Headers vergeben und in die Zelle des Datentyps "IF_CONF_Header" eintippen.
   Der Header der Struktur mit seinen drei Parametern (siehe unten) wird angelegt.
4. Legen Sie einen Block für den ersten zu ändernden Parameter an, indem Sie den gewünschten Datentyp (bspw. "IF_CONF_APN") in die Zelle des Datentyps eintippen.
5. Wiederholen Sie den letzten Schritt für all diejenigen Parameter, die Sie mithilfe der Anweisung TC_CONFIG im CP ändern wollen.
6. Aktualisieren Sie abschließend im Header die Blockanzahl im Parameter "subfieldCnt".

Header von IF_CONF

Tabelle 4-108 IF_CONF_Header

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	fieldType	UINT		Blocktyp: Muss immer 0 sein.
2 ... 3	fieldId	UINT		Block-ID: Muss immer 0 sein.
4 ... 5	subfieldCnt	UINT		Gesamtanzahl der in der Struktur enthaltenen Blöcke

Allgemeine Parameter der Parameterblöcke

Jeder Block enthält folgende allgemeine Parameter:

- Id
  Dieser Parameter kennzeichnet den jeweiligen Block und darf nicht verändert werden.
- Length
  Dieser Parameter gibt die Länge des Blocks an. Der Wert dient nur Informationszwecken. Blöcke mit Strings und / oder Arrays haben eine variable Länge. Durch versteckte Bytes kann die tatsächliche Länge von Blöcken größer als die Summe der angezeigten Parameter sein.
- Mode
  Für diesen Parameter sind die folgenden Werte zulässig:

Tabelle 4-109 Werte von "Mode"

Wert	Bedeutung
1	Permanente Gültigkeit der Projektierungsdaten Nicht relevant beim CP
2	Temporäre Gültigkeit der Projektierungsdaten einschließlich Löschen vorhandener permanenter Projektierungsdaten Die permanenten Projektierungsdaten werden durch die Parameterblöcke von IF_CONF ersetzt.

Block für den Parameterbereich "GPRS-Zugang"

Tabelle 4-110 IF_CONF_APN

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	4	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 174
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
AccesspointGPRS	STRING [98]		APN: Name des Zugangspunkts des GSM-Netzwerkbetreibers zum Internet
AccesspointUser	STRING [42]		APN-Benutzername
AccesspointPassword	STRING [22]		APN-Passwort

Block für den Parameterbereich "CP-Identifikation"

Tabelle 4-111 IF_CONF_Login

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	5	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 54
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
ModemName	STRING [22]		Zugangs-ID Der Wert ist nicht parametrierbar.
ModemPassword	STRING [22]		Telecontrol-Passwort (max. 20 Zeichen) Das Passwort ist beim CP 1242-7 (6GK7 242-7KX30-0XE0) nicht über den SDT änderbar.

Block für den Parameterbereich "Telecontrol-Server-Zugang"

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit einem über DNS auflösbaren Namen adressiert wird. Wenn der Telecontrol-Server mit seiner IP-Addresse adressiert wird, dann wird der Block "IF_CONF_TCS_IP_V4" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-112 IF_CONF_TCS_Name

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	6	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 266
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
TcsName	-	-	- reserviert -
	STRING [254]		Durch DNS auflösbarer Name des Telecontrol-Servers
RemotePort	UINT		Port des Telecontrol-Servers
Rank	UINT		Priorität des Servers [1, 2] 1 = Erster Telecontrol-Server, 2 = Zweiter Telecontrol-Server

Block für den Parameterbereich "Telecontrol-Server-Zugang"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit seiner IP-Addresse adressiert wird. Wenn der Telecontrol-Server mit seinem DNS-Namen adressiert wird, dann wird der Block "IF_CONF_TCS_Name" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-113 IF_CONF_TCS_IP_v4

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	7	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
RemoteAddress	IP_V4		IP-Adresse des Telecontrol-Servers
RemotePort	UINT		Port des Telecontrol-Servers
Rank	UINT		Priorität des Servers [1, 2] 1 = Erster Telecontrol-Server, 2 = Zweiter Telecontrol-Server

Block für den Parameterbereich "Betriebsart"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Betriebsart" und "Modem-Einstellungen".

Tabelle 4-114 IF_CONF_GPRS_Mode

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	8	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 10
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
GPRSmode	UINT		Betriebsart des CP: • 0 = Telecontrol • 1 = GPRS direkt
TemporaryStation	BOOL		Bit 0: Temporäre Verbindung Bei aktivierter Option baut der CP eine Verbindung nur zeitweilig zum Senden von Daten auf. Nach erfolgreichem Übertragen der Telegramme baut der CP die Verbindung wieder ab. • 1: aktiviert (temporäre Verbindung) • 0: deaktiviert (permanente Verbindung)
SMS_Enabled	BOOL		Bit 1: SMS ermöglichen Die Aktivierung der Option ermöglicht der S7-Station das Versenden von SMS. • 1: aktiviert (SMS ermöglicht) • 0: deaktiviert (keine SMS)

Block für den Parameter "SMSC"

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 4-115 IF_CONF_SMS_Provider

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	10	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 28
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
SMSProvider	STRING [20]		Teilnehmernummer der SMS-Zentrale (SMSC) des GSM-Netzwerkbetreibers, mit dem der Mobilfunk-Vertrag für diese Station abgeschlossen wurde.

Block für den Parameter "PIN"

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 4-116 IF_CONF_PIN

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	11	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 16
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
Pin	STRING [8]		PIN der im CP gesteckten SIM-Karte Der Parameter ist nicht relevant, wenn die PIN richtig projektiert wurde. Bei falsch projektierter PIN kann die richtige PIN hiermit eingegeben werden.

Block für Überwachungszeiten

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Keep-Alive-Wartezeit" und "Betriebsart" des CP.

Tabelle 4-117 IF_CONF_TC_Timeouts

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	12	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 12
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
KeepAliveTimeout	-	-	- Reserviert - (nicht parametrierbar)

Parameter	Datentyp	Anfangswert	Beschreibung
SendTimeout	UINT		Verbindungsüberwachungszeit: Überwachungszeit der Verbindung zum Kommunikationspartner (Sekunden) Relevant in den Betriebsarten "Telecontrol" und "GPRS direkt"
RedialTimeout	UINT		Verbindungsaufbau-Verzögerung: Basiswert für die Wartezeit bis zum nächsten Verbindungsaufbauversuch nach erfolglosem Verbindungsaufbau. Nach jeweils 3 Versuchen wird der Basiswert bis max. 900 s verdoppelt. Wertebereich: 10...600 s. Bsp.: Basiswert 20 ergibt folgende Wählintervalle: dreimal 20 s, dreimal 40 s, dreimal 80 s etc. bis max. 900 s. Wenn ein zweiter Telecontrol-Server / Router projektiert ist, dann versucht der CP beim 4. Mal, sich mit dem zweiten Partner zu verbinden. Wenn der zweite Partner auch nicht erreichbar ist, dann versucht der CP beim 7. Mal, sich wieder mit dem ersten Partner zu verbinden. Nicht relevant für SMS-Verbindungen

Block für den Parameterbereich "Weckberechtigung"

Tabelle 4-118 IF_CONF_WakeupList

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	13	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 246
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
WakeupPhone [1...10]	ARRAY [1...10] of STRING [22]		Rufnummer des zum Wecken autorisierten Teilnehmers Der Stern (*) am Ende einer Rufnummer dient als Platzhalter für Durchwahlnummern.

Block für den Parameterbereich "Bevorzugte GSM-Netze"

Tabelle 4-119 IF_CONF_PrefProvider

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	14	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 46
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
Provider [1...5]	ARRAY [1...5] of STRING [6]		Alternative GSM-Netze mit Priorität 1 bis 5, in die sich der CP bevorzugt einwählt. Bis zu 5 Netze sind projektierbar. Nr. 1 mit höchster Priorität, Nr. 5 mit niedrigster Priorität. Eingabe des Public Land Mobile Network (PLMN) des Netzwerkbetreibers, bestehend aus Mobile Country Code (MCC) und Mobile Network Code (MNC). Beispiel (Testnetz der Siemens AG): 26276

Block für den Parameterbereich "DNS-Konfiguration"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Tabelle 4-120 IF_CONF_DNS

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	16	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
DNS_IP [1]	IP_V4		IP-Adresse des 1. Domain Name System Servers
DNS_IP [2]	IP_V4		IP-Adresse des 2. Domain Name System Servers

Block für den Parameterbereich "Uhrzeitsynchronisation"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Tabelle 4-121 IF_CONF_NTP

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	17	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 24
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
NTP_IP [1]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 1
...	...		(IP-Adresse von NTP-Server 2...3)
NTP_IP [4]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 4

Block zur Aktivierung / Deaktivierung von TeleService-Benutzern

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

SDT zur Aktivierung oder Deaktivierung von TeleService-Benutzern, die bereits im STEP 7-Projekt des CP projektiert sind. In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen" > "TeleService-Benutzerverwaltung".

Tabelle 4-122 IF_CONF_GPRS_UserList

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	19	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 506
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
GPRS_User [1...10]	ARRAY [1...10] of GPRS_User		TeleService-Benutzer Nr. 1 bis max. Nr. 10

Der Array wird gebildet aus den Parametersätzen für die TeleService-Benutzer ("GPRS_User" [1...n]).

Tabelle 4-123 GPRS_User [n] (Parameter für TeleService-Benutzer)

Parameter	Datentyp	Anfangswert	Beschreibung
UserName [n]	STRING [22]		TeleService-Benutzername
Password [n]	STRING [22]		- Der String muss leer sein! -
Diag_Allowed [n]	BOOL		- Reserviert - (nicht parametrierbar)
Teleserv_Allowed [n]	BOOL		Aktivierung des TeleService-Benutzers • 0 = Benutzer ist deaktiviert • 1 = Benutzer ist aktiviert
FW_Load_Allowed [n]	BOOL		- Reserviert - (nicht parametrierbar)

Block für die Parametrierung des TeleService-Zugangs (DNS-Name des Servers)

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-124 IF_CONF_TS_Name

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	20	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 266
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
ts_name	String [254]		Durch DNS auflösbarer Name ders TeleService-Servers
RemotePort	UINT		Port der Engineering-Station
Rank	UINT		Priorität des Servers [1] oder [2] 1 = Server 1, 2 = Server 2

Block für die Parametrierung des TeleService-Zugangs (IP-Adresse des Servers)

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-125 IF_CONF_TS_IF_V4

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	21	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)

Parameter	Datentyp	Anfangswert	Beschreibung
RemoteAddress	IP_V4		IP-Adresse des TeleService-Servers
RemotePort	UINT		Port des TeleService-Servers
Rank	UINT		Priorität des Servers [1] oder [2] 1 = Server 1, 2 = Server 2

S7-300/400-CP

CP 443-1 OPC UA

Programmbausteine für den OPC UA-Client

Übersicht der Programmbausteine für die OPC UA-Client-Funktion

Für die OPC UA-Client-Funktion des CP stehen die nachfolgend aufgeführten Funktionsbausteine in der Bibliothek "SIMATIC NET CP" zur Verfügung. Verwenden Sie die Bausteine in der CPU.

Einige Bausteine benötigen spezielle benutzerdefinierte PLC-Datentypen (UDTs), die Sie nachfolgend unterhalb des jeweiligen Programmbausteins finden. Die UDTs stehen als vorgefertigte Datentypen zur Verfügung. Legen Sie hierzu unter dem Eintrag "PLC-Datentypen" der Station einen neuen Datentyp an und tippen Sie im Feld "Datentyp" den Namen des UDT ein, beispielsweise "UA_SessionConnectInfo". Die jeweils benötigten Parameter werden dann automatisch in der Deklarationstabelle des UDT angelegt.

- UA_Connect
  Baut eine Verbindung für eine Sitzung mit einem OPC UA-Server auf.
  - UA_SessionConnectInfo
    Enthält die Verbindungsinformationen für den Bausteinparameter "SessionConnectInfo".
  - UA_UserIdentityToken
    Enthält die Daten der Benutzerauthentifizierung für den Bausteinparameter "UserIdentityToken".
- UA_NamespaceGetIndex
  Holt den Namensraumindex einer Namensraum-URI.
- UA_NodeGetHandleList
  Registriert Node-Ids am verbundenen Server und holt die Node-Handles in Form einer Liste.
  - UA_NodeID
    Enthält die Parameter zur Identifizierung des Ziel-Knotens für den Bausteinparameter "NodeID".
- UA_NodeReleaseHandleList
  Gibt am Server die Node-Handles der verwendeten Liste frei.
- UA_Disconnect
  Baut die Verbindung einer aktuellen Sitzung mit einem OPC UA-Server ab.

- UA_ReadList
  Liest mithilfe der Liste der Node-Handles die Daten aus Knoten des verbundenen Servers.
  - UA_NodeAdditionalInfo
    Spezifiziert das Item-Attribut und den Indexbereich für den Bausteinparameter "NodeAddInfos".
  - UA_IndexRange
    Legt den Start- und End-Index für den Bausteinparameter "IndexRange" fest.
  - UA_AnyPointer
    Verweist auf den Speicherbereich der CPU, in dem die Prozessdaten für den Bausteinparameter "Variables" abgelegt werden.
  - UA_TimeStamps
    Enthält die Zeitstempel der Elemente des Datenbereichs des Bausteinparameters "Variables".
- UA_WriteList
  Schreibt mithilfe der Liste der Node-Handles Daten in Knoten des verbundenen Servers.
  - UA_NodeAdditionalInfo
    Spezifiziert das Item-Attribut und den Indexbereich für den Bausteinparameter "NodeAddInfos".
  - UA_IndexRange
    Legt den Start- und End-Index für den Bausteinparameter "IndexRange" fest.
  - UA_AnyPointer
    Verweist auf den Speicherbereich der CPU, in dem die Prozessdaten für den Bausteinparameter "Variables" abgelegt sind.

Zusätzlich benötigte Systemfunktionen (SFC)

Für die vollständige Funktion der Programmbausteine für den OPC UA-Client werden folgende Systemfunktionen benötigt:

- BLKMOV (SFC20)
- TIME_TCK (SFC64)

UA_Connect benötigt für seine Funktion zusätzlich:

- LGC_GADR (SFC49)
- RDSYSST (SFC51)

Anlegen der SFCs

Legen Sie die Systemfunktionen an, indem Sie im Bausteinverzeichnis der CPU einen Organisationsbaustein öffnen, bswp. den OB1.

Klappen Sie im Bausteinkatalog, der mit dem Öffnen des OB1 eingeblendet wird, die entsprechende Bausteingruppe auf.

Sie finden die SFCs in folgenden Bausteinordnen:

- BLKMOV (SFC20): Gruppe "Einfache Anweisungen" > Ordner "Verschieben"
- TIME_TCK (SFC64): Gruppe "Erweiterte Anweisungen" > Ordner "Datum und Uhrzeit"

- LGC_GADR (SFC 49): Gruppe "Erweiterte Anweisungen" > Ordner "Adressierung"
- RDSYSST (SFC 51): Gruppe "Erweiterte Anweisungen" > Ordner "Diagnose"

Ziehen Sie die SFCs in das Netzwerk des Organisationsbausteins und versorgen Sie die Parameter der SFCs.

Aufruf und Zusammenspiel der Programmbausteine

Hinweis

Allgemeine Hinweise zu den Programmbausteinen und Instanz-DBs

Legen Sie die Funktionsbausteine mit "Standard-Zugriff" an. Lassen Sie in den jeweils zugeordneten Instanz-Datenbausteinen die Option "Remanenz" deaktiviert.

Jeder Programmbaustein muss zyklisch solange aufgerufen werden, bis der Zustandsparameter "Done" oder "Error" auf 1 wechselt.

Innerhalb einer Sitzung kann gleichzeitig nur ein einziger Baustein aufgerufen werden. Verschiedene Bausteine können innerhalb einer Sitzung nur nacheinander bearbeitet werden.

Mit UA_Connect bauen Sie vom CP als OPC UA-Client eine Verbindung zu einem UA-Server auf und öffnen damit eine Sitzung. Der Baustein muss für jede Verbindung zu einem UA-Server aufgerufen werden. Bei Bedarf für mehr als eine Verbindung zu einem UA-Server kann der Baustein mehrmals mit unterschiedlichen Instanz-Datenbausteinen (IDBs) aufgerufen werden. Sie können dies beispielsweise nutzen, um über eine Verbindung zu lesen und über eine zweite Verbindung zu schreiben oder um über mehrere Verbindungen unterschiedliche Daten zu lesen bzw. zu schreiben.

Für Verbindungen zu mehreren UA-Servern muss der Bausteine UA_Connect mehrmals mit unterschiedlichen Eingangsparametern aufgerufen werden. Der CP kann als UA-Client Verbindungen mit bis zu fünf UA-Servern aufbauen.

Zur Abfrage des Namensraumindex einer Namensraum-URI des UA-Servers rufen Sie innerhalb einer Sitzung einen UA_NamespaceGetIndex auf. Zur Abfrage mehrerer Namensraum-URIs können Sie den Baustein mehrmals mit unterschiedlichen Eingangsparametern aufrufen.

Zum Anlegen einer Handle-Liste als Vorbereitung der Lese- und Schreib-Dienste rufen Sie UA_NodeGetHandleList auf. Der Baustein wird für jeden Ziel-Knoten separat mit den entsprechenden Angaben am Eingangsparameter "NodeIDs" aufgerufen.

Anschließend können Sie über die Bausteine UA_ReadList und UA_WriteList die Daten der Items lesen und schreiben. Die beiden Bausteine können Sie innerhalb einer Sitzung beliebig oft aufrufen.

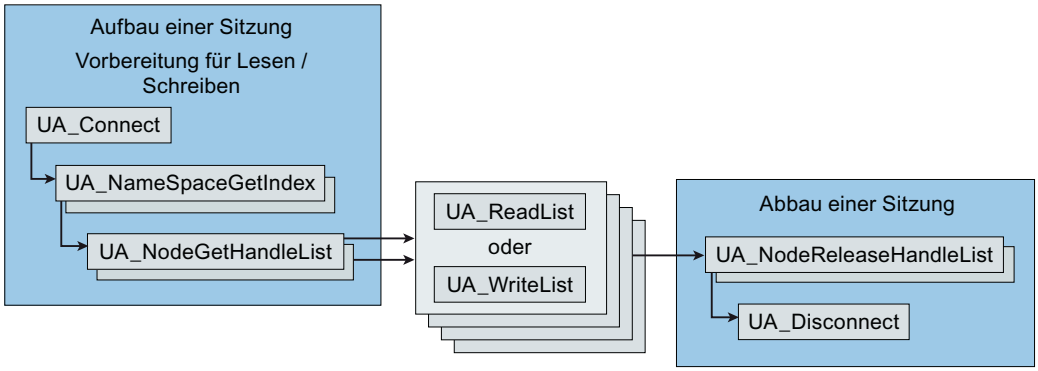

Bild 4-3 Aufruf und Zusammenspiel der Client-Programmbausteine

Wenn die Verbindung nicht mehr benötigt wird, dann geben Sie die Handles am Server über den Baustein UA_NodeReleaseHandleList wieder frei. Die Handles werden damit am Server wieder gelöscht.

Die Verbindung wird mit UA_Disconnect abgebaut und die Sitzung damit beendet.

Siehe auch

Zeitliche Überwachung der Bausteine (Seite 4064)

Zeitliche Überwachung der Bausteine

Parameter für das Zeitverhalten der Bausteine

Für die Steuerung und die Überwachung des zeitlichen Verhaltens der Programmbausteine stehen die folgenden drei Parameter zur Verfügung:

- Timeout
  Eingangsparameter an allen Bausteinen für den OPC UA-Client
- SessionTimeout
  Eingangsparameter am Baustein UA_Connect
- MonitorConnection
  Eingangsparameter am Baustein UA_Connect

Da die Parameter Einfluss auf den Ablauf der OPC UA-Kommunikation insgesamt haben, werden nachfolgend Hinweise zu diesen Parametern vor die allgemeine Bausteinbeschreibung gesetzt.

Timeout

Mit diesem Eingangsparameter überwachen Sie jeden Aufruf eines Bausteins. Wenn ein Bausteinaufruf nicht innerhalb der jeweils projektierten maximal zulässigen Zeit abgeschlossen werden kann, wird der Zustandsparameter "Error" auf 1 gesetzt und die Bearbeitung abgebrochen.

SessionTimeout

Mit diesem Eingangsparameter überwachen Sie die Dauer einer Sitzung ohne Datenverkehr, ohne dass der Baustein "UA_Disconnect" zum Abbau einer Verbindung aufgerufen wird.

Der minimale Wert von "SessionTimeout" beträgt generell 30 Sekunden. Bei Überschreitung des Werts wird die Verbindung abgebaut.

Der Parameter erlaubt einem Server, gebundene Ressourcen abzubauen, wenn der Client die Sitzung für längere Zeit nicht nutzt.

Eine ausreichend lange Zeit gewährleistet dagegen, dass eine Sitzung nicht sofort abgebaut wird, wenn Daten beispielsweise aufgrund von Netzwerkstörungen nicht sofort übertragen werden können.

Der Wert von "SessionTimeout" sollte in der Regel größer sein als die Werte von "Timeout" bei den verwendeten Bausteinen "UA_Connect".

MonitorConnection

Mit diesem Eingangsparameter wird der Zustand einer Verbindung ohne Datenverkehr geprüft. Nach Ablauf dieser Verbindungsüberwachungszeit wird ein Telegramm zur Prüfung des Verbindungszustandes mit dem Server gesendet.

Damit bei fehlendem Datenverkehr und dem Ansprechen der Verbindungsüberwachungszeit nicht gleich die Sitzung über den Parameter "SessionTimeout" abgebrochen wird, wird empfohlen, einen Wert für "MonitorConnection" zu wählen, der kleiner ist als die Hälfte von "SessionTimeout".

Siehe auch

Programmbausteine für den OPC UA-Client (Seite 4061)

UA_Connect

Funktion

Funktion des Bausteins

Mit dem Baustein UA_Connect bauen Sie vom CP als OPC UA-Client eine Verbindung zu einem UA-Server auf und öffnen damit eine Sitzung.

Als Zieladresse geben Sie am Parameter "ServerEndpointUrl" die URL des UA-Servers an.

Die Angaben zu den Verbindungsinformationen am Parameter "SessionConnectInfo" hinterlegen Sie in einem Datenbaustein.

UA_Connect benötigt zusätzlich die Systemfunktionen LGC_GADR (SFC 49) und RDSYSST (SFC 51).

Siehe auch

Parameter (Seite 4066)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-126 Parameter des Bausteins UA_Connect

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ServerEndpointUrl	IN	STRING	Max. 254 Zeichen	Adresse (URL) des Verbindungspartners (Server). Nur IPv4-Adressen sind zugelassen.
SessionConnectInfo	IN	UDT	Siehe "Bedeutung"	Verbindungsinformationen, siehe UASession-ConnectInfo (Seite 4079).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
ConnectionHdl	OUT	DWORD	1 .. 5	Eindeutige Kennzeichnung einer aufgebauten Verbindung Sie wird von den anderen Bausteinen als Eingangsparameter benötigt.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085). Hinweis: Wenn der Fehler-Code B000F002 ausgegeben wird, dann ist die OPC UA-Sitzung noch nicht aufgebaut. Rufen Sie den Baustein erneut auf.

UA_NamespaceGetIndex

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NamespaceGetIndex holen Sie den Namensraumindex der Namensraum-URI des verbundenen UA-Servers.

Siehe auch

Parameter (Seite 4067)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-127 Parameter des Bausteins UA_NamespaceGetIndex

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus. Solange Execute = 1 ist, können die anderen Eingangsparameter nicht verändert werden.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NamespaceUri	IN	STRING	Max. 254 Zeichen	Angabe der Adresse (Namensraum-URI) des Ziel-Servers Beispiel für den Namensraum der CPU-Symbole in der S7-400 mit CP 443-1 OPC UA: http://www.siemens.com/simatic-classic-s7-opcua
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
NamespaceIndex	OUT	WORD	0 .. 65535	Ausgabe des Namensraum-Index des am Parameter "NamespaceUri" angegebenen Namensraums. Die Namensraum-Indices 0 und 1 sind von der OPC Foundation reserviert und spezifiziert.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).

UA_NodeGetHandleList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NodeGetHandleList registrieren Sie NodeIDs am verbundenen UA-Server.

Der UA-Server liefert die Handles zurück. Diese werden vom Baustein am Parameter "NodeHdls" ausgegeben. Auf die Items der Node-Handles greifen Sie mit den Bausteinen "UA_ReadList" und "UA_WriteList" an deren Eingangsparameter "NodeHdls" zu.

Die NodeIDs des Ziel-Servers, die Sie am Eingangsparameter "NodeIDs" angeben, müssen Ihnen bekannt sein. Diese legen Sie in einem UDT "UANodeIDs" ab.

Die Anzahl der Ziel-Knoten am Eingangsparameter "NodeIDCount" ist immer 1, daher müssen Sie den Baustein für jede NodeID separat aufrufen.

Siehe auch

Parameter (Seite 4069)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-128 Parameter des Bausteins UA_NodeGetHandleList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeIDCount	IN	WORD	1	Anzahl der Elemente im Datenbereich von "NodeIDs". Die Anzahl muss 1 sein.
NodeIDs	IN	ARRAY of UDT	Siehe "Bedeutung"	Struktur der Parameter des Ziel-Knotens des OPC UA-Servers. Die Anzahl der NodeIDs (1) muss identisch sein mit der Anzahl am Ausgangsparameter "NodeHdls". Zum Aufbau des UDT siehe UANodeID (Seite 4082).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeHdls	OUT	ARRAY of DWORD	0 .. 4 294 967 295	Array der Handles, die der OPC UA-Server als Antwort auf die Anfrage des Client vergibt. Die Handles dienen der eindeutigen Identifikation des Items im Server. Die Handles sind solange gültig, bis sie durch Aufruf des Bausteins UA_NodeReleaseHandle-List aufgelöst, d. h. ungültig werden. Wenn eine Sitzung abgebaut wird, löst der CP alle registrierten Node-Handles dieser Sitzung auf. Der Parameter wird für den Eingangsparameter "NodeHdls" der Bausteine "UA_ReadList" und "UA_WriteList" verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält die Fehlernummern für alle Elemente des Datenbereichs von "NodeIDs". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).

UA_NodeReleaseHandleList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NodeReleaseHandleList geben Sie am verbundenen UA-Server die Node Handles der aktuellen Sitzung frei. Die Liste wird damit gelöscht.

Der Eingangsparameter "NodeHdls" referenziert den Datenbaustein, in den die Handles vom Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" geschrieben wurden.

Die Anzahl der Handles am Eingangsparameter "NodeHdlCount" ist immer 1, daher müssen Sie den Baustein pro Handle separat aufrufen.

Siehe auch

Parameter (Seite 4071)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-129 Parameter des Bausteins UA_NodeReleaseHandleList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung / Bemerkung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im Datenbereich von "NodeHdls". Die Anzahl muss 1 sein.
NodeHdls	IN	ARRAY of DWORD	0 .. 4 294 967 295	Array der IDs der aufzulösenden Handles. Bei erfolgreicher Bausteinbearbeitung werden die Handles am Server aufgelöst und damit ungültig.
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung / Bemerkung
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehler-Codes für alle Elemente des Datenbereichs von "NodeHdls".

UA_Disconnect

Funktion

Funktion des Bausteins

Mit dem Baustein UA_Disconnect bauen Sie eine bestehende Verbindung zwischen dem CP als OPC UA-Client und einem UA-Server ab. Die bestehende Sitzung wird damit beendet.

Siehe auch

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-130 Parameter des Bausteins UA_Disconnect

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).

UA_ReadList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_ReadList lesen Sie Daten aus dem verbundenen UA-Server.

Die zu lesenden Attribute programmieren Sie in einem UDT "UANodeAdditionalInfo", der vom Eingangsparameter "NodeAddInfos" referenziert wird.

Die am Eingangsparameter "NodeHdls" benötigten Handles werden aus dem Array des Ausgangsparameters "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.

Die Anzahl der NodeIDs am Eingangsparameter "NodeIDCount" ist 1, daher müssen Sie den Baustein für jeden Ziel-Knoten separat aufrufen.

Siehe auch

Parameter (Seite 4074)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-131 Parameter des Bausteins UA_ReadList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im zu lesenden Zielbereich. Die Anzahl muss 1 sein.
NodeHdls	IN	ARRAY of DWORD	1	Array der Handles der Ziel-Knoten (Items) des OPC UA-Servers. Die Handles werden aus dem Array des Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.
NodeAddInfos	IN	ARRAY of UDT	Siehe "Bedeutung"	Spezifiziert das zu lesende Attribut der Items und den Indexbereich. Zum Aufbau des UDT siehe UANodeAdditionalInfo (Seite 4082).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Variables	IN_OUT	UDT	Siehe "Bedeutung"	Verweis auf die zu lesenden Daten. Zum Aufbau des UDT siehe UAAnyPointer (Seite 4084). Zu den unterstützten Datentypen der Prozessdaten siehe Kapitel Datentypen (Seite 4076).
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehlernummern für alle Elemente des Datenbereichs von "Variables". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).
TimeStamps	OUT	UDT	Siehe "Bedeutung"	Enthält Zeitstempel für alle Elemente des Datenbereichs von "Variables". Zum Aufbau des UDT siehe UATimeStamp (Seite 4083).

UA_WriteList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_WriteList schreiben Sie Daten in den verbundenen UA-Server.

Die zu schreibenden Attribute programmieren Sie in einem UDT "UANodeAdditionalInfo", der vom Eingangsparameter "NodeAddInfos" referenziert wird.

Die am Eingangsparameter "NodeHdls" benötigten Handles werden aus dem Array des Ausgangsparameters "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.

Die Anzahl der NodeIDs am Eingangsparameter "NodeIDCount" ist 1, daher müssen Sie den Baustein für jeden Ziel-Knoten separat aufrufen.

Siehe auch

Parameter (Seite 4075)

Programmbausteine für den OPC UA-Client (Seite 4061)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-132 Parameter des Bausteins UA_WriteList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im zu schreibenden Zielbereich. Die Anzahl muss 1 sein.

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeHdls	IN	ARRAY of DWORD	1	Array der Handles der Ziel-Knoten (Items) des OPC UA-Servers. Die Handles werden aus dem Array des Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.
NodeAddInfos	IN	ARRAY of UDT	Siehe "Bedeutung"	Spezifiziert das zu schreibende Attribut der Items und den Indexbereich. Zum Aufbau des UDT siehe UANodeAdditionalInfo (Seite 4082).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Variables	IN_OUT	UDT	Siehe "Bedeutung"	Verweis auf die zu schreibenden Daten. Zum Aufbau des UDT siehe UAAnyPointer (Seite 4084). Zu den unterstützten Datentypen der Prozessdaten siehe Kapitel Datentypen (Seite 4076).
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehlernummern für alle Elemente des Datenbereichs von "Variables". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 4085).

Datentypen

Datentypen und Speicherbereiche

Der CP 443-1 OPC UA unterstützt die nachfolgend aufgeführten Datentypen für Prozessdaten.

Die entsprechenden Symbole in der CPU sind für die jeweils aufgeführten Operandenbereiche zulässig. Die Speicherbereiche haben folgende Abkürzungen. Die Kürzel in Klammern sind die Abkürzungen in Abhängigkeit der eingestellten Mnemonik (Deutsch/Englisch).

- E = Prozessabbild der Eingänge (E/I)
- A = Prozessabbild der Ausgänge (A/Q)
- PE = Peripheriebereich Eingänge (PE/PI)
- PA = Peripheriebereich Ausgänge (PA/PQ)
- M = Merker
- DB = Datenbaustein / Systemdatenbaustein / Instanzdatenbaustein

Peripheriebereiche werden nur vom OPC UA-Server unterstützt.

Tabelle 4-133 Datentypen

Größe [bit], Typ	Bereich	Datentyp			Anmerkung	Operandenbereich der CPU
		SIMATIC S7	IEC 61131-3	OPC UA		
1	0 .. 255	BOOL	BOOL	Boolean	0 = false 1 .. 255 = true	E, A, PE, PA, M, DB
Zahlen						
8, UInt8	0 .. 255	BYTE	BYTE, USINT	Byte		E, A, PE, PA, M, DB
8, UInt8	0 .. 255	CHAR	CHAR	Byte	ASCII-Zeichen 32 .. 255	E, A, PE, PA, M, DB
16, UInt16	0 .. 65535	WORD	WORD, UINT	UInt16		E, A, PE, PA, M, DB
16, Int16	-32768 .. 32767	INT	INT	Int16		E, A, PE, PA, M, DB
32, UInt32	0 .. 4294967295	DWORD	DWORD, UDINT	UInt32		E, A, PE, PA, M, DB
32, Int32	2147483648 .. 2147483647	DINT	DINT	Int32		E, A, PE, PA, M, DB
32, Float	±1.175 495e-38 .. 3.402823e+38	REAL	REAL	Float		DB
Zeit und Datum						
16, UInt16	0 .. 999	S5TIME	TIME	UInt16	SIMATIC-Zeit Zur Codierung siehe Online-Hilfe von STEP 7.	DB
16, UInt16	D#1990-1-1 .. D#2168-12-31 (0 .. 65378)	DATE	DATE	UInt16	IEC-Datum [d]	DB
32, UInt32	TOD#0:0:0.0 .. TOD#23:59:59.999 (0 .. 86399999)	TIME_OF_DAY (TOD)	TIME_OF_DAY	UInt32	Uhrzeit [ms]	DB

Größe [bit], Typ	Bereich	Datentyp			Anmerkung	Operandenbereich der CPU
		SIMATIC S7	IEC 61131-3	OPC UA		
32, UInt32	-24D_20H_31M_23S_648MS .. T#24D_20H_31M_23S_647MS (-2147483648 .. 2147483647)	TIME	TIME	UInt32	IEC-Zeit [ms]	DB
Zusammengesetzte Datentypen						
64	1990-1-1-0:0:0.0 .. 2089-12-31-23:59:59.999	DATE_AND_TIME	DT DATE_AND_TIME	UInt32+ UInt32	Datum und Uhrzeit Zur Codierung siehe Online-Hilfe von STEP 7.	DB
254 * 8		STRING *	STRING	String	Max. 254 Byte Nutzdaten	DB
		ARRAY *			Multidimensionales Feld aus elementaren oder zusammengesetzten Datentypen	DB
		UDT *			Anwenderdefinierter Datentyp für spezifische Parametergruppen	DB

* Zur Codierung siehe unten.

Die generelle Beschreibung zur Codierung der Datentypen finden Sie in der Online-Hilfe von STEP 7.

Besonderheiten bei zusammengesetzten Datentypen

Bei zusammengesetzten Datentypen, die von der OPC UA-Funktion des CP verwendet werden, gelten besondere Codierungs-Vorschriften, die nachfolgend beschrieben sind.

STRING

Ein STRING kann max. 254 Zeichen enthalten.

Es sind ASCII-Zeichen aus dem Bereich 32 (0x20) .. 255 (0xFF) erlaubt.

Davon sind folgende Zeichen nicht erlaubt: 127 (0x7F), 129 (0x81), 141 (0x8D), 143 (0x8F), 144 (0x90), 157 (0x9D)

Besonderheiten für den OPC UA-Server

Die Zeichen werden im OPC UA-Server des CP im Format UTF-8 codiert.

Besonderheiten für den OPC UA-Client

Bei Prozessdaten mit dem Datentyp STRING, die vom UDT "UAAnyPointer" referenziert werden, wird als Datentyp "2" (BYTE) angegeben.

Wählen Sie als Wiederholfaktor im UAAnyPointer immer den maximalen Wert von 256 (0x100): 254 für die Nutzdaten + 2 für den Header.

ARRAY

Besonderheiten für den OPC UA-Client

Die Client-Programmbausteine unterstützen nur die Übertragung von eindimensionalen Arrays aus dem Speicherbereich "DB".

Die Übertragung des Datentyps "ARRAY of BOOL" wird nicht von den Client-Programmbausteinen unterstützt.

Bei einem ARRAY of STRING muss bei den Grenzen der Dimension jedes String-Elements "n" mal die maximale Länge (256) angegeben werden, wobei "n" die Anzahl der Strings im Array ist.

Besonderheiten für den OPC UA-Server

Der Datentyp "ARRAY of BOOL" ist in allen Fällen lesbar.

Der Datentyp "ARRAY of BOOL" ist nur dann schreibbar, wenn die Länge ein Vielfaches von 8 ist.

UDT

Den Aufbau der UDTs, die von den Programmbausteinen für die OPC UA-Client-Funktion benötigt werden, finden Sie im Kapitel AUTOHOTSPOT.

Parametertypen

Die Parametertypen "TIMER" und "COUNTER" werden nur für den OPC UA-Server des CP unterstützt.

UDTs für den CP 443-1 OPC UA

UASessionConnectInfo

UASessionConnectInfo

Entnehmen Sie der folgenden Tabelle die Bedeutung der Verbindungsinformationen für den Parameter "SessionConnectInfo" des Bausteins UA_Connect.

Tabelle 4-134 Parameter des UDT UASessionConnectInfo

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SessionName	STRING	0 .. 254 Zeichen	Name der Sitzung Wenn der String leer bleibt, wird folgendes als Sitzungsname systemseitig eingetragen: <Connection>+<Dezimalwert von "ConnectionHdl">
ApplicationName	STRING	0 .. 254 Zeichen	Der Parameter wird durch STEP 7 mit dem in HW Konfig projektierten Applikationsnamen des CP (OPC UA-Client) belegt.

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SecurityMsgMode	WORD	0 .. 3	Security-Verfahren • 0 = Bestmögliches Verfahren • 1 = Kein Security-Verfahren • 2 = Authentifizieren • 3 = Authentifizieren und verschlüsseln
SecurityPolicy	WORD	0 .. 4	Security-Profil • 0 = Bestmögliches Security-Profil • 1 = Kein Security-Profil • 2 = Basic128Rsa15 • 3 = Basic256 • 4 = Basic256Sha256
CertificateStore	STRING	0 .. 254 Zeichen	Zertifikatspeicher des CP (OPC UA-Client) Der Parameter wird durch STEP 7 eingesetzt.
ClientCertificate-Name	STRING	0 .. 254 Zeichen	Name des Client-Zertifikats Der Parameter wird durch STEP 7 eingesetzt.
ServerUri	STRING	0 .. 254 Zeichen	Server-Adresse (URI), die auch im Server-Zertifikat hinterlegt ist. Der Parameter darf nur leer gelassen werden für Sitzungen ohne Sercurity-Verfahren. Beispiel: urn:<ApplicationName>:GUID
CheckServerCertificate	BOOL	0, 1	Abgleich (Prüfung) des vom Server gesendeten Zertifikats mit dem Server-Zertifikat, das im Zertifikatspeicher des Client-CP gespeichert ist: • 0 = Keine Prüfung • 1 = Prüfung **Hinweis:** Die Einstellungen dieses Parameters werden von den Security-Einstellungen in der Projektierung überschrieben. Wenn Sie die Prüfung aktivieren (1), aber die Security-Funktionen in der Projektierung des CP deaktiviert sind, dann wird der Verbindungsaufbau mit einer Fehlermeldung abgebrochen. Empfehlung: Lassen Sie den Parameter in der Voreinstellung (0), da die Security-Funktionen über die Projektierung des CP aktiviert werden.
TransportProfile	WORD	1	1 = UATP_UATcp Nur dieses Transportprofil gemäß der PLCopen-Spezifikation wird unterstützt.
UserIdentityToken	UDT	Siehe "Bedeutung"	Daten der Benutzerauthentifizierung, siehe UAUserIdentityToken (Seite 4081).
VendorSpecificParameter	WORD	Siehe "Bedeutung"	Eingabe der logischen Adresse des CP 443-1 OPC UA. Sie finden diese im STEP 7-Eigenschaftendialog des CP als Eingangsadresse im Register "Adressen".

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SessionTimeout	TIME	30000 .. 86 400 000	Maximale Zeitspanne, über die eine Sitzung ohne Datenverkehr aufrecht erhalten wird (Millisekunden). Bei Überschreitung des Werts wird die Sitzung (Verbindung) abgebrochen. In diesem Fall müssen Sie die Verbindung über den Aufruf von UA_Connect neu aufbauen. Bei Programmierung von Werten außerhalb des zulässigen Wertebereichs (30 Sekunden ... 1 Tag) wird der Wert 1200000 (20 Minuten) verwendet.
MonitorConnection	TIME	5000 .. 86 400 000	Verbindungsüberwachungszeit (Millisekunden) Zeitspanne ohne Datenverkehr, nach welcher der Client durch Versenden eines Telegramms die Verbindung zum Server prüft. Voreinstellung: 15000 (15 Sekunden)
LocaleIDs	ARRAY[1..5] of String[6]	Beispiele: en-US, de-DE, zh-CHS ...	Optionale Sprach- und Regionenkennung gemäß RFC 3066. 0 = keine oder unbekannte LocaleID.

UAUserIdentityToken

UAUserIdentityToken

Entnehmen Sie der folgenden Tabelle die Bedeutung der Benutzerauthentifizierungsdaten, welche den Parameter "UserIdentityToken" des UDT "UASessionConnectInfo" versorgen.

Tabelle 4-135 Parameter des UDT UAUserIdentityToken

Parameter	S7-Datentyp	Wertebereich	Bedeutung
UserIdentity TokenType	WORD	0, 1	0 = Keine Authentifizierung 1 = Authentifizierung Wenn der Kommunikationspartner (Server) eine Authentifizierung über Benutzername und Passwort verlangt, dann setzen Sie diesen Parameter auf 1 und die beiden nachfolgenden entsprechend den Anforderungen des Servers.
TokenParam1	STRING	1 .. 254 Zeichen	Benutzername
TokenParam2	STRING	1 .. 254 Zeichen	Passwort

UANodeID

UANodeID

Entnehmen Sie der folgenden Tabelle die Bedeutung der Parameter von UANodeID zur Identifizierung des Ziel-Knotens im OPC UA-Server. UANodeID versorgt den Parameter "NodeIDs" des Bausteins UA_NodeGetHandleList".

Tabelle 4-136 Parameter des UDT UANodeID

Parameter	S7-Datentyp	Wertebereich	Bedeutung
NamespaceIndex	WORD	0 .. 65535	Index des Namensraums des Servers
Identifier	STRING	Max. 254	Spezifiziert die NodeID im Namensraum-Index.
IdentifierType	WORD	1 .. 2	Legt das Format und den Geltungsbereich (i. d. R. der Server) der NodeID fest. Unterstützte Typen: • 1: UAIdentifierType_String String-Identifier unterscheiden Groß-/Kleinschreibung. • 2: UAIdentifierType_Numeric Numerischer Identifier

UANodeAdditionalInfo

UANodeAdditionalInfo

Spezifiziert das Attribut des Items und den Indexbereich für den Bausteinparameter "NodeAddInfos".

Tabelle 4-137 Parameter des UDT UANodeAdditionalInfo

Parameter	S7-Datentyp	Wertebereich	Bedeutung
AttributeID	WORD	13	Attribut der Items. Nur Attribut 13 (UAAI_Value) für den Wert des Items wird unterstützt.
IndexRangeCount	WORD	0 .. 1	Anzahl der Indexbereiche Der Wert ist bei elementaren Datentypen unerheblich. Für Items mit Datentyp ARRAY gilt: • 0: Ein einziger Index. Das gesamte Array wird gelesen/geschrieben. • 1: Ein durch "IndexRange" definierter Teilbereich eines Arrays.
IndexRange	UDT	Siehe "Bedeutung"	Bereich des Arrays für den Parameter IndexRangeCount. Zum Aufbau des UDT siehe UAIndexRange (Seite 4083).

UAIndexRange

UAIndexRange

Legt den Start- und End-Index für den Parameter "IndexRange" des UDT "UANodeAdditionalInfo" fest.

Tabelle 4-138 Parameter des UDT UAIndexRange

Parameter	Datentyp	Wertebereich	Bedeutung
StartIndex	WORD	0 .. 65535	Index, ab dem gelesen wird.
EndIndex	WORD	0 .. 65535	Index, bis zu dem gelesen wird.

Regeln:

Wenn IndexRangeCount = 1 verwendet wird, gelten folgende Regeln für die Programmierung des Indexbereichs:

- StartinIndex und EndIndex müssen zugewiesen werden.
- Für den Zugriff auf mehr als ein Element muss der StartIndex kleiner als der EndIndex sein. Nichtbeachtung führt zu einem Fehler mit der Nummer 80360000.
- Wenn Sie ein einzelnes Element des Arrays lesen wollen, dann geben Sie für StartIndex und EndIndex den gleichen Wert an.
- Um auf alle Elemente des Arrays zugreifen zu können, müssen StartIndex und EndIndex abhängig von der Gesamtzahl der Elemente des Arrays zugewiesen werden.
  Angaben außerhalb der Array-Größe führen zu einem Fehler mit der Nummer 80370000.

Beispiel für die Zuweisung von "StartIndex" und "EndIndex" bei Arrays:

Aus einem Array mit der Größe von 10 Elementen (Nr. 1 .. 10) sollen die Elemente 3 bis 5 gelesen werden. Die beiden Indices werden wie folgt programmiert:

- StartIndex = 2 (Element Nr. 3)
- EndIndex = 4 (Element Nr. 5)

UATimeStamp

UATimeStamp

Der UDT enthält die Zeitstempel der Elemente des Datenbereichs in Millisekunden. UATimeStamp versorgt den Parameter "TimeStamps" des Bausteins "UA_ReadList".

Tabelle 4-139 Parameter des UDT UATimeStamp

Parameter	S7-Datentyp	Wertebereich	Bedeutung
TimestampL	WORD	Siehe unten	Niederwertiges Wort (Bit 0..15)
TimestampH	WORD	Siehe unten	Höherwertiges Wort (Bit 16..31)

Wertebereich des kompletten Doppelworts:

TOD#0:0:0.0 .. TOD#23:59:59.999 (0 .. 86 399 999 Millisekunden)

UAAnyPointer

UAAnyPointer

Der UDT "UAAnyPointer" wird vom Parameter "Variables" des Programmbausteins referenziert.

Der UDT verweist auf den Speicherbereich der Prozessdaten in der CPU. Prozessdaten in den Operandenbereichen "Eingänge" oder "Ausgänge" müssen hierfür in einen Datenbaustein kopiert werden.

Legen Sie den UDT in einem Datenbaustein ab.

Tabelle 4-140 Parameter des UDT UAAnyPointer

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SyntaxID	WORD	10	Der Wert für die Syntax-ID ist immer 10.
DataType	WORD	Siehe "Bedeutung"	Datentyp der Ziel-Knoten. Zum Wertebereich siehe unten, Tabelle "Codierung des Datentyps".
RepetitionFactor	WORD	0 .. 65535	Wiederholfaktor Für weitere Informationen siehe STEP 7-Online-Hilfe.
DB_Number	WORD	0 .. 65535	Nummer des Datenbausteins (DB) Geben Sie alternativ die Nummer eines DB oder einen Speicherbereich an. Wenn Sie einen DB angeben, dann geben Sie unter "MemArea" eine Null ein.
MemArea	WORD	Siehe "Bedeutung"	Speicherbereich Geben Sie alternativ die Nummer eines DB oder einen Speicherbereich an. Wenn Sie einen Speicherbereich angeben, dann geben Sie unter "DB_Number" eine Null ein. Zum Wertebereich des Speicherbereichs siehe unten, Tabelle "Codierung des Speicherbereichs".
ByteOffset	WORD	Abhängig vom Speicherbereich, siehe "Bedeutung"	Byte-Offset im angegebenen Speicherbereich, ab dem auf die Daten zugegriffen wird.
BitOffset	WORD	0 .. 7	Bit-Offset im angegebenen Speicherbereich

Datentyp

Die folgende Tabelle erläutert die Codierung des Datentyps im Parameter "DataType" des UDT UAAnyPointer.

Tabelle 4-141 Codierung des Datentyps

Hexadezimal-Code	S7-Datentyp	Beschreibung
b#16#01	BOOL	Bit
b#16#02	BYTE	Byte (8 Bits)
b#16#03	CHAR	Zeichen (8 Bits)
b#16#04	WORD	Wort (16 Bits)
b#16#05	INT	Ganzzahl (16 Bits)
b#16#06	DWORD	Wort (32 Bits)
b#16#07	DINT	Ganzzahl (32 Bits)
b#16#08	REAL	Gleitpunktzahl (32 Bits)
b#16#09	DATE	Datum
b#16#0A	TIME_OF_DAY (TOD)	Uhrzeit
b#16#0B	TIME	Zeit
b#16#0C	S5TIME	Datentyp S5TIME
b#16#0E	DATE_AND_TIME (DT)	Datum und Zeit (64 Bits)
b#16#02 *	STRING	Zeichenkette

* Der String verwendet den untergeordneten Datentyp BYTE. Beachten Sie die Besonderheiten im Kapitel Datentypen (Seite 4076).

Speicherbereich

Die folgende Tabelle erläutert die Codierung des Speicherbereichs im Parameter "MemArea" des UDT UAAnyPointer.

Tabelle 4-142 Codierung des Speicherbereichs

Hexadezimal-Code	Bereich	Beschreibung
b#16#80	P	Speicherbereich der Peripherie
b#16#81	E	Speicherbereich der Eingänge
b#16#82	A	Speicherbereich der Ausgänge
b#16#83	M	Speicherbereich der Merker
b#16#84	DB	Datenbaustein

Fehlernummern

Auswertung der Ausgangsparameter

Die Ausgangsparameter Done (BOOL), Error (BOOL) und ErrorID (DWORD) der OPC UA-Client-Programmbausteine werden bei jedem Bausteinaufruf aktualisiert. Werten Sie die Zustände der Ausgangsparameter im Anwenderprogramm aus.

Beispiel:

Nach dem Aufruf liefert der Programmbaustein UA_Connect folgende Werte an den drei Ausgangsparametern:

- Done = 1
- Error = 0
- ErrorID = 00000000

Bedeutung: Der Auftrag wurde erfolgreich abgeschlossen.

Bedeutung der Fehlernummern (hexadezimal) von ErrorID

Tabelle 4-143 Bedeutung von ErrorID in Zusammenhang mit Done und Error

Done	Error	ErrorID (hex)	Name	Bedeutung
1	0	00000000		Die Bausteinbearbeitung wurde erfolgreich durchgeführt.
CP-spezifische Fehlernummern				
0	1	B0006001		Fehlende positive Flanke zum Aufruf des Programmbausteins
0	1	B0007001		Zeitüberschreitung
0	1	B0008004		Ungültiger NodeIDCount
0	1	B0008086		Ungültiges Node-Handle
0	1	B0008087		Die maximale Anzahl an Verbindungen (5) ist erreicht.
0	1	B0008090		Falsche logische Adresse. Verbindungsaufbau abgebrochen.
0	1	B0008093		Die logische Adresse kann nicht verarbeitet werden. Eine unzulässige Baugruppe wird verwendet.
0	1	B00080A1		Das Node-Handle wurde nicht für diese Verbindung eingerichtet.
0	1	B00080C3		Die Verbindungsressourcen der CPU sind vorübergehend ausgeschöpft.
0	1	B0009085		Der maximale Speicherplatz für Items (64000 Byte) des OPC UA-Client ist überschritten.
0	1	B000800A		Der OPC UA-Client wurde in der Projektierung nicht aktiviert.
0	1	B0008723		Der Lese-/Schreibauftrag referenziert einen nicht existierenden Datenbereich in einem DB.
0	1	B000F001		In einem Zyklus wurden zwei Funktionsbausteine aufgerufen.
0	1	B000F002		OPC UA-Sitzung noch nicht aufgebaut. Rufen Sie den Baustein UA_Connect erneut auf.
0	1	B000F003		Kommunikationsfehler
0	1	B000F004		Ungültiger Wert für den Parameter "SyntaxID" im UDT "UAAnyPointer"
0	1	B000F005		Ungültiger Datentyp in UAAnyPointer
0	1	B000F006		Ungültiger Wert für den Parameter "RepetitionFactor" im UDT "UAAnyPointer"
0	1	B000F007		Ungültiger Operandenbereich von UAAnyPointer

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	B000F010		Ungültiger Operandenbereich in einem Lese- oder Schreibauftrag: Arrays von Eingängen, Arrays von Ausgängen, Arrays von Merkern, Arrays von Lokaldaten und Arrays von Lokaldaten des Vorgänger-Bausteins sind nicht erlaubt.
0	1	B000F011		Der Bit-Offset muss Null sein.
0	1	B0010001		Falsches Datenpaket in UA_Connect
0	1	B0010002		Ungültige URL eines Server-Endpunkts
0	1	B0010003		Ungültiger Wert für Timeout
0	1	B0010004		Ungültiger Wert für SessionTimeout
0	1	B0010005		Ungültiger Wert für MonitorConnection
0	1	B0010006		Ungültiges Transportprofil
0	1	B0010007		Ungültige Server-URI
0	1	B0010008		Ungültiges Security-Verfahren
0	1	B0010009		Ungültiges Security-Profil
0	1	B001000A		Ungültiger Wert für UserIdentityToken
0	1	B001000B		Ungültiger Applikationsname
0	1	B0010101		Ungültiges Format der Server-Endpunkt-URL. Es muss mit "opc.tcp://" beginnen.
0	1	B0010102		Der Pfad der Server-Endpunkt-URL wird nicht unterstützt.
0	1	B0010103		Der Client darf sich nicht mit dem Server des eigenen CP verbinden.
0	1	B0010104		Ungültiger Port
0	1	B0010105		Ungültige Adresse. DNS wird nicht unterstützt.
0	1	B0020001		Falsches Datenpaket in UA_NameSpaceGetIndex
0	1	B0030001		Falsches Datenpaket in UA_NodeGetHandleList
0	1	B0030002		Ungültiger NodeID-Identifier
0	1	B0030003		Überschreitung der max. Anzahl an Items (64000)
0	1	B0030004		Nicht unterstütztes Attribut "ValueRank" einer Variable. Die ValueRank-Werte 1 (eindimensionales Array) und -1 (skalar) werden unterstützt.
0	1	B0040001		Falsches Datenpaket in UA_NodeReleaseHandleList
0	1	B0050001		Falsches Datenpaket in UA_Disconnect
0	1	B0050002		UA_Disconnect kann nicht aufgerufen werden, da keine Verbindung besteht.
0	1	B0060001		Falsches Datenpaket in UA_ReadList
0	1	B0060002		Die Länge der zu lesenden Daten überschreitet das Maximum von 64000.
0	1	B0060003		Ungültiger Bereich von IndexRangeCount
0	1	B0060004		Ungültiger IndexRange. StartIndex darf nicht größer sein als EndIndex und EndIndex darf nicht größer sein als die Array-Länge.
0	1	B0060005		Ungültige AttributeID
0	1	B0060006		Der gelesene Datenbereich ist zu klein. Korrigieren Sie die Angaben im UDT "UAIndexRange".
0	1	B0060007		Ungültige String-Länge. Der Wert (Nutzdaten) muss 254 sein.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	B0060008		Ungültiger Wert des Knotens. Der Wert kann nicht in S7-Variable eingelesen werden.
0	1	B0060010		Ungültiger Operandenbereich in einem Lese- oder Schreibauftrag: Arrays von Eingängen, Arrays von Ausgängen, Arrays von Merkern, Arrays von Lokaldaten und Arrays von Lokaldaten des Vorgänger-Bausteins sind nicht erlaubt.
0	1	B0070001		Falsches Datenpaket in UA_WriteList
0	1	B0070002		Die Länge der zu schreibenden Daten überschreitet das Maximum.
0	1	B0070003		Ungültiger Bereich von IndexRangeCount
0	1	B0070004		Ungültiger IndexRange. StartIndex darf nicht größer sein als EndIndex und EndIndex darf nicht größer sein als die Array-Länge.
0	1	B0070005		Ungültige AttributeID
0	1	B0070006		Der Datenbereich ist außerhalb des zugelassenen Bereichs.
0	1	B0070009		Ungültiges Zeichen im String
Fehlernummern nach IEC 61131-3 für OPC UA				
0	1	002D0000	GoodSubscriptionTransferred	The subscription was transferred to another session.
0	1	002E0000	GoodCompletesAsynchronously	The processing will complete asynchronously.
0	1	002F0000	GoodOverload	Sampling has slowed down due to resource limitations.
0	1	00300000	GoodClamped	The value written was accepted but was clamped.
0	1	00960000	GoodLocalOverride	The value has been overridden.
0	1	00A20000	GoodEntryInserted	The data or event was successfully inserted into the historical database.
0	1	00A30000	GoodEntryReplaced	The data or event field was successfully replaced in the historical database.
0	1	00A50000	GoodNoData	No data exists for the requested time range or event filter.
0	1	00A60000	GoodMoreData	The data or event field was successfully replaced in the historical database.
0	1	00A70000	GoodCommunicationEvent	The communication layer has raised an event.
0	1	00A80000	GoodShutdownEvent	The system is shutting down.
0	1	00A90000	GoodCallAgain	The operation is not finished and needs to be called again.
0	1	00AA0000	GoodNonCriticalTimeout	A non-critical timeout occurred.
0	1	00BA0000	GoodResultsMayBeIncomplete	The server should have followed a reference to a node in a remote server but did not. The result set may be incomplete.
0	1	406C0000	UncertainReferenceOutOfServer	One of the references to follow in the relative path references to a node in the address space in another server.
0	1	408F0000	UncertainNoCommunicationLastUsableValue	Communication to the data source has failed. The variable value is the last value that had a good quality.
0	1	40900000	UncertainLastUsableValue	Whatever was updating this value has stopped doing so.
0	1	40910000	UncertainSubstituteValue	The value is an operational value that was manually overwritten.
0	1	40920000	UncertainInitialValue	The value is an initial value for a variable that normally receives its value from another variable.
0	1	40930000	UncertainSensorNotAccurate	The value is at one of the sensor limits.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	40940000	UncertainEngineeringUnits Exceeded	The value is outside of the range of values defined for this parameter.
0	1	40950000	UncertainSubNormal	The value is derived from multiple sources and has less than the required number of Good sources.
0	1	40A40000	UncertainDataSubNormal	The value is derived from multiple values and has less than the required number of Good values.
0	1	40BC0000	UncertainReferenceNot Deleted	The server was not able to delete all target references.
0	1	40C00000	UncertainNotAllNodes Available	The list of references may not be complete because the underlying system is not available.
0	1	80010000	BadUnexpectedError	An unexpected error occurred
0	1	80020000	BadInternalError	An internal error occurred as a result of a programming or configuration error.
0	1	80030000	BadOutOfMemory	Not enough memory to complete the operation.
0	1	80040000	BadResourceUnavailable	An operating system resource is not available
0	1	80050000	BadCommunicationError	A low level communication error occurred.
0	1	80060000	BadEncodingError	Encoding halted because of invalid data in the objects being serialized.
0	1	80070000	BadDecodingError	Decoding halted because of invalid data in the stream.
0	1	80080000	BadEncodingLimitsExceeded	The message encoding/decoding limits imposed by the stack have been exceeded.
0	1	80B80000	BadRequestTooLarge	The resources (memory) of the CPU are temporarily occupied
0	1	80B90000	BadResponseTooLarge	The response message size exceeds limits set by the client.
0	1	80B90000	BadUnknownResponse	An unrecognized response was received from the server.
0	1	800A0000	BadTimeout	The operation timed out.
0	1	800B0000	BadServiceUnsupported	The server does not support the requested service.
0	1	800C0000	BadShutdown	The operation was cancelled because the application is shutting down.
0	1	800D0000	BadServerNotConnected	The operation could not complete because the client is not connected to the server.
0	1	800E0000	BadServerHalted	The server has stopped and cannot process any requests.
0	1	800F0000	BadNothingToDo	There was nothing to do because the client passed a list of operations with no elements.
0	1	80100000	BadTooManyOperations	The request could not be processed because it specified too many operations.
0	1	80110000	BadDataTypeIdUnknown	The extension object cannot be (de)serialized because the data type id is not recognized.
0	1	80120000	BadCertificateInvalid	The certificate provided as a parameter is not valid.
0	1	80130000	BadSecurityChecksFailed	The certificate provided as a parameter is not valid.
0	1	80140000	BadCertificateTimeInvalid	The Certificate has expired or is not yet valid.
0	1	80150000	BadCertificateIssuer TimeInvalid	An Issuer Certificate has expired or is not yet valid.
0	1	80160000	BadCertificateHost NameInvalid	The HostName used to connect to a Server does not match a HostName in the Certificate.
0	1	80170000	BadCertificateUriInvalid	The URI specified in the Application Description does not match the URI in the Certificate.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80180000	BadCertificateUseNotAllowed	The Certificate may not be used for the requested operation.
0	1	80190000	BadCertificateIssuerUseNotAllowed	The Issuer Certificate may not be used for the requested operation.
0	1	801A0000	BadCertificateUntrusted	The Certificate is not trusted.
0	1	801B0000	BadCertificateRevocationUnknown	It was not possible to determine if the Certificate has been revoked.
0	1	801C0000	BadCertificateIssuerRevocationUnknown	It was not possible to determine if the Issuer Certificate has been revoked.
0	1	801D0000	BadCertificateRevoked	The Certificate has been revoked.
0	1	801E0000	BadCertificateIssuerRevoked	The Issuer Certificate has been revoked.
0	1	801F0000	BadUserAccessDenied	User does not have permission to perform the requested operation.
0	1	80200000	BadIdentityTokenInvalid	The user identity token is not valid.
0	1	80210000	BadIdentityTokenRejected	The user identity token is valid but the server has rejected it.
0	1	80220000	BadSecureChannelIdInvalid	The specified secure channel is no longer valid.
0	1	80230000	BadInvalidTimestamp	The timestamp is outside the range allowed by the server.
0	1	80240000	BadNonceInvalid	The nonce does appear to be not a random value or it is not the correct length.
0	1	80250000	BadSessionIdInvalid	The session id is not valid.
0	1	80260000	BadSessionClosed	The session was closed by the client.
0	1	80270000	BadSessionNotActivated	The session cannot be used because ActivateSession has not been called.
0	1	80280000	BadSubscriptionIdInvalid	The subscription id is not valid.
0	1	802A0000	BadRequestHeaderInvalid	The header for the request is missing or invalid.
0	1	802B0000	BadTimestampsToReturnInvalid	The timestamps to return parameter is invalid.
0	1	802C0000	BadRequestCancelledByClient	The request was cancelled by the client.
0	1	80310000	BadNoCommunication	Communication with the data source is defined, but not established, and there is no last known value available.
0	1	80320000	BadWaitingForInitialData	Waiting for the server to obtain values from the underlying data source.
0	1	80330000	BadNodeIdInvalid	The syntax of the node id is not valid.
0	1	80340000	BadNodeIdUnknown	The node id refers to a node that does not exist in the server address space.
0	1	80350000	BadAttributeIdInvalid	The attribute is not supported for the specified Node.
0	1	80360000	BadIndexRangeInvalid	The syntax of the index range parameter is invalid.
0	1	80370000	BadIndexRangeNoData	No data exists within the range of indexes specified.
0	1	80380000	BadDataEncodingInvalid	The data encoding is invalid.
0	1	80390000	BadDataEncodingUnsupported	The server does not support the requested data encoding for the node.
0	1	803A0000	BadNotReadable	The access level does not allow reading or subscribing to the Node.
0	1	803B0000	BadNotWritable	The access level does not allow writing to the Node.
0	1	803C0000	BadOutOfRange	The value was out of range.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	803D0000	BadNotSupported	The requested operation is not supported.
0	1	803E0000	BadNotFound	A requested item was not found or a search operation ended without success.
0	1	803F0000	BadObjectDeleted	The object cannot be used because it has been deleted.
0	1	80400000	BadNotImplemented	Requested operation is not implemented.
0	1	80410000	BadMonitoringModeInvalid	The monitoring mode is invalid.
0	1	80420000	BadMonitoredItemIdInvalid	The monitoring item id does not refer to a valid monitored item.
0	1	80430000	BadMonitoredItem FilterInvalid	The monitored item filter parameter is not valid.
0	1	80440000	BadMonitoredItem FilterUnsupported	The server does not support the requested monitored item filter.
0	1	80450000	BadFilterNotAllowed	A monitoring filter cannot be used in combination with the attribute specified.
0	1	80460000	BadStructureMissing	A mandatory structured parameter was missing or null.
0	1	80470000	BadEventFilterInvalid	The event filter is not valid.
0	1	80480000	BadContentFilterInvalid	The content filter is not valid.
0	1	80490000	BadFilterOperandInvalid	The operand used in a content filter is not valid.
0	1	804A0000	BadContinuation PointInvalid	The continuation point provide is longer valid.
0	1	804B0000	BadNoContinuationPoints	The operation could not be processed because all continuation points have been allocated.
0	1	804C0000	BadReferenceTypeIdInvalid	The operation could not be processed because all continuation points have been allocated.
0	1	804D0000	BadBrowseDirectionInvalid	The browse direction is not valid.
0	1	804E0000	BadNodeNotInView	The node is not part of the view.
0	1	804F0000	BadServerUriInvalid	The ServerUri is not a valid URI.
0	1	80500000	BadServerNameMissing	No ServerName was specified
0	1	80510000	BadDiscoveryUrlMissing	No DiscoveryUrl was specified.
0	1	80520000	BadSempahoreFileMissing	The semaphore file specified by the client is not valid.
0	1	80530000	BadRequestTypeInvalid	The security token request type is not valid.
0	1	80540000	BadSecurityModeRejected	The security mode does not meet the requirements set by the Server.
0	1	80550000	BadSecurityPolicyRejected	The security policy does not meet the requirements set by the Server.
0	1	80560000	BadTooManySessions	The server has reached its maximum number of sessions.
0	1	80570000	BadUserSignatureInvalid	The user token signature is missing or invalid.
0	1	80580000	BadApplicationSignature Invalid	The signature generated with the client certificate is missing or invalid.
0	1	80590000	BadNoValidCertificates	The client did not provide at least one software certificate that is valid and meets the profile requirements for the server.
0	1	805A0000	BadRequestCancelled ByRequest	The request was cancelled by the client with the Cancel service.
0	1	805B0000	BadParentNodeIdInvalid	The parent node id does not to refer to a valid node.
0	1	805C0000	BadReferenceNotAllowed	The reference could not be created because it violates constraints imposed by the data model.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	805D0000	BadNodeIdRejected	The requested node id was reject because it was either invalid or server does not allow node ids to be specified by the client.
0	1	805F0000	BadNodeClassInvalid	The node class is not valid.
0	1	80600000	BadBrowseNameInvalid	The browse name is invalid.
0	1	80610000	BadBrowseNameDuplicated	The browse name is not unique among nodes that share the same relationship with the parent.
0	1	80620000	BadNodeAttributesInvalid	The node attributes are not valid for the node class.
0	1	80630000	BadTypeDefinitionInvalid	The type definition node id does not reference an appropriate type node.
0	1	80640000	BadSourceNodeIdInvalid	The source node id does not reference a valid node.
0	1	80650000	BadTargetNodeIdInvalid	The target node id does not reference a valid node.
0	1	80660000	BadDuplicateReference NotAllowed	The reference type between the nodes is already defined.
0	1	80670000	BadInvalidSelfReference	The server does not allow this type of selfreference on this node.
0	1	80680000	BadReferenceLocalOnly	The reference type is not valid for a reference to a remote server.
0	1	80690000	BadNoDeleteRights	The server will not allow the node to be deleted.
0	1	806A0000	BadServerIndexInvalid	The server index is not valid.
0	1	806B0000	BadViewIdUnknown	The view id does not refer to a valid view node.
0	1	806D0000	BadTooManyMatches	The requested operation has too many matches to return.
0	1	806E0000	BadQueryTooComplex	The requested operation requires too many resources in the server.
0	1	806F0000	BadNoMatch	The requested operation has no match to return.
0	1	80700000	BadMaxAgeInvalid	The max age parameter is invalid.
0	1	80710000	BadHistoryOperationInvalid	The history details parameter is not valid.
0	1	80720000	BadHistoryOperation Unsupported	The server does not support the requested operation.
0	1	80730000	BadWriteNotSupported	The server not does support writing the combination of value, status and timestamps provided.
0	1	80740000	BadTypeMismatch	The value supplied for the attribute is not of the same type as the attribute's value.
0	1	80750000	BadMethodInvalid	The method id does not refer to a method for the specified object.
0	1	80760000	BadArgumentsMissing	The client did not specify all of the input arguments for the method.
0	1	80770000	BadTooManySubscriptions	The server has reached its maximum number of subscriptions.
0	1	80780000	BadTooManyPublish Requests	The server has reached the maximum number of queued publish requests.
0	1	80790000	BadNoSubscription	There is no subscription available for this session.
0	1	807A0000	BadSequenceNumber Unknown	The sequence number is unknown to the server.
0	1	807B0000	BadMessageNotAvailable	The requested notification message is no longer available.
0	1	807C0000	BadInsufficientClientProfile	The Client of the current Session does not support one or more Profiles that are necessary for the Subscription.
0	1	80BF0000	BadStateNotActive	The sub-state machine is not currently active.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	807D0000	BadTcpServerTooBusy	The server cannot process the request because it is too busy.
0	1	807E0000	BadTcpMessageTypeInvalid	The type of the message specified in the header invalid.
0	1	807F0000	BadTcpSecureChannel Unknown	The SecureChannelId and/or TokenId are not currently in use.
0	1	80800000	BadTcpMessageTooLarge	The size of the message specified in the header is too large.
0	1	80810000	BadTcpNotEnough Resources	There are not enough resources to process the request.
0	1	80820000	BadTcpInternalError	An internal error occurred.
0	1	80830000	BadTcpEndpointUrlInvalid	The Server does not recognize the QueryString specified.
0	1	80840000	BadRequestInterrupted	The request could not be sent because of a network interruption.
0	1	80850000	BadRequestTimeout	Timeout occurred while processing the request.
0	1	80860000	BadSecureChannelClosed	The secure channel has been closed.
0	1	80870000	BadSecureChannelToken Unknown	The token has expired or is not recognized.
0	1	80880000	BadSequenceNumberInvalid	The sequence number is not valid.
0	1	80890000	BadConfigurationError	There is a problem with the configuration that affects the usefulness of the value.
0	1	808A0000	BadNotConnected	The variable should receive its value from another variable, but has never been configured to do so.
0	1	808B0000	BadDeviceFailure	There has been a failure in the device/data source that generates the value that has affected the value.
0	1	808C0000	BadSensorFailure	There has been a failure in the sensor from which the value is derived by the device/data source.
0	1	808D0000	BadOutOfService	The source of the data is not operational.
0	1	808E0000	BadDeadbandFilterInvalid	The dead band filter is not valid.
0	1	80970000	BadRefreshInProgress	This Condition refresh failed, a Condition refresh operation is already in progress.
0	1	80980000	BadConditionAlreadyDisabled	This condition has already been disabled.
0	1	80990000	BadConditionDisabled	Property not available, this condition is disabled.
0	1	809A0000	BadEventIdUnknown	The specified event id is not recognized.
0	1	809B0000	BadNoData	No data exists for the requested time range or event filter.
0	1	809D0000	BadDataLost	Data is missing due to collection started/stopped/lost.
0	1	809E0000	BadDataUnavailable	Expected data is unavailable for the requested time range due to an unmounted volume an off-line archive or tape or similar reason for temporary unavailability.
0	1	809F0000	BadEntryExists	The data or event was not successfully inserted because a matching entry exists.
0	1	80A00000	BadNoEntryExists	The data or event was not successfully updated because no matching entry exists.
0	1	80A10000	BadTimestampNotSupported	The client requested history using a timestamp format the server does not support (i. e. requested ServerTimestamp when server only supports SourceTimestamp).
0	1	80AB0000	BadInvalidArgument	One or more arguments are invalid.
0	1	80AC0000	BadConnectionRejected	Could not establish a network connection to remote server.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80AD0000	BadDisconnect	The server has disconnected from the client.
0	1	80AE0000	BadConnectionClosed	The network connection has been closed.
0	1	80AF0000	BadInvalidState	The operation cannot be completed because the object is closed uninitialized or in some other invalid state.
0	1	80B00000	BadEndOfStream	Cannot move beyond end of the stream.
0	1	80B10000	BadNoDataAvailable	No data is currently available for reading from a non-blocking stream.
0	1	80B20000	BadWaitingForResponse	The asynchronous operation is waiting for a response.
0	1	80B30000	BadOperationAbandoned	The asynchronous operation was abandoned by the caller.
0	1	80B40000	BadExpectedStreamToBlock	The stream did not return all data requested (possibly because it is a non-blocking stream).
0	1	80B50000	BadWouldBlock	Non-blocking behavior is required and the operation would block.
0	1	80B60000	BadSyntaxError	A value had an invalid syntax.
0	1	80B70000	BadMaxConnections Reached	The operation could not be finished because all available connections are in use.
0	1	80BB0000	BadEventNot Acknowledgeable	The event cannot be acknowledged.
0	1	80BD0000	BadInvalidTimestamp Argument	The defined timestamp to return was invalid.
0	1	80BE0000	BadProtocolVersion Unsupported	The applications do not have compatible protocol versions.
0	1	80C10000	BadFilterOperatorInvalid	An unrecognized operator was provided in a filter.
0	1	80C20000	BadFilterOperator Unsupported	A valid operator was provided, but the server does not provide support for this filter operator.
0	1	80C30000	BadFilterOperandCount Mismatch	The number of operands provided for the filter operator was less than expected for the operand provided.
0	1	80C40000	BadFilterElementInvalid	The referenced element is not a valid element in the content filter.
0	1	80C50000	BadFilterLiteralInvalid	The referenced literal is not a valid value.
0	1	80C90000	BadViewTimestampInvalid	The view timestamp is not available or not supported.
0	1	80CA0000	BadViewParameterMismatch	The view parameters are not consistent with each other.
0	1	80CB0000	BadViewVersionInvalid	The view version is not available or not supported.
0	1	80CC0000	BadConditionAlready Enabled	This condition has already been enabled.
0	1	80CD0000	BadDialogNotActive	The dialog condition is not active.
0	1	80CF0000	BadConditionBranch AlreadyAcked	The condition branch has already been acknowledged.
0	1	80D00000	BadConditionBranch AlreadyConfirmed	The condition branch has already been confirmed.
0	1	80D10000	BadConditionAlreadyShelved	The condition has already been shelved.
0	1	80D20000	BadConditionNotShelved	The condition is not currently shelved.
0	1	80D30000	BadShelvingTimeOutOfRange	The shelving time not within an acceptable range.
0	1	80D40000	BadAggregateListMismatch	The requested number of Aggregates does not match the requested number of NodeIds.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80D50000	BadAggregateNotSupported	The requested Aggregate is not support by the server.
0	1	80D60000	BadAggregateInvalidInputs	The aggregate value could not be derived due to invalid data inputs.
0	1	80DB0000	BadTooManyMonitoredItems	The request could not be processed because there are too many monitored items in the subscription.
0	1	80D70000	BadBoundNotFound	No data found to provide upper or lower bound value.
0	1	80D80000	BadBoundNotSupported	The server cannot retrieve a bound for the variable.
0	1	00D90000	GoodDataIgnored	The request specifies fields which are not valid for the Event-Type or cannot be saved by the historian.
0	1	80DA0000	BadAggregateConfiguration Rejected	The aggregate configuration is not valid for specified node.
Fehlernummern von PLCOpen für OPC UA				
0	1	A0000001	PLCopenUA_Bad_FW_ PermanentError	Internal, permanent error.
0	1	A0000002	PLCopenUA_Bad_FW_ TempError	Temp. Error; FB could retry to reach FW.
0	1	A0000003	PLCopenUA_Bad_ ConnectionError	Connection could not be established.
0	1	A0000004	PLCopenUA_Bad_ HostNotFound	The requested hostname could not be found.
0	1	A0000005	PLCopenUA_Bad_ AlreadyConnected	Connection was already established.
0	1	A0000006	PLCopenUA_Bad_ SecurityFailed	Connection failed due to security setup.
0	1	A0000007	PLCopenUA_Bad_ Suspended	Connection is suspended.
0	1	A0000008	PLCopenUA_Bad_ ConnectionInvalidHdl	Provided ConnectionHdl is not known.
0	1	A0000009	PLCopenUA_Bad_ NSNotFound	A namespace with the requested name cannot be found on server.
0	1	A000000A	PLCopenUA_Bad_ ResultTooLong	Target PLC variable is too short for retrieved data.
0	1	A000000B	PLCopenUA_Bad_ InvalidType	Invalid or unsupported Type.
0	1	A000000C	PLCopenUA_Bad_ NodeInvalidHdl	Provided NodeHdl is not known.
0	1	A000000D	PLCopenUA_Bad_ MethodInvalidHdl	Provided MethodHdl is not known.
0	1	A000000E	PLCopenUA_Bad_ ReadFailed	Read failed for unknown reason.
0	1	A000000F	PLCopenUA_Bad_ WriteFailed	Write failed for unknown reason.
0	1	A0000010	PLCopenUA_Bad_ CallFailed	Method Call failed for unknown reason.
0	1	A0000011	PLCopenUA_Bad_ InParamFailed	Method Call Input parameter conversion failed.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	A0000012	PLCopenUA_Bad_ OutParamFailed	Method Call Output parameter conversion failed. ATTENTI-ON: this means the MethodCall was executed successfully but the returned values could not be converted.
0	1	A0000013	PLCopenUA_Bad_ SubscriptionInvalidHdl	Provided SubscriptionHdl is not known.
0	1	A0000014	PLCopenUA_Bad_ MonitoredItemInvalidHdl	Provided MonitoredItemHdl is not known.

Punkt-zu-Punkt

Überblick Freeport-Kommunikation

STEP 7 bietet erweiterte Anweisungen, mit denen die Freeport-Kommunikation über ein im Anwenderprogramm vorgegebenes Protokoll realisiert werden kann. Diese Anweisungen lassen sich in zwei Kategorien unterteilen:

- Konfigurationsanweisungen
- Kommunikationsanweisungen

Hinweis

CPU-Mengengerüste

Die Freeport- Anweisungen kommunizieren mit den Kommunikationsmodulen durch Lesen oder Schreiben von Datensätzen.
Beim Einsatz der Anweisungen ist daher das Mengengerüst der jeweiligen CPU für das Lesen und Schreiben von Datensätzen zu beachten.
Wenn mehrere Anweisungen auf einer CPU gleichzeitig Datensätze lesen oder schreiben sollen, so müssen diese Anweisungen evtl. durch das Anwenderprogramm zeitlich versetzt aufgerufen werden.

Konfigurationsanweisungen

Bevor das Anwenderprogramm die Freeport-Kommunikation starten kann, müssen die Kommunikationsschnittstelle und die Parameter zum Senden und Empfangen der Daten konfiguriert werden.

Die Schnittstellenkonfiguration und die Datenkonfiguration können für jedes CM in der Gerätekonfiguration oder mit den folgenden Anweisungen Ihres Anwenderprogramms durchgeführt werden:

- Port_Config (Seite 4101)
- Send_Config (Seite 4104)
- Receive_Config (Seite 4106)
- P3964_Config (Seite 4111)

ACHTUNG
Gerätekonfiguration <-> Konfigurationsanweisungen Die Parameter der Gerätekonfiguration werden bei jedem Power On der CPU (Spannungswiederkehr) zum CM übertragen. Die Parameter der Konfigurationsanweisungen werden entsprechend Ihrem Anwenderprogramm zum CM übertragen. Es gibt keinen Abgleich zwischen den Parametern der Gerätekonfiguration und den Parametern der Konfigurationsanweisungen, d. h. die Parameter der Konfigurationsanweisungen werden nicht in die Gerätekonfiguration der CPU übernommen. Sie haben mit ihrem Anwenderprogramm die Kontrolle darüber, wann welche Parameter im CM wirksam sind.

Kommunikationsanweisungen

Mit den Anweisungen für die Freeport-Kommunikation kann das Anwenderprogramm Daten an die Kommunikationsschnittstellen senden und von diesen Daten empfangen. Die CMs senden und empfangen Daten an die bzw. von den Kommunikationsteilnehmern.

- Send_P2P (Seite 4113)
- Receive_P2P (Seite 4117)

Hinweis

Datenkonsistenz

- Falls die zu sendenden Daten konsistent übertragen werden sollen, dürfen diese nach dem Flankenanstieg des Parameters REQ bis zum Setzen von DONE durch die Anweisung Send_P2P nicht verändert werden.
- Falls die Empfangsdaten konsistent ausgelesen werden sollen, dürfen diese nur in dem Zyklus ausgewertet werden, in dem NDR = TRUE ist.

Mit zusätzlichen Anweisungen kann der Empfangspuffer zurückgesetzt und es können spezielle RS232-Signale abgefragt und gesetzt werden.

- Receive_Reset (Seite 4119)
- Signal_Get (Seite 4120)
- Signal_Set (Seite 4121)

Mit den folgenden Anweisungen können erweiterte Funktionen, sofern vom Modul unterstützt, gelesen bzw. geschrieben werden.

- Get_Features (Seite 4123)
- Set_Features (Seite 4124)

Alle Freeport-Anweisungen arbeiten asynchron. Die Anweisungen müssen deshalb solange aufgerufen werden, bis durch den Ausgangsparameter DONE die vollständige Ausführung angezeigt wird.

Mit Hilfe einer Abfragearchitektur kann das Anwenderprogramm den Sende- und Empfangsstatus feststellen. Send_P2P und Receive_P2P können gleichzeitig ausgeführt werden. Die Kommunikationsmodule puffern die Sende- und Empfangsdaten je nach Bedarf bis zu einer modulspezifischen maximalen Puffergröße.

Hinweis

Auflösung von Bitzeiten

Für verschiedene Parameter wird die Anzahl der Bitzeiten bei der konfigurierten Datenübertragungsrate angegeben. Durch Angabe des Parameters in Bitzeiten ist der Parameter unabhängig von der Datenübertragungsrate. Alle Parameter mit der Einheit Bitzeiten können mit einer maximalen Anzahl von 65535 angegeben werden.

Anwendung der Anweisungen

Die Freeport-Anweisungen müssen zyklisch aufgerufen werden, um nach empfangenen Daten oder auf das Übertragungsende eines Sendevorgangs abzufragen.

Abhängig von der Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.
Wurde der Auftrag mit DONE = TRUE bzw. NDR = TRUE durchlaufen, wurde der Auftrag fehlerfrei ausgeführt.

Hinweis

STATUS sichern

Die Parameter DONE, NDR, ERROR und STATUS stehen nur für einen Bausteindurchlauf an. Zur Anzeige des STATUS sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.

Master

Typische Sequenz für einen Master:

1. Die Anweisung Send_P2P veranlasst eine Übertragung zum CM.
   Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen.
2. Die Anweisung Send_P2P wird in aufeinander folgenden Zyklen ausgeführt, um den Status des Übertragungsvorgangs abzufragen.
3. Wenn die Anweisung Send_P2P am Ausgang DONE meldet, dass die Übertragung beendet ist, kann der Anwendercode den Empfang der Antwort vorbereiten.
4. Die Anweisung Receive_P2P wird wiederholt ausgeführt, um auf eine Antwort abzufragen. Wenn das CM Antwortdaten erfasst hat, kopiert die Anweisung Receive_P2P die Antwort in die CPU und meldet am Ausgang NDR, dass neue Daten empfangen wurden.
5. Das Anwenderprogramm kann die Antwort verarbeiten.
6. Zurück zu Schritt 1 und Wiederholung der Sequenz.

Slave

Typische Sequenz für einen Slave:

1. Das Anwenderprogramm führt die Anweisung Receive_P2P in jedem Zyklus aus.
2. Wenn das CM eine Anforderung empfangen hat, meldet die Anweisung Receive_P2P am Ausgang NDR, dass neue Daten bereit sind, und die Anforderung wird in die CPU kopiert.
3. Das Anwenderprogramm verarbeitet die Anforderung und erzeugt eine Antwort.
4. Mit der Anweisung Send_P2P wird die Antwort an den Master zurückgesendet.
5. Die Anweisung Send_P2P muss wiederholt ausgeführt werden, um sicherzustellen, dass der Sendevorgang stattfindet.
6. Zurück zu Schritt 1 und Wiederholung der Sequenz.

Der Slave muss dafür sorgen, dass Receive_P2P entsprechend oft aufgerufen wird, damit eine Übertragung vom Master empfangen werden kann, bevor dieser beim Warten auf eine Antwort wegen Zeitüberschreitung den Vorgang abbricht. Um diese Aufgabe zu erfüllen, kann das Anwenderprogramm Receive_P2P aus einem Zyklus-OB heraus aufrufen, dessen Zykluszeit hinreichend kurz ist, um eine Übertragung vom Master vor dem Ablauf der Timeout-Einstellung zu empfangen. Wird die Zykluszeit für den OB so eingestellt, dass zwei Ausführungen innerhalb der Timeout-Einstellung des Masters erfolgen, kann das Anwenderprogramm alle Übertragungen ohne Verlust empfangen.

Gemeinsame Parameter für Freeport-Operationen

Tabelle 4-144 Gemeinsame Eingangsparameter der Freeport-Anweisungen

Parameter	Beschreibung
REQ	Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Eine weitere Flanke am REQ darf erst nach Auftragsabschluss (DONE bzw. ERROR) erzeugt werden. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen. Wenn Sie eine Freeport-Anweisung in Ihr Programm einfügen, werden Sie von STEP 7 aufgefordert, den Instanz-DB anzugeben (oder von STEP 7 einen entsprechenden Instanz-DB anlegen zu lassen). Verwenden Sie für jeden PtP-Anweisungsaufruf einen eindeutigen DB.
PORT	Eine Portadresse wird während der Konfiguration des Kommunikationsmoduls vergeben. Mit dem Parameter PORT wird der Anweisung die Zuordnung zu einem bestimmten Kommunikationsmodul mitgeteilt. Nach der Konfiguration kann ein symbolischer Name für den Standardport ausgewählt werden. Der zugewiesene CM-Portwert ist die Eigenschaft "HW-Kennung" der Gerätekonfiguration bei S7-1200/1500 bzw. die "Eingangsadresse" bei S7-300/400. Der symbolische Name des Ports wird in der Symboltabelle zugewiesen.

Die Ausgangsparameter DONE, NDR, ERROR und STATUS der Freeport-Anweisungen zeigen den Ausführungsstatus der Freeport-Funktionen an.

Tabelle 4-145 Ausgangsparameter DONE, NDR, ERROR und STATUS

Parameter	Datentyp	Standard	Beschreibung
DONE	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass die letzte Anforderung fehlerfrei abgeschlossen wurde; andernfalls FALSE.
NDR	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass neue Daten empfangen wurden; andernfalls FALSE.
ERROR	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass die letzte Anforderung mit Fehler abgeschlossen wurde, der entsprechende Fehlercode befindet sich in STATUS; andernfalls FALSE.
STATUS	Word	16#0000 oder 16#7000	Ergebniszustand: • Ist das Bit DONE oder NDR gesetzt, wird STATUS auf 0/16#7000 oder auf einen spezifischen Fehlercode gesetzt. • Ist das Bit ERROR gesetzt, wird am STATUS ein Fehlercode angezeigt. • Wird keines der oben aufgeführten Bits gesetzt, kann die Anweisung Statusergebnisse zurückgeben, die den aktuellen Zustand der Funktion beschreiben. Der Wert in STATUS ist bis zum nächsten Aufruf dieser Anweisung (mit ein und derselben Portadresse) gültig.

Tabelle 4-146 Durchgangsparameter COM_RST

Parameter	Datentyp	Standard	Beschreibung
COM_RST	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST wieder auf FALSE gesetzt. Hinweis: Sie müssen COM_RST im Anlauf auf TRUE setzen und sollten den Parameter danach nicht mehr ändern, d. h. beim Aufruf der Anweisung keinen Wert zuweisen. Nach Initialisierung des Instanz-DBs wird COM_RST von der Anweisung zurückgesetzt.

Hinweis

Beachten Sie, dass die Parameter DONE, NDR, ERROR und STATUS nur einen Zyklus lang gesetzt sind.

Tabelle 4-147 Gemeinsame Fehlercodes

Fehlercode	Beschreibung
16#0000	Kein Fehler
16#7000	Funktion nicht aktiv
16#7001	Erstaufruf nach dem Starten einer Anforderung.

Fehlercode	Beschreibung
16#7002	Folgeaufruf nach dem Starten einer Anforderung.
16#8x3A	Unzulässiger Pointer in Parameter x

Tabelle 4-148 Gemeinsame Fehlerklassen des Parameters STATUS

Beschreibung der Klasse	Fehlerklassen	Beschreibung
Portkonfiguration	16#81Ax	Für die Beschreibung häufiger Fehler in der Schnittstellenkonfiguration
Sendekonfiguration	16#81Bx	Für die Beschreibung von Fehlern in der Sendekonfiguration
Empfangskonfiguration	16#81Cx	Für die Beschreibung von Fehlern in der Empfangskonfiguration
Senden	16#81Dx	Für die Beschreibung von Laufzeitfehlern beim Senden
Empfangen	16#81Ex	Für die Beschreibung von Laufzeitfehlern beim Empfangen
RS232-Begleitsignale	16#81Fx	Für die Beschreibung von Fehlern in Verbindung mit der Signalverarbeitung

Port_Config: PtP-Kommunikationsport projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Port_Config (Port-Konfiguration) können Sie Parameter wie die Datenübertragungsrate über Ihr Programm im laufenden Betrieb ändern. Die im CM anstehenden Daten werden mit der Ausführung von Port_Config gelöscht.

Die Konfigurationsänderungen von Port_Config werden nicht in der CPU sondern auf dem CM gespeichert. Bei Spannungswiederkehr wird das CM mit den in der Gerätekonfiguration gespeicherten Daten parametriert.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
PROTOCOL	IN	UInt	Word	0	Protokoll • 0 = Freeport-Protokoll • 1 = Protokoll 3964(R)
BAUD	IN	UInt	Word	6	Datenübertragungsrate des Ports: • 1 = 300 Bit/s • 2 = 600 Bit/s • 3 = 1200 Bit/s • 4 = 2400 Bit/s • 5 = 4800 Bit/s • 6 = 9600 Bit/s • 7 = 19200 Bit/s • 8 = 38400 Bit/s • 9 = 57600 Bit/s • 10 = 76800 Bit/s • 11 = 115200 Bit/s
PARITY	IN	UInt	Word	1	Parität des Ports: • 1 = Keine Parität • 2 = Gerade Parität • 3 = Ungerade Parität • 4 = Mark-Parität • 5 = Space-Parität • 6 = Beliebig
DATABITS	IN	UInt	Word	1	Bits pro Zeichen: • 1 = 8 Datenbits • 2 = 7 Datenbits

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
STOPBITS	IN	UInt	Word	1	Stoppbits: • 1 = 1 Stoppbit • 2 = 2 Stoppbits
FLOWCTRL	IN	UInt	Word	1	Flusskontrolle: • 1 = Keine Flusskontrolle • 2 = XON/XOFF • 3 = Hardware-RTS immer EIN • 4 = Hardware-RTS geschaltet • 5 = Hardware-RTS immer EIN, DTR/DSR ignorieren
XON-CHAR	IN	Char		16#0011	Gibt das Zeichen an, das als XON-Zeichen dient. Dies ist typischerweise ein DC1-Zeichen (11H). Dieser Parameter wird nur ausgewertet, wenn die Software-Flusskontrolle aktiviert ist.
XOFF-CHAR	IN	Char		16#0013	Gibt das Zeichen an, das als XOFF-Zeichen dient. Dies ist typischerweise ein DC3-Zeichen (13H). Dieser Parameter wird nur ausgewertet, wenn die Software-Flusskontrolle aktiviert ist.
WAITIME	IN	UInt	Word	2000	Gibt an, wie lange nach dem Empfang eines XOFF-Zeichens auf ein XON-Zeichen gewartet werden soll bzw. wie lange nach CTS = OFF auf CTS = ON gewartet werden soll (0 bis 65535 ms). Dieser Parameter wird nur ausgewertet, wenn die Flusskontrolle aktiviert ist.
MODE	IN	USInt	Byte	0	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master; CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave; CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1)]

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
LINE_PRE	IN	USInt	Byte	0	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) : Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V : Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.
BRK_DET	IN	USInt	Byte	0	Break-Erkennung Zulassig sind: • 0 = Break-Erkennung deaktiviert • 1 = Break-Erkennung aktiviert
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Ein Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Ein Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

[1)] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Send_Config: PtP-Sender projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Send_Config (Sendekonfiguration) können Sie im laufenden Betrieb Sendeparameter über Ihr Programm ändern (Bedingungen, die den Beginn und das Ende der zu sendenden Daten kennzeichnen). In einem CM anstehende Daten werden gelöscht, wenn Send_Config ausgeführt wird.

Die Konfigurationsänderungen von Send_Config werden nicht in der CPU sondern auf dem CM gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Aktiviert die Konfigurationsänderung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
RTSONDLY	IN	UInt	Word	0	Anzahl der Millisekunden, die nach Aktivierung von RTS gewartet werden soll, bevor eine Übertragung von Sende-Daten gestartet wird. Dieser Parameter ist nur gültig, wenn die Hardwareflusskontrolle aktiviert ist. Der gültige Bereich ist 0 bis 65535 ms. Der Wert 0 deaktiviert die Funktion.
RTSOFFDLY	IN	UInt	Word	0	Anzahl der Millisekunden, die nach der Übertragung von Sende-Daten gewartet werden soll, bevor RTS deaktiviert wird: Dieser Parameter ist nur gültig, wenn die Hardwareflusskontrolle aktiviert ist. Der gültige Bereich ist 0 bis 65535 ms. Der Wert 0 deaktiviert die Funktion.
BREAK	IN	UInt	Word	0	Dieser Parameter gibt an, dass vor dem Start jedes Telegramms für die angegebene Anzahl von Bitzeiten ein BREAK gesendet wird. Maximal sind 65535 Bitzeiten möglich. Der Wert 0 deaktiviert die Funktion.
IDLELINE	IN	UInt	Word	0	Dieser Parameter gibt an, dass vor dem Start jedes Telegramms für die angegebene Anzahl von Bitzeiten die Leitung im Leerlauf bleibt. Maximal sind 65535 Bitzeiten möglich. Der Wert 0 deaktiviert die Funktion.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
USR_END	IN	STRING[2]		0	Eingabe der Endezeichen. Es können maximal 2 Endezeichen projektiert werden. Es wird einschließlich der/des Endezeichen(s) gesendet, unabhängig von der parametrierten Telegrammlänge.
APP_END	IN	STRING[5]		0	Eingabe anzuhängende Zeichen. Es können maximal 5 Zeichen angehängt werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Receive_Config: PtP-Empfänger projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Receive_Config (Empfangskonfiguration) können Sie über Ihr Programm Empfangsparameter im laufenden Betreib ändern. Diese Anweisung konfiguriert die Bedingungen, die den Beginn und das Ende empfangener Daten kennzeichnen. In einem CM anstehende Daten werden gelöscht, wenn Receive_Config ausgeführt wird.

Die Konfigurationsänderungen von Receive_Config werden auf dem CM **nicht remanent** gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt. Die Anweisung Receive_Config muss deshalb aus dem Anwenderprogramm bei Spannungswiederkehr der CPU oder des Kommunikationsmoduls erneut aufgerufen werden, um die in der Gerätekonfiguration gespeicherten Parameter wieder zu überschreiben.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Aktiviert die Konfigurationsänderung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
RECEIVE_CONDITIONS	IN	Variant	Any	-	Die Datenstruktur von Receive_Conditions gibt die Anfangs- und Endbedingungen an, anhand derer Anfang und Ende eines Telegramms identifiziert werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Anfangsbedingungen für die Anweisung Receive_P2P

Die Anweisung Receive_P2P nutzt die in der Gerätekonfiguration oder von der Anweisung Receive_Config angegebene Konfiguration, um Anfang und Ende von Telegrammen der Freeport-Kommunikation zu ermitteln. Der Telegrammanfang wird von den Startbedingungen festgelegt. Der Telegrammanfang kann anhand von einer oder mehreren Startbedingungen ermittelt werden.
Ist sowohl Break als auch Idle Line aktiviert, so muss zunächst Break und dann zusätzlich Idle Line erfüllt sein. Danach genügt eine der weiteren Bedingungen (Startzeichen oder Anfangssequenz), um die Datenübertragung zu starten.
Die Startbedingung "Beliebiges Zeichen" kann nicht mit anderen Startbedingungen kombiniert werden.

Datentypstruktur des Parameters Receive_Conditions, Teil 1 (Startbedingungen)

Tabelle 4-149 Struktur von Receive_Conditions für Startbedingungen

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
START.START-COND	IN	Word	16#0002	Angabe der Startbedingung • 01H - Erkennung des Startzeichens • 02H - Beliebiges Zeichen • 04H - Erkennung eines Line Break • 08H - Erkennung einer Idle Line • 10H - Erkennung Anfangssequenz 1 • 20H - Erkennung Anfangssequenz 2 • 40H - Erkennung Anfangssequenz 3 • 80H - Erkennung Anfangssequenz 4 Die Startbedingungen können durch Addition der Werte kombiniert werden.
START.IDLE-TIME	IN	Word	16#0028	Die Anzahl der erforderlichen Bitzeiten im Leerlauf, damit ein neuer Telegrammanfang erkannt wird (Standardwert: W#16#28). Nur in Verbindung mit der Bedingung "Erkennung einer Idle Line". 0 bis FFFF
START.START-CHAR	IN	Byte	16#0002	Das Startzeichen für die Bedingung "Startzeichen". (Standardwert: B#16#2)
START.SEQ[1]. CTL	IN	Byte	0	Anfangssequenz 1, Vergleich für jedes Zeichen deaktivieren/aktivieren: (Standardwert: B#16#0) Dies sind die Aktivierungsbits für jedes Zeichen der Startzeichenfolge. • 01H - Zeichen 1 • 02H - Zeichen 2 • 04H - Zeichen 3 • 08H - Zeichen 4 • 10H - Zeichen 5 Wird ein Bit für ein bestimmtes Zeichen deaktiviert, bedeutet dies, dass an dieser Position der Zeichenfolge jedes Zeichen eine gültige Startzeichenfolge darstellt (z. B 1FH = alle 5 Zeichen werden interpretiert).
START.SEQ[1]. STR[1] .. START.SEQ[1]. STR.[5]	IN	Char[5]	0	Anfangssequenz 1, Startzeichen (5 Zeichen).
START.SEQ[2]. CTL	IN	Byte	0	Anfangssequenz 2, Vergleich für jedes Zeichen deaktivieren/aktivieren. Standardwert: B#16#0)
START.SEQ[2]. STR[1] .. START.SEQ[2]. STR.[5]	IN	Char[5]	0	Anfangssequenz 2, Startzeichen (5 Zeichen).
START.SEQ[3]. CTL	IN	Byte	0	Anfangssequenz 3, Vergleich für jedes Zeichen deaktivieren/aktivieren. Standardwert: B#16#0

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
START.SEQ[3]. STR[1] .. START.SEQ[3]. STR.[5]	IN	Char[5]	0	Anfangssequenz 3, Startzeichen (5 Zeichen).
START.SEQ[4]. CTL	IN	Byte	0	Anfangssequenz 4, Vergleich für jedes Zeichen deaktivieren/ aktivieren. Standardwert: B#16#0
START.SEQ[4]. STR[1] .. START.SEQ[4]. STR.[5]	IN	Char[5]	0	Anfangssequenz 4, Startzeichen (5 Zeichen),

Beispiel

Sehen Sie sich die folgende empfangene hexadezimal-codierten Daten an: "68 10 aa 68 bb 10 aa 16". Die konfigurierten Startzeichenfolgen finden Sie in der folgenden Tabelle. Startzeichenfolgen werden ausgewertet, nachdem das erste Zeichen 68H erfolgreich empfangen wurde. Nach erfolgreichem Empfang des vierten Zeichens (das zweite 68H) ist die Startbedingung 1 erfüllt. Wenn die Startbedingungen erfüllt sind, beginnt die Auswertung der Endebedingungen.

Die Verarbeitung der Startzeichenfolge kann aufgrund verschiedener Fehler bei Parität, Framing oder Zeitabständen zwischen den Zeichen beendet werden. Diese Fehler führen dazu, dass die Daten nicht empfangen werden, weil die Startbedingung nicht erfüllt wurde (es erfolgt eine Fehlermeldung).

Tabelle 4-150 Startbedingungen

Startbedingung	Erstes Zeichen	Erstes Zeichen +1	Erstes Zeichen +2	Erstes Zeichen +3	Erstes Zeichen +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Endebedingungen für die Anweisung Receive_P2P

Das Ende eines Telegramms wird durch das erste Auftreten einer oder mehrerer konfigurierter Endebedingungen festgelegt.

Sie können die Endebedingungen entweder in den Eigenschaften der Kommunikationsschnittstelle in der Gerätekonfiguration oder über die Anweisung Receive_Config konfigurieren. Nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls werden die Empfangsparameter (Start- und Endebedingungen) wieder auf die Einstellungen der Gerätekonfiguration gesetzt. Wenn das STEP 7 Anwenderprogramm Receive_Config ausführt, werden die Einstellungen auf die Parameter von Receive_Config gesetzt.

Datentypstruktur des Parameters Receive_Conditions, Teil 2 (Endebedingungen)

Tabelle 4-151 Struktur von Receive_Conditions für Endebedingungen

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
END.END-COND	IN	Word	0	Dieser Parameter gibt die Bedingung für das Telegrammende an: • 01H - Antwortzeit-Zeitüberschreitung • 02H - Nachrichten-Zeitüberschreitung • 04H - Zeichenverzugszeit • 08H - Maximale Telegrammlänge • 10H - Lese Nachrichtenlänge aus Nachricht (N+LEN+M) • 20H - Endesequenz • 40H - Feste Telegrammlänge
END.FIXLEN	IN	Word	1	Feste Telegrammlänge: Wird nur verwendet, falls die Endebedingung "Feste Telegrammlänge" ausgewählt ist. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.MAXLEN	IN	Word	1	Maximale Telegrammlänge: Wird nur verwendet, falls die Endebedingung "Maximale Telegrammlänge" ausgewählt ist. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.N	IN	Word	0	Byteposition des Längenfelds im Telegramm. Wird nur bei der Endebedingung N+LEN+M verwendet. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.LENGTH-SIZE	IN	Word	0	Größe des Längenfeldes (1, 2 oder 4 Byte). Wird nur bei der Endebedingung N+LEN+M verwendet.
END.LENGTHM	IN	Word	0	Anzahl der Zeichen nach dem Längenfeld, die nicht im Wert des Längenfelds enthalten sind. Diese Angabe wird nur bei der Endebedingung N+LEN+M verwendet. 0 bis 255 Bytes
END.RCVTIME	IN	Word	200	Geben Sie an, wie lange nach dem Senden eines Telegramms auf das erste empfangene Zeichen gewartet werden soll. Die Empfangsanweisung wird mit einer Fehlermeldung beendet, wenn nicht innerhalb der angegebenen Zeit ein Zeichen erfolgreich empfangen wird. Diese Angabe wird nur bei der Bedingung "Antwortzeit-Überschreitung" verwendet. (0 bis 65535 ms). Hinweis: Dieser Parameter kann nicht als alleiniges Endekriterium sondern nur in Verbindung mit mindestens einer weiteren Endbedingung eingesetzt werden.
END.MSGTIME	IN	Word	200	Geben Sie an, wie lange nach dem Empfang des ersten Zeichens auf den vollständigen Empfang des gesamten Telegramms gewartet werden soll. Dieser Parameter wird nur verwendet, wenn die Bedingung "Nachrichten-Zeitüberschreitung" ausgewählt ist. (0 bis 65535 ms)
END.CHARGAP	IN	Word	12	Geben Sie die max. Anzahl der Bitzeiten zwischen den Zeichen an. Falls die Anzahl der Bitzeiten zwischen den Zeichen den angegebenen Wert überschreitet, ist die Endebedingung erfüllt. Diese Angabe wird nur bei der Bedingung "Zeichenverzugszeit" verwendet. (0 bis 65535 Bitzeiten)

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
END.SEQ.CTL	IN	Byte	0	Zeichenfolge 1, Vergleich für jedes Zeichen deaktivieren/aktivieren: Dies sind die Aktivierungsbits für jedes Zeichen der Endezeichenfolge. Zeichen 1 ist Bit 0, Zeichen 2 ist Bit 1, ..., Zeichen 5 ist Bit 4. Wird ein Bit für ein bestimmtes Zeichen deaktiviert, bedeutet dies, dass an dieser Position der Zeichenfolge jedes Zeichen eine Übereinstimmung darstellt.
END.SEQ.STR[1] .. END.SEQ.STR[5]	IN	Char[5]	0	Zeichenfolge 1, Startzeichen (5 Zeichen)

Tabelle 4-152 Allgemeine Parameter der Anweisung Receive_P2P

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
GENERAL.MBUF_SIZE	IN	Byte	255	Eingabe der Anzahl der Telegramme, die im Empfangspuffer des CM gepuffert werden sollen. Sind keine weiteren Bedingungen aktiv, die das Verhalten des Empfangspuffers beeinflussen (Überschreiben verhindern, Datenflusskontrolle), so werden nach Erreichen des Grenzwerts weitere Telegramme verworfen. (1 bis 255 Telegramme)
GENERAL.OW_PROT	IN	Byte	0	Aktiviert das Nicht-Überschreiben des gepufferten Telegramms, wenn der CM ein neues Telegramm empfängt und der Empfangspuffer des CM noch nicht ausgelesen wurde. Damit wird verhindert, dass bereits gepufferte Empfangstelegramme evtl. verloren gehen. • 0 - nicht aktiviert • 1 - aktiviert
GENERAL.CLR_MBUF	IN	Byte	0	Aktiviert das Löschen des Empfangspuffers im Anlauf der CPU. Ist dieser Parameter aktiviert, wird der Empfangspuffer beim CPU-Übergang von STOP nach RUN automatisch gelöscht. Im Empfangspuffer befinden sich dann nur Telegramme die nach dem CPU-Anlauf empfangen wurden. • 0 - nicht aktiviert • 1 - aktiviert

P3964_Config: Protokoll 3964(R) projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung P3964_Config (Protokollkonfiguration) können Sie Protokollparameter für 3964(R) wie Zeichenverzugszeit, Priorität und Blockcheck über ihr Programm im laufenden Betrieb ändern.

Die Konfigurationsänderungen von P3964_Config werden nicht in der CPU sondern auf dem CM gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BCC	IN	USInt	Byte	1	Aktiviert/deaktiviert die Verwendung des Blockchecks • 0 = ohne Blockcheck • 1 = mit Blockcheck
Priority	IN	USInt	Byte	1	Einstellung der Priorität • 0 = niedrige Priorität • 1 = hohe Priorität
CharacterDelayTime	IN	UInt	Word	16#00DC	Einstellung der Zeichenverzugszeit (abhängig von der eingestellten Datenübertragungsgeschwindigkeit) (Standardwert: 220 ms) 1 bis 65535 ms
AcknDelayTime	IN	UInt	Word	16#07D0	Einstellung der Quittungsverzugszeit (abhängig von der eingestellten Datenübertragungsgeschwindigkeit) (Standardwert: 2000 ms) 1 bis 65535 ms
BuildupAttempts	IN	USInt	Byte	16#0006	Einstellung der Anzahl Aufbauversuche (Standardwert: 6 Aufbauversuche) 1 bis 255

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RepetitionAttempts	IN	USInt	Byte	16#0006	Einstellung der Anzahl Übertragungsversuche (Standardwert: 6 Aufbauversuche) 1 bis 255
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Send_P2P: Daten senden

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Send_P2P (Punkt-zu-Punkt-Daten senden) startet die Übertragung der Daten und überträgt den Inhalt des zugewiesenen Puffers zum Kommunikationsmodul. Das Programm der CPU wird weiterhin ausgeführt, während das CM die Daten mit der Datenübertragungsrate sendet. Es darf zu jeder Zeit nur eine Sendeanweisung pro Kommunikationsmodul anstehen. Das CM gibt einen Fehler aus, wenn eine zweite Anweisung Send_P2P ausgeführt wird, während das CM bereits ein Telegramm sendet.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BUFFER	IN	Variant	Any	0	Dieser Parameter zeigt auf den Speicherbereich des Sendepuffers. **Hinweise:** • Boolsche Daten oder Boolesche Felder werden nicht unterstützt. • Liegt der Sendepuffer im optimierten Speicherbereich, dann beträgt die maximale zulässige Länge der gesendeten Daten 1024 Byte. Ausnahme: Arrays von Byte, Word oder DWord werden bis zu einer Länge von 4096 Byte unterstützt. • Ist der Sendepuffer ein String oder WString, dann wird der Inhalt des Strings ohne die aktuelle und maximale Länge übertragen. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 4116)"
LENGTH	IN	UInt	Word	0	Länge der zu übertragenden Daten in Byte. Mit LENGTH = 0 wird der im Parameter BUFFER adressierte Speicherbereich komplett übertragen. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 4116)"
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Send_P2P Mit 1 wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf 0 gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Parameter

Wenn eine Sendeanweisung in Bearbeitung ist, sind die Ausgänge DONE und ERROR im Zustand FALSE. Nach dem Ende einer Sendeanweisung wird einer der Ausgänge DONE oder ERROR für einen Zyklus auf TRUE gesetzt, um den Zustand der Sendeanweisung zu melden. Wenn ERROR im Zustand TRUE ist, kann der Fehlercode am Ausgang STATUS ausgewertet werden.

Die Anweisung gibt den Status 16#7001 aus, wenn die Kommunikationsschnittstelle die Sendedaten annimmt. Nachfolgende Ausführungen von Send_P2P geben den Wert 16#7002 aus, wenn das CM immer noch sendet. Nach dem Ende der Sendeanweisung gibt das CM den Status 16#0000 für die Sendeanweisung aus (sofern kein Fehler aufgetreten ist). Nachfolgende Ausführungen von Send_P2P mit REQ = 0 geben den Status 16#7000 (frei) aus.

Das folgende Diagramm zeigt die Beziehung zwischen den Ausgangswerten und REQ. Hierbei wird vorausgesetzt, dass die Anweisung zyklisch aufgerufen wird, um den Status des Sendevorgangs zu prüfen (dargestellt durch die STATUS-Werte).

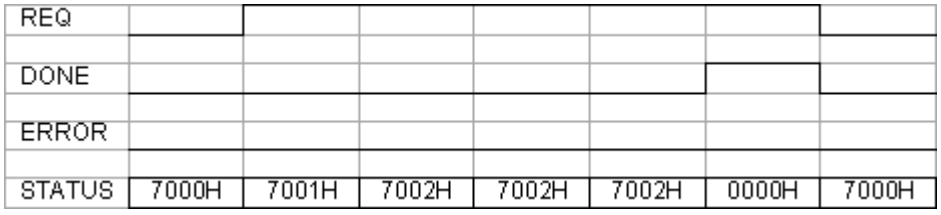

Die folgende Abbildung zeigt, wie die Parameter DONE und STATUS nur einen Zyklus lang gültig sind, wenn an der REQ-Leitung (einen Zyklus lang) ein Impuls anliegt, um die Sendeanweisung anzustoßen.

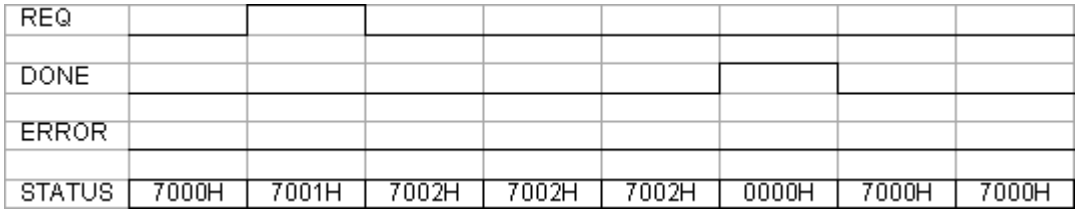

Die folgende Abbildung zeigt die Beziehung der Parameter DONE, ERROR und STATUS im Fehlerfall.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	80D1H	7000H	7000H

Die Werte DONE, ERROR und STATUS sind nur solange gültig, bis Send_P2P erneut mit dem gleichen Instanz-DB ausgeführt wird.

Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen

Interaktion der Parameter BUFFER und LENGTH für Send_P2P

Die Mindestdatengröße, die von der Anweisung Send_P2P gesendet werden kann, ist ein Byte.

Der Parameter BUFFER legt die Größe der zu sendenden Daten fest, falls beim Aufruf am Parameter LENTH eine "0" übergeben wird. Damit ist die Angabe einer Variable ausreichend.

Sie können für den Parameter BUFFER weder den Datentyp Bool noch Arrays vom Typ Bool verwenden. Wenn große Datenmengen übertragen werden, empfiehlt sich die Abbildung auf die Datentypen Array oder Struktur.

Tabelle 4-153 Parameter BUFFER

BUFFER	Beschreibung
Elementarer Datentyp	Beim Senden gilt: Der LENGTH-Wert muss die Bytezahl dieses Datentyps enthalten. Beispiel: Bei einem Word-Wert muss LENGTH zwei sein. Bei einem DWord- oder Real-Wert muss LENGTH vier sein.
Struktur	Beim optimized Speicher gilt: Die maximale erlaubte Länge des BUFFER ist 1024 Byte; ansonsten sind modulabhängig bis zu 4 kByte zulässig. Beim Senden gilt: Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge der Struktur enthalten, dann werden nur die ersten LENGTH Bytes der Struktur aus BUFFER gesendet.
Array	Beim optimized Speicher gilt: Falls der Array-Datentyp ungleich Byte, Word oder DWord ist, dann ist die maximale erlaubte Länge des Buffers 1024 Byte. Beim nicht optimized Speicher können modulabhängig bis zu 4 kByte übertragen werden, unabhängig von der Datenstruktur. Beim Senden gilt: Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge des Arrays enthalten, wobei es sich um ein Vielfaches der Bytezahl des Datenelements handeln muss. Beispiel: Der Parameter LENGTH eines Arrays vom Typ Word muss ein Vielfaches von zwei sein und bei einem Array vom Typ Real ein Vielfaches von vier. Wenn LENGTH angegeben wird, wird die Anzahl der Array-Elemente übertragen. Wenn beispielsweise BUFFER ein Array mit 15 DWord -Elementen (insgesamt 60 Bytes) enthält und Sie geben LENGTH = 20 an, werden die ersten fünf DWord -Elemente aus dem Array übertragen.
String	Der Parameter LENGTH enthält die Anzahl der zu sendenden Zeichen. Nur die Zeichen des String werden übertragen. Die Bytes mit der maximalen und der tatsächlichen Länge des String werden nicht gesendet.

Tabelle 4-154 Parameter LENGTH

LENGTH	Beschreibung
= 0	Der komplette Inhalt des über BUFFER spezifizierten Speicherbereichs wird übertragen. Falls BUFFER auf einen String zeigt, wird der komplette String-Inhalt übertragen, ohne die Bytes mit der maximalen und tatsächlichen Länge.
> 0	Es wird der Inhalt bis zur parametrierten Länge des über BUFFER spezifizierten Speicherbereichs übertragen.

Receive_P2P: Daten empfangen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Receive_P2P (Daten über eine Punkt-zu-Punkt-Kommunikation empfangen) prüft die Telegramme, die im CM empfangen wurden. Wenn ein Telegramm verfügbar ist, wird es vom CM zur CPU übertragen. Ein Empfangsfehler wird am Parameter STATUS angezeigt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BUFFER	IN	Variant	Any	0	Dieser Parameter zeigt auf die Anfangsadresse des Empfangspuffers. Dieser Puffer muss groß genug sein, um die maximale Telegrammlänge zu empfangen. Hinweis: • Boolesche Daten oder Boolesche Felder werden nicht unterstützt. • Liegt der Empfangspuffer im optimierten Speicherbereich, dann beträgt die maximal zulässige Länge der empfangenen Daten 1024 Byte. Ausnahme: Arrays von Byte, Word oder DWord werden bis zu einer Länge von 4096 Byte unterstützt. • Ist der Empfangspuffer ein String oder WString, dann werden die empfangenen Daten in den Inhalt des Strings geschrieben und die aktuelle Länge des Strings wird entsprechend gesetzt. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 4116)"
LENGTH	OUT	UInt	Word	0	Länge des empfangenen Telegramms in Byte Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 4116)"
NDR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, wenn neue Daten bereit sind und die Anweisung fehlerfrei beendet wurde.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Falls ERROR im Zustand TRUE ist, kann der Fehlercode am Ausgang STATUS ausgewertet werden. Der STATUS-Wert liefert den Grund für die Beendigung der Empfangsoperation im CM.
Dies ist typischerweise ein positiver Wert, der angibt, dass die Empfangsoperation erfolgreich war und welches Telegrammendekriterium erkannt wurde.
Ist der STATUS-Wert negativ (das höchstwertige Bit des Hexadezimalwerts ist gesetzt), wurde die Empfangsoperation wegen einer Fehlerbedingung wie Paritäts-, Framing- oder Überlauffehler beendet.

Jedes Kommunikationsmodul kann eine modulspezifische Anzahl von Telegrammen puffern. Sind mehrere Telegramme im CM verfügbar, gibt die Anweisung Receive_P2P das älteste verfügbare Telegramm aus (FIFO).

Receive_Reset: Empfangspuffer löschen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Receive_Reset (Empfänger zurücksetzen) löscht den Empfangspuffer im CM.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE bedeutet, dass die letzte Anforderung fehlerfrei ausgeführt wurde.
ERROR	OUT	Bool		FALSE	TRUE bedeutet, dass die letzte Anforderung mit Fehlern ausgeführt wurde. Ist dieser Ausgang TRUE, enthält Ausgang STATUS die zugehörigen Fehlercodes.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Signal_Get: Status lesen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Signal_Get (RS232-Signale abrufen) liest die aktuellen Zustände der RS232-Begeitsignale und zeigt sie an den entsprechenden Ausgängen der Anweisung an.

Hinweis

Einschränkung

- Diese Anweisung ist nur bei den CMs RS232 BA und RS232 HF anwendbar.
- Falls bei der Betriebsart RS232C eingestellt ist, ist diese Anweisung auch beim CM PtP (ET200SP) anwendbar.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
NDR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, wenn die RS232-Begleitsignale gelesen wurden und die Anweisung fehlerfrei beendet wurde.
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))
DTR	OUT	Bool		FALSE	Datengerät bereit, Modul bereit (Ausgang)
DSR	OUT	Bool		FALSE	Datengerät bereit, Kommunikationsteilnehmer bereit (Eingang)
RTS	OUT	Bool		FALSE	Sendeanforderung, Modul sendebereit (Ausgang)
CTS	OUT	Bool		FALSE	Sendebereitschaft, Kommunikationsteilnehmer kann Daten empfangen (Eingang)
DCD	OUT	Bool		FALSE	Datenträgersignal erkannt, Signalpegel empfangen
RING	OUT	Bool		FALSE	Rufanzeige, Meldung eines eingehenden Rufs

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Signal_Set: Begleitsignale setzen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Signal_Set (RS232-Signale setzen) erlaubt es der RS232-Kommunikationssignale zu setzen.

Hinweis

Einschränkungen

- Diese Anweisung ist nur bei den CMs RS232 BA und RS232 HF anwendbar.
- Falls bei der Betriebsart RS232C eingestellt ist, ist diese Anweisung auch beim CM PtP (ET200SP) anwendbar.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
SIGNAL	IN	Byte		0	Auswahl des zu setzenden Signals (mehrere zulässig): • 01H = RTS • 02H = DTR • 04H = DSR (gilt nur für Schnittstellentyp DCE)
RTS	IN	Bool		FALSE	Sendeanforderung, Modul sendebereit Diesen Wert am Ausgang setzen (TRUE oder FALSE), Standardwert: FALSE
DTR	IN	Bool		FALSE	Datenendgerät bereit, Modul bereit Diesen Wert am Ausgang setzen (TRUE oder FALSE), Standardwert: FALSE
DSR	IN	Bool		FALSE	Datenendgerät bereit (gilt nur für Schnittstellentyp DCE), nicht verwendet.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
DONE	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Get_Features: Erweiterte Funktionen holen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Get_Features (Erweiterte Funktionen holen) können, sofern vom Modul unterstützt, Informationen über die Fähigkeit des Moduls zur CRC-Unterstützung und zur Generierung von Diagnosemeldungen abgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	PORT	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
NDR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, wenn neue Daten bereit sind und die Anweisung fehlerfrei beendet wurde
MODBUS_CRC	OUT	Bool		FALSE	Modbus CRC-Unterstützung
DIAG_ALARM	OUT	Bool		FALSE	Generierung von Diagnosemeldungen
SUPPLY_VOLT	OUT	Bool		FALSE	Diagnose für fehlende Versorgungsspannung L+ verfügbar
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Set_Features: Erweiterte Funktionen setzen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Set_Features (Erweiterte Funktionen auswählen) können, sofern vom Modul unterstützt, die CRC-Unterstützung und die Generierung von Diagnosemeldungen aktiviert werden.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Die Anweisung zum Setzen erweiterter Funktionen wird bei steigender Flanke an diesem Eingang gestartet.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
EN_MODBUS_CRC	IN	Bool		FALSE	Modbus CRC-Unterstützung aktivieren
EN_DIAG_ALARM	IN	Bool		FALSE	Generierung von Diagnosemeldungen aktivieren
EN_SUPPLY_VOLT	IN	Bool		FALSE	Diagnose für fehlende Versorgungsspannung L+ einschalten
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Eine Ausführung lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4126))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 4099)".

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/ maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüberschreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichenverzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endesequenz"	-
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertragung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8088	Längenangabe größer als der im Empfangspuffer eingestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehlercode auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wählen Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		

Fehlercode	Beschreibung	Abhilfe
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).

Fehlercode	Beschreibung	Abhilfe
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.

Fehlercode	Beschreibung	Abhilfe
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)

Fehlercode	Beschreibung	Abhilfe
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.

Fehlercode	Beschreibung	Abhilfe
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

Fehlercode	Beschreibung	Abhilfe
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions

Fehlercode	Beschreibung	Abhilfe
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode	Beschreibung	Abhilfe
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)

Fehlercode	Beschreibung	Abhilfe
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG

Fehlercode	Beschreibung	Abhilfe
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.

Fehlercode	Beschreibung	Abhilfe
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.

Fehlercode	Beschreibung	Abhilfe
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.
16#8828 [1)]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1)]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1)]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1)]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1)]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8428 [1)]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG

Fehlercode	Beschreibung	Abhilfe
16#8455 [1]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.

Fehlercode	Beschreibung	Abhilfe
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

USS

Überblick über die USS-Kommunikation

USS-Kommunikation

Die USS-Anweisungen steuern den Betrieb von Antrieben, die das Protokoll der universellen seriellen Schnittstelle (USS) unterstützen. Mit den PtP-Kommunikationsmodulen können Sie über RS485-Verbindungen und den USS-Anweisungen mit mehreren Antrieben kommunizieren. Jeder RS485-Port kann bis zu 16 Antriebe betreiben.

Das USS-Protokoll nutzt ein Master/Slave-Netzwerk für die Kommunikation über einen seriellen Bus. Der Master verwendet einen Adressparameter, um Daten an einen ausgewählten Slave zu senden. Ein Slave selbst kann niemals senden, ohne dafür zuvor eine Anforderung erhalten zu haben. Die Kommunikation zwischen den einzelnen Slaves ist nicht möglich. Die USS-Kommunikation funktioniert im Halbduplex-Betrieb. Die folgende Abbildung zeigt ein Netzwerkdiagramm für eine Beispielanwendung mit 16 Antrieben.

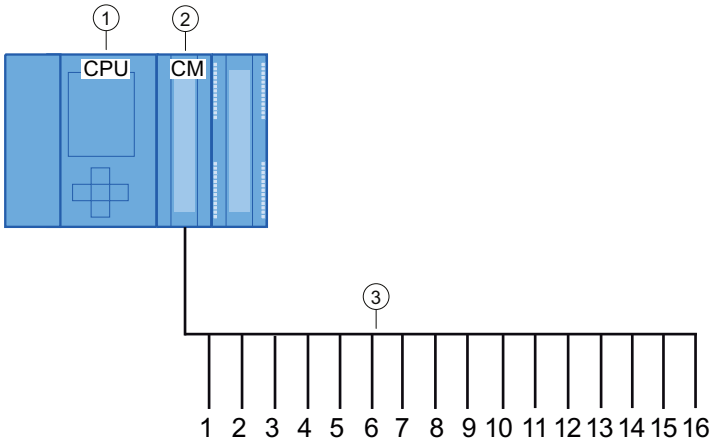

① CPU
② CM
③ USS-Antriebe an einem USS-Netz

Bild 4-4 Verschaltungsbeispiel mit S7-1500 Kommunikationsmodul

Hinweis

Kommunikation mit einem Antrieb über RS232

Prinzipiell können auch CM PtP RS232 BA und CM PtP RS232 HF zur Kommunikation mit einem Antrieb eingesetzt werden. An einem RS232-Port kann jedoch nur **ein** Antrieb angeschlossen werden.

Kommunikation mit einem Antrieb über RS422

Prinzipiell kann auch die RS422-Schnittstelle der CM PtP RS422/485 BA und CM PtP RS422/485 HF zur Kommunikation mit einem Antrieb eingesetzt werden. An einem RS422-Port kann jedoch nur **ein** Antrieb angeschlossen werden.

USS-Anweisungen in Ihrem Programm

- USS_Port_Scan: Die Anweisung USS_Port_Scan ermöglicht Ihnen die Kommunikation über ein Kommunikationsmodul mit bis zu 16 Antrieben über ein USS-Netzwerk (muss zyklisch aufgerufen werden).
  Es ist nur eine Anweisung USS_Port_Scan pro PtP-Kommunikationsport im Programm vorhanden, die die Übertragung zu allen Antrieben steuert.
- USS_Drive_Control: Die Anweisung USS_Drive_Control ermöglicht Ihnen, die Sendedaten von USS_Port_Scan für einen Antrieb aufzubereiten und dessen Empfangsdaten anzuzeigen.
  USS_Drive_Control konfiguriert die zu sendenden Daten und wertet Daten aus, die in einer vorherigen Anforderung durch USS_Port_Scan empfangen wurden.
- USS_Read_Param: Die Anweisung USS_Read_Param ermöglicht Ihnen, Parameter aus einem Antrieb auszulesen.
- USS_Write_Param: Die Anweisung USS_Write_Param ermöglicht Ihnen, Parameter in einem Antrieb zu ändern.

Voraussetzungen für den Einsatz des USS-Protokolls

Die vier USS-Anweisungen verwenden 2 FBs und 2 FCs zur Unterstützung des USS-Protokolls. Für jedes USS-Netzwerk wird ein Instanz-Datenbaustein für USS_Port_Scan und ein Instanz-Datenbaustein gemeinsam für alle Aufrufe von USS_Drive_Conrol verwendet.

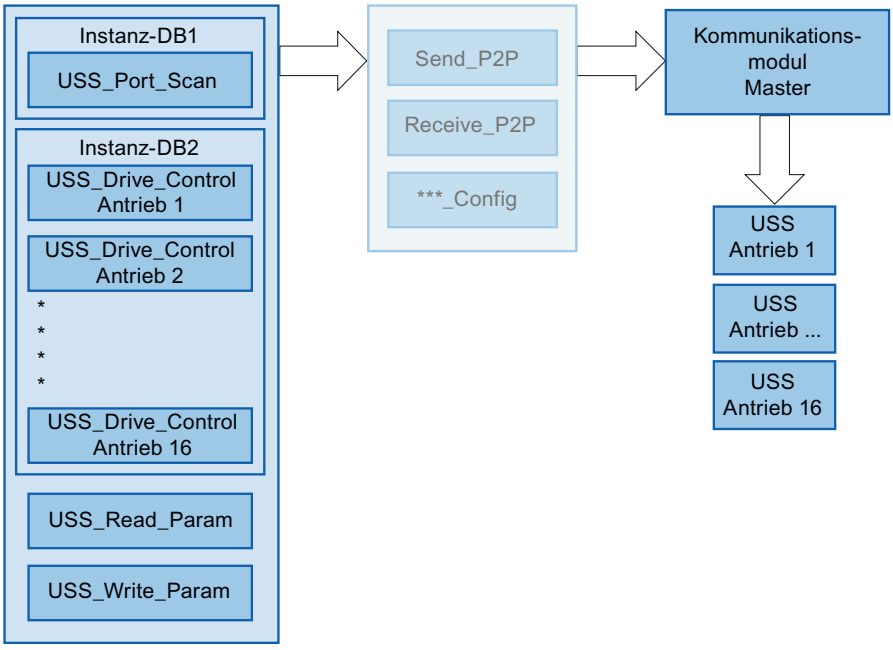

Bild 4-5 Programmablauf - USS

Alle Antriebe (max. 16), die an einen RS485-Port angeschlossen sind, sind Teil desselben USS-Netzwerks. Alle Antriebe, die an einen anderen RS485-Port angeschlossen sind, sind Teil eines anderen USS-Netzwerks. Jedes USS-Netzwerk wird mithilfe eines eindeutigen Instanz-Datenbausteins für alle Anweisungen USS_Drive_Control und eines weiteren Instanz-Datenbausteins für die Anweisung USS_Port_Scan verwaltet. Alle Anweisungen, die zu einem USS-Netzwerk gehören, müssen den Instanz-Datenbaustein für USS_Drive_Conrol gemeinsam nutzen. Die Anweisungen USS_Port_Scan, USS_Read_Param und

USS_Write_Param haben dafür den Parameter USS_DB, der mit dem (statischen) Parameter USS_DB des Instanz-DBs der Anweisung USS_Drive_Control verschaltet werden muss.

- Die Anweisungen USS_Drive_Control und USS_Port_Scan sind Funktionsbausteine (FB). Falls Sie die Anweisung USS_Drive_Control oder USS_Port_Scan in den Programmiereditor einfügen, werden Sie im Dialog "Aufrufoptionen" aufgefordert, einen DB für diesen FB zuzuweisen. Wenn es sich um die erste Anweisung USS_Drive_Control in diesem Programm für dieses USS-Netzwerk handelt, können Sie die DB-Standardzuweisung übernehmen (oder ggf. den Namen ändern), und der neue DB wird für Sie erstellt. Wenn es sich jedoch nicht um die erste Anweisung USS_Drive_Control für diesen Antrieb handelt, müssen Sie im Dialog "Aufrufoptionen" in der Klappliste den DB auswählen, der diesem USS-Netzwerk bereits zuvor zugewiesen wurde.
- Bei den Anweisungen USS_Write_Param und USS_Read_Param handelt es sich um Funktionen (FCs). Wenn Sie diese FCs im Editor einfügen, wird kein DB zugewiesen. Falls Sie diese FCs oder die Anweisung USS_Port_Scan im Editor einfügen, müssen Sie dem Eingang USS_DB dieser Anweisungen den Parameter USS_DB des jeweiligen Instanz-DB von USS_Drive_Control zuweisen. Doppelklicken Sie auf das Parameterfeld und klicken Sie dann auf das Symbol, um die verfügbaren DBs anzuzeigen. Geben Sie einen Punkt "." ein und wählen Sie dann aus der Klappliste den Parameter USS_DB aus.
- Die Funktion USS_Port_Scan steuert die Kommunikation zwischen der CPU und den Antrieben über den Punkt-zu-Punkt-RS485-Kommunikationsport. Bei jedem Aufruf dieser Funktion wird eine Kommunikation mit einem Antrieb bearbeitet. Ihr Programm muss diese Funktion schnell genug aufrufen, sodass die Antriebe keine Zeitüberschreitung melden. Um ein konstantes Zeitverhalten des Telegrammversands zu garantieren, sollten Sie diese Anweisung in einem Weckalarm-OB aufrufen.
- Die Anweisung USS_Drive_Control gibt Ihrem Programm Zugriff auf einen angegebenen Antrieb im USS-Netzwerk. Seine Ein- und Ausgänge entsprechen den Zuständen und den Bedienfunktionen des Antriebs. Sind 16 Antriebe im Netzwerk vorhanden, so muss USS_Drive_Control in Ihrem Programm 16-mal aufgerufen werden, also jeweils einmal für jeden Antrieb.
  Sie sollten die Anweisung USS_Drive_Control nur aus einem zyklischen OB aufrufen.
- Mit den Funktionen USS_Read_Param und USS_Write_Param werden die Betriebsparameter des Antriebs gelesen und geschrieben. Diese Parameter steuern die interne Funktionsweise des Antriebs. Eine Definition dieser Parameter finden Sie im Handbuch des Antriebs. Ihr Programm kann eine beliebige Anzahl dieser Funktionen enthalten, es kann jedoch immer nur eine Lese- oder Schreibanforderung für einen Antrieb aktiv sein. Sie dürfen die Funktionen USS_Read_Param und USS_Write_Param nur aus dem Zyklus-OB eines Hauptprogramms aufrufen.

ACHTUNG

Aufrufe der USS-Anweisungen

Rufen Sie USS_Drive_Control, USS_Read_Param und USS_Write_Param nur aus einem Zyklus-OB des Hauptprogramms auf. Die Anweisung USS_Port_Scan kann aus einem beliebigen OB aufgerufen werden, üblicherweise wird sie aus einem Weckalarm-OB aufgerufen.

Verwenden Sie die Anweisungen USS_Drive_Control, USS_Read_Param und USS_Write_Param nicht in einem OB mit einer höheren Priorität als die entsprechende Anweisung USS_Port_Scan. Fügen Sie beispielsweise nicht USS_Port_Scan in das Hauptprogramm und USS_Read_Param in einen Weckalarm-OB ein. Wird die Ausführung des USS_Port_Scan durch eine andere Anweisung unterbrochen, kann es zu unerwarteten Fehlern kommen.

Hinweis

Parameterkennungswert

Für die Antriebe müssen Sie die Verwendung von 4 PKW-Wörtern einrichten.

Zeit für die Kommunikation mit dem Antrieb berechnen

Die Kommunikation mit dem Antrieb läuft asynchron zum Zyklus der CPU ab. Die CPU durchläuft üblicherweise mehrere Zyklen, bevor die Kommunikation mit einem Antrieb beendet ist.

Damit die beim Antrieb eingestellte Zeitüberwachung nicht anspricht, müssen die Sendetelegramme zum Antrieb innerhalb dieser Zeitüberwachung gesendet werden. Dabei muss die Zahl der Wiederholversuche mit eingerechnet werden, falls zur Fertigstellung der Transaktion aufgrund von Kommunikationsfehlern mehrere Versuche notwendig sind. Standardmäßig werden für das USS Protokoll bei jeder Transaktion bis zu 2 Wiederholungen durchgeführt.
Der maximale Zeitabstand zwischen zwei Sendetelegrammen berechnet sich wie folgt:

N * (5 * Zykluszeit + Telegrammlaufzeit + Timeout des Empfangstelegramms max.) * (Anzahl Sendeversuche)

N	Anzahl der Antriebe in diesem Netzwerk
Faktor 5	Für das Senden und Empfangen von Telegrammen werden typischerweise 5 Zyklen benötigt.
Zykluszeit	Max. Zykluszeit des Weckalarm-OBs, in der die Anweisung USS_Port_Scan aufgerufen wird.
Telegrammlaufzeit	Telegrammlaufzeit = (Anzahl Zeichen pro Telegramm) * (11 Bit pro Zeichen) / (Datenübertragungsrate in Bit/s)
Anzahl Sendeversuche	Anzahl Wiederholversuche + 1

Timeout des Empfangstelegramms	RCVTIME (falls keine Antwort vom Antrieb empfangen wird)
Timeout des Empfangstelegramms max.	RCVTIME + MSGTIME (falls kurz vor Ablauf von RCVTIME eine unvollständige Antwort empfangen wird und die Überwachung von MSGTIME abläuft **oder** falls zum Ablauf von RCVTIME noch eine Antwort in Bearbeitung ist, wird der Timeout um die MSGTIME verlängert)

Für den "Timeout des Empfangstelegramms" gelten folgende Zeiten (ms):

Bit/s	115200	57600	38400	19200	9600	4800	2400	1200
Receive_Conditions.END.RCVTIME	25	29	33	56	72	100	100	100
Receive_Conditions.END.MSGTIME	25	29	33	56	72	124	240	460

Timeout des Empfangstelegramms max. = (Receive_Conditions.END.RCVTIME (0,072 s) + Receive_Conditions.END.MSGTIME (0,072 s))

Beispiel:

5 Antriebe
Datenübertragungsrate = 9600 Bit/s
28 Zeichen pro Telegramm
Zykluszeit = 0.020 s
Anzahl Wiederholversuche = 2

Zeitabstand = 5 * (5 * 0,02 s + 1 * 28 * 11 / 9600 s + 0,072 s + 0,072 s) * 3 = 3 s

Die Zeitüberwachung des Antriebs muss in diesem Fall auf ca. 3 Sekunden eingestellt werden.

USS_Port_Scan / USS_Port_Scan_31: Kommunikation über USS-Netzwerk

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Port_Scan_31

Der Einsatz der Anweisung USS_Port_Scan_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Port_Scan bearbeitet die Kommunikation über ein USS-Netzwerk für max. 16 Antriebe.

Die Anweisung USS_Port_Scan_31 bearbeitet die Kommunikation über ein USS-Netzwerk für max. 31 Antriebe.

STEP 7 erstellt automatisch den Instanz-DB, wenn Sie die Anweisung einfügen.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
PORT	IN	Port	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BAUD	IN	DInt		9600	Datenübertragungsrate für die USS-Kommunikation Zulässig sind: • 1200 Bit/s • 2400 Bit/s • 4800 Bit/s • 9600 Bit/s • 19200 Bit/s • 38400 Bit/s • 57600 Bit/s • 115200 Bit/s
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
COM_RST	INOUT	---	Bool	FALSE	Initialisierung der Anweisung USS_Port_Scan / USS_Port_Scan_31 Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
ERROR	OUT	Bool		FALSE	Falls TRUE, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 4160)).

Es ist nur eine Anweisung USS_Port_Scan / USS_Port_Scan_31 pro PtP-Kommunikationsport im Programm vorhanden und jeder Aufruf dieser Anweisung steuert eine Übertragung zu oder von allen Antrieben dieses Netzwerks. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den gleichen Instanz-DB nutzen.

Ihr Programm muss die Anweisung USS_Port_Scan / USS_Port_Scan_31 so oft ausführen, dass kein Timeout im Antrieb auftritt (siehe Voraussetzungen für den Einsatz des USS-Protokolls (Seite 4142) Abschnitt "Zeit für die Kommunikation mit dem Antrieb berechnen").

Die Anweisung USS_Port_Scan / USS_Port_Scan_31 wird üblicherweise aus einem Weckalarm-OB aufgerufen, um Antriebs-Timeouts zu verhindern und die letzten USS-Datenaktualisierungen für Aufrufe von USS_Drive_Control / USS_Drive_Control_31 verfügbar zu haben.

USS_Port_Scan / USS_Port_Scan_31 Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von USS_Port_Scan / USS_Port_Scan_31, die in Ihrem Programm verwendet werden können.

Tabelle 4-155 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
MODE	USInt	4	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master; CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave; CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1)]
LINE_PRE	USInt	2	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) : Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V : Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.
BRK_DET	USInt	0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
RETRIES_MAX	SInt/Byte	2	Zahl der Wiederholversuche bei Auftreten von Kommunikationsfehlern. Mit diesem Parameter können Sie die Zahl der Wiederholversuche für das Senden eines Anforderungstelegramms setzen, falls das Antworttelegramm nicht innerhalb der eingestellten Zeit empfangen wird.

Variable	Datentyp	Standard	Beschreibung
EN_DI-AG_ALARM	Bool	0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
EN_SUP-PLY_VOLT	Bool	0	Diagnose für fehlende Versorgungsspannung L+ einschalten • 0 - nicht aktiviert • 1 - aktiviert

[1] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Die Version 2.5 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

Anweisungsversionen

USS_Port_Scan:

Die Version 2.5 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

USS_Port_Scan_31:

Die Version 1.2 ist funktional identisch zur Version 1.1 und wurde aufgrund interner Maßnahmen erhöht.

USS_Drive_Control / USS_Drive_Control_31: Daten für den Antrieb vorbereiten und anzeigen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Drive_Control_31

Der Einsatz der Anweisung USS_Drive_Control_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Drive_Control bereitet Sendedaten für maximal 16 Antriebe vor und wertet die Antwortdaten der Antriebe aus.

Die Anweisung USS_Drive_Control_31 bereitet Sendedaten für maximal 31 Antriebe vor und wertet die Antwortdaten der Antriebe aus.

Es muss für jeden Antrieb eine eigene Instanz der Anweisung verwendet werden, und es müssen alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, den gleichen Instanz-Datenbaustein verwenden. Sie müssen den DB-Namen eingeben, wenn Sie die erste Anweisung USS_Drive_Control / USS_Drive_Control_31

einfügen. Dann verweisen Sie auf diesen DB, der beim Einfügen der ersten Anweisung angelegt wurde.

STEP 7 erstellt automatisch den DB, wenn Sie die Anweisung einfügen.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RUN	IN	Bool		FALSE	Startbit des Antriebs: Ist dieser Parameter TRUE, so ermöglicht dieser Eingang den Betrieb des Antriebs mit der voreingestellten Drehzahl. Wenn RUN im Betrieb des Antriebs nach FALSE wechselt, läuft der Motor bis zum Stillstand aus. Dieses Verhalten unterscheidet sich von der Abschaltung der Spannungsversorgung (OFF2) und vom Bremsen des Motors (OFF3).
OFF2	IN	Bool		FALSE	Bit "Zum Stillstand auslaufen": Ist dieser Parameter FALSE, so veranlasst dieses Bit das Auslaufen des Antriebs, ohne zu bremsen.
OFF3	IN	Bool		FALSE	Schnelles Stoppbit: Ist dieser Parameter FALSE, so verursacht dieses Bit einen schnellen Halt durch Abbremsen des Antriebs.
F_ACK	IN	Bool		FALSE	Fehlerquittierungsbit: Mit diesem Bit wird das Fehlerbit eines Antriebs zurückgesetzt. Das Bit wird nach dem Löschen des Fehlers gesetzt und der Antrieb erkennt damit, dass der vorherige Fehler nicht mehr gemeldet werden muss.
DIR	IN	Bool		FALSE	Richtungssteuerung des Antriebs: Dieses Bit wird gesetzt, wenn der Antrieb in Vorwärtsrichtung laufen soll (wenn SPEED_SP positiv ist; siehe Tabelle "Interaktion der Parameter SPEED_SP und DIR").
DRIVE	IN	USInt	Byte	1	Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PZD_LEN	IN	USInt	Byte	2	Wortlänge: Dies ist die Anzahl der PZD-Datenwörter. Gültige Werte sind 2, 4, 6 oder 8 Wörter.
SPEED_SP	IN	Real		0.0	Drehzahlsollwert: Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Frequenz. Ein positiver Wert bedeutet, dass der Antrieb vorwärts läuft (wenn DIR wahr ist). Gültig ist der Bereich von 200,00 bis -200,00.
CTRL3	IN	Word		0	Steuerwort 3: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL4	IN	Word		0	Steuerwort 4: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL5	IN	Word		0	Steuerwort 5: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
CTRL6	IN	Word		0	Steuerwort 6: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL7	IN	Word		0	Steuerwort 7: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL8	IN	Word		0	Steuerwort 8: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung USS_Drive_Control / USS_Drive_Control_31 Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
NDR	OUT	Bool		FALSE	Neue Daten bereit: Ist dieser Parameter TRUE, so meldet das Bit, dass am Ausgang Daten einer neuen Kommunikationsanforderung bereitstehen.
ERROR	OUT	Bool		FALSE	Fehler aufgetreten: Falls TRUE, weist dies darauf hin, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 4160)).
RUN_EN	OUT	Bool		FALSE	Betrieb freigegeben: Dieses Bit meldet, ob der Antrieb läuft.
D_DIR	OUT	Bool		FALSE	Antriebsrichtung: Dieses Bit meldet, ob der Antrieb vorwärts läuft. • FALSE – vorwärts • TRUE – rückwärts
INHIBIT	OUT	Bool		FALSE	Antrieb gesperrt: Dieses Bit meldet den Zustand des Sperrbits für den Antrieb. • FALSE – nicht gesperrt • TRUE – gesperrt
FAULT	OUT	Bool		FALSE	Antriebsfehler: Dieses Bit meldet, dass im Antrieb ein Fehler aufgetreten ist. Sie müssen die Störung beheben und Bit F_ACK setzen, um dieses Bit zu löschen.
SPEED	OUT	Real		0.0	Istwert Antriebsdrehzahl (skalierter Wert von STATUS 2 des Antriebs): Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Drehzahl.
STATUS1	OUT	Word		0	STATUS 1 des Antriebs Dieser Wert enthält feste Zustandsbits eines Antriebs.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
STATUS3	OUT	Word		0	STATUS 3 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS4	OUT	Word		0	STATUS 4 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS5	OUT	Word		0	STATUS 5 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS6	OUT	Word		0	STATUS 6 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS7	OUT	Word		0	STATUS 7 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS8	OUT	Word		0	STATUS 8 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.

Wenn die erste Ausführung von USS_Drive_Control / USS_Drive_Control_31 erfolgt, wird der von der USS-Adresse (Parameter DRIVE) angegebene Antrieb im Instanz-DB initialisiert. Nach der Initialisierung können nachfolgende Anweisungen USS_Port_Scan / USS_Port_Scan_31 die Kommunikation mit dem Antrieb an dieser Antriebsnummer beginnen.

Wenn Sie die Antriebsnummer ändern, muss die CPU zunächst in STOP und dann wieder in RUN versetzt werden, damit der Instanz-DB initialisiert wird. Die Eingangsparameter werden im USS-Sendepuffer konfiguriert und die Ausgänge werden, sofern vorhanden, aus einem "vorherigen" gültigen Antwortpuffer gelesen. USS_Drive_Control / USS_Drive_Control_31 konfiguriert nur die zu sendenden Daten und wertet Daten aus, die in einer vorherigen Anforderung empfangen wurden.

Sie können die Drehrichtung des Antriebs entweder über den Eingang D_IR (Bool) oder über das Vorzeichen (positiv oder negativ) am Eingang SPEED_SP (Real) steuern. Die folgende Tabelle erläutert, wie diese Eingänge zusammen funktionieren, um die Drehrichtung des Antriebs zu bestimmen, vorausgesetzt der Motor dreht vorwärts.

Tabelle 4-156 Interaktion der Parameter SPEED_SP und DIR

SPEED_SP	DIR	Drehrichtung des Antriebs
Wert > 0	0	Rückwärts
Wert > 0	1	Vorwärts
Wert < 0	0	Vorwärts
Wert < 0	1	Rückwärts

USS_Drive_Control / USS_Drive_Control_31 Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von USS_Drive_Control / USS_Drive_Control_31, die in Ihrem Programm verwendet werden können.

Tabelle 4-157 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
USS_DB. w_USSExtendedError	Word	16#0	USS Drive Extended Error Code - Antriebsspezifischer Wert Die Bedeutung der Fehlermeldung ist abhängig davon, welche Anweisung **zuvor** einen Fehler (ERROR = TRUE) gemeldet hat. Folgende Fälle werden unterschieden: • USS_Write_Param / USS_Write_Param_31: Die Bedeutung des Fehlercodes finden Sie in der Beschreibung ihres Antriebs. • USS_Read_Param / USS_Read_Param_31: Die Bedeutung des Fehlercodes finden Sie in der Beschreibung ihres Antriebs. • USS_Port_Scan / USS_Port_Scan_31: Nummer des von der Fehlermeldung betroffenen Antriebs.

Anweisungsversionen

USS_Drive_Control:

Die Version 2.0 ist funktional identisch zur Version 1.2 und wurde aufgrund interner Maßnahmen erhöht.

USS_Drive_Control_31:

Die Version 2.0 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

USS_Read_Param / USS_Read_Param_31: Daten aus dem Antrieb lesen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Read_Param_31

Der Einsatz der Anweisung USS_Read_Param_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Read_Param liest einen Parameter aus einem von maximal 16 Antrieben.

Die Anweisung USS_Read_Param_31 liest einen Parameter aus einem von maximal 31 Antrieben.

Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den Instanzdatenbaustein der Anweisung USS_Drive_Control / USS_Drive_Control_31 nutzen. USS_Read_Param / USS_Read_Param_31 muss aus einem Zyklus-OB des Hauptprogramms aufgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		–	Mit einer positiven Flanke an REQ wird eine neue Leseanforderung gestellt.
DRIVE	IN	USInt	Byte	–	Adresse des Antriebs: DRIVE ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PARAM	IN	UInt		–	Parameternummer: PARAM gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich für diesen Parameter liegt zwischen 0 und 2047. Bei einigen Antrieben kann über das höchstwertige Byte des Parameters INDEX auf Parameterwerte größer als 2047 zugreifen werden. Weitere Informationen für den Zugriff auf einen erweiterten Bereich finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	IN	UInt		–	Parameterindex: INDEX gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höchstwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
DONE[1]	OUT	Bool		FALSE	Ist dieser Parameter TRUE, so steht am Ausgang VALUE der zuvor angeforderte Wert des Leseparameters an. Dieses Bit wird gesetzt, wenn die Anweisung USS_Drive_Control / USS_Drive_Control_31 die Leseantwort des Antriebs erkennt. Dieses Bit wird beim nächsten Aufruf von USS_Read_Param / USS_Read_Param_31 zurückgesetzt.
ERROR	OUT	Bool		FALSE	ERROR = TRUE: Es ist ein Fehler aufgetreten und der Ausgang STATUS ist gültig. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
STATUS	OUT	UInt		0	Fehlercode (siehe Fehlermeldungen (Seite 4160)).
VALUE	OUT	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Dies ist der Wert des Parameters, der gelesen wurde und er ist nur gültig, wenn das Bit DONE wahr ist.

[1] Das Bit DONE weist darauf hin, dass gültige Daten aus dem referenzierten Motorantrieb ausgelesen und an die CPU geliefert wurden. Es weist nicht darauf hin, dass die Anweisung in der Lage ist, sofort einen weiteren Parameter auszulesen. Eine leere Leseanforderung muss an den Motorantrieb gesendet und auch von der Anweisung quittiert werden, bevor der Parameterkanal zur Verwendung durch den jeweiligen Antrieb frei wird. Der sofortige Aufruf von USS_Read_Param / USS_Read_Param_31 oder USS_Write_Param / USS_Write_Param_31 für den spezifischen Motorantrieb führt zu dem Fehler 16#818A.

Anweisungsversionen

USS_Read_Param:

Die Version 1.5 ist funktional identisch zur Version 1.4 und wurde aufgrund interner Maßnahmen erhöht.

USS_Read_Param_31:

Die Version 1.1 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

USS_Write_Param / USS_Write_Param_31: Daten im Antrieb ändern

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Write_Param_31

Der Einsatz der Anweisung USS_Write_Param_31 ist nur auf einer S7-1500 CPU möglich.

Hinweis

Für EEPROM-Schreibanweisungen (EEPROM in einem USS-Antrieb) gilt:

Halten Sie die Anzahl der EEPROM-Schreiboperationen möglichst gering, um eine möglichst lange Lebensdauer des EEPROM zu gewährleisten.

Beschreibung

Die Anweisung USS_Write_Param ändert einen Parameter in einem von maximal 16 Antrieben.

Die Anweisung USS_Write_Param_31 ändert einen Parameter in einem von maximal 31 Antrieben.

Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den Instanzdatenbaustein des USS_Drive_Control / USS_Drive_Control_31 nutzen.

USS_Write_Param / USS_Write_Param_31 muss aus dem Zyklus-OB eines Hauptprogramms aufgerufen werden.

Parameter

Tabelle 4-158 Datentypen für die Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Mit einer positiven Flanke an REQ wird eine neue Schreibanforderung gestellt.
DRIVE	IN	USInt	Byte	–	Adresse des Antriebs: DRIVE ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PARAM	IN	UInt		–	Parameternummer: PARAM gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich für diesen Parameter liegt zwischen 0 und 2047. Bei einigen Antrieben kann über das höchstwertige Byte des Parameters INDEX auf Parameterwerte größer als 2047 zugreifen. Weitere Informationen für den Zugriff auf einen erweiterten Bereich finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	IN	UInt		–	Parameterindex: INDEX gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höchstwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.
EEPROM	IN	Bool		–	Im EEPROM des Antriebs speichern: Wenn TRUE, wird die Transaktion eines Parameters zum Schreiben in den Antrieb im EEPROM des Antriebs gespeichert. Wenn FALSE, so wird der geschriebene Wert nur temporär gespeichert und geht beim nächsten Einschalten des Antriebs verloren.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/ WinAC		
VALUE	IN	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Wert des Parameters, in den geschrieben werden soll. Er muss bei der positiven Flanke von REQ gültig sein.
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
DONE[1]	OUT	Bool		FALSE	Falls TRUE, wurde der Eingang VALUE in den Antrieb geschrieben. Dieses Bit wird gesetzt, wenn die Anweisung USS_Drive_Control / USS_Drive_Control_31 die Schreibantwort des Antriebs erkennt. Dieses Bit wird beim nächsten Aufruf von USS_Write_Param / USS_Write_Param_31 zurückgesetzt.
ERROR	OUT	Bool		FALSE	Falls TRUE, ist ein Fehler aufgetreten und der Ausgang STATUS ist gültig. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.
STATUS	OUT	UInt		0	Fehlercode (siehe Fehlermeldungen (Seite 4160)).

[1] Das Bit DONE weist darauf hin, dass gültige Daten aus dem referenzierten Motorantrieb ausgelesen und an die CPU geliefert wurden. Es weist nicht darauf hin, dass die USS-Bibliothek in der Lage ist, sofort einen weiteren Parameter auszulesen. Eine leere Schreibanforderung muss an den Motorantrieb gesendet und auch von der Anweisung quittiert werden, bevor der Parameterkanal zur Verwendung durch den jeweiligen Antrieb frei wird. Der sofortige Aufruf von USS_Read_Param / USS_Read_Param_31 oder USS_Write_Param / USS_Write_Param_31 FC für den spezifischen Motorantrieb führt zu dem Fehler 0x818A.

Anweisungsversionen

USS_Write_Param:

Die Version 1.6 ist funktional identisch zur Version 1.5 und wurde aufgrund interner Maßnahmen erhöht.

USS_Write_Param_31:

Die Version 1.1 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

Allgemeine Informationen zur Antriebseinrichtung

Voraussetzungen für die Antriebseinrichtung

- Für die Antriebe müssen Sie die Verwendung von 4 PKW-Wörtern (**P**arameter**k**ennungs**w**ert) einrichten.
- Konfigurieren Sie die Antriebe für 2, 4, 6 oder 8 PZD-Wörter (**P**ro**z**ess**d**atenbereich).
- Sorgen Sie dafür, dass die Anzahl der PZD-Wörter im Antrieb dem Eingang PZD_LEN der Anweisung USS_Drive_Control des Antriebs entspricht.
- Sorgen Sie dafür, dass die Datenübertragungsrate aller Antriebe dem Eingang BAUD der Anweisung USS_Port_Scan entspricht.
- Sorgen Sie dafür, dass der Antrieb für die USS-Kommunikation eingerichtet ist.
- Stellen Sie sicher, dass im Antrieb festgelegt ist, dass der Frequenzsollwert über die USS-Schnittstelle bereitgestellt wird.
- Stellen Sie sicher, dass die Antriebsadresse festgelegt ist (Bereiche: 1-16).
  Diese Adresse muss dem Eingang DRIVE am Baustein USS_Drive_Control des Antriebs entsprechen.
- Stellen Sie sicher, dass das RS485-Netzwerk ordnungsgemäß abgeschlossen ist.

SINAMICS V20-Antrieb anschließen und einrichten

Ein Anwendungsbeispiel zum Betrieb eines SINAMICS V20 an einer S7-1200 finden Sie im Internet. (http://support.automation.siemens.com/WW/view/de/63696870)

SINAMICS V20-Antrieb anschließen

Anschlussbeispiel eines SIEMENS G120(C)-Antriebs an einem USS-Netzwerk. Anschlussbeispiele für andere Antriebe finden im Handbuch des entsprechenden Antriebs.

Der Anschluss eines SINAMICS G120(C)-Antriebs an das USS-Netzwerk erfolgt über eine Steckverbindung. Der Anschluss ist kurzschlussfest und potenzialgetrennt.

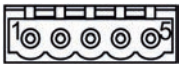

1 0 V Bezugspotential
2 RS485N, Empfangen und Senden (-)
3 RS485N, Empfangen und Senden (+)
4 Kabelschirm
5 Nicht verwendet

Bild 4-6 Anschluss USS

ACHTUNG
Unterschiedliche Bezugsspannungen Wenn Sie Geräte miteinander verbinden, die nicht die gleiche Bezugsspannung haben, kann dies unerwünschte Ströme im Verbindungskabel hervorrufen. Diese unerwünschten Ströme können Kommunikationsfehler verursachen oder Sachschaden in den Geräten hervorrufen. Stellen Sie sicher, dass alle Geräte, die Sie über ein Kommunikationskabel miteinander verbinden, entweder den gleichen Bezugsleiter im Stromkreis haben oder elektrisch getrennt sind, damit keine unerwünschten Ströme auftreten. Stellen Sie sicher, dass die Schirmung mit Masse oder Pin 1 des Bussteckers am Antrieb verbunden ist. Stellen Sie sicher, dass die Verdrahtungsklemme 2 (GND) des G120(C) mit Masse verbunden ist.

Falls der RS485-Master (z. B. S7-1200 CPU mit Kommunikkationsmoduls CM1241) über einen PROFIBUS-Stecker angeschlossen ist, verdrahten Sie die Busleitungen wie folgt:

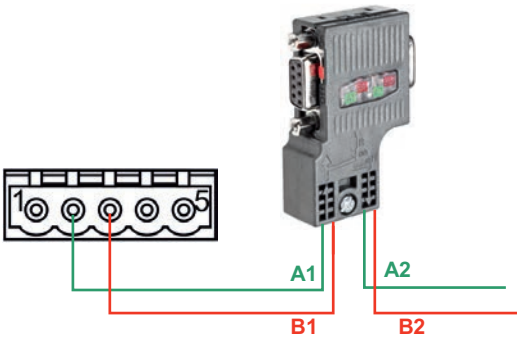

Bild 4-7 Anschluss Kommunikationsmodul

Handelt es sich beim RS485-Master um einen abschließenden Teilnehmer im Netz, oder handelt es sich um eine Punkt-zu-Punkt-Verbindung, müssen Sie die Klemmen A1 und B1 (nicht A2 und B2) des PROFIBUS-Steckers verwenden, weil diese die Abschlusseinstellungen ermöglichen (z. B. beim DP-Steckverbinder 6ES7972-0BB52-0XA0).

Falls der G120(C) als abschließender Teilnehmer im Netz konfiguriert ist, müssen Sie den Schalter für den Busabschlusswiderstand auf "ON" stellen.

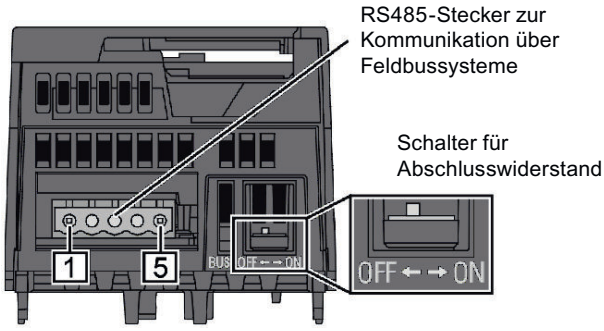

Bild 4-8 Anschluss abschließender Teilnehmer

Einrichten eines G120(C)-Antriebs

Bevor Sie einen Antrieb an die S7-1500 oder ET 200SP anschließen, müssen Sie sicherstellen, dass der Antrieb über folgende Systemparameter verfügt.

Schritt	Anweisung	Betriebsanleitung	
		G120 [1]	G120C [2]
1	Führen Sie eine Grundinbetriebnahme des Antriebs mit dem Operator Panel BOP-2 durch. Der Umrichter bietet unterschiedliche vordefinierte Belegungen (Makros) für seine Ein- und Ausgänge und die Feldbus-Schnittstelle. Wählen Sie im 9. Schritt der Grundinbetriebnahme (MAC PAR p15) das Makro 21 für die USS-Kommunikation. Damit sind auch folgende Parameter vorbelegt: • Datenübertragungsrate (p2020): 38400 Bit/s • Anzahl PZD (p2022): 2 • Anzahl PKW (p2023): variabel Hinweis: Die Grundinbetriebnahme kann auch mit der Inbetriebnahmesoftware STARTER oder mit SINAMICS Startdrive durchgeführt werden.	Kap. 4.4.3	Kap. 6.4.1
2	Legen Sie die USS-Adresse des Umrichters über die Adressschalter auf der Control Unit des G120 bzw. auf dem G120(C) fest. • Gültiger Adressbereich: 1 … 30 Hinweis: Sie können die USS-Adresse auch über Parameter p2021 oder über STARTER bzw. SINAMICS Startdrive einstellen.	Kap. 6.2.2.1	Kap. 8.4.2.1
	Bei den folgenden Schritten greifen Sie mit dem BOP-2 direkt auf Parameter zu, indem Sie deren Nummern wählen und deren Werte modifizieren.	Kap. 4.4.2	Kap. 6.4.2
3	Passen Sie die folgenden kommunikationsbezogenen Umrichterparameter Ihrer Anwendung an: • Datenübertragungsrate (p2020), sofern ≠ 38400 Bit/s (Achten Sie auf Übereinstimmung mit Parameter BAUD der Kommunikationsanweisung USS_Port_Scan.) • Anzahl PZD (p2022), sofern ≠ 2 (Achten Sie auf Übereinstimmung mit Parameter PZD_LEN der Kommunikationsanweisung USS_Drive_Control.) • Anzahl PKW (p2023) = 4 (Ändern Sie den über Makro 21 defaultmäßig auf "variabel" (127) eingestellten Wert auf "4" (die Anweisungen USS_Read_Param und USS_Write_Param erfordern dies.) • Feldbus SS Überwachungszeit [ms] (p2040)	Kap. 6.2.2.2	Kap. 8.4.2.1
4	Legen Sie die Quelle für den Drehzahlsollwert fest. • n_soll Ausw (p1000[0]) = 6 (Der Drehzahlsollwert wird über den USS-Bus bereitgestellt.)		

Schritt	Anweisung	Betriebsanleitung	
		G120 [1]	G120C [2]
5	Stellen Sie die Bezugsgröße für Drehzahl und Frequenz ein. • n_Bezug f_Bezug (p2000) = (6.00 min^{-1} bis 210000.00 min^{-1}) (Alle relativ angegebenen Drehzahlen oder Frequenzen beziehen sich auf diese Bezugsgröße. Die Bezugsgröße entspricht 100 % bzw. 4000_{hex} (Wort) oder 4000 0000_{hex} (Doppelwort). Dabei gilt: $f_{Bezugsgröße}$ (in Hz) = $n_{Bezugsgröße}$ (in ((min^{-1}) / 60) x Polpaarzahl))		
6	Übertragen Sie die Parameter in den nichtflüchtigen Speicher. • Par speichern (p0971) = 1		

1) G120 (http://support.automation.siemens.com/WW/view/de/62089662)

2) G120(C) (http://support.automation.siemens.com/WW/view/de/61462568)

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/ maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüberschreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichenverzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endesequenz"	-
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertragung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#8088	Längenangabe größer als der im Empfangspuffer eingestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehlercode auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wählen Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).

Fehlercode	Beschreibung	Abhilfe
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).

Fehlercode	Beschreibung	Abhilfe
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).

Fehlercode	Beschreibung	Abhilfe
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		

Fehlercode	Beschreibung	Abhilfe
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		

Fehlercode	Beschreibung	Abhilfe
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)

Fehlercode	Beschreibung	Abhilfe
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.

Fehlercode	Beschreibung	Abhilfe
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)

Fehlercode	Beschreibung	Abhilfe
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem

Fehlercode	Beschreibung	Abhilfe
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.

Fehlercode	Beschreibung	Abhilfe
16#8828 [1]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8428 [1]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %

Fehlercode	Beschreibung	Abhilfe
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

MODBUS (RTU)

Überblick über die Modbus RTU-Kommunikation

Modbus RTU-Kommunikation

Modbus RTU (Remote Terminal Unit) ist ein Standardprotokoll für die Kommunikation im Netzwerk und verwendet die RS232- oder RS422/485-Verbindung für die serielle Datenübertragung zwischen Modbus-Geräten im Netzwerk.

Modbus RTU nutzt ein Master/Slave-Netzwerk, in dem die gesamte Kommunikation von einem einzigen Master-Gerät ausgelöst wird, während die Slaves lediglich auf die Anforderung des Masters reagieren können. Der Master sendet eine Anforderung an eine Slave-Adresse und nur der Slave mit dieser Slave-Adresse antwortet auf den Befehl.

Sonderfall: Modbus-Slaveadresse 0 sendet ein Broadcast-Telegramm an alle Slaves (ohne Slave-Antwort).

Modbus-Funktionscodes

- Eine CPU, die als Modbus RTU-Master betrieben wird, kann Daten und E/A-Zustände in einem über eine Kommunikationsverbindung verbundenen Modbus RTU-Slave lesen und schreiben.
- Eine CPU, die als Modbus RTU-Slave betrieben wird, ermöglicht es einem über eine Kommunikationsverbindung verbundenen Modbus RTU-Master, Daten und E/A-Zustände in der eigenen CPU zu lesen und zu schreiben.

Tabelle 4-159 Funktionen zum Lesen von Daten: Dezentrale E/A und Programmdaten lesen

Modbus-Funktionscode	Funktionen zum Lesen von Daten aus dem Slave (Server) - Standard-adressierung
01	Ausgangsbits lesen: 1 bis 2000/1992[1)] Bits pro Anforderung
02	Eingangsbits lesen: 1 bis 2000/1992[1)] Bits pro Anforderung
03	Halteregister lesen: 1 bis 125/124[1)] Worte pro Anforderung
04	Eingangswörter lesen: 1 bis 125/124[1)] Worte pro Anforderung

1) bei erweiterter Adressierung

Tabelle 4-160 Funktionen zum Schreiben von Daten: Dezentrale E/A schreiben und Programmdaten ändern

Modbus-Funktionscode	Funktionen zum Schreiben von Daten in den Slave (Server) - Standard-adressierung
05	Ein Ausgangsbit schreiben: 1 Bit pro Anforderung
06	Ein Halteregister schreiben: 1 Wort pro Anforderung
15	Ein oder mehrere Ausgangsbits schreiben: 1 bis 1960 Bits pro Anforderung
16	Ein oder mehrere Halteregister schreiben: 1 bis 122 Worte pro Anforderung

- Die Modbus-Funktionscodes 08 und 11 bieten Diagnosemöglichkeiten für die Kommunikation mit dem Slave-Gerät.
- Modbus-Slaveadresse 0 sendet ein Broadcast-Telegramm an alle Slaves (ohne Slave-Antwort; für Funktionscodes 5, 6, 15, 16).

Tabelle 4-161 Stationsadressen im Modbus-Netzwerk

Station		Adresse
RTU-Station	Standardstationsadresse	1 bis 247 sowie 0 für Broadcast
	Erweiterte Stationsadresse	1 bis 65535 sowie 0 für Broadcast

Modbus-Speicheradressen

Die tatsächlich verfügbare Anzahl von Modbus-Speicheradressen (Ein-/Ausgangsadresen) richtet sich nach der CPU-Variante, nach dem verfügbaren Arbeitsspeicher.

Modbus RTU-Anweisungen in Ihrem Programm

- Modbus_Comm_Load: Eine Ausführung von Modbus_Comm_Load ist erforderlich, um PtP-Parameter wie Datenübertragungsgeschwindigkeit, Parität und Datenflusskontrolle einzurichten. Nachdem das Kommunikationsmodul für das Modbus RTU-Protokoll konfiguriert ist, kann er nur von der Anweisung Modbus_Master oder der Anweisung Modbus_Slave verwendet werden.
- Modbus_Master: Mit der Modbus-Master-Anweisung kann die CPU als Modbus-RTU-Mastergerät für die Kommunikation mit einem oder mehreren Modbus-Slavegeräten eingesetzt werden.
- Modbus_Slave: Mit der Modbus-Slave-Anweisung kann die CPU als Modbus-RTU-Slavegerät für die Kommunikation mit einem Modbus-Mastergerät eingesetzt werden.

Modbus_Comm_Load: Kommunikationsmodul für Modbus konfigurieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Modbus_Comm_Load konfiguriert ein Kommunikationsmodul für die Kommunikation über das Modbus-RTU-Protokoll). Wenn Sie die Anweisung Modbus_Comm_Load in Ihr Programm einfügen, wird automatisch ein Instanz-Datenbaustein zugewiesen.

Die Konfigurationsänderungen von Modbus_Comm_Load werden nicht in der CPU sondern auf dem CM gespeichert. Bei Spannungswiederkehr und Ziehen/Stecken wird das CM mit den

in der Gerätekonfiguration gespeicherten Daten parametriert. In diesen Fällen muss die Anweisung Modbus_Comm_Load erneut aufgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	Port	Laddr	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BAUD	IN	UDInt	DWord	9600	Auswahl der Datenübertragungsgeschwindigkeit Zulässige Werte sind: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 bit/s.
PARITY	IN	UInt	Word	0	Auswahl der Parität: • 0 – Keine • 1 – Ungerade • 2 – Gerade
FLOW_CTRL	IN	UInt	Word	0	Auswahl der Flusskontrolle: • 0 – (Standard) Keine Flusskontrolle • 1 – Hardware-Flusskontrolle mit RTS immer EIN (gilt nicht bei RS422/485-CMs) • 2 – Hardware-Flusskontrolle mit RTS geschaltet (gilt nicht bei RS422/485-CMs)
RTS_ON_DLY	IN	UInt	Word	0	Auswahl RTS-Einschaltverzögerung: • 0 – Keine Verzögerung von "RTS aktiv", bis das erste Zeichen des Telegramms gesendet wird. • 1 bis 65535 – Verzögerung in Millisekunden von "RTS aktiv", bis das erste Zeichen des Telegramms gesendet wird (gilt nicht bei RS422/485-CMs). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RTS_OFF_DLY	IN	UInt	Word	0	Auswahl RTS-Ausschaltverzögerung: • 0 – Keine Verzögerung nach der Übertragung des letzten Zeichens bis "RTS inaktiv" • 1 bis 65535 – Verzögerung in Millisekunden von der Übertragung des letzten Zeichens bis "RTS inaktiv" (gilt nicht bei RS422/485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.
RESP_TO	IN	UInt	Word	1000	Antwort-Timeout: 5 ms bis 65535 ms - Zeit in Millisekunden, die Modbus_Master auf eine Antwort vom Slave wartet. Wenn der Slave während dieses Zeitraums nicht antwortet, wiederholt Modbus_Master die Anforderung oder beendet die Anforderung mit einem Fehler, wenn die angegebene Anzahl von Wiederholungen (siehe unten, Parameter RETRIES) gesendet wurde.
MB_DB	IN/OUT	MB_BASE		-	Ein Verweis auf den Instanz-Datenbaustein der Anweisungen Modbus_Master oder Modbus_Slave. Der Parameter MB_DB muss mit dem (statischen und daher an der Anweisung unsichtbaren) Parameter MB_DB der Anweisung Modbus_Master oder Modbus_Slave verschaltet werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Comm_Load Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DONE	OUT	Bool		FALSE	Das Bit DONE ist einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde.
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Der Fehlercode im Parameter STATUS ist nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 4203))

Modbus_Comm_Load wird ausgeführt, um einen Port für das Modbus-RTU-Protokoll zu konfigurieren. Nachdem ein Port für das Modbus-RTU-Protokoll konfiguriert ist, kann er nur von den Anweisungen Modbus_Master oder Modbus_Slave verwendet werden.

Für die Konfiguration jedes Kommunikationsports, der für die Modbus-Kommunikation eingesetzt wird, muss eine Ausführung von Modbus_Comm_Load genutzt werden. Sie müssen jedem Port, den Sie nutzen, einen eindeutigen Instanz-DB Modbus_Comm_Load zuweisen. Führen Sie Modbus_Comm_Load nur dann erneut aus, wenn Kommunikationsparameter wie Datenübertragungsgeschwindigkeit oder Parität geändert werden müssen sowie ggf. nach Netzwiederkehr.

Wenn Sie z. B. Modbus_Master oder Modbus_Slave in Ihr Programm einfügen, wird der Anweisung ein Instanz-Datenbaustein zugewiesen. Sie müssen den Parameter MB_DB der Anweisung Modbus_Comm_Load mit dem Parameter MB_DB der Anweisung Modbus_Master bzw. Modbus_Slave verschalten.

Modbus_Comm_Load Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von Modbus_Comm_Load, die in Ihrem Programm verwendet werden können.

Tabelle 4-162 Statische Variablen im Instanz-DB

Variable	Datentyp		Standard	Beschreibung
	S7-1200/1500	S7-300/400/WinAC		
ICHAR_GAP	Word		0	Maximale Zeichenverzugszeit zwischen den Zeichen. Dieser Parameter wird in Millisekunden angegeben und dient dazu, den erwarteten Zeitraum zwischen den empfangenen Zeichen zu erhöhen. Die entsprechende Anzahl Bitzeiten für diesen Parameter wird zum Modbus-Standardwert von 35 Bitzeiten (3,5 Zeichenzeiten) addiert.
RETRIES	Word		2	Anzahl der Wiederholungsversuche, die der Master durchführt, bevor der Fehlercode 0x80C8 für "Keine Antwort" zurückgegeben wird.
EN_SUPPLY_VOLT	Bool		0	Diagnose für fehlende Versorgungsspannung L+ einschalten
MODE	USInt	Byte	0	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master, CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave, CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1)]
LINE_PRE	USInt	Byte	0	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung): Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V: Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.

Variable	Datentyp		Stan-dard	Beschreibung
	S7-1200/1500	S7-300/400/WinAC		
BRK_DET	USInt	Byte	0	Break-Erkennung Zulässig sind: • 0 = Break-Erkennung deaktiviert • 1 = Break-Erkennung aktiviert
EN_DI-AG_ALARM	Bool		0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
STOP_BITS	USINT	Byte	1	Anzahl Stopp-Bits; • 1 = 1 Stopp-Bit, • 2 = 2 Stopp-Bits, • 0, 3 bis 255 = reserviert

[1] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Anweisungsversionen

Die Version 3.1 ist funktional identisch zur Version 3.0 und wurde aufgrund interner Maßnahmen erhöht.

Modbus_Master: Als Modbus-Master kommunizieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Modbus_Master kommuniziert als Modbus-Master über einen Port, der mit der Anweisung Modbus_Comm_Load konfiguriert wurde. Wenn Sie die Anweisung Modbus_Master in Ihr Programm einfügen, wird automatisch ein Instanz-Datenbaustein zugewiesen. Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit dem (statischen) Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	FALSE = Keine Anforderung TRUE = Anforderung, Daten an den Modbus-Slave zu senden
MB_ADDR	IN	UInt	Word	-	Modbus-RTU-Stationsadresse: Standard-Adressierungsbereich (1 bis 247 sowie 0 für Broadcast) Erweiterter Adressierungsbereich (1 bis 65535 sowie 0 für Broadcast) Der Wert 0 ist für den Broadcast des Telegramms an alle Modbus Slaves reserviert. Für den Broadcast werden nur die Modbus-Funktionscodes 05, 06, 15 und 16 unterstützt.
MODE	IN	USInt	Byte	0	Auswahl Modus: Gibt die Art der Anforderung an (Lesen, Schreiben oder Diagnose). Weitere Informationen finden Sie in der Tabelle der Modbus-Funktionen unten.
DATA_ADDR	IN	UDInt	DWord	0	Anfangsadresse im Slave: Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll. Die gültigen Adressen finden Sie in der Tabelle der Modbus-Funktionen unten.
DATA_LEN	IN	UInt	Word	0	Datenlänge: Gibt die Anzahl der Bits oder Wörter an, auf die diese Anforderung zugreifen soll. Die gültigen Längen finden Sie in der Tabelle der Modbus-Funktionen unten.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Master Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DATA_PTR	IN/OUT	Variant	Any	-	Daten-Pointer: Zeigt auf die Merker- oder DB-Adresse für die zu schreibenden oder zu lesenden Daten. Ab Anweisungsversion V3.0: Der Parameter darf auf einen optimierten Speicherbereich zeigen. Im optimierten Speicherbereich ist ein Single-Element oder ein Array erlaubt mit folgenden Datentypen: Bool, Byte, Char, Word, Int, DWord, DInt, Real, USInt, UInt, UDInt, SInt, WChar. Jeder andere Datentyp führt zur Fehlermeldung 16#818C.
DONE	OUT	Bool		FALSE	Das Bit DONE ist einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde.
BUSY	OUT	Bool		-	• FALSE – Keine Auftrag für Modbus_Master aktiv • TRUE – Auftrag für Modbus_Master in Bearbeitung

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Der Fehlercode im Parameter STATUS ist nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 4203))

Variablen im Datenbaustein des Modbus-Master

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von Modbus_Master, die in Ihrem Programm verwendet werden können.

Tabelle 4-163 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
Blocked_Proc_Timeout	Real	3.0	Zeitdauer (in Sekunden), die auf eine blockierte Modbus-Master-Instanz gewartet werden soll, bevor diese Instanz als AKTIV entfernt wird. Dies kann beispielsweise vorkommen, wenn eine Master-Anforderung ausgegeben wurde und das Programm dann aufhört, die Masterfunktion aufzurufen, bevor es die Anforderung vollständig beendet hat. Der Zeitwert muss größer als 0 und kleiner als 55 Sekunden sein, ansonsten tritt ein Fehler auf. Siehe hierzu auch Abschnitte "Regeln für die Kommunikation des Modbus-Master" und "Aufruf der Anweisung Modbus_Master mit verschienenen Parametereinstellungen".
Extended_Addressing	Bool	FALSE	Konfiguriert die Slave-Stationsadresse als einzelnes oder doppeltes Byte. • FALSE = Ein-Byte-Adresse; 0 bis 247 • TRUE = Zwei-Byte-Adresse (entspricht der erweiterten Adressierung); 0 bis 65535

Variable	Datentyp	Standard	Beschreibung
Compatibility_Mode [1]	Bool	FALSE	Kompatibilitätsmodus mit CP 341, CP 441-2 und ET 200S 1SI mit Treiber für Modbus RTU und mit ET 200S 1SI für Modbus. Der Standardwert ist 0. • FALSE = nach Modbus-Spezifikation, nicht kompatibel • TRUE = kompatibel – Bei FC1 und FC2 gilt: Die aus dem Empfangstelegramm gelesenen Daten werden wortweise in den adressierten Speicher der CPU geschrieben und dabei byteweise ausgetauscht. Falls die Anzahl zu übertragender Bits nicht ein Vielfaches von 16 ist, werden die nicht relevanten Bits im letzten Wort auf Null gesetzt. – Bei FC15 gilt: Die zu übertragenden Worte werden wortweise aus dem adressierten Speicher gelesen und byteweise in das Sendetelegramm geschrieben. Falls die Anzahl zu übertragender Bits nicht ein Vielfaches von 8 ist, werden die nicht relevanten Bits im letzten Byte unverändert aus dem adressierten Speicher gelesen und im Sendetelegram eingetragen.
MB_DB	MB_BA-SE	-	Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit diesem Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

[1] Die PtP-Kommunikationsmodule verhalten sich gemäß der Modbus Spezifikation. Um ein Verhalten wie bei CP 341, CP 441-2 und ET 200SP 1SI für Modbus bisher üblich zu erhalten, verwenden Sie den Parameter "Compatibility_Mode".

Ihr Programm kann in die Variablen Blocked_Proc_Timeout und Extended_Addressing Werte schreiben, um die Modbus-Master-Operationen zu steuern.

Regeln für die Kommunikation des Modbus-Master

- Modbus_Comm_Load muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung Modbus_Master mit diesem Port kommunizieren kann.
- Wenn ein Port als Modbus Master verwendet werden soll, darf dieser Port nicht von Modbus_Slave verwendet werden. Eine oder mehrere Instanzen von Modbus_Master [1] können mit diesem Port verwendet werden. Doch alle Ausführungen von Modbus_Master müssen den gleichen Instanz-DB für den Port verwenden.
- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss die Anweisung Modbus_Master auf abgeschlossene Aufträge abfragen (DONE, ERROR).
- Es wird empfohlen, alle Ausführungen von Modbus_Master für einen bestimmten Port aus einem Programmzyklus-OB aufzurufen. Modbus-Master-Anweisungen können nur in jeweils einem Programmzyklus oder in jeweils einer zyklischen/zeitgesteuerten Bearbeitungsebene ausgeführt werden. Sie dürfen nicht in verschiedenen Bearbeitungsebenen bearbeitet werden. Die Vorrangunterbrechung einer Modbus-Master-Anweisung durch eine andere Modbus-Master-Anweisung in einer Bearbeitungsebene mit höherer Priorität führt zu nicht ordnungsgemäßem Betrieb. Die Modbus-Master-Anweisungen dürfen nicht in Anlauf-, Diagnose- oder Zeitfehlerebenen bearbeitet werden.

[1)] Mit "Instanz von Modbus Master" ist hier ein Aufruf der Anweisung Modbus_Master mit gleicher Verschaltung mit einer Anweisung Modbus_Comm_Load und gleicher Einstellung für die Parameter MB_ADDR, MODE, DATA_ADDR und DATA_LEN gemeint.

Beispiel.

Modbus_Master wird aufgerufen mit MODE=0 und DATA_ADDR=10

Dieser Auftrag ist nun aktiv, bis er mit DONE=1 oder ERROR=1 beendet wird oder bis die am Parameter Blocked_Proc_Timeout parametrierte Zeitüberwachung abgelaufen ist. Wird nach Ablauf der Zeitüberwachung ein neuer Auftrag gestartet, bevor der vorherige Auftrag beendet ist, so wird der vorherige Auftrag ohne Fehlermeldung abgebrochen.

Falls nun, während dieser Auftrag läuft, ein zweiter Aufruf der Anweisung mit den gleichen Instanzdaten, aber anderen Einstellungen der Parameter MODE und DATA_ADDR erfolgt, so wird dieser zweite Aufruf mit ERROR=1 und STATUS=8200 beendet.

Aufruf der Anweisung Modbus_Master mit verschienenen Parametereinstellungen

Sollen mehrere Aufrufe der Anweisung Modbus_Master mit unterschiedlichen Einstellungen für MB_ADDR, MODE, DATA_ADDR oder DATA_LEN im Ihrem Programm platziert werden, müssen Sie sicherstellen, dass zu einer Zeit nur einer dieser Aufrufe aktiv ist, anderenfalls wird die Fehlermeldung 16#8200 (die Schnittstelle ist durch eine laufende Anfrage belegt) ausgegeben.

Falls ein Aufruf nicht fertig bearbeitet werden kann, wird die Zeitüberwachung über den Parameter Blocked_Proc_Timeout aktiv und beendet den laufenden Auftrag.

Parameter REQ

FALSE = Keine Anforderung; TRUE = Anforderung, Daten an den Modbus-Slave zu senden

Freigabe der angeforderten Übertragung. Dadurch wird der Inhalt des Puffers zur Punkt-zu-Punkt-Kommunikationsschnittstelle übertragen.

Über die Parameter DATA_ADDR und MODE wählen Sie den Modbus-Funktionscode aus.

DATA_ADDR (Modbus-Anfangsadresse im Slave): Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll.

Die Anweisung Modbus_Master nutzt statt eines Funktionscodeeingangs einen Eingang MODE. Die Kombination aus MODE und DATA_ADDR legt den Funktionscode fest, der im eigentlichen Modbus-Telegramm verwendet wird. Die folgende Tabelle zeigt die Zuordnung zwischen dem Parameter MODE, dem Modbus-Funktionscode und dem Modbus-Adressbereich in DATA_ADDR.

Tabelle 4-164 Modbus-Funktionen

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
0				Bits pro Anforderung			01	Ausgangsbits lesen:		
	1	bis	9999	1	bis	2000/1992 [1]		0	bis	9998

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
0				Bits pro Anforderung			02	Eingangsbits lesen:		
	10001	bis	19999	1	bis	2000/1992 [1]		0	bis	9998
0				Wörter pro Anforderung			03	Halteregister lesen:		
	40001	bis	49999	1	bis	125/124 [1]		0	bis	9998
	400001	bis	465535	1	bis	125/124 [1]		0	bis	65534
0				Wörter pro Anforderung			04	Eingangswörter lesen:		
	30001	bis	39999	1	bis	125/124 [1]		0	bis	9998
1				Bit pro Anforderung			05	Ein Ausgangsbit schreiben:		
	1	bis	9999	1				0	bis	9998
1				1 Wort pro Anforderung			06	Ein Halteregister schreiben:		
	40001	bis	49999	1				0	bis	9998
	400001	bis	465535	1				0	bis	65524
1				Bits pro Anforderung			15	Mehrere Ausgangsbits schreiben:		
	1	bis	9999	2	bis	1968/1960 [1]		0	bis	9998
1				Wörter pro Anforderung			16	Mehrere Halteregister schreiben:		
	40001	bis	49999	2	bis	123/122		0	bis	9998
	400001	bis	465534	2	bis	123/122 [1]		0	bis	65534
2 [2]				Bits pro Anforderung			15	Ein oder mehrere Ausgangsbits schreiben:		
	1	bis	9999	1	bis	1968/1960 [1]		0	bis	9998
2 [2]				Wörter pro Anforderung			16	Ein oder mehrere Halteregister schreiben:		
	40001	bis	49999	1	bis	123		0	bis	9998
	400001	bis	465535	1	bis	122 [1]		0	bis	65534
11	Beide Operanden DATA_ADDR und DATA_LEN von Modbus_Master werden bei dieser Funktion ignoriert.						11	Statuswort und Ereigniszähler der Slave-Kommunikation lesen. Das Statuswort zeigt beschäftigt an (0 – nicht beschäftigt, 0xFFFF - beschäftigt). Der Ereigniszähler wird bei jeder erfolgreichen Abarbeitung eines Telegramms inkrementiert.		
80				1 Wort pro Anforderung			08	Slave-Zustand über Daten-Diagnosecode 0x0000 prüfen (Prüfschleifentest – Slave gibt ein Echo der Anforderung zurück)		
	-			1				-		
81				1 Wort pro Anforderung			08	Slave-Ereigniszähler über Daten-Diagnosecode 0x000A zurücksetzen		
	-			1				-		

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
104 [3]				Wörter pro Anforderung			04	Eingangswörter lesen		
	0	bis	65535	1	bis	125/124 [1]		0	bis	65535
3 bis 10, 12 bis 79, 82 bis 103, 105 bis 255	-			-				Reserviert		

[1] Bei der erweiterten Adressierung, siehe Parameter Extended_Adressing, ist die maximale Datenlänge je nach Datentyp der Funktion um 1 Byte oder 1 Wort geringer.

[2] MODE 2 erlaubt das Schreiben von einem oder mehreren Ausgangsbits und einem oder mehreren Holding Register unter Verwendung der Modbus-Funktionen 15 und 16.
MODE 1 verwendet die Modbus-Funktionen 5 und 6, um 1 Ausgangsbit und 1 Holding Register zu schreiben und die Modbus-Funktionen 15 und 16, um mehrere Ausgangsbits und mehrere Holding Register zu schreiben.

[3] Bei S7-300/400/WinAC gilt: Reserviert

Parameter DATA_PTR

Der Parameter DATA_PTR zeigt auf die DB- oder M-Adresse, in die geschrieben oder aus der gelesen wird. Wenn Sie einen Datenbaustein verwenden, müssen Sie einen globalen Datenbaustein anlegen, der den Datenspeicher für Lese- und Schreibvorgänge auf Modbus-Slaves bereitstellt.

Hinweis

S7-1200/1500 - Der über DATA_PTR adressierte Datenbaustein muss die direkte Adressierung zulassen

Der Datenbaustein muss die direkte (absolute) und die symbolische Adressierung zulassen.

Hinweis

Verwendung von Funktionscode 5

Mit Funktionscode 5 kann ein einzelnes Bit gesetzt oder gelöscht werden.

Zum Setzen eines Bits muss im ersten Wort des über DATA_PTR adressierten DB- oder Merkerbereichs der Wert "16#FF00" vorgegeben werden.

- Bei S7-1200 kann zum Setzen eines Bits auch der Wert "16#0100" vorgegeben werden.
- Zum Rücksetzen eines Bits muss im ersten Wort des über DATA_PTR adressierten DB- oder Merkerbereichs der Wert "16#0000" vorgegeben werden.

Alle anderen Werte werden mit der ERROR = TRUE und STATUS = 16#8384 abgewiesen.

Datenbausteinstrukturen für den Parameter DATA_PTR

- Diese Datentypen gelten für das Lesen von Wörtern des Modbus-Adressbereichs (DATA_PTR) 30001 bis 39999, 40001 bis 49999 und 400001 bis 465535 sowie für das Schreiben von Wörtern des Modbus-Adressbereichs (Parameter DATA_PTR) 40001 bis 49999 und 400001 bis 465535.
  - Standard-Array der Datentypen WORD, UINT oder INT
  - Benannte Struktur vom Typ WORD, UINT oder INT, bei der jedes Element einen eindeutigen Namen und einen 16-Bit-Datentyp hat.
  - Benannte komplexe Struktur, bei der jedes Element einen eindeutigen Namen und einen 16- oder 32-Bit-Datentyp hat.
- Zum Lesen und Schreiben von Bits für den Modbus-Adressbereich (Parameter DATA_PTR) 00001 bis 09999 und zum Lesen von Bits von 10001 bis 19999.
  - Standardfeld aus Boole'schen Datentypen.
  - Benannte Boole'sche Struktur aus eindeutig benannten Boole'schen Variablen.
- Es ist zwar nicht erforderlich, doch empfehlenswert, dass jede Anweisung Modbus_Master einen eigenen getrennten Speicherbereich hat. Grund hierfür ist, dass die Möglichkeit der Datenzerstörung größer ist, wenn mehrere Anweisungen Modbus_Master in demselben Speicherbereich lesen und schreiben.
- Es ist nicht erforderlich, dass die Datenbereiche für DATA_PTR in demselben globalen Datenbaustein liegen. Sie können einen Datenbaustein mit mehreren Bereichen für Modbus-Lesevorgänge, einen Datenbaustein für Modbus-Schreibvorgänge oder einen Datenbaustein für jede Slave-Station anlegen.

Anweisungsversionen

Die Version 3.0 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

Siehe auch

Modbus_Slave: Als Modbus-Slave kommunizieren (Seite 4187)

Modbus_Slave

Modbus_Slave: Als Modbus-Slave kommunizieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Modbus_Slave kann Ihr Programm über einen CM (RS422/485 oder RS232) als Modbus-Slave kommunizieren. STEP 7 erstellt automatisch einen Instanz-DB, wenn Sie die Anweisung einfügen. Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit dem (statischen) Parameter MB_DB der Anweisung Modbus_Slave verschaltet werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
MB_ADDR	IN	UInt	Word	-	Stationsadresse des Modbus-Slaves: Standard-Adressierungsbereich (1 bis 247) Erweiterter Adressierungsbereich (0 bis 65535) Hinweis: 0 ist die Broadcast-Adresse
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Slave Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
MB_HOLD_REG	IN/OUT	Variant	Any	-	Pointer auf den Modbus-Halteregister-DB: Bei dem Modbus-Halteregister kann es sich um den Speicherbereich der Merker oder um einen Datenbaustein handeln. Ab Anweisungsversion V4.0: Der Parameter muss auf einen Speicherbereich zeigen, der eine Länge von mindestens 16 Bit hat. Eine geringere Länge führt zur Fehlermeldung 16#8187. Dies betrifft Single-Elemente, Arrays, STRUCTs und UDTs. Z.B. führt ein Single Bool oder ein Array, das aus weniger als 16 Bool-Elementen besteht, zu der Fehlermeldung. Wenn die Länge kein Vielfaches von 16 Bit ist, können die Restbits am Ende des Speicherbereichs von der Anweisung Modbus_Slave weder gelesen noch geschrieben werden. Der Parameter darf auf einen optimierten Speicherbereich zeigen. Im optimierten Speicherbereich ist ein Single-Element oder ein Array erlaubt mit folgenden Datentypen: Bool, Byte, Char, Word, Int, DWord, DInt, Real, USInt, UInt, UDInt, SInt, WChar. Jeder andere Datentyp führt zur Fehlermeldung 16#818C.
NDR	OUT	Bool		FALSE	Neue Daten bereit: • FALSE – Keine neuen Daten • TRUE – Gibt an, dass neue Daten vom Modbus-Master geschrieben wurden Das Bit NDR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung ohne Fehler beendet wurde.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
DR	OUT	Bool		FALSE	Daten lesen: • FALSE – Keine Daten gelesen • TRUE – Gibt an, dass die Anweisung vom Modbus-Master empfangene Daten im Zielbereich abgelegt hat. Das Bit DR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung ohne Fehler beendet wurde.
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Wenn die Ausführung mit einem Fehler beendet wird, ist der Fehlercode im Parameter STATUS nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 4203))

Die Funktionscodes der Modbus-Kommunikation (1, 2, 4, 5 und 15) können Bits und Wörter direkt im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge der CPU lesen und schreiben. Bei diesen Funktionscodes muss der Parameter MB_HOLD_REG als Datentyp größer als ein Byte definiert werden. Die folgende Tabelle zeigt die Beispielzuordnung der Modbus-Adressen zum Prozessabbild in der CPU.

Tabelle 4-165 Zuordnung der Modbus-Adressen zum Prozessabbild

Modbus-Funktionen						S7-1200			
Code	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
01	Bits lesen	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7
02	Bits lesen	Eingang	0	bis	8191	Prozessabbild der Eingänge	E0.0	bis	E1023.7
04	Wörter lesen	Eingang	0	bis	511	Prozessabbild der Eingänge	EW0	bis	EW1022
05	Bit schreiben	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7
15	Bits schreiben	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7

Tabelle 4-166 Zuordnung der Modbus-Adressen zum Prozessabbild

Modbus-Funktionen						S7-1500 / S7-300 / S7-400			
Funktionscode	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
01	Bits lesen	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6
02	Bits lesen	Eingang	0	bis	9998	Prozessabbild der Eingänge	E0.0	bis	E1249.6
04	Wörter lesen	Eingang	0	bis	9998	Prozessabbild der Eingänge	EW0	bis	EW19996

Modbus-Funktionen						S7-1500 / S7-300 / S7-400			
Funktionscode	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
05	Bit schreiben	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6
15	Bits schreiben	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6

Hinweis

Der verfügbare Adressbereich kann, abhängig vom Speicherausbau der CPU, auch kleiner sein.

Die Funktionscodes der Modbus-Kommunikation (3, 6, 16) nutzen ein Modbus-Halteregister, bei dem es sich um einen Adressbereich im Speicherbereich der Merker oder um einen Datenbaustein handeln kann. Der Typ des Halteregisters wird vom Parameter MB_HOLD_REG der Anweisung Modbus_Slave angegeben.

Hinweis

S7-1200/1500 - Typ des Datenbausteins MB_HOLD_REG

Ein Datenbaustein mit Modbus-Halteregister muss die direkte (absolute) und die symbolische Adressierung zulassen.

Tabelle 4-167 Diagnosefunktionen

Modbus-Diagnosefunktionen von Modbus_Slave der S7-1200		
Funktionscodes	Teilfunktion	Beschreibung
08	0000H	Echotest Abfragedaten ausgeben: Die Anweisung Modbus_Slave gibt einem Modbus-Master das Echo eines empfangenen Datenworts zurück.
08	000AH	Kommunikationsereigniszähler löschen: Die Anweisung Modbus_Slave löscht den Kommunikationsereigniszähler, der für Modbus-Funktion 11 verwendet wird.
11		Kommunikationsereigniszähler abrufen: Die Anweisung Modbus_Slave nutzt einen internen Kommunikationsereigniszähler, um die Anzahl erfolgreicher Modbus-Lese- und Modbus-Schreibanforderungen, die an den Modbus-Slave gesendet werden, zu erfassen. Der Zähler wird bei Funktion 8, Funktion 11 und Broadcast-Anforderungen nicht hochgezählt. Außerdem wird bei Anforderungen, die zu Kommunikationsfehlern führen (z. B. Paritäts- oder CRC-Fehler), nicht hochgezählt.

Die Anweisung Modbus_Slave unterstützt Broadcast-Schreibanforderungen von Modbus-Mastern, solange die Anforderungen den Zugriff auf gültige Adressen umfassen. Für Funktionscodes, die bei der Broadcast-Funktion nicht unterstützt werden, erzeugt Modbus_Slave den Fehlercode 16#8188.

Variablen des Modbus-Slave in der Anweisungsversion V3.0

Diese Tabelle zeigt die öffentlichen statischen Variablen im Instanz-Datenbaustein von Modbus_Slave, die in Ihrem Programm verwendet werden können.

Tabelle 4-168 Variablen des Modbus-Slave

Variable	Datentyp	Standard	Beschreibung
HR_Start_Offset	Word	0	Gibt die Anfangsadresse der Modbus-Halteregisters an (Standard = 0)
QB_Start	Word	0	Anfangsadresse des zulässigen schreibbaren Adressierungsbereichs der Ausgänge (Byte 0 bis 65535) Hinweis: Die Variable ist für S7-300, S7-400 und WinAC nicht verfügbar.
QB_Count	Word	0xFFFF	Anzahl der durch den Modbus-Master schreibbaren Ausgangs-Bytes. Hinweis: Die Variable ist für S7-300, S7-400 und WinAC nicht verfügbar.
Extended_Addressing	Bool	FALSE	Erweiterte Adressierung, konfiguriert die Slave-Adressierung als einzelnes oder doppeltes Byte (FALSE = einzelne Byte-Adresse, TRUE = doppelte Byte-Adresse)
Request_Count	Word	0	Die Anzahl aller von diesem Slave empfangenen Anforderungen
Slave_Message_Count	Word	0	Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen
Bad_CRC_Count	Word	0	Die Anzahl der empfangenen Anforderungen, die einen CRC-Fehler aufweisen
Broadcast_Count	Word	0	Die Anzahl der empfangenen Broadcast-Anforderungen
Exception_Count	Word	0	Modbus-spezifische Fehler, die mit einer Exception an den Master quittiert werden
Success_Count	Word	0	Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen ohne Protokollfehler
MB_DB	MB_BASE	-	Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit diesem Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

Ihr Programm kann in die Variablen HR_Start_Offset und Extended_Addressing Werte schreiben und die Modbus-Slave-Operationen steuern. Die anderen Variablen können gelesen werden, um den Modbus-Zustand zu überwachen.

Regeln für die Modbus-Slave-Kommunikation

- Modbus_Comm_Load muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung Modbus_Slave über diesen Port kommunizieren kann.
- Wenn ein Port als Slave auf einen Modbus-Master reagieren soll, darf dieser Port nicht mit der Anweisung Modbus_Master programmiert werden.
- Nur eine Instanz von Modbus_Slave kann mit einem bestimmten Port verwendet werden, andernfalls kann unerwartetes Verhalten auftreten.

- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss den Kommunikationsprozess durch Abfrage der Anweisung Modbus_Slave auf abgeschlossene Sende- und Empfangsvorgänge steuern.
- Die Anweisung Modbus_Slave muss regelmäßig mit einer Häufigkeit ausgeführt werden, die eine zeitnahe Antwort auf eingehende Anforderungen eines Modbus-Masters ermöglicht. Es wird empfohlen, Modbus_Slave in jedem Zyklus aus einem Programmzyklus-OB auszuführen. Modbus_Slave kann aus einem Weckalarm-OB ausgeführt werden, doch ist dies nicht empfehlenswert, weil übermäßige Zeitverzögerungen im Interruptprogramm die Ausführung anderer Interruptprogramme temporär blockieren können.

Zeitsteuerung des Modbus-Signals

Modbus_Slave muss regelmäßig ausgeführt werden, um jede Anforderung des Modbus-Masters zu empfangen und entsprechend zu antworten. Die Häufigkeit der Ausführung von Modbus_Slave richtet sich nach dem vom Modbus-Master vorgegebenen Timeout-Wert für die Antwort. Dies ist in der nachstehenden Abbildung dargestellt.

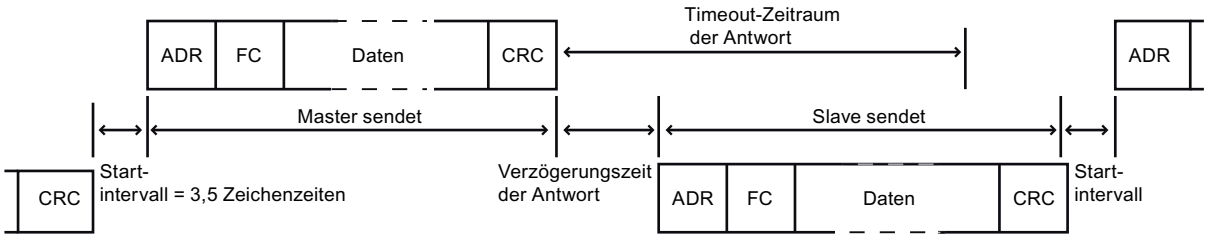

Der Timeout-Zeitraum der Antwort (RESP_TO) ist die Zeitdauer, die ein Modbus-Master auf den Beginn einer Antwort von einem Modbus-Slave wartet. Dieser Zeitraum wird nicht vom Modbus-Protokoll definiert, sondern von einem Parameter der Anweisung Modbus_Comm_Load. Da sowohl das Empfangen als auch das Senden eines Telegramms mehrere, mindestens aber je drei Aufrufe der Anweisung Modbus_Slave erfordert, sollten Sie Modbus_Slave mindestens zwölfmal während des Timeout-Zeitraums für die Antwort des Modbus-Masters ausführen, damit das Empfangen und Senden von Daten durch den Modbus-Slave doppelt so oft ausgeführt wird, wie durch den Timeout-Zeitraum vorgegeben.

HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 40001 oder 400001. Diese Adressen entsprechen der Anfangsadresse des Halteregisters im Zielsystemspeicher. Sie können jedoch die Variable HR_Start_Offset konfigurieren, um eine andere Anfangsadresse als 40001 oder 400001 für das Modbus-Halteregister zu konfigurieren.

Die Adresse 0 im Empfangstelegramm entspricht der Anfangsadresse des Halteregisters im Zielsystemspeicher. Über die Variable HR_Start_Offset können Sie eine andere Anfangsadresse als 0 für das Modbus-Halteregister konfigurieren.

Sie können z. B. ein Halteregister mit Beginn an MW100 und einer Länge von 100 Wörtern konfigurieren. Mit HR_Start_Offset = 20 entspricht die Adresse 20 im Empfangstelegramm der

Anfangsadresse des Halteregisters im Zielspeicher (MW100). Jede Adresse im Empfangstelegramm unter 20 und über 119 führt zu einem Adressierungsfehler.

Tabelle 4-169 Beispiel für die Adressierung des Modbus-Halteregisters, falls DATA_PTR ein Zeiger auf MW100 mit der Länge 100 Worte ist

HR_Start_Offset	Adresse	Minimum	Maximum
0	Modbus-Adresse (Wort)	0	99
	S7-1500 Adresse	MW100	MW298
20	Modbus-Adresse (Wort)	20	119
	S7-1500 Adresse	MW100	MW298

HR_Start_Offset ist ein Wortwert, der die Anfangsadresse des Modbus-Halteregisters angibt und im Instanz-Datenbaustein Modbus_Slave gespeichert ist. Sie können diese öffentliche statische Variable über die Parameter-Klappliste auswählen, nachdem Sie Modbus_Slave in Ihr Programm eingefügt haben.

Wenn Sie beispielsweise Modbus_Slave in ein KOP-Netzwerk eingefügt haben, können Sie in ein vorheriges Netzwerk gehen und den Wert HR_Start_Offset mit dem Move-Befehl zuweisen. Der Wert muss vor der Ausführung von Modbus_Slave zugewiesen werden.

Modbus-Slave-Variable über den Standard-DB-Namen eingeben:

1. Positionieren Sie den Cursor im Parameterfeld OUT1 und geben Sie das Zeichen m ein.
2. Wählen Sie in der Klappliste den gewünschten Instanz-DB der Anweisung Modbus_Slave aus.
3. Positionieren Sie den Cursor rechts vom DB-Namen (nach den Anführungszeichen) und geben Sie einen Punkt ein.
4. Wählen Sie in der Klappliste "Modbus_Slave_DB.HR_Start_Offset" aus.

Anweisungsversionen

Die Version 4.0 ist funktional identisch zur Version 3.0 und wurde aufgrund interner Maßnahmen erhöht.

Zugriff auf Datenbereiche in DBs anstelle des direkten Zugriffs auf MODBUS-Adressen ab Version V5.0

Zugriff auf Datenbereiche in DBs anstelle des direkten Zugriffs auf MODBUS-Adressen ab Version V4.0

Ab der Anweisungs-Version V4.0 von Modbus_Slave und ab den Firmware-Versionen V2.5 (S7-1500-CPUs) bzw. V4.2 (S7-1200-CPUs) können Sie anstelle des direkten Zugriffs auf Prozessabbilder und Holding-Register auf Datenbereiche in DBs zugreifen. Dabei muss für den DB das Attribut "Optimierter Bausteinzugriff" deaktiviert sein und er darf nicht ausschließlich im Ladespeicher liegen.

Wenn eine MODBUS-Anforderung eintrifft und Sie für den MODBUS-Datentyp des zugehörigen Funktionscodes keinen Datenbereich definiert haben, wird die Anforderung wie in den bisherigen Anweisungs-Versionen behandelt, d. h. es erfolgt ein direkter Zugriff auf Prozessabbilder und Holding-Register.

Wenn Sie hingegen für den MODBUS-Datentyp des Funktionscode einen Datenbereich definiert haben, liest die Anweisung Modbus_Slave aus diesem Datenbereich oder sie schreibt in ihn. Ob gelesen oder geschrieben wird, ist vom Auftragstyp abhängig.

Eine einzelne MODBUS-Anforderung kann immer nur von einem Datenbereich lesen oder in einen Datenbereich schreiben. Wenn Sie also z. B. Holding-Register lesen wollen, die sich über mehrere Datenbereiche erstrecken, benötigen Sie dafür mehrere MODBUS-Anforderungen.

Regeln zur Definition der Datenbereiche

Sie können bis zu acht Datenbereiche in unterschiedlichen DBs definieren; jeder DB darf nur einen Datenbereich enthalten. Eine einzelne MODBUS-Anforderung kann nur von genau einem Datenbereich lesen oder in genau einen Datenbereich schreiben. Dabei entspricht jeder Datenbereich einem MODBUS-Adressbereich. Die Definition erfolgt in der statischen Variable Data_Area_Array des Instanz-DB; Data_Area_Array ist ein Feld, das aus acht Elementen besteht.

Wenn Sie weniger als acht Datenbereiche verwenden wollen, müssen die gewünschten Datenbereiche lückenlos hintereinander liegen. Der erste Leereintrag in den Datenbereichen beendet die Datenbereich-Suche bei der Bearbeitung. Wenn Sie also z. B. die Feldelemente 1, 2, 4 und 5 definiert haben, werden nur die Feldelemente 1 und 2 erkannt, da das Feldelement 3 leer ist.

Das Feld Data_Area_Array besteht aus 8 Elementen: Data_Area_Array[1] … Data_Area_Array[8]

Jedes Feldelement Data_Area_Array[x], 1 <= x <= 8, ist ein UDT vom Typ MB_DataArea und hat folgenden Aufbau:

Parameter	Datentyp	Bedeutung
data_type	UInt	Kennung für den MODBUS-Datentyp, der auf diesen Datenbereich abgebildet wird: • 0: Kennung für ein leeres Feldelement bzw. für einen nicht verwendeten Datenbereich. In diesem Fall sind die Werte von db, start und length irrelevant. • 1: Prozessabbild der Ausgänge (verwendet bei den Funktionscodes 1, 5 und 15) • 2: Prozessabbild der Eingänge (verwendet beim Funktionscode 2) • 3: Holding-Register (verwendet bei den Funktionscodes 3, 6 und 16) • 4: Input-Register (verwendet beim Funktionscode 4) Hinweis: Wenn Sie für einen MODBUS-Datentyp einen Datenbereich definiert haben, kann die Anweisung MB_SERVER auf diesen MODBUS-Datentyp nicht mehr direkt zugreifen. Wenn die Adresse einer MODBUS-Anforderung für einen solchen Datentyp keinem definierten Datenbereich entspricht, wird in STATUS der Wert W#16#8383 zurückgeliefert.
db	UInt	Nummer des Datenbausteins, in den die nachfolgend definierten MODBUS-Register oder Bits abgebildet werden. Die DB-Nummer muss in den Datenbereichen eindeutig sein. Dieselbe DB-Nummer darf nicht in mehreren Datenbereichen definiert werden. Der DB muss ein DB mit Standardzugriff sein und er darf nicht ausschließlich im Ladespeicher liegen. Datenbereiche beginnen stets bei der Byte-Adresse 0 des DB. Zulässige Werte: 1 bis 60999

Parameter	Datentyp	Bedeutung
start	UInt	Erste MODBUS-Adresse, die in den Datenbaustein ab Adresse 0.0 abgebildet wird. Zulässige Werte: 0 bis 65535
length	UInt	Anzahl der Bits (für die Werte 1 und 2 von data_type) bzw. Anzahl der Register (für die Werte 3 und 4 von data_type). Die MODBUS-Adressbereiche ein und desselben MODBUS-Datentyps dürfen sich nicht überlappen. Zulässige Werte: 1 bis 65535

Beispiele zur Definition von Datenbereichen

- Erstes Beispiel: data_type = 3, db = 1, start = 10, length = 6
  Die Holding Register (data_type = 3) werden im Datenbaustein 1 (db = 1) abgebildet. Die Modbus-Adresse 10 (start = 10) befindet sich an Datenwort 0. Die letzte gültige Modbus-Adresse 15 (length = 6) befindet sich an Datenwort 5.
- Zweites Beispiel: data_type = 2, db = 15, start = 1700, length = 112
  Die Eingänge (data_type = 2) werden im Datenbaustein 15 (db = 15) abgebildet. Die Modbus-Adresse 1700 (start = 1700) befindet sich an Datenwort 0. Die letzte gültige Modbus-Adresse 1811 (length = 112) befindet sich an Datenwort 111.

Beschränkung des Lesezugriffs auf Prozessabbilder ab Version V5.0

Beschränkung des Lesezugriffs auf Prozessabbilder

Ab der Anweisungsversion V4.0 von Modbus_Slave können Sie im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge je einen Bereich definieren, auf den remote MODBUS-Geräte lesend zugreifen dürfen. Ein Lesezugriff remoter MODBUS-Geräte auf Adressen außerhalb dieser Prozessabbild-Bereiche ist dann nicht mehr möglich.

Hinweis

Beschränkung des Schreibzugriffs auf Prozessabbilder

Die Möglichkeit, den Schreibzugriff auf das Prozessabbild der Ausgänge auf einen bestimmten Bereich zu beschränken, ist ab der Anweisungs-Version V3.0 möglich.

Definition der Lesebereiche in den Prozessabbildern

Die Definition der Lesebereiche in den Prozessabbildern erfolgt in den folgenden statischen Variablen des Instanz-DB:

- QB_Read_Start: Adresse des ersten Bytes im Prozessabbild der Ausgänge, das von einem remoten MODBUS-Gerät gelesen werden darf (gilt für den Funktionscode 1)
- QB_Read_Count: Anzahl der Bytes im Prozessabbild der Ausgänge, die von einem remoten MODBUS-Gerät gelesen werden dürfen (gilt für den Funktionscode 1)

- IB_Read_Start: Adresse des ersten Bytes im Prozessabbild der Eingänge, das von einem remoten MODBUS-Gerät gelesen werden darf (gilt für die Funktionscodes 2 und 4)
- IB_Read_Count: Anzahl der Bytes im Prozessabbild der Eingänge, die von einem remoten MODBUS-Gerät gelesen werden dürfen (gilt für die Funktionscodes 2 und 4)

Statische Variablen im Instanz-DB zur Definition der Schreib- und der Lesebereiche in den Prozessabbildern

Die folgende Tabelle beschreibt die oben aufgeführten statischen Variablen im Instanz-DB der Anweisung Modbus_Slave, mit denen Sie die Lesebereiche in den Prozessabbildern definieren.

Der Vollständigkeit halber sind auch diejenigen statischen Variablen angegeben, mit denen Sie ab Version V3.0 die Schreibbereiche in den Prozessabbildern definieren (QB_Start und QB_Count).

Variable	Datentyp	Startwert
QB_Start	UInt	0
QB_Count	UInt	65535
QB_Read_Start	UInt	0
QB_Read_Count	UInt	65535
IB_Read_Start	UInt	0
IB_Read_Count	UInt	65535

Telegrammaufbau

Extended_Addressing

Auf die Extended_Addressing-Variable wird in ähnlicher Weise zugegriffen wie auf die HR_Start_Offset-Referenz, außer dass es sich bei der Extended_Addressing-Variable um einen Booleschen Wert handelt.

Für die Modbus-Slave-Adressierung kann, mit Extended_Adressing = FALSE, ein einzelnes Byte (dies ist der Modbus-Standard) oder zwei Bytes konfiguriert werden (Extended_Adressing = TRUE). Die erweiterte Adressierung dient zur Adressierung von mehr als 247 Geräten in einem einzigen Netzwerk. Mit Extended_Adressing = TRUE können Sie maximal 65535 Adressen ansprechen. Im Folgenden wird ein Modbus-Telegramm als Beispiel gezeigt.

Tabelle 4-170 Slave-Adresse mit einem Byte (Byte 0)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Anforderung	Slave-Adresse	Funktionscode	Anfangsadresse		Daten		
Gültige Antwort	Slave-Adresse	Funktionscode	Länge	Daten...			
Fehlermeldung	Slave-Adresse	0xxx	Exceptioncode				

Tabelle 4-171 Slave-Adresse mit zwei Bytes (Byte 0 und Byte 1)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anforderung	Slave-Adresse		Funktions-code	Anfangsadresse		Daten	
Gültige Antwort	Slave-Adresse		Funktions-code	Länge	Daten...		
Fehlermeldung	Slave-Adresse		0xxx	Exceptionc-ode			

Beschreibung der Telegramme

Der Datenverkehr zwischen Master und Slave bzw. zwischen Slave und Master beginnt mit der Slave-Adresse, gefolgt vom Funktionscode. Dann werden die Daten übertragen. Der Aufbau des Datenfeldes ist abhängig vom verwendeten Funktionscode. Am Ende des Telegramms wird die Prüfsumme (CRC) übertragen.

Funktionscode 1 - Diese Funktion ermöglicht das Lesen einzelner Ausgangsbits

Tabelle 4-172 FC 1 - Ausgangsbits lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 1	Anfangsadresse		Anzahl Ausgänge	
Gültige Antwort	Slave-Adresse	Funktions-code 1	Länge [1]	Ausgangsdaten [3)]		
Fehlermeldung	Slave-Adresse	0x81	Exceptioncode [2]	---		

[1] Länge: Falls die Anzahl der Ausgänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

[3] Die Ausgangsdaten können mehrere Bytes umfassen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 1	Anfangsadresse		Anzahl Ausgänge	
Gültige Ant-wort	Slave-Adresse		Funktions-code 1	Länge [1]	Ausgangsdaten		
Fehlermel-dung	Slave-Adresse		0x81	Exceptionc-ode [2]	---		

[1] Länge: Falls die Anzahl der Ausgänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

[3] Die Ausgangsdaten können mehrere Bytes umfassen

Funktionscode 2 - Diese Funktion ermöglicht das Lesen einzelner Eingangsbits

Tabelle 4-173 FC 2 - Eingangsbits lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 2	Anfangsadresse		Anzahl Eingänge	
Gültige Antwort	Slave-Adresse	Funktions-code 2	Länge [1]	Eingangsdaten		
Fehlermeldung	Slave-Adresse	0x82	Exceptioncode [2]	---		

[1] Länge: Falls die Anzahl der Eingänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 2	Anfangsadresse		Anzahl Eingänge	
Gültige Ant-wort	Slave-Adresse		Funktions-code 2	Länge [1]	Eingangsdaten		
Fehlermel-dung	Slave-Adresse		0x82	Exceptionc-ode [2]	---		

[1] Länge: Falls die Anzahl der Eingänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 3 - Diese Funktion ermöglicht das Lesen einzelner Register

Tabelle 4-174 FC 3 - Halteregister lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 3	Anfangsadresse		Anzahl Register	
Gültige Antwort	Slave-Adresse	Funktions-code 3	Länge [1]	Registerdaten		
Fehlermeldung	Slave-Adresse	0x83	Exceptioncode [2]	---		

[1] Länge: Anzahl Bytes

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 3	Anfangsadresse		AnzahlRegister	
Gültige Ant-wort	Slave-Adresse		Funktions-code 3	Länge [1]	Registerdaten		
Fehlermel-dung	Slave-Adresse		0x83	Exceptionc-ode [2]	---		

[1] Länge: Anzahl Bytes

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 4 - Diese Funktion ermöglicht das Lesen einzelner Register

Tabelle 4-175 FC 4 - Eingangswörter lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 4	Anfangsadresse		Anzahl Eingangswörter	
Gültige Antwort	Slave-Adresse	Funktions-code 4	Länge [1]	Eingangsdaten		
Fehlermeldung	Slave-Adresse	0x84	Exceptioncode [2]	---		

[1] Länge: 2 * Anzahl Eingangswörter

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 4	Anfangsadresse		Anzahl Eingangswörter	
Gültige Ant-wort	Slave-Adresse		Funktions-code 4	Länge [1]	Eingangsdaten		
Fehlermel-dung	Slave-Adresse		0x84	Exceptionc-ode [2]	---		

[1] Länge: 2 * Anzahl Eingangswörter

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 5 - Mit dieser Funktion kann ein einzelnes Bit gesetzt oder gelöscht werden

Tabelle 4-176 FC 5 - Ein Ausgangsbit schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 5	Anfangsadresse		Wert	
Gültige Antwort	Slave-Adresse	Funktions-code 5	Länge	Wert		
Fehlermeldung	Slave-Adresse	0x85	Exceptioncode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 5	Anfangsadresse		Wert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 5	Länge	Wert		
Fehlermel-dung	Slave-Adresse		0x85	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 6 - Diese Funktion ermöglicht das Schreiben einzelner Register

Tabelle 4-177 FC 6 - Halteregister schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 6	Adresse		Register	
Gültige Antwort	Slave-Adresse	Funktions-code 6	Adresse		Register	
Fehlermeldung	Slave-Adresse	0x86	Exceptioncode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 6	Adresse		Register	
Gültige Ant-wort	Slave-Adresse		Funktions-code 6	Adresse		Register	
Fehlermel-dung	Slave-Adresse		0x86	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 8 - Diese Funktion dient der Überprüfung der Kommunikationsverbindung

Tabelle 4-178 FC 8 - Slave-Zustand

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 8	Diagnosecode		Testwert	
Gültige Antwort	Slave-Adresse	Funktions-code 8	Diagnosecode		Testwert	
Fehlermeldung	Slave-Adresse	0x88	Exceptioncode [1]	---		

[1] E-Code: 01 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 8	Diagnosecode		Testwert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 8	Diagnosecode		Testwert	
Fehlermel-dung	Slave-Adresse		0x88	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 03 oder 04

Funktionscode 11 - Diese Funktion ermöglicht das Lesen von 2 Byte "Status-Word" und ein 2 Byte "Event-Counter"

Tabelle 4-179 FC 11 - Ereigniszähler der Slave-Kommunikation

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 11	---			
Gültige Antwort	Slave-Adresse	Funktions-code 11	Status		Ereigniszähler	
Fehlermeldung	Slave-Adresse	0x8B	Exceptioncode [1]	---		

[1] E-Code: 01 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 11	---			
Gültige Ant-wort	Slave-Adresse		Funktions-code 11	Status		Ereigniszähler	
Fehlermel-dung	Slave-Adresse		0x8B	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 04

Funktionscode 15 - Diese Funktion ermöglicht das Schreiben mehrerer Bits

Tabelle 4-180 FC 15 - Ein/mehrere Ausgangsbits schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Anfrage	Slave-Adresse	Funktionscode 15	Anfangsadresse		Anzahl Ausgangswörter		Bytezähler [1]	Wert	
Gültige Antwort	Slave-Adresse	Funktionscode 15	Anfangsadresse		Anzahl Ausgangswörter		---		
Fehlermeldung	Slave-Adresse	0x8F	Exceptioncode [2]	---					

[1] Bytezähler: Falls die Anzahl der Bytes dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Anfrage	Slave-Adresse		Funktionscode 15	Anfangsadresse		Anzahl Ausgangswörter		Bytezähler [1]	Wert	
Gültige Antwort	Slave-Adresse		Funktionscode 15	Anfangsadresse		Anzahl Ausgangswörter		---		
Fehlermeldung	Slave-Adresse		0x8F	Exceptioncode [2]		---				

[1] Bytezähler: Falls die Anzahl der Bytes dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 16 - Diese Funktion ermöglicht das Schreiben einzelner oder mehrerer Register

Tabelle 4-181 FC 16 - Ein/mehrere Halteregister schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Anfrage	Slave-Adresse	Funktionscode 16	Anfangsadresse		Anzahl Register		Bytezähler [1]	Wert	
Gültige Antwort	Slave-Adresse	Funktionscode 16	Anfangsadresse		Anzahl Register		---		
Fehlermeldung	Slave-Adresse	0x90	Exceptioncode [2]	---					

[1] Bytezähler: Anzahl der Register * 2

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Anfrage	Slave-Adresse		Funktions-code 16	Anfangsadresse		Anzahl Register		Bytezäh-ler [1]	Wert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 16	Anfangsadresse		Anzahl Register		---		
Fehlermel-dung	Slave-Adresse		0x90	Exceptioncode [2]		---				

[1] Bytezähler: Anzahl der Register * 2

[2] E-Code: 01 oder 02 oder 03 oder 04

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/ maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüber-schreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichen-verzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endese-quenz"	-
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertra-gung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8088	Längenangabe größer als der im Empfangspuffer ein-gestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehler-code auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wäh-len Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)

Fehlercode	Beschreibung	Abhilfe
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		

Fehlercode	Beschreibung	Abhilfe
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)

Fehlercode	Beschreibung	Abhilfe
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.

Fehlercode	Beschreibung	Abhilfe
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.

Fehlercode	Beschreibung	Abhilfe
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions

Fehlercode	Beschreibung	Abhilfe
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.

Fehlercode	Beschreibung	Abhilfe
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.

Fehlercode	Beschreibung	Abhilfe
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.

Fehlercode	Beschreibung	Abhilfe
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 4180) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.

Fehlercode	Beschreibung	Abhilfe
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.

Fehlercode	Beschreibung	Abhilfe
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.
16#8828 [1)]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1)]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1)]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1)]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1)]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		

Fehlercode	Beschreibung	Abhilfe
16#8428 [1]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb

Fehlercode	Beschreibung	Abhilfe
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

ET 200S Serielle Schnittstelle

Hinweise zum Einsatz an einer S7-1500

S_SEND / S_RCV / S_USSI / S_MODB an S7-1500

Beachten Sie folgendes, falls Sie die Anweisungen S_SEND, S_RCV, S_USSI oder S_MODB der dezentralen Peripherie ET 200S 1SI an einer S7-1500 einsetzen:

- S_SEND
  Datentyp des Parameters DB_NO: UInt.
  Es können nur Daten aus nicht optimierten Speicherbereichen gesendet werden. Beachten Sie dies bei der Parametrierung der Parameter DB_NO, DBB_NO und LEN.
- S_RCV
  Datentyp des Parameters DB_NO: UInt.
  Es können nur Daten aus nicht optimierten Speicherbereichen empfangen werden. Beachten Sie dies bei der Parametrierung der Parameter DB_NO, DBB_NO und LEN.
- S_USSI
  - Datentyp der Parameter DBND, DBPA und DBCP: UInt
    Zulässige DB-Nummern für die Parameter DBND, DBPA und DBCP: 60000 bis 60999
  - Verhalten der CPU im Fehlerfall
    Im Fehlerfall wird die CPU nicht in STOP geschaltet.
    Falls die CPU im Fehlerfall in STOP gehen soll, muss der Parameter ANZ im Anwenderprogramm ausgewertet werden
    (falls ANZ ungleich Null ist, muss die Anweisung STP aufgerufen werden):
- S_MODB
  Der Datentyp des Parameters DB_NO: DB_ANY.
  Es können nur Daten aus nicht optimierten Speicherbereichen ausgetauscht werden. Beachten Sie dies bei der Parametrierung des Parameters DB_NO und des Datenbausteins mit der Modbus-Datenumsetzungstabelle.

Anweisungen der Bibliothek "Kommunikationsprozessor > ET200S Serielle Schnittstelle"

Beachten Sie folgendes, falls Sie die Anweisungen der dezentralen Peripherie ET 200S 1SI an einer S7-1500 einsetzen (Bibliothek: "Kommunikationsprozessor > ET200S Serielle Schnittstelle"):

- Am Parameter LADDR muss die Hardware-Kennung des Kommunikationsmoduls parametriert werden.
- Sowohl der Sende- als auch der Empfangspuffer müssen im nicht optimierten Speicherbereich liegen.

Anweisungen der ET 200S 1SI

Die Beschreibungen der Anweisungen der ET 200S 1SI finden Sie in den folgenden Kapiteln:

- ET 200S 1SI für 3964(R) und ASCII (Seite 6720)
- ET 200S 1SI für Modbus (Seite 6739)
- ET 200S 1SI für USS (Seite 6757)

Punkt-zu-Punkt

PORT_CFG: Kommunikationsparameter dynamisch projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "PORT_CFG" ermöglicht die dynamische Projektierung von Kommunikationsparametern für einen Punkt-zu-Punkt-Kommunikationsport.

Die ursprüngliche statische Projektierung des Ports richten Sie in der Hardware-Konfiguration ein. Diese Projektierung können Sie durch die Ausführung der Anweisung "PORT_CFG" ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen.

Mit "PORT_CFG" können Sie die folgenden Einstellungen der Kommunikationsparameter beeinflussen:

- Parität
- Baudrate
- Anzahl der Bits pro Zeichen
- Anzahl der Bitstopps
- Art und Eigenschaften der Flusskontrolle

Die Änderungen durch die Anweisung "PORT_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Serielle Daten können Sie über die elektrischen Verbindungen RS-232 (Halb- und Vollduplex) und RS-485 (Halbduplex) übertragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PORT_CFG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Aktiviert die Projektierungsänderung bei einer steigenden Flanke
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
PROTOCOL	Input	UINT	E, A, M, D, L oder Konstante	Übertragungsprotokoll: • 0: Punkt-zu-Punkt-Kommunikationsprotokoll • 1..n: Zukünftige Definition für spezifische Übertragungsprotokolle
BAUD	Input	UINT	E, A, M, D, L oder Konstante	Baudrate des Ports: • 1: 300 Baud • 2: 600 Baud • 3: 1200 Baud • 4: 2400 Baud • 5: 4800 Baud • 6: 9600 Baud (Standardeinstellung) • 7: 19200 Baud • 8: 38400 Baud • 9: 57600 Baud • 10: 76800 Baud • 11: 115200 Baud
PARITY	Input	UINT	E, A, M, D, L oder Konstante	Parität des Ports: • 1: Keine Parität (Standardeinstellung) • 2: Gerade Parität • 3: Ungerade Parität • 4: Markierung Parität • 5: Leerzeichen Parität
DATABITS	Input	UINT	E, A, M, D, L oder Konstante	Bits pro Zeichen: • 1: 8 Bits pro Zeichen (Standardeinstellung) • 2: 7 Bits pro Zeichen
STOPBITS	Input	UINT	E, A, M, D, L oder Konstante	Anzahl der Stoppbits: • 1: 1 Stoppbit (Standardeinstellung) • 2: 2 Stoppbits
FLOWCTRL	Input	UINT	E, A, M, D, L oder Konstante	Datenflusskontrolle: • 1: Keine (Standard) • 2: XON/XOFF • 3: Hardware-Flusskontrolle (RTS immer eingeschaltet) • 4: Hardware-Flusskontrolle (RTS abschaltbar während der Übertragung)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
XONCHAR	Input	CHAR	E, A, M, D, L oder Konstante	Gibt das Zeichen an, das als XON-Zeichen verwendet wird. Als Standard ist das Zeichen DC1 (11H) eingestellt.
XOFFCHAR	Input	CHAR	E, A, M, D, L oder Konstante	Gibt das Zeichen an, das als XOFF-Zeichen verwendet wird. Als Standard ist das Zeichen DC3 (13H) eingestellt.
WAITTIME	Input	UINT	E, A, M, D, L oder Konstante	Gibt die Wartezeit für XON oder CTS nach dem Beginn der Übertragung an. Der angegebene Wert muss größer als 0 sein. Als Standard sind 2000 Millisekunden eingestellt.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
80A0	Das angegebene Protokoll ist ungültig.
80A1	Die angegebene Baudrate ist ungültig.
80A2	Die angegebene Parität ist ungültig.
80A3	Die angegebene Anzahl der Bits pro Zeichen ist ungültig.
80A4	Die angegebene Anzahl der Stoppbits ist ungültig.
80A5	Die angegebene Art der Flusskontrolle ist ungültig.
80A6	Fehlerhafter Wert am Parameter WAITTIME Wenn die Datenflusskontrolle aktiviert ist, muss der Wert am Parameter WAITTIME größer als Null sein.
80A7	Ungültige Werte an den Parametern XONCHAR und XOFFCHAR.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SEND_CFG: Serielle Übertragungsparameter dynamisch projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "SEND_CFG" ermöglicht die dynamische Projektierung von seriellen Übertragungsparametern für einen Punkt-zu-Punkt-Kommunikationsport. Alle für die Übertragung anstehenden Nachrichten werden nach Ausführung von "SEND_CFG" verworfen.

Die ursprüngliche statische Projektierung des Ports richten Sie in der Hardware-Konfiguration ein. Diese Projektierung können Sie durch die Ausführung der Anweisung "SEND_CFG" ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen. Mit "SEND_CFG" können Sie die folgenden Einstellungen der Übertragungsparameter beeinflussen:

- Zeitabstand zwischen der Aktivierung von RTS (Request to Send) und dem Start der Übertragung
- Zeitabstand zwischen Übertragungsende und der Deaktivierung von RTS
- Bitzeiten für Pausen festlegen

Die Änderungen durch die Anweisung "SEND_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Serielle Daten können Sie über die elektrischen Verbindungen RS-232 (Halb- und Vollduplex) und RS-485 (Halbduplex) übertragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND_CFG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Aktiviert die Projektierungsänderung bei einer steigenden Flanke
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
RTSONDLY	Input	UINT	E, A, M, D, L oder Konstante	Zeit, die nach der Aktivierung von RTS bis zum Start der Übertragung vergehen soll. Gültige Werte für diesen Parameter sind: • 0 (Standard) • 0 bis 65535 ms in Schritten von 1 ms Dieser Parameter gilt nicht bei RS-485-Baugruppen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RTSOFFDLY	Input	UINT	E, A, M, D, L oder Konstante	Zeit, die nach dem Übertragungsende bis zur Deaktivierung von RTS vergehen soll. Gültige Werte für diesen Parameter sind: • 0 (Standard) • 0 bis 65535 ms in Schritten von 1 ms Dieser Parameter gilt nicht bei RS-485-Baugruppen.
BREAK	Input	UINT	E, A, M, D, L oder Konstante	Gibt die Bitzeiten für eine Pause an, die zum Nachrichtenbeginn gesendet wird. Als Standard werden 12 Bitzeiten eingestellt. Maximal können 25000 Bitzeiten angegeben werden.
IDLELINE	Input	UINT	E, A, M, D, L oder Konstante	Gibt die Bitzeiten für Idle Line an nach der Pause zum Nachrichtenbeginn. Als Standard werden 12 Bitzeiten eingestellt. Maximal können 25000 Bitzeiten angegeben werden.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
80B0	Die Konfiguration einer Übertragungsunterbrechung ist nicht erlaubt
80B1	Die angegebene Pausenzeit überschreitet das erlaubte Maximum von 25000 Bitzeiten
80B2	Die für Idle Line angegebene Zeit überschreitet das erlaubte Maximum von 25000 Bitzeiten
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RCV_CFG: Serielle Empfangsparameter dynamisch projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "RCV_CFG" ermöglicht die dynamische Projektierung von seriellen Empfangsparametern für einen Punkt-zu-Punkt-Kommunikationsport. Mit dieser Anweisung können Sie die Bedingungen projektieren, die den Anfang und das Ende einer zu übertragenden Nachricht bestimmen. Der Empfang von Nachrichten, die diesen Bedingungen entsprechen, kann dann durch die Anweisung "RCV_PTP (Seite 4234)" freigegeben werden.

Die ursprüngliche statische Projektierung des Ports richten Sie in den Eigenschaften der Hardware-Konfiguration ein. Führen Sie die Anweisung "RCV_CFG" in Ihrem Programm aus, um die Projektierung zu ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen. Die Änderungen durch die Anweisung "RCV_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Alle für die Übertragung anstehenden Nachrichten werden nach Ausführung der Anweisung "RCV_CFG" verworfen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV_CFG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Aktiviert die Projektierungsänderung bei einer steigenden Flanke.
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
CONDITIONS	Input	CONDITIONS	D, L	Datenstruktur, die Bedingungen für Anfang und Ende der Datenübertragung definiert.
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datentyp CONDITIONS

Mithilfe der Struktur CONDITIONS legen Sie die Start- und Endbedingungen für die Nachrichtenübertragung fest. Die Struktur CONDITIONS ist in dem Instanz-DB der Anweisung "RCV_CFG" enthalten. Über die Struktur CONDITIONS definieren Sie die die Start- und Endbedingungen, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt:

- Die Startbedingung für die Datenübertragung definieren Sie in der Struktur START
- Die Endbedingung für die Datenübertragung definieren Sie in der Struktur END

Dabei können eine oder mehrere Start- und Endbedingungen definiert werden. Bei der Angabe mehrerer Start- oder Endbedingungen werden diese durch ein ODER-Gatter verknüpft.

Die folgende Tabelle zeigt die Struktur CONDITIONS:

Parameter	Datentyp	Beschreibung
START	STRUCT	Startbedingungen

Parameter		Datentyp	Beschreibung
	STARTCOND	UINT	Gibt die Startbedingung an (Details, siehe unten). Die Startbedingung kann als ein 16-Bit Hexadezimalwert angegeben werden. Mögliche Werte für die Startbedingung sind: • 1: Startzeichen • 2: Beliebiges Zeichen (Standardeinstellung) • 4: Line break • 8: Idle line • 16: Zeichenkette 1 • 32: Zeichenkette 2 • 64: Zeichenkette 3 • 128: Zeichenkette 4 Auch mehrere Startbedingungen können am Parameter STARTCOND festlegt werden. Dafür wird die Summe aus den Werten der einzelnen Bedingungen angegeben. Wenn beispielsweise als Startbedingung "Idle line" ODER "Zeichenkette 1" ODER "Zeichenkette 4" definiert werden soll, muss der Wert "152" angegeben werden.
	IDLETIME	UINT	Gibt die maximale Ruhezeit der Leitung an, bevor der Empfang gestartet wird. Gültige Werte für diesen Parameter sind: • 40 Bitzeiten (Standardeinstellung) • 0 bis 2500 Bitzeiten
	STARTCHAR	BYTE	Gibt das Startzeichen an. Diese Einstellung ist nur dann wirksam, wenn die projektierte Startbedingung "Startzeichen" ist. Gültige Werte für diesen Parameter sind: • 02 (STX): Standardeinstellung • B#16#00 bis B#16#FF
	SEQ[1].CTL	BYTE	Zeichenkette 1: Steuerung für jedes Zeichen Durch die Bitposition der Zeichen kann festgelegt werden, welche Zeichen der Zeichenkette berücksichtigt oder ignoriert werden. Um die Zeichen auszuwerten, müssen die entsprechenden Bits gesetzt werden. • Bit 0: 1 Zeichen • Bit 1: 2 Zeichen • Bit 2: 3 Zeichen • Bit 3: 4 Zeichen • Bit 4: 5 Zeichen Ein Zeichen wird ignoriert, wenn das entsprechende Bit zurückgesetzt ist.
	SEQ[1].STR	CHAR[5]	Zeichenkette 1: Startzeichen (5 Zeichen)
	SEQ[2].CTL	BYTE	Zeichenkette 2: Steuerung für jedes Zeichen ignorieren/vergleichen
	SEQ[2].STR	CHAR[5]	Zeichenkette 2: Startzeichen (5 Zeichen)
	SEQ[3].CTL	BYTE	Zeichenkette 3: Steuerung für jedes Zeichen ignorieren/vergleichen
	SEQ[3].STR	CHAR[5]	Zeichenkette 3: Startzeichen (5 Zeichen)
	SEQ[4].CTL	BYTE	Zeichenkette 4: Steuerung für jedes Zeichen ignorieren/vergleichen
	SEQ[4].STR	CHAR[5]	Zeichenkette 4: Startzeichen (5 Zeichen)
END		STRUCT	Endbedingungen

Parameter		Datentyp	Beschreibung
	ENDCOND	UINT	Gibt die Endbedingung an (Details, siehe unten). Die Endbedingung kann als ein 16-Bit Hexadezimalwert angegeben werden. Mögliche Werte für die Endbedingung sind: • 1: Antwort-Zeitüberschreitung • 2: Nachrichten-Zeitüberschreitung • 4: Zeitüberschreitung innerhalb der Zeichenkette • 8: Maximale Länge • 16: N+LEN+M; Die Angebe über die Meldungslänge ist in der Meldung integriert und wird ausgewertet. • 32: Zeichenkette 1 Auch mehrere Endbedingungen können am Parameter ENDCOND festlegt werden. Dafür wird die Summe aus den Werten der einzelnen Endbedingungen angegeben. Wenn beispielsweise die Endbedingung "Maximale Länge" ODER "Sequenz 1" definiert werden soll, muss der Wert "40" angegeben werden.
	MAXLEN	UINT	Gibt die maximale Anzahl der Zeichen in einer Nachricht an. Gültige Werte* für diesen Parameter sind: • 1 Zeichen (Standardeinstellung) • 0 bis 1024 Zeichen Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Maximale Länge" eingestellt ist.
	N	UINT	Offset des Längenfeldes in der Nachricht Gültige Werte für diesen Parameter sind: • 0 Zeichen (Standardeinstellung) • 0 bis 1024 Zeichen Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist.
	LENGTHSIZE	UINT	Größe des Längenfeldes in Bytes Gültige Werte* für diesen Parameter sind: • 0 Bytes (Standardeinstellung) • 1 Byte • 2 Bytes • 4 Bytes Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist.
	LENGTHM	UINT	Gibt die Anzahl von Endzeichen an, die dem Längenfeld folgen aber nicht in der Länge der Nachricht enthalten sind. Gültige Werte für diesen Parameter sind: • 0 Zeichen (Standardeinstellung) • 0 bis 255 Zeichen Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist.
	RCVTIME	UINT	Gibt die Maximaldauer für den Empfang des ersten Zeichens einer Nachricht an. Gültige Werte für diesen Parameter sind: • 200 ms (Standardeinstellung)

Parameter		Datentyp	Beschreibung
			• 0 bis 65535 ms in Schritten von 1 ms Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Antwort-Zeitüberschreitung" eingestellt ist.
	MSGTIME	UINT	Gibt die Maximaldauer für den Empfang einer Nachricht an. Gültige Werte für diesen Parameter sind: • 200 ms (Standardeinstellung) • 0 bis 65535 ms in Schritten von 1 ms Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Nachrichten-Zeitüberschreitung" eingestellt ist.
	CHARGAP	UINT	Gibt den Zeitabstand beim Empfang aufeinander folgender Zeichen an. Gültige Werte für diesen Parameter sind: • 12 Bitzeiten (Standardeinstellung) • 0 bis 2500 Bitzeiten Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Zeitüberschreitung innerhalb der Zeichenkette" eingestellt ist.
	SEQ.CTL	BYTE	Zeichenkette: Steuerung für jedes Zeichen Durch die Bitposition der Zeichen kann festgelegt werden, welche Zeichen der Zeichenkette berücksichtigt oder ignoriert werden. Um die Zeichen auszuwerten, müssen die entsprechenden Bits gesetzt werden. • Bit 0: 1 Zeichen • Bit 1: 2 Zeichen • Bit 2: 3 Zeichen • Bit 3: 4 Zeichen • Bit 4: 5 Zeichen Ein Zeichen wird ignoriert, wenn das entsprechende Bit zurückgesetzt ist.
	SEQ.STR	CHAR[5]	Zeichenkette: Startzeichen (5 Zeichen)
	* Diese Wertebereiche gelten auch für die entsprechenden Hardware-Einstellungen zum Festlegen des Nachrichtenendes.		

Startbedingungen für den Nachrichtenempfang (Parameter STARTCOND)

Der Beginn einer Nachricht wird vom Empfänger erkannt, wenn eine projektierte Startbedingung zutrifft. Die folgenden Bedingungen können als Startbedingungen für den Nachrichtenempfang festgelegt werden:

- Startzeichen: Der Beginn einer Nachricht wird beim Auftreten eines bestimmten Zeichens erkannt. Dieses Zeichen wird als erstes Zeichen der Nachricht gespeichert. Alle Zeichen, die vor dem Startzeichen empfangen worden sind, werden verworfen.
- Beliebiges Zeichen: Jedes beliebige Zeichen kann den Beginn einer Nachricht definieren. Dieses Zeichen wird als erstes Zeichen der Nachricht gespeichert.
- Line Break: Der Beginn einer Nachricht wird erkannt, wenn der Empfangsdatenstrom für mehr als ein Zeichen Länge unterbrochen wurde.

- Idle Line: Der Beginn einer Nachricht wird erkannt, wenn die Übertragungsleitung für eine bestimmte Zeit (angegeben in Bitzeiten) im Ruhezustand ist und anschließend wieder Zeichen übertragen werden.
- Zeichenkette (Sequenz): Der Beginn einer Nachricht wird erkannt, wenn eine festgelegte Zeichenfolge im Datenstrom auftritt. Sie können bis zu vier Zeichensequenzen mit jeweils bis zu fünf Zeichen festlegen.
  Beispiel: Eine empfangene hexadezimal-codierte Meldung enthält die folgenden Zeichen: "68 10 aa 68 bb 10 aa 16". Die konfigurierten Startzeichenfolgen finden Sie in der folgenden Tabelle. Startzeichenfolgen werden ausgewertet, nachdem das erste Zeichen 68H erfolgreich empfangen wurde. Nach erfolgreichem Empfang des vierten Zeichens (das zweite 68H) ist die Startbedingung "1" erfüllt. Wenn die Startbedingungen erfüllt sind, beginnt die Auswertung der Endebedingungen.
  Die Verarbeitung der Startzeichenfolge kann aufgrund verschiedener Fehler bei Parität, Framing oder Zeitabständen zwischen den Zeichen beendet werden. Diese Fehler führen dazu, dass die Meldung nicht empfangen wird, weil die Startbedingung nicht erfüllt wurde.

Startbedingung	Erstes Zeichen	Erstes Zeichen +1	Erstes Zeichen +2	Erstes Zeichen +3	Erstes Zeichen +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Endbedingungen für den Nachrichtenempfang (Parameter ENDCOND)

Das Ende einer Nachricht wird vom Empfänger erkannt, wenn eine projektierte Endbedingung zutrifft. Die folgenden Bedingungen können als Endbedingungen für den Nachrichtenempfang festgelegt werden:

- Antwort-Zeitüberschreitung: Der Nachrichtenempfang wird beendet, wenn die vorgegebene Maximaldauer für den Empfang eines Zeichens überschritten wird. Die Maximaldauer wird am Parameter RCVTIME definiert. Der Ablauf der definierten Zeit startet, sobald die letzte Übertragung beendet ist und die Anweisung RCV_PTP den Empfang der Nachricht freigibt. Wenn innerhalb der definierten Zeit (RCVTIME) kein Zeichen empfangen wurde, meldet die Anweisung RCV_PTP einen Fehler.
- Nachrichten-Zeitüberschreitung: Der Nachrichtenempfang wird beendet, wenn die vorgegebene Maximaldauer für den Empfang einer Nachricht überschritten wird. Die Maximaldauer wird am Parameter MSGTIME definiert. Der Ablauf der definierten Zeit wird gestartet, sobald das erste Zeichen der Nachricht empfangen wurde.
- Zeitüberschreitung innerhalb der Zeichenkette: Der Nachrichtenempfang wird beendet, wenn der Zeitabstand beim Empfang von zwei aufeinander folgenden Zeichen größer als der Wert am Parameter CHARGAP ist.
- Maximale Länge: Der Nachrichtenempfang wird beendet, wenn die am Parameter MAXLEN definierte Länge der Nachricht überschritten wird.

- Nachrichtenlänge auslesen (N+LEN+M): Der Nachrichtenempfang wird beendet, wenn eine bestimmte Nachrichtenlänge erreicht wird. Diese Länge wird durch die Werte der folgenden Parameter errechnet:
  - N: Position des Zeichens in der Nachricht, ab dem das Längenfeld beginnt.
  - LENGTHSIZE: Größe des Längenfeldes in Bytes
  - LENGTHM: Anzahl der Endzeichen, die dem Längenfeld folgen. Diese Zeichen werden bei der Auswertung der Nachrichtenlänge nicht berücksichtigt.
- Zeichenkette: Der Nachrichtenempfang wird beendet, wenn eine festgelegte Zeichenfolge empfangen wird. Die Zeichenkette kann maximal fünf Zeichen enthalten. Für jedes Zeichen der Zeichenkette kann über die Bitposition festgelegt werden, ob dieses bei der Auswertung berücksichtigt oder ignoriert wird.

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
80C0	Fehler in der Startbedingung
80C1	• Fehler in der Endbedingung • Keine Endbedingung definiert
80C2	Empfangsalarm aktiviert
80C3	Bei eingestellter Endbedingung "Maximale Länge" wurde am Parameter MAXLEN ein Wert eingegeben, der gleich 0 oder größer als 4132 ist.
80C4	Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter N ein Wert eingegeben, der größer als 4131 ist.
80C5	Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter LENGTHSIZE ein Wert eingegeben, der gleich 0 oder ungültig ist.
80C6	Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter LENGTHM ein Wert eingegeben, der größer als 255 ist.
80C7	Bei eingestellter Endbedingung "N+LEN+M" wurde eine Nachrichtenlänge errechnet, die größer als 4132 ist.
80C8	Bei eingestellter Endbedingung "Antwort-Zeitüberschreitung" wurde am Parameter RCVTIME ein Wert eingegeben, der gleich 0 ist.
80C9	Bei eingestellter Endbedingung "Zeitüberschreitung innerhalb der Zeichenkette" wurde am Parameter CHARGAP ein Wert eingegeben, der gleich 0 oder größer als 2500 ist.
80CA	Bei eingestellter Startbedingung "Idle line" wurde am Parameter IDLETIME ein Wert eingegeben, der gleich 0 oder größer als 2500 ist.
80CB	Trotz eingestellter Endbedingung "Zeichenkette" sind alle Zeichen der Zeichenkette als irrelevant gekennzeichnet.
80CC	Trotz eingestellter Startbedingung "Zeichenkette" sind alle Zeichen der Zeichenkette als irrelevant gekennzeichnet.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SEND_PTP: Daten des Sendepuffers übertragen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung "SEND_PTP" stoßen Sie die Übertragung von Daten an. Die Anweisung "SEND_PTP" führt dabei nicht die eigentliche Übertragung der Daten durch. Die Daten des Sendepuffers werden an das Punkt-zu-Punkt-Kommunikationsmodul (CM) übertragen. Das CM führt dann die eigentliche Übertragung durch.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND_PTP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Freigabe der angeforderten Übertragung bei steigender Flanke dieses Freigabeeingangs. Dadurch wird der Inhalt des Puffers zum Punkt-zu-Punkt-Kommunikationsmodul (CM) übertragen.
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
BUFFER	Input	VARIANT	E, A, M, D, L oder Konstante	Zeiger auf die Anfangsadresse des Sendepuffers. Boolsche Werte oder Array of BOOL werden nicht unterstützt.
LENGTH	Input	UINT	E, A, M, D, L oder Konstante	Länge des Sendepuffers
PTRCL	Input	BOOL	E, A, M, D, L oder Konstante	Dieser Parameter wählt den Puffer für die normale Punkt-zu-Punkt-Kommunikation oder für spezifische Siemens-Protokolle aus, die im angeschlossenen CM implementiert sind. FALSE = vom Anwenderprogramm gesteuerte Punkt-zu-Punkt-Operationen (einzige gültige Option)
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#...)	Beschreibung
7000	Die Sendeoperation ist nicht aktiv.
7001	Die Sendeoperation verarbeitet den ersten Aufruf.
7002	Die Sendeoperation verarbeitet nachfolgende Aufrufe (Abfragen nach dem ersten Aufruf).
8080	Die eingegebene Kennung des Kommunikationsports ist ungültig.
8088	Die Länge des Parameters LENGHT entspricht nicht der Länge der zu sendenden Daten. Siehe auch: Parameter LENGHT und BUFFER.
80D0	Eine neue Sendeanforderung wurde während einer laufenden Übertragung empfangen.
80D1	Die Übertragung wurde abgebrochen, weil das CTS-Signal innerhalb der angegebenen Wartezeit nicht bestätigt wurde.
80D2	Die Sendeanforderung wurde abgebrochen, da der Kommunikationspartner (DCE) keine Empfangsbereitschaft (DSR) signalisiert.
80D3	Die Sendeanforderung wurde abgebrochen, da die maximale Größe der Warteschleife überschritten wurde (mehr als 1024 Byte).
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Parameter LENGTH und BUFFER

Die Mindestdatengröße, die von der Anweisung "PTP_SEND" gesendet werden kann, ist ein Byte. Der Parameter BUFFER legt die Größe der zu sendenden Daten fest. Sie können für den Parameter BUFFER weder den Datentyp BOOL noch Array of BOOL verwenden.

Parameter LENGTH	Parameter BUFFER	Beschreibung
LENGTH = 0	Nicht verwendet	Die vollständigen Daten werden wie vom Parameter BUFFER definiert gesendet. Ist LENGTH = 0, brauchen Sie die Anzahl der übertragenen Bytes nicht anzugeben.
LENGTH > 0	Elementarer Datentyp	Der LENGTH-Wert muss die Bytezahl dieses Datentyps enthalten. Andernfalls erfolgt keine Übertragung, und der Fehler 8088 wird ausgegeben.
	STRUCT	Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge der Struktur enthalten. In diesem Fall werden nur die ersten LENGTH-Bytes übertragen.
	ARRAY	Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge des Felds enthalten. In diesem Fall werden nur die Feldelemente übertragen, die vollständig in die LENGTH-Bytes passen. Der LENGTH-Wert muss ein Vielfaches der Bytezahl der Datenelemente sein. Andernfalls ist STATUS = 8088, ERROR = 1 und keine Übertragung erfolgt.
	STRING	Die vollständige Speicheranordnung des Zeichenkettenformats wird übertragen, sowie die die Information über maximale Länge der Zeichenkette und aktuelle Länge der Zeichenkette. Der LENGTH-Wert muss Bytes für maximale Länge, tatsächliche Länge und die Zeichen der Zeichenkette enthalten. Beim Datentyp STRING haben alle Längen und Zeichen die Größe eines Bytes. Wird für den Parameter BUFFER eine Zeichenkette verwendet, muss der LENGTH-Wert auch zwei Bytes für die beiden Längenfelder enthalten.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RCV_PTP: Empfang von Nachrichten freigeben

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung "RCV_PTP" geben Sie den Empfang einer gesendeten Nachricht frei. Dabei muss jede Nachricht einzeln freigegeben werden. Die gesendeten Daten stehen im Empfangsbereich erst dann zur Verfügung, wenn die Nachricht vom entsprechenden Kommunikationspartner quittiert wurde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV_PTP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L oder Konstante	Freigabe des Empfangs bei einer steigenden Flanke
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
BUFFER	Input	VARIANT	E, A, M, D, L oder Konstante	Zeigt auf die Anfangsadresse des Empfangspuffers. Verwenden Sie in dem Empfangspuffer keine Variable vom Typ STRING.
NDR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung
LENGTH	Output	UINT	E, A, M, D, L	Länge der Nachricht im Empfangspuffer

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#....)	Beschreibung
80E0	Der Nachrichtenempfang wurde beendet, weil der Empfangspuffer voll ist.
80E1	Der Nachrichtenempfang wurde aufgrund eines Paritätsfehlers beendet.
80E2	Der Nachrichtenempfang wurde aufgrund eines Framing-Fehlers beendet.
80E3	Der Nachrichtenempfang wurde aufgrund eines Überlauffehlers beendet.
80E4	Der Nachrichtenempfang wurde beendet, weil die errechnete Nachrichtenlänge (N+LEN+M) die Größe des Empfangspuffers überschreitet.
8080	Die eingegebene Kennung des Kommunikationsports ist ungültig.
8088	Über den Parameter BUFFER wurde auf einen Datentyp STRING referenziert.
0094	Der Nachrichtenempfang wurde beendet, weil die maximale Zeichenlänge empfangen wurde.
0095	Der Nachrichtenempfang wurde aufgrund einer Zeitüberschreitung beendet.
0096	Der Nachrichtenempfang wurde aufgrund einer Zeitüberschreitung innerhalb der Zeichenkette beendet.

Fehlercode* (W#16#....)	Beschreibung
0097	Der Nachrichtenempfang wurde aufgrund einer Antwort-Zeitüberschreitung beendet.
0098	Der Nachrichtenempfang wurde beendet, weil die Längenbedingung "N+LEN+M" erfüllt wurde.
0099	Der Nachrichtenempfang wurde beendet, weil die als Endbedingung definierte Zeichenkette empfangen wurde.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

RCV_RST: Empfangspuffer löschen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung "RCV_RST" löschen Sie den Empfangspuffer eines Kommunikationspartners.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV_RST":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Freigabe zum Löschen des Empfangspuffers bei einer steigenden Flanke
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

SGN_GET: RS-232-Signale abfragen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung "SGN_GET" fragen Sie den aktuellen Zustand mehrerer Signale einer RS-232-Kommunikationsbaugruppe ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SGN_GET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Freigabe der Abfrage bei einer steigenden Flanke
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID)
NDR	Output	BOOL	E, A, M, D, L	Wird für einen Zyklus gesetzt, wenn neue Daten bereit stehen und die Anweisung fehlerfrei ausgeführt wurde.
DTR	Output	BOOL	E, A, M, D, L	Datenendgerät bereit, Baugruppe bereit
DSR	Output	BOOL	E, A, M, D, L	Datensatz bereit, Kommunikationspartner bereit
RTS	Output	BOOL	E, A, M, D, L	Sendeanforderung, Baugruppe sendebereit
CTS	Output	BOOL	E, A, M, D, L	Sendebereit, Kommunikationspartner kann Daten empfangen (Reaktion auf RTS = EIN der Baugruppe).
DCD	Output	BOOL	E, A, M, D, L	Datenträgererkennung, Empfangssignalpegel
RING	Output	BOOL	E, A, M, D, L	Klingelanzeige, Anzeige eines eingehenden Anrufs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#....)	Beschreibung
80F0	Die Kommunikationsbaugruppe ist eine RS485-Baugruppe, es stehen keine Signale zur Verfügung.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

SGN_SET: RS-232-Signale setzen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung "SGN_SET" setzen Sie den Zustand der Ausgangssignale einer RS-232-Kommunikationsbaugruppe.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SGN_SET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Aktiviert den Vorgang bei einer steigenden Flanke Initialwert: FALSE
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports (HW-ID) Initialwert: FALSE
SIGNAL	Input	BYTE	E, A, M, D, L oder Konstante	Gib die Signale an, die gesetzt werden: • 01H = RTS setzen • 02H = DTR setzen • 04H = DSR setzen Initialwert: FALSE
RTS	Input	BOOL	E, A, M, D, L oder Konstante	Sendeanforderung, Baugruppe sendebereit Initialwert: FALSE
DTR	Input	BOOL	E, A, M, D, L oder Konstante	Datenendgerät bereit, Baugruppe bereit Initialwert: FALSE
DSR	Input	BOOL	E, A, M, D, L oder Konstante	Datensatz bereit (gilt nur für Schnittstellen vom Typ DCE) Initialwert: FALSE
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt Initialwert: FALSE
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Kein Fehler • 1: Fehler aufgetreten Initialwert: FALSE
STATUS	Output	WORD	E, A, M, D, L	Status der Anweisung Initialwert: 0

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#....)	Beschreibung
80F0	Die Kommunikationsbaugruppe ist eine RS485-Baugruppe, es stehen keine Signale zur Verfügung.
80F1	Es können keine Signale gesetzt werden, da die Hardware-Flusskontrolle aktiviert ist.
80F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.
80F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 4240)".

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Allgemeine Statusinformationen der Kommunikationsbausteine

Allgemeine Informationen zum Ausführungsstatus der Kommunikationsbausteine

Die folgende Tabelle zeigt, welche allgemeinen Informationen am Parameter STATUS der Kommunikationsbausteine ausgegeben werden können:

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
7000	Keine Auftragsbearbeitung aktiv.
7001	Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.
7002	Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".
8x3A	Unzulässiger Zeiger am Parameter x.
8070	Alle internen Instanz-Speicher sind im Einsatz.
8080	Die eingegebene Kennung des Kommunikationsports ist ungültig
8081	Zeitüberschreitung, Baugruppenfehler, interner Fehler
8082	Parametrierung fehlgeschlagen, da Parametrierung im Hintergrund gerade durchgeführt wird.
8083	Pufferüberlauf:Das CM oder CB hat eine Empfangsmeldung mit einer Länge, die größer ist als vom Längenparameter zugelassen, zurückgegeben.
8085	Fehler bei der Längenangabe am Parameter LENGHT. Die angegebene Länge ist "0" oder größer als der maximal zulässige Wert.
8090	Länge der Nachricht ungültig, Baugruppe ungültig, Nachricht ungültig
8091	Fehlerhafter Typ in der Parametrierungsnachricht
8092	Datensatzlänge in der Parametrierungsnachricht ungültig
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

USS

Überblick über die USS-Anweisungen

Einleitung

Die USS-Anweisungen steuern den Betrieb von Antrieben, die das Protokoll der universellen seriellen Schnittstelle (USS) unterstützen. Mit den USS-Anweisungen können Sie über eine RS-485-Verbindung mit mehreren Antrieben kommunizieren.

Hierzu benötigen Sie ein CM 1241 RS485-Kommunikationsmodul oder ein CB 1241 RS485-Kommunikationsboard. In einer S7-1200 CPU können bis zu drei CM 1241 RS485-Module und ein CB 1241 RS485-Board eingebaut werden.

Jeder RS485-Port kann bis zu sechszehn Antriebe betreiben.

Das USS-Protokoll nutzt ein Master/Slave-Netzwerk für die Kommunikation über einen seriellen Bus. Der Master verwendet einen Adressparameter, um eine Meldung an einen ausgewählten Slave zu senden. Ein Slave selbst kann niemals senden, ohne dafür zuvor eine Anforderung zu erhalten. Die direkte Meldungsübertragung zwischen den einzelnen Slaves ist nicht möglich. Die USS-Kommunikation funktioniert im Halbduplex-Betrieb.

Das folgende Bild zeigt ein Beispiel eines USS-Netzwerkdiagramms:

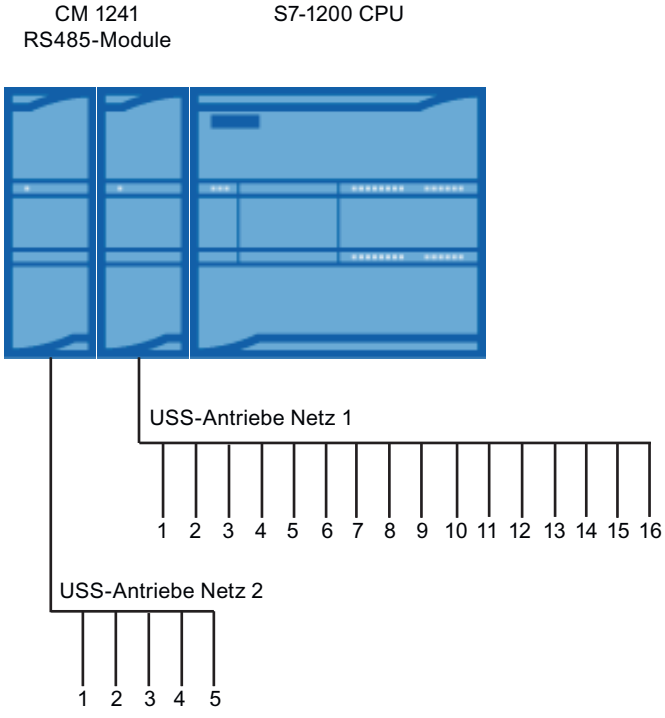

Voraussetzungen für den Einsatz des USS-Protokolls

Allgemeine Voraussetzungen für die Antriebseinrichtung

- Für die Antriebe muss die Verwendung von 4 PKW-Wörtern eingerichtet werden.
- Die Antriebe können für 2, 4, 6 oder 8 PZD-Wörter konfiguriert werden.
- Die Anzahl der PZD-Wörter im Antrieb muss dem Eingang PZD_LEN der Anweisung "USS_DRIVE (Seite 4246)" des Antriebs entsprechen.
- Die Baudrate aller Antriebe muss dem der Baudrate am Eingangsparameter BAUD der Anweisung "USS_PORT (Seite 4245)" entsprechen.
- Der Antrieb muss für die Fernsteuerung eingerichtet werden.
- Für den Frequenzsollwert an der COM-Verbindung des Antriebs muss USS festgelegt werden.
- Für die Antriebsadresse muss 1 bis 16 festgelegt sein. Diese Adresse muss der Adresse am Eingangsparameter DRIVE der Anweisung "USS_DRIVE (Seite 4246)" entsprechen.
- Für die Richtungssteuerung des Antriebs muss die Verwendung der Polarität des Antriebssollwerts eingerichtet werden.
- Das RS485-Netzwerk muss ordnungsgemäß abgeschlossen sein.

Definition: PKW- / PZD-Bereich

- Der PKW-Bereich bezieht sich dabei auf das Handling der Parameter-Kennung-Wert (PKW)-Schnittstelle.
  Die PKW-Schnittstelle ist keine physikalische Schnittstelle, sondern es wird damit ein Mechanismus beschrieben, der den Parameteraustausch zwischen zwei Kommunikationspartnern regelt. D. h., Lesen und Schreiben von Parameterwerten, Parameterbeschreibungen und zugehörige Texten sowie die Behandlung von Parameteränderung durch Spontanmeldungen. Alle Aufgaben, die über die PKW-Schnittstelle erfolgen, sind im wesentlichen Aufgaben für Bedienen und Beobachten, Service und Diagnose.
- Der PZD-Bereich beinhaltet die für die Automatisierung notwendigen Signale:
  - Steuerwort(e) und Sollwert(e) vom Master zum Slave
  - Zustandswort(e) und Istwert(e) vom Slave zum Master.

Beide Bereiche zusammen ergeben den Nutzdatenblock. Dieser wird als Auftragstelegramm vom Master an den Slave übertragen oder als Antworttelegramm vom Slave an den Master.

Beschreibung

Jedes Kommunikationsmodul CM 1241 RS485 unterstützt maximal 16 Antriebe. Ein einziger Instanz-Datenbaustein enthält temporäre Speicher- und Pufferfunktionen für alle Antriebe im USS-Netzwerk, die mit einem von Ihnen installierten PtP-Kommunikationsmodul verbunden sind. Die USS-Anweisungen für diese Antriebe haben gemeinsam Zugriff auf die Informationen in diesem Datenbaustein.

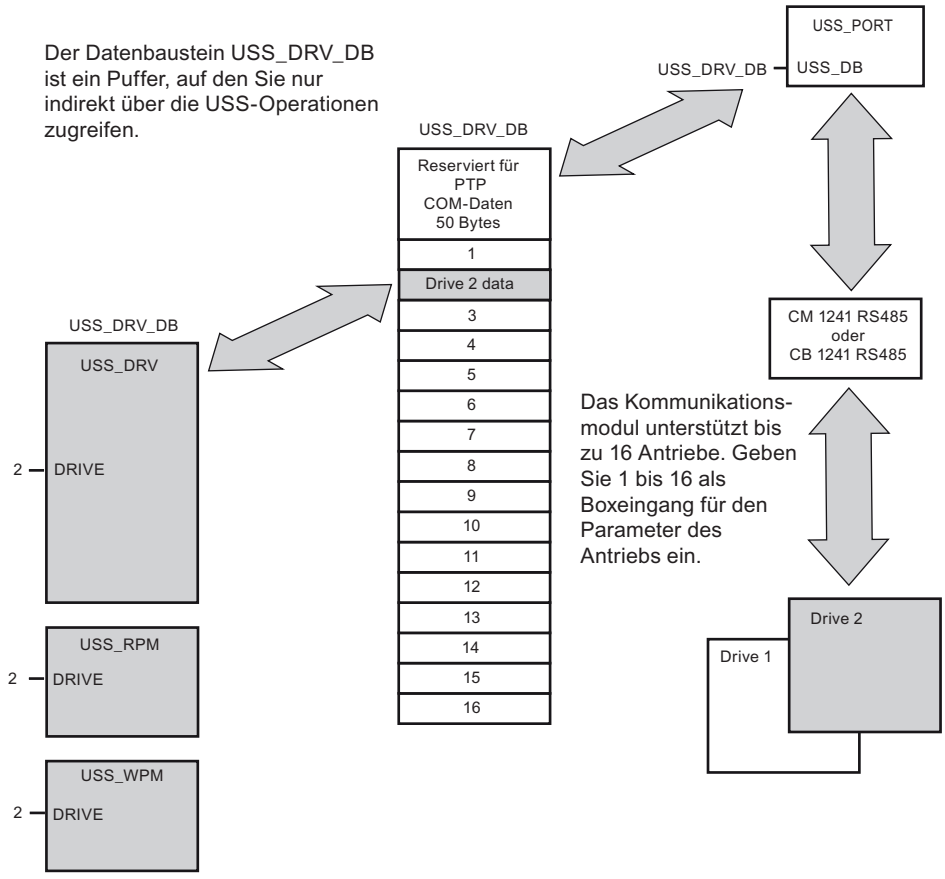

- Alle Antriebe (max. 16), die an einen RS485 Port angeschlossen sind, sind Teil desselben USS-Netzwerks. Alle Antriebe, die an einen anderen RS485 Port angeschlossen sind, sind Teil eines anderen USS-Netzwerks. Weil die S7-1200 bis zu drei Module CM 1241 RS485 unterstützt, können Sie bis zu drei USS-Netzwerke einrichten, jeweils mit maximal 16 Antrieben in jedem Netzwerk, so dass insgesamt 48 USS-Antriebe unterstützt werden.
- Jedes USS-Netzwerk wird über einen eindeutigen Datenbaustein verwaltet (für drei USS-Netzwerke mit drei Modulen CM 1241 RS485 sind drei Datenbausteine erforderlich). Alle Anweisungen, die zu einem USS-Netzwerk gehören, müssen diesen Datenbaustein gemeinsam nutzen. Dies umfasst alle Anweisungen "USS_DRIVE (Seite 4246)", "USS_PORT (Seite 4245)", "USS_RPM (Seite 4249)" und "USS_WPM (Seite 4251)" für die Steuerung aller Antriebe in einem USS-Netzwerk.

- Die Anweisung "USS_DRIVE (Seite 4246)" ist ein Funktionsbaustein (FB). Wenn Sie die Anweisung "USS_DRIVE" im Editor einfügen, werden Sie im Dialog "Aufrufoptionen" aufgefordert, der Anweisung einen DB zuzuweisen.
  - Wenn es sich um die erste Anweisung "USS_DRIVE" in diesem Programm für dieses USS-Netzwerk handelt, können Sie die DB-Standardzuweisung übernehmen (oder ggf. den Namen ändern), und der neue DB wird erstellt.
  - Wenn es sich jedoch nicht um die erste Anweisung "USS_DRIVE" für dieses Netzwerk handelt, müssen Sie im Dialog "Aufrufoptionen" in der Klappliste den entsprechenden DB auswählen, der diesem USS-Netzwerk bereits zuvor zugewiesen wurde.
- Bei allen Anweisungen "USS_PORT (Seite 4245) , USS_RPM (Seite 4249)" und "USS_WPM (Seite 4251)" handelt es sich um Funktionen (FCs). Wenn Sie diese FCs im Editor einfügen, wird kein DB zugewiesen. Stattdessen müssen Sie dem Eingang USS_DB dieser Anweisungen den jeweiligen DB zuweisen (doppelklicken Sie auf das Parameterfeld und dann auf das Symbol, um die verfügbaren DBs anzuzeigen).
- Die Anweisung "USS_PORT (Seite 4245)" steuert die Kommunikation zwischen der CPU und den Antrieben über das PtP-Kommunikationsmodul. Bei jedem Aufruf dieser Anweisung wird eine Kommunikation mit einem Antrieb bearbeitet. Ihr Programm muss diese Funktion schnell genug aufrufen, so dass die Antriebe keine Zeitüberschreitung melden. Die Anweisung kann aus dem Hauptprogramm oder einem beliebigen Alarm-OB aufgerufen werden.
- Der Funktionsbaustein "USS_DRIVE (Seite 4246)" gibt Ihrem Programm Zugriff auf einen angegebenen Antrieb im USS-Netzwerk. Seine Ein- und Ausgänge entsprechen den Zuständen und den Bedienfunktionen des Antriebs. Sind 16 Antriebe im Netzwerk vorhanden, so muss "USS_DRIVE" in Ihrem Programm mindestens 16mal aufgerufen werden, also jeweils einmal für jeden Antrieb. Wie schnell diese Bausteine aufgerufen werden, hängt von der erforderlichen Geschwindigkeit für die Steuerung der Antriebsfunktionen ab.
  Sie können die Anweisung "USS_DRIVE" nur aus dem Hauptprogramm-OB aufrufen.

  ⚠ VORSICHT

  Rufen Sie "USS_DRIVE", "USS_RPM", "USS_WPM" nur aus dem Hauptprogramm-OB auf. Die Anweisung "USS_PORT" kann aus einem beliebigen OB aufgerufen werden, üblicherweise wird sie aus einem Verzögerungsalarm-OB aufgerufen. Wird die Anweisung "USS_PORT" während der Ausführung unterbrochen, kann es zu unerwarteten Fehlern kommen.

Mit den Anweisungen "USS_RPM" und "USS_WPM" werden die Betriebsparameter des Antriebs gelesen und geschrieben. Diese Parameter steuern die interne Funktionsweise des Antriebs. Eine Definition dieser Parameter finden Sie im Handbuch des Antriebs.

Ihr Programm kann eine beliebige Anzahl dieser Anweisungen enthalten, es kann jedoch immer nur eine Lese- oder Schreibanforderung für einen Antrieb aktiv sein. Sie dürfen die Anweisungen "USS_RPM" und "USS_WPM" nur aus einem Hauptprogramm-OB aufrufen.

Zeit für die Kommunikation mit dem Antrieb berechnen

Die Kommunikation mit dem Antrieb läuft asynchron zum Zyklus der S7-1200 ab. Die S7-1200 durchläuft üblicherweise mehrere Zyklen, bevor die Kommunikation mit einem Antrieb beendet ist.

Das Intervall von "USS_PORT" ist die Zeit, die für eine Transaktion des Antriebs erforderlich ist. Die folgende Tabelle zeigt die Mindestintervalle für "USS_PORT" für jede Baudrate. Wenn Sie die Anweisung "USS_PORT" häufiger aufrufen, als es das "USS_PORT"-Intervall vorgibt, wird die Anzahl der Transaktionen nicht erhöht. Das Timeout-Intervall des Antriebs ist die Zeitdauer, die für eine Transaktion zur Verfügung steht, wenn zur Fertigstellung der Transaktion aufgrund von Kommunikationsfehlern 3 Versuche nötig sind. Standardmäßig werden beim USS-Protokoll bei jeder Transaktion bis zu 2 Wiederholungen durchgeführt.

Baudrate	Berechnetes Mindestintervall für Aufruf von USS_PORT (ms)	Intervall-Timeout für Antriebsmeldung pro Antrieb (ms)
1200	790	2370
2400	405	1215
4800	212,5	638
9600	116,3	349
19200	68,2	205
38400	44,1	133
57600	36,1	109
115200	28,1	85

USS_PORT: Kommunikation über USS-Netzwerk bearbeiten

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "USS_PORT" bearbeitet die Kommunikation über das USS-Netzwerk. Verwenden Sie im Programm jeweils eine Anweisung "USS_PORT" pro PtP-Kommunikationsport, um die Übertragung zu oder von jeweils einem Antrieb zu steuern.

Alle USS-Anweisungen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den gleichen Instanz-Datenbaustein nutzen.

Aufruf

Ihr Programm muss die Anweisung "USS_PORT" oft genug ausführen, damit keine Zeitüberschreitungen im Antrieb auftreten. Rufen Sie daher die Anweisung "USS_PORT" aus einem Weckalarm-OB auf, um Zeitüberschreitungen beim Antrieb zu verhindern und die letzten USS-Datenaktualisierungen für Aufrufe von "USS_DRIVE (Seite 4246)" verfügbar zu haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS_PORT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PORT	Input	PORT	D, L oder Konstante	PtP-Kommunikationsport-Kennung Konstante, auf die in der Standardvariablentabelle im Register "Konstanten" verwiesen werden kann.
BAUD	Input	DINT	E, A, M, D, L oder Konstante	Baudrate für die USS-Kommunikation.
USS_DB	InOut	USS_BASE	D	Referenz auf den Instanz-DB der Anweisung "USS_DRIVE (Seite 4246)".
ERROR	Output	BOOL	E, A, M, D, L	Tritt ein Fehler auf, wird ERROR auf TRUE gesetzt. Am Ausgang STATUS wird ein entsprechender Fehlercode ausgegeben.
STATUS (Seite 4252)	Output	WORD	E, A, M, D, L	Zustandswert der Anforderung. Er gibt das Ergebnis des Zyklus oder der Initialisierung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 4252)" zur Verfügung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

USS_DRIVE: Daten mit dem Antrieb austauschen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "USS_DRIVE" tauscht Daten mit dem Antrieb aus, indem Anfragemeldungen erzeugt und die Antwortmeldungen des Antriebs ausgewertet werden. Für jeden Antrieb muss eine eigene Anweisung verwendet werden, doch müssen alle USS-Anweisungen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, den gleichen Instanz-Datenbaustein verwenden. Sie müssen den DB-Namen eingeben, wenn Sie die erste Anweisung "USS_DRIVE" einfügen. Dann nutzen Sie diesen DB weiter, der beim Einfügen der ersten Anweisung angelegt wurde.

Wenn die erste Ausführung von "USS_DRIVE" erfolgt, wird der von der USS-Adresse (Parameter DRIVE) angegebene Antrieb im Instanz-DB initialisiert. Nach dieser Initialisierung können nachfolgende Anweisungen "USS_PORT (Seite 4245)" die Kommunikation mit dem Antrieb an dieser Antriebsnummer beginnen.

Wenn Sie die Antriebsnummer ändern, muss das Zielsystem zunächst in STOP und dann wieder in RUN versetzt werden, damit der Instanz-DB initialisiert wird. Die Eingangsparameter werden im USS-Sendepuffer konfiguriert und die Ausgänge werden, sofern vorhanden, aus einem "vorherigen" gültigen Antwortpuffer gelesen. Während der Ausführung der Anweisung

"USS_DRIVE" findet keine Datenübertragung statt. Nach der Ausführung von "USS_PORT (Seite 4245)" wird mit den Antrieben kommuniziert. "USS_DRIVE" konfiguriert nur die zu sendenden Meldungen und wertet Daten aus, die in einer vorherigen Anforderung empfangen wurden.

Sie können die Drehrichtung des Antriebs entweder über den Eingang DIR (BOOL) oder über das Vorzeichen (positiv oder negativ) am Eingang SPEED_SP (REAL) steuern. Die folgende Tabelle erläutert, wie diese Eingänge zusammen funktionieren, um die Drehrichtung des Antriebs zu bestimmen, vorausgesetzt der Motor dreht vorwärts.

SPEED_SP	DIR	Drehrichtung des Antriebs
Wert > 0	0	Rückwärts
Wert > 0	1	Vorwärts
Wert < 0	0	Vorwärts
Wert < 0	1	Rückwärts

Parameter

Erweitern Sie die Box, um alle Parameter anzuzeigen. Klicken Sie dazu auf den unteren Bereich der Box. Die Parameteranschlüsse, die grau dargestellt sind, sind optional und müssen nicht zugewiesen werden.

Die folgende Tabelle zeigt die Parameter der Anweisung "USS_DRIVE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RUN	Input	BOOL	E, A, M, D, L oder Konstante	Startbit des Antriebs: Hat dieser Parameter den Wert TRUE, so ermöglicht dieser Eingang den Betrieb des Antriebs mit der voreingestellten Drehzahl.
OFF2	Input	BOOL	E, A, M, D, L oder Konstante	Bit "Zum Stillstand auslaufen": Hat dieser Parameter den Wert FALSE, so veranlasst dieses Bit das Auslaufen des Antriebs, ohne zu bremsen.
OFF3	Input	BOOL	E, A, M, D, L oder Konstante	Schnelles Stoppbit - Hat dieser Parameter den Wert FALSE, so verursacht dieses Bit einen schnellen Halt durch Abbremsen des Antriebs.
F_ACK	Input	BOOL	E, A, M, D, L oder Konstante	Fehlerquittierungsbit - Mit diesem Bit wird das Fehlerbit eines Antriebs zurückgesetzt. Es wird nach dem Löschen des Fehlers gesetzt und der Antrieb erkennt damit, dass der vorherige Fehler nicht mehr gemeldet werden muss.
DIR	Input	BOOL	E, A, M, D, L oder Konstante	Richtungssteuerung des Antriebs - Dieses Bit wird gesetzt, wenn der Antrieb in Vorwärtsrichtung laufen soll (wenn SPEED_SP positiv ist).
DRIVE	Input	USINT	E, A, M, D, L oder Konstante	Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PZD_LEN	Input	USINT	E, A, M, D, L oder Konstante	Wortlänge - Dies ist die Anzahl der PZD-Datenwörter. Gültige Werte sind 2, 4, 6 oder 8 Wörter. Voreinstellung ist 2.
SPEED_SP	Input	REAL	E, A, M, D, L oder Konstante	Drehzahlsollwert - Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Frequenz. Ein positiver Wert bedeutet, dass der Antrieb vorwärts läuft (wenn DIR dem Wert TRUE hat).

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CTRL3	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 3 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.
CTRL4	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 4 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.
CTRL5	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 5 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.
CTRL6	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 6 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren.
CTRL7	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 7 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.
CTRL8	Input	WORD	E, A, M, D, L oder Konstante	Steuerwort 8 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.
NDR	Output	BOOL	E, A, M, D, L	Neue Daten bereit - Hat dieser Parameter den Wert TRUE, so meldet das Bit, dass am Ausgang Daten einer neuen Kommunikationsanforderung bereitstehen.
ERROR	Output	BOOL	E, A, M, D, L	Fehler aufgetreten - Hat dieser Parameter den Wert TRUE, so meldet dieses Bit, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT" gemeldet.
STATUS (Seite 4252)	Output	WORD	E, A, M, D, L	Zustandswert der Anforderung. Er gibt das Zyklusergebnis an. Dies ist kein vom Antrieb ausgegebenes Zustandswort.
RUN_EN	Output	BOOL	E, A, M, D, L	Betrieb freigegeben - Dieses Bit meldet, ob der Antrieb läuft.
D_DIR	Output	BOOL	E, A, M, D, L	Antriebsrichtung - Dieses Bit meldet, ob der Antrieb vorwärts läuft.
INHIBIT	Output	BOOL	E, A, M, D, L	Antrieb gesperrt - Dieses Bit meldet den Zustand des Sperrbits für den Antrieb.
FAULT	Output	BOOL	E, A, M, D, L	Antriebsfehler - Dieses Bit meldet, dass im Antrieb ein Fehler aufgetreten ist. Der Anwender muss die Störung beheben und Bit F_ACK setzen, um dieses Bit zu löschen.
SPEED	Output	REAL	E, A, M, D, L	Istwert Antriebsdrehzahl (skalierter Wert von Zustandswort 2 des Antriebs) - Der Wert der Drehzahl des Antriebs prozentual zur konfigurierten Drehzahl.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS1	Output	WORD	E, A, M, D, L	Zustandswort 1 des Antriebs - Dieser Wert enthält feste Zustandsbits eines Antriebs.
STATUS3	Output	WORD	E, A, M, D, L	Zustandswort 3 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS4	Output	WORD	E, A, M, D, L	Zustandswort 4 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS5	Output	WORD	E, A, M, D, L	Zustandswort 5 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS6	Output	WORD	E, A, M, D, L	Zustandswort 6 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS7	Output	WORD	E, A, M, D, L	Zustandswort 7 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS8	Output	WORD	E, A, M, D, L	Zustandswort 8 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

USS_RPM: Parameter aus dem Antrieb auslesen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "USS_RPM" liest einen Parameter aus dem Antrieb. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, müssen den gleichen Datenbaustein nutzen. "USS_RPM" muss aus dem Hauptprogramm-OB aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS_RPM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Sendeanforderung: Hat dieser Parameter den Wert TRUE, so wird eine neue Leseanforderung benötigt. Dies wird ignoriert, wenn die Anforderung für diesen Parameter bereits ansteht.
DRIVE	Input	USINT	E, A, M, D, L oder Konstante	Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PARAM	Input	UINT	E, A, M, D, L oder Konstante	Parameternummer: Dieser Eingang gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich dieses Parameters ist 0 bis 2047. Weitere Informationen für den Zugriff auf Parameter oberhalb dieses Bereichs finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	Input	UINT	E, A, M, D, L oder Konstante	Parameterindex: Dieser Eingang gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höherwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.
USS_DB	InOut	USS_BASE	D	Verweis auf den Instanz-DB, der erstellt und initialisiert wird, wenn eine Anweisung "USS_DRIVE" in Ihr Programm eingefügt wird.
DONE	Output	BOOL	E, A, M, D, L	Hat dieser Parameter den Wert TRUE, so steht am Ausgang VALUE der zuvor angeforderte Wert des Leseparameters an. Dieses Bit wird gesetzt, wenn die Anweisung "USS_DRIVE" die Leseantwort des Antriebs erkennt. Dieses Bit wird zurückgesetzt, wenn: • Sie die Antwortdaten über eine andere Abfrage "USS_RPM" anfordern oder • der zweite der nächsten beiden Aufrufe von "USS_DRIVE (Seite 4246)" ausgeführt wird
ERROR	Output	BOOL	E, A, M, D, L	Fehler aufgetreten - Hat dieser Parameter den Wert TRUE, so meldet dieses Bit, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT (Seite 4245)" gemeldet.
STATUS (Seite 4252)	Output	WORD	E, A, M, D, L	Dies ist der Zustandswert der Anforderung. Er gibt das Ergebnis der Leseanforderung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 4252)" zur Verfügung.
VALUE	Output	VARIANT	E, A, M, D, L	Dies ist der Wert des Parameters, der gelesen wurde und er ist nur gültig, wenn das Bit DONE den Wert TRUE hat.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

USS_WPM: Parameter im Antrieb ändern

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung "USS_WPM" ändert einen Parameter im Antrieb. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, müssen den gleichen Datenbaustein nutzen. "USS_WPM" muss aus dem Hauptprogramm-OB aufgerufen werden.

Hinweis

EEPROM-Schreiboperationen

Übertreiben Sie die Verwendung der EEPROM-Schreiboperation nicht. Halten Sie die Anzahl der EEPROM-Schreiboperationen möglichst gering, um die Lebensdauer des EEPROM zu verlängern.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS_WPM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Sendeanforderung: Hat dieser Parameter den Wert TRUE, so wird eine neue Schreibanforderung benötigt. Dies wird ignoriert, wenn die Anforderung für diesen Parameter bereits ansteht.
DRIVE	Input	USINT	E, A, M, D, L oder Konstante	Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PARAM	Input	UINT	E, A, M, D, L oder Konstante	Parameternummer: Dieser Eingang gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich dieses Parameters ist 0 bis 2047. Weitere Informationen für den Zugriff auf Parameter oberhalb dieses Bereichs finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	Input	UINT	E, A, M, D, L oder Konstante	Parameterindex: Dieser Eingang gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höherwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EEPROM	Input	BOOL	E, A, M, D, L oder Konstante	In EEPROM des Antriebs speichern: Hat dieser Parameter den Wert TRUE, so werden in den Antriebsparameter geschriebene Werte im EEPROM des Antriebs gespeichert. Hat dieser Parameter den Wert FALSE, so wird der geschriebene Wert nur temporär gespeichert und geht beim nächsten Einschalten des Antriebs verloren.
VALUE	Input	VARIANT	E, A, M, D, L oder Konstante	Wert des Parameters, in den geschrieben werden soll. Er muss beim Zustandswechsel von REQ gültig sein.
USS_DB	InOut	USS_BASE	D	Dies ist ein Verweis auf den Instanz-DB, der erstellt und initialisiert wird, wenn eine Anweisung "USS_DRIVE (Seite 4246)" in Ihr Programm eingefügt wird.
DONE	Output	BOOL	E, A, M, D, L	Hat dieser Parameter den Wert TRUE, wurde der Eingang VALUE in den Antrieb geschrieben. Dieses Bit wird gesetzt, wenn die Anweisung "USS_DRIVE (Seite 4246)" die Schreibantwort des Antriebs erkennt. Dieses Bit wird zurückgesetzt, wenn: Sie die Bestätigung des Antriebs über eine weitere Abfrage "USS_WPM" anfordern, dass der Schreibvorgang durchgeführt wurde, oder wenn der zweite der nächsten beiden Aufrufe von "USS_DRIVE (Seite 4246)" ausgeführt wird.
ERROR	Output	BOOL	E, A, M, D, L	Fehler aufgetreten: Hat dieser Parameter den Wert TRUE, ist ein Fehler aufgetreten und wird am Ausgang STATUS angezeigt. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT (Seite 4245)" gemeldet.
STATUS (Seite 4252)	Output	WORD	E, A, M, D, L	Dies ist der Zustandswert der Anforderung. Er gibt das Ergebnis der Schreibanforderung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 4252)" zur Verfügung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS der USS-Anweisungen

Parameter STATUS

Die folgende Tabelle enthält die Statuscodes der USS-Operation, die am Ausgang STATUS der USS-Anweisungen ausgegeben werden.

STATUS* (W#16#....)	Beschreibung
0000	Kein Fehler
8180	Die Länge der Antwort des Antriebs entsprach nicht den vom Antrieb empfangenen Zeichen. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.

STATUS* (W#16#....)	Beschreibung
8181	Der Parameter VALUE ist nicht vom Datentyp WORD, REAL oder DWORD
8182	Der Anwender hat einen Parameterwert vom Typ Wort eingegeben und die Antwort vom Antrieb im Format DWORD oder REAL empfangen
8183	Der Anwender hat einen Parameterwert vom Typ DWORD oder REAL eingegeben und die Antwort vom Antrieb im Format Wort empfangen
8184	Das Antworttelegramm des Antriebs hatte eine falsche Prüfsumme. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.
8185	Unzulässige Antriebsadresse (gültiger Adressbereich für Antriebe: 1-16)
8186	Drehzahlsollwert liegt außerhalb des gültigen Bereichs (gültiger Sollwertbereich für die Drehzahl: -200% bis 200%)
8187	Falsche Antriebsnummer hat auf die gesendete Anforderung geantwortet. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.
8188	Unzulässige PZD-Wortlänge angegeben (gültiger Bereich = 2, 4, 6 oder 8 Wörter)
8189	Unzulässige Baudrate angegeben
818A	Anforderungskanal für Parameter wird von einer anderen Anforderung für diesen Antrieb verwendet
818B	Antrieb hat nicht auf Anforderungen und Wiederholungen reagiert. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.
818C	Antrieb hat einen erweiterten Fehler zu einer Parameteranforderung ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.
818D	Antrieb hat einen Fehler wegen unzulässigen Zugriffs bei einer Parameteranforderung ausgegeben. Weitere Informationen dazu, weshalb der Parameterzugriff begrenzt sein kann, finden Sie im Handbuch zu Ihrem Antrieb.
818E	Antrieb wurde nicht initialisiert: Dieser Fehlercode wird an "USS_RPM (Seite 4249)" oder "USS_WPM (Seite 4251)" ausgegeben, wenn die Anweisung "USS_DRIVE (Seite 4246)" nicht mindestens einmal für diesen Antrieb aufgerufen wurde. Dadurch wird verhindert, dass die Initialisierung im ersten Zyklus von "USS_DRIVE (Seite 4246)" eine anstehende Anforderung zum Lesen oder Schreiben von Parametern überschreibt, weil dabei der Antrieb als neuer Eintrag initialisiert wird. Um diesen Fehler zu beheben, rufen Sie die Anweisung "USS_DRIVE (Seite 4246)" für diesen Antrieb auf.
80Ax-80Fx	Spezifische Fehler, die von der USS-Bibliothek aufgerufenen Anweisungen für die Punkt-zu-Punkt-Kommunikation (PtP) zurückgegeben werden: Diese Fehlercodes werden von der USS-Bibliothek nicht verändert und sind in den Beschreibungen der PtP-Anweisungen definiert.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

USS_Extended_Error - Zusätzliche Fehlercodes für USS-Antriebe

USS-Antriebe unterstützen Lese- und Schreibzugriff auf die internen Parameter eines Antriebs. Diese Funktion ermöglicht die dezentrale Steuerung und Konfiguration des Antriebs. Zugriffe der Antriebsparameter können aufgrund von Fehlern wie Wert außerhalb des Bereichs oder unzulässige Anforderungen in der aktuellen Betriebsart des Antriebs fehlschlagen. Der Antrieb erzeugt einen Fehlercode, der in der Variablen "USS_Extended_Error" im Instanz-DB der Anweisung "USS_DRIVE (Seite 4246)" ausgegeben wird. Dieser Fehlercode gilt nur für die letzte Ausführung der Anweisung "USS_RPM (Seite 4249)" oder "USS_WPM (Seite 4251)". Der Fehlercode des Antriebs wird in der Variablen "USS_Extended_Error" abgelegt, wenn der Wert von STATUS hexadezimal 818C ist. Der Fehlercode von "USS_Extended_Error" richtet sich nach der Variante des Antriebs. Eine Beschreibung der erweiterten Fehlercodes von Lese- und Schreibfunktionen für Parameter finden Sie im Handbuch des Antriebs.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

MODBUS (RTU)

MB_COMM_LOAD: Port an PtP-Baugruppe für Modbus-RTU konfigurieren

Beschreibung

Die Anweisung "MB_COMM_LOAD" konfiguriert einen Port für die Kommunikation über das Modbus-RTU-Protokoll. Die folgende Hardware kann hierzu verwendet werden:

- Bis zu drei Punkt-zu-Punkt-Baugruppen (PtP) CM 1241 RS485 oder CM 1241 RS232
- Zusätzlich ein Kommunikationsboard CB 1241 RS485

Nach der Konfiguration des Ports kommunizieren Sie über Modbus, indem Sie die Anweisung "MB_SLAVE" oder "MB_MASTER" ausführen.

Aufruf

"MB_COMM_LOAD" muss einmal aufgerufen werden, um den Port für das Modbus-RTU-Protokoll zu konfigurieren. Nach Abschluss der Konfiguration kann der Port von den Anweisungen "MB_MASTER (Seite 4257)" und "MB_SLAVE (Seite 4265)" verwendet werden.

"MB_COMM_LOAD" muss nur dann erneut aufgerufen werden, wenn einer der Kommunikationsparameter geändert werden muss. Jeder Aufruf von "MB_COMM_LOAD" löscht die Kommunikationspuffer. Um Datenverlust bei der Kommunikation zu vermeiden, rufen sie die Anweisung deshalb nicht unnötig auf.

Für die Konfiguration der Ports jedes Kommunikationsmoduls, das für die Modbus-Kommunikation eingesetzt wird, muss eine Instanz von "MB_COMM_LOAD" genutzt werden. Sie müssen jedem Port, den Sie nutzen, einen eindeutigen Instanz-Datenbaustein "MB_COMM_LOAD" zuweisen. Die S7-1200 CPU ist auf drei Kommunikationsmodule begrenzt.

Wenn Sie die Anweisung "MB_MASTER (Seite 4257)" oder "MB_SLAVE (Seite 4265)" einfügen, wird ein Instanz-Datenbaustein zugewiesen. Dieser Instanz-Datenbaustein wird referenziert, indem Sie diesen am Parameter MB_DB der Anweisung "MB_COMM_LOAD" angeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_COMM_LOAD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Ausführung der Anweisung bei steigender Flanke.
PORT	Input	PORT	E, A, M, D, L oder Konstante	Kennung des Kommunikationsports: Nachdem Sie das Kommunikationsmodul in die Gerätekonfiguration eingefügt haben, erscheint die Portkennung in der Klappliste an der Verbindung der PORT-Box. Auf diese Konstante kann auch in der Variablentabelle im Register "Konstanten" verwiesen werden.
BAUD	Input	UDINT	E, A, M, D, L oder Konstante	Auswahl der Baudrate: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 Alle anderen Werte sind ungültig.
PARITY	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der Parität: • 0 – Keine • 1 – Ungerade • 2 – Gerade
FLOW_CTRL	Input	UINT	E, A, M, D, L oder Konstante	Auswahl der Flusskontrolle: • 0 – (Standard) Keine Flusskontrolle • 1 – Hardware-Flusskontrolle mit RTS immer EIN (gilt nicht bei RS485-Ports) • 2 - Hardware-Flusskontrolle mit RTS geschaltet
RTS_ON_DLY	Input	UINT	E, A, M, D, L oder Konstante	Auswahl RTS-Einschaltverzögerung: • 0 – (Standard) Keine Verzögerung von "RTS aktiv", bis das erste Zeichnung der Meldung gesendet wird. • 1 bis 65535 – Verzögerung in Millisekunden von "RTS aktiv", bis das erste Zeichen der Meldung gesendet wird (gilt nicht bei RS-485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RTS_OFF_DLY	Input	UINT	E, A, M, D, L oder Konstante	Auswahl RTS-Ausschaltverzögerung: • 0 – (Standard) Keine Verzögerung nach der Übertragung des letzten Zeichens bis "RTS inaktiv" • 1 bis 65535 – Verzögerung in Millisekunden von der Übertragung des letzten Zeichens bis "RTS inaktiv" (gilt nicht bei RS-485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.
RESP_TO	Input	UINT	E, A, M, D, L oder Konstante	Timeout der Antwort: Zeit in Millisekunden, die "MB_MASTER (Seite 4257)" auf eine Antwort vom Slave wartet. Wenn der Slave während dieses Zeitraums nicht antwortet, wiederholt "MB_MASTER (Seite 4257)" die Anforderung oder beendet die Anforderung mit einem Fehler, sofern die angegebene Anzahl von Wiederholungen gesendet wurde. 5 ms bis 65535 ms (Standardwert = 1000 ms).
MB_DB	Input	MB_BASE	D	Ein Verweis auf den Instanz-Datenbaustein der Anweisungen "MB_MASTER (Seite 4257)" oder "MB_SLAVE (Seite 4265)". Nachdem Sie "MB_SLAVE (Seite 4265)" oder "MB_MASTER (Seite 4257)" in Ihr Programm eingefügt haben, ist die DB-Kennung in der Klappliste an der Verbindung der MB_DB-Box verfügbar.
DONE	Output	BOOL	E, A, M, D, L	Ausführung der Anweisung ohne Fehler beendet.
ERROR	Output	BOOL	E, A, M, D, L	Fehler: • 0 – Kein Fehler erkannt • 1 – Gibt an, dass ein Fehler erkannt wurde. Am Parameter STATUS wird ein Fehlercode ausgegeben.
STATUS	Output	WORD	E, A, M, D, L	Fehlercode der Portprojektierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
8180	Ungültiger Wert für die Port-ID (falsche Adresse für das Kommunikations-Modul).
8181	Ungültiger Wert für die Baudrate.
8182	Ungültiger Wert für die Parität.
8183	Ungültiger Wert für die Flusskontrolle.
8184	Ungültiger Wert für die Zeitüberschreitung der Antwort (der Zeitraum bis eine Zeitüberschreitung gemeldet wird muss mindestens 25ms betragen).

Fehlercode* (W#16#....)	Beschreibung
8185	Fehlerhafter Zeiger am Parameter MB_DB auf den Instanz-DB der Anweisung "MB_MASTER (Seite 4257)" oder "MB_SLAVE (Seite 4265)".
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

MB_COMM_LOAD Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von MB_COMM_LOAD, die Sie in Ihrem Programm verwenden können.

Tabelle 4-182 Statische Variablen im Instanz-DB

Variable	Datentyp	Vorbelegung	Beschreibung
ICHAR_GAP	WORD	0	Verzögerung für den Zeichenabstand zwischen den Zeichen. Dieser Parameter wird in Millisekunden angegeben und dient dazu, den erwarteten Zeitraum zwischen den empfangenen Zeichen zu erhöhen. Die entsprechende Anzahl Bitzeiten für diesen Parameter wird zum Modbus-Standardwert von 35 Bitzeiten (3,5 Zeichenzeiten) addiert.
RETRIES	WORD	2	Die Anzahl der wiederholten Versuche, die der Master durchführt, bevor der Fehlercode 0x80C8 für "Keine Antwort" zurückgegeben wird.
STOP_BITS	USINT	1	Anzahl der Stopp-Bits pro Zeichen. Gültige Werte: 1 und 2

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

MB_MASTER: Als Modbus-Master über PtP-Port kommunizieren

Beschreibung MB_MASTER

Beschreibung

Mit der Anweisung "MB_MASTER" kann Ihr Programm über den Port eines Punkt-zu-Punkt-Moduls (CM) oder eines Kommunikationsboards (CB) als Modbus-Master kommunizieren. Sie können auf Daten in einem oder mehreren Modbus-Slave-Geräten zugreifen.

Damit die Anweisung "MB_MASTER" mit einem Port kommunizieren kann, muss "MB_COMM_LOAD (Seite 4254)" vorher ausgeführt werden.

Wenn Sie die Anweisung "MB_MASTER" in Ihr Programm einfügen, wird ein Instanz-DB erstellt. Diesen Instanz-DB geben Sie am Eingangsparameter MB_DB der Anweisung "MB_COMM_LOAD (Seite 4254)" an.

Regeln für die Kommunikation des Modbus-Masters

- Ein Port der für Modbus-Masteranforderungen verwendet wird, kann nicht für "MB_SLAVE" genutzt werden.
- Ein Port kann für einen oder mehrere Aufrufe von "MB_MASTER" genutzt werden, wenn der gleiche Instanz-DB verwendet wird.
- Die Modbus-Operationen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss die Anweisung "MB_MASTER" auf abgeschlossene Sende- und Empfangsvorgänge abfragen.
- Aufruf der Anweisung:
  - Rufen Sie die Anweisung "MB_MASTER" nach Möglichkeit in einem zyklischen Programm-OB auf. Die Anweisung kann auch in einem Verzögerungs- oder Weckalarm-OB aufgerufen werden.
  - Rufen Sie nicht mehrere "MB_MASTER"-Anweisungen in Organisationsbausteinen mit unterschiedlichen Prioritätsklassen auf. Bei einer vorweggenommenen Ausführung einer "MB_MASTER"-Anweisung aus einer höheren Prioritätsklasse kann es zu einer fehlerhaften Ausrührung kommen.
  - Rufen Sie "MB_MASTER"-Anweisungen nicht in einem Anlauf-, Diagnose- oder Zeitfehler-OB auf.
- Nach Beginn einer Übertragung muss am Parameter EN (KOP/FUP) durchgehend der Wert "1" anliegen, bis durch die Anweisung an der Ausgangsparameter DONE oder ERROR auf "1" gesetzt wird. Ein erneuter Aufruf durch den Parameter REQ während die Anweisung ausgeführt wird führt zu einem Fehler. Nach der Ausführung der Anweisung bleibt das Bit am Parameter REQ noch für den Zeitraum gesetzt, der im Instanz-DB an dem Parameter BLOCKED_PROC_TIMEOUT vorgegeben wurde.
- Wenn "MB_MASTER" eine Anforderung an einen Slave sendet, müssen Sie die Ausführung von "MB_MASTER" fortsetzen, bis die Antwort vom Slave eingegangen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_MASTER":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ (Seite 4260)	Input	BOOL	E, A, M, D, L	Anforderungseingabe: • 0 – Keine Anforderung • 1 – Anforderung, Daten an Modbus-Slave(s) zu senden
MB_ADDR	Input	UINT	E, A, M, D, L oder Konstante	Modbus-RTU-Stationsadresse: • Standard-Adressbereich: 0 bis 247 • Erweiterter Adressbereich: 0 bis 65535 Der Wert "0" ist für den Broadcast einer Meldung an alle Modbus-Slaves reserviert. Für den Broadcast werden nur die Modbus-Funktionscodes 05, 06, 15 und 16 unterstützt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE (Seite 4261)	Input	USINT	E, A, M, D, L oder Konstante	Auswahl Modus: Gibt die Art der Anforderung an: Lesen, Schreiben oder Diagnose In der Tabelle der Modbus-Funktionen finden Sie weitere Einzelheiten.
DATA_ADDR (Seite 4261)	Input	UDINT	E, A, M, D, L oder Konstante	Anfangsadresse im Slave: Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll. Die gültigen Adressen finden Sie in der Tabelle der Modbus-Funktionen.
DATA_LEN	Input	UINT	E, A, M, D, L oder Konstante	Datenlänge: Gibt die Anzahl der Bits oder Wörter an, auf die diese Anforderung zugreifen soll. Die gültigen Längen finden Sie in der Tabelle der Modbus-Funktionen.
DATA_PTR (Seite 4262)	In_Out	VARIANT	M, D	Zeiger auf die DB- oder Merker-Adresse der CPU für die zu schreibenden oder zu lesenden Daten. Bei einem DB muss dieser mit der Zugriffsart "Standard - kompatibel mit S7-300/400" erstellt werden.
DONE	Output	BOOL	E, A, M, D, L	• 0: Transaktion nicht abgeschlossen • 1: Transaktion ohne Fehler beendet
BUSY	Output	BOOL	E, A, M, D, L	• 0: Keine Transaktion von "MB_MASTER" in Bearbeitung • 1: Transaktion von "MB_MASTER" in Bearbeitung
ERROR	Output	BOOL	E, A, M, D, L	• 0: Kein Fehler • 1: Fehler; der Fehlercode wird am Parameter STATUS angezeigt
STATUS	Output	WORD	E, A, M, D, L	Ausführungsbedingung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Tabelle 4-183 Kommunikations- und Konfigurationsfehlermeldungen der Anweisung

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
80C8	Slave-Timeout. Überprüfen Sie die Baudrate, Parität und die Anschlüsse des Slaves.
80D1	Der Empfänger hat eine Flusskontrolle zur Unterbrechung einer laufenden Sendung angefordert, aber die Sendung innnerhalb der Wartezeit nicht wieder aktiviert. Dieser Fehler wird auch bei der Hardware-Flusskontrolle erzeugt, wenn der Empfänger innerhalb der Wartezeit kein CTS erkennt.
80D2	Die Sendeanforderung wurde abgebrochen, weil vom DCE kein DSR-Signal empfangen wird.
80E0	Die Meldung wurde beendet, weil der Empfangspuffer voll ist.
80E1	Die Meldung wurde wegen eines Paritätsfehlers beendet.
80E2	Die Meldung wurde wegen eines Telegrammfehlers beendet.
80E3	Die Meldung wurde wegen eines Überlauffehlers beendet.

Fehlercode* (W#16#....)	Beschreibung
80E4	Die Meldung wurde abgebrochen, weil die angegebene Länge die Gesamtpuffergröße überschreitet.
8180	Ungültiger Wert für die Port-ID.
8186	Ungültige Modbus-Stationsadresse
8188	Ungültiger Wert am Parameter MODE für einen Broadcast-Aufruf.
8189	Ungültiger Wert für die Datenadresse.
818A	Ungültiger Wert für die Datenlänge.
818B	Ungültiger Pointer auf die lokale Datenquelle bzw. das lokale Datenziel: Größe nicht richtig
818C	Ungültiger Zeiger am Parameter DATA_PTR. Verwenden Sie einen Zeiger auf einen Merkerbereich oder einen DB mit der Zugriffsart "Standard - kompatibel mit S7-300/400".
8200	Port ist durch Bearbeitung einer Sendeanforderung belegt
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Tabelle 4-184 Fehlermeldungen des Modbus-Protokolls

Fehlercode* (W#16#....)	Antwortcode des Slaves	Beschreibung
8380	-	CRC-Fehler
8381	01	Funktionscode nicht unterstützt
8382	03	Fehler in der Datenlänge
8383	02	Fehler in der Datenadresse oder Adresse außerhalb des gültigen Bereichs von DATA_PTR
8384	> 03	Fehler im Datenwert
8385	03	Wert des Datendiagnosecodes wird nicht unterstützt (Funktionscode 08)
8386	-	Funktionscode der Antwort stimmt nicht mit dem Funktionscode der Anfrage überein.
8387	-	Antwort des falschen Slaves
8388	-	Die Antwort des Slaves auf einen Schreibaufruf ist nicht korrekt. Die vom Slave gesendeten Daten stimmen nicht mit der Anfrage der Masers überein.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Parameter REQ

Beschreibung

- REQ = FALSE: Keine Anforderung
- REQ = TRUE: Anforderung, Daten an Modbus-Slave(s) zu senden

Sie können diesen Eingang über einen pegel- oder flankengesteuerten Kontakt steuern.

Immer wenn dieser Eingang aktiviert wird, wird eine Zustandsmaschine gestartet, um sicherzustellen, dass eine andere Anweisung "MB_MASTER", die denselben Instanz-DB verwendet, erst dann eine Anforderung abgeben kann, wenn die aktuelle Anforderung abgearbeitet ist. Alle anderen Eingangszustände werden erfasst und intern für die aktuelle Anforderung gespeichert, bis die Antwort empfangen oder ein Fehler erkannt wird.

Wird dieselbe Instanz von "MB_MASTER" erneut mit dem REQ-Eingang = 1 ausgeführt, bevor die aktuelle Anforderung abgearbeitet ist, werden keine nachfolgenden Übertragungen durchgeführt. Wenn die Anforderung jedoch abgearbeitet ist, wird zu dem Zeitpunkt, zu dem "MB_MASTER" erneut mit REQ-Eingang = 1 ausgeführt wird, eine neue Anforderung abgegeben.

Parameter DATA_ADDR und MODE

Beschreibung

Über den Parameter DATA_ADDR spezifizieren Sie die Anfangsadresse für den Datenzugriff auf den Modbus-Slave.

Über die Parameter MODE und die Modbus-Adresse legen Sie den Funktionscode fest, der an den Modbus-Slave übertragen wird. Die folgende Tabelle zeigt den Zusammenhang zwischen dem Parameter MODE, dem Funktioncode und dem Modbus-Adressbereich.

MODE	Modbus-Funktion	Datenlänge	Operation und Daten	Modbus-Adresse
0	01	1 bis 2000 1 bis 1992 (1)	Ausgangsbits lesen: 1 bis (1992 oder 2000) Bits pro Anfrage	1 bis 9999
0	02	1 bis 2000 1 bis 1992 (1)	Eingangsbits lesen: 1 bis (1992 oder 2000) Bits pro Anfrage	10001 bis 19999
0	03	1 bis 125 1 bis 124 (1)	Halteregister lesen: 1 bis (124 oder 125) WORD pro Anfrage	40001 bis 49999 oder 400001 bis 465535
0	04	1 bis 125 1 bis 124 (1)	Eingangs-WORD lesen: 1 bis (124 oder 125) WORD pro Anfrage	30001 bis 39999
1	05	1	Ein Ausgangsbit schreiben: Ein Bit pro Anfrage	1 bis 9999
1	06	1	Ein Halteregister schreiben: 1 WORD pro Anfrage	40001 bis 49999 oder 400001 bis 465535
1	15	2 bis 1968 2 bis 1960 (1)	Mehrere Ausgangsbits schreiben: 2 bis (1960 oder 1968) Bits pro Anfrage	1 bis 9999
1	16	2 bis 123 2 bis 122 (1)	Mehrere Halteregister schreiben: 2 bis (122 oder 123) WORD pro Anfrage	40001 bis 49999 oder 400001 bis 465535
2	15	1 bis 1968 2 bis 1960 (1)	Ein oder mehrere Ausgangsbits schreiben: 1 bis (1960 oder 1968) Bits pro Anfrage	1 bis 9999
2	16	1 bis 123 2 bis 122 (1)	Ein oder mehrere Halteregister schreiben: 1 bis (122 oder 123) WORD pro Anfrage	40001 bis 49999 oder 400001 bis 465535

MODE	Modbus-Funktion	Datenlänge	Operation und Daten	Modbus-Adresse
11	11	0	Auslesen des Kommunikations-Statusworts der Slaves und des Ereigniszählers: Das Statuswort zeigt die Ausführung der Anweisung an (0: wird nicht ausgeführt; 0xFFFF: wird ausgeführt). Der Ereigniszähler wird bei jeder erfolgreichen Übertragung einer Nachricht erhöht. Die Parameter DATA_ADDR und DATA_LEN der Anweisung "MB_MASTER" werden bei dieser Funktion ignoriert.	-
80	08	1	Überprüfen des Slave-Status durch Auslesen des Fehlercodes (0x0000): 1 WORD pro Anfrage	-
81	08	1	Ereigniszähler des Slaves zurücksetzen über den Diagnose-Code 0x000A: 1 WORD pro Anfrage	-
3 bis 10, 12 bis 79, 82 bis 2555			Reserviert	-
(1) Für den Modus "Erweiterter Adressbereich" wird die maximale Datenlänge um ein Byte bzw. ein WORD reduziert, je nachdem welcher Datentyp für die Funktion verwendet wird.				

Parameter DATA_PTR

Beschreibung

Der Parameter DATA_PTR ist ein Zeiger auf einen Datenbaustein oder einen Merker, von dem Daten geschrieben oder gelesen werden. Wenn Sie einen Datenbaustein verwenden, erstellen Sie einen globalen Datenbaustein mit Zugriffsart "Standard - kompatibel mit S7-300/400".

Datenbausteinstrukturen für den Parameter DATA_PTR

- Diese Datentypen gelten für das **Lesen von Wörtern** der Modbus-Adressen 30001 bis 39999, 40001 bis 49999 und 400001 bis 465536 sowie für das **Schreiben von Wörtern** in die Modbus-Adressen 40001 bis 49999 und 400001 bis 465536.
  - Standardfeld der Datentypen WORD, UINT oder INT (siehe unten).
  - Benannte Struktur vom Typ WORD, UINT oder INT, bei der jedes Element einen eindeutigen Namen und einen 16-Bit-Datentyp hat.
  - Benannte komplexe Struktur, bei der jedes Element einen eindeutigen Namen und einen 16- oder 32-Bit-Datentyp hat.
- Zum Lesen und Schreiben von Bits für die Modbus-Adressen 00001 bis 09999 und 10001 bis 19999.
  - Standardfeld aus Booleschen Datentypen.
  - Benannte Boolesche Struktur aus eindeutig benannten Booleschen Variablen.

- Es ist zwar nicht erforderlich, doch empfehlenswert, dass jede Anweisung "MB_MASTER" einen eigenen getrennten Speicherbereich in einem globalen Datenbaustein hat. Grund hierfür ist, dass die Möglichkeit der Datenzerstörung größer ist, wenn mehrere Anweisungen "MB_MASTER" in demselben Bereich eines globalen Datenbausteins lesen und schreiben.
- Es ist nicht erforderlich, dass die Speicherbereiche für DATA_PTR in demselben globalen Datenbaustein liegen. Sie können einen Datenbaustein mit mehreren Bereichen für Modbus-Lesevorgänge, einen Datenbaustein für Modbus-Schreibvorgänge oder einen Datenbaustein für jede Slave-Station anlegen.

Instanz-DB der Anweisung "MB_MASTER"

Statische Variablen des Instanz-DBs

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-DBs der Anweisung, die Sie im Anwenderprogramm verwenden können.

Variable	Datentyp	Beschreibung
MB_STATE	UINT	Interner Status der Modbus-Anweisung.
BLOCKED_ PROC_TIMEOUT	REAL	Zeitintervall nach Beendigung des Aufrufs der Anweisung und rücksetzen des Bits ACTIVE im Instanz-DB. Der Zeitpuffer wird verwendet, um zu vermeiden, dass das Ausführen der Anweisung beendet wird, bevor ein Auftrag vollständig gesendet wurde. Der voreingestellte Zeitintervall beträgt 500ms.
EXTENDED_ ADDRESSING	BOOL	Konfiguration der Adressierung: • 0: Standard-Adressbereich (1 Byte) • 1: Erweiterter Adressbereich (2 Byte) Weitere Informationen zur Variable EXTENDED_ADDRESSING finden Sie in dem folgenden Abschnitt: Instanz-DB der Anweisung "MB_SLAVE" (Seite 4269)

Beispielprogramm für einen Modbus-Master

Netzwerke (KOP)

Netzwerk 1: Parameter des RS-485-Moduls nur einmal während des ersten Zyklus initialisieren.

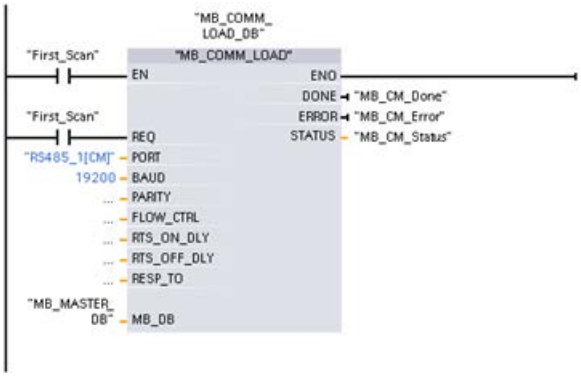

Netzwerk 2: 100 Wörter aus dem Halteregister des Slave lesen.

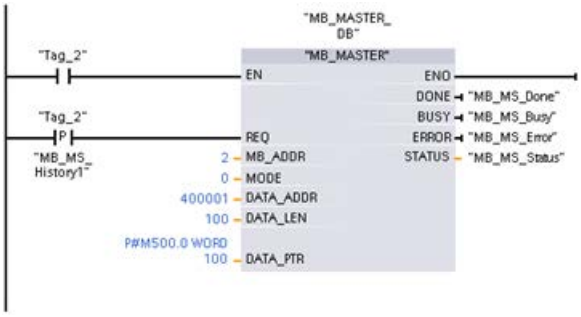

Netzwerk 3: Dies ist ein optionales Netzwerk, das die Werte der ersten 3 Wörter anzeigt, sobald die Leseoperation durchgeführt wurde.

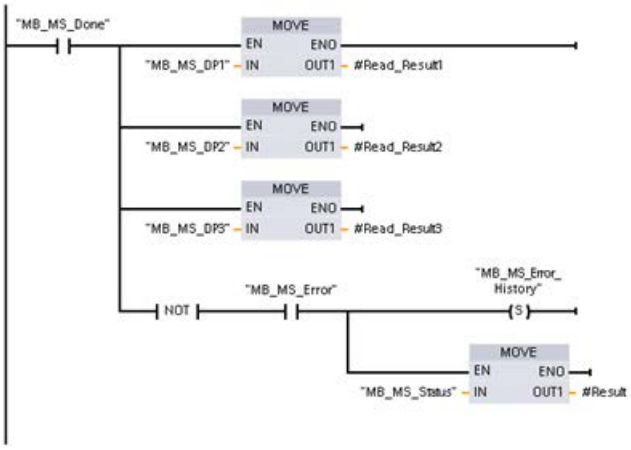

Netzwerk 4: 64 Bits in das Prozessabbild der Ausgänge schreiben, Beginn an Slave-Adresse A2.0.

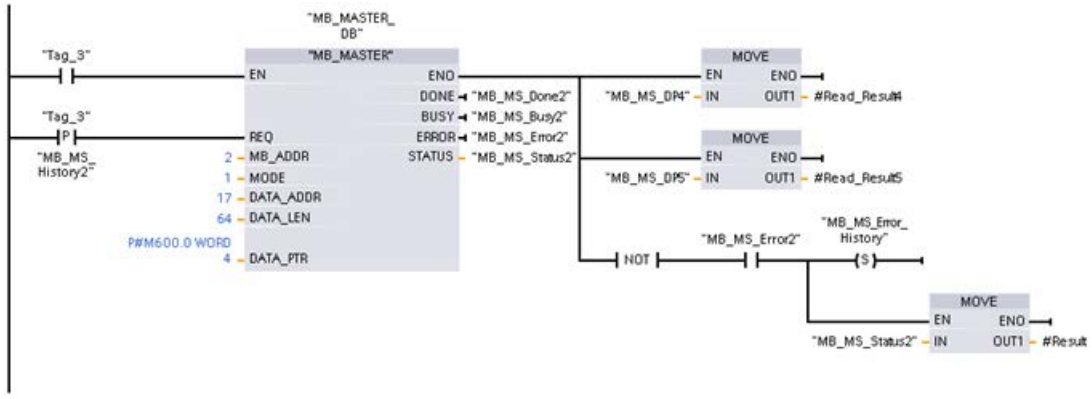

MB_SLAVE: Als Modbus-Slave über PtP-Port kommunizieren

Beschreibung MB_SLAVE

Beschreibung

Mit der Anweisung "MB_SLAVE" kann Ihr Programm über den Port eines Punkt-zu-Punkt-Moduls (PtP) oder eines Kommunikationsboards (CB) als Modbus-Slave kommunizieren. Ein Modbus-RTU-Master kann eine Anforderung senden, und Ihr Programm antwortet über die Ausführung von "MB_SLAVE".

Wenn Sie die Anweisung "MB_SLAVE" in Ihr Programm einfügen, müssen Sie einen eindeutigen Instanz-Datenbaustein zuweisen. Dieser Instanz-Datenbaustein wird verwendet, wenn Sie diesen an dem Parameter MB_DB der Anweisung "MB_COMM_LOAD (Seite 4254)" angeben.

Die Funktionscodes der Modbus-Kommunikation (Funktionscodes 1, 2, 4, 5 und 15) können Bits und Wörter direkt im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge im Zielsystem lesen und schreiben. Die folgende Tabelle zeigt die Zuordnung der Modbus-Adressen zum Prozessabbild in der CPU.

Modbus-Funktionen von "MB_SLAVE"						S7-1200	
Codes	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse
01	Bits lesen	Ausgang	1	bis	8192	Prozessabbild der Ausgänge	A0.0 bis A1023.7
02	Bits lesen	Eingang	10001	bis	18192	Prozessabbild der Eingänge	E0.0 bis E1023.7
04	Wörter lesen	Eingang	30001	bis	30512	Prozessabbild der Eingänge	EW0 bis EW1022
05	Bit schreiben	Ausgang	1	bis	8192	Prozessabbild der Ausgänge	A0.0 bis A1023.7
15	Bits schreiben	Ausgang	1	bis	8192	Prozessabbild der Ausgänge	A0.0 bis A1023.7

Die Funktionscodes der Modbus-Kommunikation (Funktionscodes 3, 6, 16) nutzen ein eigenes Halteregister. Hierzu kann ein Merker oder ein Datenbaustein mit der Zugriffsart "Standard - kompatibel mit S7-300/400" verwendet werden.

Den Typ des Halteregisters geben Sie durch den Parameter MB_HOLD_REG der Anweisung "MB_SLAVE" vor. Die folgende Tabelle zeigt die Zuordnung des Modbus-Halteregisters zur DB-Adresse von MB_HOLD_REG im Zielsystem.

Modbus-Funktionen von "MB_SLAVE"				S7-1200	
Codes	Funktion	Datenbereich	Adressbereich (WORD-Nummer)	Adresse im DB (BYTE-Nummer)	Merker-Adresse (BYTE-Nummer)
03	Wörter lesen	Halteregister	40001 bis 49999 oder	DW0 bis DW19998 oder	MW0 bis CPU-Limit
			400001 bis 465535	DW0 bis DW131068	
06	Wort schreiben	Halteregister	40001 bis 49999 oder	DW0 bis DW19998 oder	
			400001 bis 465535	DW0 bis DW131068	
16	Wörter schreiben	Halteregister	40001 bis 49999 oder	DW0 bis DW19998 oder	
			400001 bis 465535	DW0 bis DW131068	

Die folgende Tabelle zeigt die unterstützten Modbus-Diagnosefunktionen.

Modbus-Diagnosefunktionen von "MB_SLAVE" der S7-1200		
Codes	Teilfunktion	Beschreibung
08	0000H	Echotest Abfragedaten ausgeben: Die Anweisung "MB_SLAVE" gibt einem Modbus-Master das Echo eines empfangenen Datenworts zurück.
08	000AH	Kommunikationsereigniszähler löschen: Die Anweisung "MB_SLAVE" löscht den Kommunikationsereigniszähler, der für Modbus-Funktion 11 verwendet wird.
11	-	Kommunikationsereigniszähler abrufen: Die Anweisung "MB_SLAVE" nutzt einen internen Kommunikationsereigniszähler, um die Anzahl erfolgreicher Modbus-Lese- und Modbus-Schreibanforderungen, die an den Modbus-Slave gesendet werden, zu erfassen. Der Zähler wird bei Funktion 8, Funktion 11 und Broadcast-Anforderungen nicht hochgezählt. Außerdem wird bei Anforderungen, die zu Kommunikationsfehlern führen (z.B. Paritäts- oder CRC-Fehler), nicht hochgezählt.

Die Anweisung "MB_SLAVE" unterstützt Broadcast-Schreibanforderungen von Modbus-Mastern, solange die Anforderungen den Zugriff auf gültige Adressen umfassen.

Unabhängig von der Gültigkeit einer Anforderung gibt "MB_SLAVE" keine Antwort an einen Modbus-Master als Ergebnis einer Broadcast-Anforderung aus.

Regeln für die Modbus-Slave-Kommunikation

- "MB_COMM_LOAD" muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung "MB_SLAVE" mit diesem Port kommunizieren kann.
- Wenn ein Port als Slave auf einen Modbus-Master reagieren soll, kann dieser Port nicht von "MB_MASTER (Seite 4257)" verwendet werden. Nur eine Instanz von "MB_SLAVE" kann für einen bestimmten Port verwendet werden.

- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss den Kommunikationsprozess durch Abfrage der Anweisung "MB_SLAVE" auf abgeschlossene Sende- und Empfangsvorgänge steuern.
- Die Anweisung "MB_SLAVE" muss regelmäßig mit einer Häufigkeit ausgeführt werden, die eine zeitnahe Antwort auf eingehende Anforderungen eines Modbus-Masters ermöglicht. Es ist daher zu empfehlen, die Anweisung in einem zyklischen Programm-OB aufzurufen. Der Aufruf der Anweisung "MB_SLAVE" in einem Alarm-OB ist möglich, jedoch nicht zu empfehlen, da es zu hohen Zeitverzögerungen bei der Ausführung kommen kann.

Häufigkeit der Ausführung von "MB_SLAVE"

Die Anweisung "MB_SLAVE" muss regelmäßig ausgeführt werden, um jede Anforderung des Modbus-Masters zu empfangen und entsprechend zu antworten. Die Häufigkeit der Ausführung von "MB_SLAVE" richtet sich nach dem vom Modbus-Master vorgegebenen Timeout-Wert für die Antwort. Dies ist in der nachstehenden Abbildung dargestellt.

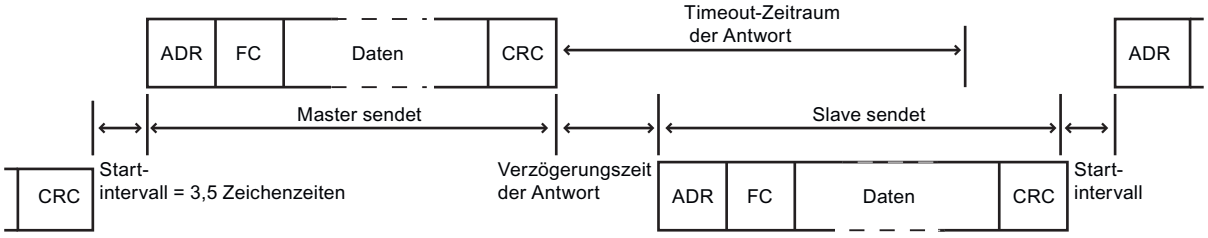

Der Timeout-Zeitraum für die Antwort ist die Zeitdauer, die ein Modbus-Master auf den Beginn einer Antwort von einem Modbus-Slave wartet. Dieser Zeitraum wird nicht vom Modbus-Protokoll definiert, sondern von einem Parameter des jeweiligen Modbus-Masters. Die Häufigkeit der Ausführung (Zeit zwischen einer Ausführung und der nächsten Ausführung) von "MB_SLAVE" muss sich nach den jeweiligen Parametern des Modbus-Masters richten. Sie sollten "MB_SLAVE" mindestens zweimal während des Timeout-Zeitraums für die Antwort des Modbus-Masters ausführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_SLAVE":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MB_ADDR	Input	V1.0: USINT V2.0: UINT	E, A, M, D, L oder Konstante	Stationsadresse des Modbus-Slaves (Adressbereich: 0 bis 255)
MB_HOLD_REG	In_Out	VARIANT	D	Zeiger auf den Modbus-Halteregister-DB. Der DB muss mit der Zugriffsart "Standard - kompatibel mit S7-300/400" erstellt werden.
NDR	Output	BOOL	E, A, M, D, L	Neue Daten bereit: • 0: Keine neuen Daten • 1: Gibt an, dass neue Daten vom Modbus-Master geschrieben wurden

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DR	Output	BOOL	E, A, M, D, L	Daten lesen: • 0: Keine Daten gelesen • 1: Gibt an, dass Daten vom Modbus-Master gelesen wurden
ERROR	Output	BOOL	E, A, M, D, L	• 0: Kein Fehler erkannt • 1: Fehler; ein entsprechender Fehlercode wird am Parameter STATUS ausgegeben
STATUS	Output	WORD	E, A, M, D, L	Fehlercode

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

STATUS* (W#16#....)	Beschreibung	
80C8	Das angegebene Antwort-Timeout (siehe RCVTIME oder MSGTIME) ist "0".	
80D1	Der Empfänger hat eine Flusskontrolle zur Unterbrechung einer laufenden Sendung angefordert, aber die Sendung innnerhalb der Wartezeit nicht wieder aktiviert. Dieser Fehler wird auch bei der Hardware-Flusskontrolle erzeugt, wenn der Empfänger innerhalb der Wartezeit kein CTS erkennt.	
80D2	Die Sendeanforderung wurde abgebrochen, weil vom DCE kein DSR-Signal empfangen wird.	
80E0	Die Meldung wurde beendet, weil der Empfangspuffer voll ist	
80E1	Die Meldung wurde wegen eines Paritätsfehlers abgebrochen	
80E2	Die Meldung wurde wegen eines Telegrammfehlers abgebrochen	
80E3	Die Meldung wurde wegen eines Überlauffehlers abgebrochen	
80E4	Die Meldung wurde abgebrochen, weil die angegebene Länge die Gesamtpuffergröße überschreitet	
8180	Ungültiger Wert für die Port-ID.	
8186	Ungültige Modbus-Stationsadresse	
8187	Ungültiger Pointer auf MB_HOLD_REG-DB	
818C	Pointer auf einen typsicheren DB-Typ MB_HOLD_REG (klassischer DB-Typ erforderlich)	
Antwortcode an Modbus-Master gesendet (B#16#...)		
8380	Keine Antwort	CRC-Fehler
8381	01	Funktionscode nicht unterstützt oder nicht unterstützt innerhalb eines Broadcasts
8382	03	Fehler in der Datenlänge
8383	02	Fehler in der Datenadresse oder Adresse außerhalb des gültigen Bereichs von MB_HOLD_REG
8384	03	Fehler im Datenwert
8385	03	Wert des Datendiagnosecodes wird nicht unterstützt (Funktionscode 08)
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Instanz-DB der Anweisung "MB_SLAVE"

Statische Variablen des Instanz-DBs

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-DBs der Anweisung, die Sie im Anwenderprogramm verwenden können. Ihr Programm kann in die Variablen HR_Start_Offset und Extended_Addressing Werte schreiben und die Modbus-Slave-Operationen steuern.

Die anderen Variablen können gelesen werden, um den Modbus-Zustand zu überwachen.

Variable	Datentyp	Beschreibung
HR_Start_Offset	WORD	Startadresse des Modbus-Halteregisters (Standard="0")
Extended_ Addressing	BOOL	Konfiguration der Adressierung: • 0: Standard-Adressbereich (1 Byte) • 1: Erweiterter Adressbereich (2 Byte)
Request_Count	WORD	Anzahl aller Anfragen, die durch den Slave empfangen wurden
Slave_Message_Count	WORD	Anzahl der Anfragen, die an diesen spezifischen Slave gesendet wurden
Bad_CRC_Count	WORD	Anzahl der empfangenen Anfragen mit CRC-Fehler
Broadcast_Count	WORD	Anzahl der empfangenen Broadcast-Anfragen
Exception_Count	WORD	Anzahl der Modbus-spezifischer Fehler, die das Rücksenden einer Ausnahme erfordern
Success_Count	WORD	Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen ohne Protokollfehler

HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 40001 oder 400001. Diese Adressen entsprechen der Anfangsadresse des Halteregisters im Zielsystemspeicher. Über die Variable HR_Start_Offset können Sie den Versatz zu einer anderen Anfangsadresse vorgeben.

Beispiel: Ein Halteregister beginnt an MW100 und hat eine Länge von 100 WORD. Bei einem Versatz von 20 am Parameter HR_Start_Offset beginnt das Halteregister an der Adresse 40021 statt 40001. Jede Adresse unter 40021 und über 400119 führt zu einem Adressierungsfehler.

	HR_Start_Offset = 0			HR_Start_Offset = 20	
	Modbus-Wortadresse	S7-1200 Byteadresse		Modbus-Wortadresse	S7-1200 Byteadresse
Minimum	40001	MW100		40021	MW100
Maximum	40099	MW198		40119	MW198

Extended_Addressing

Für die Modbus-Slave-Adressierung kann ein einzelnes Byte (Standard-Adressbereich) oder ein doppeltes Byte (Erweiterter Adressbereich) konfiguriert werden. Die erweiterte Adressierung dient zur Adressierung von mehr als 247 Geräten in einem einzigen Netzwerk. Wenn Sie sich für die erweiterte Adressierung entscheiden, können Sie maximal 64.000 Adressen ansprechen. Im Folgenden wird ein Telegramm der Modbus-Funktion 1 als Beispiel gezeigt.

Tabelle 4-185 Slave-Adresse mit einem Byte (Byte 0)

Funktion 1	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Anforderung	Slave-Adresse	F-Code	Anfangsadresse		Länge der Spulen		
Gültige Antwort	Slave-Adresse	F-Code	Länge	Spulendaten			
Fehlerantwort	Slave-Adresse	0x81	E-Code				

Tabelle 4-186 Slave-Adresse mit zwei Bytes (Byte 0 und Byte 1)

Funktion 1	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anforderung	Slave-Adresse		F-Code	Anfangsadresse		Länge der Spulen	
Gültige Antwort	Slave-Adresse		F-Code	Länge	Spulendaten		
Fehlerantwort	Slave-Adresse		0x81	F-Code			

Beispielprogramm für einen Modbus-Slave

Netzwerke (KOP)

Netzwerk 1: Parameter des RS-485-Moduls nur einmal während des ersten Zyklus initialisieren.

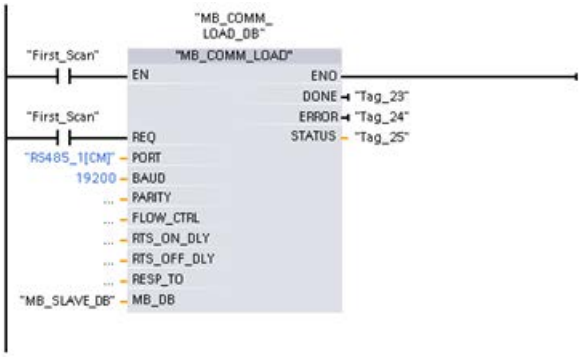

Netzwerk 2: In jedem Zyklus auf Anforderungen des Modbus-Master prüfen. Für das Modbus-Halteregister sind 100 Wörter mit Beginn an MW1000 konfiguriert.

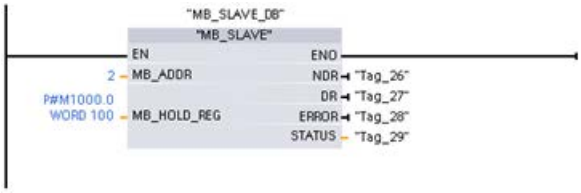

MODBUS (TCP)

MODBUS (TCP) für die Bibliotheks-Versionen V2.1 und V3.x der S7-1200-CPUs

MB_CLIENT: Als Modbus-TCP-Client über PROFINET kommunizieren

Beschreibung MB_CLIENT

Beschreibung

Die Anweisung "MB_CLIENT" kommuniziert als Modbus-TCP-Client über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötige Sie kein zusätzliches Hardwaremodul. Über die Anweisung "MB_CLIENT" bauen Sie eine Verbindung zwischen dem Client und dem Server auf, senden Anfragen und empfangen Antworten und steuern den Verbindungsabbau des Modbus-TCP-Servers.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_CLIENT":

Parameter	Deklaration	Datentyp	Beschreibung
REQ (Seite 4274)	Input	BOOL	Kommunikationsanfrage mit dem Modbus TCP-Server Der Paramter REQ ist pegelgesteuert. Das heißt, solange der Eingang gesetzt ist (REQ=true), sendet die Anweisung Kommunikationsanfragen. • Mit der Kommunikationsanfrage wird der Instanz-DB für andere Clients gesperrt. • Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. • Wird während einer laufenden Modbus-Anfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.
DISCONNECT (Seite 4274)	Input	BOOL	Über den Parameter steuern Sie den Verbindungsaufbau- und abbau zu dem Modbus-Server: • 0: Kommunikationsverbindung zu der angegebennen IP-Adresse und Port-Nummer herstellen. • 1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS der Wert 7003 ausgegeben. Ist der Parameter REQ beim Verbindungsaufbau gesetzt, wird die Abfrage sofort gesendet.
CONNECT_ID	Input	UINT	Eindeutige ID zur Identifikation der Verbindung. Jeder Instanz der Anweisungen "MB_CLIENT" und "MB_SERVER (Seite 4281)" muss eine eindeutige Verbindungs-ID zugewiesen werden.
IP_OCTET_1	Input	USINT	1. Oktett der IP-Adresse* des Modbus TCP -Servers.
IP_OCTET_2	Input	USINT	2. Oktett der IP-Adresse* des Modbus TCP -Servers.
IP_OCTET_3	Input	USINT	3. Oktett der IP-Adresse* des Modbus TCP -Servers.
IP_OCTET_4	Input	USINT	4. Oktett der IP-Adresse* des Modbus TCP -Servers.
IP_PORT	Input	UINT	IP-Port-Nummer des Servers, zu dem der Client die Verbindung herstellt und über das TCP/IP-Protokoll kommuniziert (Standard-Wert: 502).
MB_MODE (Seite 4275)	Input	USINT	Auswahl des Modus der Anfrage (Lesen, Schreiben oder Diagnose).
MB_DATA_ADDR (Seite 4275)	Input	UDINT	Anfangsadresse der Daten, auf welche die Anweisung "MB_CLIENT" zugreift.
DATA_LEN	Input	UINT	Datenlänge: Anzahl der Bits oder Wörter für den Datenzugriff (siehe "Parameter MB_MODE und MB_DATA_ADDR" - Datenlänge).
MB_DATA_PTR (Seite 4277)	InOut	VARIANT	Zeiger auf das Modbus-Datenregister: Das Register ist ein Puffer für die vom Modbus-Server empfangenen oder zum Modbus-Server zu sendenden Daten. Der Zeiger muss auf einen globalen Datenbaustein mit Standardzugriff verweisen. Die Anzahl der Bits, die adressiert werden, muss durch 8 teilbar sein.
DONE	Out	BOOL	Das Bit am Ausgangsparameter DONE wird auf "1" gesetzt, sobald der letzte Auftrag ohne Fehler ausgeführt wurde.

Parameter	Deklaration	Datentyp	Beschreibung
BUSY	Out	BOOL	• 0: Kein Auftrag von "MB_CLIENT " in Bearbeitung • 1: Auftrag von "MB_ CLIENT " in Bearbeitung
ERROR	Out	BOOL	• 0: Kein Fehler • 1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter STATUS angezeigt.
STATUS (Seite 4278)	Out	WORD	Fehlercode der Anweisung.
* 8 Bit langer Anteil der 32 Bit IPv4 IP-Adresse des Modbus TCP Servers.			

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Konsistente Input-Daten während eines "MB_CLIENT"-Aufrufs

Sobald ein Modbus-Client eine Modbus-Anweisung aufruft, wird der Status der Input-Parameter intern gespeichert und dann bei Folgeaufruf verglichen. Der Vergleich wird verwendet um zu bestimmen, ob dieser bestimmte Aufruf die laufende Abfrage initialisiert hat. Mehrere Aufrufe von "MB_CLIENT" können durchgeführt werden, indem Sie einen gemeinsamen Instanz-DB verwenden. Die Werte der Input-Parameter dürfen hierbei nicht verändert werden, solange eine "MB_CLIENT"-Instanz ausgeführt wird. Werden die Input-Parameter während der Ausführung verändert, kann durch "MB_CLIENT" nicht überprüft werden, ob die Instanz gerade ausgeführt wird.

Mehrfachverbindungen von Clients

Ein Modbus TCP-Client kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten. Modbus-TCP Verbindungen können hierbei auch gemeinsam von Client- und/oder Server-Verbindungen genutzt werden.

Bei einzelnen Client-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_CLIENT"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jede "MB_CLIENT"-Verbindung muss eine eindeutige IP-Adresse des Servers spezifiziert sein.
- Jede "MB_CLIENT"-Verbindung benötigt eine eindeutige Verbindungs-ID.
  Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Eindeutige Nummern des IP-Ports werden je nach Server-Konfiguration benötigt oder nicht.

Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_CLIENT".

Variable	Datentyp	Startwert	Beschreibung
Blocked_Proc_Time-out	REAL	3.0	Wartezeit in Sekunden, bevor bei einer blockierten Modbus-Instanz die statische Variable ACTIVE rückgesetzt wird. Dies kann zum Beispiel auftreten, wenn bei einer ausgegebenen Client-Anfrage die Ausführung der Client-Funktion abgebrochen wird, bevor die Anfrage vollständig ausgeführt wurde. Die maximale Wartezeit beträgt 55 Sekunden.
MB_Transaction_ID	WORD	1	Transaktions-ID des Modbus-TCP-Protokolls. Der Startwert von "1" sollte nur geändert werden, wenn der Modbus-TCP Server einen anderen Wert benötigt.
MB_Unit_ID	BYTE	255	Unit Identifier Modbus-Gerätekennung: Ein Modbus TCP-Server wird über seine IP-Adresse angesprochen. Deshalb wird der Parameter MB_UNIT_ID bei der Modbus TCP-Adressierung nicht verwendet. Der Parameter MB_UNIT_ID entspricht dem Feld der Slaveadresse beim Modbus RTU-Protokoll. Wenn ein Modbus TCP-Server als Gateway zu einem Modbus RTU-Protokoll verwendet wird, kann das Slavegerät im seriellen Netzwerk über MB_UNIT_ID identifiziert werden. Der Parameter MB_UNIT_ID würde in diesem Fall die Anforderung an die richtige Modbus RTU-Slaveadresse weiterleiten. Beachten Sie bitte, dass einige Modbus TCP-Geräte den Parameter MB_UNIT_ID möglicherweise für die Initialisierung innerhalb eines eingeschränkten Wertebereichs benötigen.
RCV_TIMEOUT	REAL	2.0	Zeitintervall in Sekunden, in der die Anweisung "MB_CLIENT" auf eine Antwort des Servers wartet.
Connected	BOOL	0	Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.

Siehe auch

MB_CLIENT Beispiel 1: Mehrere Anfragen über eine TCP-Verbindung senden (Seite 4287)

MB_CLIENT Beispiel 2: Mehrere Anfragen über mehrere TCP-Verbindungen senden (Seite 4288)

MB_CLIENT Beispiel 3: Mehrere Anfragen koordinieren (Seite 4289)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter REQ und DISCONNECT

Beschreibung

Wenn keine Instanz der Anweisung "MB_CLIENT" ausgeführt wird und wenn am Parameter DISCONNECT der Wert "0" anliegt, wird bei REQ=1 ein neuer Auftrag ausgeführt. Besteht noch keine Verbindung, wird diese bei der Ausführung hergestellt.

Wird die gleiche Instanz der Anweisung "MB_CLIENT" erneut ausgeführt (DISCONNECT=0 und REQ=1), bevor der aktive Auftrag ausgeführt wurde, wird dieser nicht im Anschluss an den aktiven Auftrag ausgeführt. Erst nach Beendigung des aktiven Auftrags kann ein neuer Auftrag angestoßen werden (REQ=1).

Den Status der Ausführung können Sie über den Ausgangsparameter DONE überwachen. Diesen können Sie nutzen, um bei einer sequenziellen Ausführung der Anweisung "MB_CLIENT" den Ausführungsstatus zu überwachen.

Siehe auch

Beschreibung MB_CLIENT (Seite 4271)

Parameter MB_MODE, MB_DATA_ADDR und DATA_LEN

Beschreibung

Für die Werte 0, 1 und 2 von MB_MODE legt die Kombination der Parameter MB_MODE, MB_DATA_ADDR und DATA_LEN den Modbus-Funktionscode fest, der in der aktuellen Modbus-Nachricht verwendet wird:

- MB_MODE enthält die Information, ob gelesen oder geschrieben werden soll. MB_MODE=0: Lesen, MB_MODE=1 und 2: Schreiben
- MB_DATA_ADDR enthält die Information, was gelesen bzw. geschrieben werden soll, sowie eine Adressinformation, aus der die Anweisung "MB_CLIENT" die remote Adresse berechnet.
- DATA_LEN enthält die Anzahl der zu lesenden / schreibenden Werte.

Beispiele:

- Die Kombination MB_MODE=1, MB_DATA_ADDR=1, DATA_LEN=1 legt den Funktionscode 05 fest. Ab der remoten Adresse 0 wird 1 Ausgangs-Bit geschrieben.
- Die Kombination MB_MODE=1, MB_DATA_ADDR=1, DATA_LEN=2 legt den Funktionscode 15 fest. Ab der remoten Adresse 0 werden 2 Ausgangs-Bits geschrieben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Eingangsparametern MB_MODE, MB_DATA_ADDR, DATA_LEN der Anweisung "MB_CLIENT" und der zugehörigen Modbus-Funktion.

MB_MODE	MB_DATA_ADDR	DATA_LEN	Modbus-Funktion	Funktion und Datentyp
0	1 bis 9.999	1 bis 2.000	01	1 bis 2.000 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	10.001 bis 19.999	1 bis 2.000	02	1 bis 2.000 Eingangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	40.001 bis 49.999	1 bis 125	03	1 bis 125 Halteregister auf der remoten Adresse 0 bis 9.998 lesen
0	30.001 bis 39.999	1 bis 125	04	1 bis 125 Eingangs-Worte auf der remoten Adresse 0 bis 9.998 lesen
1	1 bis 9.999	1	05	1 Ausgangs-Bit auf der remoten Adresse 0 bis 9.998 schreiben

MB_MODE	MB_DATA_ADDR	DATA_LEN	Modbus-Funktion	Funktion und Datentyp
1	40.001 bis 49.999	1	06	1 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben
1	1 bis 9.999	2 bis 1.968	15	2 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben
1	40.001 bis 49.999	2 bis 123	16	2 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben
2	1 bis 9.999	1 bis 1.968	15	1 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben
2	40.001 bis 49.999	1 bis 123	16	1 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben
11	Die Parameter MB_DATA_ADDR und DATA_LEN werden bei der Ausführung dieser Funktion nicht ausgewertet.		11	Status-Wort und Ereigniszähler des Servers lesen: • Das Status-Wort bildet den Bearbeitungszustand ab (0 - nicht in Bearbeitung, 0xFFFF - in Bearbeitung). • Der Ereigniszähler wird inkrementiert, wenn die Modbus-Anfrage erfolgreich ausgeführt wurde. Wenn bei der Ausführung einer Modbus-Funktion ein Fehler aufgetreten ist, wird zwar eine Nachricht vom Server gesendet, der Ereigniszähler jedoch nicht inkrementiert.
80	-	1	08	Überprüfung des Server-Status durch den Diagnosecode 0x0000 (Rückschleifentest - der Server sendet die Abfrage zurück): • 1 WORD pro Aufruf
81	-	1	08	Ereigniszähler des Servers über den Diagnosecode 0x000A rücksetzen: • 1 WORD pro Aufruf
3 bis 10, 12 bis 79, 82 bis 255				reserviert

Hinweis

Unit Identifier für RTU-Geräte über MB-TCP

Die Gerätekennung (Unit Identifier) für RTU-Geräte über Modbus-TCP finden Sie als Variable "MB_Unit_ID" in den editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_CLIENT". Der Unit Identifier dient der eindeutigen Zuordnung der Koppelpartner.

Siehe auch

Beschreibung MB_CLIENT (Seite 4271)

Parameter MB_DATA_PTR

Beschreibung

Der Parameter MB_DATA_PTR ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie für einen Puffer im Speicherbereich (M) einen Zeiger im Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Der Parameter MB_DATA_PTR verwendet einen Kommunikationspuffer:

- Bei den Kommunikationsfunktionen der Anweisung "MB_CLIENT":
  - Lesen und schreiben von 1-Bit Daten der Modbus-Server Adressen 00001 bis 09999 und 10001 bis 19999.
  - Lesen von 16-Bit WORD-Daten der Modbus-Server Adresssen 30001 bis 39999 und 40001 bis 49999.
  - Schreiben von 16-Bit WORD-Daten der Modbus-Server Adresssen 40001 bis 49999.
- Bei der Datenübertragung (Länge: Bit oder WORD) von oder zu dem globalen DB oder dem Speicherbereich (M), den Sie über den Parameter MB_DATA_PTR zugewiesen haben.

Wenn Sie für den Zeiger auf den Puffer am Parameter MB_DATA_PTR einen Datenbaustein verwenden, müssen Sie den DB-Elementen Datentypen zuweisen.

- Verwenden Sie den 1-Bit Datentyp BOOL für eine Modbus Bit-Adresse
- Verwenden Sie einen 16-Bit Datentyp wie WORD, UINT, INT oder REAL für eine Modbus WORD-Adresse.
- Verwenden Sie einen 32-Bit Datentyp (Doppelwort) wie DWORD, DINT oder REAL für zwei Modbus WORD-Adressen.
- Über MB_DATA_PTR können Sie auch auf komplexe DB-Elemente zugreifen wie:
  - Standard-Arrays
  - Strukturen mit eindeutigen Element-Namen
  - Komplexe Strukturen mit eindeutiger Benennung der Elemente Datentyplängen von 16 oder 32 Bit.
- Die Datenbereiche für den Parameter MB_DATA_PTR können auch unterschiedlichen globalen Datenbausteinen stehen (oder in unterschiedlichen Speicherbereichen). Sie können zum Beispiel einen Datenbaustein für die Lesevorgänge und einen anderen für die Schreibvorgänge verwenden oder für jede "MB_CLIENT"-Station einen eigenen Datenbaustein.

Siehe auch

Beschreibung MB_CLIENT (Seite 4271)

Parameter STATUS

Parameter STATUS (allgemeine Statusinformationen) (Bibliotheks-Version V2.1)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
7000	Kein Aufruf aktiv (REQ=0).
7001	Erstaufruf mit REQ=1: Bearbeitung angestoßen; BUSY hat den Wert 1.
7002	Zwischenaufruf (REQ irrelevant). Bearbeitung bereits aktiv; BUSY hat den Wert 1.
7003	Verbindung wird abgebaut.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (allgemeine Statusinformationen) (Bibliotheks-Version V3.x)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
7000	Kein Aufruf aktiv (REQ=0).
7001	Erstaufruf mit REQ=1: Bearbeitung angestoßen; BUSY hat den Wert 1.
7002	Zwischenaufruf (REQ irrelevant). Bearbeitung bereits aktiv; BUSY hat den Wert 1.
7003	Verbindung wird abgebaut.
7004	Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.
7005	Daten wurden gesendet.
7006	Daten wurden empfangen.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (Protokoll-Fehler)

STATUS* (W#16#)	Lokaler bzw. remoter Fehler	Fehlercode in der Antwort von MB_SERVER (B#16#)	Beschreibung
80C8	lokal	-	Keine Antwort des Servers im definierten Zeitraum. Überprüfen Sie die Verbindung zu dem Modbus-Server. Dieser Fehler wird erst nach Abschluss der konfigurierten erneuten Versuche gemeldet. Empfängt die Anweisung "MB_CLIENT" keine Antwort mit der ursprünglich übermittelten Transaktions-ID (Variable MB_TRANSACTION_ID) innerhalb des definierten Zeitraums, wird dieser Fehlercode ausgegeben.
8380	lokal	-	Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8381	remote	01	Funktionscode wird nicht unterstützt.

STA-TUS* (W#16#)	Lokaler bzw. remoter Fehler	Fehlercode in der Antwort von MB_SERVER (B#16#)	Beschreibung
8382	lokal	-	• Die Länge des Modbus-Frames im Frame-Header stimmt mit der Anzahl der empfangenen Bytes nicht überein. • Die Anzahl der Bytes stimmt mit den tatsächlich übertragenen Bytes nicht überein (nur Funktionen 1-4). Dies ist z. B. dann der Fall, wenn "MB_CLIENT" eine ungerade Anzahl von Worten anfordert, "MB_SERVER" aber immer eine gerade Anzahl von Worten sendet. • Die Startadresse im empfangenen Frame stimmt mit der gespeicherten Startadresse nicht überein (Funktionen 5, 6, 15, 16). • Anzahl der Wörter stimmt mit den tatsächlich übertragenen Wörtern nicht überein (Funktionen 15 und 16).
	remote	03	Unzulässige Längenangabe im empfangenen Modbus-Frame. Überprüfen Sie die Server-Seite.
8383	lokal	-	Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs von MB_DATA_PTR (Seite 4277).
	remote	02	Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs des Servers
8384	lokal	-	• Ungültiger Exception Code emfangen. • Es wurde ein anderer Datenwert empfangen, als ursprünglich vom Client gesendet wurde (Funktionen 5, 6 und 8). • Ungültiger Status-Wert empfangen (Funktion 11)
	remote	03	Fehler bei Datenwert für die Funktion 5
8385	lokal	-	• Diagnosecode nicht unterstützt. • Es wurde ein anderer Subfunktionscode empfangen, als ursprünglich vom Client gesendet wurde (Funktion 8).
	remote	03	Diagnosecode nicht unterstützt
8386	lokal	-	Empfangener Funktions-Code entspricht nicht dem ursprüglich gesendetem.
8387	lokal	-	• Die zugewiesene Verbindungs-ID ist eine andere als bei vorherigen Abfragen. Es kann nur eine Verbindungs-ID für jeden Instanz-DB der Anweisung "MB_CLIENT" verwendet werden. • Der Fehlercode wird auch ausgegeben, wenn die ID des vom Server empfangenen Modbus TCP Protokolls ungleich "0" ist.
8388	lokal	-	Durch den Modbus-Server wurde eine andere Datenlänge gesendet, als abgefragt wurde. Dieser Fehler tritt nur bei Verwendung der Modbus-Funktionen 15 oder 16 auf.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".			

Parameter STATUS (Parameter-Fehler)

Zusätzlich zu den Fehlern, die in der folgenden Tabelle aufgelistet sind, können bei der Anweisung "MB_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

STATUS* (W#16#)	Beschreibung
80BB	Ungültiger Wert am Parameter ACTIVE_EST (Kennung für die Art des Verbindungsaufbaus, siehe T_CON_PARAM): • Serverseitig ist nur der passive Verbindungsaufbau zulässig (ACTIVE_EST = FALSE). • Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ACTIVE_EST = TRUE).
8188	Ungültiger Wert am Parameter MB_MODE.
8189	Ungültige Adressierung der Daten am Parameter MB_DATA_ADDR.
818A	Ungültige Datenlänge am Parameter MB_DATA_LEN.
818B	Ungültiger Zeiger am Parameter MB_DATA_PTR. Überprüfen Sie zudem die Werte der Parameter MB_DATA_ADDR (Seite 4275) und MB_DATA_LEN.
818C	• Der Zeiger am Parameter MB_DATA_PTR (Seite 4271) verweist auf einen optimierten Datenbaustein. Verwenden Sie entweder einen Datenbaustein mit Standardzugriff oder einen Speicherbereich. • Zeitüberschreitung bei dem Parameter BLOCKED_PROC_TIMEOUT (siehe statische Variablen der Anweisung). Das Limit von 55 Sekunden wurde überschritten.
8200	• Eine weitere Modbus-Anfrage wird zur Zeit über den Port verarbeitet. • Eine weitere Instanz von MB_CLIENT mit den gleichen Verbindungsparametern bearbeitet eine bereits bestehende Modbus-Anfrage.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Fehlercodes der intern verwendeten Kommunikationsanweisungen.

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_CLIENT" möglich. Im Abschnitt Static werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Siehe auch

Parameter MB_HOLD_REG (Seite 4284)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

MB_SERVER: Als Modbus-TCP-Server über PROFINET kommunizieren

Beschreibung MB_SERVER

Beschreibung

Die Anweisung "MB_SERVER" kommuniziert als Modbus-TCP-Server über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötige Sie kein zusätzliches Hardwaremodul. Die Anweisung "MB_SERVER" verarbeitet Verbindungsanfragen eins Modbus-TCP-Clients, empfängt Anfragen von Modbus-Funktionen und sendet Antwort-Meldungen.

ACHTUNG
Security-Hinweis Beachten Sie, dass jeder Client des Netzwerks lesenden und schreibenden Zugriff erhält auf das Prozessabbild der Ein- und Ausgänge und auf den durch das Modbus-Halteregister definierten Datenbaustein oder Merkerbereich. Zum Schutz vor unbefugten Lese- und Schreibvorgänge besteht die Möglichkeit, den Zugriff auf eine IP-Adresse einzuschränken. Beachten Sie jedoch, dass sich die freigegebene Adresse auch für einen unbefugten Zugriff nutzen lässt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_SERVER":

Parameter	Deklaration	Datentyp	Beschreibung
DISCONNECT	Input	BOOL	Die Anweisung "MB_SERVER" geht eine passive Verbindung mit einer Partner-Baugruppe ein, d.h. der Server reagiert auf eine TCP-Verbindungsanfrage von jeder anfragenden IP-Adresse. Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird: • 0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufbebaut. • 1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine anderen Vorgänge ausgeführt. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS der Wert 7003 ausgegeben.
CONNECT_ID	Input	UINT	Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisungen "MB_CLIENT (Seite 4271)" und "MB_SERVER" muss eine eindeutige ID am Parameter CONNECT_ID verwenden.
IP_PORT	Input	UINT	Startwert=502. Die Nummer des IP-Ports legt fest, welcher IP-Port für Verbindungsanfragen des Modbus-Clients überwacht wird. Die folgenden TCP-Port-Nummern dürfen nicht für die passive Verbindung der "MB_SERVER"-Anweisung verwendet werden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

Parameter	Deklaration	Datentyp	Beschreibung
MB_HOLD_REG (Seite 4284)	InOut	VARIANT	Zeiger auf das Modbus-Halteregister der "MB_SERVER"-Anweisung: Verwenden Sie als Halteregister einen globalen Datenbaustein mit Standardzugriff. Das Halteregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (lesen), 6 (schreiben) und 16 (lesen) zugreifen darf.
NDR	Output	BOOL	"New Data Ready": • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client geschrieben
DR	Output	BOOL	"Data Read": • 0: Keine Daten gelesen • 1: Daten durch den Modbus-Client gelesen
ERROR	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_SERVER" ein Fehler auf, wird der Ausgang am Parameter ERROR auf TRUE gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS angezeigt.
STATUS (Seite 4285)	Output	WORD	Fehlercode der Anweisung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Abbilden der Modbus-Adressen auf dem Prozessabbild

Die Anweisung "MB_SERVER" ermöglicht eingehenden Modbus-Funktionen (1, 2, 4, 5 und 15) direkt lesend und schreibend auf das Prozessabbild der Eingänge und Ausgänge der S7-1200 CPU zuzugreifen (Verwendung der Datentypen BOOL und WORD).

Für den Datentransfer der Funktionscodes 3, 6 und 16 muss das Halteregister (Parameter MB_HOLD_REG) länger als ein Byte definiert werden. Die folgende Tabelle zeigt die Abbildung der Modbus-Adressen auf das Prozessabbild der CPU.

Modbus Funktion						S7-1200	
Code	Funktion	Datenbereich	Adressraum			Datenbereich	CPU Adresse
01	Lesen: Bits	Output	0	bis	8191	Prozessabbild Ausgang	A0.0 bis A1023.7
02	Lesen: Bits	Input	0	bis	8191	Prozessabbild Eingang	E0.0 bis E1023.7
04	Lesen: WORD	Input	0	bis	1021	Prozessabbild Eingang	EW0 bis EW1022
05	Schreiben: Bits	Output	0	bis	8191	Prozessabbild Ausgang	A0.0 bis A1023.7
15	Schreiben: Bits	Output	0	bis	8191	Prozessabbild Ausgang	A0.0 bis A1023.7

Eingehende Modbus-Meldungen mit den Funktionscodes 3, 6 und 16 schreiben oder lesen vom Modbus-Halteregister (das Halteregister spezifizieren Sie am Parameter MB_HOLD_REG).

Mehrfache Server-Verbindungen

Sie können mehrfache Server-Verbindungen erstellen. Dies erlaubt es einer einzelnen CPU gleichzeitig Verbindungen zu mehreren Modbus-TCP-Clients herzustellen.

Ein Modbus-TCP-Server kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt.

Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Modbus-TCP Verbindungen können hierbei auch gemeinsam von Client- und/oder Server-Verbindungen genutzt werden.

Bei Server-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_SERVER"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Jede "MB_SERVER"-Verbindung muss mit einer eindeutigen Nummer des IP-Ports erstellt werden. Für jeden Port wird nur eine Verbindung unterstützt.
- Jede "MB_SERVER"-Verbindung muss eine eindeutige Verbindungs-ID verwenden. Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Für jede Verbindung muss die Anweisung "MB_SERVER" einzeln aufgerufen werden.

Modbus-Diagnosefunktionen

Die folgende Tabelle enthält die Beschreibung der Modbus-Diagnosefunktionen.

Code	Sub-Funktion	Beschreibung
08	0x0000	Echo-Test: Die Anweisung "MB_SERVER" empfängt ein Datenwort und sendet dieses unverändert an den Modbus-Master zurück.
08	0x000A	Ereigniszähler zurücksetzen: Die Anweisung "MB_SERVER" setzt den Ereigniszähler für die Kommunikation zurück, welcher für die Modbus-Funktion 11 verwendet wird.
11	-	Ereigniszähler der Kommunikation holen: Die Anweisung "MB_SERVER" verwendet einen internen Ereigniszähler bei der Kommunikation zur Erfassung der Anzahl der erfolgreich ausgeführten Lese- und Schreibanfragen, die an den Modbus-Server gesendet werden. Der Ereigniszähler wird bei den Funktionen 8, 11 oder Broadcast-Anfragen nicht inkrementiert. Gleiches gilt für Anfragen, die einen Kommunikationsfehler zur Folge haben (zum Beispiel bei Paritätsfehlern oder CRC-Fehlern). Die Broadcast-Funktion ist für Modbus-TCP nicht verfügbar, da nur eine Client/Server-Verbindung gleichzeitig bestehen kann.

Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_SERVER", die im Programm verwendet werden können. Auf die Variable HR_Start_Offset können Sie schreibend zugreifen. Auf die übrigen Variablen können Sie lesend zugreifen, um den Modbus-Status zu überwachen.

Variable	Datentyp	Startwert	Beschreibung
HR_Start_Offset	WORD	0	Zuweisen der Anfangsadresse des Modbus-Halteregisters.
Request_Count	WORD	0	Anzahl aller Anfragen, die durch den Server empfangen wurden.
Server_Message_Count	WORD	0	Anzahl aller empfangenen Meldungen für den jeweiligen Server.

Variable	Datentyp	Startwert	Beschreibung
Xmt_Rcv_Count	WORD	0	Zähler zur Erfassung der Anzahl der Übertragungen, bei denen ein Fehler aufgetreten ist. Der Zähler wird auch erhöht, wenn eine ungültige Modbus-Meldung empfangen wurde.
Exception_Count	WORD	0	Zähler zur Erfassung der Anzahl von Fehlern, die spezifisch für Modbus sind und ein Ausnahmefehler hervorrufen.
Success_Count	WORD	0	Zähler zur Erfassung der Anzahl der Anfragen, die keine Fehler in dem übertragenen Protokoll enthalten.
Connected	BOOL	0	Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.

Beispiel: Adressierung über statische Variable HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 0 (aus Sicht des MB_CLIENT bei 40.001). Diese Adressen entsprechen dem Adressraum des CPU-Speicherbereichs für das Halteregister. Sie können die Variable HR_Start_Offset auch so definieren, dass das Modbus-Halteregister eine andere Anfangsadresse als 0 erhält.

Beispiel: Das Halteregister beginnt am MW100 und hat eine Länge von 100 WORD. Ein Offset-Wert am Parameter HR_Start_Offset bewirkt, dass sich die Anfangsadresse des Halteregisters von 0 auf 20 verschiebt. Jede Adressierung des Halteregisters, die unterhalb der Adresse 20 und oberhalb der Adresse 119 liegt, ruft einen Fehler hervor.

HR_Start_Offset	Adresse	Minimum	Maximum
0	Modbus-Adresse (WORD)	0	99
	CPU-Adresse	MW100	MW298
20	Modbus-Adresse (WORD)	20	119
	CPU-Adresse	MW100	MW298

Siehe auch

MB_SERVER Beispiel: Mehrere TCP-Verbindungen (Seite 4290)

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter MB_HOLD_REG

Beschreibung

Der Parameter MB_HOLD_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

- Die Obergrenze für die Anzahl von Adressen im Datenbaustein (D) ist durch den maximalen Arbeitsspeicher der CPU vorgegeben.
- Die Obergrenze für die Anzahl von Merkern (M) ist durch die Größe des Speicherbereichs der CPU vorgegeben.

Die folgende Tabelle zeigt einige Beispiele zur Abbildung von Modbus-Adressen auf das Halteregister für die Modbus-Funktionen 3 (WORD lesen), 6 (WORD schreiben), 16 (mehrere WORD schreiben) und 23 (mehrere Wörter schreiben und lesen).

Modbus-Adressen	MB_HOLD_REG Parameter - Beispiele		
0	MW100	DB10.DBW0	"Recipe".ingredient[1]
1	MW102	DB10.DBW2	"Recipe".ingredient[2]
2	MW104	DB10.DBW4	"Recipe".ingredient[3]
3	MW106	DB10.DBW6	"Recipe".ingredient[4]
4	MW108	DB10.DBW8	"Recipe".ingredient[5]

Siehe auch

Beschreibung MB_SERVER (Seite 4281)

Parameter STATUS

Parameter STATUS (allgemeine Statusinformationen) (Bibliotheks-Version V2.1)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
7000	Kein Aufruf aktiv (REQ=0).
7001	Erstaufruf mit REQ=1: Bearbeitung angestoßen; BUSY hat den Wert 1.
7002	Zwischenaufruf (REQ irrelevant). Bearbeitung bereits aktiv; BUSY hat den Wert 1.
7003	Verbindung wird abgebaut.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (allgemeine Statusinformationen) (Bibliotheks-Version V3.x)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
7000	Kein Aufruf aktiv (REQ=0).
7001	Erstaufruf. Verbindungsaufbau angestoßen.
7002	Zwischenaufruf. Verbindung wird aufgebaut.
7003	Verbindung wird abgebaut.
7005	Daten werden gesendet.

STATUS* (W#16#)	Beschreibung
7006	Daten werden empfangen.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (Protokoll-Fehler)

STATUS* (W#16#)	Fehlercode in der Antwort von MB_SERVER (B#16#)	Beschreibung
8380	-	Empfangener Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8381	01	Funktionscode wird nicht unterstützt.
8382	03	Fehler bei Datenlänge
8383	02	Fehler bei der Datenadresse oder Zugriff außerhalb des Adressbereichs von MB_HOLD_REG (Seite 4284).
8384	03	Fehler beim Datenwert (Funktion 5)
8385	03	Diagnosecode nicht unterstützt
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Parameter STATUS (Parameter-Fehler)

STATUS* (W#16#)	Beschreibung
80BB	Ungültiger Wert am Parameter ACTIVE_EST (Kennung für die Art des Verbindungsaufbaus, siehe T_CON_PARAM): • Serverseitig ist nur der passive Verbindungsaufbau zulässig (ACTIVE_EST = FALSE). • Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ACTIVE_EST = TRUE).
8187	Ungültiger Zeiger am Parameter MB_HOLD_REG. Datenbereich ist zu klein.
818C	• Der Zeiger am Parameter MB_HOLD_REG verweist auf einen optimierten Datenbaustein. Verwenden Sie entweder einen Datenbaustein mit Standardzugriff oder einen Speicherbereich. • Fehler durch Zeitüberschreitung der Ausführung (bei mehr als 55 Sekunden).
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Fehlercodes der intern verwendeten Kommunikationsanweisungen.

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_SERVER" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_SERVER" möglich. Im Abschnitt Static werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Siehe auch

Beschreibung MB_SERVER (Seite 4281)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Beispiele

MB_CLIENT Beispiel 1: Mehrere Anfragen über eine TCP-Verbindung senden

Beschreibung

Über eine TCP-Verbindung können mehrere Anfragen des Modbus-Client gesendet werden. Hierzu verwenden Sie den gleichen Instanz-DB, die gleiche Verbindungs-ID und die gleiche Port-Nummer.

Zur gleichen Zeit kann immer nur ein Client aktiv sein. Nachdem die Ausführung eines Clients beendet wurde, wird der nächste Client ausgeführt. Die Reihenfolge der Ausführung müssen sie im Programm definieren.

In dem folgenden Beispielprogramm wird als Ergänzung der Wert des Ausgangsparameters STATUS kopiert.

Netzwerk 1: Modbus Funktion 1 - 16 Output-Bits lesen

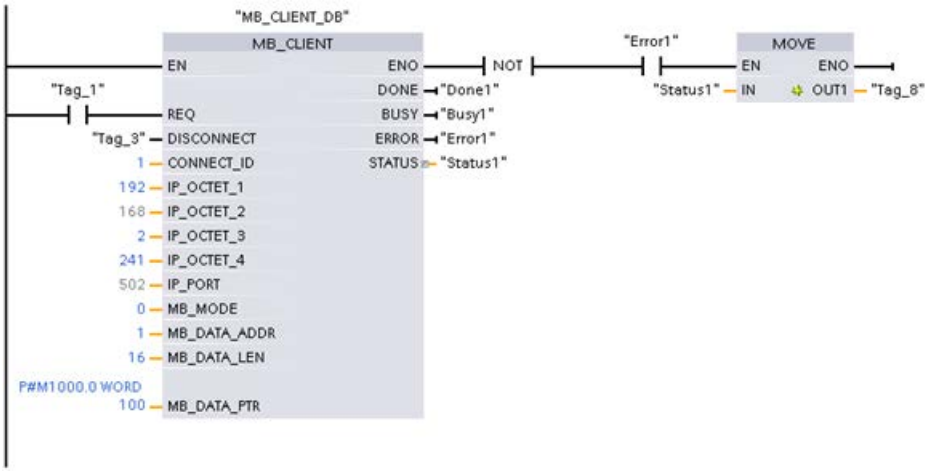

Netzwerk 2: Modbus Funktion 2 - 32 Input-Bits lesen

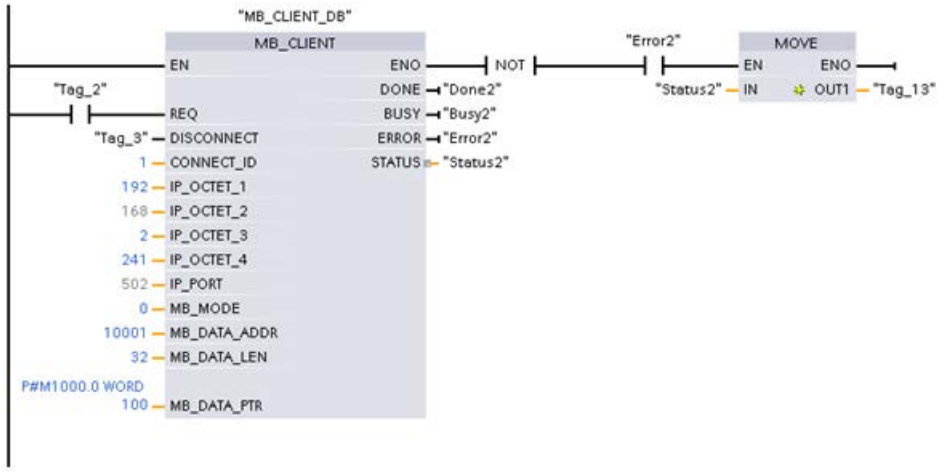

MB_CLIENT Beispiel 2: Mehrere Anfragen über mehrere TCP-Verbindungen senden

Beschreibung

Anfragen des Modbus-Client können über verschiedene TCP-Verbindungen gesendet werden. Hierzu verwenden Sie jeweils einen unterschiedlichen Instanz-DB und eine andere Verbindungs-ID.

Verwenden Sie jeweils eine unterschiedliche Port-Nummer, wenn die Verbindungen zu dem gleichen Modbus-Server erfolgen. Erfolgen die Verbindungen zu jeweils unterschiedlichen Modbus-Servern, können Sie die Port-Nummer frei vergeben.

Netzwerk 1: Modbus-Funktion 4 - Input (WORD) lesen

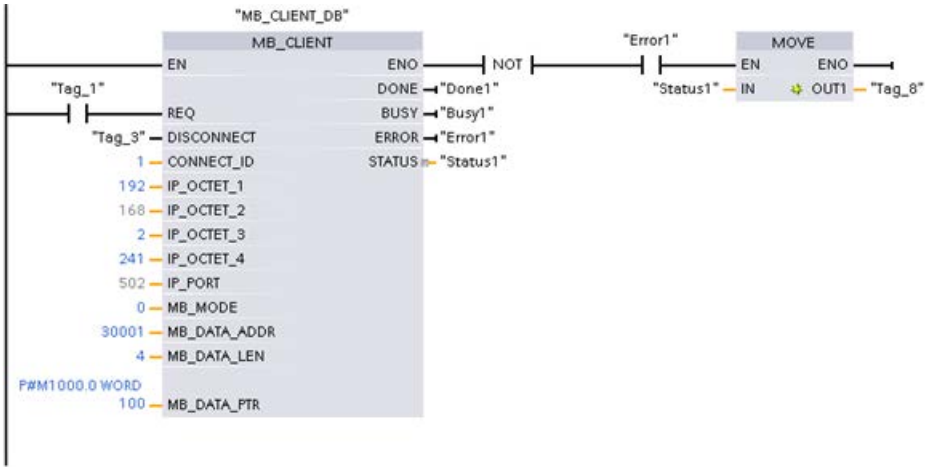

Netzwerk 2: Modbus-Funktion 3 - Halteregister (WORD) lesen

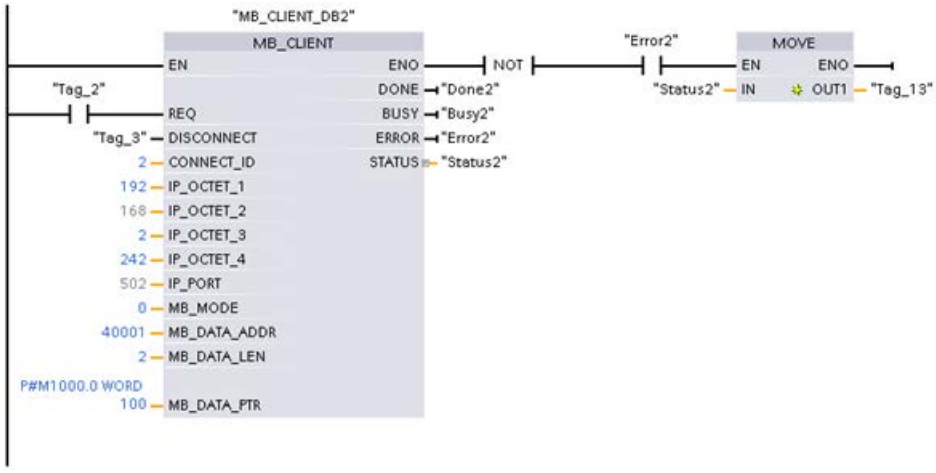

MB_CLIENT Beispiel 3: Mehrere Anfragen koordinieren

Beschreibung

Sie müssen sicherstellen, dass die einzelnen Modbus- Anfragen ausgeführt werden. Die Koordination der Anfragen steuern Sie über das Programm. Das folgende Beispiel zeigt, wie die Output-Parameter der ersten und zweiten Client-Anfrage verwendet werden können, um die Ausführung der Anweisungen zu koordinieren.

Netzwerk 1: Modbus-Funktion 3 - Halteregister lesen (WORD)

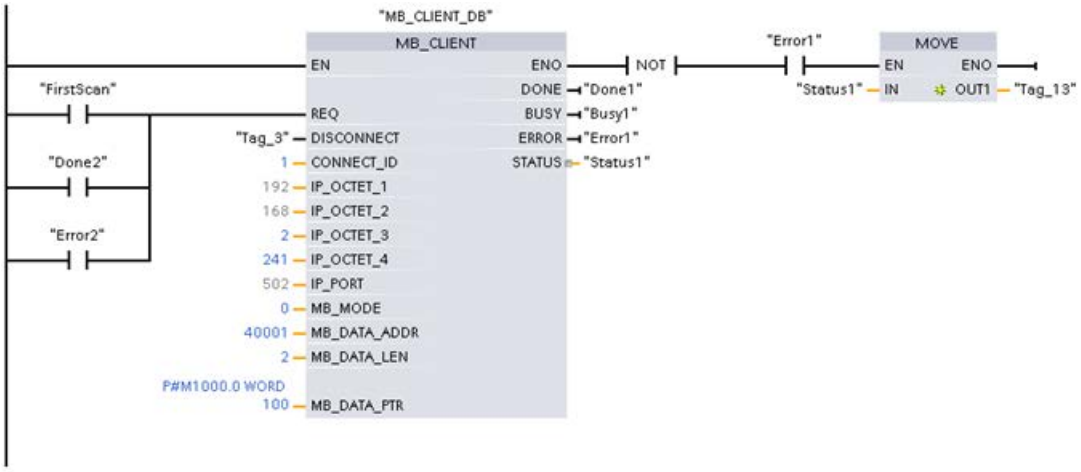

Netzwerk 2: Modbus-Funktion 3 - Halteregister lesen (WORD)

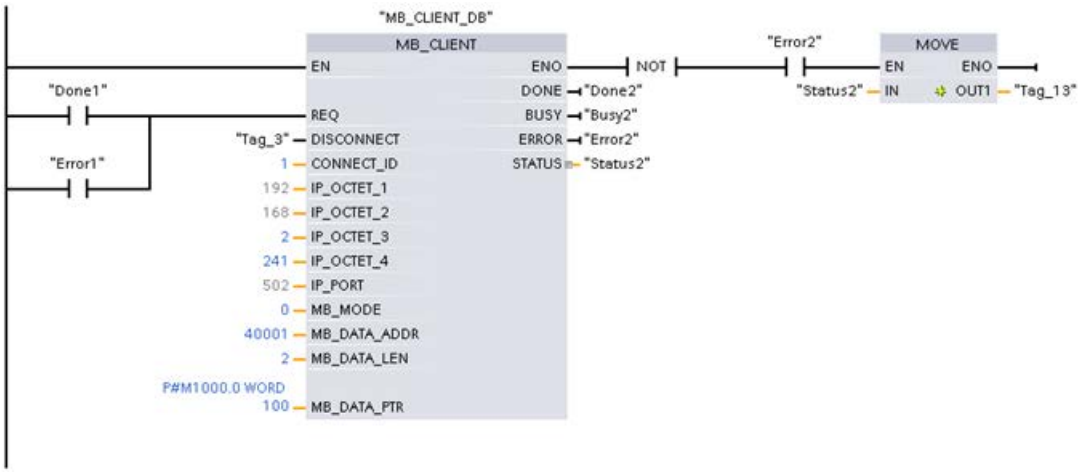

MB_SERVER Beispiel: Mehrere TCP-Verbindungen

Beschreibung

Sie können mehrere Modbus-TCP Server-Verbindungen verwenden. Hierzu muss die Anweisung "MB_SERVER" für jede Verbindung unabhängig aufgerufen werden.

Jede Verbindung benötigt:

- einen unabhängigen Instanz-Datenbaustein der Anweisung
- eine eindeutige Verbindungs-ID
- einen eigenen IP-Port (bei der S7-1200 ist pro IP-Port nur eine Verbindung zulässig)

Zur Optimierung der Leistung sollte "MB_SERVER" für jede Verbindung einmal pro Programmzyklus ausgeführt werden.

Netzwerk 1: Verbindung #1 mit zugehörigem IP-Port, Verbindungs-ID und Instanz-DB

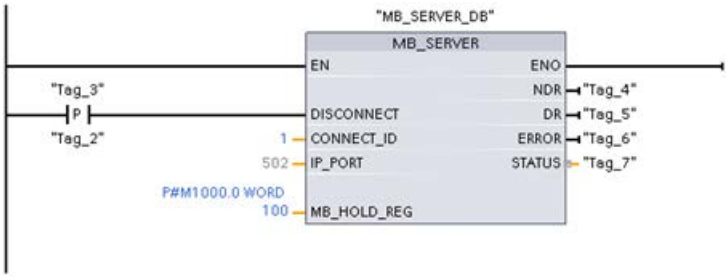

Netzwerk 2: Verbindung #1 mit zugehörigem IP-Port, Verbindungs-ID und Instanz-DB

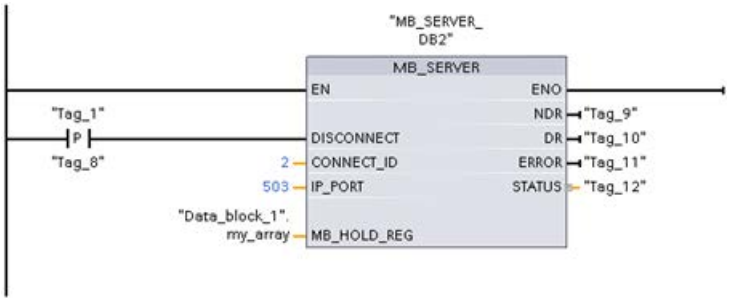

MODBUS (TCP) für die Bibliotheks-Versionen ab V4.0 der S7-1200-CPUs und ab V3.x der S7-1500-CPUs

MB_CLIENT: Als Modbus-TCP-Client über PROFINET kommunizieren

Beschreibung MB_CLIENT

Beschreibung

Die Anweisung "MB_CLIENT" kommuniziert als Modbus-TCP-Client über die PROFINET-Verbindung. Über die Anweisung "MB_CLIENT" bauen Sie eine Verbindung zwischen dem Client und dem Server auf, senden Modbus-Anfragen und empfangen Antworten und steuern den Verbindungsabbau des Modbus-TCP-Clients.

Für die S7-1200 mit Firmware-Version V4.0 können Sie die Anweisung "MB_CLIENT" bis einschließlich Bibliotheks-Version V3.1 benutzen. Mit der S7-1200 ab Firmware-Version V4.1 und der S7-1500 können Sie die Anweisung "MB_CLIENT" aller Bibliotheks-Versionen benutzen.

Die Verbindung kann über die lokale Schnittstelle der CPU oder CM/CP erfolgen.

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

Mehrfache Client-Verbindungen

Ein Modbus TCP-Client kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten. Modbus-TCP Verbindungen können hierbei auch gemeinsam von "MB_CLIENT"- und/oder "MB_SERVER"-Instanzen genutzt werden.

Bei einzelnen Client-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_CLIENT"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jede "MB_CLIENT"-Verbindung muss eine eindeutige IP-Adresse des Servers spezifiziert sein.
- Jede "MB_CLIENT"-Verbindung benötigt eine eindeutige Verbindungs-ID.
  Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Eindeutige Nummern des IP-Ports werden je nach Server-Konfiguration benötigt oder nicht.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_CLIENT":

Parameter	Deklaration	Datentyp	Beschreibung
REQ (Seite 4296)	Input	BOOL	Modbus-Anfrage an den Modbus TCP-Server Der Parameter REQ ist pegelgesteuert. Das heißt, solange der Eingang gesetzt ist (REQ=true), sendet die Anweisung Kommunikationsanfragen. • Nach dem Starten der Modbus-Anfrage wird der Instanz-DB für andere Clients gesperrt. • Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. • Wird während einer laufenden Modbus-Anfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.
DISCONNECT (Seite 4296)	Input	BOOL	Über den Parameter steuern Sie den Verbindungsaufbau- und abbau zu dem Modbus-Server: • 0: Kommunikationsverbindung zu dem am Parameter CONNECT konfigurierten Verbindungspartner herstellen (siehe Parameter CONNECT). • 1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS der Wert 0003 ausgegeben. Ist der Parameter REQ beim Verbindungsaufbau gesetzt, wird die Modbus-Anfrage sofort gesendet.
MB_MODE (Seite 4297)	Input	USINT	Auswahl des Modus der Modbus-Anfrage (Lesen, Schreiben oder Diagnose) bzw. direkte Auswahl einer Modbus-Funktion
MB_DATA_ADDR (Seite 4297)	Input	UDINT	abhängig von MB_MODE
MB_DATA_LEN	Input	UINT	Datenlänge: Anzahl der Bits oder Wörter für den Datenzugriff (siehe Parameter MB_MODE, MB_DATA_ADDR und MB_DATA_LEN (Seite 4297)).
MB_DATA_PTR (Seite 4299)	InOut	VARIANT	Zeiger auf einen Datenpuffer für die vom Modbus-Server zu empfangenden oder zum Modbus-Server zu sendenden Daten.
CONNECT (Seite 4300)	InOut	VARIANT	Zeiger auf die Struktur der Verbindungsbeschreibung Die folgenden Strukturen (Systemdatentypen) können verwendet werden: • TCON_IP_v4: Enthält alle Adressierungsparameter, die für den Aufbau einer programmierten Verbindung benötigt werden. Bei Verwendung von TCON_IP_v4 wird die Verbindung beim Aufruf der Anweisung "MB_CLIENT" hergestellt. • TCON_Configured: Enthält die Adressierungsparameter einer konfigurierten Verbindung. Bei Verwendung von TCON_Configured wird eine vorhandene Verbindung genutzt, die nach dem Laden der Hardware-Konfiguration durch die CPU hergestellt wurde.

Parameter	Deklaration	Datentyp	Beschreibung
DONE	Out	BOOL	Das Bit am Ausgangsparameter DONE wird auf "1" gesetzt, sobald der letzte Modbus-Auftrag ohne Fehler ausgeführt wurde.
BUSY	Out	BOOL	• 0: Keine Modbus-Anfrage in Bearbeitung • 1: Modbus-Anfrage wird bearbeitet Der Ausgangsparameter BUSY wird beim Verbindungsaufbau und -abbau nicht gesetzt.
ERROR	Out	BOOL	• 0: Kein Fehler • 1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter STATUS angezeigt.
STATUS (Seite 4303)	Out	WORD	Detaillierte Statusinformation der Anweisung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Konsistente Input-Daten während eines "MB_CLIENT"-Aufrufs

Sobald eine Modbus-Client Anweisung aufgerufen wird, werden die Werte der Input-Parameter intern gespeichert und dann beim Folgeaufruf verglichen. Der Vergleich wird verwendet, um zu bestimmen, ob dieser bestimmte Aufruf die laufende Anfrage initialisiert hat. Mehrere Aufrufe von "MB_CLIENT" können durchgeführt werden, indem Sie einen gemeinsamen Instanz-DB verwenden. Die Werte der Input-Parameter dürfen hierbei nicht verändert werden, solange eine "MB_CLIENT"-Instanz ausgeführt wird. Werden die Input-Parameter während der Ausführung verändert, kann durch "MB_CLIENT" nicht überprüft werden, ob die Instanz gerade ausgeführt wird.

Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_CLIENT".

Variable	Datentyp	Startwert	Beschreibung
Blocked_Proc_Timeout	REAL	3.0	Wartezeit in Sekunden, bevor bei einer blockierten Modbus-Instanz die statische Variable ACTIVE rückgesetzt wird. Dies kann zum Beispiel auftreten, wenn bei einer ausgegebenen Client-Anfrage die Ausführung der Client-Funktion abgebrochen wird, bevor die Anfrage vollständig ausgeführt wurde. Die Wartezeit muss zwischen 0,5 s und 55 s liegen.
MB_Transaction_ID	WORD	1	Transaktions-ID des Modbus-TCP-Protokolls. Der Startwert von "1" sollte nur geändert werden, wenn der Modbus-TCP Server einen anderen Wert benötigt.

Variable	Datentyp	Startwert	Beschreibung
MB_Unit_ID	BYTE	255	Unit Identifier Modbus-Gerätekennung: Ein Modbus TCP-Server wird über seine IP-Adresse angesprochen. Deshalb wird der Parameter MB_UNIT_ID bei der Modbus TCP-Adressierung nicht verwendet. Der Parameter MB_UNIT_ID entspricht dem Feld der Slaveadresse beim Modbus RTU-Protokoll. Wenn ein Modbus TCP-Server als Gateway zu einem Modbus RTU-Protokoll verwendet wird, kann das Slavegerät im seriellen Netzwerk über MB_UNIT_ID identifiziert werden. Der Parameter MB_UNIT_ID würde in diesem Fall die Anforderung an die richtige Modbus RTU-Slaveadresse weiterleiten. Beachten Sie bitte, dass einige Modbus TCP-Geräte den Parameter MB_UNIT_ID möglicherweise für die Initialisierung innerhalb eines eingeschränkten Wertebereichs benötigen. Über die folgenden Links erhalten Sie weitere Informationen zum Parameter MB_Unit_ID: https://support.industry.siemens.com/cs/ww/de/view/102420337 (https://support.industry.siemens.com/cs/ww/de/view/102420337) http://support.automation.siemens.com/WW/view/de/109736516 (http://support.automation.siemens.com/WW/view/de/109736516)
RCV_TIMEOUT	REAL	2.0	Zeitintervall in Sekunden, in dem die Anweisung "MB_CLIENT" auf eine Antwort des Servers wartet. Es muss zwischen 0,5 s und 55 s liegen.
Connected	BOOL	0	Gibt an, ob die Verbindung zum zugewiesenen Server verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.
RETRIES	WORD	0	Anzahl der Sende-Versuche, die die Anweisung "MB_CLIENT" unternimmt, bevor sie den Fehler W#16#80C8 zurückgibt.

Hinweis

Variable MB_Transaction_ID

Wenn die Transaktions-ID in der Antwort des Modbus-TCP-Servers nicht mit der Transaktions-ID des Auftrags von "MB_CLIENT" übereinstimmt, wartet die Anweisung "MB_CLIENT" für die Zeitdauer RCV_TIMEOUT * RETRIES auf die Antwort des Modbus-TCP-Servers mit der korrekten Transaktions-ID; danach gibt sie den Fehler W#16#80C8 zurück.

Beispiel

Ein Beispielprojekt zur Modbus-TCP-Kommunikation zwischen zwei S7-1500 CPUs finden Sie im Service und Support Portal unter der Beitrags-ID 94766380 (http://support.automation.siemens.com/WW/view/de/94766380).

In diesem Beispiel werden zwei Modbus-Funktionen verwendet. Für jede Modbus-Funktion wird über ein Modbus-Bausteinpärchen (MB_CLIENT und MB_SERVER) eine Modbus-TCP-Verbindung aufgebaut.

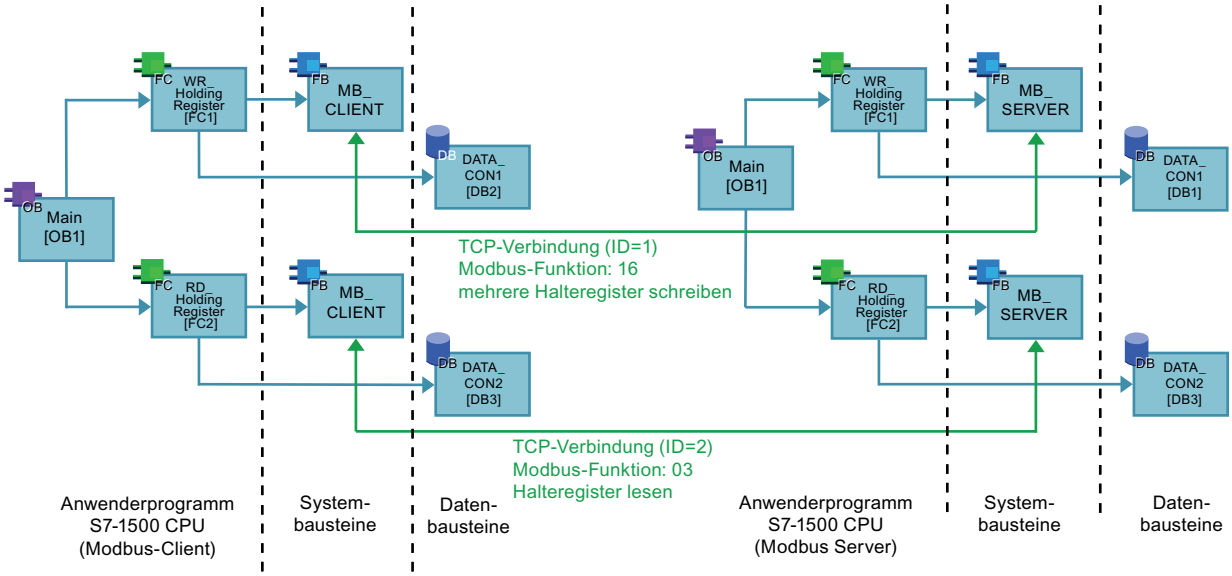

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter REQ und DISCONNECT

Beschreibung

Wenn kein Auftrag aktiv ist und wenn am Parameter DISCONNECT der Wert "0" anliegt, wird bei REQ=1 ein neuer Auftrag ausgeführt. Besteht noch keine Verbindung, wird diese bei der Ausführung hergestellt.

Wird die gleiche Instanz der Anweisung "MB_CLIENT" erneut ausgeführt (DISCONNECT=0 und REQ=1), bevor der aktive Auftrag ausgeführt wurde, wird dieser nicht im Anschluss an den aktiven Auftrag ausgeführt. Erst nach Beendigung des aktiven Auftrags kann ein neuer Auftrag angestoßen werden (REQ=1).

Der Status der Ausführung wird über die Ausgangsparameter ausgegeben. Diese können Sie nutzen, um bei einer sequenziellen Ausführung der Anweisung "MB_CLIENT" den Ausführungsstatus zu überwachen.

Parameter MB_MODE, MB_DATA_ADDR und MB_DATA_LEN

Beschreibung

Für die Werte 0, 1 und 2 von MB_MODE legt die Kombination der Parameter MB_MODE, MB_DATA_ADDR und MB_DATA_LEN den Modbus-Funktionscode fest, der in der aktuellen Modbus-Nachricht verwendet wird:

- MB_MODE enthält die Information, ob gelesen oder geschrieben werden soll. MB_MODE=0: Lesen, MB_MODE=1 und 2: Schreiben (Hinweis: Bei MB_MODE=2 wird nicht zwischen Modbus-Funktion 15 und 05 bzw. zwischen Modbus-Funktion 16 und 06 unterschieden.)
- MB_DATA_ADDR enthält die Information, was gelesen bzw. geschrieben werden soll, sowie eine Adressinformation, aus der die Anweisung "MB_CLIENT" die remote Adresse berechnet.
- MB_DATA_LEN enthält die Anzahl der zu lesenden / schreibenden Werte.

Beispiele:

- Die Kombination MB_MODE=1, MB_DATA_ADDR=1, MB_DATA_LEN=1 legt den Funktionscode 05 fest. Ab der remoten Adresse 0 wird 1 Ausgangs-Bit geschrieben.
- Die Kombination MB_MODE=1, MB_DATA_ADDR=1, MB_DATA_LEN=2 legt den Funktionscode 15 fest. Ab der remoten Adresse 0 werden 2 Ausgangs-Bits geschrieben.

Für die Werte 101 bis 106 und 115 bis 116 von MB_MODE gilt:

- MB_MODE legt den Modbus-Funktionscode fest.
- MB_DATA_ADDR enthält die remote Adresse.
- MB_DATA_LEN enthält die Anzahl der zu lesenden / schreibenden Werte.

Beispiel:

- MB_MODE=104, MB_DATA_ADDR=17.834, MB_DATA_LEN=125
  - MB_MODE=104 legt den Funktionscode 04 (Eingangs-Worte lesen) fest.
  - MB_DATA_ADDR=17.834 legt die remote Adresse 17.834 fest.
  - MB_DATA_LEN=125 legt fest, dass 125 Werte gelesen werden.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Eingangsparametern MB_MODE, MB_DATA_ADDR, MB_DATA_LEN der Anweisung "MB_CLIENT" und der zugehörigen Modbus-Funktion.

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus-Funktion	Funktion und Datentyp
0	1 bis 9.999	1 bis 2.000	01	1 bis 2.000 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	10.001 bis 19.999	1 bis 2.000	02	1 bis 2.000 Eingangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	• 40.001 bis 49.999 • 400.001 bis 465.535	1 bis 125	03	• 1 bis 125 Halteregister auf der remoten Adresse 0 bis 9.998 lesen • 1 bis 125 Halteregister auf der remoten Adresse0 bis 65.534 lesen

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus-Funktion	Funktion und Datentyp
0	30.001 bis 39.999	1 bis 125	04	1 bis 125 Eingangs-Worte auf der remoten Adresse 0 bis 9.998 lesen
1	1 bis 9.999	1	05	1 Ausgangs-Bit auf der remoten Adresse 0 bis 9.998 schreiben
1	• 40.001 bis 49.999 • 400.001 bis 465.535	1	06	• 1 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 1 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
1	1 bis 9.999	2 bis 1.968	15	2 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben
1	• 40.001 bis 49.999 • 400.001 bis 465.535	2 bis 123	16	• 2 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 2 bis 123 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
2	1 bis 9.999	1 bis 1.968	15	1 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben
2	• 40.001 bis 49.999 • 400.001 bis 465.535	1 bis 123	16	• 1 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 1 bis 123 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
11	Die Parameter MB_DATA_ADDR und MB_DATA_LEN werden bei der Ausführung dieser Funktion nicht ausgewertet.		11	Status-Wort und Ereigniszähler des Servers lesen: • Das Status-Wort bildet den Bearbeitungszustand ab (0 - nicht in Bearbeitung, 0xFFFF - in Bearbeitung). • Der Ereigniszähler wird inkrementiert, wenn die Modbus-Anfrage erfolgreich ausgeführt wurde. Wenn bei der Ausführung einer Modbus-Funktion ein Fehler aufgetreten ist, wird zwar eine Nachricht vom Server gesendet, der Ereigniszähler jedoch nicht inkrementiert.
80	-	1	08	Überprüfung des Server-Status durch den Diagnosecode 0x0000 (Rückschleifentest - der Server sendet die Abfrage zurück): • 1 WORD pro Aufruf
81	-	1	08	Ereigniszähler des Servers über den Diagnosecode 0x000A rücksetzen: • 1 WORD pro Aufruf
101	0 bis 65.535	1 bis 2.000	01	1 bis 2.000 Ausgangs-Bits auf der remoten Adresse 0 bis 65.535 lesen
102	0 bis 65.535	1 bis 2.000	02	1 bis 2.000 Eingangs-Bits auf der remoten Adresse 0 bis 65.535 lesen
103	0 bis 65.535	1 bis 125	03	1 bis 125 Halteregister auf der remoten Adresse 0 bis 65.535 lesen
104	0 bis 65.535	1 bis 125	04	1 bis 125 Eingangs-Worte auf der remoten Adresse 0 bis 65.535 lesen
105	0 bis 65.535	1	05	1 Ausgangs-Bit auf der remoten Adresse 0 bis 65.535 schreiben

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus-Funktion	Funktion und Datentyp
106	0 bis 65.535	1	06	1 Halteregister auf der remoten Adresse 0 bis 65.535 schreiben
115	0 bis 65.535	1 bis 1.968	15	1 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 65.535 schreiben
116	0 bis 65.535	1 bis 123	16	1 bis 123 Halteregister auf der remoten Adresse 0 bis 65.535 schreiben
3 bis 10, 12 bis 79, 82 bis 100, 107 bis 114, 117 bis 255				reserviert

Hinweis

Unit Identifier für RTU-Geräte über MB-TCP

Die Gerätekennung (Unit Identifier) für RTU-Geräte über Modbus-TCP finden Sie als Variable "MB_Unit_ID" in den editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_CLIENT". Der Unit Identifier dient der eindeutigen Zuordnung der Koppelpartner.

Siehe auch

Beschreibung MB_CLIENT (Seite 4291)

Parameter MB_DATA_PTR

Beschreibung

Der Parameter MB_DATA_PTR ist ein Zeiger auf einen Datenpuffer für die vom Modbus-Server zu empfangenden oder zum Modbus-Server zu sendenden Daten. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie für einen Puffer im Speicherbereich (M) einen Zeiger im Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Abhängig vom Speicherbereich, in dem der Datenpuffer liegt, kann MB_DATA_PTR auf unterschiedliche Datenstrukturen verweisen:

- Wenn man einen Global-DB mit optimiertem Zugriff verwendet, kann MB_DATA_PTR auf eine Variable mit elementarem Datentyp oder ein Array von elementaren Datentypen verweisen. Folgende Datentypen werden unterstützt:

Datentyp	Länge in Bits
Bool	1
Byte, SInt, USInt, Char	8
Word, Int, WChar, UInt	16
DWord, DInt, UDInt, Real	32

  Dabei können Sie bei allen Modbus-Funktionen sämtliche unterstützten Datentypen verwenden. MB_CLIENT kann z. B. ein empfangenes Bit auch in eine Variable vom Typ Byte auf eine vorgegebene Adresse schreiben, ohne andere Bits in diesem Byte zu verändern. Man muss also nicht zwingend ein Array von Bits haben, um die bitorientierten Funktionen ausführen zu können.

- Wenn man als Speicherbereich einen Merkerbereich oder einen Global-DB mit Standardzugriff verwendet, besteht für MB_DATA_PTR keine Einschränkung mehr auf die elementaren Datentypen; MB_DATA_PTR kann dann auch auf komplexe Datenstrukturen wie PLC-Datentypen (UDTs) und Systemdatentypen (SDTs) verweisen.

Hinweis

Verwendung eines Merkerbereichs als Datenpuffer

Wenn Sie für MB_DATA_PTR einen Merkerbereich als Datenpuffer verwenden, müssen Sie dessen Größe beachten. Sie beträgt bei S7-1500-CPUs 16 KByte, bei S7-1200-CPUs 8 KByte.

Parameter CONNECT

Verbindungsbeschreibungen am Parameter CONNECT bis einschließlich Anweisungs-Version V4.1 von MB_CLIENT

Für die Anweisung "MB_CLIENT" können zwei verschiedene Verbindungsbeschreibungen verwendet werden:

- Programmierte Verbindungen mit der Struktur TCON_IP_v4
  Die Verbindungsparameter werden in der Struktur TCON_IP_v4 hinterlegt und die Verbindung wird erst mit Aufruf der Anweisung "MB_CLIENT" eingerichtet.
- Konfigurierte Verbindungen mit der Struktur TCON_Configured
  Die konfigurierte Verbindung wurde bereits durch die CPU aufgebaut. Über die Struktur TCON_Configured geben Sie an, welche bestehende Verbindung für die Anweisung verwendet werden soll.

Jede Instanz der Anweisung "MB_CLIENT" benötigt eine eindeutige Verbindung. Erstellen Sie entsprechend für jede Instanz der Anweisung eine eigene Struktur TCON_IP_v4 oder TCON_Configured zur Verbindungsbeschreibung.

Verbindungsbeschreibung für programmierte Verbindungen

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_IP_v4:

- Achten Sie darauf, in der Struktur TCON_IP_v4 nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceID	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	-	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisung "MB_CLIENT" muss eine eindeutige ID verwenden.
4	ConnectionType	BYTE	11	Verbindungstyp Wählen Sie 11 (dezimal) für TCP. Andere Verbindungstypen sind nicht zulässig. Wird ein anderer Verbindungstyp (z. B. UDP) verwendet, wird am Parameter STATUS der Anweisung eine entsprechende Fehlermeldung ausgegeben.
5	ActiveEstablished	BOOL	TRUE	Kennung für die Art des Verbindungsaufbaus Wählen Sie TRUE für den aktiven Verbindungsaufbau.
6 … 9	RemoteAddress	ARRAY [1..4] of BYTE	-	IP-Adresse des Verbindungspartners (Modbus Server), z. B. für 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1
10 … 11	RemotePort	UINT	502	Port-Nummer des entfernten Verbindungspartners (Wertebereich: 1 bis 49151). Verwenden Sie die IP-Port-Nummer des Servers, zu dem der Client die Verbindung herstellt und über das TCP/IP-Protokoll kommuniziert (Standard-Wert: 502).
12 … 13	LocalPort	UINT	0	Port-Nummer des lokalen Verbindungspartners: • Port-Nummern: 1 bis 49151 • Beliebiger Port: "0"

Hinweis

Migration der Anweisung "MB_CLIENT" Version 2.1

Die Parameter CONNECT_ID, IP_PORT und IP_OCTET_x sind in Version 3.0 der Anweisung "MB_CLIENT" in der Struktur TCON_IP_v4 abgebildet:

- Der Parameter CONNECT_ID der Anweisung "MB_CLIENT" V2.1 entspricht dem Parameter ID von TCON_IP_v4.
- Der Parameter IP_PORT der Anweisung "MB_CLIENT" V2.1 entspricht dem Parameter RemotePort von TCON_IP_v4.
- Die vier Parameter IP_OCTET_x der Anweisung "MB_CLIENT" V2.1 entsprechen dem Array des Parameters RemoteAddress von TCON_IP_v4.

Verbindungsbeschreibung für konfigurierte Verbindungen

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_Configured.

- Achten Sie darauf, in der Struktur TCON_Configured nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceId	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	-	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Geben Sie die Verbindungs-ID der bestehenden Verbindung an.
4	ConnectionType	BYTE	0	Verbindungstyp Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.

Verbindungsbeschreibungen am Parameter CONNECT für die Anweisungs-Versionen > V4.1 und für die Firmware-Versionen >= V2.5 der S7-1500-CPUs

Neben den Verbindungen mit den Strukturen TCON_IP_v4 und TCON_Configured sind für die Anweisungs-Versionen > V4.1 von MB_CLIENT und ab Firmware-Version V2.5 der S7-1500-CPUs die folgenden Verbindungsbeschreibungen für TCP möglich:

- TCON_IP_V4_SEC
  Zur Beschreibung siehe AUTOHOTSPOT
- TCON_QDN
  Zur Beschreibung siehe AUTOHOTSPOT
- TCON_QDN_SEC
  Zur Beschreibung siehe AUTOHOTSPOT

Für weitergehende Informationen zu gesicherten Verbindungen siehe AUTOHOTSPOT

Parameter STATUS

Parameter STATUS (allgemeine Statusinformationen)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
7000	Kein Auftrag aktiv und keine Verbindung aufgebaut (REQ=0, DISCONNECT=1).
7001	Verbindungsaufbau angestoßen.
7002	Zwischenaufruf. Verbindung wird aufgebaut.
7003	Verbindung wird abgebaut.
7004	Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.
7005	Daten werden gesendet.
7006	Daten werden empfangen.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (Protokoll-Fehler)

STATUS* (W#16#)	Lokaler bzw. remoter Fehler	Fehlercode in der Antwort von MB_SERVER (B#16#)	Beschreibung
80C8	lokal	-	Keine Antwort des Servers im definierten Zeitraum. Überprüfen Sie die Verbindung zu dem Modbus-Server. Dieser Fehler wird erst nach Abschluss der konfigurierten erneuten Versuche gemeldet. Empfängt die Anweisung "MB_CLIENT" keine Antwort mit der ursprünglich übermittelten Transaktions-ID (siehe statische Variable MB_TRANSACTION_ID) innerhalb des definierten Zeitraums, wird dieser Fehlercode ausgegeben.
8380	lokal	-	Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8381	remote	01	Funktionscode wird nicht unterstützt.
8382	lokal	-	• Die Länge des Modbus-Frames im Frame-Header stimmt mit der Anzahl der empfangenen Bytes nicht überein. • Die Anzahl der Bytes stimmt mit den tatsächlich übertragenen Bytes nicht überein (nur Funktionen 1-4). Dies ist z. B. dann der Fall, wenn "MB_CLIENT" eine ungerade Anzahl von Worten anfordert, "MB_SERVER" aber immer eine gerade Anzahl von Worten sendet. • Die Startadresse im empfangenen Frame stimmt mit der gespeicherten Startadresse nicht überein (Funktionen 5, 6, 15, 16). • Anzahl der Wörter stimmt mit den tatsächlich übertragenen Wörtern nicht überein (Funktionen 15 und 16).
	remote	03	Unzulässige Längenangabe im empfangenen Modbus-Frame. Überprüfen Sie die Server-Seite.

STATUS* (W#16#)	Lokaler bzw. remoter Fehler	Fehlercode in der Antwort von MB_SERVER (B#16#)	Beschreibung
8383	lokal	-	Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs von MB_DATA_PTR (Seite 4299).
	remote	02	Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs des Servers
8384	lokal	-	• Ungültiger Exception Code emfangen. • Es wurde ein anderer Datenwert empfangen, als ursprünglich vom Client gesendet wurde (Funktionen 5, 6 und 8). • Ungültiger Status-Wert empfangen (Funktion 11)
	remote	03	Fehler bei Datenwert für die Funktion 5
8385	lokal	-	• Diagnosecode nicht unterstützt. • Es wurde ein anderer Subfunktionscode empfangen, als ursprünglich vom Client gesendet wurde (Funktion 8).
	remote	03	Diagnosecode nicht unterstützt
8386	lokal	-	Empfangener Funktionscode entspricht nicht dem ursprünglich gesendeten.
8387	lokal	-	Die Protokoll-ID des vom Server empfangenen Modbus TCP Frames ist ungleich "0".
8388	lokal	-	Durch den Modbus-Server wurde eine andere Datenlänge gesendet, als abgefragt wurde. Dieser Fehler tritt nur bei Verwendung der Modbus-Funktionen 5, 6, 15 oder 16 auf.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".			

Parameter STATUS (Parameter-Fehler)

STATUS* (W#16#)	Beschreibung
80B6	Ungültiger Verbindungstyp, nur TCP-Verbindungen werden unterstützt.
80BB	Ungültiger Wert am Parameter ActiveEstablished (Kennung für die Art des Verbindungsaufbaus, siehe Parameter CONNECT (Seite 4300)): • Serverseitig ist nur der passive Verbindungsaufbau zulässig (ActiveEstablished = FALSE). • Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ActiveEstablished = TRUE).
8188	Ungültiger Wert am Parameter MB_MODE.
8189	Ungültige Adressierung der Daten am Parameter MB_DATA_ADDR.
818A	Ungültige Datenlänge am Parameter MB_DATA_LEN.
818B	Ungültiger Zeiger am Parameter MB_DATA_PTR. Überprüfen Sie zudem die Werte der Parameter MB_DATA_ADDR (Seite 4299) und MB_DATA_LEN.
818C	Zeitfehler bei dem Parameter BLOCKED_PROC_TIMEOUT oder RCV_TIMEOUT (siehe statische Variablen der Anweisung). BLOCKED_PROC_TIMEOUT und RCV_TIMEOUT müssen zwischen 0,5 s und 55 s liegen.

STATUS* (W#16#)	Beschreibung
8200	• Eine andere Modbus-Anfrage wird zur Zeit über den Port verarbeitet. • Eine weitere Instanz von MB_CLIENT mit den gleichen Verbindungsparametern bearbeitet eine bereits bestehende Modbus-Anfrage.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Fehlercodes der intern verwendeten Kommunikations-Anweisungen

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" und "TRESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_CLIENT" möglich. Im Abschnitt "Static" werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Hinweis

Kommunikationsfehler beim Senden oder Empfangen von Daten

Wenn beim Senden bzw. Empfangen von Daten ein Kommunikationsfehler auftritt (80C4 (Temporary communications error. The specified connection is temporarily down.), 80C5 (Remote partner closed connection actively.), 80A1 (The specified connection is disconnected or is not yet established.)), dann wird die bestehende Verbindung abgebaut.

Das bedeutet auch, dass Sie alle STATUS-Werte sehen, die beim Verbindungsabbau zurückgegeben werden, und erst wenn die Verbindung abgebaut ist, wird der STATUS-Code ausgegeben, der die Ursache für den Verbindungsabbau gewesen ist.

Beispiel: Tritt beim Empfangen der Daten ein temporärer Kommunikationsfehler auf, dann wird zunächst der STATUS 7003 (ERROR=false) ausgegeben und dann 80C4 (ERROR=true).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

MB_SERVER: Als Modbus-TCP-Server über PROFINET kommunizieren

Beschreibung MB_SERVER

Beschreibung

Die Anweisung "MB_SERVER" kommuniziert als Modbus-TCP-Server über eine PROFINET-Verbindung. Die Anweisung "MB_SERVER" verarbeitet Verbindungsanfragen eines Modbus-TCP-Clients, empfängt und bearbeitet Modbus-Anfragen und sendet Antwort-Meldungen.

Für die S7-1200 mit Firmware-Version V4.0 können Sie die Anweisung "MB_SERVER" bis einschließlich Bibliotheks-Version V3.1 benutzen. Mit der S7-1200 ab Firmware-Version V4.1 und der S7-1500 können Sie die Anweisung "MB_SERVER" aller Bibliotheks-Versionen benutzen.

Die Verbindung kann über die lokale Schnittstelle der CPU oder CM/CP erfolgen.

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

ACHTUNG
Security-Hinweis Beachten Sie, dass jeder Client des Netzwerks lesenden und schreibenden Zugriff erhält auf das Prozessabbild der Ein- und Ausgänge und auf den durch das Modbus-Halteregister definierten Datenbaustein oder Merkerbereich. Zum Schutz vor unbefugten Lese- und Schreibvorgänge besteht die Möglichkeit, den Zugriff auf eine IP-Adresse einzuschränken. Beachten Sie jedoch, dass sich die freigegebene Adresse auch für einen unbefugten Zugriff nutzen lässt.

Mehrfache Server-Verbindungen

Sie können mehrfache Server-Verbindungen erstellen. Dies erlaubt es einer einzelnen CPU, gleichzeitig Verbindungen von mehreren Modbus-TCP-Clients anzunehmen.

Ein Modbus-TCP-Server kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt.

Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Modbus-TCP Verbindungen können hierbei auch gemeinsam von "MB_CLIENT"- und/oder "MB_SERVER"-Instanzen genutzt werden.

Bei Server-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_SERVER"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Jede "MB_SERVER"-Verbindung muss eine eindeutige Verbindungs-ID verwenden.
  Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Für jede Verbindung muss die Anweisung "MB_SERVER" einzeln aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_SERVER":

Parameter	Deklaration	Datentyp	Beschreibung
DISCONNECT	Input	BOOL	Die Anweisung "MB_SERVER" geht eine passive Verbindung mit einer Partner-Baugruppe ein. Der Server reagiert auf eine Verbindungsanfrage von der IP-Adresse, die in der SDT "TCON_IP_v4" am CONNECT-Parameter eingegeben ist. Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird: • 0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufgebaut. • 1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine anderen Vorgänge ausgeführt. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS der Wert 0003 ausgegeben.
MB_HOLD_REG (Seite 4311)	InOut	VARIANT	Zeiger auf das Modbus-Halteregister der "MB_SERVER"-Anweisung MB_HOLD_REG muss immer auf einen Speicherbereich verweisen, der größer als zwei Byte ist. Das Halteregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (Lesen), 6 (Schreiben), 16 (Mehrfachschreiben) und 23 (Schreiben und Lesen in einem Auftrag) zugreifen darf. Verwenden Sie als Halteregister einen globalen Datenbaustein mit optimiertem Zugriff oder den Speicherbereich der Merker.
CONNECT (Seite 4312)	InOut	VARIANT	Zeiger auf die Struktur der Verbindungsbeschreibung Die folgenden Strukturen (SDTs) können verwendet werden: • TCON_IP_v4: Enthält alle Adressierungsparameter, die für den Aufbau einer programmierten Verbindung benötigt werden. Defaultmäßig ist sie 0.0.0.0 (beliebige IP-Adresse), Sie können auch eine bestimmte IP-Adresse angeben, wodurch der Server nur auf die Anfragen von dieser Adresse reagiert. Bei Verwendung von TCON_IP_v4 wird die Verbindung beim Aufruf der Anweisung "MB_SERVER" hergestellt. • TCON_Configured (nur bei S7-1500): Enthält die Adressierungsparameter einer konfigurierten Verbindung. Bei Verwendung von TCON_Configured wird die Verbindung nach dem Laden der Hardware-Konfiguration durch die CPU hergestellt.
NDR	Output	BOOL	"New Data Ready": • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client geschrieben
DR	Output	BOOL	"Data Read": • 0: Keine Daten gelesen • 1: Daten durch den Modbus-Client gelesen

Parameter	Deklaration	Datentyp	Beschreibung
ERROR	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_SERVER" ein Fehler auf, wird der Ausgang am Parameter ERROR auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS angezeigt.
STATUS (Seite 4314)	Output	WORD	Detaillierte Statusinformation der Anweisung.

Hinweis

Benutzung der Modbus-Funktion 23 in der Anweisung "MB_SERVER"

Die Anweisung "MB_SERVER" unterstützt die Benutzung der Modbus-Funktion 23, mit der Sie in einem Auftrag auf ein Halteregister schreiben und von einem Halteregister lesen können; die Anweisung "MB_CLIENT" unterstützt diese Funktion jedoch NICHT und liefert einen Fehlercode zurück.

Beachten Sie außerdem, dass bei einem Auftrag, der sowohl einen Lesezugriff als auch einen Schreibzugriff enthält, der Schreibzugriff vor dem Lesezugriff ausgeführt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Statische Variablen von MB_SERVER in der Anweisungs-Version V4.2

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-Datenbausteins der Anweisung "MB_SERVER", die im Programm verwendet werden können. Auf die Variable HR_Start_Offset können Sie schreibend zugreifen. Auf die übrigen Variablen können Sie lesend zugreifen, um den Modbus-Status zu überwachen.

Variable	Datentyp	Startwert	Beschreibung
HR_Start_Offset	WORD	0	Zuweisen der Anfangsadresse des Modbus-Halteregisters.
QB_Start	WORD	0	Anfangsadresse des zulässigen Adressierungsbereichs der Ausgänge, die geschrieben werden können (Byte 0 bis 65535)
QB_Count	WORD	0xFFFF	Anzahl der Ausgangs-Bytes, die durch den Modbus-Master geschrieben werden können. Beispiel: QB_Start=0 und QB_Count=10: Ausgangs-Bytes 0 bis 9 können geschrieben werden. QB_Count=0: Kein Ausgangsbyte kann geschrieben werden.
Request_Count	WORD	0	Anzahl aller Anfragen, die durch den Server empfangen wurden.
Server_Message_Count	WORD	0	Anzahl aller empfangenen Meldungen für den jeweiligen Server.
Xmt_Rcv_Count	WORD	0	Zähler zur Erfassung der Anzahl der Übertragungen, bei denen ein Fehler aufgetreten ist. Der Zähler wird nur dann erhöht, wenn eine ungültige Modbus-Anfrage empfangen wurde.
Exception_Count	WORD	0	Zähler zur Erfassung der Anzahl von Fehlern, die spezifisch für Modbus sind und eine Fehlermeldung an "MB_CLIENT" hervorrufen.

Variable	Datentyp	Startwert	Beschreibung
Success_Count	WORD	0	Ereigniszähler zur Erfassung der Anzahl der Anfragen, die vom Server erfolgreich ausgeführt werden konnten.
Connected	BOOL	FALSE	Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: TRUE = verbunden, FALSE = nicht verbunden.

Abbilden der Modbus-Adressen auf dem Prozessabbild

Die Anweisung "MB_SERVER" ermöglicht eingehenden Modbus-Funktionen (1, 2, 4, 5 und 15) direkt lesend und schreibend auf das Prozessabbild der Eingänge und Ausgänge der CPU zuzugreifen (Verwendung der Datentypen BOOL und WORD).

Bei den S7-1200-CPUs beträgt der Adressraum sowohl für das Prozessabbild der Eingänge als auch für das Prozessabbild der Ausgänge 1 KByte, bei den S7-1500-CPUs jeweils 32 KByte.

Die folgende Tabelle zeigt den Adressraum der oben genannten Modbus-Funktionen.

Modbus-Funktion					
Funktions-code	Funktion	Datenbereich	Adressraum		
01	Lesen: Bits	Output	0	bis	65.535
02	Lesen: Bits	Input	0	bis	65.535
04	Lesen: WORD	Input	0	bis	65.535
05	Schreiben: Bit	Output	0	bis	65.535
15	Schreiben: Bits	Output	0	bis	65.535

Eingehende Modbus-Anfragen mit den Funktionscodes 3, 6, 16 und 23 schreiben oder lesen die Modbus-Halteregister (das Halteregister spezifizieren Sie am Parameter MB_HOLD_REG).

Beispiel: Adressierung über statische Variable HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 0 (aus Sicht des MB_CLIENT bei 40.001). Diese Adressen entsprechen dem Adressraum des CPU-Speicherbereichs für das Halteregister. Sie können die Variable HR_Start_Offset auch so definieren, dass das Modbus-Halteregister eine andere Anfangsadresse als 0 erhält.

Beispiel: Das Halteregister beginnt am MW100 und hat eine Länge von 100 WORD. Ein Offset-Wert am Parameter HR_Start_Offset bewirkt, dass sich die Anfangsadresse des Halteregisters von 0 auf 20 verschiebt. Jede Adressierung des Halteregisters, die unterhalb der Adresse 20 und oberhalb der Adresse 119 liegt, ruft einen Fehler hervor.

HR_Start_Offset	Adresse	Minimum	Maximum
0	Modbus-Adresse (WORD)	0	99
	CPU-Adresse	MW100	MW298
20	Modbus-Adresse (WORD)	20	119
	CPU-Adresse	MW100	MW298

Modbus-Funktionen

In der folgenden Tabelle sind alle Modbus-Funktionen aufgelistet, die von der Anweisung "MB_SERVER" unterstützt werden.

Funktionscode	Beschreibung
01	Ausgangs-Bits lesen
02	Eingangs-Bits lesen
03	Halteregister lesen
04	Eingangs-Worte lesen
05	Ein Ausgangs-Bit schreiben
06	Ein Halteregister schreiben
08	Diagnosefunktion: • Echo-Test (Subfunktion 0x0000): Die Anweisung "MB_SERVER" empfängt ein Datenwort und sendet dieses unverändert an den Modbus-Client zurück. • Ereigniszähler zurücksetzen (Subfunktion 0x000A): Die Anweisung "MB_SERVER" setzt die folgenden Ereigniszähler zurück: "Success_Count", "Xmt_Rcv_Count", "Exception_Count", "Server_Message_Count" und "Request_Count".
11	Diagnosefunktion: Ereigniszähler der Kommunikation holen Die Anweisung "MB_SERVER" verwendet einen internen Ereigniszähler bei der Kommunikation zur Erfassung der Anzahl der erfolgreich ausgeführten Lese- und Schreibanfragen, die an den Modbus-Server gesendet werden. Der Ereigniszähler wird bei den Funktionen 8 oder 11 nicht inkrementiert. Gleiches gilt für Anfragen, die einen Kommunikationsfehler zur Folge haben, zum Beispiel falls ein Protokollfehler aufgetreten ist (z. B. der Funktionscode in der empfangenen Modbus-Anfrage nicht unterstützt wird).
15	Ausgangs-Bits schreiben
16	Halteregister schreiben
23	Mit einem Request Halteregister schreiben und Halteregister lesen

Beispiel

Ein Beispielprojekt zur Modbus-TCP-Kommunikation zwischen zwei S7-1500 CPUs finden Sie im Service und Support Portal unter der Beitrags-ID 94766380 (http://support.automation.siemens.com/WW/view/de/94766380).

In diesem Beispiel werden zwei Modbus-Funktionen verwendet. Für jede Modbus-Funktion wird über ein Modbus-Bausteinpärchen (MB_CLIENT und MB_SERVER) eine Modbus-TCP-Verbindung aufgebaut.

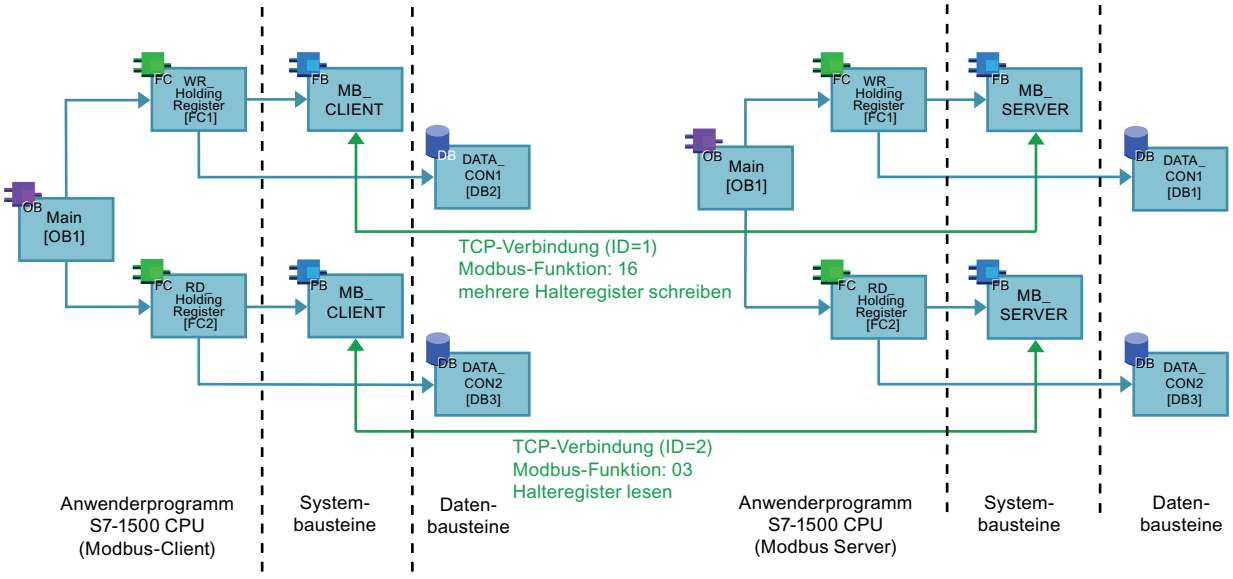

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 618)

Parameter MB_HOLD_REG

Beschreibung

Der Parameter MB_HOLD_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Speicherbereich können Sie einen globalen Datenbaustein oder einen Merker (M) verwenden.

- Die Obergrenze für die Anzahl von Adressen im Datenbaustein (D) ist durch die maximale Größe eines DB Ihrer CPU vorgegeben.
- Die Obergrenze für die Anzahl von Merkern (M) ist durch den maximalen Merkerbereich Ihrer CPU vorgegeben.

Die folgende Tabelle zeigt einige Beispiele zur Abbildung von Modbus-Adressen auf das Halteregister für die Modbus-Funktionen 3 (mehrere WORD lesen), 6 (ein WORD schreiben), 16 (mehrere WORD schreiben) und 23 (mehrere Wörter schreiben und lesen).

Modbus-Adressen	MB_HOLD_REG Parameter - Beispiele		
0	MW100	DB10.DBW0	"Recipe".ingredient[1]
1	MW102	DB10.DBW2	"Recipe".ingredient[2]
2	MW104	DB10.DBW4	"Recipe".ingredient[3]
3	MW106	DB10.DBW6	"Recipe".ingredient[4]
4	MW108	DB10.DBW8	"Recipe".ingredient[5]

Parameter CONNECT

Verbindungsbeschreibungen am Parameter CONNECT bis einschließlich Anweisungs-Version V4.2 von MB_SERVER

Für die Anweisung "MB_SERVER" können zwei verschiedene Verbindungsbeschreibungen verwendet werden:

- Programmierte Verbindungen mit der Struktur TCON_IP_v4
  Die Verbindungsparameter werden in der Struktur TCON_IP_v4 hinterlegt und die Verbindung wird erst mit Aufruf der Anweisung "MB_SERVER" eingerichtet.
- Konfigurierte Verbindungen mit der Struktur TCON_Configured (nur bei S7-1500)
  Die konfigurierte Verbindung wurde bereits durch die CPU aufgebaut. Über die Struktur TCON_Configured geben Sie an, welche bestehende Verbindung für die Anweisung verwendet werden soll.

Jede Instanz der Anweisung "MB_SERVER" benötigt eine eindeutige Verbindung. Erstellen Sie entsprechend für jede Instanz der Anweisung eine eigene Struktur TCON_IP_v4 oder TCON_Configured zur Verbindungsbeschreibung.

Verbindungsbeschreibung für programmierte Verbindungen

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_IP_v4.

Achten Sie darauf, in der Struktur TCON_IP_v4 nur Verbindungen vom Typ TCP anzugeben.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceID	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle
2 … 3	ID	CONN_OUC	-	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisung "MB_SERVER" muss eine eindeutige ID verwenden. Die ID darf auch nicht gleichzeitig von einer anderen Anweisung einer anderen Kommunikationsart verwendet werden.
4	ConnectionType	BYTE	11	Verbindungstyp Wählen Sie 11 (dezimal) für TCP. Andere Verbindungstypen sind nicht zulässig. Wird ein anderer Verbindungstyp (z. B. UDP) verwendet, wird am Parameter STATUS der Anweisung eine entsprechende Fehlermeldung ausgegeben.
5	ActiveEstablished	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus Wählen Sie FALSE für den passiven Verbindungsaufbau.
6 … 9	RemoteAddress	ARRAY [1..4] of BYTE	0.0.0.0	IP-Adresse des Verbindungspartners, z. B. für 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1 Soll die Anweisung "MB_SERVER" Verbindungsanfragen von jedem Verbindungspartner akzeptieren, verwenden Sie als IP-Adresse "0.0.0.0".

Byte	Parameter	Datentyp	Startwert	Beschreibung
10 … 11	RemotePort	UINT	0	Port-Nummer des entfernten Verbindungspartners (Wertebereich: 1 bis 49151). Soll die Anweisung "MB_SERVER" Verbindungsanfragen von jedem Port des entfernten Partner akzeptieren, verwenden Sie als Port-Nummer "0".
12 … 13	LocalPort	UINT	502	Port-Nummer des lokalen Verbindungspartners (Wertebereich: 1 bis 49151). Die Nummer des IP-Ports legt fest, welcher IP-Port für Verbindungsanfragen des Modbus-Clients überwacht wird. Der Standardwert ist 502.

Hinweis

Migration der Anweisung "MB_SERVER" Version 2.1

Die Parameter CONNECT_ID und IP_PORT sind in Version 3.0 der Anweisung "MB_SERVER" in der Struktur TCON_IP_v4 abgebildet:

- Der Parameter CONNECT_ID der Anweisung "MB_SERVER" V2.1 entspricht dem Parameter ID von TCON_IP_v4.
- Der Parameter IP_PORT der Anweisung "MB_SERVER" V2.1 entspricht dem Parameter LocalPort von TCON_IP_v4.

Verbindungsbeschreibung für konfigurierte Verbindungen

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON_Configured.

Achten Sie darauf, in der Struktur TCON_Configured nur Verbindungen vom Typ TCP anzugeben.

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 … 1	InterfaceID	HW_ANY	-	Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).
2 … 3	ID	CONN_OUC	-	Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). ID muss CPU-weit eindeutig sein. ID darf auch nicht gleichzeitig von einer anderen Anweisung einer anderen Kommunikationsart verwendet werden. Geben Sie die Verbindungs-ID der bestehenden Verbindung an.
4	ConnectionType	BYTE	-	Verbindungstyp Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.

Verbindungsbeschreibungen am Parameter CONNECT für die Anweisungs-Versionen > V4.2 und für die Firmware-Versionen >= V2.5 der S7-1500-CPUs

Neben den Verbindungen mit den Strukturen TCON_IP_v4 und TCON_Configured sind für die Anweisungs-Versionen > V4.2 von MB_SERVER und ab Firmware-Version V2.5 der S7-1500-CPUs die folgenden Verbindungsbeschreibungen für TCP möglich:

- TCON_IP_V4_SEC
  Zur Beschreibung siehe AUTOHOTSPOT
- TCON_QDN
  Zur Beschreibung siehe AUTOHOTSPOT
- TCON_QDN_SEC
  Zur Beschreibung siehe AUTOHOTSPOT

Für weitergehende Informationen zu gesicherten Verbindungen siehe AUTOHOTSPOT

Parameter STATUS

Parameter STATUS (allgemeine Statusinformationen)

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
7000	Kein Aufruf aktiv und keine Verbindung aufgebaut (REQ=0, DISCONNECT=1).
7001	Erstaufruf. Verbindungsaufbau angestoßen.
7002	Zwischenaufruf. Verbindung wird aufgebaut.
7003	Verbindung wird abgebaut.
7005	Daten werden gesendet.
7006	Daten werden empfangen.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS (Protokoll-Fehler)

STATUS* (W#16#)	Fehlercode in der Fehlermeldung von "MB_SERVER" (B#16#)	Beschreibung
8380	-	Empfangener Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8381	01	Funktionscode wird nicht unterstützt.

STATUS* (W#16#)	Fehlercode in der Fehlermeldung von "MB_SERVER" (B#16#)	Beschreibung
8382	03	Fehler bei Datenlänge: • Unzulässige Längenangabe im empfangenen Modbus-Frame • Die im Header des Modbus-Frames eingetragene Frame-Länge stimmt mit der Anzahl der tatsächlich empfangenen Bytes nicht überein. • Die im Header des Modbus-Frames eingetragene Anzahl der Bytes stimmt mit der Anzahl der tatsächlich empfangenen Bytes nicht überein (Funktionen 15 und 16).
8383	02	Fehler bei der Datenadresse oder Zugriff außerhalb des Adressbereichs des Halteregisters (Parameter MB_HOLD_REG (Seite 4311)).
8384	03	Fehler beim Datenwert (Funktion 05)
8385	03	Diagnosecode wird nicht unterstützt (bei Funktion 08)
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Parameter STATUS (Parameter-Fehler)

STATUS* (W#16#)	Beschreibung
80BB	Ungültiger Wert am Parameter ActiveEstablished (Kennung für die Art des Verbindungsaufbaus, siehe Parameter CONNECT (Seite 4312)): Serverseitig ist nur der passive Verbindungsaufbau zulässig (active_established = FALSE).
8187	Ungültiger Zeiger am Parameter MB_HOLD_REG. Datenbereich ist zu klein.
8389	Ungültige Datenbereich-Definition: • Unzulässiger Wert von data_type • DB-Nummer unzulässig oder nicht vorhanden: – Unzulässiger Wert von db – DB-Nummer existiert nicht – DB-Nummer wird bereits von einem anderen Datenbereich verwendet – DB mit optimiertem Zugriff – DB liegt nicht im Arbeitsspeicher • Unzulässiger Wert von length • Überlappung von MODBUS-Adressbereichen, die zu demselben MODBUS-Datentyp gehören
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Fehlercodes der intern verwendeten Kommunikations-Anweisungen

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_SERVER" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" und "T_RESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_SERVER" möglich. Im Abschnitt "Static" werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Hinweis

Kommunikationsfehler beim Senden oder Empfangen von Daten

Wenn beim Senden bzw. Empfangen von Daten ein Kommunikationsfehler auftritt (80C4 (Temporärer Kommunikationsfehler. Die angegebene Verbindung ist vorübergehend abgebaut.), 80C5 (Der remote Partner hat die Verbindung aktiv beendet.), 80A1 (Die angegebene Verbindung ist unterbrochen oder wurde noch nicht aufgebaut.)), dann wird die bestehende Verbindung abgebaut.

Das bedeutet auch, dass Sie alle STATUS-Werte sehen, die beim Verbindungsabbau zurückgegeben werden, und erst wenn die Verbindung abgebaut ist, wird der STATUS-Code ausgegeben, der die Ursache für den Verbindungsabbau gewesen ist.

Beispiel: Tritt beim Empfangen der Daten ein temporärer Kommunikationsfehler auf, dann wird zunächst der STATUS 7003 (ERROR=false) ausgegeben und dann 80C4 (ERROR=true).

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

GET_ERR_ID: Fehler-ID lokal abfragen (Seite 916)

Zugriff auf Datenbereiche in DBs anstelle des direkten Zugriffs auf MODBUS-Adressen ab Version V5.0

Zugriff auf Datenbereiche in DBs anstelle des direkten Zugriffs auf MODBUS-Adressen ab Version V5.0

Ab der Anweisungs-Version V5.0 von MB_SERVER und ab den Firmware-Versionen V2.5 (S7-1500-CPUs) bzw. V4.2 (S7-1200-CPUs) können Sie anstelle des direkten Zugriffs auf Prozessabbilder und Holding-Register auf Datenbereiche in DBs zugreifen. Dabei muss für den DB das Attribut "Optimierter Bausteinzugriff" deaktiviert sein und er darf nicht ausschließlich im Ladespeicher liegen.

Wenn eine MODBUS-Anforderung eintrifft und Sie für den MODBUS-Datentyp des zugehörigen Funktionscodes keinen Datenbereich definiert haben, wird die Anforderung wie in den bisherigen Anweisungs-Versionen behandelt, d. h. es erfolgt ein direkter Zugriff auf Prozessabbilder und Holding-Register.

Wenn Sie hingegen für den MODBUS-Datentyp des Funktionscode einen Datenbereich definiert haben, liest die Anweisung MB_SERVER aus diesem Datenbereich oder sie schreibt in ihn. Ob gelesen oder geschrieben wird, ist vom Auftragstyp abhängig.

Eine einzelne MODBUS-Anforderung kann immer nur von einem Datenbereich lesen oder in einen Datenbereich schreiben. Wenn Sie also z. B. Holding-Register lesen wollen, die sich über mehrere Datenbereiche erstrecken, benötigen Sie dafür mehrere MODBUS-Anforderungen.

Regeln zur Definition der Datenbereiche

Sie können bis zu acht Datenbereiche in unterschiedlichen DBs definieren; jeder DB darf nur einen Datenbereich enthalten. Eine einzelne MODBUS-Anforderung kann nur von genau einem Datenbereich lesen oder in genau einen Datenbereich schreiben. Dabei entspricht jeder Datenbereich einem MODBUS-Adressbereich. Die Definition erfolgt in der statischen Variable Data_Area_Array des Instanz-DB; Data_Area_Array ist ein Feld, das aus acht Elementen besteht.

Wenn Sie weniger als acht Datenbereiche verwenden wollen, müssen die gewünschten Datenbereiche lückenlos hintereinander liegen. Der erste Leereintrag in den Datenbereichen beendet die Datenbereich-Suche bei der Bearbeitung. Wenn Sie also z. B. die Feldelemente 1, 2, 4 und 5 definiert haben, werden nur die Feldelemente 1 und 2 erkannt, da das Feldelement 3 leer ist.

Das Feld Data_Area_Array besteht aus 8 Elementen: Data_Area_Array[1] … Data_Area_Array[8]

Jedes Feldelement Data_Area_Array[x], 1 <= x <= 8, ist ein UDT vom Typ MB_DataArea und hat folgenden Aufbau:

Parameter	Datentyp	Bedeutung
data_type	UInt	Kennung für den MODBUS-Datentyp, der auf diesen Datenbereich abgebildet wird: • 0: Kennung für ein leeres Feldelement bzw. für einen nicht verwendeten Datenbereich. In diesem Fall sind die Werte von db, start und length irrelevant. • 1: Prozessabbild der Ausgänge (verwendet bei den Funktionscodes 1, 5 und 15) • 2: Prozessabbild der Eingänge (verwendet beim Funktionscode 2) • 3: Holding-Register (verwendet bei den Funktionscodes 3, 6 und 16) • 4: Input-Register (verwendet beim Funktionscode 4) Hinweis: Wenn Sie für einen MODBUS-Datentyp einen Datenbereich definiert haben, kann die Anweisung MB_SERVER auf diesen MODBUS-Datentyp nicht mehr direkt zugreifen. Wenn die Adresse einer MODBUS-Anforderung für einen solchen Datentyp keinem definierten Datenbereich entspricht, wird in STATUS der Wert W#16#8383 zurückgeliefert.
db	UInt	Nummer des Datenbausteins, in den die nachfolgend definierten MODBUS-Register oder Bits abgebildet werden. Die DB-Nummer muss in den Datenbereichen eindeutig sein. Dieselbe DB-Nummer darf nicht in mehreren Datenbereichen definiert werden. Der DB muss ein DB mit Standardzugriff sein und er darf nicht ausschließlich im Ladespeicher liegen. Datenbereiche beginnen stets bei der Byte-Adresse 0 des DB. Zulässige Werte: 1 bis 60999

Parameter	Datentyp	Bedeutung
start	UInt	Erste MODBUS-Adresse, die in den Datenbaustein ab Adresse 0.0 abgebildet wird. Zulässige Werte: 0 bis 65535
length	UInt	Anzahl der Bits (für die Werte 1 und 2 von data_type) bzw. Anzahl der Register (für die Werte 3 und 4 von data_type). Die MODBUS-Adressbereiche ein und desselben MODBUS-Datentyps dürfen sich nicht überlappen. Zulässige Werte: 1 bis 65535

Beispiele zur Definition von Datenbereichen

- Erstes Beispiel: data_type = 3, db = 1, start = 10, length = 6
  Die Holding Register (data_type = 3) werden im Datenbaustein 1 (db = 1) abgebildet. Die Modbus-Adresse 10 (start = 10) befindet sich an Datenwort 0. Die letzte gültige Modbus-Adresse 15 (length = 6) befindet sich an Datenwort 5.
- Zweites Beispiel: data_type = 2, db = 15, start = 1700, length = 112
  Die Eingänge (data_type = 2) werden im Datenbaustein 15 (db = 15) abgebildet. Die Modbus-Adresse 1700 (start = 1700) befindet sich an Datenwort 0. Die letzte gültige Modbus-Adresse 1811 (length = 112) befindet sich an Datenwort 111.

Beschränkung des Lesezugriffs auf Prozessabbilder ab Version V5.0

Beschränkung des Lesezugriffs auf Prozessabbilder

Ab der Anweisungs-Version V5.0 von MB_SERVER können Sie im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge je einen Bereich definieren, auf den remote MODBUS-Geräte lesend zugreifen dürfen. Ein Lesezugriff remoter MODBUS-Geräte auf Adressen außerhalb dieser Prozessabbild-Bereiche ist dann nicht mehr möglich.

Hinweis

Beschränkung des Schreibzugriffs auf Prozessabbilder

Die Möglichkeit, den Schreibzugriff auf das Prozessabbild der Ausgänge auf einen bestimmten Bereich zu beschränken, ist ab der Anweisungs-Version V4.2 möglich.

Definition der Lesebereiche in den Prozessabbildern

Die Definition der Lesebereiche in den Prozessabbildern erfolgt in den folgenden statischen Variablen des Instanz-DB:

- QB_Read_Start: Adresse des ersten Bytes im Prozessabbild der Ausgänge, das von einem remoten MODBUS-Gerät gelesen werden darf (gilt für den Funktionscode 1)
- QB_Read_Count: Anzahl der Bytes im Prozessabbild der Ausgänge, die von einem remoten MODBUS-Gerät gelesen werden dürfen (gilt für den Funktionscode 1)

- IB_Read_Start: Adresse des ersten Bytes im Prozessabbild der Eingänge, das von einem remoten MODBUS-Gerät gelesen werden darf (gilt für die Funktionscodes 2 und 4)
- IB_Read_Count: Anzahl der Bytes im Prozessabbild der Eingänge, die von einem remoten MODBUS-Gerät gelesen werden dürfen (gilt für die Funktionscodes 2 und 4)

Statische Variablen im Instanz-DB zur Definition der Schreib- und der Lesebereiche in den Prozessabbildern

Die folgende Tabelle beschreibt die oben aufgeführten statischen Variablen im Instanz-DB der Anweisung MB_SERVER, mit denen Sie die Lesebereiche in den Prozessabbildern definieren.

Der Vollständigkeit halber sind auch diejenigen statischen Variablen angegeben, mit denen Sie ab Version V4.2 die Schreibbereiche in den Prozessabbildern definieren (QB_Start und QB_Count).

Variable	Datentyp	Startwert
QB_Start	UInt	0
QB_Count	UInt	65535
QB_Read_Start	UInt	0
QB_Read_Count	UInt	65535
IB_Read_Start	UInt	0
IB_Read_Count	UInt	65535

Die statischen Variablen NDR_immediate und DR_immediate

NDR_immediate und DR_immediate

Die statischen Variablen NDR_immediate und DR_immediate sind ab der Anweisungs-Version V5.0 im Instanz-Datenbaustein der Anweisung MB_SERVER enthalten.

Variable	Datentyp	Startwert	Bedeutung
NDR_immediate	BOOL	FALSE	Identische Bedeutung wie der Parameter NDR (New Data Ready). NDR_immediate wird im selben Aufruf von MB_SERVER aktualisiert, in dem eine MODBUS-TCP-Schreibanforderung bearbeitet wird.
DR_immediate	BOOL	FALSE	Identische Bedeutung wie der Parameter DR (Data Read). DR_immediate wird im selben Aufruf von MB_SERVER aktualisiert, in dem eine MODBUS-TCP-Leseanforderung bearbeitet wird.

Unterschiede zwischen den Versionen <= V3.x und >= V4.0 der Modbus-TCP-Bibliothek

Unterschiede zwischen den Versionen <= V3.x und >= V4.0 der Modbus-Anweisungen

Bei den MODBUS Anweisungen MB_SERVER / MB_CLIENT bestehen folgende Unterschiede zwischen den Versionen:

- Adressparameter
  - Bei den Versionen <= V3.x wurden die Adressdaten für den Modbus TCP Server über die Eingangsparameter "IP_x" angegeben.
  - Die Versionen >= V4.0 verwenden hierzu Systemdatentypen TCON_IP_V4 und TCON_Configured am Eingangsparameter CONNECT.
- Tritt bei der Bearbeitung ein Fehler auf, stehen ihnen ab der Version 4.0 der Modbus-Anweisungen mehr STATUS-Meldungen zur Verfügung, die entsprechend ausgewertet werden können.

Detaillierte Informationen entnehmen Sie den Anweisungsbeschreibungen zu MB_CLIENT (Seite 4291) und MB_SERVER (Seite 4306).

Hinweis

Wechsel der Modbus-TCP-Bibliothek von einer Version <= V3.x zu einer Version >= V4.0

Beim Austausch der Modbus-TCP-Bibliothek von einer Version <= V3.x zu einer Version >= V4.0 müssen Sie auch die Open-User-Communication-Bibliothek austauschen. Überprüfen Sie danach alle für Ihr Programm relevanten Anweisungen.

Redundante MODBUS (TCP)-Kommunikation für Bibliotheks-Versionen ab V5.0

MB_RED_CLIENT: Als MODBUS-TCP-Client redundant über PROFINET kommunizieren

Beschreibung MB_RED_CLIENT

Beschreibung

Mit dieser Anweisung ist es möglich, eine Kopplung zwischen einer 1200/1500-CPU und einem Gerät, welches das Protokoll Modbus/TCP unterstützt, aufzubauen.

Die Anweisung "MB_RED_CLIENT" kommuniziert als Modbus/TCP-Client über die PROFINET-Verbindung. Über die Anweisung "MB_RED_CLIENT" bauen Sie eine redundante Verbindung zwischen dem Client und dem Server auf, senden Modbus-Anfragen, empfangen Antworten und steuern den Verbindungsabbau des Modbus/TCP-Clients.

Die Anweisung "MB_RED_CLIENT" kann in folgenden CPUen eingesetzt werden:

	"MB_RED_CLIENT" V1.0	"MB_RED_CLIENT" V1.1
S7-1200	ab FW V4.2	ab FW V4.2
S7-1500	ab FW V2.5	ab FW V2.5
S7-1500R	--	ab FW V2.6
S7-1500H	--	ab FW V2.6

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

Mehrfache Client-Verbindungen

Die CPUs sind in der Lage mehrere Modbus/TCP-Clientverbindungen zu bearbeiten. Die maximale Anzahl der Verbindungen ist von der verwendeten CPU abhängig und kann den technischen Daten der CPU entnommen werden. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus/TCP-Clients und -Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Bei einzelnen Client-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_RED_CLIENT"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jede "MB_RED_CLIENT"-Verbindung muss eine eindeutige IP-Adresse des Servers spezifiziert sein.
- Jede "MB_RED_CLIENT"-Verbindung benötigt eine eindeutige Verbindungs-ID. Die Verbindungs-IDs müssen CPU-weit eindeutig sein.

Informationen bezüglich des Redundanz-Aufbaus erhalten Sie hier: Arbeitsweise und Redundanzverhalten (Seite 4323).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_RED_CLIENT":

Parameter	Deklaration	Datentyp	Beschreibung
REG_KEY (Seite 4331)	Input	STRING[17]	Freischaltcode für die Lizenzierung Die Anweisung "MB_RED_CLIENT" muss auf jeder CPU einzeln lizenziert werden.
USE_ALL_CONN	Input	BOOL	Festlegung, über wie viele der projektierten Verbindungen das Telegramm gesendet werden soll • 0: Telegramm über 1 Verbindung senden, Umschaltung auf die nächste Verbindung nur im Fehlerfall • 1: Telegramm über alle projektierten Verbindungen senden

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Modbus-Anfrage an den Modbus/TCP-Server Der Parameter REQ ist pegelgesteuert. Das heißt, solange der Eingang gesetzt ist (REQ = TRUE), sendet die Anweisung Kommunikationsanfragen. Ist die Verbindung noch nicht aufgebaut, wird diese aufgebaut und anschließend sofort das Modbus-Telegramm gesendet. Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. Wird während einer laufenden Modbus-Anfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.
DISCONNECT	Input	BOOL	Über den Parameter steuern Sie den Verbindungsaufbau- und -abbau zu dem Modbus-Server: • 0: Kommunikationsverbindung zu dem am Parameter CONNECT konfigurierten Verbindungspartner herstellen (siehe Parameter CONNECT). • 1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS_x der Wert 0003 ausgegeben.
MB_MODE (Seite 4334)	Input	USINT	Auswahl des Modus der Modbus-Anfrage (Lesen, Schreiben oder Diagnose) bzw. direkte Auswahl einer Modbus-Funktion
MB_DATA_ADDR (Seite 4334)	Input	UDINT	Modbus-Adresse abhängig von MB_MODE
MB_DATA_LEN	Input	UINT	Datenlänge: Anzahl der Bits oder Register für den Datenzugriff.
MB_DATA_PTR (Seite 4334)	InOut	VARIANT	Zeiger auf einen Datenpuffer für die vom Modbus-Server zu empfangenden oder zum Modbus-Server zu sendenden Daten.
LICENSED (Seite 4331)	Output	BOOL	• 0: Anweisung ist nicht lizenziert • 1: Anweisung ist lizenziert
IDENT_CODE (Seite 4331)	Output	STRING[18]	Identifikation für die Lizenzierung. Fordern Sie mit diesem String den Freischaltcode REG_KEY an.
DONE	Output	BOOL	Das Bit am Ausgangsparameter DONE wird auf "1" gesetzt, sobald der aktivierte Modbus-Auftrag auf mindestens 1 Verbindung ohne Fehler ausgeführt wurde.
BUSY	Output	BOOL	• 0: Keine Modbus-Anfrage in Bearbeitung • 1: Modbus-Anfrage wird bearbeitet Der Ausgangsparameter BUSY wird beim Verbindungsaufbau und -abbau nicht gesetzt.
ERROR	Output	BOOL	• 0: Kein Fehler • 1: Der aktivierte Modbusauftrag konnte auf keiner der projektierten Verbindungen erfolgreich übertragen werden. Die Fehlerursache wird über den Parameter STATUS_x angezeigt.
STATUS_0A (Seite 4337)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 0A.
STATUS_1A (Seite 4337)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 1A.
STATUS_0B (Seite 4337)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 0B.

Parameter	Dekla-ration	Datentyp	Beschreibung
STATUS_1B (Seite 4337)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 1B.
RED_ERR_S7 (Seite 4337)	Output	BOOL	• 0: kein Redundanzfehler auf der SIMATIC-Seite • 1: Redundanzfehler auf der SIMATIC-Seite
RED_ERR_DEV (Seite 4337)	Output	BOOL	• 0: kein Redundanzfehler auf der Seite des Koppelpartners • 1: Redundanzfehler auf der Seite des Koppelpartners
TOT_COM_ERR (Seite 4337)	Output	BOOL	• 0: mindestens 1 projektierte Verbindung ist aufgebaut • 1: vollständiger Kommunikationsverlust, alle projektierten Verbindungen sind abgebaut

Hinweis

Konsistente Input-Daten während eines "MB_RED_CLIENT"-Aufrufs

Sobald eine Modbus-Client Anweisung aufgerufen wird, werden die Werte der Eingangsparameter intern gespeichert. Sie dürfen während der Bearbeitung des Telegramms nicht verändert werden.

Arbeitsweise und Redundanzverhalten

Beschreibung

Die Kommunikationsteilnehmer können standalone oder redundant aufgebaut werden. Ist einer der beiden Teilnehmer standalone aufgebaut, spricht man von einseitiger Redundanz. Sind beide Partner redundant aufgebaut, bezeichnet man es als beidseitige Redundanz.

Portnummer für Client und Server

Der Modbus Client verwendet eine Portnummer ab 2000. Der Modbus Server wird normalerweise über die Portnummer 502 angesprochen.

Einseitige Redundanz

Beschreibung

Für jede Verbindung zwischen den Kommunikationspartnern muss je eine Verbindung projektiert werden. Die Verbindungspunkte der **SIMATIC S7** haben die Bezeichnungen **0** und **1**, die Verbindungspunkte des **Kommunikationspartners** haben die Bezeichnungen **A** und **B**.

Die R- bzw. H-CPU 1 bezeichnet den Verbindungspunkt 0, die R- bzw. H-CPU 2 bezeichnet den Verbindungspunkt 1.

Projektierung

Ist die S7 redundant aufgebaut, wird eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt A des Koppelpartners und eine Verbindung vom Verbindungspunkt 1 der S7 zum Knotenpunkt A des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **A** => Verbindung **1A**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

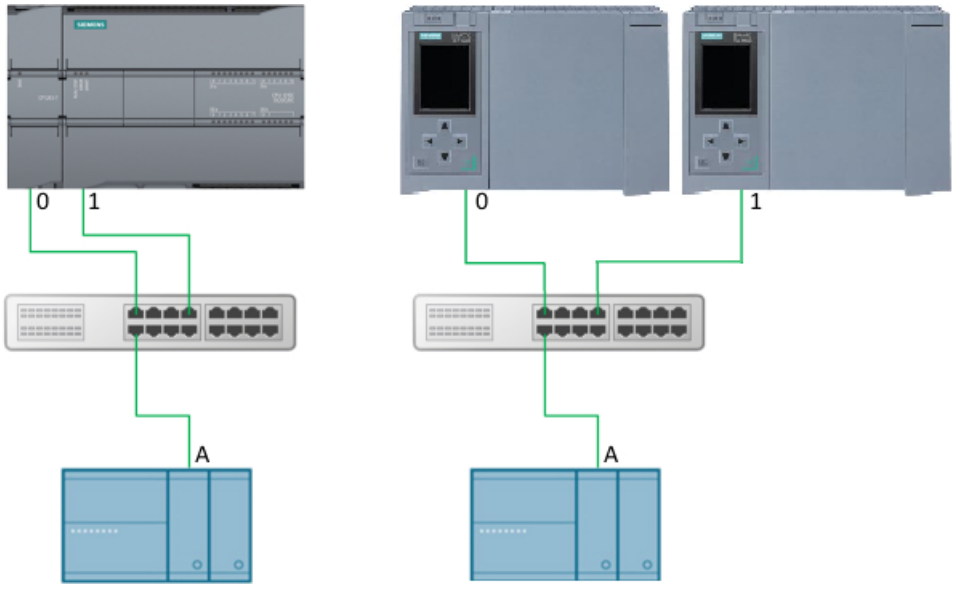

Bild 4-9 Modbus/TCP: Einseitige Redundanz S7

Ist die S7 standalone und der Koppelpartner redundant aufgebaut, wird eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt A des Koppelpartners und eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt B des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **B** => Verbindung **0B**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

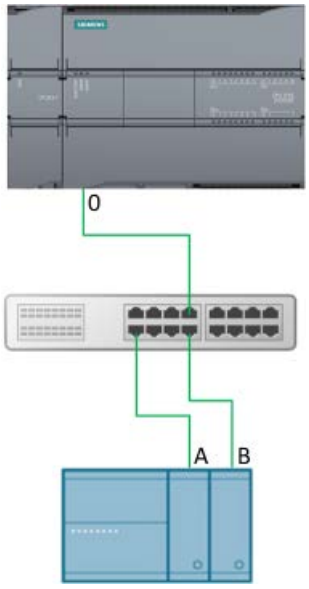

Bild 4-10 Modbus/TCP: Einseitige Redundanz Partner

Beidseitige Redundanz

Beschreibung

Für jede Verbindung zwischen den Kommunikationspartnern muss je 1 Verbindung projektiert werden. Die Verbindungspunkte der **SIMATIC S7** haben die Bezeichnungen **0** und **1**, die Verbindungspunkte des **Kommunikationspartners** haben die Bezeichnungen **A** und **B**.

Die R- bzw. H-CPU 1 bezeichnet den Verbindungspunkt 0, die R- bzw. H-CPU 2 bezeichnet den Verbindungspunkt 1.

Projektierung

Bei beidseitiger Redundanz werden zwei Verbindungen vom Verbindungspunkt 0 und zwei Verbindungen vom Verbindungspunkt 1 der S7 zu den Knotenpunkten A und B des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **A** => Verbindung **1A**
- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **B** => Verbindung **0B**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **B** => Verbindung **1B**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

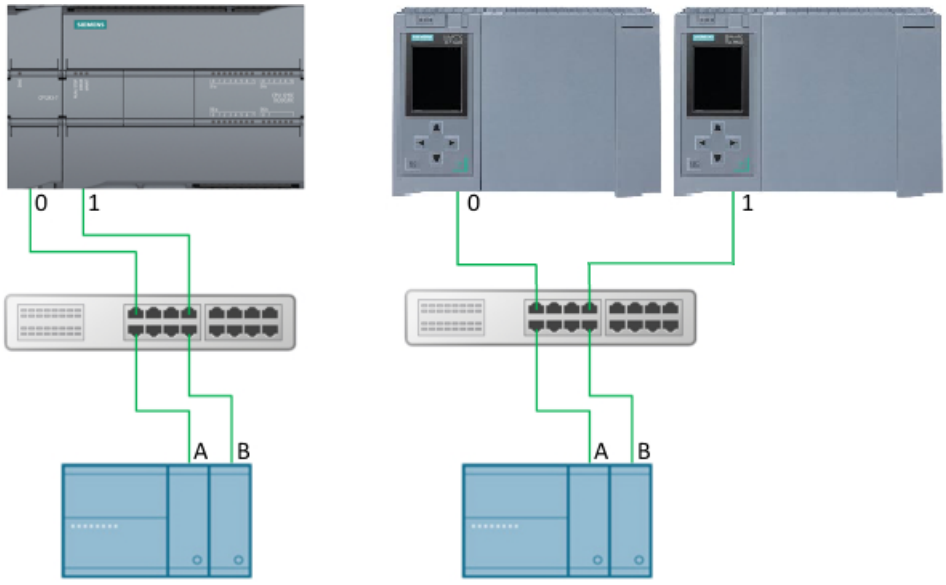

Bild 4-11 Modbus/TCP: Beidseitige Redundanz

Telegrammbearbeitung

Allgemein

Die Telegramme können wahlweise über eine oder über alle projektierten Verbindungen gesendet werden.

Telegramme über 1 Verbindung senden

Mit der Einstellung USE_ALL_CONN = FALSE wird das MODBUS Telegramm über eine - die derzeit aktive - Verbindung gesendet. Im Falle einer Zeitüberschreitung (keine Antwort vom Server) oder eines Verbindungsfehlers wird versucht, das Telegramm nacheinander über die anderen (maximal 4) projektierten Verbindungen zu senden. Die Reihenfolge dabei ist 0A, 1A, 0B und 1B. Konnte ein Telegramm erfolgreich über eine Verbindung übertragen werden, wird diese Verbindung als „aktiv“ markiert und der weitere Telegrammverkehr wird über diese Verbindung abgewickelt. Im Falle eines Verbindungsfehlers der aktiven Verbindung wird erneut versucht, nacheinander über alle projektierten Verbindungen das Telegramm zu senden. Schlagen alle Sendeversuche fehl, werden ERROR und STATUS_x entsprechend gesetzt.

Wird ein Antworttelegramm empfangen, wird eine Plausibilitätsprüfung durchgeführt. Verläuft diese positiv, werden die erforderlichen Aktionen durchgeführt und der Auftrag wird ohne Fehler beendet, der Ausgang DONE wird gesetzt. Wurden bei der Prüfung Fehler erkannt, wird der Auftrag mit Fehler beendet, das Bit ERROR wird gesetzt und eine Fehlernummer an STATUS_x angezeigt. In diesem Fall erfolgt kein erneuter Sendeversuch des Telegramms auf der nächsten projektierten Verbindung. Eine Umschaltung auf die anderen projektierten Verbindungen findet nur statt, wenn ein Verbindungsfehler erkannt oder keine Antwort empfangen wurde.

Telegramme über alle Verbindungen senden

Mit der Einstellung USE_ALL_CONN = TRUE wird das MODBUS Telegramm über alle projektierten, aufgebauten Verbindungen gesendet. Nach dem Empfang eines

Antworttelegramms auf einer der Verbindungen wird eine Plausibilitätsprüfung durchgeführt. Verläuft diese positiv, werden die erforderlichen Aktionen durchgeführt. Wurde auf mindestens eine Verbindung ein gültiges Antworttelegramm empfangen, wird der Ausgang DONE gesetzt.

Redundanz-Ausgänge RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR

Beschreibung

Anhand der Zustände der Status-Ausgänge werden die Redundanz-Bits RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR gesetzt.

Anzeige der Alarm-Bits für beidseitigen Redundanz-Aufbau

Anzahl der fehlerhaften Verbindungen	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	okay	okay	okay	FALSE	FALSE	FALSE
1	okay	okay	okay	Error	FALSE	FALSE	FALSE
	okay	okay	Error	okay	FALSE	FALSE	FALSE
	okay	Error	okay	okay	FALSE	FALSE	FALSE
	Error	okay	okay	okay	FALSE	FALSE	FALSE
2	okay	okay	Error	Error	TRUE	FALSE	FALSE
	okay	Error	okay	Error	FALSE	TRUE	FALSE
	Error	okay	okay	Error	FALSE	FALSE	FALSE
	okay	Error	Error	okay	FALSE	FALSE	FALSE
	Error	okay	Error	okay	FALSE	TRUE	FALSE
	Error	Error	okay	okay	TRUE	FALSE	FALSE
3	Error	Error	Error	okay	TRUE	TRUE	FALSE
	Error	Error	okay	Error	TRUE	TRUE	FALSE
	Error	okay	Error	Error	TRUE	TRUE	FALSE
	okay	Error	Error	Error	TRUE	TRUE	FALSE
4	Error	Error	Error	Error	TRUE	TRUE	TRUE

Anzeige der Alarm-Bits für einseitigen Redundanz-Aufbau

Anzahl der fehlerhaften Verbindungen	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	0AFF	okay	0AFF	FALSE	FALSE	FALSE
1	okay	0AFF	Error	0AFF	TRUE	TRUE	FALSE
	Error	0AFF	okay	0AFF	TRUE	TRUE	FALSE
2	Error	0AFF	Error	0AFF	TRUE	TRUE	TRUE

Parametrierung

Beschreibung

Die Anweisung "MB_RED_CLIENT" **V1.0** und **V1.1** kann für S7-1200 und für S7-1500 eingesetzt werden.

S7-1200

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisert. Hierbei werden die Verbindungen mit Hilfe der Struktur TCON_IP_V4 parametriert und aufgebaut.

S7-1500

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisiert. Hierbei sind sowohl parametrierte Verbindungen über die Struktur TCON_IP_V4 als auch konfigurierte Verbindungen über die Struktur TCON_Configured möglich.

Die Anweisung "MB_RED_CLIENT" **V1.1** kann für S7-1200R und für S7-1500H eingesetzt werden.

S7-1500R und S7-1500H

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisert. Hierbei werden die Verbindungen mit Hilfe der Struktur TCON_IP_V4 parametriert und aufgebaut.

Konfiguration von "MB_RED_CLIENT"

Mit Hilfe des Konfigurations-Dialoges der Anweisung "MB_RED_CLIENT" werden folgende Einstellungen vorgenommen:

- Verbindungsparameter für die Verbindungen 0A, 1A, 0B und 1B (Redundanzaufbau siehe Arbeitsweise und Redundanzverhalten (Seite 4323))
- Interne Parameter (optional)

Der Konfigurationsdialog kann über die Anweisung oder über die Technologieobjekte geöffnet werden.

Verbindungsparameter

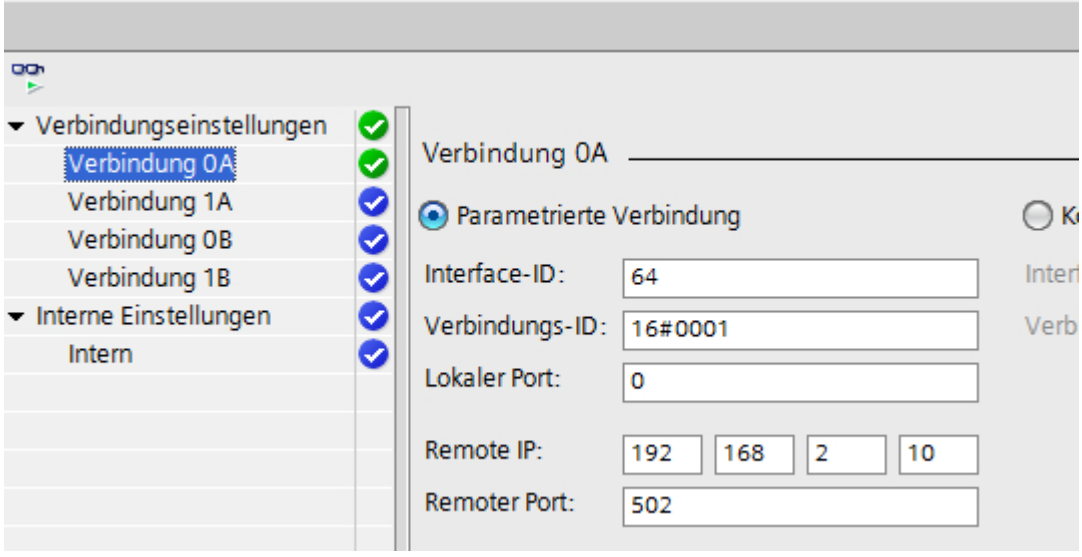

Bild 4-12 Modbus/TCP: parametrierte Client-Verbindung

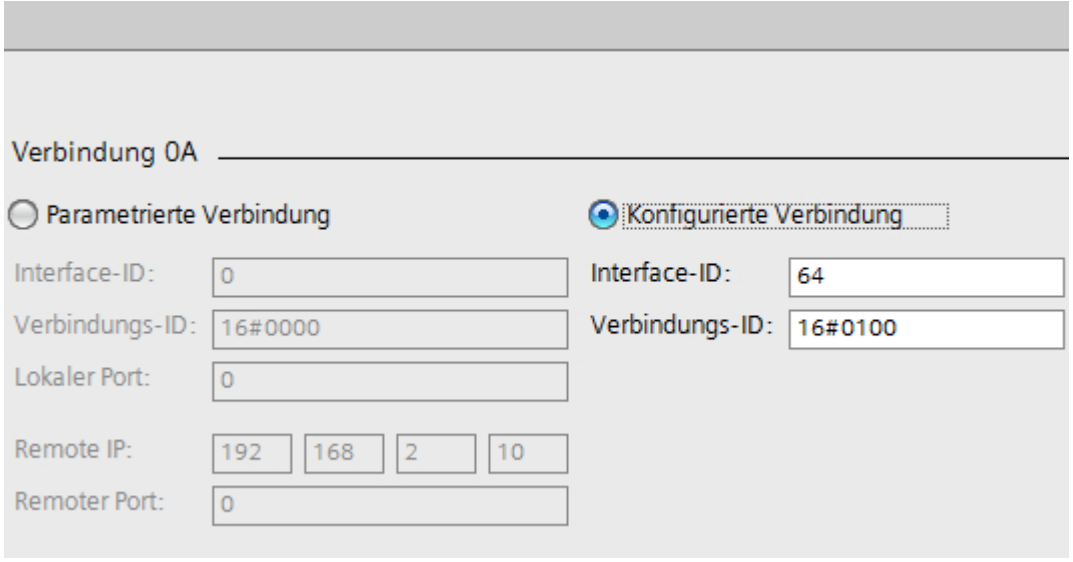

Bild 4-13 Modbus/TCP: konfigurierte Client-Verbindung

Variable	Startwert	Beschreibung
Parametrierte Verbindungen		
Interface ID	64	HW-Identifier der verwendeten PN-Schnittstelle
Connection ID	16#0000	Verbindungs-IDs für die verwendeten Verbindungen Diese Verbindungs-IDs müssen CPU-weit eindeutig sein.

Variable	Startwert	Beschreibung
Local port	0	Lokale Portnummer des Clients. Standardmässig wird keine Portnummer für den Client eingetragen.
Remote IP	0.0.0.0	Remote IP-Adresse des Servers
Remote port	502	Remote Portnummer des Servers Der Standard-Port für Modbus/TCP-Server ist 502.
Konfigurierte Verbindungen		
Interface ID	64	HW-Identifier der verwendeten PN-Schnittstelle
Connection ID	16#0000	Verbindungs-IDs für die vewendeten Verbindungen Diese Verbindungen werden in der Netzsicht konfiguriert.

Interne Parameter (optional)

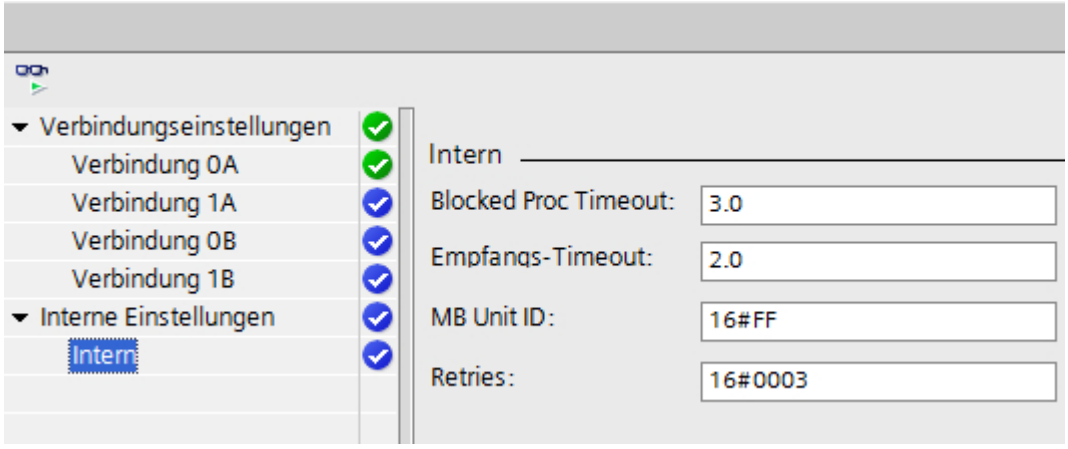

Variable	Datentyp	Start-wert	Beschreibung
Blocked Proc Time	REAL	3.0	Wartezeit in Sekunden, bevor bei einer blockierten Modbus-Instanz die statische Variable ACTIVE rückgesetzt wird. Dies kann zum Beispiel auftreten, wenn bei einer ausgegebenen Client-Anfrage die Ausführung der Client-Funktion abgebrochen wird, bevor die Anfrage vollständig ausgeführt wurde. Die Wartezeit muss zwischen 0,5 s und 55 s liegen.
Receive time-out	REAL	2.0	Zeitintervall in Sekunden, in dem die Anweisung "MB_RED_CLIENT" auf eine Antwort des Servers wartet. Es muss zwischen 0,5 s und 55 s liegen.
MB_Unit_ID	BYTE	255	Modbus-Gerätekennung: Ein Modbus TCP-Server wird über seine IP-Adresse angesprochen. Deshalb wird der Parameter MB_UNIT_ID bei der Modbus TCP-Adressierung nicht verwendet. Der Parameter MB_UNIT_ID entspricht dem Feld der Slaveadresse beim Modbus RTU-Protokoll. Wenn ein Modbus/TCP-Server als Gateway zu einem Modbus RTU-Protokoll verwendet wird, kann das Slavegerät im seriellen Netzwerk über MB_UNIT_ID identifiziert werden. Der Parameter MB_UNIT_ID würde in diesem Fall die Anforderung an die richtige Modbus RTU-Slaveadresse weiterleiten. Beachten Sie bitte, dass einige Modbus/TCP-Geräte den Parameter MB_UNIT_ID möglicherweise für die Initialisierung innerhalb eines eingeschränkten Wertebereichs benötigen.
Retries	WORD	3	Anzahl der Sende-Versuche, die die Anweisung "MB_RED_CLIENT" unternimmt, bevor sie den Fehler W#16#80C8 zurückgibt.

Hinweis

Variable MB_Transaction_ID

Wenn die Transaktions-ID in der Antwort des Modbus-TCP-Servers nicht mit der Transaktions-ID des Auftrags von "MB_RED_CLIENT" übereinstimmt, wartet die Anweisung "MB_RED_CLIENT" für die Zeitdauer RCV_TIMEOUT * RETRIES auf die Antwort des Modbus-TCP-Servers mit der korrekten Transaktions-ID, danach gibt sie den Fehler W#16#80C8 zurück.

Lizenzierung

Die Anweisung "MB_RED_CLIENT" ist kostenpflichtig und muss auf jeder CPU einzeln lizenziert werden. Die Lizenzierung erfolgt in 2 Schritten:

- dem Auslesen des IDENT_CODE und
- der Eingabe des Freischaltcodes REG_KEY.

Hinweis

S7-1500R und S7-1500H

Bei einer S7-R/H-Station wird nur die CPU1 lizenziert. Daher ist ein Tausch der CPU1 nach der Lizenzierung nicht mehr möglich.

Vorgehensweise

Auslesen des IDENT_CODE

Für das Auslesen des IDENT_CODE gehen Sie wie folgt vor:

1. Parametrieren Sie die Anweisung "MB_RED_CLIENT" Ihren Anforderungen entsprechend in einem zyklischen OB. Laden Sie das Programm in die CPU und setzen Sie diese in RUN.
2. Öffnen Sie den Instanz-DB der Modbus-Anweisung und klicken Sie auf die Schaltfläche „Alle beobachten“ .

3. Am Ausgang IDENT_CODE wird eine 18-stellige Zeichenfolge angezeigt.

	Name	Data type	Start value	Monitor value
1	Input			
2	REG_KEY	String[17]	''	''
3	USE_ALL_CONN	Bool	false	FALSE
4	REQ	Bool	false	FALSE
5	DISCONNECT	Bool	false	FALSE
6	MB_MODE	USInt	0	0
7	MB_DATA_ADDR	UDInt	0	0
8	MB_DATA_LEN	UInt	0	0
9	Output			
10	LICENSED	Bool	false	FALSE
11	IDENT_CODE	String[18]	''	'RTPCFIGDCDIIHJHAH4'
12	DONE	Bool	false	FALSE
13	BUSY	Bool	false	FALSE
14	ERROR	Bool	false	FALSE

4. Kopieren Sie diesen String per Copy/Paste aus dem Datenbaustein und fügen ihn in das Formular ein, welches Ihnen nach der Bestellung des Produktes per E-Mail zugesandt wurde bzw. auf der CD enthalten ist.
5. Senden Sie das Formular über einen Service Request an den Customer Support (https://support.industry.siemens.com/my/ww/de/requests/#createRequest). Sie erhalten daraufhin den Freischaltcode für ihre CPU.

Eingabe des Freischaltcodes REG_KEY

Die Angabe des Freischaltcodes REG_KEY muss an jeder "MB_RED_CLIENT"-Anweisung erfolgen. Der REG_KEY sollte in einem Global-Datenbaustein gespeichert werden, über den alle "MB_RED_CLIENT"-Anweisungen den notwendigen Freischaltcode erhalten.

Für die Eingabe des Freischaltcodes REG_KEY gehen Sie wie folgt vor:

- Fügen Sie mit „Neuen Baustein hinzufügen…“ einen neuen Global-Datenbaustein mit einem eindeutigen symbolischen Namen, z.B. „Lizenz_DB“, ein.
- Legen Sie in diesem Baustein einen Parameter REG_KEY mit dem Datentyp STRING[17] an.

Lizenz_DB

	Name	Datentyp	Offset	Startwert
1	Static			
2	REG_KEY	String[17]	0.0	''

Bild 4-14 REG_KEY in DB

- Kopieren Sie den übermittelten 17-stelligen Freischaltcode per Copy/Paste in die Spalte ‚Startwert'.
- Geben Sie im zyklischen OB am Parameter REG_KEY der "MB_RED_CLIENT"-Anweisung den Namen des Lizenz-DBs und den Namen des Strings an, z.B. Lizenz_DB.REG_KEY.
- Laden Sie die geänderten Bausteine in die CPU. Die Eingabe des Freischaltcodes kann zur Laufzeit erfolgen, ein Wechsel von STOP -> RUN ist nicht erforderlich.

Die Modbus/TCP-Kommunikation über die Anweisung "MB_RED_CLIENT" ist nun für diese CPU lizenziert, das Ausgangsbit LICENSED ist TRUE.

Fehlende oder fehlerhafte Lizenzierung

Ist kein oder ein falscher Freischaltcode eingetragen, blinkt die ERROR-LED der CPU. Zusätzlich wird bei S7-1200 und S7-1500 zyklisch ein Eintrag in den Diagnosepuffer bezüglich der fehlenden Lizenz vorgenommen. Bei R- und H-Stationen wird nur der Bereichslängenfehler angezeigt.

Ereignisse

☑ CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit

Nr.	Datum und Uhrzeit	Ereignis
1	06.06.2018 12:37:26.424	Bereichslängenfehler in FB 1086
2	06.06.2018 12:37:26.424	Kein gültiger License Key für das Funktionspaket
3	06.06.2018 12:37:20.422	Bereichslängenfehler in FB 1086
4	06.06.2018 12:37:20.422	Kein gültiger License Key für das Funktionspaket
5	06.06.2018 12:37:14.420	Bereichslängenfehler in FB 1086
6	06.06.2018 12:37:14.420	Kein gültiger License Key für das Funktionspaket
7	06.06.2018 12:37:08.418	Bereichslängenfehler in FB 1086
8	06.06.2018 12:37:08.418	Kein gültiger License Key für das Funktionspaket
9	06.06.2018 12:37:02.416	Bereichslängenfehler in FB 1086

Einfrieren aufheben

Bei einem fehlenden oder falschen Freischaltcode wird die Modbus/TCP-Kommunikation bearbeitet, allerdings wird am Ausgang STATUS_x stets W#16#0A90 „Keine gültige Lizenz vorhanden“ angezeigt. Das Ausgangsbit LICENSED ist FALSE.

⚠ WARNUNG

Falls in der S7-1500 (R/H) der OB121 in der Steuerung fehlt, wird die CPU in den STOP-Zustand gesetzt.

Eingangsparameter MB_MODE, MB_DATA_ADDR, MB_DATA_LEN und MB_DATA_PTR

Beschreibung

Die Kombination der Parameter MB_MODE, MB_DATA_ADDR und MB_DATA_LEN legt den Modbus-Funktionscode fest, der in der aktuellen Modbus-Nachricht verwendet wird:

- **MB_MODE** enthält die Information, ob gelesen oder geschrieben werden soll.
  Lesen: MB_MODE = 0, 101, 102, 103 und 104
  Schreiben: MB_MODE = 1, 2, 105, 106, 115 und 116 (Hinweis: Bei MB_MODE = 2 wird nicht zwischen Modbus-Funktion 15 und 05 bzw. zwischen Modbus-Funktion 16 und 06 unterschieden.)
- **MB_DATA_ADDR** enthält die Information, was gelesen bzw. geschrieben werden soll, sowie eine Adressinformation, aus der die Anweisung "MB_RED_CLIENT" die remote Adresse berechnet.
- **MB_DATA_LEN** enthält die Anzahl der zu lesenden / schreibenden Werte.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Eingangsparametern MB_MODE, MB_DATA_ADDR, MB_DATA_LEN der Anweisung "MB_RED_CLIENT" und der zugehörigen Modbus-Funktion.

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus-Funktion	Funktion und Datentyp
0	1 bis 9.999	1 bis 2.000	01	1 bis 2.000 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	10.001 bis 19.999	1 bis 2.000	02	1 bis 2.000 Eingangs-Bits auf der remoten Adresse 0 bis 9.998 lesen
0	• 40.001 bis 49.999 • 400.001 bis 465.535	1 bis 125	03	• 1 bis 125 Halteregister auf der remoten Adresse 0 bis 9.998 lesen • 1 bis 125 Halteregister auf der remoten Adresse0 bis 65.534 lesen
0	30.001 bis 39.999	1 bis 125	04	1 bis 125 Eingangs-Worte auf der remoten Adresse 0 bis 9.998 lesen
1	1 bis 9.999	1	05	1 Ausgangs-Bit auf der remoten Adresse 0 bis 9.998 schreiben
1	• 40.001 bis 49.999 • 400.001 bis 465.535	1	06	• 1 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 1 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
1	1 bis 9.999	2 bis 1.968	15	2 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben

MB_ MO-DE	MB_DA-TA_ADDR	MB_DA TA_LE N	Mod-bus-Funk tion	Funktion und Datentyp
1	• 40.001 bis 49.999 • 400.001 bis 465.535	2 bis 123	16	• 2 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 2 bis 123 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
2	1 bis 9.999	1 bis 1.968	15	1 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 9.998 schreiben
2	• 40.001 bis 49.999 • 400.001 bis 465.535	1 bis 123	16	• 1 bis 123 Halteregister auf der remoten Adresse 0 bis 9.998 schreiben • 1 bis 123 Halteregister auf der remoten Adresse 0 bis 65.534 schreiben
11	Die Parameter MB_DA-TA_ADDR und MB_DA-TA_LEN werden bei der Ausführung dieser Funk-tion nicht ausgewertet.		11	Status-Wort und Ereigniszähler des Servers lesen: • Das Status-Wort bildet den Bearbeitungszustand ab (0 - nicht in Bearbeitung, 0xFFFF - in Bearbeitung). • Der Ereigniszähler wird inkrementiert, wenn die Modbus-Anfrage erfolgreich ausgeführt wurde. Wenn bei der Ausführung einer Modbus-Funktion ein Fehler aufgetreten ist, wird zwar eine Nachricht vom Server gesendet, der Ereigniszähler jedoch nicht inkrementiert.
80	-	1	08	Überprüfung des Server-Status durch den Diagnosecode 0x0000 (Rückschlei-fentest - der Server sendet die Abfrage zurück): • 1 WORD pro Aufruf
81	-	1	08	Ereigniszähler des Servers über den Diagnosecode 0x000A rücksetzen: • 1 WORD pro Aufruf
101	0 bis 65.535	1 bis 2.000	01	1 bis 2.000 Ausgangs-Bits auf der remoten Adresse 0 bis 65.535 lesen
102	0 bis 65.535	1 bis 2.000	02	1 bis 2.000 Eingangs-Bits auf der remoten Adresse 0 bis 65.535 lesen
103	0 bis 65.535	1 bis 125	03	1 bis 125 Halteregister auf der remoten Adresse 0 bis 65.535 lesen
104	0 bis 65.535	1 bis 125	04	1 bis 125 Eingangs-Worte auf der remoten Adresse 0 bis 65.535 lesen
105	0 bis 65.535	1	05	1 Ausgangs-Bit auf der remoten Adresse 0 bis 65.535 schreiben
106	0 bis 65.535	1	06	1 Halteregister auf der remoten Adresse 0 bis 65.535 schreiben
115	0 bis 65.535	1 bis 1.968	15	1 bis 1.968 Ausgangs-Bits auf der remoten Adresse 0 bis 65.535 schreiben

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus-Funktion	Funktion und Datentyp
116	0 bis 65.535	1 bis 123	16	1 bis 123 Halteregister auf der remoten Adresse 0 bis 65.535 schreiben
3 bis 10, 12 bis 79, 82 bis 100, 107 bis 114, 117 bis 255				reserviert

Beispiel

Variable	Bedeutung
MB_MODE = 1 MB_DATA_ADDR = 1 MB_DATA_LEN = 1	Ab der remoten Adresse 0 wird 1 Ausgangs-Bit mit Funktionscode 5 geschrieben.
MB_MODE = 1 MB_DATA_ADDR = 1 MB_DATA_LEN = 2	Ab der remoten Adresse 0 werden 2 Ausgangs-Bits mit Funktionscode 15 geschrieben.
MB_MODE = 104 MB_DATA_ADDR = 17834 MB_DATA_LEN = 125	Ab der remoten Adresse 17.834 werden 125 Eingangsworte mit Funktionscode 4 gelesen

MB_DATA_PTR

Der Parameter MB_DATA_PTR ist ein Zeiger auf einen Datenpuffer für die vom Modbus-Server zu empfangenden oder zum Modbus-Server zu sendenden Daten. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie für einen Puffer im Speicherbereich (M) einen Zeiger im Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Abhängig vom Speicherbereich, in dem der Datenpuffer liegt, kann MB_DATA_PTR auf unterschiedliche Datenstrukturen verweisen:

- Wenn man einen Global-DB mit optimiertem Zugriff verwendet, kann MB_DATA_PTR auf eine Variable mit elementarem Datentyp oder ein Array von elementaren Datentypen verweisen. Folgende Datentypen werden unterstützt:

Datentyp	Länge in Bits
Bool	1
Byte, SInt, USInt, Char	8
Word, Int, WChar, UInt	16
DWord, DInt, UDInt, Real	32

  Dabei können Sie bei allen Modbus-Funktionen sämtliche unterstützten Datentypen verwenden. MB_RED_CLIENT kann z. B. ein empfangenes Bit auch in eine Variable vom Typ Byte auf eine vorgegebene Adresse schreiben, ohne andere Bits in diesem Byte zu verändern. Man muss also nicht zwingend ein Array von Bits haben, um die bitorientierten Funktionen ausführen zu können.

- Wenn man als Speicherbereich einen Merkerbereich oder einen Global-DB mit Standardzugriff verwendet, besteht für MB_DATA_PTR keine Einschränkung mehr auf die elementaren Datentypen; MB_DATA_PTR kann dann auch auf komplexe Datenstrukturen wie PLC-Datentypen (UDTs) und Systemdatentypen (SDTs) verweisen.

Hinweis

Verwendung eines Merkerbereichs als Datenpuffer

Wenn Sie für MB_DATA_PTR einen Merkerbereich als Datenpuffer verwenden, müssen Sie dessen Größe beachten. Sie beträgt bei S7-1500(R/H)-CPUs 16 KByte, bei S7-1200-CPUs 8 KByte.

Ausgangsparameter STATUS_x, RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR

Parameter STATUS_x (allgemeine Statusinformationen)

An den Status-Ausgängen der Anweisung "MB_RED_CLIENT" werden die Fehlermeldungen angezeigt.

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
0A90	Die Anweisung "MB_RED_CLIENT" ist nicht lizenziert. Nähere Informationen finden Sie hier: Lizenzierung (Seite 4331).
0AFF	Die Verbindung ist nicht parametriert und wird nicht verwendet. Die Verbindung 0A muss parametriert sein.
7000	Kein Auftrag aktiv und keine Verbindung aufgebaut (REQ=0, DISCONNECT=1).
7001	Verbindungsaufbau angestoßen.
7002	Zwischenaufruf. Verbindung wird aufgebaut.

STATUS* (W#16#)	Beschreibung
7003	Verbindung wird abgebaut.
7004	Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.
7005	Daten werden gesendet.
7006	Daten werden empfangen.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS_x (Protokoll-Fehler)

STATUS* (W#16#)	Beschreibung
80C8	Keine Antwort des Servers im definierten Zeitraum. Überprüfen Sie die Verbindung zu dem Modbus-Server. Dieser Fehler wird erst nach Abschluss der konfigurierten erneuten Versuche gemeldet. Empfängt die Anweisung "MB_RED_CLIENT" keine Antwort mit der ursprünglich übermittelten Transaktions-ID (siehe statische Variable MB_TRANSACTION_ID) innerhalb des definierten Zeitraums, wird dieser Fehlercode ausgegeben.
8380	Empfangenes Modbus-Telegramm hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8382	• Die Länge des Modbus-Telegramms im Telegramm-Header stimmt mit der Anzahl der empfangenen Bytes nicht überein. • Die Anzahl der Bytes stimmt mit den tatsächlich übertragenen Bytes nicht überein (nur Funktionen 1-4). • Die Startadresse im empfangenen Telegramm stimmt mit der gespeicherten Startadresse nicht überein (Funktionen 5, 6, 15, 16). • Anzahl der Wörter stimmt mit den tatsächlich übertragenen Wörtern nicht überein (Funktionen 15 und 16).
8383	Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs von MB_DATA_PTR (Seite 4334).
8384	• Ungültiger Exception Code emfangen. • Es wurde ein anderer Datenwert empfangen, als ursprünglich vom Client gesendet wurde (Funktionen 5, 6 und 8). • Ungültiger Status-Wert empfangen (Funktion 11)
8385	• Diagnosecode nicht unterstützt. • Es wurde ein anderer Subfunktionscode empfangen, als ursprünglich vom Client gesendet wurde (Funktion 8).
8386	Empfangener Funktionscode entspricht nicht dem ursprünglich gesendeten.
8387	Die Protokoll-ID des vom Server empfangenen Modbus/TCP-Telegramms ist ungleich "0".
8388	Durch den Modbus-Server wurde eine andere Datenlänge gesendet, als bearbeitet wurde. Dieser Fehler tritt nur bei Verwendung der Modbus-Funktionen 5, 6, 15 oder 16 auf.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS_x (Parameter-Fehler)

STA-TUS* (W#16#)	Beschreibung	
80B6	Ungültiger Verbindungstyp, nur TCP-Verbindungen werden unterstützt.	
80BB	Ungültiger Wert am Parameter ActiveEstablished: Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ActiveEstablished = TRUE).	
8188	Ungültiger Wert am Parameter MB_MODE.	
8189	Ungültige Adressierung der Daten am Parameter MB_DATA_ADDR.	
818A	Ungültige Datenlänge am Parameter MB_DATA_LEN.	
818B	Ungültiger Zeiger am Parameter MB_DATA_PTR. Überprüfen Sie zudem die Werte der Parameter MB_DATA_ADDR (Seite 4334) und MB_DATA_LEN.	
818C	Zeitfehler bei dem Parameter BLOCKED_PROC_TIMEOUT oder RCV_TIMEOUT (siehe statische Variablen der Anweisung). BLOCKED_PROC_TIMEOUT und RCV_TIMEOUT müssen zwischen 0,5 s und 55 s liegen.	
8200	• Eine andere Modbus-Anfrage wird zur Zeit über den Port verarbeitet. • Eine weitere Instanz von MB_RED_CLIENT mit den gleichen Verbindungsparametern bearbeitet eine bereits bestehende Modbus-Anfrage.	
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Hinweis

Fehlercodes der intern verwendeten Kommunikations-Anweisungen

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_RED_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" und "TRESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_RED_CLIENT" möglich. Im Abschnitt "Static" werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Hinweis

Kommunikationsfehler beim Senden oder Empfangen von Daten

Wenn beim Senden bzw. Empfangen von Daten ein Kommunikationsfehler auftritt (80C4 (Temporary communications error. The specified connection is temporarily down.), 80C5 (Remote partner closed connection actively.), 80A1 (The specified connection is disconnected or is not yet established.)), dann wird die bestehende Verbindung abgebaut.

Das bedeutet auch, dass Sie alle STATUS-Werte sehen, die beim Verbindungsabbau zurückgegeben werden, und erst wenn die Verbindung abgebaut ist, wird der STATUS-Code ausgegeben, der die Ursache für den Verbindungsabbau gewesen ist.

Beispiel: Tritt beim Empfangen der Daten ein temporärer Kommunikationsfehler auf, dann wird zunächst der STATUS 7003 (ERROR=false) ausgegeben und dann 80C4 (ERROR=true).

MB_RED_SERVER: Als MODBUS-TCP-Server redundant über PROFINET kommunizieren

Beschreibung MB_RED_SERVER

Beschreibung

Mit dieser Anweisung ist es möglich, eine Kopplung zwischen einer 1200/1500-CPU und einem Gerät, welches das Protokoll Modbus/TCP unterstützt, aufzubauen.

Die Anweisung "MB_RED_SERVER" kommuniziert als Modbus/TCP-Server über die PROFINET-Verbindung. Die Anweisung "MB_RED_SERVER" verarbeitet Verbindungsanfragen eines Modbus/TCP-Clients, empfängt und bearbeitet Modbus-Anfragen und sendet Antwort-Meldungen.

Die Anweisung "MB_RED_SERVER" kann in folgenden CPUen eingesetzt werden:

	"MB_RED_SERVER" V1.0	"MB_RED_SERVER" V1.1
S7-1200	ab FW V4.2	ab FW V4.2
S7-1500	ab FW V2.5	ab FW V2.5
S7-1500R	--	ab FW V2.6
S7-1500H	--	ab FW V2.6

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

ACHTUNG
Security-Hinweis Beachten Sie, dass jeder Client des Netzwerks lesenden und schreibenden Zugriff erhält auf das Prozessabbild der Ein- und Ausgänge und auf den durch das Modbus-Halteregister definierten Datenbaustein oder Merkerbereich. Zum Schutz vor unbefugten Lese- und Schreibvorgänge besteht die Möglichkeit, den Zugriff auf eine IP-Adresse einzuschränken. Beachten Sie jedoch, dass sich die freigegebene Adresse auch für einen unbefugten Zugriff nutzen lässt.

Mehrfache Server-Verbindungen

Die CPUs sind in der Lage:

- mehrere Serververbindungen zu bearbeiten und
- auf einem Serverport gleichzeitig mehrere Verbindungen von verschiedenen Clients anzunehmen.

Die maximale Anzahl der Verbindungen ist von der verwendeten CPU abhängig und kann den technischen Daten der CPU entnommen werden. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus/TCP-Clients und -Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Bei den Server-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB_RED_SERVER"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jeden Client, der sich auf den Server-Port verbinden will, ist je eine eindeutige Verbindung/Verbindungs-ID notwendig.
- Die Verbindungs-IDs müssen CPU-weit eindeutig sein.

Informationen bezüglich des Redundanz-Aufbaus erhalten Sie hier: Arbeitsweise und Redundanzverhalten (Seite 4344).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB_RED_SERVER":

Parameter	Deklaration	Datentyp	Beschreibung
REG_KEY (Seite 4356)	Input	STRING[17]	Freischaltcode für die Lizenzierung Die Anweisung "MB_RED_SERVER" muss auf jeder CPU einzeln lizenziert werden.
DISCONNECT	Input	BOOL	Die Anweisung "MB_RED_SERVER" geht eine passive Verbindung mit einer Partner-Baugruppe ein. Der Server reagiert auf eine Verbindungsanfrage von den IP-Adressen, die in den Verbindungsbeschreibungen spezifiziert oder unspezifiziert angegeben sind. Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird: • 0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufgebaut. • 1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine weiteren Anfragen des Clients bearbeitet und der Verbindungsabbau eingeleitet. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS_x der Wert 0003 ausgegeben.
MB_HOLD_REG (Seite 4358)	InOut	VARIANT	Zeiger auf das Modbus-Halteregister der "MB_RED_SERVER"-Anweisung MB_HOLD_REG muss immer auf einen Speicherbereich verweisen, der größer als zwei Byte ist. Das Halteregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (Lesen), 6 (Schreiben), 16 (Mehrfachschreiben) und 23 (Schreiben und Lesen in einem Auftrag) zugreifen darf.
LICENSED (Seite 4356)	Output	BOOL	• 0: Anweisung ist nicht lizenziert • 1: Anweisung ist lizenziert
IDENT_CODE (Seite 4356)	Output	STRING[18]	Identifikation für die Lizenzierung. Fordern Sie mit diesem String den Freischaltcode REG_KEY an.
DR_NDR_0A	Output	BOOL	"Data Read" or "New Data Ready" auf Verbindung 0A: • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client gelesen oder geschrieben
ERROR_0A	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_RED_SERVER" auf Verbindung 0A ein Fehler auf, wird der Ausgang am Parameter ERROR_0A auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS_0A angezeigt.

Parameter	Dekla-ration	Datentyp	Beschreibung
STATUS_0A (Seite 4360)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 0A.
DR_NDR_1A	Output	BOOL	"Data Read" or "New Data Ready" auf Verbindung 1A: • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client gelesen oder geschrieben
ERROR_1A	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_RED_SERVER" auf Verbindung 1A ein Fehler auf, wird der Ausgang am Parameter ERROR_1A auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS_1A angezeigt.
STATUS_1A (Seite 4360)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 1A.
DR_NDR_0B	Output	BOOL	"Data Read" or "New Data Ready" auf Verbindung 0B: • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client gelesen oder geschrieben
ERROR_0B	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_RED_SERVER" auf Verbindung 0B ein Fehler auf, wird der Ausgang am Parameter ERROR_0B auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS_0B angezeigt.
STATUS_0B (Seite 4360)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 0B.
DR_NDR_1B	Output	BOOL	"Data Read" or "New Data Ready" auf Verbindung 1B: • 0: Keine neuen Daten • 1: Neue Daten durch den Modbus-Client gelesen oder geschrieben
ERROR_1B	Output	BOOL	Tritt während eines Aufrufs der Anweisung "MB_RED_SERVER" auf Verbindung 1B ein Fehler auf, wird der Ausgang am Parameter ERROR_1B auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS_1B angezeigt.
STATUS_1B (Seite 4360)	Output	WORD	Detaillierte Statusinformation der Anweisung auf Verbindung 1B.
RED_ERR_S7 (Seite 4344)	Output	BOOL	• 0: kein Redundanzfehler auf der SIMATIC-Seite • 1: Redundanzfehler auf der SIMATIC-Seite
RED_ERR_DEV (Seite 4344)	Output	BOOL	• 0: kein Redundanzfehler auf der Seite des Koppelpartners • 1: Redundanzfehler auf der Seite des Koppelpartners
TOT_COM_ERR (Seite 4344)	Output	BOOL	• 0: mindestens eine projektierte Verbindung ist aufgebaut • 1: vollständiger Kommunikationsverlust, alle projektierten Verbindungen sind abgebaut

Abbilden der Modbus-Adressen auf dem Prozessabbild

Die Anweisung "MB_RED_SERVER" ermöglicht eingehenden Modbus-Funktionen (1, 2, 4, 5 und 15) direkt lesend und schreibend auf das Prozessabbild der Eingänge und Ausgänge der CPU zuzugreifen (Verwendung der Datentypen BOOL und WORD).

Bei den S7-1200-CPUs beträgt der Adressraum sowohl für das Prozessabbild der Eingänge als auch für das Prozessabbild der Ausgänge 1 KByte, bei den S7-1500-CPUs jeweils 32 KByte.

Die folgende Tabelle zeigt den Adressraum der oben genannten Modbus-Funktionen.

Modbus-Funktion					
Funktionscode	Funktion	Datenbereich	Adressraum		
01	Lesen: Bits	Output	0	bis	65.535
02	Lesen: Bits	Input	0	bis	65.535
04	Lesen: WORD	Input	0	bis	65.535
05	Schreiben: Bit	Output	0	bis	65.535
15	Schreiben: Bits	Output	0	bis	65.535

Eingehende Modbus-Anfragen mit den Funktionscodes 3, 6, 16 und 23 schreiben oder lesen die Modbus-Halteregister (das Halteregister spezifizieren Sie am Parameter MB_HOLD_REG oder über Data_Area_Array).

Modbus-Funktionen

In der folgenden Tabelle sind alle Modbus-Funktionen aufgelistet, die von der Anweisung "MB_RED_SERVER" unterstützt werden.

Funktionscode	Beschreibung
01	Ausgangs-Bits lesen
02	Eingangs-Bits lesen
03	Halteregister lesen
04	Eingangs-Worte lesen
05	Ein Ausgangs-Bit schreiben
06	Ein Halteregister schreiben
08	Diagnosefunktion: • Echo-Test (Subfunktion 0x0000): Die Anweisung "MB_RED_SERVER" empfängt ein Datenwort und sendet dieses unverändert an den Modbus-Client zurück. • Ereigniszähler zurücksetzen (Subfunktion 0x000A): Die Anweisung "MB_RED_SERVER" setzt die folgenden Ereigniszähler zurück: "Success_Count", "Xmt_Rcv_Count", "Exception_Count", "Server_Message_Count" und "Request_Count".
11	Diagnosefunktion: Ereigniszähler der Kommunikation holen Die Anweisung "MB_RED_SERVER" verwendet einen internen Ereigniszähler bei der Kommunikation zur Erfassung der Anzahl der erfolgreich ausgeführten Lese- und Schreibanfragen, die an den Modbus-Server gesendet werden. Der Ereigniszähler wird bei den Funktionen 8 oder 11 nicht inkrementiert. Gleiches gilt für Anfragen, die einen Kommunikationsfehler zur Folge haben, zum Beispiel falls ein Protokollfehler aufgetreten ist (z. B. der Funktionscode in der empfangenen Modbus-Anfrage nicht unterstützt wird).
15	Ausgangs-Bits schreiben
16	Halteregister schreiben
23	Mit einem Request Halteregister schreiben und Halteregister lesen

Arbeitsweise und Redundanzverhalten

Beschreibung

Die Kommunikationsteilnehmer können standalone oder redundant aufgebaut werden. Ist einer der beiden Teilnehmer standalone aufgebaut, spricht man von einseitiger Redundanz. Sind beide Partner redundant aufgebaut, bezeichnet man es als beidseitige Redundanz.

Portnummer für Client und Server

Der Modbus Client verwendet eine Portnummer ab 2000. Der Modbus Server wird normalerweise über die Portnummer 502 angesprochen. Abhängig von der CPU ist es möglich, den Port 502 für mehrere Verbindungen zu projektieren (Mulitport). Wurde für zwei oder mehr Verbindungen der lokale Port 502 projektiert, werden bei unspezifizierten Verbindungen die anfragenden Clients zufällig auf die vorhandenen Server-Verbindungen verteilt. Der 1. Client, der sich auf die Anweisung "MB_RED_SERVER" verbinden will, erhält nicht automatisch die Verbindung 0A zugeordnet. Ist die Zuordnung der Client-Anfragen zu den Server-Verbindungen erfolgt, bleibt die Zuordnung während des Telegrammverkehrs so lange bestehen bis die Verbindung abgebaut wird.

Einseitige Redundanz

Beschreibung

Für jede Verbindung zwischen den Kommunikationspartnern muss je eine Verbindung projektiert werden. Die Verbindungspunkte der **SIMATIC S7** haben die Bezeichnungen **0** und **1**, die Verbindungspunkte des **Kommunikationspartners** haben die Bezeichnungen **A** und **B**.

Die R- bzw. H-CPU 1 bezeichnet den Verbindungspunkt 0, die R- bzw. H-CPU 2 bezeichnet den Verbindungspunkt 1.

Projektierung

Ist die S7 redundant aufgebaut, wird eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt A des Koppelpartners und eine Verbindung vom Verbindungspunkt 1 der S7 zum Knotenpunkt A des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **A** => Verbindung **1A**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

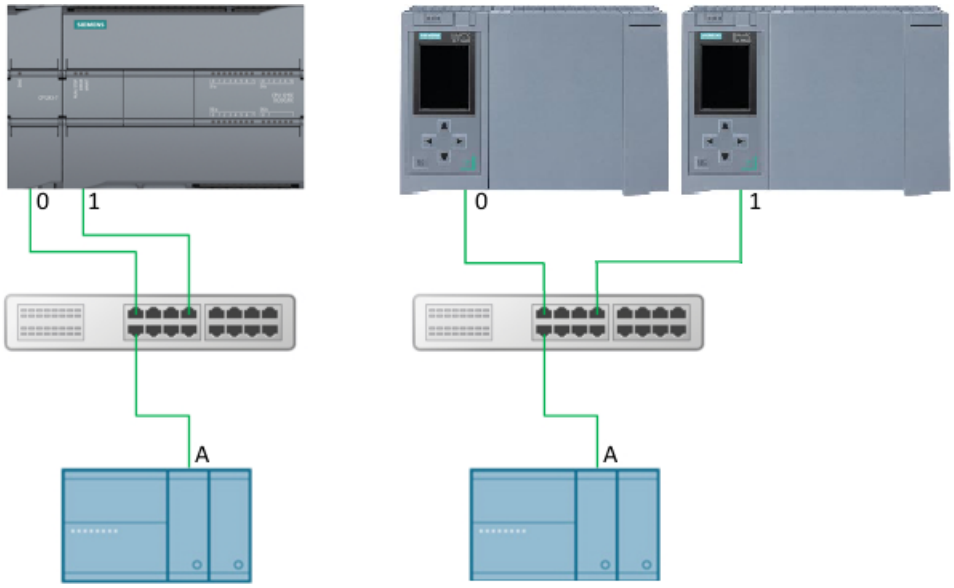

Bild 4-15 Modbus/TCP: Einseitige Redundanz S7

Ist die S7 standalone und der Koppelpartner redundant aufgebaut, wird eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt A des Koppelpartners und eine Verbindung vom Verbindungspunkt 0 der S7 zum Knotenpunkt B des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **B** => Verbindung **0B**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

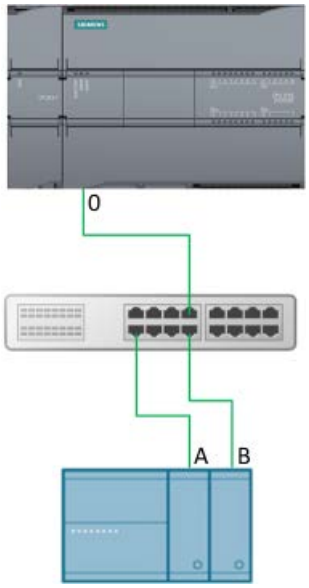

Bild 4-16 Modbus/TCP: Einseitige Redundanz Partner

Beidseitige Redundanz

Beschreibung

Für jede Verbindung zwischen den Kommunikationspartnern muss je eine Verbindung projektiert werden. Die Verbindungspunkte der **SIMATIC S7** haben die Bezeichnungen **0** und **1**, die Verbindungspunkte des **Kommunikationspartners** haben die Bezeichnungen **A** und **B**.

Die R- bzw. H-CPU 1 bezeichnet den Verbindungspunkt 0, die R- bzw. H-CPU 2 bezeichnet den Verbindungspunkt 1.

Projektierung

Bei beidseitiger Redundanz werden zwei Verbindungen vom Verbindungspunkt 0 und zwei Verbindungen vom Verbindungspunkt 1 der S7 zu den Knotenpunkten A und B des Koppelpartners angelegt.

- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **A** => Verbindung **0A**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **A** => Verbindung **1A**
- Verbindung von der S7 Verbindungspunkt **0** zum Partner/Knoten **B** => Verbindung **0B**
- Verbindung von der S7 Verbindungspunkt **1** zum Partner/Knoten **B** => Verbindung **1B**

Die Abbildung veranschaulicht die Verbindungsbezeichnungen graphisch.

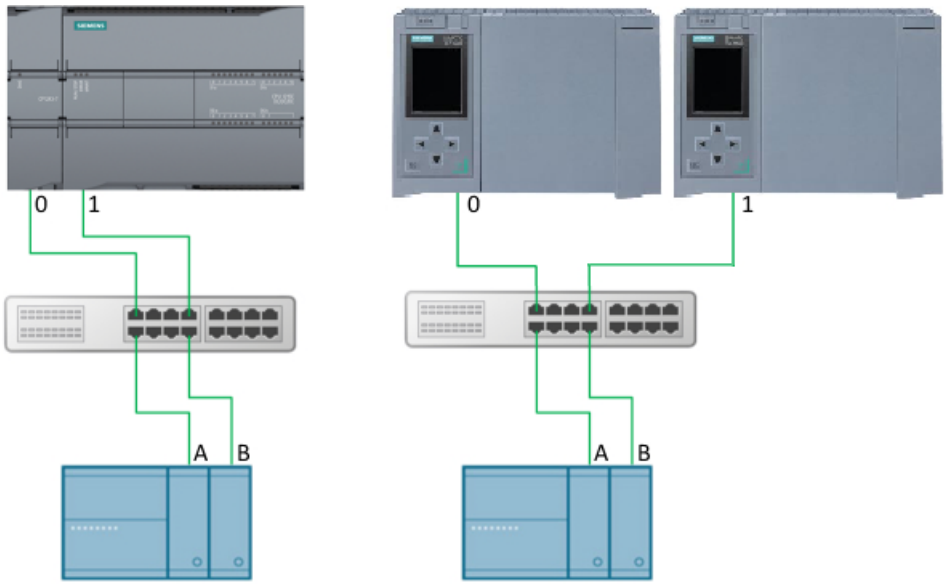

Bild 4-17 Modbus/TCP: Beidseitige Redundanz

Telegrammbearbeitung

Die Telegramme können über alle projektierten Verbindungen empfangen werden. Der Client kann wahlweise über 1 Verbindung oder über alle Verbindungen Telegramme senden. Wurde auf einer Verbindung ein Telegramm empfangen, wird der Status am entsprechenden Ausgang DR_NDR_x bzw. ERROR_x angezeigt. Jede Verbindung läuft autark und hat keinen Einfluß auf die Anzeige der anderen Verbindungen.

Redundanz-Ausgänge RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR

Beschreibung

Anhand der Zustände der Status-Ausgänge werden die Redundanz-Bits RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR gesetzt.

Anzeige der Alarm-Bits für beidseitigen Redundanz-Aufbau

Anzahl der fehlerhaften Verbindungen	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	okay	okay	okay	FALSE	FALSE	FALSE
1	okay	okay	okay	Error	FALSE	FALSE	FALSE
	okay	okay	Error	okay	FALSE	FALSE	FALSE
	okay	Error	okay	okay	FALSE	FALSE	FALSE
	Error	okay	okay	okay	FALSE	FALSE	FALSE
2	okay	okay	Error	Error	TRUE	FALSE	FALSE
	okay	Error	okay	Error	FALSE	TRUE	FALSE
	Error	okay	okay	Error	FALSE	FALSE	FALSE
	okay	Error	Error	okay	FALSE	FALSE	FALSE
	Error	okay	Error	okay	FALSE	TRUE	FALSE
	Error	Error	okay	okay	TRUE	FALSE	FALSE
3	Error	Error	Error	okay	TRUE	TRUE	FALSE
	Error	Error	okay	Error	TRUE	TRUE	FALSE
	Error	okay	Error	Error	TRUE	TRUE	FALSE
	okay	Error	Error	Error	TRUE	TRUE	FALSE
4	Error	Error	Error	Error	TRUE	TRUE	TRUE

Anzeige der Alarm-Bits für einseitigen Redundanz-Aufbau

Anzahl der fehlerhaften Verbindungen	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	0AFF	okay	0AFF	FALSE	FALSE	FALSE
1	okay	0AFF	Error	0AFF	TRUE	TRUE	FALSE
	Error	0AFF	okay	0AFF	TRUE	TRUE	FALSE
2	Error	0AFF	Error	0AFF	TRUE	TRUE	TRUE

Parametrierung

Beschreibung

Die Anweisung "MB_RED_SERVER" **V1.0** und **V1.1** kann für S7-1200 und für S7-1500 eingesetzt werden.

S7-1200

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisiert. Hierbei werden die Verbindungen mit Hilfe der Struktur TCON_IP_V4 parametriert und aufgebaut.

S7-1500

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisiert. Hierbei sind sowohl parametrierte Verbindungen über die Struktur TCON_IP_V4 als auch konfigurierte Verbindungen über die Struktur TCON_Configured möglich.

Die Anweisung "MB_RED_SERVER" **V1.1** kann für S7-1200R und für S7-1500H eingesetzt werden.

S7-1500R und S7-1500H

Die Verbindungen werden über die lokale Schnittstelle der CPU oder eines CM/CP realisert. Hierbei werden die Verbindungen mit Hilfe der Struktur TCON_IP_V4 parametriert und aufgebaut.

Konfiguration von "MB_RED_SERVER"

Mit Hilfe des Konfigurations-Dialoges der Anweisung "MB_RED_SERVER" werden folgende Einstellungen vorgenommen:

- Verbindungsparameter für die Verbindungen 0A, 1A, 0B und 1B (Redundanzaufbau siehe Arbeitsweise und Redundanzverhalten (Seite 4344))
- Interne Parameter (optional)

Der Konfigurationsdialog kann über die Anweisung oder über die Technologieobjekte geöffnet werden.

Verbindungsparameter

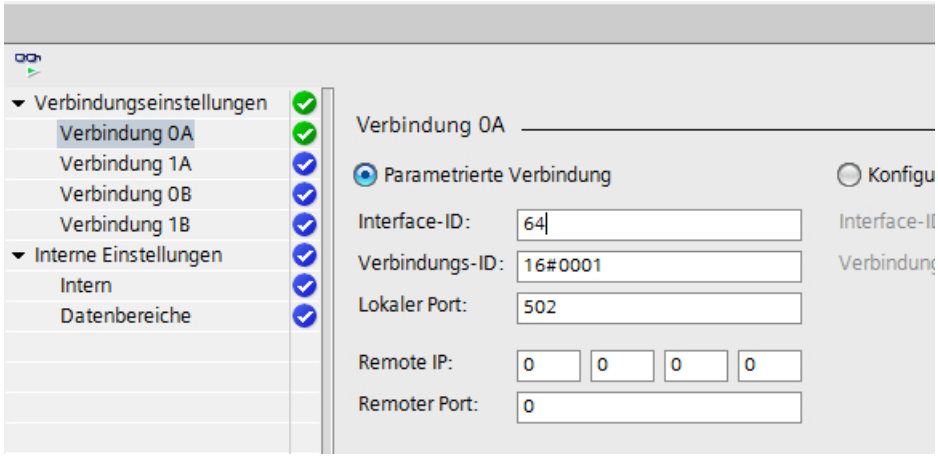

Bild 4-18 Modbus/TCP: parametrierte Server-Verbindung

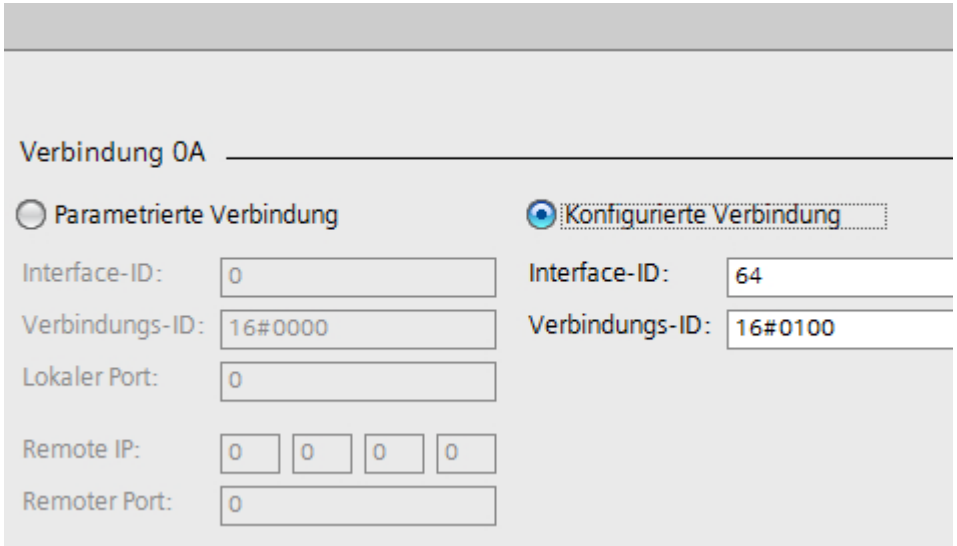

Bild 4-19 Modbus/TCP: konfigurierte Server-Verbindung

Variable	Startwert	Beschreibung
Parametrierte Verbindungen		
Interface ID	64	HW-Identifier der verwendeten PN-Schnittstelle
Connection ID	16#0000	Verbindungs-IDs für die verwendeten Verbindungen. Diese Verbindungs-IDs müssen CPU-weit eindeutig sein.
Local port	502	Lokale Portnummer des Serverbausteins. Der Standard-Port für Modbus/TCP-Server ist 502.
Remote IP	0.0.0.0	Remote IP-Adresse des Clients. Standardmässig wird keine IP-Adresse für den Client eingetragen.
Remote port	0	Remote Portnummer des Clients. Standardmässig wird keine Portnummer für den Client eingetragen.
Konfigurierte Verbindungen		

Variable	Startwert	Beschreibung
Interface ID	64	HW-Identifier der verwendeten PN-Schnittstelle
Connection ID	16#0000	Verbindungs-IDs für die vewendeten Verbindungen. Diese Verbindungen werden in der Netzsicht konfiguriert.

Interne Parameter (optional)

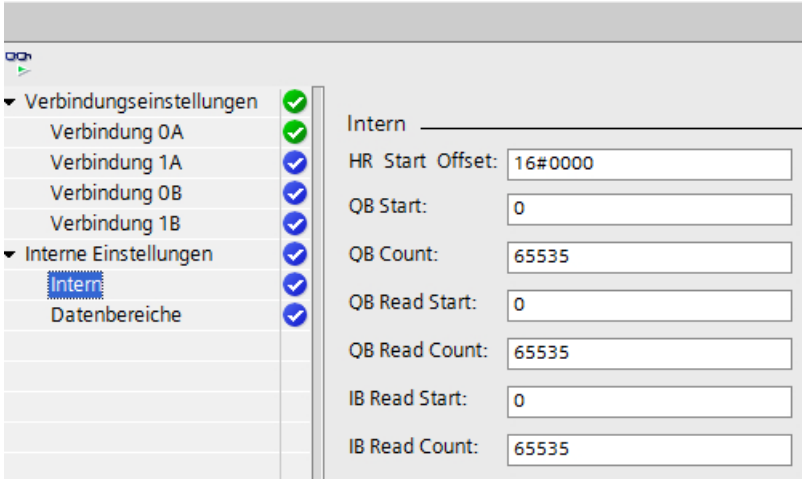

Variable	Datentyp	Start-wert	Beschreibung
HR_Start_Offset	WORD	0	Zuweisen der Anfangsadresse des Modbus-Halteregisters.
QB_Start	UINT	0	Anfangsadresse des zulässigen Adressierungsbereichs der Ausgänge, die geschrieben werden können (Byte 0 bis 65535)
QB_Count	UINT	0	Anzahl der Ausgangs-Bytes, die durch den Modbus-Master geschrieben werden können. *Beispiel:* QB_Start=0 und QB_Count=10: Ausgangs-Bytes 0 bis 9 können geschrieben werden. QB_Count=0: Kein Ausgangsbyte kann geschrieben werden.
QB_Read_Start	UINT	0	Anfangsadresse des zulässigen Adressierungsbereichs der Ausgänge, die gelesen werden können (Byte 0 bis 65535)
QB_Read_Count	UINT	0	Anzahl der Ausgangs-Bytes, die durch den Modbus-Master gelesen werden können. *Beispiel:* QB_Read_Start=0 und QB_Read_Count=10: Ausgangs-Bytes 0 bis 9 können gelesen werden. QB_Read_Count=0: Kein Ausgangsbyte kann gelesen werden.
IB_Read_Start	UINT	0	Anfangsadresse des zulässigen Adressierungsbereichs der Eingänge, die gelesen werden können (Byte 0 bis 65535)

Variable	Datentyp	Start-wert	Beschreibung
IB_Read_Count	UINT	0	Anzahl der Eingangs-Bytes, die durch den Modbus-Master gelesen werden können. *Beispiel:* IB_Read_Start=0 und IB_Read_Count=10: Eingangs-Bytes 0 bis 9 können gelesen werden. IB_Read_Count=0: Kein Eingangsbyte kann gelesen werden.
Data_Area_Array	ARRAY [1..8]		
data_type	UINT	0	Data Type: 0 bis 4
db	UINT	0	Datenbausteinnummer
start	UINT	0	1. Modbusadresse im Datenbaustein
length	UINT	0	Anzahl der Modbus-Werte im Datenbaustein

Adressierung über statische Variable HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 0.

Beispiel: Das Halteregister beginnt am MW100 und hat eine Länge von 100 WORD.

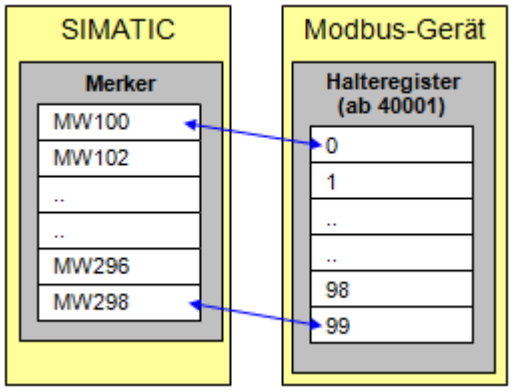

Bild 4-20 Modbus/TCP: HR_Start_Offset = 0

Sie können die Variable HR_Start_Offset so definieren, dass das Modbus-Halteregister eine andere Anfangsadresse als 0 erhält.

Beispiel: Ein Offset-Wert 20 am Parameter HR_Start_Offset bewirkt, dass sich die Anfangsadresse des Halteregisters von 0 auf 20 verschiebt. Jede Adressierung des Halteregisters, die unterhalb der Adresse 20 und oberhalb der Adresse 119 liegt, ruft einen Fehler hervor.

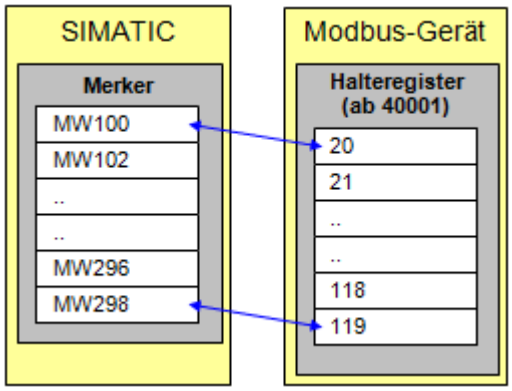

Bild 4-21 Modbus/TCP: HR_Start_Offset = 20

Data_Area_Array [1..8]

Beschreibung

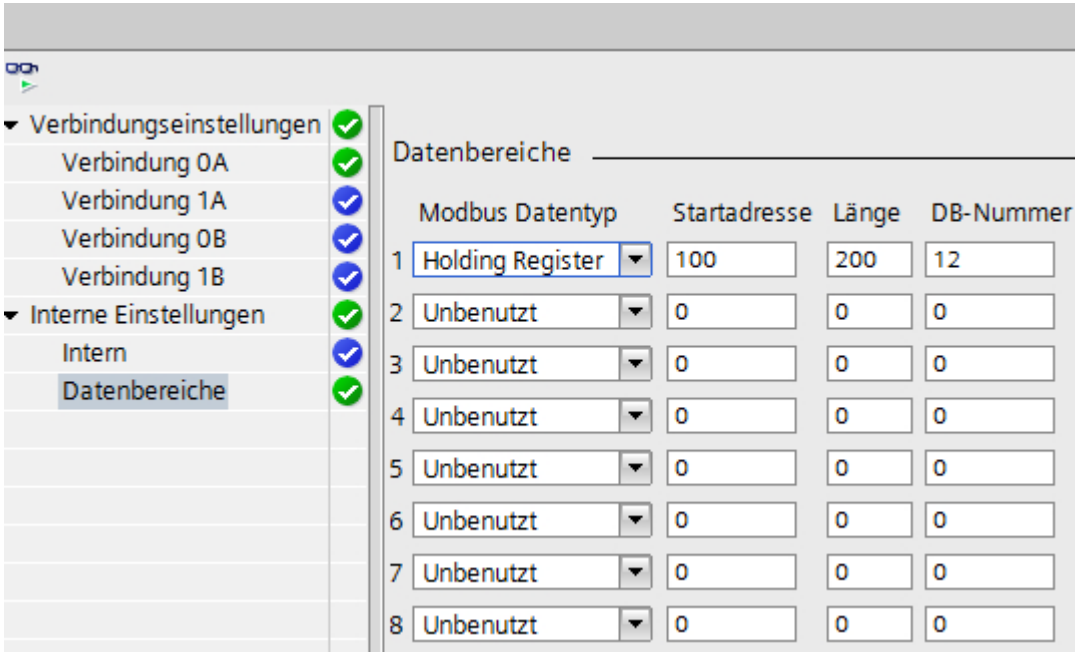

Es werden 8 Datenbereiche für die Abbildung der MODBUS Adressen im S7-Speicher angeboten. Ist der erste Datenbereich definiert, wird der Parameter an MB_HOLD_REG nicht ausgewertet. Stattdessen werden die Modbus-Register und Bits abhängig vom Auftragstyp in Datenbausteine geschrieben oder aus ihnen gelesen. Im folgenden Programmverlauf können diese Werte weiterverarbeitet werden.

Mit einem Auftrag kann immer nur aus einem DB gelesen / in einen DB geschrieben werden. Zugriffe auf Register oder Bitwerte, die in mehreren DBs liegen, auch wenn die Nummern lückenlos hintereinander liegen, sind auf zwei Aufträge aufzuteilen. Dies ist bei der Projektierung zu berücksichtigen. Es ist möglich in einen Datenbaustein mehr Modbusbereiche (Register oder Bitwerte) abzubilden als mit einem Telegramm bearbeitet werden können.

data_type

Mit dem Parameter data_type wird angegeben, welche MODBUS Datentypen in diesem Datenbaustein abgebildet werden. Wird in data_type der Wert 0 eingetragen, wird der entsprechende Datenbereich nicht verwendet. Sollen mehrere Data_Area verwendet werden, müssen diese nacheinander definiert werden. Alle Einträge nach einem data_type = 0 werden nicht mehr bearbeitet.

Kennung	Datentyp	Beschreibung
0	Bereich nicht verwendet	
1	Ausgangsbits (Coils)	Bit
2	Eingangsbits (Inputs)	Bit
3	Halteregister (Holding Register)	Word
4	Eingangswörter (Input Register)	Word

db

Der Parameter db legt den Datenbaustein fest, in dem die nachfolgend definierten MODBUS Register oder Bitwerte abgebildet werden. Die DB-Nummer 0 ist nicht erlaubt, da diese für das System reserviert ist.

start, length

Mit start wird die erste Modbusadresse, die im Datenwort 0 des DB abgebildet wird, angegeben. Der Parameter length definiert die Länge, wie viele MODBUS-Adressen in dem Datenbaustein abgebildet werden. Die definierten Datenbereiche dürfen sich nicht überlappen. Der Parameter length darf nicht 0 sein.

Beispiel: Adressabbildung mit Data_Area_Array

Datenbereich 1	data_type	3: Halteregister
	db	11
	start	0
	length	500
Datenbereich 2	data_type	3: Halteregister
	db	12
	start	720
	length	181
Datenbereich 3	data_type	4: Eingangswörter
	db	13
	start	720
	length	281
Datenbereich 4	data_type	1: Ausgangsbits
	db	14
	start	640
	length	611

Datenbereich 5	data_type	2: Eingangsbits
	db	15
	start	1700
	length	601
Datenbereich 6	data_type	1: Ausgangsbits
	db	16
	start	1700
	length	601
Datenbereich 7	data_type	Nicht verwendet
	db	0
	start	0
	length	0
Datenbereich 8	data_type	Nicht verwendet
	db	0
	start	0
	length	0

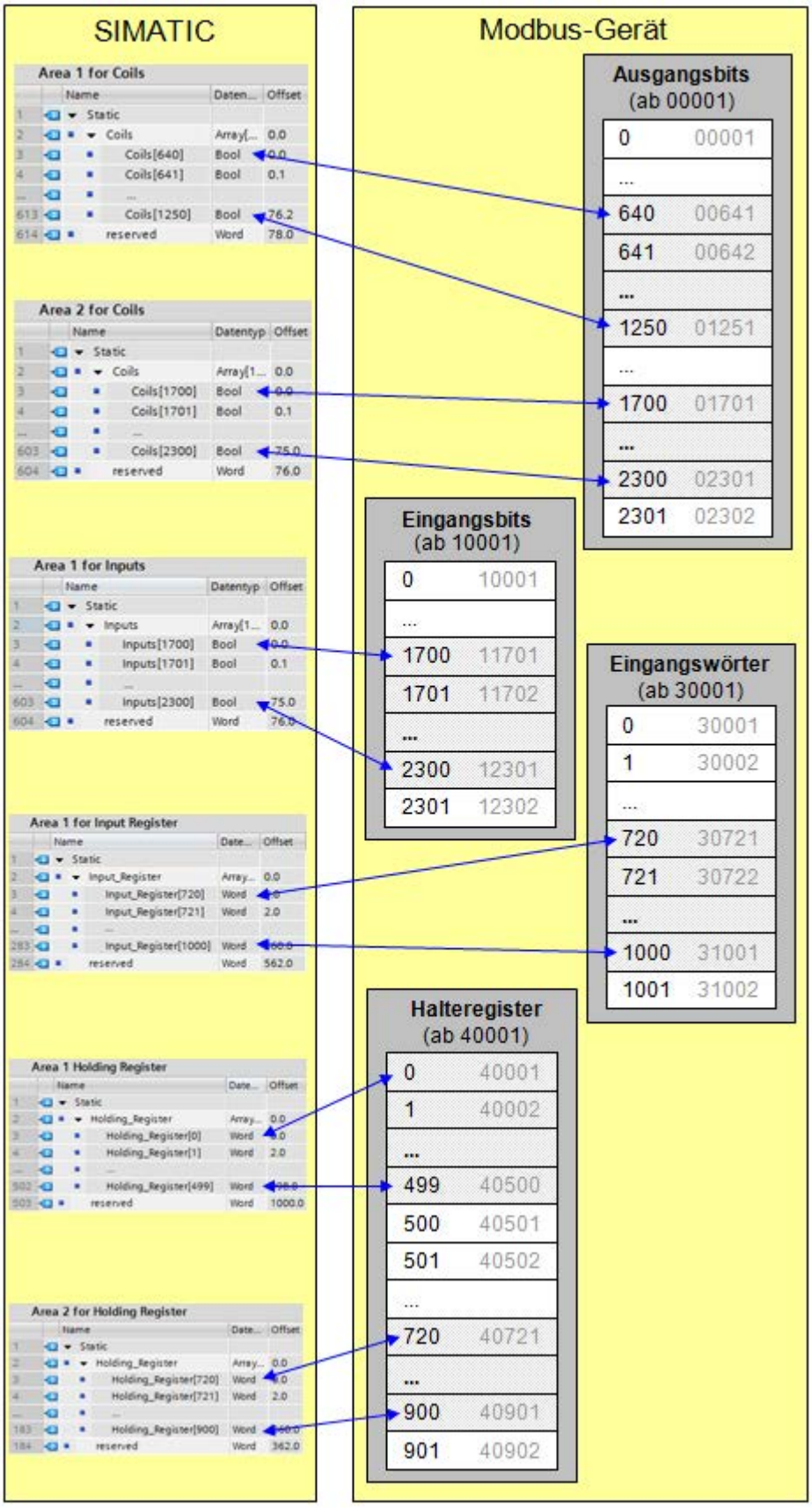
SIMATIC
Area 1 for Coils
Name Daten... Offset
1 Static
2 Coils Array[... 0.0
3 Coils[640] Bool 0.0
4 Coils[641] Bool 0.1
... ...
613 Coils[1250] Bool 76.2
614 reserved Word 78.0
Area 2 for Coils
Name Datentyp Offset
1 Static
2 Coils Array[1... 0.0
3 Coils[1700] Bool 0.0
4 Coils[1701] Bool 0.1
... ...
603 Coils[2300] Bool 75.0
604 reserved Word 76.0
Area 1 for Inputs
Name Datentyp Offset
1 Static
2 Inputs Array[1... 0.0
3 Inputs[1700] Bool 0.0
4 Inputs[1701] Bool 0.1
... ...
603 Inputs[2300] Bool 75.0
604 reserved Word 76.0
Area 1 for Input Register
Name Date... Offset
1 Static
2 Input_Register Array... 0.0
3 Input_Register[720] Word 0.0
4 Input_Register[721] Word 2.0
... ...
283 Input_Register[1000] Word 560.0
284 reserved Word 562.0
Area 1 Holding Register
Name Date... Offset
1 Static
2 Holding_Register Array... 0.0
3 Holding_Register[0] Word 0.0
4 Holding_Register[1] Word 2.0
... ...
502 Holding_Register[499] Word 998.0
503 reserved Word 1000.0
Area 2 for Holding Register
Name Date... Offset
1 Static
2 Holding_Register Array... 0.0
3 Holding_Register[720] Word 0.0
4 Holding_Register[721] Word 2.0
... ...
183 Holding_Register[900] Word 360.0
184 reserved Word 362.0
Modbus-Gerät
Ausgangsbits
(ab 00001)
0 00001
...
640 00641
641 00642
...
1250 01251
...
1700 01701
...
2300 02301
2301 02302
Eingangsbits
(ab 10001)
0 10001
...
1700 11701
1701 11702
...
2300 12301
2301 12302
Eingangswörter
(ab 30001)
0 30001
1 30002
...
720 30721
721 30722
...
1000 31001
1001 31002
Halteregister
(ab 40001)
0 40001
1 40002
...
499 40500
500 40501
501 40502
...
720 40721
...
900 40901
901 40902

Lizenzierung

Die Anweisung "MB_RED_SERVER" ist kostenpflichtig und muss auf jeder CPU einzeln lizenziert werden. Die Lizenzierung erfolgt in 2 Schritten:

- dem Auslesen des IDENT_CODE und
- der Eingabe des Freischaltcodes REG_KEY.

Hinweis

S7-1500R und S7-1500H

Bei einer S7-R/H-Station wird nur die CPU1 lizenziert. Daher ist ein Tausch der CPU1 nach der Lizenzierung nicht mehr möglich.

Vorgehensweise

Auslesen des IDENT_CODE

Für das Auslesen des IDENT_CODE gehen Sie wie folgt vor:

1. Parametrieren Sie die Anweisung "MB_RED_SERVER" Ihren Anforderungen entsprechend in einem zyklischen OB. Laden Sie das Programm in die CPU und setzen Sie diese in RUN.
2. Öffnen Sie den Instanz-DB der Modbus-Anweisung und klicken Sie auf die Schaltfläche „Alle beobachten“ .
3. Am Ausgang IDENT_CODE wird eine 18-stellige Zeichenfolge angezeigt.

	Name	Data type	Start value	Monitor value
1	Input			
2	REG_KEY	String[17]	''	''
3	USE_ALL_CONN	Bool	false	FALSE
4	REQ	Bool	false	FALSE
5	DISCONNECT	Bool	false	FALSE
6	MB_MODE	USInt	0	0
7	MB_DATA_ADDR	UDInt	0	0
8	MB_DATA_LEN	UInt	0	0
9	Output			
10	LICENSED	Bool	false	FALSE
11	IDENT_CODE	String[18]	''	'RTPCFIGDCDIIHJHAH4'
12	DONE	Bool	false	FALSE
13	BUSY	Bool	false	FALSE
14	ERROR	Bool	false	FALSE

4. Kopieren Sie diesen String per Copy/Paste aus dem Datenbaustein und fügen ihn in das Formular ein, welches Ihnen nach der Bestellung des Produktes per E-Mail zugesandt wurde bzw. auf der CD enthalten ist.
5. Senden Sie das Formular über einen Service Request an den Customer Support (https://support.industry.siemens.com/my/ww/de/requests/#createRequest). Sie erhalten daraufhin den Freischaltcode für ihre CPU.

Eingabe des Freischaltcodes REG_KEY

Die Angabe des Freischaltcodes REG_KEY muss an jeder "MB_RED_SERVER"-Anweisung erfolgen. Der REG_KEY sollte in einem Global-Datenbaustein gespeichert werden, über den alle "MB_RED_SERVER"-Anweisungen den notwendigen Freischaltcode erhalten.

Für die Eingabe des Freischaltcodes REG_KEY gehen Sie wie folgt vor:

- Fügen Sie mit „Neuen Baustein hinzufügen…“ einen neuen Global-Datenbaustein mit einem eindeutigen symbolischen Namen, z.B. „Lizenz_DB“, ein.
- Legen Sie in diesem Baustein einen Parameter REG_KEY mit dem Datentyp STRING[17] an.

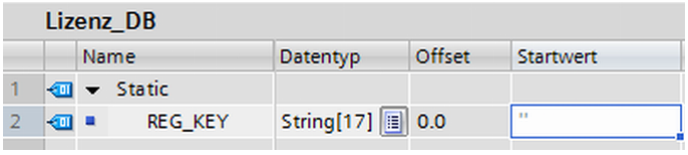

Bild 4-22 REG_KEY in DB

- Kopieren Sie den übermittelten 17-stelligen Freischaltcode per Copy/Paste in die Spalte ‚Startwert’.
- Geben Sie im zyklischen OB am Parameter REG_KEY der "MB_RED_SERVER"-Anweisung den Namen des Lizenz-DBs und den Namen des Strings an, z.B. Lizenz_DB.REG_KEY.
- Laden Sie die geänderten Bausteine in die CPU. Die Eingabe des Freischaltcodes kann zur Laufzeit erfolgen, ein Wechsel von STOP -> RUN ist nicht erforderlich.

Die Modbus/TCP-Kommunikation über die Anweisung "MB_RED_SERVER" ist nun für diese CPU lizenziert, das Ausgangsbit LICENSED ist TRUE.

Fehlende oder fehlerhafte Lizenzierung

Ist kein oder ein falscher Freischaltcode eingetragen, blinkt die ERROR-LED der CPU. Zusätzlich wird bei S7-1200 und S7-1500 zyklisch ein Eintrag in den Diagnosepuffer bezüglich der fehlenden Lizenz vorgenommen. Bei R- und H-Stationen wird nur der Bereichslängenfehler angezeigt.

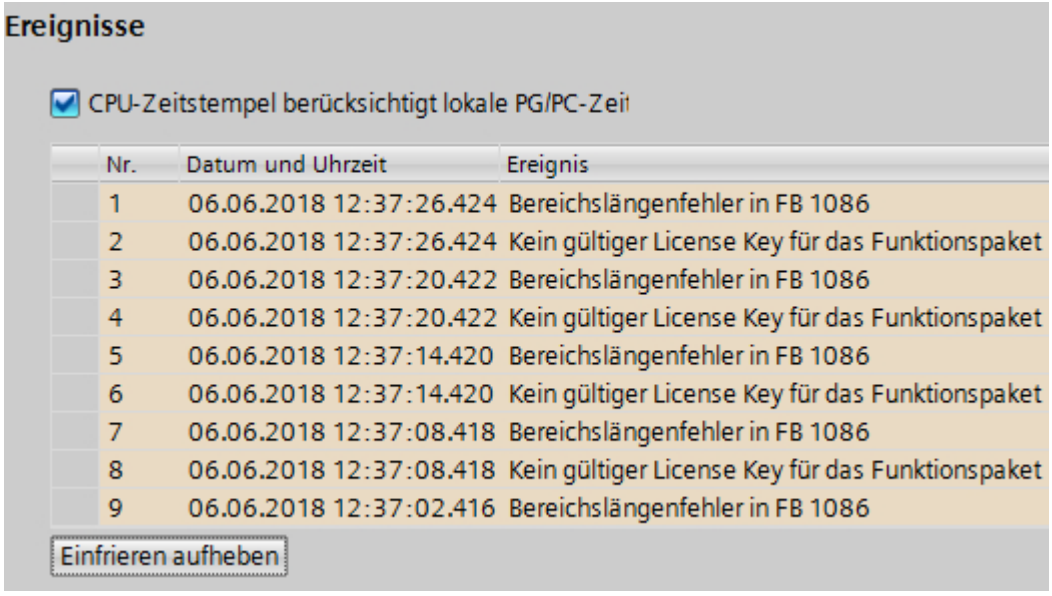

Bei einem fehlenden oder falschen Freischaltcode wird die Modbus/TCP-Kommunikation bearbeitet, allerdings wird am Ausgang STATUS_x stets W#16#0A90 „Keine gültige Lizenz vorhanden“ angezeigt. Das Ausgangsbit LICENSED ist FALSE.

⚠ WARNUNG
Falls in der S7-1500 (R/H) der OB121 in der Steuerung fehlt, wird die CPU in den STOP-Zustand gesetzt.

Eingangsparameter MB_HOLD_REG

Beschreibung

Der Parameter MB_HOLD_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, auf die ein Modbus-Client lesend oder schreibend zugreifen kann. Als Speicherbereich können Sie einen globalen Datenbaustein (D) oder Merker (M) verwenden.

- Die Obergrenze für die Anzahl von Adressen im Datenbaustein (D) ist durch die maximale Größe eines DB Ihrer CPU vorgegeben.
- Die Obergrenze für die Anzahl von Merkern (M) ist durch den maximalen Merkerbereich Ihrer CPU vorgegeben.

Die folgenden Bilder zeigen einige Beispiele zur Abbildung von Modbus-Adressen auf das Halteregister für die Modbus-Funktionen 3 (mehrere WORD lesen), 6 (ein WORD schreiben), 16 (mehrere WORD schreiben) und 23 (mehrere WORD schreiben und lesen).

MB_HOLD_REG: Datenbaustein mit Offset 0

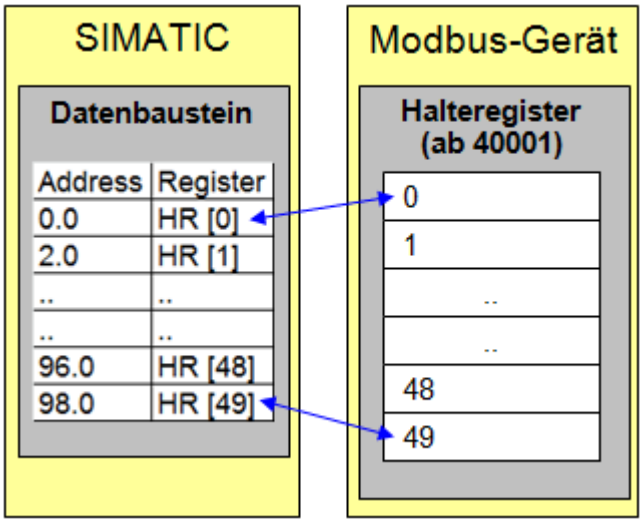

MB_HOLD_REG: Datenbaustein mit Offset 60

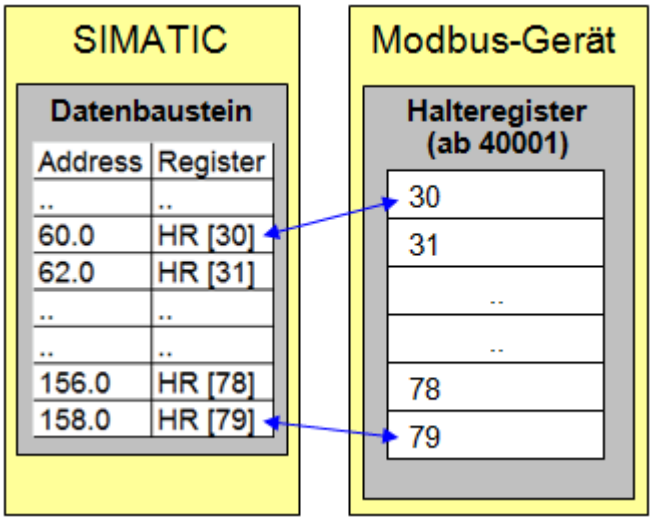

Data_Area_Array [1..8]

Zur Verwendung der optionalen Parameter Data_Area_Array [1..8] siehe Parametrierung (Seite 4348).

Ausgangsparameter ERROR_x, STATUS_x, RED_ERR_S7, RED_ERR_DEV und TOT_COM_ERR

Parameter STATUS_x (allgemeine Statusinformationen)

An den Status-Ausgängen der Anweisung "MB_RED_SERVER" werden die Fehlermeldungen angezeigt.

STATUS* (W#16#)	Beschreibung
0000	Anweisung fehlerfrei ausgeführt.
0001	Verbindung hergestellt.
0003	Verbindungsabbau durchgeführt.
0A90	Die Anweisung "MB_RED_SERVER" ist nicht lizenziert. Nähere Informationen finden Sie hier: Lizenzierung (Seite 4356).
0AFF	Die Verbindung ist nicht parametriert und wird nicht verwendet. Die Verbindung 0A muss parametriert sein.
7000	Kein Aufruf aktiv und keine Verbindung aufgebaut (REQ=0, DISCONNECT=1).
7001	Erstaufruf. Verbindungsaufbau angestoßen.
7002	Zwischenaufruf. Verbindung wird aufgebaut.
7003	Verbindung wird abgebaut.
7005	Daten werden gesendet.
7006	Daten werden empfangen.
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Parameter STATUS_x (Protokoll-Fehler)

STATUS* (W#16#)	Fehlercode in der Fehlermeldung von "MB_RED_SERVER" (B#16#)	Beschreibung
8380	-	Empfangenes Modbus-Telegramm hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.
8381	01	Funktionscode wird nicht unterstützt.
8382	03	Fehler bei Datenlänge: • Unzulässige Längenangabe im empfangenen Modbus-Telegramm • Die im Header des Modbus-Telegramms eingetragene Telegramm-Länge stimmt mit der Anzahl der tatsächlich empfangenen Bytes nicht überein. • Die im Header des Modbus-Telegramms eingetragene Anzahl der Bytes stimmt mit der Anzahl der tatsächlich empfangenen Bytes nicht überein (Funktionen 15 und 16).
8383	02	Fehler bei der Datenadresse oder Zugriff außerhalb des Adressbereichs des Halteregisters (Parameter MB_HOLD_REG (Seite 4358)).
8384	03	Fehler beim Datenwert (Funktion 05)
8385	03	Diagnosecode wird nicht unterstützt (bei Funktion 08)
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Parameter STATUS_x (Parameter-Fehler)

STATUS* (W#16#)	Beschreibung
80BB	Ungültiger Wert am Parameter ActiveEstablished Serverseitig ist nur der passive Verbindungsaufbau zulässig (active_established = FALSE).
8187	Ungültiger Zeiger am Parameter MB_HOLD_REG. Datenbereich ist zu klein.
8389	Ungültige Datenbereich-Definition: • Unzulässiger Wert von data_type • DB-Nummer unzulässig oder nicht vorhanden: – Unzulässiger Wert von db – DB-Nummer existiert nicht – DB-Nummer wird bereits von einem anderen Datenbereich verwendet – DB mit optimiertem Zugriff – DB liegt nicht im Arbeitsspeicher • Unzulässiger Wert von length • Überlappung von MODBUS-Adressbereichen, die zu demselben MODBUS-Datentyp gehören
* Die Statuscodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Hinweis

Fehlercodes der intern verwendeten Kommunikations-Anweisungen

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB_RED_SERVER" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" und "T_RESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB_RED_SERVER" möglich. Im Abschnitt "Static" der einzelnen Instanzen werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

Hinweis

Kommunikationsfehler beim Senden oder Empfangen von Daten

Wenn beim Senden bzw. Empfangen von Daten ein Kommunikationsfehler auftritt (80C4 (Temporärer Kommunikationsfehler. Die angegebene Verbindung ist vorübergehend abgebaut.), 80C5 (Der remote Partner hat die Verbindung aktiv beendet.), 80A1 (Die angegebene Verbindung ist unterbrochen oder wurde noch nicht aufgebaut.)), dann wird die bestehende Verbindung abgebaut.

Das bedeutet auch, dass Sie alle STATUS-Werte sehen, die beim Verbindungsabbau zurückgegeben werden, und erst wenn die Verbindung abgebaut ist, wird der STATUS-Code ausgegeben, der die Ursache für den Verbindungsabbau gewesen ist.

Beispiel: Tritt beim Empfangen der Daten ein temporärer Kommunikationsfehler auf, dann wird zunächst der STATUS 7003 (ERROR=false) ausgegeben und dann 80C4 (ERROR=true).

4.1.6.6 TeleService

TM_MAIL: E-Mail übertragen

Beschreibung TM_MAIL

Beschreibung

Die Anweisung "TM_MAIL" ist eine asynchron arbeitende Anweisung, d.h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Beim Aufruf der Anweisung "TM_MAIL" ist die Angabe einer Instanz zwingend erforderlich. In der Instanz darf das Attribut "Remanent" nicht gesetzt werden. Durch dieses Attribut wird gewährleistet, dass die Instanz beim Übergang der CPU von STOP nach RUN initialisiert wird und danach ein neuer Auftrag zum Senden einer E-Mail angestoßen werden kann.

Sie starten das Versenden einer E-Mail mit einem Flankenwechsel von "0" auf "1" am Parameter REQ. Über die Ausgangsparameter "BUSY", "DONE", "ERROR" und die Ausgangsparameter "STATUS" und "SFC_STATUS" wird der Zustand des Auftrags angezeigt. Dabei entspricht "SFC_STATUS" dem Ausgangsparameter "STATUS" der aufgerufenen Kommunikations-Bausteine.

Die Ausgangsparameter DONE, ERROR, STATUS und SFC_STATUS werden jeweils nur einen Zyklus lang angezeigt, wenn der Zustand des Ausgangsparameters BUSY von "1" auf "0" wechselt. In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung "TM_MAIL" aktuell befindet bzw. wann das Senden der E-Mail beendet ist.

DONE	BUSY	ERROR	Beschreibung
0	1	0	Der Auftrag ist in Bearbeitung.
1	0	0	Der Auftrag wurde erfolgreich abgeschlossen.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache entnehmen Sie den Parametern STATUS und SFC_STATUS.
0	0	0	Der Anweisung "TM_MAIL" wurde kein (neuer) Auftrag erteilt.

Wechselt die CPU in den Betriebszustand STOP, während "TM_MAIL" aktiv ist, so wird die Kommunikationsverbindung zum Mail-Server abgebrochen. Die Kommunikationsverbindung zum Mail-Server geht auch verloren, wenn am Industrial Ethernet-Bus

Kommunikationsprobleme auftreten. In diesen Fällen wird das Versenden der E-Mail unterbrochen und sie erreicht ihre Empfänger nicht.

ACHTUNG
Ändern von Anwenderprogrammen Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf die Aufrufe von "TM_MAIL" auswirken, nur ändern: • wenn die CPU im Betriebszustand "STOP" ist oder • wenn keine Mail gesendet wird (REQ = 0 und BUSY = 0). Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "TM_MAIL" oder Aufrufe der Instanz von "TM_MAIL" enthalten. Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den TPC/IP-Kommunikationsfunktionen über Industrial Ethernet geraten. Nach dem Übertragen der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

Datenkonsistenz

Der Eingangsparameter ADDR_MAIL_SERVER der Anweisung wird bei jedem Anstoß zum Versenden einer E-Mail von der Anweisung "TM_MAIL" neu übernommen. Bei einer Änderung während des Betriebs wird der "neue" Wert erst bei einem erneuten Anstoß einer E-Mail wirksam.

Die Parameter WATCH_DOG_TIME, TO_S, CC, FROM, SUBJECT, TEXT, ATTACHMENT, sowie ggf. USERNAME und PASSWORD werden hingegen während des Betriebs der Anweisung "TM_MAIL" von diesem übernommen und dürfen somit erst geändert werden, wenn der Auftrag beendet wurde (BUSY = 0)

Parametrierung des TS Adapter IE

Am TS Adapter IE müssen Sie die Parameter für ausgehende Rufe so eingeben, dass der TS Adapter IE eine Verbindung zum Einwahlserver Ihres Internet Service Providers aufbaut.

Wenn Sie für Verbindungsaufbau "bei Bedarf" einstellen, dann wird die Verbindung erst dann aufgebaut, wenn eine Mail versendet werden soll.

Bei einer analogen Modemverbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TM_MAIL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L oder Konstante	Steuerparameter REQUEST: aktiviert das Versenden einer E-Mail bei steigender Flanke.
ID	Input	CONN_OUC (Word)	D, L oder Konstante	Referenz auf die aufzubauende Verbindung. Siehe Parameter ID der Anweisungen TCON (Seite 3689) , TDISCON (Seite 3700) , TSEND (Seite 3705) und TRCV (Seite 3709). Hier muss eine Nummer eingegeben werden, die in keiner weiteren Instanz dieser Anweisungen im Anwenderprogramm verwendet wird.
TO_S (Seite 4366)	Input	STRING	D	Eingangsparameter Empfängeradressen: STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).
CC (Seite 4366)	Input	STRING	D	Eingangsparameter CC-Empfängeradressen (optional): STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel). Wird hier ein Leerstring zugewiesen, wird die E-Mail nicht an einen CC-Empfänger gesendet.
SUBJECT	Input	STRING	D	Eingangsparameter Betreff der E-Mail: STRING mit einer maximalen Länge von 240 Zeichen.
TEXT	Input	STRING	D	Eingangsparameter Text der E-Mail (optional): Referenz auf einen Daten-String mit einer maximalen Länge von 240 Zeichen. Wird an diesem Parameter ein Leerstring zugewiesen, wird die E-Mail ohne Text versendet.
ATTACHMENT	Input	VARIANT	E, A, M, D, L	Eingangsparameter Anhang der E-Mail (optional): Referenz auf ein Byte-/Wort-/Doppelwort-Feld mit einer maximalen Länge von 65534 Bytes. Der angegebene Anhang muss im Dateiformat BIN sein. Wird kein Wert zugewiesen, wird die E-Mail ohne Anhang versendet.
DONE	Output	BOOL	E, A, M, D, L	• DONE = 0: Auftrag ist noch nicht gestartet oder wird noch ausgeführt. • DONE = 1: Auftrag wurde fehlerfrei ausgeführt.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Das Senden der E-Mail ist noch nicht abgeschlossen. • BUSY = 0: Die Bearbeitung von "TM_MAIL" wurde beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS und SFC_STATUS liefern detaillierte Auskunft über die Art des Fehlers.
STATUS (Seite 4367)	Output	WORD	E, A, M, D, L	Ausgangs-/Zustandsparameter STATUS: Rückgabewert bzw. Fehlerinformation der Anweisung "TM_MAIL".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ADDR_MAIL_SERVER	Static*	DWORD	E, A, M, D, L	Eingangsparameter IP-Adresse des Mail-Servers: Anzugeben als Datenwort im HEX-Format, z.B.: IP-Adresse = 192.168.0.200. ADDR_MAIL_SERVER = DW#16#C0A800C8, wobei: • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 und • 200 = 16#C8 entspricht.
WATCH_DOG_TIME	Static*	TIME	E, A, M, D, L	Eingangsparameter max. Zeitspanne: In der vorgegebenen Zeit (WATCH_DOG_TIME) sollte die Anweisung "TM_MAIL" eine Verbindung aufbauen. Wird diese Zeit überschritten, so wird der Baustein mit einem Fehler beendet. Die Zeit, bis der Baustein beendet und der Fehler ausgegeben wird, kann die WATCH_DOG_TIME überschreiten, da der Verbindungsabbau ebenso Zeit beansprucht. Zu Anfang sollten Sie eine Zeit von 2 Minuten einstellen. Bei der ISDN-Telefonverbindung kann diese Zeit deutlich kleiner gewählt werden.
USERNAME	Static*	STRING	D	Eingangsparameter Benutzername: STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Benutzername zwingend erforderlich.
PASSWORD	Static*	STRING	D	Eingangsparameter Passwort: STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Passwort zwingend erforderlich.
FROM (Seite 4366)	Static*	STRING	D	Eingangsparameter Absenderadresse: STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).
SFC_STATUS (Seite 4367)	Static*	WORD	E, A, M, D, L	Ausgangs-/Zustandsparameter "SFC_STATUS": Fehlerinformation der aufgerufenen Kommunikations-Bausteine.
* Die Werte der Parameter werden nicht bei jedem Aufruf der Anweisung "TM_MAIL" geändert. Die Werte liegen in den statischen Parametern der Instanz und werden nur einmal vor dem ersten Aufruf der Anweisung beschrieben.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Optionale Parameter

Die optionalen Parameter CC, TEXT und ATTACHMENT werden nur mit der E-Mail versendet, wenn die die entsprechenden Parameter einen String der Länge > 0 enthalten.

Hinweis

Parameter ATTACHMENT

Der angegebene Anhang muss im Dateiformat BIN sein. Wenn Text übermittelt werden soll, haben Sie folgende Möglichkeiten:

- Sie verknüpfen in Windows das Dateiformat BIN mit einem Texteditor.
- Sie verwenden die Windows-Funktionen "Senden an" oder "Öffnen mit".

SMTP-Authentisierung

Unter Authentisierung versteht man ein Verfahren zur Sicherstellung einer Identität z.B. durch eine Passwortabfrage.

Die Anweisung "TM_MAIL" unterstützt das SMTP-Authentisierungsverfahren AUTH-LOGIN, das von den meisten Mail-Servern gefordert wird. Informationen über die Authentisierungsverfahren Ihres Mail-Servers entnehmen Sie bitte dem Handbuch des Mail-Servers oder der Website Ihres Internet Service Providers.

Um das Authentisierungsverfahren AUTH-LOGIN zu nutzen benötigt die Anweisung "TM_MAIL" den Benutzernamen, mit dem er sich beim Mail-Server anmelden kann. Dieser Benutzername entspricht dem Nutzernamen mit dem Sie ein Mailkonto auf Ihrem Mail-Server eingerichtet haben. Er wird über den Parameter USERNAME der Anweisung "TM_MAIL" bekannt gegeben.

Des Weiteren benötigt die Anweisung "TM_MAIL" zur Anmeldung das dazugehörige Passwort. Dieses Passwort entspricht dem Passwort, das Sie beim Einrichten Ihres Mailkontos vergeben haben. Es wird über den Parameter PASSWORD der Anweisung "TM_MAIL" bekannt gegeben.

Der Benutzername und das Passwort werden jeweils unverschlüsselt (BASE64-kodiert) an den Mailserver übertragen.

Wird im DB kein Benutzername angegeben, so wird das Authentisierungsverfahren AUTH-LOGIN auch nicht verwendet. Die E-Mail wird dann ohne Authentisierung versendet.

Beispiel

Ein ausführliches Applikationsbeispiel finden Sie im Siemens Industry Online Support unter folgender FAQ ID: 67262019 (https://support.industry.siemens.com/cs/de/de/view/67262019).

Parameter TO_S, CC und FROM

Beschreibung

Die Parameter TO_S, CC und FROM sind Strings mit z. B. folgendem Inhalt:

- TO: <wenna@mydomain.com>, <ruby@mydomain.com>,
- CC: <admin@mydomain.com>, <judy@mydomain.com>,
- FROM: <admin@mydomain.com>

Folgende Regeln sind bei der Eingabe der Parameter zu beachten:

- Die Zeichen "TO:", "CC:" und "FROM:" müssen eingegeben werden.
- Vor jeder Adresse muss ein Leerzeichen und eine spitze Klammer auf "<" eingegeben werden.
- Nach jeder Adresse muss eine spitze Klammer zu ">" eingegeben werden.
- Nach jeder Adresse in TO und CC muss ein Komma eingegeben werden.
- Unter FROM darf nur eine E-Mail Adresse eingetragen werden, bei dieser darf kein Komma am Ende stehen

Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "TM_MAIL" keine Syntaxprüfung der Parameter TO_S, CC und FROM durch.

Parameter STATUS und SFB_STATUS

Beschreibung

Die Rückgabewerte der Anweisung "TM_MAIL" können folgendermaßen klassifiziert werden:

- W#16#0000: "TM_MAIL" wurde erfolgreich beendet
- W#16#7xxx: Status von "TM_MAIL"
- W#16#8xxx: beim internen Aufruf eines Kommunikations-Bausteins oder vom Mail-Server wurde ein Fehler gemeldet.

Die folgende Tabelle zeigt die Rückgabewerte von "TM_MAIL" mit Ausnahme der Fehlercodes der intern aufgerufenen Kommunikations-Bausteine.

Rückgabewert STATUS* (W#16#...):	Rückgabewert SFB_STATUS (W#16#...):	Erläuterung	Hinweise
0000	-	Die Bearbeitung von "TM_MAIL" wurde fehlerfrei abgeschlossen.	Ein fehlerfreies Beenden von "TM_MAIL" bedeutet nicht, dass die versendete E-Mail ankommt (siehe unten - Hinweis Punkt 1)
7001		"TM_MAIL" ist aktiv (BUSY = 1).	Erstaufruf; Auftrag angestoßen
7002	7002	"TM_MAIL" ist aktiv (BUSY = 1).	Zwischenaufruf; Auftrag ist bereits aktiv
8xxx	xxxx	Die Bearbeitung von "TM_MAIL" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen abgeschlossen.	Ausführliche Informationen zur Auswertung des Parameters SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Kommunikations-Anweisungen.
8010	xxxx	Fehler beim Verbindungsaufbau.	Weitere Informationen zur Auswertung des Parameters SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TCON (Seite 3689)".

Rückgabewert STATUS* (W#16#...):	Rückgabewert SFB_STATUS (W#16#...):	Erläuterung	Hinweise
8011	xxxx	Fehler beim Senden der Daten.	Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TSEND (Seite 3705)".
8012	xxxx	Fehler beim Empfangen der Daten.	Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TRCV (Seite 3709)".
8013	xxxx	Fehler beim Verbindungsaufbau.	Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TCON (Seite 3689)" und "TDISCON (Seite 3700)".
8014	-	Aufbau einer Verbindung nicht möglich.	Möglicherweise haben Sie eine fehlerhafte Mail-Server-IP-Adresse (ADDR_MAIL_SERVER) oder eine zu kleine Zeitspanne (WATCH_DOG_TIME) für den Verbindungsaufbau eingegeben. Es besteht auch die Möglichkeit, dass die CPU keine Verbindung zum Netzwerk hat oder die CPU-Konfiguration fehlerhaft ist.
8016	-	Fehler beim Kopieren des Anhangs	-
82xx, 84xx, bzw. 85xx	-	Die Fehlermeldung stammt vom Mail-Server und entspricht bis auf die "8" der Fehlernummer des SMTP-Protokolls. In den nachfolgenden Spalten sind einige Fehlercodes aufgeführt, die auftreten können:	Siehe Punkt 2 im Hinweis.
8450	-	Aktion nicht ausgeführt: Mailbox nicht verfügbar/nicht erreichbar.	Versuchen Sie es später noch einmal.
8451	-	Aktion abgebrochen: Lokaler Fehler in der Verarbeitung	Versuchen Sie es später noch einmal.
8500	-	Syntax-Fehler: Fehler nicht erkannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann auch sein, dass der E-Mail-Server das Authentisierungs-verfahren LOGIN nicht unterstützt.	Überprüfen Sie die Parameter des "TM_MAIL". Versuchen Sie, eine E-Mail ohne Authentisierung zu versenden. Ersetzen Sie dazu den Parameter USERNAME durch einen leeren String.
8501	-	Syntax-Fehler: Parameter oder Argument falsch	Möglicherweise haben Sie eine fehlerhafte Adresse in TO_S oder CC eingegeben.
8502	-	Befehl unbekannt bzw. nicht implementiert.	Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" oder "." vergessen.

Rückgabewert STATUS* (W#16#...):	Rückgabewert SFB_STATUS (W#16#...):	Erläuterung	Hinweise
8535	-	SMTP-Authentisierung unvollständig.	Eventuell haben Sie einen fehlerhaften Benutzernamen oder ein fehlerhaftes Passwort eingegeben.
8550	-	Mail-Server kann nicht erreicht werden, Sie haben keine Zugriffsrechte.	Möglicherweise haben Sie einen fehlerhaften Benutzernamen bzw. Passwort eingegeben oder der Mail-Server unterstützt Ihr LOGIN nicht. Eine weitere Fehlerursache könnte eine fehlerhafte Eingabe des Domainnamens nach dem "@" in TO_S bzw. CC sein.
8552	-	Aktion abgebrochen: Überschreitung der zugewiesenen Speichergröße	Versuchen Sie es später noch einmal.
8554	-	Übertragung misslungen.	Versuchen Sie es später noch einmal.
* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".			

Hinweis

Statusfehler

1. Eine fehlerhafte Eingabe der Empfängeradressen erzeugt keinen Statusfehler der Anweisung "TM_MAIL". Das Versenden der E-Mail an weitere Empfänger, auch wenn diese richtig eingegeben wurden, ist in diesem Fall nicht sichergestellt.
2. Weitere Informationen zum SMTP-Fehlercode und weitere Fehlercodes finden Sie im SMTP-Protokoll im Internet bzw. in der Fehlerdokumentation des Mail-Servers. Darüber hinaus können Sie die letzte, vom Mail-Server gesendete Fehlermeldung in Textform in Ihrem Instanz-DB im Parameter BUFFER1 einsehen. Dort finden Sie unter "Daten" die zuletzt von der Anweisung "TM_MAIL" versendeten Daten.

Siehe auch

Anzeigeformate im Programmstatus umstellen (Seite 8040)

4.2 Anweisungen

4.2.1 Asynchron arbeitende Anweisungen

4.2.1.1 Unterschied zwischen synchron und asynchron arbeitenden Anweisungen

Asynchron arbeitende Anweisungen

In der Programmbearbeitung wird zwischen synchron und asynchron arbeitenden Anweisungen unterschieden.

Die Eigenschaften "synchron" bzw. "asynchron" beziehen sich auf den zeitlichen Zusammenhang zwischen Aufruf und Ausführung der Anweisung:

- Für synchrone Anweisungen gilt: Wenn der Aufruf einer synchron arbeitenden Anweisung beendet ist, ist auch die Ausführung beendet.
- Anders bei asynchronen Anweisungen: Die Ausführung einer asynchronen Anweisung kann sich über mehrere Aufrufe erstrecken. Die CPU bearbeitet asynchrone Anweisungen parallel zum zyklischen Anwenderprogramm. Asynchron arbeitende Anweisungen belegen während ihrer Bearbeitung Ressourcen in der CPU.

Bei asynchron arbeitenden Anweisungen handelt es sich in der Regel um Anweisungen für die Übertragung von Daten (Datensätze für Module, Kommunikationsdaten, Diagnosedaten, ...).

Weitere Informationen zu asynchron arbeitenden Anweisungen finden Sie hier: SIMATIC S7-1500 / ET 200MP Manual Collection (https://support.industry.siemens.com/cs/ww/de/view/86140384) im Systemhandbuch unter Grundlagen zur Programmbearbeitung > Asynchron arbeitende Anweisungen.

Identifikation des Auftrags

Um eine Anweisung über mehrere Aufrufe auszuführen, muss die CPU einen Folgeaufruf einem bereits laufenden Auftrag der Anweisung eindeutig zuordnen können. Für die Zuordnung Aufruf zu Auftrag nutzt die CPU, abhängig vom Typ der Anweisung, einen der beiden folgenden Mechanismen:

- Über die Instanz der Anweisung (bei Typ "SFB" Systemfunktionsbaustein)
- Über den Auftrag identifizierende Eingangsparameter der Anweisung

Falls Sie mit asynchron arbeitenden Anweisung die Auslösung eines Prozessalarms oder die Ausgabe von Steuerkommandos an DP-Slaves oder eine Datenübertragung oder den Abbruch einer nichtprojektierten Verbindung angestoßen haben und Sie diese Anweisung erneut aufrufen, bevor der laufende Auftrag beendet wurde, hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt.

Übergabe von Parametern an asynchron auszuführende Programmbausteine

Codebausteine (FB/FC) und Datenbausteine (DB) können mit unterschiedlicher Zugriffsart erstellt werden ("Standard" und "Optimiert"). In den Codebausteinen können Sie beliebige Anweisungen aufrufen. Einige Anweisungen (z. B. "WRIT_DBL" und "READ_DBL") werden asynchron ausgeführt. Diese Programmbausteine können nicht mit Variablen aus TEMP versorgt werden, da die Daten während der Ausführung nicht geändert werden dürfen.

Stellen Sie sicher, dass Sie diese Anweisungen nicht in Programmen verwenden, in denen sich Codebausteine mit unterschiedlicher Zugriffsart gegenseitig aufrufen, so dass folgende Fälle eintreten würden:

- Eine Struktur aus einem Standard-Datenbaustein wird unmittelbar oder mittelbar an einen optimierten Codebaustein übergeben, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Bausteine weitergibt.
- Der umgekehrte Fall, dass eine Struktur aus einem optimierten Codebaustein unmittelbar oder mittelbar an einen Standard-Datenbaustein übergeben wird, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Programmbausteine weitergibt.

Andernfalls wird eine versteckte Kopie der übergebenen Daten in TEMP angelegt, die dann dazu führt, dass die asynchron arbeitenden Programmbausteine eine negative Quittung liefern.

Parameter REQ

Der Eingangsparameter REQ (request) dient ausschließlich dem Anstoß des Auftrags:

- Stoßen Sie den Auftrag an, indem Sie den Eingangsparameter REQ gleich "1" setzen (Fall 1).
- Ist ein bestimmter Auftrag angestoßen und noch nicht abgeschlossen und Sie rufen die Anweisung zum gleichen Auftrag erneut auf (z. B. in einem Weckalarm-OB), so wird REQ durch die Anweisung nicht ausgewertet (Fall 2).
- Ist der Auftrag beendet, aber beim nächsten Aufruf der Eingangsparameter REQ immer noch auf "1" gesetzt, so wird der Auftrag sofort wieder gestartet.

Hinweis

REQ-Bit = "1"

Bitte beachten Sie, dass bei gesetztem REQ-Bit die asynchrone Anweisung nach der Beendigung des vorherigen Aufrufs wieder, möglicherweise ungewollt, erneut gestartet wird. Für die bessere Wartbarkeit / Übersichtlichkeit des Programms ist es damit ratsam, das REQ-Bit so früh wie möglich wieder auf "0" zurückzusetzen.

Parameter RET_VAL und BUSY

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand der Auftragsausführung angezeigt.

Beachten Sie den Hinweis in dem Abschnitt: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

- Im Fall 1 (Erstaufruf mit REQ = 1) wird bei freien Systemressourcen und korrekter Versorgung der Eingangsparameter in RET_VAL der Code W#16#7001 eingetragen und BUSY wird auf "1" gesetzt.
  Falls die benötigten Systemressourcen momentan belegt sind oder ein Fehler in den Eingangsparametern vorliegt, wird in RET_VAL der zugehörige Fehlercode eingetragen und BUSY wird mit "0" beschrieben.
- Im Fall 2 (Zwischenaufruf) wird der Code W#16#7002 in RET_VAL eingetragen (Dies entspricht der Nachricht, dass der Auftrag derzeit noch bearbeitet wird.) und BUSY wird auf "1" gesetzt.
- Beim letzten Aufruf für einen Auftrag gilt:
  - Bei den Anweisungen "DPNRM_DG (Seite 5522)", "X_GET (Seite 7083)" und "I_GET (Seite 7066)" wird bei fehlerfreier Datenübertragung in RET_VAL die Anzahl gelieferter Daten in Bytes als positive Zahl eingetragen. BUSY wird in diesem Fall mit "0" beschrieben.
    Im Fehlerfall wird in RET_VAL die Fehlerinformation eingetragen. BUSY dürfen Sie in diesem Fall nicht auswerten.
  - Bei "RD_REC (Seite 5490)" wird bei fehlerfreier Datenübertragung in RET_VAL die Größe des Datensatzes in Bytes oder 0 eingetragen. BUSY wird in diesem Fall mit "0" beschrieben. Im Fehlerfall wird in RET_VAL der Fehlercode eingetragen und BUSY wird mit "0" beschrieben.
  - Bei allen anderen Anweisungen wird bei fehlerfreier Auftragsausführung in RET_VAL "0" eingetragen und BUSY wird mit "0" beschrieben. Im Fehlerfall wird in RET_VAL der Fehlercode eingetragen und BUSY wird mit "0" beschrieben.

Hinweis

Wenn der erste und der letzte Aufruf zusammenfallen, dann gilt für RET_VAL und BUSY das gleiche wie beim letzten Aufruf.

Vom Auftrag belegte Ressourcen

Jeder Auftrag belegt Ressourcen der CPU. Bei der Anweisung "RDREC" z. B. ist dies Speicherplatz für den Zielbereich. Diese Ressourcen sind exklusiv nur diesem Auftrag zugeordnet und sie sind solange belegt, wie die Anweisung aktiv ist, d. h. solange BUSY den Wert "1" hat.

Belegte Ressourcen dürfen Sie nicht anderweitig benutzen oder verändern. So dürfen Sie beispielsweise bei der Anweisung "RDREC" weder die Länge noch den Inhalt des Zielbereichs verändern.

Überblick

Die folgende Tabelle gibt Ihnen einen Überblick über die oben beschriebenen Zusammenhänge. Sie zeigt insbesondere die möglichen Werte der Ausgangsparameter an, falls die Auftragsausführung nicht nach einem Anweisungs-Aufruf abgeschlossen ist.

Hinweis

Sie müssen in Ihrem Programm nach jedem Aufruf die relevanten Ausgangsparameter bewerten.

Zusammenhang zwischen Aufruf, REQ, RET_VAL und BUSY bei einem "laufenden" Auftrag:

Lfd. Nr. des Aufrufs	Aufrufart	REQ	RET_VAL	BUSY	DONE	ERROR
1	erster Aufruf	1	W#16#7001	1	0	0
			Fehlercode	0	0	1
2 bis (n - 1)	Zwischenauf-ruf	irrelevant	W#16#7002	1	0	0
n	letzter Aufruf	irrelevant	W#16#0000 (Ausnahmen: "RD_REC (Seite 5490)", falls der Zielbereich größer ist als der übertragene Datensatz; "DPNRM_DG (Seite 5522)", "X_GET (Seite 7083)" und "I_GET (Seite 7066)", falls keine Fehler aufgetreten sind.)	0	1	0
			Fehlercode, falls Fehler aufgetreten sind.	0	0	1

4.2.2 Einfache Anweisungen

4.2.2.1 KOP

Bitverknüpfungen

---| |---: Schließerkontakt

Beschreibung

Die Aktivierung eines Schließerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Schließerkontakt geschlossen und der Signalzustand des Eingangs auf den Ausgang übertragen.

Wenn der Operand den Signalzustand "0" hat, wird der Schließerkontakt nicht aktiviert und der Signalzustand am Ausgang der Anweisung auf "0" gesetzt.

In einer Reihenschaltung werden zwei oder mehrere Schließerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Schließerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung.

Legen Sie dazu einen globalen Datenbaustein mit folgendem Inhalt an:

Name des Bausteins: SLI_gDB_NOContact	
Name	**Datentyp**
start1	BOOL
start2	BOOL
start3	BOOL
startOut	BOOL

Schreiben Sie folgenden Programmcode:

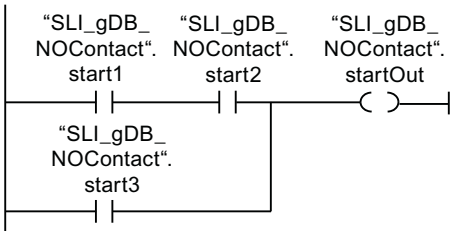

Der Operand "SLI_gDB_NOContact".startOut wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "SLI_gDB_NOContact".start1 und "SLI_gDB_NOContact".start2 haben den Signalzustand "1".
- Der Operand "SLI_gDB_NOContact".start3 hat den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---| / |---: Öffnerkontakt

Beschreibung

Die Aktivierung eines Öffnerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Öffnerkontakt geöffnet und der Signalzustand am Ausgang der Anweisung auf "0" zurückgesetzt.

Wenn der Operand den Signalzustand "0" hat, wird der Öffnerkontakt nicht aktiviert und der Signalzustand des Eingangs auf den Ausgang übertragen.

In einer Reihenschaltung werden zwei oder mehrere Öffnerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Öffnerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

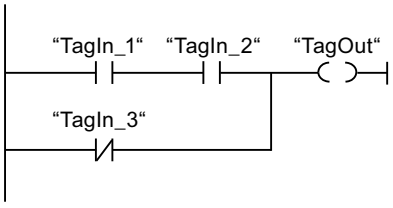

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" haben den Signalzustand "1".
- Der Operand "TagIn_3" hat den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

--|NOT|--: VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE). Wenn am Eingang der Anweisung der Signalzustand "1" ansteht, liefert der Ausgang der Anweisung den Signalzustand "0". Ist der Signalzustand am Eingang der Anweisung "0", liefert der Ausgang den Signalzustand "1".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

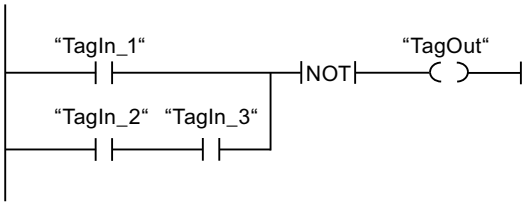

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Signalzustand an der Operanden "TagIn_2" und "TagIn_3" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---( )---: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" setzen Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Eingang der Spule "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Eingang der Spule wird direkt auf den Ausgang übertragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

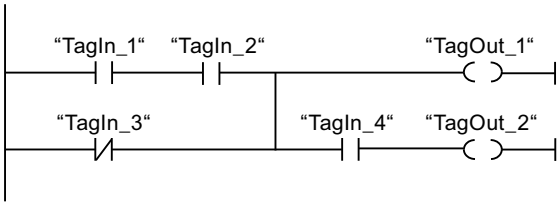

Der Operand "TagOut_1" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Der Operand "TagOut_2" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" sowie der Operand "TagIn_4" liefern den Signalzustand "1"
- Der Signalzustand des Operanden "TagIn_3" ist "0" und der des Operanden "TagIn_4" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---(R)---: Ausgang rücksetzen

Beschreibung

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "0" zurückgesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfluss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L, T, Z	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

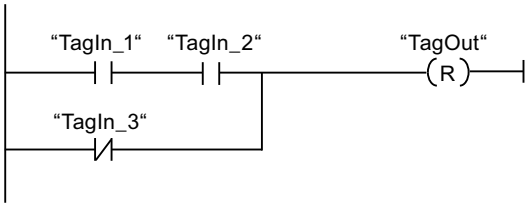

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

---(S)---: Ausgang setzen

Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfluss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

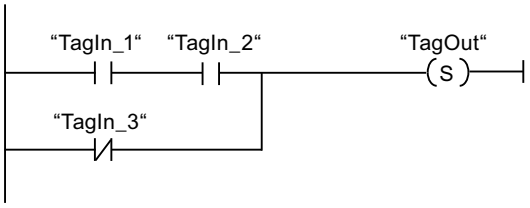

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

SR: Flipflop setzen/rücksetzen

Beschreibung

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S	Input	BOOL	E, A, M, D, L	Setzen freigeben
R1	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzen freigeben

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	InOut	BOOL	E, A, M, D, L	Operand, der gesetzt oder zurückgesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

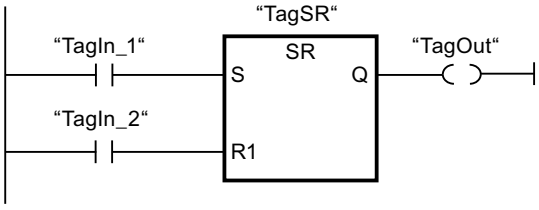

Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagSR" und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

---(MCR<): MCR-Bereiche öffnen (Seite 4530)

Grundlagen zu KOP (Seite 7425)

RS: Flipflop rücksetzen/setzen

Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen R und S1 wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R	Input	BOOL	E, A, M, D, L	Rücksetzen freigeben
S1	Input	BOOL	E, A, M, D, L, T, Z	Setzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	Operand, der zurückgesetzt oder gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

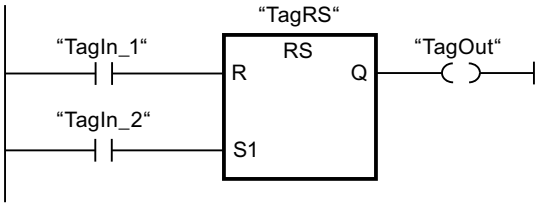

Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

---(MCR<): MCR-Bereiche öffnen (Seite 4530)

Grundlagen zu KOP (Seite 7425)

--|P|--: Operand auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

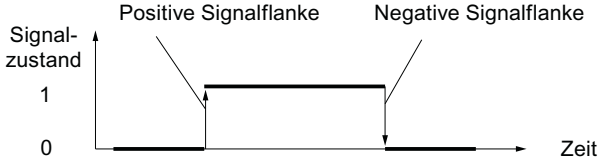

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BOOL	E, A, M, D, L, T, Z	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

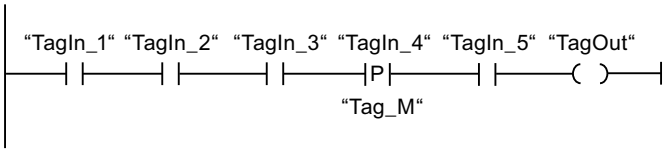

Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" liefern den Signalzustand "1".
- Am Operanden "TagIn_4" liegt eine steigende Flanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag_M" gespeichert.
- Der Signalzustand des Operanden "TagIn_5" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

--|N|--: Operand auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker <Operand2> gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

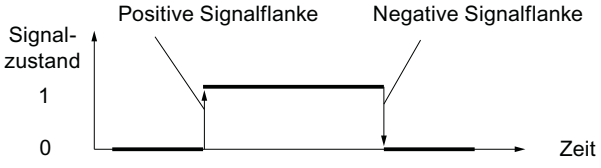

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie im Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie im Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BOOL	E, A, M, D, L, T, Z	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

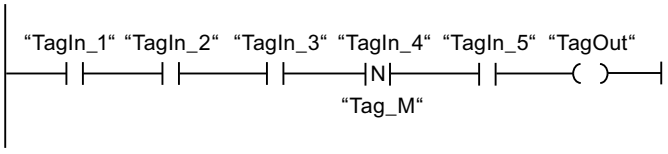

Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" liefern den Signalzustand "1".
- Am Operanden "TagIn_4" liegt eine negative Signalflanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag_M" gespeichert.
- Der Signalzustand des Operanden "TagIn_5" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

P_TRIG: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine positive Signalflanke erfasst wird. In allen anderen Fällen liefert der Ausgang den Signalzustand "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

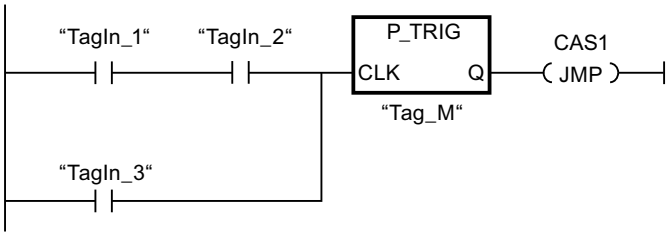

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

N_TRIG: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine negative Signalflanke erfasst wird. In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

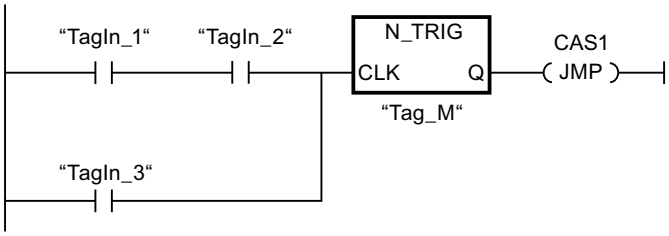

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

Zeiten

IEC-Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TP (z. B. "TP_DB")
- Deklaration als lokale Variable vom Typ TP im Abschnitt "Static" eines Bausteins (z. B. #MyTP_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Impulsausgang
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

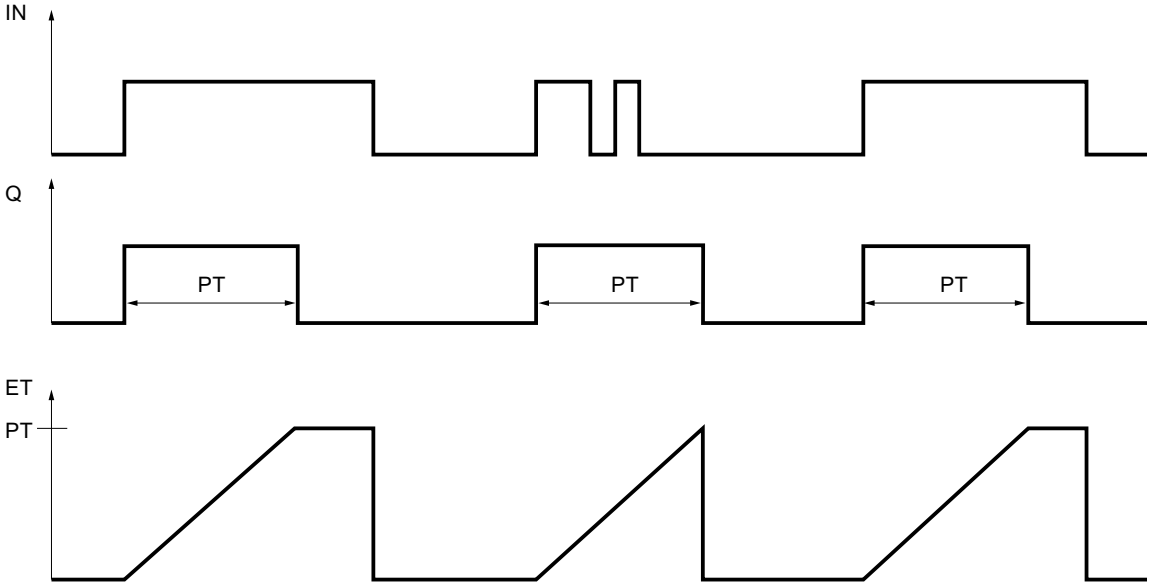

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TON (z. B. "TON_DB")
- Deklaration als lokale Variable vom Typ TON im Abschnitt "Static" eines Bausteins (z. B. #MyTON_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

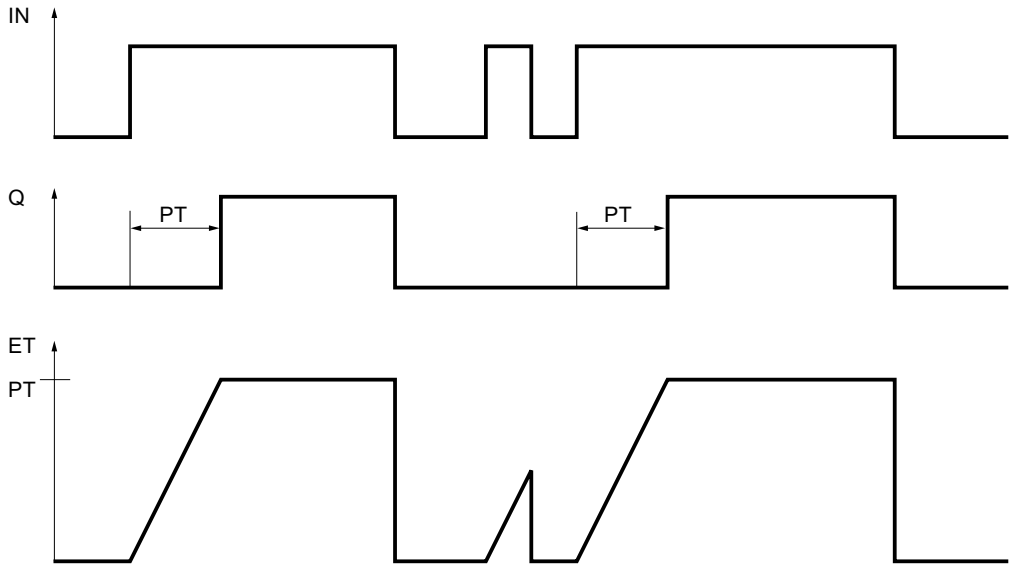

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der aktuelle Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang

IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TOF (z. B. "TOF_DB")
- Deklaration als lokale Variable vom Typ TOF im Abschnitt "Static" eines Bausteins (z. B. #MyTOF_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

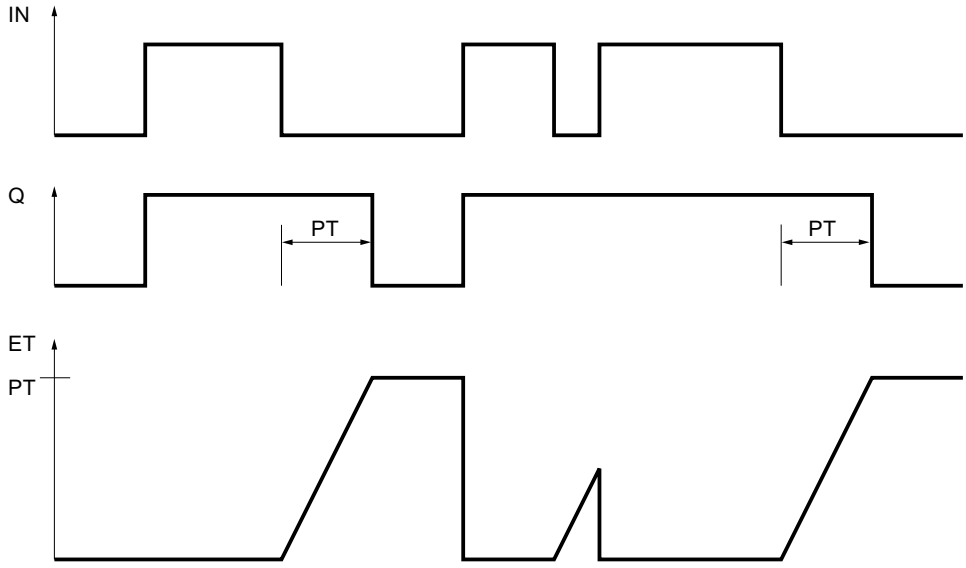

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

SIMATIC-Zeiten

S_IMPULS: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte

Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

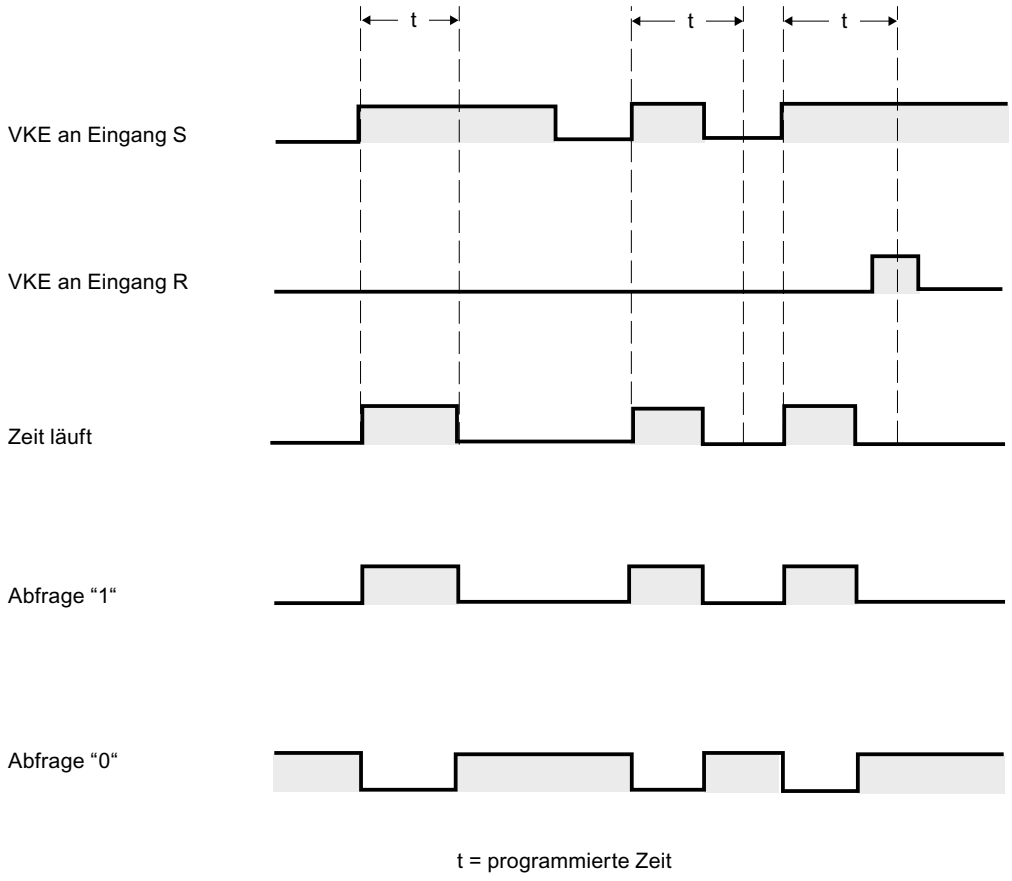

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

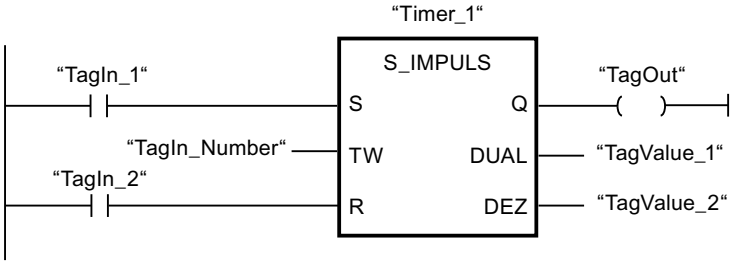

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Operand "TagIn_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

S_VIMP: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

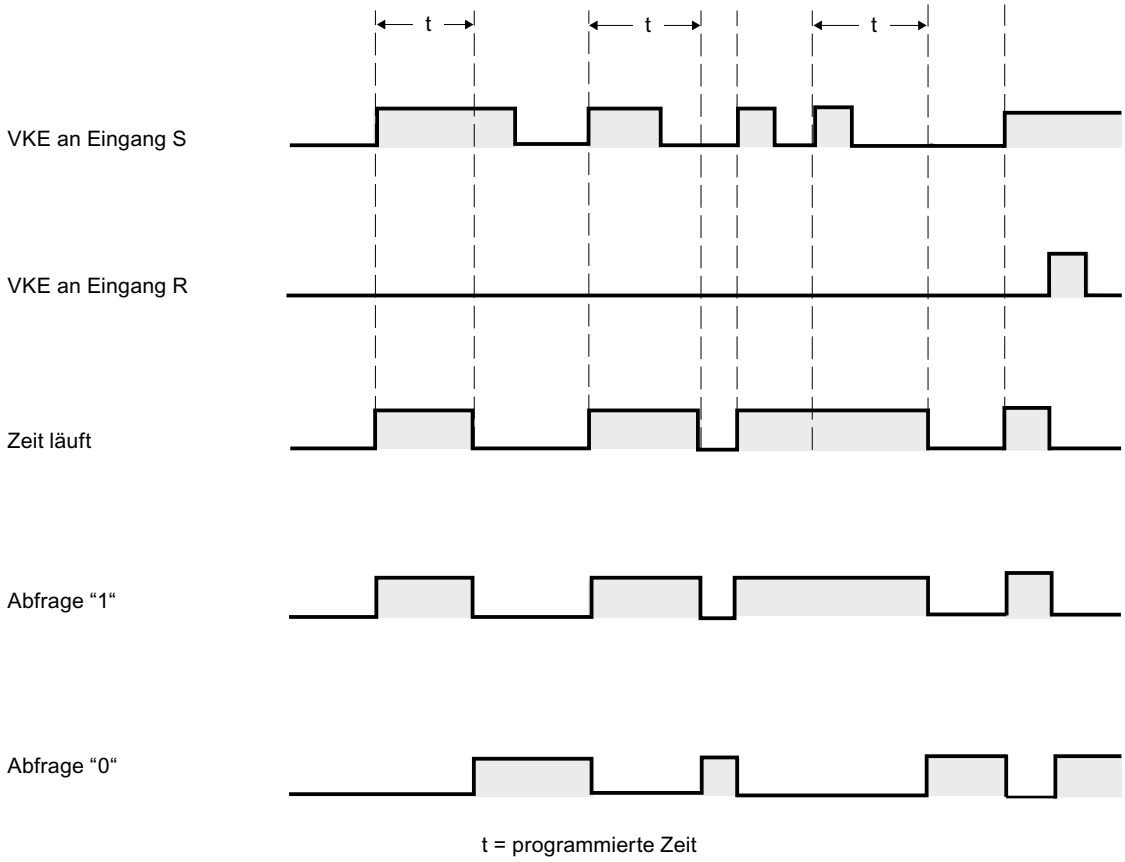

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

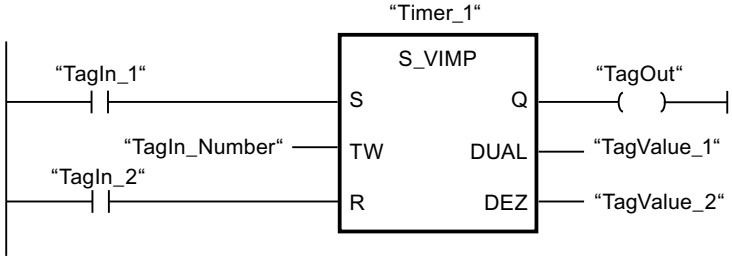

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Zeit der Anweisung geben Sie in den Platzhalter oberhalb der Box an. Die Zeit muss mit dem Datentyp TIMER deklariert sein.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

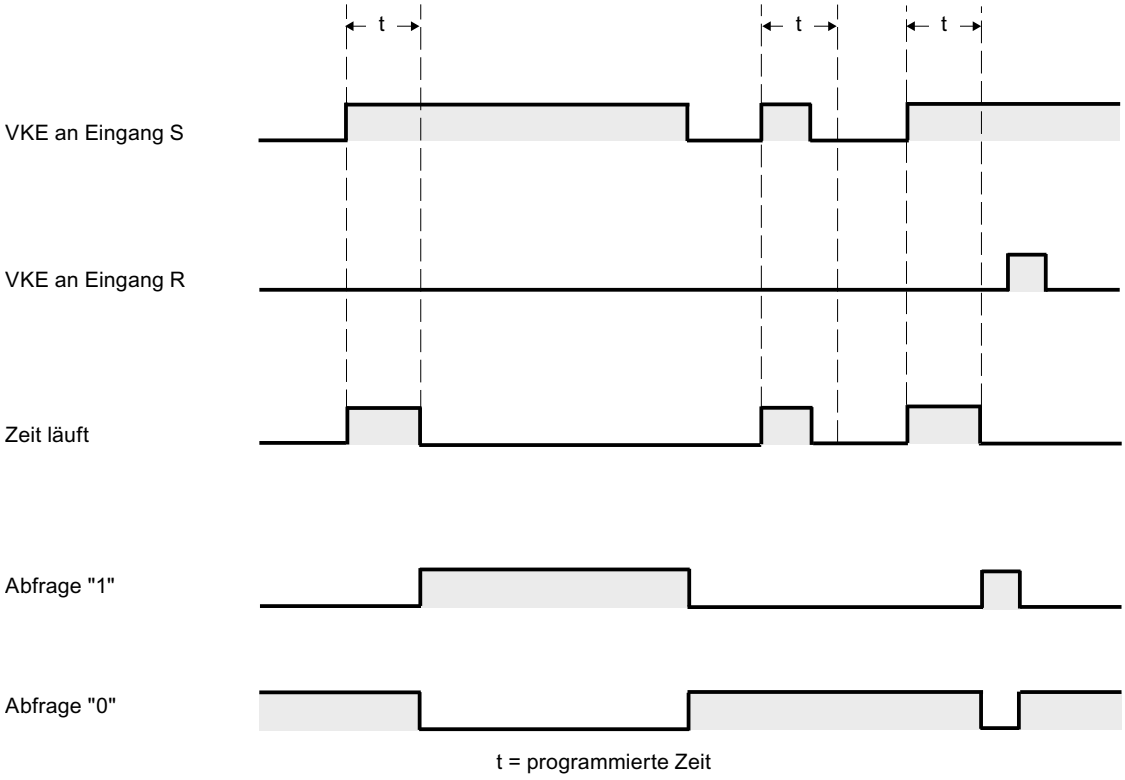

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

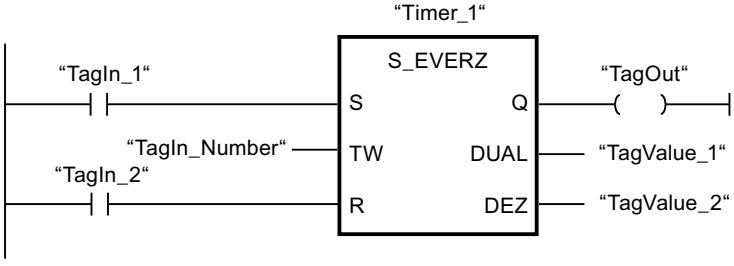

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

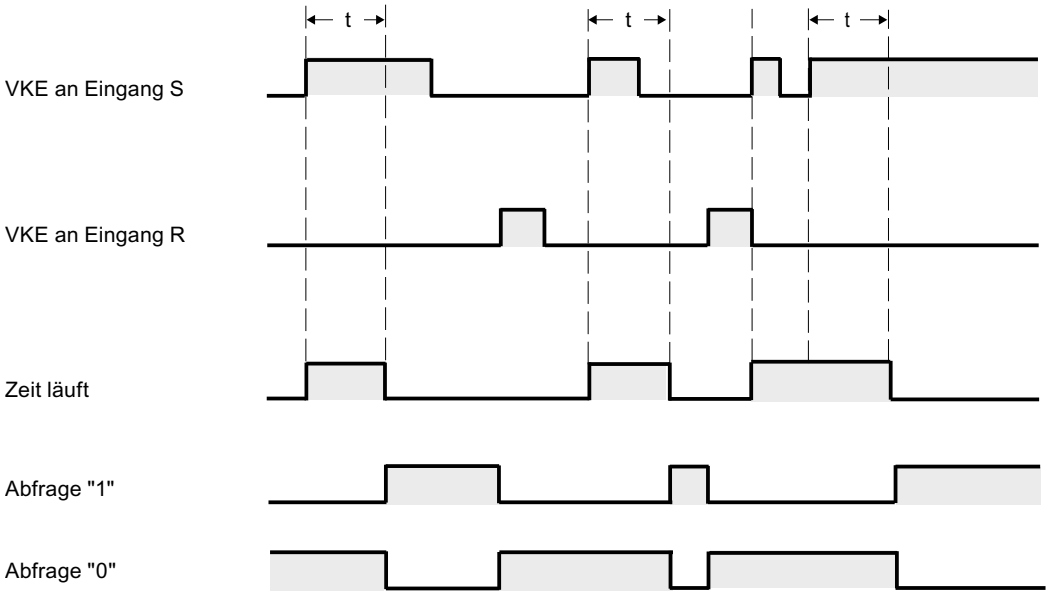

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

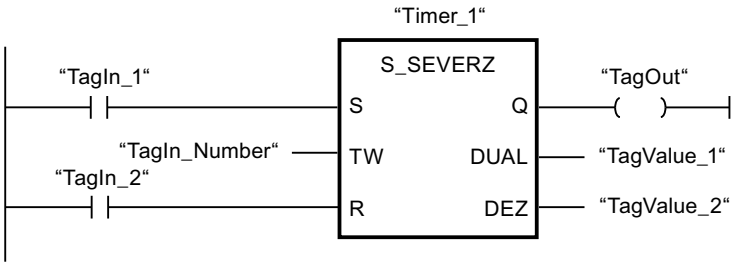

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, auch wenn der Signalzustand des Operanden "TagIn_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand am Eingang S "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

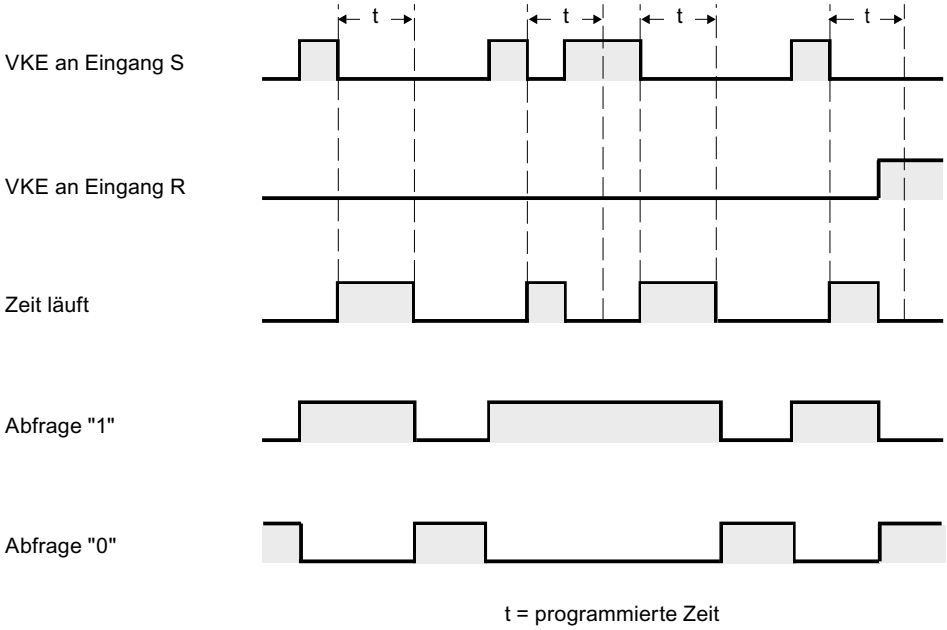

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

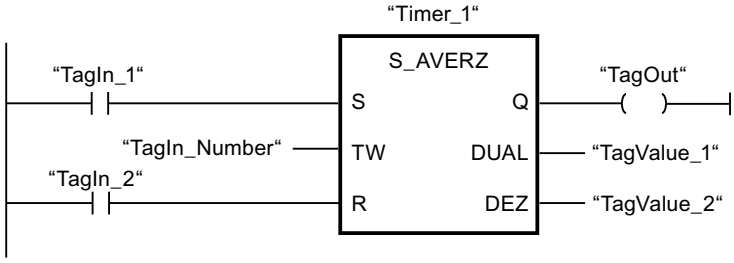

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

---(SI): Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

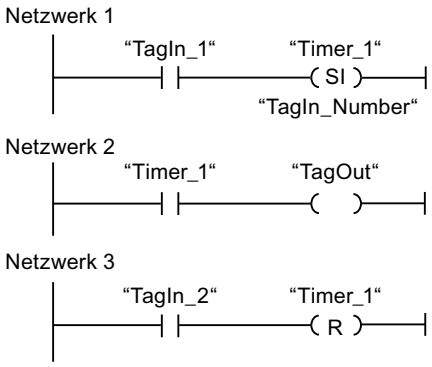

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Signalzustand des Operanden "TagIn_1" "1" ist. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_2" von "0" auf "1", dann wird die Zeit zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

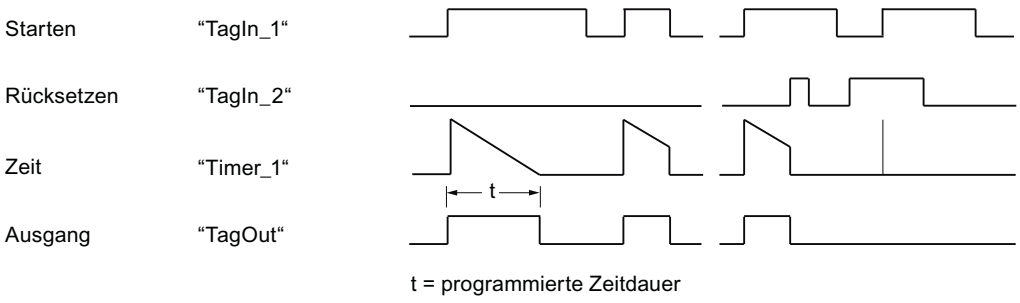

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu KOP (Seite 7425)

---(SV): Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer neu gestartet. Die Abfrage des Zeitstatus auf "1" ergibt den Signalzustand "0", wenn die Zeit abgelaufen ist.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als verlängerten Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

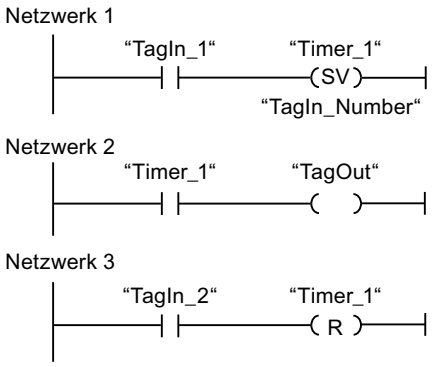

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer negativen Signalflanke im VKE beeinflusst zu werden. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_1" erneut von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

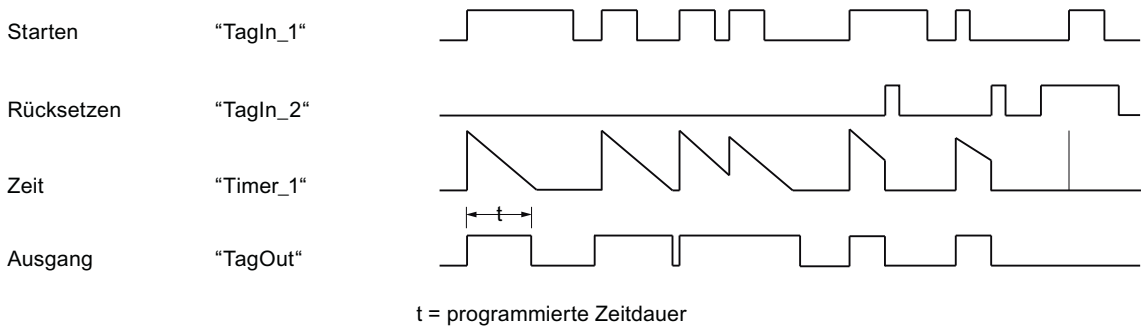

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SE): Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der angegebenen Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

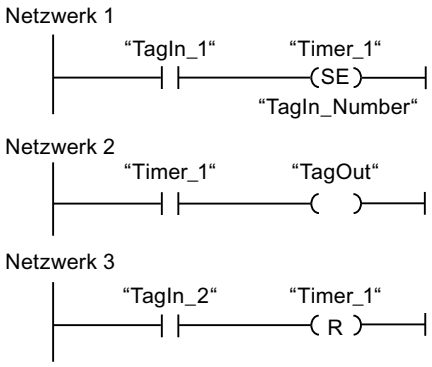

Netzwerk 1:

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab. Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit zurückgesetzt.

Netzwerk 2:

Wenn die Zeit abgelaufen ist, der Operand #TagIn_1 am Starteingang den Signalzustand "1" hat und die Zeit nicht zurückgesetzt wurde, dann ist der Operand #TagOut "1".

Netzwerk 3:

Wenn der Signalzustand des Operanden #TagIn_2 den Signalzustand "1" hat, dann wird die Zeit #Timer_1 und der Ausgang #TagOut zurückgesetzt.

Wenn Sie den #Timer_1 erneut starten wollen, dann muss der Signalzustand am Operand #TagIn_2 "0" sein und der Signalzustand am Starteingang #TagIn_1 von "0" auf "1" wechseln.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

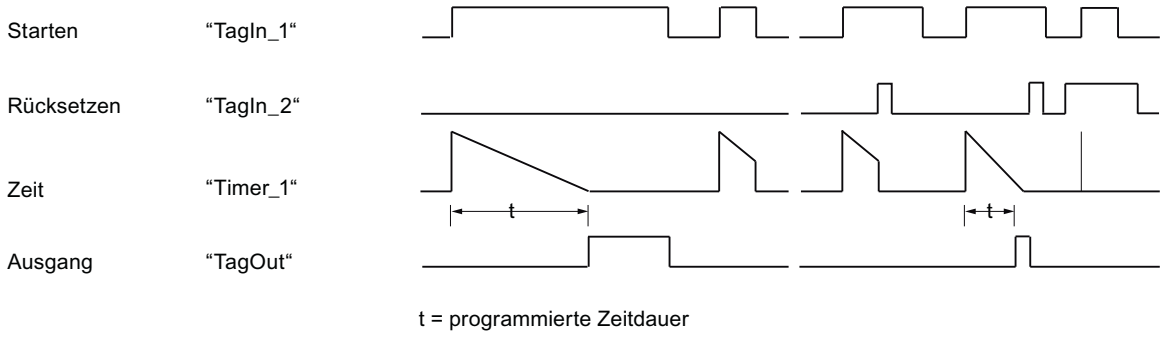

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SS): Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

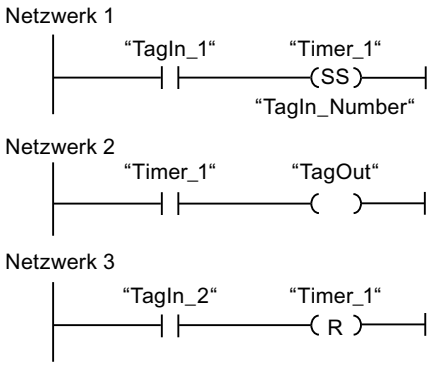

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

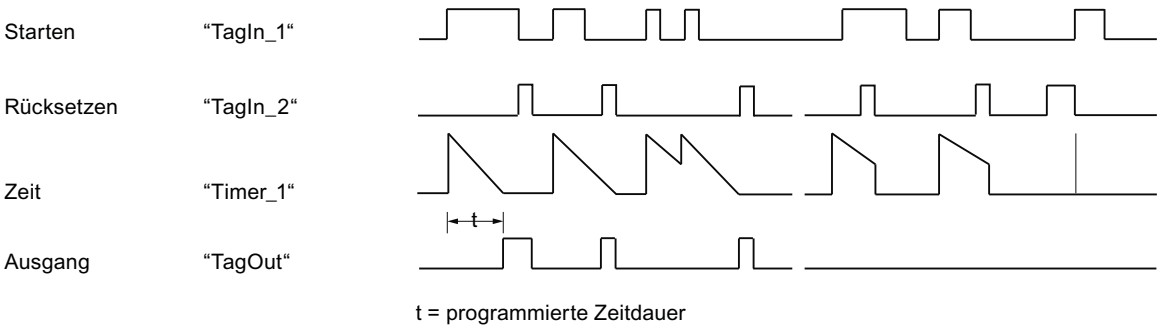

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(SA): Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (Negative Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist. Wenn das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeitdauer>	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer, mit der die Zeit abläuft.
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

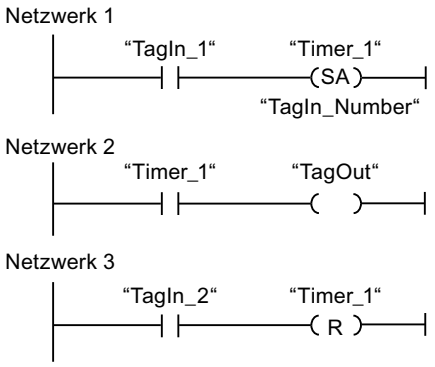

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Solange die Zeit läuft, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

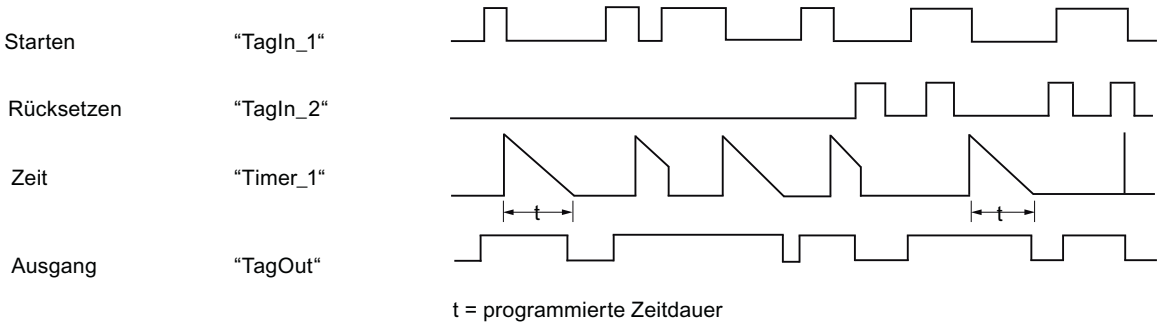

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Zähler

IEC-Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf Null zurückgesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTU (z. B. "CTU_DB")
- Deklaration als lokale Variable vom Typ CTU im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

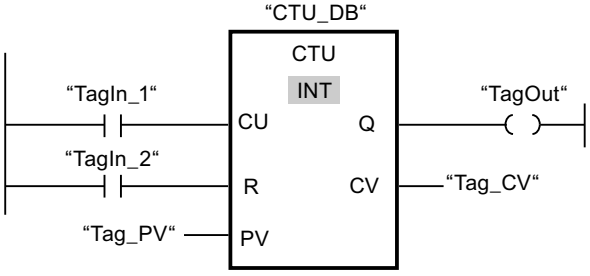

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CV" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps (INT) erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Eingang LD auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTD (z. B. "CTD_DB")
- Deklaration als lokale Variable vom Typ CTD im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

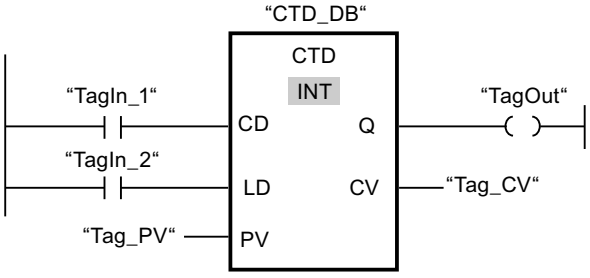

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert am Ausgang "Tag_CV" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des Datentyps (INT = -32768) erreicht ist.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hochgezählt. Wenn der untere Grenzwert des angegebenen Datentyps (INT) erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0".

Am Ausgang QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTUD (z. B. "CTUD_DB")
- Deklaration als lokale Variable vom Typ CTUD im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein

sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status der Vorwärtszähler
QD	Output	BOOL	E, A, M, D, L	Status der Rückwärtszähler
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

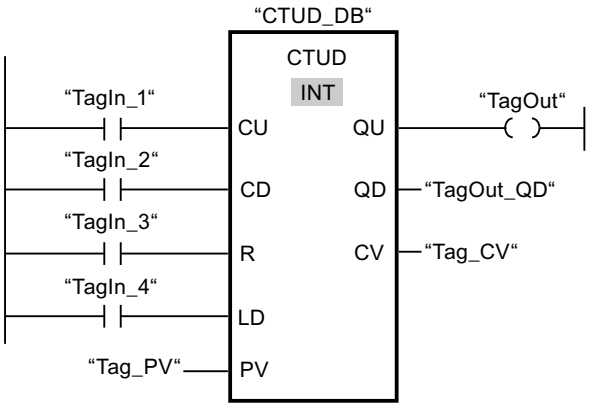

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Der Ausgang "TagOut_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut_QD" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

Instanzen (Seite 64)

SIMATIC-Zähler

Z_VORW: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

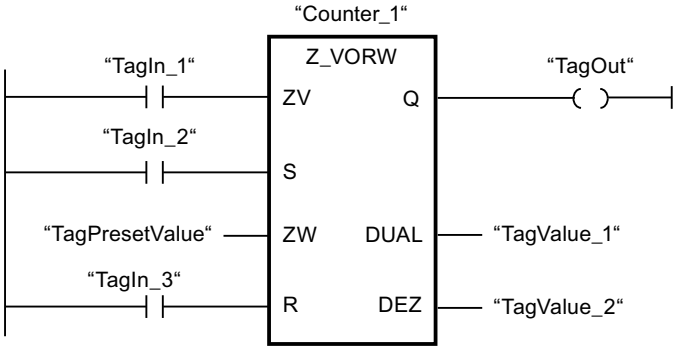

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

Z_RUECK: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZR	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

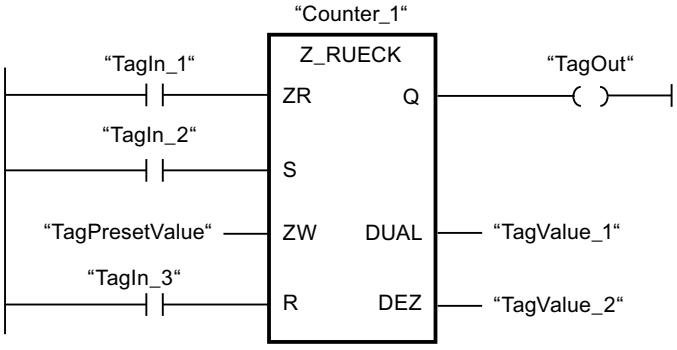

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr

inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
ZR	Input	BOOL	E, A, M, D, L, T, Z	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

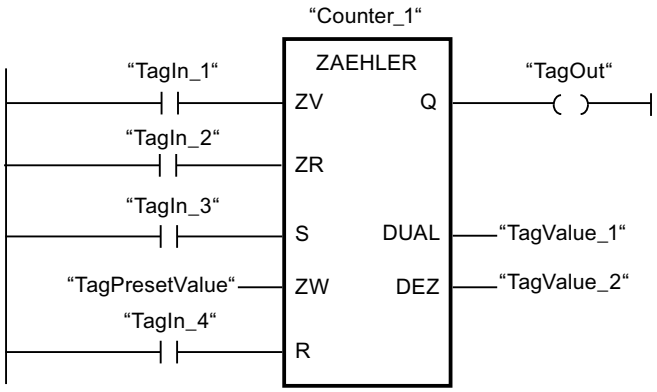

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

---(SZ): Zähleranfangswert setzen

Beschreibung

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zählwert>	Input	WORD	E, A, M, D, L oder Konstante	Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999)
<Zähler>	InOut/Input	COUNTER	Z	Zähler, der voreingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

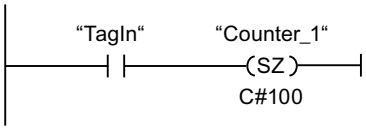

Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

---(ZV): Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

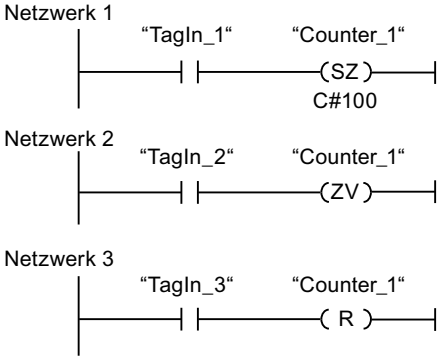

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

---(ZR): Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

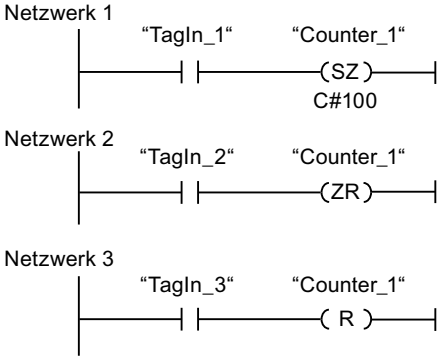

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu KOP (Seite 7425)

Vergleicher

CMP ==: Gleich

Beschreibung

Mit der Anweisung "Gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann verwenden Sie statt der Anweisung "CMP ==: Gleich" die Anweisung "IN_RANGE: Wert innerhalb Bereich".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

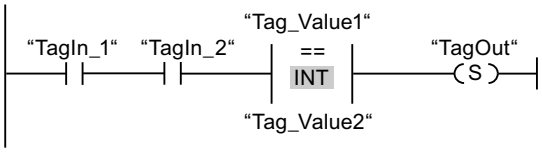

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" = "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

CMP <>: Ungleich

Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) ungleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

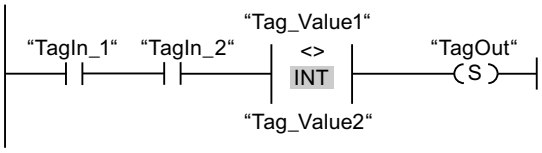

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <> "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu KOP (Seite 7425)

CMP >=: Größer gleich

Beschreibung

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die

Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

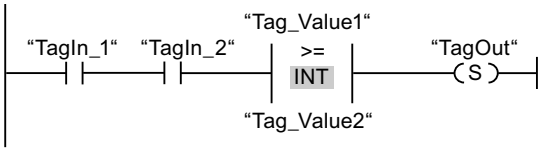

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" >= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

CMP <=: Kleiner gleich

Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

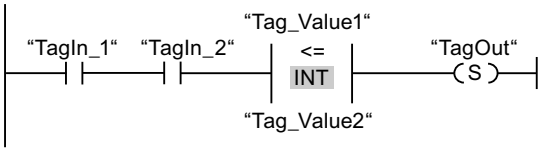

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

CMP >: Größer

Beschreibung

Mit der Anweisung "Größer" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

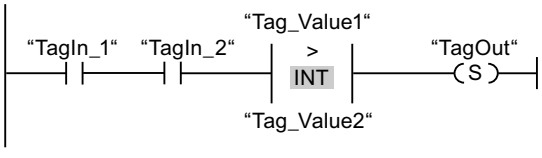

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" > "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

CMP <: Kleiner

Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
<Operand2>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

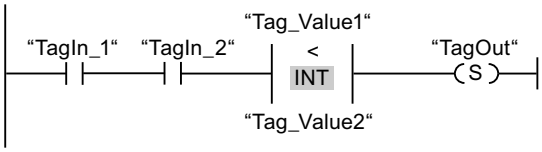

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" < "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

Mathematische Funktionen

ADD: Addieren

Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen die Summe am Ausgang OUT ab (OUT := IN1+IN2). Es können maximal nur 2 Eingänge angegeben werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Summand
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Summand
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Summe

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

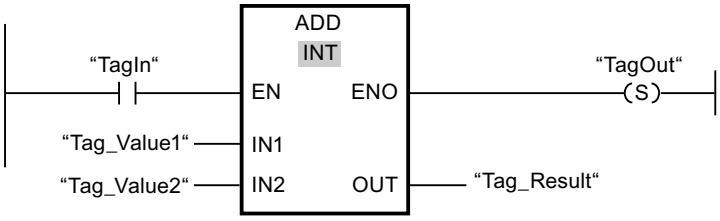

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Addieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

SUB: Subtrahieren

Beschreibung

Mit der Anweisung "Subtrahieren" ziehen Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 ab und fragen die Differenz am Ausgang OUT ab (OUT := IN1-IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Differenz

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

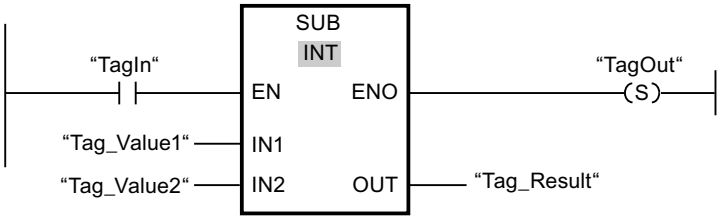

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Subtrahieren" ausgeführt. Der Wert des Operanden "Tag_Value2" wird vom Wert des Operanden "Tag_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

MUL: Multiplizieren

Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen das Produkt am Ausgang OUT ab (OUT := IN1*IN2). Es können maximal 2 Eingänge angegeben werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikator
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikand
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Produkt

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

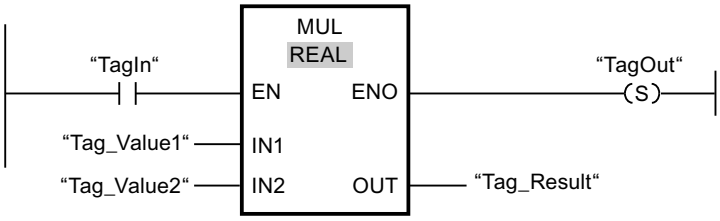

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Multiplikationsergebnis wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

DIV: Dividieren

Beschreibung

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab (OUT := IN1/IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Wert des Quotienten

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

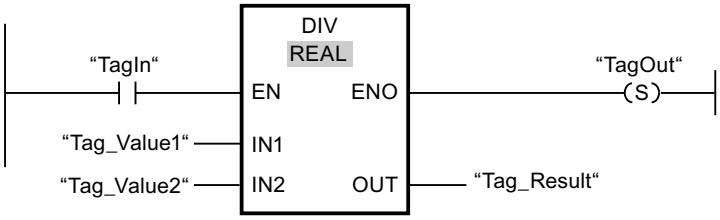

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Dividieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele KOP (Seite 7480)

Grundlagen zu KOP (Seite 7425)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	DINT	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	DINT	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	DINT	E, A, M, D, L, P	Divisionsrest

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

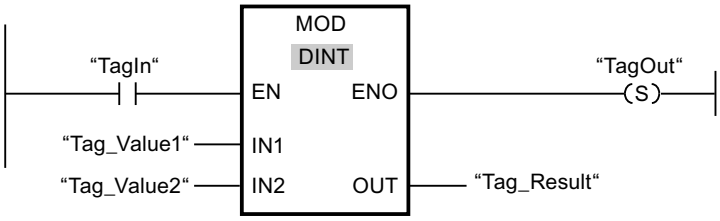

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Divisionsrest gewinnen" ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag_Result" gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist (gilt nur für Ganzzahlen).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweierkomplement des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

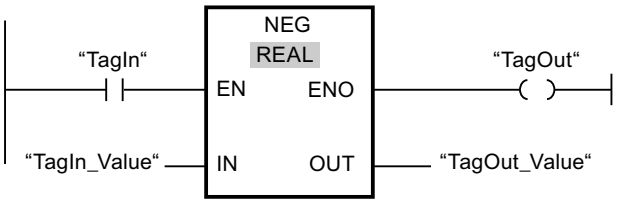

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zweierkomplement erzeugen" ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut_Value"

ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Absolutwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

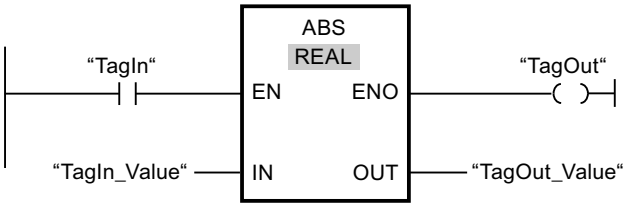

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	-6,234
OUT	TagOut_Value	6,234

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Absolutwert bilden" ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn_Value" und gibt das Ergebnis am Ausgang "TagOut_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den kleinsten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

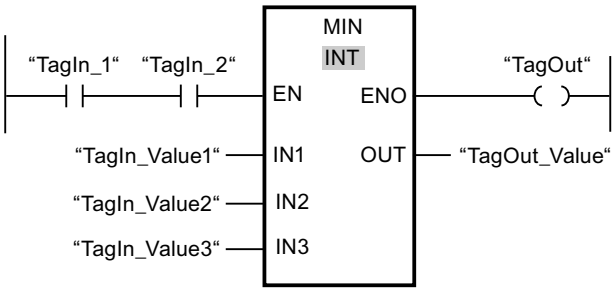

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung "Minimum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den größten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

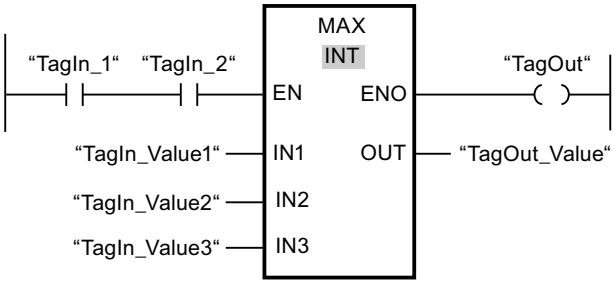

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12666
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung "Maximum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

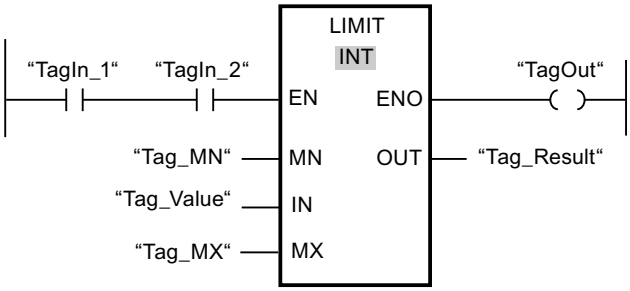

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung "Limitieren" ausgeführt. Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_MN" und "Tag_MX" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_MN" in den Ausgang "Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert einer Gleitpunktzahl und schreiben das Ergebnis in den Ausgang OUT.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswertes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

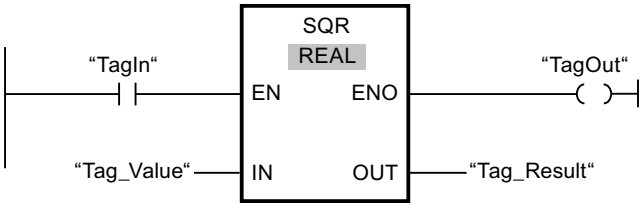

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadrat bilden" ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert die Quadratwurzel einer Gleitpunktzahl und schreiben das Ergebnis in den Ausgang OUT. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

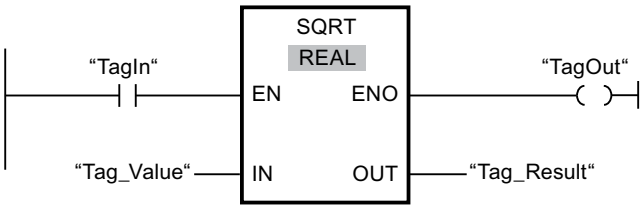

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadratwurzel bilden" ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis e (e = 2,718282). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

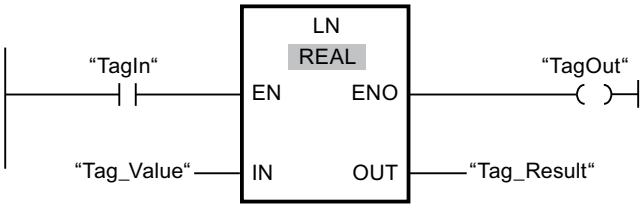

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Natürlichen Logarithmus bilden" ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag_Value" und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden (OUT = e^{IN}).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts IN

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

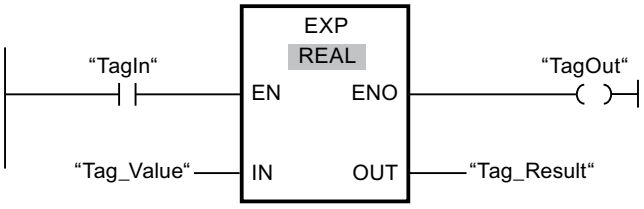

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Exponentialwert bilden" ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Sinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

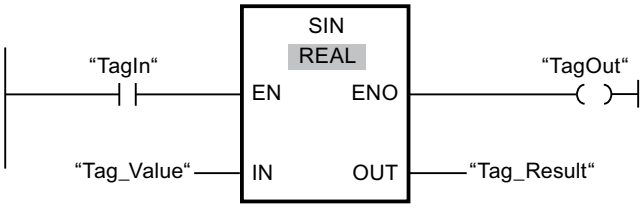

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	1.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Sinuswert bilden" ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

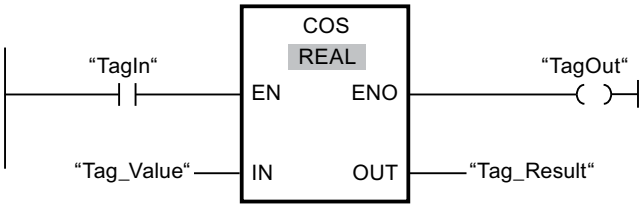

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Cosinuswert bilden" ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Tangens des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

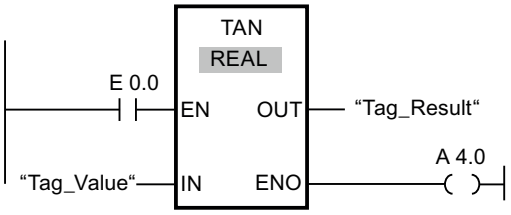

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Sinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

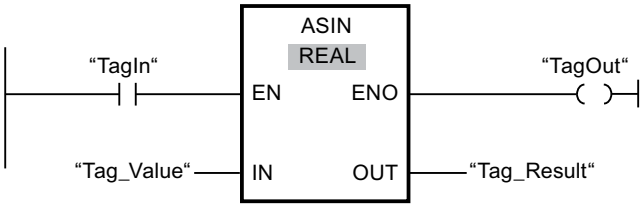

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcussinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Cosinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

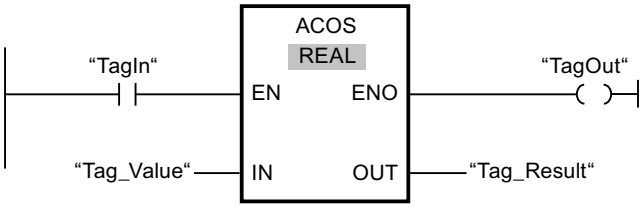

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcuscosinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Tangenswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

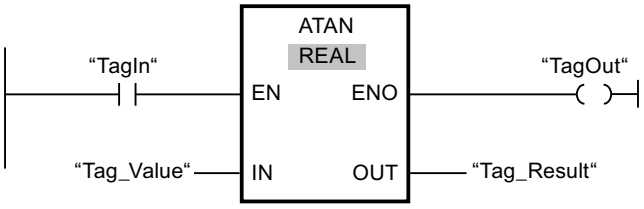

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1,0
OUT	Tag_Result	+0,785398 (π/4)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcustangenswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu KOP (Seite 7425)

Verschieben

MOVE: Wert kopieren

Beschreibung

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Die folgende Tabelle zeigt die möglichen Übertragungen:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD, CHAR
WORD	WORD; DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, INT, DINT, REAL, TIME, DATE, TOD, CHAR
INT	INT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
TIME	TIME	BYTE, WORD, DWORD, INT, DINT, TIME
S5TIME	S5TIME	WORD, S5TIME
DATE	DATE	BYTE, WORD, DWORD, INT, DINT, DATE
TOD	TOD	BYTE, WORD, DWORD, INT, DINT, TOD
CHAR	CHAR	BYTE, WORD, DWORD, CHAR
COUNTER	INT, WORD, COUNTER	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	INT, WORD, TIMER	WORD, DWORD, INT, UINT, DINT, UDINT

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen aufgefüllt.

Zum Kopieren von Feldern und Strukturen können die Anweisungen "Bereich kopieren" (BLKMOV) und "Bereich ununterbrechbar kopieren" (UBLKMOV) verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, TIMER, COUNTER	E, A, M, D, L oder Konstante	Quellwert
OUT1	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, TIMER, COUNTER	E, A, M, D, L	Zieladresse

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

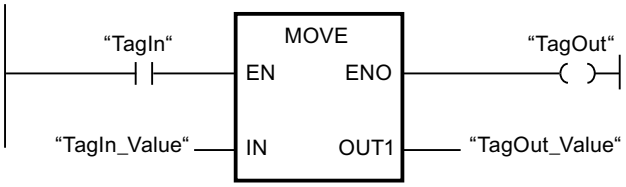

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Wert kopieren" ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn_Value" zum Operanden "TagOut_Value" und setzt den Ausgang "TagOut" auf den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

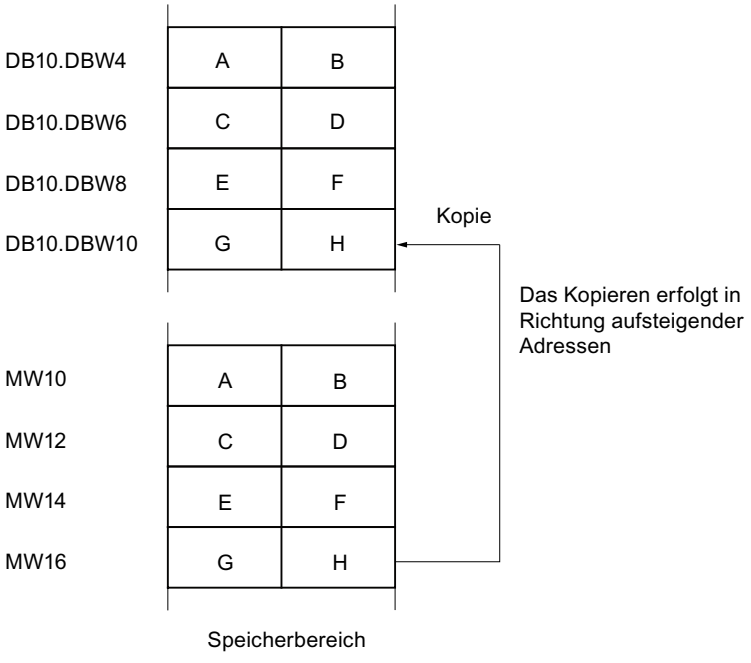

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Solange der Quellbereich nicht Teil eines Datenbausteins ist, der nur im Ladespeicher vorhanden ist, gibt es keine Begrenzung der Schachtelungstiefe.

Bei Unterbrechung einer BLKMOV-Bearbeitung hingegen, bei der aus einem nicht ablaufrelevanten DB kopiert wird, kann eine solche BLKMOV-Bearbeitung nicht mehr eingeschachtelt werden.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Nicht ablaufrelevante Datenbausteine

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Regeln beim Kopieren von nicht ablaufrelevanten Datenbausteinen

Der Quellbereich kann auch in einem nicht ablaufrelevanten Datenbaustein im Ladespeicher liegen. Datenbausteine, die nicht ablaufrelevant sind, werden mit dem Schlüsselwort UNLINKED gekennzeichnet.

Wenn ein nicht ablaufrelevanter Datenbaustein mit der Anweisung "Bereich kopieren" in den Arbeitsspeicher kopiert und gleichzeitig z. B. durch das Programmiergerät geladen wird, kann die Bearbeitung der Anweisung um mehrere Millisekunden verzögert werden. Dies führt zu einer Verlängerung des OB-Zyklus und kann zum Ansprechen der Zykluszeitüberwachung führen.

Wenn mit der Anweisung "Bereich kopieren" ein nicht ablaufrelevanter Datenbaustein kopiert wird und der Kopiervorgang unterbrochen wird, kann die Bearbeitung der Anweisung nicht mehr fortgesetzt werden.

Falls auf Ihrer CPU die Anweisung "Aus Datenbaustein im Ladespeicher lesen" vorhanden ist, müssen Sie diese Anweisung zum Lesen von nicht ablaufrelevanten Datenbausteinen im Ladespeicher benutzen. Wenn Sie die Anweisung "Bereich kopieren" benutzen, wird der Fehler W#16#8092 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

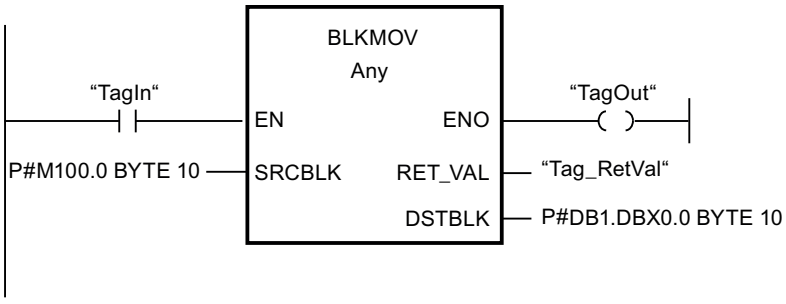

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 512 Bytes kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

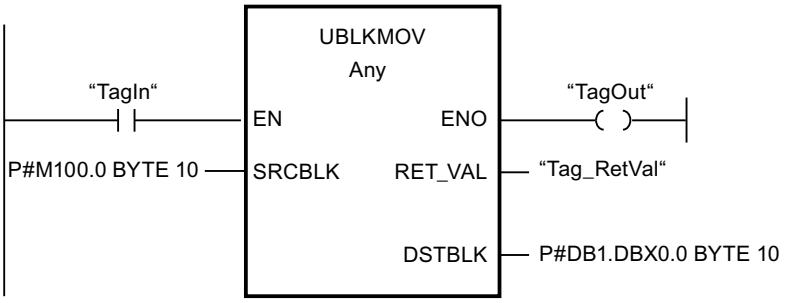

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

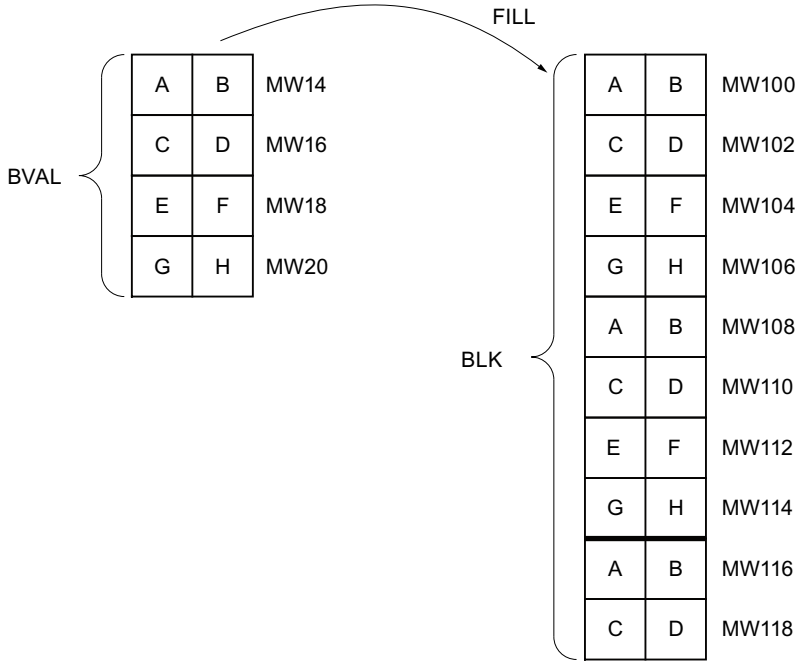

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
BVAL	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

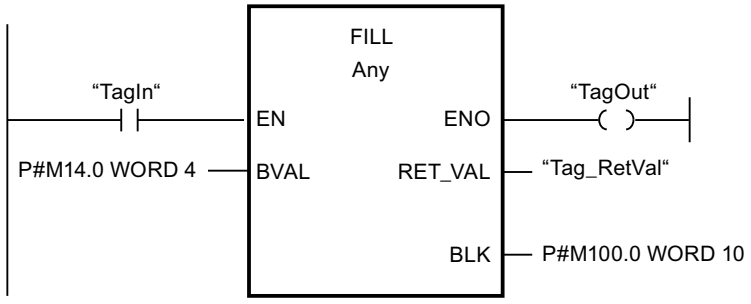

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend den in der Anweisungsbox ausgewählten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Weitere Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
OUT	Output	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

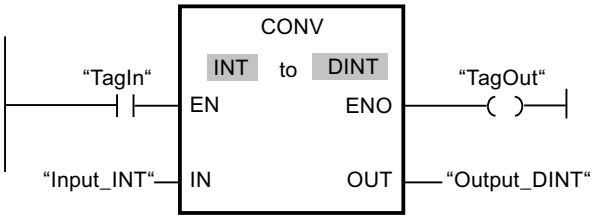

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird der Inhalt des Operanden "TagIn_Value" gelesen und in eine Ganzzahl (32 Bit) umgewandelt. Das Ergebnis wird im Operanden "TagOut_Value" abgelegt. Der Ausgang "TagOut" wird auf "1" gesetzt, wenn die Anweisung fehlerfrei ausgeführt wurde.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Explizite Konvertierungen (Seite 592)

Grundlagen zu KOP (Seite 7425)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl vom Datentyp DINT um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis der Rundung

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

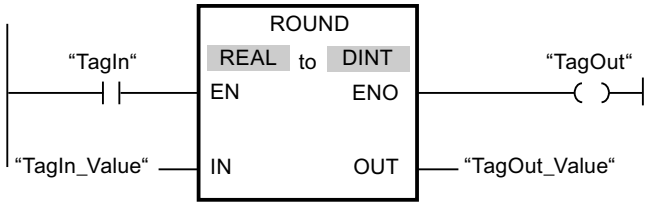

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächsten geraden Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

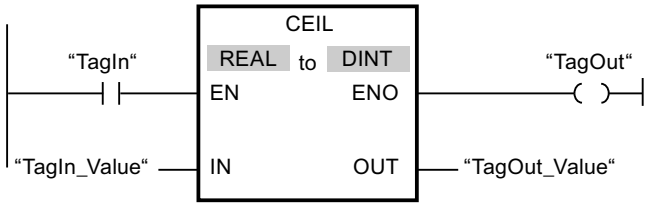

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

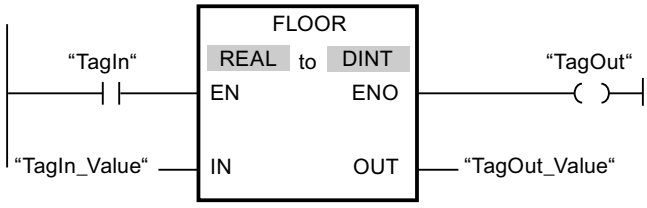

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingangwert
OUT	Output	Ganzzahlen	E, A, M, D, L	Ganzzahliger Teil des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

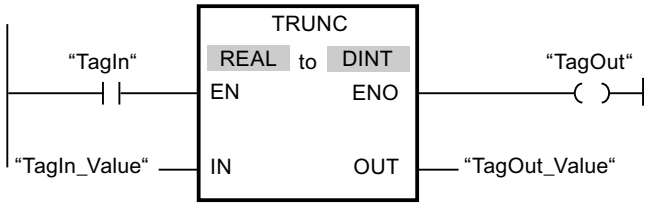

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu KOP (Seite 7425)

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

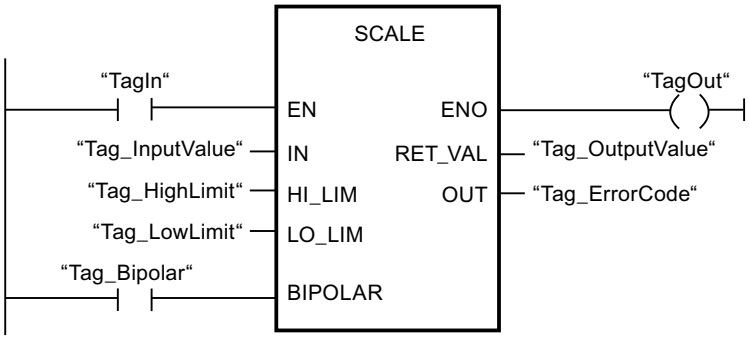

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Input	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

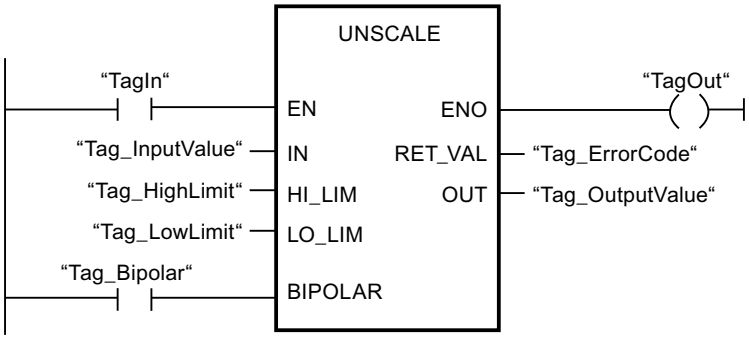

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

Programmsteuerung

---(JMP): Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

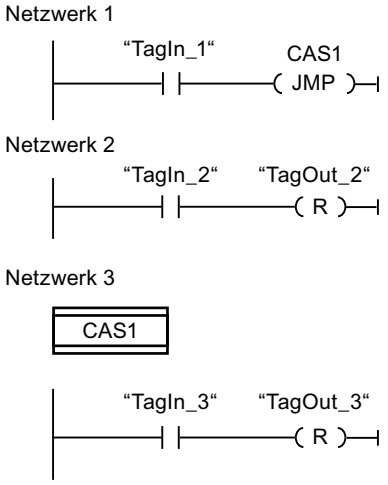

Wenn der Operand "TagIn_1"den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Grundlagen zu KOP (Seite 7425)

---(JMPN): Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das Verknüpfungsergebnis am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

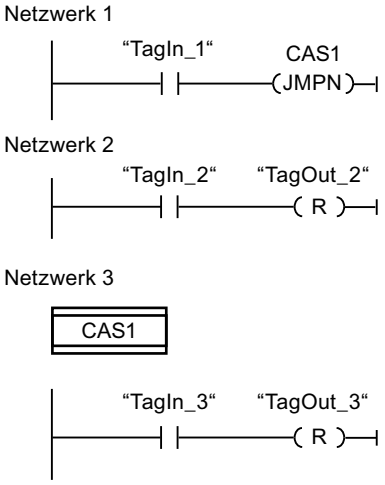

Wenn der Operand "TagIn_1" den Signalzustand "0" liefert, wird die Anweisung "Springen bei VKE = 0" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Grundlagen zu KOP (Seite 7425)

LABEL: Sprungmarke

Beschreibung

Mit einer Sprungmarke kennzeichnen Sie ein Zielnetzwerk, in dem bei einem ausgeführten Sprung die Programmbearbeitung fortgesetzt wird.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

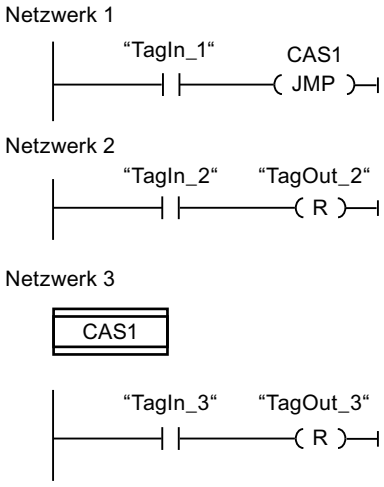

Wenn der Operand "TagIn_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

Grundlagen zu KOP (Seite 7425)

--(RET): Zurück springen

Beschreibung

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Programmbausteins.

- Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im aktuell aufgerufenen Programmbaustein beendet und im aufrufenden Programmbaustein (z. B. im aufrufenden Organisationsbaustein) nach der Aufruffunktion fortgesetzt. Der Signalzustand des Freigabeausgangs ENO der Aufruffunktion wird durch das VKE bestimmt. Das VKE wird über der Anweisung als fester Parameterwert dargestellt.
- Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt. Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Programmbausteins fortgesetzt.

Wenn die Anweisung im letzten Netzwerk des aufgerufenen Programmbausteins programmiert ist, wird der Freigabeausgang ENO der Aufruffunktion auf den aktuellen Signalzustand des VKE gesetzt.

Wenn mit der Anweisung ein Organisationsbaustein beendet wird, arbeitet die CPU im Systemprogramm weiter.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

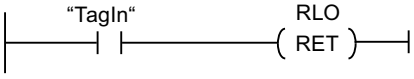

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Programmbearbeitung wird im aufgerufenen Programmbaustein beendet und im aufrufenden Programmbaustein fortgesetzt. Der Freigabeausgang ENO der Aufruffunktion wird auf den Signalzustand "1" gesetzt.

Siehe auch

Grundlagen zu KOP (Seite 7425)

---(OPN): Globalen Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Globalen Datenbaustein öffnen" schlagen Sie einen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Die darauf folgenden DB-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Globalen Datenbaustein öffnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Datenbau-stein>	Input	DB_ANY	DB	Datenbaustein, der aufge-schlagen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

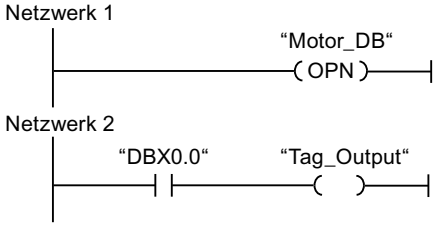

Im Netzwerk 1 wird der Datenbaustein "Motor_DB" aufgerufen. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Im Netzwerk 2 wird der Operand "DBX0.0" abgefragt. Der Signalzustand des Operanden "DBX0.0" wird dem Operanden "Tag_Output" zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

---(OPNI): Instanz-Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Instanz-Datenbaustein öffnen" schlagen Sie einen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DI-Register übertragen. Die darauf folgenden DI-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Instanz-Datenbaustein öffnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Datenbau-stein>	Input	DB_ANY	DB	Datenbaustein, der aufge-schlagen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

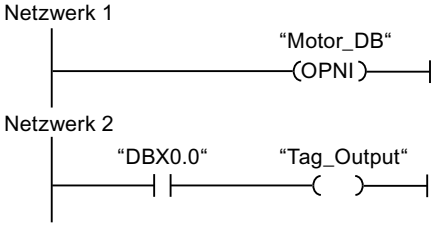

Im Netzwerk 1 wird der Datenbaustein "Motor_DB" aufgerufen. Die Nummer des Datenbausteins wird in DI-Register übertragen. Im Netzwerk 2 wird der Operand "DBX0.0" abgefragt. Der Signalzustand des Operanden "DBX0.0" wird dem Operanden "Tag_Output" zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu KOP (Seite 7425)

Laufzeitsteuerung

COMPRESS: CPU-Speicher komprimieren

Beschreibung

Mit der Anweisung "CPU-Speicher komprimieren" komprimieren Sie den Arbeitsspeicher der CPU bzw. den RAM-Ladespeicher. Dabei werden Lücken, die durch Löschen oder Laden von Bausteinen entstanden sind, aufgefüllt, indem die vorhandenen Bausteine zusammengeschoben werden.

Mit dem Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Der Komprimiervorgang wird über mehrere Programmzyklen verteilt. Die Überwachung des Komprimiervorgangs erfolgt über die Werte der einzelnen Parameter. Wenn am Parameter BUSY der Anweisung der Wert "1" ansteht, ist der Komprimiervorgang noch in Ausführung. Der Wert "1" am Parameter DONE meldet, dass der Komprimiervorgang beendet ist. Fehlerinformationen können am Parameter RET_VAL abgefragt werden.

Wenn bereits ein extern angestoßener Komprimiervorgang ausgeführt wird, führt der Aufruf der Anweisung "CPU-Speicher komprimieren" zur Fehleranzeige.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CPU-Speicher komprimieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BUSY	Output	BOOL	E, A, M, D, L	Information, ob Komprimiervorgang aktiv ist.
DONE	Output	BOOL	E, A, M, D, L	Information, ob der Komprimiervorgang beendet wurde.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler Der Status des Komprimiervorgangs kann an den Parametern BUSY und DONE abgefragt werden.
8091	Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt.
8092	Die Anweisung kann nicht ausgeführt werden, da eine der folgenden Bedingungen zutrifft: • Die Funktion "Baustein löschen" ist aktiv. • Eine Test- und Inbetriebsetzungsfunktion wirkt auf einen Baustein ein (z. B. Status). • Die Funktion "Kopieren von Bausteinen" ist aufgrund eines externen Anstoßes aktiv. • Die H-CPU befindet sich in einem Ankoppel- oder Aktualisierungsvorgang.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufgerufen wird, wird die Anweisung nicht ausgeführt und der Freigabeausgang ENO auf den Signalzustand "0" gesetzt.

Die Anweisung "Zyklusüberwachungszeit neu starten" läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter.

Siehe auch

Grundlagen zu KOP (Seite 7425)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter.

Siehe auch

Grundlagen zu KOP (Seite 7425)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Gültig für S7-300-CPUs:

Bei den S7-300-CPUs (mit Ausnahme von S7-318) ist die mit der Anweisung programmierte Verzögerungszeit eine Mindestzeit. Sie verlängert sich um die Ausführungszeit der eingeschachtelten Prioritätsklassen sowie um Systemlasten.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

PROTECT: Schutzstufe ändern

Beschreibung

Mit der Anweisung "Schutzstufe ändern" ändern Sie die projektierte Schutzstufe "1" der CPU ab.

Die parametrierte Schutzstufe "1" mit aktivierter Option "Durch Passwort aufhebbar" bewirkt, dass man ein Passwort parametrieren muss. Mit diesem Passwort können Sie auf die mit der Anweisung "Schutzstufe ändern" schreibgeschützte CPU im Betrieb schreibend zugreifen (z. B. Laden).

Die Schutzstufe legen Sie am Parameter MODE der Anweisung fest. Die aktuelle Schutzstufe der CPU wird im Inspektorfenster unter "Eigenschaften > Schutz" angezeigt.

Die folgende Tabelle zeigt, welche Bedeutung die Schutzstufen haben:

Schutz-stufe	Bedeutung
1	Alle PG-Funktionen sind erlaubt. Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
2	• Das Programm und die Konfiguration der CPU können nicht geändert werden. Das Programm in der CPU kann gelesen werden. • Die Funktionen für Prozessführung, Prozessbeobachtung und Prozesskommunikation sind erlaubt. • Alle Auskunftsfunktionen sind erlaubt. • Mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreibschutz aufheben. • Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
3	• Auch mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreib- und Leseschutz nicht aufheben. • Wenn zum Zeitpunkt des Aufrufs der Anweisung "Schutzstufe ändern" eine legitimierte Verbindung vorhanden ist, bleibt der Aufruf der Anweisung für diese Verbindung ohne Wirkung.

Wenn Sie für Ihre CPU die Schutzstufe "2" oder "3" projektiert haben, hat der Aufruf der Anweisung "Schutzstufe ändern" keine Wirkung.

Die Wirksamkeit der Schutzstufe "2" bzw. "3" nach Ausführung der Anweisung "Schutzstufe ändern" können Sie der SZL-Teilliste mit der SZL-ID W#16#0232 und dem Index W#16#0004 entnehmen.

Hinweis

Mit der Anweisung "Schutzstufe ändern" können Sie keine niedrigere Schutzstufe einstellen als die, die Sie mit STEP 7 "Hardware konfigurieren" projektiert haben. Sie haben z. B. die Schutzstufe 3 projektiert und anschließend die Anweisung "Schutzstufe ändern" mit dem Parameter MODE = 12 aufgerufen. Dies bewirkt eine Sperre der Passwort-Legitimierung. Ein erneuter Aufruf der Anweisung mit dem Parameter MODE = 0 oder 1 bewirkt, dass die Schutzstufe 3 beibehalten wird und dass der Schreib- und Leseschutz mit Eingabe des gültigen Passworts aufgehoben wird.

Auswirkungen der Betriebszustandsübergänge auf die Schutzstufe

Die folgende Tabelle zeigt, welche Auswirkungen die Betriebszustandsübergänge auf die mit der Anweisung "Schutzstufe ändern" eingestellte Schutzstufe haben.

Aktion	Auswirkungen auf die Schutzstufe
• Betriebsartenschalter auf "STOPP" • Ungepuffertes NETZ-EIN • Urlöschen durch Betriebsartenschalter • Konfigurations- und/oder Programmänderungen im Betriebszustand "STOPP" der S7-400-CPU	Die Schutzstufe wird auf "1" gesetzt. Alle PG-Funktionen sind erlaubt.
• Kaltstart • Neustart (Warmstart)	Die Schutzstufe wird durch das Betriebssystem auf "1" gesetzt. Alle PG-Funktionen sind erlaubt. Bei Bedarf können Sie durch Aufruf der Anweisung "Schutzstufe ändern" in Ihrem Programm die Schutzstufe auf "2" oder "3" setzen.
Gepuffertes NETZ-EIN	Bleibt unverändert
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOPP (durch den Aufruf der Anweisung "Programm beenden" (STP), durch Programmfehler ohne zugehörigen Fehler-OB oder durch Anwenderaktion).	Bleibt unverändert
S7-400: Wiederanlauf durch Anwenderaktion oder automatisch nach NETZ-EIN	Bleibt unverändert

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schutzstufe ändern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MODE	Input	WORD	E, A, M, D, L oder Konstante	Schutzstufe Mögliche Werte: • W#16#0000: Einstellen der Schutzstufe "1" • W#16#0001: Einstellen der Schutzstufe "2" mit Passwort-Legitimierung • W#16#000C: Einstellen der Schutzstufe "3" ohne Passwort-Legitimierung Am Parameter MODE können auch Konstanten angegeben werden.
RET_VAL	Output	INT	E, A, M, D, L	Tritt während der Bearbeitung ein Fehler auf, wird eine Fehlerinformation ausgegeben.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
8090	Unzulässiger Wert am Parameter MODE.
80C3	Die benötigten Betriebsmittel sind momentan belegt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

CiR: CiR-Vorgang steuern

Beschreibung

Mit der Anweisung "CiR-Vorgang steuern" beeinflussen Sie im Anwenderprogramm den CiR-Vorgang wie folgt:

- Den CiR-Vorgang sperren. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU stets abgelehnt. Die Sperre gilt so lange, bis Sie diese mit der Anweisung "CiR-Vorgang steuern" wieder aufheben.
- Den CiR-Vorgang bedingt sperren, indem Sie am Parameter FRZ_TIME eine Obergrenze für die CiR-Synchronisationszeit festlegen. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU nur dann zugelassen, wenn die in der CPU errechnete CiR-Synchronisationszeit die angegebene Obergrenze nicht überschreitet.
- Den CiR-Vorgang freigeben. Die Obergrenze der CiR-Synchronisationszeit wird auf den Defaultwert 1000 ms gesetzt.
- Ermitteln, ob der CiR-Vorgang freigegeben ist oder nicht. Falls der CiR-Vorgang freigegeben oder bedingt freigegeben ist, erhalten Sie im Parameter A_FT der Anweisung die aktuelle Obergrenze für die CiR-Synchronisationszeit.

Hinweis

Während der CiR-Synchronisationszeit sind die Ausgänge eingefroren und die Eingänge werden nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CiR-Vorgang steuern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte am Parameter MODE sind: • 0: Auskunftsfunktion • 1: CiR-Vorgang freigeben • 2: CiR-Vorgang sperren • 3: CiR-Vorgang bedingt sperren. Die Obergrenze der CiR-Synchronisationszeit legen Sie in FRZ_TIME fest.
FRZ_TIME	Input	TIME	E, A, M, D, L oder Konstante	Obergrenze der CiR-Synchronisationszeit in Millisekunden. Zulässiger Wertebereich: 60 ... 2500 ms (Voreinstellung: 1000 ms) Der Parameter FRZ_TIME ist nur dann relevant, wenn der Parameter MODE auf den Wert "3" gesetzt ist.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation Wenn der Parameter MODE den Wert "0" hat, liefert der Parameter RET_VAL Informationen über die Freigabe des CiR-Vorgangs.
A_FT	Output	TIME	E, A, M, D, L	Aktuell gültige Obergrenze der CiR-Synchronisationszeit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt. Dieser Fehlercode kann ausgegeben werden, nur wenn der Parameter MODE auf den Wert "1", "2" oder "3" gesetzt ist.
0001	Der CiR-Vorgang ist freigegeben. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.

Fehlercode* (W#16#...)	Erläuterung
0002	Der CiR-Vorgang ist gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0003	Der CiR-Vorgang ist bedingt gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
8001	Die CPU ist nicht bereit für einen CiR-Vorgang. Sie verwenden eine H-CPU im H-System (Solobetrieb) oder Sie arbeiten mit einer Standard-CPU im Multicomputing-Betrieb.
8002	Der Wert am Parameter MODE ist nicht zulässig.
8003	Der Wert am Parameter FRZ_TIME ist nicht zulässig.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Mit der Anweisung "CiR-Vorgang steuern" können Sie beispielsweise programmieren, dass kein CiR-Vorgang in Zeitabschnitten angestoßen wird, in denen für die Prozessbearbeitung die maximale Leistungsfähigkeit der CPU benötigt wird.

Dazu muss die Anweisung "CiR-Vorgang steuern" vor Beginn des Zeitabschnitts mit erhöhter Prozessaktivität im Programm aufgerufen und mit dem Wert "2" (CiR-Vorgang sperren) am Parameter MODE ausgeführt werden. Dadurch wird den CiR-Vorgang gesperrt.

Um den CiR-Vorgang nach der Beendigung des Zeitabschnitts mit erhöhter Prozessaktivität wieder freizugeben, muss die Anweisung "CiR-Vorgang steuern" erneut aufrufen und mit dem Wert "1" am Parameter MODE ausführt werden. Sie können alternativ den Parameter MODE auch auf den Wert "3" setzen und dadurch den CiR-Vorgang bedingt sperren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

Wortverknüpfungen

AND: UND verknüpfen

Beschreibung

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

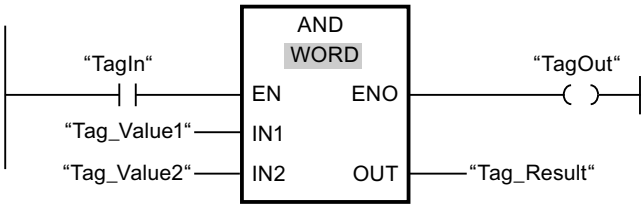

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

OR: ODER verknüpfen

Beschreibung

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

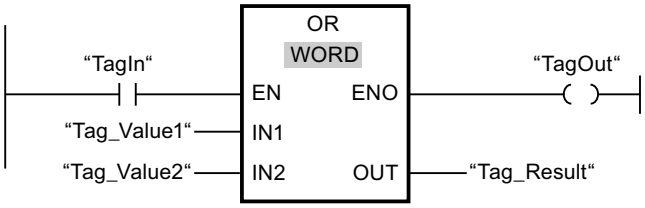

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

XOR: EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

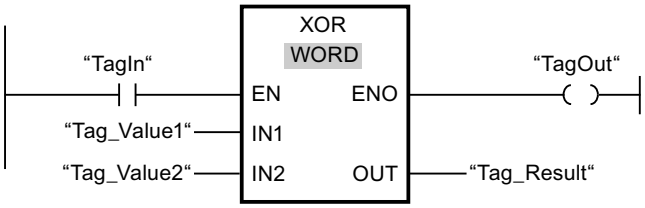

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

KOP-Elemente einfügen (Seite 7439)

Datentyp eines KOP-Elements auswählen (Seite 7444)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Grundlagen zu KOP (Seite 7425)

INVERT: Einerkomplement erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFFFFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen	E, A, M, D, L, P	Einerkomplement des Wertes am Eingang IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

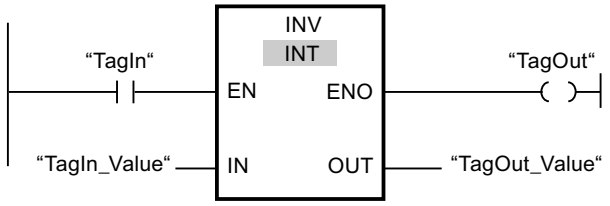

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	W#16#000F
OUT	TagOut_Value	W#16#FFF0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn_Value" um und schreibt das Ergebnis in den Ausgang "TagOut_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	DWORD	E, A, M, D, L, P	Ausgangswert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

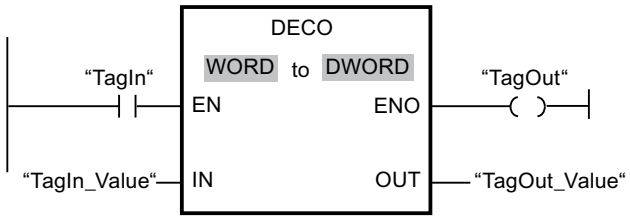

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

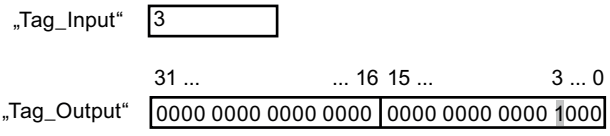

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn_Value" am Eingang ab und setzt das dritte Bit im Wert des Operanden "TagOut_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

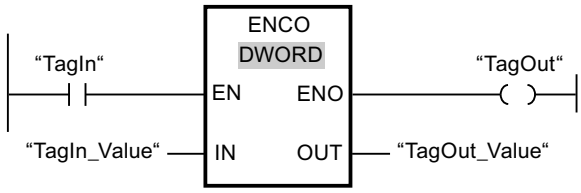

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

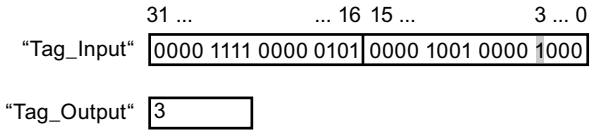

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung selektiert das niederwertigste gesetzte Bit am Eingang "TagIn_Value" und schreibt die Bitposition "3" in die Variable am Ausgang "TagOut_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie, abhängig von einem Schalter (Eingang G), einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
G	Input	BOOL	E, A, M, D, L, T, Z	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

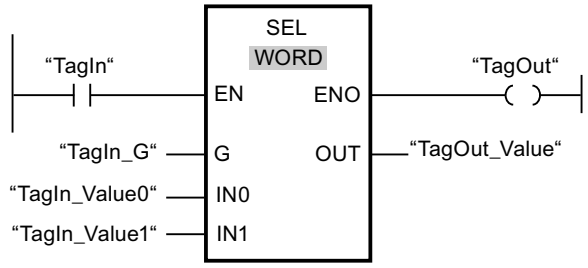

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn_G" wird der Wert am Eingang "TagIn_Value0" oder "TagIn_Value1" selektiert und in den Ausgang "TagOut_Value" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zu KOP (Seite 7425)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp INT um vier Bitstellen nach rechts verschoben wird:

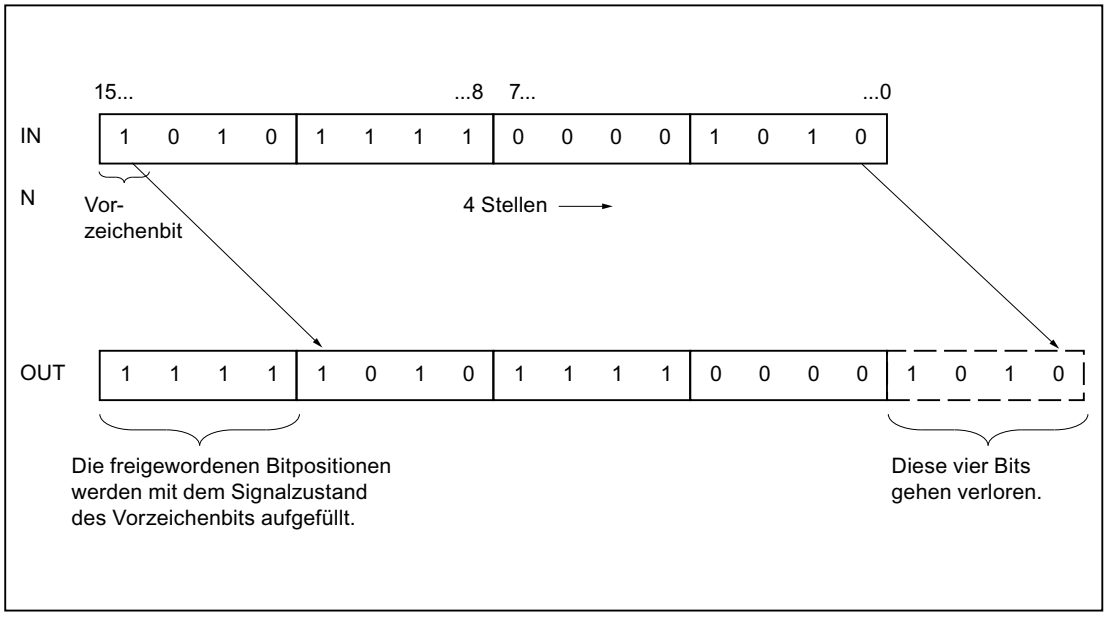

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird.
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird.
OUT	Output	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

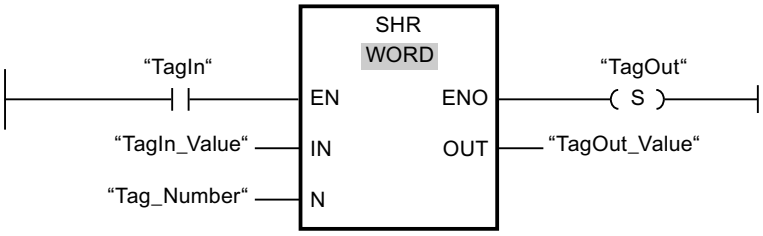

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts schieben" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zu KOP (Seite 7425)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

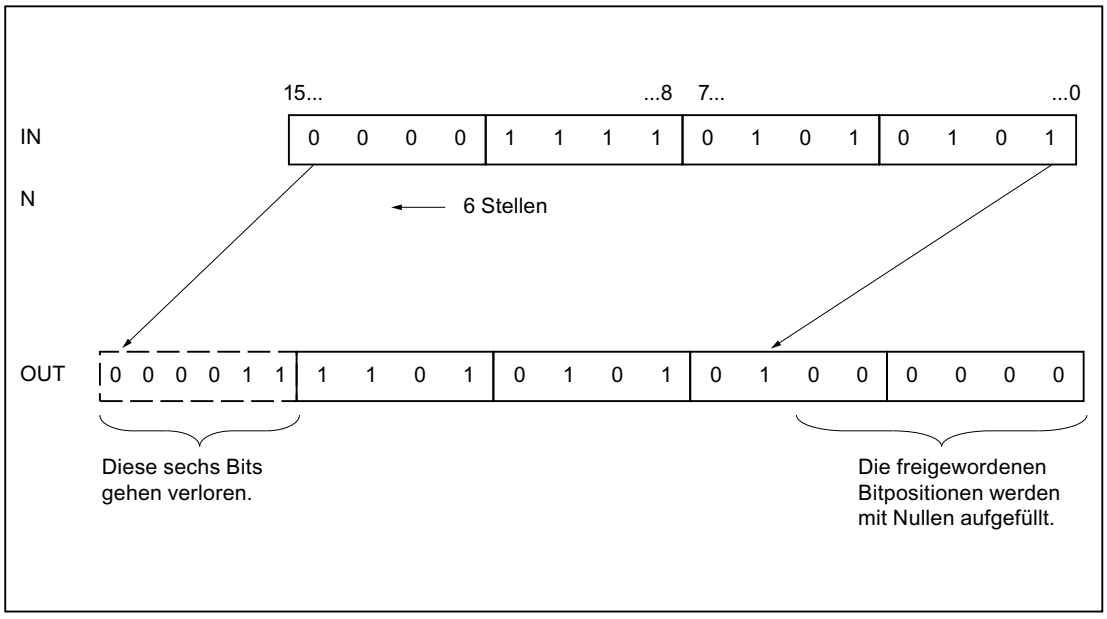

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

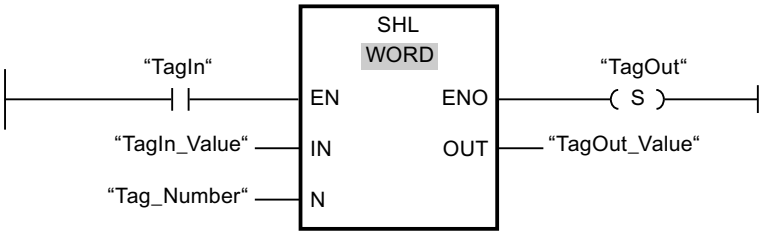

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links schieben" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zu KOP (Seite 7425)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

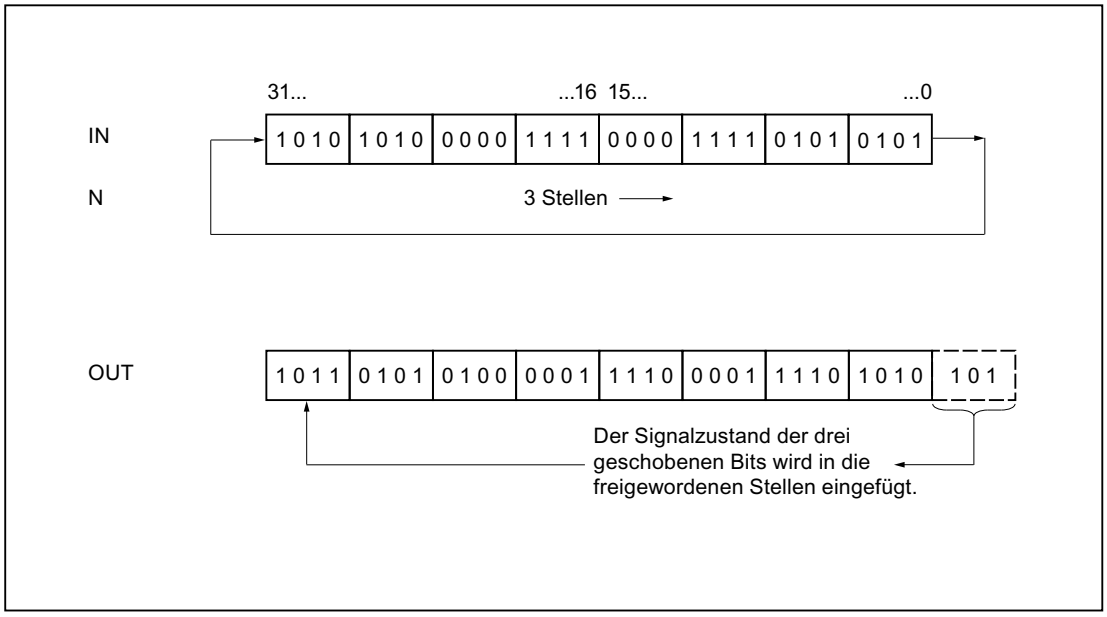

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Information zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

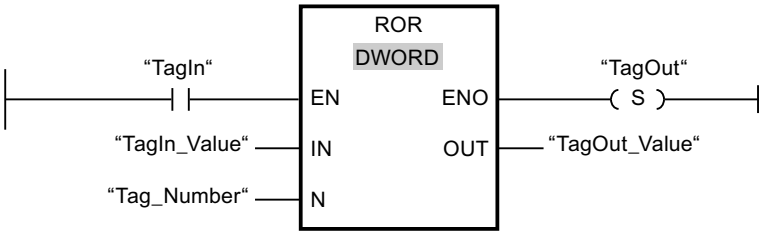

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1101 0101 0000 0111 1000 0111 1010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts rotieren" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zu KOP (Seite 7425)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

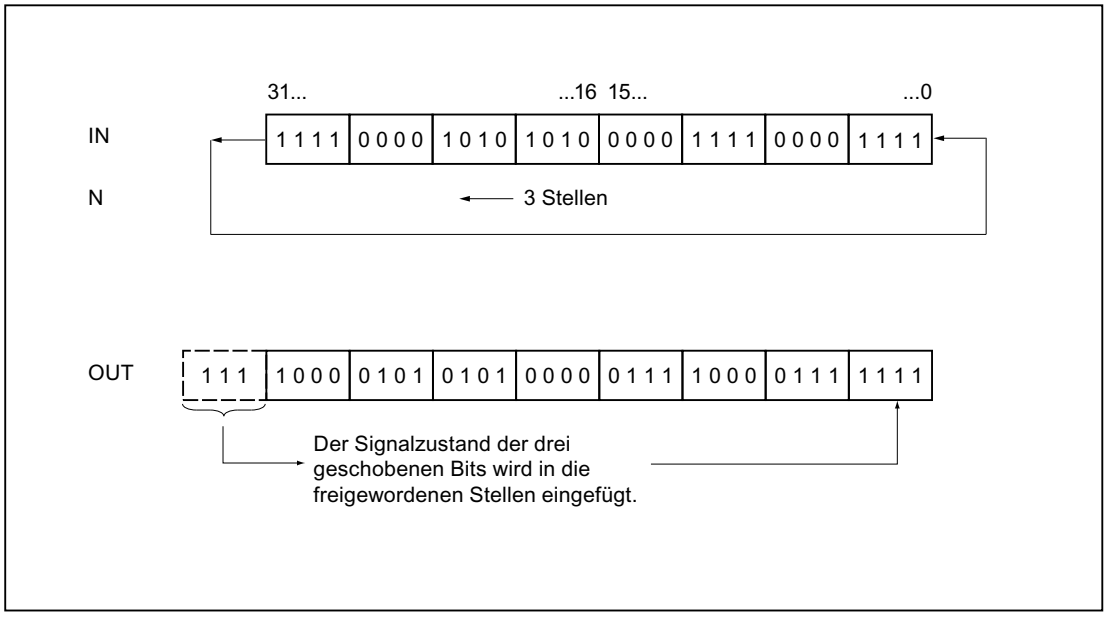

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

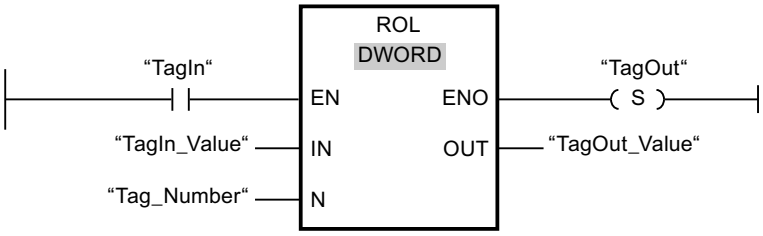

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
OUT	TagOut_Value	0001 0101 0100 0001 1110 0001 1111 1110

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links rotieren" ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zu KOP (Seite 7425)

Weitere Anweisungen

---| |---: Statusbit abfragen

Beschreibung

Mit der Anweisung "Statusbit abfragen" fragen Sie den Signalzustand der folgenden Statusbits ab:

- BIE: Binärergebnis
- OV: Überlauf
- OS: Speichernder Überlauf
- A0 und A1: Anzeigebits

Welche Statusbits abgefragt werden, legen Sie durch die Auswahl des entsprechenden Symbols aus der Klappliste der Anweisung fest. Die folgende Tabelle zeigt die Möglichkeiten zur Abfrage der Statusbits:

Symbol	Abfrage	Erläuterung
==0	A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion gleich 0 ist.
<>0	A0 = 0 und A1 = 1 oder A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungleich 0 ist.
>=0	A0 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer oder gleich 0 ist.
<=0	A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner oder gleich 0 ist.
>0	A0 = 0 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer 0 ist.
<0	A0 = 1 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner 0 ist.
UO	A0 = 1 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungültig ist. Ursache für ein ungültiges Ergebnis kann eine ungültige Gleitpunktzahl oder eine Division durch Null sein.
OV	OV = 1	Die Anweisung fragt ab, ob in einer Anweisung ein Zahlenbereichsüberlauf aufgetreten ist.
OS	OS = 1	Die Anweisung fragt ab, ob in einer Anweisung ein speichernder Zahlenbereichsüberlauf aufgetreten ist.
BIE	BIE = 1	Die Anweisung fragt den Signalzustand des Statusbits BIE.

Die Anweisung "Statusbit abfragen" funktioniert wie ein Schließerkontakt. Wenn die Abfrage erfüllt ist, liefert die Anweisung den Signalzustand "1". Bei nicht erfüllter Abfrage liefert die Anweisung den Signalzustand "0".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

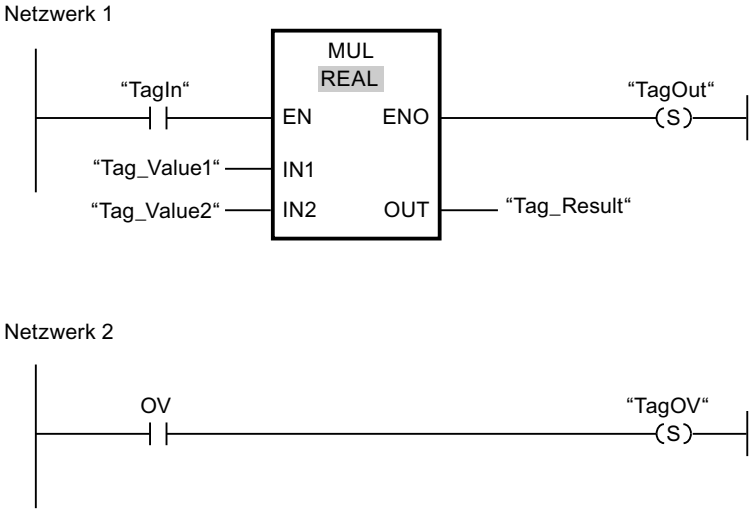

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" (MUL) ausgeführt. Bei einem Überlauf während der Ausführung der Anweisung "Multiplizieren" wird das Statusbit OV gesetzt. Nach der Abfrage des Statusbits OV wird im Netzwerk 2 die Anweisung "Ausgang setzen" (S) ausgeführt und der Operand "TagOV" gesetzt.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Setzen der Statusbits (Seite 185)

---| / |---: Statusbit negiert abfragen

Beschreibung

Mit der Anweisung "Statusbit negiert abfragen" fragen Sie den Signalzustand der folgenden Statusbits ab und negieren das Abfrageergebnis:

- BIE: Binärergebnis
- OV: Überlauf
- OS: Speichernder Überlauf
- A0 und A1: Anzeigebits

Welche Statusbits abgefragt werden, legen Sie durch die Auswahl des entsprechenden Symbols aus der Klappliste der Anweisung fest. Die folgende Tabelle zeigt die Möglichkeiten zur Abfrage der Statusbits:

Symbol	Abfrage	Erläuterung
==0	A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion gleich 0 ist.
<>0	A0 = 0 und A1 = 1 oder A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungleich 0 ist.
>=0	A0 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer oder gleich 0 ist.
<=0	A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner oder gleich 0 ist.
>0	A0 = 0 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer 0 ist.
<0	A0 = 1 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner 0 ist.
UO	A0 = 1 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungültig ist. Ursache für ein ungültiges Ergebnis kann eine ungültige Gleitpunktzahl oder eine Division durch Null sein.
OV	OV = 1	Die Anweisung fragt ab, ob in einer Anweisung einen Zahlenbereichsüberlauf aufgetreten ist.
OS	OS = 1	Die Anweisung fragt ab, in einer Anweisung einen speichernden Zahlenbereichsüberlauf aufgetreten ist.
BIE	BIE = 1	Die Anweisung fragt den Signalzustand des Statusbits BIE.

Die Anweisung "Statusbit negiert abfragen" funktioniert wie ein Öffnerkontakt. Wenn die Abfrage erfüllt ist, liefert die Anweisung den Signalzustand "0". Bei nicht erfüllter Abfrage liefert die Anweisung den Signalzustand "1".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

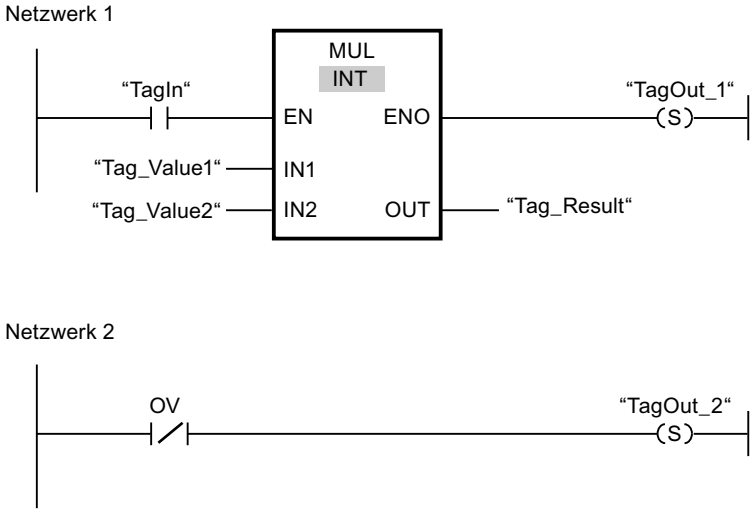

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" (MUL) ausgeführt. Wenn während der Ausführung der Anweisung kein Überlauf auftritt, wird das Statusbit OV auf "0" zurückgesetzt. Im Netzwerk 2 wird nach der Abfrage des Statusbits OV die Anweisung "Ausgang setzen" (S) ausgeführt und der Operand "TagOut_2" gesetzt.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Setzen der Statusbits (Seite 185)

---(CALL): Baustein aufrufen

Beschreibung

Mit der Anweisung "Baustein aufrufen" rufen Sie eine Funktion (FC) ohne Parameter auf. Abhängig von der vorangehenden Verknüpfung handelt es sich um einen absoluten oder bedingten Aufruf.

Im Anweisungsteil einer Funktion (FC) können Sie bei einem bedingten Aufruf als Operand keinen Parameter vom Datentyp BLOCK_FC angeben. In einem Funktionsbaustein (FB) jedoch können Sie als Operanden einen Parameter vom Typ BLOCK_FC angeben.

Ein bedingter Aufruf wird nur dann ausgeführt, wenn das Verknüpfungsergebnis (VKE) "1" ist. Wird ein bedingter Aufruf nicht ausgeführt, ist das VKE nach der Aufrufoperation "0". Wird die Operation ausgeführt, arbeitet sie folgendermaßen:

- Sie speichert die Rücksprungadresse des aufrufenden Bausteins.
- Sie speichert die beiden Datenbaustein-Register (Datenbaustein und Instanz-Datenbaustein).
- Sie aktualisiert den Lokaldatenbereich für die aufgerufene Funktion.
- Sie schiebt das MA-Bit (aktives MCR-Bit) in den Baustein-Stack (B-Stack).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Baustein aufrufen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Baustein>	Output	BLOCK_FC	-	Aufgerufene Funktion

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Die Funktion "FC3" wird aufgerufen, wenn der Operand "TagIn_1" den Signalzustand "1" führt.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

---(SAVE): VKE im BIE-Bit speichern

Beschreibung

Mit der Anweisung "VKE im BIE-Bit speichern" sichern Sie das Verknüpfungsergebnis (VKE) im Binärergebnis (BIE). In der Ausführung überträgt die Anweisung den Signalzustand des aktuellen VKEs zum Statusbit BIE. Die Anweisung arbeitet unabhängig von Bedingungen und beeinflusst keine weiteren Statusbits.

Hinweis

Das BIE-Bit kann nach der Anwendung der Anweisung "VKE im BIE-Bit speichern" durch nachfolgende Anweisungen im gleichen oder in unterlagerten Bausteinen wieder verändert werden.

Sie können die Anweisung "VKE im BIE-Bit speichern" nutzen, um den Ausführungsstatus eines Bausteins zu kontrollieren. Wenn Sie z. B. die Anweisung "VKE im BIE-Bit speichern" am Ende eines Bausteins programmieren, wird das BIE-Bit auf den Signalzustand des aktuellen VKEs des Bausteins gesetzt. Wurde der Baustein fehlerfrei ausgeführt, wird das BIE-Bit sowie der Freigabeausgang ENO im aufrufenden Baustein auf den Signalzustand "1" gesetzt. Der Signalzustand "0" wird dem BIE-Bit zugewiesen, wenn während der Bearbeitung des Bausteins Fehler aufgetreten sind. Durch die Abfrage des BIE-Bits können Sie eine Fehlerbehandlung programmieren.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

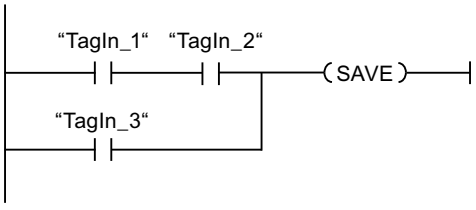

Das BIE-Bit wird auf den Signalzustand "1" gesetzt, wenn eine der folgenden Bedingungen ausgeführt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "1".

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Beispiel für den EN-/ENO-Mechanismus in KOP (Seite 154)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Setzen der Statusbits (Seite 185)

---(MCR<): MCR-Bereiche öffnen

Beschreibung

Mit der Anweisung "MCR-Bereiche öffnen" speichern Sie das Verknüpfungsergebnis (VKE) im MCR-Stack und schalten die MCR-Abhängigkeit innerhalb eines MCR-Bereichs ein (MCR = Master Control Relay).

Der MCR-Stack kann maximal 8 Einträge enthalten und arbeitet wie ein LIFO-Zwischenspeicher (Last In, First Out). Ist der Stack voll, erzeugt die Anweisung eine Fehlermeldung (MCRF).

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Den Anfang und das Ende eines MCR-Bereichs legen Sie durch die Anweisungen "MCR-Bereich aktivieren" (MCRA) und "MCR-Bereich deaktivieren" (MCRD) fest. Innerhalb des definierten MCR-Bereichs können Sie für einen oder mehrere Programmteile die MCR-Abhängigkeit ein- und ausschalten.

Die MCR-Abhängigkeit wird eingeschaltet, wenn das VKE vor der Bearbeitung der Anweisung "1" ist. Wenn das VKE vor der Anweisung "0" ist, wird die MCR-Abhängigkeit ausgeschaltet. Das Ende des MCR-abhängigen Programmabschnitts wird durch die Anweisung "MCR-Bereiche schließen" programmiert. Der Programmabschnitt zwischen den beiden Anweisungen wird durch das MCR beeinflusst. Betroffen sind dabei nur die folgenden MCR-abhängigen Anweisungen:

- Zuweisung
- Ausgang setzen (S)
- Ausgang rücksetzen (R)
- Flipflop rücksetzen/setzen (RS)
- Flipflop setzen/rücksetzen (SR)
- Wert kopieren (MOVE)

Das Ein- und Ausschalten der MCR-Abhängigkeit können Sie auch schachteln, indem Sie in einem MCR-abhängigen Programmabschnitt die Anweisung "MCR-Bereiche öffnen" programmieren. Die Anweisung können Sie bis 8 Ebenen tief schachteln. Die MCR-Abhängigkeit des übergeordneten Programmteils steuert dabei die MCR-Abhängigkeiten in allen untergeordneten Programmteilen. Wenn die MCR-Abhängigkeit im übergeordneten Programmteil abgeschaltet ist, kann diese in einem untergeordneten MCR-abhängigen Programmteil nicht eingeschaltet werden.

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins, wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

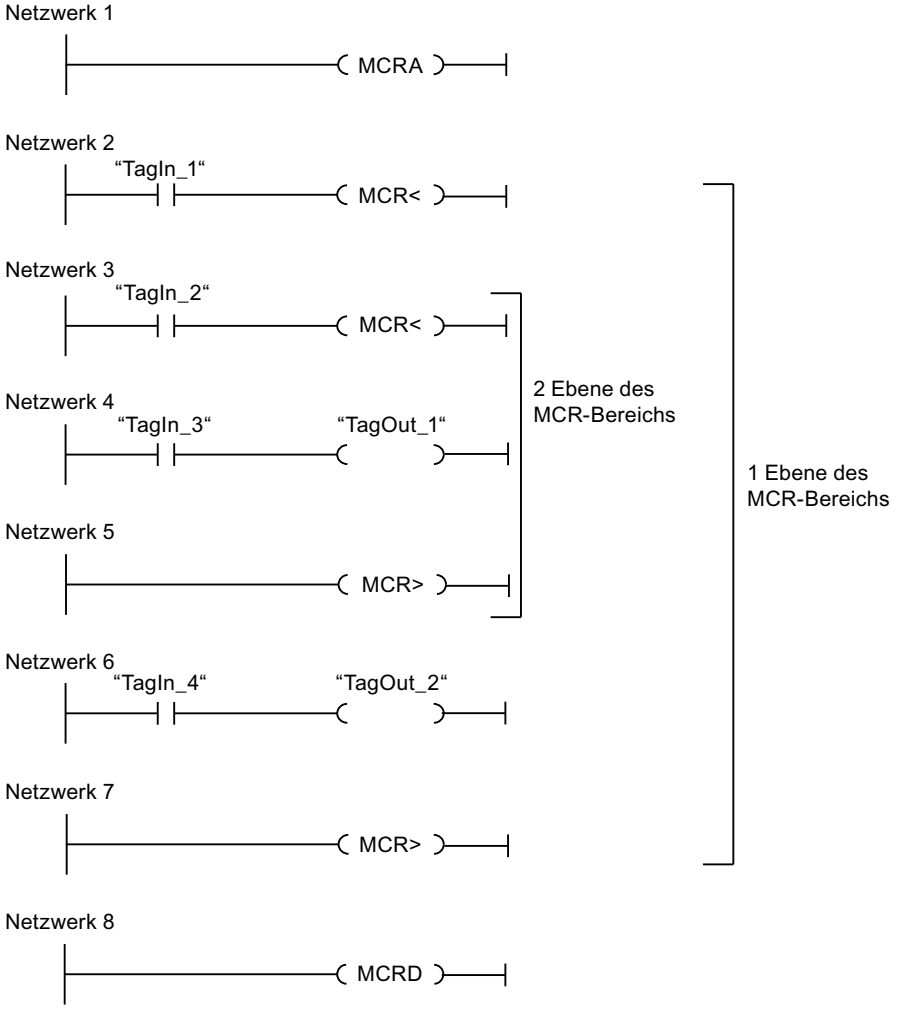

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit in der ersten Ebene des MCR-Bereichs eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_4" dem Operanden "TagOut_2" (Netzwerk 6) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_2" auf "0" zurückgesetzt. Im Netzwerk 7 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet.

Im Netzwerk 3 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs eingeschaltet, wenn der Operand "TagIn_2" den Signalzustand "1" liefert und die erste Ebene des MCR-Bereichs eingeschaltet ist. In diesem Fall wird der Signalzustand des Operanden "TagIn_3" dem Operanden "TagOut_1" zugewiesen. Wenn der Operand "TagIn_2" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_1" auf "0" zurückgesetzt. Im Netzwerk 5 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Das Master Control Relay (Seite 201)

---(MCR>): MCR-Bereiche schließen (Seite 4532)

---(MCRA): MCR-Bereich aktivieren (Seite 4534)

---(MCRD): MCR-Bereich deaktivieren (Seite 4535)

---(MCR>): MCR-Bereiche schließen

Beschreibung

Mit der Anweisung "MCR-Bereiche schließen" schließen Sie einen MCR-abhängigen (MCR = Master Control Relay) Programmabschnitt ab. Die Anweisung löscht dabei das Verknüpfungsergebnis (VKE) aus dem letzten Eintrag im MCR-Stack.

Der MCR-Stack kann maximal 8 Einträge enthalten und arbeitet wie ein LIFO-Zwischenspeicher (Last In, First Out). Ist der Stack bereits leer, erzeugt die Anweisung eine Fehlermeldung (MCRF).

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Die MCR-Abhängigkeit programmieren Sie mit der Anweisung "MCR-Bereiche öffnen". Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

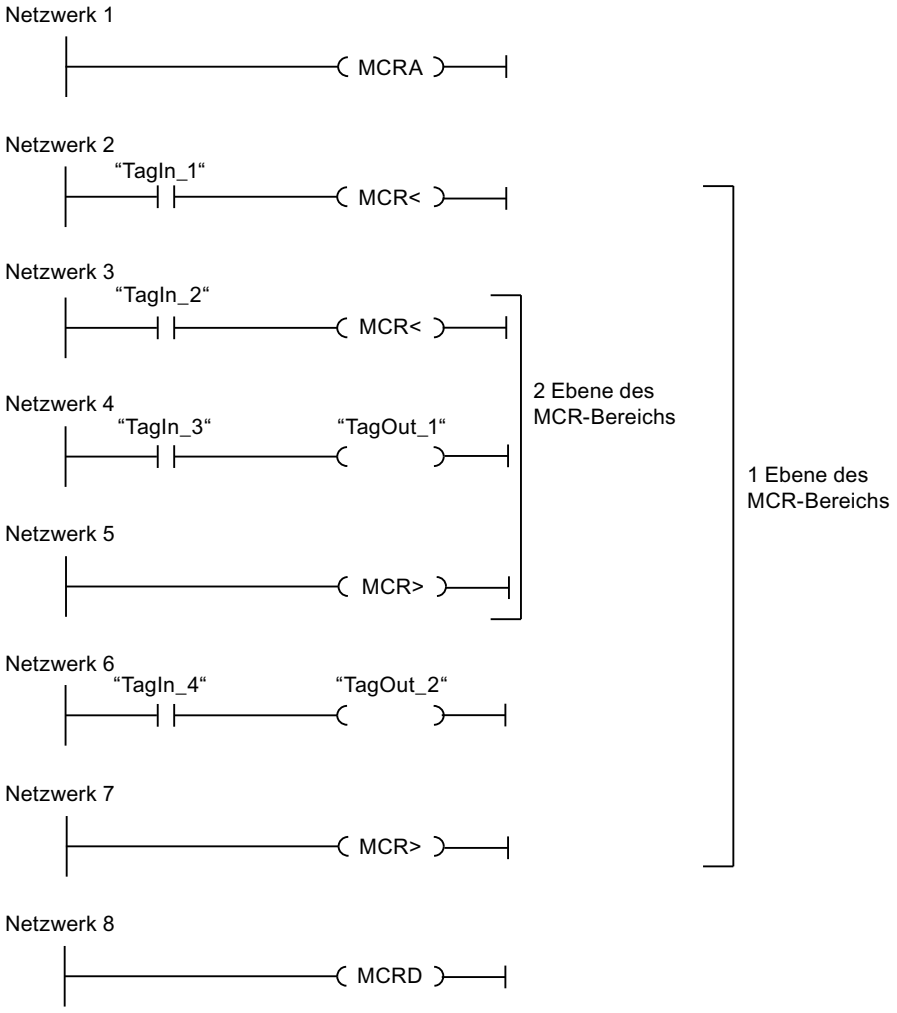

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit in der ersten Ebene des MCR-Bereichs eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_4" dem Operanden "TagOut_2" (Netzwerk 6) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_2" auf "0" zurückgesetzt. Im Netzwerk 7 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet.

Im Netzwerk 3 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs eingeschaltet, wenn der Operand "TagIn_2" den Signalzustand "1" liefert. In diesem Fall wird der Signalzustand des Operanden "TagIn_3" dem Operanden "TagOut_1" zugewiesen. Wenn der Operand "TagIn_2" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_1" unabhängig vom Signalzustand des Operanden "TagIn_3" auf "0" zurückgesetzt. Im Netzwerk 5 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

---(MCR<): MCR-Bereiche öffnen (Seite 4530)

---(MCRA): MCR-Bereich aktivieren (Seite 4534)

---(MCRD): MCR-Bereich deaktivieren (Seite 4535)

---(MCRA): MCR-Bereich aktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich aktivieren" legen Sie den Anfang eines MCR-Bereichs fest. Innerhalb dieses Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen ein- und ausschalten:

- ---(MCR<): MCR-Bereiche öffnen
- ---(MCR>): MCR-Bereiche schließen

Das Ende des MCR-Bereichs legen Sie mit der Anweisung "MCR-Bereich deaktivieren" (MCRD) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

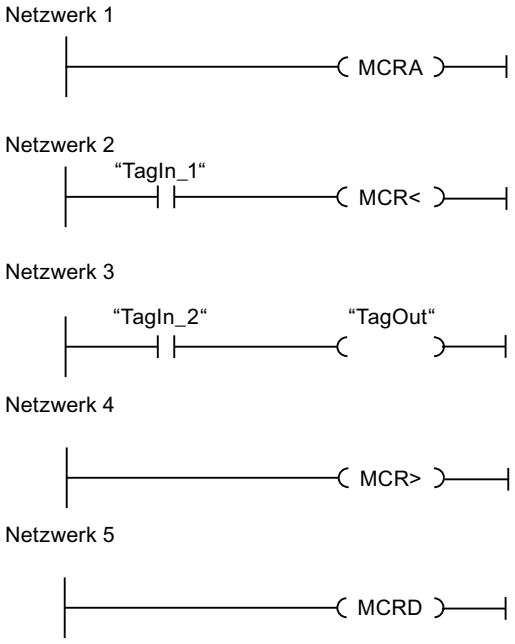

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_2" dem Operanden "TagOut" (Netzwerk 3) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut" unabhängig vom Signalzustand des Operanden "TagIn_2" auf "0" zurückgesetzt. Im Netzwerk 4 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

---(MCR<): MCR-Bereiche öffnen (Seite 4530)

---(MCR>): MCR-Bereiche schließen (Seite 4532)

---(MCRD): MCR-Bereich deaktivieren (Seite 4535)

---(MCRD): MCR-Bereich deaktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich deaktivieren" legen Sie das Ende eines MCR-Bereichs fest.

Den Anfang eines MCR-Bereichs legen Sie mit dem Anweisung "MCR-Bereich aktivieren" (MCRA) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Innerhalb eines MCR-Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen programmieren:

- ---(MCR<): MCR-Bereiche öffnen
- ---(MCR>): MCR-Bereiche schließen

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

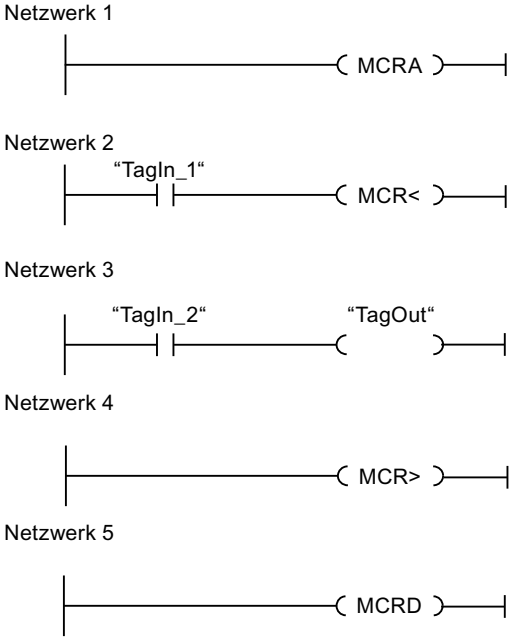

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_2" dem Operanden "TagOut" (Netzwerk 3) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut" unabhängig vom Signalzustand des Operanden "TagIn_2" auf "0" zurückgesetzt. Im Netzwerk 4 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

---(MCR<): MCR-Bereiche öffnen (Seite 4530)

---(MCR>): MCR-Bereiche schließen (Seite 4532)

---(MCRA): MCR-Bereich aktivieren (Seite 4534)

SET: Bereich bitweise setzen

Beschreibung

Mit der Anweisung "Bereich bitweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "1". Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu setzenden Bits im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Hinweis

Parameter N

Sie können am Parameter N auch ein Element eines Datenbausteins verschalten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der zu setzenden Bits

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

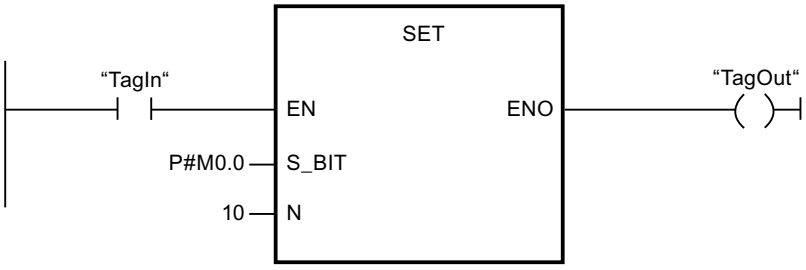

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

SETP: Bitfeld im Peripheriebereich setzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich setzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "1". Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge gesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das gesetzt wird. Die Bits des Feldes müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zu setzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Setzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bit, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bit im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bit auf den Wert der zugehörigen Bit im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bit, die zum definierten Bitfeld gehören, werden auf den Signalzustand "1" gesetzt.

Die Anweisung "Bitfeld im Peripheriebereich setzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich setzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu setzenden Bit
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste zu setzende Bit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

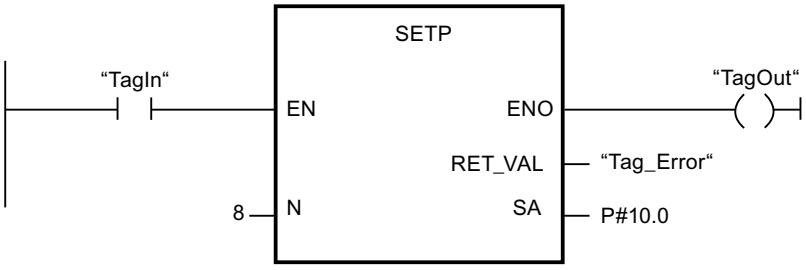

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

SETI: Bereich byteweise setzen

Beschreibung

Mit der Anweisung "Bereich byteweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "1". Den Anfangspunkt des Bereichs definieren Sie am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zu setzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich "Peripherie" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise setzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie.
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des zu setzenden Bereichs Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

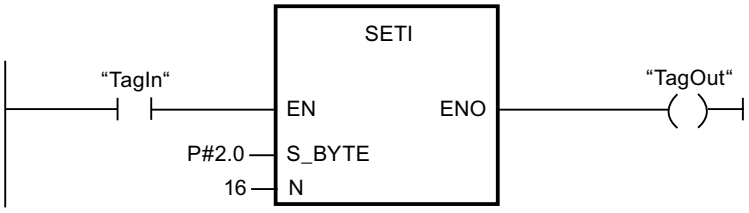

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

RESET: Bereich bitweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich bitweise rücksetzen" setzen Sie den Signalzustand der Bit in einem angegebenen Bereich auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zurückzusetzenden Bit im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der Bits, die zurückgesetzt werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

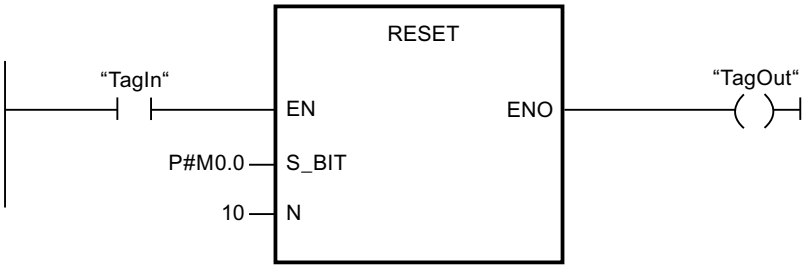

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

RESETP: Bitfeld im Peripheriebereich rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich rücksetzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "0" zurück. Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge zurückgesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das zurückgesetzt wird. Die Bits des Felds müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zurückzusetzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Zurücksetzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "0" zurückgesetzt.

Die Anweisung "Bitfeld im Peripheriebereich rücksetzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich rücksetzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der Bits, die zurückgesetzt werden.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste Bit, das zurückgesetzt wird.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

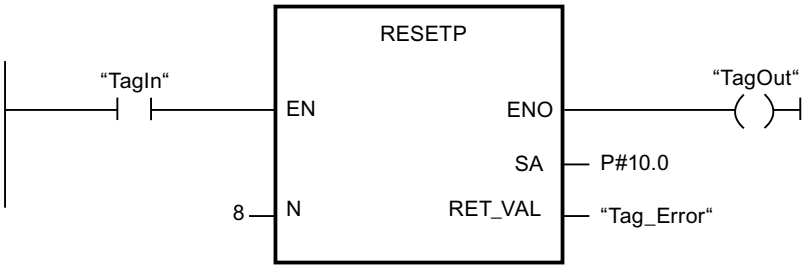

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

RESETI: Bereich byteweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich byteweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zurückzusetzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie verweisen. Da auf den Speicherbereich "Peripherie" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise rücksetzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie.
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des Bereichs, dessen Bits zurückgesetzt werden. Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

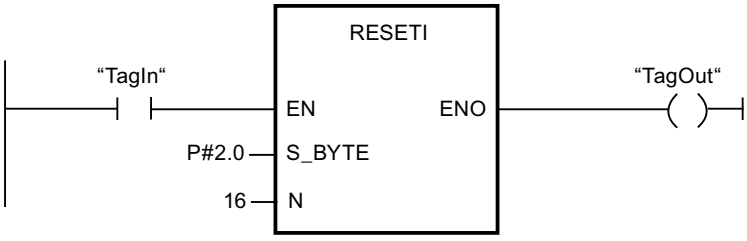

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

REPL_VAL: Ersatzwert eintragen

Beschreibung

Mit der Anweisung "Ersatzwert eintragen" tragen Sie von einem Synchronfehler-Organisationsbaustein aus einen Ersatzwert in den Akkumulator 1 (AKKU 1) ein.

Hinweis

Die Anweisung "Ersatzwert eintragen" dürfen Sie nur in Synchronfehler-Organisationsbausteinen (OB 121, OB 122) aufrufen.

Die Anweisung "Ersatzwert eintragen" verwenden Sie, wenn Sie von einer Baugruppe keinen Wert mehr lesen können. Nach jedem Zugriff auf diese Baugruppe wird der OB 122 gestartet. Mit dem Aufruf der Anweisung "Ersatzwert eintragen" können Sie bewirken, dass ein Ersatzwert in den Akkumulator 1 geladen wird. Die Programmbearbeitung wird dann mit diesem Ersatzwert fortgesetzt. Die für die Auswahl des Ersatzwerts notwendigen Informationen können Sie den lokalen Variablen des OB 122 entnehmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ersatzwert eintragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
VAL	Input	DWORD	E, A, M, D, L oder Konstante	Ersatzwert
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Die Anweisung wurde nicht von einem Synchronfehler-OB aus aufgerufen.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu KOP (Seite 7425)

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden

Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zuletzt programmierten Schritts.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

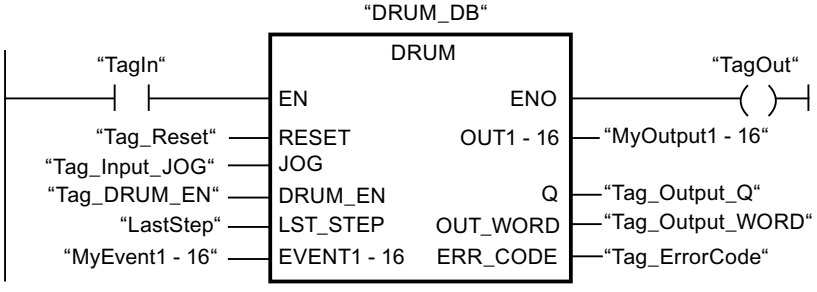

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

DRUM_X: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" realisieren Sie ein Schrittschaltwerk mit maximal 16 Schritten. Die Nummer des ersten Schritts legen Sie am Parameter DSP fest. Die Nummer des letzten Schritts geben Sie am Parameter LST_STEP vor.

In jedem Schritt werden alle 16 Ausgangsbits OUT0 bis OUT15 und der Ausgangsparameter OUT_WORD (in dem die Ausgangsbits zusammengefasst sind) beschrieben. Ein Ausgangsbit bekommt entweder das zugehörige Bit des von Ihnen vorgegebenen Felds OUT_VAL oder das Ausgangsbit des vorangegangenen Schritts zugewiesen. Welcher Wert zugewiesen wird, hängt von der Belegung der Maskenbits am Parameter S_MASK ab.

Hinweis

Die Voreinstellung der Maskenbits ist 0. Falls Sie die aktuelle Einstellung eines oder mehrerer Maskenbits ändern wollen, führen Sie diese Änderung im Instanz-Datenbaustein durch.

Die Anweisung "Schrittschaltwerk realisieren" schaltet in den nächsten Schritt, wenn am Parameter JOG eine positive Flanke aufgetreten ist. Falls sich die Anweisung bereits im letzten Schritt befindet, werden bei einer positiven Flanke am Parameter JOG die Variablen Q und EOD gesetzt. Der Parameter DCC wird in diesem Fall auf den Wert 0 gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" gesetzt wird.

Darüber hinaus können Sie auch ein zeitabhängiges Weiterschalten in den nächsten Schritt zulassen. Dazu muss der Parameter DRUM_EN auf den Wert "1" gesetzt werden. Der Übergang in den nächsten Schritt erfolgt, wenn das zum aktuellen Schritt zugehörige Ereignisbit (EVENTi) gesetzt ist und die für den aktuellen Schritt vorgegebene Zeit abgelaufen

ist. Die Zeit für einen Schritt ergibt sich aus dem Produkt von der Zeitbasis (DTBP) und dem für den aktuellen Schritt gültigen Zeitfaktor (S_PRESET).

Hinweis

Die im aktuellen Schritt noch verbleibende Bearbeitungszeit (DCC) wird nur dann reduziert, wenn das zugehörige Ereignisbit (EVENTi) gesetzt ist.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Wenn beim Aufruf der Anweisung der Wert "1" am Parameter RESET anliegt, geht das Schrittschaltwerk in den Schritt, dessen Nummer am Parameter DSP angegeben ist.

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist.

Zusätzlich können Sie das Schrittschaltwerk jederzeit über den Wert am Parameter JOG weiterschalten.

Beim Erstaufruf der Anweisung müssen Sie den Parameter RESET mit 1 belegen.

Wenn sich das Schaltwerk im letzten Schritt befindet und die für diesen Schritt vorgegebene Bearbeitungszeit abgelaufen ist, werden die Parameter Q und EOD gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" wechselt.

Die Anweisung wird nur in den Betriebszuständen ANLAUF und RUN ausgeführt.

Das Betriebssystem setzt die Anweisung "Schrittschaltwerk realisieren" bei Kaltstart zurück. Bei einem Neustart (Warmstart) wird die Anweisung nicht zurückgesetzt. Falls die Anweisung "Schrittschaltwerk realisieren" nach einem Neustart (Warmstart) initialisiert werden soll, müssen Sie die Anweisung im OB 100 mit dem Wert "1" am Parameter RESET aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" bewirkt das Rücksetzen des Schaltwerks. Beim Erstaufruf der Anweisung muss der Parameter RESET auf "1" gesetzt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
JOG	Input	BOOL	E, A, M, D, L	Eine positive Signalflanke (gegenüber dem letzten Aufruf) bewirkt das Weiterschalten in den nächsten Schritt, falls sich das Schaltwerk noch nicht im letzten Schritt befindet. Die Weiterschaltung erfolgt unabhängig davon, welchen Wert Sie am Parameter DRUM_EN angegeben haben.
DRUM_EN	Input	BOOL	E, A, M, D, L	Steuerparameter, der festlegt, ob ein zeitabhängiges Weiterschalten in den nächsten Schritt möglich ist. Bei einem Wert "1" ist ein zeitabhängiges Weiterschalten möglich.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des letzten Schritts Zulässige Werte sind 1 bis 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit Nr. i (gehört zum Schritt i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	E, A, M, D, L	Ausgangsbit Nr. j (identisch mit dem Bit Nr. j von OUT_WORD)
Q	Output	BOOL	E, A, M, D, L	Zustandsparameter, der angibt, ob die von Ihnen vorgegebene Bearbeitungszeit des letzten Schritts abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Ausgangsbits zusammengefasst in einer Variablen
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Nicht anwenderrelevant (Eingangsparameter JOG des vorhergehenden Aufrufs)
EOD	Static	BOOL	E, A, M, D, L	Identisch zum Ausgangsparameter Q
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des ersten Schritts Zulässige Werte sind 1 bis 16.
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des aktuellen Schritts
DCC	Static	DWORD	E, A, M, D, L, P	Die im aktuellen Schritt noch verbleibende Bearbeitungszeit in Millisekunden (ms) (nur relevant, falls DRUM_EN = 1 und das zugehörige Ereignisbit = 1)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Die für alle Schritte gültige Zeitbasis in Millisekunden (ms)
PREV_TIME	Static	DWORD	E, A, M, D, L oder Konstante	(Nicht anwenderrelevant: Systemzeit des vorhergehenden Aufrufs)
S_PRESET	Static	ARRAY of WORD	E, A, M, D, L oder Konstante	Eindimensionales Feld mit den Zeitfaktoren für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16]. In diesem Fall enthält S_PRESET [x] den Zeitfaktor des Schritts x.
OUT_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den in jedem Schritt auszugebenden Werten, falls Sie diese nicht über S_MASK ausgeblendet haben. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält OUT_VAL [x, y] den Wert, der dem Ausgangsbit OUTy im Schritt x zugewiesen wird.
S_MASK	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den Maskenbits für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält S_MASK [x, y] das Maskenbit für den "y"-ten auszugebenden Wert im Schritt x. Bedeutung der Maskenbits: • 0: der entsprechende Wert des vorangegangenen Schritts wird dem zugehörigen Ausgangsbit zugewiesen. • 1: der entsprechende Wert aus OUT_VAL wird dem zugehörigen Ausgangsbit zugewiesen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE* (W#16#...)	Erläuterung
0000	Kein Fehler
8081	Unzulässiger Wert am Parameter LST_STEP
8082	Unzulässiger Wert am DSC
8083	Unzulässiger Wert am DSP
8084	Das Produkt DCC = DTBP * S_PRESET[DSC] überschreitet den Wert 2**31-1 (ca. 24,86 Tage).
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT0 bis OUT15) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

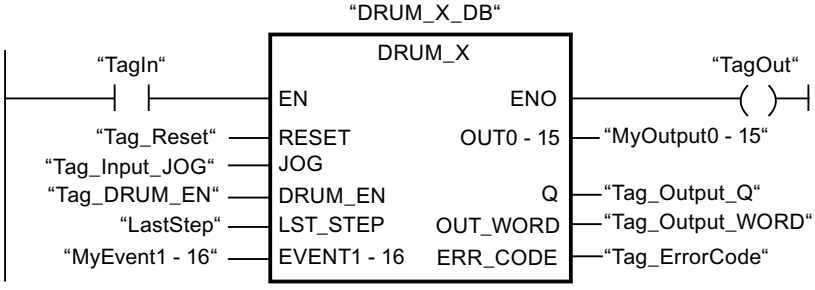

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE

Parameter	Operand	Adresse	Wert
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE

Parameter	Adresse	Wert
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	TRUE
OUT2	MyTag_Output_2	M4.2	TRUE
OUT3	MyTag_Output_3	M4.3	TRUE
OUT4	MyTag_Output_4	M4.4	TRUE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	TRUE
OUT7	MyTag_Output_7	M4.7	TRUE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	TRUE
OUT11	MyTag_Output_11	M5.3	TRUE
OUT12	MyTag_Output_12	M5.4	TRUE

Parameter	Operand	Adresse	Wert
OUT13	MyTag_Output_13	M5.5	TRUE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	FALSE
OUT2	MyTag_Output_2	M4.2	FALSE
OUT3	MyTag_Output_3	M4.3	FALSE
OUT4	MyTag_Output_4	M4.4	FALSE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	FALSE
OUT7	MyTag_Output_7	M4.7	FALSE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	FALSE
OUT11	MyTag_Output_11	M5.3	FALSE
OUT12	MyTag_Output_12	M5.4	FALSE
OUT13	MyTag_Output_13	M5.5	FALSE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET, OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET, OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zu CMD

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

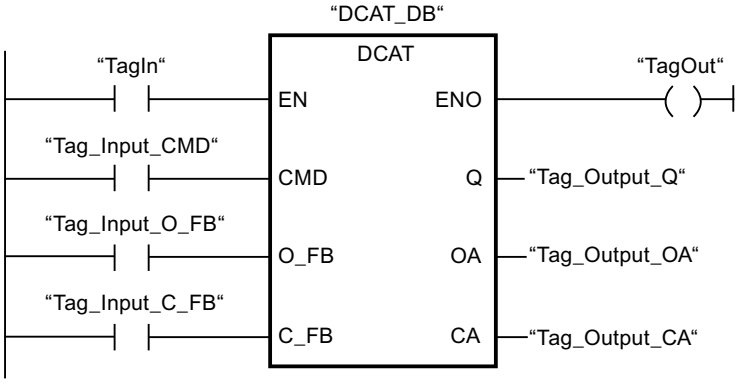

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop

Eingangsparameter								Ausgangsparameter								
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT

Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
O_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Schließen"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

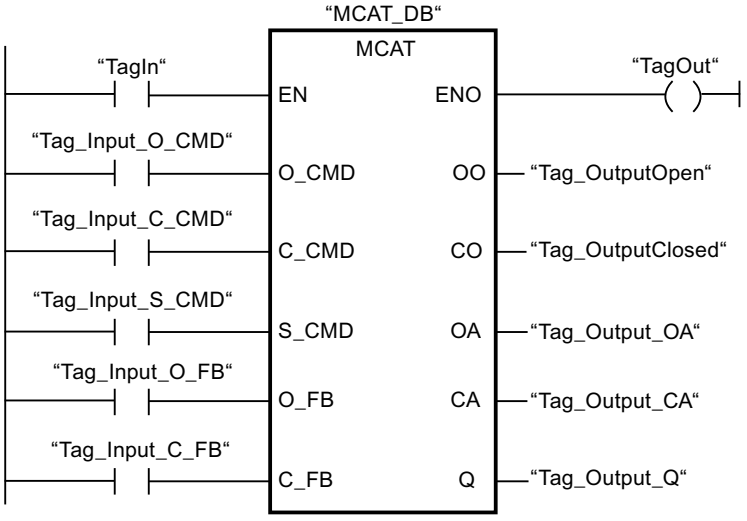

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY of WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird. Am Parameter IN können auch Konstanten angegeben werden.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit Am Parameter SAMPLE_T können auch Konstanten angegeben werden.
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit.
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

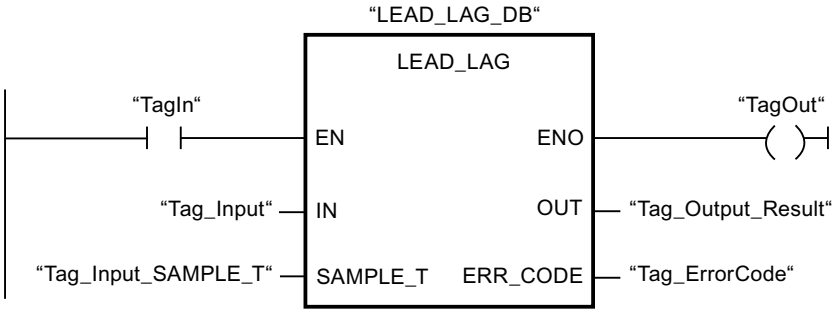

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

TONR_X: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines festgelegten Zeitraums, den Sie mithilfe des Eingangs PV angeben. Die Anweisung wird ausgeführt, wenn der Signalzustand am Starteingang TMR_EN von "0" nach "1" wechselt. Gleichzeitig läuft die Zeit mit der am Eingang PV programmierten Zeitdauer ab. Während die Anweisung ausgeführt wird, addiert sie den Zeitwert des Eingangs DELTA_T zum Zeitwert des InOut ET.

Folgende Bedingungen müssen dazu erfüllt sein:

- Der Starteingang TMR_EN liefert den Signalzustand "1".
- Der Zeitwert am InOut ET ist kleiner als der Zeitwert am Parameter PV.
- Der Eingang RESET liefert den Signalzustand "0".

Die Anweisung wird unterbrochen, wenn der Signalzustand am Starteingang TMR_EN von "1" nach "0" wechselt. Sobald der Signalzustand wieder nach "1" wechselt, läuft die Anweisung weiter.

Die Anweisung wird beendet, wenn der Zeitwert am InOut ET den Zeitwert am Eingang PV erreicht hat. Dann liefert der Ausgang Q den Signalzustand "1".

Solange der Ausgang Q den Signalzustand "1" liefert, bleibt der Zeitwert am InOut ET unverändert. Der Zeitwert und der Ausgang Q werden erst wieder auf "0" zurückgesetzt, wenn der Eingang RESET auf den Signalzustand "1" gesetzt wird.

Da die Anweisung "Zeit akkumulieren" die Zykluszeit (DELTA_T) des letzten Zyklus des Organisationsbausteins (OB) verwendet, darf sie nur in zyklischen Organisationsbausteinen aufgerufen werden.

Hinweis

Sie müssen die Zykluszeit des Organisationsbausteins aus den Anlaufinformationen in die globale Variable des Parameters DELTA_T verschieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L, T, Z	Freigabeausgang
TMR_EN (Timer Enable)	Input	BOOL	E, A, M, D, L, T, Z	Starteingang Startet die Anweisung und den Ablauf der voreingestellten Zeitdauer am Eingang PV
RESET	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
PV (Preset Value)	Input	DINT	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung
DELTA_T	Input	INT	E, A, M, D, L oder Konstante	Zykluszeit des Organisationsbausteins des vorherigen Zyklus
Q	Output	BOOL	E, A, M, D, L	Ausgang, der bei Erreichen des voreingestellten Zeitwerts gesetzt wird
ET (Elapsed Time)	InOut	DINT	E, A, M, D, L	Akkumulierter Zeitwert (Aktuell abgelaufener Zeitwert + Zeitwert des Eingangs DELTA_T)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeitwert akkumulieren":

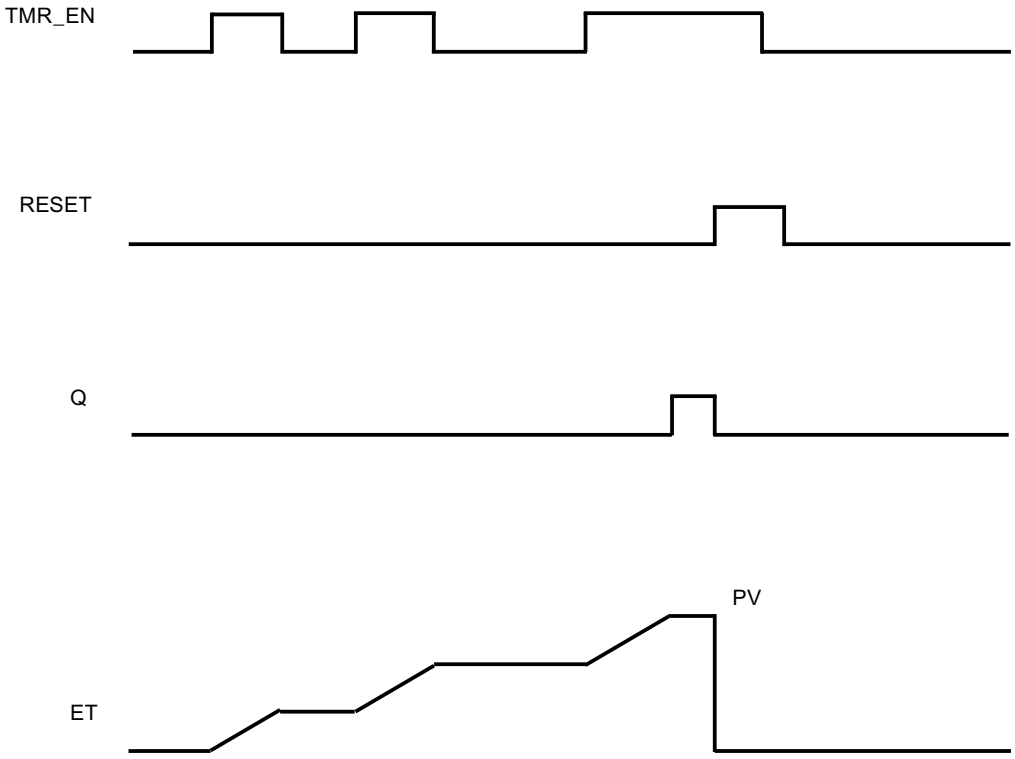

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

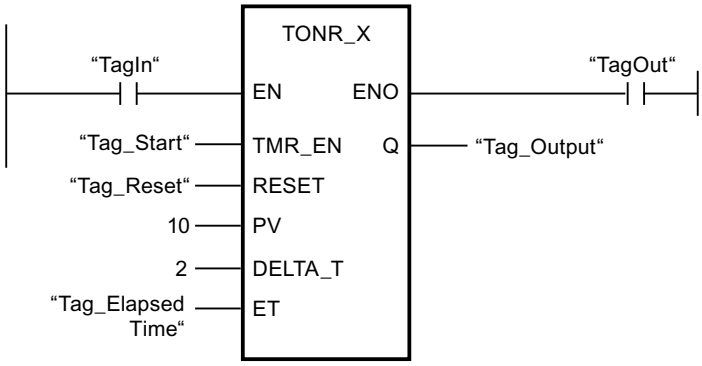

Die Anweisung wird gestartet, wenn die Operanden "TagIn" und "Tag_Start" den Signalzustand "1" liefern. Bei jedem Wechsel des Signalzustands von "0" nach "1" am Eingang TMR_EN, wird die Zykluszeit "2" am Eingang DELTA_T zum aktuellen Zeitwert des InOut ET addiert, bis der maximale Zeitwert "10" des Eingangs PV erreicht ist. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

WSR: Datum im Schieberegister speichern

Beschreibung

Mit der Anweisung "Datum im Schieberegister speichern" schreiben Sie Daten aus einer angegebenen Quelle in ein Schieberegister. Dabei wird jedes Element des Registers in die nächste Adresse verschoben. Die Daten, die in der letzten Adresse des Schieberegisters enthalten sind, gehen nach der Schiebeoperation verloren.

Bei jeder Ausführung der Anweisung mit dem Wert "0" am Parameter RESET, werden neue Daten aus der Quelle (S_DATA) gelesen und in die Anfangsadresse (START) des Schieberegisters geschoben. Der Parameter LENGTH gibt die Anzahl der zu schiebenden Elemente an. Am Parameter E_TYPE wird der Datentyp der Elemente festgelegt. Wenn am Parameter E_TYPE ein unzulässiger Datentyp angegeben wird, wird die Anweisung nicht bearbeitet.

Wenn der Parameter RESET bei der Ausführung der Anweisung auf "1" gesetzt ist, wird das Register gelöscht. Der Parameter Q wird auf den Signalzustand "1" gesetzt, wenn das Schieberegister mit Nullen belegt ist oder gelöscht wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datum im Schieberegister speichern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_DATA	Input	POINTER	E, A, M, D	Zeigt auf das Quellelement, das in die Anfangsadresse (START) geschoben wird. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
START	Input	POINTER	E, A, M, D	Zeigt auf die Anfangsadresse des Registers. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
LEN	Input	WORD	E, A, M, D, L, P	Anzahl der Elemente, die geschoben werden.
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der zu schiebenden Elemente an. Folgende Datentypen sind zulässig: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	A, M, D, L	Gibt "0" an, wenn der Parameter RESET aktiv (1) ist oder alle Elemente, die geschoben werden, den Wert "0" haben.

Beispiel

Im folgenden Beispiel werden vier Elemente des Datentyps WORD ab der am Parameter START angegebenen Adresse geschoben. Nachdem das erste Element des Registers in die nächste Adresse geschoben wurde, wird die erste Adresse mit den Daten belegt, auf die der Zeiger am Parameter S_DATA zeigt. Der Wert des letzten Elements im Register geht dabei verloren.

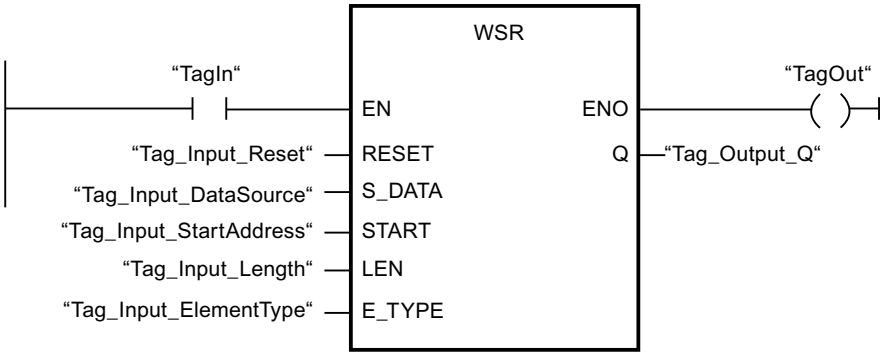

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

SHRB: Bit in Bitschieberegister schieben

Beschreibung

Mit der Anweisung "Bit in Bitschieberegister schieben" können Sie ein Bit aus einer angegebenen Quelle (DATA) in ein Schieberegister schreiben. Bei jeder Ausführung der Anweisung mit einem Wert "0" am Parameter RESET werden neue Daten aus der Quelle gelesen und in die Anfangsadresse (S_BIT) des Schieberegisters geschoben. Alle folgenden Bits werden um eins verschoben. Das Bit in der letzten Adresse (S_BIT + N) geht nach dem Schieben verloren. Wenn der Parameter RESET während der Bearbeitung der Anweisung auf "1" gesetzt ist, werden die Adressen im Schieberegister auf "0" zurückgesetzt und nicht geschoben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bit in Bitschieberegister schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DATA	Input	BOOL	E, A, M, D, L	Quellbit
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das Anfangsbit des Schieberegisters. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	WORD	E, A, M, D, L, P	Anzahl der Bits, die geschoben werden.

Beispiel

Im folgenden Beispiel wird der Parameter N auf den Wert 14 ("E" in hexadezimaler Schreibweise) gesetzt, sodass 14 Bits ab der am Parameter S_BIT angegebenen Adresse geschoben werden. Nachdem die Bits geschoben worden sind, wird die erste Adresse mit den Daten aus der Quelle am Parameter DATA belegt. Der letzte Bitwert geht verloren.

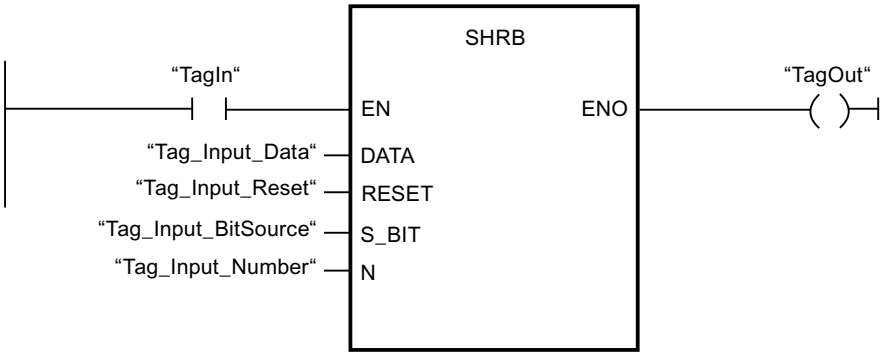

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Vor der Bearbeitung der Anweisung sind die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Nach der Bearbeitung

Nach der Bearbeitung der Anweisung werden die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

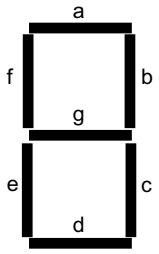

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

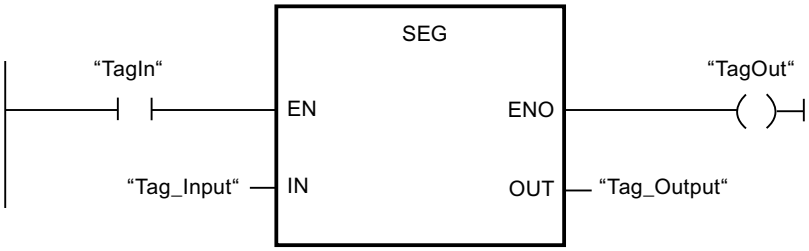

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

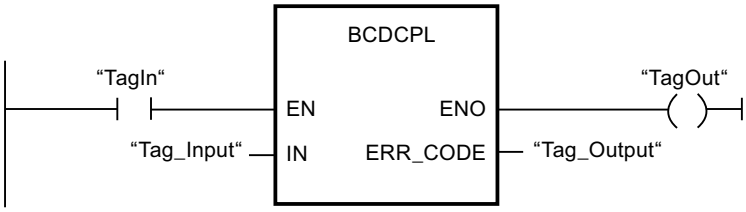

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

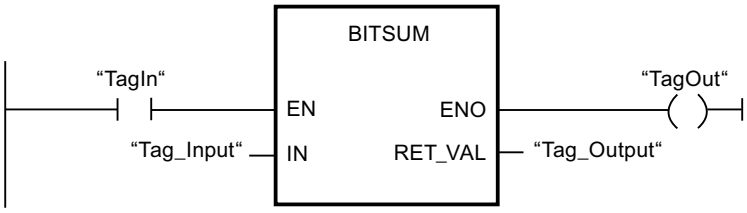

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu KOP (Seite 7425)

4.2.2.2 FUP

Bitverknüpfungen

&: UND-Verknüpfung

Beschreibung

Mit der Anweisung "UND-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der UND-Wahrheitstabelle aus.

Beträgt der Signalzustand aller Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand eines Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

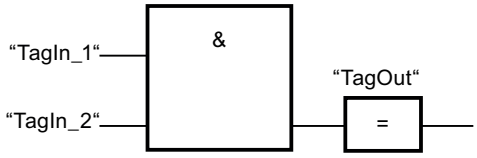

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand der Operanden "TagIn_1" und "TagIn_2" "1" ist und zurückgesetzt, wenn der Signalzustand der Operanden "TagIn_1" und "TagIn_2" "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

UND-Wahrheitstabelle (Seite 4592)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

UND-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch UND entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	1	1
0	1	0
1	0	0
0	0	0

Siehe auch

&: UND-Verknüpfung (Seite 4591)

>=1: ODER-Verknüpfung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der ODER–Wahrheitstabelle aus.

Beträgt der Signalzustand eines der Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand aller Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

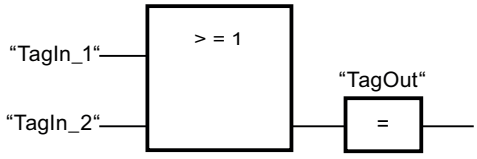

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand des Operanden "TagIn_1" oder des Operanden "TagIn_2" "1" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

ODER-Wahrheitstabelle (Seite 4593)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

ODER-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	0	1
0	1	1
1	1	1
0	0	0

Siehe auch

>=1: ODER-Verknüpfung (Seite 4592)

X: EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" fragen Sie das Ergebnis einer Signalzustandsabfrage entsprechend der EXKLUSIV ODER-Wahrheitstabelle ab.

Bei der Anweisung "EXKLUSIV ODER-Verknüpfung" ist der Signalzustand "1", wenn der Signalzustand eines der beiden angegebenen Operanden "1" ist. Bei der Abfrage von mehr als zwei Operanden ist das gemeinsame Verknüpfungsergebnis "1", wenn eine ungerade Anzahl der abgefragten Operanden das Abfrageergebnis "1" liefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

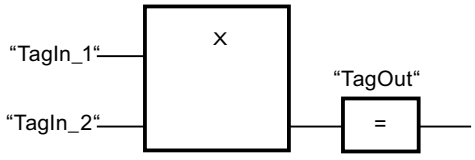

Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand eines der beiden Operanden "TagIn_1" und "TagIn_2" "1" ist. Wenn beide Operanden den Signalzustand "1" oder "0" liefern, wird der Ausgang "TagOut" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

EXKLUSIV ODER-Wahrheitstabelle (Seite 4594)

Grundlagen zu FUP (Seite 7493)

EXKLUSIV ODER-Wahrheitstabelle

Verknüpfungsergebnisse

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch EXKLUSIV ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Verknüpfungsergebnis
1	0	1
0	1	1
1	1	0
0	0	0

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von drei Operanden durch EXKLUSIV ODER entstehen:

Signalzustand des ersten Operanden	Signalzustand des zweiten Operanden	Signalzustand des dritten Operanden	Verknüpfungsergebnis
1	0	0	1
0	1	1	0
0	1	0	1
1	0	1	0
0	0	1	1
1	1	0	0
1	1	1	1
0	0	0	0

Siehe auch

X: EXKLUSIV ODER-Verknüpfung (Seite 4593)

Eingang einfügen

Beschreibung

Mit der Anweisung "Eingang einfügen" erweitern Sie die Box einer der folgenden Anweisungen durch einen Eingang:

- "UND-Verknüpfung"
- "ODER-Verknüpfung"
- "EXKLUSIV ODER-Verknüpfung"

Durch die Erweiterung einer Anweisungsbox können Sie den Signalzustand mehrerer Operanden abfragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingang einfügen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

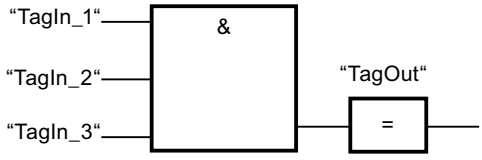

Die Box der Anweisung wurde durch einen zusätzlichen Eingang erweitert, an dem der Signalzustand des Operanden "TagIn_3" abgefragt wird. Der Ausgang "TagOut" wird gesetzt, wenn die Operanden "TagIn_1", "TagIn_2" und "TagIn_3" den Signalzustand "1" liefern.

Siehe auch

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zum Statuswort (Seite 183)

&: UND-Verknüpfung (Seite 4591)

>=1: ODER-Verknüpfung (Seite 4592)

X: EXKLUSIV ODER-Verknüpfung (Seite 4593)

Grundlagen zu FUP (Seite 7493)

VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE).

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

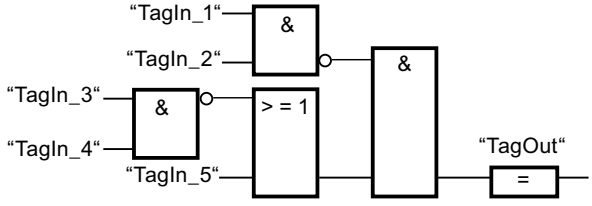

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Eingang "TagIn_1" und/oder "TagIn_2" liefert den Signalzustand "0".
- Der Eingang "TagIn_3" und/oder "TagIn_4" liefert den Signalzustand "0" oder der Eingang "TagIn_5" liefert den Signalzustand "1".

Siehe auch

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

=: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" setzten Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Boxeingang "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Boxeingang wird direkt dem über der Zuweisen-Box stehenden Operanden zugewiesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

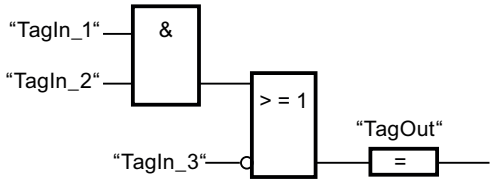

Der Operand "TagOut" am Ausgang der Anweisung "Zuweisung" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Eingänge "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand am Eingang "TagIn_3" ist "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

R: Ausgang rücksetzen

Beschreibung

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "0" rückgesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, T, Z, D, L	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

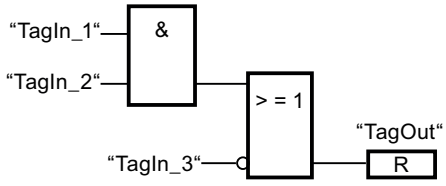

Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn_3" ist "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

S: Ausgang setzen

Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

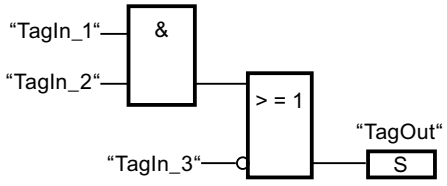

Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".
- Der Operand "TagIn_3" liefert den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

SR: Flipflop setzen/rücksetzen

Beschreibung

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S	Input	BOOL	E, A, M, D, L, T, Z	Setzen freigeben
R1	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	Operand, der gesetzt oder rückgesetzt wird
Q	Output	BOOL	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

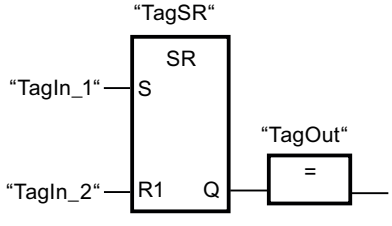

Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagSR" und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zu FUP (Seite 7493)

RS: Flipflop rücksetzen/setzen

Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen R und S1 wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R	Input	BOOL	E, A, M, T, Z, D, L	Rücksetzen freigeben
S1	Input	BOOL	E, A, M, T, Z, D, L	Setzen freigeben
<Operand>	InOut	BOOL	E, A, M, D, L	Operand, der rückgesetzt oder gesetzt wird
Q	Output	BOOL	E, A, M, D, L	Signalzustand des Operanden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

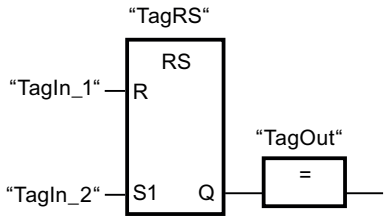

Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Der Operand "TagIn_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "0" und der Operand "TagIn_2" liefert den Signalzustand "1".
- Die Operanden "TagIn_1" und "TagIn_2" liefern den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zu FUP (Seite 7493)

P: Operand auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

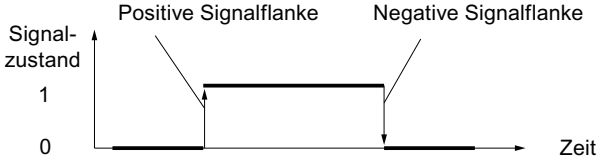

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BOOL	E, A, M, T, Z, D, L	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

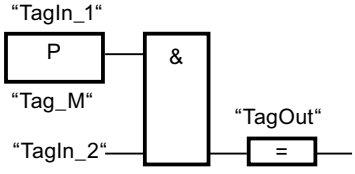

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn_1" liegt eine positive Signalflanke vor.
- Der Signalzustand des Operanden "TagIn_2" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

N: Operand auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "Operand auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im Signalzustand bei einer negativen und einer positiven Signalflanke:

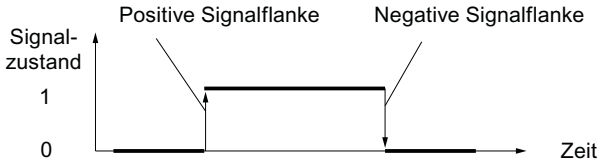

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BOOL	E, A, M, T, Z, D, L	Abzufragendes Signal
<Operand2>	InOut	BOOL	E, A, M, D, L	Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

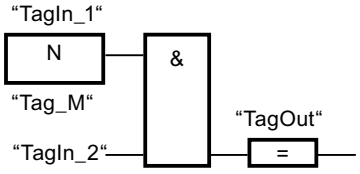

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn_1" liegt eine negative Signalflanke vor.
- Der Signalzustand des Operanden "TagIn_2" ist "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zu FUP (Seite 7493)

P_TRIG: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1",

sobald eine positive Signalflanke erfasst wird. In allen anderen Fällen liefert der Ausgang den Signalzustand "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

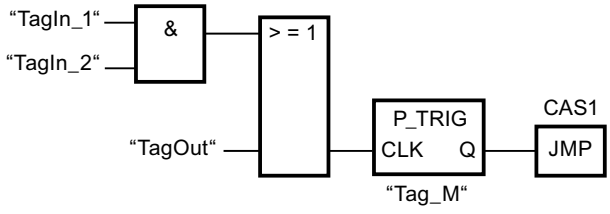

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zu FUP (Seite 7493)

N_TRIG: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker (<Operand>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Der Ausgang Q der Anweisung liefert einen Programmzyklus lang den Signalzustand "1", sobald eine negative Signalflanke erfasst wird. In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CLK	Input	BOOL	E, A, M, D, L	Aktuelles VKE
<Operand>	InOut	BOOL	M, D	Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird.
Q	Output	BOOL	E, A, M, D, L	Ergebnis der Flankenauswertung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

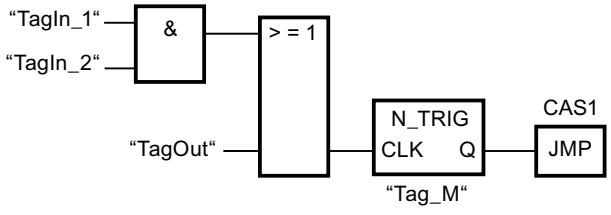

Im Flankenmerker "Tag_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zu FUP (Seite 7493)

Zeiten

IEC-Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TP (z. B. "TP_DB")
- Deklaration als lokale Variable vom Typ TP im Abschnitt "Static" eines Bausteins (z. B. #MyTP_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines

anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, T, Z	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Impulsausgang
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

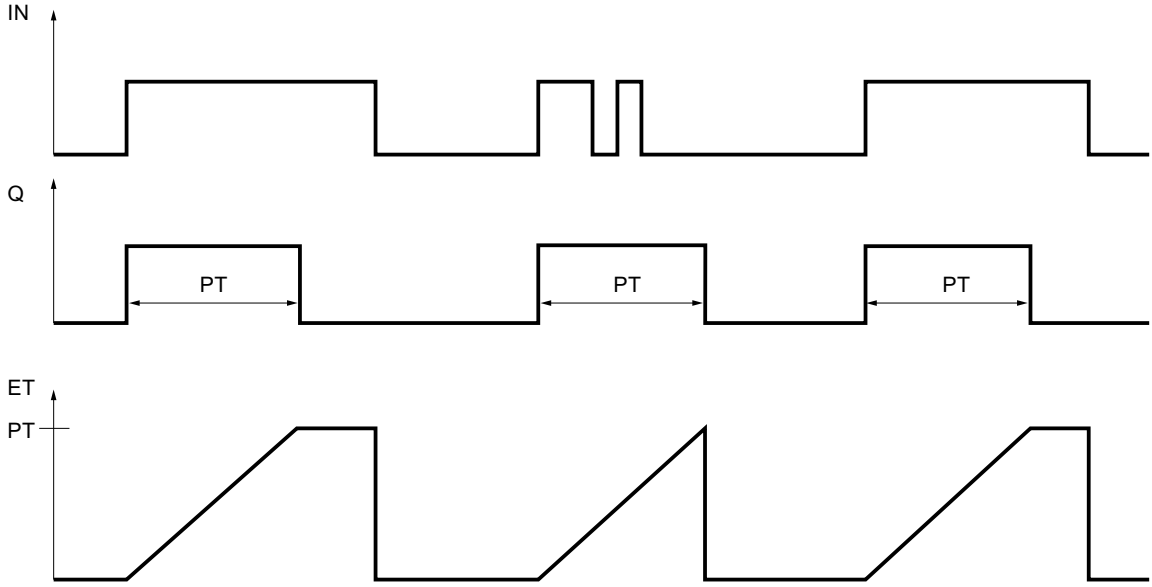

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TON (z. B. "TON_DB")
- Deklaration als lokale Variable vom Typ TON im Abschnitt "Static" eines Bausteins (z. B. #MyTON_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, T, Z	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

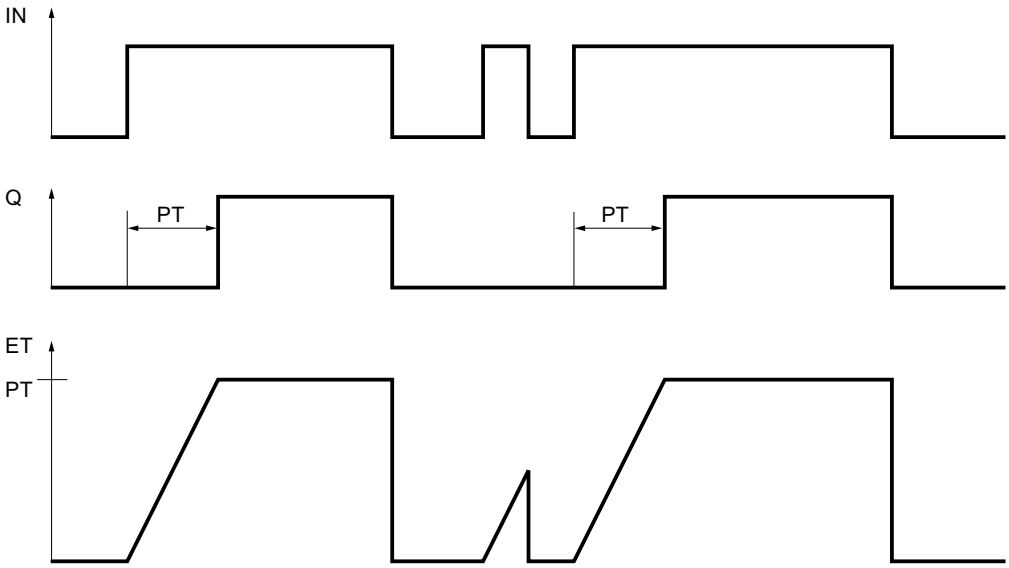

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der aktuelle Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang

IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TOF (z. B. "TOF_DB")
- Deklaration als lokale Variable vom Typ TOF im Abschnitt "Static" eines Bausteins (z. B. #MyTOF_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L, T, Z	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

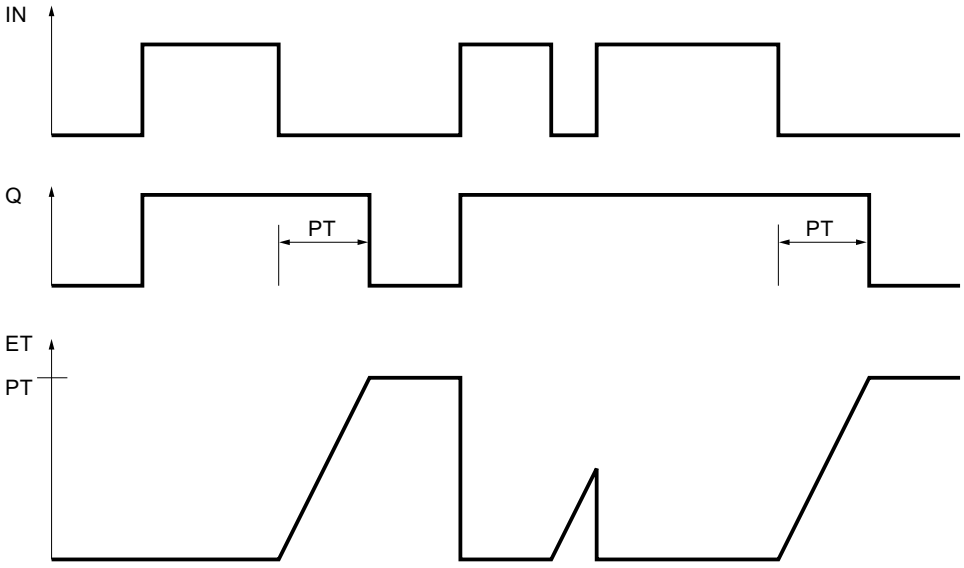

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

SIMATIC-Zeiten

S_IMPULS: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte

Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

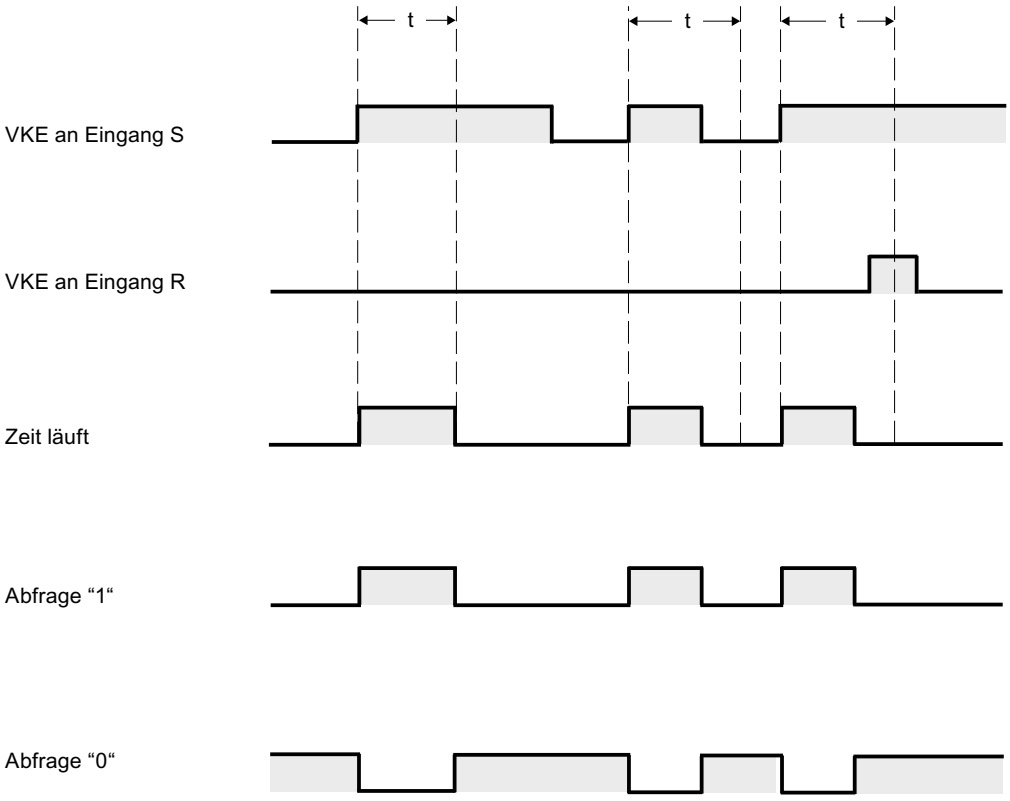

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

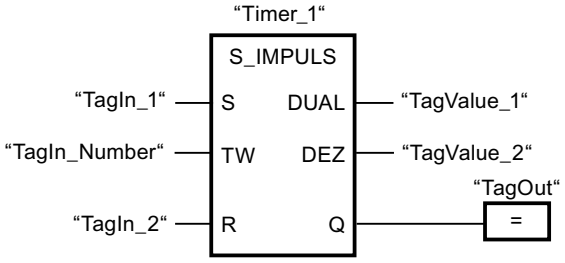

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Operand "TagIn_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

S_VIMP: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

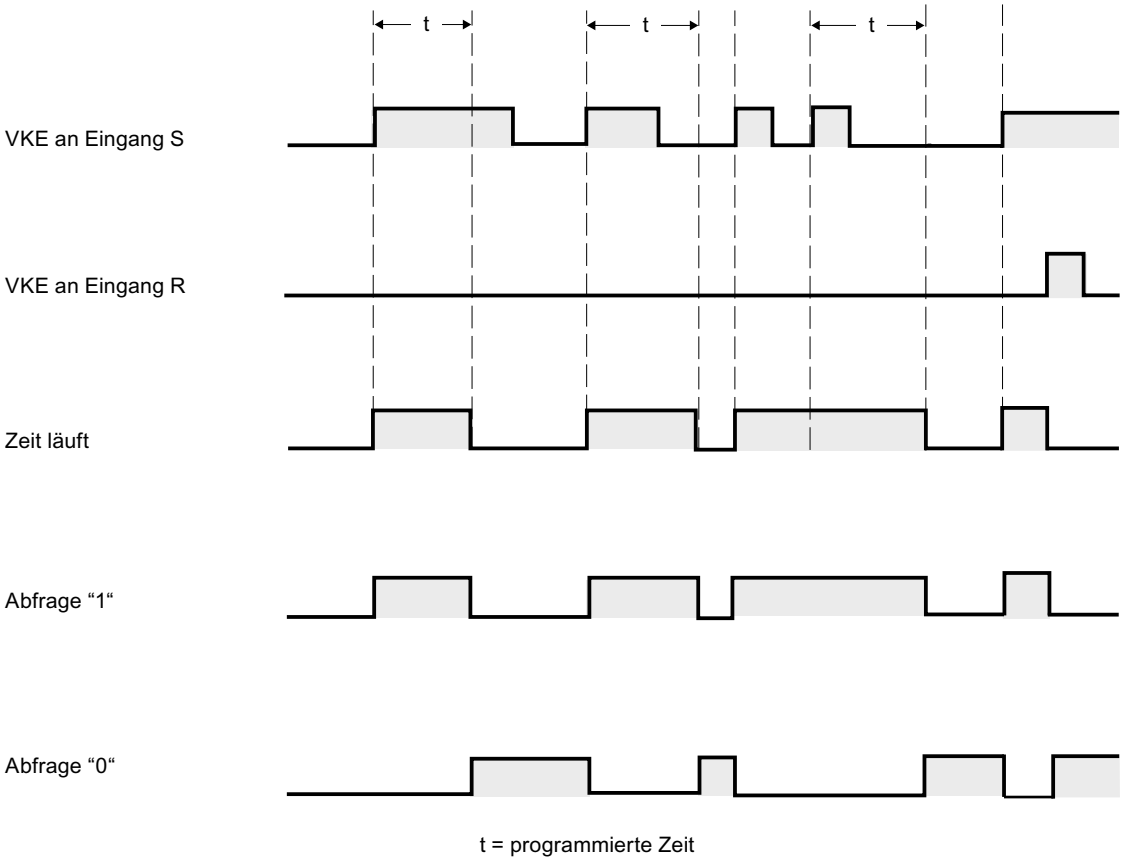

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

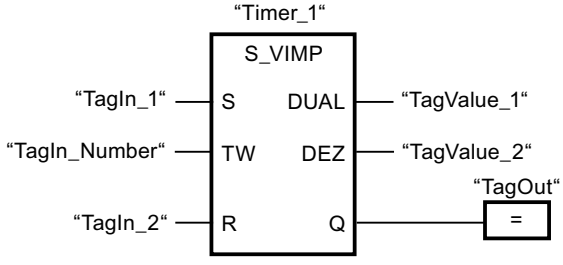

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

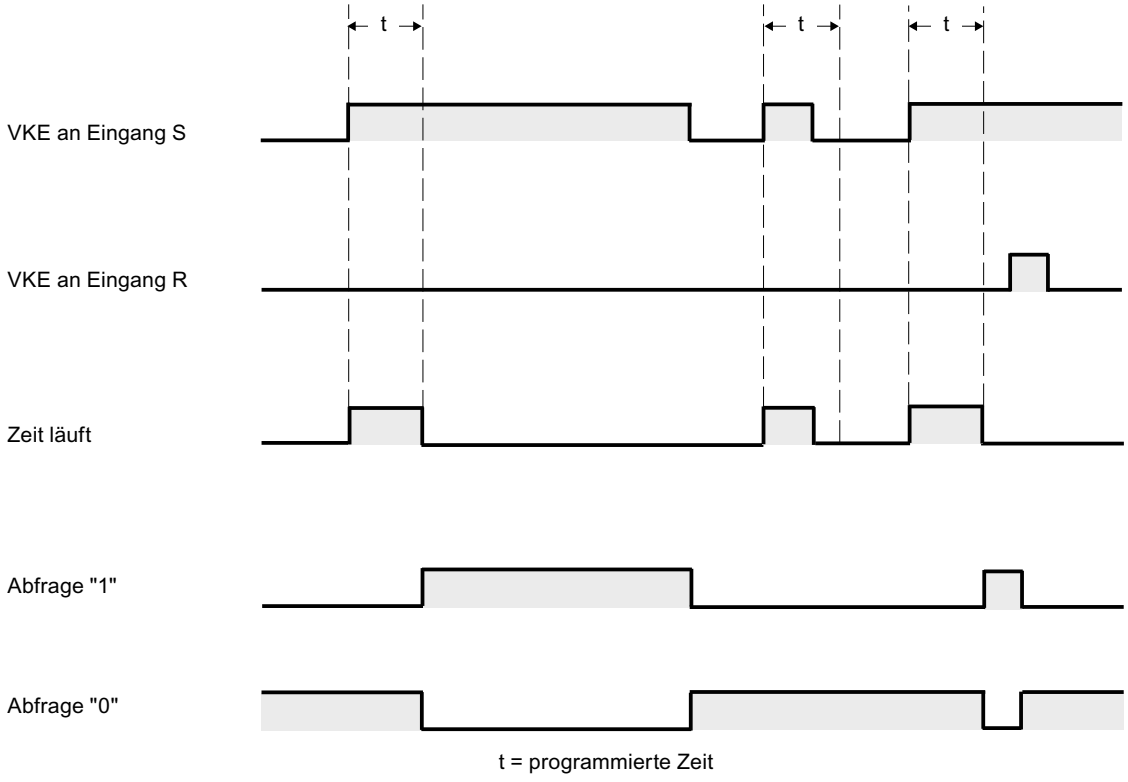

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

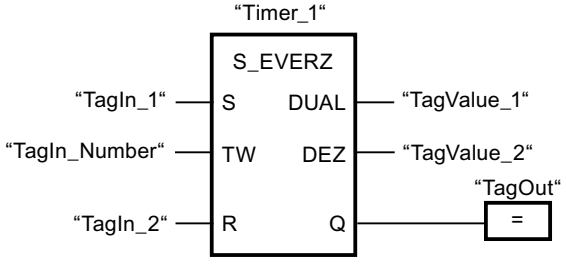

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

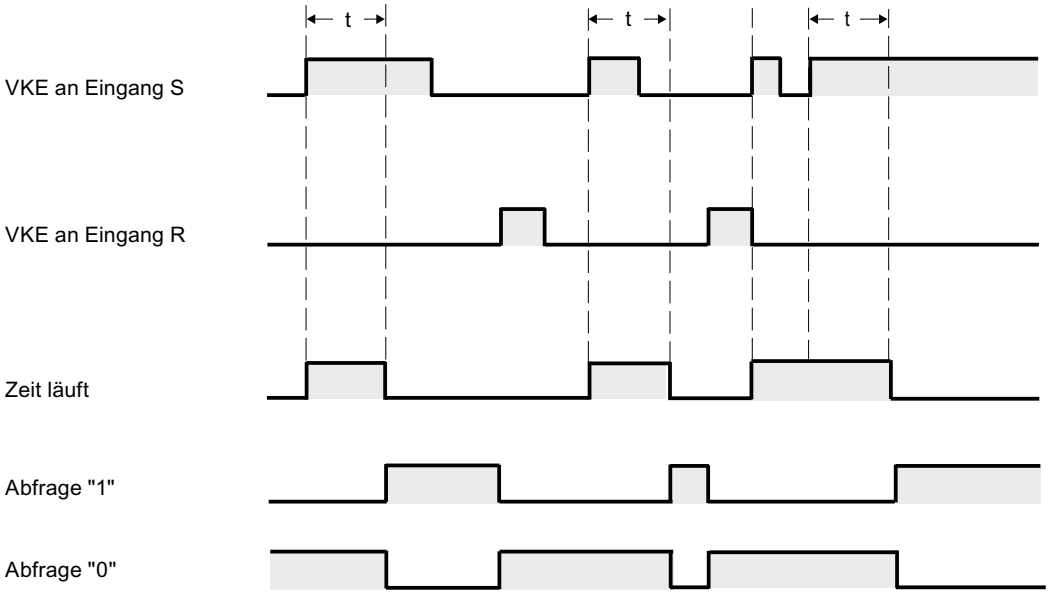

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

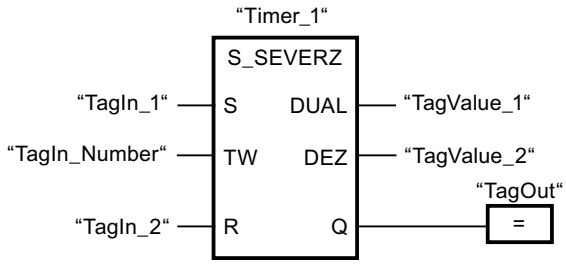

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, auch wenn der Signalzustand des Operanden "TagIn_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	InOut/Input	TIMER	T	Zeit der Anweisung Die Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
R	Input	BOOL	E, A, M, T, Z, D, L, P	Rücksetzeingang
DUAL	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (Dualcodiert)
DEZ	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

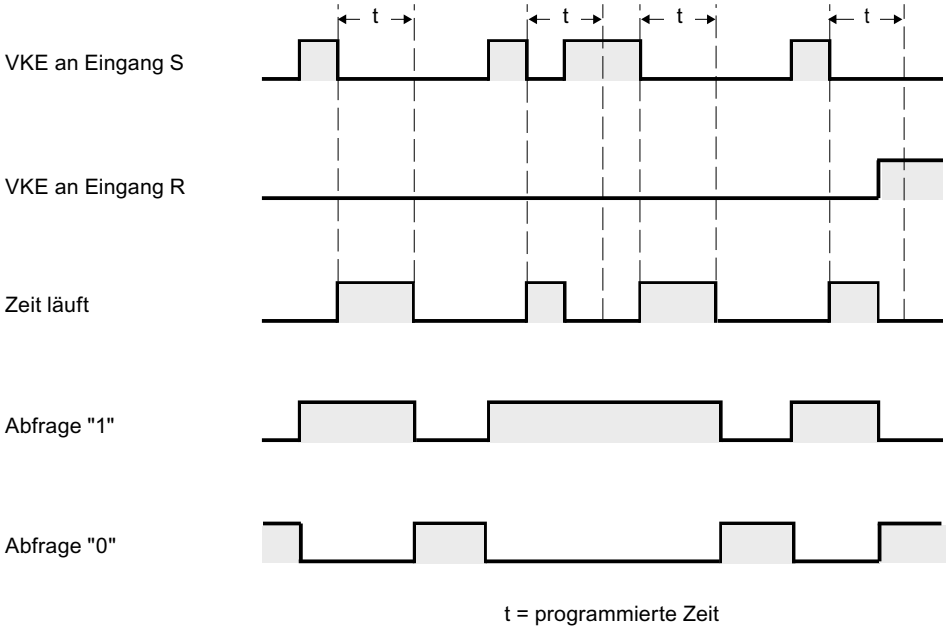

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

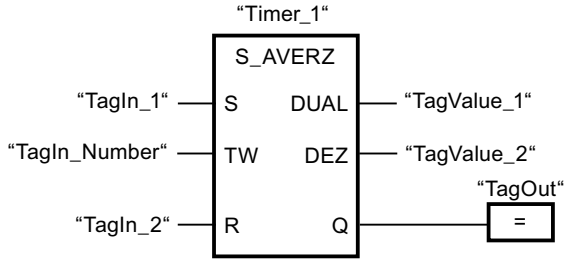

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

SI: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

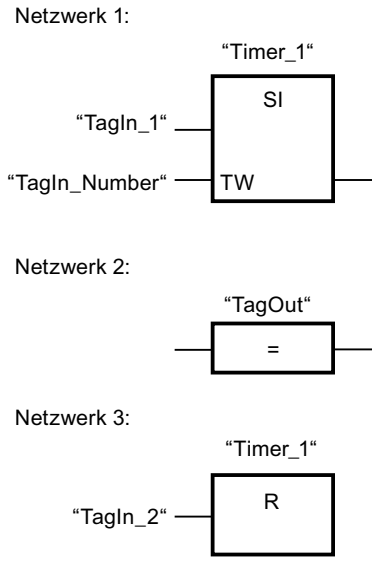

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, solange der Signalzustand des Operanden "TagIn_1" "1" ist. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_2" von "0" auf "1", dann wird die Zeit zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

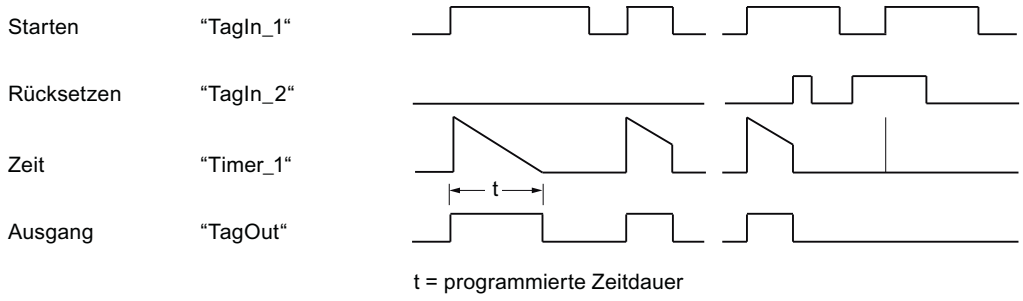

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu FUP (Seite 7493)

SV: Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer neu gestartet. Die Abfrage des Zeitstatus auf "1" ergibt den Signalzustand "0", wenn die Zeit abgelaufen ist.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

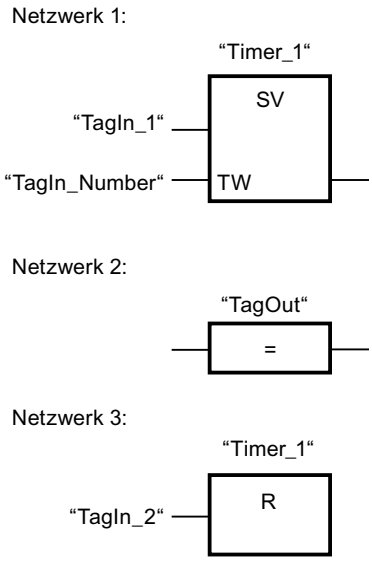

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab, ohne von einer negativen Signalflanke im VKE beeinflusst zu werden. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn_1" erneut von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

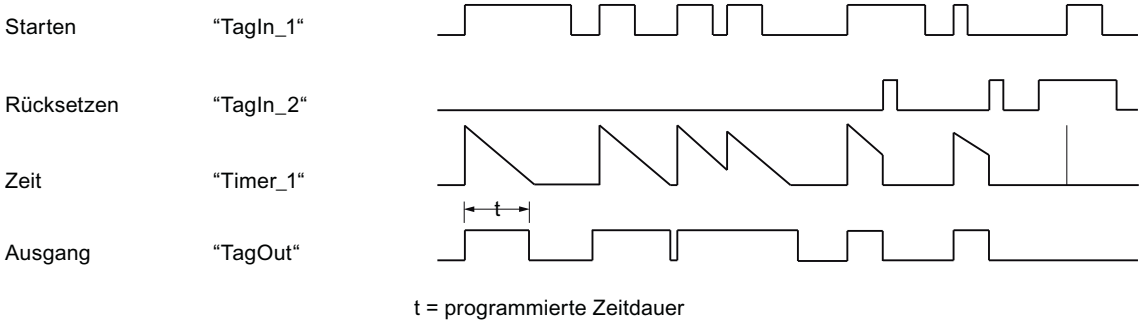

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SE: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der angegebenen Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch L: Zeitwert laden (Seite 4945).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1:

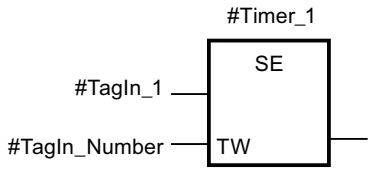

Wenn der Signalzustand des Operanden #TagIn_1 von "0" auf "1" wechselt, wird die Zeit #Timer_1 gestartet. Die Zeit läuft mit dem Zeitwert des Operanden #TagIn_Number ab. Wechselt der Signalzustand des Operanden #TagIn_1 von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit zurückgesetzt.

Netzwerk 2:

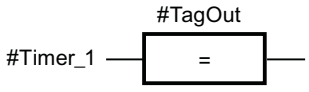

Wenn die Zeit abgelaufen ist, der Operand #TagIn_1 am Starteingang den Signalzustand "1" hat und die Zeit nicht zurückgesetzt wurde, dann ist der Operand #TagOut "1".

Netzwerk 3:

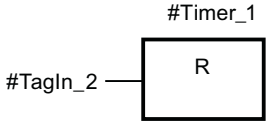

Wenn der Signalzustand des Operanden #TagIn_2 den Signalzustand "1" hat, dann wird die Zeit #Timer_1 und der Ausgang #TagOut zurückgesetzt.

Wenn Sie den #Timer_1 erneut starten wollen, dann muss der Signalzustand am Operand #TagIn_2 "0" sein und der Signalzustand am Starteingang #TagIn_1 von "0" auf "1" wechseln.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

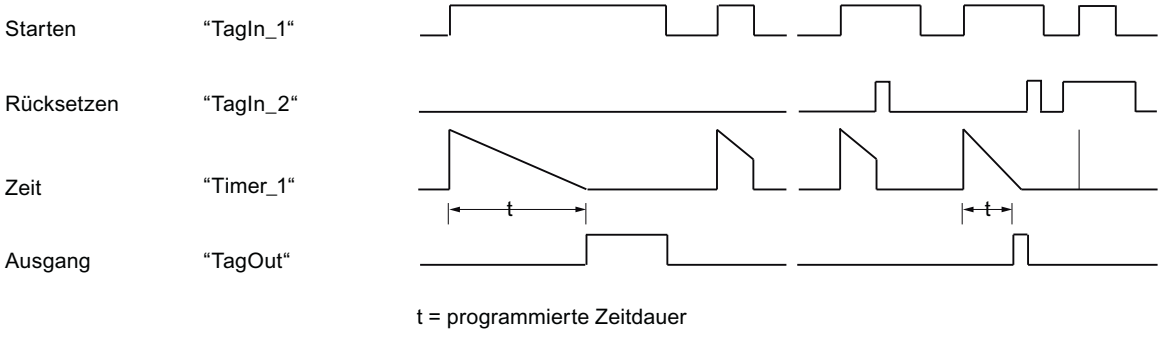

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SS: Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

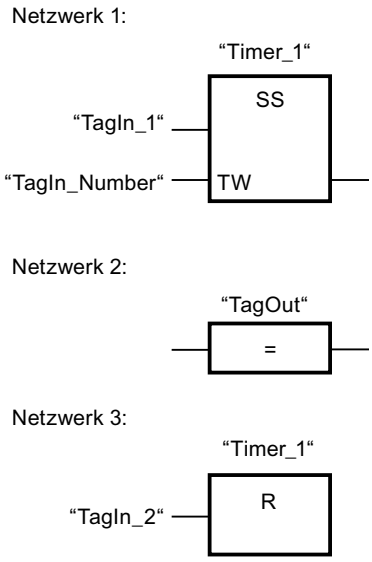

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

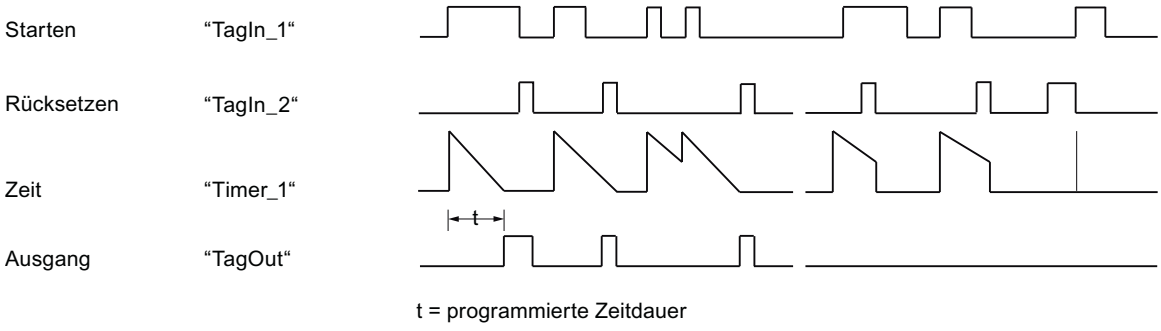

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

SA: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist. Wenn das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L, P	Starteingang
TW	Input	S5TIME, WORD	E, A, M, D, L oder Konstante	Zeitdauer
<Zeit>	Output	TIMER	T	Zeit, die gestartet wird. Die Anzahl der Zeiten ist von der CPU abhängig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

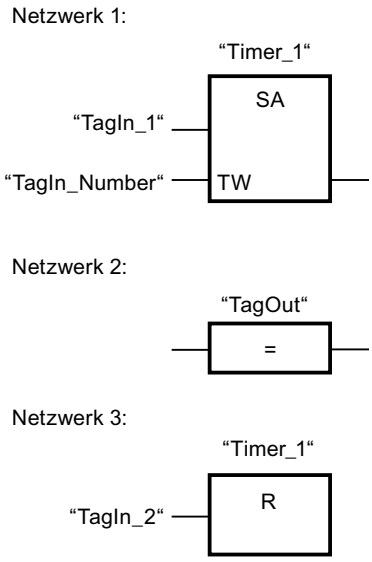

Wenn der Signalzustand des Operanden "TagIn_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn_Number" ab. Solange die Zeit läuft, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn_1" von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn_2" "1" ist, dann wird die Zeit "Timer_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

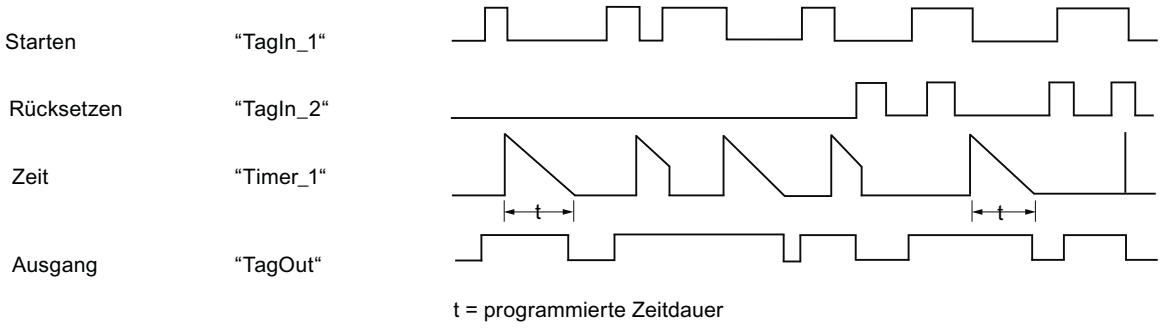

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu FUP (Seite 7493)

Zähler

IEC-Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf Null zurückgesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTU (z. B. "CTU_DB")
- Deklaration als lokale Variable vom Typ CTU im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

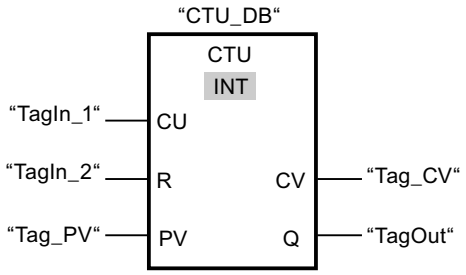

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CV" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps (INT) erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Eingang LD auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTD (z. B. "CTD_DB")
- Deklaration als lokale Variable vom Typ CTD im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, T, Z, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

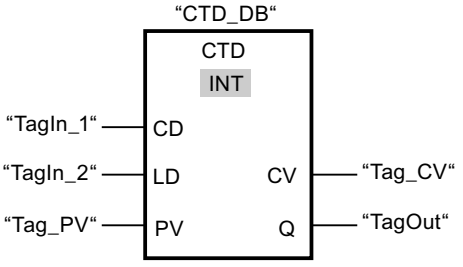

Wenn der Signalzustand des Operanden "TagIn_1" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert am Ausgang "Tag_CV" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des Datentyps (INT = -32768) erreicht ist.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu FUP (Seite 7493)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des angegebenen Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hochgezählt. Wenn der untere Grenzwert des angegebenen Datentyps (INT) erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Ausgang QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTUD (z. B. "CTUD_DB")
- Deklaration als lokale Variable vom Typ CTUD im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein

sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, T, Z, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

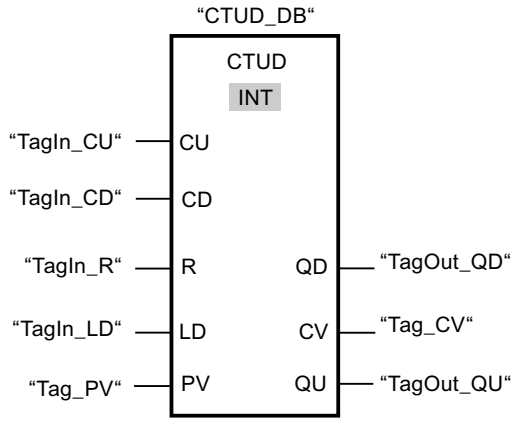

Wenn der Signalzustand am Eingang "TagIn_CU" oder am Eingang "TagIn_CD" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_CU" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn_CD" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von INT = 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "TagOut_QU" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut_QU" den Signalzustand "0".

Der Ausgang "TagOut_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut_QD" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

SIMATIC-Zähler

Z_VORW: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L, T, Z	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

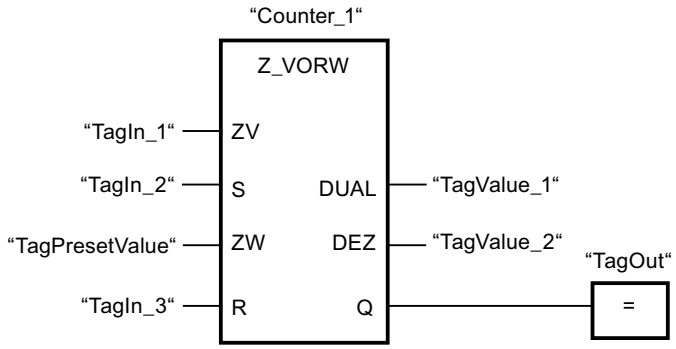

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Z_RUECK: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZR	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

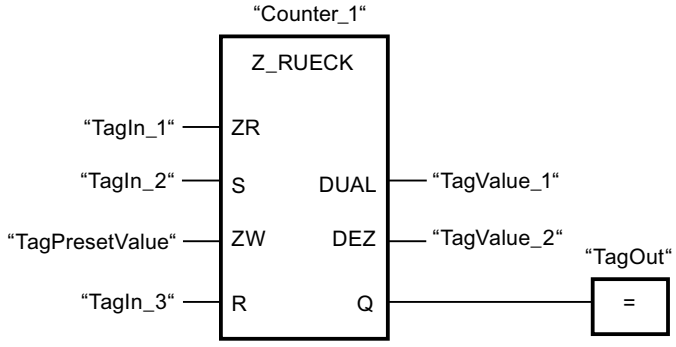

Wenn der Signalzustand am Eingang "TagIn_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zähler>	InOut	COUNTER	Z	Zähler der Anweisung Die Anzahl der Zähler ist von der CPU abhängig.
ZV	Input	BOOL	E, A, M, D, L, T, Z	Vorwärtszähleingang
ZR	Input	BOOL	E, A, M, D, L, T, Z	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L, T, Z	Eingang zum Voreinstellen des Zählers
ZW	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999)
R	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
DUAL	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)
DEZ	Output	WORD, S5TIME, DATE	E, A, M, D, L, P	Aktueller Zählwert (BCD-Format)
Q	Output	BOOL	E, A, M, D, L	Status des Zählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

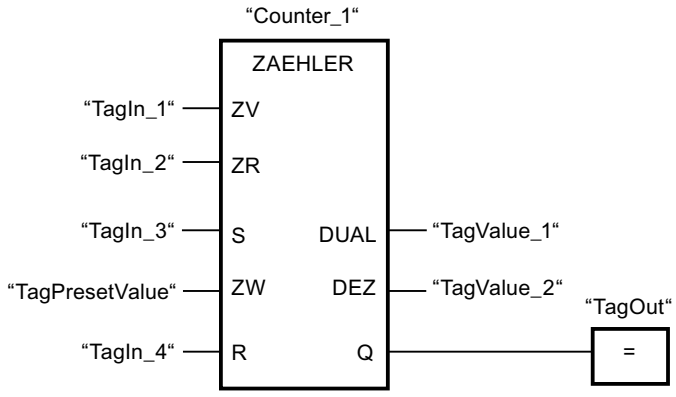

Wenn der Signalzustand am Eingang "TagIn_1" oder am Eingang "TagIn_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke am Eingang "TagIn_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue_1" und BCD-codiert im Operanden "TagValue_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SZ: Zähleranfangswert setzen

Beschreibung

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Starteingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
ZW	Input	WORD	E, A, M, D, L oder Konstante	Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999)
<Zähler>	InOut/Input	COUNTER	Z	Zähler, der voreingestellt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

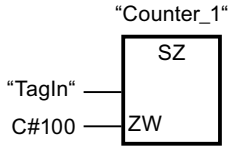

Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ZV: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

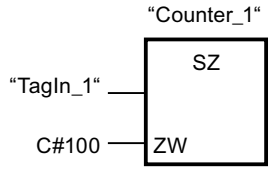

Netzwerk 2

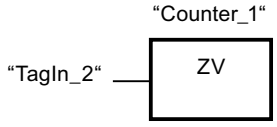

Netzwerk 3

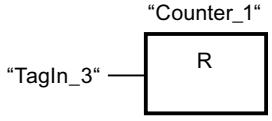

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ZR: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, T, Z, D, L	Starteingang
<Zähler>	InOut/Input	COUNTER	Z	Zähler, dessen Wert verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

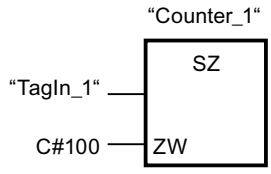

Netzwerk 2

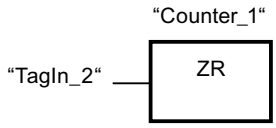

Netzwerk 3

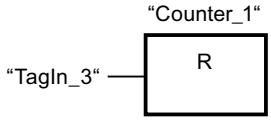

Wechselt der Signalzustand des Operanden "TagIn_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter_1" auf "0" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Vergleicher

CMP ==: Gleich

Beschreibung

Mit der Anweisung "Gleich" fragen Sie ab, ob der Wert am Eingang IN1 gleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Hinweis

Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann verwenden Sie statt der Anweisung "CMP ==: Gleich" die Anweisung "IN_RANGE: Wert innerhalb Bereich".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

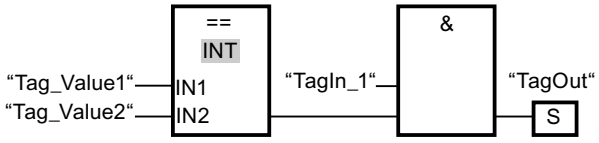

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" = "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

CMP <>: Ungleich

Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der Wert am Eingang IN1 ungleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

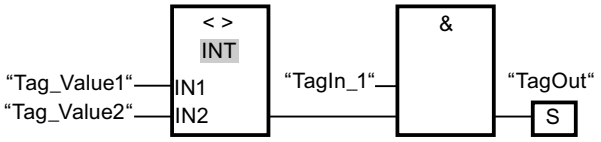

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <> "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

CMP >=: Größer gleich

Beschreibung

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der Wert am Eingang IN1 größer oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

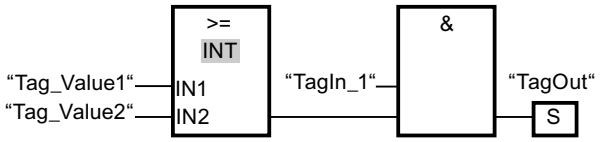

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" >= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

CMP <=: Kleiner gleich

Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der Wert am Eingang IN1 kleiner oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

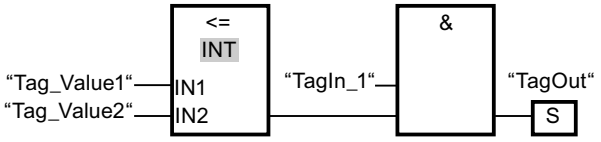

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" <= "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

CMP >: Größer

Beschreibung

Mit der Anweisung "Größer" fragen Sie ab, ob der Wert am Eingang IN1 größer als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

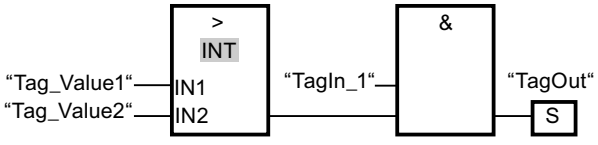

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" > "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

CMP <: Kleiner

Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der am Eingang IN1 kleiner als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Vergleichswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Vergleichswert

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

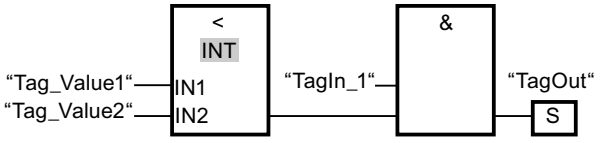

Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag_Value1" < "Tag_Value2" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Programmierbeispiele FUP (Seite 7543)

Mathematische Funktionen

ADD: Addieren

Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen die Summe am Ausgang OUT ab (OUT := IN1+IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Erster Summand
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Zweiter Summand
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Summe

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

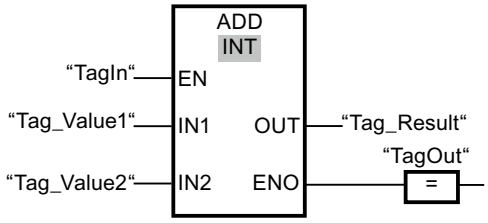

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

SUB: Subtrahieren

Beschreibung

Mit der Anweisung "Subtrahieren" subtrahieren Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 und fragen die Differenz am Ausgang OUT ab (OUT := IN1-IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Minuend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Subtrahend
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Differenz

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

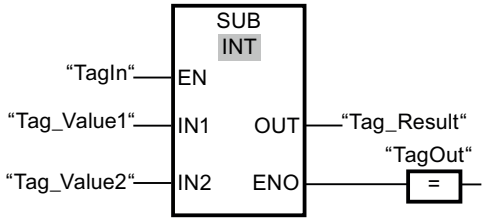

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value2" wird vom Wert des Operanden "Tag_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

MUL: Multiplizieren

Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen das Produkt am Ausgang OUT ab (OUT := IN1*IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikator
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Multiplikand
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Produkt

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

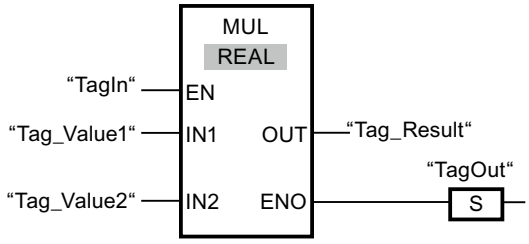

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" multipliziert. Das Multiplikationsergebnis wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Übersicht über die gültigen Datentypen (Seite 243)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

DIV: Dividieren

Beschreibung

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab (OUT := IN1/IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Dividend
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Wert des Quotienten

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

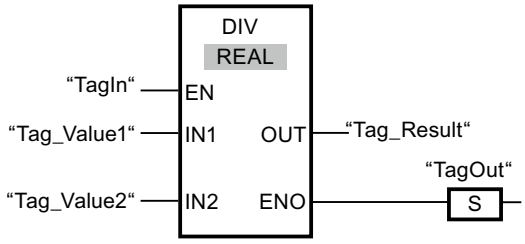

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Dividieren" ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Programmierbeispiele FUP (Seite 7543)

Grundlagen zu FUP (Seite 7493)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	DINT	E, A, M, D, L, P oder Konstante	Dividend

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN2	Input	DINT	E, A, M, D, L, P oder Konstante	Divisor
OUT	Output	DINT	E, A, M, D, L, P	Divisionsrest

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

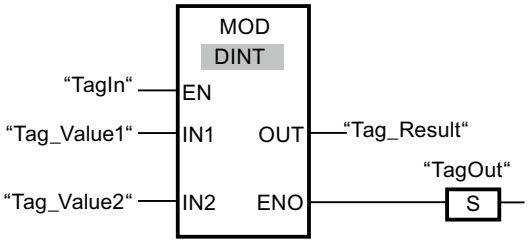

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird durch den Wert des Operanden "Tag_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag_Result" gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist (gilt nur für Ganzzahlen).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

Parameter	Deklaration	Datentyp		Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweierkomplement des Eingangswerts

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

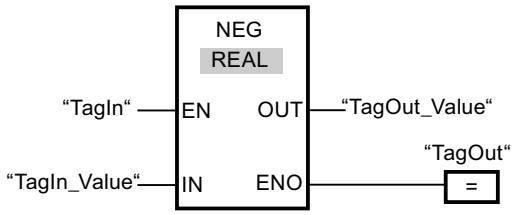

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zweierkomplement erzeugen" ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Absolutwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

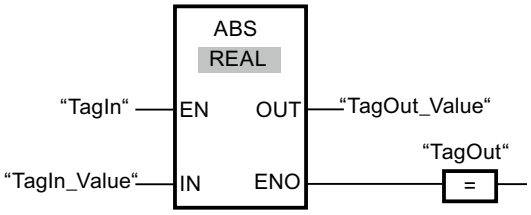

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	-6,234
OUT	TagOut_Value	6,234

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn_Value" und gibt das Ergebnis am Ausgang "TagOut_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den kleinsten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

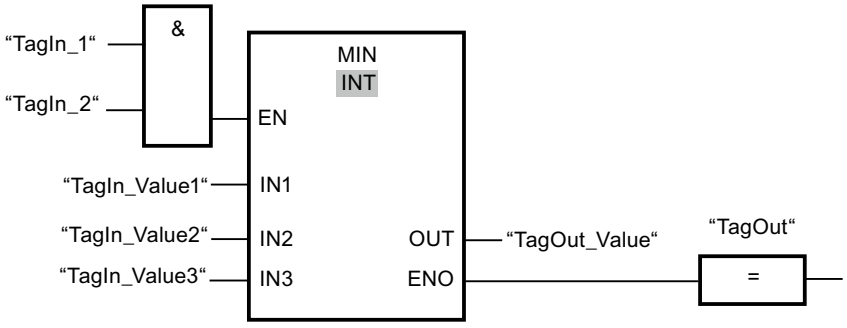

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Eingänge und Ausgänge entfernen (Seite 7464)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den größten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

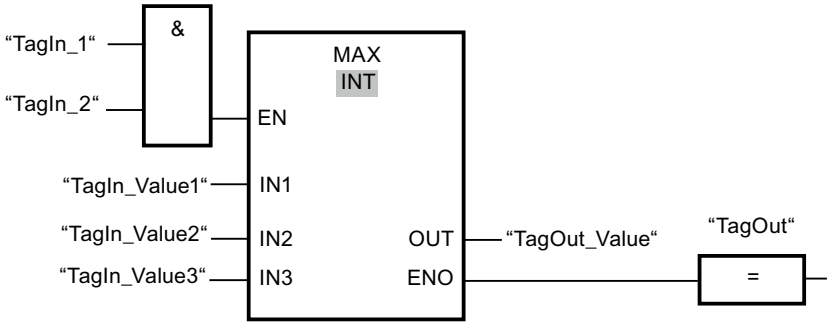

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "TagOut_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

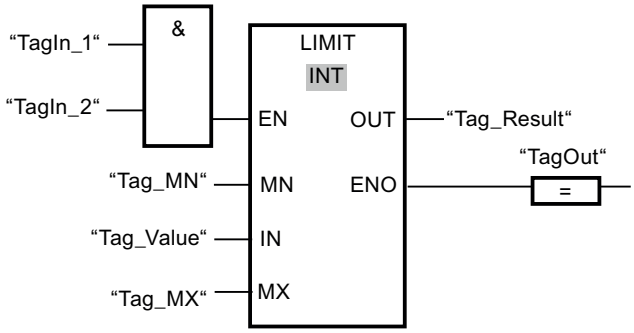

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" haben, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_MN" und "Tag_MX" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_MN" in den Ausgang "Tag_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert einer Gleitpunktzahl und schreiben das Ergebnis in den Ausgang OUT.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswertes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

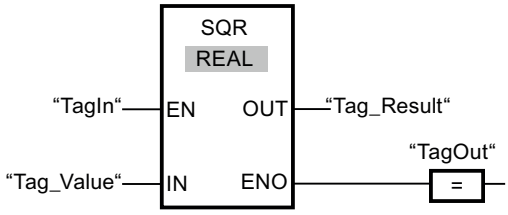

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert einer Gleitpunktzahl die Quadratwurzel und schreiben das Ergebnis in den Ausgang OUT. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

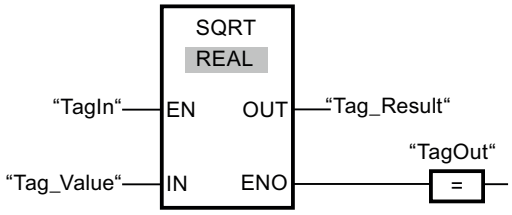

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis e (e = 2718282). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

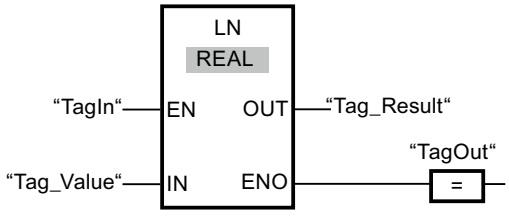

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag_Value" und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden (OUT = e^{IN}).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts IN

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

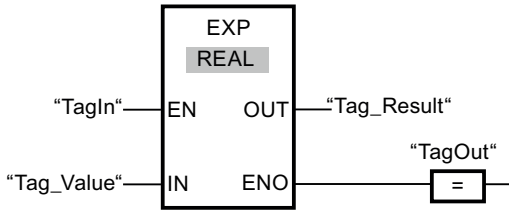

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_Value" und gibt das Ergebnis am Ausgang "Tag_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Sinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

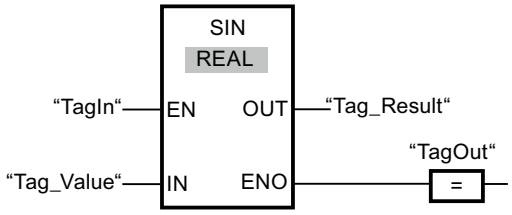

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	1.0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

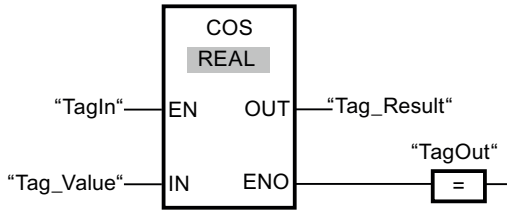

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Tangens des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

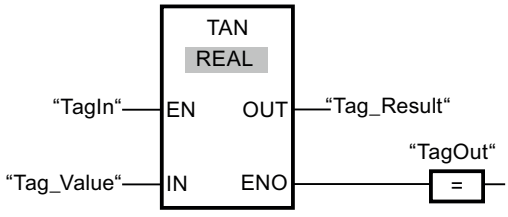

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Sinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

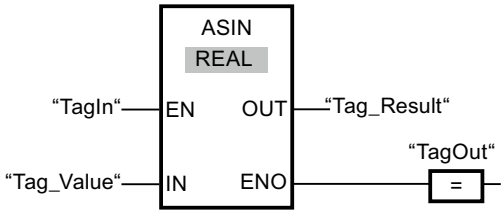

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Cosinuswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

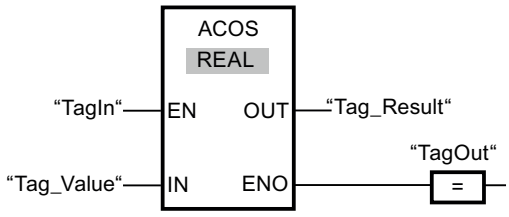

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Tangenswert
OUT	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

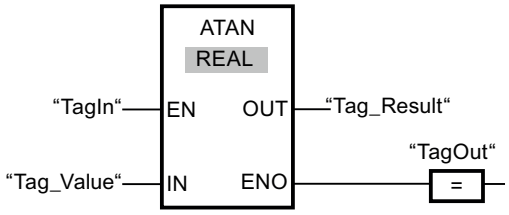

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1,0
OUT	Tag_Result	+0,785398 (π/4)

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Ungültige Gleitpunktzahlen (Seite 266)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Verschieben

MOVE: Wert kopieren

Beschreibung

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Die folgende Tabelle zeigt die möglichen Übertragungen:

Quelle (IN)	Ziel (OUT1)	
	Mit IEC-Prüfung	Ohne IEC-Prüfung
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD, CHAR
WORD	WORD; DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, INT, DINT, REAL, TIME, DATE, TOD, CHAR
INT	INT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
TIME	TIME	BYTE, WORD, DWORD, INT, DINT, TIME
S5TIME	S5TIME	WORD, S5TIME
DATE	DATE	BYTE, WORD, DWORD, INT, DINT, DATE
TOD	TOD	BYTE, WORD, DWORD, INT, DINT, TOD
CHAR	CHAR	BYTE, WORD, DWORD, CHAR
COUNTER	INT, WORD, COUNTER	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	INT, WORD, TIMER	WORD, DWORD, INT, UINT, DINT, UDINT

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen überschrieben.

Zum Kopieren von Feldern und Strukturen können die Anweisungen "Bereich kopieren" (BLKMOV) und "Bereich ununterbrechbar kopieren" (UBLKMOV) verwendet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, TIMER, COUNTER	E, A, M, D, L oder Konstante	Quellwert
OUT1	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, CHAR, TIMER, COUNTER	E, A, M, D, L	Zieladresse

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

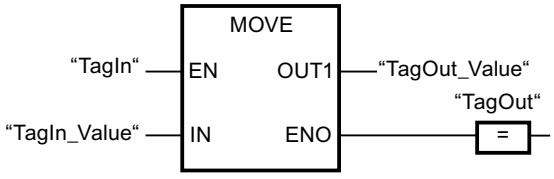

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn_Value" zum Operanden "TagOut_Value" und setzt den Ausgang "TagOut" auf den Signalzustand "1".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

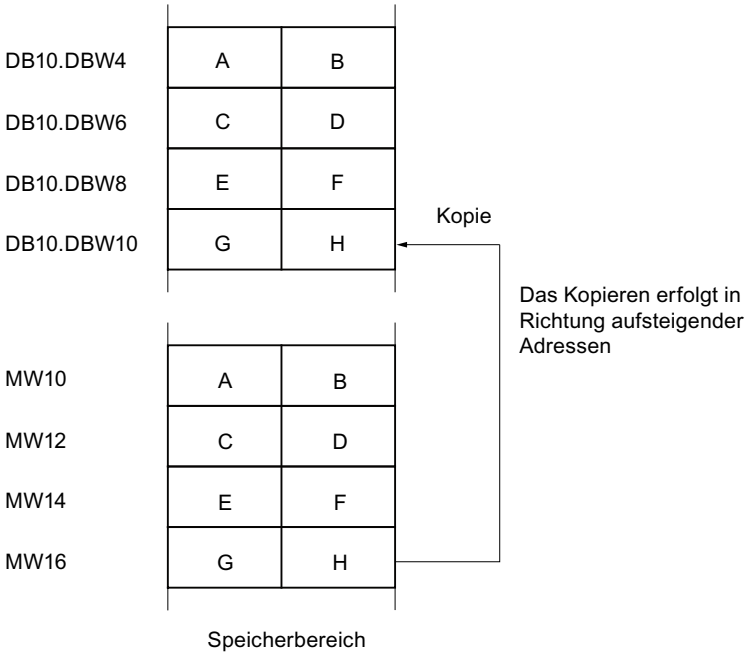

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Solange der Quellbereich nicht Teil eines Datenbausteins ist, der nur im Ladespeicher vorhanden ist, gibt es keine Begrenzung der Schachtelungstiefe.

Bei Unterbrechung einer BLKMOV-Bearbeitung hingegen, bei der aus einem nicht ablaufrelevanten DB kopiert wird, kann eine solche BLKMOV-Bearbeitung nicht mehr eingeschachtelt werden.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Nicht ablaufrelevante Datenbausteine

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Regeln beim Kopieren von nicht ablaufrelevanten Datenbausteinen

Der Quellbereich kann auch in einem nicht ablaufrelevanten Datenbaustein im Ladespeicher liegen. Datenbausteine, die nicht ablaufrelevant sind, werden mit dem Schlüsselwort UNLINKED gekennzeichnet.

Wenn ein nicht ablaufrelevanter Datenbaustein mit der Anweisung "Bereich kopieren" in den Arbeitsspeicher kopiert und gleichzeitig z. B. durch das Programmiergerät geladen wird, kann die Bearbeitung der Anweisung um mehrere Millisekunden verzögert werden. Dies führt zu einer Verlängerung des OB-Zyklus und kann zum Ansprechen der Zykluszeitüberwachung führen.

Wenn mit der Anweisung "Bereich kopieren" ein nicht ablaufrelevanter Datenbaustein kopiert wird und der Kopiervorgang unterbrochen wird, kann die Bearbeitung der Anweisung nicht mehr fortgesetzt werden.

Falls auf Ihrer CPU die Anweisung "Aus Datenbaustein im Ladespeicher lesen" vorhanden ist, müssen Sie diese Anweisung zum Lesen von nicht ablaufrelevanten Datenbausteinen im Ladespeicher benutzen. Wenn Sie die Anweisung "Bereich kopieren" benutzen, wird der Fehler W#16#8092 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

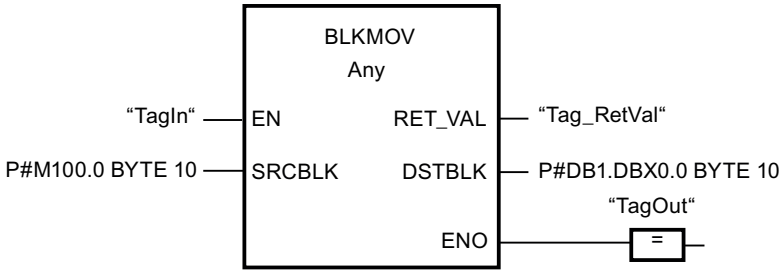

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 512 Bytes kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

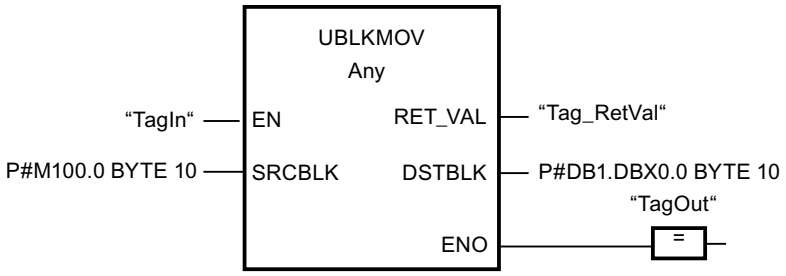

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

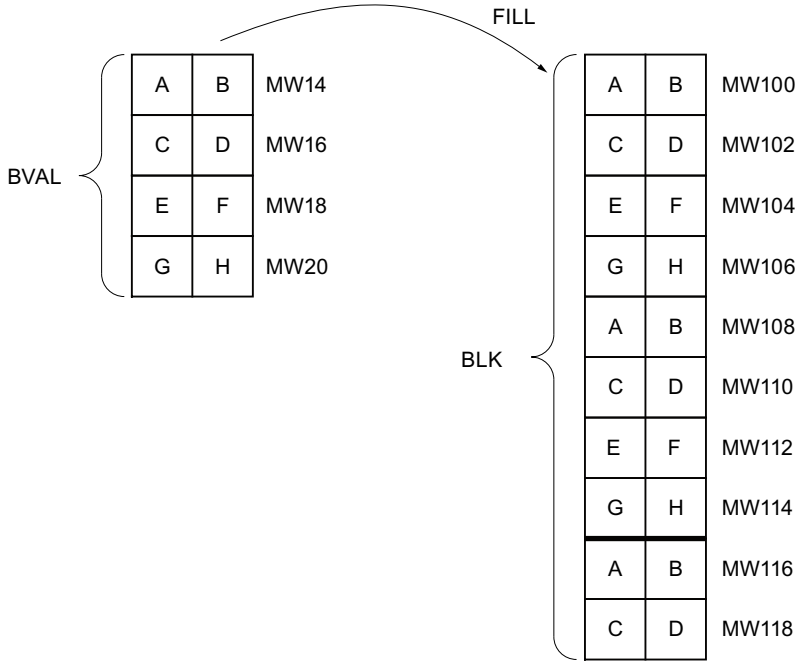

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
BVAL	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

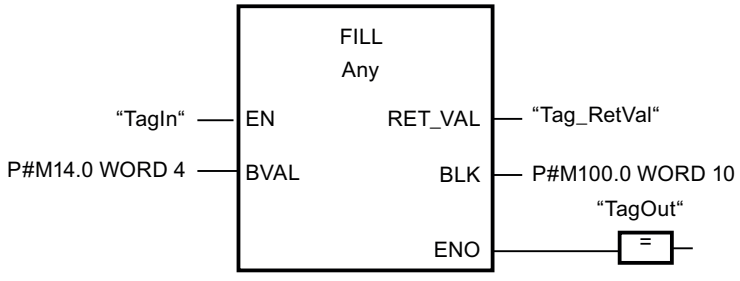

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend den in der Anweisungsbox ausgewählten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Weitere Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
OUT	Output	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

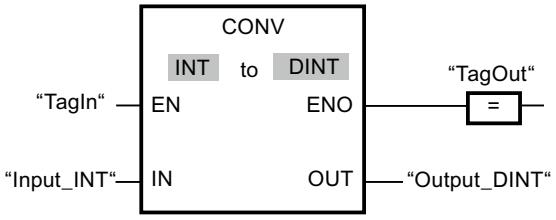

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird der Inhalt des Operanden "TagIn_Value" gelesen und in eine Ganzzahl (16 Bit) umgewandelt. Das Ergebnis wird im Operanden "TagOut_Value" abgelegt. Der Ausgang "TagOut" wird auf "1" gesetzt, wenn die Anweisung fehlerfrei ausgeführt wurde.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Explizite Konvertierungen (Seite 592)

Grundlagen zu FUP (Seite 7493)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl vom Datentyp DINT um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird.
OUT	Output	DINT	E, A, M, D, L, P	Ergebnis der Rundung

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

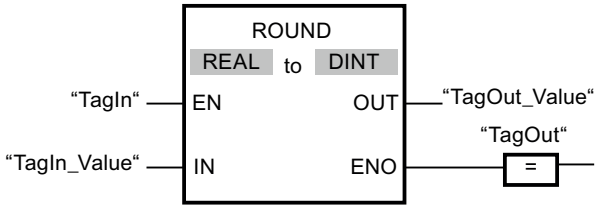

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächsten geraden Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

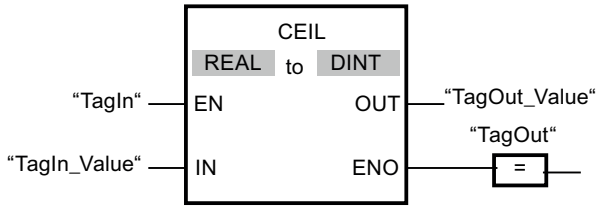

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

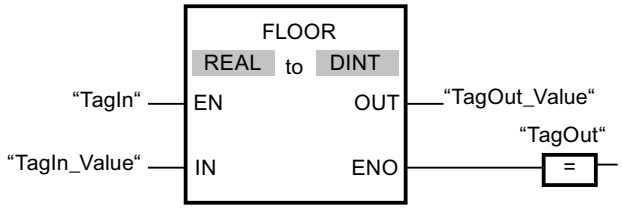

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Gleitpunktzahl am Eingang "TagIn_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu FUP (Seite 7493)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L oder Konstante	Eingangwert
OUT	Output	Ganzzahlen	E, A, M, D, L	Ganzzahliger Anteil des Eingangswerts

Aus den Klapplisten "???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

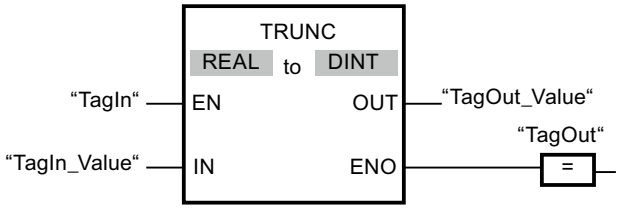

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

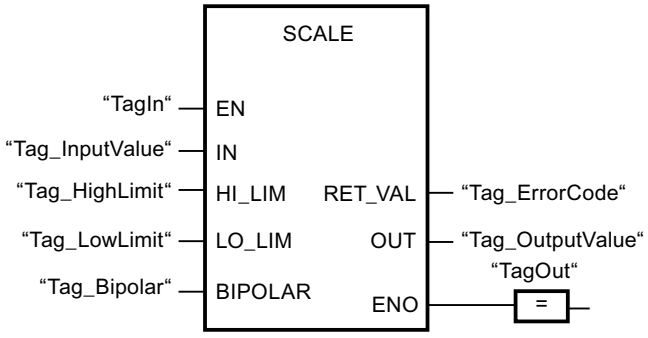

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandelt sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Input	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

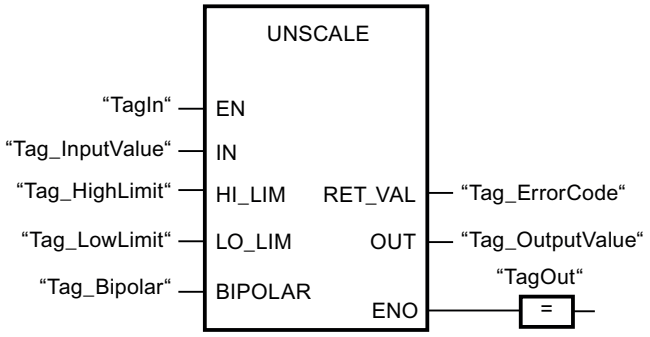

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

Programmsteuerung

JMP: Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

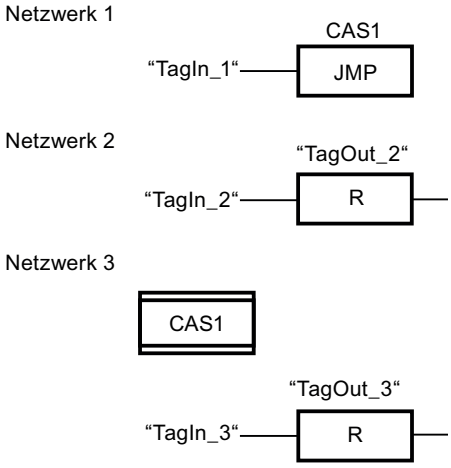

Wenn der Operand "TagIn_1"den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

JMPN: Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das VKE am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

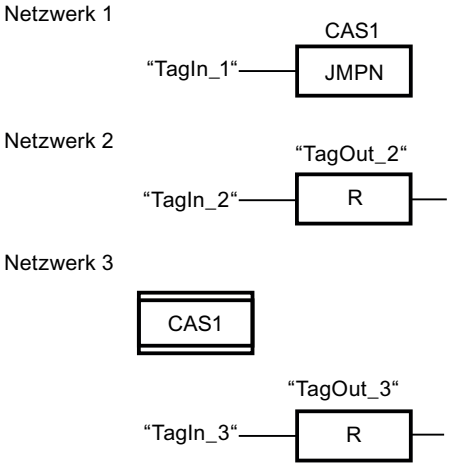

Wenn der Operand "TagIn_1" den Signalzustand "0" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

LABEL: Sprungmarke

Beschreibung

Die Sprungmarke kennzeichnet ein Zielnetzwerk, in dem nach einer ausgeführten Sprunganweisung die Programmbearbeitung fortgesetzt werden soll.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

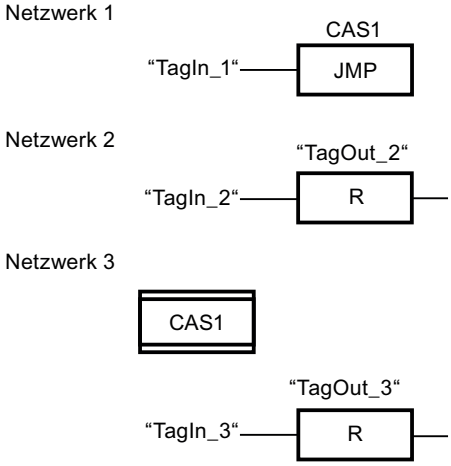

Wenn der Operand "TagIn_1" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut_3" zurückgesetzt.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

RET: Zurück springen

Beschreibung

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Programmbausteins.

- Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im aktuell aufgerufenen Programmbaustein beendet und im aufrufenden Programmbaustein (z. B. im aufrufenden Organisationsbaustein) nach der Aufruffunktion fortgesetzt. Der Signalzustand des Freigabeausgangs ENO der Aufruffunktion wird durch das VKE bestimmt. Das VKE wird über der Anweisung als fester Parameterwert dargestellt.
- Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt. Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Programmbausteins fortgesetzt.

Wenn die Anweisung im letzten Netzwerk des aufgerufenen Programmbausteins programmiert ist, wird der Freigabeausgang ENO der Aufruffunktion auf den aktuellen Signalzustand des VKE gesetzt.

Wenn mit der Anweisung ein Organisationsbaustein beendet wird, arbeitet die CPU im Systemprogramm weiter.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

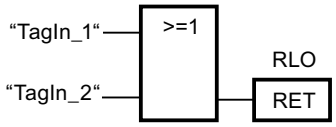

Wenn einer der Operanden "TagIn_1" oder "TagIn_2" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Programmbearbeitung im aufgerufenen Programmbaustein wird beendet und im aufrufenden Programmbaustein fortgesetzt. Der Freigabeausgang ENO der Aufruffunktion wird auf den Signalzustand "1" zurückgesetzt.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

OPN: Globalen Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Globalen Datenbaustein öffnen" schlagen Sie einen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Die darauf folgenden DB-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Globalen Datenbaustein öffnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Datenbau-stein>	Input	DB_ANY	-	Datenbaustein, der aufge-schlagen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

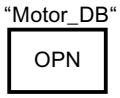

Netzwerk 2

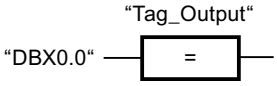

Im Netzwerk 1 wird der Datenbaustein "Motor_DB" aufgerufen. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Im Netzwerk 2 wird der Operand "DBX0.0" abgefragt. Der Signalzustand des Operanden "DBX0.0" wird dem Operanden "Tag_Output" zugewiesen.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

OPNI: Instanz-Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Instanz-Datenbaustein öffnen" schlagen Sie einen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DI-Register übertragen. Die darauf folgenden DI-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Instanz-Datenbaustein öffnen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Datenbau-stein>	Input	DB_ANY	-	Datenbaustein, der aufge-schlagen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Netzwerk 1

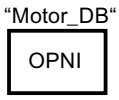

Netzwerk 2

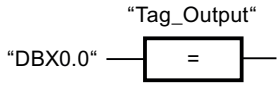

Im Netzwerk 1 wird der Datenbaustein "Motor_DB" aufgerufen. Die Nummer des Datenbausteins wird in DI-Register übertragen. Im Netzwerk 2 wird der Operand "DBX0.0" abgefragt. Der Signalzustand des Operanden "DBX0.0" wird dem Operanden "Tag_Output" zugewiesen.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Laufzeitsteuerung

COMPRESS: CPU-Speicher komprimieren

Beschreibung

Mit der Anweisung "CPU-Speicher komprimieren" komprimieren Sie den Arbeitsspeicher der CPU bzw. den RAM-Ladespeicher. Dabei werden Lücken, die durch Löschen oder Laden von Bausteinen entstanden sind, aufgefüllt, indem die vorhandenen Bausteine zusammengeschoben werden.

Mit dem Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Der Komprimiervorgang wird über mehrere Programmzyklen verteilt. Die Überwachung des Komprimiervorgangs erfolgt über die Werte der einzelnen Parameter. Wenn am Parameter BUSY der Anweisung der Wert "1" ansteht, ist der Komprimiervorgang noch in Ausführung. Der Wert "1" am Parameter DONE meldet, dass der Komprimiervorgang beendet ist. Fehlerinformationen können am Parameter RET_VAL abgefragt werden.

Wenn bereits ein extern angestoßener Komprimiervorgang ausgeführt wird, führt der Aufruf der Anweisung "CPU-Speicher komprimieren" zur Fehleranzeige.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CPU-Speicher komprimieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BUSY	Output	BOOL	E, A, M, D, L	Information, ob Komprimiervorgang aktiv ist.
DONE	Output	BOOL	E, A, M, D, L	Information, ob der Komprimiervorgang beendet wurde.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler Der Status des Komprimiervorgangs kann an den Parametern BUSY und DONE abgefragt werden.
8091	Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt.
8092	Die Anweisung kann nicht ausgeführt werden, da eine der folgenden Bedingungen zutrifft: • Die Funktion "Baustein löschen" ist aktiv. • Eine Test- und Inbetriebsetzungsfunktion wirkt auf einen Baustein ein (z. B. Status). • Die Funktion "Kopieren von Bausteinen" ist aufgrund eines externen Anstoßes aktiv. • Die H-CPU befindet sich in einem Ankoppel- oder Aktualisierungsvorgang.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufgerufen wird, wird die Anweisung nicht ausgeführt und der Freigabeausgang ENO auf den Signalzustand "0" gesetzt.

Die Anweisung "Zyklusüberwachungszeit neu starten" läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOPP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang vom RUN in STOPP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOPP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Gültig für S7-300-CPUs:

Bei den S7-300-CPUs (mit Ausnahme von S7-318) ist die mit der Anweisung programmierte Verzögerungszeit eine Mindestzeit. Sie verlängert sich um die Ausführungszeit der eingeschachtelten Prioritätsklassen sowie um Systemlasten.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

PROTECT: Schutzstufe ändern

Beschreibung

Mit der Anweisung "Schutzstufe ändern" ändern Sie die projektierte Schutzstufe "1" der CPU ab.

Die parametrierte Schutzstufe "1" mit aktivierter Option "Durch Passwort aufhebbar" bewirkt, dass man ein Passwort parametrieren muss. Mit diesem Passwort können Sie auf die mit der Anweisung "Schutzstufe ändern" schreibgeschützte CPU im Betrieb schreibend zugreifen (z. B. Laden).

Die Schutzstufe legen Sie am Parameter MODE der Anweisung fest. Die aktuelle Schutzstufe der CPU wird im Inspektorfenster unter "Eigenschaften > Schutz" angezeigt.

Die folgende Tabelle zeigt, welche Bedeutung die Schutzstufen haben:

Schutzstufe	Bedeutung
1	Alle PG-Funktionen sind erlaubt. Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
2	• Das Programm und die Konfiguration der CPU können nicht geändert werden. Das Programm in der CPU kann gelesen werden. • Die Funktionen für Prozessführung, Prozessbeobachtung und Prozesskommunikation sind erlaubt. • Alle Auskunftsfunktionen sind erlaubt. • Mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreibschutz aufheben. • Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
3	• Auch mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreib- und Leseschutz nicht aufheben. • Wenn zum Zeitpunkt des Aufrufs der Anweisung "Schutzstufe ändern" eine legitimierte Verbindung vorhanden ist, bleibt der Aufruf der Anweisung für diese Verbindung ohne Wirkung.

Wenn Sie für Ihre CPU die Schutzstufe "2" oder "3" projektiert haben, hat der Aufruf der Anweisung "Schutzstufe ändern" keine Wirkung.

Die Wirksamkeit der Schutzstufe "2" bzw. "3" nach Ausführung der Anweisung "Schutzstufe ändern" können Sie der SZL-Teilliste mit der SZL-ID W#16#0232 und dem Index W#16#0004 entnehmen.

Hinweis

Mit der Anweisung "Schutzstufe ändern" können Sie keine niedrigere Schutzstufe einstellen als die, die Sie mit STEP 7 "Hardware konfigurieren" projektiert haben. Sie haben z. B. die Schutzstufe 3 projektiert und anschließend die Anweisung "Schutzstufe ändern" mit dem Parameter MODE = 12 aufgerufen. Dies bewirkt eine Sperre der Passwort-Legitimierung. Ein erneuter Aufruf der Anweisung mit dem Parameter MODE = 0 oder 1 bewirkt, dass die Schutzstufe 3 beibehalten wird und dass der Schreib- und Leseschutz mit Eingabe des gültigen Passworts aufgehoben wird.

Auswirkungen der Betriebszustandsübergänge auf die Schutzstufe

Die folgende Tabelle zeigt, welche Auswirkungen die Betriebszustandsübergänge auf die mit der Anweisung "Schutzstufe ändern" eingestellte Schutzstufe haben.

Aktion	Auswirkungen auf die Schutzstufe
• Betriebsartenschalter auf "STOP" • Ungepuffertes NETZ-EIN • Urlöschen durch Betriebsartenschalter • Konfigurations- und/oder Programmänderungen im Betriebszustand "STOP" der S7-400-CPU	Die Schutzstufe wird auf "1" gesetzt. Alle PG-Funktionen sind erlaubt.
• Kaltstart • Neustart (Warmstart)	Die Schutzstufe wird durch das Betriebssystem auf "1" gesetzt. Alle PG-Funktionen sind erlaubt. Bei Bedarf können Sie durch Aufruf der Anweisung "Schutzstufe ändern" in Ihrem Programm die Schutzstufe auf "2" oder "3" setzen.
Gepuffertes NETZ-EIN	Bleibt unverändert
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch den Aufruf der Anweisung "Programm beenden" (STP), durch Programmfehler ohne zugehörigen Fehler-OB oder durch Anwenderaktion).	Bleibt unverändert
S7-400: Wiederanlauf durch Anwenderaktion oder automatisch nach NETZ-EIN	Bleibt unverändert

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schutzstufe ändern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	WORD	E, A, M, D, L oder Konstante	Schutzstufe Mögliche Werte: • W#16#0000: Einstellen der Schutzstufe "1" • W#16#0001: Einstellen der Schutzstufe "2" mit Passwort-Legitimierung • W#16#000C: Einstellen der Schutzstufe "3" ohne Passwort-Legitimierung Am Parameter MODE können auch Konstanten angegeben werden.
RET_VAL	Output	INT	E, A, M, D, L	Tritt während der Bearbeitung ein Fehler auf, wird eine Fehlerinformation ausgegeben.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
8090	Unzulässiger Wert am Parameter MODE.
80C3	Die benötigten Betriebsmittel sind momentan belegt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

CiR: CiR-Vorgang steuern

Beschreibung

Mit der Anweisung "CiR-Vorgang steuern" beeinflussen Sie im Anwenderprogramm den CiR-Vorgang wie folgt:

- Den CiR-Vorgang sperren. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU stets abgelehnt. Die Sperre gilt so lange, bis Sie diese mit der Anweisung "CiR-Vorgang steuern" wieder aufheben.
- Den CiR-Vorgang bedingt sperren, indem Sie am Parameter FRZ_TIME eine Obergrenze für die CiR-Synchronisationszeit festlegen. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU nur dann zugelassen, wenn die in der CPU errechnete CiR-Synchronisationszeit die angegebene Obergrenze nicht überschreitet.
- Den CiR-Vorgang freigeben. Die Obergrenze der CiR-Synchronisationszeit wird auf den Defaultwert 1 000 ms gesetzt.
- Ermittelt, ob der CiR-Vorgang freigegeben ist oder nicht. Falls der CiR-Vorgang freigegeben oder bedingt freigegeben ist, erhalten Sie im Parameter A_FT der Anweisung die aktuelle Obergrenze für die CiR-Synchronisationszeit.

Hinweis

Während der CiR-Synchronisationszeit sind die Ausgänge eingefroren und die Eingänge werden nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CiR-Vorgang steuern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte am Parameter MODE sind: • 0: Auskunftsfunktion • 1: CiR-Vorgang freigeben • 2: CiR-Vorgang sperren • 3: CiR-Vorgang bedingt sperren. Die Obergrenze der CiR-Synchronisationszeit legen Sie in FRZ_TIME fest.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FRZ_TIME	Input	TIME	E, A, M, D, L oder Konstante	Obergrenze der CiR-Synchronisationszeit in Millisekunden. Zulässiger Wertebereich: 60 ... 2500 ms (Voreinstellung: 1000 ms) Der Parameter FRZ_TIME ist nur dann relevant, wenn der Parameter MODE auf den Wert "3" gesetzt ist.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation Wenn der Parameter MODE den Wert "0" hat, liefert der Parameter RET_VAL Informationen über die Freigabe des CiR-Vorgangs.
A_FT	Output	TIME	E, A, M, D, L	Aktuell gültige Obergrenze der CiR-Synchronisationszeit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "1", "2" oder "3" gesetzt ist.
0001	Der CiR-Vorgang ist freigegeben. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0002	Der CiR-Vorgang ist gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0003	Der CiR-Vorgang ist bedingt gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
8001	Die CPU ist nicht bereit für einen CiR-Vorgang. Sie verwenden eine H-CPU im H-System (Solobetrieb) oder Sie arbeiten mit einer Standard-CPU im Multicomputing-Betrieb.
8002	Der Wert am Parameter MODE ist nicht zulässig.
8003	Der Wert am Parameter FRZ_TIME ist nicht zulässig.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Mit der Anweisung "CiR-Vorgang steuern" können Sie beispielsweise programmieren, dass kein CiR-Vorgang in Zeitabschnitten angestoßen wird, in denen für die Prozessbearbeitung die maximale Leistungsfähigkeit der CPU benötigt wird.

Dazu muss die Anweisung "CiR-Vorgang steuern" vor Beginn des Zeitabschnitts mit erhöhter Prozessaktivität im Programm aufgerufen und mit dem Wert "2" (CiR-Vorgang sperren) am Parameter MODE ausgeführt werden. Dadurch wird den CiR-Vorgang gesperrt.

Um den CiR-Vorgang nach der Beendigung des Zeitabschnitts mit erhöhter Prozessaktivität wieder freizugeben, muss die Anweisung "CiR-Vorgang steuern" erneut aufrufen und mit dem Wert "1" am Parameter MODE ausführt werden. Sie können alternativ den Parameter MODE auch auf den Wert "3" setzen und dadurch den CiR-Vorgang bedingt sperren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

Wortverknüpfungen

AND: UND verknüpfen

Beschreibung

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

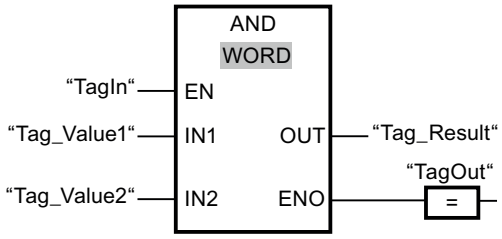

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	00000000 00000101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

OR: ODER verknüpfen

Beschreibung

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüp-fung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

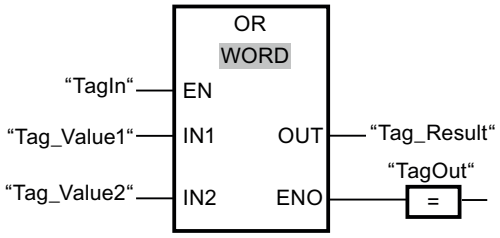

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	01010101 01011111

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

XOR: EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN1	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Erster Wert der Verknüpfung
IN2	Input	WORD, DWORD	E, A, M, D, L, P oder Konstante	Zweiter Wert der Verknüpfung
OUT	Output	WORD, DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

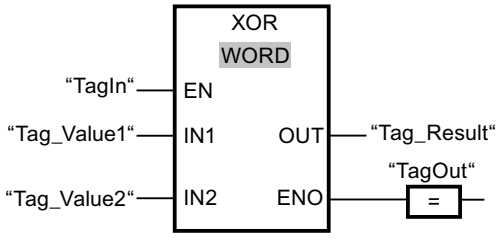

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	01010101 01011010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung gausgeführt. Der Wert des Operanden "Tag_Value1" wird mit dem Wert des Operanden "Tag_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

INVERT: Einerkomplement erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	Ganzzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	Ganzzahlen	E, A, M, D, L, P	Einerkomplement des Wertes am Eingang IN

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

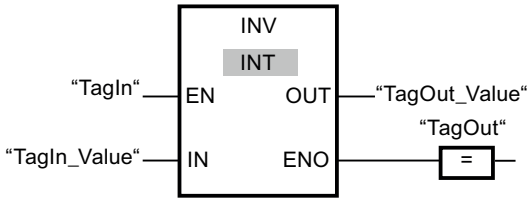

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	W#16#000F
OUT	TagOut_Value	W#16#FFF0

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn_Value" um und schreibt das Ergebnis in den Ausgang "TagOut_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	DWORD	E, A, M, D, L, P	Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

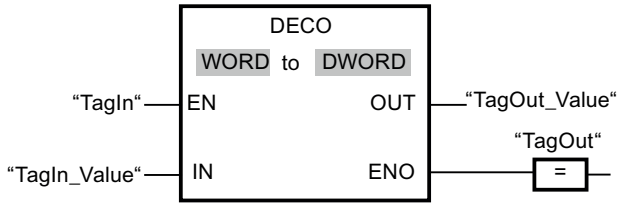

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

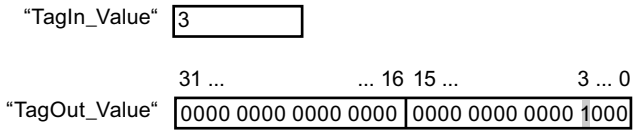

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn_Value" am Eingang und setzt das dritte Bit im Wert des Operanden "TagOut_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben ihn am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

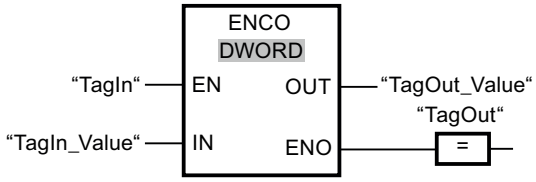

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

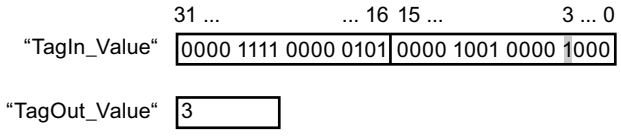

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung ausgeführt. Die Anweisung selektiert das niederwertigste gesetzte Bit am Eingang "TagIn_Value" und schreibt die Bitposition "3" in die Variable am Ausgang "TagOut_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt den Signalzustand "1" am Freigabeeingang EN voraus und dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
G	Input	BOOL	E, A, M, D, L, T, Z	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

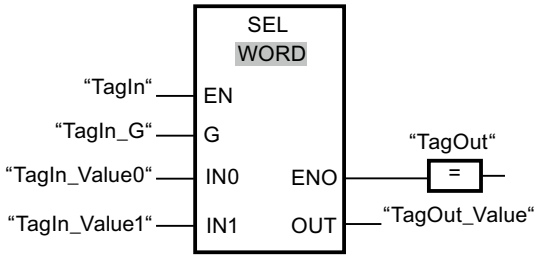

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn_G" wird der Wert am Eingang "TagIn_Value0" oder "TagIn_Value1" selektiert und in den Ausgang "TagOut_Value" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp INT um vier Bitstellen nach rechts verschoben wird:

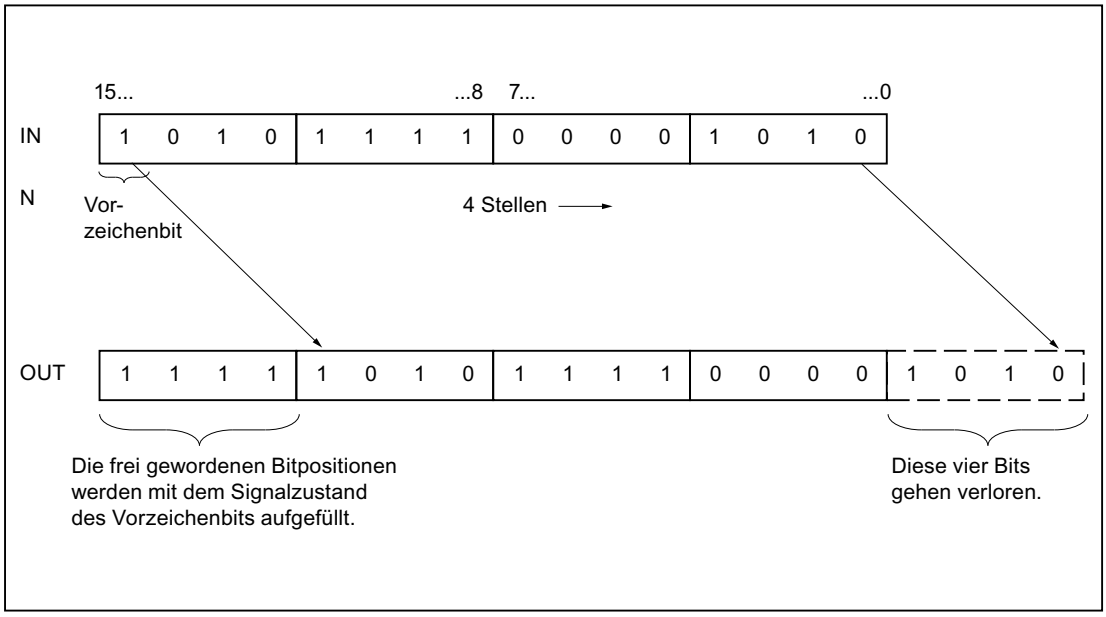

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

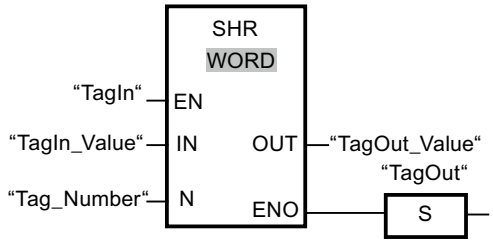

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

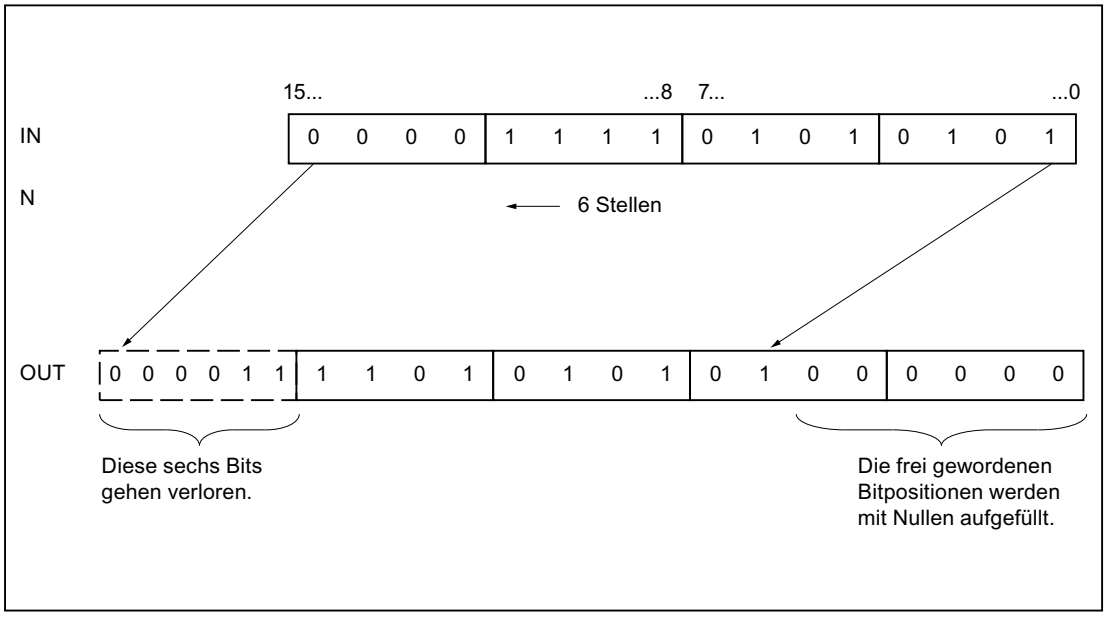

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
OUT	Output	BYTE, WORD, DWORD, INT, DINT	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

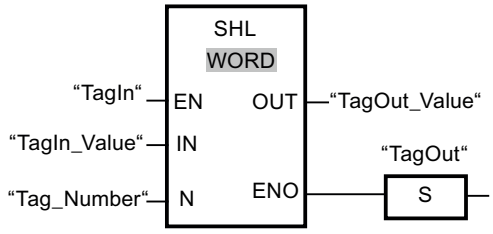

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

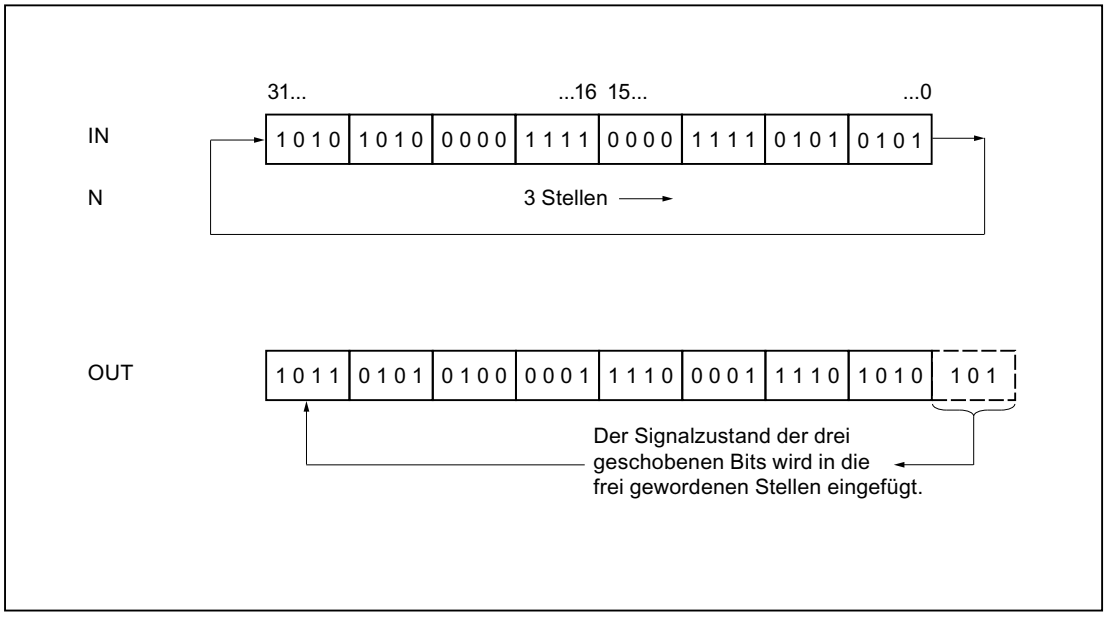

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
OUT	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

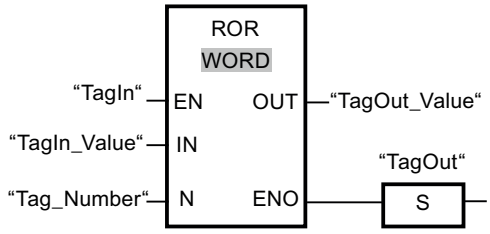

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1101 0101 0000 0111 1000 0111 1010

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

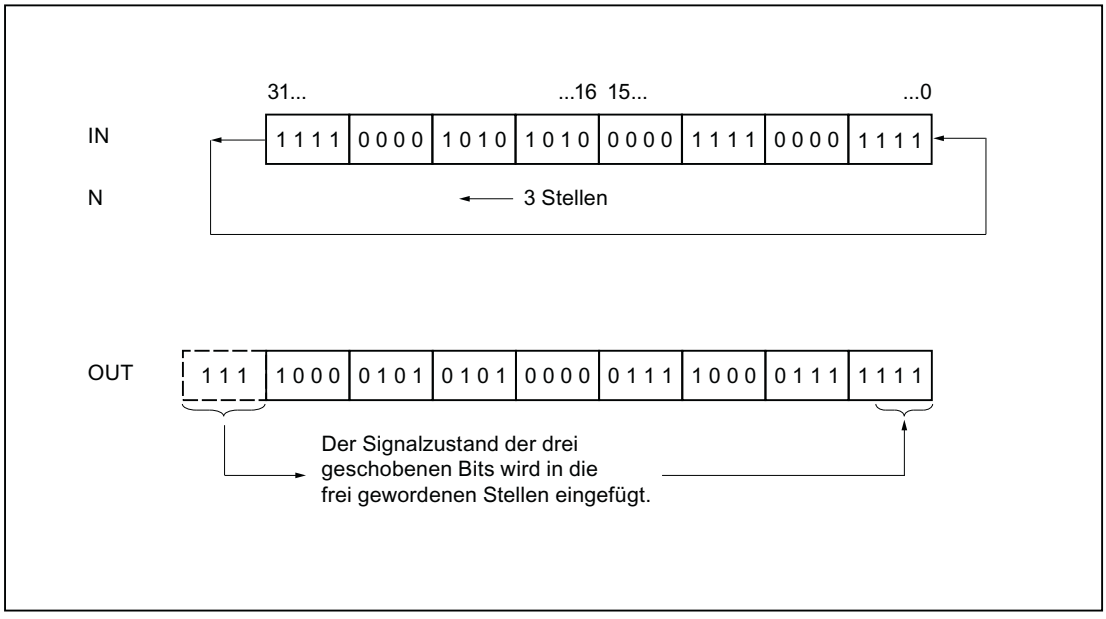

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, T, Z, D, L	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird.
N	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird.
OUT	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

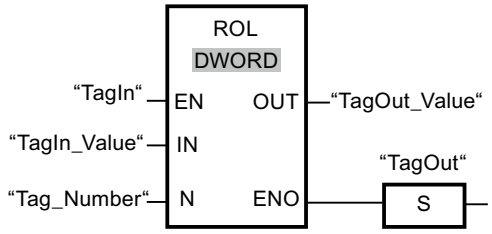

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	TagIn_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
OUT	TagOut_Value	0001 0101 0100 0001 1110 0001 1111 1110

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Inhalt des Operanden "TagIn_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

Datentyp eines FUP-Elements auswählen (Seite 7511)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Weitere Anweisungen

---| |---: Statusbit abfragen

Beschreibung

Mit der Anweisung "Statusbit abfragen" fragen Sie den Signalzustand der folgenden Statusbits ab:

- BIE: Binärergebnis
- OV: Überlauf
- OS: Speichernder Überlauf
- A0 und A1: Anzeigebits

Welche Statusbits abgefragt werden, legen Sie durch die Auswahl des entsprechenden Symbols aus der Klappliste der Anweisung fest. Die folgende Tabelle zeigt die Möglichkeiten zur Abfrage der Statusbits:

Symbol	Abfrage	Erläuterung
==0	A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion gleich 0 ist.
<>0	A0 = 0 und A1 = 1 oder A0 = 0 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungleich 0 ist.
>=0	A0 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer oder gleich 0 ist.
<=0	A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner oder gleich 0 ist.
>0	A0 = 0 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion größer 0 ist.
<0	A0 = 1 und A1 = 0	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion kleiner 0 ist.
UO	A0 = 1 und A1 = 1	Die Anweisung fragt ab, ob das Ergebnis einer mathematischen Funktion ungültig ist. Ursache für ein ungültiges Ergebnis kann eine ungültige Gleitpunktzahl oder eine Division durch Null sein.
OV	OV = 1	Die Anweisung fragt ab, ob in einer Anweisung ein Zahlenbereichsüberlauf aufgetreten ist.

Symbol	Abfrage	Erläuterung
OS	OS = 1	Die Anweisung fragt ab, ob in einer Anweisung ein speichernder Zahlenbereichsüberlauf aufgetreten ist.
BIE	BIE = 1	Die Anweisung fragt den Signalzustand des Statusbits BIE.

Wenn die Abfrage erfüllt ist, liefert die Anweisung den Signalzustand "1". Bei nicht erfüllter Abfrage liefert die Anweisung den Signalzustand "0".

Eine Abfrage der Statusbits auf "0" können Sie mithilfe der Anweisung "VKE invertieren" programmieren.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

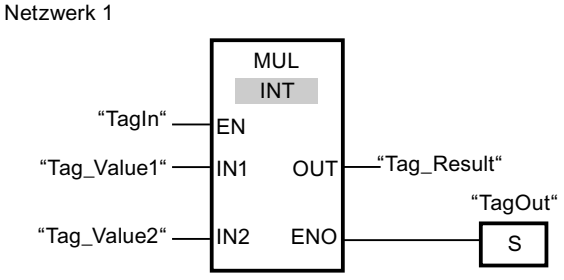

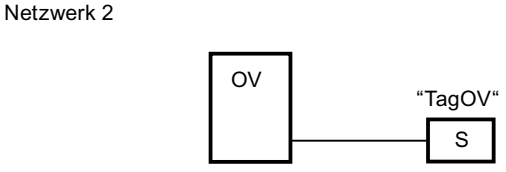

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" (MUL) ausgeführt. Bei einem Überlauf während der Ausführung der Anweisung "Multiplizieren" wird das Statusbit OV gesetzt. Nach der Abfrage des Statusbits OV wird im Netzwerk 2 die Anweisung "Ausgang setzen" (S) ausgeführt und der Operand "TagOV" gesetzt.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Setzen der Statusbits (Seite 185)

CALL: Baustein aufrufen

Beschreibung

Mit der Anweisung "Baustein aufrufen" rufen Sie eine Funktion (FC) ohne Parameter auf. Abhängig von der vorangehenden Verknüpfung handelt es sich um einen absoluten oder bedingten Aufruf.

Im Anweisungsteil einer Funktion (FC) können Sie bei einem bedingten Aufruf als Operand keinen Parameter vom Datentyp BLOCK_FC angeben. In einem Funktionsbaustein (FB) jedoch können Sie als Operanden einen Parameter vom Typ BLOCK_FC angeben.

Ein bedingter Aufruf wird nur dann ausgeführt, wenn das Verknüpfungsergebnis (VKE) "1" ist. Wird ein bedingter Aufruf nicht ausgeführt, ist das VKE nach der Aufrufoperation "0". Wird die Operation ausgeführt, arbeitet sie folgendermaßen:

- Sie speichert die Rücksprungadresse des aufrufenden Bausteins.
- Sie speichert die beiden Datenbaustein-Register (Datenbaustein und Instanz-Datenbaustein).
- Sie aktualisiert den Lokaldatenbereich für die aufgerufene Funktion.
- Sie schiebt das MA-Bit (aktives MCR-Bit) in den Baustein-Stack (B-Stack).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Baustein aufrufen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Baustein>	Output	BLOCK_FC	-	Aufgerufene Funktion

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

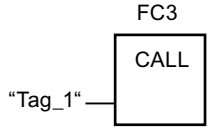

Die Funktion "FC3" wird aufgerufen, wenn der Operand "Tag_1" den Signalzustand "1" führt.

Siehe auch

Das Master Control Relay (Seite 201)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

SAVE: VKE im BIE-Bit speichern

Beschreibung

Mit der Anweisung "VKE im BIE-Bit speichern" sichern Sie das Verknüpfungsergebnis (VKE) im Binärergebnis (BIE). In der Ausführung überträgt die Anweisung den Signalzustand des aktuellen Verknüpfungsergebnisses zum Statusbit BIE. Die Anweisung arbeitet unabhängig von Bedingungen und beeinflusst keine weiteren Statusbits.

Hinweis

Das BIE-Bit kann nach der Anwendung der Anweisung "VKE im BIE-Bit speichern" durch nachfolgende Anweisungen im gleichen oder in unterlagerten Bausteinen wieder verändert werden

Sie können die Anweisung "VKE im BIE-Bit speichern" nutzen, um den Ausführungsstatus eines Bausteins zu kontrollieren. Wenn Sie z. B. die Anweisung "VKE im BIE-Bit speichern" am Ende eines Bausteins programmieren, wird das BIE-Bit auf den Signalzustand des aktuellen Verknüpfungsergebnisses des Bausteins gesetzt. Ist der Baustein fehlerfrei ausgeführt, wird das BIE-Bit sowie der Freigabeausgang ENO im aufrufenden Baustein auf den Signalzustand "1" gesetzt. Der Signalzustand "0" wird dem BIE-Bit zugewiesen, wenn während der Bearbeitung des Bausteins Fehler aufgetreten sind. Durch die Abfrage des BIE-Bits können Sie eine Fehlerbehandlung programmieren.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

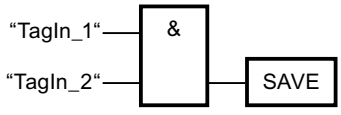

Das BIE-Bit wird auf den Signalzustand "1" gesetzt, wenn die Operanden "TagIn_1" und "TagIn_2" den Signalzustand "1" liefern.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Beispiel für den EN-/ENO-Mechanismus in FUP (Seite 159)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

MCR<: MCR-Bereiche öffnen

Beschreibung

Mit der Anweisung "MCR-Bereiche öffnen" speichern Sie das Verknüpfungsergebnis (VKE) im MCR-Stack und schalten die MCR-Abhängigkeit innerhalb eines MCR-Bereichs ein (MCR = Master Control Relay).

Der MCR-Stack kann maximal 8 Einträge enthalten und arbeitet wie ein LIFO-Zwischenspeicher (Last In, First Out). Ist der Stack voll, erzeugt die Anweisung eine Fehlermeldung (MCRF).

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Den Anfang und das Ende eines MCR-Bereichs legen Sie durch die Anweisungen "MCR-Bereich aktivieren" (MCRA) und "MCR-Bereich deaktivieren" (MCRD) fest. Innerhalb des definierten MCR-Bereichs können Sie für einen oder mehrere Programmteile die MCR-Abhängigkeit ein- und ausschalten.

Die MCR-Abhängigkeit wird eingeschaltet, wenn das VKE vor der Bearbeitung der Anweisung "1" ist. Wenn das VKE vor der Anweisung "0" ist, wird die MCR-Abhängigkeit ausgeschaltet. Das Ende des MCR-abhängigen Programmabschnitts wird durch die Anweisung "MCR-Bereiche schließen" programmiert. Der Programmabschnitt zwischen den beiden Anweisungen wird durch das MCR beeinflusst. Betroffen sind dabei nur die folgenden MCR-abhängigen Anweisungen:

- Zuweisung
- Ausgang setzen (S)
- Ausgang rücksetzen (R)
- Flipflop rücksetzen/setzen (RS)
- Flipflop setzen/rücksetzen (SR)
- Wert kopieren (MOVE)

Das Ein- und Ausschalten der MCR-Abhängigkeit können Sie auch schachteln, indem Sie in einem MCR-abhängigen Programmabschnitt die Anweisung "MCR-Bereiche öffnen" programmieren. Die Anweisung können Sie bis 8 Ebenen tief schachteln. Die MCR-Abhängigkeit des übergeordneten Programmteils steuert dabei die MCR-Abhängigkeiten in allen untergeordneten Programmteilen. Wenn die MCR-Abhängigkeit im übergeordneten Programmteil abgeschaltet ist, kann diese in einem untergeordneten MCR-abhängigen Programmteil nicht eingeschaltet werden.

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins, wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

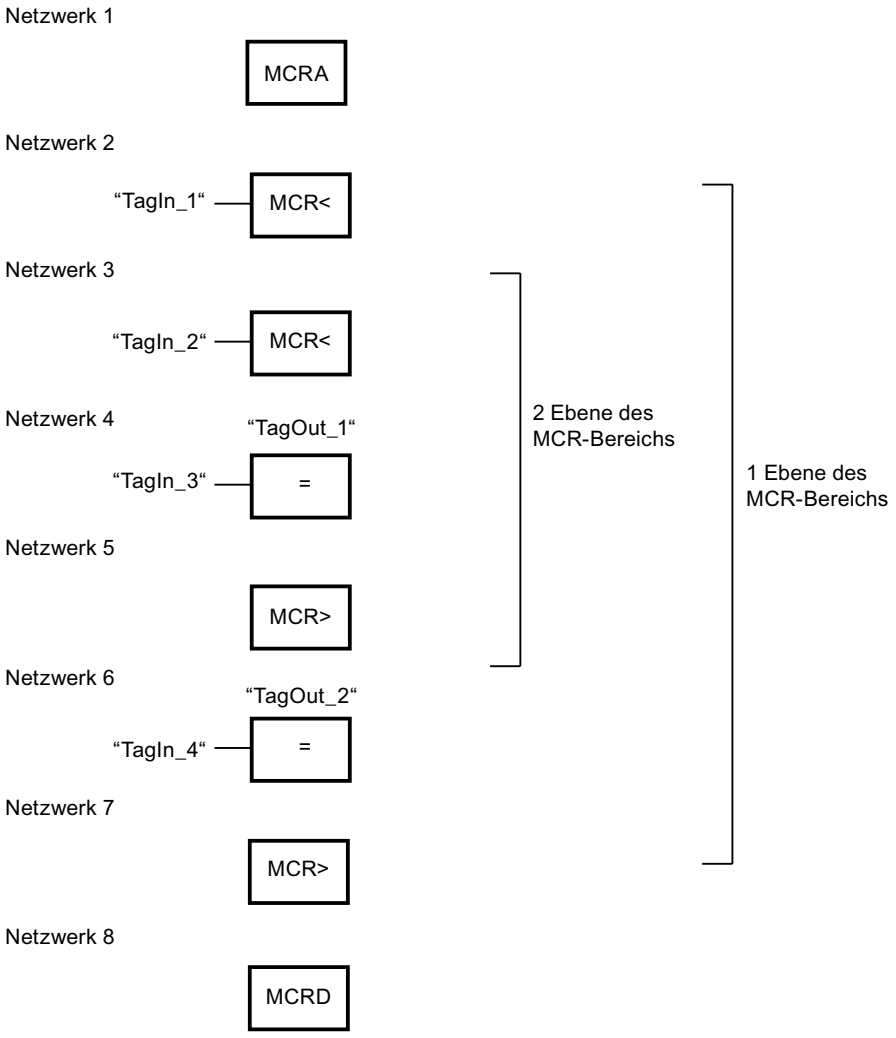

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit in der ersten Ebene des MCR-Bereichs eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_4" dem Operanden "TagOut_2" (Netzwerk 6) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_2" auf "0" zurückgesetzt. Im Netzwerk 7 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet.

Im Netzwerk 3 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs eingeschaltet, wenn der Operand "TagIn_2" den Signalzustand "1" liefert und die erste Ebene des MCR-Bereichs eingeschaltet ist. In diesem Fall wird der Signalzustand des Operanden "TagIn_3" dem Operanden "TagOut_1" zugewiesen. Wenn der Operand "TagIn_2" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_1" auf "0" zurückgesetzt. Im Netzwerk 5 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

MCR>: MCR-Bereiche schließen (Seite 4753)

MCRA: MCR-Bereich aktivieren (Seite 4755)

MCRD: MCR-Bereich deaktivieren (Seite 4757)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

MCR>: MCR-Bereiche schließen

Beschreibung

Mit der Anweisung "MCR-Bereiche schließen" schließen Sie einen MCR-abhängigen Programmabschnitt ab (MCR = Master Control Relay). Die Anweisung löscht dabei das Verknüpfungsergebnis (VKE) aus dem letzten Eintrag im MCR-Stack.

Der MCR-Stack kann maximal 8 Einträge enthalten und arbeitet wie ein LIFO-Zwischenspeicher (Last In, First Out). Ist der Stack bereits leer, erzeugt die Anweisung eine Fehlermeldung (MCRF).

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Die MCR-Abhängigkeit programmieren Sie mit der Anweisung "MCR-Bereiche öffnen". Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

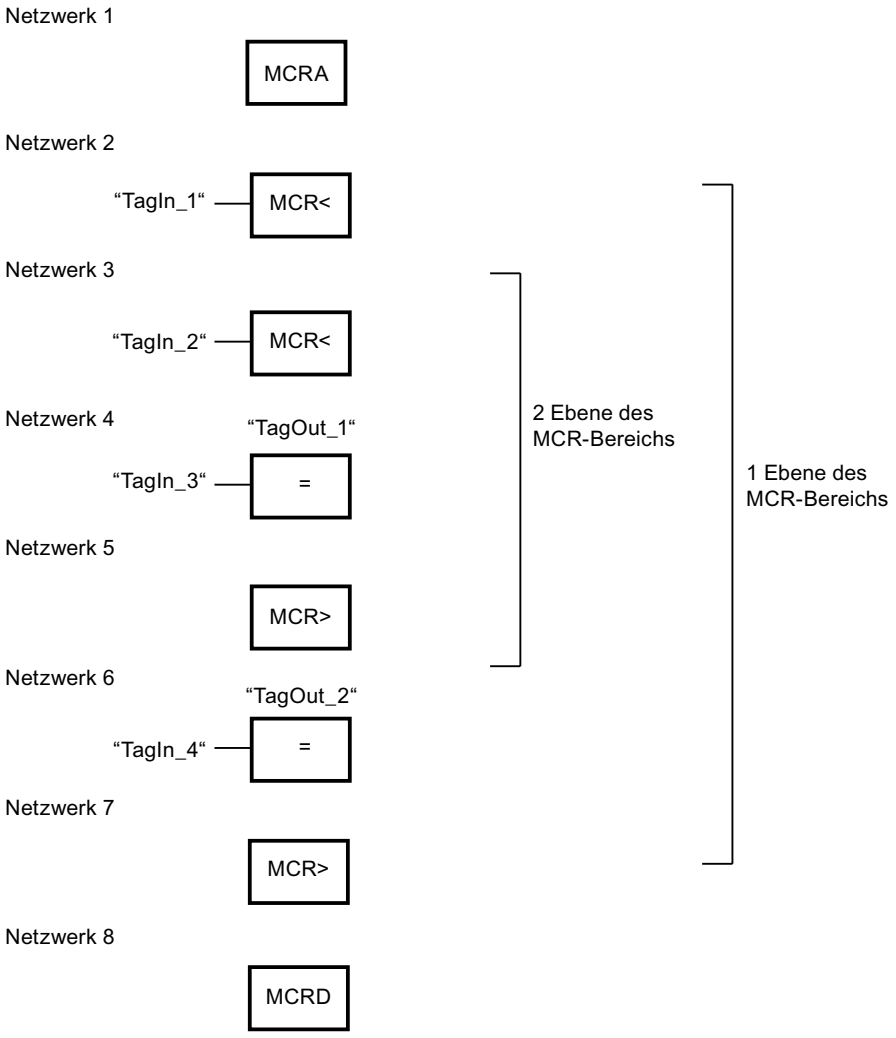

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit in der ersten Ebene des MCR-Bereichs eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_4" dem Operanden "TagOut_2" (Netzwerk 6) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_2" auf "0" zurückgesetzt. Im Netzwerk 7 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet.

Im Netzwerk 3 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs eingeschaltet, wenn der Operand "TagIn_2" den Signalzustand "1" liefert. In diesem Fall wird der Signalzustand des Operanden "TagIn_3" dem Operanden "TagOut_1" zugewiesen. Wenn der Operand "TagIn_2" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut_1" unabhängig vom Signalzustand des Operanden "TagIn_3" auf "0" zurückgesetzt. Im Netzwerk 5 wird die MCR-Abhängigkeit in der zweiten Ebene des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

MCR<: MCR-Bereiche öffnen (Seite 4751)

MCRA: MCR-Bereich aktivieren (Seite 4755)

MCRD: MCR-Bereich deaktivieren (Seite 4757)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

MCRA: MCR-Bereich aktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich aktivieren" legen Sie den Anfang eines MCR-Bereichs fest (MCR = Master Control Relay). Innerhalb dieses Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen ein- und ausschalten:

- MCR<: MCR-Bereiche öffnen
- MCR>: MCR-Bereiche schließen

Das Ende des MCR-Bereichs legen Sie mit der Anweisung "MCR-Bereich deaktivieren" (MCRD) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

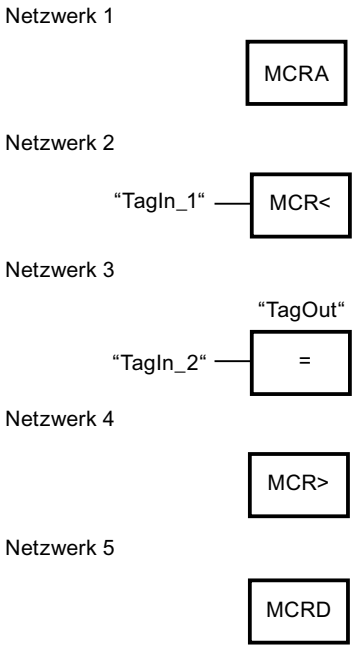

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_2" dem Operanden "TagOut" (Netzwerk 3) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut" unabhängig vom Signalzustand des Operanden "TagIn_2" auf "0" zurückgesetzt. Im Netzwerk 4 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

MCR<: MCR-Bereiche öffnen (Seite 4751)

MCR>: MCR-Bereiche schließen (Seite 4753)

MCRD: MCR-Bereich deaktivieren (Seite 4757)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

MCRD: MCR-Bereich deaktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich deaktivieren" legen Sie das Ende eines MCR-Bereichs fest (MCR = Master Control Relay).

Den Anfang eines MCR-Bereichs legen Sie mit dem Anweisung "MCR-Bereich aktivieren" (MCRA) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Innerhalb eines MCR-Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen programmieren:

- MCR<: MCR-Bereiche öffnen
- MCR>: MCR-Bereiche schließen

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

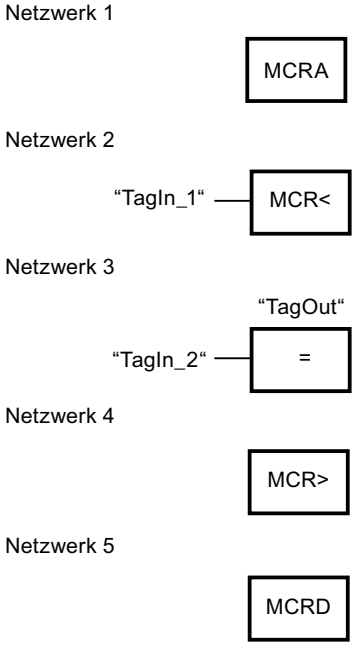

Im Netzwerk 1 wird der Anfang des MCR-Bereichs programmiert. Wenn der Operand "TagIn_1" den Signalzsutand"1" liefert, wird die MCR-Abhängigkeit eingeschaltet. In diesem Fall wird der Signalzustand des Operanden "TagIn_2" dem Operanden "TagOut" (Netzwerk 3) zugewiesen. Wenn der Operand "TagIn_1" den Signalzustand "0" führt, wird die MCR-Abhängigkeit ausgeschaltet und der Operand "TagOut" auf "0" zurückgesetzt. Im Netzwerk 4 wird die MCR-Abhängigkeit des MCR-Bereichs ausgeschaltet. Im Netzwerk 8 wird das Ende des MCR-Bereichs programmiert.

Siehe auch

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

MCR<: MCR-Bereiche öffnen (Seite 4751)

MCR>: MCR-Bereiche schließen (Seite 4753)

MCRA: MCR-Bereich aktivieren (Seite 4755)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

SET: Bereich bitweise setzen

Beschreibung

Mit der Anweisung "Bereich bitweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "1". Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu setzenden Bits im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Hinweis

Parameter N

Sie können am Parameter N auch ein Element eines Datenbausteins verschalten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits gesetzt werden Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der zu setzenden Bits

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

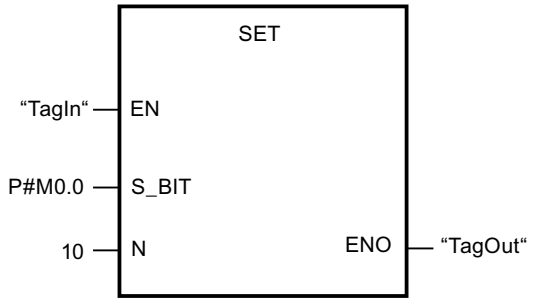

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SETP: Bitfeld im Peripheriebereich setzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich setzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "1". Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge gesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das gesetzt wird. Die Bits des Feldes müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zu setzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Setzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Für die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "1" gesetzt.

Die Anweisung "Bitfeld im Peripheriebereich setzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich setzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu setzenden Bits
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste zu setzende Bit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

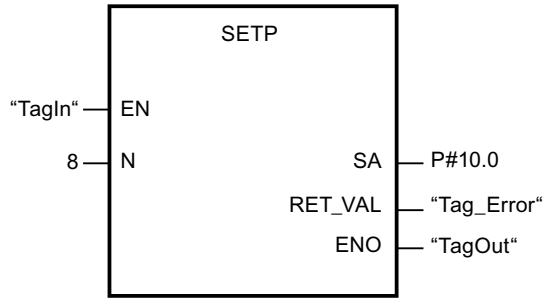

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

SETI: Bereich byteweise setzen

Beschreibung

Mit der Anweisung "Bereich byteweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "1". Den Anfangspunkt des Bereichs definieren Sie am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zu setzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich "P" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise setzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits gesetzt werden Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des zu setzenden Bereichs Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

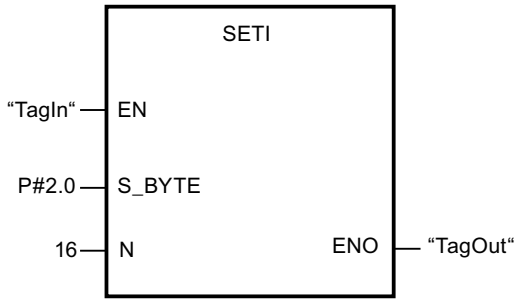

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

RESET: Bereich bitweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich bitweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu rücksetzenden Bits im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits zurückgesetzt werden Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der Bits, die zurückgesetzt werden

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

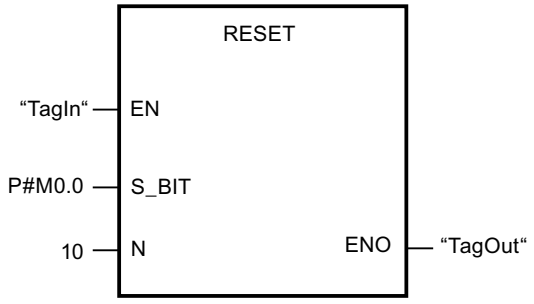

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

RESETP: Bitfeld im Peripheriebereich rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich rücksetzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "0" zurück. Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge zurückgesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das zurückgesetzt wird. Die Bits des Felds müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zurückzusetzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Rücksetzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "0" zurückgesetzt.

Die Anweisung "Bitfeld im Peripheriebereich rücksetzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich rücksetzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der Bits, die zurückgesetzt werden.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste Bit, das zurückgesetzt wird.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

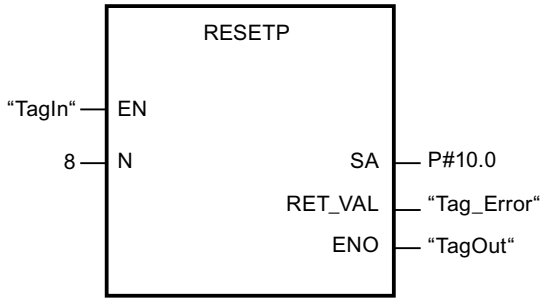

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

RESETI: Bereich byteweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich byteweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der rückzusetzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich "P" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise rücksetzen" hat keine Wirkung und der Freigabeausgang ENO wird auf "0" gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits zurückgesetzt werden Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des Bereichs, dessen Bits zurückgesetzt werden. Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

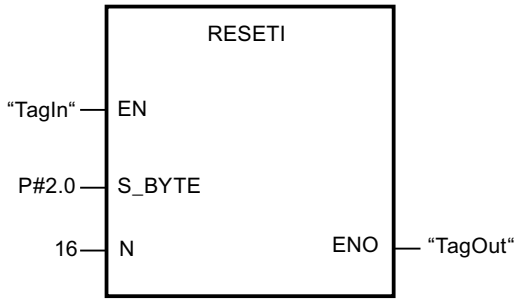

Wenn der Operand "TagIn" den Signalzustand "1" liefert, setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Das Master Control Relay (Seite 201)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

REPL_VAL: Ersatzwert eintragen

Beschreibung

Mit der Anweisung "Ersatzwert eintragen" tragen Sie von einem Synchronfehler-Organisationsbaustein aus einen Ersatzwert in den Akkumulator 1 (AKKU 1) ein.

Hinweis

Die Anweisung "Ersatzwert eintragen" dürfen Sie nur in Synchronfehler-Organisationsbausteinen (OB 121, OB 122) aufrufen.

Die Anweisung "Ersatzwert eintragen" verwenden Sie, wenn Sie von einer Baugruppe keinen Wert mehr lesen können. Nach jedem Zugriff auf diese Baugruppe wird der OB 122 gestartet. Mit dem Aufruf der Anweisung "Ersatzwert eintragen" können Sie bewirken, dass ein Ersatzwert in den Akkumulator 1 geladen wird. Die Programmbearbeitung wird dann mit diesem Ersatzwert fortgesetzt. Die für die Auswahl des Ersatzwerts notwendigen Informationen können Sie den lokalen Variablen des OB 122 entnehmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ersatzwert eintragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
VAL	Input	DWORD	E, A, M, D, L oder Konstante	Ersatzwert
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Die Anweisung wurde nicht von einem Synchronfehler-OB aus aufgerufen.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu FUP (Seite 7493)

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden

Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zuletzt programmierten Schritts
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Ergebnis

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

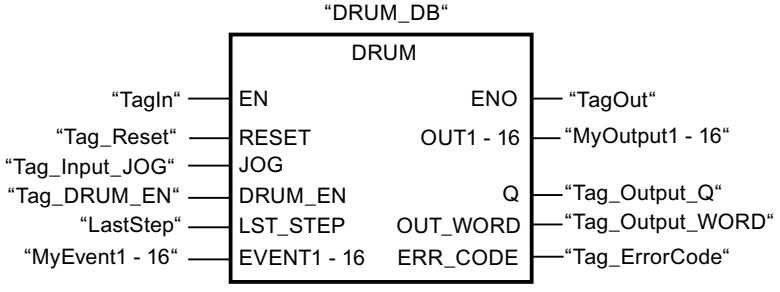

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

DRUM_X: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" realisieren Sie ein Schrittschaltwerk mit maximal 16 Schritten. Die Nummer des ersten Schritts legen Sie am Parameter DSP fest. Die Nummer des letzten Schritts geben Sie am Parameter LST_STEP vor.

In jedem Schritt werden alle 16 Ausgangsbits OUT0 bis OUT15 und der Ausgangsparameter OUT_WORD (in dem die Ausgangsbits zusammengefasst sind) beschrieben. Ein Ausgangsbit bekommt entweder das zugehörige Bit des von Ihnen vorgegebenen Felds OUT_VAL oder das Ausgangsbit des vorangegangenen Schritts zugewiesen. Welcher Wert zugewiesen wird, hängt von der Belegung der Maskenbits am Parameter S_MASK ab.

Hinweis

Die Voreinstellung der Maskenbits ist 0. Falls Sie die aktuelle Einstellung eines oder mehrerer Maskenbits ändern wollen, führen Sie diese Änderung im Instanz-Datenbaustein durch.

Die Anweisung "Schrittschaltwerk realisieren" schaltet in den nächsten Schritt, wenn am Parameter JOG eine positive Flanke aufgetreten ist. Falls sich die Anweisung bereits im letzten Schritt befindet, werden bei einer positiven Flanke am Parameter JOG die Variablen Q und EOD gesetzt. Der Parameter DCC wird in diesem Fall auf den Wert 0 gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" gesetzt wird.

Darüber hinaus können Sie auch ein zeitabhängiges Weiterschalten in den nächsten Schritt zulassen. Dazu muss der Parameter DRUM_EN auf den Wert "1" gesetzt werden. Der Übergang in den nächsten Schritt erfolgt, wenn das zum aktuellen Schritt zugehörige Ereignisbit (EVENTi) gesetzt ist und die für den aktuellen Schritt vorgegebene Zeit abgelaufen

ist. Die Zeit für einen Schritt ergibt sich aus dem Produkt von der Zeitbasis (DTBP) und dem für den aktuellen Schritt gültigen Zeitfaktor (S_PRESET).

Hinweis

Die im aktuellen Schritt noch verbleibende Bearbeitungszeit (DCC) wird nur dann reduziert, wenn das zugehörige Ereignisbit (EVENTi) gesetzt ist.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Wenn beim Aufruf der Anweisung der Wert "1" am Parameter RESET anliegt, geht das Schrittschaltwerk in den Schritt, dessen Nummer am Parameter DSP angegeben ist.

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist.

Zusätzlich können Sie das Schrittschaltwerk jederzeit über den Wert am Parameter JOG weiterschalten.

Beim Erstaufruf der Anweisung müssen Sie den Parameter RESET mit 1 belegen.

Wenn sich das Schaltwerk im letzten Schritt befindet und die für diesen Schritt vorgegebene Bearbeitungszeit abgelaufen ist, werden die Parameter Q und EOD gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" wechselt.

Die Anweisung wird nur in den Betriebszuständen ANLAUF und RUN ausgeführt.

Das Betriebssystem setzt die Anweisung "Schrittschaltwerk realisieren" bei Kaltstart zurück. Bei einem Neustart (Warmstart) wird die Anweisung nicht zurückgesetzt. Falls die Anweisung "Schrittschaltwerk realisieren" nach einem Neustart (Warmstart) initialisiert werden soll, müssen Sie die Anweisung im OB 100 mit dem Wert "1" am Parameter RESET aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" bewirkt das Rücksetzen des Schaltwerks. Beim Erstaufruf der Anweisung muss der Parameter RESET auf "1" gesetzt werden. Am Parameter RESET können auch Konstanten angegeben werden.
JOG	Input	BOOL	E, A, M, D, L	Eine positive Flanke (gegenüber dem letzten Aufruf) bewirkt das Weiterschalten in den nächsten Schritt, falls sich das Schaltwerk noch nicht im letzten Schritt befindet. Die Weiterschaltung erfolgt unabhängig davon, welchen Wert Sie am Parameter DRUM_EN angegeben haben.
DRUM_EN	Input	BOOL	E, A, M, D, L	Steuerparameter, der festlegt, ob ein zeitabhängiges Weiterschalten in den nächsten Schritt möglich ist. Bei einem Wert "1" ist ein zeitabhängiges Weiterschalten möglich.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des letzten Schritts Zulässige Werte sind 1 bis 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit Nr. i (gehört zum Schritt i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	E, A, M, D, L	Ausgabebit j
Q	Output	BOOL	E, A, M, D, L	Zustandsparameter, der angibt, ob die von Ihnen vorgegebene Bearbeitungszeit des letzten Schritts abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Ausgangsbits zusammengefasst in einer Variablen
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Nicht anwenderrelevant (Eingangsparameter JOG des vorhergehenden Aufrufs)
EOD	Static	BOOL	E, A, M, D, L	Identisch zum Ausgangsparameter Q
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des ersten Schritts Zulässige Werte sind 1 bis 16.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des aktuellen Schritts
DCC	Static	DWORD	E, A, M, D, L, P	Die im aktuellen Schritt noch verbleibende Bearbeitungszeit in Millisekunden (ms) (nur relevant, falls DRUM_EN = 1 und das zugehörige Ereignisbit = 1)
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Die für alle Schritte gültige Zeitbasis in Millisekunden (ms)
PREV_TIME	Static	DWORD	E, A, M, D, L oder Konstante	(Nicht anwenderrelevant: Systemzeit des vorhergehenden Aufrufs)
S_PRESET	Static	ARRAY of WORD	E, A, M, D, L oder Konstante	Eindimensionales Feld mit den Zeitfaktoren für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16]. In diesem Fall enthält S_PRESET [x] den Zeitfaktor des Schritts x.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den in jedem Schritt auszugebenden Werten, falls Sie diese nicht über S_MASK ausgeblendet haben. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält OUT_VAL [x, y] den Wert, der dem Ausgangsbit OUTy im Schritt x zugewiesen wird.
S_MASK	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den Maskenbits für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält S_MASK [x, y] das Maskenbit für den "y"-ten auszugebenden Wert im Schritt x. Bedeutung der Maskenbits: • 0: der entsprechende Wert des vorangegangenen Schritts wird dem zugehörigen Ausgangsbit zugewiesen • 1: der entsprechende Wert aus OUT_VAL wird dem zugehörigen Ausgangsbit zugewiesen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE* (W#16#...)	Erläuterung
0000	Kein Fehler
8081	Unzulässiger Wert am Parameter LST_STEP
8082	Unzulässiger Wert am DSC
8083	Unzulässiger Wert am DSP
8084	Das Produkt DCC = DTBP * S_PRESET[DSC] überschreitet den Wert 2**31-1 (ca. 24,86 Tage).
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT0 bis OUT15) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

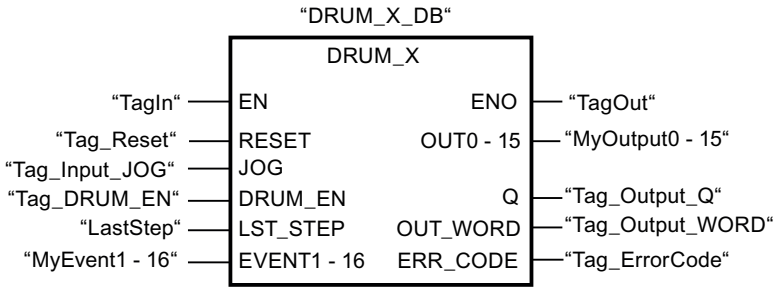

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001

Parameter	Adresse	Wert
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE

Parameter	Adresse	Wert
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	TRUE
OUT2	MyTag_Output_2	M4.2	TRUE
OUT3	MyTag_Output_3	M4.3	TRUE
OUT4	MyTag_Output_4	M4.4	TRUE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	TRUE
OUT7	MyTag_Output_7	M4.7	TRUE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	TRUE
OUT11	MyTag_Output_11	M5.3	TRUE
OUT12	MyTag_Output_12	M5.4	TRUE
OUT13	MyTag_Output_13	M5.5	TRUE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	FALSE
OUT2	MyTag_Output_2	M4.2	FALSE
OUT3	MyTag_Output_3	M4.3	FALSE
OUT4	MyTag_Output_4	M4.4	FALSE

Parameter	Operand	Adresse	Wert
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	FALSE
OUT7	MyTag_Output_7	M4.7	FALSE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	FALSE
OUT11	MyTag_Output_11	M5.3	FALSE
OUT12	MyTag_Output_12	M5.4	FALSE
OUT13	MyTag_Output_13	M5.5	FALSE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET, OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET, OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS der Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB den Signalzustand "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zu CMD

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

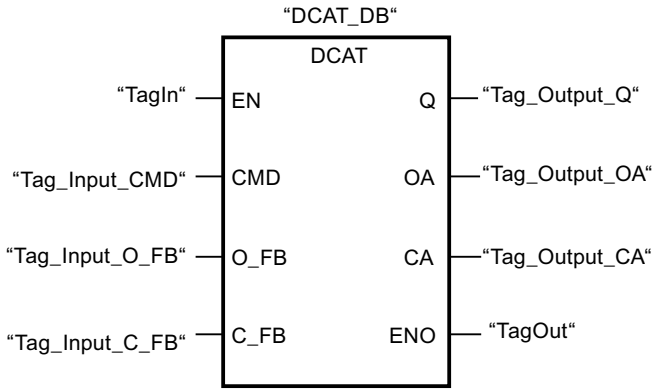

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT

Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
O_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Schließen"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

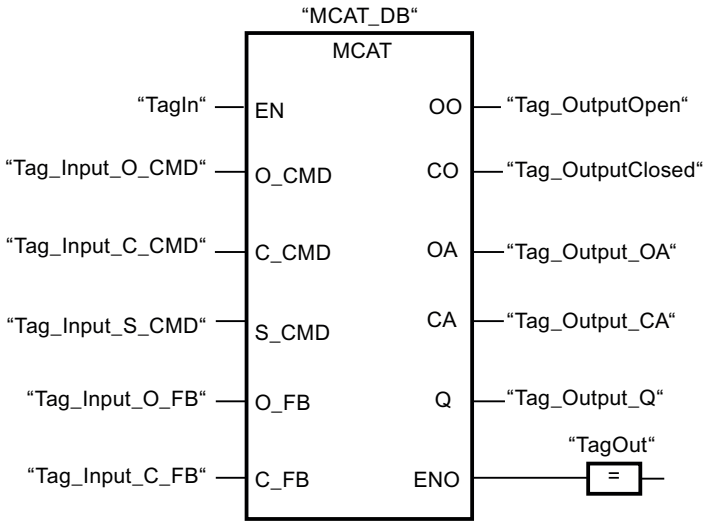

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden

Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Übereinstimmung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Übereinstimmung in den Parameter OUT_STEP angegeben. Wird keine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Übereinstimmung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY of WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als

Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

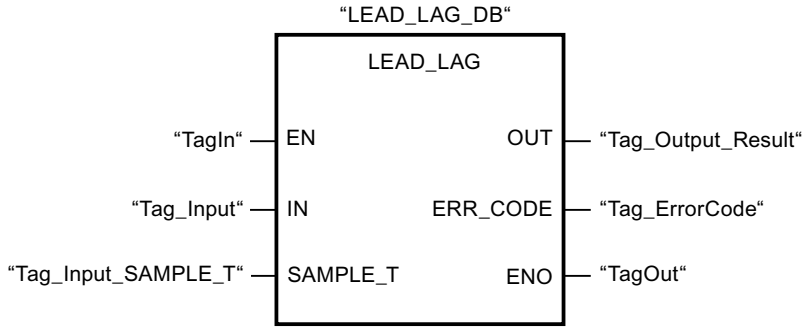

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

TONR_X: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines festgelegten Zeitraums, den Sie mithilfe des Eingangs PV angeben. Die Anweisung wird ausgeführt, wenn der Signalzustand am Starteingang TMR_EN von "0" nach "1" wechselt. Gleichzeitig läuft die Zeit mit der am Eingang PV programmierten Zeitdauer ab. Während die Anweisung ausgeführt wird, addiert sie den Zeitwert des Eingangs DELTA_T zum Zeitwert des InOut ET.

Folgende Bedingungen müssen dazu erfüllt sein:

- Der Starteingang TMR_EN liefert den Signalzustand "1".
- Der Zeitwert am InOut ET ist kleiner als der Zeitwert am Parameter PV.
- Der Eingang RESET liefert den Signalzustand "0".

Die Anweisung wird unterbrochen, wenn der Signalzustand am Starteingang TMR_EN von "1" nach "0" wechselt. Sobald der Signalzustand wieder nach "1" wechselt, läuft die Anweisung weiter.

Die Anweisung wird beendet, wenn der Zeitwert am InOut ET den Zeitwert am Eingang PV erreicht hat. Dann liefert der Ausgang Q den Signalzustand "1".

Solange der Ausgang Q den Signalzustand "1" liefert, bleibt der Zeitwert am InOut ET unverändert. Der Zeitwert und der Ausgang Q werden erst wieder auf "0" zurückgesetzt, wenn der Eingang RESET auf den Signalzustand "1" gesetzt wird.

Da die Anweisung "Zeit akkumulieren" die Zykluszeit (DELTA_T) des letzten Zyklus des Organisationsbausteins (OB) verwendet, darf sie nur in zyklischen Organisationsbausteinen aufgerufen werden.

Hinweis

Sie müssen die Zykluszeit des Organisationsbausteins aus den Anlaufinformationen in die globale Variable des Parameters DELTA_T verschieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L, T, Z	Freigabeausgang
TMR_EN (Timer Enable)	Input	BOOL	E, A, M, D, L, T, Z	Starteingang Startet die Anweisung und den Ablauf der voreingestellten Zeitdauer am Eingang PV
RESET	Input	BOOL	E, A, M, D, L, T, Z	Rücksetzeingang
PV (Preset Value)	Input	DINT	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung
DELTA_T	Input	INT	E, A, M, D, L oder Konstante	Zykluszeit des Organisationsbausteins des vorherigen Zyklus
Q	Output	BOOL	E, A, M, D, L	Ausgang, der bei Erreichen des voreingestellten Zeitwerts gesetzt wird
ET (Elapsed Time)	InOut	DINT	E, A, M, D, L	Akkumulierter Zeitwert (Aktuell abgelaufener Zeitwert + Zeitwert des Eingangs DELTA_T)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeitwert akkumulieren":

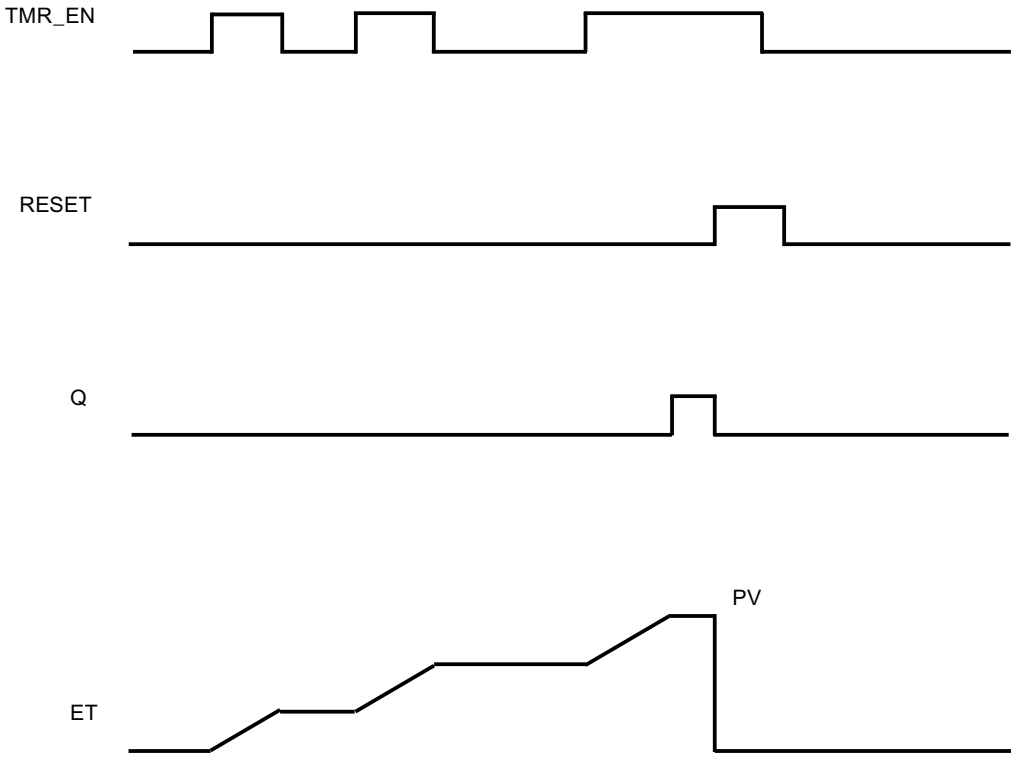

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

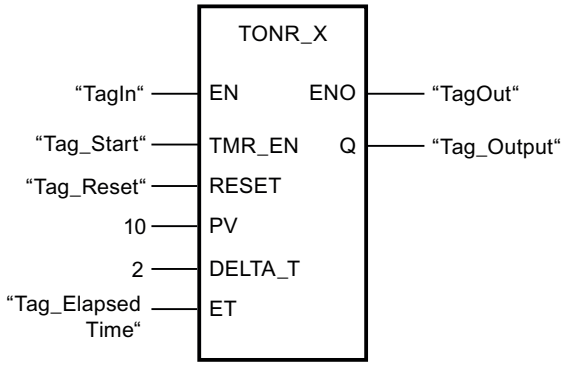

Die Anweisung wird gestartet, wenn die Operanden "TagIn" und "Tag_Start" den Signalzustand "1" liefern. Bei jedem Wechsel des Signalzustands von "0" nach "1" am Eingang TMR_EN, wird die Zykluszeit "2" am Eingang DELTA_T zum aktuellen Zeitwert des InOut ET addiert, bis der maximale Zeitwert "10" des Eingangs PV erreicht ist. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

WSR: Datum im Schieberegister speichern

Beschreibung

Mit der Anweisung "Datum im Schieberegister speichern" schreiben Sie Daten aus einer angegebenen Quelle in ein Schieberegister. Dabei wird jedes Element des Registers in die nächste Adresse verschoben. Die Daten, die in der letzten Adresse des Schieberegisters enthalten sind, gehen nach der Schiebeoperation verloren.

Bei jeder Ausführung der Anweisung mit dem Wert "0" am Parameter RESET, werden neue Daten aus der Quelle (S_DATA) gelesen und in die Anfangsadresse (START) des Schieberegisters geschoben. Der Parameter LENGTH gibt die Anzahl der zu schiebenden Elemente an. Am Parameter E_TYPE wird der Datentyp der Elemente festgelegt. Wenn am Parameter E_TYPE ein unzulässiger Datentyp angegeben wird, wird die Anweisung nicht bearbeitet.

Wenn der Parameter RESET bei der Ausführung der Anweisung auf "1" gesetzt ist, wird das Register gelöscht. Der Parameter Q wird auf den Signalzustand "1" gesetzt, wenn das Schieberegister mit Nullen belegt ist oder gelöscht wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datum im Schieberegister speichern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_DATA	Input	POINTER	E, A, M, D	Zeigt auf das Quellelement, das in die Anfangsadresse (START) geschoben wird. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
START	Input	POINTER	E, A, M, D	Zeigt auf die Anfangsadresse des Registers. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
LEN	Input	WORD	E, A, M, D, L, P	Anzahl der Elemente, die geschoben werden.
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der zu schiebenden Elemente an. Folgende Datentypen sind zulässig: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	A, M, D, L	Gibt "0" an, wenn der Parameter RESET aktiv (1) ist oder alle Elemente, die geschoben werden, den Wert "0" haben.

Beispiel

Im folgenden Beispiel werden vier Elemente des Datentyps WORD ab der am Parameter START angegebenen Adresse geschoben. Nachdem das erste Element des Registers in die nächste Adresse geschoben wurde, wird die erste Adresse mit den Daten belegt, auf die der Zeiger am Parameter S_DATA zeigt. Der Wert des letzten Elements im Register geht dabei verloren.

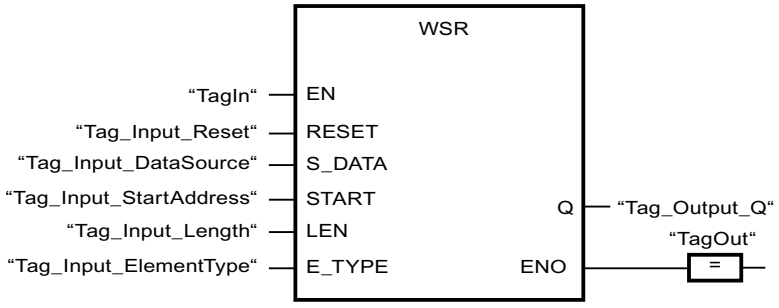

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SHRB: Bit in Bitschieberegister schieben

Beschreibung

Mit der Anweisung "Bit in Bitschieberegister schieben" können Sie ein Bit aus einer angegebenen Quelle (DATA) in ein Schieberegister schreiben. Bei jeder Ausführung der Anweisung mit einem Wert "0" am Parameter RESET werden neue Daten aus der Quelle gelesen und in die Anfangsadresse (S_BIT) des Schieberegisters geschoben. Alle folgenden Bits werden um eins verschoben. Das Bit in der letzten Adresse (S_BIT + N) geht nach dem Schieben verloren. Wenn der Parameter RESET während der Bearbeitung der Anweisung auf "1" gesetzt ist, werden die Adressen im Schieberegister auf "0" zurückgesetzt und nicht geschoben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bit in Bitschieberegister schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
DATA	Input	BOOL	E, A, M, D, L	Quellbit
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das Anfangsbit des Schieberegisters Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	WORD	E, A, M, D, L, P	Anzahl der Bits, die geschoben werden.

Beispiel

Im folgenden Beispiel wird der Parameter N auf den Wert 14 ("E" in hexadezimaler Schreibweise) gesetzt, sodass 14 Bits ab der am Parameter S_BIT angegebenen Adresse geschoben werden. Nachdem die Bits geschoben worden sind, wird die erste Adresse mit den Daten aus der Quelle am Parameter DATA belegt. Der letzte Bitwert geht verloren.

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Vor der Bearbeitung der Anweisung sind die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Nach der Bearbeitung

Nach der Bearbeitung der Anweisung werden die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

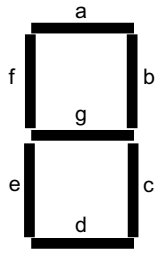

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

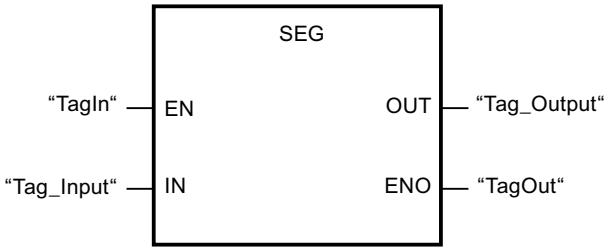

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

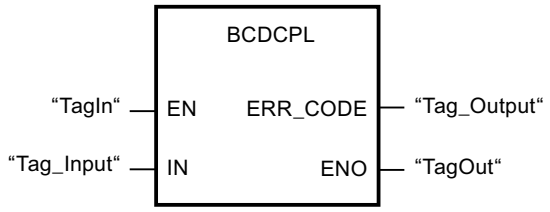

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN	Input	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
ENO	Output	BOOL	E, A, M, D, L	Freigabeausgang
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

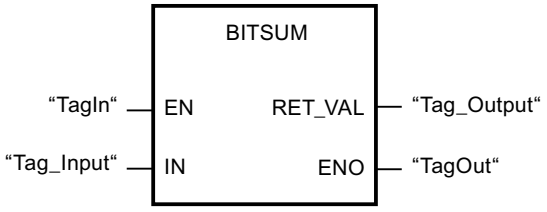

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum EN-/ENO-Mechanismus (Seite 150)

Grundlagen zu FUP (Seite 7493)

4.2.2.3 AWL

IEC-Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie einen Operanden für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die am Parameter PT programmierte Zeitdauer an. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Signals am Parameter IN. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Im Programmcode müssen Sie die Anweisung "Impuls erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TP (z. B. "TP_DB")
- Deklaration als lokale Variable vom Typ TP im Abschnitt "Static" eines Bausteins (z. B. #MyTP_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein
Q	Output	BOOL	E, A, M, D, L	Impulsausgang
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Impuls erzeugen" nach dem Starten:

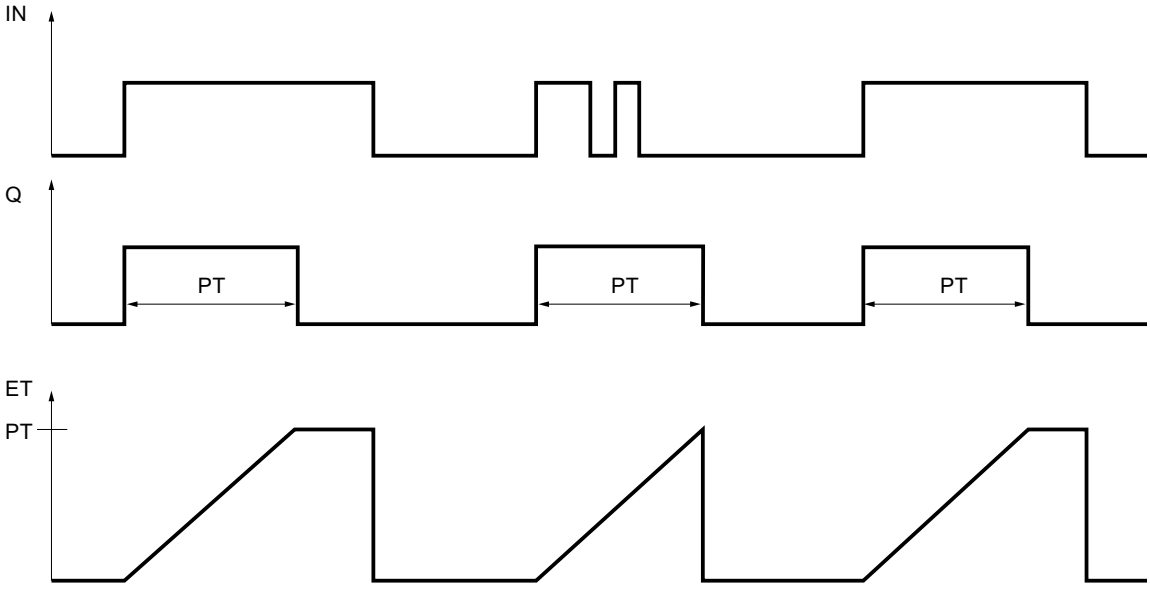

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TP, "TP_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TP_DB" wird der Anweisung zugeordnet.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Zeitdauer des Impulses
Q := "Tag_Output"	// Der Operand wird für die Zeitdauer gesetzt, die durch den Operanden "Tag_PresetTIME" festgelegt wurde.
ET := "Tag_Elapsed"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie eine positive Signalflanke um die Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, führt der Parameter Q den Signalzustand "1". Der Operand am Parameter Q bleibt so lange gesetzt, wie der Starteingang IN noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Im Programmcode müssen Sie die Anweisung "Einschaltverzögerung erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TON (z. B. "TON_DB")
- Deklaration als lokale Variable vom Typ TON im Abschnitt "Static" eines Bausteins (z. B. #MyTON_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein
Q	Output	BOOL	E, A, M, D, L	Signalzustand, der um die Zeit PT verzögert wird
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Einschaltverzögerung erzeugen" nach dem Starten:

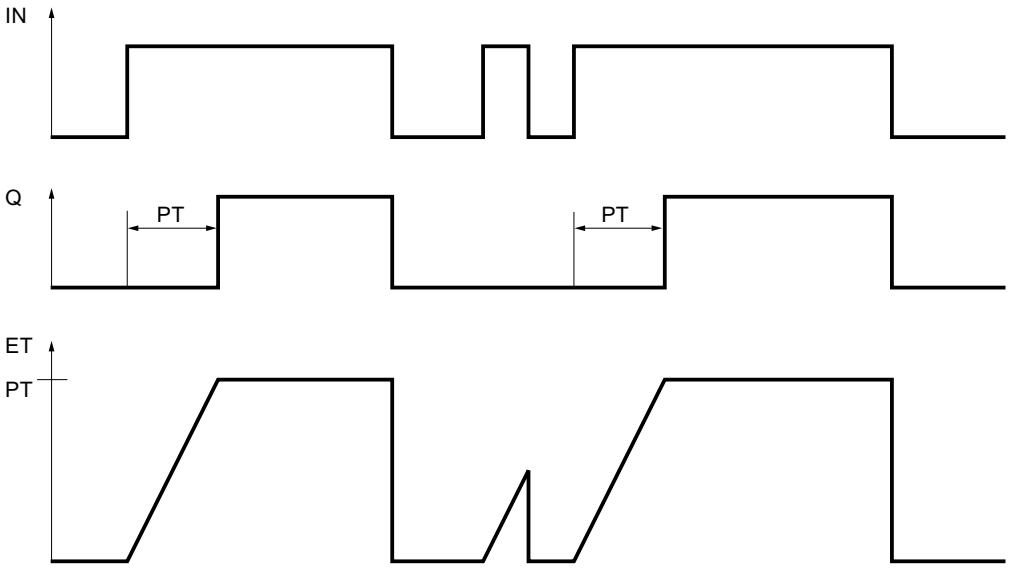

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TON, "TON_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TON_DB" wird der Anweisung zugeordnet.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Gibt die Zeitdauer an, um die die positive Signalflanke am Parameter IN verzögert wird.
Q := "Tag_Output"	// Der Operand wird gesetzt, wenn die durch die Variable "Tag_PresetTIME" vorgegebene Zeitdauer PT abgelaufen ist.
	// Der Parameter Q bleibt so lange gesetzt, wie die Variable "Tag_Start" noch "1" führt.
	// Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie eine negative Signalflanke um die Zeitdauer PT. Der Operand am Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Operand am Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der aktuelle Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET solange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt.

Im Programmcode müssen Sie die Anweisung "Ausschaltverzögerung erzeugen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TOF (z. B. "TOF_DB")
- Deklaration als lokale Variable vom Typ TOF im Abschnitt "Static" eines Bausteins (z. B. #MyTOF_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Zurücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein
Q	Output	BOOL	E, A, M, D, L	Signalzustand, der um die Zeit PT verzögert wird
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Verhalten der Anweisung "Ausschaltverzögerung erzeugen" nach dem Starten:

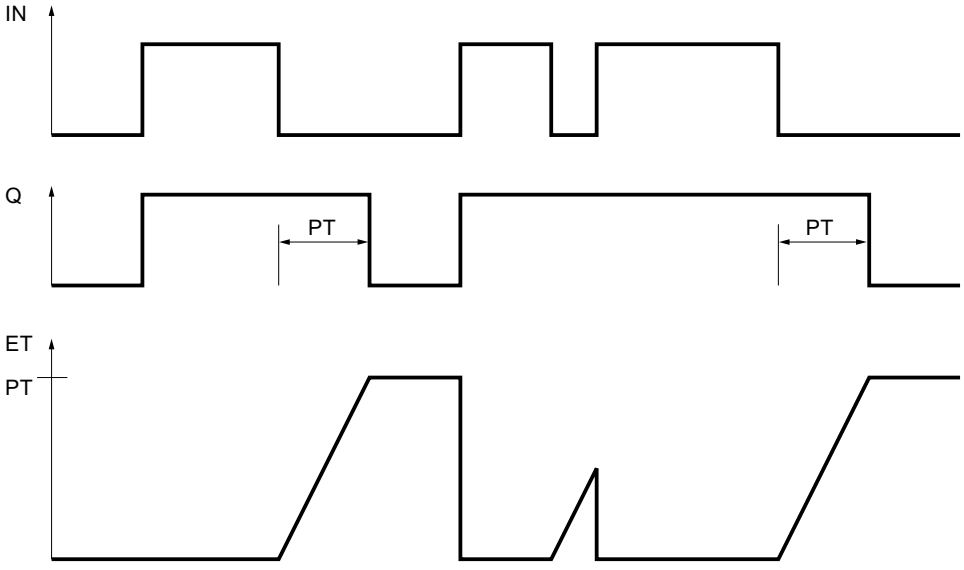

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TOF, "TOF_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "TOF_DB" wird der Anweisung zugeordnet.
IN := "Tag_Start"	// Bei einer positiven Signalflanke am Operanden wird die Anweisung ausgeführt.
PT := "Tag_PresetTIME"	// Gibt die Zeitdauer an, um die die negative Signalflanke am Parameter IN verzögert wird.
Q := "Tag_Output"	// Der Operand wird gesetzt, wenn die Anweisung durch eine positive Signalflanke am Parameter IN gestartet wird.
	// Bei einem Wechsel von "1" auf "0" am Parameter IN bleibt der Operand "Tag_Output" gesetzt, solange die durch die Variable "Tag_PresetTIME" vorgegebene Zeitdauer läuft.
	// Wenn die Zeit am Parameter PT abgelaufen ist, wird der Operand zurückgesetzt.
ET := "Tag_ElapsedTIME"	// Aktueller Zeitwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

IEC-Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Zählwert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des Datentyps INT erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange der Parameter R den Signalzustand "1" führt, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

Im Programmcode müssen Sie die Anweisung "Vorwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTU (z. B. "CTU_DB")
- Deklaration als lokale Variable vom Typ CTU im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL CTU, "CTU_DB"	// Die Anweisung wird aufgerufen. Der Datenbaustein "CTU_DB" wird der Anweisung zugeordnet.
CU := "Tag_StartCTU"	// Wenn der Signalzustand des Operanden "Tag_StartCTU" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um Eins erhöht.
	// Der Zählwert wird so lange erhöht, bis er den oberen Grenzwert von INT = 32767 erreicht.
R := "Tag_ResetCounter"	// Wenn der Signalzustand des Operanden "Tag_ResetCounter" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf "0" zurückgesetzt.
PV := "Tag_PresetValue"	// Der Operand bestimmt, ab welchem Wert der Operand am Parameter Q gesetzt wird.
Q := "Tag_CounterStatus"	// Der Operand ist gesetzt, solange der aktuelle Zählwert größer oder gleich dem Wert am Parameter PV ist.
CV := "Tag_CounterValue"	// Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen bearbeiten (Seite 7592)

AWL-Anweisungen einfügen (Seite 7572)

Instanzen (Seite 64)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Zählwert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps INT erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange der Parameter LD den Signalzustand "1" führt, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

Im Programmcode müssen Sie die Anweisung "Rückwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTD (z. B. "CTD_DB")
- Deklaration als lokale Variable vom Typ CTD im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben. Falls Instanzen der Anweisung "Rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL CTD, "CTD_DB"           // Die Anweisung wird aufgerufen. Der Datenbaustein
                             "CTD_DB" wird der Anweisung zugeordnet.
```

AWL	Erläuterung
CD := "Tag_StartCTD"	// Wenn der Signalzustand des Operanden "Tag_StartCTD" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins verringert. // Der Zählwert am Parameter CV wird so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.
LD := "Tag_LoadPV"	// Wenn der Signalzustand des Operanden "Tag_LoadPV" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf den Wert des Operanden "Tag_PresetValue" gesetzt.
PV := "Tag_PresetValue"	// Gibt den Wert an, auf welchen der Zähler bei einem Signalzustand "1" am Parameter LD gesetzt wird.
Q := "Tag_CounterStatus"	// Der Operand wird gesetzt, wenn der aktuelle Zählwert kleiner oder gleich Null ist.
CV := "Tag_CounterValue"	// Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des angegebenen Datentyps INT erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hochgezählt. Wenn der untere Grenzwert des angegebenen Datentyps INT erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange der Parameter R den Signalzustand "1" führt, hat eine Änderung im Signalzustand

der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

Im Programmcode müssen Sie die Anweisung "Vorwärts und rückwärts zählen" mit der Anweisung "Baustein aufrufen" (CALL) aufrufen.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTUD (z. B. "CTUD_DB")
- Deklaration als lokale Variable vom Typ CTUD im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	**Erläuterung**
`CALL CTUD, "CTUD_DB"`	`// Die Anweisung wird aufgerufen. Der Datenbaustein "CTUD_DB" wird der Anweisung zugeordnet.`
`CU := "Tag_StartCTU"`	`// Wenn der Signalzustand des Operanden "Tag_StartCTU" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins erhöht.`
	`// Der Zählwert wird bei einer positiven Flanke am Parameter CU so lange erhöht, bis er den oberen Grenzwert von INT = 32767 erreicht.`
`CD := "Tag_StartCTD"`	`// Wenn der Signalzustand des Operanden "Tag_StartCTD" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der aktuelle Wert des Operanden "Tag_CounterValue" um eins verringert.`
	`// Der Zählwert am Parameter CV wird so lange verringert, bis er den unteren Grenzwert von -32768 erreicht.`
`R := "Tag_ResetCounter"`	`// Wenn der Signalzustand des Operanden "Tag_ResetCounter" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf "0"zurückgesetzt.`
`LD := "Tag_LoadPV"`	`// Wenn der Signalzustand des Operanden "Tag_LoadPV" auf "1" wechselt, wird der Operand "Tag_CounterValue" auf den Wert des Operanden "Tag_PresetValue" gesetzt.`
`PV := "Tag_PresetValue"`	`// Gibt den Wert an, auf welchen der Zähler bei einem Signalzustand "1" am Parameter LD gesetzt wird.`
`QU := "Tag_CounterStatus"`	`// Der Operand ist gesetzt, solange der aktuelle Zählwert größer oder gleich dem Wert am Parameter PV ist.`

AWL	Erläuterung
QD := "Tag_CounterStatus"	// Der Operand wird gesetzt, wenn der aktuelle Zählwert kleiner oder gleich Null ist.
CV := "Tag_CounterValue"	//Aktueller Zählwert

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Instanzen (Seite 64)

Mathematische Funktionen

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den kleinsten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Parameter OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Operanden sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Parameter INn und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL MIN                     // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
IN1 := "TagIn_Value1"        // Erster Eingangswert, der verglichen wird.
IN2 := "TagIn_Value2"        // Zweiter Eingangswert, der verglichen wird.
IN3 := "TagIn_Value3"        // Dritter Eingangswert, der verglichen wird.
OUT := "Tag_Minimum"         // Der Operand "Tag_Minimum" wird mit dem Wert des
                             Operanden "TagIn_Value1" beschrieben, da dieser den
                             kleinsten Wert aufweist.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den größten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Operanden sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Parameter INn und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL MAX                     // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                                liste "???" aus.
value_type := Int            // Datentyp der Parameter IN und OUT
IN1 := "TagIn_Value1"        // Erster Eingangswert, der verglichen wird.
IN2 := "TagIn_Value2"        // Zweiter Eingangswert, der verglichen wird.
IN3 := "TagIn_Value3"        // Dritter Eingangswert, der verglichen wird.
```

AWL	Erläuterung
`OUT := "Tag_Maximum"`	`// Der Operand "Tag_Maximum" wird mit dem Wert des Operanden "TagIn_Value2" beschrieben, da dieser den größten Wert aufweist.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Maximum	14444

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Die Ausführung der Anweisung setzt voraus, dass alle Variablen an den Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Parameter MN ist größer als der Wert am Parameter MX.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L. P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Parameter MN, IN, MX und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL LIMIT                   // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
MN := "Tag_LowLimit"         // Untergrenze
IN := "Tag_InputValue"       // Eingangswert
MX := "Tag_HighLimit"        // Obergrenze
OUT := "Tag_Result"          // Der Operand "Tag_Result" wird mit dem Wert des Ope-
                             randen "Tag_LowLimit" beschrieben, da der Wert des
                             Operanden "Tag_InputValue" außerhalb des definierten
                             Grenzbereichs liegt.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Verschieben

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

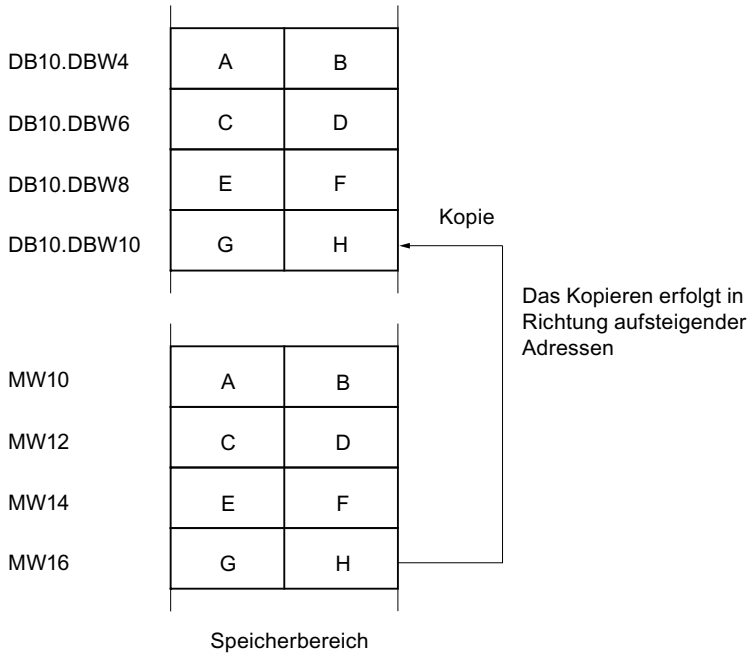

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Solange der Quellbereich nicht Teil eines Datenbausteins ist, der nur im Ladespeicher vorhanden ist, gibt es keine Begrenzung der Schachtelungstiefe.

Bei Unterbrechung einer BLKMOV-Bearbeitung hingegen, bei der aus einem nicht ablaufrelevanten DB kopiert wird, kann eine solche BLKMOV-Bearbeitung nicht mehr eingeschachtelt werden.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Nicht ablaufrelevante Datenbausteine

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Regeln beim Kopieren von nicht ablaufrelevanten Datenbausteinen

Der Quellbereich kann auch in einem nicht ablaufrelevanten Datenbaustein im Ladespeicher liegen. Datenbausteine, die nicht ablaufrelevant sind, werden mit dem Schlüsselwort UNLINKED gekennzeichnet.

Wenn ein nicht ablaufrelevanter Datenbaustein mit der Anweisung "Bereich kopieren" in den Arbeitsspeicher kopiert und gleichzeitig z. B. durch das Programmiergerät geladen wird, kann die Bearbeitung der Anweisung um mehrere Millisekunden verzögert werden. Dies führt zu einer Verlängerung des OB-Zyklus und kann zum Ansprechen der Zykluszeitüberwachung führen.

Wenn mit der Anweisung "Bereich kopieren" ein nicht ablaufrelevanter Datenbaustein kopiert wird und der Kopiervorgang unterbrochen wird, kann die Bearbeitung der Anweisung nicht mehr fortgesetzt werden.

Falls auf Ihrer CPU die Anweisung "Aus Datenbaustein im Ladespeicher lesen" vorhanden ist, müssen Sie diese Anweisung zum Lesen von nicht ablaufrelevanten Datenbausteinen im Ladespeicher benutzen. Wenn Sie die Anweisung "Bereich kopieren" benutzen, wird der Fehler W#16#8092 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL BLKMOV	// Die Anweisung wird aufgerufen.
SRCBLK := P#M100.0 BYTE 10	// 10 Bytes ab MB100 werden kopiert.
RET_VAL := "Tag_ErrorCode"	// Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in die Variable "Tag_ErrorCode" ausgegeben.
DSTBLK := P#M200.0 BYTE 10	// Die kopierten Bytes werden in die 10 Bytes ab MB200 geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zu AWL (Seite 7557)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 512 Bytes kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL UBLKMOV	// Die Anweisung wird aufgerufen.
SRCBLK := P#M100.0 BYTE 10	// 10 Bytes ab MB100 werden kopiert.
RET_VAL := "Tag_ErrorCode"	// Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in die Variable "Tag_ErrorCode" ausgegeben.
DSTBLK := P#M200.0 BYTE 10	// Die kopierten Bytes werden in die 10 Bytes ab MB200 geschrieben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zu AWL (Seite 7557)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

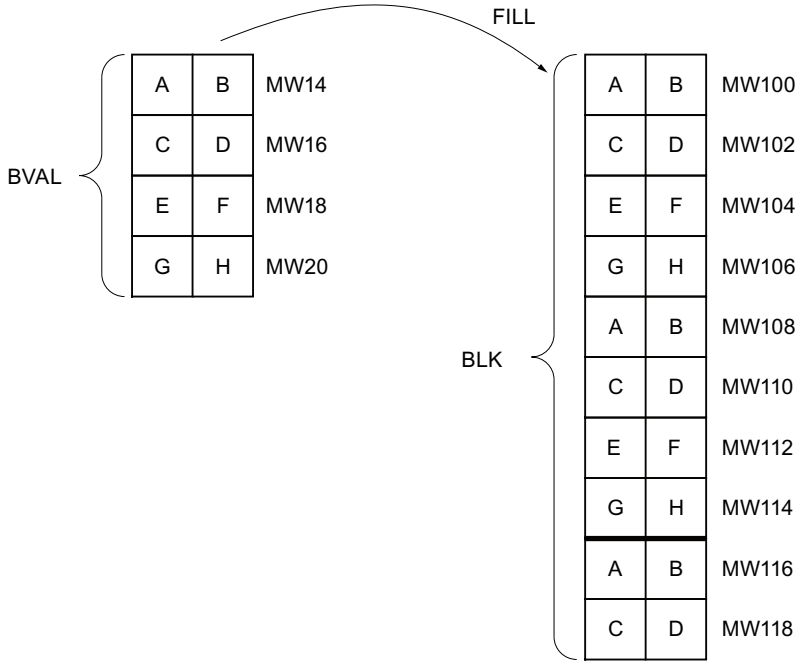

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL FILL                    // Die Anweisung "Bereich befüllen" wird aufgerufen.
BVAL := P#M14.0 WORD 4       // Der Bereich von MW14 bis MW20 wird kopiert.
RET_VAL := "Tag_ErrorCode"   // Fehlerinformationen
BLK := P#M100.0 WORD 10      // Der Speicherbereich von MW100 bis MW118 wird mit
                             dem Inhalt der 4 Wörter befüllt, die im Speicherbe-
                             reich am Parameter BVAL enthalten sind.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zu AWL (Seite 7557)

Umwandler

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1"
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SCALE                   // Die Anweisung wird aufgerufen.
IN := "Tag_InputValue"       // Gibt den Wert an, der umgewandelt und skaliert wird.
HI_LIM := "Tag_HighLimit"    // Oberer Grenzwert
LO_LIM := "Tag_LowLimit"     // Unterer Grenzwert
BIPOLAR := "Tag_Bipolar"     // Gibt an, ob der Wert am Parameter IN als bipolar
                             oder unipolar interpretiert wird.
RET_VAL := "Tag_ErrorCode"   // Fehlerinformationen
```

```
AWL                          Erläuterung
OUT := "Tag_OutputValue"     // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zu AWL (Seite 7557)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandelt sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).

Fehler-code* (W#16#...)	Erläuterung
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                              Erläuterung
CALL UNSCALE                     // Die Anweisung wird aufgerufen.
IN := "Tag_InputValue"           // Gibt den Wert an, der deskaliert wird.
HI_LIM := "Tag_HighLimit"        // Oberer Grenzwert
LO_LIM := "Tag_LowLimit"         // Unterer Grenzwert
BIPOLAR := "Tag_Bipolar"         // Gibt an, ob der Wert am Parameter IN als bipolar
                                 oder unipolar interpretiert wird.
RET_VAL := "Tag_ErrorCode"       // Fehlerinformationen
OUT := "Tag_OutputValue"         // Ergebnis der Anweisung
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zu AWL (Seite 7557)

Programmsteuerung

Laufzeitsteuerung

COMPRESS: CPU-Speicher komprimieren

Beschreibung

Mit der Anweisung "CPU-Speicher komprimieren" komprimieren Sie den Arbeitsspeicher der CPU bzw. den RAM-Ladespeicher. Dabei werden Lücken, die durch Löschen oder Laden von Bausteinen entstanden sind, aufgefüllt, indem die vorhandenen Bausteine zusammengeschoben werden.

Mit dem Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Der Komprimiervorgang wird über mehrere Programmzyklen verteilt. Die Überwachung des Komprimiervorgangs erfolgt über die Werte der einzelnen Parameter. Wenn am Parameter BUSY der Anweisung der Wert "1" ansteht, ist der Komprimiervorgang noch in Ausführung. Der Wert "1" am Parameter DONE meldet, dass der Komprimiervorgang beendet ist. Fehlerinformationen können am Parameter RET_VAL abgefragt werden.

Wenn bereits ein extern angestoßener Komprimiervorgang ausgeführt wird, führt der Aufruf der Anweisung "CPU-Speicher komprimieren" zur Fehleranzeige.

Kontrolle über den Komprimiervorgang

Bei einem einmaligen Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Sie haben jedoch keine Kontrolle darüber, ob das Komprimieren erfolgreich durchgeführt wurde.

Wenn Sie diese Kontrolle wünschen, müssen Sie wie folgt vorgehen:

Die Anweisung "CPU-Speicher komprimieren" ist im Zyklus aufzurufen. Nach jedem Aufruf ist zunächst der Parameter RET_VAL zu bewerten. Für den Fall, dass er den Wert "0" hat, sind die Parameter BUSY und DONE zu bewerten. Ist BUSY = "1" und DONE = "0", so weist dies darauf hin, dass der Komprimiervorgang noch aktiv ist. Erst wenn BUSY den Wert "0" und DONE den Wert "1" annimmt, wurde der Komprimiervorgang erfolgreich beendet. Falls danach die Anweisung "CPU-Speicher komprimieren" wieder aufgerufen wird, wird erneut ein Komprimieren angestoßen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CPU-Speicher komprimieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BUSY	Output	BOOL	E, A, M, D, L	Information, ob Komprimiervorgang aktiv ist
DONE	Output	BOOL	E, A, M, D, L	Information, ob der Komprimiervorgang beendet wurde

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler Der Status des Komprimiervorgangs kann an den Parametern BUSY und DONE abgefragt werden.
8091	Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt.
8092	Die Anweisung kann nicht ausgeführt werden, da eine der folgenden Bedingungen zutrifft: • Die Funktion "Baustein löschen" ist aktiv. • Eine PG-Funktion greift auf einen zu verschiebenden Baustein zu. • Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt. • Die H-CPU befindet sich einem Ankoppel- oder Aktualisierungsvorgang.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zu AWL (Seite 7557)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufgerufen wird, wird die Anweisung nicht ausgeführt.

Die Anweisung "Zyklusüberwachungszeit neu starten" läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zu AWL (Seite 7557)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zu AWL (Seite 7557)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine parametrierte Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) parametrieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Gültig für S7-300-CPUs:

Bei den S7-300-CPUs (mit Ausnahme von S7-318) ist die mit der Anweisung programmierte Verzögerungszeit eine Mindestzeit. Sie verlängert sich um die Ausführungszeit der eingeschachtelten Prioritätsklassen sowie um Systemlasten.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zu AWL (Seite 7557)

PROTECT: Schutzstufe ändern

Beschreibung

Mit der Anweisung "Schutzstufe ändern" ändern Sie die projektierte Schutzstufe "1" der CPU ab.

Die parametrierte Schutzstufe "1" mit aktivierter Option "Durch Passwort aufhebbar" bewirkt, dass man ein Passwort parametrieren muss. Mit diesem Passwort können Sie auf die mit der Anweisung "Schutzstufe ändern" schreibgeschützte CPU im Betrieb schreibend zugreifen (z. B. Laden).

Die Schutzstufe legen Sie am Parameter MODE der Anweisung fest. Die aktuelle Schutzstufe der CPU wird im Inspektorfenster unter "Eigenschaften > Schutz" angezeigt.

Die folgende Tabelle zeigt, welche Bedeutung die Schutzstufen haben:

Schutz-stufe	Bedeutung
1	Alle PG-Funktionen sind erlaubt. Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
2	• Das Programm und die Konfiguration der CPU können nicht geändert werden. Das Programm in der CPU kann gelesen werden. • Die Funktionen für Prozessführung, Prozessbeobachtung und Prozesskommunikation sind erlaubt. • Alle Auskunftsfunktionen sind erlaubt. • Mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreibschutz aufheben. • Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
3	• Auch mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreib- und Leseschutz nicht aufheben. • Wenn zum Zeitpunkt des Aufrufs der Anweisung "Schutzstufe ändern" eine legitimierte Verbindung vorhanden ist, bleibt der Aufruf der Anweisung für diese Verbindung ohne Wirkung.

Wenn Sie für Ihre CPU die Schutzstufe "2" oder "3" projektiert haben, hat der Aufruf der Anweisung "Schutzstufe ändern" keine Wirkung.

Die Wirksamkeit der Schutzstufe "2" bzw. "3" nach Ausführung der Anweisung "Schutzstufe ändern" können Sie der SZL-Teilliste mit der SZL-ID W#16#0232 und dem Index W#16#0004 entnehmen.

Hinweis

Mit der Anweisung "Schutzstufe ändern" können Sie keine niedrigere Schutzstufe einstellen als die, die Sie mit STEP 7 "Hardware konfigurieren" projektiert haben. Sie haben z. B. die Schutzstufe 3 projektiert und anschließend die Anweisung "Schutzstufe ändern" mit dem Parameter MODE = 12 aufgerufen. Dies bewirkt eine Sperre der Passwort-Legitimierung. Ein erneuter Aufruf der Anweisung mit dem Parameter MODE = 0 oder 1 bewirkt, dass die Schutzstufe 3 beibehalten wird und dass der Schreib- und Leseschutz mit Eingabe des gültigen Passworts aufgehoben wird.

Auswirkungen der Betriebszustandsübergänge auf die Schutzstufe

Die folgende Tabelle zeigt, welche Auswirkungen die Betriebszustandsübergänge auf die mit der Anweisung "Schutzstufe ändern" eingestellte Schutzstufe haben.

Aktion	Auswirkungen auf die Schutzstufe
• Betriebsartenschalter auf STOP • Ungepuffertes NETZ-EIN • Urlöschen durch Betriebsartenschalter • Konfigurations- und/oder Programmänderungen im Betriebszustand STOP der S7-400-CPU	Die Schutzstufe wird auf "1" gesetzt. Alle PG-Funktionen sind erlaubt.
• Kaltstart • Neustart (Warmstart)	Die Schutzstufe wird durch das Betriebssystem auf "1" gesetzt. Alle PG-Funktionen sind erlaubt. Bei Bedarf können Sie durch Aufruf der Anweisung "Schutzstufe ändern" in Ihrem Programm die Schutzstufe auf "2" oder "3" setzen.
Gepuffertes NETZ-EIN	Bleibt unverändert
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch den Aufruf der Anweisung "Programm beenden" (STP), durch Programmfehler ohne zugehörigen Fehler-OB oder durch Anwenderaktion).	Bleibt unverändert
S7-400: Wiederanlauf durch Anwenderaktion oder automatisch nach NETZ-EIN	Bleibt unverändert

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schutzstufe ändern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	WORD	E, A, M, D, L oder Konstante	Schutzstufe Mögliche Werte: • W#16#0000: Einstellen der Schutzstufe "1" • W#16#0001: Einstellen der Schutzstufe "2" mit Passwort-Legitimierung • W#16#000C: Einstellen der Schutzstufe "3" ohne Passwort-Legitimierung Am Parameter MODE können auch Konstanten angegeben werden.
RET_VAL	Output	INT	E, A, M, D, L	Tritt während der Bearbeitung ein Fehler auf, wird eine Fehlerinformation ausgegeben.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
8090	Unzulässiger Wert am Parameter MODE.
80C3	Die benötigten Betriebsmittel sind momentan belegt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Die CPU soll im Betriebszustand RUN grundsätzlich vor Zugriffen geschützt werden. Der Schutz soll nur aufgehoben werden, wenn Sie den Betriebsartenschalter auf STOP schalten. Diese Funktionalität erreichen Sie, indem Sie in einem Anlauf-OB die Anweisung "Schutzstufe ändern" aufrufen und mit dem Wert W#16#0001 (Schutzstufe "2") am Parameter MODE ausführen. Dadurch bleibt auch nach einem Neustart (Warmstart) die Schutzstufe "2" eingestellt.

```
AWL                            Erläuterung
CALL PROTECT                   // Die Anweisung wird aufgerufen.
MODE := "Tag_ProtectionLevel"  // Schutzstufe einstellen
RET_VAL := "Tag_ErrorCode"     // Fehlerinformation
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu AWL (Seite 7557)

CiR: CiR-Vorgang steuern

Beschreibung

Mit der Anweisung "CiR-Vorgang steuern" beeinflussen Sie im Anwenderprogramm den CiR-Vorgang wie folgt:

- Den CiR-Vorgang sperren. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU stets abgelehnt. Die Sperre gilt so lange, bis Sie diese mit der Anweisung "CiR-Vorgang steuern" wieder aufheben.
- Den CiR-Vorgang bedingt sperren, indem Sie am Parameter FRZ_TIME eine Obergrenze für die CiR-Synchronisationszeit festlegen. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU nur dann zugelassen, wenn die in der CPU errechnete CiR-Synchronisationszeit die angegebene Obergrenze nicht überschreitet.
- Den CiR-Vorgang freigeben. Die Obergrenze der CiR-Synchronisationszeit wird auf den Defaultwert 1000 ms gesetzt.
- Ermittelt, ob der CiR-Vorgang freigegeben ist oder nicht. Falls der CiR-Vorgang freigegeben oder bedingt freigegeben ist, erhalten Sie im Parameter A_FT der Anweisung die aktuelle Obergrenze für die CiR-Synchronisationszeit.

Hinweis

Während der CiR-Synchronisationszeit sind die Ausgänge eingefroren und die Eingänge werden nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CiR-Vorgang steuern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte am Parameter MODE sind: • 0: Auskunftsfunktion • 1: CiR-Vorgang freigeben • 2: CiR-Vorgang sperren • 3: CiR-Vorgang bedingt sperren. Die Obergrenze der CiR-Synchronisationszeit legen Sie in FRZ_TIME fest.
FRZ_TIME	Input	TIME	E, A, M, D, L oder Konstante	Obergrenze der CiR-Synchronisationszeit in Millisekunden. Zulässiger Wertebereich: 60 ... 2500 ms (Voreinstellung: 1000 ms) Der Parameter FRZ_TIME ist nur dann relevant, wenn der Parameter MODE auf den Wert "3" gesetzt ist.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation Wenn der Parameter MODE den Wert "0" hat, liefert der Parameter RET_VAL Informationen über die Freigabe des CiR-Vorgangs.
A_FT	Output	TIME	E, A, M, D, L	Aktuell gültige Obergrenze der CiR-Synchronisationszeit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt. Dieser Fehlercode kann ausgegeben werden, nur wenn der Parameter MODE auf den Wert "1", "2" oder "3" gesetzt ist.
0001	Der CiR-Vorgang ist freigegeben. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.

Fehler-code* (W#16#...)	Erläuterung
0002	Der CiR-Vorgang ist gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0003	Der CiR-Vorgang ist bedingt gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
8001	Die CPU ist nicht bereit für einen CiR-Vorgang. Sie verwenden eine H-CPU im H-System (Solobetrieb) oder Sie arbeiten mit einer Standard-CPU im Multicomputing-Betrieb.
8002	Der Wert am Parameter MODE ist nicht zulässig.
8003	Der Wert am Parameter FRZ_TIME ist nicht zulässig.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Mit der Anweisung "CiR-Vorgang steuern" können Sie beispielsweise programmieren, dass kein CiR-Vorgang in Zeitabschnitten angestoßen wird, in denen für die Prozessbearbeitung die maximale Leistungsfähigkeit der CPU benötigt wird.

Dazu muss die Anweisung "CiR-Vorgang steuern" vor Beginn des Zeitabschnitts mit erhöhter Prozessaktivität im Programm aufgerufen und mit dem Wert "2" (CiR-Vorgang sperren) am Parameter MODE ausgeführt werden. Dadurch wird den CiR-Vorgang gesperrt.

Um den CiR-Vorgang nach der Beendigung des Zeitabschnitts mit erhöhter Prozessaktivität wieder freizugeben, muss die Anweisung "CiR-Vorgang steuern" erneut aufrufen und mit dem Wert "1" am Parameter MODE ausführt werden. Sie können alternativ den Parameter MODE auch auf den Wert "3" setzen und dadurch den CiR-Vorgang bedingt sperren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zu AWL (Seite 7557)

Wortverknüpfungen

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit am Parameter OUT, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Parameter IN größer als 31 ist, wird eine Modulo-32-Operation ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD, INT	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird
OUT	Output	DWORD	E, A, M, D, L, P	Ausgangswert

Aus der Klappliste "???" können Sie den Datentyp des Parameters IN wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	**Erläuterung**
`CALL DECO`	`// Die Anweisung wird aufgerufen.`
	`// Wählen Sie den gewünschten Datentyp aus der Klappliste "???" aus.`
`IN := "Tag_Input"`	`// Eingangswert, der das zu setzende Bit im Ausgangswert vorgibt.`
`OUT := "Tag_Output"`	`// Ergebnis`

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

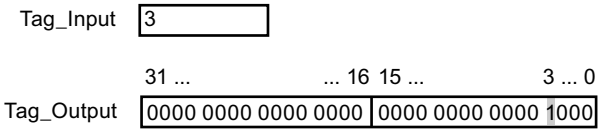

Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "Tag_Input" am Eingang ab und setzt das dritte Bit im Wert des Operanden "Tag_Output" am Ausgang.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Parameter OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Parameter IN und schreibt dessen Bitnummer in den Operanden am Parameter OUT. Wenn der Parameter IN den Wert DW#16#00000001 oder DW#16#00000000 liefert, wird am Parameter OUT der Wert "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL ENCO	// Die Anweisung wird aufgerufen.
IN := "Tag_Input"	// Eingangswert, dessen niederwertigstes gesetztes Bit abgelesen wird.
OUT := "Tag_Output"	// Bitnummer des niederwertigsten gesetzten Bits im Eingangswert

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

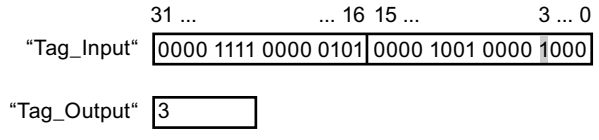

Die Anweisung selektiert das niederwertigste gesetzte Bit der Variablen "Tag_Input" und schreibt dessen Bitposition "3" in die Variable "Tag_Output".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
G	Input	BOOL	E, A, M, D, L	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Parameter IN0, IN1 und OUT wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SEL                     // Die Anweisung wird aufgerufen.
                             // Wählen Sie den gewünschten Datentyp aus der Klapp-
                             liste "???" aus.
G := "Tag_Input_G"           // Signalzustand des Schalters
IN0 := "Tag_Input0"          // Erster Eingangswert
IN1 := "Tag_Input1"          // Zweiter Eingangswert
OUT := "Tag_Output"          // Wert des ausgewählten Eingangs
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Weitere Anweisungen

SET: Bereich bitweise setzen

Beschreibung

Mit der Anweisung "Bereich bitweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "1". Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu setzenden Bits im angegebenen Bereich legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Hinweis

Parameter N

Sie können am Parameter N auch ein Element eines Datenbausteins verschalten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der zu setzenden Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL SET	// Die Anweisung "Bereich bitweise setzen" wird aufgerufen
S_BIT := P#M0.0	// Zeiger auf das erste zu setzende Bit (M0.0)
N := 10	// Anzahl der zu setzenden Bits

Im Beispiel setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

SETP: Bitfeld im Peripheriebereich setzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich setzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "1". Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge gesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das gesetzt wird. Die Bits des Feldes müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zu setzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Setzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "1" gesetzt.

Die Anweisung "Bitfeld im Peripheriebereich setzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich setzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Parameter N liefert den Wert "0".
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu setzenden Bits
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste zu setzende Bit

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL SETP	// Die Anweisung wird aufgerufen.
N := 8	// Anzahl der zu setzenden Bits
RET_VAL := "Tag_ErrorCode"	// Fehlerinformation
SA := P#10.0	// Zeiger auf das erste zu setzende Bit (A10.0)

Im Beispiel setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

SETI: Bereich byteweise setzen

Beschreibung

Mit der Anweisung "Bereich byteweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "1". Den Anfangspunkt des Bereichs definieren Sie am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zu setzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich P im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise setzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des zu setzenden Bereichs Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SETI                    // Die Anweisung wird aufgerufen.
S_BYTE := P#2.0              // Zeiger auf das erste Byte des Bereichs(P2.0)
N := 16                      // Anzahl der zu setzenden Bits
```

Im Beispiel setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7 (Peripherie-Ausgänge).

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

RESET: Bereich bitweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich bitweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zurückzusetzenden Bits im angegebenen Bereich legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich bitweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der Bits, die zurückgesetzt werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL RESET	// Die Anweisung wird aufgerufen.
S_BIT := P#M0.0	// Zeiger auf das erste Bit (M0.0), das zurückgesetzt wird.

AWL	Erläuterung
N := 10	// Anzahl der Bits, die zurückgesetzt werden

Im Beispiel setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.2 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

RESETP: Bitfeld im Peripheriebereich rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich rücksetzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "0" zurück. Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge zurückgesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das zurückgesetzt wird. Die Bits des Felds müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zurückzusetzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Rücksetzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "0" zurückgesetzt.

Die Anweisung "Bitfeld im Peripheriebereich rücksetzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich rücksetzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Parameter N liefert den Wert "0".
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld im Peripheriebereich rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der Bits, die zurückgesetzt werden.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste Bit, das zurückgesetzt wird.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL RESETP	// Die Anweisung wird aufgerufen.
N := 8	// Anzahl der Bits, die zurückgesetzt werden.
RET_VAL := "Tag_ErrorCode"	// Fehlerinformation
SA := P#10.0	// Zeiger auf das erste Bit (A10.0), das zurückgesetzt wird.

Im Beispiel setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

RESETI: Bereich byteweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich byteweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zurückzusetzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich P im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise rücksetzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich byteweise rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des Bereichs, dessen Bits zurückgesetzt werden. Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL RESETI                  // Die Anweisung wird aufgerufen.
S_BYTE := P#2.0              // Zeiger auf das erste Byte des Bereichs(P2.0)
N := 16                      // Anzahl der Bits, die zurückgesetzt werden.
```

Im Beispiel setzt die Anweisung 2 Bytes (16 Bits) im Bereich von P2.0 bis P3.7 (Peripherie-Ausgänge) zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

REPL_VAL: Ersatzwert eintragen

Beschreibung

Mit der Anweisung "Ersatzwert eintragen" tragen Sie von einem Synchronfehler-Organisationsbaustein aus einen Ersatzwert in den Akkumulator 1 (AKKU 1) ein.

Hinweis

Die Anweisung "Ersatzwert eintragen" dürfen Sie nur in Synchronfehler-Organisationsbausteinen (OB 121, OB 122) aufrufen.

Die Anweisung "Ersatzwert eintragen" verwenden Sie, wenn Sie von einer Baugruppe keinen Wert mehr lesen können. Nach jedem Zugriff auf diese Baugruppe wird der OB 122 gestartet. Mit dem Aufruf der Anweisung "Ersatzwert eintragen" können Sie bewirken, dass ein Ersatzwert in den Akkumulator 1 geladen wird. Die Programmbearbeitung wird dann mit diesem Ersatzwert fortgesetzt. Die für die Auswahl des Ersatzwerts notwendigen Informationen können Sie den lokalen Variablen des OB 122 entnehmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ersatzwert eintragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VAL	Input	DWORD	E, A, M, D, L oder Konstante	Ersatzwert
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Die Anweisung wurde nicht von einem Synchronfehler-OB aus aufgerufen.
8xyy	Weitere Fehlerinformationen finden Sie hier: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter

"Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zuletzt programmierten Schritts.
EVENT(i), 1<= i <=16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter erster Schritt des Schrittschaltwerks (1 bis 16)
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Verbleibende Bearbeitungszeit für den aktuellen Schritt
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Systemzeit des vorherigen Aufrufs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL DRUM, "DRUM_DB"	// Die Anweisung wird aufgerufen und der Instanz-Datenbaustein "DRUM_DB" angelegt.
RESET := "Tag_Reset"	// Rücksetzeingang
JOG := "Tag_Input_Jog"	// Bei einer positiven Flanke geht die Anweisung zum nächsten Schritt über.
DRUM_EN := "Tag_Input_DrumEN"	// Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.

AWL	Erläuterung
LST_STEP := "Tag_Number_Last-Step"	// Nummer des zuletzt programmierten Schritts
EVENT1 := "MyTag_Event_1"	// Ereignisbit 1
EVENT2 := "MyTag_Event_2"	// Ereignisbit 2
EVENT3 := "MyTag_Event_3"	// Ereignisbit 3
EVENT4 := "MyTag_Event_4"	// Ereignisbit 4
EVENT5 := "MyTag_Event_5"	// Ereignisbit 5
EVENT6 := "MyTag_Event_6"	// Ereignisbit 6
EVENT7 := "MyTag_Event_7"	// Ereignisbit 7
EVENT8 := "MyTag_Event_8"	// Ereignisbit 8
EVENT9 := "MyTag_Event_9"	// Ereignisbit 9
EVENT10 := "MyTag_Event_10"	// Ereignisbit 10
EVENT11 := "MyTag_Event_11"	// Ereignisbit 11
EVENT12 := "MyTag_Event_12"	// Ereignisbit 12
EVENT13 := "MyTag_Event_13"	// Ereignisbit 13
EVENT14 := "MyTag_Event_14"	// Ereignisbit 14
EVENT15 := "MyTag_Event_15"	// Ereignisbit 15
EVENT16 := "MyTag_Event_16"	// Ereignisbit 16
OUT1 := "MyTag_Output_1"	// Ausgabebit 1
OUT2 := "MyTag_Output_2"	// Ausgabebit 2
OUT3 := "MyTag_Output_3"	// Ausgabebit 3
OUT4 := "MyTag_Output_4"	// Ausgabebit 4
OUT5 := "MyTag_Output_5"	// Ausgabebit 5
OUT6 := "MyTag_Output_6"	// Ausgabebit 6
OUT7 := "MyTag_Output_7"	// Ausgabebit 7
OUT8 := "MyTag_Output_8"	// Ausgabebit 8
OUT9 := "MyTag_Output_9"	// Ausgabebit 9
OUT10 := "MyTag_Output_10"	// Ausgabebit 10
OUT11 := "MyTag_Output_11"	// Ausgabebit 11
OUT12 := "MyTag_Output_12"	// Ausgabebit 12
OUT13 := "MyTag_Output_13"	// Ausgabebit 13
OUT14 := "MyTag_Output_14"	// Ausgabebit 14
OUT15 := "MyTag_Output_15"	// Ausgabebit 15
OUT16 := "MyTag_Output_16"	// Ausgabebit 16
Q := "Tag_Output_Q"	// Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD := "Tag_OutputWord"	// Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE := "Tag_ErrorCode"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE

Parameter	Adresse	Wert
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE

Parameter	Operand	Adresse	Wert
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE

Parameter	Adresse	Wert
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

DRUM_X: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" realisieren Sie ein Schrittschaltwerk mit maximal 16 Schritten. Die Nummer des ersten Schritts legen Sie am Parameter DSP fest. Die Nummer des letzten Schritts geben Sie am Parameter LST_STEP vor.

In jedem Schritt werden alle 16 Ausgangsbits OUT0 bis OUT15 und der Ausgangsparameter OUT_WORD (in dem die Ausgangsbits zusammengefasst sind) beschrieben. Ein Ausgangsbit bekommt entweder das zugehörige Bit des von Ihnen vorgegebenen Felds OUT_VAL oder das Ausgangsbit des vorangegangenen Schritts zugewiesen. Welcher Wert zugewiesen wird, hängt von der Belegung der Maskenbits am Parameter S_MASK ab.

Hinweis

Die Voreinstellung der Maskenbits ist 0. Falls Sie die aktuelle Einstellung eines oder mehrerer Maskenbits ändern wollen, führen Sie diese Änderung im Instanz-Datenbaustein durch.

Die Anweisung "Schrittschaltwerk realisieren" schaltet in den nächsten Schritt, wenn am Parameter JOG eine positive Signalflanke aufgetreten ist. Falls sich die Anweisung bereits im letzten Schritt befindet, werden bei einer positiven Signalflanke am Parameter JOG die Variablen Q und EOD gesetzt. Der Parameter DCC wird in diesem Fall auf den Wert 0 gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" gesetzt wird.

Darüber hinaus können Sie auch ein zeitabhängiges Weiterschalten in den nächsten Schritt zulassen. Dazu muss der Parameter DRUM_EN auf den Wert "1" gesetzt werden. Der Übergang in den nächsten Schritt erfolgt, wenn das zum aktuellen Schritt zugehörige Ereignisbit (EVENTi) gesetzt ist und die für den aktuellen Schritt vorgegebene Zeit abgelaufen ist. Die Zeit für einen Schritt ergibt sich aus dem Produkt von der Zeitbasis (DTBP) und dem für den aktuellen Schritt gültigen Zeitfaktor (S_PRESET).

Hinweis

Die im aktuellen Schritt noch verbleibende Bearbeitungszeit (DCC) wird nur dann reduziert, wenn das zugehörige Ereignisbit (EVENTi) gesetzt ist.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Wenn beim Aufruf der Anweisung der Wert "1" am Parameter RESET anliegt, geht das Schrittschaltwerk in den Schritt, dessen Nummer am Parameter DSP angegeben ist.

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist.

Zusätzlich können Sie das Schrittschaltwerk jederzeit über den Wert am Parameter JOG weiterschalten.

Beim Erstaufruf der Anweisung müssen Sie den Parameter RESET mit "1" belegen.

Wenn sich das Schaltwerk im letzten Schritt befindet und die für diesen Schritt vorgegebene Bearbeitungszeit abgelaufen ist, werden die Parameter Q und EOD gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" wechselt.

Die Anweisung wird nur in den Betriebszuständen ANLAUF und RUN ausgeführt.

Das Betriebssystem setzt die Anweisung "Schrittschaltwerk realisieren" bei einem Kaltstart zurück. Bei einem Neustart (Warmstart) wird die Anweisung nicht zurückgesetzt. Falls die Anweisung "Schrittschaltwerk realisieren" nach einem Neustart (Warmstart) initialisiert werden soll, müssen Sie die Anweisung im OB 100 mit dem Wert "1" am Parameter RESET aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" bewirkt das Rücksetzen des Schaltwerks. Beim Erstaufruf der Anweisung muss der Parameter RESET auf "1" gesetzt werden.
JOG	Input	BOOL	E, A, M, D, L	Eine positive Flanke (gegenüber dem letzten Aufruf) bewirkt das Weiterschalten in den nächsten Schritt, falls sich das Schaltwerk noch nicht im letzten Schritt befindet. Die Weiterschaltung erfolgt unabhängig davon, welchen Wert Sie am Parameter DRUM_EN angegeben haben.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DRUM_EN	Input	BOOL	E, A, M, D, L	Steuerparameter, der festlegt, ob ein zeitabhängiges Weiterschalten in den nächsten Schritt möglich ist. Bei einem Wert "1" ist ein zeitabhängiges Weiterschalten möglich.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des letzten Schritts Zulässige Werte sind 1 bis 16.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i) (gehört zum Schritt i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	E, A, M, D, L	Ausgangsbit (j) (identisch mit dem Bit-Nr. j von OUT_WORD)
Q	Output	BOOL	E, A, M, D, L	Zustandsparameter, der angibt, ob die von Ihnen vorgegebene Bearbeitungszeit des letzten Schritts abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Ausgangsbits zusammengefasst in einer Variablen
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Nicht anwenderrelevant (Eingangsparameter JOG des vorhergehenden Aufrufs)
EOD	Static	BOOL	E, A, M, D, L	Identisch zum Ausgangsparameter Q
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des ersten Schritts Zulässige Werte sind 1 bis 16.
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des aktuellen Schritts
DCC	Static	DWORD	E, A, M, D, L, P	Die im aktuellen Schritt noch verbleibende Bearbeitungszeit in Millisekunden (ms) (nur relevant, falls DRUM_EN = 1 und das zugehörige Ereignisbit = 1)
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Die für alle Schritte gültige Zeitbasis in Millisekunden (ms)
PREV_TIME	Static	DWORD	E, A, M, D, L oder Konstante	(Nicht anwenderrelevant: Systemzeit des vorhergehenden Aufrufs)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_PRESET	Static	ARRAY of WORD	E, A, M, D, L oder Konstante	Eindimensionales Feld mit den Zeitfaktoren für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16]. In diesem Fall enthält S_PRESET [x] den Zeitfaktor des Schritts x.
OUT_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den in jedem Schritt auszugebenden Werten, falls Sie diese nicht über S_MASK ausgeblendet haben. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält OUT_VAL [x, y] den Wert, der dem Ausgangsbit OUTy im Schritt x zugewiesen wird.
S_MASK	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den Maskenbits für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält S_MASK [x, y] das Maskenbit für den "y"-ten auszugebenden Wert im Schritt x. Bedeutung der Maskenbits: • 0: der entsprechende Wert des vorangegangenen Schritts wird dem zugehörigen Ausgangsbit zugewiesen • 1: der entsprechende Wert aus OUT_VAL wird dem zugehörigen Ausgangsbit zugewiesen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE* (W#16#...)	Erläuterung
0000	Kein Fehler
8081	Unzulässiger Wert am Parameter LST_STEP
8082	Unzulässiger Wert am DSC
8083	Unzulässiger Wert am DSP
8084	Das Produkt DCC = DTBP * S_PRESET[DSC] überschreitet den Wert 2**31-1 (ca. 24,86 Tage).
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT0 bis OUT15) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL DRUM_X, "DRUM_X_DB"	// Die Anweisung wird aufgerufen und der Instanz-Datenbaustein "DRUM_DB" angelegt.
RESET := "Tag_Reset"	// Rücksetzeingang
JOG := "Tag_Input_Jog"	// Bei einer positiven Flanke geht die Anweisung zum nächsten Schritt über.
DRUM_EN := "Tag_Input_DrumEN"	// Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP := "Tag_Number_Last-Step"	// Nummer des zuletzt programmierten Schritts
EVENT1 := "MyTag_Event_1"	// Ereignisbit 1
EVENT2 := "MyTag_Event_2"	// Ereignisbit 2
EVENT3 := "MyTag_Event_3"	// Ereignisbit 3
EVENT4 := "MyTag_Event_4"	// Ereignisbit 4
EVENT5 := "MyTag_Event_5"	// Ereignisbit 5
EVENT6 := "MyTag_Event_6"	// Ereignisbit 6
EVENT7 := "MyTag_Event_7"	// Ereignisbit 7
EVENT8 := "MyTag_Event_8"	// Ereignisbit 8
EVENT9 := "MyTag_Event_9"	// Ereignisbit 9
EVENT10 := "MyTag_Event_10"	// Ereignisbit 10

```
AWL                              Erläuterung
EVENT11 := "MyTag_Event_11"      // Ereignisbit 11
EVENT12 := "MyTag_Event_12"      // Ereignisbit 12
EVENT13 := "MyTag_Event_13"      // Ereignisbit 13
EVENT14 := "MyTag_Event_14"      // Ereignisbit 14
EVENT15 := "MyTag_Event_15"      // Ereignisbit 15
EVENT16 := "MyTag_Event_16"      // Ereignisbit 16
OUT0 := "MyTag_Output_0"         // Ausgabebit 0
OUT1 := "MyTag_Output_1"         // Ausgabebit 1
OUT2 := "MyTag_Output_2"         // Ausgabebit 2
OUT3 := "MyTag_Output_3"         // Ausgabebit 3
OUT4 := "MyTag_Output_4"         // Ausgabebit 4
OUT5 := "MyTag_Output_5"         // Ausgabebit 5
OUT6 := "MyTag_Output_6"         // Ausgabebit 6
OUT7 := "MyTag_Output_7"         // Ausgabebit 7
OUT8 := "MyTag_Output_8"         // Ausgabebit 8
OUT9 := "MyTag_Output_9"         // Ausgabebit 9
OUT10 := "MyTag_Output_10"       // Ausgabebit 10
OUT11 := "MyTag_Output_11"       // Ausgabebit 11
OUT12 := "MyTag_Output_12"       // Ausgabebit 12
OUT13 := "MyTag_Output_13"       // Ausgabebit 13
OUT14 := "MyTag_Output_14"       // Ausgabebit 14
OUT15 := "MyTag_Output_15"       // Ausgabebit 15
Q := "Tag_Output_Q"              // Der Signalzustand "1" zeigt an, dass die Zeit für
                                 den letzten Schritt abgelaufen ist.
OUT_WORD := "Tag_OutputWord"     // Wortadresse, in die das Schrittschaltwerk die Aus-
                                 gabewerte schreibt.
ERR_CODE := "Tag_ErrorCode"      // Fehlerinformation
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE

Parameter	Operand	Adresse	Wert
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE

Parameter	Adresse	Wert
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	TRUE
OUT2	MyTag_Output_2	M4.2	TRUE
OUT3	MyTag_Output_3	M4.3	TRUE
OUT4	MyTag_Output_4	M4.4	TRUE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	TRUE
OUT7	MyTag_Output_7	M4.7	TRUE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	TRUE
OUT11	MyTag_Output_11	M5.3	TRUE
OUT12	MyTag_Output_12	M5.4	TRUE
OUT13	MyTag_Output_13	M5.5	TRUE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	FALSE
OUT2	MyTag_Output_2	M4.2	FALSE
OUT3	MyTag_Output_3	M4.3	FALSE
OUT4	MyTag_Output_4	M4.4	FALSE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	FALSE
OUT7	MyTag_Output_7	M4.7	FALSE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	FALSE
OUT11	MyTag_Output_11	M5.3	FALSE
OUT12	MyTag_Output_12	M5.4	FALSE
OUT13	MyTag_Output_13	M5.5	FALSE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Wirkungen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET, OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET, OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS der Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zu CMD

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

```
AWL                              Erläuterung
CALL DCAT, "DCAT_DB"             // Die Anweisung wird aufgerufen und der Instanz-Da-
                                 tenbaustein "DCAT_DB" angelegt.
CMD := "Tag_Input_CMD"           // Öffnen oder Schließen des Geräts
O_FB := "Tag_Input_O_FB"         // Feedback beim Öffnen
C_FB := "Tag_Input_C_FB"         // Feedback beim Schließen
Q := "Tag_Output_Q"              // Zeigt den Status des Parameters CMD
OA := "Tag_Output_OA"            // Alarmausgang beim Öffnen
CA := "Tag_Output_CA"            // Alarmausgang beim Schließen
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen

Eingangsparameter								Ausgangsparameter								
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT

Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Schließen"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

```
AWL                              Erläuterung
CALL MCAT, "MCAT_DB"             // Die Anweisung wird aufgerufen und der Instanz-Da-
                                 tenbaustein "MCAT_DB" angelegt.
O_CMD := "Tag_Input_O_CMD"       // Befehlseingabe "Öffnen"
C_CMD := "Tag_Input_C_CMD"       // Befehlseingabe "Schließen"
S_CMD := "Tag_Input_S_CMD"       // Befehlseingabe "Stoppen"
O_FB := "Tag_Input_O_FB"         // Feedback beim Öffnen
C_FB := "Tag_Input_C_FB"         // Feedback beim Schließen
OO := "Tag_OutputOpen"           // Ausgang "Öffnen"
CO := "Tag_OutputClosed"         // Ausgang "Schließen"
OA := "Tag_Output_OA"            // Alarmausgang beim Öffnen
CA := "Tag_Output_CA"            // Alarmausgang beim Schließen
Q := "Tag_Output_Q"              // Fehlerinformation
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für Schritt 2 verglichen. Der Signalzustand des Parameters OUT wird auf TRUE gesetzt, da die Eingabebits mit der Maske für Schritt 2 übereinstimmen.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
`CALL IMC, "IMC_DB"`	`// Die Anweisung wird aufgerufen und der Instanz-Datenbaustein "IMC_DB" angelegt.`

```
AWL                                Erläuterung
IN_BIT0 := "Tag_Input_BIT0"    // Eingangsbit 0
IN_BIT1 := "Tag_Input_BIT1"    // Eingangsbit 1
IN_BIT2 := "Tag_Input_BIT2"    // Eingangsbit 2
IN_BIT3 := "Tag_Input_BIT3"    // Eingangsbit 3
IN_BIT4 := "Tag_Input_BIT4"    // Eingangsbit 4
IN_BIT5 := "Tag_Input_BIT5"    // Eingangsbit 5
IN_BIT6 := "Tag_Input_BIT6"    // Eingangsbit 6
IN_BIT7 := "Tag_Input_BIT7"    // Eingangsbit 7
IN_BIT8 := "Tag_Input_BIT8"    // Eingangsbit 8
IN_BIT9 := "Tag_Input_BIT9"    // Eingangsbit 9
IN_BIT10 := "Tag_Input_BIT10"  // Eingangsbit 10
IN_BIT11 := "Tag_Input_BIT11"  // Eingangsbit 11
IN_BIT12 := "Tag_Input_BIT12"  // Eingangsbit 12
IN_BIT13 := "Tag_Input_BIT13"  // Eingangsbit 13
IN_BIT14 := "Tag_Input_BIT14"  // Eingangsbit 14
IN_BIT15 := "Tag_Input_BIT15"  // Eingangsbit 15
CMP_STEP := "Tag_CMP_STEP"     // Schrittnummer der Maske
OUT := "Tag_Output"            // Zeigt an, ob eine Entsprechung gefunden wurde
ERR_CODE := "Tag_ErrorCode"    // Fehlerinformation
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02

Parameter	Operand	Wert
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "IMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY of WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Im folgenden Beispiel werden alle 16 Eingabebits mit der Maske für die Schritte 0 bis 5 verglichen, bis eine Entsprechung gefunden wird. Dabei werden nur die Masken für die Schritte 0 bis 2 abgefragt, weil die Maske für Schritt 2 dem Eingabebit entspricht.

Hinweis

Statische Parameter können Sie im Baustein initialisieren.

AWL	Erläuterung
CALL SMC, "SMC_DB"	// Die Anweisung wird aufgerufen und der Instanz-Datenbaustein "SMC_DB" angelegt.
IN_BIT0 := "Tag_Input_BIT0"	// Eingangsbit 0
IN_BIT1 := "Tag_Input_BIT1"	// Eingangsbit 1
IN_BIT2 := "Tag_Input_BIT2"	// Eingangsbit 2
IN_BIT3 := "Tag_Input_BIT3"	// Eingangsbit 3
IN_BIT4 := "Tag_Input_BIT4"	// Eingangsbit 4
IN_BIT5 := "Tag_Input_BIT5"	// Eingangsbit 5
IN_BIT6 := "Tag_Input_BIT6"	// Eingangsbit 6
IN_BIT7 := "Tag_Input_BIT7"	// Eingangsbit 7
IN_BIT8 := "Tag_Input_BIT8"	// Eingangsbit 8
IN_BIT9 := "Tag_Input_BIT9"	// Eingangsbit 9
IN_BIT10 := "Tag_Input_BIT10"	// Eingangsbit 10
IN_BIT11 := "Tag_Input_BIT11"	// Eingangsbit 11
IN_BIT12 := "Tag_Input_BIT12"	// Eingangsbit 12
IN_BIT13 := "Tag_Input_BIT13"	// Eingangsbit 13
IN_BIT14 := "Tag_Input_BIT14"	// Eingangsbit 14
IN_BIT15 := "Tag_Input_BIT15"	// Eingangsbit 15
OUT := "Tag_Output"	// Zeigt an, ob eine Entsprechung gefunden wurde.
OUT_STEP := "Tag_Output_STEP"	// Enthält die Schrittnummer mit der entsprechenden Maske.
ERR_CODE := "Tag_ErrorCode"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

Im Instanz-Datenbaustein "SMC_DB" der Anweisung sind die folgenden Werte für die Maske des Schritts 2 gespeichert:

Parameter	Adresse	Wert
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE

Parameter	Adresse	Wert
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output	TRUE
OUT_STEP	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT,

sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

AWL	Erläuterung
CALL LEAD_LAG, "LEAD_LAG_DB"	// Die Anweisung wird aufgerufen und der Instanz-Datenbaustein "LEAD_LAG_DB" angelegt.
IN := "Tag_Input"	// Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T := "Tag_Input_SAMPE_T"	// Abtastzeit
OUT := "Tag_Output_Result"	// Ergebnis der Anweisung
ERR_CODE := "Tag_ErrorCode"	// Fehlerinformation

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0

Parameter	Adresse	Wert
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

Master Control Relay (Seite 201)

TONR_X: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines festgelegten Zeitraums, den Sie mithilfe des Eingangs PV angeben. Die Anweisung wird ausgeführt, wenn der Signalzustand am Starteingang TMR_EN von "0" nach "1" wechselt. Gleichzeitig läuft die Zeit mit der am Eingang PV programmierten Zeitdauer ab. Während die Anweisung ausgeführt wird, addiert sie den Zeitwert des Eingangs DELTA_T zum Zeitwert des InOut ET.

Folgende Bedingungen müssen dazu erfüllt sein:

- Der Starteingang TMR_EN liefert den Signalzustand "1".
- Der Zeitwert am InOut ET ist kleiner als der Zeitwert am Parameter PV.
- Der Eingang RESET liefert den Signalzustand "0".

Die Anweisung wird unterbrochen, wenn der Signalzustand am Starteingang TMR_EN von "1" nach "0" wechselt. Sobald der Signalzustand wieder nach "1" wechselt, läuft die Anweisung weiter.

Die Anweisung wird beendet, wenn der Zeitwert am InOut ET den Zeitwert am Eingang PV erreicht hat. Dann liefert der Ausgang Q den Signalzustand "1".

Solange der Ausgang Q den Signalzustand "1" liefert, bleibt der Zeitwert am InOut ET unverändert. Der Zeitwert und der Ausgang Q werden erst wieder auf "0" zurückgesetzt, wenn der Eingang RESET auf den Signalzustand "1" gesetzt wird.

Da die Anweisung "Zeit akkumulieren" die Zykluszeit (DELTA_T) des letzten Zyklus des Organisationsbausteins (OB) verwendet, darf sie nur in zyklischen Organisationsbausteinen aufgerufen werden.

Hinweis

Sie müssen die Zykluszeit des Organisationsbausteins aus den Anlaufinformationen in die globale Variable des Parameters DELTA_T verschieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TMR_EN (Timer Enable)	Input	BOOL	E, A, M, D, L	Starteingang Startet die Anweisung und den Ablauf der voreingestellten Zeitdauer am Eingang PV
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
PV (Preset Value)	Input	DINT	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung
DELTA_T	Input	INT	E, A, M, D, L oder Konstante	Zykluszeit des Organisationsbausteins des vorherigen Zyklus
Q	Output	BOOL	E, A, M, D, L	Ausgang, der bei Erreichen des voreingestellten Zeitwerts gesetzt wird
ET (Elapsed Time)	InOut	DINT	E, A, M, D, L	Akkumulierter Zeitwert (Aktuell abgelaufener Zeitwert + Zeitwert des Eingangs DELTA_T)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeitwert akkumulieren":

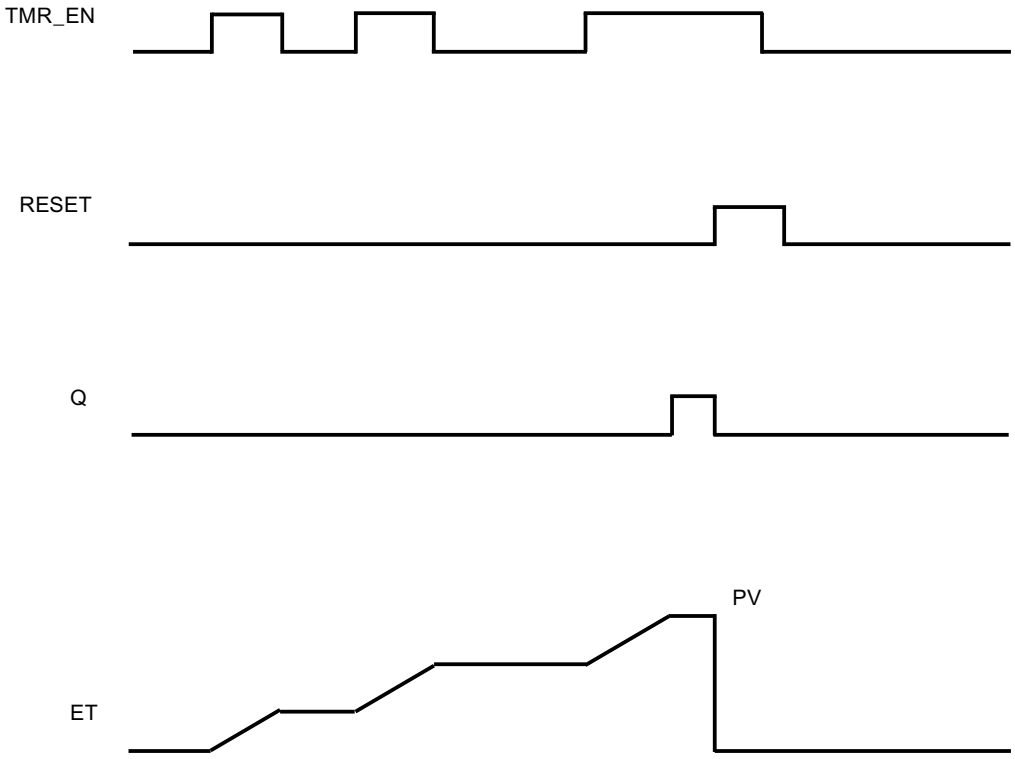

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
CALL TONR_X	// Die Anweisung wird aufgerufen.
TMR_EN := "Tag_Start"	// Starteingang
RESET := "Tag_Reset"	// Rücksetzeingang
PV := 10	// Maximale Dauer der Zeiterfassung
DELTA_T := 2	// Zykluszeit des Organisationsbausteins des vorherigen Zyklus
ET := "Tag_Elapsed_Time"	// Akkumulierte Zeit
Q := "Tag_Output"	// Zeigt an, ob die Zeit des Parameters PV erreicht ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

WSR: Datum im Schieberegister speichern

Beschreibung

Mit der Anweisung "Datum im Schieberegister speichern" schreiben Sie Daten aus einer angegebenen Quelle in ein Schieberegister. Dabei wird jedes Element des Registers in die nächste Adresse verschoben. Die Daten, die in der letzten Adresse des Schieberegisters enthalten sind, gehen nach der Schiebeoperation verloren.

Bei jeder Ausführung der Anweisung mit dem Wert "0" am Parameter RESET werden neue Daten aus der Quelle (S_DATA) gelesen und in die Anfangsadresse (START) des Schieberegisters geschoben. Der Parameter LEN gibt die Anzahl der zu schiebenden Elemente an. Am Parameter E_TYPE wird der Datentyp der Elemente festgelegt. Wenn am Parameter E_TYPE ein unzulässiger Datentyp angegeben wird, wird die Anweisung nicht bearbeitet.

Wenn der Parameter RESET bei der Ausführung der Anweisung auf "1" gesetzt ist, wird das Register gelöscht. Der Parameter Q wird auf den Signalzustand "1" gesetzt, wenn das Schieberegister mit Nullen belegt ist oder gelöscht wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Datum im Schieberegister speichern":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_DATA	Input	POINTER	E, A, M, D	Zeigt auf das Quellelement, das in die Anfangsadresse (START) geschoben wird. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
START	Input	POINTER	E, A, M, D	Zeigt auf die Anfangsadresse des Registers. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
LEN	Input	WORD	E, A, M, D, L, P	Anzahl der Elemente, die geschoben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der zu schiebenden Elemente an. Folgende Datentypen sind zulässig: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	A, M, D, L	Gibt "1" an, wenn der Parameter RESET aktiv (1) ist oder alle Elemente, die geschoben werden, den Wert "0" haben.

Beispiel

Im folgenden Beispiel werden vier Elemente des Datentyps WORD ab der am Parameter START angegebenen Adresse geschoben. Nachdem das erste Element des Registers in die nächste Adresse geschoben wurde, wird die erste Adresse mit den Daten belegt, auf die der Zeiger am Parameter S_DATA zeigt. Der Wert des letzten Elements im Register geht dabei verloren.

```
AWL                                Erläuterung
CALL WSR                           // Die Anweisung wird aufgerufen.
RESET := "Tag_Input_Reset"         // Rücksetzeingang
S_DATA := "Tag_Input_Data-         // Quellelement, das in die Anfangsadresse geschoben
Source"                            wird.
START := "Tag_Input_StartAdd-      // Anfangsadresse
ress"
LEN := "Tag_Input_LEN"             // Anzahl der zu schiebenden Elemente
E_TYPE := "Tag_Input_Element-      // Datentyp der Elemente
Type"
Q := "Tag_Output_Q"                // Status des Registers
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_SourceData	P#DB2.DBX6.0 DW6 = W#16#1234

Parameter	Operand	Wert
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
LEN	Tag_Input_LEN	W#16#04
E_TYPE	Tag_Input_ElementType	W#16#04
Q	Tag_Output_Q	1

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_SourceData	P#DB2.DBX6.0 DW6 = W#16#1234
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
LEN	Tag_Input_LEN	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SHRB: Bit in Bitschieberegister schieben

Beschreibung

Mit der Anweisung "Bit in Bitschieberegister schieben" schreiben Sie ein Bit aus einer angegebenen Quelle (DATA) in ein Schieberegister. Bei jeder Ausführung der Anweisung mit einem Wert "0" am Parameter RESET werden neue Daten aus der Quelle gelesen und in die Anfangsadresse (S_BIT) des Schieberegisters geschoben. Alle folgenden Bits werden um eins verschoben. Das Bit in der letzten Adresse (S_BIT + N) geht nach dem Schieben verloren. Wenn der Parameter RESET während der Bearbeitung der Anweisung auf "1" gesetzt ist, werden die Adressen im Schieberegister auf "0" zurückgesetzt und nicht geschoben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bit in Bitschieberegister schieben":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	Input	BOOL	E, A, M, D, L	Quellbit
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das Anfangsbit des Schieberegisters. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	WORD	E, A, M, D, L, P	Anzahl der Bits, die geschoben werden.

Beispiel

Im folgenden Beispiel wird der Parameter N auf den Wert 14 ("E" in hexadezimaler Schreibweise) gesetzt, sodass 14 Bits ab der am Parameter S_BIT angegebenen Adresse geschoben werden. Nachdem die Bits geschoben worden sind, wird die erste Adresse mit den Daten aus der Quelle am Parameter DATA belegt. Der letzte Bitwert geht verloren.

```
AWL                              Erläuterung
CALL SHRB                        // Die Anweisung wird aufgerufen.
DATA := "Tag_Input_Data"         // Quellbit, das in die Anfangsadresse geschoben wird.
RESET := "Tag_Input_Reset"       // Rücksetzeingang
S_BIT := "Tag_Input_BitSour-     // Anfangsadresse
ce"
N := "Tag_Input_Number"          // Anzahl der zu schiebenden Bits
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Vor der Bearbeitung der Anweisung sind die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Nach der Bearbeitung

Nach der Bearbeitung der Anweisung werden die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

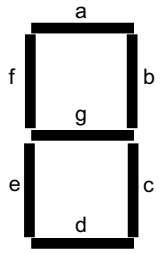

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL SEG                     // Die Anweisung wird aufgerufen.
IN := "Tag_Input"            // Quellwort
OUT := "Tag_Output"          // Bitmuster für die 7-Segment-Anzeige
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16#065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL BCDCPL                  // Die Anweisung wird aufgerufen.
IN := "Tag_Input"            // 7-stellige BCD-Zahl
ERR_CODE := "Tag_Output"     // Zehnerkomplement der angegebenen BCD-Zahl
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden.
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
CALL BITSUM                  // Die Anweisung wird aufgerufen.
IN := "Tag_Input"            // Operand, dessen gesetzte Bit gezählt werden.
```

AWL	Erläuterung
RET_VAL := "Tag_Output"	// Anzahl der gesetzten Bit des Operanden "Tag_Input"

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits (Seite 185)

Grundlagen zu AWL (Seite 7557)

AWL Mnemonik

Bitverknüpfungen

U: UND-Verknüpfung

Beschreibung

Mit der Anweisung "UND-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach UND. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden.

Nach der Ausführung der Anweisung ist das VKE "1", wenn beide verknüpften Signalzustände auch "1" sind. Wenn einer der verknüpften Signalzustände "0" ist, ist das VKE nach der Ausführung der Anweisung "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, müssen alle Abfrageergebnisse "1" sein, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "UND-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

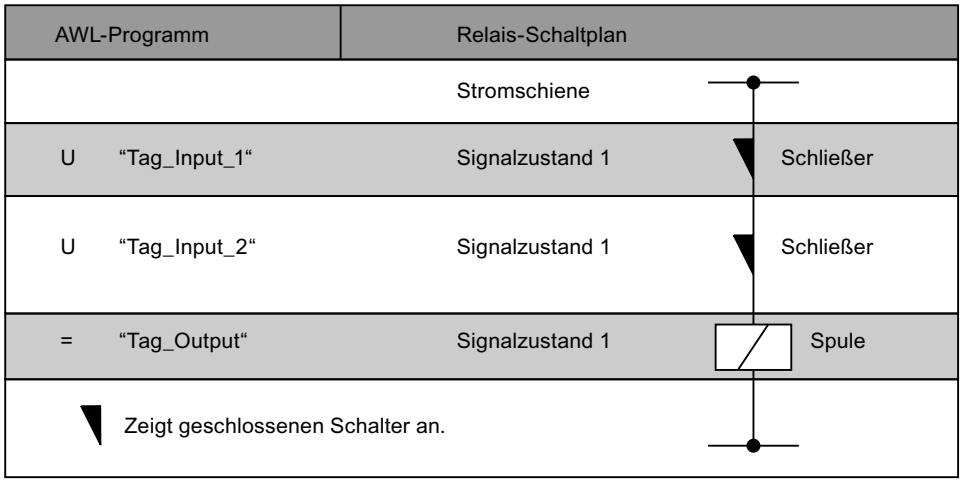

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

UN: Negierte UND-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte UND-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach UND. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Wenn der Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Wenn der Signalzustand des Operanden "1" ist, ist das Abfrageergebnis "0".

Nach der Ausführung der Anweisung ist das VKE "1", wenn das Abfrageergebnis und das VKE vor der Ausführung der Anweisung den Signalzustand "1" liefern.

Nach der Ausführung der Anweisung ist das VKE "0", wenn das Abfrageergebnis oder das VKE vor der Ausführung der Anweisung den Signalzustand "0" liefern.

Mit der Anweisung "Negierte UND-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte UND-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm		Relais-Schaltplan	
		Stromschiene	
U	“Tag_Input_1“	Signalzustand 0	Schließer
UN	“Tag_Input_2“	Signalzustand 1	Öffner
=	“Tag_Output“	Signalzustand 0	Spule

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

O: ODER-Verknüpfung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden.

Nach der Ausführung der Anweisung ist das VKE "1", wenn einer der verknüpften Signalzustände "1" ist. Wenn beide verknüpften Signalzustände "0" ist, ist das VKE nach der Ausführung der Anweisung "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Relais-Schaltplan
	Stromschiene
O “Tag_Input_1“	Signalzustand 1 Schließer (“Tag_Input_1“) / Signalzustand 0 Schließer (“Tag_Input_2“)
O “Tag_Input_2“	
= “Tag_Output“	Signalzustand 1 / Spule
	Zeigt geschlossenen Schalter an.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

ON: Negierte ODER-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Wenn der Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Wenn der Signalzustand des Operanden "1" ist, ist das Abfrageergebnis "0".

Nach der Ausführung der Anweisung ist das VKE "1", wenn das Abfrageergebnis oder das VKE vor der Ausführung der Anweisung den Signalzustand "1" liefert. Wenn beide verknüpften Signalzustände "0" sind, ist das VKE nach der Ausführung der Anweisung auch "0".

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame Verknüpfungsergebnis auch "1" ist.

Mit der Anweisung "Negierte ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

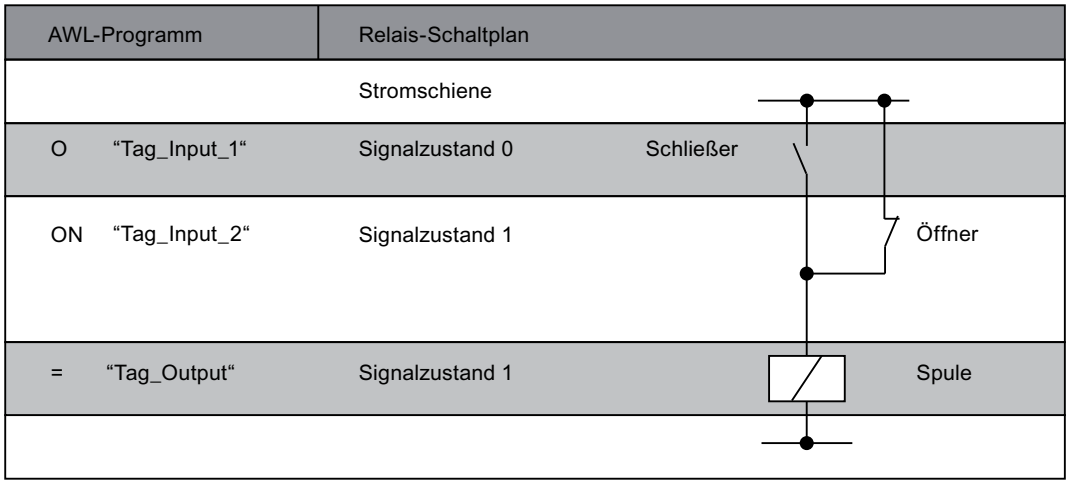

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

X: EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "1" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach EXKLUSIV ODER. Das Abfrageergebnis ist dabei identisch mit dem Signalzustand des abgefragten Operanden. Nach der Ausführung der Anweisung ist das VKE "1", wenn die miteinander verknüpften Signalzustände ungleich sind. Das VKE ist "0", wenn beide Signalzustände gleich sind.

Wenn z. B. das VKE vor der Ausführung der Anweisung "1" ist und der abgefragte Operand "1" liefert, wird das VKE auf "0" zurückgesetzt. Wenn das VKE "1" ist und der Operand den Signalzustand "0" liefert, wird das VKE nach der Ausführung der Anweisung auf "1" gesetzt.

Wenn Sie die Anweisung mehrfach nacheinander anwenden, genügt es, wenn eines der Abfrageergebnisse "1" ist, damit das gemeinsame VKE auch "1" ist. Wenn alle Abfrageergebnisse "1" oder "0" sind, wird das VKE auf "0" zurückgesetzt.

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Relais-Schaltplan
	Stromschiene
X “Tag_Input_1“	Kontakt
X “Tag_Input_2“	Kontakt
= “Tag_Output“	Spule

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

XN: Negierte EXKLUSIV ODER-Verknüpfung

Beschreibung

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung" fragen Sie den Signalzustand eines Binäroperanden auf "0" ab und verknüpfen das Abfrageergebnis mit dem Signalzustand des Verknüpfungsergebnisses (VKE) nach ODER. Das Abfrageergebnis ist dabei der negierte Signalzustand des abgefragten Operanden. Nach der Ausführung der Anweisung ist das VKE "1", wenn die miteinander verknüpften Signalzustände ungleich sind. Das VKE ist "0", wenn beide Signalzustände gleich sind.

Wenn z. B. der abgefragte Operand den Signalzustand "0" liefert, ist das Abfrageergebnis "1". Die Verknüpfung des Abfrageergebnisses mit einem VKE "1" nach EXKLUSIV ODER ergibt den Signalzustand "0". Die Verknüpfung mit einem VKE "0" ergibt den Signalzustand "1".

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung" können Sie auch den Signalzustand des Statusworts abfragen. Dazu geben Sie das entsprechende Statusbit (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BIE) als Parameter der Anweisung an.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	BOOL	E, A, M, D, L, T, Z	Operand, dessen Signalzustand abgefragt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm		Relais-Schaltplan	
		Stromschiene	
X	“Tag_Input_1“	Kontakt	
XN	“Tag_Input_2“	Kontakt	
=	“Tag_Output“	Spule	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

O: ODER-Verknüpfung von UND-Funktionen

Beschreibung

Mit der Anweisung "ODER-Verknüpfung von UND-Funktionen" bewirken Sie, dass mehrere UND-Verknüpfungen vor einer ODER-Verknüpfung ausführt werden. Das Gesamtergebnis der UND-Verknüpfungen wird gespeichert und mit dem aktuellen Verknüpfungsergebnis (VKE) sowie mit eventuellen weiteren Signalabfragen nach ODER verknüpft.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

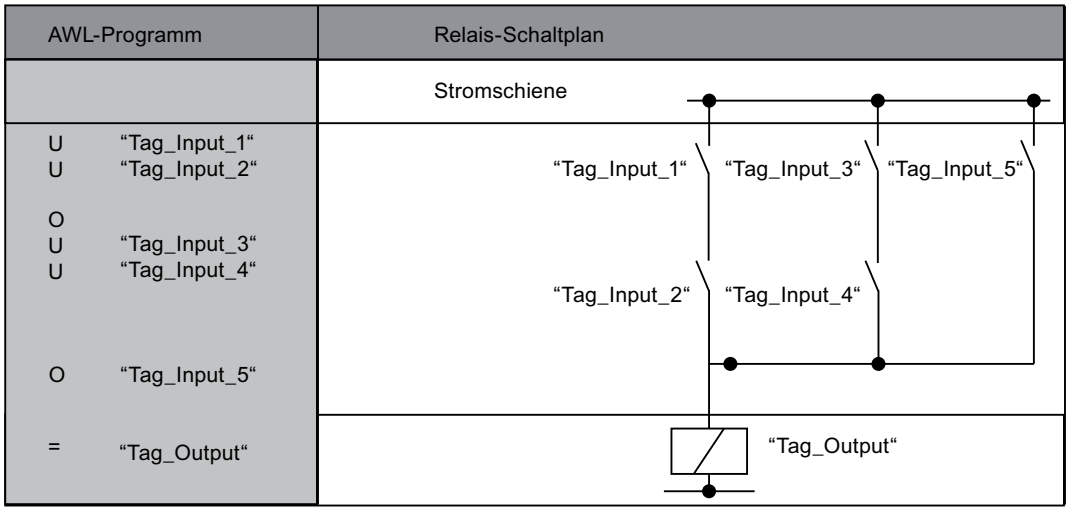

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

U(: UND-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "UND-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer UND-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Das gespeicherte VKE wird mit dem Verknüpfungsergebnis des gesamten Klammerausdrucks nach UND verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der UND-Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

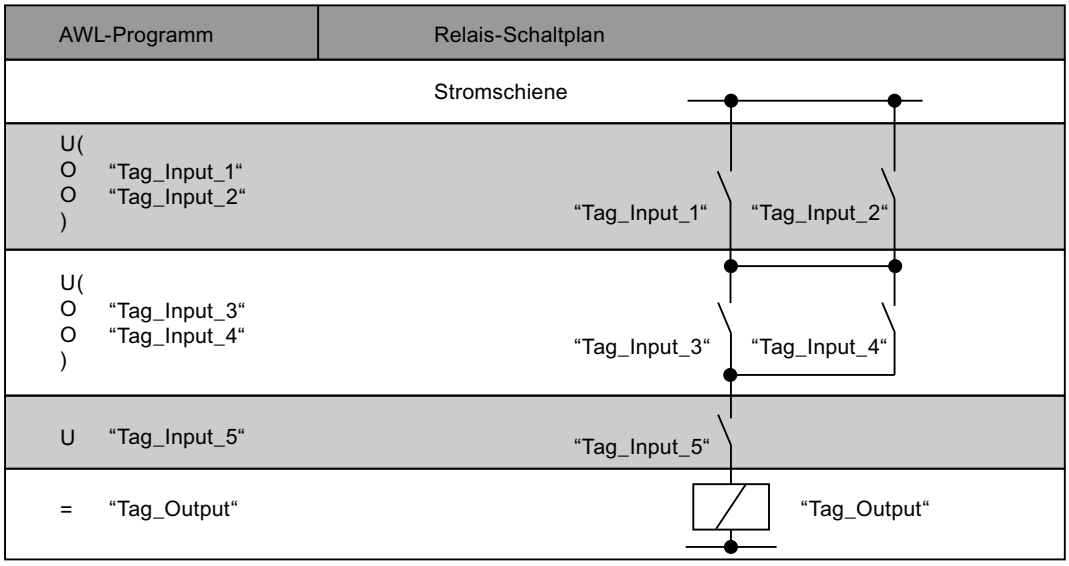

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

UN(: Negierte UND-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte UND-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer UND-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen wird das VKE des gesamten Klammerausdrucks negiert und mit dem gespeicherten VKE nach UND verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der UND-Anweisungen verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
UN(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach UND verknüpfen und das Gesamtergebnis negieren.
O "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach ODER verknüpfen.
O "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach ODER verknüpfen.
)	// Ende des Klammerausdrucks
UN "Tag_Input_3"	// Signalzustand des Operanden auf "0" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

O(: ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer ODER-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Das gespeicherte VKE wird nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen mit dem VKE des gesamten Klammerausdrucks nach ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der ODER-Anweisung verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
O(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach ODER verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ON(: Negierte ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer ODER-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Nach der Bearbeitung der Anweisungen im Klammerausdruck wird das VKE des Klammerausdrucks negiert und mit dem gespeicherten VKE nach ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der ODER-Anweisung verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ON(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach ODER verknüpfen und das Gesamtergebnis negieren.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer EXKLUSIV ODER-Verknüpfung bearbeitet werden.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Das gespeicherte VKE wird nach der Bearbeitung der Anweisungen im Klammerausdruck mit dem VKE des gesamten Klammerausdrucks nach EXKLUSIV ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der EXKLUSIV ODER-Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
X(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach EXKLUSIV ODER verknüpfen.
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
X "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung

Beschreibung

Mit der Anweisung "Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung" bewirken Sie, dass die Anweisungen eines Klammerausdrucks vor einer EXKLUSIV ODER-Verknüpfung bearbeitet werden und das Gesamtergebnis des Klammerausdrucks negiert wird.

Bei der Ausführung der Anweisung speichert die CPU das aktuelle Verknüpfungsergebnis (VKE) sowie das Binärergebnis BIE und das Statusbit OR. Nach der Bearbeitung der sich im Klammerausdruck befindlichen Anweisungen wird das VKE des Klammerausdrucks negiert und mit dem gespeicherten VKE nach EXKLUSIV ODER verknüpft. Wenn nach dem Klammerausdruck weitere Signalabfragen folgen, werden diese ebenfalls mit der EXKLUSIV ODER -Funktion verknüpft.

Klammerausdrücke können Sie auch schachteln, indem Sie innerhalb eines Klammerausdrucks eine Anweisung programmieren, die ebenfalls einen Klammerausdruck enthält. Die Schachtelungstiefe ist dabei auf 7 Ebenen beschränkt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
XN(	// Anfang des Klammerausdrucks
	// VKE des Klammerausdrucks nach EXKLUSIV ODER verknüpfen und das Gesamtergebnis negieren.
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
)	// Ende des Klammerausdrucks
XN "Tag_Input_3"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Signalzustand des VKE dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Abfragen der Statusbits mit Bitverknüpfungsanweisungen (Seite 193)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

): Verzweigung schließen

Beschreibung

Mit der Anweisung "Verzweigung schließen" legen Sie das Ende eines Klammerausdrucks fest. Nach der Ausführung der Anweisung wird das Gesamtergebnis des Klammerausdrucks nach der Anweisung verknüpft, die zum Öffnen des Klammerausdrucks verwendet wurde.

Zum Öffnen eines Klammerausdrucks können Sie die folgenden Anweisungen verwenden:

- U(: UND-Verknüpfung mit Verzweigung
- UN(: Negierte UND-Verknüpfung mit Verzweigung
- O(: ODER-Verknüpfung mit Verzweigung
- ON(: Negierte ODER-Verknüpfung mit Verzweigung
- X(: EXKLUSIV ODER-Verknüpfung mit Verzweigung
- XN(: Negierte EXKLUSIV ODER-Verknüpfung mit Verzweigung

Nach der Anweisung "Verzweigung schließen" setzt die CPU das Binärergebnis BIE auf den Signalzustand, den es vor dem Klammerausdruck hatte.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

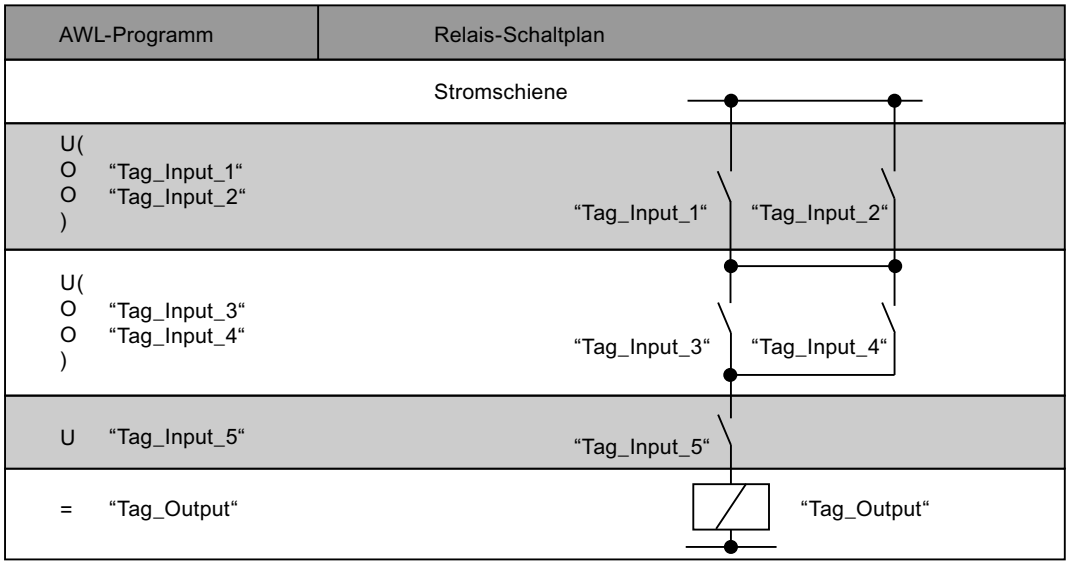

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Abfragen der Statusbits in AWL (Seite 193)

=: Zuweisung

Beschreibung

Mit der Anweisung "Zuweisung" weisen Sie einem angegebenen Operanden den Signalzustand des in der CPU gespeicherten Verknüpfungsergebnisses (VKE) zu. Wenn das VKE den Signalzustand "1" hat, wird der Operand gesetzt. Wenn der Signalzustand "0" ist, wird der Operand auf "0" zurückgesetzt.

Die Ausführung der Anweisung "Zuweisung" wird durch das Master Control Relay (MCR) beeinflusst. Wenn das MCR-Bit auf "0" gesetzt ist, wird der angegebene Operand auf den Signalzustand "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, dem das aktuelle VKE zugewiesen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

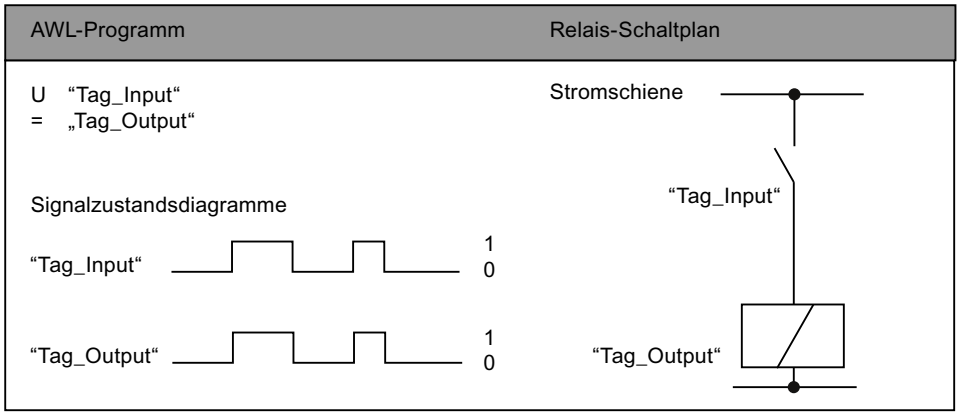

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

R: Rücksetzen

Beschreibung

Mit der Anweisung "Rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Nach der Ausführung der Anweisung wird der angegebene Operand auf "0" zurückgesetzt.

Bei einem aktuellen VKE von "0", bleibt der Signalzustand des angegebenen Operanden unverändert.

Die Ausführung der Anweisung "Rücksetzen" wird durch das Master Control Relay (MCR) beeinflusst. Wenn das MCR-Bit auf "0" gesetzt wird, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rücksetzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" zurückgesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input_1"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`U "Tag_Input_2"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`R "Tag_Output"`	`// Operanden auf "0" zurücksetzen, wenn das VKE = "1" ist.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

S: Setzen

Beschreibung

Mit der Anweisung "Setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Nach der Ausführung der Anweisung wird der angegebene Operand auf "1" gesetzt. Bei einem aktuellen VKE von "0", bleibt der Signalzustand des angegebenen Operanden unverändert.

Die Ausführung der Anweisung "Setzen" wird durch das Master Control Relay (MCR) beeinflusst. Wenn das MCR-Bit auf "0" gesetzt wird, bleibt der Signalzustand des angegebenen Operanden unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Setzen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Operand, der bei VKE = "1" gesetzt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
S "Tag_Output"	// Operanden auf "1" setzen, wenn das VKE = "1" ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

NOT: VKE invertieren

Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE). Die Anweisung "VKE invertieren" können Sie an beliebiger Stelle, auch innerhalb einer Verknüpfung, anwenden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
U "Tag_Input_1"              // Signalzustand des Operanden "Tag_Input_1" auf "1"
                             abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_2"              // Signalzustand des Operanden "Tag_Input_2" auf "1"
                             abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_3"              // Signalzustand des Operanden "Tag_Input_3" auf "1"
                             abfragen und mit dem aktuellen VKE nach UND verknüpfen.
NOT                          // Signalzustand des VKE invertieren
= "Tag_Output"               // Signalzustand des VKE dem Operanden "Tag_Output"
                             zuweisen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Input_1	1	1	1	0	0	0	1	0
Tag_Input_2	0	1	1	0	1	1	0	1
Tag_Input_3	0	0	1	0	0	1	1	1
Tag_Output	1	1	0	1	1	1	1	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SET: VKE auf 1 setzen

Beschreibung

Mit der Anweisung "VKE auf 1 setzen" setzen Sie das aktuelle Verknüpfungsergebnis (VKE) auf den Signalzustand "1".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

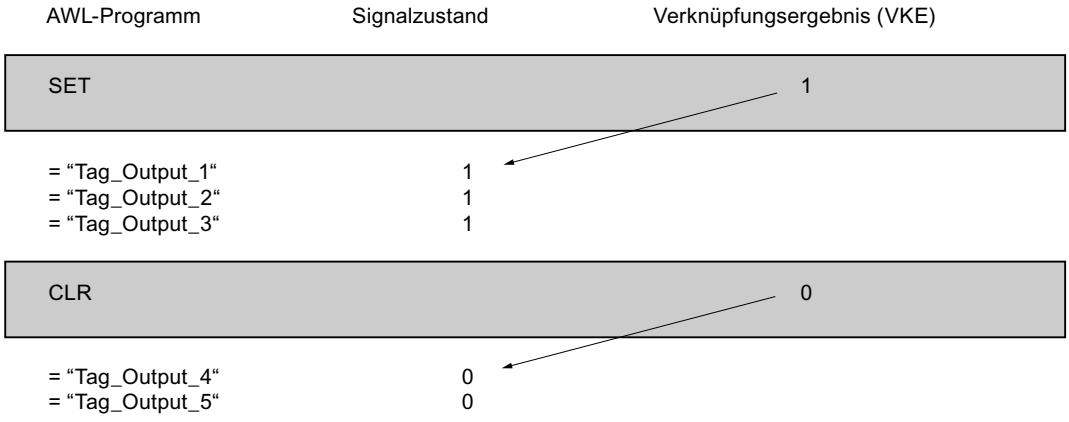

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

CLR: VKE auf 0 rücksetzen

Beschreibung

Mit der Anweisung "VKE auf 0 rücksetzen" setzen Sie das aktuelle Verknüpfungsergebnis (VKE) auf den Signalzustand "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL-Programm	Signalzustand	Verknüpfungsergebnis (VKE)
SET		1
= “Tag_Output_1“	1	
= “Tag_Output_2“	1	
= “Tag_Output_3“	1	
CLR		0
= “Tag_Output_4“	0	
= “Tag_Output_5“	0	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SAVE: VKE im BIE-Bit speichern

Beschreibung

Mit der Anweisung "VKE im BIE-Bit speichern" sichern Sie das Verknüpfungsergebnis (VKE) im Binärergebnis (BIE). In der Ausführung überträgt die Anweisung den Signalzustand des aktuellen Verknüpfungsergebnisses zum Statusbit BIE. Die Anweisung arbeitet unabhängig von Bedingungen und beeinflusst keine weiteren Statusbits.

Hinweis

Das BIE-Bit kann nach der Anwendung der Anweisung "VKE im BIE-Bit speichern" durch nachfolgende Anweisungen im gleichen oder in unterlagerten Bausteinen wieder verändert werden.

Sie können die Anweisung "VKE im BIE-Bit speichern" nutzen, um den Ausführungsstatus eines Bausteins zu kontrollieren. Wenn Sie z. B. die Anweisung "VKE im BIE-Bit speichern" am Ende eines Bausteins anwenden, wird das BIE-Bit auf den Signalzustand des aktuellen Verknüpfungsergebnisses des Bausteins gesetzt.

Das BIE-Bit hilft bei der Realisierung des EN-/ENO-Mechanismus bei Bausteinaufrufen. Weitere Informationen dazu finden sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand des Operanden "Tag_Input" auf "1" abfragen.
SAVE	// Signalzustand des VKE in das BIE-Bit übertragen.
BEB	// Baustein bei VKE = "1" beenden.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

Beispiel für den Nachbau des EN-/ENO-Mechanismus in AWL (Seite 162)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

FN: VKE auf negative Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand des Verknüpfungsergebnisses (VKE) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker <Operand> gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Das folgende Bild zeigt den Wechsel im VKE bei einer negativen und einer positiven Signalflanke:

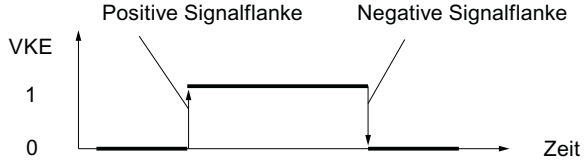

Die Abfrage der negativen Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine negative Signalflanke erfasst wird, wird das VKE nach der

Flankenauswertung einen Programmzyklus lang auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand des VKE "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Flankenmerker , in dem das VKE der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

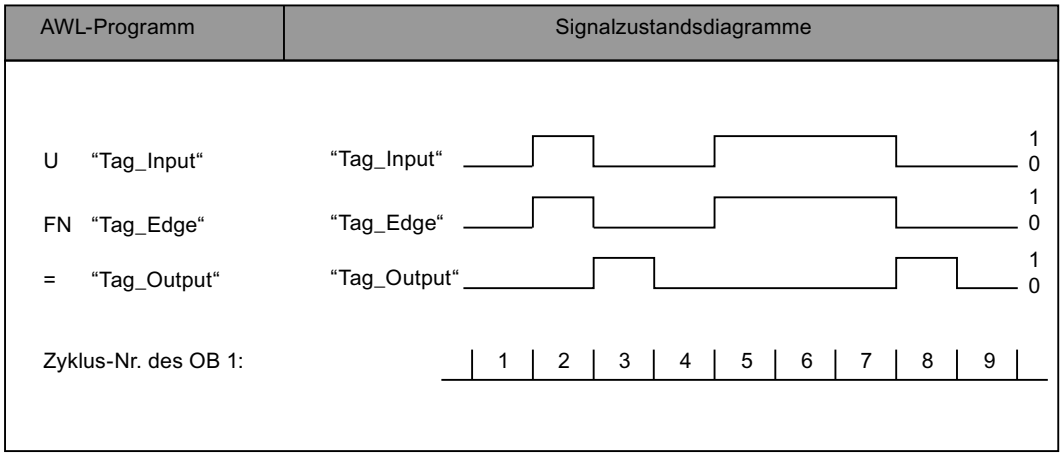

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

FP: VKE auf positive Signalflanke abfragen

Beschreibung

Mit der Anweisung "VKE auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand des Verknüpfungsergebnisses (VKE) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker <Operand> gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Das folgende Bild zeigt den Wechsel im VKE bei einer negativen und einer positiven Signalflanke:

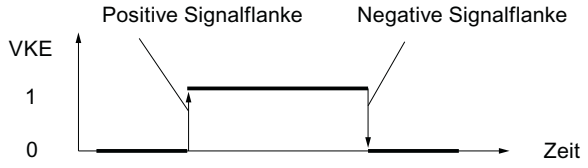

Die Abfrage der positiven Signalflanke wird bei jeder Ausführung der Anweisung durchgeführt. Wenn eine positive Signalflanke erfasst wird, wird das VKE nach der Flankenauswertung einen Programmzyklus lang auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand des VKE "0".

Hinweis

Verwendung der Adresse des Flankenmerkers

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalflanke abfragen":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	BOOL	E, A, M, D, L	Flankenmerker , in dem das VKE der vorherigen Abfrage gespeichert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

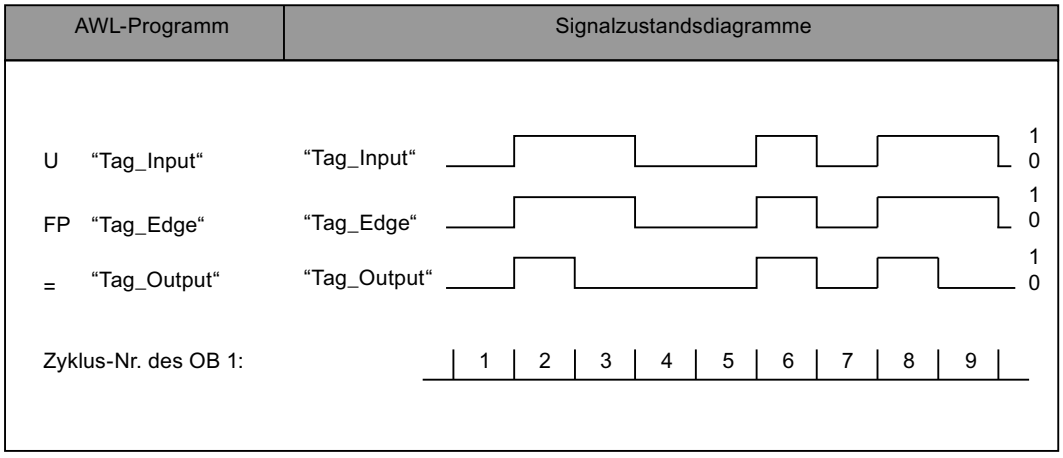

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Bitverknüpfungen (Seite 185)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

Zeiten

FR: Zeit freigeben

Beschreibung

Mit der Anweisung "Zeit freigeben" bewirken Sie den Neustart einer Zeit. Die Anweisung wird mit einer positiven Signalflanke ausgeführt und setzt den internen Flankenmerker für das Starten einer Zeit zurück. Bei der Bearbeitung einer Anweisung zum Starten der Zeit mit einem Verknüpfungsergebnis "1" läuft die Zeit dann neu an, auch wenn keine positive Signalflanke vor der Startanweisung vorliegt. Bei einem Verknüpfungsergebnis "0" vor einer Startanweisung hat das Rücksetzen des internen Flankenmerkers keinen Einfluss auf die Zeit.

Ein Wechsel des Verknüpfungsergebnisses auf "0" vor der Anweisung "Zeit freigeben" hat nach dem Neustart der Zeit keine Wirkung.

Im Akkumulator 1 muss ein gültiger Zeitwert im Format S5TIME stehen, auch wenn bei der Bearbeitung einer Startfunktion die Zeit nicht gestartet wird.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit freigeben":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die freigegeben wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_EnableInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyTimer"	// Bei einer positiven Signalflanke Zeit freigeben.
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

L: Zeitwert laden (Seite 4945)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L: Zeitwert laden

Beschreibung

Mit der Anweisung "Zeitwert laden" laden Sie den Zeitwert einer angegebenen Zeit in Akkumulator 1. Der dualcodierte Zeitwert wird dabei ohne die Zeitbasis geladen. Anstelle der Zeitbasis werden in den Akkumulator 1 Nullen geschrieben.

Nach dem Laden entspricht der im Akkumulator 1 stehende Wert einer positiven Zahl des Datentyps INT.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Das folgende Bild zeigt beispielweise, wie der Zeitwert in den Akkumulator geladen wird:

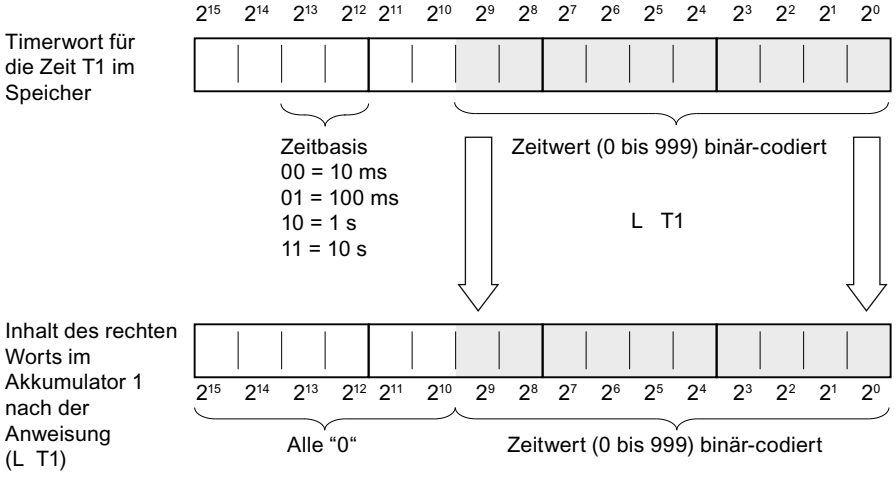

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitwert laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, deren aktueller Zeitwert geladen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_EnableInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyTimer"	// Bei einer positiven Signalflanke Zeit freigeben.
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer(10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

LC: Zeitwert im BCD-Format laden

Beschreibung

Mit der Anweisung "Zeitwert im BCD-Format laden" laden Sie den Zeitwert einer angegebenen Zeit, der dualcodiert vorliegt, im BCD-Format in Akkumulator 1. Beim Laden wird dabei auch die Zeitbasis in Akkumulator 1 übertragen.

Nach dem Laden entspricht der im Akkumulator 1 stehende Wert einer Zeitdauer im Format S5TIME. Das linke Wort im Akkumulator 1 wird mit Nullen belegt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Das folgende Bild zeigt, wie der Zeitwert in den Akkumulator geladen wird:

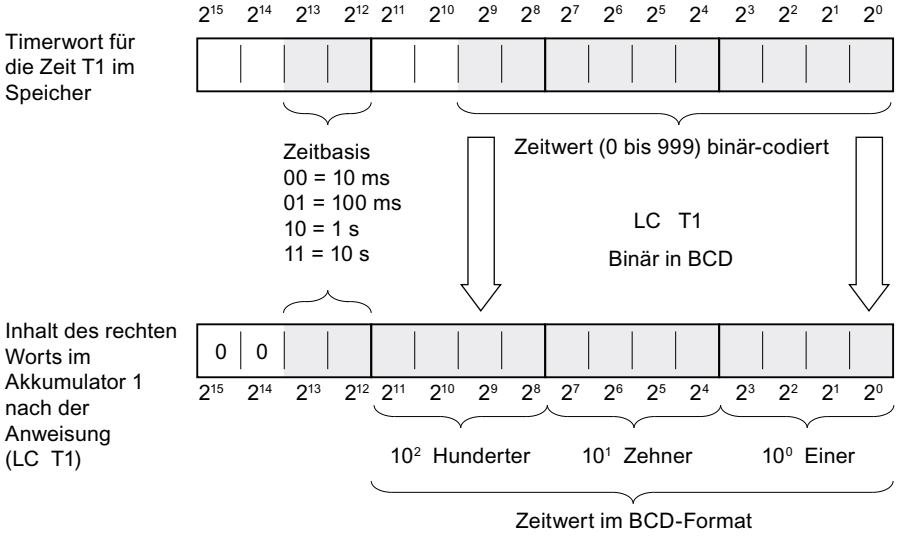

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitwert im BCD-Format laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, deren aktueller Zeitwert im BCD-Format geladen wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_EnableInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyTimer"	// Bei einer positiven Signalflanke Zeit freigeben.
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer(10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
LC "MyTimer"	// Aktuellen Zeitwert der Zeit im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

R: Zeit rücksetzen

Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine angegebene Zeit auf "0" zurück. Die Anweisung wird ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) "1" ist. Solange das VKE "1" vor der Anweisung ansteht, liefert die angegebene Zeit den Wert "0". Die Anweisung setzt auch den Zeitwert und die Zeitbasis der programmierten Zeitdauer auf "0" zurück.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Hinweis

Die Anweisung "Zeit rücksetzen" setzt den internen Flankenmerker nicht zurück. Zum Zurücksetzen des internen Flankenmerkers müssen Sie die Anweisung "Zeit freigeben" programmieren oder eine Anweisung zum Starten der Zeit mit dem Signalzustand "0" ausführen.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die zurückgesetzt wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Reset"`	`// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`R "MyTimer"`	`// Bei VKE = "1" Zeit zurücksetzen.`
`U "MyTimer"`	`// Status der Zeit auf "1" abfragen.`
`= "Tag_TimerStatus"`	`// Zeitstatus dem Operanden zuweisen.`
`L "MyTimer"`	`// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.`
`T "Tag_TimerValue"`	`// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.`

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SI: Zeit als Impuls starten

Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, solange das Verknüpfungsergebnis (VKE) vor der Anweisung "1" bleibt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die Zeit angehalten. Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", solange die Zeit läuft.

Ein Neustart der Zeit kann mit einer positiven Signalflanke vor der Anweisung "Zeit als Impuls starten" oder mithilfe der Anweisung "Zeit freigeben" bewirkt werden. Voraussetzung dafür ist, dass die Zeit nicht zurückgesetzt ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als Impuls starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält, wird die Synchronfehlerbehandlung (STOP oder OB122) durchgeführt. Das Statusbit OV wird nicht beeinflusst.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Impuls gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SI "MyTimer"	// Zeit als Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

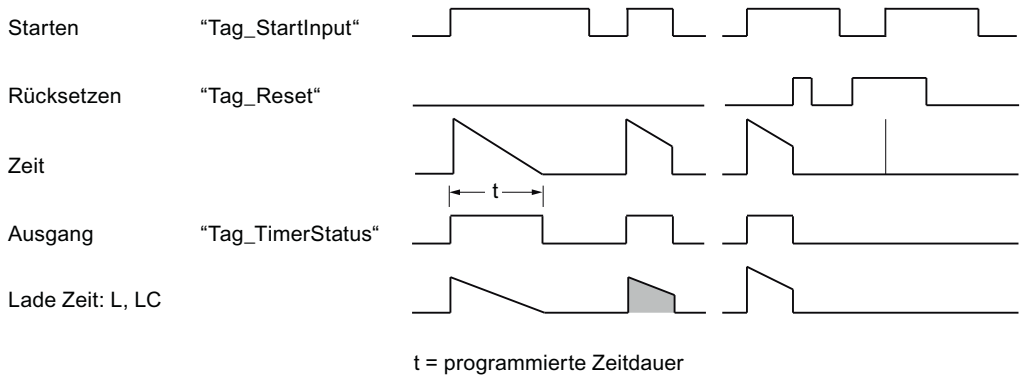

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SV: Zeit als verlängerten Impuls starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, auch wenn das Verknüpfungsergebnis vor der Anweisung auf "0" wechselt. Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", solange die Zeit läuft.

Die Anweisung startet bei jeder positiven Signalflanke die Zeit mit der programmierten Zeitdauer neu, auch wenn die Zeit noch nicht abgelaufen ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als verlängerten Impuls starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält, wird die Synchronfehlerbehandlung (STOP oder OB 122) durchgeführt. Das Statusbit OV wird nicht beeinflusst.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als verlängerter Impuls gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SV "MyTimer"	// Zeit als verlängerter Impuls starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

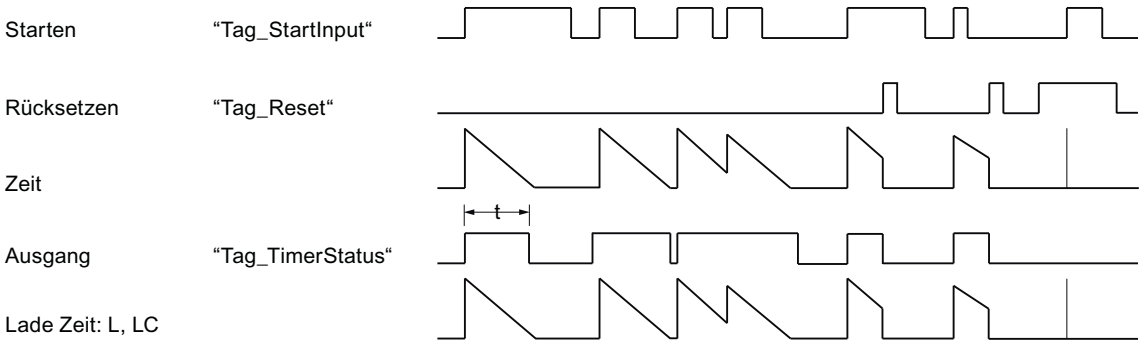

t = programmierte Zeitdauer

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

SE: Zeit als Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn am Starteingang der Signalzustand "1" erfasst wird. Solange dieser Signalzustand "1" bleibt, läuft die Zeit mit der in Akkumulator 1 stehenden Zeitdauer ab. Wenn die Zeit abgelaufen ist und der Signalzustand am Starteingang noch "1" ist, ergibt die Abfrage des Zeitstatus ebenfalls "1". Wenn das Signal am Starteingang "0" ist, wird die Zeit zurückgesetzt. Die Abfrage des Zeitstatus ergibt in diesem Fall den Signalzustand "0". Sobald das Signal am Starteingang wieder auf "1" wechselt, beginnt die Zeit erneut abzulaufen.

Der Signalzustand des Ausgangs der Zeit ist identisch zum Signalzustand des Starteingangs. Der Starteingang ist direkt mit dem Ausgang verschaltet und hat keine Beziehung zur Zeit.

Die Zeitdauer im Akkumulator 1 setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält und das VKE der Vorverknüpfung den Wert "0" liefert, wird das Statusbit OV auf "1" gesetzt. Es wird keine Synchronfehlerbehandlung gestartet. Wenn der Akkumulator 1 einen gültigen BCD-Wert enthält, wird das Statusbit OV auf "0" zurückgesetzt.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Einschaltverzögerung gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SE "MyTimer"	// Zeit als Einschaltverzögerung starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert an den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format in den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

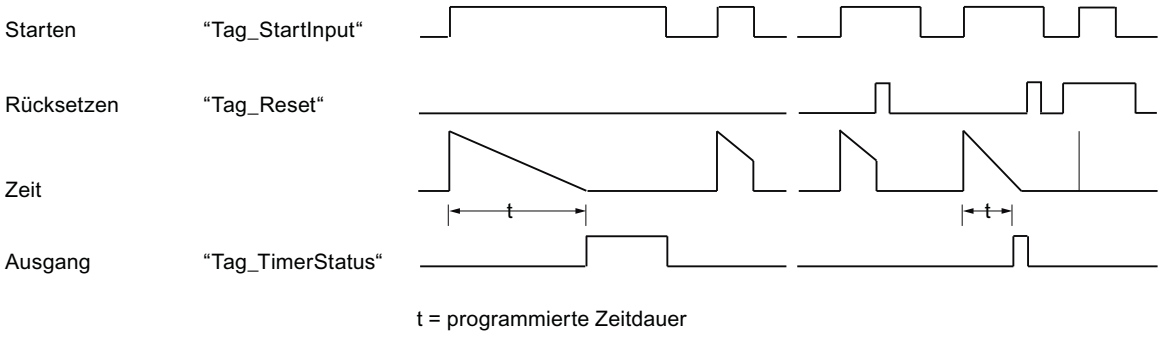

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SS: Zeit als speichernde Einschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine angegebene Zeit bei einer positiven Signalflanke. Wenn eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab, auch wenn das Verknüpfungsergebnis (VKE) vor der Anweisung auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "1". Das Abfrageergebnis wird dabei durch den Signalzustand des aktuellen Verknüpfungsergebnisses vor der Anweisung nicht beeinflusst.

Die Anweisung startet bei jeder positiven Signalflanke die Zeit mit der programmierten Zeitdauer neu, auch wenn die Zeit noch nicht abgelaufen ist.

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält, wird die Synchronfehlerbehandlung (STOP oder OB 122) durchgeführt. Das Statusbit OV wird nicht beeinflusst.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als speichernde Einschaltverzögerung gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SS "MyTimer"	// Zeit als speichernde Einschaltverzögerung starten, wenn eine positive Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert in den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format in den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

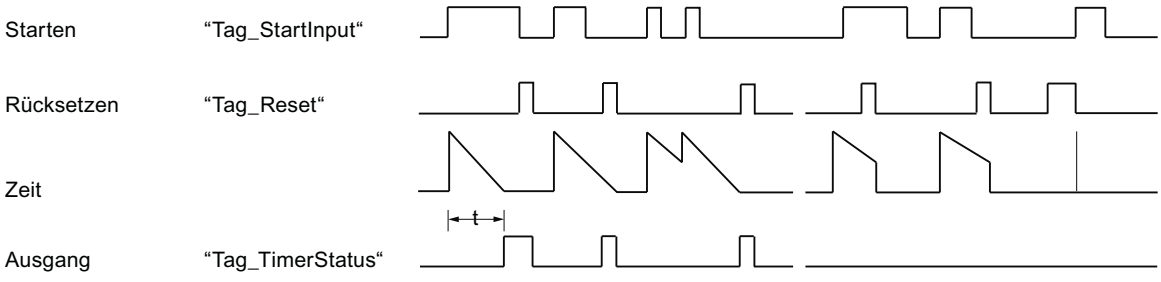

t = programmierte Zeitdauer

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SA: Zeit als Ausschaltverzögerung starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine angegebene Zeit bei einer negativen Signalflanke. Wenn eine negative Signalflanke erfasst wird, wird die Anweisung ausgeführt und die Zeit gestartet. Die Zeit läuft dann mit der in Akkumulator 1 stehenden Zeitdauer ab. Wenn das Verknüpfungsergebnis (VKE) vor dem Ablauf der Zeit auf "1" wechselt, wird die Zeit zurückgesetzt. Erst wenn eine negative Signalflanke bei der Ausführung der Anweisung erfasst wird, wird die Zeit wieder gestartet.

Die Abfragen des Zeitstatus auf "1" liefern das Abfrageergebnis "1", wenn der Signalzustand des Verknüpfungsergebnisses bei der Ausführung der Anweisung "1" ist oder die programmierte Zeit läuft. Wenn die Zeit nicht läuft oder das VKE auf "0" ist, liefern die Abfragen des Zeitstatus auf "1" das Abfrageergebnis "0".

Die Zeitdauer im Akkumulator 1 setzt sich aus einem Zeitwert und einer Zeitbasis zusammen. Wenn eine angegebene Zeit mit der Anweisung "Zeit als Ausschaltverzögerung starten" gestartet wird, wird der Zeitwert entsprechend der Zeitbasis heruntergezählt. Die Zeit läuft so lange, bis der Zählwert auf Null heruntergezählt ist.

Im Akkumulator 1 muss eine gültige Zeitdauer im Format S5TIME stehen, auch wenn bei der Bearbeitung der Anweisung die Zeit nicht gestartet wird. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält, wird die Synchronfehlerbehandlung (STOP oder OB 122) durchgeführt. Das Statusbit OV wird nicht beeinflusst.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie hier: L: Zeitwert laden (Seite 4945)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Zeit>	Input	TIMER	T	Zeit, die als Ausschaltverzögerung gestartet wird

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_StartInput"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
L S5T#10s	// Zeitdauer (10s) laden
SA "MyTimer"	// Zeit als Ausschaltverzögerung starten, wenn eine negative Signalflanke am Operanden "Tag_StartInput" oder am Operanden "Tag_EnableInput" vorliegt und der Operand "Tag_StartInput" den Signalzustand "1" führt.
U "Tag_Reset"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyTimer"	// Bei VKE = "1" Zeit zurücksetzen und den Zeitablauf anhalten.
U "MyTimer"	// Status der Zeit auf "1" abfragen.
= "Tag_TimerStatus"	// Zeitstatus dem Operanden zuweisen.
L "MyTimer"	// Aktuellen Zeitwert der Zeit in Akkumulator 1 laden.
T "Tag_TimerValue"	// Aktuellen Zeitwert an den Operanden "Tag_TimerValue" übertragen.
LC "MyTimer"	// Aktuellen Zeitwert im BCD-Format in Akkumulator 1 laden.
T "Tag_TimerValue_BCD"	// Zeitwert im BCD-Format an den Operanden "Tag_TimerValue_BCD" übertragen.

Das folgende Bild zeigt das Zeitdiagramm des Beispiels:

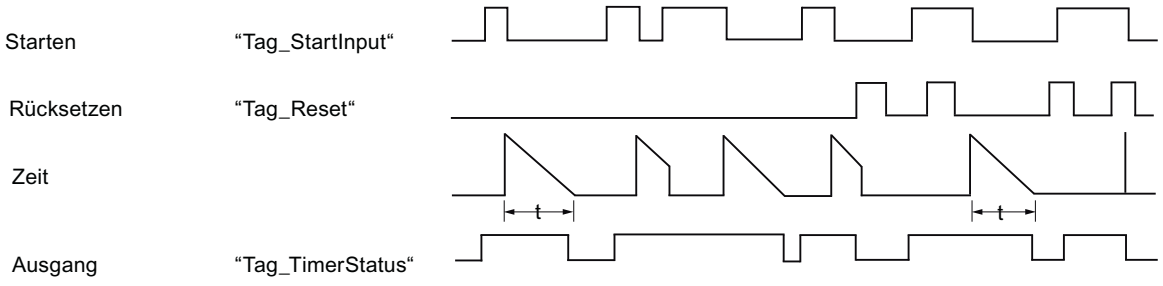

t = programmierte Zeitdauer

Siehe auch

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Zähler

FR: Zähler freigeben

Beschreibung

Mit der Anweisung "Zähler freigeben" bewirken Sie, dass die Anweisungen "Zähler setzen", "Vorwärts zählen" und "Rückwärts zählen" ohne eine aktuell vorliegende positive Signalflanke ausgeführt werden. Bei der Ausführung setzt die Anweisung den internen Flankenmerker auf den Signalzustand "0" zurück. Die Abfrage vor der entsprechenden Zählanweisung muss weiterhin den Signalzustand "1" führen.

Die Anweisung "Zähler freigeben" wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt.

Das Freigeben eines Zählers ist für die Ausführung einer Zählanweisung nicht erforderlich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler freigeben":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, der freigegeben wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
FR "MyCounter"	// Zähler freigeben
U "Tag_Input_2"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ZR "MyCounter"	// Rückwärts zählen
U "MyCounter"	// Zählerstatus abfragen
= "Tag_Output"	// Abfrageergebnis dem Operanden zuweisen.

Im Beispiel wird der Zähler "MyCounter" freigegeben, wenn der Signalzustand des Operanden "Tag_Input_1" von "0" auf "1" wechselt. Wenn der Operand "Tag_Input_2" den Signalzustand "1" führt, wird der Zählwert beim Freigeben des Zählers um eins verringert. Bei einem Signalzustand "0" des Operanden "Tag_Input_2", bleibt der Zählwert unverändert. Der Zählwert wird auch unabhängig von der Zählerfreigabe um Eins verringert, wenn der Signalzustand des Operanden "Tag_Input_2" von "0" auf "1" wechselt.

Die Abfrage des Zählerstatus liefert das Ergebnis "1", wenn Zählerstand größer Null ist. Wenn der Zählerstand gleich Null ist, ist das Abfrageergebnis "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L: Zählwert laden

Beschreibung

Mit der Anweisung "Zählwert laden" übertragen Sie den aktuellen Wert eines angegebenen Zählers dualcodiert in das rechte Wort des Akkumulators 1. Der geladene Zählwert steht dann als 16-Bit Ganzzahl zur weiteren Bearbeitung zur Verfügung.

Das folgende Bild zeigt die Funktionsweise der Anweisung:

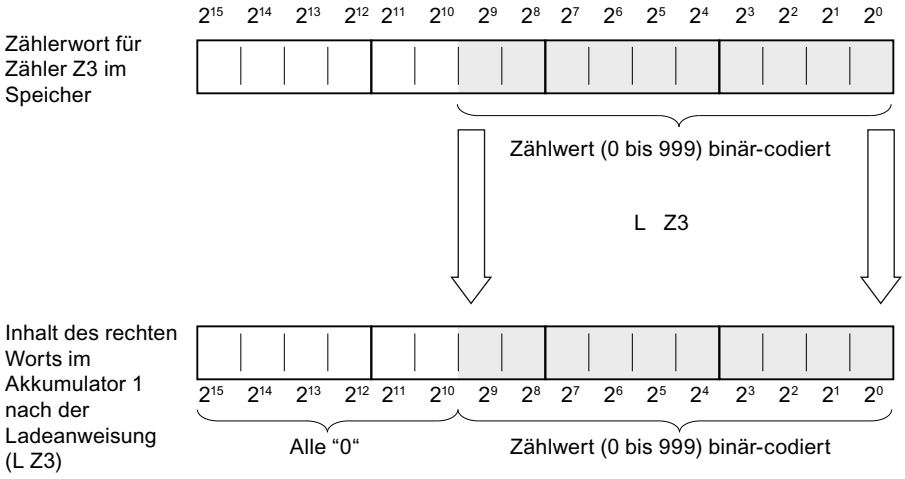

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zählwert laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen aktueller Wert geladen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "MyCounter"	// Aktuellen Zählwert dualcodiert in den Akkumulator 1 laden.
L "Tag_Value"	// Zählwert in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value" in Akkumulator 1 laden.
>=I	// Abfrage, ob der Zählwert größer oder gleich dem Wert des Operanden "Tag_Value" ist.
= "Tag_Output"	// Abfrageergebnis dem Operanden zuweisen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

LC: Zählwert im BCD-Format laden

Beschreibung

Mit der Anweisung "Zählwert im BCD-Format laden" übertragen Sie den aktuellen Wert eines angegebenen Zählers als BCD in das rechte Wort des Akkumulators 1. Der geladene Zählwert steht dann in BCD-Form zur weiteren Bearbeitung zur Verfügung.

Das folgende Bild zeigt die Funktionsweise der Anweisung "Zählwert im BCD-Format laden":

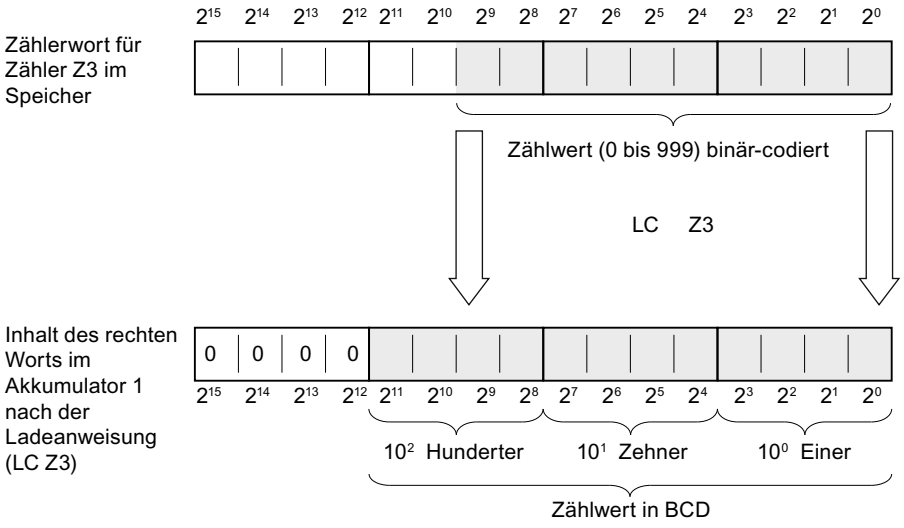

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zählwert im BCD-Format laden":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen aktueller Wert geladen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`LC "MyCounter"`	`// Aktuellen Zählwert als BCD in den Akkumulator 1 laden.`
`T "Tag_Output"`	`// Aktuellen Zählwert in den Operanden "Tag_Output" transferieren und speichern.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

R: Zähler rücksetzen

Beschreibung

Mit der Anweisung "Zähler rücksetzen" setzen Sie den aktuellen Zählwert eines angegebenen Zählers auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung den Signalzustand "1" führt. Solange das VKE vor der Anweisung "Zähler rücksetzen" auf "1" steht, liefern die Abfragen des Zählerstatus auf "1" das Ergebnis "0".

Durch das Zurücksetzen eines Zählers wird der interne Flankenmerker für die Anweisungen "Zähler setzen", "Vorwärts zählen" und "Rückwärts zählen" nicht zurückgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler rücksetzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert auf "0" zurückgesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
R "MyCounter"	// Zähler zurücksetzen

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

S: Zähler setzen

Beschreibung

Mit der Anweisung "Zähler setzen" setzen Sie einen angegebenen Zähler auf den Wert, der sich zum Zeitpunkt der Abfrage im Akkumulator 1 befindet.

Die Anweisung "Zähler setzen" wird nur ausgeführt, wenn das Verknüpfungsergebnis vor der Anweisung von "0" auf "1" wechselt. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Zum Setzen eines Zählers muss im Akkumulator 1 ein gültiger Zählwert stehen. Gültige Zählwerte sind Werte im BCD-Format mit drei Dekaden. Es sind dabei nur positive Zählwerte erlaubt, negative Zählwerte kann die Zählanweisung nicht bearbeiten. Wenn der Akkumulator 1 keinen gültigen BCD-Wert enthält, wird die Synchronfehlerbehandlung (STOP oder OB122) durchgeführt. Das Statusbit OV wird nicht beeinflusst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zähler setzen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, der gesetzt wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input"`	`// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`L "Tag_CountValue"`	`// Zählwert in Akkumulator 1 laden.`
`S "MyCounter"`	`// Zähler auf den Wert setzen, der im Akkumulator 1 steht.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ZV: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den aktuellen Zählwert um eins, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt. Der Zählwert wird bei jeder positiven Signalflanke weiterhin erhöht, bis er den oberen Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, hat das VKE vor der Anweisung keinen Einfluss mehr auf den Zählwert.

Zur Ausführung der Anweisung "Vorwärts zählen" ist immer eine positive Signalflanke erforderlich. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert bei einer positiven Flanke um eins erhöht wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input"	// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
ZV "MyCounter"	// Zählwert bei einer positiven Flanke um eins erhöhen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

ZR: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den aktuellen Zählwert um eins, wenn das Verknüpfungsergebnis (VKE) vor der Anweisung von "0" auf "1" wechselt. Der Zählwert wird bei jeder positiven Signalflanke weiterhin verringert, bis er den unteren Grenzwert "0" erreicht. Wenn der Grenzwert erreicht ist, hat das VKE vor der Anweisung keinen Einfluss mehr auf den Zählwert. Ein Zählen mit negativem Zählwert findet nicht statt.

Zur Ausführung der Anweisung "Rückwärts zählen" ist immer eine positive Signalflanke erforderlich. Die positive Signalflanke kann dabei auch durch die Anweisung "Zähler freigeben" bewirkt werden, die den internen Flankenmerker auf "0" zurücksetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

Operand	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	COUNTER	Z	Zähler, dessen Zählwert bei einer positiven Flanke um eins verringert wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input"`	`// Signalzustand auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`ZR "MyCounter"`	`// Zählwert bei einer positiven Flanke um eins verringern.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Zeit- und Zählanweisungen (Seite 187)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

Vergleicher

? I: 16-Bit Ganzzahlen vergleichen

Beschreibung

Mit der Anweisung "16-Bit Ganzzahlen vergleichen" vergleichen Sie den Inhalt des rechten Wortes im Akkumulator 2 mit dem Inhalt des rechten Worts im Akkumulator 1. Die Inhalte der Akkumulatoren 1 und 2 werden als 16-Bit Ganzzahlen interpretiert. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==I	16-Bit Ganzzahlen auf "gleich" vergleichen
<>I	16-Bit Ganzzahlen auf "ungleich" vergleichen
>I	16-Bit Ganzzahlen auf "größer" vergleichen
<I	16-Bit Ganzzahlen auf "kleiner" vergleichen
>=I	16-Bit Ganzzahlen auf "größer oder gleich" vergleichen
<=I	16-Bit Ganzzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==I	0	1	0
<>I	1	0	1
>I	1	0	0
<I	0	0	1
>=I	1	1	0
<=I	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Wert des Operanden "Tag_Input_1" in Akkumulator 1 laden.
L "Tag_Input_2"	// Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Input_2" in Akkumulator 1 laden.
>I	// Vergleich, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
= "Tag_Output"	// Ergebnis des Vergleichs dem Operanden "Tag_Output" zuweisen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	10	10	5
Tag_Input_2	5	10	10
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

? D: 32-Bit Ganzzahlen vergleichen

Beschreibung

Mit der Anweisung "32-Bit Ganzzahlen vergleichen" vergleichen Sie den Inhalt des Akkumulators 2 (AKKU 2) mit dem Inhalt des Akkumulators 1 (AKKU 1). Die Inhalte der Akkumulatoren 1 und 2 werden als 32-Bit Ganzzahlen interpretiert. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==D	32-Bit Ganzzahlen auf "gleich" vergleichen
<>D	32-Bit Ganzzahlen auf "ungleich" vergleichen

Vergleichsfunktion	Bedeutung
>D	32-Bit Ganzzahlen auf "größer" vergleichen
<D	32-Bit Ganzzahlen auf "kleiner" vergleichen
>=D	32-Bit Ganzzahlen auf "größer oder gleich" vergleichen
<=D	32-Bit Ganzzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf den Signalzustand "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==D	0	1	0
<>D	1	0	1
>D	1	0	0
<D	0	0	1
>=D	1	1	0
<=D	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Wert des Operanden "Tag_Input_1" in Akkumulator 1 laden.
L "Tag_Input_2"	// Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumulator 2 verschoben.
	// Wert des Operanden "Tag_Input_2" in Akkumulator 1 laden.
>D	// Vergleich, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
= "Tag_Output"	// Ergebnis des Vergleichs dem Operanden "Tag_Output" zuweisen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	150 000	150 000	100 000
Tag_Input_2	100 000	150 000	150 000
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

? R: Gleitpunktzahlen vergleichen

Beschreibung

Mit der Anweisung "Gleitpunktzahlen vergleichen" vergleichen Sie den Inhalt des Akkumulators 2 (AKKU 2) mit dem Inhalt des Akkumulators 1 (AKKU 1). Die Inhalte der Akkumulatoren 1 und 2 werden als Gleitpunktzahlen interpretiert. Zur Ausführung des Vergleichs müssen in den Akkumulatoren 1 und 2 gültige Gleitpunktzahlen stehen. Die Anweisung führt den Vergleich abhängig von der angegebenen Vergleichsfunktion durch.

Die folgende Tabelle zeigt die verfügbaren Vergleichsfunktionen und deren Bedeutung:

Vergleichsfunktion	Bedeutung
==R	Gleitpunktzahlen auf "gleich" vergleichen
<>R	Gleitpunktzahlen auf "ungleich" vergleichen
>R	Gleitpunktzahlen auf "größer" vergleichen
<R	Gleitpunktzahlen auf "kleiner" vergleichen
>=R	Gleitpunktzahlen auf "größer oder gleich" vergleichen
<=R	Gleitpunktzahlen auf "kleiner oder gleich" vergleichen

Das Ergebnis des Vergleichs beeinflusst das Verknüpfungsergebnis (VKE) sowie die Statusbits A1 und A0. Wenn die Bedingung des Vergleichs erfüllt ist, wird das VKE auf den Signalzustand "1" gesetzt. Wenn die Bedingung des Vergleichs nicht erfüllt ist, wird das VKE auf den Signalzustand "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie die einzelnen Vergleichsfunktionen das VKE beeinflussen:

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
==R	0	1	0
<>R	1	0	1

Vergleichsfunktion	VKE, wenn AKKU 2 > AKKU 1	VKE, wenn AKKU 2 = AKKU 1	VKE, wenn AKKU 2 < AKKU 1
>R	1	0	0
<R	0	0	1
>=R	1	1	0
<=R	0	1	1

Das Setzen der Statusbits A1 und A0 hängt von der Relation der beiden am Vergleich beteiligten Werte ab.

Hinweis

Bei einem Vergleich ungültiger Gleitpunktzahlen setzt die CPU das Verknüpfungsergebnis auf "0" und die Statusbits A0, A1, OV und OS auf "1".

Die Inhalte der Akkumulatoren 1 und 2 werden durch die Ausführung der Vergleichsfunktionen nicht verändert. Nach der Ausführung einer Vergleichsfunktion können Sie das Verknüpfungsergebnis mit Zuweisungs- und Sprungfunktionen auswerten.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input_1"`	`// Wert des Operanden "Tag_Input_1" in Akkumulator 1 laden.`
`L "Tag_Input_2"`	`// Inhalt des Akkumulators 1 (Tag_Input_1) in Akkumulator 2 verschieben.`
	`// Wert des Operanden "Tag_Input_2" in Akkumulator 1 laden.`
`>R`	`// Vergleich, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.`
`= "Tag_Output"`	`// Ergebnis des Vergleichs dem Operanden "Tag_Output" zuweisen.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte		
Tag_Input_1	10 515	10 515	5 232
Tag_Input_2	5 232	10 515	10 515
Tag_Output	1	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Vergleichsanweisungen (Seite 190)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Mathematische Funktionen

Ganzzahlen

+I: Ganzzahlen (16 Bit) addieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) addieren" addieren Sie die Werte, die sich in den rechten Wörtern der Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Summe speichert die Anweisung im rechten Wort des Akkumulators 1. Das linke Wort des Akkumulators 1 bleibt unverändert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Addition laden.
L "Tag_Value_2"	// Zweiten Wert der Addition laden.
+I	// Werte addieren
T "Tag_Result"	// Summe in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

-I: Ganzzahlen (16 Bit) subtrahieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) subtrahieren" subtrahieren Sie den Wert des rechten Worts des Akkumulators 1 vom Wert des rechten Worts im Akkumulator 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Differenz speichert die Anweisung im rechten Wort des Akkumulators 1. Das linke Wort des Akkumulators 1 bleibt unverändert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Addition in Akkumulator 1 laden.
-I	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

*I: Ganzzahlen (16 Bit) multiplizieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) multiplizieren" multiplizieren Sie die Werte, die sich in den rechten Wörtern der Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 16-Bit Ganzzahlen.

Das Produkt speichert die Anweisung im Akkumulator 1 als 32-Bit Ganzzahl (DINT).

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Multiplikation laden.
L "Tag_Value_2"	// Zweiten Wert der Multiplikation laden.
*I	// Werte multiplizieren
T "Tag_Result"	// Produkt in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

/I: Ganzzahlen (16 Bit) dividieren

Beschreibung

Mit der Anweisung "Ganzzahlen (16 Bit) dividieren" dividieren Sie den Wert des rechten Worts im Akkumulator 2 durch den Wert des rechten Worts im Akkumulator 1. Die Werte der Akkumulatoren interpretiert die Anweisung als 16-Bit Ganzzahlen.

Die Anweisung liefert zwei Ergebnisse und zwar den Quotienten und den übriggebliebenen Rest der Division. Der Quotient ist das ganzzahlige Ergebnis der Division. Diesen speichert die Anweisung im rechten Wort des Akkumulators 1. Den Rest der Division speichert die Anweisung im linken Wort des Akkumulators 1. Bei einem negativen Dividenden ist der Rest der Division auch negativ.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten und als Rest jeweils den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Dividend in Akkumulator 1 laden.`
`L "Tag_Value_2"`	`// Dividend in Akkumulator 2 verschieben.`
	`// Divisor in Akkumulator 1 laden.`
`/I`	`// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.`
`T "Tag_Result"`	`// Ergebnis in den Operanden "Tag_Result" übertragen.`

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der rechten Wörter in den Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Ganzzahlen (16 Bit) dividieren":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Ganzzahlen (16 Bit) dividieren":

Akkumulator 1	Wert
Bit 0 bis 15 (Rechtes Wort)	3
Bit 16 bis 31 (Linkes Wort)	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

+D: Ganzzahlen (32 Bit) addieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) addieren" addieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 32-Bit Ganzzahlen. Die Summe speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Ersten Wert der Addition laden.
L "Tag_Value_2"             // Zweiten Wert der Addition laden.
+D                          // Werte addieren
T "Tag_Result"              // Summe in den Operanden "Tag_Result" übertragen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

-D: Ganzzahlen (32 Bit) subtrahieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) subtrahieren" subtrahieren Sie den Wert des Akkumulators 1 vom Wert des Akkumulators 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 32-Bit Ganzzahlen. Das Ergebnis der Subtraktion speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Subtraktion in Akkumulator 1 laden.
-D	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

*D: Ganzzahlen (32 Bit) multiplizieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) multiplizieren" multiplizieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als 32-Bit Ganzzahlen. Das Produkt speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Ersten Wert der Multiplikation laden.`
`L "Tag_Value_2"`	`// Zweiten Wert der Multiplikation laden.`
`*D`	`// Werte multiplizieren`
`T "Tag_Result"`	`// Produkt in den Operanden "Tag_Result" übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

/D: Ganzzahlen (32 Bit) dividieren

Beschreibung

Mit der Anweisung "Ganzzahlen (32 Bit) dividieren" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als 32-Bit Ganzzahlen. Das ganzzahlige Ergebnis der Division (Quotient) speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Dividend in Akkumulator 1 laden.`
`L "Tag_Value_2"`	`// Dividend in Akkumulator 2 verschieben.`
	`// Divisor in Akkumulator 1 laden.`
`/D`	`// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.`
`T "Tag_Result"`	`// Quotienten in den Operanden "Tag_Result" übertragen.`

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Ganzzahlen (32 Bit) dividieren":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Ganzzahlen (32 Bit) dividieren":

Akkumulator	Wert
Akkumulator 1	3

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

+: Konstanten addieren

Beschreibung

Mit der Anweisung "Konstanten addieren" addieren Sie den Wert einer angegebenen Konstante zum Inhalt des Akkumulators 1. Den Konstantenwert können Sie als 16- oder 32-Bit Ganzzahl angeben. Das Ergebnis speichert die Anweisung im Akkumulator 1. Bei Addition von 16-Bit Ganzzahlen beeinflusst die Anweisung nur das rechte Wort des Akkumulators 1.

Um die Addition einer 16-Bit Ganzzahl als DINT-Rechnung auszuführen, müssen Sie vor der Konstante "L#" eingeben. Die Anweisung "Konstanten addieren" erlaubt die Verwendung negativer Zahlen.

Die Anweisung beeinflusst keine Statusbits.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Konstanten addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	INT, DINT	Konstante, die addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" laden.
+ 25                         // Den Wert des Akkumulators 1 um 25 erhöhen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 an den Operanden "Tag_Re-
                             sult_1" übertragen.
```

AWL	Erläuterung
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
+ -100	// Den Wert des Akkumulators 1 um 100 verringern.
<I	// Abfragen, ob der Wert des Akkumulators 2 kleiner als der Wert des Akkumulators 1 ist.
SPB NEXT	// Bei einem Abfrageergebnis = "1" zur Sprungmarke NEXT springen.
	// Bei einem Abfrageergebnis = "0" die nächstfolgende Anweisung bearbeiten.
T "Tag_Result_2"	Inhalt des Akkumulators 1 in den Operanden "Tag_Result_2" übertragen.
L "Tag_Value_4"	// Wert des Operanden "Tag_Value_4" laden.
+ L#20	// Inhalt des Akkumulators 1 um 20 erhöhen.
	// Die Addition erfolgt entsprechend einer DINT-Rechnung.
T "Tag_Result_3"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result_3" übertragen.
NEXT: L "Tag_Value_5"	// Sprungmarke "NEXT"
	// Wert des Operanden "Tag_Value_5" laden.
+D	// Werte des Akkumulators 1 und 2 addieren.
T "Tag_Result_4"	// Summe in den Operanden "Tag_Result_4" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

INC: Inkrementieren

Beschreibung

Mit der Anweisung "Inkrementieren" erhöhen Sie den im Akkumulator 1 stehenden Wert um den Wert, der als Parameter der Anweisung angegeben wird. Der angegebene Wert kann in einem Wertebereich von 0 bis 255 liegen. Das Erhöhen erfolgt nur im rechten Byte des Akkumulators 1. Ein Übertrag auf die links liegenden Bytes findet nicht statt.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren 2, 3 und 4 bleiben unverändert.

Wenn der Parameterwert außerhalb des Wertebereichs liegt, dann wird das Ergebnis verfälscht ohne eine Fehlermeldung zu generieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

Parameter	Format	Beschreibung
<Konstante>	8-Bit Ganzzahl	Wert, um den erhöht wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value"                // Wert des Operanden in Akkumulator 1 laden.
INC 5                        // Wert des Akkumulators 1 um 5 erhöhen.
T "Tag_Result"               // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

DEC: Dekrementieren

Beschreibung

Mit der Anweisung "Dekrementieren" verringern Sie den im Akkumulator 1 stehenden Wert um den Wert, der als Parameter der Anweisung angegeben wird. Der angegebene Wert kann in einem Wertebereich von 0 bis 255 liegen. Das Verringern erfolgt nur im rechten Byte des Akkumulators 1. Ein Übertrag auf die links liegenden Bytes findet nicht statt.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren 2, 3 und 4 bleiben unverändert.

Wenn der Parameterwert außerhalb des Wertebereichs liegt, dann wird das Ergebnis verfälscht ohne eine Fehlermeldung zu generieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

Parameter	Format	Beschreibung
<Konstante>	8-Bit Ganzzahl	Wert, um den verringert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value"               // Wert des Operanden in Akkumulator 1 laden.
DEC 5                       // Wert des Akkumulators 1 um 5 verringern.
T "Tag_Result"              // Inhalt des Akkumulators 1 in den Operanden transfe-
                               rieren.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

MOD: Divisionsrest gewinnen

Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1 und speichern den Divisionsrest.

Die Anweisung interpretiert die Werte der Akkumulatoren 1 und 2 als 32-Bit Ganzzahlen. Den Divisionsrest speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Divisionsrest negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Ergebnis den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
MOD	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Divisionsrest in den Operanden "Tag_Result" übertragen.

Vor der Ausführung

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 und 2 vor der Ausführung der Anweisung "Divisionsrest gewinnen":

Akkumulator	Wert
Akkumulator 2	13
Akkumulator 1	4

Nach der Ausführung

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 nach der Ausführung der Anweisung "Divisionsrest gewinnen":

Akkumulator	Wert
Akkumulator 1	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Gleitpunktzahlen

+R: Gleitpunktzahlen addieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen addieren" addieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als Gleitpunktzahlen. Die Summe speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Summe negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Addition laden.
L "Tag_Value_2"	// Zweiten Wert der Addition laden.
+R	// Werte addieren
T "Tag_Result"	// Summe in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

-R: Gleitpunktzahlen subtrahieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen subtrahieren" subtrahieren Sie den Wert des Akkumulators 1 vom Wert des Akkumulators 2. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als Gleitpunktzahlen. Die Differenz speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob die Differenz negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Subtraktion in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Zweiten Wert der Subtraktion in Akkumulator 1 laden.
-R	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Result"	// Differenz in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

*R: Gleitpunktzahlen multiplizieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen multiplizieren" multiplizieren Sie die Werte, die sich in den Akkumulatoren 1 und 2 befinden. Diese Werte interpretiert die Anweisung als Gleitpunktzahlen. Das Produkt speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Produkt negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Ersten Wert der Multiplikation laden.
L "Tag_Value_2"	// Zweiten Wert der Multiplikation laden.
*R	// Werte multiplizieren
T "Tag_Result"	// Produkt in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

/R: Gleitpunktzahlen dividieren

Beschreibung

Mit der Anweisung "Gleitpunktzahlen dividieren" dividieren Sie den Wert des Akkumulators 2 durch den Wert des Akkumulators 1. Die Werte der Akkumulatoren 1 und 2 interpretiert die Anweisung als Gleitpunktzahlen. Das ganzzahlige Ergebnis der Division (Quotient) speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob der Quotient negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei einer Division durch Null liefert die Anweisung als Quotienten den Wert Null. In diesem Fall werden die Statusbits A0, A1, OV und OS auf den Signalzustand "1" gesetzt.

Bei Eingabe einer ungültigen Gleitpunktzahl oder des Zeichens für Unendlich (+/- ∞) schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Bei Steuerungen mit zwei Akkumulatoren bleibt der Inhalt des Akkumulators 2 nach der Ausführung der Anweisung unverändert.

Bei Steuerungen mit vier Akkumulatoren wird der Inhalt des Akkumulators 3 in Akkumulator 2 und der Inhalt des Akkumulators 4 in Akkumulator 3 kopiert. Der Inhalt des Akkumulators 4 bleibt nach der Ausführung der Anweisung unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Dividend in Akkumulator 1 laden.
L "Tag_Value_2"	// Dividend in Akkumulator 2 verschieben.
	// Divisor in Akkumulator 1 laden.
/R	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren.
T "Tag_Result"	// Quotienten in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" bilden Sie den Absolutwert des im Akkumulator 1 stehenden Werts. Die Anweisung interpretiert den Inhalt des Akkumulators 1 als eine Gleitpunktzahl und setzt das Vorzeichen der Mantisse auf "0". Das Ergebnis wird im Akkumulator 1 gespeichert.

Die Anweisung "Absolutwert bilden" beeinflusst keine Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Gleitpunktzahl in Akkumulator 1 laden.
ABS	// Absolutwert bilden
T "Tag_Result"	// Ergebnis in den Operanden "Tag_Result" transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	-1.5E+02
Tag_Result	1.5E+02

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Gleitpunktzahlen erweitert

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" berechnen Sie das Quadrat einer im Akkumulator 1 stehenden Gleitpunktzahl. Das Ergebnis wird im Akkumulator 1 gespeichert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
SQR	// Quadrat bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie die Quadratwurzel der Gleitpunktzahl, die in Akkumulator 1 steht. Das Ergebnis wird im Akkumulator 1 gespeichert.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn im Akkumulator 1 ein Wert kleiner Null oder eine ungültige Gleitpunktzahl steht, schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1".

Wenn im Akkumulator 1 der Wert "0" steht, liefert die Anweisung den Wert "0" zurück.

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.

AWL	Erläuterung
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
+R	// Werte der Akkumulatoren 1 und 2 addieren.
	// Summe im Akkumulator 1 speichern.
SQRT	// Quadratwurzel ziehen
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2.718282) und der im Akkumulator 1 stehenden Gleitpunktzahl. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
EXP	// Exponentialwert bilden
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus der im Akkumulator 1 stehenden Gleitpunktzahl den natürlichen Logarithmus zur Basis e (e = 2.718282).

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn im Akkumulator 1 ein Wert kleiner oder gleich Null steht, schreibt die Anweisung einen ungültigen Wert in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.`
`LN`	`// Natürlichen Logarithmus bilden`
	`// Ergebnis im Akkumulator 1 speichern.`
`T "Tag_Result"`	`// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels, der im Akkumulator 1 steht. Der Winkel muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
SIN	// Sinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl in einem Wertebereich von "-1" bis "1" vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von $-\pi/2$ bis $+\pi/2$ ($\pi = 3{,}14159$) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
ASIN	// Arcussinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus des Winkels, der im Akkumulator 1 steht. Die Größe des Winkels muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
COS	// Cosinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl in einem Wertebereich von "-1" bis "1" vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von "0" bis "π" (π = 3,14159) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
ACOS	// Arcuscosinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens des Winkels, der im Akkumulator 1 steht. Die Größe des Winkels muss im Bogenmaß angegeben werden und als Gleitpunktzahl im Akkumulator 1 vorliegen. Das Ergebnis speichert die Anweisung im Akkumulator 1.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt. Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
TAN	// Tangenswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem im Akkumulator 1 stehenden Wert die Größe des Winkels, dem dieser Wert entspricht. Der Wert des Akkumulators 1 muss als Gleitpunktzahl vorliegen.

Das Ergebnis der Anweisung wird im Bogenmaß zurückgeliefert und im Akkumulator 1 gespeichert. Abhängig vom Wert des Akkumulators 1 kann das Ergebnis in einem Wertebereich von $-\pi/2$ bis $+\pi/2$ ($\pi = 3{,}14159$) liegen.

Nach der Ausführung der Anweisung zeigen die Statusbits A0 und A1 an, ob das Ergebnis negativ, Null oder positiv ist. Wenn das Ergebnis außerhalb des zulässigen Zahlenbereichs liegt, werden die Statusbits OV und OS auf "1" gesetzt.

Wenn der im Akkumulator 1 stehende Wert den zulässigen Wertebereichs überschreitet, schreibt die Anweisung eine ungültige Gleitpunktzahl in den Akkumulator 1 und setzt die Statusbits A0, A1, OV und OS auf "1". Bei ungültigen Gleitpunktzahlen setzt die Anweisung die Statusbits A0, A1, OV und OS ebenfalls auf "1".

Die Anweisung beeinflusst nur den Inhalt des Akkumulators 1. Die Inhalte der restlichen Akkumulatoren bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
ATAN	// Arcustangenswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Laden und Transferieren

Laden

L: Laden

Beschreibung

Mit der Anweisung "Laden" laden Sie den Inhalt eines angegebenen Operanden in den Akkumulator 1. Der zu ladende Operand kann byte-, wort- oder doppelwortweise in einem der folgenden Speicherbereiche adressiert sein:

- Prozessabbild der Eingänge und Ausgänge (E, A)
- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)
- Zeiger
- Peripherie (PE)
- Timer (T)
- Zähler (Z)

Der Speicherbereich des Akkumulators 1 ist byteweise organisiert und 32 Bits breit.

Den Inhalt eines zu ladenden Operanden im Byte-Format schreibt die Anweisung rechtsbündig in den Akkumulator 1. Die restlichen Bytes im Akkumulator 1 werden mit "0" aufgefüllt.

Den Inhalt eines zu ladenden Operanden im Wort-Format schreibt die Anweisung in das rechte Wort des Akkumulators 1. Das höher adressierte Byte wird hierbei in das rechte Byte (Bits 0 bis 7) des Akkumulators 1 übertragen. Das niedriger adressierte Byte wird links daneben geschrieben. Die restlichen Bytes im linken Wort des Akkumulators 1 werden mit "0" aufgefüllt.

Den Inhalt eines zu ladenden Operanden im Doppelwort-Format schreibt die Anweisung in den 32 Bits des Akkumulators 1. Das am höchsten adressierte Byte wird hierbei in das rechte Byte (Bits 0 bis 7) des Akkumulators 1 übertragen. Das am niedrigsten adressierte Byte wird in die Bits 24 bis 31 geschrieben.

Die folgende Tabelle zeigt beispielsweise, wie der Inhalt des Akkumulators 1 beim Laden von Operanden in Byte-, Wort-, und Doppelwortformat verändert wird:

Anweisung	Akkumulator 1							
	31 . . .	. . . 24	23 . . .	. . . 16	15 . . .	. . . 8	7 . . .	. . . 0
Laden eines Bytes: L MB10	0000	0000	0000	0000	0000	0000	<MB10>	

Anweisung	Akkumulator 1					
Laden eines Worts: L MW10	0000	0000	0000	0000	<MB10>	<MB11>
Laden eines Doppelworts: L MD10	<MB10>		<MB11>		<MB12>	<MB13>

Beim Laden des angegebenen Operanden wird der vorherige Inhalt des Akkumulators 1 in den Akkumulator 2 verschoben. Die Anweisung "Laden" überträgt dabei den gesamten Inhalt des Akkumulators 1 zum Akkumulator 2. Der vorherige Inhalt des Akkumulators 2 geht dabei verloren.

Bei Steuerungen mit vier Akkumulatoren werden die Inhalte der Akkumulatoren 3 und 4 durch die Anweisung nicht beeinflusst.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Bitfolgen, Ganzzahlen, Zeiten, REAL, DATE, TOD, CHAR	E, A, PE, M, L, DB, DI, T, Z, Zeiger, Parameter	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" in Akkumulator 1
                             laden.
L "Tag_Value_2"              // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                             schieben.
                             // Wert des Operanden "Tag_Value_2" in Akkumulator 1
                             laden.
*R                           // Werte der Akkumulatoren 1 und 2 multiplizieren.
                             // Produkt im Akkumulator 1 speichern.
L "Tag_Value_3"              // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                             schieben.
                             // Wert des Operanden "Tag_Value_3" in Akkumulator 1
                             laden.
+R                           // Werte der Akkumulatoren 1 und 2 addieren.
                             // Summe im Akkumulator 1 speichern.
T "Tag_Result"               // Ergebnis (Inhalt des Akkumulators 1) in den Operan-
                             den "Tag_Result" übertragen.
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

Programmierbeispiele AWL (Seite 7595)

L STW: Statuswort in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Statuswort in Akkumulator 1 laden" laden Sie das Statuswort in den Akkumulator 1. Bei der Ausführung schreibt die Anweisung die einzelnen Statusbits in das rechte Wort des Akkumulators 1. Die restlichen Bits des Akkumulators 1 werden mit "0" beschrieben.

Die folgende Tabelle zeigt den Inhalt des Akkumulators nach der Ausführung der Anweisung "Statuswort in Akkumulator 1 laden":

	Akkumulator 1									
Bitnummer	31-9	8	7	6	5	4	3	2	1	0
Inhalt	0	BIE	A1	A0	OV	OS	OR	STA	VKE	/ER

Die Anweisung wird unabhängig vom Signalzustand des Statusbits ausgeführt.

Hinweis

Die Steuerungen der S7-300 Baureihe (außer S7-318) laden die Statusbits /ER, STA und OR nicht in den Akkumulator 1. Diese Stellen werden mit "0" beschrieben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L STW`	`// Statuswort in den Akkumulator 1 laden.`
`T "Tag_STW"`	`// Inhalt des Akkumulators 1 (Statusbits) in den Operanden "Tag_STW" übertragen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

LAR1: AR1 mit Inhalt von Akkumulator 1 laden

Beschreibung

Mit der Anweisung "AR1 mit Inhalt von Akkumulator 1 laden" laden Sie das Adressregister 1 (AR1) mit dem Inhalt des Akkumulators 1. Der Wert des Akkumulators 1 muss dem Format eines Bereichszeigers (POINTER) entsprechen.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L P#10.0	// Zeiger (P#10.0) in Akkumulator 1 laden.
LAR1	// Adressregister 1 mit dem Inhalt des Akkumulators 1 laden.
L MW [AR1,P#4.0]	// MW14 in den Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR1,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

LAR1 <D>: AR1 mit Doppelwort oder Bereichszeiger laden

Beschreibung

Mit der Anweisung "AR1 mit Doppelwort oder Bereichszeiger laden" laden Sie das Adressregister 1 (AR1) mit dem Inhalt eines Doppelworts oder eines bereichsinternen oder bereichsübergreifenden Zeigers. Der Inhalt des Doppelworts muss dabei dem Format eines Bereichszeigers entsprechen.

Der Zeiger und das Doppelwort können einen der folgenden Speicherbereiche adressieren:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR1 mit Doppelwort oder Bereichszeiger laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Input	DWORD, POINTER	D, M, L	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#10.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
L MW [AR1,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR1,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE="1") den Operanden "Tag_Output" auf "1" setzen.
LAR1 MD20	// Adressregister 1 mit dem Inhalt von MD20 laden (MD20 = P#30.0).
U E [AR1,P#2.1]	// Signalzustand des Bits E32.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

LAR1 AR2: AR1 mit Inhalt von AR2 laden

Beschreibung

Mit der Anweisung "AR1 mit Inhalt von AR2 laden" laden Sie den Inhalt des Adressregisters 2 (AR2) in das Adressregister 1 (AR1).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
LAR1 AR2	// Inhalt des Adressregisters 2 in das Adressregister 1 kopieren.
L MW [AR1,P#4.0]	// MW24 in Akkumulator 1 laden.
T "Tag_2"	// Transferiere Akkumulator 1 in "Tag_2".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

LAR2: AR2 mit Inhalt von Akkumulator 1 laden

Beschreibung

Mit der Anweisung "AR2 mit Inhalt von Akkumulator 1 laden" laden Sie das Adressregister 2 (AR2) mit dem Inhalt des Akkumulators 1. Der Wert des Akkumulators 1 muss dem Format eines Bereichszeigers (POINTER) entsprechen.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L P#10.0	// Zeiger (P#10.0) in Akkumulator 1 laden.
LAR2	// Adressregister 2 mit dem Inhalt des Akkumulators 1 laden.
L MW [AR2,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR2,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

LAR2 <D>: AR2 mit Doppelwort oder Bereichszeiger laden

Beschreibung

Mit der Anweisung "AR2 mit Doppelwort oder Bereichszeiger laden" laden Sie das Adressregister 2 (AR2) mit dem Inhalt eines Doppelworts oder eines bereichsinternen oder bereichsübergreifenden Zeigers. Der Inhalt des Doppelworts muss dabei dem Format eines Bereichszeigers (POINTER) entsprechen.

Der Zeiger und das Doppelwort können einen der folgenden Speicherbereiche adressieren:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Syntax

Für die Anweisung "AR2 mit Doppelwort oder Bereichszeiger laden" wird die folgende Syntax verwendet:

```
LAR2 <D>
```

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR2 mit Doppelwort oder Bereichszeiger laden":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Input	DWORD, POINTER	D, M, L	Operand, dessen Inhalt geladen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#10.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
L MW [AR2,P#4.0]	// MW14 in Akkumulator 1 laden.
L "Tag_Value"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Inhalt des Operanden "Tag_Value" in Akkumulator 1 laden.

AWL	Erläuterung
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U E [AR2,P#2.1]	// Signalzustand des Bits E12.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output"	// Bei erfüllter Bedingung (VKE = "1") den Operanden "Tag_Output" auf "1" setzen.
LAR2 MD20	// Adressregister 2 mit dem Inhalt von MD20 laden (MD20 = P#30.0).
U E [AR2,P#2.1]	// Signalzustand des Bits E32.1 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

Transferieren

T: Transferieren

Beschreibung

Mit der Anweisung "Transferieren" übertragen Sie den Inhalt des Akkumulators 1 byte-, wort-, oder doppelwortweise in einen angegebenen Operanden. Der Inhalt des Akkumulators 1 wird dabei nicht verändert. Der angegebene Operand muss in einem der folgenden Speicherbereiche adressiert sein:

- Prozessabbild der Eingänge und Ausgänge (E, A)
- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)
- Peripherie (PA)

Die Anzahl der zu übertragenden Bytes hängt vom Format des angegebenen Operanden ab. Wenn der Operand im Byte-Format vorliegt, überträgt die Anweisung den Inhalt der Bits 0 bis 7 des Akkumulators 1.

Wenn der angegebene Operand im Wort-Format vorliegt, überträgt die Anweisung den Inhalt des rechtsbündig im Akkumulator 1 stehenden Worts. Das rechte Byte des Akkumulators 1 wird in das höher adressierte Byte des Operanden übertragen. In das niedriger adressierte Byte des Operanden werden die Bits 8 bis 15 des Akkumulators 1 transferiert.

Wenn der angegebene Operand im Doppelwort-Format vorliegt, überträgt die Anweisung den gesamten Inhalt des Akkumulators 1. Das rechte Byte des Akkumulators 1 wird in das am höchsten adressierte Byte des Operanden übertragen. In das am niedrigsten adressierte Byte des Operanden werden die Bits 24 bis 31 des Akkumulators 1 transferiert.

Die folgende Tabelle zeigt ein Beispiel, wie der Inhalt des Akkumulators 1 byte-, wort- und doppelwortweise transferiert wird:

Anweisung	Akkumulator 1			
Bits	31 24	23 16	15 8	7 0
Inhalt des Akkumulators 1	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)
Transferieren eines Bytes				Byte (n+3)
Transferieren eines Worts			Byte (n+2)	Byte (n+3)
Transferieren eines Doppelworts	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Ausführung der Anweisung "Transferieren" wird durch das Master Control Relay (MCR) beeinflusst. Wenn das MCR-Bit auf "0" gesetzt ist, wird der Wert "0" in den angegebenen Operanden übertragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Output	Bitfolgen, Ganzzahlen, Zeiten, REAL, DATE, TOD, CHAR	E, A, PA, M, L, DB, DI	Operand, in den der Inhalt des Akkumulators 1 übertragen wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result_1"	// Produkt (Inhalt des Akkumulators 1) in den Operanden "Tag_Result_1" übertragen.

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
+R	// Werte der Akkumulatoren 1 und 2 addieren.
	// Summe im Akkumulator 1 speichern.
T "Tag_Result_2"	// Summe (Inhalt des Akkumulators 1) in den Operanden "Tag_Result_2" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Das Master Control Relay (Seite 201)

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

Programmierbeispiele AWL (Seite 7595)

T STW: Akkumulator 1 in Statuswort transferieren

Beschreibung

Mit der Anweisung "Akkumulator 1 in Statuswort transferieren" übertragen Sie die Bits 0 bis 8 des Akkumulators 1 in das Statuswort.

Die folgende Tabelle zeigt die Bits im Akkumulator 1, mit denen die Statusbits nach der Ausführung der Anweisung überschrieben werden:

	Akkumulator 1								
Bitnummer	8	7	6	5	4	3	2	1	0
Statusbits	BIE	A1	A0	OV	OS	OR	STA	VKE	/ER

Die Anweisung wird unabhängig vom Signalzustand des Statusbits ausgeführt.

Hinweis

Die Bits des Statusworts /ER, STA und OR werden nicht durch die Anweisung "Akkumulator 1 in Statuswort transferieren" geladen. Lediglich Bit 1, 4, 5, 6, 7 und 8 werden an die entsprechenden Bitpositionen des niederwertigen Worts des Akkumulators 1 geladen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_STW"	// Wert des Operanden "Tag_STW" in den Akkumulator 1 laden.
T STW	// Bit 0 bis 8 des Akkumulators 1 in das Statuswort übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR: AR1 und AR2 tauschen

Beschreibung

Mir der Anweisung "AR1 und AR2 tauschen" tauschen Sie die Inhalte der Adressregister 1 (AR1) und 2 (AR2).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#10.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.0 laden.
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
TAR	// Inhalt der Adressregister tauschen.
L MD [AR1,P#2.0]	// MD22 in Akkumulator 1 laden.
T MD [AR2,P#2.0]	// MD12 in Akkumulator 2 transferieren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR1: AR1 in Akkumulator 1 transferieren

Beschreibung

Mit der Anweisung "AR1 in Akkumulator 1 transferieren" übertragen Sie den Inhalt des Adressregisters 1 (AR1) in den Akkumulator 1. Der Inhalt des Akkumulators 1 wird dabei in den Akkumulator 2 verschoben.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR1 P#15.0	// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#15.0 laden.
TAR1	// Inhalt des Adressregisters 1 (P#15.0) in den Akkumulator 1 transferieren.
T "Tag_Pointer"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Pointer" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR1 <D>: AR1 in Doppelwort transferieren

Beschreibung

Mit der Anweisung "AR1 in Doppelwort transferieren" übertragen Sie den Inhalt des Adressregisters 1 (AR1) in ein Doppelwort. Das Doppelwort muss dabei in einem der folgenden Speicherbereiche adressiert sein:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR1 in Doppelwort transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Output	DWORD	D, M, L	Operand, in welchen der Inhalt des Adressregisters 1 transferiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
TAR1 %DBD20	// Inhalt des Adressregisters 1 in das Datendoppelwort DBD20 übertragen.
TAR1 %DID30	// Inhalt des Adressregisters 1 in das Instanzdoppelwort DID30 übertragen.
TAR1 %LD18	// Inhalt des Adressregisters 1 in das Lokaldaten-Doppelwort LD18 übertragen.
TAR1 %MD24	// Inhalt des Adressregisters 1 in das Merkerdoppelwort MD24 übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR1 AR2: AR1 in AR2 transferieren

Beschreibung

Mit der Anweisung "AR1 in AR2 transferieren" kopieren Sie den Inhalt des Adressregisters 1 (AR1) in das Adressregister 2 (AR2).

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`LAR1 P#10.4`	`// Adressregister 1 mit dem Inhalt des bereichsinternen Zeigers P#10.4 laden.`
`TAR1 AR2`	`// Inhalt des Adressregisters 1 in das Adressregister 2 transferieren.`
`U E [AR2,P#3.2]`	`// Signalzustand des Bits E13.6 auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`= "Tag1"`	`// Operanden "Tag_1" auf "1" setzen.`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR2: AR2 in Akkumulator 1 transferieren

Beschreibung

Mit der Anweisung "AR2 in Akkumulator 1 transferieren" übertragen Sie den Inhalt des Adressregisters 2 (AR2) in den Akkumulator 1. Der Inhalt des Akkumulators 1 wird dabei in den Akkumulator 2 verschoben.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
LAR2 P#20.0	// Adressregister 2 mit dem Inhalt des bereichsinternen Zeigers P#20.0 laden.
TAR2	// Inhalt des Adressregisters 2 (P#20.0) in den Akkumulator 1 transferieren.
T "Tag_Pointer"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Pointer" übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAR2 <D>: AR2 in Doppelwort transferieren

Beschreibung

Mit der Anweisung "AR2 in Doppelwort transferieren" übertragen Sie den Inhalt des Adressregisters 2 (AR2) in ein Doppelwort. Das Doppelwort muss dabei in einem der folgenden Speicherbereiche adressiert sein:

- Merker (M)
- Temporäre Lokaldaten (L)
- Datenbausteine (DB, DI)

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR2 in Doppelwort transferieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<D>	Output	DWORD	D, M, L	Operand, in welchen der Inhalt des Adressregisters 2 transferiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
TAR2 %DBD20	// Inhalt des Adressregisters 2 in das Datendoppelwort DBD20 übertragen.
TAR2 %DID30	// Inhalt des Adressregisters 2 in das Instanzdoppelwort DID30 übertragen.
TAR2 %LD18	// Inhalt des Adressregisters 2 in das Lokaldaten-Doppelwort LD18 übertragen.
TAR2 %MD24	// Inhalt des Adressregisters 2 in das Merkerdoppelwort MD24 übertragen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

Umwandler

BTI: BCD in Ganzzahl (16 Bit) umwandeln

Beschreibung

Mit der Anweisung "BCD in Ganzzahl (16 Bit) umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine 16-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine dreistellige binär codierte Dezimalzahl (BCD).

Die Bits 1 bis 11 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-999" bis "+999".

Das Vorzeichen des Ergebniswerts wird aus dem Bit 15 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" des Bits ist das Vorzeichen positiv. Wenn der Signalzustand des Bits "1" ist, ist das Vorzeichen negativ. Die Bits 12 bis 14 werden bei der Umwandlung nicht verwendet.

Wenn im BCD-Code ein BCD-Fehler enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in STOP.

Das Ergebnis der Umwandlung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// BCD in Akkumulator 1 laden.
BTI	// BCD in eine 16-Bit Ganzzahl umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

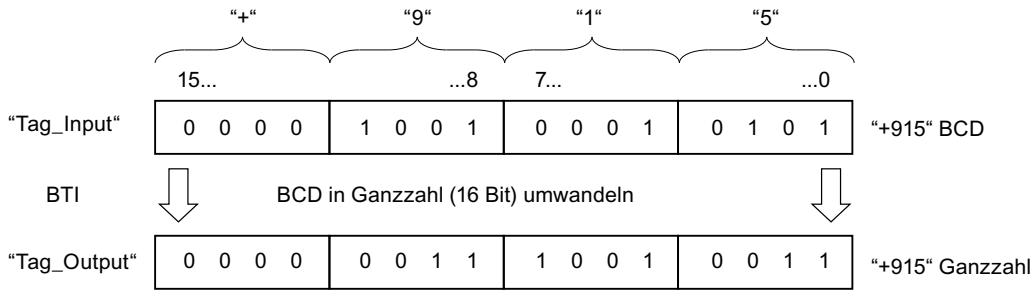

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

ITB: Ganzzahl (16 Bit) in BCD umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) in BCD umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine dreistellige binär codierte Dezimalzahl (BCD) um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 16-Bit Ganzzahl.

Die Bits 0 bis 11 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-999" bis "+999". Wenn der umzuwandelnde Wert außerhalb dieses Bereichs liegt, werden die Bits OV und OS auf den Signalzustand "1" gesetzt. Die Umwandlung wird in diesem Fall nicht durchgeführt.

Das Vorzeichen des Ergebniswerts wird aus den Bits 12 bis 15 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" der Bits ist das Vorzeichen positiv. Wenn der Signalzustand aller vier Bits "1" ist, ist das Vorzeichen negativ.

Das Ergebnis der Umwandlung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// Ganzzahl (16 Bit) in Akkumulator 1 laden.
ITB	// Ganzzahl in BCD umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

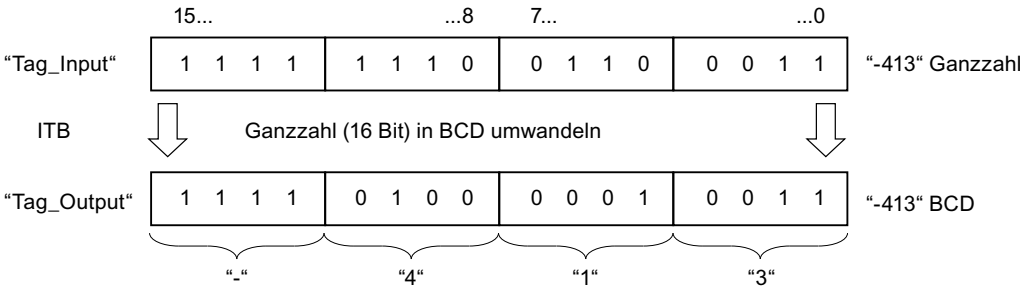

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

BTD: BCD in Ganzzahl (32 Bit) umwandeln

Beschreibung

Mit der Anweisung "BCD in Ganzzahl (32 Bit) umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine siebenstellige binär codierte Dezimalzahl (BCD).

Die Bits 0 bis 27 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von "-9999999" bis "+9999999".

Das Vorzeichen des Ergebniswerts wird aus dem Bit 31 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" des Bits ist das Vorzeichen positiv. Wenn der Signalzustand des Bits "1" ist, ist das Vorzeichen negativ. Die Bits 28 bis 30 werden bei der Umwandlung nicht verwendet.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Wenn im BCD-Code eine Pseudotetrade enthalten ist, meldet die CPU einen Programmierfehler und ruft den Organisationsbaustein "OB121" auf. Wenn der Organisationsbaustein OB121 nicht vorhanden ist, geht die CPU in den Stoppzustand.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// BCD in Akkumulator 1 laden.`
`BTD`	`// BCD wird in eine 32-Bit Ganzzahl umgewandelt.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

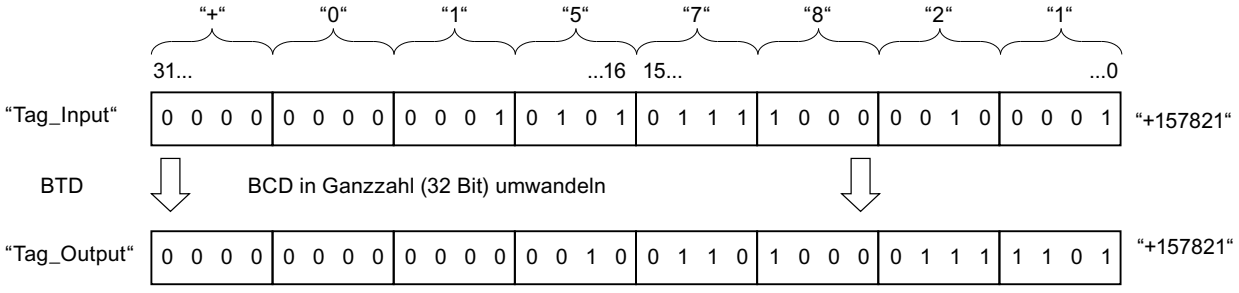

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

ITD: Ganzzahl (16 Bit) in Ganzzahl (32 Bit) umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) in Ganzzahl (32 Bit) umwandeln" wandeln Sie den im rechten Wort des Akkumulators 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 16-Bit Ganzzahl.

Bei der Umwandlung werden die Bits 0 bis 15 unverändert übernommen. Die Bits 16 bis 31 des Ergebniswerts werden mit dem Signalzustand des Bits 15 aufgefüllt.

Der Ergebniswert wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// Ganzzahl (16 Bit) in Akkumulator 1 laden .
ITD	// 16-Bit Ganzzahl in eine 32-Bit Ganzzahl umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Akkumulator 1							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Vor der Ausführung	XXXX	XXXX	XXXX	XXXX	1111	1111	1111	0110
Nach der Ausführung	1111	1111	1111	1111	1111	1111	1111	0110
	X : Der Signalzustand der Bits ist für die Umwandlung irrelevant							

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

DTB: Ganzzahl (32 Bit) in BCD umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) in BCD umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine siebenstellige binär codierte Dezimalzahl (BCD) um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 32-Bit Ganzzahl.

Die Bits 0 bis 27 des Akkumulators 1 geben den Wert an, der umgewandelt wird. Zulässig sind die Werte im Bereich von"-9999999" und "+9999999". Wenn der umzuwandelnde Wert außerhalb dieses Bereichs liegt, werden die Bits OV und OS auf den Signalzustand "1" gesetzt. Die Umwandlung wird in diesem Fall nicht durchgeführt.

Das Vorzeichen des Ergebniswerts wird aus den Bits 28 bis 31 des Akkumulators 1 gelesen. Bei einem Signalzustand "0" der Bits ist das Vorzeichen positiv. Wenn der Signalzustand aller vier Bits "1" ist, ist das Vorzeichen negativ.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// 32-Bit Ganzzahl in Akkumulator 1 laden.
DTB	// 32-Bit Ganzzahl in BCD umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

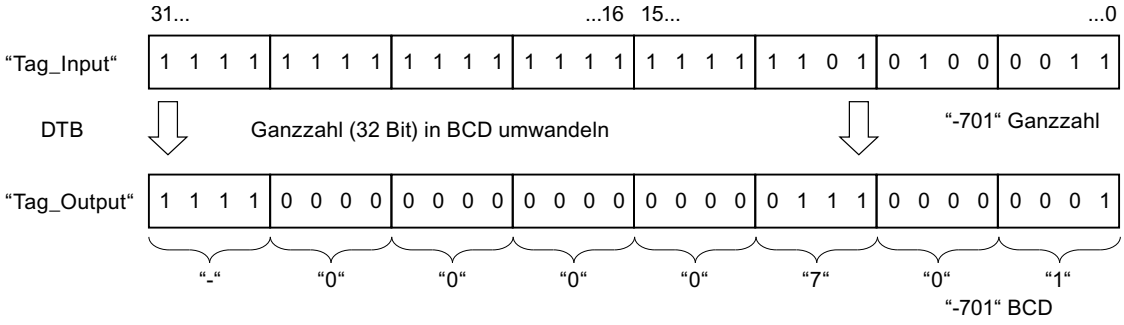

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

DTR: Ganzzahl (32 Bit) in Gleitpunktzahl umwandeln

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) in Gleitpunktzahl umwandeln" wandeln Sie den im Akkumulator 1 stehenden Wert in eine Gleitpunktzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine 32-Bit Ganzzahl.

Da eine 32-Bit Ganzzahl eine höhere Genauigkeit als eine Gleitpunktzahl hat, wird das Ergebnis der Umwandlung zur nächsten ganzen Zahl gerundet.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input"	// 32-Bit Ganzzahl in Akkumulator 1 laden.
DTR	// 32-Bit Ganzzahl in eine Gleitpunktzahl umwandeln.
T "Tag_Output"	// Ergebnis in den Operanden "Tag_Output" transferieren.

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

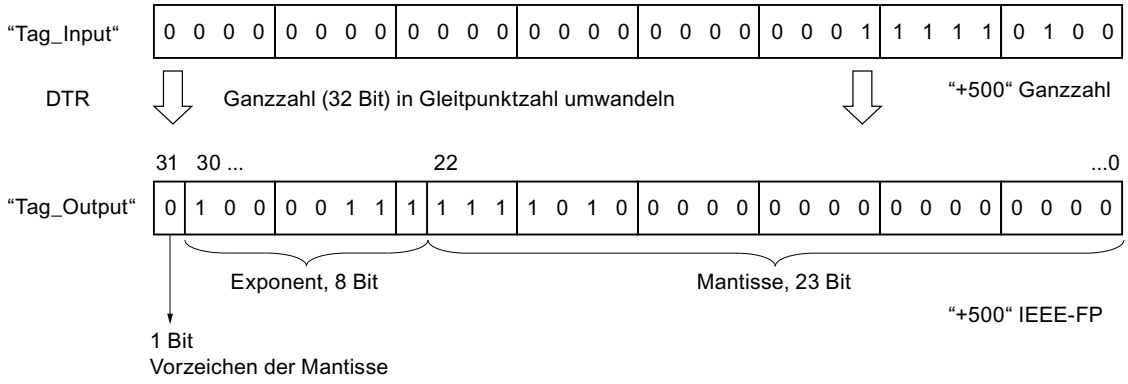

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

INVI: Einerkomplement zu Ganzzahl (16 Bit) erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement zu Ganzzahl (16-Bit) erzeugen" negieren Sie den im rechten Wort des Akkumulators 1 stehenden Wert Bit für Bit.

Bei der Ausführung invertiert die Anweisung den Signalzustand jedes einzelnen Bits im rechten Wort des Akkumulators 1. Die Einsen werden durch Nullen ersetzt und die Nullen durch Einsen.

Das Ergebnis wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Input"               // Wert in Akkumulator 1 laden.
INVI                        // Einerkomplement erzeugen
T "Tag_Output"              // Ergebnis in den Operanden "Tag_Output" transferie-
                               ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	0011	1010	1110
Tag_Output	1001	1100	0101	0001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

INVD: Einerkomplement zu Ganzzahl (32 Bit) erzeugen

Beschreibung

Mit der Anweisung "Einerkomplement zu Ganzzahl (32-Bit) erzeugen" negieren Sie den im Akkumulator 1 stehenden Wert Bit für Bit.

Bei der Ausführung invertiert die Anweisung den Signalzustand jedes einzelnen Bits im Akkumulator 1. Die Einsen werden durch Nullen ersetzt und die Nullen durch Einsen.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Input"               // Wert in Akkumulator 1 laden.
INVD                        // Einerkomplement erzeugen
T "Tag_Output"              // Ergebnis in den Operanden "Tag_Output" transferie-
                               ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	1111	1000	1100	0110	0011	1010	1110
Tag_Output	1001	0000	0111	0011	1001	1100	0101	0001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

NEGI: Ganzzahl (16 Bit) negieren

Beschreibung

Mit der Anweisung "Ganzzahl (16 Bit) negieren" wechseln Sie das Vorzeichen des im rechten Wort des Akkumulators 1 stehenden Werts durch Zweierkomplementbildung. Den Wert des Akkumulators 1 interpretiert die Anweisung als 16-Bit Ganzzahl.

Die Ausführung der Anweisung ist gleichbedeutend mit einer Multiplikation mit "-1". Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Der Inhalt des linken Worts im Akkumulator 1 wird durch die Anweisung nicht beeinflusst.

Die Anweisung "Ganzzahl (16 Bit) negieren" beeinflusst die Statusbits A0, A1, OV und OS. Die folgende Tabelle zeigt, wie die Anweisung die Statusbit abhängig vom Ergebnis beeinflusst:

Ergebnis:	A1	A0	OV	OS
+1 bis +32767	1	0	0	-
0	0	0	0	-
-1 bis -32767	0	1	0	-
(-) 32768	0	1	1	1

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                     Erläuterung
L "Tag_Input"           // Wert in Akkumulator 1 laden.
NEGI                    // 16-Bit Ganzzahl negieren
T "Tag_Output"          // Ergebnis in den Operanden "Tag_Output" transferie-
                        ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1101	0011	1000
Tag_Output	1010	0010	1100	1000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

NEGD: Ganzzahl (32 Bit) negieren

Beschreibung

Mit der Anweisung "Ganzzahl (32 Bit) negieren" wechseln Sie das Vorzeichen des im Akkumulator 1 stehenden Werts durch Zweierkomplementbildung. Den Wert des Akkumulators 1 interpretiert die Anweisung als 32-Bit Ganzzahl.

Die Ausführung der Anweisung ist gleichbedeutend mit einer Multiplikation mit "-1". Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Anweisung "Ganzzahl (32 Bit) negieren" beeinflusst die Statusbits A0, A1, OV und OS. Die folgende Tabelle zeigt, wie die Anweisung die Statusbit abhängig vom Ergebnis beeinflusst:

Ergebnis	A1	A0	OV	OS
+1 bis +2147483647	1	0	0	-
0	0	0	0	-
-1 bis -2147483647	0	1	0	-
(-) 2147483648	0	1	1	1

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                     Erläuterung
L "Tag_Input"           // Wert in Akkumulator 1 laden.
NEGD                    // 32-Bit Ganzzahl negieren.
T "Tag_Output"          // Ergebnis in den Operanden "Tag_Output" transferie-
                        ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Status	Wert							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1111	0110	0100	0101	1101	0011	1000
Tag_Output	1010	0000	1001	1011	1010	0010	1100	1000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen mit Ganzzahlen (Seite 188)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

NEGR: Gleitpunktzahl negieren

Beschreibung

Mit der Anweisung "Gleitpunktzahl negieren" invertieren Sie den Signalzustand des Bits 31 im Akkumulator 1. Den Wert des Akkumulators 1 interpretiert die Anweisung als eine Gleitpunktzahl. Im Bit 31 der Gleitpunktzahl steht das Vorzeichen der Mantisse.

Die Ausführung der Anweisung ist gleichbedeutend mit einer Multiplikation mit "-1". Nach der Ausführung der Anweisung wird das Vorzeichen der Mantisse invertiert.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Anweisung beeinflusst keine Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Input"               // Gleitpunktzahl in Akkumulator 1 laden.
NEGR                        // Gleitpunktzahl negieren.
T "Tag_Output"              // Ergebnis in den Operanden "Tag_Output" transferie-
                               ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Input	1.5E+02
Tag_Output	-1.5E+02

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

TAW: Bytes im rechten Wort des Akkumulators 1 tauschen

Beschreibung

Mit der Anweisung "Bytes im rechten Wort des Akkumulators 1 tauschen" tauschen Sie die Reihenfolge der beiden rechten Bytes im rechten Wort des Akkumulators 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert B	Wert D	Wert C

Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Die Bytes im linken Wort des Akkumulators 1 werden durch die Anweisung nicht beeinflusst und bleiben unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Input"               // Wert in Akkumulator laden.
TAW                         // Bytes im rechten Wort des Akkumulators 1 tauschen.
T "Tag_Output"              // Ergebnis in den Operanden "Tag_Output" transferie-
                               ren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Input	0000	1111	0000	1111
Tag_Output	0000	1111	1111	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

TAD: Bytes im gesamten Akkumulator 1 tauschen

Beschreibung

Mit der Anweisung "Bytes im gesamten Akkumulator 1 tauschen" tauschen Sie die Reihenfolge der Bytes im Akkumulator 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert D	Wert C	Wert B	Wert A

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Wert des Operanden in Akkumulator 1 laden.`
`TAD`	`// Bytes des Akkumulators 1 tauschen.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Input	1111	0000	0000	1111	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000	0000	1111	1111	0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

RND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächstliegenden Ganzzahl.

Wenn die Gleitpunktzahl genau zwischen gerader und ungerader Zahl liegt, wird die gerade Zahl als Ergebnis gewählt.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND`	`// Gleitpunktzahl zu einer Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	101.5	-101.5
Tag_Output	102	-102

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl umwandeln. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und liefert deren ganzzahligen Anteil als Ergebnis zurück.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`TRUNC`	`// Gleitpunktzahl in eine Ganzzahl vom Datentyp DINT umwandeln.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	101.5	-101.5
Tag_Output	101	-101

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

RND+: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächst höheren Ganzzahl. Das Ergebnis der Anweisung ist eine Zahl vom Datentyp DINT, die größer oder gleich der umzuwandelnden Gleitpunktzahl ist.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND+`	`// Gleitpunktzahl zur nächst höheren Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	100.5	-100.5
Tag_Output	101	-100

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

RND-: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" wandeln Sie den im Akkumulator 1 stehenden Wert in eine 32-Bit Ganzzahl um. Den umzuwandelnden Wert interpretiert die Anweisung als eine Gleitpunktzahl und rundet diese zur nächst niederen Ganzzahl. Das Ergebnis der Anweisung ist eine Zahl vom Datentyp DINT, die kleiner oder gleich der umzuwandelnden Gleitpunktzahl ist.

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die Umwandlung wird nicht durchgeführt und die Statusbits OV und OS werden gesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der im Akkumulator 1 stehende Wert ist keine gültige Gleitpunktzahl.
- Das Ergebnis liegt außerhalb des Bereichs, der für eine Ganzzahl vom Datentyp DINT zulässig ist.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Input"`	`// Gleitpunktzahl in Akkumulator 1 laden.`
`RND-`	`// Gleitpunktzahl zur nächst niederen Ganzzahl runden.`
`T "Tag_Output"`	`// Ergebnis in den Operanden "Tag_Output" transferieren.`

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte	
Tag_Input	100.5	-100.5
Tag_Output	100	-101

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Anweisungen mit Gleitpunktzahlen (Seite 189)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmsteuerung

Sprünge

Sprungmarke

Beschreibung

Mit einer Sprungmarke können Sie die Programmstelle kennzeichnen, an der bei einem ausgeführten Sprung die Programmbearbeitung fortgesetzt wird. Die Bezeichnung der Sprungmarke kann aus bis zu 128 Buchstaben, Ziffern oder Unterstrichen bestehen.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

Es kann sowohl vorwärts als auch rückwärts gesprungen werden. Die maximale Sprungweite liegt bei -32768 bzw. +32767 Wörtern des Programmcodes. Die tatsächliche maximale Anzahl der Anweisungen, die übersprungen werden können, hängt von der Kombination der Anweisungen innerhalb des Programms (Ein-, Zwei- oder Dreiwortanweisungen) ab.

Wenn Sie die MCR-Funktionalität verwenden, muss die Sprungmarke im gleichen MCR-Bereich wie die Sprunganweisung liegen.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL	// Bei einem Abfrageergebnis "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	//Bei einem Abfrageergebnis "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.

AWL	Erläuterung
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" laden.
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Operanden "Tag_Input_4" auf "1" abfragen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPA: Absolut springen

Beschreibung

Mit der Anweisung "Absolut springen" unterbrechen Sie die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Die Anweisung wird immer, unabhängig von Bedingungen ausgeführt.

Die Anweisung "Absolut springen" beeinflusst die Statusbits nicht.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolut springen":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL_1	// Bei einem Abfrageergebnis "1" zur Sprungmarke "MyLABEL_1" springen und die Programmbearbeitung dort fortsetzen.
	//Bei einem Abfrageergebnis "0" lineare Programmbearbeitung fortsetzen.

AWL	Erläuterung
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
SPA MyLABEL_2	// Zur Sprungmarke "MyLABEL_2" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL_1: L "Tag_Input_4"	// Sprungmarke
	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.
MyLABEL_2: U "Tag_Input_5"	// Sprungmarke "MyLABEL_2"
	// Operanden "Tag_Input_5" auf "1" abfragen.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

SPB: Springen bei VKE = 1

Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie, abhängig vom Verknüpfungsergebnis, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "1" ist. Wenn das aktuelle VKE vor der Anweisung "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei VKE = 1" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE und das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPB MyLABEL	// Bei VKE = "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
MyLABEL: L "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" transferieren.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

SPBN: Springen bei VKE = 0

Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "0" ist. Wenn das aktuelle VKE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei VKE = 0" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE und das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 0":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Operanden "Tag_Input_1" auf "1" abfragen und nach UND verknüpfen.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und nach UND verknüpfen.
SPBN MyLABEL	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "1" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" laden.
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Den Operanden "Tag_Input_4" auf "1" abfragen und mit dem VKE nach UND verknüpfen.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

SPBB: Springen bei VKE = 1 und VKE speichern

Beschreibung

Mit der Anweisung "Springen bei VKE = 1 und VKE speichern" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Parallel dazu wird der Signalzustand des aktuellen VKE in das Binärergebnis (BIE) kopiert.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE 1 ist. In diesem Fall setzt die Anweisung das Binärergebnis auf "1".

Wenn das aktuelle VKE vor der Anweisung "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt. In diesem Fall weist die Anweisung dem Binärergebnis den Signalzustand "0" zu.

Die Anweisung "Springen bei VKE = 1 und VKE speichern" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE und das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 1 und VKE speichern":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Input_1"	// Ersten Vergleichswert laden.
L "Tag_Input_2"	// Zweiten Vergleichswert laden.
>I	// Abfrage, ob der Wert des Operanden "Tag_Input_1" größer als der Wert des Operanden "Tag_Input_2" ist.
SPBB MyLABEL	// Aktuelles VKE in BIE kopieren.
	// Bei VKE = "1" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	//Bei VKE = "0" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output_1"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_1" laden.
MyLABEL: L "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Inhalt des Operanden "Tag_Input_4" in Akkumulator 1 laden.
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SPBNB: Springen bei VKE = 0 und VKE speichern

Beschreibung

Mit der Anweisung "Springen bei VKE = 0 und VKE speichern" unterbrechen Sie, abhängig vom Verknüpfungsergebnis (VKE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist. Parallel dazu wird der Signalzustand des aktuellen Verknüpfungsergebnisses in das Binärergebnis (BIE) kopiert.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das aktuelle VKE "0" ist. In diesem Fall setzt die Anweisung das Binärergebnis auf "0".

Wenn das aktuelle VKE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt. In diesem Fall weist die Anweisung dem Binärergebnis den Signalzustand "1" zu.

Die Anweisung "Springen bei VKE = 0 und VKE speichern" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das VKE und das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei VKE = 0 und VKE speichern":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
U "Tag_Input_1"	// Operanden "Tag_Input_1" auf "1" abfragen und nach UND verknüpfen.
U "Tag_Input_2"	// Operanden "Tag_Input_2" auf "1" abfragen und nach UND verknüpfen.
SPBNB MyLABEL	// Aktuelles VKE in BIE kopieren.
	// Bei VKE = "0" zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei VKE = "1" nächstfolgende Anweisung bearbeiten.
L "Tag_Input_3"	// Inhalt des Operanden "Tag_Input_3" in Akkumulator 1 laden.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" transferieren.
MyLABEL: U "Tag_Input_4"	// Programmbearbeitung bei einem ausgeführten Sprung an dieser Stelle fortsetzen.
	// Den Operanden "Tag_Input_4" auf "1" abfragen und mit dem VKE nach UND verknüpfen.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SPBI: Springen bei BIE = 1

Beschreibung

Mit der Anweisung "Springen bei BIE = 1" unterbrechen Sie abhängig vom Binärergebnis die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit BIE "1" ist. Wenn das Statusbit BIE "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei BIE = 1" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei BIE = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
SET                          // VKE am Anfang des Bausteins auf "1" setzen.
SAVE                         // Signalzustand des VKE in das BIE-Bit übertragen.
....                         // Beliebiges Programm
UN OV                        // OV-Bit am Ende des Bausteins abfragen.
SAVE                         // Bei einem Überlauf BIE-Bit auf "0" setzen.
                             // BIE auf "1" setzen, wenn kein Überlauf vorliegt.
SPBI END                     // Bei BIE = "1" zur Sprungmarke END springen.
                             // Bei BIE = "0" die nächstfolgende Anweisung bearbei-
                             ten.
R "Tag_Output_1"             // Operanden "Tag_Output_1" auf "0" zurücksetzen.
END: S "Tag_Output_2"        // Operanden "Tag_Output_2" auf "1" setzen.
```

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPBIN: Springen bei BIE = 0

Beschreibung

Mit der Anweisung "Springen bei BIE = 0" unterbrechen Sie, abhängig vom Binärergebnis (BIE), die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit BIE "0" ist. Wenn das Statusbit BIE vor der Anweisung "1" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Die Anweisung "Springen bei BIE = 0" setzt sowohl bei erfüllter als auch bei nicht erfüllter Bedingung das Statusbit STA auf "1" und die Statusbits OR und /ER auf "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei BIE = 0":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
SET                         // VKE am Anfang des Bausteins auf "1" setzen.
SAVE                        // Signalzustand des VKE in das BIE-Bit übertragen.
....                        // Beliebiges Programm
UN OV                       // OV-Bit am Ende des Bausteins abfragen.
SAVE                        // Bei einem Überlauf BIE-Bit auf "0" setzen.
                            // BIE auf "1" setzen, wenn kein Überlauf vorliegt.
SPBIN END                   // Bei BIE = "0" zur Sprungmarke END springen.
                            // Bei BIE = "1" die nächstfolgende Anweisung bearbei-
                            ten.
```

AWL	Erläuterung
S "Tag_Output"	// Operanden "Tag_Output" auf "1" setzen.
END: BE	// Baustein beenden

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPO: Springen bei OV = 1

Beschreibung

Mit der Anweisung "Springen bei OV = 1" unterbrechen Sie, abhängig vom Statusbit OV, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit OV "1" ist. Das Statusbit OV wird gesetzt, wenn z. B. das Ergebnis einer Berechnung außerhalb des zulässigen Bereichs liegt oder beim Vergleichen von Gleitpunktzahlen ein ungültiger Wert vorliegt.

Wenn das Statusbit OV "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei OV = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert laden
L "Tag_Value_2"	// Wert laden
*I	// Werte multiplizieren

AWL	Erläuterung
SPO OVER	// Bei einem Signalzustand "1" des Statusbits OV zur Sprungmarke "OVER" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem Signalzustand "0" des Statusbits OV nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Ergebnis der Multiplikation in den Operanden "Tag_Result" übertragen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
OVER: SET	// Programmbearbeitung bei erfüllter Bedingung (OV = "1")an dieser Stelle fortsetzen.
	// VKE auf den Signalzustand "1" setzen.
R "Tag_Output"	// Operanden "Tag_Output" auf "0" zurücksetzen.
NEXT: U "MyTag_1"	// Sprungmarke "NEXT"
	// Operanden "MyTag_1" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_2"	// Operanden "MyTag_2" auf "1" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = 1 den Operanden "Tag_Output_2" auf "1" setzen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPS: Springen bei OS = 1

Beschreibung

Mit der Anweisung "Springen bei OS = 1" unterbrechen Sie, abhängig vom Statusbit OS, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn das Statusbit OS "1" ist. Das Statusbit OS wird immer gesetzt, wenn ein Zahlenbereichsüberlauf das Statusbit OV auf "1" setzt. Im Gegensatz zu Statusbit OV bleibt das Statusbit OS gesetzt, auch wenn anschließend ein Ergebnis im erlaubten Zahlenbereich liegt.

Wenn das Statusbit OS "0" ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei OS = 1":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert laden
L "Tag_Value_2"              // Wert laden
*I                           // Geladene Werte multiplizieren.
                             // Produkt im Akkumulator 1 speichern.
L "Tag_Value_3"              // Wert laden
+I                           // Geladenen Wert mit dem Produkt addieren.
                             // Produkt im Akkumulator 1 speichern.
L "Tag_Value_4"              // Wert laden
-I                           // Geladenen Wert von der berechneten Summe subtrahie-
                             ren.
                             // Produkt im Akkumulator 1 speichern.
SPS OVER                     // Bei einem Überlauf in einer der drei vorangehenden
                             Anweisungen wird das Statusbit OS auf "1" gesetzt.
                             // Bei einem Signalzustand "1" des Statusbits OS wird
                             die Programmbearbeitung an der Stelle der Sprungmarke
                             "OVER" fortsetzt.
                             // Bei einem Signalzustand "0" des Statusbits OS wird
                             die nächstfolgende Anweisung bearbeitet.
T "Tag_Result"               // Ergebnis der gesamten Berechnung in den Operanden
                             "Tag_Result" übertragen.
SPA NEXT                     // Zur Sprungmarke "NEXT" springen und die Programmbe-
                             arbeitung dort fortsetzen.
OVER: SET                    // Programmbearbeitung bei erfüllter Bedingung (OV =
                             "1")an dieser Stelle fortsetzen.
                             // VKE auf den Signalzustand "1" setzen.
R "Tag_Output"               // Operanden "Tag_Output" auf "0" zurücksetzen.
NEXT: U "MyTag_1"            // Sprungmarke "NEXT"
                             // Operanden "MyTag_1" auf "1" abfragen und nach UND
                             verknüpfen.
U "MyTag_2"                  // Operanden "MyTag_2" auf "1" abfragen und nach UND
                             verknüpfen.
S "Tag_Output_2"             // Bei VKE = 1 den Operanden "Tag_Output_2" auf "1"
                             setzen.
```

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPZ: Springen bei Ergebnis Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf hat der Akkumulator 1 den Wert "0".
- Bei der Ausführung der Anweisung "Ganzahlen (16Bit) addieren" (+I) oder "Ganzahlen (32Bit) addieren" (+D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Der Inhalt des Akkumulators 2 ist nach der Ausführung einer Vergleichsanweisung gleich dem Inhalt des Akkumulators 1.
- Der Inhalt des Akkumulators 1 ist nach der Ausführung einer Wortverknüpfung Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "0".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden "Tag_Value" laden.
SRW 1	// Inhalt des rechten Worts im Akkumulator 1 um eine Stelle nach rechts schieben.
SPZ ZERO	// Bei einem Signalzustand "0" des zuletzt hinausgeschobenen Bits zur Sprungmarke "OVER" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem Signalzustand "1" des zuletzt hinausgeschobenen Bits nächstfolgende Anweisung bearbeiten.
L "MyTag_1"	// Wert des Operanden "MyTag_1" in Akkumulator 1 laden.
INC 1	// Rechtes Byte des Akkumulators 1 mit dem Wert "1" addieren.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
ZERO: L "MyTag_2"	// Sprungmarke "ZERO"
	// Wert des Operanden "MyTag_2" in Akkumulator 1 laden.
INC 1	// Rechtes Byte des Akkumulators 1 mit dem Wert "1" addieren.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPN: Springen bei Ergebnis nicht Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis nicht Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 unterschiedlich ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 ungleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, *I, +D, -D, *D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, *I, /I, +D, -D, *D, /D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven oder negativen Bereich auf.
- Der Inhalt des Akkumulators 2 ist nach der Ausführung einer Vergleichsanweisung ungleich dem Inhalt des Akkumulators 1.
- Der Inhalt des Akkumulators 1 ist nach der Ausführung einer Wortverknüpfung nicht Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "1".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis nicht Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Valu_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
XOW                          // Nach EXKLUSIV ODER verknüpfen
```

AWL	Erläuterung
SPN NOZERO	// Wenn der Wert im Akkumulator 1 ungleich Null ist, wird der Sprung zur Sprungmarke "NOZERO" ausgeführt und die Programmbearbeitung dort fortsetzt.
	// Wenn der Wert im Akkumulator 1 gleich Null ist, wird die nächstfolgende Anweisung bearbeitet.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
NOZERO: UN "MyTag_2"	// Sprungmarke "NOZERO"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei einem VKE "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPP: Springen bei Ergebnis größer Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis größer Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "0" und des Statusbits A1 "1" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 größer Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, /I, *D, /D) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven Bereich auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 größer als der Inhalt des Akkumulators 1.
- Nach der Ausführung einer Wortverknüpfung ist der Inhalt des Akkumulators 1 nicht Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits "1".

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis größer Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
-I	// Wert des Operanden "Tag_Value_2" vom Wert des Operanden "Tag_Value_1" subtrahieren.
SPP POSITIVE	// Bei einem positiven Ergebniswert im Akkumulator 1 zur Sprungmarke "POSITIVE" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem negativen Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.

AWL	Erläuterung
POSITIVE: UN "MyTag_2"	// Sprungmarke "POSITIVE"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPM: Springen bei Ergebnis kleiner Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis kleiner Null" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "1" und des Statusbits A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 kleiner Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, *D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im negativen Bereich auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 kleiner als der Inhalt des Akkumulators 1.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis kleiner Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
-I                           // Wert des Operanden "Tag_Value_2" vom Wert des Ope-
                             randen "Tag_Value_1" subtrahieren.
SPM NEGATIVE                 // Bei einem negativen Ergebniswert im Akkumulator 1
                             zur Sprungmarke "NEGATIVE" springen und die Programm-
                             bearbeitung dort fortsetzen.
                             // Bei einem positiven Ergebniswert im Akkumulator 1
                             nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"                 // Operanden "MyTag_1" auf "0" abfragen und nach UND
                             verknüpfen.
S "Tag_Output_1"             // Bei VKE = "1" den Operanden "Tag_Output_1" auf "1"
                             setzen.
SPA NEXT                     // Zur Sprungmarke "NEXT" springen und die Programmbe-
                             arbeitung dort fortsetzen.
NEGATIVE: UN "MyTag_2"       // Sprungmarke "NEGATIVE"
                             // Operanden "MyTag_2" auf "0" abfragen und nach UND
                             verknüpfen.
S "Tag_Output_2"             // Bei VKE = "1" den Operanden "Tag_Output_2" auf "1"
                             setzen.
NEXT: U "MyTag_3"            // Sprungmarke "NEXT"
                             // Operanden "MyTag_3" auf "1" abfragen und nach UND
                             verknüpfen.
U "MyTag_4"                  // Operanden "MyTag_4" auf "1" abfragen und nach UND
                             verknüpfen.
```

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPPZ: Springen bei Ergebnis größer oder gleich Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis größer oder gleich Null" unterbrechen Sie, abhängig vom Statusbit A0, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A0 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 größer oder gleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D) tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (*I, /I, *D, /D) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 größer oder gleich dem Inhalt des Akkumulators 1.
- Eine Anweisung für Wortverknüpfung wurde ausgeführt.
- Eine Schiebeanweisung wurde ausgeführt.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis größer oder gleich Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"              // Wert des Operanden "Tag_Value_2" laden.
-I                           // Wert des Operanden "Tag_Value_2" vom Wert des Ope-
                             randen "Tag_Value_1" subtrahieren.
```

AWL	Erläuterung
SPPZ REGULAR	// Bei einem positiven Ergebniswert oder Null im Akkumulator 1 zur Sprungmarke "REGULAR" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem negativen Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Bei VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
REGULAR: UN "MyTag_2"	// Sprungmarke "REGULAR"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE = "1" den Operanden "Tag_Output_2" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPMZ: Springen bei Ergebnis kleiner oder gleich Null

Beschreibung

Mit der Anweisung "Springen bei Ergebnis kleiner oder gleich Null" unterbrechen Sie, abhängig vom Statusbit A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand des Statusbits A1 "0" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Nach der Ausführung einer arithmetischen Anweisung ohne Überlauf ist der Wert des Akkumulators 1 kleiner oder gleich Null.
- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, *I, +D, *D) tritt ein Überlauf im negativen Bereich auf.

- Bei der Ausführung einer arithmetischen Anweisung mit Ganzzahlen (+I, -I, +D, -D, NEGI, NEGD) tritt ein Überlauf im positiven Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf im negativen Bereich auf.
- Bei der Ausführung einer arithmetischen Anweisung (Gleitpunktarithmetik) mit Überlauf tritt eine stufenweise Unterschreitung des zulässigen Bereichs auf.
- Nach der Ausführung einer Vergleichsanweisung ist der Inhalt des Akkumulators 2 kleiner oder gleich dem Inhalt des Akkumulators 1.
- Nach der Ausführung einer Wortverknüpfung ist der Wert des Akkumulators 1 gleich Null.
- Nach der Ausführung einer Schiebeanweisung ist der Wert des zuletzt hinausgeschobenen Bits gleich Null.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis kleiner oder gleich Null":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
-I	// Wert des Operanden "Tag_Value_2" vom Wert des Operanden "Tag_Value_1" subtrahieren.
SPMZ MyLABEL	// Bei einem negativen Ergebniswert oder Null im Akkumulator 1 zur Sprungmarke "MyLABEL" springen und die Programmbearbeitung dort fortsetzen.
	// Bei einem positiven Ergebniswert im Akkumulator 1 nächstfolgende Anweisung bearbeiten.
UN "MyTag_1"	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_1"	// Be VKE = "1" den Operanden "Tag_Output_1" auf "1" setzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL: UN "MyTag_2"	// Sprungmarke "MyLABEL"
	// Operanden "MyTag_2" auf "0" abfragen und nach UND verknüpfen.
S "Tag_Output_2"	// Bei VKE="1" den Operanden "Tag_Output_2" auf "1" setzen

AWL	Erläuterung
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPU: Springen bei Ergebnis ungültig

Beschreibung

Mit der Anweisung "Springen bei Ergebnis ungültig" unterbrechen Sie, abhängig von den Statusbits A0 und A1, die lineare Programmbearbeitung und setzen sie an der Stelle fort, die durch die angegebene Sprungmarke gekennzeichnet ist.

Der Sprung zur angegebenen Sprungmarke wird nur ausgeführt, wenn der Signalzustand der Statusbits A0 und A1 "1" ist. Dies tritt auf, wenn eine der folgenden Bedingungen zutrifft:

- Bei einer mathematischen Anweisung (/I, /D, MOD) wird es durch Null dividiert.
- Bei der Ausführung einer arithmetischen Anweisung mit Gleitpunktzahlen tritt ein Überlauf auf und der Ergebniswert ist eine ungültige Gleitpunktzahl.
- Bei der Ausführung einer Vergleichsanweisung mit Gleitpunktzahlen wurde eine ungültige Gleitpunktzahl verwendet oder ist als Ergebniswert entstanden.

In allen anderen Fällen wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der nächstfolgenden Anweisung fortgesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Springen bei Ergebnis ungültig":

Parameter	Beschreibung
<Sprungmarke>	Symbolischer Name des Sprungziels

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" laden.
L "Tag_Value_2"	// Wert des Operanden "Tag_Value_2" laden.
/I	// Wert des Operanden "Tag_Value_1" durch den Wert des Operanden "Tag_Value_2" dividieren.
SPU ERROR	// Bei einer Division durch Null zur Sprungmarke "ERROR" springen und die Programmbearbeitung dort fortsetzen.
	// In anderem Fall die nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.
U "MyTag_1"	// Operanden "MyTag_1" auf "1" abfragen und nach UND verknüpfen.
R "MyTag_1"	// Bei VKE = "1" den Operanden "MyTag_1" auf "0" zurücksetzen.
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
ERROR: UN "MyTag_1"	// Sprungmarke "ERROR"
	// Operanden "MyTag_1" auf "0" abfragen und nach UND verknüpfen.
S "MyTag_1"	// Bei VKE = "1" den Operanden "MyTag_1" auf "1" setzen.
NEXT: U "MyTag_3"	// Sprungmarke "NEXT"
	// Operanden "MyTag_3" auf "1" abfragen und nach UND verknüpfen.
U "MyTag_4"	// Operanden "MyTag_4" auf "1" abfragen und nach UND verknüpfen.

Siehe auch

Grundlagen zum Statuswort (Seite 183)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Setzen der Statusbits (Seite 185)

SPL: Sprungliste definieren

Beschreibung

Mit der Anweisung "Sprungliste definieren" programmieren Sie eine Liste, die aus mehreren Einträgen der Anweisung "Absolut springen" (SPA) besteht. Die Liste beginnt unmittelbar nach der Anweisung "Sprungliste definieren" und darf maximal 255 Einträge enthalten. Die Nummerierung der Sprungfunktionen in der Liste beginnt bei Null. Die Sprungliste muss lückenlos programmiert werden. Das Ende der Sprungliste wird durch eine Sprungmarke gekennzeichnet, die bei der Anweisung "Sprungliste definieren" angegeben wird.

Welche Sprungfunktion der Liste ausgeführt wird, hängt vom Wert im rechten Byte des Akkumulators 1 ab. Wenn z. B. im Akkumulator 1 der Wert "0" steht, wird die erste Sprungfunktion ausgeführt. Bei einem Wert "1" im Akkumulator 1, wird die zweite Sprungfunktion ausgeführt. Wenn der Wert im Akkumulator "1" größer als die Anzahl der Listeneinträge ist, verweist die Anweisung "Sprungliste definieren" zum Ende der Liste.

Die Anweisung "Sprungverteiler'" wird unabhängig von Bedingungen ausgeführt und beeinflusst die Statusbits nicht.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Sprungnummer in Akkumulator 1 laden.
SPL END	// Anfang der Sprungliste
SPA MyLABEL_1	// Bei einem Wert "0" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_1" springen.
SPA MyLABEL_2	// Bei einem Wert "1" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_2" springen.
SPA MyLABEL_3	// Bei einem Wert "2" im Akkumulator 1 ausführen und zur Sprungmarke "MyLABEL_3" springen.
END: L "MyTag_1"	// Ende der Sprungliste
	// Inhalt des Operanden "MyTag_1" in Akkumulator 1 laden.
L "MyTag_2"	// Inhalt des Operanden "MyTag_2" in Akkumulator 1 laden.
+I	// Werte addieren
T "Tag_Output_2"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output_2" laden.
MyLABEL_1: ...	// Sprungmarke "MyLABEL_1"
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL_2: ...	// Sprungmarke "MyLABEL_2"
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
MyLABEL_3: ...	// Sprungmarke "MyLABEL_3"

AWL	Erläuterung
....	// Beliebiges Programm
SPA NEXT	// Zur Sprungmarke "NEXT" springen und die Programmbearbeitung dort fortsetzen.
NEXT: ...	// Sprungmarke "NEXT"

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

LOOP: Schleifensprung

Beschreibung

Mit der Anweisung "Schleifensprung" programmieren Sie Programmschleifen.

Die Anweisung interpretiert das rechte Wort des Akkumulators 1 als eine vorzeichenlose 16-Bit Ganzzahl im Bereich von 0 bis 65535. In der Ausführung dekrementiert die Anweisung zuerst den Inhalt des Akkumulators 1 um eins. Wenn der Wert im Akkumulator 1 nach dem Dekrementieren nicht Null ist, wird der Sprung zur angegebenen Sprungmarke ausgeführt. Wenn der Wert Null ist, wird der Sprung nicht ausgeführt und die Programmbearbeitung mit der unmittelbar darauffolgenden Anweisung fortgesetzt. Der Wert im Akkumulator 1 gibt somit die Anzahl der zu durchlaufenden Programmschleifen vor. Diese Anzahl müssen Sie in einem Schleifenzähler speichern.

Der Operand <Anzahl> enthält die Anzahl der Schleifendurchläufe. Der Operand <Schleifenzähler> enthält noch auszuführende Schleifendurchläufe. Am Ende der Programmschleife wird der Inhalt des Schleifenzählers in den Akkumulator 1 geladen und von der Anweisung um eins dekrementiert. Der Sprung zur Sprungmarke am Anfang der Programmschleife wird ausgeführt, wenn der Wert im Akkumulator nach dem Dekrementieren ungleich Null ist.

Die Anweisung "Schleifensprung" beeinflusst keine Statusbits.

Hinweis

Wenn der Schleifenzähler nicht vorbelegt ist oder mit dem Wert "0" vorbelegt ist, wird der Schleifenzähler immer weiter ins Negative dekrementiert, wodurch eine endlose Schleife entsteht, die die CPU in den Betriebszustand STOP versetzen kann.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Anzahl der auszuführenden Schleifendurchläufe in das rechte Wort des Akkumulators 1 laden.
START: T "Tag_Counter"	// Anfang der Programmschleife
	// Inhalt des Akkumulators 1 in den Schleifenzähler transferieren.
L "MyTag_1"	// Wert des Operanden "MyTag_1" laden.
L "MyTag_2"	// Wert des Operanden "MyTag_2" laden.
*D	// Werte multiplizieren
T "MyTag_1"	// Ergebnis der Multiplikation in den Operanden "MyTag_1" transferieren.
L "Tag_Counter"	// Inhalt des Schleifenzählers in den Akkumulator 1 laden.
LOOP START	// Wert im Akkumulator 1 um eins dekrementieren.
	// Bei einem Wert ungleich Null zum Anfang der Programmschleife springen.
	// Bei einem Wert Null die nächstfolgende Anweisung bearbeiten.
L "MyTag_2"	// Wert des Operanden "MyTag_2" laden.
L 100	// Den Wert 100 laden.
>I	// Vergleichen, ob der Wert des Operanden "MyTag_2" größer als 200 ist.
= "MyTag_3"	// Vergleichergebnis in den Operanden "MyTag_3" schreiben.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Abfragen der Statusbits mit Sprunganweisungen (Seite 195)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Datenbausteine

AUF: Globalen Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Globalen Datenbaustein öffnen" schlagen Sie einen globalen Datenbaustein (DB) auf. Die Nummer des Datenbausteins wird in das DB-Register übertragen. Die darauf folgenden DB-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Die Anweisung "Globalen Datenbaustein öffnen" wird unabhängig von Bedingungen ausgeführt und beeinflusst weder das Verknüpfungsergebnis noch die Akkumulatorinhalte.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Globalen Datenbaustein öffnen":

Operand	Deklaration	Datenbausteintyp	Beschreibung
<Datenbaustein>	-	DB	Datenbaustein, der geöffnet wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`AUF "GlobalDataBlock"`	`// Globalen Datenbaustein öffnen.`
`L %DBW0`	`// Datenwort DBW0 des geöffneten globalen Datenbausteins in Akkumulator 1 laden.`
`T "MyTag"`	`// Inhalt des Akkumulators 1 in den Operanden "MyTag_1" transferieren.`

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

AUFDI: Instanz-Datenbaustein öffnen

Beschreibung

Mit der Anweisung "Instanz-Datenbaustein öffnen" schlagen Sie einen Instanz-Datenbaustein (DI) auf. Die Nummer des Datenbausteins wird in das DI-Register übertragen. Die darauf folgenden DI-Befehle greifen in Abhängigkeit der Registerinhalte auf die entsprechenden Bausteine zu.

Die Anweisung "Instanz-Datenbaustein öffnen" wird unabhängig von Bedingungen ausgeführt und beeinflusst weder das Verknüpfungsergebnis noch die Akkumulatorinhalte.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Instanz-Datenbaustein öffnen":

Operand	Deklaration	Datenbausteintyp	Beschreibung
<Datenbaustein>	-	DI	Datenbaustein, der geöffnet wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`AUFDI "InstanceDataBlock"`	`// Instanz-Datenbaustein öffnen`
`L %DIW0`	`// Datenwort DIW0 des geöffneten Instanz-Datenbausteins in Akkumulator 1 laden.`
`T "MyTag"`	`// Inhalt des Akkumulators 1 in den Operanden "MyTag" transferieren.`

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

TDB: Datenbausteinregister tauschen

Beschreibung

Mit der Anweisung "Datenbausteinregister tauschen" tauschen Sie die Inhalte der Datenbausteinregister. Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst die Statusbits nicht.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUF "GlobalDataBlock"	// Globalen Datenbaustein öffnen
AUFDI "InstanceDataBlock"	// Instanz-Datenbaustein öffnen
TDB	// Datenbausteinregister tauschen
	// Das DB-Register verweist auf "InstanceDataBlock" und das DI-Register verweist auf "GlobalDataBlock".
L %DIW0	// DW0 aus "GlobalDataBlock" in Akkumulator 1 laden.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L DBLG: Länge eines Global-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Länge eines Global-Datenbausteins in Akkumulator 1 laden" laden Sie die Länge eines über das Datenbausteinregister geöffneten Global-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Global-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen. Die Länge des Datenbausteins ist gleichbedeutend mit der Anzahl der Datenbytes.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUF "GlobalDataBlock"	// Globalen Datenbaustein öffnen
L DBLG	// Länge des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
<D	// Vergleichen, ob die Länge des Datenbausteins kleiner als der Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis in den Operanden "MyTag_2" schreiben.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L DBNO: Nummer eines Global-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Nummer eines Global-Datenbausteins in Akkumulator 1 laden" laden Sie die Nummer eines über das Datenbausteinregister geöffneten Global-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Global-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`AUF "GlobalDataBlock"`	`// Globalen Datenbaustein öffnen`
`L DBNO`	`// Nummer des geöffneten Datenbausteins in Akkumulator 1 laden.`
`L "MyTag_1"`	`// Vergleichswert laden`
`==I`	`// Vergleichen, ob die Nummer des Datenbausteins gleich dem Wert des Operanden "MyTag_1" ist.`
`= "MyTag_2"`	`// Abfrageergebnis in den Operanden "MyTag_2" schreiben.`

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L DILG: Länge eines Instanz-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Länge eines Instanz-Datenbausteins in Akkumulator 1 laden" laden Sie die Länge eines über das Datenbausteinregister geöffneten Instanz-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Instanz-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen. Die Länge des Datenbausteins ist gleichbedeutend mit der Anzahl der Datenbytes.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUFDI "InstanceDataBlock"	// Instanz-Datenbaustein öffnen
L DILG	// Länge des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden
<I	// Vergleichen, ob die Länge des Datenbausteins kleiner als der Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis in den Operanden "MyTag_2" schreiben.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

L DINO: Nummer eines Instanz-Datenbausteins in Akkumulator 1 laden

Beschreibung

Mit der Anweisung "Nummer eines Instanz-Datenbausteins in Akkumulator 1 laden" laden Sie die Nummer eines über das Datenbausteinregister geöffneten Instanz-Datenbausteins in Akkumulator 1. Der vorherige Inhalt des Akkumulators 1 wird dabei in Akkumulator 2 verschoben.

Wenn vor der Ausführung der Anweisung kein Instanz-Datenbaustein über das Datenbausteinregister geöffnet worden ist, wird in Akkumulator 1 der Wert "0" geladen.

Die Statusbits werden durch die Anweisung nicht beeinflusst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
AUFDI "InstanceDataBlock"	// Instanz-Datenbaustein öffnen
L DINO	// Nummer des geöffneten Datenbausteins in Akkumulator 1 laden.
L "MyTag_1"	// Vergleichswert laden

AWL	Erläuterung
==I	// Vergleichen, ob die Nummer des Datenbausteins gleich dem Wert des Operanden "MyTag_1" ist.
= "MyTag_2"	// Abfrageergebnis dem Operanden "MyTag_2" zuweisen.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Codebausteine

BE: Bausteinende

Beschreibung

Mit der Anweisung "Bausteinende" beenden Sie die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird. Die Programmbearbeitung wird dann mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Beim Beenden des Bausteins wird das aktuelle Verknüpfungsergebnis in den aufrufenden Baustein übertragen.

Die Anweisung "Bausteinende" wird unabhängig von Bedingungen ausgeführt. Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
SIN	// Sinuswert bilden
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt im Akkumulator 1 speichern.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.
BE	// Baustein beenden

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

BEB: Bedingtes Bausteinende

Beschreibung

Mit der Anweisung "Bedingtes Bausteinende" beenden Sie, abhängig vom Verknüpfungsergebnis (VKE), die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird.

Wenn das VKE "1" ist, wird die Anweisung ausgeführt. Der aktuell bearbeitete Baustein wird beendet und die Programmbearbeitung im aufrufenden Baustein fortgesetzt. Die Programmbearbeitung wird mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Beim Beenden des Bausteins wird das aktuelle Verknüpfungsergebnis in den aufrufenden Baustein übertragen. Die MCR-Abhängigkeit stellt sich im aufrufenden Baustein so ein, wie sie vor dem Bausteinaufruf war. Die Datenbausteine, die zum Zeitpunkt des Bausteinaufrufs geöffnet waren, werden erneut geöffnet.

Wenn das VKE bei der Bearbeitung der Anweisung "0" ist, wird die Anweisung nicht ausgeführt. Die CPU setzt in diesem Fall das VKE auf "1" und bearbeitet die nächstfolgende Anweisung.

Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U "Tag_Input"	// Operanden "Tag_Input" auf "1" abfragen und mit dem VKE nach UND verknüpfen.
BEB	// Bei erfüllter Bedingung (VKE = "1") Baustein beenden.
	// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

BEA: Absolutes Bausteinende

Beschreibung

Mit der Anweisung "Absolutes Bausteinende" beenden Sie die Bearbeitung des aktuell bearbeiteten Bausteins und wechseln zur Programmstelle, an der der Baustein aufgerufen wird. Die Programmbearbeitung wird dann mit der Anweisung fortgesetzt, die direkt nach dem Bausteinaufruf steht.

Beim Beenden des Bausteins wird das aktuelle Verknüpfungsergebnis in den aufrufenden Baustein übertragen. Die MCR-Abhängigkeit stellt sich im aufrufenden Baustein so ein, wie sie vor dem Bausteinaufruf war. Die Datenbausteine, die zum Zeitpunkt des Bausteinaufrufs geöffnet waren, werden erneut geöffnet.

Die Anweisung "Absolutes Bausteinende" wird unabhängig von Bedingungen ausgeführt. Im Gegensatz zur Anweisung "Bausteinende" können Sie die Anweisung "Absolutes Bausteinende" mehrfach innerhalb eines Bausteins programmieren. Wenn die Bearbeitung der Anweisung durch eine Sprunganweisung übersprungen wird, wird der aktuelle Programmablauf nicht beendet, sondern am Sprungziel innerhalb des Bausteins fortgesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
U "Tag_Input"	// Operanden "Tag_Input" auf "1" abfragen und mit dem VKE nach UND verknüpfen.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1") Programmbearbeitung bei der Sprungmarke "NEXT" fortsetzen.
	// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.
BEA	// Baustein beenden
NEXT: T "Tag_Result"	// Sprungmarke "NEXT"
	// Inhalt des Akkumulators 1 in den Operanden "Tag_Result" übertragen.

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

CALL: Baustein aufrufen

Beschreibung

Mit der Anweisung "Baustein aufrufen" rufen Sie die folgenden Bausteinarten im Programm auf:

- Funktionen
- Funktionsbausteine

Die Anweisung "Baustein aufrufen" wird unabhängig von Bedingungen ausgeführt. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt.

Einen aufgerufenen Baustein können Sie mit Daten versorgen. Die Datenübergabe erfolgt über die Bausteinparameter. Die Parameter des aufgerufenen Bausteins werden nach der Aufrufanweisung im aufrufenden Baustein aufgelistet. Diesen können Sie die benötigten

Aktualparameter zuordnen. Bei der Ausführung der Anweisung "Baustein aufrufen" werden die Daten an den aufgerufenen Baustein übergeben. Durch die Datenübergabe werden die Inhalte des Statusworts sowie der Adress- und Datenbaustein-Register verändert.

Die nicht versorgten Parameter eines Funktionsbausteins behalten ihren aktuellen Wert bei. Bei Aufruf von Funktionen müssen alle Parameter versorgt werden. Falls ein aufgerufener Baustein keine Parameter hat, wird keine Parameterliste angezeigt.

Wenn der aufgerufene Baustein einen Instanz-Datenbaustein benötigt, geben Sie ihn durch ein Komma getrennt beim Aufruf an. Der angegebene Datenbaustein muss vor dem Aufruf angelegt sein.

Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Die MCR-Abhängigkeit stellt sich im aufrufenden Baustein so ein, wie sie vor dem Bausteinaufruf war. Die Datenbausteine, die zum Zeitpunkt des Bausteinaufrufs geöffnet waren, werden erneut geöffnet.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`CALL "MyFunction"`	`// Funktion "MyFunction" aufrufen`
`  Input_1 := "Tag_Input_1",`	`// Aktualparameter zuordnen`
`  Input_2 := "Tag_Input_2"`	
`  Output_1 := "Tag_Output_1"`	
`  Output_2 := "Tag_Output_1"`	
`CALL "MyFunctionBlock", "MyFB_DB"`	`// Funktionsbaustein "MyFunctionBlock" aufrufen`
`  Value_1 := "Tag_Value_1"`	`// Aktualparameter zuordnen`
`  Value_2 := "Tag_Value_2"`	
`  Output := "Tag_Output"`	
`CALL "LIMIT"`	`// Anweisung "Limitieren" aufrufen`
`  MN := "Tag_LowLimit"`	`// Aktualparameter zuordnen`
`  IN := "Tag_InputValue"`	
`  MX := "Tag_HighLimit"`	
`  OUT := "Tag_Output"`	
`CALL "CTU", "CTU_DB"`	`// Zähler "Vorwärts zählen" aufrufen`
`  CU := "Tag_StartCTU"`	`// Aktualparameter zuordnen`
`  R := "Tag_ResetCounter"`	
`  PV := "Tag_PresetValue"`	
`  Q := "Tag_CounterStatus"`	
`  CV := "Tag_CounterValue"`	

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

CC: Baustein bedingt aufrufen

Beschreibung

Mit der Anweisung "Baustein bedingt aufrufen" rufen Sie abhängig vom Verknüpfungsergebnis (VKE) Funktionen (FC) und Funktionsbausteine (FB) auf, die keine Parameter und keine Instanz-Datenbausteine haben.

Die Anweisung wird nur ausgeführt, wenn das aktuelle Verknüpfungsergebnis (VKE) vor der Bearbeitung der Anweisung "1" ist. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt. Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Beim Wechseln des Bausteins werden die Statusbits OS und /ER auf "0" zurückgesetzt. Die Statusbits A0, A1 und OV werden nicht beeinflusst.

Die MCR-Abhängigkeit stellt sich im aufrufenden Baustein so ein, wie sie vor dem Bausteinaufruf war. Die Datenbausteine, die zum Zeitpunkt des Bausteinaufrufs geöffnet waren, werden erneut geöffnet.

Der Inhalt der Akkumulatoren und der Adressregister wird durch die Anweisung "Baustein bedingt aufrufen" nicht verändert.

Wenn das aktuelle VKE "0" ist, wird die Anweisung und damit auch der Bausteinaufruf nicht ausgeführt und das VKE auf "1" gesetzt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input_1"`	`// Operanden "Tag_Input_1" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`CC "MyFunction"`	`// Bei erfüllter Bedingung (VKE = "1") die Funktion "MyFunction" aufrufen.` `// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.`
`U "Tag_Input_2"`	`// Operanden "Tag_Input_2" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`CC "MyFunctionBlock"`	`// Bei erfüllter Bedingung (VKE = "1") den Funktionsbaustein "MyFunctionBlock" aufrufen.` `// Bei nicht erfüllter Bedingung (VKE = "0") die nächstfolgende Anweisung bearbeiten.`

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

UC: Baustein unbedingt aufrufen

Beschreibung

Mit der Anweisung "Baustein unbedingt aufrufen" rufen Sie Funktionen (FC) und Funktionsbausteine (FB) auf, die keine Parameter und keine Instanz-Datenbausteine haben.

Die Anweisung wird unabhängig von Bedingungen ausgeführt. Nach der Ausführung der Anweisung wird die Programmbearbeitung im aufgerufenen Baustein fortgesetzt. Wenn der aufgerufene Baustein bearbeitet ist, wechselt die CPU zurück zum aufrufenden Baustein und fährt mit der Bearbeitung dieses Bausteins nach der Aufrufanweisung fort.

Beim Wechseln des Bausteins werden die Statusbits OS und /ER auf "0" zurückgesetzt. Die Statusbits A0, A1 und OV werden nicht beeinflusst.

Die MCR-Abhängigkeit stellt sich im aufrufenden Baustein so ein, wie sie vor dem Bausteinaufruf war. Die Datenbausteine, die zum Zeitpunkt des Bausteinaufrufs geöffnet waren, werden erneut geöffnet.

Der Inhalt der Akkumulatoren und der Adressregister wird durch die Anweisung "Baustein unbedingt aufrufen" nicht verändert.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`U "Tag_Input_1"`	`// Operanden "Tag_Input_1" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`UC "MyFunction"`	`// Funktion "MyFunction" aufrufen.`
`U "Tag_Input_2"`	`// Operanden "Tag_Input_2" auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.`
`UC "MyFunctionBlock"`	`// Funktionsbaustein "MyFunctionBlock" aufrufen.`

Siehe auch

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Wortverknüpfungen

UW: Wortweise nach UND verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach UND verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach UND. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn beide zu verknüpfenden Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach UND gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	0	0	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach UND verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach UND verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.`

AWL	Erläuterung
UW W#16#F6B5	// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach UND verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
UW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach UND verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	0101	0000	0011	0001
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	0100	1000	0000	0010

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

OW: Wortweise nach ODER verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach ODER verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach ODER. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.`
`OW W#16#F6B5`	`// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach ODER verknüpfen.` `// Ergebnis im rechten Wort des Akkumulators 1 speichern.`

AWL	Erläuterung
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
OW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1111	1111	1011	1111
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1111	1110	1011	1011

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Programmierbeispiele AWL (Seite 7595)

XOW: Wortweise nach EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "Wortweise nach EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert des rechten Worts im Akkumulator 1 mit dem Wert des rechten Worts im Akkumulator 2 oder einer angegebenen Konstante bitweise nach EXKLUSIV ODER. Das Ergebnis wird im rechten Wort des Akkumulators 1 abgelegt. Der Inhalt des linken Worts im Akkumulator 1 bleibt unverändert.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 15 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn der Signalzustand der beiden zu verknüpfenden Bits unterschiedlich ist. Wenn beide zu verknüpfende Bits den gleichen Signalzustand führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach EXKLUSIV ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	0

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach EXKLUSIV ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	WORD	Wert, der mit dem im rechten Wort des Akkumulators 1 stehenden Wert nach EXKLUSIV ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.

AWL	Erläuterung
XOW W#16#F6B5	// Wert des rechten Worts im Akkumulator 1 mit der Konstante (W#16#F6B5) nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in das rechte Wort des Akkumulators 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in das rechte Wort des Akkumulators 1 laden.
XOW	// Wert des rechten Worts im Akkumulator 2 ("Tag_Value_2") mit dem Wert des rechten Worts im Akkumulator 1 ("Tag_Value_3")nach EXKLUSIV ODER verknüpfen.
	// Ergebnis im rechten Wort des Akkumulators 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_Value_1	0101	1001	0011	1011
Konstante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1010	1111	1000	1110
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1011	0110	1011	1001

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

UD: Doppelwortweise nach UND verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach UND verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach UND. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn beide zu verknüpfenden Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach UND gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	0	0	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach UND verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach UND verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in den Akkumulator 1 laden.
UD DW#16#39C657AC            // Wert des Akkumulators 1 mit der Konstante
                             (DW#16#39C657AC) nach UND verknüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"             // Ergebnis in den Operanden "Tag_Result_1" übertra-
                             gen.
L "Tag_Value_2"              // Wert des Operanden in den Akkumulator 1 laden.
```

AWL	Erläuterung
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in den Akkumulator 1 laden.
UD	// Wert des Akkumulators 2 ("Tag_Value_2") mit dem Wert des Akkumulators 1 ("Tag_Value_3")nach UND verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0001	1001	0100	0100	0001	0101	0010	1000
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0010	0001	0100	0010	0101	0101	0010	1010

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

OD: Doppelwortweise nach ODER verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach ODER verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach ODER. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	1

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in den Akkumulator 1 laden.
OD DW#16#39C657AC	// Wert des Akkumulators 1 mit der Konstante (DW#16#39C657AC) nach ODER verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"	// Ergebnis in den Operanden "Tag_Result_1" übertragen.
L "Tag_Value_2"	// Wert des Operanden in den Akkumulator 1 laden.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_3" in den Akkumulator 1 laden.
OD	// Wert des Akkumulators 2 ("Tag_Value_2") mit dem Wert des Akkumulators 1 ("Tag_Value_3")nach ODER verknüpfen.
	// Ergebnis im Akkumulator 1 speichern
T "Tag_Result_2"	// Ergebnis in den Operanden "Tag_Result_2" übertragen.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39 C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_1	0111	1111	1110	0110	1101	1111	1011	1111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Re-sult_2	0111	1101	1100	0111	0101	1111	1110	1111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

XOD: Doppelwortweise nach EXKLUSIV ODER verknüpfen

Beschreibung

Mit der Anweisung "Doppelwortweise nach EXKLUSIV ODER verknüpfen" verknüpfen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2 oder mit dem Wert einer angegebenen Konstante bitweise nach EXKLUSIV ODER. Das Ergebnis wird im Akkumulator 1 abgelegt.

Die Anweisung verknüpft das Bit 0 des Akkumulators 1 mit dem Bit 0 des Akkumulators 2 oder einer Konstante und legt das Ergebnis im Bit 0 des Akkumulators 1 ab. Auf die gleiche Weise werden auch die Bits 1 bis 31 verknüpft.

Ein Ergebnisbit hat den Signalzustand "1", wenn der Signalzustand der beiden zu verknüpfenden Bits unterschiedlich ist. Wenn beide zu verknüpfende Bits den gleichen Signalzustand führen, wird das entsprechende Ergebnisbit auf "0" zurückgesetzt.

Die folgende Tabelle zeigt, wie das Ergebnis bei einer Wortverknüpfung nach EXKLUSIV ODER gebildet wird:

Akkumulator 2 / Konstante	0	0	1	1
Akkumulator 1	0	1	0	1
Ergebnis	0	1	1	0

Die Anweisung wird unabhängig von Bedingungen ausgeführt und beeinflusst das Verknüpfungsergebnis nicht.

Die Anweisung beeinflusst die Statusbits A0, A1 und OV.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach EXKLUSIV ODER verknüpfen":

Parameter	Datentyp	Beschreibung
<Konstante>	DWORD	Wert, der mit dem im Akkumulator 1 stehenden Wert nach EXKLUSIV ODER verknüpft wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in den Akkumulator 1 laden.
XOD DW#16#39C657AC           // Wert des Akkumulators 1 mit der Konstante
                             (DW#16#39C657AC) nach EXKLUSIV ODER verknüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_1"             // Ergebnis in den Operanden "Tag_Result_1" übertra-
                             gen.
L "Tag_Value_2"              // Wert des Operanden in den Akkumulator 1 laden.
L "Tag_Value_3"              // Inhalt des Akkumulators 1 in den Akkumulator 2 ver-
                             schieben.
                             // Wert des Operanden "Tag_Value_3" in den Akkumulator
                             1 laden.
XOD                          // Wert des Akkumulators 2 ("Tag_Value_2") mit dem
                             Wert des Akkumulators 1 ("Tag_Value_3")nach EXKLUSIV
                             ODER verknüpfen.
                             // Ergebnis im Akkumulator 1 speichern.
T "Tag_Result_2"             // Ergebnis in den Operanden "Tag_Result_2" übertra-
                             gen.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Konstante (DW#16#39 C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0110	0110	1010	0010	1100	1010	1001	0111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011

Operand	Wert							
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0101	1100	1000	0101	0000	1010	1000	0111

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Setzen der Statusbits bei Wortverknüpfungen (Seite 192)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Schieben und Rotieren

Schieben

SSI: Mit Vorzeichen wortweise schieben

Beschreibung

Mit der Anweisung "Mit Vorzeichen wortweise schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit dem Signalzustand des Bits 15 (Vorzeichenbit bei INT-Zahlen) aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SSI <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SSI: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit dem Signalzustand des Bits 15 aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden in Akkumulator 1 laden.
SSI 6                       // Bits 0 bis 15 des Akkumulators 1 um 6 Stellen nach
                            rechts schieben.
                            // Freigewordene Bitstellen mit dem Zustand des Bits
                            15 auffüllen.
T "Tag_Result_1"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
L 3                         // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"             // Schiebezahl in das rechte Byte des Akkumulators 2
                            verschieben.
                            // Wert des Operanden in Akkumulator 1 laden.
SSI                         // Bits 0 bis 15 des Akkumulators 1 um 3 Stellen nach
                            rechts schieben.
                            // Freigewordene Bitstellen mit dem Zustand des Bits
                            15 auffüllen.
T "Tag_Result_2"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1111	1110	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	1011	1010	0101

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SSD: Mit Vorzeichen doppelwortweise schieben

Beschreibung

Mit der Anweisung "Mit Vorzeichen doppelwortweise schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit dem Signalzustand des Bits 31 (Vorzeichenbit bei DINT-Zahlen) aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SSD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SSD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit dem Signalzustand des Bits 31 aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in Akkumulator 1 laden.
SSD 7	// Bits 0 bis 31 des Akkumulators 1 um 7 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit dem Zustand des Bits 31 auffüllen.
T "Tag_Result_1"	// Inhalt des Akkumulators 1 in den Operanden transferieren.
L 4	// Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SSD	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit dem Zustand des Bits 31 auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	1000	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	1111	0001	1110	1100	1000	1011	1010
Tag_Value_2	0010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	0010	1000	1010	0010	1001	1011	1100

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SLW: Wortweise nach links schieben

Beschreibung

Mit der Anweisung "Wortweise nach links schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach links. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SLW <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SLW: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.

Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach links schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SLW 5                        // Bits 0 bis 15 des Akkumulators 1 um 5 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

AWL	Erläuterung
L 4	// Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SLW	// Bits 0 bis 15 des Akkumulators 1 um 4 Stellen nach links schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1010	0111	0110	0000
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	1101	0010	1011	0000

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SRW: Wortweise nach rechts schieben

Beschreibung

Mit der Anweisung "Wortweise nach rechts schieben" schieben Sie das rechte Wort (Bits 0 bis 15) des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Bits 16 bis 31 des Akkumulators 1 bleiben unverändert. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung. (<Anzahl>)
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SRW <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 15. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SRW: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Wenn die angegebene Schiebezahl größer 15 ist, werden alle Bits im rechten Wort des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wortweise nach rechts schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SRW 6                        // Bits 0 bis 15 des Akkumulators 1 um 6 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 4                          // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Schiebezahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
SRW                          // Bits 0 bis 15 des Akkumulators 1 um 4 Stellen nach
                             rechts schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	0000	0001	0111	0100

Operand	Wert							
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Re-sult_2	0101	1111	0110	0100	0000	0101	1101	0010

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SLD: Doppelwortweise nach links schieben

Beschreibung

Mit der Anweisung "Doppelwortweise nach links schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach links. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SLD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SLD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach links schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                          Erläuterung
L "Tag_Value_1"              // Wert des Operanden in Akkumulator 1 laden.
SLD 5                        // Bits 0 bis 31 des Akkumulators 1 um 5 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
L 4                          // Schiebezahl in Akkumulator 1 laden.
L "Tag_Value_2"              // Schiebezahl in das rechte Byte des Akkumulators 2
                             verschieben.
                             // Wert des Operanden in Akkumulator 1 laden.
SLD                          // Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach
                             links schieben.
                             // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"             // Inhalt des Akkumulators 1 in den Operanden transfe-
                             rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1110	1100	1000	1011	1010	0111	0110	0000
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	1000	1010	0010	1001	1011	1100	1101	0000

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

SRD: Doppelwortweise nach rechts schieben

Beschreibung

Mit der Anweisung "Doppelwortweise nach rechts schieben" schieben Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Die beim Schieben frei werdenden Stellen werden mit Nullen aufgefüllt. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die geschoben wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- SRD <Anzahl>: Die Schiebezahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 31. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- SRD: Die Schiebezahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 31 werden alle Bits des Akkumulators 1 mit Nullen aufgefüllt.

Wenn die angegebene Schiebezahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Doppelwortweise nach rechts schieben":

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die geschoben wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden in Akkumulator 1 laden.
SRD 7                       // Bits 0 bis 31 des Akkumulators 1 um 7 Stellen nach
                            rechts schieben.
                            // Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_1"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
L 4                         // Schiebezahl in Akkumulator 1 laden.
```

AWL	Erläuterung
L "Tag_Value_2"	// Schiebezahl in das rechte Byte des Akkumulators 2 verschieben.
	// Wert des Operanden in Akkumulator 1 laden.
SRD	// Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach rechts schieben.
	// Freigewordene Bitstellen mit Nullen auffüllen.
T "Tag_Result_2"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0000	0000	1011	1110	1100	1000	1011	1010
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	1010	1000	1010	0010	1001	1011	1100

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Rotieren

RLD: Doppelwortweise nach links rotieren

Beschreibung

Mit der Anweisung "Doppelwortweise nach links rotieren" rotieren Sie den gesamten Inhalt des Akkumulators 1 bitweise nach links. Bei der Ausführung schiebt die Anweisung die Bits 0 bis 31 des Akkumulators 1 bitweise nach links und füllt die beim Schieben frei werdenden Stellen mit den hinausgeschobenen Bitstellen auf. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die rotiert wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- RLD <Anzahl>: Die Rotierzahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 32. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- RLD: Die Rotierzahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 32 wird die Rotierzahl mit einer Modulo-Division durch 32 berechnet. Wenn z. B. die Rotierzahl 34 ist, werden die Bits des Akkumulators 1 um 2 Bitstellen rotiert.

Wenn die angegebene Rotierzahl 32 ist, bleibt der Inhalt des Akkumulators 1 unverändert.

Wenn die angegebene Rotierzahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die rotiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden in Akkumulator 1 laden.
RLD 4                       // Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach
                            links rotieren.
T "Tag_Result_1"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
L 6                         // Rotierzahl in Akkumulator 1 laden.
L "Tag_Value_2"             // Rotierzahl in das rechte Byte des Akkumulators 2
                            verschieben.
                            // Wert des Operanden in Akkumulator 1 laden.
RLD                         // Bits 0 bis 31 des Akkumulators 1 um 6 Stellen nach
                            links rotieren.
T "Tag_Result_2"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	**0101**	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	0110	0100	0101	1101	0011	1011	**0101**

Operand	Wert							
Tag_Value_2	**1010**	**10**00	1010	0010	1001	1011	1100	1101
Tag_Result_2	0010	1000	1010	0110	1111	0011	01**10**	**1010**

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

RRD: Doppelwortweise nach rechts rotieren

Beschreibung

Mit der Anweisung "Doppelwortweise nach rechts rotieren" rotieren Sie den gesamten Inhalt des Akkumulators 1 bitweise nach rechts. Bei der Ausführung schiebt die Anweisung die Bits 0 bis 31 des Akkumulators 1 bitweise nach rechts und füllt die beim Schieben frei werdenden Stellen mit den hinausgeschobenen Bitstellen auf. Die Anweisung wird unabhängig vom VKE ausgeführt. Das Statusbit A1 wird auf den Signalzustand des zuletzt hinausgeschobenen Bits gesetzt.

Zur Angabe der Anzahl der Bitstellen, um die rotiert wird, gibt es die folgenden Möglichkeiten:

- Angabe einer positiven Ganzzahl als Parameter der Anweisung (<Anzahl>).
- Angabe über den Wert des rechten Bytes im Akkumulator 2. Das Byte wird als eine positive Ganzzahl interpretiert.

Die folgenden Formate können Sie verwenden:

- RRD <Anzahl>: Die Rotierzahl wird von dem Operanden <Anzahl> angegeben. Zulässig sind Werte zwischen 0 und 32. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn <Anzahl> größer Null ist.
- RRD: Die Rotierzahl wird von dem Wert in Akkumulator 2 angegeben. Zulässig sind Werte zwischen 0 und 255. Die Statusbits A0 und OV werden auf "0" zurückgesetzt, wenn der Inhalt von Akkumulator 2 größer Null ist.
  Bei Werten größer als 32 wird die Rotierzahl mit einer Modulo-Division durch 32 berechnet. Wenn z. B. die Rotierzahl 34 ist, werden die Bits des Akkumulators 1 um 2 Bitstellen rotiert.

Wenn die angegebene Rotierzahl 32 ist, bleibt der Inhalt des Akkumulators 1 unverändert.

Wenn die angegebene Rotierzahl gleich Null ist, wird die Anweisung trotzdem ausgeführt. Das Statusbit A1 wird auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Format	Beschreibung
<Anzahl>	Positive Ganzzahlen: SINT, INT, UINT, USINT	Anzahl der Bitstellen, um die rotiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
AWL                         Erläuterung
L "Tag_Value_1"             // Wert des Operanden in Akkumulator 1 laden.
RRD 4                       // Bits 0 bis 31 des Akkumulators 1 um 4 Stellen nach
                            rechts rotieren.
T "Tag_Result_1"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
L 6                         // Rotierzahl in Akkumulator 1 laden.
L "Tag_Value_2"             // Rotierzahl in das rechte Byte des Akkumulators 2
                            verschieben.
                            // Wert des Operanden in Akkumulator 1 laden.
RRD                         // Bits 0 bis 31 des Akkumulators 1 um 6 Stellen nach
                            rechts rotieren.
T "Tag_Result_2"            // Inhalt des Akkumulators 1 in den Operanden transfe-
                            rieren.
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	**1011**
Tag_Result_1	**1011**	0101	1111	0110	0100	0101	1101	0011
Tag_Value_2	1010	1000	1010	0010	1001	1011	11**00**	**1101**
Tag_Result_2	**0011**	**01**10	1010	0010	1000	1010	0110	1111

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

Übersicht über die gültigen Datentypen (Seite 243)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

RLDA: Durch Statusbit A1 nach links rotieren

Beschreibung

Mit der Anweisung "Durch Statusbit A1 nach links rotieren" schieben Sie den Inhalt des Akkumulators 1 um 1 Bit nach links. Die beim Schieben freiwerdende Bitstelle (0) wird mit dem Signalzustand des Statusbits A1 aufgefüllt. Das Statusbit A1 erhält den Signalzustand des hinausgeschobenen Bits (31).

Die Anweisung wird unabhängig vom VKE ausgeführt. Sie setzt das Statusbit A0 auf "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden in Akkumulator 1 laden.
RLDA	// Bits 0 bis 31 des Akkumulators 1 um 1 Stelle nach links rotieren.
	// Bit 1 mit dem Signalzustand des Statusbits A1 auffüllen.
	// Hinausgeschobenes Bit in das Statusbit A1 schreiben.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	A1	Wert							
Tag_Value	**1**	**0**101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	**0**	1011	1110	1100	1000	1011	1010	0111	011**1**

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

RRDA: Durch Statusbit A1 nach rechts rotieren

Beschreibung

Mit der Anweisung "Durch Statusbit A1 nach rechts rotieren" schieben Sie den Inhalt des Akkumulators 1 um 1 Bit nach rechts. Die beim Schieben freiwerdende Bitstelle (31) wird mit dem Signalzustand des Statusbits A1 aufgefüllt. Das Statusbit A1 erhält den Signalzustand des hinausgeschobenen Bits (0).

Die Anweisung wird unabhängig vom VKE ausgeführt. Sie setzt das Statusbit A0 auf "0" zurück.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value"	// Wert des Operanden in Akkumulator 1 laden.
RRDA	// Bits 0 bis 31 des Akkumulators 1 um 1 Stelle nach rechts rotieren.
	// Bit 31 mit dem Signalzustand des Statusbits A1 auffüllen.
	// Hinausgeschobenes Bit in das Statusbit A1 schreiben.
T "Tag_Result"	// Inhalt des Akkumulators 1 in den Operanden transferieren.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	A1	Wert							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1010	1111	1011	0010	0010	1110	1001	1101

Das Statusbit A1 hat den Wert "1", da das Bit 0 des Operanden "Tag_Value" auch den Wert "1" hat.

Siehe auch

Setzen der Statusbits bei Schiebe- und Rotieranweisungen (Seite 193)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Weitere Anweisungen

Akkumulator

TAK: Inhalte der Akkumulatoren 1 und 2 tauschen

Beschreibung

Mit der Anweisung "Inhalte der Akkumulatoren 1 und 2 tauschen" tauschen Sie den Inhalt des Akkumulators 1 mit dem Inhalt des Akkumulators 2. Bei Steuerungen mit vier Akkumulatoren werden die Inhalte der Akkumulatoren 3 und 4 durch die Anweisung nicht beeinflusst.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in den Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
>I	// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1")zur Sprungmarke "NEXT" springen.
	// Bei nicht erfüllter Bedingung (VKE = "0")die nächstfolgende Anweisung bearbeiten.
TAK	// Inhalte der Akkumulatoren 1 und 2 tauschen.
NEXT: -I	// Sprungmarke "NEXT"
	// Wert des Akkumulators 1 vom Wert des Akkumulators 2 subtrahieren.
T "Tag_Output"	// Inhalt des Akkumulators 1 in den Operanden "Tag_Output" transferieren.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

PUSH: Inhalt in den nächst höheren Akkumulator schieben

Beschreibung

Mit der Anweisung "Inhalt in den nächst höheren Akkumulator schieben" schieben Sie die Inhalte der Akkumulatoren 1 bis 3 jeweils in den nächst höheren Akkumulator.

Bei Steuerungen mit 2 Akkumulatoren wird nur der Inhalt des Akkumulators 1 in den Akkumulator 2 kopiert.

Bei Steuerungen mit 4 Akkumulatoren wird der Inhalt des Akkumulators 1 in den Akkumulator 2, der Inhalt des Akkumulators 2 in den Akkumulator 3 und der Inhalt des Akkumulators 3 in den Akkumulator 4 verschoben. Der Inhalt des Akkumulators 4 geht verloren.

Der Inhalt des Akkumulators 1 wird durch die Anweisung nicht beeinflusst und bleibt nach der Ausführung der Anweisung unverändert.

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 bis 4 vor und nach der Ausführung der Anweisung:

Status	Akkumulator			
	1	2	3	4
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert A	Wert B	Wert C

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden in Akkumulator 1 laden.
PUSH	// Inhalt der Akkumulatoren 1 bis 3 jeweils in den nächst höheren Akkumulator schieben.
L "Tag_Value_2"	// Wert des Operanden in Akkumulator 1 laden.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

POP: Inhalt in den nächst niedrigeren Akkumulator schieben

Beschreibung

Mit der Anweisung "Inhalt in den nächst niedrigeren Akkumulator schieben" schieben Sie die Inhalte der Akkumulatoren 4 bis 2 jeweils in den darunter liegenden Akkumulator.

Bei Steuerungen mit 2 Akkumulatoren wird der Inhalt des Akkumulators 2 in den Akkumulator 1 kopiert.

Bei Steuerungen mit 4 Akkumulatoren wird der Inhalt des Akkumulators 4 in den Akkumulator 3, der Inhalt des Akkumulators 3 in den Akkumulator 2 und der Inhalt des Akkumulators 2 in den Akkumulator 1 verschoben. Der Inhalt des Akkumulators 1 geht verloren.

Der Inhalt des Akkumulators 4 wird durch die Anweisung nicht beeinflusst und bleibt nach der Ausführung der Anweisung unverändert.

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 bis 4 vor und nach der Ausführung der Anweisung:

Status	Akkumulator			
	1	2	3	4
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert B	Wert C	Wert D	Wert D

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`T "Tag_Value_1"`	`// Inhalt des Akkumulators 1 in den Operanden "Tag_Value_1" übertragen.`
`POP`	`// Inhalt der Akkumulatoren 4 bis 2 jeweils in den darunter liegenden Akkumulator schieben.`
`T "Tag_Value_2"`	`// Inhalt des Akkumulators 1 in den Operanden "Tag_Value_2" übertragen.`

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

ENT: Akkumulator 2 und 3 in den nächst höheren Akkumulator schieben

Beschreibung

Mit der Anweisung "Akkumulator 2 und 3 in den nächst höheren Akkumulator schieben" schieben Sie den Inhalt des Akkumulators 3 in den Akkumulator 4 und den Inhalt des Akkumulators 2 in den Akkumulator 3. Die Inhalte der Akkumulatoren 1 und 2 bleiben dabei unverändert.

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 bis 4 vor und nach der Ausführung der Anweisung:

Status	Akkumulator			
	1	2	3	4
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert B	Wert B	Wert C

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
*R	// Werte der Akkumulatoren 1 und 2 multiplizieren.
	// Produkt (Zwischenergebnis) im Akkumulator 1 speichern.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 (Zwischenergebnis) in den Akkumulator 2 schieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.
ENT	// Inhalt des Akkumulators 2 (Zwischenergebnis) in den Akkumulator 3 kopieren.
	// Inhalt des Akkumulators 3 in den Akkumulator 4 kopieren.
L "Tag_Value_4"	// Inhalt des Akkumulators 1 ("Tag_Value_3") in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_4" in Akkumulator 1 laden.
+R	// Werte der Akkumulatoren 1 und 2 addieren.
	// Summe im Akkumulator 1 speichern.
	// Wert des Akkumulators 3 (Zwischenergebnis) in den Akkumulator 2 übertragen.
LEAVE	// Inhalt des Akkumulators 3 (Zwischenergebnis) in den Akkumulator 2 schieben.
	// Inhalt des Akkumulators 4 in Akkumulator 3 schieben.
/R	// Werte der Akkumulatoren 1 und 2 dividieren.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis (Inhalt des Akkumulators 1) in den Operanden "Tag_Result" übertragen.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

LEAVE: Akkumulator 3 und 4 in den nächst niedrigeren Akkumulator schieben

Beschreibung

Mit der Anweisung "Akkumulator 3 und 4 in den nächst niedrigeren Akkumulator schieben" schieben Sie den Inhalt des Akkumulators 3 in den Akkumulator 2 und den Inhalt des Akkumulators 4 in den Akkumulator 3. Die Inhalte der Akkumulatoren 1 und 4 bleiben dabei unverändert.

Anweisungen zu arithmetischen Berechnungen enthalten die Funktionalität der Anweisung "Akkumulator 3 und 4 in den nächst niedrigeren Akkumulator schieben".

Die folgende Tabelle zeigt die Inhalte der Akkumulatoren 1 bis 4 vor und nach der Ausführung der Anweisung:

Status	Akkumulator			
	1	2	3	4
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert C	Wert D	Wert D

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
L "Tag_Value_1"	// Wert des Operanden "Tag_Value_1" in den Akkumulator 1 laden.
L "Tag_Value_2"	// Inhalt des Akkumulators 1 in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.
UD	// Wert des Akkumulators 2 ("Tag_Value_1") mit dem Wert des Akkumulators 1 ("Tag_Value_2")nach UND verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
L "Tag_Value_3"	// Inhalt des Akkumulators 1 (Zwischenergebnis) in Akkumulator 2 schieben.
	// Wert des Operanden "Tag_Value_3" in Akkumulator 1 laden.

AWL	Erläuterung
ENT	// Inhalt des Akkumulators 2 (Zwischenergebnis) in den Akkumulator 3 kopieren.
	// Inhalt des Akkumulators 3 in den Akkumulator 4 kopieren.
L "Tag_Value_4"	// Inhalt des Akkumulators 1 ("Tag_Value_3") in den Akkumulator 2 verschieben.
	// Wert des Operanden "Tag_Value_4" in Akkumulator 1 laden.
UD	// Wert des Akkumulators 2 ("Tag_Value_3") mit dem Wert des Akkumulators 1 ("Tag_Value_4")nach UND verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
LEAVE	// Inhalt des Akkumulators 3 (Zwischenergebnis) in Akkumulator 2 schieben.
	// Inhalt des Akkumulators 4 in Akkumulator 3 schieben.
OD	// Wert des Akkumulators 2 mit dem Wert des Akkumulators 1 nach ODER verknüpfen.
	// Ergebnis im Akkumulator 1 speichern.
T "Tag_Result"	// Ergebnis in den Operanden "Tag_Result" übertragen.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Adressregister

+AR1: Akkumulator 1 zu AR1 addieren

Beschreibung

Mit der Anweisung "Akkumulator 1 zu AR1 addieren" addieren Sie einen Wert zum Inhalt des Adressregisters 1. Die Art des im Adressregister 1 stehenden Zeigers und der Operandenbereich bleiben dabei erhalten.

Zur Angabe des zu addierenden Werts gibt es die folgenden Möglichkeiten:

- Angabe über eine Konstante: Die Anweisung addiert den Wert der Konstanten zum Adressregister 1. Der Wert der Konstante muss dem Format eines bereichsinternen Zeigers (POINTER) entsprechen.
- Angabe über den Wert des rechten Worts im Akkumulator 1: Die Anweisung interpretiert den Wert des rechten Worts im Akkumulator 1 als eine 16-Bit Ganzzahl und erweitert diese vorzeichenrichtig auf 24 Bit. Den Wert des Akkumulators 1 addiert die Anweisung anschließend zum Adressregister 1. Zulässig sind Werte von -32768 bis +32767.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Akkumulator 1 zu AR1 addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	POINTER	Wert, der zum Adressregister 1 addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`+AR1 P#10.0`	`// Zeiger zum Adressregister 1 addieren.`
`L "Tag_Value"`	`// Wert des Operanden "Tag_Value" in das rechte Wort des Akkumulators 1 laden.`
`+AR1`	`// Wert des Akkumulators 1 zum Adressregister 1 addieren.`
`TAR1 %MD24`	`// Inhalt des Adressregisters 1 in das Doppelwort MD24 übertragen.`

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

+AR2: Akkumulator 1 zu AR2 addieren

Beschreibung

Mit der Anweisung "Akkumulator 1 zu AR2 addieren" addieren Sie einen Wert zum Inhalt des Adressregisters 2. Die Art des im Adressregister 2 stehenden Zeigers und der Operandenbereich bleiben dabei erhalten.

Zur Angabe des zu addierenden Werts, gibt es die folgenden Möglichkeiten:

- Angabe über eine Konstante: Die Anweisung addiert den Wert der Konstanten zum Adressregister 2. Der Wert der Konstante muss dem Format eines bereichsinternen Zeigers (POINTER) entsprechen.
- Angabe über den Wert des rechten Worts im Akkumulator 1: Die Anweisung interpretiert den Wert des rechten Worts im Akkumulator 1 als eine 16-Bit Ganzzahl und erweitert diese vorzeichenrichtig auf 24 Bit. Den Wert des Akkumulators 1 addiert die Anweisung anschließend zum Adressregister 2. Zulässig sind Werte von -32 768 bis +32 767.

Die CPU führt die Anweisung unabhängig vom Verknüpfungsergebnis und den Statusbits aus. Die Anweisung beeinflusst weder das Verknüpfungsergebnis noch die Statusbits.

Die Inhalte der Akkumulatoren werden durch die Anweisung nicht verändert.

Hinweis

Das Adressregister AR2 wird bei der Bearbeitung von Multiinstanzen verwendet. Wenn Sie die Anweisung "Akkumulator 1 zu AR2 addieren" programmieren, müssen Sie zuvor den Inhalt des Adressregisters 2 "retten" und später wieder laden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Akkumulator 1 zu AR2 addieren":

Parameter	Datentyp	Beschreibung
<Konstante>	POINTER	Wert, der zum Adressregister 1 addiert wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`+AR2 P#10.0`	`// Zeiger zum Adressregister 2 addieren.`
`L "Tag_Value"`	`// Wert des Operanden "Tag_Value" in das rechte Wort des Akkumulators 1 laden.`
`+AR2`	`// Wert des Akkumulators 1 zum Adressregister 2 addieren.`
`TAR2 %MD24`	`// Inhalt des Adressregisters 1 in das Doppelwort MD24 übertragen.`

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Master Control Relay

MCR(: MCR-Bereiche öffnen

Beschreibung

Mit der Anweisung "MCR-Bereiche öffnen" schalten Sie, abhängig vom Verknüpfungsergebnis (VKE), die MCR-Abhängigkeit innerhalb eines MCR-Bereichs ein und aus.

Den Anfang und das Ende eines MCR-Bereichs legen Sie durch die Anweisungen "MCR-Bereich aktivieren" (MCRA) und "MCR-Bereich deaktivieren" (MCRD) fest. Innerhalb des

definierten MCR-Bereichs können Sie für einen oder mehrere Programmteile die MCR-Abhängigkeit ein- und ausschalten.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Die MCR-Abhängigkeit wird eingeschaltet, wenn das VKE vor der Bearbeitung der Anweisung "1" ist. Wenn das VKE vor der Anweisung "0" ist, wird die MCR-Abhängigkeit ausgeschaltet. Das Ende des MCR-abhängigen Programmabschnitts wird durch die Anweisung "MCR-Bereiche schließen" programmiert. Der Programmabschnitt zwischen den beiden Anweisungen wird durch das MCR beeinflusst. Betroffen sind dabei nur die MCR-abhängigen Anweisungen (z. B. die Anweisung "Transferieren" (T))

Das Ein- und Ausschalten der MCR-Abhängigkeit können Sie auch schachteln, indem Sie in einem MCR-abhängigen Programmabschnitt die Anweisung "MCR-Bereiche öffnen" programmieren. Die Anweisung können Sie bis 8 Ebenen tief schachteln. Die MCR-Abhängigkeit des übergeordneten Programmteils steuert dabei die MCR-Abhängigkeiten in allen untergeordneten Programmteilen. Wenn die MCR-Abhängigkeit im übergeordneten Programmteil abgeschaltet ist, kann diese in einem untergeordneten MCR-abhängigen Programmteil nicht eingeschaltet werden.

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins, wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
MCRA	// Anfang des MCR-Bereichs
U "Tag_Input_1"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
MCR(	// Anfang der MCR-Abhängigkeit
	// Beim Signalzustand "1" des VKE, die MCR-Abhängigkeit einschalten.
	// Beim Signalzustand "0" des VKE, die MCR-Abhängigkeit ausschalten.
U "Tag_Input_2"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_1"	// Bei ausgeschalteter MCR-Abhängigkeit Operanden auf "0" zurücksetzen.
	// Bei eingeschalteter MCR-Abhängigkeit Operanden abhängig vom VKE setzen oder zurücksetzen.
U "Tag_Input_3"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

AWL	Erläuterung
MCR(	// Schachtelung der MCR-Abhängigkeit
U "Tag_Input_4"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
U "Tag_Input_5"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_2"	// Bei ausgeschalteter MCR-Abhängigkeit Operanden auf "0" zurücksetzen.
	// Bei eingeschalteter MCR-Abhängigkeit Operanden abhängig vom VKE setzen oder zurücksetzen.
)MCR	// Ende der Schachtelung
L "Tag_InputValue_1"	// Wert des Operanden in Akkumulator 1 laden.
T "Tag_OutputValue_1"	// Bei ausgeschalteter MCR-Abhängigkeit den Wert "0" in den Operanden schreiben.
	// Bei eingeschalteter MCR-Abhängigkeit Inhalt des Akkumulators 1 in den Operanden übertragen.
)MCR	// Ende der MCR-Abhängigkeit
U "Tag_Input_6"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
MCR(	// Anfang der MCR-Abhängigkeit
	// Beim Signalzustand "1" des VKE, die MCR-Abhängigkeit einschalten.
	// Beim Signalzustand "0" des VKE, die MCR-Abhängigkeit ausschalten.
L "Tag_InputValue_2"	// Wert des Operanden in Akkumulator 1 laden.
T "Tag_OutputValue_2"	// Bei ausgeschalteter MCR-Abhängigkeit den Wert "0" in den Operanden schreiben.
	// Bei eingeschalteter MCR-Abhängigkeit Inhalt des Akkumulators 1 in den Operanden übertragen.
)MCR	// Ende der MCR-Abhängigkeit
MCRD	// Ende des MCR-Bereichs
U "Tag_Input_7"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_3"	// Operanden abhängig vom VKE setzen oder zurücksetzen.

Siehe auch

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

)MCR: MCR-Bereiche schließen

Beschreibung

Mit der Anweisung "MCR-Bereiche schließen" schließen Sie einen MCR-abhängigen Programmabschnitt ab. Die Anweisung löscht dabei das VKE aus dem letzten Eintrag im MCR-Stack.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Die MCR-Abhängigkeit programmieren Sie mit der Anweisung "MCR-Bereiche öffnen". Die Anweisungen "MCR-Bereiche öffnen" und "MCR-Bereiche schließen" müssen immer paarweise programmiert werden. Wenn eine der beiden Anweisungen nicht programmiert ist, wird ein MCR-Stackfehler ausgelöst.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
MCRA	// Anfang des MCR-Bereichs
U "Tag_Input_1"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
MCR(	// Anfang der MCR-Abhängigkeit
	// Beim Signalzustand "1" des VKE die MCR-Abhängigkeit einschalten.
	// Beim Signalzustand "0" des VKE die MCR-Abhängigkeit ausschalten.
U "Tag_Input_2"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_1"	// Bei ausgeschalteter MCR-Abhängigkeit Operanden auf "0" zurücksetzen.
	// Bei eingeschalteter MCR-Abhängigkeit Operanden abhängig vom VKE setzen oder zurücksetzen.
L "Tag_InputValue"	// Wert des Operanden in Akkumulator 1 laden.
T "Tag_OutputValue"	// Bei ausgeschalteter MCR-Abhängigkeit den Wert "0" in den Operanden schreiben.
	// Bei eingeschalteter MCR-Abhängigkeit Inhalt des Akkumulators 1 in den Operanden übertragen.
)MCR	// Ende der MCR-Abhängigkeit
MCRD	// Ende des MCR-Bereichs
U "Tag_Input_3"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_2"	// Operanden abhängig vom VKE setzen oder zurücksetzen.

Siehe auch

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

MCRA: MCR-Bereich aktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich aktivieren" legen Sie den Anfang eines MCR-Bereichs fest. Innerhalb dieses Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen ein- und ausschalten:

- MCR(: MCR-Bereiche öffnen
-)MCR: MCR-Bereiche schließen

Das Ende des MCR-Bereichs legen Sie mit der Anweisung "MCR-Bereich deaktivieren" (MCRD) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
MCRA	// Anfang des MCR-Bereichs
U "Tag_Input_1"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
MCR(	// Anfang der MCR-Abhängigkeit
	// Beim Signalzustand "1" des VKE, die MCR-Abhängigkeit einschalten.
	// Beim Signalzustand "0" des VKE, die MCR-Abhängigkeit ausschalten.
U "Tag_Input_2"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.

AWL	Erläuterung
= "Tag_Output_1"	// Bei ausgeschalteter MCR-Abhängigkeit Operanden auf "0" zurücksetzen.
	// Bei eingeschalteter MCR-Abhängigkeit Operanden abhängig vom VKE setzen oder zurücksetzen.
L "Tag_InputValue"	// Wert des Operanden in Akkumulator 1 laden.
T "Tag_OutputValue"	// Bei ausgeschalteter MCR-Abhängigkeit den Wert "0" in den Operanden schreiben.
	// Bei eingeschalteter MCR-Abhängigkeit Inhalt des Akkumulators 1 in den Operanden übertragen.
)MCR	// Ende der MCR-Abhängigkeit
MCRD	// Ende des MCR-Bereichs
U "Tag_Input_3"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_2"	// Operanden abhängig vom VKE setzen oder zurücksetzen.

Siehe auch

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

MCRD: MCR-Bereich deaktivieren

Beschreibung

Mit der Anweisung "MCR-Bereich deaktivieren" legen Sie das Ende eines MCR-Bereichs fest.

Den Anfang eines MCR-Bereichs legen Sie mit dem Anweisung "MCR-Bereich aktivieren" (MCRA) fest. Die Anweisungen zum Definieren des Anfangs und des Endes eines MCR-Bereichs müssen immer paarweise verwendet werden.

Hinweis

Beachten Sie die Informationen im Abschnitt "Wichtige Hinweise zur Benutzung der MCR-Funktionalität" unter "Siehe auch".

Innerhalb eines MCR-Bereichs können Sie die MCR-Abhängigkeit mithilfe der folgenden Anweisungen programmieren:

- MCR(: MCR-Bereiche öffnen
-)MCR: MCR-Bereiche schließen

Wenn Sie innerhalb eines MCR-Bereichs einen Baustein aufrufen, ist im aufgerufenen Baustein die MCR-Abhängigkeit deaktiviert. Mit dem Verlassen eines Bausteins wird die MCR-Abhängigkeit so eingestellt, wie sie vor dem Bausteinaufruf war.

Die Anweisung wird ausgeführt, ohne die Statusbits zu berücksichtigen oder zu beeinflussen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
MCRA	// Anfang des MCR-Bereichs
U "Tag_Input_1"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
MCR(	// Anfang der MCR-Abhängigkeit
	// Beim Signalzustand "1" des VKE, die MCR-Abhängigkeit einschalten.
	// Beim Signalzustand "0" des VKE, die MCR-Abhängigkeit ausschalten.
U "Tag_Input_2"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_1"	// Bei ausgeschalteter MCR-Abhängigkeit Operanden auf "0" zurücksetzen.
	// Bei eingeschalteter MCR-Abhängigkeit Operanden abhängig vom VKE setzen oder zurücksetzen.
L "Tag_InputValue"	// Wert des Operanden in Akkumulator 1 laden.
T "Tag_OutputValue"	// Bei ausgeschalteter MCR-Abhängigkeit den Wert "0" in den Operanden schreiben.
	// Bei eingeschalteter MCR-Abhängigkeit Inhalt des Akkumulators 1 in den Operanden übertragen.
)MCR	// Ende der MCR-Abhängigkeit
MCRD	// Ende des MCR-Bereichs
U "Tag_Input_3"	// Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
= "Tag_Output_2"	// Operanden abhängig vom VKE setzen oder zurücksetzen.

Siehe auch

Wichtige Hinweise zur Benutzung der MCR-Funktionalität (Seite 202)

Setzen der Statusbits bei Anweisungen zur Programmsteuerung (Seite 191)

Grundlagen zum Statuswort (Seite 183)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Nullanweisungen

BLD: Bildaufbau (Nullanweisung)

Beschreibung

Mit der Anweisung "Bildaufbau (Nullanweisung)" führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Die Anweisung dient zur Wiedererkennung von Codesequencen bei der Parameterübergabe oder bei KOP/FUP-Netzwerken. Sie wird automatisch erzeugt, wenn ein KOP- oder FUP-Programm in AWL angezeigt wird. Der Parameterwert ist die Kennnummer der Anweisung und wird vom Programmiergerät erzeugt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bildaufbau (Nullanweisung)":

Parameter	Datentyp	Beschreibung
<Kennnummer>	WORD	Kennnummer der Anweisung

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

NOP 0: Nullanweisung

Beschreibung

Mit der Anweisung "Nullanweisung" mit einem Parameter 0 führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Der Anweisungscode enthält ein Bitmuster mit 16 Nullen. Die Anweisung ist nur für das Programmiergerät wichtig, wenn ein Programm angezeigt wird.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

AWL	Erläuterung
`L "Tag_Value_1"`	`// Wert des Operanden "Tag_Value_1" in den Akkumulator 1 laden.`
`L "Tag_Value_2"`	`// Inhalt des Akkumulators 1 in Akkumulator 2 verschieben.`
	`// Wert des Operanden "Tag_Value_2" in Akkumulator 1 laden.`
`>I`	`// Vergleichen, ob der Wert des Akkumulators 2 größer als der Wert des Akkumulators 1 ist.`

AWL	Erläuterung
U	// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen.
SPB NEXT	// Bei erfüllter Bedingung (VKE = "1")zur Sprungmarke "NEXT" springen.
	// Bei nicht erfüllter Bedingung (VKE = "0")die nächstfolgende Anweisung bearbeiten.
TAK	// Inhalte der Akkumulatoren 1 und 2 tauschen.
NEXT: NOP 0	// Sprungmarke "NEXT"

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

NOP 1: Nullanweisung

Beschreibung

Mit der Anweisung "Nullanweisung" mit einem Parameter 1 führen Sie keine Funktion aus und beeinflussen die Statusbits nicht. Der Anweisungscode enthält ein Bitmuster mit 16 Einsen. Die Anweisung ist nur für das Programmiergerät wichtig, wenn ein Programm angezeigt wird.

Siehe auch

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

4.2.2.4 SCL

Zeiten

IEC-Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Parameter Q für die Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT an. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Signals am Parameter IN. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Parameter IN "0" ist, wird der Parameter ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TP (z. B. "TP_DB")
- Deklaration als lokale Variable vom Typ TP im Abschnitt "Static" eines Bausteins (z. B. #MyTP_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TP();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P	Zeitdauer des Impulses. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Operand, der für die Zeitdauer PT gesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

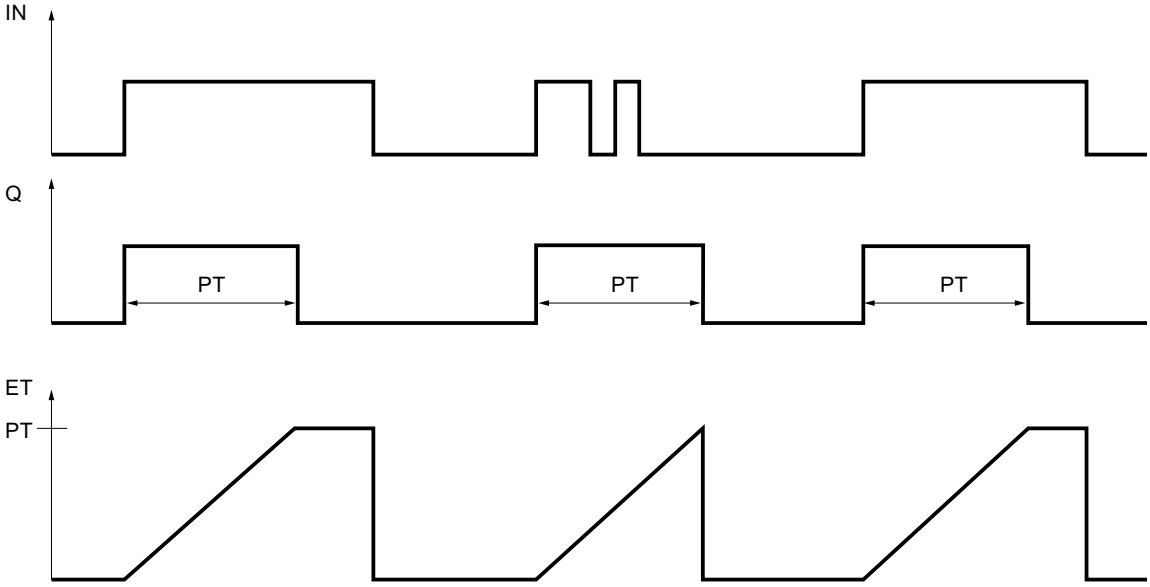

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TP_DB".TP(IN := "Tag_Start",
           PT := "Tag_PresetTime",
           Q => "Tag_Status",
           ET => "Tag_ElapsedTime");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Parameters Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Flanke am Parameter IN erfasst wird.

Am Parameter ET können sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Parameter ET wird zurückgesetzt, sobald der Signalzustand am Parameter IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TON (z. B. "TON_DB")
- Deklaration als lokale Variable vom Typ TON im Abschnitt "Static" eines Bausteins (z. B. #MyTON_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TON();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P	Zeitdauer der Einschaltverzögerung. Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Operand, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

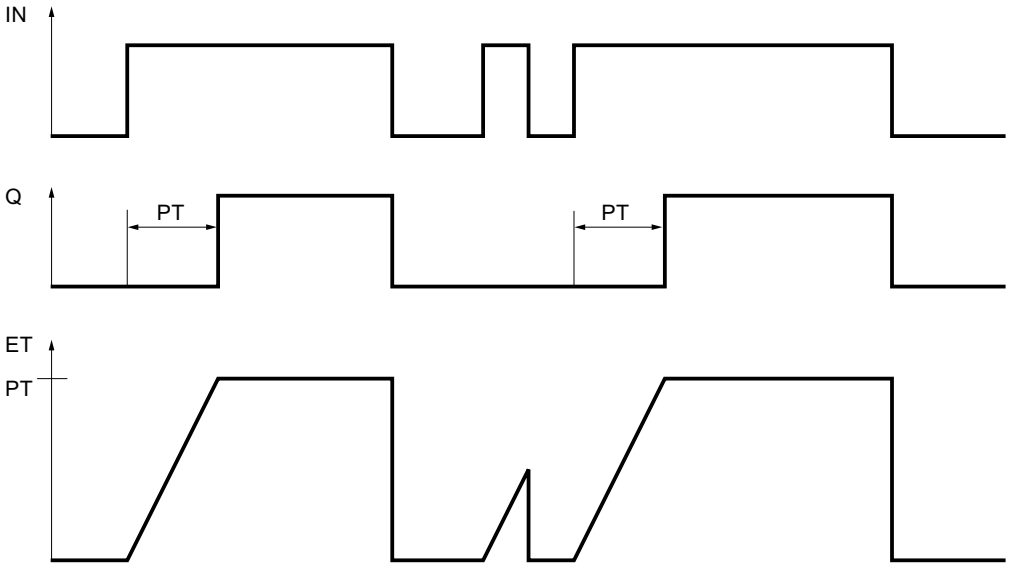

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Start" auf "1" gesetzt, wenn der Operand "Tag_Status" den Signalzustand "1" führt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf "1" gesetzt, solange der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Parameters Q um die programmierte Zeitdauer PT. Der Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der aktuelle Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET solange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt. Wenn der Parameter IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Parameter ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TOF (z. B. "TOF_DB")
- Deklaration als lokale Variable vom Typ TOF im Abschnitt "Static" eines Bausteins (z. B. #MyTOF_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung

erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Zurücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

IEC-Zeit als Instanz-Datenbaustein vom Systemdatentyp IEC_<Zeit> (Shared DB)

Die IEC-Zeit können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zeit_DB>.TOF();

IEC-Zeit als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Die IEC-Zeit können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Timer();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Operand, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagram

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

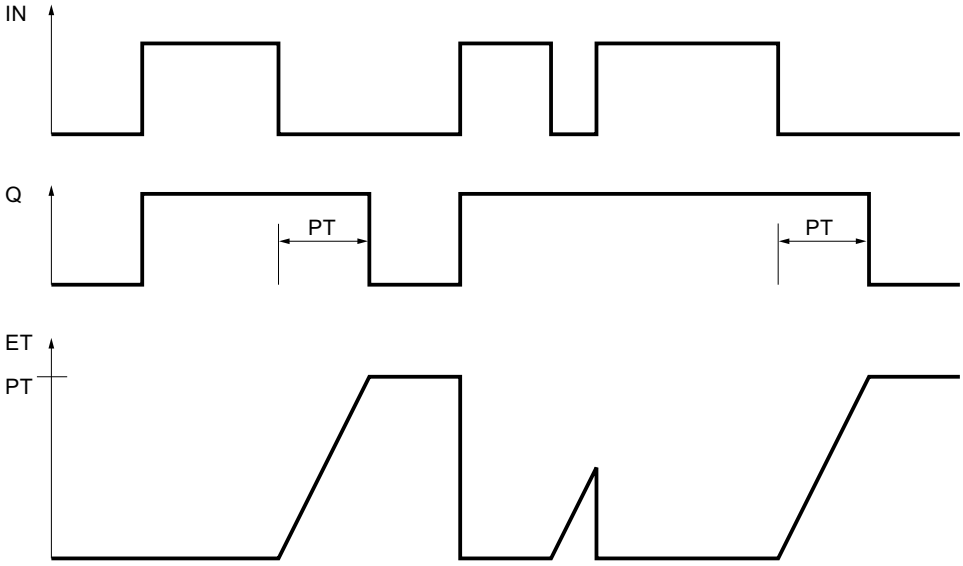

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"TOF_DB".TOF(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Bei einem Wechsel im Signalzustand des Operanden "Tag_Start" von "0" auf "1" wird der Operand "Tag_Status" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

SIMATIC-Zeiten

S_PULSE: Zeit als Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie die am Parameter T_NO programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten und der Parameter Q auf "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Der Parameter Q liefert den Signalzustand "1", solange die Zeit läuft und der Signalzustand am Parameter S "1" ist. Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, liefert der Parameter Q den Signalzustand "0". Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Parameter Q ebenfalls den Signalzustand "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":

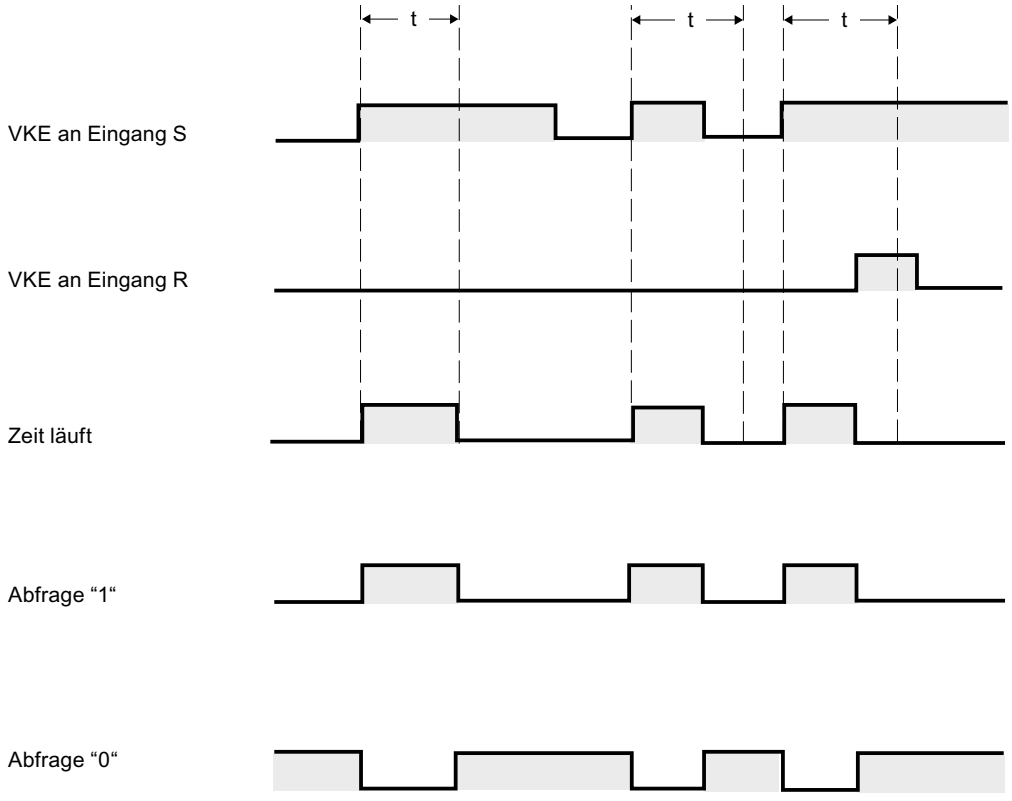

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_PULSE(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "Tag_Number" ab, solange der Operand "Tag_1" den Signalzustand "1" liefert.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird der Operand "Tag_Status" auf "0" zurückgesetzt. Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Operand "Tag_Status" ebenfalls den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu SCL (Seite 7607)

S_PEXT: Zeit als verlängerten Impuls parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt. Solange die Zeit läuft, liefert der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist, wird der Parameter Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der am Parameter TV programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Parameter R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Parameter R hat keine Auswirkungen.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

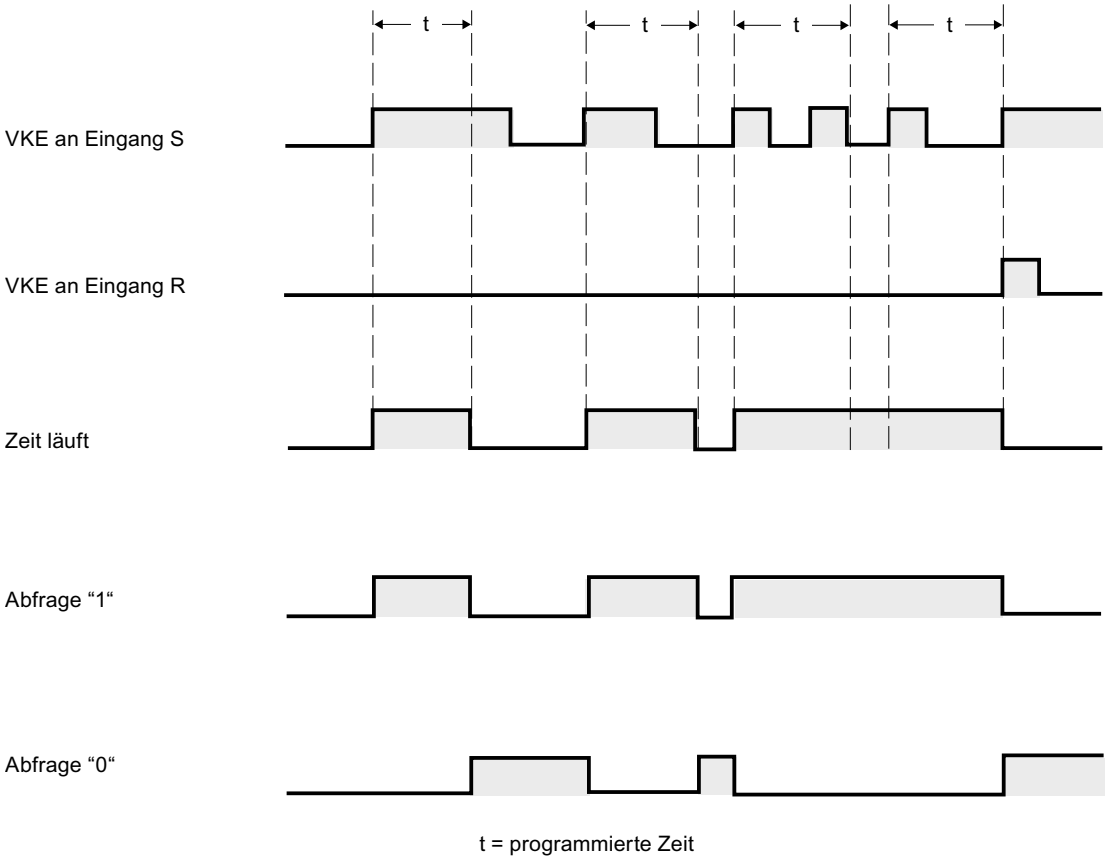

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_PEXT(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Solange die Zeit läuft, liefert der Operand "Tag_Status" den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Operand "Tag_Status" auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu SCL (Seite 7607)

S_ODT: Zeit als Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und am Parameter S der Signalzustand "1" noch anliegt, liefert der Parameter Q den Signalzustand "1". Wechselt der Signalzustand am Parameter S von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Parameter Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Parameter R zurückgesetzt, auch wenn die Zeit nicht läuft und das Verknüpfungsergebnis (VKE) am Parameter S "1" ist.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

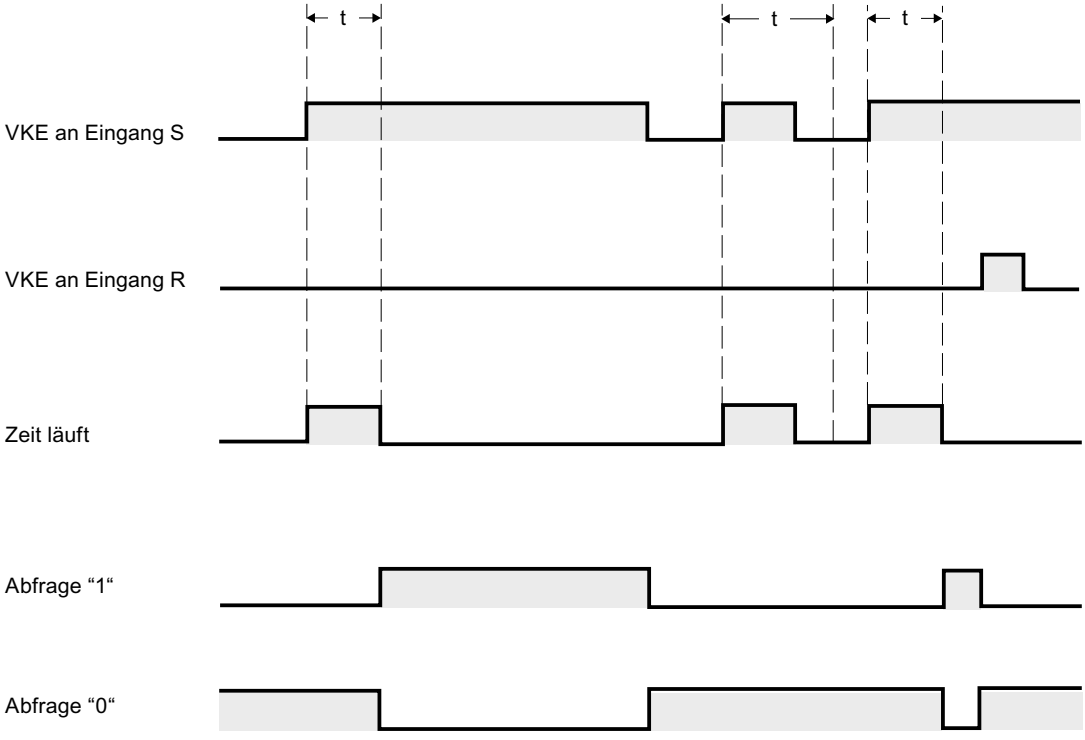

t = programmierte Zeit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_ODT(T_NO := "Timer_1",
                      S := "Tag_1",
                      TV := "Tag_Number",
                      R := "Tag_Reset",
                      Q := "Tag_Status",
                      BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab, solange der Signalzustand des Operanden "Tag_1" "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und der Operand "Tag_Status" den Signalzustand "1" hat, wird der Operand "Tag_Status" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "Tag_1" von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Operand "Tag_Status" liefert in diesem Fall den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu SCL (Seite 7607)

S_ODTS: Zeit als speichernde Einschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt.

Wenn die Zeit abgelaufen ist, liefert der Parameter Q den Signalzustand "1" unabhängig vom Signalzustand am Parameter S. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer TV neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Parameter S. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

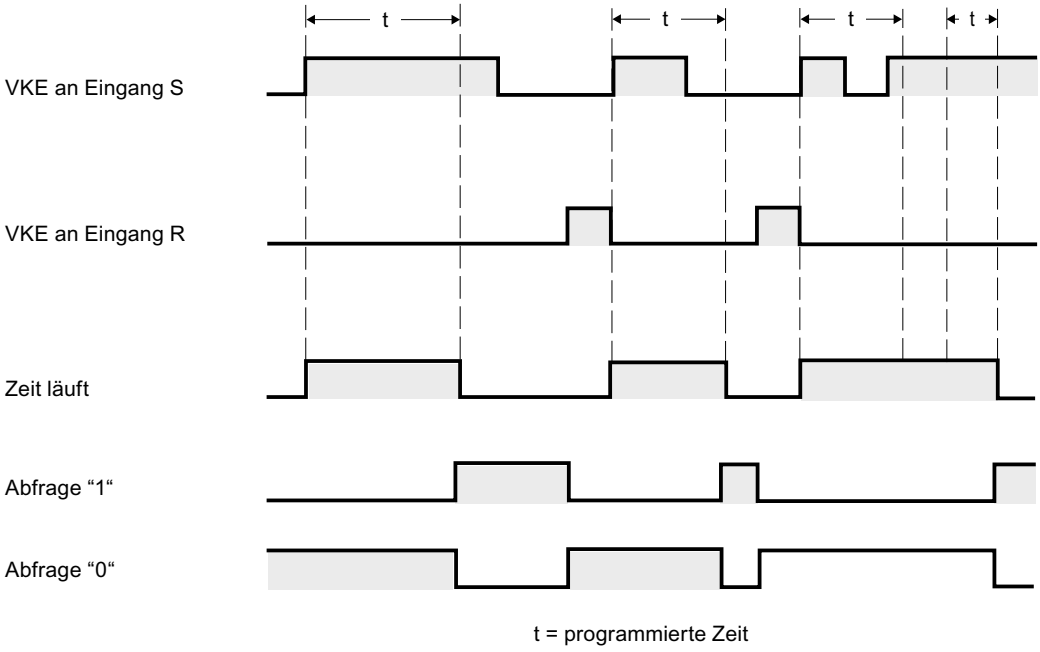

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_ODTS(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "0" auf "1" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab.

Wenn die Zeit abgelaufen ist, liefert der Operand "Tag_Status" den Signalzustand "1" unabhängig vom Signalzustand des Operanden "Tag_1". Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu SCL (Seite 7607)

S_OFFDT: Zeit als Ausschaltverzögerung parametrieren und starten

Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine negative Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab. Solange die Zeit läuft oder der Parameter S den Signalzustand "1" liefert, führt der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Parameter Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
T_NO	Input	TIMER, INT	T	Zeit, die gestartet wird. Anzahl der Zeiten ist von der CPU abhängig.
S	Input	BOOL	E, A, M, D, L	Starteingang
TV	Input	S5TIME, WORD	E, A, M, D, L	Voreingestellter Zeitwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R	Input	BOOL	E, A, M, D, L,P	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L, P	Status der Zeit
BI	Output	WORD	E, A, M, D, L, P	Aktueller Zeitwert dualcodiert
Funktionswert		S5TIME	E, A, M, D, L	Aktueller Zeitwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

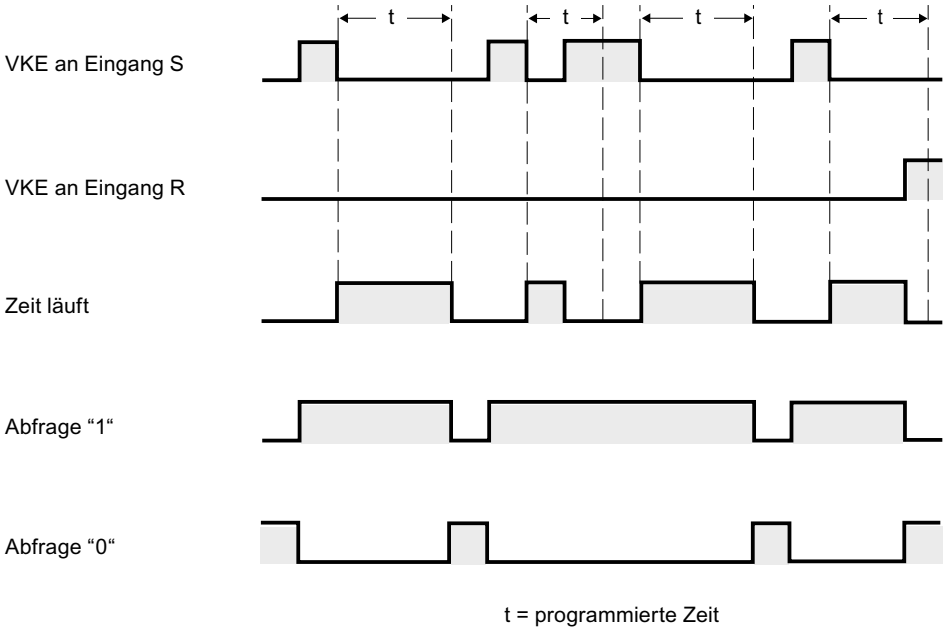

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_OFFDT(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_1" von "1" auf "0" wechselt, wird die Zeit "Timer_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag_Number" ab. Solange die Zeit läuft oder der Operand "Tag_1" den Signalzustand "1" liefert, führt der Operand "Tag_Status" den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "Tag_1" "0" ist, wird der Operand "Tag_Status" auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1" während die Zeit läuft, wird die Zeit zurückgesetzt. Erst wenn eine negative Flanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

L: Zeitwert laden (Seite 4945)

Grundlagen zu SCL (Seite 7607)

Zähler

IEC-Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTU (z. B. "CTU_DB")
- Deklaration als lokale Variable vom Typ CTU im Abschnitt "Static" eines Bausteins (z. B. #MyCTU_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTU();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Syntax

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
PV	Input	INT	E, A, M, D, L, P	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"CTU_DB".CTU(CU := "Tag_Start",
             R := "Tag_Reset",
             PV := "Tag_PresetValue",
             Q => "Tag_Status",
             CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CounterValue" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag_CounterValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps (INT) erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0".

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTD (z. B. "CTD_DB")
- Deklaration als lokale Variable vom Typ CTD im Abschnitt "Static" eines Bausteins (z. B. #MyCTD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben. Falls Instanzen der Anweisung "Rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTD();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"CTD_DB".CTD(CD := "Tag_Start",
             LD := "Tag_Load",
             PV := "Tag_PresetValue",
             Q => "Tag_Status",
             CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Rückwärts zählen" ausgeführt und der Wert des Operanden "Tag_CounterValue" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des Datentyps (INT = -32768) erreicht ist.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag_CounterValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des angegebenen Datentyps (INT) erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hochgezählt. Wenn der untere Grenzwert des angegebenen Datentyps (INT) erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Parametern CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange der Parameter R den Signalzustand "1" führt, hat eine Änderung im Signalzustand der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTUD (z. B. "CTUD_DB")
- Deklaration als lokale Variable vom Typ CTUD im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

IEC-Zähler als Datenbaustein vom Systemdatentyp IEC_<Zähler> (Shared DB)

Den IEC-Zähler können Sie als Datenbaustein wie folgt deklarieren:

<IEC_Zähler_DB>.CTUD();

IEC-Zähler als lokale Variable aus der Bausteinschnittstelle (Multiinstanz)

Den IEC-Zähler können Sie als lokale Variable wie folgt deklarieren:

#myLocal_Counter();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status des Vorwärtszählers
QD	Output	BOOL	E, A, M, D, L	Status des Rückwärtszählers
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"CTUD_DB".CTUD(CU := "Tag_Start1",
               CD := "Tag_Start2",
               LD := "Tag_Load",
               R := "Tag_Reset",
               PV := "Tag_PresetValue",
               QU => "Tag_CU_Status",
               QD => "Tag_CD_Status",
               CV => "Tag_CounterValue");
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_Start1" vorliegt, wird der aktuelle Zählwert um eins erhöht und im Operanden "Tag_CounterValue" abgelegt. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_Start2" vorliegt, wird der Zählwert um eins verringert und ebenfalls im Operanden "Tag_CounterValue" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Parameter CU so lange erhöht, bis er den oberen Grenzwert von 32 767 erreicht. Bei einer positiven Signalflanke am Parameter CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32 768 erreicht.

Der Operand "Tag_CU_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_CU_Status" den Signalzustand "0".

Der Operand "Tag_CD_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_CD_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Instanzen (Seite 64)

Grundlagen zu SCL (Seite 7607)

SIMATIC-Zähler

S_CU: Parametrieren und vorwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Eingang CU "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CU und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L, P	Aktueller Zählwert
Funktionswert		WORD	E, A, M, D, L, P	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CU(C_NO := "Counter_1",
                     CU := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Wenn der Signalzustand am Parameter "Tag_Start" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "Tag_Value" gespeichert.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

S_CD: Parametrieren und rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Parameter CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig.
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L, P	Aktueller Zählwert
Funktionswert		WORD	E, A, M, D, L, P	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CD := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
```

SCL

```
                    CV => "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag_Value" gespeichert.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

S_CUD: Parametrieren und vorwärts/rückwärts zählen

Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Wenn in einem Programmzyklus an den Parametern CU und CD eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) an den Parametern CU und CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Parametern CU, CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
C_NO	Input	COUNTER, INT	Z	Zähler Anzahl der Zähler ist von der CPU abhängig
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
S	Input	BOOL	E, A, M, D, L	Eingang zum Voreinstellen des Zählers
PV	Input	WORD	E, A, M, D, L, P	Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format
R	Input	BOOL	E, A, M, D, L	Rücksetzeingang
Q	Output	BOOL	E, A, M, D, L	Status des Zählers
CV	Output	WORD	E, A, M, D, L, P	Aktueller Zählwert (Hexadezimal)
Funktionswert		WORD	E, A, M, D, L, P	Aktueller Zählwert im BCD-Format

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := S_CUD(C_NO := "Counter_1",
                      CU := "Tag_CU",
                      CD := "Tag_CD",
                      S := "Tag_1",
                      PV := "Tag_PresetValue",
                      R := "Tag_Reset",
                      Q => "Tag_Status",
                      CV => "Tag_Value");
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_CU" erfasst wird und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag_CD" erfasst wird und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand des Operanden "Tag_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag_Value" gespeichert.

Der Operand "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag_Value" abgelegt als auch als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Mathematische Funktionen

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag eines Eingangswertes und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Absolutbetrag des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

Der Absolutbetrag des Eingangswerts wird im Format des Eingangswerts als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	-2
Tag_Result1	2
Tag_Value1	4
Tag_Value2	-1
Tag_Result2	4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der angegebenen Eingänge und geben den kleinsten Wert als Ergebnis aus. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Für die Ausführung der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := MIN(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	12222
IN2	Tag_Value2	14444
IN3	Tag_Value3	13333
Funktionswert	Tag_Result	12222

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("Tag_Value1") in den Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der angegebenen Eingänge und geben den größten Wert als Ergebnis aus. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Für die Ausführung der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Zweiter Eingangswert
INn	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Zusätzlich eingefügte Eingänge, deren Werte verglichen werden
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := MAX(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	Tag_Value1	12222
IN2	Tag_Value2	14444
IN3	Tag_Value3	13333
Funktionswert	Tag_Result	14444

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("Tag_Value2") in den Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert des Parameters IN auf die Werte der Parameter MN und MX. Der Wert des Parameters MN darf dabei nicht größer als der Wert des Parameters MX sein.

Wenn der Wert des Parameters IN die Bedingung MN <= IN <= MX erfüllt, wird er als Ergebnis der Anweisung ausgeben. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Wert des Parameters MN als Ergebnis ausgegeben. Bei einer Überschreitung der Obergrenze MX wird der Wert des Parameters MX als Ergebnis ausgegeben.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert.

Die Ausführung der Anweisung setzt voraus, dass die Operanden aller Parameter vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Obergrenze
Funktionswert		Ganzzahlen, Gleitpunktzahlen, TIME, DATE, DT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := LIMIT(MN := "Tag_Minimum",
                      IN := "Tag_Value",
                      MX := "Tag_Maximum");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_Minimum	12000
IN	Tag_Value	8000
MX	Tag_Maximum	16000
Funktionswert	Tag_Result	12000

Der Wert des Operanden "Tag_Value" wird mit den Werten der Operanden "Tag_Minimum" und "Tag_Maximum" verglichen. Da der Wert des Operanden "Tag_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_Minimum" in den Operanden "Tag_Result" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den Eingangswert und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := SQR("Tag_Value");
"Tag_Result2" := SQR((SQR("Tag_Value1"))*"Tag_Value2");
```

Das Quadrat des Eingangswerts wird im Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	2.5
Tag_Result1	6.25
Tag_Value1	6.0
Tag_Value2	2.0
Tag_Result2	5184.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem Eingangswert die Quadratwurzel und speichern das Ergebnis im angegebenen Operanden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück. Wenn der Eingangswert "0" ist, ist das Ergebnis auch "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := SQRT("Tag_Value");
"Tag_Result2" := SQRT((SQR("Tag_Value1"))+"Tag_Value2");
```

Die Quadratwurzel des Eingangswerts wird im Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	4.0
Tag_Result1	2.0
Tag_Value1	3.0
Tag_Value2	16.0
Tag_Result2	5.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem Eingangswert den natürlichen Logarithmus zur Basis e (e = 2,718282). Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := LN("Tag_Value");
"Tag_Result2" := LN("Tag_Value1"+"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	2.5
Tag_Result1	0.916
Tag_Value1	1.5
Tag_Value2	3.2
Tag_Result2	1.548

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem Eingangswert und speichern das Ergebnis im angegebenen Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := EXP("Tag_Value");
"Tag_Result2" := EXP("Tag_Value1"/"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Resultxy" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	20.5
Tag_Result1	799 902 200
Tag_Value1	15.5
Tag_Value2	30.2
Tag_Result2	1.671

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SIN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+1.570796 (π/2)
Tag_Result	1.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := COS("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+1.570796 (π/2)
Tag_Result	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert (Größe eines Winkels im Bogenmaß)
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := TAN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	+3.141593 (π)
Tag_Result	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus einem Sinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Sinuswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL
```
"Tag_Result" := ASIN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	1.0
Tag_Result	+1.570796 (π/2)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus einem Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von 0 bis +π liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Cosinuswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL
```
"Tag_Result" := ACOS("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	0
Tag_Result	+1.570796 (π/2)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus einem Tangenswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	Gleitpunktzahlen	E, A, M, D, L, P	Tangenswert
Funktionswert		Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ATAN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value	1,0
Tag_Result	+0,785398 (π/4)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu SCL (Seite 7607)

Verschieben

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

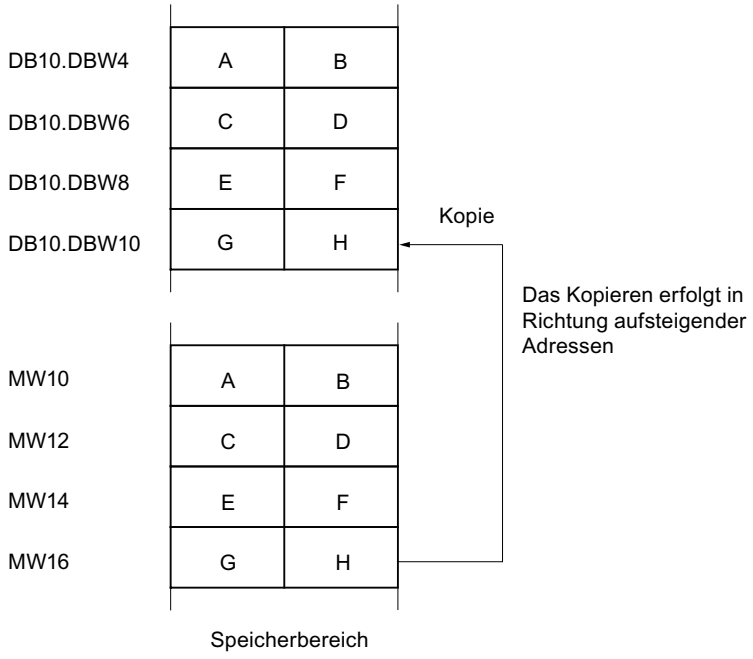

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Unterbrechbarkeit

Solange der Quellbereich nicht Teil eines Datenbausteins ist, der nur im Ladespeicher vorhanden ist, gibt es keine Begrenzung der Schachtelungstiefe.

Bei Unterbrechung einer BLKMOV-Bearbeitung hingegen, bei der aus einem nicht ablaufrelevanten DB kopiert wird, kann eine solche BLKMOV-Bearbeitung nicht mehr eingeschachtelt werden.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Nicht ablaufrelevante Datenbausteine

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Regeln beim Kopieren von nicht ablaufrelevanten Datenbausteinen

Der Quellbereich kann auch in einem nicht ablaufrelevanten Datenbaustein im Ladespeicher liegen. Datenbausteine, die nicht ablaufrelevant sind, werden mit dem Schlüsselwort UNLINKED gekennzeichnet.

Wenn ein nicht ablaufrelevanter Datenbaustein mit der Anweisung "Bereich kopieren" in den Arbeitsspeicher kopiert und gleichzeitig z. B. durch das Programmiergerät geladen wird, kann die Bearbeitung der Anweisung um mehrere Millisekunden verzögert werden. Dies führt zu

einer Verlängerung des OB-Zyklus und kann zum Ansprechen der Zykluszeitüberwachung führen.

Wenn mit der Anweisung "Bereich kopieren" ein nicht ablaufrelevanter Datenbaustein kopiert wird und der Kopiervorgang unterbrochen wird, kann die Bearbeitung der Anweisung nicht mehr fortgesetzt werden.

Falls auf Ihrer CPU die Anweisung "Aus Datenbaustein im Ladespeicher lesen" vorhanden ist, müssen Sie diese Anweisung zum Lesen von nicht ablaufrelevanten Datenbausteinen im Ladespeicher benutzen. Wenn Sie die Anweisung "Bereich kopieren" benutzen, wird der Fehler W#16#8092 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
DSTBLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
Funktionswert (RET_VAL)		INT	E, A, M, D, L, P	Fehlerinformationen
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := BLKMOV(SRCBLK := P#M100.0 BYTE 10,
                       DSTBLK => P#DB1.DBX0.0 BYTE 10);
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 512 Bytes kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn Quell- und Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
DSTBLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
Funktionswert (RET_VAL)		INT	E, A, M, D, L P	Fehlerinformationen
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := UBLKMOV(SRCBLK := P#M100.0 BYTE 10,
                        DSTBLK => P#DB1.DBX0.0 BYTE 10);
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_RetVal" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

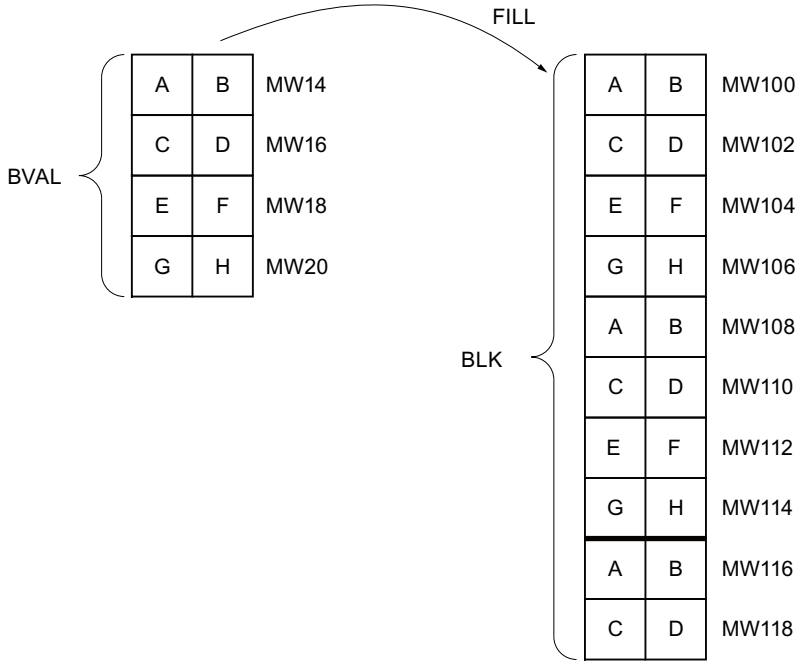

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
BLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
Funktionswert (RET_VAL)		INT	E, A, M, D, L, P	Fehlerinformationen
1) Der Parameter BLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter BVAL

Wenn Sie als Eingangsparameter eine Struktur übergeben, dann müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_RetVal" := FILL(BVAL := P#M14.0 WORD 4,
                     BLK => P#M100.0 WORD 10);
```

Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" programmieren Sie explizite Konvertierungen. Die zu konvertierenden Datentypen bestimmen Sie in einem Dialog, der automatisch beim Einfügen der Anweisung geöffnet wird. Bei der Ausführung liest die Anweisung den Quellwert und konvertiert diesen in den angegebenen Zielwert.

Weitere Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Quelltyp>	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird.
<Zieltyp>	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Datum und Uhrzeit, Zeichenfolgen	E, A, M, D, L, P	Ergebnis der Konvertierung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_INT" := REAL_TO_INT("Tag_REAL");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Datentyp	Wert
Tag_REAL	REAL	20.56
Tag_INT	INT	21

Bei der Konvertierung wird der Wert des Operanden "Tag_REAL" zur nächsten Ganzzahl gerundet und im Operanden "Tag_INT" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Explizite Konvertierungen (Seite 592)

Grundlagen zu SCL (Seite 7607)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl oder eine Gleitpunktzahl um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	REAL	E, A, M, D, L, P	Eingangswert, der gerundet wird.
Funktionswert		DINT	E, A, M, D, L, P	Ergebnis der Rundung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROUND("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	1.50000000	-1.50000000
Tag_Result	2	-2

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie einen Wert zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst höhere ganze Zahl um. Der Funktionswert kann dabei größer oder gleich dem Eingangswert sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	REAL	E, A, M, D, L, P	Eingangswert
Funktionswert		DINT	E, A, M, D, L, P	Aufgerundeter Eingangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := CEIL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0,5	-0,5
Tag_Result	1	0

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie einen Wert zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst niedrigere ganze Zahl um. Der Funktionswert kann dabei kleiner oder gleich dem Eingangswert sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	REAL	E, A, M, D, L, P	Eingangswert
Funktionswert		DINT	E, A, M, D, L, P	Abgerundeter Eingangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := FLOOR("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0.5	-0.5
Tag_Result	0	-1

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Eingangswert eine Ganzzahl ohne Rundung. Die Anweisung selektiert nur den ganzzahligen Anteil des Eingangswerts und gibt diesen ohne die Nachkommastellen als Funktionswert zurück.

Um den Datentyp der Anweisung zu verändern, verwenden Sie die folgende Syntax:

TRUNC_<Datentyp>();

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Ausdruck>	Input	REAL	E, A, M, D, L, P	Eingangswert
_<Datentyp>		Ganzzahlen Vorbelegung: DINT		Datentyp des Funktionswerts: • Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe. • Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.
Funktionswert		Ganzzahlen	E, A, M, D, L	Ganzzahliger Anteil des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result1" := TRUNC("Tag_Value1");
"Tag_Result2" := TRUNC("Tag_Value2"+"Tag_Value3");
"Tag_Result3" := TRUNC_REAL("Tag_Value4");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_Value1	-1.5
Tag_Result1	-1
Tag_Value2	2.1
Tag_Value3	3.2
Tag_Result2	5
Tag_Result3	2.0
Tag_Value4	2.4

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P	Eingabewert, der skaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
Funktionswert (RET_VAL)		WORD	E, A, M, D, L, P	Fehlerinformationen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als 27 648 oder kleiner als 0 (unipolar) bzw. -27 648 (bipolar).
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweiser der Anweisung:

```
"Tag_ErrorCode" := SCALE(IN := "Tag_InputValue",
                         HI_LIM := "Tag_HighLimit"
                         LO_LIM := "Tag_LowLimit"
                         BIPOLAR := "Tag_Bipolar",
                         OUT => "Tag_Result");
```

Die Fehlerinformation der Anweisung wird am Operanden "Tag_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P	Eingabewert, der in einen ganzzahligen Wert deskaliert wird.
HI_LIM	Input	REAL	E, A, M, D, L, P	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
Funktionswert (RET_VAL)		WORD	E, A, M, D, L, P	Fehlerinformationen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_ErrorCode" := UNSCALE(IN := "Tag_InputValue",
                           HI_LIM := "Tag_HighLimit"
                           LO_LIM := "Tag_LowLimit"
                           BIPOLAR := "Tag_Bipolar",
                           OUT => "Tag_Result");
```

Die Fehlerinformation der Anweisung wird am Operanden "Tag_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	22
RET_VAL	Tag_ErrorCode	W#16#0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

Programmsteuerung

IF: Bedingt ausführen

Beschreibung

Mit der Anweisung "Bedingt ausführen" verzweigen Sie den Programmfluss abhängig von einer Bedingung. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Parameter

Je nach Verzweigungsart können Sie die folgenden Formen der Anweisung programmieren:

- Verzweigung durch IF:
  IF <Bedingung> THEN <Anweisungen>
  END_IF;

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung mit der nächsten Anweisung nach END_IF fortgesetzt.

- Verzweigung durch IF und ELSE:
  IF <Bedingung> THEN <Anweisungen1>
  ELSE <Anweisungen0>
  END_IF;

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, werden die Anweisungen ausgeführt, die nach ELSE programmiert sind. Danach wird die Programmbearbeitung mit der nächsten Anweisung nach END_IF fortgesetzt.

- Verzweigung durch IF, ELSIF und ELSE:
  IF <Bedingung1> THEN <Anweisungen1>
  ELSIF <Bedingung2> THEN <Anweisung2>
  ELSE <Anweisungen0>
  END_IF;

Wenn die erste Bedingung (<Bedingung1>) erfüllt ist, werden die Anweisungen (<Anweisungen1>) nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END_IF fortgesetzt.

Wenn die erste Bedingung nicht erfüllt ist, wird die zweite Bedingung (<Bedingung2>) geprüft. Wenn die zweite Bedingung (<Bedingung2>) erfüllt ist, werden die Anweisungen <Anweisungen2> nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END_IF fortgesetzt.

Wenn keine der Bedingungen erfüllt ist, werden die Anweisungen (<Anweisungen0>) nach ELSE ausgeführt und danach wird die Programmbearbeitung nach END_IF fortgesetzt.

Sie können beliebig viele Kombinationen von ELSIF und THEN innerhalb der IF-Anweisung schachteln. Die Programmierung eines ELSE-Zweigs ist optional.

Die Syntax der IF-Anweisung besteht aus den folgenden Teilen:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der ausgewertet wird.
<Anweisungen>	-		Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn keine Bedingung innerhalb der Programmschleife erfüllt ist.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF "Tag_1" = 1
THEN "Tag_Value" := 10;
ELSIF "Tag_2" = 1
THEN "Tag_Value" := 20;
ELSIF "Tag_3" = 1
THEN "Tag_Value" := 30;
ELSE "Tag_Value" := 0;
END_IF;
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_1	1	0	0	0
Tag_2	0	1	0	0
Tag_3	0	0	1	0
Tag_Value	10	20	30	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CASE: Mehrfach verzweigen

Beschreibung

Mit der Anweisung "Mehrfach verzweigen" bearbeiten Sie abhängig vom Wert eines numerischen Ausdrucks eine von mehreren Anweisungsfolgen.

Der Wert des Ausdrucks muss eine Ganzzahl sein. Bei der Ausführung der Anweisung wird der Wert des Ausdrucks mit den Werten mehrerer Konstanten verglichen. Wenn der Wert des Ausdrucks mit dem Wert einer Konstante übereinstimmt, werden die Anweisungen ausgeführt, die direkt nach dieser Konstante programmiert sind. Die Konstanten können dabei die folgenden Werte annehmen:

- Eine Ganzzahl (z. B. 5)
- Ein Bereich aus Ganzzahlen (z. B. 15..20)
- Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)

Die Anweisung können Sie wie folgt deklarieren:

CASE <Variable> OF

<Konstante1>: <Anweisungen1>;

<Konstante2>: <Anweisungen2>;

<KonstanteX>: <AnweisungenX>; // X >= 3

ELSE <Anweisungen0>;

END_CASE;

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Variable>	Ganzzahlen	E, A, M, D, L	Wert, der mit den programmierten Konstantenwerten verglichen wird.
<Konstante>	Ganzzahlen	-	Konstantenwerte, die die Bedingung zur Ausführung einer Anweisungsfolge bilden. Die Konstanten können die folgenden Werte annehmen: • Eine Ganzzahl (z. B. 5) • Ein Bereich aus Ganzzahlen (z. B. 15..20) • Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)
<Anweisung>	-	-	Beliebige Anweisungen, die ausgeführt werden, wenn der Wert des Ausdrucks mit dem Wert einer Konstanten übereinstimmt. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn die Werte nicht übereinstimmen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) übereinstimmt, werden die Anweisungen (<Anweisungen1>) ausgeführt, die direkt nach der ersten Konstante programmiert sind. Danach wird die Programmbearbeitung nach END_CASE fortgesetzt.

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) nicht übereinstimmt, wird dieser mit dem Wert der Konstante verglichen, die als nächste programmiert ist. Auf diese Weise wird die CASE-Anweisung ausgeführt, bis eine Übereinstimmung der Werte vorliegt. Wenn der Wert des Ausdrucks keinem der programmierten Konstantenwerte entspricht, werden die Anweisungen (<Anweisungen0>) ausgeführt, die nach ELSE programmiert sind. ELSE ist ein optionaler Teil der Syntax und kann entfallen.

Die CASE-Anweisung kann auch geschachtelt werden, indem ein Anweisungsblock durch CASE ersetzt wird. END_CASE bildet den Abschluss der CASE-Anweisung.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
CASE "Tag_Value" OF
     0 :
          "Tag_1" := 1;
     1,3,5 :
          "Tag_2" := 1;
     6..10 :
          "Tag_3" := 1;
     16,17,20..25 :
          "Tag_4" := 1;
     ELSE
          "Tag_5" := 1;
END_CASE;
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Werte				
Tag_Value	0	1, 3 , 5	6, 7, 8, 9, 10	16,17, 20, 21, 22, 23, 24, 25	2
Tag_1	1	-	-	-	-
Tag_2	-	1	-	-	-
Tag_3	-	-	1	-	-
Tag_4	-	-	-	1	-
Tag_5	-	-	-	-	1
1: Der Operand wird auf den Signalzustand "1" gesetzt. -: Der Signalzustand des Operanden bleibt unverändert.					

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

FOR: In Zählschleife ausführen

Beschreibung

Mit der Anweisung "In Zählschleife ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Laufvariable innerhalb eines angegebenen Wertebereichs liegt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung

"Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Hinweis

Informationen zur Durchlaufanzahl und zur Laufvariable

Die Durchlaufanzahl kann während der Programmbearbeitung nicht verändert werden.

Aus Performance Gründen sollte die Laufvariable in der Bausteinschnittstelle im Abschnitt "Temp" deklariert werden. Darüberhinaus sollte die Laufvariable während der Programmbearbeitung nicht verändert werden, da die Anweisung sonst, abhängig von der Implementierung, ein undefiniertes Ergebnis liefern kann.

Grenzwerte für FOR-Anweisungen

Um "sichere" FOR-Anweisungen, die nicht endlos laufen, zu programmieren, beachten Sie folgende Regel und Grenzwerte:

FOR <Laufvariable> := <Anfangswert> TO <Endwert> BY <Schrittweite> DO <Anweisungen>;

END_FOR;

Wenn...	...dann	Bemerkung
Anfangswert < Endwert	Endwert < (PMAX - schritt)	Laufvariable läuft in positiver Richtung
Anfangswert > Endwert AND Schrittwert < O	Endwert > (NMAX - schritt)	Laufvariable läuft in negativer Richtung

Grenzwerte

Für die möglichen Datentypen gelten unterschiedliche Grenzwerte:

Datentyp	PMAX	NMAX
Laufvariable vom Typ INT	32767	-32768
Laufvariable vom Typ DINT	2147483647	-2147483648

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Laufvariable>	INT, DINT	E, A, M, D, L	Operand, dessen Wert bei der Schleifenbearbeitung ausgewertet wird. Der Datentyp der Laufvariablen bestimmt den Datentyp der restlichen Parameter.
<Anfangswert>	INT, DINT	E, A, M, D, L	Ausdruck, dessen Wert bei Beginn der Schleifenbearbeitung der Laufvariablen zugewiesen wird.

Parameter	Datentyp	Speicherbereich	Beschreibung
<Endwert>	INT, DINT	E, A, M, D, L	Ausdruck, dessen Wert den letzten Durchlauf der Programmschleife bestimmt. Vor jedem Schleifendurchlauf wird überprüft, welchen Wert die Laufvariable hat: • Endwert nicht erreicht: Die Anweisungen nach DO werden ausgeführt • Endwert erreicht: Die FOR-Schleife wird ein letztes Mal ausgeführt • Endwert überschritten: Die FOR-Schleife wird beendet Eine Veränderung des Endwerts während der Ausführung der Anweisung ist nicht zulässig.
<Schrittweite>	INT, DINT	E, A, M, D, L	Ausdruck, um dessen Wert die Laufvariable nach jedem Schleifendurchlauf erhöht (positive Schrittweite) oder verringert (negative Schrittweite) wird. Die Angabe der Schrittweite ist optional. Wenn keine Schrittweite angegeben ist, wird der Wert der Laufvariablen nach jedem Schleifendurchlauf um 1 erhöht. Eine Veränderung der Schrittweite während der Ausführung der Anweisung ist nicht zulässig.
<Anweisungen>	-		Anweisungen, die bei jedem Schleifendurchlauf ausgeführt werden, solange der Wert der Laufvariablen im Wertebereich liegt. Der Wertebereich wird durch den Anfangs- und Endwert bestimmt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i := 2 TO 8 BY 2
DO "a_array[i] := "Tag_Value"*"b_array[i]";
END_FOR;
```

Der Operand "Tag_Value" wird mit den Elementen (2, 4, 6, 8) der ARRAY-Variablen "b_array" multipliziert. Das Ergebnis wird in den Elementen (2, 4, 6, 8) der ARRAY-Variablen "a_array" eingelesen.

Siehe auch

CONTINUE: Schleifenbedingung erneut prüfen (Seite 1918)

EXIT: Schleife sofort verlassen (Seite 1918)

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

WHILE: Bei erfüllter Bedingung ausführen

Beschreibung

Mit der Anweisung "Bei erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Durchführungsbedingung erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung können Sie wie folgt deklarieren:

WHILE <Bedingung> DO <Anweisungen>;

END_WHILE;

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der vor jedem Schleifendurchlauf ausgewertet wird.
<Anweisungen>	-		Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung nach END_WHILE fortgesetzt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
WHILE
      "Tag_Value1" <> "Tag_Value2"
      DO "Tag_Result"
      := "Tag_Input";
END_WHILE;
```

Solange die Werte der Operanden "Tag_Value1" und "Tag_Value2" ungleich sind, wird dem Operanden "Tag_Result" der Wert des Operanden "Tag_Input" zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

CONTINUE: Schleifenbedingung erneut prüfen (Seite 5190)

EXIT: Schleife sofort verlassen (Seite 5191)

Grundlagen zu SCL (Seite 7607)

REPEAT: Bei nicht erfüllter Bedingung ausführen

Beschreibung

Mit der Anweisung "Bei nicht erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Abbruchbedingung nicht erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung können Sie wie folgt deklarieren:

REPEAT <Anweisungen>;

UNTIL <Bedingung> END_REPEAT;

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
<Anweisungen>	-		Anweisungen, die solange ausgeführt werden, wie die programmierte Bedingung den Wert FALSE hat. Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.
<Bedingung>	BOOL	E, A, M, D, L	Ausdruck, der nach jedem Schleifendurchlauf ausgewertet wird. Wenn der Ausdruck den Wert FALSE hat, wird die Programmschleife erneut bearbeitet. Wenn der Ausdruck den Wert TRUE hat, wird die Programmbearbeitung nach END_REPEAT fortgesetzt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
REPEAT "Tag_Result"
        := "Tag_Value";
UNTIL "Tag_Error"
END_REPEAT;
```

Solange der Wert des Operanden "Tag_Error" den Signalzustand "0" hat, wird dem Operanden "Tag_Result" der Wert des Operanden "Tag_Value" zugewiesen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

CONTINUE: Schleifenbedingung erneut prüfen

Beschreibung

Mit der Anweisung "Schleifenbedingung erneut prüfen" beenden Sie den aktuellen Programmdurchlauf einer FOR-, WHILE- oder REPEAT-Schleife.

Nach der Ausführung der Anweisung werden die Bedingungen zur Fortsetzung der Programmschleife erneut ausgewertet. Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i
     := 1 TO 15 BY 2 DO
     IF (i < 5) THEN
     CONTINUE;
END_IF;
      "DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung i < 5 erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) nicht bearbeitet. Die Laufvariable (i) wird um die Schrittweite "2" erhöht und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.

Wenn die Bedingung i < 5 nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable wird ebenfalls um die Schrittweite "2" erhöht und geprüft.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

EXIT: Schleife sofort verlassen

Beschreibung

Mit der Anweisung "Schleife sofort verlassen" brechen Sie die Bearbeitung einer FOR-, WHILE- oder REPEAT-Schleife, unabhängig von Bedingungen, an beliebiger Stelle ab. Die Programmbearbeitung wird nach dem Abschluss der Schleife (END_FOR, END_WHILE, END_REPEAT) fortgesetzt.

Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
FOR i := 15 TO 1 BY -2 DO
IF (i < 5)
THEN EXIT;
END_IF;
"DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung i < 5 erfüllt ist, wird die Bearbeitung der Schleife abgebrochen. Die Programmbearbeitung wird nach END_FOR fortgesetzt.

Wenn die Bedingung i < 5 nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] :=1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable (i) wird um die Schrittweite "-2" verringert und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable (i) im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

GOTO: Springen

Beschreibung

Mit der Anweisung "Springen" setzen Sie die Programmbearbeitung an einer Stelle fort, die durch eine Sprungmarke gekennzeichnet ist.

Die Sprungmarke und die Anweisung "Springen" müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb eines Bausteins nur einmal vergeben werden. Jede Sprungmarke kann von mehreren Sprunganweisungen angesprungen werden.

Ein Sprung von "außen" in eine Programmschleife ist nicht zulässig, aber ein Sprung von einer Programmschleife nach "außen" ist möglich.

Für die Sprungmarke sind folgende Grammatikregeln zu beachten:

- Buchstaben (a - z, A - Z)
- Eine Kombination von Buchstaben und Zahlen. Dabei ist die Reihenfolge zu beachten, d. h. zuerst die Buchstaben, dann die Zahlen (a - z, A - Z, 0 - 9).
- Sonderzeichen oder eine Kombination von Buchstaben und Zahlen in umgekehrter Reihenfolge, d. h. zuerst die Zahlen, dann die Buchstaben (0 - 9, a - z, A - Z), dürfen nicht verwendet werden.

Die Anweisung können Sie wie folgt deklarieren:

GOTO <Sprungmarke>;

...

<Sprungmarke>: <Anweisungen>

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Beschreibung
<Sprungmarke>	-	Sprungmarke, zu der gesprungen wird
<Anweisungen>	-	Anweisungen, die nach dem Sprung ausgeführt werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
CASE "Tag_Value" OF
1 : GOTO MyLABEL1;
2 : GOTO MyLABEL2;
3 : GOTO MyLABEL3;
ELSE GOTO MyLABEL4;
END_CASE;
MyLABEL1: "Tag_1" := 1;
MyLABEL2: "Tag_2" := 1;
```

SCL

```
MyLABEL3: "Tag_3" := 1;
MyLABEL4: "Tag_4" := 1;
```

Abhängig vom Wert des Operanden "Tag_Value" wird die Programmbearbeitung an der Stelle fortgesetzt, die durch die entsprechende Sprungmarke gekennzeichnet ist. Wenn z. B. der Operand "Tag_Value" den Wert "2" hat, wird die Programmbearbeitung bei der Sprungmarke "MyLABEL2" fortgesetzt. Die Programmzeile, die durch die Sprungmarke "MyLABEL1" gekennzeichnet ist, wird in diesem Fall übersprungen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RETURN: Baustein verlassen

Beschreibung

Mit der Anweisung "Baustein verlassen" beenden Sie die Programmbearbeitung im aktuell bearbeiteten Baustein und setzen sie im aufrufenden Baustein fort.

Am Bausteinende kann die Anweisung entfallen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
IF "Tag_Error" <>0 THEN RETURN;
END_IF;
```

Wenn der Signalzustand des Operanden "Tag_Error" ungleich Null ist, wird die Programmbearbeitung im aktuell bearbeiteten Baustein beendet.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

(*...*): Kommentarabschnitt einfügen

Beschreibung

Mit der Anweisung "Kommentarabschnitt einfügen" fügen Sie einen Kommentarabschnitt ein. Der Text, der innerhalb der Klammern "(*...*)" steht, wird wie ein Kommentar behandelt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
(*Dies ist ein Kommentarabschnitt.*)
```

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

REGION: Programmcode strukturieren

Beschreibung

Mit der Anweisung "Programmcode strukturieren" können Sie den Programmcode innerhalb eines SCL-Bausteins strukturieren, indem Sie ihn in unterschiedliche Bereiche einteilen.

Weitere Informationen zum Arbeiten mit Bereichen finden Sie hier: Arbeiten mit Bereichen (Seite 7646)

Die Anweisung können Sie wie folgt deklarieren:

REGION <Name>

<Anweisungen>

END_REGION

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Datentyp	Speicherbereich	Beschreibung
REGION	-	-	Schlüsselwort, mit dem der Bereich beginnt
<Name>	-	-	Text zur Bezeichnung der REGION
<Anweisungen>	-	-	Programmcode, den die REGION umschließt
END_REGION	-	-	Schlüsselwort, mit dem der Bereich abschließt

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
REGION Feeder System
     // Strukturiert den Quellcode für das Feeder-System
```

SCL

```
   IF "Variable_1" = 0 THEN
      "Variable_2" := 10;
   END_IF;
END_REGION
```

Der Bereich Feeder System umfasst den gesamten Programmcode.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Laufzeitsteuerung

COMPRESS: CPU-Speicher komprimieren

Beschreibung

Mit der Anweisung "CPU-Speicher komprimieren" komprimieren Sie den Arbeitsspeicher der CPU bzw. den RAM-Ladespeicher. Dabei werden Lücken, die durch Löschen oder Laden von Bausteinen entstanden sind, aufgefüllt, indem die vorhandenen Bausteine zusammengeschoben werden.

Mit dem Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Der Komprimiervorgang wird über mehrere Programmzyklen verteilt. Die Überwachung des Komprimiervorgangs erfolgt über die Werte der einzelnen Parameter. Wenn am Parameter BUSY der Anweisung der Wert "1" ansteht, ist der Komprimiervorgang noch in Ausführung. Der Wert "1" am Parameter DONE meldet, dass der Komprimiervorgang beendet ist. Die Fehlerinformationen können am Parameter RET_VAL abgefragt werden.

Wenn bereits ein extern angestoßener Komprimiervorgang ausgeführt wird, führt der Aufruf der Anweisung "CPU-Speicher komprimieren" zur Fehleranzeige.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BUSY	Output	BOOL	E, A, M, D, L	Information, ob Komprimier-vorgang aktiv ist.
DONE	Output	BOOL	E, A, M, D, L	Information, ob der Kompri-miervorgang beendet wurde.
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler Der Status des Komprimiervorgangs kann an den Parametern BUSY und DONE abgefragt werden.
8091	Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt.
8092	Die Anweisung kann nicht ausgeführt werden, da eine der folgenden Bedingungen zutrifft: • Die Funktion "Baustein löschen" ist aktiv. • Eine PG-Funktion greift auf einen zu verschiebenden Baustein zu. • Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt. • Die H-CPU befindet sich einem Ankoppel- oder Aktualisierungsvorgang.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufgerufen wird, wird die Anweisung nicht ausgeführt.

Die Anweisung "Zyklusüberwachungszeit neu starten" läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Gültig für S7-300-CPUs:

Bei den S7-300-CPUs (mit Ausnahme von S7-318) ist die mit der Anweisung programmierte Verzögerungszeit eine Mindestzeit. Sie verlängert sich um die Ausführungszeit der eingeschachtelten Prioritätsklassen sowie um Systemlasten.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

PROTECT: Schutzstufe ändern

Beschreibung

Mit der Anweisung "Schutzstufe ändern" ändern Sie die projektierte Schutzstufe "1" der CPU ab.

Die parametrierte Schutzstufe "1" mit aktivierter Option "Durch Passwort aufhebbar" bewirkt, dass man ein Passwort parametrieren muss. Mit diesem Passwort können Sie auf die mit der Anweisung "Schutzstufe ändern" schreibgeschützte CPU im Betrieb schreibend zugreifen (z. B. Laden).

Die Schutzstufe legen Sie am Parameter MODE der Anweisung fest. Die aktuelle Schutzstufe der CPU wird im Inspektorfenster unter "Eigenschaften > Schutz" angezeigt.

Die folgende Tabelle zeigt, welche Bedeutung die Schutzstufen haben:

Schutzstufe	Bedeutung
1	Alle PG-Funktionen sind erlaubt. Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
2	• Das Programm und die Konfiguration der CPU können nicht geändert werden. Das Programm in der CPU kann gelesen werden. • Die Funktionen für Prozessführung, Prozessbeobachtung und Prozesskommunikation sind erlaubt. • Alle Auskunftsfunktionen sind erlaubt. • Mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreibschutz aufheben. • Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
3	• Auch mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreib- und Leseschutz nicht aufheben. • Wenn zum Zeitpunkt des Aufrufs der Anweisung "Schutzstufe ändern" eine legitimierte Verbindung vorhanden ist, bleibt der Aufruf der Anweisung für diese Verbindung ohne Wirkung.

Wenn Sie für Ihre CPU die Schutzstufe "2" oder "3" projektiert haben, hat der Aufruf der Anweisung "Schutzstufe ändern" keine Wirkung.

Die Wirksamkeit der Schutzstufe "2" bzw. "3" nach Ausführung der Anweisung "Schutzstufe ändern" können Sie der SZL-Teilliste mit der SZL-ID W#16#0232 und dem Index W#16#0004 entnehmen.

Hinweis

Mit der Anweisung "Schutzstufe ändern" können Sie keine niedrigere Schutzstufe einstellen als die, die Sie mit STEP 7 "Hardware konfigurieren" projektiert haben. Sie haben z. B. die Schutzstufe 3 projektiert und anschließend die Anweisung "Schutzstufe ändern" mit dem Parameter MODE = 12 aufgerufen. Dies bewirkt eine Sperre der Passwort-Legitimierung. Ein erneuter Aufruf der Anweisung mit dem Parameter MODE = 0 oder 1 bewirkt, dass die Schutzstufe 3 beibehalten wird und dass der Schreib- und Leseschutz mit Eingabe des gültigen Passworts aufgehoben wird.

Auswirkungen der Betriebszustandsübergänge auf die Schutzstufe

Die folgende Tabelle zeigt, welche Auswirkungen die Betriebszustandsübergänge auf die mit der Anweisung "Schutzstufe ändern" eingestellte Schutzstufe haben.

Aktion	Auswirkungen auf die Schutzstufe
• Betriebsartenschalter auf "STOPP" • Ungepuffertes NETZ-EIN • Urlöschen durch Betriebsartenschalter • Konfigurations- und/oder Programmänderungen im Betriebszustand "STOPP" der S7-400-CPU	Die Schutzstufe wird auf "1" gesetzt. Alle PG-Funktionen sind erlaubt.
• Kaltstart • Neustart (Warmstart)	Die Schutzstufe wird durch das Betriebssystem auf "1" gesetzt. Alle PG-Funktionen sind erlaubt. Bei Bedarf können Sie durch Aufruf der Anweisung "Schutzstufe ändern" in Ihrem Programm die Schutzstufe auf "2" oder "3" setzen.
Gepuffertes NETZ-EIN	Bleibt unverändert
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOPP (durch den Aufruf der Anweisung "Programm beenden" (STP), durch Programmfehler ohne zugehörigen Fehler-OB oder durch Anwenderaktion).	Bleibt unverändert
S7-400: Wiederanlauf durch Anwenderaktion oder automatisch nach NETZ-EIN	Bleibt unverändert

Syntax

Für die Anweisung "Schutzstufe ändern" wird die folgende Syntax verwendet:

SCL

```
PROTECT(MODE := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	WORD	E, A, M, D, L	Schutzstufe Mögliche Werte: • W#16#0000: Einstellen der Schutzstufe "1" • W#16#0001: Einstellen der Schutzstufe "2" mit Passwort-Legitimierung • W#16#000C: Einstellen der Schutzstufe "3" ohne Passwort-Legitimierung
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Tritt während der Bearbeitung ein Fehler auf, wird eine Fehlerinformation ausgegeben.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
8090	Unzulässiger Wert am Parameter MODE.
80C3	Die benötigten Betriebsmittel sind momentan belegt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

CiR: CiR-Vorgang steuern

Beschreibung

Mit der Anweisung "CiR-Vorgang steuern" beeinflussen Sie im Anwenderprogramm den CiR-Vorgang wie folgt:

- Den CiR-Vorgang sperren. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU stets abgelehnt. Die Sperre gilt so lange, bis Sie diese mit der Anweisung "CiR-Vorgang steuern" wieder aufheben.
- Den CiR-Vorgang bedingt sperren, indem Sie am Parameter FRZ_TIME eine Obergrenze für die CiR-Synchronisationszeit festlegen. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU nur dann zugelassen, wenn die in der CPU errechnete CiR-Synchronisationszeit die angegebene Obergrenze nicht überschreitet.
- Den CiR-Vorgang freigeben. Die Obergrenze der CiR-Synchronisationszeit wird auf den Defaultwert 1000 ms gesetzt.
- Ermittelt, ob der CiR-Vorgang freigegeben ist oder nicht. Falls der CiR-Vorgang freigegeben oder bedingt freigegeben ist, erhalten Sie im Parameter A_FT der Anweisung die aktuelle Obergrenze für die CiR-Synchronisationszeit.

Hinweis

Während der CiR-Synchronisationszeit sind die Ausgänge eingefroren und die Eingänge werden nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L	Auftragskennung Zulässige Werte am Parameter MODE sind: • 0: Auskunftsfunktion • 1: CiR-Vorgang freigeben • 2: CiR-Vorgang sperren • 3: CiR-Vorgang bedingt sperren. Die Obergrenze der CiR-Synchronisationszeit legen Sie in FRZ_TIME fest.
FRZ_TIME	Input	TIME	E, A, M, D, L	Obergrenze der CiR-Synchronisationszeit in Millisekunden. Zulässiger Wertebereich: 60 ... 2500 ms (Voreinstellung: 1000 ms) Der Parameter FRZ_TIME ist nur dann relevant, wenn der Parameter MODE auf den Wert "3" gesetzt ist.
A_FT	Output	TIME	E, A, M, D, L	Aktuell gültige Obergrenze der CiR-Synchronisationszeit
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformation Wenn der Parameter MODE den Wert "0" hat, liefert der Parameter RET_VAL Informationen über die Freigabe des CiR-Vorgangs.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt. Dieser Fehlercode kann ausgegeben werden, nur wenn der Parameter MODE auf den Wert "1", "2" oder "3" gesetzt ist.
0001	Der CiR-Vorgang ist freigegeben. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.

Fehler-code* (W#16#...)	Erläuterung
0002	Der CiR-Vorgang ist gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0003	Der CiR-Vorgang ist bedingt gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
8001	Die CPU ist nicht bereit für einen CiR-Vorgang.
8002	Der Wert am Parameter MODE ist nicht zulässig.
8003	Der Wert am Parameter FRZ_TIME ist nicht zulässig.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Mit der Anweisung "CiR-Vorgang steuern" können Sie beispielsweise programmieren, dass kein CiR-Vorgang in Zeitabschnitten angestoßen wird, in denen für die Prozessbearbeitung die maximale Leistungsfähigkeit der CPU benötigt wird.

Dazu muss die Anweisung "CiR-Vorgang steuern" vor Beginn des Zeitabschnitts mit erhöhter Prozessaktivität im Programm aufgerufen und mit dem Wert "2" (CiR-Vorgang sperren) am Parameter MODE ausgeführt werden. Dadurch wird den CiR-Vorgang gesperrt.

Um den CiR-Vorgang nach der Beendigung des Zeitabschnitts mit erhöhter Prozessaktivität wieder freizugeben, muss die Anweisung "CiR-Vorgang steuern" erneut aufrufen und mit dem Wert "1" am Parameter MODE ausführt werden. Sie können alternativ den Parameter MODE auch auf den Wert "3" setzen und dadurch den CiR-Vorgang bedingt sperren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

Wortverknüpfungen

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert des Parameters IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P	Position des Bits im Ausgangswert, das gesetzt wird.
Funktionswert		DWORD	E, A, M, D, L, P	Aktueller Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := DECO(IN := "Tag_Value");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

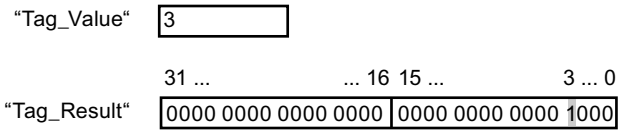

Die Anweisung liest die Zahl "3" aus dem Wert des Operanden "Tag_Value" ab und setzt das dritte Bit im Wert des Operanden "Tag_Result".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben diese als Ergebnis aus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P	Eingangswert
Funktionswert		INT	E, A, M, D, L, P	Bitnummer des Bits im Eingangswert, das ausgelesen wird.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ENCO(IN := "Tag_Value");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

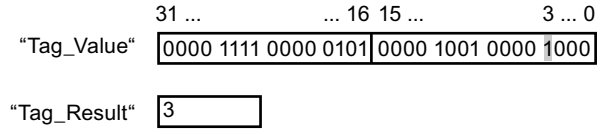

Die Anweisung liest das niederwertigste gesetzte Bit des Operanden "Tag_Value" ab und schreibt die Bitposition "3" in den Operanden "Tag_Result". Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Parameter G) einen der Parameter IN0 oder IN1 aus und geben dessen Inhalt als Ergebnis aus. Wenn der Parameter G den Signalzustand "0" hat, wird der Wert am Parameter IN0 kopiert. Wenn der Parameter G den Signalzustand "1" hat, wird der Wert am Parameter IN1 kopiert und als Funktionswert zurückgeliefert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom Datentyp der gleichen Klasse sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
G	Input	BOOL	E, A, M, D, L	Schalter
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Zweiter Eingangswert
Funktionswert		Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SEL(G := "Tag_Value",
                    IN0 := "Tag_0",
                    IN1 := "Tag_1");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Value	0	1
Tag_0	W#16#0000	W#16#4C
Tag_1	W#16#FFFF	D#16#5E
Tag_Result	W#16#0000	D#16#5E

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MUX: Multiplexen

Beschreibung

Mit der Anweisung "Multiplexen" kopieren Sie den Wert eines ausgewählten Eingangsparameters und geben diesen aus. Durch den Parameter K bestimmen Sie die Nummer des Eingangsparameters, dessen Wert kopiert wird. Die Nummerierung fängt dabei mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Sie können maximal 32 Eingangsparameter deklarieren.

Wenn der Wert des Parameters K größer als die Anzahl der Eingänge ist und der Parameter INELSE nicht gesetzt ist, ist der Funktionswert der Anweisung ungültig und der Freigabeausgang ENO wird auf "0" gesetzt.

Alle parametrierten Variablen müssen vom gleichen Datentyp sein. Der Funktionswert nimmt den höchstwertigsten Datentyp an.

Hinweis

Parameter K hat eine negative Ganzzahl

Wenn Sie an den Eingangsparametern eine Variable mit gültigem Datentyp angeben und der Parameter K eine negative Ganzzahl hat, dann wird der Wert der Variable verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
K	Input	Ganzzahlen	E, A, M, D, L, P	Gibt den Parameter an, dessen Inhalt übertragen wird. • Wenn K = 0 => Parameter IN0 • Wenn K = 1 => Parameter IN1, etc.
IN0	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Zweiter Eingangswert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
INn	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Optionale Eingangswerte
INELSE	Input	Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Gibt den Wert an, der bei K <> n kopiert wird.
Funktionswert		Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, Zeichenfolgen, TOD, DATE, DT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := MUX(K := "Tag_Number",
                    IN0 := "Tag_1",
                    IN1 := "Tag_2",
                    INELSE := "Tag_3");
```

Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_Number	1	4
Tag_1	DW#16#00000000	DW#16#00000000
Tag_2	DW#16#003E4A7D	DW#16#003E4A7D
Tag_3	DW#16#FFFF0000	DW#16#FFFF0000
Tag_Result	DW#16#003E4A7D	DW#16#FFFF0000

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach rechts und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach rechts verschoben.

Die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp INT um vier Bitstellen nach rechts verschoben wird:

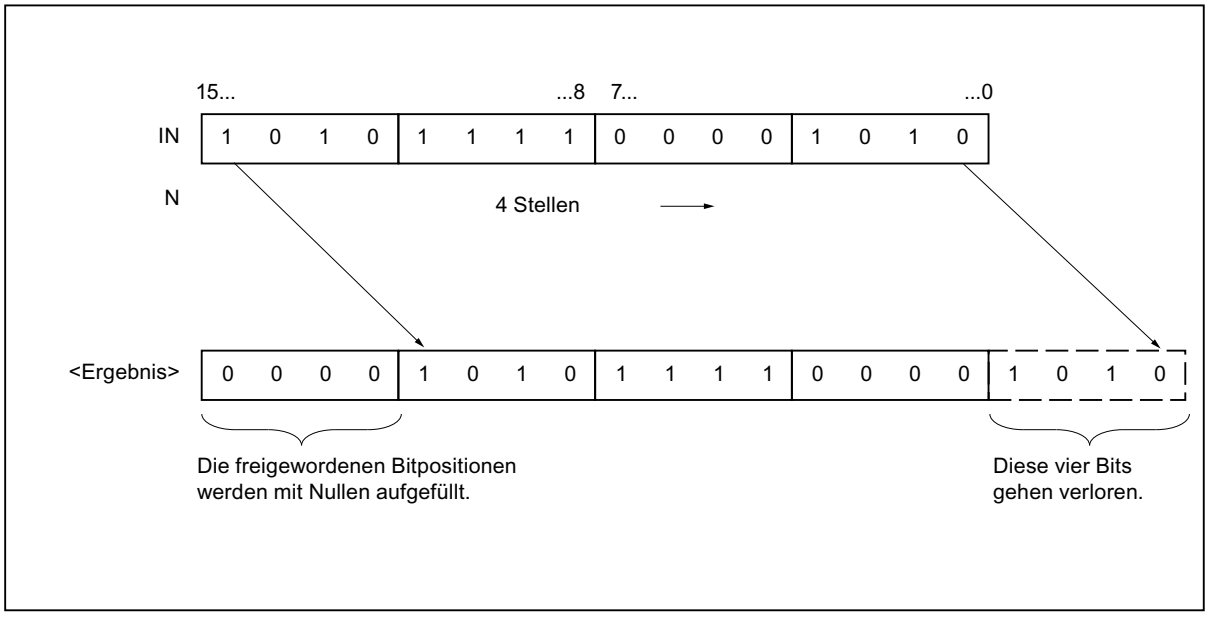

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P	Wert, der verschoben wird.
N	Input	Ganzzahlen	E, A, M, D, L, P	Anzahl der Bits, um die der Wert (IN) verschoben wird. Der Wert am Parameter N muss positiv sein.
Funktionswert		Bitfolgen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := SHR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	3
Funktionswert	Tag_Result	0000 0111 1111 0101

Der Wert des Operanden "Tag_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen werden im Ergebniswert mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

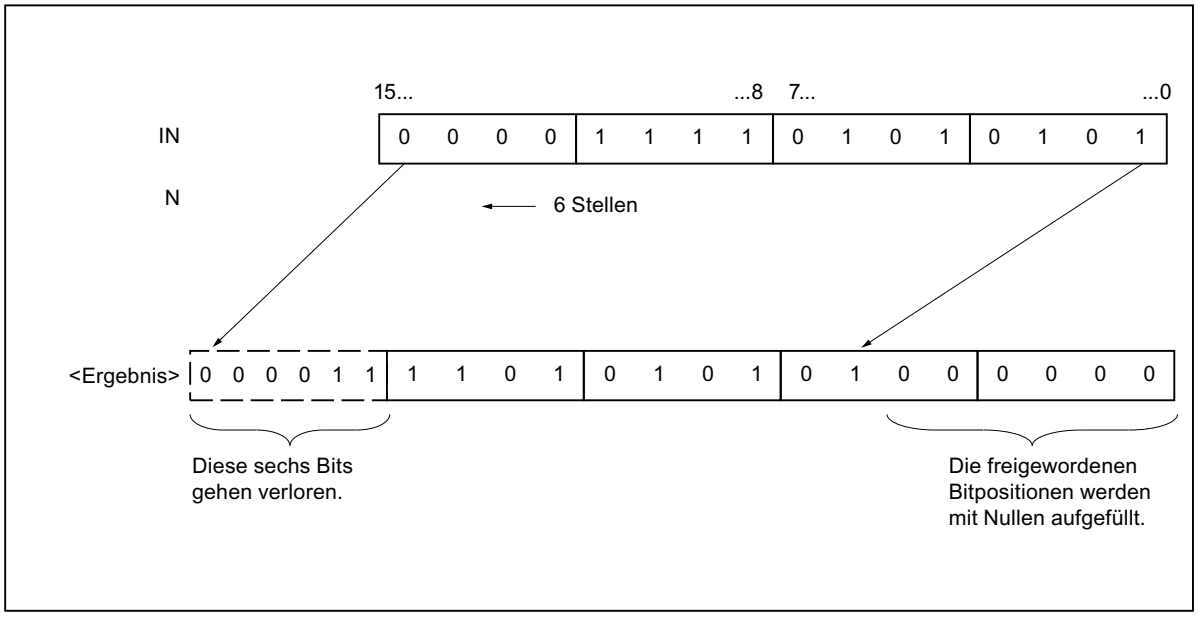

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P	Wert, der verschoben wird
N	Input	Ganzzahlen	E, A, M, D, L, P	Anzahl der Bits, um die der Wert IN verschoben wird. Der Wert am Parameter N muss positiv sein.
Funktionswert		Bitfolgen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := SHL(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	4
Funktionswert	Tag_Result	1111 1010 1111 0000

Der Wert des Operanden "Tag_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach rechts und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

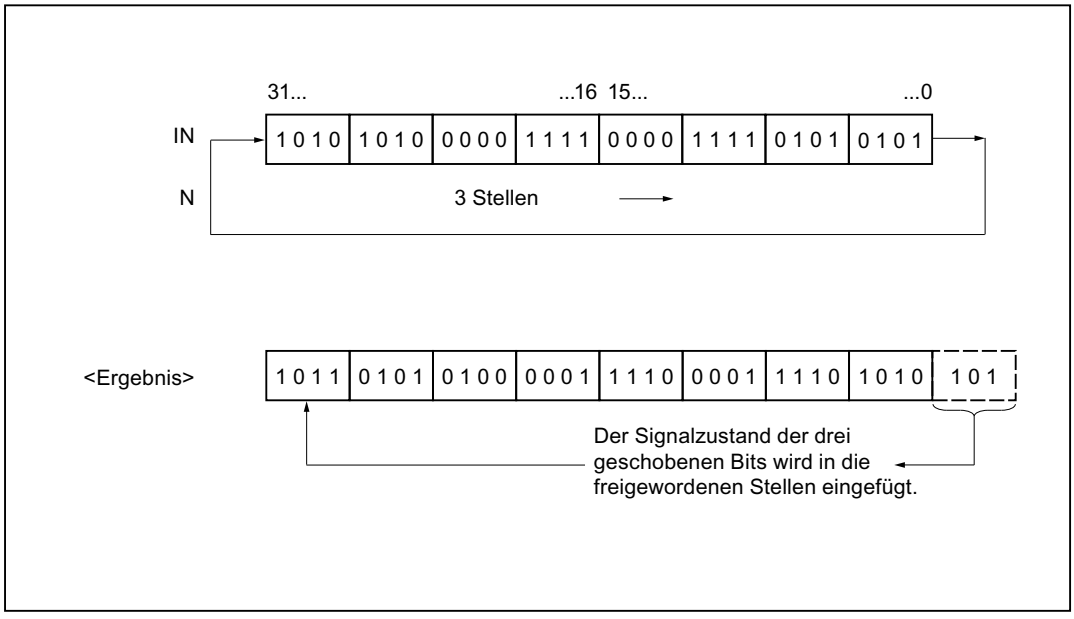

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P	Wert, der rotiert wird
N	Input	Ganzzahlen	E, A, M, D, L, P	Anzahl der Bitstellen, um die der Wert IN rotiert wird. Der Wert am Parameter N muss positiv sein.
Funktionswert		Bitfolgen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
Funktionswert	Tag_Result	1010 1101 0101 0000 0111 1000 0111 1010

Der Inhalt des Operanden "Tag_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

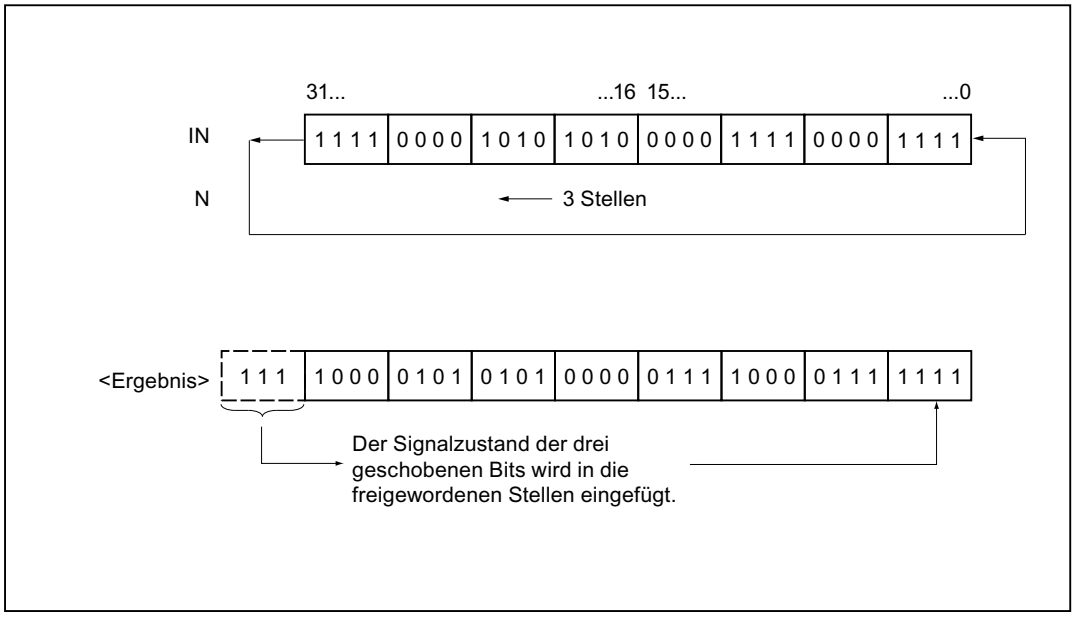

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	Bitfolgen	E, A, M, D, L, P	Wert, der rotiert wird
N	Input	Ganzzahlen	E, A, M, D, L, P	Anzahl der Bitstellen, um die der Wert IN rotiert wird. Der Wert am Parameter N muss positiv sein.
Funktionswert		Bitfolgen	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := ROL(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN	Tag_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
Funktionswert	Tag_Result	0001 0101 0100 0001 1110 0001 1111 1110

Der Inhalt des Operanden "Tag_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag_Result" als Funktionswert zurückgeliefert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

SCL-Anweisungen eingeben (Seite 7642)

SCL-Anweisungen bearbeiten (Seite 7674)

Grundlagen zu SCL (Seite 7607)

Weitere Anweisungen

SET: Bereich bitweise setzen

Beschreibung

Mit der Anweisung "Bereich bitweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "1". Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu setzenden Bits im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Hinweis

Parameter N

Sie können am Parameter N auch ein Element eines Datenbausteins verschalten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der zu setzenden Bits

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SET(S_BIT := P#M0.0,
    N := 10);
```

Im Beispiel setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.1.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SETP: Bitfeld im Peripheriebereich setzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich setzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "1". Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge gesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das gesetzt wird. Die Bits des Feldes müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zu setzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Setzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge. Dies kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Für die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "1" gesetzt.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich setzen" hat keine Wirkung, wenn der Parameter N den Wert "0" liefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L	Anzahl der zu setzenden Bits
SA	Output	POINTER	P	Zeiger auf das erste zu setzende Bit
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Error" := SETP(N := 8,
                    SA => P#10.0);
```

Im Beispiel setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

SETI: Bereich byteweise setzen

Beschreibung

Mit der Anweisung "Bereich byteweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "1". Den Anfangspunkt des Bereichs definieren Sie am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zu setzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich "P" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise setzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P	Zeigt auf das erste Byte des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P	Größe des zu setzenden Bereichs Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SETI(S_BYTE := byte_array[100],
     N := 16);
```

Im Beispiel setzt die Anweisung 2 Bytes (16 Bits) im Bereich von 100.0 bis 101.7 des byte_arrays.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RESET: Bereich bitweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich bitweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu rücksetzenden Bits im angegebenen Bereich, legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P	Anzahl der Bits, die zurückgesetzt werden.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
RESET(S_BIT := %M0.0,
      N := 10);
```

Im Beispiel setzt die Anweisung 10 Bits im Bereich von M0.0 bis M1.1 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

RESETP: Bitfeld im Peripheriebereich rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich rücksetzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "0" zurück. Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge zurückgesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das zurückgesetzt wird. Die Bits des Felds müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zurückzusetzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Rücksetzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bits, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bits im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bits auf den Wert der zugehörigen Bits im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bits, die zum definierten Bitfeld gehören, werden auf den Signalzustand "0" zurückgesetzt.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich rücksetzen" hat keine Wirkung, wenn der Parameter N den Wert "0" liefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L	Anzahl der Bits, die zurückgesetzt werden.
SA	Output	POINTER	P	Zeiger auf das erste Bit, das zurückgesetzt wird.
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Error" := RESETP(N := 8,
                      SA => P#10.0);
```

Im Beispiel setzt die Anweisung 8 Bits im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

RESETI: Bereich byteweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich byteweise rücksetzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich von Bytes auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zurückzusetzenden Bits am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich "P" im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bits im Prozessabbild der Ausgänge (A) wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise rücksetzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie (P).
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P	Zeigt auf das erste Byte des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P	Größe des Bereichs, dessen Bits zurückgesetzt werden. Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
RESETI(S_BYTE := byte_array[100],
       N := 16);
```

Im Beispiel setzt die Anweisung 2 Bytes (16 Bits) im Bereich von 100.0 bis 101.7 des byte_arrays zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

REPL_VAL: Ersatzwert eintragen

Beschreibung

Mit der Anweisung "Ersatzwert eintragen" tragen Sie von einem Synchronfehler-Organisationsbaustein aus einen Ersatzwert in den Akkumulator 1 (AKKU 1) ein.

Hinweis

Die Anweisung "Ersatzwert eintragen" dürfen Sie nur in Synchronfehler-Organisationsbausteinen (OB 121, OB 122) aufrufen.

Die Anweisung "Ersatzwert eintragen" verwenden Sie, wenn Sie von einer Baugruppe keinen Wert mehr lesen können. Nach jedem Zugriff auf diese Baugruppe wird der OB 122 gestartet. Mit dem Aufruf der Anweisung "Ersatzwert eintragen" können Sie bewirken, dass ein Ersatzwert in den Akkumulator 1 geladen wird. Die Programmbearbeitung wird dann mit diesem Ersatzwert fortgesetzt. Die für die Auswahl des Ersatzwerts notwendigen Informationen können Sie den lokalen Variablen des OB 122 entnehmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VAL	Input	DWORD	E, A, M, D, L	Ersatzwert
Funktionswert (RET_VAL)		INT	E, A, M, D, L	Fehlerinformation

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Die Anweisung wurde nicht von einem Synchronfehler-OB aus aufgerufen.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu SCL (Seite 7607)

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L	Nummer des zuletzt programmierten Schritts.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"DRUM_DB"(RESET := "Tag_Reset"
          JOG := "Tag_Input_Jog"
          DRUM_EN := "Tag_Input_DrumEN"
          LST_STEP := "Tag_Number_LastStep"
          EVENT1 := "MyTag_Event_1"
          EVENT2 := "MyTag_Event_2"
          EVENT3 := "MyTag_Event_3"
          EVENT4 := "MyTag_Event_4"
          EVENT5 := "MyTag_Event_5"
```

SCL

```
EVENT6 := "MyTag_Event_6"
EVENT7 := "MyTag_Event_7"
EVENT8 := "MyTag_Event_8"
EVENT9 := "MyTag_Event_9"
EVENT10 := "MyTag_Event_10"
EVENT11 := "MyTag_Event_11"
EVENT12 := "MyTag_Event_12"
EVENT13 := "MyTag_Event_13"
EVENT14 := "MyTag_Event_14"
EVENT15 := "MyTag_Event_15"
EVENT16 := "MyTag_Event_16"
OUT1 => "MyTag_Output_1"
OUT2 => "MyTag_Output_2"
OUT3 => "MyTag_Output_3"
OUT4 => "MyTag_Output_4"
OUT5 => "MyTag_Output_5"
OUT6 => "MyTag_Output_6"
OUT7 => "MyTag_Output_7"
OUT8 => "MyTag_Output_8"
OUT9 => "MyTag_Output_9"
OUT10 => "MyTag_Output_10"
OUT11 => "MyTag_Output_11"
OUT12 => "MyTag_Output_12"
OUT13 => "MyTag_Output_13"
OUT14 => "MyTag_Output_14"
OUT15 => "MyTag_Output_15"
OUT16 => "MyTag_Output_16"
Q => "Tag_Output_Q"
OUT_WORD => "Tag_OutputWord"
ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE

Parameter	Operand	Adresse	Wert
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE

Parameter	Adresse	Wert
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE

Parameter	Operand	Adresse	Wert
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung werden nach der Ausführung der Anweisung die folgenden Werte geändert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

DRUM_X: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" realisieren Sie ein Schrittschaltwerk mit maximal 16 Schritten. Die Nummer des ersten Schritts legen Sie am Parameter DSP fest. Die Nummer des letzten Schritts geben Sie am Parameter LST_STEP vor.

In jedem Schritt werden alle 16 Ausgangsbits OUT0 bis OUT15 und der Ausgangsparameter OUT_WORD (in dem die Ausgangsbits zusammengefasst sind) beschrieben. Ein Ausgangsbit bekommt entweder das zugehörige Bit des von Ihnen vorgegebenen Felds OUT_VAL oder das Ausgangsbit des vorangegangenen Schritts zugewiesen. Welcher Wert zugewiesen wird, hängt von der Belegung der Maskenbits am Parameter S_MASK ab.

Hinweis

Die Voreinstellung der Maskenbits ist 0. Falls Sie die aktuelle Einstellung eines oder mehrerer Maskenbits ändern wollen, führen Sie diese Änderung im Instanz-Datenbaustein durch.

Die Anweisung "Schrittschaltwerk realisieren" schaltet in den nächsten Schritt, wenn am Parameter JOG eine positive Signalflanke aufgetreten ist. Falls sich die Anweisung bereits im letzten Schritt befindet, werden bei einer positiven Flanke am Parameter JOG die Variablen Q und EOD gesetzt. Der Parameter DCC wird in diesem Fall auf den Wert 0 gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" gesetzt wird.

Darüber hinaus können Sie auch ein zeitabhängiges Weiterschalten in den nächsten Schritt zulassen. Dazu muss der Parameter DRUM_EN auf den Wert "1" gesetzt werden. Der Übergang in den nächsten Schritt erfolgt, wenn das zum aktuellen Schritt zugehörige Ereignisbit (EVENTi) gesetzt ist und die für den aktuellen Schritt vorgegebene Zeit abgelaufen ist. Die Zeit für einen Schritt ergibt sich aus dem Produkt von der Zeitbasis (DTBP) und dem für den aktuellen Schritt gültigen Zeitfaktor (S_PRESET).

Hinweis

Die im aktuellen Schritt noch verbleibende Bearbeitungszeit (DCC) wird nur dann reduziert, wenn das zugehörige Ereignisbit (EVENTi) gesetzt ist.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Wenn beim Aufruf der Anweisung der Wert "1" am Parameter RESET anliegt, geht das Schrittschaltwerk in den Schritt, dessen Nummer am Parameter DSP angegeben ist.

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit

und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist.

Zusätzlich können Sie das Schrittschaltwerk jederzeit über den Wert am Parameter JOG weiterschalten.

Beim Erstaufruf der Anweisung müssen Sie den Parameter RESET mit 1 belegen.

Wenn sich das Schaltwerk im letzten Schritt befindet und die für diesen Schritt vorgegebene Bearbeitungszeit abgelaufen ist, werden die Parameter Q und EOD gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" wechselt.

Die Anweisung wird nur in den Betriebszuständen ANLAUF und RUN ausgeführt.

Das Betriebssystem setzt die Anweisung "Schrittschaltwerk realisieren" bei Kaltstart zurück. Bei einem Neustart (Warmstart) wird die Anweisung nicht zurückgesetzt. Falls die Anweisung "Schrittschaltwerk realisieren" nach einem Neustart (Warmstart) initialisiert werden soll, müssen Sie die Anweisung im OB 100 mit dem Wert "1" am Parameter RESET aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" bewirkt das Rücksetzen des Schaltwerks. Beim Erstaufruf der Anweisung muss der Parameter RESET auf "1" gesetzt werden.
JOG	Input	BOOL	E, A, M, D, L	Eine positive Signalflanke (gegenüber dem letzten Aufruf) bewirkt das Weiterschalten in den nächsten Schritt, falls sich das Schaltwerk noch nicht im letzten Schritt befindet. Die Weiterschaltung erfolgt unabhängig davon, welchen Wert Sie am Parameter DRUM_EN angegeben haben.
DRUM_EN	Input	BOOL	E, A, M, D, L	Steuerparameter, der festlegt, ob ein zeitabhängiges Weiterschalten in den nächsten Schritt möglich ist. Bei einem Wert "1" ist ein zeitabhängiges Weiterschalten möglich.
LST_STEP	Input	BYTE	E, A, M, D, L	Nummer des letzten Schritts Zulässige Werte sind 1 bis 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit Nr. i (gehört zum Schritt i)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUTj, 0 ≤ j ≤ 15	Output	BOOL	E, A, M, D, L	Ausgangsbit Nr. j (identisch mit dem Bit Nr. j von OUT_WORD)
Q	Output	BOOL	E, A, M, D, L	Zustandsparameter, der angibt, ob die von Ihnen vorgegebene Bearbeitungszeit des letzten Schritts abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Ausgangsbits zusammengefasst in einer Variablen
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Nicht anwenderrelevant (Eingangsparameter JOG des vorhergehenden Aufrufs)
EOD	Static	BOOL	E, A, M, D, L	Identisch zum Ausgangsparameter Q
DSP	Static	BYTE	E, A, M, D, L, P	Nummer des ersten Schritts Zulässige Werte sind 1 bis 16.
DSC	Static	BYTE	E, A, M, D, L, P	Nummer des aktuellen Schritts
DCC	Static	DWORD	E, A, M, D, L, P	Die im aktuellen Schritt noch verbleibende Bearbeitungszeit in Millisekunden (ms) (nur relevant, falls DRUM_EN = 1 und das zugehörige Ereignisbit = 1)
DTBP	Static	WORD	E, A, M, D, L, P	Die für alle Schritte gültige Zeitbasis in Millisekunden (ms)
PREV_TIME	Static	DWORD	E, A, M, D, L	(Nicht anwenderrelevant: Systemzeit des vorhergehenden Aufrufs)
S_PRESET	Static	ARRAY of WORD	E, A, M, D, L	Eindimensionales Feld mit den Zeitfaktoren für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16]. In diesem Fall enthält S_PRESET [x] den Zeitfaktor des Schritts x.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den in jedem Schritt auszugebenden Werten, falls Sie diese nicht über S_MASK ausgeblendet haben. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält OUT_VAL [x, y] den Wert, der dem Ausgangsbit OUTy im Schritt x zugewiesen wird..
S_MASK	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den Maskenbits für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält S_MASK [x, y] das Maskenbit für den "y"-ten auszugebenden Wert im Schritt x. Bedeutung der Maskenbits: • 0: der entsprechende Wert des vorangegangenen Schritts wird dem zugehörigen Ausgangsbit zugewiesen • 1: der entsprechende Wert aus OUT_VAL wird dem zugehörigen Ausgangsbit zugewiesen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE* (W#16#...)	Erläuterung
0000	Kein Fehler
8081	Unzulässiger Wert am Parameter LST_STEP
8082	Unzulässiger Wert am DSC
8083	Unzulässiger Wert am DSP
8084	Das Produkt DCC = DTBP * S_PRESET[DSC] überschreitet den Wert 2**31-1 (ca. 24,86 Tage).
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT0 bis OUT15) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"DRUM_DB"(RESET := "Tag_Reset"
          JOG := "Tag_Input_Jog"
          DRUM_EN := "Tag_Input_DrumEN"
          LST_STEP := "Tag_Number_LastStep"
          EVENT1 := "MyTag_Event_1"
          EVENT2 := "MyTag_Event_2"
          EVENT3 := "MyTag_Event_3"
          EVENT4 := "MyTag_Event_4"
          EVENT5 := "MyTag_Event_5"
          EVENT6 := "MyTag_Event_6"
          EVENT7 := "MyTag_Event_7"
          EVENT8 := "MyTag_Event_8"
          EVENT9 := "MyTag_Event_9"
          EVENT10 := "MyTag_Event_10"
          EVENT11 := "MyTag_Event_11"
          EVENT12 := "MyTag_Event_12"
          EVENT13 := "MyTag_Event_13"
          EVENT14 := "MyTag_Event_14"
          EVENT15 := "MyTag_Event_15"
          EVENT16 := "MyTag_Event_16"
          OUT0 => "MyTag_Output_0"
          OUT1 => "MyTag_Output_1"
          OUT2 => "MyTag_Output_2"
          OUT3 => "MyTag_Output_3"
          OUT4 => "MyTag_Output_4"
          OUT5 => "MyTag_Output_5"
          OUT6 => "MyTag_Output_6"
          OUT7 => "MyTag_Output_7"
          OUT8 => "MyTag_Output_8"
          OUT9 => "MyTag_Output_9"
          OUT10 => "MyTag_Output_10"
          OUT11 => "MyTag_Output_11"
          OUT12 => "MyTag_Output_12"
          OUT13 => "MyTag_Output_13"
          OUT14 => "MyTag_Output_14"
```

SCL

```
OUT15 => "MyTag_Output_15"
Q => "Tag_Output_Q"
OUT_WORD => "Tag_OutputWord"
ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE

Parameter	Adresse	Wert
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	TRUE
OUT2	MyTag_Output_2	M4.2	TRUE
OUT3	MyTag_Output_3	M4.3	TRUE
OUT4	MyTag_Output_4	M4.4	TRUE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	TRUE
OUT7	MyTag_Output_7	M4.7	TRUE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	TRUE
OUT11	MyTag_Output_11	M5.3	TRUE
OUT12	MyTag_Output_12	M5.4	TRUE
OUT13	MyTag_Output_13	M5.5	TRUE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	FALSE
OUT2	MyTag_Output_2	M4.2	FALSE
OUT3	MyTag_Output_3	M4.3	FALSE
OUT4	MyTag_Output_4	M4.4	FALSE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	FALSE
OUT7	MyTag_Output_7	M4.7	FALSE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	FALSE
OUT11	MyTag_Output_11	M5.3	FALSE
OUT12	MyTag_Output_12	M5.4	FALSE
OUT13	MyTag_Output_13	M5.5	FALSE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	FALSE

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde(O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET, OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET, OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms.
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms.
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zu CMD

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"DCAT_DB"(CMD := "Tag_Input_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          Q => "Tag_Output_Q",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar

Eingangsparameter	Ausgangsparameter
< PT	ET < PT
>= PT	ET >= PT
Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
O_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Schließen"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"MCAT_DB"(O_CMD := "Tag_Iput_O_CMD",
          C_CMD := "Tag_Input_C_CMD",
          S_CMD := "Tag_Input_S_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          OO => "Tag_OutputOpen",
          CO => "Tag_OutputClosed",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA",
          Q => "Tag_Output_Q");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF BOOL	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Syntax

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY OF WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird. Am Parameter IN können auch Konstanten angegeben werden.
SAMPLE_T	Input	INT	E, A, M, D, L, P	Abtastzeit Am Parameter SAMPLE_T können auch Konstanten angegeben werden.
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P	Voreilzeit in der gleichen Einheit wie die Abtastzeit.
LG_TIME	Static	REAL	E, A, M, D, L, P	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P	Vorheriger Ausgang

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

SCL

```
"LEAD_LAG_DB"(IN := "Tag_Input",
              SAMPLE_T := "Tag_Input_SAMPLE_T",
              OUT => "Tag_Output_Result",
              ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

TONR_X: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines festgelegten Zeitraums, den Sie mithilfe des Eingangs PV angeben. Die Anweisung wird ausgeführt, wenn der Signalzustand am Starteingang TMR_EN von "0" nach "1" wechselt. Gleichzeitig läuft die Zeit mit der am Eingang PV programmierten Zeitdauer ab. Während die Anweisung ausgeführt wird, addiert sie den Zeitwert des Eingangs DELTA_T zum Zeitwert des InOut ET.

Folgende Bedingungen müssen dazu erfüllt sein:

- Der Starteingang TMR_EN liefert den Signalzustand "1".
- Der Zeitwert am InOut ET ist kleiner als der Zeitwert am Parameter PV.
- Der Eingang RESET liefert den Signalzustand "0".

Die Anweisung wird unterbrochen, wenn der Signalzustand am Starteingang TMR_EN von "1" nach "0" wechselt. Sobald der Signalzustand wieder nach "1" wechselt, läuft die Anweisung weiter.

Die Anweisung wird beendet, wenn der Zeitwert am InOut ET den Zeitwert am Eingang PV erreicht hat. Dann liefert der Ausgang Q den Signalzustand "1".

Solange der Ausgang Q den Signalzustand "1" liefert, bleibt der Zeitwert am InOut ET unverändert. Der Zeitwert und der Ausgang Q werden erst wieder auf "0" zurückgesetzt, wenn der Eingang RESET auf den Signalzustand "1" gesetzt wird.

Da die Anweisung "Zeit akkumulieren" die Zykluszeit (DELTA_T) des letzten Zyklus des Organisationsbausteins (OB) verwendet, darf sie nur in zyklischen Organisationsbausteinen aufgerufen werden.

Hinweis

Sie müssen die Zykluszeit des Organisationsbausteins aus den Anlaufinformationen in die globale Variable des Parameters DELTA_T verschieben.

Syntax

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TMR_EN (Timer Enable)	Input	BOOL	E, A, M, D, L	Starteingang Startet die Anweisung und den Ablauf der voreingestellten Zeitdauer am Eingang PV
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
PV (Preset Value)	Input	DINT	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung
DELTA_T	Input	INT	E, A, M, D, L oder Konstante	Zykluszeit des Organisationsbausteins des vorherigen Zyklus
Q	Output	BOOL	E, A, M, D, L	Ausgang, der bei Erreichen des voreingestellten Zeitwerts gesetzt wird
ET (Elapsed Time)	InOut	DINT	E, A, M, D, L	Akkumulierter Zeitwert (Aktuell abgelaufener Zeitwert + Zeitwert des Eingangs DELTA_T)
Funktionswert		VOID		Leerer Funktionswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeitwert akkumulieren":

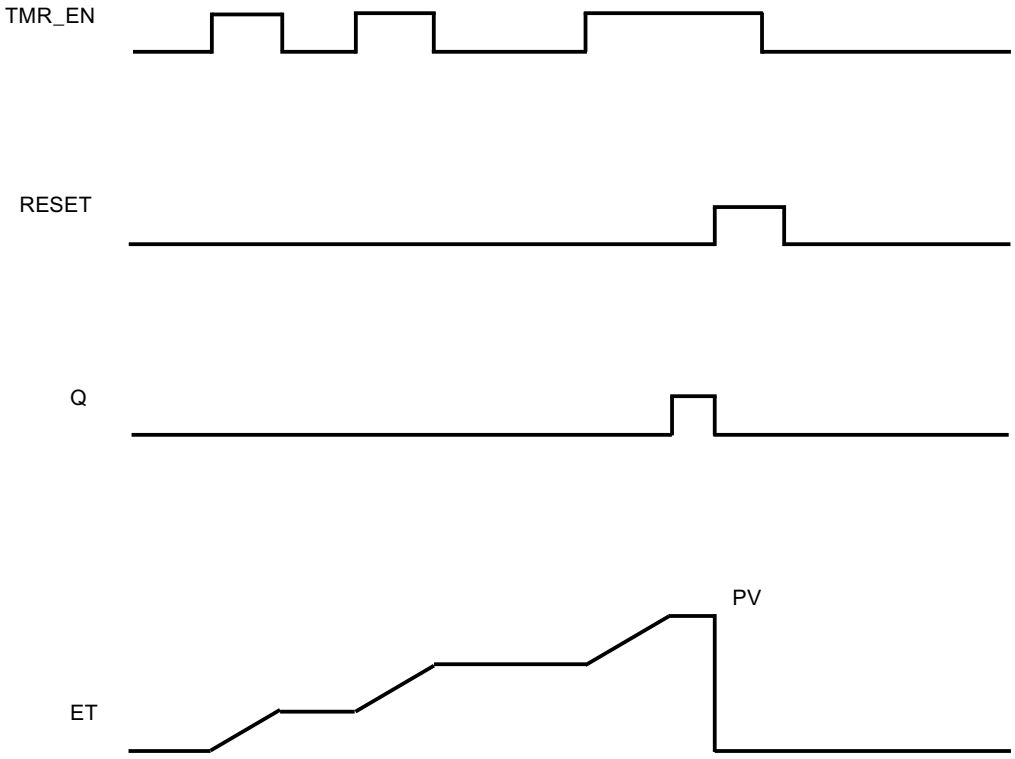

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := TONR_X(TMR_EN := "Tag_Start",
                       RESET := "Tag_Reset",
                       PV := 10,
                       DELTA_T := 2,
                       Q => "Tag_Output",
                       ET := "Tag_Elapsed_Time");
```

Bei jedem Wechsel des Signalzustands von "0" nach "1" am Eingang TMR_EN, wird die Zykluszeit "2" am Eingang DELTA_T zum aktuellen Zeitwert des InOut ET addiert, bis der maximale Zeitwert "10" des Eingangs PV erreicht ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

WSR: Datum im Schieberegister speichern

Beschreibung

Mit der Anweisung "Datum im Schieberegister speichern" schreiben Sie Daten aus einer angegebenen Quelle in ein Schieberegister. Dabei wird jedes Element des Registers in die nächste Adresse verschoben. Die Daten, die in der letzten Adresse des Schieberegisters enthalten sind, gehen nach der Schiebeoperation verloren.

Bei jeder Ausführung der Anweisung mit dem Wert "0" am Parameter RESET, werden neue Daten aus der Quelle (S_DATA) gelesen und in die Anfangsadresse (START) des Schieberegisters geschoben. Der Parameter LEN gibt die Anzahl der zu schiebenden Elemente an. Am Parameter E_TYPE wird der Datentyp der Elemente festgelegt. Wenn am Parameter E_TYPE ein unzulässiger Datentyp angegeben wird, wird die Anweisung nicht bearbeitet.

Wenn der Parameter RESET bei der Ausführung der Anweisung auf "1" gesetzt ist, wird das Register gelöscht. Der Parameter Q wird auf den Signalzustand "1" gesetzt, wenn das Schieberegister mit Nullen belegt ist oder gelöscht wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_DATA	Input	POINTER	E, A, M, D	Zeigt auf das Quellelement, das in die Anfangsadresse (START) geschoben wird. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
START	Input	POINTER	E, A, M, D	Zeigt auf die Anfangsadresse des Registers. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
LEN	Input	WORD	E, A, M, D, L, P	Anzahl der Elemente, die geschoben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der zu schiebenden Elemente an. Folgende Datentypen sind zulässig: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	E, A, M, D, L	Gibt "0" an, wenn der Parameter RESET aktiv (1) ist oder alle Elemente, die geschoben werden, den Wert "0" haben.
Funktionswert		VOID	E, A, M, D, L	Leerer Funktionswert

Beispiel

Im folgenden Beispiel werden vier Elemente des Datentyps WORD ab der am Parameter START angegebenen Adresse geschoben. Nachdem das erste Element des Registers in die nächste Adresse geschoben wurde, wird die erste Adresse mit den Daten belegt, auf die der Zeiger am Parameter S_DATA zeigt. Der Wert des letzten Elements im Register geht dabei verloren.

SCL

```
WSR(RESET := "Tag_Input_Reset",
    S_DATA := "Tag_Input_DataSource",
    START := "Tag_Input_StartAddress",
    LEN := "Tag_Input_Length",
    E_TYPE := "Tag_Input_ElementType",
    Q => "Tag_Output_Q");
```

Es wird kein Funktionswert übergeben.

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	DB1.DBX0.0 DW6 = W#16#1234

Parameter	Operand	Wert
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	W#16#04
Q	Tag_Output_Q	1

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SHRB: Bit in Bitschieberegister schieben

Beschreibung

Mit der Anweisung "Bit in Bitschieberegister schieben" schreiben Sie ein Bit aus einer angegebenen Quelle (DATA) in ein Schieberegister. Bei jeder Ausführung der Anweisung mit einem Wert "0" am Parameter RESET werden neue Daten aus der Quelle gelesen und in die Anfangsadresse (S_BIT) des Schieberegisters geschoben. Alle folgenden Bits werden um eins verschoben. Das Bit in der letzten Adresse (S_BIT + N) geht nach dem Schieben verloren. Wenn der Parameter RESET während der Bearbeitung der Anweisung auf "1" gesetzt ist, werden die Adressen im Schieberegister auf "0" zurückgesetzt und nicht geschoben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	Input	BOOL	E, A, M, D, L	Quellbit
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das Anfangsbit des Schieberegisters. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	WORD	E, A, M, D, L, P	Anzahl der Bits, die geschoben werden.
Funktionswert		VOID	E, A, M, D, L	Leerer Funktionswert

Beispiel

Im folgenden Beispiel wird der Parameter N auf den Wert 14 ("E" in hexadezimaler Schreibweise) gesetzt, sodass 14 Bits ab der am Parameter S_BIT angegebenen Adresse geschoben werden. Nachdem die Bits geschoben worden sind, wird die erste Adresse mit den Daten aus der Quelle am Parameter DATA belegt. Der letzte Bitwert geht verloren.

SCL

```
SHRB(DATA := "Tag_Input_Data",
     RESET := "Tag_Input_Reset",
     S_BIT := "Tag_Input_BitSource",
     N := "Tag_Input_Number");
```

Es wird kein Funktionswert übergeben.

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Vor der Bearbeitung der Anweisung sind die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Nach der Bearbeitung

Nach der Bearbeitung der Anweisung werden die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	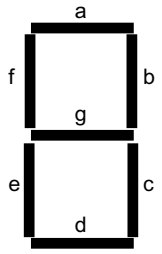
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige
Funktionswert		VOID	E, A, M, D, L	Leerer Funktionswert

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
SEG(IN := "Tag_Input",
    OUT => "Tag_Output");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW16#065B4F66	00000110 01011011 01001111 01100110 Anzeige: 1234

Es wird kein Funktionswert übergeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Operanden angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	DWORD	E, A, M, D, L, P	7-stellige BCD-Zahl
Funktionswert		DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := BCDCPL("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Operand	Wert*
Tag_Input	DW#16#01234567
Tag_Result	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu SCL (Seite 7607)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	DWORD	E, A, M, D, L, P	Operand, dessen gesetzte Bit gezählt werden
Funktionswert		INT	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

SCL

```
"Tag_Result" := BITSUM("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Operand	Wert*
Tag_Input	DW#16#12345678
Tag_Result	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu SCL (Seite 7607)

4.2.2.5 GRAPH

GRAPH-Kette

Schritt und Transition

Beschreibung

Mit dem Strukturelement "Schritt und Transition" können Sie gleichzeitig einen Schritt und eine Transition in Ihre Ablaufkette einfügen.

Siehe auch:

Schritt (Seite 5267)

Transition (Seite 5268)

Siehe auch

Schritt (Seite 5267)

Transition (Seite 5268)

Kettenende (Seite 5269)

Sprung zu Schritt (Seite 5269)

Alternativzweig (Seite 5270)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Schritte und Transitionen (Seite 7692)

Schritt

Beschreibung

Schritte dienen dazu, komplexe Automatisierungsaufgaben in übersichtliche Teilaufgaben zu untergliedern, die anschließend mithilfe von Aktionen gelöst werden können. Die einzelnen Schritte werden in Ablaufketten organisiert, sodass bei Programmablauf jeder Schritt in der festgelegten Reihenfolge abgearbeitet werden kann. Jeder Schritt muss einen eindeutigen Namen und eine eindeutige Nummer erhalten.

Damit die Bearbeitung tatsächlich erfolgt, muss der Schritt durch eine der folgenden Bedingungen aktiv werden:

- Der Schritt wurde als Initialschritt definiert.
- Die Transition des Vorgängerschritts ist erfüllt.
- Der Schritt wird durch eine ereignisabhängige Aktion aufgerufen.

Nachdem alle Aktionen bearbeitet wurden, wird der Schritt wieder inaktiv.

Schritte, in denen keine Aktionen programmiert sind, werden Leerschritte genannt. Ein solcher Leerschritt verhält sich wie ein aktiver Schritt und die nachfolgende Transition wird immer erfüllt.

Siehe auch

Grundlagen zu Schritten (Seite 7692)

Schritt und Transition (Seite 5267)

Transition (Seite 5268)

Kettenende (Seite 5269)

Sprung zu Schritt (Seite 5269)

Alternativzweig (Seite 5270)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Transition

Beschreibung

Transitionen stehen zwischen den Schritten und enthalten die Bedingungen zum Weiterschalten von einem Schritt zum nächsten. Wenn die Weiterschaltbedingungen einer Transition erfüllt sind, wird der nächste Schritt aktiv und seine Aktionen werden bearbeitet. Sie können die Bedingungen einer Transition entweder in KOP der FUP programmieren.

Siehe auch

Grundlagen zu Transitionen (Seite 7694)

Schritt und Transition (Seite 5267)

Schritt (Seite 5267)

Kettenende (Seite 5269)

Sprung zu Schritt (Seite 5269)

Alternativzweig (Seite 5270)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Kettenende

Beschreibung

Mit dem Strukturelement "Kettenende" können Sie eine Ablaufkette oder eine Verzweigung beenden. In einer Simultanverzweigung muss dem Kettenende jedoch eine Transition vorangehen.

Hinweis

Werden alle Zweige einer Ablaufkette mit einem Kettenende abgeschlossen, kann die Ablaufkette entweder durch den Parameter "INIT_SQ" oder in der Task Card "Testen" in der Palette "Kettensteuerung" mit der Schaltfläche "Initialisieren" neu gestartet werden.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 5267)

Schritt (Seite 5267)

Transition (Seite 5268)

Sprung zu Schritt (Seite 5269)

Alternativzweig (Seite 5270)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Sprung zu Schritt

Beschreibung

Mit einem Sprung können Sie die Programmausführung mit einem beliebigem Schritt innerhalb des GRAPH-Funktionsbausteins fortsetzen. Sprünge können am Ende des Hauptzweiges oder eines Alternativzweiges eingefügt werden, sodass eine zyklische Bearbeitung der Ablaufkette möglich ist. Sprung und Sprungziel werden in der Ablaufkette als Pfeile dargestellt, wobei beim Sprungziel die Absprungtransition angegeben ist.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 5267)

Schritt (Seite 5267)

Transition (Seite 5268)

Kettenende (Seite 5269)

Alternativzweig (Seite 5270)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Alternativzweig

Beschreibung

Mithilfe von Alternativzweigen können Sie ODER-Verzweigungen programmieren. Das bedeutet, dass Sie nach einem Schritt Zweige einfügen, die mit einer Transition beginnen. Abhängig davon, welche Transition zuerst erfüllt ist, wird der entsprechende Zweig abgearbeitet. Sind mehrere Transitionen gleichzeitig erfüllt, hat jeweils die am weitesten links stehende Transition die höchste Priorität und der entsprechende Zweig wird abgearbeitet. Alternativzweige enden wieder mit einer Transition.

In einer Ablaufkette können Sie bis zu 125 Alternativzweige programmieren.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Schritt und Transition (Seite 5267)

Schritt (Seite 5267)

Transition (Seite 5268)

Kettenende (Seite 5269)

Sprung zu Schritt (Seite 5269)

Simultanzweig (Seite 5270)

Verzweigung schließen (Seite 5271)

Simultanzweig

Beschreibung

Mithilfe von Simultanzweigen können Sie UND-Verzweigungen programmieren. Das bedeutet, dass Sie mit einer Transition mehrere Schritte aktiv setzen, deren Aktionen dann ausgeführt werden. Darum beginnen und enden Simultanzweige immer mit einem Schritt.

Die nachfolgenden Transitionen der Simultanzweige befinden sich auf dem Hauptzweig, wobei die verschiedenen Simultanzweige an verschiedenen Stellen des Hauptzweigs geschlossen werden können. Beachten Sie, dass Zweige, die Sie in einer Transition zusammenführen, erst dann in den nächsten Schritt schalten, wenn diese Zweige komplett abgearbeitet sind.

In einer Ablaufkette können Sie bis zu 249 Simultanzweige programmieren.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)
Schritt und Transition (Seite 5267)
Schritt (Seite 5267)
Transition (Seite 5268)
Kettenende (Seite 5269)
Sprung zu Schritt (Seite 5269)
Alternativzweig (Seite 5270)
Verzweigung schließen (Seite 5271)

Verzweigung schließen

Beschreibung

Mit dem Element "Verzweigung schließen" können Sie Simultan- oder Alternativzweige zu ihrem Elternzweig hin schließen. Dies ist notwendig, wenn Sie die Verzweigung nicht mit einem Sprung oder einem Kettenende abschließen möchten. In einem Simultanzweig können Sie "Verzweigung schließen" nur nach einem Schritt einfügen.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)
Schritt und Transition (Seite 5267)
Schritt (Seite 5267)
Transition (Seite 5268)
Kettenende (Seite 5269)
Sprung zu Schritt (Seite 5269)
Alternativzweig (Seite 5270)
Simultanzweig (Seite 5270)

GRAPH-Aktionen

Zeiten

TP: Impuls erzeugen

Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Parameter ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Parameter IN "0" ist, wird der Parameter ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TP (z. B. "TP_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Impuls erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Impuls erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Parameter ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer des Impulses Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Impulsausgang
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":

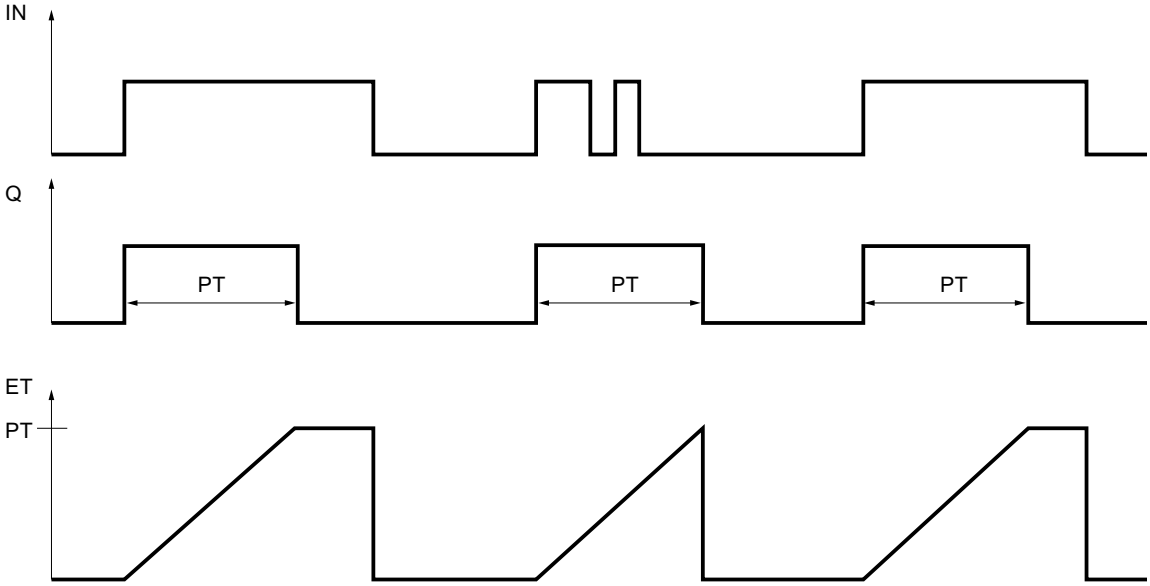

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TP TIME, "IEC_TP_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTIME" gespeichert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

TON: Einschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang IN noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TON (z. B. "TON_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Einschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Einschaltverzögerung

erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Einschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":

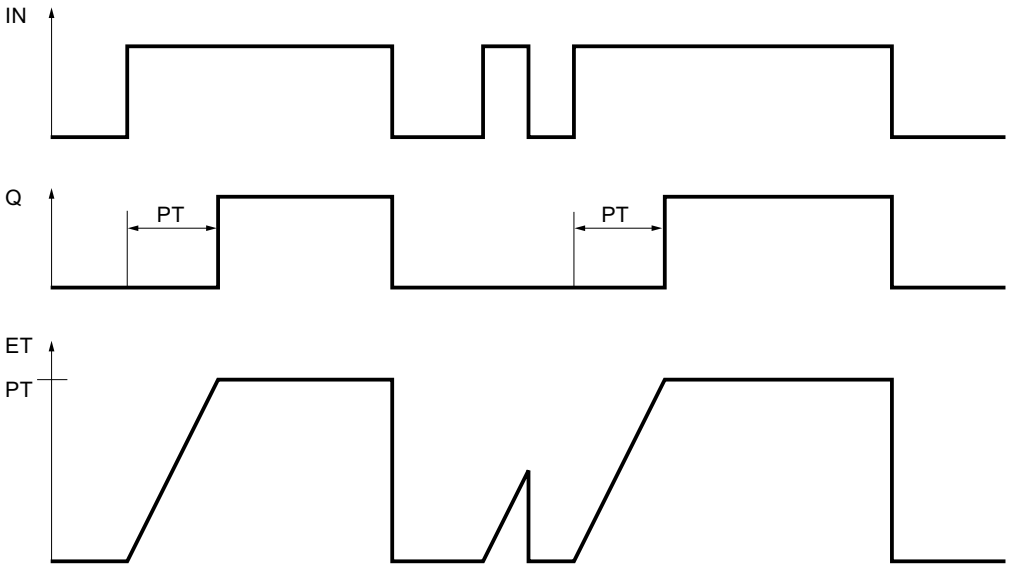

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TON TIME, "IEC_TON_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTime" abgelegt. Wenn der Signalzustand am Operanden "Tag_Start" von "1" auf "0" wechselt, wird der Operand "Tag_Status" zurückgesetzt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

TOF: Ausschaltverzögerung erzeugen

Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Die Ausführung der Anweisung setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden. Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden. Eine IEC-Zeit können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ TOF (z. B. "TOF_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Ausschaltverzögerung erzeugen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "0" am Parameter PT aufrufen. Falls Instanzen der Anweisung "Ausschaltverzögerung erzeugen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Hinweis

Überspringen der Anweisung

Wenn die Anweisung im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

Hinweis

Aktualisierung der Anweisungsdaten

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung. Bei einem Zugriff auf die Ausgänge Q oder ET werden die Daten nicht aktualisiert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	BOOL	E, A, M, D, L	Starteingang
PT	Input	TIME	E, A, M, D, L, P oder Konstante	Zeitdauer der Ausschaltverzögerung Der Wert am Parameter PT muss positiv sein.
Q	Output	BOOL	E, A, M, D, L	Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.
ET	Output	TIME	E, A, M, D, L	Aktueller Zeitwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":

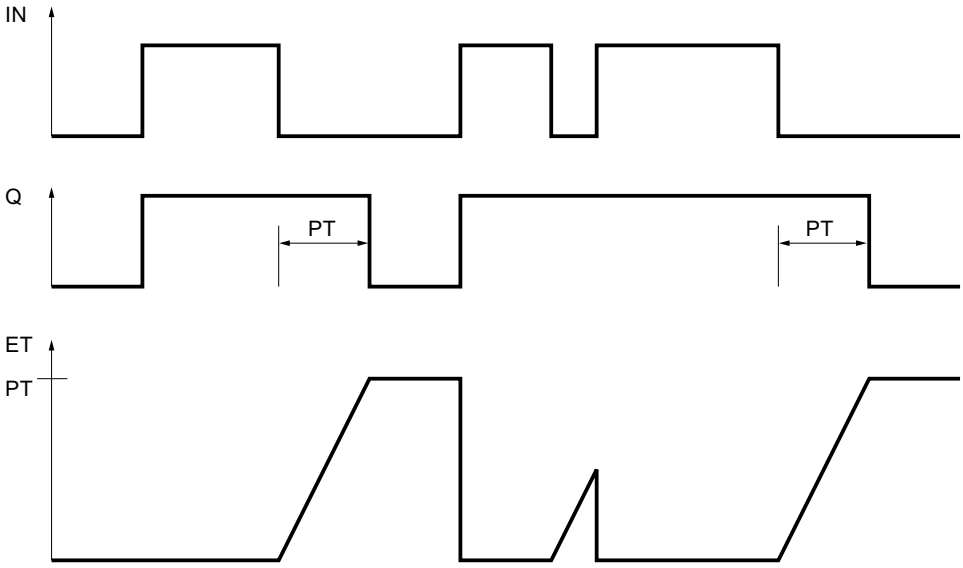

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TOF TIME, "IEC_TOF_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
```

GRAPH

```
ET => "Tag_ElapsedTIME"
)
```

Bei einem Wechsel im Signalzustand des Operanden "Tag_Start" von "0" auf "1" wird der Operand "Tag_Status" gesetzt. Wenn der Signalzustand des Operanden "Tag_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag_Status" gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag_Status" zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag_ElapsedTIME" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

Zähler

CTU: Vorwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert des Ausgangs CV wird auf Null zurückgesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTU (z. B. "CTU_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit Wert "1" am Parameter R der Anweisung aufrufen. Falls Instanzen der Anweisung "Vorwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Zähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang Q gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTU INT,"IEC_CTU_DB"
      (CU := "Tag_Start"
       R := "Tag_ResetCOUNTER"
       PV := "Tag_PresetValue"
       Q => "Tag_Status"
       CV => "Tag_CounterValue"
       )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag_CounterValue" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "Tag_Status" übernommen. Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle

Zählwert größer oder gleich dem Wert des Operanden "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

CTD: Rückwärts zählen

Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Eingang LD auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTD (z. B. "CTD_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen der Anweisung nach einem Neustart (Warmstart) initialisiert sein sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit dem Wert "1" am Parameter LD der Anweisung aufrufen. Am Parameter PV wird in diesem Fall der gewünschte Anfangswert für den Parameter CV angegeben.

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CD	Input	BOOL	E, A, M, D, L	Zähleingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
Q	Output	BOOL	E, A, M, D, L	Zählerstatus
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTD INT, "IEC_CTD_DB"
      (CD := "Tag_Start"
       LD := "Tag_LoadPV"
       PV := "Tag_PresetValue"
       Q => "Tag_Status"
       CV => "Tag_CounterValue"
       )
```

Wenn der Signalzustand des Operanden "Tag_Start" von "0" auf "1" wechselt, wird die Anweisung ausgeführt und der Wert am Ausgang "Tag_CounterValue" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Ausgang "Tag_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_Status" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

CTUD: Vorwärts und rückwärts zählen

Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Wert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Ausgang CV um eins erhöht. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hochgezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0".

Am Ausgang QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Einen IEC-Zähler können Sie wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Typ CTUD (z. B. "CTUD_DB")

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Das Betriebssystem setzt die Instanzen der Anweisung "Vorwärts und rückwärts zählen" bei einem Kaltstart zurück. Falls Instanzen nach einem Neustart (Warmstart) initialisiert sein

sollen, müssen Sie in einem Anlauf-OB die zu initialisierenden Instanzen mit den folgenden Parameterwerten aufrufen:

- Bei Verwendung als Vorwärtszähler muss der Wert am Parameter R auf "1" gesetzt werden.
- Bei Verwendung als Rückwärtszähler muss der Wert am Parameter LD auf "1" gesetzt werden. Am Parameter PV geben Sie in diesem Fall den gewünschten Anfangswert für den Parameter CV an.

Falls Instanzen der Anweisung "Vorwärts und rückwärts zählen" innerhalb eines anderen Bausteins enthalten sind, erreichen Sie das Rücksetzen dieser Instanzen z. B. durch Initialisierung des übergeordneten Bausteins.

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CU	Input	BOOL	E, A, M, D, L	Vorwärtszähleingang
CD	Input	BOOL	E, A, M, D, L	Rückwärtszähleingang
R	Input	BOOL	E, A, M, D, L, T, Z, P	Rücksetzeingang
LD	Input	BOOL	E, A, M, D, L, P	Ladeeingang
PV	Input	INT	E, A, M, D, L, P oder Konstante	Wert, bei dem der Ausgang QU gesetzt wird. / Wert, auf den der Ausgang CV bei LD = 1 gesetzt wird.
QU	Output	BOOL	E, A, M, D, L	Status der Vorwärtszähler
QD	Output	BOOL	E, A, M, D, L	Status der Rückwärtszähler
CV	Output	INT	E, A, M, D, L, P	Aktueller Zählwert

Aus der Klappliste "???" können Sie den Datentyp der Anweisung auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL CTUD INT, "IEC_CTUD_DB"
      (CU := "Tag_StartCTU"
       CD := "Tag_StartCTD"
       R := "Tag_ResetCOUNTER"
       LD := "Tag_LoadPV"
       PV := "Tag_PresetValue"
       QU => "Tag_CounterStatusUP"
       QD => "Tag_CounterStatusDOWN"
```

GRAPH

```
CV => "Tag_CounterValue"
)
```

Wenn der Signalzustand am Eingang "Tag_StartCTU" oder am Eingang "Tag_StartCTD" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "Tag_StartCTU" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag_CounterValue" abgelegt. Wenn eine positive Signalflanke am Eingang "Tag_StartCTD" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag_CounterValue" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "Tag_CounterStatusUP" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag_CounterStatusUP" den Signalzustand "0".

Der Ausgang "Tag_CounterStatusDOWN" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag_CounterStatusDOWN" den Signalzustand "0".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

Mathematische Funktionen

NEG: Zweierkomplement erzeugen

Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Operandenwerts. Wenn z. B. ein positiver Wert ansteht, wird das negative Äquivalent dieses Werts ausgegeben.

Der Wert des Ausgangs OUT ist ungültig, wenn das Ergebnis der Anweisung außerhalb des Bereichs liegt, der für den am Ausgang OUT angegebenen Datentyp zulässig ist (gilt nur für Ganzzahlen).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweierkomplement des Eingangswerts

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := NEG_REAL("Tag_InValue")
```

Die Anweisung wechselt das Vorzeichen des Werts am Operanden "TagIn_Value" und gibt das Ergebnis am Operanden "TagOut_Value" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ABS: Absolutwert bilden

Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Operandenwerts.

Der Wert des Ergebnis ist ungültig, wenn eine Gleitpunktzahl einen ungültigen Wert aufweist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Absolutwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ABS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
TagIn_Value	-6,234
TagOut_Value	6,234

Die Anweisung berechnet den Absolutbetrag des Werts am Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

MIN: Minimum ermitteln

Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den kleinsten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MIN INT
      (IN1 := "TagIn_Value1"
       IN2 := "TagIn_Value2"
       IN3 := "TagIn_Value3"
       OUT => "Tag_Minimum"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn_Value1") in den Ausgang "Tag_Minimum".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

MAX: Maximum ermitteln

Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der Eingänge IN1, IN2 und IN3 und schreiben den größten Wert in den Ausgang OUT. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Erster Eingangswert
IN2	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Zweiter Eingangswert
IN3	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Dritter Eingangswert
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL MAX INT
      (IN1 := "TagIn_Value1"
       IN2 := "TagIn_Value2"
       IN3 := "TagIn_Value3"
       OUT => "Tag_Maximum"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
IN1	TagIn_Value1	12666
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn_Value2") in den Ausgang "Tag_Maximum".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

LIMIT: Limitieren

Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung MN <= IN <= MX erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert. Die Bearbeitung der Anweisung setzt voraus, dass die Variablen an allen Eingängen vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

Hinweis

An den Eingangsparametern sind auch bei deaktivierter IEC-Prüfung ausschließlich die Datentypen INT, DINT und REAL zulässig. Wenn Sie die Datentypen WORD oder DWORD an den Eingangsparametern verwenden, liefert die Anweisung ein undefiniertes Ergebnis.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Untergrenze
IN	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Eingangswert
MX	Input	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Obergrenze
OUT	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL LIMIT INT
      (MN := "Tag_LowLimit"
       IN := "Tag_InputValue"
       MX := "Tag_HighLimit"
       OUT => "Tag_Result"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

Der Wert des Operanden "Tag_InputValue" wird mit den Werten der Operanden "Tag_LowLimit" und "Tag_HighLimit" verglichen. Da der Wert des Operanden "Tag_InputValue" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag_LowLimit" in den Ausgang "Tag_Result" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SQR: Quadrat bilden

Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den Wert der Gleitpunktzahl des Operanden und schreiben das Ergebnis in den Ausgang.

Wenn der Operand eine ungültige Gleitpunktzahl enthält, dann ist der Wert des Ergebnises ungültig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Quadrat des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SQR("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	5.0
Tag_OutValue	25.0

Die Anweisung quadriert den Wert des Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SQRT: Quadratwurzel bilden

Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem Eingangswert die Quadratwurzel und speichern das Ergebnis im angegebenen Operanden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück. Wenn der Wert des Operanden "0" ist, ist das Ergebnis auch "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L	Quadratwurzel des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SQRT("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	25.0
Tag_OutValue	5.0

Die Anweisung zieht aus dem Wert des Operanden "Tag_InValue" die Quadratwurzel und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

LN: Natürlichen Logarithmus bilden

Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem Eingangswert den natürlichen Logarithmus zur Basis e (e = 2,718282). Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Natürlicher Logarithmus des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := LN("Tag_InValue")
```

Die Anweisung bildet den natürlichen Logarithmus des Werts am Operanden "Tag_InValue" und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

EXP: Exponentialwert bilden

Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2,718282) und dem Wert des Operanden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Exponentialwert des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := EXP("Tag_InValue")
```

Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag_InValue" und gibt das Ergebnis (e^{IN}) am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SIN: Sinuswert bilden

Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	REAL	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	REAL	E, A, M, D, L, P	Sinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SIN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+1.570796 (π/2)
Tag_OutValue	1.0

Die Anweisung berechnet den Sinus des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

COS: Cosinuswert bilden

Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	REAL	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	REAL	E, A, M, D, L, P	Cosinus des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := COS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+1.570796 (π/2)
Tag_OutValue	0

Die Anweisung berechnet den Cosinus des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

TAN: Tangenswert bilden

Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Operanden im Bogenmaß angegeben.

Der Wert des Ergebnis ist ungültig, wenn der Wert des Operanden keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	REAL	E, A, M, D, L, P oder Konstante	Größe des Winkels im Bogenmaß
<Ergebnis>	Output	REAL	E, A, M, D, L, P	Tangens des angegebenen Winkels

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := TAN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	+3.141593 (π)
Tag_OutValue	0

Die Anweisung berechnet den Tangens des am Operanden "Tag_InValue" angegebenen Winkels und gibt das Ergebnis am Operanden "Tag_OutValue" aus.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ASIN: Arcussinuswert bilden

Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Operanden angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Operanden ist keine gültige Gleitpunktzahl.
- Der Wert des Operanden liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	REAL	E, A, M, D, L, P oder Konstante	Sinuswert
<Ergebnis>	Output	REAL	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ASIN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1.0
Tag_OutValue	+1.570796 (π/2)

Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ACOS: Arcuscosinuswert bilden

Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Operanden angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße kann in einem Wertebereich von 0 bis +π liegen.

Der Wert des Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Operanden ist keine gültige Gleitpunktzahl.
- Der Wert des Operanden liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	REAL	E, A, M, D, L, P oder Konstante	Cosinuswert
<Ergebnis>	Output	REAL	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ACOS("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0
Tag_OutValue	+1.570796 (π/2)

Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ATAN: Arcustangenswert bilden

Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Operanden angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Operanden dürfen nur gültige Gleitpunktzahlen (oder -NaN/+NaN) angegeben werden. Die berechnete Winkelgröße kann in einem Wertebereich von -π/2 bis +π/2 liegen.

Der Wert am Ausgang OUT ist ungültig, wenn der Wert am Eingang IN keine gültige Gleitpunktzahl ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Tangenswert
<Ergebnis>	Output	Gleitpunktzahlen	E, A, M, D, L, P	Größe des Winkels im Bogenmaß

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ATAN("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1.0
Tag_OutValue	+0.785398 (π/4)

Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Operanden "Tag_InValue" entspricht. Das Ergebnis der Anweisung wird am Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Ungültige Gleitpunktzahlen (Seite 266)

Grundlagen zu GRAPH (Seite 7685)

Verschieben

SWAP: Anordnung ändern

Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes des Operanden.

Abhängig vom verwendeten Datentyp, können Sie Bytes im gesamten Akkumulator 1 oder nur im rechten Wort des Akkumulators 1 tauschen.

Das folgende Bild zeigt, wie die Bytes des Operanden vom Datentyp DWORD vertauscht werden:

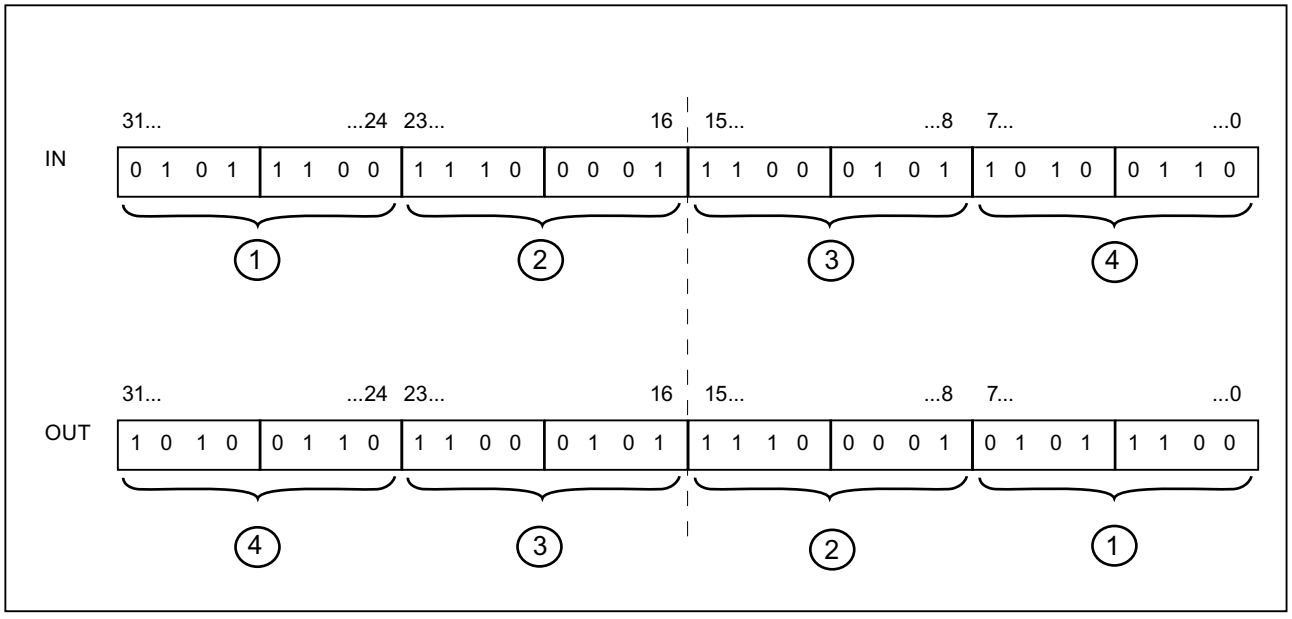

Anordnung von Bytes im rechten Wort des Akkumulators 1

Beim Datentyp WORD ändern Sie die Anordnung der Bytes im rechten Wort des Akkumulators 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert A	Wert B	Wert D	Wert C

Das Ergebnis der Anweisung wird im rechten Wort des Akkumulators 1 gespeichert. Die Bytes im linken Wort des Akkumulators 1 werden durch die Anweisung nicht beeinflusst und bleiben unverändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	WORD, DWORD	E, A, M, D, L oder Konstante	Operand, dessen Bytes vertauscht werden
<Ergebnis>	Output	WORD, DWORD	E, A, M, D, L	Ergebnis

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SWAP("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert			
Tag_InValue	0000	1111	0000	1111
Tag_OutValue	0000	1111	1111	0000

Anordnung von Bytes im gesamten Akkumulator 1 ändern:

Bei Datentyp DWORD ändern Sie die Anordnung der Bytes im gesamten Akkumulator 1.

Die folgende Tabelle zeigt den Inhalt des Akkumulators 1 vor und nach der Ausführung der Anweisung:

Status	Bytes im Akkumulator 1			
Vor der Ausführung	Wert A	Wert B	Wert C	Wert D
Nach der Ausführung	Wert D	Wert C	Wert B	Wert A

Das Ergebnis der Anweisung wird im Akkumulator 1 gespeichert. Der Inhalt des Akkumulators 2 bleibt unverändert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert							
Tag_InValue	1111	0000	0000	1111	0000	0000	1111	1111
Tag_OutValue	1111	1111	0000	0000	0000	1111	1111	0000

Die Anordnung der Bytes im Operanden "Tag_InValue" wird vertauscht im Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

BLKMOV: Bereich kopieren

Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

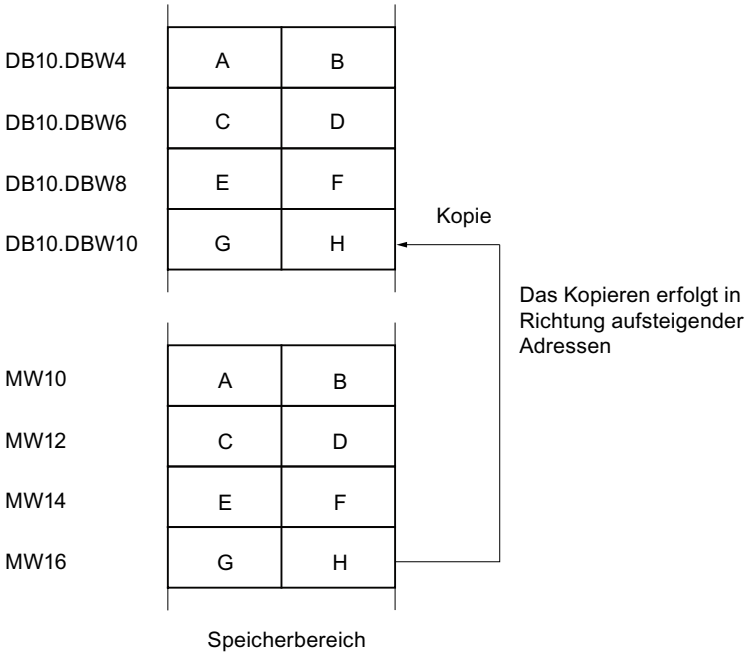

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

Unterbrechbarkeit

Solange der Quellbereich nicht Teil eines Datenbausteins ist, der nur im Ladespeicher vorhanden ist, gibt es keine Begrenzung der Schachtelungstiefe.

Bei Unterbrechung einer BLKMOV-Bearbeitung hingegen, bei der aus einem nicht ablaufrelevanten DB kopiert wird, kann eine solche BLKMOV-Bearbeitung nicht mehr eingeschachtelt werden.

Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Nicht ablaufrelevante Datenbausteine

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

Regeln beim Kopieren von nicht ablaufrelevanten Datenbausteinen

Der Quellbereich kann auch in einem nicht ablaufrelevanten Datenbaustein im Ladespeicher liegen. Datenbausteine, die nicht ablaufrelevant sind, werden mit dem Schlüsselwort UNLINKED gekennzeichnet.

Wenn ein nicht ablaufrelevanter Datenbaustein mit der Anweisung "Bereich kopieren" in den Arbeitsspeicher kopiert und gleichzeitig z. B. durch das Programmiergerät geladen wird, kann die Bearbeitung der Anweisung um mehrere Millisekunden verzögert werden. Dies führt zu einer Verlängerung des OB-Zyklus und kann zum Ansprechen der Zykluszeitüberwachung führen.

Wenn mit der Anweisung "Bereich kopieren" ein nicht ablaufrelevanter Datenbaustein kopiert wird und der Kopiervorgang unterbrochen wird, kann die Bearbeitung der Anweisung nicht mehr fortgesetzt werden.

Falls auf Ihrer CPU die Anweisung "Aus Datenbaustein im Ladespeicher lesen" vorhanden ist, müssen Sie diese Anweisung zum Lesen von nicht ablaufrelevanten Datenbausteinen im Ladespeicher benutzen. Wenn Sie die Anweisung "Bereich kopieren" benutzen, wird der Fehler W#16#8092 ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
DSTBLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BLKMOV ANY
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_ErrorCode" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

UBLKMOV: Bereich ununterbrechbar kopieren

Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 512 Bytes kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRCBLK	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der kopiert wird (Quellbereich).
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DSTBLK	Output [1]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).
1) Der Parameter DSTBLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8091	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL UBLKMOV ANY
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

Die Anweisung kopiert 10 Bytes ab MB100 und schreibt diese in den DB1. Wenn ein Fehler während des Kopiervorgangs auftritt, wird dessen Fehlercode in der Variable "Tag_ErrorCode" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

FILL: Bereich befüllen

Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch ANY-Zeiger.

Hinweis

In dieser Anweisung können Sie nur Datentypen verwenden, die auf einer CPU der Baureihe S7-300/400 gültig sind.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:

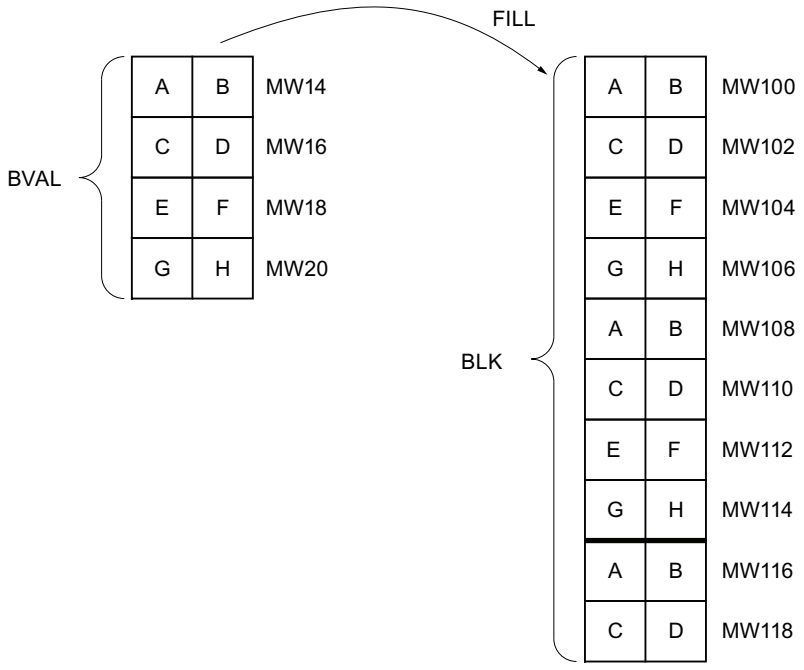

Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

Hinweis

Falls der tatsächlich vorhandene Ziel- oder Quellbereich kleiner als ein am Parameter BVAL oder BLK angegebener Ziel- oder Quellbereich ist, hängt das Verhalten vom CPU-Typ ab:

- Für S7-300-CPUs gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#837F ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für S7-400 V4-CPUs (V4, V4 H-CPU und V4.5. H-CPU) gilt:
  Es werden keine Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.
- Für alle anderen S7-400-CPUs gilt:
  Es werden Daten übertragen. Am Parameter RET_VAL wird der Fehlercode W#16#8122 bzw. W#16#8323 ausgegeben. Dieser Fehlercode muss in Ihrem Anwenderprogramm ausgewertet werden.

Regeln beim Kopieren von Strukturen

Wenn Sie als Eingangsparameter eine Struktur übergeben, müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BVAL	Input	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.
RET_VAL	Output	INT	E, A, M, D, L, P	Fehlerinformationen: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BLK	Output [1)]	ANY	E, A, M, D, L, P	Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.
1) Der Parameter BLK ist als Output deklariert, muss aber in der Bausteinschnittstelle als InOut deklariert werden.				

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8092	Der Quell- oder Zielbereich liegt nur im Ladespeicher.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL FILL ANY
     (BVAL := P#M14.0 WORD 4
      RET_VAL => "Tag_ErrorCode"
      BLK => P#M100.0 WORD 10
      )
```

Die Anweisung kopiert den Quellbereich von MW14 bis MW20 und befüllt den Zielbereich von MW100 bis MW118 mit dem Inhalt der 4 Wörter, die im Speicherbereich im Parameter BVAL enthalten sind.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

Umwandler

CONVERT: Wert konvertieren

Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Operanden und konvertieren ihn entsprechend der parametrierten Datentypen.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P oder Konstante	Wert, der konvertiert wird
<Ergebnis	Output	Ganzzahlen, Gleitpunktzahlen, BCD16, BCD32	E, A, M, D, L, P	Ergebnis der Konvertierung

Aus den Klapplisten "???" können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := INT_TO_DINT("Tag_InValue")
```

Der Inhalt des Operanden "Tag_InValue" wird gelesen und in eine Ganzzahl (32 Bit) umgewandelt. Das Ergebnis wird im Operanden "Tag_OutValue" abgelegt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ROUND: Zahl runden

Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Operanden zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in eine Ganzzahl vom Datentyp DINT um. Wenn der Eingangswert zwischen einer geraden und einer ungeraden Zahl liegt, wird die gerade Zahl gewählt.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert, der gerundet wird
<Ergebnis>	Output	DINT	E, A, M, D, L, P	Ergebnis der Rundung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ROUND("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	2	-2

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächsten geraden Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Operanden zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung kann dabei größer oder gleich dem Eingangswert sein.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst höherer Ganzzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := CEIL("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	1	0

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächst höheren Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie den Wert am Operanden zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung kann dabei kleiner oder gleich dem Eingangswert sein.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Ganzzahlen, Gleitpunktzahlen	E, A, M, D, L, P	Ergebnis mit nächst niedrigerer Ganzzahl

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := FLOOR("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	0	-1

Die Gleitpunktzahl am Operanden "Tag_InValue" wird zur nächst niedrigeren Ganzzahl gerundet und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

TRUNC: Ganzzahl erzeugen

Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Operanden eine Ganzzahl. Der Wert wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen als Ergebnis aus.

Der Wert des Ergebnis ist ungültig, wenn Fehler während der Bearbeitung auftreten, z. B. ein Überlauf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Gleitpunktzahlen	E, A, M, D, L oder Konstante	Eingangwert
<Ergebnis>	Output	Ganzzahlen	E, A, M, D, L	Ganzzahliger Teil des Eingangswerts

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := TRUNC("Tag_InValue")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	1	-1

Der ganzzahlige Anteil der Gleitpunktzahl am Operanden "Tag_InValue" wird in eine Ganzzahl umgewandelt und am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT	E, A, M, D, L, P oder Konstante	Eingabewert, der skaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1".
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SCALE
      (IN := "Tag_InputValue"
       HI_LIM := "Tag_HighLimit"
       LO_LIM := "Tag_LowLimit"
       BIPOLAR := "Tag_Bipolar"
       RET_VAL => "Tag_ErrorCode"
       OUT => "Tag_OutputValue"
       )
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1

Parameter	Operand	Wert
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Am Operanden "Tag_InputValue" ist der Wert angegeben, der umgewandelt und skaliert werden soll. Der obere und der untere Grenzwert ist durch die Operanden "Tag_HighLimit" und "Tag_LowLimit" definiert. Mithilfe des Operanden "Tag_Bipolar" = TRUE geben Sie an, dass der Wert am Parameter IN als bipolar interpretiert werden soll. Am Operanden "Tag_OutputValue" wird das Ergebnis der Anweisung ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

UNSCALE: Deskalieren

Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO_LIM und HI_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN nicht innerhalb der von HI_LIM und LO_LIM definierten Grenzen liegt, wird ein Fehler ausgegeben. Und das Ergebnis wird auf die nächstgelegene Grenze gesetzt.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO_LIM > HI_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert, der in einen ganzzahligen Wert deskaliert wird
HI_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Oberer Grenzwert
LO_LIM	Input	REAL	E, A, M, D, L, P oder Konstante	Unterer Grenzwert
BIPOLAR	Input	BOOL	E, A, M, D, L oder Konstante	Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen: 1: Bipolar 0: Unipolar
OUT	Output	INT	E, A, M, D, L, P	Ergebnis der Anweisung
RET_VAL	Output	WORD	E, A, M, D, L, P	Fehlerinformationen

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0008	Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM).

Fehler-code* (W#16#...)	Erläuterung
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie hier: Anzeigeformate im Programmstatus umstellen (Seite 8040)	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL UNSCALE
      (IN := "Tag_InputValue"
       HI_LIM := "Tag_HighLimit"
       LO_LIM := "Tag_LowLimit"
       BIPOLAR := "Tag_Bipolar"
       RET_VAL => "Tag_ErrorCode"
       OUT => "Tag_OutputValue"
       )
```

Die folgende Tabelle zeigt die Werte der einzelnen Operanden vor der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

Die folgende Tabelle zeigt die Werte der einzelnen Operanden nach der Ausführung der Anweisung:

Parameter	Operand	Wert
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Am Operanden "Tag_InputValue" ist der Wert angegeben, der umgewandelt und deskaliert werden soll. Der obere und der untere Grenzwert ist durch die Operanden "Tag_HighLimit" und "Tag_LowLimit" definiert. Mithilfe des Operanden "Tag_Bipolar" = TRUE geben Sie an, dass der Wert am Parameter IN als bipolar interpretiert werden soll. Am Operanden "Tag_OutputValue" wird das Ergebnis der Anweisung ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Grundlagen zu GRAPH (Seite 7685)

Programmsteuerung

Laufzeitsteuerung

COMPRESS: CPU-Speicher komprimieren

Beschreibung

Mit der Anweisung "CPU-Speicher komprimieren" komprimieren Sie den Arbeitsspeicher der CPU bzw. den RAM-Ladespeicher. Dabei werden Lücken, die durch Löschen oder Laden von Bausteinen entstanden sind, aufgefüllt, indem die vorhandenen Bausteine zusammengeschoben werden.

Mit dem Aufruf der Anweisung "CPU-Speicher komprimieren" stoßen Sie den Komprimiervorgang an. Der Komprimiervorgang wird über mehrere Programmzyklen verteilt. Die Überwachung des Komprimiervorgangs erfolgt über die Werte der einzelnen Parameter. Wenn am Parameter BUSY der Wert "1" ansteht, ist der Komprimiervorgang noch in Ausführung. Der Wert "1" am Parameter DONE meldet, dass der Komprimiervorgang beendet ist. Fehlerinformationen können am Parameter RET_VAL abgefragt werden.

Wenn bereits ein extern angestoßener Komprimiervorgang ausgeführt wird, führt der Aufruf der Anweisung "CPU-Speicher komprimieren" zur Fehleranzeige.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation: Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.
BUSY	Output	BOOL	E, A, M, D, L	Information, ob Komprimiervorgang aktiv ist
DONE	Output	BOOL	E, A, M, D, L	Information, ob der Komprimiervorgang beendet wurde

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler Der Status des Komprimiervorgangs kann an den Parametern BUSY und DONE abgefragt werden.
8091	Ein extern angestoßener Komprimiervorgang wird bereits ausgeführt.
8092	Die Anweisung kann nicht ausgeführt werden, da eine der folgenden Bedingungen zutrifft: • Die Funktion "Baustein löschen" ist aktiv. • Eine Test- und Inbetriebsetzungsfunktion wirkt auf einen Baustein ein (z. B. Status). • Die Funktion "Kopieren von Bausteinen" ist aufgrund eines externen Anstoßes aktiv. • Die H-CPU befindet sich in einem Ankoppel- oder Aktualisierungsvorgang.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

RE_TRIGR: Zyklusüberwachungszeit neu starten

Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zyklusüberwachungszeit der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm oder einem Diagnosealarm, aufgerufen wird, wird die Anweisung nicht ausgeführt.

Die Anweisung "Zyklusüberwachungszeit neu starten" läuft innerhalb einer Zeitspanne (10-mal der maximale Programmzyklus) erfolgreich ab, unabhängig von der Zahl der Aufrufe. Nachdem diese Zeit abgelaufen ist, kann der Programmzyklus nicht mehr verlängert werden.

Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

STP: Programm beenden

Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOP hängen von der CPU-Konfiguration ab.

Parameter

Die Anweisung "Programm beenden" hat keine Parameter und liefert keine Fehlerinformationen.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

WAIT: Zeitverzögerung programmieren

Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine parametrierte Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT an.

Sie können Verzögerungszeiten von -32768 bis zu 32767 Mikrosekunden (µs) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung.

Die Ausführung der Anweisung kann durch höherpriore Ereignisse unterbrochen werden und liefert keine Fehlerinformationen.

Gültig für S7-300-CPUs:

Bei den S7-300-CPUs (mit Ausnahme von S7-318) ist die mit der Anweisung programmierte Verzögerungszeit eine Mindestzeit. Sie verlängert sich um die Ausführungszeit der eingeschachtelten Prioritätsklassen sowie um Systemlasten.

Hinweis

Negative Verzögerungszeit

Wenn Sie am Parameter WT eine negative Verzögerungszeit angeben, dann liefern der Freigabeausgang ENO, bzw. das VKE und das BR-Bit, den Signalzustand FALSE. Eine negative Verzögerungszeit hat keinen Einfluss auf die CPU. In KOP und FUP werden die folgenden, an den Freigabeausgang ENO angebundenen, Anweisungen nicht ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WT	Input	INT	E, A, M, D, L, P oder Konstante	Verzögerungszeit in Mikrosekunden (µs)

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

PROTECT: Schutzstufe ändern

Beschreibung

Mit der Anweisung "Schutzstufe ändern" ändern Sie die projektierte Schutzstufe "1" der CPU ab.

Die parametrierte Schutzstufe "1" mit aktivierter Option "Durch Passwort aufhebbar" bewirkt, dass man ein Passwort parametrieren muss. Mit diesem Passwort können Sie auf die mit der Anweisung "Schutzstufe ändern" schreibgeschützte CPU im Betrieb schreibend zugreifen (z. B. Laden).

Die Schutzstufe legen Sie am Parameter MODE der Anweisung fest. Die aktuelle Schutzstufe der CPU wird im Inspektorfenster unter "Eigenschaften > Schutz" angezeigt.

Die folgende Tabelle zeigt, welche Bedeutung die Schutzstufen haben:

Schutzstufe	Bedeutung
1	Alle PG-Funktionen sind erlaubt. Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
2	• Das Programm und die Konfiguration der CPU können nicht geändert werden. Das Programm in der CPU kann gelesen werden. • Die Funktionen für Prozessführung, Prozessbeobachtung und Prozesskommunikation sind erlaubt. • Alle Auskunftsfunktionen sind erlaubt. • Mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreibschutz aufheben. • Eine gegebenenfalls vorhandene Sperre der Passwort-Legitimierung wird durch den Aufruf der Anweisung "Schutzstufe ändern" aufgehoben.
3	• Auch mit Kenntnis des gültigen Passworts können Sie den eingestellten Schreib- und Leseschutz nicht aufheben. • Wenn zum Zeitpunkt des Aufrufs der Anweisung "Schutzstufe ändern" eine legitimierte Verbindung vorhanden ist, bleibt der Aufruf der Anweisung für diese Verbindung ohne Wirkung.

Wenn Sie für Ihre CPU die Schutzstufe "2" oder "3" projektiert haben, hat der Aufruf der Anweisung "Schutzstufe ändern" keine Wirkung.

Die Wirksamkeit der Schutzstufe "2" bzw. "3" nach Ausführung der Anweisung "Schutzstufe ändern" können Sie der SZL-Teilliste mit der SZL-ID W#16#0232 und dem Index W#16#0004 entnehmen.

Hinweis

Mit der Anweisung "Schutzstufe ändern" können Sie keine niedrigere Schutzstufe einstellen als die, die Sie mit STEP 7 "Hardware konfigurieren" projektiert haben. Sie haben z. B. die Schutzstufe 3 projektiert und anschließend die Anweisung "Schutzstufe ändern" mit dem Parameter MODE = 12 aufgerufen. Dies bewirkt eine Sperre der Passwort-Legitimierung. Ein erneuter Aufruf der Anweisung mit dem Parameter MODE = 0 oder 1 bewirkt, dass die Schutzstufe 3 beibehalten wird und dass der Schreib- und Leseschutz mit Eingabe des gültigen Passworts aufgehoben wird.

Auswirkungen der Betriebszustandsübergänge auf die Schutzstufe

Die folgende Tabelle zeigt, welche Auswirkungen die Betriebszustandsübergänge auf die mit der Anweisung "Schutzstufe ändern" eingestellte Schutzstufe haben.

Aktion	Auswirkungen auf die Schutzstufe
• Betriebsartenschalter auf STOP • Ungepuffertes NETZ-EIN • Urlöschen durch Betriebsartenschalter • Konfigurations- und/oder Programmänderungen im Betriebszustand STOP der S7-400-CPU	Die Schutzstufe wird auf "1" gesetzt. Alle PG-Funktionen sind erlaubt.
• Kaltstart • Neustart (Warmstart)	Die Schutzstufe wird durch das Betriebssystem auf "1" gesetzt. Alle PG-Funktionen sind erlaubt. Bei Bedarf können Sie durch Aufruf der Anweisung "Schutzstufe ändern" in Ihrem Programm die Schutzstufe auf "2" oder "3" setzen.
Gepuffertes NETZ-EIN	Bleibt unverändert
Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch den Aufruf der Anweisung "Programm beenden" (STP), durch Programmfehler ohne zugehörigen Fehler-OB oder durch Anwenderaktion).	Bleibt unverändert
S7-400: Wiederanlauf durch Anwenderaktion oder automatisch nach NETZ-EIN	Bleibt unverändert

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	WORD	E, A, M, D, L oder Konstante	Schutzstufe Mögliche Werte: • W#16#0000: Einstellen der Schutzstufe "1" • W#16#0001: Einstellen der Schutzstufe "2" mit Passwort-Legitimierung • W#16#000C: Einstellen der Schutzstufe "3" ohne Passwort-Legitimierung Am Parameter MODE können auch Konstanten angegeben werden.
RET_VAL	Output	INT	E, A, M, D, L	Tritt während der Bearbeitung ein Fehler auf, wird eine Fehlerinformation ausgegeben.

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
8090	Unzulässiger Wert am Parameter MODE.
80C3	Die benötigten Betriebsmittel sind momentan belegt.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

CiR: CiR-Vorgang steuern

Beschreibung

Mit der Anweisung "CiR-Vorgang steuern" beeinflussen Sie im Anwenderprogramm den CiR-Vorgang wie folgt:

- Den CiR-Vorgang sperren. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU stets abgelehnt. Die Sperre gilt so lange, bis Sie diese mit der Anweisung "CiR-Vorgang steuern" wieder aufheben.
- Den CiR-Vorgang bedingt sperren, indem Sie am Parameter FRZ_TIME eine Obergrenze für die CiR-Synchronisationszeit festlegen. In diesem Fall wird das Laden einer geänderten Konfiguration vom Programmiergerät in die CPU nur dann zugelassen, wenn die in der CPU errechnete CiR-Synchronisationszeit die angegebene Obergrenze nicht überschreitet.
- Den CiR-Vorgang freigeben. Die Obergrenze der CiR-Synchronisationszeit wird auf den Defaultwert 1000 ms gesetzt.
- Ermitteln, ob der CiR-Vorgang freigegeben ist oder nicht. Falls der CiR-Vorgang freigegeben oder bedingt freigegeben ist, erhalten Sie im Parameter A_FT der Anweisung die aktuelle Obergrenze für die CiR-Synchronisationszeit.

Hinweis

Während der CiR-Synchronisationszeit sind die Ausgänge eingefroren und die Eingänge werden nicht ausgewertet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte am Parameter MODE sind: • 0: Auskunftsfunktion • 1: CiR-Vorgang freigeben • 2: CiR-Vorgang sperren • 3: CiR-Vorgang bedingt sperren. Die Obergrenze der CiR-Synchronisationszeit legen Sie in FRZ_TIME fest.
FRZ_TIME	Input	TIME	E, A, M, D, L oder Konstante	Obergrenze der CiR-Synchronisationszeit in Millisekunden. Zulässiger Wertebereich: 60 ... 2500 ms (Voreinstellung: 1000 ms) Der Parameter FRZ_TIME ist nur dann relevant, wenn der Parameter MODE auf den Wert "3" gesetzt ist.
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation Wenn der Parameter MODE den Wert "0" hat, liefert der Parameter RET_VAL Informationen über die Freigabe des CiR-Vorgangs.
A_FT	Output	TIME	E, A, M, D, L	Aktuell gültige Obergrenze der CiR-Synchronisationszeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt. Dieser Fehlercode kann ausgegeben werden, nur wenn der Parameter MODE auf den Wert "1", "2" oder "3" gesetzt ist.
0001	Der CiR-Vorgang ist freigegeben. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.

Fehler-code* (W#16#...)	Erläuterung
0002	Der CiR-Vorgang ist gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
0003	Der CiR-Vorgang ist bedingt gesperrt. Dieser Fehlercode kann nur ausgegeben werden, wenn der Parameter MODE auf den Wert "0" gesetzt ist.
8001	Die CPU ist nicht bereit für einen CiR-Vorgang. Sie verwenden eine H-CPU im H-System (Solobetrieb) oder Sie arbeiten mit einer Standard-CPU im Multicomputing-Betrieb.
8002	Der Wert am Parameter MODE ist nicht zulässig.
8003	Der Wert am Parameter FRZ_TIME ist nicht zulässig.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Mit der Anweisung "CiR-Vorgang steuern" können Sie beispielsweise programmieren, dass kein CiR-Vorgang in Zeitabschnitten angestoßen wird, in denen für die Prozessbearbeitung die maximale Leistungsfähigkeit der CPU benötigt wird.

Dazu muss die Anweisung "CiR-Vorgang steuern" vor Beginn des Zeitabschnitts mit erhöhter Prozessaktivität im Programm aufgerufen und mit dem Wert "2" (CiR-Vorgang sperren) am Parameter MODE ausgeführt werden. Dadurch wird den CiR-Vorgang gesperrt.

Um den CiR-Vorgang nach der Beendigung des Zeitabschnitts mit erhöhter Prozessaktivität wieder freizugeben, muss die Anweisung "CiR-Vorgang steuern" erneut aufrufen und mit dem Wert "1" am Parameter MODE ausführt werden. Sie können alternativ den Parameter MODE auch auf den Wert "3" setzen und dadurch den CiR-Vorgang bedingt sperren.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

Wortverknüpfungen

NOT: Invertieren

Beschreibung

Mit der Anweisung "Invertieren" invertieren Sie den Signalzustand der Bits des Operanden. Bei der Bearbeitung der Anweisung wird der Wert des Operanden mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und als Ergebnis ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	Eingangswert
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L, P	Einerkomplement des Operandenwertes

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"TagOut_Value" := NOT("TagIn_Value")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
TagIn_Value	W#16#000F
TagOut_Value	W#16#FFF0

Die Anweisung kehrt den Signalzustand der einzelnen Bits am Operanden "TagIn_Value" um und schreibt das Ergebnis in den Operanden "TagOut_Value".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

DECO: Decodieren

Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit am Parameter OUT, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD, INT	E, A, M, D, L, P oder Konstante	Position des Bits im Ausgangswert, das gesetzt wird.
OUT	Output	DWORD	E, A, M, D, L, P	Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL DECO INT_DWORD
     (IN := "Tag_Input"
      OUT => "Tag_Output"
      )
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

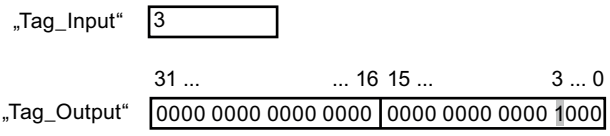

Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "Tag_Input" am Eingang ab und setzt das dritte Bit im Wert des Operanden "Tag_Output" am Ausgang.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ENCO: Encodieren

Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT. Wenn der Eingang IN den Wert DW#16#00000001 oder DW#16#00000000 liefert, wird am Parameter OUT der Wert "0" ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Eingangswert
OUT	Output	INT	E, A, M, D, L, P	Ausgangswert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL ENCO DWORD
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

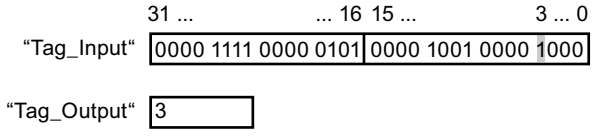

Die Anweisung selektiert das niederwertigste gesetzte Bit am Eingang "TagInput" und schreibt die Bitposition "3" in die Variable am Ausgang "Tag_Output".

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SEL: Selektieren

Beschreibung

Mit der Anweisung "Selektieren" wählen Sie, abhängig von einem Schalter (Eingang G), einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom gleichen Datentyp sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
G	Input	BOOL	E, A, M, D, L, T, Z	Schalter
IN0	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Erster Eingangswert
IN1	Input	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Zweiter Eingangswert
OUT	Output	Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, CHAR, TOD, DATE	E, A, M, D, L, P	Ergebnis

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SEL INT
      (G := "Tag_Input_G"
       IN0 := "Tag_Input0"
       IN1 := "Tag_Input1"
       OUT => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
G	Tag_Input_G	0	1
IN0	Tag_Input0	W#16#0000	W#16#4C
IN1	Tag_Input1	W#16#FFFF	W#16#5E
OUT	Tag_Output	W#16#0000	W#16#5E

Abhängig vom Signalzustand am Eingang "Tag_Input_G" wird der Wert am Eingang "Tag_Input0" oder "Tag_Input1" selektiert und in den Ausgang "Tag_Output" kopiert.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

Schieben und Rotieren

SHR: Rechts schieben

Beschreibung

Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operand1 bitweise nach rechts. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operand1 mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt des Operand1 vom Datentyp INT um vier Bitstellen nach rechts verschoben wird:

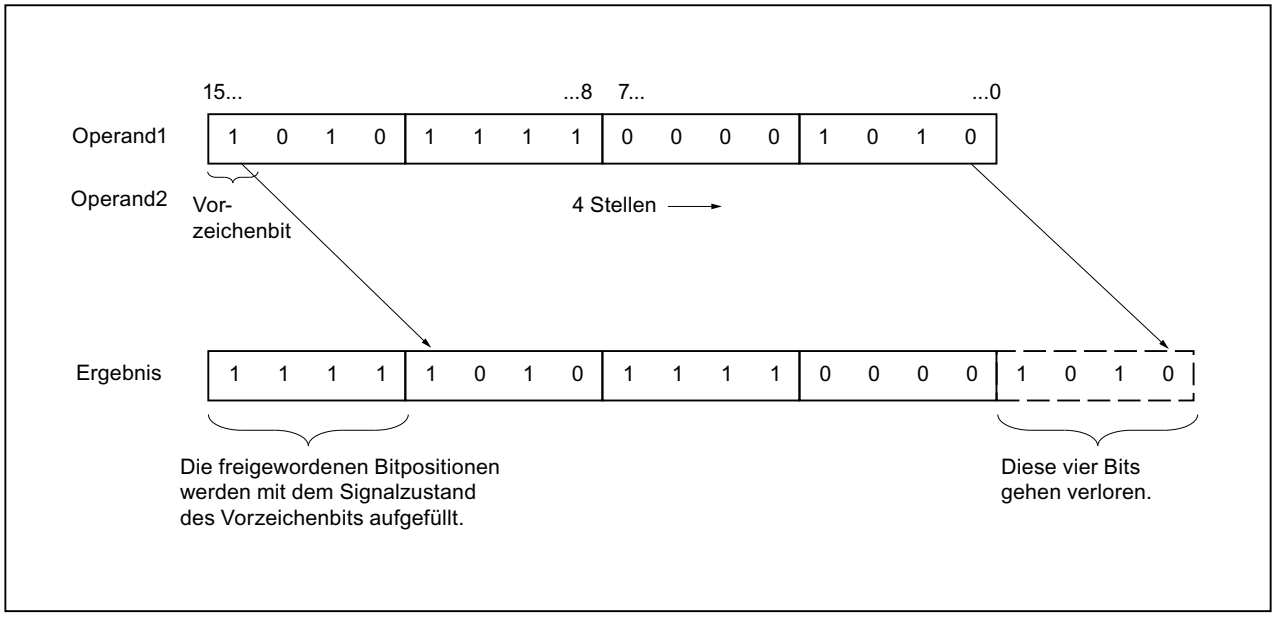

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird
<Operand2>	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SHR_INT("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0011 1111 1010 1111
Tag_Number	3
Tag_OutValue	0000 0111 1111 0101

Der Inhalt des Operanden "Tag_InValue" wird um drei Bitstellen nach rechts verschoben und als Ergebnis im Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SHL: Links schieben

Beschreibung

Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operand1 bitweise nach links. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operand1 werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt des Operand1 vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:

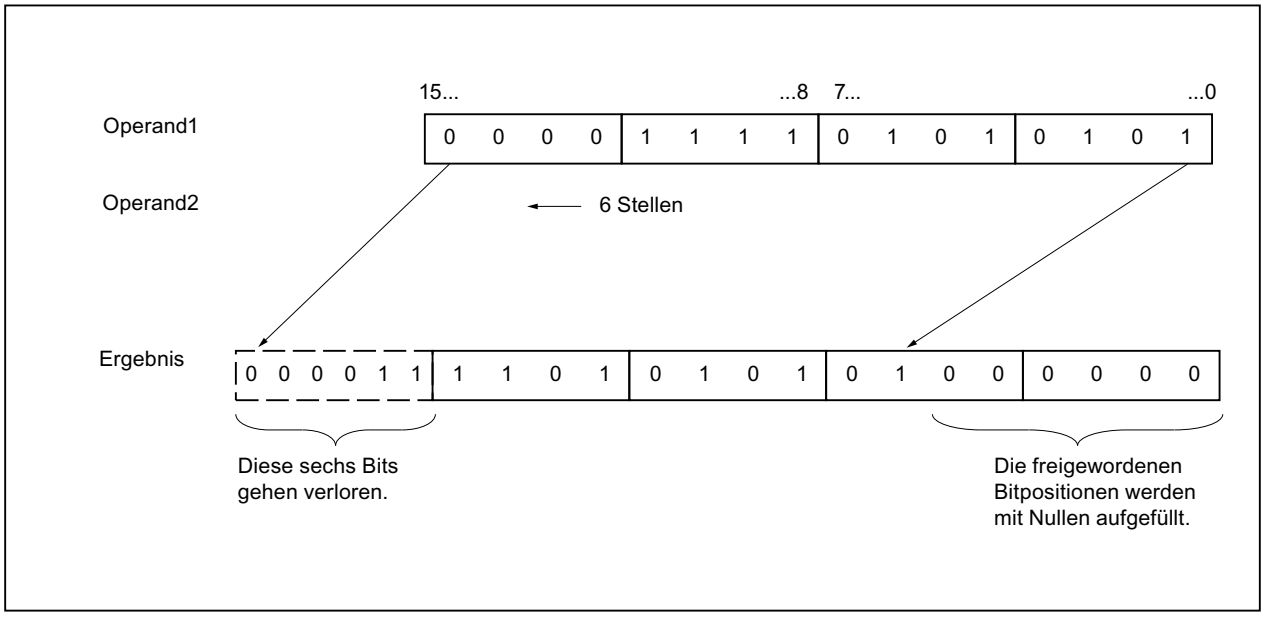

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Bitfolgen, Ganzzahlen	E, A, M, D, L, P oder Konstante	Wert, der verschoben wird
<Operand2>	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert verschoben wird
<Ergebnis>	Output	Bitfolgen, Ganzzahlen	E, A, M, D, L	Ergebnis der Anweisung

Aus der Klappliste "???" können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := SHL_WORD("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	0011 1111 1010 1111
Tag_Number	4
Tag_OutValue	1111 1010 1111 0000

Der Inhalt des Operanden "Tag_InValue" wird um vier Bitstellen nach links verschoben und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ROR: Rechts rotieren

Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operand1 bitweise nach rechts. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die angegebene Anzahl der Bitstellen rotiert.

Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

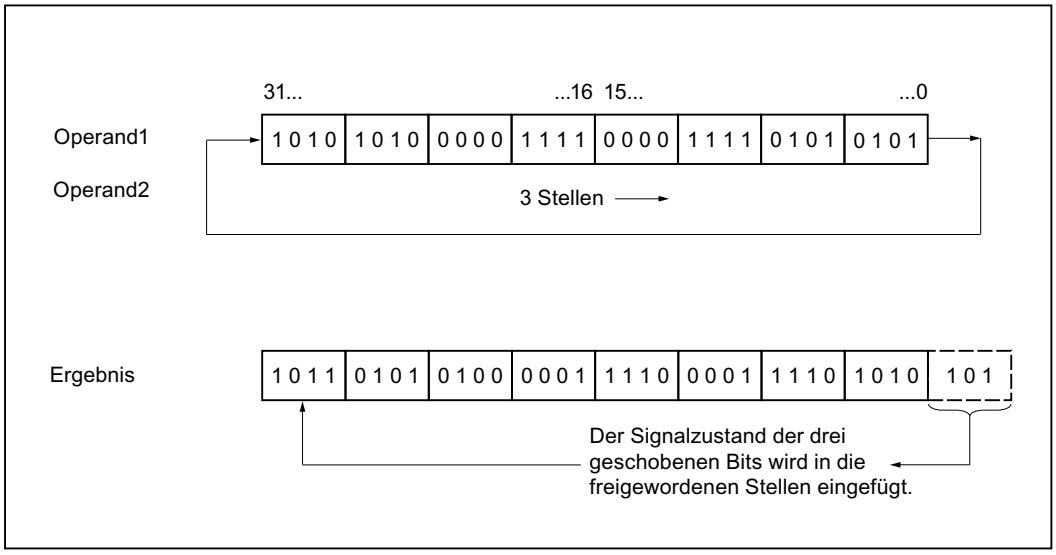

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird
<Operand2>	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
<Ergebnis>	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Information zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ROR("Tag_InValue", "Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1010 1010 0000 1111 0000 1111 0101 0101
Tag_Number	5
Tag_OutValue	1010 1101 0101 0000 0111 1000 0111 1010

Der Inhalt des Operanden "Tag_InValue" wird um fünf Bitstellen nach rechts rotiert und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

ROL: Links rotieren

Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operand1 bitweise nach links. Mit dem Operand2 legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird.

Wenn der Wert des Operand2 "0" ist, wird der Wert des Operand1 in das Ergebnis kopiert.

Wenn der Wert des Operand2 größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Operand1 um die angegebene Anzahl der Bitstellen rotiert.

Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:

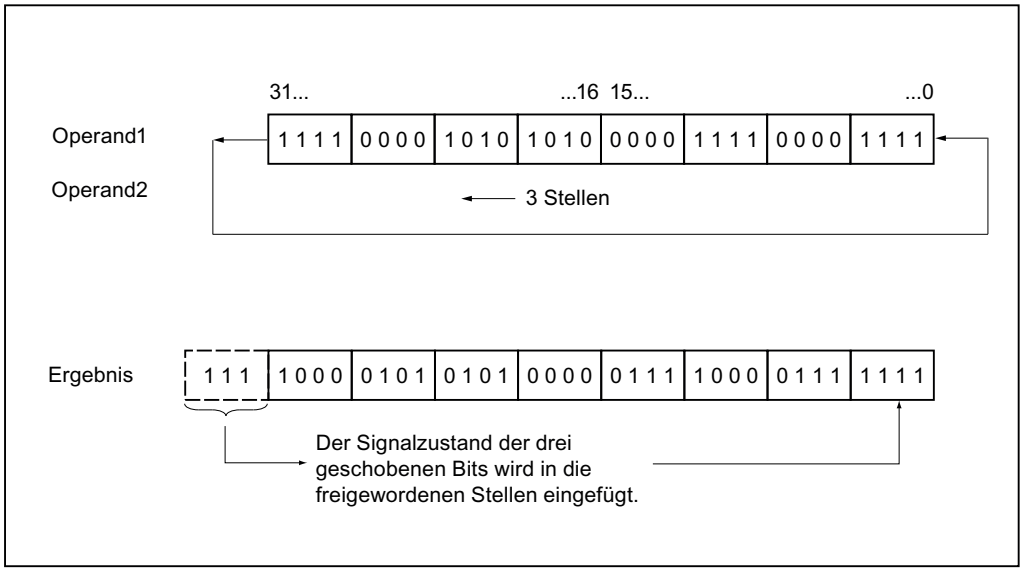

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	BYTE, WORD, DWORD, DINT	E, A, M, D, L, P oder Konstante	Wert, der rotiert wird
<Operand2>	Input	BYTE, WORD, INT	E, A, M, D, L, P oder Konstante	Anzahl der Bitstellen, um die der Wert rotiert wird
<Ergebnis>	Output	BYTE, WORD, DWORD, DINT	E, A, M, D, L	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
"Tag_OutValue" := ROL("Tag_InValue","Tag_Number")
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Operand	Wert
Tag_InValue	1111 0000 1010 1010 0000 1111 0000 1111
Tag_Number	5
Tag_OutValue	0001 0101 0100 0001 1110 0001 1111 1110

Der Inhalt des Operanden "Tag_InValue" wird um fünf Bitstellen nach links rotiert und als Ergebnis am Operanden "Tag_OutValue" ausgegeben.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

Weitere Anweisungen

SET: Bereich bitweise setzen

Beschreibung

Mit der Anweisung "Bereich bitweise setzen" setzen Sie den Signalzustand der Bits in einem angegebenen Bereich auf "1". Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie (P) verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zu setzenden Bits im angegebenen Bereich legen Sie am Parameter N der Anweisung fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Hinweis

Parameter N

Sie können am Parameter N auch ein Element eines Datenbausteins verschalten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der zu setzenden Bits

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SET
     (S_BIT := P#M0.0
      N := 10
      )
```

Die Anweisung setzt 10 Bits im Bereich von M0.0 bis M1.2.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SETP: Bitfeld im Peripheriebereich setzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich setzen" setzen Sie die Bits eines Peripheriebereichs auf den Signalzustand "1". Parallel zu den Peripheriebits werden auch gleichzeitig die entsprechenden Bits im Prozessabbild der Ausgänge gesetzt.

Durch die Parameter N und SA definieren Sie das Bitfeld, das gesetzt wird. Die Bits des Feldes müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zu setzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Setzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bit, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bit im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bit auf den Wert der zugehörigen Bit im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bit, die zum definierten Bitfeld gehören, werden auf den Signalzustand "1" gesetzt.

Die Anweisung "Bitfeld im Peripheriebereich setzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich setzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu setzenden Bit
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste zu setzende Bit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SETP
      (N := 8
       RET_VAL => "Tag_Error"
       SA => P#10.0
       )
```

Die Anweisung setzt 8 Bit im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

SETI: Bereich byteweise setzen

Beschreibung

Mit der Anweisung "Bereich byteweise setzen" setzen Sie den Signalzustand der Bit in einem angegebenen Bereich von Byte auf "1". Den Anfangspunkt des Bereichs definieren Sie am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zu setzenden Bit am Parameter N angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie (P) verweisen. Da auf den Speicherbereich P im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bit im Prozessabbild der Ausgänge wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise setzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie.
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits gesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des zu setzenden Bereichs Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SETI
      (S_BYTE := P#2.0
       N := 16
       )
```

Die Anweisung setzt 2 Byte (16 Bit) im Bereich von P2.0 bis P3.7.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

RESET: Bereich bitweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich bitweise rücksetzen" setzen Sie den Signalzustand der Bit in einem angegebenen Bereich auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BIT. Wenn der Zeiger auf den Speicherbereich der externen Peripherie verweist, wird die Anweisung nicht ausgeführt. Die Anzahl der zurückzusetzenden Bit im angegebenen Bereich legen Sie am Parameter N fest. Wenn der Parameter N den Wert "0" hat, hat der Aufruf der Anweisung keine Wirkung.

Die Anweisung wird nur ausgeführt, wenn das MCR-Bit "1" ist. Wenn das MCR-Bit den Signalzustand "0" führt, werden die Bits im angegebenen Bereich nicht verändert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das erste Bit des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Anzahl der Bits, die zurückgesetzt werden

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL RESET
     (S_BIT := P#M0.0
      N := 10
      )
```

Die Anweisung setzt 10 Bits im Bereich von M0.0 bis M1.2 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

RESETP: Bitfeld im Peripheriebereich rücksetzen

Beschreibung

Mit der Anweisung "Bitfeld im Peripheriebereich rücksetzen" setzen Sie die Bit eines Peripheriebereichs auf den Signalzustand "0" zurück. Parallel zu den Peripheriebit werden auch gleichzeitig die entsprechenden Bit im Prozessabbild der Ausgänge zurückgesetzt.

Durch die Parameter N und SA der Anweisung definieren Sie das Bitfeld, das zurückgesetzt wird. Die Bit des Felds müssen in dem Teil des Peripheriebereichs liegen, dem ein Prozessabbild zugeordnet ist.

Falls zu einem Teil des selektierten Bitfelds keine Peripherie gesteckt ist, versucht die Anweisung das gesamte Bitfeld zurückzusetzen. In diesem Fall wird am Parameter RET_VAL eine Fehlerinformation ausgegeben.

Das Zurücksetzen des Peripheriebereichs erfolgt byteweise. Falls das über die Parameter N und SA definierte Bitfeld nicht auf einer Bytegrenze beginnt oder endet, hat der Aufruf der Anweisung die folgende Wirkung:

- Die Bit, die im ersten oder im letzten zu übertragenden Byte liegen und nicht zum definierten Bitfeld gehören, erhalten den Wert der zugehörigen Bit im Prozessabbild der Ausgänge.

  ⚠ WARNUNG

  Unbeabsichtigtes Verhalten

  Das Setzen der Bit auf den Wert der zugehörigen Bit im Prozessabbild der Ausgänge kann z. B. zum unbeabsichtigten Anlaufen von Motoren bzw. Abschalten von Kühlsystemen führen.

- Die Bit, die zum definierten Bitfeld gehören, werden auf den Signalzustand "0" zurückgesetzt.

Die Anweisung "Bitfeld im Peripheriebereich rücksetzen" muss in einem MCR-Bereich aufgerufen werden.

Der Aufruf der Anweisung "Bitfeld im Peripheriebereich rücksetzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Parameter N liefert den Wert "0"
- Das MCR-Bit hat den Signalzustand "1".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N	Input	INT	E, A, M, D, L oder Konstante	Anzahl der Bits, die zurückgesetzt werden
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
SA	Output	POINTER	P	Zeiger auf das erste Bit, das zurückgesetzt wird

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL RESETP
     (N := 8
      RET_VAL => "Tag_Error"
      SA => P#10.0
      )
```

Die Anweisung setzt 8 Bit im Peripheriebereich entsprechend den Ausgängen A10.0 bis A10.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anzeigeformate im Programmstatus umstellen (Seite 8040)

Grundlagen zu GRAPH (Seite 7685)

RESETI: Bereich byteweise rücksetzen

Beschreibung

Mit der Anweisung "Bereich byteweise rücksetzen" setzen Sie den Signalzustand der Bit in einem angegebenen Bereich von Byte auf "0" zurück. Den Anfangspunkt des Bereichs definieren Sie durch den Zeiger am Parameter S_BYTE. Die Größe des Bereichs definieren Sie, indem Sie die Anzahl der zurückzusetzenden Bit am Parameter N der Anweisung angeben.

Hinweis

Der Wert des Parameters N muss ein Vielfaches der Zahl 8 sein (z. B. 8, 16, 24 usw.).

Der am Parameter S_BYTE angegebene Zeiger muss auf den Speicherbereich der externen Peripherie verweisen. Da auf den Speicherbereich P im Byte-, Wort- und Doppelwortformat zugegriffen wird, muss der Zeiger am Parameter S_BYTE auf einen Operanden zeigen, der einem Byte-Grenzwert entspricht, d. h. die Bitnummer des Zeigers muss "0" sein.

Hinweis

Der Signalzustand der entsprechenden Bit im Prozessabbild der Ausgänge wird auf "0" zurückgesetzt.

Der Aufruf der Anweisung "Bereich byteweise rücksetzen" hat keine Wirkung, wenn eine der folgenden Bedingungen zutrifft:

- Der Zeiger am Parameter S_BYTE zeigt nicht auf den Speicherbereich der externen Peripherie.
- Der Zeiger am Parameter S_BYTE zeigt auf einen Operanden, der keinem Byte-Grenzwert entspricht.
- Der Wert des Parameters N ist kein Vielfaches der Zahl 8.
- Das MCR-Bit hat den Signalzustand "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
S_BYTE	Input	POINTER	P oder Konstante	Zeigt auf das erste Byte des Bereichs, dessen Bits zurückgesetzt werden. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	INT	E, A, M, D, L, P oder Konstante	Größe des Bereichs, dessen Bits zurückgesetzt werden. Der Bereich wird über die Anzahl der Bits als Vielfaches der Zahl 8 (z. B. 8, 16, 24 usw.) angegeben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL RESETI
     (S_BYTE := P#2.0
      N := 16
```

GRAPH
)

Die Anweisung setzt 2 Byte (16 Bit) im Bereich von P2.0 bis P3.7 zurück.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

REPL_VAL: Ersatzwert eintragen

Beschreibung

Mit der Anweisung "Ersatzwert eintragen" tragen Sie von einem Synchronfehler-Organisationsbaustein aus einen Ersatzwert in den Akkumulator 1 (AKKU 1) ein.

Hinweis

Die Anweisung "Ersatzwert eintragen" dürfen Sie nur in Synchronfehler-Organisationsbausteinen (OB 121, OB 122) aufrufen.

Die Anweisung "Ersatzwert eintragen" verwenden Sie, wenn Sie von einer Baugruppe keinen Wert mehr lesen können. Nach jedem Zugriff auf diese Baugruppe wird der OB 122 gestartet. Mit dem Aufruf der Anweisung "Ersatzwert eintragen" können Sie bewirken, dass ein Ersatzwert in den Akkumulator 1 geladen wird. Die Programmbearbeitung wird dann mit diesem Ersatzwert fortgesetzt. Die für die Auswahl des Ersatzwerts notwendigen Informationen können Sie den lokalen Variablen des OB 122 entnehmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VAL	Input	DWORD	E, A, M, D, L oder Konstante	Ersatzwert
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter RET_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET_VAL:

Fehler-code* (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Die Anweisung wurde nicht von einem Synchronfehler-OB aus aufgerufen.
8xyy	Weitere Fehlerinformationen finden Sie unter "Siehe auch".
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

DRUM: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) mit den programmierten Werten des Parameters OUT_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.
JOG	Input	BOOL	E, A, M, D, L	Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.
DRUM_EN	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zuletzt programmierten Schritts.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit (i); Anfänglicher Signalzustand ist "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	E, A, M, D, L	Ausgabebit (j)
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zum Parameter JOG

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EOD	Static	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.
DSP	Static	BYTE	E, A, M, D, L, P	Voreingestellter Schritt des Schrittschaltwerks
DSC	Static	BYTE	E, A, M, D, L, P	Aktueller Schritt des Schrittschaltwerks
DCC	Static	DWORD	E, A, M, D, L, P	Aktueller Zählwert des Schrittschaltwerks
DTBP	Static	WORD	E, A, M, D, L, P	Voreingestellte Zeitbasis des Schrittschaltwerks
PrevTime	Static	TIME	E, A, M, D, L oder Konstante	Vorherige Systemzeit
S_PRESET	Static	ARRAY[1..16] of WORD	E, A, M, D, L	Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.
OUT_VAL	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].
S_MASK	Static	ARRAY[1..16, 0..15] of BOOL	E, A, M, D, L	Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE*	Erläuterung
W#16#0000	Kein Fehler
W#16#000B	Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.
W#16#000C	Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP.
W#16#000D	Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL DRUM, "DRUM_DB"
     (RESET := "Tag_Reset"
      JOG := "Tag_Input_Jog"
      DRUM_EN := "Tag_Input_Drum_EN"
      LST_STEP := "Tag_Number_LastStep"
      EVENTn := "MyTag_Event_n"
      OUTn => "MyTag_Output_n"
      Q => "Tag_Output_Q"
      OUT_WORD => "Tag_OutputWord"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE

Parameter	Adresse	Wert
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

DRUM_X: Schrittschaltwerk realisieren

Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" realisieren Sie ein Schrittschaltwerk mit maximal 16 Schritten. Die Nummer des ersten Schritts legen Sie am Parameter DSP fest. Die Nummer des letzten Schritts geben Sie am Parameter LST_STEP vor.

In jedem Schritt werden alle 16 Ausgangsbits OUT0 bis OUT15 und der Ausgangsparameter OUT_WORD (in dem die Ausgangsbits zusammengefasst sind) beschrieben. Ein Ausgangsbit bekommt entweder das zugehörige Bit des von Ihnen vorgegebenen Felds OUT_VAL oder das Ausgangsbit des vorangegangenen Schritts zugewiesen. Welcher Wert zugewiesen wird, hängt von der Belegung der Maskenbits am Parameter S_MASK ab.

Hinweis

Die Voreinstellung der Maskenbits ist 0. Falls Sie die aktuelle Einstellung eines oder mehrerer Maskenbits ändern wollen, führen Sie diese Änderung im Instanz-Datenbaustein durch.

Die Anweisung "Schrittschaltwerk realisieren" schaltet in den nächsten Schritt, wenn am Parameter JOG eine positive Flanke aufgetreten ist. Falls sich die Anweisung bereits im letzten Schritt befindet, werden bei einer positiven Flanke am Parameter JOG die Variablen Q und EOD gesetzt. Der Parameter DCC wird in diesem Fall auf den Wert 0 gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" gesetzt wird.

Darüber hinaus können Sie auch ein zeitabhängiges Weiterschalten in den nächsten Schritt zulassen. Dazu muss der Parameter DRUM_EN auf den Wert "1" gesetzt werden. Der Übergang in den nächsten Schritt erfolgt, wenn das zum aktuellen Schritt zugehörige Ereignisbit (EVENTi) gesetzt ist und die für den aktuellen Schritt vorgegebene Zeit abgelaufen ist. Die Zeit für einen Schritt ergibt sich aus dem Produkt von der Zeitbasis (DTBP) und dem für den aktuellen Schritt gültigen Zeitfaktor (S_PRESET).

Hinweis

Die im aktuellen Schritt noch verbleibende Bearbeitungszeit (DCC) wird nur dann reduziert, wenn das zugehörige Ereignisbit (EVENTi) gesetzt ist.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Wenn beim Aufruf der Anweisung der Wert "1" am Parameter RESET anliegt, geht das Schrittschaltwerk in den Schritt, dessen Nummer am Parameter DSP angegeben ist.

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist.

Zusätzlich können Sie das Schrittschaltwerk jederzeit über den Wert am Parameter JOG weiterschalten.

Beim Erstaufruf der Anweisung müssen Sie den Parameter RESET mit 1 belegen.

Wenn sich das Schaltwerk im letzten Schritt befindet und die für diesen Schritt vorgegebene Bearbeitungszeit abgelaufen ist, werden die Parameter Q und EOD gesetzt. Die Anweisung verbleibt solange im letzten Schritt, bis der Parameter RESET auf den Wert "1" wechselt.

Die Anweisung wird nur in den Betriebszuständen ANLAUF und RUN ausgeführt.

Das Betriebssystem setzt die Anweisung "Schrittschaltwerk realisieren" bei Kaltstart zurück. Bei einem Neustart (Warmstart) wird die Anweisung nicht zurückgesetzt. Falls die Anweisung "Schrittschaltwerk realisieren" nach einem Neustart (Warmstart) initialisiert werden soll, müssen Sie die Anweisung im OB 100 mit dem Wert "1" am Parameter RESET aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Der Signalzustand "1" bewirkt das Rücksetzen des Schaltwerks. Beim Erstaufruf der Anweisung muss der Parameter RESET auf "1" gesetzt werden.
JOG	Input	BOOL	E, A, M, D, L	Eine positive Signalflanke (gegenüber dem letzten Aufruf) bewirkt das Weiterschalten in den nächsten Schritt, falls sich das Schaltwerk noch nicht im letzten Schritt befindet. Die Weiterschaltung erfolgt unabhängig davon, welchen Wert Sie am Parameter DRUM_EN angegeben haben.
DRUM_EN	Input	BOOL	E, A, M, D, L	Steuerparameter, der festlegt, ob ein zeitabhängiges Weiterschalten in den nächsten Schritt möglich ist. Bei einem Wert "1" ist ein zeitabhängiges Weiterschalten möglich.
LST_STEP	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des letzten Schritts Zulässige Werte sind 1 bis 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	E, A, M, D, L	Ereignisbit Nr. i (gehört zum Schritt i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	E, A, M, D, L	Ausgangsbit Nr. j (identisch mit dem Bit Nr. j von OUT_WORD)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Q	Output	BOOL	E, A, M, D, L	Zustandsparameter, der angibt, ob die von Ihnen vorgegebene Bearbeitungszeit des letzten Schritts abgelaufen ist.
OUT_WORD	Output	WORD	E, A, M, D, L, P	Ausgangsbits zusammengefasst in einer Variablen
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
JOG_HIS	Static	BOOL	E, A, M, D, L	Nicht anwenderrelevant (Eingangsparameter JOG des vorhergehenden Aufrufs)
EOD	Static	BOOL	E, A, M, D, L	Identisch zum Ausgangsparameter Q
DSP	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des ersten Schritts Zulässige Werte sind 1 bis 16.
DSC	Static	BYTE	E, A, M, D, L, P oder Konstante	Nummer des aktuellen Schritts
DCC	Static	DWORD	E, A, M, D, L, P	Die im aktuellen Schritt noch verbleibende Bearbeitungszeit in Millisekunden (ms) (nur relevant, falls DRUM_EN = 1 und das zugehörige Ereignisbit = 1)
DTBP	Static	WORD	E, A, M, D, L, P oder Konstante	Die für alle Schritte gültige Zeitbasis in Millisekunden (ms)
PREV_TIME	Static	DWORD	E, A, M, D, L oder Konstante	(Nicht anwenderrelevant: Systemzeit des vorhergehenden Aufrufs)
S_PRESET	Static	ARRAY of WORD	E, A, M, D, L oder Konstante	Eindimensionales Feld mit den Zeitfaktoren für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16]. In diesem Fall enthält S_PRESET [x] den Zeitfaktor des Schritts x.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_VAL	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den in jedem Schritt auszugebenden Werten, falls Sie diese nicht über S_MASK ausgeblendet haben. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält OUT_VAL [x, y] den Wert, der dem Ausgangsbit OUTy im Schritt x zugewiesen wird.
S_MASK	Static	ARRAY of BOOL	E, A, M, D, L	Zweidimensionales Feld mit den Maskenbits für jeden Schritt. Empfohlene Wahl der Indizes: [1 bis 16, 0 bis 15]. In diesem Fall enthält S_MASK [x, y] das Maskenbit für den "y"-ten auszugebenden Wert im Schritt x. Bedeutung der Maskenbits: • 0: der entsprechende Wert des vorangegangenen Schritts wird dem zugehörigen Ausgangsbit zugewiesen. • 1: der entsprechende Wert aus OUT_VAL wird dem zugehörigen Ausgangsbit zugewiesen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

ERR_CODE* (W#16#...)	Erläuterung
0000	Kein Fehler
8081	Unzulässiger Wert am Parameter LST_STEP
8082	Unzulässiger Wert am DSC
8083	Unzulässiger Wert am DSP
8084	Das Produkt DCC = DTBP * S_PRESET[DSC] überschreitet den Wert 2**31-1 (ca. 24,86 Tage).
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Im folgenden Beispiel geht die Anweisung vom Schritt 1 zum Schritt 2 über. Die Ausgabebits (OUT0 bis OUT15) und das Ausgabewort (OUT_WORD) werden entsprechend der für Schritt 2 konfigurierten Maske und den Werten des Parameters OUT_VAL gesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL DRUM_X, "DRUM_X_DB"
     (RESET := "Tag_Reset"
      JOG := "Tag_Input_Jog"
      DRUM_EN := "Tag_Input_Drum_EN"
      LST_STEP := "Tag_Number_LastStep"
      EVENTn := "MyTag_Event_n"
      OUTn => "MyTag_Output_n"
      Q => "Tag_Output_Q"
      OUT_WORD => "Tag_OutputWord"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Versorgung der Eingangsparameter werden für dieses Beispiel die folgenden Werte verwendet:

Parameter	Operand	Adresse	Wert
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE

Parameter	Adresse	Wert
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

An den Ausgangsparametern liegen vor der Ausführung der Anweisung die folgenden Werte an:

Parameter	Operand	Adresse	Wert
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	TRUE
OUT2	MyTag_Output_2	M4.2	TRUE
OUT3	MyTag_Output_3	M4.3	TRUE
OUT4	MyTag_Output_4	M4.4	TRUE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	TRUE
OUT7	MyTag_Output_7	M4.7	TRUE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	TRUE
OUT11	MyTag_Output_11	M5.3	TRUE
OUT12	MyTag_Output_12	M5.4	TRUE
OUT13	MyTag_Output_13	M5.5	TRUE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	TRUE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Adresse	Wert
OUT0	MyTag_Output_0	M4.0	TRUE
OUT1	MyTag_Output_1	M4.1	FALSE
OUT2	MyTag_Output_2	M4.2	FALSE
OUT3	MyTag_Output_3	M4.3	FALSE
OUT4	MyTag_Output_4	M4.4	FALSE
OUT5	MyTag_Output_5	M4.5	TRUE
OUT6	MyTag_Output_6	M4.6	FALSE
OUT7	MyTag_Output_7	M4.7	FALSE
OUT8	MyTag_Output_8	M5.0	TRUE
OUT9	MyTag_Output_9	M5.1	TRUE
OUT10	MyTag_Output_10	M5.2	FALSE
OUT11	MyTag_Output_11	M5.3	FALSE
OUT12	MyTag_Output_12	M5.4	FALSE
OUT13	MyTag_Output_13	M5.5	FALSE
OUT14	MyTag_Output_14	M5.6	TRUE
OUT15	MyTag_Output_15	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Im Instanz-Datenbaustein "DRUM_X_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

DCAT: Diskreter Steuerungszeitalarm

Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O_FB bzw. C_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD_HIS, ET, OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET, OA, CA und CMD_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD_HIS den Signalzustand "1" anliegt und der Parameter O_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD_HIS und O_FB auf "1" gesetzt sind und der Parameter C_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD_HIS und C_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD_HIS und O_FB den Signalzustand "0" liefern und der Parameter C_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O_FB und C_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BOOL	E, A, M, D, L	Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an.
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Zeigt den Status des Parameters CMD
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
CMD_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit zu CMD

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL DCAT, "DCAT_DB"
      (CMD := "Tag_Input_CMD"
       O_FB := "Tag_Input_O_FB"
       C_FB := "Tag_Input_C_FB"
       Q => "Tag_Output_Q"
       OA => "Tag_Output_OA"
       CA => "Tag_Output_CA"
       )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Im Instanz-Datenbaustein "DCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

MCAT: Motorsteuerungszeitalarm

Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

Eingangsparameter								Ausgangsparameter								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Zustand
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarm
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Öffnen starten
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Öffnen
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Geöffnet
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Alarm öffnen
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Schließen starten
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Schließen
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Geschlossen
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Alarm schließen
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Gestoppt

Eingangsparameter	Ausgangsparameter
Legende:	
INC	Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren
PT	PT wird auf den gleichen Wert wie ET gesetzt
X	Nicht anwendbar
< PT	ET < PT
>= PT	ET >= PT
Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE	

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
O_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Öffnen"
C_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Schließen"
S_CMD	Input	BOOL	E, A, M, D, L	Befehlseingabe "Stoppen"
O_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Öffnen
C_FB	Input	BOOL	E, A, M, D, L	Feedback-Eingang beim Schließen
OO	Output	BOOL	E, A, M, D, L	Ausgang "Öffnen"
CO	Output	BOOL	E, A, M, D, L	Ausgang "Schließen"
OA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Öffnen
CA	Output	BOOL	E, A, M, D, L	Alarmausgang beim Schließen
Q	Output	BOOL	E, A, M, D, L	Der Signalzustand "0" zeigt eine Fehlerbedingung an.
ET	Static	DINT	E, A, M, D, L	Aktuelle, abgelaufene Zeit; ein Takt = 1 ms
PT	Static	DINT	E, A, M, D, L	Voreingestellter Zeitwert; ein Takt = 1 ms
PREV_TIME	Static	DWORD	E, A, M, D, L	Vorherige Systemzeit
O_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Öffnen"
C_HIS	Static	BOOL	E, A, M, D, L	Verlaufsbit "Schließen"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL MCAT, "MCAT_DB"
      (O_CMD := "Tag_Input_O_CMD"
       C_CMD := "Tag_Input_C_CMD"
       S_CMD := "Tag_Input_S_CMD"
       O_FB := "Tag_Input_O_FB"
       C_FB := "Tag_Input_C_FB"
       OO => "Tag_OutputOpen"
       CO => "Tag_OutputClosed"
       OA => "Tag_Output_OA"
       CA => "Tag_Output_CA"
       Q => "Tag_Output_Q"
       )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Im Instanz-Datenbaustein "MCAT_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

IMC: Eingabebits mit den Bits einer Maske vergleichen

Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
CMP_STEP	Input	BYTE	E, A, M, D, L, P oder Konstante	Schrittnummer der Maske, mit der verglichen wird.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
CMP_VAL	Static	ARRAY OF BOOL	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE ausgegeben werden:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000A	Der Wert am Parameter CMP_STEP ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SMC: Matrixscanner

Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN_BIT0 bis IN_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN_BIT0 wird mit dem Wert der Maske CMP_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_BIT0	Input	BOOL	E, A, M, D, L	Eingabebit 0 wird mit Bit 0 der Maske verglichen.
IN_BIT1	Input	BOOL	E, A, M, D, L	Eingabebit 1 wird mit Bit 1 der Maske verglichen.
IN_BIT2	Input	BOOL	E, A, M, D, L	Eingabebit 2 wird mit Bit 2 der Maske verglichen.
IN_BIT3	Input	BOOL	E, A, M, D, L	Eingabebit 3 wird mit Bit 3 der Maske verglichen.
IN_BIT4	Input	BOOL	E, A, M, D, L	Eingabebit 4 wird mit Bit 4 der Maske verglichen.
IN_BIT5	Input	BOOL	E, A, M, D, L	Eingabebit 5 wird mit Bit 5 der Maske verglichen.
IN_BIT6	Input	BOOL	E, A, M, D, L	Eingabebit 6 wird mit Bit 6 der Maske verglichen.
IN_BIT7	Input	BOOL	E, A, M, D, L	Eingabebit 7 wird mit Bit 7 der Maske verglichen.
IN_BIT8	Input	BOOL	E, A, M, D, L	Eingabebit 8 wird mit Bit 8 der Maske verglichen.
IN_BIT9	Input	BOOL	E, A, M, D, L	Eingabebit 9 wird mit Bit 9 der Maske verglichen.
IN_BIT10	Input	BOOL	E, A, M, D, L	Eingabebit 10 wird mit Bit 10 der Maske verglichen.
IN_BIT11	Input	BOOL	E, A, M, D, L	Eingabebit 11 wird mit Bit 11 der Maske verglichen.
IN_BIT12	Input	BOOL	E, A, M, D, L	Eingabebit 12 wird mit Bit 12 der Maske verglichen.
IN_BIT13	Input	BOOL	E, A, M, D, L	Eingabebit 13 wird mit Bit 13 der Maske verglichen.
IN_BIT14	Input	BOOL	E, A, M, D, L	Eingabebit 14 wird mit Bit 14 der Maske verglichen.
IN_BIT15	Input	BOOL	E, A, M, D, L	Eingabebit 15 wird mit Bit 15 der Maske verglichen.
OUT	Output	BOOL	E, A, M, D, L	Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde. Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT_STEP	Output	BYTE	E, A, M, D, L, P	Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde.
LAST	Static	BYTE	E, A, M, D, L, P	Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.
CMP_VAL	Static	ARRAY of WORD	E, A, M, D, L	Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehlercode* (W#16#...)	Erläuterung
0000	Kein Fehler
000E	Der Wert am Parameter LAST ist größer als 15.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

LEAD_LAG: Lead- und Lag-Algorithmus

Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD_TIME, LG_TIME und SAMPLE_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei LG_TIME > 4 + SAMPLE_T an folgende Funktion an:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet. Den erstellten Datenbaustein finden Sie in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	REAL	E, A, M, D, L, P oder Konstante	Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.
SAMPLE_T	Input	INT	E, A, M, D, L, P oder Konstante	Abtastzeit
OUT	Output	REAL	E, A, M, D, L, P	Ergebnis der Anweisung
ERR_CODE	Output	WORD	E, A, M, D, L, P	Fehlerinformation
LD_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Voreilzeit in der gleichen Einheit wie die Abtastzeit
LG_TIME	Static	REAL	E, A, M, D, L, P oder Konstante	Nacheilzeit in der gleichen Einheit wie die Abtastzeit
GAIN	Static	REAL	E, A, M, D, L, P oder Konstante	Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).
PREV_IN	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Eingang
PREV_OUT	Static	REAL	E, A, M, D, L, P oder Konstante	Vorheriger Ausgang

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Parameter ERR_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR_CODE:

Fehler-code* (W#16#...)	Erläuterung
0000	Kein Fehler
0009	Der Wert am Parameter GAIN ist kleiner oder gleich Null.
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".	

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

GRAPH

```
CALL LEAD_LAG, "LEAD_LAG_DB"
      (IN := "Tag_Input"
       SAMPLE_T := "Tag_Input_SAMPLE_T"
       OUT => "Tag_Output_Result"
       ERR_CODE => "Tag_ErrorCode"
       )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

Im Instanz-Datenbaustein "LEAD_LAG_DB" der Anweisung sind die folgenden Werte gespeichert:

Parameter	Adresse	Wert
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
OUT	Tag_Output_Result	2.0

Im Instanz-Datenbaustein "LEAD_LAD_DB" der Anweisung werden die folgenden Werte gespeichert:

Parameter	Operand	Wert
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

TONR_X: Zeit akkumulieren

Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines festgelegten Zeitraums, den Sie mithilfe des Eingangs PV angeben. Die Anweisung wird ausgeführt, wenn der Signalzustand am Starteingang TMR_EN von "0" nach "1" wechselt. Gleichzeitig läuft die Zeit mit der am Eingang PV programmierten Zeitdauer ab. Während die Anweisung ausgeführt wird, addiert sie den Zeitwert des Eingangs DELTA_T zum Zeitwert des InOut ET.

Folgende Bedingungen müssen dazu erfüllt sein:

- Der Starteingang TMR_EN liefert den Signalzustand "1".
- Der Zeitwert am InOut ET ist kleiner als der Zeitwert am Parameter PV.
- Der Eingang RESET liefert den Signalzustand "0".

Die Anweisung wird unterbrochen, wenn der Signalzustand am Starteingang TMR_EN von "1" nach "0" wechselt. Sobald der Signalzustand wieder nach "1" wechselt, läuft die Anweisung weiter.

Die Anweisung wird beendet, wenn der Zeitwert am InOut ET den Zeitwert am Eingang PV erreicht hat. Dann liefert der Ausgang Q den Signalzustand "1".

Solange der Ausgang Q den Signalzustand "1" liefert, bleibt der Zeitwert am InOut ET unverändert. Der Zeitwert und der Ausgang Q werden erst wieder auf "0" zurückgesetzt, wenn der Eingang RESET auf den Signalzustand "1" gesetzt wird.

Da die Anweisung "Zeit akkumulieren" die Zykluszeit (DELTA_T) des letzten Zyklus des Organisationsbausteins (OB) verwendet, darf sie nur in zyklischen Organisationsbausteinen aufgerufen werden.

Hinweis

Sie müssen die Zykluszeit des Organisationsbausteins aus den Anlaufinformationen in die globale Variable des Parameters DELTA_T verschieben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TMR_EN (Timer Enable)	Input	BOOL	E, A, M, D, L	Starteingang Startet die Anweisung und den Ablauf der voreingestellten Zeitdauer am Eingang PV
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
PV (Preset Value)	Input	DINT	E, A, M, D, L, P oder Konstante	Maximale Dauer der Zeiterfassung
DELTA_T	Input	INT	E, A, M, D, L oder Konstante	Zykluszeit des Organisationsbausteins des vorherigen Zyklus
Q	Output	BOOL	E, A, M, D, L	Ausgang, der bei Erreichen des voreingestellten Zeitwerts gesetzt wird
ET (Elapsed Time)	InOut	DINT	E, A, M, D, L	Akkumulierter Zeitwert (Aktuell abgelaufener Zeitwert + Zeitwert des Eingangs DELTA_T)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeitwert akkumulieren":

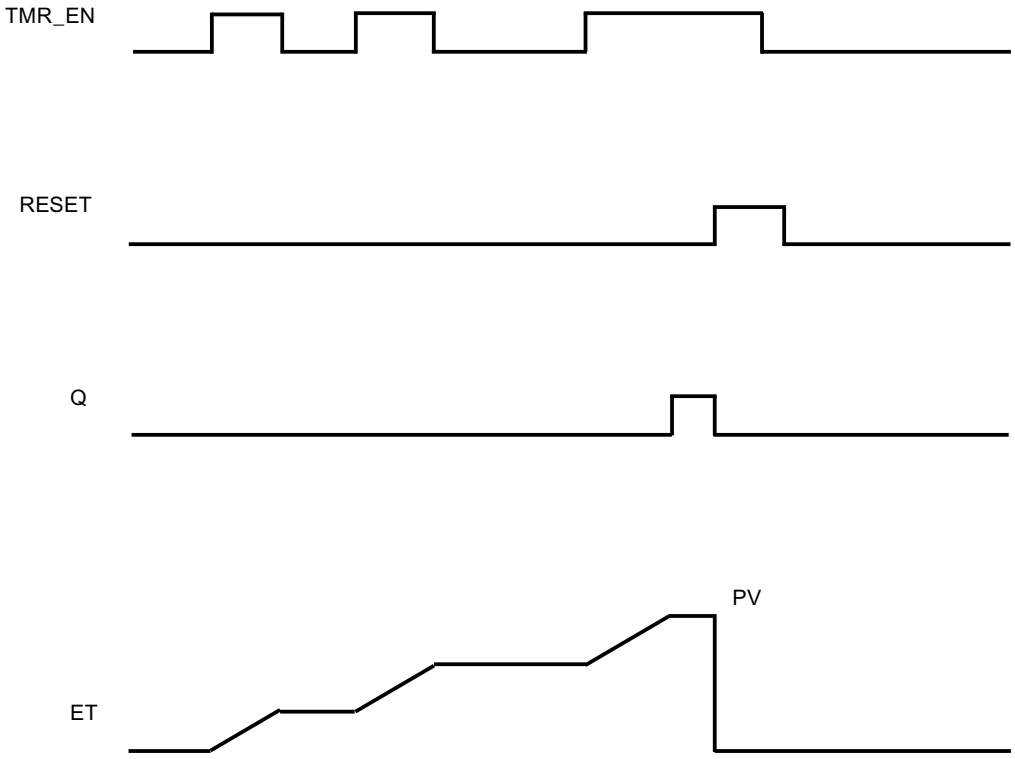

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL TONR_X
     (TMR_EN := "Tag_Start"
      RESET := "Tag_Reset"
      PV := 10
      DELTA_T := 2
      Q => "Tag_Output"
      ET := "Tag_Elapsed_Time"
      )
```

Bei jedem Wechsel des Signalzustands von "0" nach "1" am Eingang TMR_EN, wird die Zykluszeit "2" am Eingang DELTA_T zum aktuellen Zeitwert des InOut ET addiert, bis der maximale Zeitwert "10" des Eingangs PV erreicht ist.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

WSR: Datum im Schieberegister speichern

Beschreibung

Mit der Anweisung "Datum im Schieberegister speichern" schreiben Sie Daten aus einer angegebenen Quelle in ein Schieberegister. Dabei wird jedes Element des Registers in die nächste Adresse verschoben. Die Daten, die in der letzten Adresse des Schieberegisters enthalten sind, gehen nach der Schiebeoperation verloren.

Bei jeder Ausführung der Anweisung mit dem Wert "0" am Parameter RESET werden neue Daten aus der Quelle (S_DATA) gelesen und in die Anfangsadresse (START) des Schieberegisters geschoben. Der Parameter LENGTH gibt die Anzahl der zu schiebenden Elemente an. Am Parameter E_TYPE wird der Datentyp der Elemente festgelegt. Wenn am Parameter E_TYPE ein unzulässiger Datentyp angegeben wird, wird die Anweisung nicht bearbeitet.

Wenn der Parameter RESET bei der Ausführung der Anweisung auf "1" gesetzt ist, wird das Register gelöscht. Der Parameter Q wird auf den Signalzustand "1" gesetzt, wenn das Schieberegister mit Nullen belegt ist oder gelöscht wird.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_DATA	Input	POINTER	E, A, M, D	Zeigt auf das Quellelement, das in die Anfangsadresse (START) geschoben wird. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
START	Input	POINTER	E, A, M, D	Zeigt auf die Anfangsadresse des Registers. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
LEN	Input	WORD	E, A, M, D, L, P	Anzahl der Elemente, die geschoben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der zu schiebenden Elemente an. Folgende Datentypen sind zulässig: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	A, M, D, L	Gibt "0" an, wenn der Parameter RESET aktiv (1) ist oder alle Elemente, die geschoben werden, den Wert "0" haben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Im folgenden Beispiel werden vier Elemente des Datentyps WORD ab der am Parameter START angegebenen Adresse geschoben. Nachdem das erste Element des Registers in die nächste Adresse geschoben wurde, wird die erste Adresse mit den Daten belegt, auf die der Zeiger am Parameter S_DATA zeigt. Der Wert des letzten Elements im Register geht dabei verloren.

GRAPH

```
CALL WSR
      (RESET := "Tag_Input_Reset"
       S_DATA := "Tag_Input_DataSource"
       START := "Tag_Input_StartAddress"
       LEN := "Tag_Input_Length"
       E_TYPE := "Tag_Input_ElementType"
       Q => "Tag_Output_Q"
       )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444

Parameter	Operand	Wert
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SHRB: Bit in Bitschieberegister schieben

Beschreibung

Mit der Anweisung "Bit in Bitschieberegister schieben" schreiben Sie ein Bit aus einer angegebenen Quelle (DATA) in ein Schieberegister. Bei jeder Ausführung der Anweisung mit einem Wert "0" am Parameter RESET werden neue Daten aus der Quelle gelesen und in die Anfangsadresse (S_BIT) des Schieberegisters geschoben. Alle folgenden Bit werden um eins verschoben. Das Bit in der letzten Adresse (S_BIT + N) geht nach dem Schieben verloren. Wenn der Parameter RESET während der Bearbeitung der Anweisung auf "1" gesetzt ist, werden die Adressen im Schieberegister auf "0" zurückgesetzt und nicht geschoben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	Input	BOOL	E, A, M, D, L	Quellbit
RESET	Input	BOOL	E, A, M, D, L	Rücksetzeingang
S_BIT	Input	POINTER	E, A, M, D	Zeigt auf das Anfangsbit des Schieberegisters. Bei einer bereichsübergreifenden, registerindirekten Adressierung muss der Zeiger im Doppelwortformat angegeben werden.
N	Input	WORD	E, A, M, D, L, P	Anzahl der Bits, die geschoben werden.

Beispiel

Im folgenden Beispiel wird der Parameter N auf den Wert 14 ("E" in hexadezimaler Schreibweise) gesetzt, sodass 14 Bits ab der am Parameter S_BIT angegebenen Adresse geschoben werden. Nachdem die Bits geschoben worden sind, wird die erste Adresse mit den Daten aus der Quelle am Parameter DATA belegt. Der letzte Bitwert geht verloren.

GRAPH

```
CALL SHRB
      (DATA := "Tag_Input_Data"
       RESET := "Tag_Input_Reset"
       S_BIT := "Tag_Input_BitSource"
       N := "Tag_Input_Number"
       )
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

Parameter	Operand	Wert
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Vor der Bearbeitung der Anweisung sind die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Nach der Bearbeitung

Nach der Bearbeitung der Anweisung werden die folgenden Signalzustände in den 14 Bits des Registers gespeichert:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

SEG: Bitmuster für 7-Segment-Anzeige erzeugen

Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

Eingangsziffer (Binär)	Belegung der Segmente - g f e d c b a	Anzeige (Hexadezimal)	Sieben-Segment-Anzeige
0000	00111111	0	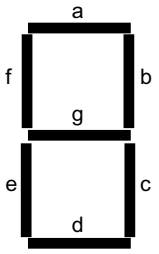
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	WORD	E, A, M, D, L, P oder Konstante	Quellwort mit vier Hexadezimalziffern
OUT	Output	DWORD	E, A, M, D, L, P	Bitmuster für die 7-Segment-Anzeige

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL SEG
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Parameter	Operand	Wert	
Hexadezimal	Binär		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Anzeige: 1234

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

BCDCPL: Zehnerkomplement erzeugen

Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

10000000 (als BCD)

– 7-stelliger BCD-Wert

Zehnerkomplement (als BCD)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	7-stellige BCD-Zahl
ERR_CODE	Output	DWORD	E, A, M, D, L, P	Ergebnis der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BCDCPL
```

GRAPH

```
      (IN := "Tag_Input"
       ERR_CODE => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zum Statuswort (Seite 183)

Grundlagen zu GRAPH (Seite 7685)

BITSUM: Anzahl der gesetzten Bits zählen

Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DWORD	E, A, M, D, L, P oder Konstante	Operand, dessen gesetzte Bit gezählt werden
RET_VAL	Output	INT	E, A, M, D, L, P	Anzahl der gesetzten Bit

Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

GRAPH

```
CALL BITSUM
      (IN := "Tag_Input"
       RET_VAL => "Tag_Output"
       )
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

Parameter	Operand	Wert*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 Bits)
*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".		

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Grundlagen zu GRAPH (Seite 7685)

GRAPH KOP-Anweisungen

Allgemeine KOP-Anweisungen in GRAPH

Beschreibung

Beim Programmieren in GRAPH stehen Ihnen unterschiedliche KOP-Anweisungen zur Verfügung.

Die folgende Tabelle zeigt eine Übersicht der Anweisungen und ihrer Verfügbarkeit:

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
Allgemein		
Netzwerk einfügen	x	
Leerbox	x	x
Verzweigung öffnen	x	x
Verzweigung schließen	x	x
Bitverknüpfungen		
---| |---: Schließerkontakt (Seite 4373)	x	x
---| / |---: Öffnerkontakt (Seite 4375)	x	x
--| NOT |--: VKE invertieren (Seite 4376)	x	x
---( )---: Zuweisung (Seite 4376)	x	
---(R)---: Ausgang rücksetzen (Seite 4377)	x	
---(S)---: Ausgang setzen (Seite 4378)	x	
SR: Flipflop setzen/rücksetzen (Seite 4379)	x	
RS: Flipflop rücksetzen/setzen (Seite 4380)	x	
--|P|--: Operand auf positive Signalflanke abfragen (Seite 4382)	x	
--|N|--: Operand auf negative Signalflanke abfragen (Seite 4383)	x	
P_TRIG: VKE auf positive Signalflanke abfragen (Seite 4385)	x	
N_TRIG: VKE auf negative Signalflanke abfragen (Seite 4386)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
Zeiten		
IEC-Zeiten		
TP: Impuls erzeugen (Seite 4387)	x	
TON: Einschaltverzögerung erzeugen (Seite 4389)	x	
TOF: Ausschaltverzögerung erzeugen (Seite 4391)	x	
SIMATIC-Zeiten		
S_IMPULS: Zeit als Impuls parametrieren und starten (Seite 4393)	x	
S_VIMP: Zeit als verlängerten Impuls parametrieren und starten (Seite 4396)	x	
S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten (Seite 4399)	x	
S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten (Seite 4402)	x	
S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten (Seite 4404)	x	
Zähler		
IEC-Zähler		
CTU: Vorwärts zählen (Seite 4417)	x	
CTD: Rückwärts zählen (Seite 4419)	x	
CTUD: Vorwärts und rückwärts zählen (Seite 4421)	x	
SIMATIC-Zähler		
Z_VORW: Parametrieren und vorwärts zählen (Seite 4424)	x	
Z_RUECK: Parametrieren und rückwärts zählen (Seite 4426)	x	
ZAEHLER: Parametrieren und vorwärts/rückwärts zählen (Seite 4427)	x	
Vergleicher		
CMP ==: Gleich (Seite 4433)	x	x
CMP <>: Ungleich (Seite 4434)	x	x
CMP >=: Größer gleich (Seite 4435)	x	x
CMP <=: Kleiner gleich (Seite 4437)	x	x
CMP >: Größer (Seite 4438)	x	x
CMP <: Kleiner (Seite 4439)	x	x
CMP>T: Größer Schrittaktivierungszeit (Seite 5396)		x
CMP>U: Größer ununterbrochene Schrittaktivierungszeit (Seite 5397)		x
Mathematische Funktionen		
ADD: Addieren (Seite 4440)	x	
SUB: Subtrahieren (Seite 4441)	x	
MUL: Multiplizieren (Seite 4443)	x	
DIV: Dividieren (Seite 4444)	x	
MOD: Divisionsrest gewinnen (Seite 4446)	x	
NEG: Zweierkomplement erzeugen (Seite 4447)	x	
ABS: Absolutwert bilden (Seite 4448)	x	
MIN: Minimum ermitteln (Seite 4449)	x	
MAX: Maximum ermitteln (Seite 4451)	x	
LIMIT: Limitieren (Seite 4453)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
SQR: Quadrat bilden (Seite 4454)	x	
SQRT: Quadratwurzel bilden (Seite 4456)	x	
LN: Natürlichen Logarithmus bilden (Seite 4457)	x	
EXP: Exponentialwert bilden (Seite 4458)	x	
SIN: Sinuswert bilden (Seite 4459)	x	
COS: Cosinuswert bilden (Seite 4461)	x	
TAN: Tangenswert bilden (Seite 4462)	x	
ASIN: Arcussinuswert bilden (Seite 4463)	x	
ACOS: Arcuscosinuswert bilden (Seite 4465)	x	
ATAN: Arcustangenswert bilden (Seite 5161)	x	
Verschieben		
MOVE: Wert kopieren (Seite 4468)	x	
BLKMOV: Bereich kopieren (Seite 4470)	x	
UBLKMOV: Bereich ununterbrechbar kopieren (Seite 4474)	x	
FILL: Bereich befüllen (Seite 4477)	x	
Umwandler		
CONVERT: Wert konvertieren (Seite 4480)	x	
ROUND: Zahl runden (Seite 4481)	x	
CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen (Seite 4483)	x	
FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen (Seite 4484)	x	
TRUNC: Ganzzahl erzeugen (Seite 4485)	x	
SCALE: Skalieren (Seite 4487)	x	
UNSCALE: Deskalieren (Seite 4489)	x	
Programmsteuerung		
Laufzeitsteuerung		
RE_TRIGR: Zyklusüberwachungszeit neu starten (Seite 4499)	x	
STP: Programm beenden (Seite 4499)	x	
WAIT: Zeitverzögerung programmieren (Seite 4500)	x	
PROTECT: Schutzstufe ändern (Seite 4501)	x	
COMPRESS: CPU-Speicher komprimieren (S7-400) (Seite 4497)	x	
CiR: CiR-Vorgang steuern (S7-400) (Seite 4503)	x	
Wortverknüpfungen		
AND: UND verknüpfen (Seite 4506)	x	
OR: ODER verknüpfen (Seite 4507)	x	
XOR: EXKLUSIV ODER verknüpfen (Seite 4508)	x	
INVERT: Einerkomplement erzeugen (Seite 4510)	x	
DECO: Decodieren (Seite 4511)	x	
ENCO: Encodieren (Seite 4512)	x	
SEL: Selektieren (Seite 4514)	x	
Schieben und Rotieren		
SHR: Rechts schieben (Seite 4515)	x	

KOP-Anweisung	Permanente Anweisung	Transition/ Supervision/ Interlock
SHL: Links schieben (Seite 4517)	x	
ROR: Rechts rotieren (Seite 4519)	x	
ROL: Links rotieren (Seite 4521)	x	
Weitere Anweisungen		
SET: Bereich bitweise setzen (Seite 4537)	x	
SETP: Bitfeld im Peripheriebereich setzen (S7-400) (Seite 4538)	x	
SETI: Bereich byteweise setzen (Seite 4540)	x	
RESET: Bereich bitweise rücksetzen (Seite 4542)	x	
RESETP: Bitfeld im Peripheriebereich rücksetzen (S7-400) (Seite 4544)	x	
RESETI: Bereich byteweise rücksetzen (Seite 4546)	x	
REPL_VAL: Ersatzwert eintragen (Seite 4548)	x	
DRUM: Schrittschaltwerk realisieren (Seite 4549)	x	
DRUM_X: Schrittschaltwerk realisieren (Seite 4555)	x	
DCAT: Diskreter Steuerungszeitalarm (Seite 4563)	x	
MCAT: Motorsteuerungszeitalarm (Seite 4566)	x	
IMC: Eingabebits mit den Bits der Maske vergleichen (Seite 4570)	x	
SMC: Matrixscanner (Seite 4572)	x	
LEAD_LAG: Lead- und Lag-Algorithmus (Seite 4575)	x	
TONR_X: Zeit akkumulieren (Seite 4578)	x	
WSR: Datum im Schieberegister speichern (Seite 4581)	x	
SHRB: Bit in Bitschieberegister schieben (Seite 4584)	x	
SEG: Bitmuster für 7-Segment-Anzeige erzeugen (Seite 4586)	x	
BCDCPL: Zehnerkomplement erzeugen (Seite 4588)	x	
BITSUM: Anzahl der gesetzten Bits zählen (Seite 4590)	x	

CMP>T: Größer Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Gesamtdauer eines Schritts. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
<Operand2>	Input	DINT, TIME, S5TIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

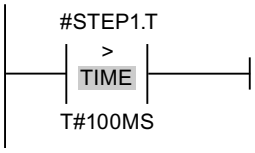

Solange die Aktivierungszeit von #STEP1.T unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

CMP>U: Größer ununterbrochene Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer ununterbrochene Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Dauer eines Schritts, abzüglich eventueller Störungen. Dabei werden etwaige Ereignisse oder Störungen zeitlich nicht mit erfasst und nur die reine Schrittdauer überwacht.

In der programmierten Bedingung wird die gesamte Aktivierungszeit des Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer ununterbrochene Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts abzüglich der Störungen
<Operand2>	Input	DINT, TIME, S5TIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

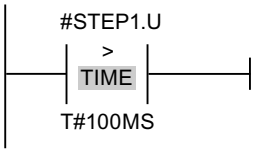

Solange die Aktivierungszeit abzüglich eventueller Störungen von #STEP1.U unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

GRAPH FUP-Anweisungen

Allgemeine FUP-Anweisungen in GRAPH

Beschreibung

Beim Programmieren in GRAPH stehen Ihnen unterschiedliche FUP-Anweisungen zur Verfügung.

Die folgende Tabelle zeigt eine Übersicht der Anweisungen und ihrer Verfügbarkeit:

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
Allgemein		
Neues Netzwerk: Netzwerk einfügen	x	
Leerbox	x	x
Abzweig: Verzweigung öffnen	x	x
-|: Eingang einfügen (Seite 4595)	x	x
-o|: VKE invertieren (Seite 4596)	x	x
Bitverknüpfungen		
&: UND-Verknüpfung (Seite 4591)	x	x
>=1: ODER-Verknüpfung (Seite 4592)	x	x
x: EXCLUSIV ODER-Verknüpfung (Seite 4593)	x	
= : Zuweisung (Seite 4597)	x	
-(R): Ausgang rücksetzen (Seite 4598)	x	
-(S): Ausgang setzen (Seite 4599)	x	
SR: Flipflop setzen/rücksetzen (Seite 4600)	x	
RS: Flipflop rücksetzen/setzen (Seite 4601)	x	
-|P|-: Operand auf positive Signalflanke abfragen (Seite 4602)	x	
-|N|-: Operand auf negative Signalflanke abfragen (Seite 4604)	x	
P_TRIG: VKE auf positive Signalflanke abfragen (Seite 4605)	x	
N_TRIG: VKE auf negative Signalflanke abfragen (Seite 4607)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
Zeiten		
IEC-Zeiten		
TP: Impuls erzeugen (Seite 4608)	x	
TON: Einschaltverzögerung erzeugen (Seite 4610)	x	
TOF: Ausschaltverzögerung erzeugen (Seite 4612)	x	
SIMATIC-Zeiten		
S_IMPULS: Zeit als Impuls parametrieren und starten (Seite 4614)	x	
S_VIMP: Zeit als verlängerten Impuls parametrieren und starten (Seite 4617)	x	
S_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten (Seite 4620)	x	
S_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten (Seite 4623)	x	
S_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten (Seite 4625)	x	
Zähler		
IEC-Zähler		
CTU: Vorwärts zählen (Seite 4638)	x	
CTD: Rückwärts zählen (Seite 4640)	x	
CTUD: Vorwärts und rückwärts zählen (Seite 4642)	x	
SIMATIC-Zähler		
Z_VORW: Parametrieren und vorwärts zählen (Seite 4645)	x	
Z_RUECK: Parametrieren und rückwärts zählen (Seite 4647)	x	
ZAEHLER: Parametrieren und vorwärts/rückwärts zählen (Seite 4648)	x	
Vergleicher		
CMP==: Gleich (Seite 4654)	x	x
CMP<>: Ungleich (Seite 4655)	x	x
CMP>=: Größer gleich (Seite 4656)	x	x
CMP<=: Kleiner gleich (Seite 4657)	x	x
CMP>: Größer (Seite 4659)	x	x
CMP<: Kleiner (Seite 4660)	x	x
CMP>T: Größer Schrittaktivierungszeit (Seite 5401)		x
CMP>U: Größer ununterbrochene Schrittaktivierungszeit (Seite 5402)		x
Mathematische Funktionen		
ADD: Addieren (Seite 4661)	x	
SUB: Subtrahieren (Seite 4662)	x	
MUL: Multiplizieren (Seite 4664)	x	
DIV: Dividieren (Seite 4665)	x	
MOD: Divisionsrest gewinnen (Seite 4666)	x	
NEG: Zweierkomplement erzeugen (Seite 4667)	x	
ABS: Absolutwert bilden (Seite 4669)	x	
MIN: Minimum ermitteln (Seite 4670)	x	
MAX: Maximum ermitteln (Seite 4672)	x	
LIMIT: Limitieren (Seite 4674)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
SQR: Quadrat bilden (Seite 4676)	x	
SQRT: Quadratwurzel bilden (Seite 4677)	x	
LN: Natürlichen Logarithmus bilden (Seite 4678)	x	
EXP: Exponentialwert bilden (Seite 4680)	x	
SIN: Sinuswert bilden (Seite 4681)	x	
COS: Cosinuswert bilden (Seite 4682)	x	
TAN: Tangenswert bilden (Seite 4684)	x	
ASIN: Arcussinuswert bilden (Seite 4685)	x	
ACOS: Arcuscosinuswert bilden (Seite 4687)	x	
ATAN: Arcustangenswert bilden (Seite 4688)	x	
Verschieben		
MOVE: Wert kopieren (Seite 4690)	x	
BLKMOV: Bereich kopieren (Seite 4692)	x	
UBLKMOV: Bereich ununterbrechbar kopieren (Seite 4696)	x	
FILL: Bereich befüllen (Seite 4699)	x	
Umwandler		
CONVERT: Wert konvertieren (Seite 4702)	x	
ROUND: Zahl runden (Seite 4703)	x	
CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen (Seite 4705)	x	
FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen (Seite 4706)	x	
TRUNC: Ganzzahl erzeugen (Seite 4707)	x	
SCALE: Skalieren (Seite 4709)	x	
UNSCALE: Deskalieren (Seite 4711)	x	
Programmsteuerung		
Laufzeitsteuerung		
RE_TRIGR: Zykluszeitüberwachungszeit neu starten (Seite 4721)	x	
STP: Programm beenden (Seite 4721)	x	
PROTECT: Schutzstufe ändern (Seite 4723)	x	
WAIT: Zeitverzögerung programmieren (Seite 4722)	x	
COMPRESS: CPU-Speicher komprimieren (S7-400) (Seite 4719)	x	
CiR: CiR-Vorgang steuern (S7-400) (Seite 4726)	x	
Wortverknüpfungen		
AND: UND verknüpfen (Seite 4728)	x	
OR: ODER verknüpfen (Seite 4730)	x	
XOR: EXKLUSIV ODER verknüpfen (Seite 4731)	x	
INVERT: Einerkomplement erzeugen (Seite 4733)	x	
DECO: Decodieren (Seite 4734)	x	
ENCO: Encodieren (Seite 4735)	x	
SEL: Selektieren (Seite 4737)	x	
Schieben und Rotieren		
SHR: Rechts schieben (Seite 4738)	x	

FUP-Anweisung	Permanente Operation	Transition/ Supervision/ Interlock
SHL: Links schieben (Seite 4740)	x	
ROR: Rechts rotieren (Seite 4742)	x	
ROL: Links rotieren (Seite 4744)	x	
Weitere Anweisungen		
SET: Bereich bitweise setzen (Seite 4758)	x	
SETP: Bitfeld im Peripheriebereich setzen (S7-400) (Seite 4759)	x	
SETI: Bereich byteweise setzen (Seite 4762)	x	
RESET: Bereich bitweise rücksetzen (Seite 4764)	x	
RESETP: Bitfeld im Peripheriebereich rücksetzen (S7-400) (Seite 4765)	x	
RESETI: Bereich byteweise rücksetzen (Seite 4767)	x	
REPL_VAL: Ersatzwert eintragen (Seite 4769)	x	
DRUM: Schrittschaltwerk realisieren (Seite 4770)	x	
DRUM_X: Schrittschaltwerk realisieren (Seite 4776)	x	
DCAT: Diskreter Steuerungszeitalarm (Seite 4785)	x	
MCAT: Motorsteuerungszeitalarm (Seite 4788)	x	
IMC: Eingabebits mit den Bits der Maske vergleichen (Seite 4792)	x	
SMC: Matrixscanner (Seite 4795)	x	
LEAD_LAG: Lead- und Lag-Algorithmus (Seite 4797)	x	
TONR_X: Zeit akkumulieren (Seite 4800)	x	
WSR: Datum im Schieberegister speichern (Seite 4803)	x	
SHRB: Bit in Bitschieberegister schieben (Seite 4806)	x	
SEG: Bitmuster für 7-Segment-Anzeige erzeugen (Seite 4810)	x	
BCDCPL: Zehnerkomplement erzeugen (Seite 4811)	x	
BITSUM: Anzahl der gesetzten Bits zählen (Seite 4813)	x	

Siehe auch

JMP: Springen bei VKE = 1 (Seite 4713)

JMPN: Springen bei VKE = 0 (Seite 4714)

LABEL: Sprungmarke (Seite 4715)

CMP>T: Größer Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Gesamtdauer eines Schritts. Dabei werden etwaige Ereignisse oder Fehler zeitlich mit erfasst.

In der programmierten Bedingung wird die laufende oder die letzte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn

die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts
<Operand2>	Input	DINT, TIME, S5TIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

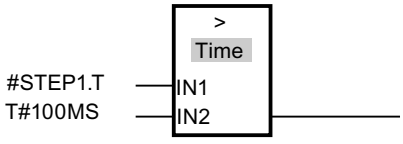

Solange die Aktivierungszeit von #STEP1.T unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

CMP>U: Größer ununterbrochene Schrittaktivierungszeit

Beschreibung

Mit der Anweisung "Größer ununterbrochene Schrittaktivierungszeit" überwachen Sie in Supervisionsbedingungen die Dauer eines Schritts, abzüglich eventueller Störungen. Dabei werden etwaige Ereignisse oder Störungen zeitlich nicht mit erfasst und nur die reine Schrittdauer überwacht.

In der programmierten Bedingung wird die gesamte Aktivierungszeit eines Schritts (Operand1) mit einer definierten Dauer (Operand2) in Millisekunden verglichen. Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer ununterbrochene Schrittaktivierungszeit":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
<Operand1>	Input	Ganzzahlen	E, A, M, D, L oder Konstante	Laufende oder letzte Aktivierungszeit des Schritts abzüglich der Störungen
<Operand2>	Input	DINT, TIME, S5TIME	E, A, M, D, L oder Konstante	Zeitdauer, mit der verglichen wird

Beispiel

Das folgende Beispiel zeigt die Anweisung im Netzwerk:

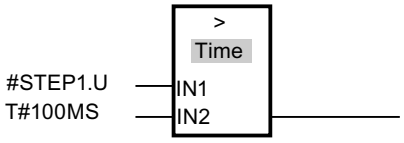

Solange die Aktivierungszeit abzüglich eventueller Störungen von #STEP1.U unter 100 ms liegt, ist das VKE "0". Sobald die Aktivierungszeit 100 ms überschreitet, wechselt das VKE auf "1".

4.2.3 Erweiterte Anweisungen

4.2.3.1 Datum und Uhrzeit

T_COMP: Zeitvariablen vergleichen

Beschreibung

Mit der Anweisung vergleichen Sie die Inhalte zweier Variablen mit dem Datentyp DT und geben das Vergleichsergebnis als Rückgabewert am Parameter OUT aus. Der Parameter OUT wird auf "1" gesetzt, wenn die verwendete Vergleichsbedingung erfüllt ist.

Die folgenden Vergleichsmöglichkeiten können verwendet werden:

Symbol	Beschreibung
EQ	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 gleich ist.
NE	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 ungleich ist.
GE	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt Parameter IN1 größer (jünger) oder gleich dem Zeitpunkt am Parameter IN2 ist.

Symbol	Beschreibung
LE	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 kleiner (älter) oder gleich dem Zeitpunkt am Parameter IN2 ist.
GT	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 größer (jünger) als am Parameter IN2 ist.
LT	Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 kleiner (älter) als am Parameter IN2 ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_COMP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DT	D, L	Zu vergleichender Wert
IN2	Input	DT	D, L	Zu vergleichender Wert
OUT	Output	BOOL	E, A, M, D, L	Rückgabewert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

T_CONV: Zeiten umwandeln und extrahieren

Beschreibung

Mit der Anweisung "T_CONV" konvertieren Sie einen Datentyp DT (DATE_AND_TIME), S5TIME und TIME in einen Datentyp DATE, INT, TOD (TIME_OF_DAY), TIME oder S5TIME.

Parameter

Die Art der Konvertierung bestimmen Sie durch die Auswahl der Datentypen in den Anweisungsboxen des Ein- und Ausgangs der Anweisung. Das Ergebnis der Konvertierung können Sie am Parameter OUT abfragen.

Die folgende Tabelle zeigt die Parameter der Anweisung "T_CONV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DT, S5TIME, TIME	E, A, M, D, L	Zu konvertierender Wert. Das Datenformat der Variable am Parameter IN muss mit dem über die Anweisungsbox ausgewählten Datentyp übereinstimmen.
OUT	Return	DATE, INT, TOD, TIME, S5TIME	E, A, M, D, L	Ergebnis der Konvertierung. Das Datenformat der Variable am Parameter OUT muss mit dem über die Anweisungsbox ausgewählten Datentyp übereinstimmen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiele

Tabelle 4-187 Umwandlung des Datenformats TIME in das Format S5TIME

Parameter	Deklaration	Datentyp	Beschreibung
IN	Output	TIME	Eingangsvariable im Datenformat TIME.
OUT	Input	S5TIME	Die Funktion wandelt das Datenformat TIME in das Format S5TIME. Bei der Wandlung wird abgerundet. Ist der Eingangsparameter größer als das darstellbare S5TIME-Format (größer als TIME#02:46:30.000), wird als Ergebnis S5TIME#999.3 ausgegeben und das Binärergebnis BIE auf "0" gesetzt.

Tabelle 4-188 Extraktion des Wochentags aus dem Format DATE_AND_TIME

Parameter	Deklaration	Datentyp	Beschreibung
IN	Output	DT	Eingangsvariable im Datenformat DATE_AND_TIME.
OUT	Input	INT	Die Funktion extrahiert den Wochentag aus dem Format DATE_AND_TIME (DT). Der Wochentag wird im Datenformat INTEGER ausgegeben: 1: Sonntag 2: Montag 3: Dienstag 4: Mittwoch 5: Donnerstag 6: Freitag 7: Samstag

T_ADD: Zeiten addieren

Beschreibung

Mit Anweisung "T_ADD" addieren Sie auf einen Zeitpunkt (Format DT) eine Zeitdauer (Format TIME) und geben als Ergebnis einen neuen Zeitpunkt (Format DT) aus.

Der Zeitpunkt DT an dem Parameter IN1 muss im Bereich von DT#1990-01-01-00:00:00.000 und DT#2089-12-31-23:59:59.999 liegen. Die Anweisung führt keine Eingangsprüfung durch. Liegt das Ergebnis der Addition nicht im oben angegebenen Bereich, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_ADD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DT	D, L	Zeitpunkt im Format DT (DATE_AND_TIME).
IN2	Input	TIME	E, A, M, D, L oder Konstante	Zu addierende Zeitdauer im Format TIME.
OUT	Return	DT	D, L	Ergebnis der Addition im Format DT (DATE_AND_TIME).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

T_SUB: Zeiten subtrahieren

Beschreibung

Mit der Anweisung "T_SUB" subtrahieren Sie von einem Zeitpunkt (Format DT) eine Zeitdauer (Format TIME) und geben als Ergebnis einen neuen Zeitpunkt (Format DT) aus.

Der Zeitpunkt DT an dem Parameter IN1 muss im Bereich von DT#1990-01-01-00:00:00.000 und DT#2089-12-31-23:59:59.999 liegen. Die Anweisung führt keine Eingangsprüfung durch. Liegt das Ergebnis der Subtraktion nicht im oben angegebenen Bereich, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_SUB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DT	D, L	Zeitpunkt im Format DT (DATE_AND_TIME).
IN2	Input	TIME	E, A, M, D, L oder Konstante	Zu subtrahierende Zeitdauer im Format TIME.
OUT	Return	DT	D, L	Ergebnis der Subtraktion im Format DT (DATE_AND_TIME).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

T_DIFF: Zeitdifferenz

Beschreibung

Mit der Anweisung "T_DIFF" ermitteln Sie die Differenz zwischen zwei Zeitpunkten, indem Sie zwei Zeitpunkte des Formats DT subtrahieren. Der Zeitpunkt am Eingang IN2 wird von dem Zeitpunkt am Eingang IN1 subtrahiert. Das Ergebnis der Subtraktion wird am Ausgang OUT im Format TIME ausgegeben:

- Ist der erste Zeitpunkt am Eingang IN1 größer (jünger) als der zweite Zeitpunkt am Eingang IN2, ist das Ergebnis am Ausgang OUT positiv.
- ist der erste Zeitpunkt am Eingang IN1 kleiner (älter) als der zweite Zeitpunkt am Eingang IN2, ist das Ergebnis am Ausgang OUT negativ.

Liegt das Ergebnis der Subtraktion außerhalb des TIME-Zahlenbereichs, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_DIFF":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DT	D, L	Minuend
IN2	Input	DT	D, L	Subtrahend
OUT	Return	TIME	E, A, M, D, L	Differenz im Format TIME

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

T_COMBINE: Zeiten zusammenfassen

Beschreibung

Die Anweisung "T_COMBINE" fasst die Datenformate DATE und TIME_OF_DAY (TOD) zusammen und wandelt diese Formate in das Datenformat DATE_AND_TIME (DT). Die Anweisung meldet keine Fehler.

- Der Eingangswert DATE muss zwischen DATE#1990-01-01 und DATE#2089-12-31 liegen (wird nicht geprüft).
- Am Eingangswert IN2 wird der Datentyp TIME_OF_DAY verwendet.
- Am Ausgangswert OUT wird das Datenformat DATE_AND_TIME ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T_COMBINE":

odParameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	DATE	D, L oder Konstante	Eingangsvariable im Format DATE
IN2	Input	TOD	D, L oder Konstante	Eingangsvariable im Format TOD
OUT	Return	DT	D, L	Rückgabewert im Format DT

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Uhrzeitfunktionen

WR_SYS_T: Uhrzeit stellen

Beschreibung

Mit dem Aufruf von "WR_SYS_T" stellen Sie die Uhrzeit und das Datum der CPU-Uhr. Die Uhr läuft dann ab der eingestellten Uhrzeit und dem eingestellten Datum. Die Angaben zu Datum und Uhrzeit geben Sie im Datenformat DT am Eingang IN der Anweisung an. Zur Umwandlung der Formate DATE und TOD in das benötigte Datenformat DT können Sie die Anweisung "T_COMBINE (Seite 5407)" verwenden.

Ist die Uhr eine Master-Uhr, dann startet die CPU beim Aufruf zusätzlich die Synchronisation der Uhrzeit. Die Synchronisationsintervalle stellen Sie per Projektierung ein. Angaben zur lokalen Zeitzone oder Sommerzeit können mit der Anweisung "WR_SYS_T" nicht übertragen werden.

Am Ausgang RET_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_SYS_T":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	DT	D, L	Datum und Uhrzeit
RET_VAL	Return	INT	E, A, M, D, L	Status der Anweisung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#....)	Beschreibung
0000	Kein Fehler
8080	Fehler im Datum
8081	Jahr ungültig
8082	Monat ungültig
8083	Tag ungültig
8084	Angabe der Stunden ungültig
8085	Angabe der Minuten ungültig
8086	Angabe der Sekunden ungültig
8087	Angabe der Nanosekunden ungültig
80B0	Die Echtzeituhr ist ausgefallen

Beispiel

Das Datum und die Uhrzeit geben Sie als Datentyp DT ein. Beispiel für den 15. Januar 2010, 10:30 Uhr und 30 Sekunden: DT#2010-01-15-10:30:30. Die Eingabe der Uhrzeit ist nur sekundengranular möglich. Der Wochentag wird von "WR_SYS_T" aus dem Datum errechnet.

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

RD_SYS_T: Uhrzeit lesen

Beschreibung

Mit der Anweisung "RD_SYS_T" lesen Sie das aktuelle Datum und die aktuelle Uhrzeit der CPU-Uhr aus. Die ausgelesenen Daten werden im Format DT am Ausgang OUT der Anweisung ausgegeben. Im ausgegebenen Wert sind dabei keine Angaben zur lokalen Zeitzone oder Sommerzeit enthalten. Am Ausgang RET_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_SYS_T":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
OUT	Output	DT	D,L	Datum und Uhrzeit der CPU

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#....)	Beschreibung
0000	Kein Fehler
8222	Das Ergebnis liegt außerhalb des zulässigen Wertebereichs.
8223	Das Ergebnis kann nicht im angegebenen Datentyp gespeichert werden.

Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

TIME_TCK: Systemzeit lesen

Beschreibung

Mit der Anweisung "TIME_TCK" lesen Sie die Systemzeit der CPU. Die Systemzeit ist ein Zeitzähler, der von 0 bis max. 2147483647 ms zählt. Bei einem Überlauf der Systemzeit wird wieder ab "0" gezählt. Das Zeitraster und die Genauigkeit der Systemzeit beträgt 1 ms. Die Systemzeit wird nur von den Betriebszuständen der CPU beeinflusst. Die Systemzeit können Sie zum Beispiel nutzen, um die Dauer von Vorgängen durch Differenzbildung der Ergebnisse von zwei "TIME_TCK"-Aufrufen zu messen. Die Anweisung stellt keine Fehlerinformationen zur Verfügung.

Die folgende Tabelle gibt eine Übersicht, wie sich die Systemzeit abhängig von den Betriebszuständen der CPU ändert.

Betriebzustand	Systemzeit ...
Anlauf	... wird ständig aktualisiert.
RUN	
STOP	... wird angehalten und behält den aktuellen Wert.
Wiederanlauf (nicht bei S7-300)	... läuft mit dem Wert weiter, der beim Übergang in STOP gespeichert wurde.
Neustart (Warmstart)	... wird gelöscht und beginnt wieder von "0" an zu laufen.
Kaltstart	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TIME_TCK":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	TIME	E, A, M, D, L	Der Parameter RET_VAL enthält die gelesene Systemzeit im Bereich von 0 bis 2^{31} -1 ms.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

SET_CLKS: Uhrzeit stellen und Uhrzeitstatus setzen

Beschreibung

Mit der Anweisung "SET_CLKS" stellen Sie die Uhrzeit Ihrer CPU und setzen den Uhrzeitstatus.

Hinweis

Setzen Sie "SET_CLKS" nur dann ein, wenn bei Ihrer CPU die Uhrzeit nicht synchronisiert wird. Andernfalls wird bei jeder Synchronisation der Uhrzeitstatus der Master-Uhr übernommen. Ein mittels "SET_CLKS" vorgegebener Wert wird dadurch überschrieben.

Hinweis

Den aktuellen Uhrzeitstatus Ihrer CPU können Sie durch Auslesen der SZL-ID W#16#0132 Index W#16#0008 (Seite 5713) mit der Anweisung "RDSYSST (Seite 5683)" ermitteln.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_CLKS":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Betriebsart: • B#16#01: Uhrzeit stellen • B#16#02: Uhrzeitstatus setzen • B#16#03: Uhrzeit stellen und Uhrzeitstatus setzen
PDT	Input	DT	D, L	vorgegebene Uhrzeit
CORR	Input	INT	E, A, M, D, L oder Konstante	Korrekturwert (im 0,5 h-Raster) Mögliche Werte: –24 bis +26
SUMMER	Input	BOOL	E, A, M, D, L	Sommer-/Winterzeitkennung: • 0 = Winterzeit • 1 = Sommerzeit

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ANN_1	Input	BOOL	E, A, M, D, L	Ankündigungsstunde: 1: Beim nächsten Stundenwechsel findet eine Umschaltung von Sommer- nach Winterzeit oder umgekehrt statt.
RET_VAL	Return	INT	E, A, M, D, L	Fehlercode

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

Über den Parameter MODE können Sie vorgeben, ob Sie nur die Uhrzeit, nur den Uhrzeitstatus oder beide Werte ändern wollen. Dies ist in folgender Tabelle erläutert:

MODE	Bedeutung
B#16#01	Uhrzeit stellen Der Aufruf entspricht dem Aufruf der Anweisung "SET_CLK". Die Eingangsparameter CORR, SUMMER und ANN_1 werden nicht ausgewertet.
B#16#02	Uhrzeitstatus setzen Der Eingangsparameter PDT wird nicht ausgewertet. Aus den übrigen Eingangsparametern werden folgende Elemente des Uhrzeitstatus gebildet: • Korrekturwert einschließlich Vorzeichen • Ankündigungsstunde • Sommer-/Winterzeitindikator Die Uhrzeitauflösung wird passend zur Uhrzeitauflösung Ihrer CPU gesetzt. Das Bit Synchronisationsausfall des Uhrzeitstatus wird mit FALSE beschrieben. Die Uhrzeit bleibt unverändert.
B#16#03	Uhrzeit stellen und Uhrzeitstatus setzen

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	kein Fehler
8080	MODE außerhalb des zulässigen Wertebereichs
8081	CORR außerhalb des zulässigen Wertebereichs (nur bei MODE = B#16#02 oder bei MODE = B#16#03)
8082	PDT außerhalb des zulässigen Wertebereichs: Datum und/oder Uhrzeit unzulässig
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

SNC_RTCB: Uhrzeitslaves synchronisieren

Definition: Synchronisation von Uhrzeitslaves

Unter der Synchronisation von Uhrzeitslaves versteht man die Übertragung des Datums und der Uhrzeit vom Uhrzeitmaster eines Bussegments (z. B. S7-400-K-Bus, MPI, S7-Rückwandbus) auf sämtliche Uhrzeitslaves eben dieses Bussegments.

Beschreibung

Um sicherzustellen, dass die Uhrzeit aller Baugruppen im Netz übereinstimmt, werden die Uhrzeit-Slaves vom Systemprogramm in regelmäßigen, parametrierbaren Abständen synchronisiert.

Eine Synchronisation kann unabhängig vom parametrierten Synchronisationsintervall durch einen Aufruf der Anweisung "SNC_RTCB" durchgeführt werden. Dabei werden für alle an einem Bussegment vorhandenen Uhrzeitslaves das Datum und Uhrzeit vom Uhrzeit-Master an den Uhrzeit-Slave übertragen.

Voraussetzung für die erfolgreiche Synchronisation ist, dass "SNC_RTCB" auf einer CPU aufgerufen wird, deren Echtzeituhr für mindestens ein Bussegment als Uhrzeitmaster parametriert wurde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SNC_RTCB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Bei der Synchronisation ist kein Fehler aufgetreten.
0001	Die vorhandene Uhr wurde für kein Bussegment als Uhrzeitmaster parametriert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Zusätzliche Standardbibliotheken

Hantierung des Betriebsstundenzählers

Betriebsstundenzähler

Einleitung

Die CPUs verfügen über eine spezifische Anzahl von Betriebsstundenzählern (siehe Technische Daten der CPUs).

- Falls Ihre CPU 16-Bit-Betriebsstundenzähler hat, können Sie mit den Anweisungen "SET_RTM (Seite 5416)", "CTRL_RTM (Seite 5417)" und "READ_RTM (Seite 5418)" die Betriebsstundenzähler setzen, starten, stoppen und auslesen.
- Falls Ihre CPU 32-Bit-Betriebsstundenzähler hat, können Sie mit der Anweisung "RTM (Seite 5415)" die Betriebsstundenzähler setzen, starten, stoppen und auslesen.

Hinweis

Die Anweisungen "SET_RTM (Seite 5416)", "CTRL_RTM (Seite 5417)" und "READ_RTM (Seite 5418)" können Sie auch für die 32-Bit-Betriebsstundenzähler verwenden. In diesem Fall verhalten sich die Betriebsstundenzähler jedoch wie 16-Bit-Betriebsstundenzähler (Wertebereich 0 bis 32767 Stunden).

Siehe auch: Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#000B (Seite 5715).

Anwendung

Einen Betriebsstundenzähler können Sie für die unterschiedlichsten Anwendungen nutzen:

- Betriebsdauer der CPU berechnen
- Betriebsdauer von angesteuerten Betriebsmitteln berechnen.

Eigenschaften des Betriebsstundenzählers

Mit dem Start beginnt der Betriebsstundenzähler immer ab dem letzten Zählerstand zu zählen. Soll er ab einem anderen Anfangswert beginnen, dann müssen Sie diesen explizit zuweisen ("SET_RTM (Seite 5416)" bzw. "RTM (Seite 5415)" mit MODE=4). Geht die CPU in STOP oder Sie stoppen den Betriebsstundenzähler, dann "merkt" sich die CPU den aktuellen Wert des Betriebsstundenzählers. Bei Neustart (Warmstart) oder Kaltstart der CPU muss der Betriebsstundenzähler erneut gestartet werden ("CTRL_RTM (Seite 5417)" bzw. "RTM (Seite 5415)" mit MODE=1).

Nach einem Update des Betriebssystems sowie nach Rücksetzen der CPU in den Auslieferungszustand sind die Betriebsstundenzähler auf den Wert "0" zurückgesetzt.

Wertebereich

- CPU mit 16-Bit-Betriebsstundenzähler: 0 bis 32 767 Stunden
- CPU mit 32-Bit-Betriebsstundenzähler: 0 bis (2 hoch 31) -1 Stunden = 2 147 483 647 Stunden

RTM: Betriebsstundenzähler

Beschreibung

Mit der Anweisung "RTM" können Sie einen 32-Bit-Betriebsstundenzähler Ihrer CPU setzen, starten, stoppen und auslesen.

Falls Sie alle 32-Bit-Betriebsstundenzähler Ihrer CPU auslesen möchten, verwenden Sie die Anweisung "RDSYSST (Seite 5683)" mit den Parameter SZL_ID=W#16#0132 und:

- INDEX=W#16#000B (für die Betriebsstundenzähler 0 bis 7) bzw.
- INDEX=W#16#000C (für die Betriebsstundenzähler 8 bis 15).

Siehe auch: Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#000B (Seite 5715).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RTM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NR	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des Betriebsstundenzählers Die Nummerierung beginnt mit 0. Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den Technischen Daten.
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung: • 0: auslesen (Status wird dann in CQ, aktueller Wert wird in CV eingetragen). Läuft der Betriebsstundenzähler länger als (2^31) - 1 Stunden, dann bleibt er beim letzten darstellbaren Wert stehen und liefert die Fehlermeldung "Überlauf". • 1: starten (mit dem letzten Zählerstand) • 2: stoppen • 4: setzen (auf den in PV angegebenen Wert) • 5: setzen (auf den in PV angegebenen Wert) und starten • 6: setzen (auf den in PV angegebenen Wert) und stoppen
PV	Input	DINT	E, A, M, D, L oder Konstante	Neuer Wert für den Betriebsstundenzähler

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
CQ	Output	BOOL	E, A, M, D, L	Status des Betriebsstundenzählers (1: läuft)
CV	Output	DINT	E, A, M, D, L	Aktueller Wert des Betriebsstundenzählers

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Kompatibilität zu Programmen, die für eine CPU mit 16-Bit-Betriebsstundenzählern entwickelt wurden

Sie können die 32-Bit-Betriebsstundenzähler auch mit den Anweisungen "SET_RTM (Seite 5416)", "CTRL_RTM (Seite 5417)" und "READ_RTM (Seite 5418)". In diesem Fall verhalten sich die 32-Bit-Betriebsstundenzähler jedoch wie 16-Bit-Betriebsstundenzähler (Wertebereich 0 bis 32767 Stunden).

Im Teillistenauszug mit der SZL-ID W#16#0132 und dem Index W#16#0008 werden Ihnen die 32-Bit-Betriebsstundenzähler 0 bis 7 als 16-Bit-Betriebsstundenzähler angezeigt. Damit können Sie Programme, die für eine CPU mit 16-Bit-Betriebsstundenzählern entwickelt wurden, und die den Teillistenauszug mit der SZL-ID W#16#0132 und dem Index W#16#0008 benutzen, weiterhin einsetzen.

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Falsche Nummer des Betriebsstundenzählers
8081	Ein negativer Wert wurde dem Parameter PV übergeben.
8082	Überlauf des Betriebsstundenzählers
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

SET_RTM: Betriebsstundenzähler setzen

Beschreibung

Mit der Anweisung stellen Sie einen Betriebsstundenzähler der CPU auf einen vorgegebenen Wert. Sie können eine CPU-spezifische Anzahl von Betriebsstundenzählern einstellen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_RTM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NR	Input	BYTE	E, A, M, D, L oder Konstante	Eingang NR enthält die Nummer des Betriebsstundenzählers, den Sie einstellen möchten. Die Nummerierung beginnt mit "0". Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den Technischen Daten.
PV	Input	INT	E, A, M, D, L oder Konstante	Eingang PV enthält die Einstellung für den Betriebsstundenzähler.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Falsche Nummer des Betriebsstundenzählers
8081	Ein negativer Wert wurde dem Parameter PV übergeben.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CTRL_RTM: Betriebsstundenzähler starten und stoppen

Beschreibung

Mit der Anweisung starten oder stoppen Sie einen Betriebsstundenzähler.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CTRL_RTM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NR	Input	BYTE	E, A, M, D, L oder Konstante	Eingang NR enthält die Nummer des Betriebsstundenzählers, den Sie starten bzw. stoppen möchten. Die Nummerierung beginnt mit "0". Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den Technischen Daten.
S	Input	BOOL	E, A, M, D, L	Eingang S startet bzw. stoppt den Betriebsstundenzähler. • Setzen Sie den Signalzustand auf "0", wenn Sie den Betriebsstundenzähler stoppen möchten. • Setzen Sie den Signalzustand auf "1", wenn Sie den Betriebsstundenzähler starten möchten.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Falsche Nummer des Betriebsstundenzählers
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

READ_RTM: Betriebsstundenzähler lesen

Beschreibung

Mit der Anweisung lesen Sie einen Betriebsstundenzähler aus. Die Anweisung liefert als Ausgangsdaten die aktuelle Betriebsstundenzahl und den Status des Zählers, d. h. "gestoppt" oder "zählt".

Läuft der Betriebsstundenzähler länger als 32767 Stunden, dann bleibt er bei 32767 stehen und liefert die Fehlermeldung "Überlauf".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_RTM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NR	Input	BYTE	E, A, M, D, L, P oder Konstante	Eingang NR enthält die Nummer des Betriebsstundenzählers, den Sie starten bzw. stoppen möchten. Die Nummerierung beginnt mit "0". Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den Technischen Daten.
RET_VAL	Return	INT	E, A, M, D, L, P	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
CQ	Output	BOOL	E, A, M, D, L	Ausgang CQ gibt an, ob der Betriebsstundenzähler läuft oder angehalten ist. Der Signalzustand "0" zeigt an, dass der Betriebsstundenzähler gestoppt ist. Der Signalzustand "1" zeigt an, dass der Betriebsstundenzähler läuft.
CV	Output	INT	E, A, M, D, L, P	Ausgang CV gibt den aktuellen Wert des Betriebsstundenzählers an.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Kein Fehler
8080	Falsche Nummer des Betriebsstundenzählers
8081	Überlauf des Betriebsstundenzählers
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Lokalzeit

LOC_TIME: Lokalzeit errechnen

Beschreibung

Die Anweisung liest Uhrzeitstatus und Uhrzeit der CPU und errechnet daraus die Lokalzeit. Sie funktioniert nur bei S7-400-CPUs. Die Anweisung kann in OBs jeder Prioritätsklasse aufgerufen werden.

Arbeitsweise

"LOC_TIME" verwendet intern die Anweisung "RDSYSST (Seite 5683)", um die Baugruppenzeit, den Korrekturwert mit Vorzeichen sowie die Sommerzeit-Kennung aus dem aktuellen Uhrzeitstatus der CPU auszulesen. Fehler von "RDSYSST (Seite 5683)" beenden den Baustein. Die Fehleranzeigen werden an RET_VAL übergeben.

Der Korrekturwert ist die Anzahl der halben Stunden zwischen Basis-Uhrzeit und Lokalzeit. Zusätzlich ist er als Sommer- oder Winterzeit gekennzeichnet.

Wenn die CPU keinen gültigen Uhrzeitstatus hat, wird "LOC_TIME" mit Fehler beendet.

Nach erfolgreichem Lesen des Uhrzeitstatus wird der aus dem Korrekturwert ermittelte Zeitunterschied zur Baugruppenzeit addiert und am Ausgang LT ausgegeben.

Der Ausgang SUMMER übernimmt den Wert des Sommer-/Winterzeit-Bits aus dem Uhrzeitstatus.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOC_TIME":

Parameter	Deklaration	Datentyp	Beschreibung
RET_VAL	Return	INT	Fehlercode
LT	Output	DATE_AND_TIME	Lokalzeit (Local Time)
SUMMER	Output	BOOL	Sommerzeitkennung • 0: Winter • 1: Sommer

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL, LT und SUMMER

Die folgende Tabelle zeigt die Ausgabewerte und die Fehler von "LOC_TIME":

RET_VAL	LT	SUMMER	Beschreibung
0	Lokalzeit	TRUE/FALSE	Anweisung fehlerfrei gelaufen
1	Lokalzeit	TRUE/FALSE	Kein Fehler, aber Datumssprung
8xyy	DT#90-01-01-0:0:0	FALSE	Allgemeine Fehlercodes von "RDSYSST (Seite 5683)"
8091	DT#90-01-01-0:0:0	FALSE	Uhrzeitstatus nicht vorhanden

BT_LT: Lokalzeit aus Basiszeit berechnen

Beschreibung

Die Anweisung errechnet die Lokalzeit zu der Basiszeit, die am Eingang vorgegeben wurde.

Die am Eingang BT eingegebene Basiszeit wird mithilfe der Daten, die in einem Datenbaustein (DB) hinterlegt sind, in die Lokalzeit umgerechnet und am Ausgang LT ausgegeben.

Der Datenbaustein (DB) enthält die Anzahl der 30 Minuten-Einheiten, durch die sich Basis- und Lokalzeit unterscheiden sowie die Differenz zwischen Sommer- und Winterzeit, ebenfalls in Einheiten von 30 Minuten.

Wenn sich bei der Berechnung ein Datumsüberlauf ergibt, wird dies durch einen speziellen Rückgabewert gekennzeichnet.

Die Anweisung "BT_LT" kann in OBs jeder Prioritätsklasse aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BT_LT":

Parameter	Deklaration	Datentyp	Beschreibung
BT	Input	DATE_AND_TIME	Basiszeit
WS_DAT	Input	BLOCK_DB	Information zur Zeitzone sowie zur Sommer-/Winterzeit -Umschaltung (Regel-DB); Verwenden Sie am Parameter WS_DAT einen Zeiger auf einen Datenbaustein vom Typ "WS_RULES (Seite 5435)".
RET_VAL	Return	INT	Fehlercode
LT	Output	DATE_AND_TIME	Lokalzeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter vom Datentyp BLOCK_DB können nur an FB-Aufrufe weitergereicht werden, nicht aber an FC-Aufrufe.

Parameter RET_VAL und LT

Die folgende Tabelle zeigt die Ausgabewerte von "BT_LT":

RET_VAL	LT	Beschreibung
0	Lokalzeit	Anweisung fehlerfrei gelaufen
1	Lokalzeit	Kein Fehler, aber Datumssprung
8082	DT#90-01-01-0:0:0	Ungültige Daten im Regel-Datenbaustein

LT_BT: Basiszeit aus Lokalzeit berechnen

Beschreibung

Die Anweisung errechnet die Basiszeit zu der Lokalzeit, die am Eingang vorgegeben wurde.

Die am Eingang LT eingegebene Lokalzeit wird mithilfe der Daten, die in einem Datenbaustein (DB) hinterlegt wurden, in die Basiszeit umgerechnet und am Ausgang BT ausgegeben.

Der Datenbaustein (DB) enthält die Anzahl der 30 Minuten-Einheiten, durch die sich Basis- und Lokalzeit unterscheiden sowie die Differenz zwischen Sommer- und Winterzeit, ebenfalls in Einheiten von 30 Minuten. Wenn sich bei der Berechnung ein Datumsüberlauf ergibt, wird dies durch einen speziellen Rückgabewert gekennzeichnet.

Besonderheiten beim Wechsel zwischen Sommer- und Winterzeit:

- Bei der Umschaltung von Winter- nach Sommerzeit wird die Lokalzeit um eine Stunde vorgestellt. Das bedeutet, dass die dazwischen liegende Stunde nicht durchlaufen wird. Für einen Zeitpunkt LT innerhalb dieser Stunde wird von der "LT_BT" in Sommerzeit "gedacht". Dies wird mit dem Rückgabewert 4 bzw. 5 gemeldet.
- Bei der Umschaltung von Sommer- nach Winterzeit wird die Lokalzeit um eine Stunde zurückgestellt. Das bedeutet, dass diese eine Stunde zweimal durchlaufen wird ("doppelte Stunde"). (Für ME(S)Z gilt dafür die Bezeichnung 2A und 2B).
  Für einen Zeitpunkt LT innerhalb dieser Stunde ist also eine eindeutige Abbildung auf eine Basiszeit nicht möglich.
  "LT_BT" erhält als Eingangsparameter eine LT und muss vor der Umrechnung in BT entscheiden, ob der Wert im Sommer oder im Winter liegt. Für den Fall, dass die LT innerhalb der doppelten Stunde liegt, wird die LT als Winterzeit interpretiert. Dies wird mit dem Rückgabewert 2 bzw. 3 gemeldet.

Die Anweisung "LT_BT" kann in OBs jeder Prioritätsklasse aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LT_BT":

Parameter	Deklaration	Datentyp	Beschreibung
LT	Input	DATE_AND_TIME	Lokalzeit
WS_DAT	Input	BLOCK_DB	Information zur Zeitzone sowie zur Sommer-/Winterzeit -Umschaltung (Regel-DB); Verwenden Sie am Parameter WS_DAT einen Zeiger auf einen Datenbaustein vom Typ "WS_RULES (Seite 5435)".
RET_VAL	Return	INT	Fehlercode
BT	Output	DATE_AND_TIME	Basiszeit

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter vom Datentyp BLOCK_DB können nur an FB-Aufrufe weitergereicht werden, nicht aber an FC-Aufrufe.

Parameter RET_VAL und BT

Die folgende Tabelle zeigt die Ausgabewerte von "LT_BT":

RET_VAL	BT	Beschreibung
0	Basiszeit	Anweisung fehlerfrei gelaufen
1	Basiszeit	Kein Fehler, aber Datumssprung
2	Basiszeit	Die LT am Eingang liegt innerhalb der "doppelten" Stunde.
3	Basiszeit	Wie 2, zusätzlich Datumssprung
4	Basiszeit	Die LT am Eingang liegt innerhalb der "verbotenen" Stunde.
5	Basiszeit	Wie 4, zusätzlich Datumssprung
8082	DT#90-01-01-0:0:0	Ungültige Daten im Regel-Datenbaustein

S_LTINT: Uhrzeitalarm Lokalzeit

Beschreibung

Die Anweisung stellt den gewünschten Uhrzeitalarm zum vorgegebenen Zeitpunkt. Dieser Zeitpunkt wird in Lokalzeit angegeben.

Die am Eingang LT eingegebene Lokalzeit wird mithilfe der Regel, die in einem Datenbaustein (DB) hinterlegt ist, in die Basiszeit umgerechnet.

Der Datenbaustein (DB) enthält die Anzahl der 30 Minuten-Einheiten, durch die sich Basis- und Lokalzeit unterscheiden sowie die Differenz zwischen Sommer- und Winterzeit, ebenfalls in Einheiten von 30 Minuten. Mit der errechneten Basiszeit wird der angegebene Uhrzeit-Alarm-OB parametriert und aktiviert. Wenn sich bei der Berechnung ein Datumsüberlauf ergibt, wird dies durch einen speziellen Rückgabewert gekennzeichnet.

Besonderheiten beim Wechsel zwischen Sommer- und Winterzeit:

- Bei der Umschaltung von Winter- nach Sommerzeit wird die Lokalzeit um eine Stunde vorgestellt. Das bedeutet, dass die dazwischen liegende Stunde nicht durchlaufen wird. Für einen Zeitpunkt LT innerhalb dieser Stunde wird von der "S_LTINT" in Sommerzeit gerechnet. Dies wird mit dem Rückgabewert (RET_VAL) 4 bzw. 5 gemeldet.
- Bei der Umschaltung von Sommer- nach Winterzeit wird die Lokalzeit um eine Stunde zurückgestellt. Das bedeutet, dass diese eine Stunde zweimal durchlaufen wird ("doppelte Stunde"). (Für ME(S)Z gilt dafür die Bezeichnung 2A und 2B).
  Für einen Zeitpunkt LT innerhalb dieser Stunde ist also eine eindeutige Abbildung auf eine Basiszeit nicht möglich. "S_LTINT" erhält als Eingangsparameter eine LT und muss vor der Umrechnung in BT entscheiden, ob der Wert im Sommer oder im Winter liegt. Für den Fall, dass die LT innerhalb der doppelten Stunde liegt, wird die LT als Winterzeit interpretiert. Dies wird mit dem Rückgabewert (RET_VAL) 2 bzw. 3 gemeldet.

Die Anweisung "S_LTINT" kann in OBs jeder Prioritätsklasse aufgerufen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S_LTINT":

Parameter	Deklaration	Datentyp	Beschreibung
OB_NR	Input	INT	Nummer des zu startenden OB (zul. 10 – 17)
SDT	Input	DATE_AND_TIME	Startdatum und Uhrzeit in Lokalzeit Siehe auch: "SET_TINT (Seite 5607)"
PERIOD	Input	WORD	Periode vom Ausgangspunkt SDT an: • W#16#0000 = Einmal • W#16#0201 = Minütlich • W#16#0401 = Stündlich • W#16#1001 = Täglich • W#16#1201 = Wöchentlich • W#16#1401 = Monatlich • W#16#1801 = Jährlich • W#16#2001 = Monatsende
WS_DAT	Input	BLOCK_DB	Information zur Zeitzone sowie zur Sommer-/Winterzeit -Umschaltung (Regel-DB); Verwenden Sie am Parameter WS_DAT einen Zeiger auf einen Datenbaustein vom Typ "WS_RULES (Seite 5435)".
RET_VAL	Return	INT	Fehlercode

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter vom Datentyp BLOCK_DB können nur an FB-Aufrufe weitergereicht werden, nicht aber an FC-Aufrufe.

Parameter RET_VAL

RET_VAL	Beschreibung
0	Anweisung fehlerfrei gelaufen
1	Kein Fehler, aber Datumssprung
2	Die LT am Eingang war innerhalb der "doppelten" Stunde.
3	Wie 2, zusätzlich Datumssprung
4	Die LT am Eingang liegt innerhalb der "verbotenen" Stunde.
5	Wie 4, zusätzlich Datumssprung
8082	Ungültige Daten im Regel-Datenbaustein
8090	Fehlerhafter Parameter OB_NR
8091	Fehlerhafter Parameter SDT
8092	Fehlerhafter Parameter PERIOD
80A1	Der eingestellte Startzeitpunkt liegt in der Vergangenheit.

RET_VAL	Beschreibung
80A2	OB ist nicht geladen
80A3	OB kann nicht gestartet werden

SET_SW: Sommer-/Winterzeit setzen ohne Uhrzeitstatus

Beschreibung

Die Anweisung unterstützt die Sommer-/Winterzeit-Umschaltung in CPUs, die nicht über den Uhrzeitstatus verfügen. Zu diesem Zweck verstellt die Anweisung "SET_SW" die CPU-Uhr entsprechend dem aktuellen Zeitpunkt und den Umschaltregeln im Regel-DB.

"SET_SW" wird im Organisationsbaustein OB1 und in einem Uhrzeitalarm-OB einmal mit derselben Instanz aufgerufen. Im Uhrzeitalarm-OB werden die Parameter der Anweisung "SET_SW" nicht beschaltet. Am Eingang WS_DAT wird der Datenbaustein (DB) angegeben, der die Regel für Beginn und Ende der Sommerzeit enthält.

Das folgende Bild zeigt das Zeitdiagramm, auf dem die Funktion von "SET_SW" basiert:

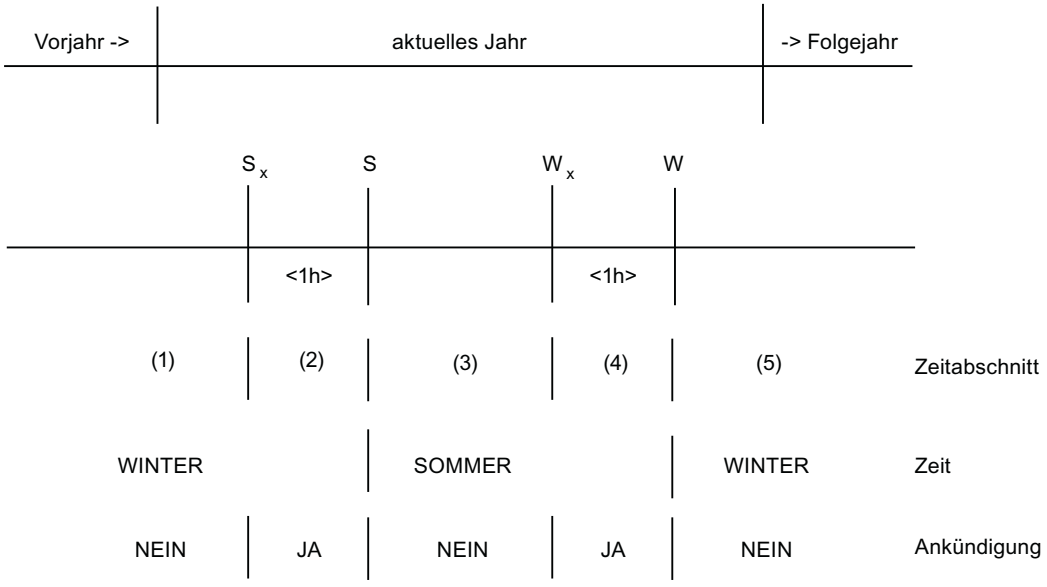

Es sind vier Umschaltzeitpunkte (S_A, S, W_A, W) definiert. Deren genaue Lage leitet sich aus den Regeln im Regel-DB ab. Die Ankündigungs-Stunden liegen eine Stunde vor den Umschaltzeitpunkten und signalisieren das bevorstehende Umschalten.

Bei steigender Flanke am Eingang REQ wird die Sommer-/Winterzeit-Umschaltung initialisiert und aktiviert. Sie bleibt so lange gültig, wie REQ = 1 ist.

Zur Zeitumschaltung parametriert und aktiviert "SET_SW" den angegebenen Uhrzeitalarm-OB mit dem nächstgelegenen Umschaltzeitpunkt im Modus "einmalig". Danach übernimmt dieser Uhrzeitalarm-OB die Kontrolle. Wenn er fällig wird, verändert er die Uhrzeit der CPU entsprechend dem aktuellen Aufrufzeitpunkt und parametriert/aktiviert sich selbst wieder mit dem jeweils nächsten anstehenden Umschaltzeitpunkt.

Liegt die aktuelle Zeit im Abschnitt (3), wird der Uhrzeitalarm-OB mit der Startzeit W_A ("Winterzeit-Ankündigung") parametriert.

Bei fallender Flanke an REQ wird die Sommer-/Winterzeit-Umschaltung deaktiviert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_SW":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerung der Umschaltung
WS_DAT	Input	BLOCK_DB	Information zur Zeitzone sowie zur Sommer-/Winterzeit -Umschaltung (Regel-DB); Verwenden Sie am Parameter WS_DAT einen Zeiger auf einen Datenbaustein vom Typ "WS_RULES (Seite 5435)".
OB_NR	Input	INT	Nummer des zu verwendenden Uhrzeitalarms
STATUS	Output	INT	Fehlercode
ISSUMMER	Output	BOOL	1 = Sommerzeit
ANN_1	Output	BOOL	1 = Ankündigung einer Umschaltung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter vom Datentyp BLOCK_DB können nur an FB-Aufrufe weitergereicht werden, nicht aber an FC-Aufrufe.

Parameter STATUS

STATUS	Bedeutung
0	Kein Fehler
8xyy	Allgemeine Fehlercodes von "RDSYSST (Seite 5683)"
8082	Ungültige Daten in WS_DAT
8083	Die Monatsnummer für die Umschaltung auf Winterzeit ist größer als die Monatsnummer für die Umschaltung auf Sommerzeit, und die Ankündigungsstunde Winter liegt vor dem Sommer, oder die Monatsnummer für die Umschaltung auf Winterzeit ist kleiner als die Monatsnummer für die Umschaltung auf Sommerzeit, und die Ankündigungsstunde Sommer liegt vor dem Winter.
8090	Unzulässige OB_NR. (Nur Uhrzeitalarme zulässig: OB10 – OB17, je nach CPU)
80A2	Unzulässige OB_NR: der OB ist in der CPU nicht geladen
80A3	Der OB kann nicht gestartet werden. Siehe auch: "QRY_TINT (Seite 5610)"

Anwendungsbereich

"SET_SW" kann in CPUs verwendet werden, die nicht über den Uhrzeitstatus verfügen. CPUs mit Uhrzeitstatus verwenden "SET_SW_S (Seite 5427)".

Aufrufende OBs

"SET_SW" muss mit derselben Instanz in folgenden Organisationsbausteinen (OBs) aufgerufen werden:

- **OB1 (zyklisches Programm):**
  "SET_SW" wertet den REQ-Eingang aus. Bei steigender Flanke an REQ stellt er abhängig vom aktuellen Zeitpunkt den Startzeitpunkt des angegebenen Uhrzeitalarms auf die nächstliegende der möglichen 4 Umschaltzeiten ein und aktiviert den Uhrzeitalarm mit der Kennung 'einmalig'. Der Uhrzeitalarm wird ausgelöst, wenn dieser Zeitpunkt erreicht wird.
- **OB1x (Uhrzeitalarm):**
  Der OB1x wurde aktiv, weil ein Umschaltzeitpunkt erreicht wurde. "SET_SW" stellt die Uhrzeit der CPU entsprechend der aktuellen Zeit (Sommerzeit!), parametriert sich auf den nächsten Umschaltzeitpunkt und aktiviert diesen. Im Abschnitt (5) stellt er sich auf die Sommerzeit-Ankündigung (S_A) des Folgejahrs.

"SET_SW" darf nicht mehrfach instanziert werden.

Mit folgenden Anweisungen können Sie den Uhrzeitalarm abfragen, parametrieren, aktivieren und löschen: "QRY_TINT (Seite 5610)", "SET_TINT (Seite 5607)", "ACT_TINT (Seite 5609)", "CAN_TINT (Seite 5608)"

Hinweis

Ein Verstellen der CPU-Uhrzeit, z. B. per PG, bei aktivem "SET_SW" (REQ = 1) ist nur dann ohne besondere Zusatzmaßnahmen erlaubt, wenn die Zeit nicht über einen der Umschaltzeitpunkte hinweg verstellt wird.

Wird beim Stellen der Uhr einer der Umschaltzeitpunkte übersprungen, müssen Sie die Sommer-/Winterzeit-Umschaltung vor dem Verstellen mit REQ = 0 deaktivieren und nach dem Verstellen durch REQ = 1 neu starten.

Beim Stellen der CPU-Uhr müssen Sie natürlich beachten, dass Sie im Sommer die Sommerzeit-Stunde auch einstellen.

SET_SW_S: Sommer-/Winterzeit setzen mit Uhrzeitstatus

Beschreibung

Die Anweisung unterstützt die Sommer-/Winterzeit-Umschaltung in CPUs, die über den Uhrzeitstatus verfügen. Zu diesem Zweck verstellt die Anweisung "SET_SW_S" den Uhrzeitstatus entsprechend dem aktuellen Zeitpunkt und den Umschaltregeln im Regel-DB.

"SET_SW_S" wird im OB1 und in einem Uhrzeitalarm-OB 1x mit derselben Instanz aufgerufen. Im Uhrzeitalarm-OB werden die Parameter der Anweisung "SET_SW_S" nicht beschaltet. Am Eingang WS_DAT wird der Datenbaustein (DB) angegeben, der die Regel für Beginn und Ende der Sommerzeit enthält.

Das folgende Bild zeigt das Zeitdiagramm, auf dem die Funktion von "SET_SW_S" basiert:

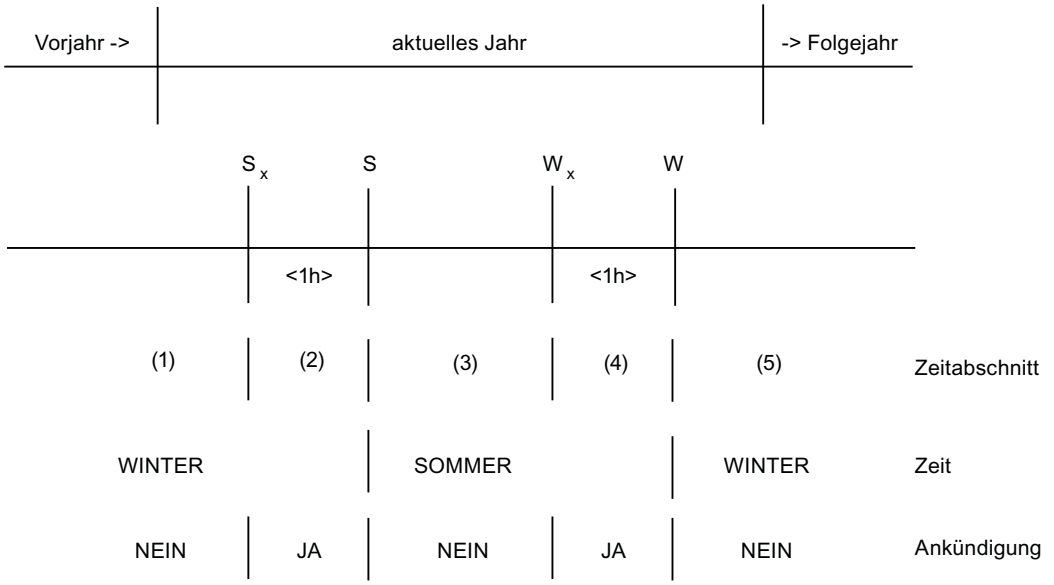

Es sind vier Umschaltzeitpunkte (S_A, S, W_A, W) definiert. Deren genaue Lage leitet sich aus den Regeln im Regel-DB ab. Die Ankündigungs-Stunden liegen 1 Stunde vor den Umschaltzeitpunkten und signalisieren das bevorstehende Umschalten.

Bei steigender Flanke am Eingang REQ wird die Sommer-/Winterzeit-Umschaltung initialisiert und aktiviert. Sie bleibt so lange gültig, wie REQ = 1 ist.

Zur Zeitumschaltung parametriert und aktiviert "SET_SW_S" den angegebenen Uhrzeitalarm-OB mit dem nächstgelegenen Umschaltzeitpunkt im Modus "einmalig". Danach übernimmt dieser Uhrzeitalarm-OB die Kontrolle. Wenn er fällig wird, verändert er die Uhrzeit und den Uhrzeitstatus der CPU entsprechend dem aktuellen Aufrufzeitpunkt und parametriert/aktiviert sich selbst wieder mit dem jeweils nächsten anstehenden Umschaltzeitpunkt.

Beispiel: Liegt die aktuelle Zeit im Bereich (3), wird der Uhrzeitalarm-OB mit der Startzeit WA ("Winterzeit-Ankündigung") parametriert.

Bei fallender Flanke an REQ wird die Sommer-/Winterzeit-Umschaltung deaktiviert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_SW_S":

Parameter	Deklaration	Datentyp	Beschreibung
REQ	Input	BOOL	Steuerung der Umschaltung
WS_DAT	Input	BLOCK_DB	Information zur Zeitzone sowie zur Sommer-/Winterzeit -Umschaltung (Regel-DB); Verwenden Sie am Parameter WS_DAT einen Zeiger auf einen Datenbaustein vom Typ "WS_RULES (Seite 5435)".
OB_NR	Input	INT	Nummer des zu verwendenden Uhrzeitalarms
STATUS	Output	INT	Fehlercode
ISSUMMER	Output	BOOL	Es ist Sommer
ANN_1	Output	BOOL	Ankündigung der Umschaltung

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter vom Datentyp BLOCK_DB können nur an FB-Aufrufe weitergereicht werden, nicht aber an FC-Aufrufe.

Parameter STATUS

STATUS	Bedeutung
0	Kein Fehler
8xyy	Allgemeine Fehlercodes von "RDSYSST (Seite 5683)"
8082	Ungültige Daten in WS_DAT
8083	Die Monatsnummer für die Umschaltung auf Winterzeit ist größer als die Monatsnummer für die Umschaltung auf Sommerzeit, und die Ankündigungsstunde Winter liegt vor dem Sommer, oder die Monatsnummer für die Umschaltung auf Winterzeit ist kleiner als die Monatsnummer für die Umschaltung auf Sommerzeit, und die Ankündigungsstunde Sommer liegt vor dem Winter.
8090	Unzulässige OB_NR. (Nur Uhrzeitalarme zulässig: OB10 – OB17, je nach CPU)
8091	Uhrzeitstatus ungültig ('status_valid' im Uhrzeitstatus stand auf FALSE)
80A2	Unzulässige OB_NR: der OB ist in der CPU nicht geladen
80A3	Der OB kann nicht gestartet werden. (siehe "QRY_TINT (Seite 5610)")

Anwendungsbereich

"SET_SW_S" kann in CPUs verwendet werden, die über den Uhrzeitstatus verfügen.

Aufrufende OBs

"SET_SW_S" muss mit derselben Instanz in folgenden Organisationsbausteinen (OBs) aufgerufen werden:

- **OB1 (zyklisches Programm):**
  "SET_SW_S" wertet den REQ-Eingang aus. Bei steigender Flanke an REQ stellt er abhängig vom aktuellen Zeitpunkt den Startzeitpunkt des angegebenen Uhrzeitalarms auf die nächstliegende der möglichen 4 Umschaltzeiten ein und aktiviert den Uhrzeitalarm mit der Kennung 'einmalig'. Der Uhrzeitalarm wird also ausgelöst, wenn dieser Zeitpunkt erreicht wird.
- **OB1x (Uhrzeitalarm):**
  Der OB1x wurde aktiv, weil ein Umschaltzeitpunkt erreicht wurde. "SET_SW_S" stellt die Uhrzeit und den Uhrzeitstatus der CPU entsprechend der aktuellen Zeit, parametriert sich auf den nächsten Umschaltzeitpunkt und aktiviert diesen. Im Abschnitt (5) stellt er sich auf die Sommerzeitankündigung des Folgejahrs.

"SET_SW_S" darf nicht mehrfach instanziert werden.

Mit folgenden Anweisungen können Sie den Uhrzeitalarm abfragen, parametrieren, aktivieren und löschen: "QRY_TINT (Seite 5610)", "SET_TINT (Seite 5607)", "ACT_TINT (Seite 5609)", "CAN_TINT (Seite 5608)"

Hinweis

Ein Verstellen der CPU-Uhrzeit, z. B. per PG, bei aktivem "SET_SW_S" (REQ = 1) ist nur dann ohne besondere Zusatzmaßnahmen erlaubt, wenn die Zeit nicht über einen der Umschaltzeitpunkte hinweg verstellt wird.

Wird beim Stellen der Uhr einer der Umschaltzeitpunkte übersprungen, müssen Sie die Sommer-/Winterzeit-Umschaltung vor dem Verstellen mit REQ = 0 deaktivieren und nach dem Verstellen durch REQ = 1 neu starten.

Beim Stellen der CPU-Uhr und des Uhrzeitstatus müssen Sie darauf achten, dass Sie die Sommerzeit-Kennung oder die Ankündigungs-Stunden-Kennung setzen, abhängig davon, ob Sie Sommer- oder Winterzeit einstellen.

TIMESTMP: Zeitgestempelte Meldungen übertragen

Beschreibung

Die Anweisung überträgt zeitgestempelte Meldungen einer IM153-2 in seinen Instanz-DB. Dort stehen die Daten für eine weitere Verarbeitung durch das Anwenderprogramm zur Verfügung.

- IM153-2
  Die IM153-2 erfasst Änderungen von Binärsignalen, versieht diese mit einem Zeitstempel und legt diese Information in Datensätzen mit maximal 20 Meldungen ab. Bei bestimmten Ereignissen gibt sie Sondermeldungen aus. Wenn ein Datensatz zum Abholen bereit ist, löst die IM153-2 einen Prozessalarm aus.
- Aufruf im Prozessalarm-OB
  "TIMESTMP" liest die Startinformation des Prozessalarm-OBs und speichert die Information, welche die Zeitstempelung betrifft, für die weitere Bearbeitung in einem Umlaufpuffer. "TIMESTMP" kann die Daten von maximal 17 Prozessalarmen speichern.
- Aufruf im zyklischen OB:
  In der zyklischen Bearbeitung liest "TIMESTMP" den abzuholenden Datensatz mithilfe der Anweisung "RD_REC (Seite 5490)" (Datensatz lesen) und speichert ihn im Meldepuffer MESSAGE. Sind mehrere Datensätze abzuholen, wird der älteste zuerst gelesen. Wenn das Bit TIMECONV am Eingang gesetzt ist, rechnet "TIMESTMP" alle Zeitstempel aus dem ISP-Format in das Format DATE_AND_TIME (DT) um.
  Ist der Parameter BUFRDY = TRUE, können die Meldungen im Meldepuffer vom Anwenderprogramm weiterverarbeitet und z. B. zu einem Empfänger (B&B-Gerät, Drucker) übertragen werden. Die Anzahl der vorhandenen Meldungen wird im Parameter MSG_QTY angezeigt. Nach der Bearbeitung muss das Anwenderprogramm BUFRDY zurücksetzen, um den Meldepuffer für weitere Meldungen frei zu geben.
  Der Parameter BUFNOTREAD zeigt die Anzahl der empfangenen Prozessalarme an, zu denen "TIMESTMP" noch keine Datensätze aus der IM153-2 gelesen hat. Bei BUFNOTREAD = 15 sind alle Datensätze der IM153-2 gefüllt. Die IM153-2 erkennt dann keine weiteren Signaländerungen. Weitere Datensätze, die bei einer Änderung von Binärsignalen durch die IM153-2 erzeugt werden, gehen verloren. Der Parameter MSGLOST (Meldungsverlust) wird auf "1" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TIMESTMP":

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	Input	INT	Logische Adresse (Diagnoseadresse) der IM153-2
LADDR2	Input	INT	Logische Adresse (Diagnoseadresse) von Slot 2 der IM153-2
TIMECONV	Input	BOOL	1=ISP-Zeitstempel in S7-DT-Format wandeln
MSG_QTY	Output	INT	Anzahl der gültigen Meldungen im Meldungspuffer
BUFNOTREAD	Output	INT	Anzahl der alarmierten, aber noch nicht gelesenen Datensätze der IM153-2
READERR	Output	BOOL	1=Fehler beim Lesen von der IM153-2
READSTATUS	Output	INT	Rückgabewert der Anweisung "RD_REC (Seite 5490)"
BUFRDY	InOut	BOOL	1=Meldungspuffer bereit
MSGLOST	InOut	BOOL	1=Meldungen verloren
MESSAGE	Static	ARRAY[1..20] OF STRUCT	Meldepuffer für max. 20 Meldungen der IM153-2
SLOT_NO	./.	BYTE	Slotnummer/2=Sondermeldung
CH_NO	./.	BYTE	Kanalnummer/Kennung der Sondermeldung
SIGNAL	./.	BYTE	Signalzustand/Merkmal der Sondermeldung
TIME1	./.	DWORD	Sekunden ab 1900,0 (ISP)/Jahr, Monat, Tag, Stunde (DT)
TIME2	./.	DWORD	Sekundenteile (ISP)/Minute, Sekunde, Millisekunde, Wochentag (DT)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hardware-Konfiguration

Sie können "TIMESTMP" nur in Konfigurationen einsetzen, die eine Uhrzeitweiterleitung und eine Zeitstempelung von Binärsignalen unterstützen. Diese sind:

- CPUs
  - CPU 4xx mit CP443-5 zur Weiterleitung der Uhrzeittelegramme (ab 6GK7443-5DX02-0XE0), oder
  - CPU 4xx mit Weiterleitung der Uhrzeittelegramme (z. B. CPU416-2 DP ab Firmwarestand 3.0)
- Dezentrale Peripherie
  - Anschaltbaugruppe IM153-2 redundanzfähig mit Uhrzeitstempelung (ab 6ES7153-2AA02 Firmwarestand V1.2.3 (Ausgabestand 10))
    Oder:
    IM153-2 FO redundanzfähig mit integrierter LWL-Schnittstelle (ab 6ES7153-2AB01 ab Firmwarestand V1.2.3 (Ausgabestand 9))
  - Digitaleingabe SM321 (6ES7321-7BH00-0AB0) potenzialgetrennt 16 DE, DC 24V, Prozessalarm, Diagnose
    Oder:
    Digitaleingabe SM321 (6ES7321-7TH00-0AB0) potenzialgetrennt 16 DE, DC 24V, NAMUR, diagnosefähig, mit Leittechnikfunktionen
    Die Zeitstempelung sollten Sie nur für ausgewählte und für Ihre Anwendung wichtige Binärsignale einsetzen. Es wird empfohlen, wegen der Auslastung des PROFIBUS-DP und der IM153-2 diese Signale gegebenenfalls über mehrere IM153-2 bzw. auch über mehrere Stationen zu verteilen.

Software-Konfiguration

- Aufrufende OBs
  "TIMESTMP" muss mit derselben Instanz in folgenden Organisationsbausteinen (OBs) aufgerufen werden, wobei die Parameter nur im zyklischen Programm ver- und entsorgt werden dürfen:
  - OB1 (zyklisches Programm) oder alternativ ein Weckalarm-OB (OB30 bis OB38)
  - OB86 (Baugruppenträgerausfall)
  - OB100 (Anlauf)
    Falls Sie andere Anlauf-OBs (OB101, OB102) verwenden, muss "TIMESTMP" auch dort aufgerufen werden.
  - OB40 oder, falls in der Hardware-Konfiguration auswählbar, auch ein anderer Prozessalarm-OB (OB41 bis OB47)
- Adressierung
  - Verwenden Sie für jede IM153-2 eine eigene Instanz von "TIMESTMP". Tragen Sie am Eingang LADDR die in der Hardware-Konfiguration eingestellte logische Adresse (Diagnoseadresse) der IM153-2 ein.
  - Wenn Sie den DP-Master in der Betriebsart "S7-kompatibel" verwenden, tragen sie am Eingang LADDR2 denselben Wert wie für LADDR ein. Verwenden Sie die Betriebsart "DPV1", so tragen Sie bei LADDR2 die Diagnoseadresse von Steckplatz 2 der IM153-2 ein.
  - Wenn Sie die Adressen während des Betriebs ändern, löscht "TIMESTMP" die bisher gespeicherten Prozessalarmdaten.

Fehlerbehandlung

Wenn "TIMESTMP" nicht behebbare Fehler beim Lesen eines Datensatzes festgestellt hat, setzt "TIMESTMP" den Parameter READERR auf TRUE. Für eine weitere Analyse steht der Rückgabewert der Anweisung WR_USMSG im Parameter READSTATUS zur Verfügung. Da "TIMESTMP" im nächsten Zyklus gegebenenfalls einen weiteren Datensatz liest, sind READERR und READSTATUS nur für einen Zyklus gültig. Sie sollten daher im Anwenderprogramm eine entsprechende Auswertung vorsehen.

Werden Datensätze gelesen, während der DP-Master ausgefallen ist, gibt "TIMESTMP" READERR = TRUE und READSTATUS =80B2h ("Der projektierte Steckplatz ist nicht belegt.") aus.

Anlaufverhalten IM153-2

Im Anlauf/Wiederanlauf/Kaltstart gibt die IM153-2 nochmals Prozessalarme für die Datensätze ab, die vor dem Anlauf belegt waren, aber noch nicht abgeholt wurden.

In den ersten freien Datensatz trägt die IM153-2 folgende Meldungen ein:

- Sondermeldung "Beginn Anlaufdaten"
- Signaländerungen, die unmittelbar vor einem CPU-STOP aufgetreten sind ("kommend" oder "gehend" gemäß der parametrierten Flanke)

- Aktueller Signalstatus für alle zeitzustempelnden Binärsignale ("kommend" oder "gehend" gemäß der parametrierten Flanke)
- Sondermeldung "Ende Anlaufdaten"

Anlaufverhalten von "TIMESTMP"

Wird "TIMESTMP" in einem Anlauf-OB (OB100, OB101, OB102) aufgerufen, löscht "TIMESTMP" alle gespeicherten Prozessalarmdaten und setzt BUFRDY zurück. Sind noch alarmierte und nicht gelesene Datensätze bekannt, setzt "TIMESTMP" MSGLOST (Meldungsverlust) auf TRUE. Das Anwenderprogramm muss MSGLOST wieder zurücksetzen.

Verhalten von "TIMESTMP" im OB86 (Baugruppenausfall)

Wird "TIMESTMP" im OB86 aufgerufen und kommt das Ereignis von der zugehörigen IM153-2, so reagiert "TIMESTMP" wie im Anlauf. Dies ist z. B. bei Ausfall und Wiederkehr der IM153-2 und Wiederkehr des DP-Masters der Fall.

Redundanz

Bei H-Systemen mit zwei IM153-2 ist bezüglich der Zeitstempelung in folgenden Fällen Redundanz vorhanden:

- Die Kommunikation läuft zwischen beiden IM153-2 über K-Bus.
- Das Aufdaten der aktiven und passiven IM153-2 ist fehlerfrei abgeschlossen.

Während einer Umschaltung zwischen aktiver und passiver IM153-2 ist die Zeitstempelung unterbrochen. Mit der Sondermeldung "Umschalten bei Redundanz Anfang/Ende" wird der Unterbrechungszeitraum angezeigt.

Im Normalfall teilt die aktive der passiven IM153-2 den aktuellen Peripheriezustand mit. Wenn diese Kommunikation gestört ist, wird die Sondermeldung "Redundanz-Informationsverlust kommend" ausgegeben. Sobald zwischen aktiver und passiver IM153-2 die Kommunikation wieder möglich ist, wird die Sondermeldung "Redundanz-Informationsverlust gehend" ausgegeben. Ein nachträglicher Abgleich ist nicht möglich, d. h. ein Umschalten im Fehlerzustand der IM153-2 führt zu Meldungsverlust.

Aufbau des Meldepuffers

Der Meldepuffer enthält einen Datensatz der IM153-2. Die Interpretation des Zeitstempels hängt von der Einstellung des Parameters TIMECONV ab.

Zeitformate von "TIMESTMP"

Wenn der Eingang TIMECONV = TRUE ist, werden alle Zeitstempel im Datentyp DATE_AND_TIME (BCD-Format) gespeichert:

Die folgende Tabelle zeigt den Aufbau des Datentyps DATE_AND_TIME:

Bytes	Inhalt	Bereich
0	Jahr	90 … 89 Entsprechend 1990 ... 2089
1	Monat	01 ... 12
2	Tag	01 ... 31
3	Stunde	00 ... 23
4	Minute	00 ... 59
5	Sekunde	00 ... 59
6	2 MSD (most significant decade) von ms	00 ... 99
7 (4 MSB)	LSD (least significant decade) von ms	0 ... 9
7 (4 LSB)	Wochentag	1 ... 7 (1 = Sonntag)

Wenn der Eingang TIMECONV = FALSE ist, werden alle Zeitstempel im ISP-Format (2 Doppelworte) gespeichert.

Die folgende Tabelle zeigt die Aufteilung der Bytes:

Bytes	Inhalt	Bereich
0 … 3	Sekunden ab 1.1.1900 0:00:00,000 Uhr	Entsprechend 1.1.1900 ... 6.2.2036
4 … 7	Sekundenbruchteile in Vielfachen von $1/2^{32}$ s	0 … <1

Die Zeitkonvertierung ist in dem Datumsbereich gültig, der von den Zeitformaten gemeinsam abgedeckt ist, d. h. vom 1.1.1990 bis einschließlich 6.2.2036.

WS_RULES: Regel-DB zur Umrechnung der Basiszeit in Lokalzeit

Datenbaustein vom Typ "WS_RULES"

Ein Datenbaustein vom Typ "WS_RULES" hat den folgenden Aufbau:

Name		Datentyp	Startwert	Beschreibung
// Umrechnung Basiszeit<->Lokalzeit und "Alarm stellen nach Lokalzeit"				
B2L		Struct		
	S	INT	2	// Offset Basis->Lokalzeit [30 Min] im Winter, zulässig -24 ... +24.
	T	INT	2	// Differenz Winter- und Sommerzeit [30 Min], zulässig: 2
// Regel für: Winterzeit -> Sommerzeit. Default: letzter Sonntag im März; 2:00 Uhr				
W2S		Struct		// in WINTERZEIT
	M	BYTE	B#16#3	// Monat der Umstellung
	W	BYTE	B#16#9	// n-tes Auftreten des Wochentags (1=erstes, 2=zweites,..,9=letztes)
	D	BYTE	B#16#1	// Wochentag (Sonntag = 1)
	H	BYTE	B#16#2	// Stunde

Name		Datentyp	Startwert	Beschreibung
// Regel für: Sommerzeit -> Winterzeit. Default: letzter Sonntag im Oktober, 3:00 Uhr				
S2W				// in SOMMERZEIT
	M	BYTE	B#16#10	// Monat der Umstellung
	W	BYTE	B#16#9	// n-tes Auftreten des Wochentags (9 = letzte)
	D	BYTE	B#16#1	// Wochentag (Sonntag = 1)
	H	BYTE	B#16#3	// Stunde

Hinweis

Datenbaustein vom Typ "WS_RULES" erstellen

Einen Datenbaustein vom Typ "WS_RULES" können sie erstellen, indem Sie beim Erstellen eines neuen DBs als Typ "WS_RULES" auswählen.

Hinweis

Alle Parameter die das Format BYTE haben, werden als BCD-Werte (binär-codierten Dezimalwert) interpretiert! W2S ist in Winterzeit, S2W in Sommerzeit anzugeben. Die Festlegung der Sommer-/Winterzeitumschaltpunkte durch eine Regel ist in der EU ab dem Jahr 2002 vorgeschrieben.

4.2.3.2 String + Char

S_COMP: Zeichenketten vergleichen

Beschreibung

Die Anweisung vergleicht die Inhalte zweier Variablen im Format STRING und gibt das Vergleichsergebnis als Rückgabewert aus. Die zu vergleichenden Variablen werden an den Eingängen IN1 und IN2 verschaltet. Die Eingangsparameter können nur mit einer symbolisch definierten Variable belegt werden.

Über die Anweisungsbox wählen Sie die Vergleichsbedingung aus. Ist die Vergleichsbedingung (z. B. größer oder gleich) erfüllt, wird am Ausgangsparameter OUT der Signalzustand auf "1" gesetzt.

Die folgenden Vergleichsbedingungen können verwendet werden:

Symbol	Beschreibung
EQ	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 gleich der Zeichenkette am Parameter IN2 ist.
NE	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 ungleich der Zeichenkette am Parameter IN2 ist.
GT (1)	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 größer als die Zeichenkette am Parameter IN2 ist.

Symbol	Beschreibung
LT (1)	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 kleiner als die Zeichenkette am Parameter IN2 ist.
GE (1)	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 größer oder gleich der Zeichenkette am Parameter IN2 ist.
LE (1)	Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 kleiner oder gleich der Zeichenkette am Parameter IN2 ist.
(1) Die Zeichen werden beginnend von links über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Ist der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette, gilt die längere Zeichenkette als größer.	

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S_COMP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING*	D, L	Eingangsvariable im Format STRING
IN2	Input	STRING*	D, L	Eingangsvariable im Format STRING
OUT	Output	BOOL	E, A, M, D, L	Vergleichsergebnis
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

S_CONV: Zeichenkette konvertieren

Beschreibung

Mit der Anweisung konvertieren Sie den Wert am Eingang IN in das Datenformat, das Sie am Ausgang OUT angegeben haben. Über die Anweisung können Sie Zeichenketten in Ganz- und Gleitpunktzahlen oder Ganz- und Gleitpunktzahlen in Zeichenketten konvertieren.

Hinweis

Explizite Konvertierungen in SCL

Weitere Informationen und SCL-spezifische Besonderheiten bei expliziten Konvertierungen finden Sie hier:

- Explizite Konvertierungen (Seite 592)

Konvertierung einer Zeichenkette in Ganz- und Gleitpunktzahlen

Folgende Konvertierungen einer Zeichenkette in Ganz- und Gleitpunktzahlen sind möglich:

- Konvertierung einer Zeichenkette in eine Variable im INT-Format. Das erste Zeichen in der Zeichenkette darf ein Vorzeichen oder eine Ziffer sein, die dann folgenden Zeichen müssen aus Ziffern bestehen. Ist die Länge der Zeichenkette Null oder größer als 6 oder befinden sich unerlaubte Zeichen in der Zeichenkette, findet keine Konvertierung statt und das BIE-Bit wird auf "0" gesetzt. Liegt das Ergebnis der Wandlung außerhalb des INT-Zahlenbereichs, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.
- Konvertierung einer Zeichenkette in eine Variable im Format DINT. Das erste Zeichen in der Zeichenkette darf ein Vorzeichen oder eine Ziffer sein, die dann folgenden Zeichen müssen aus Ziffern bestehen. Ist die Länge der Zeichenkette Null oder größer als 11 oder befinden sich unerlaubte Zeichen in der Zeichenkette, findet keine Konvertierung statt und das BIE-Bit wird auf "0" gesetzt. Liegt das Ergebnis der Konvertierung außerhalb des DINT-Zahlenbereichs, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.
- Konvertierung einer Zeichenkette in eine Variable im Format REAL. Die Zeichenkette muss in folgendem Format vorliegen: ±v.nnnnnnnE±xx
  - ± = Vorzeichen
  - v = 1 Vorkommastelle
  - n = 7 Nachkommastellen
  - x = 2 Exponentenstellen

  Ist die Länge der Zeichenkette kleiner als 14 oder ist sie nicht wie oben gezeigt aufgebaut, findet keine Konvertierung statt und das BIE-Bit wird auf "0" gesetzt. Liegt das Ergebnis der Konvertierung außerhalb des REAL-Zahlenbereichs, wird das Ergebnis auf den entsprechenden Wert begrenzt und das Binärergebnis BIE auf "0" gesetzt.

Konvertierung einer Ganz- und Gleitpunktzahl in eine Zeichenkette

Folgende Konvertierungen von Ganz- und Gleitpunktzahlen in eine Zeichenkette sind möglich:

- Konvertierung einer Variable im INT-Format in eine Zeichenkette. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Ist die am Rückgabeparameter angegebene Variable zu kurz, findet keine Konvertierung statt und das BIE-Bit wird auf "0" gesetzt. Der Ausgangsparameter kann nur mit einer symbolisch definierten Variable belegt werden.
- Konvertierung einer Variable im DINT-Format in eine Zeichenkette. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Ist die am Rückgabeparameter angegebene Variable zu kurz, findet keine Konvertierung statt und das BIE-Bit wird auf "0" gesetzt. Der Ausgangsparameter kann nur mit einer symbolisch definierten Variable belegt werden.
- Konvertierung einer Variable im REAL-Format in eine Zeichenkette. Die Zeichenkette wird mit 14 Stellen darstellt: ±v.nnnnnnnE±xx
  - ± = Vorzeichen
  - v = 1 Vorkommastelle
  - n = 7 Nachkommastellen
  - x = 2 Exponentenstellen

  Ist die am Rückgabeparameter angegebene Variable zu kurz oder liegt am Parameter IN keine gültige Gleitpunktzahl an, findet keine Wandlung statt und das BIE-Bit wird auf "0" gesetzt. Der Ausgangsparameter kann nur mit einer symbolisch definierten Variable belegt werden.

Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "S_CONV":

Tabelle 4-189 Parameter bei der Konvertierung einer Zeichenkette in Ganz- und Gleitpunktzahlen

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L	Zu konvertierender Wert
OUT	Output	INT, DINT, REAL	E, A, M, D, L	Ergebnis der Konvertierung.
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Tabelle 4-190 Parameter bei der Konvertierung einer Ganz- und Gleitpunktzahl in eine Zeichenkette

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	INT, DINT, REAL	E, A, M, D, L oder Konstante	Zu konvertierender Wert
OUT	Output	STRING*	D, L	Ergebnis der Konvertierung
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

ATH: ASCII-Zeichenkette in Hexadezimalzahl konvertieren

Beschreibung

Mit der Anweisung konvertieren Sie die am Eingang IN angegebene ASCII-Zeichenkette in eine Hexadezimalzahl. Das Ergebnis der Konvertierung wird an der vom Parameter OUT angegebenen Adresse abgelegt.

- Die Anzahl der zu konvertierenden ASCII-Zeichen legen Sie durch den Parameter N fest.
- Maximal können 65 535 gültige ASCII-Zeichen konvertiert werden.
- Nur die Ziffern 0 bis 9 und die Großbuchstaben A bis F können interpretiert werden. Alle anderen Zeichen werden in Nullen umgewandelt.

Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist das Ausgabewort nur halb so lang wie das Eingabewort. Die ASCII-Zeichen werden nach dem Umwandeln in dem Ausgang in der gleichen Reihenfolge angeordnet, in der sie eingelesen wurden. Wenn es sich um eine ungerade Anzahl von ASCII-Zeichen handelt, wird die Hexadezimalzahl in dem Halbbyte rechts von der zuletzt umgewandelten Hexadezimalzahl mit Nullen aufgefüllt.

Wenn ein ungültiges ASCII-Zeichen erkannt wird, dann wird es als "0" umgewandelt und eine Fehlermeldung am Parameter RET_VAL ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATH":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	POINTER	D, L	Zeiger auf die Anfangsadresse einer ASCII-Zeichenkette
N	Input	WORD	E, A, M, D, L, P oder Konstante	Anzahl der zu konvertierenden ASCII-Zeichen
RET_VAL	Output	WORD	E, A, M, D, L, P	Status der Anweisung
OUT	Output	POINTER	E, A, M, D, L	Zeiger auf die Adresse des Ergebnisses
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#....)	Beschreibung
0000	Kein Fehler
0007	Ungültiges Zeichen

Beispiel

Wenn der Signalzustand am Eingang E 0.0 = 1 (aktiviert) ist, dann wird die Anweisung ausgeführt. Der Eingangsparameter N = 5 gibt an, dass fünf ASCII-Zeichen umgewandelt werden sollen. Die ASCII-Zeichen werden in dem Datenbaustein 1 gespeichert, der an der Adresse beginnt, die von dem Pointer IN angegeben wird: DB1.DBX10.0. Die Zeichenkette wird an der Adresse ausgegeben, die von dem Pointer OUT angegeben wird: Beginn bei DB2.DBX0.0 (Datenbaustein 2). Weil es sich hier um eine ungerade Anzahl an ASCII-Zeichen handelt, enthält die letzte Hexadezimalziffer nur Nullen in dem rechten Halbbyte, wodurch der Hexadezimalwert 0xC0 entsteht.

Wurde die Anweisung fehlerfrei ausgeführt wird RET_VAL auf den Wert W#16#0000 gesetzt.

Die folgende Tabelle zeigt Beispiele für die Konvertierung von ASCII-Zeichenketten in Hexadezimalzahl.

IN	N	OUT	BIE-Status
'0123'	4	16#0123	1
'123AFx1a23'	10	16#123AF01023	0

Tabelle 4-191 Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte.

ASCII-Zeichen	ASCII-codierter Hexadezimalwert	Hexadezimalziffer
"0"	30	0
"1"	31	1
"2"	32	2
"3"	33	3
"4"	34	4
"5"	35	5
"6"	36	6
"7"	37	7
"8"	38	8
"9"	39	9
"A"	41	A
"B"	42	B
"C"	43	C
D	44	D
E	45	E
F	46	F

HTA: Hexadezimalzahl in ASCII-Zeichenkette konvertieren

Beschreibung

Mit der Anweisung konvertieren Sie die am Eingang IN angegebene Hexadezimalzahl in eine ASCII-Zeichenkette. Das Ergebnis der Konvertierung wird an der am Parameter OUT angegebenen Adresse abgelegt.

Die Anzahl der zu konvertierenden Hexadezimalbytes legen Sie durch den Parameter N fest. Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist der Ausgabewert doppelt so lang wie der Eingabewert. Jedes Halbbyte der Hexadezimalzahl wird in ein Zeichen unter Beibehaltung der ursprünglichen Reihenfolge umgewandelt.

In die ASCII-Zeichenkette können maximal 65 635 Zeichen geschrieben werden. Das Ergebnis der Konvertierung wird mit den Ziffern 0 bis 9 und Großbuchstaben A bis F dargestellt.

Die Anweisung erkennt keine Fehlerbedingungen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "HTA":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	POINTER*	E, A, M, D	Anfangsadresse der Hexadezimalziffer
N	Input	WORD	E, A, M, D, L, P oder Konstante	Anzahl der zu konvertierenden Hexadezimalbytes
OUT	Output	POINTER*	D, L	Adresse, an der das Ergebnis abgelegt wird.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Beispiel

Wenn der Signalzustand am Eingang E 0.0 = 1 (aktiviert) ist, dann wird die Anweisung ausgeführt. Der Eingangsparameter N = 3 gibt an, dass drei Hexadezimalzeichen umgewandelt werden sollen. Die Hexadezimalbytes werden in dem Datenbaustein 1 gespeichert, der an der Adresse beginnt, die von dem Pointer IN angegeben wird: DB1.DBX10.0. Die Zeichenkette wird an der Adresse ausgegeben, die von dem Pointer OUT angegeben wird: Beginn bei DB2.DBX0.0 (Datenbaustein 2).

Die folgende Tabelle zeigt Beispiele für die Konvertierung von Hexadezimalzahlen in ASCII-Zeichenketten:

IN	N	OUT	BIE-Status
W#16#0123	2	'0123'	1
16#123AF01023	4	'123AF010'	0

Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte:

Hexadezimalziffer	ASCII-codierter Hexadezimalwert	ASCII-Zeichen
0	30	"0"
1	31	"1"
2	32	"2"
3	33	"3"
4	34	"4"
5	35	"5"
6	36	"6"

Hexadezimalziffer	ASCII-codierter Hexadezimalwert	ASCII-Zeichen
7	37	"7"
8	38	"8"
9	39	"9"
A	41	"A"
B	42	"B"
C	43	"C"
D	44	"D"
E	45	"E"
F	46	"F"

Weitere Anweisungen

LEN: Länge einer Zeichenkette ermitteln

Beschreibung

Die Anweisung gibt die aktuelle Länge einer Zeichenkette (Anzahl der gültigen Zeichen) als Rückgabewert aus.

Eine STRING-Variable enthält zwei Längen:

- Die maximale Länge (sie wird bei der Definition der Variablen in eckigen Klammern vorgegeben)
- Die aktuelle Länge (das ist die Anzahl der momentan gültigen Zeichen)

Die aktuelle Länge ist kleiner oder gleich der maximalen Länge. Die Anzahl der durch eine Zeichenkette belegten Bytes ist um 2 größer als die maximale Länge. Ein Leerstring ('') hat die Länge Null. Die maximale Länge beträgt 254. Die Anweisung meldet keine Fehler.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEN":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L* *	Eingangsvariable im Format STRING; Der Eingangsparameter kann nur mit einer symbolisch definierten Variable belegt werden.
OUT	Return	INT	E, A, M, D, L	Anzahl der aktuellen Zeichen
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps). ** Bei SCL: Auch Verwendung von Konstanten möglich.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

CONCAT: Zeichenketten zusammenfassen

Beschreibung

Die Anweisung fasst zwei STRING-Variablen zu einer Zeichenkette zusammen. Ist die Ergebniszeichenkette länger als die am Ausgangsparameter angelegte Variable, wird die Ergebniszeichenkette auf die maximal eingerichtete Länge begrenzt und das BIE-Bit auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CONCAT". Die Parameter können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING*	D, L	Eingangsvariable im Format STRING
IN2	Input	STRING*	D, L	Eingangsvariable im Format STRING
OUT	Return	STRING*	D, L	Zusammengefasste Zeichenkette
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

LEFT: Linke Zeichen einer Zeichenkette lesen

Beschreibung

Die Anweisung liefert die ersten L Zeichen einer Zeichenkette.

- Ist L größer als die aktuelle Länge der STRING-Variable, wird der Eingangswert zurückgeliefert.
- Bei L = 0 und bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert.
- Ist L negativ wird ein Leerstring ausgegeben und das BIE-Bit auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEFT". Der Parameter IN und der Rückgabewert RET_VAL können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L	Eingangsvariable im Format STRING
L	Input	INT	E, A, M, D, L oder Konstante	Länge der linken Zeichenkette

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Return	STRING*	D, L	Ausgangsvariable im Format STRING
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

RIGHT: Rechte Zeichen einer Zeichenkette lesen

Beschreibung

Die Anweisung liefert die letzten L Zeichen einer Zeichenkette.

- Ist L größer als die aktuelle Länge der STRING-Variable, wird der Eingangswert zurückgeliefert.
- Ist L negativ, wird ein Leerstring ausgegeben und das Binärergebnis BIE auf "0" gesetzt.
- Bei L = 0 und bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RIGHT". Der Parameter IN und der Rückgabewert RET_VAL können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L	Eingangsvariable im Format STRING
L	Input	INT	E, A, M, D, L oder Konstante	Länge der rechten Zeichenkette
OUT	Return	STRING*	D, L	Ausgangsvariable im Format STRING
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

MID: Mittlere Zeichen einer Zeichenkette lesen

Beschreibung

Die Anweisung liefert den mittleren Teil einer Zeichenkette (L Zeichen ab dem P. Zeichen einschließlich).

- Geht die Summe aus L und (P-1) über die aktuelle Länge der STRING-Variable hinaus, wird eine Zeichenkette ab dem P. Zeichen bis zum Ende des Eingangswerts geliefert.
- In allen anderen Fällen (P liegt außerhalb der aktuellen Länge, P und/oder L gleich Null oder negativ) wird ein Leerstring ausgegeben und das BIE-Bit auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MID". Der Parameter IN und der Rückgabewert RET_VAL können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L	Eingangsvariable im Format STRING
L	Input	INT	E, A, M, D, L oder Konstante	Länge der mittleren Zeichenkette
P	Input	INT	E, A, M, D, L oder Konstante	Position des ersten Zeichens
OUT	Return	STRING*	D, L	Ausgangsvariable im Format STRING
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

DELETE: Zeichen in Zeichenkette löschen

Beschreibung

Die Anweisung löscht in einer Zeichenkette L Zeichen ab dem P. Zeichen (einschließlich).

- Ist L und/oder P gleich Null oder ist P größer als die aktuelle Länge der Eingangszeichenkette, wird die Eingangszeichenkette zurückgeliefert.
- Ist die Summe aus L und P größer als die Eingangszeichenkette, wird bis zum Ende der Zeichenkette gelöscht.
- Ist L und/oder P negativ wird ein Leerstring ausgegeben und das BIE-Bit auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DELETE". Der Eingangsparameter IN und der Ausgangsparameter RET_VAL können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN	Input	STRING*	D, L	STRING-Variable, in der gelöscht wird
L	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu löschenden Zeichen
P	Input	INT	E, A, M, D, L oder Konstante	Position des 1. zu löschenden Zeichens
OUT	Return	STRING*	D, L	Ergebniszeichenkette
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

INSERT: Zeichen in Zeichenkette einfügen

Beschreibung

Die Anweisung fügt die Zeichenkette am Parameter IN2 in die Zeichenkette am Parameter IN1 nach dem P. Zeichen ein.

- Ist P gleich Null, wird die zweite Zeichenkette vor der ersten Zeichenkette eingefügt.
- Ist P größer als die aktuelle Länge der ersten Zeichenkette, wird die zweite Zeichenkette an die erste angehängt.
- Ist P negativ, wird ein Leerstring ausgegeben und das BIE-Bit auf "0" gesetzt.

Das BIE-Bit wird auch auf "0" gesetzt, wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter angegebene Variable; in diesem Fall wird die Ergebniszeichenkette auf die maximal eingerichtete Länge begrenzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "INSERT". Die Eingangsparameter IN1 und IN2 und der Ausgangsparameter können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING*	D, L	STRING-Variable, in die eingefügt wird
IN2	Input	STRING*	D, L	einzufügende STRING-Variable
P	Input	INT	E, A, M, D, L oder Konstante	Einfügeposition

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OUT	Return	STRING*	D, L	Ergebniszeichenkette
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

REPLACE: Zeichen in Zeichenkette ersetzen

Beschreibung

Die Anweisung ersetzt die Anzahl von L Zeichen der ersten Zeichenkette (IN1) ab dem P. Zeichen (einschließlich) durch die vollständige zweite Zeichenkette (IN2).

- Ist L gleich Null und P ungleich Null, wird die erste Zeichenkette zurückgeliefert.
- Ist L gleich Null und P gleich Null, wird die zweite Zeichenkette der ersten Zeichenkette vorangestellt.
- Ist L ungleich Null und P gleich Null oder Eins, wird ab dem 1. Zeichen (einschließlich) ersetzt.
- Liegt P außerhalb der ersten Zeichenkette, wird die zweite Zeichenkette an die erste Zeichenkette angehängt.
- Ist L und/oder P negativ, wird ein Leerstring ausgegeben und das BIE-Bit auf "0" gesetzt. Das BIE-Bit wird auch auf "0" gesetzt, wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter angegebene Variable; in diesem Fall wird die Ergebniszeichenkette auf die maximal eingerichtete Länge begrenzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "REPLACE". Die Eingangsparameter IN1 und IN2 und der Ausgangsparameter können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING*	D, L	STRING-Variable, in die eingesetzt wird
IN2	Input	STRING*	D, L	einzusetzende STRING-Variable
L	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu ersetzenden Zeichen
P	Input	INT	E, A, M, D, L oder Konstante	Position des 1. ersetzten Zeichens
OUT	Return	STRING*	D, L	Ergebniszeichenkette
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

FIND: Zeichen in Zeichenkette finden

INDBeschreibung

Die Anweisung liefert die Position der zweiten Zeichenkette (IN2) innerhalb der ersten Zeichenkette (IN1). Die Suche beginnt links. Es wird das erste Auftreten der Zeichenkette gemeldet.

Ist die zweite Zeichenkette in der ersten nicht vorhanden, wird Null zurückgemeldet. Die Anweisung meldet keine Fehler.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FIND". Die Eingangsparameter IN1 und IN2 können nur mit einer symbolisch definierten Variable belegt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN1	Input	STRING*	D, L	STRING-Variable, in der gesucht wird
IN2	Input	STRING*	D, L	zu suchende STRING-Variable
OUT	Return	INT	E, A, M, D, L	Position der gefundenen Zeichenkette
* Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps).				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

4.2.3.3 Prozessabbild

UPDAT_PI: Prozessabbild der Eingänge aktualisieren

Beschreibung

Mit der Anweisung aktualisieren Sie das OB 1-Prozessabbild (=Teilprozessabbild 0) der Eingänge oder ein per Projektierung definiertes Teilprozessabbild der Eingänge.

Falls Sie als Meldeverfahren für die systemseitige Prozessabbildaktualisierung das wiederholte Melden aller Peripheriezugriffsfehler projektiert haben, wird die Aktualisierung des ausgewählten Prozessabbilds stets durchgeführt.

Andernfalls wird diese Aktualisierung nur dann durchgeführt, wenn das ausgewählte Teilprozessabbild nicht systemseitig aktualisiert wird, d. h.

- wenn Sie dieses Teilprozessabbild keinem Alarm-OB zugeordnet haben, oder
- wenn Sie das Teilprozessabbild 0 ausgewählt und (per Projektierung) die Aktualisierung des OB 1-Teilprozessabbilds ausgeschaltet haben.

Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Eingänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Eingänge.
Ein Teilprozessabbild, das Sie mit "UPDAT_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC_PI (Seite 5452)" aktualisieren.

Die systemseitige Aktualisierung des OB 1-Prozessabbilds der Eingänge und der Teilprozessabbilder der Eingänge, die Sie einem Alarm-OB zugeordnet haben, findet unabhängig von "UPDAT_PI"-Aufrufen statt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT_PI":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PART	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zu aktualisierenden Teilprozessabbilds der Eingänge. Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 0 bis 15 (0 bedeutet OB 1-Prozessabbild, n mit 1 < n < 15 bedeutet Teilprozessabbild n).
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L	Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Unzulässiger Wert beim Parameter PART
8091	Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU.
8092	Das Teilprozessabbild wird systemseitig mit einem OB aktualisiert und Sie haben dafür nicht das wiederholte Melden aller Peripheriezugriffsfehler projektiert. Eine Aktualisierung mit "UPDAT_PI" wurde nicht durchgeführt.

Fehlercode (W#16#...)	Erläuterung
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.
8xyy	allgemeine Fehlerinformation, siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPRD_DAT (Seite 5496)" möglich.

UPDAT_PO: Prozessabbild der Ausgänge aktualisieren

Beschreibung

Mit der Anweisung übertragen Sie die Signalzustände des OB 1-Prozessabbilds (=Teilprozessabbild 0) der Ausgänge oder eines per Projektierung definierten Teilprozessabbilds der Ausgänge zu den Ausgabebaugruppen.

Falls Sie für das ausgewählte Teilprozessabbild einen Konsistenzbereich definiert haben, werden die zugehörigen Daten konsistent an die entsprechende Peripheriebaugruppe übertragen.

Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Ausgänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Ausgänge. Ausgänge, die Sie mit "UPDAT_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC_PO (Seite 5454)" aktualisieren.

Die systemseitige Übertragung des OB 1-Prozessabbilds der Ausgänge und der Teilprozessabbilder der Ausgänge, die Sie einem Alarm-OB zugeordnet haben, zu den Ausgabebaugruppen erfolgt unabhängig von "UPDAT_PO"-Aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT_PO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PART	Input	BYTE	E, A, M, D, L oder Konstante	Nummer des zu übertragenden Teilprozessabbilds der Ausgänge. Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 0 bis 15. (0 bedeutet OB 1-Prozessabbild, n mit 1 < n < 15 bedeutet Teilprozessabbild n)
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L	Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Unzulässiger Wert beim Parameter PART
8091	Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU.
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPWR_DAT (Seite 5499)" möglich.

SYNC_PI: Prozessabbild der Eingänge synchronisieren

Beschreibung

Mit der Anweisung "SYNC_PI" aktualisieren Sie ein Teilprozessabbild der Eingänge taktsynchron. Ein an einen DP-Takt bzw. PN-Sendetakt angebundenes Anwenderprogramm kann mit dieser Anweisung die erfassten Eingangsdaten in einem Teilprozessabbild der Eingänge taktsynchron und konsistent aktualisieren.

Aufruf

"SYNC_PI" ist unterbrechbar und kann nur in den OBs 61, 62, 63 und 64 aufgerufen werden.

Hinweis

Der Aufruf der Anweisung "SYNC_PI" in den OBs 61 bis 64 ist nur dann erlaubt, wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben.

Ein Teilprozessabbild, das Sie mit "SYNC_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT_PI (Seite 5449)" aktualisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC_PI":

Parameter	Deklaration	Datentyp	Speicherbereich	Wertebereich	Beschreibung
PART	Input	BYTE	E, A, M, D, L oder Konstante	1 bis 30	Nummer des Teilprozessabbildes der Eingänge, welches taktsynchron aktualisiert werden soll.
RET_VAL	Return	INT	E, A, M, D, L	-	Fehlerinformation
FLADDR	Output	WORD	E, A, M, D, L	-	Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Ereignisklasse Fehlercode	Erläuterung
W#16#0001	Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Slaves bzw. IO-Devices sind jedoch konsistent übertragen worden.
W#16#8090	Unzulässiger Wert am Parameter PART oder Aktualisierung des angegebenen Teilprozessabbilds der Eingänge ist in diesem OB nicht erlaubt. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.
W#16#8091	Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.
W#16#80A0	Bei der Aktualisierung wurde ein Zugriffsfehler erkannt. Betroffene Eingänge wurden auf "0" gesetzt.
W#16#80A1	Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert. Der DP- bzw. PN-Zyklus ist zu kurz, um ausreichend Zeit für die Bearbeitung der Anweisung zu gewährleisten. Sie müssen daher die Zeiten TDP (auch bekannt als T_DC), Ti und To vergrößern.

Ereignisklasse Fehlercode	Erläuterung
W#16#80A2	Zugriffsfehler mit Konsistenzwarnung Bei der Aktualisierung des angegebenen Teilprozessabbilds der Eingänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. • Die Daten der fehlerhaften Eingänge wurden nicht von der Peripherie gelesen. Im Teilprozessabbild der Eingänge werden die betroffenen Eingänge auf Null gesetzt. • Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Eingangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.
W#16#80C1	Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.
W#16#8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Hinweis

Falls Sie die Anweisung "SYNC_PI" für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPRD_DAT (Seite 5496)" möglich.

SYNC_PO: Prozessabbild der Ausgänge synchronisieren

Beschreibung

Mit der Anweisung "SYNC_PO" aktualisieren Sie ein Teilprozessabbild der Ausgänge taktsynchron. Ein an einen DP-Takt bzw. PN-Sendetakt angebundenes Anwenderprogramm kann mit dieser Anweisung die berechneten Ausgangsdaten eines Teilprozessabbildes der Ausgänge taktsynchron und konsistent in die Peripherie übertragen.

Aufruf

"SYNC_PO" ist unterbrechbar und kann nur in den OBs 61, 62, 63 und 64 ausgerufen werden.

Hinweis

Der Aufruf der Anweisung "SYNC_PO" in den OBs 61 bis 64 ist nur dann erlaubt, wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben.

Ein Teilprozessabbild, das Sie mit "SYNC_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT_PO (Seite 5451)" aktualisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC_PO":

Parameter	Deklaration	Datentyp	Speicherbereich	Wertebereich	Beschreibung
PART	Input	BYTE	E, A, M, D, L oder Konstante	1 bis 30	Nummer des Teilprozessabbildes der Ausgänge, welches taktsynchron aktualisiert werden soll.
RET_VAL	Return	INT	E, A, M, D, L	-	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
FLADDR	Output	WORD	E, A, M, D, L	-	Adresse des ersten Fehler verursachenden Bytes.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Ereignisklasse Fehlercode	Erläuterung
W#16#0001	Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Slaves bzw. IO-Devices sind jedoch konsistent übertragen worden.
W#16#8090	Unzulässiger Wert am Parameter PART oder Aktualisierung des angegebenen Teilprozessabbildes der Ausgänge ist in diesem OB nicht erlaubt. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
W#16#8091	Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
W#16#80A0	Bei der Aktualisierung des angegebenen Teilprozessabbilds der Ausgänge wurde ein Zugriffsfehler erkannt. Fehlerhafte Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben diese Ausgänge unverändert.
W#16#80A1	Zugriffsfehler mit Konsistenzwarnung Bei der Aktualisierung des angegebenen Teilprozessabbilds der Ausgänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. • Die Daten der fehlerhaften Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben die betroffenen Ausgänge unverändert. • Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Ausgangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.
W#16#80A2	Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
W#16#80C1	Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.
W#16#8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Hinweis

Falls Sie die Anweisung "SYNC_PO" für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPWR_DAT (Seite 5499)" möglich.

4.2.3.4 Dezentrale Peripherie

RDREC: Datensatz lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer INDEX von der mittels ID adressierten Komponente. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

Mit MLEN geben Sie vor, wie viele Bytes Sie maximal lesen möchten. Den Zielbereich RECORD sollten Sie daher mindestens MLEN Bytes lang wählen.

Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.

Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Hinweis

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "RDREC" von den E/A-Baugruppen gelesen werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Hinweis

Die Schnittstelle der Anweisung "RDREC" ist identisch mit der des in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "RDREC".

Arbeitsweise

"RDREC" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie RDREC mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

Solange die Datensatzübertragung läuft, darf der Zielbereich nicht verändert werden, und zwar weder dessen Länge noch dessen Inhalt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RDREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Datensatzübertragung durchführen
ID	Input	DWORD	E, A, M, D, L oder Konstante	logische Adresse der DP-Slave/PROFINET IO-Komponente (Baugruppe bzw. Modul) • Bei einer Ausgabebaugruppe muss Bit 15 gesetzt werden (Bsp. für Adresse 5: ID:=DW#16#8005). • Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
INDEX	Input	INT	E, A, M, D, L oder Konstante	Datensatznummer
MLEN	Input	INT	E, A, M, D, L oder Konstante	maximale Länge der zu lesenden Datensatzinformation in Bytes
VALID	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde empfangen und ist gültig
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Lesevorgang trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus bzw. Fehlerinformation Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 5468).

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LEN	Output	INT	E, A, M, D, L	Länge der gelesenen Datensatzinformation
RECORD	InOut	ANY	E, A, M, D, L	Zielbereich für den gelesenen Datensatz Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie die Anweisung für das Lesen eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX, MLEN und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

WRREC: Datensatz schreiben

Beschreibung

Mit der Anweisung übertragen Sie den Datensatz RECORD zu der mittels ID adressierten Komponente. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

Mit LEN (ausgeblendet) geben Sie die Länge des zu übertragenden Datensatzes in Bytes vor. Den Quellbereich RECORD sollten Sie daher mindestens LEN Bytes lang wählen.

Der Wert TRUE des Ausgangsparameters DONE zeigt an, dass der Datensatz erfolgreich übertragen wurde.

Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Hinweis

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "WRREC" zu den E/A-Baugruppen geschrieben werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Hinweis

Die Schnittstelle der Anweisung "WRREC" ist identisch mit der des in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten Anweisung "WRREC".

Arbeitsweise

"WRREC" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "WRREC" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Beachten Sie, dass Sie dem Aktualparameter von RECORD bei allen zu ein und demselben Auftrag gehörenden Aufrufen von "WRREC" denselben Wert zuweisen. Dasselbe gilt für die Aktualparameter von LEN.

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Datensatzübertragung durchführen
ID	Input	DWORD	E, A, M, D, L oder Konstante	logische Adresse der DP-Slave/PROFINET IO-Komponente (Baugruppe bzw. Modul) Bei einer Ausgabebaugruppe muss Bit 15 gesetzt werden (Bsp. für Adresse 5: ID:=DW#16#8005). Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
INDEX	Input	INT	E, A, M, D, L oder Konstante	Datensatznummer
LEN	Input	INT	E, A, M, D, L oder Konstante	(ausgeblendet) maximale Länge des zu übertragenden Datensatzes in Bytes
DONE	Output	BOOL	E, A, M, D, L	Datensatz wurde übertragen
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Schreibvorgang trat ein Fehler auf.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus bzw. Fehlerinformation Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 5468).
RECORD	InOut	ANY	E, A, M, D, L	Datensatz Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie die Anweisung für das Schreiben eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

GETIO: Prozessabbild lesen

Beschreibung

Mit der Anweisung lesen Sie alle Eingänge eines DP-Normslaves / PROFINET IO-Devices konsistent aus. Die Anweisung ruft dabei die Anweisung "DPRD_DAT (Seite 5496)" auf. Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den durch INPUTS aufgespannten Zielbereich eingetragen.

Der Zielbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Komponente projektiert haben.

Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "GETIO"-Aufruf jeweils nur auf die Daten einer Komponente / DP-Kennung unter der projektierten Anfangsadresse zugreifen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	DWORD	E, A, M, D, L oder Konstante	• low word: logische Adresse der DP-Slave- / PROFINET-IO-Komponente (Baugruppe bzw. Modul) • high word: irrelevant
STATUS	Output	DWORD	E, A, M, D, L	enthält die Fehlerinformation von "DPRD_DAT (Seite 5496)" in der Form DW#16#40xxxx00
LEN	Output	INT	E, A, M, D, L	Anzahl der gelesenen Daten in Bytes
INPUTS	InOut	ANY	E, A, M, D	Zielbereich für die gelesenen Daten. Er muss genauso lang sein, wie Sie für die selektierte DP-Slave- / PROFINET-IO-Komponente projektiert haben. Es ist nur der Datentyp BYTE zulässig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Siehe auch: DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen (Seite 5496).

SETIO: Prozessabbild übertragen

Beschreibung

Mit der Anweisung "SETIO" übertragen Sie die Daten aus dem durch OUTPUTS aufgespannten Quellbereich konsistent zum adressierten DP-Normslave / PROFINET IO-Device und ggf. ins Prozessabbild (nämlich dann, wenn Sie den betroffenen Adressbereich des DP-Normslaves als Konsistenzbereich in einem Prozessabbild projektiert haben). "SETIO" ruft dabei die Anweisung "DPWR_DAT (Seite 5499)" auf.

Der Quellbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Komponente projektiert haben.

Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "SETIO"-Aufruf nur auf eine Komponente / DP-Kennung des DP-Slaves zugreifen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	DWORD	E, A, M, D, L oder Konstante	• low word: logische Adresse der DP-Slave-/ PROFINET-IO-Komponente (Baugruppe bzw. Modul) • high word: irrelevant
LEN	Input	INT	E, A, M, D, L	irrelevant
STATUS	Output	DWORD	E, A, M, D, L	enthält die Fehlerinformation von "DPWR_DAT (Seite 5499)" in der Form DW#16#40xxxx00
OUTPUTS	InOut	ANY	E, A, M, D	Quellbereich für die zu schreibenden Daten. Er muss genauso lang sein, wie Sie für die selektierte DP-Slave- / PROFINET-IO-Komponente projektiert haben. Es ist nur der Datentyp BYTE zulässig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Siehe auch: DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben (Seite 5499).

GETIO_PART: Prozessabbildbereich lesen

Beschreibung

Mit der Anweisung "GETIO_PART" lesen Sie konsistent einen Teil des zu einem DP-Normslave / PROFINET IO-Device gehörenden Prozessabbildbereichs. "GETIO_PART" ruft dabei die Anweisung "UBLKMOV" auf.

Hinweis

Sie müssen dem OB, in dem "GETIO_PART" aufgerufen wird, ein Teilprozessabbild der Eingänge zuordnen. Sie müssen weiterhin vor Aufruf von "GETIO_PART" den zugehörigen DP-Normslave bzw. das zugehörige PROFINET IO-Device in dieses Teilprozessabbild der Eingänge aufnehmen. Falls Ihre CPU keine Teilprozessabbilder kennt oder Sie "GETIO_PART" im OB 1 aufrufen wollen, müssen Sie vor Aufruf von "GETIO_PART" den zugehörigen DP-Normslave bzw. das zugehörige PROFINET IO-Device in das Prozessabbild der Eingänge aufnehmen.

Über die Parameter OFFSET und LEN legen Sie die auszulesende Teilmenge des Prozessabbildbereichs der über ID adressierten Komponente fest.

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE, und die gelesenen Daten werden in den durch INPUTS aufgespannten Zielbereich eingetragen.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE, und STATUS erhält die Fehlerinformation von "UBLKMOV".
- Falls der Zielbereich (Parameter INPUTS) kleiner ist als LEN, werden so viele Bytes übertragen, wie INPUTS aufnehmen kann. ERROR erhält den Wert FALSE. Falls der Zielbereich größer ist als LEN, werden die ersten LEN Bytes des Zielbereichs beschrieben. ERROR erhält den Wert FALSE.

Hinweis

"GETIO_PART" überprüft im Prozessabbild der Eingänge keine Grenzen zwischen Daten, die zu verschiedenen PROFIBUS-DP- bzw. PROFINET-IO-Komponenten gehören. Sie müssen daher selbst darauf achten, dass der über OFFSET und LEN festgelegte Prozessabbildbereich zu einer Komponente gehört. Das Komponenten übergreifende Lesen kann für zukünftige Systeme nämlich nicht garantiert werden und gefährdet die Übertragbarkeit auf Systeme anderer Hersteller.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO_PART":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	DWORD	E, A, M, D, L oder Konstante	• low word: logische Adresse der DP-Slave- / PROFINET-IO-Komponente (Baugruppe bzw. Modul) • high word: irrelevant
OFFSET	Input	INT	E, A, M, D, L oder Konstante	Nummer des ersten zu lesenden Bytes im Prozessabbild der Komponente (kleinstmöglicher Wert: 0)
LEN	Input	INT	E, A, M, D oder Konstante	Anzahl der zu lesenden Bytes
STATUS	Output	DWORD	E, A, M, D, L	enthält die Fehlerinformation von "UBLKMOV" in der Form DW#16#40xxxx00, falls ERROR = TRUE

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Fehleranzeige: ERROR = TRUE, falls beim Aufruf von "UBLKMOV" ein Fehler auftritt.
INPUTS	InOut	ANY	E, A, M, D	Zielbereich für die gelesenen Daten: • Falls der Zielbereich kleiner ist als LEN, werden so viele Bytes übertragen, wie INPUTS aufnehmen kann. ERROR erhält den Wert FALSE. • Falls der Zielbereich größer ist als LEN, werden die ersten LEN Bytes des Zielbereichs beschrieben. ERROR erhält den Wert FALSE.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS und ERROR

Siehe Anweisung "UBLKMOV".

SETIO_PART: Prozessabbildbereich übertragen

Beschreibung

Mit der Anweisung "SETIO_PART" übertragen Sie die Daten aus dem durch OUTPUTS aufgespannten Quellbereich konsistent in einen Teil des zu einem DP-Normslave / PROFINET IO-Device gehörenden Prozessabbildbereichs. "SETIO_PART" ruft dabei die Anweisung "UBLKMOV" auf.

Hinweis

Sie müssen dem OB, in dem "SETIO_PART" aufgerufen wird, ein Teilprozessabbild der Ausgänge zuordnen. Sie müssen weiterhin vor Aufruf von "SETIO_PART" den zugehörigen DP-Normslave bzw. das zugehörige PROFINET IO-Device in dieses Teilprozessabbild der Ausgänge aufnehmen. Falls Ihre CPU keine Teilprozessabbilder kennt oder Sie "SETIO_PART" im OB 1 aufrufen wollen, müssen Sie vor Aufruf von "SETIO_PART" den zugehörigen DP-Normslave bzw. das zugehörige PROFINET IO-Device in das Prozessabbild der Ausgänge aufnehmen.

Über die Parameter OFFSET und LEN legen Sie die zu beschreibende Teilmenge des Prozessabbildbereichs der über ID adressierten Komponente fest.

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE, und STATUS erhält die Fehlerinformation von "UBLKMOV".
- Falls der Quellbereich (Parameter OUTPUTS) kleiner ist als LEN, werden so viele Bytes übertragen, wie OUTPUTS enthält. ERROR erhält den Wert FALSE. Falls der Quellbereich größer ist als LEN, werden die ersten LEN Bytes aus OUTPUTS übertragen. ERROR erhält den Wert FALSE.

Hinweis

"SETIO_PART" überprüft im Prozessabbild der Ausgänge keine Grenzen zwischen Daten, die zu verschiedenen PROFIBUS-DP- bzw. PROFINET-IO-Komponenten gehören. Sie müssen daher selbst darauf achten, dass der über OFFSET und LEN festgelegte Prozessabbildbereich zu einer Komponente gehört. Das Komponenten übergreifende Schreiben kann für zukünftige Systeme nämlich nicht garantiert werden und gefährdet die Übertragbarkeit auf Systeme anderer Hersteller.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO_PART":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ID	Input	DWORD	E, A, M, D, L oder Konstante	• low word: logische Adresse der DP-Slave- / PROFINET-IO-Komponente (Baugruppe bzw. Modul) • high word: irrelevant
OFFSET	Input	INT	E, A, M, D, L oder Konstante	Nummer des ersten zu beschreibenden Bytes im Prozessabbild der Komponente (kleinstmöglicher Wert: 0)
LEN	Input	INT	E, A, M, D, L oder Konstante	Anzahl der zu schreibenden Bytes
STATUS	Output	DWORD	E, A, M, D, L	enthält die Fehlerinformation von "UBLKMOV" in der Form DW#16#40xxxx00, falls ERROR = TRUE

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Fehleranzeige: ERROR = TRUE, falls beim Aufruf von UBLKMOV ein Fehler auftritt.
OUTPUTS	InOut	ANY	E, A, M, D	Quellbereich für die zu schreibenden Daten: • Falls der Quellbereich kleiner ist als LEN, werden so viele Bytes übertragen, wie OUTPUTS enthält. ERROR erhält den Wert FALSE. • Falls der Quellbereich größer ist als LEN, werden die ersten LEN Bytes aus OUTPUTS übertragen. ERROR erhält den Wert FALSE.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS und ERROR

Siehe Anweisung "UBLKMOV".

RALRM: Alarm empfangen

Beschreibung RALRM

Beschreibung

Die Anweisung empfängt einen Alarm samt der zugehörigen Information von einer Peripheriebaugruppe (zentraler Aufbau) bzw. einer Komponente eines DP-Slaves oder eines PROFINET IO-Devices und stellt diese Information an seinen Ausgangsparametern zur Verfügung.

Die Information in den Ausgangsparametern enthält sowohl die Startinformation des aufgerufenen OB als auch Informationen aus der Alarmquelle.

Rufen Sie die Anweisung nur innerhalb desjenigen Alarm-OB auf, den das Betriebssystem der CPU aufgrund des zu untersuchenden Alarms aus der Peripherie gestartet hat.

Hinweis

Wenn Sie die Anweisung in einem OB aufrufen, dessen Startereignis kein Alarm aus der Peripherie ist, stellt Ihnen die Anweisung an ihren Ausgängen entsprechend weniger Information zur Verfügung.

Achten Sie darauf, dass Sie beim Aufruf der Anweisung in verschiedenen OBs unterschiedliche Instanz-DBs verwenden. Falls Sie die aus einem "RALRM"-Aufruf resultierenden Daten außerhalb des zugehörigen Alarm-OB auswerten, sollten Sie sogar pro OB-Startereignis einen eigenen Instanz-DB benutzen.

Hinweis

Die Schnittstelle der Anweisung "RALRM" ist identisch mit der des in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "RALRM".

Aufruf von "RALRM"

Die Anweisung können Sie in drei verschiedenen Betriebsarten (MODE) aufrufen, die in der folgenden Tabelle erläutert werden.

MODE	RALRM ...
0	... zeigt die Alarm auslösende Komponente im Ausgangsparameter ID an und beschreibt den Ausgangsparameter NEW mit TRUE.
1	... beschreibt sämtliche Ausgangsparameter unabhängig von der Alarm auslösenden Komponente.
2	... prüft, ob die im Eingangsparameter F_ID angegebene Komponente den Alarm ausgelöst hat. • Falls nein, erhält NEW den Wert FALSE • Falls ja, erhält NEW den Wert TRUE, und alle anderen Ausgangsparameter werden beschrieben

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RALRM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	DWORD	E, A, M, D, L oder Konstante	logische Anfangsadresse der Komponente (Baugruppe bzw. Modul), von der Alarme empfangen werden sollen
MLEN	Input	INT	E, A, M, D, L oder Konstante	maximale Länge der zu empfangenen Alarminformation in Bytes
NEW	Output	BOOL	E, A, M, D, L	Ein neuer Alarm wurde empfangen.
STATUS (Seite 5468)	Output	DWORD	E, A, M, D, L	Fehlercode der Anweisung bzw. des DP-Masters
ID	Output	DWORD	E, A, M, D, L	logische Anfangsadresse der Komponente (Baugruppe bzw. Modul), von der ein Alarm empfangen wurde Bit 15 enthält die E/A-Kennung: 0 bei einer Eingangs-, 1 bei einer Ausgangsadresse.
LEN	Output	INT	E, A, M, D, L	Länge der empfangenen Alarminformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TINFO (Seite 5472)	InOut	ANY	E, A, M, D, L	(task information) Zielbereich für OB-Startinformation und Verwaltungsinformation Hinweis: Beachten Sie, dass der Parameter TINFO bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
AINFO (Seite 5475)	InOut	ANY	E, A, M, D, L	(alarm information) Zielbereich für Kopfinformation und Alarmzusatzinformation Für AINFO sollten Sie eine Länge von mindestens MLEN Bytes vorsehen. Hinweis: Beachten Sie, dass der Parameter AINFO bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie den Zielbereich TINFO bzw. AINFO zu kurz wählen, kann die Anweisung nicht die gesamte Information eintragen.

Siehe auch: Zielbereich TINFO und AINFO (Seite 5483).

Parameter STATUS

Parameter STATUS-Struktur

Der Ausgangsparameter STATUS enthält Fehlerinformationen. Wird er als ARRAY[1...4] OF BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	• B#16#00, falls kein Fehler • Funktionskennung aus DPV1-PDU: Im Fehlerfall wird B#16#80 ausgegeben (bei Datensatz lesen: B#16#DE, bei Datensatz schreiben: B#16#DF). Falls kein DPV1-Protokollelement benutzt wird: B#16#C0.
STATUS[2]	Error_Decode	Ort der Fehlerkennung

Feldelement	Name	Bedeutung
STATUS[3]	Error_Code_1	Fehlerkennung
STATUS[4]	Error_Code_2	herstellerspezifische Erweiterung der Fehlerkennung

Feldelement STATUS[2]

STATUS[2] kann folgende Werte annehmen:

Error_Decode (B#16#...)	Quelle	Bedeutung
00 bis 7F	CPU	kein Fehler oder keine Warnung
80	DPV1	Fehler nach IEC 61158-6
81 bis 8F	CPU	B#16#8x zeigt einen Fehler im x-ten Aufrufparameter der Anweisung an.
FE, FF	DP Profile	profilspezifischer Fehler

Feldelement STATUS[3]

STATUS[3] kann folgende Werte annehmen:

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
00	00		kein Fehler, keine Warnung
70	00	reserved, reject	Erstaufruf; keine Datensatzübertragung aktiv
	01	reserved, reject	Erstaufruf; Datensatzübertragung angestoßen
	02	reserved, reject	Zwischenaufruf; Datensatzübertragung ist bereits aktiv
80	90	reserved, pass	logische Anfangsadresse ungültig
	92	reserved, pass	unzulässiger Typ bei ANY-Pointer
	93	reserved, pass	Die mittels ID bzw. F_ID adressierte DP-Komponente ist nicht konfiguriert.
	95		Fehler beim Lesen der Alarmzusatzinformation (Beim Auslesen der Alarmzusatzinformation bei zentraler Peripherie oder bei dezentraler Peripherie über externe DP-Anschaltung wird dieser Fehler als "Sammelfehler" ausgegeben.) Hinweis: Beim Ankoppeln und Aufdaten kann es vorkommen, dass die Alarmzusatzinformation temporär nicht zur Verfügung steht.

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	96		Die Master-CPU ist in STOP gegangen. Zu diesem Zeitpunkt war ein OB in Bearbeitung. Die Anweisung "RALRM (Seite 5466)" kann weder die OB-Startinformation noch die Verwaltungsinformation noch die Kopfinformation noch die Alarmzusatzinformation liefern. Die OB-Startinformation können Sie mit der Anweisung "RD_SINFO (Seite 5681)" auslesen. Darüber hinaus können Sie bei den OBs 4x, 55, 56, 57, 82 und 83 mit Hilfe der Anweisung "DPNRM_DG (Seite 5522)" das aktuelle Diagnosetelegramm des betroffenen DP-Slaves asynchron lesen (Adressinformation aus OB-Startinformation).
	A0	read error	negative Quittung beim Lesen von der Baugruppe
	A1	write error	negative Quittung beim Schreiben zur Baugruppe
	A2	module failure	DP-Protokollfehler bei Layer 2 (z. B. Slaveausfall oder Busstörungen)
	A3	reserved, pass	Allgemeiner Kommunikationsfehler oder IO-Device / DP-Slave nicht erreichbar
	A4	reserved, pass	Kommunikation am K-Bus gestört
	A5	reserved, pass	–
	A7	reserved, pass	DP-Slave oder Baugruppe ist beschäftigt (temporärer Fehler)
	A8	version conflict	DP-Slave oder Baugruppe meldet nicht passende Versionen
	A9	feature not supported	Funktion wird vom DP-Slave oder der Baugruppe nicht unterstützt
	AA bis AF	user specific	DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler seiner Anwendung. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.
	B0	invalid index	Baugruppe kennt den Datensatz nicht Datensatznummer ≥256 ist unzulässig
	B1	write length error	Die Längenangabe im Parameter RECORD ist falsch; • bei "RALRM (Seite 5466)": Längenfehler in AINFO, • bei "RDREC (Seite 5456)" und "WRREC (Seite 5458)": Längenfehler in MLEN
	B2	invalid slot	• Der projektierte Steckplatz ist nicht belegt. • Für PROFINET IO und PROFIBUS DP: IO-Device / DP-Slave nicht erreichbar
	B3	type conflict	Ist-Baugruppentyp ungleich Soll-Baugruppentyp

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	B4	invalid area	DP-Slave oder Baugruppe meldet einen Zugriff auf einen unzulässigen Bereich
	B5	state conflict	DP-Slave oder Baugruppe ist nicht bereit
	B6	access denied	DP-Slave oder Baugruppe verweigert den Zugriff
	B7	invalid range	DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Werts
	B8	invalid parameter	DP-Slave oder Baugruppe meldet einen unzulässigen Parameter
	B9	invalid type	DP-Slave oder Baugruppe meldet einen unzulässigen Typ • Bei "RDREC (Seite 5456)": Puffer zu klein (Lesen von Teilmengen ist nicht möglich) • Bei "WRREC (Seite 5458)": Puffer zu klein (Schreiben von Teilmengen ist nicht möglich)
	BA bis BF	user specific	DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler beim Zugriff. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach. Hinweis zum Wert B#16#BA: Bei PROFINET im H-System gilt: Wenn ein Datensatzauftrag mit dem Rückgabewert W#16#80BA abgewiesen wird, muss der Auftrag wiederholt werden.
	C0	read constrain conflict	Bei "WRREC (Seite 5458)": Die Daten können nur im STOP-Zustand der CPU geschrieben werden. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Sie können die Daten nur online mit PG/PC schreiben. Bei "RDREC (Seite 5456)": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. In diesem Fall können Sie die Daten nur online mit PG/PC lesen.
	C1	write constrain conflict	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.
	C2	resource busy	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.
	C3	resource unavailable	Die benötigten Betriebsmittel sind momentan belegt.
	C4		Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Erläuterung laut DPV1	Bedeutung
	C5		DP-Slave oder Baugruppe nicht verfügbar.
	C6		Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs
	C7		Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters
	C8 bis CF		DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler zu seinen Ressourcen. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.
	Dx	user specific	DP-Slave-spezifisch. Siehe Beschreibung des DP-Slaves.
8x (x = 1, ... 9, A, B, C, D, E, F)	00 bis FF		Fehler im y-ten Aufrufparameter (y = 1, ...15) • Error_Code_1 = 00: Unzulässige Betriebsart • Für alle anderen Werte von Error_Code_1 Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
FE, FF	00 bis FF		profilspezifischer Fehler

Feldelement STATUS[4]

STATUS[4] wird bei DPV1-Fehlern vom DP-Master an die CPU und die Anweisung durchgereicht. Ohne DPV1-Fehler wird der Wert auf 0 gesetzt mit folgenden Ausnahmen bei den Anweisungen "RDREC (Seite 5456)" und "WRREC (Seite 5458)":

- STATUS[4] enthält die Länge des Zielbereichs aus RECORD, falls LEN > Länge des Zielbereichs aus RECORD
- STATUS[4]=LEN, falls die tatsächliche Datensatzlänge < LEN < Länge des Zielbereichs aus RECORD
- STATUS[4]=0, falls STATUS[4] > 255 gesetzt werden müsste

Bei PROFINET IO enthält STATUS[4] den Wert 0.

Parameter TINFO

Datenstruktur des Zielbereichs TINFO

Byte	Bedeutung
0 bis 19	Startinformation des OB, in dem "RALRM" aktuell aufgerufen wurde
20 und 21	Adresse, genaue Beschreibung s. u.
22 bis 31	Verwaltungsinformation, genaue Beschreibung s. u.

Struktur der Adresse (Byte 20 und 21)

Die Adresse enthält:

- bei zentralem Aufbau die Baugruppenträgernummer (0-31).

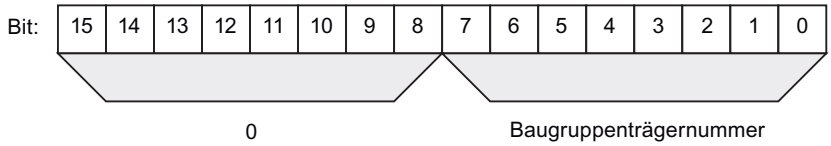

- bei dezentralem Aufbau mit PROFIBUS DP
  - die DP-Mastersystem-ID (1-32)
  - die Stationsnummer (0-127).

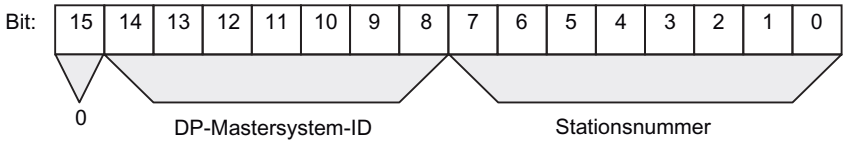

- bei dezentralem Aufbau mit PROFINET IO
  - die letzten beiden Stellen der PROFINET IO-System-ID (0-15), um die vollständige PROFINET IO-System-ID zu erhalten, müssen Sie 100 (dezimal) dazu addieren
  - die Stationsnummer (0-2047).

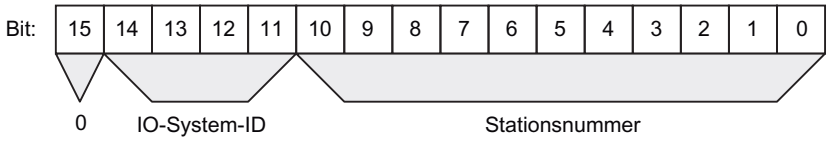

Struktur der Verwaltungsinformation Byte 20 bis 25

Byte-Nr. zu TINFO	Datentyp	Bedeutung	
20	BYTE	zentral:	0
		dezentral:	PROFIBUS DP: DP-Mastersystem-ID (mögliche Werte: 1 bis 32)
			PROFINET IO: s. o.
21	BYTE	zentral:	Baugruppenträgernummer (mögliche Werte: 0 bis 31)
		dezentral:	Nummer der DP-Station (mögliche Werte: 0 bis 127)
			PROFINET IO: s. o.

Byte-Nr. zu TINFO	Datentyp	Bedeutung				
22	BYTE	zentral:	• 0: Datensatz 0 oder Datensatz 1			
		dezentral:	Bit 0 bis 3:	Slavetyp	0000: 0001: 0010: 0011: 0100 – 0111: 1000: ab 1001:	DP (Struktur Datensatz 0) DPS7 (Struktur Datensatz 0 oder Datensatz 1) DPS7 V1 (Struktur Datensatz 0 oder Datensatz 1) DPV1 (Struktur gemäß PROFIBUS DP Standard) reserviert PROFINET IO (Struktur gemäß PROFINET IO Standard) reserviert
			• Bit 4 bis 7:	Profiltyp		reserviert
23	BYTE	zentral:	• 0			
		dezentral:	• Bit 0 bis 3:	Alarminfotyp	0000:	Transparent, bei PROFINET IO ist das immer der Fall (Alarm kommt von einer projektierten dezentralen Baugruppe)
					0001:	Stellvertreter (Alarm eines Nicht-DPV1-Slaves/Nicht IO-Device oder eines nicht projektierten Steckplatzes)
					0010:	Erzeugter (In der CPU erzeugter Alarm)
					ab 0011:	reserviert
			• Bit 4 bis 7:	Strukturversion	0000: ab 0001:	Initial reserviert
24	BYTE	zentral:	• 0			
		dezentral:	**Flags der PROFIBUS DP-Master-Anschaltung/PROFINET IO-Controller-Anschaltung**			
			Bit 0 = 0:	Alarm von einer integrierten Anschaltung (PROFINET IO oder PROFIBUS DP)		
			Bit 0 = 1:	Alarm von einer externen Anschaltung (PROFINET IO oder PROFIBUS DP)		
			Bit 1 bis 7:	reserviert		
25	BYTE	zentral:	• 0			
		dezentral:	**Flags der PROFIBUS DP-Slave-Anschaltung**			
			• Bit 0:	EXT_DIAG_FLAG aus dem Diagnosetelegramm bzw. 0, falls dieses Bit beim Alarm nicht vorliegt. Das Bit ist 1, wenn der DP-Slave gestört ist.		
			• Bit 1 bis 7:	reserviert		
			Flags der PROFINET IO-Controller-Anschaltung			

Byte-Nr. zu TINFO	Datentyp	Bedeutung		
			• Bit 0:	ARDiagnosisstate bzw. 0, falls keine Information beim Alarm vorliegt. Das Bit ist 1, wenn das IO-Device gestört ist.
			• Bit 1 bis 7:	reserviert

Struktur der Verwaltungsinformation Byte 26 bis 27 bei PROFIBUS und beim zentralen Aufbau

Byte-Nr. zu TINFO	Datentyp	Bedeutung	
26 und 27	WORD	zentral:	0
	WORD	dezentral:	PROFIBUS Identnummer als eindeutige Identifikation des PROFIBUS DP-Slave
28 und 29	WORD	0	(Bytes 28 und 29 können entfallen)
30 und 31	WORD	0	(Bytes 30 und 31 können entfallen)

Struktur der Verwaltungsinformation Byte 26 bis 31 bei PROFINET IO

Byte-Nr. zu TINFO	Datentyp	Bedeutung	
26 und 27	WORD	dezentral:	PROFINET IO-Device Identnummer als eindeutige Identifikation des PROFINET IO-Devices
28 und 29	WORD	dezentral:	Herstellerkennung
30 und 31	WORD	dezentral:	Identnummer der Instanz

Parameter AINFO

Datenstruktur des Zielbereichs AINFO bei Alarmen von PROFIBUS DP oder zentraler Peripherie

Die Angaben für PROFINET IO finden Sie weiter unten.

Byte	Bedeutung	
0 bis 3	Kopfinformation, genaue Beschreibung s. u.	
4 bis 199	Alarmzusatzinformation: Daten zum jeweiligen Alarm:	
	zentral:	ARRAY[0] bis ARRAY[195]
	dezentral:	ARRAY[0] bis ARRAY[59]

Aufbau der Kopfinformation bei Alarmen von PROFIBUS DP oder zentraler Peripherie

Byte	Datentyp	Bedeutung	
0	BYTE	Länge der empfangenen Alarminformation in Bytes	
		zentral:	4 bis 224
		dezentral:	4 bis 63

Byte	Datentyp	Bedeutung		
1	BYTE	zentral:	reserviert	
		dezentral:	Kennung für den Alarmtyp	
			1: 2: 3: 4: 5: 6: 31	Diagnosealarm Prozessalarm Ziehen-Alarm Stecken-Alarm Statusalarm Update-Alarm Ausfall eines Erweiterungsgeräts, eines DP-Mastersystems oder einer DP-Station
			32 bis 126:	herstellerspezifischer Alarm
2	BYTE	Steckplatznummer der Alarm auslösenden Komponente		
3	BYTE	zentral:	reserviert	
		dezentral:	Specifier	
			Bits 0 und 1:	0: keine weitere Information; 1: kommendes Ereignis, Steckplatz gestört 2: gehendes Ereignis, Steckplatz nicht mehr gestört 3: gehendes Ereignis, Steckplatz weiterhin gestört
			Bit 2:	Add_Ack
			Bits 3 bis 7:	Sequenznummer

Datenstruktur des Zielbereichs AINFO bei Alarmen von PROFINET IO

Byte	Bedeutung
0 bis 25	Kopfinformation, genaue Beschreibung s. u.
26 bis 1431	Alarmzusatzinformation: Genormte Diagnosedaten zum jeweiligen Alarm: ARRAY[0] bis ARRAY[1405] Hinweis: Die Alarmzusatzinformation kann auch entfallen.

Aufbau der Kopfinformation bei Alarmen von PROFINET IO

Byte	Datentyp	Bedeutung
0 und 1	WORD	• Bits 0 bis 7: Bausteintyp • Bits 8 bis 15: reserviert
2 und 3	WORD	Bausteinlänge
4 und 5	WORD	Version: • Bits 0 bis 7: low byte • Bits 8 bis 15: high byte

Byte	Datentyp	Bedeutung
6 und 7	WORD	Kennung für den Alarmtyp: • 1: Diagnosealarm (kommend) • 2: Prozessalarm • 3: Ziehen-Alarm • 4: Stecken-Alarm • 5: Status-Alarm • 6: Update-Alarm • 7: Redundanz-Alarm • 8: Vom Supervisor gesteuert (Controlled_by_Supervisor) • 9: Vom Supervisor freigegeben (Released_by_Supervisor) • 10: Nicht die projektierte Baugruppe gesteckt • 11: Wiederkehr des Submoduls • 12: Diagnosealarm (gehend) • 13: Querverkehr-Verbindungsmeldung • 14: Nachbarschaftsänderungsmeldung • 15: Taktsynchronisationsmeldung (busseitig) • 16: Taktsynchronisationsmeldung (geräteseitig) • 17: Netzwerkkomponentenmeldung • 18: Uhrzeitsynchronisationsmeldung (busseitig) • 19 bis 31: Reserviert • 32 bis 127: Herstellerspezifischer Alarm • 128 bis 65535: Reserviert
8 bis 11	DWORD	API (Application Process Identifier)
12 bis 13	WORD	Steckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)
14 bis 15	WORD	Submodulsteckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)
16 bis 19	DWORD	Modul-Identifikation; eindeutige Information der Alarmquelle

Byte	Datentyp	Bedeutung
20 bis 23	DWORD	Submodul-Identifikation; eindeutige Information der Alarmquelle
24 bis 25	WORD	Alarm Specifier: • Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047) • Bit 11: Kanaldiagnose: 0: keine Kanaldiagnose vorhanden 1: Kanaldiagnose vorhanden • Bit 12: Status der herstellerspezifischen Diagnose: 0: keine herstellerspezifische Statusinformation vorhanden 1: Herstellerspezifische Statusinformation vorhanden • Bit 13: Status der Diagnose beim Submodul: 0: keine Statusinformation vorhanden, alle Fehler wurden beseitigt 1: Mindestens eine Kanaldiagnose und/oder Statusinformation vorhanden • Bit 14: reserviert • Bit 15: Application Relationship Diagnosis State: – 0: keine der innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose – 1: Mindestens eine innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose

Aufbau der Alarmzusatzinformation bei Alarmen von PROFINET IO

Die Alarmzusatzinformation bei PROFINET IO ist abhängig von der Formatkennung. Sie kann aus mehreren Datenblöcken mit gleicher oder unterschiedlicher Formatkennung bestehen. Folgende Formatkennungen sind möglich:

- W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose

Byte	Datentyp	Bedeutung
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose
2 bis n	BYTE	Siehe Handbuch des Herstellers.

- W#16#8000: Kanaldiagnose
  Die Kanaldiagnose wird in Blöcken zu jeweils 6 Byte ausgegeben. Die Alarmzusatzinformation (ohne Formatkennung) wird nur für die gestörten Kanäle ausgegeben.

Byte	Datentyp	Bedeutung	
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#8000: Kanaldiagnose	
2 bis 3	WORD	Kanalnummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535): • W#16#0000 bis W#16#7FFF: Kanalnummer des Submoduls/Moduls • W#16#8000: ist der Stellvertreter für das gesamte Submodul • W#16#8001 bis W#16#FFFF: reserviert	
4	BYTE	Bits 0 bis 2:	Reserviert
		Bits 3 bis 4:	Fehlerart: • 0: reserviert • 1: kommender Fehler • 2: gehender Fehler • 3: gehender Fehler, weitere Fehler vorhanden
		Bits 5 bis 7:	Kanalart: • 0: reserviert • 1: Eingabekanal • 2: Ausgabekanal • 3: Ein-/Ausgabekanal

Byte	Datentyp	Bedeutung
5	BYTE	Datenformat: • B#16#00: freies Datenformat • B#16#01: Bit • B#16#02: 2 Bit • B#16#03: 4 Bit • B#16#04: Byte • B#16#05: Wort • B#16#06: Doppelwort • B#16#07: 2 Doppelworte • B#16#08 bis B#16#FF: reserviert
6 bis 7	WORD	Fehlertyp: • W#16#0000: reserviert • W#16#0001: Kurzschluss • W#16#0002: Unterspannung • W#16#0003: Überspannung • W#16#0004: Überlast • W#16#0005: Übertemperatur • W#16#0006: Leitungsbruch • W#16#0007: Oberer Grenzwert überschritten • W#16#0008: Unterer Grenzwert überschritten • W#16#0009: Fehler • W#16#000A bis W#16#000F: reserviert • W#16#0010 bis W#16#001F: herstellerspezifisch • W#16#0020 bis W#16#00FF: reserviert • W#16#0100 bis W#16#7FFF: herstellerspezifisch • W#16#8000: Gerätediagnose liegt vor • W#16#8001 bis W#16#FFFF: reserviert Nicht jeder Kanal unterstützt jeden Fehlertyp. Details finden Sie in der Beschreibung der Diagnosedaten des entsprechenden Geräts.

Hinweis

Der Teil "Kanalnummer" bis "Fehlertyp kann 0 bis n mal auftreten.

W#16#8001

W#16#8001: MULTIPLE (unterschiedliche Arten von Diagnoseinformationen werden übertragen)

Die Alarmzusatzinformation wird in diesem Fall in Blockstruktur mit variabler Länge übertragen.

Byte	Datentyp	Bedeutung
0 bis 1	WORD	Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten: W#16#8001: Herstellerspezifische Diagnose und/oder Kanaldiagnose
2 bis 3	WORD	Bausteintyp
4 bis 5	WORD	Bausteinlänge
6	BYTE	Version: high byte
7	BYTE	Version: low byte
8 bis 11	DWORD	API (nur falls das low byte von Version = 1)
12 bis 13	WORD	Steckplatznummer
14 bis 15	WORD	Substeckplatznummer
16 bis 17	WORD	Kanalnummer
18 bis 19	WORD	Kanaleigenschaften
20 bis 21	WORD	Formatkennung: • W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose • W#16#8000: Kanaldiagnose • W#16#8002: Erweiterte Kanaldiagnose • W#16#8003: Abgestufte erweiterte Kanaldiagnose • W#16#8004 bis W#16#80FF: reserviert
22 bis n	BYTE	Daten abhängig von der Formatkennung

Hinweis

Der Teil ab "Bausteintyp" kann 1 bis n mal auftreten.

W#16#8002

W#16#8002: Erweiterte Kanaldiagnose

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8002
2 bis 3	Kanalnummer
4 bis 5	Kanaleigenschaften
6 bis 7	Fehlertyp
8 bis 9	Zusätzlicher Fehlerwert
10 bis 13	Zusätzliche Fehlerinformation

W#16#8003

W#16#8003: Abgestufte erweiterte Kanaldiagnose

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8003
2 bis 3	Kanalnummer
4 bis 5	Kanaleigenschaften
6 bis 7	Fehlertyp
8 bis 9	Zusätzlicher Fehlerwert
10 bis 13	Zusätzliche Fehlerinformation
14 bis 17	Qualified Channel Qualifier

W#16#8100

W#16#8100: Wartungsinformation

Byte	Bedeutung
0 bis 1	Formatkennung W#16#8100
2 bis 3	Bausteintyp
4 bis 5	Bausteinlänge
6 bis 7	Bausteinversion
8 bis 9	Reserviert
10 bis 13	Wartungszustand

Hinweis

Genauere Informationen zum Aufbau der Alarmzusatzinformation entnehmen Sie dem *Programmierhandbuch SIMATIC PROFINET IO Von PROFIBUS DP nach PROFINET IO* und dem aktuellen Stand der Norm IEC 61158-6-10-1.

Zielbereich TINFO und AINFO

Zielbereich TINFO und AINFO

Abhängig von dem jeweiligen OB, in dem die Anweisung "RALRM" aufgerufen wird, werden die Zielbereiche TINFO und AINFO nur teilweise beschrieben. Welche Information jeweils eingetragen wird, entnehmen Sie der folgenden Tabelle.

Alarmtyp	OB	TINFO OB-Status-information	TINFO Verwaltungs-information	AINFO Kopfinfor-ma-tion	AINFO Alarmzusatzinformation	
Prozessalarm	4x	ja	ja	ja	zentral:	nein
					dezentral:	wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Statusalarm	55	ja	ja	ja	ja	ja
Update-Alarm	56	ja	ja	ja	ja	ja
Herstellerspezifi-scher Alarm	57	ja	ja	ja	ja	ja
Peripherie-Redundanzfehler	70	ja	ja	nein	nein	nein
Diagnosealarm	82	ja	ja	ja	zentral:	Datensatz 1
					dezentral:	wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Ziehen-/Stecken-Alarm	83	ja	ja	ja	zentral:	Nein
					dezentral:	wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert
Sonderform des Zie-hen-Alarms: Vom Supervisor ge-steuert	83	ja	ja	ja	Nur PROFINET IO	
Sonderform des Ste-cken-Alarms: Vom Supervisor freige-geben	83	ja	ja	ja	Nur PROFINET IO	
Nicht projektierte Bau-gruppe gesteckt	83	ja	ja	ja	Nur PROFINET IO	
Baugruppen-träger-ausfall/ Stationsausfall	86	ja	ja	nein	nein	
... alle anderen OBs		ja	nein	nein	nein	

D_ACT_DP: DP-Slaves deaktivieren / aktivieren

Beschreibung

Mit der Anweisung "D_ACT_DP" können Sie projektierte DP-Slaves/PROFINET IO-Devices gezielt deaktivieren und wieder aktivieren. Darüber hinaus können Sie für jeden eingesetzten DP-Slave bzw. für jedes eingesetzte PROFINET IO-Device ermitteln, ob diese Komponente momentan aktiviert oder deaktiviert ist.

Falls Sie mit Hilfe von der Anweisung einen Netzübergang vom Typ IE/PB Link PN IO deaktivieren, fallen alle daran angeschlossenen PROFIBUS DP-Slaves aus. Diese Ausfälle werden gemeldet.

Die Anweisung ist nicht anwendbar auf PROFIBUS PA-Feldgeräte, die über DP/PA Link an ein DP-Mastersystem angeschlossen sind.

Hinweis

So lange ein oder mehrere "D_ACT_DP"-Aufträge aktiv sind, können Sie keine geänderte Konfiguration vom PG in die CPU laden (im Rahmen von CiR).
Während des Ladens einer geänderten Konfiguration vom PG in die CPU im laufenden Betrieb (CiR) weist die CPU die Aktivierung eines "D_ACT_DP"-Auftrags ab.

Für die Bearbeitung eines Deaktivier- bzw. Aktivierauftrags sind mehrere Durchläufe durch den Zykluskontrollpunkt notwendig. Sie können daher nicht in einer programmierten Schleife auf das Ende eines solchen Auftrags warten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "D_ACT_DP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Pegelgetriggerter Steuerparameter REQ=1: Aktivieren bzw. Deaktivieren durchführen
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Mögliche Werte: • 0: Auskunft einholen, ob die angesprochene Komponente aktiviert oder deaktiviert ist • 1: DP-Slave/PROFINET IO-Device aktivieren • 2: DP-Slave/PROFINET IO-Device deaktivieren • 3: DP-Slave/PROFINET IO-Device aktivieren und nach durchgeführtem Wechsel des Aktivierungszustands OB 86 aufrufen • 4: DP-Slave/PROFINET IO-Device deaktivieren und nach durchgeführtem Wechsel des Aktivierungszustands OB 86 aufrufen
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Beliebige logische Adresse des DP-Slaves/ PROFINET IO-Devices. Bei einer Ausgabeadresse muss das höchstwertige Bit gesetzt sein.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	Aktivkennung: • BUSY=1: Der Auftrag ist noch aktiv. • BUSY=0: Der Auftrag wurde beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
0001	DP-Slave/PROFINET IO-Device ist aktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)
0002	DP-Slave/PROFINET IO-Device ist deaktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)
7000	Erstaufruf mit REQ=0. Der über LADDR festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1. Der über LADDR festgelegte Auftrag wurde angestoßen; BUSY hat den Wert "1".

Fehlercode (W#16#...)	Erläuterung
7002	Zwischenaufruf (REQ irrelevant). Der aktivierte Auftrag ist noch in Bearbeitung; BUSY hat den Wert "1".
8090	• Sie haben keine Baugruppe mit der in LADDR angegebenen Adresse projektiert. • Sie betreiben Ihre CPU als I-Slave und haben in LADDR eine Adresse dieses I-Slaves angegeben.
8092	Der laufende Deaktiviervorgang eines DP-Slaves/PROFINET IO-Devices (MODE=2) kann nicht durch seine Aktivierung (MODE=1) abgebrochen werden. Aktivieren Sie die Komponente zu einem späteren Zeitpunkt.
8093	Zu der in LADDR angegebenen Adresse gehört kein DP-Slave/PROFINET IO-Device (Es liegt keine Projektierung vor.), oder der Parameter MODE ist nicht bekannt.
8094	Sie haben versucht, ein Gerät zu aktivieren, das ein möglicher Partner eines Werkzeugwechselports ist. Zu diesem Zeitpunkt ist an diesem Werkzeugwechselport aber bereits ein anderes Gerät aktiviert. Das aktivierte Gerät bleibt aktiviert.
80A1	Die angesprochene Komponente konnte nicht parametriert werden. (Dieser Fehlercode ist nur möglich bei MODE = 1.) Hinweis: Diese Fehlerinformation liefert "D_ACT_DP" nur dann, wenn während der Parametrierung des aktivierten Slaves/Devices diese Komponente wieder ausfällt. Wenn nur die Parametrierung einer einzelnen Baugruppe nicht erfolgreich war, liefert "D_ACT_DP" die Fehlerinformation W#16#0000.
80A2	Der angesprochene DP-Slave gibt keine Rückmeldung (Diese Fehlerinformation gibt es nicht bei PROFINET IO-Devices. Der Vorgang wird bei PROFINET nicht zeitüberwacht.).
80A3	Der betroffene DP-Master/PROFINET IO-Controller unterstützt diese Funktion nicht.
80A4	Die CPU unterstützt diese Funktion bei externen DP-Mastern/PROFINET IO-Controllern nicht.
80A6	Steckplatzfehler im DP-Slave/PROFINET IO-Device; es kann nicht auf alle Nutzdaten zugegriffen werden (Dieser Fehlercode ist nur möglich bei MODE=1). Hinweis: Diese Fehlerinformation liefert "D_ACT_DP" nur dann, wenn nach der Parametrierung der aktivierten Komponente und vor dem Ende von "D_ACT_DP" die Komponente wieder ausfällt. Wenn nur eine einzelne Baugruppe nicht verfügbar ist, liefert "D_ACT_DP" die Fehlerinformation W#16#0000.
80C1	"D_ACT_DP" wurde gestartet und wird mit einer anderen logischen Adresse fortgesetzt (Dieser Fehlercode ist möglich bei MODE=1 und bei MODE=2).
80C3	• Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an Aktivierungs-/Deaktivierungsaufträgen. (Dieser Fehlercode ist nur möglich bei MODE = 1 und MODE = 2.) • Die CPU erhält gerade eine geänderte Konfiguration. Das Deaktivieren und Aktivieren von DP-Slaves/PROFINET IO-Devices ist daher momentan nicht möglich.
80C5	DP: Vom Anwender noch nicht abgeholte Aufträge werden durch Wiederanlauf verworfen.
80C6	PROFINET: Vom Anwender noch nicht abgeholte Aufträge werden durch Wiederanlauf verworfen.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anwendungsbereich

Wenn Sie in einer CPU DP-Slaves/PROFINET IO-Devices konfigurieren, die real nicht vorhanden sind oder aktuell nicht benötigt werden, greift die CPU dennoch regelmäßig auf diese DP-Slaves/PROFINET IO Devices zu. Nach deren Deaktivierung unterbleiben weitere CPU-Zugriffe. Bei PROFIBUS DP kann dadurch der schnellstmögliche DP-Buszyklus erreicht werden, und die entsprechenden Fehlerereignisse treten nicht mehr auf.

Beispiele

Im Serienmaschinenbau ist aus Sicht des Maschinenbauers eine Vielzahl von Maschinenoptionen möglich. Jede ausgelieferte Maschine beinhaltet jedoch nur eine individuelle Kombination ausgewählter Optionen.

Sämtliche möglichen Maschinenoptionen sind vom Hersteller als DP-Slaves/PROFINET IO Devices projektiert, um ein gemeinsames Anwenderprogramm über alle möglichen Optionen erstellen und pflegen zu können. Mit "D_ACT_DP" können Sie im Anlauf der Maschine alle nicht vorhandenen DP-Slaves/PROFINET IO Devices deaktivieren.

Eine ähnliche Situation besteht bei Werkzeugmaschinen, die viele Werkzeuge vorhalten, aber zu einem Zeitpunkt nur wenige einsetzen. Hier werden die Werkzeuge als DP-Slaves/ PROFINET IO Devices realisiert. Das Anwenderprogramm aktiviert mit Hilfe von "D_ACT_DP" die aktuell benötigten und deaktiviert die erst später wieder einzusetzenden Werkzeuge.

Arbeitsweise

"D_ACT_DP" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "D_ACT_DP" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Identifikation eines Auftrags

Falls Sie einen Deaktivierungs- bzw. Aktivierungsauftrag angestoßen haben und Sie "D_ACT_DP" erneut aufrufen, bevor dieser beendet wurde, hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Stimmt der Eingangsparameter LADDR überein, so gilt der Aufruf als Folgeaufruf.

Deaktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO Device mit "D_ACT_DP" deaktivieren, werden dadurch dessen Prozessausgänge auf die projektierten Ersatzwerte bzw. auf 0 gesetzt (sicherer Zustand). Der zugehörige DP-Master/PROFINET IO Controller spricht diese Komponente im Weiteren nicht mehr an. Deaktivierte DP-Slaves/PROFINET IO-Devices werden an den Fehler-LEDs des DP-Masters/PROFINET IO-Controllers oder der CPU nicht als gestört oder fehlend gekennzeichnet.

Das Prozessabbild der Eingänge von deaktivierten DP-Slaves/PROFINET IO-Devices wird mit 0 aktualisiert, d. h. es wird wie bei ausgefallenen DP-Slaves/PROFINET IO-Devices behandelt.

Falls Sie in Ihrem Programm mittels Direktzugriff auf die Nutzdaten eines zuvor deaktivierten DP-Slaves/PROFINET IO-Devices zugreifen, wird der Peripheriezugriffsfehler-OB (OB 122) aufgerufen und das zugehörige Startereignis in den Diagnosepuffer eingetragen. Falls Sie per Anweisung (z. B. "RD_REC (Seite 5490)") auf einen deaktivierten DP-Slave bzw. ein deaktiviertes PROFINET IO-Device zugreifen, erhalten Sie in RET_VAL dieselbe Fehlerinformation wie bei einem nicht verfügbaren DP-Slave/PROFINET IO-Device.

Das Deaktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OB (OB 85), auch wenn dessen Ein- bzw. Ausgänge zum systemseitig

zu aktualisierenden Prozessabbild gehören. Es erfolgt auch kein Eintrag in den Diagnosepuffer.

Ob das Deaktivieren eines DP-Slaves/PROFINET IO-Devices einen Start des Baugruppenträgerausfall-OB (OB 86) und einen Eintrag in den Diagnosepuffer zur Folge hat, ist vom Parameter MODE abhängig:

- MODE = 2: Kein OB 86-Start und kein Diagnosepuffereintrag
- MODE = 4: OB 86-Start und Diagnosepuffereintrag

Fällt eine DP-Station/PNIO-Station aus, nachdem Sie sie mit "D_ACT_DP" deaktiviert haben, wird der Ausfall vom Betriebssystem nicht erkannt. Es erfolgt daher weder ein OB 86-Start noch ein Diagnosepuffereintrag.

Bei PROFIBUS DP gilt: Falls Sie DP-Slaves deaktivieren wollen, die als Sender am Querverkehr beteiligt sind, wird empfohlen, zuerst die Empfänger (Mithörer) zu deaktivieren, die mithören, welche Eingangsdaten der Sender seinem DP-Master schickt. Erst im Anschluss daran deaktivieren Sie den Sender.

Aktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO-Device mit "D_ACT_DP" wieder aktivieren, wird diese Komponente vom zugehörigen DP-Master/PROFINET IO-Controller konfiguriert und parametriert (wie bei der Wiederkehr einer ausgefallenen DP-Station/PROFINET IO-Station). Die Aktivierung ist abgeschlossen, wenn die Komponente Nutzdaten transferieren kann.

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OB (OB 85), auch wenn dessen Ein- bzw. Ausgänge zum systemseitig zu aktualisierenden Prozessabbild gehören. Es erfolgt auch kein Eintrag in den Diagnosepuffer.

Ob das Deaktivieren eines DP-Slaves/PROFINET IO-Devices einen Start des Baugruppenträgerausfall-OB (OB 86) und einen Eintrag in den Diagnosepuffer zur Folge hat, ist vom Parameter MODE abhängig:

- MODE = 1: Kein OB 86-Start und kein Diagnosepuffereintrag
- MODE = 3: OB 86-Start und Diagnosepuffereintrag

Wenn Sie versuchen, einen deaktivierten DP-Slave, der physikalisch vom DP-Bus getrennt ist, mit "D_ACT_DP" zu aktivieren, liefert die Anweisung nach ca. einer Minute den Fehlercode W#16#80A2, und der DP-Slave bleibt deaktiviert. Falls er zu einem späteren Zeitpunkt wieder Verbindung zum DP-Bus hat, müssen Sie den Slave mittels "D_ACT_DP" wieder aktivieren.

Wenn Sie versuchen, ein PROFINET IO-Device zu aktivieren, das physikalisch vom PN-Bus getrennt ist, bleibt "D_ACT_DP" tätig. Es erfolgt kein automatischer Abbruch nach einer gewissen Zeit wie bei DP-Slaves. Sie müssen den laufenden Auftrag selbst abbrechen.

Hinweis

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices kann geraume Zeit dauern. Falls Sie einen laufenden Aktivierungsauftrag abbrechen wollen, starten Sie "D_ACT_DP" mit dem gleichen Wert für LADDR und MODE = 2. Sie wiederholen den Aufruf von "D_ACT_DP" mit MODE = 2 so lange, bis der erfolgreiche Abbruch des Aktivierungsauftrags mit RET_VAL = 0 angezeigt wird.

Falls Sie DP-Slaves aktivieren wollen, die am Querverkehr beteiligt sind, wird empfohlen, zuerst die Sender und anschließend die Empfänger (Mithörer) zu aktivieren.

CPU-Anlauf

Das Betriebssystem der CPU zeigt in den verschiedenen Anlaufarten folgendes Verhalten hinsichtlich der DP-Slaves/PROFINET IO-Devices:

- In den Anlaufarten Kaltstart und Neustart (Warmstart) werden deaktivierte Slaves/Devices automatisch wieder aktiviert.
  - Bei S7-400: Das Aktivieren der DP-Slaves/PROFINET IO-Devices kann längere Zeit dauern. In diesem Fall läuft die CPU trotzdem an, und bis zur fertigen Aktivierung treten Peripheriezugriffsfehler auf. Die Reaktion der CPU ist abhängig von der Parametrierung (OB 85-Aufruf bei Peripheriezugriffsfehler) und dem Programm im OB 85. Sie können diese Peripheriezugriffsfehler nicht unterdrücken.
  - Bei S7-300: Das Aktivieren der DP-Slaves/PROFINET IO-Devices kann längere Zeit dauern. In diesem Fall wartet die CPU, bis die DP-Slaves / PROFINET IO-Devices aktiviert wurden. Die maximale Wartezeit für das Aktivieren der DP-Slaves / PROFINET IO-Devices beträgt ca. 1 Minute. Innerhalb dieser Zeit treten keine Peripheriezugriffsfehler auf.
- In der Anlaufart Wiederanlauf bleibt der Aktivierungsstatus von Slaves/Devices unverändert: Aktivierte Slaves/Devices bleiben aktiviert, deaktivierte Slaves/Devices bleiben deaktiviert.

Nach dem CPU-Anlauf versucht die CPU zyklisch zu allen projektierten und nicht deaktivierten Slaves/Devices, die nicht vorhanden oder nicht ansprechbar sind, Kontakt aufzunehmen.

Hinweis

Ein Aufruf von "D_ACT_DP" in den Anlauf-OBs wird nicht unterstützt.

Weitere Anweisungen

RD_REC: Datensatz von Peripherie lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM von der adressierten Baugruppe. Sie starten den Lesevorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Lesevorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Lesevorgang noch nicht abgeschlossen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370). Der gelesene Datensatz wird nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen.

Hinweis

Falls Sie einen Datensatz, dessen Nummer größer als eins ist, aus einer FM oder einem CP auslesen, die Sie vor Februar 1997 erworben haben (im Folgenden als "alte Baugruppen" bezeichnet), dann verhält sich "RD_REC" anders als bei einer neuen Baugruppe. Dieser Sonderfall wird im Abschnitt "Verwendung alter S7-300-FMs und -CPs bei DS-Nr. >1" (siehe unten) behandelt.

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "RD_REC" von den E/A-Baugruppen gelesen werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_REC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Lesen
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Adresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 0 bis 240)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Zusätzlich: Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz und bei der Übertragung kein Fehler auftrat.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.
RECORD	Output	ANY	E, A, M, D, L	Zielbereich für den gelesenen Datensatz. Sie müssen bei asynchroner Bearbeitung von "RD_REC" darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen dieselbe Längeninformation haben. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RECORD

Hinweis

Wenn Sie sicherstellen wollen, dass immer der gesamte Datensatz gelesen wird, wählen Sie einen Zielbereich mit der Länge 241 Bytes. Bei einer fehlerfreien Datenübertragung steht dann in RET_VAL die tatsächliche Datensatzlänge.

Verwendung alter S7-300-FMs und -CPs bei DS-Nr. > 1

Falls Sie mit der Anweisung "RD_REC" aus einer alten S7-300-FM oder einem alten S7-300-CP einen Datensatz auslesen wollen, dessen Nummer größer als eins ist, müssen Sie Folgendes beachten:

- Falls der Zielbereich größer als die tatsächliche Länge des gewünschten Datensatzes ist, werden keine Daten in RECORD eingetragen. RET_VAL wird mit W#16#80B1 beschrieben.
- Falls der Zielbereich kleiner als die tatsächliche Länge des gewünschten Datensatzes ist, liest die CPU so viele Bytes ab Datensatzanfang, wie in der Längenangabe von RECORD angegeben ist, und trägt sie in RECORD ein. RET_VAL wird mit "0" beschrieben.
- Falls die Längenangabe von RECORD gleich der tatsächlichen Länge des gewünschten Datensatzes ist, liest die CPU den gewünschten Datensatz und trägt ihn in RECORD ein, RET_VAL wird mit "0" beschrieben.

Parameter RET_VAL

- Trat während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
- Falls bei der Übertragung kein Fehler auftrat, enthält RET_VAL
  - 0, falls der gesamte Zielbereich mit Daten aus dem selektierten Datensatz gefüllt wurde (Der Datensatz kann aber unvollständig sein.).
  - die Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz.

  Hinweis

  Falls der allgemeine Fehler W#16#8745 auftritt, zeigt dies nur an, dass beim Schreibvorgang auf das Prozessabbild der Zugriff auf mindestens ein Byte nicht möglich war. Der Datensatz wurde ordnungsgemäß von der Baugruppe gelesen und in den E/A-Speicherbereich geschrieben.

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) der folgenden Tabelle sind zwei Fälle zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A4, 80Cx):
  Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt, d. h. es ist sinnvoll, dass Sie die Anweisung erneut aufrufen (ggf. mehrfach).
  Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#809x, 80A0, 80A1, 80Bx):
  Bei dieser Fehlerart kann sich der Fehler nicht ohne Ihr Zutun beheben. Ein erneuter Aufruf der Anweisung ist erst sinnvoll, wenn Sie den Fehler beseitigt haben. Beispiel für einen permanenten Fehler: Falsche Längenangabe in RECORD (W#16#80B1).

  Hinweis

  Wenn Sie mit "WR_REC (Seite 5494)" Datensätze zu einem DPV1-Slave übertragen oder mit RD_REC Datensätze von einem DPV1-Slave lesen und dieser im DPV1-Mode arbeitet, bewertet der DP-Master die vom Slave erhaltene Fehlerinformation wie folgt:

  Liegt die Fehlerinformation in den Bereichen W#16#8000 bis W#16#80FF oder W#16#F000 bis W#16#FFFF, reicht der DP-Master die Fehlerinformation an die Anweisung weiter. Liegt sie außerhalb, gibt er den Wert W#16#80A2 an die Anweisung und suspendiert den Slave.

  Zur Beschreibung der von DPV1-Slaves stammenden Fehlerinformationen siehe STATUS[3] Parameter STATUS (Seite 5468).

Parameter RET_VAL für WR_REC und RD_REC

Fehlercode (W#16#...)	Erläuterung	Einschränkung
0000	kein Fehler	-
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.	-

Fehlercode (W#16#...)	Erläuterung	Einschränkung
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.	Dezentrale Peripherie
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert 1.	Dezentrale Peripherie
8090	Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.	-
8092	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.	Nur bei S7-400
8093	Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig. (Zulässig sind S7-300-Baugruppen bei S7-300, S7-400-Baugruppen bei S7-400, S7-DP-Baugruppen bei S7-300 und S7-400.)	-
80A0	Negative Quittung beim Lesen von Baugruppe: Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt	nur bei "RD_REC"
80A1	Negative Quittung beim Schreiben zur Baugruppe: Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt	nur bei "WR_REC (Seite 5494)"
80A2	• DP-Protokollfehler bei Layer2 (z. B. Slaveausfall oder Busstörungen) • Bei ET200S ist im DPV0-Mode kein Datensatz-Lesen möglich.	Dezentrale Peripherie
80A3	DP-Protokollfehler bei User-Interface/User	Dezentrale Peripherie
80A4	Kommunikation am K-Bus gestört	Fehler tritt auf zwischen CPU und externer DP-Anschaltung
80B0	• Anweisung für Baugruppentyp nicht möglich • Baugruppe kennt den Datensatz nicht. • Datensatznummer 241 ist unzulässig. • Bei "WR_REC (Seite 5494)" sind die Datensätze 0 und 1 nicht erlaubt.	-
80B1	Die Längenangabe in Parameter RECORD ist falsch.	• Bei "WR_REC (Seite 5494)": Länge falsch • bei "RD_REC" (nur möglich bei Verwendung alter S7-300-FMs und S7-300-CPs):Angabe > DS-Länge • bei DPNRM_DG: Angabe < DS-Länge
80B2	Der projektierte Steckplatz ist nicht belegt.	-
80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp im SDB1	-
80B7	DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Wertes.	nur bei "RD_REC"

Fehlercode (W#16#...)	Erläuterung	Einschränkung
80C0	Bei "WR_REC (Seite 5494)": Die Daten können nur im STOP-Zustand der CPU geschrieben werden. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Sie können die Daten nur online mit PG/PC schreiben. Bei "RD_REC": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. In diesem Fall können Sie die Daten nur online mit PG/PC lesen. Bei "DPNRM_DG (Seite 5522)": Es liegen keine Diagnosedaten vor.	Bei "WR_REC (Seite 5494)" oder "RD_REC" oder "DPNRM_DG (Seite 5522)"
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.	-
80C2	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.	-
80C3	Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.	-
80C4	Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.	-
80C5	Dezentrale Peripherie nicht verfügbar.	Dezentrale Peripherie
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)	Dezentrale Peripherie
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)	-

WR_REC: Datensatz in Peripherie schreiben

Beschreibung

Mit der Anweisung "WR_REC" übertragen Sie den Datensatz RECORD zu der adressierten Baugruppe.

Sie starten den Schreibvorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Schreibvorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Schreibvorgang noch nicht abgeschlossen.

Hinweis

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "WR_REC" zu den E/A-Baugruppen geschrieben werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_REC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Schreiben
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Adresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 2 bis 240)
RECORD	Input	ANY	E, A, M, D, L	Datensatz. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
RET_VAL	Output	Return	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RECORD

Die zu übertragenden Daten werden aus dem Parameter RECORD beim ersten Aufruf gelesen. Falls die Übertragung des Datensatzes länger als einen Aufruf dauert, so ist bei den Folgeaufrufen der Anweisung (zum gleichen Auftrag) der Inhalt des Parameters RECORD nicht mehr relevant.

Parameter RET_VAL

Siehe auch: RD_REC: Datensatz von Peripherie lesen (Seite 5490)

Hinweis

Falls der allgemeine Fehler W#16#8544 auftritt, zeigt dies nur an, dass der Zugriff auf mindestens ein Byte des den Datensatz enthaltenden E/A-Speicherbereichs gesperrt war. Die Datenübertragung wurde fortgesetzt.

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen

Beschreibung

Mit der Anweisung "DPRD_DAT" lesen Sie konsistente Daten eines DP-Normslaves/ PROFINET IO-Devices aus, wobei für die Maximallänge Folgendes gilt:

- Die Maximallänge entnehmen Sie der Dokumentation Ihrer CPU.
- Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den durch RECORD aufgespannten Zielbereich eingetragen.

Der Zielbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Baugruppe projektiert haben. Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "DPRD_DAT"-Aufruf jeweils nur auf die Daten einer Baugruppe/ DP-Kennung unter der projektierten Anfangsadresse zugreifen.

Hinweis

Eine CPU der Baureihen S7-300/400 unterstützt bis zu 64 Bytes konsistente Daten. Bei Bereichen konsistenter Daten größer als 4 Bytes müssen Sie die Anweisung "DPRD_DAT" verwenden. Diese Anweisung kann bei Bedarf auch für einen Datenbereich ab 1 Byte verwendet werden. Bei einem Zugriffsfehler wird der Fehlercode W#16#8090 gemeldet.

Hinweis

S7-400-CPUs ab Firmware-Stand V6.0

Bei S7-400-CPUs ab Firmware-Stand V6.0 wird der Aufruf von DPRD_DAT grundsätzlich für alle Adressen akzeptiert. Sie können also auf eigene Verantwortung mit DPRD_DAT auf Adressen zugreifen, für die Sie keine Gesamtkonsistenz projektiert haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPRD_DAT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Projektierte Anfangsadresse aus dem E-Bereich der Baugruppe, aus der gelesen werden soll. Hinweis: Adresse muss hexadezimal angegeben werden. Z. B. Anfangsadresse 100 bedeutet: LADDR:=W#16#64.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
RECORD	Output	ANY	E, A, M, D, L	Zielbereich für die gelesenen Nutzdaten. Er muss genauso lang sein, wie Sie für die selektierte Baugruppe projektiert haben. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Hinweis

Falls Sie auf DPV1-Slaves zugreifen, können Fehlerinformationen dieser Slaves vom DP-Master an die Anweisung weitergereicht werden. Zur Beschreibung dieser Fehlerinformationen siehe STATUS[3] Parameter STATUS (Seite 5468).

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	• Für die angegebene logische Basisadresse haben sie keine Baugruppe projektiert. • Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet. (Bei S7-400-CPUs nur bis einschließlich Firmware-Stand V5.3; ab Firmware-Stand V6.0 entfällt diese Ursache.) • Sie haben die Anfangsadresse im Parameter LADDR nicht hexadezimal angegeben.
8092	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.
8093	Für die unter LADDR angegebene logische Adresse existiert keine DP-Baugruppe/kein PROFINET IO-Device, von der/dem Sie konsistente Daten lesen können.

Fehlercode (W#16#...)	Erläuterung
80A0	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt
80B0	• Bei S7-300-CPUs (alle Firmware-Stände) und S7-400-CPUs bis einschließlich Firmware-Stand V5.3: Slaveausfall an externer DP-Anschaltung • Bei S7-400-CPUs ab Firmware-Stand V6.0: Slaveausfall oder Zugriffsfehler auf nicht-konsistente Daten an externer DP-Anschaltung (Dass ein Fehler aufgetreten ist, wird der CPU durch den CP gemeldet.)
80B1	Die Länge des angegebenen Zielbereichs ist ungleich der projektierten Nutzdatenlänge.
80B2	Systemfehler bei externer DP-Anschaltung
80B3	Systemfehler bei externer DP-Anschaltung
80C0	Die Daten wurden noch nicht von der Baugruppe gelesen
80C2	Systemfehler bei externer DP-Anschaltung
80Fx	Systemfehler bei externer DP-Anschaltung
87xy	Systemfehler bei externer DP-Anschaltung
808x	Systemfehler bei externer DP-Anschaltung
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anwendungsbereich

Sie benötigen "DPRD_DAT", weil Sie mit den Ladebefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Eingänge zugreifen, maximal vier Bytes zusammenhängend auslesen können.

Hinweis

Sie können konsistente Daten ggf. auch über das Prozessabbild der Eingänge einlesen.

- Ob Ihre S7-300-CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation.
- Alle S7-400-CPUs beherrschen diese Funktionalität.

⚠ VORSICHT

Peripheriezugriff

Vermeiden Sie bei Verwendung von "DPRD_DAT" Zugriffe auf Peripheriebereiche, denen Teilprozessabbilder mit OB6x-Anbindung (Taktsynchronalarme) zugeordnet sind.

Datenkonsistenz

Siehe auch: Abschnitt Datenkonsistenz (Seite 6383).

DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben

Beschreibung

Mit der Anweisung übertragen Sie die Daten in RECORD konsistent zum adressierten DP-Normslave / PROFINET IO-Device und ggf. ins Prozessabbild (nämlich dann, wenn Sie den betroffenen Adressbereich des DP-Normslaves als Konsistenzbereich in einem Prozessabbild projektiert haben).

Für die Maximallänge der zu übertragenden Daten gilt: Die Maximallänge entnehmen Sie der Dokumentation Ihrer CPU. Die Datenübertragung erfolgt synchron, d. h. nach Beendigung der Anweisung ist der Schreibvorgang abgeschlossen.

Der Quellbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Baugruppe projektiert haben. Bei einem DP-Normslave mit modularem Aufbau können Sie nur auf eine Baugruppe des DP-Slaves zugreifen.

Hinweis

Eine CPU der Baureihen S7-1200/1500 unterstützt bis zu 64 Bytes konsistente Daten. Bei Bereichen konsistenter Daten größer als 4 Bytes müssen Sie die Daten über die Anweisung "DPWR_DAT" in ein Modul oder Gerät schreiben. Diese Anweisung kann bei Bedarf auch für einen Datenbereich ab 1 Byte verwendet werden. Bei einem Zugriffsfehler wird der Fehlercode W#16#8090 gemeldet.

Hinweis

S7-400-CPUs ab Firmware-Stand V6.0

Bei S7-400-CPUs ab Firmware-Stand V6.0 wird der Aufruf von DPWR_DAT grundsätzlich für alle Adressen akzeptiert. Sie können also auf eigene Verantwortung mit DPWR_DAT auf Adressen zugreifen, für die Sie keine Gesamtkonsistenz projektiert haben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPWR_DAT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Projektierte Anfangsadresse aus dem PAA-Bereich der Baugruppe, auf die geschrieben werden soll. Hinweis: Adresse muss hexadezimal angegeben werden. Z. B. Anfangsadresse 100 bedeutet: LADDR:=W#16#64.
RECORD	Input	ANY	E, A, M, D, L	Quellbereich für die zu schreibenden Nutzdaten. Er muss genauso lang sein, wie Sie für die selektierte Baugruppe projektiert haben. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Hinweis

Falls Sie auf DPV1-Slaves zugreifen, können Fehlerinformationen dieser Slaves vom DP-Master an die Anweisung weitergereicht werden. Zur Beschreibung dieser Fehlerinformationen siehe STATUS[3] Parameter STATUS (Seite 5468).

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
808x	Systemfehler bei externer DP-Anschaltung
8090	• Für die angegebene logische Basisadresse haben sie keine Baugruppe projektiert. • Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet. (Bei S7-400-CPUs nur bis einschließlich Firmware-Stand V5.3; ab Firmware-Stand V6.0 entfällt diese Ursache.) • Sie haben die Anfangsadresse im Parameter LADDR nicht hexadezimal angegeben.
8092	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.
8093	Für die unter LADDR angegebene logische Adresse existiert keine DP-Baugruppe / kein PROFINET IO-Device, auf die/das Sie konsistente Daten schreiben können.
80A1	Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt

Fehlercode (W#16#...)	Erläuterung
80B0	• Bei S7-300-CPUs (alle Firmware-Stände) und S7-400-CPUs bis einschließlich Firmware-Stand V5.3: Slaveausfall an externer DP-Anschaltung • Bei S7-400-CPUs ab Firmware-Stand V6.0: Slaveausfall oder Zugriffsfehler auf nicht-konsistente Daten an externer DP-Anschaltung (Dass ein Fehler aufgetreten ist, wird der CPU durch den CP gemeldet.)
80B1	Die Länge des angegebenen Quellbereichs ist ungleich der projektierten Nutzdatenlänge.
80B2	Systemfehler bei externer DP-Anschaltung
80B3	Systemfehler bei externer DP-Anschaltung
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags sind von der Baugruppe noch nicht bearbeitet.
80C2	Systemfehler bei externer DP-Anschaltung
80Fx	Systemfehler bei externer DP-Anschaltung
85xy	Systemfehler bei externer DP-Anschaltung
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Anwendungsbereich

Sie benötigen die Anweisung "DPWR_DAT", weil Sie mit den Transferbefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Ausgänge zugreifen, maximal vier Bytes zusammenhängend schreiben können.

Hinweis

Sie können konsistente Daten ggf. auch über das Prozessabbild der Ausgänge schreiben. Ob Ihre S7-300-CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation.

- Alle S7-400-CPUs beherrschen diese Funktionalität.
- Nutzen Sie beim Schreiben konsistenter Daten nicht beide Möglichkeiten gleichzeitig: Nutzen Sie entweder "DPWR_DAT" oder das Schreiben über das Prozessabbild der Ausgänge.

⚠ VORSICHT

Peripherizugriff

Vermeiden Sie bei Verwendung von "DPWR_DAT" Zugriffe auf Peripheriebereiche, denen Teilprozessabbilder mit OB6x-Anbindung (Taktsynchronalarme) zugeordnet sind.

Datenkonsistenz

Siehe auch: Datenkonsistenz (Seite 6383).

iDevice / iSlave

RCVREC: Datensatz empfangen

Beschreibung

Ein I-Device kann vom übergeordneten Controller einen Datensatz empfangen. Der Empfang erfolgt im Anwendungsprogramm mit der Anweisung "RCVREC" (receive record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für einen Datensatzempfang vorliegt.
- Bereitstellen des Datensatzes an den Ausgangsparametern.
- Verschicken einer Antwort am den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).

Das I-Device muss sich im Betriebszustand RUN oder im Betriebszustand ANLAUF befinden.

Mit MLEN geben Sie vor, wie viele Bytes Sie maximal empfangen möchten. Den Zielbereich RECORD sollten Sie daher mindestens MLEN Bytes lang wählen.

Wenn ein Datensatz empfangen wurde (MODE=1 oder MODE=2), wird im Ausgangsparameter NEW angezeigt, dass der Datensatz in RECORD gespeichert wurde. Achten Sie dabei auf eine ausreichende Länge von RECORD. Der Ausgangsparameter LEN enthält die tatsächliche Länge des empfangenen Datensatzes in Bytes.

Für die positive Antwort an den übergeordneten Controller beschreiben Sie CODE1 und CODE2 mit Null. Wenn der empfangene Datensatz abgelehnt werden soll, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

Hinweis

Wenn das I-Device eine Anforderung für einen Datensatzempfang erhalten hat, müssen Sie die Zustellung dieser Anforderung innerhalb einer bestimmten Zeitdauer erkannt haben. Nach dem Erkennen müssen Sie dem übergeordneten Controller innerhalb derselben Zeitdauer eine Antwort geschickt haben. Andernfalls tritt beim I-Device ein Timeout-Fehler auf, der zur Folge hat, dass das Betriebssystem des I-Device eine negative Antwort an den übergeordneten Controller schickt. Den Wert für die Zeitdauer entnehmen Sie den Technischen Daten Ihrer CPU.

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

Betriebsarten

Die Betriebsart der Anweisung "RCVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

MODE	Bedeutung
0	Überprüfen, ob eine Anforderung für einen Datensatzempfang vorliegt Falls auf dem I-Device ein Datensatz vom übergeordneten Controller vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, INDEX und LEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung.
1	Empfangen eines Datensatzes für irgendeinen Subslot des I-Device Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.
2	Empfangen eines Datensatzes für einen bestimmten Subslot des I-Device Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.
3	Positive Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, akzeptiert den vorliegenden Datensatz und schickt eine positive Quittung an den übergeordneten Controller.
4	Negative Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, lehnt den vorliegenden Datensatz ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.

Hinweis

Nach dem Eintreffen eines Datensatzes (NEW=1) müssen Sie "RCVREC" noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2
- Zweiter Aufruf mit MODE=3 oder MODE=4

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCVREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	DWORD	E, A, M, D, L oder Konstante	Subslot im Transferbereich des I-Device für den zu erhaltenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen.
MLEN	Input	INT	E, A, M, D, L oder Konstante	Maximale Länge des zu empfangenden Datensatzes in Bytes.
CODE1	Input	BYTE	E, A, M, D, L oder Konstante	Null (bei MODE=3) bzw. Error Code 1 (bei MODE=4)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CODE2	Input	BYTE	E, A, M, D, L oder Konstante	Null (bei MODE=3) bzw. Error Code 2 (bei MODE=4)
NEW	Output	BOOL	E, A, M, D, L	• MODE=0: Neuer Datensatz wurde empfangen • MODE=1 oder 2: Datensatz wurde in RECORD übertragen
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation
SLOT	Output	INT	E, A, M, D, L	identisch zu F_ID
SUBSLOT	Output	INT	E, A, M, D, L	identisch zu F_ID
INDEX	Output	INT	E, A, M, D, L	Nummer des empfangenen Datensatzes
LEN	Output	INT	E, A, M, D, L	Länge des empfangenen Datensatzes
RECORD	InOut	ANY	E, A, M, D, L	Zielbereich für den empfangenen Datensatz. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 5468)

PRVREC: Datensatz bereitstellen

Beschreibung

Ein I-Device kann vom übergeordneten Controller eine Anforderung erhalten, einen Datensatz bereitzustellen. Die Bereitstellung erfolgt im Anwenderprogramm mit der Anweisung "PRVREC" (provide record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für eine Datensatzbereitstellung vorliegt.
- Übertragung des angeforderten Datensatz an den übergeordneten Controller.
- Schicken einer Antwort an den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).

Das I-Device muss sich im Betriebszustand RUN oder im Betriebszustand ANLAUF befinden.

Mit LEN geben Sie vor, wie viele Bytes der zu versendende Datensatz maximal haben soll. Den Zielbereich RECORD sollten Sie daher mindestens LEN Bytes lang wählen.

Wenn eine Anforderung für eine Datensatzbereitstellung vorliegt (MODE=0), wird der Ausgangsparameter NEW auf TRUE gesetzt.

Wenn die Anforderung für eine Datensatzbereitstellung akzeptiert wird, beschreiben Sie für die positive Antwort an den übergeordneten Controller RECORD mit dem angeforderten Datensatz und CODE1 und CODE2 mit Null. Wenn die Anforderung für eine Datensatzbereitstellung abgelehnt werden soll, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

Hinweis

Wenn das I-Device eine Anforderung für eine Datensatzbereitstellung erhalten hat, müssen Sie die Zustellung dieser Anforderung innerhalb einer bestimmten Zeitdauer erkannt haben. Nach dem Erkennen müssen Sie dem übergeordneten Controller innerhalb derselben Zeitdauer eine Antwort geschickt haben. Andernfalls tritt beim I-Device ein Timeout-Fehler auf, der zur Folge hat, dass das Betriebssystem des I-Device eine negative Antwort an den übergeordneten Controller schickt. Den Wert für die Zeitdauer entnehmen Sie den Technischen Daten Ihrer CPU.

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

Betriebsarten

Die Betriebsart der Anweisung "PRVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

MODE	Bedeutung
0	Überprüfen, ob eine Anforderung für eine Datensatzbereitstellung vorliegt Falls auf dem I-Device eine Anforderung vom übergeordneten Controller für eine Datensatzbereitstellung vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, INDEX und RLEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung.
1	Empfangen einer Anforderung für eine Datensatzbereitstellung für irgendeinen Subslot des I-Device Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.
2	Empfangen einer Anforderung für eine Datensatzbereitstellung für einen bestimmten Subslot des I-Device Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.
3	Datensatz bereitstellen und positive Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, stellt den angeforderten Datensatz in RECORD zur Verfügung und schickt eine positive Quittung an den übergeordneten Controller.
4	Negative Antwort an den übergeordneten Controller schicken Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, lehnt diese Anforderung ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.

Hinweis

Nach dem Eintreffen einer Anforderung (NEW=1) müssen Sie die Anweisung noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2
- Zweiter Aufruf mit MODE=3 oder MODE=4

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PRVREC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Betriebsart
F_ID	Input	DWORD	E, A, M, D, L oder Konstante	Subslot im Transferbereich des I-Device für den zu versendenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen.
CODE1	Input	BYTE	E, A, M, D, L oder Konstante	Null (bei MODE=3) bzw. Error Code 1 (bei MODE=4)
CODE2	Input	BYTE	E, A, M, D, L oder Konstante	Null (bei MODE=3) bzw. Error Code 2 (bei MODE=4)
LEN	Input	INT	E, A, M, D, L oder Konstante	Maximale Länge des zu versendenden Datensatzes in Bytes
NEW	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde vom übergeordneten Controller angefordert.
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation
SLOT	Output	INT	E, A, M, D, L	identisch zu F_ID
SUBSLOT	Output	INT	E, A, M, D, L	identisch zu F_ID
INDEX	Output	INT	E, A, M, D, L	Nummer des zu versendenden Datensatzes
RLEN	Output	INT	E, A, M, D, L	Länge des zu versendenden Datensatzes
RECORD	InOut	ANY	E, A, M, D, L	Bereitgestellter Datensatz Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 5468)

SALRM: Alarm senden

Beschreibung SALRM

Beschreibung

Mit der Anweisung senden Sie aus dem Anwenderprogramm eines intelligenten Slaves einen Alarm eines Steckplatzes im Übergabebereich (virtueller Steckplatz) an den zugehörigen DP-Master. Das führt zum Start des zugehörigen OB beim DP-Master.

Dem Alarm können Sie alarmspezifische Zusatzinformationen mitgeben. Die gesamte Zusatzinformation können Sie im DP-Master mit der Anweisung auslesen.

Die Anweisung ist nur im S7-kompatiblen Modus verwendbar.

- DP: Der Master bindet den I-Slave mittels GSD-Datei ein.
- S7-kompatibel: Der I-Slave wird per Projektierung an einem Master angebunden.

Hinweis

Die Schnittstelle der Anweisung "SALRM" ist identisch mit der des in der Norm "PROFIBUS and PROFINET Guideline Communication Function Blocks on PROFIBUS DP and PROFINET IO" definierten FB "SALRM".

Arbeitsweise

Die Anweisung "SALRM" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Alarmübertragung, indem Sie Anweisung mit REQ = 1 aufrufen.

Der Sendevorgang ist solange aktiv, bis die Alarmbearbeitung vom DP-Master quittiert oder abgebrochen wird.

Über den Ausgangsparameter BUSY und die Bytes 2 und 3 des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die Bytes 2 und 3 von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Die Alarmübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

Identifikation des Auftrags

Falls Sie mit der Anweisung "SALRM" das Senden eines Alarms an den DP-Master angestoßen haben und Sie diese Anweisung erneut aufrufen, bevor der laufende Auftrag beendet wurde, hängt das weitere Verhalten der Anweisung davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt.

Stimmen die Parameter ID und ATYPE mit einem noch nicht abgeschlossenen Auftrag überein, so gilt der Aufruf als Folgeaufruf.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SALRM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Alarmübertragung durchführen
ID	Input	DWORD	E, A, M, D, L oder Konstante	irgendeine logische Adresse des Übergabebereichs zum DP-Master (virtuelle Steckplätze) (aus Sicht des DP-Slaves) außer der Diagnoseadresse der Station und der logischen Adresse des Steckplatzes 2. Die relevante Information steht im low word. Das high word belegen Sie mit Null. Bit 15 enthält die E/A-Kennung: 0 bei einer Eingangs-, 1 bei einer Ausgangsadresse.
ATYPE (Seite 5511)	Input	INT	E, A, M, D, L oder Konstante	Alarm Type Kennung für den Alarmtyp. Mögliche Werte: • 1: Diagnosealarm • 2: Prozessalarm
ASPEC (Seite 5511)	Input	INT	E, A, M, D, L oder Konstante	Alarm Specifier: • 0: keine weitere Information • 1: kommendes Ereignis, Steckplatz gestört • 2: gehendes Ereignis, Steckplatz nicht mehr gestört • 3: gehendes Ereignis, Steckplatz weiterhin gestört
LEN (Seite 5512)	Input	INT	E, A, M, D, L oder Konstante	Länge der zu versendenden Alarmzusatzinformation in Bytes Größter Wert: 16
DONE	Output	BOOL	E, A, M, D, L	DONE=1: Alarm wurde übertragen
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Die Alarmübertragung ist noch nicht abgeschlossen.
ERROR (Seite 5512)	Output	BOOL	E, A, M, D, L	ERROR = 1: Es ist ein Fehler aufgetreten.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS (Seite 5512)	Output	DWORD	E, A, M, D, L	Fehlerinformation
AINFO (Seite 5513)	InOut	ANY	E, A, M, D, L	Alarm Info Quellbereich für Alarmzusatzinformation Hinweis: Beachten Sie, dass der Parameter AINFO bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Auswirkung des "SALRM"-Aufrufs auf die Baugruppenzustandsinformation und die Sammelfehler-LED (SF)

Ein intelligenter Slave hat wie jede CPU die Eigenschaften seiner Steckplätze in der Baugruppenzustandsinformation abgelegt.

Siehe auch: SZL-ID W#16#xy91 - Baugruppenzustandsinformation (Seite 5720).

Wenn Sie mittels "SALRM" einen Diagnosealarm versenden, dann beeinflusst das Betriebssystem des intelligenten Slaves die slavelokale Baugruppenzustandsinformation und die LED SF anhand von Bit 0 im Byte 0 von AINFO (Dieses Bit wird als "Baugruppe gestört" in die Zustandsinformation übernommen.). Im I-Slave erfolgt jedoch kein Diagnosepuffereintrag, und es wird kein Diagnosealarm-OB gestartet.

Konsistenz der Baugruppenzustandsinformation zwischen DP-Master und I-Slave

Im Folgenden werden unterschiedliche Szenarien und ihre Auswirkungen auf die Baugruppenzustandsinformation betrachtet:

- Stationswiederkehr (hat OB 86-Start beim DP-Master und beim I-Slave zur Folge)
  Sowohl beim I-Slave als auch beim S7-Master wird die Baugruppenzustandsinformation beeinflusst ("Baugruppe gestört" wird rückgesetzt.). Falls aus Anwendersicht nach Stationswiederkehr beim I-Slave Störungen anstehen, müssen Sie diese per "SALRM"-Aufruf an den DP-Master melden.
- STOP-RUN-Übergang des DP-Masters (hat OB 82-Start beim I-Slave zur Folge)
  Die Baugruppenzustandsinformation auf dem I-Slave bleibt unverändert. Der DP-Master setzt das Bit "Baugruppe gestört" in der zugehörigen Baugruppenzustandsinformation zurück.
  Um die Konsistenz der Baugruppenzustandsinformation zwischen DP-Master und I-Slave im S7-kompatiblen Mode zu gewährleisten, müssen Sie auf dem I-Slave wie folgt reagieren:
  - Für jeden fehlerfreien virtuellen Steckplatz senden Sie per "SALRM" einen gehenden Diagnosealarm an den DP-Master.
  - Für jeden gestörten virtuellen Steckplatz senden Sie per "SALRM" einen kommenden Diagnosealarm an den DP-Master.
- STOP-RUN-Übergang des I-Slaves (hat OB 82-Start beim DP-Master zur Folge)
  Die Baugruppenzustandsinformation auf dem DP-Master bleibt unverändert, die auf dem I-Slave wird zurückgesetzt ("Baugruppe gestört" wird rückgesetzt.).
  Um die Konsistenz der Baugruppenzustandsinformation zwischen DP-Master und I-Slave zu gewährleisten, müssen Sie auf dem I-Slave wie folgt reagieren:
  - Für jeden fehlerfreien virtuellen Steckplatz senden Sie per "SALRM" einen gehenden Diagnosealarm an den DP-Master
  - Für jeden gestörten virtuellen Steckplatz senden Sie per "SALRM" einen kommenden Diagnosealarm an den DP-Master

  Hinweis

  Da "SALRM" asynchron arbeitet, können die "SALRM"-Aufrufe in den Anlauf-OBs nicht beendet werden, d. h. sie müssen im zyklischen Programm zu Ende geführt werden.

  Alle oben genannten Differenzen zwischen der Baugruppenzustandsinformation im Master und im I-Slave können nur bei denjenigen Steckplätzen auftreten, die per "SALRM" mit Diagnose-Alarmen beaufschlagt werden. Das wiederum bedeutet, dass die beschriebenen Abhilfen auch nur für solche Steckplätze erforderlich sind.

Parameter ATYPE

Parameter ATYPE

In der folgenden Tabelle wird für alle zulässigen Werte von ATYPE angegeben, welcher OB im zugehörigen DP-Master gestartet wird und in welchem DP-Mode die jeweilige Alarmart zulässig ist.

ATYPE	Bedeutung in DPV1-Norm	Zugehöriger OB im S7-DP-Master	DP-Mode	
			DP	S7-kompatibel
1	Diagnosis Alarm	Diagnose-Alarm-OB (OB82)	–	ja
2	Process Alarm	Prozessalarm-OBs (OBs 40 bis 47)	–	ja

Hinweis

Die Verwendbarkeit der Alarmarten kann zusätzlich zur obigen Tabelle vom DP-Master eingeschränkt werden.

Abhängigkeit der Alarmart vom Betriebszustand eines S7-Masters

Bei einem Slave im S7-kompatiblen Mode (Betrieb an einem S7-Master) können Prozess- und Diagnosealarme nur dann versendet werden, wenn sich der zugehörige DP-Master im Betriebszustand RUN (DP: Operate) befindet. Ist der DP-Master im Betriebszustand STOP (DP: Clear), wird der Alarm zurückgehalten, und "SALRM" liefert die Fehlerinformation W#16#80C8. Für ein Nachholen der Alarmversendung zu einem späteren Zeitpunkt sind Sie selbst verantwortlich.

Parameter ASPEC

Parameter ASPEC

Dieser Parameter zeigt laut Norm den Diagnosezustand des virtuellen Steckplatzes an. Daher dürfen Sie ASPEC nur beim Senden eines Diagnosealarms einen von Null verschiedenen Wert zuweisen.

Da in der S7-Alarmzusatzinformation des Diagnosealarms (Datensatz 0) eine Kommend/Gehend-Information vorhanden ist (siehe Diagnosedaten>Main Byte 0 Bit 0), müssen Sie im Byte 0 der Alarmzusatzinformation das Bit 0 (Baugruppenstörung) wie folgt beschreiben:

ASPEC	Bit "Baugruppenstörung" in AINFO (Seite 5513)
0	0
1	1
2	0
3	1

Parameter LEN

Parameter LEN

In LEN geben Sie die Länge der zu versendenden Alarmzusatzinformation in Bytes an. Der maximal zulässige Wertebereich beträgt 0 bis 16.

In der folgenden Tabelle wird für alle möglichen Alarmtypen angegeben, welche Werte für LEN in den einzelnen Modi eines intelligenten Slaves zulässig sind.

Alarmtyp	DP	S7-kompatibel
Diagnosealarm	–	4 bis 16
Prozessalarm	–	4

In der folgenden Tabelle wird beschrieben, welche Reaktion "SALRM" zeigt, wenn Sie LEN einen anderen Wert als die Länge von AINFO (Seite 5513) in BYTE zuweisen.

Wert von LEN	Verhalten von "SALRM"
<= Längenangabe von AINFO	"SALRM" sendet Alarm an DP-Master. Es werden so viele Bytes der Alarmzusatzinformation übertragen, wie in LEN angegeben ist.
außerhalb des zulässigen Wertebereichs, z. B. < 0 oder > 16	"SALRM" sendet keinen Alarm. Fehlerinformation: W#16#80B1, STATUS[4]=B#16#FF
> Längenangabe von AINFO	"SALRM" sendet Alarm an DP-Master. Es werden so viele Bytes der Alarmzusatzinformation übertragen, wie in der Längenangabe von AINFO angegeben ist. Fehlerinformation: W#16#00B1, STATUS[4]=Längenangabe von AINFO

Parameter STATUS und ERROR

Parameter STATUS und ERROR

Der Ausgangsparameter STATUS enthält Fehlerinformationen. Wird er als ARRAY[1 ... 4] OF BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Bedeutung
STATUS[1]	• B#16#00: kein Fehler • B#16#C0: vom I-Slave erkannter Fehler
STATUS[2], STATUS[3]	entspricht dem Ausgangsparameter RET_VAL von Anweisungen
STATUS[4]	B#16#00 mit Ausnahme einiger Längenkonflikte zwischen LEN und der Länge von AINFO (Seite 5513). Diese Ausnahmen sind Bestandteil der folgenden Tabelle.

In der folgenden Tabelle werden alle spezifischen Fehlerinformationen von "SALRM" angegeben.

ERROR	STATUS[2,3] (W#16# ...)	Erläuterung
0	0000	Der Auftrag wurde fehlerfrei durchgeführt. Falls LEN < Länge von AINFO ist, wurden nur LEN Bytes der Alarmzusatzinformation übertragen.
0	00B1	LEN > Länge von AINFO. Der Auftrag wurde durchgeführt. Die in AINFO angegebene Alarmzusatzinformation wurde übertragen. STATUS[4] enthält die Länge von AINFO.
0	7000	Erstaufruf mit REQ=0 (Leerdurchlauf). Es wurde kein Alarm gesendet. BUSY hat den Wert "0".
0	7001	Erstaufruf mit REQ=1. Der Auftrag wurde angestoßen. BUSY hat den Wert "1".
0	7002	Zwischenaufruf (REQ irrelevant). Der versendete Alarm wurde vom DP-Master noch nicht quittiert. BUSY hat den Wert "1".
1	8090	Die in ID angegebene Adresse liegt außerhalb des zulässigen Adressbereichs oder wurde nicht projektiert.
1	8091	• Sie haben den Alarm per Projektierung gesperrt. • Der Alarm ist für den Slavetyp nicht zulässig.
1	8092	Unzulässiger Datentyp in AINFO (Zulässig sind BYTE und BLOCK-DB)
1	8093	ID gehört zu einem virtuellen Steckplatz, von dem aus keine Alarmanforderung möglich ist.
1	80B0	ASPEC • passt nicht zum Bit 0 des Bytes 0 von AINFO • muss beim verwendeten Alarmtyp den Wert 0 haben • liegt außerhalb des zulässigen Wertebereichs
1	80B1	LEN liegt außerhalb des zulässigen Wertebereichs. STATUS[4] enthält B#16#FF.
1	80B5	Aufruf von "SALRM" beim DP-Master ist nicht zulässig.
1	80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
1	80C5	Dezentrale Peripherie ist momentan nicht verfügbar (z. B. Stationsausfall)
1	80C8	Die Funktion ist im aktuellen Betriebszustand des DP-Masters nicht erlaubt (Der DP-Master ist ein S7-Master und befindet sich im Betriebszustand STOP).

Parameter AINFO

Parameter AINFO

Der Parameter AINFO ist der Quellbereich für die Alarmzusatzinformation. Aus Sicht des intelligenten Slaves können Sie diesen Bereich mit beliebigen Werten beschreiben. Wenn Sie jedoch einen DP-Master aus dem S7-Spektrum einsetzen, muss die mit dem Alarm gesendete Zusatzinformation den S7-Konventionen entsprechen.

Falls Sie einen **Diagnosealarm** versenden (ATYPE (Seite 5511) =1), sind Sie selbst für eine sinnvolle Belegung des Datensatzes 0 und ggf. des Datensatzes 1 verantwortlich.

In der folgenden Tabelle erhalten Sie einen Vorschlag für eine S7-konforme Belegung. Dabei wurde das Setzen des Bits "Baugruppenstörung" (s. o.) bereits vorgenommen. Der Vorschlag entspricht bis auf das genannte Bit der Defaultbelegung (die nach NETZEIN, nach einem STOP-RUN-Übergang des intelligenten Slaves oder einer Stationswiederkehr erfolgt).

Datensatz-Nr.	Belegung
0	B#16#01, 0B, 00, 00
1	bei S7-kompatiblem Mode: Datensatz 0 + 12 Bytes mit Null

Weitere Informationen

Siehe auch: Aufbau der Diagnosedaten (Seite 5632).

PROFIBUS

DP_PRAL: Prozessalarm aus DP-Normslave auslösen

Beschreibung

Mit der Anweisung lösen Sie aus dem Anwenderprogramm eines intelligenten Slaves beim zugehörigen DP-Master einen Prozessalarm aus. Das führt zum Start des OB 40 beim DP-Master.

Mit dem Eingangsparameter AL_INFO können Sie die Ursache für den von Ihnen gewünschten Prozessalarm kennzeichnen. Diese Alarmkennung wird an den DP-Master übertragen und kann von Ihnen im OB 40 (Variable OB40_POINT_ADDR) ausgewertet werden.

Der angeforderte Prozessalarm wird durch die Eingangsparameter IOID und LADDR eindeutig festgelegt. Für jeden projektierten Adressbereich im Übergabespeicher können Sie zu einem beliebigen Zeitpunkt genau einen Prozessalarm auslösen.

Arbeitsweise

"DP_PRAL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Prozessalarmanforderung, indem Sie "DP_PRAL" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Der Auftrag ist abgeschlossen, wenn die Bearbeitung des OB 40 im DP-Master beendet ist.

Hinweis

Betreiben Sie den DP-Slave als Normslave, ist der Auftrag abgeschlossen, sobald das Diagnosetelegramm vom DP-Master abgeholt wurde.

Identifikation eines Auftrags

Die Eingangsparameter IOID und LADDR legen einen Auftrag eindeutig fest.

Falls Sie die Anweisung "DP_PRAL" auf einem DP-Slave aufgerufen haben und Sie diese Anweisung erneut aufrufen, bevor der DP-Master den angeforderten Prozessalarm quittiert hat, dann hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt: Stimmen die Parameter IOID und LADDR mit einem noch nicht abgeschlossenen Auftrag überein, so wird der Aufruf unabhängig vom Wert des Parameters AL_INFO als Folgeaufruf interpretiert, und in RET_VAL wird der Wert W#16#7002 eingetragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DP_PRAL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ=1: Prozessalarm auf dem zugehörigen DP-Master auslösen
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs im Übergabespeicher (aus Sicht des DP-Slaves): • B#16#00: Bit15 von LADDR gibt an, ob Ein- (Bit15=0) oder Ausgangsadresse (Bit15=1) vorliegt. • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Anfangsadresse des Adressbereichs im Übergabespeicher (aus Sicht des DP-Slaves). Handelt es sich um einen Bereich, der zu einer Mischbaugruppe gehört, ist die kleinere der beiden Adressen anzugeben.
AL_INFO	Input	DWORD	E, A, M, D, L oder Konstante	Alarmkennung Diese wird dem OB 40, der auf dem zugehörigen DP-Master gestartet werden soll, mitgegeben (Variable OB40_POINT_ADDR). Falls Sie den intelligenten Slave an einem Fremdmaster betreiben, müssen Sie im Master das Diagnosetelegramm auswerten.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der ausgelöste Prozessalarm wurde vom DP-Master noch nicht quittiert.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Es ist keine Prozessalarmanforderung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1. Eine Prozessalarmanforderung an den DP-Master wurde gestellt; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Der ausgelöste Prozessalarm wurde vom DP-Master noch nicht quittiert; BUSY hat den Wert "1".
8090	Anfangsadresse des Adressbereichs im Übergabespeicher fehlerhaft
8091	Alarm durch Projektierung gesperrt.
8093	Über das Parameterpaar IOID und LADDR wird eine Baugruppe angesprochen, von der aus eine Prozessalarmanforderung nicht möglich ist.
80B5	Aufruf in DP-Master nicht zulässig
80C3	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C5	Dezentrale Peripherie ist momentan nicht verfügbar (z. B. Stationsausfall).
80C8	Die Funktion ist im aktuellen Betriebszustand des DP-Masters nicht erlaubt.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DPSYC_FR: DP-Slaves synchronisieren / Eingänge einfrieren

Beschreibung

Mit der Anweisung können Sie eine oder mehrere Gruppen von DP-Slaves synchronisieren.

Dazu schicken Sie eines der folgenden Steuerkommandos oder eine Kombination davon an die betroffenen Gruppen:

- SYNC (gleichzeitiges Ausgeben und Einfrieren von Ausgangszuständen bei den DP-Slaves)
- UNSYNC (hebt das Steuerkommando SYNC wieder auf)
- FREEZE (Einfrieren von Eingangszuständen bei den DP-Slaves und Einlesen der eingefrorenen Eingänge)
- UNFREEZE (hebt das Steuerkommando FREEZE wieder auf)

Bevor Sie die oben genannten Steuerkommandos abschicken, müssen Sie die DP-Slaves per Projektierung in Gruppen eingeteilt haben. Sie müssen wissen, welcher DP-Slave welcher

Gruppe mit welcher Nummer zugeordnet ist und welche Eigenschaften hinsichtlich SYNC-/FREEZE-Verhalten die einzelnen Gruppen haben.

Hinweis

Beachten Sie, dass die Steuerkommandos SYNC und FREEZE auch bei einem Neustart/Kaltstart ihre Gültigkeit behalten.

Beachten Sie außerdem, dass zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag bzw. nur ein FREEZE/UNFREEZE-Auftrag angestoßen sein darf.

Arbeitsweise

"DPSYC_FR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "DPSYC_FR" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Identifikation eines Auftrags

Falls Sie einen SYNC-/FREEZE-Auftrag angestoßen haben und Sie "DPSYC_FR" erneut aufrufen, bevor dieser beendet wurde, dann hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Stimmen die Eingangsparameter LADDR, GROUP und MODE überein, so gilt der Aufruf als Folgeaufruf.

Ausgänge von DP-Baugruppen beschreiben

Das Beschreiben der Ausgänge von DP-Baugruppen wird angestoßen durch

- Transfer-Befehle auf DP-Peripherie,
- das Schreiben des Prozessabbilds der Ausgänge in die Baugruppen (durch das Betriebssystem am Ende des OB 1 oder durch Aufruf der Anweisung "UPDAT_PO (Seite 5451)"),
- Aufruf der Anweisung "DPWR_DAT (Seite 5499)".

Im Normalfall überträgt der DP-Master die Ausgangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) an die Ausgänge der DP-Slaves.

Wenn Sie bestimmte Ausgangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig auf die Ausgänge an den Prozess ausgeben wollen, schicken Sie mit Hilfe der Anweisung "DPSYC_FR" das Steuerkommando SYNC an den zugehörigen DP-Master.

Was bewirkt SYNC?

Mit dem Steuerkommando SYNC werden die DP-Slaves der genannten Gruppen in den Sync-Modus geschaltet, d. h. der DP-Master überträgt die aktuellen Ausgangsdaten und veranlasst die betroffenen DP-Slaves, die Ausgänge einzufrieren. Bei den folgenden Ausgabetelegrammen speichern die DP-Slaves die Ausgangsdaten in einem internen Puffer; der Zustand der Ausgänge bleibt unverändert.

Nach jedem Steuerkommando SYNC legen die DP-Slaves der selektierten Gruppen die Ausgangsdaten ihres internen Puffers auf die Ausgänge an den Prozess.

Die Ausgänge werden erst dann wieder zyklisch aktualisiert, wenn Sie mit Hilfe der Anweisung "DPSYC_FR" das Steuerkommando UNSYNC absetzen.

Hinweis

Falls sich DP-Slaves der ausgewählten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den SYNC-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

Eingänge von DP-Baugruppen lesen

Die Eingangsdaten von DP-Baugruppen werden gelesen

- mit Ladebefehlen auf DP-Peripherie,
- bei der Aktualisierung des Prozessabbilds der Eingänge (durch das Betriebssystem am Anfang des OB 1 oder durch Aufruf der Anweisung "UPDAT_PI (Seite 5449)"),
- durch Aufruf der Anweisung "DPRD_DAT (Seite 5496)".

Im Normalfall erhält der DP-Master diese Eingangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) von seinen DP-Slaves und stellt sie der CPU zur Verfügung.

Wenn Sie bestimmte Eingangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig vom Prozess einlesen wollen, schicken Sie mit Hilfe der Anweisung "DPSYC_FR" das Steuerkommando FREEZE an den zugehörigen DP-Master.

Was bewirkt FREEZE?

Mit dem Steuerkommando FREEZE werden die betroffenen DP-Slaves in den Freeze-Modus geschaltet, d. h. der DP-Master veranlasst die betroffenen DP-Slaves, den aktuellen Zustand der Eingänge einzufrieren. Anschließend überträgt er die eingefrorenen Daten in den Eingangsbereich der CPU.

Nach jedem Steuerkommando FREEZE frieren die DP-Slaves den Zustand der Eingänge erneut ein.

Der DP-Master erhält erst dann wieder zyklisch den aktuellen Zustand der Eingänge, wenn Sie mit Hilfe der Anweisung "DPSYC_FR" das Steuerkommando UNFREEZE absetzen.

Hinweis

Falls sich DP-Slaves der ausgewälten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den Freeze-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

Datenkonsistenz

Aufgrund der asynchronen Arbeitsweise und der Unterbrechbarkeit durch höhere Prioritätsklassen sollten Sie bei Einsatz der Anweisung "DPSYC_FR" darauf achten, dass die Prozessabbilder konsistent zu den tatsächlichen Ein- und Ausgängen der Peripherie sind.

Dies ist gewährleistet, wenn Sie eine der im Folgenden genannten Konsistenzregeln beachten:

- Definieren Sie für die "SYNC-Ausgänge" und die "FREEZE-Eingänge" geeignete Teilprozessabbilder (nur möglich bei S7-400). Rufen Sie die Anweisung "UPDAT_PO (Seite 5451)" unmittelbar vor dem jeweiligen Erstaufruf eines SYNC-Auftrags auf. Rufen Sie die Anweisung "UPDAT_PI (Seite 5449)" unmittelbar nach dem jeweiligen Letztaufruf eines FREEZE-Auftrags auf.
- Alternativ dazu: Verwenden Sie für Ausgänge, die von einem SYNC-Auftrag betroffen sind, und für Eingänge, die von einem FREEZE-Auftrag betroffen sind, nur direkte Peripheriezugriffe. Sie dürfen bei einem laufenden SYNC-Auftrag die betroffenen Ausgänge nicht beschreiben und bei einem laufenden FREEZE-Auftrag die betroffenen Eingänge nicht einlesen.

Einsatz von DPWR_DAT und DPRD_DAT

Falls Sie die Anweisung "DPWR_DAT (Seite 5499)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Ausgänge einen SYNC-Auftrag abschicken.

Falls Sie die Anweisung "DPRD_DAT (Seite 5496)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Eingänge einen FREEZE-Auftrag abschicken.

Anlauf und "DPSYC_FR"

Das Absetzen der Steuerkommandos SYNC und FREEZE in den Anlauf-OBs liegt in alleiniger Verantwortung des Anwenders.

Falls die Ausgänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im SYNC-Modus arbeiten sollen, müssen Sie im Anlauf diese Ausgänge initialisieren und die Anweisung "DPSYC_FR" mit dem Steuerkommando SYNC vollständig abarbeiten.

Falls die Eingänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im FREEZE-Modus arbeiten sollen, müssen Sie im Anlauf für diese Eingänge die Anweisung "DPSYC_FR" mit dem Steuerkommando FREEZE vollständig abarbeiten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPSYC_FR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Pegelgetriggerter Steuerparameter REQ=1: Anstoß des SYNC-/FREEZE-Auftrags
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Adresse des DP-Masters
GROUP	Input	BYTE	E, A, M, D, L oder Konstante	Gruppenauswahl Bit 0 = 1: Gruppe 1 ausgewählt Bit 1 = 1: Gruppe 2 ausgewählt : Bit 7 = 1: Gruppe 8 ausgewählt Pro Auftrag können Sie mehrere Gruppen auswählen. Der Wert B#16#0 ist nicht zugelassen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung (Codierung gemäß EN 50 170 Volume 2, PROFIBUS) Bit 0: reserviert (Wert 0) Bit 1: reserviert (Wert 0) Bit 2: • = 1: UNFREEZE wird ausgeführt • = 0: keine Bedeutung Bit 3: • = 1: FREEZE wird ausgeführt • = 0: keine Bedeutung Bit 4: • = 1: UNSYNC wird ausgeführt • = 0: keine Bedeutung Bit 5: • = 1: SYNC wird ausgeführt • = 0: keine Bedeutung Bit 6: reserviert (Wert 0) Bit 7: reserviert (Wert 0) Mögliche Werte: • bei genau einer Kennung pro Auftrag: – B#16#04 (UNFREEZE) – B#16#08 (FREEZE) – B#16#10 (UNSYNC) – B#16#20 (SYNC) • bei mehreren Kennungen pro Auftrag: – B#16#14 (UNSYNC, UNFREEZE) – B#16#18 (UNSYNC, FREEZE) – B#16#24 (SYNC, UNFREEZE) – B#16#28 (SYNC, FREEZE)
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Sie müssen RET_VAL nach jedem Bausteindurchlauf auswerten.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der SYNC-/FREEZE-Auftrag ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Hinweis

Falls Sie auf DPV1-Slaves zugreifen, können Fehlerinformationen dieser Slaves vom DP-Master an die Anweisung weitergereicht werden. Zur Beschreibung dieser Fehlerinformationen siehe STATUS[3], Parameter STATUS (Seite 5468).

Fehlercode (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Der über LADDR, GROUP und MODE festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert 0.
7001	Erstaufruf mit REQ=1. Der über LADDR, GROUP und MODE festgelegte Auftrag wurde angestoßen; BUSY hat den Wert 1.
7002	Zwischenaufruf (REQ irrelevant). Der aktivierte SYNC-/FREEZE-Auftrag läuft noch; BUSY hat den Wert 1.
8090	Die über LADDR ausgewählte Baugruppe ist kein DP-Master.
8093	Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig (Projektierung oder Version des DP-Masters).
8094	Parameter GROUP falsch
8095	Parameter MODE falsch
80B0	Die über GROUP ausgewählte Gruppe ist nicht projektiert.
80B1	Die über GROUP ausgewählte Gruppe ist dieser CPU nicht zugeordnet.
80B2	Der über MODE spezifizierte SYNC-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.
80B3	Der über MODE spezifizierte FREEZE-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.
80C2	Temporärer Ressourcenmangel des DP-Masters: Der DP-Master bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.
80C3	Dieser SYNC-/UNSYNC-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm.
80C4	Dieser FREEZE-/UNFREEZE-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein FREEZE-/UNFREEZE-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm.
80C5	Kurzschluss unmittelbar an der DP-Schnittstelle
80C6	Auftragsabbruch wegen Peripherieabwurf durch die CPU
80C7	Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DPNRM_DG: Diagnosedaten eines DP-Slaves lesen

Beschreibung

Mit der Anweisung "DPNRM_DG" lesen Sie die aktuellen Diagnosedaten eines DP-Slaves in der Form, wie sie durch EN 50 170 Volume 2, PROFIBUS festgelegt sind. Die gelesenen Daten werden nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen.

Den prinzipiellen Aufbau der Slave-Diagnose entnehmen Sie folgender Tabelle, weitergehende Informationen den Handbüchern zu den DP-Slaves.

Byte	Bedeutung
0	Stationsstatus 1
1	Stationsstatus 2
2	Stationsstatus 3
3	Master-Stationsnummer
4	Herstellerkennung (high byte)
5	Herstellerkennung (low byte)
6 ...	Weitere slavespezifische Diagnose

Arbeitsweise

Der Lesevorgang wird asynchron ausgeführt, d. h. er kann sich über mehrere Aufrufe erstrecken. Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Aufruf

Sie starten den Lesevorgang, indem Sie beim Aufruf der Anweisung den Eingangsparameter REQ mit "1" belegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPNRM_DG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Lesen
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Projektierte Diagnoseadresse des DP-Slaves Hinweis: Adresse muss hexadezimal angegeben werden. Z. B. Diagnoseadresse 1022 bedeutet: LADDR:=W#16#3FE.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Falls kein Fehler auftrat, steht in RET_VAL die Länge der tatsächlich übertragenen Daten.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RECORD	Output	ANY	E, A, M, D, L	Zielbereich für die gelesenen Diagnosedaten. Es ist nur der Datentyp BYTE zulässig. Die Mindestlänge des zu lesenden Datensatzes bzw. des Zielbereichs beträgt 6. Die Maximallänge des zu lesenden Datensatzes beträgt 240; bei Normslaves, bei denen die Anzahl der Normdiagnosedaten größer als 240 Bytes ist und maximal 244 Bytes beträgt, werden die ersten 240 Bytes in den Zielbereich übertragen und das entsprechende Overflow-Bit in den Daten gesetzt. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der Lesevorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Informationen zur Datentypkonvertierung in den einzelnen Programmiersprachen finden Sie unter "Übersicht über die Datentypkonvertierung (Seite 369)".

Parameter RECORD

Die CPU bewertet die tatsächliche Länge der gelesenen Diagnosedaten:

Falls die Längenangabe von RECORD

- kleiner als die Anzahl der gelieferten Daten ist, werden die Daten verworfen, und in RET_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich der Anzahl der gelieferten Daten ist, werden die Daten in den Zielbereich übernommen, und in RET_VAL wird die tatsächliche Länge als positiver Wert eingetragen.

Hinweis

Sie müssen darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen, die zu einem Auftrag gehören, übereinstimmen.

Ein Auftrag ist eindeutig festgelegt durch den Eingangsparameter LADDR.

Parameter RET_VAL

- Trat während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode,
- Falls bei der Übertragung kein Fehler auftrat, enthält RET_VAL die Länge der gelesenen Daten in Bytes als positive Zahl.

Hinweis

Die Anzahl der gelesenen Daten ist bei einem DP-Slave von seinem Diagnosezustand abhängig.

Zur Auswertung der Fehlerinformationen des Parameters RET_VAL siehe Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171). Dort finden Sie auch die allgemeinen Fehlerinformationen der Anweisungen. Die spezifischen Fehlerinformationen von "DPNRM_DG" sind eine Teilmenge der Fehlerinformationen von "RD_REC (Seite 5490)".

Normslaves mit mehr als 240 Bytes Diagnosedaten

Bei Normslaves, bei denen die Anzahl der Normdiagnosedaten zwischen 241 und 244 Bytes liegt, müssen Sie folgendes beachten:

Falls die Längenangabe von RECORD

- kleiner als 240 Bytes ist, werden die Daten verworfen, und in RET_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich 240 Bytes ist, werden die ersten 240 Bytes der Normdiagnosedaten in den Zielbereich übertragen und das entsprechende Overflow-Bit in den Daten gesetzt.

Systemressourcen bei S7-400

Beim Aufruf von "DPNRM_DG" zu einem derzeit nicht bearbeiteten Auftrag werden bei S7-400 Ressourcen der CPU (Speicherplatz) belegt. Sie können "DPNRM_DG" für mehrere DP-Slaves kurz nacheinander aufrufen, wenn Sie die für Ihre CPU maximale Anzahl "gleichzeitig" aktiver "DPNRM_DG"-Aufträge nicht überschreiten.

Bei mehreren "gleichzeitig" aktiven Aufträgen wird gewährleistet, dass alle Aufträge durchgeführt werden und keine gegenseitige Beeinflussung stattfindet.

Wenn Sie die Begrenzung der Systemressourcen erreichen, so erhalten Sie dies in RET_VAL mitgeteilt. In diesem Fall wiederholen Sie den Auftrag.

Siehe auch

Übersicht über die Datentypkonvertierung (Seite 369)

DP_TOPOL: Topologie für DP-Mastersystem ermitteln

Beschreibung

Mit der Anweisung stoßen Sie die Topologieermittlung für ein ausgewähltes DP-Mastersystem an. Mit dem Aufruf von der Anweisung werden alle Diagnose-Repeater an einem DP-Mastersystem angesprochen.

Hinweis

Die Topologieermittlung kann zu einem Zeitpunkt nur für ein DP-Mastersystem durchgeführt werden.

Die Topologieermittlung ist die Voraussetzung für die detaillierte Anzeige des Fehlerorts bei auftretenden Leitungsfehlern. Führen Sie nach dem Aufbau und nach jeder Änderung des physikalischen Aufbaus eines DP-Mastersystems die Topologieermittlung mit der Anweisung "DP_TOPOL" erneut durch.

Änderungen des physikalischen Aufbaus sind:

- Änderung von Leitungslängen
- Hinzufügen oder Entfernen von Teilnehmern oder Komponenten mit Repeaterfunktion
- Ändern von Teilnehmeradressen

Wird von einem Diagnose-Repeater ein Fehler gemeldet, beschreibt "DP_TOPOL" die Ausgänge DPR und DPRI für die Dauer eines "DP_TOPOL"-Durchlaufs. Werden von mehreren Diagnose-Repeatern des ausgewählten DP-Mastersystems Fehler gemeldet, schreibt "DP_TOPOL" in DPR und DPRI Informationen zum ersten fehlermeldenden Diagnose-Repeater. Die vollständige Diagnoseinformation können Sie mit der Anweisung "DPNRM_DG (Seite 5522)" oder am PG auslesen. Falls kein Diagnose-Repeater einen Fehler meldet, haben die Ausgänge DPR und DPRI den Wert NULL.

Wenn Sie nach dem Auftreten eines Fehlers erneut eine Topologieermittlung anstoßen wollen, müssen Sie "DP_TOPOL" zunächst zurücksetzen. Dies geschieht, indem Sie "DP_TOPOL" mit REQ=0 und R=1 aufrufen.

Arbeitsweise

"DP_TOPOL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Ermittlung der Bustopologie, indem Sie "DP_TOPOL" mit REQ=1 aufrufen. Falls Sie den Vorgang abbrechen möchten, rufen Sie "DP_TOPOL" mit R=1 auf.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Hinweis

Die Topologieermittlung kann mehrere Minuten dauern.

Identifikation eines Auftrags

Der Eingangsparameter DP_ID legt einen Auftrag eindeutig fest.

Falls Sie "DP_TOPOL" aufgerufen haben und Sie diese Anweisung erneut aufrufen, bevor die Topologieermittlung abgeschlossen ist, hängt das weitere Verhalten der Anweisung davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt: Stimmt der Parameter DP_ID mit einem noch nicht abgeschlossenen Auftrag überein, so wird der Aufruf als Folgeaufruf interpretiert, und in RET_VAL wird der Wert W#16#7002 eingetragen. Handelt es sich hingegen um einen weiteren Auftrag, weist die CPU diesen ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DP_TOPOL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ=1: Topologieermittlung anstoßen
R	Input	BOOL	E, A, M, D, L	R=1: Abbruch der Topologieermittlung
DP_ID	Input	INT	E, A, M, D, L oder Konstante	DP-Mastersystem-ID desjenigen DP-Mastersystems, dessen Topologie ermittelt werden soll
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Topologieermittlung ist noch nicht abgeschlossen.
DPR	Output	BYTE	E, A, M, D, L	PROFIBUS-Adresse des Fehler meldenden Diagnose-Repeaters
DPRI	Output	BYTE	E, A, M, D, L	Messsegment des Fehler meldenden Diagnose-Repeaters: • Bit 0 = 1: Temporäre Störungen auf Segment DP2 • Bit 1 = 1: Dauerhafte Störungen auf Segment DP2 • Bit 4 = 1: Temporäre Störungen auf Segment DP3 • Bit 5 = 1: Dauerhafte Störungen auf Segment DP3

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) der folgenden Tabelle sind zwei Fälle zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A4, 80C3, 80C5):
  Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt, d. h. es ist sinnvoll, dass Sie "DP_TOPOL" erneut aufrufen (ggf. mehrfach).
  Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#8082, 80B0, 80B2):
  Bei dieser Fehlerart kann sich der Fehler nicht ohne Ihr Zutun beheben. Ein erneuter Aufruf von "DP_TOPOL" ist erst sinnvoll, wenn Sie den Fehler beseitigt haben. Beispiel für einen permanenten Fehler: DP-Master / CPU unterstützt diesen Dienst nicht. (W#16#80B0).

Fehlercode (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Es wird keine Topologieerrmittlung angestoßen. BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1. Der Auftrag, die Topologieermittlung durchzuführen, wurde gestellt. BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Die Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".
7010	Sie haben versucht, die Topologieermittlung abzubrechen. Es gibt aber keinen laufenden Auftrag mit der angegebenen DP_ID. BUSY hat den Wert "0".
7011	Erstaufruf mit R=1. Der Abbruch der Topologieermittlung wurde angestoßen. BUSY hat den Wert "1".
7012	Zwischenaufruf: Der Abbruch der Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".
7013	Letztaufruf: Die Topologieermittlung wurde abgebrochen. BUSY hat den Wert "0".
8082	Es ist kein DP-Mastersystem mit der angegebenen DP_ID projektiert.
80A2	Fehler bei Topologieermittlung; genauere Informationen entnehmen Sie den Ausgangsparametern DPR und DPRI.
80A3	Fehler bei Topologieermittlung: Überwachungszeit ist abgelaufen (Timeout).
80A4	Kommunikation am K-Bus gestört
80B0	DP-Master / CPU unterstützt diesen Dienst nicht.
80B2	Fehler bei Topologieermittlung: Am ausgewählten DP-Mastersystem wurde kein Diagnose-Repeater erkannt.
80C3	Benötigte Betriebsmittel sind momentan belegt. Mögliche Ursache: Sie haben eine zweite Topologieermittlung angestoßen (nur eine Topologieermittlung ist zu einem Zeitpunkt zulässig).
80C5	DP-Mastersystem ist momentan nicht verfügbar.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

ASi

ASi_3422: AS-i-Masterverhalten steuern

Beschreibung

Die Anweisung "ASi_3422" wird zur Verwendung der Betriebsart "Erweiterter Betrieb" bei dem CP 342-2 benötigt. Der erweiterte Betrieb ermöglicht die vollständige Steuerung des Masterverhaltens über das Anwenderprogramm. Der Zugriff auf die Ein- und Ausgänge erfolgt nach wie vor wie im Standardbetrieb des CP 342-2.

Kommandoaufrufe an den CP342-2 erfolgen aus dem Anwenderprogramm über die Anweisung. Hierfür spezifizieren Sie in einem Sendepuffer einen Kommandoaufruf und starten den Auftrag.

Die Anweisung "ASi_3422" übergibt den Kommandoaufruf an den CP342-2 weiter. Mit Abschluss des Auftrags wird der Auftragsstatus übergeben und mögliche Antwortdaten in einem Empfangspuffer zur Verfügung gestellt.

Aufruf

Die Anweisung muss für jeden vorhandenen CP342-2 zyklisch aufgerufen werden. Zu einem Zeitpunkt kann je CP342-2 immer nur ein Auftrag bearbeitet werden. Ein laufender Auftrag ist nicht unterbrechbar und wird von der Anweisung zeitlich nicht überwacht. Die Anweisung "ASi_3422" ist nicht unterbrechbar ablauffähig. Aufrufe dürfen daher nicht in Programmablaufebenen programmiert werden, die sich gegenseitig unterbrechen (z.B. durch Aufruf in OB 1 und in OB 35).

Gestalten Sie im Anwenderprogramm die Kommandobearbeitung wie folgt:

1. Bei Neustart des S7−Anwenderprogrammes rufen Sie "ASi_3422" einmalig mit dem Parameterwert STARTUP = TRUE auf.
2. Im Anwenderprogramm spezifizieren Sie in einem Sendepuffer den Kommandoaufruf. Diesen Sendepuffer übergeben Sie mit dem Aufrufparameter SEND.
   Weitere Informationen zu der Kommandoschnittstelle des CPs und den ASi-Slave-Kommandos finden Sie in der Hardwaredokumentation des Kommunikationsprozessors.
3. Je nach Kommandotyp benötigen Sie einen Antwortpuffer. Diesen Antwortpuffer übergeben Sie mit dem Aufrufparameter RECV. Für Statusinformationen wird der Antwortpuffer bei dieser Schnittstelle nicht benötigt.
4. Aktivieren Sie den Auftrag über den Parameter ACT=1.
5. Anschließend fragen Sie die Parameter DONE, ERROR und STATUS ab.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ASi_3422":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACT	Input	BOOL	E, A, M, D, L	Die Kommandobearbeitung durch die Funktion erfolgt pegelgetriggert, d.h., solange ACT = 1 ist, wird eine Kommandobearbeitung gestartet, falls nicht bereits ein Aufruf in Bearbeitung ist.
STARTUP	Input	BOOL	E, A, M, D, L	Ein CPU-Anlauf wird der Anweisung durch STARTUP = 1 mitgeteilt. Nach dem ersten Durchlauf der Funktion muss STARTUP vom Anwender rückgesetzt werden.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Baugruppen-Anfangsadresse Die Baugruppen-Anfangsadresse ist gemäß den Angaben über die steckplatzorientierte Adressvergabe bei Signalbaugruppen zu ermitteln.
SEND	Input	ANY	E, A, M, D, L	Sendepuffer Der Parameter verweist auf einen Speicherbereich, in dem das Kommando vom Anwender zu spezifizieren ist. z.B.: P#DB20.DBX 20.0 Byte 10.
RECV	Input	ANY	E, A, M, D, L	Empfangspuffer Dieser Puffer ist nur bei Kommandos relevant, die Antwortdaten liefern. Der Parameter verweist auf einen Speicherbereich, in dem eine Kommandoantwort abgelegt wird. Die Länge des hier parametrierten Datenbereiches ist irrelevant. z.B.: P#DB30.DBX 20.0 Byte 1
DONE	Output	BOOL	A, M, D, L	Mit DONE = 1 wird ‘Auftrag fertig ohne Fehler’ signalisiert.
ERROR	Output	BOOL	A, M, D, L	Mit ERROR = 1 wird 'Auftrag fertig mit Fehler' signalisiert.
STATUS	InOut	DWORD	M, D	1. Wort: Auftragsstatus / Fehlercode (siehe nachfolgende Tabelle); Bei ‘Auftrag fertig mit Fehler’ wird zur näheren Fehlerbeschreibung ein Fehlercode generiert. 2. Wort: Wird von der Anweisung für interne Zwecke benötigt und darf nicht verändert werden. Für Anweisungs−Aufrufe an unterschiedlichen logischen Adressen (LADDR) müssen unterschiedliche Doppelworte für den Parameter STATUS vergeben werden!

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter LADDR

Die Anfangsadresse dieses Adressbereiches ist durch den Steckplatz des CP 342-2 bestimmt.

Der CP 342-2 belegt 16 Eingangsbyte und 16 Ausgangsbyte im E/A-Adressraum des S7-Automatisierungsgerätes (Analogbereich des AS). Von den 16 Byte Adressbereich des CP 342-2 werden 31 x 4 Bit für die AS-i-Slave-Daten belegt. Die restlichen 4 Bit sind für spätere Anwendungen reserviert.

Parameter DONE, ERROR und STATUS

Tritt während der Bearbeitung der Funktion ein Fehler auf, dann steht zusätzlich zu den o.g. Informationen in ERROR und STATUS im BIE-Bit eine "0". Die Abfrage des BIE-Bits erfolgt unterschiedlich bei KOP- und AWL Anwenderprogrammen:

- KOP: Abfrage über Ausgangsparameter ENO
- AWL: Direktabfrage des BIE-Bits

Die folgende Tabelle enthält die möglichen Anzeigen im 1. Wort von STATUS in Abhängigkeit von DONE und ERROR.

DONE	ERROR	STATUS	Bedeutung
0	0	8181	Auftrag läuft.
1	0	0000	Auftrag fertig ohne Fehler.
0	1	8090	Adresse am Parameter LADDR ungültig.
0	1	80A0	Negative Quittung beim Lesen von Baugruppe.
0	1	80A1	Negative Quittung beim Schreiben zur Baugruppe.
0	1	80B0	Baugruppe kennt den Datensatz nicht.
0	1	80B1	Angegebene Datensatzlänge ist falsch.
0	1	80C0	Datensatz kann nicht gelesen werden.
0	1	80C1	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	80C2	Es liegt ein Auftragstau vor.
0	1	80C3	Betriebsmittel (Speicher) belegt.
0	1	80C4	Kommunikationsfehler
0	1	8182	Bedeutung: Kennung nach Neustart mit STARTUP=TRUE (kein Fehler).
0	1	8184	Datentyp des Formaloperanden RECV unzulässig.
0	1	8381	Slave-Adresse falsch
0	1	8382	Slave ist nicht aktiviert (nicht in LAS).
0	1	8383	Fehler am AS-Interface
0	1	8384	Kommando (im Zustand des CPs) nicht zulässig.
0	1	8385	Slave 0 existiert.
0	1	83A1	Slave mit zu ändernder Adresse nicht an AS-Interface gefunden.
0	1	83A2	Slave 0 vorhanden.
0	1	83A3	Slave mit neuer Adresse bereits am AS-Interface vorhanden.
0	1	83A4	Slave-Adresse lässt sich nicht löschen.
0	1	83A5	Slave-Adresse lässt sich nicht setzen.
0	1	83A6	Slave-Adresse lässt sich nicht permanent speichern.
0	1	83F8	Auftragsnummer unbekannt.

DONE	ERROR	STATUS	Bedeutung
0	1	83F9	EEPROM-Fehler
0	1	8F22	Bereichslängenfehler beim Lesen eines Parameters.
		8F23	Bereichslängenfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich ein Parameter vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfelds bei einem ANY-Parameter nicht durch 8 teilbar ist.
0	1	8F24	Bereichsfehler beim Lesen eines Parameters.
		8F25	Bereichsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich ein Parameter in einem Bereich befindet, der für eine Systemfunktion unzulässig ist.
0	1	8F28	Ausrichtungsfehler beim Lesen eines Parameters.
		8F29	Ausrichtungsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass der Verweis auf einen Parameter ein Operand ist, dessen Bitadresse ungleich "0" ist.
0	1	8F30	Parameter liegt im schreibgeschützten Global-DB.
		8F31	Parameter liegt im schreibgeschützten Instanz-DB. Dieser Fehlercode zeigt an, dass ein Parameter sich in einem schreibgeschützten Datenbaustein befindet.
0	1	8F32	Parameter enthält zu große DB-Nummer
0	1	8F3A	Der Parameter enthält die Nummer eines DB, der nicht geladen ist.
0	1	8F42	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.
		8F43	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.
0	1	8F44	Dieser Fehlercode zeigt an, dass der lesende Zugriff auf einen Parameter verweigert wurde.
		8F45	Dieser Fehlercode zeigt an, dass der schreibende Zugriff auf einen Parameter verweigert wurde.
0	1	8F7F	Interner Fehler

ASI_CTRL: AS-i-Masterverhalten steuern

Beschreibung ASI_CTRL

Beschreibung

Mit der Anweisung "ASI_CTRL" kann das Verhalten des AS-i-Masters über das Anwenderprogramm der PLC gesteuert werden. Die Anweisung bearbeitet das Kommandoprotokoll selbstständig und ermöglicht auch die Parametrierungen an SIMATIC AS-i-Mastern sowie das Auslesen von Informationsdaten. Die Möglichkeiten und die Arbeitsweise der Kommandoschnittstelle sind im Handbuch des AS-i-Masters beschrieben.

Unterstützt werden sowohl zentral steckende AS-i-Master als auch dezentrale AS-i-Master über PROFIBUS DP. Auch Kombinationen mit PROFINET IO (Beispiel: IE/PB Link PN IO) sind möglich.

Die folgende Abbildung zeigt schematisch die Funktionalität der Anweisung "ASI_CTRL":

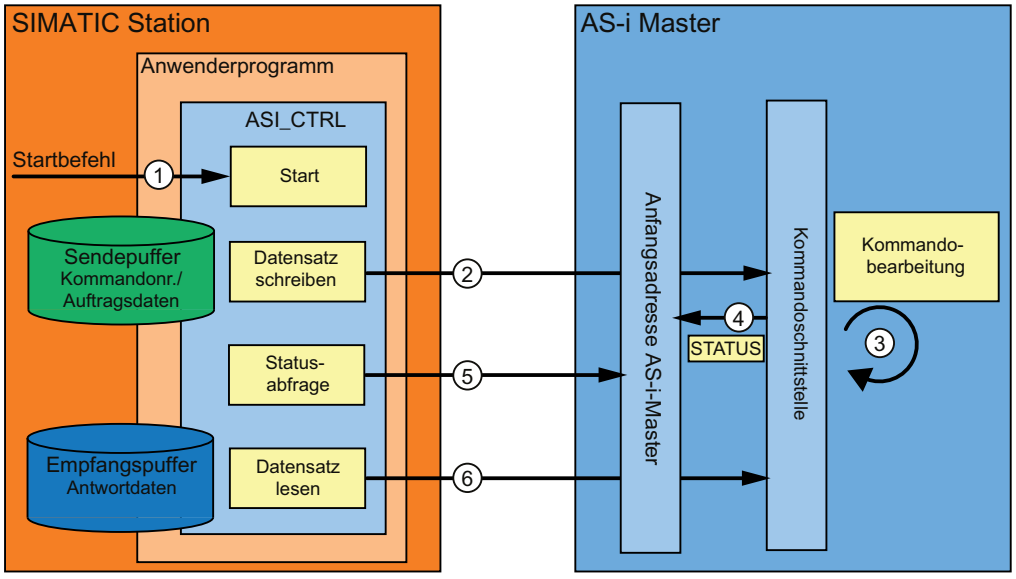

① Start der Bearbeitung am Parameter REQ.

② Das Programm setzt das gewünschte Kommando über die Anweisung "RDREC" an den AS-i-Master ab.

③ Der AS-i-Master bearbeitet das Kommando.

④ Der aktuelle Status des AS-i-Master wird im Eingangsbereich der Binärdaten (logische Basisadresse) abgelegt.

⑤ Die Anweisung "ASI_CTRL" fragt zyklisch die 4 Statusbits ab und wertet diese aus.

⑥ Ist die Kommandobearbeitung beendet, wird der Kommandoauftrag über "RDREC" abgeschlossen. Im Datenfeld von "RDREC" befinden sich je nach Kommando die Antwortdaten des Kommandos oder weitere Statusinformationen.

Abhängigkeiten zwischen Anweisungs-Versionen

Die Version V1.3 der Anweisung "ASI_CTRL" erfordert die Version V1.1 der Anweisung "WRREC".

Die Version V1.2 der Anweisung "ASI_CTRL" erfordert die Version V1.1 der Anweisung "WRREC".

Unterschiede im Kommandoaufruf bei IE/ AS-i Link und CPs bzw. DP/AS-i Links

Für den Kommandoaustausch einer Steuerung mit einem AS-i Master gibt es wesentliche Unterschiede in der Vorgehensweise.

- Beim **IE/AS-i Link** (PROFINET) wird die Datensatzschnittstelle verwendet. Die verschiedenen Kommandos werden entweder durch "Datensatz schreiben" (Anweisung "WRREC") oder durch "Datensatz lesen" ("Anweisung "RDREC") von verschiedenen Datensatznummern aufgerufen.
- **S7-300-CPs** und **DP/AS-i Links** (PROFIBUS) verwenden die Kommandoschnittstelle. Alle Kommandos werden durch ein paarweises "Datensatz schreiben" ("Anweisung "WRREC") und "Datensatz lesen" ("Anweisung "RDREC") von Datensatznummer 2 aufgerufen. Der Typ des Kommandos wird durch den Dateninhalt beim Schreibauftrag definiert.

Änderungen gegenüber der Anweisung "ASi_3422"

Die Anweisung "ASI_CTRL" ist eine Überarbeitung der Anweisung "ASi_3422 (Seite 5529)" und weist Verbesserungen hinsichtlich Funktionalität und Kompatibilität auf. Im Detail wurden folgende Änderungen durchgeführt:

- Zum Schreiben und Lesen von Diagnosedatensätze wurden die Anweisungen "WR_REC (Seite 5494)" und "RD_REC (Seite 5490)" durch die Anweisungen "RDREC (Seite 5456)" und "WRREC (Seite 5458)" ersetzt. Sie sind in der Funktion identisch, unterstützen jedoch die Datenübertragung über PROFINET IO.
- Der Bausteintyp der Anweisung wurde von einer Funktion (FC) in einen Funktionsbaustein (FB) gewandelt. "ASI_CTRL" verfügt über einen Instanzdatenbaustein und ist multiinstanzfähig.
- Die Benennung der Formalparameter von "ASI_CTRL" ist konform zu den SIMATIC Systembausteinen. Der Eingangsparameter STARTUP entfällt. Die Definition des Parameters STATUS wurde angelehnt an die Anweisungen "RDREC (Seite 5456)" und "WRREC (Seite 5458)". Zusätzlich wurden die Statuskennungen für den Parameter DONE und den neu eingeführten Parameter BUSY angepasst.

Arbeitsweise der Anweisung "ASI_CTRL"

Die Anweisung "ASI_CTRL" ist ein asynchron arbeitender Funktionsbaustein, das bedeutet die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Gestartet wird ein Auftrag mit REQ = TRUE.
- Über die Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt.
- Während einer Auftragsbearbeitung ist der Parameter BUSY gesetzt. Bei Erstaufruf enthält STATUS dabei den Wert 00700100_H. Bei allen Folgeaufrufen, welche zu diesem Auftrag gehören, den Wert 00700200_H. Mit Abschluss des Auftrages wird das Ergebnis an den Parametern DONE bzw. ERROR ausgegeben.
  - Ist kein Fehler aufgetreten, wird DONE gesetzt. Bei Aufträgen mit Antwortdaten vom AS-i-Master werden diese im angegebenen Empfangspuffer zur Verfügung gestellt. In diesem Fall wird im Parameter STATUS zusätzlich die Menge der gelieferten Daten in Bytes angezeigt. Bei Aufträgen ohne Antwortdaten wird in STATUS der Wert 00000000_H eingetragen.
  - Tritt während der Auftragsbearbeitung ein Fehler auf, wird ERROR gesetzt. In diesem Fall ist der Inhalt des Empfangspuffers ungültig. Zur näheren Beschreibung des aufgetretenen Fehlers wird im Parameter STATUS ein Fehlercode eingetragen.

Anzahl Kommandoaufrufe

Wenn Sie die Anweisung "ASI_CTRL" zur Kommandobeauftragung nutzen, dürfen Sie nicht gleichzeitig weitere Kommandos über "RDREC (Seite 5456)" und "WRREC (Seite 5458)" an denselben AS-i-Master absetzen. Dies gilt auch für Mehrfachaufrufe der Anweisung an denselben AS-i-Master.

Die Anweisung "ASI_CTRL" ist nicht unterbrechbar ablauffähig. Aufrufe dürfen daher nicht in Programmablaufebenen programmiert werden, die sich gegenseitig unterbrechen (z.B. durch Aufruf in OB 1 und in OB 35).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ASI_CTRL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Mit REQ = TRUE wird ein neuer Auftrag gestartet, falls nicht bereits ein Auftrag in Bearbeitung ist. Es findet keine Flankenauswertung statt.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Anfangsadresse des AS-i-Masters im S7−Adressraum (logische Basisadresse). Die Anfangsadresse wird beim Projektieren des Masters in der Hardware-Konfiguration festgelegt.
SD	Input	ANY	E, A, M, D, L	Sendepuffer Der Parameter verweist auf einen Speicherbereich, in dem das Kommando zu spezifizieren ist (siehe "ASi-Kommandos (Seite 5537)"). Beispiel: P#DB101.DBX 0.0 BYTE 223

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RD	Input	ANY	E, A, M, D, L	Empfangspuffer Dieser Puffer ist nur bei Kommandos relevant, die Antwortdaten liefern. Der Parameter verweist auf einen Speicherbereich, in dem eine Kommandoantwort abgelegt wird (siehe "ASi-Kommandos (Seite 5537)"). Beispiel: P#DB102.DBX 224.0 BYTE 221
DONE	Output	BOOL	A, M, D, L	DONE = TRUE: Auftrag ohne Fehler beendet.
BUSY	Output	BOOL	A, M, D, L	BUSY = TRUE: Auftrag in Bearbeitung.
ERROR	Output	BOOL	A, M, D, L	ERROR = TRUE: Auftrag mit Fehler abgebrochen.
STATUS	Output	DWORD	M, D	Auftragsstatus / Fehlercode Siehe Beschreibung "Parameter STATUS".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Parameter LADDR, SD und RD

Die Parameter LADDR, SD und RD dürfen während einer Auftragsbearbeitung in keinem Bausteindurchlauf verändert werden, sondern müssen konstant bleiben.

Parameter STATUS

Die folgende Tabelle enthält die möglichen Anzeigen von STATUS in Abhängigkeit von DONE und ERROR.

DONE	BUSY	ERROR	STATUS	Bedeutung
0	0	0	00700000_H	Erstaufruf mit REQ = FALSE; keine Auftrag aktiv.
0	1	0	00700100_H	Erstaufruf mit REQ = TRUE; Auftrag angestoßen.
0	1	0	00700200_H	Folgeaufruf (REQ irrelevant); Auftrag wird derzeit noch bearbeitet.
1	0	0	00000000_H	Auftrag ohne Fehler abgeschlossen. Keine Antwortdaten.
1	0	0	$0000xx00_H$	Auftrag ohne Fehler abgeschlossen. Anzahl xx-Bytes Antwortdaten.
0	0	1	$C0818400_H$	Datentyp des Formaloperanden RD unzulässig.
0	0	1	$C0818500_H$	Kommunikationsfehler mit AS-i-Master (falsche Adresse am Parameter LADDR projektiert).
0	0	1	$C0838100_H$	Falsche AS-i-Slave-Adresse.
0	0	1	$C0838200_H$	Der AS-i-Slave ist nicht aktiviert (nicht in LAS).
0	0	1	$C0838300_H$	Fehler am AS−Interface (der Parameter SD ist eventuell zu klein eingestellt).
0	0	1	$C0838400_H$	Das Kommando ist im aktuellen Zustand des AS-i-Masters nicht zulässig.

DONE	BUSY	ERROR	STATUS	Bedeutung
0	0	1	C0838500 H	Ein AS-i-Slave mit Adresse "0" existiert.
0	0	1	C0838600 H	Der AS-i-Slave hat unzulässige Konfigurationsdaten (E/A bzw. ID−Codes).
0	0	1	C083A100 H	Der angesprochene AS-i-Slave wurde am AS−Interface nicht gefunden.
0	0	1	C083A200 H	Ein AS-i-Slave mit Adresse "0" existiert.
0	0	1	C083A300 H	Ein AS-i-Slave mit der neuen Adresse ist bereits am AS−Interface vorhanden.
0	0	1	C083A400 H	Die AS-i-Slave-Adresse lässt sich nicht löschen.
0	0	1	C083A500 H	Die AS-i-Slave-Adresse lässt sich nicht setzen.
0	0	1	C083A600 H	Die AS-i-Slave-Adresse lässt sich nicht permanent speichern.
0	0	1	C083A700 H	Fehler beim Lesen des Extended ID1−Codes.
0	0	1	C083A800 H	Die Zieladresse ist nicht plausibel (z. B. wird eine B-Slave-Adresse für einen Standard−Slave verwendet).
0	0	1	C083B100 H	Es ist ein Längenfehler beim Stringtransfer aufgetreten.
0	0	1	C083B200 H	Es ist ein Protokollfehler beim Stringtransfer aufgetreten.
0	0	1	C083F800 H	Die Auftragsnummer oder der Auftragsparameter sind unbekannt.
0	0	1	C083F900 H	Der AS-i-Master hat einen EEPROM-Fehler festgestellt.

ASi-Kommandos

Beschreibung

Durch die Kommandoschnittstelle ist die Möglichkeit gegeben, Parametrier- und Informationsdaten zwischen Steuerung und AS-i-Master auszutauschen.

Mit diesen Kommandos

- wird die komplette Funktionalität des Masterprofiles M4 der AS-i-Masterspezifikation zur Verfügung gestellt.
- kann der AS-i-Master komplett von der Steuerung aus projektiert werden.

Hinweis

Unterstützte AS-i-Kommandos

Welche AS-i Kommandos unterstützt werden sowie eine detaillierte Beschreibung finden Sie im Handbuch des jeweiligen AS-i-Masters.

Allgemeine Struktur des Sendepuffers

Die allgemeine Struktur des Sendepuffers für Kommandos und Auftragsdaten zeigt die folgende Tabelle. Der Bereich für die Kommandonummer muss immer belegt sein. Die Anzahl Bytes für die Auftragsdaten sind in der Beschreibung des jeweiligen Kommandos enthalten (siehe Dokumentation des AS-i-Masters). "q" entspricht dabei der Anfangsadresse des Sendepuffers.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Bedeutung							
q + 0	Kommandonummer							
q + 1	Auftragsdaten							
q + 2	Auftragsdaten							
q + ...	Auftragsdaten							

Allgemeine Struktur des Empfangspuffers

Die allgemeine Struktur des Empfangspuffers für die Antwortdaten des Kommandos zeigt folgende Tabelle. Die Anzahl Bytes für die Antwortdaten ist vom jeweiligen Kommando abhängig. Einige Kommandos liefern keine Antwortdaten und benötigen somit lediglich die Angabe eines virtuellen Empfangspuffers, der nicht mit Daten befüllt wird. "n" entspricht dabei der Anfangsadresse des Empfangspuffers.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Bedeutung							
n + 0	Antwortdaten							
n + 1	Antwortdaten							
n + 2	Antwortdaten							
n + ...	Antwortdaten							

ACHTUNG

Speicherbereiche können überschrieben werden

Ist der Empfangspuffer der Anweisung "ASI_CTRL" zu kurz, werden eventuell angrenzende Speicherbereiche überschrieben. Die Längenangabe im ANY-Zeiger des Parameters RD beim Aufruf der Anweisung "ASI_CTRL" ist irrelevant. Die benötigte Länge des Empfangspuffers ist der Beschreibung des Kommandos zu entnehmen.

Für die Kommandonummern 39_H, 41_H, 42_H, 43_H und 44_H gilt:

Der Empfangspuffer muss eine Länge von 221 Byte haben (Byte 0 bis 220), auch wenn das Kommando weniger Daten zurückliefert. Je nach Kommando werden die im Empfangspuffer enthaltenen höchsten Bytes vom AS-i-Master eventuell mit Nullwerten überschrieben.

AS-i-Kommandos

Die folgende Tabelle zeigt eine Auswahl möglicher AS-i-Kommandos.

Name	Parameter	Rückgabe	Codierung
Parameterwert_projektieren (Set_Permanent_Parameter)	Slave–Adresse, Parameter		00 $_H$
Projektierten_Parameterwert_lesen (Get_Permanent_Parameter)	Slave–Adresse	Parameter	01 $_H$
Parameterwert_schreiben (Write_Parameter)	Slave–Adresse, Parameter	Parameterecho	02 $_H$
Parameterwert_lesen (Read_Parameter)	Slave–Adresse	Parameterwert	03 $_H$
Ist_Parameterwerte_projektieren (Store_Actual_Parameters)			04 $_H$
Konfigurationsdaten_projektieren	Slave–Adresse, Konfiguration		25 $_H$
Projektierte_Konfigurationsdaten_lesen	Slave–Adresse	projektierte Konfigurationsdaten	26 $_H$
Ist_Konfigurationsdaten_projektieren (Store_Actual_Configuration)			07 $_H$
Ist-Konfigurationsdaten_lesen	Slave-Adresse	Ist-Konfigurationsdaten	28 $_H$
LPS_projektieren	LPS		29 $_H$
Offlinemodus_setzen	Mode		0A $_H$
Autoprogrammieren_wählen	Mode		0B $_H$
Betriebsmodus_setzen	Mode		0C $_H$
AS–iSlave–Adresse_ändern (Change_AS–iSlave_Address)	Adresse1, Adresse2		0D $_H$
AS–iSlavestatus_lesen	Slave–Adresse	Fehlerrecord des AS-iSlaves	0F $_H$
Listen_und_Flags_lesen		LDS, LAS, LPS, Flags	30 $_H$
Gesamtkonfiguration_lesen		Ist-Konfigurationsdaten, Aktuelle Parameter, LAS, Flags	39 $_H$
Gesamtkonfiguration_projektieren	Gesamtkonfiguration		3A $_H$
Parameterliste_schreiben	Parameter Liste		3C $_H$
Parameterecholiste_lesen		Parameterecho-Liste	33 $_H$
CTT2-Request_schreiben	Slave Adresse CTT2-String	CTT2-String	44 $_H$
Versionskennung_lesen		Versions-String	14 $_H$
AS–iSlave-ID_lesen	Slave–Adresse	ID-Code	17 $_H$
AS–iSlave-Extended–ID1_lesen	Slave–Adresse	Extended ID1–Code	37 $_H$
AS–iSlave-Extended–ID1_schreiben	Extended ID1–Code		3F $_H$
AS–iSlave-Extended–ID2_lesen	Slave–Adresse	Extended ID2–Code	38 $_H$
AS–iSlave-EA_lesen	Slave–Adresse	E/A-Konfiguration	18 $_H$
Peripheriefehlerliste_lesen		LPF	3E $_H$
AS–i–Slave_Parameter–String_schreiben	Slave–Adresse, Parameter–String		40 $_H$

Name	Parameter	Rückgabe	Codierung
AS–iSlave_Parameter–String_lesen	Slave–Adresse	Parameter–String	41_H
AS–iSlave_ID–String_lesen	Slave–Adresse	ID–String	42_H
AS–iSlave_Diagnose–String_lesen	Slave–Adresse	Diagnose–String	43_H
AS–iStrang_Fehlerzähler_lesen			$4A_H$
AS–iStrang_Fehlerzähler_lesen_und_löschen			$4B_H$
AS–iSlave_Fehlerzähler_lesen	Slave–Adresse		$4C_H$
AS–iSlave_Fehlerzähler_lesen_und_löschen	Slave–Adresse		$4D_H$
Zusätzliches Kommando für den DP/ AS-i F-Link:			
AS–i-Status/Diag_der_F-Slaves		Status / Diagnose aller ASIsafe-Salves	51_H

Hinweis

Erneute Initialisierung der Kommandoschnittstelle des AS-i-Masters

Ein weiteres Kommando, welches in der Tabelle nicht erwähnt wird, ist das Kommando 77_H. Mit diesem Aufruf wird die Kommandoschnittstelle des AS-i-Masters neu initialisiert. Falls der angegebene AS-i-Master zu diesem Zeitpunkt mit einem Kommando beschäftigt sein sollte, wird dieses abgebrochen.

Ab Version V2.1.20 des DP/AS-i LINK Advanced wird zusätzlich das Kommando $0E_H$ angeboten. Mittels dieses Aufrufes kann die Funktion Erdschlussüberwachung eines Strangs freigegeben oder gesperrt werden.

4.2.3.5 PROFIenergy

Beschreibung PROFIenergy

PROFIenergy

PROFIenergy ist ein hersteller- und geräteunabhängiges Profil für das Energiemanagement mit PROFINET. Mit PROFIenergy lassen sich Geräte koordiniert und zentral abschalten, um während Produktionspausen oder ungeplanten Unterbrechungen den Stromverbrauch zu reduzieren.

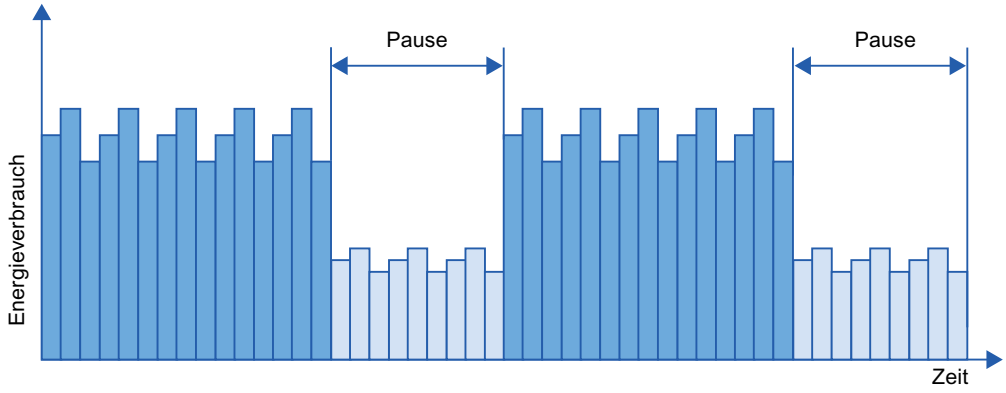

Die Abschaltung der PROFINET-Geräte bzw. der Powermodule erfolgt über spezielle Kommandos im Anwenderprogramm des PROFINET IO-Controllers. Es wird keine zusätzliche Hardware benötigt. Die PROFIenergy-Kommandos werden direkt von den PROFINET-Geräten interpretiert.

PROFIenergy-Controller (PE-Controller)

Der PE-Controller ist eine PLC, die bei unterlagerten Geräten den Ruhezustand aktiviert bzw. deaktiviert. Die Abschaltung bzw. Reaktivierung von einzelnen Produktionskomponenten oder ganzen Produktionslinien erfolgt dabei über das Anwenderprogramm. Über entsprechende Anweisungen (Funktionsbausteine) werden Kommandos (z.B. "Start_Pause" oder "End_Pause") an das unterlagerte Gerät abgesetzt. Das Senden der Kommandos erfolgt über das PROFINET-Kommunikationsprotokoll.

PROFIenergy-Entity (PE-Entity)

Die PE-Entity empfängt die PROFIenergy-Kommandos des PE-Controllers und setzt die Kommandos entsprechend um (z.B. durch Rücksenden eines Messwerts oder Aktivierung eines Energiesparmodus). Die Umsetzung der PE-Entity in einem PROFIenergy-fähigem Gerät ist geräte- und herstellerspezifisch.

Die PE-Entity kann z. B. umgesetzt werden:

- Innerhalb des Proxys eines Moduls: Die PE-Kommandos sind jeweils gültig für das adressierte Modul und ggf. unterlagerte Module.
- Innerhalb des Proxys einer Baugruppe: Die PE-Kommandos sind jeweils gültig für veschiedene Module innerhalb der Baugruppe.

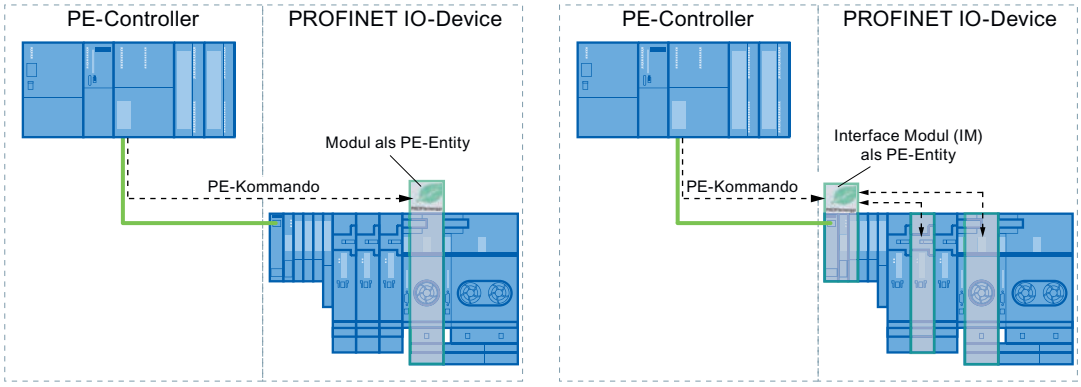

- Innerhalb eines Moduls ohne Proxy-Funktion innerhalb des Netzwerks: Die PE-Kommandos sind in diesem Fall nur für das jeweilige Modul gültig.

PROFIenergy-Anweisungen

- Anweisungen für IO-Controller
  - Mit der Anweisung "PE_START_END (Seite 5543)" wird auf einfachste Art und Weise der Ruhezustand der PROFINET-Geräte aktiviert oder deaktiviert (PROFIenergy-Kommandos "Start_Pause" und "End_Pause"). Das geschieht über eine positive und negative Signalflanke in der Anweisung.
  - Mit der Anweisung "PE_CMD (Seite 5547)" lassen sich alle PROFIenergy-Kommandos einschließlich "Start_Pause" und "End_Pause" übertragen. Darüber hinaus kann mit weiteren Kommandos z. B. der aktuelle Status der PROFINET-Geräte oder das Verhalten während der Pausen abgefragt werden.
  - Mit der Anweisung "PE_DS3_Write_ET200S (Seite 5552)" werden die Einstellungen für das Schaltverhalten von bis zu 8 Steckplätzen der ET 200S festgelegt. Die Anweisung gehört nicht zu den PROFIenergy-Anweisungen, ergänzt aber die PROFIenergy-Funktionen für eine ET 200S.
- Anweisung für iDevices
  Mit der Anweisung "PE_I_DEV (Seite 5576)" lässt sich PROFIenergy auch auf iDevices realisieren. Die Anweisung empfängt PROFIenergy-Kommandos auf dem iDevice und gibt sie an das Anwenderprogramm zur Bearbeitung weiter. Nach der Bearbeitung des Kommandos durch das Anwenderprogramm ruft es die Anweisung "PE_I_DEV (Seite 5576)" erneut auf, um die Quittung an den IO-Controller zu senden. Für diese Antworten steht ihnen für jedes Kommando ein entsprechender Hilfsbaustein zur Verfügung, der die Anweisung "PE_I_DEV (Seite 5576)" mit den Antwortdaten versorgt.

PROFIenergy-Kommandos (PE-Kommandos)

PE-Kommandos werden von dem PE-Controller an die PE-Entity gesendet. Das PE-Kommando kann entweder ein Steuerungskommando sein, um eine PE-Entity in einen Energiesparmodus zu schalten oder ein Kommando zum Auslesen eines Status oder Messwerts:

- PE-Kommandos zur Steuerung
  PROFIenergy unterstützt zwei Steuerungskommandos, die entweder über die Anweisung "PE_Start_End (Seite 5543)" oder die Anweisung "PE_CMD (Seite 5547)" ausgeführt werden können:
  - Start_Pause: Starten eines geeigneten Energiesparmodus (PE Energy-saving mode)
  - End_Pause: Beenden des Energiesparmodus (Wechsel in den Modus PE_ready_to_operate)
- PE-Kommandos zum Auslesen eines Status oder Messwerts
  Über die folgenden Statuskommandos können mit der Anweisung "PE_CMD (Seite 5547)" bestimmte Zustandsinformationen durch die Steuerung ausgelesen werden:
  - PE_Identify: Auslesen, welche PE-Kommandos von der PE-Entity unterstützt werden.
  - PEM_Status: Auslesen des momentan aktiven Modus einer PE-Entity (z. B. PE_ready_to_operate).
  - Query_Modes: Ausgabe einer Übersicht zu allen unterstützten Energiesparmodi, inklusive der Zeit- und Energieinformationen
  - Query_Measurement: Ausgabe der Messwerte einer PE-Entity

Applikationsbeispiele

Beispiele zur Anwendung der PROFIenergy-Anweisungen stehen über den Industry Online Support unter dem Beitrag "PROFIenergy - Energie sparen mit SIMATIC S7 (http://support.automation.siemens.com/WW/view/de/41986454)" zur Verfügung.

Siehe auch

Service und Support (http://support.automation.siemens.com/)

IO-Controller

PE_START_END: Energiesparmodus starten und beenden

Beschreibung

Mit der Anweisung "PE_START_END" wird für die angegebene PE-Entity (z.B. die ET 200S) die Energiesparpause gestartet und beendet.

Die Anweisung "PE_START_END" wird im PE-Controller verwendet, vorzugsweise wenn an den zugeordneten PE-Devices nur Feldgeräte angeschlossen sind, bei denen keine Energiedaten ausgelesen werden müssen. Zum Lesen der Energiedaten kann alternativ die Anweisung "PE_CMD (Seite 5547)" verwendet werden.

Die Energiesparmodi werden im Anwenderprogramm des PE-Controllers projektiert. Der tatsächlich eingenommene Energiesparmodus wird nach Ausführung von "PE_START_END" von der PE-Entity zurückgemeldet und am Parameter PE_MODE_ID ausgegeben.

Schreib- und Leseaufträge der Anweisung "PE_START_END"

Die Anweisung "PE_START_END" versendet intern mittels "WRREC (Seite 5458)" einen PROFIenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE_START_END" auf die Quittung von der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 5456)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 5456)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:

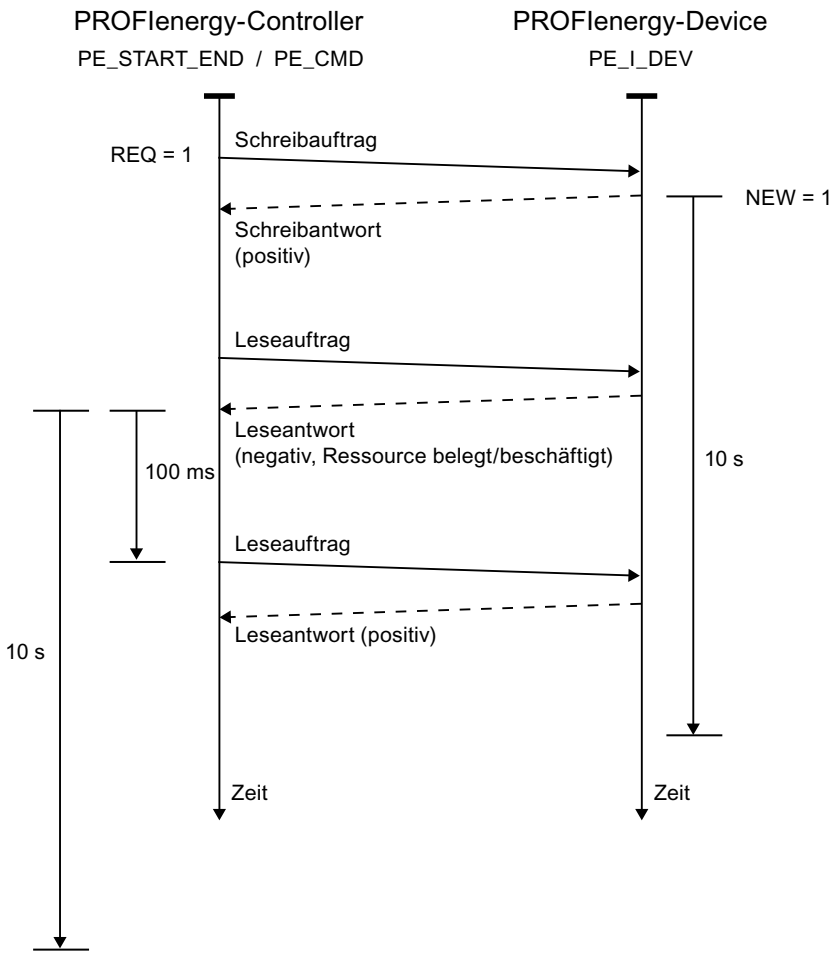

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_START_END":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
START	Input	BOOL	E, A, M, D, L	Senden des PE-Kommandos "Start_Pause" an die PE-Entity mit der Adresse am Parameter ID.
END	Input	BOOL	E, A, M, D, L	Senden des PE-Kommandos "End_Pause" an die PE-Entity mit der Adresse am Parameter ID.
ID	Input	DWORD	E, A, M, D, L oder Konstante	Adresse der PE-Entity (z.B. ET 200S). Die Adresse kann aus der Hardware-Konfiguration übernommen werden.
PAUSE_TIME	Input	TIME	E, A, M, D, L oder Konstante	Geplante Pausendauer. • Bereich: T#1MS bis T#24D20H31M23S647MS • Startwert: T#0MS
VALID	Output	BOOL	E, A, M, D, L	PE-Kommando erfolgreich abgesetzt.
BUSY	Output	BOOL	E, A, M, D, L	Bearbeitung des PE-Kommandos läuft noch.
ERROR	Output	BOOL	E, A, M, D, L	Bei der Bearbeitung trat ein Fehler auf. Die Fehlermeldung wird am Parameter STATUS ausgegeben.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus / Fehlernummer (siehe "Parameter STATUS").
PE_MODE_ID	Output	BYTE	E, A, M, D, L	Identifikationsnummer des Energiesparmodus (Energiesparlevel, der während der Pausendauer eingenommen wird).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter PAUSE_TIME

Mit dem Parameter PAUSE_TIME wird der PE-Entity die Dauer der Energiesparpause vorgegeben. In der PE-Entity wird geprüft, ob die vorgegebene Pausendauer ausreichend groß ist und umgesetzt werden kann. Die Mindest-Pausendauer (Time_min_Pause) muss dabei größer sein, als die Summe der Zeiten, die das Gerät zum Wechsel in den Energiesparmodus (Time_to_Pause) und zum Wechsel zurück in den Betriebsmodus (Time_to_Operate) benötigt.

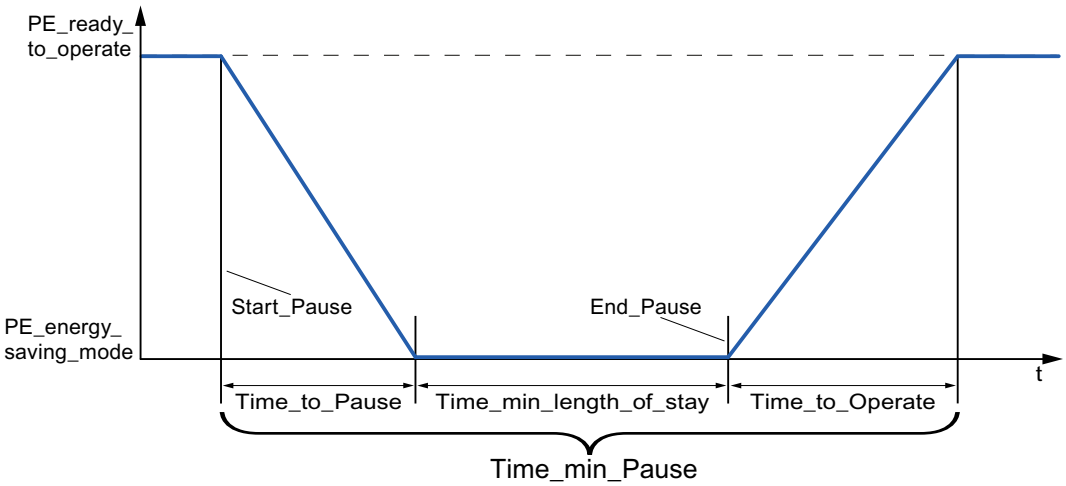

Bei der ET 200S erfolgt eine Prüfung, ob die geplante Pausendauer größer oder gleich der minimalen Pausendauer (PM-E_Pause_Min) ist, die auf der ET 200S hinterlegt ist. Diese beträgt fest 10 Sekunden. Wird eine kleinere Pause verwendet, bleiben die Powermodule (PM-E) der ET 200S eingeschaltet.

Es erfolgt kein automatisches Wiedereinschalten nach Ablauf der Pausenzeit, die Baugruppe bleibt bis zu dem "END"-Befehl im Zustand AUS. Damit wird ein unkoordiniertes Wiedereinschalten, was zu ungewollten Lastspitzen führen kann, verhindert.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#00: Kein Fehler • B#16#DE: Fehler beim Lesen des Datensatzes • B#16#DF: Fehler beim Schreiben des Datensatzes • B#16#C0: Fehlermeldung durch die Anweisung oder durch die intern verwendeten Kommunikationsanweisungen "RDREC (Seite 5456)" und "WRREC (Seite 5458)".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • 80: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch • FE: DP/PNIO-Profile - PROFIenergy-spezifischer Fehler

Feldelement	Name	Bedeutung
STATUS[3]	Error_Code_1	Fehlerkennung • Bei Error_Decode = **80**: – **80**: Gleichzeitig eine steigende Flanke an den Eingangsparametern START und END. – **81**: Längenkonflikt bei den Parametern CMD_PARAM und CMD_PARAM_LEN. – **82-8F**: weitere Fehlermeldungen (reserviert) • Bei Error_Decode = **FE**: – **01**: "Service Request ID" ungültig – **02**: Falsche "Request_Reference" – **03**: "Modifier" ungültig – **04**: "Data Structure Identifier RQ" ungültig – **05**: "Data Structure Identifier RS" ungültig – **06**: "PE energy-saving modes" werden nicht unterstützt – **07**: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten) – **08**: "Count" ungültig – **50**: Es steht kein passender "energy mode" zur Verfügung. – **51**: Angegebener Zeitwert wird nicht unterstützt. – **52**: "PE_Mode_ID" unzulässig
STATUS[4]	Error_Code_2	herstellerspezifische Erweiterung der Fehlerkennung

Hinweis

Fehlermeldungen der Anweisungen RDREC und WRREC

Die Anweisung "PE_START_END" verwendet zur Kommunikation die Anweisungen "WRREC (Seite 5458)" und "RDREC (Seite 5456)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 5458)" und "RDREC (Seite 5456)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 5468).

PE_CMD: Energiesparmodus starten und beenden / Zustandsinformationen auslesen

Beschreibung

Die Anweisung "PE_CMD" wird im PE-Controller verwendet und löst in der PE-Entity eine Energiesparpause aus oder beendet eine Pause. Zusätzlich können mit "PE_CMD" weitere Informationen und Energiemesswerte aus einer PE-Entity ausgelesen werden.

Die Anweisung kann vorzugsweise bei PE-Controllern eingesetzt werden, an deren zugeordneten PE-Devices Feldgeräte angeschlossen sind, von denen Energiemesswerte ausgelesen werden sollen. Ist dies nicht der Fall, kann zum starten und beenden der Pausen auch die Anweisung "PE_START_END (Seite 5543)" verwendet werden.

Übertragung der PROFIenergy-Kommandos (PE-Kommandos)

Die Anweisung "PE_CMD" überträgt ein PROFIenergy-Kommando an eine PE-Entity.

Die Anweisung kann auch eingesetzt werden, wenn das PROFIenergy-Profil zukünftig um weitere Kommandos erweitert wird. Die Kommandos, die nach dem aktuellen PROFIenergy-Profil verwendet werden können, sind in der Beschreibung der Parameter CMD und CMD_MODIFIER aufgelistet (siehe Tabelle "Parameter CMD und CMD_MODIFIER").

- Den einzelnen PE-Kommandos, die mit der Anweisung an die PE-Entity übertragen werden, sind definierte "Service_Request_IDs" zugeordnet. Die Service_Request_IDs 01...05 und 16 werden am Parameter CMD zugewiesen.
- Über den Parameter CMD_MODIFIER werden die beiden PE-Kommandos 04 (Query_Modes) und 16 (Query_Measurement) näher spezifiziert.
- Für einzelne PE-Kommandos werden am Parameter CMD_PARA zusätzliche Werte übergeben (siehe Beschreibung der einzelnen PE-Kommandos). Der Parameter CMD_PARA_LEN legt die Länge der Daten am Parameter CMD_PARA fest.

Die Kommandos werden ohne Plausibilitätstest übertragen. Die Antwortdaten der PE-Entity werden in dem durch ANY-Zeiger adressierten Datenbereich RESPONSE_DATA abgelegt (die Inhalte der Antworttelegramme sind in den Beschreibungen des jeweiligen PE-Kommandos enthalten).

Schreib- und Leseaufträge der Anweisung "PE_CMD"

Die Anweisung "PE_CMD" versendet mittels "WRREC (Seite 5458)" einen PROFIenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE_CMD" auf die Quittung der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 5456)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 5456)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:

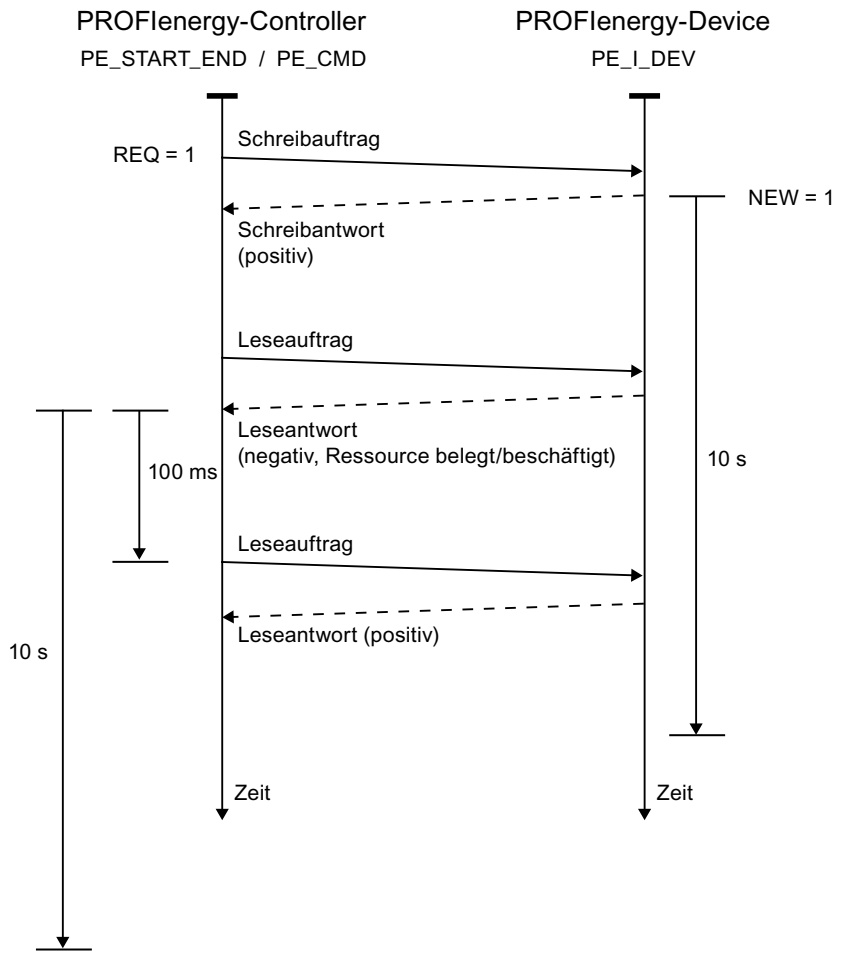

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_CMD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Startet die Übertragung des PE- Kommandos bei positiver Flanke.
ID	Input	DWORD	E, A, M, D, L oder Konstante	Adresse der PE-Entity Die Adresse kann bei einem PROFINET-IO Device aus der Hardware-Konfiguration übernommen werden.
CMD	Input	BYTE	E, A, M, D, L, P oder Konstante	Service-Request-ID des PROFIenergy-Kommandos entsprechend dem PROFIenergy-Profil (siehe "Parameter CMD und CMD_MODIFIER"). Nach Erweiterungen des PROFIenergy-Profils sind weitere Service-Request-IDs möglich.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD_MODIFIER	Input	BYTE	E, A, M, D, L, P oder Konstante	PROFIenergy-Unterkommando (nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER") Nach Erweiterungen des PROFIenergy-Profils sind weitere Unterkommandos möglich.
CMD_PARA	Input	ANY	E, A, M, D, L	Parameter für die PE-Kommandos: • Get mode: PE_mode_ID • Get measurement values: List of Measurement_Ids Es wird der komplette Service Data Request eingetragen.
CMD_PARA_LEN	Input	INT	E, A, M, D, L, P oder Konstante	Tatsächliche Länge der Parameter zu dem Kommando (<= Länge in CMD_PARA, wird von der Anweisung geprüft).
RESPONSE_DATA	InOut	ANY	E, A, M, D, L	PROFIenergy-Information Je nach Kommando gesamtes Antwort-Telegramm einschließlich Block-Header. Hinweis: Wenn der Puffer zu klein ist, werden nur die Anzahl Bytes eingetragen, wie im ANY-Zeiger angegeben.
VALID	Output	BOOL	E, A, M, D, L	Kommando erfolgreich abgesetzt.
BUSY	Output	BOOL	E, A, M, D, L	Kommandobearbeitung läuft noch.
ERROR	Output	BOOL	E, A, M, D, L	Bei der Bearbeitung trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	Bausteinstatus / Fehlernummer (siehe "Parameter STATUS"):

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter CMD und CMD_MODIFIER

CMD	CMD_ MODIFIER	PROFIenergy-Kommando	Beschreibung
01	0	Start_Pause (Seite 5566)	Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus.
02	0	End_Pause (Seite 5567)	Beenden des Energiesparmodus.
03	1	Query_Modes - List energy saving Modes (Seite 5567)	Ausgeben der unterstützten Energiesparmodi.
	2	Query_Modes - Get Mode (Seite 5568)	Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.
04	0	PEM_Status (Seite 5570)	Status des Energiesparmodus abfragen.
05	0	PE_Identify (Seite 5572)	Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.

CMD	CMD_ MODIFIER	PROFIenergy-Kommando	Beschreibung
16	1	Query_Measurement - Get_Measurement_List (Seite 5573)	Auflistung der durch die PE-Entity unterstützten Messwerte.
	2	Query_Measurement - Get_Measurement_Values (Seite 5575)	Ausgabe der Messwerte der PE-Entity.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#**00**: Kein Fehler • B#16#**DE**: Fehler beim Lesen des Datensatzes • B#16#**DF**: Fehler beim Schreiben des Datensatzes • B#16#**C0**: Fehlermeldung der intern verwendeten Kommunikationsanweisungen "RDREC (Seite 5456)" und "WRREC (Seite 5458)".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • **80**: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch • **FE**: DP/PNIO-Profile - PROFIenergy-spezifischer Fehler
STATUS[3]	Error_Code_1	Fehlerkennung • Bei Error_Decode = **80**: – **81**: Längenkonflikt bei den Parametern CMD_PARA und CMD_PARA_LEN bzw. maximale Datensatzlänge (4095 Byte) überschritten. – **82-8F**: weitere Fehlermeldungen (reserviert) • Bei Error_Decode = **FE**: – **01**: "Service Request ID" ungültig – **02**: Falsche "Request_Reference" – **03**: "Modifier" ungültig – **04**: "Data Structure Identifier RQ" ungültig – **05**: "Data Structure Identifier RS" ungültig – **06**: "PE energy-saving modes" werden nicht unterstützt – **07**: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten) – **08**: "Count" ungültig – **50**: Es steht kein passender Energiesparmodus (energy mode) zur Verfügung. – **51**: Angegebener Zeitwert wird nicht unterstützt. – **52**: "PE_Mode_ID" ungültig
STATUS[4]	Error_Code_2	herstellerspezifische Erweiterung der Fehlerkennung

Hinweis

Fehlermeldungen der Anweisungen RDREC und WRREC

Die Anweisungen "PE_CMD" verwenden zur Kommunikation die Anweisungen "WRREC (Seite 5458)" und "RDREC (Seite 5456)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 5458)" und "RDREC (Seite 5456)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 5468).

PE_DS3_Write_ET200S: Schaltverhalten der Powermodule einstellen

Beschreibung

Die Anweisung "PE_DS3_Write_ET200S" sendet grundsätzliche Einstellungen zum Schaltverhalten der Powermodule an die ET 200S. Mit der Anweisung "PE_DS3_Write_ET200S" kann das Schaltverhalten für bis zu 8 Slots der ET 200S (z.B. für Powermodule) festgelegt werden.

Hinweis

Diese Anweisung ist nicht Bestandteil des PROFIenergy-Profils, sondern ergänzt SIMATIC-spezifische Funktionen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_DS3_Write_ET200S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ENABLE	Input	BOOL	E, A, M, D, L	Eine positive Flanke stößt die Übertragung des Datensatzes an. Der Datensatz muss nach Spannung AUS/EIN erneut übertragen werden.
ID	Input	DWORD	E, A, M, D, L oder Konstante	Adresse der ET 200S Die Adresse kann aus der Hardware-Konfiguration übernommen werden.
SLOT_NO_X	Input	INT	E, A, M, D, L, P oder Konstante	Steckplatznummer des X. schaltbaren Powermoduls.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FUNC_X	Input	INT	E, A, M, D, L, P oder Konstante	Funktion der Baugruppe auf diesem Steckplatz. Über den Parameter FUNC_X legen Sie das Schaltverhalten des PM-E (Powermodul der ET 200S) fest: • FALSE: – Bei "PAUSE_START": -Kein Einfluss auf PM-E -PM-E bleibt eingeschaltet – Bei "PAUSE_STOP": -Schaltet PM_E wieder ein • TRUE: – Bei "PAUSE_START": -Schaltet PM_E aus – Bei "PAUSE_STOP": -Schaltet PM-E wieder ein
BUSY	Output	BOOL	E, A, M, D, L	Übertragung noch nicht abgeschlossen.
DONE	Output	BOOL	E, A, M, D, L	Übertragung ohne Fehler beendet.
ERROR	Output	BOOL	E, A, M, D, L	Übertragung mit Fehler beendet.
STATUS	Output	DWORD	E, A, M, D, L, P	Fehlernummer (siehe Parameter STATUS der Anweisung "PE_Start_End (Seite 5543)")

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_WOL: Energiesparmodus über WakeOnLan starten und beenden

Beschreibung PE_WOL

Beschreibung

Die Anweisung "PE_WOL" sendet die PROFIenergy-Kommandos "Start_Pause" und "End_Pause" an mehrere PROFIenergy-fähige Devices im PROFINET I/O Systemen.

Über die Anweisung ist die Koordination mehrerer PE-Devices möglich, vorausgesetzt die PE-Devices unterstützen die Funktion "Wake on LAN" über eine UDP-Verbindung.

Die Anweisung "PE_WOL" ist nur auf einer CPU mit integrierter Ethernet-Schnittstelle ausführbar. Diese CPU muss in der Lage sein, Bausteine mit einer Größe von ca. 40kB zu laden. Sie können den Baustein nicht mit PROFINET I/O Systemen einsetzen, wenn diese über einen Ethernet CP angebunden sind.

Die Anweisung "PE_WOL" wird asynchron abgearbeitet.

Definition: Wake on LAN

Die Wake On LAN Funktionalität erlaubt Datenverarbeitungseinrichtungen die Wiederaufnahme der Arbeit aus einem nahezu ausgeschalteten Zustand heraus, durch den Empfang eines speziellen Ethernet-Pakets.

Damit dieses Verfahren funktionieren kann, muss die Datenverarbeitungseinrichtung über einen Netzwerk-Controller verfügen, der für den Empfang eines solchen Pakets ausgerüstet ist.

Dieses Paket (Magic Packet™) hat eine spezielles Format. Es beinhaltet 15mal die MAC Adresse des Netzwerkadapters.

Auswahl der Devices

Die Auswahl der Devices erfolgt über den Anwender-Datenbaustein am Parameter PENERGY. Der Anwender-DB stellt die Datenbasis für die Bearbeitung mehrerer Devices dar.

Vor der Initialisierung von "PE_WOL" müssen Sie in dem Anwender-DB mindestens folgende Informationen hinterlegen:

- ID des PROFINET I/O Systems
- Daten der Verbindung, die für "Wake On LAN" genutzt wird
- Port-Nummer, die für "Wake On LAN" genutzt wird
- Für jedes Device
  - Pausenzeit (PauseTime)
  - Schalten des Device in den PE_SLEEP_MODE (EnableSleep)

Über den den Parameter COM_RST initialisieren Sie die Anweisung "PE_WOL". Nach der Initialisierung werden die im Anwender-DB hinterlegten Aufträge nacheinander abgearbeitet.

Die folgende Grafik zeigt beispielhaft, wie das PE-Kommando "Start_Pause" an mehrere Devices übertragen wird:

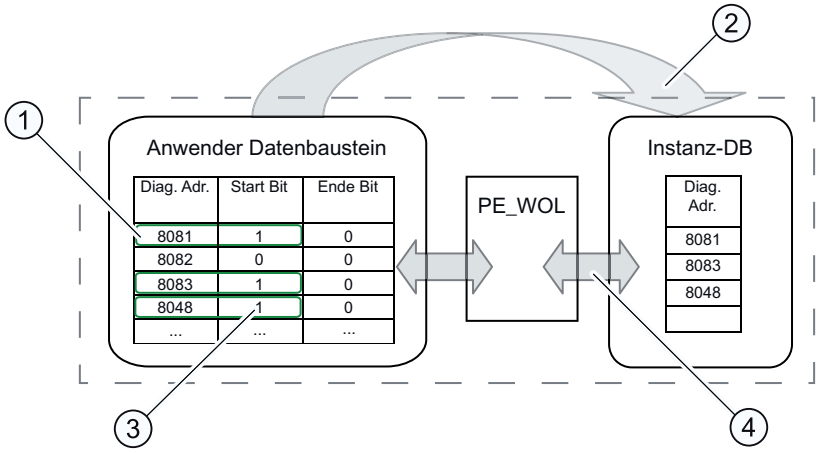

(1) Schritt 1: Das Bit "CmdStartPause" der abzuschaltenden Devices wird durch den Anwender auf "1" gesetzt.

(2) Schritt 2: Die Diagnose-Adressen der abzuschaltenden Devices (CmdStartPause = "1") werden in die Warteschlange eingekettet.

(3) Schritt 3: Das Bit "CmdStartPause" wird automatisch zurückgesetzt, nachdem die Aufträge eingekettet wurden.

(4) Schritt 4: Die Anweisung "PE_WOL" bearbeitet die Aufträge, sobald diese eingekettet sind.

Über die Parameter START und END kann ein PROFIenergy-Kommando "CmdStartPause" bzw. "CmdEndPause" für alle erkannten Geräte im PROFINET IO System gesendet werden.

Über den Parameter STATUS wird der Stand der Auftragsbearbeitung sowie mögliche Fehler bei der Bearbeitung ausgegeben.

Bedienung der Anweisung über den Anwender-DB

Die Bedienung der Anweisung "PE_WOL" kann ausschließlich über den Anwender-DB erfolgen. Hierbei gilt ein grundsätzlicher Ablauf:

1. Anwahl des auszuführenden Kommandos für ein Device:
   - START_PAUSE ("CmdStartPause" im Anwender-DB)
   - ENDE_PAUSE ("CmdEndPause" im Anwender-DB)
   - UPDATE_STATUS ("CmdUpdateStatus" im Anwender-DB)
2. Setzen des Bits zur Aktualisierung ("Update" im Header des Anwender-DBs)
   Zwischen zwei Aktualisierungen sollte mindestens ein CPU Zyklus verstreichen mit einem "Update" = False, da sonst eine Flankenerkennung nicht gewährleistet werden kann.

Priorisierung der PE-Kommandos

Die folgende Grafik zeigt den zeitlichen Ablauf der drei möglichen Kommandos.

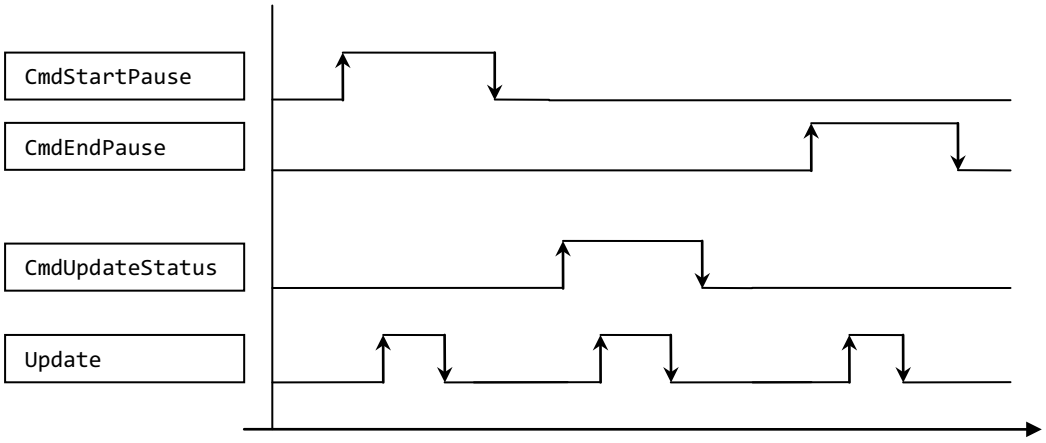

Diese werden nacheinander abgearbeitet, unabhängig davon ob der jeweils vorherige Kommandoaufruf erfolgreich war oder mit Fehler beendet wurde.

Bei gleichzeitigem Setzen von, z. B. "CmdEndPause" und "CmdUpdateStatus" wird nur ein Kommando ausgeführt. Es liegt innerhalb des Bausteins eine Priorisierung vor:

- Das Kommando "CmdStartPause" hat die höchste Priorität und wird folglich immer ausgeführt, sofern angewählt.
- Das Kommando "CmdEndPause" hat die zweithöchste Priorität.
- Das Kommando "CmdUpdateStatus" hat die niedrigste Priorität.

Werden alle drei Kommandos gleichzeitig gesetzt, bleiben die nicht ausgeführten Kommandos vorgewählt. In diesem Falle wird die nächste steigende Flanke das nächste Kommando ausführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_WOL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
COM_RST (Seite 5556)	Input	BOOL	E, A, M, D, L	Setzt den Baustein zurück und führt eine Neuinitialisierung durch. Solange hier ein True gesetzt ist, wird die Initialisierung eingeleitet, jedoch noch nicht vollständig durchgeführt. Erst die fallende Flanke setzt die Initialisierung fort und wechselt nach der Initialisierung in den normalen Betriebsmodus.
START (Seite 5558)	Input	BOOL	E, A, M, D, L	Eine steigende Flanke führt ein PROFIenergy-Kommando "CmdStartPause" für alle erkannten Geräte durch, die diese Funktion unterstützen.
END (Seite 5559)	Input	BOOL	E, A, M, D, L	Eine steigende Flanke führt ein PROFIenergy-Kommando "CmdEndPause" für alle erkannten Geräte durch, die diese Funktion unterstützen.
PENERGY (Seite 5559)	InOut	ANY	D	Zeiger auf den Anwender-DB, der die Datenbasis für die Bearbeitung mehrerer Devices enthält.
STATUS (Seite 5564)	Output	DWORD	E, A, M, D, L	Status-/Fehlernummer zum aktuellen Zustand der Anweisung (siehe "Parameter STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter COM_RST

Ablauf der Initialisierungsroutine

Über den Parameter COM_RST starten Sie die Initialisierung der Anweisung "PE_WOL".

Das folgende Ablaufdiagramm zeigt die Initialisierungsroutine.

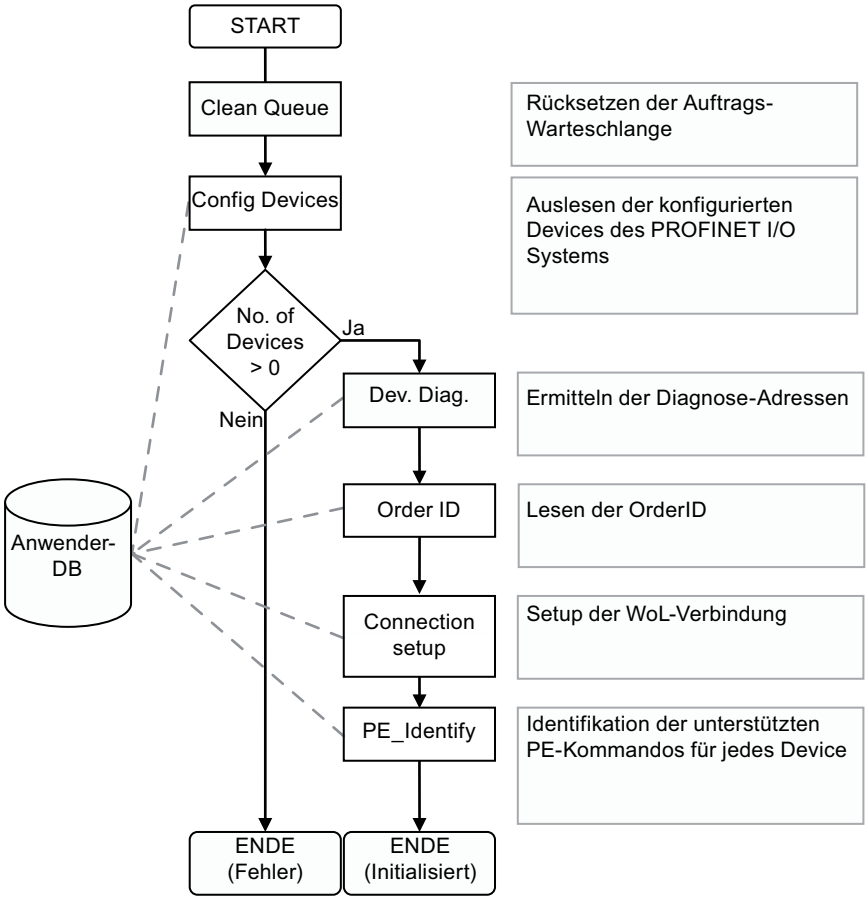
START
Clean Queue
Rücksetzen der Auftrags-Warteschlange
Config Devices
Auslesen der konfigurierten Devices des PROFINET I/O Systems
No. of Devices > 0
Ja
Nein
Dev. Diag.
Ermitteln der Diagnose-Adressen
Order ID
Lesen der OrderID
Anwender-DB
Connection setup
Setup der WoL-Verbindung
PE_Identify
Identifikation der unterstützten PE-Kommandos für jedes Device
ENDE (Fehler)
ENDE (Initialisiert)

Parameter START

Ablauf des CmdStartPause Kommandos

Das Ablaufdiagramm zeigt die intern genutzten Funktionen und Interaktion mit einem Device beim Ausführen des CmdStartPause Kommandos.

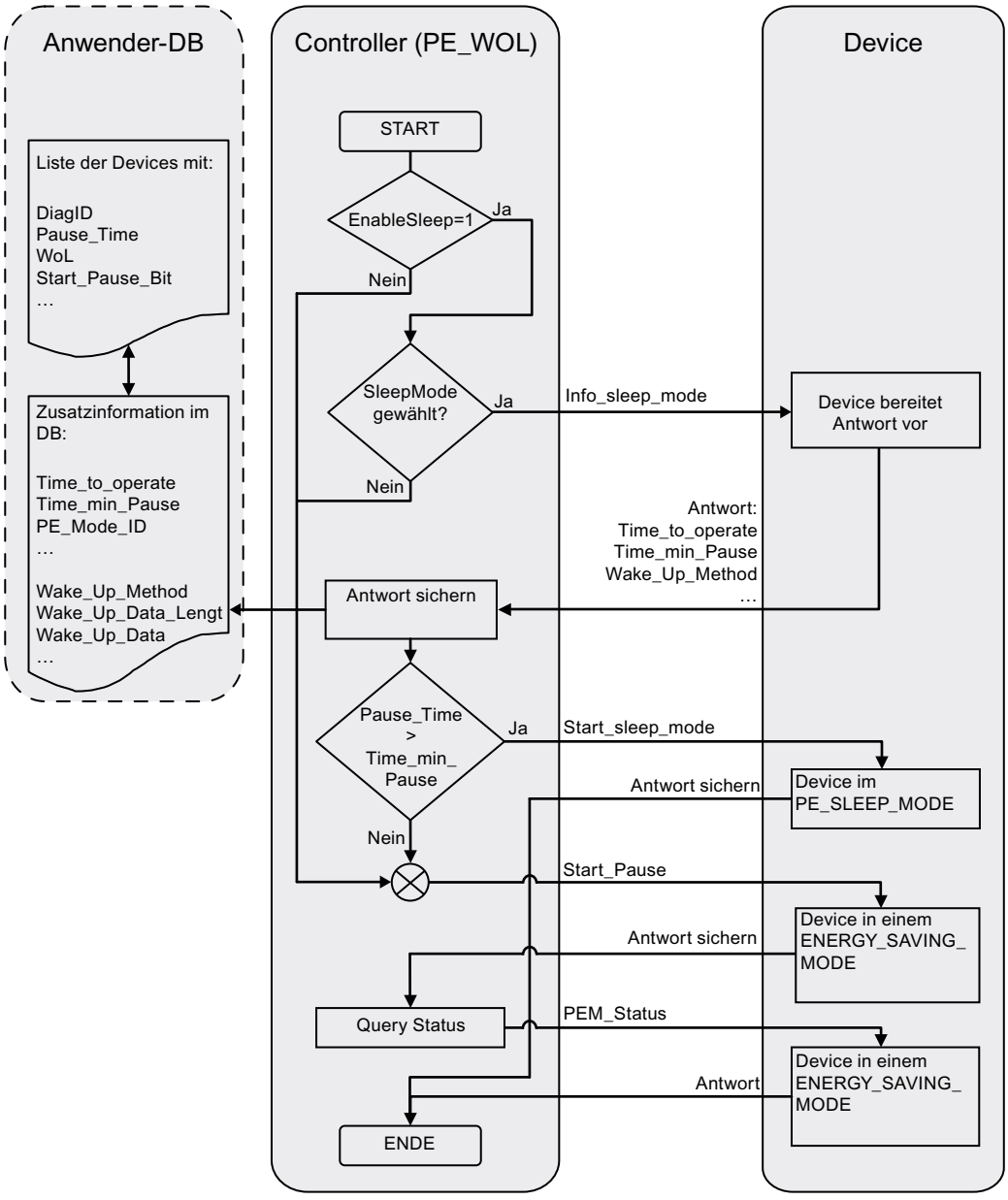

Parameter END

Ablauf des CmdEndPause Kommandos

Das Ablaufdiagramm zeigt die intern genutzten Funktionen und Interaktion mit einem Device beim Ausführen des CmdEndPause Kommandos.

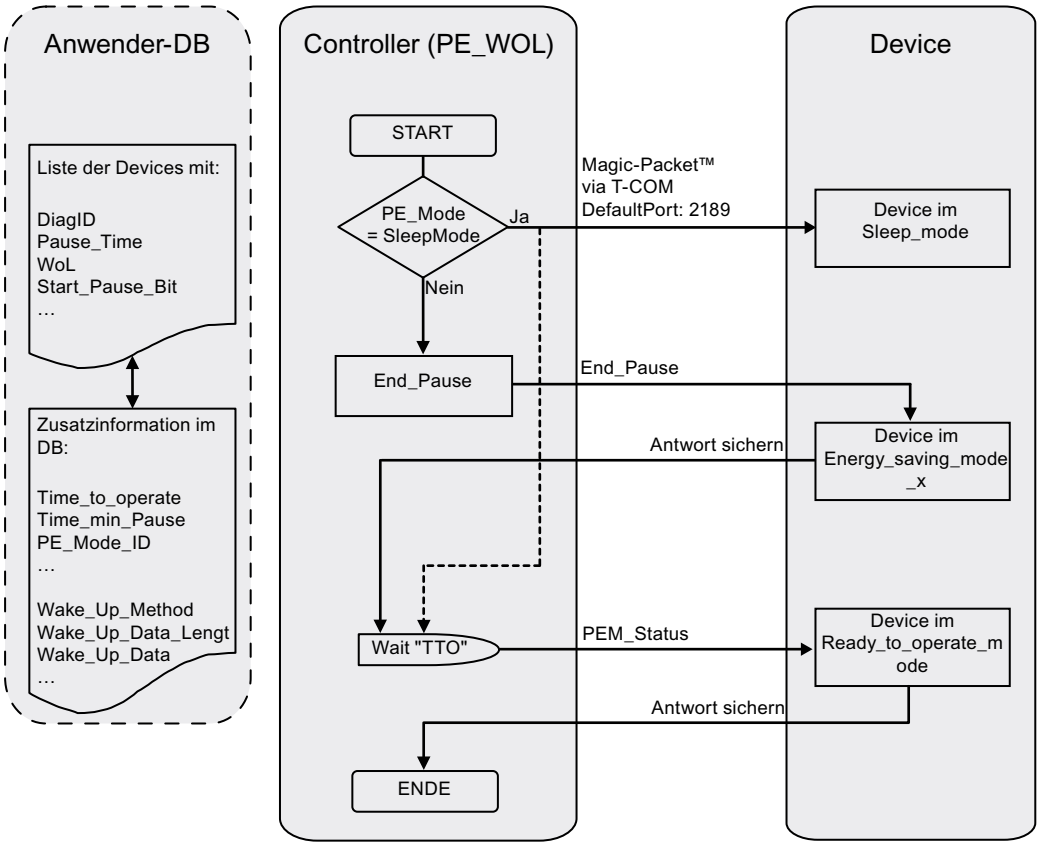

Parameter PENERGY

Datenbaustein am Parameter PENERGY

Der Anwender-DB für die PROFIenergy Anweisung "PE_WOL" stellt eine Datenbasis für die Bearbeitung mehrerer Devices dar.

Der Datenbaustein ist generell in zwei Teile gegliedert. Diese sind:

- Kopfteil 110 Byte (Header)
- Device-Teil für maximal 256 Devices mit je 100 Byte (Device). Diese enthalten:
  - Device-spezifische Daten (Device)
  - PROFIenergy-spezifische Daten (PE)
  - Daten der Auftragsbearbeitung (Task)
  - Benutzerdaten (UserData)

Die einzelnen Teilbereiche haben eine bestimmte Größe, die konstant ist. Insgesamt ergeben diese eine konstante Größe von 25.746 Byte für den Datenbaustein.

Verbindungsparameter "Connection"

Die Anweisung "PE_WOL" reserviert eine Verbindungsressource aus dem Bereich der "Open User Communication". Diese wird als UDP Verbindung genutzt. Hierfür müssen im Datenbaustein folgende Parameter definiert werden:

- Verbindungs-ID (Parameter "Connection.id")
  Die Verbindungs-ID ist eine ganze Zahl, die im Bereich von 1 bis 32 liegt. Die Standardeinstellung ist 31. Sie dient der Identifikation der firmwareseitig zugeordneten Kommunikationsressourcen, wie Sende- und Empfangspuffer.
  Die Verbindungs-ID muss CPU-weit eindeutig vergeben werden.
- Port-Nummer, die für die "Wake On LAN" Funktion genutzt wird (Parameter "Header.PortNo")
  Nummer des UDP-Ports über die ein "Wake On LAN" Paket versendet wird. Diese Port-Nummern sind Bestandteil der Kommunikationsressourcen, die firmwareseitig durch die Verbindungs-ID identifiziert und bereitgestellt werden. Die hier verwendete Standardeinstellung für den Port von 2189 ist durch die IANA zurzeit nicht zugeordnet. Die Port-Nummer wird auf die Verbindungsprojektierung übertragen. Hierdurch werden die Parameter "Connection.local_tsap_id[1]" und "Connection.rem_tsap_id[1]" beeinflusst.
- Schnittstellen-ID (Parameter "Connection.local_device_id")
  Die Schnittstellen-ID ist ebenfalls Teil der Verbindungsbeschreibung. Diese ID identifiziert die zu nutzende Schnittstelle einer CPU für diese Verbindung. Es sind mehrere gültige Werte möglich.
  Diese müssen jedoch auf die verwendete CPU und Schnittstelle angepasst werden:
  - B#16#01 bei ET200S CPU oder WinAC RTX mit Ethernetschnittstelle in Subslot IF1
  - B#16#02 bei CPU 315(F)-2PN/DP oder CPU 317(F)-2PN/DP
  - B#16#03 bei CPU 319(F)-2PN/DP
  - B#16#05 bei CPU 41x(F)-3PN/DP
  - B#16#06 bei WinAC RTX mit Ethernetschnittstelle in Subslot IF2
  - B#16#0B bei WinAC RTX mit Ethernetschnittstelle in Subslot IF3
  - B#16#0F bei WinAC RTX mit Ethernetschnittstelle in Subslot IF4

Aufbau des Datenbausteins

Der Datenbaustein hat den folgenden Aufbau:

Name	Datentyp	Offset	Kommentar
Header	STRUCT	-	Kopfinformationen
Update (1)	BOOL	0.0	Signal zur Anzeige einer Änderung im Datenbereich. • True= zeigt eine Änderung durch den Anwender an. • False= zeigt die Übernahme der Änderungen an.
Initialized	BOOL	0.1	Signal zur Anzeige der abgeschlossenen Initialisierung. • True=Initialisierung fertig. • False= zeigt an, dass der Baustein nicht initialisiert ist.
LinkUp	BOOL	0.2	Zeigt eine erfolgreiche Konfiguration der Ethernetschnitt-stelle an. • True=Schnittstelle einsatzbereit. • False=Schnittstelle noch nicht konfiguriert.
LinkDown	BOOL	0.3	Zeigt eine unkonfigurierte Schnittstelle an. • True=Schnittstelle ist nicht konfiguriert. • False=Schnittstelle wird gerade oder ist konfiguriert.
PROFINET_ID (1)	INT	2.0	ID des PROFINET I/O Systems
Reserved	ARRAY [1..37] OF BYTE	4.0	Reserviert
LastDeviceID	INT	42.0	Enthält die größte Device-ID in diesem PROFINET I/O System.
PortNo (1)	INT	44.0	Port-Nummer, die für die "Wake On LAN" Funktion genutzt wird (Default= 2189).
Connection	STRUCT	-	Enthält die Verbindungsprojektierung der "Wake on LAN" Verbindung.

Name	Datentyp	Offset	Kommentar
block_length	WORD	46.0	Länge der Struktur (immer B#16#40).
id (1)	WORD	48.0	Verbindungs-ID
connection_type (1)	BYTE	50.0	Verbindungstyp (UDP = B#16#13)
active_est (1)	BOOL	51.0	Aktiver Verbindungsaufbau (bei UDP immer passiv)
local_device_id (1)	BYTE	52.0	Enthält die Schnittstellen-ID (CPU abhängig)
local_tsap_id_len (1)	BYTE	53.0	Enthält die Länge in Bytes des eigenen/lokalen UDP-Ports
rem_subnet_id_len (1)	BYTE	54.0	ungenutzt (immer B#16#00)
rem_staddr_len (1)	BYTE	55.0	Enthält die Länge der entfernten IP-Adresse oder B#16#00.
rem_tsap_id_len (1)	BYTE	56.0	Enthält die Länge in Bytes des entfernten UDP-Ports.
next_staddr_len (1)	BYTE	57.0	Enthält die Länge der Default-Router-Adresse (nicht relevant).
local_tsap_id (1)	ARRAY[1..8] OF INT	58.0	Enthält die eigene/lokale Port-Nummer.
rem_subnet_id (1)	ARRAY[1..6] OF BYTE	74.0	ungenutzt (immer B#16#00)
rem_staddr (1)	ARRAY[1..6] OF BYTE	80.0	Enthält die entfernte IP-Adresse.
rem_tsap_id (1)	ARRAY[1..8] OF INT	86.0	Enthält die entfernte UDP-Port-Nummer.
next_staddr (1)	ARRAY[1..6] OF BYTE	102.0	irrelevant
spare (1)	WORD	108.0	
Device	ARRAY[1..256] OF STRUCT	110.0	Array der Devices
Device	STRUCT	110.0	Enthält Daten für jedes Device
DiagID	WORD	110.0	Diagnoseadresse des Device. Wird durch die Hardware-Konfiguration zugewiesen.
InterfaceID	WORD	112.0	Diagnoseadresse der Device-Schnittstelle. Wird durch die Hardware-Konfiguration zugewiesen.
MACAdr	ARRAY[1..6] OF BYTE	114.0	Enthält die MAC-Adresse des Device.
IPAdr	ARRAY[1..4] OF BYTE	120.0	Enthält die IP-Adresse des Device.
OrderID	STRUCT	124.0	Enthält die OrderID des Device.
MxLen	BYTE		Enthält die maximale Länge des Array.
ActLen	BYTE		Enthält die aktuelle Anzahl Daten im Array.
Data	ARRAY[1..20] CHAR		Enthält die Daten.
PE	STRUCT	146.0	PROFIenergy spezifische Daten.

Name	Datentyp	Offset	Kommentar
ModeID	BYTE	146.0	PE_MODE_ID nach PROFIenergy Spezifikation.
Result	BYTE	147.0	PEErrorCode nach PROFIenergy Spezifikation.
PauseTime (1)	TIME	148.0	Enthält die Pausenzeit in ms.
TimeToPause	TIME	152.0	Enthält die Zeit, die das Device benötigt, um in den Pause-Modus zu gelangen.
TimeToOperate	TIME	156.0	Enthält die Zeit, die das Device benötigt, um in den Betriebsmodus zu gelangen.
MinSleepTime	TIME	160.0	Enthält die minimale Zeit des Devices im PE_SLEEP_MODE.
SleepToOperate	TIME	164.0	Enthält die Zeit, die das Device benötigt um vom PE_SLEEP_MODE betriebsbereit zu werden.
StatusOperate	BOOL	168.0	Zeigt den Betriebs-Modus des Device an.
StatusPause	BOOL	168.1	Zeigt den Pause Modus des Devices an.
StatusSleep	BOOL	168.2	Zeigt den PE_SLEEP_MODE des Device an.
StatusTransitOK	BOOL	168.3	Zeigt den Übergang von einem Energiezustand in einen anderen an.
StatusInTransit	BOOL	168.4	Zeigt einen aktuellen Zustandsübergang an.
StatusTransitNOK	BOOL	168.5	Zeigt an, dass der Zustandswechsel nicht erfolgreich war.
StatusError	BOOL	168.6	Zeigt einen Fehler mit dem Device an.
StatusRetryEx	BOOL	168.7	Zeigt eine nicht erfolgreiche Ausführung eines Kommandos an. Es wird nicht mehr versucht, dieses Kommando erneut auszuführen.
CmdStartPause (1)	BOOL	169.0	Platziert ein START_PAUSE Kommando für dieses Device in die Warteschleife.
CmdEndPause (1)	BOOL	169.1	Platziert ein END_PAUSE Kommando für dieses Device in die Warteschleife.
CmdUpdateStatus (1)	BOOL	169.2	Platziert ein PEM_STATUS Kommando für dieses Device in die Warteschleife.
EnableSleep (1)	BOOL	169.3	Erlaubt den PE_SLEEP_MODE für dieses Device. • True= Device soll in den PE_SLEEP_MODE, wenn die Pausenzeit lang genug ist • False=Device darf nicht in den PE_SLEEP_MODE.
Services	WORD	170.0	Zeigt alle unterstützten PROFIenergy Services an.
UserData (2)	ARRAY[1..24] OF BYTE	172.0	Nutzer definierte Daten
Task	STRUCT	196.0	Auftragsbearbeitung

Name	Datentyp	Offset	Kommentar
Cmd	BYTE	196.0	Interne Bits für die Auftragsbearbeitung
CmdJ	BYTE	197.0	Internal Bits für die Auftragsbearbeitung
TimeStart	BOOL	198.0	Startet eine Verzögerungszeit.
TimeStarted	BOOL	198.1	Die Verzögerungszeit wurde gerade gestartet.
TimeDone	BOOL	198.2	Zeigt den Ablauf der Verzögerungszeit an.
Done	BOOL	198.3	Zeigt an, dass dieser Auftrag beendet wird.
DelayedCmd	BOOL	198.4	Zeigt an, dass ein verzögertes Kommando noch ansteht.
IsV1_0	BOOL	198.5	Zeigt an, dass dieses Gerät ein Spec. V1.0 Gerät ist.
IsWakeOnLAN	BOOL	198.6	Zeigt an, dass dieses Gerät per "Wake On LAN" geweckt wird.
RetryCount	BYTE	199.0	Wiederholungszähler für PE_COMMANDS
Duration	TIME	200.0	Enthält den Zeitwert für die Verzögerung in ms.
StartTime	TIME	204.0	Enthält den Startzeitpunkt der Verzögerungszeit.
MachineState	INT	208.0	Enthält den internen Status des Auftrags.

(1) Durch den Anwender zu befüllen.

(2) Für den Anwender frei verwendbar.

Parameter STATUS

Parameter STATUS

Der Ausgabewert am Parameter STATUS ist in drei Bereiche unterteilt:

- Bit 31 bis 24: MESSAGE
- Bit 23 bis 16: LOCATION
- Bit 15 bis 0: INFORMATION

Die Bedeutung der einzelnen Fehlercodes der drei Bereiche entnehmen Sie den folgenden Tabellen:

Tabelle 4-192 Mögliche Werte für MESSAGE

Fehlercode (W#16#...)	Beschreibung
00	Kein Fehler.
50	Anweisung initialisiert.
51	Konfiguration des PROFINET I/O Systems wird ermittelt.
52	Die Anweisung konnte keine konfigurierten Devices am PROFINET I/O System ermitteln.
53	Die logischen Adressen der konfigurierten Devices werden ermittelt.
54	Schnittstelleninformationen der Devices werden ausgelesen.
55	I&M Daten (nur Datensatz 0) der konfigurierten Devices werden ermittelt.
56	PROFINET-Schnittstelle für den Versand des "Wake on LAN" MagicPaket™ über UDP wird konfiguriert.
57	PROFIenergy-Fähigkeiten der angeschlossenen Geräte wird ermittelt.
62	Ungültige PROFINET I/O System ID erkannt. Die verursachende Nummer wird im Feld INFORMATION angezeigt.

Fehlercode (W#16#...)	Beschreibung
70	Die Anweisung ist initialisiert und bearbeitet Aufträge. Der Wert im Feld INFORMATION enthält die Anzahl aktuell aktiver Aufträge.
80	Die Anweisung hat während der Bearbeitung von Aufträgen die Initialisierung verloren. Dies tritt im Allgemeinen dann auf, wenn entweder der Instanz-DB oder der Anwender-DB neu geladen werden.
FF	Unbekannter Fehler aufgetreten.

Tabelle 4-193 Mögliche Werte für LOCATION

Fehlercode (W#16#...)	Beschreibung
00	Die Anweisung ist entweder nicht initialisiert oder im Leerlauf.
70	Die Anweisung wartet auf Aufträge.
71	Die Anweisung hängt einen Auftrag in die Auftragsliste ein.
72	Die Anweisung bereitet den Auftragsversand vor.
73	Die Anweisung versendet einen Auftrag an ein Device.
74	Die Anweisung wartet auf eine Antwort vom Device.
75	Die Anweisung wertet die Antwort des Devices aus.
76	Die Anweisung entfernt den Auftrag aus der Auftragsliste.
FF	Unbekannter Fehler aufgetreten.

Tabelle 4-194 Mögliche Werte für INFORMATION

Fehlercode (W#16#...)	Beschreibung
0000	Keine Zusatzinformationen vorhanden und keine Aufträge aktiv.
0001 -00FF	Es werden aktuell 1 – 255 Aufträge bearbeitet.
8001	Fehler am 1. Parameter
8002	Fehler am 2. Parameter
8003	Fehler am 3. Parameter
8004	Fehler am 4. Parameter
8005	Fehler am 5. Parameter. Dieser Fehler wird gemeldet, sofern es keine oder eine ungültige Verschaltung zum Anwender-DB gibt. Mögliche Ursachen: • Der Anwender-DB ist zu klein. • Der Anwender-DB ist schreibgeschützt. • Der Anwender-DB ist nicht im RAM vorhanden. • Der Anwender-DB ist ungültig für die eingesetzte CPU.
8100	Es wurde versucht, mehr als die maximal 256 möglichen Aufträge zu platzieren. Es handelt sich hierbei um einen temporären Fehler, der sich mit dem Abschluss einiger Aufträge behebt. Der platzierte Auftrag wurde nicht entgegen genommen und muss erneut platziert werden.
8200	Es wurde versucht, ein ungültiges oder nicht unterstütztes PROFIenergy-Kommano (PE_COMMAND) zu senden.

Fehlercode (W#16#...)	Beschreibung
84xx	Es ist ein Kommunikationsfehler aufgetreten. Die Nummer des Devices, das den Fehler erzeugt hat, wird an "xx" ausgegeben.
85xx	Es wurde ein Fehler vom Device xx zurückgemeldet. Die Nummer des Devices, das den Fehler erzeugt hat, wird an "xx" ausgegeben.
8600	Die angeforderte WakeUp-Methode wird derzeit nicht unterstützt.
FFFF	Unbekannter Fehler aufgetreten.

PROFIenergy-Kommandos

PE-Kommando "Start_Pause"

Beschreibung

Mit dem PE-Kommando "Start_Pause" starten Sie den Energiesparmodus. Das Kommando Start_Pause kann verwendet werden um:

- Die PE-Entity vom Status "Betriebsbereit" (PE_ready_to_operate) in einen Energiesparmodus (PE_energy_saving_mode) zu schalten.
- Die PE-Entity von einem Energiesparmodus in einen anderen zu schalten.
  Der Energieverbrauch kann durch das Umschalten des Energiesparmodus steigen oder sinken.

Aufruf des PE-Kommandos "Start_Pause"

Der Aufruf des Kommandos "Start_Pause" mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	1	Aufruf des PE-Kommandos "Start_Pause".
CMD_MODIFIER	0	Bei Kommando "Start_Pause" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	4	Länge des Parameters CMD_PARA von 4 Byte.
CMD_PARA	ANY	ANY-Zeiger auf Wert für "Pause_Time" (TIME).

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe Anweisung "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID	1 bis 255	BYTE	Identifikationsnummer des Energiesparmodus
Reserviert	0	BYTE	-

PE-Kommando "End_Pause"

Beschreibung

Mit dem PE-Kommando "End_Pause" beenden Sie den Energiesparmodus der PE-Entity.

Aufruf des PE-Kommandos "End_Pause"

Der Aufruf des Kommandos "End_Pause" mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	2	Aufruf des PE-Kommandos "End_Pause".
CMD_MODIFIER	0	Bei Kommando "End_Pause" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Time_to_operate	-	DWORD	Erwartete Zeit, um in den Modus "PE_ready_to_operate" umzuschalten.

PE-Kommando "Query_modes" - "List_Energy_Saving_Modes"

Beschreibung

Mit dem PE-Kommando "Query_modes" und dem Unterkommando (Modifier) "List_Energy_Saving_Modes" geben Sie alle durch die PE-Entity unterstützten Energiesparmodi (PE_Mode_ID) aus.

Das Abfrageergebnis wird in Form des Antworttelegramms in den über den Parameter RESPONSE_DATA referenzierten Datenbaustein geschrieben.

Aufruf des PE-Kommandos "Query_modes" - "List_Energy_Saving_Modes"

Der Aufruf des Kommandos"List_Energy_Saving_Modes" mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	3	Aufruf des PE-Kommandos "Query_modes".
CMD_MODIFIER	1	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "List_Energy_Saving_Modes" zur Ausgabe der Anzahl und Typen der unterstützten Energiesparmodi.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Number_of_PE_ Mode_IDs	1	BYTE	Die Anzahl der Energiesparmodi.
PE_Mode_IDs	-	Array [...] of BYTE	Array mit den IDs der unterstützten Energiesparmodi. Die Bedeutung der einzelnen IDs ist abhängig von der PE-Entity.

PE-Kommando "Query_modes" - "Get_Mode"

Beschreibung

Mit dem PE-Kommando "Query_modes" und dem Unterkommando (Modifier) "Get_Mode" geben Sie die Attribute zu dem momentan aktivierten Energiesparmodus aus.

Aufruf des PE-Kommandos "Query_modes" - "Get_Mode"

Der Aufruf des Kommandos mit dem mit der Anweisung "PE_CMD" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	3	Aufruf des PE-Kommandos "Query_modes"
CMD_MODIFIER	2	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Mode" zur Ausgabe des Status des momentan aktivierten Modus.

Parameter	Wert	Beschreibung
CMD_PARA_LEN	1	Länge des Parameters CMD_PARA von 1 Byte.
CMD_PARA	ANY	ANY -Zeiger auf den Wert für PE_MODE_ID.

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID	• 0 Modus "PE_power_off" • 1...254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	ID des momentan aktiven Energiesparmodus.
PE_Mode_Attributes	Bit 0: • = 0: Nur statische Energieverbrauchs- und Zeitwerte verfügbar. • = 1: Dynamische Energieverbrauchs- und Zeitwerte verfügbar. Bit 1 bis 7: • Reserviert	BYTE	
Time_min_Pause [1]	Zeitdifferenz ohne Datum	DWORD	Minimale Pausenzeit für den PE-Modus. Die minimale Pausenzeit ist die Summe aus den Werten der folgenden Attribute: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay Siehe Beschreibung "Parameter PAUSE_TIME" der Anweisung "PE_START_END: Energiesparmodus starten und beenden (Seite 5543)".
Time_to_Pause [1]	Zeitdifferenz ohne Datum	DWORD	Ausschaltzeit: Zeitdauer vom Aufruf des Energiesparmodus bis zum Start des Energiesparmodus (Übergangsdauer von PE_ready_to_operate zu PE_energy_saving_mode). Die Ausschaltzeit ist abhängig von der PE-Entity.

Attribut	Wert	Datentyp	Beschreibung
Time_to_operate [1]	Zeitdifferenz ohne Datum	DWORD	Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate). Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet.
Time_min_length_of_stay [1]		DWORD	Mindestdauer, in der der Energiesparmodus in der PE-Entity aktiv ist.
Time_max_length_of_stay [1]		DWORD	Maximale Dauer, in der der Energiesparmodus in der PE-Entity aktiv ist.
Mode_Power_Consumption [2]		REAL	Stromverbrauch der PE-Entity im aktivierten Energiesparmodus. Einheit: kW
Energy_Consumption_to_pause [2]		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Bereitschaftsmodus (PE_ready_to_operate) zum Energiesparmodus (PE_energy_saving_mode) Einheit: kWh
Energy_Consumption_to_operate [2]		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate) Einheit: kWh
[1] Ist die Zeitdauer unendlich, wird 0xFFFFFFFF ausgegeben. Ist die Zeitdauer null, wird "0" ausgegeben. [2] Sind die Daten zu Energie- und Stromverbrauch durch die PE-Entity nicht definiert, wird als Wert "0,0" ausgegeben.			

PE-Kommando "PEM_Status"

Beschreibung

Mit dem PE-Kommando "PEM_Status" fragen Sie den Status eines momentan aktiven Energiesparmodus einer PE-Entity ab.

Aufruf des PE-Kommandos "PEM_Status"

Der Aufruf des Kommandos "PEM_Status" mit der Anweisung "PE_CMD" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	4	Aufruf des PE-Kommandos "PEM_Status".
CMD_MODIFIER	0	Bei Kommando "PEM_Status" keine weitere Spezifikation des Kommandoaufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
PE_Mode_ID_ Source	• 0 Modus "PE_power_off" • 1 bis 254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	Modus, in dem sich die PE-Entity vor dem Absetzen eines PE-Kommandos befindet.
PE_Mode_ID_ Destination	• 0 Modus "PE_power_off" • 1 bis 254 Energiesparmodus der PE-Entity (herstellerspezifisch) • 255 Modus "PE_ready_to_operate"	BYTE	Modus, in dem sich die PE-Entity nach der Durchführung eines PE-Kommandos befindet.
Time_to_operate	Zeitdifferenz ohne Datum.	DWORD	Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate). Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet.
Remaining_time_to_ destination	Zeitdifferenz ohne Datum.	DWORD	Verbleibende Zeit zur Umschaltung in den anderen Modus.
Mode_Power_ Consumption		REAL	Stromverbrauch der PE-Entity im aktivierten Energiesparmodus. Einheit: kW
Energy_ Consumption_ to_Destination		REAL	Energieverbrauch für momentane PE-Transition Einheit: kWh
Energy_ Consumption_ to_operate		REAL	Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy-saving mode) zum Bereitschaftsmodus (PE_ready_to_operate) Einheit: kWh

PE-Kommando "PE_Identify"

Beschreibung

Mit dem PE-Kommando "PE_Identify" lesen sie die Anzahl und die Beschreibung der durch die PE-Entity unterstützten PE-Kommandos aus. Welche und wie viele Kommandos unterstützt werden, ist abhängig von der PE-Entity. Da PE_Identify selbst ein PE-Kommando ist, werden bei einer positiven Antwort mindestens drei unterstützte PE-Kommandos ausgegeben: Start_Pause, End_Pause und PE_Identify.

Aufruf des PE-Kommandos "PE_Identify"

Der Aufruf des Kommandos "PE_Identify" mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	5	Aufruf des Kommandos "PE_Identify".
CMD_MODIFIER	0	Bei Kommando "PE_Identify" keine weitere Spezifikation des Kommando-aufrufs.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count [1]	6	BYTE	Anzahl der unterstützten PROFIenergy-Kommandos
Start_Pause	1	BYTE	Erstes unterstütztes PE-Kommando (Service_Request_ID)
End_Pause	2	BYTE	...
Query_Modes	3	BYTE	...
PEM_Status	4	BYTE	...
PE_Identify	5	BYTE	...
Query_Measurement	16	BYTE	Letztes unterstütztes PE-Kommando (Service_Request_ID)

[1] Die Anzahl der unterstützten Kommandos ist herstellerspezifisch und hängt von der verwendeten PE-Entity ab. Die angegebenen Werte sind ein Beispiel für ein Antworttelegramm, wenn alle 6 PE-Kommandos unterstützt werden.

PE-Kommando "Query_Measurement" - "Get_Measurement_list"

Beschreibung

Mit dem PE-Kommando "Query_Measurement" und dem Unterkommando (Modifier) "Get_measurement_list" führen Sie eine Abfrage durch, welche Messwerte durch die PE-Entity unterstützt werden. Die unterstützten Messwerte werden als Liste in dem über den Parameter RESPONSE_DATA referenzierten Datenbaustein hinterlegt.

Aufruf des PE-Kommandos "Query_Measurement" - "Get_Measurement_list"

Der Aufruf des Kommandos mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	16	Aufruf des Kommandos "Query_Measurement"
CMD_MODIFIER	1	Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Measurement_List" zur Ausgabe einer Liste der unterstützten Messwerte.
CMD_PARA_LEN	0	Länge des Parameters CMD_PARA von 0 Byte.
CMD_PARA	irrelevant	-

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count	-	BYTE	Anzahl der Measurement-IDs
reserved	-	BYTE	
...			
Measurement_ID	-	WORD	Erste unterstützte Measurement_ID. Die Measurement_ID ist herstellerspezifisch. Weitere Informationen entnehmen Sie dem Handbuch der jeweiligen PE-Entity.
Accuracy_Domain	-	BYTE	Siehe Tabelle "Genauigkeitsbereiche".
Accuracy_Class	-	BYTE	Siehe Tabellen "Genauigkeitsklassen".
Range	-	REAL	Spezifiziert den Skalen-Endwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet).
...			
Measurement_ID	-	WORD	Letzte unterstützte Measurement_ID
Accuracy_Domain	-	BYTE	Siehe Tabelle "Genauigkeitsbereiche".
Accuracy_Class	-	BYTE	Siehe Tabellen "Genauigkeitsklassen".
Range	-	REAL	Spezifiziert den Skalenendwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet).

Genauigkeitsbereiche

Genauigkeitsbereich	Beschreibung
0	Reserviert
1	Die Genauigkeitsabweichung wird in Prozent des Skalenendwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).
2	Die Genauigkeitsabweichung wird in Prozent des aktuellen Messwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).
3	Die Messgenauigkeit erfolgt nach der Norm IEC 61557-12. Die Funktions-Leistungsklassen für Geräte zur Leistungs-Messung und Überwachung (PMD) ohne externe Sensoren und System-Leistungsklassen für PMD mit externen Sensoren sind in der entsprechend der Tabelle "Genauigkeitsklassen des Genauigkeitsbereichs 3" codiert.
4	Die Angabe der Genauigkeit erfolgt entsprechend der Norm EN 50470-3, Kapitel 8 (siehe auch Tabelle: Genauigkeitsklassen des Genauigkeitsbereichs 4).

Genauigkeitsklassen

Tabelle 4-195 Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2

Genauigkeitsklasse	0	1	2	3	4	5	6	7	8
Bedeutung	reserviert	0,01%	0,02%	0,05%	0,1%	0,2%	0,5%	1%	1,5%

Genauigkeitsklasse	9	10	11	12	13	14	15	>15
Bedeutung	2%	2,5%	3%	5%	10%	20%	>20%	Nicht definiert

Tabelle 4-196 Genauigkeitsklassen des Genauigkeitsbereichs 3

Genauigkeitsklasse	0	1	2	3	4	5	6	7	8
Bedeutung	reserviert	0,02	0,05	0,1	0,2	0,5	1	1,5	2

Genauigkeitsklasse	9	10	11	12	13	14	>13
Bedeutung	2,5	3	5	10	20	20%	Nicht definiert

Tabelle 4-197 Genauigkeitsklassen des Genauigkeitsbereichs 4

Genauigkeitsklasse	0	1	2	3	4	5	6	>7
Bedeutung	reserviert	0,5	1,0	1,5	2,0	2,5	3,0	Nicht definiert

PE-Kommando "Query_Measurement" - "Get_Measurement_values"

Beschreibung

Mit dem PE-Kommando "Query_Measurement" und dem Unterkommando (Modifier) "Get_measurement_values" geben Sie die von der PE-Entity unterstützten Messwerte aus. Die Messwerte werden als Liste in dem über den Parameter RESPONSE_DATA referenzierten Datenbaustein hinterlegt.

Aufruf des PE-Kommandos "Query_Measurement" - "Get_Measurement_values"

Der Aufruf des Kommandos mit der Anweisung "PE_CMD (Seite 5547)" erfolgt mit den folgenden Parametern:

Parameter	Wert	Beschreibung
CMD	16	Aufruf des Kommandos "Query_Measurement"
CMD_MODIFIER	2	Spezifikation des Kommandoaufrufs: Auswahl des Kommandos "Get_Measurement_Values" zur Ausgabe einer Liste der unterstützten Messwerte.
CMD_PARA_LEN	0	Abhängig von der Anzahl der Messwerte. Die Länge des Parameters ergibt sich aus dem Attribut count und der Summe der Längen der Attribute für die übertragenen Messwerte.
CMD_PARA	ANY	ANY-Zeiger auf Datenstruktur mit Auflistung der abzufragenden Messwerte (siehe "Parameter CMD_PARA").

Parameter CMD_PARA

Die Struktur, die über den ANY-Zeiger am Parameter CMD_PARA angegeben wird, muss den folgenden Aufbau haben:

Attribut	Wert	Datentyp	Beschreibung
Count	-	BYTE	Anzahl der Messwerte (Measurement-IDs)
reserved	0	BYTE	Nicht verwendet
Measurement_ID	-	WORD	Erster abgefragter Messwert
...			
Measurement_ID	-	WORD	Letzter abgefragter Messwert

Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der der PE-Entity werden in den am Parameter RESPONSE_DATA (siehe "PE_CMD (Seite 5547)") referenzierten Datenbaustein geschrieben:

Attribut	Wert	Datentyp	Beschreibung
Count [1]	-	BYTE	Anzahl der Messwerte (Measurement-IDs)
reserved	0	BYTE	Nicht verwendet
Length_of_Structure	2 bis 65535	WORD	Länge der Struktur in Bytes.

Attribut	Wert	Datentyp	Beschreibung
Measurement_Data_Structure_ID	1 = simple value	BYTE	Definiert die folgende Struktur.
Measurement_ID	0 bis 65535	WORD	ID des unterstützten Messwertes.
Status_of_ Measurement_Value	1 bis 3	BYTE	Status des Messwerts: • 1: Gültig • 2: Nicht unterstützt • 3: Nicht gültig
Transmission_Data_Type	-	REAL	
End_of_demand	-	TOD	Optionale Zeitstempelung mit dem Datentyp TimeOfDay.
...			
Length_of_Structure	-	WORD	Länge der Struktur in Bytes.
Measurement_Data_Structure_ID	-	BYTE	Definiert die folgende Struktur.
Measurement_ID	-	WORD	ID des unterstützten Messwertes.
Status_of_Measurement_Value	-	BYTE	Status des Messwerts: • 1: Gültig • 2: Nicht unterstützt • 3: Nicht gültig
Transmission_Data_Type	-	REAL	
End_of_demand	-	TOD	Optionale Zeitstempelung mit dem Datentyp TimeOfDay.
[1] Wenn die Datenlänge der abgefragten Messwerte die Größe der PDU (Protocol Data Unit) der Protokollschicht übersteigt, werden die Daten unvollständig übertragen und nur die unterstützten Messwerte ausgegeben.			

iDevice / iSlave

PE_I_DEV: PROFIenergy-Kommandos im iDevice steuern

Beschreibung

Die Anweisung "PE_I_DEV" wird zur Abwicklung des PROFIenergy-Profils im intelligenten IO-Device (iDevice) verwendet. Was in einem normalen PROFIenergy-fähigen IO-Device, z. B. der ET 200S, von der Firmware erledigt wird, übernehmen im iDevice die Anweisung "PE_I_DEV" und die zugehörigen Hilfsbausteine:

- Die Anweisung "PE_I_DEV" wird zyklisch vom Anwenderprogramm des iDevice aufgerufen und empfängt alle PROFIenergy-Kommandos.
- Die Generierung der PROFIenergy-Antwort erfolgt durch Parametrierung eines Hilfsbausteins. Die Reaktion in der Pause ist frei programmierbar. Die Antwortdaten müssen innerhalb von 10 Sekunden bereitgestellt werden, da sonst am Parameter STATUS der Anweisung im IO-Controller der "Stateconflict 0x80B5" auftritt.

Zur Verwendung der Anweisung sind keine expliziten Kenntnisse des PROFIenergy-Standards erforderlich.

PROFIenergy-Hilfsbausteine (PE-Hilfsbausteine)

Über die PE-Hilfsbausteine generieren Sie das Antworttelegramm. Hierzu geben Sie die Antwortdaten (im Klartext) an den Eingangsparametern des jeweiligen Bausteins an.

- Für jedes PROFIenergy-Kommando gibt es einen entsprechenden Hilfsbaustein für eine positive Antwort:
  - PE-Kommando "Start_Pause": PE_Start_RSP (Seite 5581)
  - PE-Kommando "End_Pause": PE_End_RSP (Seite 5582)
  - PE-Kommando "Query_modes" - "List_Energy_Saving_Modes": PE_List_Modes_RSP (Seite 5584)
  - PE-Kommando "Query_modes" - "Get_Mode": PE_Get_Mode_RSP (Seite 5585)
  - PE-Kommando "PEM_Status": PE_PEM_Status_RSP (Seite 5587)
  - PE-Kommando "PE_Identify": PE_Identify_RSP (Seite 5589)
  - PE-Kommando "Query_Measurement" - "Get_Measurement_list": PE_Measurement_List_RSP (Seite 5590)
  - PE-Kommando "Query_Measurement" - "Get_Measurement_values": PE_Measurement_Value_RSP (Seite 5592)
- Unabhängig vom PROFIenergy-Kommando gibt es zusätzlich einen gemeinsamen Hilfsbaustein für eine negative Antwort (siehe PE_Error_RSP (Seite 5580)).

Verschaltung der Hilfsbausteine

Die Anweisung "PE_I_DEV" und die Hilfsbausteine sind aufeinander abgestimmt. Die Parameter werden teilweise einfach verschaltet. Die folgende Grafik zeigt, welche der Parameter verschaltet werden müssen.

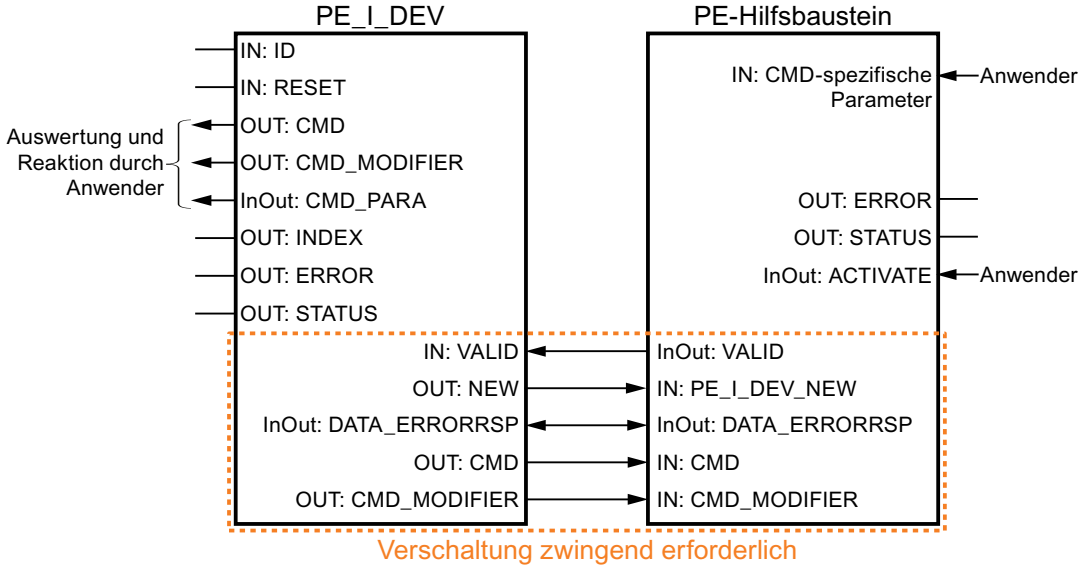

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE_I_DEV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RESET	Input	BOOL	E, A, M, D, L	Rücksetzen der Anweisung.
ID	Input	DWORD	E, A, M, D, L oder Konstante	Adresse des iDevice Die Adresse kann aus der Hardware-Konfiguration übernommen werden.
VALID	Input	BOOL	E, A, M, D, L	Die Antwortdaten für den PROFIenergy-Controller stehen bereit und können gesendet werden.
CMD_PARA	Output	ANY	E, A, M, D, L	Parameter für: • Get mode: PE_mode_ID • Get measurement values: Liste der Measurement_IDs (Liste der IDs der zu lesenden Variablen; es können entweder eine einzelne Variable oder auch mehrere Variablen auf einmal gelesen werden.) Maximale Länge: 234 Byte
DATA_ERRORRSP	InOut	ANY	E, A, M, D, L	Zeiger auf den Datenbereich, der die Quittungsdaten für den PROFIenergy-Controller enthält. Dieser muss mit dem Zeiger übereinstimmen, der auch bei den Hilfsbausteinen verwendet wird.
INDEX	Output	INT	E, A, M, D, L	Datensatznummer des PROFIenergy Records (0x80A0 fest)
CMD	Output	INT	E, A, M, D, L	Service-Request-ID des PROFIenergy-Kommandos entsprechend dem PROFIenergy-Profil (siehe "Parameter CMD und CMD_MODIFIER"). Nach Erweiterungen des PROFIenergy-Profils sind weitere PE- Kommandos (Service-Request-IDs) möglich.
CMD_MODIFIER	Output	INT	E, A, M, D, L	PROFIenergy-Unterkommando: • Nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER". • Bei allen anderen Kommandos: "0". Nach Erweiterungen des PROFIenergy-Profils sind weitere Unterkommandos möglich.
NEW	Output	BOOL	E, A, M, D, L	Neue Daten vom PROFIenergy-Controller verfügbar.
ERROR	Output	BOOL	E, A, M, D, L	Befehl mit Fehler beendet.
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation (siehe "Parameter STATUS").

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter CMD und CMD_MODIFIER

CMD	CMD_ MODIFIER	PROFIenergy-Kommando	Beschreibung
01	0	Start_Pause	Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus.
02	0	End_Pause	Beenden des Energiesparmodus.
03	1	Query_Modes - List energy saving Modes	Ausgeben der unterstützten Energiesparmodi.
	2	Query_Modes - Get Mode	Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.
04	0	PEM_Status	Status des Energiesparmodus abfragen.
05	0	PE_Identify	Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.
16	1	Query_Measurement - Get_Measurement_List	Auflistung der durch die der PE-Entity unterstützten Messwerte.
	2	Query_Measurement - Get_Measurement_Values	Ausgabe der Messwerte der der PE-Entity.

Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

Feldelement	Name	Bedeutung
STATUS[1]	Function_Num	Fehlerursache • B#16#**00**: Kein Fehler • B#16#**DE**: Fehler beim Lesen des Datensatzes • B#16#**DF**: Fehler beim Schreiben des Datensatzes • B#16#**C0**: Fehlermeldung durch die Anweisung "PE_I_DEV" oder der intern verwendeten Kommunikationsanweisungen "PRVREC" und "RCVREC".
STATUS[2]	Error_Decode	Ort der Fehlerkennung • **80**: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch
STATUS[3]	Error_Code_1	Fehlerkennung (bei Error_Decode = **80**): • **B1**: Write length error (Fehler bei der Schreiblänge oder unzureichende Längenangabe über den Datentyp ANY).
STATUS[4]	Error_Code_2	Bei PROFINET-Fehler: Ausgabe der Fehlermeldung des IO-Controllers. Ist kein PROFINET-Fehler aufgetreten ist der Wert in STATUS[4] = "0".

Hinweis

Fehlermeldungen der Anweisungen "PRVREC" und "RCVREC"

Die Anweisung "PE_I_DEV" verwendet zur Kommunikation die Anweisungen "PRVREC (Seite 5504)" und "RCVREC (Seite 5502)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "PRVREC" und "RCVREC" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 5468).

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Hilfsbausteine der Anweisung PE_I_DEV

PE_Error_RSP: Negative Antwort auf Kommando generieren

Beschreibung

Der Hilfsbaustein "PE_Error_RSP" (Response with failure) generiert eine negative Antwort, falls das angeforderte Kommando generell oder temporär nicht unterstützt wird. Die Generierung der Antwort erfolgt unabhängig vom angeforderten Kommando.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Error_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
ERROR_CODE	Input	BYTE	E, A, M, D, L oder Konstante	Fehlernummer
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Start_RSP: Antwort auf Kommando zum Pausenstart generieren

Beschreibung

Der Hilfsbaustein "PE_Start_RSP" (Start Pause) generiert die Antwort auf das PE-Kommando Start_Pause (Seite 5566). Die Anweisung gibt zurück, welchen Energiesparzustand das Gerät einnimmt (Parameter PE_MODE_ID).

Wenn Sie bei verschieden langen Pausen unterschiedlich reagieren, können Sie das in der Rückmeldung über den eingenommenen Energiesparzustand zurückmelden (PE_Mode_ID = 1 für eine kurze Pause, PE_Mode_ID = 2 für eine längere Pause usw.).

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Start_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
PE_MODE_ID	Input	BYTE	E, A, M, D, L oder Konstante	PE-Modus, den der Prozess einnimmt

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_End_RSP: Antwort auf Kommando zum Pausenende generieren

Beschreibung

Der Hilfsbaustein "PE_End_RSP" generiert die Antwort auf das PE-Kommando End_Pause (Seite 5567). Als Antwort wird die benötigte Zeitdauer zurückgegeben, um von dem aktuellen Betriebszustand in den Betriebszustand "Ready_To_Operate" zu wechseln.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_End_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
Time_to_Operate	Input	DWORD	E, A, M, D, L oder Konstante	Zeitdauer, um vom aktuellen Betriebszustand nach "Ready_To_Operate" zu wechseln.
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_List_Modes_RSP: Abgefragte Engiesparmodi als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_List_Modes_RSP" generiert die Antwort auf das PE-Kommando List_Energy_Saving_Modes (Seite 5567). Die generierte Antwort enthält die Anzahl und die IDs der unterstützten Energiesparzustände.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_List_Modes_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Number_of_PE_Mode_IDs	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der unterstützen Energiesparzustände. Zulässige Werte: 1 bis 254
PE_MODE_ID	Input	ANY	E, A, M, D, L	Zeigt auf den Bereich, in dem die IDs der unterstützen Energiesparzustände (PE_Mode_ID) hinterlegt sind. Zulässiger Bereich: 1 bis 254.
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Get_Mode_RSP: Abgefragte Energiedaten als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Get_Mode_RSP" generiert die Antwort auf das Kommando Get_Mode (Seite 5568). Die Antwort enthält die Zeiten und Leistungs- bzw. Energiedaten eines einzelnen Energiesparzustands.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Get_Mode_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
PE_Mode_ID	Input	BYTE	E, A, M, D, L oder Konstante	ID des momentan aktiven Energiesparmodus.
PE_Mode_Attributes	Input	BYTE	E, A, M, D, L oder Konstante	Zusatzinformation zum Energiesparmodus

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Time_min_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Mindest-Pausendauer für diesen PE-Energiesparmodus. Diese bildet die Summe der drei Parameter: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay
Time_to_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Zeitdauer von der Flanke am Parameter START (siehe "PE_I_DEV (Seite 5576)") bis zum Erreichen des angeforderten PE-Energiesparmodus.
Time_to_Operate	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Einschaltdauer bis "PE_ready_to_operate". Der Parameter "Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximalwert sein oder durch die PE-Entity dynamisch berechnet werden.
Time_min_Lenght_of_stay	Input	DWORD	E, A, M, D, L oder Konstante	Minimale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben muss.
Time_max_Lenght_of_stay	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben kann.
Mode_Power_Consumption	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch im aktuellen PE-Modus in [kW].
Energy_Consum_to_Pause	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch von "PE_ready_to_operate" bis zum aktuellen PE –Modus in [kWh].
Energy_Consum_to_operate	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_PEM_Status_RSP: PEM-Status als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_PEM_Status_RSP" generiert die Antwort auf das Kommando PEM_Status (Seite 5570).

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_PEM_Status_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
PE_MODE_ID_ Source	Input	BYTE	E, A, M, D, L oder Konstante	Source und Destination für PEM_STATUS. Werte: • 0x00: PE_POWER_OFF • 0x01 – 0xFE: Herstellerspezifisch • 0xFF: PE_READY_TO_OPERATE
PE_MODE_ID_ Destination	Input	BYTE	E, A, M, D, L oder Konstante	
Time_to_Operate[1]	Input	DWORD	E, A, M, D, L oder Konstante	Maximale Einschaltdauer bis "PE_ready_to_operate". "Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximal-Wert sein oder durch die PE-Entity dynamisch berechnet werden.
Remaining_time_ to_destination[1]	Input	DWORD	E, A, M, D, L oder Konstante	Optional: Verbleibende Zeit bis zum angeforderten PE-Modus. Dynamischer Wert oder statischer Maximal-Wert

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
Mode_Power_ Consumption[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch im aktuellen PE-Modus in [kW].
Energy_ Consumption_ to_Destination[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch bis zum angeforderten PE-Modus in [kWh].
Energy_ Consumption_ to_operate[2]	Input	DWORD	E, A, M, D, L oder Konstante	Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

[1] Wenn die Zeitdauer unbegrenzt ist, kann der max. Wert "0xFFFFFFFF" angegeben werden. Wenn die Zeitdauer "Null" ist, kann "0x00" verwendet werden.

[2] Wenn ein Energieverbrauchswert nicht definiert ist, kann "0,0"angegeben werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Identify_RSP: Unterstützte PROFIenergy-Kommandos als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Identify_RSP" generiert die Antwort auf das Kommando PE_Identify (Seite 5572). In der Antwort auf das Kommando geben Sie an, welche PROFIenergy-Kommandos unterstützt werden. Beachten Sie, dass PE_IDENTIFY selbst ein PE-Kommando ist und in der Antwort entsprechend mit angegeben wird.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Identify_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
Start_Pause	Input	BOOL	E, A, M, D, L oder Konstante	Jeweils ein Parameter für jedes der entsprechenden PROFIenergy-Kommandos: • 0: PE-Kommando wird nicht unterstützt • 1: PE-Kommando wird unterstützt
End_Pause	Input	BOOL	E, A, M, D, L oder Konstante	
Query_Modes	Input	BOOL	E, A, M, D, L oder Konstante	
PEM_Status	Input	BOOL	E, A, M, D, L oder Konstante	
PE_Identify	Input	BOOL	E, A, M, D, L oder Konstante	
Query_Measurement	Input	BOOL	E, A, M, D, L oder Konstante	
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Measurement_List_RSP: Liste der unterstützten Messwerte als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Measurement_List_RSP" generiert die Antwort auf das Kommando Get_measurement_list (Seite 5573). In der Antwort geben Sie an, welche Messwerte (Measurement-IDs) unterstützt werden.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Measurement_List_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD_MODI-FIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MO-DIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Count	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der unterstützten Messwerte (Measurement-IDs)
Measurement_ List	Input	ANY	D	Zeiger auf das Array mit den unterstützten Measurement_IDs. Zum Aufbau des Arrays gemäß PROFIenergy Profil siehe: PE-Kommando "Query_Measurement" - "Get_Measurement_list" (Seite 5573)
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

PE_Measurement_Value_RSP: Angeforderte Messwerte als Antwort generieren

Beschreibung

Der Hilfsbaustein "PE_Measurement_Value_RSP" generiert die Antwort auf das Kommando Get_measurement_values (Seite 5575). In der Antwort geben Sie die Werte der angeforderten Messwerte zurück.

Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE_Measurement_Value_RSP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PE_I_DEV_NEW	Input	BOOL	E, A, M, D, L oder Konstante	Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.
CMD	Input	INT	E, A, M, D, L oder Konstante	Service-Request-ID des PROFIenergy-Kommandos Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden.
CMD_MODIFIER	Input	INT	E, A, M, D, L oder Konstante	PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.
Count	Input	BYTE	E, A, M, D, L oder Konstante	Anzahl der Messwerte (Measurement_Values).
Measurement_Values	Input	ANY	D	Zeiger auf das Array mit den Messwerten (Measurement_IDs). Zum Aufbau des Arrays gemäß PROFIenergy Profil, siehe PE-Kommando "Query_Measurement" - "Get_Measurement_values" (Seite 5575)
ACTIVATE	InOut	BOOL	E, A, M, D, L	Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 5576)" eine positive Flanke erkannt wurde.
VALID	InOut	BOOL	E, A, M, D, L	Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 5576)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA_ ERRORRSP	InOut	ANY	D	Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 5576)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte
ERROR	Output	BOOL	E, A, M, D, L	• "0": Kein Fehler • "1": Fehler aufgetreten.
STATUS	Output	WORD	E, A, M, D, L	• "0": Kein Fehler • "0x80B1": Fehler bei der ANY-Angabe, z. B. falscher Bereich

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

4.2.3.6 Baugruppenparametrierung

Datensätze schreiben und lesen

Prinzip

Es gibt Baugruppen, die über einen Systemdatenbereich verfügen, auf den Sie von Ihrem Programm aus nur schreibend zugreifen können. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt (siehe folgende Tabelle).

Darüber hinaus können Baugruppen auch einen Systemdatenbereich besitzen, auf den Sie von Ihrem Programm aus nur lesend zugreifen können. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt.

Hinweis

Es gibt Baugruppen, die über beide Systemdatenbereiche verfügen. Dabei handelt es sich um physikalisch unterschiedliche Bereiche, die lediglich die logische Aufteilung in Datensätze gemeinsam haben.

Nur beschreibbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur beschreibbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie geschrieben werden können.

Datensatz-Nummer	Inhalt	Größe	Einschränkung	beschreibbar mit Anweisung
0	Parameter	bei S7-300: 2 bis 14 Bytes	nur bei S7-400 beschreibbar	WR_DPARM (Seite 5603) PARM_MOD (Seite 5598)
1	Parameter	bei S7-300: 2 bis 14 Bytes (DS0 und DS1 haben zusammen genau 16 Bytes)	-	WR_PARM (Seite 5602) WR_DPARM (Seite 5603) PARM_MOD (Seite 5598)
2 bis 127	Anwenderdaten	je ≤ 240 Bytes	-	WR_PARM (Seite 5602) WR_DPARM (Seite 5603) PARM_MOD (Seite 5598) WR_REC (Seite 5494)
128 bis 240	Parameter	je ≤ 240 Bytes	-	WR_PARM (Seite 5602) WR_DPARM (Seite 5603) PARM_MOD (Seite 5598) WR_REC (Seite 5494)

Nur lesbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur lesbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie gelesen werden können.

Datensatz-Nummer	Inhalt	Größe	lesbar mit Anweisung
0	baugruppenspezifische Diagnosedaten (systemeinheitlich festgelegt)	4 Bytes	RDSYSST (Seite 5683) (SZL_ID00B1H) RD_REC (Seite 5490)
1	kanalspezifische Diagnosedaten (incl. Datensatz 0)	• bei S7-300: 16 Bytes • bei S7-400: 4 bis 220 Bytes	RDSYSST (Seite 5683) (SZL_ID00B2H und 00B3H) RD_REC (Seite 5490)

Datensatz-Nummer	Inhalt	Größe	lesbar mit Anweisung
2 bis 127	Anwenderdaten	je ≤ 240 Bytes	RD_REC (Seite 5490)
128 bis 240	Diagnosedaten	je ≤ 240 Bytes	RD_REC (Seite 5490)

Systemressourcen

Wenn Sie mehrere asynchron laufende Datensatzübertragungen kurz nacheinander anstoßen, so wird gewährleistet, dass alle Aufträge durchgeführt werden und keine gegenseitige Beeinflussung stattfindet.

Wird die Begrenzung der Systemressourcen erreicht, so erhalten Sie dies in RET_VAL mitgeteilt. Der temporäre Fehlerfall kann durch eine Wiederholung des Auftrags behoben werden.

Die maximale Anzahl "gleichzeitig" aktiver Aufträge eines Anweisungstyps ist CPU-abhängig.

RD_DPAR: Datensatz einer Baugruppe lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer INDEX der adressierten Komponente aus den projektierten Systemdaten. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.

Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

Arbeitsweise

Die Anweisung "RD_DPAR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "RD_DPAR" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die Bytes 2 und 3 des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die Bytes 2 und 3 von STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPAR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Lesen
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Irgendeine logische Adresse der Baugruppe In Bit 15 geben Sie an, ob es sich um eine Ein- (Bit 15 = 0) oder eine Ausgangsadresse (Bit 15= 1) handelt.
INDEX	Input	INT	E, A, M, D, L oder Konstante	Datensatznummer
VALID	Output	BOOL	E, A, M, D, L	Neuer Datensatz wurde empfangen und ist gültig
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Auftrag ist noch nicht beendet.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Beim Lesevorgang trat ein Fehler auf.
STATUS	Output	DWORD	E, A, M, D, L	• Aufrufkennung (Bytes 2 und 3) bzw. Fehlercode • Byte 1: B#16#00, falls kein Fehler. Andernfalls Funktionskennung aus DPV1-PDU: Im Fehlerfall bei Datensatz lesen B#16#DE, im Fehlerfall bei Datensatz schreiben B#16#DF. Falls kein DPV1-Protokollelement benutzt wird: B#16#C0. • Byte 4: herstellerspezifische Erweiterung der Fehlerkennung
LEN	Output	INT	E, A, M, D, L	Länge der gelesenen Datensatzinformation
RECORD	InOut	ANY	E, A, M, D, L	Zielbereich für den gelesenen Datensatz Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Siehe auch: PARM_MOD: Datensätze einer Baugruppe übertragen (Seite 5598).

RD_DPARA: Datensatz einer Baugruppe asynchron lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM einer ausgewählten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD aufgespannten Zielbereich eingetragen.

Arbeitsweise

Die Anweisung "RD_DPARA" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie die Anweisung mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPARA":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Lesen
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Irgendeine Adresse der Baugruppe. Bei einer Ausgabeadresse muss das höchstwertige Bit gesetzt sein.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 0 bis 240)
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Falls bei der Übertragung kein Fehler auftrat, sind die folgenden Fälle zu unterscheiden: • RET_VAL enthält die Länge des tatsächlich gelesenen Datensatzes in Bytes, falls der Zielbereich größer ist als der gelesene Datensatz. • RET_VAL enthält "0", falls die Länge des gelesenen Datensatzes gleich der Länge des Zielbereichs ist.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Auftrag ist noch nicht beendet.
RECORD	Output	ANY	E, A, M, D, L	Zielbereich für den gelesenen Datensatz. Es ist nur der Datentyp BYTE zulässig. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Siehe auch: PARM_MOD: Datensätze einer Baugruppe übertragen (Seite 5598).

PARM_MOD: Datensätze einer Baugruppe übertragen

Beschreibung

Mit der Anweisung "PARM_MOD" übertragen Sie alle Datensätze einer Baugruppe, die Sie projektiert haben, zur Baugruppe. Dabei ist es ohne Bedeutung, ob die Datensätze statisch oder dynamisch sind.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PARM_MOD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ= 1: Anforderung zu Schreiben
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Basisadresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Fehlerinformationen

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) sind zwei Fälle zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A4, 80Cx):
  Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt, d. h. es ist sinnvoll, dass Sie die Anweisung erneut aufrufen (ggf. mehrfach).
  Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#809x, 80A1, 80Bx, 80Dx):
  Bei dieser Fehlerart kann sich der Fehler nicht ohne Ihr Zutun beheben. Ein erneuter Aufruf der Anweisung ist erst wieder sinnvoll, wenn Sie den Fehler beseitigt haben.
  Beispiel für einen permanenten Fehler: Falsche Länge des zu übertragenden Datensatzes (W#16#80B1).

Hinweis

Wenn Sie mit einer der Anweisungen WR_PARM, WR_DPARM oder PARM_MOD Datensätze zu einem DPV1-Slave übertragen und dieser im DPV1-Mode arbeitet, bewertet der DP-Master die vom Slave erhaltene Fehlerinformation wie folgt:

Liegt die Fehlerinformation in den Bereichen W#16#8000 bis W#16#80FF oder W#16#F000 bis W#16#FFFF, reicht der DP-Master die Fehlerinformation an die ANweisung weiter. Liegt sie außerhalb, gibt er den Wert W#16#80A2 an die Anweisung und suspendiert den Slave.

Anzeigen für die Anweisungen "RD_DPARM (Seite 5601)", "WR_PARM (Seite 5602)", "WR_DPARM (Seite 5603)" und "PARM_MOD"

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung	Einschränkung
0000	kein Fehler	-
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.	-
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".	Dezentrale Peripherie

Fehlercode (W#16#...)	Erläuterung	Einschränkung
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".	Dezentrale Peripherie
8090	Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.	-
8092	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.	Nur bei S7-400 für RD_DPARM (Seite 5601) und WR_PARM (Seite 5602)
8093	Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig. (Zulässig sind S7-300-Baugruppen bei S7-300, S7-400-Baugruppen bei S7-400, S7-DP-Baugruppen bei S7-300 und S7-400.)	-
80A1	Negative Quittung beim Senden des Datensatzes zur Baugruppe (Baugruppe während des Sendens gezogen oder Baugruppe defekt)	1)
80A2	DP-Protokollfehler bei Layer 2, evtl. Hardware-/Schnittstellenfehler im DP-Slave	Dezentrale Peripherie 1)
80A3	DP-Protokollfehler bei User Interface/User	Dezentrale Peripherie 1)
80A4	Kommunikation am K-Bus gestört	Fehler tritt auf zwischen CPU und externer DP-Anschaltung 1)
80B0	Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.	1)
80B1	Die Länge des zu übertragenden Datensatzes ist falsch. Bei RD_DPARM (Seite 5601): Die Länge des durch RECORD aufgespannten Zielbereichs ist zu klein.	-
80B2	Der projektierte Steckplatz ist nicht belegt.	1)
80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp im SDB1	1)
80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.	1)
80C2	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.	1)
80C3	Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.	
80C4	Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte ihren Aufbau auf elektrische Störquellen.	1)
80C5	Dezentrale Peripherie nicht verfügbar oder deaktiviert.	Dezentrale Peripherie 1)
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)	Dezentrale Peripherie 1)
80D0	Im zugehörigen SDB ist kein Eintrag für die Baugruppe vorhanden.	-
80D1	Die Datensatznummer ist im zugehörigen SDB für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.).	-
80D2	Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.	-
80D3	Auf den SDB kann nicht zugegriffen werden, da er nicht vorhanden ist.	-
80D4	SDB-Strukturfehler: SDB-interner Zeiger zeigt außerhalb SDB	nur bei S7-300
80D5	Der Datensatz ist statisch.	Nur bei "WR_PARM (Seite 5602)"

Fehlercode (W#16#...)	Erläuterung	Einschränkung
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)	-
[1] kommt bei "RD_DPARM" nicht vor		

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

RD_DPARM: Datensatz aus projektierten Systemdaten lesen

Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM der adressierten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD aufgespannten Zielbereich eingetragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_DPARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Basisadresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer (zulässige Werte: 0 bis 240)
RET_VAL	Return	INT	E, A, M, D, L	Länge des gelesenen Datensatzes in Bytes, falls der gelesene Datensatz in den Zielbereich passt und bei der Übertragung kein Fehler auftrat. Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
RECORD	Output	ANY	E, A, M, D, L	Zielbereich für den gelesenen Datensatz. Es ist nur der Datentyp BYTE zulässig.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Siehe auch: PARM_MOD: Datensätze einer Baugruppe übertragen (Seite 5598).

WR_PARM: Datensatz einer Baugruppe schreiben

Beschreibung

Mit der Anweisung übertragen Sie den Datensatz RECORD zur adressierten Baugruppe. Die Parameter, die zur Baugruppe übertragen werden, überschreiben nicht die per Projektierung erstellten Parameter.

Voraussetzungen

Der zu übertragende Datensatz darf nicht statisch sein. Die Information, welche Datensätze einer Baugruppe statisch sind, entnehmen Sie der Baugruppenbeschreibung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_PARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zu Schreiben
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Basisadresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer
RECORD	Input	ANY	E, A, M, D, L	Datensatz Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RECORD

Die zu übertragenden Daten werden aus dem Parameter RECORD beim ersten Aufruf gelesen. Falls die Übertragung des Datensatzes länger als einen Aufruf dauert, so ist bei den Folgeaufrufen der Anweisung (zum gleichen Auftrag) der Inhalt des Parameters RECORD nicht mehr relevant.

Parameter RET_VAL

Siehe auch: PARM_MOD: Datensätze einer Baugruppe übertragen (Seite 5598).

Hinweis

Nur für S7-400

Falls der allgemeine Fehler W#16#8544 auftritt, zeigt dies nur an, dass der Zugriff auf mindestens ein Byte des den Datensatz enthaltenden E/A-Speicherbereichs gesperrt war. Die Datenübertragung wurde fortgesetzt.

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

WR_DPARM: Datensatz übertragen

Beschreibung

Mit der Anweisung übertragen Sie den Datensatz mit der Nummer RECNUM aus den Projektierungsdaten zur adressierten Baugruppe. Dabei ist es ohne Bedeutung, ob der Datensatz statisch oder dynamisch ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_DPARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zu Schreiben
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Basisadresse der Baugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RECNUM	Input	BYTE	E, A, M, D, L oder Konstante	Datensatznummer
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Siehe auch: PARM_MOD: Datensätze einer Baugruppe übertragen (Seite 5598).

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

4.2.3.7 Alarme

Uhrzeitalarm

Verwendung der Anweisungen SET_TINT, CAN_TINT, ACT_TINT, QRY_TINT

Definition

Ein Uhrzeitalarm ist die Ursache für den uhrzeitgesteuerten Aufruf eines Uhrzeitalarm-OBs.

Voraussetzungen für Aufruf

Damit ein Uhrzeitalarm-OB vom Betriebssystem aufgerufen werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Uhrzeitalarm-OB muss parametriert sein (Startdatum und -uhrzeit, Ausführung)
  - Per Projektierung oder
  - mit der Anweisung "SET_TINT (Seite 5607)" im Anwenderprogramm.
- Der Uhrzeitalarm-OB muss aktiviert sein
  - Per Projektierung oder
  - mit der Anweisung "ACT_TINT (Seite 5609)" im Anwenderprogramm.
- Der Uhrzeitalarm-OB darf nicht per Projektierung abgewählt worden sein.
- Der Uhrzeitalarm-OB muss in der CPU vorhanden sein.
- Falls das Stellen über die Anweisung "ACT_TINT (Seite 5609)" erfolgt, dürfen Startdatum und -uhrzeit bei **einmaliger** Ausführung nicht abgelaufen sein; bei **periodischer** Ausführung wird der Uhrzeitalarm-OB zur nächsten abgelaufenen Periode aufgerufen (Startzeitpunkt + Vielfaches der Periodendauer).

Hinweis

Sie können den Uhrzeitalarm mit dem Projektierungswerkzeug parametrieren und im Anwenderprogramm aktivieren ("ACT_TINT (Seite 5609)").

Zweck der Anweisungen "SET_TINT", "CAN_TINT", "ACT_TINT" und "QRY_TINT"

Die Anweisungen verwenden Sie, um Uhrzeitalarme

- zu stellen ("SET_TINT (Seite 5607)")
- zu stornieren ("CAN_TINT (Seite 5608)")
- zu aktivieren ("ACT_TINT (Seite 5609)")
- abzufragen ("QRY_TINT (Seite 5610)").

Auswirkungen auf den Uhrzeitalarm

Die folgende Tabelle zeigt, welche Situationen welche Auswirkungen auf den Uhrzeitalarm haben.

Wenn ...	dann ...
ein Uhrzeitalarm neu gestellt wird (Aufruf von "SET_TINT (Seite 5607)"),	wird automatisch der laufende Uhrzeitalarm storniert.
der Uhrzeitalarm storniert wird (Aufruf von "CAN_TINT (Seite 5608)"),	werden Startdatum und -uhrzeit gelöscht. Der Uhrzeitalarm muss daher zunächst neu gestellt werden, bevor er aktiviert werden kann.

Wenn ...	dann ...
der Uhrzeitalarm-OB nicht vorhanden ist zum Zeitpunkt des Aufrufs,	wird automatisch ein Prioritätsklassenfehler erzeugt; d. h. das Betriebssystem ruft den OB 85 auf. Wenn der OB 85 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP.
die Uhr synchronisiert bzw. die Uhrzeit vorgestellt wird	Falls Startdatum/-uhrzeit beim Vorstellen übersprungen wurden: • Das Betriebssystem ruft den OB 80 auf.[1] • Im Anschluss an den OB 80 wird jeder übersprungene Uhrzeitalarm-OB aufgerufen (einmal, unabhängig von der Anzahl der übersprungenen Perioden), falls er nicht im OB 80 manipuliert wurde.[2] Wenn der OB 80 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP.
die Uhr synchronisiert bzw. die Uhrzeit zurückgestellt wird	S7-400-CPUs und CPU 318: Wenn Uhrzeitalarm-OBs im zurückgestellten Zeitbereich aufgerufen worden sind, dann werden sie beim erneuten Durchlaufen dieser Zeit nicht noch einmal aufgerufen. S7-300-CPUs: Die betroffenen Uhrzeitalarm-OBs werden alle ausgeführt.

[1] In der Startereignisinformation des OB 80 ist verschlüsselt, welche Uhrzeitalarm-OBs durch das Vorstellen nicht aufgerufen werden konnten. Die Uhrzeit in der Startereignisinformation entspricht der vorgestellten Zeit.

[2] Die Uhrzeit in der Startereignisinformation des nachgeholten Uhrzeitalarm-OBs entspricht der Startzeit des ersten übersprungenen Uhrzeitalarms.

Verhalten bei Neustart (Warmstart) oder Kaltstart

Bei Neustart (Warmstart) und Kaltstart werden alle im Anwenderprogramm per Anweisungen eingestellten Uhrzeitalarm-Einstellungen gelöscht. Wirksam sind dann die per Projektierung eingestellten Parameter.

Ausführung des Uhrzeitalarm-OBs

Die folgende Tabelle zeigt die unterschiedlichen Auswirkungen des Parameters "Ausführung" auf. Dieser Parameter ist einzustellen per Projektierung bzw. mit der Anweisung "SET_TINT (Seite 5607)" (Eingangsparameter PERIOD).

Ausführung des Uhrzeitalarm-OBs	Reaktion
keine (nur per Projektierung einstellbar)	Der Uhrzeitalarm-OB wird nicht ausgeführt, auch wenn er in der CPU vorhanden ist. Umparametrieren, d. h. Uhrzeitalarm stellen ("SET_TINT (Seite 5607)") ist im Anwenderprogramm möglich.
einmalig	Der Uhrzeitalarm ist nach Aufruf des Uhrzeitalarm-OBs storniert und kann neu gestellt und aktiviert werden.
periodisch (minütlich, stündlich, täglich, wöchentlich, monatlich, jährlich)	Wenn Startdatum und -uhrzeit bereits abgelaufen sind zum Zeitpunkt des Aktivierens, dann unterbricht der Uhrzeitalarm-OB das zyklische Programm zum Zeitpunkt "Startdatum/-uhrzeit + Vielfaches der eingestellten Periodendauer". In sehr seltenen Fällen kann der Uhrzeitalarm-OB beim nächsten Aufruf noch in Bearbeitung sein. Folge: • Zeitfehler (Betriebssystem ruft OB 80 auf; bei nicht vorhandenem OB 80 geht die CPU in den Betriebszustand STOP) • Uhrzeitalarm-OB wird nachbearbeitet.

SET_TINT: Uhrzeitalarm einstellen

Beschreibung

Mit der Anweisung können Sie Startdatum und -uhrzeit der Uhrzeitalarm-Organisationsbausteine einstellen. Bei der Startuhrzeit werden Ihre Sekunden- und Millisekundenangabe ignoriert und auf "0" gesetzt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, der zum Zeitpunkt SDT + Vielfaches von PERIOD gestartet wird (OB 10 bis OB 17).
SDT	Input	DT	D, L	Startdatum und Startuhrzeit Bei der Startuhrzeit werden Ihre Sekunden- und Millisekundenangabe ignoriert und auf 0 gesetzt. Falls Sie einen monatlichen Start eines Uhrzeitalarm-OB vorgeben wollen, sind beim Startdatum nur die Tage 1, 2, ... 28 möglich.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PERIOD	Input	WORD	E, A, M, D, L oder Konstante	Periode vom Ausgangspunkt SDT an: • W#16#0000 = Einmal • W#16#0201 = Minütlich • W#16#0401 = Stündlich • W#16#1001 = Täglich • W#16#1201 = Wöchentlich • W#16#1401 = Monatlich • W#16#1801 = Jährlich • W#16#2001 = Monatsende
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
8091	Fehlerhafter Parameter SDT
8092	Fehlerhafter Parameter PERIOD
80A1	Der eingestellte Startzeitpunkt liegt in der Vergangenheit. (Dieser Fehlercode tritt nur bei PERIOD = W#16#0000 auf.)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CAN_TINT: Uhrzeitalarm stornieren

Beschreibung

Mit der Anweisung können Sie Startdatum und -uhrzeit eines angegebenen Uhrzeitalarm-Organisationsbausteins löschen. Der Uhrzeitalarm wird dadurch deaktiviert und der Organisationsbaustein wird nicht mehr aufgerufen.

Möchten Sie den Uhrzeitalarm wieder nutzen, müssen Sie zuerst den Startzeitpunkt neu stellen (Anweisung "SET_TINT (Seite 5607)") und dann den Uhrzeitalarm aktivieren (Anweisung "ACT_TINT (Seite 5609)").

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, dessen Startdatum und -uhrzeit gelöscht werden soll (OB 10 bis OB 17).
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
80A0	Kein Startdatum/-uhrzeit festgelegt für den betreffenden Uhrzeitalarm-OB
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

ACT_TINT: Uhrzeitalarm aktivieren

Beschreibung

Mit der Anweisung können Sie einen Uhrzeitalarm-Organisationsbaustein aktivieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ACT_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, der aktiviert werden soll (OB 10 bis OB 17).
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
80A0	Startdatum/-uhrzeit sind nicht gestellt für den betreffenden Uhrzeitalarm-OB.
80A1	Die aktivierte Zeit liegt in der Vergangenheit; Fehler tritt nur auf bei Ausführung=einmal.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

QRY_TINT: Status des Uhrzeitalarms abfragen

Beschreibung

Mit der Anweisung können Sie sich den Status eines Uhrzeitalarm-Organisationsbausteins am Ausgangsparameter STATUS anzeigen lassen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY_TINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, dessen Status abgefragt wird (OB 10 bis OB 17).
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.
STATUS	Output	WORD	E, A, M, D, L	Status des Uhrzeitalarms; siehe folgende Tabelle.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Hinweis

Falls RET_VAL einen von Null verschiedenen Wert enthält, dürfen Sie STATUS nicht auswerten.

Bit	Wert	Bedeutung
0	0	In Run.
	1	Während des Anlaufs.

Bit	Wert	Bedeutung
1	0	Der Uhrzeitalarm ist freigegeben.
	1	Der Uhrzeitalarm ist gesperrt durch "DIS_IRT (Seite 5626)".
2	0	Der Uhrzeitalarm ist nicht aktiviert oder abgelaufen.
	1	Der Uhrzeitalarm ist aktiviert.
4	0	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben existiert nicht.
	1	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben ist vorhanden.
Andere		Immer "0"

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Verzögerungsalarm

Verwendung von Verzögerungsalarmen

Definition

Nachdem Sie die Anweisung "SRT_DINT (Seite 5613)" aufgerufen haben, wird vom Betriebssystem nach Ablauf der parametrierten Verzögerungszeit ein Alarm generiert, d. h. der parametrierte Verzögerungsalarm-OB wird aufgerufen.

Voraussetzungen für Aufruf

Damit ein Verzögerungsalarm vom Betriebssystem aufgerufen werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Verzögerungsalarm-OB muss gestartet sein durch die Anweisung "SRT_DINT (Seite 5613)".
- Der Verzögerungsalarm-OB darf nicht per Projektierung abgewählt sein.
- Der Verzögerungsalarm-OB muss in der CPU vorhanden sein.

Zweck der Anweisungen "SRT_DINT", "CAN_DINT" und "QRY_DINT"

Die Anweisungen verwenden Sie, um Verzögerungsalarme

- zu starten ("SRT_DINT (Seite 5613)")
- zu stornieren ("CAN_DINT (Seite 5614)")
- abzufragen ("QRY_DINT (Seite 5614)").

Auswirkungen auf den Verzögerungsalarm

Die folgende Tabelle zeigt, welche Situationen welche Auswirkungen auf den Verzögerungsalarm haben.

Wenn ...	und ...	dann ...
ein Verzögerungsalarm gestartet wird (Aufruf von "SRT_DINT (Seite 5613)")	der Verzögerungsalarm bereits gestartet ist,	wird die Verzögerungszeit überschrieben; der Verzögerungsalarm wird erneut gestartet.
	der Verzögerungsalarm-OB nicht vorhanden ist zum Zeitpunkt seines Aufrufs,	erzeugt das Betriebssystem einen Prioritätsklassenfehler (Aufruf des OB 85; falls OB 85 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP).
	der Start in einem Anlauf-OB erfolgte und die Verzögerungszeit abgelaufen ist, bevor die CPU im RUN ist,	verzögert sich der Aufruf des Verzögerungsalarm-OBs, bis die CPU im RUN ist.
die Verzögerungszeit abgelaufen ist	ein zuvor gestarteter Verzögerungsalarm-OB noch in Bearbeitung ist,	erzeugt das Betriebssystem einen Zeitfehler (Aufruf des OB 80; falls der OB 80 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP).

Verhalten bei Neustart (Warmstart) und Kaltstart

Bei Neustart (Warmstart) und Kaltstart werden alle im Anwenderprogramm über Anweisungen eingestellte Verzögerungsalarm-Einstellungen gelöscht.

Start in einem Anlauf-OB

Ein Verzögerungsalarm kann in einem Anlauf-OB gestartet werden. Für den Aufruf des Verzögerungsalarm-OBs müssen zwei Bedingungen erfüllt sein:

- Die Verzögerungszeit ist abgelaufen.
- Die CPU ist im Betriebszustand RUN.

Wenn die Verzögerungszeit abgelaufen ist und die CPU sich noch nicht im Betriebszustand RUN befindet, dann verzögert sich der Aufruf des Verzögerungsalarm-OB, bis die CPU in RUN ist. Der Verzögerungsalarm-OB wird dann noch vor der ersten Anweisung im OB Main[OB 1] aufgerufen.

SRT_DINT: Verzögerungsalarm starten

Beschreibung

Mit der Anweisung können Sie einen Verzögerungsalarm starten, der nach Ablauf der parametrierten Verzögerungszeit (Parameter DTIME) einen Verzögerungsalarm-OB aufruft.

Am Parameter SIGN können Sie ein anwenderspezifisches Kennzeichen angeben, um den Start des Verzögerungsalarms zu kennzeichnen. Die Werte von DTIME und SIGN erscheinen wieder in der Startereignisinformation des angegebenen OB, wenn er ausgeführt wird.

Die Zeit zwischen Aufruf der Anweisung "SRT_DINT" und dem Start des Verzögerungsalarm-OBs ist maximal **eine Millisekunde** kleiner als die parametrierte Zeit, sofern keine Unterbrechungsereignisse den Aufruf verzögern.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SRT_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, der nach einer Verzögerungszeit gestartet wird (OB 20 bis OB 23)
DTIME	Input	TIME	E, A, M, D, L oder Konstante	Zeitwert der Verzögerung (1 bis 60000 ms) Größere Zeiten können Sie z. B. dadurch realisieren, dass Sie in einem Verzögerungsalarm-OB einen Zähler benutzen.
SIGN	Input	WORD	E, A, M, D, L oder Konstante	Kennzeichen, das beim Aufruf des Verzögerungsalarm-OBs in der Startereignisinformation des OBs erscheint
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
8091	Fehlerhafter Parameter DTIME
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CAN_DINT: Verzögerungsalarm stornieren

Beschreibung

Mit der Anweisung können Sie einen gestarteten Verzögerungsalarm (siehe "SRT_DINT (Seite 5613)") stornieren. Der Verzögerungsalarm-OB wird dann nicht aufgerufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, der storniert werden soll (OB 20 bis OB 23).
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
80A0	Verzögerungsalarm ist nicht gestartet.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

QRY_DINT: Zustand eines Verzögerungsalarms abfragen

Beschreibung

Mit der Anweisung können Sie den Zustand eines Verzögerungsalarms abfragen. Verzögerungsalarme werden von den Organisationsbausteinen OB 20 bis OB 23 verwaltet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY_DINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OB_NR	Input	INT	E, A, M, D, L oder Konstante	Nummer des OB, dessen Zustand abgefragt wird (OB 20 bis OB 23).
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.
STATUS	Output	WORD	E, A, M, D, L	Zustand des Verzögerungsalarms; siehe folgende Tabelle.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Hinweis

Falls RET_VAL einen von Null verschiedenen Wert enthält, dürfen Sie STATUS nicht auswerten.

Bit	Wert	Bedeutung
0	0	In Run.
	1	Während des Anlaufs.
1	0	Der Verzögerungsalarm ist freigegeben.
	1	Der Verzögerungsalarm ist gesperrt durch "DIS_IRT (Seite 5626)".
2	0	Der Verzögerungsalarm ist nicht aktiviert oder abgelaufen.
	1	Der Verzögerungsalarm ist aktiviert.
4	0	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben existiert nicht.
	1	Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben ist geladen.
Andere		Immer "0"

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Fehlerhafter Parameter OB_NR
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Synchronfehlerereignisse

Synchronfehlerereignisse maskieren

Einleitung

Synchronfehlerereignisse sind Programmier- und Zugriffsfehlerereignisse. Solche Fehlerereignisse treten auf durch die Programmierung mit falschen Operandenbereichen, -nummern oder falschen Adressen. Diese Fehlerereignisse **maskieren** heißt,

- maskierte Fehlerereignisse lösen keinen Fehler-OB aus und führen nicht zu einer programmierten Ersatzreaktion.
- von den maskierten Fehlern "merkt sich" die CPU die Fehler, die aufgetreten sind, in einem Ereignisstatusregister.

Die Maskierung erfolgt durch Aufruf der Anweisung "MSK_FLT (Seite 5622)".

Fehlerereignisse **demaskieren** heißt, eine vorher vorgenommene Maskierung wieder rückgängig zu machen und das zugehörige Bit im Ereignisstatusregister der aktuellen Prioritätsklasse zu löschen. Die Maskierung wird aufgehoben

- durch einen Aufruf der Anweisung "DMSK_FLT (Seite 5623)"
- wenn die aktuelle Prioritätsklasse beendet ist (nur bei S7-400).

Tritt ein Fehlerereignis nach seiner Demaskierung auf, dann wird vom Betriebssystem der zugehörige Fehler-OB gestartet. Für die Reaktion auf Programmierfehler programmieren Sie den OB 121 und für die Reaktion auf Zugriffsfehler programmieren Sie den OB 122.

Maskierte und aufgetretene Fehlerereignisse können Sie mit der Anweisung "READ_ERR (Seite 5624)" auslesen.

Hinweis

Unabhängig von einer Maskierung oder Demaskierung der Fehlerereignisse erfolgt bei S7-300 mit Ausnahme der CPU 318 ein Eintragen des Fehlerereignisses in den Diagnosepuffer, und die Sammelfehler-LED der CPU leuchtet.

Fehlerbehandlung allgemein

Wenn in einem Anwenderprogramm Programmier- und Zugriffsfehler auftreten, dann können Sie verschieden darauf reagieren:

- Sie können einen Fehler-OB programmieren und das Betriebssystem ruft diesen Fehler-OB auf, wenn ein entsprechendes Fehlerereignis auftritt.
- Sie können den Aufruf des Fehler-OB für jede Prioritätsklasse einzeln sperren. In diesem Fall geht die CPU nicht in STOP, wenn in der Prioritätsklasse ein entsprechender Fehler auftritt. Die CPU trägt die aufgetretenen Fehler in ein Ereignisstatusregister ein. Sie können an diesem Eintrag aber nicht erkennen, wann und wie oft dieser Fehler aufgetreten ist.

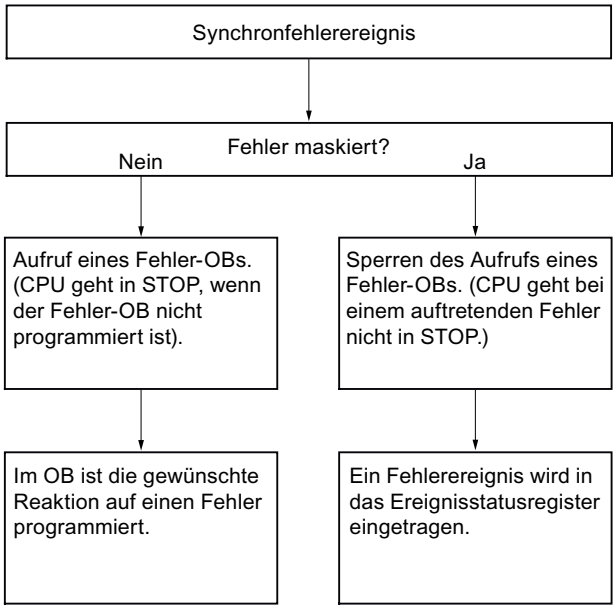

Fehlermaske

Die Synchronfehlerereignisse sind einem bestimmten Bitmuster zugeordnet, der **Fehlermaske**. Diese Fehlermaske finden Sie in den Eingangs- und Ausgangsparametern der Anweisungen "MSK_FLT (Seite 5622)", "DMSK_FLT (Seite 5623)", "READ_ERR (Seite 5624)" wieder.

Bei den Synchronfehlerereignissen wird zwischen Programmier- und Zugriffsfehlern unterschieden, die Sie in zwei Fehlermasken maskieren können. Die jeweiligen Fehlermasken finden Sie in den folgenden Bildern.

Programmierfehlermaske

Das folgende Bild zeigt Ihnen das Bitmuster der Fehlermaske für Programmierfehler. Die Fehlermaske für Programmierfehler steht in den Parametern "PRGFLT_..." (siehe im Folgenden "Programmierfehler, Low-Word" bzw. "Programmierfehler, High-Word").

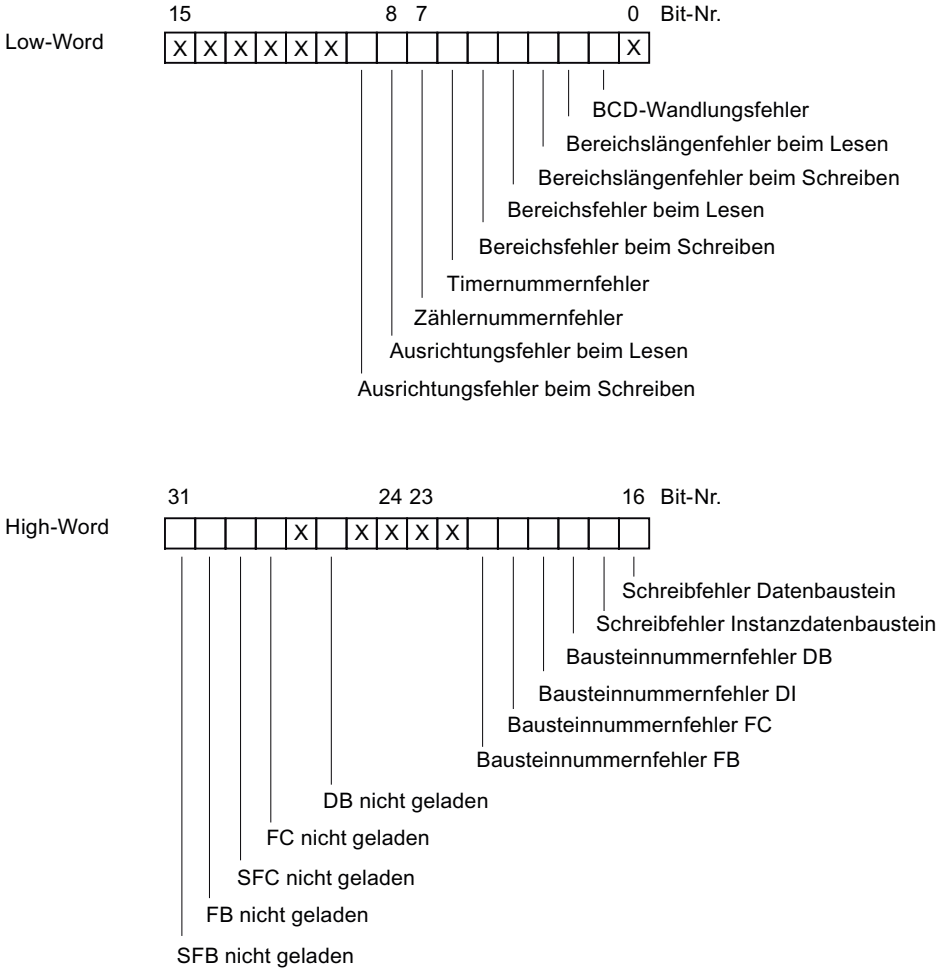

Legende: X nicht relevant

Nicht relevante Bits

Im Bild obigen bedeutet **x** für die ...

• ... Eingangsparameter	für "MSK_FLT (Seite 5622)", "DMSK_FLT (Seite 5623)", "READ_ERR (Seite 5624)"	= **"0"**
• ... Ausgangsparameter	für "MSK_FLT (Seite 5622)", "DMSK_FLT (Seite 5623)"	= **"1"** für S7-300 = **"0"** für S7-400
	für "READ_ERR (Seite 5624)"	= **"0"**

Zugriffsfehlermaske für alle CPUs

Das folgende Bild zeigt Ihnen für alle CPUs das Bitmuster der Fehlermaske für Zugriffsfehler. Die Fehlermaske für Zugriffsfehler steht in den Parametern ACCFLT_...

Eine Erläuterung der Zugriffsfehler finden Sie in den folgenden Tabellen (siehe unten).

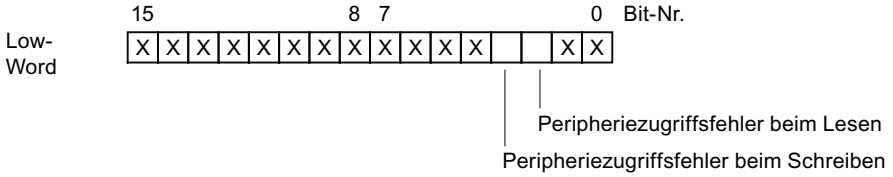

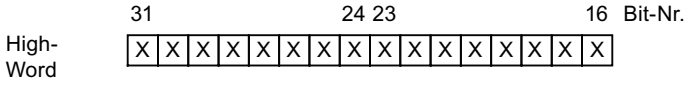

Legende: X nicht relevant

Beispiel

Das folgende Bild zeigt Ihnen für alle CPUs, wie das Low-Word der Fehlermaske für Zugriffsfehler mit allen maskierten Fehlern aussieht.

- als Eingangsparameter für "MSK_FLT (Seite 5622)"
- als Ausgangsparameter für "MSK_FLT (Seite 5622)"

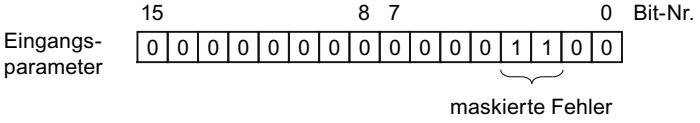

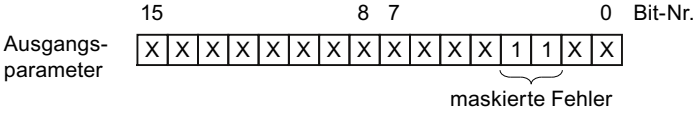

Legende:
- X nicht relevant
- 0 nicht maskiert
- 1 maskiert

Programmierfehler Low-Word

Die folgende Tabelle enthält die Fehler, die dem Low-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Mögliche Fehlerursachen für Programmierfehler, Low-Word

Fehler	Ereignis-ID (W#16#...)	Fehler tritt auf, ...
BCD-Wandlungsfehler	2521	... wenn der zu wandelnde Wert keine BCD-Zahl ist (z. B. 5E8).
Bereichslängenfehler beim Lesen	2522	... wenn ein adressierter Operand nicht vollständig innerhalb des möglichen Operandenbereichs liegt. Beispiel: MW 320 soll gelesen werden, obwohl der Merkerbereich nur 256 Byte groß ist.
Bereichslängenfehler beim Schreiben	2523	... wenn ein adressierter Operand nicht vollständig innerhalb des möglichen Operandenbereichs liegt. Beispiel: Ein Wert soll auf MW 320 geschrieben werden, obwohl der Merkerbereich nur 256 Byte groß ist.
Bereichsfehler beim Lesen	2524	... wenn bei indirekter, bereichsübergreifender Adressierung eine falsche Bereichskennung für den Operanden angegeben wird. Beispiel: Richtig: LAR1 P#E 12.0 L W[AR1, P#0.0] Falsch: LAR1 P#12.0 L W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet.
Bereichsfehler beim Schreiben	2525	... wenn bei indirekter, bereichsübergreifender Adressierung eine falsche Bereichskennung für den Operanden angegeben wird. Beispiel: Richtig: LAR1 P#E 12.0 T W[AR1, P#0.0] Falsch: LAR1 P#12.0 T W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet.
Timernummernfehler	2526	... wenn auf einen nicht vorhandenen Timer zugegriffen wird. Beispiel:SI T [MW 0] mit MW 0 = 129; der Timer 129 soll gestartet werden, obwohl nur 128 Timer vorhanden sind.
Zählernummernfehler	2527	... wenn auf einen nicht vorhandenen Zähler zugegriffen wird. Beispiel: ZV Z [MW 0] mit MW 0 = 600; es wird auf den Zähler 600 zugegriffen, obwohl nur 512 Zähler vorhanden sind (CPU 416-1).

Fehler	Ereignis-ID (W#16#...)	Fehler tritt auf, ...
Ausrichtungsfehler beim Lesen	2528	... wenn ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert wird. Beispiel: Richtig: LAR1 P#M12.0 L B[AR1, P#0.0] Falsch: LAR1 P#M12.4 L B[AR1, P#0.0]
Ausrichtungsfehler beim Schreiben	2529	... wenn ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert wird. Beispiel: Richtig: LAR1 P#M12.0 T B[AR1, P#0.0] Falsch: LAR1 P#M12.4 T B[AR1, P#0.0]

Programmierfehler High-Word

Die folgende Tabelle enthält die Fehler, die dem High-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Mögliche Fehlerursachen für Programmierfehler, High-Word

Fehler	Ereignis-ID (W#16#...)	Fehler tritt auf, ...
Schreibfehler Datenbaustein	2530	... wenn der Datenbaustein, auf den geschrieben wird, schreibgeschützt ist.
Schreibfehler Instanzdatenbaustein	2531	... wenn der Instanzdatenbaustein, auf den geschrieben wird, schreibgeschützt ist.
Bausteinnummernfehler DB	2532	... wenn ein Datenbaustein aufgeschlagen wird, dessen Nummer größer als die größte zulässige Nummer ist.
Bausteinnummernfehler DI	2533	... wenn ein Instanzdatenbaustein aufgeschlagen wird, dessen Nummer größer als die größte zulässige Nummer ist.
Bausteinnummernfehler FC	2534	... wenn eine Funktion aufgerufen wird, deren Nummer größer als die größte zulässige Nummer ist.
Bausteinnummernfehler FB	2535	... wenn ein Funktionsbaustein aufgerufen wird, dessen Nummer größer als die größte zulässige Nummer ist.
DB nicht geladen	253A	... wenn der aufzuschlagende Datenbaustein nicht geladen ist.
Anweisung nicht geladen	253C bis 253F	... wenn die aufzurufende Anweisung nicht geladen ist.

Zugriffsfehler

Die folgende Tabelle enthält für alle CPUs die Fehler, die der Fehlermaske für Zugriffsfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Fehler	Ereignis-ID (W#16#...)	Fehler tritt auf, ...
Peripheriezugriffsfehler beim Lesen	2942	... wenn der Adresse im Peripheriebereich keine Signalbaugruppe zugeordnet ist. Oder ... wenn der Zugriff auf diesen Peripheriebereich nicht innerhalb der eingestellten Baugruppenüberwachungszeit quittiert wird (Quittungsverzug).
Peripheriezugriffsfehler beim Schreiben	2943	... wenn der Adresse im Peripheriebereich keine Signalbaugruppe zugeordnet ist. oder ... wenn der Zugriff auf diesen Peripheriebereich nicht innerhalb der eingestellten Baugruppenüberwachungszeit quittiert wird (Quittungsverzug).

MSK_FLT: Synchronfehlerereignisse maskieren

Beschreibung

Die Anweisung nutzen Sie, um die Reaktion der CPU auf Synchronfehlerereignisse zu steuern. Dazu maskieren Sie die jeweiligen Synchronfehlerereignisse (Fehlermasken siehe Synchronfehlerereignisse maskieren (Seite 5616)). Mit dem Aufruf von "MSK_FLT" maskieren Sie die Synchronfehlerereignisse in der aktuellen Prioritätsklasse.

Wenn Sie in den Eingangsparametern einzelne Bits der Synchronfehlermasken auf "1" setzen, dann behalten andere, vorher bereits gesetzte Bits ihren Wert "1". Sie erhalten neue Fehlermasken, die Sie über die Ausgangsparameter auslesen können. Die von Ihnen maskierten Synchronfehlerereignisse rufen keinen OB auf, sondern werden nur in einem Ereignisstatusregister eingetragen. Das Ereignisstatusregister können Sie mit der Anweisung "READ_ERR (Seite 5624)" auslesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MSK_FLT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_SET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu maskierende Programmierfehler
ACCFLT_SET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu maskierende Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PRGFLT_MASKED	Output	DWORD	E, A, M, D, L	Maskierte Programmierfehler
ACCFLT_MASKED	Output	DWORD	E, A, M, D, L	Maskierte Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Keiner der Fehler war bereits maskiert.
0001	Mindestens einer der Fehler war bereits maskiert, trotzdem werden die anderen Fehler maskiert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DMSK_FLT: Synchronfehlerereignisse demaskieren

Beschreibung

Die Anweisung nutzen Sie, um die mit "MSK_FLT (Seite 5622)" maskierten Fehlerereignisse zu demaskieren. Dazu müssen Sie in den Eingangsparametern die entsprechenden Bits der Fehlermasken auf "1" setzen. Mit dem Aufruf von "DMSK_FLT" demaskieren Sie die entsprechenden Synchronfehlerereignisse der aktuellen Prioritätsklasse. Gleichzeitig werden die abgefragten Einträge im Ereignisstatusregister gelöscht. Die neuen Fehlermasken können Sie über die Ausgangsparameter auslesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DMSK_FLT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_ RESET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu demaskierende Programmierfehler
ACCFLT_ RESET_MASK	Input	DWORD	E, A, M, D, L oder Konstante	Zu demaskierende Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PRGFLT_ MASKED	Output	DWORD	E, A, M, D, L	Noch maskierte Programmierfehler
ACCFLT_ MASKED	Output	DWORD	E, A, M, D, L	Noch maskierte Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Alle angegebenen Fehler wurden demaskiert.
0001	Mindestens einer der Fehler war nicht maskiert, trotzdem werden die anderen Fehler demaskiert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

READ_ERR: Ereignisstatusregister auslesen

Beschreibung

Mit der Anweisung lesen Sie das Ereignisstatusregister aus. Das Ereignisstatusregister entspricht in seinem Aufbau den Programmier- und Zugriffsfehlermasken, die Sie mit "MSK_FLT (Seite 5622)" und "DMSK_FLT (Seite 5623)" als Eingangsparameter programmieren können.

In die Eingangsparameter tragen Sie ein, welche Synchronfehlerereignisse Sie aus dem Ereignisstatusregister abfragen wollen. Mit dem Aufruf von "READ_ERR" lesen Sie die gewünschten Einträge aus dem Ereignisstatusregister aus und löschen gleichzeitig diese Einträge.

Aus dem Ereignisstatusregister entnehmen Sie die Information, welche der maskierten Synchronfehler in der aktuellen Prioritätsklasse mindestens einmal aufgetreten sind. Ein gesetztes Bit bedeutet, dass der zugehörige maskierte Synchronfehler mindestens einmal aufgetreten ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_ERR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PRGFLT_QUERY	Input	DWORD	E, A, M, D, L oder Konstante	Abfrage der Programmierfehler
ACCFLT_QUERY	Input	DWORD	E, A, M, D, L oder Konstante	Abfrage der Zugriffsfehler
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PRGFLT_CLR	Output	DWORD	E, A, M, D, L	Aufgetretene Programmierfehler
ACCFLT_CLR	Output	DWORD	E, A, M, D, L	Aufgetretene Zugriffsfehler

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Alle abgefragten Fehler sind maskiert.
0001	Mindestens einer der abgefragten Fehler ist nicht maskiert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Asynchronfehlerereignisse

Alarm- und Asynchronfehlerereignisse verzögern und sperren

Zweck der Anweisungen "DIS_IRT", "EN_IRT", "DIS_AIRT", "EN_AIRT"

Die Anweisungen verwenden Sie, um die Bearbeitung der Alarme und asynchronen Fehlerereignisse

- zu sperren über alle nachfolgenden CPU-Zyklen mit "DIS_IRT (Seite 5626)" oder
- höherpriore Prioritätsklassen zu verzögern bis zum OB-Ende mit "DIS_AIRT (Seite 5626)" sowie
- wieder freizugeben mit "EN_IRT (Seite 5628)" bzw. "EN_AIRT (Seite 5630)".

Die Bearbeitung von Alarmen und asynchronen Fehlerereignissen programmieren Sie im Anwenderprogramm. Dazu müssen Sie die zugehörigen OBs programmieren.

Vorteil der Anweisungen "DIS_AIRT" und "EN_AIRT"

Das Verzögern von höherprioren Alarm- und Asynchronfehlerereignissen mit der Anweisung "DIS_AIRT (Seite 5629)" und die Freigabe mit der Anweisung "EN_AIRT (Seite 5630)" hat für Sie folgenden Vorteil:

Die Anzahl der Alarmverzögerungen wird von der CPU mitgezählt. Eine von Ihnen programmierte Verzögerung von Alarm- bzw. Asynchronfehlerereignissen kann nicht durch aufgerufene Standard-FCs aufgehoben werden, wenn in diesen Standard-FCs ebenfalls Alarm- und Asynchronfehlerereignisse gesperrt und wieder freigegeben werden.

Alarmklassen

Die Alarme sind in verschiedene Alarmklassen unterteilt. Die folgende Tabelle enthält alle Alarmklassen und die zugehörigen OBs.

Alarmklasse	OB
Uhrzeitalarme	OB 10 bis OB 17
Verzögerungsalarme	OB 20 bis OB 23
Weckalarme	OB 30 bis OB 38
Prozessalarme	OB 40 bis OB 47

Alarmklasse	OB
Alarme für DPV1	OB 55 bis OB 57
Multicomputingalarm	OB 60
Redundanzfehleralarme	OB 70, OB 72
Asynchrone Fehleralarme	OB 80 bis OB 87 (siehe unten)
Synchrone Fehleralarme	OB 121, OB 122 (Die Bearbeitung der synchronen Fehleralarme maskieren bzw. demaskieren Sie mit den Anweisungen "MSK_FLT (Seite 5622)", "DMSK_FLT (Seite 5623)" und "READ_ERR (Seite 5624)")

Asynchrone Fehlerereignisse

Die folgende Tabelle enthält alle asynchronen Fehlerereignisse, auf die Sie im Anwenderprogramm mit einem entsprechenden OB reagieren können.

Asynchrone Fehlerereignisse	OB
Zeitfehler (z. B. Überschreiten der Zykluszeit)	OB 80
Stromversorgungsfehler (z. B. Batteriefehler)	OB 81
Diagnosealarm (z. B. defekte Sicherung an einer Signalbaugruppe)	OB 82
Ziehen / Stecken-Alarm	OB 83
CPU-Hardwarefehler (z. B. Schnittstellenfehler)	OB 84
Programmablauffehler	OB 85
Baugruppenträgerausfall	OB 86
Kommunikationsfehler	OB 87

DIS_IRT: Unterbrechungsereignis sperren

Beschreibung

Mit der Anweisung sperren Sie die Bearbeitung neuer Alarm- und Asynchronfehlerereignisse. Sperren heißt, dass das Betriebssystem der CPU bei einem Unterbrechungsereignis

- **weder** einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft,
- **noch** die festgelegte Reaktion bei nicht programmiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

Die Sperre der Bearbeitung von Alarm- und Asynchronfehlerereignissen bleibt über alle Prioritätsklassen gültig. Die Sperre kann nur mit der Anweisung "EN_IRT (Seite 5628)" aufgehoben werden bzw. auch bei Neustart (Warmstart) und bei Kaltstart.

Ob das Betriebssystem das aufgetretene Alarm- bzw. Asynchronfehlerereignis in den Diagnosepuffer einträgt, ist abhängig von Ihrer Wahl des Eingangsparameters MODE.

Hinweis

Beachten Sie, dass bei Programmierung der Anweisung "DIS_IRT"alle eintreffenden Alarme verworfen werden!

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS_IRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Gibt an, welche Alarm- und Asynchronfehlerereignisse gesperrt werden.
OB_NR	Input	INT	E, A, M, D, L oder Konstante	OB-Nummer
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

MODE (B#16#...)	Bedeutung
00	Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden gesperrt. (Nicht gesperrt werden Synchronfehlerereignisse.) Dem Parameter OB_NR weisen Sie den Wert "0" zu. Die Einträge in den Diagnosepuffer erfolgen weiterhin.
01	Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden gesperrt. Die Alarmklasse kennzeichnen Sie wie folgt: • Uhrzeitalarme: 10 • Verzögerungsalarme: 20 • Weckalarme: 30 • Prozessalarme: 40 • Alarme für DPV1: 50 • Multicomputingalarm: 60 • Redundanzfehleralarme: 70 • Asynchrone Fehleralarme: 80 Die Einträge in den Diagnosepuffer erfolgen weiterhin.
02	Alle neu auftretenden Ereignisse eines angegebenen Alarms werden gesperrt. Den Alarm kennzeichnen Sie durch die OB-Nummer. Die Einträge in den Diagnosepuffer erfolgen weiterhin.
80	Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden gesperrt und sie werden auch nicht mehr in den Diagnosepuffer eingetragen. Dem Parameter OB_NR weisen Sie den Wert "0" zu. Das Betriebssystem trägt das Ereignis W#16#5380 in den Diagnosepuffer ein.
81	Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden gesperrt und sie werden auch nicht mehr in den Diagnosepuffer eingetragen. Das Betriebssystem trägt das Ereignis W#16#5380 in den Diagnosepuffer ein.
82	Alle neu auftretenden Ereignisse eines angegebenen Alarms werden gesperrt und sie werden auch nicht mehr in den Diagnosepuffer eingetragen. Das Betriebssystem trägt das Ereignis W#16#5380 in den Diagnosepuffer ein.

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

EN_IRT: Unterbrechungsereignis freigeben

Beschreibung

Mit der Anweisung geben Sie die mit der Anweisung "DIS_IRT (Seite 5626)" gesperrte Bearbeitung neuer Alarm- und Asynchronfehlerereignisse wieder frei. Freigeben heißt, dass das Betriebsystem der CPU bei einem Unterbrechungsereignis

- einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft
  oder
- die festgelegte Reaktion bei nichtprogrammiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN_IRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Gibt an, welche Alarm- und Asynchronfehlerereignisse freigegeben werden (siehe unten).
OB_NR	Input	INT	E, A, M, D, L oder Konstante	OB-Nummer
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

MODE	Bedeutung
0	Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden freigegeben.
1	Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden freigegeben. Die Alarmklasse kennzeichnen Sie wie folgt: • Uhrzeitalarme: 10 • Verzögerungsalarme: 20 • Weckalarme: 30 • Prozessalarme: 40 • Alarme für DPV1: 50 • Multicomputingalarm: 60 • Redundanzfehleralarme: 70 • Asynchrone Fehleralarme: 80
2	Alle neu auftretenden Ereignisse eines angegebenen Alarms werden freigegeben. Den Alarm kennzeichnen Sie durch die OB-Nummer.

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	Es ist kein Fehler aufgetreten.
8090	Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.
8091	Der Eingangsparameter MODE enthält einen unzulässigen Wert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DIS_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen verzögern

Beschreibung

Mit der Anweisung verzögern Sie die Bearbeitung von Alarm-OBs und Asynchronfehler-OBs, deren Priorität höher ist als die des aktuellen OB. Sie können in einem OB die Anweisung "DIS_AIRT" mehrmals aufrufen. Die Aufrufe der der Anweisung werden vom Betriebssystem gezählt. Die Bearbeitungsverzögerung gilt solange, bis Sie mit der Anweisung "EN_AIRT (Seite 5630)" jede mit "DIS_AIRT" verzögerte Bearbeitung von Alarm-OBs und Asynchronfehler-OBs aufheben **oder** der aktuelle OB abgearbeitet ist.

Die anstehenden Alarm- oder Asynchronfehlerereignisse werden bearbeitet, sobald die Bearbeitungsverzögerung mit der Anweisung "EN_AIRT (Seite 5630)" aufgehoben wurde oder die aktuelle OB-Bearbeitung beendet ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS_AIRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Anzahl der Verzögerungen (= Anzahl der DIS_AIRT-Aufrufe)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Die folgende Tabelle enthält den Rückgabewert von "DIS_AIRT", der über den Parameter RET_VAL ausgegeben wird.

Rückgabewert	Beschreibung
n	"n" zeigt nach Ablauf der Anweisung die Anzahl der Bearbeitungsverzögerungen, also der "DIS_AIRT"-Aufrufe an (Die Alarmbearbeitung ist erst wieder freigegeben, wenn n = 0 ist; siehe "EN_AIRT (Seite 5630)").

EN_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen freigeben

Beschreibung

Mit der Anweisung geben Sie die mit "DIS_AIRT (Seite 5629)" verzögerte Bearbeitung von höherprioren Alarm- bzw. Asynchronfehlerereignissen wieder frei. Sie müssen dabei jede einzelne Bearbeitungsverzögerung mit der Anweisung "EN_AIRT" beenden.

Beispiel

Wenn Sie zum Beispiel mit 5 "DIS_AIRT (Seite 5629)"-Aufrufen Alarme 5mal verzögert haben, dann müssen Sie mit 5 "EN_AIRT"-Aufrufen jede dieser Alarmverzögerungen auch wieder aufheben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN_AIRT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Anzahl der noch programmierten Verzögerungen nach Ablauf von "EN_AIRT" bzw. Fehlermeldung.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wie Sie die Fehlerinformationen des Parameter RET_VAL auswerten, ist im Kapitel Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171) erläutert. In diesem Kapitel finden Sie auch die allgemeinen Fehlerinformationen der Anweisungen. Die folgende Tabelle enthält den für "EN_AIRT" spezifischen Rückgabewert bzw. die Fehlerinformation, die über den Parameter RET_VAL ausgegeben werden kann.

Rückgabewert und Fehler	Beschreibung
n	"n" zeigt nach Ablauf der Anweisung die Anzahl der Bearbeitungsverzögerungen, also der "EN_AIRT"-Aufrufe an (Die Alarmbearbeitung ist erst wieder freigegeben, wenn n = 0 ist).
W#16#8080	Obwohl die Alarmbearbeitung bereits freigegeben war, wurde "EN_AIRT" aufgerufen.

Multicomputingalarm

MP_ALM: Multicomputingalarm

Beschreibung

Der Aufruf der Anweisung löst beim Multicomputing den Multicomputingalarm aus. Das führt zum synchronisierten Start des OB 60 auf allen zugehörigen CPUs. Beim Einprozessorbetrieb und beim Betrieb im segmentierten Baugruppenträger wird der OB 60 nur auf derjenigen CPU gestartet, auf der Sie die Anweisung aufgerufen haben.

Mit dem Eingangsparameter JOB können Sie die Ursache für den von Ihnen gewünschten Multicomputingalarm kennzeichnen. Diese Auftragskennung wird an alle zugehörigen CPUs übertragen und kann von Ihnen im Multicomputingalarm-OB (OB 60) ausgewertet werden.

Sie können die Anweisung an jeder Stelle Ihres Programms aufrufen. Da der Aufruf aber nur im Betriebszustand RUN sinnvoll ist, wird beim Aufruf im Betriebszustand ANLAUF der Multicomputingalarm unterdrückt. Dies wird Ihnen über einen Funktionswert mitgeteilt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MP_ALM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
JOB	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Mögliche Werte: 1 bis 15
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Der Eingangsparameter JOB enthält einen unzulässigen Wert.
80A0	Auf der eigenen oder auf einer anderen CPU ist die OB 60-Bearbeitung des vorangegangenen Multicomputingalarms noch nicht abgeschlossen.
80A1	Falscher Betriebszustand (ANLAUF statt RUN)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.3.8 Meldungen

Aufbau der Diagnosedaten

Datensatz 0 und 1 der Systemdaten

Die Diagnosedaten einer Baugruppe stehen in den Datensätzen 0 und 1 des Systemdatenbereichs siehe Datensätze schreiben und lesen (Seite 5593):

- Der Datensatz 0 enthält 4 Byte Diagnosedaten, die den aktuellen Zustand einer Signalbaugruppe beschreiben.
- Der Datensatz 1 enthält
  - die 4 Byte Diagnosedaten, die auch im Datensatz 0 stehen, und
  - die baugruppenspezifischen Diagnosedaten.

Aufbau und Inhalt der Diagnosedaten

Im Folgenden ist der Aufbau und der Inhalt der einzelnen Bytes der Diagnosedaten beschrieben. Generell gilt: Wenn ein Fehler auftritt, dann wird das entsprechende Bit auf "1" gesetzt.

Byte	Bit	Bedeutung	Bemerkung	Datensatz
0	0	Baugruppenstörung		0 und 1
	1	Fehler intern		
	2	Fehler extern		
	3	Kanalfehler vorhanden		
	4	Externe Hilfsspannung fehlt		
	5	Frontstecker fehlt		
	6	Parametrierung fehlt		
	7	Falsche Parameter in der Baugruppe		

Byte	Bit	Bedeutung	Bemerkung	Datensatz
1	0 bis 3	Baugruppenklasse	0101: Analogbaugruppe 0000: CPU 1000: Funktionsbaugruppe 1100: CP 1111: Digitalbaugruppe 0011: DP-Normslave 1011: I-Slave 0100: IM	0 und 1
	4	Kanalinformation vorhanden		
	5	Anwenderinformation vorhanden		
	6	Diagnosealarm von Stellvertreter		
	7	Wartungsbedarf (nur bei PROFINET IO)		
2	0	Speichermodul falsch oder fehlt		0 und 1
	1	Kommunikationsstörung		
	2	Betriebszustand	0: RUN 1: STOP	
	3	Zykluszeitüberwachung angesprochen		
	4	Baugruppeninterne Versorgungsspannung ausgefallen		
	5	Batterie leer		
	6	Gesamte Pufferung ausgefallen		
	7	Wartungsanforderung (nur bei PROFINET IO)		
3	0	Erweiterungsgeräteausfall		0 und 1
	1	Prozessorausfall		
	2	EPROM-Fehler		
	3	RAM-Fehler		
	4	ADU/DAU-Fehler		
	5	Sicherungsausfall		
	6	Prozessalarm verloren		
	7	reserviert		

Byte	Bit	Bedeutung	Bemerkung	Datensatz
4	0 bis 6	Kanaltyp	B#16#70: Digitaleingabe B#16#72: Digitalausgabe B#16#71: Analogeingabe B#16#73: Analogausgabe B#16#74: FM-POS B#16#75: FM-REG B#16#76: FM-ZAEHL B#16#77: FM-TECHNO B#16#78: FM-NCU B#16#79: bis B#16#7D: reserviert B#16#7E: US300 B#16#7F: reserviert	1
	7	Weiterer Kanaltyp vorhanden?	0: nein 1: ja	
5	0 bis 7	Anzahl der Diagnosebits, die eine Baugruppe pro Kanal ausgibt.	Die Anzahl der Diagnosebits pro Kanal ist auf Bytegrenzen aufgerundet.	1
6	0 bis 7	Anzahl der gleichartigen Kanäle einer Baugruppe	Wenn auf einer Baugruppe unterschiedliche Kanaltypen existieren, dann wird für jeden Kanaltyp die Struktur ab Byte 4 im Datensatz 1 wiederholt.	1
7	0	Kanalfehler Kanal 0/ Kanalgruppe 0	Erstes Byte des Kanalfehlervektors (Die Länge des Kanalfehlervektors richtet sich nach der Kanalzahl und ist auf Bytegrenzen aufgerundet.)	1
	1	Kanalfehler Kanal 1/ Kanalgruppe 1		
	2	Kanalfehler Kanal 2/ Kanalgruppe 2		
	3	Kanalfehler Kanal 3/ Kanalgruppe 3		
	4	Kanalfehler Kanal 4/ Kanalgruppe 4		
	5	Kanalfehler Kanal 5/ Kanalgruppe 5		
	6	Kanalfehler Kanal 6/ Kanalgruppe 6		
	7	Kanalfehler Kanal 7/ Kanalgruppe 7		
...	-	Kanalspezifische Fehler (siehe Aufbau der kanalspezifischen Diagnosedaten (Seite 5634))		1

Aufbau der kanalspezifischen Diagnosedaten

Kanalspezifische Fehler

Ab dem Byte, das unmittelbar hinter dem Kanalfehlervektor liegt, werden für jeden Kanal der Baugruppe die kanalspezifischen Fehler angezeigt. Im Folgenden zeigen wir Ihnen den Aufbau der kanalspezifischen Diagnose für die unterschiedlichen Kanaltypen. Für die Bitbelegung gilt:

- 1 = Fehler
- 0 = kein Fehler

Analogeingabekanal

Diagnosebyte für einen Analogeingabekanal

Bit	Bedeutung	Bemerkung
0	Projektierungs-/Parametrierungsfehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x50
1	Gleichtaktfehler (Common-Mode-Fehler)	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x51
2	P-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x52
3	M-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x53
4	Drahtbruch	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x54
5	Referenzkanal-Fehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x55
6	Messbereichsunterschreitung	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x56
7	Messbereichsüberschreitung	meldbar mit WR_USMSG (Seite 5667)und EVENTN = W#16#8x57

Analogausgabekanal

Diagnosebyte für einen Analogausgabekanal

Bit	Bedeutung	Bemerkung
0	Projektierungs-/Parametrierungsfehler	meldbar mit WR_USMSG (Seite 5667)und EVENTN = W#16#8x60
1	Gleichtaktfehler (Common-Mode-Fehler)	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x61
2	P-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x62
3	M-Kurzschluss	meldbar mit WR_USMSG (Seite 5667)und EVENTN = W#16#8x63
4	Drahtbruch	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x64
5	0	Reserviert
6	Lastspannnung fehlt	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x66
7	0	reserviert

Digitaleingabekanal

Diagnosebyte für einen Digitaleingabekanal

Bit	Bedeutung	Bemerkung
0	Projektierungs-/Parametrierungsfehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x70
1	Massefehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x71
2	P-Kurzschluss (Geber)	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x72
3	M-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x73
4	Drahtbruch	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x74
5	Geberversorgung fehlt	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x75
6	0	reserviert
7	0	reserviert

Digitalausgabekanal

Diagnosebyte für einen Digitalausgabekanal

Bit	Bedeutung	Bemerkung
0	Projektierungs-/Parametrierungsfehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x80
1	Massefehler	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x81
2	P-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x82
3	M-Kurzschluss	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x83
4	Drahtbruch	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x84
5	Sicherungsfall	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x85
6	Lastspannung fehlt	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x86
7	Übertemperatur	meldbar mit WR_USMSG (Seite 5667) und EVENTN = W#16#8x87

Anweisungen zur Erzeugung von PLC-Meldungen mit Instanz-DB

Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen

Anweisungen zur Erzeugung bausteinbezogener Meldungen

Sie erzeugen eine PLC-Meldung, indem Sie in Ihrem Programm eine der folgenden Anweisungen aufrufen:

- "NOTIFY (Seite 5648)"
- "NOTIFY_8P (Seite 5640)"
- "ALARM (Seite 5650)"
- "ALARM_8 (Seite 5643)"
- "ALARM_8P (Seite 5645)"

Diese Anweisungen haben folgende Eigenschaften:

- Bei "NOTIFY (Seite 5648)" und "NOTIFY_8P (Seite 5640)" führt jeder beim Aufruf erkannte Signalwechsel 0 -> 1 oder 1 > 0 zum Senden einer Meldung.
- Auch bei "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" und "ALARM_8P (Seite 5645)" führt bei inaktivem Quittierungsgetriggerten Melden jeder beim Aufruf erkannte Signalwechsel zum Senden einer Meldung.
  Falls Sie hingegen das Quittierungsgetriggerte Melden aktiviert haben, führt nicht jeder erkannte Signalwechsel zum Senden einer Meldung (genaueres siehe unten). In diesem Fall wird eine gesendete Meldung in der Meldungsanzeige nicht angezeigt.
- Nach dem Bausteindurchlauf sind die Begleitwerte (Eingänge SD_i) vollständig erfasst und der Meldung zugeordnet (siehe "Sende- und Empfangsparameter" in Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384).)
  Bezüglich der Konsistenz gegenüber höherprioren Prioritätsklassen gilt für die Begleitwerte: Jeder Begleitwert SD_i ist in sich konsistent.
- Mit den Zustandsparametern DONE, ERROR und STATUS überwachen Sie den Bearbeitungszustand des Bausteins (siehe "Zustandsparameter" in Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)).

Hinweis

Die Parameter ID und EV_ID werden nur beim Erstaufruf des Bausteins ausgewertet (die Aktualparameter oder die vordefinierten Werte aus der Instanz).

Anmeldungen von Anzeigegeräten

Voraussetzung dafür, dass die Anweisungen zur Erzeugung von PLC-Meldungen bei einem erkannten Signalwechsel eine Meldung senden, ist, dass sich mindestens ein Anzeigegerät für PLC-Meldungen angemeldet hat. Liegt keine Anmeldung vor, liefert der Parameter STATUS den Wert "1".

Signalwechselerfassung

Pro Instanz einer Meldeanweisung steht ein Meldespeicher mit 2 Speicherplätzen zur Verfügung.
Zunächst ist dieser Meldespeicher leer. Sobald die Anweisung einen Signalwechsel am Eingang SIG bzw. an einem der Eingänge SIG_1, ... SIG_8 erkennt, wird dieser in den ersten Speicherplatz eingetragen. Er bleibt so lange belegt, bis die zugehörige Meldung versendet wird.
Der nächste erkannte Signalwechsel am Eingang SIG bzw. an einem der Eingänge SIG_1, ... SIG_8 wird dann in den zweiten Speicherplatz eingetragen. Falls der erste Speicherplatz noch belegt ist und weitere Signalwechsel folgen, wird stets der zweite Speicherplatz im Meldespeicher überschrieben.
Dieser Meldungsverlust wird Ihnen über die Ausgangsparameter ERROR und STATUS (ERROR=0, STATUS=11) angezeigt. Außerdem erhalten die angemeldeten Anzeigegeräte mit der nächsten Meldung, die gesendet werden kann, darüber eine Mitteilung.
Falls der erste Speicherplatz frei wird, wird der zweite Speicherplatz in den ersten übertragen. Dadurch wird der zweite Speicherplatz wieder frei.

Quittierungsgetriggertes Melden

Um das Meldeaufkommen in Ihrer Anlage zu reduzieren, können Sie bei den Anweisungen "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" und "ALARM_8P (Seite 5645)" das Verfahren des Quittierungsgetriggerten Meldens einsetzen.

Es beruht darauf, dass nach dem Erzeugen einer Kommend-Meldung (Signalwechsel von 0 nach 1) für ein Signal so lange keine weiteren Meldungen für dieses Signal erzeugt werden, bis Sie es an einem Anzeigegerät quittiert haben. Die nächste Meldung, die nach dem Quittieren auf dem Anzeigegerät angezeigt wird, ist eine Gehend-Meldung (Signalwechsel von 1 nach 0). Anschließend beginnt der Meldezyklus mit einer Kommend-Meldung (Signalwechsel von 0 nach 1), die quittiert werden muss, erneut. Auf diese Weise können Sie über das Anzeigegerät das Melden von Signalwechseln (bis auf die Gehend-Meldung) kontrollieren.

Das Meldeverfahren (Quittierungsgetriggertes Melden aktiv oder inaktiv) für die Anweisungen "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" und "ALARM_8P (Seite 5645)" legen Sie bei der Projektierung CPU-weit fest.

Damit die Meldungsauswertung innerhalb Ihrer Anlage einheitlich ist, sollten Sie darauf achten, dass beim Einsatz des Quittierungsgetriggerten Meldens alle Anzeigegeräte dieses Verfahren beherrschen.

Hinweis

(Zum Einsatz von Anzeigegeräten, die das Quittierungsgetriggerte Melden nicht beherrschen.)

Falls Sie in Ihrer CPU das Quittierungsgetriggerte Melden aktiviert haben, schickt die CPU die Meldungen nur an diejenigen Anzeigegeräte, die dieses Meldeverfahren beherrschen. Falls keine Anzeigegeräte das Quittierungsgetriggerte Melden beherrschen, sendet die CPU also auch keine Meldungen. Dies wird Ihnen mit ERROR=1 und STATUS=1 einmalig angezeigt.

Meldungsquittierung bei den Anweisungen "ALARM", "ALARM_8" und "ALARM_8P"

Es wird das zentrale Quittierkonzept eingesetzt. D. h., wenn Sie an einem Anzeigegerät eine Meldung quittiert haben, wird diese Quittierinformation zunächst an die meldungserzeugende CPU geschickt. Von dort wird die Quittierinformation an alle dafür angemeldeten Teilnehmer verteilt.

Sie quittieren stets ein Signal und nicht eine einzelne Meldung. Falls z. B. mehrere steigende Flanken eines Signals gemeldet wurden und Sie das gekommene Ereignis quittieren, so gelten alle vorangegangenen gekommenen Ereignisse mit gleicher Meldungsnummer als quittiert.

Quittierungsanzeige

Die Anweisungen "NOTIFY (Seite 5648)" und "NOTIFY_8P (Seite 5640)" besitzen keine Quittierungsanzeige. Bei "ALARM (Seite 5650)" können Sie den Quittierungszustand den Ausgangsparametern ACK_UP und ACK_DN, bei "ALARM_8P (Seite 5645)" und "ALARM_8 (Seite 5643)" dem Ausgangsparameter ACK_STATE entnehmen. Diese Ausgänge werden zum Zeitpunkt des Bausteinaufrufs aktualisiert, falls der Steuerparameter EN_R den Wert "1" hat.

Meldungen per Anweisung oder Anzeigegerät sperren und freigeben

Es kann sinnvoll sein, Meldungen zu unterdrücken, z. B. bei Umbaumaßnahmen in Ihrer Anlage. Sie haben daher die Möglichkeit, Meldungen von einem Anzeigegerät oder von Ihrem Programm aus zu sperren und wieder freizugeben. Die Sperre / Freigabe gilt für alle Teilnehmer, die sich für die zugehörige Meldung angemeldet haben. Sie bleibt so lange bestehen, bis Sie die zugehörige Meldung wieder freigeben.

Falls Sie Meldungen gesperrt haben, so wird Ihnen das über die Ausgangsparameter ERROR und STATUS mitgeteilt (ERROR = 1, STATUS = 21).

Arbeitsspeicherbedarf der Anweisungen zur Erzeugung bausteinbezogener Meldungen

Die Anweisungen zur Erzeugung von PLC-Meldungen benötigen zu ihrer reibungslosen Funktion einen i. A. begleitwertabhängigen Kommunikationsdatenpuffer im Arbeitsspeicher der CPU (Codebereich). Die Größe des belegten Speichers entnehmen Sie der folgenden Tabelle.

Anweisung	Benötigter Speicherbedarf im Arbeitsspeicher der CPU in Byte
NOTIFY	200 + 2 * Länge der beim Erstaufruf an SD_1,...SD_10 angegebenen Begleitwerte
NOTIFY_8P	200 + 2 * Länge der beim Erstaufruf an SD_1,...SD_10 angegebenen Begleitwerte
ALARM	200 + 2 * Länge der beim Erstaufruf an SD_1,...SD_10 angegebenen Begleitwerte
ALARM_8	100
ALARM_8P	200 + 2 * Länge der beim Erstaufruf an SD_1,...SD_10 angegebenen Begleitwerte
AR_SEND	54

Anzahl übertragbarer Daten

Die Anzahl der über die Begleitwerte SD_i der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)" und "ALARM_8P (Seite 5645)" übertragbaren Daten darf eine maximale Länge nicht überschreiten. Diese maximale Datenlänge berechnet sich wie folgt:

maxleng =

min (pdu_lokal, pdu_remote) - diff - 4 * Anzahl benutzter SD_i-Parameter

Dabei ist:

- min (pdu_lokal, pdu_remote) der kleinste Wert der Zahlen pdu_lokal und pdu_remote
- pdu_lokal die maximale Länge der Datenblöcke der eigenen CPU (siehe Technische Daten Ihrer CPU)
- pdu_remote die maximale Länge der Datenblöcke der Anzeigegeräte
- diff = 48, falls das Quittierungsgetriggerte Melden aktiv ist, und 44, falls es inaktiv ist

Beispiel

Eine CPU 414-2 sende Meldungen über Industrial Ethernet an HMI. Das Quittierungsgetriggerte Melden sei inaktiv.

Es werden die Begleitwerte SD_1, SD_2 und SD_3 verwendet.

pdu_lokal = 480 Byte, pdu_remote = 480 Byte,

Anzahl benutzter SD_i-Parameter: 3

Damit gilt:

maxleng = min (480, 480) - 44 - 4 * 3 = 480 - 44 - 12 = 424

Die maximal übertragbare Datenlänge beträgt pro Anweisung also 424 Byte.

NOTIFY_8P: Bis zu acht Signalwechsel melden

Beschreibung

Die Anweisung "NOTIFY_8P" ist die Erweiterung von "NOTIFY (Seite 5648)" auf acht Signale.

Eine Meldung wird erzeugt, wenn bei mindestens einem Signal ein Signalwechsel erkannt wurde. Beim Erstaufruf von "NOTIFY_8P" wird stets eine Meldung erzeugt. Für alle acht Signale gibt es eine gemeinsame Meldungsnummer, die am Anzeigegerät in acht Teilmeldungen aufgesplittet wird.

Pro Instanz von "NOTIFY_8P" steht ein Meldespeicher mit 2 Speicherplätzen zur Verfügung. Genaueres zur Zwischenspeicherung von Signalwechseln siehe Abschnitt Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen (Seite 5637).

Hinweis

Trotz Meldeverlust werden dem Anzeigegerät die letzten beiden Signalwechsel jedes Signals bekannt gemacht.

⚠ WARNUNG

Aufruf von "NOTIFY_8P"

Bevor Sie "NOTIFY_8P" in einem Automatisierungssystem aufrufen, müssen Sie sicherstellen, dass alle angeschlossenen Anzeigegeräte diesen Baustein kennen. Wenn Sie dies nicht beachten, wird die Kommunikation zwischen dem Automatisierungssystem und den angeschlossenen Anzeigegeräten abgebrochen. Dies hat zur Folge, dass Sie Ihre Anlage mit den angeschlossenen Anzeigegeräten nicht mehr erreichen können.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "NOTIFY_8P":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG_i, 1 ≤ i ≤ 8	Input	BOOL	E, A, M, D, L	i-tes zu überwachendes Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
EV_ID	Input	C_NOTIFY_8P	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0) EV_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von "NOTIFY_8P" mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Meldungsnummer. Die Meldungsnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Meldungsnummern gewährleistet. Die Meldungsnummer muss innerhalb Ihres Anwenderprogramms eindeutig sein.
SEVERITY	Input	WORD	E, A, M, D, L oder Konstante	Gewicht des Ereignisses Mögliche Werte: 0 bis 127 (Wert 0 bedeutet höchstes Gewicht); Defaultwert: 64 Dieser Parameter ist für die Bearbeitung der Meldung nicht relevant.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Meldungsgenerierung abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation
SD_i, 1 ≤ i ≤ 10	InOut	ANY	E, A, M, D, T, Z	i-ter Begleitwert Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren. (z.B.: P# DB10.DBX5.0 Byte 10)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "NOTIFY_8P" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Meldungsverlust: Der vorangegangene Signalwechsel bzw. die vorangegangene Meldung konnte nicht gesendet werden und wird durch die aktuelle Meldung ersetzt.
0	22	• Fehler im Zeiger auf die Begleitwerte SD_i: – bezüglich der Datenlänge oder des Datentyps – Begleitwerte im Anwenderspeicher nicht erreichbar, z. B. wegen gelöschtem DB oder Bereichslängenfehler – Die aktivierte Meldung wird ohne oder ggf. mit der gerade noch möglichen Anzahl von Begleitwerten gesendet. • Der von Ihnen gewählte Aktualparameter von SEVERITY liegt oberhalb des zulässigen Bereichs. Die aktivierte Meldung wird mit SEVERITY=127 gesendet.
0	25	Die Kommunikation wurde angestoßen. Die Meldung ist in Bearbeitung.
1	1	Kommunikationsprobleme: Verbindungsabbruch oder keine Anmeldung vorhanden
1	4	Beim Erstaufruf • liegt die angegebene EV_ID außerhalb des zulässigen Bereichs • liegt ein formaler Fehler des ANY-Pointers SD_i vor • wurde der für die CPU pro "NOTIFY_8P" maximal versendbare Speicherbereich überschritten
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "NOTIFY_8P" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben
1	18	EV_ID wurde bereits von einem der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P", "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" oder "ALARM_8P (Seite 5645)" verwendet.
1	20	Zu wenig Arbeitsspeicher vorhanden.
1	21	Die Meldung mit der angegebenen EV_ID ist gesperrt.

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_8: PLC-Meldungen ohne Begleitwerte für acht Signale erzeugen

Beschreibung

Die Anweisung ist - abgesehen von den nicht vorhandenen Begleitwerten SD_1, ..., SD_10 - identisch zur Anweisung "NOTIFY_8P (Seite 5640)".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_8":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, der bewirkt, dass der Ausgang ACK_STATE beim Aufruf aktualisiert wird (EN_R=1) bzw. nicht aktualisiert wird (EN_R=0).
SIG_i, 1≤i ≤8	Input	BOOL	E, A, M, D, L	i-tes zu überwachendes Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
EV_ID	Input	C_ALARM_8	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0) EV_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von "ALARM_8" mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Meldungsnummer. Die Meldungsnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Meldungsnummern gewährleistet. Die Meldungsnummer muss innerhalb Ihres Anwenderprogramms eindeutig sein.
SEVERITY	Input	WORD	E, A, M, D, L oder Konstante	Gewicht des Ereignisses Mögliche Werte: 0 bis 127 (Wert 0 bedeutet höchstes Gewicht) Dieser Parameter ist für die Bearbeitung der Meldung nicht relevant.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Meldungsgenerierung abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation
ACK_STATE	Output	WORD	E, A, M, D, L	Bitfeld mit dem aktuellen Quittierzustand aller acht Meldungen (1: Ereignis quittiert, 0: Ereignis nicht quittiert): • Bit 0 bis 7 ist abgebildet auf das gekommene Ereignis bei SIG_1 bis SIG_8 • Bit 8 bis 15 ist abgebildet auf das gegangene Ereignis bei SIG_1 bis SIG_8 • Initialisierungszustand: W#16#FFFF, d. h. alle kommenden und gehenden Ereignisse sind quittiert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "ALARM_8" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Meldungsverlust: Der vorangegangene Signalwechsel bzw. die vorangegangene Meldung konnte nicht gesendet werden und wird durch die aktuelle Meldung ersetzt.
0	22	Der von Ihnen gewählte Aktualparameter von SEVERITY liegt oberhalb des zulässigen Bereichs. Die aktivierte Meldung wird mit SEVERITY=127 gesendet.
0	25	Die Kommunikation wurde angestoßen. Die Meldung ist in Bearbeitung.
1	1	Kommunikationsprobleme: Verbindungsabbruch oder keine Anmeldung vorhanden Bei aktiviertem Quittierungsgetriggerten Melden: temporäre Anzeige, falls keine Anzeigegeräte das Quittierungsgetriggerte Melden beherrschen
1	4	Beim Erstaufruf liegt die angegebene EV_ID außerhalb des zulässigen Bereichs.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "ALARM_8" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben
1	18	EV_ID wurde bereits von einem der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)", "ALARM_8" oder "ALARM_8P (Seite 5645)" verwendet.
1	20	Zu wenig Arbeitsspeicher vorhanden.
1	21	Die Meldung mit der angegebenen EV_ID ist gesperrt.

Hinweis

Nach dem ersten Aufruf sind alle Bits des Ausgangs ACK_STATE gesetzt, und die Vergangenheitswerte der Eingänge SIG_i, 1 ≤ i ≤ 8, werden zu 0 angenommen.

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_8P: PLC-Meldungen mit Begleitwerten für acht Signale erzeugen

Beschreibung

Die Anweisung ist die Erweiterung von "ALARM (Seite 5650)" auf acht Signale.

Falls Sie das Verfahren des Quittierungsgetriggerten Meldens nicht aktiviert haben, wird stets eine Meldung generiert, wenn bei mindestens einem Signal ein Signalwechsel erkannt wurde (Ausnahme: Beim Erstaufruf wird stets eine Meldung gesendet.). Für alle acht Signale gibt es eine gemeinsame Meldungsnummer, die am Anzeigegerät in acht Teilmeldungen aufgeteilt wird. Sie können jede Teilmeldung einzeln quittieren oder auch mehrere Teilmeldungen auf einmal.

Über den Ausgangsparameter ACK_STATE können Sie den Quittierungszustand der einzelnen Meldungen in Ihrem Programm weiterverarbeiten. Falls Sie eine Meldung einer "ALARM_8P"-Anweisung sperren oder freigeben, so betrifft das immer die gesamte "ALARM_8P"-Anweisung. Das Sperren und Freigeben einzelner Signale ist nicht möglich.

Pro Instanz von "ALARM_8P" steht ein Meldespeicher mit zwei Speicherplätzen zur Verfügung. Genaueres zur Zwischenspeicherung von Signalwechseln siehe Abschnitt Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen (Seite 5637).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_8P":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, der bewirkt, dass der Ausgang ACK_STATE beim Aufruf aktualisiert wird (EN_R=1) bzw. nicht aktualisiert wird (EN_R=0).
SIG_i, 1≤i ≤8	Input	BOOL	E, A, M, D, L	i-tes zu überwachendes Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
EV_ID	Input	C_ALARM_8P	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0); EV_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von "ALARM_8P" mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Meldungsnummer. Die Meldungsnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Meldungsnummern gewährleistet. Die Meldungsnummer muss innerhalb Ihres Anwenderprogramms eindeutig sein.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SEVERITY	Input	WORD	E, A, M, D, L oder Konstante	Gewicht des Ereignisses Mögliche Werte: 0 bis 127 (Wert 0 bedeutet höchstes Gewicht) Dieser Parameter ist für die Bearbeitung der Meldung nicht relevant.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Meldungsgenerierung abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation
ACK_STATE	Output	WORD	E, A, M, D, L	Bitfeld mit dem aktuellen Quittierzustand aller acht Meldungen (1: Ereignis quittiert, 0: Ereignis nicht quittiert): • Bit 0 bis 7 ist abgebildet auf das gekommene Ereignis bei SIG_1 bis SIG_7 • Bit 8 bis 15 ist abgebildet auf das gegangene Ereignis bei SIG_1 bis SIG_7 • Initialisierungszustand: W#16#FFFF, d. h. alle kommenden und gehenden Ereignisse sind quittiert
SD_j, 1≤ j ≤10	InOut	ANY	E, A, M, D, T, Z	j-ter Begleitwert Die Begleitwerte gelten für alle Meldungen. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z.B.: P# DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "ALARM_8P" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Meldungsverlust: Der vorangegangene Signalwechsel bzw. die vorangegangene Meldung konnte nicht gesendet werden und wird durch die aktuelle Meldung ersetzt.
0	22	• Fehler im Zeiger auf die Begleitwerte SD_i: – bezüglich der Datenlänge oder des Datentyps – Begleitwerte im Anwenderspeicher nicht erreichbar, z. B. wegen gelöschtem DB oder Bereichslängenfehler – Die aktivierte Meldung wird ohne Begleitwerte gesendet. • Der von Ihnen gewählte Aktualparameter von SEVERITY liegt oberhalb des zulässigen Bereichs. Die aktivierte Meldung wird mit SEVERITY=127 gesendet.
0	25	Die Kommunikation wurde angestoßen. Die Meldung ist in Bearbeitung.
1	1	Kommunikationsprobleme: Verbindungsabbruch oder keine Anmeldung vorhanden. Bei aktiviertem Quittierungsgetriggerten Melden: temporäre Anzeige, falls keine Anzeigegeräte das Quittierungsgetriggerte Melden beherrschen.
1	4	Beim Erstaufruf: • liegt die angegebene EV_ID außerhalb des zulässigen Bereichs. • liegt ein formaler Fehler des ANY-Pointers SD_i vor. • wurde der für die CPU pro "ALARM_8P" maximal versendbare Speicherbereich überschritten.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde: • ein Instanz-DB, der nicht zu "ALARM_8P" gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben.
1	18	EV_ID wurde bereits von einem der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" oder "ALARM_8P" verwendet.
1	20	Zu wenig Arbeitsspeicher vorhanden.
1	21	Die Meldung mit der angegebenen EV_ID ist gesperrt.

Hinweis

Nach dem ersten Aufruf sind alle Bits des Ausgangs ACK_STATE gesetzt, und die Vergangenheitswerte der Eingänge SIG_i, 1< i < 8, werden zu "0" angenommen.

Siehe auch

Programmmeldungen anlegen (Seite 7826)

NOTIFY: Ein Signalwechsel melden

Beschreibung

Die Anweisung überwacht ein Signal. Sie generiert sowohl bei steigender Flanke (kommendes Ereignis) als auch bei fallender Flanke (gehendes Ereignis) eine Meldung, an die Sie bis zu zehn Begleitwerte anhängen können. Die Meldung wird an alle dafür angemeldeten Teilnehmer gesendet. Beim Erstaufruf wird eine Meldung mit dem aktuellen Signalzustand gesendet.

Die Begleitwerte werden zum Zeitpunkt der Flankenauswertung erfasst und der Meldung zugeordnet.

Pro Instanz der Anweisung steht ein Meldespeicher mit 2 Speicherplätzen zur Verfügung. Genaueres zur Zwischenspeicherung von Signalwechseln siehe Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen (Seite 5637).

Die Anweisung "NOTIFY" entspricht der Norm IEC 1131-5.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "NOTIFY":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L	Das zu überwachende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
EV_ID	Input	C_NOTIFY	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0) EV_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von "NOTIFY" mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Meldungsnummer. Die Meldungsnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Meldungsnummern gewährleistet. Die Meldungsnummer muss innerhalb Ihres Anwenderprogramms eindeutig sein.
SEVERITY	Input	WORD	E, A, M, D, L oder Konstante	Gewicht des Ereignisses Mögliche Werte: 0 bis 127 (Wert 0 bedeutet höchstes Gewicht) Dieser Parameter ist für die Bearbeitung der Meldung nicht relevant.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Meldungsgenerierung abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation
SD_i, 1≤i ≤10	InOut	ANY	E, A, M, D, T, Z	i-ter Begleitwert Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren. (z.B.: P# DB10.DBX5.0 Byte 10)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Meldungsverlust: Der vorangegangene Signalwechsel bzw. die vorangegangene Meldung konnte nicht gesendet werden und wird durch die aktuelle Meldung ersetzt.
0	22	• Fehler im Zeiger auf die Begleitwerte SD_i: – bezüglich der Datenlänge oder des Datentyps – Begleitwerte im Anwenderspeicher nicht erreichbar, z. B. wegen gelöschtem DB oder Bereichslängenfehler – Die aktivierte Meldung wird ohne oder ggf. mit der gerade noch möglichen Anzahl von Begleitwerten gesendet. • Der von Ihnen gewählte Aktualparameter von SEVERITY liegt oberhalb des zulässigen Bereichs. Die aktivierte Meldung wird mit SEVERITY=127 gesendet.
0	25	Die Kommunikation wurde angestoßen. Die Meldung ist in Bearbeitung.
1	1	Kommunikationsprobleme: Verbindungsabbruch oder keine Anmeldung vorhanden
1	4	Beim Erstaufruf • liegt die angegebene EV_ID außerhalb des zulässigen Bereichs • liegt ein formaler Fehler des ANY-Pointers SD_i vor • wurde der für die CPU pro "NOTIFY" maximal versendbare Speicherbereich überschritten
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde: • ein Instanz-DB, der nicht zu "NOTIFY"gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben.
1	18	EV_ID wurde bereits von einem der Anweisungen "NOTIFY", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)", "ALARM_8 (Seite 5643)" oder "ALARM_8P (Seite 5645)" verwendet.

ERROR	STATUS (dezimal)	Erläuterung
1	20	Zu wenig Arbeitsspeicher vorhanden.
1	21	Die Meldung mit der angegebenen EV_ID ist gesperrt.

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM: PLC-Meldungen mit Quittierungsanzeige erzeugen

Beschreibung

Die Anweisung überwacht ein Signal.

- Standardbetrieb (d. h. das Quittierungsgetriggerte Melden ist ausgeschaltet): Der Baustein generiert sowohl bei steigender Flanke (kommendes Ereignis) als auch bei fallender Flanke (gehendes Ereignis) eine Meldung, an die Sie bis zu zehn Begleitwerte anhängen können.
- Quittierungsgetriggertes Melden eingeschaltet: Der Baustein erzeugt nach generierter Kommend-Meldung für das Signal so lange keine weiteren Meldungen, bis Sie diese Kommend-Meldung an einem Anzeigegerät quittiert haben.
  Siehe auch: Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen (Seite 5637).

Die Meldung wird an alle dafür angemeldeten Teilnehmer gesendet.

Beim Erstaufruf wird eine Meldung mit dem aktuellen Signalzustand gesendet.

Der Ausgang ACK_UP wird rückgesetzt, wenn eine steigende Flanke vorliegt. Er wird gesetzt, wenn Ihre Quittierung des gekommenen Ereignisses von einem angemeldeten Anzeigegerät eingetroffen ist.

Analog gilt für den Ausgang ACK_DN: Er wird rückgesetzt, wenn eine fallende Flanke vorliegt. Er wird gesetzt, wenn Ihre Quittierung des gegangenen Ereignisses von einem angemeldeten Anzeigegerät eingetroffen ist. Nach dem Eintreffen Ihrer Quittierung von einem angemeldeten Anzeigegerät wird die Quittierinformation an alle dafür angemeldeten Teilnehmer weitergeleitet.

Pro Instanz der Anweisung "ALARM" steht ein Meldespeicher mit zwei Speicherplätzen zur Verfügung. Genaueres zur Zwischenspeicherung von Signalwechseln siehe im Abschnitt "Signalwechselerfassung" bei der Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen (Seite 5637).

Die Anweisung "ALARM" entspricht der Norm IEC 1131-5.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, der bewirkt, dass die Ausgänge ACK_UP und ACK_DN beim Bausteinaufruf aktualisiert werden (EN_R=1) bzw. nicht aktualisiert werden (EN_R=0). Bei EN_R=0 bleiben die Ausgangsparameter ACK_UP und ACK_DN unverändert.
SIG	Input	BOOL	E, A, M, D, L	Das zu überwachende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
EV_ID	Input	C_ALARM	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0) EV_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von "ALARM"mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Meldungsnummer. Die Meldungsnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Meldungsnummern gewährleistet. Die Meldungsnummer muss innerhalb Ihres Anwenderprogramms eindeutig sein.
SEVERITY	Input	WORD	E, A, M, D, L oder Konstante	Gewicht des Ereignisses Mögliche Werte: 0 bis 127 (Wert 0 bedeutet höchstes Gewicht) Dieser Parameter ist für die Bearbeitung der Meldung nicht relevant.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Meldungsgenerierung abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation
ACK_DN	Output	BOOL	E, A, M, D, L	Gegangenes Ereignis wurde auf einem Anzeigegerät quittiert. Initialisierungszustand: 1
ACK_UP	Output	BOOL	E, A, M, D, L	Gekommenes Ereignis wurde auf einem Anzeigegerät quittiert. Initialisierungszustand: 1
SD_i, 1≤i ≤10	InOut	ANY	E, A, M, D, T, Z	i-ter Begleitwert Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren. (z.B.: P# DB10.DBX5.0 Byte 10)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "ALARM" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Meldungsverlust: Der vorangegangene Signalwechsel bzw. die vorangegangene Meldung konnte nicht gesendet werden und wird durch die aktuelle Meldung ersetzt.
0	22	• Fehler im Zeiger auf die Begleitwerte SD_i: – bezüglich der Datenlänge oder des Datentyp – Begleitwerte im Anwenderspeicher nicht erreichbar, z. B. wegen gelöschtem DB oder Bereichslängenfehler – Die aktivierte Meldung wird ohne Begleitwerte gesendet. • Der von Ihnen gewählte Aktualparameter von SEVERITY liegt oberhalb des zulässigen Bereichs. Die aktivierte Meldung wird mit SEVERITY=127 gesendet.
0	25	Die Kommunikation wurde angestoßen. Die Meldung ist in Bearbeitung.
1	1	Kommunikationsprobleme: Verbindungsabbruch oder keine Anmeldung vorhanden Bei aktiviertem Quittierungsgetriggerten Melden: temporäre Anzeige, falls keine Anzeigegeräte das Quittierungsgetriggerte Melden beherrschen
1	4	Beim Erstaufruf • liegt die angegebene EV_ID außerhalb des zulässigen Bereichs • liegt ein formaler Fehler des ANY-Pointers SD_i vor • wurde der für die CPU pro ALARM maximal versendbare Speicherbereich überschritten
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu ALARM gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben
1	18	EV_ID wurde bereits von einem der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM", "ALARM_8 (Seite 5643)" oder "ALARM_8P (Seite 5645)" verwendet.
1	20	Zu wenig Arbeitsspeicher vorhanden.
1	21	Die Meldung mit der angegebenen EV_ID ist gesperrt.

Hinweis

Nach dem ersten Aufruf haben die Ausgänge ACK_UP und ACK_DN den Wert "1", und der Vergangenheitswert des Eingangs SIG wird zu "0" angenommen.

Siehe auch

Programmmeldungen anlegen (Seite 7826)

Anweisungen zur Erzeugung von PLC-Meldungen ohne Instanz-DB

Anlaufverhalten der Anweisungen zur Erzeugung von PLC-Meldungen

Neustart- (Warmstart-) Verhalten

Im Neustart (Warmstart) werden die Instanz-DBs der Anweisungen zur Erzeugung von PLC-Meldungen in den nicht initialisierten Zustand versetzt. Die in den Instanz-DBs gespeicherten Aktualparameter bleiben unverändert. Mit dem nächsten Bausteinaufruf werden die Parameter ID und EV_ID neu ausgewertet.

Kaltstartverhalten

Im Kaltstart werden die Inhalte der Instanz-DBs der Anweisungen zur Erzeugung von PLC-Meldungen auf die Anfangswerte gesetzt.

Wiederanlaufverhalten

Im Wiederanlauf verhalten sich die Anweisungen zur Erzeugung von PLC-Meldungen wie fortsetzbare Anwenderfunktionsbausteine. Sie laufen an der Unterbrechungsstelle weiter.

Verhalten nach Urlöschen

Urlöschen führt immer zum Abbruch aller Verbindungen, so dass kein Teilnehmer mehr für Meldungen angemeldet ist. Das Anwenderprogramm wird gelöscht. Falls Sie eine FLASH Card gesteckt haben, werden die ablaufrelevanten Programmteile von dort erneut in die CPU geladen, und die CPU führt einen Neustart (Warmstart) oder Kaltstart durch (implizit stets Kaltstart, da nach Urlöschen alle Anwenderdaten initialisiert sind).

Störverhalten der Anweisungen zur Erzeugung von PLC-Meldungen

Verbindungsabbruch

Die den Instanzen zugeordneten Verbindungen werden auf Abbruch überwacht. Bei einem Verbindungsabbruch wird der betroffene Teilnehmer aus der CPU-internen Liste der für die PLC-Meldungen angemeldeten Teilnehmer ausgetragen. Noch anstehende Meldungen für diesen Teilnehmer werden gelöscht.

Sind nach einem Verbindungsabbruch noch andere Teilnehmer angemeldet, so erhalten diese weiterhin Meldungen. Erst wenn die Verbindungen zu allen angemeldeten Teilnehmern abgebrochen sind, senden die Anweisungen nicht mehr. Dann erfolgt eine Anzeige an den Ausgangsparametern ERROR und STATUS (ERROR = 1, STATUS = 1).

Fehlerschnittstelle zum Anwenderprogramm

Tritt bei der Bearbeitung einer Anweisung zur Erzeugung von PLC-Meldungen ein Fehler auf, so wird der Ausgangsparameter ERROR auf "1" gesetzt und in den Ausgangsparameter STATUS die zugehörige Fehlerkennung eingetragen. Diese Fehlerinformationen können Sie in Ihrem Programm auswerten.

Beispiele für mögliche Fehler:

- Senden nicht möglich wegen Ressourcenmangel
- Fehler beim Zugriff auf eines der zu überwachenden Signale

Einführung in die Erzeugung von PLC-Meldungen mit Anweisungen

Anweisungen zur Erzeugung von PLC-Meldungen

Sie können mit folgenden Anweisungen eine PLC-Meldung erzeugen:

- "ALARM_SQ (Seite 5659)"
- "ALARM_S (Seite 5657)"
- "ALARM_DQ (Seite 5664)"
- "ALARM_D (Seite 5662)"

Diese Anweisungen haben folgende Eigenschaften:

- Die gesendeten PLC-Meldungen von "ALARM_SQ (Seite 5659)" und "ALARM_DQ (Seite 5664)" mit Signalzustand "1" sind von einem angemeldeten Anzeigegerät aus quittierbar. Die PLC-Meldungen von "ALARM_S" (Seite 5657) und "ALARM_D (Seite 5662)" sind immer implizit quittiert.
- Nicht ein erkannter Flankenwechsel, sondern jeder Aufruf erzeugt eine PLC-Meldung.
- Nach dem Bausteindurchlauf ist der Begleitwert SD vollständig erfasst und der PLC-Meldung zugeordnet.
  Bezüglich der Konsistenz gegenüber höherprioren Prioritätsklassen gilt für den Begleitwert: Konsistent sind
  - die einfachen Datentypen (Bit, Byte, Wort und Doppelwort)
  - ein Array des Datentyps Byte bis zu einer CPU-spezifischen Maximallänge

ALARM_SC

Mit Hilfe der Anweisung "ALARM_SC (Seite 5666)" können Sie

- den Quittierzustand der letzten "Gekommen-Meldung" und den Signalzustand beim letzten Aufruf von "ALARM_SQ (Seite 5659)" / "ALARM_DQ (Seite 5664)" bzw.
- den Signalzustand beim letzten Aufruf von "ALARM_S (Seite 5657)" / "ALARM_D (Seite 5662)" ermitteln.

Anmeldung von Anzeigegeräten

Eine Voraussetzung dafür, dass die Anweisungen zur Erzeugung von PLC-Meldungen beim Aufruf eine PLC-Meldung senden, ist, dass sich mindestens ein Anzeigegerät für PLC-Meldungen angemeldet hat.

Meldungspufferung

Damit auch bei einer hohen Belastung des Kommunikationssystems möglichst keine PLC-Meldungen verloren gehen, können die Anweisungen "ALARM_SQ (Seite 5659)", "ALARM_S (Seite 5657)", "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" jeweils zwei Meldungen zwischenspeichern.

Falls dennoch PLC-Meldungen verloren gehen, wird Ihnen dies über RET_VAL mitgeteilt. Mit der nächsten PLC-Meldung, die gesendet werden kann, erhalten die angemeldeten Anzeigegeräte darüber eine Mitteilung.

Meldungsquittierung bei "ALARM_SQ" und "ALARM_DQ"

Wenn Sie an einem Anzeigegerät eine "Gekommen-Meldung" quittiert haben, wird diese Quittierinformation zunächst an die meldungsverursachende CPU geschickt. Von dort wird die Quittierinformation an alle dafür angemeldeten Teilnehmer verteilt.

Meldungen sperren und freigeben

PLC-Meldungen, die Sie mit "ALARM_SQ (Seite 5659)" oder "ALARM_S (Seite 5657)" oder "ALARM_DQ (Seite 5664)" oder "ALARM_D (Seite 5662)" erzeugt haben, können Sie nicht sperren und anschließend wieder freigeben.

Änderungen Ihres Programms, das "ALARM_SQ" / "ALARM_S"-Aufrufe enthält

Hinweis

Beim Laden eines bereits in der CPU vorhandenen Bausteins mit "ALARM_SQ (Seite 5659)" / "ALARM_S (Seite 5657)" -Aufrufen ist es möglich, dass der bisherige Baustein eine kommende PLC-Meldung abgesetzt hat, der neue Baustein aber keine zugehörige gehende PLC-Meldung absetzt. Dadurch bleibt diese Meldung im internen Meldungsspeicher der CPU stehen. Dieser Zustand kann auch auftreten, wenn Sie Bausteine mit "ALARM_SQ (Seite 5659)" / "ALARM_S (Seite 5657)" -Aufrufen löschen.

Sie können solche PLC-Meldungen aus dem internen Meldungsspeicher der CPU entfernen, indem Sie die CPU in den Betriebszustand STOP überführen und anschließend einen Neustart (Warmstart) oder Kaltstart durchführen.

Änderungen Ihres Programms, das "ALARM_DQ" / "ALARM_D"-Aufrufe enthält

Auch wenn Ihr Programm "ALARM_DQ (Seite 5664)"- und/oder "ALARM_D (Seite 5662)"-Aufrufe enthält, können mit den oben beschriebenen Programmänderungen PLC-Meldungen im internen Meldespeicher stehen bleiben und dadurch Systemressourcen dauerhaft belegen.

Im Gegensatz zu Systemressourcen, die durch "ALARM_SQ (Seite 5659)" / "ALARM_S (Seite 5657)"-Aufrufe belegt wurden, können Systemressourcen, die durch "ALARM_DQ (Seite 5664)" / "ALARM_D (Seite 5662)"-Aufrufe belegt werden, von Ihnen wieder freigegeben werden, ohne dass Sie Ihre CPU in den Betriebszustand STOP überführen. Dies geschieht durch Einsatz der Anweisung "DEL_SI (Seite 5679)". Bevor Sie durch Aufruf von "DEL_SI" dynamisch belegte Systemressourcen freigeben, kann es sinnvoll sein, mit Hilfe der Anweisung "READ_SI" Informationen über die momentan dynamisch belegten Systemressourcen Ihrer CPU auszulesen (siehe Anweisung "READ_SI (Seite 5676)").

Anzahl übertragbarer Daten

Die Anzahl der über den Begleitwert SD der Anweisungen "ALARM_S (Seite 5657)", "ALARM_SQ (Seite 5659)", "ALARM_D (Seite 5662)" und "ALARM_DQ (Seite 5664)" übertragbaren Daten darf eine maximale Länge nicht überschreiten. Diese maximale Datenlänge berechnet sich wie folgt:

maxleng = min (pdu_lokal, pdu_remote) - 48

Dabei ist:

- pdu_lokal die maximale Länge der Datenblöcke der eigenen CPU (SZL_ID W#16#0131, INDEX 1, Variable pdu)
- pdu_remote die maximale Länge der Datenblöcke der Anzeigegeräte

Beispiel

Eine CPU 414-2 sende Meldungen an ein PG 760 (über MPI).

pdu_lokal = 480 Byte, pdu_remote = 480 Byte,

Damit gilt:

maxleng = min (480, 480) - 48 = 480 - 48 = 432

Die maximal übertragbare Datenlänge beträgt pro Anweisung also 432 Byte.

ALARM_S: Alarmmeldung generieren

Beschreibung

Die Anweisung generiert bei jedem Aufruf eine Meldung, an die Sie einen Begleitwert anhängen können. Die Meldung wird an alle dafür angemeldeten Teilnehmer gesendet. Die Anweisung stellt Ihnen also einen einfachen Meldemechanismus zur Verfügung. Sie müssen darauf achten, dass Sie die Anweisung nur dann aufrufen, wenn der Wert des meldungsauslösenden Signals SIG gegenüber dem letzten Aufruf invertiert ist. Ist dies nicht der Fall, so wird Ihnen dies über RET_VAL mitgeteilt, und es wird keine Meldung gesendet. Beim allerersten Aufruf der Anweisung müssen Sie dafür Sorge tragen, dass am Eingang SIG "1" anliegt. Sonst bekommen Sie über RET_VAL eine Fehlerinformation, und es wird keine Meldung gesendet.

Hinweis

In neu zu erstellenden Programmen sollten Sie nur noch die Anweisung "ALARM_D (Seite 5662)" verwenden (sofern Ihre CPU diese Anweisung unterstützt), da diese verbesserte Möglichkeiten zum Verwalten der Systemressourcen bieten.

Belegung von Systemressourcen

Bei der Meldungserzeugung belegt das Betriebssystem für die Dauer eines Signalzyklus eine Systemressource.

Der Signalzyklus dauert vom Aufruf mit SIG=1 bis zum erneuten Aufruf mit SIG=0. Falls innerhalb des Signalzyklus ein Überladen oder Löschen des meldungserzeugenden Bausteins erfolgt, bleibt die zugehörige Systemressource bis zum nächsten Neustart (Warmstart) belegt.

Meldungsquittierung

Meldungen, die Sie mit "ALARM_S" gesendet haben, sind immer implizit quittiert. Den Signalzustand beim letzten "ALARM_S" -Aufruf können Sie mit Hilfe von "ALARM_SC (Seite 5666)" ermitteln.

Zwischenspeicherung von Signalzuständen

Die Anweisung "ALARM_S" belegt Systemressourcen. Darin werden u. a. die letzten beiden Signalzustände inkl. Zeitstempel und Begleitwert zwischengespeichert. Erfolgt ein Aufruf von "ALARM_S" zu einem Zeitpunkt, an dem die Signalzustände der beiden letzten "gültigen" Aufrufe noch nicht gesendet sind (Signal-Overflow), werden der aktuelle und der letzte Signalzustand verworfen und eine Overflow-Kennung im Zwischenspeicher gesetzt. Zum nächstmöglichen Zeitpunkt wird das vorletzte Signal samt Overflow-Kennung gesendet.

Beispiel:

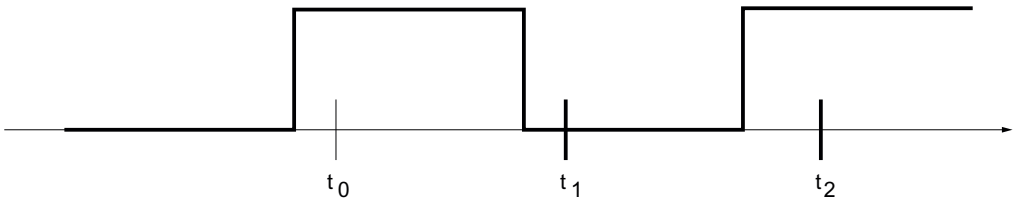

Seien t_0, t_1 und t_2 die Aufrufzeitpunkte von "ALARM_S". Falls die Signalzustände von t_0 und t_1 zum Zeitpunkt t_2 noch nicht gesendet sind, werden die Signalzustände von t_1 und t_2 verworfen, und beim Signalzustand von t_0 wird die Overflow-Kennung gesetzt.

Instanzoverflow

Falls die Anzahl von "ALARM_S"-Aufrufen größer ist als die maximale Anzahl von Systemressourcen Ihrer CPU, kann es zu einem Ressourcenengpass (Instanz-Overflow) kommen. Dies wird Ihnen sowohl durch eine Fehlerinformation in RET_VAL als auch an den angemeldeten Anzeigegeräten mitgeteilt.

Die maximale Anzahl von "ALARM_S"-Aufrufen ist CPU-abhängig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L	Das meldungsauslösende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE
EV_ID	Input	C_ALARM_S	E, A, M, D, L	Meldungsnummer (nicht erlaubt: "0")
SD	Input	ANY	E, A, M, D, T, Z	Begleitwert Maximale Länge: 12 Byte Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	• Der Begleitwert ist länger als die maximal zulässige Länge. • Der Zugriff auf den Anwenderspeicher ist nicht möglich (z. B. Zugriff auf gelöschten DB). Die Meldung wird gesendet. • Der Begleitwert zeigt auf einen Wert im Lokaldatenbereich. Die Meldung wird gesendet. (nur S7-400)

Fehlercode (W#16#...)	Erläuterung
0002	Warnung: Der letzte freie Meldequittierspeicher wurde belegt. (nur S7-400)
8081	Die angegebene EV_ID liegt außerhalb des zulässigen Bereichs.
8082	Meldungsverlust, da Ihre CPU keine Ressourcen für die Erzeugung von PLC- Meldungen durch Anweisungen mehr frei hat.
8083	Meldungsverlust, da derselbe Signalwechsel bereits vorliegt, aber noch nicht gesendet werden konnte (Signal-Overflow).
8084	Beim aktuellen und beim vorangegangenen "ALARM_S"-Aufruf hat das meldungsauslösende Signal SIG denselben Wert.
8085	Für die angegebene EV_ID liegt keine Anmeldung vor.
8086	Ein Aufruf für die angegebene EV_ID ist bereits in einer Prioritätsklasse niedrigerer Priorität in Bearbeitung.
8087	Beim ersten Aufruf von "ALARM_S" hatte das meldungsauslösende Signal den Wert "0".
8088	Die angegebene EV_ID wird bereits von einer anderen Systemressource ("ALARM_S", ALARM_DQ (Seite 5664) , ALARM_D (Seite 5662)) belegt.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_SQ: Alarmmeldung mit Quittierung generieren

Beschreibung

Die Anweisung generiert bei jedem Aufruf eine Meldung, an die Sie einen Begleitwert anhängen können. Die Meldung wird an alle dafür angemeldeten Teilnehmer gesendet. "ALARM_SQ" stellt Ihnen also einen einfachen Meldemechanismus zur Verfügung. Sie müssen darauf achten, dass Sie "ALARM_SQ" nur dann aufrufen, wenn der Wert des meldungsauslösenden Signals SIG gegenüber dem letzten Aufruf invertiert ist. Ist dies nicht der Fall, so wird Ihnen dies über RET_VAL mitgeteilt, und es wird keine Meldung gesendet. Beim allerersten Aufruf von "ALARM_SQ" müssen Sie dafür Sorge tragen, dass am Eingang SIG "1" anliegt. Sonst bekommen Sie über RET_VAL eine Fehlerinformation, und es wird keine Meldung gesendet.

Hinweis

In neu zu erstellenden Programmen sollten Sie nur noch die Anweisung "ALARM_DQ (Seite 5664)" verwenden (sofern Ihre CPU diese Anweisung unterstützt), da diese verbesserte Möglichkeiten zum Verwalten der Systemressourcen bieten.

Belegung von Systemressourcen

Bei der Meldungserzeugung mit "ALARM_SQ" belegt das Betriebssystem für die Dauer eines Signalzyklus eine Systemressource.

Der Signalzyklus dauert bei "ALARM_SQ" vom Aufruf mit SIG=1 bis zum erneuten Aufruf mit SIG=0. Zu dieser Zeitspanne kommt ggf. noch die Zeit bis zur Quittierung des kommenden Signals durch eines der angemeldeten Anzeigegeräte hinzu.

Falls innerhalb des Signalzyklus ein Überladen oder Löschen des meldungserzeugenden Bausteins erfolgt, bleibt die zugehörige Systemressource bis zum nächsten Neustart (Warmstart) belegt.

Meldungsquittierung

Sie können die von "ALARM_SQ" gesendeten Meldungen mit Signalzustand "1" an einem angemeldeten Anzeigegerät quittieren. Den Quittierzustand der letzten "Gekommen-Meldung" und den Signalzustand beim letzten "ALARM_SQ" -Aufruf können Sie mit Hilfe der Anweisung "ALARM_SC (Seite 5666)" ermitteln.

Zwischenspeicherung von Signalzuständen

"ALARM_SQ" belegt Systemressourcen. Darin werden u. a. die letzten beiden Signalzustände inkl. Zeitstempel und Begleitwert zwischengespeichert. Erfolgt ein Aufruf von "ALARM_SQ" zu einem Zeitpunkt, an dem die Signalzustände der beiden letzten "gültigen" Aufrufe noch nicht gesendet sind (Signal-Overflow), werden der aktuelle und der letzte Signalzustand verworfen und eine Overflow-Kennung im Zwischenspeicher gesetzt. Zum nächstmöglichen Zeitpunkt wird das vorletzte Signal samt Overflow-Kennung gesendet.

Beispiel:

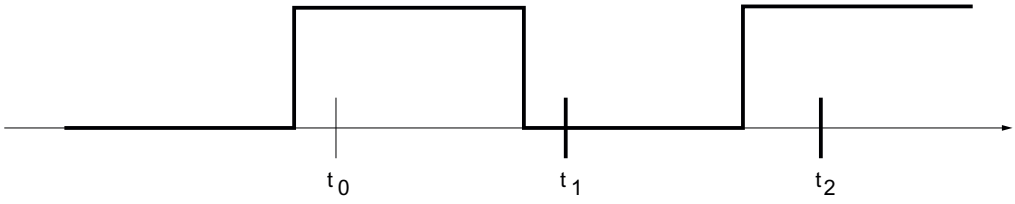

Seien t_0, t_1 und t_2 die Aufrufzeitpunkte von "ALARM_SQ". Falls die Signalzustände von t_0 und t_1 zum Zeitpunkt t_2 noch nicht gesendet sind, werden die Signalzustände von t_1 und t_2 verworfen, und beim Signalzustand von t_0 wird die Overflow-Kennung gesetzt.

Instanzoverflow

Falls die Anzahl von "ALARM_SQ"-Aufrufen größer ist als die maximale Anzahl von Systemressourcen Ihrer CPU, kann es zu einem Ressourcenengpass (Instanz-Overflow) kommen. Dies wird Ihnen sowohl durch eine Fehlerinformation in RET_VAL als auch an den angemeldeten Anzeigegeräten mitgeteilt.

Die maximale Anzahl von "ALARM_SQ"-Aufrufen ist CPU-abhängig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_SQ":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L	Das meldungsauslösende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE
EV_ID	Input	C_ALARM_S	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0)
SD	Input	ANY	E, A, M, D, T, Z	Begleitwert Maximale Länge: 12 Byte Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	• Der Begleitwert ist länger als die maximal zulässige Länge. • Der Zugriff auf den Anwenderspeicher ist nicht möglich (z. B. Zugriff auf gelöschten DB). Die Meldung wird gesendet. • Der Begleitwert zeigt auf einen Wert im Lokaldatenbereich. Die Meldung wird gesendet. (nur S7-400)
0002	Warnung: Der letzte freie Meldequittierspeicher wurde belegt. (nur S7-400)
8081	Die angegebene EV_ID liegt außerhalb des zulässigen Bereichs.
8082	Meldungsverlust, da Ihre CPU keine Ressourcen für die Erzeugung von PLC-Meldungen durch Anweisungen mehr frei hat.
8083	Meldungsverlust, da derselbe Signalwechsel bereits vorliegt, aber noch nicht gesendet werden konnte (Signal-Overflow).
8084	Beim aktuellen und beim vorangegangenen "ALARM_SQ"-Aufruf hat das meldungsauslösende Signal SIG denselben Wert.
8085	Für die angegebene EV_ID liegt keine Anmeldung vor.
8086	Ein Aufruf für die angegebene EV_ID ist bereits in einer Prioritätsklasse niedrigerer Priorität in Bearbeitung.
8087	Beim ersten Aufruf von "ALARM_SQ" hatte das meldungsauslösende Signal den Wert "0".
8088	Die angegebene EV_ID wird bereits von einer anderen Systemressource (ALARM_S (Seite 5657) , ALARM_DQ (Seite 5664) , ALARM_D (Seite 5662)) belegt.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_D: Stets quittierte PLC-Meldungen erzeugen

Beschreibung

Die Anweisung generiert bei jedem Aufruf eine PLC-Meldung, an die Sie einen Begleitwert anhängen können. Darin stimmt Sie mit der Anweisung "ALARM_S (Seite 5657)" überein.

Bei der Meldungserzeugung mit "ALARM_D" belegt das Betriebssystem für die Dauer eines Signalzyklus eine Systemressource.

Der Signalzyklus dauert bei "ALARM_D" vom Aufruf mit SIG=1 bis zum erneuten Aufruf mit SIG=0. Falls innerhalb des Signalzyklus ein Überladen oder Löschen des meldungserzeugenden Bausteins erfolgt, bleibt die zugehörige Systemressource bis zum nächsten Neustart (Warmstart) belegt.

Die zusätzliche Funktionalität von "ALARM_D" gegenüber "ALARM_S (Seite 5657)" besteht nun darin, dass Sie diese belegten Systemressourcen verwalten können.

- Mit Hilfe von READ_SI können Sie Informationen über belegte Systemressourcen auslesen.
- Mit DEL_SI können Sie belegte Systemressourcen wieder freigeben. Dies ist insbesondere bei dauerhaft belegten Systemressourcen von Bedeutung. Eine aktuell belegte Systemressource bleibt z. B. dann bis zum nächsten Neustart (Warmstart) belegt, wenn Sie bei einer Programmänderung einen FB-Aufruf löschen und dieser "ALARM_D"-Aufrufe enthält. Wenn Sie bei einer Programmänderung einen FB mit "ALARM_D"-Aufrufen erneut laden, kann es vorkommen, dass "ALARM_D" keine PLC-Meldungen mehr erzeugen.

Die Anweisung "ALARM_D" hat einen Parameter mehr als die Anweisung "ALARM_S (Seite 5657)", nämlich den Eingang CMP_ID. Mit ihm ordnen Sie die durch die Anweisung "ALARM_D" erzeugten PLC-Meldungen logischen Bereichen zu, z. B. Teilanlagen. Falls Sie die "ALARM_D" in einem FB aufrufen, bietet es sich an, "CMP_ID" mit der Nummer des zugehörigen Instanz-DB zu belegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_D":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L	Das meldungsauslösende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE
EV_ID	Input	C_ALARM_S	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMP_ID	Input	DWORD	E, A, M, D, L oder Konstante	component identifier (nicht erlaubt: 0) Kennung zur Identifikation des Teilsystems, dem die zugehörige Meldung zugeordnet ist Empfohlene Werte: • low word: 1 bis 65535 • high word: 0 Wenn Sie sich an diese Empfehlung halten, treten keine Konflikte mit von SIEMENS erstellten Programmpaketen auf.
SD	Input	ANY	E, A, M, D, T, Z	Begleitwert Maximale Länge: 12 Byte Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	• Der Begleitwert ist länger als die maximal zulässige Länge, oder • Der Zugriff auf den Anwenderspeicher ist nicht möglich (z. B. Zugriff auf gelöschten DB). Die Meldung wird gesendet. • Der Begleitwert zeigt auf einen Wert im Lokaldatenbereich. Die Meldung wird gesendet. (nur S7-400)
0002	Warnung: Der letzte freie Meldequittierspeicher wurde belegt. (nur S7-400)
8081	Die angegebene EV_ID liegt außerhalb des zulässigen Bereichs.
8082	Meldungsverlust, da Ihre CPU keine Ressourcen für die Erzeugung von PLC-Meldungen durch Anweisungen mehr frei hat.
8083	Meldungsverlust, da derselbe Signalwechsel bereits vorliegt, aber noch nicht gesendet werden konnte (Signal-Overflow).
8084	Beim aktuellen und beim vorangegangenen "ALARM_D"- Aufruf hat das meldungsauslösende Signal SIG denselben Wert.
8085	Für die angegebene EV_ID liegt keine Anmeldung vor.
8086	Ein Aufruf für die angegebene EV_ID ist bereits in einer Prioritätsklasse niedrigerer Priorität in Bearbeitung.
8087	Beim ersten Aufruf von "ALARM_D" hatte das meldungsauslösende Signal den Wert "0".
8088	Die angegebene EV_ID wird bereits von einer anderen Systemressource ("ALARM_SQ (Seite 5659)", "ALARM_S (Seite 5657)", "ALARM_DQ" (Seite 5664)) belegt.
8089	Sie haben CMP_ID den Wert "0" zugewiesen.
808A	CMP_ID passt nicht zu EV_ID
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_DQ: Quittierbare PLC-Meldungen erzeugen

Beschreibung

Die Anweisung "ALARM_DQ" generiert bei jedem Aufruf eine Meldung, an die Sie einen Begleitwert anhängen können. Darin stimmt sie mit der Anweisung "ALARM_SQ (Seite 5659)" überein.

Bei der Meldungserzeugung mit "ALARM_DQ" belegt das Betriebssystem für die Dauer eines Signalzyklus eine Systemressource.

Der Signalzyklus dauert bei "ALARM_DQ" vom Aufruf mit SIG=1 bis zum erneuten Aufruf mit SIG=0. Zu dieser Zeitspanne kommt ggf. noch die Zeit bis zur Quittierung des kommenden Signals durch eines der angemeldeten Anzeigegeräte hinzu.

Falls innerhalb des Signalzyklus ein Überladen oder Löschen des meldungserzeugenden Bausteins erfolgt, bleibt die zugehörige Systemressource bis zum nächsten Neustart (Warmstart) belegt.

Die zusätzliche Funktionalität von "ALARM_DQ" gegenüber "ALARM_SQ (Seite 5659)" besteht darin, dass Sie bei "ALARM_DQ" die belegten Systemressourcen verwalten können.

- Mit Hilfe von "READ_SI (Seite 5676)" können Sie Informationen über belegte Systemressourcen auslesen.
- Mit "DEL_SI (Seite 5679)" können Sie belegte Systemressourcen wieder freigeben. Dies ist insbesondere bei dauerhaft belegten Systemressourcen von Bedeutung. Eine aktuell belegte Systemressource bleibt z. B. dann bis zum nächsten Neustart (Warmstart) belegt, wenn Sie bei einer Programmänderung einen FB-Aufruf löschen und dieser FB "ALARM_DQ"-Aufrufe enthält. Wenn Sie bei einer Programmänderung einen FB mit "ALARM_DQ"-Aufrufen erneut laden, kann es vorkommen, dass "ALARM_DQ" keine Meldungen mehr erzeugt.

Die Anweisung "ALARM_DQ" hat einen Parameter mehr als die Anweisungen "ALARM_SQ (Seite 5659)", nämlich den Eingang CMP_ID. Mit ihm ordnen Sie die durch die Anweisungen "ALARM_DQ" erzeugten Meldungen logischen Bereichen zu, z. B. Teilanlagen. Falls Sie die "ALARM_DQ" in einem FB aufrufen, bietet es sich an, CMP_ID mit der Nummer des zugehörigen Instanz-DB zu belegen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_DQ":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SIG	Input	BOOL	E, A, M, D, L	Das meldungsauslösende Signal
ID	Input	WORD	E, A, M, D, L oder Konstante	Datenkanal für Meldungen: W#16#EEEE
EV_ID	Input	C_ALARM_S	E, A, M, D, L	Meldungsnummer (nicht erlaubt: 0)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMP_ID	Input	DWORD	E, A, M, D, L oder Konstante	component identifier (nicht erlaubt: 0); Kennung zur Identifikation des Teilsystems, dem die zugehörige Meldung zugeordnet ist Empfohlene Werte: • low word: 1 bis 65535 • high word: 0 Wenn Sie sich an diese Empfehlung halten, treten keine Konflikte mit von SIEMENS erstellten Programmpaketen auf.
SD	Input	ANY	E, A, M, D, T, Z	Begleitwert Maximale Länge: 12 Byte Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME
RET_VAL	Return	INT	E, A, M, D, L	Anzeigen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	• Der Begleitwert ist länger als die maximal zulässige Länge, oder • Der Zugriff auf den Anwenderspeicher ist nicht möglich (z. B. Zugriff auf gelöschten DB). Die Meldung wird gesendet. • Der Begleitwert zeigt auf einen Wert im Lokaldatenbereich. Die Meldung wird gesendet. (nur S7-400)
0002	Warnung: Der letzte freie Meldequittierspeicher wurde belegt. (nur S7-400)
8081	Die angegebene EV_ID liegt außerhalb des zulässigen Bereichs.
8082	Meldungsverlust, da Ihre CPU keine Ressourcen für die Erzeugung von PLC-Meldungen durch Anweisungen mehr frei hat.
8083	Meldungsverlust, da derselbe Signalwechsel bereits vorliegt, aber noch nicht gesendet werden konnte (Signal-Overflow).
8084	Beim aktuellen und beim vorangegangenen "ALARM_DQ"-Aufruf hat das meldungsauslösende Signal SIG denselben Wert.
8085	Für die angegebene EV_ID liegt keine Anmeldung vor.
8086	Ein Aufruf für die angegebene EV_ID ist bereits in einer Prioritätsklasse niedrigerer Priorität in Bearbeitung.
8087	Beim ersten Aufruf von "ALARM_DQ" hatte das meldungsauslösende Signal den Wert "0".
8088	Die angegebene EV_ID wird bereits von einer anderen Systemressource (ALARM_SQ (Seite 5659) , ALARM_S (Seite 5657) , ALARM_D (Seite 5662)) belegt.
8089	Sie haben CMP_ID den Wert "0" zugewiesen.
808A	CMP_ID passt nicht zu EV_ID
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Programmmeldungen anlegen (Seite 7826)

ALARM_SC: Quittierzustand der letzten ALARM_SQ-Gekommen-Meldung ermitteln

Beschreibung

Mit Hilfe der Anweisung können Sie:

- den Quittierzustand der letzten "ALARM_SQ (Seite 5659)" / "ALARM_DQ (Seite 5664)"-Gekommen-Meldung und den Zustand des meldungsauslösenden Signals beim letzten Aufruf von "ALARM_SQ (Seite 5659)" / "ALARM_DQ (Seite 5664)" bzw.
- den Zustand des meldungsauslösenden Signals beim letzten Aufruf von "ALARM_S (Seite 5657)" / "ALARM_D (Seite 5662)"

ermitteln. Die Meldung bzw. das Signal ist über die von Ihnen vorgegebene Meldungsnummer eindeutig referenziert, falls Sie die Meldungsnummern mit Hilfe der Meldungsprojektierung vergeben haben.

Die Anweisung greift auf den temporär belegten Speicher der Anweisungen "ALARM_SQ (Seite 5659)", "ALARM_S (Seite 5657)", "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" zu.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ALARM_SC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EV_ID	Input	C_ALARM_S	E, A, M, D, L	Meldungsnummer, zu der Sie den Signalzustand beim letzten Aufruf bzw. den Quittierzustand der letzten Gekommen-Meldung (nur bei ALARM_SQ und bei ALARM_DQ) ermitteln möchten
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
STATE	Output	BOOL	E, A, M, D, L	Zustand des meldungsauslösenden Signals beim letzten Aufruf
Q_STATE	Output	BOOL	E, A, M, D, L	• falls der angegebene Parameter EV_ID zu einem "ALARM_S" / "ALARM_D"-Aufruf gehört: "1" • falls der angegebene Parameter EV_ID zu einem "ALARM_SQ" / "ALARM_DQ"-Aufruf gehört: Quittierzustand der letzten Gekommen-Meldung: – 0: nicht quittiert – 1: quittiert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8081	Die angegebene EV_ID liegt außerhalb des zulässigen Bereichs.
8082	Zu dieser EV_ID ist momentan kein Speicherplatz belegt (Mögliche Ursache: Das zugehörige Signal hatte noch nie den Zustand "1", oder es hat bereits wieder den Zustand "0" angenommen.).
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Programmmeldungen anlegen (Seite 7826)

WR_USMSG: Anwender-Diagnoseereignis in den Diagnosepuffer schreiben

Beschreibung

Mit der Anweisung schreiben Sie ein Anwender-Diagnoseereignis in den Diagnosepuffer. Zusätzlich können Sie die zugehörige Anwender-Diagnosemeldung an alle dafür angemeldeten Teilnehmer senden (durch Belegung des Eingangsparameters SEND = TRUE). Tritt ein Fehler auf, dann gibt der Ausgangsparameter RET_VAL die Fehlerinformation aus.

Anwender-Diagnosemeldung senden

Ein Anwender-Diagnoseereignis wird mit der Anweisung in den Diagnosepuffer eingetragen. Zusätzlich können Sie die zugehörige Anwender-Diagnosemeldung auch an alle dafür angemeldeten Teilnehmer senden (durch Belegung des Eingangsparameters SEND = TRUE). Die Anwender-Diagnosemeldung wird dann in den Sendepuffer geschrieben und von dort automatisch an die dafür angemeldeten Teilnehmer gesendet.

Sie können prüfen, ob das Senden von Anwender-Diagnosemeldungen momentan möglich ist. Dazu rufen Sie die Anweisung "RDSYSST (Seite 5683)" mit den Parametern SZL_ID = W#16#0132 und INDEX = W#16#0005 auf. Das vierte Wort des dabei gelieferten Datensatzes zeigt an, ob das Senden derzeit möglich ist (1) oder nicht (0).

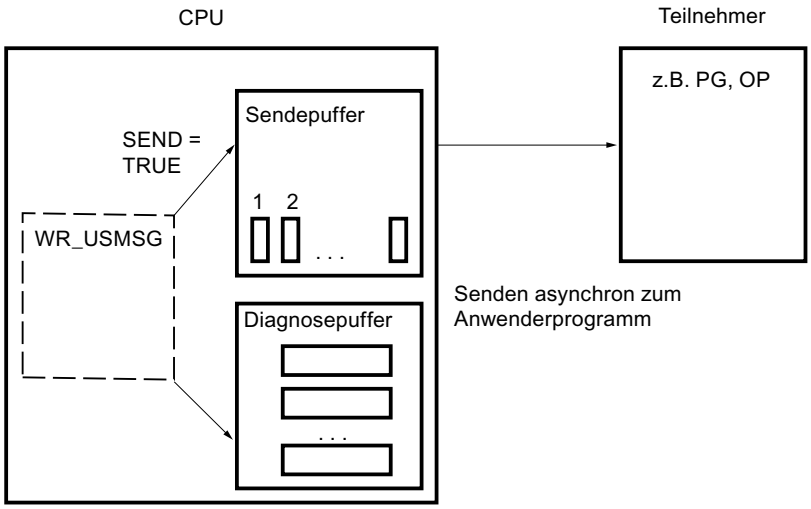

Sendepuffer voll

Der Eintrag der Anwender-Diagnosemeldung in den Sendepuffer kann nur dann erfolgen, wenn der Sendepuffer nicht voll ist. Wie viele Einträge in den Sendepuffer möglich sind, hängt vom Typ der CPU ab.

Wenn der Sendepuffer voll ist, dann

- erfolgt trotzdem der Eintrag des Diagnoseereignisses in den Diagnosepuffer
- wird im Parameter RET_VAL angezeigt, dass der Sendepuffer voll ist. (RET_VAL = W#16#8092).

Teilnehmer nicht angemeldet

Wenn eine Anwender-Diagnosemeldung gesendet werden soll (SEND = TRUE) und kein Teilnehmer angemeldet ist, dann

- erfolgt trotzdem der Eintrag des Anwender-Diagnoseereignisses in den Diagnosepuffer.
- wird im Parameter RET_VAL angezeigt, dass kein Teilnehmer angemeldet ist (W#16#0091 oder W#16#8091. Der Wert W#16#8091 erscheint bei älteren Ausgabeständen der CPU.).

Aufbau eines Eintrags

Ein Eintrag in den Diagnosepuffer ist intern wie folgt aufgebaut:

Byte	Inhalt
1 und 2	Ereignis-ID
3	Prioritätsklasse
4	OB-Nummer
5 und 6	Reserviert
7 und 8	Zusatzinformation 1
9, 10, 11 und 12	Zusatzinformation 2
13 bis 20	Zeitstempel

Ereignis-ID

Jedem Ereignis ist eine Ereignis-ID zugeordnet.

Zusatzinformation

In der Zusatzinformation sind zusätzliche Informationen zum Ereignis abgelegt. Diese Zusatzinformationen können für jedes Ereignis unterschiedlichen Inhalt besitzen. Wenn Sie ein Anwender-Diagnoseereignis erzeugen, dann können Sie den Inhalt dieser Einträge selbst bestimmen.

Wenn Sie eine Anwender-Diagnosemeldung versenden, können Sie die Zusatzinformationen als Begleitwerte in den (Ereignis-ID-spezifischen) Meldetext integrieren.

Zeitstempel

Der Zeitstempel ist vom Typ Date_and_Time.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR_USMSG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SEND	Input	BOOL	E, A, M, D, L	Senden der Anwender-Diagnosemeldung an alle angemeldeten Teilnehmer freigeben
EVENTN	Input	WORD	E, A, M, D, L oder Konstante	Ereignis-ID. Sie vergeben die Ereignis-ID. Die Vorgabe erfolgt nicht durch den Melde-server.
INFO1	Input	ANY	E, A, M, D, L	Zusatzinformation: 1 Wort lang
INFO2	Input	ANY	E, A, M, D, L	Zusatzinformation: 2 Worte lang
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter SEND

Mit SEND = TRUE wird die Anwender-Diagnosemeldung an alle dafür angemeldeten Teilnehmer gesendet. Das Senden erfolgt nur dann, wenn mindestens ein Teilnehmer angemeldet ist und der Sendepuffer nicht voll ist. Das Senden erfolgt asynchron zum Anwenderprogramm.

Parameter EVENTN

Der Parameter EVENTN enthält die Ereignis-ID des Anwenderereignisses. Sie dürfen Ereignis-IDs der Form W#16#8xyz , W#16#9xyz, W#16#Axyz, W#16#Bxyz eintragen.

Die IDs der Form W#16#8xyz und W#16#9xyz gehören zu vordefinierten Ereignissen, die IDs der Form W#16#Axyz und W#16#Bxyz zu frei definierten Ereignissen.

Ein kommendes Ereignis wird durch x = 1, ein gehendes Ereignis durch x = 0 gekennzeichnet. Für Ereignisse der Klasse A und B gilt: yz ist die in der Meldungsprojektierung für die zugehörige Meldung vergebene Meldungsnummer in hexadezimaler Darstellung.

Parameter INFO1

Der Parameter INFO1 enthält eine Information, die ein Wort lang ist. Für INFO1 sind folgende Datentypen zulässig:

- WORD
- INT
- ARRAY [0...1] OF CHAR

Sie können den Parameter INFO1 als Begleitwert in den Meldetext integrieren und damit der Meldung aktuelle Informationen hinzufügen.

Parameter INFO2

Der Parameter INFO2 enthält eine Information, die zwei Worte lang ist. Für INFO2 sind folgende Datentypen zulässig:

- DWORD
- DINT
- REAL
- TIME
- ARRAY [0...3] OF CHAR

Sie können den Parameter INFO2 als Begleitwert in den Meldetext integrieren und damit der Meldung aktuelle Informationen hinzufügen.

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	kein Fehler
0091	kein Teilnehmer angemeldet (Eintrag des Diagnoseereignisses in den Diagnosepuffer erfolgt)
8083	Datentyp INFO1 nicht zulässig
8084	Datentyp INFO2 nicht zulässig
8085	EVENTN nicht zulässig
8086	Länge von INFO1 nicht zulässig
8087	Länge von INFO2 nicht zulässig
8091	(Dieser Fehlercode erscheint nur bei älteren Ausgabeständen der CPU.) kein Teilnehmer angemeldet (Eintrag des Diagnoseereignisses in den Diagnosepuffer erfolgt)
8092	Senden derzeit nicht möglich, Sendepuffer ist voll (Eintrag des Diagnoseereignisses in den Diagnosepuffer erfolgt)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

AR_SEND: Archivdaten senden

Beschreibung

Die Anweisung sendet Archivdaten an dafür angemeldete Bedien- und Beobachtungssysteme. Diese teilen im Anmeldetelegramm der CPU die relevante Archivnummer mit. Die Archivdaten können je nach Arbeitsspeicher der CPU und verwendetem Operandenbereich bis zu 65 534 Byte groß sein. Beim Aufbau der Archivdaten müssen Sie die Vorgaben des verwendeten Bedien- und Beobachtungssystems berücksichtigen.

Die Aktivierung des Sendevorgangs erfolgt nach Aufruf des Bausteins und positiver Flanke am Steuereingang REQ. Die Anfangsadresse der zu sendenden Archivdaten wird durch SD_1 vorgegeben, die Länge des Datenblocks durch LEN. Das Senden der Daten erfolgt asynchron zur Bearbeitung des Anwenderprogramms. Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit "1" angezeigt. Bei einer positiven Flanke am Steuereingang R wird ein laufender Sendevorgang abgebrochen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AR_SEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request
R	Input	BOOL	E, A, M, D, L	Steuerparameter reset: Abbruch des aktuellen Auftrags
ID	Input	WORD	E, A, M, D oder Konstante	Datenkanal für Meldungen: W#16#EEEE ID wird nur beim Erstaufruf ausgewertet.
AR_ID	Input	C_AR_SEND	E, A, M, D, L oder Konstante	Archivnummer (nicht erlaubt: 0); AR_ID wird nur beim Erstaufruf ausgewertet. Danach gilt bei jedem Aufruf von AR_SEND mit dem zugehörigen Instanz-DB die beim Erstaufruf verwendete Archivnummer. Die Archivnummer wird automatisch vergeben. Dadurch ist die Konsistenz der Archivnummern gewährleistet.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: Sendevorgang abgeschlossen
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: ERROR=TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Genaueres siehe Parameter STATUS.
STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS: Anzeige einer Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SD_1	InOut	ANY	E, A, M, D, T, Z	Zeiger auf Archivdaten. Die Längenangabe wird nicht ausgewertet. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. Sie müssen die Archivdaten zielsystemspezifisch aufbauen. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z.B.: P# DB10.DBX5.0 Byte 10).
LEN	InOut	WORD	E, A, M, D, L	Länge des zu sendenden Datenblocks in Byte

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme
1	2	Negative Quittung, Funktion nicht ausführbar
1	3	Für die angegebene AR_ID liegt keine Anmeldung vor.
1	4	• Fehler im Archivdatenzeiger SD_1 bezüglich der Datenlänge oder des Datentyps • Beim Erstaufruf liegt die angegebene AR_ID außerhalb des zulässigen Bereichs.
1	5	Resetanforderung wurde ausgeführt
1	7	RESET-Auftrag irrelevant, da die laufende Funktion abgeschlossen oder nicht aktiviert wurde (Baustein im falschen Zustand)
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu AR_SEND gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben
1	18	AR_ID wurde bereits von einem AR_SEND verwendet.
1	20	Zu wenig Arbeitsspeicher vorhanden.

Datenkonsistenz

Um Datenkonsistenz zu gewährleisten, dürfen Sie den aktuell benutzten Teil des Sendebereichs SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

Weitere

EN_MSG: PLC-Meldungen freigeben

Beschreibung

Mit der Anweisung geben Sie gesperrte PLC-Meldungen wieder frei. Die Sperrung können Sie von einem Anzeigegerät aus oder mit "DIS_MSG (Seite 5675)" vorgenommen haben.

Die freizugebenden Meldungen legen Sie über die Eingangsparameter MODE und MESGN fest. Voraussetzung für den erfolgreichen Start eines Freigabevorgangs mit der Anweisung ist, dass momentan kein weiterer Freigabevorgang von der Anweisung aktiv ist.

Sie starten den Freigabevorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen.

Arbeitsweise

Der Vorgang des Freigebens wird asynchron durchgeführt, d. h. er kann sich über mehrere Aufrufe von der Anweisung erstrecken (siehe auch Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)):

- Beim Erstaufruf (REQ =1) prüft die Anweisung die Eingangsparameter und versucht, die benötigten Systemressourcen zu belegen. Im Gutfall wird in RET_VAL W#16#7001 eingetragen, BUSY wird gesetzt, und das Freigeben wird angestoßen.
  Andernfalls wird in RET_VAL die zugehörige Fehlerinformation eingetragen, und der Auftrag ist abgeschlossen. BUSY darf in diesem Fall nicht ausgewertet werden.
- Bei eventuellen Zwischenaufrufen wird in RET_VAL W#16#7002 eingetragen (Auftrag wird von der CPU noch bearbeitet), und BUSY wird gesetzt. Ein Zwischenaufruf beeinflusst den laufenden Auftrag nicht.
- Beim Letztaufruf wird in RET_VAL W#16#0000 eingetragen, falls kein Fehler vorlag. BUSY wird in diesem Fall mit 0 beschrieben. Im Fehlerfall wird in RET_VAL die Fehlerinformation eingetragen, und BUSY darf nicht ausgewertet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN_MSG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Freigeben anstoßen
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Parameter für die Auswahl der freizugebenden Meldungen, siehe folgende Tabelle;

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MESGN	Input	DWORD	E, A, M, D, L oder Konstante	Meldungsnummer Nur relevant für MODE = 6. Damit kann eine Einzelmeldung freigegeben werden.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation (siehe unten);
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Vorgang des Freigebens ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

In der folgenden Tabelle sind die zulässigen Werte des Eingangsparameters MODE aufgeführt.

Wert	Bedeutung
0	Alle mit Anweisungen erzeugten PLC-Meldungen der CPU
1	Alle mit Anweisungen erzeugten PLC-Meldungen der CPU, d. h. alle Meldungen von "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)", "ALARM_8P (Seite 5645)" und "ALARM_8 (Seite 5643)"
6	Einzelmeldung der Klasse "Mit Anweisungen erzeugte PLC-Meldungen"

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Das Freigeben wurde ohne Fehler beendet.
7000	Erstaufruf mit REQ = 0: Das Freigeben wurde nicht aktiviert.
7001	Erstaufruf mit REQ = 1: Das Freigeben wurde angestoßen.
7002	Zwischenaufruf: Das Freigeben ist bereits aktiv.
8081	Fehler beim Zugriff auf einen Parameter
8082	MODE hat unzulässigen Wert.
8083	Die Meldungsnummer liegt außerhalb des zulässigen Wertebereichs.
8084	Für die über MODE und ggf. MESGN spezifizierte(n) Meldung(en) liegt keine Anmeldung vor.
80C3	Das Freigeben der über MODE und ggf. MESGN spezifizierten Meldung(en) kann momentan nicht angestoßen werden, da bereits ein anderer Freigabevorgang von EN_MSG aktiv ist.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DIS_MSG: PLC-Meldungen sperren

Beschreibung

Mit der Anweisung sperren Sie PLC-Meldungen, die Sie mit Anweisungen erzeugt haben. Die zu sperrenden Meldungen legen Sie über die Eingangsparameter MODE und MESGN fest. Voraussetzung für den erfolgreichen Start eines Sperrvorgangs mit der Anweisung "DIS_MSG" ist, dass momentan kein weiterer Sperrvorgang von "DIS_MSG" aktiv ist.

Meldungen, die beim Aufruf von "DIS_MSG" bereits zum Verschicken vorbereitet sind, aber noch in einem internen Puffer stehen, sind von der Sperre nicht mehr betroffen, d. h. sie werden noch verschickt.

Dass die Meldungsübertragung gesperrt ist, wird Ihnen in den Ausgängen ERROR und STATUS der Anweisungen "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5645)", "ALARM (Seite 5650)", "ALARM_8P (Seite 5645)" und "ALARM_8 (Seite 5643)" mitgeteilt.

Sie starten den Vorgang, indem Sie beim Aufruf von "DIS_MSG" den Eingangsparameter REQ mit "1" belegen.

Arbeitsweise

Der Vorgang des Sperrens wird asynchron ausgeführt, d. h. er kann sich über mehrere Aufrufe erstrecken (siehe auch Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)):

- Beim Erstaufruf (REQ =1) prüft "DIS_MSG" die Eingangsparameter und versucht die benötigten Systemressourcen zu belegen. Im Gutfall wird in RET_VALW#16#7001 eingetragen, BUSY wird gesetzt, und das Sperren wird angestoßen.
  Andernfalls wird in RET_VAL die zugehörige Fehlerinformation eingetragen, und der Auftrag ist abgeschlossen. BUSY darf in diesem Fall nicht ausgewertet werden.
- Bei eventuellen Zwischenaufrufen wird in RET_VALW#16#7002 eingetragen (Auftrag wird von der CPU noch bearbeitet), und BUSY wird gesetzt. Ein Zwischenaufruf beeinflusst den laufenden Auftrag nicht.
- Beim letzten Aufruf wird in RET_VALW#16#0000 eingetragen, falls kein Fehler vorlag. BUSY wird in diesem Fall mit "0" beschrieben. Im Fehlerfall wird in RET_VAL die Fehlerinformation eingetragen, und BUSY darf nicht ausgewertet werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS_MSG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Sperren anstoßen
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Parameter für die Auswahl der zu sperrenden Meldungen, siehe folgende Tabelle
MESGN	Input	DWORD	E, A, M, D, L oder Konstante	Meldungsnummer Nur relevant für MODE = 5, 6, 7. Damit kann eine Einzelmeldung gesperrt werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Fehlercode (siehe unten)
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Vorgang des Sperrens ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter MODE

In der folgenden Tabelle sind die zulässigen Werte des Eingangsparameters MODE aufgeführt.

Wert	Bedeutung
0	Alle mit Anweisungen erzeugten PLC-Meldungen der CPU
1	Alle mit Anweisungen erzeugten PLC-Meldungen der CPU, d. h. alle Meldungen von "NOTIFY (Seite 5648)", "NOTIFY_8P (Seite 5640)", "ALARM (Seite 5650)", "ALARM_8P (Seite 5645)" und "ALARM_8 (Seite 5643)"
6	Einzelmeldung der Klasse "Mit Anweisungen erzeugte PLC-Meldungen"

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Das Sperren wurde ohne Fehler beendet.
7000	Erstaufruf mit REQ = 0: Das Sperren wurde nicht aktiviert.
7001	Erstaufruf mit REQ = 1: Das Sperren wurde angestoßen.
7002	Zwischenaufruf: Das Sperren ist bereits aktiv.
8081	Fehler beim Zugriff auf einen Parameter
8082	MODE hat unzulässigen Wert.
8083	Die Meldungsnummer liegt außerhalb des zulässigen Wertebereichs.
8084	Für die über MODE und ggf. MESGN spezifizierte(n) Meldung(en) liegt keine Anmeldung vor.
80C3	Das Sperren der über MODE und ggf. MESGN spezifizierten Meldung(en) kann momentan nicht angestoßen werden, da bereits ein anderer Sperrvorgang von "DIS_MSG" aktiv ist.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

READ_SI: Dynamisch belegte Systemressourcen auslesen

Entstehung dynamisch belegter Systemressourcen bei der Meldungserzeugung mit den Anweisungen "ALARM_DQ" und "ALARM_D"

Bei der Meldungserzeugung mit "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" belegt das Betriebssystem temporär Speicherplatz im Systemspeicher.

Wenn Sie z. B. einen in der CPU vorhandenen FB mit "ALARM_DQ (Seite 5664)" - oder "ALARM_D (Seite 5662)" -Aufrufen löschen, kann es vorkommen, dass die zugehörigen

Systemressourcen dauerhaft belegt bleiben. Wenn Sie den FB mit "ALARM_DQ (Seite 5664)"- oder "ALARM_D (Seite 5662)" -Aufrufen erneut laden, kann es vorkommen, dass die Anweisungen "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" nicht mehr ordnungsgemäß bearbeitet werden.

Beschreibung

Mit der Anweisung "READ_SI" können Sie momentan verwendete Systemressourcen auslesen, die bei der Meldungserzeugung mit "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" belegt wurden. Dies geschieht über die dort verwendeten Werte von EV_ID und CMP_ID. Diese werden der Anweisung "READ_SI" im Parameter SI_ID übergeben.

Die Anweisung "READ_SI" hat 4 mögliche Betriebsarten, die in der folgenden Tabelle erläutert werden. Die gewünschte Betriebsart stellen Sie über den Parameter MODE ein.

MODE	Welche der von "ALARM_DQ" / "ALARM_D"belegten Systemressourcen werden ausgelesen?
1	Alle (Aufruf von "READ_SI" erfolgt mit SI_ID:=0)
2	Diejenige Systemressource, die beim "ALARM_DQ" / "ALARM_D"-Aufruf mit EV_ID:=ev_id belegt wurde (Aufruf von READ_SI erfolgt mit SI_ID:=ev_id)
3	Alle Systemressourcen, die beim "ALARM_DQ" / "ALARM_D"-Aufruf mit CMP_ID:=cmp_id belegt wurden (Aufruf von READ_SI erfolgt mit SI_ID:=cmp_id)
0	Weitere Systemressourcen, die beim vorherigen Aufruf mit MODE=1 oder MODE=3 nicht ausgelesen werden konnten, weil Sie das Zielfeld SYS_INST zu klein gewählt hatten

Arbeitsweise

Wenn Sie beim Aufruf von READ_SI mit MODE=1 oder MODE=3 den Zielbereich SYS_INST ausreichend groß gewählt haben, enthält er nach dem Aufruf den Inhalt aller über den Parameter MODE ausgewählten Systemressourcen, die aktuell belegt sind.

Falls aktuell sehr viele Systemressourcen belegt sind, ist die Laufzeit entsprechend hoch. Bei hoher Auslastung Ihrer CPU kann dann die projektierte maximale Zyklusüberwachungszeit überschritten werden.

Dieses Laufzeitproblem können Sie wie folgt umgehen: Sie wählen den Zielbereich SYS_INST relativ klein. Falls die Anweisung nicht alle auszulesenden Systemressourcen in SYS_INST eintragen kann, wird Ihnen dies über RET_VAL=W#16#0001 mitgeteilt. Dann rufen Sie READ_SI mit MODE=0 und gleicher SI_ID wie beim vorherigen Aufruf so lange auf, bis RET_VAL den Wert W#16#0000 annimmt.

Hinweis

Da das Betriebssystem keine Koordinierung der zu einem Leseauftrag gehörenden READ_SI-Aufrufe vornimmt, sollten Sie alle READ_SI-Aufrufe in ein und derselben Prioritätsklasse durchführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_SI":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte: • 1: Alle Systemressourcen lesen • 2: Diejenige Systemressource lesen, die beim "ALARM_DQ"-/ "ALARM_D"-Aufruf mit EV_ID = ev_id belegt wurde • 3: Diejenigen Systemressourcen lesen, die beim "ALARM_DQ"- / "ALARM_D" -Aufruf mit CMP_ID = cmp_id belegt wurden • 0: Folgeaufruf
SI_ID	Input	DWORD	E, A, M, D, L oder Konstante	Kennung für die auszulesende(n) Systemressource(n) Zulässige Werte: • 0, falls MODE=1 • Meldungsnummer ev_id, falls MODE=2 • Kennung cmp_id zur Identifikation des Teilsystems, falls MODE=3
RET_VAL	Return	INT	E, A, M, D, L	Rückgabewert (Fehlerinformation bzw. Auftragszustand)
N_SI	Output	INT	E, A, M, D, L	Anzahl der in SYS_INT ausgegebenen Systemressourcen
SYS_INST	Output	ANY	D	Zielbereich für die gelesenen Systemressourcen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
0001	Es konnten nicht alle Systemressourcen gelesen werden, da Sie den Zielbereich SYS_INST zu kurz gewählt haben.
8081	(nur bei MODE=2 oder 3) Sie haben SI_ID den Wert 0 zugewiesen.
8082	(nur bei MODE=1) Sie haben SI_ID einen von 0 verschiedenen Wert zugewiesen.
8083	(nur bei MODE=0) Sie haben SI_ID einen anderen Wert zugewiesen als beim vorangegangenen Aufruf mit MODE=1 oder 3.
8084	Sie haben MODE einen unzulässigen Wert zugewiesen.
8085	READ_SI wird bereits in einem anderen OB bearbeitet.
8086	Der Zielbereich SYS_INST ist zu klein für eine Systemressource.

Fehlercode (W#16#...)	Erläuterung
8087 oder 8092	Der Zielbereich SYS_INST liegt nicht in einem DB oder Fehler im ANY-Pointer.
8xyy	allgemeine Fehlerinformation Siehe auch: ALARM_D: Stets quittierte PLC-Meldungen erzeugen (Seite 5662)

Parameter Zielbereich SYS_INST - Aufbau

Der Zielbereich für die gelesenen belegten Systemressourcen muss in einem DB liegen. Sinnvollerweise definieren Sie den Zielbereich als Feld von Strukturen, wobei eine Struktur wie folgt aufgebaut ist:

Strukturelement	Datentyp	Beschreibung
SFC_NO	WORD	Nr. der Anweisung, die die Systemressource belegt hat
LEN	BYTE	Länge der Struktur in Bytes, inkl. SFC_NO und LEN: B#16#0C
SIG_STAT	BOOL	Signalzustand
ACK_STAT	BOOL	Quittierungszustand des kommenden Ereignisses (steigende Flanke)
EV_ID	DWORD	Meldungsnummer
CMP_ID	DWORD	Kennung zur Identifikation des Teilsystems

DEL_SI: Dynamisch belegte Systemressourcen löschen

Entstehung dynamisch belegter Systemressourcen bei der Meldungserzeugung mit den Anweisungen "ALARM_DQ" und "ALARM_D"

Bei der Meldungserzeugung mit "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" belegt das Betriebssystem temporär Speicherplatz im Systemspeicher.

Wenn Sie z. B. einen in der CPU vorhandenen FB mit "ALARM_DQ (Seite 5664)" - oder "ALARM_D (Seite 5662)" -Aufrufen löschen, kann es vorkommen, dass die zugehörigen Systemressourcen dauerhaft belegt bleiben. Wenn Sie den FB mit "ALARM_DQ (Seite 5664)" -/ "ALARM_D (Seite 5662)" -Aufrufen erneut laden, kann es vorkommen, dass "ALARM_DQ (Seite 5664)" und "ALARM_D (Seite 5662)" nicht mehr ordnungsgemäß bearbeitet werden.

Beschreibung

Mit der Anweisung "DEL_SI" können Sie momentan verwendete Systemressourcen löschen.

"DEL_SI" hat drei mögliche Betriebsarten, die in der folgenden Tabelle erläutert werden. Die gewünschte Betriebsart stellen Sie über den Parameter MODE ein.

MODE	Welche der von "ALARM_DQ"/ "ALARM_D" belegten Systemressourcen werden gelöscht?
1	Alle (Aufruf von DEL_SI erfolgt mit SI_ID:=0)
2	Diejenige Systemressource, die beim "ALARM_DQ" (Seite 5664) / ALARM_D (Seite 5662) -Aufruf mit EV_ID:=ev_id belegt wurde (Aufruf von DEL_SI erfolgt mit SI_ID:=ev_id)
3	Alle Systemressourcen, die beim "ALARM_DQ" (Seite 5664) / ALARM_D (Seite 5662) -Aufruf mit CMP_ID:=cmp_id belegt wurden (Aufruf von DEL_SI erfolgt mit SI_ID:=cmp_id)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DEL_SI":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Input	INT	E, A, M, D, L oder Konstante	Auftragskennung Zulässige Werte: • 1: Alle Systemressourcen löschen • 2: Diejenige Systemressource löschen, die beim "ALARM_DQ" (Seite 5664) / "ALARM_D (Seite 5662)"-Aufruf mit EV_ID = ev_id belegt wurde • 3: Diejenigen Systemressourcen löschen, die beim "ALARM_DQ" (Seite 5664) / "ALARM_D (Seite 5662)"-Aufruf mit CMP_ID = cmp_id belegt wurden
SI_ID	Input	DWORD	E, A, M, D, L oder Konstante	Kennung für die zu löschende(n) Systemressource(n) Zulässige Werte: • 0, falls MODE=1 • Meldungsnummer ev_id, falls MODE=2 • Kennung cmp_id zur Identifikation des Teilsystems, falls MODE=3
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8081	(nur bei MODE=2 oder 3) Sie haben SI_ID den Wert "0" zugewiesen.
8082	(nur bei MODE=1) Sie haben SI_ID einen von "0" verschiedenen Wert zugewiesen.
8084	Sie haben MODE einen unzulässigen Wert zugewiesen.
8085	"DEL_SI" wird momentan bereits bearbeitet.
8086	Es konnten nicht alle ausgewählten Systemressourcen gelöscht werden, da mindestens eine zum Aufrufzeitpunkt von "DEL_SI" in Bearbeitung war.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.3.9 Diagnose

RD_SINFO: Startinformation des aktuellen OBs auslesen

Beschreibung

Mit der Anweisung lesen Sie die Startinformation:

- des zuletzt aufgerufenen OB, der noch nicht vollständig abgearbeitet wurde
- des zuletzt gestarteten Anlauf-OBs

Beide Startinformationen enthalten keinen Zeitstempel. Erfolgt der Aufruf im OB 100 oder OB 101 oder OB 102, werden zwei identische Startinformationen zurückgeliefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_SINFO".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
TOP_SI	Output	STRUCT	D, L	Startinformation des aktuellen OBs
START_UP_SI	Output	STRUCT	D, L	Startinformation des zuletzt gestarteten Anlauf-OBs

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Die Anweisung liefert keine spezifischen, sondern nur allgemeine Fehlerinformationen zurück. Die allgemeinen Fehlerinformationen und deren Auswertung sind unter dem Ausgangsparameter "RET_VAL" beschrieben.

Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171).

Parameter TOP_SI und START_UP_SI

Die Ausgangsparameter TOP_SI und START_UP_SI sind zwei identisch aufgebaute Strukturen. Deren Aufbau ist in der folgenden Tabelle wiedergegeben.

Strukturelement	Datentyp	Beschreibung
EV_CLASS	BYTE	• Bits 0 bis 3: Ereigniskennung • Bits 4 bis 7: Ereignisklasse
EV_NUM	BYTE	Ereignisnummer
PRIORITY	BYTE	Nummer der Ablaufebene (Bedeutung von B#16#FE: OB nicht vorhanden oder gesperrt oder im aktuellen Betriebszustand nicht startbar.)
NUM	BYTE	OB-Nummer
TYP2_3	BYTE	Datenkennung 2_3: kennzeichnet die in ZI2_3 eingetragene Information

Strukturelement	Datentyp	Beschreibung
TYP1	BYTE	Datenkennung 1: kennzeichnet die in ZI1 eingetragene Information
ZI1	WORD	Zusatzinformation 1
ZI2_3	DWORD	Zusatzinformation 2_3

Hinweis

Die in obiger Tabelle angegebenen Strukturelemente entsprechen inhaltlich genau den temporären Variablen eines OBs.

Bitte beachten Sie jedoch, dass die temporären Variablen in den einzelnen OBs andere Namen und andere Datentypen haben können. Beachten Sie weiterhin, dass die Aufrufschnittstelle der OBs zusätzlich Datum und Uhrzeit, zu denen der OB angefordert wurde, enthält.

Die Bits 4 bis 7 des Strukturelements EV_CLASS enthalten die Ereignisklasse. Hier sind folgende Werte möglich:

- 1: Startereignisse von Standard-OBs
- 2: Startereignisse von Synchronfehler-OBs
- 3: Startereignisse von Asynchronfehler-OBs

Das Strukturelement PRIORITY liefert die zum aktuellen OB gehörige Prioritätsklasse.

Neben diesen beiden Elementen ist noch NUM von Bedeutung. NUM enthält die Nummer des aktuellen OBs bzw. des zuletzt gestarteten Anlauf-OBs.

Beispiel

Der zuletzt aufgerufene OB, der noch nicht vollständig abgearbeitet wurde, sei der OB 80, der zuletzt gestartete Anlauf-OB sei der OB 100.

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters TOP_SI von der Anweisung "RD_SINFO" und den zugehörigen lokalen Variablen des OB 80.

TOP_SI Strukturelement	Datentyp	OB 80 - Zugehörige lokale Variable	Datentyp
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters START_UP_SI von der Anweisung "RD_SINFO" und den zugehörigen lokalen Variablen des OB 100.

START_UP_SI Strukturelement	Datentyp	OB 100 - Lokale Variable	Datentyp
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

RDSYSST: Systemzustandsliste auslesen

Beschreibung

Mit der Anweisung lesen Sie eine Teilliste der Systemzustandslisten (SZL) oder einen SZL-Teillistenauszug aus.

Sie starten den Lesevorgang, indem Sie beim Aufruf von "RDSYSST"den Eingangsparameter REQ mit "1" belegen. Falls der Lesevorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Lesevorgang noch nicht abgeschlossen.

Hinweis

Falls Sie die Anweisung "RDSYSST" im Diagnosealarm-OB mit der SZL-IDW#16#00B1 oder W#16#00B2 oder W#16#00B3 aufrufen und auf die Baugruppe zugreifen, die den Diagnosealarm gestellt hat, dann wird der Lesevorgang sofort ausgeführt.

Mit der "RDSYSST" werden nur komplette Datensätze übertragen.

Systemressourcen

Wenn Sie mehrere asynchron laufende Lesevorgänge (die Aufträge mit Parameter SZL_IDW#16#00B4 und W#16#4C91 und W#16#4092 und W#16#4292 und W#16#4692 und ggf. W#16#00B1 und W#16#00B3) kurz nacheinander anstoßen, so wird vom Betriebssystem gewährleistet, dass alle Aufträge durchgeführt werden und keine gegenseitige Beeinflussung stattfindet. Wird die Begrenzung der Systemressourcen erreicht, so erhalten Sie dies in RET_VAL mitgeteilt. Den temporären Fehlerfall können Sie durch eine Wiederholung des Auftrags beheben.

Die maximale Anzahl "gleichzeitig" aktiver Aufträge der Anweisung ist CPU-abhängig.

Hinweis

Bei S7-400-CPUs können durch die Anweisung "RDSYSST" maximal 432 Bytes in den Zielbereich übertragen werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RDSYSST":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ=1: Anstoß der Bearbeitung
SZL_ID	Input	WORD	E, A, M, D, L oder Konstante	SZL-ID der Teilliste oder des Teillistenauszugs
INDEX	Input	WORD	E, A, M, D, L oder Konstante	Typ oder Nummer eines Objekts in einer Teilliste
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, dann enthält der Parameter RET_VAL einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	TRUE: Lesevorgang noch nicht abgeschlossen
SZL_HEADER	Output	STRUCT	D, L	siehe unten
DR	Output	ANY	E, A, M, L, D	Zielbereich für die gelesene SZL-Teilliste bzw. den gelesenen SZL-Teillistenauszug: • Falls Sie nur die Kopfinformation einer SZL-Teilliste ausgelesen haben, dürfen Sie DR nicht auswerten, sondern nur SZL_HEADER. • Andernfalls gibt das Produkt aus LENTHDR und N_DR an, wie viele Bytes in DR eingetragen wurden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter SZL_HEADER

Der Parameter SZL_HEADER ist eine Struktur, die wie folgt definiert ist:

```
SZL_HEADER:   STRUCT
    LENTHDR:  WORD
    N_DR:     WORD
END_STRUCT
```

LENTHDR ist die Länge eines Datensatzes der SZL-Teilliste oder des SZL-Teillistenauszugs.

- Falls Sie nur die Kopfinformation einer SZL-Teilliste ausgelesen haben, enthält N_DR die Anzahl der vorhandenen zugehörigen Datensätze.
- Andernfalls enthält N_DR die Anzahl der in den Zielbereich übertragenen Datensätze.

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	kein Fehler
0081	Länge des Ergebnisfeldes zu klein (Es werden trotzdem so viele Datensätze wie möglich geliefert. Der SZL-Header zeigt diese Anzahl an.)
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
8081	Länge des Ergebnisfeldes zu klein (Platz reicht nicht für einen Datensatz)
8082	SZL_ID ist falsch oder in der CPU bzw. in der Anweisung unbekannt.
8083	INDEX falsch oder nicht erlaubt
8085	Die Information ist systembedingt momentan nicht verfügbar, z. B. wegen Ressourcenmangels.
8086	Datensatz ist nicht lesbar wegen eines Systemfehlers (Bus, Baugruppen, Betriebssystem).
8087	Datensatz ist nicht lesbar, weil die Baugruppe nicht vorhanden ist oder nicht quittiert.
8088	Datensatz ist nicht lesbar, weil die tatsächliche Baugruppenkennung von der Sollbaugruppenkennung abweicht.
8089	Datensatz ist nicht lesbar, weil die Baugruppe nicht diagnosefähig ist oder den Datensatz nicht unterstützt.
80A2	DP-Protokollfehler (Layer-2-Fehler) (temporärer Fehler)
80A3	DP-Protokollfehler bei User-Interface/User (temporärer Fehler)
80A4	Kommunikation am K-Bus gestört (Fehler tritt auf zwischen CPU und externer DP-Anschaltung) (temporärer Fehler)
80C5	Dezentrale Peripherie nicht verfügbar (temporärer Fehler)
80C6	Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)
80D2	Datensatz ist nicht lesbar, weil die Baugruppe nicht diagnosefähig ist.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Parameter SZL_ID

Hinweis

Auslesbare Teillisten der S7-CPUs

- Die über die "RDSYSST" auslesbaren Teillisten bei S7-300 entnehmen Sie der zugehörigen Operationsliste.
- Die über die "RDSYSST" auslesbaren Teillisten bei S7-400 sind in der nachfolgenden Tabelle angegeben.

SZL_ID (W#16#...)	Teilliste	INDEX (W#16#...)
Baugruppen-Identifikation		
0111 (Seite 5694)	ein Identifikationsdatensatz	
	Identifikation der Baugruppe	0001
	Identifikation der Basishardware	0006
	Identifikation der Basisfirmware	0007
CPU-Merkmale		
0012 (Seite 5695)	alle Merkmale	irrelevant
0112 (Seite 5695)	Merkmale einer Gruppe	
	MC7-Bearbeitungseinheit	0000
	Zeitsystem	0100
	Systemverhalten	0200
	MC7-Sprachbeschreibung	0300
	Verfügbarkeit der Anweisung "C_DIAG (Seite 5748)"	0400
0F12 (Seite 5695)	nur SZL-Teillistenkopfinformation	irrelevant
Anwenderspeicherbereiche		
0113 (Seite 5697)	ein Datensatz für den angegebenen Speicherbereich	
	Arbeitsspeicher	0001
Systembereiche		
0014 (Seite 5698)	Datensätze aller Systembereiche	irrelevant
0F14 (Seite 5698)	nur SZL-Teillistenkopfinformation	irrelevant
Bausteintypen		
0015 (Seite 5699)	Datensätze aller Bausteintypen	irrelevant
Identifikation einer Komponente		
001C (Seite 5700)	Identifikation aller Komponenten	irrelevant
011C (Seite 5700)	Identifikation einer Komponente	
	Name des Automatisierungssystems	0001
	Name der Baugruppe	0002
	Anlagenkennzeichen der Baugruppe	0003
	Urheberrechtseintrag	0004
	Seriennummer der Baugruppe	0005
	Baugruppen-Typname	0007
	Seriennummer der Memory Card	0008
	Hersteller und Profil einer CPU-Baugruppe	0009
	Ortskennzeichen einer Baugruppe	000B
0F1C (Seite 5700)	Nur SZL-Teillistenkopfinformation	irrelevant
Alarmstatus		

SZL_ID (W#16#...)	Teilliste	INDEX (W#16#...)
0222 (Seite 5706)	Datensatz zum angegebenen Alarm	OB-Nummer
Zuordnung zwischen Teilprozessabbildern und CPUs		
0025 (Seite 5709)	Zuordnung zwischen allen Teilprozessabbildern und OBs	irrelevant
0125 (Seite 5709)	Zuordnung zwischen einem Teilprozessabbild und dem zugehörigen OB	Teilprozessabbild-Nr.
0225 (Seite 5709)	Zuordnung zwischen einem OB und den zugehörigen Teilprozessabbildern	OB-Nr.
0F25 (Seite 5709)	Nur SZL-Teillistenkopfinformation	irrelevant
Kommunikationszustandsdaten		
0132 (Seite 5712)	Zustandsdaten zu einem Kommunikationsteil	
	Diagnose	0005
	Zeitsystem	0008
0232 (Seite 5712)	Zustandsdaten zu einem Kommunikationsteil	
	CPU-Schutzstufe und Bedienschalterstellungen	0004
Zustand der Baugruppen-LEDs (nicht bei allen CPUs auslesbar)		
0174 (Seite 5717)	Zustand einer LED	LED-Kennung
DP-Mastersystem-Information		
0090 (Seite 5719)	Informationen über alle der CPU bekannten DP-Mastersysteme	0000
0190 (Seite 5719)	Informationen über ein DP-Mastersystem	DP-Mastersystem-ID
0F90 (Seite 5719)	Nur SZL-Teillistenkopfinformation	0000
Baugruppenzustandsinformation (Es werden maximal 27 Datensätze geliefert.)		
0091 (Seite 5720)	Baugruppenzustandsinformation aller gesteckten Baugruppen/Submodule	irrelevant
0191 (Seite 5720)	Zustandsinformation aller nicht deaktivierten Baugruppen/Baugruppenträger mit falscher Baugruppenkennung	irrelevant
0291 (Seite 5720)	Baugruppenzustandsinformation aller gestörten und nicht deaktivierten Baugruppen	irrelevant
0391 (Seite 5720)	Baugruppenzustandsinformation aller nicht verfügbaren Baugruppen	irrelevant
0591 (Seite 5720)	Baugruppenzustandsinformation aller Submodule der Hostbaugruppe	irrelevant
0991 (Seite 5720)	Baugruppenzustandsinformation eines DP-Mastersystems	DP-Mastersystem-ID
0C91 (Seite 5720)	Baugruppenzustandsinformation einer Baugruppe im zentralen Aufbau oder an einer integrierten DP-Anschaltung oder an einer PROFINET-Anschaltung (integriert oder extern)	logische Basisadresse

SZL_ID (W#16#...)	Teilliste	INDEX (W#16#...)
4C91 (Seite 5720)	Baugruppenzustandsinformation einer Baugruppe an einer externen DP-Anschaltung	logische Basisadresse
0D91 (Seite 5720)	Baugruppenzustandsinformation aller Baugruppen im angegebenen Baugruppenträger / in der angegebenen Station (DP oder PROFINET)	Baugruppenträger oder DP-Mastersystem-ID und Stationsnummer oder Stationsnummer und die letzten beiden Stellen der PNIO-Subsystem-ID
0E91 (Seite 5720)	Baugruppenzustandsinformation aller zugeordneten Baugruppen	irrelevant
0F91 (Seite 5720)	Nur SZL-Teillistenkopfinformation	irrelevant
Baugruppenträger- / Stationszustandsinformation		
0092 (Seite 5726)	Sollzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines DP-Mastersystems.	0 / DP-Mastersystem-ID
4092 (Seite 5726)	Sollzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist.	DP-Mastersystem-ID
0192 (Seite 5726)	Aktivierungsstatus der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist.	DP-Mastersystem-ID
0292 (Seite 5726)	Istzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines DP-Mastersystems.	0 / DP-Mastersystem-ID
4292 (Seite 5726)	Istzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist.	DP-Mastersystem-ID
0392 (Seite 5726)	Zustand der Batteriepufferung eines Racks/Baugruppenträgers einer CPU, wenn mindestens eine Batterie ausgefallen ist.	0
0492 (Seite 5726)	Zustand der gesamten Batteriepufferung aller Racks/Baugruppenträger einer CPU.	0
0592 (Seite 5726)	Zustand der 24 V-Versorgung aller Racks/Baugruppenträger einer CPU.	0
0692 (Seite 5726)	Diagnose-Zustand der Erweiterungsgeräte im zentralen Aufbau/ der Stationen eines DP-Masterssystems, das über eine integrierte DP-Anschaltung angeschlossen ist.	0 / DP-Mastersystem-ID
4692 (Seite 5726)	Diagnose-Zustand der Stationen eines DP-Masterssystems, das über eine externe DP-Anschaltung angeschlossen ist.	DP-Mastersystem-ID
Baugruppenträger- / Stationszustandsinformation		
0094 (Seite 5729)	Sollzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines IO-Controllersystems.	0 / PNIO-Subsystem-ID
0194 (Seite 5729)	Aktivierungsstatus einer Station eines IO-Controllersystems, die projektiert und deaktiviert ist.	PNIO-Subsystem-ID
0294 (Seite 5729)	Istzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines IO-Controllersystems.	0 / PNIO-Subsystem-ID
0694 (Seite 5729)	Diagnosezustand der Erweiterungsgeräte im zentralen Aufbau / der Stationen eines IO-Controllersystems.	0 / PNIO-Subsystem-ID
0794 (Seite 5729)	Wartungszustand des zentralen Racks / der Stationen eines IO-Controllersystems.	0 / PNIO-Subsystem-ID

SZL_ID (W#16#...)	Teilliste	INDEX (W#16#...)
0F94 (Seite 5729)	Nur Kopfinformation	-
Erweiterte DP-Mastersystem-Information		
0195 (Seite 5731)	Erweiterte Informationen über ein DP-Mastersystem	DP-Mastersystem-ID
0F95 (Seite 5731)	Nur SZL-Teillistenkopfinformation	0000
Baugruppenzustandsinformation PROFINET IO und PROFIBUS DP		
0696 (Seite 5732)	Baugruppenzustandsinformation aller Submodule einer angegebenen Baugruppe (nur bei PROFINET IO an einer integrierten Anschaltung).	Adresse mit E/A-Kennung
0C96 (Seite 5732)	Baugruppenzustandsinformation einer Baugruppe /eines Submoduls zentral oder an einer integrierten PROFIBUS DP-Anschaltung oder an einer PROFINET-Anschaltung (integriert oder extern).	Anfangsadresse mit E/A-Kennung
Diagnosepuffer (Es werden maximal 21 Datensätze geliefert)		
00A0 (Seite 5739)	alle im aktuellen Betriebszustand lieferbaren Einträge	irrelevant
01A0 (Seite 5739)	die neuesten Einträge, die Anzahl wird über dem Index angegeben	Anzahl
0FA0 (Seite 5739)	nur SZL-Teillistenkopfinformation	irrelevant
Diagnosedaten auf Baugruppen		
00B1 (Seite 5740)	die ersten vier Diagnosebytes einer Baugruppe (DS0)	logische Basisadresse
00B2 (Seite 5742)	alle Diagnosedaten einer Baugruppe (220 byte, DS1) (keine DP-Baugruppe)	Baugruppenträger, Steckplatz
00B3 (Seite 5743)	alle Diagnosedaten einer Baugruppe (220 byte, DS1)	logische Basisadresse
00B4 (Seite 5743)	Diagnosedaten eines DP-Slaves	projektierte Diagnoseadresse

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

Systemzustandsliste

Überblick über die Systemzustandsliste (SZL)

Definition: Systemzustandsliste

Die Systemzustandsliste beschreibt den aktuellen Zustand eines Automatisierungssystems. Der Inhalt der SZL (Systemzustandsliste) kann durch Auskunftsfunktionen nur gelesen, nicht aber geändert werden. Die Teillisten sind virtuelle Listen, das heißt, sie werden vom Betriebssystem der Zentralbaugruppen nur auf Anforderung zusammengestellt.

Sie können immer nur eine Teilliste über "RDSYSST (Seite 5683)" auslesen. Hier sind die Teillisten der Systemzustandliste beschrieben, die mittels "RDSYSST (Seite 5683)" im Anwenderprogramm ausgelesen werden können und deren Auskünfte sich auf

- CPUs beziehen oder
- auf solche Baugruppen, deren Teillisten nicht baugruppenspezifisch sind (z. B. SZL-IDs W#16#00B1, W#16#00B2, W#16#00B3).

Baugruppenspezifische Teillisten z. B. für CPs und FMs entnehmen Sie bitte der jeweiligen Baugruppenbeschreibung.

Inhalt

Die Systemzustandsliste enthält Informationen über:

- Systemdaten
- Baugruppenzustandsinformation in der CPU
- Diagnosedaten auf Baugruppen
- Diagnosepuffer

Systemdaten

Systemdaten sind feste oder parametrierte Kenndaten einer CPU. Sie beschreiben folgende Leistungsmerkmale:

- Ausbau der CPU
- Zustand der Prioritätsklassen
- Kommunikation.

Baugruppenzustandsinformation

Die Baugruppenzustandsinformation beschreibt den aktuellen Zustand der Komponenten, die durch die Systemdiagnose überwacht werden.

Diagnosedaten auf Baugruppen

Die einer CPU zugeordneten diagnosefähigen Baugruppen besitzen Diagnosedaten, die auf den Baugruppen selbst abgelegt sind.

Diagnosepuffer

Der Diagnosepuffer enthält Diagnoseeinträge in der Reihenfolge ihres Auftretens.

Aufbau einer SZL-Teilliste

Grundlagen

Sie können eine Teilliste oder einen Teillistenauszug mit "RDSYSST (Seite 5683)" auslesen. Dabei legen Sie über die Parameter SZL_ID (Systemzustandsliste ID) und INDEX fest, was Sie auslesen wollen.

Aufbau

Eine Teilliste besteht aus

- einem Kopf und
- den Datensätzen.

Kopf

Der Kopf einer Teilliste besteht aus:

- SZL-ID
- INDEX
- Länge eines Datensatzes dieser Teilliste in Byte
- Anzahl der Datensätze, die diese Teilliste enthält.

Index

Für bestimmte Teillisten bzw. Teillistenauszüge ist die Angabe einer Objekttypkennung bzw. einer Objektnummer notwendig. Dazu wird der Index benutzt. Wird er bei einer Auskunft nicht benötigt, so ist sein Inhalt irrelevant.

Datensätze

Ein Datensatz einer Teilliste hat eine bestimmte Länge. Sie hängt davon ab, welche Informationen in der Teilliste abgelegt sind. Wie die Datenworte in einem Datensatz belegt sind, hängt ebenfalls von der Teilliste ab.

SZL-ID

SZL-ID (Systemzustandsliste ID)

Jede Teilliste innerhalb der Systemzustandsliste besitzt eine Nummer. Die Ausgabe einer Teilliste kann komplett oder partiell angefordert werden. Die möglichen Teillistenauszüge sind fest definiert und werden ihrerseits durch eine Nummer gekennzeichnet. Die SZL-ID setzt sich zusammen aus der Nummer der Teilliste und der Nummer des Teillistenauszugs und der Baugruppenklasse.

Aufbau

Die SZL-ID ist ein Wort lang. Sie ist wie folgt zusammengesetzt:

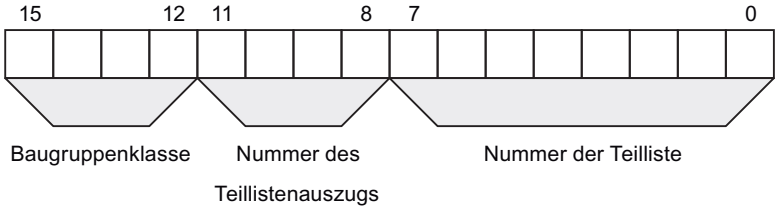

Bild 4-23 Aufbau der SZL-ID

Baugruppenklasse

Beispiele für Baugruppenklassen:

Baugruppenklasse	zugehörige Verschlüsselung (binär)
CPU	0000
IM	0100
FM	1000
CP	1100

Nummer des Teillistenauszugs

Die Nummern der Teillistenauszüge und ihre Bedeutung sind abhängig von der jeweiligen Teilliste. Über die Nummer des Teillistenauszugs geben Sie an, welche Untermenge einer Teilliste Sie lesen möchten.

Nummer der Teilliste

Über die Nummer der Teilliste geben Sie an, welche Teilliste der Systemzustandsliste Sie auslesen möchten.

Mögliche SZL-Teillisten

Subset

In einer Baugruppe ist immer nur eine Untermenge aller möglichen Teillisten verfügbar. Welche Teillisten verfügbar sind, hängt von der Baugruppe ab.

Mögliche SZL-Teillisten

In der folgenden Tabelle sind alle möglichen Teillisten mit zugehöriger Nummer in der SZL-ID (Systemzustandsliste ID) dargestellt.

Teilliste	SZL-ID
Baugruppen-Identifikation	W#16#xy11 (Seite 5694)
CPU-Merkmale	W#16#xy12 (Seite 5695)
Anwenderspeicherbereiche	W#16#xy13 (Seite 5697)
Systembereiche	W#16#xy14 (Seite 5698)
Bausteintypen	W#16#xy15 (Seite 5699)
Identifikation einer Komponente	W#16#xy1C (Seite 5700)
Alarmstatus	W#16#xy22 (Seite 5706)
Zuordnung zwischen Teilprozessabbildern und OBs	W#16#xy25 (Seite 5709)
Kommunikation: Zustandsdaten	W#16#xy32 (Seite 5712)
Zustand der Baugruppen-LEDs	W#16#xy74 (Seite 5717)
DP-Mastersystem-Information	W#16#xy90 (Seite 5719)
Baugruppenzustandsinfo	W#16#xy91 (Seite 5720)
Baugruppenträger- /Stationszustandsinformation	W#16#xy92 (Seite 5726)
Baugruppenträger- /Stationszustandsinformation	W#16#xy94 (Seite 5729)
Erweiterte DP-Mastersystem-Information	W#16#xy95 (Seite 5731)
Baugruppenzustandsinformation PROFINET IO und PROFIBUS DP	W#16#xy96 (Seite 5732)
Diagnosepuffer	W#16#xyA0 (Seite 5739)
Baugruppendiagnoseinfo (DS 0)	W#16#00B1 (Seite 5740)

Teilliste	SZL-ID
Baugruppendiagnoseinfo (DS 1), physikalische Adresse	W#16#00B2 (Seite 5742)
Baugruppendiagnoseinfo (DS 1), logische Adresse	W#16#00B3 (Seite 5743)
Diagnosedaten eines DP-Slaves	W#16#00B4 (Seite 5743)

SZL-ID W#16#xy11 - Baugruppen-Identifikation

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy11 erhalten Sie die Baugruppen-Identifikation dieser Baugruppe.

Kopf

Der Kopf der SZL mit der SZL-IDW#16#xy11 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0111: ein einzelner Identifikationsdatensatz
INDEX	Nummer eines bestimmten Datensatzes W#16#0001: Identifikation der Baugruppe W#16#0006: Identifikatiion der Basishardware W#16#0007: Identifikation der Basisfirmware
LENTHDR	W#16#001C: Ein Datensatz ist 14 Worte lang (28 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der SZL mit der SZL-IDW#16#xy11 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	Nummer eines Identifikations-Datensatzes
MRPC	20 Byte	Machine-Readable Product Code Bei INDEX W#16#0007: reserviert Bei INDEXW#16#0001 und W#16#0006: MLFB der Baugruppe; String aus 19 Zeichen und einem Blank (20H); z.B. für CPU 314: "6ES7 314-0AE01-0AB0 "
ModuleType	1 Wort	reserviert

Name	Länge	Bedeutung
OutMod1	1 Wort	Bei INDEXW#16#0001: Ausgabestand der Baugruppe Bei INDEXW#16#0006 und W#16#0007: "V" und erste Ziffer der Versionskennung
OutMod2	1 Wort	Bei INDEXW#16#0001: reserviert Bei INDEXW#16#0006 und W#16#0007: restliche Ziffern der Versionskennung

SZL-ID W#16#xy12 - CPU-Merkmale

Zweck

Baugruppen vom Baugruppentyp CPU besitzen eine Reihe von Merkmalen mit starkem Hardwarebezug. Jedem dieser Merkmale ist eine Kennung zugeordnet. Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy12 erhalten Sie die Merkmale der Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy12 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs: W#16#0012: alle Merkmale W#16#0112: Merkmale einer Gruppe; Die Gruppe geben Sie im Parameter INDEX an. W#16#0F12: nur SZL-Teillistenkopfinfo
INDEX	Gruppe W#16#0000: MC7-Bearbeitungseinheit W#16#0100: Zeitsystem W#16#0200: Systemverhalten W#16#0300: MC7-Sprachbeschreibung der CPU W#16#0400: Verfügbarkeit der Anweisung "C_DIAG (Seite 5748)"
LENTHDR	W#16#0002: Ein Datensatz ist ein Wort lang (2 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy12 ist ein Wort lang. Für jedes vorhandene Merkmal ist eine Merkmalkennung abgelegt. Eine Merkmalkennung ist ein Wort lang.

Hinweis

Alle für Ihre CPU relevanten Datensätze werden ausgegeben. Sie folgen lückenlos aufeinander.

Merkmalkennung

In der folgenden Tabelle sind alle Merkmalkennungen aufgeführt.

Kennung	Bedeutung
W#16#0000 - 00FF	MC7-Bearbeitungseinheit (Gruppe mit Index 0000)
W#16#0001	Codegenerierende MC7-Bearbeitung
W#16#0002	MC7-Interpreter
W#16#0100 - 01FF	Zeitsystem (Gruppe mit Index 0100)
W#16#0101	1-ms-Zeitauflösung
W#16#0102	10ms-Zeitauflösung
W#16#0103	keine Echtzeituhr
W#16#0104	BCD-Uhrzeitformat
W#16#0105	gesamte Uhrzeitfunktionalität (Uhrzeit stellen, Uhrzeitstatus setzen und lesen, Uhrzeitsynchronisation: Uhrzeitslave und Uhrzeitmaster)
W#16#0106	OB_RT ist verfügbar
W#16#0200 - 02FF	Systemverhalten (Gruppe mit Index 0200)
W#16#0201	Mehrprozessorfähig
W#16#0202	Kaltstart, Neustart (Warmstart) und Wiederanlauf verfügbar
W#16#0203	Kaltstart und Neustart (Warmstart) verfügbar
W#16#0204	Neustart (Warmstart) und Wiederanlauf verfügbar
W#16#0205	Nur Neustart (Warmstart) verfügbar
W#16#0206	Umkonfigurieren im RUN für Dezentrale Peripherie unter Nutzung von vordefinierten Reserven möglich
W#16#0208	Berücksichtigung von Motion Control-Funktionalität
W#16#0300 - 03FF	MC7-Sprachbeschreibung der CPU (Gruppe mit Index 0300)
W#16#0301	belegt
W#16#0302	Alle 32-Bit-Festpunktbefehle
W#16#0303	Alle Gleitpunktrechenbefehle
W#16#0304	sin, asin, cos, acos, tan, atan, sqr, sqrt, ln, exp
W#16#0305	Akku3/Akku4 mit dazugehörigen Befehlen (ENT, PUSH, POP, LEAVE)
W#16#0306	Master Control Relay- Befehle
W#16#0307	Adressregister 1 vorhanden mit den dazugehörigen Befehlen
W#16#0308	Adressregister 2 vorhanden mit den dazugehörigen Befehlen
W#16#0309	Befehle zur bereichsübergreifenden Adressierung
W#16#030A	Befehle zur bereichsinternen Adressierung
W#16#030B	Alle speicherindirekt adressierenden Befehle über M
W#16#030C	Alle speicherindirekt adressierenden Befehle über DB
W#16#030D	Alle speicherindirekt adressierenden Befehle über DI
W#16#030E	Alle speicherindirekt adressierenden Befehle über LD
W#16#030F	Alle Befehle zur Parameterübergabe in Anweisungen
W#16#0310	Flankenmerkerbefehle über E
W#16#0311	Flankenmerkerbefehle über A
W#16#0312	Flankenmerkerbefehle über M
W#16#0313	Flankenmerkerbefehle über DB

Kennung	Bedeutung
W#16#0314	Flankenmerkerbefehle über DI
W#16#0315	Flankenmerkerbefehle über LD
W#16#0316	Dynamische Auswertung des ERAB-Bits
W#16#0317	Dynamischer Lokaldatenbereich mit den zugehörigen Befehlen
W#16#0318	belegt
W#16#0319	belegt
W#16#0401	belegt
W#16#0402	reserviert
W#16#0403	"C_DIAG (Seite 5748)" ist verfügbar

SZL-ID W#16#xy13 - Anwenderspeicherbereiche

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy13 erhalten Sie Informationen über die Speicherbereiche der Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy13 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0113: Datensatz für einen Speicherbereich Den Speicherbereich geben Sie über den Parameter INDEX an.
INDEX	Angabe eines Speicherbereiches (nur bei SZL-IDW#16#0113) W#16#0001: Arbeitsspeicher
LENTHDR	W#16#0024: Ein Datensatz ist 18 Worte lang (36 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy13 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	Index eines Speicherbereichs W#16#0001: Arbeitsspeicher
Code	1 Wort	Speichertyp: • W#16#0001: flüchtiger Speicher (RAM) • W#16#0002: Nicht flüchtiger Speicher (FEPROM) • W#16#0003: gemischter Speicher (RAM und FEPROM)
Size	2 Worte	Gesamtgröße des selektierten Speichers (Summe von Area1 und Area2)

Name	Länge	Bedeutung
Mode	1 Wort	logischer Modus des Speichers • Bit 0: flüchtiger Speicherbereich • Bit 1: nichtflüchtiger Speicherbereich • Bit 2: gemischter Speicherbereich Für Arbeitsspeicher: • Bit 3: Code und Daten getrennt • Bit 4: Code und Daten gemeinsam
Granularity	1 Wort	stets mit "0" belegt
Area1	2 Worte	Größe des flüchtigen Speicherbereichs in Byte
Reserved1	2 Worte	Größe des belegten flüchtigen Speicherbereichs
Block1	2 Worte	Größter freier Block im flüchtigen Speicherbereich Falls "0": keine Information vorhanden oder Information nicht ermittelbar.
Area2	2 Worte	Größe des nichtflüchtigen Speicherbereichs in Byte
Reserved2	2 Worte	Größe des belegten nichtflüchtigen Speicherbereichs
Block2	2 Worte	Größter freier Block im nichtflüchtigen Speicherbereich Falls "0": keine Information vorhanden oder Information nicht ermittelbar.

SZL-ID W#16#xy14 - Systembereiche

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy14 erhalten Sie Informationen über die Systembereiche der Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy14 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0014: alle Systembereiche einer Baugruppe W#16#0F14: nur SZL-Teillistenkopfinfo
INDEX	irrelevant
LENTHDR	W#16#0008: Ein Datensatz ist 4 Worte lang (8 Byte)
N_DR	Anzahl der Datensätze Achten Sie darauf, dass Sie die Anzahl der Datensätze mindestens mit 9 parametrieren. Wenn Sie nämlich den Zielbereich zu klein wählen, liefert Ihnen "RDSYSST (Seite 5683)" keinen Datensatz.

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy14 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	Index des Systembereichs • W#16#0001: PAE (Anzahl in Byte) • W#16#0002: PAA (Anzahl in Byte) • W#16#0003: Merker (Anzahl in Bit) **Hinweis:** Dieser Index wird nur von CPU geliefert, deren Merkeranzahl in einem Wort dargestellt werden kann. Falls Ihre CPU diesen Wert nicht liefert, werten Sie den Index W#16#0008 aus. • W#16#0004: Zeiten (Anzahl) • W#16#0005: Zähler (Anzahl) • W#16#0006: Anzahl der Byte im logischen Adressraum • W#16#0007: Lokaldaten (gesamter Lokaldatenbereich der CPU in Byte) **Hinweis:** Dieser Index wird nur von CPUs geliefert, bei denen die Länge des gesamten Lokaldatenbereichs in einem Wort dargestellt werden kann. Falls Ihre CPU diesen Wert nicht liefert, werten Sie bitte den Index W#16#0009 aus. • W#16#0008: Merker (Anzahl in Byte) • W#16#0009: Lokaldaten (gesamter Lokaldatenbereich der CPU in KByte)
code	1 Wort	Speichertyp • W#16#0001: flüchtiger Speicher (RAM) • W#16#0002: Nicht flüchtiger Speicher (FEPROM) • W#16#0003: gemischter Speicher (RAM und FEPROM)
number	1 Wort	Anzahl der Elemente des Systembereichs
retentive	1 Wort	Anzahl der remanenten Elemente

SZL-ID W#16#xy15 - Bausteintypen

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy15 erhalten Sie die Bausteintypen, die in einer Baugruppe vorhanden sind.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy15 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0015: Datensätze aller Bausteintypen einer Baugruppe
INDEX	irrelevant

Inhalt	Bedeutung
LENTHDR	W#16#000A: Ein Datensatz ist 5 Worte lang (10 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy15 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	Bausteintyp-Nummer • W#16#0800: OB • W#16#0A00: DB • W#16#0B00: SDB • W#16#0C00: FC • W#16#0E00: FB
MaxNumber	1 Wort	maximale Anzahl der Bausteine des Typs • bei OBs: max. mögliche Anzahl OBs einer CPU • bei DBs: max. mögliche Anzahl DBs einschließlich DB0 • bei SDBs: max. mögliche Anzahl SDBs einschließlich SDB2 • bei FCs und FBs: max. mögliche Anzahl ladbarer Bausteine
MaxLenght	1 Wort	maximale Gesamtgröße des Ladeobjekts in Kbyte
MaxWorkMem	2 Worte	maximale Länge des Arbeitsspeicheranteils eines Bausteins in Byte

SZL-ID W#16#xy1C - Identifikation einer Komponente

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy1C können Sie die CPU bzw. das Automatisierungssystem identifizieren.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy1C ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs
	• W#16#001C: Identifikation aller Komponenten • W#16#011C: Identifikation einer Komponente • W#16#0F1C: Nur SZL-Teillistenkopfinformation

Inhalt	Bedeutung
INDEX	• Kennung der Komponente für den Teillistenauszug mit der SZL-IDW#16#011C und W#16#031C – W#16#0001: Name des Automatisierungssystems – W#16#0002: Name der Baugruppe – W#16#0003: Anlagenkennzeichen der Baugruppe – W#16#0004: Urheberrechtseintrag – W#16#0005: Seriennummer der Baugruppe – W#16#0007: Baugruppen-Typname – W#16#0008: Seriennummer der Memory Card Bei Baugruppen, die keine Möglichkeit haben, eine Memory Card zu stecken, wird kein Datensatz geliefert. – W#16#0009: Hersteller und Profil einer CPU-Baugruppe – W#16#000A: OEM-Kennung einer Baugruppe (nur S7-300) – W#16#000B: Ortskennzeichen einer Baugruppe – W#16#000C: Seriennummer von Sync-Modul 1 – W#16#000D: Seriennummer von Sync-Modul 2 • Rack-Nr. für den Teillistenauszug mit der SZL-IDW#16#021C (Byte0: Rack-Nr., Byte1: B#16#00)
LENTHDR	W#16#0022: Ein Datensatz ist 17 Worte lang (34 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy1C hat folgenden Aufbau:

INDEX = W#16#0001

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0001 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#01
name	12 Worte	Name des Automatisierungssystems (maximal 24 Zeichen; bei kürzerem Namen wird mit B#16#00 aufgefüllt)
res	4 Worte	reserviert

INDEX = W#16#0002

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0002 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#02
name	12 Worte	Name der Baugruppe (maximal 24 Zeichen; bei kürzerem Namen wird mit B#16#00 aufgefüllt)
res	4 Worte	reserviert

INDEX = W#16#0003

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0003 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#03
tag	16 Worte	Anlagenkennzeichen der Baugruppe (maximal 32 Zeichen; bei kürzerem Anlagenkennzeichen wird mit B#16#00 aufgefüllt)

INDEX = W#16#0004

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0004 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#04
copyright	13 Worte	Konstante Zeichenfolge "Original Siemens Equipment"
res	3 Worte	reserviert

INDEX = W#16#0005

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0005 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#05
serialn	12 Worte	Seriennummer der Baugruppe als Zeichenfolge mit max. 24 Zeichen. Bei kürzerer Nummer wird mit B#16#00 aufgefüllt. Hinweis: Die Seriennummer ist weltweit eindeutig für SIMATIC-Komponenten. Sie ist fest an die CPU-Hardware gebunden, d. h. sie bleibt bei einem Firmware-Update unverändert.
res	4 Worte	reserviert

INDEX = W#16#0007

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0007 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#07
cputypname	16 Worte	Baugruppen-Typname als Zeichenfolge mit max. 32 Zeichen. Bei kürzerem Namen wird mit B#16#00 aufgefüllt.

INDEX = W#16#0008

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0008 • bei den Teillistenauszügen W#16#011C, W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#08
sn_mc/mmc	16 Worte	Seriennummer der Memory Card/Micro Memory Card als Zeichenfolge mit max. 32 Zeichen. Bei kürzerer Nummer wird mit B#16#00 aufgefüllt. • bei einer Siemens-Seriennummer: Seriennummer ohne Zusatz • bei einer Produktseriennummer (PSN) einer S7-Micro Memory Card: "MMC" plus Seriennummer (PSN) • bei einer Seriennummer einer S7-Memory Card: "MC" plus Seriennummer Falls keine Memory Card gesteckt ist, endet die Zeichenfolge unmittelbar nach "MMC" bzw. "MC".

INDEX = W#16#0009

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#0009 • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte 0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#09
manufacturer_id	1 Wort	siehe PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
profile_id	1 Wort	siehe PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
profile_specific_typ	1 Wort	siehe PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
res	13 Worte	reserviert

INDEX = W#16#000A

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#000A • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#0A
oem_copy-right_string	13 Worte	OEM-Copyright-Kennung als Zeichenfolge mit maximal 20 Zeichen. Bei kürzerem Namen ist mit B#16#00 aufgefüllt.
oem_id	1 Wort	OEM-ID. Sie wird von Siemens vergeben.
oem_add_id	2 Worte	OEM-Zusatzkennung. Sie ist vom Anwender vorgebbar.

INDEX = W#16#000B

Name	Länge	Bedeutung
index	1 Wort	• bei einer Standard-CPU und dem Teillistenauszug W#16#011C: Kennung der Komponente: W#16#000B • bei den Teillistenauszügen W#16#021C und W#16#031C: Byte0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 Byte 1: Kennung der Komponente: B#16#0B
loc_id	16 Worte	Ortskennzeichen als Zeichenfolge mit maximal 32 Zeichen. Bei kürzerem Namen ist mit B#16#00 aufgefüllt.

INDEX = W#16#000C

Name	Länge	Bedeutung
index	1 Wort	• Byte0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 • Byte 1: Kennung der Komponente: B#16#0C
mlfb	5 Worte	10. bis 19. Zeichen der Bestellnummer von Sync-Modul 1
fill	1 Wort	vier Leerzeichen (ASCII)
es	1 Wort	Erzeugnisstand (ASCII), z. B. "01"
fill	1 Byte	zwei Leerzeichen (ASCII)
vendor_sn	17 Byte	Seriennummer

Hinweis

Wenn das Sync-Modul 1 nicht vorhanden oder nicht identifizierbar ist, wird der Datensatz nach der Variable "index" mit B#16#00 aufgefüllt.

INDEX = W#16#000D

Name	Länge	Bedeutung
index	1 Wort	• Byte0: – Bits 0 bis 2: Rack-Nr.; – Bit 3: 0 = Reserve-CPU, 1 = Master-CPU; – Bits 4 bis 7: 1111 • Byte 1: Kennung der Komponente: B#16#0D
mlfb	5 Worte	10. bis 19. Zeichen der Bestellnummer von Sync-Modul 2
fill	1 Wort	vier Leerzeichen (ASCII)
es	1 Wort	Erzeugnisstand (ASCII), z. B. "01"
fill	1 Byte	zwei Leerzeichen (ASCII)
vendor_sn	17 Byte	Seriennummer

Hinweis

Wenn das Sync-Modul 2 nicht vorhanden oder nicht identifizierbar ist, wird der Datensatz nach der Variable "index" mit B#16#00 aufgefüllt.

SZL-ID W#16#xy22 - Alarmstatus

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy22 erhalten Sie Informationen über den aktuellen Zustand der Alarmbearbeitung und der Alarmgenerierung der Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy22 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0222 Datensatz zum angegebenen Alarm. Den Alarm (OB-Nr.) geben Sie im Parameter INDEX an.
INDEX	OB-Nr. bzw. Alarmklasse (bei SZL-IDW#16#0222) W#16#0000: freier Zyklus W#16#000A: Uhrzeitalarm W#16#0014: Verzögerungsalarm W#16#001E: Weckalarm W#16#0028: Prozessalarm W#16#0032: DP-Alarm W#16#003C: Taktsynchronalarm W#16#0050: Asynchroner Fehleralarm W#16#005A: Hintergrund W#16#0064: Anlauf W#16#0078: Synchroner Fehleralarm
LENTHDR	W#16#001C: Ein Datensatz ist 14 Worte lang (28 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#xy22 hat folgenden Aufbau:

Name	Länge	Bedeutung
Info	10 Worte	Startinfo des jeweiligen OBs, mit folgenden Ausnahmen: • Beim OB 1 sind die aktuelle minimale (in Byte 8 und 9) und maximale Zykluszeit (in Byte 10 und 11) zu entnehmen (Zeitbasis: ms, Bytezählung bei 0 beginnend). • Während ein Auftrag für einen Verzögerungsalarm aktiv ist, enthalten die Bytes 8 bis 11 (Bytezählung bei 0 beginnend) die von der parametrierten Verzögerungszeit noch verbleibende Restzeit in ms. • Beim OB 80 sind die projektierte minimale (in Byte 8 und 9) und maximale Zykluszeit (in Byte 10 und 11) lesbar (Zeitbasis: ms, Bytezählung bei 0 beginnend). • Bei Fehleralarmen ohne die aktuellen Informationen • Bei Alarmen ist in der Zustandsinfo die aktuelle Parametrierung der Alarmquelle enthalten. • Bei Synchronfehlern wird als Prioritätsklasse B#16#7F eingetragen, wenn die OBs noch nicht bearbeitet wurden, sonst die Prioritätsklasse des letzten Aufrufs. • Hat ein OB mehrere Startereignisse und sind diese zum Auskunftszeitpunkt noch nicht eingetreten, so wird als Ereignisnr. W#16#xyzz zurückgeliefert mit: – x: Ereignisklasse – zz: kleinste definierte Nummer der Gruppe – y: undefiniert. Andernfalls wird die Nummer des letzten aufgetretenen Startereignisses verwendet.
AlarmStatus1	1 Wort	Bearbeitungskennungen: • Bit 0: Alarmereignis ist durch Parametrierung – = 0: freigegeben – = 1: gesperrt • Bit 1: Alarmereignis wurde per "DIS_IRT (Seite 5626)" – = 0: nicht gesperrt – = 1: gesperrt • Bit 2 = 1: Alarmquelle ist aktiv (Generierauftrag liegt vor bei Zeitalarmen (Uhrzeitalarm-OB gestartet), Verzögerungsalarm-OB gestartet, Weckalarm-OB wurde projektiert) • Bit 4: Alarm-OB – = 0: ist nicht geladen – = 1: ist geladen • Bit 5: Alarm-OB ist durch TIS – = 1: gesperrt • Bit 6: Eintrag in Diagnosepuffer – = 1: gesperrt

Name	Länge	Bedeutung
AlarmStatus2	1 Wort	Reaktion bei nicht geladenem/gesperrtem OB • Bit 0 = 1: Alarmquelle sperren • Bit 1 = 1: Alarmereignisfehler generieren • Bit 2 = 1: CPU geht in den Betriebszustand STOP • Bit 3 = 1: Alarm nur verwerfen
AlarmStatus3	2 Worte	Verwerfen durch TIS-Funktionen: Bit Nr. x gesetzt bedeutet: Die Ereignisnummer, die um x größer ist als die kleinste Ereignisnummer des betreffenden OBs ist durch TIS-Funktion verworfen.

SZL-ID W#16#xy25 - Zuordnung zwischen Teilprozessabbildern und OBs

Zweck

Über die Teilliste mit der SZL-ID (Systemzustandsliste ID) W#16#xy25 erhalten Sie die Zuordnung zwischen Teilprozessabbildern und OBs.

Die Teilliste gibt Auskunft über

- Teilprozessabbilder, die Sie einzelnen OBs zur systemseitigen Aktualisierung zugeordnet haben
- Teilprozessabbilder, die Sie einzelnen Taktsynchronalarm-OBs (OBs 61 bis 64) zugeordnet haben. Die Teilprozessabbildaktualisierung erfolgt hier durch Aufruf der Anweisungen "SYNC_PI (Seite 5452)" und "SYNC_PO (Seite 5454)".
  Die Zuordnung zwischen DP-Mastersystemen und den Taktsynchronalarm-OBs erhalten Sie über die SZL W#16#xy95 (Seite 5731).

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy25 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs • W#16#0025: Zuordnung zwischen allen Teilprozessabbildern und OBs innerhalb der CPU • W#16#0125: Zuordnung zwischen einem Teilprozessabbild und dem zugehörigen OB Die Teilprozessabbild-Nr. geben Sie im Parameter INDEX an. • W#16#0225: Zuordnung zwischen einem OB und den zugehörigen Teilprozessabbildern Die OB-Nr. geben Sie im Parameter INDEX an. Hinweis: Nur den Taktsynchronalarm-OBs (OBs 61 bis 64) können Sie mehrere Teilprozessabbilder zuordnen. • W#16#0F25: Nur SZL-Teillistenkopfinfo
INDEX	• Für SZL-ID W#16#0025: irrelevant • Für SZL-ID W#16#0125: Teilprozessabbild-Nr. • Für SZL-ID W#16#0225: OB-Nr. • Für SZL-ID W#16#0F25: irrelevant
LENTHDR	W#16#0004: Ein Datensatz ist 2 Worte lang (4 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#xy25 hat folgenden Aufbau:

Name	Länge	Bedeutung
PPI*_nr	1 Byte	Teilprozessabbild-Nr.
PPI*_use	1 Byte	Art der Zuordnung zwischen Teilprozessabbild und OB: • Bit 0 = 1: Eingangs-Teilprozessabbild ist dem angegebenen OB zur systemseitigen Aktualisierung zugeordnet. • Bit 1 = 1: Ausgangs-Teilprozessabbild ist dem angegebenen OB zur systemseitigen Aktualisierung zugeordnet. • Bit 2 = 1: Eingangs-Teilprozessabbild ist dem angegebenen Taktsynchronalarm-OB zugeordnet und kann durch Aufruf von "SYNC_PI (Seite 5452)" in diesem OB aktualisiert werden. • Bit 3 = 1: Ausgangs-Teilprozessabbild ist dem angegebenen Taktsynchronalarm-OB zugeordnet und kann durch Aufruf von "SYNC_PO (Seite 5454)" in diesem OB aktualisiert werden. • Bits 4 bis 7: 0
ob_nr	1 Byte	OB-Nr.
res	1 Byte	reserviert
*PPI: Parial Process Image		

Teillistenauszüge

- Teillistenauszug mit SZL-ID = W#16#0025:
  Die Datensätze aller Teilprozessabbilder, die Sie bei der Projektierung einem OB zugeordnet haben, werden in aufsteigender Reihenfolge geliefert. Bei Teilprozessabbildern ohne OB-Zuordnung hat ob_nr den Wert Null. Bei Teilprozessabbildern wird kein Datensatz geliefert.
- Teillistenauszug mit SZL-ID = W#16#0125:
  Falls Sie bei der Projektierung dem angesprochenen Teilprozessabbild einen OB zugeordnet haben, wird ein Datensatz geliefert. Falls Sie keinen OB zugeordnet haben, wird kein Datensatz geliefert.

  Hinweis

  Dem Teilprozessabbild 0 ist der OB 1 fest zugeordnet. Bei der Auskunft über Teilprozessabbild 0 erhalten Sie also stets einen Datensatz.

- Teillistenauszug mit SZL-ID = W#16#0225:
  Zu jedem Teilprozessabbild, das dem angesprochenen OB zugeordnet ist, wird ein Datensatz geliefert. Falls Sie dem angesprochenen OB bei der Projektierung kein Teilprozessabbild zugeordnet haben, wird kein Datensatz geliefert.

  Hinweis

  Den Taktsynchronalarm-OBs können mehrere Teilprozessabbilder zugeordnet sein. In diesem Fall werden mehrere Datensätze geliefert.

- Teillistenauszug mit SZL-ID = W#16#0F25:
  Als Anzahl wird die maximal mögliche Anzahl der Datensätze geliefert.

Beispiele zur Bedeutung der Datensätze

Aufrufparameter von "RDSYSST (Seite 5683)"	Gelieferte Variablen	Erläuterung
SZL_ID = W#16#0125, INDEX = W#16#0008	PPI*_nr = B#16#08, PPI*_use = B#16#03, ob_nr = B#16#15	Es wird ein Datensatz geliefert. Das Eingangs- und das Ausgangsprozessabbild 8 sind dem OB 21 zur systemseitigen Prozessabbildaktualisierung zugeordnet.
SZL_ID = W#16#0125, INDEX = W#16#0009	–	Es wird kein Datensatz geliefert. Damit: Das Teilprozessabbild 9 ist keinem OB zugeordnet.
SZL_ID = W#16#0225, INDEX = W#16#003D	PPI*_nr = B#16#0A, PPI*_use = B#16#C0, ob_nr = B#16#3D PPI*_nr = B#16#10, PPI*_use = B#16#C0, ob_nr = B#16#3D	Es werden zwei Datensätze geliefert. Dem OB 61 sind die Eingangs- und Ausgangsprozessabbilder 10 und 16 zugeordnet. Diese können im OB 61 durch Aufruf von "SYNC_PI (Seite 5452)" und "SYNC_PO (Seite 5454)" aktualisiert werden.

Aufrufparameter von "RDSYSST (Seite 5683)"	Gelieferte Variablen	Erläuterung
SZL_ID = W#16#0225, INDEX = W#16#0001	PPI*_nr = B#16#00, PPI*_use = B#16#03, ob_nr = B#16#01	Es wird ein Datensatz geliefert. Dem OB 1 sind das Eingangs- und Ausgangsprozessabbild 0 zugeordnet. Sie werden systemseitig aktualisiert.
*PPI: Parial Process Image		

SZL-ID W#16#xy32 - Kommunikationszustandsdaten

SZL-ID W#16#xy32 - Kommunikationszustandsdaten

Zweck

Über die Teilliste mit der SZL-IDW#16#xy32 erhalten Sie die Kommunikationszustandsdaten der Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#xy32 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs • W#16#0132: Zustandsdaten zu einem Kommunikationsteil (immer nur ein Datensatz). Den Kommunikationsteil geben Sie im Parameter INDEX an. • W#16#0232: Zustandsdaten zu einem Kommunikationsteil. Den Kommunikationsteil geben Sie im Parameter INDEX an.
INDEX	Kommunikationsteil • Für SZL-IDW#16#0132: – W#16#0005 Diagnose – W#16#0008 Zeitsystem – W#16#000B Zeitsystem – W#16#000C Zeitsystem • Für SZL-IDW#16#0232: W#16#0004 CPU-Schutzstufe, Bedienschalterstellungen und Versionskennungen / Prüfsummen
LENTHDR	W#16#0028: Ein Datensatz ist 20 Worte lang (40 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-IDW#16#0132 ist immer 20 Worte lang. Die Datensätze sind unterschiedlich belegt. Die Belegung hängt vom Parameter INDEX ab, d. h. davon, zu welchem Kommunikationsteil der Datensatz gehört.

Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#0005

Inhalt

Der Teillistenauszug mit der SZL-IDW#16#0132 und dem IndexW#16#0005 enthält Informationen über den Diagnosezustand der Baugruppe.

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#0132 und dem IndexW#16#0005 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	W#16#0005: Diagnose
ext	1 Wort	Erweiterter Funktionsumfang • 0: nein • 1: ja
send	1 Wort	Automatisches Senden • 0: nein • 1: ja
possible	1 Wort	Senden von Anwenderdiagnosemeldungen derzeit möglich • 0: nein • 1: ja
res	16 Worte	reserviert

Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#0008

Inhalt

Der Teillistenauszug mit der SZL-IDW#16#0132 und dem IndexW#16#0008 enthält Informationen über den Zustand des Zeitsystems der Baugruppe.

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#0132 und dem IndexW#16#0008 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	W#16#0008: Zeitsystem-Zustand
cycle	1 Wort	reserviert
corr	1 Wort	Korrekturfaktor für die Uhrzeit
clock 0	1 Wort	Betriebsstundenzähler 0: Zeit in Stunden
clock 1	1 Wort	Betriebsstundenzähler 1: Zeit in Stunden
clock 2	1 Wort	Betriebsstundenzähler 2: Zeit in Stunden
clock 3	1 Wort	Betriebsstundenzähler 3: Zeit in Stunden

Name	Länge	Bedeutung
clock 4	1 Wort	Betriebsstundenzähler 4: Zeit in Stunden
clock 5	1 Wort	Betriebsstundenzähler 5: Zeit in Stunden
clock 6	1 Wort	Betriebsstundenzähler 6: Zeit in Stunden
clock 7	1 Wort	Betriebsstundenzähler 7: Zeit in Stunden
time	4 Worte	Aktuelles Datum und Uhrzeit (Format: date_and_time)
rtm _0	1 Byte	Runtime meter Bit x: Betriebsstundenzähler x, 0 ≤ x ≤ 7 (Bit = 1: Betriebsstundenzähler läuft)
res	1 Byte	reserviert
rtmOV _0	1 Byte	Runtime meter Overflow Bit x: Überlauf von Betriebsstundenzähler x, 0 ≤ x ≤ 7 (Bit = 1: Überlauf)
res	1 Byte	reserviert
status	1 Wort	Uhrzeit-Status (Bitbelegung siehe unten)
res	3 Byte	reserviert
status_valid	1 Byte	Gültigkeit der Variablen status: B#16#01: status gültig

Uhrzeit-Status (status)

Bit	Defaultwert	Beschreibung
15	0	Vorzeichen für Korrekturwert (0: positiv, 1: negativ)
14 bis 10	00000	Korrekturwert Dieser Parameter ermöglicht die Korrektur der mittels Telegramm erhaltenen Baugruppenzeit z. B. auf Ortszeit: `Lokalzeit = Baugruppenzeit ± Korrekturwert * 0,5 h` Diese Korrektur berücksichtigt die Zeitzone und die Zeitdifferenz wegen Sommer- / Winterzeit
9	0	reserviert
8	0	reserviert
7	0	Ankündigungsstunde Dieser Parameter gibt an, ob beim nächsten Stundenwechsel eine Umschaltung von Sommer- nach Winterzeit oder umgekehrt stattfindet. (0: findet nicht statt, 1: findet statt)
6	0	Sommer- / Winterzeit-Indikator Dieser Parameter zeigt an, ob die mittels Korrekturwert errechnete Lokalzeit die Sommerzeit oder die Winterzeit ist. (0: Winterzeit, 1: Sommerzeit)
5	0	Dieser Parameter wird bei S7 nicht verwendet.
4 bis 3	00	Uhrzeitauflösung Dieser Parameter gibt die Genauigkeit der übertragenen Uhrzeit wider. (00: 0.001 s, 01: 0.01 s, 10: 0.1 s, 11: 1 s)
2	0	Dieser Parameter wird bei S7 nicht verwendet.

Bit	Defaultwert	Beschreibung
1	0	Dieser Parameter wird bei S7 nicht verwendet.
0	0	Synchronisationsausfall Dieser Parameter gibt an, ob die im Telegramm übertragene Uhrzeit von einem externen Uhrzeitmaster, z. B. SICLOCK synchronisiert ist. (0: Synchronisation ausgefallen, 1: Synchronisation erfolgt) **Hinweis:** Die Auswertung dieses Bits ist bei einer CPU nur bei ständiger externer Uhrzeitsynchronisation sinnvoll.

Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#000B

Inhalt

Der Teillistenauszug mit der SZL-IDW#16#0132 und dem IndexW#16#000B enthält Informationen über den Zustand der 32-Bit-Betriebsstundenzähler 0 bis 7 der Baugruppe.

Hinweis

Im Teillistenauszug mit der SZL-IDW#16#0132 und dem IndexW#16#0008 werden Ihnen diese Betriebsstundenzähler als 16-Bit-Betriebsstundenzähler angezeigt.
Damit können Sie Programme, die für eine CPU mit 16-Bit-Betriebsstundenzählern entwickelt wurden, und die den Teillistenauszug mit der SZL-IDW#16#0132 und dem IndexW#16#0008 benutzen, weiterhin einsetzen.

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#0132 und dem Index W#16#000B hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	W#16#000B: Zeitsystem-Zustand
rtm _0	1 Byte	Runtime meter Bit x: Zustand von Betriebsstundenzähler x , 0 ≤ x ≤ 7 (Bit = 1: Betriebsstundenzähler läuft)
res	1 Byte	reserviert
rtmOV _0	1 Byte	Runtime meter overflow Bit x: Überlauf von Betriebsstundenzähler x, 0 ≤ x ≤ 7 (Bit = 1: Überlauf)
res	1 Byte	reserviert
clock 0	2 Worte	Betriebsstundenzähler 0: Zeit in Stunden
clock 1	2 Worte	Betriebsstundenzähler 1: Zeit in Stunden
clock 2	2 Worte	Betriebsstundenzähler 2: Zeit in Stunden
clock 3	2 Worte	Betriebsstundenzähler 3: Zeit in Stunden
clock 4	2 Worte	Betriebsstundenzähler 4: Zeit in Stunden
clock 5	2 Worte	Betriebsstundenzähler 5: Zeit in Stunden
clock 6	2 Worte	Betriebsstundenzähler 6: Zeit in Stunden

Name	Länge	Bedeutung
clock 7	2 Worte	Betriebsstundenzähler 7: Zeit in Stunden
res	1 Wort	reserviert

Datensatz des Teillistenauszugs mit der SZL-ID W#16#0132 Index W#16#000C

Inhalt

Der Teillistenauszug mit der SZL-IDW#16#0132 und dem INDEXW#16#000C enthält Informationen über den Zustand der 32-Bit-Betriebsstundenzähler 8 bis 15 der Baugruppe.

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#0132 und dem INDEXW#16#000C hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	W#16#000C: Zeitsystem-Zustand
rtm_0	1 Byte	Runtime meter Bit x: Zustand von Betriebsstundenzähler (8+x) , 0 ≤ x ≤ 7 (Bit = 1: Betriebs-stundenzähler läuft)
res	1 Byte	reserviert
rtmOV _0	1 Byte	Runtime meter overflow Bit x: Überlauf von Betriebsstundenzähler (8+x), 0 ≤ x ≤ 7 (Bit = 1: Überlauf)
res	1 Byte	reserviert
clock 8	2 Worte	Betriebsstundenzähler 8: Zeit in Stunden
clock 9	2 Worte	Betriebsstundenzähler 9: Zeit in Stunden
clock 10	2 Worte	Betriebsstundenzähler 10: Zeit in Stunden
clock 11	2 Worte	Betriebsstundenzähler 11: Zeit in Stunden
clock 12	2 Worte	Betriebsstundenzähler 12: Zeit in Stunden
clock 13	2 Worte	Betriebsstundenzähler 13: Zeit in Stunden
clock 14	2 Worte	Betriebsstundenzähler 14: Zeit in Stunden
clock 15	2 Worte	Betriebsstundenzähler 15: Zeit in Stunden
res	1 Wort	reserviert

Datensatz des Teillistenauszugs mit der SZL-ID W#16#0232 Index W#16#0004

Inhalt

Der Teillistenauszug mit der SZL-IDW#16#0232 und dem Index W#16#0004 enthält Informationen über die CPU-Schutzstufe, die Bedienschalterstellungen und Prüfsummen der Hardware-Konfiguration und des Anwenderprogramms.

Datensatz

Ein Datensatz des Teillistenauszugs mit der SZL-IDW#16#0232 und dem Index W#16#0004 hat folgenden Aufbau:

Name	Länge	Bedeutung
Index	1 Wort	• Byte 1: B#16#04: CPU-Schutzstufe, Bedienschalterstellungen und Prüfsummen • Byte 0: Standard-CPU: B#16#00 Bit 3: 0 = Reserve-CPU, 1 = Master-CPU Bits 4 bis 7: 1111
ProtLevel_ModeSwitch	1 Wort	Durch Betriebsartenschalter eingestellte Schutzstufe (1, 2, 3)
ProtLevel_Par	1 Wort	Parametrierte Schutzstufe (0, 1, 2, 3; 0: kein Passwort vergeben, parametrierte Schutzstufe ungültig).
ProtLevel_Valid	1 Wort	Gültige Schutzstufe der CPU
ModeSwitch	1 Wort	Stellung des Betriebsartenschalters (1:RUN, 2:RUN-P, 3:STOP, 4:MRES, 0:undefiniert bzw. nicht ermittelbar)
Startup_ModeSwitch	1 Wort	Stellung des Anlaufartenschalters (1:CRST, 2:WRST, 0:undefiniert, nicht vorhanden oder nicht ermittelbar)
res	1 Wort	Reserviert
ID_Val_CheckSum	1 Wort	Kennung für die Gültigkeit der vier folgenden Prüfsummen (0: ungültig)
ID_CheckSum1_HWConf	1 Wort	Prüfsumme 1 der Hardware-Konfiguration (Intel-Format): Exklusiv-Oder-Verknüpfung über die Längen aller Systemdatenbausteine
ID_CheckSum2_HWConf	1 Wort	Prüfsumme 2 der Hardware-Konfiguration (Intel-Format): Exklusiv-Oder-Verknüpfung über die Prüfsummen aller Systemdatenbausteine
ID_CheckSum1_UserProg	1 Wort	Prüfsumme 1 des Anwenderprogramms (Intel-Format): Exklusiv-Oder-Verknüpfung über die Längen der folgenden Bausteine: OBs, DBs, FBs, FCs
ID_CheckSum2_UserProg	1 Wort	Prüfsumme 2 des Anwenderprogramms (Intel-Format): Exklusiv-Oder-Verknüpfung über die Prüfsummen der folgenden Bausteine: OBs, DBs, FBs, FCs
res	2 Worte	Reserviert
sfc_req	1 Wort	Anforderung von Schutzstufe 2 bzw. 3 durch die Anweisung "PROTECT" (1: Anforderung ist erfolgt)
sfc_act	1 Wort	Aktivierung von Schutzstufe 2 bzw. 3 durch die Anweisung "PROTECT" (1: Aktivierung ist erfolgt)
res	4 Worte	Reserviert

SZL-ID W#16#xy74 - Zustand der Baugruppen-LEDs

Zweck

Über die Teilliste mit der SZL-ID W#16#xy74 erhalten Sie bei den Standard-CPUs (sofern dort vorhanden) den Zustand der Baugruppen-LEDs.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy74 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0174 Zustand einer LED. Die LED wählen Sie über den Parameter INDEX aus.
INDEX	LED-Kennung (nur relevant bei SZL-IDW#16#0174) • W#16#0001: SF (Sammelfehler) • W#16#0002: INTF (interner Fehler) • W#16#0003: EXTF (externer Fehler) • W#16#0004: RUN • W#16#0005: STOP • W#16#0006: FRCE (Forcen) • W#16#0007: CRST (Neustart) • W#16#0008: BAF (Batteriefehler/Überlast, Kurzschluss von Batteriespannung am Bus) • W#16#0009: USR (anwenderdefiniert) • W#16#000A: USR1 (anwenderdefiniert) • W#16#000B: BUS1F (Busfehler Schnittstelle 1) • W#16#000C: BUS2F (Busfehler Schnittstelle 2) bzw. BUS5F (Busfehler Schnittstelle 5) bei den CPUs 414-3 PN/DP,416-3 PN/DP und 416F-3 PN/DP • W#16#0012: IFM1F (Schnittstellenfehler Interface-Modul 1) • W#16#0013: IFM2F (Schnittstellenfehler Interface-Modul 2) • W#16#0014: BUS3F (Busfehler Schnittstelle 3) • W#16#0015: MAINT (Wartungsanforderung) • W#16#0016: DC24V • W#16#0017: BUS5F (Busfehler Schnittstelle 5) • W#16#0080: IF (init failure) • W#16#0081: UF (user failure) • W#16#0082: MF (monitoring failure) • W#16#0083: CF (communication failure) • W#16#0084: TF (task failure) • W#16#00EC: APPL_STATE_RED • W#16#00ED: APPL_STATE_GREEN
LENTHDR	W#16#0004: Ein Datensatz ist 2 Worte lang (4 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-ID W#16#xy74 hat folgenden Aufbau:

Name	Länge	Bedeutung
CPU_LED_ID	1 Wort	• Byte 0: – Standard-CPU: B#16#00 – Bit 3: 0=Reserve-CPU, 1=Master-CPU – Bits 4 bis 7: 1111 • Byte 1: LED-Kennung
LED_on	1 Byte	Zustand der LED: • 0: aus • 1: an
LED_flash	1 Byte	Blinkzustand der LED: • 0: blinkt nicht • 1: blinkt normal (2 Hz) • 2: blinkt langsam (0,5 Hz))

SZL-ID W#16#xy90 - DP-Mastersystem-Information

Zweck

Über die Teilliste mit der SZL-ID W#16#xy90 erhalten Sie die Zustandsinformation über alle der CPU bekannten DP-Mastersysteme.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy90 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs W#16#0090: Informationen über alle der CPU bekannten DP-Mastersysteme W#16#0190: Informationen über ein DP-Mastersystem W#16#0F90: Nur SZL-Teillistenkopfinformation
INDEX	• Für den Teillistenauszug mit der SZL-ID W#16#0190: – Low Byte: B#16#00 – High Byte: DP-Mastersystem-ID • Für die Teillistenauszüge mit der SZL-ID W#16#0090 und W#16#0F90: – W#16#0000
LENTHDR	W#16#000E: Ein Datensatz ist 7 Worte lang (14 Byte)
N_DR	Anzahl der Datensätze Für den Teillistenauszug mit der SZL-ID W#16#0190: 0 bis 1 Für den Teillistenauszug mit der SZL-ID W#16#0090: • bei einer Standard-CPU: 0 bis 14

Datensatz

Ein Datensatz der Teilliste mit der ID W#16#xy90 hat folgenden Aufbau:

Name	Länge	Bedeutung	
dp_m_id	1 Byte	DP-Mastersystem-ID	
dp_m_rack	1 Byte	Baugruppenträger-Nr. des DP-Masters • bei einer Standard-CPU: 0	
dp_m_slot	1 Byte	Steckplatz des DP-Masters bzw. Steckplatz der CPU (bei integrierter DP-Anschaltung)	
dp_m_SubModule	1 Byte	• bei integrierter DP-Anschaltung: Schnittstellen-Nr. des DP-Masters: – 1: X2 – 2: X1 – 3: IF1 – 4: IF2 • bei externer DP-Anschaltung: 0	
logadr	1 Wort	logische Anfangsadresse des DP-Masters	
res	1 Wort	reserviert	
res	1 Wort	reserviert	
dp_m_state	1 Byte	weitere Eigenschaften des DP-Mastersystems	
		Bit 0:	DP-Mode • 0: S7-kompatibel • 1: DPV1
		Bit 1:	DP-Zyklus • 0: nicht äquidistant • 1: äquidistant
		Bit 2 bis 6:	• reserviert
		Bit 7:	DP-Master-Typ • 0: integrierter DP-Master • 1: externer DP-Master
res	3 Byte	reserviert	

SZL-ID W#16#xy91 - Baugruppenzustandsinformation

Zweck

Über die Teilliste mit der SZL-ID W#16#xy91 erhalten Sie die Zustandsinformation über alle der CPU zugeordneten Baugruppen.

Hinweis

Die Auswertung der SZL-ID W#16#xy91 für ein Modul mit gepackten Adressen (ET 200S) ist nicht möglich.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy91 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs • W#16#0091 Baugruppenzustandsinformation aller gesteckten und projektierten Baugruppen/ Submodule der CPU (nur S7-400) • W#16#0191 Zustandsinformation aller nicht deaktivierten Baugruppen / Baugruppenträger mit falscher Baugruppenkennung (nur S7-400) • W#16#0291 Baugruppenzustandsinformation aller gestörten und nicht deaktivierten Baugruppen (nur S7-400) • W#16#0391 Baugruppenzustandsinformation aller nicht verfügbaren Baugruppen (nur S7-400) • W#16#0591 Baugruppenzustandsinformation aller Submodule der Hostbaugruppe • W#16#0991 Baugruppenzustandsinformation eines DP-Mastersystems • W#16#0A91 Zustandsinformation aller DP-Sub- und DP-Mastersysteme (nur S7-300 ohne CPU 318-2 DP) bzw. PROFINET IO-Systeme • W#16#0C91 Baugruppenzustandsinformation einer Baugruppe im zentralen Aufbau oder an einer integrierten DP-Anschaltung oder an einer PROFINET-Anschaltung über die logische Basisadresse • W#16#4C91 Baugruppenzustandsinformation einer Baugruppe an einer externen DP-Anschaltung über die logische Basisadresse Wenn Sie mehr als 4 externe DP-Anschaltungen einsetzen, kann es fälschlicherweise zum RET_VAL W#16#80A4 kommen. • W#16#0D91 Baugruppenzustandsinformation aller Baugruppen im angegebenen Baugruppenträger/ in der angegebenen Station (DP oder PROFINET) • W#16#0E91 Baugruppenzustandsinformation aller zugeordneten Baugruppen • W#16#0F91 Nur SZL-Teillistenkopfinformation

Inhalt	Bedeutung
INDEX	• Für den Teillistenauszug mit der SZL-ID W#16#0C91: – S7-400: Bits 0 bis 14: logische Basisadresse der Baugruppe, Bit 15: 0 = Eingang, 1 = Ausgang – S7-300: Bits 0 bis 14: irgendeine logische Adresse der Baugruppe, Bit 15: 0 = Eingang, 1 = Ausgang • Für den Teillistenauszug mit der SZL-ID W#16#4C91 (nur S7-400): – Bits 0 bis 14: logische Basisadresse der Baugruppe, – Bit 15: 0 = Eingang / 1 = Ausgang • Für den Teillistenauszug mit den SZL-IDs W#16#0091, W#16#0191,W#16#0291, W#16#0391, W#16#0491, W#16#0591, W#16#0A91, W#16#0E91, W#16#0F91: INDEX ist irrelevant, alle Baugruppen (in Baugruppenträger und in dezentraler Peripherie) • Für den Teillistenauszug mit der SZL-ID W#16#0991 (nur S7-400): W#16#xx00: alle Baugruppen eines DP-Mastersystems (xx enthält die DP-Mastersystem-ID) • Für den Teillistenauszug mit der SZL-ID W#16#0D91 – W#16#00xx: alle Baugruppen und Submodule eines Baugruppenträgers (xx enthält die Nummer des Baugruppenträgers) – W#16#xxyy: alle Baugruppen einer DP-Station bzw. alle IO Devices einer PROFINET IO-Station (PROFIBUS DP: xx enthält die DP-Mastersystem-ID, yy die Stationsnummer; PROFINET IO: Bit 0 bis 10: Stationsnummer, Bit 11 bis 14: die letzten beiden Stellen der PNIO-Subsystem-ID, Bit 15: 1 (siehe unten drittes Bild bei adr1)
LENTHDR	W#16#0010: Ein Datensatz ist 8 Worte lang (16 Byte)
N_DR	Anzahl der Datensätze; produktspezifisch kann die Anzahl der Anweisung "RDSYSST (Seite 5683)" übergebenen Datensätze geringer sein

Bei W#16#0091, W#16#0191 und W#16#0F91 werden pro Baugruppenträger 2 zusätzliche Datensätze geliefert:

- ein Datensatz für die Stromversorgung, soweit vorhanden und projektiert und
- ein Datensatz für den Baugruppenträger.

Die Reihenfolge der Datensätze bei zentralem Aufbau beträgt: PS, Steckplatz1, Steckplatz 2,..., Steckplatz 18, Baugruppenträger.

Ein Datensatz der Teilliste mit der ID W#16#xy91 hat folgenden Aufbau:

Name	Länge	Bedeutung
adr1	1 Wort	• bei zentralem Aufbau: Baugruppenträgernummer • bei dezentralem Aufbau mit PROFIBUS DP: DP-Mastersystem-ID, Stationsnummer • bei dezentralem Aufbau mit PROFINET IO: Bit 15 = 1 (PROFINET IO-Kennung), die letzten beiden Stellen der PROFINET IO-System-ID, Stationsnummer Hinweis: Eine PROFINET-Schnittstelle wird grundsätzlich als "Submodul im zentralen Aufbau" behandelt, und zwar unabhängig von ihrer Verwendung für PROFINET IO.
adr2	1 Wort	• bei zentralem Aufbau und bei dezentralem Aufbau mit PROFIBUS DP: Steckplatznummer und Submodulsteckplatznummer • bei dezentralem Aufbau mit PROFINET IO: Steckplatznummer Hinweis: Eine PROFINET-Schnittstelle wird grundsätzlich als "Submodul im zentralen Aufbau" behandelt, und zwar unabhängig von ihrer Verwendung für PROFINET IO.
LogicalAddr	1 Wort	Erste zugeordnete logische E/A-Adresse (Basisadresse)
SetType	1 Wort	PROFINET IO: Solltyp (s. u.), sonst reserviert
ActualType	1 Wort	PROFINET IO: Isttyp (s. u.), sonst reserviert
res	1 Wort	00xx=CPU-Nr.1-4 (nur S7-400) bei PROFINET IO: • SZL-ID=W#16#0C91: Anzahl der real existierenden Submodule (ohne Submodul 0) • SZL-ID=W#16#0D91: Anzahl der Submodule (ohne Submodul 0) • SZL-ID=W#16#4C91: Anzahl der real existierenden Submodule (ohne Submodul 0) • SZL-ID=W#16#4D91: Anzahl der real existierenden Submodule (ohne Submodul 0)

Name	Länge	Bedeutung
IOStat	1 Wort	E/A-Status • Bit 0 = 1: Baugruppe gestört (über Diagnosealarm erkannt) • Bit 1 = 1: Baugruppe vorhanden • Bit 2 = 1: Baugruppe nicht verfügbar • Bit 3 = 1: Baugruppe deaktiviert • Bit 4 = 1: Störung der Station (nur Stellvertreter-Slot) • Bit 5 = 1: Ein CiR-Vorgang für diese Baugruppe /Station ist gerade aktiv oder noch nicht abgeschlossen • Bit 6 = 1: reserviert für S7-400 • Bit 7 = 1: Baugruppe im Lokalbussegment • Bit 8 bis 15: Datenkennung für logische Adresse (Eingang: B#16#B4, Ausgang: B#16#B5, integrierte DP-Anschaltung: B#16#FE, externe DP-Anschaltung: B#16#FF)
AreaID_ModuleWidth	1 Wort	Bereichskennung/Baugruppenbreite • Bit 0 bis 2 : Baugruppenbreite • Bit 3: Reserviert • Bit 4 bis 6 : Bereichskennung – 0 = S7-400 – 1 = S7-300 – 2 = ET-Bereich – 3 = P-Bereich – 4 = Q-Bereich – 5 = IM3-Bereich – 6 = IM4-Bereich • Bit 7: Reserviert

Bei bestimmten Baugruppen werden im Datensatz folgende Werte dargestellt:

Name	PS (nur S7-400)	CPU	IFM-CPU (S7-300)	Baugruppenträger (nurS7-400)
adr1	Nummer des Baugruppenträgers	Standardinformation wie oben beschrieben	Standardinformation wie oben beschrieben	Nummer des Baugruppenträgers
adr2	W#16#01FF	W#16#0200 oder W#16#0200 bis W#16#1800	W#16#0200	W#16#00FF
LogicalAddr	W#16#0000	W#16#7FFF	W#16#007C	W#16#0000
SetType	Standardinformation wie oben beschrieben	W#16#00C0 oder W#16#0081 oder W#16#0082	W#16#00C0	Standardinformation wie oben beschrieben
IOStat	W#16#0000	Standardinformation wie oben beschrieben	Standardinformation wie oben beschrieben	W#16#0000
AreaID_Module-Width	W#16#0000	W#16#0011 oder W#16#0001 oder W#16#0002	W#16#0011	W#16#0000

Parameter adr1

Der Parameter adr1 enthält:

- bei zentralem Aufbau die Baugruppenträgernummer (0-31).

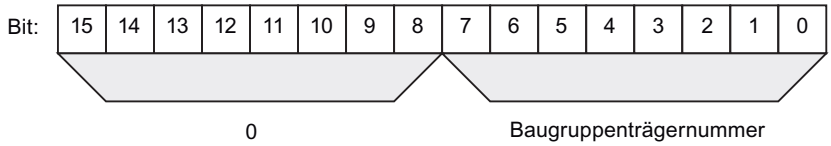

- bei dezentralem Aufbau mit PROFIBUS DP
  - die DP-Mastersystem-ID (1-32)
  - die Stationsnummer (0-127).

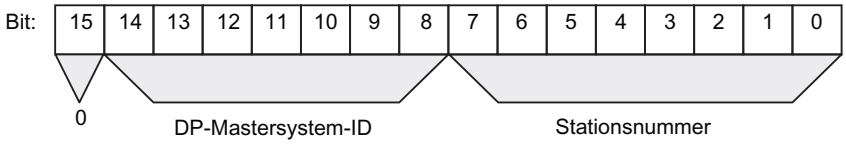

- bei dezentralem Aufbau mit PROFINET IO
  - Kennbit für PROFINET IO (Bit 15)
  - die letzten beiden Stellen der PROFINET IO-System-ID (0-15), um die vollständige PROFINET IO-System-ID zu erhalten, müssen Sie 100 (dezimal) dazu addieren
  - die Stationsnummer (0-2047).

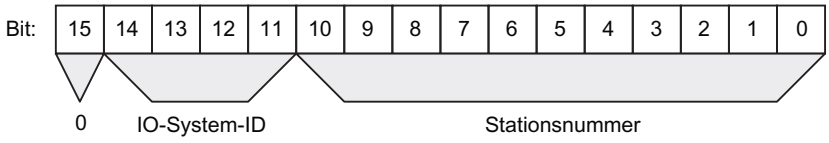

Parameter adr2

Der Parameter adr2 enthält:

- bei zentralem Aufbau und bei dezentralem Aufbau mit PROFIBUS DP die Steckplatznummer und die Submodulsteckplatznummer.

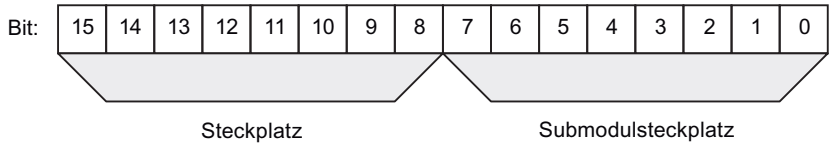

- bei dezentralem Aufbau mit PROFINET IO die Steckplatznummer.

Parameter Solltyp (SetType) und Isttyp (ActualType) für PROFINET IO

Typkennung (W#16#...)	Bedeutung
8100	wird bei Soll- und Isttyp eingetragen, wenn keine Typprüfung möglich ist.
8101	wird bei Solltyp eingetragen, wenn eine Typprüfung möglich ist
8101	wird als Isttyp eingetragen, wenn Soll=Ist ist.
8102	wird als Isttyp eingetragen, wenn Soll<>Ist ist.

SZL-ID W#16#xy92 - Baugruppenträger-/Stationszustandsinformation

Zweck

Über die Teilliste mit der SZL-ID W#16#xy92 erhalten Sie Auskünfte über den Soll- und den Ist-Ausbau von Baugruppenträgern bei zentralem Aufbau und Stationen eines DP-Mastersystems.

Auslesen der SZL mittels "RDSYSST" bei einer S7-400 CPU

Wenn Sie die Teilliste mit "RDSYSST (Seite 5683)" auslesen, müssen Sie streng darauf achten, dass die Parameter SZL_ID und INDEX von "RDSYSST (Seite 5683)" zueinander passen.

SZL_ID	INDEX
W#16#0092 oder W#16#0192 oder W#16#0292 oder W#16#0392 oder W#16#0492 oder W#16#0592 oder W#16#0692	DP-Mastersystem-ID eines DP-Mastersystems, das über eine **integrierte** DP-Anschaltung angeschlossen ist.
:	:
W#16#4092 oder W#16#4292 oder W#16#4692	DP-Mastersystem-ID eines DP-Mastersystems, das über eine **externe** DP-Anschaltung angeschlossen ist.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy92 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs: • W#16#0092: Sollzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist. • W#16#4092: Sollzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist • W#16#0192: Aktivierungsstatus der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist • W#16#0292: Istzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist • W#16#0392: Zustand der Batteriepufferung eines Racks / Baugruppenträgers einer CPU, wenn mindestens eine Batterie ausgefallen ist • W#16#0492: Zustand der gesamten Batteriepufferung aller Racks / Baugruppenträger einer CPU • W#16#0592: Zustand der 24 V-Versorgung aller Racks / Baugruppenträger einer CPU • W#16#4292: Istzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist • W#16#0692: Diagnose-Zustand der Erweiterungsgeräte im zentralen Aufbau / der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist • W#16#4692: Diagnose-Zustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist
INDEX	0/ DP-Mastersystem-ID
LENTHDR	W#16#0010: Datensatz ist acht Worte lang (16 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der ID W#16#xy92 hat folgenden Aufbau:

Inhalt	Länge	Bedeutung
status_0 bis status_15	16 Byte	Rackstatus/ Stationsstatus, Pufferungsstatus oder Aktivierungsstatus (Der Aktivierungsstatus ist nur für DP – Baugruppen relevant). • W#16#0092: – Bit=0: Baugruppenträger/-Station nicht projektiert – Bit=1: Baugruppenträger/-Station projektiert • W#16#4092: – Bit=0: Station nicht projektiert – Bit=1: Station projektiert • W#16#0192: – Bit=0: Station ist nicht projektiert oder projektiert und aktiviert – Bit=1: Station ist projektiert und deaktiviert • W#16#0292: – Bit=0: Baugruppenträger/-Station ausgefallen, deaktiviert oder nicht projektiert – Bit=1: Baugruppenträger/-Station vorhanden, aktiviert und nicht ausgefallen • W#16#4292: – Bit=0: Station ausgefallen, deaktiviert oder nicht projektiert – Bit=1: Station vorhanden, aktiviert und nicht ausgefallen • W#16#0692: – Bit=0: alle Baugruppen des Erweiterungsgeräts/ einer Station sind vorhanden, verfügbar und nicht gestört, und die Station ist aktiviert – Bit=1: mindestens eine Baugruppe des Erweiterungsgeräts/ einer Station ist nicht ok, oder die Station ist deaktiviert • W#16#4692: – Bit=0: alle Baugruppen einer Station sind vorhanden, verfügbar und nicht gestört, und die Station ist aktiviert – Bit=1: mindestens eine Baugruppe einer Station ist nicht ok, oder die Station ist deaktiviert.
status_0	1 Byte	Bit 0: Zentralgerät (INDEX = 0) bzw. Station 1 (INDEX <> 0) Bit 1: 1. Erweiterungsgerät bzw. Station 2 : : Bit 7: 7. Erweiterungsgerät bzw. Station 8
status_1	1 Byte	Bit 0: 8. Erweiterungsgerät bzw. Station 9 : : Bit 7: 15. Erweiterungsgerät bzw. Station 16

Inhalt	Länge	Bedeutung
status_2	1 Byte	Bit 0: 16. Erweiterungsgerät bzw. Station 17 : : Bit 5: 21. Erweiterungsgerät bzw. Station 22 Bit 6: 0 bzw. Station 23 Bit 7: 0 bzw. Station 24
status_3	1 Byte	Bit 0: 0 bzw. Station 25 : : Bit 5: 0 bzw. Station 30 Bit 6: Erweiterungsgerät im SIMATIC-S5-Bereich bzw. Station 31 Bit 7: 0 bzw. Station 32
status_4	1 Byte	Bit 0: 0 bzw. Station 33 : : Bit 7: 0 bzw. Station 40
: :		
status_15	1 Byte	Bit 0: 0 bzw. Station 121 : : Bit 7: 0 bzw. Station 128

SZL-ID W#16#0x94 - Baugruppenträger-/Stationszustandsinformation

Zweck

Über die Teilliste mit der SZL-ID W#16#0x94 erhalten Sie Auskunft über den Soll- und den Ist-Ausbau von Baugruppenträgern bei zentralem Aufbau und Stationen eines PROFIBUS DP-Mastersystems/PROFINET IO-Controllersystems.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#0y94 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs: • W#16#0094: Sollzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines IO-Controllersystems (Status-Bit = 1: Rack/Station projektiert) • W#16#0194: Aktivierungsstatus einer Station eines IO-Controllersystems, die projektiert und deaktiviert ist (Status-Bit =1) • W#16#0294: Istzustand der Baugruppenträger im zentralen Aufbau / der Stationen eines IO-Controllersystems (Status-Bit = 1: Rack/Station vorhanden, aktiviert und nicht ausgefallen) • W#16#0694: Diagnosezustand der Erweiterungsgeräte im zentralen Aufbau / der Stationen eines IO-Controllersystems (Status-Bit = 1: mind. eine Baugruppe des Rack/Station ist gestört oder deaktiviert) • W#16#0794: Diagnose- / Wartungszustand des zentralen Racks / der Stationen eines IO-Controllersystems (Status-Bit = 0: keine Störung und keine Wartung notwendig, Status-Bit = 1: Rack / Station ist gestört oder / und Wartungsbedarf oder / und Wartungsanforderung) • W#16#0F94: Nur Kopfinformationen
INDEX	• 0: Zentrale Baugruppe • 1-32: Dezentrale Baugruppe am PROFIBUS DP • 100-115: Dezentrale Baugruppe am PROFINET IO
LENTHDR	Länge des nachfolgenden Datensatzes
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der ID W#16#0y94 hat folgenden Aufbau:

Inhalt	Länge	Bedeutung
index	1 Wort	• 0: Zentrale Baugruppe • 1-32: Dezentrale Baugruppe am PROFIBUS DP • 100-115: Dezentrale Baugruppe am PROFINET IO
status_0	BOOL	Sammelinformation • 1: mindestens eines der nachfolgenden Statusbits hat den Wert 1 • 0: alle nachfolgenden Statusbits haben den Wert 0
status_1	BOOL	Zustand Station 1
status_2	BOOL	Zustand Station 2

Inhalt	Länge	Bedeutung
..		
status_2047	BOOL	Zustand Station 2047

Ein Statusbit von nicht projektierten Racks/Stationen/Devices enthält den Wert "0".

Hinweis

Wichtiger Unterschied zur Vorgänger SZL-ID W#16#xy92

Die Daten sind gegenüber der Vorgänger SZL-ID W#16#xy92 um ein Bit verschoben, da das Bit status_0 als Sammelinformation dient.

SZL-ID W#16#xy95 - Erweiterte DP-Mastersystem / PROFINET IO-System-Information

Zweck

Über die Teilliste mit der SZL-ID W#16#xy95 erhalten Sie die erweiterte Zustandsinformation über alle der CPU bekannten DP-Mastersysteme / PROFINET IO-Systeme. Gegenüber der Teilliste mit der SZL-ID W#16#xy90 sind Aussagen zu PROFINET IO-Systemen und zusätzliche Aussagen zur Taktsynchronität eines DP-Mastersystems enthalten.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy95 ist wie folgt aufgebaut:

Inhalt	Bedeutung	
SZL-ID	Die SZL-ID des Teillistenauszugs	
	W#16#0195:	Erweiterte Informationen über ein DP-Mastersystem/PROFINET IO-System
	W#16#0F95:	Nur SZL-Teillistenkopfinformation
INDEX	• Für den Teillistenauszug mit der SZL-ID W#16#0195: – Low Byte: B#16#00 – High Byte: DP-Mastersystem-ID/PROFINET IO System-ID • Für den Teillistenauszug mit der SZL-ID W#16#0F95: – W#16#0000	
LENTHDR	W#16#0028:	Ein Datensatz ist 20 Worte lang (40 Byte)
N_DR	Anzahl der Datensätze: Für den Teillistenauszug mit der SZL-ID W#16#0195: 0 bis 1	

Datensatz

Ein Datensatz der Teilliste mit der ID W#16#xy95 hat folgenden Aufbau:

Name	Länge	Bedeutung	
dp_m_id	1 Byte	DP-Mastersystem-ID/PROFINET IO System-ID	
dp_m_rack	1 Byte	Baugruppenträger-Nr. des DP-Masters / IO-Controllers • bei einer Standard-CPU: 0	
dp_m_slot	1 Byte	Steckplatz des DP-Masters / IO-Controllers bzw. Steckplatz der CPU (bei integrierter DP-Anschaltung)	
dp_m_SubModule	1 Byte	• bei integrierter Anschaltung: Schnittstellen-Nr. des DP-Masters / IO-Controllers: – 1: X2 – 2: X1 – 3: IF1 – 4: IF2 • bei externer Anschaltung: 0	
LogicalAddr	2 Byte	logische Anfangsadresse des DP-Masters / IO-Controllers	
res	2 Byte	reserviert	
res	2 Byte	reserviert	
dp_m_state	1 Byte	• weitere Eigenschaften des DP-Mastersystems / PROFINET IO-Systems	
		Bit 0:	DP-Mode (nur bei PROFIBUS DP) • 0: S7-kompatibel • 1: DPV1
		Bit 1:	DP- bzw. PN-Zyklus • 0: nicht äquidistant • 1: äquidistant
		Bit 2 bis 6:	• reserviert
		Bit 7:	DP-Master- / IO-Controller-Typ • 0: integrierter DP-Master / IO-Controller • 1: externer DP-Master / IO-Controller
dp_address	1 Byte	DP-Teilnehmernummer (PROFIBUS-Adresse)	
res	2 Byte	reserviert	
SyncCyclInter_OB	1 Byte	zugeordneter Taktsynchronalarm-OB (nur relevant, falls DP- bzw. PN-Zyklus äquidistant)	
res	1 Byte	reserviert	
baudrate	4 Byte	Baudrate des DP-Mastersystems bzw. PNIO-Systems als Hex-Wert	
IsoDPCycle	4 Byte	Zeitdauer des äquidistanten DP- bzw. PN-Zyklus in µs	
res	16 Byte	reserviert	

SZL-ID W#16#xy96 - Baugruppenzustandsinformation PROFINET IO und PROFIBUS DP

Zweck

Über die Teilliste mit der SZL-ID W#16#xy96 erhalten Sie die Zustandsinformation über alle der CPU zugeordneten Baugruppen.

Ergänzend zu SZL-ID W#16#xy91 (Seite 5720) erhalten Sie über die Teilliste mit der SZL-ID W#16#xy96 zusätzliche Zustandsdaten von Modulen und Submodulen.

Sie erhalten dabei sowohl PROFINET IO-spezifische Informationen als auch Informationen zu PROFIBUS DP-Baugruppen und Zentrale Baugruppen.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy96 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs • W#16#0696 Baugruppenzustandsinformation aller Submodule einer angegebenen Baugruppe (bei PROFIBUS DP und zentralen Baugruppen ist die Submodulebene nicht vorhanden). • W#16#0C96 Baugruppenzustandsinformation einer Baugruppe/eines Submoduls zentral oder an einer PROFIBUS DP/PROFINET-Anschaltung über die Anfangsadresse.
INDEX	• Bits 0 bis 14: Adresse der Baugruppe • Bit 15: 0 = Eingang, 1 = Ausgang
LENTHDR	Länge des nachfolgenden Datensatzes
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der ID W#16#xy96 hat folgenden Aufbau:

Inhalt	Länge	Bedeutung
LogicalAddr	1 Wort	• Bits 0 bis 14: Adresse der Baugruppe • Bit 15: 0 = Eingang, 1 = Ausgang
System	1 Wort	Kennung für zentrale Baugruppe / DP-Mastersystem-ID / PROFINET IO-System ID: • 0: Zentrale Baugruppe • 1-32: Dezentrale Baugruppe am PROFIBUS DP • 100-115: Dezentrale Baugruppe am PROFINET IO
API*	2 Worte	Projektiertes Anwendungsprofil bei einem dezentralen PROFINET-Gerät. Profile sind branchen-/technologiespezifische Festlegungen, die über die PROFINET-Norm hinausgehen. Profil 0 bedeutet, dass die Daten den Festlegungen der PROFINET-Norm entsprechen.
Station	1 Wort	Baugruppenträger-Nr./Stationsnummer/Gerätenummer
Slot	1 Wort	Steckplatz-Nr.
Subslot	1 Wort	Submodulsteckplatz (falls kein Submodul gesteckt werden kann, ist hier "0" anzugeben)
Offset	1 Wort	Offset im Nutzdatenadressraum der zugehörigen Baugruppe

<table>
<tr><th>Inhalt</th><th>Länge</th><th colspan="3">Bedeutung</th></tr>
<tr><td rowspan="9">SetType</td><td rowspan="9">7 Worte</td><td colspan="3">Solltyp
Der Solltyp ist bei PROFINET IO hierarchisch aufgebaut</td></tr>
<tr><td>Wort</td><td>PROFINET IO</td><td>PROFIBUS DP</td></tr>
<tr><td>1:</td><td>Herstellernummer oder Profilidentifikation (z. B.: W#16#FF00 für PROFIBUS)</td><td>0000</td></tr>
<tr><td>2:</td><td>Gerät</td><td>0000</td></tr>
<tr><td>3:</td><td>Laufende Nummer oder Profilindex</td><td>0000</td></tr>
<tr><td>4:</td><td>1. Wort des Doppelwortes Modul Identifikation</td><td>Typkennung</td></tr>
<tr><td>5:</td><td>2. Wort des Doppelwortes Modul Identifikation</td><td>0000</td></tr>
<tr><td>6:</td><td>1. Wort des Doppelwortes Submodul Identifikation</td><td>0000</td></tr>
<tr><td>7:</td><td>2. Wort des Doppelwortes Submodul Identifikation</td><td>0000</td></tr>
<tr><td>SetType_UnEqual_Actual-Type</td><td>1 Wort</td><td colspan="3">Kennung Soll/Ist
• Bit 0 = 0: Soll entspricht Ist
• Bit 0 = 1: Soll ungleich Ist
• Bit 1 bis 15: reserviert</td></tr>
<tr><td>res</td><td>1 Wort</td><td colspan="3">Reserviert</td></tr>
<tr><td>IOStat</td><td>1 Wort</td><td colspan="3">E/A-Status
• Bit 0 = 1: Baugruppe gestört (über Diagnosealarm erkannt)
• Bit 1 = 1: Baugruppe vorhanden
Bit 2 = 1: Baugruppe nicht verfügbar
• Bit 3 = 1: Baugruppe deaktiviert
Bit 4 = 1: Störung der Station (nur Stellvertreter-Slot)
• Bit 5 = 1: M7: Baugruppe kann Hostbaugruppe für Submodule sein
S7: Ein CiR-Vorgang für diese Baugruppe /Station ist gerade aktiv oder noch nicht abgeschlossen
• Bit 6 = 1: reserviert für S7-400
• Bit 7 = 1: Baugruppe im Lokalbussegment (nur bei S7-300)
• Bit 8 = 1: Baugruppe Wartungsbedarf ("grün")
• Bit 9 = 1: Baugruppe Wartungsanforderung ("gelb")
• Bit 10 bis 15: reserviert</td></tr>
</table>

Inhalt	Länge	Bedeutung
AreaID_ModuleWidth	1 Wort	Bereichskennung/Baugruppenbreite • Bit 0 bis 2: Baugruppenbreite • Bit 3: Reserviert • Bit 4 bis 6 : Bereichskennung – 0 = S7-400 – 1 = S7-300 – 2 = PROFINET IO (dezentral) – 3 = P-Bereich – 4 = Q-Bereich – 5 = IM3-Bereich – 6 = IM4-Bereich • Bit 7: Reserviert • Bit 7: Reserviert
res	5 Worte	Reserviert
* Application Process Instance		

Teilliste mit der SZL-ID W#16#0696 für Baugruppen am PROFIBUS DP

Hierbei kommt es zur Fehlermeldung "Submodulebene nicht vorhanden".

SZL-ID W#16#xy9C - Werkzeugwechslerinformationen (PROFINET IO)

Zweck

Über die Teilliste mit der SZL-ID W#16#xy9C erhalten Sie Informationen über die projektierten Werkzeugwechsler und deren Werkzeuge.

Werkzeugwechsler sind IO-Devices, die Werkzeuge verwalten. Ein Werkzeug besteht aus einem oder mehreren IO-Devices. Jedes Werkzeug ist dabei eindeutig einem Port eines Werkzeugwechslers zugeordnet.

Bei einem Werkzeugwechsel werden zunächst alle zum derzeit aktiven Werkzeug gehörenden IO-Devices deaktiviert und anschließend die zum neuen Werkzeug gehörenden IO-Devices aktiviert. Das Deaktivieren und das Aktivieren erfolgen durch die Anweisung "D_ACT_DP (Seite 5484)".

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xy9C ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs· • W#16#009C Informationen über alle Werkzeugwechsler und deren Werkzeuge an einem PROFINET IO-System • W#16#019C Informationen über alle Werkzeugwechsler an einem PROFINET IO-System • W#16#029C Informationen über einen Werkzeugwechsler und dessen Werkzeuge· • W#16#039C Informationen über ein Werkzeug und dessen IO-Devices· • W#16#0F9C Nur SZL-Teillistenkopfinformation
INDEX	Für den Teillistenauszug mit der SZL-ID • W#16#009C: PROFINET IO System-ID • W#16#019C: PROFINET IO System-ID • W#16#029C: logische Adresse des Werkzeugwechslers • W#16#039C: logische Adresse irgendeines IO-Devices des Werkzeugs • W#16#0F9C: PROFINET IO System-ID
LENTHDR	W#16#000C (Länge des nachfolgenden Datensatzes in Byte: 12)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-ID W#16#xy9C hat folgenden Aufbau:

Inhalt	Länge	Bedeutung
StationW	1 WORD	Stationsnummer (des Datensatzes, kann Werkzeugwechsler oder Station eines Werkzeugs sein)
LogAdrW	1 WORD	• Bits 0 bis 14: Adresse der Baugruppe • Bit 15: 0 = Eingang, 1 = Ausgang
StationWZK	1 WORD	Stationsnummer des Werkzeugkopfs (Der Werkzeugkopf ist das am Port des Werkzeugwechslers direkt angeschlossene IO-Device eines Werkzeugs.)
StationWZW	1 WORD	Stationsnummer des Werkzeugwechslers
SlotWZW	1 WORD	Steckplatz des Werkzeugwechslers
SubslotWZW	1 WORD	Submodulsteckplatz des Werkzeugwechslers

Hinweis

Wenn ein Werkzeugwechsler mehrere Ports besitzt, an dem Werkzeuge verwaltet werden, wird für jeden Port ein Datensatz geliefert.

Beispiel

Die Verwendung der einzelnen Teillisten wird anhand des folgenden Beispiels nochmals verdeutlicht.

Am Strang eines PROFINET IO-Systems befinden sich zwei Werkzeugwechsler (IOD 3 und IOD 10), deren Werkzeuge die folgende Struktur haben:

- Werkzeugwechsler IOD 3 (IO-Device 3) mit folgenden drei Werkzeugen am Werkzeugwechslerport 2:
  - Werkzeug 1 (IOD 4, IOD 5 und IOD 6)
  - Werkzeug 2 (IOD 7)
  - Werkzeug 3 (IOD 8 und IOD 9)
- Werkzeugwechsler IOD 10 mit zwei Werkzeugwechslerports, von denen jeder zwei Werkzeuge besitzt
  - Werkzeugwechslerport 3: Werkzeug 1 (IOD 11 und IOD 12), Werkzeug 2 (IOD 13)
  - Werkzeugwechslerport 4: Werkzeug 1 (IOD 14, IOD 15 und IOD 16), Werkzeug 2 (IOD 17 und IOD 18)

Damit ergibt sich der folgende Aufbau:

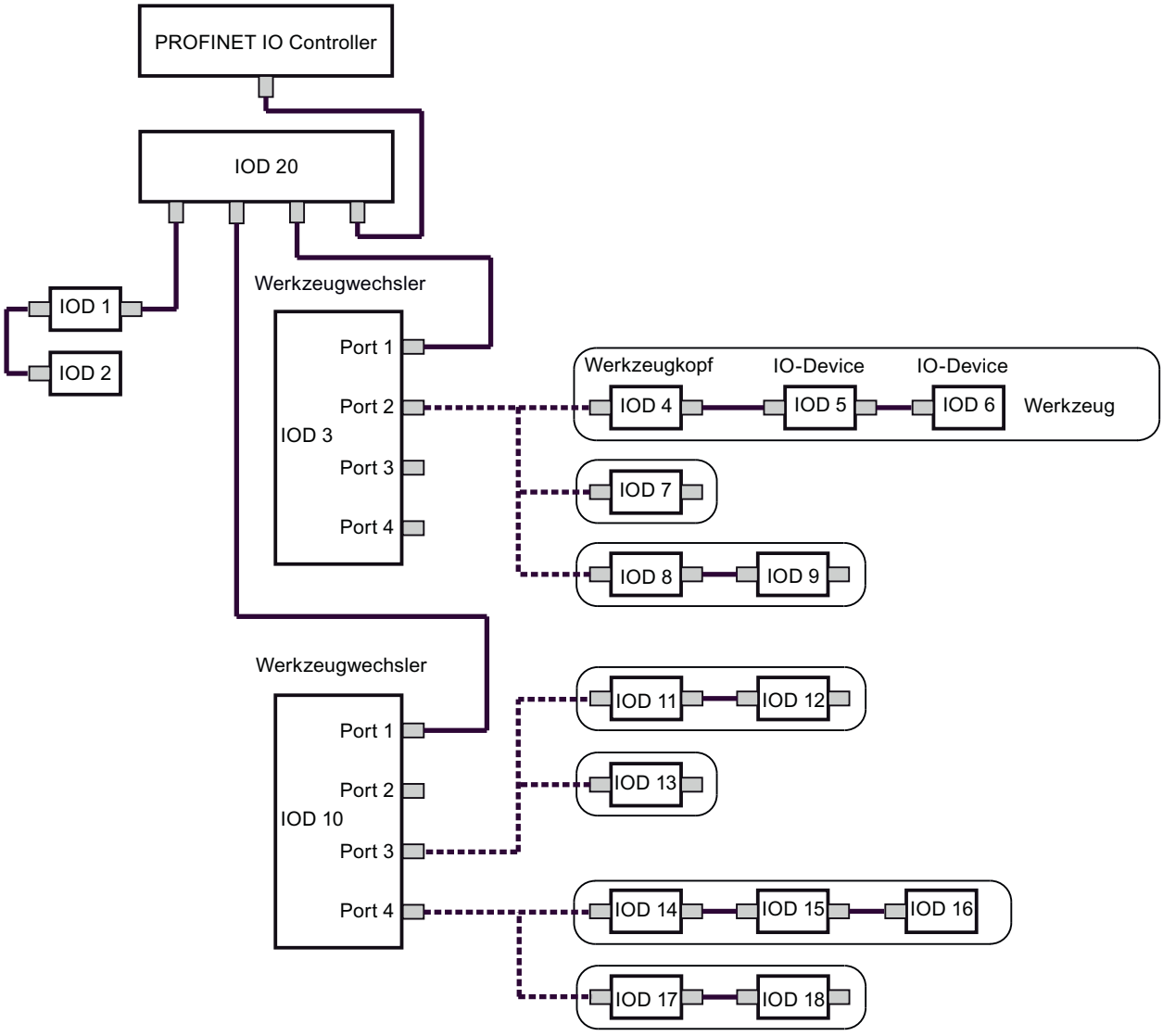

Die Teillisten der SZL liefern Datensätze für die folgenden IO-Devices:

- Teilistenauszug mit SZL-ID W#16#009C (Index: PROFINET IO System-ID): liefert 17 Datensätze für die folgenden IO-Devices:
  - Werkzeugwechsler: IOD 3, IOD 10 (für Port 3) und IOD 10 (für Port 4)
  - Werkzeuge: IOD 4, IOD 5, IOD 6, IOD 7, IOD 8, IOD 9, IOD 11, IOD 12, IOD 13, IOD 14, IOD 15, IOD 16, IOD 17 und IOD 18
- Teilistenauszug mit SZL-ID W#16#019C (Index: PROFINET IO System-ID): liefert 3 Datensätze für die folgenden IO-Devices:
  - Werkzeugwechsler: IOD 3, IOD 10 (für Port 3) und IOD 10 (für Port 4)
  - Werkzeuge: keine

- Teilistenauszug mit SZL-ID W#16#029C (Index: Adresse des IOD 3): liefert 3 Datensätze für die folgenden IO-Devices:
  - Werkzeugwechsler: IOD 3
  - Werkzeuge: IOD 4, IOD 5, IOD 6, IOD 7, IOD 8 und IOD 9
- Teilistenauszug mit SZL-ID W#16#029C (Index: Adresse des IOD 10): liefert 10 Datensätze für die folgenden IO-Devices:
  - Werkzeugwechsler: IOD 10 (für Port 3) und IOD 10 (für Port 4)
  - Werkzeuge: IOD 11, IOD 12, IOD 13, IOD 14, IOD 15, IOD 16, IOD 17 und IOD 18
- Teilistenauszug mit SZL-ID W#16#039C (Index: Logische Adresse des IOD 4): liefert 3 Datensätze für die folgenden IO-Devices: (Dies gilt analog für die logische Adresse des IOD 5 und IOD 6)
  - Werkzeugwechsler: keine
  - Werkzeuge: IOD 4, IOD 5 und IOD 6
- Teilistenauszug mit SZL-ID W#16#039C (Index: Logische Adresse des IOD 13): liefert einen Datensatz für die folgenden IO-Devices:
  - Werkzeugwechsler: keine
  - Werkzeuge: IOD 13

SZL-ID W#16#xyA0 - Diagnosepuffer

Zweck

Über die Teilliste mit der SZL-ID W#16#xyA0 erhalten Sie die Einträge im Diagnosepuffer der Baugruppe.

Hinweis

Die S7-300-CPUs liefern maximal so viele Datensätze wie die im Betriebszustand RUN angezeigte Anzahl der Diagnosepuffereinträge (voreingestellter Wert: 10). Die S7-400-CPUs liefern maximal 21 Datensätze.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#xyA0 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	Die SZL-ID des Teillistenauszugs: • W#16#00A0: Alle im aktuellen Betriebszustand lieferbaren Einträge • W#16#01A0: Die neuesten Eintragungen; die Anzahl der neuesten Eintragungen geben Sie über den Parameter INDEX an. Wenn die Anzahl der Meldungen im Diagnosepuffer noch kleiner ist, als die von Ihnen projektierte maximale Anzahl der Meldungen, kann Ihnen unter Umständen die Anweisung "RDSYSST (Seite 5683)" bei diesem Teillistenauszug ungültige Werte liefern. Vermeiden Sie daher einen ungepufferten NETZAUS! • W#16#0FA0: Nur SZL-Teillistenkopfinfo
INDEX	Nur für SZL-ID W#16#01A0: Anzahl der neuesten Eintragungen
LENTHDR	W#16#0014: Ein Datensatz ist 10 Worte lang (20 Byte)
N_DR	Anzahl der Datensätze

Datensatz

Ein Datensatz der Teilliste mit der SZL-ID W#16#xyA0 hat folgenden Aufbau:

Name	Länge in Worten	Bedeutung
ID	1 Wort	Ereignis-ID
info	5 Worte	Informationen zum Ereignis bzw. zu dessen Wirkung
time	4 Worte	Zeitstempel des Ereignisses

Diagnosepuffer

Weitere Informationen zu Ereignissen im Diagnosepuffer erhalten Sie über in der Kontexthilfe zum angezeigten Ereignis.

SZL-ID W#16#00B1 - Baugruppendiagnoseinfo

Zweck

Über die Teilliste mit der SZL-ID W#16#00B1 erhalten Sie die ersten 4 Diagnosebytes einer diagnosefähigen Baugruppe.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#00B1 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	W#16#00B1
INDEX	• Bit 0 bis 14: logische Basisadresse • Bit 15: 0 für Eingang, 1 für Ausgang
LENTHDR	W#16#0004: Ein Datensatz ist 2 Worte lang (4 Byte)
N_DR	1

Datensatz

Ein Datensatz der Teilliste mit der SZL-ID W#16#00B1 hat folgenden Aufbau:

Name	Länge	Bedeutung
byte0	1 Byte	• Bit 0: Baugruppenstörung/OK (Sammelstörkennung) • Bit 1: Fehler intern • Bit 2: Fehler extern • Bit 3: Kanalfehler vorhanden • Bit 4: Externe Hilfsspannung fehlt • Bit 5: Frontstecker fehlt • Bit 6: Baugruppe nicht parametriert • Bit 7: Falsche Parameter in Baugruppe
byte1	1 Byte	• Bit 0 bis Bit 3: Baugruppenklasse (CPU, FM, CP, IM, SM, ...) • Bit 4: Kanalinformation vorhanden • Bit 5: Anwenderinformation vorhanden • Bit 6: Diagnosealarm von Stellvertreter • Bit 7: Wartungsbedarf (nur bei PROFINET IO)

Name	Länge	Bedeutung
byte2	1 Byte	• Bit 0: Anwendermodul falsch/fehlt • Bit 1: Kommunikationsstörung • Bit 2: Betriebszustand RUN/STOP (0 = RUN, 1 = STOP) • Bit 3: Zeitüberwachung angesprochen (watch dog) • Bit 4: BG-interne Versorgungsspannung ausgefallen • Bit 5: Batterie leer (BFS) • Bit 6: Gesamte Pufferung ausgefallen • Bit 7: Wartungsanforderung (nur bei PROFINET IO)
byte3	1 Byte	• Bit 0: Erweiterungsgeräteausfall (von IM erkannt) • Bit 1: Prozessorausfall • Bit 2: EPROM-Fehler • Bit 3: RAM-Fehler • Bit 4: ADU/DAU-Fehler • Bit 5: Sicherungsausfall • Bit 6: Prozessalarm verloren • Bit 7: Reserve (mit 0 initialisiert)

SZL-ID W#16#00B2 - Diagnosedatensatz1 über physikalische Adresse

Zweck

Über die Teilliste mit der SZL-ID W#16#00B2 erhalten Sie den Diagnosedatensatz1 einer zentral gesteckten Baugruppe (also nicht für DP und Submodule). Die Baugruppe geben Sie über Baugruppenträger und Steckplatznummer an.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#00B2 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	W#16#00B2
INDEX	W#16#xxyy: • xx enthält die Nummer des Baugruppenträgers • yy enthält die Steckplatznummer
LENTHDR	Länge des Datensatzes hängt von der Baugruppe ab
N_DR	1

Datensatz

Wie groß ein Datensatz der Teilliste mit der SZL-ID W#16#00B2 ist und wie der Datensatz belegt ist, hängt von der jeweiligen Baugruppe ab. Nähere Informationen hierzu finden Sie im jeweiligen Handbuch zur Baugruppe.

SZL-ID W#16#00B3 - Baugruppendiagnosedaten über logische Basisadresse

Zweck

Über die Teilliste mit der SZL-ID W#16#00B3 erhalten sie alle Diagnosedaten einer Baugruppe. Diese Auskunft ist auch für DP und Submodule möglich. Sie wählen die Baugruppe über ihre logische Basisadresse aus.

Kopf

Der Kopf der Teilliste mit der SZL-ID W#16#00B3 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	W#16#00B3
INDEX	• Bit 0 bis 14: logische Basisadresse • Bit 15: 0 für Eingang, 1 für Ausgang
LENTHDR	Länge eines Datensatzes (baugruppenabhängig)
N_DR	1

Datensatz

Wie groß ein Datensatz der Teilliste mit der SZL-ID W#16#00B3 ist und wie er belegt ist, hängt von der jeweiligen Baugruppe ab. Nähere Informationen hierzu finden Sie im jeweiligen Handbuch zur Baugruppe.

SZL-ID W#16#00B4 - Diagnosedaten eines DP-Slaves

Zweck

Über die Teilliste mit der SZL-ID W#16#00B4 erhalten Sie die Diagnosedaten eines DP-Slaves. Diese Diagnosedaten sind nach EN50 170 Volume 2, PROFIBUS aufgebaut. Sie wählen die Baugruppe über ihre projektierte Diagnoseadresse aus.

Kopf

Der Kopf der Teilliste mit der SZL-IDW#16#00B4 ist wie folgt aufgebaut:

Inhalt	Bedeutung
SZL-ID	W#16#00B4
INDEX	Projektierte Diagnoseadresse des DP-Slaves
LENTHDR	Länge eines Datensatzes. Die Maximallänge beträgt 240 Byte; bei Normslaves, bei denen die Anzahl der Normdiagnosedaten größer als 240 Byte ist und maximal 244 Byte beträgt, werden die ersten 240 Byte gelesen und das entsprechende Overflow-Bit in den Daten gesetzt.
N_DR	1

Datensatz

Ein Datensatz der Teilliste mit der SZL-ID W#16#00B4 hat folgenden Aufbau:

Name	Länge	Bedeutung
status1	1 Byte	Stationsstatus1
status2	1 Byte	Stationsstatus2
status3	1 Byte	Stationsstatus3
Station_No	1 Byte	Master-Stationsnummer
Vendor_ID_highByte	1 Byte	Herstellerkennung (high byte)
Vendor_ID_lowByte	1 Byte	Herstellerkennung (low byte)
....	....	Weitere slavespezifische Diagnose

OB_RT: OB Programmlaufzeit ermitteln

Beschreibung

Mit der Anweisung können Sie die Laufzeit einzelner OBs über verschiedene Zeiträume ermitteln.

Hinweis

Die Anweisung liefert die zuletzt aufgezeichneten Zeitwerte für den gewünschten OB unabhängig davon, ob dieser momentan geladen ist oder nicht. Die Daten werden auch durch Löschen oder Überladen nicht zurückgesetzt, sondern nur durch einen Neustart (Warmstart).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "OB_RT":

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung
OB_NR	Input	INT	E, A, M, D, L	OB, dessen zuletzt ermittelte Zeiten abgefragt werden sollen. Gültige OB-Nummern sind alle in Ihrer CPU realisierten OBs mit Ausnahme von OB 121 und OB 122. Die Bearbeitung von Synchronfehlern zählt zur Bearbeitungszeit des jeweils fehlerverursachenden OB. Die Angabe der OBs 121 und 122 oder nicht in der CPU realisierten OBs führt zu einer Fehlermeldung. Bei OB_NR=0 werden die Daten des OB, in dessen Kontext die Anweisung aufgerufen wurde, übergeben. Bei Aufruf von der Anweisung "OB_RT" in den OBs 121 oder 122 mit OB_NR=0 werden die Zeiten des alarmverursachenden OB inclusive der Zeiten im OB 12x ausgegeben.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Andernfalls enthält RET_VAL die Nummer des OB, für den diese Daten abgerufen wurden.
PRIO	Output	INT	E, A, M, D, L	In PRIO wird die Prioritätsklasse des abgefragten OB ausgegeben.
LAST_RT	Output	DINT	E, A, M, D, L	Laufzeit der letzten abgeschlossenen Bearbeitung des angegebenen OB in Mikrosekunden. Falls der OB, zu dem Sie Laufzeiten ermitteln wollen, momentan gerade in Bearbeitung ist, gilt: Beim ersten Aufruf von der Anweisung "OB_RT" während der aktuellen Bearbeitung des gewünschten OB wird in LAST_RT die Laufzeit der letzten abgeschlossenen OB-Bearbeitung angegeben. Bei jedem weiteren Aufruf der Anweisung "OB_RT" während der aktuellen Bearbeitung des gewünschten OB wird in LAST_RT • DW#16#FFFF FFFF angegeben, falls im gewünschten OB bereits ein "OB_RT"-Aufruf mit OB_NR=0 erfolgt ist. • die Laufzeit der letzten abgeschlossenen OB-Bearbeitung angegeben, falls im gewünschten OB kein "OB_RT"-Aufruf mit OB_NR=0 erfolgt ist. Hinweis: Die Unterbrechungszeiten durch höherpriore OBs sind in LAST_RT nicht enthalten. Die OB-spezifischen Betriebssystemleistungen (z. B. die Erzeugung und die Bereitstellung der OB-Startinformation, die Aktualisierung des Prozessabbilds, die Aktualisierung des Teilprozessabbilds) sind in LAST_RT enthalten.

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung
LAST_ET	Output	DINT	E, A, M, D, L	Zeitspanne zwischen OB-Anforderung und Bearbeitungsende des angegebenen OB in Mikrosekunden, und zwar für die letzte abgeschlossene Bearbeitung des angegebenen OB. Falls der OB, zu dem Sie Laufzeiten ermitteln wollen, momentan gerade in Bearbeitung ist, gilt: Beim ersten Aufruf der Anweisung "OB_RT" während der aktuellen Bearbeitung des gewünschten OB wird in LAST_ET die Zeitspanne zwischen der letzten vollständig bearbeiteten OB-Anforderung und dem Bearbeitungsende des angegebenen OB angegeben. Bei jedem weiteren Aufruf der Anweisung "OB_RT" während der aktuellen Bearbeitung des gewünschten OB wird in LAST_ET • DW#16#FFFF FFFF angegeben, falls im gewünschten OB bereits ein "OB_RT"-Aufruf mit OB_NR=0 erfolgt ist. • die Zeitspanne zwischen der letzten vollständig bearbeiteten OB-Anforderung und dem Bearbeitungsende des gewünschten OB angegeben, falls im gewünschten OB kein "OB_RT"-Aufruf mit OB_NR=0 erfolgt ist. Hinweis: Die Unterbrechungszeiten durch höherpriore OBs sind in LAST_ET enthalten.
CUR_T	Output	DINT	E, A, M, D, L	Zeitpunkt der OB-Anforderung des angegebenen OB, der momentan gerade bearbeitet wird, als Relativzeit in Mikrosekunden. Falls der angegebene OB momentan nicht bearbeitet wird, enthält CUR_T den Wert "0". Hinweis: Die Systemzeit ist ein Zähler, der von "0" bis "2 147 483 647" Mikrosekunden zählt. Beim Überlauf startet der Zähler wieder bei "0".
CUR_RT	Output	DINT	E, A, M, D, L	Bisherige Laufzeit der aktuellen Bearbeitung des angegebenen OB in Mikrosekunden. CUR_RT ist "0", wenn der OB nicht oder noch nicht bearbeitet wird. Nach Ende der Bearbeitung wird die Laufzeit in LAST_RT übertragen, und CUR_RT wird auf 0 gesetzt. Hinweis: Die Unterbrechungszeiten durch höherpriore OBs sind in CUR_RT nicht enthalten. Die bis zum Aufrufzeitpunkt von der Anweisung "OB_RT" erbrachten OB-spezifischen Betriebssystemleistungen sind in CUR_RT enthalten.

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung
CUR_ET	Output	DINT	E, A, M, D, L	Die seit der Anforderung des angegebenen OB, der momentan gerade bearbeitet wird, vergangene Zeitspanne in Mikrosekunden. CUR_ET ist "0", wenn der angegebene OB momentan nicht bearbeitet wird. Nach Ende der Bearbeitung wird die Laufzeit in LAST_ET übertragen, und CUR_ET wird auf "0" gesetzt. Hinweis: Die Unterbrechungszeiten durch höherpriore OBs sind in CUR_ET enthalten.
NEXT_ET	Output	DINT	E, A, M, D, L	Falls weitere Bearbeitungen des angegebenen OB anstehen, bevor die aktuelle Anforderung beendet wurde, wird in NEXT_ET die Zeitspanne vom aktuellen Zeitpunkt bis zum Zeitpunkt der nächst folgenden Anforderung in Mikrosekunden angezeigt. NEXT_ET ist "0", wenn außer dem aktuellen zur Bearbeitung anstehenden bzw. in Bearbeitung stehenden Startereignis für den betroffenen OB kein weiteres Startereignis existiert. WinLC RTX und die S7-400-CPUs verwenden diesen Parameter nicht. Bei ihnen hat NEXT_ET den Wert "FFFF FFFF". Hinweis: Die Unterbrechungszeiten durch höherpriore OBs sind in NEXT_ET enthalten.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Die Zeiten enthalten auch die Laufzeiten eventuell eingeschachtelter Bearbeitungen von Synchronfehler-Alarmen (OB 121, OB 122).

Hinweis

Falls Sie in OB_NR eine OB-Nr. angeben, die im Mengengerüst Ihrer CPU zwar vorhanden ist, der zugehörige OB vom Betriebssystem aber noch nicht aufgerufen bzw. von Ihnen noch nicht in die CPU geladen wurde, enthält RET_VAL die angegebene OB-Nr., PRIO die projektierte (ggf. Default-) Priorität des angegebenen OB. Die Laufzeitparameter (CUR_RT, CUR_ET, LAST_RT, LAST_ET, NEXT_ET) liefern den Initialwert DW#16#FFFF FFFF zurück. Bei den folgenden Anlaufarten bzw. Betriebszustandswechseln werden die Laufzeitparameter auf ihren Initialwert gesetzt: Neustart, Kaltstart, Wiederanlauf.

Parameter RET_VAL

Ereignisklasse Fehlercode	Erläuterung
1 bis 102	Nummer des OB, zu dem Informationen übergeben werden
W#16#8080	Parameter OB_NR enthält einen unzulässigen Wert.
W#16#8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

C_DIAG: Aktuellen Verbindungszustand ermitteln

Beschreibung

Mit der Anweisung können Sie den aktuellen Zustand aller S7-Verbindungen und aller hochverfügbaren S7-Verbindungen (bzw. deren Teilverbindungen) ermitteln.

Durch geeignete Auswertung dieser Verbindungsdaten können Sie den Ausfall von S7-Verbindungen und hochverfügbaren S7-Verbindungen erkennen und ggf. an ein Bedien- und Beobachtungssystem melden. Bei den überwachten Verbindungen kann es sich sowohl um Verbindungen zwischen Automatisierungssystemen als auch um Verbindungen zwischen einem Automatisierungssystem und einem Bedien- und Beobachtungssystem handeln.

Hinweis

Der Betriebszustandswechsel RUN -> STOP -> RUN einer CPU hat keinen Einfluss auf den Zustand der projektierten Verbindungen. Ausnahme: Beim Wechsel einer H-Station vom Systemzustand Redundant in den Systemzustand Stop werden bei hochverfügbaren Verbindungen diejenigen Teilverbindungen abgebaut, die auf der Reserve-CPU enden.

Nach Netzausfall hingegen werden alle projektierten Verbindungen neu aufgebaut, so dass sich der Verbindungszustand ändert.

Beim Erstaufruf von der Anweisung im oder nach dem Anlauf sind die Verbindungsinformationen also unterschiedlich, je nachdem, ob der letzte Betriebszustand der CPU STOP oder NETZAUS war.

Arbeitsweise

Die Anweisung "C_DIAG" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung kann sich über mehrere Aufrufe erstrecken.

Sie starten den Auftrag, indem Sie "C_DIAG" mit REQ=1 aufrufen.

Falls der Auftrag sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Auftrag noch in Bearbeitung.

Aufruf

Um den Ausfall von S7-Verbindungen und hochverfügbaren S7-Verbindungen zu erkennen, rufen Sie "C_DIAG" in einem Weckalarm-OB auf, der z. B. alle 10 Sekunden vom Betriebssystem gestartet wird.

Da sich der Zustand einer Verbindung im Normalfall nur selten ändert, ist es bei diesen zyklischen Aufrufen sinnvoll, dass die Verbindungsdaten nur dann in das Anwenderprogramm kopiert werden, wenn sie sich gegenüber dem letzten Aufruf geändert haben (Aufruf mit MODE=B#16#02, siehe unten).

Die Anweisung "C_DIAG" hat vier mögliche Betriebsarten, die in der folgenden Tabelle erläutert werden.

MODE (B#16#...)	"C_DIAG" kopiert Verbindungsdaten ins Anwenderprogramm	"C_DIAG" übergibt Quittierungsinfo an das Betriebssystem
00	Nein	Ja
01	Ja	Ja
02	• Ja, falls sich Verbindungsdaten geändert haben • Nein, falls sich keine Verbindungsdaten geändert haben	Ja
03	Ja	Nein

Durch die Übergabe der Quittierungsinformation an das Betriebssystem werden die seit dem letzten Aufruf von "C_DIAG" (mit MODE=B#16#00, 01 oder 02) aufgetretenen Zustandsänderungen der Verbindungsdaten quittiert.

Hinweis

Wenn Sie "C_DIAG" in einem Weckalarm-OB in der Betriebsart "Bedingtes Kopieren" (MODE=B#16#02) betreiben, müssen Sie dafür sorgen, dass nach einem Kaltstart der CPU im Zielbereich keine Initialwerte stehen. Dies erreichen Sie, indem Sie im OB 102 "C_DIAG" mit der Betriebsart "Unbedingtes Kopieren mit Quittierung" (MODE=B#16#01) einmal aufrufen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "C_DIAG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request to activate REQ=1: Anstoß des Auftrags, falls noch nicht gestartet
MODE	Input	BYTE	E, A, M, D, L oder Konstante	Auftragskennung Mögliche Werte: • B#16#00: "C_DIAG" kopiert keine Verbindungsdaten, sondern übergibt dem Betriebssystem lediglich eine Quittierungsinformation. • B#16#01: "C_DIAG" kopiert die Verbindungsdaten unabhängig von deren Änderungszustand ins Anwenderprogramm und übergibt dem Betriebssystem eine Quittierungsinformation. • B#16#02: Falls sich die Verbindungsdaten geändert haben, kopiert "C_DIAG" diese ins Anwenderprogramm. Falls sie sich nicht geändert haben, erfolgt kein Kopiervorgang. "C_DIAG" übergibt dem Betriebssystem in beiden Fällen eine Quittierungsinformation. • B#16#03: "C_DIAG" kopiert die Verbindungsdaten unabhängig von deren Änderungszustand ins Anwenderprogramm. Sie übergibt dem Betriebssystem keine Quittierungsinformation.
RET_VAL	Return	INT	E, A, M, D, L	Rückgabewert (Fehlercode bzw. Auftragszustand)
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der Auftrag ist noch nicht beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
N_CON	Output	INT	E, A, M, D, L	Index der letzten Struktur in CON_ARR, bei der .DIS_PCON oder .DIS_CON den Wert TRUE haben. Im Anwenderprogramm brauchen also nur die ersten N_CON Elemente von CON_ARR überprüft werden. Hinweis: Die erste Struktur im Feld CON_ARR hat den Index "1".
CON_ARR	Output	ANY	E, A, M, D, L	Zielbereich für die gelesenen Verbindungsdaten. Es ist nur der Datentyp BYTE zulässig. Jeder Verbindung ist eine Struktur zugeordnet. Wählen Sie den Zielbereich so groß, dass er auch bei der maximal möglichen Verbindungsanzahl Ihrer CPU alle Strukturen aufnehmen kann. Beachten Sie dies insbesondere nach dem Austausch einer CPU, wenn die neue CPU mehr Verbindungsressourcen hat als die bisherige.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter CON_ARR

Der Zielbereich für die gelesenen Verbindungsdaten ist ein Feld von Strukturen. Dabei ist jeder Verbindung eine Struktur zugeordnet.

Das Feld braucht nicht von vorne her mit gültigen Einträgen belegt zu sein, und zwischen zwei gültigen Einträgen können ungültige Einträge stehen.

Die Verbindungen sind nicht nach Verbindungsreferenzen geordnet.

Hinweis

Beim Kopieren der Verbindungsdaten aus dem Betriebssystem in den von Ihnen vorgegebenen Zielbereich wird die Konsistenz für die Daten einer Verbindung gewährleistet.

Aufbau einer Struktur

Parameter	Datentyp	Beschreibung
CON_ID	WORD	Verbindungsreferenz, die Sie per Projektierung für diese Verbindung vergeben haben W#16#FFFF: ungültige Kennung, d. h. Verbindung nicht projektiert. Falls zusätzlich CON_ARR[i].DIS_PCON oder CON_ARR[i].DIS_CON (siehe unten) gesetzt ist, wurde diese Verbindung seit dem letzten Aufruf von "C_DIAG" umprojektiert oder gelöscht.
STAT_CON	BYTE	Aktueller Zustand der S7-Verbindung bzw. hochverfügbaren S7-Verbindung Mögliche Werte: • B#16#00: S7-Verbindung nicht aufgebaut • B#16#10: Hochverfügbare S7-Verbindung nicht aufgebaut • B#16#01: S7-Verbindung wird momentan aufgebaut • B#16#11: Hochverfügbare S7-Verbindung wird momentan aufgebaut • B#16#02: S7-Verbindung ist aufgebaut • B#16#12: Hochverfügbare S7-Verbindung ist aufgebaut, jedoch nicht hochverfügbar • B#16#13: Hochverfügbare S7-Verbindung ist aufgebaut und hochverfügbar
PROD_CON	BYTE	Teilverbindungs-Nr. der Produktivverbindung Mögliche Werte: 0, 1, 2, 3
STBY_CON	BYTE	Teilverbindungs-Nr. der Standbyverbindung (B#16#FF: keine Standbyverbindung) Mögliche Werte: 0, 1, 2, 3 Hinweis: Nur eine hochverfügbare S7-Verbindung kann eine Standbyverbindung haben.
DIS_PCON	BOOL	Die Übergänge W#16#12 -> W#16#13 und W#16#13 -> W#16#12 von CON_ARR[i].STAT_CON seit dem letzten Anweisungs-Aufruf setzen CON_ARR[i].DIS_PCON auf "1". Alle anderen Zustandsänderungen der Verbindung i lassen CON_ARR[i].DIS_PCON unverändert. Hinweis: • Bei MODE=B#16#01 und 02 wird dasjenige Bit im Betriebssystem, das DIS_PCON entspricht, mit dem Kopieren der Verbindungsdaten in den Zielbereich rückgesetzt. • Bei MODE=B#16#03 bleibt dasjenige Bit im Betriebssystem, das DIS_PCON entspricht, unverändert.
DIS_CON	BOOL	Jede Änderung von CON_ARR[i].STAT_CON seit dem letzten "C_DIAG"-Aufruf mit Ausnahme der Übergänge W#16#12 -> W#16#13 und W#16#13 -> W#16#12 setzt CON_ARR[i].DIS_CON auf 1. Hinweis: • Bei MODE=B#16#01 und 02 wird dasjenige Bit im Betriebssystem, das "DIS_CON" entspricht, mit dem Kopieren der Verbindungsdaten in den Zielbereich rückgesetzt. • Bei MODE=B#16#03 bleibt dasjenige Bit im Betriebssystem, das "DIS_CON" entspricht, unverändert.
RES0	BYTE	reserviert (B#16#00)
RES1	BYTE	reserviert (B#16#00)

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	• MODE=B#16#00, 01 oder 02: keine Änderung des Verbindungszustands (Strukturelement STAT_CON) seit dem letzten Aufruf. Der Auftrag wurde fehlerfrei durchgeführt. • MODE=B#16#03: Der Kopiervorgang wurde fehlerfrei durchgeführt.
0001	• MODE=B#16#00, 01 oder 02: Änderung des Verbindungszustands (Strukturelement STAT_CON) bei mindestens einer Verbindung seit dem letzten Aufruf. Der Auftrag wurde fehlerfrei durchgeführt. • MODE=B#16#03: RET_VAL W#16#0001 ist nicht möglich:
7000	Erstaufruf mit REQ=0. Der über MODE festgelegte Auftrag wird nicht bearbeitet. BUSY hat den Wert "0",
7001	Erstaufruf mit REQ=1. Der über MODE festgelegte Auftrag wurde angestoßen. BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant). Der aktivierte Auftrag läuft noch. BUSY hat den Wert "1".
8080	Der Parameter MODE enthält einen unzulässigen Wert.
8081	Der Parameter CON_ARR enthält einen unzulässigen Datentyp.
8082	Der Parameter CON_ARR enthält eine zu kleine Längenangabe. Die Anweisung kopiert keine Daten in den Zielbereich.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.3.10 Datenbausteinfunktionen

READ_DBL: Aus Datenbaustein im Ladespeicher lesen

Beschreibung

Mit der Anweisung kopieren Sie einen DB oder einen Teil eines DB, der im Ladespeicher (Micro Memory Card) liegt, in den Datenbereich eines Ziel-DB. Der Ziel-DB muss ablaufrelevant sein, d. h. er darf nicht mit dem Attribut UNLINKED erzeugt worden sein. Der Inhalt des Ladespeichers wird beim Kopiervorgang nicht verändert.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Zielbereich während der Bearbeitung von "READ_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Für die Parameter SRCBLK und DSTBLK gelten folgende Einschränkungen:

- Bei einem ANY-Pointer vom Typ BOOL muss die Länge durch 8 teilbar sein.
- Bei einem ANY-Pointer vom Typ STRING muss die Länge gleich 1 sein.

Die Länge des Quell-DB können Sie bei Bedarf mit der Anweisung "TEST_DB (Seite 5765)" ermitteln.

Hinweis

READ_DBL wird asynchron bearbeitet und eignet sich deshalb nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu lesen.

Ein einmal begonnener Auftrag wird stets zu Ende geführt. Wenn die maximale Anzahl gleichzeitig aktiver "READ_DBL"-Anweisungen erreicht ist und Sie "READ_DBL" zu diesem Zeitpunkt in einer Prioritätsklasse hoher Priorität erneut aufrufen, dann wird der Fehlercode W#16#80C3 zurückgeliefert. Es ist daher nicht sinnvoll, den hochprioren Auftrag sofort wieder zu starten.

Arbeitsweise

Die Anweisung "READ_DBL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "READ_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ_DBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Lesen
SRCBLK	Input	ANY	D	Zeiger auf den Datenbaustein im Ladespeicher, aus dem gelesen wird
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Lesevorgang ist noch nicht beendet.
DSTBLK	Output	ANY	D	Zeiger auf den Datenbaustein im Arbeitsspeicher, in den geschrieben werden soll

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert.
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
8081	Der Quellbereich ist größer als der Zielbereich. **Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.**
8093	Beim Parameter DSTBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
80B1	Beim Parameter SRCBLK ist kein Datenbaustein angegeben, oder der dort angegebene Datenbaustein ist kein Ladespeicherobjekt (z. B. ein per "CREAT_DB (Seite 5759)" erzeugter DB).
80B4	DB mit F-Attribut darf nicht gelesen werden.
80C0	Der Ziel-DB wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.
80C3	Die maximale Anzahl gleichzeitig aktiver "READ_DBL"-Anweisungen ist momentan bereits erreicht.
8xyy	allgemeine Fehlercodes Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

WRIT_DBL: In Datenbaustein im Ladespeicher schreiben

Beschreibung

Mit der Anweisung übertragen Sie den Inhalt eines DBs oder eines DB-Bereichs aus dem Arbeitsspeicher in einen DB oder einen DB-Bereich im Ladespeicher (Micro Memory Card). Der Quell-DB muss ablaufrelevant sein, d. h. er darf nicht mit dem Attribut UNLINKED erzeugt worden sein. Er darf jedoch per Anweisung "CREAT_DB" erzeugt worden sein.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Quellbereich während der Bearbeitung von "WRIT_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Für die Parameter SRCBLK und DSTBLK gelten folgende Einschränkungen:

- Bei einem ANY-Pointer vom Typ BOOL muss die Länge durch 8 teilbar sein.
- Bei einem ANY-Pointer vom Typ STRING muss die Länge gleich 1 sein.

Die Länge des Ziel-DB können Sie bei Bedarf mit der Anweisung "TEST_DB (Seite 5765)" ermitteln.

Die "WRIT_DBL" verändert nicht die Prüfsumme des Anwenderprogramms, wenn Sie einen mittels Anweisung erzeugten DB beschreiben. Beim Beschreiben eines geladenen DB

hingegen verändert das erste Schreiben auf diesen DB die Prüfsumme des Anwenderprogramms.

Hinweis

"WRIT_DBL" eignet sich nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu schreiben, da auf eine Micro Memory Card technologiebedingt nur eine bestimmte Anzahl von Schreibzugriffen möglich ist.

Arbeitsweise

Die Anweisung "WRIT_DBL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "WRIT_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRIT_DBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Schreiben
SRCBLK	Input	ANY	D	Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Schreibvorgang ist noch nicht beendet.
DSTBLK	Output	ANY	D	Zeiger auf den Datenbaustein im Ladespeicher, in den geschrieben werden soll.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert.
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1:Datenübertragung angestoßen; BUSY hat den Wert "1".

Fehlercode (W#16#...)	Beschreibung
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".
8081	Der Quellbereich ist größer als der Zielbereich. **Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.**
8092	Falsche Betriebsart: Während "WRIT_DBL" aktiv war, ist die CPU in STOP gegangen. Beim nächsten Übergang nach RUN wird dieser Fehlercode geliefert. Rufen Sie "WRIT_DBL" erneut auf.
8093	Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
80B1	Beim Parameter DSTBLK ist kein Datenbaustein angegeben, oder der dort angegebene Datenbaustein ist kein Ladespeicherobjekt (z. B. ein per "CREAT_DB (Seite 5759)" erzeugter DB).
80B4	DB mit F-Attribut darf nicht verändert werden
80C0	Das Ziel wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet. Beispiel: Sie laden einen DB von der CPU in das PG. Den Inhalt dieses DB wollen Sie per "WRIT_DBL" verändern.
80C3	Die maximale Anzahl gleichzeitig aktiver "WRIT_DBL"-Anweisungen ist momentan bereits erreicht.
8xyy	allgemeine Fehlercodes Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

DEL_DB: Datenbaustein löschen

Beschreibung

Mit der Anweisung löschen Sie einen im Arbeitsspeicher und gegebenenfalls im Ladespeicher der CPU liegenden Datenbaustein. Der zu löschende DB darf weder in der aktuellen noch in einer niederprioren Ablaufebene aufgeschlagen sein. D. h., er darf weder in einem der beiden DB-Register noch im B-Stack eingetragen sein.

Bei Aufruf der Anweisung startet die CPU sonst den OB 121. Wenn der OB 121 nicht vorhanden ist, wechselt die CPU in STOP; bei S7-300 (Ausnahme: CPU 318) erfolgt das Löschen des DB ohne Aufruf des OB 121.

Hinweis

Das Löschen von Instanz-DBs mit "DEL_DB" ist nicht sinnvoll und führt stets zu Programmfehlern. Vermeiden sie daher das Löschen von Instanz-DBs mit "DEL_DB"!

Die folgende Tabelle erläutert, wann ein DB mit "DEL_DB" gelöscht werden kann.

Wenn der DB ...	dann ist er mit "DEL_DB" ...
durch Aufruf der Anweisung "CREAT_DB" erzeugt wurde,	löschbar.
auf die CPU übertragen wurde und nicht mit dem Attribut "nur im Ladespeicher ablegen" erzeugt wurde	löschbar.
mit dem Attribut "nur im Ladespeicher ablegen" erzeugt wurde	• bei S7-300 löschbar • bei S7-400 nicht löschbar
durch Aufruf der Anweisung "CREA_DBL" erzeugt wurde	löschbar
auf der Flashcard gespeichert ist,	nicht löschbar.

Unterbrechbarkeit

Die Anweisung ist von höherprioren Ablaufebenen unterbrechbar. Falls dort die Anweisung erneut aufgerufen wird, wird dieser zweite Aufruf abgebrochen, und in RET_VAL wird W#16#8091 eingetragen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DEL_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB_NUMBER	Input	WORD	E, A, M, D, L oder Konstante	Nummer des zu löschenden DB
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8091	Bei ineinandergeschachtelten "DEL_DB"-Aufrufen wurde die maximale Schachtelungstiefe der verwendeten CPU überschritten.
8092	Die Funktion "Löschen eines DB" ist momentan nicht durchführbar, weil • die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist • Sie den zu löschenden DB gerade von der CPU in ein Offline-Projekt kopieren. • die WinAC-Software-CPU einen Fehler im Betriebssystem des Computers festgestellt hat, auf dem WinAC installiert ist.
80A1	Fehler beim Eingangsparameter DB_NUMBER: Der gewählte Aktualparameter • hat den Wert "0" • ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.
80B1	Der DB mit der angegebenen Nummer ist im Arbeitsspeicher der CPU nicht vorhanden.
80B3	Der DB befindet sich auf der Flashcard.
80B4	Der DB konnte nicht gelöscht werden. Mögliche Ursachen: • Er gehört zu einem F-Programm. • Er ist ein Instanz-DB eines Bausteins der S7-Kommunikation (nur bei S7-400). • Er ist ein Technologie-DB.
80C1	Die Funktion "Löschen eines DB" ist wegen temporären Ressourcenengpasses momentan nicht durchführbar.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CREAT_DB: Datenbaustein erzeugen

Beschreibung

Mit der Anweisung erzeugen Sie im Anwenderprogramm einen Datenbaustein, der keine vorbesetzten Werte enthält. Er enthält stattdessen Zufallsdaten. Die Anweisung erzeugt einen Datenbaustein mit einer Nummer aus einem angegebenen Bereich sowie mit vorgegebener Größe. Aus dem angegebenen Bereich vergibt die Anweisung die kleinste mögliche Nummer an den DB. Einen DB mit bestimmter Nummer erzeugen Sie, indem Sie der Ober- und Untergrenze des vorzugebenden Bereichs die gleiche Nummer geben. Die Nummern der im Anwenderprogramm bereits enthaltenen DBs können Sie nicht mehr vergeben. Die Länge des DBs müssen Sie in einer geraden Zahl angeben.

Unterbrechbarkeit

Die Anweisung kann unterbrochen werden durch höherpriore OBs. Wenn in einem höherprioren OB wiederum eine Anweisung "CREAT_DB" aufgerufen wird, wird dieser Aufruf mit dem Fehlercode W#16#8091 abgewiesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CREAT_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LOW_LIMIT	Input	WORD	E, A, M, D, L oder Konstante	Der untere Grenzwert ist die kleinste Nummer in dem Bereich der Nummern, die Sie Ihrem Datenbaustein zuordnen können.
UP_LIMIT	Input	WORD	E, A, M, D, L oder Konstante	Der obere Grenzwert ist die größte Nummer in dem Bereich der Nummern, die Sie Ihrem Datenbaustein zuordnen können.
COUNT	Input	WORD	E, A, M, D, L oder Konstante	Der Zählwert gibt die Anzahl der Datenbytes an, die Sie für ihren Datenbaustein reservieren möchten. Sie müssen hier eine gerade Zahl an Bytes (maximal 65534) angeben.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
DB_NUMBER	Output	WORD	E, A, M, D, L	Die Datenbausteinnummer ist die Nummer des erstellten Datenbausteins. Im Fehlerfall (Bit 15 von RET_VAL wurde gesetzt) wird in DB_NUMBER der Wert "0" eingetragen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	kein Fehler
8091	Sie haben CREAT_DB geschachtelt aufgerufen.
8092	Die Funktion "Erzeugen eines DB" ist momentan nicht durchführbar, weil • die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist • die Anzahl der in der CPU vorhandenen DBs die maximal mögliche Anzahl bereits erreicht hat • sich die H-CPU im Ankoppel- oder Aufdatvorgang befindet. • die WinAC-Software-CPU einen Fehler im Betriebssystem des Computers festgestellt hat, auf dem WinAC installiert ist. • das vorherige Löschen noch nicht abgeschlossen ist
80A1	Fehler in der Nummer des DB: • die Nummer ist "0" • die Nummer überschreitet die CPU-spezifische DB-Anzahl • Untergrenze > Obergrenze
80A2	Fehler in der Länge des DB: • die Länge ist "0" • die Länge wurde als ungerade Zahl angegeben • die Länge ist größer als die CPU zulässt
80B1	Es ist keine DB-Nummer frei.
80B2	Es steht nicht genügend freier Speicherplatz zur Verfügung.
80B3	Es steht nicht genügend zusammenhängender Speicher zur Verfügung. (Komprimieren durchführen!)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CREA_DB: Datenbaustein erzeugen / Remanent

Beschreibung

Mit der Anweisung erzeugen Sie im Anwenderprogramm einen Datenbaustein, der keine vorbesetzten Werte enthält. Er enthält stattdessen Zufallsdaten. Die Anweisung erzeugt einen Datenbaustein mit einer Nummer aus einem angegebenen Bereich sowie mit vorgegebener Größe. Aus dem angegebenen Bereich vergibt die Anweisung die kleinste mögliche Nummer an den DB. Einen DB mit bestimmter Nummer erzeugen Sie, indem Sie der Ober- und Untergrenze des vorzugebenden Bereichs die gleiche Nummer geben. Die Nummern der im Anwenderprogramm bereits enthaltenen DBs können Sie nicht mehr vergeben. Die Länge des DBs müssen Sie in einer geraden Zahl angeben.

Je nach Wahl des Parameters ATTRIB hat der erzeugte DB entweder die Eigenschaft RETAIN oder NON_RETAIN:

- RETAIN (=remanent) bedeutet, dass der DB im remanenten Teil des Arbeitsspeichers erzeugt wird, d. h. die Aktualwerte des DB bleiben bei jedem Netz-Aus/Netz-Ein-Übergang und jedem Neustart (Warmstart) erhalten.
- NON_RETAIN (= nicht remanent) bedeutet, dass der DB im nicht remanenten Teil des Arbeitsspeichers erzeugt wird, d. h. die Aktualwerte des DB sind nach jedem Netz-Aus/Netz-Ein-Übergang und nach jedem Neustart (Warmstart) undefiniert.

Falls nicht zwischen remanentem und nicht remanentem Arbeitsspeicher unterschieden wird, wird der Parameter ATTRIB ignoriert, d. h. die Werte des DB bleiben nach jedem Netz-Aus/Netz-Ein-Übergang und nach jedem Neustart (Warmstart) erhalten.

Unterbrechbarkeit

Die Anweisung "CREA_DB" kann unterbrochen werden durch höherpriore OBs. Wenn in einem höherprioren OB wiederum eine Anweisung "CREA_DB" aufgerufen wird, wird dieser Aufruf mit dem Fehlercode W#16#8091 abgewiesen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CREA_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LOW_LIMIT	Input	WORD	E, A, M, D, L oder Konstante	Der untere Grenzwert ist die kleinste Nummer in dem Bereich der Nummern, die Sie Ihrem Datenbaustein zuordnen können.
UP_LIMIT	Input	WORD	E, A, M, D, L oder Konstante	Der obere Grenzwert ist die größte Nummer in dem Bereich der Nummern, die Sie Ihrem Datenbaustein zuordnen können.
COUNT	Input	WORD	E, A, M, D, L oder Konstante	Der Zählwert gibt die Anzahl der Datenbytes an, die Sie für ihren Datenbaustein reservieren möchten. Sie müssen hier eine gerade Zahl an Bytes (maximal 65534) angeben.
ATTRIB	Input	BYTE	E, A, M, D, L oder Konstante	DB-Attribute: • B#16#00: RETAIN • B#16#04: NON_RETAIN
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
DB_NUMBER	Output	WORD	E, A, M, D, L	Die Datenbausteinnummer ist die Nummer des erstellten Datenbausteins. Im Fehlerfall (Bit 15 von RET_VAL wurde gesetzt) wird in DB_NUMBER der Wert "0" eingetragen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	kein Fehler
8091	Sie haben CREA_DB geschachtelt aufgerufen.
8092	Die Funktion "Erzeugen eines DB" ist momentan nicht durchführbar, weil • die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist • die WinAC-Software-CPU einen Fehler im Betriebssystem des Computers festgestellt hat, auf dem WinAC installiert ist.
8094	Unzulässiger Wert in ATTRIB
80A1	Fehler in der Nummer des DB: • die Nummer ist "0" • die Nummer überschreitet die CPU-spezifische DB-Anzahl • Untergrenze > Obergrenze
80A2	Fehler in der Länge des DB: • die Länge ist "0" • die Länge wurde als ungerade Zahl angegeben • die Länge ist größer als die CPU zulässt
80B1	Es ist keine DB-Nummer frei.
80B2	Es steht nicht genügend freier Speicherplatz zur Verfügung.
80B3	Es steht nicht genügend zusammenhängender Speicher zur Verfügung. (Komprimieren durchführen!)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

CREA_DBL: Datenbaustein im Ladespeicher erzeugen

Beschreibung

Mit der Anweisung erzeugen Sie im Ladespeicher (Micro Memory Card) einen neuen Datenbaustein. Die Anweisung erzeugt einen Datenbaustein mit einer Nummer aus einem angegebenen Bereich sowie mit vorgegebener Größe. Aus dem angegebenen Bereich vergibt die Anweisung die kleinste mögliche Nummer an den DB. Einen DB mit einer bestimmten Nummer erzeugen Sie, in dem Sie der Ober- und Untergrenze des anzugebenden Bereichs die gleiche Nummer geben. Die Nummer der im Anwenderprogramm bereits enthaltenen DBs können Sie nicht mehr vergeben. Ist bereits ein DB mit gleicher Nummer im Arbeitsspeicher und/oder im Ladespeicher vorhanden oder liegt der DB als kopierte Variante vor, wird die Anweisung beendet und eine Fehlerinformation erzeugt.

Hinweis

Ob bereits ein DB mit gleicher Nummer existiert, können Sie mit der Anweisung "TEST_DB (Seite 5765)" ermitteln.

Der DB wird mit dem Inhalt des Datenbereichs beschrieben, auf den der Parameter SRCBLK (source block) zeigt. Dieser Datenbereich muss ein DB oder ein Bereich aus einem DB sein.

Um Datenkonsistenz zu gewährleisten, dürfen Sie diesen Datenbereich während der Bearbeitung von "CREA_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Ein DB mit den Attribut READ_ONLY kann nur per "CREA_DBL" erzeugt und initialisiert werden.

"CREA_DBL" verändert nicht die Prüfsumme des Anwenderprogramms.

Arbeitsweise

Die Anweisung "CREA_DBL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "CREA_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CREA_DBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung	
REQ	Input	BOOL	E, A, M, D, L	REQ = 1: Anforderung zum Erzeugen des DB	
LOW_LIMIT	Input	WORD	E, A, M, D, L	Untergrenze des Bereichs, aus dem "CREA_DBL" Ihrem DB eine Nummer vergibt	
UP_LIMIT	Input	WORD	E, A, M, D, L	Obergrenze des Bereichs, aus dem "CREA_DBL" Ihrem DB eine Nummer vergibt	
COUNT	Input	WORD	E, A, M, D, L	Der Zählwert gibt die Anzahl der Datenbytes an, die Sie für Ihren DB reservieren wollen. Sie müssen hier eine gerade Anzahl von Bytes angeben.	
ATTRIB	Input	BYTE	E, A, M, D, L	DB-Eigenschaften:	
				Bit 0 = 1:	UNLINKED: Der DB ist nur im Ladespeicher.
				Bit 1 = 1:	READ_ONLY: Der DB ist schreibgeschützt.
				Bit 2 = 1:	NON_RETAIN: Der DB ist nicht remanent.
				Bit 3 bis 7:	reserviert
SRCBLK	Input	ANY	D	Zeiger auf den Datenbaustein, mit dessen Werten der zu erzeugende Datenbaustein initialisiert wird	
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation	
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: Der Vorgang ist noch nicht beendet.	
DB_NUM	Output	WORD	E, A, M, D, L	Nummer des erzeugten DB	

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Beschreibung
0000	kein Fehler
0081	Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden mit 0 aufgefüllt.
7000	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".
7001	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1"
8081	Der Quellbereich ist größer als der Zielbereich. **Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.**
8091	Sie haben "CREA_DBL" geschachtelt aufgerufen.
8092	Die Funktion "Erzeugen eines DB" ist momentan nicht durchführbar, weil • die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist • die maximale Anzahl an Bausteinen auf Ihrer CPU bereits erreicht ist
8093	Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.
8094	Beim Parameter ATTRIB wurde ein noch nicht unterstütztes Attribut angegeben.
80A1	Fehler bei der DB-Nummer: • die Nummer ist 0 oder liegt nicht im gültigen Bereich der CPU (CPU15xx: 60001 ... 60999) • die Nummer überschreitet die CPU-spezifische DB-Anzahl • Untergrenze > Obergrenze
80A2	Fehler bei der DB-Länge: • die Länge ist "0" • die Länge ist eine ungerade Zahl • die Länge ist größer, als die CPU zulässt
80B1	es ist keine DB-Nummer frei
80B2	zu wenig Arbeitsspeicher
80BB	nicht genügend Ladespeicher
80C0	Das Ziel wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.
80C3	Die maximale Anzahl gleichzeitig aktiver "CREA_DBL"-Anweisungen ist momentan bereits erreicht.
8xyy	allgemeine Fehlercodes, z. B.: • Quell-DB nicht oder nur als kopierte Variante vorhanden • Quellbereich in DB nicht vorhanden Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

TEST_DB: Datenbaustein testen

Beschreibung

Mit der Anweisung erhalten Sie bei S7-300 Informationen über einen im Arbeits- oder Ladespeicher der CPU liegenden Datenbaustein, bei S7-400 Informationen über einen im Arbeitsspeicher der CPU liegenden Datenbaustein. Die Anweisung ermittelt für den ausgewählten DB die Anzahl der Datenbytes und überprüft, ob der DB schreibgeschützt ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TEST_DB":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DB_NUMBER	Input	WORD	E, A, M, D, L oder Konstante	Nummer des zu überprüfenden DB
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
DB_LENGTH	Output	WORD	E, A, M, D, L	Anzahl der Datenbytes, die der ausgewählte DB enthält.
WRITE_PROT	Output	BOOL	E, A, M, D, L	Information über die Schreibschutzkennung des ausgewählten DB ("1" bedeutet schreibgeschützt).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
80A1	Fehler beim Eingangsparameter DB_NUMBER: Der gewählte Aktualparameter • hat den Wert "0" • ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.
80B1	Der DB mit der angegebenen Nummer ist auf der CPU nicht vorhanden.
80B2	Der DB wurde mit dem Schlüsselwort UNLINKED erzeugt.
8xyy	allgemeine Fehlerinformation, Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.3.11 Tabellenfunktionen

ATT: Wert in Tabelle eintragen

Beschreibung

Die Anweisung ergänzt den Parameter DATA als nächsten Eintrag in einer Tabelle und erhöht die Anzahl der Einträge um eins. Die Tabelle besteht aus Wörtern. Mit der Anweisung können Sie Tabelleneinträge erstellen, die von "FIFO (Seite 5767)" und "LIFO (Seite 5771)" verwendet werden.

- Der erste Eintrag einer "FIFO (Seite 5767)"- oder "LIFO (Seite 5771)"-Tabelle gibt die maximale Länge der Tabelle an.
- Der zweite Eintrag einer Tabelle gibt die Anzahl der vorhandenen Einträge an.
- Der dritte Eintrag der Tabelle enthält das erste Wort an Daten.

Hinweis

Beim Erstellen der Tabelle müssen Sie die ersten beiden Einträge initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	Input	WORD	E, A, M, D, L, P oder Konstante	Daten, die in die Tabelle eingegeben werden sollen.
TABLE	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der "FIFO"- bzw. "LIFO"-Tabelle.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Wenn die Anzahl der Einträge gleich oder größer als die Tabellenlänge ist, dann werden die Daten nicht in die Tabelle eingetragen und der Signalzustand von "BIE" wird auf "0" gesetzt.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "ATT" ausgeführt. In diesem Beispiel wird DATA als fünfter Eintrag in die Tabelle eingetragen, und die Anzahl der Einträge erhöht sich von 4 auf 5.

Wenn die Anweisung "ATT" fehlerfrei ausgeführt wird, dann werden die Signalzustände von "ENO" und A 0.0 auf "1" gesetzt.

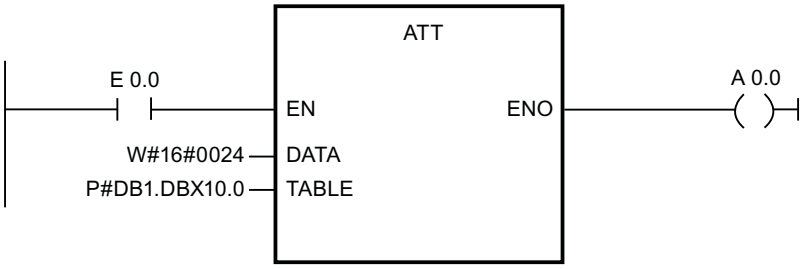

Vor der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0004 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0000 DBW24 = W#16#0000

Nach der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000

FIFO: Ersten Wert der Tabelle ausgeben

Beschreibung

Die Anweisung gibt den ältesten Eintrag der "FIFO"-Tabelle als Funktionswert aus. Die Anzahl der Einträge wird um eins verringert. Wenn noch Einträge in der Tabelle vorhanden sind, werden diese nach unten geschoben. Die "FIFO"-Tabelle besteht aus Wörtern. Sie können mit "ATT (Seite 5766)" Werte in die "FIFO"-Tabelle eintragen.

- Der erste Eintrag einer Tabelle gibt die maximale Länge der Tabelle an.
- Der zweite Eintrag einer Tabelle gibt die Anzahl der vorhandenen Einträge an.
- Der dritte Eintrag in der Tabelle enthält das erste Wort an Daten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FIFO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TABLE	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der "FIFO"-Tabelle.
RET_VAL	Return	WORD	E, A, M, D, L, P	Der älteste Eintrag aus der "FIFO"-Tabelle.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn die "FIFO"-Tabelle leer ist (Anzahl der Einträge = 0), dann wird RET_VAL nicht verändert und der Signalzustand von "BIE / ENO" wird auf "0" gesetzt.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird "FIFO" ausgeführt. In diesem Beispiel wird der älteste Eintrag der Tabelle als Funktionswert (MW 2.0) ausgegeben. Die Anzahl der Einträge reduziert sich von 5 auf 4, und die übrigen Einträge werden in der Tabelle nach unten geschoben.

Wenn "FIFO" fehlerfrei ausgeführt wird, dann werden die Signalzustände von "BIE / ENO" und A 0.0 auf "1" gesetzt.

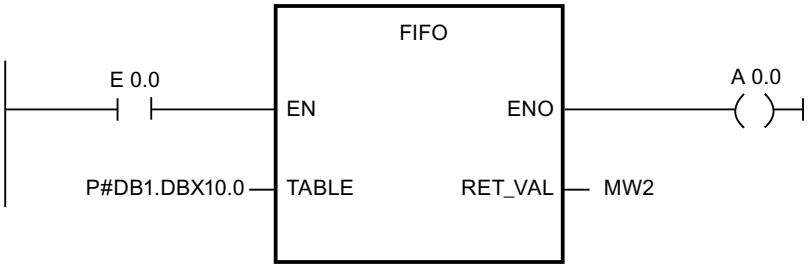

Vor der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0000

Nach der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0004 DBW14 = W#16#0029 DBW16 = W#16#0090 DBW18 = W#16#0002 DBW20 = W#16#0024 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0012

TBL_FIND: Wert in Tabelle suchen

Beschreibung

Mit der Anweisung können Sie nach Mustern im Speicher suchen. Die Anweisung "TBL_FIND" führt einen Vergleich (Parameter CMD) zwischen dem Quellmuster (Parameter PATRN) und den Einträgen der Quelltabelle (Parameter SRC) aus. Gesucht wird der nächste Eintrag (ab dem Eintrag, der von dem Parameter INDX angegeben wird) in der Tabelle, der den Vergleichskriterien entspricht. Die Nummer des Eintrags wird im Parameter INDX abgelegt. Wird keine Entsprechung gefunden, dann zeigt der Parameter INDX über das Ende der Tabelle hinaus und der Ausgang von der Anweisung "TBL_FIND" wird ausgeschaltet.

- Wenn CMD = 1 ist, dann sucht die Anweisung "TBL_FIND" nach dem ersten Wert, der dem Wert am Parameter PATRN entspricht.
- Wenn CMD = 2 ist, dann sucht die Anweisung "TBL_FIND" nach dem ersten Wert, der dem Wert am Parameter PATRN nicht entspricht.
- Der erste Eintrag in der Tabelle gibt die maximale Länge der Tabelle an.
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.

Hinweis

Sie müssen den ersten Eintrag (Tabellenlänge) in der Tabelle initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TBL_FIND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der Tabelle.
PATRN	Input	*Pointer	E, A, M, D	Zeigt auf das Muster, das gesucht werden soll.
CMD	Input	BYTE	E, A, M, D, L, P	Gibt den Befehl an: • B#16#01 = gleich • B#16#02 = ungleich

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an: B#16#02 = BYTE B#16#04= WORD B#16#05=INT B#16#06=DWORD B#16#07=DINT B#16#08=REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
INDX	InOut	WORD	E, A, M, D, L	Index der Tabelle, der folgende Informationen angibt: • Eingang: Nummer des Eintrages, bei dem die Suche begonnen werden soll. • Ausgang: Nummer des Eintrages, der dem vorgegebenen Muster entspricht.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, dann werden die Tabellenwerte nicht verändert. Der Signalzustand von "BIE / ENO" wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0007	Der Wert des Parameters INDX ist höher als der Wert für die Tabellenlänge.
W#16#0008	Es wurde keine Entsprechung gefunden.
W#16#0009	Der Parameter E_TYPE und/oder CMD ist/sind ungültig.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "TBL_FIND" ausgeführt. In diesem Beispiel werden die Daten in der Tabelle in Wörtern gespeichert, die an der Adresse beginnen, auf die der Parameter SRC zeigt, weil der Parameter E_TYPE "4" ist. Diese Wörter werden mit dem Muster "5555" verglichen, das in der Adresse gespeichert wird, die von dem Parameter PATRN angegeben wird. Weil CMD = 1 ist, wird der erste Wert in dem Parameter SRC gesucht, der mit dem Muster übereinstimmt. Der Parameter INDX zeigt auf die Adresse, an der die Suche beginnen soll. Nach der Bearbeitung der Operation gibt der Parameter INDX die Nummer des Eintrags in der Tabelle an, in dem eine Entsprechung für das Muster gefunden wurde.

Wenn die Anweisung "TBL_FIND" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" gesetzt, und RET_VAL wird auf den Wert W#16#0000 gesetzt.

Die Anweisung "MOVE" dient lediglich dazu, den Wert von "MW2" in jedem Zyklus zurückzusetzen.

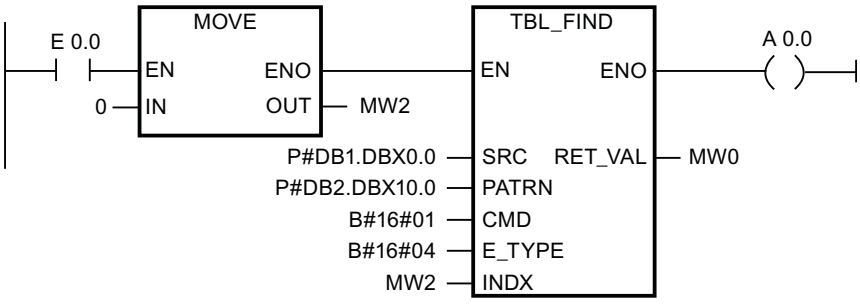

Vor der Bearbeitung:

SRC (Tabellenlänge)	DBW0 = W#16#0004 DBW2 = W#16#1111 DBW4 = W#16#3333 DBW6 = W#16#5555 DBW8 = W#16#7777	Der erste Tabelleneintrag gibt die Länge der Suchoperation an. Der Wert 0004 gibt an, dass die nächsten vier Einträge durchsucht werden.
INDX	MW2 = W#16#0000	INDX wird von DW0 (vor der Bearbeitung) nach DW6 (nach der Bearbeitung) übertragen, weil das Muster von DW6 gleich dem Muster von 5555 von PATRN ist.
PATRN	DBW10 = W#16#5555	

Nach der Bearbeitung:

INDX	MW2 = W#16#0003	

LIFO: Letzten Wert der Tabelle ausgeben

Beschreibung

Die Anweisung gibt den neuesten Eintrag der "LIFO"-Tabelle als Funktionswert aus. Die Anzahl der Einträge wird um eins verringert. Die "LIFO"-Tabelle besteht aus Wörtern. Sie können mit der Anweisung "ATT (Seite 5766)" Werte in die "LIFO"-Tabelle eintragen.

- Der erste Eintrag der Tabelle gibt die maximal zulässige Anzahl der Tabelleneinträge an (Tabellenlänge).
- Der zweite Eintrag einer Tabelle gibt die Anzahl der vorhandenen Einträge an.
- Der dritte Eintrag der Tabelle enthält das erste Wort an Daten.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LIFO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TABLE	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der "LIFO"-Tabelle.
RET_VAL	Return	WORD	E, A, M, D, L, P	Der neueste Eintrag aus der "LIFO"-Tabelle.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn die "LIFO"-Tabelle leer ist (Anzahl der Einträge = 0), dann wird RET_VAL nicht verändert und der Signalzustand von "BIE / ENO" wird auf "0" gesetzt.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "LIFO" ausgeführt. In diesem Beispiel wird der neueste Eintrag der "LIFO"-Tabelle als Funktionswert (MW 2.0) ausgegeben. Die Anzahl der Einträge reduziert sich von 5 auf 4.

Wenn "LIFO" fehlerfrei ausgeführt wird, dann werden die Signalzustände von "ENO" und A 0.0 auf "1" gesetzt.

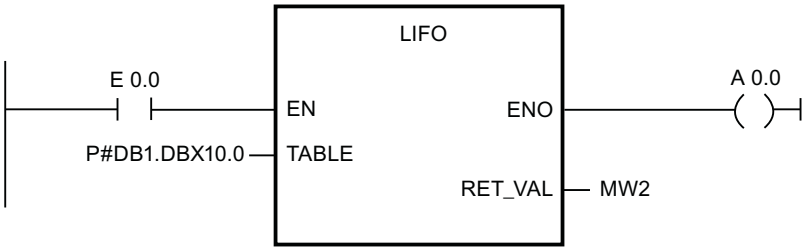

Vor der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0000

Nach der Bearbeitung:

TABLE (Tabellenlänge)	DBW10 = W#16#0006
Anzahl der Einträge	DBW12 = W#16#0004 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0024

TBL: Tabellenoperation ausführen

Beschreibung

Die Anweisung führt die angegebene Operation (Parameter CMD) mit der Quelltabelle aus und speichert das Ergebnis in dem gleichen Eintrag in der Tabelle.

- Der erste Eintrag der Tabelle gibt die maximal zulässige Anzahl der Tabelleneinträge an (Tabellenlänge).
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.
- Wenn der Parameter E_TYPE auf REAL gesetzt wird, dann ist der Wert von CMD für das Einerkomplement ungültig.

Hinweis

Beim Erstellen der Tabelle müssen Sie den ersten Eintrag initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der Tabelle.
CMD	Input	BYTE	E, A, M, D, L, P	Gibt die Operation an, die ausgeführt werden soll. Die folgenden Operationen sind gültig: B#16#03 = Einerkomplement B#16#04 = Löschen B#16#05 = Negieren B#16#06 = Quadratwurzel

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Folgende Datentypen sind gültig: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn der Parameter CMD oder der Parameter E_TYPE ungültig ist und/oder der Parameter CMD und der Parameter E_TYPE nicht übereinstimmen, dann werden die Werte in der Tabelle nicht verändert. Der Signalzustand von BIE / ENO wird auf "0" und RET_VAL wird auf den Wert W#16#0008 gesetzt.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "TBL" ausgeführt. In diesem Beispiel zeigt der Parameter SRC auf die Adresse in dem Datenbaustein, die von der Operation bearbeitet werden soll. Da der Parameter E_TYPE = 4 ist, werden die Daten in der Tabelle in den Wörtern gespeichert, die an der Adresse beginnen, die von dem Parameter SRC angegeben wird. Weil CMD = 4 ist (Löschen), werden alle Wörter in der Tabelle gelöscht (auf "0" gesetzt), wenn die Anweisung "TBL" ausgeführt wird. Da in dem ersten Eintrag in der Tabelle eine Länge von "5" angegeben wird, werden die nächsten fünf Adressen in der Tabelle gelöscht.

Wenn die Anweisung "TBL" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" und RET_VAL auf den Wert W#16#0000 gesetzt.

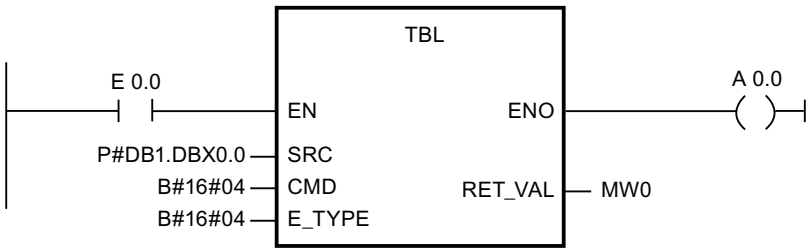

Vor der Bearbeitung:

SRC (Tabellenlänge)	DBW0 = W#16#0005 DBW2 = W#16#2000 DBW4 = W#16#3000 DBW6 = W#16#4000 DBW8 = W#16#5000 DBW10 = W#16#6000

Nach der Bearbeitung:

SRC (Tabellenlänge)	DBW0 = W#16#0005 DBW2 = W#16#0000 DBW4 = W#16#0000 DBW6 = W#16#0000 DBW8 = W#16#0000 DBW10 = W#16#0000

TBL_WRD: Wert aus der Tabelle kopieren

Beschreibung

Die Anweisung "TBL_WRD" (Wert aus der Tabelle kopieren) kopiert den Eintrag, der von dem Parameter INDX angegeben wird, aus der Tabelle SRC zu dem Ausgang DEST. Dabei wird der Wert von INDX erhöht, sofern der Wert kleiner ist als die maximale Länge, die in dem ersten Wort der Tabelle angegeben wird (SRC[0]). Wenn der Parameter INDX den letzten Tabelleneintrag angibt, dann wird das Bit des Ausgangs nach der Bearbeitung auf "0" gesetzt.

- Der erste Eintrag der Tabelle gibt die maximal zulässige Anzahl der Tabelleneinträge an (Tabellenlänge).
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.

Hinweis

Beim Erstellen der Tabelle müssen Sie den ersten Eintrag initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TBL_WRD":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse der Tabelle.
DEST	Input	*Pointer	E, A, M, D	Zeigt auf die Anfangsadresse des Ziels.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Folgende Datentypen sind gültig: • B#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
Q	Output	BOOL	A, M, D, L	Gibt den Wert "0" an, wenn die Variable INDX beim Aufruf von "TBL_WRD" den letzten Wert der Tabelle enthält.
INDX	InOut	WORD	E, A, M, L	Nummer des Eintrags, der kopiert werden soll.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, wird "TBL_WRD" nicht ausgeführt. Der Signalzustand von ENO wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0007	Der Index ist "0".
W#16#0008	E_TYPE ist ungültig.
W#16#0009	Der Index liegt außerhalb des Tabellenbereichs.

Beispiel

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird "TBL_WRD" ausgeführt. Da in diesem Beispiel E_TYPE = 4 ist, wird ein Datenwort, das in der Tabelle in dem Eintrag gespeichert wird, auf den der Parameter SRC zeigt, in den Eintrag kopiert, der von dem Parameter DEST angegeben wird. Der Wert von INDX zeigt auf den Eintrag in der Tabelle, der kopiert werden soll. Nach der erfolgreichen Ausführung von "TBL_WRD" wird der Wert von INDX automatisch um einen Eintrag erhöht. Enthält der Parameter INDX nicht den letzten Tabelleneintrag, wenn "TBL_WRD" aufgerufen wird, dann wird der Ausgang am Parameter ENO nach der Bearbeitung auf "1" gesetzt.

Wenn "TBL_WRD" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO auf "1" und RET_VAL auf den Wert W#16#0000 gesetzt.

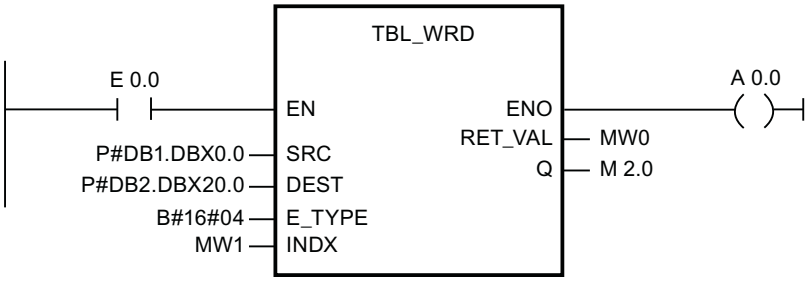

Vor der Bearbeitung:

SRC (Tabellenlänge)	DBW0 = W#16#0004 DBW2 = W#16#2000 DBW4 = W#16#3000 DBW6 = W#16#4000 DBW8 = W#16#5000
INDX	MW1 = W#16#0001
DEST	DBW20 = W#16#0000

Vor der Bearbeitung:

INDX	MW1 = W#16#0002
DEST	DBW20 = W#16#2000

WRD_TBL: Wert logisch mit Tabellenelement verknüpfen und speichern

Beschreibung

Die Anweisung führt die angegebene Operation (Parameter CMD) mit dem Quellelement und dem Eintrag der Tabelle an dem Versatz aus, der von dem Parameter INDX angegeben wird. Die Operation wird nur ausgeführt, solange der Parameter INDX kleiner ist als die Länge der Tabelle, die in dem ersten Wort in der Tabelle gespeichert wird.

- Der erste Eintrag der Tabelle gibt die maximal zulässige Anzahl der Tabelleneinträge an (Tabellenlänge).
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.
- Ist der Parameter E_TYPE vom Datentyp REAL, dann kann es sich bei dem Parameter CMD nur um "Übertragen" handeln.

Hinweis

Beim Erstellen der Tabelle müssen Sie den ersten Eintrag initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRD_TBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SRC	Input	*Pointer	E, A, M, D	Zeigt auf die Quelldaten
TABLE	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der Tabelle.
CMD	Input	BYTE	E, A, M, D, L, P	Zeigt auf die Operation, die ausgeführt werden soll. Folgende Operationen sind gültig: B#16#0E = Übertragen B#16#07 = UND-Verknüpfung B#16#08 = ODER-Verknüpfung B#16#09 = EXKLUSIV ODER-Verknüpfung
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Folgende Datentypen sind gültig: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird
Q	Output	BOOL	A, M, D, L	Gibt den Wert "0" an, wenn die Variable am Parameter INDX die Nummer des letzten Tabelleneintrags enthält.
INDX	InOut	WORD	E, A, M, D, L	Nummer des Eintrags, mit dem die Operation ausgeführt werden soll.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, wird die Anweisung "WRD_TBL" nicht ausgeführt. Der Signalzustand von ENO wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0007	Der Index ist 0.
W#16#0008	Parameter CMD oder Parameter E_TYPE ist ungültig bzw. Parameter CMD und Parameter E_TYPE stimmen nicht überein.
W#16#0009	Index liegt außerhalb des Tabellenbereichs.

Beispiel (KOP)

Wenn der Signalzustand am Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "WRD_TBL" ausgeführt. Da der Parameter E_TYPE in diesem Beispiel "6" ist, wird das Doppelwort der Daten in der Tabelle an der Adresse gespeichert, auf die der Parameter TABLE zeigt. Das erste Wort der Tabelle zeigt an, dass die Tabelle drei Doppelwörter enthält. Der Wert von INDX zeigt auf den Tabelleneintrag, der bearbeitet werden soll. Da CMD = 8 ist, wird eine ODER-Verknüpfung mit dem Wert ausgeführt, auf den der Parameter INDX zeigt. Weil INDX = 2 ist, wird das zweite Doppelwort (66665544) mit dem Wert, auf den der Parameter SRC (11111111) zeigt, durch ODER verknüpft. Nach der Bearbeitung der Operation wird das Ergebnis der ODER-Verknüpfung (77775555) zurück in die Tabelle geschrieben, und der Wert von INDX wird automatisch um einen Eintrag erhöht. Zeigt der Parameter INDX auf den letzten Eintrag einer Tabelle, dann wird das Bit des Ausgangs "A" nach der Bearbeitung der Operation auf "0" gesetzt. In diesem Beispiel enthält der Wert von INDX nicht den letzten Eintrag in dieser Tabelle, deshalb ist der Ausgang "A" nach der Bearbeitung "1".

Wenn die Anweisung "WRD_TBL" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" gesetzt, und RET_VAL wird auf den Wert "W#16#0000" gesetzt.

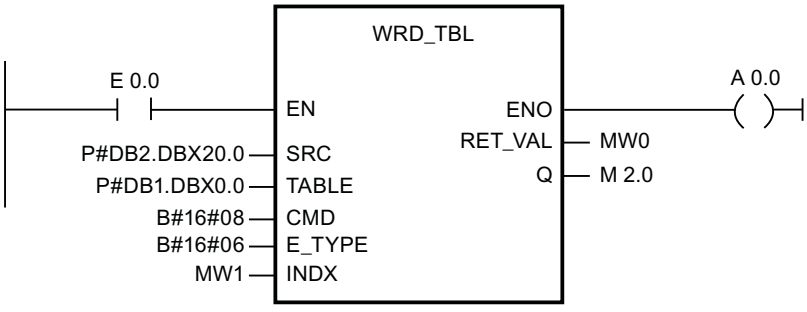

Vor der Bearbeitung:

SRC	DBD20 = DW#16#11111111
TABLE (Tabellenlänge)	DBW0 = W#16#0003 DBD2 = DW#16#99998877 DBD6 = DW#16#66665544 DBD10 = DW#16#33332222
INDX	MW1 = W#16#0002

Nach der Bearbeitung:

TABLE (Tabellenlänge)	DBW0 = W#16#0003 DBD2 = DW#16#99998877 DBD6 = DW#16#77775555 DBD10 = DW#16#33332222
INDX	MW1 = W#16#0003

DEV: Standardabweichung berechnen

Beschreibung

Die Anweisung errechnet die Standardabweichung von einer Gruppe von Werten, die in einer Tabelle (Anweisung "TBL (Seite 5773)") gespeichert sind. Das Ergebnis wird in OUT abgelegt. Die Standardabweichung wird nach folgender Formel berechnet:

$$\text{Standardabweichung} = \sqrt{\frac{(N * QuSumme) - Summe^2}{N * (N - 1)}}$$

Dabei gilt:

- Summe = Summe der Werte in der Anweisung "TBL (Seite 5773)"
- N = Anzahl der Werte in der Anweisung "TBL (Seite 5773)"
- QuSumme = Summe aller Werte in der Anweisung "TBL (Seite 5773)" zum Quadrat

Alle Berechnungen verwenden IEEE-Gleitpunktwerte, wobei die Datentypen, sofern erforderlich, automatisch von der Anweisung "DEV" umgewandelt werden.

- Der erste Eintrag in der Tabelle gibt die Anzahl der Einträge in der Tabelle (Tabellenlänge) an.
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.
- Die Größe der Tabelleneinträge und der errechnete Wert (Parameter OUT) werden von dem Parameter E_TYPE festgelegt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DEV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TBL	Input	*Pointer	E, A, M, D	**Zeigt auf die Anfangsadresse der Tabelle mit den Werten.
OUT	Input	*Pointer	E, A, M, D	**Zeigt auf die Adresse, die den errechneten Wert für die Standardabweichung enthält.
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Die folgenden Datentypen sind gültig: B#16#05 = INT B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung ** Quell- und Zieldatenbaustein müssen identisch sein				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, wird die Anweisung "DEV" nicht ausgeführt. Der Signalzustand von BIE / ENO wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0001	Für einen Parameter von "DEV" wurde ein ungültiger Speicherbereich angegeben.
W#16#0002	Der Parameter E_TYPE ist ungültig.
W#16#0004	Die Tabellenlänge beträgt Null.

Beispiel (KOP)

Wenn der Signalzustand von Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "DEV" ausgeführt. In diesem Beispiel gibt es fünf Tabellenwerte. Dies gibt das erste Wort in der Tabelle an. Die Tabellenwerte sind vom Datentyp REAL. Dies gibt der Parameter E_TYPE an.

Wenn "DEV" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" und RET_VAL auf "W#16#0000" gesetzt.

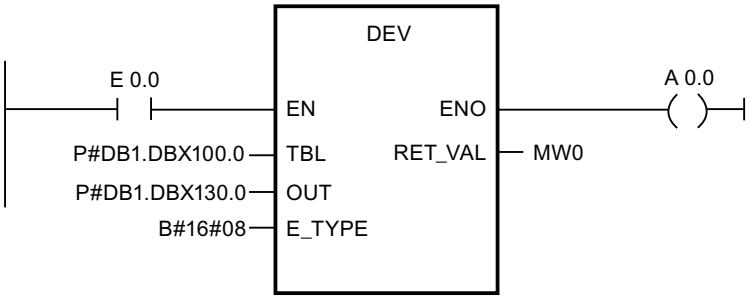

Vor der Bearbeitung:

TBL (Tabellenlänge)	DBW100 = W#16#0005 DBD102 = 2.0 DBD106 = 4.0 DBD110 = 8.0 DBD114 = 16.0 DBD118 = 32.0
OUT	DBD130 = 0.0

Nach der Bearbeitung:

OUT	DBD130 = 12.19836055

CDT: Korrelierte Datentabellen

Beschreibung

Die Anweisung vergleicht einen Eingabewert (Parameter IN) mit einer bereits vorhandenen Tabelle mit Eingabewerten (Parameter IN_TBL) und sucht den ersten Wert, der größer oder gleich dem Eingabewert ist. Mithilfe des Index des gefundenen Werts wird der Wert dann zu dem entsprechenden Ausgabewert (Parameter OUT) in die Tabelle der Ausgabewerte (Parameter OUT_TBL) kopiert.

- Die Werte in der Eingabetabelle müssen in aufsteigender Reihenfolge sortiert sein. Das heißt der kleinste Wert befindet sich im ersten Tabelleneintrag, der größte Wert befindet sich im letzten Tabelleneintrag.
- Die Größe des Eingabewerts, die Tabellenwerte und die Ausgabewerte werden von dem Parameter E_TYPE festgelegt.
- Der erste Eintrag in der Tabelle gibt die Anzahl der Tabelleneinträge (Tabellenlänge) an.
- Der zweite Eintrag der Tabelle enthält den ersten Wert in der Tabelle.
- Die Anzahl der Einträge muss in beiden Tabellen identisch und größer als Null sein.

Hinweis

Beim Erstellen der Tabellen müssen Sie jeweils den ersten Eintrag initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CDT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IN_TBL	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der Eingangstabelle.
OUT_TBL	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der Ausgangstabelle.
IN	Input	*Pointer	E, A, M, D	Zeigt auf die Eingangstabelle.
OUT	Input	*Pointer	E, A, M, D	Zeigt auf die Ausgangstabelle.
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Die folgenden Datentypen sind gültig: B#16#05 = INT B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, wird CDT nicht ausgeführt. Der Signalzustand von BIE / ENO wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0001	Für einen Parameter von CDT wurde ein ungültiger Speicherbereich angegeben.
W#16#0002	Der Parameter E_TYPE ist ungültig.
W#16#0003	Die Längen der Eingangs- und Ausgangstabellen stimmen nicht überein.
W#16#0004	Die Tabelle hat eine Länge von Null.
W#16#0007	In dem Parameter IN_TBL befindet sich kein Wert, der größer als oder gleich dem Eingabewert ist.

Beispiel (KOP)

Wenn der Signalzustand von Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "CDT" ausgeführt. In diesem Beispiel enthalten sowohl die Parameter IN_TBL und OUT_TBL jeweils fünf Tabelleneinträge. Dies wird von dem ersten Wort der jeweiligen Tabelle angegeben. Bei dem Datentyp der Tabellenwerte handelt es sich um INTEGER. Diesen gibt der Parameter E_TYPE an. Der Wert von dem Parameter IN ist 22. Der Wert von dem Parameter IN_TBL, der größer als oder gleich "22" ist, ist "64", der den Index "5" hat. Der entsprechende Wert in OUT_TBL ist "25". Deshalb wird der Wert "25" in den Parameter OUT geschrieben.

Wenn die Anweisung "CDT" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" und RET_VAL auf W#16#0000 gesetzt.

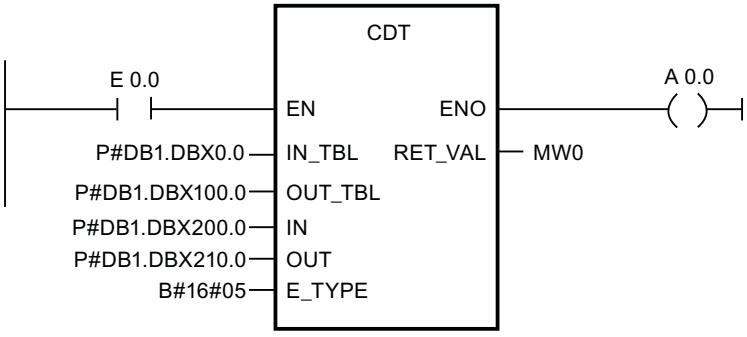

Vor der Bearbeitung:

IN_TBL (Tabellenlänge)	DBW0 = W#16#0005 DBW2 = 2 DBW4 = 4 DBW6 = 8 DBW8 = 16 DBW10 = 64
OUT_TBL (Tabellenlänge)	DBW100 = W#16#0005 DBW102 = 5 DBW104 = 10 DBW106 = 15 DBW108 = 20 DBW110 = 25
IN	DBW200 = 22
OUT	DBW210 = 0

Nach der Bearbeitung:

OUT DBW210 = 25

TBL_TBL: Tabellen verknüpfen

Beschreibung

Die Anweisung führt den am Parameter CMD angegebenen Befehl mit den entsprechenden Einträgen der beiden Quelltabellen (Parameter TBL1 und TBL2) aus und schreibt das Ergebnis in die entsprechenden Einträge in der Zieltabelle (DEST_TBL).

- Die Datentypen INT, DINT und REAL sind nur für arithmetische Operationen gültig.
- Der erste Eintrag in der Tabelle gibt die Anzahl der Tabelleneinträge (Tabellenlänge) an.
- Die Anzahl der Einträge aller Tabellen muss gleich oder größer als Null sein.

Hinweis

Beim Erstellen der Tabellen müssen Sie jeweils den ersten Eintrag initialisieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TBL_TBL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TBL1	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der ersten Quelltabelle.
TBL2	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der zweiten Quelltabelle.
DEST_TBL	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der Zieltabelle.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CMD	Input	BYTE	E, A, M, D, L, P	Gibt den auszuführenden Befehl an. Die folgenden Befehle sind gültig: B#16#07 = UND-Verknüpfung B#16#08 = ODER-Verknüpfung B#16#09 = EXKLUSIV ODER-Verknüpfung B#16#0a = Addieren B#16#0b = Subtrahieren B#16#0c = Multiplizieren B#16#0d = Dividieren
E_TYPE	Input	BYTE	E, A, M, D, L, P	Gibt den Datentyp der Tabelleneinträge an. Gültig sind die folgenden Datentypen: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Bedingungen auftritt, wird die Anweisung "TBL_TBL" nicht ausgeführt. Der Signalzustand von BIE / ENO wird auf "0" und der Rücksprungwert entsprechend gesetzt:

RET_VAL	Erläuterung
W#16#0001	Für einen Parameter von der Anweisung "TBL_TBL" wurde ein ungültiger Speicherbereich angegeben.
W#16#0002	Der Parameter E_TYPE ist ungültig.
W#16#0003	Die Längen der Eingangs- und der Ausgangstabellen stimmen nicht überein.
W#16#0004	Die Tabelle hat eine Länge von Null.
W#16#0005	Der Parameter E_TYPE und CMD stimmen nicht überein.
W#16#0006	Der Parameter CMD ist ungültig

Beispiel (KOP)

Wenn der Signalzustand von Eingang E 0.0 = 1 ist (aktiviert), dann wird die Anweisung "TBL_TBL" ausgeführt. In diesem Beispiel enthalten alle Tabellen jeweils drei Tabelleneinträge. Dies wird von dem ersten Wort der jeweiligen Tabelle angegeben. Bei dem Datentyp der Tabellenwerte handelt es sich um WORD. Dies gibt E_TYPE an. CMD gibt den Befehl an, mit dem TBL1 und TBL2 verknüpft werden sollen.

Wird die Anweisung "TBL_TBL" fehlerfrei ausgeführt, dann werden die Signalzustände von ENO und A 0.0 auf "1" gesetzt und RET_VAL wird auf den Wert W#16#0000 gesetzt.

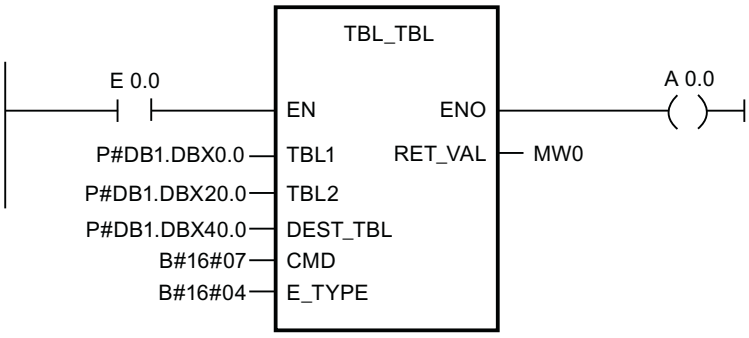

Vor der Bearbeitung:

TBL1 (Tabellenlänge)	DBW0 = W#16#0003 DBW2 = W#16#00FF DBW4 = W#16#FF00 DBW6 = W#16#FFFF
TBL2 (Tabellenlänge)	DBW20 = W#16#0003 DBW22 = W#16#1111 DBW24 = W#16#2222 DBW26 = W#16#3333
DEST_TBL (Tabellenlänge)	DBW40 = W#16#0003 DBW42 = W#16#0000 DBW44 = W#16#0000 DBW46 = W#16#0000

Nach der Bearbeitung:

DST_TBL (Tabellenlänge)	DBW40 = W#16#0003 DBW42 = W#16#0011 DBW44 = W#16#2200 DBW46 = W#16#3333

PACK: Sammle/Verteile Tabellendaten

Beschreibung

Die Anweisung überträgt Daten, die sich zwischen beliebigen Adressen und einer Tabelle befinden. Die Übertragungsrichtung wird von dem Parameter DIR angegeben. Jede Operation der Anweisung "PACK" bearbeitet maximal fünf Bausteine mit Daten (P_DATA1 bis P_DATA5). Wenn der Parameter DIR den Wert "an" enthält, dann überträgt die Anweisung "PACK" Daten aus den Adressen an die angegebene Tabelle. Wenn der Parameter DIR den Wert "aus" enthält, dann werden die Daten aus der Tabelle an die angegebenen Adressen verteilt.

Folgende Richtlinien gelten beim Übertragen von Daten "an" eine Tabelle:

- Einzelne Bits (BOOL) werden in das nächste verfügbare Bit in der Tabelle übertragen.
- Datentypen mit 8 Bit werden in das nächste verfügbare Byte in der Tabelle übertragen. Wenn ein Byte in die Tabelle geschrieben wird, dann werden die nicht verwendeten Bits von dem vorherigen Byte mit Nullen belegt.
- Datentypen mit 16 und 32 Bit werden in das nächste verfügbare Wort in der Tabelle übertragen. Wenn ein Wort in die Tabelle geschrieben wird, dann werden die nicht verwendeten Bits in dem vorherigen Wort mit Nullen belegt.

Folgende Richtlinien gelten beim Übertragen von Daten "aus" einer Tabelle:

- Es dürfen keine Abschnitte in einer Tabelle ausgelassen werden.
- Alle angegebenen Daten vom Datentyp BOOL werden aus der Tabelle übertragen.
- Datentypen mit 8 Bit werden aus dem ersten verfügbaren Byte in der Tabelle übertragen. Deshalb sind nicht verwendete Bits aus dem vorherigen Byte in der Tabelle nicht in dem Byte enthalten, das aus der Tabelle übertragen wird.
- Datentypen mit 16 und 32 Bit werden aus dem ersten verfügbaren Wort in der Tabelle übertragen. Deshalb sind nicht verwendete Bits aus dem vorherigen Wort in der Tabelle nicht in dem Wort enthalten, das aus der Tabelle übertragen wird.

Folgende Datentypen sind für den Pointer ANY gültig und werden von "PACK" unterstützt: BOOL, WORD, INT, BYTE, DINT, REAL, CHAR, DWORD.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PACK":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
TABLE	Input	*Pointer	E, A, M, D	Zeigt auf den Beginn der Tabelle.
P_DATA1	Input	ANY	E, A, M, D	Zeigt auf den Beginn des Bausteins, dessen Daten übertragen werden sollen.
P_DATA2	Input	ANY	E, A, M, D	Zeigt auf den Beginn des Bausteins, dessen Daten übertragen werden sollen.
P_DATA3	Input	ANY	E, A, M, D	Zeigt auf den Beginn des Bausteins, dessen Daten übertragen werden sollen.
P_DATA4	Input	ANY	E, A, M, D	Zeigt auf den Beginn des Bausteins, dessen Daten übertragen werden sollen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
P_DATA5	Input	ANY	E, A, M, D	Zeigt auf den Beginn des Bausteins, dessen Daten übertragen werden sollen.
ERR_CODE	Output	WORD	E, A, M, D, L, P	Gibt den Wert W#16#0000 aus, wenn die Operation fehlerfrei ausgeführt wird.
DIR	Static	BOOL	E, A, M, D, L	Richtung der Übertragung. Folgende Signalzustände sind möglich: 0=an, 1=aus.
* Pointer im Doppelwortformat bei der bereichsübergreifenden, registerindirekten Adressierung				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Wenn eine der folgenden Fehlerbedingungen auftritt, dann wird die Anweisung "PACK" nicht ausgeführt. Der Signalzustand von BIE / ENO wird auf "0" und ERR_CODE entsprechend gesetzt:

ERR_CODE	Erläuterung
W#16#0001	Für einen Parameter von der Anweisung "PACK" wurde ein ungültiger Speicherbereich angegeben.
W#16#0002	Die Anweisung "E_TYPE" ist ungültig.

Beispiel (KOP)

Wenn der Signalzustand von Eingang E 0.0 = 1 ist (aktiviert), dann wird "PACK" ausgeführt. In diesem Beispiel werden vier Bausteine mit Daten "an" die Tabelle übertragen.

Wenn "PACK" fehlerfrei ausgeführt wird, dann werden die Signalzustände von ENO und A 0.0 auf "1" und ERR_CODE auf "W#16#0000" gesetzt.

Hinweis

Statische Parameter können Sie mithilfe des Datenbaustein-Editors initialisieren.

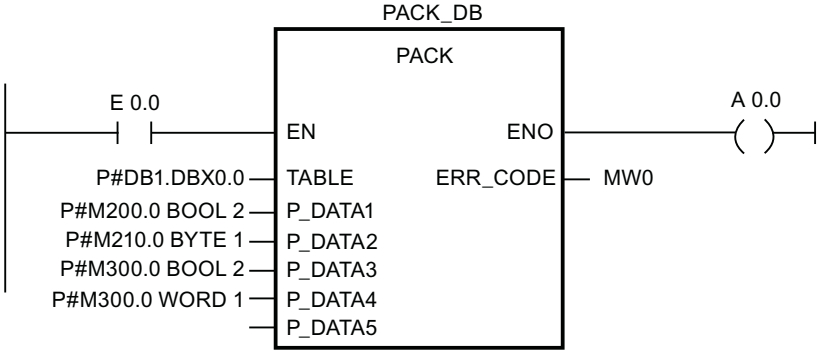

Vor der Bearbeitung:

TABLE	DBB0 = B#16#00 DBB1 = B#16#00 DBB2 = B#16#00 DBB3 = B#16#00 DBB4 = B#16#00 DBB5 = B#16#00
P_DATA1	M 200.0 = TRUE M 200.1 = TRUE
P_DATA2	MB210 = B#16#FF
P_DATA3	M 300.0 = TRUE M 300.1 = TRUE
P_DATA4	MW330 = W#16#FFFF
Instanz-DB PACK_DB	
DIR	DBX58.0 = FALSE

Nach der Bearbeitung:

TABLE	DBB0 = B#16#03 DBB1 = B#16#FF DBB2 = B#16#03 DBB3 = B#16#00 DBB4 = B#16#FF DBB5 = B#16#FF

4.2.3.12 Adressierung

GEO_LOG: Anfangsadresse einer Baugruppe ermitteln

Beschreibung

Vom Kanal einer Signalbaugruppe sei der zugehörige Baugruppensteckplatz der Baugruppe bekannt. Mit der Anweisung "GEO_LOG" ermitteln Sie daraus die zugehörige Anfangsadresse der Baugruppe, d. h. die kleinste E- oder A-Adresse.

Wenn Sie die Anweisung "GEO_LOG" auf Power-Module oder Module mit gepackten Adressen (ET 200S) anwenden, wird die Diagnoseadresse zurückgeliefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEO_LOG":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MASTER	Input	INT	E, A, M, D, L oder Konstante	Bereichskennung: • 0, falls sich der Steckplatz im zentralen Aufbau befindet: Baugruppenträger 0 bis 3 (bei S7-300) bzw. 0 bis 21 (bei S7-400) • 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet • 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet
STATION	Input	INT	E, A, M, D, L oder Konstante	• Nr. des Baugruppenträgers, falls Bereichskennung = 0 • Stationsnummer des Feldgeräts, falls Bereichskennung > 0
SLOT	Input	INT	E, A, M, D, L oder Konstante	Steckplatz-Nr.
SUBSLOT	Input	INT	E, A, M, D, L oder Konstante	Submodulsteckplatz (falls kein Submodul gesteckt werden kann, ist hier 0 anzugeben)
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
LADDR	Output	WORD	E, A, M, D, L	Anfangsadresse der Baugruppe Bit 15 von LADDR gibt an, ob eine Ein- (Bit 15 = 0) oder eine Ausgangsadresse (Bit 15 = 1) vorliegt.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8094	Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.
8095	Unzulässiger Wert beim Parameter STATION
8096	Unzulässiger Wert beim Parameter SLOT
8097	Unzulässiger Wert beim Parameter SUBSLOT
8099	Der Steckplatz ist nicht projektiert.

Fehlercode (W#16#...)	Erläuterung
809A	Die Submoduladresse für den ausgewählten Steckplatz ist nicht projektiert.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

LOG_GEO: Den zu einer logischen Adresse gehörenden Steckplatz ermitteln

Beschreibung

Mit der Anweisung ermitteln Sie den zu einer logischen Adresse gehörenden Baugruppensteckplatz sowie den Offset im Nutzdatenadressraum der Baugruppe.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG_GEO":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Irgendeine logische Adresse der Baugruppe In Bit 15 geben Sie an, ob es sich um eine Ein- (Bit 15 = 0) oder eine Ausgangsadresse (Bit 15= 1) handelt.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
AREA	Output	INT	E, A, M, D, L	Bereichskennung: Sie gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind.
MASTER	Output	INT	E, A, M, D, L	Bereichskennung: • 0, falls sich der Steckplatz im zentralen Aufbau befindet: Baugruppenträger 0 bis 3 (bei S7-300) bzw. 0 bis 21 (bei S7-400) • 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet • 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet
STATION	Output	INT	E, A, M, D, L	Nr. des Baugruppenträgers, falls: • Bereichskennung = 0 • Stationsnummer des Feldgeräts, falls Bereichskennung > 0
SLOT	Output	INT	E, A, M, D, L	Steckplatz-Nr.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SUBSLOT	Output	INT	E, A, M, D, L	Submodulnummer
OFFSET	Output	INT	E, A, M, D, L	Offset im Nutzdatenadressraum der zugehörigen Baugruppe

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter AREA

Wert von AREA	System	Bedeutung von MASTER, STATION, SLOT, SUBSLOT und OFFSET
0	S7-400	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. SUBSLOT: 0 OFFSET: Differenz zwischen logischer Adresse und logischer Basisadresse
1	S7-300	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. SUBSLOT: 0 OFFSET: Differenz zwischen logischer Adresse und logischer Basisadresse
2	PROFIBUS DP	MASTER: DP-Mastersystem-ID STATION: Stationsnummer SLOT: Steckplatz-Nr. in der Station SUBSLOT: 0 OFFSET: Offset im Nutzdatenadressraum der zugehörigen Baugruppe
	PROFINET IO	MASTER: PROFINET IO-System-ID STATION: Stationsnummer SLOT: Steckplatz-Nr. in der Station SUBSLOT: Submodulnummer OFFSET: Offset im Nutzdatenadressraum der zugehörigen Baugruppe
3	S5-P-Bereich	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. der Adaptionskapsel SUBSLOT: 0 OFFSET: Adresse im S5-x-Bereich
4	S5-Q-Bereich	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. der Adaptionskapsel SUBSLOT: 0 OFFSET: Adresse im S5-x-Bereich
5	S5-IM3-Bereich	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. der Adaptionskapsel OFFSET: Adresse im S5-x-Bereich
6	S5-IM4-Bereich	MASTER: 0 STATION: Baugruppenträger-Nr. SLOT: Steckplatz-Nr. der Adaptionskapsel SUBSLOT: 0 OFFSET: Adresse im S5-x-Bereich

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Angegebene logische Adresse ungültig
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

RD_LGADR: Sämtliche logischen Adressen einer Baugruppe ermitteln

Beschreibung

Sie gehen von einer logischen Adresse einer Baugruppe, eines zentralen Submoduls oder eines Submoduls bei PNIO aus. Mit der Anweisung ermitteln Sie alle vereinbarten logischen Adressen dieser Baugruppe bzw. dieses Submoduls. Die Zuordnung von logischen Adressen zu Baugruppen bzw. Submodulen haben Sie vorher projektiert. Die Anweisung "RD_LGADR" trägt die ermittelten logischen Adressen in den Parametern PEADDR bzw. PAADDR in aufsteigender Reihenfolge ein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD_LGADR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#00: Bit 15 von LADDR gibt an, ob Ein- (Bit 15=0) oder Ausgangsadresse (Bit 15=1) vorliegt. • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA)
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Eine logische Adresse
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
PEADDR	Output	ANY	E, A, M, D, L	Feld für die PE-Adressen, Feldelemente müssen vom Datentyp WORD sein.
PECOUNT	Output	INT	E, A, M, D, L	Anzahl der zurückgelieferten PE-Adressen
PAADDR	Output	ANY	E, A, M, D, L	Feld für die PA-Adressen, Feldelemente müssen vom Datentyp WORD sein.
PACOUNT	Output	INT	E, A, M, D, L	Anzahl der zurückgelieferten PA-Adressen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Angegebene logische Adresse ungültig oder unzulässiger Wert beim Parameter IOID
80A0	Fehler beim Ausgangsparameter PEADDR: Der Datentyp der Feldelemente ist nicht WORD. (Diesen Fehlercode gibt es nur bei S7-400 und bei der CPU 318.)
80A1	Fehler beim Ausgangsparameter PAADDR: Der Datentyp der Feldelemente ist nicht WORD. (Diesen Fehlercode gibt es nur bei S7-400 und bei der CPU 318.)
80A2	Fehler beim Ausgangsparameter PEADDR: Das angegebene Feld konnte nicht alle logischen Adressen aufnehmen.
80A3	Fehler beim Ausgangsparameter PAADDR: Das angegebene Feld konnte nicht alle logischen Adressen aufnehmen.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

GADR_LGC: Die logische Basisadresse einer Baugruppe ermitteln

Beschreibung

Vom Kanal einer Signalbaugruppe seien der zugehörige Baugruppensteckplatz und der Offset im Nutzdatenadressraum der Baugruppe bekannt. Mit der Anweisung "GADR_LGC" ermitteln Sie daraus die zugehörige logische Basisadresse der Baugruppe, d. h. die kleinste E- oder A-Adresse.

Wenn Sie die Anweisung "GADR_LGC" auf Power-Module oder Module mit gepackten Adressen (ET 200S) anwenden, wird die Diagnoseadresse zurückgeliefert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GADR_LGC":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SUBNETID	Input	BYTE	E, A, M, D, L oder Konstante	Bereichskennung: • 0, falls sich der Steckplatz in einem der Baugruppenträger 0 (Zentralgerät) oder 1 bis 21 (Erweiterungsgeräte) befindet • DP-Mastersystem-ID des zugehörigen dezentralen Peripheriesystems, falls sich der Steckplatz in einem dezentralen Peripheriegerät befindet
RACK	Input	WORD	E, A, M, D, L oder Konstante	• Nr. des Baugruppenträgers, falls Bereichskennung 0 • Stationsnummer des dezentralen Peripheriegeräts, falls Bereichskennung > 0

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SLOT	Input	WORD	E, A, M, D, L oder Konstante	Steckplatz-Nr.
SUBSLOT	Input	BYTE	E, A, M, D, L oder Konstante	Submodulsteckplatz (falls kein Submodul gesteckt werden kann, ist hier 0 anzugeben)
SUBADDR	Input	WORD	E, A, M, D, L oder Konstante	Offset im Nutzdatenadressraum der Baugruppe
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
IOID	Output	BYTE	E, A, M, D, L	Kennung des Adressbereichs: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Im Fall einer Mischbaugruppe liefert die Anweisung die Bereichskennung der niedrigeren Adresse. Bei gleichen Adressen liefert die Anweisung die Kennung B#16#54.
LADDR	Output	WORD	E, A, M, D, L	Logische Basisadresse der Baugruppe

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8093	Unzulässiger Wert beim Parameter SUBNETID (die Anweisung "GADR_LGC" ist nicht zulässig für PROFINET IO).
8094	Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.
8095	Unzulässiger Wert beim Parameter RACK
8096	Unzulässiger Wert beim Parameter SLOT
8097	Unzulässiger Wert beim Parameter SUBSLOT
8098	Unzulässiger Wert beim Parameter SUBADDR
8099	Der Steckplatz ist nicht projektiert.
809A	Die Subadresse für den ausgewählten Steckplatz ist nicht projektiert (nur möglich bei zentraler Peripherie für CPU und IM)
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

LGC_GADR: Den zu einer logischen Adresse gehörenden Steckplatz ermitteln

Beschreibung

Mit der Anweisung ermitteln Sie den zu einer logischen Adresse gehörenden Baugruppensteckplatz sowie den Offset im Nutzdatenadressraum der Baugruppe.

Hinweis

Die Anwendung der Anweisung "LGC_GADR" auf ein Modul mit gepackten Adressen (ET 200S) ist nicht möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LGC_GADR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L oder Konstante	Kennung des Adressbereichs: • B#16#00: Bit15 von LADDR gibt an, ob Ein- (Bit15=0) oder Ausgangsadresse (Bit15=1) vorliegt. • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Adresse Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
AREA	Output	BYTE	E, A, M, D, L	Bereichskennung: Sie gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind.
RACK	Output	WORD	E, A, M, D, L	Baugruppenträger-Nr.
SLOT	Output	WORD	E, A, M, D, L	Steckplatz-Nr.
SUBADDR	Output	WORD	E, A, M, D, L	Offset im Nutzdatenadressraum der zugehörigen Baugruppe.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Falls Sie ein Automatisierungssystem S7-400H im redundanten Betrieb betreiben und beim Aufruf der Anweisung "LGC_GADR" im Parameter LADDR die logische Adresse einer Baugruppe eines geschalteten DP-Slaves angeben, wird im high byte des Parameters RACK die DP-Mastersystem-ID des aktiven Kanals geliefert. Falls kein aktiver Kanal existiert, wird die DP-Mastersystem-ID des zugehörigen DP-Mastersystems der Master-CPU ausgegeben.

Parameter AREA

Der Ausgangsparameter AREA gibt an, wie die Ausgangsparameter RACK, SLOT und SUBADDR zu interpretieren sind. Die folgende Tabelle erläutert diese Abhängigkeit.

Wert von AREA	System	Bedeutung von RACK, SLOT und SUBADDR
0	S7-400	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. SUBADDR : Differenz zwischen logischer Adresse und logischer Basisadresse
1	S7-300	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. SUBADDR : Differenz zwischen logischer Adresse und logischer Basisadresse
2	DP	RACK : (low Byte): Stationsnummer RACK : (high Byte): DP-Mastersystem-ID SLOT : Steckplatz-Nr. in der Station SUBADDR : Offset im Nutzdatenadressraum der zugehörigen Baugruppe
3	S5-P-Bereich	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. der Adaptionskapsel SUBADDR : Adresse im S5-x-Bereich
4	S5-Q-Bereich	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. der Adaptionskapsel SUBADDR : Adresse im S5-x-Bereich
5	S5-IM3-Bereich	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. der Adaptionskapsel SUBADDR : Adresse im S5-x-Bereich
6	S5-IM4-Bereich	RACK : Baugruppenträger-Nr. SLOT : Steckplatz-Nr. der Adaptionskapsel SUBADDR : Adresse im S5-x-Bereich

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8090	Angegebene logische Adresse ungültig oder unzulässiger Wert beim Parameter IOID

Fehlercode (W#16#...)	Erläuterung
8093	Für die über die Parameter IOID und LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.3.13 Weitere Funktionen

iSlave

SET_ADDR: Netzadresse als eigener iSlave setzen

Beschreibung

Über die Anweisung können Sie die MAC-Adresse eines iSlave DP-Interface ändern. Die Adressierung wird über die Diagnoseadresse des DP-Interfaces vorgenommen. Für den DP-Master fällt der DP-Slave mit der alten Adresse aus dem Bus und ein DP-Slave mit der neuen Adresse wird in den Bus aufgenommen.

- Ist die Adresse schon von einer Station im DP-Netzwerk belegt, fallen beide DP-Slaves aus dem Netzwerk.
- Wird die Adresse ebenfalls als MPI-Adresse für die PG-Verbindung genutzt, wird der Zugang zum Gerät evtl. gesperrt. Die Information, ob die Geräte überhaupt erreichbar sind, wird in dem Dialog "Erreichbare Teilnehmer" angezeigt.

Die Netzwerkadresse wird nicht in den SDB (Systemdatenbaustein) geschrieben und nicht in den Arbeitsspeicher geladen. Beim Übergang der CPU in STOP und bei einem Warmstart bleibt die Netzwerkadresse erhalten. Nach einem Reset oder Kaltstart wird die ursprünglich konfigurierte Adresse verwendet.

Die Anweisung kann nicht für PROFINET IO verwendet werden, da die Ethernetadresse weltweit als disjunkt definiert werden können. Bei einer Verwendung mit PROFINET IO wird der Fehlercode 16#809B ausgegeben.

Arbeitsweise

Die Anweisung "SET_ADDR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "READ_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET_ADDR":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ=1: Funktionsaufruf
LADDR	Input	WORD	E, A, M, D, L oder Konstante	Logische Basisadresse
ADDR	Input	BYTE	E, A, M, D, L oder Konstante	Neue Teilnehmeradresse
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Der Auftrag ist noch nicht beendet.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
700x	Der Auftrag ist noch nicht beendet. Parameter BUSY=1
8091	Neue Stationsadresse (ADDR) ist ungültig
8092	Station ist keine eigene Slave-Adresse des ISlave
8093	LADDR ist ungültig oder kein Interface
809B	Anweisung kann nicht ausgeführt werden (z. B. ist das Interface kein DP-Slave oder nicht aktiv).
80C3	Ressourcen nicht ausreichend (z. B. mehrfacher Aufruf von Anweisungen mit verschiedenen Parametern).
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.4 Technologie

4.2.4.1 PID Control

PID Basisfunktionen

CONT_C

Beschreibung CONT_C

Die Anweisung CONT_C dient zum Regeln von technischen Prozessen mit kontinuierlichen Ein- und Ausgangsgrößen auf den Automatisierungssystemen SIMATIC S7. Über die Parametrierung können Sie Teilfunktionen des PID-Reglers zu- oder abschalten und damit diesen an die Regelstrecke anpassen. Neben den Funktionen im Soll- und Istwertzweig realisiert die Anweisung einen fertigen PID-Regler mit kontinuierlichem Stellgrößen-Ausgang und Beeinflussungsmöglichkeit des Stellwertes von Hand.

Anwendung

Den Regler können Sie als PID-Festwertregler einzeln oder auch in mehrschleifigen Regelungen als Kaskaden-, Mischungs- oder Verhältnisregler einsetzen. Die Arbeitsweise basiert auf dem PID-Regelalgorithmus des Abtastreglers mit analogem Ausgangssignal, gegebenenfalls um eine Impulsformerstufe zur Bildung von pulsbreitenmodulierten Ausgangssignalen für Zwei- oder Dreipunktregelungen mit proportionalen Stellgliedern ergänzt.

Aufruf

Die Anweisung CONT_C verfügt über eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangsparameter COM_RST = TRUE gesetzt ist. Der Integrierer wird bei der Initialisierung auf den Initialisierungswert I_ITVAL gesetzt. Alle anderen Signalausgänge werden auf Null gesetzt. Nach Durchlauf der Initialisierungsroutine muss COM_RST = FALSE gesetzt werden.

Die Berechnung der Werte in den Regelungsbausteinen erfolgt nur dann korrekt, wenn der Baustein in regelmäßigen Abständen aufgerufen wird. Deshalb sollten Sie die Regelungsbausteine in einem Weckalarm-OB (OB 30 bis OB 38) aufrufen. Die Abtastzeit geben Sie am Parameter CYCLE vor.

Wenn Sie die Anweisung CONT_C als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen CONT_C direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Fehlerinformationen

Das Fehlermeldewort RET_VAL wird vom Baustein nicht ausgewertet.

Siehe auch

Unterschiede zu CONT_C S7-300/400 (Seite 3096)

Arbeitsweise CONT_C

Sollwertzweig

Der Sollwert wird am Eingang SP_INT im Gleitpunktformat eingegeben.

Istwertzweig

Der Istwert kann im Peripherie- und im Gleitpunktformat eingelesen werden. Die Funktion CRP_IN wandelt den Peripheriewert PV_PER in ein Gleitpunktformat von -100 .... +100 % nach folgender Vorschrift um:

Ausgang von CRP_IN = PV_PER * 100 / 27648

Die Funktion PV_NORM normiert den Ausgang von CRP_IN nach folgender Vorschrift:

Ausgang von PV_NORM = (Ausgang von CRP_IN) *PV_FAC + PV_OFF

PV_FAC ist mit 1 und PV_OFF ist mit 0 vorbelegt.

Regeldifferenzbildung

Die Differenz von Soll- und Istwert ergibt die Regeldifferenz. Zur Unterdrückung einer kleinen Dauerschwingung aufgrund der Stellgrößen-Quantisierung (z. B. bei einer Puls-Breitenmodulation mit PULSEGEN) wird die Regeldifferenz über eine Totzone (DEADBAND) geleitet. Bei DEADB_W = 0 ist die Totzone ausgeschaltet.

PID-Algorithmus

Der PID-Algorithmus arbeitet im Stellungsalgorithmus. Der Proportional-, Integral (INT) und Differentialanteil (DIF) sind parallel geschaltet und einzeln zu- und abschaltbar. Damit lassen sich P-, PI-, PD- und PID-Regler parametrieren. Aber auch reine I-Regler sind möglich.

Handwertverarbeitung

Es kann zwischen Hand- und Automatikbetrieb umgeschaltet werden. Bei Handbetrieb wird die Stellgröße einem Handwert nachgeführt.

Der Integrierer (INT) wird intern auf LMN - LMN_P - DISV und der Differenzierer (DIF) auf 0 gesetzt und intern abgeglichen. Das Umschalten in den Automatikbetrieb ist damit stoßfrei.

Stellwertverarbeitung

Der Stellwert wird mit der Funktion LMNLIMIT auf vorgebbare Werte begrenzt. Das Überschreiten der Grenzen durch die Eingangsgröße wird durch Meldebits angezeigt.

Die Funktion LMN_NORM normiert den Ausgang von LMNLIMIT nach folgender Vorschrift:

LMN = (Ausgang von LMNLIMIT) * LMN_FAC + LMN_OFF

LMN_FAC ist mit 1 und LMN_OFF mit 0 vorbelegt.

Der Stellwert steht auch im Peripherieformat zur Verfügung. Die Funktion CRP_OUT wandelt den Gleitpunktwert LMN in einen Peripheriewert nach folgender Vorschrift um:

LMN_PER = LMN * 27648 / 100

Störgrößenaufschaltung

Am Eingang DISV kann eine Störgröße additiv aufgeschaltet werden.

Siehe auch

Unterschiede zu CONT_C S7-300/400 (Seite 3096)

Blockschaltbild CONT_C

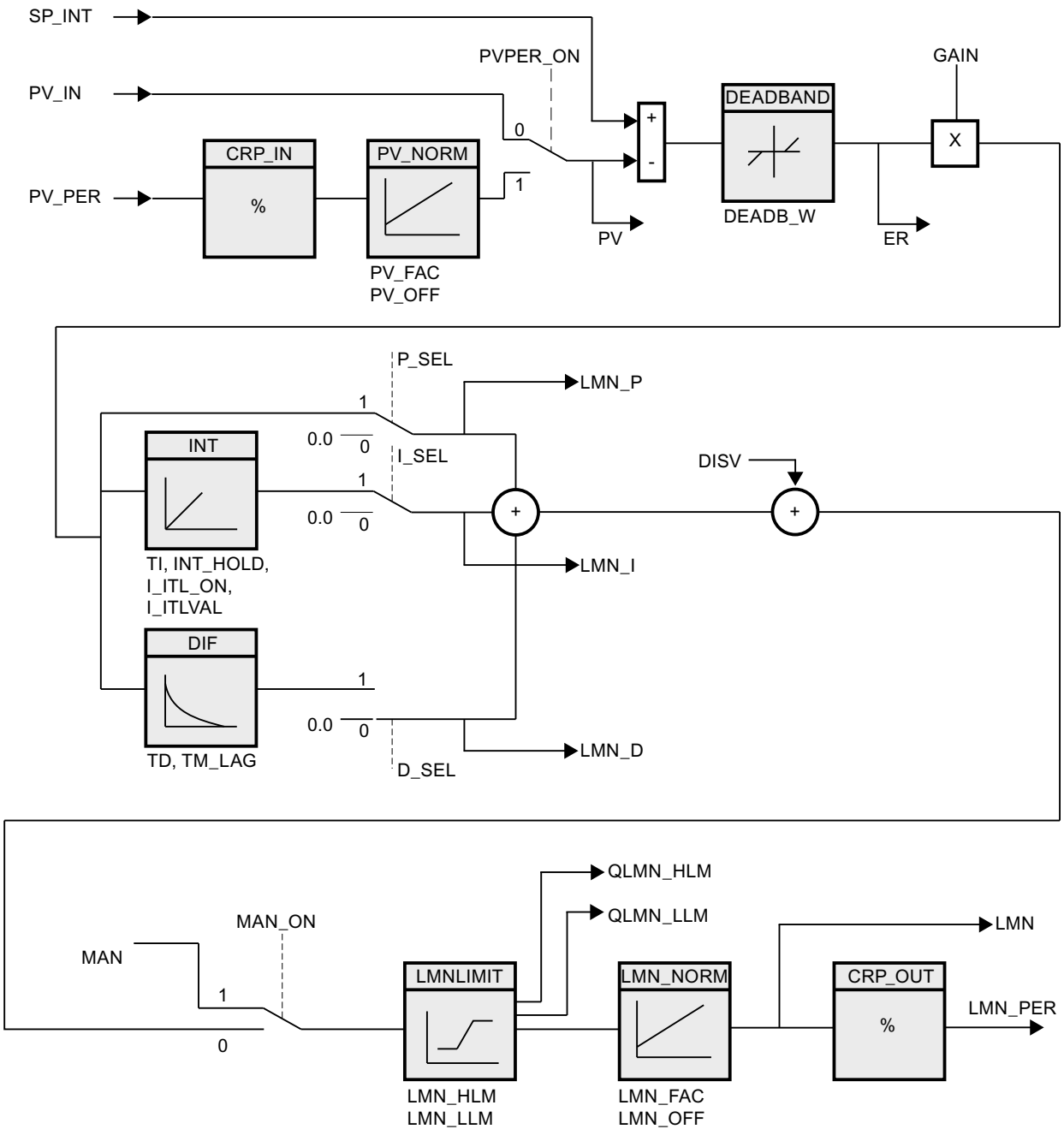

Siehe auch

Unterschiede zu CONT_C S7-300/400 (Seite 3096)

Eingangsparameter CONT_C

Tabelle 4-198

Parameter	Daten-typ	Vorbe-legung	Beschreibung
COM_RST	BOOL	FALSE	Die Anweisung hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang "Neustart" gesetzt ist.
MAN_ON	BOOL	TRUE	Ist der Eingang "Handbetrieb einschalten" gesetzt, ist der Regelkreis unterbrochen. Als Stellwert wird ein Handwert vorgegeben.
PVPER_ON	BOOL	FALSE	Soll der Istwert von der Peripherie eingelesen werden, so muss der Eingang PV_PER mit der Peripherie verschaltet werden und der Eingang "Istwert Peripherie einschalten" gesetzt werden.
P_SEL	BOOL	TRUE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der P-Anteil ist eingeschaltet, wenn der Eingang "P-Anteil einschalten" gesetzt ist.
I_SEL	BOOL	TRUE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der I-Anteil ist eingeschaltet, wenn der Eingang "I-Anteil einschalten" gesetzt ist.
INT_HOLD	BOOL	FALSE	Der Ausgang des Integrierers kann eingefroren werden. Hierzu muss der Eingang "I-Anteil einfrieren" gesetzt werden.
I_ITL_ON	BOOL	FALSE	Der Ausgang des Integrierers kann auf den Eingang I_ITLVAL gesetzt werden. Hierzu muss der Eingang "I-Anteil setzen" gesetzt werden.
D_SEL	BOOL	FALSE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der D-Anteil ist eingeschaltet, wenn der Eingang "D-Anteil einschalten" gesetzt ist.
CYCLE	TIME	T#1s	Die Zeit zwischen den Bausteinaufrufen muss konstant sein. Der Eingang "Abtastzeit" gibt die Zeit zwischen den Bausteinaufrufen an. CYCLE >= 1ms
SP_INT	REAL	0.0	Der Eingang "Interner Sollwert" dient zur Vorgabe eines Sollwertes. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe 1).
PV_IN	REAL	0.0	Am Eingang "Istwert Eingang" kann ein Inbetriebsetzungswert parametriert oder ein externer Istwert im Gleitpunktformat verschaltet werden. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe 1).
PV_PER	WORD	W#16#0000	Der Istwert in Peripherieformat wird am Eingang "Istwert Peripherie" mit dem Regler verschaltet.
MAN	REAL	0.0	Der Eingang "Handwert" dient zur Vorgabe eines Handwertes mittels Bedien-Beobachterfunktion. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe 2).
GAIN	REAL	2.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI	TIME	T#20s	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. TI >= CYCLE
TD	TIME	T#10s	Der Eingang "Differenzierzeit" bestimmt das Zeitverhalten des Differenzierers. TD >= CYCLE
TM_LAG	TIME	T#2s	Verzögerungszeit des D-Anteils Der Algorithmus des D-Anteils beinhaltet eine Verzögerung, die am Eingang "Verzögerungszeit des D-Anteils" parametriert werden kann. TM_LAG >= CYCLE/2
DEADB_W	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W >= 0.0 (%) oder eine physikalische Größe 1)

Parameter	Datentyp	Vorbelegung	Beschreibung
LMN_HLM	REAL	100.0	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. Zulässig sind Real-Werte ab LMN_LLM (%) oder eine physikalische Größe 2).
LMN_LLM	REAL	0.0	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. Zulässig sind Real-Werte bis LMN_HLM (%) oder eine physikalische Größe 2).
PV_FAC	REAL	1.0	Der Eingang "Istwertfaktor" wird mit dem Istwert multipliziert. Der Eingang dient zur Anpassung des Istwertbereiches.
PV_OFF	REAL	0.0	Der Eingang "Istwertoffset" wird mit dem Istwert addiert. Der Eingang dient zur Anpassung des Istwertbereiches.
LMN_FAC	REAL	1.0	Der Eingang "Stellwertfaktor" wird mit dem Stellwert multipliziert. Der Eingang dient zur Anpassung des Stellwertbereiches.
LMN_OFF	REAL	0.0	Der Eingang "Stellwertoffset" wird mit dem Stellwert addiert. Der Eingang dient zur Anpassung des Stellwertbereiches.
I_ITLVAL	REAL	0.0	Der Ausgang des Integrierers kann am Eingang I_ITL_ON gesetzt werden. Am Eingang "Initialisierungswert für I-Anteil" steht der Initialisierungwert. Zulässig sind Werte von -100.0 bis 100.0 (%) oder eine physikalische Größe 2).
DISV	REAL	0.0	Für eine Störgrößenaufschaltung wird die Störgröße am Eingang "Störgröße" verschaltet. Zulässig sind Werte von -100.0 bis 100.0 (%) oder eine physikalische Größe 2).

1) Parameter im Soll-, Istwertzweig mit gleicher Einheit

2) Parameter im Stellwertzweig mit gleicher Einheit

Siehe auch

Unterschiede zu CONT_C S7-300/400 (Seite 3096)

Ausgangsparameter CONT_C

Tabelle 4-199

Parameter	Datentyp	Vorbelegung	Beschreibung
LMN	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert in Gleitpunktformat ausgegeben.
LMN_PER	WORD	W#16#0000	Der Stellwert in Peripherieformat wird am Ausgang "Stellwert Peripherie" mit dem Regler verschaltet.
QLMN_HLM	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Ausgang "Obere Begrenzung des "Stellwertes angesprochen" meldet das Erreichen der oberen Begrenzung.
QLMN_LLM	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Ausgang "Untere Begrenzung des "Stellwertes angesprochen" meldet das Erreichen der unteren Begrenzung.
LMN_P	REAL	0.0	Der Ausgang "P-Anteil" beinhaltet den Proportionalanteil der Stellgröße.
LMN_I	REAL	0.0	Der Ausgang "I-Anteil" beinhaltet den Integralanteil der Stellgröße.
LMN_D	REAL	0.0	Der Ausgang "D-Anteil" beinhaltet den Differentialanteil der Stellgröße.

Parameter	Daten-typ	Vorbe-legung	Beschreibung
PV	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.

Siehe auch

Unterschiede zu CONT_C S7-300/400 (Seite 3096)

CONT_S

Beschreibung CONT_S

Die Anweisung CONT_S dient zum Regeln von technischen Prozessen mit binären Stellwertausgangssignalen für integrierende Stellglieder auf den Automatisierungssystemen SIMATIC S7. Über die Parametrierung lassen sich Teilfunktionen des PI-Schrittreglers zu- oder abschalten und damit an die Regelstrecke anpassen. Neben den Funktionen im Istwertzweig realisiert die Anweisung einen fertigen PI-Regler mit binärem Stellwertausgang und Beeinflussungsmöglichkeit des Stellwertes von Hand. Der Schrittregler arbeitet ohne Stellungsrückmeldung.

Anwendung

Der Regler kann als PI-Festwertregler einzeln oder in unterlagerten Regelkreisen bei Kaskaden-, Mischungs- oder Verhältnisregelungen eingesetzt werden, jedoch nicht als Führungsregler. Die Arbeitsweise basiert auf dem PI-Regelalgorithmus des Abtastreglers und wird um die Funktionsglieder zur Erzeugung des binären Ausgangssignals aus dem analogen Stellsignal ergänzt.

Aufruf

Die Anweisung CONT_S verfügt über eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangsparameter COM_RST = TRUE gesetzt ist. Alle Signalausgänge werden auf Null gesetzt. Nach Durchlauf der Initialisierungsroutine muss COM_RST = FALSE gesetzt werden.

Die Berechnung der Werte in den Regelungsbausteinen erfolgt nur dann korrekt, wenn der Baustein in regelmäßigen Abständen aufgerufen wird. Deshalb sollten Sie die Regelungsbausteine in einem Weckalarm-OB (OB 30 bis OB 38) aufrufen. Die Abtastzeit geben Sie am Parameter CYCLE vor.

Wenn Sie die Anweisung CONT_S als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen CONT_S direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Fehlerinformationen

Das Fehlermeldewort RET_VAL wird vom Baustein nicht ausgewertet.

Siehe auch

Unterschiede zu CONT_S S7-300/400 (Seite 3096)

Arbeitsweise CONT_S

Sollwertzweig

Der Sollwert wird am Eingang SP_INT im Gleitpunktformat eingegeben.

Istwertzweig

Der Istwert kann im Peripherie- und im Gleitpunktformat eingelesen werden. Die Funktion CRP_IN wandelt den Peripheriewert PV_PER in ein Gleitpunktformat von -100 .... +100 % nach folgender Vorschrift um:

Ausgang von CRP_IN = PV_PER * 100 / 27648

Die Funktion PV_NORM normiert den Ausgang von CRP_IN nach folgender Vorschrift:

Ausgang von PV_NORM = (Ausgang von CRP_IN) * PV_FAC + PV_OFF

PV_FAC ist mit 1 und PV_OFF ist mit 0 vorbelegt.

Regeldifferenzbildung

Die Differenz von Soll- und Istwert ergibt die Regeldifferenz. Zur Unterdrückung einer kleinen Dauerschwingung aufgrund der Stellgrößen-Quantisierung (begrenzte Auflösung des Stellwertes durch das Stellventil) wird die Regeldifferenz über eine Totzone (DEADBAND) geleitet. Bei DEADB_W = 0 ist die Totzone ausgeschaltet.

PI-Schrittalgorithmus

Die Anweisung arbeitet ohne Stellungsrückmeldung. Der I-Anteil des PI-Algorithmus und die gedachte Stellungsrückmeldung werden in **einem** Integrator (INT) berechnet und als Rückführungswert mit dem verbliebenen P-Anteil verglichen. Die Differenz geht auf ein Dreipunktglied (THREE_ST) und einen Impulsformer (PULSEOUT), der die Impulse für das Stellventil bildet. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Störgrößenaufschaltung

Am Eingang DISV kann eine Störgröße additiv aufgeschaltet werden.

Siehe auch

Unterschiede zu CONT_S S7-300/400 (Seite 3096)

Blockschaltbild CONT_S

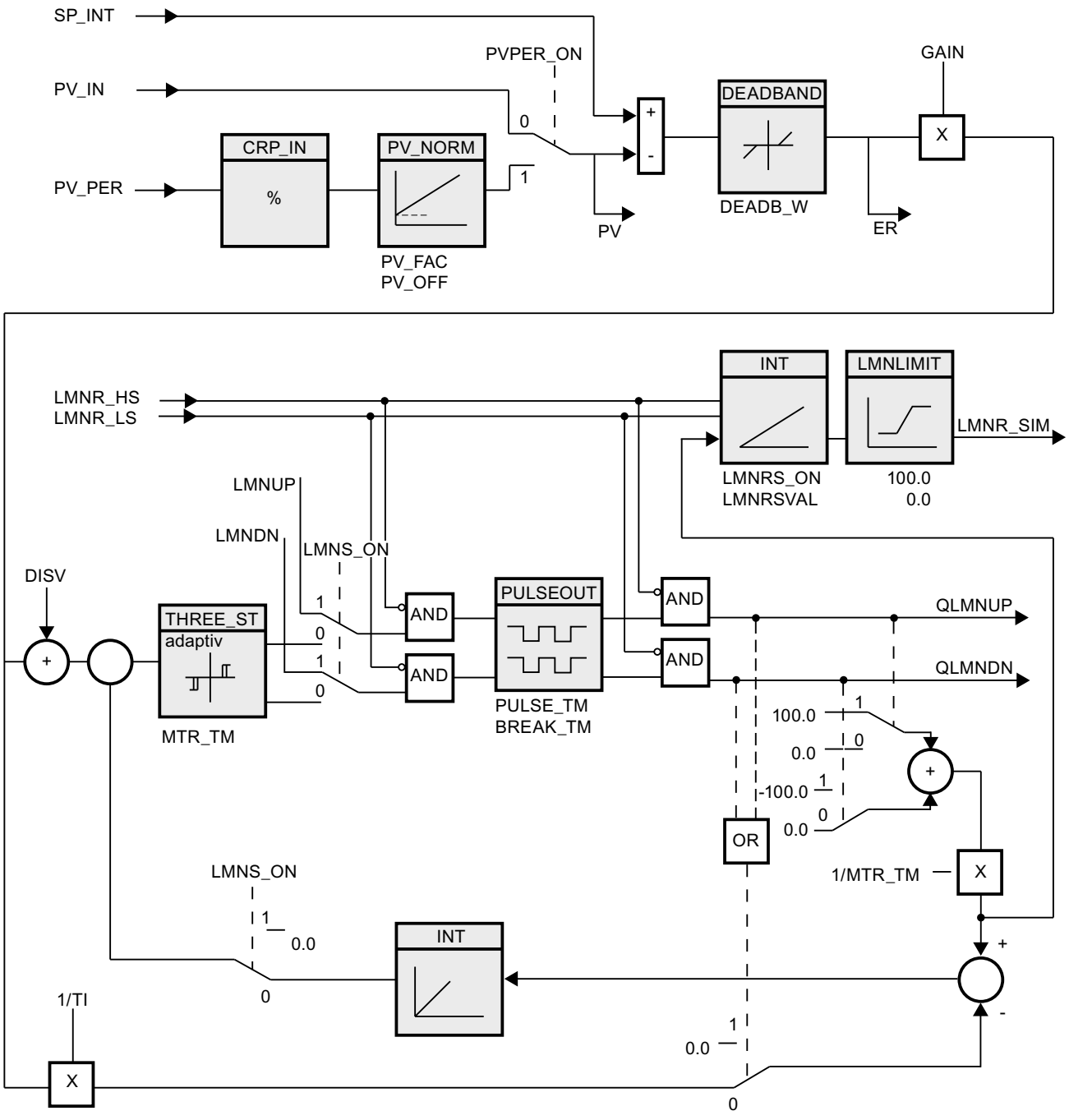

Siehe auch

Unterschiede zu CONT_S S7-300/400 (Seite 3096)

Eingangsparameter CONT_S

Tabelle 4-200

Parameter	Datentyp	Vorbelegung	Beschreibung
COM_RST	BOOL	FALSE	Die Anweisung hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang "Neustart" gesetzt ist.
LMNR_HS	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "Oberes Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNR_HS=TRUE heißt: Das Stellventil befindet sich am oberen Anschlag.
LMNR_LS	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "Unteres Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNR_LS=TRUE heißt: Das Stellventil befindet sich am unteren Anschlag.
LMNS_ON	BOOL	FALSE	Am Eingang "Handbetrieb der Stellwertsignale einschalten" wird die Stellwertsignalverarbeitung auf Hand geschaltet.
LMNUP	BOOL	FALSE	Bei Handbetrieb der Stellwertsignale wird am Eingang "Stellwertsignal Hoch" das Ausgangssignal QLMNUP bedient.
LMNDN	BOOL	FALSE	Bei Handbetrieb der Stellwertsignale wird am Eingang "Stellwertsignal Tief" das Ausgangssignal QLMNDN bedient.
PVPER_ON	BOOL	FALSE	Soll der Istwert von der Peripherie eingelesen werden, so muss der Eingang PV_PER mit der Peripherie verschaltet werden und der Eingang "Istwert Peripherie einschalten" gesetzt werden.
CYCLE	TIME	T#1s	Die Zeit zwischen den Bausteinaufrufen muss konstant sein. Der Eingang "Abtastzeit" gibt die Zeit zwischen den Bausteinaufrufen an. CYCLE >= 1ms
SP_INT	REAL	0.0	Der Eingang "Interner Sollwert" dient zur Vorgabe eines Sollwertes. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe [1].
PV_IN	REAL	0.0	Am Eingang "Istwert Eingang" kann ein Inbetriebsetzungswert parametriert oder ein externer Istwert im Gleitpunktformat verschaltet werden. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe [1].
PV_PER	WORD	W#16#0000	Der Istwert in Peripherieformat wird am Eingang "Istwert Peripherie" mit dem Regler verschaltet.
GAIN	REAL	2.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI	TIME	T#20s	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. TI >= CYCLE
DEADB_W	REAL	1.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. Zulässig sind Werte von 0 bis 100 % oder eine physikalische Größe [1].
PV_FAC	REAL	1.0	Der Eingang "Istwertfaktor" wird mit dem Istwert multipliziert. Der Eingang dient zur Anpassung des Istwertbereiches.
PV_OFF	REAL	0.0	Der Eingang "Istwertoffset" wird mit dem Istwert addiert. Der Eingang dient zur Anpassung des Istwertbereiches.
PULSE_TM	TIME	T#3s	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden. PULSE_TM >= CYCLE
BREAK_TM	TIME	T#3s	Am Parameter "Mindestpausendauer" kann eine minimale Pausenlänge parametriert werden. BREAK_TM >= CYCLE

Parameter	Datentyp	Vorbelegung	Beschreibung
MTR_TM	TIME	T#30s	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils vom Anschlag zu Anschlag eingetragen. MTR_TM >= CYCLE
DISV	REAL	0.0	Für eine Störgrößenaufschaltung wird die Störgröße am Eingang "Störgröße" verschaltet. Zulässig sind Werte von -100 bis 100 % oder eine physikalische Größe [2].

[1] Parameter im Soll-, Istwertzweig mit gleicher Einheit

[2] Parameter im Stellwertzweig mit gleicher Einheit

Siehe auch

Unterschiede zu CONT_S S7-300/400 (Seite 3096)

Ausgangsparameter CONT_S

Tabelle 4-201

Parameter	Datentyp	Vorbelegung	Beschreibung
QLMNUP	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Hoch" gesetzt, soll das Stellventil geöffnet werden.
QLMNDN	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Tief" gesetzt, soll das Stellventil geschlossen werden.
PV	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.

Siehe auch

Unterschiede zu CONT_S S7-300/400 (Seite 3096)

PULSEGEN

Beschreibung PULSEGEN

Die Anweisung PULSEGEN dient zum Aufbau eines PID-Reglers mit Impulsausgang für proportionale Stellglieder. PULSEGEN transformiert die Eingangsgröße INV (= LMN des PID-Reglers) durch Modulation der Impulsbreite in eine Impulsfolge mit konstanter Periodendauer, welche der Zykluszeit entspricht, mit der die Eingangsgröße aktualisiert wird.

Anwendung

Mit der Anweisung PULSEGEN lassen sich PID-Zwei- oder Dreipunktregler mit Pulsweitenmodulation aufbauen. Die Funktion wird meistens in Verbindung mit dem kontinuierlichen Regler CONT_C angewendet.

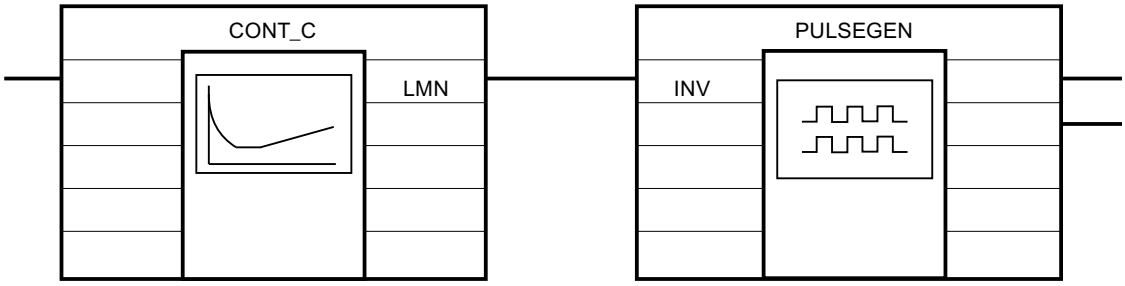

Aufruf

Die Anweisung PULSEGEN verfügt über eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangsparameter COM_RST = TRUE gesetzt ist. Alle Signalausgänge werden auf Null gesetzt. Nach Durchlauf der Initialisierungsroutine muss COM_RST = FALSE gesetzt werden.

Die Berechnung der Werte in den Regelungsbausteinen erfolgt nur dann korrekt, wenn der Baustein in regelmäßigen Abständen aufgerufen wird. Deshalb sollten Sie die Regelungsbausteine in einem Weckalarm-OB (OB 30 bis OB 38) aufrufen. Die Abtastzeit geben Sie am Parameter CYCLE vor.

Verhalten im Fehlerfall

Das Fehlermeldewort RET_VAL wird vom Baustein nicht ausgewertet.

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Arbeitsweise PULSEGEN

Impulsweitenmodulation

Die Dauer eines Impulses pro Periodendauer ist proportional der Eingangsgröße. Dabei ist der durch PER_TM parametrierte Zyklus nicht identisch mit dem Bearbeitungszyklus der Anweisung PULSEGEN. Vielmehr setzt sich ein Zyklus PER_TM aus mehreren Bearbeitungszyklen der Anweisung PULSEGEN zusammen, wobei die Anzahl der PULSEGEN-Aufrufe pro PER_TM-Zyklus ein Maß für die Genauigkeit der Impulsbreite darstellt.

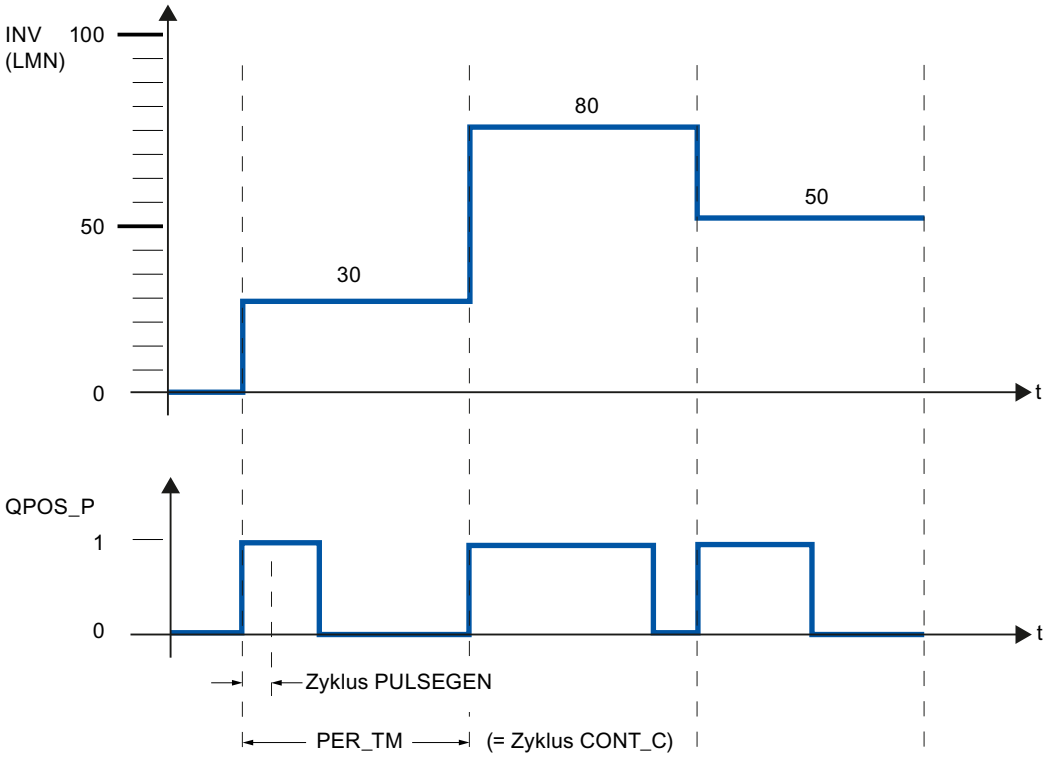

Eine Eingangsgröße 30% und 10 PULSEGEN-Aufrufe pro PER_TM bedeuten also:

- "eins" am Ausgang QPOS_P für die ersten drei Aufrufe von PULSEGEN (30% von 10 Aufrufen)
- "null" am Ausgang QPOS_P für sieben weitere Aufrufe von PULSEGEN (70% von 10 Aufrufen)

Blockschaltbild

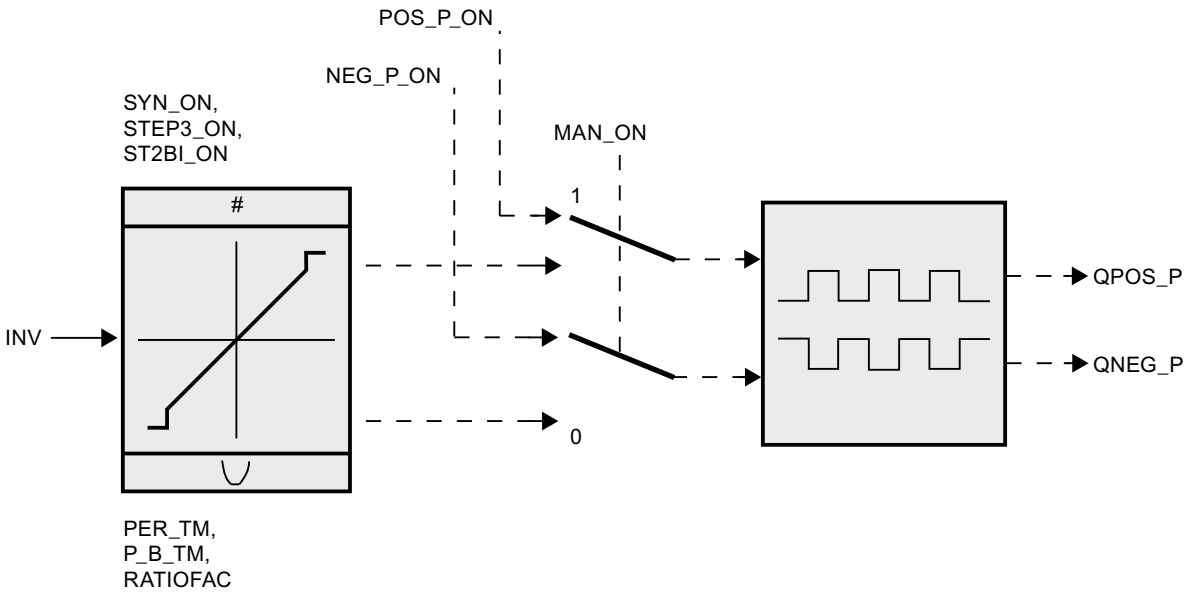

Stellwertgenauigkeit

Durch ein "Abtastverhältnis" von 1:10 (CONT_C-Aufrufe zu PULSEGEN-Aufrufe) ist die Stellwertgenauigkeit in diesem Beispiel auf 10% beschränkt, d.h. vorgegebene Eingangswerte INV können nur im Raster von 10% auf eine Impulslänge am Ausgang QPOS_P abgebildet werden.

Entsprechend erhöht sich die Genauigkeit mit der Anzahl der PULSEGEN-Aufrufe pro CONT_C-Aufruf.

Wird z. B. PULSEGEN 100 mal häufiger aufgerufen als CONT_C, so erreicht man eine Auflösung von 1% des Stellwertbereiches.

Hinweis

Die Untersetzung der Aufrufhäufigkeit müssen Sie selbst programmieren.

Automatische Synchronisation

Es besteht die Möglichkeit, die Impulsausgabe mit der Anweisung, der die Eingangsgröße INV (z. B. CONT_C) aktualisiert, automatisch zu synchronisieren. Damit ist gewährleistet, dass eine ändernde Eingangsgröße auch schnellstmöglich als Impuls ausgegeben wird.

Der Impulsformer wertet immer im Zeitabstand der Periodendauer PER_TM die Eingangsgröße INV aus und wandelt den Wert in ein Impulssignal der entsprechenden Länge.

Da aber INV meistens in einer langsameren Weckalarmebene berechnet wird, sollte der Impulsformer möglichst schnell nach der Aktualisierung von INV mit der Umwandlung des diskreten Wertes in ein Impulssignal beginnen.

Dazu kann der Baustein den Start der Periode nach folgendem Verfahren selbst synchronisieren:

Hat sich INV geändert und befindet sich der Bausteinaufruf nicht im ersten oder in den letzten zwei Aufrufzyklen einer Periode, so wird eine Synchronisation durchgeführt. Die Impulsdauer wird neu berechnet und beim nächsten Zyklus mit einer neuen Periode wird mit der Ausgabe begonnen.

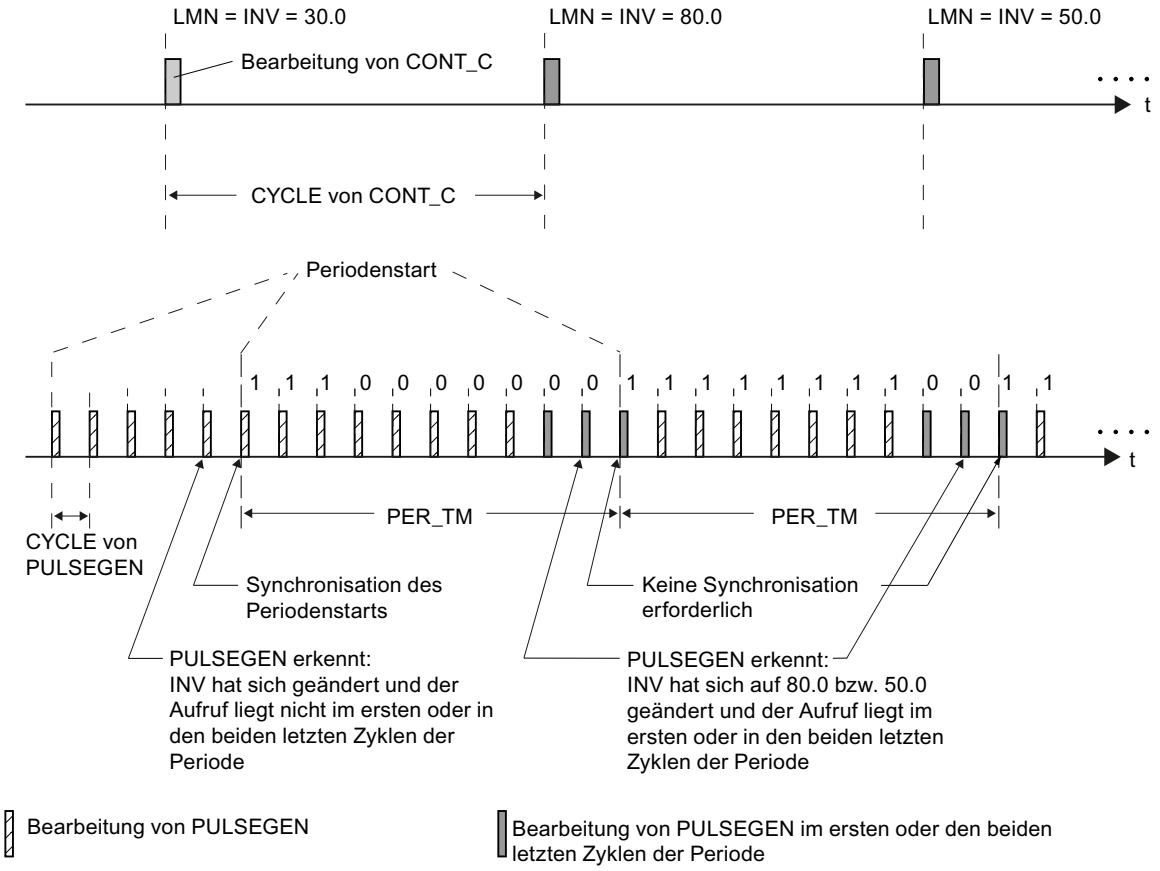

Die automatische Synchronisation ist abgeschaltet, wenn SYN_ON = FALSE.

Hinweis

Durch den Beginn der neuen Periode wird der Altwert von INV (d.h. von LMN) nach erfolgter Synchronisation mehr oder weniger ungenau auf das Impulssignal abgebildet.

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Betriebsarten PULSEGEN

Betriebsarten

Je nach Parametrierung des Impulsformers können PID-Regler mit Dreipunktverhalten oder mit bipolarem bzw. unipolarem Zweipunktausgang konfiguriert werden. Nachstehende Tabelle zeigt die Einstellung der Schalterkombinationen für die möglichen Betriebsarten.

Betriebsart	MAN_ON	STEP3_ON	ST2BI_ON
Dreipunktregelung	FALSE	TRUE	beliebig
Zweipunktregelung mit bipolarem Stellbereich (-100 % ... 100 %)	FALSE	FALSE	TRUE
Zweipunktregelung mit unipolarem Stellbereich (0 % ... 100 %)	FALSE	FALSE	FALSE
Handbetrieb	TRUE	beliebig	beliebig

Handbetrieb bei Zwei- bzw. Dreipunkt-Regelung

Im Handbetrieb (MAN_ON = TRUE) können die Binärausgänge des Dreipunkt- bzw. Zweipunktreglers über die Signale POS_P_ON und NEG_P_ON unabhängig von INV gesetzt werden.

Regelung	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Dreipunktregelung	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Zweipunktregelung	FALSE	beliebig	FALSE	TRUE
	TRUE	beliebig	TRUE	FALSE

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Dreipunktregelung

Dreipunktregelung

In der Betriebsart "Dreipunktregelung" können drei Zustände des Stellsignals erzeugt werden. Dazu werden die Zustandswerte der binären Ausgangssignale QPOS_P und QNEG_P den jeweiligen Betriebszuständen des Stellgliedes zugeordnet. Die Tabelle zeigt das Beispiel einer Temperaturregelung:

Ausgangssignale	heizen	aus	kühlen
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

Aus der Eingangsgröße wird über eine Kennlinie die Impulsdauer berechnet. Die Form dieser Kennlinie wird durch die Mindestimpuls- bzw. Mindestpausendauer und dem Verhältnisfaktor definiert. Der normale Wert für den Verhältnisfaktor ist 1.

Die Knickpunkte an den Kennlinien werden durch die Mindestimpuls- bzw. Mindestpausendauer verursacht.

Mindestimpuls- bzw. Mindestpausendauer

Eine richtig parametrierte Mindestimpuls- bzw. Mindestpausendauer P_B_TM kann kurze Ein- oder Ausschaltzeiten verhindern, die die Lebensdauer von Schaltgliedern und Stelleinrichtungen beeinträchtigen. Kleine Absolutwerte der Eingangsgröße LMN, die eine Impulsdauer kleiner als P_B_TM erzeugen würden, werden unterdrückt. Große Eingangswerte, die eine Impulsdauer größer als PER_TM - P_B_TM erzeugen würden, werden auf 100 % bzw. -100 % gesetzt.

Die Dauer der positiven oder negativen Impulse errechnet sich aus Eingangsgröße (in %) mal Periodendauer:

Impulsdauer = INV / 100 * PER_TM

Das folgende Bild zeigt eine symmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 1)

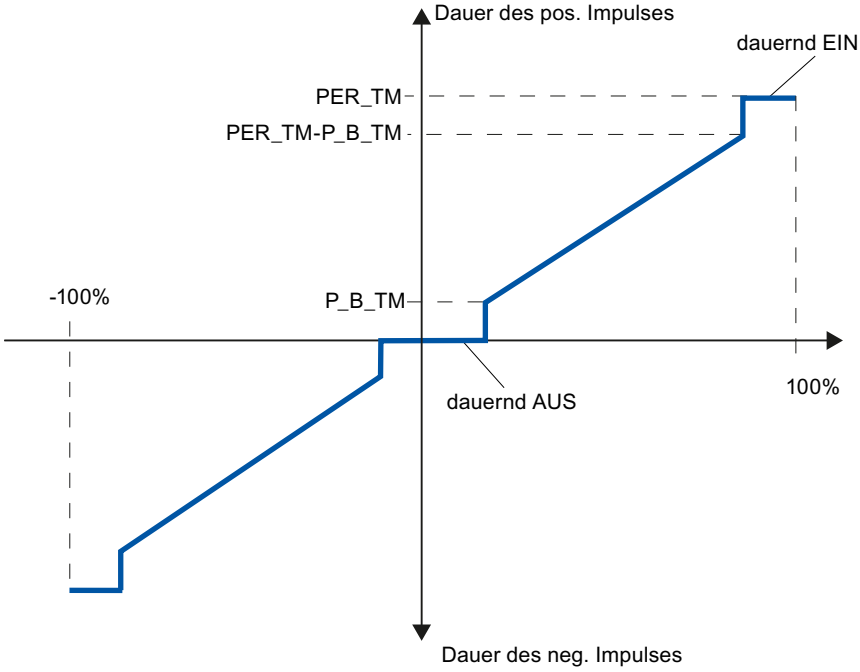

Dreipunktregelung unsymmetrisch

Über den Verhältnisfaktor RATIOFAC kann das Verhältnis der Dauer von positiven zu negativen Impulsen verändert werden. Bei einem thermischen Prozess lassen sich damit z. B. unterschiedliche Streckenzeitkonstanten für Heizen und Kühlen berücksichtigen.

Verhältnisfaktor < 1

Die aus Eingangsgröße mal Periodendauer berechnete Impulsdauer am negativen Impulsausgang wird mit dem Verhältnisfaktor multipliziert.

Positive Impulsdauer = INV /100 * PER_TM

Negative Impulsdauer = INV / 100 * PER_TM * RATIOFAC

Das folgende Bild zeigt die unsymmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 0.5)

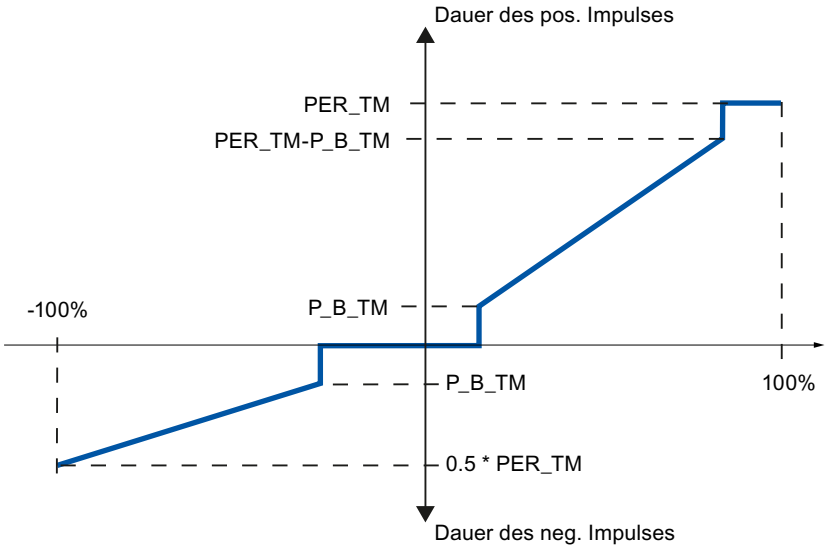

Verhältnisfaktor > 1

Die aus Eingangsgröße mal Periodendauer berechnete Impulsdauer am positiven Impulsausgang wird durch den Verhältnisfaktor dividiert.

positive Impulsdauer = INV / 100 * PER_TM / RATIOFAC

negative Impulsdauer = INV / 100 * PER_TM

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Zweipunktregelung

Bei der Zweipunktregelung wird nur der positive Impulsausgang QPOS_P von PULSEGEN mit dem betreffenden Ein/Aus-Stellglied verbunden. Je nach genutztem Stellwertbereich hat der Zweipunktregler einen bipolaren oder einen unipolaren Stellwertbereich.

Zweipunktregelung mit bipolarem Stellwertbereich (-100%...100%)

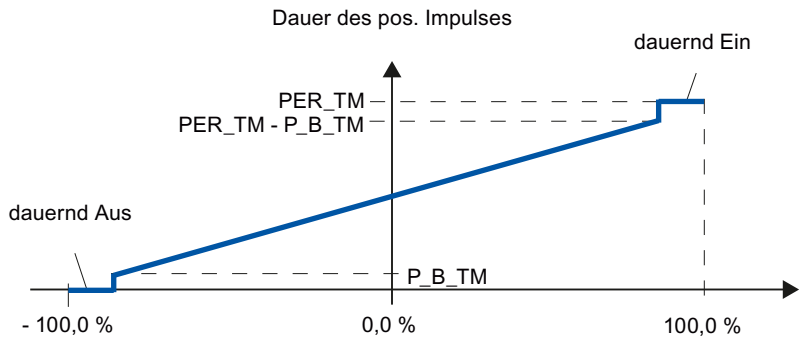

Zweipunktregelung mit unipolarem Stellwertbereich (0%...100%)

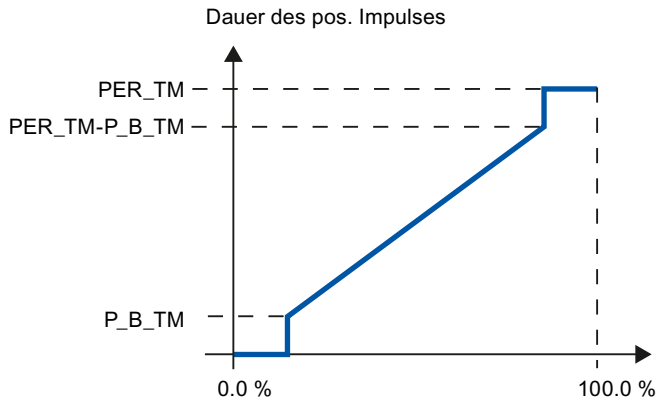

An QNEG_P steht das negierte Ausgangssignal zur Verfügung, falls die Verschaltung des Zweipunktreglers im Regelkreis ein logisch invertiertes Binärsignal für die Stellimpulse erfordert.

Impuls	Stellglied Ein	Stellglied Aus
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Eingangsparameter PULSEGEN

Die Werte der Eingangsparameter werden im Baustein nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Tabelle 4-202

Parameter	Datentyp	Vorbelegung	Beschreibung
INV	REAL	0.0	Am Eingangsparameter "Eingangsvariable" wird eine analoge Stellwertgröße aufgeschaltet. Zulässig sind Werte von -100 bis 100 %.
PER_TM	TIME	T#1s	Am Parameter "Periodendauer" wird die konstante Periodendauer der Pulsweitenmodulation eingegeben. Sie entspricht der Abtastzeit des Reglers. Das Verhältnis Abtastzeit des Impulsformers zu Abtastzeit des Reglers bestimmt die Genauigkeit der Pulsweitenmodulation. PER_TM >=20*CYCLE
P_B_TM	TIME	T#50ms	Am Parameter "Mindestimpuls- bzw. Mindestpausendauer" kann eine minimale Impuls- bzw. Pausenlänge parametriert werden. P_B_TM >= CYCLE
RATIOFAC	REAL	1.0	Durch den Eingangsparameter "Verhältnisfaktor" kann das Verhältnis der Dauer von negativen zu positiven Impulsen verändert werden. Bei einem thermischen Prozess können damit unterschiedliche Zeitkonstanten für Heizen und Kühlen (z. B. Prozess mit elektrischer Heizung und Wasserkühlung) kompensiert werden. Zulässig sind Werte von 0.1 bis 10.0.
STEP3_ON	BOOL	TRUE	Am Eingangsparameter "Dreipunktregelung einschalten" wird die entsprechende Betriebsart aktiviert. Bei Dreipunktregelung arbeiten beide Ausgangssignale.
ST2BI_ON	BOOL	FALSE	Am Eingangsparameter "Zweipunktregelung für bipolaren Stellwertbereich einschalten" kann zwischen den Betriebsarten "Zweipunktregelung für bipolaren Stellwertbereich" und "Zweipunktregelung für unipolaren Stellwertbereich" gewählt werden. Hierbei muss STEP3_ON = FALSE sein.
MAN_ON	BOOL	FALSE	Durch Setzen des Eingangsparameter "Handbetrieb einschalten" können die Ausgangssignale von Hand gesetzt werden.
POS_P_ON	BOOL	FALSE	Bei Handbetrieb Dreipunktregelung kann am Eingangsparameter "Positiver Impuls ein" das Ausgangssignal QPOS_P bedient werden. Bei Handbetrieb Zweipunktregelung wird QNEG_P immer invertiert zu QPOS_P gesetzt.
NEG_P_ON	BOOL	FALSE	Bei Handbetrieb Dreipunktregelung kann am Eingangsparameter "Negativer Impuls ein" das Ausgangssignal QNEG_P bedient werden. Bei Handbetrieb Zweipunktregelung wird QNEG_P immer invertiert zu QPOS_P gesetzt.
SYN_ON	BOOL	TRUE	Es besteht die Möglichkeit durch Setzen des Eingangsparameters "Synchronisation einschalten" die Impulsausgabe mit dem Baustein, der die Eingangsgröße INV aktualisiert, automatisch zu synchronisieren. Damit ist gewährleistet, dass eine sich ändernde Eingangsgröße auch schnellstmöglich als Impuls ausgegeben wird.
COM_RST	BOOL	FALSE	Der Baustein hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang "Neustart" gesetzt ist.
CYCLE	TIME	T#10ms	Die Zeit zwischen den Bausteinaufrufen muss konstant sein. Der Eingang "Abtastzeit" gibt die Zeit zwischen den Bausteinaufrufen an. CYCLE >= 1ms

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

Ausgangsparameter PULSEGEN

Tabelle 4-203

Parameter	Datentyp	Vorbelegung	Beschreibung
QPOS_P	BOOL	FALSE	Der Ausgangsparameter "Ausgangssignal positiver Impuls" ist gesetzt, wenn ein Impuls ausgegeben werden soll. Bei Dreipunktregelung ist es der positive Impuls. Bei Zweipunktregelung wird QNEG_P immer invertiert zu QPOS_P gesetzt
QNEG_P	BOOL	FALSE	Der Ausgangsparameter "Ausgangssignal negativer Impuls" ist gesetzt, wenn ein Impuls ausgegeben werden soll. Bei Dreipunktregelung ist es der negative Impuls. Bei Zweipunktregelung wird QNEG_P immer invertiert zu QPOS_P gesetzt.

Siehe auch

Unterschiede zu PULSEGEN S7-300/400 (Seite 3097)

TCONT_CP

Beschreibung TCONT_CP

Die Anweisung TCONT_CP dient zum Regeln von Temperaturprozessen mit kontinuierlicher oder impulsförmiger Ansteuerung. Die Arbeitsweise basiert auf dem PID-Regelalgorithmus, der mit zusätzlichen Funktionen für Temperaturprozesse ausgestattet ist. Zur Verbesserung des Regelverhaltens bei Temperaturstrecken hat der Baustein eine Regelzone und eine Reduzierung des P-Anteils bei Sollwertsprüngen.

Die PI/PID-Parameter kann die Anweisung mittels Regleroptimierung selbst einstellen.

Anwendung

Der Regler versorgt ein Stellglied, d. h. Sie können mit einem Regler entweder nur heizen oder nur kühlen. Bei Einsatz für einen Kühlprozess müssen Sie GAIN mit einem negativen Wert parametrieren. Die so parametrierte Reglerinvertierung bewirkt, dass nun z. B. bei einem Anstieg der Temperatur sich die Stellgröße LMN und damit die Kühlleistung erhöht.

Aufruf

Die Anweisung TCONT_CP muss äquidistant aufgerufen werden. Benutzen Sie hierzu eine Weckalarmebene (z.B. OB35 bei S7-300).

Die Anweisung TCONT_CP hat eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangs-Parameter COM_RST = TRUE gesetzt ist. Der Integrierer wird bei der Initialisierung auf den Initialisierungswert I_ITVAL gesetzt. Alle anderen Signalausgänge werden auf Null gesetzt. Der Baustein setzt nach dem Bearbeiten der Initialisierungsroutine COM_RST wieder

auf FALSE zurück. Wenn Sie eine Initialisierung bei Neustart der CPU wünschen, rufen Sie den Baustein im OB 100 mit COM_RST = TRUE auf.

Wenn Sie die Anweisung TCONT_CP als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen TCONT_CP direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Arbeitsweise TCONT_CP

Sollwertzweig

Der Sollwert wird am Eingang SP_INT im Gleitpunktformat physikalisch oder in Prozent eingegeben. Sollwert und Istwert müssen an der Regeldifferenzbildung die gleiche Einheit besitzen.

Istwertauswahl (PVPER_ON)

Der Istwert kann abhängig von PVPER_ON im Peripherie- oder im Gleitpunktformat eingelesen werden.

PVPER_ON	Istwerteingabe
TRUE	Der Istwert wird über die Analogperipherie (PEW xxx) am Eingang PV_PER eingelesen.
FALSE	Der Istwert wird im Gleitpunktformat am Eingang PV_IN eingelesen.

Istwertformatumwandlung CRP_IN (PER_MODE)

Die Funktion CRP_IN wandelt den Peripheriewert PV_PER abhängig vom Schalter PER_MODE in ein Gleitpunktformat nach folgender Vorschrift um:

PER_MODE	Ausgang von CRP_IN	Analogeingabe-Typ	Einheit
0	PV_PER * 0.1	Thermoelemente; PT100/NI100; Standard	°C;°F
1	PV_PER * 0.01	PT100/NI100; Klima;	°C;°F
2	PV_PER * 100/27648	Spannung/Strom	%

Istwertnormierung PV_NORM (PF_FAC, PV_OFFS)

Die Funktion PV_NORM berechnet den Ausgang von CRP_IN nach folgender Vorschrift:

"Ausgang von PV_NORM" = "Ausgang von CRP_IN" * PV_FAC + PV_OFFS

Sie kann für folgende Zwecke eingesetzt werden:

- Istwert-Anpassung mit PV_FAC als Istwertfaktor und PV_OFFS als Istwertoffset
- Normierung von Temperatur nach Prozent
  Sie wollen den Sollwert in Prozent eingeben und müssen nun den gemessenen Temperaturwert in Prozent umrechnen.
- Normierung von Prozent nach Temperatur
  Sie wollen den Sollwert in der physikalischen Größe Temperatur eingeben und müssen nun den gemessenen Spannungs-/Strom-Wert in eine Temperatur umrechnen.

Berechnung der Parameter:

- PV_FAC = Bereich von PV_NORM/Bereich von CRP_IN;
- PV_OFFS = UG(PV_NORM) - PV_FAC * UG(CRP_IN);
  mit UG: Untergrenze

Mit den Defaultwerten (PV_FAC = 1.0 und PV_OFFS = 0.0) ist die Normierung abgeschaltet. Der effektiv wirksame Istwert wird am Ausgang PV ausgegeben.

Hinweis

Bei Impulsregelung muss der Istwert im schnellen Impulsaufruf dem Baustein übergeben werden (Grund: Mittelwertfilterung). Sonst kann sich die Regelqualität verschlechtern.

Beispiel zur Istwertnormierung

Wenn Sie den Sollwert in Prozent vorgeben wollen und Sie einen Temperaturbereich von -20 bis 85 °C an CRP_IN anliegen haben, müssen Sie den Temperaturbereich in Prozent umnormieren.

Das folgende Bild stellt ein Beispiel für die Anpassung des Temperaturbereichs -20 bis 85 °C auf intern 0 bis 100 % dar:

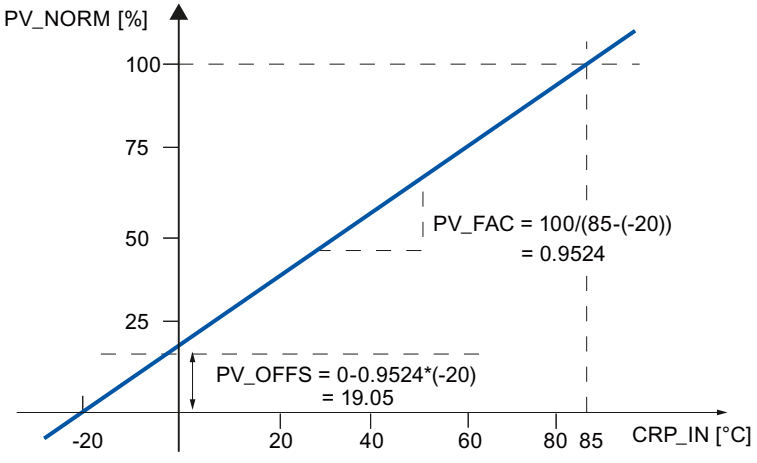

Regeldifferenzbildung

Die Differenz von Soll- und Istwert ergibt die Regeldifferenz vor der Totzone.

Soll- und Istwert müssen in der gleichen Einheit vorliegen.

Totzone (DEADB_W)

Zur Unterdrückung einer kleinen Dauerschwingung aufgrund der Stellgrößen-Quantisierung (z. B. bei einer Pulsweitenmodulation mit PULSEGEN) wird die Regeldifferenz über eine Totzone (DEADBAND) geleitet. Bei DEADB_W = 0.0 ist die Totzone ausgeschaltet. Die wirksame Regeldifferenz wird am Parameter ER angezeigt.

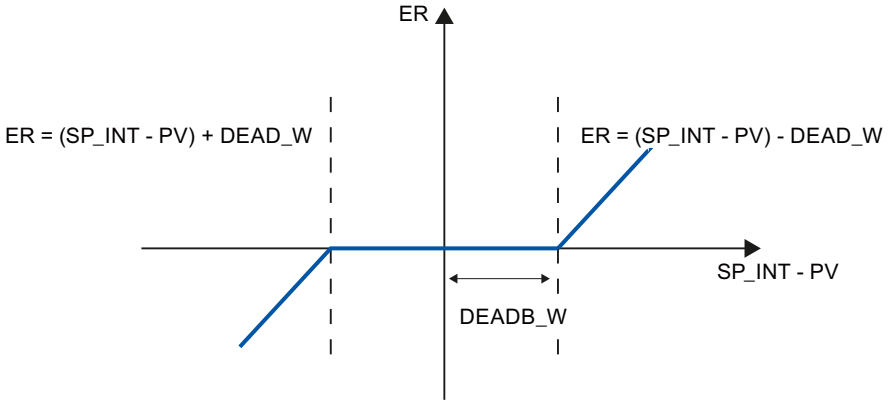

PID-Algorithmus

In folgendem Bild ist das Blockschaltbild des PID-Algorithmus dargestellt.

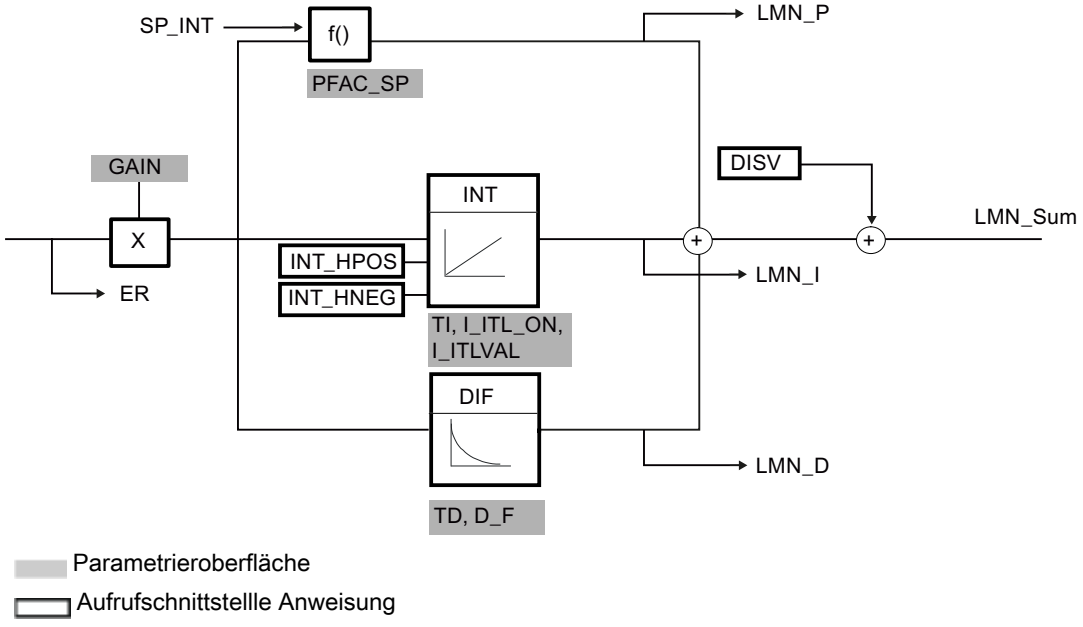

PID-Algorithmus (GAIN, TI, TD, D_F)

Der PID-Algorithmus arbeitet im Stellungsalgorithmus. Der Proportional-, Integral (INT) und Differentialanteil (DIF) sind parallel geschaltet und lassen sich einzeln zu- und abschalten. Damit können Sie P-, PI-, PD- und PID-Regler parametrieren.

Die Regleroptimierung unterstützt PI- und PID-Regler. Reglerinvertierung erfolgt über ein negatives GAIN (Kühlregler).

Wenn Sie TI und TD auf 0.0 setzen erhalten Sie einen reinen P-Regler am Arbeitspunkt.

Die Sprungantwort im Zeitbereich ist:

$$LMN_Sum(t) = GAIN \cdot ER(0) \cdot \left(1 + \frac{1}{TI} \cdot t + D_F \cdot e^{\frac{-t}{TD\ /\ D_F}} \right)$$

Darin bedeutet:

LMN_Sum(t) die Stellgröße bei Automatikbetrieb des Reglers

ER (0) die Sprunghöhe der normierten Regeldifferenz

GAIN die Reglerverstärkung

TI die Integrationszeit

TD die Differenzierzeit

D_F der Differenzierfaktor

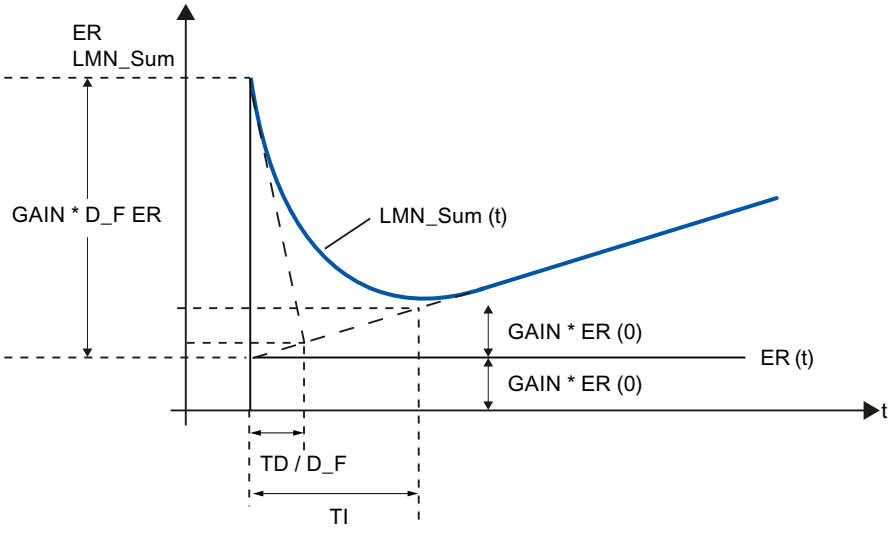

Integrierer (TI, I_ITL_ON, I_ITLVAL)

Bei Handbetrieb wird er wie folgt nachgeführt: LMN_I = LMN - LMN_P - DISV.

Bei Begrenzung des Stellwerts wird der I-Anteil angehalten. Bei einer Regeldifferenz, die den I-Anteil in Richtung innerer Stellbereich bewegt, wird der I-Anteil wieder freigegeben.

Weitere Modifikationen des I-Anteils ergeben sich durch folgende Maßnahmen:

- Abschalten des I-Anteils des Reglers erfolgt mit TI = 0.0
- Abschwächung des P-Anteils bei Sollwertänderungen
- Regelzone
- Die Stellwertgrenzen lassen sich online ändern

Abschwächung des P-Anteils bei Sollwertänderungen (PFAC_SP)

Um Überschwingen zu vermeiden können Sie den P-Anteil über den Parameter "Proportionalfaktor bei Sollwertänderungen" (PFAC_SP) abschwächen. Über PFAC_SP können Sie zwischen 0.0 und 1.0 kontinuierlich wählen, wie stark der P-Anteil bei Sollwertänderungen wirken soll:

- PFAC_SP = 1.0: P-Anteil bei Sollwertänderung voll wirksam
- PFAC_SP = 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Die Abschwächung des P-Anteils wird durch eine Kompensation am I-Anteil erreicht.

Differenzierer (TD, D_F)

- Abschalten des D-Anteils des Reglers erfolgt mit TD = 0.0.
- Bei zugeschaltetem D-Anteil sollte folgende Gleichung eingehalten werden:
  TD = 0.5 * CYCLE * D_F

Parametrierung eines P- oder PD-Reglers mit Arbeitspunkt

Schalten Sie in der Parametrieroberfläche den I-Anteil (TI = 0.0) und evtl. den D-Anteil (TD = 0.0) ab. Führen Sie außerdem folgende Parametrierung durch:

- I_ITL_ON = TRUE
- I_ITLVAL = Arbeitspunkt;

Störgrößenaufschaltung (DISV)

Am Eingang DISV können Sie eine Störgröße additiv aufschalten.

Stellwertberechnung

In folgendem Bild ist das Blockschaltbild der Stellwertberechnung dargestellt:

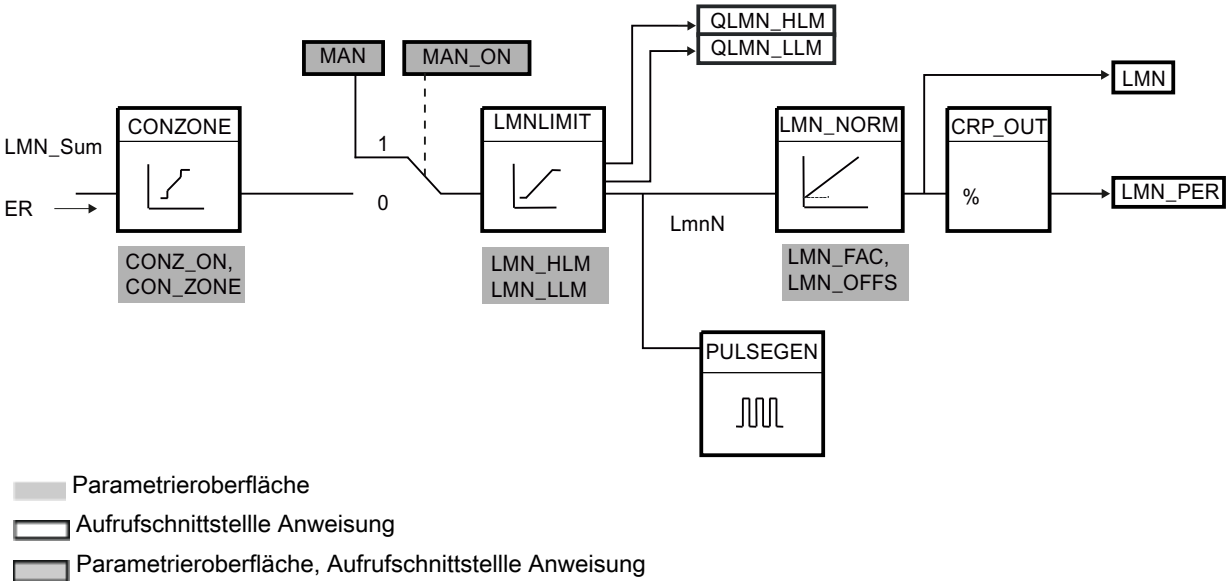

Parametrieroberfläche

Aufrufschnittstellle Anweisung

Parametrieroberfläche, Aufrufschnittstellle Anweisung

Regelzone (CONZ_ON, CON_ZONE)

Wenn CONZ_ON = TRUE arbeitet der Regler mit einer Regelzone. Das bedeutet, dass der Regler nach folgendem Algorithmus angesteuert wird:

- Übersteigt der Istwert PV den Sollwert SP_INT um mehr als CON_ZONE, wird als Stellgröße der Wert LMN_LLM ausgegeben.
- Unterschreitet der Istwert PV den Sollwert SP_INT um mehr als CON_ZONE, wird LMN_HLM ausgegeben.
- Bewegt sich der Istwert PV innerhalb der Regelzone (CON_ZONE), nimmt der Stellwert den Wert vom PID-Algorithmus LMN_Sum an.

Hinweis

Der Wechsel der Stellgröße von LMN_LLM oder LMN_HLM auf LMN_Sum erfolgt unter Einhaltung einer Hysterese von 20% der Regelzone.

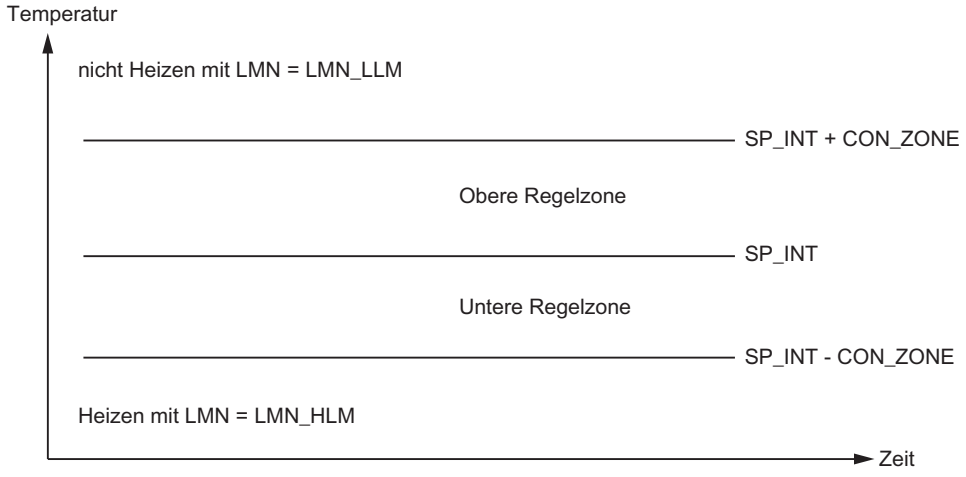

Hinweis

Bevor Sie die Regelzone von Hand einschalten müssen Sie sicherstellen, dass die Regelzonenbreite nicht zu klein eingestellt ist. Bei zu klein eingestellter Regelzonenbreite treten Schwingungen im Verlauf der Stellgröße und des Istwertverlaufs auf.

Vorteil der Regelzone

Beim Eintritt in die Regelzone führt der zugeschaltete D-Anteil zu einem sehr schnellen Reduzieren der Stellgröße. Daher ist die Regelzone nur bei eingeschaltetem D-Anteil sinnvoll. Ohne Regelzone würde im wesentlichen nur der sich reduzierende P-Anteil die Stellgröße reduziert. Die Regelzone führt zu einem schnelleren Einschwingen ohne Über-/Unterschwingen, wenn die ausgegebene minimale oder maximale Stellgröße weit von der für den neuen Arbeitspunkt stationär notwendigen Stellgröße entfernt ist.

Handwertverarbeitung (MAN_ON, MAN)

Sie können zwischen Hand- und Automatikbetrieb umschalten. Bei Handbetrieb wird die Stellgröße einem Handwert nachgeführt.

Der Integrierer (INT) wird intern auf LMN - LMN_P - DISV und der Differenzierer (DIF) auf 0 gesetzt und intern abgeglichen. Das Umschalten in den Automatikbetrieb ist damit stoßfrei.

Hinweis

Während einer Optimierung ist der Parameter MAN_ON nicht wirksam.

Stellwertbegrenzung LMNLIMIT (LMN_HLM, LMN_LLM)

Der Stellwert wird mit der Funktion LMNLIMIT auf die Stellwertgrenzen LMN_HLM und LMN_LLM begrenzt. Das Erreichen der Grenzen wird durch die Meldebits QLMN_HLM und QLMN_LLM angezeigt.

Bei Begrenzung des Stellwerts wird der I-Anteil angehalten. Bei einer Regeldifferenz, die den I-Anteil in Richtung innerer Stellbereich bewegt, wird der I-Anteil wieder freigegeben.

Stellwertgrenzen online ändern

Wird der Stellwertbereich verringert und der neue unbegrenzte Stellwert liegt außerhalb der Grenzen, wird der I-Anteil und damit der Stellwert verschoben.

Der Stellwert wird um die Änderung der Stellwertgrenze verringert. War der Stellwert vor der Änderung unbegrenzt, wird er genau auf die neue Grenze gesetzt (hier beschrieben für die obere Stellwertgrenze).

Stellwertnormierung LMN_NORM (LMN_FAC, LMN_OFFS)

Die Funktion LMN_NORM normiert den Stellwert nach folgender Vorschrift:

LMN = LmnN * LMN_FAC + LMN_OFFS

Sie kann für folgende Zwecke eingesetzt werden:

- Stellwert-Anpassung mit LMN_FAC als Stellwertfaktor und LMN_OFFS als Stellwertoffset

Der Stellwert steht auch im Peripherieformat zur Verfügung. Die Funktion CRP_OUT wandelt den Gleitpunktwert LMN in einen Peripheriewert nach folgender Vorschrift um:

LMN_PER = LMN * 27648/100

Mit den Defaultwerten (LMN_FAC = 1.0 und LMN_OFFS = 0.0) ist die Normierung abgeschaltet. Der effektiv wirksame Stellwert wird am Ausgang LMN ausgegeben.

Reglerparameter speichern SAVE_PAR

Wenn Sie die aktuellen Reglerparameter als brauchbar einstufen, können Sie diese vor einer manuellen Änderung in eigens dafür vorgesehenen Strukturparametern im Instanz-DB der Anweisung TCONT_CP speichern. Bei einer Regleroptimierung werden die gespeicherten Parameter durch die vor der Optimierung gültigen Werte überschrieben.

PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON und CONZONE werden in die Struktur PAR_SAVE geschrieben.

Gespeicherte Reglerparameter zurückladen UNDO_PAR

Die zuletzt gespeicherten Reglerparameter können mit dieser Funktion wieder für den Regler aktiviert werden (nur im Handbetrieb).

Wechsel zwischen PI- und PID-Parametern LOAD_PID (PID_ON)

Nach einer Optimierung werden die PI- und PID-Parameter in den Strukturen PI_CON und PID_CON hinterlegt. Mit LOAD_PID in Abhängigkeit von PID_ON können Sie im Handbetrieb die PI- bzw. PID-Parameter auf die wirksamen Reglerparameter schreiben.

PID-Parameter PID_ON = TRUE	PI-Parameter PID_ON = FALSE
• GAIN = PID_CON.GAIN • TI = PID_CON.TI • TD = PID_CON.TD	• GAIN = PI_CON.GAIN • TI = PI_CON.TI

Hinweis

Die Reglerparameter werden mit UNDO_PAR oder LOAD_PID nur dann zurückgeschrieben, wenn die Reglerverstärkung ungleich Null ist:

Bei LOAD_PID werden die Parameter nur kopiert, falls das jeweilige GAIN <> 0 ist (entweder vom PI- oder PID-Parametersatz). Damit ist der Fall berücksichtigt, dass noch keine Optimierung durchgeführt wurde bzw. PID-Parameter fehlen. War PID_ON = TRUE und PID.GAIN = FALSE, wird PID_ON auf FALSE gesetzt und die PI-Parameter kopiert.

- D_F, PFAC_SP werden durch die Optimierung voreingestellt. Sie können anschließend vom Anwender modifiziert werden. LOAD_PID verändert diese Parameter nicht.
- Die Regelzone wird bei LOAD_PID immer neu berechnet (CON_ZONE = 250/GAIN), auch wenn CONZ_ON = FALSE.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Arbeitsweise Pulsgenerator

Die Funktion PULSEGEN wandelt den analogen Stellwert LmnN durch Pulsweitenmodulation in eine Impulsfolge mit der Periodendauer PER_TM um. PULSEGEN wird mit PULSE_ON = TRUE eingeschaltet und wird im Zyklus CYCLE_P bearbeitet.

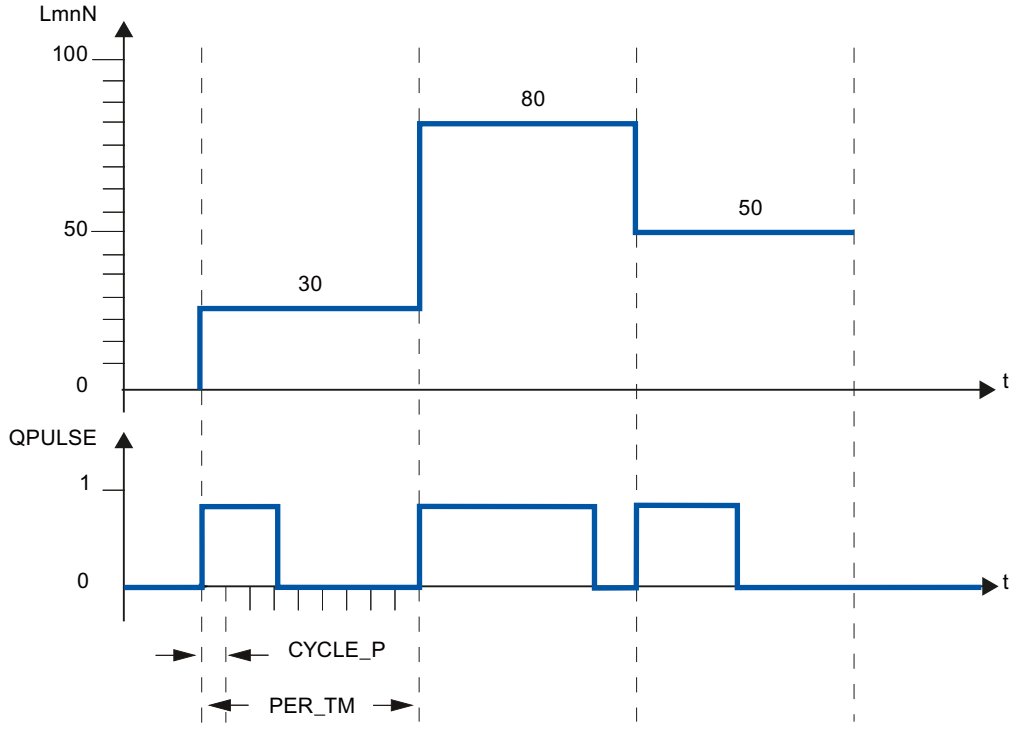

Ein Stellwert LmnN = 30 % und 10 PULSEGEN-Aufrufe pro PER_TM bedeuten also:

- TRUE am Ausgang QPULSE für die ersten drei Aufrufe des PULSEGEN (30 % von 10 Aufrufen)
- FALSE am Ausgang QPULSE für sieben weitere Aufrufe des PULSEGEN (70 % von 10 Aufrufen)

Die Dauer eines Impulses pro Periodendauer ist proportional der Stellgröße und errechnet sich aus:

Impulsdauer = PER_TM * LmnN /100

Durch die Unterdrückung von Mindestimpuls- bzw. -Pausendauer erhält die Umformkennlinie Knickpunkte im Anfangs- und Endbereich.

Im folgenden Bild ist die Zweipunktregelung mit unipolarem Stellwertbereich (0 % bis 100 %) dargestellt:

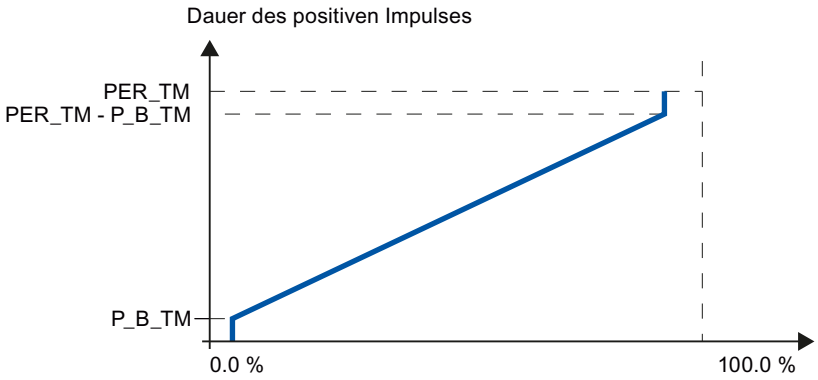

Mindestimpuls- bzw. Mindestpausendauer (P_B_TM)

Kurze Ein- oder Ausschaltzeiten beeinträchtigen die Lebensdauer von Stellgliedern und Stelleinrichtungen. Diese können dadurch vermieden werden, dass eine Mindestimpulsdauer bzw. Mindestpausendauer P_B_TM parametriert wird.

Kleine Absolutwerte der Eingangsgröße LmnN, die eine Impulsdauer kleiner als P_B_TM erzeugen würden, werden unterdrückt.

Große Eingangswerte, die eine Impulsdauer größer als PER_TM - P_B_TM erzeugen würden, werden auf 100 % gesetzt. Hierdurch wird die Dynamik der Impulsformung verringert.

Für die Mindestimpulsdauer bzw. Mindestpausendauer werden Einstellwerte P_B_TM ≤ 0,1 * PER_TM empfohlen.

Die Knickpunkte an den Kennlinien im obigen Bild werden durch die Mindestimpulsdauer bzw. Mindestpausendauer verursacht.

Im folgenden Bild ist das Schaltverhalten des Impulsausgangs dargestellt:

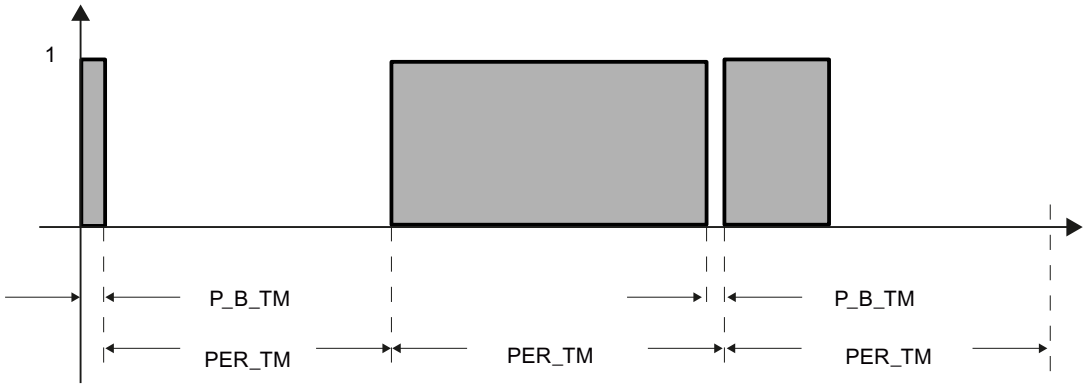

Genauigkeit der Impulsformung

Je kleiner die Impulsrasterbreite CYCLE_P gegenüber der Periodendauer PER_TM ist, desto genauer ist die Pulsweitenmodulation. Für eine hinreichend genaue Regelung sollte folgende Beziehung gelten:

CYCLE_P ≤ PER_TM/50

Damit wird der Stellwert mit einer Auflösung von ≤ 2 % in Impulse gewandelt.

Hinweis

Wenn Sie den Regler im Impulsformerzyklus aufrufen, müssen Sie Folgendes beachten:

Bei Aufruf des Reglers im Impulsformerzyklus wird der Istwert gemittelt. Dies hat zur Folge, dass am Ausgang PV unterschiedliche Werte zum Eingang PV_IN bzw. PV_PER stehen können. Wenn Sie eine Sollwertnachführung realisieren wollen, müssen Sie sich den Istwert am Eingangsparameter PV_IN zu den Aufrufzeitpunkten der Gesamtreglerbearbeitung (QC_ACT = TRUE) speichern. Bei den dazwischen liegenden Aufrufen des Impulsformers versorgen Sie die Eingangsparameter PV_IN und SP_INT mit dem gespeicherten Istwert.

Siehe auch

Beschreibung TCONT_CP (Seite 5820)

Arbeitsweise TCONT_CP (Seite 5821)

Blockschaltbild TCONT_CP (Seite 5833)

Eingangsparameter TCONT_CP (Seite 5834)

Ausgangsparameter TCONT_CP (Seite 5835)

Durchgangsparameter TCONT_CP (Seite 5836)

Statische Variablen TCONT_CP (Seite 5836)

Parameter STATUS_H (Seite 5841)

Parameter STATUS_D (Seite 5842)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Blockschaltbild TCONT_CP

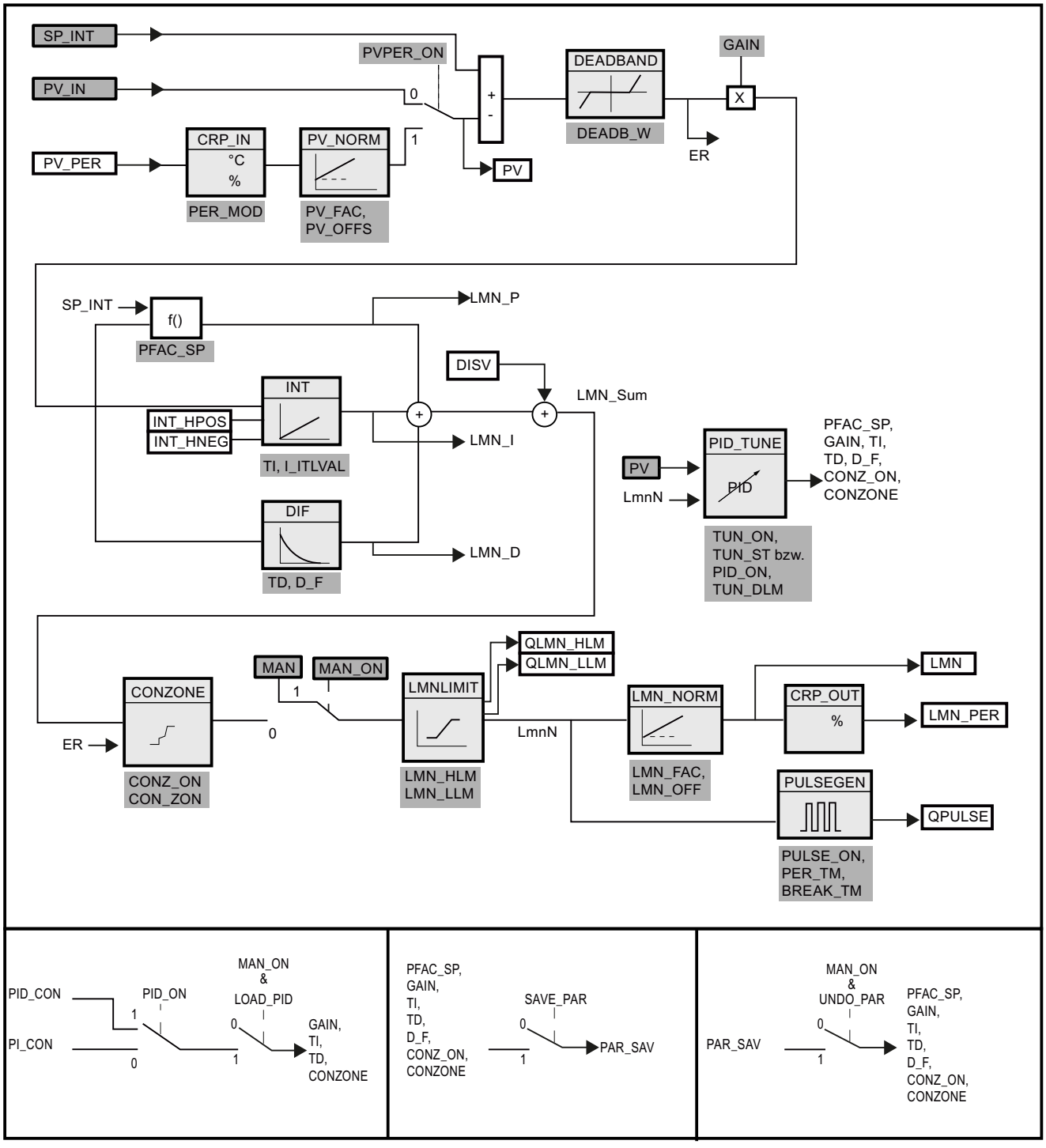

Siehe auch

Beschreibung TCONT_CP (Seite 5820)
Arbeitsweise TCONT_CP (Seite 5821)
Arbeitsweise Pulsgenerator (Seite 5830)
Eingangsparameter TCONT_CP (Seite 5834)
Ausgangsparameter TCONT_CP (Seite 5835)
Durchgangsparameter TCONT_CP (Seite 5836)
Statische Variablen TCONT_CP (Seite 5836)
Parameter STATUS_H (Seite 5841)
Parameter STATUS_D (Seite 5842)
Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Eingangsparameter TCONT_CP

Tabelle 4-204

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PV_IN	0.0	REAL	0.0	Am Eingang "Istwert Eingang" kann ein Inbetriebsetzungswert parametriert oder ein externer Istwert im Gleitpunktformat verschaltet werden. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
PV_PER	4.0	INT	0	Der Istwert in Peripherieformat wird am Eingang "Istwert Peripherie" mit dem Regler verschaltet.
DISV	6.0	REAL	0.0	Für eine Störgrößenaufschaltung wird die Störgröße am Eingang "Störgröße" verschaltet.
INT_HPOS	10.0	BOOL	FALSE	Der Ausgang des Integrierers kann in positive Richtung blockiert werden. Hierzu muss der Eingang INT_HPOS auf TRUE gesetzt werden. Bei einer Kaskaderegelung wird INT_HPOS des Führungsregler mit QLMN_HLM des Folgereglers verschaltet.
INT_HNEG	10.1	BOOL	FALSE	Der Ausgang des Integrierers kann in negative Richtung blockiert werden. Hierzu muss der Eingang INT_HNEG auf TRUE gesetzt werden. Bei einer Kaskaderegelung wird INT_HNEG des Führungsreglers mit QLMN_LLM des Folgereglers verschaltet.
SELECT	12.0	INT	0	Ist der Impulsformer eingeschaltet, gibt es mehrere Möglichkeiten den PID-Algorithmus und Impulsformer aufzurufen: • SELECT = 0: Der Regler wird in einer schnellen Weckalarmebene aufgerufen und es werden PID-Algorithmus und Impulsformer bearbeitet. • SELECT = 1: Der Regler wird im OB1 aufgerufen und es wird nur der PID-Algorithmus bearbeitet. • SELECT = 2: Der Regler wird in einer schnellen Weckalarmebene aufgerufen und es wird nur der Impulsformer bearbeitet. • SELECT = 3: Der Regler wird in einer langsamen Weckalarmebene aufgerufen und es wird nur der PID-Algorithmus bearbeitet.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)
Blockschaltbild TCONT_CP (Seite 5833)
Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Ausgangsparameter TCONT_CP

Tabelle 4-205

Parameter	Adresse	Datentyp	Vorbelegungt	Beschreibung
PV	14.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
LMN	18.0	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert in Gleitpunktformat ausgegeben.
LMN_PER	22.0	INT	0	Der Stellwert in Peripherieformat wird am Ausgang "Stellwert Peripherie" mit dem Regler verschaltet.
QPULSE	24.0	BOOL	FALSE	Der Stellwert wird pulsweitenmoduliert am Ausgang QPULSE ausgegeben.
QLMN_HLM	24.1	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Ausgang QLMN_HLM meldet das Erreichen der oberen Begrenzung.
QLMN_LLM	24.2	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Ausgang QLMN_LLM meldet das Erreichen der unteren Begrenzung.
QC_ACT	24.3	BOOL	TRUE	Der Parameter zeigt an, ob beim nächsten Bausteinaufruf der kontinuierliche Regleranteil bearbeitet wird (nur relevant, wenn SELECT den Wert 0 oder 1 hat).

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)
Blockschaltbild TCONT_CP (Seite 5833)
Parameter STATUS_H (Seite 5841)
Parameter STATUS_D (Seite 5842)
Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Durchgangsparameter TCONT_CP

Tabelle 4-206

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CYCLE	26.0	REAL	0.1 s	Vorgabe der Abtastzeit für den PID-Algorithmus. Der Optimierer berechnet in Phase 1 die Abtastzeit und trägt sie in CYCLE ein. CYCLE > 0.001 s
CYCLE_P	30.0	REAL	0.02 s	An diesem Eingang geben Sie die Abtastzeit für den Impulsformer-Anteil ein. Die Anweisung TCONT_CP berechnet in Phase 1 die Abtastzeit und trägt sie in CYCLE_P ein. CYCLE_P > 0.001 s
SP_INT	34.0	REAL	0.0	Der Eingang "Interner Sollwert" dient zur Vorgabe eines Sollwertes. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
MAN	38.0	REAL	0.0	Der Eingang "Handwert" dient zur Vorgabe eines Handwertes. Im Automatikbetrieb wird er dem Stellwert nachgeführt.
COM_RST	42.0	BOOL	FALSE	Der Baustein hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang COM_RST gesetzt ist.
MAN_ON	42.1	BOOL	TRUE	Ist der Eingang "Handbetrieb einschalten" gesetzt, ist der Regelkreis unterbrochen. Als Stellwert wird der Handwert MAN vorgegeben.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Statische Variablen TCONT_CP

Tabelle 4-207

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
DEADB_W	44.0	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
I_ITLVAL	48.0	REAL	0.0	Der Ausgang des Integrierers kann am Eingang I_ITL_ON gesetzt werden. Am Eingang "Initialisierungswert für I-Anteil" steht der Initialisierungswert. Bei Neustart COM_RST = TRUE wird der I-Anteil auf den Initialisierungswert gesetzt. Zulässig sind Werte von -100 bis 100 %.
LMN_HLM	52.0	REAL	100.0	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. LMN_HLM > LMN_LLM

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMN_LLM	56.0	REAL	0.0	Der Stellwert wird immer auf eine obere und untere Grenze begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. LMN_LLM < LMN_HLM
PV_FAC	60.0	REAL	1.0	Der Eingang "Istwertfaktor" wird mit dem "Istwert Peripherie" multipliziert. Der Eingang dient zur Anpassung des Istwertbereiches.
PV_OFFS	64.0	REAL	0.0	Der Eingang "Istwertoffset" wird zum "Istwert Peripherie" addiert. Der Eingang dient zur Anpassung des Istwertbereiches.
LMN_FAC	68.0	REAL	1.0	Der Eingang "Stellwertfaktor" wird mit dem Stellwert multipliziert. Der Eingang dient zur Anpassung des Stellwertbereiches.
LMN_OFFS	72.0	REAL	0.0	Der Eingang "Stellwertoffset" wird zum Stellwert addiert. Der Eingang dient zur Anpassung des Stellwertbereiches.
PER_TM	76.0	REAL	1.0 s	Am Parameter PER_TM wird die Periodendauer der Pulsweitenmodulation eingegeben. Das Verhältnis Periodendauer zu Abtastzeit des Impulsformers bestimmt die Genauigkeit der Pulsweitenmodulation. PER_TM ≥ CYCLE
P_B_TM	80.0	REAL	0.02 s	Am Parameter "Mindestimpuls- bzw. Mindestpausendauer" kann eine minimale Impuls- bzw. Pausenlänge parametriert werden. P_B_TM wird intern auf > CYCLE_P begrenzt.
TUN_DLMN	84.0	REAL	20.0	Die Prozessanregung für die Regleroptimierung erfolgt durch einen Stellwertsprung von TUN_DLMN. Zulässig sind Werte von -100 bis 100 %.
PER_MODE	88.0	INT	0	An diesem Schalter können Sie den Typ der AE-Baugruppe eingeben. Der Istwert am Eingang PV_PER wird dadurch am Ausgang PV folgendermaßen normiert. • PER_MODE = 0: Thermoelemente; PT100/NI100; Standard PV_PER * 0.1 Einheit: °C, °F • PER_MODE = 1: PT100/NI100; Klima PV_PER * 0.01 Einheit: °C, °F • PER_MODE = 2: Strom/Spannung PV_PER * 100/27648 Einheit: %
PVPER_ON	90.0	BOOL	FALSE	Soll der Istwert von der Peripherie eingelesen werden, so muss der Eingang PV_PER mit der Peripherie verschaltet werden und der Eingang "Istwert Peripherie einschalten" gesetzt werden.
I_ITL_ON	90.1	BOOL	FALSE	Der Ausgang des Integrierers kann auf den Eingang I_ITLVAL gesetzt werden. Hierzu muss der Eingang "I-Anteil setzen" gesetzt werden.
PULSE_ON	90.2	BOOL	FALSE	Mit PULSE_ON = TRUE wird der Impulsformer eingeschaltet.
TUN_KEEP	90.3	BOOL	FALSE	Ein Übergang in den Automatikbetrieb erfolgt erst, wenn TUN_KEEP FALSE wird.
ER	92.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
LMN_P	96.0	REAL	0.0	Der Ausgang "P-Anteil" beinhaltet den Proportionalanteil der Stellgröße.
LMN_I	100.0	REAL	0.0	Der Ausgang "I-Anteil" beinhaltet den Integralanteil der Stellgröße.

Parameter	Adresse	Da-tentyp	Vor-bele-gung	Beschreibung
LMN_D	104.0	REAL	0.0	Der Ausgang "D-Anteil" beinhaltet den Differentialanteil der Stellgröße.
PHASE	108.0	INT	0	Am Ausgang PHASE wird die aktuelle Ablaufphase der Regleroptimierung angezeigt. • PHASE = 0: kein Optimierbetrieb; Automatik- oder Handbetrieb • PHASE = 1: Optimierungsbereitschaft; Parameter prüfen, Warten auf Anregung, Messen der Abtastzeiten • PHASE = 2: Eigentliche Optimierung: Wendepunktsuche bei konstantem Stellwert. Eintrag der Abtastzeit in den Instanz-DB. • PHASE = 3: Berechnen der Prozess-Parameter. Abspeichern der vor der Optimierung gültigen Reglerparameter. • PHASE = 4: Reglerentwurf • PHASE = 5: Nachführen des Reglers auf neue Stellgröße • PHASE = 7: Überprüfung des Streckentyps
STATUS_H	110.0	INT	0	STATUS_H zeigt einen Diagnosewert über die Suche des Wendepunktes beim Heizvorgang an.
STATUS_D	112.0	INT	0	STATUS_D zeigt einen Diagnosewert über den Reglerentwurf beim Heizvorgang an.
QTUN_RUN	114.0	BOOL	0	Die Optimierung wurde durch Aufschalten der Optimierungsstellgröße begonnen und befindet sich noch in Phase 2 (Wendepunktsuche).
PI_CON	116.0	STRUCT		PI Reglerparameter
GAIN	+0.0	REAL	0.0	PI Reglerverstärkung %/phys. Einheit
TI	+4.0	REAL	0.0 s	PI Integrationszeit [s]
PID_CON	124.0	STRUCT		PID Reglerparameter
GAIN	+0.0	REAL	0.0	PID Reglerverstärkung
TI	+4.0	REAL	0.0s	PID Integrationszeit [s]
TD	+8.0	REAL	0.0s	PID Differenzierzeit [s]
PAR_SAVE	136.0	STRUCT		In dieser Struktur werden die PID-Parameter gespeichert.
PFAC_SP	+0.0	REAL	1.0	Proportionalfaktor bei Sollwertänderungen Zulässig sind Werte von 0.0 bis 1.0.
GAIN	+4.0	REAL	0.0	Reglerverstärkung %/phys. Einheit
TI	+8.0	REAL	40.0 s	Integrationszeit [s]
TD	+12.0	REAL	10.0 s	Differenzierzeit [s]
D_F	+16.0	REAL	5.0	Differenzierfaktor Zulässig sind Werte von 5.0 bis 10.0.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CON_ZONE	+20.0	REAL	100.0	Regelzonenbreite Ist die Regeldifferenz größer als die Regelzonenbreite, so wird die obere Stellwertbegrenzung als Stellwert ausgegeben. Ist die Regeldifferenz kleiner als die negative Regelzonenbreite, so wird die untere Stellwertbegrenzung als Stellwert ausgegeben. CON_ZONE ≥ 0.0
CONZ_ON	+24.0	BOOL	FALSE	Regelzone einschalten
PFAC_SP	162.0	REAL	1.0	PFAC_SP gibt den wirksamen P-Anteil bei Sollwertänderung an. Er wird zwischen 0 und 1 eingestellt. • 1: P-Anteil ist auch bei Sollwertänderungen voll wirksam. • 0: P-Anteil ist bei Sollwertänderungen nicht wirksam. Zulässig sind Werte von 0.0 bis 1.0.
GAIN	166.0	REAL	2.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an. Eine Invertierung des Regelsinns wird durch das negative Vorzeichen von GAIN erreicht. %/phys. Einheit
TI	170.0	REAL	40.0 s	Der Eingang "Integrationszeit" (Nachstellzeit) bestimmt das Zeitverhalten des Integrierers.
TD	174.0	REAL	10.0 s	Der Eingang "Differenzierzeit" (Vorhaltezeit) bestimmt das Zeitverhalten des Differenzierers.
D_F	178.0	REAL	5.0	Der Differenzierfaktor bestimmt die Verzögerung des D-Anteils. D_F = Differenzierzeit/"Verzögerung des D-Anteils" Zulässig sind Werte von 5.0 bis 10.0.
CON_ZONE	182.0	REAL	100.0	Ist die Regeldifferenz größer als die Regelzonenbreite, so wird die obere Stellwertbegrenzung als Stellwert ausgegeben. Ist die Regeldifferenz kleiner als die negative Regelzonenbreite, so wird die untere Stellwertbegrenzung als Stellwert ausgegeben. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
CONZ_ON	186.0	BOOL	FALSE	Mit CONZ_ON =TRUE können Sie die Regelzone einschalten.
TUN_ON	186.1	BOOL	FALSE	Wenn TUN_ON=TRUE wird der Stellwert gemittelt bis entweder durch einen Sollwertsprung oder durch TUN_ST=TRUE die Stellwertanregung TUN_DLMN aufgeschaltet wird.
TUN_ST	186.2	BOOL	FALSE	Soll bei der Regleroptimierung am Arbeitspunkt der Sollwert konstant bleiben, wird durch TUN_ST=1 ein Stellwertsprung um TUN_DLMN aufgeschaltet.
UNDO_PAR	186.3	BOOL	FALSE	Lädt die Reglerparameter PFAC_SP, GAIN, TI, TD, D_F CONZ_ON und CON_ZONE aus der Datenstruktur PAR_SAVE (nur im Handbetrieb).
SAVE_PAR	186.4	BOOL	FALSE	Sichert die Reglerparameter PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON und CON_ZONE in die Datenstruktur PAR_SAVE.
LOAD_PID	186.5	BOOL	FALSE	Lädt die Reglerparameter GAIN, TI, TD in Abhängigkeit von PID_ON aus der Datenstruktur PI_CON bzw. PID_CON (nur im Handbetrieb)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PID_ON	186.6	BOOL	TRUE	Am Eingang PID_ON können Sie festlegen, ob der optimierte Regler als PI- oder als PID-Regler arbeiten soll. • PID-Regler: PID_ON = TRUE • PI-Regler: PID_ON = FALSE Es kann jedoch sein, dass bei manchen Streckentypen trotz PID_ON = TRUE nur ein PI-Regler entworfen wird.
GAIN_P	188.0	REAL	0.0	Identifizierte Prozessverstärkung. Beim Streckentyp I wird GAIN_P tendenziell zu klein geschätzt.
TU	192.0	REAL	0.0	Identifizierte Verzugszeit vom Prozess. TU ≥ 3*CYCLE
TA	196.0	REAL	0.0	Identifizierte Ausgleichszeit vom Prozess. Beim Streckentyp I wird TA tendenziell zu klein geschätzt.
KIG	200.0	REAL	0.0	Maximaler Istwertanstieg bei einer Stellgrößenanregung von 0 nach 100 % [1/s] GAIN_P = 0.01 * KIG * TA
N_PTN	204.0	REAL	0.0	Der Parameter gibt die Ordnung der Strecke an. Auch "nicht ganzzahlige Werte" sind möglich. Zulässig sind Werte von 1.01 bis 10.0.
TM_LAG_P	208.0	REAL	0.0	Zeitkonstante eines PTN-Modells (sinnvolle Werte nur für N_PTN >= 2).
T_P_INF	212.0	REAL	0.0	Zeit von der Prozessanregung bis zum Wendepunkt.
P_INF	216.0	REAL	0.0	Istwertänderung von der Prozessanregung bis zum Wendepunkt. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
LMN0	220.0	REAL	0.0	Stellwert zu Beginn der Optimierung Wird in Phase 1 ermittelt (Mittelwert). Zulässig sind Werte von 0 bis 100 %.
PV0	224.0	REAL	0.0	Istwert zu Beginn der Optimierung
PVDT0	228.0	REAL	0.0	Istwertsteigung zu Beginn der Optimierung [1/s] Vorzeichen angepasst.
PVDT	232.0	REAL	0.0	Momentane Istwertsteigung [1/s] Vorzeichen angepasst.
PVDT_MAX	236.0	REAL	0.0	Max. Änderung des Istwertes pro Sekunde [1/s] Maximale Ableitung des Istwertes am Wendepunkt (Vorzeichen angepasst, immer > 0), wird verwendet zur Berechnung von TU und KIG.
NOI_PVDT	240.0	REAL	0.0	Rauschanteil in PVDT_MAX in % Je größer der Rauschanteil, desto ungenauer (sanfter) die Reglerparameter.
NOISE_PV	244.0	REAL	0.0	Absolutes Rauschen im Istwert Differenz zwischen maximalem und minimalem Istwert in Phase 1.
FIL_CYC	248.0	INT	1	Anzahl der Zyklen des Mittelwertfilters Der Istwert wird über FIL_CYC Zyklen gemittelt. FIL_CYC wird bei Bedarf automatisch von 1 bis max. 1024 erhöht.
POI_CMAX	250.0	INT	2	Maximale Anz. Zyklen nach Wendepunkt Diese Zeit wird genutzt, um bei Messrauschen einen weiteren (d. h. besseren) Wendepunkt zu finden. Erst dann wird die Optimierung beendet.
POI_CYCL	252.0	INT	0	Anzahl Zyklen nach Wendepunkt

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Parameter STATUS_H

STATUS_H	Beschreibung	Abhilfe
0	Default bzw. keine oder noch keine neuen Reglerparameter	
10000	Optimierung beendet + geeignete Reglerparameter gefunden	
2xxxx	Optimierung beendet + Reglerparameter unsicher	
2xx2x	Wendepunkt nicht erreicht (nur bei Anregung über Sollwertsprung)	Falls Regler schwingt, Reglerparameter abschwächen oder Versuch wiederholen mit kleinerer Stellwertdifferenz TUN_DLMN.
2x1xx	Schätzfehler (TU < 3*CYCLE)	CYCLE verringern und Versuch wiederholen. Sonderfall reine PT1-Strecke: Versuch nicht wiederholen, Reglerparameter evtl. abschwächen.
2x3xx	Schätzfehler TU zu groß	Versuch wiederholen mit besseren Bedingungen.
21xxx	Schätzfehler N_PTN < 1	Versuch wiederholen mit besseren Bedingungen.
22xxx	Schätzfehler N_PTN > 10	Versuch wiederholen mit besseren Bedingungen.
3xxxx	Optimierung abgebrochen in Phase 1 durch fehlerhafte Parametrierung:	
30002	Effektive Stellwertdifferenz < 5 %	Stellwertdifferenz TUN_DLMN korrigieren.
30005	Die Abtastzeiten CYCLE und CYCLE_P weichen um mehr als 5% von den gemessenen Werten ab.	Vergleichen Sie CYCLE und CYCLE_P mit der Zykluszeit der Weckalarmebene und beachten Sie evt. vorhandene Aufrufverteiler. Prüfen Sie die Auslastung der CPU. Eine zu stark ausgelastete CPU führt zu verlängerten Abtastzeiten, die nicht zu CYCLE bzw. CYCLE_P passen.

Hinweis

Wenn Sie die Optimierung in Phase 1 oder 2 abbrechen, wird STATUS_H = 0. STATUS_D zeigt jedoch immer noch den Zustand der letzten Reglerberechnung an.

Je größer der Wert von STATUS_D, desto höher die Ordnung der Regelstrecke, desto größer das Verhältnis TU/TA und desto sanfter die Reglerparameter.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

Parameter STATUS_D

STATUS_D	Beschreibung
0	Es wurden keine Reglerparameter berechnet
110	N_PTN <= 1.5 Streckentyp I schnell
121	N_PTN > 1.5 Streckentyp I
200	N_PTN > 1.9 Streckentyp II (Übergangsbereich)
310	N_PTN >= 2.1 Streckentyp III schnell
320	N_PTN > 2.6 Streckentyp III
111, 122, 201, 311, 321	Parameter wurden von Phase 7 korrigiert.

Hinweis

Je größer der Wert von STATUS_D, desto höher die Ordnung der Regelstrecke, desto größer das Verhältnis TU/TA und desto sanfter die Reglerparameter.

Siehe auch

Arbeitsweise Pulsgenerator (Seite 5830)

Blockschaltbild TCONT_CP (Seite 5833)

Unterschiede zu TCONT_CP S7-300/400 (Seite 3097)

TCONT_S

Beschreibung TCONT_S

Die Anweisung TCONT_S dient zum Regeln von technischen Temperatur-Prozessen mit binären Stellwertausgangssignalen für integrierende Stellglieder auf den Automatisierungssystemen SIMATIC S7. Die Arbeitsweise basiert auf dem PI-Regelalgorithmus des Abtastreglers. Der Schrittregler arbeitet ohne Stellungsrückmeldung.

Anwendung

Sie können den Regler auch in einer Reglerkaskade als unterlagerten Stellungsregler einsetzen. Über den Sollwerteingang SP_INT geben Sie die Stellgliedposition vor. In diesem Fall müssen Sie den Istwerteingang und den Parameter TI (Integrationszeit) auf Null setzen. Anwendungsfall ist z. B. eine Temperaturregelung mit Heizleistungsregelung über Puls-Pause-Ansteuerung und Kühlleistungsregelung über eine Ventilklappe. Um die Klappe ganz zu schließen, sollte die Stellgröße (ER*GAIN) negativ werden.

Aufruf

Die Anweisung TCONT_S muss äquidistant aufgerufen werden. Benutzen Sie hierzu eine Weckalarmebene (z.B. OB35 bei S7-300). Die Abtastzeit wird am Parameter CYCLE vorgegeben.

Wenn Sie die Anweisung TCONT_S als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen TCONT_S direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Abtastzeit CYCLE

Die Abtastzeit CYCLE muss mit der Zeitdifferenz zwischen zwei Aufrufen (Zykluszeit des Weckalarm-OB unter Beachtung der Untersetzungen) übereinstimmen.

Die Regler-Abtastzeit sollte 10 % der ermittelten Integrationszeit des Reglers (TI) nicht übersteigen. In der Regel müssen Sie aber die Abtastzeit weit kleiner einstellen, um die geforderte Genauigkeit des Schrittreglers sicherzustellen.

Geforderte Genauigkeit G	MTR_TM	CYCLE = MTR_TM*G	Kommentar
0,5 %	10 s	0,05 s	Die Abtastzeit wird bestimmt durch die geforderte Genauigkeit des Schrittreglers.

Anlauf

Die Anweisung TCONT_S hat eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangsparameter COM_RST = TRUE gesetzt ist. Der Baustein setzt nach dem Bearbeiten der Initialisierungsroutine COM_RST wieder auf FALSE zurück. Alle Ausgänge werden auf ihre Anfangswerte gesetzt. Wenn Sie eine Initialisierung bei Neustart der CPU wünschen, rufen Sie den Baustein im OB100 mit COM_RST = TRUE auf.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Arbeitsweise TCONT_S

Sollwertzweig

Der Sollwert wird am Eingang SP_INT im Gleitpunktformat physikalisch oder in Prozent eingegeben. Sollwert und Istwert müssen an der Regeldifferenzbildung die gleiche Einheit besitzen.

Istwertauswahl (PVPER_ON)

Der Istwert kann abhängig von PVPER_ON im Peripherie- oder im Gleitpunktformat eingelesen werden.

PVPER_ON	Istwerteingabe
TRUE	Der Istwert wird über die Analogperipherie (PEW xxx) am Eingang PV_PER eingelesen.
FALSE	Der Istwert wird im Gleitpunktformat am Eingang PV_IN eingelesen.

Istwertformatumwandlung CRP_IN (PER_MODE)

Die Funktion CRP_IN wandelt den Peripheriewert PV_PER abhängig vom Schalter PER_MODE in ein Gleitpunktformat nach folgender Vorschrift um:

PER_MODE	Ausgang von CRP_IN	Analogeingabe-Typ	Einheit
0	PV_PER * 0.1	Thermoelemente; PT100/NI100; Standard	°C;°F
1	PV_PER * 0.01	PT100/NI100; Klima;	°C;°F
2	PV_PER * 100/27648	Spannung/Strom	%

Istwertnormierung PV_NORM (PF_FAC, PV_OFFS)

Die Funktion PV_NORM berechnet den Ausgang von CRP_IN nach folgender Vorschrift:

"Ausgang von PV_NORM" = "Ausgang von CRP_IN" * PV_FAC + PV_OFFS

Sie kann für folgende Zwecke eingesetzt werden:

- Istwert-Anpassung mit PV_FAC als Istwertfaktor und PV_OFFS als Istwertoffset
- Normierung von Temperatur nach Prozent
  Sie wollen den Sollwert in Prozent eingeben und müssen nun den gemessenen Temperaturwert in Prozent umrechnen.
- Normierung von Prozent nach Temperatur
  Sie wollen den Sollwert in der physikalischen Größe Temperatur eingeben und müssen nun den gemessenen Spannungs-/Strom-Wert in eine Temperatur umrechnen.

Berechnung der Parameter:

- PV_FAC = Bereich von PV_NORM/Bereich von CRP_IN;
- PV_OFFS = UG(PV_NORM) - PV_FAC * UG(CRP_IN);
  mit UG: Untergrenze

Mit den Defaultwerten (PV_FAC = 1.0 und PV_OFFS = 0.0) ist die Normierung abgeschaltet. Der effektiv wirksame Istwert wird am Ausgang PV ausgegeben.

Beispiel zur Istwertnormierung

Wenn Sie den Sollwert in Prozent vorgeben wollen und Sie einen Temperaturbereich von -20 bis 85 °C an CRP_IN anliegen haben, müssen Sie den Temperaturbereich in Prozent umnormieren.

Das folgende Bild stellt ein Beispiel für die Anpassung des Temperaturbereichs -20 bis 85 °C auf intern 0 bis 100 % dar:

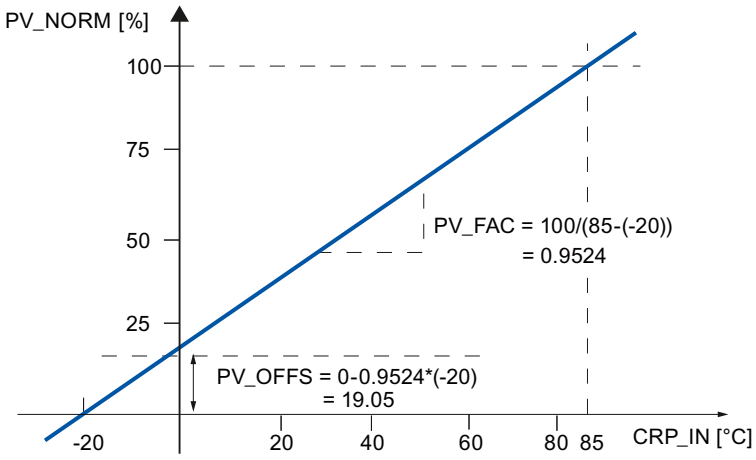

Regeldifferenzbildung

Die Differenz von Soll- und Istwert ergibt die Regeldifferenz vor der Totzone.

Soll- und Istwert müssen in der gleichen Einheit vorliegen.

Totzone (DEADB_W)

Zur Unterdrückung einer kleinen Dauerschwingung aufgrund der Stellgrößen-Quantisierung (z. B. bei einer Pulsweitenmodulation mit PULSEGEN) wird die Regeldifferenz über eine Totzone (DEADBAND) geleitet. Bei DEADB_W = 0.0 ist die Totzone ausgeschaltet.

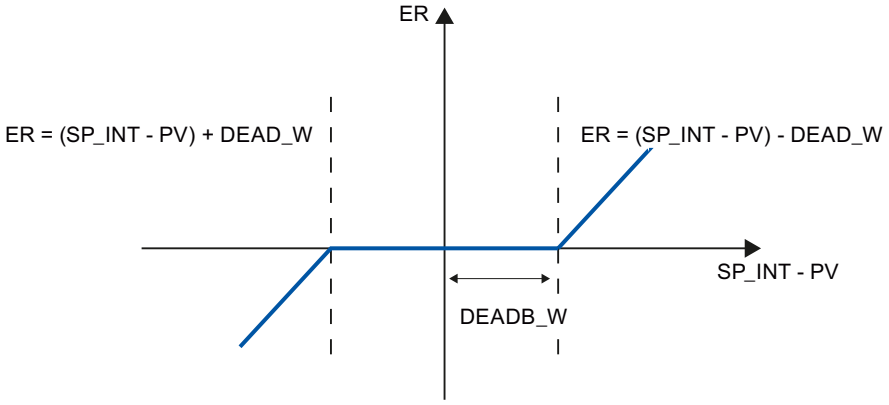

PI-Schrittregler-Algorithmus

Die Anweisung TCONT_S arbeitet ohne Stellungsrückmeldung. Der I-Anteil des PI-Algorithmus und die gedachte Stellungsrückmeldung werden in einem Integrator (INT) berechnet und als Rückführungswert mit dem verbliebenen P-Anteil verglichen. Die Differenz geht auf ein Dreipunktglied (THREE_ST) und einen Impulsformer (PULSEOUT), der die Impulse für das Stellventil bildet. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Abschwächung des P-Anteils bei Sollwertänderungen

Um Überschwingen zu vermeiden können Sie den P-Anteil über den Parameter "Proportionalfaktor bei Sollwertänderungen" (PFAC_SP) abschwächen. Über PFAC_SP können Sie nun zwischen 0.0 und 1.0 kontinuierlich wählen, wie stark der P-Anteil bei Sollwertänderungen wirken soll:

- PFAC_SP = 1.0: P-Anteil bei Sollwertänderung voll wirksam
- PFAC_SP = 0.0: Kein P-Anteil bei Sollwertänderung

Ein Wert von PFAC_SP < 1.0 kann wie beim kontinuierlichen Regler das Überschwingen reduzieren, falls die Motorlaufzeit MTR_TM klein gegenüber der Ausgleichszeit TA ist und das Verhältnis TU/TA < 0.2 ist. Erreicht MTR_TM 20 % von TA, ist nur noch eine geringe Verbesserung zu erzielen.

Störgrößenaufschaltung

Am Eingang DISV kann eine Störgröße additiv aufgeschaltet werden.

Handwertverarbeitung (LMNS_ON, LMNUP, LMNDN)

Mit LMNS_ON kann zwischen Hand- und Automatikbetrieb umgeschaltet werden. Bei Handbetrieb wird das Stellglied angehalten und der Integrierer (INT) wird intern auf 0 gesetzt. Über LMNUP und LMNDN kann das Stellglied AUF und ZU gefahren werden. Das Umschalten in den Automatikbetrieb erfolgt stoßbehaftet. Die anstehende Regeldifferenz führt über GAIN zu einer sprungförmigen Änderung der internen Stellgröße. Durch das integral wirkende Stellglied wird jedoch nur eine rampenförmige Ansteuerung des Prozesses bewirkt.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Blockschaltbild TCONT_S

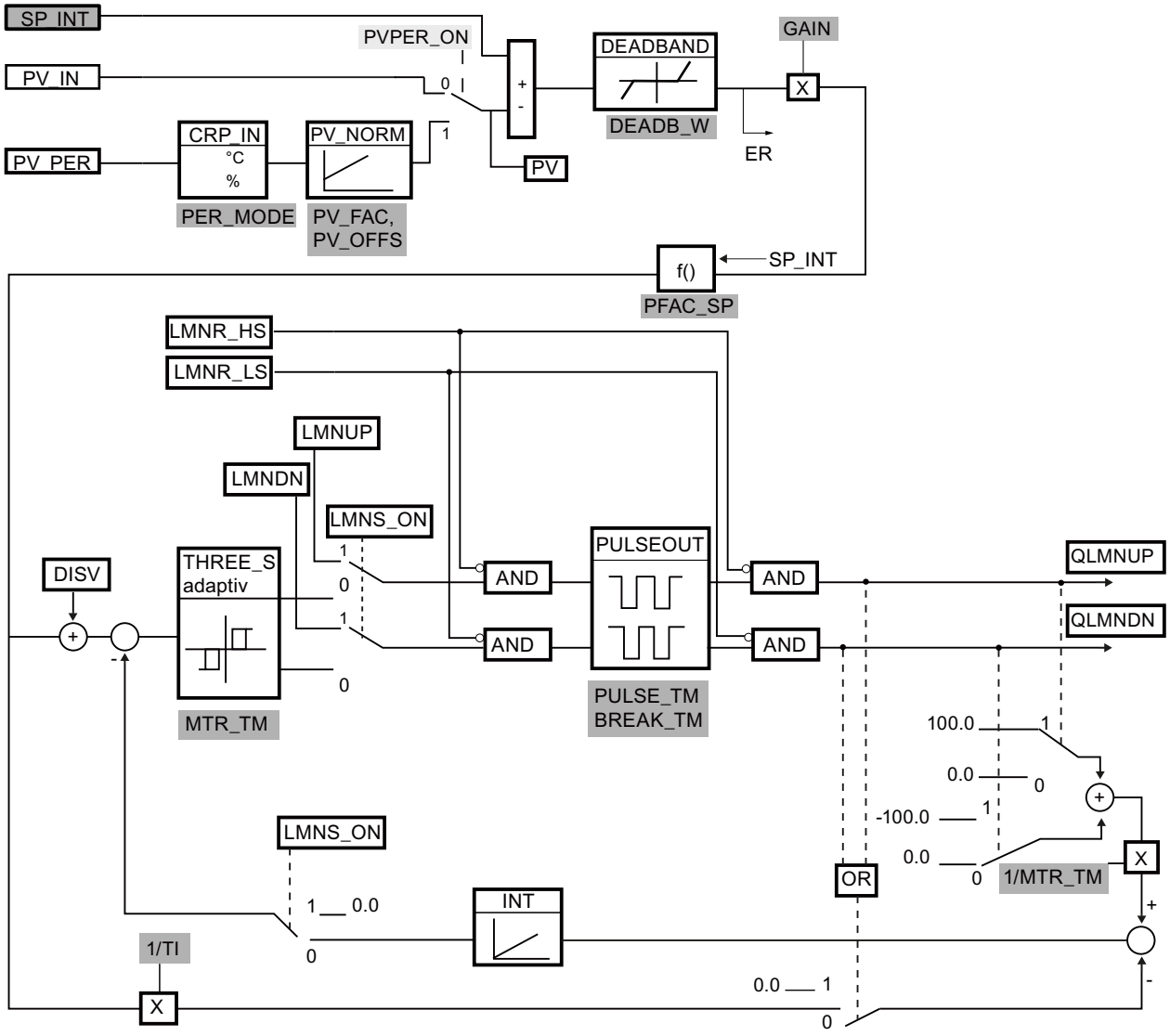

Parametrieroberfläche

Aufrufschnittstellle Anweisung

Parametrieroberfläche, Aufrufschnittstellle Anweisung

Siehe auch

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Beschreibung TCONT_S (Seite 5842)

Arbeitsweise TCONT_S (Seite 5844)

Eingangsparameter TCONT_S (Seite 5849)

Ausgangsparameter TCONT_S (Seite 5850)

Durchgangsparameter TCONT_S (Seite 5850)

Statische Variablen TCONT_S (Seite 5850)

Eingangsparameter TCONT_S

Tabelle 4-208

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CYCLE	0.0	REAL	0.1 s	An diesem Eingang geben Sie die Abtastzeit für den Regler ein. CYCLE ≥ 0.001
SP_INT	4.0	REAL	0.0	Der Eingang "Interner Sollwert" dient zur Vorgabe eines Sollwertes. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
PV_IN	8.0	REAL	0.0	Am Eingang "Istwert Eingang" kann ein Inbetriebsetzungswert parametriert oder ein externer Istwert im Gleitpunktformat verschaltet werden. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.
PV_PER	12.0	INT	0	Der Istwert in Peripherieformat wird am Eingang "Istwert Peripherie" mit dem Regler verschaltet.
DISV	14.0	REAL	0.0	Für eine Störgrößenaufschaltung wird die Störgröße am Eingang "Störgröße" verschaltet.
LMNR_HS	18.0	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "Oberes Anschlagsignal der Stellungsrückmeldung" verschaltet. • LMNR_HS=TRUE: Das Stellventil befindet sich am oberen Anschlag.
LMNR_LS	18.1	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "Unteres Anschlagsignal der Stellungsrückmeldung" verschaltet. • LMNR_LS=TRUE: Das Stellventil befindet sich am unteren Anschlag.
LMNS_ON	18.2	BOOL	TRUE	Am Eingang "Handbetrieb der Stellwertsignale einschalten" wird die Stellwertsignalverarbeitung auf Hand geschaltet.
LMNUP	18.3	BOOL	FALSE	Bei Handbetrieb der Stellwertsignale wird am Eingang "Stellwertsignal Hoch" das Ausgangssignal QLMNUP bedient.
LMNDN	18.4	BOOL	FALSE	Bei Handbetrieb der Stellwertsignale wird am Eingang "Stellwertsignal Tief" das Ausgangssignal QLMNDN bedient.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Ausgangsparameter TCONT_S

Tabelle 4-209

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNUP	20.0	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Hoch" gesetzt, soll das Stellventil geöffnet werden.
QLMNDN	20.1	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Tief" gesetzt, soll das Stellventil geschlossen werden.
PV	22.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	26.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Durchgangsparameter TCONT_S

Tabelle 4-210

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	30.0	BOOL	FALSE	Der Baustein hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang COM_RST gesetzt ist.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Statische Variablen TCONT_S

Tabelle 4-211

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PV_FAC	32.0	REAL	1.0	Der Eingang "Istwertfaktor" wird mit dem Istwert multipliziert. Der Eingang dient zur Anpassung des Istwertbereiches.
PV_OFFS	36.0	REAL	0.0	Der Eingang "Istwertoffset" wird mit dem Istwert addiert. Der Eingang dient zur Anpassung des Istwertbereiches. Die gültigen Werte sind abhängig von den eingesetzten Sensoren.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
DEADB_W	40.0	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W ≥ 0.0
PFAC_SP	44.4	REAL	1.0	PFAC_SP gibt den wirksamen P-Anteil bei Sollwertänderung an. • 1: P-Anteil ist bei Sollwertänderungen voll wirksam • 0: P-Anteil ist bei Sollwertänderungen nicht wirksam Zulässig sind Werte von 0.0 bis 1.0.
GAIN	48.0	REAL	2.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an. Eine Invertierung des Regelsinns wird durch das negative Vorzeichen von GAIN erreicht. %/phys. Einheit
TI	52.0	REAL	40.0 s	Der Eingang "Integrationszeit" (Nachstellzeit) bestimmt das Zeitverhalten des Integrierers.
MTR_TM	56.0	REAL	30 s	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils von Anschlag zu Anschlag eingetragen. MTR_TM ≥ CYCLE
PULSE_TM	60.0	REAL	0.0 s	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden.
BREAK_TM	64.0	REAL	0.0 s	Am Parameter "Mindestpausendauer" kann eine minimale Pausenlänge parametriert werden.
PER_MODE	68.0	INT	0	An diesem Schalter können Sie den Typ der AE-Baugruppe eingeben. Der Istwert am Eingang PV_PER wird dadurch am Ausgang PV folgendermaßen normiert. • PER_MODE = 0: Thermoelemente; PT100/NI100; Standard PV_PER * 0.1 Einheit: °C, °F • PER_MODE = 1: PT100/NI100; Klima PV_PER * 0.01 Einheit: °C, °F • PER_MODE = 2: Strom/Spannung PV_PER * 100/27648 Einheit: %
PVPER_ON	70.0	BOOL	FALSE	Soll der Istwert von der Peripherie eingelesen werden, so muss der Eingang PV_PER mit der Peripherie verschaltet werden und der Eingang "Istwert Peripherie einschalten" gesetzt werden.

Siehe auch

Blockschaltbild TCONT_S (Seite 5848)

Unterschiede zu TCONT_S S7-300/400 (Seite 3098)

Integrierte Systemfunktionen

CONT_C_SF

CONT_C_SF

Die Anweisung CONT_C_SF ist in den S7-300 Compact-CPUs integriert. Die Anweisung muss beim Laden in die S7-300 CPU nicht übertragen werden. Der Funktionsumfang entspricht der Anweisung CONT_C.

Siehe auch

Beschreibung CONT_C (Seite 5800)

Arbeitsweise CONT_C (Seite 5801)

Blockschaltbild CONT_C (Seite 5803)

Eingangsparameter CONT_C (Seite 5804)

Ausgangsparameter CONT_C (Seite 5805)

CONT_S_SF

CONT_S_SF

Die Anweisung CONT_S_SF ist in den S7-300 Compact-CPUs integriert. Die Anweisung muss beim Laden in die S7-300 CPU nicht übertragen werden. Der Funktionsumfang entspricht der Anweisung CONT_S.

Siehe auch

Beschreibung CONT_S (Seite 5806)

Arbeitsweise CONT_S (Seite 5807)

Blockschaltbild CONT_S (Seite 5808)

Eingangsparameter CONT_S (Seite 5809)

Ausgangsparameter CONT_S (Seite 5810)

PULSEGEN_SF

PULSEGEN_SF

Die Anweisung PULSEGEN_SF ist in den S7-300 Compact-CPUs integriert. Die Anweisung muss beim Laden in die S7-300 CPU nicht übertragen werden. Der Funktionsumfang entspricht der Anweisung PULSEGEN.

Siehe auch

Beschreibung PULSEGEN (Seite 5810)
Arbeitsweise PULSEGEN (Seite 5812)
Betriebsarten PULSEGEN (Seite 5815)
Dreipunktregelung (Seite 5815)
Zweipunktregelung (Seite 5817)
Eingangsparameter PULSEGEN (Seite 5819)
Ausgangsparameter PULSEGEN (Seite 5820)

PID Self Tuner

TUN_EC

Beschreibung TUN_EC

Die Anweisung TUN_EC dient zur Einstellung eines kontinuierlichen PID-Reglers. Sie verschalten TUN_EC je nach Regler mit einer der folgenden Anweisungen:

- FM 355 C, FM 355 S Impulsregler, FM 455 C oder FM 455 S Impulsregler
  - PID_FM
- PID Basisfunktionen
  - CONT_C mit oder ohne PULSEGEN
- Standard PID Control (Optionspaket PID Professional)
  - PID_CP
- Modular PID Control (Optionspaket PID Professional)
  - PID
  - LMNGEN_C

Aufruf

TUN_EC wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

Bei ADAPT1ST = TRUE wird die Initialisierungsroutine der Anweisung TUN_EC durchlaufen. Dabei werden folgende Vorbelegungen gemacht:

- Es wird auf PI-Regler geschaltet. Hierzu werden die Parameter von PI_CON an den Regler übertragen. Haben die Parameter PI_CON.GAIN und PI_CON.TI beide den Wert 0.0, dann werden Sie mit PI_CON.GAIN = 1.5 und PI_CON.TI = 3600.0 s vorbelegt.
- PROCESS.GAIN = 999.0

- RATIOFAC = 1.0
- STATUS_H = 0; STATUS_D = 0; STATUS_C = 0

Hinweis

Wenn nicht explizit gewünscht, rufen Sie bei Neustart (Warmstart) der CPU TUN_ES **nicht** wie bei anderen Anweisungen üblich in seiner Initialisierungsroutine auf, da sonst alle Prozessparameter verloren gehen.

Wenn Sie die Anweisung TUN_EC als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Inbetriebnahmeoberfläche zur Verfügung. Sie müssen TUN_EC über eine Beobachtungstabelle in Betrieb nehmen.

Abtastzeit CYCLE

Die Abtastzeit geben Sie am Parameter CYCLE vor. Die Abtastzeit sollte nicht größer als 10 % der ermittelten Integrationszeit des Reglers sein. Sie muss mit der Zeitdifferenz zwischen zwei Aufrufen (Zykluszeit des Weckalarm-OB unter Beachtung der Untersetzungen) übereinstimmen.

Für eine zuverlässige Identifikation der Prozessparameter sollte die kleinste Zeitkonstante des Prozesses (die Verzugszeit) mindestens 10 bis 20 mal abgetastet werden.

In Verbindung mit den Reglerbaugruppen FM355/455 muss die Abtastzeit von TUN_EC ungefähr gleich der resultierenden Zykluszeit der Regler auf der FM sein (Parametereinstellung >Baugruppenparameter).

Verhalten im Fehlerfall

Das Fehlermeldewort RET_VAL wird vom Baustein nicht ausgewertet.

Siehe auch

Arbeitsweise TUN_EC (Seite 5857)

Eingangsparameter TUN_EC (Seite 5862)

Ausgangsparameter TUN_EC (Seite 5864)

Durchgangsparameter TUN_EC (Seite 5866)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Anwendungsbereich TUN_EC

Den Self-Tuner können Sie besonders gut einsetzen bei:

- Temperaturregelungen, daneben aber auch bei
- Füllstandsregelungen und
- Durchflussregelungen

Bei Durchflussregelungen muss man unterscheiden, ob nur das Stellventil zu regeln ist oder ob sich nach dem Stellventil ein verzugsbehafteter Prozess anschließt. Bei reiner Stellventilregelung ist der Self-Tuner nicht anwendbar.

Anforderungen an den Prozess

Der Prozess muss folgenden Bedingungen genügen:

- **Stabiles, verzugsbehaftetes, asymptotisches Einschwingverhalten**
  Nach einem Sprung der Stellgröße muss der Istwert in einen stationären Zustand übergehen. Ausgeschlossen sind damit Prozesse, die schon ohne Regelung ein oszillatorisches Verhalten zeigen, sowie Regelstrecken ohne Ausgleich (Integrator in der Regelstrecke).
- **Nicht zu große Verzugszeiten**
  Der Einsatzbereich lässt sich anhand des Verhältnisses von Verzugszeit t_u und Ausgleichszeit t_a spezifizieren. Die Verzugszeit umfasst auch eine evtl. vorhandene Todzeit.
  Die Ersteinstellung bzw. Adaption ist bevorzugt ausgelegt für den Bereich $t_u < 1/10\ t_a$, in dem die meisten Temperaturstrecken liegen.
  Für den Bereich besonders großer Verzugszeiten $1/10\ t_a < t_u < 1/3\ t_a$ ist meistens noch der Entwurf eines brauchbaren PI-Reglers möglich.
- **Hinreichend lineares Verhalten über einen hinreichend großen Arbeitsbereich**
  Das heißt, dass sowohl für die Identifikation als auch für den normalen Regelungsbetrieb nichtlineare Effekte innerhalb dieses Arbeitsbereichs vernachlässigt werden dürfen. Allerdings ist es möglich, beim Wechsel des Arbeitspunktes den Prozess neu zu identifizieren, wenn der Adaptionsvorgang wieder in einer kleinen Umgebung des neuen Arbeitspunktes durchgeführt wird und während der Adaption die Nichtlinearität nicht durchfahren wird.
  Wenn bestimmte statische Nichtlinearitäten (z. B. Ventilkennlinien) bekannt sind, ist es in jedem Fall sinnvoll, sie vorab mit einem Polygonzug zu kompensieren, um das Prozessverhalten zu linearisieren.
- **Geringe Störeinflüsse bei Temperaturprozessen**
  Störeinflüsse wie die Wärmeübertragung an benachbarten Zonen oder die Wärmezunahme bzw. -abnahme durch Aggregatzustandswechsel dürfen den Gesamttemperaturprozess nicht zu stark beeinflussen. Gegebenenfalls ist eine Adaption am Arbeitspunkt notwendig. Z. B. müssen bei der Optimierung von Zonen eines Extruders alle Zonen gleichzeitig aufgeheizt werden.

- **Qualität der Messsignale**
  Die Messsignale müssen in ausreichender Qualität vorliegen, d. h. das Verhältnis von Störsignal zu Nutzsignal muss hinreichend klein sein.
- **Nicht zu große Prozessverstärkungen**
  Die Prozessverstärkungen dürfen nicht zu groß sein. Eine Normierung der Istwerte wird nicht verlangt. Die Prozessverstärkung K ist also ggf. mit physikalischen Einheiten behaftet, z. B.:
  K =ΔPV/ΔLMN, [K]= °C/ %
  Der endgültige Reglerentwurf basiert auf einer Berechnung der Prozessverstärkung K und kann damit prinzipiell beliebige Werte von K kompensieren. In der Lernphase ist K jedoch zunächst unbekannt, und bei extremen Kombinationen von Verstärkung und Lernsprunghöhe können Überschwinger nicht vermieden werden. Durch Reduzierung des Parameters LHLM_TUN im Heizvorgang oder Erhöhung von LLLM_TUN im Kühlvorgang wird auch der Überschwinger reduziert.

Anforderungen an Prozesse mit integral wirkenden Stellventilen

Bei Prozessen mit integral wirkenden Stellventilen gibt es neben den oben genannten noch weitere Anforderungen:

Die Motorlaufzeit des Stellventils muss kleiner sein als die Zeit, die benötigt wird, um nach einem Stellwertsprung einen Wendepunkt zu finden (vgl. Bild Prozessantwort). Ist dies nicht der Fall, so handelt es sich oftmals um eine Durchflussregelung, bei der nur das Stellventil als dominierender Prozessanteil wirksam ist. Der Einsatz des Self-Tuners ist dann nicht ratsam. Sie können den PI-Schrittregler dann nach folgender Faustformel von Hand einstellen:

GAIN = 1, TI = Motorstellzeit

Siehe auch

Arbeitsweise TUN_EC (Seite 5857)

Eingangsparameter TUN_EC (Seite 5862)

Ausgangsparameter TUN_EC (Seite 5864)

Durchgangsparameter TUN_EC (Seite 5866)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Arbeitsweise TUN_EC

Zur Steuerung bzw. Optimierung des Reglers verfügt TUN_EC über folgende Funktionen:

- **Handbetrieb**
  Der Self-Tuner zwingt den Regler in den Handbetrieb und übergibt ihm seine Handparameter.
- **Hand/Automatik-Umschaltung mit prädiktiver Stellwertvorgabe**
  Nach einer erfolgreichen Regleroptimierung kann der Self-Tuner beim Deaktivieren des Handbetriebs den Regler durch Aufschalten eines prädiktiven Stellwertes so in den Automatikbetrieb umschalten, dass ein schnelles Einregeln in den Sollwert erfolgt. Diese Art der Umschaltung ist stoßbehaftet. Sie können auch stoßfreie Umschaltung am Self-Tuner einstellen.
- **Regleroptimierung**
  Bei der Optimierung der Reglerparameter wird zuerst die Strecke identifiziert. Aufgrund der Streckencharakteristik wird ein Reglerentwurf vorgenommen; darauf basierend, werden die neuen Reglerparameter berechnet und diese für den Regler aktiviert:
  - **Ersteinstellung**
    Die Ersteinstellung kann nur bei einem positiven Sollwertsprung erfolgen. Üblicherweise wird sie bei der ersten Inbetriebnahme oder bei einer gravierenden Veränderung des Streckenverhaltens eingesetzt.
  - **Nachoptimierung**
    Beim Wechsel des Arbeitspunkts oder bei Änderungen des Prozessverhaltens kann der Regler online während eines positiven Sollwertsprungs nachoptimiert werden.
  - **Kühloptimierung**
    Für Regelungen, die mit zwei entgegengesetzt wirkenden Stellgliedern arbeiten (z. B. bei Temperaturregelung: Heiz- und Kühlstellglied), ermittelt der Self-Tuner bei einem negativen Stellwertsprung das Verhältnis der Prozessverstärkungen unter Einfluss des zweiten Stellglieds (um beim Beispiel Temperaturregelung zu bleiben: unter Einfluss des Kühlstellglieds).

  Optimierung des Führungsverhaltens
  Der Reglerentwurf ist grundsätzlich auf optimales Störverhalten ausgelegt. Die daraus resultierenden "scharfen" Parameter würden zu Überschwingern von 10 % bis 20 % der Sprunghöhe bei Sollwertsprüngen führen.
  Eine erste Abhilfe stellt die **Regelzone** dar. Zur Optimierung des Anregelverhaltens bei großen Sollwertsprüngen berechnet der Self-Tuner bei einer Optimierung eine Regelzonenbreite. Innerhalb der Zone ist der Regler aktiv. Außerhalb der Zone steuert der Self-Tuner den Regler mit der maximalen bzw. mit der minimalen Stellgröße des Reglers.
  Eine weitere Abhilfe stellt die **Struktur-Zerlegung** dar. Dabei werden der P- bzw. D-Anteil in die Rückführung gelegt. Somit wirkt eine Sollwertänderung nur am I-Anteil des Reglers.
  Falls für den Regler eine Struktur-Zerlegung nicht möglich ist oder die Regelzone versagt, kann zur Optimierung des Führungsverhaltens folgende Funktion aktiviert werden:
- **Struktur-Umschaltung:**
  Vorübergehendes Abschalten des I-Anteils bzw. prädiktive Stellwertvorgabe bei großen positiven Sollwertsprüngen. Diese Funktion ist bei aktivierter Regelzone unwirksam.

- **Speichern der Reglerparameter**
  Wenn Sie die aktuellen Reglerparameter als brauchbar einstufen, können Sie diese vor einer manuellen Änderung in eigens dafür vorgesehenen Parametern im Instanz-DB des Self-Tuners speichern. Bei einer Regleroptimierung werden die gespeicherten Parameter durch die vor der Optimierung gültigen Werte überschrieben.
- **Gespeicherte Reglerparameter zurück laden**
  Die zuletzt gespeicherten Reglerparameter können mit dieser Funktion wieder für den Regler aktiviert werden.
- **Ändern der Reglerparameter**
  Wenn Sie für den Regler selbst Reglerparameter einstellen wollen, können Sie diese über spezielle Parameter im Instanz-DB des Self-Tuners vorgeben und mit dieser Funktion dem Regler übergeben.

Aktivierung der Funktionen

Funktion	Startbit	Freigabebit	Startbedingung: PHASE	Startbedingung: Sollwertsprung an SP
Handbetrieb	—	MAN_ON	0 bis 7	—
Hand/Automatik-Umschaltung mit prädiktiver Stellwertvorgabe	—	PRED_ON	Übergang 7 -> 4	—
Ersteinstellung	ADAPT1ST	—	0, 4	pos.: > MIN_STEP
Ersteinstellung oder Nachoptimierung	ADAPT_ON	—	4	pos.: > MIN_STEP
Kühloptimierung	COOLID_ON	—	4	—
Regelzone	—	CONZ_ON	4	
Struktur-Umschaltung	—	STRUC_ON	4	pos.: > MIN_STEP, neg.: > MIN_STEP × RATIO_FAC
Reglerparameter speichern	SAVE_PAR	—	2 bis 7	—
Reglerparameter zurück laden	UNDO_PAR	—	4	—
Reglerparameter ändern	LOAD_PAR	—	4	—

Hinweis

Startbits werden nach erfolgreichem Start der Funktion zurückgesetzt, Freigabebits bleiben gesetzt.

Siehe auch

Beschreibung TUN_EC (Seite 5853)

Eingangsparameter TUN_EC (Seite 5862)

Ausgangsparameter TUN_EC (Seite 5864)

Durchgangsparameter TUN_EC (Seite 5866)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Regelzone und Struktur-Umschaltung

Regelzone

Der Self-Tuner arbeitet mit einer Regelzone. Das bedeutet, dass der Regler nach folgendem Algorithmus angesteuert wird:

- Ist die Regeldifferenz größer als CON_ZONE, wird mit maximaler Leistung geheizt.
- Ist die Regeldifferenz kleiner als (-CON_ZONE) × RATIOFAC, wird mit maximaler Leistung gekühlt.

Bei der Erst- oder Nachoptimierung wird eine Regelzone CON_ZONE vom Self-Tuner ermittelt und bei passendem Prozesstyp aktiviert: CONZ_ON = TRUE. Sie können im Regelbetrieb die Regelzone ändern oder ganz abschalten (mit CONZ_ON = FALSE).

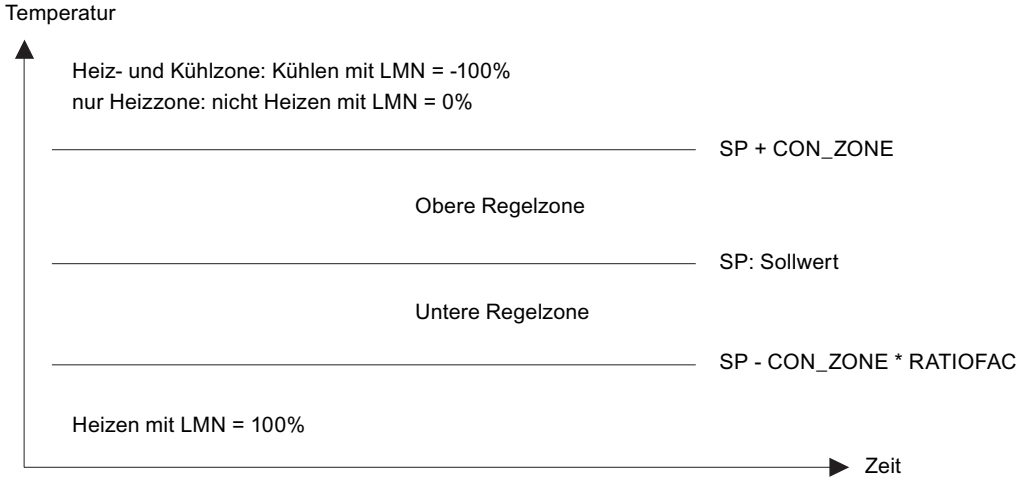

Bei eingeschalteter Regelzone (CONZ_ON = TRUE) ist die Struktur-Umschaltung STRUC_ON nicht wirksam.

Hinweis

Beim separaten Reglerentwurf für Strecken mit großer Verzugszeit wird keine Regelzone ermittelt, d. h. wenn nach der Adaption STATUS_D = 3 ist.

Stellen Sie vor dem händischen Einschalten der Regelzone sicher, dass die Regelzonenbreite nicht zu klein eingestellt ist. Bei zu klein eingestellter Regelzonenbreite treten Schwingungen im Verlauf der Stellgröße und des Istwertverlaufs auf.

Struktur-Zerlegung

Die Struktur-Zerlegung ist eine Funktion des Reglers. Bei Regelbetrieb mit Struktur-Zerlegung liegen der P- bzw. D-Anteil in der Rückführung, d. h., eine Sollwertänderung wirkt nur am I-Anteil des Reglers. Bei Sollwertsprüngen tritt kein Stellwertsprung auf, sondern ein rampenförmiger Verlauf des Stellwertes. Das Störverhalten ändert sich dadurch nicht.

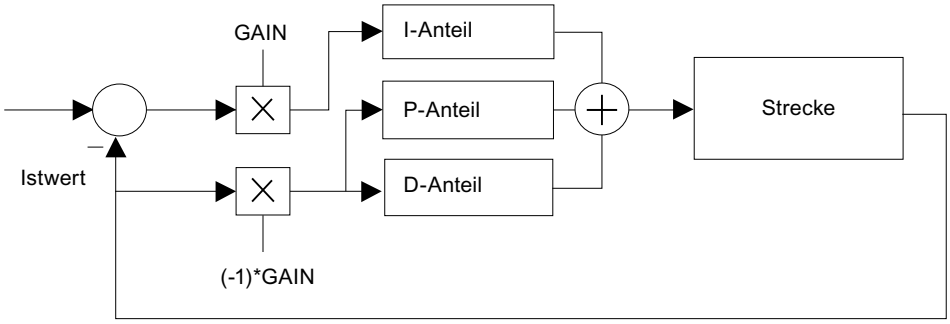

Bild 4-24 Regelbetrieb mit Struktur-Zerlegung

Bei Regelbetrieb ohne Struktur-Zerlegung liegen P- und D-Anteil genau wie der I-Anteil in der Regeldifferenz. Bei dieser Betriebsart können Sie eine Struktur-Umschaltung anwählen (STRUC_ON = TRUE).

Regelbetrieb mit Struktur-Zerlegung ist in den folgenden Regelungspaketen möglich:

- FM 355/455
- Standard PID Control
- Modular PID Control

Hinweis

Im Einfachregler PID Control von STEP 7-Basis und in der CPU 314 IFM ist nur Regelbetrieb ohne Struktur-Zerlegung möglich. Die Struktur-Umschaltung sollte nur dann gewählt werden, wenn keine anderen Alternativen zur Verfügung stehen.

Struktur-Umschaltung bei P- und D-Anteil in der Regeldifferenz

In der Voreinstellung ist die Struktur-Umschaltung abgeschaltet. Sie sollte auch nur dann gewählt werden, wenn keine anderen Alternativen wie z. B. Regelzone oder Struktur-Zerlegung zur Verfügung stehen.

Bei eingeschalteter Regelzone (CONZ_ON = TRUE) ist die Struktur-Umschaltung STRUC_ON nicht wirksam.

Die Struktur-Umschaltung können Sie mit STRUC_ON = TRUE einschalten. Der Baustein wählt abhängig von der Gesamtkreisverstärkung (Produkt aus Prozessverstärkung und Reglerverstärkung) automatisch zwischen zwei Steuerungsmaßnahmen aus:

- I-Anteil vorübergehend deaktiviert, P(D)-Regelung:
  Bei einem positiven Sollwertsprung ≥ MIN_STEP werden der I-Anteil des Reglers vorübergehend abgeschaltet und die Verstärkung etwas vergrößert, d. h. ein reiner P(D)-Regler benützt. In Sollwertnähe wird der I-Anteil wieder zugeschaltet und die Verstärkung wieder reduziert (PHASE = 5).
- Vorübergehende prädiktive Stellwertvorgabe:
  Strecken mit großer Verzugszeit sind nicht gut P(D)-regelbar. Daher wird nach einem positiven Sollwertsprung ≥ MIN_STEP die für den neuen Sollwert stationär erforderliche Stellgröße konstant ausgegeben. In Sollwertnähe wird stoßfrei wieder auf den PI- bzw. PID-Reglerbetrieb zurückgeschaltet. Dies ergibt ein langsames, aber überschwingfreies Verhalten (PHASE = 6).

Hinweis

Wenn Sie bei positiven Sollwertsprüngen (z. B. bei Aufheizvorgängen) mit der Struktur-Umschaltung keine guten Ergebnisse erzielen (träges Einschwingverhalten), können Sie unter der Voraussetzung, dass kleine Überschwinger auftreten dürfen, die Struktur-Umschaltung mit STRUC_ON = FALSE abschalten.

Betriebsarten der Reglerstruktur	CONZ_ON	PFDB_SEL, DFDB_SEL [1)]	STRUC_ON
Regelzone einschalten	TRUE	beliebig	beliebig
Struktur-Zerlegung einschalten	beliebig	TRUE	FALSE
Struktur-Umschaltung einschalten	FALSE	FALSE	TRUE

[1)] Reglerparameter PFDB_SEL, DFDB_SEL: Bei CONT_C sind diese Parameter nicht vorhanden. Dies entspricht der Einstellung PFDB_SEL = DFDB_SEL = FALSE (ohne Struktur-Zerlegung).

Hinweis

Hinweise zur Wahl der Betriebsarten:

- Der Self-Tuner berechnet eine Regelzone und aktiviert diese automatisch, falls das Verhältnis der beiden Zeitkonstanten des Prozesses f = TM_LAG1 / TM_LAG2 > 10 ist.
- Bei Sprüngen innerhalb der Regelzone wirkt die Struktur-Zerlegung, die defaultmäßig angewählt wird, falls der Regler es zulässt.
- Falls Sie kleine Sollwertsprünge innerhalb der Regelzone möglichst schnell ausregeln wollen und dabei kleine Überschwinger in Kauf nehmen, deaktivieren Sie die Struktur-Zerlegung.
  Zahlenbeispiel: Regelzone 20 °C; dann treten Überschwinger von ca. 10 % bis 20 % der Sprunghöhe (< 20 °C), d. h. von weniger als 2 bis 4 °C auf.
- Auch wenn der Prozess keine Regelzone erlaubt oder Sie diese deaktiviert haben, verhindert die Struktur-Zerlegung Überschwinger bei Sollwertsprüngen aller Größen. Die Anstiegszeiten werden aber etwas länger als mit Regelzone.
- Falls der Prozess keine Regelzone und der Regler keine Struktur-Zerlegung erlaubt, aktivieren Sie die Struktur-Umschaltung. Die Struktur-Umschaltung darf nicht gleichzeitig mit der Struktur-Zerlegung aktiviert werden.

Eingangsparameter TUN_EC

Tabelle 4-212

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP	0.0	REAL	0.0	Am Eingang SP geben Sie den Sollwert ein. Er wird über SP_OUT mit dem Regler verschaltet. Beachten Sie bitte, dass der Sollwert innerhalb der Sollwertgrenzen des Reglers liegt.
PV	4.0	REAL	0.0	PV wird mit dem Istwert des Reglers verschaltet.
LMN	8.0	REAL	0.0	LMN wird mit dem Stellwert des Reglers verschaltet. Zulässig sind Werte von -100 bis 100 %.
MAN	12.0	REAL	0.0	Am Eingang MAN geben Sie den Handwert ein. Wenn MAN_ON = TRUE, wird der Handwert MAN auf den Stellwert des Reglers übertragen. Bei MAN_ON = TRUE und MAN = 0.0 ist die Heizung /Kühlung aus. Bei MAN_ON = FALSE wird der Stellwert vom Regler vorgegeben. Zulässig sind Werte von -100 bis 100 %.
MIN_STEP	16.0	REAL	10.0	Am Eingang MIN_STEP können Sie die minimale Sollwertsprunghöhe vorgeben, ab der eine Ersteinstellung bzw. Adaption durchgeführt werden kann. Bei eingeschalteter Struktur-Umschaltung (STRUC_ON = TRUE) muss zur Aktivierung der Struktur-Umschaltung der Sollwertsprung größer als MIN_STEP sein. Empfehlung: MIN_STEP ≥ 10 % des benötigten Soll-Istwert-Bereichs

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LHLM_TUN	20.0	REAL	80.0	Am Eingang LHLM_TUN geben Sie den Stellwertsprung für die Ersteinstellung ein. Bei anschließenden Nachoptimierungen wird der Stellwertsprung automatisch berechnet. Zulässig sind Werte von 0 bis 100 %.
LLLM_TUN	24.0	REAL	-20.0	Am Eingang LLM_TUN geben Sie den Stellwertsprung für die Kühloptimierung ein. Zulässig sind Werte von -100 bis 0 %.
NORM_FAC	28.0	REAL	1.0	Bei Reglern mit normierter Reglerverstärkung (Standard PID Control) wird ein Normierungsfaktor aufgeschaltet. Anhand der Bereichsgrenzen des Istwerts des Reglers wird der Normierungsfaktor berechnet und mit dem Eingang NORM_FAC verschaltet.
MAN_ON	32.0	BOOL	TRUE	Wenn MAN_ON = TRUE, wird der Stellwert des Reglers über den Eingang MAN vorgegeben. Bei MAN_ON = FALSE wird der Stellwert vom Regler vorgegeben.
PID_ON	32.1	BOOL	TRUE	Am Eingang PID_ON können Sie festlegen, ob der optimierte Regler als PI- oder als PID-Regler arbeiten soll. PID-Regler: PID_ON = TRUE PI-Regler: PID_ON = FALSE Es kann jedoch sein, dass bei manchen Streckentypen trotz PID_ON = TRUE ein PI-Regler entworfen wird.
STRUC_ON	32.2	BOOL	FALSE	Wenn STRUC_ON = TRUE, dann sorgt bei Sollwertsprüngen eine Strukturumschaltung (von PI(D) auf P(D)-Regelung) dafür, dass Überschwinger weitestgehend vermieden werden. Der Eingang STRUC_ON darf nicht = TRUE gesetzt werden, wenn beim Regler P- und D-Anteil in die Rückführung geschaltet sind.
WRITE_DIS	32.3	BOOL	FALSE	In Verbindung mit der Reglerbaugruppe muss der Self-Tuner wissen, ob der Treiberbaustein bereit ist, Daten an die FM zu senden. Dies ist der Fall, wenn LOAD_OP = LOAD_PAR = FALSE. Das Ergebnis aus der ODER-Verknüpfung von LOAD_OP und LOAD_PAR wird mit WRITE_DIS verschaltet.
PRED_ON	32.4	BOOL	TRUE	Beim Umschalten in den Automatikbetrieb können Sie wählen, ob Sie stoßfrei vom letzten Handwert oder stoßbehaftet mit einem vom Self-Tuner berechneten Stellwert umschalten wollen. Ist PRED_ON = TRUE (Defaulteinstellung), wird stoßbehaftet umgeschaltet. Die stoßbehaftete Umschaltung ist erst nach einer Optimierung wirksam und bewirkt besonders bei PI-Regelung ein schnelleres Einschwingen in den gewünschten Sollwert.
CYCLE	34.0	TIME	100 ms	Am Eingang CYCLE muss die Abtastzeit parametriert werden. CYCLE muss mit der Abtastzeit des einzustellenden Reglers und mit der Zeitdifferenz zwischen zwei Aufrufen (Zykluszeit des Weckalarm-OB unter Beachtung der Untersetzungen) übereinstimmen. CYCLE > 1 ms

Siehe auch

Beschreibung TUN_EC (Seite 5853)

Arbeitsweise TUN_EC (Seite 5857)

Ausgangsparameter TUN_EC (Seite 5864)

Durchgangsparameter TUN_EC (Seite 5866)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Ausgangsparameter TUN_EC

Tabelle 4-213

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MAN_OUT	38.0	REAL	0.0	MAN_OUT wird mit dem Hand-Eingangswert des Reglers verbunden. Wenn MAN_ON = TRUE, dann gibt der Self-Tuner am Ausgang MAN_OUT den am Eingang MAN vorgegebenen Wert aus.
SP_OUT	42.0	REAL	0.0	SP_OUT wird mit dem Sollwerteingang des Reglers verbunden werden. Der Self-Tuner schreibt den am Eingang SP vorgegebenen Sollwert auf den Ausgang SP_OUT. Bei Handbetrieb MAN_ON = TRUE wird der Sollwert dem Istwert nachgeführt: SP_OUT = PV.
GAIN	46.0	REAL	1.5	GAIN wird mit dem Proportionalbeiwert (Reglerverstärkung) des Reglers verschaltet.
TI	50.0	TIME	1 h	TI wird mit der Integrationszeit (Nachstellzeit) des Reglers verschaltet.
TD	54.0	TIME	0 ms	TD wird mit der Differenzierzeit (Vorhaltezeit) des Reglers verschaltet.
TM_LAG	58.0	TIME	1 s	TM_LAG wird mit der Verzögerungszeit des Reglerdifferenzierers verschaltet.
RATIOFAC	62.0	REAL	1.0	RATIOFAC wird mit dem Verhältnisfaktor Heizen/Kühlen des Reglers oder der Anweisung PULSEGEN verschaltet. Der Parameter wird nur bei Kühlung gebraucht.
PHASE	66.0	INT	0	Parameter TUN_EC.PHASE (Seite 5867) zeigt den aktuellen Ablauf der Regleroptimierung. Bei der Optimierung von Standard PID Control wird der Ausgang PHASE mit dem Eingang PHASE der Anweisung PID_CP verschaltet.
STATUS_H	68.0	INT	0	Parameter TUN_EC.STATUS_H (Seite 5870) zeigt Optimierungsergebnisse über die Suche des Wendepunktes beim Heizvorgang an.
STATUS_D	70.0	INT	0	Parameter TUN_EC.STATUS_D (Seite 5871) zeigt Optimierungsergebnisse über den Reglerentwurf beim Heizvorgang an.
STATUS_C	72.0	INT	0	Parameter TUN_EC.STATUS_C (Seite 5871) zeigt Optimierungsergebnisse über die Suche des Wendepunktes beim Kühlvorgang an.
QMAN_ON	74.0	BOOL	FALSE	QMAN_ON wird mit der Hand- Automatikumschaltung des Reglers verschaltet.
QI_SEL	74.1	BOOL	TRUE	QI_SEL wird mit dem Ein- Ausschalter des Regler-I-Anteils I_SEL verschaltet. Bei der FM355/455 muss in Abhängigkeit von QI_SEL TI zu Null geschaltet werden: Wenn QI_SEL = FALSE, dann TI = 0.
QD_SEL	74.2	BOOL	FALSE	QD_SEL wird mit dem Ein- Ausschalter des Regler-D-Anteils D_SEL verschaltet. Bei der FM355/455 muss in Abhängigkeit von QD_SEL TD zu Null geschaltet werden: Wenn QD_SEL = FALSE, dann TD = 0.
QWRITE	74.3	BOOL	FALSE	QWRITE wird mit den Eingängen LOAD_PAR und LOAD_OP der Anweisung PID_FM der FM355/455 verschaltet. Dadurch werden die Datensätze des Reglerkanals in die FM geladen.
PROCESS	76.0	STRUCT		Die Struktur PROCESS enthält die Parameter des Prozesses.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
GAIN	+0.0	REAL	999.0	Am Ausgang PROCESS.GAIN wird die Prozessverstärkung angezeigt.
TM_LAG1	+4.0	REAL	0.0	Am Ausgang PROCESS.TM_LAG1 wird die langsame Prozesszeitkonstante angezeigt.
TM_LAG2	+8.0	REAL	0.0	Am Ausgang PROCESS.TM_LAG2 wird die schnelle Prozesszeitkonstante angezeigt.
PV00	+12.0	REAL	0.0	Am Ausgang PROCESS.PV00 wird der Gleichwert des Prozesses angezeigt. Der Gleichwert ist der stationäre Istwert bei Stellwert = Null. Dies entspricht der Umgebungstemperatur bei kalter Anlage.
KIG	+16.0	REAL	0.0	Am Ausgang PROCESS.KIG wird das maximale Anstiegsverhältnis des Istwerts beim Heizen angezeigt.
KIG_C	+20.0	REAL	0.0	Am Ausgang PROCESS.KIG_C wird das maximale Abstiegsverhältnis des Istwerts beim Kühlen angezeigt.
PI_CON	100.0	STRUCT		Die Struktur PI_CON enthält die Parameter des PI-Reglers.
GAIN	+0.0	REAL	1.5	PI_CON.GAIN zeigt den Proportionalbeiwert des PI-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TI	+4.0	REAL	3600.0	PI_CON.TI zeigt die Integrationszeit des PI-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
PID_CON	108.0	STRUCT		Die Struktur PID_CON enthält die Parameter des PID-Reglers.
GAIN	+0.0	REAL	1.5	PID_CON.GAIN zeigt den Proportionalbeiwert des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TI	+4.0	REAL	3600.0	PID_CON.TI zeigt die Integrationszeit des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TD	+8.0	REAL	0.0	PID_CON.TD zeigt die Differenzierzeit des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
PI_CON_OLD	120.0	STRUCT		Die Struktur PI_CON_OLD enthält die alten Parameter des PI-Reglers.
GAIN	+0.0	REAL	1.5	Am Ausgang PI_CON_OLD.GAIN wird der alte Proportionalbeiwert des PI-Reglers angezeigt.
TI	+4.0	REAL	3600.0	Am Ausgang PI_CON_OLD.TI wird die alte Integrationszeit des PI-Reglers angezeigt.
PID_CON_OLD	128.0	STRUCT		Die Struktur PID_CON_OLD enthält die alten Parameter des PID-Reglers.
GAIN	+0.0	REAL	0.0	Am Ausgang PID_CON_OLD.GAIN wird der alte Proportionalbeiwert des PID-Reglers angezeigt.
TI	+4.0	REAL	0.0	Am Ausgang PID_CON_OLD.TI wird die alte Integrationszeit des PID-Reglers angezeigt.
TD	+8.0	REAL	0.0	Am Ausgang PID_CON_OLD.TD wird die alte Differenzierzeit des PID-Reglers angezeigt.
TU	140.0	REAL	0.0	Am Ausgang TU wird die ermittelte Verzugszeit angezeigt.
TA	144.0	REAL	0.0	Am Ausgang TA wird die ermittelte Ausgleichszeit angezeigt.

Siehe auch

Beschreibung TUN_EC (Seite 5853)
Arbeitsweise TUN_EC (Seite 5857)
Eingangsparameter TUN_EC (Seite 5862)
Durchgangsparameter TUN_EC (Seite 5866)

Durchgangsparameter TUN_EC

Tabelle 4-214

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
ADAPT1ST	148.0	BOOL	FALSE	Wenn Sie die erste Optimierung an Ihrem Prozess durchführen oder Sie die Erkenntnisse aus vorherigen Optimierungen wie Stellwertsprunghöhe oder Filterzeitkonstanten nicht nutzen wollen, dann müssen Sie eine Ersteinstellung durchführen. Hierzu müssen Sie ADAPT1ST = TRUE setzen, dann macht der Self-Tuner einen Initialisierungslauf und stellt dabei die unter Initialisierungslauf beschriebenen Vorbelegungen ein. Der Self-Tuner setzt ADAPT1ST wieder auf FALSE zurück. Es ist nicht sinnvoll, ADAPT1ST bei einem Neustart (Warmstart) der CPU im OB100 auf TRUE zu setzen.
ADAPT_ON	148.1	BOOL	FALSE	Wenn Sie eine Nachoptimierung anstoßen wollen, setzen Sie ADAPT_ON = TRUE. Beim darauffolgenden Sollwertsprung größer als MIN_STEP wird mit der Identifikation begonnen (PHASE = 2). Der Self-Tuner setzt ADAPT_ON wieder auf FALSE.
STEADY	148.2	BOOL	FALSE	Wenn der Istwert einen stationären Zustand erreicht hat, können Sie mit STEADY die Optimierung vorzeitig beenden, obwohl der Self-Tuner sich noch in PHASE 3 befindet. Auch können Sie mit STEADY den Reglerentwurf im stationären Zustand erneut durchführen, nachdem der Self-Tuner in PHASE 4 gegangen ist.
COOLID_ON	148.3	BOOL	FALSE	Wenn Sie eine Kühloptimierung anstoßen wollen, setzen Sie COOLID_ON = TRUE. Der Stellwert wird auf LLLM_TUN gesetzt und COOLID_ON auf FALSE zurückgesetzt. Bei mehreren Heizzonen sollten zunächst alle Ersteinstellungen bzw. Adaptionen fertig sein, bevor eine Kühloptimierung gestartet wird.
UNDO_PAR	148.4	BOOL	FALSE	Die alten Reglerparameter werden auf die aktuellen Reglerparameter übertragen und abhängig von PID_ON sofort aktiv. PI_CON = PI_CON_OLD PID_CON = PID_CON_OLD Die Funktion zum Rückgängigmachen von Änderungen ist nur im Automatik-Betrieb (PHASE 4) möglich.
SAVE_PAR	148.5	BOOL	FALSE	Die aktuellen Reglerparameter werden auf die alten Reglerparameter übertragen. PI_CON_OLD = PI_CON PID_CON_OLD = PID_CON
LOAD_PAR	148.6	BOOL	FALSE	Die geänderten Parameter PI_CON und PID_CON werden an den Regler übertragen und somit aktiv.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CONZ_ON	148.7	BOOL	FALSE	Wenn CONZ_ON = TRUE, arbeitet der Regler mit Regelzone. Bei einer Regleroptimierung entscheidet der Self-Tuner, ob mit oder ohne Regelzone geregelt werden soll und ermittelt eine optimale Regelzone. Nach der Optimierung in PHASE 4 können Sie die Regelzone mit CONZ_ON = FALSE abschalten oder die Regelzone umparametrieren.
CON_ZONE	150.0	REAL	0.0	CON_ZONE gibt die Breite der Regelzone an. Bei einer Regleroptimierung entscheidet der Self-Tuner, ob mit oder ohne Regelzone geregelt werden soll und ermittelt eine optimale Regelzone. Nach der Optimierung in PHASE 4 können Sie die Regelzone mit CONZ_ON = FALSE abschalten oder die Regelzone umparametrieren.

Siehe auch

Beschreibung TUN_EC (Seite 5853)

Arbeitsweise TUN_EC (Seite 5857)

Eingangsparameter TUN_EC (Seite 5862)

Ausgangsparameter TUN_EC (Seite 5864)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Parameter TUN_EC.PHASE

PHASE	Betriebsart	Beschreibung
0	Automatikbetrieb: Regelbetrieb mit voreingestellten Parametern	Bei Erstellung eines Instanz-DB zum Self-Tuner ist der Parameter PHASE mit Null vorbelegt. Da MAN_ON mit TRUE vorbelegt ist, wird der Self-Tuner nach dem ersten Durchlauf nach PHASE 7 (Handbetrieb) wechseln. Sie können direkt vom Handbetrieb aus die Ersteinstellung starten (ADAPT_ON = TRUE) oder auf Automatikbetrieb umschalten. Beim erstmaligen Umschalten in den Automatikbetrieb MAN_ON = FALSE werden die Parameter von PI_CON übernommen. Die Vorbelegungswerte für GAIN und TI lauten: GAIN = 1.5, TI = 1h.
2	Regleroptimierung: Wendepunktsuche bei konstantem Stellwert	Sobald Sie einen Sollwertsprung ≥ MIN_STEP in positiver Richtung zum Arbeitspunkt des warmen Prozesszustandes vorgeben, wird bei Ersteinstellung MAN_OUT mit LHLM_TUN und bei Nachoptimierung MAN_OUT mit einem vom Self-Tuner berechneten Wert belegt. Zusätzlich wird QMAN_ON = TRUE gesetzt. Beide Werte werden an den PID-Regler übertragen. Der PID-Regler wird damit im Handbetrieb gesteuert. MIN_STEP sollte größer als 10 % des Arbeitsbereichs von Soll- und Istwert sein.

PHASE	Betriebsart	Beschreibung
3	Regleroptimierung: Regelung mit einem PI-Grobregler	Wenn der Wendepunkt der Sprungantwort erkannt wird (STATUS_H = 2) oder der Istwert 70 % der Sprunghöhe des Sollwertes erreicht hat (STATUS_H = 3), wird ein vorsichtig eingestellter PI-Regler entworfen. Die Reglerverstärkung GAIN wird auf den Bereich 0.2 bis 20.0 begrenzt. Der Regler arbeitet ab sofort als PI-Regler und versucht die Strecke in einen stationären Zustand einzuregeln. Der Self-Tuner legt die neu ermittelten PI-Reglerparameter in GAIN, TI und PI_CON ab. Die alten Werte des PI-Reglers werden in PI_CON_OLD gesichert. Dauert es sehr lange, bis der stationäre Zustand erreicht ist (kriechendes Einschwingverhalten bei Temperaturprozessen), so können Sie schon beim annähernd stationären Zustand durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten anstoßen. Auch zu einem späteren Zeitpunkt in Phase 3 oder 4 können Sie durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten neu anstoßen. Oft wird der Reglerentwurf dadurch noch etwas verbessert. Sollte ein Überschwinger auftreten oder wird kein Wendepunkt gefunden, so kann das bei der Ersteinstellung an einer zu groß gewählten Stellwertsprunghöhe LHLM_TUN liegen und muss nicht zwangsläufig zu einer schlechten Reglereinstellung führen. Bei der nächsten Ersteinstellung sollten Sie LHLM_TUN ca. 20 % kleiner wählen. Hat der Baustein den stationären Zustand erkannt oder ist die Zeit 30 × TI (TI: Nachstellzeit des unter PHASE = 3 eingestellten PI-Reglers) seit dem Sollwertsprung verstrichen, wird ein verbesserter Reglerentwurf vorgenommen und auf PHASE = 4 umgeschaltet. Der Self-Tuner legt die neu ermittelten PI/PID-Reglerparameter in GAIN, TI, TD und PI_CON bzw. PID_CON ab. Die alten Werte des PID-Reglers werden in PID_CON_OLD gesichert. Wenn möglich, wird ein PID-Regler und ein PI-Regler entworfen. Falls PID_ON = TRUE, wird der PID-, anderenfalls der PI-Regler aktiviert. In PHASE 4 kann jederzeit zwischen diesen beiden Reglern stoßfrei umgeschaltet werden. Die Reglerverstärkung GAIN wird so begrenzt, dass die Kreisverstärkung des offenen Kreises (Produkt aus Reglerverstärkung und Prozessverstärkung) unter 80.0 liegt.
4	Automatikbetrieb: normaler Regelbetrieb mit/ohne Regelzone	In dieser Phase regelt der Regler mit seinen optimierten Parametern. Aus diesem Zustand heraus können Sie eine Kühloptimierung, eine Nachoptimierung oder eine Ersteinstellung starten.
5	Automatikbetrieb: Struktur-Umschaltung bei Sollwertsprung: I-Anteil deaktiviert, P(D)-Regelung	Bei einem positiven Sollwertsprung ≥ MIN_STEP werden der I-Anteil des Reglers vorübergehend abgeschaltet und die Verstärkung etwas vergrößert, d. h. ein reiner P(D)-Regler benützt. In Sollwertnähe wird der I-Anteil wieder zugeschaltet und die Verstärkung wieder reduziert
6	Automatikbetrieb: Struktur-Umschaltung bei Sollwertsprung: prädiktive Stellwertvorgabe	Strecken mit großer Verzugszeit sind nicht gut P(D)-regelbar. Daher wird nach einem positiven Sollwertsprung ≥ MIN_STEP die für den neuen Sollwert stationär erforderliche Stellgröße konstant ausgegeben. In Sollwertnähe wird stoßfrei wieder auf den PI- bzw. PID-Reglerbetrieb zurückgeschaltet. Dies ergibt ein langsames, aber überschwingfreies Verhalten.
7	Handbetrieb	Wenn Sie den Eingang MAN_ON auf TRUE setzen, wird der Ausgang QMAN_ON auf TRUE und der Ausgang MAN_OUT auf MAN gesetzt. Damit wird der PID-Regler in Handbetrieb geschaltet. Der Handbetrieb hat Priorität vor allen anderen Betriebsarten. Eine Ersteinstellung, Nachoptimierung oder eine Kühloptimierung wird abgebrochen. Beim Abschalten des Handbetriebs (MAN_ON = FALSE) geht der Regler in den Automatikbetrieb (PHASE = 4 im Ablauf des Self-Tuners) und regelt mit den vorhandenen Reglerparametern weiter.

Phasen der Heizoptimierung bei Ersteinstellung

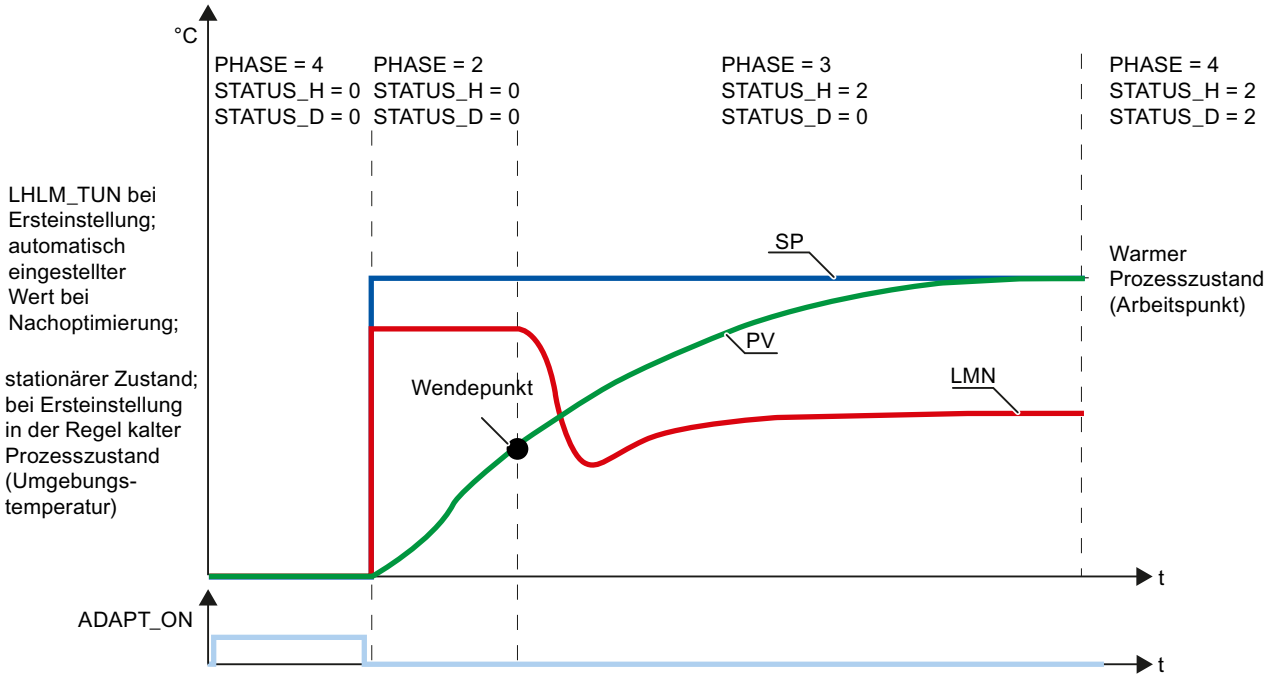

Phasen einer Kühloptimierung

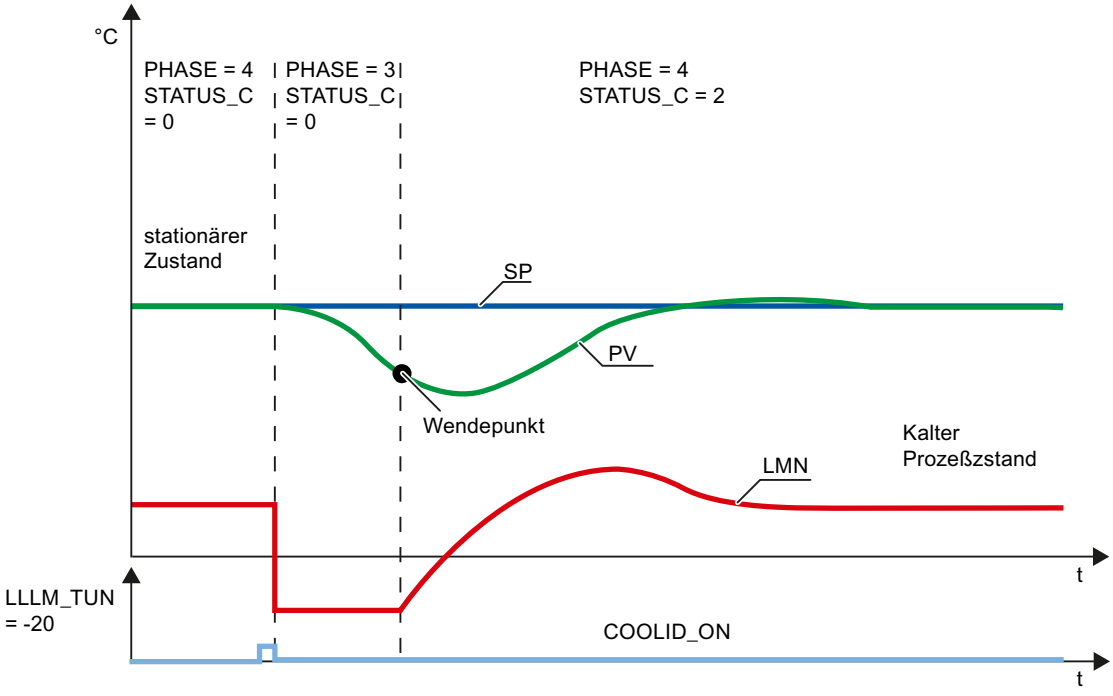

Siehe auch

Beschreibung TUN_EC (Seite 5853)
Arbeitsweise TUN_EC (Seite 5857)
Eingangsparameter TUN_EC (Seite 5862)
Durchgangsparameter TUN_EC (Seite 5866)
Parameter TUN_EC.STATUS_H (Seite 5870)
Parameter TUN_EC.STATUS_C (Seite 5871)
Parameter TUN_EC.STATUS_D (Seite 5871)

Parameter TUN_EC.STATUS_H

STATUS_H	Beschreibung	Abhilfe
0	Initialisierung	
1	Wendepunkt zu tief Die Optimierung führt in den meisten Fällen noch zu guten Reglerparametern	Um näher an den Idealfall zu kommen, vergrößern Sie LHLM_TUN oder verkleinern Sie die Sollwertsprunghöhe.
2	Idealfall: Wendepunkt gefunden	
3	Wendepunkt zu hoch Es wird ein Sicherheitsfaktor eingerechnet; die Optimierung führt in den meisten Fällen noch zu brauchbaren, aber etwas vorsichtigen Reglerparametern.	Um näher an den Idealfall zu kommen, verkleinern Sie LHLM_TUN oder vergrößern Sie die Sollwertsprunghöhe.
11, 12	Die Verzugszeit konnte nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann. Wenn der Prozess reines **VZ1-Verhalten** (keine Verzugszeit) hat, kann der Self-Tuner oftmals noch gute Reglerparameter entwerfen. Da mit einer großzügig geschätzten Verzugszeit weitergerechnet wird, kann es sein, dass STATUS_D nicht zu 1 (Regler für annäherndes VZ1-Verhalten) ermittelt und die Regelzone abgeschaltet wird, obwohl eine Regelzone bei VZ1-Prozessen durchaus sinnvoll ist.	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten. Sind Sie mit den Reglerparametern nicht zufrieden, können Sie zumindest die Werte aus PROCESS.GAIN und PROCESS.TM_LAG1 für ein VZ1-Modell und einen eigenen Reglerentwurf nach Lehrbuch verwenden.
13	Die Verzugszeit konnte nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann.	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Siehe auch

Beschreibung TUN_EC (Seite 5853)
Arbeitsweise TUN_EC (Seite 5857)
Eingangsparameter TUN_EC (Seite 5862)
Durchgangsparameter TUN_EC (Seite 5866)
Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_C (Seite 5871)

Parameter TUN_EC.STATUS_D (Seite 5871)

Parameter TUN_EC.STATUS_C

STATUS_C	Beschreibung	Abhilfe
0	Initialisierung	
1	Wendepunkt nicht gefunden Die Optimierung führt in den meisten Fällen noch zu guten Reglerparametern.	Um näher an den Idealfall zu kommen, vergrößern Sie LLLM_TUN (z. B. von -20.0 auf -10.0)
2	Idealfall: Kühlwendepunkt gefunden	
11, 12, 13	Fehlschätzung Verhältnisfaktor wird auf den Vorbelegungswert gesetzt: RATIOFAC = 1.0	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Siehe auch

Beschreibung TUN_EC (Seite 5853)

Arbeitsweise TUN_EC (Seite 5857)

Eingangsparameter TUN_EC (Seite 5862)

Durchgangsparameter TUN_EC (Seite 5866)

Parameter TUN_EC.PHASE (Seite 5867)

Parameter TUN_EC.STATUS_H (Seite 5870)

Parameter TUN_EC.STATUS_D (Seite 5871)

Parameter TUN_EC.STATUS_D

STATUS_D	Beschreibung	Abhilfe
0	Initialisierung	
1	Der Prozess hat fast PT1-Verhalten;	evtl. können Sie auf den D-Anteil des Reglers verzichten.
2	PI und PID-Regler möglich, VZ2-Verhalten	
3	Separater PID-Reglerentwurf für sehr große Verzugszeiten, Todzeiten oder hohe Prozessordnung In diesem Fall wird die Regelzone abgeschaltet.	
4	Fehlschätzung Bei Nachoptimierung kein Reglerentwurf. Bei Ersteinstellung wird mit dem groben PI-Regler aus PHASE 3 weitergearbeitet	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Siehe auch

Beschreibung TUN_EC (Seite 5853)
Arbeitsweise TUN_EC (Seite 5857)
Eingangsparameter TUN_EC (Seite 5862)
Durchgangsparameter TUN_EC (Seite 5866)
Parameter TUN_EC.PHASE (Seite 5867)
Parameter TUN_EC.STATUS_H (Seite 5870)
Parameter TUN_EC.STATUS_C (Seite 5871)

TUN_ES

Beschreibung TUN_ES

Die Anweisung TUN_ES dient zur Einstellung eines PID-Schrittreglers. Sie verschalten TUN_ES je nach Regler mit einer der folgenden Anweisungen:

- FM 355 S oder FM 455 S
  - PID_FM
- PID Basisfunktionen
  - CONT_S
  - TCONT_S
- Standard PID Control (Optionspaket PID Professional)
  - PID_ES
- Modular PID Control (Optionspaket PID Professional)
  - PID
  - LMNGEN_S

Aufruf

TUN_ES wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

Bei ADAPT1ST = TRUE wird die Initialisierungsroutine der Anweisung TUN_ES durchlaufen. Dabei werden folgende Vorbelegungen gemacht:

- Es wird auf PI-Regler geschaltet. Hierzu werden die Parameter von PI_CON an den Regler übertragen. Haben die Parameter PI_CON.GAIN und PI_CON.TI beide den Wert 0.0, dann werden Sie mit PI_CON.GAIN = 1.5 und PI_CON.TI = 3600.0 s vorbelegt.
- PROCESS.GAIN = 999.0
- STATUS_H = 0; STATUS_D = 0

Hinweis

Wenn nicht explizit gewünscht, rufen Sie bei Neustart (Warmstart) der CPU TUN_ES **nicht** wie bei anderen Anweisungen üblich in seiner Initialisierungsroutine auf, da sonst alle Prozessparameter verloren gehen.

Wenn Sie die Anweisung TUN_ES als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Inbetriebnahmeoberfläche zur Verfügung. Sie müssen TUN_EC über eine Beobachtungstabelle in Betrieb nehmen.

Abtastzeit CYCLE

Die Abtastzeit geben Sie am Parameter CYCLE vor. Die Abtastzeit sollte nicht größer als 10 % der ermittelten Integrationszeit des Reglers sein. Außerdem sollte die Abtastzeit nicht größer als 1 % der Motorstellzeit sein, damit sich eine Auflösung der Stellungsrückmeldung von mindestens 1 % ergibt. Sie können die Abtastzeit am Parameter CYCLE der Anweisung TUN_ES und des Reglers einstellen. Sie muss mit der Zeitdifferenz zwischen zwei Aufrufen (Zykluszeit des Weckalarm-OB unter Beachtung der Untersetzungen) übereinstimmen.

In Verbindung mit den Reglerbaugruppen FM355/455 muss die Abtastzeit von TUN_ES ungefähr gleich der Resultierenden Zykluszeit der Regler auf der FM sein (Parametereinstellung >Baugruppenparameter).

Verhalten im Fehlerfall

Das Fehlermeldewort RET_VAL wird vom Baustein nicht ausgewertet

Siehe auch

Arbeitsweise TUN_ES (Seite 5876)

Eingangsparameter TUN_ES (Seite 5880)

Ausgangsparameter TUN_ES (Seite 5881)

Durchgangsparameter TUN_ES (Seite 5884)

Parameter TUN_ES.PHASE (Seite 5885)

Parameter TUN_ES.STATUS_H (Seite 5888)

Parameter TUN_ES.STATUS_D (Seite 5889)

Anwendungsbereich TUN_ES

Den Self-Tuner können Sie besonders gut einsetzen bei:

- Temperaturregelungen, daneben aber auch bei
- Füllstandsregelungen und
- Durchflussregelungen

Bei Durchflussregelungen muss man unterscheiden, ob nur das Stellventil zu regeln ist oder ob sich nach dem Stellventil ein verzugsbehafteter Prozess anschließt. Bei reiner Stellventilregelung ist der Self-Tuner nicht anwendbar.

Anforderungen an den Prozess

Der Prozess muss folgenden Bedingungen genügen:

- **Stabiles, verzugsbehaftetes, asymptotisches Einschwingverhalten**
  Nach einem Sprung der Stellgröße muss der Istwert in einen stationären Zustand übergehen. Ausgeschlossen sind damit Prozesse, die schon ohne Regelung ein oszillatorisches Verhalten zeigen, sowie Regelstrecken ohne Ausgleich (Integrator in der Regelstrecke).
- **Nicht zu große Verzugszeiten**
  Der Einsatzbereich lässt sich anhand des Verhältnisses von Verzugszeit t_u und Ausgleichszeit t_a spezifizieren. Die Verzugszeit umfasst auch eine evtl. vorhandene Todzeit.
  Die Ersteinstellung bzw. Adaption ist bevorzugt ausgelegt für den Bereich $t_u < 1/10\ t_a$, in dem die meisten Temperaturstrecken liegen.
  Für den Bereich besonders großer Verzugszeiten $1/10\ t_a < t_u < 1/3\ t_a$ ist meistens noch der Entwurf eines brauchbaren PI-Reglers möglich.
- **Hinreichend lineares Verhalten über einen hinreichend großen Arbeitsbereich**
  Das heißt, dass sowohl für die Identifikation als auch für den normalen Regelungsbetrieb nichtlineare Effekte innerhalb dieses Arbeitsbereichs vernachlässigt werden dürfen. Allerdings ist es möglich, beim Wechsel des Arbeitspunktes den Prozess neu zu identifizieren, wenn der Adaptionsvorgang wieder in einer kleinen Umgebung des neuen Arbeitspunktes durchgeführt wird und während der Adaption die Nichtlinearität nicht durchfahren wird.
  Wenn bestimmte statische Nichtlinearitäten (z. B. Ventilkennlinien) bekannt sind, ist es in jedem Fall sinnvoll, sie vorab mit einem Polygonzug zu kompensieren, um das Prozessverhalten zu linearisieren.
- **Geringe Störeinflüsse bei Temperaturprozessen**
  Störeinflüsse wie die Wärmeübertragung an benachbarten Zonen oder die Wärmezunahme bzw. -abnahme durch Aggregatzustandswechsel dürfen den Gesamttemperaturprozess nicht zu stark beeinflussen. Gegebenenfalls ist eine Adaption am Arbeitspunkt notwendig. Z. B. müssen bei der Optimierung von Zonen eines Extruders alle Zonen gleichzeitig aufgeheizt werden.

- **Qualität der Messsignale**
  Die Messsignale müssen in ausreichender Qualität vorliegen, d. h. das Verhältnis von Störsignal zu Nutzsignal muss hinreichend klein sein.
- **Nicht zu große Prozessverstärkungen**
  Die Prozessverstärkungen dürfen nicht zu groß sein. Eine Normierung der Istwerte wird nicht verlangt. Die Prozessverstärkung K ist also ggf. mit physikalischen Einheiten behaftet, z. B.:
  K =ΔPV/ΔLMN, [K]= °C/ %
  Der endgültige Reglerentwurf basiert auf einer Berechnung der Prozessverstärkung K und kann damit prinzipiell beliebige Werte von K kompensieren. In der Lernphase ist K jedoch zunächst unbekannt, und bei extremen Kombinationen von Verstärkung und Lernsprunghöhe können Überschwinger nicht vermieden werden. Durch Reduzierung des Parameters LHLM_TUN im Heizvorgang oder Erhöhung von LLLM_TUN im Kühlvorgang wird auch der Überschwinger reduziert.

Anforderungen an Prozesse mit integral wirkenden Stellventilen

Bei Prozessen mit integral wirkenden Stellventilen gibt es neben den oben genannten noch weitere Anforderungen:

Die Motorlaufzeit des Stellventils muss kleiner sein als die Zeit, die benötigt wird, um nach einem Stellwertsprung einen Wendepunkt zu finden (vgl. Bild Prozessantwort). Ist dies nicht der Fall, so handelt es sich oftmals um eine Durchflussregelung, bei der nur das Stellventil als dominierender Prozessanteil wirksam ist. Der Einsatz des Self-Tuners ist dann nicht ratsam. Sie können den PI-Schrittregler dann nach folgender Faustformel von Hand einstellen:

GAIN = 1, TI = Motorstellzeit

Siehe auch

Arbeitsweise TUN_ES (Seite 5876)

Eingangsparameter TUN_ES (Seite 5880)

Ausgangsparameter TUN_ES (Seite 5881)

Durchgangsparameter TUN_ES (Seite 5884)

Parameter TUN_ES.PHASE (Seite 5885)

Parameter TUN_ES.STATUS_H (Seite 5888)

Parameter TUN_ES.STATUS_D (Seite 5889)

Arbeitsweise TUN_ES

Zur Steuerung, bzw. Optimierung des Reglers verfügt TUN_ES über folgende Funktionen:

- **Handbetrieb**
  Der Self-Tuner zwingt den Regler in den Handbetrieb und übergibt ihm seine Handparameter. Bei Schrittreglern mit Stellungsrückmeldung können Sie den Handwert im Gleitpunktformat und Handwertsignale wie auf/zu vorgeben. Bei Schrittreglern ohne Stellungsrückmeldung können Sie nur die Handwertsignale vorgeben.
- **Regleroptimierung**
  Bei der Optimierung der Reglerparameter wird zuerst die Stellgliedlaufzeit gemessen und anschließend die Strecke identifiziert. Aufgrund der Streckencharakteristik wird ein Reglerentwurf vorgenommen; darauf basierend, werden die neuen Reglerparameter berechnet und für den Regler aktiviert:
  - **Ersteinstellung**
    Die Ersteinstellung kann nur bei einem positiven Sollwertsprung erfolgen. Üblicherweise wird sie bei der ersten Inbetriebnahme oder bei einer gravierenden Veränderung des Streckenverhaltens eingesetzt.
  - **Nachoptimierung**
    Beim Wechsel des Arbeitspunkts oder bei Änderungen des Prozessverhaltens kann der Regler online während eines positiven Sollwertsprungs nachoptimiert werden. Die Nachoptimierung ist nur bei Schrittreglern mit Stellungsrückmeldung möglich.
- **Optimierung des Führungsverhaltens**
  Der Reglerentwurf ist grundsätzlich auf optimales Störverhalten ausgelegt. Die daraus resultierenden "scharfen" Parameter würden zu Überschwingern von 10 bis 20 % der Sprunghöhe bei Sollwertsprüngen führen.
  Eine erste Abhilfe stellt die **Struktur-Zerlegung** dar. Dabei werden der P- bzw. D-Anteil in die Rückführung gelegt. Somit wirkt eine Sollwertänderung nur am I-Anteil des Reglers. Die Strukturzerlegung darf nur bei Schrittreglern mit Stellungsrückmeldung angewendet werden.
  Falls für den Regler eine Struktur-Zerlegung nicht möglich ist, kann zur Optimierung des Führungsverhaltens folgende Funktion aktiviert werden:
- **Struktur-Umschaltung**
  Umschalten quasi auf P(D)-Regelbetrieb bzw. prädiktive Stellwertvorgabe bei großen positiven Sollwertsprüngen. Bei Schrittregler ohne Stellungsrückmeldung ist nur das Umschalten quasi auf P(D)-Regelbetrieb möglich.
- **Speichern der Reglerparameter**
  Wenn Sie die aktuellen Reglerparameter als brauchbar einstufen, können Sie diese vor einer manuellen Änderung der Reglerparameter in eigens dafür vorgesehenen Parametern im Instanzdatenbaustein des Self-Tuners speichern. Bei einer Regleroptimierung werden die gespeicherten Parameter durch die vor der Optimierung gültigen Werte überschrieben.
- **Gespeicherte Reglerparameter zurück laden**
  Die zuletzt gespeicherten Reglerparameter können mit dieser Funktion wieder für den Regler aktiviert werden.
- **Ändern der Reglerparameter**
  Wenn Sie für den Regler selbst Reglerparameter einstellen wollen, können Sie diese über spezielle Parameter im Instanz-DB des Self-Tuners vorgeben und mit dieser Funktion dem Regler übergeben.

Mit TUN_ES kann keine Kühloptimierung und kein Regelzonenbetrieb durchgeführt werden.

Aktivierung der Funktionen

Funktion	Startbit	Freigabebit	Startbedingung: PHASE	Startbedingung: Sollwertsprung an SP
Handbetrieb	—	MAN_ON LMNS_ON	0 bis 7	—
Ersteinstellung mit Messung der Motorlaufzeit	ADAPT1ST	—	0, 4	pos.: > MIN_STEP
Ersteinstellung ohne Messung der Stellgliedlaufzeit oder Nachoptimierung (nur möglich beim Schrittregler ohne Stellungsrückmeldung)	ADAPT_ON	—	4	pos.: > MIN_STEP
Struktur-Umschaltung	—	STRUC_ON	4	pos.: > MIN_STEP neg.: > MIN_STEP
Reglerparameter speichern	SAVE_PAR	—	2 bis 7	—
Reglerparameter zurückladen	UNDO_PAR	—	4	—
Reglerparameter ändern	LOAD_PAR	—	4	—

Hinweis

Startbits werden nach erfolgreichem Start der Funktion zurückgesetzt, Freigabebits bleiben gesetzt.

Siehe auch

Beschreibung TUN_ES (Seite 5872)

Eingangsparameter TUN_ES (Seite 5880)

Ausgangsparameter TUN_ES (Seite 5881)

Durchgangsparameter TUN_ES (Seite 5884)

Parameter TUN_ES.PHASE (Seite 5885)

Parameter TUN_ES.STATUS_H (Seite 5888)

Parameter TUN_ES.STATUS_D (Seite 5889)

Struktur-Umschaltung

Struktur-Zerlegung

Die Struktur-Zerlegung ist eine Funktion des Reglers. Bei Regelbetrieb mit Struktur-Zerlegung liegen der P- bzw. D-Anteil in der Rückführung, d. h. eine Sollwertänderung wirkt nur am I-Anteil des Reglers. Bei Sollwertsprüngen tritt kein Stellwertsprung auf, sondern ein rampenförmiger Verlauf des Stellwertes.

Das Störverhalten ändert sich dadurch nicht.

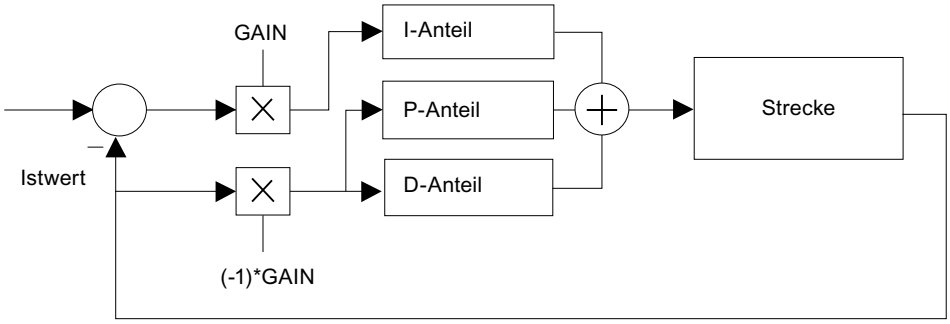

Bild 4-25 Regelbetrieb mit Struktur-Zerlegung

Bei Regelbetrieb ohne Struktur-Zerlegung liegen P- und D-Anteil genau wie der I-Anteil in der Regeldifferenz. Bei dieser Betriebsart können Sie eine Struktur-Umschaltung anwählen (STRUC_ON = TRUE).

Regelbetrieb mit Struktur-Zerlegung ist in den folgenden Regelungspaketen möglich:

- FM355/455
- Standard PID Control
- Modular PID Control

Hinweis

- Wenn keine Stellungsrückmeldung des Stellventils zur Verfügung steht, darf die Struktur-Zerlegung nicht angewendet werden. Das heißt, Sie müssen den P-Anteil und den D-Anteil in der Regeldifferenz belassen.
- Bei Regelbetrieb mit Struktur-Zerlegung ist die Totzone in der Regeldifferenz nahezu wirkungslos.
- Mit CONT_S und TCONT_S ist nur Regelbetrieb ohne Struktur-Zerlegung möglich.

Struktur-Umschaltung bei P- und D-Anteil in der Regeldifferenz

In der Voreinstellung ist die Struktur-Umschaltung abgeschaltet. Sie sollte auch nur dann gewählt werden, wenn keine Struktur-Zerlegung zur Verfügung steht. Die Struktur-Umschaltung können Sie mit STRUC_ON = TRUE einschalten. Abhängig von der Gesamtkreisverstärkung (Produkt aus Prozessverstärkung und Reglerverstärkung) wählt der Baustein automatisch zwischen zwei Steuerungsmaßnahmen aus:

- I-Anteil vorübergehend vergrößern, quasi P(D)-Regelung:
  Bei einem positiven Sollwertsprung ≥ MIN_STEP werden der I-Anteil des Reglers vorübergehend stark vergrößert und die Verstärkung etwas verkleinert, d. h. quasi ein P(D)-Regler benutzt. In Sollwertnähe wird wieder auf die ursprünglichen Werte des I-Anteils und der Verstärkung zurückgeschaltet (PHASE = 5).
- vorübergehende prädiktive Stellwertvorgabe:
  Strecken mit großen Verzugszeiten sind nicht gut P(D)-regelbar. Daher wird nach einem positiven Sollwertsprung ≥ MIN_STEP die für den neuen Sollwert stationär erforderliche Stellgröße konstant ausgegeben. In Sollwertnähe wird stoßfrei wieder auf den PI- bzw. PID-Reglerbetrieb zurückgeschaltet. Dies ergibt ein langsames, aber überschwingfreies Verhalten (PHASE = 6).

Hinweis

Bei Schrittregelung ohne Stellungsrückmeldung ist nur die erste Steuerungsmaßnahme (PHASE = 5) möglich.

Wenn Sie bei positiven Sollwertsprüngen (z. B. bei Aufheizvorgängen) mit der Struktur-Umschaltung keine guten Ergebnisse erzielen (träges Einschwingverhalten), können Sie unter der Voraussetzung, dass kleine Überschwinger auftreten dürfen, die Struktur-Umschaltung mit STRUC_ON = FALSE abschalten.

Betriebsarten der Reglerstruktur	PFDB_SEL, DFDB_SEL [1]	STRUC_ON
Struktur-Zerlegung einschalten	TRUE	FALSE
Struktur-Umschaltung einschalten	FALSE	TRUE

[1] Reglerparameter PFDB_SEL, DFDB_SEL: Bei CONT_S sind diese Parameter nicht vorhanden. Dies entspricht der Einstellung PFDB_SEL = DFDB_SEL = FALSE (ohne Struktur-Zerlegung).

Hinweis

- Bei Sollwertsprüngen verhindert die Struktur-Zerlegung ein Überschwingen der Regelgröße. Wählen Sie, wenn möglich, die Struktur-Zerlegung.
- Falls der Regler keine Struktur-Zerlegung erlaubt, aktivieren Sie die Struktur-Umschaltung. Die Struktur-Umschaltung darf nicht gleichzeitig mit der Struktur-Zerlegung aktiviert werden. Wenn Sie bei positiven Sollwertwertsprüngen ein schnelleres Einregeln wünschen und kleine Überschwinger zulässig sind, können Sie die Struktur-Umschaltung abschalten.

Eingangsparameter TUN_ES

Tabelle 4-215

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP	0.0	REAL	0.0	Am Eingang SP geben Sie den Sollwert ein. Er wird über SP_OUT mit dem Regler verschaltet. Beachten Sie bitte, dass der Sollwert innerhalb der Sollwertgrenzen des Reglers liegt.
PV	4.0	REAL	0.0	PV wird mit dem Istwert des Reglers verschaltet.
MAN	8.0	REAL	0.0	Am Eingang MAN geben Sie den Handwert ein. Wenn MAN_ON = TRUE, wird der Handwert MAN auf den Stellwert des Reglers übertragen. Bei MAN_ON = TRUE und MAN = 0.0 ist die Heizung aus. Bei MAN_ON = FALSE wird der Stellwert vom Regler vorgegeben. Zulässig sind Werte von -100 bis 100 %.
LMNR	12.0	REAL	0.0	LMNR wird mit der Stellungsrückmeldung des Reglers (sofern vorhanden) verschaltet. Zulässig sind Werte von 0.0 bis 100 %.
MIN_STEP	16.0	REAL	10.0	Am Eingang MIN_STEP können Sie die minimale Sollwertsprunghöhe vorgeben, ab der eine Ersteinstellung bzw. Adaption durchgeführt werden kann. Bei eingeschalteter Struktur-Umschaltung (STRUC_ON=TRUE) muss zur Aktivierung der Struktur-Umschaltung der Sollwertsprung größer als MIN_STEP sein. Empfehlung: MIN_STEP ≥ 10 % des benötigten Soll/Istwert-Bereichs
LHLM_TUN	20.0	REAL	80.0	Am Eingang LHLM_TUN geben Sie den Stellwertsprung für die Ersteinstellung ein. Bei anschließenden Nachoptimierungen wird der Stellwertsprung automatisch berechnet. Zulässig sind Werte von 0.0 bis 100 %.
NORM_FAC	24.0	REAL	1.0	Bei Reglern mit normierter Reglerverstärkung (Standard PID Control) wird ein Normierungsfaktor aufgeschaltet. Anhand der Bereichsgrenzen des Istwerts des Reglers wird der Normierungsfaktor berechnet und mit dem Eingang NORM_FAC verschaltet.
PULSE_TM	28.0	TIME	0 ms	PULSE_TM wird mit der Mindestimpulsdauer des Reglers verschaltet. PULSE_TM ≥ 1 ms
C_LMNUP	32.0	BOOL	FALSE	C_LMNUP wird mit dem Stellwertsignal QLMNUP des Reglers verschaltet.
C_LMNDN	32.1	BOOL	FALSE	C_LMNDN wird mit dem Stellwertsignal QLMNDN des Reglers verschaltet.
MAN_ON	32.2	BOOL	FALSE	Wenn MAN_ON = TRUE, wird der Stellwert des Reglers über den Eingang MAN vorgegeben. Bei MAN_ON = FALSE wird der Stellwert vom Regler vorgegeben.
LMNR_HS	32.3	BOOL	FALSE	LMNR_HS wird mit dem Anschlagsignal LMNR_HR des Reglers verschaltet.
LMNR_ON	32.4	BOOL	FALSE	Wenn LMNR_ON = TRUE, können Sie einen Schrittregler mit Stellungsrückmeldung mit LMNS_ON = TRUE oder mit MAN_ON = TRUE auf Handbetrieb umschalten. Wenn LMNR_ON = FALSE, können Sie einen Schrittregler ohne Stellungsrückmeldung mit LMNS_ON = TRUE auf Handbetrieb umschalten.
LMNS_ON	32.5	BOOL	TRUE	Wenn LMNS_ON gesetzt ist, können Sie an den Parametern LMNUP und LMNDN die Stellsignale des Reglers bedienen.
LMNUP	32.6	BOOL	FALSE	Über LMNUP können Sie das Stellsignal AUF des Reglers bedienen.
LMNDN	32.7	BOOL	FALSE	Über LMNDN können Sie das Stellsignal ZU des Reglers bedienen.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PID_ON	33.0	BOOL	FALSE	Am Eingang PID_ON können Sie festlegen, ob der optimierte Regler als PI- oder als PID-Regler arbeiten soll: PID-Regler: PID_ON = TRUE PI-Regler: PID_ON = FALSE Es kann jedoch sein, dass bei manchen Streckentypen trotz PID_ON = TRUE ein PI-Regler entworfen wird.
STRUC_ON	33.1	BOOL	FALSE	Wenn STRUC_ON = TRUE, dann sorgt bei Sollwertsprüngen eine Strukturumschaltung (von PI(D) auf P(D)-Regelung) dafür, dass Überschwinger weitestgehend vermieden werden. Der Eingang STRUC_ON darf nicht = TRUE gesetzt werden, wenn beim Regler P- und D-Anteil in die Rückführung geschaltet sind.
WRITE_DIS	33.2	BOOL	FALSE	In Verbindung mit der Reglerbaugruppe muss der Self-Tuner wissen, ob der Treiberbaustein bereit ist, Daten an die FM zu senden. Dies ist der Fall, wenn LOAD_OP = LOAD_PAR = FALSE. Das Ergebnis aus der ODER-Verknüpfung von LOAD_OP und LOAD_PAR wird mit WRITE_DIS verschaltet.
CYCLE	34.0	TIME	100 ms	Am Eingang CYCLE muss die Abtastzeit parametriert werden. CYCLE muss mit der Abtastzeit des einzustellenden Reglers und mit der Zeitdifferenz zwischen zwei Aufrufen (Zykluszeit des Weckalarm-OB unter Beachtung der Untersetzungen) übereinstimmen. ≥ 1 ms

Siehe auch

Beschreibung TUN_ES (Seite 5872)
Arbeitsweise TUN_ES (Seite 5876)
Ausgangsparameter TUN_ES (Seite 5881)
Durchgangsparameter TUN_ES (Seite 5884)
Parameter TUN_ES.PHASE (Seite 5885)
Parameter TUN_ES.STATUS_H (Seite 5888)
Parameter TUN_ES.STATUS_D (Seite 5889)

Ausgangsparameter TUN_ES

Tabelle 4-216

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MAN_OUT	38.0	REAL	0.0	MAN_OUT wird mit dem Hand-Eingangswert des Reglers verbunden. Wenn MAN_ON = TRUE, dann gibt der Self-Tuner am Ausgang MAN_OUT den am Eingang MAN vorgegebenen Wert aus.
SP_OUT	42.0	REAL	0.0	SP_OUT wird mit dem Sollwerteingang des Reglers verbunden. Der Self-Tuner schreibt den am Eingang SP vorgegebenen Sollwert auf den Ausgang SP_OUT. Bei Handbetrieb MAN_ON = TRUE wird der Sollwert dem Istwert nachgeführt: SP_OUT = PV.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
GAIN	46.0	REAL	1.5	GAIN wird mit dem Proportionalbeiwert (Reglerverstärkung) des Reglers verschaltet.
TI	50.0	TIME	1 h	TI wird mit der Integrationszeit (Nachstellzeit) des Reglers verschaltet.
TD	54.0	TIME	0 ms	TD wird mit der Differenzierzeit (Vorhaltezeit) des Reglers verschaltet.
TM_LAG	58.0	TIME	1 s	TM_LAG wird mit der Verzögerungszeit des Reglerdifferentiators verschaltet.
MTR_TM	62.0	TIME	30 s	MTR_TM wird mit der Motorlaufzeit des Reglers verschaltet.
DEADB_W	66.0	REAL	0.0	DEADB_W wird mit der Todtzonenbreite des Reglers verschaltet.
PHASE	70.0	INT	0	Parameter TUN_ES.PHASE (Seite 5885) zeigt den aktuellen Ablauf der Regleroptimierung. Bei der Optimierung von Standard PID Control wird der Ausgang PHASE mit dem Eingang PHASE der Anweisung PID_ES verschaltet.
STATUS_H	72.0	INT	0	Parameter TUN_ES.STATUS_H (Seite 5888) zeigt Optimierungsergebnisse über die Suche des Wendepunktes beim Heizvorgang an.
STATUS_D	74.0	INT	0	Parameter TUN_ES.STATUS_D (Seite 5889) zeigt Optimierungsergebnisse über den Reglerentwurf beim Heizvorgang an.
QMAN_ON	76.0	BOOL	FALSE	QMAN_ON wird mit der Hand-Automatikumschaltung des Reglers verschaltet.
QLMNS_ON	76.1	BOOL		QLMNS_ON wird mit dem Parameter "Handstellsignale einschalten" des Reglers verschaltet
QLMNUP	76.2	BOOL		QLMNUP wird mit dem Parameter "Handstellsignal auf" des Reglers verschaltet
QLMNDN	76.3	BOOL		QLMNDN wird mit dem Parameter "Handstellsignal zu" des Reglers verschaltet
QI_SEL	76.4	BOOL	TRUE	QI_SEL wird mit dem Ein- Ausschalter des Regler-I-Anteils I_SEL verschaltet. Bei der FM355/455 muss TI in Abhängigkeit von QI_SEL zu Null geschaltet werden: Wenn QI_SEL = FALSE, dann TI = 0.
QD_SEL	76.5	BOOL	FALSE	QD_SEL wird mit dem Ein- Ausschalter des Regler-D-Anteils D_SEL verschaltet. Bei der FM355/455 muss TD in Abhängigkeit von QD_SEL zu Null geschaltet werden: Wenn QD_SEL = FALSE, dann TD = 0.
QWRITE	76.6	BOOL	FALSE	QWRITE wird mit den Eingängen LOAD_PAR und LOAD_OP der Anweisung PID_FM der FM355/455 verschaltet. Dadurch werden die Datensätze des Reglerkanals in die FM geladen.
PROCESS	78.0	STRUCT		Die Struktur PROCESS enthält die Parameter des Prozesses.
GAIN	+0.0	REAL	999.0	Am Ausgang PROCESS.GAIN wird die Prozessverstärkung angezeigt.
TM_LAG1	+4.0	REAL	0.0	Am Ausgang PROCESS.TM_LAG1 wird die langsame Prozesszeitkonstante angezeigt.
TM_LAG2	+8.0	REAL	0.0	Am Ausgang PROCESS.TM_LAG2 wird die schnelle Prozesszeitkonstante angezeigt.
PV00	+12.0	REAL	0.0	Am Ausgang PROCESS. PV00 wird der Gleichwert des Prozesses angezeigt. Der Gleichwert ist der stationäre Istwert bei Stellwert = Null. Dies entspricht der Umgebungstemperatur bei kalter Anlage.
KIG	+16.0	REAL	0.0	Am Ausgang PROCESS.KIG wird das maximale Anstiegsverhältnis des Istwerts beim Heizen angezeigt.
MTR_TM	+20.0	REAL	0.0	Am Ausgang MTR_TM wird die Motorlaufzeit angezeigt.
PI_CON	102.0	STRUCT		Die Struktur PI_CON enthält die Parameter des PI-Reglers.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
GAIN	+0.0	REAL	1.5	PI_CON.GAIN zeigt den Proportionalbeiwert des PI-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TI	+4.0	REAL	3600.0	PI_CON.TI zeigt die Integrationszeit des PI-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
PID_CON	110.0	STRUCT		Die Struktur PID_CON enthält die Parameter des PID-Reglers.
GAIN	+0.0	REAL	0.0	PID_CON.GAIN zeigt den Proportionalbeiwert des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TI	+4.0	REAL	0.0	PID_CON.TI zeigt die Integrationszeit des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
TD	+8.0	REAL	0.0	PID_CON.TD zeigt die Differenzierzeit des PID-Reglers. Sie können den Wert manuell nachbessern und durch Setzen von LOAD_PAR wirksam werden lassen.
PI_CON_OLD	122.0	STRUCT		Die Struktur PI_CON_OLD enthält die alten Parameter des PI-Reglers.
GAIN	+0.0	REAL	1.5	Am Ausgang PI_CON_OLD.GAIN wird der alte Proportionalbeiwert des PI-Reglers angezeigt.
TI	+4.0	REAL	3600.0	Am Ausgang PI_CON_OLD.TI wird die alte Integrationszeit des PI-Reglers angezeigt.
PID_CON_OLD	130.0	STRUCT		Die Struktur PID_CON_OLD enthält die alten Parameter des PID-Reglers.
GAIN	+0.0	REAL	1.5	Am Ausgang PID_CON_OLD.GAIN wird der alte Proportionalbeiwert des PID-Reglers angezeigt.
TI	+4.0	REAL	3600.0	Am Ausgang PID_CON_OLD.TI wird die alte Integrationszeit des PID-Reglers angezeigt.
TD	+8.0	REAL	0.0	Am Ausgang PID_CON_OLD.TD wird die alte Differenzierzeit des PID-Reglers angezeigt.
TU	142.0	REAL	0.0	Am Ausgang TU wird die ermittelte Verzugszeit angezeigt.
TA	146.0	REAL	0.0	Am Ausgang TA wird die ermittelte Ausgleichszeit angezeigt.

Siehe auch

Beschreibung TUN_ES (Seite 5872)

Arbeitsweise TUN_ES (Seite 5876)

Eingangsparameter TUN_ES (Seite 5880)

Durchgangsparameter TUN_ES (Seite 5884)

Durchgangsparameter TUN_ES

Tabelle 4-217

Parameter	Ares-se	Daten-typ	Vorbe-legung	Beschreibung
ADAPT1ST	150.0	BOOL	FAL-SE	Wenn Sie die erste Optimierung an Ihrem Prozess durchführen oder Sie die Erkenntnisse aus vorherigen Optimierungen wie Stellwertsprunghöhe oder Filterzeitkonstanten nicht nutzen wollen, dann müssen Sie eine Ersteinstellung durchführen. Hierzu müssen Sie ADAPT1ST = TRUE setzen; dann macht der Self-Tuner einen Initialisierungslauf und stellt dabei die unter Initialisierungslauf beschriebenen Vorbelegungen ein. Der Self-Tuner setzt ADAPT1ST wieder auf FALSE zurück. Es ist nicht sinnvoll, ADAPT1ST bei einem Neustart (Warmstart) der CPU im OB100 auf TRUE zu setzen.
ADAPT_ON	150.1	BOOL	FAL-SE	Wenn Sie eine Nachoptimierung anstoßen wollen, setzen Sie ADAPT_ON = TRUE. Beim darauffolgenden Sollwertsprung größer als MIN_STEP wird mit der Identifikation begonnen (PHASE = 2). Der Self-Tuner setzt ADAPT_ON wieder auf FALSE.
STEADY	150.2	BOOL	FAL-SE	Wenn der Istwert einen stationären Zustand erreicht hat, können Sie mit STEADY die Optimierung vorzeitig beenden, obwohl der Self-Tuner sich noch in PHASE 3 befindet. Auch können Sie mit STEADY, nachdem der Self-Tuner in PHASE 4 gegangen ist, den Reglerentwurf im stationären Zustand erneut durchführen.
UNDO_PAR	150.3	BOOL	FAL-SE	Die alten Reglerparameter werden auf die aktuellen Reglerparameter übertragen und sind abhängig von PID_ON sofort aktiv. PI_CON = PI_CON_OLD PID_CON = PID_CON_OLD Die Funktion zum Rückgängigmachen von Änderungen ist nur im Automatik-Betrieb (PHASE 4) möglich.
SAVE_PAR	150.4	BOOL	FAL-SE	Die aktuellen Reglerparameter werden auf die alten Reglerparameter übertragen. PI_CON_OLD = PI_CON PID_CON_OLD = PID_CON
LOAD_PAR	150.5	BOOL	FAL-SE	Die geänderten Parameter PI_CON und PID_CON werden an den Regler übertragen und somit aktiv.

Siehe auch

Beschreibung TUN_ES (Seite 5872)

Arbeitsweise TUN_ES (Seite 5876)

Eingangsparameter TUN_ES (Seite 5880)

Ausgangsparameter TUN_ES (Seite 5881)

Parameter TUN_ES.PHASE (Seite 5885)

Parameter TUN_ES.STATUS_H (Seite 5888)

Parameter TUN_ES.STATUS_D (Seite 5889)

Parameter TUN_ES.PHASE

PHASE	Betriebsart	Beschreibung
0	Automatikbetrieb: Regelbetrieb mit voreingestellten Parametern	Bei Erstellung eines Instanz-DB zum Self-Tuner ist der Parameter PHASE mit Null vorbelegt. Da LMNS_ON mit TRUE vorbelegt ist, wird nach dem ersten Durchlauf der Self-Tuner nach PHASE 7 (Handbetrieb) wechseln. Sie können direkt vom Handbetrieb aus die Ersteinstellung starten (ADAPT_ON = TRUE) oder auf Automatikbetrieb umschalten. Beim erstmaligen Umschalten in den Automatikbetrieb MAN_ON = FALSE werden die Parameter von PI_CON übernommen. Die Vorbelegungswerte für GAIN und TI lauten: GAIN = 1.5, TI = 1h. Verlassen der Phase über MAN_ON, LMNS_ON oder ADAPT1ST
1	Regleroptimierung: Heizung aus und Warten auf einen Sollwertsprung	Nach dem Starten der Optimierung über ADAPT1ST = TRUE fährt der Self-Tuner das Stellventil zu (Stellwert = 0 bzw. LMNS_ON = LMNDN = TRUE) und wartet auf einen positiven Sollwertsprung ≥ MIN_STEP.
2	Regleroptimierung: Motorlaufzeitmessung, Anfahren auf eine konstante Stellventilstellung und Wendepunktsuche	Sobald Sie einen Sollwertsprung ≥ MIN_STEP in positiver Richtung zum Arbeitspunkt des warmen Prozesszustandes vorgeben, wird bei Ersteinstellung MAN_OUT mit LHLM_TUN und bei Nachoptimierung MAN_OUT mit einem vom Self-Tuner berechneten Wert belegt. MIN_STEP sollte größer als 10 % des Arbeitsbereichs von Soll- und Istwert sein. Bei der Schrittregelung mit Stellungsrückmeldung (LMNR_ON = TRUE) erhält QMAN_ON den Wert TRUE. Der Regler steuert das Ventil auf den Wert von MAN_OUT, und TUN_ES berechnet die Motorstellzeit MTR_TM. Bei der Schrittregelung ohne Stellungsrückmeldung werden QLMNS_ON und QLMNUP auf TRUE gesetzt, und das Ventil wird bis zum oberen Anschlag gefahren. Mit Erreichen des oberen Anschlags (LMNR_HS =TRUE) berechnet TUN_ES die Motorstellzeit und gibt sie an den Regler weiter. Anschließend wird das Ventil bis auf den Wert von MAN_OUT geschlossen (QLMNDN = TRUE).

PHASE	Betriebsart	Beschreibung
3	Regleroptimierung: Regelung mit einem PI-Grobregler	Wenn der Wendepunkt der Sprungantwort erkannt wird (STATUS_H = 2) oder der Istwert 70 % der Sprunghöhe des Sollwertes erreicht hat (STATUS_H = 3), wird ein vorsichtig eingestellter PI-Regler entworfen. Die Reglerverstärkung GAIN wird auf den Bereich 0.2 bis 20.0 begrenzt. Der Regler arbeitet ab sofort als PI-Regler und versucht die Strecke in einen stationären Zustand einzuregeln. Der Self-Tuner legt die neu ermittelten PI-Reglerparameter in GAIN, TI und PI_CON ab. Die alten Werte des PI-Reglers werden in PI_CON_OLD gesichert. Dauert es sehr lange, bis der stationäre Zustand erreicht ist (kriechendes Einschwingverhalten bei Temperaturprozessen), so können Sie schon beim annähernd stationären Zustand durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten anstoßen. Auch zu einem späteren Zeitpunkt in Phase 3 oder 4 können Sie durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten neu anstoßen. Oft wird der Reglerentwurf dadurch noch etwas verbessert. Sollte ein Überschwinger auftreten oder wird kein Wendepunkt gefunden, so kann das bei der Ersteinstellung an einer zu groß gewählten Stellwertsprunghöhe LHLM_TUN liegen und muss nicht zwangsläufig zu einer schlechten Reglereinstellung führen. Bei der nächsten Ersteinstellung sollten Sie LHLM_TUN ca. 20 % kleiner wählen. Hat der Baustein den stationären Zustand erkannt oder ist die Zeit 30 × TI (TI: Nachstellzeit des unter PHASE = 3 eingestellten PI-Reglers) seit dem Sollwertsprung verstrichen, wird ein verbesserter Reglerentwurf vorgenommen und auf PHASE = 4 umgeschaltet. Der Self-Tuner legt die neu ermittelten PI/PID-Reglerparameter in GAIN, TI, TD und PI_CON bzw. PID_CON ab. Die alten Werte des PID-Reglers werden in PID_CON_OLD gesichert. Wenn möglich, wird ein PID-Regler und ein PI-Regler entworfen. Falls PID_ON = TRUE, wird der PID-, anderenfalls der PI-Regler aktiviert. In PHASE 4 kann jederzeit zwischen diesen beiden Reglern stoßfrei umgeschaltet werden. Bei schwierigen Prozessen entwirft der Baustein unabhängig von PID_ON immer einen PI-Regler. Die Reglerverstärkung GAIN wird so begrenzt, dass die Kreisverstärkung des offenen Kreises (Produkt aus Reglerverstärkung und Prozessverstärkung) unter 40.0 liegt.
4	Automatikbetrieb: normaler Regelbetrieb mit/ohne Regelzone	In dieser Phase regelt der Regler mit seinen optimierten Parametern. Aus diesem Zustand heraus können Sie eine Nachoptimierung oder eine Ersteinstellung starten.
5	Automatikbetrieb, Struktur-Umschaltung bei Sollwertsprung: I-Anteil deaktiviert, P(D)-Regelung	Bei einem positiven Sollwertsprung ≥ MIN_STEP werden der I-Anteil des Reglers vorübergehend stark vergrößert und die Verstärkung etwas verkleinert, d. h. quasi ein P(D)-Regler benutzt. In Sollwertnähe wird wieder auf die ursprünglichen Werte des I-Anteils und der Verstärkung zurückgeschaltet (PHASE = 5 im Ablauf des Self-Tuners).

PHASE	Betriebsart	Beschreibung
6	Automatikbetrieb, Struktur-Umschaltung bei Sollwertsprung: prädiktive Stellwertvorgabe	Nur beim Schrittregler mit Stellungsrückmeldung Strecken mit großen Verzugszeiten sind nicht gut P(D)-regelbar. Daher wird nach einem positiven Sollwertsprung ≥ MIN_STEP die für den neuen Sollwert stationär erforderliche Stellgröße konstant ausgegeben. In Sollwertnähe wird stoßfrei wieder auf den PI- bzw. PID-Reglerbetrieb zurückgeschaltet. Dies ergibt ein langsames, aber überschwingfreies Verhalten.
7	Handbetrieb	Der Handbetrieb hat Priorität vor allen anderen Betriebsarten. Eine Ersteinstellung oder Nachoptimierung wird abgebrochen. Beim Abschalten des Handbetriebs (MAN_ON = FALSE und LMNS_ON = FALSE) geht der Regler in den Automatikbetrieb (PHASE = 4 im Ablauf des Self-Tuners) und regelt mit den vorhandenen Reglerparametern weiter. Falls Sie einen Schrittregler mit Stellungsrückmeldung (LMNR_ON = TRUE) verwenden, können Sie mit LMNS_ON = TRUE oder mit MAN_ON = TRUE auf Handbetrieb umschalten. Bei einem Schrittregler ohne Stellungsrückmeldung (LMNR_ON = FALSE) können Sie nur mit LMNS_ON = TRUE auf Handbetrieb umschalten. Wenn Sie den Eingang MAN_ON auf TRUE setzen, wird der Ausgang QMAN_ON auf TRUE und der Ausgang MAN_OUT auf MAN gesetzt. Wenn Sie den Eingang LMNS_ON auf TRUE setzen, wird der Ausgang QLMNUP auf LMNUP und der Ausgang QLMNDN auf LMNDN gesetzt.

Heizoptimierung beim Schrittregler ohne Stellungsrückmeldung

Das Bild zeigt die Phasenschritte der Ersteinstellung mit Messung der Motorlaufzeit.

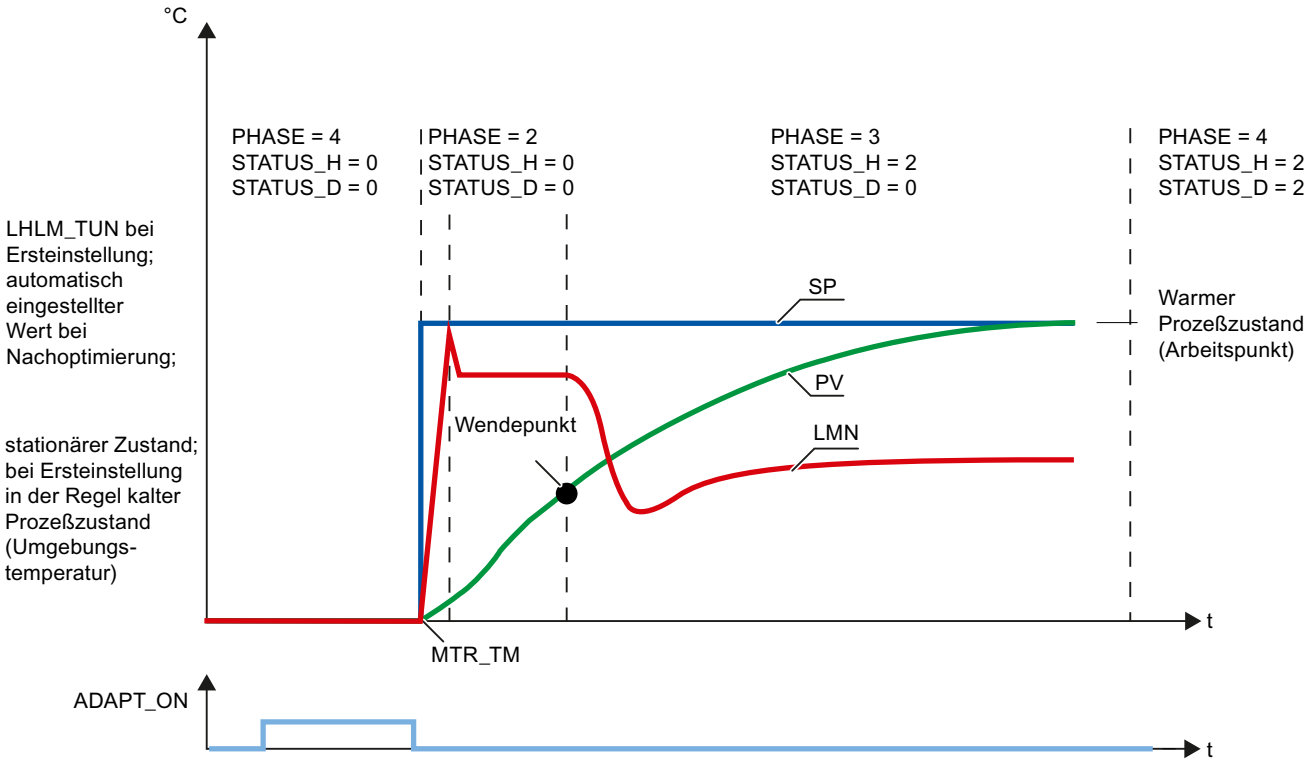

Siehe auch

Beschreibung TUN_ES (Seite 5872)
Arbeitsweise TUN_ES (Seite 5876)
Eingangsparameter TUN_ES (Seite 5880)
Durchgangsparameter TUN_ES (Seite 5884)
Parameter TUN_ES.STATUS_H (Seite 5888)
Parameter TUN_ES.STATUS_D (Seite 5889)

Parameter TUN_ES.STATUS_H

STATUS_H	Beschreibung	Abhilfe
0	Initialisierung	
1	Wendepunkt zu tief Die Optimierung führt in den meisten Fällen noch zu guten Reglerparametern..	Um näher an den Idealfall zu kommen, vergrößern Sie LHLM_TUN oder verkleinern Sie die Sollwertsprunghöhe
2	Idealfall: Wendepunkt gefunden	
3	Wendepunkt zu hoch Es wird ein Sicherheitsfaktor eingerechnet; die Optimierung führt in den meisten Fällen noch zu brauchbaren, aber etwas vorsichtigen Reglerparametern.	Um näher an den Idealfall zu kommen, verkleinern Sie LHLM_TUN oder vergrößern Sie die Sollwertsprunghöhe.
11, 12	Die Verzugszeit konnte nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann. Wenn der Prozess reines VZ1-Verhalten (keine Verzugszeit) hat, kann der Self-Tuner oftmals noch gute Reglerparameter entwerfen. Da mit einer großzügig geschätzten Verzugszeit weitergerechnet wird, kann es sein, dass STATUS_D nicht zu 1 (Regler für annäherndes VZ1-Verhalten) ermittelt und die Regelzone abgeschaltet wird, obwohl eine Regelzone bei VZ1-Prozessen durchaus sinnvoll ist.	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten. Sind Sie mit den Reglerparametern nicht zufrieden, können Sie zumindest die Werte aus PROCESS.GAIN und PROCESS.TM_LAG1 für ein VZ1-Modell und einen eigenen Reglerentwurf nach Lehrbuch verwenden.
13	Die Verzugszeit konnte nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann.	Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Siehe auch

Beschreibung TUN_ES (Seite 5872)
Arbeitsweise TUN_ES (Seite 5876)
Eingangsparameter TUN_ES (Seite 5880)
Durchgangsparameter TUN_ES (Seite 5884)
Parameter TUN_ES.PHASE (Seite 5885)
Parameter TUN_ES.STATUS_D (Seite 5889)

Parameter TUN_ES.STATUS_D

STATUS_D	Beschreibung
0	Initialisierung
1	nur PI-Reglerentwurf, da Prozess fast PT1-Verhalten hat
2	Idealfall: PI und PID-Regler möglich, VZ2-Verhalten
3	nur PI-Reglerentwurf, da sehr große Verzugszeit / Todzeit oder hohe Ordnung
4	Fehlschätzung Bei Nachoptimierung kein Reglerentwurf Bei Ersteinstellung wird mit dem groben PI-Regler aus PHASE = 3 weitergearbeitet. Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Siehe auch

Beschreibung TUN_ES (Seite 5872)

Arbeitsweise TUN_ES (Seite 5876)

Eingangsparameter TUN_ES (Seite 5880)

Durchgangsparameter TUN_ES (Seite 5884)

Parameter TUN_ES.PHASE (Seite 5885)

Parameter TUN_ES.STATUS_H (Seite 5888)

Standard PID Control

Technische Daten der Anweisungen

CPU–Belastung

Die Regelleistung und damit die Schnelligkeit der Bearbeitung von konkreten Regelkreisen hängt ausschließlich von der Leistung derverwendeten CPU ab. Um die Belastung einer bestimmten CPU durch die Installation von Standard PID Control abschätzen zu können, gibt es folgende Anhaltspunkte:

- Für eine beliebige Anzahl von Reglern muss die Anweisung im Anwenderspeicher der CPU nur einmal vorhanden sein.
- Pro Regler benötigen Sie einen DB mit ca. 0,5 KByte
- Speicherbedarf für einen L–Stack besteht nicht.
- Alarme werden durch PID_CP oder PID_ES nicht verzögert.
- Eckdaten für typische Laufzeiten (Bearbeitungszeiten) der Bausteine bei Parametrierung gemäß der Vorbelegung im Regelbetrieb:

Anweisung	Randbedingungen	Bearbeitungszeit (in ms) CPU 315–2AG10	Bearbeitungszeit (in ms) CPU 416–2XK02
PID_CP	typische Randbedingungen	1,3	0,14
PID_ES	ohne Stellungsrückmeldung, typische Randbedingungen	1,5	0,16

Arbeitsspeicher–Belegung

Die Größe des erforderlichen Bereichs im Anwenderspeicher und damit die Anzahl der Regelkreise, die aufgrund der verfügbaren Speicherkapazität theoretisch installiert werden könnten, ist den folgenden Tabellen zu entnehmen:

Anweisung	Ladespeicherbedarf	Arbeitsspeicher-bedarf	Lokaldaten
PID_CP	8956 Bytes	7796 Bytes	122 Bytes
PID_ES	9104 Bytes	7982 Bytes	152 Bytes
LP_SCHED	1064 Bytes	976 Bytes	20 Bytes

Instanz–DB bzw. Globaler DB	Ladespeicherbedarf	Arbeitsspeicherbedarf
DB zu PID_CP	1168 Bytes	510 Bytes
DB zu PID_ES	1124 Bytes	484 Bytes
DB_LOOP (bei 5 Regelkreisen)	184 Bytes	100 Bytes
DB_RMPSK (mit Startpunkt und 4 Stützstellen)	142 Bytes	78 Bytes

Abtastzeit

Die kleinste einstellbare Abtastzeit ist abhängig von der Leistungsklasse der eingesetzten CPU.

Hinweis

Die begrenzte Rechengenauigkeit schränkt die realisierbare Abtastzeit nach unten ein. Bei kleiner werdender Abtastzeit nehmen die Konstanten der Algorithmen immer kleinere Zahlenwerte an. Dies kann zu einer falschen Berechnung der Stellgröße führen.

Empfehlung:

- S7-300: Abtastzeit ≥ 20 ms
- S7-400: Abtastzeit ≥ 5 ms

PID_CP

Beschreibung PID_CP

Neben den Funktionen im Soll- und Istwertzeig realisiert die Anweisung einen fertigen PID-Regler mit kontinuierlichem Stellgrößen-Ausgang und Beeinflussungsmöglichkeit des Stellwertes von Hand. Teilfunktionen lassen sich zu oder abschalten.

Mit der Anweisung haben Sie die Möglichkeit, technische Prozesse und Anlagen mit kontinuierlichen Ein- und Ausgangsgrößen auf den Automatisierungssystemen SIMATIC S7 zu regeln. Es können sowohl träge Strecken (Temperaturen, Füllstände usw.) als auch sehr schnelle Strecken (Durchflüsse, Drehzahlen usw.) geregelt werden.

Der Regler kann als Festwertregler einzeln oder auch in mehrschleifigen Regelungen als Kaskaden-, Mischungs- oder Verhältnisregler eingesetzt werden.

Anlauf

PID_CP verfügt über eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangs–Parameter COM_RST = TRUE gesetzt ist.

Der I-Anteil wird beim Anlauf auf den Initialisierungswert I_ITLVAL gesetzt und am Ausgang LMN_I ausgegeben. Beim Aufruf in einer Weckalarmebene arbeitet er von diesem Wert aus weiter.

Bei eingeschaltetem Zeitplangeber werden die Zeitabschnitte DB_NBR PI[0 ... NBR_PTS].TMV zwischen den Stützpunkten aufsummiert und am Ausgang die Gesamtzeit T_TM und Gesamtrestzeit RT_TM angezeigt. Bei Online–Änderungen von PI[n].TMV oder bei Vorgabe von TM_CONT und TM_SNBR ändern sich Gesamtzeit und Gesamtrestzeit der Fahrkurve. Da die Berechnung von T_TM und RS_TM bei vielen Stützpunkten die Bearbeitungszeit der Funktion RMP_SOAK stark vergrößert, wird diese Berechnung nur nach Neustart oder bei TUPDT_ON = TRUE durchgeführt.

Alle anderen Ausgänge werden auf ihre Vorbelegungswerte gesetzt.

Wiederanlauf

Bei Wiederanlauf wird von dem Betriebszustand ausgegangen, der zum Zeitpunkt der Unterbrechung geherrscht hat. Die Regelung arbeitet mit den Werten weiter, die im Augenblick der Unterbrechung berechnet waren.

Aufruf

Die Anweisung PID_CP muss äquidistant aufgerufen werden. Benutzen Sie hierzu eine Weckalarmebene (z.B. OB35).

Wenn Sie die Anweisung PID_CP als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_CP direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Regeldifferenzbildung PID_CP

Die folgenden Funktionen sind immer wirksam und können nicht abgeschaltet werden.

- Sollwertbegrenzung SP_LIMIT,
- Istwertüberwachung auf Absolutwerte PV_ALARM,
- Istwertüberwachung auf Größe der Änderungsgeschwindigkeit ROCALARM,
- Regeldifferenzüberwachung auf Absolutwerte ER_ALARM,
- Begrenzung des Ausgangssignals LMNLIMIT.

Die Vorbelegungswerte sind so gewählt, dass der Betrieb ohne eine individuelle Anpassung auskommt. Um gute Regelergebnisse zu erzielen, müssen Sie die Parameter dieser Funktionen immer an ihren Prozess anpassen.

Istwert aufbereiten

Der Istwert wird auf die physikalische Größe normiert.

AUTOHOTSPOT

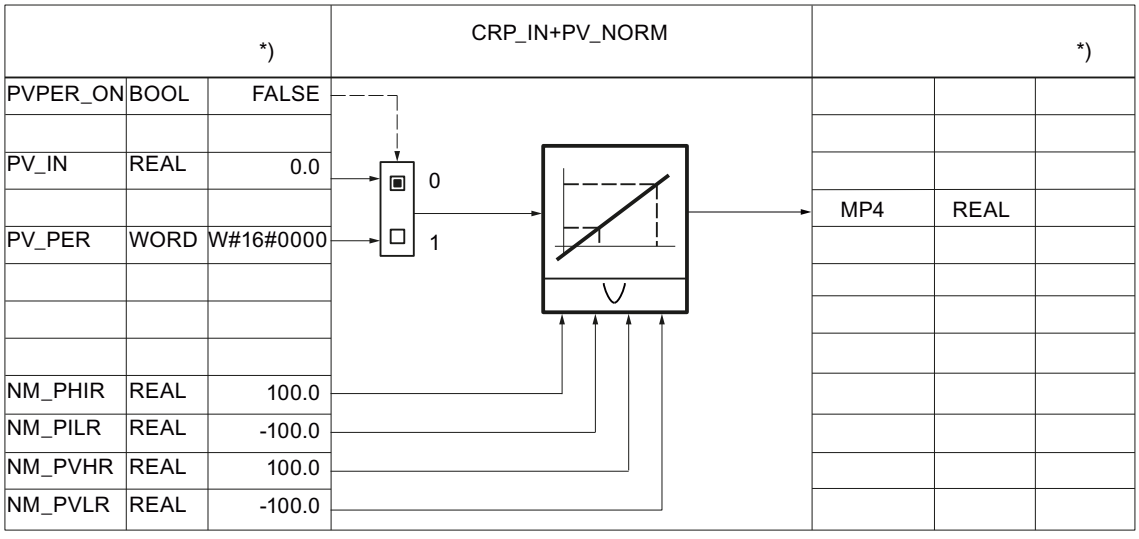

*) Vorbelegung bei Neuerstellung des Technologieobjekts.

Der Istwert kann mit einem Verzögerungsglied 1. Ordnung geglättet werden.

AUTOHOTSPOT

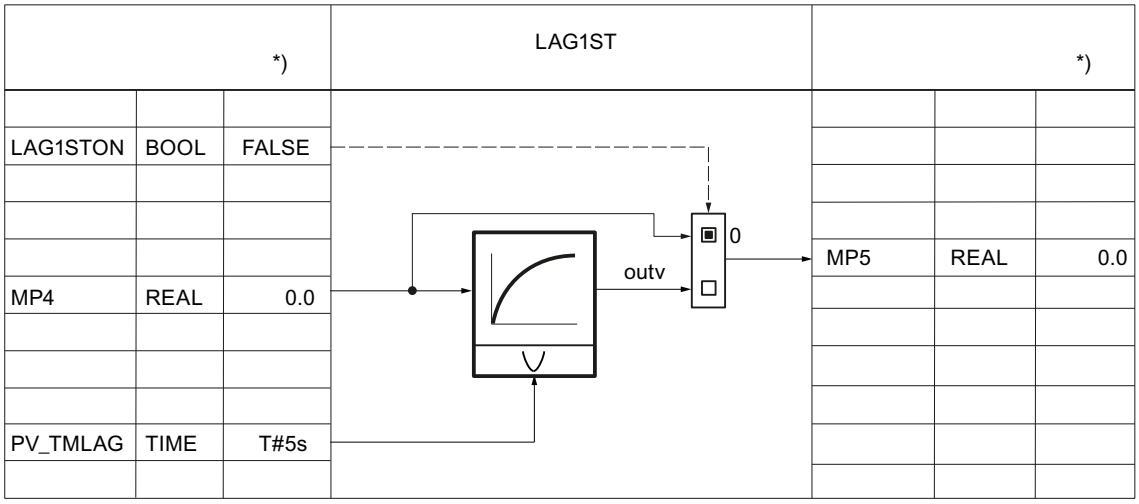

Der Istwert kann mit der Quadratwurzel linearisiert werden. Dazu sind folgende Parameter notwendig:

AUTOHOTSPOT

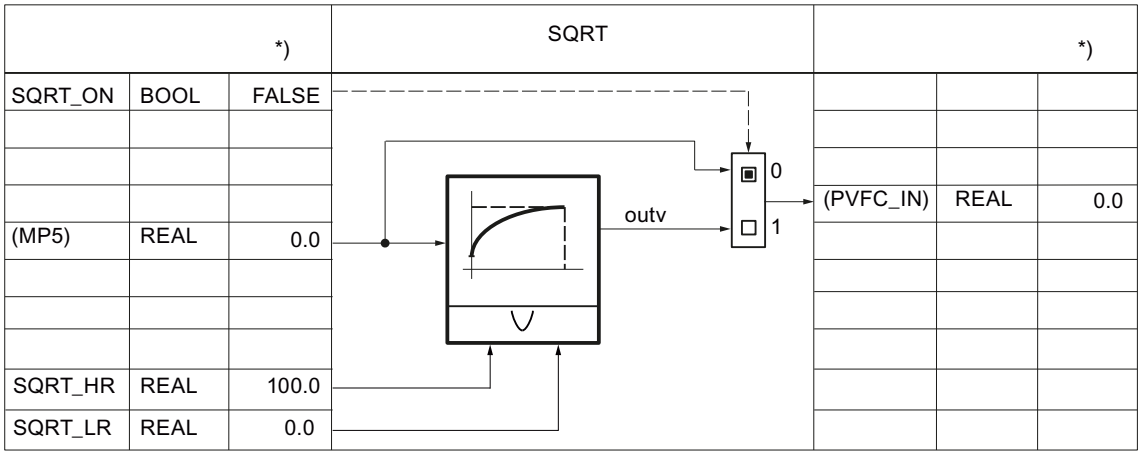

Der Istwert kann mit einer anwenderspezifischen Funktion aufbereitet werden. Dazu sind die folgenden Parameter notwendig. Die Verschaltung muss in der Anwender-FC programmiert werden.

AUTOHOTSPOT

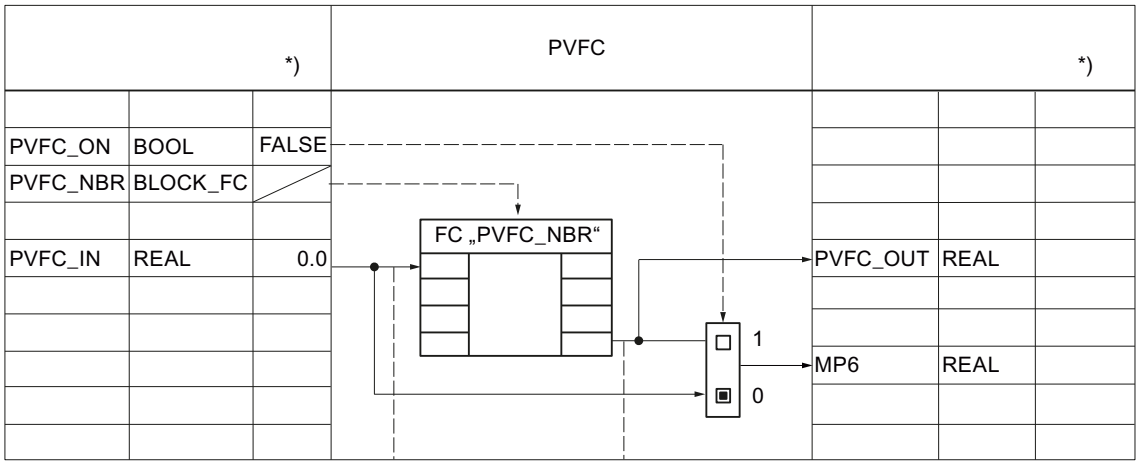

Der Istwert wird immer mit den folgenden Parametern auf Warn- und Alarmgrenzen überwacht.

AUTOHOTSPOT

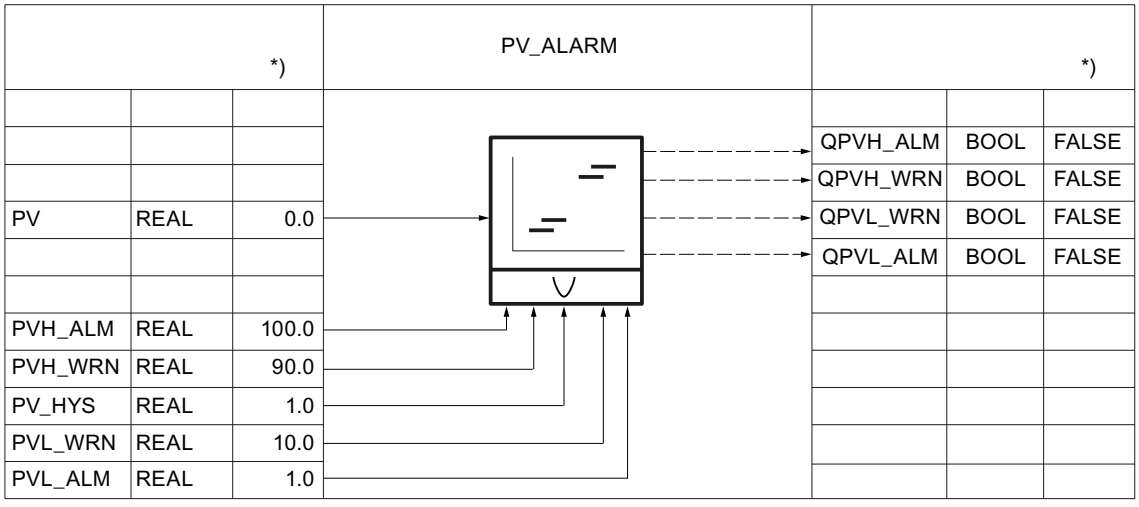

Die Änderungsgeschwindigkeit des Istwerts wird immer mit den folgenden Parametern begrenzt.

AUTOHOTSPOT

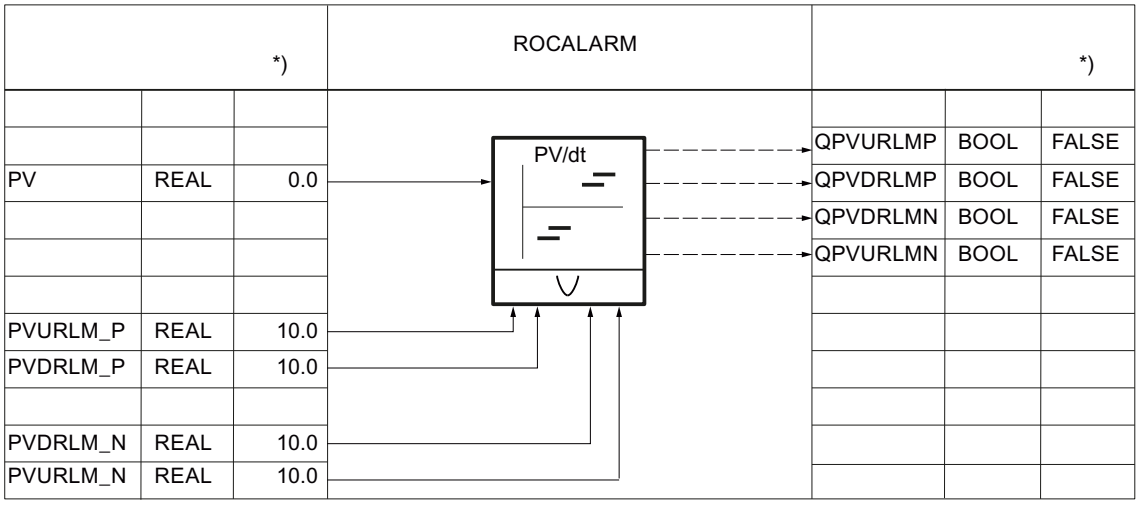

Sollwert aufbereiten

Der interne Sollwert SP_IN kann mit den folgenden Parametern vergrößert oder verkleinert werden.

AUTOHOTSPOT

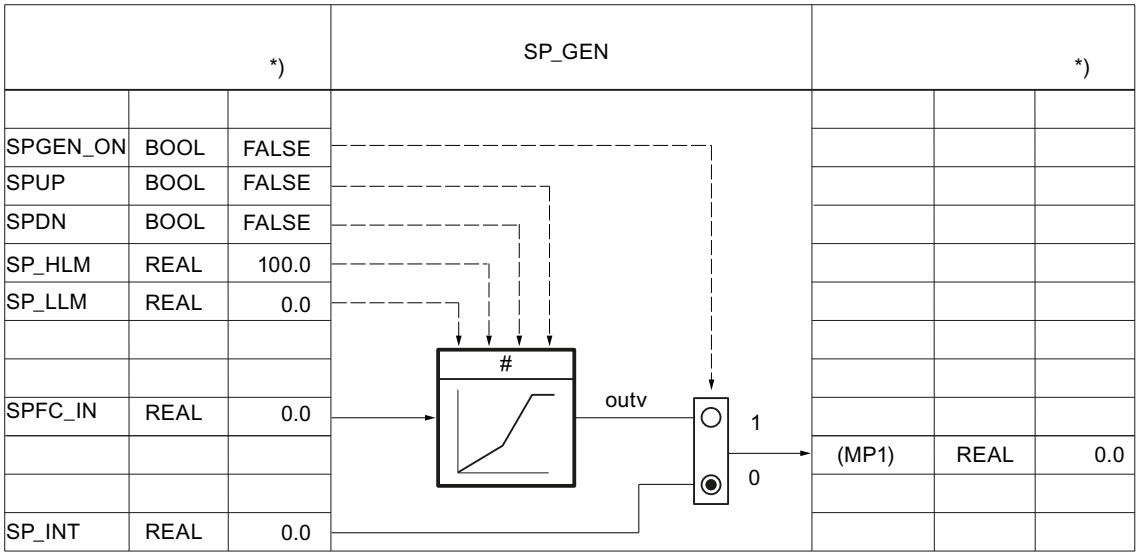

Der Sollwert kann mit den folgenden Parametern über einen Zeitplangeber vorgegeben werden.

AUTOHOTSPOT

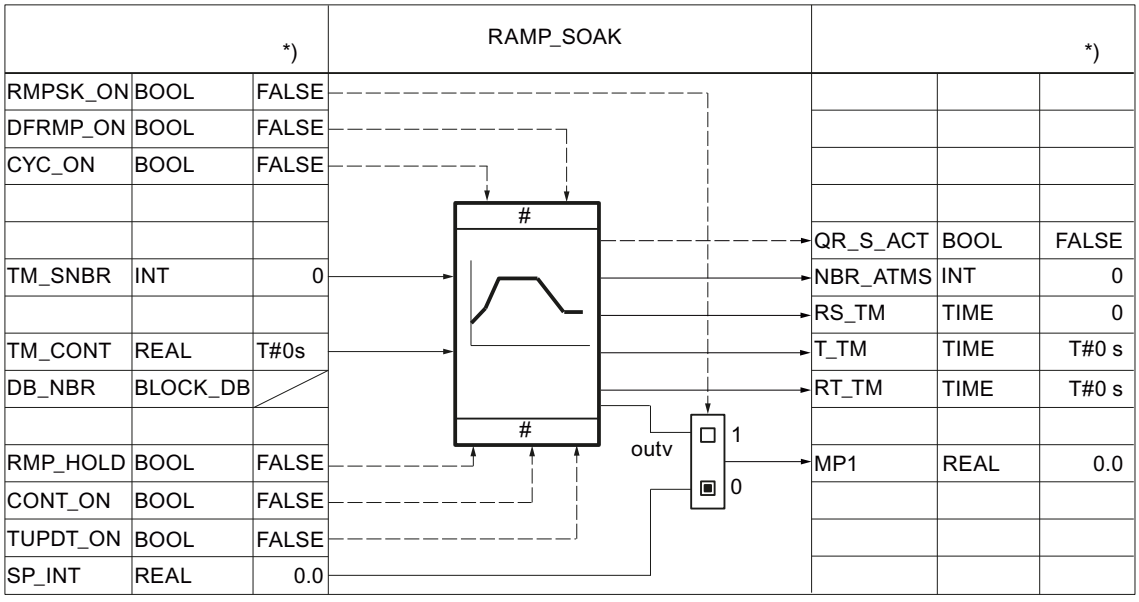

Für Mischungs-/Verhältnisregler kann der externe Sollwert SP_EXT mit den folgenden Parametern normiert werden.

AUTOHOTSPOT

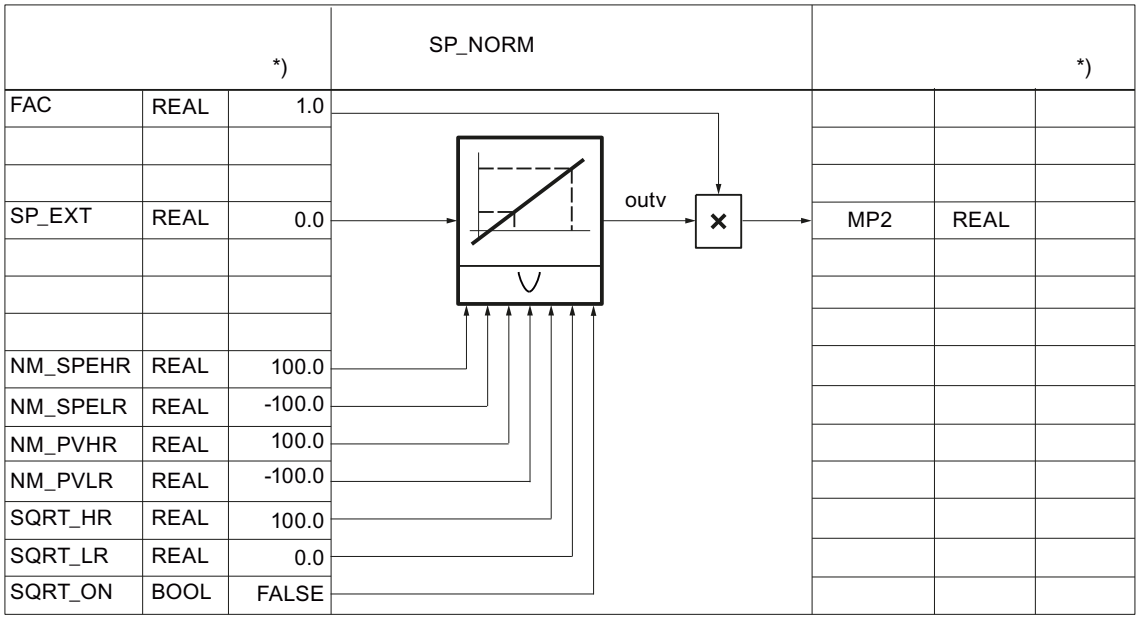

Der Sollwert kann mit einer anwenderspezifischen Funktion weiter aufbereitet werden. Dazu sind die folgenden Parameter notwendig. Die Verschaltung muss in der Anwender-FC programmiert werden.

Der Eingangswert SPFC_IN ist ein impliziter Parameter. Dieser ist am Konfigurationswerkzeug entweder über den Messpunkt MP1 (SPEXT_ON = FALSE) oder über den Messpunkt MP2 (SPEXT_ON = TRUE) zu beobachten. Der Ausgangswert ist am Messpunkt MP3 zugänglich. Der Eingang SPFC_IN ist dann auf den Sollwertzweig durchgeschaltet, wenn SPFC_ON =FALSE ist.

AUTOHOTSPOT

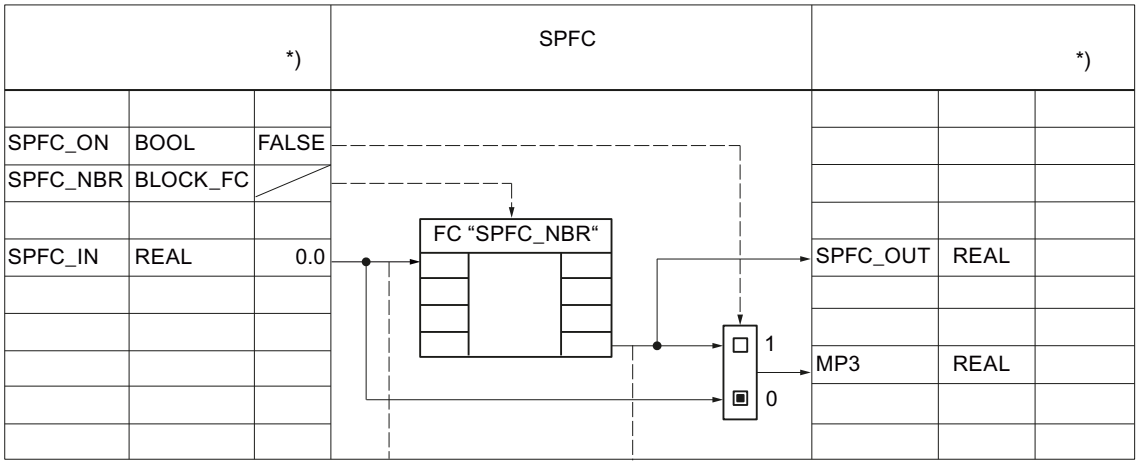

Die Änderungsgeschwindigkeit des Sollwerts kann mit den folgenden Parametern begrenzt werden.

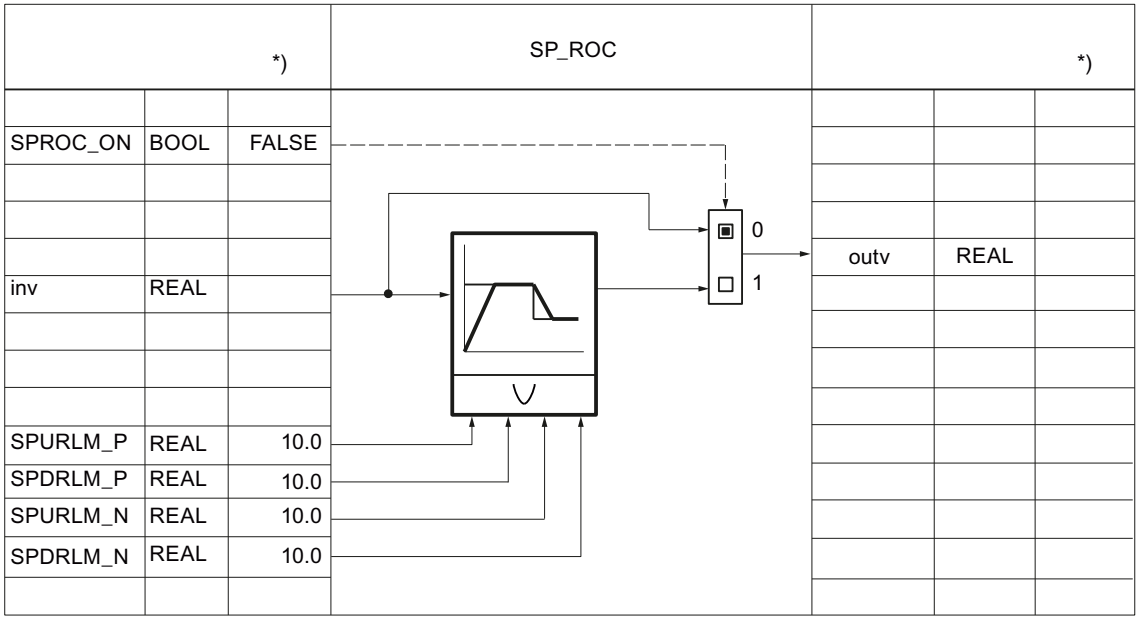

Der Sollwert wird immer mit den folgenden Parametern auf eine absolute Ober- und Untergrenze begrenzt.

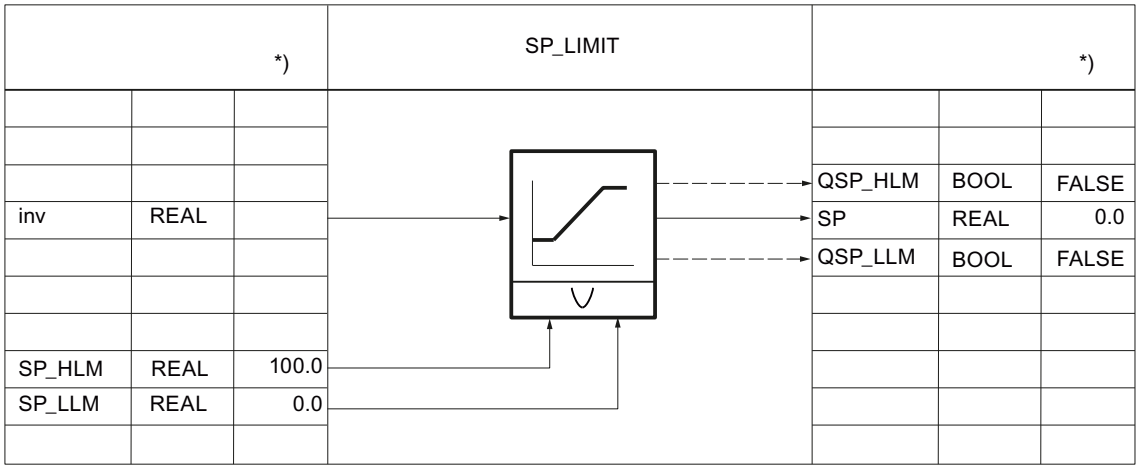

Regeldifferenz bilden

Die Regeldifferenz kann mit den folgenden Parametern über eine Totzone geleitet werden.

AUTOHOTSPOT

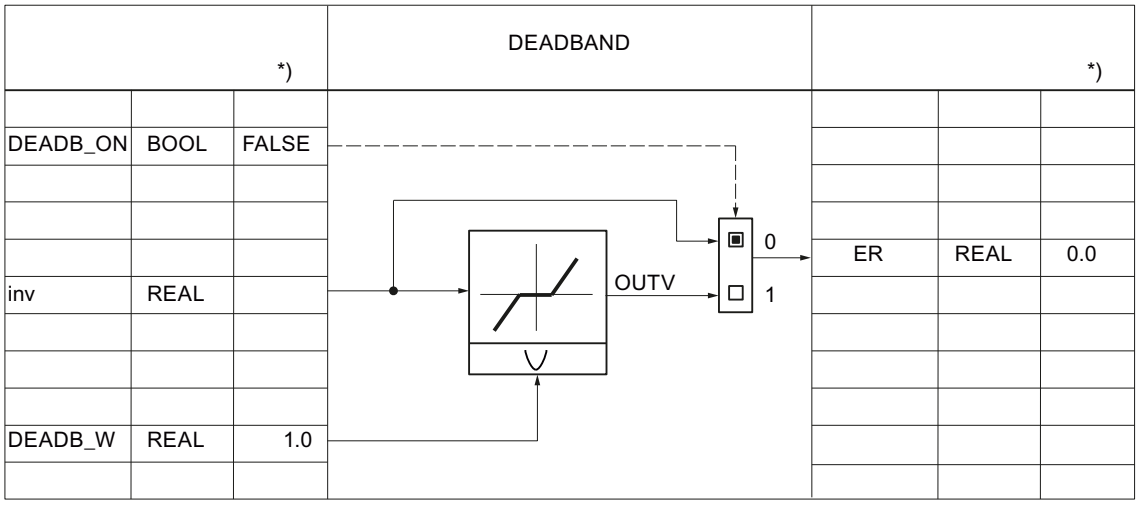

Die Regeldifferenz wird immer mit den folgenden Parametern auf Warn- und Alarmgrenzen überwacht.

AUTOHOTSPOT

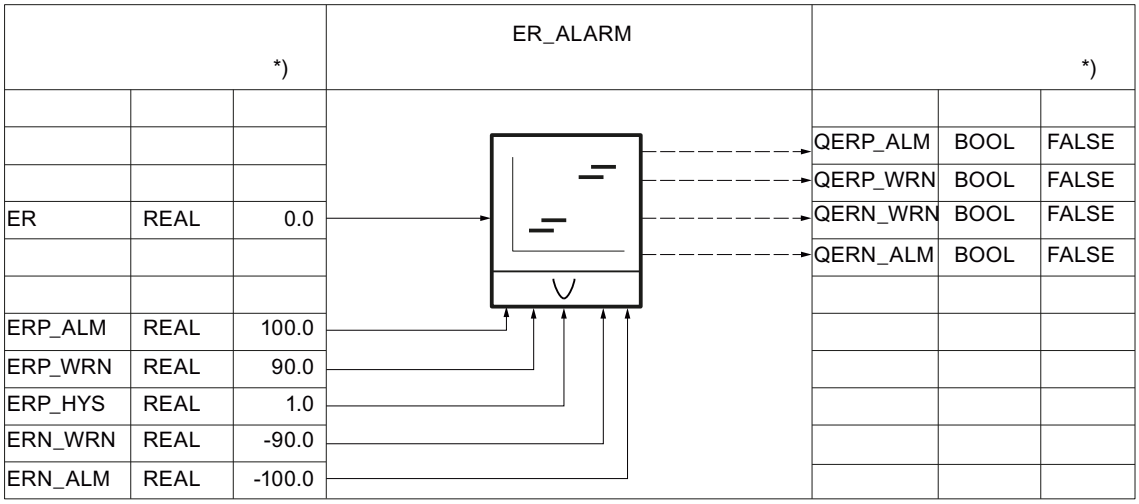

Regelalgorithmus PID_CP

Regelalgorithmus

AUTOHOTSPOT

Um eine wirksame Reglerstruktur festzulegen, stehen insgesamt fünf Schalter zur Verfügung.

AUTOHOTSPOT

Reglerstruktur	Schalter				
	P_SEL	I_SEL	D_SEL	PFDB_SEL	DFDB_SEL
P–Regelung	TRUE	FALSE	FALSE	FALSE	FALSE
P–Regelung (P in Rückführung)	TRUE	FALSE	FALSE	TRUE	FALSE
PI–Regelung	TRUE	TRUE	FALSE	FALSE	FALSE
PI–Regelung (P in Rückführung)	TRUE	TRUE	FALSE	TRUE	FALSE
PD–Regelung	TRUE	FALSE	TRUE	FALSE	FALSE
PD–Regelung (P in Rückführung)	TRUE	FALSE	TRUE	FALSE	TRUE
PID–Regelung	TRUE	TRUE	TRUE	FALSE	FALSE
PID–Regelung (P/D in Rückführung)	TRUE	TRUE	TRUE	FALSE	TRUE

Der P-Anteil wird durch die Proportionalverstärkung GAIN bestimmt und kann an LMN_P beobachtet werden.

AUTOHOTSPOT

Der I-Anteil wird mit den folgenden Parametern berechnet.

AUTOHOTSPOT

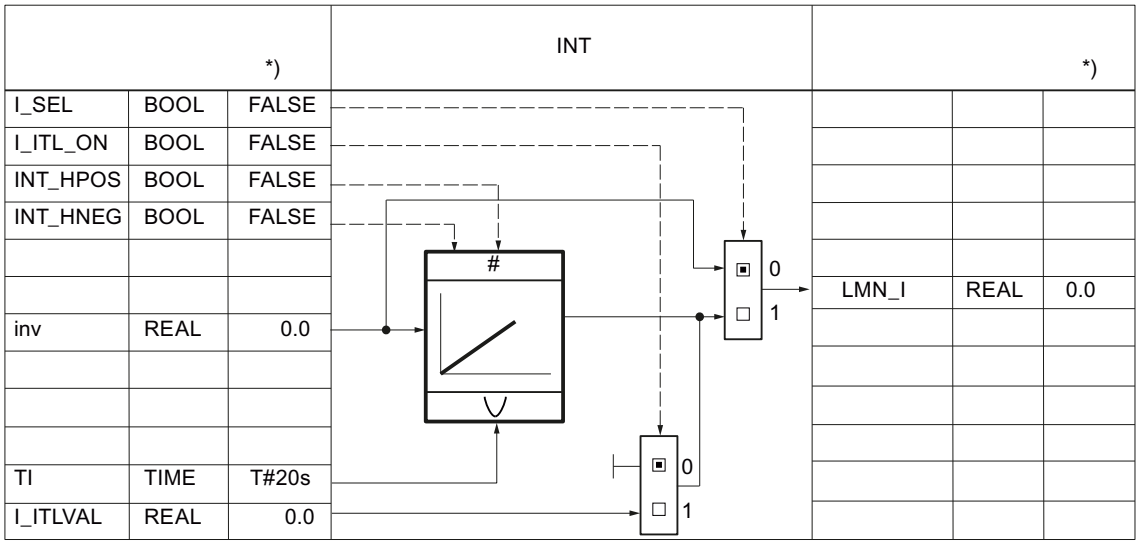

*) Vorbelegung bei Neuerstellung des Technologieobjekts

Der D-Anteil wird mit den folgenden Parametern berechnet.

AUTOHOTSPOT

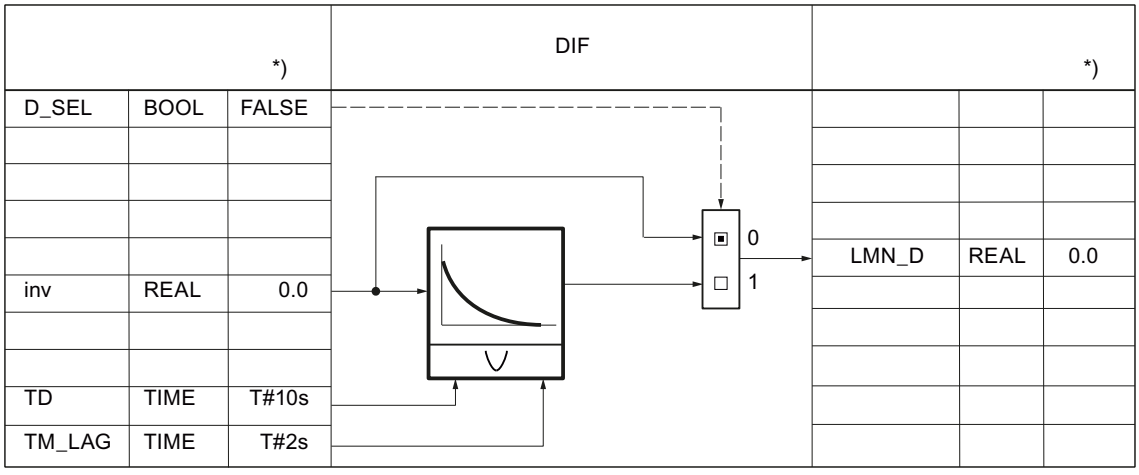

Stellwert PID_CP

Kontinuierlicher Stellwert

Der PID–Regelalgorithmus erzeugt einen analogen Stellwert, gegebenenfalls um einen Pulsgenerator zur Bildung von pulsweitenmodulierten Ausgangssignalen für Zweipunktregelung oder Dreipunktregelungen mit proportionalen Stellgliedern ergänzt.

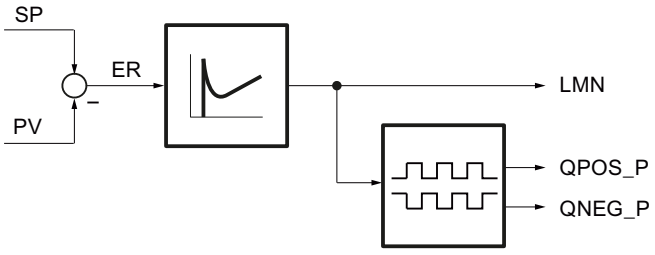

Der Stellwert kann mit den folgenden Parametern von Hand beeinflusst werden.

AUTOHOTSPOT

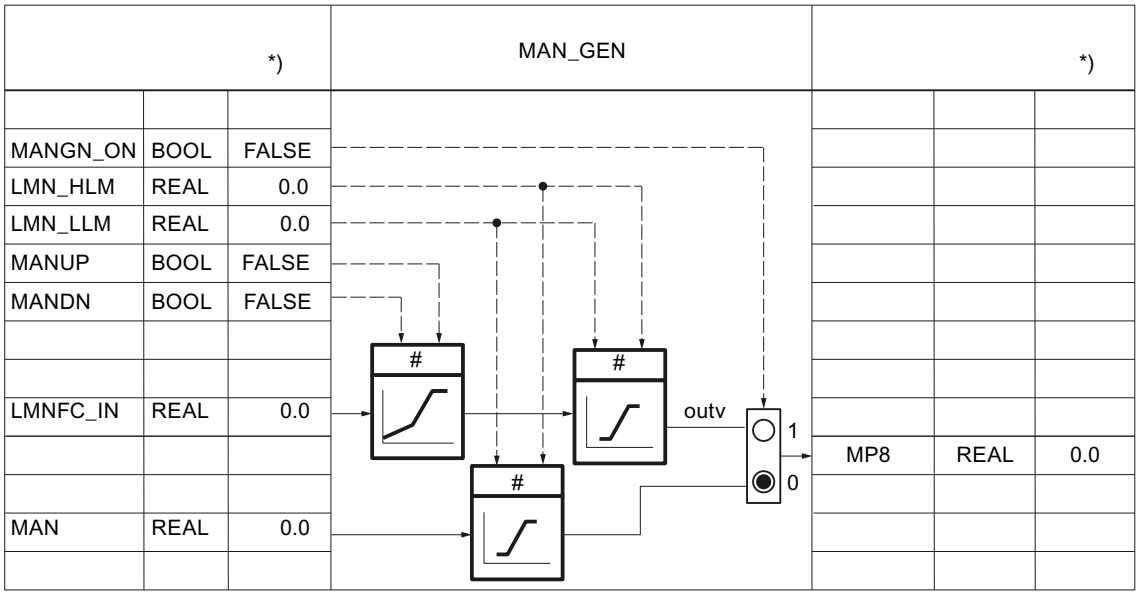

Der Stellwert kann mit einer anwenderspezifischen Funktion weiter aufbereitet werden. Dazu sind die folgenden Parameter notwendig. Die Verschaltung muss in der Anwender-FC programmiert werden.

AUTOHOTSPOT

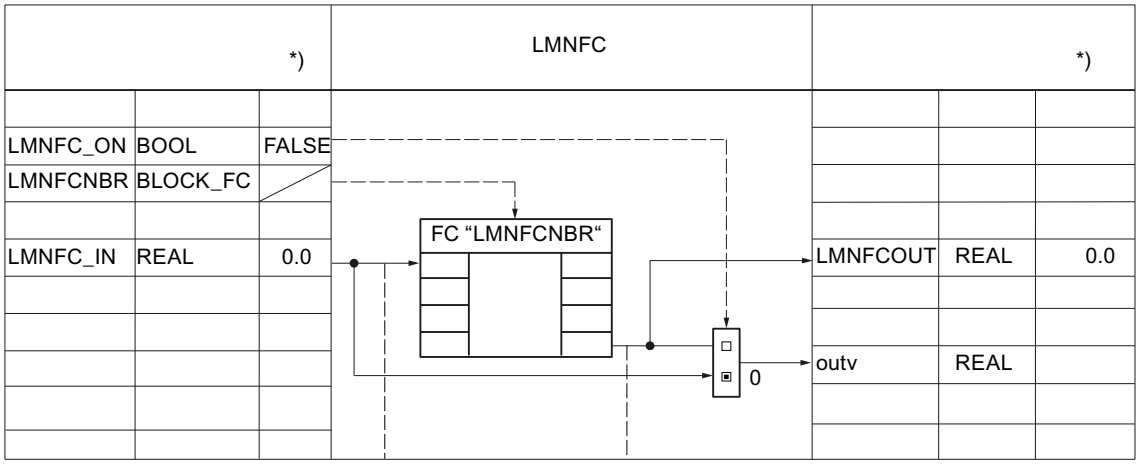

Die Änderungsgeschwindigkeit des Stellwerts kann mit den folgenden Parametern begrenzt werden.

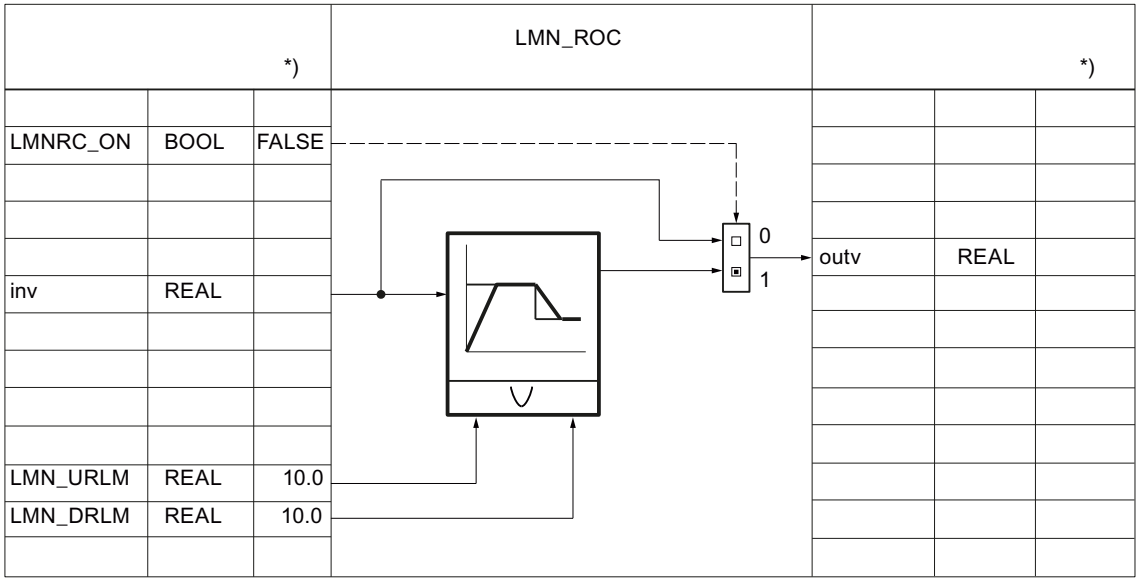

Der Stellwert wird immer mit den folgenden Parametern auf eine absolute Ober- und Untergrenze begrenzt.

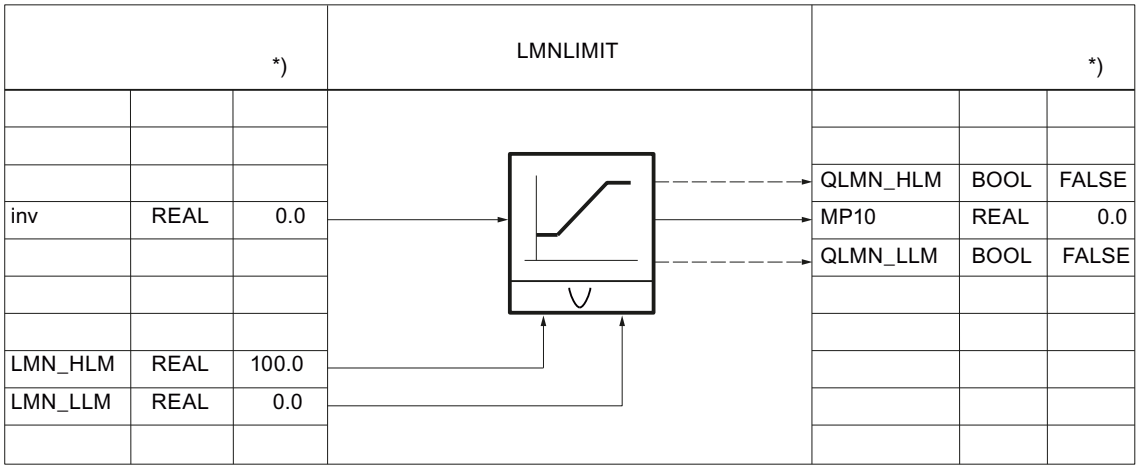

Der Stellwert wird mit den folgenden Parametern auf eine physikalische Größe normiert.

AUTOHOTSPOT

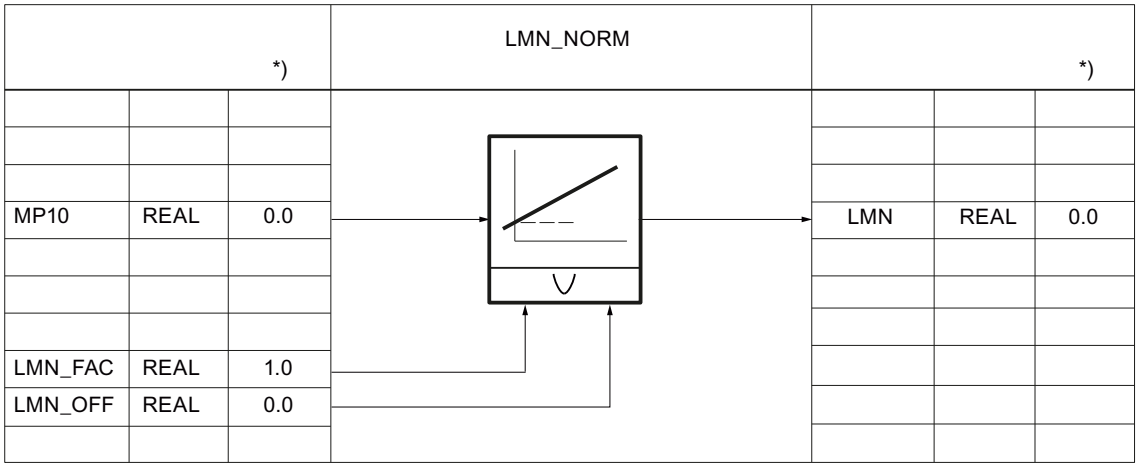

Der Stellwert im Gleitpunktformat wird in das Peripherieformat umgewandelt. Eine Prüfung auf positiven/negativen Überlauf und auf Erreichen des Über–/Untersteuerungsbereiches wird dabei nicht durchgeführt. Baugruppentypen werden nicht berücksichtigt.

Folgende Tabelle gibt einen Überblick über Bereiche und Zahlenwerte vor und nach der Bearbeitung.

Stellwert LMN in %	Peripheriewert LMN_PER
118.515	32767
100.000	27648
0.003617	1
0.000	0
-0.003617	-1
-100.000	-27648
-118.519	-32768

		*)	CRP_OUT			*)
LMN	REAL	0.0	%	LMN_PER	WORD	W#16#0000

Der kontinuierliche Stellwert kann über eine Pulsweitenmodulation als Impulsfolge ausgegeben werden. Dazu sind folgende Parameter notwendig:

AUTOHOTSPOT

		*)	PULSEGEN			*)
COM_RST	BOOL	FALSE				
CYCLE	TIME	T#10ms				
SELECT	BYTE	0				
PULSE_ON	BOOL	FALSE				
STEP3_ON	BOOL	TRUE				
ST2BI_ON	BOOL	FALSE				
			#			
				QC_ACT	BOOL	TRUE
MP10	REAL	0.0		QPOS_P	BOOL	FALSE
				QNEG_P	BOOL	FALSE
CYCLE_P	TIME	T#10ms				
PER_TM_P	TIME	T#1s				
PER_TM_N	TIME	T#1s				
P_B_TM_P	TIME	T#50ms				
P_B_TM_N	TIME	T#50ms				
RATIOFAC	REAL	1.0				

PID_CP in Kaskadenschaltungen

Auftrennung der Reglerkaskade

In einer Kaskade stehen mehrere Regler direkt miteinander in Beziehung. Sie müssen daher Vorkehrungen treffen, damit nach Auftrennung der Kaskadenstruktur an einer beliebigen Stelle der Kaskadenbetrieb wieder problemlos aufgenommen werden kann.

In Kaskadenregelungen wird deshalb in den unterlagerten Reglern über eine Oder–Logik aus den Zustandssignalen der Schalter im Sollwert– und Stellgrößenzweig ein Signal QCAS gebildet. Dieses Signal betätigt in den überlagerten Reglern einen Schalter, der die Regler in

den Nachführbetrieb überführt. Die Nachführgröße ist dabei immer der Sollwert des unterlagerten Kreises.

Die Umschaltung vom Nachführbetrieb in den Automatikbetrieb erfolgt stoßfrei wie bei der Hand– Automatikumschaltung.

PID_CP kann in überlagerten Regelkreisen als Führungsregler und in unterlagerten Regelkreisen als Folgeregler eingesetzt werden.

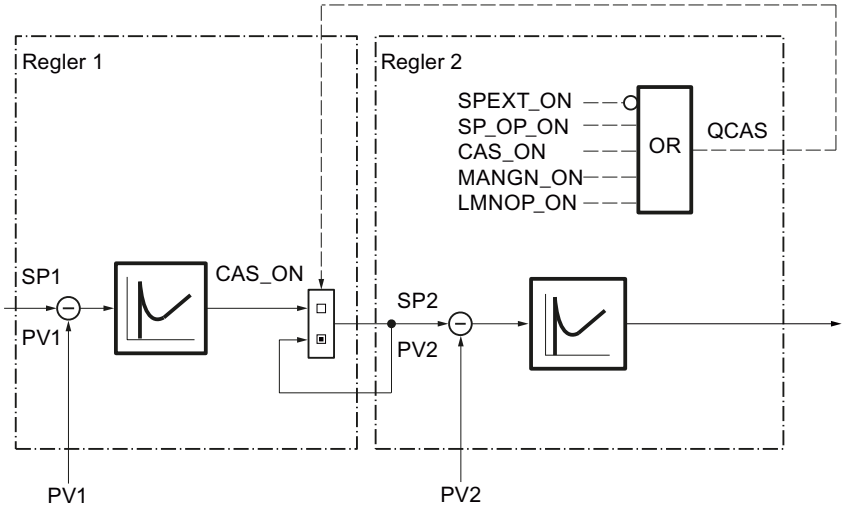

Hinweis

Die Verschaltung vom Stellwert des Führungsreglers LMN muss immer auf den externen Sollwert SP_EXT des Folgereglers gehen.

Bausteinverschaltung

Nachstehendes Bild zeigt das Prinzip der Reglerverschaltung in mehrschleifigen Kaskaden.

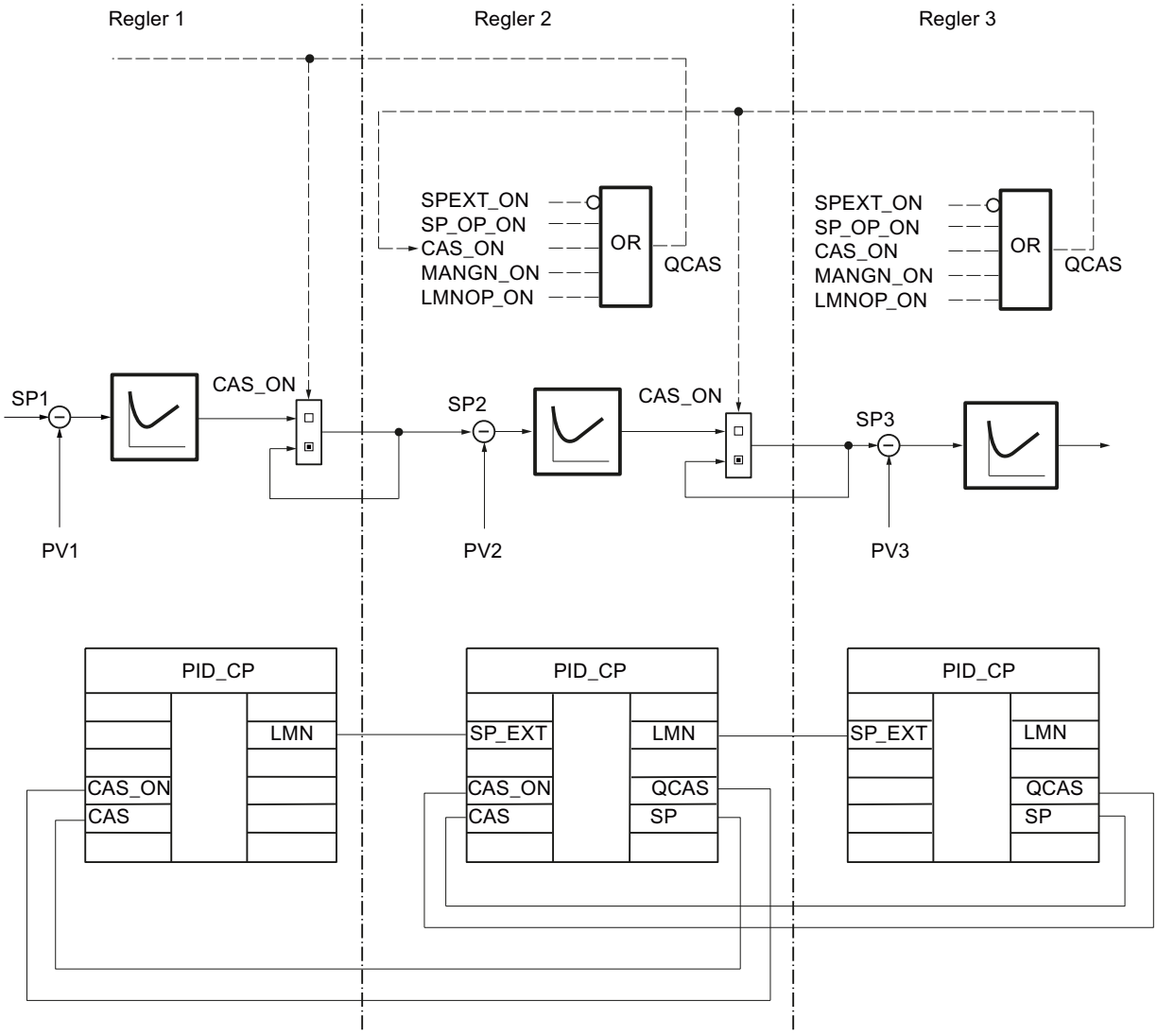

Eingangsparameter PID_CP

Tabelle 4-218

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL		FALSE	Neustart (Initialisierungsroutine der Anweisung wird bearbeitet)
I_SEL	0.1	BOOL		TRUE	I–Anteil einschalten
D_SEL	0.2	BOOL		FALSE	D–Anteil einschalten
MAN_ON	0.3	BOOL		TRUE	Handbetrieb einschalten (Regelkreis aufgetrennt, LMN von Hand eingestellt)

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
CAS_ON	0.4	BOOL		FALSE	Kaskadenbetrieb einschalten (verschaltet mit QCAS des unterlagerten Reglers)
SELECT	1.0	BYTE	0, 1, 2, 3	0	falls PULS_ON = TRUE: 0: PID und Pulsgenerator 1: PID (Baustein-Aufruf im OB 1) 2: Pulsgenerator (Baustein–Aufruf in Weckalarm–OB) 3: PID (Baustein.-Aufruf in Weckalarm–OB)
CYCLE	2.0	TIME	> 20 ms (S7–300)	T#1s	Abtastzeit (Zeit zwischen zwei Bausteinaufrufen = konstant) Achten Sie unbedingt darauf, dass Sie diesen Parameter mit der Weckalarmtaktzeit des OB, in dem die Anweisung PID_CP läuft, parametrieren! Andernfalls arbeiten die zeitabhängigen Funktionen nicht korrekt. (Ausnahme: Sie verwenden eine Zeituntersetzung, z. B. über den Regleraufrufverteiler.)
CYCLE_P	6.0	TIME		T#10 ms	Abtastzeit des Pulsgenerators Achten Sie unbedingt darauf, dass Sie diesen Parameter mit der Weckalarmtaktzeit des OB, in dem die Anweisung PID_CP läuft, parametrieren! Andernfalls arbeiten die zeitabhängigen Funktionen nicht korrekt. (Ausnahme: Sie verwenden eine Zeituntersetzung, z. B. über den Regleraufrufverteiler.)
SP_INT	10.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	interner Sollwert SP_INT kann mit SPUP und SPDN über ein HMI verändert werden
SP_EXT	14.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	externer Sollwert SP_EXT wird für Mischungs- und Verhältnisregler sowie Kaskadenregler verwendet und kann normiert werden.
PV_IN	18.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	Istwert–Eingang (PV im Gleitpunktformat)
PV_PER	22.0	INT		W#16#0000	Istwert von Peripherie (PV im Peripherieformat)
GAIN	24.0	REAL	gesamter Wertebereich (dimensionslos)	2.0	Proportionalbeiwert (= Reglerverstärkung)
TI	28.0	TIME	TI ≥ CYCLE	T#20s	Integrationszeit
TD	32.0	TIME	TD ≥ CYCLE	T#10s	Differenzierzeit
TM_LAG	36.0	TIME	TM_LAG ≥ CYCLE/2	T#2s	Verzögerungszeit des D–Anteils
DISV	40.0	REAL		0.0	Störgröße Zulässiger Wertebereich: -100.0 ... 100.0

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
CAS	44.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	Eingang für Kaskadenbetrieb (Verschaltung mit PV des unterlagerten Reglers)
SP_HLM	48.0	REAL	technischer Wertebereich (physikalische Größe)	100.0	Sollwert obere Begrenzung
SP_LLM	52.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	Sollwert untere Begrenzung
LMN_HLM	56.0	REAL	LMN_LLM ... 100.0	100.0	Stellwert: obere Begrenzung
LMN_LLM	60.0	REAL	-100.0 ... LMN_HLM	0.0	Stellwert: untere Begrenzung
DB_NBR	64.0	BLOCK_DB		DB 1	Datenbausteinnummer (DB mit den Stützpunkten der Fahrkurve)
SPFC_NBR	66.0	BLOCK_FC		FC 0	Sollwert FC–Nummer (selbstdefinierter FC im Sollwertzweig)
PVFC_NBR	68.0	BLOCK_FC		FC 0	Istwert FC–Nummer (selbstdefinierter FC im Istwertzweig)
LMNFCNBR	70.0	BLOCK_FC		FC 0	Stellwert FC–Nummer (selbstdefinierter FC im Stellwertzweig)

Ausgangsparameter PID_CP

Tabelle 4-219

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN	72.0	REAL	0.0	Stellwert (Stellwert im Gleitpunktformat)
LMN_PER	76.0	INT	W#16#0000	Stellwert für Peripherie (LMN im Peripherieformat)
SP	78.0	REAL	0.0	Sollwert (effektiv wirksamer Sollwert)
PV	82.0	REAL	0.0	Istwert (Ausgabe des effektiven Istwertes bei Kaskadebetrieb)
QCAS	86.0	BOOL	FALSE	Signal für Kaskadebetrieb (wird mit CAS_ON des überlagerten Reglers verschaltet)
QC_ACT	86.1	BOOL	TRUE	Anzeige, ob beim nächsten Bausteinaufruf der Regelungsteil bearbeitet wird (nur relevant, wenn SELECT den Wert 0 oder 1 hat)
QPOS_P	86.2	BOOL	FALSE	Pulsgenerator: positiver Puls ein
QNEG_P	86.3	BOOL	FALSE	Pulsgenerator: negativer Puls ein

Durchgangsparameter PID_CP

Tabelle 4-220

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
MAN	88.0	REAL	0.0	Hand–Stellwert (zur Stellwertvorgabe mittels B&B–Funktionen)

Statische Variablen PID_CP

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
PVH_ALM	92.0	REAL	PVH_WRN...100.0	100.0	Istwert: oberer Grenzwert 'Alarm'
PVH_WRN	96.0	REAL	PVL_WRN... PVH_ALM	90.0	Istwert: oberer Grenzwert 'Warnung'
PVL_WRN	100.0	REAL	PVL_ALM... PVH_WRN	-90.0	Istwert: unterer Grenzwert 'Warnung'
PVL_ALM	104.0	REAL	-100.0...PVL_WRN	-100.0	Istwert: unterer Grenzwert 'Alarm'
SPGEN_ON	108.0	BOOL		FALSE	Sollwertbedienung einschalten (Verstellen des Sollwerts über Auf–/Ab–Schalter)
SPUP	108.1	BOOL		FALSE	Sollwert hoch
SPDN	108.2	BOOL		FALSE	Sollwert tief
RMPSK_ON	108.3	BOOL		FALSE	Zeitplangeber einschalten (Sollwert wird als Fahrkurve vorgegeben)
SPEXT_ON	108.4	BOOL		FALSE	Externen Sollwert einschalten (zur Verschaltung mit anderen Reglerbausteinen)
MANGN_ON	108.5	BOOL		FALSE	Handbedienung einschalten (LMN per Schalter eingestellt)
MANUP	108.6	BOOL		FALSE	Hand–Stellwert Hoch
MANDN	108.7	BOOL		FALSE	Hand–Stellwert Tief
DFRMP_ON	109.0	BOOL		FALSE	Ausgang des Zeitplangebers vorbelegen (der Ausgang wird mit SP_INT vorbelegt)
CYC_ON	109.1	BOOL		FALSE	Zyklische Wiederholung einschalten (Fahrkurve wird automatisch wiederholt)
RMP_HOLD	109.2	BOOL		FALSE	Zeitplangeber anhalten (Ausgang des Zeitplangebers wird eingefroren)
CONT_ON	109.3	BOOL		FALSE	(Zeitplan) weitermachen (Fahrkurve wird an nächster Stützstelle fortgesetzt)
TUPDT_ON	109.4	BOOL		FALSE	Gesamtzeit neu berechnen (Gesamtzeit der Fahrkurve wird neu berechnet)
SPFC_ON	109.5	BOOL		FALSE	Sollwert–FC aufrufen

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SPROC_ON	109.6	BOOL		FALSE	Rampenfunktion einschalten (Begrenzung der Änderungsgeschwindigkeit von SP)
PVPER_ON	109.7	BOOL		FALSE	Istwert von Peripherie einschalten (Verschaltung mit Peripheriebaugruppen)
LAG1STON	110.0	BOOL		FALSE	Verzögerungsglied 1. Ordnung einschalten
SQRT_ON	110.1	BOOL		FALSE	Quadratwurzelfunktion einschalten
PVFC_ON	110.2	BOOL		FALSE	Istwert–FC aufrufen
DEADB_ON	110.3	BOOL		FALSE	Totzone einschalten (Kleine Störungen und Rauschen werden gefiltert)
P_SEL	110.4	BOOL		TRUE	P–Anteil einschalten
PFDB_SEL	110.5	BOOL		FALSE	P–Anteil in Rückführung schalten
INT_HPOS	110.6	BOOL		FALSE	Einfrieren des I–Anteils in positiver Richtung
INT_HNEG	110.7	BOOL		FALSE	Einfrieren des I–Anteils in negativer Richtung
I_ITL_ON	111.0	BOOL		FALSE	I–Anteil vorbesetzen
DFDB_SEL	111.1	BOOL		FALSE	D–Anteil in Rückführung schalten
DISV_SEL	111.2	BOOL		FALSE	Störgröße aufschalten
LMNFC_ON	111.3	BOOL		FALSE	Stellwert–FC aufrufen
LMNRC_ON	111.4	BOOL		FALSE	Stellwert–Rampenfunktion einschalten (Begrenzung Änderungsgeschwindigkeit von LMN)
SMOO_CHG	111.5	BOOL		TRUE	Stoßfreie Umschaltung von Hand nach Automatik
PULSE_ON	111.6	BOOL		FALSE	Pulsgenerator ein
STEP3_ON	111.7	BOOL		TRUE	Pulsgenerator: Dreipunktregelung einschalten
ST2BI_ON	112.0	BOOL		FALSE	Pulsgenerator: Zweipunktregelung für bipolaren Stellbereich einschalten (für unipolaren Stellbereich muss STEP3_ON = FALSE sein)
TM_SNBR	114.0	INT	≥ 0 (dimensionslos)	0	Nr. des nächsten Stützpunktes zum Weitermachen
TM_CONT	116.0	TIME	gesamter Wertebereich (dimensionslos)	T#0s	Weitermachzeitpunkt (Zeit nach Stützpunkt TM_SNBR zu der der Zeitplangeber die Fahrkurve weiter bearbeitet)
FAC	120.0	REAL	gesamter Wertebereich (dimensionslos)	1.0	Faktor (Verhältnis– oder Mischungsfaktor)
NM_SPEHR	124.0	REAL		100.0	Sollwert–Normierung: Arbeitsbereich Eingang oben
NM_SPELR	128.0	REAL		-100.0	Sollwert–Normierung: Arbeitsbereich Eingang unten
SPFC_OUT	132.0	REAL	-100.0 ... 100.0	0.0	Sollwert FC–Ausgang (wird mit dem Ausgang des FC im Sollwertzweig verschaltet)
SPURLM_P	136.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Anstiegsbegrenzung im positiven Bereich

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SPDRLM_P	140.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Abstiegsbegrenzung im positiven Bereich
SPURLM_N	144.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Anstiegsbegrenzung im negativen Bereich
SPDRLM_N	148.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Abstiegsbegrenzung im negativen Bereich
NM_PIHR	152.0	REAL		100.0	Istwert–Normierung: Messbereich Eingang oben
NM_PILR	156.0	REAL		-100.0	Istwert–Normierung: Messbereich Eingang unten
NM_PVHR	160.0	REAL		100.0	Istwert–Normierung: Messbereich Ausgang oben
NM_PVLR	164.0	REAL		-100.0	Istwert–Normierung: Messbereich Ausgang unten
PV_TMLAG	168.0	TIME	gesamter Wertebereich	T#5s	Istwertverzögerungszeit (Verzögerung des PT1–Gliedes im PV–Zweig)
SQRT_HR	172.0	REAL		100.0	Quadratwurzel: Arbeitsbereich Ausgang oben
SQRT_LR	176.0	REAL		0.0	Quadratwurzel: Arbeitsbereich Ausgang unten
PVFC_OUT	180.0	REAL	-100.0 ... 100.0	0.0	Istwert FC–Ausgang (wird mit dem Ausgang des FC im Istwertzweig verschaltet)
PVURLM_P	184.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Anstiegsbegrenzung im positiven Bereich
PVDRLM_P	188.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Abstiegsbegrenzung im positiven Bereich
PVURLM_N	192.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Anstiegsbegrenzung im negativen Bereich
PVDRLM_N	196.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Abstiegsbegrenzung im negativen Bereich
PV_HYS	200.0	REAL	≥ 0	1.0	Istwert Hysterese (Vermeidet 'Flackern' der Überwachungsanzeige)
DEADB_W	204.0	REAL	0.0 ... 100.0	1.0	Totzonenbreite (= Bereich Null bis Totzonenobergrenze) (bestimmt Größe des Unempfindlichkeitsbereichs)
ERP_ALM	208.0	REAL	0 ... 200.0	100.0	Regeldifferenz: positiver Grenzwert 'Alarm'
ERP_WRN	212.0	REAL	0 ... 200.0	90.0	Regeldifferenz: positiver Grenzwert 'Warnung'
ERN_WRN	216.0	REAL	-200.0 ... 0	-90.0	Regeldifferenz: negativer Grenzwert 'Warnung'
ERN_ALM	220.0	REAL	-200.0 ... 0	-100.0	Regeldifferenz: negativer Grenzwert 'Alarm'
ER_HYS	224.0	REAL	≥ 0 [%]	1.0	Regeldifferenz Hysterese (Vermeidet 'Flackern' der Überwachungsanzeige)
I_ITLVAL	228.0	REAL	-100.0 ... 100.0 [%]	0.0	Initialisierungswert für I–Anteil

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
LMNFCOUT	232.0	REAL	-100.0 ... 100.0 [%]	0.0	Stellwert FC–Ausgang (wird mit dem Ausgang des FC im Stellwertzweig verschaltet)
LMN_URLM	236.0	REAL	≥ 0 [%/s]	10.0	Stellwert–Anstiegsbegrenzung
LMN_DRLM	240.0	REAL	≥ 0 [%/s]	10.0	Stellwert–Abstiegsbegrenzung
LMN_FAC	244.0	REAL	gesamter Wertebereich (dimensionslos)	1.0	Stellwertfaktor (Faktor zur Anpassung des Stellwertbereichs)
LMN_OFF	248.0	REAL	gesamter Wertebereich (dimensionslos)	0.0	Stellwert–Offset (Nullpunkt der Stellwert–Normierung)
PER_TM_P	252.0	TIME		T#1s	Pulsgenerator: Periodendauer des positiven Pulses
PER_TM_N	256.0	TIME		T#1s	Pulsgenerator: Periodendauer des negativen Pulses
P_B_TM_P	260.0	TIME		T#50ms	Pulsgenerator: Mindestimpuls– bzw. Mindestpausendauer des positiven Pulses
P_B_TM_N	264.0	TIME		T#50ms	Pulsgenerator: Mindestimpuls– bzw. Mindestpausendauer des negativen Pulses
RATIOFAC	268.0	REAL	0.1 ... 10.0 (dimensionslos)	1.0	Pulsgenerator: Verhältnisfaktor (Verhältnis von positiver Impulsdauer und negativer Impulsdauer)
PHASE	272.0	INT		0	Phase vom PID Self Tuner
QPVH_ALM	274.0	BOOL		FALSE	Istwert: oberer Grenzwert 'Alarm' angesprochen
QPVH_WRN	274.1	BOOL		FALSE	Istwert: oberer Grenzwert 'Warnung' angesprochen
QPVL_WRN	274.2	BOOL		FALSE	Istwert: unterer Grenzwert 'Warnung' angesprochen
QPVL_ALM	274.3	BOOL		FALSE	Istwert: unterer Grenzwert 'Alarm' angesprochen
QR_S_ACT	274.4	BOOL		FALSE	Zeittabelle für Fahrkurve wird bearbeitet
QSP_HLM	274.5	BOOL		FALSE	Sollwert: obere Begrenzung angesprochen
QSP_LLM	274.6	BOOL		FALSE	Sollwert: untere Begrenzung angesprochen
QPVURLMP	274.7	BOOL		FALSE	Istwert: Anstiegsgrenzwert im positiven Bereich angesprochen
QPVDRLMP	275.0	BOOL		FALSE	Istwert: Abstiegsgrenzwert im positiven Bereich angesprochen
QPVURLMN	275.1	BOOL		FALSE	Istwert: Anstiegsgrenzwert im negativen Bereich angesprochen
QPVDRLMN	275.2	BOOL		FALSE	Istwert: Abstiegsgrenzwert im negativen Bereich angesprochen
QERP_ALM	275.3	BOOL		FALSE	Regeldifferenz: positiver Grenzwert 'Alarm' angesprochen
QERP_WRN	275.4	BOOL		FALSE	Regeldifferenz: positiver Grenzwert 'Warnung' angesprochen
QERN_WRN	275.5	BOOL		FALSE	Regeldifferenz; negativer Grenzwert 'Warnung' angesprochen

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
QERN_ALM	275.6	BOOL		FALSE	Regeldifferenz: negativer Grenzwert 'Alarm' angesprochen
QLMN_HLM	275.7	BOOL		FALSE	Stellwert: obere Begrenzung angesprochen
QLMN_LLM	276.0	BOOL		FALSE	Stellwert: untere Begrenzung angesprochen
NBR_ATMS	278.0	INT		0	Nummer des vom Zeitplangeber aktuell angefahrenen Stützpunktes
RS_TM	280.0	TIME		T#0s	"Aktuelle Restzeit" der Fahrkurve bis zum nächsten Stützpunkt
T_TM	284.0	TIME		T#0s	"Gesamtzeit" der Fahrkurve
RT_TM	288.0	TIME		T#0s	"Gesamtrestzeit" = Zeit bis zum Ende der Fahrkurve
ER	292.0	REAL		0.0	Regeldifferenz
LMN_P	296.0	REAL		0.0	P–Anteil
LMN_I	300.0	REAL		0.0	I–Anteil
LMN_D	304.0	REAL		0.0	D–Anteil
SPFC_IN	308.0	REAL		0.0	Sollwert FC–Eingang (wird mit Eingang des selbstdefinierten FC verschaltet)
PVFC_IN	312.0	REAL		0.0	Istwert FC–Eingang (wird mit Eingang des selbstdefinierten FC verschaltet)
LMNFC_IN	316.0	REAL		0.0	Stellwert FC–Eingang (wird mit Eingang des selbstdefinierten FC verschaltet)
SP_OP_ON	320.0	BOOL		FALSE	Sollwertbedienung einschalten (der Wert von SP_OP wird als Sollwert übernommen)
PV_OP_ON	320.1	BOOL		FALSE	Istwertbedienung einschalten (der Wert von PV_OP wird als Sollwert übernommen)
LMNOP_ON	320.2	BOOL		FALSE	Stellwertbedienung einschalten (der Wert von LMN_OP wird als Sollwert übernommen)
SP_OP	322.0	REAL		0.0	Sollwertbedienung bei der Inbetriebnahme
PV_OP	326.0	REAL		0.0	Istwertbedienung bei der Inbetriebnahme
LMN_OP	330.0	REAL		0.0	Stellwertbedienung bei der Inbetriebnahme
MP1	334.0	REAL		0.0	Messpunkt 1: Interner Sollwert
MP2	338.0	REAL		0.0	Messpunkt 2: Externer Sollwert
MP3	342.0	REAL		0.0	Messpunkt 3: Unbegrenzter Sollwert
MP4	346.0	REAL		0.0	Messpunkt 4: Istwert von der Peripheriebaugruppe
MP5	350.0	REAL		0.0	Messpunkt 5: Istwert nach dem Verzögerungsglied 1. Ordnung
MP6	354.0	REAL		0.0	Messpunkt 6: Wirksamer Istwert (PV)
MP7	358.0	REAL		0.0	Messpunkt 7: Stellwert vom PID–Algorithmus
MP8	362.0	REAL		0.0	Messpunkt 8: Hand–Stellwert

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
MP9	366.0	REAL		0.0	Messpunkt 9: Unbegrenzter Stellwert
MP10	370.0	REAL		0.0	Messpunkt 10: Begrenzter Stellwert

Blockschaltbilder PID_CP

Signalflussdastellungen

Analog zur Darstellung der Schalter im Konfigurationswerkzeug bedeutet der schwarze Punkt in den Schaltersymbolen, dass das Schaltsignal den nebenstehenden boolschen Wert (0 = FALSE oder 1 = TRUE) hat und dass der Signalweg jeweils über diesen Punkt durchgeschaltet ist. Die Schaltsignale (Binärsignale) sind durch gestrichelte Linien gekennzeichnet.

Das bedeutet, es wird der Sollwert absolut über SP_INT vorgegeben, das gleiche gilt für die Eingabe des Istwertes über PV_IN. Als Regelfunktion ist ein normaler PI–Regler mit P–Funktion im Vorwärtszweig eingestellt. Der Regelkreis ist geöffnet, und die Stellgröße wird im Prozent–Bereich über den Eingang MAN beeinflusst. Alle übrigen Funktionen sind passiv oder - falls nicht abschaltbar - durch Vorbesetzung von "Rand"–Parametern bei Signalverlauf innerhalb des Messbereichs bzw. des Arbeitsbereiches nicht wirksam.

Symbole und Kennungen in den Signalflussplänen

Die Bezeichnungen der anschließbaren Prozessgrößen sind unterlegt dargestellt. Sie erkennen daran, wo die Reglerstruktur mit der S7–Peripherie oder direkt mit den Messgliedern und den Stellgliedern des Prozesses verbunden werden kann.

Parameterbezeichnungen mit der Buchstabengruppe "OP" (z. B. SP_OP/ SP_OP_ON) verweisen auf den an dieser Stelle möglichen Eingriff über das Konfigurationswerkzeug von Standard PID Control. Das Konfigurationswerkzeug hat eine eigene Schnittstelle zum Regler–FB.

Zwischengrößen im Signalverlauf können durch die mit Kreisen markierten Messpunkte MP1 ... MP12 beobachtet werden. Diese Zwischengrößen werden zum Abgleich von Werten vor Auslösung "stoßfreier" Umschaltvorgänge oder zur Einsichtnahme in aktuelle Zustände der betreffenden Regelung benötigt. Die Messpunktgrößen im Kurvenschreiber des Konfigurationswerkzeugs können (statisch und dynamisch) dargestellt werden.

Der Übersichtlichkeit wegen sind die Parameter zur Einstellung und Dimensionierung der Bearbeitungsvorgänge (Algorithmen) bei den einzelnen Funktionsblöcken jeweils mit angegeben. Wir verweisen dazu auf die Beschreibungen im Referenzteil und auf die Darstellung einzelner Teilfunktionen in den folgenden Abschnitten.

Aufbereitung des Sollwerts

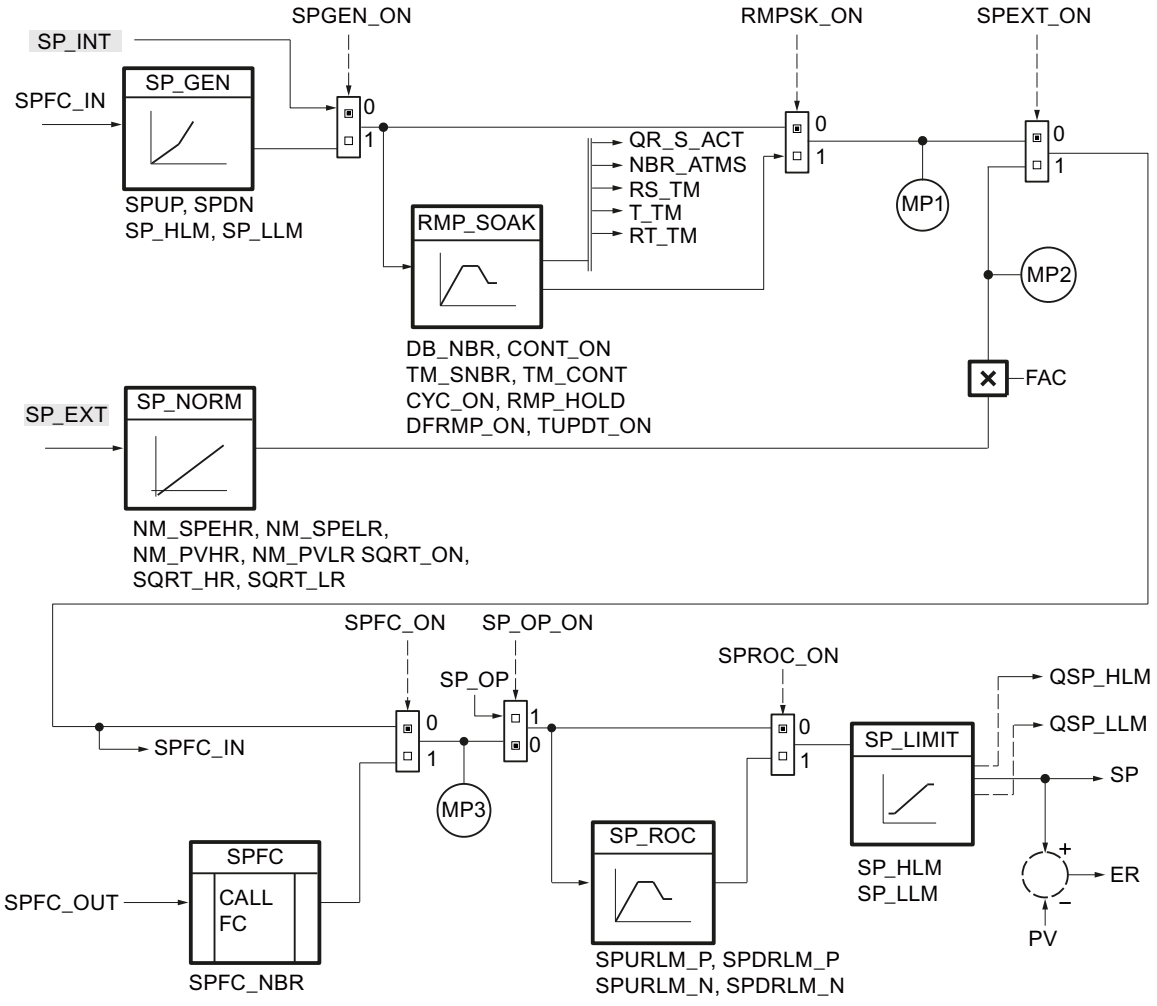

Aufbereitung des Istwerts

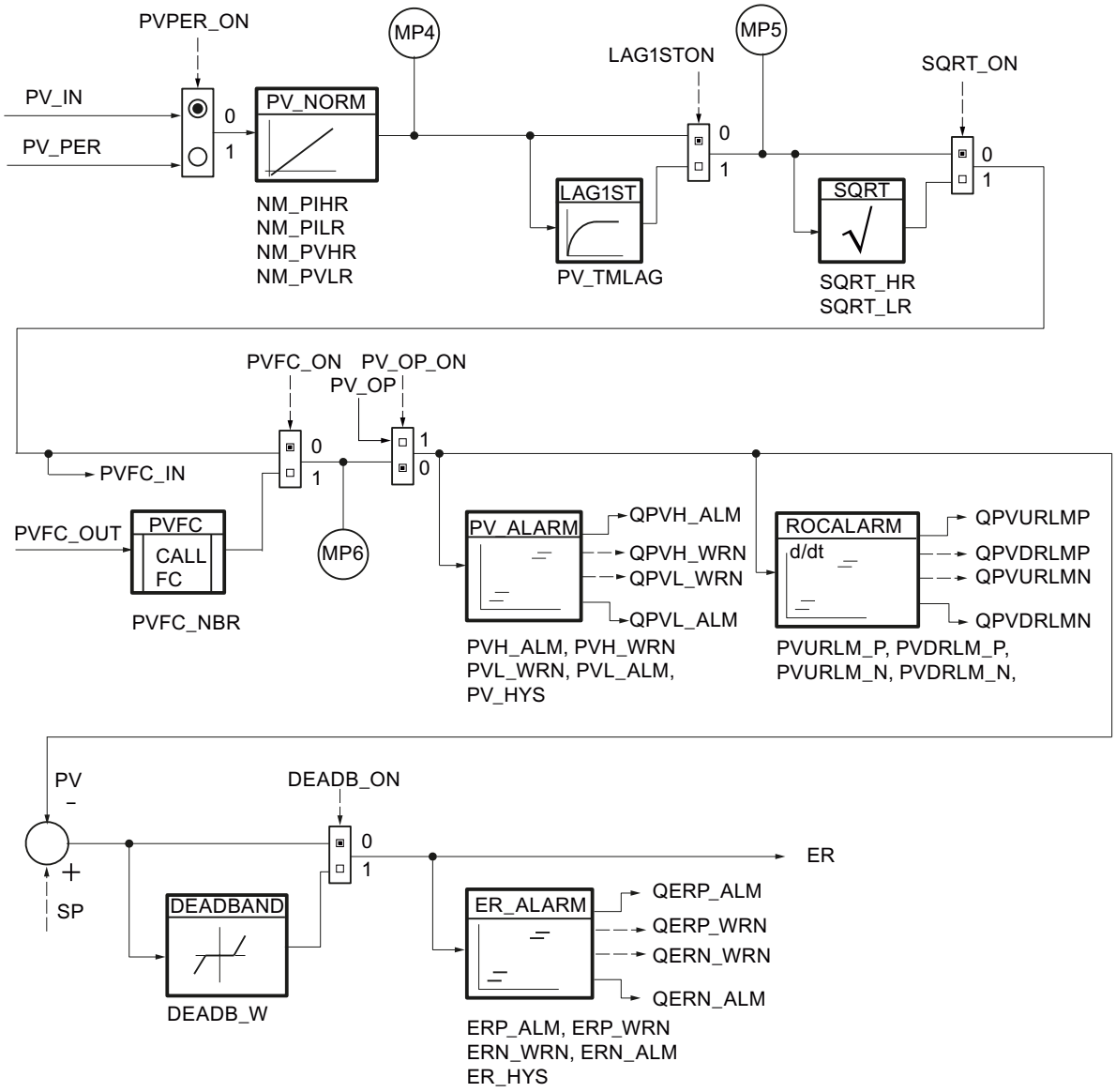

Regelalgorithmus

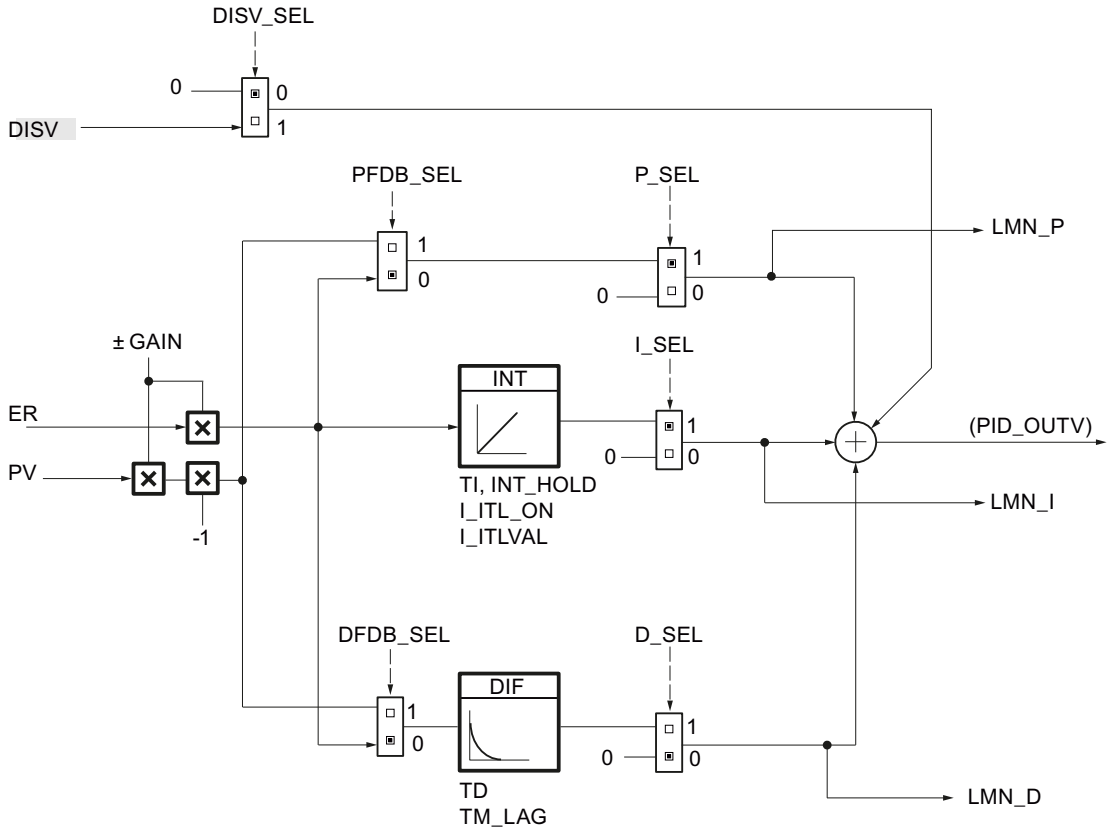

Verarbeitung des analogen Stellwerts

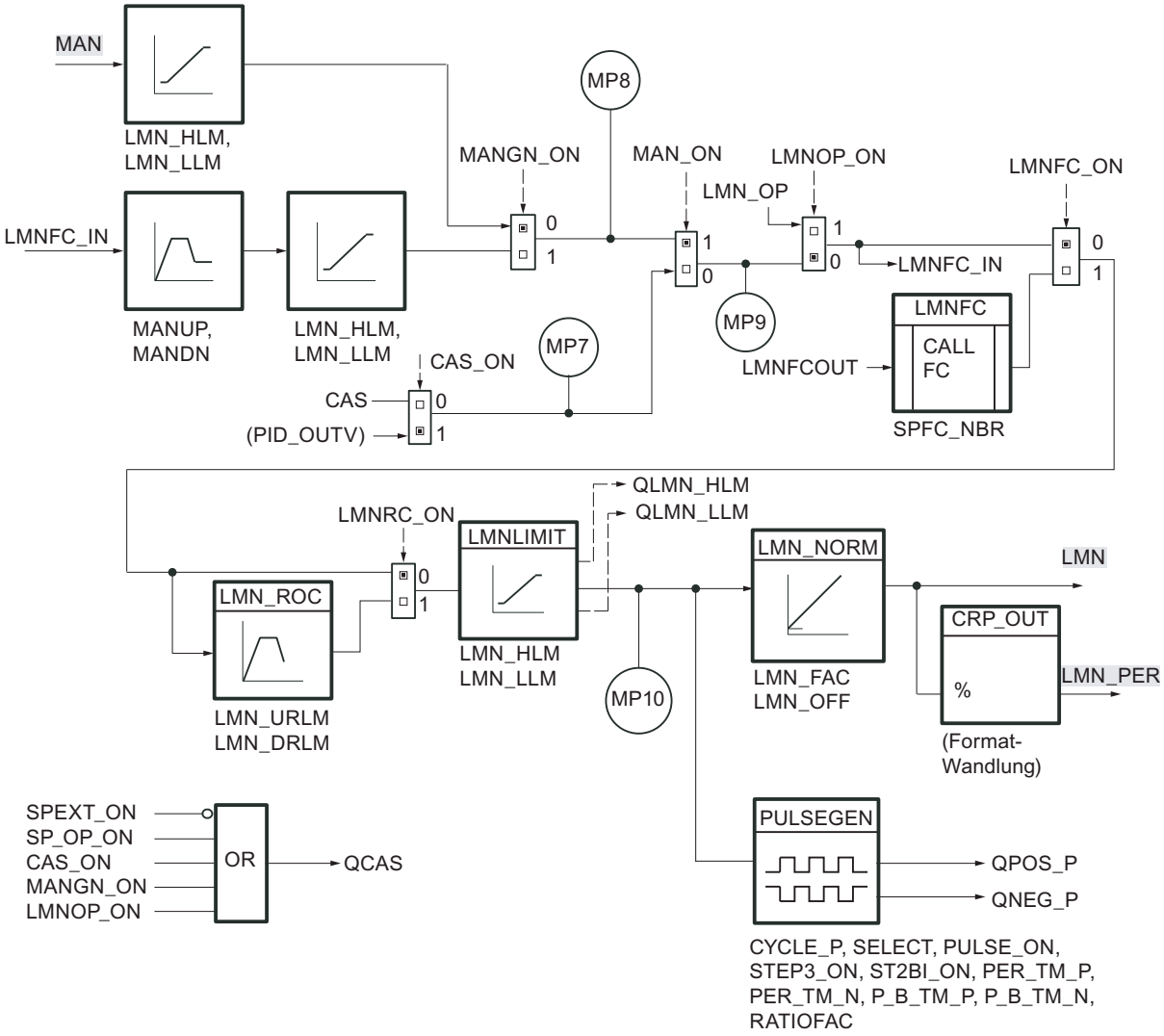

Das OR–Gatter erzeugt abhängig von der Kombination der Schaltzustände ein Freigabesignal für die Kaskadenkoppelung.

Globaler Datenbaustein DB_RMPSK

Hinweis

Globalen Datenbaustein selbst erstellen

Die Stützpunkte für den Zeitplangeber (RMP_SOAK) werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
NBR_PTS	INT	4	Anzahl der Stützpunkte -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Struktur für die Daten der Stützpunkte. Das Array muss von 0...NBR_PTS deklariert werden.
PI[0].OUTV	REAL	0.0	Ausgangswert [0] Startpunkt technischer Wertebereich
PI[0].TMV	TIME	T#1s	Zeitwert [0] Startpunkt
PI[1].OUTV	REAL	0.0	Ausgangswert [1] Stützpunkt 1 technischer Wertebereich
PI[1].TMV	TIME	T#1s	Zeitwert [1] Stützpunkt 1
PI[2].OUTV	REAL	0.0	Ausgangswert [2] Stützpunkt 2 technischer Wertebereich
PI[2].TMV	TIME	T#1s	Zeitwert [2] Stützpunkt 2
PI[3].OUTV	REAL	0.0	Ausgangswert [3] Stützpunkt 3 technischer Wertebereich
PI[3].TMV	TIME	T#1s	Zeitwert [3] Stützpunkt 3
PI[4].OUTV	REAL	0.0	Ausgangswert [4] Stützpunkt 4 technischer Wertebereich
PI[4].TMV	TIME	T#0s	Zeitwert [4] Stützpunkt 4 0 ms

Datenbaustein über eine externe Quelldatei generieren

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren.

Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den unten stehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen "DB":
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Externe Quellen".
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.

12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_RMPSK"
TITLE = data block ramp soak
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_RMPSK
   STRUCT
      NBR_PTS : Int;              // number of points (excluding
                                  starting point)
      PI : Array[0..4] of Struct
         OUTV : Real;             // output variable
         TMV : Time;              // output time value
      END_STRUCT;
   END_STRUCT;


BEGIN
   NBR_PTS := 4;
   PI[0].TMV := T#1s;
   PI[1].TMV := T#1s;
   PI[2].TMV := T#1s;
   PI[3].TMV := T#1s;
   PI[4].TMV := T#0s;
END_DATA_BLOCK
```

PID_ES

Beschreibung PID_ES

Neben den Funktionen im Soll- und Istwertzeig realisiert die Anweisung einen fertigen PID-Regler mit binärem Stellwert-Ausgang. Es ist möglich, den Stellwert von Hand zu beeinflussen. Teilfunktionen lassen sich zu- oder abschalten.

Die Anweisung ermöglicht es, technische Prozesse und Anlagen mit integrierenden Stellgliedern auf den Automatisierungssystemen SIMATIC S7 zu regeln. Es können sowohl träge Strecken (Temperaturen, Füllstände usw.) als auch sehr schnelle Strecken (Durchflüsse, Drehzahlen usw.) geregelt werden.

Der Regler kann als Festwertregler einzeln oder in unterlagerten Regelkreisen bei Kaskaden-, Mischungs- oder Verhältnisregelungen eingesetzt werden, jedoch nicht als Führungsregler.

Die Signalverarbeitung im Soll-/Istwertzweig sowie bei der Behandlung und Überwachung der Regeldifferenz ist identisch mit der des kontinuierlichen Reglers.

Anlauf

PID_ES verfügt über eine Initialisierungsroutine, die durchlaufen wird, wenn der Eingangs–Parameter COM_RST = TRUE gesetzt ist.

Der I-Anteil wird beim Anlauf auf den Initialisierungswert I_ITLVAL gesetzt und am Ausgang LMN_I ausgegeben. Beim Aufruf in einer Weckalarmebene arbeitet er von diesem Wert aus weiter.

Bei eingeschaltetem Zeitplangeber werden die Zeitabschnitte DB_NBR PI[0 ... NBR_PTS].TMV zwischen den Stützpunkten aufsummiert und am Ausgang die Gesamtzeit T_TM und Gesamtrestzeit RT_TM angezeigt. Bei Online–Änderungen von PI[n].TMV oder bei Vorgabe von TM_CONT und TM_SNBR ändern sich Gesamtzeit und Gesamtrestzeit der Fahrkurve. Da die Berechnung von T_TM und RS_TM bei vielen Stützpunkten die Bearbeitungszeit der Funktion RMP_SOAK stark vergrößert, wird diese Berechnung nur nach Neustart oder bei TUPDT_ON = TRUE durchgeführt.

Alle anderen Ausgänge werden auf ihre Vorbelegungswerte gesetzt.

Wiederanlauf

Bei Wiederanlauf wird von dem Betriebszustand ausgegangen, der zum Zeitpunkt der Unterbrechung geherrscht hat. Die Regelung arbeitet mit den Werten weiter, die im Augenblick der Unterbrechung berechnet waren.

Aufruf

Die Anweisung PID_ES muss äquidistant aufgerufen werden. Benutzen Sie hierzu eine Weckalarmebene (z.B. OB35).

Wenn Sie die Anweisung PID_ES als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID_ES direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

Regeldifferenzbildung PID_ES

Die folgenden Funktionen sind immer wirksam und können nicht abgeschaltet werden.

- Sollwertbegrenzung SP_LIMIT,
- Istwertüberwachung auf Absolutwerte PV_ALARM,
- Istwertüberwachung auf Größe der Änderungsgeschwindigkeit ROCALARM,
- Regeldifferenzüberwachung auf Absolutwerte ER_ALARM,
- Begrenzung des Ausgangssignals LMNLIMIT.

Die Vorbelegungswerte sind so gewählt, dass der Betrieb ohne eine individuelle Anpassung auskommt. Um gute Regelergebnisse zu erzielen, müssen Sie die Parameter dieser Funktionen immer an ihren Prozess anpassen.

Istwert aufbereiten

Der Istwert wird auf die physikalische Größe normiert.

AUTOHOTSPOT

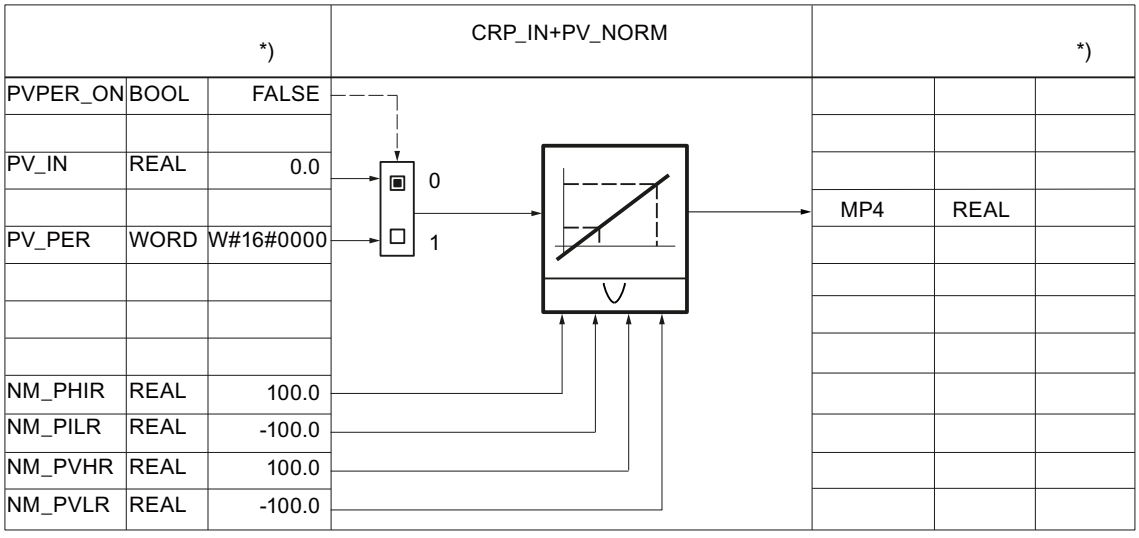

*) Vorbelegung bei Neuerstellung des Technologieobjekts.

Der Istwert kann mit einem Verzögerungsglied 1. Ordnung geglättet werden.

AUTOHOTSPOT

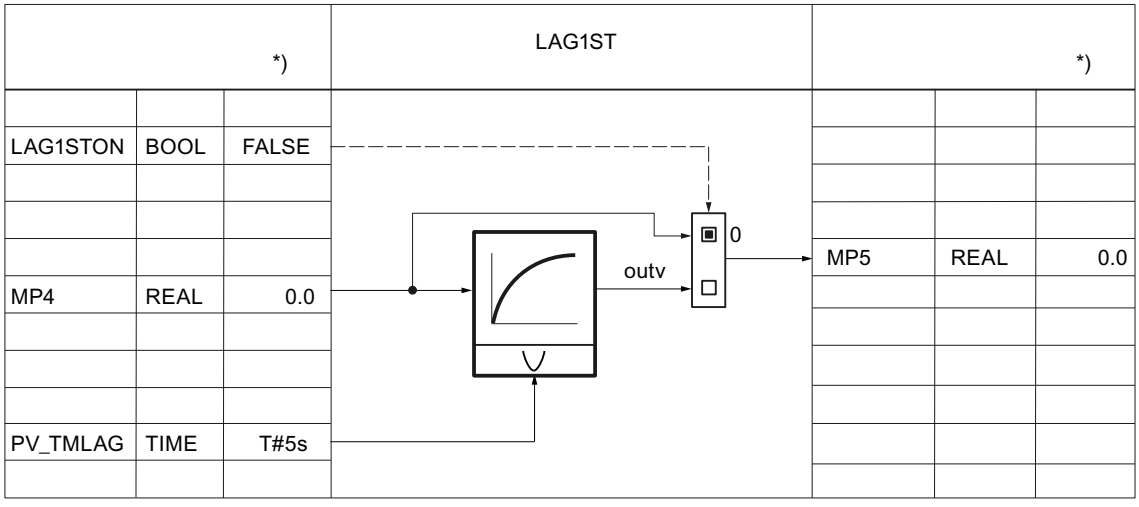

Der Istwert kann mit der Quadratwurzel linearisiert werden. Dazu sind folgende Parameter notwendig:

AUTOHOTSPOT

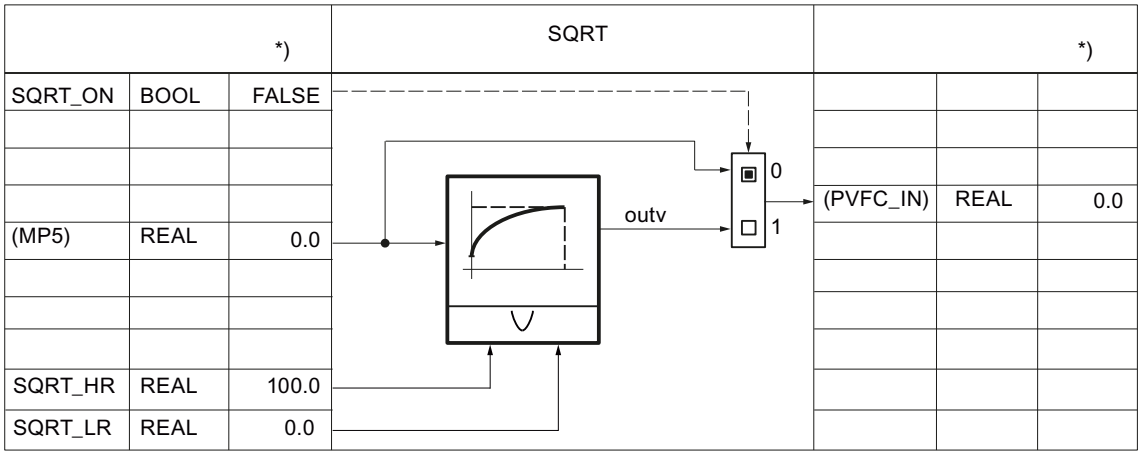

Der Istwert kann mit einer anwenderspezifischen Funktion aufbereitet werden. Dazu sind die folgenden Parameter notwendig. Die Verschaltung muss in der Anwender-FC programmiert werden.

AUTOHOTSPOT

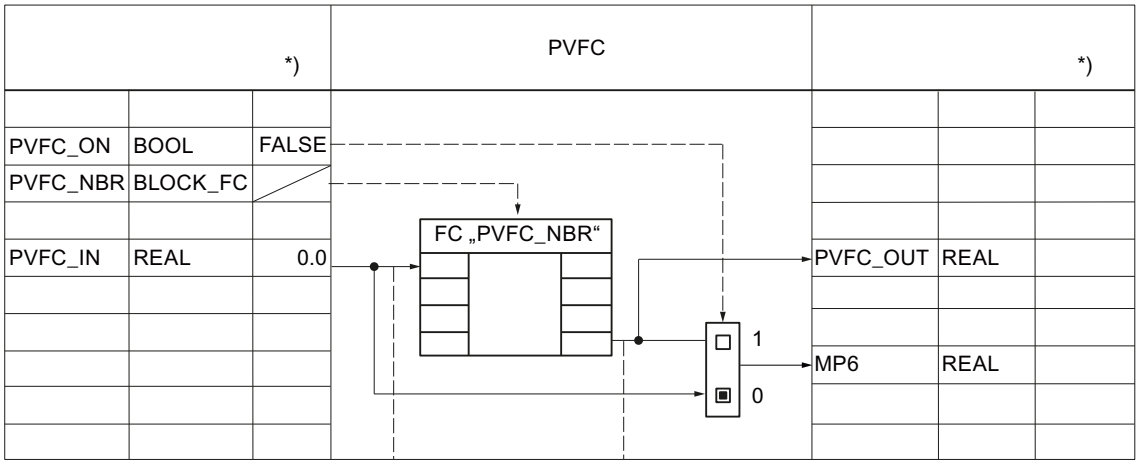

Der Istwert wird immer mit den folgenden Parametern auf Warn- und Alarmgrenzen überwacht.

AUTOHOTSPOT

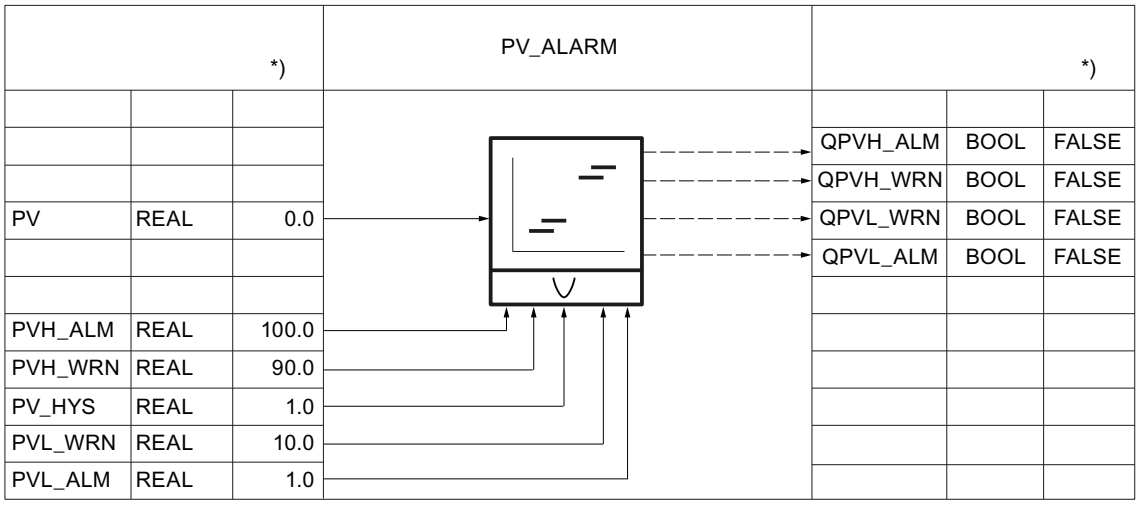

Die Änderungsgeschwindigkeit des Istwerts wird immer mit den folgenden Parametern begrenzt.

AUTOHOTSPOT

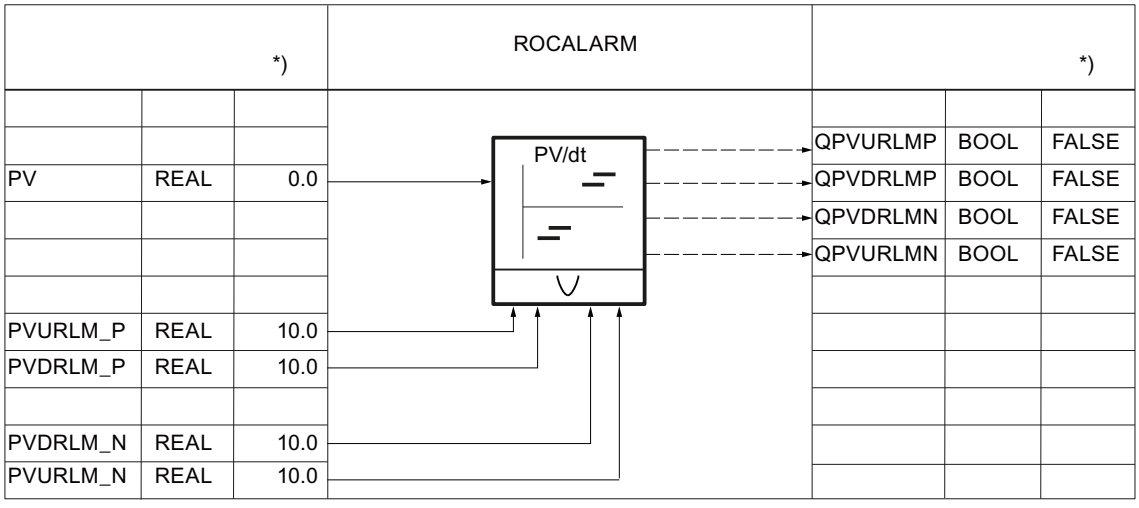

Sollwert aufbereiten

Der interne Sollwert SP_IN kann mit den folgenden Parametern vergrößert oder verkleinert werden.

AUTOHOTSPOT

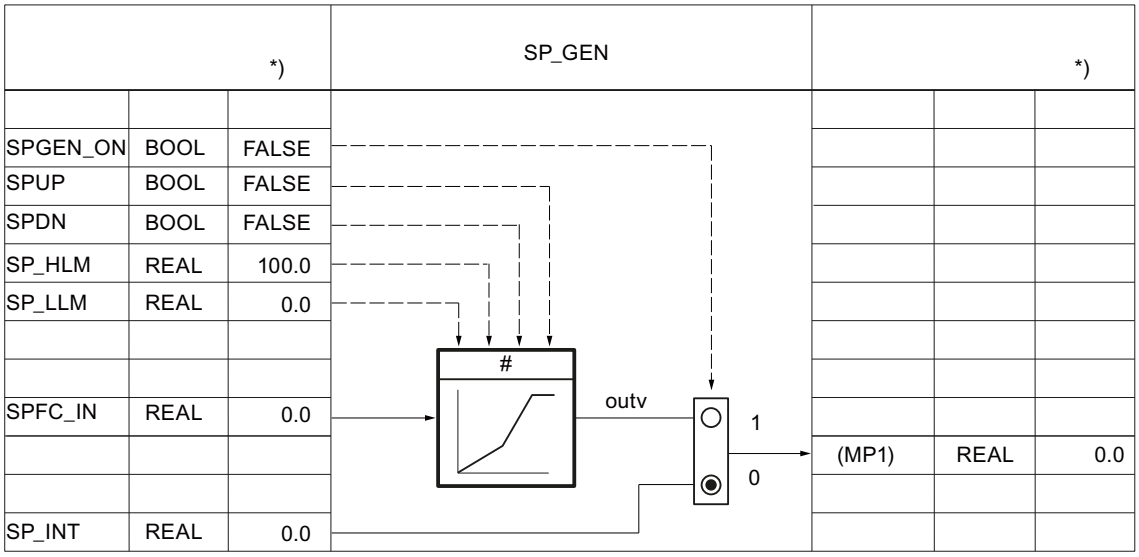

Der Sollwert kann mit den folgenden Parametern über einen Zeitplangeber vorgegeben werden.

AUTOHOTSPOT

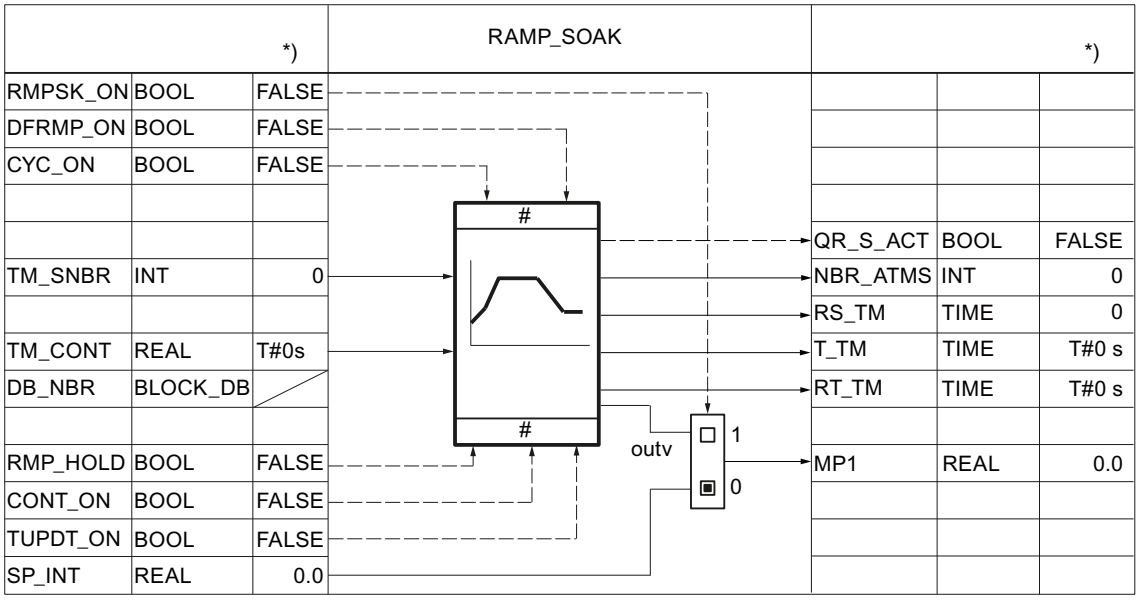

Für Mischungs-/Verhältnisregler kann der externe Sollwert SP_EXT mit den folgenden Parametern normiert werden.

AUTOHOTSPOT

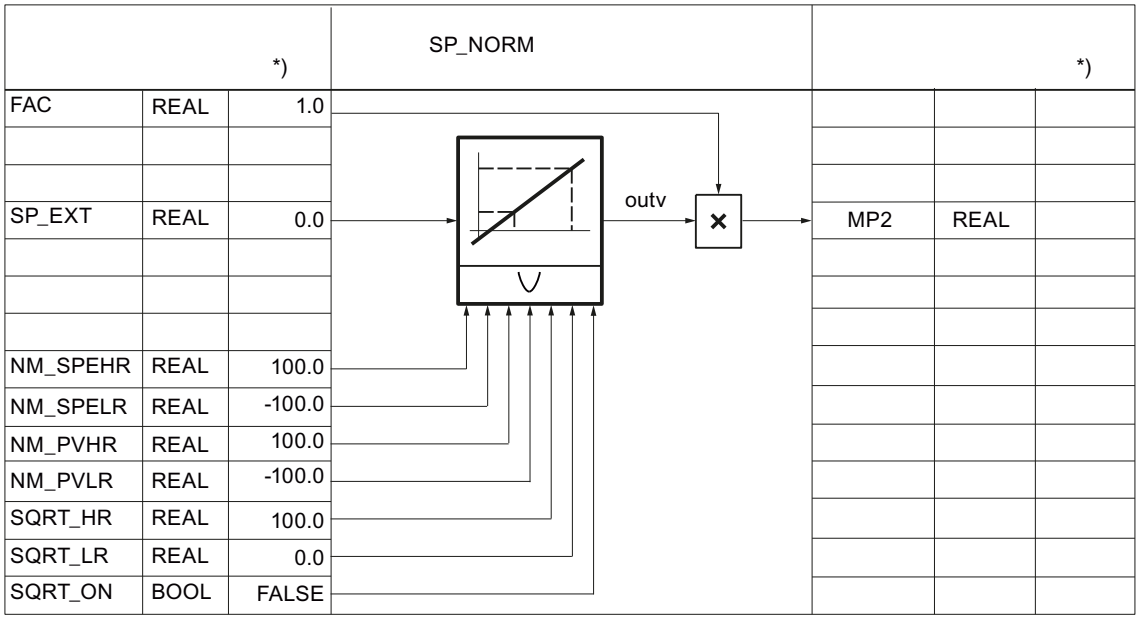

Der Sollwert kann mit einer anwenderspezifischen Funktion weiter aufbereitet werden. Dazu sind die folgenden Parameter notwendig. Die Verschaltung muss in der Anwender-FC programmiert werden.

Der Eingangswert SPFC_IN ist ein impliziter Parameter. Dieser ist am Konfigurationswerkzeug entweder über den Messpunkt MP1 (SPEXT_ON = FALSE) oder über den Messpunkt MP2 (SPEXT_ON = TRUE) zu beobachten. Der Ausgangswert ist am Messpunkt MP3 zugänglich. Der Eingang SPFC_IN ist dann auf den Sollwertzweig durchgeschaltet, wenn SPFC_ON =FALSE ist.

AUTOHOTSPOT

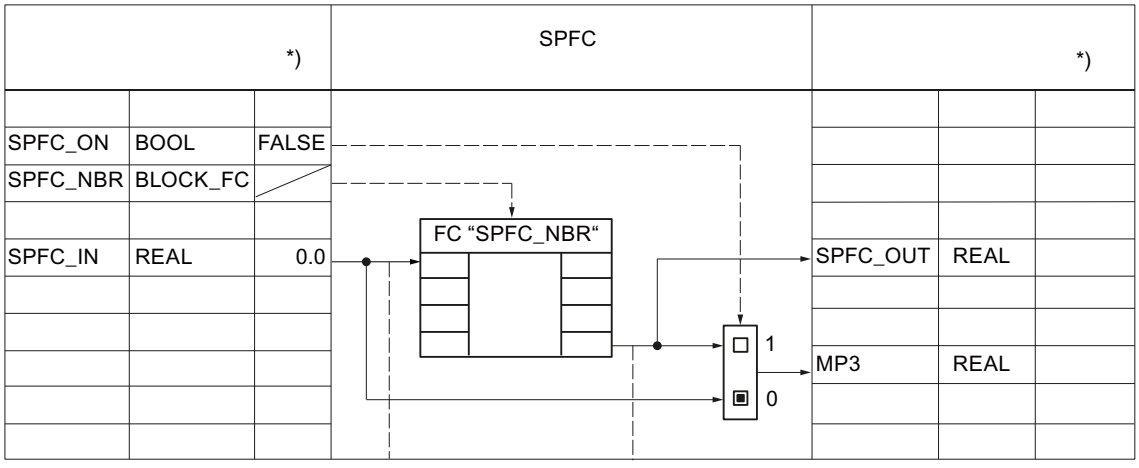

Die Änderungsgeschwindigkeit des Sollwerts kann mit den folgenden Parametern begrenzt werden.

		*)	SP_ROC			*)
SPROC_ON	BOOL	FALSE				
inv	REAL			outv	REAL	
SPURLM_P	REAL	10.0				
SPDRLM_P	REAL	10.0				
SPURLM_N	REAL	10.0				
SPDRLM_N	REAL	10.0				

Der Sollwert wird immer mit den folgenden Parametern auf eine absolute Ober- und Untergrenze begrenzt.

		*)	SP_LIMIT			*)
inv	REAL			QSP_HLM	BOOL	FALSE
				SP	REAL	0.0
				QSP_LLM	BOOL	FALSE
SP_HLM	REAL	100.0				
SP_LLM	REAL	0.0				

Regeldifferenz bilden

Die Regeldifferenz kann mit den folgenden Parametern über eine Totzone geleitet werden.

AUTOHOTSPOT

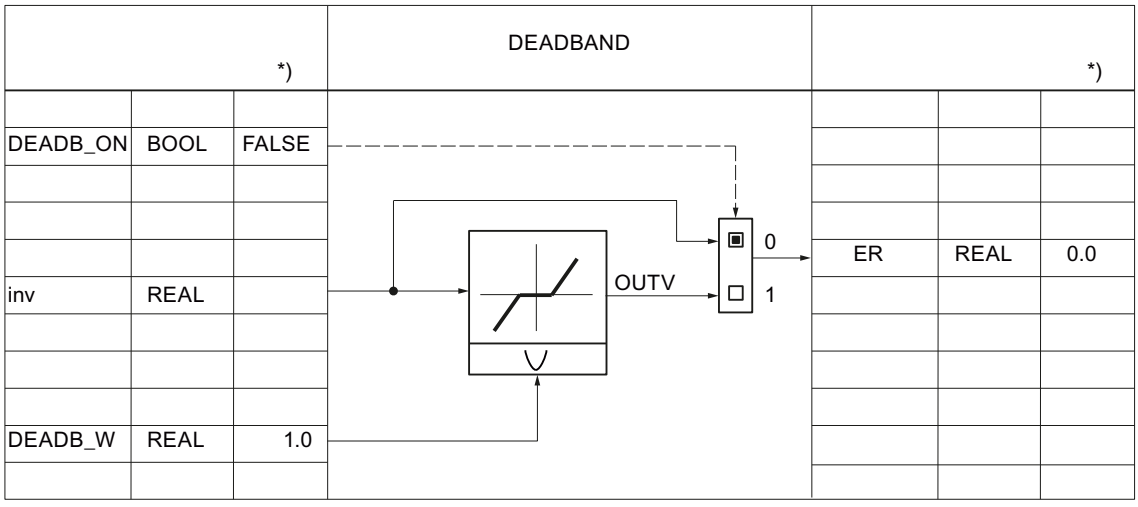

Die Regeldifferenz wird immer mit den folgenden Parametern auf Warn- und Alarmgrenzen überwacht.

AUTOHOTSPOT

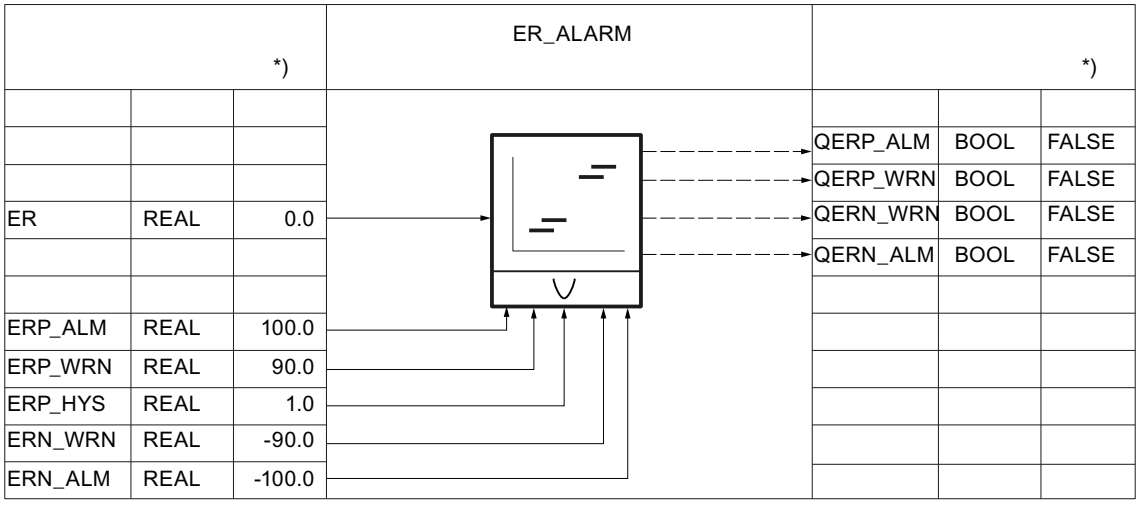

Regelalgorithmus PID_ES

Regelalgorithmus

AUTOHOTSPOT

Um eine wirksame Reglerstruktur festzulegen, stehen insgesamt fünf Schalter zur Verfügung.

AUTOHOTSPOT

Reglerstruktur	Schalter				
	P_SEL	I_SEL	D_SEL	PFDB_SEL	DFDB_SEL
P–Regelung	TRUE	FALSE	FALSE	FALSE	FALSE
P–Regelung (P in Rückführung)	TRUE	FALSE	FALSE	TRUE	FALSE
PI–Regelung	TRUE	TRUE	FALSE	FALSE	FALSE
PI–Regelung (P in Rückführung)	TRUE	TRUE	FALSE	TRUE	FALSE
PD–Regelung	TRUE	FALSE	TRUE	FALSE	FALSE
PD–Regelung (P in Rückführung)	TRUE	FALSE	TRUE	FALSE	TRUE
PID–Regelung	TRUE	TRUE	TRUE	FALSE	FALSE
PID–Regelung (P/D in Rückführung)	TRUE	TRUE	TRUE	FALSE	TRUE

Der P-Anteil wird durch die Proportionalverstärkung GAIN bestimmt und kann an LMN_P beobachtet werden.

AUTOHOTSPOT

Der I-Anteil wird mit den folgenden Parametern berechnet.

AUTOHOTSPOT

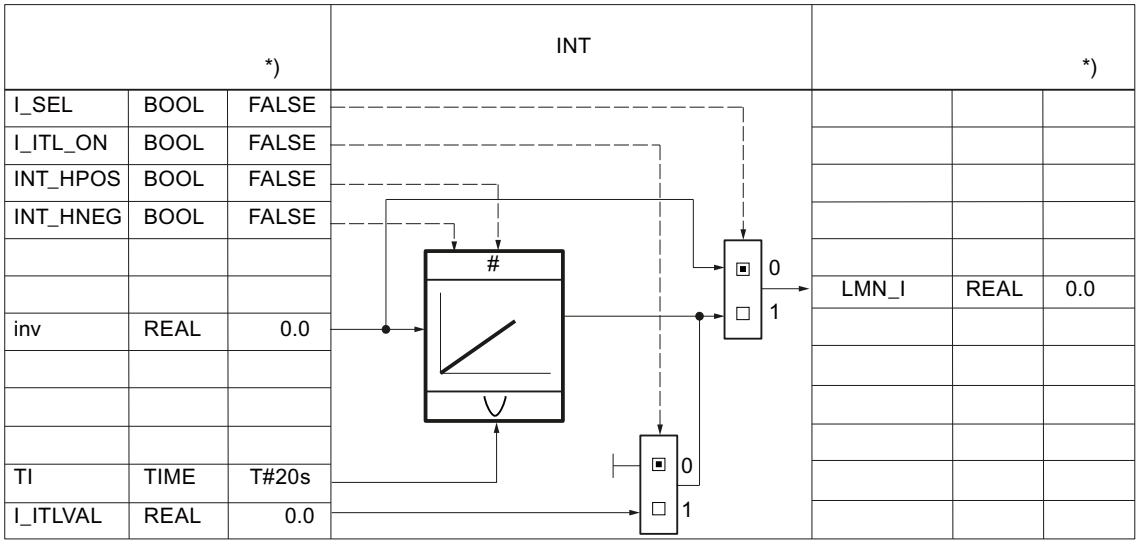

*) Vorbelegung bei Neuerstellung des Technologieobjekts

Der D-Anteil wird mit den folgenden Parametern berechnet.

AUTOHOTSPOT

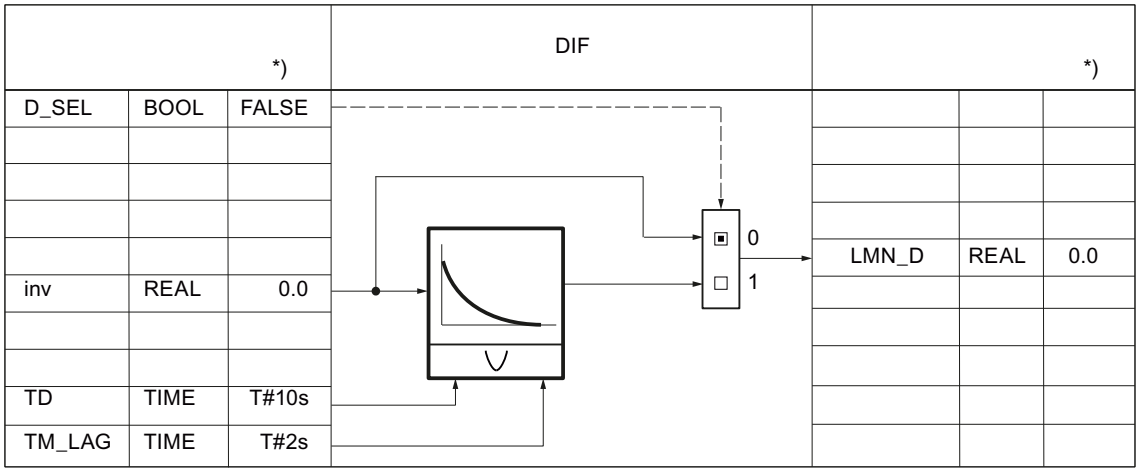

Stellwert PID_ES

Normierung der Stellungsrückmeldung

Die Stellungsrückmeldung im Peripherieformat wird in das Gleitpunktformat umgewandelt. Die Stellungsrückmeldung im Peripherieformat kann an MP10 beobachtet werden.

Folgende Tabelle gibt einen Überblick über Bereiche und Zahlenwerte vor und nach der Bearbeitung.

Peripheriewert LMNR_PER	LMNR in %
32767	118.515
27648	100.000
1	0.003617
0	0.000
-1	-0.003617
-27648	-100.000
-32768	-118.519

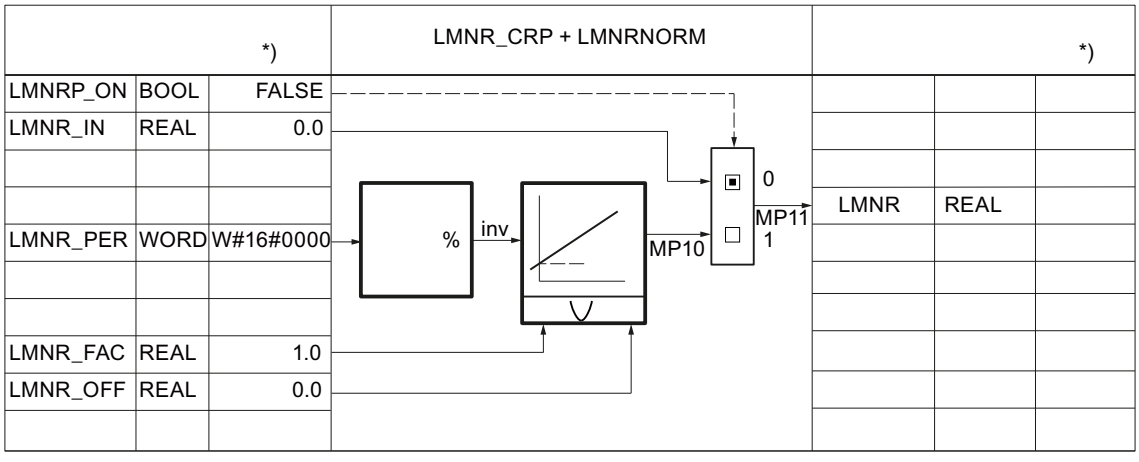

*) Vorbelegung bei Neuerstellung des Technologieobjekts.

Begrenzung des Stellwerts

Der Stellwert wird immer mit den folgenden Parametern auf eine absolute Ober- und Untergrenze begrenzt. Der implizite Parameter kann an MP9 beobachtet werden.

AUTOHOTSPOT

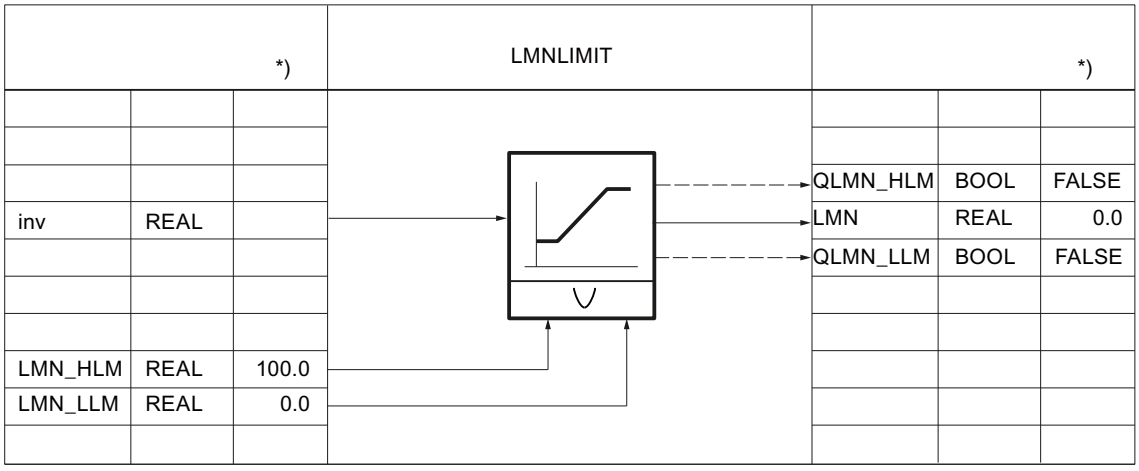

Schrittregler mit Stellungsrückmeldung

Die Stellsignale werden beim Schrittregler mit Stellungsrückmeldung mit den folgenden Parametern gebildet:

AUTOHOTSPOT

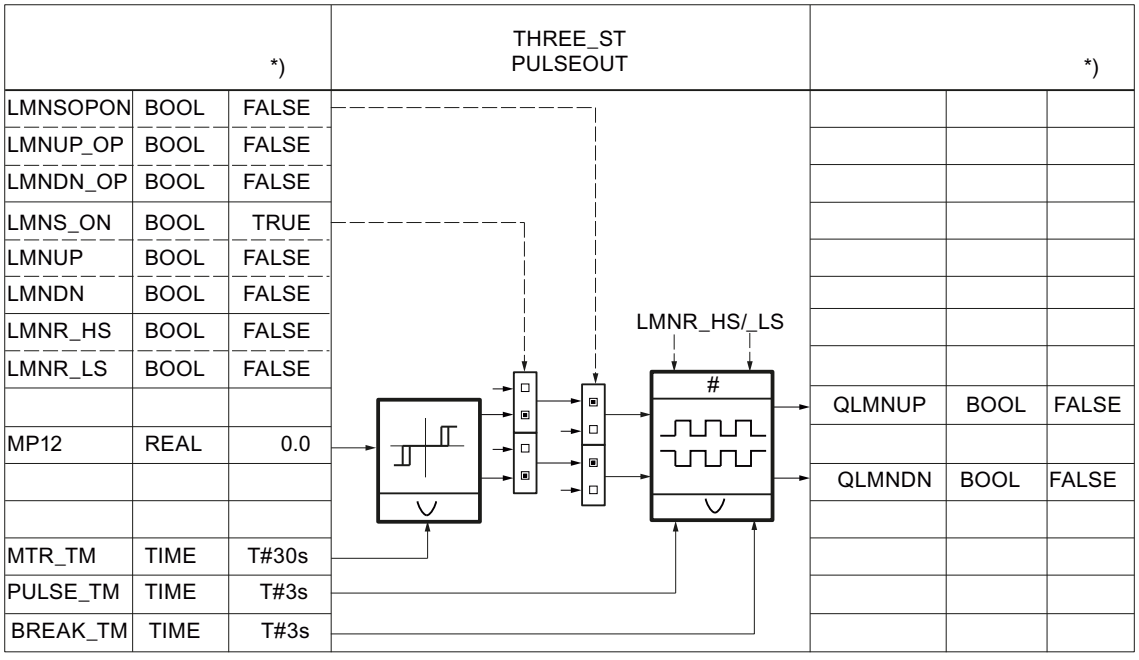

Schrittregler ohne Stellungsrückmeldung

Die Stellungsrückmeldung kann simuliert werden. Die Nachbildung der Stellungsrückmeldung geschieht durch einen Integrator mit der Motorstellzeit MTR_TM als Integrationszeitkonstante. Im Zustand LMNRS_ON = FALSE wird am Ausgang des Integrators LMNR_ SIM der Startwert des Parameters LMNRSVAL ausgegeben. Nach dem Umschalten auf LMNRS_ON = TRUE läuft die Simulation mit diesem Startwert an.

Ist LMNR_HS = TRUE, wird die Integration nach oben angehalten, ist LMNR_LS = TRUE, wird sie nach unten angehalten. Ein Abgleich der simulierten Stellungsrückmeldung an den Anschlagsignalen erfolgt nicht.

AUTOHOTSPOT

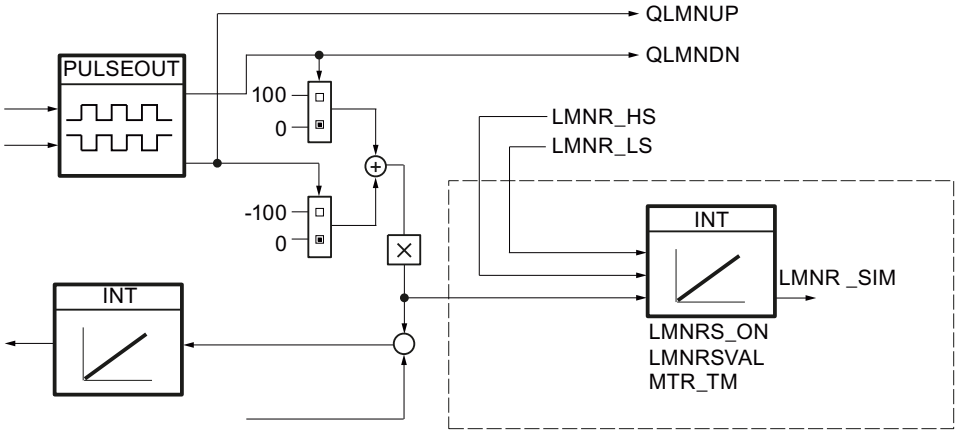

Hinweis

Die Stellungsrückmeldung wird lediglich simuliert. Sie muss nicht mit der wirklichen Position des Stellgliedes übereinstimmen. Ist eine reale Stellungsrückmeldung vorhanden, dann sollte diese in jedem Falle genutzt werden.

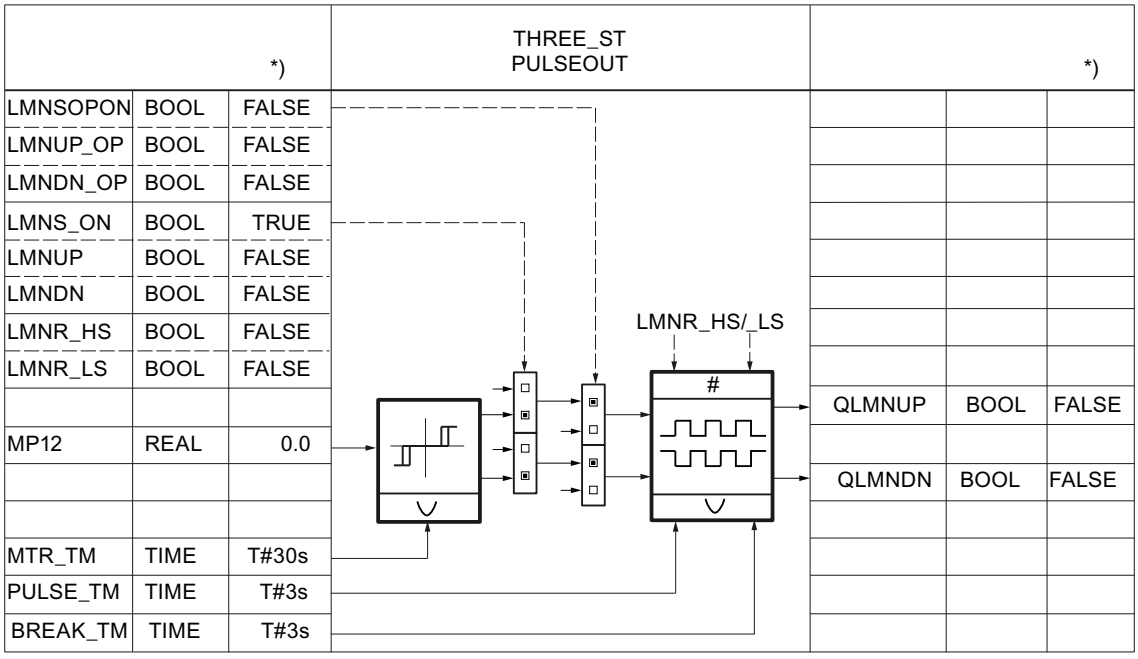

PID_ES in Kaskadenschaltungen

Auftrennung der Reglerkaskade

In einer Kaskade stehen mehrere Regler direkt miteinander in Beziehung. Sie müssen daher Vorkehrungen treffen, damit nach Auftrennung der Kaskadenstruktur an einer beliebigen Stelle der Kaskadenbetrieb problemlos wieder aufgenommen werden kann.

In Kaskadenregelungen wird deshalb in den unterlagerten Reglern über eine Oder–Logik aus den Zustandssignalen der Schalter im Sollwertzweig und Stellgrößenzweig ein Signal QCAS gebildet. Dieses Signal betätigt in den überlagerten Reglern einen Schalter, der die Regler in den Nachführbetrieb überführt. Die Nachführgröße ist dabei immer der Sollwert SP des unterlagerten Kreises.

Die Umschaltung vom Nachführbetrieb in den Automatikbetrieb erfolgt stoßfrei wie bei der Hand– Automatikumschaltung.

Hinweis

PID_ES kann in Kaskadenschaltungen nur als Folgeregler in unterlagerten Regelkreisen eingesetzt werden.

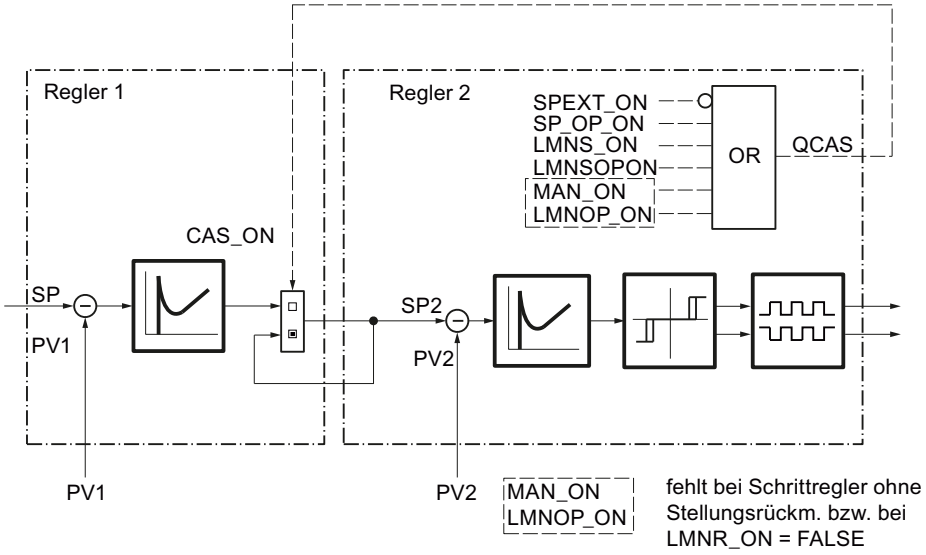
Regler 1
Regler 2
SPEXT_ON
SP_OP_ON
LMNS_ON
LMNSOPON
MAN_ON
LMNOP_ON
OR
QCAS
CAS_ON
SP
PV1
SP2
PV2
PV1
PV2
MAN_ON
LMNOP_ON
fehlt bei Schrittregler ohne
Stellungsrückm. bzw. bei
LMNR_ON = FALSE

Bausteinverschaltung

Das folgende Bild zeigt das Prinzip der Reglerverschaltung in mehrschleifigen Kaskaden.

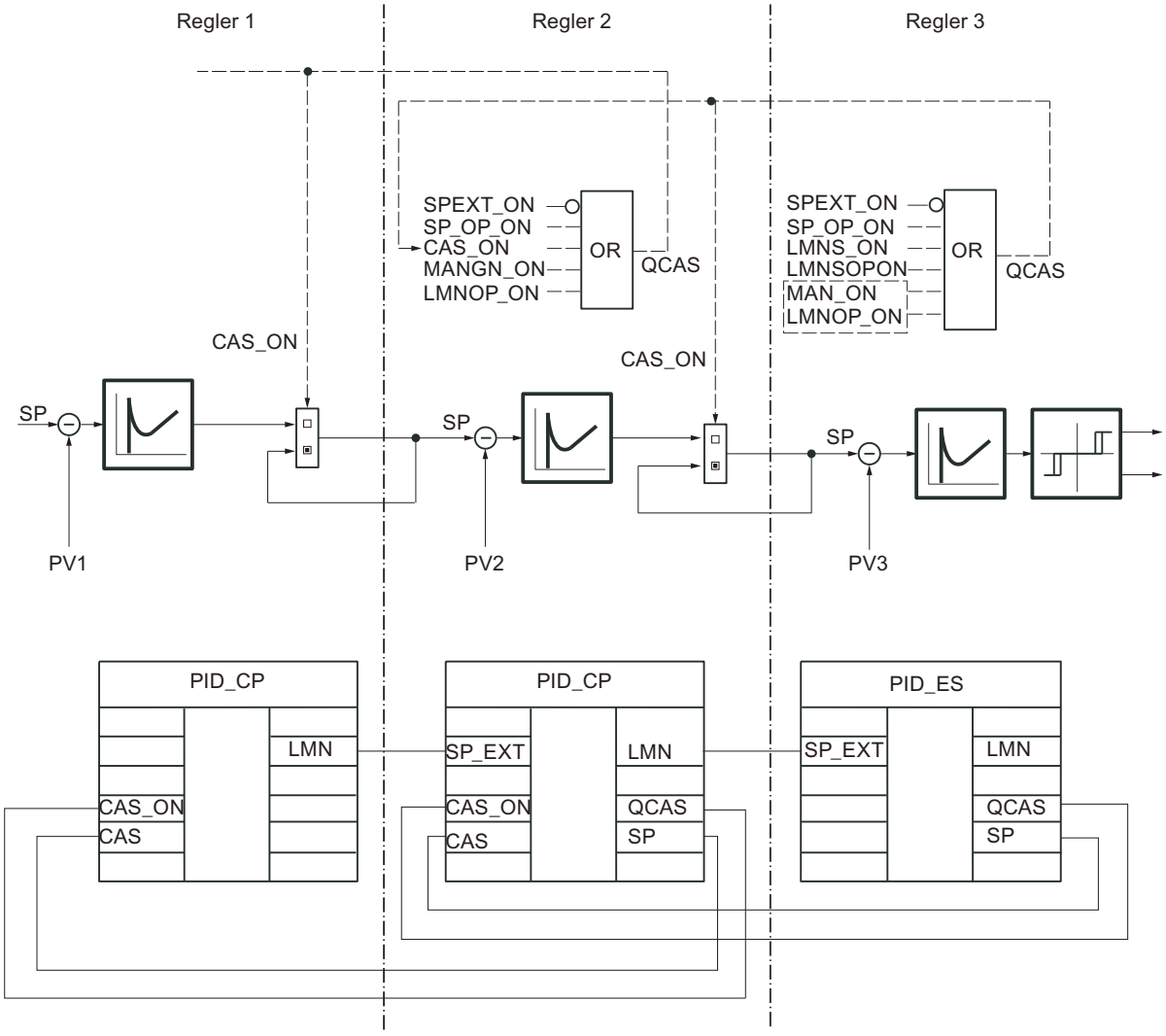

Eingangsparameter PID_ES

Tabelle 4-221

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL		FALSE	Neustart (Initialisierungsroutine der Anweisung wird bearbeitet)
I_SEL	0.1	BOOL		TRUE	I–Anteil einschalten
D_SEL	0.2	BOOL		FALSE	D–Anteil einschalten

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
MAN_ON	0.3	BOOL		TRUE	Handbetrieb einschalten (Regelkreis aufgetrennt, LMN von Hand eingestellt)
LMNR_HS	0.4	BOOL		FALSE	Oberes Anschlagsignal der Stellungsrückmeldung
LMNR_LS	0.5	BOOL		FALSE	Unteres Anschlagsignal der Stellungsrückmeldung
CYCLE	2.0	TIME	> 20 ms (S7–300)	T#1s	Abtastzeit (Zeit zwischen Bausteinaufrufen = konstant) Achten Sie unbedingt darauf, dass Sie diesen Parameter mit der Weckalarmtaktzeit des OB, in dem die Anweisung PID_CP läuft, parametrieren! Andernfalls arbeiten die zeitabhängigen Funktionen nicht korrekt. (Ausnahme: Sie verwenden eine Zeituntersetzung, z. B. über den Regleraufrufverteiler.)
SP_INT	6.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	interner Sollwert SP_INT kann mit SPUP und SPDN über ein HMI verändert werden
SP_EXT	10.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	externer Sollwert SP_EXT wird für Mischungs- und Verhältnisregler sowie Kaskadenregler verwendet und kann normiert werden.
PV_IN	14.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	Istwert–Eingang (PV im Gleitpunktformat)
PV_PER	18.0	INT		W#16#0000	Istwert von Peripherie
GAIN	20.0	REAL	gesamter Wertebereich (dimensionslos)	2.0	Proportionalbeiwert (= Reglerverstärkung)
TI	24.0	TIME	TI ≥ CYCLE	T#20s	Integrationszeit
TD	28.0	TIME	TD ≥ CYCLE	T#10s	Differenzierzeit
TM_LAG	32.0	TIME	TM_LAG ≥ CYCLE/2	T#2s	Verzögerungszeit des D–Anteils
DISV	36.0	REAL	-100.0 ... 100.0 [%]	0.0	Störgröße
LMNR_IN	40.0	REAL	0.0 ... 100.0 [%]	0.0	Stellungsrückmeldung (LMNR im Gleitpunktformat)
LMNR_PER	44.0	INT		W#16#0000	Stellungsrückmeldung von Peripherie (LMNR im Peripherieformat)
SP_HLM	46.0	REAL	technischer Wertebereich (physikalische Größe)	100.0	Sollwert obere Begrenzung
SP_LLM	50.0	REAL	technischer Wertebereich (physikalische Größe)	0.0	Sollwert untere Begrenzung

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
LMN_HLM	54.0	REAL	LMN_LLM ... 100.0 [%]	100.0	Stellwert: obere Begrenzung
LMN_LLM	58.0	REAL	0.0 ... LMN_HLM [%]	0.0	Stellwert: untere Begrenzung
DB_NBR	62.0	BLOCK_DB		DB 1	Datenbausteinnummer (DB mit den Stützpunkten der Fahrkurve)
SPFC_NBR	64.0	BLOCK_FC		FC 0	Sollwert FC–Nummer (selbstdefinierter FC im Sollwertzweig)
PVFC_NBR	66.0	BLOCK_FC		FC 0	Istwert FC–Nummer (selbstdefinierter FC im Istwertzweig)

Ausgangsparameter PID_ES

Tabelle 4-222

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
QLMNUP	68.0	BOOL	FALSE	Stellwertsignal hoch
QLMNDN	68.1	BOOL	FALSE	Stellwertsignal tief
QCAS	68.2	BOOL	FALSE	Signal für Kaskadebetrieb (wird mit CAS_ON des überlagerten Reglers verschaltet)
LMN	70.0	REAL	0.0	Stellwertsignal (nach Regelalgorithmus)
SP	74.0	REAL	0.0	Sollwert (effektiv wirksamer Sollwert)
PV	78.0	REAL	0.0	Istwert (Ausgabe des effektiven Istwertes bei Kaskadebetrieb)

Durchgangsparameter PID_ES

Tabelle 4-223

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
MAN	82.0	REAL	0.0	Hand–Stellwert (zur Stellwertvorgabe mittels B&B–Funktionen)

Statische Variablen PID_ES

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
PVH_ALM	86.0	REAL	PVH_WRN...100.0	100.0	Istwert: oberer Grenzwert 'Alarm' Zulässiger Wertebereich:
PVH_WRN	90.0	REAL	PVL_WRN... PVH_ALM	90.0	Istwert: oberer Grenzwert 'Warnung' Zulässiger Wertebereich:

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
PVL_WRN	94.0	REAL	PVL_ALM... PVH_WRN	-90.0	Istwert: unterer Grenzwert 'Warnung' Zulässiger Wertebereich:
PVL_ALM	98.0	REAL	-100.0...PVL_WRN	-100.0	Istwert: unterer Grenzwert 'Alarm' Zulässiger Wertebereich:
SPGEN_ON	102.0	BOOL		FALSE	Sollwertbedienung einschalten (Verstellen des Sollwerts über Auf–/Ab–Schalter)
SPUP	102.1	BOOL		FALSE	Sollwert hoch
SPDN	102.2	BOOL		FALSE	Sollwert tief
RMPSK_ON	102.3	BOOL		FALSE	Zeitplangeber einschalten (Sollwert wird als Fahrkurve vorgegeben)
SPEXT_ON	102.4	BOOL		FALSE	Externen Sollwert einschalten (zur Verschaltung mit anderen Reglerbausteinen)
MANGN_ON	102.5	BOOL		FALSE	Handbedienung einschalten (LMN per Schalter eingestellt)
MANUP	102.6	BOOL		FALSE	Hand–Stellwert Hoch
MANDN	102.7	BOOL		FALSE	Hand–Stellwert Tief
LMNS_ON	103.0	BOOL		FALSE	Handbetrieb der Stellsignale einschalten
LMNUP	103.1	BOOL		FALSE	Stellwertsignal Hoch (das Ausgangssignal QLMNUP wird bedient)
LMNDN	103.2	BOOL		FALSE	Stellwertsignal Tief (das Ausgangssignal QLMNDN wird bedient)
DFRMP_ON	103.3	BOOL		FALSE	Ausgang des Zeitplangebers vorbelegen (der Ausgang wird mit SP_INT vorbelegt)
CYC_ON	103.4	BOOL		FALSE	Zyklische Wiederholung einschalten (Fahrkurve wird automatisch wiederholt)
RMP_HOLD	103.5	BOOL		FALSE	Zeitplangeber anhalten (Ausgang des Zeitplangebers wird eingefroren)
CONT_ON	103.6	BOOL		FALSE	(Zeitplan) weitermachen (Fahrkurve wird an nächster Stützstelle fortgesetzt)
TUPDT_ON	103.7	BOOL		FALSE	Gesamtzeit neu berechnen (Gesamtzeit der Fahrkurve wird neu berechnet)
SPFC_ON	104.0	BOOL		FALSE	Sollwert–FC aufrufen
SPROC_ON	104.1	BOOL		FALSE	Rampenfunktion einschalten (Begrenzung der Änderungsgeschwindigkeit von SP)
PVPER_ON	104.2	BOOL		FALSE	Istwert von Peripherie einschalten (Verschaltung mit Peripheriebaugruppen)
LAG1STON	104.3	BOOL		FALSE	Verzögerungsglied 1. Ordnung einschalten
SQRT_ON	104.4	BOOL		FALSE	Quadratwurzelfunktion einschalten

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
PVFC_ON	104.5	BOOL		FALSE	Istwert–FC aufrufen
DEADB_ON	104.6	BOOL		FALSE	Totzone einschalten (Kleine Störungen und Rauschen werden gefiltert)
P_SEL	104.7	BOOL		TRUE	P–Anteil einschalten
PFDB_SEL	105.0	BOOL		FALSE	P–Anteil in Rückführung schalten
INT_HPOS	105.1	BOOL		FALSE	I–Anteil in positiver Richtung einfrieren
INT_HNEG	105.2	BOOL		FALSE	I–Anteil in negativer Richtung einfrieren
I_ITL_ON	105.3	BOOL		FALSE	I–Anteil vorbesetzen
DFDB_SEL	105.4	BOOL		FALSE	D–Anteil in Rückführung schalten
DISV_SEL	105.5	BOOL		FALSE	Störgröße aufschalten
LMNR_ON	105.6	BOOL		FALSE	Stellungsrückmeldung einschalten (Betriebsarten: Schrittregler mit/ohne Stellungsrückmeldung) Nicht im Regelbetrieb umschalten!
LMNRP_ON	105.7	BOOL		FALSE	Stellungsrückmeldung von Peripherie einschalten
TM_SNBR	106.0	INT	≥ 0 (dimensionslos)	0	Nr. des nächsten Stützpunktes zur Fortsetzung der Fahrkurve
TM_CONT	108.0	TIME	gesamter Wertebereich (dimensionslos)	T#0s	Zeitspanne bis zur Fortsetzung der Fahrkurve (Zeitspanne, die abläuft, bevor der Zeitplangeber nach einer Unterbrechung die Fahrkurve am Stützpunkt TM_SNBR weiter bearbeitet)
FAC	112.0	REAL	gesamter Wertebereich (dimensionslos)	1.0	Faktor (Verhältnisfaktor oder Mischungsfaktor)
NM_SPEHR	116.0	REAL		100.0	Sollwert–Normierung: Eingang oben
NM_SPELR	120.0	REAL		-100.0	Sollwert–Normierung: Eingang unten
SPFC_OUT	124.0	REAL	-100.0 ... 100.0	0.0	Sollwert FC–Ausgang (wird mit dem Ausgang des FC im Sollwertzweig verschaltet)
SPURLM_P	128.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Anstiegsbegrenzung im positiven Bereich
SPDRLM_P	132.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Abstiegsbegrenzung im positiven Bereich
SPURLM_N	136.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Anstiegsbegrenzung im negativen Bereich
SPDRLM_N	140.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Sollwert–Abstiegsbegrenzung im negativen Bereich
NM_PIHR	144.0	REAL		100.0	Istwert–Normierung: Eingang oben

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
NM_PILR	148.0	REAL		-100.0	Istwert–Normierung: Eingang unten
NM_PVHR	152.0	REAL		100.0	Istwert–Normierung: Ausgang oben
NM_PVLR	156.0	REAL		-100.0	Istwert–Normierung: Ausgang unten
PV_TMLAG	160.0	TIME	gesamter Wertebereich	T#5s	Istwertverzögerungszeit (Verzögerung des PT1–Gliedes im PV–Zweig)
SQRT_HR	164.0	REAL		100.0	Quadratwurzel: Messbereich Ausgang oben
SQRT_LR	168.0	REAL		0.0	Quadratwurzel: Messbereich Ausgang unten
PVFC_OUT	172.0	REAL	-100.0 ... 100.0	0.0	Istwert FC–Ausgang (wird mit dem Ausgang des FC im Istwertzweig verschaltet)
PVURLM_P	176.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Anstiegsbegrenzung im positiven Bereich
PVDRLM_P	180.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Abstiegsbegrenzung im positiven Bereich
PVURLM_N	184.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Anstiegsbegrenzung im negativen Bereich
PVDRLM_N	188.0	REAL	≥ 0 [physikalische Größe/s]	10.0	Istwert–Abstiegsbegrenzung im negativen Bereich
PV_HYS	192.0	REAL	≥ 0	1.0	Istwert Hysterese (Vermeidet 'Flackern' der Überwachungsanzeige)
DEADB_W	196.0	REAL	0.0 ... 100.0	1.0	Totzonenbreite (bestimmt Größe des Unempfindlichkeitsbereichs)
ERP_ALM	200.0	REAL	0 ... 200.0	100.0	Regeldifferenz: positiver Grenzwert 'Alarm'
ERP_WRN	204.0	REAL	0 ... 200.0	90.0	Regeldifferenz: positiver Grenzwert 'Warnung'
ERN_WRN	208.0	REAL	-200.0 ... 0	-90.0	Regeldifferenz: negativer Grenzwert 'Warnung'
ERN_ALM	212.0	REAL	-200.0 ... 0	-100.0	Regeldifferenz: negativer Grenzwert 'Alarm'
ER_HYS	216.0	REAL	≥0	1.0	Regeldifferenz Hysterese (Vermeidet 'Flackern' der Überwachungsanzeige)
I_ITLVAL	220.0	REAL	-100.0 ... 100.0 [%]	0.0	Initialisierungswert für I–Anteil
LMNR_FAC	224.0	REAL	gesamter Wertebereich (dimensionslos)	1.0	Stellungsrückmeldungs–Faktor (Faktor zur Anpassung des Rückmeldungsbereichs)
LMNR_OFF	228.0	REAL	-100.0 ... 100.0 [%]	0.0	Stellungsrückmeldungs–Offset (Nullpunkt der Rückmeldungsnormierung)
PULSE_TM	232.0	TIME	= n * CYCLE / n=0,1,2...	T#3s	Mindestimpulsdauer
BREAK_TM	236.0	TIME	= n * CYCLE / n=0,1,2...	T#3s	Mindestpausendauer

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
MTR_TM	240.0	TIME	≥ CYCLE	T#30s	Motorstellzeit
PHASE	244.0	INT		0	Phase vom PID Self Tuner
QPVH_ALM	246.0	BOOL		FALSE	Istwert: oberer Grenzwert 'Alarm' angesprochen
QPVH_WRN	246.1	BOOL		FALSE	Istwert: oberer Grenzwert 'Warnung' angesprochen
QPVL_WRN	246.2	BOOL		FALSE	Istwert: unterer Grenzwert 'Warnung' angesprochen
QPVL_ALM	246.3	BOOL		FALSE	Istwert: unterer Grenzwert 'Alarm' angesprochen
QR_S_ACT	246.4	BOOL		FALSE	Zeittabelle für Fahrkurve wird bearbeitet
QSP_HLM	246.5	BOOL		FALSE	Sollwert: obere Begrenzung angesprochen
QSP_LLM	246.6	BOOL		FALSE	Sollwert: untere Begrenzung angesprochen
QPVURLMP	246.7	BOOL		FALSE	Istwert: Anstiegsgrenzwert im positiven Bereich angesprochen
QPVDRLMP	247.0	BOOL		FALSE	Istwert: Abstiegsgrenzwert im positiven Bereich angesprochen
QPVURLMN	247.1	BOOL		FALSE	Istwert: Anstiegsgrenzwert im negativen Bereich angesprochen
QPVDRLMN	247.2	BOOL		FALSE	Istwert: Abstiegsgrenzwert im negativen Bereich angesprochen
QERP_ALM	247.3	BOOL		FALSE	Regeldifferenz: positiver Grenzwert 'Alarm' angesprochen
QERP_WRN	247.4	BOOL		FALSE	Regeldifferenz: positiver Grenzwert 'Warnung' angesprochen
QERN_WRN	247.5	BOOL		FALSE	Regeldifferenz; negativer Grenzwert 'Warnung' angesprochen
QERN_ALM	247.6	BOOL		FALSE	Regeldifferenz: negativer Grenzwert 'Alarm' angesprochen
QLMN_HLM	247.7	BOOL		FALSE	Stellwert: obere Begrenzung angesprochen
QLMN_LLM	248.0	BOOL		FALSE	Stellwert: untere Begrenzung angesprochen
NBR_ATMS	250.0	INT		0	Nummer des vom Zeitplangeber aktuell angefahrenen Stützpunktes
RS_TM	252.0	TIME		T#0s	"Aktuelle Restzeit" der Fahrkurve bis zum nächsten Stützpunkt
T_TM	256.0	TIME		T#0s	"Gesamtzeit" = abgelaufene Zeit der Fahrkurve
RT_TM	260.0	TIME		T#0s	"Gesamtrestzeit" = Zeit bis zum Ende der Fahrkurve
ER	264.0	REAL		0.0	Regeldifferenz
LMN_P	268.0	REAL		0.0	P–Anteil
LMN_I	272.0	REAL		0.0	I–Anteil
LMN_D	276.0	REAL		0.0	D–Anteil

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SPFC_IN	280.0	REAL		0.0	Sollwert FC–Eingang (wird mit Eingang des selbstdefinierten FC verschaltet)
PVFC_IN	284.0	REAL		0.0	Istwert FC–Eingang (wird mit Eingang des selbstdefinierten FC verschaltet)
SP_OP_ON	288.0	BOOL		FALSE	Sollwertbedienung einschalten (der Wert von SP_OP wird als Sollwert übernommen)
PV_OP_ON	288.1	BOOL		FALSE	Istwertbedienung einschalten (der Wert von PV_OP wird als Sollwert übernommen)
LMNOP_ON	288.2	BOOL		FALSE	Stellwertbedienung einschalten (der Wert von LMN_OP wird als Sollwert übernommen)
LMNSO-PON	288.3	BOOL		FALSE	Stellwertsignalbedienung einschalten (LMNUP_OP und LMNDN_OP werden als Stellsignale übernommen)
LMNUP_OP	288.4	BOOL		FALSE	Stellwertsignal Hoch
LMNDN_OP	288.5	BOOL		FALSE	Stellwertsignal Tief
LMNRS_ON	288.6	BOOL		FALSE	Simulation der Stellungsrückmeldung einschalten
SP_OP	290.0	REAL		0.0	Sollwertbedienung am Konfigurierwerkzeug
PV_OP	294.0	REAL		0.0	Istwertbedienung am Konfigurierwerkzeug
LMN_OP	298.0	REAL		0.0	Stellwertbedienung am Konfigurierwerkzeug
LMNRSVAL	302.0	REAL		0.0	Startwert der simulierten Stellungsrückmeldung
LMNR_SIM	306.0	REAL		0.0	Aktueller Wert der simulierten Stellungsrückmeldung
MP1	310.0	REAL		0.0	Messpunkt 1: Interner Sollwert
MP2	314.0	REAL		0.0	Messpunkt 2: Externer Sollwert
MP3	318.0	REAL		0.0	Messpunkt 3: Unbegrenzter Sollwert
MP4	322.0	REAL		0.0	Messpunkt 4: Istwert von der Peripheriebaugruppe
MP5	326.0	REAL		0.0	Messpunkt 5: Istwert nach dem Verzögerungsglied 1. Ordnung
MP6	330.0	REAL		0.0	Messpunkt 6: Wirksamer Istwert (PV)
MP7	334.0	REAL		0.0	Messpunkt 7: Stellwert vom PID–Algorithmus
MP8	338.0	REAL		0.0	Messpunkt 8: Hand–Stellwert
MP9	342.0	REAL		0.0	Messpunkt 9: Unbegrenzter Stellwert
MP10	346.0	REAL		0.0	Messpunkt 10: Stellungsrückmeldung Peripherie

Parameter	Offset	Datentyp	Wertebereich	Vorbelegung	Beschreibung
MP11	350.0	REAL		0.0	Messpunkt 11: Rückkopplungswert (LMNR_ON = FALSE) Stellungsrückmeldung (LMNR_ON = TRUE)
MP12	354.0	REAL		0.0	Messpunkt 12: Dreipunktglied Eingang

Blockschaltbilder PID_ES

Signalflussdarstellungen

Analog zur Darstellung der Schalter im Konfigurationswerkzeug bedeutet der schwarze Punkt in den Schaltersymbolen, dass das Schaltsignal den nebenstehenden boolschen Wert (0 = FALSE oder 1 = TRUE) hat und dass der Signalweg jeweils über diesen Punkt durchgeschaltet ist. Die Schaltsignale (Binärsignale) sind durch gestrichelte Linien gekennzeichnet.

Das bedeutet, es wird der Sollwert absolut über SP_INT vorgegeben, das gleiche gilt für die Eingabe des Istwertes über PV_IN. Als Regelfunktion ist ein normaler PI–Regler mit P–Funktion im Vorwärtszweig eingestellt. Der Regelkreis ist geöffnet, und die Stellgröße wird im Prozent–Bereich über den Eingang MAN beeinflusst. Alle übrigen Funktionen sind passiv oder - falls nicht abschaltbar - durch Vorbesetzung von "Rand"–Parametern bei Signalverlauf innerhalb des Messbereichs bzw. des Arbeitsbereiches nicht wirksam.

Symbole und Kennungen in den Signalflussplänen

Die Bezeichnungen der anschließbaren Prozessgrößen sind unterlegt dargestellt. Sie erkennen daran, wo die Reglerstruktur mit der S7–Peripherie oder direkt mit den Messgliedern und den Stellgliedern des Prozesses verbunden werden kann.

Parameterbezeichnungen mit der Buchstabengruppe "OP" (z. B. SP_OP/ SP_OP_ON) verweisen auf den an dieser Stelle möglichen Eingriff über das Konfigurationswerkzeug von Standard PID Control. Das Konfigurationswerkzeug hat eine eigene Schnittstelle zum Regler–FB.

Zwischengrößen im Signalverlauf können durch die mit Kreisen markierten Messpunkte MP1 ... MP12 beobachtet werden. Diese Zwischengrößen werden zum Abgleich von Werten vor Auslösung "stoßfreier" Umschaltvorgänge oder zur Einsichtnahme in aktuelle Zustände der betreffenden Regelung benötigt. Die Messpunktgrößen im Kurvenschreiber des Konfigurationswerkzeugs können (statisch und dynamisch) dargestellt werden.

Der Übersichtlichkeit wegen sind die Parameter zur Einstellung und Dimensionierung der Bearbeitungsvorgänge (Algorithmen) bei den einzelnen Funktionsblöcken jeweils mit angegeben. Wir verweisen dazu auf die Beschreibungen im Referenzteil und auf die Darstellung einzelner Teilfunktionen in den folgenden Abschnitten.

Aufbereitung des Sollwerts

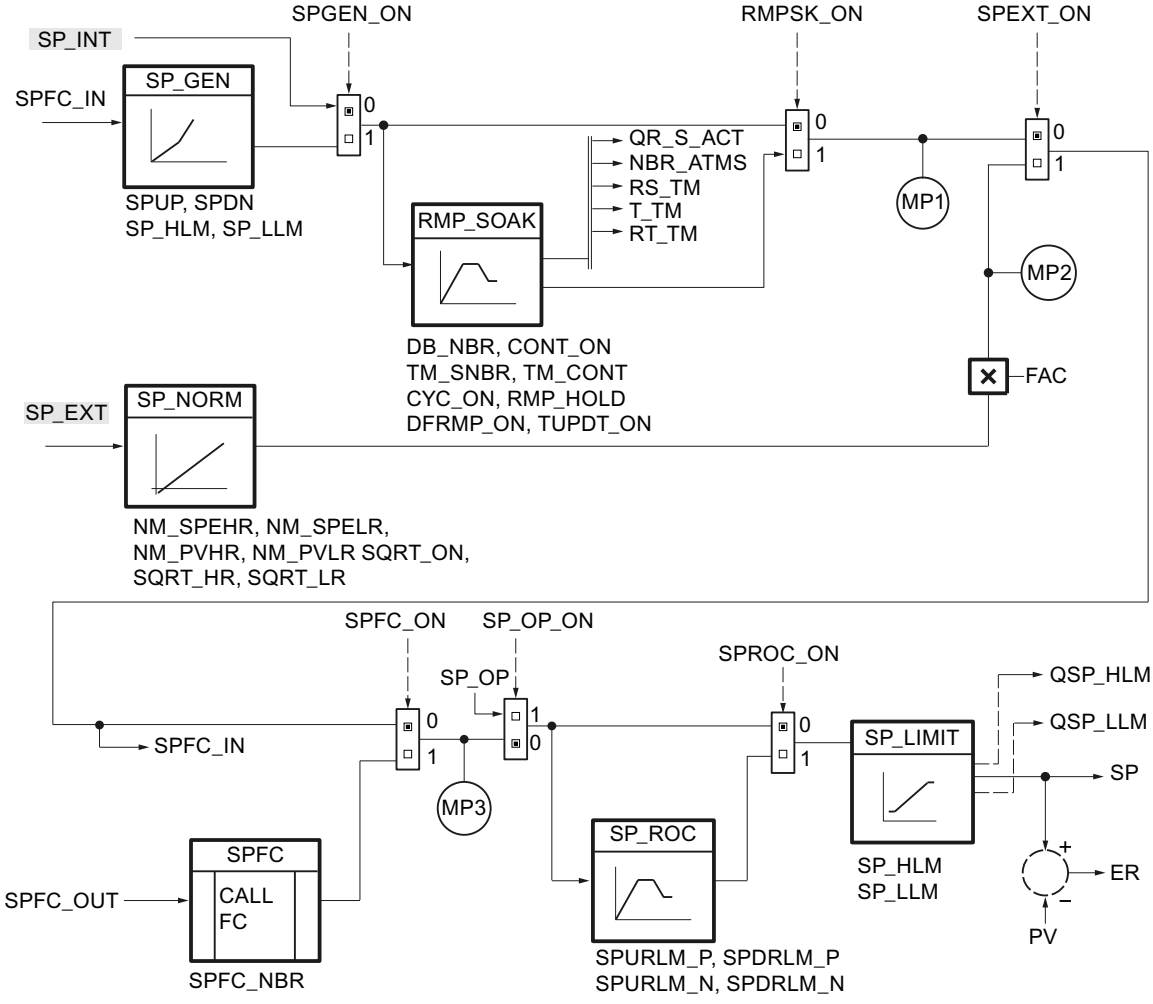

Aufbereitung des Istwerts

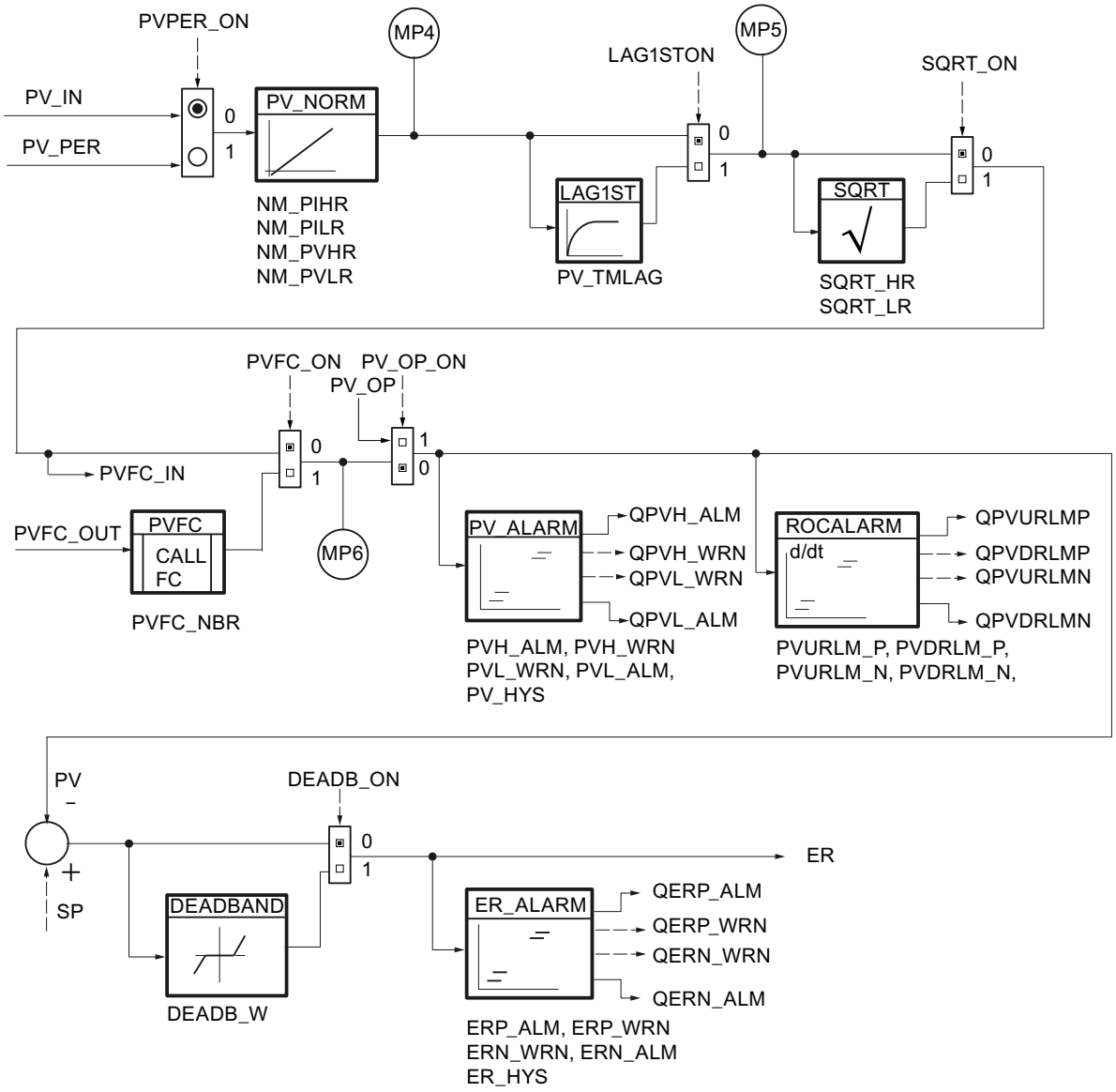

Regelalgorithmus

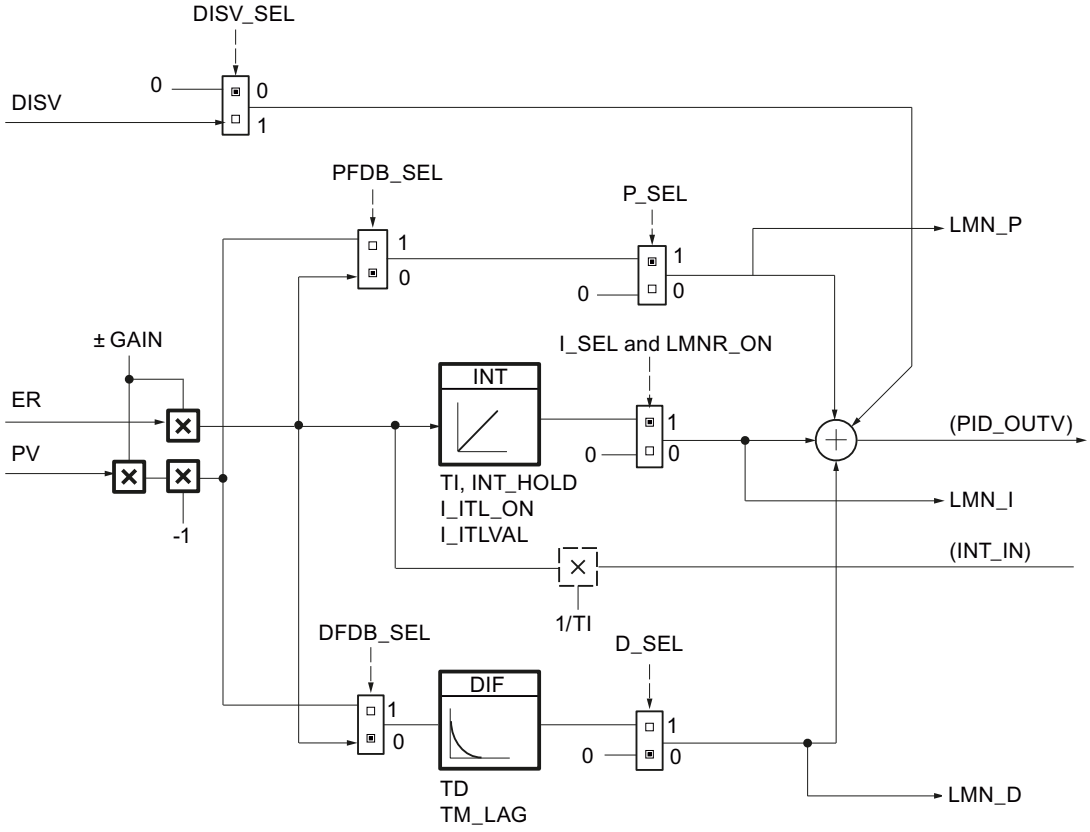

Schrittregler ohne Stellungsrückmeldung (LMNR_ON = FALSE)

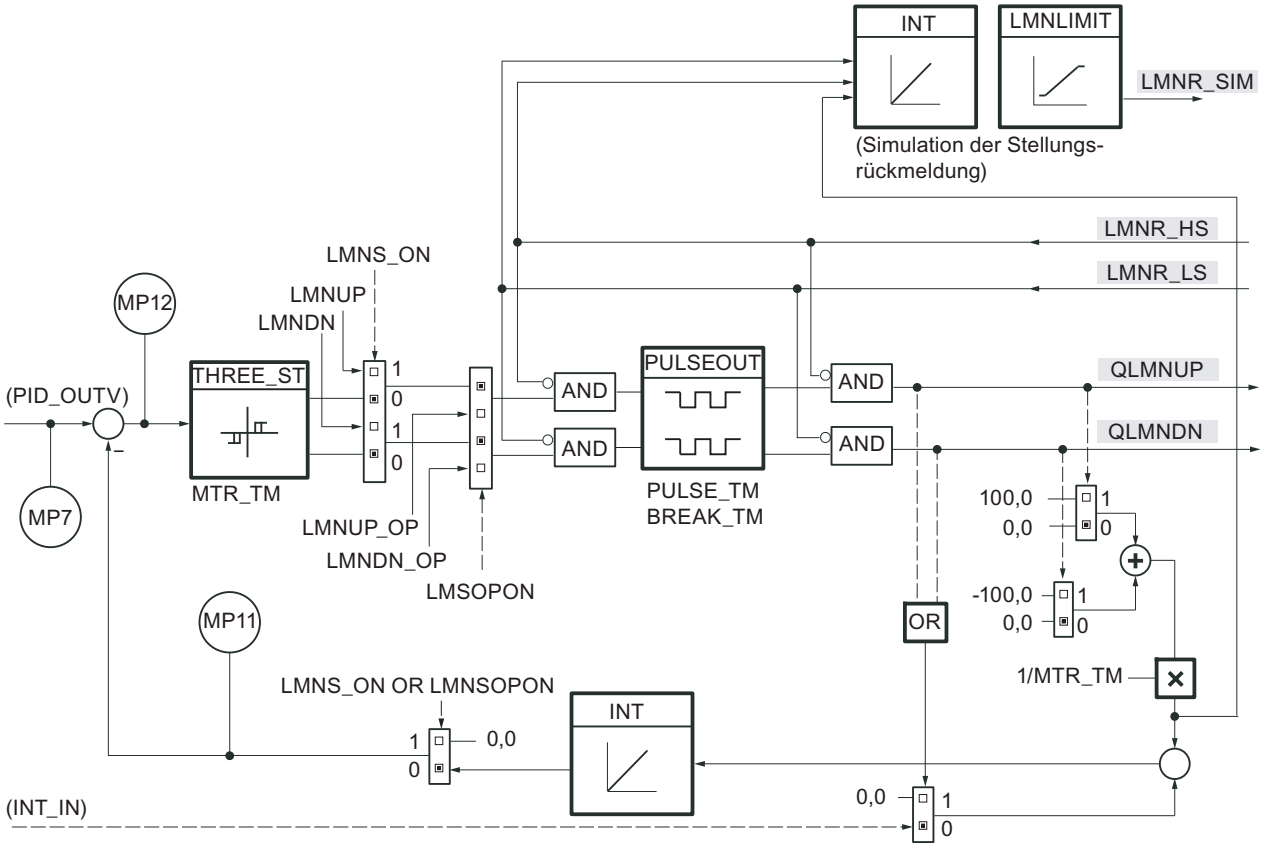

Schrittregler mit Stellungsrückmeldung (LMNR_ON = TRUE)

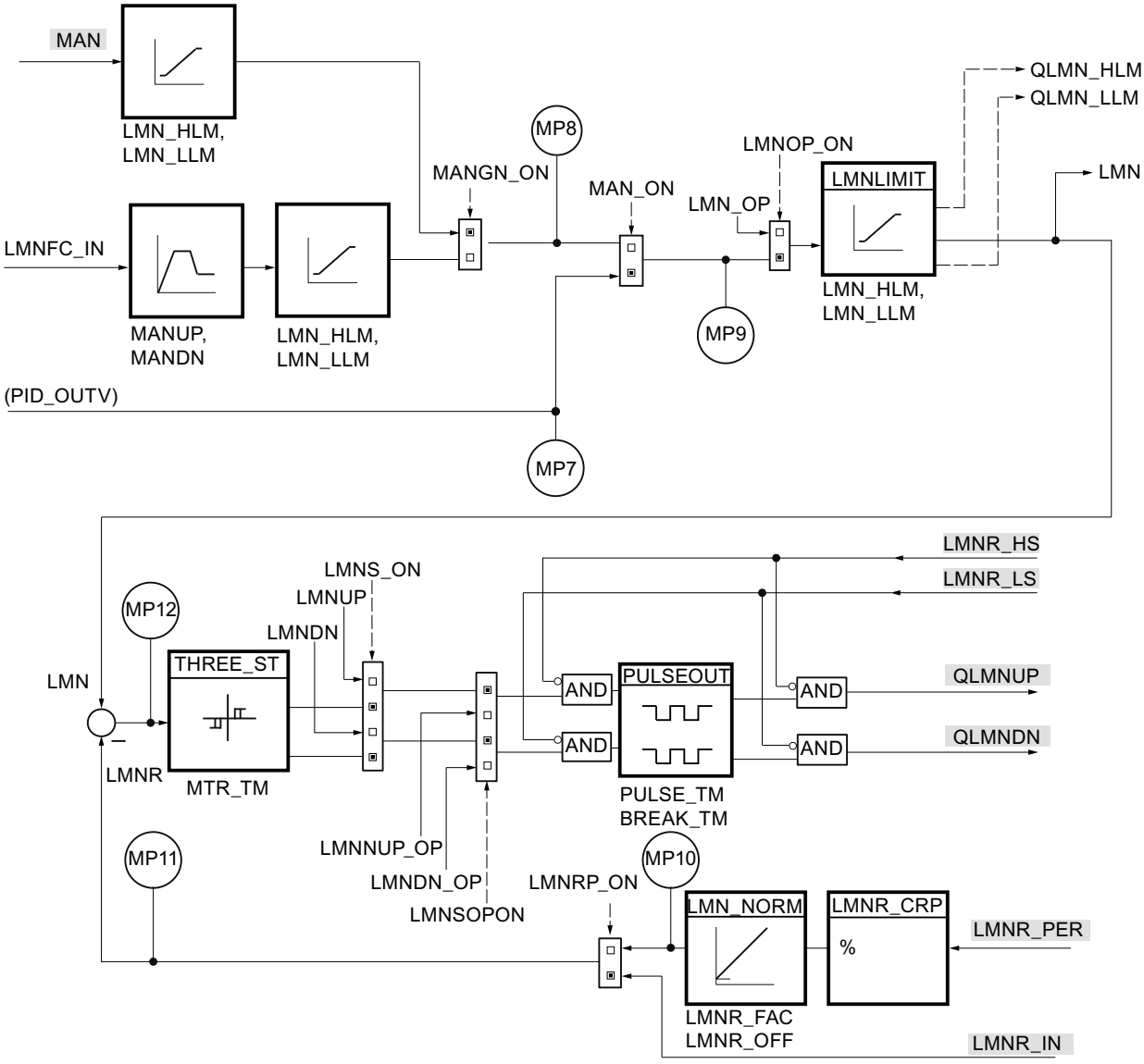

Globaler Datenbaustein DB_RMPSK

Hinweis

Globalen Datenbaustein selbst erstellen

Die Stützpunkte für den Zeitplangeber (RMP_SOAK) werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
NBR_PTS	INT	4	Anzahl der Stützpunkte -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Struktur für die Daten der Stützpunkte. Das Array muss von 0...NBR_PTS deklariert werden.
PI[0].OUTV	REAL	0.0	Ausgangswert [0] Startpunkt technischer Wertebereich
PI[0].TMV	TIME	T#1s	Zeitwert [0] Startpunkt
PI[1].OUTV	REAL	0.0	Ausgangswert [1] Stützpunkt 1 technischer Wertebereich
PI[1].TMV	TIME	T#1s	Zeitwert [1] Stützpunkt 1
PI[2].OUTV	REAL	0.0	Ausgangswert [2] Stützpunkt 2 technischer Wertebereich
PI[2].TMV	TIME	T#1s	Zeitwert [2] Stützpunkt 2
PI[3].OUTV	REAL	0.0	Ausgangswert [3] Stützpunkt 3 technischer Wertebereich
PI[3].TMV	TIME	T#1s	Zeitwert [3] Stützpunkt 3
PI[4].OUTV	REAL	0.0	Ausgangswert [4] Stützpunkt 4 technischer Wertebereich
PI[4].TMV	TIME	T#0s	Zeitwert [4] Stützpunkt 4 0 ms

Datenbaustein über eine externe Quelldatei generieren

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren.

Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den unten stehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen "DB":
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Externe Quellen".
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.

12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_RMPSK"
TITLE = data block ramp soak
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_RMPSK
   STRUCT
      NBR_PTS : Int;              // number of points (excluding
                                  starting point)
      PI : Array[0..4] of Struct
         OUTV : Real;             // output variable
         TMV : Time;              // output time value
      END_STRUCT;
   END_STRUCT;


BEGIN
   NBR_PTS := 4;
   PI[0].TMV := T#1s;
   PI[1].TMV := T#1s;
   PI[2].TMV := T#1s;
   PI[3].TMV := T#1s;
   PI[4].TMV := T#0s;
END_DATA_BLOCK
```

LP_SCHED

Beschreibung LP_SCHED

LP_SCHED wird verwendet, wenn die Zahl der Weckalarme einer CPU nicht ausreicht, um die gewünschten (verschiedenen) Abtastzeiten zu realisieren. Er ermöglicht den Aufruf von bis zu 256 Regelkreisen mit Abtastzeiten, die ein ganzzahliges Vielfaches des Weckalarmzeittakts betragen.

LP_SCHED liest aus dem globalen Datenbaustein DB_LOOP die von Ihnen vorgegebenen Parameter, berechnet sich daraus die für die Aufrufverteilung notwendigen Variablen und legt diese wieder im Datenbaustein DB_LOOP ab.

Globaler Datenbaustein DB_LOOP (Seite 5957) ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Hinweis

Der Baustein überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch, ob der Parameter GLP_NBR (größte Regelkreisnummer) zur Länge des Datenbausteins passt. Bei inkorrekter Parametrierung geht die CPU mit "internem Systemfehler" in STOP.

Aufruf

LP_SCHED müssen Sie in einem Weckalarm-OB vor allen Reglern aufrufen.

Bei der Wertzuweisung an die Eingangsparameter beachten Sie das Folgende:

- TM_BASE: An diesem Eingang geben Sie den Zeittakt des Weckalarm–OB an, in dem die Anweisung LP_SCHED aufgerufen wird.
- COM_RST: Bei einem Anlauf der CPU müssen Sie LP_SCHED einmal mit COM_RST = TRUE aufrufen. Im zyklischen Betrieb (Weckalarm) müssen Sie LP_SCHED mit COM_RST = FALSE aufrufen.
- DB_NBR: An diesem Eingang geben Sie die Nummer des Datenbausteins "DB_LOOP" an, auf den LP_SCHED zugreifen soll.

Nach dem Aufruf der Anweisung LP_SCHED müssen Sie alle zugehörigen Regler bedingt aufrufen. Der Aufruf der Regler PID_CP oder PID_ES kann nicht durch LP_SCHED erfolgen, da beim Aufruf der Anweisungen deren Ein– und Ausgangsparameter belegt werden müssen.

Im Folgenden ist ein Beispiel zum Aufruf der Anweisung LP_SCHED und zum bedingten Aufruf eines Reglers angegeben.

AWL

```
CALL "LP_SCHED"
    TM_BASE:=                                   Hier wird der Zeittakt des Weckalarms paramet-
                                                riert. Bsp.: T#100ms oder #CYCLE mit CYCLE =
                                                Eingangsparameter des Bausteins, in dem die
                                                Anweisung LP_SCHED aufgerufen wird.
    COM_RST:=                                   Hier wird der Anweisung LP_SCHED mitgeteilt,
                                                ob ein Initialisierungslauf der aufgerufenen
                                                Regelkreise stattfinden soll. Bsp.: FALSE
                                                oder #COM_RST mit COM_RST = Eingangsparameter
                                                des Bausteins, in dem die Anweisung LP_SCHED
                                                aufgerufen wird.
    DB_NBR:=                                    Hier wird die Nummer des DB "DB_LOOP" paramet-
                                                riert, den die Anweisung LP_SCHED bearbeiten
                                                soll. Bsp.: "DB_LOOP" mit DB_LOOP = in der
                                                Symboltabelle vergebener Name des DB.
U "DB_LOOP".LOOP_DAT[1].ENABLE
SPBN M002                                       Regelkreisaufruf, falls ENABLE = TRUE
//CALL FBx,DBy
  //COM_RST:= "DB_LOOP".LOOP_DAT[1].COM_RST
```

```
AWL
      //CYCLE:= "DB_LOOP".LOOP_DAT[1].CYCLE
    CLR
    = "DB_LOOP".LOOP_DAT[1].ENABLE                    ENABLE-Bit zurücksetzen
M002:                                                 weiter im Programm, z. B. bedingter Aufruf
                                                      des nächsten Regelkreis-FB
```

Anlauf

Bei COM_RST = TRUE werden folgende Vorbelegungen eingestellt:

Aktuelle Regelkreisnummer: ALP_NBR = 0

Die Aufrufdaten aller Regelkreise bis **GLB_NBR** werden wie folgt vorbelegt:

Freigabe:	ENABLE =	NOT MAN_DIS
Abtastzeit:	CYCLE =	GV(MAN_CYC)
Neustart:	COM_RST =	TRUE
Lokale Aufrufnummer:	ILP_COU =	0
GV(MAN_CYC) = MAN_CYC wird gerundet auf ein ganzzahliges Vielfaches von TM_BASE*GLP_NBR		

Bei einem Anlauf der CPU müssen Sie LP_SCHED aus dem zugehörigen Anlauf–OB heraus aufrufen und dabei dem Eingang COM_RST den Wert TRUE zuweisen. Rufen Sie anschließend PID_CP oder PID_ES bedingt auf, damit die Regler initialisiert werden.

Im Weckalarm–OB müssen Sie COM_RST wieder den Wert FALSE zuweisen.

Arbeitsweise LP_SCHED

Die Aufrufbearbeitung

Die Bearbeitung eines Reglers soll erfolgen, wenn das jeweilige ENABLE–Bit im Datenbaustein DB_LOOP den Wert TRUE hat. Dieses Bit wurde von LP_SCHED zuvor beschrieben. Falls der Regler bearbeitet wurde, müssen Sie dem ENABLE–Bit nach der Bearbeitung den Wert FALSE zuweisen.

Beim Aufruf der Regler müssen Sie deren Eingangsparameter COM_RST und CYCLE mit den Variablen COM_RST[x] und CYCLE[x] des Datenbausteins DB_LOOP verschalten. CYCLE[x] enthält die tatsächliche Abtastzeit des Reglers x und wird von LP_SCHED bei jedem Zyklus beschrieben.

Sie können im laufenden Betrieb den Aufruf einzelner Regler manuell sperren und darüber hinaus einzelne Regler rücksetzen.

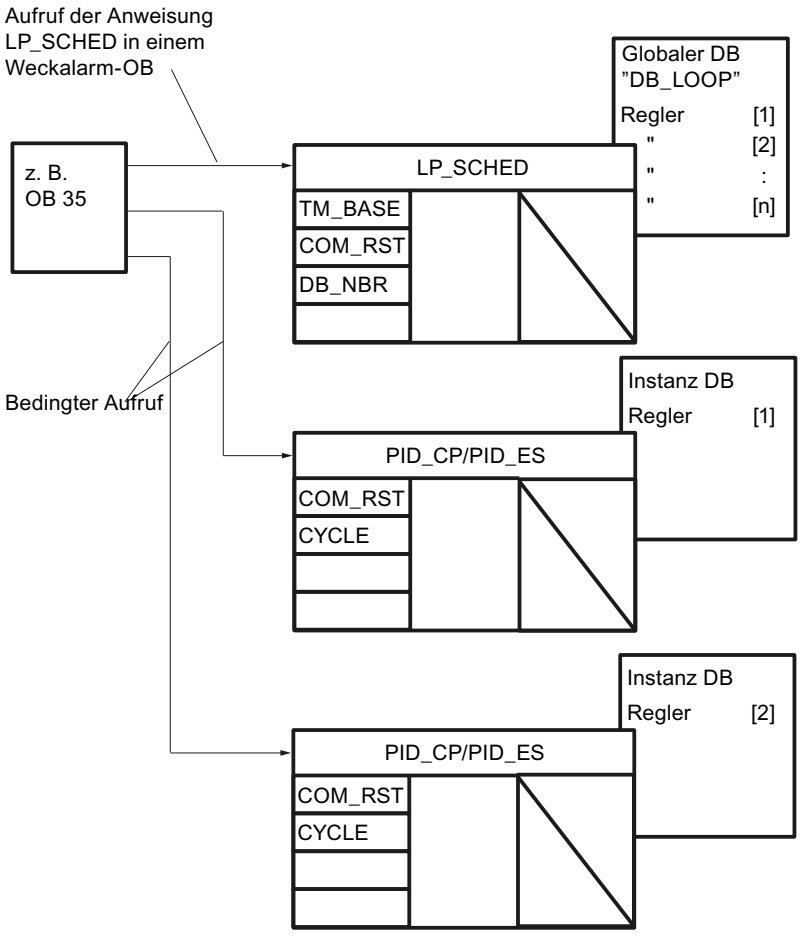

Parametrierung des Datenbausteins DB_LOOP

Die Parametrierung des Regleraufrufverteilers muss im DB-Editor vorgenommen werden.

Sie müssen die folgenden Variablen im Datenbaustein DB_LOOP parametrieren:

- GLP_NBR: Anzahl der Regler, deren Aufruf von LP_SCHED verwaltet wird (max. 256)
- MAN_CYC[x], x = 1 ... GLP_NBR: die von Ihnen gewünschte Abtastzeit für die einzelnen Regler. Bitte beachten Sie dabei für jeden Regler die unten angegebene Bedingung für MAN_CYC[x]. Sonst kann die parametrierte Abtastzeit nicht garantiert werden.

Die folgenden Variablen können Sie im laufenden Betrieb ändern:

- MAN_DIS[x] dient zum Sperren des Aufrufs von Regelkreis x im laufenden Betrieb.
- MAN_CRST[x] dient zum Anstoß eines Initialisierungslaufs für den Regelkreis x im laufenden Betrieb.
  Die Änderung wird mit dem nächsten Aufruf der Anweisung LP_SCHED wirksam.

Die folgenden Variablen werden von LP_SCHED beschrieben:

- In die Variable ENABLE[x] trägt LP_SCHED die Aufrufbedingung für den Regler x ein.
- Die Variablen ALP_NBR und ILP_COU[x] sind interne Variablen der Anweisung LP_SCHED. Sie können Ihnen zur Beobachtung nützlich sein.

Die Variablen COM_RST[x] und CYCLE[x] werden mit den Eingangsparametern COM_RST und CYCLE der Anweisungen PID_CP und PID_ES verschaltet.

Abtastzeit MAN_CYC[x]

LP_SCHED kann pro Aufruf maximal einen Regler bearbeiten. Es vergeht daher die Zeit

TM_BASE * GLP_NBR, bis die erstmalige Bearbeitung sämtlicher Regler abgeschlossen ist. Bei der Parametrierung der gewünschten Abtastzeiten MAN_CYC[x] müssen Sie daher für jeden Regler folgende Bedingung einhalten:

Die Abtastzeit des Reglers x muss ein ganzzahliges Vielfaches aus dem Produkt von Zeitbasis und der Anzahl der zu bearbeitenden Regler sein.

$$\text{MAN_CYC[x]} \overset{!}{=} \text{GV (TM_BASE * GLP_NBR), x = 1, ... GLP_NBR}$$

Die tatsächliche Abtastzeit CYCLE[x] des Regelkreises x wird von LP_SCHED in jedem Zyklus aus MAN_CYC[x] wie folgt ermittelt:

- Falls Sie die obige Vorschrift eingehalten haben, ist die tatsächliche Abtastzeit CYCLE[x] mit der von Ihnen vorgegebenen Abtastzeit MAN_CYC[x] identisch.
- Falls Sie die obige Bedingung nicht eingehalten haben, steht in CYCLE[x] der Wert, der sich ergibt, wenn man MAN_CYC[x] auf das nächste ganzzahlige Vielfache von TM_BASE * GLP_NBR abrundet.

Hinzufügen weiterer Regelkreise

Wenn Sie einen oder mehrere Regelkreise in den Datenbaustein DB_LOOP einfügen wollen, erhöhen Sie den Startwert für GLP_NBR und erweitern das Array LOOP_DAT[1...GLP_NBR].

Überprüfen Sie die gewünschte Abtastzeit MAN_CYC[x] für jeden Regler. Beachten Sie dabei die Bedingung für MAN_CYC.

Pulsgenerator in Verbindung mit LP_SCHED

Wenn Sie beim kontinuierlichen Regler PID_CP den Pulsgenerator eingeschaltet haben, muss statt des Parameters CYCLE die Impulsrasterbreite CYCLE_P mit dem Parameter LOOP_DAT[x].CYCLE verschaltet werden.

Beispiel für eine Aufrufverteilung

Das folgende Beispiel zeigt die Aufrufreihenfolge von vier Reglern in einem Weckalarm–OB. Pro Einheit der Zeitbasis TM_BASE wird maximal ein Regler bearbeitet. Die Aufrufreihenfolge ergibt sich aus der Reihenfolge der Regler im Datenbaustein DB_LOOP.

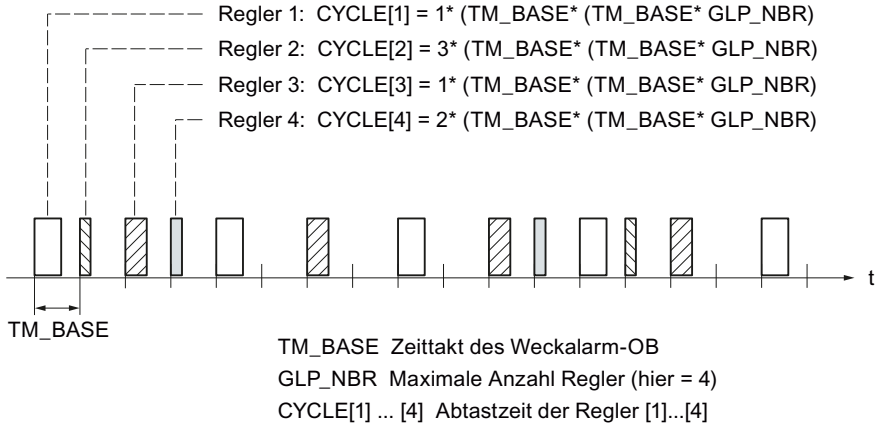

Aufruf von mehr als einem Regler pro Zyklus

Wenn in einem Zyklus mehr als ein Regler bearbeitet werden soll, dann darf LP_SCHED auch mehrmals aufgerufen werden. Alle Aufrufe der Anweisung LP_SCHED müssen vor dem Aufruf der Regler erfolgen. Am Eingangsparameter TM_BASE müssen Sie dann den Zeittakt des Weckalarm–OB dividiert durch die Anzahl der Aufrufe LP_SCHED eingeben.

Beispiel: LP_SCHED wird im OB 35 zweimal aufgerufen und der OB 35 wird alle 100 ms bearbeitet. Dann muss der Eingangsparameter TM_BASE mit 50 ms parametriert werden.

Laufzeiten

Die Summe der Laufzeiten der Anweisung LP_SCHED und der Regler, die in einem Zyklus bearbeitet werden, darf nicht größer sein als der Zeittakt des Weckalarm–OB.

Bedienungen im laufenden Betrieb

Folgende Änderungen im Datenbaustein "DB_LOOP" sind im laufenden Betrieb erlaubt, wenn dabei nur der jeweilige Parameter geändert und nicht der komplette Datenbaustein auf die CPU geladen wird:

- Sperren einzelner Regler
  Durch Beschreiben der Variable MAN_DIS[x] mit dem Wert TRUE wird die Bearbeitung des Reglers x im laufenden Betrieb gesperrt. LP_SCHED setzt das ENABLE–Bit dieses Reglers so lange nicht mehr auf TRUE, bis Sie MAN_DIS[x] wieder mit dem Wert FALSE beschreiben.
- Initialisieren eines Regelkreises
  Durch Beschreiben der Variable MAN_CRST[x] mit dem Wert TRUE können Sie einen einzelnen Regler neu starten: In diesem Fall beschreibt LP_SCHED bei der nächsten Bearbeitung des Regler x die Variable COM_RST[x] mit TRUE. Bei der übernächsten Bearbeitung dieses Regelkreises beschreibt LP_SCHED die Variablen MAN_CRST[x] und COM_RST[x] mit FALSE.
- Änderung der Abtastzeit eines Regelkreises
  Den Parameter MAN_CYC[x] dürfen Sie im laufenden Betrieb ändern.

Hinweis

Wird der ganze Datenbaustein DB_LOOP neu auf die CPU geladen, ohne dass die CPU einen Anlauf durchführen soll, dann müssen Sie die internen Regelkreiszähler ILP_COU[x], x = 1, ... GLP_NBR und die Nummer des aktuellen Reglkers ALP_NBR mit Null vorbelegen.

Beobachtung der Anweisung LP_SCHED

In die Variable ALP_NBR des Datenbausteins DB_LOOP trägt LP_SCHED die Nummer des nächsten zu bearbeitenden Reglers ein. Die Nummer des jeweiligen Reglers ergibt sich aus der Platzierung seiner Aufrufdaten in der Reihenfolge der Eintragungen im Datenbaustein.

Die Variable ILP_COU[x] ist der interne Zähler. Er enthält die Zeitdauer bis zum nächsten Aufruf des entsprechenden Reglers. Die Zeiteinheit von ILP_COU ist das Produkt aus Zeitbasis TM_BASE und Anzahl der Regler GLP_NBR. Ist ILP_COU = 0, dann setzt LP_SCHED das ENABLE–Bit des betreffenden Reglers.

Wenn Regelkreise wider Erwarten nicht aufgerufen werden

Wenn LP_SCHED aufgerufen wird, aber einzelne Regelkreise nicht bearbeitet werden, kann dies folgende Ursachen haben:

- Durch Beschreiben der Variable MAN_DIS[x] mit dem Wert TRUE ist die Regelkreisbearbeitung im laufenden Betrieb gesperrt.
- Im Parameter GLP_NBR ist die Anzahl der Regler, die von LP_SCHED bearbeitet werden sollen, zu klein angegeben.
- Die von Ihnen vorgegebenen Abtastzeiten MAN_CYC[x] der einzelnen Regelkreise dürfen nicht kleiner sein als das Produkt aus Zeitbasis TM_BASE und Anzahl der Regler GLP_NBR. Regler, bei denen diese Bedingung verletzt wird, werden nicht bearbeitet.

Eingangsparameter LP_SCHED

Tabelle 4-224

Parameter	Datentyp	Wertebereich	Vorbelegung	Erläuterung
TM_BASE	TIME	≥ 20 ms (S7-300) ≥ 5 ms (S7-400)	100 ms	Zeitbasis (Zeittakt der Weckalarmebene, in der LP_SCHED aufgerufen wird) Gültig sind Werte
COM_RST	BOOL		FALSE	Neustart (Neustartroutine des LP_SCHED wird bearbeitet)
DB_NBR	BLOCK_DB		DB 1	Datenbausteinnummer (DB mit den Aufrufdaten der Regelkreise)

Globaler Datenbaustein DB_LOOP

Die Daten für die Aufrufverteilung mit LP_SCHED werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
GLP_NBR	INT	2	Maximale Anzahl Regler 1 - 256
ALP_NBR	INT	0	Nummer des aktuellen Reglers 0 - 256
LOOP_DAT	Array [1...GLP_NBR] of Struct		Struktur für die Daten der einzelnen Regler. Das Array muss von 1...GLP_NBR deklariert werden.
LOOP_DAT[1].MAN_CYC	TIME	T#1s	Daten Regler 1: manuelle Abtastzeit >= 20 ms (S7-300) >= 5 ms (S7-400)
LOOP_DAT[1].MAN_DIS	BOOL	FALSE	Daten Regler 1: manuell Regleraufruf sperren
LOOP_DAT[1].MAN_CRST	BOOL	FALSE	Daten Regler 1: manuell Neustart setzen
LOOP_DAT[1].ENABLE	BOOL	FALSE	Daten Regler 1: Reglerfreigabe
LOOP_DAT[1].COM_RST	BOOL	FALSE	Daten Regler 1: Neustart
LOOP_DAT[1].ILP_COU	INT	0	Daten Regler 1: interner Zähler
LOOP_DAT[1].CYCLE	TIME	T#1s	Daten Regler 1: Abtastzeit >= 20 ms (S7-300) >= 5 ms (S7-400)
LOOP_DAT[2].MAN_CYC			Daten Regler 2: manuelle Abtastzeit
...			

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den untenstehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.

3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen „.DB“:
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner „Externe Quellen“.
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_LOOP"
TITLE = data block loop scheduler
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_LOOP
    STRUCT
        GLP_NBR : Int;                  // greatest loop number
        ALP_NBR : Int;                  // actual loop number
        LOOP_DAT : Array[1..2] of Struct
            MAN_CYC : Time;             // loop data: manual sample time
            MAN_DIS : Bool;             // loop data: manual loop disable
            MAN_CRST : Bool;            // loop data: manual complete restart
            ENABLE : Bool;              // loop data: enable loop
            COM_RST : Bool;             // loop data: complete restart
            ILP_COU : Int;              // loop data: internal loop counter
            CYCLE : Time;               // loop data: sample time
        END_STRUCT;
    END_STRUCT;


BEGIN
    GLP_NBR := 2;
    LOOP_DAT[1].MAN_CYC := T#1S;
    LOOP_DAT[1].CYCLE := T#1S;
    LOOP_DAT[2].MAN_CYC := T#200MS;
    LOOP_DAT[2].CYCLE := T#200MS;
END_DATA_BLOCK
```

Modular PID Control

Allgemein

Konventionen zu Parameter– und Blocknamen im Blockschaltbild

Für die Bezeichnungen der Parameter wurden nicht mehr als 8 Zeichen gewählt.

Es gelten die folgenden Konventionen zur Bezeichnung der Parameter:

Erstbuchstabe:	
Q	allgemeiner Ausgang vom Typ BOOL (boolsche Größe)
SP	Führungsgröße
PV	Messwert oder Regelgröße
LMN	Stellgröße oder auszugebendes analoges Ausgangssignal
DISV	Störgröße
Folgebuchstaben:	
MAN	Handwert
INT	intern
EXT	extern
_ON	BOOL-Größe zum Einschalten einer Funktion

Aufrufdaten

Die meisten Anweisungen von Modular PID Control benötigen regelkreisspezifische Aufrufdaten wie Neustartbit und Abtastzeit. An den Eingängen COM_RST und CYCLE werden diese Werte übertragen.

Bemerkungen zu den Baustein-Parameterleisten (Eingangs-, Ausgangs- und Durchgangsparameter)

- **Vorbelegung:** Hier stehen die Vorbelegungswerte bei Neuerstellung einer Instanz.
- **Wertebereich**: Die Werte am Eingangsparameter sollten den zulässigen Wertebereich nicht überschreiten. Eine Überprüfung des Bereiches innerhalb der Bausteinabarbeitung erfolgt **nicht**. Bei Angabe technischer Wertebereich ist eine physikalische Größe gemeint, deren Wert zwischen ca. $\pm 10^{6}$ liegen sollte.

A_DEAD_B

Beschreibung A_DEAD_B

Bei verrauschtem Istwert wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam; dies führt durch hohe Schalthäufigkeit (Schrittregler) zu einem erhöhten Verschleiß des Stellgliedes. Durch Unterdrückung des Rauschanteils wird ein Oszillieren des Reglerausgangs vermieden.

Arbeitsweise

Die Anweisung filtert höherfrequente Störsignale aus der Regeldifferenz und bildet einen unempfindlichen Bereich um den Nullpunkt. Liegt die Eingangsgröße innerhalb dieses Bereichs, so wird am Ausgang Null ausgegeben. Die Bereichsbreite wird der Amplitude des Störsignals automatisch angepasst.

Die Anweisung arbeitet nach der Funktion:

OUTV = INV + DB_WIDTH	für	INV < -DB_WIDTH
OUTV = 0.0	für	-DB_WIDTH ≤ INV ≤ +DB_WIDTH
OUTV = INV - DB_WIDTH	für	INV > +DB_WIDTH

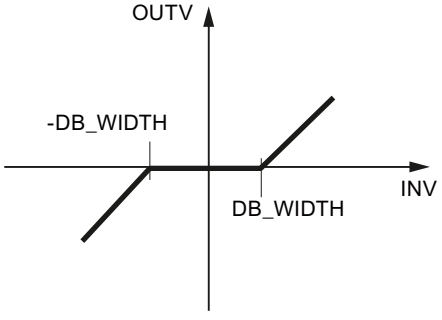

Adaption der Totzone

Die Adaption der Totzone kann ein- und ausgeschaltet werden.

- Adaption Aus
  Wenn die Adaption ausgeschaltet ist (ADAPT_ON = FALSE), wird als wirksame Totzonenbreite DB_WIDTH der letzte DB_WIDTH-Wert eingefroren.
- Adaption Ein
  Mit ADAPT_ON = TRUE kann ein Adaptionsalgorithmus zugeschaltet werden, der die wirksame Totzonenbreite berechnet. Dadurch wird die Totzonenbreite an die Amplitude des Rauschsignals, das der Eingangsgröße überlagert ist, angepasst und damit der Rauschanteil auch bei schwankender Rauschamplitude unterdrückt.
  Bei azyklischem Bausteinaufruf muss die Adaption abgeschaltet werden (ADAP_ON = FALSE).

Wenn bei der Inbetriebnahme die Adaption eingeschaltet wird und sich nach einer gewissen Zeit eine stabile Totzonenbreite einstellt, kann die Adaption ausgeschaltet werden. Die über die Adaption eingestellte Totzonenbreite bleibt erhalten, solange kein Neustart durchgeführt wird.

Adaption Ein

Aus Stabilitätsgründen wird die wirksame Totzonenbreite DB_WIDTH nach unten auf den an der Eingangsleiste vorgebbaren Wert DB_WL_LM begrenzt. Überschreitet das vom Messrauschen überlagerte Eingangssignal INV die augenblicklich eingestellte Totzonenbreite in negativer (1), positiver (2) und wieder negativer Richtung (3) innerhalb der Periodendauer 1/CRIT_FRQ, wird die wirksame Totzonenbreite um den Betrag 0.1 erhöht.

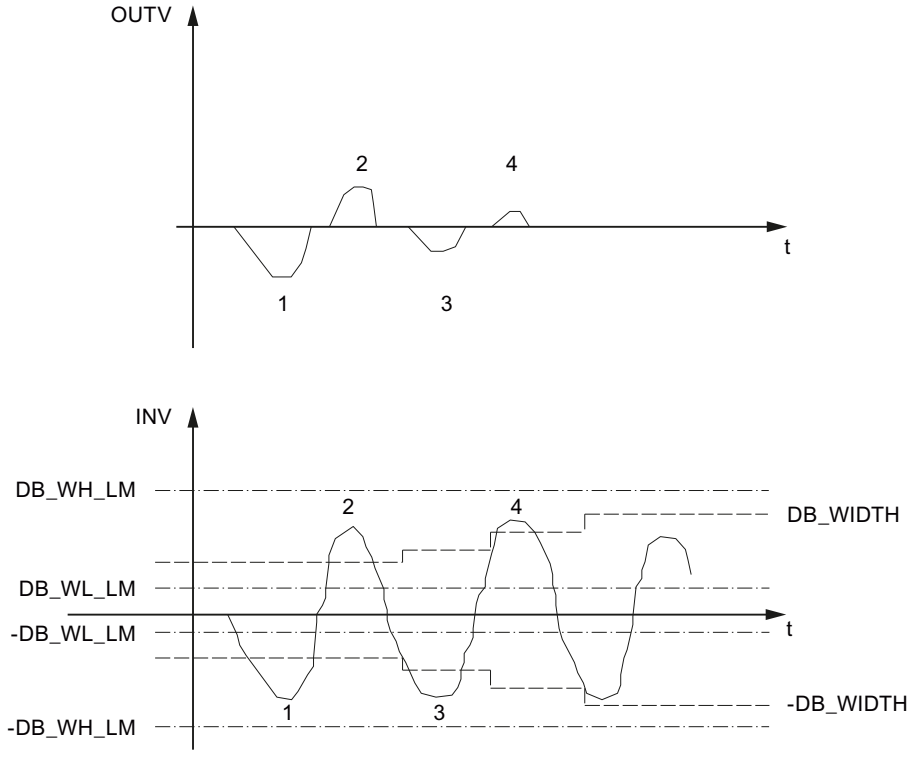

Der Vorgang wird bei positiver oder negativer Überschreitung gestartet. Bei jedem weiteren Überschreiten (3 -> 4) der gegenüberliegenden Totzonenbreite innerhalb der halben Periodendauer wird sie erneut um 0.1 vergrößert. Dieser Vorgang wiederholt sich so oft, bis die Totzonenbreite mit der Amplitude des Messrauschens übereinstimmt. Um nicht beliebig große Eingangssignale zu unterdrücken, wird die wirksame Totzonenbreite durch den Eingang DB_WH_LM nach oben begrenzt. Tritt dagegen innerhalb der Zeit RET_FAC * 1/CRIT_FRQ kein Überschreiten der Totzonenbreite auf, so wird sie um den Betrag 0.1 zurückgenommen.

CRIT_FRQ gibt dabei die Grenzfrequenz an, ab der ein Signalanteil als Störung aufgefasst wird. Sie wird nach oben und unten wie folgt begrenzt:

0.01 ≤ CRIT_FRQ ≤ 1/(3*CYCLE) mit CYCLE: Abtastzeit in sec

Der Parameter RET_FAC legt das Vielfache von 1/CRIT_FRQ fest, nach dem die Totzonenbreite wieder verkleinert wird.

Die Adaptionslogik arbeitet nur, wenn die Eingangsgröße ohne Störanteil nahe bei Null liegt.

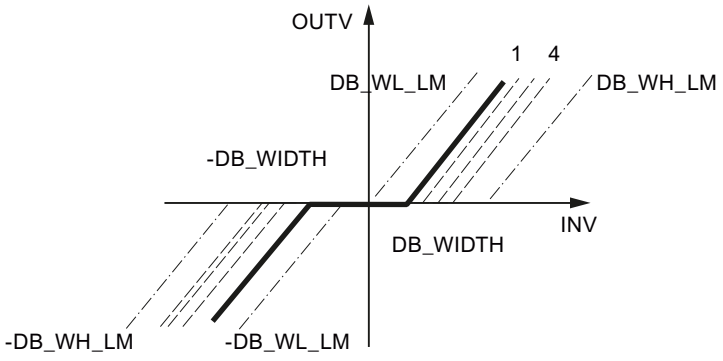

Neustart

Bei Neustart wird OUTV=0.0 und die wirksame Totzonenbreite DB_WIDTH = DB_WL_LM gesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter A_DEAD_B

Parameter	Datentyp	Vorbelegung	Beschreibung
INV	REAL	0.0	Eingangsgröße technischer Wertebereich
DB_WH_LM	REAL	5.0	obere Begrenzung der Totzonenbreite technischer Wertebereich DB_WH_LM > DB_WL_LM
DB_WL_LM	REAL	1.0	untere Begrenzung der Totzonenbreite technischer Wertebereich DB_WL_LM < DB_WH_LM
CRIT_FRQ	REAL	0.1	Grenzfrequenz CRIT_FRQ ≥ 0.01 und CRIT_FRQ ≤ 1/(3 * CYCLE)
RET_FAC	INT	1	Rücknahmefaktor RET_FAC ≥ 1
ADAPT_ON	BOOL	FALSE	Adaptionsalgorithmus einschalten
COM_RST	BOOL	FALSE	Neustart
CYCLE	TIME	T#1s	Abtastzeit CYCLE ≥ 1ms

Ausgangsparameter A_DEAD_B

Parameter	Datentyp	Vorbelegung	Beschreibung
OUTV	REAL	0.0	Ausgangsgröße
DB_WIDTH	REAL	0.0	wirksame Totzonenbreite

CRP_IN

Beschreibung CRP_IN

Die Anweisung passt den Wertebereich der analogen Peripherie an die interne Darstellung der Modularen Regelung an. Die Anweisung kann z.B. im Istwertzweig aufgerufen werden.

Arbeitsweise

CRP_IN wandelt einen Eingangswert im Peripherieformat in einen normierten Gleitpunktwert der modularen Regelung.

Peripherie-Wert	Ausgangswert in %
32767	118,515
27648	100,000
1	0,003617
0	0,000
-1	-0,003617
-27648	-100,000
-32768	-118,519

Der Gleitpunktwert kann über einen Skalierungsfaktor und einen Offset angepasst werden. Der Ausgang ergibt sich aus:

OUTV = INV_PER * 100/27648 * FACTOR + OFFSET

Hinweis

Eine Prüfung auf positiven/negativen Überlauf wird nicht durchgeführt.

Inbetriebnahmewert verwenden

Zur Inbetriebnahme, zum Testen oder bei Peripheriestörungen kann auf einen Inbetriebnahmewert umgeschaltet werden. Ist START_ON = TRUE, wird am Ausgang OUTV der Wert in STARTVAL ausgegeben.

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter CRP_IN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV_PER	0.0	WORD	0	Eingangsgröße für Peripherie technischer Wertebereich
FACTOR	2.0	REAL	1.0	Skalierungsfaktor
OFFSET	6.0	REAL	0.0	Offset technischer Wertebereich
START_ON	10.0	BOOL	TRUE	Inbetriebnahmewert einschalten
STARTVAL	12.0	REAL	0.0	Inbetriebnahmewert technischer Wertebereich

Ausgangsparameter CRP_IN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	16.0	REAL	0.0	Ausgangsgröße

CRP_OUT

Beschreibung CRP_OUT

Die Anweisung passt einen Gleitpunktwert der Modularen Regelung an das Peripherieformat an.

Arbeitsweise

CRP_OUT wandelt einen Eingangswert (normierter Gleitpunktwert der Modularen Regelung) in das Peripherieformat der analogen Peripherie.

Eingangswert in %	Peripheriewert
118,515	32767
100,000	27648
0,003617	1

Eingangswert in %	Peripheriewert
0,000	0
-0,003617	-1
-100,000	-27648
-118,519	-32768

Der Gleitpunktwert kann über einen Skalierungsfaktor und einen Offset angepasst werden. Der Ausgang ergibt sich aus:

OUTV_PER = (INV * FACTOR + OFFSET) * 27648/100

Hinweis

Eine Prüfung auf positiven/negativen Überlauf wird nicht durchgeführt.

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter CRP_OUT

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
FACTOR	4.0	REAL	1.0	Skalierungsfaktor
OFFSET	8.0	REAL	0.0	Offset technischer Wertebereich

Ausgangsparameter CRP_OUT

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV_PER	12.0	WORD	0	Ausgangsgröße für Peripherie

DEAD_T

Beschreibung DEAD_T

Die Anweisung findet Anwendungen in Verhältnisregelungen, bei denen die einzelnen Komponenten auf unterschiedlich langen Transportwegen zusammengeführt werden.

Arbeitsweise

Die Anweisung verzögert die Ausgabe eines Eingangswertes um eine einstellbare Zeit (Totzeit). Bei Online-Änderungen der Totzeit können beim Ausgangswert Sprünge auftreten.

Die Eingangswerte werden in einem globalen Datenbaustein zwischengespeichert.

Globaler Datenbaustein DB_DEADT (Seite 5968) ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Der globale Datenbaustein muss ein Array [0..n_{max}] of real enthalten.

n_{max} ergibt sich aus der Totzeit DEAD_TM und Abtastzeit CYCLE der Anweisung DEAD_T. Folgendes muss gelten:

- DEAD_TM muss ein ganzzahliges Vielfaches von CYCLE sein.
- n_{max}=DEAD_TM/CYCLE

Im Zyklus n wird der gespeicherte Eingangswert INV[n] aus dem globalen Datenbaustein gelesen und an OUTV ausgegeben. INV[n] wird mit dem aktuellen Eingangswert überschrieben. Der Zähler n wird um 1 erhöht. Bei n = n_{max} wird der Zähler auf 0 zurück gesetzt. Die Nummer des globalen Bausteins geben Sie am Eingangsparameter DB_NBR an.

Beispiel

Bei einer Abtastzeit von CYCLE=1s und einer Totzeit von DEAD_TM=5s müssen 5 Eingangswerte zwischengespeichert werden. Der Datenbereich muss 20 Byte lang sein.

Die Daten im globalen Datenbaustein werden wie in einem Ringspeicher abgearbeitet.

Zyklus-Nr.	Zähler n	INV an der Anweisung	Variable im globalen Datenbaustein	Wert von INVI[n] am Zyklusanfang	OUTV an der Anweisung	Wert von INVI [n] am Zyklusende
0	0	80.0	INVI[0]	0.0	0.0	80.0
1	1	100.0	INVI[1]	0.0	0.0	100.0
2	2	100.0	INVI[2]	0.0	0.0	100.0
3	3	80.0	INVI[3]	0.0	0.0	80.0
4	4	0.0	INVI[4]	0.0	0.0	0.0
5	0	0.0	INVI[0]	80.0	80.0	0.0
6	1	0.0	INVI[1]	100.0	100.0	0.0
7	2	0.0	INVI[2]	100.0	100.0	0.0
8	3	10.0	INVI[3]	80.0	80.0	10.0
9	4	100.0	INVI[4]	0.0	0.0	100.0

Der Wert von INV im Zyklus 0 wird erst im Zyklus 5 an OUTV ausgegeben. Bei einer Abtastzeit von 1 s also verzögert um die Totzeit von 5 s.

Nachführen

Bei TRACK=TRUE wird der Eingangswert unverzögert an OUTV ausgegeben. Die Zwischenspeicherung der Eingangswerte wird nicht unterbrochen, damit nach Abschalten von Nachführen die Eingangswerte nach der eingestellten Totzeit ausgegeben werden können. Bei einer fallenden Flanke an TRACK springt OUTV auf INV[DEAD_TM]!

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Hinweis

Die Anweisung überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob die Parameter DEAD_TM (Totzeit) und CYCLE (Abtastzeit) zur Länge des Datenbausteins passen. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

Neustart

Bei Neustart werden alle gespeicherten Eingangswerte gelöscht und OUTV=0.0 ausgegeben.

Eingangsparameter DEAD_T

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
DB_NBR	4.0	BLOCK_DB	DB 1	Datenbausteinnummer
DEAD_TM	6.0	TIME	10s	Totzeit DEAD_TM ≥ CYCLE
TRACK	10.0	BOOL	FALSE	Nachführen OUTV = INV
COM_RST	10.1	BOOL	FALSE	Neustart
CYCLE	12.0	TIME	1s	Abtastzeit CYCLE ≥ 1ms

Ausgangsparameter DEAD_T

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	16.0	REAL	0.0	Ausgangsgröße

Globaler Datenbaustein DB_DEADT

Die Eingangswerte für das Totzeitglied DEAD_T werden in einem globalen Datenbaustein zwischengespeichert. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
INVI	Array[0..n_{max}] of real	0.0	Eingangswert [n] technischer Wertebereich

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den untenstehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen „.DB“:
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner „Externe Quellen“.
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen". Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_DEADT"
TITLE = data block dead time
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_DEADT
    STRUCT
        INVI : Array[0..4] of Real;
    END_STRUCT;
BEGIN
END_DATA_BLOCK
```

DEADBAND

Beschreibung DEADBAND

Bei verrauschtem Istwert wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam; dies führt durch hohe Schalthäufigkeit (Schrittregler) zu einem erhöhten Verschleiß des Stellgliedes. Durch Unterdrückung des Rauschanteils wird ein Oszillieren des Reglerausgangs vermieden.

Arbeitsweise

Die Anweisung DEADBAND (Totzone) unterdrückt bei der Eingangsgröße INV kleine Schwankungen um einen festgelegten Nullpunkt; außerhalb dieser Schwankungen steigt die Ausgangsgröße OUTV proportional zur Eingangsgröße. Sie arbeitet nach der Funktion:

OUTV = INV + DEADB_W - DEADB_O	für	INV < DEADB_O - DEADB_W
OUTV = 0.0	für	DEADB_O - DEADB_W ≤ INV
	und	INV ≤ DEADB_O + DEADB_W
OUTV = INV - DEADB_W - DEADB_O	für	INV > DEADB_O + DEADB_W

Die Signalverfälschung entspricht dem Wert DEADB_W. Der Mittelpunkt der Totzone wird mit DEADB_O festgelegt.

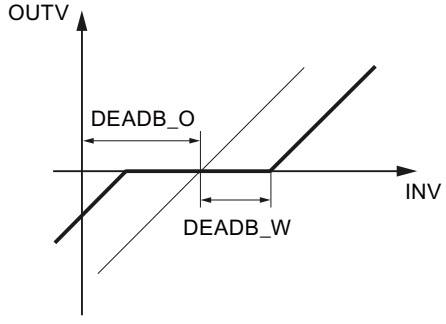

Totzonenbreite DEADB_W und Totzonenversatz DEADB_O sind parametrierbar. Bei Verwendung zur Regeldifferenzbildung muss der Offset DEADB_O=0.0 gesetzt werden.

Das folgende Bild zeigt die Überbrückung eines Rauschanteils mit Offset.

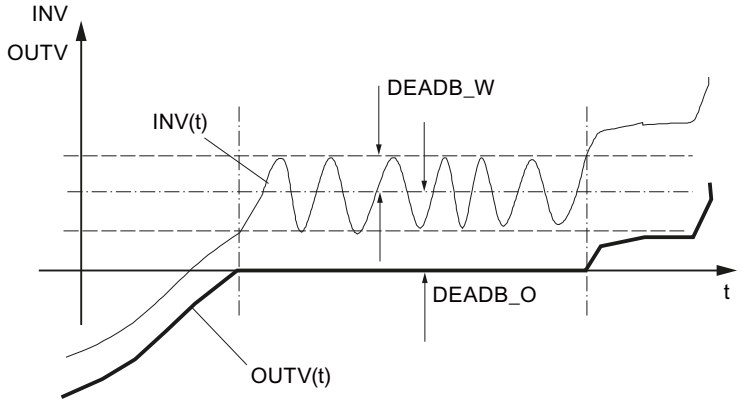

Neustart

Die Anweisung besitzt keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt. Die Totzonenbreite darf nur positive Werte annehmen.

Eingangsparameter DEADBAND

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
DEADB_W	4.0	REAL	1.0	Totzonenbreite technischer Wertebereich und ≥ 0.0
DEADB_O	8.0	REAL	0.0	Totzonenversatz technischer Wertebereich

Ausgangsparameter DEADBAND

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	12.0	REAL	0.0	Ausgangsgröße

DIF

Beschreibung DIF

Prozessgrößen werden dynamisch differenziert. So kann z. B. aus dem zurückgelegten Weg die Geschwindigkeit berechnet werden. Der Differenzierer ist einsetzbar zur Störgrößenaufschaltung, zur Vorsteuerung und zum Aufbau eines Reglers.

Arbeitsweise

Die Anweisung differenziert die Eingangsgröße über der Zeit und glättet das Signal mit einem Verzögerungsglied 1. Ordnung und arbeitet im Laplace-Bereich nach folgender Übertragungsfunktion:

- OUTV(s) / INV(s) = TD / (1+TM_LAG*s)

Das Zeitverhalten wird durch die Differenzierzeit TD und die Verzögerungszeit TM_LAG festgelegt. Die nachfolgenden Bilder zeigen Ihnen die Sprungantwort von DIF.

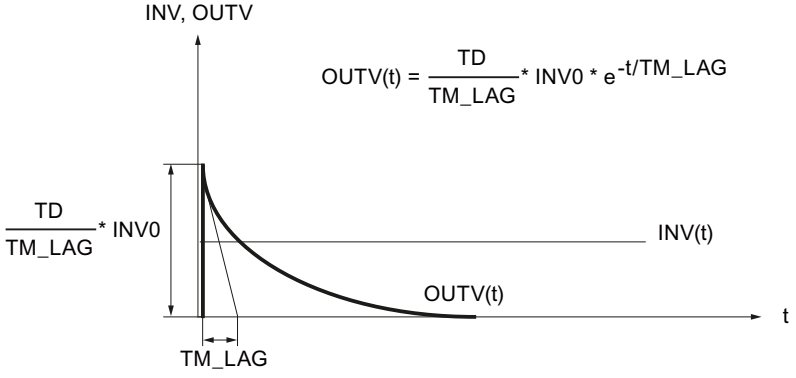

TD	Differenziereit
TM_LAG	Verzöerungszeit
INV0	Eingangssprunghöhe
INV	Eingangsgröße
OUTV	Ausgangsgröße

Ist TM_LAG ≤CYCLE/2 parametriert, arbeitet DIF ohne Verzögerung. Ein Eingangssprung wird mit dem Faktor TD/CYCLE auf den Ausgang gegeben. Nach einem Zyklus geht der Ausgang wieder auf 0.0 zurück.

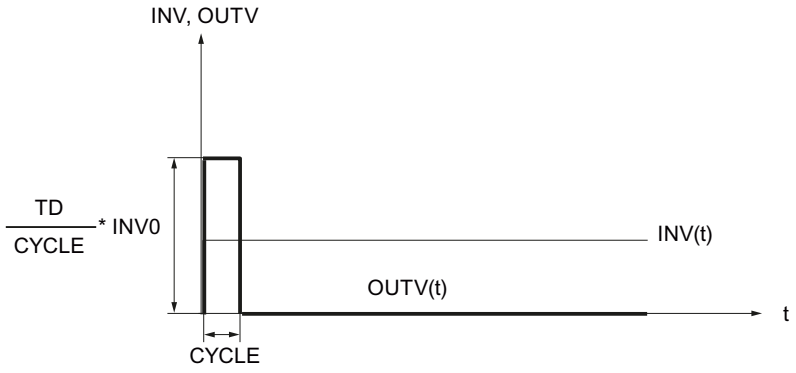

Bausteininterne Begrenzungen

Die Differenzierzeit wird auf die Abtastzeit nach unten begrenzt. Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

TD_{intern} = CYCLE	für	TD < CYCLE
TM_LAG_{intern} = CYCLE/2	für	TM_LAG < CYCLE/2

Die anderen Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt. Der Differenzierer wird intern mit dem momentanen Eingangswert INV vorbelegt. Der Übergang in den Normalbetrieb ist damit bei gleichbleibender Eingangsgröße stoßfrei.

Eingangsparameter DIF

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
TD	4.0	TIME	T#25s	Differenzierzeit ≥ CYCLE
TM_LAG	8.0	TIME	T#5s	Verzögerungszeit
COM_RST	12.0	BOOL	FALSE	Neustart
CYCLE	14.0	TIME	T#1s	Abtastzeit ≥ 1ms

Ausgangsparameter DIF

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	18.0	REAL	0.0	Ausgangsgröße

ERR_MON

Beschreibung ERR_MON

Die Anweisung wird zur Regeldifferenzbildung und -überwachung eingesetzt.

Arbeitsweise

Die Anweisung berechnet die Regeldifferenz ER = SP - PV und überwacht sie auf eingebbare Grenzen. Durch eine Sollwertänderung |ΔSP| > SP_DIFF wird das Ansprechen des Grenzwertsignals ER_LM für eine einstellbare Verzögerungszeit (TM_DELAY+TM_RAMP) unterdrückt. Während dieser Zeit wird ER durch den höheren Grenzwert ER_LMTD überwacht. Wird ER_LMTD überschritten, wird QER_LMTD=TRUE ausgegeben. Nach Ablauf der Verzögerungszeit wird ER_LMTD mit einer Rampe in ER_LM überführt. Die Steilheit der Rampe ist durch den Parameter TM_RAMP einstellbar.

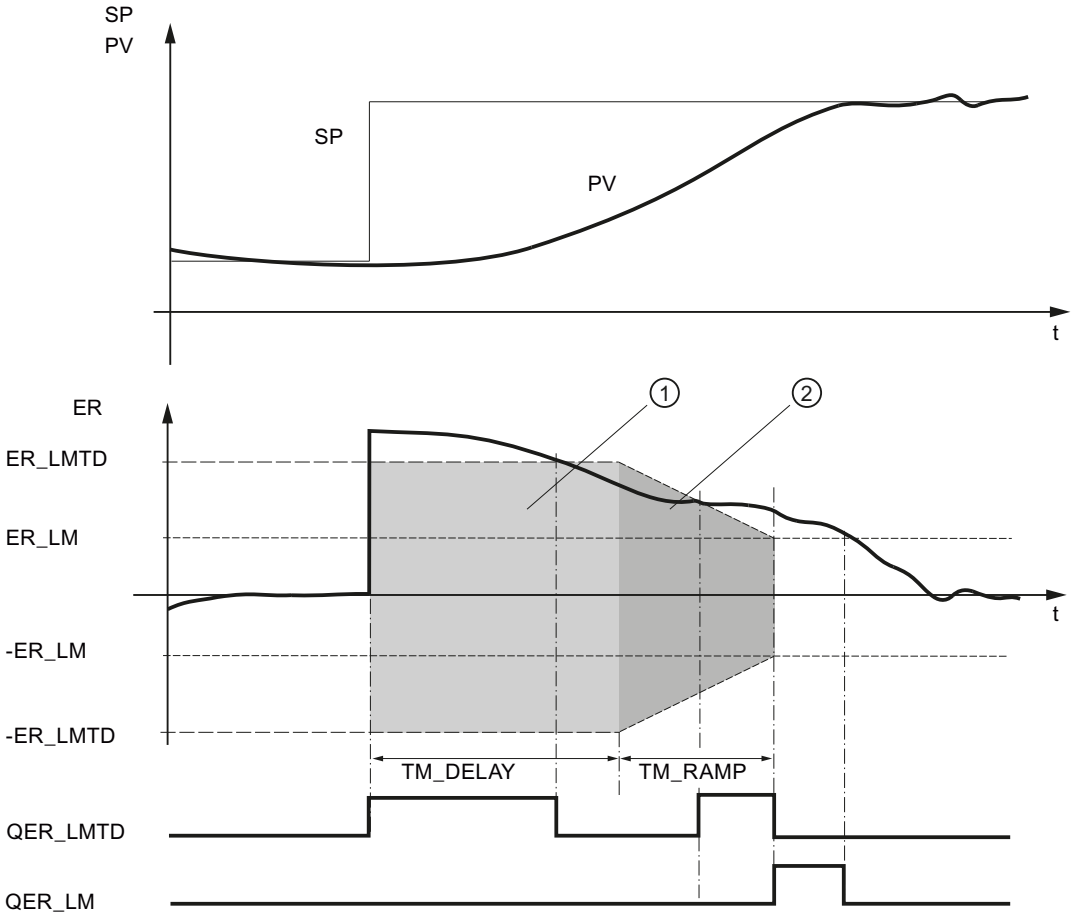

PV:	Istwert	TM_DELAY:	Einschaltverzögerung des Überwachungssignals
SP:	Sollwert		
ER:	Regeldifferenz	TM_RAMP:	Rampenzeitkonstante
ER_LM:	Grenzwert der Regeldifferenz	QER_LM:	Grenzwert der Regeldifferenz angesprochen
ER_LMTD:	Grenzwert der Regeldifferenz während Einschaltverzögerung	QER_LMTD:	Grenzwert der Regeldifferenz während Einschaltverzögerung und Rampe angesprochen
①	Unterdrücken des Grenzwertsignals	②	Rampenfunktion

Neustart

Bei Neustart werden die Meldesignale QER_LM und QER_LMTD und der Regeldifferenzausgang ER zurückgesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter ERR_MON

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
ER_LM	0.0	REAL	10.0	Grenzwert der Regeldifferenz technischer Wertebereich und > 0.0 und < ER_LMTD
ER_LMTD	4.0	REAL	100.0	Grenzwert der Regeldifferenz während Einschaltverzögerung technischer Wertebereich und > ER_LM
SP	8.0	REAL	0.0	Sollwert technischer Wertebereich
PV	12.0	REAL	0.0	Regelgröße technischer Wertebereich
SP_DIFF	16.0	REAL	10.0	Sollwertänderung technischer Wertebereich und > 0.0
TM_DELAY	20.0	TIME	T#60s	Einschaltverzögerung des Überwachungssignals
TM_RAMP	24.0	TIME	T#60s	Rampenzeitkonstante
COM_RST	28.0	BOOL	FALSE	Neustart
CYCLE	30.0	TIME	T#1s	Abtastzeit ≥ 1ms

Ausgangsparameter ERR_MON

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
QER_LM	34.0	BOOL	FALSE	Grenzwert der Regeldifferenz angesprochen
QER_LMTD	34.1	BOOL	FALSE	Grenzwert der Regeldifferenz während der Einschaltverzögerung und der Rampe angesprochen
ER	36.0	REAL	0.0	Regeldifferenz

INTEG

Beschreibung INTEG

Prozessgrößen werden dynamisch integriert. So wird z.B. aus der Geschwindigkeit der zurückgelegte Weg berechnet. Die Anweisung ist einsetzbar zum Aufbau eines Reglers mit I-Anteil.

Arbeitsweise

Die Anweisung integriert die Eingangsgröße über der Zeit und begrenzt das Integral auf einen vorgebbaren oberen und unteren Wert. Die Begrenzung der Ausgangsgröße wird über Meldebits angezeigt.

Die Anweisung enthält die folgenden Funktionen:

Funktionen	DFOUT_ON	HOLD
Integrieren	FALSE	FALSE
I-Anteil einfrieren	FALSE	TRUE
Ausgang vorbelegen	TRUE	beliebig

- **Integrieren**
  Beim Integrieren arbeitet die Anweisung nach folgender Übertragungsfunktion:
  im Laplace-Bereich:
  OUTV(s) / INV(s) = 1 / (TI*s)
  Das Zeitverhalten des I-Anteils wird durch die Integrationszeit TI festgelegt. Die zugehörige Sprungantwort ist im folgenden Bild dargestellt.

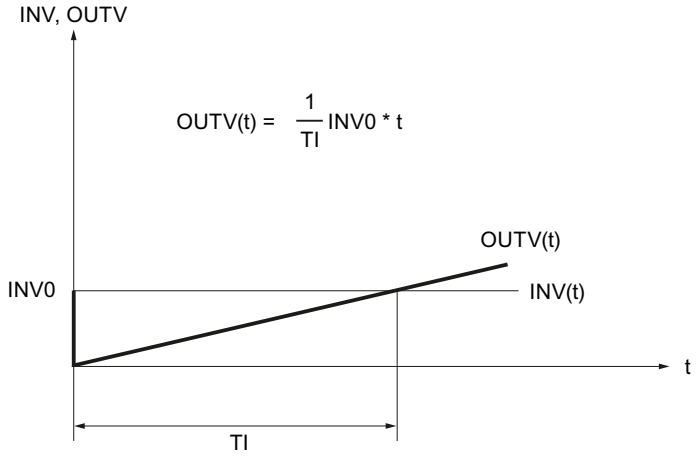

T1:	Integrationszeit
INV0:	Eingangssprunghöhe
t:	Zeit
INV:	Eingangsgröße
OUTV:	Ausgangsgröße

Der Ausgang wie auch der Speicher des I-Anteils werden an den Parametern H_LM und L_LM begrenzt. Liegt der Ausgang außerhalb des Bereichs L_LM...H_LM werden die Meldebits QH_LM oder QL_LM auf TRUE gesetzt.

- Integrator einfrieren
  Mit HOLD = TRUE bleibt der I-Anteil auf seinem momentanen Ausgangswert OUTV stehen. Beim Rücksetzen HOLD = FALSE wird ausgehend vom momentanen Ausgangswert OUTV weiterintegriert.
- Ausgang vorbelegen
  Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Die Begrenzung ist wirksam. Wird DF_OUTV_ON = FALSE zurückgesetzt, beginnt INTEG ausgehend vom Wert DF_OUTV zu integrieren.

Das folgende Bild zeigt Ihnen ein Beispiel mit DFOUT_ON, HOLD und Begrenzung

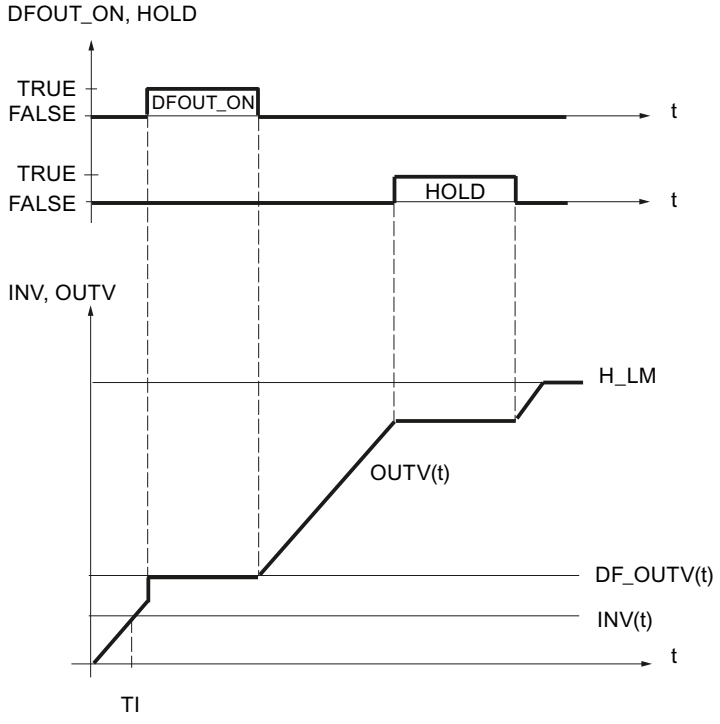

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist, wird DF_OUTV ausgegeben. Die Begrenzung des Ausgangs und deren Anzeige sind auch bei Neustart wirksam. Beim Übergang in den Normalbetrieb integriert die Anweisung von OUTV aus weiter.

Soll der I-Anteil bei Neustart von einem bestimmten Arbeitspunkt aus gestartet werden, so muss am Eingang DF_OUTV der Arbeitspunkt eingetragen werden. Beim Aufruf der Anweisung in der Neustartroutine muss DFOUT_ON= TRUE gesetzt werden und in der Weckalarmebene auf DFOUT_ON= FALSE wieder zurückgesetzt werden.

Bausteininterne Begrenzungen

Die Integrationszeit wird auf die Abtastzeit nach unten begrenzt:

TI_{intern} = CYCLE für TI < CYCLE

Die anderen Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter INTEG

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
H_LM	4.0	REAL	100.0	obere Begrenzung technischer Wertebereich H_LM > L_LM
L_LM	8.0	REAL	0.0	untere Begrenzung technischer Wertebereich L_HM < H_LM
TI	12.0	TIME	T#25s	Integrationszeit TI ≥ CYCLE
DF_OUTV	16.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
HOLD	20.0	BOOL	FALSE	I-Anteil einfrieren
DFOUT_ON	20.1	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
COM_RST	20.2	BOOL	FALSE	Neustart
CYCLE	22.0	TIME	T#1s	Abtastzeit CYCLE ≥ 1ms

Ausgangsparameter INTEG

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	26.0	REAL	0.0	Ausgangsgröße
QH_LM	30.0	BOOL	FALSE	obere Begrenzung angesprochen
QL_LM	30.1	BOOL	FALSE	untere Begrenzung angesprochen

LAG1ST

Beschreibung LAG1ST

Die Anweisung ist einsetzbar als Verzögerungs- und Glättungsglied 1. Ordnung. Die Verzögerungszeit ist parametrierbar.

Arbeitsweise

Die Anweisung glättet die Eingangsgröße nach der Verzögerung 1. Ordnung und enthält folgende Funktionen:

Funktionen	DFOUT_ON	TRACK
Glätten	FALSE	FALSE
Nachführen	FALSE	TRUE
Ausgang vorbelegen	TRUE	beliebig

- **Glätten**
  Beim Glätten arbeitet die Anweisung im Laplace-Bereich nach folgender Übertragungsfunktion:
  OUTV(s) / INV(s) = 1 / (1+TM_LAG*s)
  Das Zeitverhalten des Verzögerungsgliedes wird durch die Verzögerungszeit TM_LAG festgelegt. Die zugehörige Sprungantwort ist im folgenden Bild dargestellt.

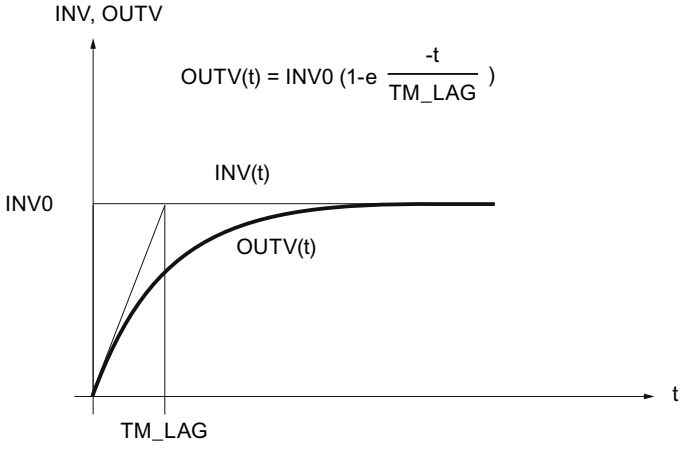

INV	Eingangsgröße
OUTV	Ausgangsgröße
INV0	Eingangssprunghöhe
TM_LAG	Verzögerungszeit

- **Nachführen**
  Mit TRACK = TRUE wird der Eingangswert INV auf den Ausgang OUTV durchgeschaltet.
- **Ausgang vorbelegen**
  Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Wird DF_OUTV_ON = FALSE zurückgesetzt, glättet das Verzögerungsglied ausgehend vom Wert DF_OUTV.
  Das nachfolgende Bild zeigt die Wirkung von DFOUT_ON und TRACK:

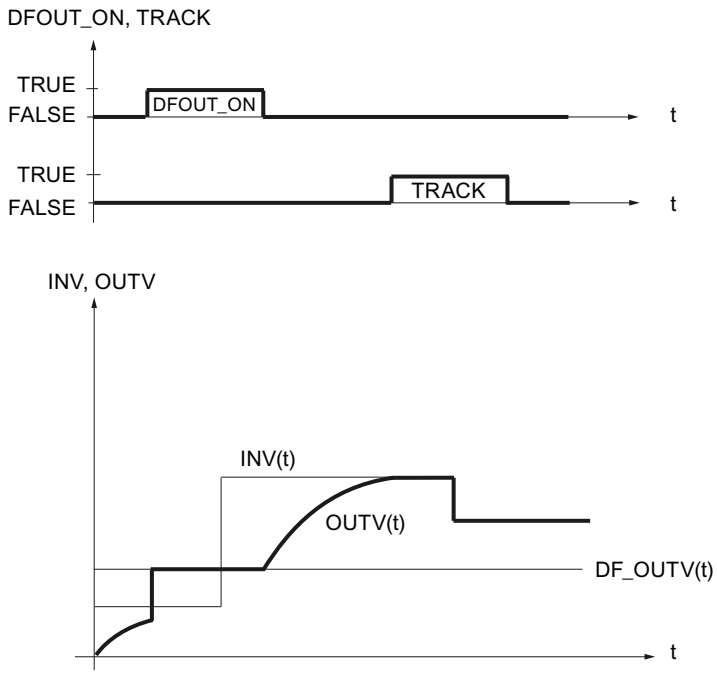

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist, wird DF_OUTV ausgegeben. Beim Übergang in das Glätten arbeitet die Anweisung von OUTV aus weiter.

Bausteininterne Begrenzungen

Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

TM_LAG_{intern} = CYCLE/**2** für TM_LAG < CYCLE/**2**

Die anderen Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter LAG1ST

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
TM_LAG	4.0	TIME	T#25s	Verzögerungszeit
DF_OUTV	8.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
TRACK	12.0	BOOL	FALSE	Nachführen OUTV=INV
DFOUT_ON	12.1	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
COM_RST	12.2	BOOL	FALSE	Neustart
CYCLE	14.0	TIME	T#1s	Abtastzeit ≥1ms

Ausgangsparameter LAG1ST

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	18.0	REAL	0.0	Ausgangsgröße

LAG2ND

Beschreibung LAG2ND

Die Anweisung dient zur Simulation von Streckenanteilen für Vorsteuerungsregelungen und Zweikreisregelungen.

Arbeitsweise

Die Anweisung realisiert ein schwingungsfähiges Verzögerungsglied 2. Ordnung und enthält folgende Funktionen:

Funktionen	DFOUT_ON	TRACK
Übertragen	FALSE	FALSE
Nachführen	FALSE	TRUE
Ausgang vorbelegen	TRUE	beliebig

- **Übertragen**
  Die Übertragungsfunktion im Laplace-Bereich ist:
  OUTV(s) / INV(s) = TRANCOEF / (1 + 2*DAM_COEF*TM_CONST*s + TM_CONST^2*s2)
  Wenn DAM_COEF >= 1 ist (aperiodischer Fall), lässt sich das Übertragungsglied in einer Reihenschaltung zweier PT1-Glieder darstellen.
  OUTV(s) / INV(s) = TRANCOEF / (1 + T1*s) * 1 / (1 + T2 * s)
  Die Zeitkonstanten werden wie folgt umgerechnet:

  $T1 = TM_CONST\ (DAM_COEF + \sqrt{DAM_COEF^2 - 1})$

  $T2 = TM_CONST\ (DAM_COEF - \sqrt{DAM_COEF^2 - 1})$

  Das folgende Bild zeigt das Blockschaltbild

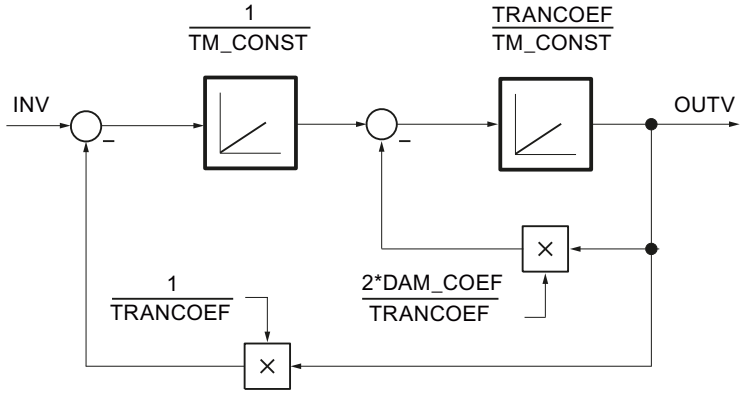

  Das folgende Bild zeigt die Sprungantwort des schwingungsfähigen LAG2ND-Gliedes.

$$OUTV(t) = TRANCOEF + \frac{TRANCOEF}{\sqrt{1-DAM_COEF^2}} \cdot e^{-\frac{DAM_COEF}{TM_CONST} t} \sin\left(\frac{\sqrt{1-DAM_COEF^2}}{TM_CONST} t - phi\right)$$

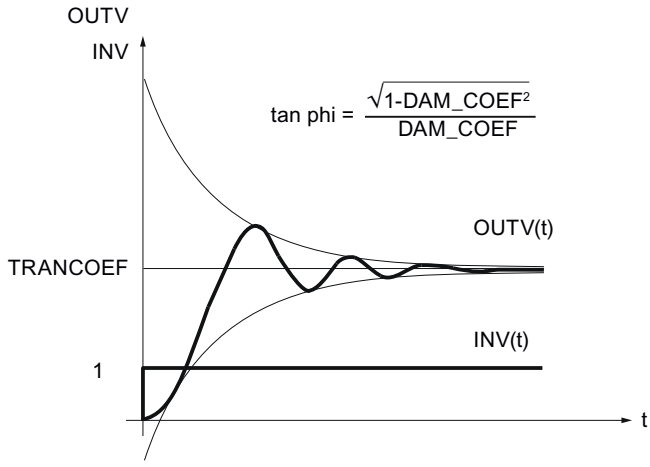

INV	Eingangsgröße
OUTV	Ausgangsgröße
TRANCOEF	Übertragungskonstante
DAM_COEF	Dämpfungskonstante
phi	Supplementwinkel

- **Nachführen**
  Ist TRACK =TRUE, wird OUTV = INV; die internen Vergangenheitswerte werden auf INV gesetzt.
- **Ausgang vorbelegen**
  Ist DFOUT_ON =TRUE, wird DF_OUTV ausgegeben, die internen Vergangenheitswerte werden auf DF_OUTV gesetzt. DFOUT_ON hat höhere Priorität als TRACK.

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 gesetzt. Ist DFOUT_ON = TRUE, wird DF_OUTV ausgegeben.

Bausteininterne Begrenzungen

Die Zeitkonstante TM_CONST wird auf die halbe Abtastzeit nach unten begrenzt.

$TM_CONST_{intern} = CYCLE/2$ für $TM_CONST < CYCLE/2$

Die anderen Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter LAG2ND

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
TM_CONST	4.0	TIME	T#10s	Zeitkonstante
DAM_COEF	8.0	REAL	1.0	Dämpfungskonstante
TRANCOEF	12.0	REAL	1.0	Übertragungskonstante
DF_OUTV	16.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
DFOUT_ON	20.0	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
TRACK	20.1	BOOL	FALSE	Nachführen OUTV=INV
COM_RST	20.2	BOOL	FALSE	Neustart
CYCLE	22.0	TIME	T#1s	Abtastzeit ≥ 1ms

Ausgangsparameter LAG2ND

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	26.0	REAL	0.0	Ausgangsgröße

LIMALARM

Beschreibung LIMALARM

Bei Regelungen können unzulässige oder gefährliche Betriebszustände auftreten, wenn Prozessgrößen (z.B. Drehzahl, Druck, Temperatur, ...) über-/unterschritten werden. Die Über-/Unterschreitung von zulässigen Grenzen muss erfasst und gemeldet werden, damit eine sinnvolle Reaktion erfolgen kann.

Die Anweisung überprüft die Eingangsgröße INV auf 4 vorgebbare Grenzen. Bei Erreichen und Überschreiten dieser Grenzen wird ein zugehöriges Grenzsignal ausgegeben. Die Ausschaltschwelle der Grenzsignale ist über eine Hysterese einstellbar.

Arbeitsweise

An den Eingängen H_LM_ALM, H_LM_WRN, L_LM_WRN und L_LM_ALM werden die Grenzen eingestellt. Überschreitet die Eingangsgröße INV die Grenzen werden die Meldebits QH_LMALM, QH_LMWRN, QL_LMWRN und QL_LMALM gesetzt. Um ein schnelles Ein-/Ausschalten der Meldebits zu vermeiden, muss der Eingangswert zusätzlich eine Hysterese HYS überwinden, bevor die Ausgänge rückgesetzt werden.

Das folgende Bild zeigt die Funktionsweise der Anweisung LIMALARM:

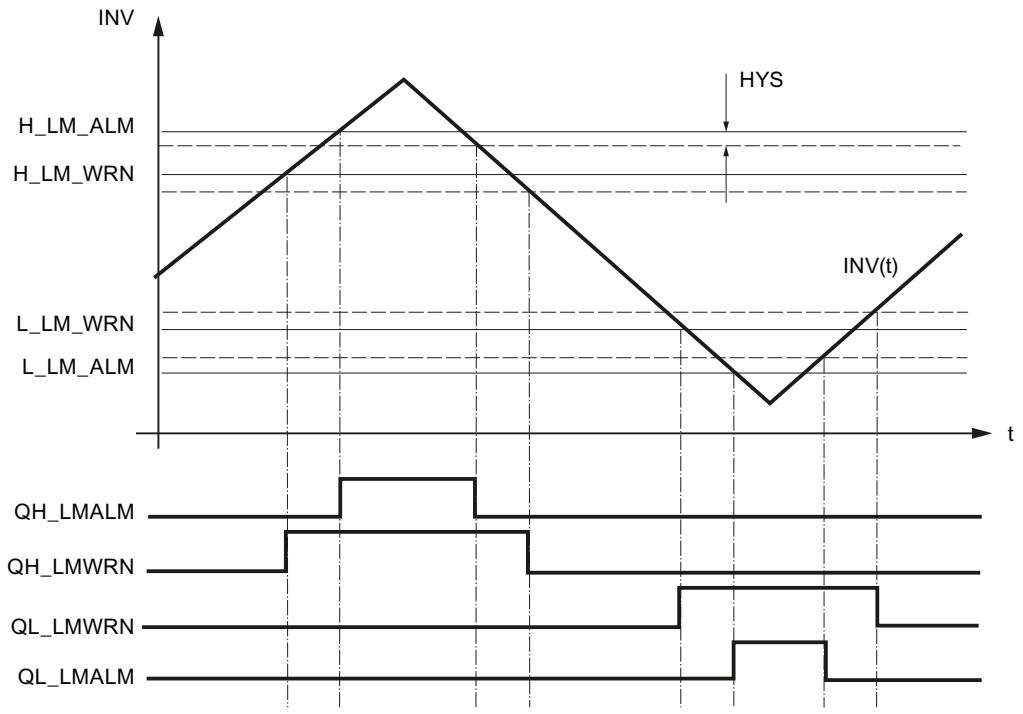

QH_LMALM = TRUE	wenn	INV steigt	und	INV >= H_LM_ALM
	oder	INV sinkt	und	INV >= H_LM_ALM - HYS
QH_LMWRN = TRUE	wenn	INV steigt	und	INV >= H_LM_WRN
	oder	INV sinkt	und	INV >= H_LM_WRN - HYS
QL_LMWRN = TRUE	wenn	INV sinkt	und	INV <= L_LM_WRN
	oder	INV steigt	und	INV <= L_LM_WRN + HYS
QL_LMALM = TRUE	wenn	INV sinkt	und	INV <= L_LM_ALM
	oder	INV steigt	und	INV <= L_LM_ALM + HYS

Für ein sinnvolles Arbeiten der Anweisung muss gelten:

L_LM_ALM < L_LM_WRN < H_LM_WRN < H_LM_ALM

Neustart

Bei Neustart werden alle Signalausgänge auf FALSE gesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter LIMALARM

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
H_LM_ALM	0.0	REAL	100.0	oberer Grenzwert Alarm technischer Wertebereich > H_LM_WRN
H_LM_WRN	4.0	REAL	90.0	oberer Grenzwert Warnung technischer Wertebereich > L_LM_WRN
L_LM_WRN	8.0	REAL	10.0	unterer Grenzwert Warnung technischer Wertebereich > L_LM_ALM
L_LM_ALM	12.0	REAL	0.0	unterer Grenzwert Alarm technischer Wertebereich < L_LM_WRN
INV	16.0	REAL	0.0	Eingangsgröße technischer Wertebereich
HYS	20.0	REAL	1.0	Hysterese technischer Wertebereich
COM_RST	24.0	BOOL	FALSE	Neustart

Ausgangsparameter LIMALARM

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
QH_LMALM	26.0	BOOL	FALSE	oberer Grenzwert Alarm angesprochen
QH_LMWRN	26.1	BOOL	FALSE	oberer Grenzwert Warnung angesprochen
QL_LMWRN	26.2	BOOL	FALSE	unterer Grenzwert Warnung angesprochen
QL_LMALM	26.3	BOOL	FALSE	unterer Grenzwert Alarm angesprochen

LIMITER

Beschreibung LIMITER

Bei dynamischer Vorgabe von Parametern (z.B. Berechnung von Sollwerten aus Prozessgrößen) können diese für den Prozess unzulässige Werte annehmen. Mit der Anweisung LIMITER können sie im zulässigen Bereich gehalten werden.

Arbeitsweise

Die Anweisung begrenzt die Ausgangsgröße OUTV auf eine vorgebbare obere und untere Begrenzung H_LM und L_LM, wenn die Eingangsgröße INV außerhalb dieser Begrenzungen liegt. Die Begrenzung von OUTV wird über die Ausgänge QH_LM und QL_LM gemeldet.

Das folgende Bild zeigt Ihnen die Funktionsweise der Anweisung LIMITER:

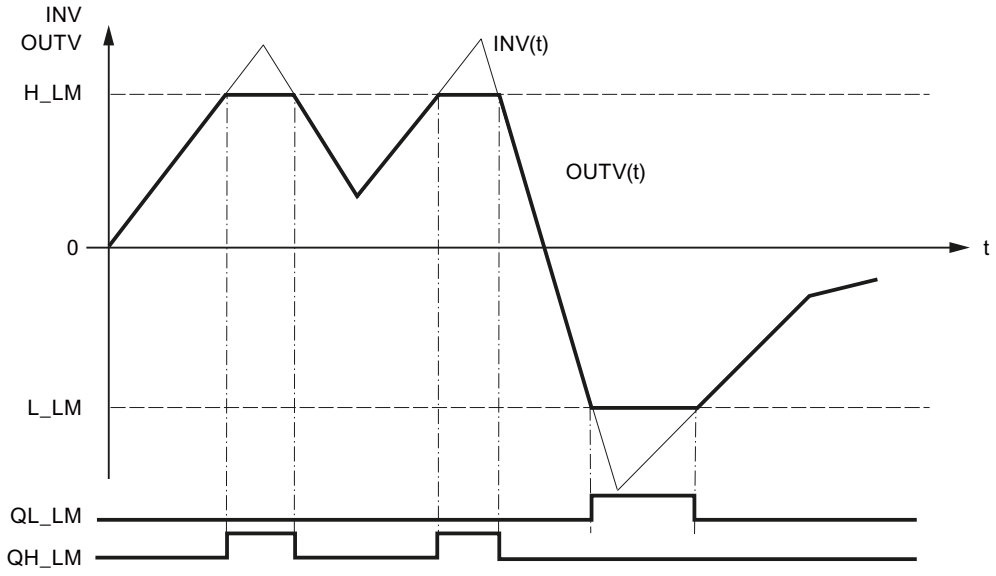

INV	OUTV	QH_LM	QL_LM
≥ H_LM	H_LM,	TRUE	FALSE
≤ L_LM	L_LM,	FALSE	TRUE
≤ L_LM und ≤H_LM	INV	FALSE	FALSE

Für ein sinnvolles Arbeiten der Anweisung muss gelten:

L_LM < H_LM

Neustart

Bei Neustart werden alle Signalausgänge auf FALSE gesetzt; am Ausgang OUTV wird 0.0 ausgegeben.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter LIMITER

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
H_LM	4.0	REAL	100.0	obere Begrenzung technischer Wertebereich > L_LM
L_LM	8.0	REAL	0.0	untere Begrenzung technischer Wertebereich < H_LM
COM_RST	12.0	BOOL	FALSE	Neustart

Ausgangsparameter LIMITER

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	14.0	REAL	0.0	Ausgangsgröße
QH_LM	18.0	BOOL	FALSE	obere Begrenzung angesprochen
QL_LM	18.1	BOOL	FALSE	untere Begrenzung angesprochen

LMNGEN_C

Beschreibung LMNGEN_C

Die Anweisung dient zum Aufbau eines kontinuierlichen PID-Reglers und ermittelt den Stellwert des Reglers. Der Stellwert kann im Automatikbetrieb ermittelt werden oder im Handbetrieb vorgegeben werden. Der Handwert kann absolut vorgegeben oder per Schalterbetrieb vergrößert und verkleinert werden. Der Stellwert und die Steigung des Stellwertes können auf vorgebbare Werte begrenzt werden.

Neustart

Bei Neustart wird der Vorbelegungswert DF_OUTV unabhängig vom Vorbelegungsbit DFOUT_ON auf den Ausgang LMN durchgeschaltet. Die Begrenzung des Ausgangs und deren Anzeige ist auch bei Neustart wirksam. Beim Übergang in den Normalbetrieb arbeitet die Anweisung von DF_OUTV aus weiter.

Die Ausgangsparameter QLMN_HLM und QLMN_LLM werden FALSE gesetzt.

Aufruf

Während die Anweisung PID in einer Weckalarmebene aufgerufen wird, deren Zykluszeit der dominierenden Streckenzeitkonstanten angepasst ist, kann die Anweisung LMNGEN_C für manuelle Eingriffe in einer schnelleren Weckalarmebene aufgerufen werden. Die Verschaltung erfolgt mit den strukturierten Ein-/Ausgangsparameter PID_LMNG und LMNG_PID.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Arbeitsweise

Die Anweisung LMNGEN_C wird immer in Verbindung mit der Anweisung PID verwendet.

Das folgende Bild zeigt Ihnen die Verschaltung des kontinuierlichen PID–Reglers.

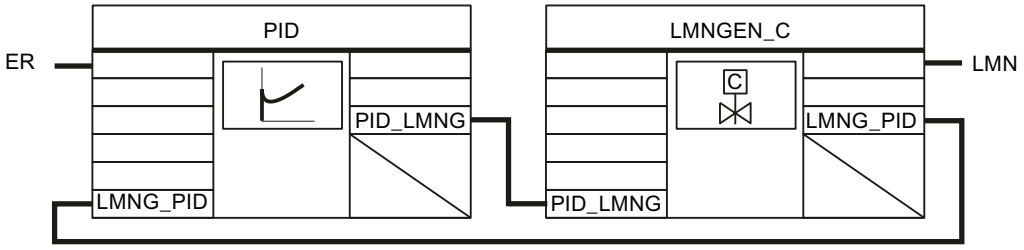

Arbeitsweise LMNGEN_C

Stellwert begrenzen

Der Stellwert wird in allen Betriebsarten begrenzt. Sie geben die obere Grenzen des Stellwerts am Eingang LMN_HLM und die untere Grenze am Eingang LMN_LLM ein. Wenn der berechnete Stellwert diese Werte über- oder unterschreitet, wird an LMN die Ober- oder Untergrenze ausgegeben. Die Begrenzung des Stellwerts auf die Obergrenze wird am Ausgang QLMN_HLM angezeigt, die Begrenzung auf die Untergrenze am Ausgang QLMN_LLM.

Stellwertrampe

Mit der Stellwertrampe begrenzen Sie die Änderungsgeschwindigkeit des Stellwerts. Die Stellwertrampe aktivieren Sie mit LMNRC_ON=TRUE. Am Eingangsparameter LMN_URLM geben Sie die maximale Änderungsgeschwindigkeit für den Anstieg des Stellwerts und am Eingangsparameter LMN_DRLM die maximale Änderungsgeschwindigkeit für den Abstieg des Stellwerts ein.

Betriebsarten

Die Betriebsarten werden mit den Parametern DFOUT_ON, MAN_ON und MANGN_ON ein- und ausgeschaltet.

Betriebsarten	DFOUT_ON	MAN_ON	MANGN_ON
Automatikbetrieb	FALSE	FALSE	FALSE
Handbetrieb	FALSE	TRUE	FALSE
Handwertgenerator	FALSE	TRUE	TRUE
Stellwert vorbelegen	TRUE	nicht relevant	nicht relevant

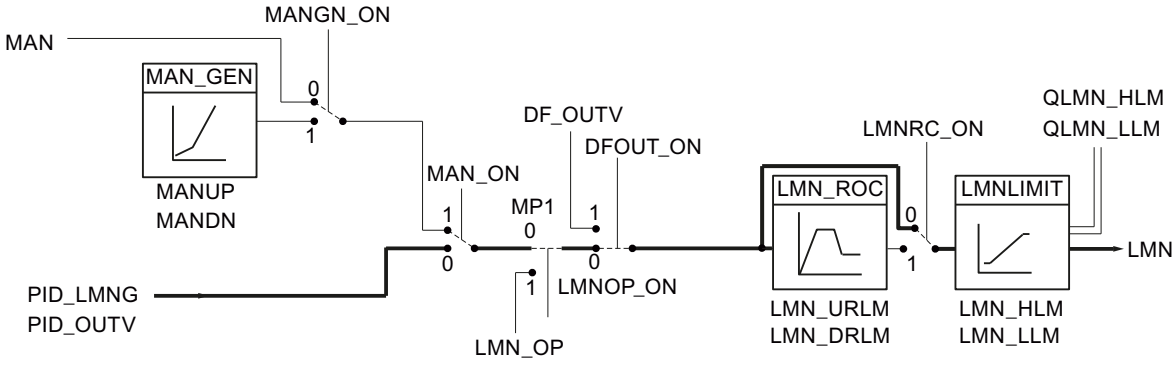

- **Automatikbetrieb**
  Der vom PID-Algorithmus ermittelte Wert wird für die Ermittlung des Stellwerts LMN verwendet. Die Umschaltung in den Automatikbetrieb ist stoßfrei, wenn die Stellwertrampe (LMNRC_ON = TRUE) eingeschaltet ist.
- **Handbetrieb**
  Mit MAN_ON = TRUE schalten Sie in den Handbetrieb und unterbrechen damit den Regelkreis. Der Handwert MAN wird direkt als Stellwert LMN ausgegeben. MAN muss innerhalb der Grenzen LMN_HLM und LMN_LLM liegen.
- **Handwertgenerator**
  Wenn zusätzlich MANGN_ON = TRUE ist, kann der Stellwert über die Schalter MANUP und MANDN vergrößert bzw. verkleinert werden. Die Änderungsgeschwindigkeit des Stellwerts hängt von den Begrenzungen ab:
  - während der ersten 3 s nach Setzen von MANUP oder MANDN
    dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
  - danach
    dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s
- **Stellwert vorbelegen**
  Mit DFOUT_ON = TRUE wird der Vorbelegungswert DF_OUTV am Ausgang LMN übernommen. Die Umschaltung von bzw. nach "Stellwert vorbelegen" ist nur stoßfrei, wenn die Stellwertrampe eingeschaltet ist. (LMNRC_ON= TRUE).

Bedienung über HMI

Mit statischen Variabeln können Sie den Stellwert über ein HMI beeinflussen (siehe nachfolgendes Bild). Die statischen Variabeln LMN_OP, LMNOP_ON und MP1 dürfen z. B. über HMI bedient und beobachtet werden.

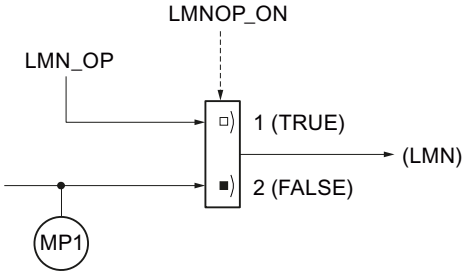

Wenn LMNOP_ON auf TRUE gesetzt wird, wird LMN_OP auf den Stellwert LMN durchgeschaltet. MP1 enthält den Stellwert, der von der Anweisung LMNGEN_C verarbeitet wird.

Wenn die Stellwertrampe eingeschaltet ist, kann stoßfrei zwischen den Werten MP1 und LMN_OP umgeschaltet werden. LMN wird entsprechend der Rampeneinstellung auf den Wert MP1 zurückgeführt.

Geänderte statische Variablen wirken erst dann auf den Prozess, wenn sie an die CPU übertragen wurden.

Eingangsparameter LMNGEN_C

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN_HLM	0.0	REAL	100.0	Stellwert obere Begrenzung technischer Wertebereich > LMN_LLM
LMN_LLM	4.0	REAL	0.0	Stellwert untere Begrenzung technischer Wertebereich < LMN_HLM
LMN_URLM	8.0	REAL	10.0	Stellwert Anstiegsbegrenzung [1/s] > 0.0
LMN_DRLM	12.0	REAL	10.0	Stellwert Abstiegsbegrenzung [1/s] > 0.0
DF_OUTV	16.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
MAN_ON	20.0	BOOL	TRUE	Handbetrieb einschalten
MANGN_ON	20.1	BOOL	FALSE	Handwertbedienung einschalten
MANUP	20.2	BOOL	FALSE	Handwert hoch
MANDN	20.3	BOOL	FALSE	Handwert tief
LMNRC_ON	20.4	BOOL	FALSE	Stellwertrampe einschalten
DFOUT_ON	20.5	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
COM_RST	20.6	BOOL	FALSE	Neustart
CYCLE	22.0	TIME	T#1s	Abtastzeit ≥ 1ms
PID_LMNG	26.0	STRUC		PID-LMNGEN Schnittstelle

Ausgangsparameter LMNGEN_C

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN	44.0	REAL	0.0	Stellwert
QLMN_HLM	48.0	BOOL	FALSE	obere Begrenzung des Stellwertes angesprochen
QLMN_LLM	48.1	BOOL	FALSE	untere Begrenzung des Stellwertes angesprochen
LMNG_PID	50.0	STRUC		PID-LMNGEN Schnittstelle

Durchgangsparameter LMNGEN_C

Tabelle 4-225

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
MAN	68.0	REAL	0.0	Handwert technischer Wertebereich

Statische Variablen LMNGEN_C

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
MP1	72.0	REAL	0.0	Stellwert, der von LMNGEN_C verarbeitet wird.
LMM_OP	76.0	REAL	0.0	Bedienung über HMI Handwert
LMNOP_ON	80.0	BOOL	FALSE	Bedienung über HMI Handbetrieb einschalten

LMNGEN_S

Beschreibung LMNGEN_S

Die Anweisung dient zum Aufbau eines PID-Schrittreglers für Stellglieder mit integrierendem Verhalten (z.B. motorgetriebene Ventile) und ermittelt den Stellwert des Reglers. Der Stellwert kann im Automatikbetrieb ermittelt werden oder im Handbetrieb vorgegeben werden. Der Handwert kann absolut vorgegeben oder per Schalterbetrieb vergrößert und verkleinert werden. Der Stellwert kann auf vorgebbare Werte begrenzt werden. Der Schrittregler kann mit oder ohne Stellungsrückmeldung arbeiten.

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt.

Aufruf

Während die Anweisung PID in einer Weckalarmebene aufgerufen wird, deren Zykluszeit der dominierenden Streckenzeitkonstanten angepasst ist, kann die Anweisung LMNGEN_S für manuelle Eingriffe in einer schnelleren Weckalarmebene aufgerufen werden. Die Verschaltung erfolgt mit den strukturierten Ein-/Ausgangsparameter PID_LMNG und LMNG_PID.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Arbeitsweise

Die Anweisung LMNGEN_S wird immer in Verbindung mit der Anweisung PID verwendet.

Das folgende Bild zeigt Ihnen die Verschaltung eines Schrittreglers.

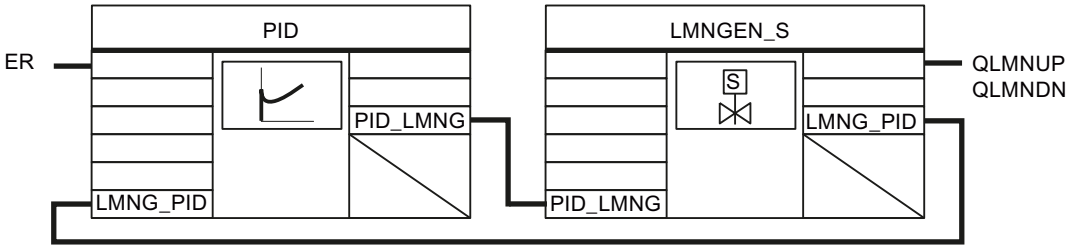

Arbeitsweise LMNGEN_S

Stellwert begrenzen

Der Stellwert wird in allen Betriebsarten begrenzt. Sie geben die obere Grenzen des Stellwerts am Eingang LMN_HLM und die untere Grenze am Eingang LMN_LLM ein. Wenn der berechnete Stellwert diese Werte über- oder unterschreitet, wird an LMN die Ober- oder Untergrenze ausgegeben. Die Begrenzung des Stellwerts auf die Obergrenze wird am Ausgang QLMN_HLM angezeigt, die Begrenzung auf die Untergrenze am Ausgang QLMN_LLM.

Anschlagsignale

An den Eingangsparametern LMNR_HS und LMNR_LS verschalten Sie die Signale für den oberen und unteren Anschlag des Stellglieds. Wenn LMNR_HS= TRUE, wird das Stellglied nicht mehr weiter geöffnet. Wenn LMNR_LS= TRUE, wird das Stellglied nicht mehr weiter geschlossen. Die Ausgangssignale QLMNUP und QLMNDN sind gesperrt. Die Endanschläge werden in allen Betriebsarten berücksichtig.

⚠ WARNUNG

Wenn keine Anschlagssignale vorhanden sind, kann der Regler einen Ventilanschlag nicht erkennen und es besteht z.B. die Möglichkeit, dass der Regler Signale zum Öffnen des Ventils ausgibt, obwohl das Ventil sich am oberen Anschlag befindet. Sind keine Anschlagsignale vorhanden, müssen die Eingänge LMNR_HS und LMNR_LS mit FALSE belegt werden.

Schrittregler mit Stellungsrückmeldung

Wenn eine Stellungsrückmeldung zur Verfügung steht, bildet die Anweisung aus der Differenz zwischen dem Stellwert und der Stellungsrückmeldung über ein Dreipunktglied und Impulsformer die Impulse zur Ansteuerung des Stellventils. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Sie aktivieren die Stellungsrückmeldung mit LMNR_ON = TRUE.

Die

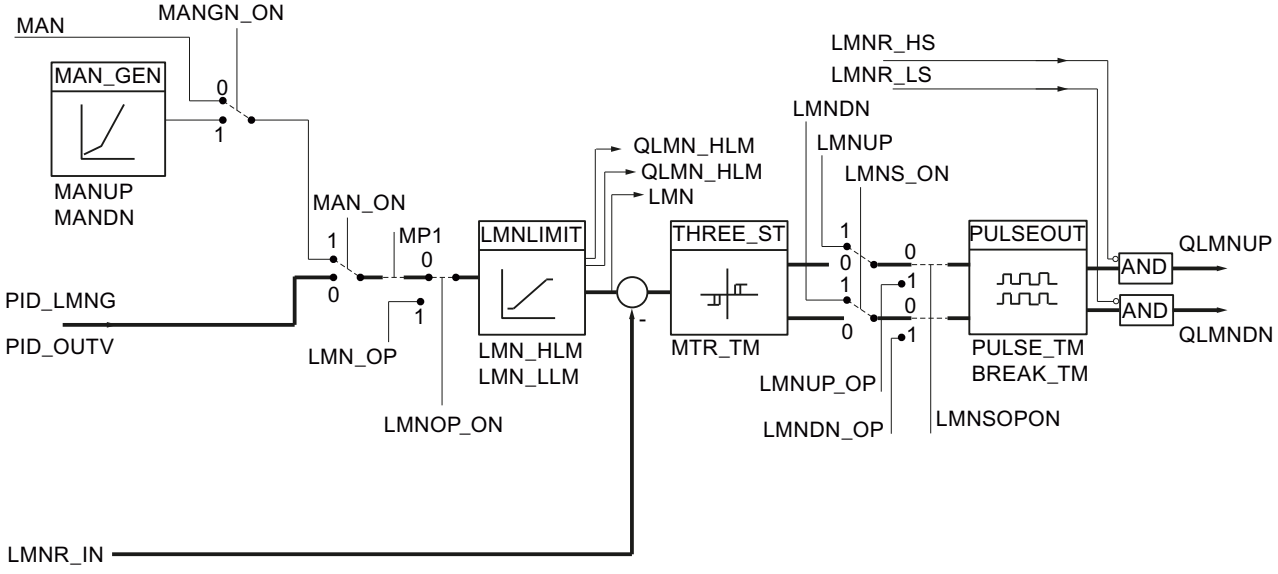

Schrittregler ohne Stellungsrückmeldung

Wenn keine Stellungsrückmeldung zur Verfügung steht, summiert die Anweisung die Differenz aus dem I-Anteil des PID-Algorithmus und einer simulierten Stellungsrückmeldung in einem Integrator und vergleicht diese als Rückführungswert mit dem verbliebenen PD-Anteil. Die Differenz geht wiederum auf das Dreipunktglied und den Impulsformer, der die Impulse für das Stellventil bildet. Über eine Adaption der Ansprechschwelle des Dreipunktgliedes wird die Schalthäufigkeit des Reglers reduziert.

Sie deaktivieren die Stellungsrückmeldung mit LMNR_ON = FALSE.

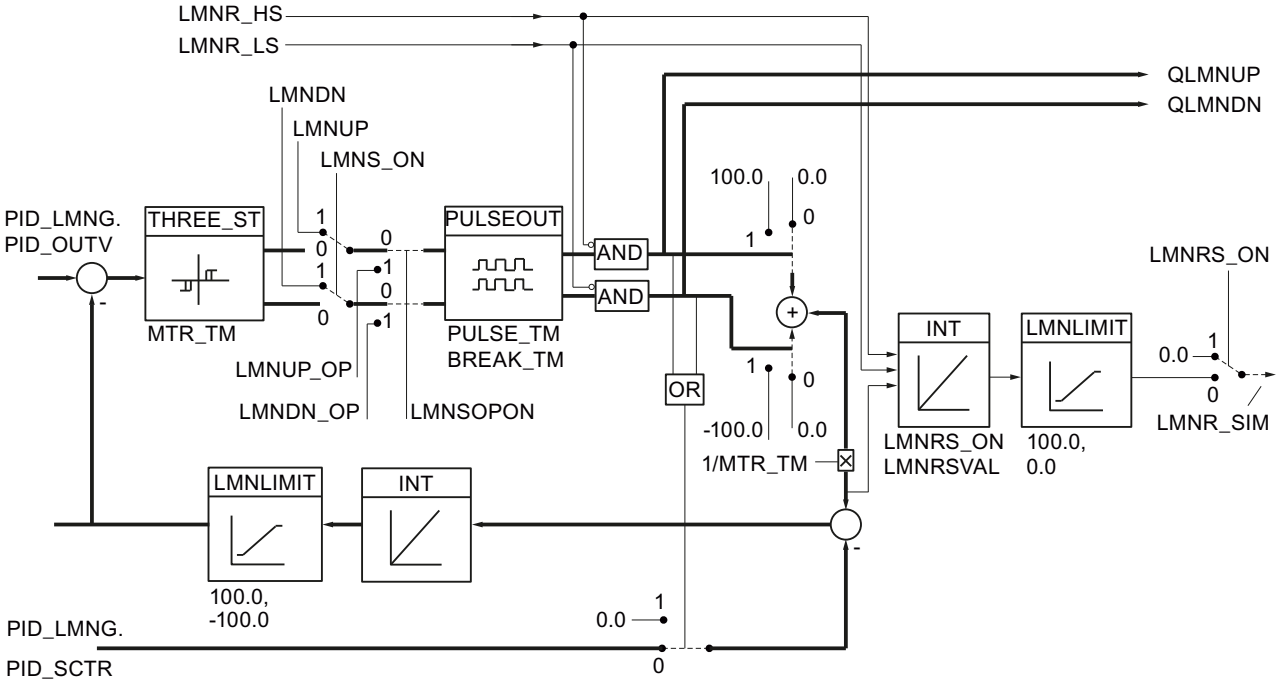

Betriebsarten

Die Betriebsarten werden mit den Parametern LMN_ON, MAN_ON und MANGN_ON ein- und ausgeschaltet.

Betriebsarten	LMN_ON	MAN_ON	MANGN_ON
Automatikbetrieb	FALSE	FALSE	FALSE
Handwert vorgeben	FALSE	TRUE	FALSE
Handwertgenerator	FALSE	TRUE	TRUE
Handbetrieb der Stellwertausgangssignale	TRUE	nicht relevant	nicht relevant

Die Betriebsarten Handwert vorgeben und Handwertgenerator sind nur bei Schrittreglern mit Stellungsrückmeldung möglich. Da bei einem Schrittregler ohne Stellungsrückmeldung keine

Information über die Ventilstellung vorliegt, kann das Stellglied nicht auf einen festen Handwert gefahren werden.

- Handwert vorgeben
  Mit MAN_ON = TRUE schalten Sie in den Handbetrieb und unterbrechen damit den Regelkreis. Der Handwert MAN wird direkt als Stellwert durchgeschaltet. MAN muss innerhalb der Grenzen LMN_HLM und LMN_LLM liegen.
- Handwertgenerator
  Wenn zusätzlich MANGN_ON = TRUE ist, kann der Stellwert über die Schalter MANUP und MANDN vergrößert bzw. verkleinert werden. Die Änderungsgeschwindigkeit des Stellwerts hängt von den Begrenzungen ab:
  - während der ersten 3 s nach Setzen von MANUP oder MANDN
    dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
  - danach
    dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s
- Handbetrieb der Stellwertausgangssignale
  Wenn LMNS_ON = TRUE, können die binären Ausgangssignale direkt beeinflusst werden. Über die Schalter LMNUP und LMNDN werden die Stellsignalausgänge QLMNUP und QLMNDN gesetzt. Die Mindestimpulszeit PULSE_TM und Mindestpausendauer BREAK_TM werden mitberücksichtigt.
- Automatikbetrieb mit Stellungsrückmeldung
  Der vom PID-Algorithmus gesendete Wert PID_LMNG.PID_OUTV wird auf vorgebbare Werte LMN_HLM und LMN_LLM begrenzt. Die Differenz zwischen Stellwert und Stellungsrückmeldung wird auf ein Dreipunktglied mit Hysterese geschaltet. Die Abschaltschwelle wird aus der Motorstellzeit berechnet. Zur Reduzierung der Schalthäufigkeit wird die Ansprechschwelle adaptiert. Der Impulsformer PULSEOUT gewährleistet, dass die Mindestimpuls– und Pausendauer eingehalten werden.
- Automatikbetrieb ohne Stellungsrückmeldung
  Die Differenz aus dem I-Anteil des PID-Algorithmus und einer simulierten Stellungsrückmeldung wird in den Integrator INT summiert. Der Eingang des Integrators ist die Differenz aus 100.0/0.0/ -100.0 (je nach Zustand der Ausgangssignale QLMNUP und QLMNDN), bezogen auf die Motorstellzeit MTR_TM und der Regeldifferenz mal Reglerverstärkung, bezogen auf die Integrationszeit TI des PID-Algorithmus (PID_LMNG.PID_SCTR). Der Ausgang des Integrators bildet den Rückführungswert, der mit dem verbliebenen PD-Anteil des PID-Algorithmus (PID_LMNG.PID_OUTV) verglichen wird. Die Differenz wird auf ein Dreipunktglied mit Hysterese geschaltet. Die Abschaltschwelle wird aus der Motorstellzeit berechnet.
  Zur Reduzierung der Schalthäufigkeit wird die Ansprechschwelle adaptiert. Der Impulsformer PULSEOUT gewährleistet, dass die Mindestimpulsdauer und die Mindestpausendauer eingehalten werden.

Simulation der Stellungsrückmeldung

> **⚠ WARNUNG**
>
> Die Stellungsrückmeldung wird lediglich simuliert. Sie muss nicht übereinstimmen mit der realen Stellungsrückmeldung des Stellventils.
>
> Ist eine reale Stellungsrückmeldung vorhanden, so sollte diese auch genutzt werden.

Die Simulation der Stellungsrückmeldung erfolgt durch einen Integrierer mit der Motorstellzeit MTR_TM als Integrationszeitkonstante. Durch eine steigende Flanke an LMNRS_ON wird die Simulation mit dem Startwert LMNRSVAL gestartet. Steht LMNRS_ON auf FALSE, so wird am Parameter LMNR_SIM der Startwert LMNRSVAL ausgegeben.

Durch die Anschlagsignale LMNR_HS = TRUE und LMNR_LS = TRUE wird die Integration nach oben und unten gesperrt.

Bedienung über HMI

Mit statischen Variablen können Sie über ein HMI den Stellwertzweig unterbrechen und eigene Stellwerte vorgeben (siehe nachfolgendes Bild).

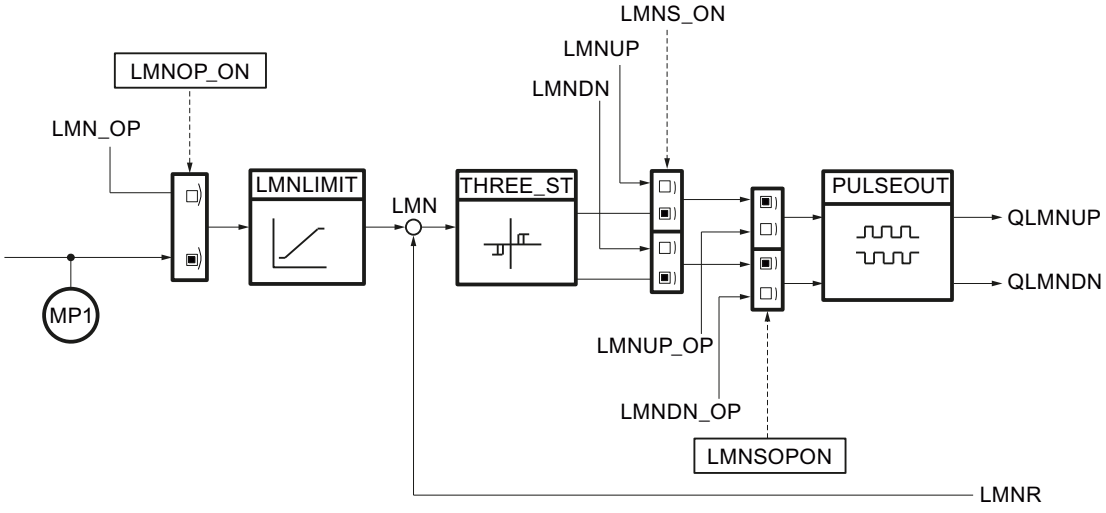

Wenn LMNOP_ON auf TRUE gesetzt wird, wird LMN_OP auf den Stellwert LMN durchgeschaltet. MP1 enthält den Stellwert, der von der Anweisung LMNGEN_S verarbeitet wird.

Wenn die Stellwertrampe eingeschaltet ist, kann stoßfrei zwischen den Werten MP1 und LMN_OP umgeschaltet werden. LMN wird entsprechend der Rampeneinstellung auf den Wert MP1 zurückgeführt.

Wenn LMNSOPON auf TRUE gesetzt wird, können Sie den Stellwert mit LMNUP_OP vergrößern oder mit LMNDN_OP verkleinern. Dies gilt für Schrittregler mit und ohne Stellungsrückmeldung.

Die statischen Variablen LMNOP_ON, LMN_OP, MP1, LMNSOPON, LMNUP_OP und LMNDN_OP dürfen nicht verschaltet werden und nur über HMI bedient und beobachtet werden.

Geänderte statische Variablen wirken erst dann auf den Prozess, wenn sie an die CPU übertragen wurden.

Eingangsparameter LMNGEN_S

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN_HLM	0.0	REAL	100.0	Stellwert obere Begrenzung technischer Wertebereich > LMN_LLM
LMN_LLM	4.0	REAL	0.0	Stellwert untere Begrenzung technischer Wertebereich < LMN_HLM
LMNR_IN	8.0	REAL	0.0	Stellungsrückmeldung technischer Wertebereich
MTR_TM	12.0	TIME	T#30s	Motorstellzeit ≥ CYCLE
PULSE_TM	16.0	TIME	T#3s	Mindestimpulsdauer ≥ CYCLE
BREAK_TM	20.0	TIME	T#3s	Mindestpausendauer ≥ CYCLE
MAN_ON	24.0	BOOL	TRUE	Handbetrieb einschalten
MANGN_ON	24.1	BOOL	FALSE	Handwertbedienung einschalten
MANUP	24.2	BOOL	FALSE	Handwert hoch
MANDN	24.3	BOOL	FALSE	Handwert tief
LMNR_ON	24.4	BOOL	FALSE	Stellungsrückmeldung einschalten
LMNR_HS	24.5	BOOL	FALSE	oberes Anschlagsignal der Stellungsrückmeldung
LMNR_LS	24.6	BOOL	FALSE	unteres Anschlagsignal der Stellungsrückmeldung
LMNS_ON	24.7	BOOL	FALSE	Handbetrieb der Stellwertsignale einschalten
LMNUP	25.0	BOOL	FALSE	Stellwertsignal Hoch
LMNDN	25.1	BOOL	FALSE	Stellwertsignal Tief
COM_RST	25.2	BOOL	FALSE	Neustart
CYCLE	26.0	TIME	T#1s	Abtastzeit ≥ 1ms
PID_LMNG	30.0	STRUC		PID-LMNGEN Schnittstelle

Ausgangsparameter LMNGEN_S

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN	48.0	REAL	0.0	Stellwert
QLMNUP	52.0	BOOL	FALSE	Stellwertsignal up

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
QLMNDN	52.1	BOOL	FALSE	Stellwertsignal Tief
QLMN_HLM	52.2	BOOL	FALSE	obere Begrenzung des Stellwertes angesprochen
QLMN_LLM	52.3	BOOL	FALSE	untere Begrenzung des Stellwertes angesprochen
LMNG_PID	54.0	STRUC		PID-LMNGEN Schnittstelle

Durchgangsparameter LMNGEN_S

Tabelle 4-226

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
MAN	72.0	REAL	0.0	Handwert technischer Wertebereich

Statische Variablen LMNGEN_S

Nicht aufgeführte Variabeln dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMNSOPON	76.0	BOOL	FALSE	Bedienung über HMI Handbetrieb der Stellwertausgangssignale einschalten
LMNUP_OP	76.1	BOOL	FALSE	Bedienung über HMI Stellwert größer
LMNDN_OP	76.2	BOOL	FALSE	Bedienung über HMI Stellwert kleiner
LMNOP_ON	76.3	BOOL	FALSE	Bedienung über HMI Handbetrieb einschalten
LMNRS_ON	76.4	BOOL	FALSE	Simulierte Stellungsrückmeldung einschalten
MP1	78.0	REAL	0.0	Bedienung über HMI Stellwert, der von LMNGEN_C verarbeitet wird.
LMM_OP	82.0	REAL	0.0	Bedienung über HMI Handwert
LMNRSVAL	86.0	REAL	0.0	Startwert für die simulierte Stellungsrückmeldung
LMNR_SIM	9.0	REAL	0.0	Simulierte Stellungsrückmeldung

NONLIN

Beschreibung NONLIN

Mit der Anweisung NONLIN kann ein Eingangswert, z.B. Messwert von einem Thermoelement, über eine vorgebbare Kennlinie (Polygonzug) angepasst werden.

Arbeitsweise

NONLIN interpoliert zu der momentanen Eingangsgröße INV die entsprechende Ausgangsgröße OUTV anhand von Stützpunkten auf einem Polygonzug. Die Stützpunkte des Polygonzuges sind in einem globalen Datenbaustein abgelegt.

Globaler Datenbaustein DB_NONLIN (Seite 6003) ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Hinweis

Die Anweisung überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob der Parameter DB_NBR. NBR_PTS (Anzahl der Stützpunkte) zur Länge des Datenbausteins passt. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

NONLIN verfügt über folgende Funktionen:

Funktionen	DFOUT_ON	TRACK	OUTV
Interpolation	FALSE	FALSE	OUTV = f(INV)
Ausgang vorbelegen	TRUE	beliebig	DF_OUTV
Nachführen	FALSE	TRUE	OUTV = INV

- **Interpolation**
  Das folgende Bild zeigt, wie die Ausgangsgröße OUTV aus der Eingangsgröße INV ermittelt wird.
  OUTV = f(INV)

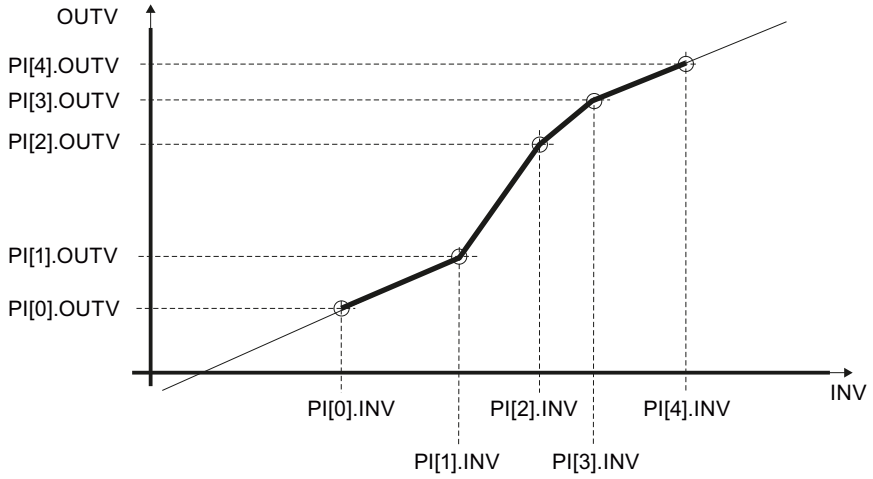

  Liegt der Eingangswert unterhalb von Stützpunkt 0, so wird mit der Steigung des Stützpunktpaares 0/1 extrapoliert. Liegt er oberhalb des letzten Stützpunktes, so wird mit der Steigung des letzten Stützpunktpaares extrapoliert. Damit die Anweisung ein sinnvolles Ergebnis liefert, müssen die Werte der Stützpunkte in aufsteigender Reihenfolge streng monoton steigend sein.

- **Ausgang vorbelegen**
  Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben, die Änderung von OUTV erfolgt sprungförmig. Bei Umschaltung nach DFOUT_ON = FALSE erfolgt die Änderung von OUTV ebenfalls sprungförmig.

- **Nachführen**
  Bei TRACK=TRUE wird der Eingangswert direkt ausgegeben (OUTV=INV). Die Änderung erfolgt wie bei DFOUT_ON sprungförmig. TRACK hat niedrigere Priorität als DFOUT_ON.

Neustart

Bei Neustart wird OUTV = 0.0 ausgegeben. Ist DFOUT_ON=TRUE, so wird DF_OUTV ausgegeben.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter NONLIN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
DF_OUTV	4.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
DB_NBR	8.0	BLOCK_DB	DB 1	Datenbausteinnummer
DFOUT_ON	10.0	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
TRACK	10.1	BOOL	FALSE	Nachführen OUTV=INV
COM_RST	10.2	BOOL	FALSE	Neustart

Ausgangsparameter NONLIN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	12.0	REAL	0.0	Ausgangsgröße

Globaler Datenbaustein DB_NONLIN

Die Stützpunkte des Polygonzugs für die Anweisung NONLIN werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
DB_NBR.NBR_PTS	INT	4	Größte Stützpunkt-Nummer Wertebereich 1 - 255
DB_NBR.PI[0].OUTV	REAL	0.0	Ausgangswert [0] Startpunkt 0 technischer Wertebereich
DB_NBR.PI[0].INV	REAL	0.0	Eingangswert [0] Startpunkt 0 technischer Wertebereich
DB_NBR.PI[1].OUTV	REAL	0.0	Ausgangswert [1] Stützpunkt 1 technischer Wertebereich
DB_NBR.PI[1].INV	REAL	0.0	Eingangswert [1] Stützpunkt 1 technischer Wertebereich
DB_NBR.PI[2].OUTV	REAL	0.0	Ausgangswert [2] Stützpunkt 2 technischer Wertebereich
DB_NBR.PI[2].INV	REAL	0.0	Eingangswert [2] Stützpunkt 2 technischer Wertebereich
DB_NBR.PI[3].OUTV	REAL	0.0	Ausgangswert [3] Stützpunkt 3 technischer Wertebereich

Parameter	Datentyp	Vorbelegung	Beschreibung
DB_NBR.PI[3].INV	REAL	0.0	Eingangswert [3] Stützpunkt 3 technischer Wertebereich
DB_NBR.PI[4].OUTV	REAL	0.0	Ausgangswert [4] Stützpunkt 4 technischer Wertebereich
DB_NBR.PI[4].INV	REAL	0.0	Eingangswert [4] Stützpunkt 4 technischer Wertebereich

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den untenstehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen „.DB“:
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner „Externe Quellen“.
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_NONLIN"
TITLE = data block nonlinear static function
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_NONLI
    STRUCT
        NBR_PTS : Int;                      // number of points (excluding starting point)
        PI : Array[0..4] of Struct
            OUTV : Real;                    // output variable
            INV : Real;                     // input variable
        END_STRUCT;
    END_STRUCT;
```

```
BEGIN
    NBR_PTS := 4;
    PI[0].OUTV := 1.0;
    PI[0].INV := 2.0;
    PI[1].OUTV := 2.0;
    PI[1].INV := 3.0;
    PI[2].OUTV := 4.0;
    PI[2].INV := 4.0;
    PI[3].OUTV := 5.0;
    PI[3].INV := 5.0;
    PI[4].OUTV := 6.0;
    PI[4].INV := 7.0;
END_DATA_BLOCK
```

NORM

Beschreibung NORM

Bei Istwerten liegt der vom Geber gelieferte Wert oft in einem für den Anwender ungünstigen Bereich (z.B. 0 bis 10V entsprechen 0 bis 1200ºC oder 0 bis 10V entsprechen 0 bis 3000U/min). Durch die Normierung von Sollwert oder Istwert liegen beide Prozessgrößen im gleichen Wertebereich.

Arbeitsweise

Die Anweisung normiert die Eingangsgröße in eine Ausgangsgröße mit einem anderen Wertebereich. Die Eingangsgröße INV wird an der Normierungsgeraden in die Ausgangsgröße OUTV überführt. Die Normierungsgerade wird durch die beiden Wertepaare IN_LVAL, OUT_LVAL und IN_HVAL, OUT_HVAL definiert.

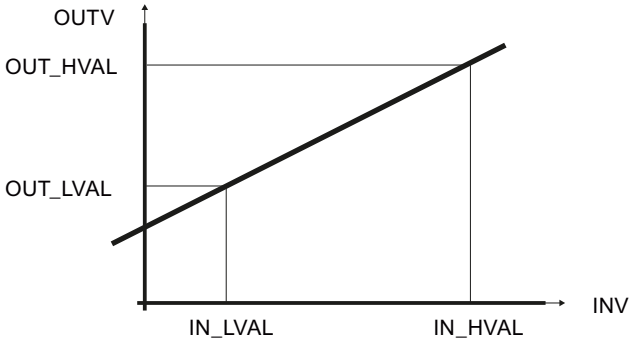

Neustart

Die Anweisung besitzt keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter NORM

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
IN_HVAL	4.0	REAL	100.0	physikalischer Eingangswert unten technischer Wertebereich. > IN_LVAL
OUT_HVAL	8.0	REAL	100.0	physikalischer Ausgangswert unten technischer Wertebereich. > OUT_LVAL
IN_LVAL	12.0	REAL	0.0	physikalischer Eingangswert oben technischer Wertebereich < IN_HVAL
OUT_LVAL	16.0	REAL	0.0	physikalischer Ausgangswert oben technischer Wertebereich < OUT_HVAL

Ausgangsparameter NORM

Parameter	Offset	Datentyp	Beschreibung
OUTV	20.0	REAL	Ausgangsgröße

OVERRIDE

Beschreibung OVERRIDE

Die Anweisung wird benötigt zur Realisierung einer Ablöseregelung (Override-Regelung).

Arbeitsweise

Zwei PID-Regler werden auf ein Stellglied geschaltet. Hierzu werden 2 PID-Anweisungen über die Anweisung OVERRIDE mit der Anweisung LMNGEN_C oder LMNGEN_S verschaltet. Wenn Sie LMNGEN_S verwenden, muss die Stellungsrückmeldung aktiviert sein (LMNR_ON=TRUE).

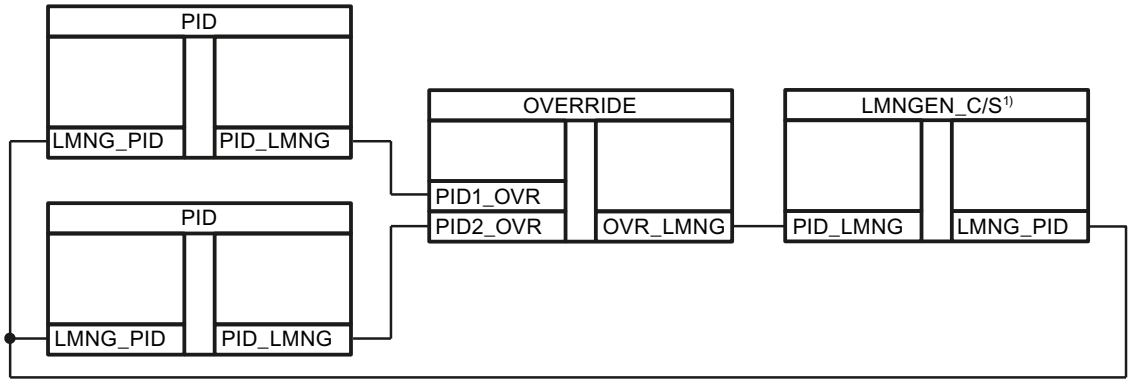

Die Anweisung vergleicht die Werte PID1_OVR und PID2_OVR und schaltet entweder den größeren, oder den kleinere

Die Anweisung leitet entweder den Wert an PID1_OVR oder an PID2_OVR an LMN_GEN_C oder LMN_GEN_S weiter.

Wenn die Schalter PID1_ON, PID2_ON ungleich sind, wird immer der eine Wert weitergeleitet. Wenn die Schalter PID1_ON, PID2_ON gleich sind, wird der größere (OVR_MODE=FALSE) oder der kleinere (OVR_MODE=TRUE)der beiden Werte weitergeleitet.

PID1_ON	PID2_ON	OVR_MOD	Funktion
TRUE	FALSE	nicht relevant	PID1_OVR wird weitergeleitet.
FALSE	TRUE		PID2_OVR wird weitergeleitet.
FALSE	FALSE	FALSE	Der größere Wert von PID1_OVR und PID2_OVR wird weitergeleitet.
TRUE	TRUE	FALSE	
FALSE	FALSE	TRUE	Der kleinere Wert von PID1_OVR und PID2_OVR wird weitergeleitet.
TRUE	TRUE	TRUE	

Wenn PID1_OVR weitergeleitet wird, gilt QPID1=TRUE, wenn PID2_OVR weitergeleitet wird, gilt QPID2=TRUE.

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter OVERRIDE

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
PID1_ON	0.0	BOOL	FALSE	PID-Regler 1 ein
PID2_ON	0.1	BOOL	FALSE	PID-Regler 2 ein
OVR_MODE	0.2	BOOL	FALSE	Ablösemodus FALSE=Maximum, TRUE=Minimum
PID1_OVR	2.0	STRUC		PID-LMNGEN Schnittstelle
PID2_OVR	20.0	STRUC		PID-LMNGEN Schnittstelle

Ausgangsparameter OVERRIDE

Parameter	Offset	Datentyp	Beschreibung
QPID1	38.0	BOOL	PID-Regler 1 aktiv
QPID2	38.1	BOOL	PID-Regler 2 aktiv
OVR_LMNG	40.0	STRUC	PID-LMNGEN Schnittstelle

PARA_CTL

Beschreibung PARA_CTL

Die Anweisung wird bei Reglerstrukturen mit Parameterumschaltungen verwendet, wenn für unterschiedliche Arbeitsbereiche optimale Reglerparameter benötigt werden.

Arbeitsweise

Mehrere Reglerparametersätze (GAIN, TI, TD und TM_LAG) können in einen PID-Regler übertragen werden.

Über die Schalter PSET1_ON...PSET4_ON kann einer der 4 Parametersätze auf die Ausgänge GAIN, TI, TD und TM_LAG durchgeschaltet werden.

Wenn	Dann gilt
PSET1_ON = TRUE	GAIN = PARASET1.GAIN
	TI = PARASET1.TR
	TD = PARASET1.TD
	TM_LAG = PARASET1.TM_LAG
PSET2_ON = TRUE	GAIN = PARASET2.GAIN
	TI = PARASET2.TR
	TD = PARASET2.TD
	TM_LAG = PARASET2.TM_LAG

PSET3_ON = TRUE	GAIN = PARASET3.GAIN
	TI = PARASET3.TR
	TD = PARASET3.TD
	TM_LAG = PARASET3.TM_LAG
PSET4_ON = TRUE	GAIN = PARASET4.GAIN
	TI = PARASET4.TR
	TD = PARASET4.TD
	TM_LAG = PARASET4.TM_LAG

Wenn 2 oder mehrere Schalter auf TRUE gesetzt sind, hat immer der Schalter mit der kleinsten Nummer die höchste Priorität. Wenn kein Schalter auf TRUE gesetzt, wird der 1. Parametersatz übertragen.

Um ein stoßfreies Umschalten zwischen den Parametersätzen zu erreichen, muss beim Parameter GAIN ein Hochlaufgeber (ROC_LIM) zwischen den Anweisung PID und PARA_CTL geschaltet werden.

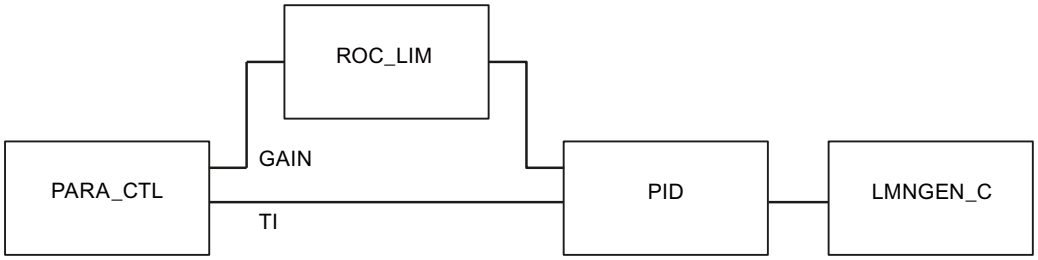

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter PARA_CTL

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
PSET1_ON	0.0	BOOL	FALSE	Parametersatznummer 1 übertragen
PSET2_ON	0.1	BOOL	FALSE	Parametersatznummer 2 übertragen
PSET3_ON	0.2	BOOL	FALSE	Parametersatznummer 3 übertragen
PSET4_ON	0.3	BOOL	FALSE	Parametersatznummer 4 übertragen
PARASET1.GAIN	2.0	REAL		Proportionalverstärkung 1
PARASET1.TI	6.0	TIME	5 s	Integrationszeit 1
PARASET1.TD	10.0	TIME	10 s	Differenzierzeit 1
PARASET1.TM_LAG	14.0	TIME	2 s	Verzögerungszeit 1

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
PARASET2.GAIN	18.0	REAL		Proportionalverstärkung 2
PARASET2.TI	22.0	TIME	5 s	Integrationszeit 2
PARASET2.TD	26.0	TIME	10 s	Differenzierzeit 2
PARASET2.TM_LAG	30.0	TIME	2 s	Verzögerungszeit 2
PARASET3.GAIN	34.0	REAL		Proportionalverstärkung 3
PARASET3.TI	38.0	TIME	5 s	Integrationszeit 3
PARASET3.TD	42.0	TIME	10 s	Differenzierzeit 3
PARASET3.TM_LAG	46.0	TIME	2 s	Verzögerungszeit 3
PARASET4.GAIN	500	REAL		Proportionalverstärkung 4
PARASET4.TI	54.0	TIME	5 s	Integrationszeit 4
PARASET4.TD	58.0	TIME	10 s	Differenzierzeit 4
PARASET4.TM_LAG	62.0	TIME	2 s	Verzögerungszeit 4

Ausgangsparameter PARA_CTL

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
GAIN	66.0	REAL	1.0	Proportionalbeiwert
TI	70.0	TIME	10s	Integrationszeit
TD	74.0	TIME	5s	Differenzierzeit
TM_LAG	78.0	TIME	2s	Verzögerungszeit

PID

Beschreibung PID

Die Anweisung beinhaltet den PID-Algorithmus zum Aufbau der folgenden Reglertypen:

Kontinuierliche PID-Regler:	PID + LMNGEN_C
PID-Impulsregler für proportionale Stellglieder:	PID + LMNGEN_C + PULSEGEN
PID-Schrittregler für integrierende Stellglieder:	PID + LMNGEN_S

Die Anweisung realisiert den PID-Algorithmus. Sie ist in reiner Parallelstruktur ausgeführt und wirkt ausschließlich als Stellungsalgorithmus. Die Proportional-, Integral- und Differentialanteile können einzeln zu- bzw. abgeschaltet werden. Damit lassen sich P-, PI-, PD- und PID-Regler parametrieren.

Die Berechnung der P- und D-Anteile können in die Rückführung gelegt werden. Durch das Verlegen des P- und des D-Anteils in den Rückführungszweig wird das Führungsverhalten "stoßfrei" bei gleicher Schnelligkeit der Ausregelung von Störgrößen. Auf die übliche Anwendung eines Sollwertintegrators zur Vermeidung von Sollwertsprüngen kann meist verzichtet werden.

Während die PID-Anweisung in einer Weckalarmebene platziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstante angepasst ist, können die stellgliedverarbeitenden Anweisungen (LMNGEN_C bzw. LMNGEN_S) in einer schnelleren Weckalarmebene untergebracht werden.

Neustart

Bei Neustart werden alle Aus- und Durchgangsparameter mit Null vorbelegt.

Bausteininterne Begrenzungen

Die Integrationszeit wird auf die halbe Abtastzeit nach unten begrenzt.

Die Differenzierzeit wird auf die Abtastzeit nach unten begrenzt.

Die Verzögerungszeit wird auf die halbe Abtastzeit nach unten begrenzt.

TI_{intern} = CYCLE/2	für	TI < CYCLE/2
TD_{intern} = CYCLE	für	TD < CYCLE
TM_LAG_{intern} = CYCLE/2	für	TM_LAG < CYCLE/2

Die anderen Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Arbeitsweise PID

Reglerstruktur

Über die Schalter P_SEL, I_SEL und D_SEL sind folgende Reglerstrukturen möglich:

Reglerstruktur	P_SEL	I_SEL	D_SEL
P-Regelung	TRUE	TRUE bzw. FALSE	FALSE
PI-Regelung	TRUE	TRUE	FALSE
PD-Regelung	TRUE	FALSE	TRUE
PID-Regelung	TRUE	TRUE	TRUE

Blockschaltbild PID-Algorithmus

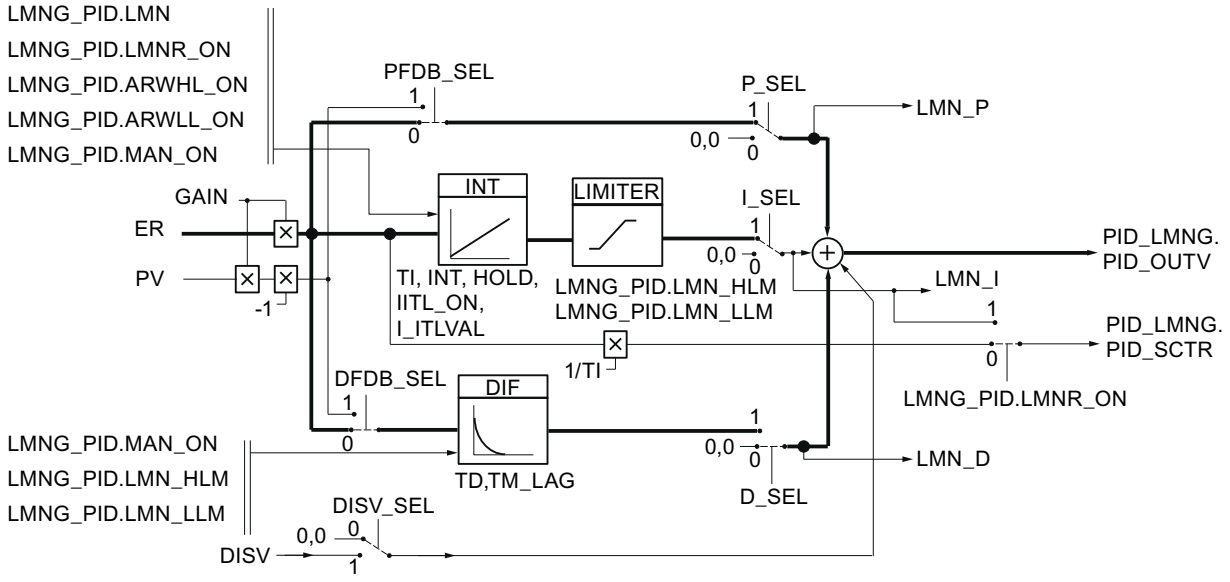

P-Anteil

Der P-Anteil kann wahlweise mit dem Schalter P_SEL ein- bzw. ausgeschaltet werden. Der P-Anteil stellt das Produkt aus Regeldifferenz ER und der Proportionalverstärkung GAIN dar. Über den Schalter PFDB_SEL kann er in die Rückführung gelegt werden. Wenn der P-Anteil in die Rückführung gelegt ist, ist der P-Anteil das Produkt aus Istwert PV und der Proportionalverstärkung.

I-Anteil

Der I-Anteil kann wahlweise mit dem Schalter I_SEL ein- bzw. ausgeschaltet werden. Das Zeitverhalten wird durch die Integrationszeit TI bestimmt. Bei ausgeschaltetem Zustand werden der I-Anteil und der interne Speicher des I-Anteils auf Null gesetzt. Mit INT_HOLD kann der I-Anteil eingefroren werden. Über den Schalter I_ITL_ON können Sie den I-Anteil mit dem Wert am Eingang I_ITLVAL belegen. Bei Stellwertbegrenzungen bleibt der I-Anteil auf dem Altwert stehen (Anti reset wind-up).

Beim Schrittregler ohne Stellungsrückmeldung gilt: LMN_I = 0.0

D-Anteil

Der D-Anteil kann über den Schalter D_SEL ein- bzw. ausgeschaltet werden. Mit dem Schalter DFDB_SEL kann er in die Rückführung gelegt werden. Die Eingangsgröße des D-Anteils ist nun die Regelgröße PV. Das Zeitverhalten wird durch die Differenzierzeit TD bestimmt.

Besonders bei schnellen Strecken und eingeschaltetem D-Anteil kann es zu einer Schwingung auf dem Stellwertsignal kommen. In diesem Fall ist eine Verbesserung des Regelverhaltens mit der im D-Anteil integrierten Verzögerung 1. Ordnung möglich. Meist genügt bereits eine kleine Verzögerungszeit TM_LAG, um den gewünschten Erfolg zu erzielen.

Störgrößenaufschaltung

Additiv zum Stellwert kann eine Störgröße DISV aufgeschaltet werden. Über den Schalter DISV_SEL kann sie wahlweise zu- bzw. abgeschaltet werden.

Eingangsparameter PID

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
ER	0.0	REAL	0.0	Regeldifferenz technischer Wertebereich
PV	4.0	REAL	0.0	Regelgröße (P– bzw. D–Anteil in Rückführung) technischer Wertebereich
GAIN	8.0	REAL	1.0	Proportionalbeiwert
TI	12.0	TIME	T#20s	Integrationszeit
I_ITLVAL	16.0	REAL	0.0	Initialisierungswert für I-Anteil technischer Wertebereich
TD	20.0	TIME	T#10s	Differenzierzeit
TM_LAG	24.0	TIME	T#2s	Verzögerungszeit des D-Anteils
DISV	28.0	REAL	0.0	Störgröße technischer Wertebereich
P_SEL	32.0	BOOL	TRUE	P-Anteil einschalten
PFDB_SEL	32.1	BOOL	FALSE	P-Anteil in Rückführung schalten
DFDB_SEL	32.2	BOOL	FALSE	D-Anteil in Rückführung schalten
I_SEL	32.3	BOOL	TRUE	I-Anteil einschalten
INT_HPOS	32.4	BOOL	FALSE	I-Anteil in positiver Richtung einfrieren
INT_HNEG	32.5	BOOL	FALSE	I-Anteil in negativer Richtung einfrieren
I_ITL_ON	32.6	BOOL	FALSE	I-Anteil setzen
D_SEL	32.7	BOOL	FALSE	D-Anteil einschalten
DISV_SEL	33.0	BOOL	TRUE	Störgröße aufschalten
SMOO_CHG	33.1	BOOL	TRUE	sanfter Übergang von Hand-Betrieb auf Automatik-Betrieb
COM_RST	33.2	BOOL	FALSE	Neustart
CYCLE	34.0	TIME	T#1s	Abtastzeit ≥ 1ms
LMNG_PID	38.0	STRUC		PID-LMNGEN Schnittstelle

Ausgangsparameter PID

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN_P	56.0	REAL	0.0	P-Anteil
LMN_I	60.0	REAL	0.0	I-Anteil

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
LMN_D	64.0	REAL	0.0	D-Anteil
PID_LMNG	68.0	STRUC		PID-LMNGEN Schnittstelle

PULSEGEN_M

Beschreibung PULSEGEN_M

Die Anweisung PULSEGEN_M dient zum Aufbau eines PID-Reglers mit Impulsausgang für proportional wirkende Stellglieder. Mit der Anweisung PULSEGEN_M lassen sich PID-Zwei- oder Dreipunktregler mit Pulsweitenmodulation aufbauen. Die Anweisung wird meistens in Verbindung mit dem kontinuierlichen Regler PID + LMNGEN_C angewendet.

Das folgende Bild zeigt Ihnen die Verschaltung der Anweisungen.

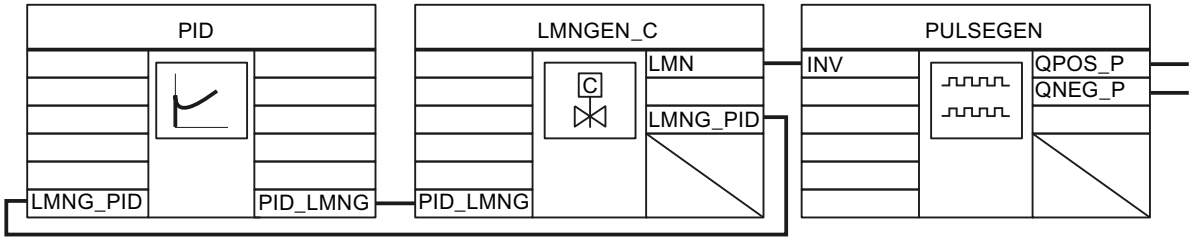

Neustart

Bei Neustart werden alle Signalausgänge auf Null gesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Arbeitsweise PULSEGEN_M

Impulsweitenmodulation

PULSEGEN_M transformiert die Eingangsgröße INV (= LMN des PID-Reglers) durch Modulation der Impulsweite in eine Impulsfolge mit konstanter Periodendauer PER_TM. Die Periodendauer entspricht der Zykluszeit, mit der die Eingangsgröße aktualisiert wird. Die Dauer eines Impulses pro Periodendauer ist proportional der Eingangsgröße. Dabei ist der durch PER_TM parametrierte Zyklus nicht identisch mit dem Bearbeitungszyklus der Anweisung PULSEGEN_M. Vielmehr setzt sich ein Zyklus PER_TM aus mehreren Bearbeitungszyklen der Anweisung PULSEGEN_M zusammen, wobei die Anzahl der PULSEGEN_M-Aufrufe pro PER_TM-Zyklus ein Maß für die Genauigkeit der Impulsbreite darstellt.

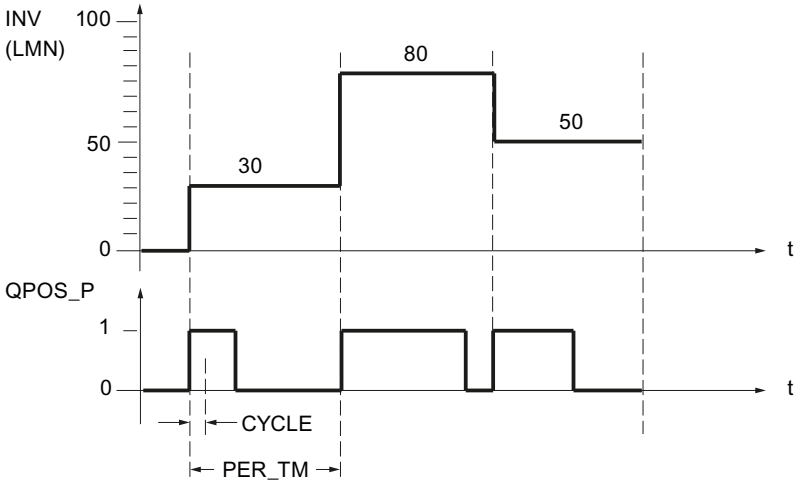

Eine Eingangsgröße 30% und 10 PULSEGEN-Aufrufe pro PER_TM bedeuten also:

- "eins" am Ausgang QPOS_P für die ersten drei Aufrufe von PULSEGEN_M (30% von 10 Aufrufen)
- "null" am Ausgang QPOS_P für sieben weitere Aufrufe von PULSEGEN_M (70% von 10 Aufrufen)

Mindestimpuls- bzw. Mindestpausendauer

Eine richtig parametrierte Mindestimpuls- bzw. Mindestpausendauer P_B_TM kann kurze Ein- oder Ausschaltzeiten verhindern, die die Lebensdauer von Stellgliedern beeinträchtigen. Kleine Absolutwerte der Eingangsgröße LMN, die eine Impulsdauer kleiner als P_B_TM erzeugen würden, werden unterdrückt. Große Eingangswerte, die eine Impulsdauer größer als PER_TM - P_B_TM erzeugen würden, werden auf 100 % bzw. -100 % gesetzt.

Folgende Formel sollte gelten: P_B_TM ≤ 0,1 × PER_TM

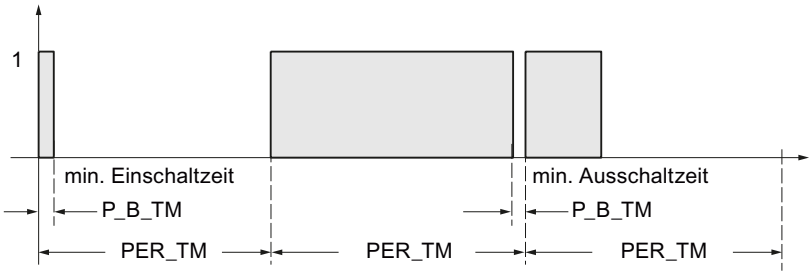

Die Dauer der positiven oder negativen Impulse errechnet sich aus Eingangsgröße (in %) mal Periodendauer:

Impulsdauer = INV / 100 * PER_TM

Genauigkeit der Pulsweitenmodulation

Während der kontinuierliche Regler PID+LMGEN_C in einer langsamen Weckalarmebene platziert wird, deren Zykluszeit der dominierenden Streckenzeitkonstante angepasst ist, muss PULSEGEN_M in einer schnelleren Weckalarmebene untergebracht werden.

Durch ein "Abtastverhältnis" von 1:10 (Regler-Aufrufe zu PULSEGEN_M-Aufrufe) ist die Stellwertgenauigkeit auf 10% beschränkt, d.h. vorgegebene Eingangswerte INV können nur mit einer Genauigkeit von 10% auf eine Impulslänge am Ausgang QPOS_P abgebildet werden.

Die Genauigkeit erhöht sich mit der Anzahl der PULSEGEN_M-Aufrufe pro Regler-Aufruf.

Je schneller die Weckalarmebene ist, desto genauer kann der Stellwert ausgegeben werden. Bei einer 100 mal schnelleren Weckalarmebene erreicht man eine Auflösung von 1 % des Stellwertbereiches.

Automatische Synchronisation

Es besteht die Möglichkeit, die Impulsausgabe mit der Regler-Anweisung automatisch zu synchronisieren. Damit ist gewährleistet, dass ein geänderter Wert der Stellgröße LMN(t) schnellstmöglich als proportional geänderte Impulsdauer des Binärsignals ausgegeben wird.

Der Impulsformer wertet jeweils im Zeitabstand der Periodendauer PER_TM die Eingangsgröße INV aus. Da aber INV in einer langsameren Weckalarmebene berechnet wird, sollte der Impulsformer möglichst schnell nach der Aktualisierung von INV mit der Umwandlung des diskreten Wertes in ein Impulssignal beginnen. Dazu kann die Anweisung den Start der Periode nach folgendem Verfahren selbst synchronisieren. Hat sich INV geändert und befindet sich der Bausteinaufruf nicht im ersten oder in den letzten zwei Aufrufzyklen einer Periode, so wird eine Synchronisation vorgenommen. Die Impulsdauer wird neu berechnet und beim nächsten Zyklus mit einer neuen Periode wird mit der Ausgabe begonnen.

Die Periodendauer PER_TM muss der Abtastzeit CYCLE des Reglers entsprechen.

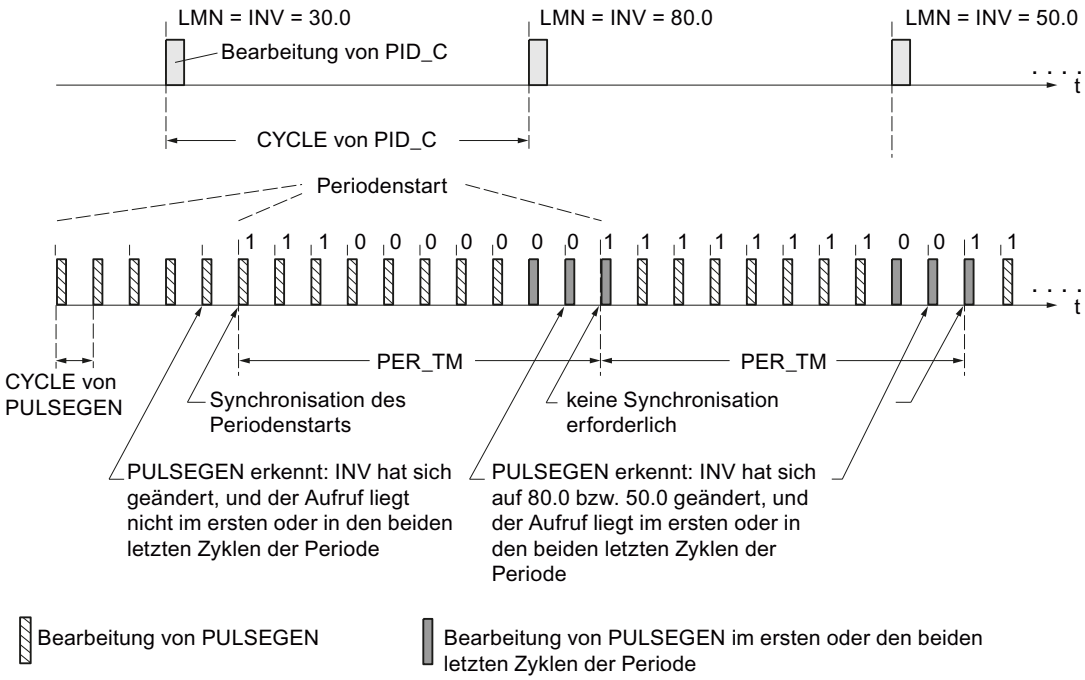

Die automatische Synchronisation ist abgeschaltet, wenn SYN_ON = FALSE.

Hinweis

Durch den Beginn der neuen Periode wird der Altwert von INV (d.h. von LMN) nach erfolgter Synchronisation mehr oder weniger ungenau auf das Impulssignal abgebildet.

Betriebsarten PULSEGEN_M

Betriebsarten

Je nach Parametrierung des Impulsformers können PID-Regler mit Dreipunktverhalten oder mit bipolarem bzw. unipolarem Zweipunktausgang konfiguriert werden. Nachstehende Tabelle zeigt die Einstellung der Schalterkombinationen für die möglichen Betriebsarten.

Betriebsart	MAN_ON	STEP3_ON	ST2BI_ON
Dreipunktregelung	FALSE	TRUE	beliebig
Zweipunktregelung mit bipolarem Stellbereich (-100 % ... 100 %)	FALSE	FALSE	TRUE
Zweipunktregelung mit unipolarem Stellbereich (0 % ... 100 %)	FALSE	FALSE	FALSE
Handbetrieb	TRUE	beliebig	beliebig

Handbetrieb bei Zwei- bzw. Dreipunkt-Regelung

Im Handbetrieb (MAN_ON = TRUE) können die Binärausgänge des Dreipunkt- bzw. Zweipunktreglers über die Signale POS_P_ON und NEG_P_ON unabhängig von INV gesetzt werden.

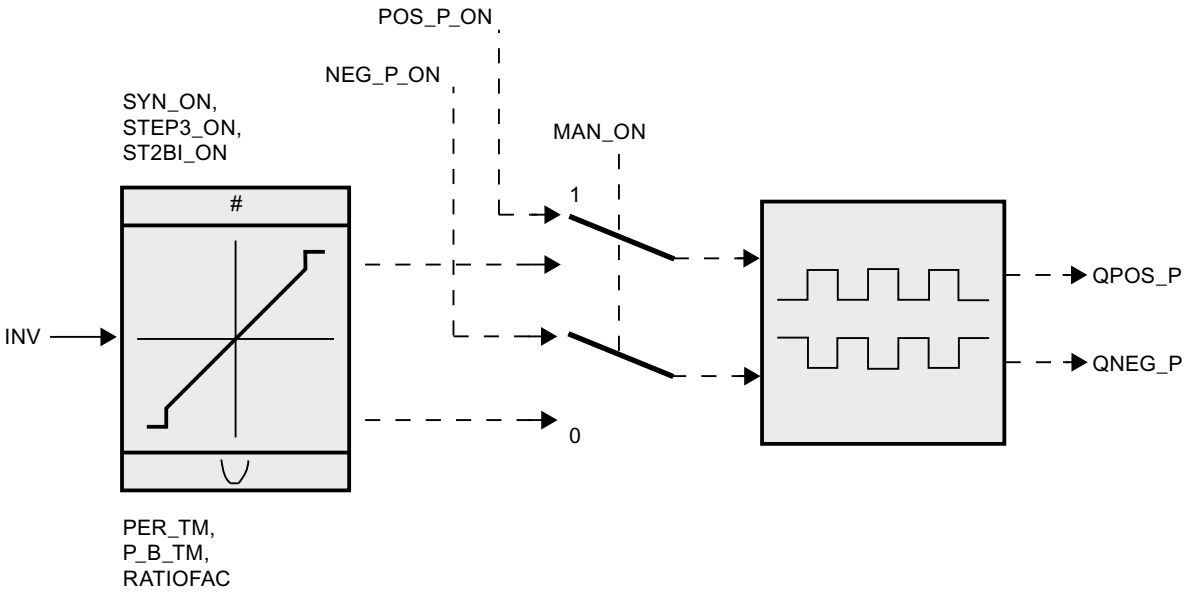

Regelung	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Dreipunktregelung	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Zweipunktregelung	FALSE	beliebig	FALSE	TRUE
	TRUE	beliebig	TRUE	FALSE

Dreipunktregelung PULSEGEN_M

Dreipunktregelung

In der Betriebsart "Dreipunktregelung" können drei Zustände des Stellsignals erzeugt werden, z. B. Mehr - Aus - Weniger, Vorwärts - Stop - Rückwärts, Heizen - Aus - Kühlen. Dazu werden die Zustandswerte der binären Ausgangssignale QPOS_P und QNEG_P den jeweiligen Betriebszuständen des Stellgliedes zugeordnet. Die Tabelle zeigt das Beispiel einer Temperaturregelung:

Ausgangssignale	heizen	aus	kühlen
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

Aus der Eingangsgröße wird über eine Kennlinie die Impulsdauer berechnet. Die Form dieser Kennlinie wird durch die Mindestimpuls- bzw. Mindestpausendauer und dem Verhältnisfaktor

definiert. Der normale Wert für den Verhältnisfaktor ist 1. Die Knickpunkte an den Kennlinien werden durch die Mindestimpuls- bzw. Mindestpausendauer P_B_TM verursacht.

Das folgende Bild zeigt eine symmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 1)

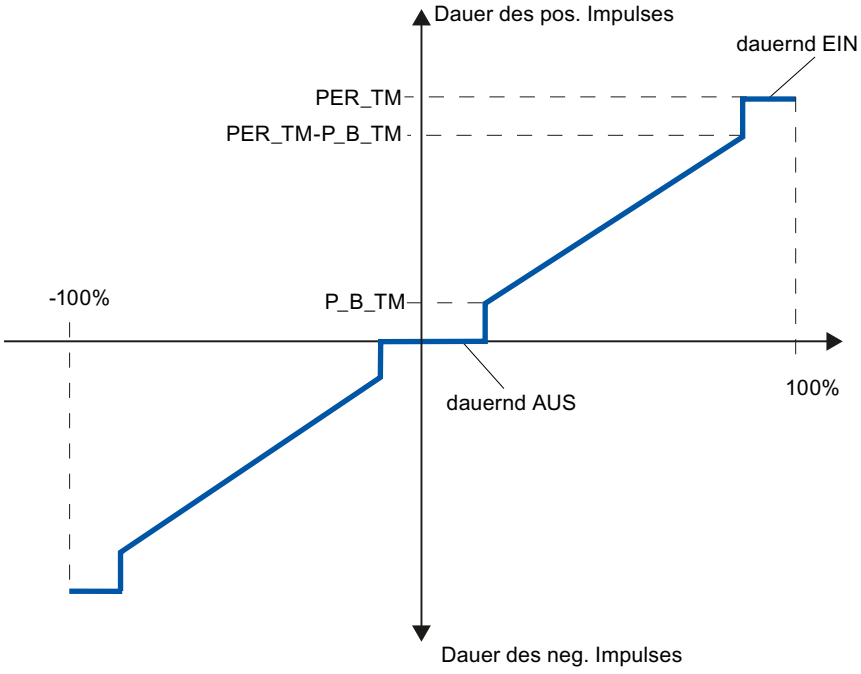

Dreipunktregelung unsymmetrisch

Über den Verhältnisfaktor RATIOFAC kann das Verhältnis der Dauer von positiven zu negativen Impulsen verändert werden. Bei einem thermischen Prozess lassen sich damit z. B. unterschiedliche Streckenzeitkonstanten für Heizen und Kühlen berücksichtigen.

Verhältnisfaktor < 1

Die aus Eingangsgröße mal Periodendauer berechnete Impulsdauer am negativen Impulsausgang wird um den Verhältnisfaktor verkürzt.

Positive Impulsdauer = INV /100 * PER_TM

Negative Impulsdauer = INV / 100 * PER_TM * RATIOFAC

Die negative Impulsdauer wird mit dem Verhältnisfaktor multipliziert.

Das folgende Bild zeigt die unsymmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 0.5)

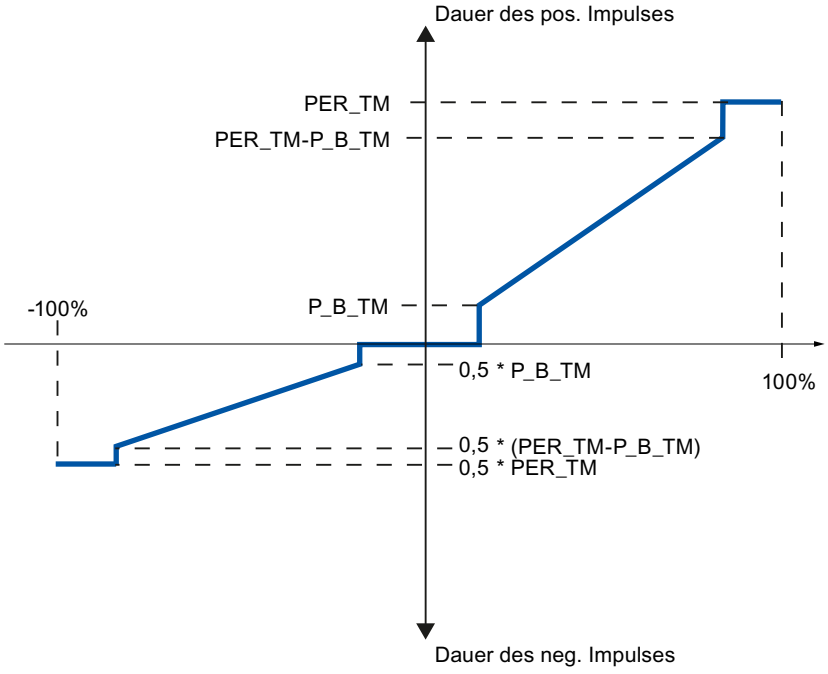

Der Verhältnisfaktor RATIOFAC beeinflusst die Mindestimpuls- bzw. Mindestpausendauer. Der Ansprechwert für negative Impulse wird mit dem Verhältnisfaktor multipliziert.

Verhältnisfaktor > 1

Die aus Eingangsgröße mal Periodendauer berechnete Impulsdauer am positiven Impulsausgang wird um den Verhältnisfaktor verkürzt.

positive Impulsdauer = INV / 100 * PER_TM / RATIOFAC

negative Impulsdauer = INV / 100 * PER_TM

Der Verhältnisfaktor RATIOFAC beeinflusst die Mindestimpuls- bzw. Mindestpausendauer. Der Ansprechwert für positive Impulse wird durch den Verhältnisfaktor dividiert.

Zweipunktregelung PULSEGEN_M

Bei der Zweipunktregelung wird nur der positive Impulsausgang QPOS_P von PULSEGEN_M mit dem betreffenden Ein/Aus-Stellglied verbunden. Je nach genutztem Stellwertbereich hat der Zweipunktregler einen bipolaren oder einen unipolaren Stellwertbereich.

Zweipunktregelung mit bipolarem Stellwertbereich (-100%...100%)

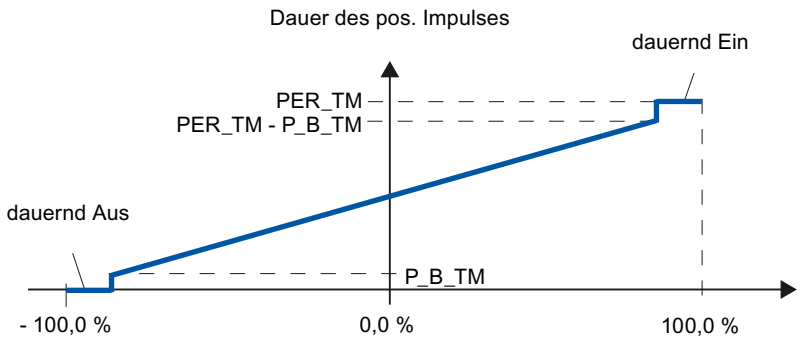

Zweipunktregelung mit unipolarem Stellwertbereich (0%...100%)

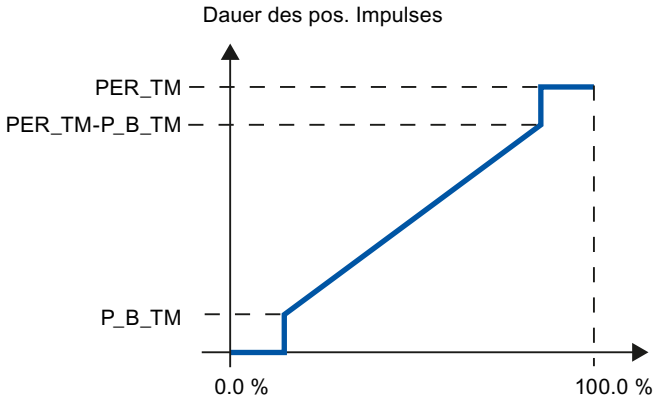

An QNEG_P steht das negierte Ausgangssignal zur Verfügung, falls die Verschaltung des Zweipunktreglers im Regelkreis ein logisch invertiertes Binärsignal für die Stellimpulse erfordert.

Impuls	Stellglied Ein	Stellglied Aus
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Eingangsparameter PULSEGEN_M

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsvariable -100.0..100.0 (%)
PER_TM	4.0	TIME	T#1s	Periodendauer PER_TM >= 20*CYCLE
P_B_TM	8.0	TIME	T#50ms	Mindestimpuls- bzw. Mindestpausendauer P_B_TM >= CYCLE
RATIOFAC	12.0	REAL	1.0	Verhältnisfaktor 0.1..10.0 (dimensionslos)

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
STEP3_ON	16.0	BOOL	TRUE	Dreipunktregelung einschalten
ST2BI_ON	16.1	BOOL	FALSE	Zweipunktregelung für bipolaren Stellwertbereich einschalten
MAN_ON	16.2	BOOL	FALSE	Handbetrieb einschalten
POS_P_ON	16.3	BOOL	FALSE	positiver Impuls ein
NEG_P_ON	16.4	BOOL	FALSE	negativer Impuls ein
SYN_ON	16.5	BOOL	TRUE	Synchronisation einschalten
COM_RST	16.6	BOOL	FALSE	Neustart
CYCLE	18.0	TIME	T#10ms	Abtastzeit CYCLE >= 1ms

Ausgangsparameter PULSEGEN_M

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
QPOS_P	22.0	BOOL	FALSE	Ausgangssignal positiver Impuls
QNEG_P	22.1	BOOL	FALSE	Ausgangssignal negativer Impuls

RMP_SOAK

Beschreibung RMP_SOAK

Mit der Anweisung RMP_SOAK geben Sie hauptsächlich in Abhängigkeit vom zeitlichen Verlauf des Prozesses unterschiedliche Sollwerte vor.

Arbeitsweise

In einem globalen Datenbaustein legen Sie die Stützpunkt der Fahrkurve ab. Bei jedem Aufruf der Anweisung wird der für diese Zeit gültige Sollwert ausgegeben.

Der Globaler Datenbaustein DB_RMPSK (Seite 6029) ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Die Anzahl der Stützpunkte geben Sie im globalen Datenbaustein in der Variablen NBR_PTS ein. Die Nummer des globalen Bausteins geben Sie am Eingangsparameter DB_NBR an.

Für jeden Stützpunkt geben den Sollwert in PI[i].OUTV ein, und die Zeit in PI[i].TMV.

Beispiel einer Fahrkurve mit Startpunkt und Stützpunkten.

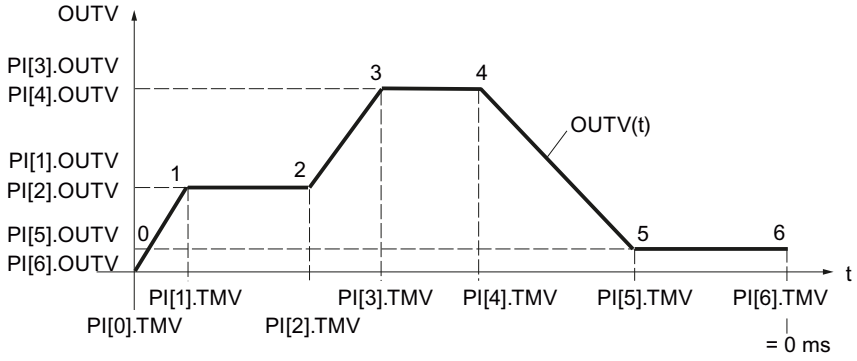

Bei n Stützpunkten ist der Zeitwert für Stützpunkt n=0 ms.

Die Zählweise der Stützpunkt–Werte und –Zeiten ist in folgendem Bild deutlicher zu erkennen:

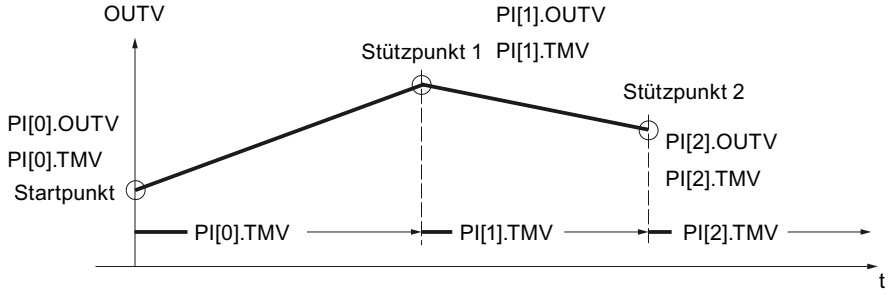

Zwischen den Stützpunkten interpoliert der Zeitplangeber nach folgender Funktion mit 0 ≤ n < (NBR_PTS - 1):

$$OUTV(t) = PI[n + 1].OUTV - \frac{RS_TM}{PI[n].TMV} (PI[n + 1].OUTV - PI[n].OUTV)$$

Hinweis

Die Anweisung überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch ob der Parameter DB_NBR.NBR_PTS (Anzahl der Stützpunkte) zur Länge des Datenbausteins passt. Bei falscher Parametrierung geht die CPU mit der Meldung "interner Systemfehler" in den Betriebszustand STOP.

Funktionen des Zeitplangebers

Die Anweisung verfügt über folgende Funktionen:

- Zeitplangeber einschalten für einmaligen Durchlauf
- Ausgang des Zeitplangebers mit festem Wert vorbelegen
- Zyklischen Betrieb des Zeitplangebers einschalten
- Bearbeitung des Zeitplangebers anhalten

- Bearbeitungsschritt und –zeit vorgeben (die Restzeit RS_TM und die Stützpunktnummer TM_SNBR werden neu definiert)
- Gesamtbearbeitungszeit und Gesamtrestzeit aktualisieren

Zur Einstellung einer gewünschten Betriebsart gilt für die Wertigkeit der Steuereingänge folgende Tabelle:

Betriebsart	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Ausgangssignal OUTV
Zeitplangeber ein	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) Endwert wird nach Bearbeitungsende gehalten.
Ausgang vorbelegen	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV
Zykl. Betrieb ein	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) nach Ende: automatischer Start
Zeitplangeber anhalten	TRUE	FALSE	TRUE	FALSE	**)	**)	aktueller Wert von OUTV(t) wird gehalten *)
Bearbeitungsschritt und Bearbeitungszeit vorgeben	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (alt) *)
			FALSE				Es wird mit neu vorgegebenen Werten weitergefahren.
Gesamtzeit aktualisieren	**)	**)	**)	**)	**)	FALSE	beeinflusst OUTV nicht
						TRUE	beeinflusst OUTV nicht
*) Bis zum nächsten Stützpunkt hat die Fahrkurve nicht die vom Anwender parametrierte Form! **) Die jeweils eingestellte Betriebsart wird unabhängig von der Wertigkeit der Steuersignale in den mit **) gekennzeichneten Feldern ausgeführt							

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Bei DFOUT_ON=TRUE wird DF_OUTV ausgegeben. Die Zeitabschnitte (0 bis NBRPTS-1) zwischen den Stützpunkten 0 bis NBRPTS werden aufsummiert und unter T_TM zur Verfügung gestellt. Der Ausgang QR_S_ACT wird zurückgesetzt; die Ausgänge NBR_ATMS und RS_TM werden mit 0 vorbesetzt.

Einschalten des Zeitplangebers

Bei einer steigenden Flanke an RMPSK_ON wird die Zeitplangeberfunktion eingeschaltet. Nach Erreichen des letzten Stützpunktes ist die Fahrkurve beendet. Um den Zeitplangeber erneut zu starten, müssen Sie erst RMPSK_ON = FALSE setzen und dann RMPSK_ON = TRUE.

Ausgang vorbelegen, Fahrkurve starten

Soll die Fahrkurve mit einem bestimmten Ausgabewert beginnen, dann muss DFOUT_ON = TRUE sein. In diesem Falle steht am Ausgang des Zeitplangebers der Signalwert DF_OUTV an.

Hinweis

Das Signal für die Ausgabe des konstanten Sollwertes DFOUT_ON hat höhere Priorität als das Startsignal für den Zeitplangeber RMPSK_ON.

Nach dem Umschalten DFOUT_ON = FALSE wird OUTV linear vom eingestellten Sollwert (DF_OUTV) zum Ausgangswert der aktuellen Stützpunktnummer PI[NBR_ATMS].OUTV gefahren.

Die interne Zeitbearbeitung läuft auch bei durchgeschaltetem festem Sollwert (RMPSK_ON = TRUE und DFOUT_ON = TRUE) weiter.

Beim Start der Fahrkurve RMPSK_ON = TRUE wird so lange der feste Sollwert DF_OUTV ausgegeben, bis DFOUT_ON nach der Dauer von T* von TRUE nach FALSE wechselt.

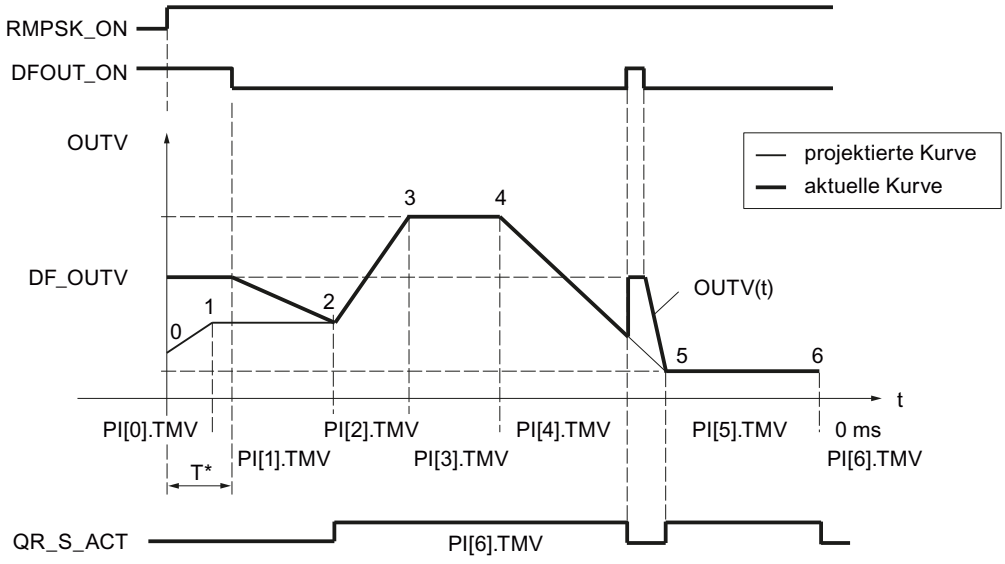

Zu diesem Zeitpunkt ist die Zeit PI[0].TMV und ein Teil der Zeit PI[1].TMV abgelaufen. OUTV wird von DF_OUTV nach PI[2].OUTV gefahren, d.h. zum Sollwert am Stützpunkt 2.

Erst ab dem Stützpunkt 2 wird die projektierte Fahrkurve ausgegeben. Die Übereinstimmung von projektierter und aktueller Fahrkurve wird angezeigt, indem QR_S_ACT = TRUE wird. Bei einer steigenden Flanke an DFOUT_ON während der Fahrkurvenbearbeitung springt der ausgegebene Wert OUTV ohne Verzögerung auf DF_OUTV.

Zyklischer Betrieb eingeschaltet

Ist die Betriebsart 'Zyklische Wiederholung' (CYC_ON = TRUE) eingeschaltet, dann kehrt der Zeitplangeber nach Ausgabe des letzten Stützpunktwertes automatisch zum Startpunkt zurück und beginnt einen neuen Durchlauf.

Zwischen dem letzten Stützpunkt und dem Startpunkt wird nicht interpoliert. Für stoßfreien Übergang muss gelten: PI[NBR_PTS].OUTV = PI[0].OUTV.

Zeitplangeber anhalten

Mit RMP_HOLD = TRUE wird der Wert der Ausgangsgröße (inkl. der Zeitbearbeitung) eingefroren. Bei Rücksetzen RMP_HOLD = FALSE wird an der Unterbrechungsstelle PI[x].TMV fortgefahren.

Die Bearbeitungszeit der Fahrkurve verlängert sich um die Zeit T*. Die Fahrkurve hat vom Startpunkt bis zur steigenden Flanke an RMP_HOLD und vom Stützpunkt 5* bis zum Stützpunkt 6* den projektierten Verlauf, d.h. das Ausgangssignal QR_S_ACT hat den Wert TRUE.

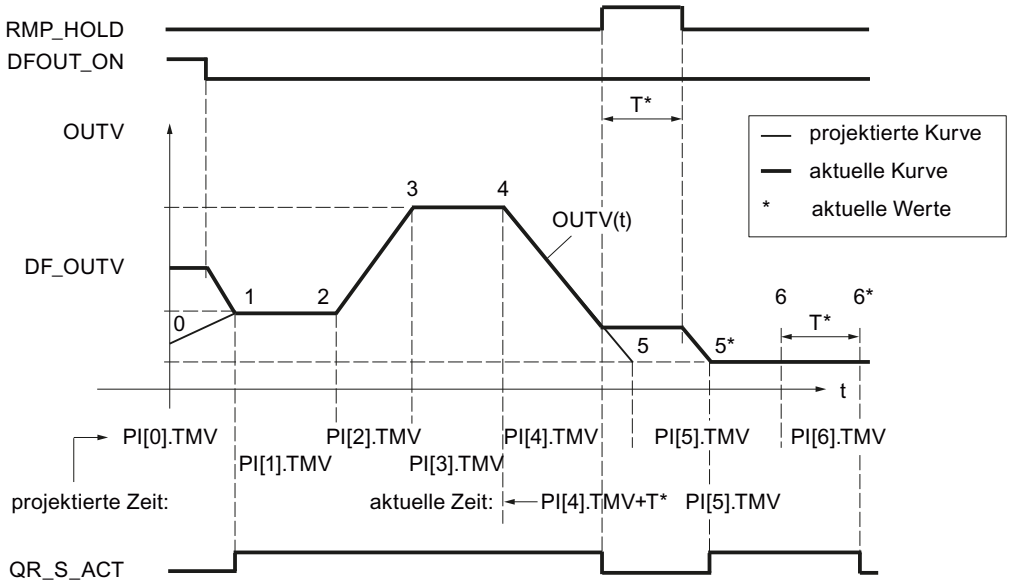

Bearbeitungsschritt und –zeit vorgeben

Mit CONT_ON wird eine angehaltene Fahrkurve an eine definierten Stelle wieder aufgenommen. Wenn der Zeitplangeber nicht angehalten ist, hat CONT_ON keine Auswirkungen.

Ist der Steuereingang für das Fortsetzen CONT_ON = TRUE gesetzt, dann wird der angehaltene Zeitplangeber am Stützpunkt TM_SNBR fortgesetzt. TM_CONT bestimmt die Restzeit, die der Zeitplangeber benötigt, bis der Stützpunkt TM_SNBR erreicht wird.

Das folgende Bild zeigt den Einfluss von RMP_HOLD und CONT_ON bei:

TM_SNBR = 5 und TM_CONT = T*

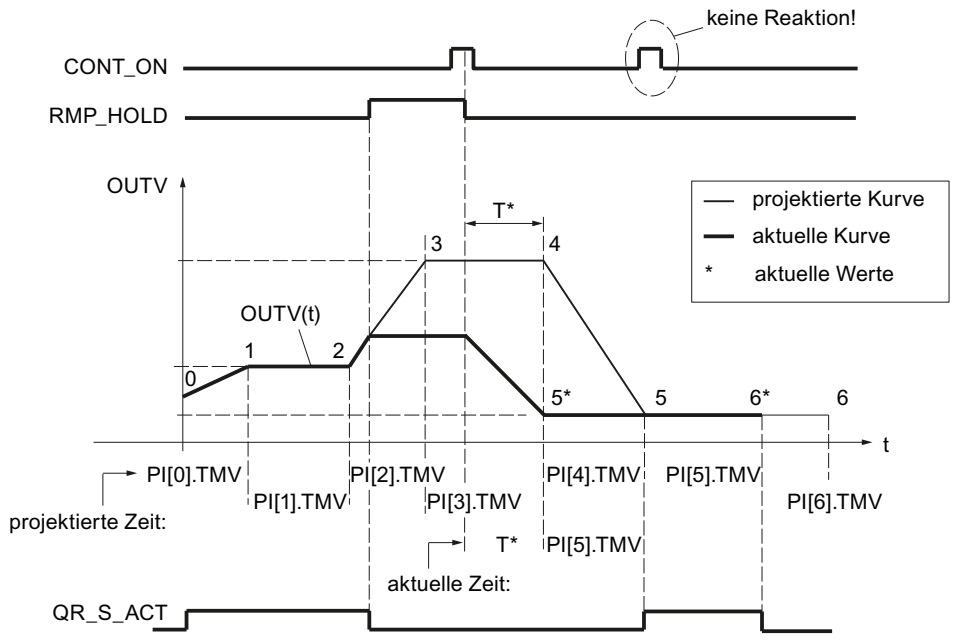

Die Stützpunkte 3 und 4 entfallen in diesem Zyklus des Zeitplangebers.

Nach Signalwechsel RMP_HOLD von TRUE nach FALSE wird der projektierte Kurvenverlauf erst ab Stützpunkt 5 wieder erreicht.

Gesamtzeit und Gesamtrestzeit aktualisieren

In jedem Zyklus werden die aktuelle Stützpunktnummer NBR_ATMS, die aktuelle Restzeit bis zum Erreichen des Stützpunktes RS_TM, die Gesamtzeit T_TM und die Gesamtrestzeit bis zum Erreichen des Fahrkurvenendes RT_TM aktualisiert.

Bei Online–Änderungen von PI[n].TMV ändert sich die Gesamtzeit und die Gesamtrestzeit der Fahrkurve. Da die Berechnung von T_TM und RT_TM bei vielen Stützpunkten die Bearbeitungszeit des Funktionsbausteins stark vergrößert, wird sie nur nach Neustart oder bei TUPDT_ON = TRUE durchgeführt. Die Zeitabschnitte PI[0 ... NBR_PTS].TMV zwischen den einzelnen Stützpunkten werden aufsummiert und am Ausgang Gesamtzeit T_TM und Gesamtrestzeit RT_TM angezeigt.

Bitte beachten Sie, dass die Ermittlung der Gesamtzeiten relativ viel CPU–Laufzeit erfordert!

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter RMP_SOAK

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
DF_OUTV	0.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
DB_NBR	4.0	BLOCK_DB	DB 1	Datenbausteinnummer, global (Stützpunktwerte) CPU-abhängig
TM_SNBR	6.0	INT	0	Nr. des nächsten Stützpunktes zum Weitermachen 0 - 255
TM_CONT	8.0	TIME	T#0s	Restzeit zum Weitermachen bis zum Stützpunkt TM_SNBR technischer Wertebereich
DFOUT_ON	12.0	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
RMPSK_ON	12.1	BOOL	FALSE	Zeitplangeber einschalten
HOLD	12.2	BOOL	FALSE	Einfrieren der Ausgangsgröße
CONT_ON	12.3	BOOL	FALSE	Weitermachen
CYC_ON	12.4	BOOL	FALSE	zyklische Wiederholung einschalten
TUPDT_ON	12.5	BOOL	FALSE	Gesamtzeit nach Verändern der Zeitwerte neu berechnen (verlängerte Bausteinlaufzeit !)
COM_RST	12.6	BOOL	FALSE	Neustart
CYCLE	14.0	TIME	T#1s	Abtastzeit technischer Wertebereich ≥ 1ms

Ausgangsparameter RMP_SOAK

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	18.0	REAL	0.0	Ausgangsgröße
QR_S_ACT	22.0	BOOL	FALSE	Zeittabelle wird bearbeitet
NBR_ATMS	24.0	INT	0	aktuelle Stützpunktnummer (Stützpunkt, der angefahren wird)
RS_TM	26.0	TIME	T#0s	aktuelle Restzeit
T_TM	30.0	TIME	T#0s	globale Daten : Gesamtzeit ΣTMVI
RT_TM	34.0	TIME	T#0s	globale Daten : Gesamtrestzeit ΣTMVI

Die Stützpunkte sowie die Anzahl der Stützpunkte NBRPTS sind in einem globalen Datenbaustein DB_NBR) abgelegt. Die Ausgabe beginnt mit Stützpunkt 0 und endet mit Stützpunkt NBR_PTS.

Globaler Datenbaustein DB_RMPSK

Die Stützpunkte für den Zeitplangeber (RMP_SOAK) werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
NBR_PTS	INT	4	Anzahl der Stützpunkte -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Struktur für die Daten der Stützpunkte. Das Array muss von 0...NBR_PTS deklariert werden.
PI[0].OUTV	REAL	0.0	Ausgangswert [0] Startpunkt technischer Wertebereich
PI[0].TMV	TIME	T#1s	Zeitwert [0] Startpunkt
PI[1].OUTV	REAL	0.0	Ausgangswert [1] Stützpunkt 1 technischer Wertebereich
PI[1].TMV	TIME	T#1s	Zeitwert [1] Stützpunkt 1
PI[2].OUTV	REAL	0.0	Ausgangswert [2] Stützpunkt 2 technischer Wertebereich
PI[2].TMV	TIME	T#1s	Zeitwert [2] Stützpunkt 2
PI[3].OUTV	REAL	0.0	Ausgangswert [3] Stützpunkt 3 technischer Wertebereich
PI[3].TMV	TIME	T#1s	Zeitwert [3] Stützpunkt 3
PI[4].OUTV	REAL	0.0	Ausgangswert [4] Stützpunkt 4 technischer Wertebereich
PI[4].TMV	TIME	T#0s	Zeitwert [4] Stützpunkt 4 0 ms

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den untenstehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen ".DB":
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Externe Quellen".
6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.

10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_RMPSK"
TITLE = data block ramp soak
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_RMPSK
 STRUCT
 NBR_PTS : Int; // number of points (excluding starting point)
 PI : Array[0..4] of Struct
 OUTV : Real; // output variable
 TMV : Time; // output time value
 END_STRUCT;
 END_STRUCT;


BEGIN
 NBR_PTS := 4;
 PI[0].TMV := T#1s;
 PI[1].TMV := T#1s;
 PI[2].TMV := T#1s;
 PI[3].TMV := T#1s;
 PI[4].TMV := T#0s;
END_DATA_BLOCK
```

ROC_LIM

Beschreibung ROC_LIM

Rampenfunktionen werden dann verwendet, wenn der Prozess keinen Eingangssprung verträgt. Dies ist zum Beispiel der Fall, wenn zwischen Motor und anzutreibender Last ein Getriebe zwischengeschaltet ist, und ein zu schnelles Ansteigen der Motordrehzahl zu einer Überlastung des Getriebes führen würde. Die Anweisung begrenzt die Änderungsgeschwindigkeit eines Ausgangswertes. Aus Sprungfunktionen werden Rampenfunktionen.

Arbeitsweise

Für den positiven und negativen Wertebereich der Eingangsgröße INV sind insgesamt vier Rampen parametrierbar, die für die Ermittlung der Ausgangsgröße verwendet werden:

- UPRLM_P: Steigende Werte im positiven Bereich
  OUTV > 0 und |OUTV| steigend
- DNRLM_P: Fallende Werte im positiven Bereich
  OUTV > 0 und |OUTV| fallend
- UPRLM_N: Steigende Werte im negativen Bereich
  OUTV < 0 und |OUTV| steigend
- DNRLM_N: Fallende Werte im negativen Bereich
  OUTV < 0 und |OUTV| fallend

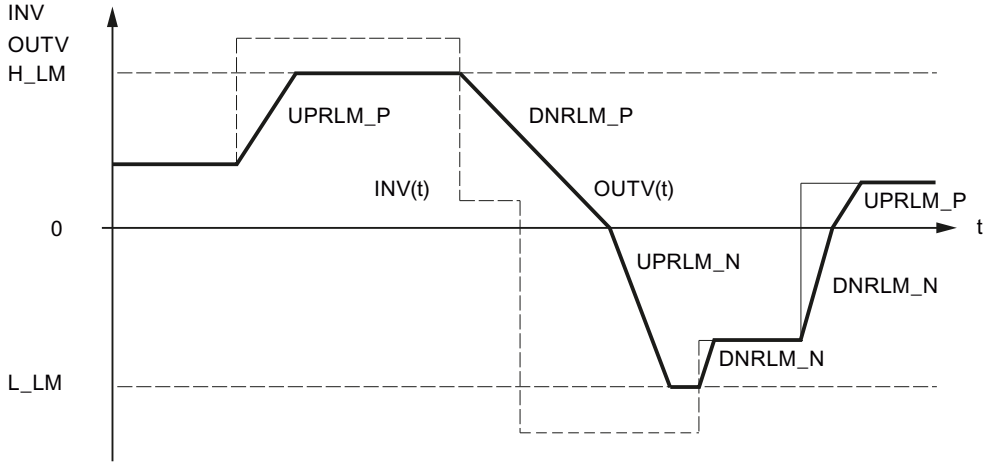

Die parametrierten Werte der Rampen beziehen sich auf einen Anstieg/Abfall der Ausgangsgröße pro Sekunde.

Wenn UPRLM_P = 10.0 und |INV| steigend, darf sich OUTV in 1 s maximal um 10.0 erhöhen.

Steigt die Eingangsgröße stärker, wird OUTV in Abhängigkeit von der Abtastzeit CYCLE pro Aufruf folgendermaßen erhöht:

CYCLE [s]	Erhöhung von OUTV
1	10.0
0.1	1.0
0.01	0.1

Die Begrenzung der Änderungsgeschwindigkeit wird an den QUPRLM_P, QDNRLM_P, QUPRLM_N und QDNRLM_N, angezeigt.

Die Ausgangsgröße kann durch H_LM und L_LM begrenzt werden. Dies wird an QH_LM und QL_LM angezeigt.

Funktionen der Anweisung

Über die Parameter DFOUT_ON, TRACK und MAN_ON können Sie die Arbeitsweise beeinflussen.

- DFOUT_ON = TRUE
  Wenn DFOUT_ON = TRUE gesetzt ist, wird DF_OUTV am Ausgang OUTV ausgegeben; bei fallender Flanke wird OUTV von DF_OUTV nach INV und bei steigender Flanke wird OUTV von INV nach DF_OUTV gefahren.
- TRACK = TRUE
  Zum Nachführen OUTV = INV wird das Bit TRACK = TRUE gesetzt; da die Eingangsgröße direkt zur Ausgangsgröße durchgeschaltet wird, werden auch Sprünge der Eingangsgröße ausgegeben.
- MAN_ON = TRUE
  Mit MAN_ON = TRUE können Sie stoßfrei zwischen Hand- und Automatikbetrieb umschalten. Hierzu muss die Anweisung ROC_LIM im Sollwertzweig direkt vor der Regeldifferenz eingebaut werden. Am Eingang PV wird der Istwert und am Eingang MAN_ON das Hand-Automatik-Bit, z. B. LMNGEN_C.MAN_ON verschaltet.

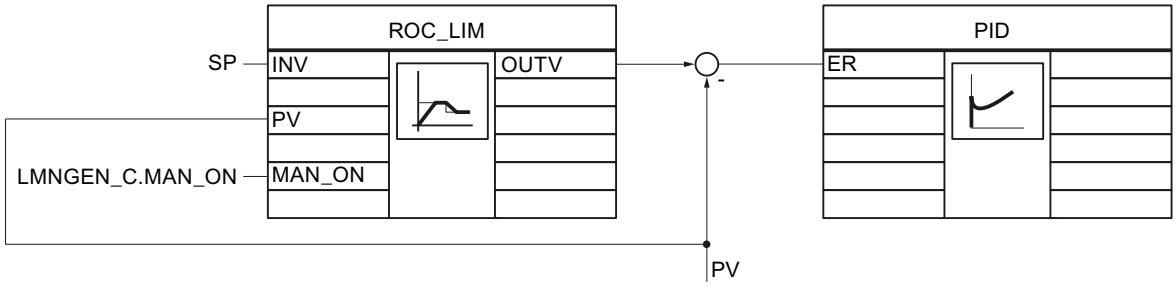

Beim Umschalten auf Betriebsart Hand MAN_ON = TRUE wird der am Eingang PV anliegende Wert sofort zum Ausgang OUTV durchgeschaltet. Da Sollwert und Istwert gleich sind, wird die Regeldifferenz zu Null und der Regler befindet sich in einem stationären Ruhezustand. Beim Rücksetzen in den Automatikbetrieb MAN_ON = FALSE fährt der Ausgang OUTV vom momentan anliegenden Wert PV rampenförmig zum Eingangswert INV. Dies bewirkt ein stoßfreies Umschalten der Reglerstruktur vom Handbetrieb in den Automatikbetrieb.
Bei MAN_ON = TRUE sind die Begrenzungen H_LM und L_LM unwirksam, DFOUT_ON wird nicht berücksichtigt.

Beispiel 1

Bei MAN_ON, TRACK, DFOUT_ON gleich FALSE ergeben sich folgende Verläufe der Signalausgänge:

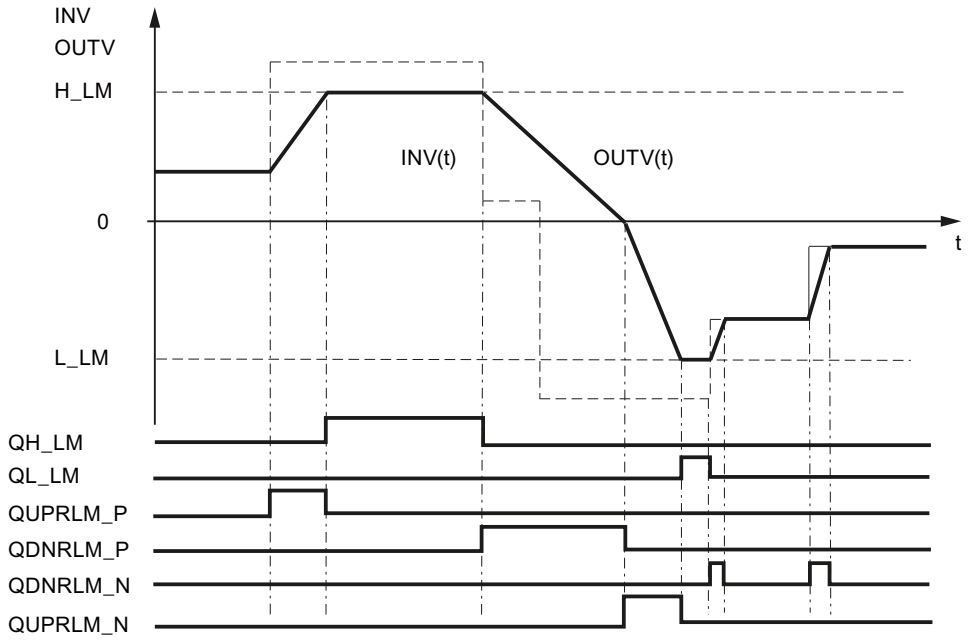

Beispiel 2

Stoßfreie Hand-Automatik-Umschaltung

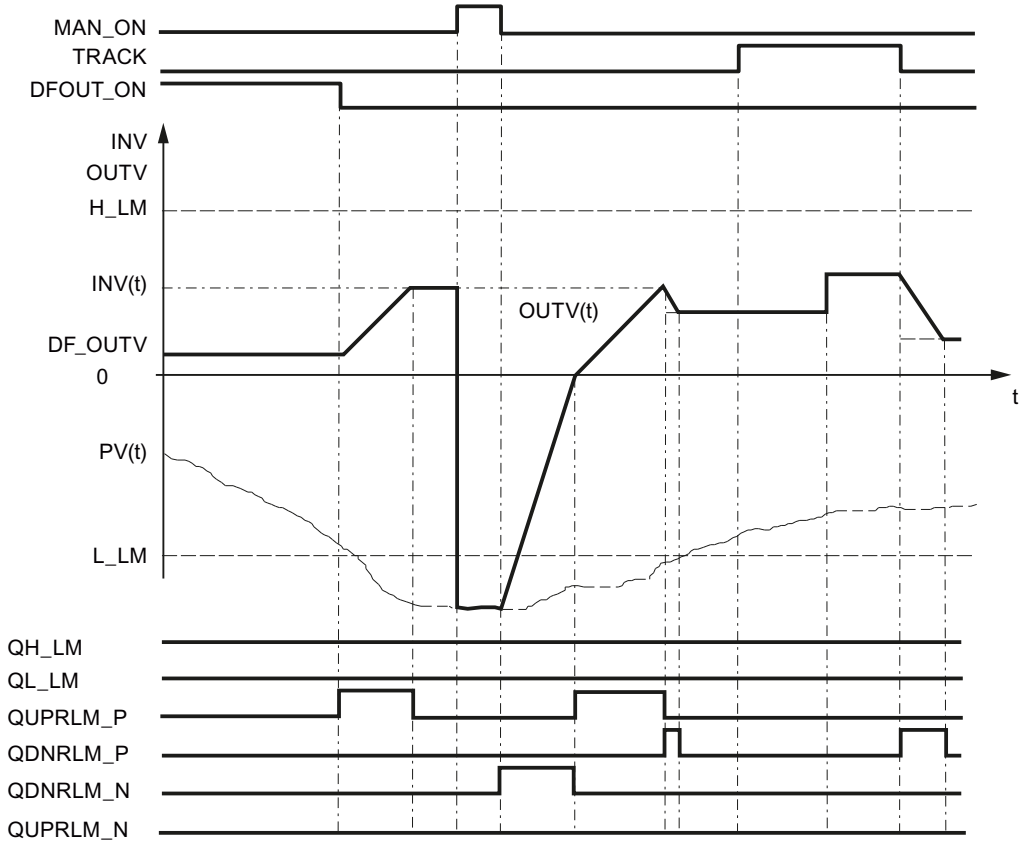

Beispiel 3

Rampe, die nur im positiven Wertebereich eingesetzt ist.

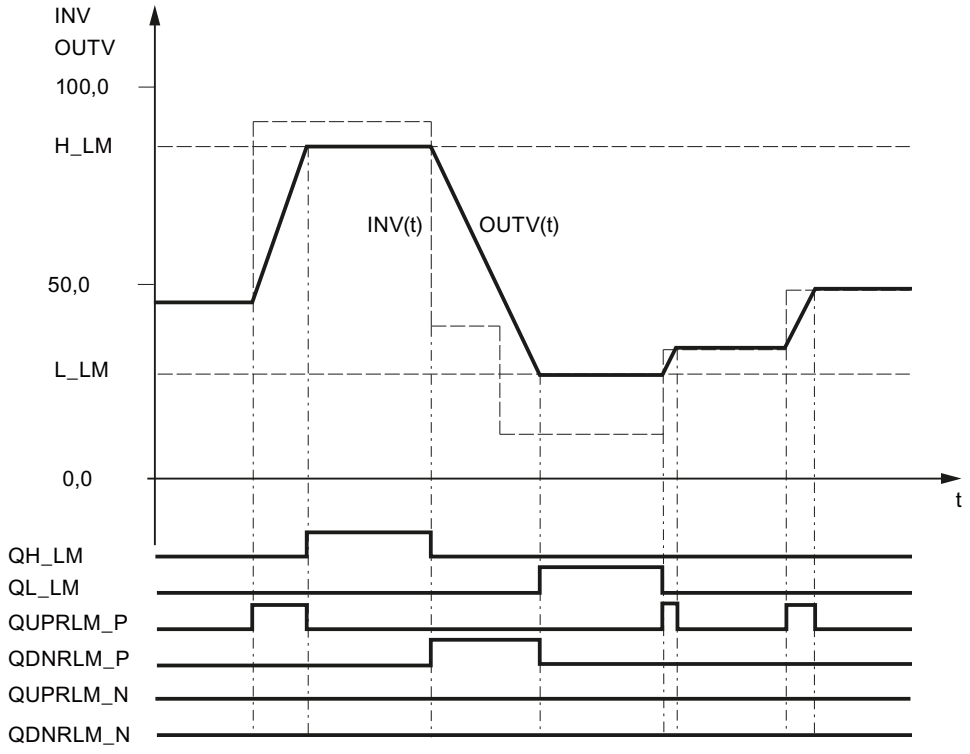

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 rückgesetzt. Wenn DFOUT_ON = TRUE gesetzt ist, wird DF_OUTV ausgegeben. Alle Signalausgänge werden auf FALSE gesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter ROC_LIM

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
UPRLM_P	4.0	REAL	10.0	Anstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde [1/s] > 0.0

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
DNRLM_P	8.0	REAL	10.0	Abstiegsbegrenzung im positiven Bereich Inkrement pro Sekunde [1/s] > 0.0
UPRLM_N	12.0	REAL	10.0	Anstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde [1/s] > 0.0
DNRLM_N	16.0	REAL	10.0	Abstiegsbegrenzung im negativen Bereich Inkrement pro Sekunde [1/s] > 0.0
H_LM	20.0	REAL	100.0	obere Begrenzung technischer Wertebereich > L_LM
L_LM	24.0	REAL	0.0	untere Begrenzung technischer Wertebereich < H_LM
PV	28.0	REAL	0.0	Regelgröße technischer Wertebereich
DF_OUTV	32.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
DFOUT_ON	36.0	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
TRACK	36.1	BOOL	FALSE	Nachführen OUTV=INV
MAN_ON	36.2	BOOL	FALSE	Handbetrieb einschalten
COM_RST	36.3	BOOL	FALSE	Neustart
CYCLE	38.0	TIME	T#1s	Abtastzeit

Ausgangsparameter ROC_LIM

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	42.0	REAL	0.0	Ausgangsgröße
QUPRLM_P	46.0	BOOL	FALSE	Anstiegsbegrenzung im positiven Bereich angesprochen
QDNRLM_P	46.1	BOOL	FALSE	Abstiegsbegrenzung im positiven Bereich angesprochen
QUPRLM_N	46.2	BOOL	FALSE	Anstiegsbegrenzung im negativem Bereich angesprochen
QDNRLM_N	46.3	BOOL	FALSE	Abstiegsbegrenzung im negativen Bereich angesprochen
QH_LM	46.4	BOOL	FALSE	obere Begrenzung angesprochen
QL_LM	46.5	BOOL	FALSE	untere Begrenzung angesprochen

SCALE_M

Beschreibung SCALE_M

Bei Istwerten liegt der vom Geber gelieferte Wert oft in einem für den Anwender ungünstigen Bereich, z. B. 0 bis 10 V entsprechen 0 bis 1200 ºC oder 0 bis 10 V entsprechen 0 bis 3000 U/min. Durch Anpassung von Soll- oder Istwert liegen beide Prozessgrößen im gleichen Wertebereich.

Arbeitsweise

Die Anweisung normiert eine analoge Eingangsgröße. Die analoge Eingangsgröße INV wird an der Normierungsgerade in die Ausgangsgröße OUTV überführt. Die Normierungsgerade wird durch die Steigung (FACTOR) und dem Abstand (OFFSET) zwischen OUTV bei INV=0 und der Koordinatenachse OUTV=0 definiert.

Formel:

OUTV = INV * FACTOR + OFFSET

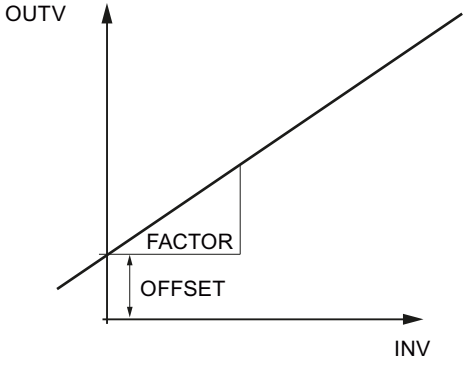

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter SCALE_M

Parameter	Offset	Datentyp	Vorbelegung	Beschreibug
INV	0.0	REAL	0.0	Eingangsgröße technischer Wertebereich
FACTOR	4.0	REAL	1.0	Skalierungsfaktor
OFFSET	8.0	REAL	0.0	Offset technischer Wertebereich

Ausgangsparameter SCALE_M

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV	12.0	REAL	0.0	Ausgangsgröße

SP_GEN

Beschreibung SP_GEN

Zur Sollwertvorgabe von Hand kann mit der Anweisung SP_GEN über zwei Eingänge ein Ausgangswert verändert werden. Für eine feinstufige Änderung sollte die Anweisung eine Abtastzeit von ≤100 ms haben.

Arbeitsweise

DFOUT_ON, OUTVUP und OUTVDN haben folgenden Einfluss auf OUTV:

DFOUT_ON	OUTVDN	OUTVUP	OUTV
TRUE	beliebig	beliebig	DF_OUTV
FALSE	TRUE	TRUE	OUTV unverändert
	FALSE		OUTV steigend
	TRUE	FALSE	OUTV fallend
	FALSE		OUTV unverändert

Wenn DFOUT_ON = TRUE gesetzt ist, wird am Ausgang DF_OUTV ausgegeben. Die Änderung von OUTV erfolgt sprungförmig. Die Umschaltung nach DFOUT_ON = FALSE ist stoßfrei.

Über den Eingang OUTVUP kann die Ausgangsgröße OUTV innerhalb der Begrenzungen H_LM und L_LM kontinuierlich vergrößert und über OUTVDN verkleinert werden. Die Änderungsgeschwindigkeit von OUTV hängt von den Begrenzungen ab.

- während der ersten 3 s nach Setzen von OUTVUP oder OUTVDN gilt:
  dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
- danach
  dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s

Das Bild zeigt Veränderung des Ausgangswertes durch Setzen von OUTVUP.

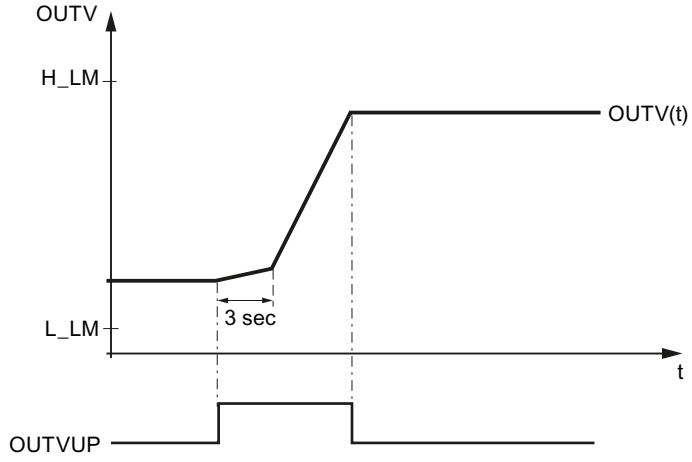

OUTVUP und OUTVDN werden nur ausgeführt, wenn DFOUT_ON = FALSE.

Die Ausgangsgröße OUTV wird durch H_LM und L_LM begrenzt. Dies wird an QH_LM und QL_LM angezeigt.

Das Bild zeigt den Einfluss von OUTV durch OUTVUP, OUTVDN und DFOUTV_ON.

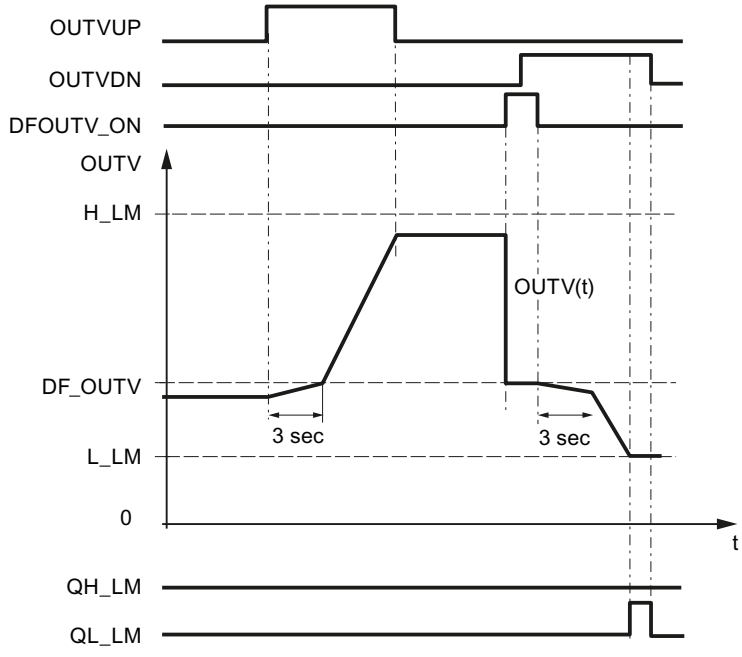

Neustart

Bei Neustart wird der Ausgang OUTV auf 0.0 gesetzt. Wenn DFOUT_ON = TRUE ist, wird DF_OUTV ausgegeben. Die Begrenzungen sind auch bei Neustart wirksam.

Bausteininterne Begrenzungen

Bausteinintern werden keine Werte begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter SP_GEN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
DF_OUTV	0.0	REAL	0.0	Vorbesetzung der Ausgangsgröße technischer Wertebereich
H_LM	4.0	REAL	100.0	obere Begrenzung technischer Wertebereich > L_LM
L_LM	8.0	REAL	0.0	untere Begrenzung technischer Wertebereich < H_LM
OUTVUP	12.0	BOOL	FALSE	OUTV Hoch
OUTVDN	12.1	BOOL	FALSE	OUTV Tief
DFOUT_ON	12.2	BOOL	FALSE	Vorbesetzung der Ausgangsgröße ein
COM_RST	12.3	BOOL	FALSE	Neustart
CYCLE	14.0	TIME	T#100ms	Abtastzeit

Ausgangsparameter SP_GEN

Parameter	Offset	Datentyp	Vorbelegung (nach Neustart)	Beschreibung
OUTV	18.0	REAL	0.0	Ausgangsgröße
QH_LM	22.0	BOOL	FALSE	obere Begrenzung angesprochen
QL_LM	22.1	BOOL	FALSE	untere Begrenzung angesprochen

SPLT_RAN

Beschreibung SPLT_RAN

Die Anweisung wird z. B benötigt, um eine Temperaturregelung auf ein Stellglied zur Kühlung und ein Stellglied zum Heizen aufzuteliien.

Arbeitsweise

Der Stellwert eines PID–Reglers wird aufgeteilt in mehrere Bereiche. Pro Bereich muss die Anweisung einmal aufgerufen werden und mit einem der Stellwertverarbeitungsbausteine LMNGEN_C oder LMNGEN_S verschaltet werden. Wenn Sie LMNGEN_S verschalten, muss die Stellungsrückmeldung aktiviert sein (LMNR_ON = TRUE).

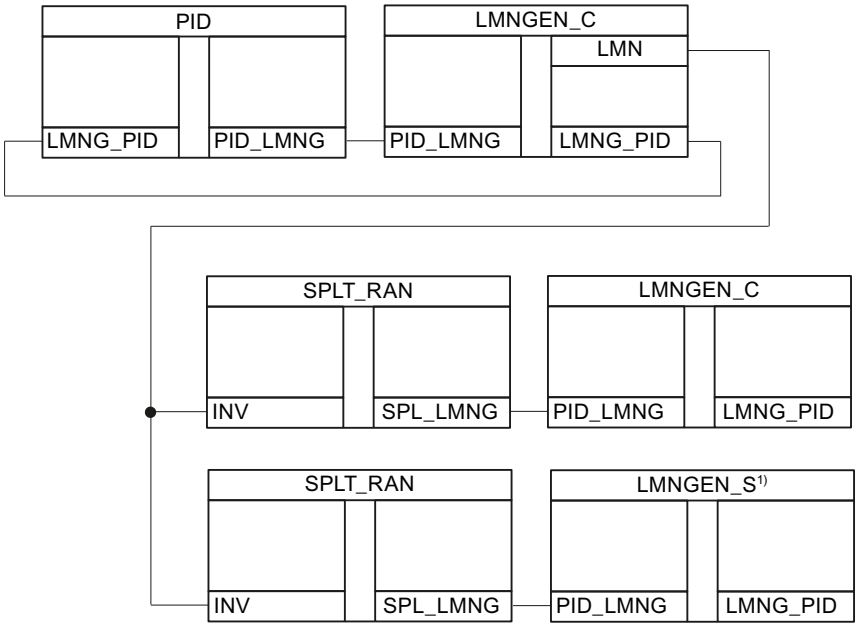

Wenn der Eingangswert INV innerhalb des Bereichs STR_INV und EDR_INV liegt, wird der Ausgangswert SPL_LMNG.PID_OUTV innerhalb des Bereichs STR_OUTV und EDR_OUTV ausgegeben. Wenn INV < STR_INV, wird STR_OUTV ausgegeben. Wenn INV > STR_INV, wird EDR_OUTV ausgegeben.

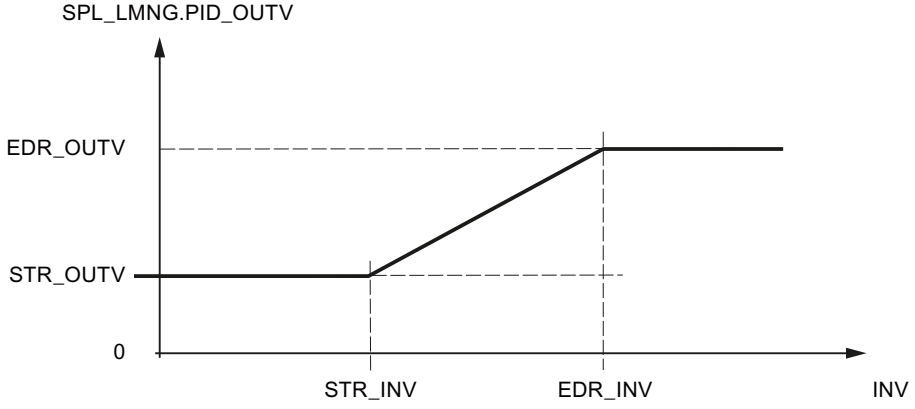

Neustart

Die Anweisung hat keine Neustartroutine.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter SPLT_RAN

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV	0.0	REAL	0.0	Eingangsvariable technischer Wertebereich
STR_INV	4.0	REAL	0.0	Bereichsanfang INV technischer Wertebereich
EDR_INV	8.0	REAL	50.0	Bereichsende INV technischer Wertebereich
STR_OUTV	12.0	REAL	0.0	Bereichsanfang OUTV technischer Wertebereich
EDR_OUTV	16.0	REAL	100.0	Bereichsende OUTV technischer Wertebereich

Ausgangsparameter SPLT_RAN

Parameter	Offset	Datentyp	Beschreibung
SPL_LMNG	20.0	STRUC	PID-LMNGEN Schnittstelle

SWITCH

Beschreibung SWITCH

Die Anweisung findet Anwendung als Eingangs- und/oder Ausgangsmultiplexer von 2 Ein-/Ausgangsgrößen.

Arbeitsweise

Die Anweisung schaltet einen von 2 analogen Eingangswerten auf einen von 2 Ausgangswerten. Die Schalter INV1_ON und OUTV1_ON bestimmen, wie die Anweisung arbeitet.

INV1_ON	OUTV1_ON	OUTV1	OUTV2
FALSE	FALSE	unverändert	INV2
TRUE	FALSE	unverändert	INV1
FALSE	TRUE	INV2	unverändert
TRUE	TRUE	INV1	unverändert

Unverändert bedeutet, dass an dem Ausgangsparameter der Wert aus dem letzten Zyklus stehen bleibt.

Neustart

Bei Neustart werden OUTV1 = 0.0 und OUTV2 = 0.0 gesetzt.

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter SWITCH

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
INV1	0.0	REAL	0.0	Eingangsgröße 1 technischer Wertebereich
INV2	4.0	REAL	0.0	Eingangsgröße 2 technischer Wertebereich
INV1_ON	8.0	BOOL	FALSE	Eingangsgröße INV1 wird durchgeschaltet
OUTV1_ON	8.1	BOOL	FALSE	Ausgangsgröße OUTV1 wird durchgeschaltet
COM_RST	8.2	BOOL	FALSE	Neustart

Ausgangsparameter SWITCH

Parameter	Offset	Datentyp	Vorbelegung	Beschreibung
OUTV1	10.0	REAL	0.0	Ausgangsgröße 1
OUTV2	14.0	REAL	0.0	Ausgangsgröße 2

LP_SCHED_M

Beschreibung LP_SCHED_M

Müssen viele Regler mit unterschiedlichen Abtastzeiten, insbesondere träge Regelungen mit großen Abtastzeiten, aufgerufen werden, dann reicht der Umfang des Prioritätsklassenmodells in Bezug auf nutzbare Weckalarmebenen nicht aus. Der Reglerauffrufverteiler (LP_SCHED) macht es nun möglich, in einer Weckalarmebene mehrere Regler mit unterschiedlichen Abtastzeiten unterzubringen. Die einzelnen Regler werden dabei mit ihren Abtastzeiten zyklisch aufgerufen.

Der Einsatz des Aufrufverteilers ist nicht obligatorisch. Die Regler-Anweisungen können auch ohne die Verteilungsfunktion direkt aus dem OB aufgerufen werden.

Die Aufrufverteilung mehrerer Regler in einer Weckalarmebene ist in der Anweisung LP_SCHED realisiert. Die Anweisung muss **vor** allen Regelkreisen aufgerufen werden. Die Daten für die einzelnen Regleraufrufe sind in einem globalen Datenbaustein (DB_LOOP) hinterlegt.

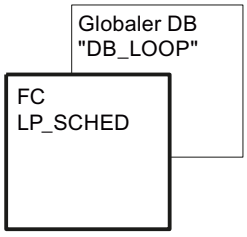

Globaler Datenbaustein DB_LOOP (Seite 6048) ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Der Aufrufverteiler bearbeitet den globalen Datenbaustein und setzt abhängig von der Reihenfolge und den parametrierten Abtastzeiten der Regler die ENABLE–Bits. Der Zeittakt der Weckalarmebene wird dadurch untersetzt. Die einzelnen Regelkreise in dieser Weckalarmebene werden entsprechend ihrer eingestellten Abtastzeit aufgerufen und bearbeitet. Nach dem Bausteinaufruf muss das ENABLE–Bit wieder zurückgesetzt werden. Die Bausteinaufrufe und das Rücksetzen der ENABLE–Bits müssen programmiert werden.

Der Aufruf einzelner Regelkreise kann manuell gesperrt werden. Auch lassen sich einzelne Regelkreise rücksetzen (Neustart).

Hinweis

Die Anweisung überprüft weder, ob ein globaler DB mit der Nummer DB_NBR wirklich vorhanden ist, noch, ob der Parameter GLP_NBR (größte Regelkreisnummer) zur Länge des Datenbausteins passt. Bei inkorrekter Parametrierung geht die CPU mit "internem Systemfehler" in STOP.

Arbeitsweise LP_SCHED_M

Neustart

Bei COM_RST = TRUE werden folgende Vorbelegungen eingestellt:

Aktuelle Regelkreisnummer: ALP_NBR = 0

Die Aufrufdaten aller Regelkreise bis **GLB_NBR** werden wie folgt vorbelegt:

Freigabe:	ENABLE =	NOT MAN_DIS
Abtastzeit:	CYCLE =	GV(MAN_CYC)
Neustart:	COM_RST =	TRUE
Lokale Aufrufnummer:	ILP_COU =	0
GV(MAN_CYC) = MAN_CYC wird gerundet auf ein ganzzahliges Vielfaches von TM_BASE*GLP_NBR		

Der Regelkreisaufruf über LP_SCHED_M

Die Anweisung LP_SCHED wird über drei Eingangsparameter in das Aufrufschema der CPU eingebunden. Am Eingang TM_BASE wird der Zeittakt der Weckalarmebene angegeben. Der Bearbeitungsaufruf der zugeordneten Regelkreise geschieht über einen bedingten Bausteinaufruf aus einer Weckalarmebene (z. B. OB 35). Hierbei werden die ENABLE–Bits im globalen Datenbaustein abgefragt.

Falls der Aufruf aus der Neustartebene heraus erfolgt, wird der Eingang COM_RST = TRUE gesetzt. In der Weckalarmebene muss dieser Aufruf wieder auf FALSE zurückgesetzt werden. Der globale Datenbaustein (siehe vorherige Tabelle) mit den zeitrelevanten Daten der Regelkreise in der betreffenden Weckalarmebene wird über den Eingangsparameter DB_NBR zugeordnet.

Parametrierung des Regelkreisaufrufs (globaler DB)

Die Parametrierung des Regleraufrufverteilers muss **ohne** Unterstützung der Projektiersoftware vorgenommen werden.

Der Datenbaustein (DB_LOOP) enthält sowohl einen Parameter, der die Anzahl der Regelkreise festlegt, die in der betreffenden Weckalarmebene insgesamt zu bearbeiten sind (max. 256), als auch einen Parameter, der die aktuell bearbeiteten Regelkreise anzeigt:

GLP_NBR	größte Regelkreisnummer
ALP_NBR	Nummer des im Zyklus bearbeiteten Regelkreises

Die Nummer des jeweiligen Regelkreises ergibt sich aus der Platzierung seiner Aufrufdaten in der Abfolge der Eintragungen im DB.

Die Aufrufdaten der einzelnen Regelkreise liegen strukturiert im Feld LOOP_DAT. Sollen Regelkreise hinzugefügt werden, so muss die Feldlänge von LOOP_DAT angepasst werden. Hierzu müssen Sie in der Deklarationsansicht den Datentyp ARRAY anpassen. Bei z.B. 10 Regelkreisen müssen Sie ARRAY[1..10] eintragen. Weiterhin müssen Sie in der Datenansicht den Parameter GLP_NBR anpassen. Er darf nie größer als die Feldlänge sein.

Die Parameter COM_RST und CYCLE in den Aufrufdaten müssen mit den entsprechenden Eingangsparametern am FB des aufgerufenen Regelkreises verschaltet werden. Diese Verschaltung ist vom Anwender zu programmieren. Ist der Parameter ENABLE gesetzt, soll der entsprechende Regelkreis aufgerufen werden. Nach dem Regleraufruf muss das ENABLE–Bit zurückgesetzt werden. Der bedingte Regleraufruf und das Rücksetzten des ENABLE–Bits müssen vom Anwender programmiert werden.

Mit Hilfe der manuell einstellbaren Parameter MAN_CYC/MAN_DIS/MAN_CRST lässt sich steuern, ob ein Regelkreises aufgerufen oder nicht aufgerufen werden kann. Diese Aufrufdaten können online - d.h. bei laufendem Betrieb - geändert werden, sofern dabei nur die Parameter überschrieben und nicht der ganze DB neu generiert wird. Es bedeutet:

MAN_CYC	Abtastzeit des betreffenden Reglers (wird in CYCLE auf ganzzahliges Vielfaches von TM_BASE * GLP_NBR gerundet)
MAN_DIS	Sperrung des Regleraufrufs
MAN_CRST	Neustart für diesen Regler

Die Aufrufbearbeitung

Der betreffende Regelkreis wird, abhängig vom Wert des ENABLE–Signals der Regleraufrufdaten, gemäß vorhergehender DB–Parametrierung bearbeitet.

Der Datenbaustein wird von oben nach unten abgearbeitet. Pro Zyklus geht der Aufrufverteiler um eine Regelkreisnummer (ALP_NBR) in der Abfolge des DB weiter. Der interne Zähler ILP_COU wird dabei jeweils um 1 heruntergezählt. Ist ILP_COU = 0, dann setzt der Aufrufverteiler das ENABLE–Bit des betreffenden Regelkreises. Das Rücksetzen der ENABLE–Bits muss der Anwender nach dem Regleraufruf programmieren.

Der Parameter MAN_CYC wird bei der Bearbeitung auf CYCLE übertragen:

CYCLE = GV (MAN_CYC), GV = ganzzahliges Vielfaches

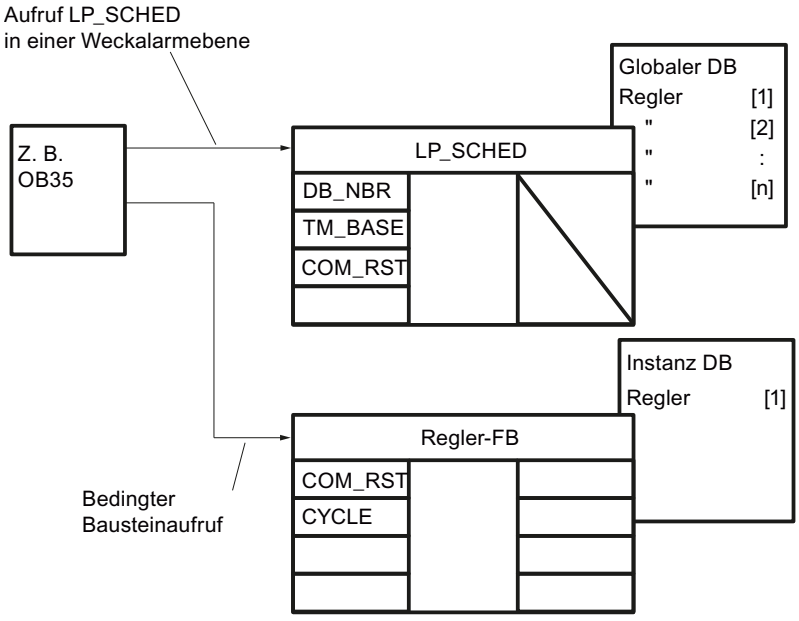

Bild 4-26 Prinzip des Regleraufrufs über den Aufrufverteiler LP_SCHED

- Einzelne Regelkreise sperren:
  Setzt man durch Parametrieren im DB das Bit "MAN_DIS", dann wird das ENABLE–Bit auf FALSE zurückgesetzt und der betreffende Regelkreis von der Bearbeitung im Aufrufverteiler ausgeschlossen.
- Einzelne Regelkreise rücksetzen (Neustart):
  Setzt man durch Parametrieren im DB das Bit "MAN_CRST", dann wird COM_RST = TRUE und MAN_CRST anschließend zurückgesetzt. Der betreffende Regelkreis wird mit seiner Neustartroutine bearbeitet. Im nächsten Aufrufzyklus wird automatisch auch COM_RST = FALSE zurückgesetzt.

Hinweis

Wird ein Regelkreis eingefügt oder gelöscht, d.h. wird der ganze DB neu generiert, ohne dass der Aufrufverteiler im Neustart bearbeitet werden soll, dann müssen die internen Regelkreiszähler (ILP_COU[n]) und der Parameter für die aktuelle Regelkreisnummer ALP_NBR mit Null vorbelegt werden.

Bedingungen für Regelkreisaufruf über LP_SCHED

Um zu gewährleisten, dass die Zeitabstände zwischen den Aufrufen eines bestimmten Reglers konstant bleiben und um die CPU gleichmäßig auszulasten, darf pro Zeittakt der Weckalarmebene nur ein Regelkreis bearbeitet werden. Bei der Parametrierung der Abtastzeiten MAN_CYC müssen deshalb in Relation zum Zeittakt (TM_BASE) folgende Bedingungen eingehalten werden:

- Die Bearbeitungszeiten der einzelnen Regelkreise müssen kleiner sein als der Zeittakt (TM_BASE) der Weckalarmebene.
- Die Abtastzeit eines Regelkreises (MAN_CYC) muss ein ganzzahliges Vielfaches (GV) des Produkts aus Zeitbasis und Anzahl der zu bearbeitenden Regler (GLP_NBR) sein: MAN_CYC = GV (TM_BASE * GLP_NBR)

Beispiel für Aufrufverteilung

Das folgende Beispiel zeigt die Aufrufreihenfolge von vier Regelkreisen in einer Weckalarmebene (siehe folgendes Bild). Pro Einheit der Zeitbasis wird immer nur ein Regelkreis bearbeitet. Die Aufrufreihenfolge und damit die Verschiebezeiten (VZ1 ... VZ5) ergeben sich aus der Reihenfolge der Aufrufdaten innerhalb des globalen DB.

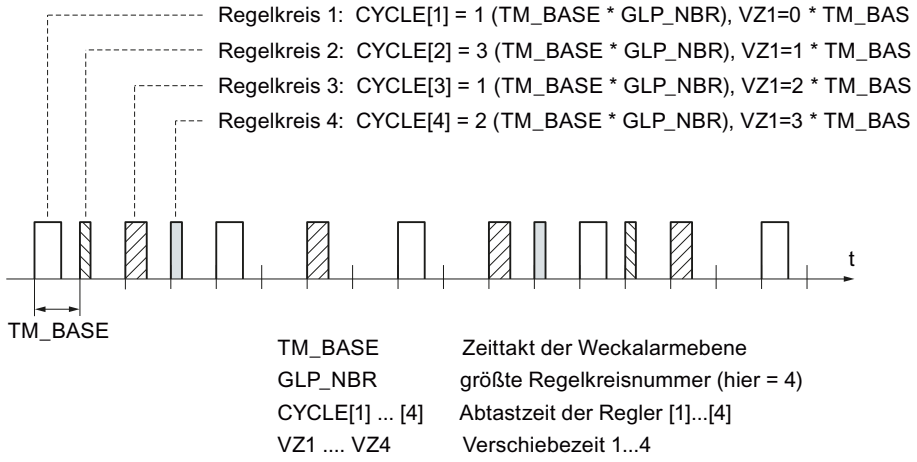

Bild 4-27 Aufrufreihenfolge von vier unterschiedlich häufig bearbeiteten Regelkreisen

Bausteininterne Begrenzungen

Die Werte der Eingangsparameter werden in der Anweisung nicht begrenzt; eine Prüfung der Parameter findet nicht statt.

Eingangsparameter LP_SCHED_M

Parameter	Datentyp	Beschreibung
TM_BASE	TIME	Zeitbasis >= 1ms
COM_RST	BOOL	Neustart
DB_NBR	BLOCK_DB	Datenbausteinnummer

Globaler Datenbaustein DB_LOOP

Die Daten für die Aufrufverteilung mit LP_SCHED_M werden in einem globalen Datenbaustein vorgegeben. Dieser globale Datenbaustein ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein nach folgendem Schema selbst erstellen und für Ihre Applikation anpassen.

Parameter	Datentyp	Vorbelegung	Beschreibung
GLP_NBR	INT	2	Maximale Anzahl Regler 1 - 256
ALP_NBR	INT	0	Nummer des aktuellen Reglers 0 - 256
LOOP_DAT	Array [1...GLP_NBR] of Struct		Struktur für die Daten der einzelnen Regler. Das Array muss von 1...GLP_NBR deklariert werden.
LOOP_DAT[1].MAN_CYC	TIME	T#1s	Daten Regler 1: manuelle Abtastzeit >= 1ms
LOOP_DAT[1].MAN_DIS	BOOL	FALSE	Daten Regler 1: manuell Regleraufruf sperren
LOOP_DAT[1].MAN_CRST	BOOL	FALSE	Daten Regler 1: manuell Neustart setzen
LOOP_DAT[1].ENABLE	BOOL	FALSE	Daten Regler 1: Reglerfreigabe
LOOP_DAT[1].COM_RST	BOOL	FALSE	Daten Regler 1: Neustart
LOOP_DAT[1].ILP_COU	INT	0	Daten Regler 1: interner Zähler
LOOP_DAT[1].CYCLE	TIME	T#1s	Daten Regler 1: Abtastzeit >= 1ms
LOOP_DAT[2].MAN_CYC			Daten Regler 2: manuelle Abtastzeit
...			

Alternativ können Sie den globalen Datenbaustein über eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Kopieren Sie den untenstehenden Text in die Zwischenablage.
2. Öffnen Sie einen externen Texteditor.
3. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
4. Speichern Sie die Datei mit der Dateinamenerweiterungen " .DB":
5. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Externe Quellen“.

6. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Das Dialogfeld "Öffnen" wird geöffnet.
7. Navigieren Sie zu der externen Quelldatei, die Sie zuvor erzeugt haben, und selektieren Sie diese.
8. Bestätigen Sie Ihre Auswahl mit "Öffnen".
9. Selektieren Sie die externe Quelldatei.
10. Wählen Sie im Kontextmenü den Befehl "Bausteine aus Quelle generieren".
11. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
12. Bestätigen Sie die Sicherheitsabfrage mit "OK".
13. Passen Sie den erzeugten Datenbaustein an Ihre Applikation an.

```
DATA_BLOCK "DB_LOOP"
TITLE = data block loop scheduler
{ S7_Optimized_Access := 'FALSE' }
NAME : DB_LOOP
 STRUCT
 GLP_NBR : Int; // greatest loop number
  ALP_NBR : Int; // actual loop number
 LOOP_DAT : Array[1..2] of Struct
 MAN_CYC : Time; // loop data: manual sample time
 MAN_DIS : Bool; // loop data: manual loop disable
 MAN_CRST : Bool; // loop data: manual complete restart
 ENABLE : Bool; // loop data: enable loop
 COM_RST : Bool; // loop data: complete restart
 ILP_COU : Int; // loop data: internal loop counter
 CYCLE : Time; // loop data: sample time
 END_STRUCT;
 END_STRUCT;


BEGIN
 GLP_NBR := 2;
 LOOP_DAT[1].MAN_CYC := T#1S;
 LOOP_DAT[1].CYCLE := T#1S;
 LOOP_DAT[2].MAN_CYC := T#200MS;
```

```
 LOOP_DAT[2].CYCLE := T#200MS;

END_DATA_BLOCK
```

4.2.4.2 Motion Control

Easy Motion Control

MC_Init

Beschreibung

Die Anweisung MC_Init bereitet die Initialisierung aller Bausteine einer Achse vor, indem sie alle Bits im Bitfeld "Init.Ix" des Achs-DB setzt (siehe "Initialisierung und Parameteränderungen").

Arbeitsweise

Die Anweisung MC_Init **verhindert ein unbeabsichtigtes Anfahren der Achse**, indem sie im Achs-DB die Fehleranzeigen "Error" = TRUE und "Err.StoppedMotion" = TRUE setzt. Wollen Sie nach dem Aufruf der Anweisung MC_Init die Achse verfahren, müssen Sie zunächst die Fehleranzeigen durch Quittieren beseitigen (siehe "Fehleranzeigen und Fehlerquittung im Achs-DB").

Die Anweisung MC_Init überprüft die Parameter im Achs-DB. Wird ein Parametrierfehler gefunden, werden im Achs-DB zur (Sammel-) Fehleranzeige die Bits "Error" und "Err.ConfigErr" sowie zur genaueren Fehleranzeige eines der Bits in der Struktur "ConfigErr" gesetzt.

Parametrierfehler können **nicht quittiert** werden. Sie müssen die Anweisung MC_Init nach Korrektur des Fehlers erneut aufrufen.

Weitere Informationen zum Fehlerkonzept siehe "Fehleranzeigen und Fehlerquittung im Achs-DB" und "Parametrierfehler".

Aufruf

Hinweis

Ein Aufruf der Anweisung MC_Init ist nur erlaubt, wenn sich die Achse im Stillstand befindet.

Rufen Sie die Anweisung MC_Init bedingt im gleichen Programmbaustein auf, in dem Sie auch die übrigen Anweisungen dieser Achse aufrufen. In folgenden Fällen müssen Sie die Anweisung MC_Init aufrufen:

- Nach jedem Anlauf der CPU (OB 100 und OB 101)
- Wenn Sie ohne die Projektiersoftware einen Parameter ändern, der in "Technologieobjekt Achse konfigurieren" durch folgendes Symbol gekennzeichnet ist:

Hinweis

Setzen Sie die für den Aufruf der Anweisung MC_Init genutzte Bedingung zurück, wenn der Aufruf einmal durchgeführt wurde. Ansonsten wiederholen die Easy Motion Control-Bausteine immer nur ihre Initialisierung, weil die Initialisierungs-Bits im Achs-DB wieder neu gesetzt werden.

MC_MoveAbsolute

Beschreibung

Mit der Anweisung MC_MoveAbsolute können Sie eine synchronisierte Achse auf das in "Position" vorgegebene absolute Ziel bewegen.

Arbeitsweise

Mit dem Erkennen einer positiven Flanke am Eingang "Execute" bestimmt die Anweisung MC_MoveAbsolute die Fahrtrichtung und beschleunigt die Achse in diese Richtung. Mit Erreichen des Bremseinsatzpunktes wird die Achse abgebremst. Die vorgegebene Geschwindigkeit muss bei der Fahrt nicht erreicht werden.

Besonderheit bei einer Linearachse:

- Die Fahrtrichtung wird bestimmt aus der Sollposition beim Start und dem Ziel.

Besonderheiten bei einer Rundachse:

- Mit der Anweisung MC_MoveAbsolute können Sie aus dem Achsstillstand nur einen Weg verfahren, der kleiner als eine Umdrehung der Rundachse ist. Für Positionierungen über mehrere Umdrehungen der Rundachse verwenden Sie die Anweisung MC_MoveRelative (Seite 6054).
- Richtungsanwahl bei Start aus dem Achsstillstand:
  - Positive Richtung: das Ziel wird in positiver Richtung angefahren,
  - Negative Richtung: das Ziel wird in negativer Richtung angefahren,
  - kürzester Weg: das Ziel wird in derjenigen Richtung angefahren, in der der Weg zwischen Startpunkt und Ziel ≤ ½ Umdrehung der Rundachse ist.
- Richtungsanwahl bei Übernahme einer Fahrt:
  - Positive Richtung: das Ziel wird in positiver Richtung angefahren. War die bisherige Fahrt in negativer Richtung, so wird die Achse abgebremst und das Ziel in positiver Richtung angefahren. Wird das Ziel beim Abbremsen mit negativer Fahrtrichtung genau erreicht, so fährt die Achse anschließend eine Umdrehung in positiver Richtung.
  - Negative Richtung: das Ziel wird in negativer Richtung angefahren. War die bisherige Fahrt in positiver Richtung, so wird die Achse abgebremst und das Ziel in negativer Richtung angefahren. Wird das Ziel beim Abbremsen mit positiver Fahrtrichtung genau erreicht, so fährt die Achse anschließend eine Umdrehung in negativer Richtung.
  - kürzester Weg: das Ziel wird in derjenigen Richtung angefahren, in der der Weg zwischen Startpunkt und Ziel ≤ ½ Umdrehung der Rundachse ist. Ist der Bremsweg größer, so fährt die Achse in der ursprünglichen Richtung auf das Ziel.
  - Haben Sie die Fahrparameter so gewählt, dass der Bremsweg der bisherigen Fahrt größer als eine Umdrehung der Rundachse ist, so wird nach dem Umkehren direkt zum Ziel gefahren.
  - Der gesamte Weg zwischen Startpunkt und Ziel kann größer als eine Umdrehung der Rundachse sein.

Signalablaufdiagramm für MC_MoveAbsolute

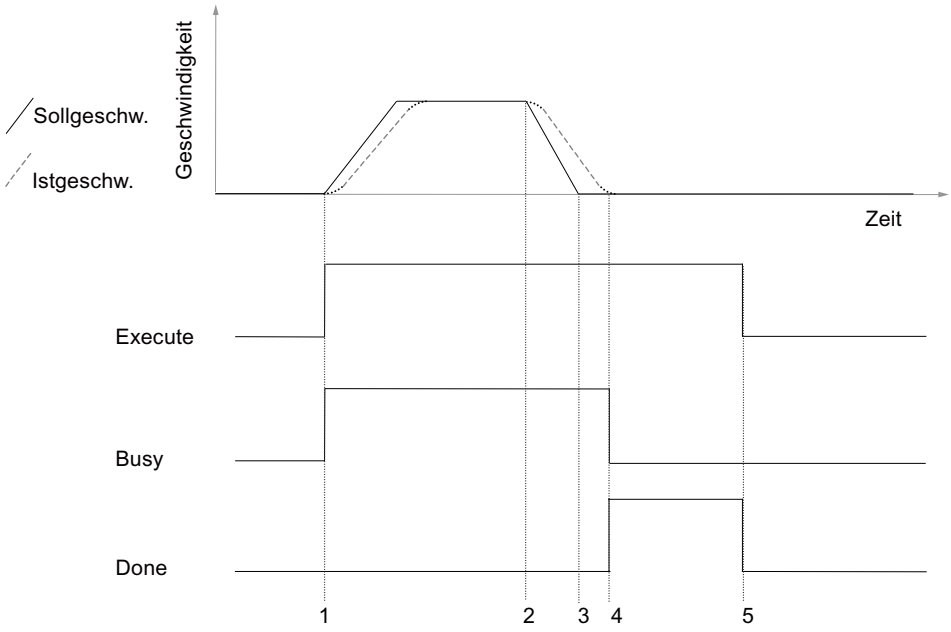

1 Start der Fahranweisung

2 Bremseinsatzpunkt

3 Sollwert hat Zielposition erreicht

4 Achse hat den Zielbereich erreicht

5 Mit Rücksetzen von „Execute“ wird „Done“ rückgesetzt. Der Vorgang ist beendet. Wenn „Execute“ vor dem Erreichen der Zielposition rückgesetzt wird, dann wird „Done“ beim Erreichen der Zielposition für einen Zyklus gesetzt und dann wieder rückgesetzt.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
Execute	INPUT	BOOL	Start der Positionierung bei einer positiven Flanke.
Position	INPUT	REAL	Absolute Zielposition in [Längeneinheit] Bei Rundachse: Anfang der Rundachse ≤ Position < Ende der Rundachse
Velocity	INPUT	REAL	Achsgeschwindigkeit in [Längeneinheit/s]
Acceleration	INPUT	REAL	Achsbeschleunigung in [Längeneinheit/s^2]
Deceleration	INPUT	REAL	Achsverzögerung in [Längeneinheit/s^2]
Direction	INPUT	INT	Richtungsanwahl für Fahrt mit Rundachse. -1 = Negative Richtung 0 = Kürzester Weg 1 = Positive Richtung

Parameter	Deklaration	Datentyp	Beschreibung
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
Done	OUTPUT	BOOL	1 = Auftrag fehlerfrei beendet. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
CommandAborted	OUTPUT	BOOL	1 = Auftrag wurde durch einen anderen Auftrag abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

UDT2

In den Bausteineigenschaften von MC_MoveAbsolute ist ein Datentyp UDT2 gelistet und reserviert. Aus diesem Grund darf im gleichen Projekt kein eigener Datentyp UDT2 genutzt werden. Dies führt zu Fehlermeldungen und Inkonsistenzen innerhalb des Projektes. Sollte ein Datentyp UDT2 genutzt werden, so ist dieser UDT umzubenennen. Zum Beispiel in UDT3.

MC_MoveRelative

Beschreibung

Mit der Anweisung MC_MoveRelative können Sie eine Achse um einen in "Distance" vorgegebenen Weg relativ zum Sollwert beim Start der Fahrt bewegen. Das Vorzeichen von "Distance" bestimmt die Richtung.

Arbeitsweise

Mit dem Erkennen einer positiven Flanke am Eingang "Execute" beschleunigt die Anweisung MC_MoveRelative die Achse in die in "Distance" angegebene Richtung. Mit Erreichen des Bremseinsatzpunktes wird die Achse abgebremst. Die vorgegebene Geschwindigkeit muss bei der Fahrt nicht erreicht werden. Das Ziel wird bestimmt aus der Sollposition beim Start der Fahrt und dem zu fahrenden Weg.

Besonderheit bei Rundachsen:

- Mit der Vorgabe von Werten größer als eine Umdrehung der Rundachse in "Distance" lassen sich Fahrten über mehrere Umdrehungen der Rundachse realisieren.

Signalablaufdiagramm für MC_MoveRelative

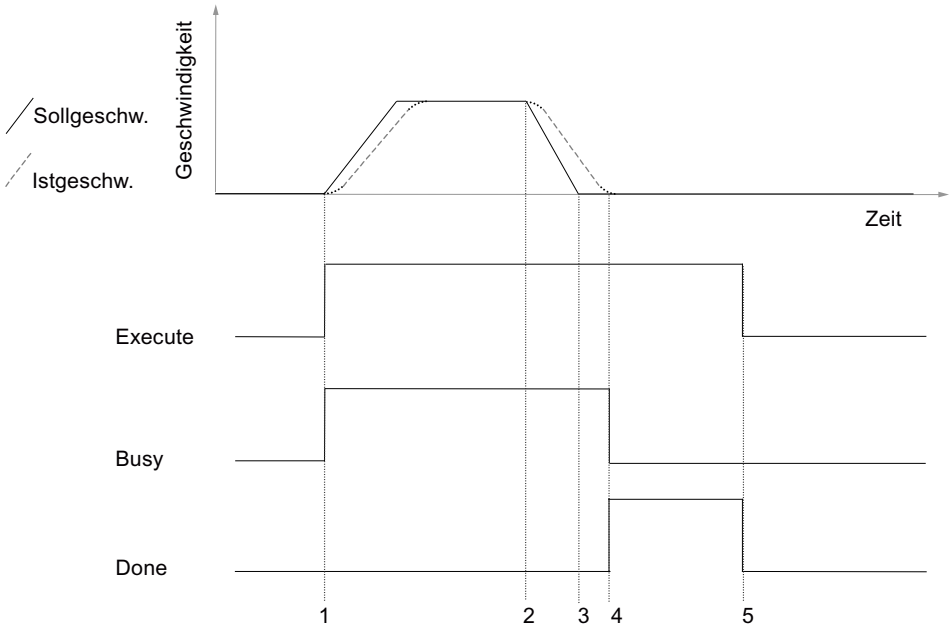

1 Start der Fahranweisung

2 Bremseinsatzpunkt

3 Sollwert hat Zielposition erreicht

4 Achse hat den Zielbereich erreicht

5 Mit Rücksetzen von „Execute“ wird „Done“ rückgesetzt. Der Vorgang ist beendet. Wenn „Execute“ vor dem Erreichen der Zielposition rückgesetzt wird, dann wird „Done“ beim Erreichen der Zielposition für einen Zyklus gesetzt und dann wieder rückgesetzt.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
Execute	INPUT	BOOL	Start der Positionierung bei einer positiven Flanke.
Distance	INPUT	REAL	Zu fahrender Weg in [Längeneinheit], bezogen auf den letzten Lagesollwert. Das Vorzeichen bestimmt die Richtung der Fahrt.
Velocity	INPUT	REAL	Achsgeschwindigkeit in [Längeneinheit/s]
Acceleration	INPUT	REAL	Achsbeschleunigung in [Längeneinheit/s^2]
Deceleration	INPUT	REAL	Achsverzögerung in [Längeneinheit/s^2]
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
Done	OUTPUT	BOOL	1 = Auftrag fehlerfrei beendet. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

Parameter	Deklaration	Datentyp	Beschreibung
CommandAborted	OUTPUT	BOOL	1 = Auftrag wurde durch einen anderen Auftrag abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

UDT2

In den Bausteineigenschaften von MC_MoveRelative ist ein Datentyp UDT2 gelistet und reserviert. Aus diesem Grund darf im gleichen Projekt kein eigener Datentyp UDT2 genutzt werden. Dies führt zu Fehlermeldungen und Inkonsistenzen innerhalb des Projektes. Sollte ein Datentyp UDT2 genutzt werden, so ist dieser UDT umzubenennen. Zum Beispiel in UDT3.

MC_MoveJog

Beschreibung

Mit der Anweisung MC_MoveJog können Sie eine Achse über zwei pegelgesteuerte Richtungseingänge "JogPos" und "JogNeg" verfahren (Tippen).

Arbeitsweise

Wenn Sie einen Richtungseingang setzen, dann beschleunigt MC_MoveJog die Achse zunächst auf die angegebene Geschwindigkeit. Solange der Richtungseingang ansteht, verfährt die Achse in diese Richtung. Wenn Sie den Richtungseingang wieder zurücksetzen, bremst die Achse.

Die beiden Richtungseingänge werden mit folgenden Abhängigkeiten ausgewertet:

- Solange eine Richtung angewählt ist, ist das Setzen der Gegenrichtung wirkungslos. Es wird kein Fehler erzeugt.
- Um während einer Fahrt die Fahrtrichtung zu wechseln, setzen Sie den anderen Richtungseingang auf TRUE und gleichzeitig den ursprünglich gewählten Richtungseingang auf FALSE.
- Sind zu Beginn einer Fahrt beide Eingänge gesetzt, wird ein Startfehler erzeugt ("Err.StartErr", siehe Fehler mit weichem Stopp).

Besonderheiten bei einer Linearachse:

- Wird der maximale Fahrweg während einer Fahrt mit einer Linearachse überschritten, dann bremst die Anweisung MC_MoveJog die Achse mit der quittierbaren Fehlermeldung "Err.DistanceErr" bis zum Stillstand ab (siehe AUTOHOTSPOT). Nach der Fehlerquittierung kann die Achse erneut verfahren werden.
- Eine **nicht synchronisierte** Linearachse wird nicht überwacht und kann deshalb ohne abzubremsen bis zum physikalischen Ende der Achse fahren. Beenden Sie daher die Fahrt rechtzeitig.

> ⚠ **WARNUNG**
>
> **Nicht synchronisierte Linearachse**
>
> Es kann zu Personen- und Sachschäden kommen.
>
> Die Anweisung MC_MoveJog kann bei einer nicht synchronisierten Linearachse bis zum physikalischen Ende der Achse fahren.
>
> Zur Vermeidung von Schäden an Personen und Gegenständen treffen Sie folgende Maßnahmen:
>
> - Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners. Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Software-Ausfalls die Anlage sicher ausgeschaltet werden kann.
> - Installieren Sie Sicherheits-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
> - Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.

- Eine **synchronisierte** Linearachse wird auf das Ende des Arbeitsbereichs abgebremst und kommt auf dem Software-Endschalter zum Stehen. Es wird die quittierbare Fehlermeldung "Err.DistanceErr" sowie die Fehlermeldung "Err.SWLimitMinExceeded" bzw. "Err.SWLimitMaxExceeded" ausgegeben. Nach der Fehlerquittierung ist nur eine Fahrt in die Gegenrichtung erlaubt.

Besonderheit bei einer Rundachse:

- Der Fahrweg der Anweisung MC_MoveJog ist bei einer Rundachse nicht beschränkt (Endlosdrehen).

Signalablaufdiagramm für MC_MoveJog

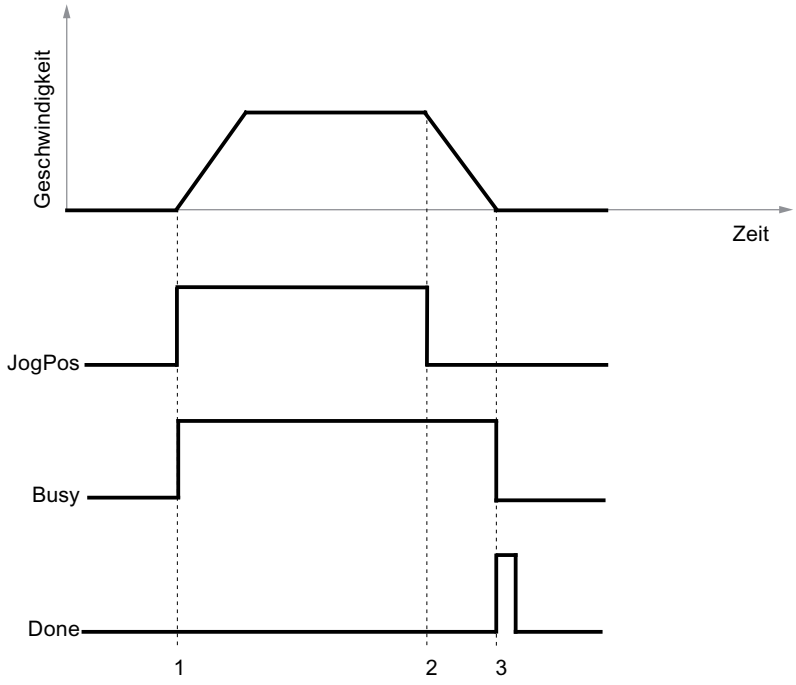

1 Bewegung wird durch Setzen von „JogPos“ gestartet.
2 Bewegung wird durch Rücksetzen von „JogPos“ gestoppt.
3 Mit Stillstand der Achse wird „Busy“ rückgesetzt und „Done“ für einen Zyklus gesetzt.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
JogPos	INPUT	BOOL	1 = Bewegung in positiver Richtung, solange "JogPos" gesetzt ist.
JogNeg	INPUT	BOOL	1 = Bewegung in negativer Richtung, solange "JogNeg" gesetzt ist.
Velocity	INPUT	REAL	Achsgeschwindigkeit in [Längeneinheit/s]
Acceleration	INPUT	REAL	Achsbeschleunigung in [Längeneinheit/s^2]
Deceleration	INPUT	REAL	Achsverzögerung in [Längeneinheit/s^2]
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
Done	OUTPUT	BOOL	1 = Auftrag fehlerfrei beendet. Das Bit wird nach Erreichen des Achsstillstands für genau einen Aufruf der Anweisung gesetzt.

Parameter	Deklaration	Datentyp	Beschreibung
CommandAborted	OUTPUT	BOOL	1 = Auftrag wurde durch einen anderen Auftrag abgebrochen. Das Bit bleibt solange anstehen, bis "JogPos" bzw. "JogNeg" zurückgesetzt wird.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "JogPos" bzw. "JogNeg" zurückgesetzt wird.

UDT2

In den Bausteineigenschaften von MC_MoveJog ist ein Datentyp UDT2 gelistet und reserviert. Aus diesem Grund darf im gleichen Projekt kein eigener Datentyp UDT2 genutzt werden. Dies führt zu Fehlermeldungen und Inkonsistenzen innerhalb des Projektes. Sollte ein Datentyp UDT2 genutzt werden, so ist dieser UDT umzubenennen. Zum Beispiel in UDT3.

MC_Home

Beschreibung

Die Anweisung MC_Home wird verwendet:

- zur Referenzpunktfahrt bei Inkrementalgeber ("Velocity" > 0)
- zum Bezugspunktsetzen bei Inkremental- und Absolutwertgeber ("Velocity" = 0)
  Bei einem Absolutwertgeber wird das Bezugspunktsetzen auch Absolutwertgeberjustage genannt.

Arbeitsweise

Referenzpunktfahrt ("Velocity" > 0)

Nach dem Setzen von "Execute" fährt die Achse mit der gewählten Geschwindigkeit und Richtung so lange, bis das Erreichen des Referenzpunkts von der Wegerfassungsbaugruppe und dem Eingangstreiber erkannt wird. Mit Erreichen des Referenzpunkts wird der Lageistwert auf den Wert von "Position" gesetzt und die Achse abgebremst. Die Achse ist nun synchronisiert ("Sync" = TRUE), steht aber nicht auf dem Referenzpunkt.

Wird der maximale Fahrweg bei der Referenzpunktfahrt überschritten, bremst die Anweisung MC_Home die Achse bis zum Stillstand ab (siehe Ablauf von Fahrbewegungen) und die Fehleranzeigen "Error" und "Err.DistanceErr" werden gesetzt (siehe Fehler mit weichem Stopp).

Wird die Referenzpunktfahrt vor Erreichen des Referenzpunkts gestoppt, ist die Achse unsynchronisiert ("Sync" = FALSE).

Die Anweisung MC_Home stoppt nicht am Ende der Achse und führt auch keinen automatischen Richtungswechsel aus.

⚠ GEFAHR
Für den Fall, dass kein Referenzpunkt gefunden wird, müssen Sie dafür sorgen, dass die Achse vor dem physikalischen Ende durch geeignete Maßnahmen zum Stehen kommt (z. B. NOT-AUS-Schalter, Sicherheits-Endschalter).

Bezugspunktsetzen ("Velocity" = 0)

Setzen Sie die Geschwindigkeit "Velocity" auf 0. Nach Setzen von "Execute" wird der Lageistwert auf den Wert von "Position" gesetzt, gleichzeitig wird im Achs-DB das Zustandsbit "Sync" = TRUE.

Nach einem Tausch des Absolutwertgebers oder mechanischen Änderungen müssen Sie das Bezugspunktsetzen erneut durchführen, auch wenn die Achse noch synchronisiert ist.

Hinweis

Bei einem Absolutwertgeber gilt:

Der Geberwert am Bezugspunkt wird im Achs-DB gespeichert. Daher sollten Sie nach dem Setzen des Bezugspunkts den Online-Achs-DB in Ihre Offline-Datenhaltung übernehmen. Andernfalls ist nach dem Laden des DB in die CPU die Achse nicht mehr synchronisiert.

Bitte beachten Sie die Hinweise zum Laden des Achs-DB.

Signalablaufdiagramm für Referenzpunktfahrt

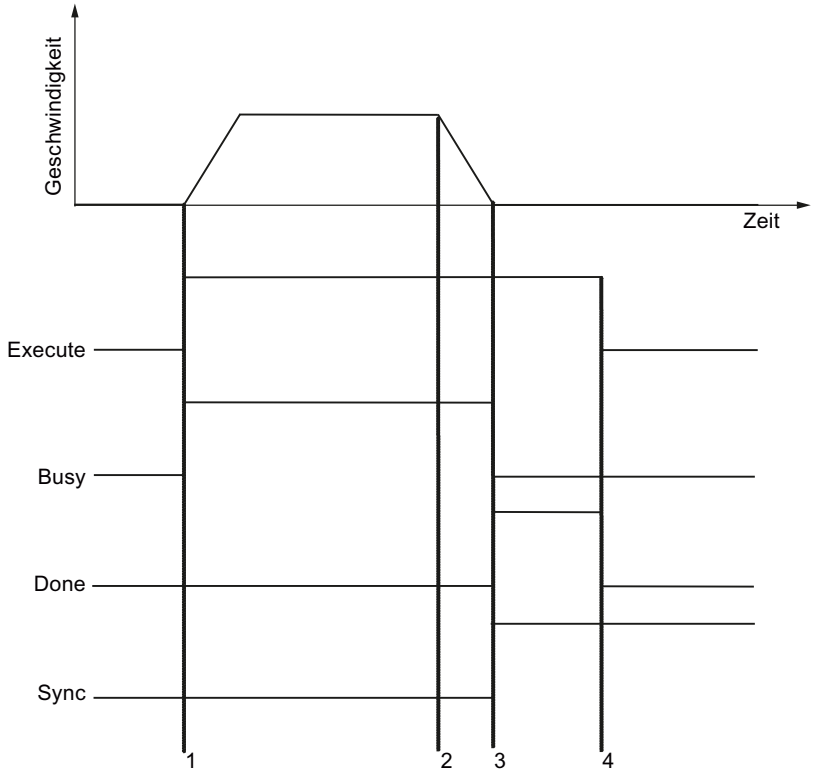

1 Start der Referenzpunktfahrt mit „Execute“.
2 Referenzpunkt gefunden, die Achse bremst.
3 Achse ist zum Stillstand gekommen, „Busy“ wird gelöscht und „Done“ wird gesetzt.
Die Achse ist synchronisiert.
4 Mit Rücksetzen von „Execute“ wird „Done“ rückgesetzt.

Signalablaufdiagramm für Bezugspunktsetzen

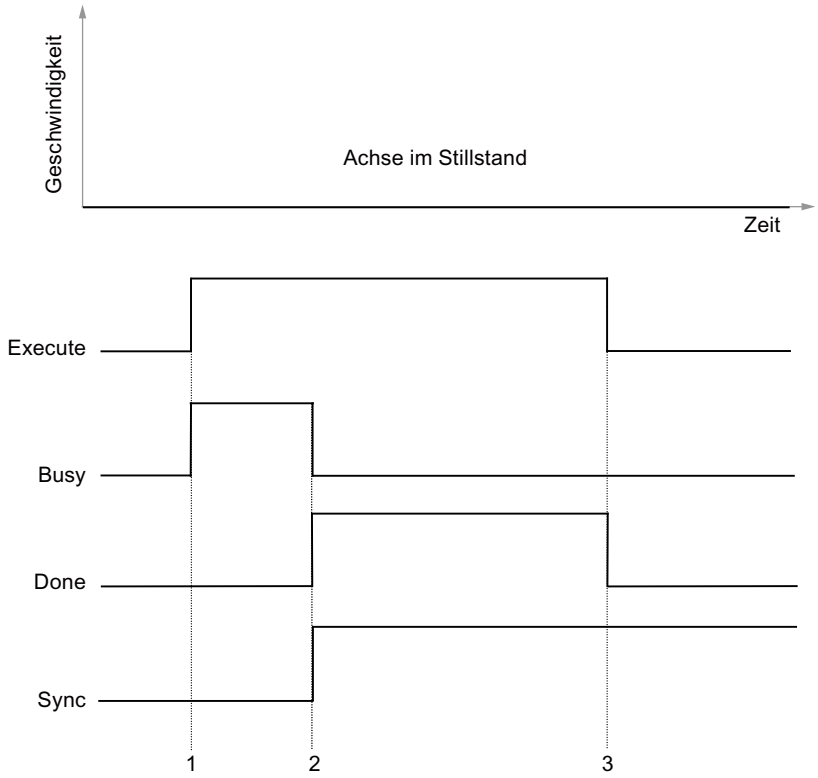

1 Start des Bezugspunktsetzens oder der Absolutwertgeberjustage mit „Execute“.
2 Die Achse meldet synchronisiert.
3 Mit Rücksetzen von „Execute“ wird „Done“ rückgesetzt. Vorgang wird beendet.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Der Start der Referenzpunktfahrt oder des Bezugspunktsetzens ist nur im Stillstand der Achse erlaubt.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
Execute	INPUT	BOOL	Start der Referenzpunktfahrt bzw. des Bezugspunktsetzens bei einer positiven Flanke.
Position	INPUT	REAL	**Inkrementalgeber:** Koordinate des Referenzpunkts oder des Bezugspunkts. **Absolutwertgeber:** Koordinate des Bezugspunkts. Der Wert muss innerhalb des Arbeitsbereiches liegen: • **Linearachse:** Innerhalb der Software-Endschalter • **Rundachse:** Anfang der Achse ≤ Position < Ende der Achse
Velocity	INPUT	REAL	Achsgeschwindigkeit in [Längeneinheit/s] Velocity > 0: Referenzpunktfahrt Velocity = 0: Bezugspunktsetzen
Acceleration	INPUT	REAL	Achsbeschleunigung in [Längeneinheit/s^2]
Deceleration	INPUT	REAL	Achsverzögerung in [Längeneinheit/s^2]
Direction	INPUT	INT	Startrichtung für Referenzpunktfahrt. 1 = Referenzpunktfahrt in positiver Richtung -1 = Referenzpunktfahrt in negativer Richtung
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
Done	OUTPUT	BOOL	1 = Auftrag fehlerfrei beendet. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
CommandAborted	OUTPUT	BOOL	1 = Auftrag wurde durch einen anderen Auftrag abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

MC_StopMotion

Beschreibung

Mit der Anweisung MC_StopMotion können Sie jede Fahranweisung abbrechen, so dass die Achse mit der eingestellten Verzögerung in den Stillstand fährt.

Typischer Anwendungsfall: Abbrechen einer Positionierung zu einem beliebigen Zeitpunkt aufgrund eines äußeren Ereignisses (z. B. Öffnen einer Schutztür).

Arbeitsweise

Mit Erkennen der positiven Flanke am Eingang "Execute" übernimmt die Anweisung MC_StopMotion die Kontrolle über die Achse und bremst sie bis zum Stillstand. Die Anweisung MC_StopMotion ist nur durch harte Fehler und nicht durch eine Fahranweisung unterbrechbar. Nach Erreichen des Stillstands kann eine erneute Fahrbewegung gestartet werden.

Die Verzögerung an der Anweisung MC_StopMotion darf beliebig gewählt werden. Ist sie kleiner als diejenige der unterbrochenen Anweisung, dann wird die Achse mit der Verzögerung der unterbrochenen Anweisung gebremst. Damit ist gewährleistet, dass die Achse nicht über das Ziel der unterbrochenen Anweisung hinausfährt.

Wenn Sie eine Fahrt mit Ziel (MC_MoveAbsolute, MC_MoveRelative) durch die Anweisung MC_StopMotion unterbrechen, dann berechnet diese den Restweg weiter. Nach Beendigung der Anweisung MC_StopMotion zeigt der Restweg die Entfernung der momentanen Sollposition vom Ziel der unterbrochenen Anweisung an. Mit diesem Wert können Sie eine Anweisung MC_MoveRelative versorgen, um nach der Unterbrechung die Achse auf das ursprüngliche Ziel zu fahren.

Signalablaufdiagramm für MC_StopMotion

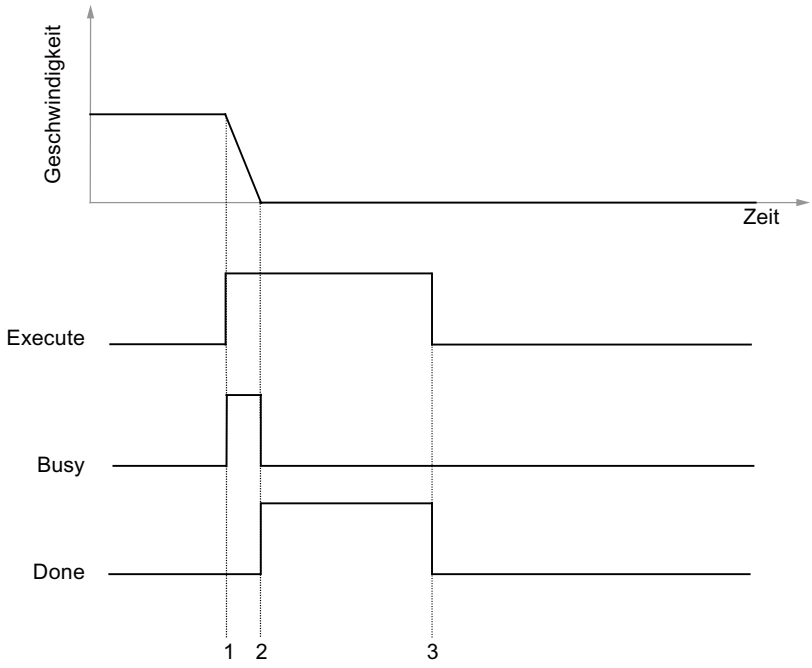

1 Während einer laufenden Bewegung wird die Stop-Anweisung gestartet.

2 Mit Erreichen des Stillstands setzt MC_StopMotion den Ausgang „Done“.

3 Mit Rücksetzen von „Execute“ ist die Stop-Anweisung beendet.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
Execute	INPUT	BOOL	Start des Bremsvorgangs bei einer positiven Flanke.
Deceleration	INPUT	REAL	Achsverzögerung für Bremsrampe in [Längeneinheit/s^2]. Bei Eingabe von Werten ≤ 0.0 wird der Wert der maximalen Verzögerung aus dem Achs-DB verwendet.
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
Done	OUTPUT	BOOL	1 = Auftrag fehlerfrei beendet. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.
CommandAborted	OUTPUT	BOOL	1 = Auftrag durch Handbetrieb am Regler abgebrochen.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

MC_Control

Beschreibung

Mit der Anweisung MC_Control können Sie zusammen mit den Eingangs- und Ausgangstreibern die Lageregelung realisieren.

Die Anweisung MC_Control kann folgende Reglerzustände annehmen:

- Regelbetrieb
- Nachführbetrieb
- Handbetrieb

Regelbetrieb

Die Anweisung MC_Control

- berechnet den Geschwindigkeitssollwert für den Ausgangstreiber aus dem Lagesollwert einer Fahr-Anweisung und dem Lageistwert des Eingangstreibers.
- überwacht
  - Stillstand,
  - Zieleinlauf,
  - Schleppabstand.
- bearbeitet die Fehlerquittung (siehe Fehleranzeigen und Fehlerquittung).

Nachführbetrieb

In diesem Zustand wird der Lagesollwert dem Lageistwert nachgeführt. Der Nachführbetrieb wird aktiv, wenn der Eingang "EnableDrive" = FALSE ist oder nach einem Achsfehler mit hartem Stopp (siehe Fehleranzeigen und Fehlerquittung). Im Nachführbetrieb ist der Ausgang "DriveEnabled" = FALSE.

Handbetrieb

In diesem Zustand wird als Geschwindigkeitssollwert der Handsollwert "ManVelocity" ausgegeben. Dieser wird auf "MaxVelocity" begrenzt. Der Lagesollwert wird dem Lageistwert nachgeführt.

Dieser Zustand wird durch "ManEnable" ein- bzw. ausgeschaltet.

⚠ WARNUNG
Der Handbetrieb ist jederzeit möglich; auch dann, wenn sich die Achse im Fehlerzustand befindet.

Arbeitsweise

Der Regelalgorithmus der Anweisung MC_Control ist ein P-Regler, der über den Parameter "Reglerverstärkung" parametriert wird ("FactorP").

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Rufen Sie die Anweisung MC_Control einmal pro Achse ohne Bedingung im gewählten OB auf.

Parameter

Parameter	Deklaration	Datentyp	Bedeutung
EnableDrive	INPUT	BOOL	0 = Antrieb sperren, Nachführbetrieb ist aktiv. 1 = Antrieb freigeben, Lageregelung ist aktiv. Wenn kein Fehler ansteht, wird daraufhin "DriveEnabled" gesetzt.
DriveEnabled	OUTPUT	BOOL	Antriebsfreigabe, Lageregelung ist aktiv. Wenn "EnableDrive" gesetzt ist und kein Fehler ansteht, wird "DriveEnabled" gesetzt. Wenn "DriveEnabled" nicht gesetzt ist, befindet sich die Achse im Nachführbetrieb und es findet keine Überwachung der Achse auf Stillstand statt. Dieses Signal sollte verwendet werden, um den Freigabeeingang am Leistungsteil anzusteuern.

MC_Simulation

Beschreibung

Mit der Anweisung MC_Simulation können Sie Ihr Verfahrprogramm inklusive Treiberbausteine testen. Sie benötigen keine Peripherie und Ihre Achse wird nicht verfahren.

Arbeitsweise

Die Anweisung MC_Simulation koppelt den Regler-Ausgangswert auf den Eingangstreiber zurück, indem sie einen Geberwert entsprechend dem im Achs-DB gewählten Gebertyp simuliert.

Den Simulationsmodus schalten Sie ein, indem Sie im Achs-DB das Bit "Sim" = TRUE setzen.

Bei eingeschalteter Simulation ist eine Referenzpunktfahrt nicht möglich. Parametrieren Sie während der Simulation den Eingang "Velocity" der Anweisung MC_Home mit 0.0. Damit wird statt der Referenzpunktfahrt ein Bezugspunktsetzen durchgeführt.

Wenn Sie während des Simulationsbetriebs ein Bezugspunktsetzen durchführen, dann wird auch der Bezugspunkt Ihrer realen Achse verändert. Daher ist nach dem Ausschalten des Simulationsbetriebs eine erneute Synchronisation Ihrer Achse notwendig.

Beim Ein- und Ausschalten der Simulation wird die Achse in den quittierpflichtigen Stopp-Zustand gebracht (siehe Fehler mit hartem Stopp) und außerdem werden alle Bits im Bitfeld "Init.Ix" im Achs-DB gesetzt.

Ist der Simulationsmodus eingeschaltet, dann geschieht Folgendes:

- Der Ausgangstreiber gibt keinen Wert an die Peripherie aus.
- Der Eingangstreiber liest keinen Geberwert von der Peripherie ein.

⚠ GEFAHR
Beim Einschalten des Simulationsmodus muss sich die Achse im Stillstand befinden.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

MC_GearIn

Beschreibung

Mit der Anweisung MC_GearIn können Sie den Geschwindigkeitssollwert einer Folgeachse über einen Getriebefaktor an den Geschwindigkeitssollwert einer Leitachse ankoppeln (Sollwertkopplung).

Arbeitsweise

Mit dem Erkennen einer positiven Flanke am Eingang "Execute" koppelt die Anweisung MC_GearIn eine Folgeachse (Parameter "Slave") über den eingestellten Getriebefaktor an den Geschwindigkeits- und Positionssollwert einer Leitachse (Parameter "Master"). Die Anweisung ist damit eine Fahranweisung der Folgeachse.

Während einer gekoppelten Fahrt ist in jedem Zyklus:

- zurückgelegter Weg der Folgeachse ab Einkoppelposition =
  zurückgelegter Weg der Leitachse ab Einkoppelposition × Getriebefaktor
- Geschwindigkeit der Folgeachse = Getriebegeschwindigkeit =
  Geschwindigkeit der Leitachse × Getriebefaktor
- Beschleunigung der Folgeachse =
  Beschleunigung der Leitachse × Getriebefaktor
- Verzögerung der Folgeachse =
  Verzögerung der Leitachse × Getriebefaktor

Damit die Folgeachse der Leitachse uneingeschränkt folgen kann, stellen Sie die folgenden Werte mit einem Reservefaktor 1,1 für die Folgeachse ein. Diese Werte müssen zulässig und erreichbar sein:

- Velocity(Folgeachse) =
  Velocity(Leitachse) × ABS(Getriebefaktor) × 1,1
- Acceleration(Folgeachse) =
  Acceleration(Leitachse) × ABS(Getriebefaktor) × 1,1
- Deceleration(Folgeachse) =
  Deceleration(Leitachse) × ABS(Getriebefaktor) × 1,1

Um die Kopplung aufzuheben, aktivieren Sie die Stopp-Anweisung oder eine beliebige Fahranweisung für die Folgeachse.

Wie die anderen Fahranweisungen kann die Anweisung MC_GearIn eine Fahrt ablösen und selbst abgelöst werden.

Die Folgeachse kann in eine bereits laufende Leitachse eingekoppelt werden, wobei die Folgeachse solange selbsttätig mit den an der Getriebeanweisung parametrierten Rampen beschleunigt bzw. bremst, bis sie die Getriebegeschwindigkeit erreicht hat.

Wenn die Getriebegeschwindigkeit nach dem Start das erste Mal erreicht wurde, wird der Ausgang "InGear" gesetzt. Ist "Execute" zu diesem Zeitpunkt nicht gesetzt, steht "InGear" für einen Aufruf der Anweisung an. "InGear" wird zurückgesetzt mit fallendem Execute, bei Fehler und beim Ablösen der Fahrt.

Solange die Getriebegeschwindigkeit erreichbar ist, fährt die Folgeachse sowohl im Geschwindigkeits- als auch im Positionsgleichlauf. Der Ausgang "Coupled" der Anweisung ist gesetzt.

Kann die Folgeachse die Getriebegeschwindigkeit nicht erreichen, fährt sie mit den an der Getriebeanweisung angegebenen Fahrparametern. Der Ausgang "Coupled" der Anweisung ist dann rückgesetzt. Sobald die Getriebegeschwindigkeit wieder erreichbar ist, wird die Folgeachse mit dieser verfahren. Der Ausgang "Coupled" der Anweisung wird gesetzt.

Während die Anweisung MC_GearIn für die Folgeachse aktiv ist, können Sie für die Leitachse beliebige Fahranweisungen aktivieren, auch mit Ablösen einer Fahrt.

Ausnahme: Beim Referenzieren der Leitachse wird die Kopplung mit Fehler ("Err.MasterErr") aufgelöst.

Der Achsparameter "Override" ist für die Folgeachse nicht sinnvoll und wird daher nicht ausgewertet.

Besonderheiten bei einer Folgeachse als Linearachse:

- Wird der maximale Fahrweg während einer Getriebefahrt überschritten, dann bremst die Anweisung MC_GearIn die Folgeachse mit der Fehlermeldung "Err.DistanceErr" bis zum Stillstand ab (siehe Ablauf von Fahrbewegungen). Die Kopplung wird aufgelöst.
- Eine **nicht synchronisierte** Linearachse wird nicht überwacht und kann deshalb ohne abzubremsen bis zum physikalischen Ende der Achse fahren. Beenden Sie daher die Fahrt rechtzeitig.
- Eine **synchronisierte** Linearachse wird auf das Ende des Arbeitsbereichs abgebremst und kommt auf dem Software-Endschalter zum Stehen. Es wird die Fehlermeldung "DistanceErr" zusammen mit "Err.SWLimitMinExceeded" bzw. "Err.SWLimitMaxExceeded" ausgegeben. Die Kopplung wird aufgelöst.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Die Anweisung MC_GearIn kann bei einer nicht synchronisierten Linearachse bis zum physikalischen Ende der Achse fahren.

Zur Vermeidung von Schäden an Personen und Gegenständen treffen Sie folgende Maßnahmen:

- Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners. Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Softwareausfalls die Anlage sicher ausgeschaltet werden kann.
- Installieren Sie Sicherheits-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
- Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.

Besonderheit bei einer Folgeachse als Rundachse:

- Der Fahrweg der Anweisung MC_GearIn ist nicht beschränkt (Endlosdrehen).

Fehlerbehandlung

Fehler in der Leitachse

- Wenn die Leitachse Fehler erkennt, die zu einem weichen Stopp führen, bremst sie ab. Die Folgeachse macht diese Bewegung gekoppelt mit und löst die Kopplung nicht auf. Wird der Fehler der Leitachse quittiert und fährt diese weiter, so läuft auch die Folgeachse wieder an.
- Wenn die Leitachse Fehler erkennt, die zu einem harten Stopp führen, oder wenn sich die Leitachse im Handbetrieb befindet, bremst die Getriebeanweisung die Folgeachse ab und löst die Kopplung auf. Sie gibt die Fehlermeldung "Err.MasterErr" aus.

Fehler in der Folgeachse

- Wenn die Folgeachse Fehler erkennt, die zu einem weichen Stopp führen, bremst die Getriebeanweisung die Folgeachse ab und löst die Kopplung auf.
- Wenn die Folgeachse Fehler erkennt, die zu einem harten Stopp führen, wird die Folgeachse angehalten und die Kopplung aufgelöst.
- Der maximale Verfahrweg einer Kopplungsfahrt bei einer Linearachse wird auf 2^{24} Inkremente bzw. 2^{24} [Längeneinheit] begrenzt. Wird diese Grenze überschritten, bremst die Anweisung MC_GearIn die Folgeachse ab, gibt die quittierbare Fehlermeldung "DistanceErr" aus und löst die Kopplung auf.

Fehler in der Kopplung

- Kann die Folgeachse die Getriebegeschwindigkeit im aktuellen Zyklus nicht erreichen, fährt sie mit den an der Anweisung angegebenen Fahrparametern. Dies gilt insbesondere beim Einkoppeln in eine laufende Leitachse sowie beim Bremsen im Fehlerfall.
  Der Parameter "Coupled" wird zurückgesetzt, die Kopplung jedoch nicht aufgelöst. Sobald die geforderten Werte wieder erreichbar sind, wird die Folgeachse mit diesen verfahren und "Coupled" gesetzt.

Zusammenwirken der Anweisungen beim Getriebe

Sie sollten beim Einsatz der Getriebeanweisung noch Folgendes beachten (siehe auch "Aufbau eines Anwenderpogramms"):

- Leitachse und Folgeachse müssen in **derselben** Ablaufebene aufgerufen werden.
- Die Folgeachse orientiert sich in jedem Bearbeitungszyklus neu an den Vorgaben der Leitachse. Die Anweisungen der Folgeachse sollten deshalb nach den Anweisungen der Leitachse aufgerufen werden. Insbesondere sollte die Getriebeanweisung der Folgeachse **nach** dem Regler der Leitachse aufgerufen werden.
  Andere Aufrufreihenfolgen, insbesondere der verschachtelte Aufruf der Anweisungen von Leit- und Folgeachse, können große Schleppabstände zwischen den Achsen erzeugen.
- Bei einer taktsynchronen Anwendung müssen die Peripheriedaten der Leit- **und** der Folgeachse im selben Teilprozessabbild liegen.

Je nach gewählter Struktur ergibt sich also eines der folgenden Bilder:

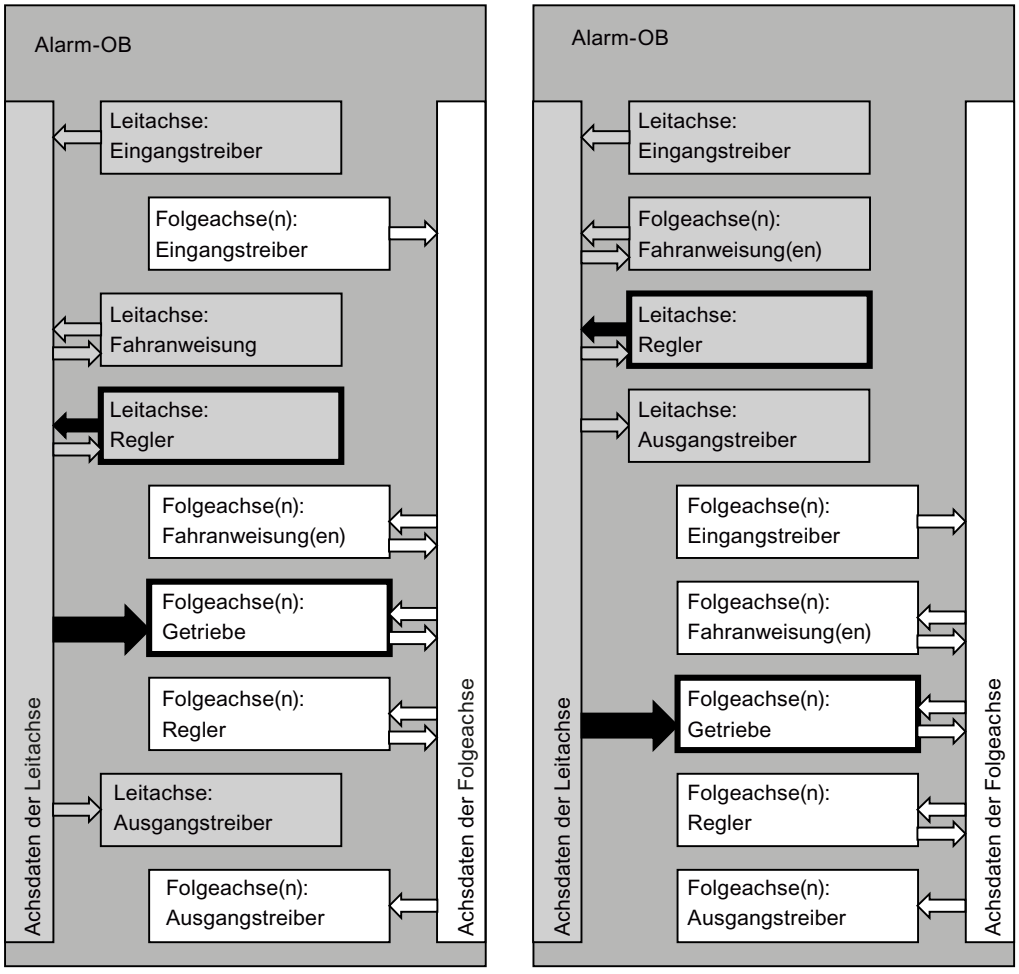

Signalablaufdiagramm

Das folgende Bild zeigt, wie die Getriebeanweisung die Fahrt einer Achse übernimmt und sich in eine laufende Leitachse einkoppelt. In diesem Beispiel ist der Getriebefaktor 0,8. Die zur Einhaltung dieses Faktors notwendige Geschwindigkeit kann von der Folgeachse nicht erreicht werden.

Die Leitachse ist zwischen t = 0,5 und t = 0,72 im Stillstand. Die Kopplung bleibt erhalten.

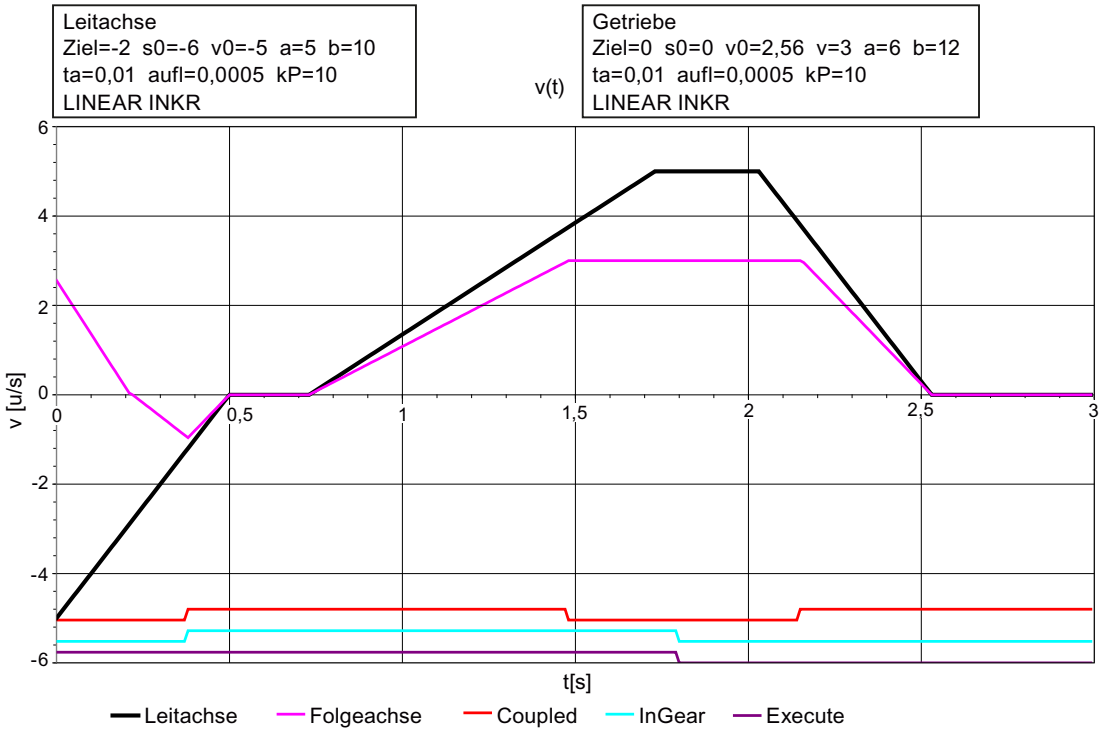

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
Execute	INPUT	BOOL	Start der Getriebefahrt bei einer positiven Flanke
Rationumerator	INPUT	REAL	Zähler des Getriebefaktors Getriebefaktor = "RatioNumerator" / "RatioDenominator" Ist "RatioNumerator" < 0, dann ist die Bewegungsrichtung von Leitachse und Folgeachse entgegengesetzt. "RatioNumerator" = 0 wird mit "DataErr" = 1 abgelehnt. Sie können den für eine Folgeachse wirksamen Getriebefaktor ändern, wenn Sie die aktive Getriebeanweisung durch eine andere Instanz mit einem geänderten Getriebefaktor ablösen.
RatioDenominator	INPUT	INT	Nenner des Getriebefaktors; muss ≥ 1 und ganzzahlig sein Sie können den für eine Folgeachse wirksamen Getriebefaktor ändern, wenn Sie die aktive Getriebeanweisung durch eine andere Instanz mit einem geänderten Getriebefaktor ablösen.
Velocity	INPUT	REAL	Achsgeschwindigkeit in [Längeneinheit/s] Die bei einer Getriebefahrt von der Folgeachse erreichbare Geschwindigkeit wird auf diesen Wert begrenzt.

Parameter	Deklaration	Datentyp	Beschreibung
Acceleration	INPUT	REAL	Achsbeschleunigung in [Längeneinheit/s²] Die bei einer Getriebefahrt von der Folgeachse erreichbare Beschleunigung wird auf diesen Wert begrenzt. Dieser Parameter wird verwendet, wenn • die Folgeachse sich in eine schneller laufende Leitachse einkoppelt, • die Leitachse stärker beschleunigt, als es die Folgeachse zulässt.
Deceleration	INPUT	REAL	Achsverzögerung in [Längeneinheit/s²] Die bei einer Getriebefahrt von der Folgeachse erreichbare Verzögerung wird auf diesen Wert begrenzt. Dieser Parameter wird verwendet, wenn • eine bereits laufende Folgeachse sich in eine langsamere Leitachse einkoppelt, • die Folgeachse aufgrund eines Fehlers bremst, • die Leitachse stärker abbremst, als es die Folgeachse zulässt.
Master	IN-OUT	AXIS_REF	Verweis auf die Achsdaten der Leitachse
Slave	IN-OUT	AXIS_REF	Verweis auf die Achsdaten der Folgeachse
Init	IN-OUT	BOOL	Die Anweisung führt bei Init = TRUE ihre Initialisierung durch und setzt das Bit anschließend zurück.
Busy	OUTPUT	BOOL	1 = Auftrag läuft.
InGear	OUTPUT	BOOL	1 = Getriebegeschwindigkeit erreicht Wird gesetzt, wenn die Getriebegeschwindigkeit das erste Mal erreicht wurde. Ist Execute zu diesem Zeitpunkt bereits zurückgesetzt, steht InGear trotzdem für einen Aufruf der Anweisung an. Wird zurückgesetzt mit fallendem Execute, bei Fehler und beim Ablösen der Fahrt.
Coupled	OUTPUT	BOOL	1 = Achse ist momentan an die Leitachse gekoppelt Gibt in jedem Bearbeitungszyklus an, ob die Getriebegeschwindigkeit erreicht wurde.
CommandAborted	OUTPUT	BOOL	1 = Auftrag wurde durch einen anderen Auftrag abgebrochen.
Error	OUTPUT	BOOL	1 = Auftrag wurde durch einen erkannten Achsfehler abgebrochen. Das Bit bleibt solange anstehen, bis "Execute" zurückgesetzt wird, mindestens jedoch für einen Aufruf der Anweisung.

Eingangstreiber

Aufgabe der Eingangstreiber

Aufgabe

Der Eingangstreiber dient zur Erfassung der Istpositionen einer Achse. Er liest die unnormierte Achsposition und wandelt diese in die gewählte Längeneinheit um.

Für eine Reihe von Wegerfassungsbaugruppen stehen passende Eingangstreiber zur Verfügung, die nachfolgend einzeln beschrieben werden. Eine Anpassung an nicht direkt unterstützte Konfigurationen ist durch die Anweisung EncoderUniversal (Seite 6096) möglich.

Initialisierung

- Bei einem **Inkrementalgeber** wird die Synchronisation mit der Achse rückgesetzt (Parameter "Sync" = FALSE in den Achsdaten). Danach ist eine erneute Referenzpunktfahrt oder ein Bezugspunktsetzen erforderlich.
- Bei einem **Absolutwertgeber** bleibt die Synchronisation mit der Achse erhalten.

Aufruf

Der Eingangstreiber einer Achse muss ohne Bedingung aufgerufen werden.

Taktsynchronität

Die Eingangstreiber unterstützen den taktsynchronen Betrieb der dafür geeigneten Peripheriebaugruppen. Der Treiber erkennt den taktsynchronen Betrieb automatisch daran, dass er in einem taktsynchronen OB aufgerufen wird.

Fehlerbehandlung

Es gibt folgende Mechanismen zur Fehlererkennung:

- Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Eingangstreiber am Eingang "EncErr" mitgeteilt werden. Solange der Eingang "EncErr" gleich TRUE ist, greift der Eingangstreiber nicht auf die Wegerfassungsbaugruppe zu.
- Einige Wegerfassungsbaugruppen liefern Fehlerinformationen, die vom Eingangstreiber selbständig ausgewertet werden.
- Andere Wegerfassungsbaugruppen liefern ihre Fehlerinformation nur über Diagnosealarm und Datensatz 1 (DS1). Deshalb müssen Sie
  - in der Hardware-Konfiguration den Diagnosealarm für die Baugruppe freigeben und
  - im OB 82 die im Lokaldatum "OB82_MDL_ADDR" gelieferte Adresse mit Ihrer Baugruppenadresse vergleichen. Stimmen diese überein, setzen Sie den Parameter "ReadDiagErr" des Eingangstreibers auf TRUE (sowohl beim Kommen als auch beim Gehen des Alarms).
    Durch "ReadDiagErr" = TRUE wird der Eingangstreiber veranlasst, den Datensatz 1 der Baugruppe über die Anweisung RD_REC zu lesen und im statischen Parameter "DiagStatus" abzulegen.
    Anschließend setzt er "ReadDiagErr" = FALSE. Der Rückgabewert der Anweisung RD_REC wird im statischen Parameter "JobErr" abgelegt

Fehlerreaktion

- Im Fehlerfall werden die Anzeigen "Error" und "Err.EncoderErr" im Achs-DB gesetzt.
- Bei einem **Inkrementalgeber** wird das Statusbit "Sync" im Achs-DB in jedem Fall rückgesetzt.
- Bei einem **Absolutwertgeber** wird das Statusbit "Sync" im Achs-DB nie zurückgesetzt.

Bitte beachten Sie auch die spezifische Fehlerbehandlung in den folgenden Kapiteln zu den einzelnen Treibern.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter "Fehleranzeigen und Fehlerquittung im Achs-DB".

Simulationsmodus

Solange das Bit Sim im Achs-DB gesetzt ist, greift der Eingangstreiber nicht auf die Wegerfassungsbaugruppe zu. Die Geberwerte werden über die Anweisung MC_Simulation (Seite 6067) simuliert.

EncoderIM178

Beschreibung

Die Anweisung EncoderIM178 ist aus Gründen der Kompatibilität mit STEP7 V5.x enthalten. Die Baugruppe IM 178-4 wird nicht mehr aktiv vermarktet. Als Alternative können Sie die IM 174 einsetzen.

Die IM 178-4 muss mit gleicher Kanalnummer und gleicher Achse beim Eingangstreiber **und** beim Ausgangstreiber verwendet werden, da sich bestimmte Adresseinstellungen auf beide Treiber beziehen. Eine Mischung mit anderen Treibern ist nicht möglich.

Parametrierung der IM 178-4 (Hardware-Konfiguration)

Selektieren Sie die IM 178-4 aus dem Hardware-Katalog unter "Dezentrale Peripherie". Projektieren Sie den Steckplatz 4 der IM 178-4 als "4Wort AA/12Wort AE/Kons. 1Wort".

Parameter	Einstellung
Synchronisation	Software
Flankenauswahl	Steigende, fallende oder beide Flanken.

Die übrigen Einstellungen parametrieren Sie je nach angeschlossenem Geber.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
RefMode	INPUT	BOOL	Art der Referenzierung: 0 = mit Digitaleingang I0 1 = mit Digitaleingang I1 und Nullmarke
DI_0	OUTPUT	BOOL	Digitaleingang 0 des entsprechenden Kanals der IM 178-4
DI_1	OUTPUT	BOOL	Digitaleingang 1 des entsprechenden Kanals der IM 178-4
DI_2	OUTPUT	BOOL	Digitaleingang 2 des entsprechenden Kanals der IM 178-4
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
Status.PARA	STAT	BOOL	PARA = TRUE: IM 178-4 wurde über die Hardware-Konfiguration korrekt parametriert. PARA = FALSE: siehe Handbuch SIMATIC IM 178-4 Drive Interface (http://support.automation.siemens.com/WW/view/de/1117389)
Status.EXTF0	STAT	BOOL	1 = Gebersignalleitungsfehler
Status.EXTF1	STAT	BOOL	1 = Geberfehler
Status.EXTF2	STAT	BOOL	1 = Nullmarkenfehler

Synchronisation

Die Synchronisation erfolgt alternativ:

- Durch Referenzpunktfahrt (nur bei Inkrementalgeber)
- Durch Bezugspunktsetzen (bei Inkremental- und Absolutwertgeber)

Den Referenzpunktschalter schließen Sie je nach Einstellung am Digitaleingang I0 oder I1 an.

Bei einer Referenzpunktfahrt kann die IM 178-4 den Referenzpunkt wahlweise in folgenden Fällen erkennen:

- Bei fallender oder steigender Flanke oder bei beiden Flanken am Digitaleingang I0 (RefMode = 0).
- Bei der steigenden Flanke der ersten Nullmarke nach dem Erkennen der fallenden Flanke am Digitaleingang I1 (RefMode = 1).

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie wie beim Bezugspunktsetzen auf die aktuelle Position referenziert.

Digitaleingänge

Die Digitaleingänge I0 bis I2 des gewählten Kanals der IM 178-4 werden in den Ausgangsparametern "DI_0" bis "DI_2" des Eingangstreibers angezeigt.

Fehlerbehandlung

Als Fehlerreaktion setzt der Eingangstreiber bei erkanntem Fehler die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB. Der Eingangstreiber setzt das Statusbit "Sync" im Achs-DB zurück, falls Sie einen Inkrementalgeber verwenden. Näheres entnehmen Sie bitte dem Handbuch SIMATIC IM 178-4 Drive Interface (http://support.automation.siemens.com/WW/view/de/1117389).

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter "Fehleranzeigen und Fehlerquittung im Achs-DB".

Fehlerquittierung

Wenn Sie die Fehlerquittung im Achs-DB von Easy Motion Control setzen, dann werden auch in der Baugruppe anstehende Fehler quittiert.

EncoderFM450

Beschreibung

Die Anweisung EncoderFM450 dient zum Einsatz eines Kanals der Baugruppe FM 450-1 als Wegerfassungsbaugruppe für Inkrementalgeber. Die Kanalnummer wird über den Achs-DB parametriert.

Parametrierung der Baugruppe (Hardware-Konfiguration)

Die Baugruppe FM 450-1 ist mit den Einstellungen gemäß der nachfolgenden Tabelle zu parametrieren.

Dialog	Einstellung
Alarmauswahl	Diagnosealarm freigeben
Betriebsarten	• Zählbereich: -31 bis +31 Bit • Betriebsart: Endlos Zählen • Torsteuerung: keine
Eingänge	Wenn Sie eine Referenzpunktfahrt durchführen wollen, müssen Sie folgende Einstellung wählen: • Synchronisation: einmalig • Die Einstellung "Nullmarke für das Setzen auswerten" kann als Referenzpunktkriterium ein- oder ausgeschaltet werden.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
ReadDiagErr	IN-OUT	BOOL	Setzen Sie dieses Bit im OB 82, wenn die Baugruppe einen Diagnosealarm ausgelöst hat (siehe unten unter Fehlerbehandlung).
DI_0	OUTPUT	BOOL	Digitaleingang 0 des gewählten Kanals
DI_1	OUTPUT	BOOL	Digitaleingang 1 des gewählten Kanals
DI_2	OUTPUT	BOOL	Digitaleingang 2 des gewählten Kanals
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
JobErr	STAT	INT	Rückgabewert der Anweisung RD_REC, wenn "ReadDiagErr" = TRUE gesetzt wurde.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Externe Hilfsspannung fehlerhaft
DiagStatus.CONFIG_ERR	STAT	STRUCT	Parametrierung fehlerhaft
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	Zeitüberwachung (Watchdog) angesprochen
DiagStatus.RAM_FLT	STAT	STRUCT	RAM defekt
DiagStatus.CH1_SIGA	STAT	STRUCT	Kanal 1 Signal A fehlerhaft
DiagStatus.CH1_SIGB	STAT	STRUCT	Kanal 1 Signal B fehlerhaft
DiagStatus.CH1_SIGZ	STAT	STRUCT	Kanal 1 Signal N fehlerhaft
DiagStatus.CH1_BETW	STAT	STRUCT	Kanal 1 Fehler zwischen Kanälen
DiagStatus.CH1_5V2	STAT	STRUCT	Kanal 1 Geberversorgung 5,2 V fehlerhaft
DiagStatus.CH2_SIGA	STAT	STRUCT	Kanal 2 Signal A fehlerhaft
DiagStatus.CH2_SIGB	STAT	STRUCT	Kanal 2 Signal B fehlerhaft
DiagStatus.CH2_SIGZ	STAT	STRUCT	Kanal 2 Signal N fehlerhaft
DiagStatus.CH2_BETW	STAT	STRUCT	Kanal 2 Fehler zwischen Kanälen
DiagStatus.CH2_5V2	STAT	STRUCT	Kanal 2 Geberversorgung 5,2 V fehlerhaft

Synchronisation

Die Synchronisation kann entweder

- durch Referenzpunktfahrt oder
- durch Bezugspunktsetzen erfolgen.

Den Referenzpunktschalter schließen Sie am Digitaleingang I2 an.

Bei einer Referenzpunktfahrt kann die Baugruppe den Referenzpunkt wahlweise in folgenden Fällen erkennen:

- bei einer steigenden Flanke am Digitaleingang I2 oder
- an der steigenden Flanke der Nullmarke, wenn der Digitaleingang I2 gesetzt ist.

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie wie beim Bezugspunktsetzen auf die aktuelle Position referenziert.

Digitaleingänge

Die Digitaleingänge I0 bis I2 des gewählten Kanals werden in den Ausgangsparametern "DI_0" bis "DI_2" des Eingangstreibers angezeigt.

Fehlerbehandlung

Die Baugruppe liefert ihre Fehlerinformation über Datensatz 1 (DS1). In folgenden Fällen liest der Eingangstreiber den Datensatz 1 der Baugruppe über die Anweisung RD_REC und legt ihn im statischen Parameter "DiagStatus" ab:

- Der Eingangstreiber wird initialisiert.
- Die Baugruppe meldet, dass sie einen Fehler erkannt hat.
- Der Eingangsparameter "ReadDiagErr" des Treibers ist TRUE.

Anschließend setzt er "ReadDiagErr" = FALSE. Der Rückgabewert der Anweisung RD_REC wird im statischen Parameter "JobErr" abgelegt (siehe JOB_Err-Meldungen).

Ist im Datensatz 1 eine Fehleranzeige eingetragen, setzt der Eingangstreiber im Achs-DB die Fehlerkennungen "Error" und "Err.EncoderErr" auf TRUE. Das Statusbit "Sync" wird auf FALSE gesetzt.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

Hinweis

Benutzen Sie eine FM 450-1 nur einkanalig, müssen Sie den zweiten Kanal mit der Geberart "24 V Spur A+B phasenversetzt" parametrieren, da dieser sonst zyklisch Diagnosealarme (z. B. wegen fehlender Beschaltung) sendet.

EncoderET200S1SSI

Beschreibung

Die Anweisung EncoderET200S1SSI dient zum Einsatz des Moduls ET 200S 1SSI Fast Mode als Wegerfassungsbaugruppe für Absolutwertgeber.

Der Treiber kann auch für die taktsynchrone Variante dieses Moduls eingesetzt werden.

Parametrierung des Moduls ET 200S 1SSI Fast Mode (Hardware-Konfiguration)

Der Eingangstreiber passt nur zum Modul 1SSI mit der Parameter-Einstellung "**Fast Mode**" als Betriebsmodus. Im TIA-Portal wird der "**Fast Mode**" nicht mehr über das Modul gewählt, sondern als Betriebsmodus eingestellt.

Selektieren Sie das Modul aus dem Hardware-Katalog unter "Dezentrale Peripherie > ET 200S > FM > Wegerfassen".

Parameter	Einstellung
Sammeldiagnose	Können Sie aktivieren, wenn Sie in Ihrem Programm eine Auswertung vornehmen wollen.
Erfassung	Freilaufend
Drehrichtungsumkehr	Deaktiviert Eine Invertierung der Drehrichtung ist mit den Anweisungen von Easy Motion Control möglich (Parameter "Richtungsanpassung Geber").
Betriebsmodus	Fast mode

Die übrigen Einstellungen parametrieren Sie je nach angeschlossenem Geber.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
Status.STS_UP	STAT	BOOL	Richtung vorwärts: Geberwerte werden größer
Status.STS_DN	STAT	BOOL	Richtung rückwärts: Geberwerte werden kleiner
Status.STS_DI	STAT	BOOL	Status des Digitaleingangs
Status.EXTF	STAT	BOOL	Sammelfehler oder Kurzschluss der Geberversorgung
Status.ERR_PARA	STAT	BOOL	Parametrierfehler
Status.RDY	STAT	BOOL	Baugruppe ist betriebsbereit

Synchronisation

Mit dem Modul ET 200S 1SSI Fast Mode ist eine Synchronisation nur über Bezugspunktsetzen möglich.

Fehlerbehandlung

Wenn der Eingangstreiber eine Fehleranzeige des Moduls erkennt, dann setzt er die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

EncoderAbsSensorDP

Beschreibung

Die Anweisung EncoderAbsSensorDP dient zum Einsatz des Absolutwertgebers SIMODRIVE sensor mit PROFIBUS DP zur Lageerfassung.

Parametrierung des Absolutwertgebers SIMODRIVE sensor (Hardware-Konfiguration)

Wählen Sie aus dem Hardware-Katalog den Geber "PROFIBUS DP > ENCODER > SIMODRIVE sensor > Version 2.2 Singleturn oder Version 2.2 Multiturn" aus. Ist der SIMODRIVE sensor noch nicht verfügbar, binden Sie diesen über die entsprechende GSD-Datei ein. Sie finden den Geber dann im Ordner "Weitere Feldgeräte > PROFIBUS DP > Encoder > Siemens AG > SIMODRIVE sensor".

Parameter	Einstellung	
Zählrichtung	Steigend im Uhrzeigersinn Eine Invertierung der Drehrichtung ist mit den Anweisungen von Easy Motion Control möglich (Parameter "Richtungsanpassung Geber").	
	Multiturn	Singleturn
Skalierungsfunktion	Eingeschaltet	Eingeschaltet
Gewünschte Messschritte (high)	0	-
Gewünschte Messschritte (low)	4096	4096
Physikalische Messschritte (high)	0	-
Physikalische Messschritte (low)	4096	4096
Gewünschte Auflösung pro	Umdrehung	Umdrehung
Gesamtauflösung (high)	256	-
Gesamtauflösung (low)	0	4096
Inbetriebnahmemodus	Ausgeschaltet	Ausgeschaltet
kürzere Diagnose	Nein	Nein
Unterer Endschalter	Ausgeschaltet	Ausgeschaltet
Unterer Endschalter (high)	0	-
Unterer Endschalter (low)	0	0
Oberer Endschalter	Ausgeschaltet	Ausgeschaltet
Oberer Endschalter (high)	0	-
Oberer Endschalter (low)	32767	4096
Geschwindigkeitsausgabe	Schritte/1000 ms	Schritte/1000 ms

Die Anpassung an Ihre gewählte Längeneinheit nehmen Sie mit den Anweisungen von Easy Motion Control vor. Daher kann der Geber - wie oben angegeben - fest parametriert werden. Ein "Teach-In" des Gebers ist nicht erforderlich (siehe Benutzeranleitung SIMODRIVE sensor Absolutwertgeber mit PROFIBUS DP (http://support.automation.siemens.com/WW/view/de/28813901)).

Am Parameter "Schritte pro Geberumdrehung" von Easy Motion Control geben Sie 4096 ein ("StepsPerRev").

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
Status.Ready	STAT	BOOL	1 = Geber ist betriebsbereit
Status.Mode	STAT	BOOL	Betriebsart: 0 = Inbetriebnahmemodus, 1 = Normalmodus
Status.SWLimitExceeded	STAT	BOOL	1 = Software-Endschalter MIN oder MAX hat angesprochen
Status.Dir	STAT	BOOL	Drehrichtung: 0 = im Uhrzeigersinn, 1 = entgegen Uhrzeigersinn

Synchronisation

Mit dem Absolutwertgeber SIMODRIVE sensor ist eine Synchronisation nur über Bezugspunktsetzen möglich.

Fehlerbehandlung

Wenn der Eingangstreiber erkennt, dass eine Fehlermeldung des Gebers ansteht, dann setzt er die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB. Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

EncoderSM338

Beschreibung

Die Anweisung EncoderSM338 dient zum Einsatz der SM 338 POS_INPUT als Wegerfassungsbaugruppe für Absolutwertgeber.

Der Treiber kann auch für die taktsynchrone Variante dieser Baugruppe eingesetzt werden.

Parametrierung der SM 338 (Hardware-Konfiguration)

Die FREEZE-Funktion muss ausgeschaltet sein.

Erkennt der Eingangstreiber den FREEZE-Zustand des Gebereingangs, dann setzt er die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB.

Die übrigen Einstellungen parametrieren Sie je nach angeschlossenem Geber.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
ReadDiagErr	IN-OUT	BOOL	Setzen Sie dieses Bit im OB 82, wenn die Baugruppe einen Diagnosealarm ausgelöst hat (siehe Fehlerbehandlung).
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
Freeze	STAT	BOOL	FREEZE-Zustand des Gebereingangs
JobErr	STAT	INT	Rückgabewert der Anweisung RD_REC, wenn "ReadDiagErr" = TRUE gesetzt wurde.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Externe Hilfsspannung fehlerhaft
DiagStatus.CONFIG_ERR	STAT	STRUCT	Parametrierung fehlerhaft
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	Zeitüberwachung (Watchdog) angesprochen
DiagStatus.RAM_FLT	STAT	STRUCT	RAM defekt
DiagStatus.CH0_INTF	STAT	STRUCT	Projektierungs- oder Parametrierfehler (Interner Kanalfehler)
DiagStatus.CH0_EXTF	STAT	STRUCT	Geberfehler (Externer Kanalfehler)

Synchronisation

Mit der SM 338 ist eine Synchronisation nur über Bezugspunktsetzen möglich.

Fehlerbehandlung

Die Baugruppe liefert ihre Fehlerinformation nur über Diagnosealarm und Datensatz 1 (DS1). Deshalb müssen Sie

- in der Hardware-Konfiguration den Diagnosealarm für die Baugruppe freigeben und
- im OB 82 die im Lokaldatum "OB82_MDL_ADDR" gelieferte Adresse mit Ihrer Baugruppenadresse vergleichen. Stimmen diese überein, setzen Sie den Parameter ReadDiagErr des Eingangstreibers auf TRUE (sowohl beim Kommen als auch beim Gehen des Alarms).
  Durch "ReadDiagErr" = TRUE wird der Eingangstreiber veranlasst, den Datensatz 1 der Baugruppe über die Anweisung RD_REC zu lesen und im statischen Parameter "DiagStatus" abzulegen.
  Anschließend setzt er "ReadDiagErr" = FALSE. Der Rückgabewert der Anweisung RD_REC wird im statischen Parameter "JobErr" abgelegt (siehe JOB_Err-Meldungen).

Ist im Datensatz 1 eine Fehleranzeige eingetragen, setzt der Eingangstreiber im Achs-DB die Fehlerkennungen "Error" und "Err.EncoderErr" auf TRUE.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

EncoderET200S1Count

Beschreibung

Die Anweisung EncoderET200S1Count dient zum Einsatz des Moduls ET 200S 1Count5V oder 1Count24V als Wegerfassungsbaugruppe für Inkrementalgeber.

Der Treiber kann auch für die taktsynchrone Variante dieses Moduls eingesetzt werden.

Parametrierung des Moduls ET 200S 1Count (Hardware-Konfiguration)

Der Eingangstreiber passt nur zu den Modulen "1 Count 5V" bzw. "1 Count 24V" aus dem Hardware-Katalog. Der Zählbetrieb wird nicht mehr über das Modul ausgewählt sondern als Betriebsmodus.

Selektieren Sie das Modul aus dem Hardware-Katalog unter "Dezentrale Peripherie > ET 200S > FM > Zählen".

Parameter	Einstellung
Sammeldiagnose	Können Sie aktivieren, wenn Sie in Ihrem Programm eine Auswertung vornehmen wollen.
Zählbetriebsart	Endloses Zählen
Funktion DI, Synchronisation	• Falls Sie eine Referenzpunktfahrt durchführen wollen, wählen Sie – Funktion DI = Synchronisation bei positiver Flanke und – Synchronisation = einmalig. • Andernfalls wählen Sie: – Funktion DI = Eingang
Torfunktion	Beliebig
Betriebsmodus	Zählen

Die übrigen Einstellungen parametrieren Sie je nach angeschlossenem Geber.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
DOut_1	INPUT	BOOL	Digitalausgang 1
DOut_2	INPUT	BOOL	Digitalausgang 2 (nur bei 1Count5V vorhanden)
DIn	OUTPUT	BOOL	Digitaleingang des Moduls
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
Status.ERR_ENCODER	STAT	BOOL	Kurzschluss / Drahtbruch Gebersignal (nur bei 1Count5V)
Status.ERR_DO2		BOOL	Kurzschluss / Drahtbruch / Übertemperatur DO2 (nur bei 1Count5V)
Status.ERR_PARA		BOOL	Parametrierfehler
Status.ERR_DO1		BOOL	Kurzschluss / Drahtbruch / Übertemperatur DO1
Status.ERR_24V		BOOL	Kurzschluss Geberversorgung

Synchronisation

Die Synchronisation erfolgt alternativ:

- Durch Referenzpunktfahrt
- Durch Bezugspunktsetzen

Den Referenzpunktschalter schließen Sie am Digitaleingang DI an.

Bei einer Referenzpunktfahrt kann das Modul ET 200S 1Count den Referenzpunkt bei einer steigenden Flanke am Digitaleingang DI erkennen.

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie (wie beim Bezugspunktsetzen) auf die aktuelle Position referenziert.

Digitaleingang

Der Digitaleingang des Moduls wird im Ausgangsparameter "DIn" des Eingangstreibers angezeigt.

Digitalausgänge

Die Digitalausgänge können über die Eingangsparameter "DOut_1" und "DOut_2" des Eingangstreibers angesteuert werden.

Fehlerbehandlung

Erkennt der Eingangstreiber eine Fehleranzeige des Moduls, dann setzt er die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB und setzt das Statusbit "Sync" im Achs-DB zurück.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

EncoderFM350

Beschreibung

Die Anweisung EncoderFM350 dient zum Einsatz der Baugruppe FM 350-1 als Wegerfassungsbaugruppe für Inkrementalgeber.

Der Treiber kann auch für die taktsynchrone Variante dieser Baugruppe eingesetzt werden.

Parametrierung der Baugruppe (Hardware-Konfiguration)

Dialog	Einstellung
Alarmauswahl	Diagnosealarme freigeben
Betriebsarten	• Zählbereich: -31 bis +31 Bit • Betriebsart: Endlos Zählen • Torsteuerung: keine
Eingänge	Wenn Sie eine Referenzpunktfahrt durchführen wollen, müssen Sie folgende Einstellung wählen: • Zähler setzen (DI-Set): einmalig • Die Einstellung "Nullmarke für das Setzen auswerten" kann als Referenzpunktkriterium ein- oder ausgeschaltet werden.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
ReadDiagErr	IN-OUT	BOOL	Setzen Sie dieses Bit im OB 82, wenn die Baugruppe einen Diagnosealarm ausgelöst hat (siehe Fehlerbehandlung).
DI_0	OUTPUT	BOOL	Digitaleingang 0
DI_1	OUTPUT	BOOL	Digitaleingang 1
DI_2	OUTPUT	BOOL	Digitaleingang 2
Die folgenden statischen Parameter der Anweisung dienen nur zu Diagnosezwecken im Fehlerfall.			
JobErr	STAT	INT	Rückgabewert der Anweisung RD_REC, wenn "ReadDiagErr" = TRUE gesetzt wurde.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Externe Hilfsspannung fehlerhaft
DiagStatus.NO_CONFIG	STAT	STRUCT	Parametrierung fehlt
DiagStatus.CONFIG_ERR	STAT	STRUCT	Parametrierung fehlerhaft
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	Zeitüberwachung (Watchdog) angesprochen
DiagStatus.RAM_FLT	STAT	STRUCT	RAM defekt
DiagStatus.CH1_SIGA	STAT	STRUCT	Signal A fehlerhaft
DiagStatus.CH1_SIGB	STAT	STRUCT	Signal B fehlerhaft
DiagStatus.CH1_SIGZ	STAT	STRUCT	Signal N fehlerhaft
DiagStatus.CH1_BETW	STAT	STRUCT	Fehler zwischen Kanälen
DiagStatus.CH1_5V2	STAT	STRUCT	Geberversorgung 5,2 V fehlerhaft

Synchronisation

Die Synchronisation kann entweder

- durch Referenzpunktfahrt oder
- durch Bezugspunktsetzen erfolgen.

Den Referenzpunktschalter schließen Sie am Digitaleingang I2 an.

Bei einer Referenzpunktfahrt kann die Baugruppe den Referenzpunkt wahlweise in folgenden Fällen erkennen:

- steigende Flanke am Digitaleingang I2
- steigende Flanke der Nullmarke, wenn der Digitaleingang I2 gesetzt ist

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie wie beim Bezugspunktsetzen auf die aktuelle Position referenziert.

Digitaleingänge

Die Digitaleingänge I0 bis I2 werden in den Ausgangsparametern "DI_0" bis "DI_2" des Eingangstreibers angezeigt.

Fehlerbehandlung

Die Baugruppe liefert ihre Fehlerinformation über Datensatz 1 (DS1). In folgenden Fällen liest der Eingangstreiber den Datensatz 1 der Baugruppe über die Anweisung RD_REC und legt ihn im statischen Parameter "DiagStatus" ab:

- Der Eingangstreiber wird initialisiert.
- Die Baugruppe meldet, dass sie einen Fehler erkannt hat.
- Der Eingangsparameter "ReadDiagErr" des Treibers ist TRUE.

Anschließend setzt er "ReadDiagErr" = FALSE. Der Rückgabewert der Anweisung RD_REC wird im statischen Parameter "JobErr" abgelegt (siehe JOB_Err-Meldungen).

Ist im Datensatz 1 eine Fehleranzeige eingetragen, setzt der Eingangstreiber im Achs-DB die Fehlerkennungen "Error" und "Err.EncoderErr" auf TRUE. Das Statusbit "Sync" wird auf FALSE gesetzt.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

EncoderCPU314C

Beschreibung

Die Anweisung EncoderCPU314C dient zur Anbindung eines Zählkanals der CPU 314C zur Wegerfassung für 24 V-Inkrementalgeber.

Es sind bis zu 4 Zählkanäle nutzbar.

Parametrierung der Zählfunktion der CPU 314C (Hardware-Konfiguration)

Der Eingangstreiber passt nur zu einer CPU 314C ab Firmware-Version V2.0.

Stellen Sie für jeden Zählkanal, den Sie verwenden wollen, folgende Parameter im Eigenschaftsdialog der Baugruppe im Steckplatz 2.7 (Zählen) ein:

Parameter	Einstellung
Betriebsart	Endlos zählen
HW-Tor	Kein HW-Tor (Optionskästchen deaktiviert)

Die übrigen Einstellungen parametrieren Sie je nach angeschlossenem Geber und den Erfordernissen Ihrer Anwendung.

Geben Sie in den Achsdaten im Parameter "Anfangsadresse der Eingänge der Wegerfassungsbaugruppe" ("InputModuleInAddress") die Adresse ein, die Sie der Hardware-Konfiguration für die Eingänge am Steckplatz 2.7 (Zählen) vergeben haben. Den Parameter

"Anfangsadresse der Ausgänge der Wegerfassungsbaugruppe" ("InputModuleOutAddress") versorgen Sie mit der Adresse der Zähler-Ausgänge.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen vom Anwenderprogramm erkannten Fehler mit.

Synchronisation

Die Synchronisation kann entweder

- durch Referenzpunktfahrt oder
- durch Bezugspunktsetzen erfolgen.

Den Referenzpunktschalter schließen Sie am Latch-Digitaleingang des entsprechenden Kanals an (DI1.4 für Kanal 0 bis DI1.7 für Kanal 3).

Bei einer Referenzpunktfahrt kann die CPU 314C den Referenzpunkt bei einer steigenden Flanke am Digitaleingang DI erkennen.

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie wie beim Bezugspunktsetzen auf die aktuelle Position referenziert.

Bei einer Referenzpunktfahrt liest der Treiber zyklisch u. a. den Zustand des Referenzpunktschalters. Die dafür erforderlichen Zugriffe erfordern eine deutlich erhöhte Laufzeit der Anweisung.

Maximale Zählfrequenz

Hinweis

Wird die maximale Zählfrequenz des Zählers der CPU 314C überschritten, so werden keine Zählwerte ausgegeben (siehe Betriebsanleitung SIMATIC S7-300 CPU 31xC Technologische Funktionen (http://support.automation.siemens.com/WW/view/de/12429336)). Die Achse wird mit dem Fehler "Schleppabstand überschritten" gestoppt.

Legen Sie daher die Maximalgeschwindigkeit ("MaxVelocity") in den Achsdaten unter Berücksichtigung der Grenzfrequenz des Zählers fest.

Fehlerbehandlung

Die CPU 314C liefert keine Fehlerinformation über den Geber.

EncoderIM174

Beschreibung

Die Anweisung EncoderIM174 dient zum Einsatz der IM 174 als Wegerfassungsbaugruppe für Inkremental- und Absolutwertgeber.

Die IM 174 stellt bis zu 4 Achsen zur Verfügung. Für jede Achse wird eine Treiberanweisung EncoderIM174 benötigt.

Parametrierung der IM 174 (Hardware-Konfiguration)

Fügen Sie eine IM 174 an einem taktsynchronen PROFIBUS DP in Ihr Projekt ein. Beachten Sie, dass die Ein- / Ausgangsadressen im Prozessabbild der CPU liegen. Notfalls korrigieren Sie diese durch manuelles Anpassen der Adressen. Weisen Sie die Ein- / Ausgangsadressen einem Teilprozessabbild zu. In der Konfiguration der S7-CPU muss dem Teilprozessabbild ein taktsynchroner OB zugeordnet werden.

Parameter	Einstellung
E/A Adresse	Für die in Easy Motion Control zu nutzende Achse ist die erste Eingangsadresse im Achs-DB anzugeben. (z. B. Slot 4: E-Adresse 256 bis 273, dann ist die Eingangsadresse 256 anzugeben)
Telegrammauswahl	In der Hardware-Konfiguration kann für jede Achse entweder PROFIBUS-Telegramm 3 oder Telegramm 81 ausgewählt werden. Diese Information müssen Sie auch am Eingang TelNumber der Anweisung EncoderIM174 angeben.
611U Konformmodus	Zur Nutzung einer externen Referenzmarke müssen Sie den 611U Konformmodus aktivieren.

Hinweis

Bei Verwendung des Telegramms 81 wird nur die Geberinformation von der IM 174 an die CPU übertragen. Das Telegramm 81 wird verwendet, wenn man mit der IM 174 nur einen Geber einlesen will.
In diesem Fall wird auch kein Ausgangstreiber für die IM 174 (OutputIM174) benötigt.

Hinweis

Standardmäßig wird von der Hardware-Konfiguration die Baugruppe IM 174 als taktsynchrone Baugruppe eingebunden. Taktsynchrone Bearbeitung bedingt, dass die Ein- / Ausgangsadressen der IM 174 im Prozessabbild der CPU liegen.

Weitergehende Informationen zum Thema Taktsynchronität unter SIMATIC S7 finden Sie im Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Die Treiberanweisung EncoderIM174 setzt taktsynchrone Bearbeitung voraus. Beachten Sie diese Vorschrift nicht, kann die Anweisung auf keine Hardware-Adressen zugreifen.

Parametrierung der IM 174 im Achs-DB

Die in der Hardware-Konfiguration für diese Achse eingestellte Eingangsadresse müssen Sie im Feld "Baugruppenadresse Eingänge" angeben.

Die Achsnummer für die IM 174 wird vom Eingangstreiber EncoderIM174 im laufenden Betrieb ermittelt. Diese kann in den Lokaldaten des zugehörigen DB ausgelesen werden.

Der Eingangstreiber EncoderIM174 ermittelt im laufenden Betrieb die eingestellte Zykluszeit im taktsynchronen PROFIBUS DP und trägt diese in den Achs-DB ein. Es wird daher dringend empfohlen, den veränderten Achs-DB aus der Steuerung zurück zu lesen, um eventuelle Änderungen zu übernehmen.

Aufruf

Die Anweisung wird in einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
RefMode	INPUT	INT	Art der Referenzierung: • 0 = Gebernullmarke • 1 = steigende Flanke an externer Nullmarke (setzt 611U Konformmodus voraus). • 4 = Messtaster 1 positive Flanke • 5 = Messtaster 1 negative Flanke • 6 = Messtaster 2 positive Flanke • 7 = Messtaster 2 negative Flanke
TelNumber	INPUT	INT	Nummer des verwendeten PROFIBUS-Telegramms: • 3 = PROFIBUS-Telegramm 3 • 81 = PROFIBUS-Telegramm 81
Err_ID	OUTPUT	WORD	Fehlercode der Anweisung: • W#16#0001 = Falscher OB, Anweisung darf nur in OB 6x installiert werden • W#16#0002 = Falsche Telegrammnummer, die IM 174 unterstützt nur PROFIBUS-Telegramm 3 und 81 • W#16#0004 = Falsche Achsnummer, die IM 174 unterstützt bis zu 4 Achsen. Die von Ihnen im Achs-DB angegebene Hardware-Adresse konnte keiner Achse einer IM 174 zugeordnet werden. • W#16#0008 = Falscher Referenziermodus angegeben. Prüfen Sie den Parameter am Eingang RefMode.

Synchronisation

Die Synchronisation erfolgt alternativ:

- Durch Referenzpunktfahrt (nur bei Inkrementalgeber)
- Durch Bezugspunktsetzen (bei Inkremental- und Absolutwertgeber)

Für Referenzpunktfahrten können zwei Arten von Referenzpunktschalter verwendet werden:

- Wenn Sie einen Geber mit Nullmarke verwenden und auf diese referenzieren wollen, parametrieren Sie den Eingang "RefMode" = 0. Der 611U Konformmodus darf dabei nicht angewählt werden.
- Möchten Sie auf eine externe Referenzmarke (BERO) referenzieren, parametrieren Sie den Eingang "RefMode" = 1. Den externen Referenzpunktschalter schließen Sie je nach der in Ihrer Anwendung genutzten Achse am Digitaleingang B1 bis B4 an. Der 611U Konformmodus muss dabei zwingend in der Hardware-Konfiguration angewählt werden.

Als Eingänge für die beiden einstellbaren Messtaster müssen die Digitaleingänge M1 oder M2 genutzt werden.

Weitergehende Informationen finden Sie im Gerätehandbuch Dezentrale Peripherie PROFIBUS-Baugruppe IM 174 (http://support.automation.siemens.com/WW/view/de/35014863).

Wenn sich die Achse beim Start der Referenzpunktfahrt bereits auf dem Referenzpunktschalter befindet, wird sie wie beim Bezugspunktsetzen auf die aktuelle Position referenziert.

Fehlerbehandlung

Als Fehlerreaktion setzt der Eingangstreiber bei erkanntem Fehler die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB. Der Eingangstreiber setzt das Statusbit "Sync" im Achs-DB zurück, falls Sie einen Inkrementalgeber verwenden.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

Fehlerquittierung

Wenn Sie die Fehlerquittung im Achs-DB von Easy Motion Control setzen, dann werden auch in der Baugruppe anstehende Fehler quittiert.

EncoderSINAMICS

Beschreibung

Die Anweisung EncoderSINAMICS dient zur Anbindung eines SINAMICS-Antriebs an Easy Motion Control. Die Anbindung kann dabei mittels PROFIBUS DP oder PROFINET IO erfolgen. Für jede SINAMICS-Achse wird eine Treiberanweisung EncoderSINAMICS benötigt.

Benötigte Werkzeuge

Für den Einsatz eines SINAMICS in Easy Motion Control und die Parametrierung eines SINAMICS benötigen Sie folgende Werkzeuge:

- Das Antriebsparametriertool STARTER oder S7-Technology inklusive aktuellem SINAMICS Support Package (SSP)

Parametrierung des SINAMICS (Hardware-Konfiguration)

Fügen Sie einen SINAMICS-Antrieb an einem PROFIBUS DP-Strang oder PROFINET IO-Strang in ihr Projekt ein. Die Treiberanweisung EncoderSINAMICS unterstützt für PROFIBUS DP die Standardtelegramme 3, 5, 102 und 105 sowie für PROFINET IO die Standardtelegramme 3 und 102.

Wenn Sie für Ihre Anwendung die Regelmethode Dynamic Servo Control (DSC) nutzen möchten, dann benötigen Sie ein geeignetes Telegramm (5, 105) und eine taktsynchrone Anbindung. Mit PROFINET IO ist diese DSC-Funktionalität zurzeit nicht möglich.

Für taktsynchrone Bearbeitung müssen die Ein- / Ausgangsadressen im Prozessabbild der CPU liegen. Notfalls korrigieren Sie diese durch manuelles Anpassen der Adressen. Weisen Sie die Ein- / Ausgangsadressen einem Teilprozessabbild zu. In der Konfiguration der S7-CPU muss dem Teilprozessabbild ein taktsynchroner OB zugeordnet werden. Weitergehende Informationen zum Thema Taktsynchronität unter SIMATIC S7 finden Sie im

Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Parameter	Einstellung
E/A Adresse	Für die in Easy Motion Control zu nutzende Achse ist die erste Eingangsadresse im Achs-DB anzugeben. (z. B. Slot 4: E-Adresse 256 bis 273, dann ist die Eingangsadresse 256 anzugeben)
Telegrammauswahl	Für die Kommunikation wird im Konfigurationsassistenten des SINAMICS eine Telegrammnummer ausgewählt und in die Hardware-Konfiguration übertragen. Diese Information müssen Sie auch am Eingang TelNumber der Anweisung EncoderSINAMICS angeben.

Hinweis

Die Standardinstallation des TIA-Portals unterstützt nicht die Anbindung des SINAMICS an eine SIMATIC S7-CPU. Prüfen Sie die Verfügbarkeit. Installieren Sie ggf. die entsprechende GSD-Datei für SINAMICS. Erst nach der zusätzlichen Installation werden die notwendigen Daten eingebunden.

Parametrierung des SINAMICS im Achs-DB

Die in der Hardware-Konfiguration für diese Achse eingestellte Eingangsadresse müssen Sie im Feld "Baugruppenadresse Eingänge" angeben.

Der Eingangstreiber EncoderSINAMICS ermittelt im laufenden Betrieb die eingestellte Zykluszeit, wenn ein äquidistanter oder taktsynchroner Betrieb gefunden wurde und trägt diese in den Achs-DB ein. Es wird daher dringend empfohlen, den veränderten Achs-DB aus der Steuerung zurück zu lesen, um eventuelle Änderungen zu übernehmen.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	
RefMode	INPUT	INT	Art der Referenzierung:	
			0 = Funktion 1 Referenzmarke 1	Referenzieren auf Gebernullmarke oder Nullmarkenersatz (p495[0] im SINAMICS)
			1 = Funktion 2 Referenzmarke 2	Zweite Gebernullmarke
			2 = Funktion 3 Referenzmarke 3	Dritte Gebernullmarke
			3 = Funktion 4 Referenzmarke 4	Vierte Gebernullmarke
TelNumber	INPUT	INT	Telegrammnummer:	
			Unterstützt werden folgende Standardtelegramme: 3, 5, 102 und 105	Für PROFINET IO sind nur die Telegramme 3 und 102 freigegeben.

Parameter	Deklaration	Datentyp	Beschreibung	
EncoderValid	OUTPUT	BOOL	1= Lageistwerte sind gültig	SINAMICS meldet, dass der Lagegeber von der betroffenen Achse gültige Werte liefert. Dieser Eingang kann als Einschaltvoraussetzung an MC_Control benutzt werden.
Err_ID	OUTPUT	WORD	Fehlercode der Anweisung:	
			W#16#0001 = Falscher OB	Anweisung darf nur in OB 6x, OB 3x oder OB 1 aufgerufen werden.
			W#16#0002 = Falsche Telegrammnummer	Die angegebene Nummer wird vom Treiber nicht unterstützt.
			W#16#0004 = Falscher Referenziermodus angegeben	Prüfen Sie den Parameter am Eingang RefMode. Gültige Werte sind 0, 1, 2 oder 3.

Hinweis

Die Telegrammnummer am Eingangstreiber muss mit der für SINAMICS projektierten Telegrammnummer identisch sein. Sonst wird die einwandfreie Funktionalität nicht gewährleistet. Im Fall einer falschen Eingabe wird kein Fehlercode ausgegeben.

Synchronisation

Die Synchronisation erfolgt alternativ:

- Durch Referenzpunktfahrt (nur bei Inkrementalgeber)
- Durch Referenzpunktsetzen (bei Inkremental- und Absolutwertgeber)

Die Referenzierart kann durch den Eingangsparameter "RefMode" am Eingangstreiber bestimmt werden. Die folgende Tabelle zeigt die für Easy Motion Control bestehenden Möglichkeiten.

RefMode	"Velocity" an MC_Home	Gebernullmarkensignal	Nullmarkenersatz p495[0]	Beschreibung
0	> 0	erste Gebernullmarke	0	Referenzierfahrt auf erste Gebernullmarke
1	> 0	zweite Gebernullmarke	0	Referenzierfahrt auf zweite Gebernullmarke
2	> 0	dritte Gebernullmarke	0	Referenzierfahrt auf dritte Gebernullmarke
3	> 0	vierte Gebernullmarke	0	Referenzierfahrt auf vierte Gebernullmarke

Durch die Eingabe an RefMode wird das entsprechende Gebersteuerwort zum Antrieb so angesteuert, dass die gewünschte Funktionalität im SINAMICS realisiert wird. Der Eingangstreiber EncoderSINAMICS ermöglicht nur die Referenzpunktfahrt auf eine oder mehrere Gebernullmarken sowie auf externe Nullmarken (Nullmarkenersatz). Fliegendes Referenzieren ist nicht möglich. Um auf Nullmarkenersatz zu referenzieren, müssen Sie im SINAMICS den Parameterwert p495[0] der betroffenen Achse auf einen Wert zwischen 1 und 6 setzen. Um eine Referenzpunktfahrt zu aktivieren, muss an der Anweisung MC_HOME eine Fahrgeschwindigkeit größer als Null eingegeben werden.

Wenn hier als Fahrgeschwindigkeit Null eingegeben wird, dann werden die Inhalte vom Eingang "RefMode" und vom Parameter p495[0] nicht berücksichtigt.

Fehlerbehandlung

Als Fehlerreaktion setzt der Eingangstreiber bei erkanntem Fehler die Fehlerkennungen "Error" und "Err.EncoderErr" im Achs-DB. Der Eingangstreiber setzt das Statusbit "Sync" im Achs-DB zurück, falls Sie einen Inkrementalgeber verwenden.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

Fehlerquittierung

Wenn Sie die Fehlerquittung im Achs-DB von Easy Motion Control setzen, dann werden auch in der Baugruppe anstehende Fehler quittiert.

EncoderUniversal

Beschreibung

Die Anweisung EncoderUniversal dient zur Anbindung von Wegerfassungsbaugruppen, für die Easy Motion Control keinen speziellen Eingangstreiber anbietet.

Arbeitsweise

Die Anweisung EncoderUniversal erwartet an ihrem Eingang "EncoderValue" einen Wert, der der Istposition Ihrer Achse entspricht. Sie greift selbst nicht auf die Peripherie zu. Sie können den von Ihrer Wegerfassungsbaugruppe gelieferten Wert direkt mit dem Treiber verschalten oder, wenn notwendig, noch vorher umrechnen.

Wenn Ihre Wegerfassungsbaugruppe spezielle Funktionen bietet, die Sie nutzen wollen, sollten Sie dies in Ihr Anwenderprogramm integrieren.

Parametrierung der Baugruppe (Hardware-Konfiguration)

Wählen Sie die Parameter entsprechend ihrer Hardware und dem eingesetzten Geber.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EncErr	INPUT	BOOL	1 = Geberfehler Hier teilen Sie dem Eingangstreiber einen von der Wegerfassungsbaugruppe oder dem Anwenderprogramm erkannten Fehler mit (z. B. Baugruppe gezogen oder Stationsausfall).
EncoderValue	INPUT	DINT	Geberwert

Synchronisation

Mit der Anweisung EncoderUniversal ist eine Synchronisation nur über Bezugspunktsetzen möglich.

Fehlerbehandlung

Fehler, die von Ihrer Wegerfassungsbaugruppe erkannt wurden, sollten Sie in Ihrem Anwenderprogramm erfassen und weiterverarbeiten.

Wenn ein solcher Fehler die Funktionen beeinträchtigt, die Sie mit Anweisungen von Easy Motion Control durchführen, dann sollten Sie diesen Fehler über den Eingang "EncErr" in die Fehlerbehandlung von Easy Motion Control einbinden.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

Ausgangstreiber

Aufgabe der Ausgangstreiber

Aufgabe

Der Ausgangstreiber dient zur Ausgabe des vom Regler errechneten Geschwindigkeitssollwertes einer Achse an die angeschlossene Ausgabebaugruppe. Die Anweisung rechnet die Geschwindigkeit von [Längeneinheit/s] in den passenden Wertebereich der Ausgabebaugruppe um und gibt diesen Wert aus.

Wenn der Eingang "EnableDrive" = FALSE ist, dann gibt der Ausgangstreiber Null an die Ausgabebaugruppe aus.

Initialisierung

Der Ausgangswert wird zu Null gesetzt.

Aufruf

Der Ausgangstreiber einer Achse muss ohne Bedingung aufgerufen werden.

Taktsynchronität

Die Ausgangstreiber unterstützen den taktsynchronen Betrieb der dafür geeigneten Peripheriebaugruppen. Der Treiber erkennt den taktsynchronen Betrieb automatisch daran, dass er in einem taktsynchronen OB aufgerufen wird.

Fehlerbehandlung

Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber am Eingang "OutErr" mitgeteilt werden. Solange der Eingang "OutErr" gleich TRUE ist, greift der Ausgangstreiber nicht auf die Ausgabebaugruppe zu.

Wenn der Eingang "OutErr" auf TRUE gesetzt ist, dann setzt der Ausgangstreiber die Fehlerkennungen "Error" und "Err.OutputErr" im Achs-DB.

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB.

Simulationsmodus

Solange das Bit "Sim" im Achs-DB gesetzt ist, greift der Ausgangstreiber nicht auf die Ausgabebaugruppe zu (siehe Anweisung MC_Simulation (Seite 6067)).

OutputIM178

Beschreibung

Die Anweisung OutputIM178 dient zum Einsatz eines Kanals der Baugruppe IM 178-4 als Ausgabebaugruppe. Die Anweisung OutputIM178 ist aus Gründen der Kompatibilität mit STEP7 V5.x enthalten. Die Baugruppe IM 178-4 wird nicht mehr aktiv vermarktet. Als Alternative können Sie die IM 174 einsetzen.

Die IM 178-4 muss mit gleicher Kanalnummer und gleicher Achse beim Eingangstreiber **und** beim Ausgangstreiber verwendet werden, da bestimmte Adresseinstellungen sich auf beide Treiber beziehen. Eine Mischung mit anderen Treibern ist nicht möglich.

Aufruf

Die Anweisung wird in einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Wenn "EnableDrive" = FALSE ist, dann wird der Wert Null zur Ausgabebaugruppe übertragen. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
Q0	INPUT	BOOL	Digitalausgang 0 des gewählten Kanals
Q1	INPUT	BOOL	Digitalausgang 1 des gewählten Kanals
Q2	INPUT	BOOL	Digitalausgang 2 des gewählten Kanals
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.

Digitalausgänge

Die Digitalausgänge "Q0" bis "Q2" des gewählten Kanals der IM 178-4 können über die Eingangsparameter "Q0" bis "Q2" des Ausgangstreibers angesteuert werden.

OutputSM432

Beschreibung

Die Anweisung OutputSM432 dient zum Einsatz eines Kanals der SM 432 als Ausgabebaugruppe.

Parametrierung der SM 432 (Hardware-Konfiguration)

Parametrieren Sie den entsprechenden Kanal der SM 432 auf ±10 V Ausgangsspannung.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Wenn "EnableDrive" = FALSE ist, dann wird der Wert Null zur Ausgabebaugruppe übertragen. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.

OutputET200S2AO

Beschreibung

Die Anweisung OutputET200S2AO dient zum Einsatz eines Kanals des ET 200S-Analogausgabemoduls 2AO U als Ausgabebaugruppe.

Der Treiber kann auch für die taktsynchrone Variante dieses Moduls eingesetzt werden.

Parametrierung des ET 200S-Analogausgabemoduls 2AO U (Hardware-Konfiguration)

Parametrieren Sie den entsprechenden Kanal des Moduls auf ±10 V Ausgangsspannung.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Wenn "EnableDrive" = FALSE ist, dann wird der Wert Null zur Ausgabebaugruppe übertragen. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.

OutputCPU314C

Beschreibung

Die Anweisung OutputCPU314C dient zum Einsatz eines Analogausgabekanals der CPU 314C für ±10 V Ausgangsspannung.

Parametrierung der Analogausgänge der CPU 314C (Hardware-Konfiguration)

Parametrieren Sie den entsprechenden Kanal (Analogausgang) der CPU 314C auf ±10 V Ausgangsspannung.

Geben Sie in den Achsdaten im Parameter "Anfangsadresse der Ausgänge der Ausgabebaugruppe" ("OutputModuleOutAddress") die Adresse ein, die Sie in der Hardware-Konfiguration für die Ausgänge am Steckplatz 2.6 (AI5/AO2) vergeben haben. Der Parameter "Anfangsadresse der Eingänge der Ausgabebaugruppe" ("OutputModuleInAddress") muss nicht versorgt werden.

Es sind maximal 2 Analogausgangs-Kanäle verfügbar, so dass Sie 2 Achsen vollständig über die CPU 314C betreiben können.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Wenn "EnableDrive" = FALSE ist, dann wird der Wert Null zum Analogausgang übertragen. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler am Analogausgang Ein vom Anwenderprogramm erkannter Fehler kann dem Ausgangstreiber hier mitgeteilt werden.

OutputSM332

Beschreibung

Die Anweisung OutputSM332 dient zum Einsatz eines Kanals der SM 332 als Ausgabebaugruppe.

Der Treiber kann auch für die taktsynchrone Variante dieser Baugruppe eingesetzt werden.

Parametrierung der SM 332 (Hardware-Konfiguration)

Parametrieren Sie den entsprechenden Kanal der SM 332 auf ±10 V Ausgangsspannung.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Wenn "EnableDrive" = FALSE ist, dann wird der Wert Null zur Ausgabebaugruppe übertragen. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.

OutputUniversal

Beschreibung

Mit der Anweisung OutputUniversal können Sie eine beliebige Analogausgabebaugruppe zusammen mit Anweisungen von Easy Motion Control einsetzen.

Arbeitsweise

Die Anweisung OutputUniversal errechnet aus der vom Regler ausgegebenen Geschwindigkeit mit Hilfe des Proportionalitätsfaktors "IntAtMaxVelocity" einen Drehzahlsollwert und stellt diesen an ihren Ausgängen zur Verfügung.

Der Treiber greift selbst nicht auf die Peripherie zu.

"IntAtMaxVelocity" ist der Wert, mit dem Sie Ihre Analogausgabebaugruppe ansteuern müssen, damit diese eine Spannung ausgibt, die die Achse mit "MaxVelocity" verfährt (den Wert dieser Spannung haben Sie bereits im Parameter "DriveInputAtMaxVel" festgelegt).

Die Kennlinie des Treibers geht immer durch Null und verläuft linear.

Der Drehzahlsollwert wird sowohl als ganzzahliger Wert mit Vorzeichen als auch als Betrag mit separatem Vorzeichen ausgegeben.

Wenn diese Ausgangswerte bereits Ihrer Applikation entsprechen, können Sie sie per Anwenderprogramm direkt an die Peripherie ausgeben. Passen die Werte noch nicht, können Sie ebenfalls per Anwenderprogramm eine zusätzliche Anpassung vornehmen und den dabei erzeugten Wert ausgeben.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Sobald "EnableDrive" = FALSE ist, wird an den Ausgängen der Wert Null ausgegeben. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
IntAtMaxVelocity	INPUT	INT	Integerwert zur Wandlung des Drehzahlsollwertes (siehe oben).
OutErr	INPUT	BOOL	1 = Fehler auf Ausgabebaugruppe Hier kann dem Ausgangstreiber ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) auf der Ausgabebaugruppe mitgeteilt werden.
OutputValue	OUTPUT	INT	Berechneter Drehzahlsollwert
OutValAbs	OUTPUT	INT	Betrag von Drehzahlsollwert
OutValSign	OUTPUT	BOOL	Vorzeichen von Drehzahlsollwert (TRUE = negativer Wert)

OutputMM4_DP

Beschreibung

Die Anweisung OutputMM4_DP dient zum Einsatz eines MICROMASTER 4 mit aufgesteckter DP-Optionsbaugruppe.

Parametrierung des MM4 (Hardware-Konfiguration)

Sie finden den MICROMASTER 4 im Hardware-Katalog unter:

"Weitere Feldgeräte > PROFIBUS DP > Antriebe > Siemens AG > SIMOVERT > MICROMASTER 4".

Eventuell müssen Sie dazu zunächst die entsprechende GSD-Datei installieren.

Platzieren Sie den MICROMASTER an Ihrem DP-Mastersystem und vergeben Sie die gewünschte DP-Adresse. An die Steckplätze 0 und 1 fügen Sie ein Modul "0 PKW, 2 PZD (PPO3)" ein. Übernehmen Sie die vorgeschlagenen Peripherie-Adressen oder wählen Sie eigene Werte.

Die hier eingetragenen Adressen übernehmen Sie in die "Anfangsadresse der Ausgänge der Ausgabebaugruppe" bzw. "Anfangsadresse der Eingänge der Ausgabebaugruppe" im Achs-DB Ihrer Achse.

Parametrierung des MM4

Parametrieren Sie den MICROMASTER nach Ihren Erfordernissen (Motordaten, DP-Adresse, Betrieb als DP-Slave, ...) anhand der MICROMASTER-Dokumentation.

Die im Parameter P2000 eingetragene "Bezugsfrequenz" müssen Sie als "Bezugswert für 100% Drehzahl" ("DriveInputAt100") in den Achs-DB Ihrer Achse übernehmen.

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Sobald "EnableDrive" = FALSE ist, wird an den MICROMASTER der Drehzahlwert Null ausgegeben. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler auf Ausgabe Ein vom Anwenderprogramm erkannter Fehler (z. B. Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.

Parameter	Deklaration	Datentyp	Beschreibung
ON	INPUT	BOOL	1 = Antrieb eingeschaltet "ON" wird im Steuerwort in das Bit 0 eingetragen.
DriveReady	OUTPUT	BOOL	1 = Antrieb fahrbereit Wenn "DriveReady" = FALSE, dann erzeugt das Starten einer Fahrt einen „DriveError“.

OutputIM174

Beschreibung

Die Anweisung OutputIM174 dient zum Einsatz eines Kanals der Baugruppe IM 174 als Ausgabebaugruppe. Die IM 174 stellt bis zu 4 Achsen zur Verfügung. Für jede Achse mit eingestelltem PROFIBUS-Telegramm 3 wird eine Treiberanweisung OutputIM174 benötigt.

Parametrierung der IM 174 (Hardware-Konfiguration)

Den Ausgangstreiber OutputIM174 benötigen Sie nur, wenn Sie eine Achse mit PROFIBUS-Telegramm 3 parametrieren. Für eine Achse mit PROFIBUS-Telegramm 81 benötigen Sie keinen Ausgangstreiber.

Fügen Sie eine IM 174 an einem taktsynchronen PROFIBUS DP in ihr Projekt ein. Beachten Sie, dass die Ein- / Ausgangsadressen im Prozessabbild der CPU liegen. Notfalls korrigieren Sie diese durch manuelles Anpassen der Adressen. Weisen Sie die Ein- / Ausgangsadressen einem Teilprozessabbild zu. In der Konfiguration der S7-CPU muss dem Teilprozessabbild ein taktsynchroner OB zugeordnet werden.

Parameter	Einstellung
E/A Adresse	Für die in Easy Motion Control zu nutzende Achse ist die erste Ausgangsadresse im Achs-DB anzugeben. (z. B. Slot 5: A-Adresse 256 bis 265, dann ist die Ausgangsadresse 256 anzugeben)

Hinweis

Standardmäßig wird von der Hardware-Konfiguration die Baugruppe IM 174 als taktsynchrone Baugruppe eingebunden. Taktsynchrone Bearbeitung bedingt, dass die Ein- / Ausgangsadressen der IM 174 im Prozessabbild der CPU liegen.

Weitergehende Informationen zum Thema Taktsynchronität unter SIMATIC S7 finden Sie im Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Die Treiberanweisung OutputIM174 setzt taktsynchrone Bearbeitung voraus. Beachten Sie diese Vorschrift nicht, kann die Anweisung auf keine Hardware-Adressen zugreifen.

Parametrierung der IM 174 im Achs DB

Die in der Hardware-Konfiguration für diese Achse eingestellte Ausgangsadresse müssen Sie im Feld "Baugruppenadresse Ausgänge" angeben.

Aufruf

Die Anweisung wird in einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EnableDrive	INPUT	BOOL	1 = Freigabe des Analogausgangs Sobald "EnableDrive" = FALSE ist, wird an den Antrieb der Drehzahlwert Null ausgegeben. Die Antriebsfreigabe wird gesperrt. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.
Err_ID	OUTPUT	WORD	Fehlercode der Anweisung W#16#0001 = Falscher OB, Anweisung darf nur in OB 6x installiert werden

OutputSINAMICS

Beschreibung

Die Anweisung OutputSINAMICS dient zur Anbindung eines SINAMICS-Antriebs an Easy Motion Control. Die Anbindung kann dabei mittels PROFIBUS DP oder PROFINET IO erfolgen. Für jede SINAMICS-Achse wird eine Treiberanweisung OutputSINAMICS benötigt.

Parametrierung des SINAMICS (Hardware-Konfiguration)

Fügen Sie einen SINAMICS-Antrieb an einem PROFIBUS DP-Strang oder PROFINET IO-Strang in ihr Projekt ein. Die Treiberanweisung OutputSINAMICS unterstützt für PROFIBUS DP die Standardtelegramme 3, 5, 102 und 105 sowie für PROFINET IO die Standardtelegramme 3 und 102.

Wenn Sie für Ihre Anwendung die Regelmethode Dynamic Servo Control (DSC) nutzen möchten, dann benötigen Sie ein geeignetes Telegramm (5, 105) und eine taktsynchrone Anbindung. Mit PROFINET IO ist diese DSC-Funktionalität zurzeit nicht möglich.

Für taktsynchrone Bearbeitung müssen die Ein- / Ausgangsadressen im Prozessabbild der CPU liegen. Notfalls korrigieren Sie diese durch manuelles Anpassen der Adressen. Weisen Sie die Ein- / Ausgangsadressen einem Teilprozessabbild zu. In der Konfiguration der S7-CPU muss dem Teilprozessabbild ein taktsynchroner OB zugeordnet werden. Weitergehende Informationen zum Thema Taktsynchronität unter SIMATIC S7 finden Sie im

Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Parameter	Einstellung
E/A Adresse	Für die in Easy Motion Control zu nutzende Achse ist die erste Ausgangsadresse im Achs-DB anzugeben. (z. B. Slot 5: A-Adresse 256 bis 265, dann ist die Ausgangsadresse 256 anzugeben)

Hinweis

Die Standardinstallation des TIA-Portals unterstützt nicht die Anbindung des SINAMICS an eine SIMATIC S7-CPU. Prüfen Sie die Verfügbarkeit. Installieren Sie ggf. die entsprechende GSD-Datei für SINAMICS. Erst nach der zusätzlichen Installation werden die notwendigen Daten eingebunden.

Parametrierung des SINAMICS im Achs-DB

Die in der Hardware-Konfiguration für diese Achse eingestellte Ausgangsadresse müssen Sie im Feld "Baugruppenadresse Ausgänge" angeben. Die Telegrammnummer muss erneut angeben werden, da der Ausgangstreiber auch allein betrieben werden könnte. Es wird geprüft, ob die Telegrammnummer mit der vom zugehörigen Eingangstreiber EncoderSINAMICS (falls vorhanden) übereinstimmt.

Einschalten der Achse

Jedem SINAMICS-Motormodul ist zwangsläufig eine zentrale Energieeinspeisung für den Zwischenkreis (ein Line Module) zugeordnet. Vor dem Einschalten der Achse am Eingang "EnableDrive" der Anweisung MC_Control muss sichergestellt werden, dass die zentrale Energieeinspeisung eingeschaltet wurde. Dazu liefert das Line Module entsprechende Rückmeldungen über seinen aktuellen Betriebszustand. Werten Sie die benötigten Rückmeldungen vom Line Module aus und verknüpfen Sie diese als Einschaltbedingung.

Weitergehende Informationen finden Sie in der Dokumentation zu SINAMICS-Antrieben (http://www.automation.siemens.com/doconweb).

Zum Einschalten eines Active Line Moduls finden Sie ein Anwendungsbeispiel im Internet (http://support.automation.siemens.com/WW/view/de/21971603).

Abschalten der Achse - Parametrierung des StopMode

Austrudeln (Coast Stop) (AUS2) – StopMode = 3:

Hier wird die Leistungsfreigabe sofort weggenommen und der Antrieb trudelt stromlos aus. Parallel wird sofort die mechanische Bremsenansteuerung aktiviert. Dabei fällt die Bremse ein und bremst aktiv den Antrieb mit ab. Es kommt je nach anliegender Last zu einer stärkeren mechanischen Belastung der Bremse.

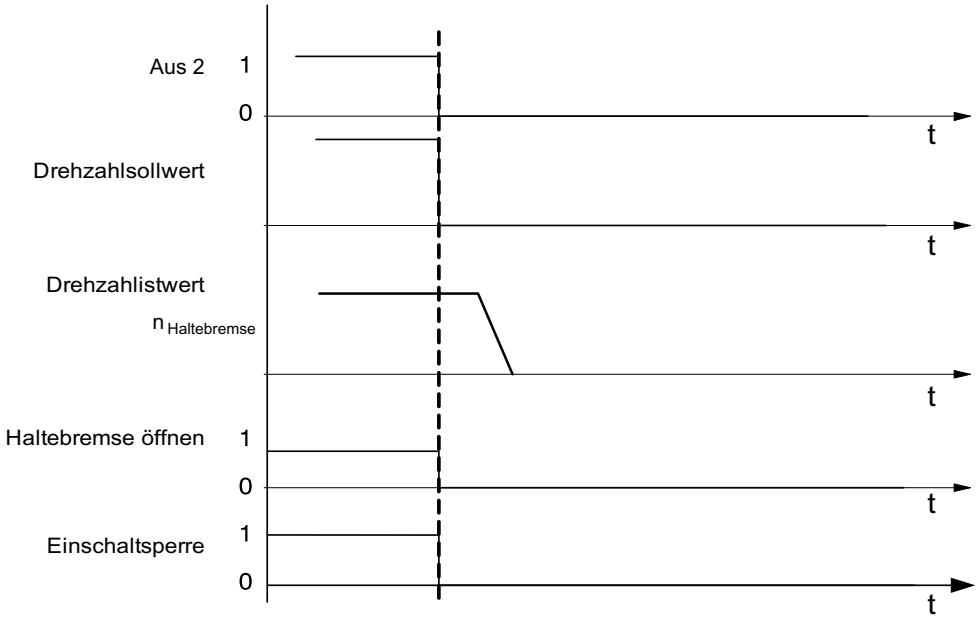

Rampenstopp (AUS1) – StopMode = 5:

Hier wird an der Hochlaufgeber-Rampe des Antriebs abgebremst. Nach Erreichen der Drehzahlschwelle fällt die Haltebremse ein, worauf nach Ablauf der Bremsschließzeit die Impulsfreigabe deaktiviert wird.

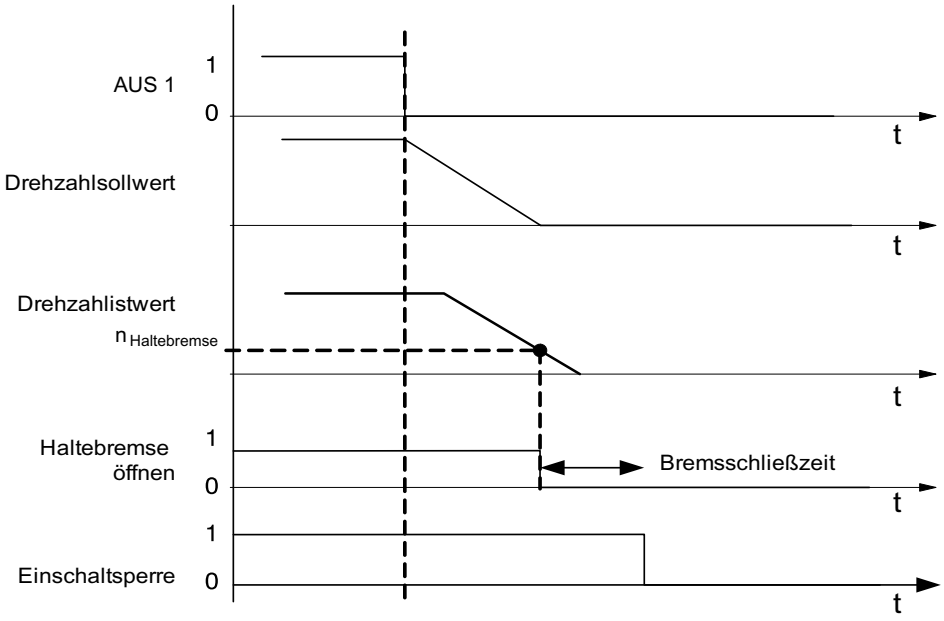

Schnellhalt (AUS3) – StopMode = 6:

Hier wird an der Schnellhalt-Rampe (Notstopp) des Antriebs gebremst. Nach Erreichen der Drehzahlschwelle fällt die Haltebremse ein, worauf nach Ablauf der Bremsschließzeit die Impulsfreigabe deaktiviert wird.

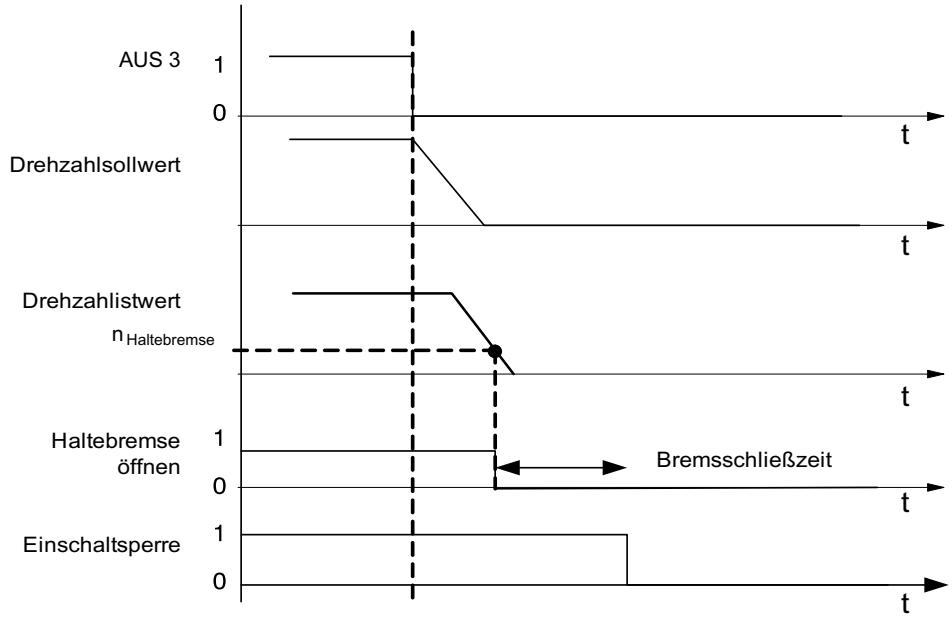

Weiterführende Informationen finden Sie in der Produktinformation SINAMICS S120 (http://support.automation.siemens.com/WW/view/de/30119684).

Aufruf

Die Anweisung wird in einem Weckalarm-OB (z. B. OB 35) oder einem Taktsynchronalarm-OB (z. B. OB 61) aufgerufen. Sie müssen alle EMC-Anweisungen einer Achse im gleichen Programmbaustein aufrufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	
TelNumber	INPUT	INT	Telegrammnummer:	
			Unterstützt werden folgende Standardtelegramme: 3, 5, 102,105	Für PROFINET IO sind nur die Telegramme 3 und 102 freigegeben. Sollten Sie als Eingangstreiber den Encoder-SINAMICS verwenden, muss die Nummer des verwendeten Telegramms zwangsläufig gleich sein.
EnableDrive	INPUT	BOOL	1 = Antriebsfreigabe	Sobald "EnableDrive" = FALSE ist, wird an den Antrieb der Drehzahlwert Null ausgegeben. Die Antriebsfreigabe wird gesperrt. Dieser Eingang kann mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden.
StopMode	INPUT	INT	Art der Abschaltung:	
			3 = Austrudeln (AUS2)	Antrieb trudelt stromlos aus
			5 = Rampenstopp (AUS1)	Stoppen der Achse über Hochlaufgeber
			6 = Schnellhalt (AUS3)	Stoppen der Achse über Schnellhalt-Rampe

Parameter	Deklaration	Datentyp	Beschreibung	
OutErr	INPUT	BOOL	1 = Fehler auf der Ausgabebaugruppe	Ein vom Anwenderprogramm erkannter Fehler (z. B. Baugruppe gezogen oder Stationsausfall) kann dem Ausgangstreiber hier mitgeteilt werden.
Operation_Enabled	OUTPUT	BOOL	1 = SINAMICS meldet "Betrieb freigegeben"	Alle Freigaben für die Achse liegen im Antrieb an und die Achse meldet die Bereitschaft zu fahren. Achse ist aufmagnetisiert.
Err_ID	OUTPUT	WORD	Fehlercode der Anweisung:	
			W#16#0001 = Falscher OB	Anweisung darf nur in OB 6x, OB 3x oder OB 1 installiert werden
			W#16#0002 = Falsche Telegrammnummer	Die angegebene Nummer wird entweder vom Treiber nicht unterstützt oder unterscheidet sich von der im SINAMICS projektierten Telegrammnummer.

OutErr: Mit einer positiven Flanke dieser Variablen wird der Antrieb (SINAMICS) ausgeschaltet. Um die Achse wieder einzuschalten, muss OutErr = FALSE sein und die Fehlermeldung quittiert worden sein.

Operation_Enabled: Diese Variable meldet den aktuellen Zustand der Achse im SINAMICS. Solange Operation_Enabled = FALSE ist, sollten Sie keinen Fahrauftrag aktivieren.

Weitere Informationen zur Anbindung des SINAMICS finden Sie im Internet (http://support.automation.siemens.com/WW/view/de/30119684).

Weitere Referenzen für Easy Motion Control

Achs-Datenbaustein (Achs-DB)

Aufgabe

Der Achs-DB ist die zentrale Datenhaltung von Easy Motion Control. Er enthält:

- Parameter, die die Achse grundlegend beschreiben (z. B. Peripherieadressen)
- Parameter der Lageregelung, die bei der Initialisierung der EMC-Anweisungen aus diesen Grundparametern berechnet werden
- Aktualwerte der Lageregelung (z. B. Position, Fehlerzustände, …)
- Initialisierungs-Bits zur Koordination des Anlaufs der Anweisungen

Inhalt

Die folgende Tabelle zeigt den Inhalt des Achs-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgelistet werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangs-wert	Beschreibung
0.0	Sample_T	REAL	0.01	Abtastzeit der Achs-Anweisungen [s]
4.0	AxisType	BOOL	FALSE	Achsart • FALSE = Linearachse • TRUE = Rundachse
4.1	EncoderType	BOOL	FALSE	Geberart • FALSE = Inkrementalgeber • TRUE = Absolutwertgeber
4.2	Sim	BOOL	FALSE	TRUE = Simulationsmodus ist aktiv
6.0	AxisLimitMax	REAL	$1.0e^{+6}$	Software-Endschalter Ende [Längeneinheit]
10.0	AxisLimitMin	REAL	$-1.0e^{+6}$	Software-Endschalter Anfang [Längeneinheit]
14.0	MaxVelocity	REAL	0.1	Maximale Achsgeschwindigkeit [Längeneinheit/s]
18.0	MaxAcceleration	REAL	0.1	Max. Achsbeschleunigung [Längeneinheit/s^2]
22.0	MaxDeceleration	REAL	0.1	Max. Achsverzögerung [Längeneinheit/s^2]
26.0	MonTimeTargetAppr	REAL	1.0	Überwachungszeit für Zieleinlauf [s]
30.0	TargetRange	REAL	1.0	Zielbereich beim Positionieren [Längeneinheit]
34.0	StandstillRange	REAL	1.5	Stillstandsbereich ohne Verfahrauftrag [Längeneinheit]
38.0	MaxFollowingDist	REAL	5.0	Maximal zulässiger. Schleppabstand [Längeneinheit]
42.0	MonitorTargetAppr	BOOL	TRUE	TRUE = Überwachung des Zielbereichs einschalten
42.1	SWLimitEnable	BOOL	TRUE	TRUE = Überwachung der Software-Endschalter einschalten
44.0	FactorP	REAL	1.0	Proportionalitätsfaktor [1/s]
48.0	ManVelocity	REAL	0.0	Sollgeschwindigkeit im Handbetrieb [Längeneinheit/s]
52.0	ManEnable	BOOL	FALSE	TRUE = Handbetrieb einschalten
54.0	EmergencyDec	REAL	1000.0	Achsverzögerung für Notstopp [Längeneinheit/s^2]
58.0	InputModuleInAddr	INT	0	Anfangsadresse der Eingänge der Wegerfassungsbaugruppe
60.0	InputModuleOutAddr	INT	0	Anfangsadresse der Ausgänge der Wegerfassungsbaugruppe
62.0	InputChannelNo	INT	0	Kanalnummer der Wegerfassungsbaugruppe
64.0	StepsPerRev	DINT	L#1	Schritte pro Geberumdrehung [Schritte/Geberumdrehung]
68.0	DisplacementPerRev	REAL	1.0	Weg pro Geberumdrehung [Längeneinheit/Geberumdrehung]
72.0	NumberRevs	INT	1	Anzahl Geberumdrehungen eines Absolutwertgebers
74.0	PolarityEncoder	INT	1	Richtungsanpassung Geber
76.0	PositionOffset	DINT	L#0	Positionsoffset / Absolutwertgeberjustage [Schritt]
80.0	OutputModuleOutAddr	INT	0	Anfangsadresse der Ausgänge der Ausgabebaugruppe
82.0	OutputModuleInAddr	INT	0	Anfangsadresse der Eingänge der Ausgabebaugruppe
84.0	OutputChannelNo	INT	0	Kanalnummer der Ausgabebaugruppe
86.0	PolarityDrive	INT	1	Richtungsanpassung Antrieb
88.0	DriveInputAtMaxVel	REAL	9.0	Bezugswert für maximale Achsgeschwindigkeit [V]
92.0	OffsetCompensation	REAL	0.0	Nullpunktkompensation [V]

Adresse	Parameter	Datentyp	Anfangs-wert	Beschreibung
96.0	DriveInputAt100	REAL	10.0	Bezugswert für 100 % Drehzahl [V]
100.0	Override	REAL	100.0	Geschwindigkeitsoverride [%]
104.0	ActPosition	REAL	0.0	Aktueller Lageistwert [Längeneinheit]
108.0	FollowingDistance	REAL	0.0	Aktueller Schleppabstand [Längeneinheit]
112.0	RemainingDistance	REAL	0.0	Aktueller Restweg zum Ziel [Längeneinheit]
116.0	NomVelocity	REAL	0.0	Sollgeschwindigkeit der Achse [Längeneinheit/s]
120.0	ActVelocity	REAL	0.0	Istgeschwindigkeit der Achse [Längeneinheit/s]
124.0	Sync	BOOL	FALSE	• FALSE = Achse nicht synchronisiert • TRUE = Achse synchronisiert
124.1	Error	BOOL	TRUE	• FALSE = Kein Fehler • TRUE = Sammelfehler
124.2	ErrorAck	BOOL	FALSE	• FALSE = Keine Quittung • TRUE = Fehler quittieren
128.0	Err.SWLimitMinExceeded	BOOL	FALSE	TRUE = Software-Endschalter Anfang überfahren
128.1	Err.SWLimitMaxExceeded	BOOL	FALSE	TRUE = Software-Endschalter Ende überfahren
128.2	Err.TargetErr	BOOL	FALSE	TRUE = Ziel außerhalb des Verfahrbereichs
128.3	Err.NoSync	BOOL	FALSE	TRUE = Achse nicht synchronisiert
128.4	Err.DirectionErr	BOOL	FALSE	TRUE = Fahrt in vorgegebene Richtung unzulässig
128.5	Err.DataErr	BOOL	FALSE	TRUE = Parameter einer Fahranweisung unzulässig
128.6	Err.StartErr	BOOL	FALSE	TRUE = Start im aktuellen Achszustand nicht möglich
128.7	Err.DistanceErr	BOOL	FALSE	TRUE = Fahrt zu weit
129.0	Err.MasterErr	BOOL	FALSE	TRUE = Leitachse in unzulässigem Zustand
130.0	Err.StoppedMotion	BOOL	TRUE	TRUE = Achse im quittierpflichtigen Stopp-Zustand
130.1	Err.EnableDriveErr	BOOL	FALSE	TRUE = Antriebsfreigabe fehlt
130.2	Err.FollowingDistErr	BOOL	FALSE	TRUE = Max. Schleppabstand überschritten
130.3	Err.StandstillErr	BOOL	FALSE	TRUE = Stillstandsbereich verlassen
130.4	Err.TargetApproachErr	BOOL	FALSE	TRUE = Fehler beim Zieleinlauf
130.5	Err.EncoderErr	BOOL	FALSE	TRUE = Geberfehler
130.6	Err.OutputErr	BOOL	FALSE	TRUE = Fehler am Ausgangstreiber
130.7	Err.ConfigErr	BOOL	FALSE	TRUE = Achsdaten fehlerhaft parametriert
131.0	Err.DriveErr	BOOL	FALSE	TRUE = Antriebsfehler
132.0	Config.Err_AxisLimit	BOOL	FALSE	TRUE = Achsgrenzen fehlerhaft
132.1	Config.Err_MaxVelocity	BOOL	FALSE	TRUE = Maximale Geschwindigkeit fehlerhaft
132.2	Config.Err_MaxAcceleration	BOOL	FALSE	TRUE = Maximale Beschleunigung fehlerhaft
132.3	Config.Err_MaxDeceleration	BOOL	FALSE	TRUE =Maximale Verzögerung fehlerhaft
132.4	Config.Err_MonTimeTarge-tAppr	BOOL	FALSE	TRUE = Überwachungszeit für Zieleinlauf fehlerhaft
132.5	Config.Err_TargetRange	BOOL	FALSE	TRUE = Zielbereich fehlerhaft
132.6	Config.Err_StandstillRange	BOOL	FALSE	TRUE = Stillstandsbereich fehlerhaft
132.7	Config.Err_MaxFollowingDist	BOOL	FALSE	TRUE = Max. Schleppabstand fehlerhaft
133.0	Config.Err_EmergencyDec	BOOL	FALSE	TRUE = Verzögerung für harten Stopp fehlerhaft
133.1	Config.Err_StepsPerRev	BOOL	FALSE	TRUE = Schritte pro Geberumdrehung fehlerhaft

Adresse	Parameter	Datentyp	Anfangs-wert	Beschreibung
133.2	Config.Err_DisplacementPer-Rev	BOOL	FALSE	TRUE = Achsweg pro Geberumdrehung fehlerhaft
133.3	Config.Err_NumberRevs	BOOL	FALSE	TRUE = Anzahl Geberumdrehungen fehlerhaft
133.4	Config.Err_PolarityEncoder	BOOL	FALSE	TRUE = Richtungsanpassung Geber fehlerhaft
133.5	Config.Err_PolarityDrive	BOOL	FALSE	TRUE = Richtungsanpassung Antrieb fehlerhaft
133.6	Config.Err_DriveInputAtMax-Vel	BOOL	FALSE	TRUE = Bezugswert für Maximalgeschwindigkeit fehlerhaft
133.7	Config.Err_AxisLength	BOOL	FALSE	TRUE = Länge der Achse fehlerhaft
134.0	Config.Err_EncoderRange	BOOL	FALSE	TRUE = Bereich des Gebers passt nicht zur Achslänge
134.1	Config.Err_MaxVelRotaryAxis	BOOL	FALSE	TRUE = Maximalgeschwindigkeit und Abtastzeit passen nicht zur Länge der Rundachse
134.2	Config.Err_DriveInputAt100	BOOL	FALSE	TRUE = Bezugswert für 100 % Drehzahl ≤ 0
136.0	EncoderValue	DINT	L#0	Aktueller Geberwert
278.0	Init	STRUCT	TRUE	Bitfeld für Initialisierung von 32 Anweisungen

4.2.4.3 Funktionsmodule

FM x50-1 Zählen

CNT_CTRL

Beschreibung

Mit der Anweisung CNT_CTRL steuern Sie die Zähl- und Messbetriebsarten.

Wenn Sie Parameteränderungen im Betrieb sowie das Setzen und Rücksetzen der Ausgänge der Zählerbaugruppe nutzen wollen, dann müssen Sie die Anweisung CNT_CTL1 (Seite 6114) verwenden.

Arbeitsweise

Die Anweisung CNT_CTRL arbeitet nur in einem nicht taktsynchronen OB.
Verwenden Sie in einem taktsynchronen OB die Anweisung CNT_CTL1 (Seite 6114) oder CNT_CTL2 (Seite 6117).

Die für die Anweisung CNT_CTRL benötigten Daten werden im Zähler-DB (Seite 6118) auf der CPU abgelegt. Die Anweisung CNT_CTRL überträgt zyklisch Daten von diesem DB zur Zählerbaugruppe und holt Daten von der Zählerbaugruppe ab.

Aufruf

Die Anweisung CNT_CTRL kann im Zyklus oder alternativ in einem zeitgesteuerten Programm einmal pro Zähler aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	INT	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
SW_GATE	INPUT	BOOL	Zählersteuerbit SW-Tor (Start/Stopp)	gesetzt und rückgesetzt	abgefragt
GATE_STP	INPUT	BOOL	Zählersteuerbit Torstopp	gesetzt und rückgesetzt	abgefragt
OT_ERR_A	INPUT	BOOL	Bedienfehler quittieren	gesetzt und rückgesetzt	abgefragt
OT_ERR	OUTPUT	BOOL	Bedienfehler aufgetreten	abgefragt	gesetzt und rückgesetzt
L_DIRECT	IN-OUT	BOOL	Anstoßbit für direktes Laden eines Zählers	gesetzt	abgefragt und rückgesetzt
L_PREPAR	IN-OUT	BOOL	Anstoßbit für vorbereitendes Laden eines Zählers	gesetzt	abgefragt und rückgesetzt
T_CMP_V1	IN-OUT	BOOL	Anstoßbit für Vergleichswert 1 übertragen	gesetzt	abgefragt und rückgesetzt
T_CMP_V2	IN-OUT	BOOL	Anstoßbit für Vergleichswert 2 übertragen	gesetzt	abgefragt und rückgesetzt
RES_SYNC	IN-OUT	BOOL	Statusbit Synchronisation löschen	gesetzt	abgefragt und rückgesetzt
RES_ZERO	IN-OUT	BOOL	Statusbit Nulldurchgang löschen	gesetzt	abgefragt und rückgesetzt

Aufträge bearbeiten

Einen Auftrag für die Zählerbaugruppe veranlassen Sie mit den Anweisungsparametern L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, RES_SYNC, RES_ZERO, OT_ERR_A und GATE_STP.

Je nach Auftrag müssen Sie dafür den Ladewert oder einen Vergleichswert vor dem Aufruf der Anweisung in den Zähler-DB eintragen.

Ein gesetzter Durchgangsparameter (L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, RES_SYNC und RES_ZERO) wird von der Anweisung CNT_CTRL nach Erfüllung des Auftrags wieder gelöscht.

Anlaufverhalten

Sobald die Anweisung CNT_CTRL einen Anlauf (CPU- oder FM-Anlauf) erkennt, wird ein anstehender Auftrag zurückgestellt und zuerst der Anlauf quittiert. Ein von Ihnen bereits angestoßener Auftrag wird erst nach beendetem Anlauf ausgeführt und geht nicht verloren.

Verhalten im Fehlerfall

Wenn beim Aufruf der Anweisung ein Bedienfehler aufgetreten ist, wird dies am Parameter OT_ERR gemeldet. Die Fehlerinformation ist dann im Zähler-DB (Variable OT_ERR_B) auszulesen. Mit Hilfe des Parameters OT_ERR_A können Sie den Bedienfehler anschließend quittieren. Es wird kein neuer Bedienfehler gemeldet, solange Sie den vorherigen Fehler nicht quittiert haben.

DIAG_INF

Beschreibung

Die Anweisung DIAG_INF liest bei gesetztem Anstoßparameter (IN_DIAG = TRUE) den Datensatz DS 1 von der Zählerbaugruppe und stellt ihn Ihnen im Zähler-DB (Seite 6118) (ab DW 54) zur Verfügung.

Arbeitsweise

Die Übertragung des Datensatzes erfolgt mit Hilfe der Anweisung RDSYSST. Der Returncode dieser Anweisung (RET_VAL) wird auf den Parameter RET_VAL der Anweisung DIAG_INF kopiert. Sobald die Anweisung DIAG_INF ausgeführt ist, wird der Anstoßparameter IN_DIAG zurückgesetzt und damit die Übertragung als beendet gemeldet.

Aufruf

Die Anweisung DIAG_INF kann im Zyklus und im Alarmprogramm aufgerufen werden. Ein Aufruf im zeitgesteuerten Programm ist jedoch nicht sinnvoll.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	INT	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
RET_VAL	OUTPUT	INT	Returncode der Anweisung RDSYSST	abgefragt	eingetragen
IN_DIAG	IN-OUT	BOOL	Anstoßbit Diagnosedatensatz DS 1 lesen	gesetzt und abgefragt	rückgesetzt

CNT_CTL1

Beschreibung

Mit der Anweisung CNT_CTL1 steuern Sie die Zähl- und Messbetriebsarten.

Zusätzlich zu den Funktionen der Anweisung CNT_CTRL ermöglicht die Anweisung CNT_CTL1 Parameteränderungen im Betrieb sowie das Setzen und Rücksetzen der Ausgänge der Zählerbaugruppe.

Arbeitsweise

Die für die Anweisung CNT_CTL1 benötigten Daten werden im Zähler-DB (Seite 6118) auf der CPU abgelegt. Die Anweisung CNT_CTL1 überträgt zyklisch Daten von diesem DB zur Zählerbaugruppe und holt Daten von der Zählerbaugruppe ab.

Aufruf

Die Anweisung CNT_CTL1 kann im Zyklus oder alternativ in einem zeitgesteuerten oder taktsynchronen Alarm-OB aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	INT	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
SW_GATE	INPUT	BOOL	Zählersteuerbit SW-Tor (Start/Stopp)	gesetzt und rückgesetzt	abgefragt
GATE_STP	INPUT	BOOL	Zählersteuerbit Torstopp	gesetzt und rückgesetzt	abgefragt
OT_ERR_A	INPUT	BOOL	Bedienfehler quittieren	gesetzt und rückgesetzt	abgefragt
SET_DO0	INPUT	BOOL	Setzen/Rücksetzen des Ausgangs DO0	gesetzt und rückgesetzt	abgefragt
SET_DO1	INPUT	BOOL	Setzen/Rücksetzen des Ausgangs DO1	gesetzt und rückgesetzt	abgefragt
OT_ERR	OUTPUT	BOOL	Bedienfehler aufgetreten	abgefragt	gesetzt und rückgesetzt
L_DIRECT 2) 3)	IN-OUT	BOOL	**Zählen:** Anstoßbit für direktes und vorbereitendes Laden eines Zählers	gesetzt	abgefragt und rückgesetzt
			Messen: Darf nicht gesetzt werden!	-	abgefragt und rückgesetzt
L_PREPAR 2) 3)	IN-OUT	BOOL	**Zählen:** Anstoßbit für vorbereitendes Laden eines Zählers	gesetzt	abgefragt und rückgesetzt
			Messen: Übertragen der Untergrenze	gesetzt	abgefragt und rückgesetzt
T_CMP_V1 2)	IN-OUT	BOOL	**Zählen:** Anstoßbit für Vergleichswert 1 übertragen	gesetzt	abgefragt und rückgesetzt
			Messen: Übertragen der Obergrenze	gesetzt	abgefragt und rückgesetzt
T_CMP_V2 2)	IN-OUT	BOOL	**Zählen:** Anstoßbit für Vergleichswert 2 übertragen	gesetzt	abgefragt und rückgesetzt
			Messen: Aktualisierungszeit	gesetzt	abgefragt und rückgesetzt
C_DOPARA 1)	IN-OUT	BOOL	Anstoßbit für Parameteränderung	gesetzt	abgefragt und rückgesetzt

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
RES_SYNC	IN-OUT	BOOL	Statusbit Synchronisation löschen	gesetzt	abgefragt und rückgesetzt
RES_ZERO	IN-OUT	BOOL	Statusbits für Nulldurchgang, Überlauf, Unterlauf und Vergleicher bzw. Messende rücksetzen	gesetzt	abgefragt und rückgesetzt

1) Diesen Parameter dürfen Sie nicht gleichzeitig mit einem der Parameter L_DIRECT, L_PREPAR, T_CMP_V1 oder T_CMP_V2 setzen.

2) Diese Parameter dürfen Sie nicht gleichzeitig mit dem Parameter C_DOPARA setzen.

3) Diese Parameter dürfen Sie nicht gleichzeitig setzen.

Aufträge bearbeiten

Einen Auftrag für die FM x50-1 veranlassen Sie mit den Anweisungsparametern L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA, RES_SYNC, RES_ZERO und OT_ERR_A.

Je nach Auftrag müssen Sie dafür die entsprechenden Werte (Ladewert, Vergleichswerte, Untergrenze, Obergrenze, Aktualisierungszeit) vor dem Aufruf der Anweisung in den Zähler-DB eintragen.

Siehe auch:

- Wertübergabe mit CNT_CTL1 (Seite 6123)
- DB-Parameter für Wertübergaben (Zählbetriebsarten) (Seite 6125)
- DB-Parameter für Wertübergaben (Messbetriebsarten) (Seite 6127)

Ein gesetzter Durchgangsparameter (L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA, RES_SYNC und RES_ZERO) wird von der Anweisung CNT_CTL1 nach Erfüllung des Auftrags wieder gelöscht. Daran können Sie erkennen, dass der Auftrag von der FM x50-1 fertig bearbeitet wurde. Bei Bedarf können Sie diese Information in Ihr Anwenderprogramm einfließen lassen.

Anlaufverhalten

Sobald die Anweisung CNT_CTL1 einen Anlauf (CPU- oder FM-Anlauf) erkennt, wird ein anstehender Auftrag zurückgestellt und zuerst der Anlauf quittiert. Ein von Ihnen bereits angestoßener Auftrag wird erst nach beendetem Anlauf ausgeführt und geht nicht verloren.

Verhalten im Fehlerfall

Wenn beim Aufruf der Anweisung ein Bedienfehler aufgetreten ist, wird dies am Parameter OT_ERR gemeldet. Die Fehlerinformation ist dann im Zähler-DB (Variable OT_ERR_B) auszulesen. Mit Hilfe des Parameters OT_ERR_A können Sie den Bedienfehler anschließend quittieren. Es wird kein neuer Bedienfehler gemeldet, solange Sie den vorherigen Fehler nicht quittiert haben.

CNT_CTL2

Beschreibung

Die Anweisung CNT_CTL2 hat im Wesentlichen die gleiche Funktionalität wie die Anweisung CNT_CTL1 (Seite 6114).

Die Unterschiede zur Anweisung CNT_CTL1 werden im Folgenden erläutert.

Arbeitsweise

Die Anweisung CNT_CTL2 ist besonders geeignet für Anwendungen, bei denen Sie schnell hintereinander denselben Auftrag (z. B. Vergleichswert laden) an die FM x50-1 abgeben wollen. Während Sie mit der Anweisung CNT_CTL1 bestenfalls in jedem fünften PROFIBUS DP-Zyklus einen neuen Auftrag anstoßen können, geht dies mit der Anweisung CNT_CTL2 in jedem zweiten PROFIBUS DP-Zyklus.

Die Anweisung ist für einen Auftrag bereit, wenn das entsprechende Anstoßbit auf 0 steht. Der Abschluss eines Auftrags wird nicht gesondert angezeigt.

Falls ein Kommunikationsproblem oder ein Daten- oder Bedienfehler auftritt, kann dies nicht mehr einem bestimmten Auftrag zugeordnet werden. Deshalb hält die Anweisung in einem solchen Fall die Auftragsbearbeitung an und erzeugt einen quittierbaren Bedienfehler mit der **Kennung 90**. Nachdem Sie den Fehler durch Setzen des Parameters OT_ERR_A quittiert haben, werden eventuell noch anstehende Aufträge (weiter-)bearbeitet.

Die Quittung eines Bedienfehlers ist erfolgreich angenommen, wenn der Parameter OT_ERR rückgesetzt ist. Zur sicheren Quittung sollten Sie den Parameter OT_ERR_A solange gesetzt lassen. Erst nach einer erfolgreichen Quittung ist es sinnvoll, weitere Aufträge anzustoßen.

Hinweis

Sie dürfen mit der Anweisung CNT_CTL2 nicht mehrere Wertübergaben gleichzeitig starten.

Aufruf

Die Anweisung CNT_CTL2 arbeitet nur in einem taktsynchronen OB.

Wenn Sie die Anweisung CNT_CTL2 in einem nicht taktsynchronen OB aufrufen, erzeugt sie einen Bedienfehler mit der **Kennung 91**. Ein Datenaustausch mit der FM x50-1 findet dann nicht statt.

Weitere Referenzen für FM x50-1 Counter V3

Zähler-Datenbaustein

Aufgabe

Der Zähler-DB ist die Datenschnittstelle zwischen dem Anwenderprogramm und der FM x50-1. Er enthält und übernimmt alle Daten, die zur Steuerung und zum Betrieb der Baugruppe notwendig sind.

Vor der Parametrierung der Baugruppe müssen Sie den Zähler-DB mit folgenden gültigen Daten belegen.

- Baugruppenadresse (Adresse 6.0)
- Kanalanfangsadresse (Adresse 8.0)
- Länge der Nutzdatenschnittstelle (Adresse 12.0)

Im Gerätehandbuch der Zählerbaugruppe finden Sie ein Beispiel, wie Sie im OB 100 die Übertragung der Baugruppenadresse, der Kanalanfangsadresse und der Länge der Nutzdatenschnittstelle in den Zähler-DB realisieren können.

Alternativ können Sie die Baugruppenadresse auch automatisch eintragen lassen über die Funktion "Moduladresse in den DB eintragen" in der Gerätekonfiguration der FM x50-1.

Aufbau

Der Zähler-DB ist in verschiedene Bereiche unterteilt:

Kanal-DB
Anweisungsparameter, Adressen
Übergabebereich für zu schreibende Werte
Steuerschnittstelle
Übergabebereich für zu lesende Werte
Fehlernummern
Rückmeldeschnittstelle
Parameter für FM 450-1
Diagnosedaten

Inhalt

Die folgende Tabelle zeigt den Inhalt des Zähler-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgelistet werden, nicht verändern.

Adresse	Variable	Datentyp	Anfangswert	Kommentar	
				Zählen FM 350-1 / FM 450-1	Messen FM 350-1
Anweisungsparameter, Adressen					
0.0	AR1_BUFFER	DWORD	DW#16#0	AR1 Puffer	AR1 Puffer
4.0	FP	BYTE	B#16#0	Flag Byte	Flag Byte
5.0	RESERVED	BYTE	B#16#0	Reserviert	Reserviert
6.0	MOD_ADR	WORD	W#16#0	Baugruppenadresse	Baugruppenadresse
8.0	CH_ADR	DWORD	DW#16#0	Kanaladresse	Kanaladresse
12.0	U_D_LGTH	BYTE	B#16#0	Länge Nutzdaten	Länge Nutzdaten
13.0	A_BYTE_0	BYTE	B#16#0	Reserviert	Reserviert
Übergabebereich für zu schreibenden Werte					
14.0	LOAD_VAL [1)]	DINT	L#0	Neuer Ladewert (write user)	Untergrenze (write user)
18.0	CMP_V1 [1)]	DINT	L#0	Neuer Vergleichswert 1 (write user)	Obergrenze (write user)
22.0	CMP_V2 [1)]	DINT	L#0	Neuer Vergleichswert 2 (write user)	Aktualisierungszeit (write user)
Steuerschnittstelle					
26.0	A_BIT0_0	BOOL	FALSE	Reserviert	Reserviert
26.1	TFB	BOOL	FALSE	Reserviert / Test frei	Reserviert
26.2	A_BIT0_2	BOOL	FALSE	Reserviert	Reserviert
26.3	A_BIT0_3	BOOL	FALSE	Reserviert	Reserviert
26.4	A_BIT0_4	BOOL	FALSE	Reserviert	Reserviert
26.5	A_BIT0_5	BOOL	FALSE	Reserviert	Reserviert
26.6	A_BIT0_6	BOOL	FALSE	Reserviert	Reserviert
26.7	A_BIT0_7	BOOL	FALSE	Reserviert	Reserviert
27.0	ENSET_UP [1)]	BOOL	FALSE	Freigabe für Setzen in Vorwärtsrichtung (write user)	-
27.1	ENSET_DN [1)]	BOOL	FALSE	Freigabe für Setzen in Rückwärtsrichtung (write user)	-
27.2	A_BIT1_2	BOOL	FALSE	Reserviert	Reserviert
27.3	A_BIT1_3	BOOL	FALSE	Reserviert	Reserviert
27.4	A_BIT1_4	BOOL	FALSE	Reserviert	Reserviert
27.5	A_BIT1_5	BOOL	FALSE	Reserviert	Reserviert
27.6	A_BIT1_6	BOOL	FALSE	Reserviert	Reserviert
27.7	A_BIT1_7	BOOL	FALSE	Reserviert	Reserviert
28.0	CTRL_DO0 [1)]	BOOL	FALSE	Freigabe Digitalausgang DO0 (write user)	Freigabe Digitalausgang DO0 (write user)
28.1	CTRL_DO1 [1)]	BOOL	FALSE	Freigabe Digitalausgang DO1 (write user)	Freigabe Digitalausgang DO1 (write user)
28.2	A_BIT2_2	BOOL	FALSE	Reserviert	Reserviert

Adresse	Variable	Datentyp	Anfangswert	Kommentar	
				Zählen FM 350-1 / FM 450-1	Messen FM 350-1
28.3	A_BIT2_3	BOOL	FALSE	Reserviert	Reserviert
28.4	A_BIT2_4	BOOL	FALSE	Reserviert	Reserviert
28.5	A_BIT2_5	BOOL	FALSE	Reserviert	Reserviert
28.6	A_BIT2_6	BOOL	FALSE	Reserviert	Reserviert
28.7	A_BIT2_7	BOOL	FALSE	Reserviert	Reserviert
29.0	A_BIT3_0	BOOL	FALSE	Reserviert	Reserviert
29.1	A_BIT3_1	BOOL	FALSE	Reserviert	Reserviert
29.2	A_BIT3_2	BOOL	FALSE	Reserviert	Reserviert
29.3	A_BIT3_3	BOOL	FALSE	Reserviert	Reserviert
29.4	A_BIT3_4	BOOL	FALSE	Reserviert	Reserviert
29.5	A_BIT3_5	BOOL	FALSE	Reserviert	Reserviert
29.6	A_BIT3_6	BOOL	FALSE	Reserviert	Reserviert
29.7	A_BIT3_7	BOOL	FALSE	Reserviert	Reserviert
Übergabebereich für zu lesende Werte					
30.0	LATCH_LOAD [1)]	DINT	L#0	Aktueller Lade- oder Latch-wert (read user)	Aktueller Messwert (read user)
34.0	ACT_CNTV [1)]	DINT	L#0	Aktueller Zählwert (read user)	Aktueller Zählwert (read user)
Fehlernummern					
38.0	DA_ERR_W [1)]	WORD	W#16#0	Datenfehler-Wort (read user)	Datenfehler-Wort (read user)
40.0	OT_ERR_B [1)]	BYTE	B#16#0	Bedienfehler-Byte (read user)	Bedienfehler-Byte (read user)
Rückmeldeschnittstelle					
41.0	E_BIT0_0	BOOL	FALSE	Reserviert	Reserviert
41.1	STS_TFB	BOOL	FALSE	Reserviert / Status Test frei	Reserviert
41.2	DIAG	BOOL	FALSE	FM 350-1: Diagnoseereignis (read user) FM 450-1: Reserviert	Diagnoseereignis (read user)
41.3	E_BIT0_3	BOOL	FALSE	Reserviert	Reserviert
41.4	DATA_ERR [1)]	BOOL	FALSE	Datenfehler-Bit (read user)	Datenfehler-Bit (read user)
41.5	E_BIT0_5	BOOL	FALSE	Reserviert	Reserviert
41.6	E_BIT0_6	BOOL	FALSE	Reserviert	Reserviert
41.7	PARA [1)]	BOOL	FALSE	Baugruppe parametriert (read user)	Baugruppe parametriert (read user)
42.0	E_BYTE_0	BYTE	B#16#0	Reserviert	Reserviert
43.0	STS_RUN [1)]	BOOL	FALSE	Status Zähler arbeitet (read user) Dieses Bit entspricht dem Bit 2^0 des Zählers. 0 = LED CR ist aus 1 = LED CR leuchtet	Status Zähler arbeitet (read user) Dieses Bit entspricht dem Bit 2^0 des Zählers. 0 = LED CR ist aus 1 = LED CR leuchtet

Adresse	Variable	Datentyp	Anfangswert	Kommentar	
				Zählen FM 350-1 / FM 450-1	Messen FM 350-1
43.1	STS_DIR [1]	BOOL	FALSE	Status Zählrichtung (read user)	Status Zählrichtung (read user)
43.2	STS_ZERO [1]	BOOL	FALSE	Status Nulldurchgang (read user)	Messende (read user)
43.3	STS_OFLW [1]	BOOL	FALSE	Status Überlauf (read user)	Status Überlauf (read user)
43.4	STS_UFLW [1]	BOOL	FALSE	Status Unterlauf (read user)	Status Unterlauf (read user)
43.5	STS_SYNC [1]	BOOL	FALSE	Status Zähler synchronisiert (read user)	-
43.6	STS_GATE [1]	BOOL	FALSE	Status internes Tor (read user)	Status internes Tor (read user)
43.7	STS_SW_G [1]	BOOL	FALSE	Status SW-Tor (read user)	Status SW-Tor (read user)
44.0	STS_SET [1]	BOOL	FALSE	Status Digitaleingang SET (read user)	Status Digitaleingang DI-Set (read user)
44.1	STS_LATCH [1]	BOOL	FALSE	Neuer Latchwert (nur im taktsynchronen Betrieb) / Reserviert	-
44.2	STS_STA [1]	BOOL	FALSE	Status Digitaleingang START (read user)	Status Digitaleingang DI-Start (read user)
44.3	STS_STP [1]	BOOL	FALSE	Status Digitaleingang STOP (read user)	Status Digitaleingang DI-Stop (read user)
44.4	STS_CMP1 [1]	BOOL	FALSE	Status Ausgang Vergleichswert 1 (read user)	Status Ausgang Vergleichswert 1 (read user)
44.5	STS_CMP2 [1]	BOOL	FALSE	Status Ausgang Vergleichswert 2 (read user)	Status Ausgang Vergleichswert 2 (read user)
44.6	STS_COMP1 [1]	BOOL	FALSE	Gespeicherter Status des Vergleichers 1 / Reserviert	-
44.7	STS_COMP2 [1]	BOOL	FALSE	Gespeicherter Status des Vergleichers 2 / Reserviert	-
45.0	E_BIT3_0	BOOL	FALSE	Reserviert	Reserviert
45.1	E_BIT3_1	BOOL	FALSE	Reserviert	Reserviert
45.2	E_BIT3_2	BOOL	FALSE	Reserviert	Reserviert
45.3	E_BIT3_3	BOOL	FALSE	Reserviert	Reserviert
45.4	E_BIT3_4	BOOL	FALSE	Reserviert	Reserviert
45.5	E_BIT3_5	BOOL	FALSE	Reserviert	Reserviert
45.6	E_BIT3_6	BOOL	FALSE	Reserviert	Reserviert
45.7	E_BIT3_7	BOOL	FALSE	Reserviert	Reserviert
Parameter für FM 450-1					
46.0	ACT_CMP1 [1]	DINT	L#0	Reserviert / Aktueller Vergleichswert 1 (read user)	Reserviert
50.0	ACT_CMP2 [1]	DINT	L#0	Reserviert / Aktueller Vergleichswert 2 (read user)	Reserviert

Adresse	Variable	Datentyp	Anfangswert	Kommentar	
				Zählen FM 350-1 / FM 450-1	Messen FM 350-1
Die nachfolgenden Diagnosedaten werden von der Anweisung DIAG_INF eingetragen					
54.0	MDL_DEFECT	BOOL	FALSE	Baugruppenfehler	Baugruppenfehler
54.1	INT_FAULT	BOOL	FALSE	Interner Fehler	Interner Fehler
54.2	EXT_FAULT	BOOL	FALSE	Externer Fehler	Externer Fehler
54.3	PNT_INFO	BOOL	FALSE	Kanalfehler (Aufschlüsselung ab DW 58)	Kanalfehler (Aufschlüsselung ab DW 58)
54.4	EXT_VOLTAGE	BOOL	FALSE	Fehler Hilfsspannung	Fehler Hilfsspannung
54.5	FLD_CNNCTR	BOOL	FALSE	Frontstecker	Frontstecker
54.6	NO_CONFIG	BOOL	FALSE	Parametrierung fehlt	Parametrierung fehlt
54.7	CONFIG_ERR	BOOL	FALSE	Parametrierung fehlerhaft	Parametrierung fehlerhaft
55.0	MDL_TYPE	BYTE	B#16#0	Baugruppentyp	Baugruppentyp
56.0	SUB_MDL_ERR	BOOL	FALSE	Falsches/fehlendes Interface-Modul	Falsches/fehlendes Interface-Modul
56.1	COMM_FAULT	BOOL	FALSE	Kommunikationsfehler	Kommunikationsfehler
56.2	MDL_STOP	BOOL	FALSE	RUN/STOP Betriebszustands-Anzeige	RUN/STOP Betriebszustands-Anzeige
56.3	WTCH_DOG_FAULT	BOOL	FALSE	Watchdog (FM)	Watchdog (FM)
56.4	INT_PS_FLT	BOOL	FALSE	Fehler interne Stromversorgung	Fehler interne Stromversorgung
56.5	PRIM_BATT_FLT	BOOL	FALSE	Batterie-Überwachung	Batterie-Überwachung
56.6	BCKUP_BATT_FLT	BOOL	FALSE	Fehlerhafte Pufferung	Fehlerhafte Pufferung
56.7	RESERVED_2	BOOL	FALSE	Reserviert	Reserviert
57.0	RACK_FLT	BOOL	FALSE	Fehler Baugruppenträger	Fehler Baugruppenträger
57.1	PROC_FLT	BOOL	FALSE	Fehler CPU	Fehler CPU
57.2	EPROM_FLT	BOOL	FALSE	Fehler EPROM	Fehler EPROM
57.3	RAM_FLT	BOOL	FALSE	Fehler RAM	Fehler RAM
57.4	ADU_FLT	BOOL	FALSE	Fehler ADU	Fehler ADU
57.5	FUSE_FLT	BOOL	FALSE	Sicherung	Sicherung
57.6	HW_INTR_FLT	BOOL	FALSE	Prozessalarm verloren	Prozessalarm verloren
57.7	RESERVED_3	BOOL	FALSE	Reserviert	Reserviert
58.0	CH_TYPE	BYTE	B#16#0	Kanaltyp	Kanaltyp
59.0	LGTH_DIA	BYTE	B#16#0	Länge der Diagnosedaten pro Kanal	Länge der Diagnosedaten pro Kanal
60.0	CH_NO	BYTE	B#16#0	Kanalanzahl	Kanalanzahl
61.0	GRP_ERR1	BOOL	FALSE	Gruppenfehler Kanal 1	Gruppenfehler Kanal 1
61.1	GRP_ERR2	BOOL	FALSE	Reserviert / Gruppenfehler Kanal 2	Bei FM 350 nicht belegt
61.2	D_BIT7_2	BOOL	FALSE	DS1 byte 7 bit 2	DS1 byte 7 bit 2
61.3	D_BIT7_3	BOOL	FALSE	DS1 byte 7 bit 3	DS1 byte 7 bit 3
61.4	D_BIT7_4	BOOL	FALSE	DS1 byte 7 bit 4	DS1 byte 7 bit 4
61.5	D_BIT7_5	BOOL	FALSE	DS1 byte 7 bit 5	DS1 byte 7 bit 5

Adresse	Variable	Datentyp	Anfangswert	Kommentar	
				Zählen FM 350-1 / FM 450-1	Messen FM 350-1
61.6	D_BIT7_6	BOOL	FALSE	DS1 byte 7 bit 6	DS1 byte 7 bit 6
61.7	D_BIT7_7	BOOL	FALSE	DS1 byte 7 bit 7	DS1 byte 7 bit 7
62.0	CH1_SIGA	BOOL	FALSE	Kanal 1, Fehler Signal A	Kanal 1, Fehler Signal A
62.1	CH1_SIGB	BOOL	FALSE	Kanal 1, Fehler Signal B	Kanal 1, Fehler Signal B
62.2	CH1_SIGZ	BOOL	FALSE	Kanal 1, Fehler Nullsignal	Kanal 1, Fehler Nullsignal
62.3	CH1_BETW	BOOL	FALSE	Kanal 1, Fehler zwischen Kanälen	Kanal 1, Fehler zwischen Kanälen
62.4	CH1_5V2	BOOL	FALSE	Kanal 1, Fehler Geberversorgung 5.2V	Kanal 1, Fehler Geberversorgung 5.2V
62.5	D_BIT8_5	BOOL	FALSE	DS1 byte 8 bit 5	DS1 byte 8 bit 5
62.6	D_BIT8_6	BOOL	FALSE	DS1 byte 8 bit 6	DS1 byte 8 bit 6
62.7	D_BIT8_7	BOOL	FALSE	DS1 byte 8 bit 7	DS1 byte 8 bit 7
63.0	D_BYTE9	BYTE	B#16#0	DS1 byte 9	DS1 byte 9
64.0	CH2_SIGA	BOOL	FALSE	Reserviert / Kanal 2, Fehler Signal A	Reserviert
64.1	CH2_SIGB	BOOL	FALSE	Reserviert / Kanal 2, Fehler Signal B	Reserviert
64.2	CH2_SIGZ	BOOL	FALSE	Reserviert / Kanal 2, Fehler Nullsignal	Reserviert
64.3	CH2_BETW	BOOL	FALSE	Reserviert / Kanal 2, Fehler zwischen Kanälen	Reserviert
64.4	CH2_5V2	BOOL	FALSE	Reserviert / Kanal 2, Fehler Geberversorgung 5.2V	Reserviert
64.5	D_BIT10_5	BOOL	FALSE	DS1 byte 10 bit 5	DS1 byte 10 bit 5
64.6	D_BIT10_6	BOOL	FALSE	DS1 byte 10 bit 6	DS1 byte 10 bit 6
64.7	D_BIT10_7	BOOL	FALSE	DS1 byte 10 bit 7	DS1 byte 10 bit 7
65.0	D_BYTE11	BYTE	B#16#0	DS1 byte 11	DS1 byte 11
66.0	D_BYTE12	BYTE	B#16#0	DS1 byte 12	DS1 byte 12
67.0	D_BYTE13	BYTE	B#16#0	DS1 byte 13	DS1 byte 13
68.0	D_BYTE14	BYTE	B#16#0	DS1 byte 14	DS1 byte 14
69.0	D_BYTE15	BYTE	B#16#0	DS1 byte 15	DS1 byte 15
[1] Variablen im DB, die Sie während der Arbeit mit der FM x50-1 eintragen oder auslesen müssen/können					

Wertübergabe mit CNT_CTL1

Wertübergabe

Werte übergeben Sie je nach Betriebsart durch Setzen dieser Anweisungsparameter:

Betriebsart	Anweisungsparameter
Zählen	L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA
Messen	L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA

Die zu übergebenden Werte müssen Sie zuvor im DB ablegen.

Siehe auch:

- DB-Parameter für Wertübergaben (Zählbetriebsarten) (Seite 6125)
- DB-Parameter für Wertübergaben (Messbetriebsarten) (Seite 6127)

Sie können mehrere Werte gleichzeitig übertragen:

Bei der Betriebsart ...	... können sie gleichzeitig übertragen	
Zählen	Ladewert	(DB-Parameter LOAD_VAL)
	Vergleichswert 1	(DB-Parameter CMP_V1)
	Vergleichswert 2	(DB-Parameter CMP_V2)
Messen	Untergrenze	(DB-Parameter LOAD_VAL)
	Obergrenze	(DB-Parameter CMP_V1)
	Aktualisierungszeit	(DB-Parameter CMP_V2)

Wenn dabei ein Wert fehlerhaft ist, müssen Sie zuerst diesen Bedienfehler mit OT_ERR_A quittieren, bevor die FM 350-1 die weiteren Werte übernehmen kann. Anschließend korrigieren Sie den mit Bedienfehler abgewiesenen Wert und übertragen ihn erneut.

Hinweis

Wenn Sie mit den Anweisungsparametern L_DIRECT, L_PREPAR, T_CMP_V1 oder T_CMP_V2 die Werte LOAD_VAL, CMP_V1 oder CMP_V2 laden, dürfen Sie nicht gleichzeitig mit dem Anweisungsparameter C_DOPARA Umparametrierungen vornehmen.

Dies würde zu einem Bedienfehler OT_ERR führen, den Sie mit OT_ERR_A quittieren müssen.

Zeitbedarf für Wertübergaben

Den Zeitbedarf für Wertübergaben entnehmen Sie folgender Tabelle:

Einsatz der FM 350-1		Zeitbedarf
Zentral		mindestens 4 OB 1-Zyklen
Dezentral (Nicht taktsynchroner Betrieb)		mindestens 5 PROFIBUS DP-Zyklen
Dezentral (Taktsynchroner Betrieb)		
• Bei Übertragung von nur einem Wert		• 5 PROFIBUS DP-Zyklen
• Bei gleichzeitigem Anstoß der Übertragung von mehreren Werten		
	• für den 1. Wert: • für den 2. Wert: • für den 3. Wert:	• 5 PROFIBUS DP-Zyklen nach dem Anstoß • 6 PROFIBUS DP-Zyklen nach dem Anstoß • 7 PROFIBUS DP-Zyklen nach dem Anstoß

DB-Parameter für Wertübergaben (Zählbetriebsarten)

Vorbereitung

Vor dem Aufruf der Anweisungen CNT_CTRL (Seite 6112), CNT_CTL1 (Seite 6114) bzw. CNT_CTL2 (Seite 6117) müssen Sie den Datenbaustein mit folgenden gültigen Daten belegen:

- Baugruppenadresse (DB-Adresse 6.0)
- Kanalanfangsadresse (DB-Adresse 8.0)
- Länge der Nutzdatenschnittstelle (DB-Adresse 12.0)

Diese Daten sollten im OB 100 bei jedem Anlauf in den DB eingetragen werden.

DB-Parameter für Wertübergaben (Zählbetriebsarten)

Nachfolgende Tabelle zeigt den Bereich des DB, in dem Sie die Parameter LOAD_VAL, CMP_V1 und CMP_V2 übergeben.

Der Parameter LOAD_VAL (Byte 14 bis 17) hat zweierlei Bedeutung:

- Wenn Sie den Anweisungsparameter L_DIRECT oder L_PREPAR setzen, wird LOAD_VAL als Ladewert interpretiert.
- Wenn Sie den Anweisungsparameter C_DOPARA setzen, können Sie im Byte 14 das Verhalten der Ausgänge DO0 und DO1 festlegen, Byte 15 und 16 werden als Hysterese und Impulsdauer interpretiert.

Von den Anweisungsparametern L_DIRECT, L_PREPAR und C_DOPARA dürfen Sie immer nur einen setzen.

DB-Adresse	Parameter	Datentyp	Bedeutung				
14.0	LO-AD_VAL	DINT	Ladewert; direkt und vorbereitend laden mit Anweisungsparameter L_DIRECT Ladewert; vorbereitend laden mit Anweisungsparameter L_PREPAR				
			Verhalten der Ausgänge DO0 und DO1, der Hysterese und der Impulsdauer; festlegen mit Anweisungsparameter C_DOPARA:				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Verhalten von Ausgang DO0**
			x	0	0	0	Inaktiv
			x	0	0	1	Aktiv von Vergleichswert bis Obergrenze
			x	0	1	0	Aktiv von Vergleichswert bis Untergrenze
			x	0	1	1	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer vorwärts/rückwärts
			x	1	0	0	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer vorwärts
			x	1	0	1	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer rückwärts
			Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Verhalten von Ausgang DO1**
			x	0	0	0	Inaktiv
			x	0	0	1	Aktiv von Vergleichswert bis Obergrenze
			x	0	1	0	Aktiv von Vergleichswert bis Untergrenze
			x	0	1	1	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer vorwärts/rückwärts
			x	1	0	0	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer vorwärts
			x	1	0	1	Aktiv bei Erreichen des Vergleichswerts für Impulsdauer rückwärts
			x	1	1	0	Schalten an Vergleichswerten
15.0			Hysterese (Wertebereich 0...255)				
16.0			Impulsdauer (Wertebereich 0..250)				
17.0			Reserve = 0				
18.0	CMP_V1	DINT	Vergleichswert 1; laden mit Anweisungsparameter T_CMP_V1				
22.0	CMP_V2	DINT	Vergleichswert 2; laden mit Anweisungsparameter T_CMP_V2				
x = irrelevant							

DB-Parameter für Wertübergaben (Messbetriebsarten)

Vorbereitung

Vor dem Aufruf der Anweisungen CNT_CTRL (Seite 6112), CNT_CTL1 (Seite 6114) bzw. CNT_CTL2 (Seite 6117) müssen Sie den Datenbaustein mit folgenden gültigen Daten belegen:

- Baugruppenadresse (DB-Adresse 6.0)
- Kanalanfangsadresse (DB-Adresse 8.0)
- Länge der Nutzdatenschnittstelle (DB-Adresse 12.0)

Diese Daten sollten im OB 100 bei jedem Anlauf in den DB eingetragen werden.

DB-Parameter für Wertübergaben (Messbetriebsarten)

Nachfolgende Tabelle zeigt den Bereich des DB, in dem Sie die Parameter LOAD_VAL, CMP_V1 und CMP_V2 übergeben.

Der Parameter LOAD_VAL (Byte 14 bis 17) hat zweierlei Bedeutung:

- Wenn Sie den Anweisungsparameter L_PREPAR setzen, wird LOAD_VAL als unterer Grenzwert interpretiert.
- Wenn Sie den Anweisungsparameter C_DOPARA setzen, können Sie im Byte 14 das Verhalten des Ausgangs DO0 festlegen.

Von den Anweisungsparametern L_PREPAR und C_DOPARA dürfen Sie immer nur einen setzen.

DB-Adresse	Parameter	Datentyp	Bedeutung			
14.0	LOAD_VAL	DINT	Unterer Grenzwert; laden mit Anweisungsparameter L_PREPAR			
			Verhalten von DO0; festlegen mit Anweisungsparameter C_DOPARA			
			Bit 2 bis 7	**Bit 1**	**Bit 0**	**Verhalten von Ausgang DO0**
			irrelevant	0	0	Kein Vergleich
			irrelevant	0	1	Aktiv außerhalb der Grenzwerte
			irrelevant	1	0	Aktiv bei Unterschreiten des unteren Grenzwertes
			irrelevant	1	1	Aktiv bei Überschreiten des oberen Grenzwertes
15.0			Reserve = 0			
16.0			Reserve = 0			
17.0			Reserve = 0			
18.0	CMP_V1	DINT	Oberer Grenzwert; laden mit Anweisungsparameter T_CMP_V1			
22.0	CMP_V2	DINT	Aktualisierungszeit; laden mit Anweisungsparameter T_CMP_V2			

FM 350-2 Zählen

CNT2WRPN

Beschreibung

Mit der Anweisung CNT2WRPN laden Sie die Zähler und Vergleicher der Zählerbaugruppe mittels Schreibaufträgen. Dazu müssen Sie die Anweisung CNT2WRPN bei Bedarf pro Baugruppe aufrufen.

Binden Sie die Anweisung CNT2WRPN nur dann in Ihr Anwenderprogramm ein, wenn Sie die Zähler und Vergleicher der Zählerbaugruppe im Betrieb neu laden müssen.

Arbeitsweise

Die Anweisung CNT2WRPN führt den Schreibauftrag (Seite 6138) (JOB_WR) aus dem Zähler-DB (Seite 6130) aus, überträgt die zugehörigen Daten aus dem Zähler-DB und zeigt den Status des Schreibauftrags (Seite 6140) an. Für den Schreibauftrag wird intern die Anweisung WRREC verwendet und der Returncode der Anweisung WRREC durchgereicht.

Aufruf

Die Anweisung CNT2WRPN kann im Zyklus oder alternativ in einem zeitgesteuerten Programm aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Vor der Bearbeitung von Schreibaufträgen müssen Sie den zum Schreibauftrag gehörenden Datenbereich mit den entsprechenden Werten versorgen. Der letzte Schreibauftrag muss abgearbeitet sein, d. h. JOB_WR.NO (Datenbyte DBB0) im Zähler-DB (Seite 6130) ist gelöscht.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	WORD	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
RET_VAL	OUTPUT	INT	Returncode der Anweisung WRREC	abgefragt	eingetragen

CNT2RDPN

Beschreibung

Mit der Anweisung CNT2RDPN lesen Sie Zähl- und Messwerte von der Zählerbaugruppe mit Leseaufträgen. Dazu rufen Sie die Anweisung CNT2RDPN zyklisch einmal pro Baugruppe auf.

Binden Sie die Anweisung CNT2RDPN nicht in Ihr Anwenderprogramm ein, wenn Sie keine Leseaufträge bearbeiten.

Arbeitsweise

Die Anweisung führt den Leseauftrag (Seite 6141) (JOB_RD) aus dem Zähler-DB (Seite 6130) aus, überträgt die zugehörigen Daten in den Zähler-DB und zeigt den Status des Leseauftrags (Seite 6141) an. Für den Leseauftrag wird intern die Anweisung RDREC verwendet und der Returncode der Anweisung RDREC durchgereicht.

Aufruf

Die Anweisung CNT2RDPN kann im Zyklus oder alternativ in einem zeitgesteuerten Programm aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Der letzte Leseauftrag muss abgearbeitet sein, d. h. JOB_RD.NO (Datenbyte DBB2) im Zähler-DB (Seite 6130) ist gelöscht.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	WORD	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
RET_VAL	OUTPUT	INT	Returncode der Anweisung RDREC	abgefragt	eingetragen

CNT2_CTR

Beschreibung

Mit der Anweisung CNT2_CTR steuern Sie die Digitalausgänge (freigeben und sperren) und die Software-Tore der Zählerbaugruppe. Außerdem erhalten Sie die Rückmeldungen der Baugruppe.

Arbeitsweise

Die Anweisung CNT2_CTR

- initialisiert den Zähler-DB (Seite 6130).
- überträgt die Steuersignale aus dem Zähler-DB (Struktur CONTROL_SIGNALS (Seite 6137)) zur Baugruppe.
- liest die Rückmeldesignale. Die gelesenen Werte werden von der Anweisung im Zähler-DB in der Struktur CHECKBACK_SIGNALS (Seite 6138) abgelegt.

Aufruf

Sie müssen die Anweisung CNT2_CTR zyklisch (im OB 1 oder in den Weckalarmen) für jede Baugruppe aufrufen. Der Aufruf in einem Alarmprogramm ist nicht zulässig.

Vor dem Aufruf der Anweisung CNT2_CTR tragen Sie die aktuellen Steuersignale in der Struktur CONTROL_SIGNALS im Zähler-DB (Seite 6130) ein. Nach dem Aufruf der Anweisung

CNT2_CTR sind die Rückmeldesignale in der Struktur CHECKBACK_SIGNALS im Zähler-DB aktualisiert und Sie können diese von dort weiterverarbeiten.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	WORD	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt

DIAG_RD

Beschreibung

Mit der Anweisung DIAG_RD können Sie im Falle eines Diagnosealarms die Diagnosealarmdaten in den Zähler-DB laden.

Arbeitsweise

Die Anweisung DIAG_RD liest 16 Byte Diagnosedaten von der Zählerbaugruppe und trägt diese Daten im Zähler-DB (Seite 6130) in den Datenbereich DIAGNOSTIC_IN_INFO ein. Diese Daten beinhalten den Diagnosezustand der gesamten Baugruppe (über alle Zähler hinweg). Das Lesen der Diagnosedaten erfolgt intern mit der Anweisung RDSYSST. Der Returncode dieser Anweisung wird als RET_VAL durchgereicht.

Aufruf

Die Anweisung DIAG_RD kann nur im Alarmprogramm aufgerufen werden.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung	Wird vom Anwender …	Wird von der Anweisung …
DB_NO	INPUT	WORD	Nummer des Zähler-Datenbausteins	eingetragen	abgefragt
RET_VAL	OUTPUT	INT	Returncode der Anweisung RDSYSST	abgefragt	eingetragen

Weitere Referenzen für FM 350-2 Counter V2

Zähler-Datenbaustein

Aufgabe

Der Zähler-DB ist die Datenschnittstelle zwischen dem Anwenderprogramm und der FM 350-2. Er enthält und übernimmt alle Daten, die zur Steuerung und zum Betrieb der Baugruppe notwendig sind.

Vor der Parametrierung der Baugruppe müssen Sie den Zähler-DB mit folgenden gültigen Daten belegen.

- Baugruppenadresse (Adresse 12.0)
- Kanalanfangsadresse (Adresse 14.0)
- DS-OFFSET (Adresse 18.0), liegt fest auf 0

Inhalt

Die folgende Tabelle zeigt den Inhalt des Zähler-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgelistet werden, nicht verändern.

Adresse	Variable	Datentyp	Anfangswert	Kommentar
0.0	NO	BYTE	B#16#0	Nummer
1.0	BUSY	BOOL	FALSE	TRUE: Schreibauftrag in Arbeit FALSE: Schreibauftrag nicht in Arbeit
1.1	DONE	BOOL	FALSE	TRUE: Schreibauftrag fertig FALSE: Schreibauftrag nicht fertig
1.2	IMPOSS	BOOL	FALSE	TRUE: Schreibauftrag nicht möglich FALSE: Schreibauftrag möglich
1.3	UNKNOWN	BOOL	FALSE	TRUE: Schreibauftrag unbekannt FALSE: Schreibauftrag bekannt
2.0	NO	BYTE	B#16#0	Nummer
3.0	BUSY	BOOL	FALSE	TRUE: Leseauftrag in Arbeit FALSE: Leseauftrag nicht in Arbeit
3.1	DONE	BOOL	FALSE	TRUE: Leseauftrag fertig FALSE: Leseauftrag nicht fertig
3.2	IMPOSS	BOOL	FALSE	TRUE: Leseauftrag nicht möglich FALSE: Leseauftrag möglich
3.3	UNKNOWN	BOOL	FALSE	TRUE: Leseauftrag unbekannt FALSE: Leseauftrag bekannt
4.0	RESERV_0	ARRAY [1..3] OF WORD	W#16#0	Reserviert
10.0	RESERV_1	WORD	W#16#0	Reserviert
12.0	MOD_ADR	WORD	W#16#0	Baugruppenadresse
14.0	CH_ADR	DWORD	DW#16#0	Kanaladresse
18.0	DS_OFFS	BYTE	B#16#0	Offset Datenbaustein
19.0	RESERV_2	BYTE	B#16#0	Reserviert
20.0	BIT0_0	BOOL	FALSE	Reserviert
20.1	BIT0_1	BOOL	FALSE	Reserviert

Adresse	Variable	Datentyp	Anfangswert	Kommentar
20.2	BIT0_2	BOOL	FALSE	Reserviert
20.3	BIT0_3	BOOL	FALSE	Reserviert
20.4	BIT0_4	BOOL	FALSE	Reserviert
20.5	BIT0_5	BOOL	FALSE	Reserviert
20.6	BIT0_6	BOOL	FALSE	Reserviert
20.7	BIT0_7	BOOL	FALSE	Reserviert
21.0	CTRL_DQ0	BOOL	FALSE	TRUE: Ausgang 0 freigegeben FALSE: Ausgang 0 nicht freigegeben
21.1	CTRL_DQ1	BOOL	FALSE	TRUE: Ausgang 1 freigegeben FALSE: Ausgang 1 nicht freigegeben
21.2	CTRL_DQ2	BOOL	FALSE	TRUE: Ausgang 2 freigegeben FALSE: Ausgang 2 nicht freigegeben
21.3	CTRL_DQ3	BOOL	FALSE	TRUE: Ausgang 3 freigegeben FALSE: Ausgang 3 nicht freigegeben
21.4	CTRL_DQ4	BOOL	FALSE	TRUE: Ausgang 4 freigegeben FALSE: Ausgang 4 nicht freigegeben
21.5	CTRL_DQ5	BOOL	FALSE	TRUE: Ausgang 5 freigegeben FALSE: Ausgang 5 nicht freigegeben
21.6	CTRL_DQ6	BOOL	FALSE	TRUE: Ausgang 6 freigegeben FALSE: Ausgang 6 nicht freigegeben
21.7	CTRL_DQ7	BOOL	FALSE	TRUE: Ausgang 7 freigegeben FALSE: Ausgang 7 nicht freigegeben
22.0	SET_DQ0	BOOL	FALSE	TRUE: Ausgang 0 gesetzt FALSE: Ausgang 0 nicht gesetzt
22.1	SET_DQ1	BOOL	FALSE	TRUE: Ausgang 1 gesetzt FALSE: Ausgang 1 nicht gesetzt
22.2	SET_DQ2	BOOL	FALSE	TRUE: Ausgang 2 gesetzt FALSE: Ausgang 2 nicht gesetzt
22.3	SET_DQ3	BOOL	FALSE	TRUE: Ausgang 3 gesetzt FALSE: Ausgang 3 nicht gesetzt
22.4	SET_DQ4	BOOL	FALSE	TRUE: Ausgang 4 gesetzt FALSE: Ausgang 4 nicht gesetzt
22.5	SET_DQ5	BOOL	FALSE	TRUE: Ausgang 5 gesetzt FALSE: Ausgang 5 nicht gesetzt
22.6	SET_DQ6	BOOL	FALSE	TRUE: Ausgang 6 gesetzt FALSE: Ausgang 6 nicht gesetzt
22.7	SET_DQ7	BOOL	FALSE	TRUE: Ausgang 7 gesetzt FALSE: Ausgang 7 nicht gesetzt
23.0	SW_GATE0	BOOL	FALSE	TRUE: SW-Tor Zähler 0 offen FALSE: SW-Tor Zähler 0 geschlossen
23.1	SW_GATE1	BOOL	FALSE	TRUE: SW-Tor Zähler 1 offen FALSE: SW-Tor Zähler 1 geschlossen

Adresse	Variable	Datentyp	Anfangswert	Kommentar
23.2	SW_GATE2	BOOL	FALSE	TRUE: SW-Tor Zähler 2 offen FALSE: SW-Tor Zähler 2 geschlossen
23.3	SW_GATE3	BOOL	FALSE	TRUE: SW-Tor Zähler 3 offen FALSE: SW-Tor Zähler 3 geschlossen
23.4	SW_GATE4	BOOL	FALSE	TRUE: SW-Tor Zähler 4 offen FALSE: SW-Tor Zähler 4 geschlossen
23.5	SW_GATE5	BOOL	FALSE	TRUE: SW-Tor Zähler 5 offen FALSE: SW-Tor Zähler 5 geschlossen
23.6	SW_GATE6	BOOL	FALSE	TRUE: SW-Tor Zähler 6 offen FALSE: SW-Tor Zähler 6 geschlossen
23.7	SW_GATE7	BOOL	FALSE	TRUE: SW-Tor Zähler 7 offen FALSE: SW-Tor Zähler 7 geschlossen
24.0	CTRL_DWORD1	DWORD	DW#16#0	Reserviert
28.0	CTRL_DWORD2	DWORD	DW#16#0	Reserviert
32.0	CTRL_DWORD3	DWORD	DW#16#0	Reserviert
36.0	BIT0_0	BOOL	FALSE	Reserviert
36.1	STS_TFB	BOOL	FALSE	TRUE: PG-Betrieb aktiv FALSE: kein PG-Betrieb
36.2	BIT0_2	BOOL	FALSE	Reserviert
36.3	BIT0_3	BOOL	FALSE	Reserviert
36.4	DATA_ERR	BOOL	FALSE	Datenfehler
36.5	BIT0_5	BOOL	FALSE	Reserviert
36.6	BIT0_6	BOOL	FALSE	Reserviert
36.7	PARA	BOOL	FALSE	TRUE: Baugruppe parametriert FALSE: Baugruppe nicht parametriert
37.0	STS_CMP0	BOOL	FALSE	TRUE: Vergleicher 0 angesprochen FALSE: Vergleicher 0 nicht angesprochen
37.1	STS_CMP1	BOOL	FALSE	TRUE: Vergleicher 1 angesprochen FALSE: Vergleicher 1 nicht angesprochen
37.2	STS_CMP2	BOOL	FALSE	TRUE: Vergleicher 2 angesprochen FALSE: Vergleicher 2 nicht angesprochen
37.3	STS_CMP3	BOOL	FALSE	TRUE: Vergleicher 3 angesprochen FALSE: Vergleicher 3 nicht angesprochen
37.4	STS_CMP4	BOOL	FALSE	TRUE: Vergleicher 4 angesprochen FALSE: Vergleicher 4 nicht angesprochen
37.5	STS_CMP5	BOOL	FALSE	TRUE: Vergleicher 5 angesprochen FALSE: Vergleicher 5 nicht angesprochen
37.6	STS_CMP6	BOOL	FALSE	TRUE: Vergleicher 6 angesprochen FALSE: Vergleicher 6 nicht angesprochen
37.7	STS_CMP7	BOOL	FALSE	TRUE: Vergleicher 7 angesprochen FALSE: Vergleicher 7 nicht angesprochen
38.0	STS_UFLW0	BOOL	FALSE	TRUE: Unterlauf Zähler 0 FALSE: kein Unterlauf Zähler 0

Adresse	Variable	Datentyp	Anfangswert	Kommentar
38.1	STS_UFLW1	BOOL	FALSE	TRUE: Unterlauf Zähler 1 FALSE: kein Unterlauf Zähler 1
38.2	STS_UFLW2	BOOL	FALSE	TRUE: Unterlauf Zähler 2 FALSE: kein Unterlauf Zähler 2
38.3	STS_UFLW3	BOOL	FALSE	TRUE: Unterlauf Zähler 3 FALSE: kein Unterlauf Zähler 3
38.4	STS_UFLW4	BOOL	FALSE	TRUE: Unterlauf Zähler 4 FALSE: kein Unterlauf Zähler 4
38.5	STS_UFLW5	BOOL	FALSE	TRUE: Unterlauf Zähler 5 FALSE: kein Unterlauf Zähler 5
38.6	STS_UFLW6	BOOL	FALSE	TRUE: Unterlauf Zähler 6 FALSE: kein Unterlauf Zähler 6
38.7	STS_UFLW7	BOOL	FALSE	TRUE: Unterlauf Zähler 7 FALSE: kein Unterlauf Zähler 7
39.0	STS_OFLW0	BOOL	FALSE	TRUE: Überlauf Zähler 0 FALSE: kein Überlauf Zähler 0
39.1	STS_OFLW1	BOOL	FALSE	TRUE: Überlauf Zähler 1 FALSE: kein Überlauf Zähler 1
39.2	STS_OFLW2	BOOL	FALSE	TRUE: Überlauf Zähler 2 FALSE: kein Überlauf Zähler 2
39.3	STS_OFLW3	BOOL	FALSE	TRUE: Überlauf Zähler 3 FALSE: kein Überlauf Zähler 3
39.4	STS_OFLW4	BOOL	FALSE	TRUE: Überlauf Zähler 4 FALSE: kein Überlauf Zähler 4
39.5	STS_OFLW5	BOOL	FALSE	TRUE: Überlauf Zähler 5 FALSE: kein Überlauf Zähler 5
39.6	STS_OFLW6	BOOL	FALSE	TRUE: Überlauf Zähler 6 FALSE: kein Überlauf Zähler 6
39.7	STS_OFLW7	BOOL	FALSE	TRUE: Überlauf Zähler 7 FALSE: kein Überlauf Zähler 7
40.0	STS_DIR0	BOOL	FALSE	TRUE: Zählrichtung Zähler 0 rückwärts FALSE: Zählrichtung Zähler 0 vorwärts
40.1	STS_DIR1	BOOL	FALSE	TRUE: Zählrichtung Zähler 1 rückwärts FALSE: Zählrichtung Zähler 1 vorwärts
40.2	STS_DIR2	BOOL	FALSE	TRUE: Zählrichtung Zähler 2 rückwärts FALSE: Zählrichtung Zähler 2 vorwärts
40.3	STS_DIR3	BOOL	FALSE	TRUE: Zählrichtung Zähler 3 rückwärts FALSE: Zählrichtung Zähler 3 vorwärts
40.4	STS_DIR4	BOOL	FALSE	TRUE: Zählrichtung Zähler 4 rückwärts FALSE: Zählrichtung Zähler 4 vorwärts
40.5	STS_DIR5	BOOL	FALSE	TRUE: Zählrichtung Zähler 5 rückwärts FALSE: Zählrichtung Zähler 5 vorwärts

Adresse	Variable	Datentyp	Anfangswert	Kommentar
40.6	STS_DIR6	BOOL	FALSE	TRUE: Zählrichtung Zähler 6 rückwärts FALSE: Zählrichtung Zähler 6 vorwärts
40.7	STS_DIR7	BOOL	FALSE	TRUE: Zählrichtung Zähler 7 rückwärts FALSE: Zählrichtung Zähler 7 vorwärts
41.0	STS_DI0	BOOL	FALSE	TRUE: Digitaleingang 0 gesetzt FALSE: Digitaleingang 0 nicht gesetzt
41.1	STS_DI1	BOOL	FALSE	TRUE: Digitaleingang 1 gesetzt FALSE: Digitaleingang 1 nicht gesetzt
41.2	STS_DI2	BOOL	FALSE	TRUE: Digitaleingang 2 gesetzt FALSE: Digitaleingang 2 nicht gesetzt
41.3	STS_DI3	BOOL	FALSE	TRUE: Digitaleingang 3 gesetzt FALSE: Digitaleingang 3 nicht gesetzt
41.4	STS_DI4	BOOL	FALSE	TRUE: Digitaleingang 4 gesetzt FALSE: Digitaleingang 4 nicht gesetzt
41.5	STS_DI5	BOOL	FALSE	TRUE: Digitaleingang 5 gesetzt FALSE: Digitaleingang 5 nicht gesetzt
41.6	STS_DI6	BOOL	FALSE	TRUE: Digitaleingang 6 gesetzt FALSE: Digitaleingang 6 nicht gesetzt
41.7	STS_DI7	BOOL	FALSE	TRUE: Digitaleingang 7 gesetzt FALSE: Digitaleingang 7 nicht gesetzt
42.0	STS_DQ0	BOOL	FALSE	TRUE: Digitalausgang 0 gesetzt FALSE: Digitalausgang 0 nicht gesetzt
42.1	STS_DQ1	BOOL	FALSE	TRUE: Digitalausgang 1 gesetzt FALSE: Digitalausgang 1 nicht gesetzt
42.2	STS_DQ2	BOOL	FALSE	TRUE: Digitalausgang 2 gesetzt FALSE: Digitalausgang 2 nicht gesetzt
42.3	STS_DQ3	BOOL	FALSE	TRUE: Digitalausgang 3 gesetzt FALSE: Digitalausgang 3 nicht gesetzt
42.4	STS_DQ4	BOOL	FALSE	TRUE: Digitalausgang 4 gesetzt FALSE: Digitalausgang 4 nicht gesetzt
42.5	STS_DQ5	BOOL	FALSE	TRUE: Digitalausgang 5 gesetzt FALSE: Digitalausgang 5 nicht gesetzt
42.6	STS_DQ6	BOOL	FALSE	TRUE: Digitalausgang 6 gesetzt FALSE: Digitalausgang 6 nicht gesetzt
42.7	STS_DQ7	BOOL	FALSE	TRUE: Digitalausgang 7 gesetzt FALSE: Digitalausgang 7 nicht gesetzt
43.0	STS_GATE0	BOOL	FALSE	TRUE: internes Tor Zähler 0 offen FALSE: internes Tor Zähler 0 geschlossen
43.1	STS_GATE1	BOOL	FALSE	TRUE: internes Tor Zähler 1 offen FALSE: internes Tor Zähler 1 geschlossen
43.2	STS_GATE2	BOOL	FALSE	TRUE: internes Tor Zähler 2 offen FALSE: internes Tor Zähler 2 geschlossen

Adresse	Variable	Datentyp	Anfangswert	Kommentar
43.3	STS_GATE3	BOOL	FALSE	TRUE: internes Tor Zähler 3 offen FALSE: internes Tor Zähler 3 geschlossen
43.4	STS_GATE4	BOOL	FALSE	TRUE: internes Tor Zähler 4 offen FALSE: internes Tor Zähler 4 geschlossen
43.5	STS_GATE5	BOOL	FALSE	TRUE: internes Tor Zähler 5 offen FALSE: internes Tor Zähler 5 geschlossen
43.6	STS_GATE6	BOOL	FALSE	TRUE: internes Tor Zähler 6 offen FALSE: internes Tor Zähler 6 geschlossen
43.7	STS_GATE7	BOOL	FALSE	TRUE: internes Tor Zähler 7 offen FALSE: internes Tor Zähler 7 geschlossen
44	USER_STAT_WORD0	WORD	W#16#0	je nach Parametrierung Zähl-/Messwerte
46	USER_STAT_WORD1	WORD	W#16#0	je nach Parametrierung Zähl-/Messwerte
48	USER_STAT_WORD2	WORD	W#16#0	je nach Parametrierung Zähl-/Messwerte
50	USER_STAT_WORD3	WORD	W#16#0	je nach Parametrierung Zähl-/Messwerte
52	LOAD-VAL0	DINT	L#0	Zähler 0 direkt laden
56	LOAD-VAL1	DINT	L#0	Zähler 1 direkt laden
60	LOAD-VAL2	DINT	L#0	Zähler 2 direkt laden
64	LOAD-VAL3	DINT	L#0	Zähler 3 direkt laden
68	LOAD-VAL4	DINT	L#0	Zähler 4 direkt laden
72	LOAD-VAL5	DINT	L#0	Zähler 5 direkt laden
76	LOAD-VAL6	DINT	L#0	Zähler 6 direkt laden
80	LOAD-VAL7	DINT	L#0	Zähler 7 direkt laden
84	LOAD-PREPARE-VAL0	DINT	L#0	Zähler 0 vorbereitend laden
88	LOAD-PREPARE-VAL1	DINT	L#0	Zähler 1 vorbereitend laden
92	LOAD-PREPARE-VAL2	DINT	L#0	Zähler 2 vorbereitend laden
96	LOAD-PREPARE-VAL3	DINT	L#0	Zähler 3 vorbereitend laden
100	LOAD-PREPARE-VAL4	DINT	L#0	Zähler 4 vorbereitend laden
104	LOAD-PREPARE-VAL5	DINT	L#0	Zähler 5 vorbereitend laden
108	LOAD-PREPARE-VAL6	DINT	L#0	Zähler 6 vorbereitend laden
112	LOAD-PREPARE-VAL7	DINT	L#0	Zähler 7 vorbereitend laden
116	CMP-VAL0	DINT	L#0	Vergleicher 0 laden
120	CMP-VAL1	DINT	L#0	Vergleicher 1 laden
124	CMP-VAL2	DINT	L#0	Vergleicher 2 laden
128	CMP-VAL3	DINT	L#0	Vergleicher 3 laden
132	CMP-VAL4	DINT	L#0	Vergleicher 4 laden
136	CMP-VAL5	DINT	L#0	Vergleicher 5 laden
140	CMP-VAL6	DINT	L#0	Vergleicher 6 laden
144	CMP-VAL7	DINT	L#0	Vergleicher 7 laden
148	ACT_CNTV0	DINT	L#0	Aktueller Zählerstand 0
152	ACT_MSRV0	DINT	L#0	Messergebnis 0
156	ACT_CNTV1	DINT	L#0	Aktueller Zählerstand 1
160	ACT_MSRV1	DINT	L#0	Messergebnis 1
164	ACT_CNTV2	DINT	L#0	Aktueller Zählerstand 2

Adresse	Variable	Datentyp	Anfangswert	Kommentar
168	ACT_MSRV2	DINT	L#0	Messergebnis 2
172	ACT_CNTV3	DINT	L#0	Aktueller Zählerstand 3
176	ACT_MSRV3	DINT	L#0	Messergebnis 3
180	ACT_CNTV4	DINT	L#0	Aktueller Zählerstand 4
184	ACT_MSRV4	DINT	L#0	Messergebnis 4
188	ACT_CNTV5	DINT	L#0	Aktueller Zählerstand 5
192	ACT_MSRV5	DINT	L#0	Messergebnis 5
196	ACT_CNTV6	DINT	L#0	Aktueller Zählerstand 6
200	ACT_MSRV6	DINT	L#0	Messergebnis 6
204	ACT_CNTV7	DINT	L#0	Aktueller Zählerstand 7
208	ACT_MSRV7	DINT	L#0	Messergebnis 7
212.0	BYTE0	BYTE	B#16#0	Reserviert
213.0	BYTE1	BYTE	B#16#0	Reserviert
214.0	BYTE2	BYTE	B#16#0	Reserviert
215.0	BYTE3	BYTE	B#16#0	Reserviert
216.0	BYTE4	BYTE	B#16#0	Kanaltyp
217.0	BYTE5	BYTE	B#16#0	Länge Kanalinfo
218.0	BYTE6	BYTE	B#16#0	Anzahl Kanäle
219.0	BYTE7	BYTE	B#16#0	Kanalfehlervektor
220.0	BYTE8	BYTE	B#16#0	Fehler Zähler 0
221.0	BYTE9	BYTE	B#16#0	Fehler Zähler 1
222.0	BYTE10	BYTE	B#16#0	Fehler Zähler 2
223.0	BYTE11	BYTE	B#16#0	Fehler Zähler 3
224.0	BYTE12	BYTE	B#16#0	Fehler Zähler 4
225.0	BYTE13	BYTE	B#16#0	Fehler Zähler 5
226.0	BYTE14	BYTE	B#16#0	Fehler Zähler 6
227.0	BYTE15	BYTE	B#16#0	Fehler Zähler 7

CONTROL_SIGNALS

CONTROL_SIGNALS

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
21.0 … 7	CTRL_DQ0 … 7	BOOL	FALSE	Freigabe der Digitalausgänge 0 … 7
22.0 … 7	SET_DQ0 … 7	BOOL	FALSE	Ansteuern der Digitalausgänge 0 … 7
23.0 … 7	SW_GATE0 … 7	BOOL	FALSE	Software-Tor Zähler 0 … 7

CHECKBACK_SIGNALS

CHECKBACK_SIGNALS

Adresse	Parameter	Datentyp	Anfangs-wert	Beschreibung
36.1	STS_TFB	BOOL	FALSE	TRUE: PG-Betrieb ist aktiv, steuern von der CPU nicht möglich FALSE: PG-Betrieb abgewählt
36.4	DATA_ERR	BOOL	FALSE	TRUE: Datenfehler aufgetreten FALSE: kein Datenfehler
36.7	PARA	BOOL	FALSE	TRUE: Zählerbaugruppe ist parametriert, alle anderen CHECKBACK_SIGNALS sind gültig FALSE: Zählerbaugruppe ist nicht parametriert
37.0 … 7	STS_CMP0 … 7	BOOL	FALSE	Status Vergleicher 0 … 7 Sie müssen den Status nach dem Aufruf der Anweisung CNT2_CTR auswerten, da er danach rückgesetzt wird
38.0 … 7	STS_UFLW0 … 7	BOOL	FALSE	Status Unterlauf Zähler 0 … 7 Sie müssen den Status nach dem Aufruf der Anweisung CNT2_CTR auswerten, da er danach rückgesetzt wird
39.0 … 7	STS_OFLW0 … 7	BOOL	FALSE	Status Überlauf Zähler 0 … 7 Sie müssen den Status nach dem Aufruf der Anweisung CNT2_CTR auswerten, da er danach rückgesetzt wird
40.0 … 7	STS_DIR0 … 7	BOOL	FALSE	Status Zählrichtung Zähler 0 … 7 Sie müssen den Status nach dem Aufruf der Anweisung CNT2_CTR auswerten, da er danach rückgesetzt wird
41.0 … 7	STS_DI0 … 7	BOOL	FALSE	Status Digitaleingang 0 … 7
42.0 … 7	STS_DQ0 … 7	BOOL	FALSE	Status Digitalausgang 0 … 7
43.0 … 7	STS_GATE0 … 7	BOOL	FALSE	Status Internes Tor Zähler 0 … 7
44	USER_STAT_DWORD0	WORD	W#16#0	je nach Parametrierung Zähl- / Messwert
46	USER_STAT_DWORD1	WORD	W#16#0	je nach Parametrierung Zähl- / Messwert
48	USER_STAT_DWORD2	WORD	W#16#0	je nach Parametrierung Zähl- / Messwert
50	USER_STAT_DWORD3	WORD	W#16#0	je nach Parametrierung Zähl- / Messwert

Schreibauftrag

Schreibauftrag

Einen neuen Schreibauftrag stoßen Sie an, indem Sie einen Schreibauftrag in JOB_WR.NO eintragen. Folgende Schreibaufträge sind zulässig:

JOB_WR.NO (DBB0)	Eintrag in CNT2_CHANTYPE	Adresse im Zähler-DB	Bedeutung für Zähl-Betriebsart	Bedeutung für Mess-Betriebsart
0	keine	keine	kein Schreibauftrag / letzter Schreibauftrag beendet	
10	LOAD_VAL0	52	Zähler 0 direkt laden	Untergrenze 0 laden

JOB_WR.NO (DBB0)	Eintrag in CNT2_CHANTYPE	Adresse im Zähler-DB	Bedeutung für Zähl-Betriebsart	Bedeutung für Mess-Betriebsart
11	LOAD_VAL1	56	Zähler 1 direkt laden	Untergrenze 1 laden
12	LOAD_VAL2	60	Zähler 2 direkt laden	Untergrenze 2 laden
13	LOAD_VAL3	64	Zähler 3 direkt laden	Untergrenze 3 laden
14	LOAD_VAL4	68	Zähler 4 direkt laden	Untergrenze 4 laden
15	LOAD_VAL5	72	Zähler 5 direkt laden	Untergrenze 5 laden
16	LOAD_VAL6	76	Zähler 6 direkt laden	Untergrenze 6 laden
17	LOAD_VAL7	80	Zähler 7 direkt laden	Untergrenze 7 laden
20	LOAD_PREPARE_VAL0	84	Zähler 0 vorbereitend laden	Obergrenze 0 laden
21	LOAD_PREPARE_VAL1	88	Zähler 1 vorbereitend laden	Obergrenze 1 laden
22	LOAD_PREPARE_VAL2	92	Zähler 2 vorbereitend laden	Obergrenze 2 laden
23	LOAD_PREPARE_VAL3	96	Zähler 3 vorbereitend laden	Obergrenze 3 laden
24	LOAD_PREPARE_VAL4	100	Zähler 4 vorbereitend laden	Obergrenze 4 laden
25	LOAD_PREPARE_VAL5	104	Zähler 5 vorbereitend laden	Obergrenze 5 laden
26	LOAD_PREPARE_VAL6	108	Zähler 6 vorbereitend laden	Obergrenze 6 laden
27	LOAD_PREPARE_VAL7	112	Zähler 7 vorbereitend laden	Obergrenze 7 laden
30	CMP_VAL0	116	Vergleicher 0 laden	
31	CMP_VAL1	120	Vergleicher 1 laden	
32	CMP_VAL2	124	Vergleicher 2 laden	
33	CMP_VAL3	128	Vergleicher 3 laden	
34	CMP_VAL4	132	Vergleicher 4 laden	
35	CMP_VAL5	136	Vergleicher 5 laden	
36	CMP_VAL6	140	Vergleicher 6 laden	
37	CMP_VAL7	144	Vergleicher 7 laden	
40	LOAD_VAL0 bis LOAD_VAL3	52 bis 67	Zähler 0 bis 3 direkt laden	Untergrenze 0 bis 3 laden
41	LOAD_VAL4 bis LOAD_VAL7	68 bis 83	Zähler 4 bis 7 direkt laden	Untergrenze 4 bis 7 laden
42	LOAD_VAL0 bis LOAD_VAL7	52 bis 83	Zähler 0 bis 7 direkt laden	Untergrenze 0 bis 7 laden
50	LOAD_PREPARE_VAL0 bis LOAD_PREPARE_VAL3	84 bis 99	Zähler 0 bis 3 vorbereitend laden	Obergrenze 0 bis 3 laden
51	LOAD_PREPARE_VAL4 bis LOAD_PREPARE_VAL7	100 bis 111	Zähler 4 bis 7 vorbereitend laden	Obergrenze 4 bis 7 laden
52	LOAD_PREPARE_VAL0 bis LOAD_PREPARE_VAL7	84 bis 111	Zähler 0 bis 7 vorbereitend laden	Obergrenze 0 bis 7 laden
60	CMP_VAL0 bis CMP_VAL3	116 bis 131	Vergleicher 0 bis 3 laden	
61	CMP_VAL4 bis CMP_VAL7	132 bis 147	Vergleicher 4 bis 7 laden	
62	CMP_VAL0 bis CMP_VAL7	116 bis 147	Vergleicher 0 bis 7 laden	

Status des Schreibauftrags

Status eines Schreibauftrags

Der Status eines Schreibauftrags wird im Zähler-DB (Datenbyte DBB1) angezeigt.

Bit im JOB_WR (DBX1.)	Bedeutung
.BUSY, 0	= 1: Schreibauftrag läuft. Die Anweisung CNT2WRPN setzt dieses Bit, sobald sie einen Schreibauftrag bearbeitet (JOB_WR.NO > 0 und JOB_WR.IMPOSS = 0). Die Anweisung CNT2WRPN löscht das Bit, sobald der Schreibauftrag beendet ist (JOB_WR.NO = 0).
.DONE, 1	= 1: Schreibauftrag beendet. Die Anweisung CNT2WRPN setzt dieses Bit, sobald ein Schreibauftrag (auch fehlerhaft) beendet ist. Die Anweisung CNT2WRPN löscht das Bit, wenn ein neuer Schreibauftrag beginnt. Sie können dieses Bit auch über das Anwenderprogramm löschen.
.IMPOSS, 2	= 1: Der Schreibauftrag kann nicht bearbeitet werden (die Zählerbaugruppe ist nicht parametriert, der Anlauf oder der PG-Betrieb ist aktiv). Sie können den Schreibauftrag (JOB_WR) anstehen lassen oder löschen. Die Anweisung CNT2WRPN löscht das Bit, wenn die obengenannten Bedingungen erfüllt sind.
.UNKOWN, 3	= 1: Schreibauftrag unbekannt. Der von Ihnen angegebene Schreibauftrag (JOB_WR) liegt nicht im zulässigen Bereich (siehe Fehlermeldung). Die Anweisung CNT2WRPN löscht dieses Bit, sobald in JOB_WR eine zulässige Nummer eingetragen wird. Die unbekannte Nummer bleibt solange erhalten.

Aufgetretene Fehler werden im Binärergebnis (BIE = 0) angezeigt. Mögliche Fehler sind:

- unbekannter Schreibauftrag (siehe JOB_WR.UNKOWN).
- Datenübertragungsfehler bei der Datenübertragung mit der Anweisung WRREC. Der Fehler wird am Ausgangsparameter RET_VAL zur Verfügung gestellt.
- Die übertragenen Daten werden von der Baugruppe auf Datenfehler geprüft und interpretiert. Tritt ein Datenfehler auf, ist im Zähler-DB das Bit CHECKBACK_SIGNALS.DATA_ERR = 1 gesetzt.

Leseauftrag

Leseauftrag

Einen neuen Leseauftrag stoßen Sie an, indem Sie einen Leseauftrag in JOB_RD.NO eintragen. Folgende Leseaufträge sind zulässig:

JOB_RD.NO (DBB2)	Eintrag in CNT2_CHAN-TYPE	Adresse im Zähler-DB	Bedeutung
0	keine	keine	kein Leseauftrag / letzter Leseauftrag beendet
100	ACT_CNTV0 ACT_MSRV0 ACT_CNTV1 ACT_MSRV1 ACT_CNTV2 ACT_MSRV2 ACT_CNTV3 ACT_MSRV3	148	aktueller Zählerstand 0 bis 3 und Messergebnis 0 bis 3
101	ACT_CNTV4 ACT_MSRV4 ACT_CNTV5 ACT_MSRV5 ACT_CNTV6 ACT_MSRV6 ACT_CNTV7 ACT_MSRV7	180	aktueller Zählerstand 4 bis 7 und Messergebnis 4 bis 7

Status des Leseauftrags

Status eines Leseauftrags

Der Status eines Leseauftrags wird im Zähler-DB angezeigt (Datenbyte DBB3)

Bit im JOB_RD (DBX3.)	Beschreibung
.BUSY, 0	= 1: Leseauftrag läuft. Die Anweisung CNT2RDPN setzt dieses Bit, sobald sie einen Leseauftrag bearbeitet (JOB_RD.NO > 0 und JOB_RD.IMPOSS = 0). Die Anweisung CNT2RDPN löscht das Bit, sobald der Leseauftrag beendet ist (JOB_RD.NO = 0).
.DONE, 1	= 1: Leseauftrag beendet. Die Anweisung CNT2RDPN setzt dieses Bit, sobald ein Leseauftrag (auch fehlerhaft) beendet ist. Die Anweisung CNT2RDPN löscht das Bit, wenn ein neuer Leseauftrag beginnt. Sie können dieses Bit auch über das Anwenderprogramm löschen.

Bit im JOB_RD (DBX3.)	Beschreibung
.IMPOSS, 2	= 1: Der Leseauftrag kann nicht bearbeitet werden (die Zählerbaugruppe ist nicht parametriert, der Anlauf oder der PG-Betrieb ist aktiv). Sie können den Leseauftrag (JOB_RD.NO) anstehen lassen oder löschen. Die Anweisung CNT2RDPN löscht das Bit, wenn die obengenannten Bedingungen erfüllt sind.
.UNKOWN, 3	= 1: Leseauftrag unbekannt. Der von Ihnen angegebene Leseauftrag (JOB_RD) liegt nicht im zulässigen Bereich (siehe Fehlerauswertung). Die Anweisung CNT2RDPN löscht dieses Bit, sobald in JOB_RD.NO eine zulässige Nummer eingetragen wird. Die unbekannte Nummer bleibt solange erhalten.

Aufgetretene Fehler werden im Binärergebnis (BIE = 0) angezeigt. Mögliche Fehler sind:

- unbekannter Leseauftrag (siehe JOB_RD.UNKOWN).
- Datenübertragungsfehler bei der Datenübertragung mit der Anweisung RDREC. Der Fehler wird am Ausgangsparameter RET_VAL zur Verfügung gestellt.

FM x51 Positionieren

ABS_INIT

Beschreibung

Die Anweisung ABS_INIT initialisiert die folgenden Daten im Kanal-DB:

- Die Steuersignale
- Die Rückmeldesignale
- Die Anstoßbits, Fertigbits und Fehlerbits der Aufträge
- Die Funktionsschalter und ihre Fertigbits und Fehlerbits
- Die Auftragsverwaltung für die Anweisung ABS_CTRL (Seite 6143) bzw. ABS_CTRL_451 (Seite 6147)

Aufruf

Die Anweisung muss nach einem Anlauf, d. h nach Einschalten der Versorgungsspannung der Baugruppe bzw. der CPU, für jeden Kanal durchlaufen werden. Rufen Sie sie deshalb z. B. im Neustart-OB OB 100 und dem Ziehen/Stecken-OB OB 83 oder in der Initialisierungsphase Ihres Anwenderprogramms auf. Somit ist sichergestellt, dass Ihr Anwenderprogramm nach einem Neustart der CPU oder einem Baugruppenanlauf nicht auf veraltete Daten zugreift.

Verwendeter Datenbaustein

Kanal-DB (Seite 6151):

Im Kanal-DB muss die Baugruppenadresse eingetragen sein. Der Eintrag sollte im Anlauf-OB (OB 100) erfolgen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Kanal-DB

Rückgabewerte

Die Anweisung liefert keinen Rückgabewert.

ABS_CTRL

Beschreibung

Mit der Anweisung ABS_CTRL können Sie die Betriebsdaten für jeden Kanal der Baugruppe lesen, die Kanäle parametrieren und während des Betriebs steuern. Dazu verwenden Sie Steuersignale, Rückmeldesignale, Funktionsschalter sowie Schreib- und Leseaufträge.

Arbeitsweise

Bei jedem Aufruf führt die Anweisung folgende Tätigkeiten aus:

- Rückmeldesignale lesen:
  Die Anweisung ABS_CTRL liest alle Rückmeldesignale für einen Kanal und trägt sie in den Kanal-DB ein. Da die Steuersignale und Aufträge erst anschließend bearbeitet werden, geben die Rückmeldesignale den Status des Kanals vor dem Aufruf der Anweisung wieder.
- Auftragsverwaltung (Seite 6160):
  Die Anweisung ABS_CTRL bearbeitet die Schreibaufträge und Leseaufträge und überträgt Daten zwischen Kanal-DB, Parameter-DB und der Baugruppe.
- Steuersignale schreiben:
  Die Steuersignale, die im Kanal-DB eingetragen sind, werden zur Baugruppe übertragen.

Aufruf

Die Anweisung ABS_CTRL muss zyklisch für jeden Kanal aufgerufen werden, z. B. im OB 1.

Bevor Sie die Anweisung ABS_CTRL aufrufen, tragen Sie alle Daten, die für die Ausführung der beabsichtigten Funktionen erforderlich sind, in den Kanal-DB ein.

Verwendete Datenbausteine

- Kanal-DB (Seite 6151):
  Der Kanal-DB ist der Instanz-DB der Anweisung ABS_CTRL. Im Kanal-DB müssen die Baugruppenadresse und die Kanalnummer eingetragen sein. Falsche Angaben können zu Peripheriezugriffsfehlern oder zu einem Zugriff auf eine andere Baugruppe führen, die Datenverfälschungen hervorrufen.
- Parameter-DB (Seite 6157):
  Wenn Sie Maschinendaten über Aufträge schreiben oder lesen wollen, benötigen Sie einen Parameter-DB, dessen Nummer im Kanal-DB eingetragen sein muss.

Rückgabewerte

Die Anweisung liefert folgende Rückgabewerte:

RETVAL	BIE	Beschreibung
1	1	Mindestens 1 Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler: Datenfehler (DATA_ERR) oder Kommunikationsfehler (JOB_ERR) aufgetreten

Steuersignale

Liegt ein STOP-Signal oder ein Bedienfehler an oder fehlt die Antriebsfreigabe, setzt die Anweisung die Steuersignale START, DIR_M und DIR_P zurück.

Sie können eine Fahrt wieder starten, nachdem Sie den Bedienfehler mit OT_ERR_A=1 quittiert haben. Bei dieser Quittierung können Sie keine weitere Aufträge und Steuersignale abgeben.

Wenn kein Bedienfehler ansteht, setzt die Anweisung die Quittung für den Bedienfehler OT_ERR_A auf 0.

Wenn der Kanal den Beginn der Fahrt meldet, setzt die Anweisung die Startsignale START, DIR_P und DIR_M zurück, außer bei der Betriebsart "Tippen".

Wenn die Achse nicht parametriert ist, hält die Anweisung alle Steuersignale mit Ausnahme der Bedienfehlerquittung OT_ERR_A zurück.

Aufträge und Steuersignale

Sie können mehrere Aufträge gleichzeitig abgeben, auch zusammen mit den für die Positionierung notwendigen Steuersignalen. Falls mindestens ein Schreibauftrag gleichzeitig mit den Steuersignalen START, DIR_M oder DIR_P abgegeben wurde, hält die Anweisung diese Steuersignale solange zurück, bis die Schreibaufträge abgearbeitet sind.

Anlauf

Rufen Sie beim Anlauf der Baugruppe bzw. der CPU die Anweisung ABS_INIT (Seite 6142) auf. Dabei werden u. a. auch die Funktionsschalter zurückgesetzt. Die Anweisung ABS_CTRL quittiert den Anlauf der Baugruppe. Während dieser Zeit sind RETVAL und JOBBUSY = 1.

Verhalten im Fehlerfall

Wenn bei einem Schreibauftrag fehlerhafte Daten geschrieben wurden, liefert der Kanal die Rückmeldung DATA_ERR = 1 im Kanal-DB. Wenn bei einem Schreib- oder Leseauftrag ein Fehler bei der Kommunikation mit der Baugruppe auftritt, wird die Fehlerursache im Parameter JOB_ERR im Kanal-DB abgelegt.

- Fehler bei einem Schreibauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt. Bei allen noch anstehenden Schreibaufträgen wird ebenfalls das Anstoßbit zurückgenommen, jedoch nur das Fehlerbit _ERR gesetzt. Die noch anstehenden Schreibaufträge werden zurückgenommen, weil hier Aufträge aufeinander aufsetzen können.
  Die anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.
- Fehler bei einem Leseauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt.
  Die noch anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.

ABS_DIAG

Beschreibung

Mit der Anweisung ABS_DIAG lesen Sie den Diagnosepuffer der Baugruppe aus und stellen ihn für eine Anzeige im B&B-System oder für eine programmierte Auswertung zur Verfügung.

Arbeitsweise

Die Anweisung liest den Diagnosepuffer aus, wenn über das Rückmeldesignal DIAG = 1 im Kanal-DB ein neuer Eintrag im Diagnosepuffer angezeigt wird. Nach dem Lesen des Diagnosepuffers wird das DIAG-Bit im Kanal-DB von der Baugruppe auf 0 gesetzt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden, z. B. im OB 1. Ein zusätzlicher Aufruf in einem Alarm-OB ist nicht zulässig. Für die vollständige Ausführung der Anweisung sind mindestens 2 Aufrufe (Zyklen) notwendig.

Verwendeter Datenbaustein

Diagnose-DB (Seite 6159):

De Diagnose-DB ist der Instanz-DB der Anweisung ABS_DIAG. Im Diagnose-DB muss die Baugruppenadresse eingetragen sein. Der neueste Eintrag des Diagnosepuffers wird in der Struktur DIAG[1] und der älteste Eintrag in der Struktur DIAG[9] eingetragen.

Rückgabewerte

Die Anweisung liefert im Parameter RETVAL des Diagnose-DB folgende Rückgabewerte:

RETVAL	BIE	Beschreibung
1	1	Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler

Aufträge

Sie können den Diagnosepuffer unabhängig von einem neuen Eintrag lesen, wenn Sie das Anstoßbit DIAGRD_EN im Diagnose-DB setzen. Nach dem Lesen des Diagnosepuffers wird das Anstoßbit auf 0 gesetzt.

Führen Sie diesen Auftrag nach einem CPU-Anlauf und einem Baugruppenanlauf aus. Damit stellen Sie sicher, dass der Inhalt des Diagnose-DBs mit dem Inhalt des Diagnosepuffers der Baugruppe übereinstimmt, auch wenn die Baugruppe keinen neuen Eintrag im Diagnosepuffer vorgenommen hat.

Anlauf

Die Anweisung führt keine Anlaufbearbeitung durch.

Verhalten im Fehlerfall

Bei einer fehlerhaften Ausführung finden Sie die Fehlerursache im Diagnose-DB im Parameter JOB_ERR.

Diagnosedaten

Fehler	Auswerten über OB 82
Baugruppenstörung	OB82_MDL_DEFECT
Interner Fehler	OB82_INT_FAULT
Externer Fehler	OB82_EXT_FAULT
Kanalfehler vorhanden	OB82_PNT_INFO
Externe Hilfsspannung fehlt	OB82_EXT_VOLTAGE
Zeitüberwachung hat angesprochen	OB82_WTCH_DOG_FLT
Baugruppeninterne Versorgungsspannung ausgefallen	OB82_INT_PS_FLT
Prozessalarm ging verloren	OB82_HW_INTR_FLT

Fehlerklassen

Die auftretenden Fehler sind in folgenden Fehlerklassen zusammengefasst:

- Fehlerklasse 1: Betriebsfehler (Seite 6166)
- Fehlerklasse 2: Bedienfehler (Seite 6167)

- Fehlerklasse 4: Datenfehler (Seite 6168)
- Fehlerklasse 5: Maschinendatenfehler (Seite 6169)
- Fehlerklasse 6: Schrittmaßtabellen-Fehler (Seite 6172)
- Fehlerklasse 15: Meldungen (Seite 6172)
- Fehlerklasse 128: Diagnosefehler (Seite 6173)

ABS_CTRL_451

Beschreibung

Mit der Anweisung ABS_CTRL_451 können Sie die Betriebsdaten für jeden Kanal der Baugruppe lesen, die Kanäle parametrieren und während des Betriebs steuern. Dazu verwenden Sie Steuersignale, Rückmeldesignale, Funktionsschalter sowie Schreib- und Leseaufträge.

Arbeitsweise

Bei jedem Aufruf führt die Anweisung folgende Tätigkeiten aus:

- Rückmeldesignale lesen:
  Die Anweisung ABS_CTRL_451 liest alle Rückmeldesignale für einen Kanal und trägt sie in den Kanal-DB ein. Da die Steuersignale und Aufträge erst anschließend bearbeitet werden, geben die Rückmeldesignale den Status des Kanals vor dem Aufruf der Anweisung wieder.
- Auftragsverwaltung (Seite 6163):
  Die Anweisung ABS_CTRL_451 bearbeitet die Schreibaufträge und Leseaufträge und überträgt Daten zwischen Kanal-DB, Parameter-DB und der Baugruppe.
- Steuersignale schreiben:
  Die Steuersignale, die im Kanal-DB eingetragen sind, werden zur Baugruppe übertragen.

Aufruf

Die Anweisung ABS_CTRL_451 muss zyklisch für jeden Kanal aufgerufen werden, z. B. im OB 1.

Bevor Sie die Anweisung ABS_CTRL_451 aufrufen, tragen Sie alle Daten, die für die Ausführung der beabsichtigten Funktionen erforderlich sind, in den Kanal-DB ein.

Verwendete Datenbausteine

- Kanal-DB (Seite 6151):
  Im Kanal-DB müssen die Baugruppenadresse und die Kanalnummer eingetragen sein. Falsche Angaben können zu Peripheriezugriffsfehlern oder zu einem Zugriff auf eine andere Baugruppe führen, die Datenverfälschungen hervorrufen.
- Parameter-DB (Seite 6157):
  Wenn Sie Maschinendaten über Aufträge schreiben oder lesen wollen, benötigen Sie einen Parameter-DB, dessen Nummer im Kanal-DB eingetragen sein muss.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Kanal-DB
RET_VAL	OUTPUT	INT	Rückgabewert

Rückgabewerte

Die Anweisung liefert folgende Rückgabewerte:

RET_VAL	BIE	Beschreibung
1	1	Mindestens 1 Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler: Datenfehler (DATA_ERR) oder Kommunikationsfehler (JOB_ERR) aufgetreten

Steuersignale

Liegt ein STOP-Signal oder ein Bedienfehler an oder fehlt die Antriebsfreigabe, setzt die Anweisung die Steuersignale START, DIR_M und DIR_P zurück.

Sie können eine Fahrt wieder starten, nachdem Sie den Bedienfehler mit OT_ERR_A=1 quittiert haben. Bei dieser Quittierung können Sie keine weitere Aufträge und Steuersignale abgeben.

Wenn kein Bedienfehler ansteht, setzt die Anweisung die Quittung für den Bedienfehler OT_ERR_A auf 0.

Wenn der Kanal den Beginn der Fahrt meldet, setzt die Anweisung die Startsignale START, DIR_P und DIR_M zurück, außer bei der Betriebsart "Tippen".

Wenn die Achse nicht parametriert ist, hält die Anweisung alle Steuersignale mit Ausnahme der Bedienfehlerquittung OT_ERR_A zurück.

Aufträge und Steuersignale

Sie können mehrere Aufträge gleichzeitig abgeben, auch zusammen mit den für die Positionierung notwendigen Steuersignalen. Falls mindestens ein Schreibauftrag gleichzeitig mit den Steuersignalen START, DIR_M oder DIR_P abgegeben wurde, hält die Anweisung diese Steuersignale solange zurück, bis die Schreibaufträge abgearbeitet sind.

Anlauf

Rufen Sie beim Anlauf der Baugruppe bzw. der CPU die Anweisung ABS_INIT (Seite 6142) auf. Dabei werden u. a. auch die Funktionsschalter zurückgesetzt. Die Anweisung ABS_CTRL_451 quittiert den Anlauf der Baugruppe. Während dieser Zeit sind RETVAL und JOBBUSY = 1.

Verhalten im Fehlerfall

Wenn bei einem Schreibauftrag fehlerhafte Daten geschrieben wurden, liefert der Kanal die Rückmeldung DATA_ERR = 1 im Kanal-DB. Wenn bei einem Schreib- oder Leseauftrag ein Fehler bei der Kommunikation mit der Baugruppe auftritt, wird die Fehlerursache im Parameter JOB_ERR im Kanal-DB abgelegt.

- Fehler bei einem Schreibauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt. Bei allen noch anstehenden Schreibaufträgen wird ebenfalls das Anstoßbit zurückgenommen, jedoch nur das Fehlerbit _ERR gesetzt. Die noch anstehenden Schreibaufträge werden zurückgenommen, weil hier Aufträge aufeinander aufsetzen können.
  Die anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.
- Fehler bei einem Leseauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt.
  Die noch anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.

ABS_DIAG_451

Beschreibung

Mit der Anweisung ABS_DIAG_451 lesen Sie den Diagnosepuffer der Baugruppe aus und stellen ihn für eine Anzeige im B&B-System oder für eine programmierte Auswertung zur Verfügung.

Arbeitsweise

Die Anweisung liest den Diagnosepuffer aus, wenn über das Rückmeldesignal DIAG = 1 im Kanal-DB ein neuer Eintrag im Diagnosepuffer angezeigt wird. Nach dem Lesen des Diagnosepuffers wird das DIAG-Bit im Kanal-DB von der Baugruppe auf 0 gesetzt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden, z. B. im OB 1. Ein zusätzlicher Aufruf in einem Alarm-OB ist nicht zulässig. Für die vollständige Ausführung der Anweisung sind mindestens 2 Aufrufe (Zyklen) notwendig.

Verwendeter Datenbaustein

Diagnose-DB (Seite 6159):

Im Diagnose-DB muss die Baugruppenadresse eingetragen sein. Der neueste Eintrag des Diagnosepuffers wird in der Struktur DIAG[1] und der älteste Eintrag in der Struktur DIAG[9] eingetragen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Diagnose-DB
RET_VAL	OUTPUT	INT	Rückgabewert

Rückgabewerte

Die Anweisung liefert im Parameter RET_VAL des Diagnose-DB folgende Rückgabewerte:

RET_VAL	BIE	Beschreibung
1	1	Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler

Aufträge

Sie können den Diagnosepuffer unabhängig von einem neuen Eintrag lesen, wenn Sie das Anstoßbit DIAGRD_EN im Diagnose-DB setzen. Nach dem Lesen des Diagnosepuffers wird das Anstoßbit auf 0 gesetzt.

Führen Sie diesen Auftrag nach einem CPU-Anlauf und einem Baugruppenanlauf aus. Damit stellen Sie sicher, dass der Inhalt des Diagnose-DBs mit dem Inhalt des Diagnosepuffers der Baugruppe übereinstimmt, auch wenn die Baugruppe keinen neuen Eintrag im Diagnosepuffer vorgenommen hat.

Anlauf

Die Anweisung führt keine Anlaufbearbeitung durch.

Verhalten im Fehlerfall

Bei einer fehlerhaften Ausführung finden Sie die Fehlerursache im Diagnose-DB im Parameter JOB_ERR.

Diagnosedaten

Fehler	Auswerten über OB 82
Baugruppenstörung	OB82_MDL_DEFECT
Interner Fehler	OB82_INT_FAULT
Externer Fehler	OB82_EXT_FAULT
Kanalfehler vorhanden	OB82_PNT_INFO
Externe Hilfsspannung fehlt	OB82_EXT_VOLTAGE
Frontstecker fehlt	OB82_FLD_CONNCTR
Zeitüberwachung hat angesprochen	OB82_WTCH_DOG_FLT

Fehler	Auswerten über OB 82
Baugruppeninterne Versorgungsspannung ausgefallen	OB82_INT_PS_FLT
Prozessalarm ging verloren	OB82_HW_INTR_FLT

Fehlerklassen

Die auftretenden Fehler sind in folgenden Fehlerklassen zusammengefasst:

- Fehlerklasse 1: Betriebsfehler (Seite 6166)
- Fehlerklasse 2: Bedienfehler (Seite 6167)
- Fehlerklasse 4: Datenfehler (Seite 6168)
- Fehlerklasse 5: Maschinendatenfehler (Seite 6169)
- Fehlerklasse 6: Schrittmaßtabellen-Fehler (Seite 6172)
- Fehlerklasse 15: Meldungen (Seite 6172)
- Fehlerklasse 128: Diagnosefehler (Seite 6173)

Weitere Referenzen für FM x51 ABS V1

Kanal-Datenbaustein

Aufgabe

Der Kanal-DB ist die Datenschnittstelle zwischen dem Anwenderprogramm und der FM x51. Er enthält und übernimmt alle Daten, die zur Steuerung und zum Betrieb eines Kanals notwendig sind.

Aufbau

Der Kanal-DB ist in verschiedene Bereiche unterteilt:

Kanal-DB
Adressen *)
Steuersignale
Rückmeldesignale
Funktionsschalter
Anstoßbits für Schreibaufträge
Anstoßbits für Leseaufträge
Fertigbits
Fehlerbits
Auftragsverwaltung für Anweisungen
Daten für Aufträge
*) Die Baugruppenadresse können Sie in der Parametrieroberfläche eintragen.

Inhalt

Die folgende Tabelle zeigt den Inhalt des Kanal-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgelistet werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
Adressen				
0.0	MOD_ADDR	INT	0	Baugruppenadresse
2.0	CH_NO	INT	1	Kanalnummer
10.0	PARADBNO	INT	-1	Nummer des Parameter-DB
Steuersignale				
14.3	OT_ERR_A	BOOL	FALSE	1 = Bedienfehler quittieren
15.0	START	BOOL	FALSE	1 = Positionierung starten
15.1	STOP	BOOL	FALSE	1 = Laufende Fahrt stoppen
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.6	SPEED252	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: Startgeschwindigkeit für Schrittmaßfahrt mit Schrittmaßnummer 252: – 0 = Schleichgang – 1 = Eilgang
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	Angeforderte Betriebsart • 0 = Keine Betriebsart • 1 = Tippen • 3 = Referenzpunktfahrt • 4 = Relative Schrittmaßfahrt • 5 = Absolute Schrittmaßfahrt
17.0	MODE_TYPE	BYTE	B#16#0	• Anfangsgeschwindigkeit für die Betriebsart Tippen – 0 = Schleichgang – 1 = Eilgang • Schrittmaßnummer für die Betriebsart Schrittmaßfahrt
Rückmeldesignale				
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.3	OT_ERR	BOOL	FALSE	1 = Bedienfehler aufgetreten
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler
22.7	PARA	BOOL	FALSE	1 = Achse ist parametriert
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positionierung läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
23.4	SPEED_OUT	BOOL	FALSE	• 0 = Schleichgang • 1 = Eilgang
23.5	ZSPEED	BOOL	FALSE	1 = Achse befindet sich im Stillstandsbereich
23.6	CUTOFF	BOOL	FALSE	1 = Achse befindet sich im Abschaltbereich
23.7	CHGOVER	BOOL	FALSE	1 = Achse befindet sich im Umschaltbereich
24.0	MODE_OUT	BYTE	B#16#0	Aktive Betriebsart
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
25.1	MSR_DONE	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Längenmessung / Kantenerfassung abgeschlossen
25.2	GO_M	BOOL	FALSE	1 = Achse fährt in Richtung minus
25.3	GO_P	BOOL	FALSE	1 = Achse fährt in Richtung plus
25.5	FVAL_DONE	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fliegendes Istwert setzen abgeschlossen
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht
26.0	ACT_POS	DINT	L#0	Aktueller Istwert (aktuelle Position der Achse)
Funktionsschalter				
34.0	PLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung plus
34.1	MLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung minus
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten
34.3	EDGE_ON	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Kantenerfassung ein
34.4	MSR_ON	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Längenmessung ein
Anstoßbits für Schreibaufträge				
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
35.2	DELDIST_EN	BOOL	FALSE	1 = Restweg löschen
35.3	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen rückgängig
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaßnummer 1 … 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaßnummer 51 … 100) schreiben
35.6	REFPT_EN	BOOL	FALSE	1 = Bezugspunkt setzen
35.7	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.0	FVAL_EN	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fliegendes Istwert setzen

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
36.1	ZOFF_EN	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Nullpunktverschiebung setzen
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben
36.3	TRG255_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 255 schreiben
36.4	DELDIAG_EN	BOOL	FALSE	1 = Diagnosepuffer löschen
Anstoßbits für Leseaufträge				
36.5	MDRD_EN	BOOL	FALSE	1 = Maschinendaten lesen
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaßnummer 1 … 50) lesen
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaßnummer 51 … 100) lesen
37.0	MSRRD_EN	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Messwerte lesen
37.1	ACTSPD_EN	BOOL	FALSE	1 = Istgeschwindigkeit, Restweg und aktuelles Schrittmaß lesen
37.2	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
Fertigbits für Funktionsschalter				
38.0	PLOOP_D	BOOL	FALSE	1 = Auftrag "Schleifenfahrt in Richtung plus" abgeschlossen
38.1	MLOOP_D	BOOL	FALSE	1 = Auftrag "Schleifenfahrt in Richtung minus" abgeschlossen
38.2	EI_D	BOOL	FALSE	1 = Auftrag "Freigabeeingang nicht auswerten" abgeschlossen
38.3	EDGE_D	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Auftrag "Kantenerfassung ein" abgeschlossen
38.4	MSR_D	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Auftrag "Längenmessung ein" abgeschlossen
Fertigbits für Schreibaufträge				
39.0	MDWR_D	BOOL	FALSE	1 = Auftrag "Maschinendaten schreiben" abgeschlossen
39.1	MD_D	BOOL	FALSE	1 = Auftrag "Maschinendaten aktivieren" abgeschlossen
39.2	DELDIST_D	BOOL	FALSE	1 = Auftrag "Restweg löschen" abgeschlossen
39.3	AVALREM_D	BOOL	FALSE	1 = Auftrag "Istwert setzen rückgängig" abgeschlossen
39.4	TRGL1WR_D	BOOL	FALSE	1 = Auftrag "Schrittmaßtabelle 1 schreiben" abgeschlossen
39.5	TRGL2WR_D	BOOL	FALSE	1 = Auftrag "Schrittmaßtabelle 2 schreiben" abgeschlossen
39.6	REFPT_D	BOOL	FALSE	1 = Auftrag "Bezugspunkt setzen" abgeschlossen
39.7	AVAL_D	BOOL	FALSE	1 = Auftrag "Istwert setzen" abgeschlossen

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
40.0	FVAL_D	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Auftrag "Fliegendes Istwert setzen" abgeschlossen
40.1	ZOFF_D	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Auftrag "Nullpunktverschiebung setzen" abgeschlossen
40.2	TRG252_254_D	BOOL	FALSE	1 = Auftrag "Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben" abgeschlossen
40.3	TRG255_D	BOOL	FALSE	1 = Auftrag "Schrittmaß für Schrittmaßnummer 255 schreiben" abgeschlossen
40.4	DELDIAG_D	BOOL	FALSE	1 = Auftrag "Diagnosepuffer löschen" abgeschlossen
Fertigbits für Leseaufträge				
40.5	MDRD_D	BOOL	FALSE	1 = Auftrag "Maschinendaten lesen" abgeschlossen
40.6	TRGL1RD_D	BOOL	FALSE	1 = Auftrag "Schrittmaßtabelle 1 lesen" abgeschlossen
40.7	TRGL2RD_D	BOOL	FALSE	1 = Auftrag "Schrittmaßtabelle 2 lesen" abgeschlossen
41.0	MSRRD_D	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Auftrag "Messwerte lesen" abgeschlossen
41.1	ACTSPD_D	BOOL	FALSE	1 = Auftrag "Istgeschwindigkeit, Restweg und aktuelles Schrittmaß lesen" abgeschlossen
41.2	ENCVAL_D	BOOL	FALSE	1 = Auftrag "Geberwerte lesen" abgeschlossen
Fehlerbits für Funktionsschalter				
42.0	PLOOP_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schleifenfahrt in Richtung plus"
42.1	MLOOP_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schleifenfahrt in Richtung minus"
42.2	EI_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Freigabeeingang nicht auswerten"
42.3	EDGE_ERR	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fehler beim Auftrag "Kantenerfassung ein"
42.4	MSR_ERR	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fehler beim Auftrag "Längenmessung ein"
Fehlerbits für Schreibaufträge				
43.0	MDWR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten schreiben"
43.1	MD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten aktivieren"
43.2	DELDIST_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Restweg löschen"
43.3	AVALREM_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Istwert setzen rückgängig"
43.4	TRGL1WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaßtabelle 1 schreiben"
43.5	TRGL2WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaßtabelle 2 schreiben"

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
43.6	REFPT_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Bezugspunkt setzen"
43.7	AVAL_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Istwert setzen"
44.0	FVAL_ERR	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fehler beim Auftrag "Fliegendes Istwert setzen"
44.1	ZOFF_ERR	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fehler beim Auftrag "Nullpunktverschiebung setzen"
44.2	TRG252_254_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben"
44.3	TRG255_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaß für Schrittmaßnummer 255 schreiben"
44.4	DELDIAG_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Diagnosepuffer löschen"
Fehlerbits für Leseaufträge				
44.5	MDRD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten lesen"
44.6	TRGL1RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaßtabelle 1 lesen"
44.7	TRGL2RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Schrittmaßtabelle 2 lesen"
45.0	MSRRD_ERR	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: 1 = Fehler beim Auftrag "Messwerte lesen"
45.1	ACTSPD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Istgeschwindigkeit, Restweg und aktuelles Schrittmaß lesen"
45.2	ENCVAL_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "aktuelle Geberwerte lesen"
Auftragsverwaltung für die Anweisung ABS_CTRL bzw. ABS_CTRL_451				
48.0	JOB_ERR	INT	0	Fehlernummer des Kommunikationsfehlers
50.0	JOBBUSY	BOOL	FALSE	1 = mindestens ein Auftrag läuft
50.1	JOBRESET	BOOL	FALSE	1 = alle Fehler- und Fertigbits zurücksetzen
Daten für Auftrag "Nullpunktverschiebung" (FM 451)				
80.0	ZOFF	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Nullpunktverschiebung
Daten für Auftrag "Istwert setzen"				
84.0	AVAL	DINT	L#0	Koordinate für "Istwert setzen"
Daten für Auftrag "Fliegendes Istwert setzen" (FM 451)				
88.0	FVAL	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Koordinate für "Fliegendes Istwert setzen"
Daten für Auftrag "Bezugspunkt setzen"				
92.0	REFPT	DINT	L#0	Koordinate für "Bezugspunkt setzen"
Daten für Auftrag "Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben"				
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 252 oder 254
Daten für Auftrag "Schrittmaß für Schrittmaßnummer 255 schreiben"				
100.0	TRG255	DINT	L#0	Schrittmaß für Schrittmaßnummer 255
104.0	CHGDIF255	DINT	L#0	Umschaltdifferenz für Schrittmaßnummer 255

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
108.0	CUTDIF255	DINT	L#0	Abschaltdifferenz für Schrittmaßnummer 255
Daten für Auftrag "Positionsdaten lesen"				
112.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
116.0	DIST_TO_GO	DINT	L#0	Restweg
120.0	ACT_TRG	DINT	L#0	Aktuelles Schrittmaß
Daten für Auftrag "Geberdaten lesen"				
124.0	ENCVAL	DINT	L#0	Geberistwert (interne Darstellung)
128.0	ZEROVAL	DINT	L#0	Letzter Nullmarkenwert (interne Darstellung)
132.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage
Daten für Auftrag "Längenmessung / Kantenerfassung" (FM 451)				
136.0	BEG_VAL	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Anfangswert der Längenmessung / Kantenerfassung
140.0	END_VAL	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Endwert der Längenmessung / Kantenerfassung
144.0	LEN_VAL	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Länge

Parameter-Datenbaustein

Aufgabe

Wenn Sie die Maschinendaten und Schrittmaßtabellen im Betrieb verändern wollen, benötigen Sie einen Parameter-DB, in dem diese Daten abgelegt sind. Die Parameter können vom Anwenderprogramm oder von einem B&B-System verändert werden.

Die in der Parametrieroberfläche angezeigten Daten können Sie in einen Parameter-DB exportieren. Einen Parameter-DB können Sie auch in die Parametrieroberfläche importieren und dort anzeigen lassen.

Zu jedem Kanal der Baugruppe kann es mehrere Sätze von Parametrierdaten geben, z. B. für verschiedene Rezepte, auf die Sie programmgesteuert umschalten können.

Aufbau

Parameter-DB
Maschinendaten
Schrittmaßtabellen

Inhalt

Die folgende Tabelle zeigt den Inhalt des Parameter-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
Maschinendaten				
4.0	EDGEDIST	DINT	L#0	• FM 351: 0 (nicht verwendet) • FM 451: Minimaler Kantenabstand
8.0	UNITS	DINT	L#1	Maßsystem
12.0	AXIS_TYPE	DINT	L#0	0 = Linearachse, 1 = Rundachse
16.0	ENDROTAX	DINT	L#100 000	Ende der Rundachse
20.0	ENC_TYPE	DINT	L#1	Gebertyp, Telegrammlänge
24.0	DISP_REV	DINT	L#80 000	Weg pro Geberumdrehung
32.0	INC_REV	DINT	L#500	Inkremente pro Geberumdrehung
36.0	NO_REV	DINT	L#1	Anzahl Geberumdrehungen
40.0	BAUDRATE	DINT	L#0	Baudrate
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
48.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage
52.0	REFPT_TYPE	DINT	L#0	Art der Referenzpunktfahrt
59.0	CNT_DIR	BOOL	FALSE	Zählrichtung: • 0 = normal • 1 = invertiert
63.0	MON_WIRE	BOOL	TRUE	1 = Überwachung Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1 = Überwachung Telegrammfehler
63.2	MON_PULSE	BOOL	TRUE	1 = Überwachung Fehlimpulse
64.0	SSW_STRT	DINT	L#-100 000 000	Software-Endschalter Anfang
68.0	SSW_END	DINT	L#100 000 000	Software-Endschalter Ende
76.0	TRG_RANGE	DINT	L#1000	Zielbereich
80.0	MON_TIME	DINT	L#2000	Überwachungszeit [ms]
84.0	ZSPEED_R	DINT	L#1000	Stillstandsbereich
88.0	ZSPEED_L	DINT	L#30 000	Obergrenze der Stillstandsgeschwindigkeit
92.0	CTRL_TYPE	DINT	L#1	Ansteuerart (1 - 4)
99.0	REFPT_SPD	BOOL	TRUE	Startgeschwindigkeit bei Referenzpunktfahrt: • 0 = Eilgang • 1 = Schleichgang
99.1	EI_TYPE	BOOL	FALSE	• FM 351: 0 (nicht verwendet) • FM 451: Freigabeeingang: – 0 = Pegelgesteuert – 1 = Flankengesteuert

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus
Schrittmaßtabelle 1				
120.0	TRGL1.TRG[1]	DINT	L#0	Schrittmaßnummer 1
. . .	. . .	. . .	. . .	. . .
316.0	TRGL1.TRG[50]	DINT	L#0	Schrittmaßnummer 50
Schrittmaßtabelle 2				
320.0	TRGL2.TRG[51]	DINT	L#0	Schrittmaßnummer 51
. . .	. . .	. . .	. . .	. . .
516.0	TRGL2.TRG[100]	DINT	L#0	Schrittmaßnummer 100

Diagnose-Datenbaustein

Aufgabe

Der Diagnose-DB ist die Datenablage für die Anweisung ABS_DIAG (Seite 6145) bzw. ABS_DIAG_451 (Seite 6149). Er enthält den von dieser Anweisung aufbereiteten Diagnosepuffer der Baugruppe.

Aufbau

Diagnose-DB
Baugruppenadresse
Interne Daten
Auftragsstatus
Anstoßbit
Aufbereiteter Diagnosepuffer

Inhalt

Die folgende Tabelle zeigt den Inhalt des Diagnose-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
0.0	MOD_ADDR	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Fehlernummer des Kommunikationsfehlers
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten vierter Eintrag
302.0	DIAG[5]	STRUCT		Diagnosedaten fünfter Eintrag
312.0	DIAG[6]	STRUCT		Diagnosedaten sechster Eintrag
322.0	DIAG[7]	STRUCT		Diagnosedaten siebter Eintrag
332.0	DIAG[8]	STRUCT		Diagnosedaten achter Eintrag
342.0	DIAG[9]	STRUCT		Diagnosedaten neunter Eintrag

Struktur des Diagnoseeintrags

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 2: Bedienfehler 4: Datenfehler 5: Maschinendatenfehler 6: Schrittmaßtabellen-Fehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer
+6.0	CH_NO	INT	0	Kanalnummer
+8.0	TRG_NO	INT	0	Schrittmaßnummer

Auftragsverwaltung (ABS_CTRL)

Aufträge

Der über die Steuersignale und Rückmeldesignale hinausgehende Datenaustausch mit der Baugruppe wird über Aufträge abgewickelt.

Um einen Auftrag abzugeben, setzen Sie das entsprechende Anstoßbit im Kanal-DB und bei Schreibaufträgen noch die entsprechenden Daten. Rufen Sie dann die Anweisung ABS_CTRL (Seite 6143) auf, um den Auftrag auszuführen.

Wenn Sie die FM 351 zentral einsetzen, benötigt ein Leseauftrag genau einen Zyklus. Wenn Sie die FM 351 dezentral einsetzen, kann ein Leseauftrag mehrere Zyklen benötigen.

Ein Schreibauftrag benötigt wegen der erforderlichen Quittungen der Baugruppe mindestens 3 Aufrufe bzw. OB-Zyklen.

Ist ein Auftrag fertig bearbeitet, nimmt die Anweisung das Anstoßbit zurück. Beim nächsten Aufruf der Anweisung wird der folgende Auftrag ermittelt und ausgeführt.

Zu jedem Auftrag gibt es neben dem Anstoßbit mit der Endung _EN wie "enable" auch ein Fertigbit und ein Fehlerbit. Diese haben im Namen die Endung _D wie "done" bzw. _ERR wie "error". Die Anweisung ABS_CTRL aktualisiert die Fertig- und Fehlerbits, wenn die Bearbeitung eines Auftrags beendet ist. Diese Bits sollten Sie nach der Auswertung oder vor Abgabe eines Auftrags auf 0 setzen.

Wenn Sie das Bit JOBRESET setzen, werden vor der Bearbeitung der anstehenden Aufträge alle Fertig- und Fehlerbits zurückgesetzt. Das Bit JOBRESET wird anschließend wieder auf 0 gesetzt.

Funktionsschalter

Die Funktionsschalter schalten Zustände des Kanals ein und aus. Ein Auftrag zum Schreiben der Funktionsschalter wird nur bei einer Änderung einer Schalterstellung ausgeführt. Die Stellung des Funktionsschalters bleibt nach der Ausführung des Auftrags erhalten.

Funktionsschalter und Aufträge können Sie bei einem Aufruf der Anweisung ABS_CTRL gleichzeitig verwenden.

Zu den Funktionsschaltern gibt es wie bei den Aufträgen Anstoßbits mit der Namensendung _ON/_OFF, Fertigbits mit der Namensendung _D und Fehlerbits mit der Namensendung _ERR.

Um die Fertigbits und Fehlerbits der Funktionsschalter auswerten zu können, sollten Sie diese Bits auf 0 setzen, bevor Sie einen Auftrag zum Ändern eines Funktionsschalters abgeben.

Reihenfolge der Auftragsabarbeitung

Sie können mehrere Aufträge gleichzeitig abgeben. Wenn keine Aufträge aktiv sind, sucht die Auftragsverwaltung der Anweisung ABS_CTRL ab Auftrag MDWR_EN, ob Anstoßbits gesetzt sind oder Änderungen an Funktionsschaltern vorgenommen wurden. Ist ein Auftrag gefunden, wird dieser bearbeitet. Ist der Auftrag abgeschlossen, sucht die Auftragsverwaltung nach dem nächsten zu bearbeitenden Auftrag. Ist der letzte Auftrag ENCVAL_EN durchsucht worden, wird wieder bei Auftrag MDWR_EN nachgesehen. Diese Suche wird so lange wiederholt, bis alle Aufträge abgearbeitet sind.

Die Aufträge werden in folgender technologisch sinnvoller Reihenfolge bearbeitet:

Reihenfolge	Adresse im Kanal-DB	Parameter	Beschreibung	Rückgesetzt von
Schreibaufträge				
1	35.0	MDWR_EN	Maschinendaten schreiben	Anweisung

Reihenfolge	Adresse im Kanal-DB	Parameter	Beschreibung	Rückgesetzt von
2	35.1 35.2 35.3 36.4	MD_EN DELDIST_EN AVALREM_EN DELDIAG_EN	Maschinendaten aktivieren Restweg löschen Istwert setzen rückgängig Diagnosepuffer löschen	Anweisung
3	35.4	TRGL1WR_EN	Schrittmaßtabelle 1 schreiben	Anweisung
4	35.5	TRGL2WR_EN	Schrittmaßtabelle 2 schreiben	Anweisung
5	35.6	REFPT_EN	Bezugspunkt setzen	Anweisung
6	 34.0 34.1 34.2	Funktionsschalter: PLOOP_ON MLOOP_ON EI_OFF	 Schleifenfahrt in Richtung plus Schleifenfahrt in Richtung minus Freigabeeingang nicht auswerten	Anwenderprogramm
7	35.7	AVAL_EN	Istwert setzen	Anweisung
10	36.2	TRG252_254_EN	Schrittmaß für Schrittmaßnummer 254 schreiben	Anweisung
11	36.3	TRG255_EN	Schrittmaß für Schrittmaßnummer 255 schreiben	Anweisung
Leseaufträge				
12	36.5	MDRD_EN	Maschinendaten lesen	Anweisung
13	36.6	TRGL1RD_EN	Schrittmaßtabelle 1 lesen	Anweisung
14	36.7	TRGL2RD_EN	Schrittmaßtabelle 2 lesen	Anweisung
16	37.1	ACTSPD_EN	Aktuelle Geschwindigkeit, Restweg und aktuelles Schrittmaß lesen	Anweisung
17	37.2	ENCVAL_EN	Geberdaten lesen	Anweisung

Diese Reihenfolge ermöglicht Ihnen, eine Positionierung mit einem Satz von Aufträgen und Steuersignalen vollständig anzustoßen. Die Aufträge gehen vom Schreiben und Aktivieren der Maschinendaten über die Einstellung des externen Freigabeeingangs bis zum Schreiben der Schrittmaße für die Schrittmaßfahrten.

Aufträge während einer laufenden Positionierung

Die in der folgenden Tabelle aufgeführten Schreibaufträge werden, wenn Sie während einer Positionierung abgegeben werden, bis zum Ende der Positionierung zurückgehalten und erst bei dem dann folgenden Aufruf der Anweisung durchgeführt.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
34.0	PLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung plus
34.1	MLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung minus
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
35.2	DELDIST_EN	BOOL	FALSE	1 = Restweg löschen
35.3	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen rückgängig
35.6	REFPT_EN	BOOL	FALSE	1 = Bezugspunktkoordinate setzen
35.7	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.4	DELDIAG_EN	BOOL	FALSE	1 = Diagnosepuffer löschen

Auftragsstatus

Den Status der Auftragsbearbeitung können Sie am Rückgabewert RETVAL und am Tätigbit JOBBUSY im Kanal-DB ablesen. Den Status eines einzelnen Auftrags können Sie anhand der Anstoßbits, Fertigbits und Fehlerbits dieses Auftrags auswerten.

	RETVAL	JOBBUSY	Anstoßbit _EN	Fertigbit _D	Fehlerbit _ERR
Auftrag aktiv	1	1	1	0	0
Auftrag fertig ohne Fehler	0	0	0	1	0
Auftrag fertig mit Fehler	-1	0	0	1	1
Schreibauftrag abgebrochen	-1	0	0	0	1

Auftragsverwaltung (ABS_CTRL_451)

Aufträge

Der über die Steuersignale und Rückmeldesignale hinausgehende Datenaustausch mit der Baugruppe wird über Aufträge abgewickelt.

Um einen Auftrag abzugeben, setzen Sie das entsprechende Anstoßbit im Kanal-DB und bei Schreibaufträgen noch die entsprechenden Daten. Rufen Sie dann die Anweisung ABS_CTRL_451 (Seite 6147) auf, um den Auftrag auszuführen.

Ein Schreibauftrag benötigt wegen der erforderlichen Quittungen der Baugruppe mindestens 3 Aufrufe bzw. OB-Zyklen. Ein Leseauftrag wird sofort ausgeführt.

Ist ein Auftrag fertig bearbeitet, nimmt die Anweisung das Anstoßbit zurück. Beim nächsten Aufruf der Anweisung wird der folgende Auftrag ermittelt und ausgeführt.

Zu jedem Auftrag gibt es neben dem Anstoßbit mit der Endung _EN wie "enable" auch ein Fertigbit und ein Fehlerbit. Diese haben im Namen die Endung _D wie "done" bzw. _ERR wie "error". Die Anweisung ABS_CTRL_451 aktualisiert die Fertig- und Fehlerbits, wenn die Bearbeitung eines Auftrags beendet ist. Diese Bits sollten Sie nach der Auswertung oder vor Abgabe eines Auftrags auf 0 setzen.

Wenn Sie das Bit JOBRESET setzen, werden vor der Bearbeitung der anstehenden Aufträge alle Fertig- und Fehlerbits zurückgesetzt. Das Bit JOBRESET wird anschließend wieder auf 0 gesetzt.

Funktionsschalter

Die Funktionsschalter schalten Zustände des Kanals ein und aus. Ein Auftrag zum Schreiben der Funktionsschalter wird nur bei einer Änderung einer Schalterstellung ausgeführt. Die Stellung des Funktionsschalters bleibt nach der Ausführung des Auftrags erhalten.

Funktionsschalter und Aufträge können Sie bei einem Aufruf der Anweisung ABS_CTRL_451 gleichzeitig verwenden.

Zu den Funktionsschaltern gibt es wie bei den Aufträgen Anstoßbits mit der Namensendung _ON/_OFF, Fertigbits mit der Namensendung _D und Fehlerbits mit der Namensendung _ERR.

Um die Fertigbits und Fehlerbits der Funktionsschalter auswerten zu können, sollten Sie diese Bits auf 0 setzen, bevor Sie einen Auftrag zum Ändern eines Funktionsschalters abgeben.

Reihenfolge der Auftragsabarbeitung

Sie können mehrere Aufträge gleichzeitig abgeben. Wenn keine Aufträge aktiv sind, sucht die Auftragsverwaltung der Anweisung ABS_CTRL_451 ab Auftrag MDWR_EN, ob Anstoßbits gesetzt sind oder Änderungen an Funktionsschaltern vorgenommen wurden. Ist ein Auftrag gefunden, wird dieser bearbeitet. Ist der Auftrag abgeschlossen, sucht die Auftragsverwaltung nach dem nächsten zu bearbeitenden Auftrag. Ist der letzte Auftrag ENCVAL_EN durchsucht worden, wird wieder bei Auftrag MDWR_EN nachgesehen. Diese Suche wird so lange wiederholt, bis alle Aufträge abgearbeitet sind.

Die Aufträge werden in folgender technologisch sinnvoller Reihenfolge bearbeitet:

Reihenfolge	Adresse im Kanal-DB	Parameter	Beschreibung	Rückgesetzt von
Schreibaufträge				
1	35.0	MDWR_EN	Maschinendaten schreiben	Anweisung
2	35.1 35.2 35.3 36.4	MD_EN DELDIST_EN AVALREM_EN DELDIAG_EN	Maschinendaten aktivieren Restweg löschen Istwert setzen rückgängig Diagnosepuffer löschen	Anweisung
3	35.4	TRGL1WR_EN	Schrittmaßtabelle 1 schreiben	Anweisung
4	35.5	TRGL2WR_EN	Schrittmaßtabelle 2 schreiben	Anweisung
5	35.6	REFPT_EN	Bezugspunkt setzen	Anweisung
6	 34.0 34.1 34.2 34.3 34.4	Funktionsschalter: PLOOP_ON MLOOP_ON EI_OFF EDGE_ON MSR_ON	 Schleifenfahrt in Richtung plus Schleifenfahrt in Richtung minus Freigabeeingang nicht auswerten Kantenerfassung ein Längenmessung ein	Anwenderpro-gramm
7	35.7	AVAL_EN	Istwert setzen	Anweisung
8	36.0	FVAL_EN	Fliegendes Istwert setzen	Anweisung
9	36.1	ZOFF_EN	Nullpunktverschiebung setzen	Anweisung
10	36.2	TRG252_254_EN	Schrittmaß für Schrittmaßnummer 252/254 schreiben	Anweisung
11	36.3	TRG255_EN	Schrittmaß für Schrittmaßnummer 255 schreiben	Anweisung
Leseaufträge				
12	36.5	MDRD_EN	Maschinendaten lesen	Anweisung
13	36.6	TRGL1RD_EN	Schrittmaßtabelle 1 lesen	Anweisung
14	36.7	TRGL2RD_EN	Schrittmaßtabelle 2 lesen	Anweisung
15	37.0	MSRRD_EN	Messwerte lesen	Anweisung
16	37.1	ACTSPD_EN	Aktuelle Geschwindigkeit, Restweg und aktu-elles Schrittmaß lesen	Anweisung
17	37.2	ENCVAL_EN	Geberdaten lesen	Anweisung

Diese Reihenfolge ermöglicht Ihnen, eine Positionierung mit einem Satz von Aufträgen und Steuersignalen vollständig anzustoßen. Die Aufträge gehen vom Schreiben und Aktivieren der Maschinendaten über die Einstellung des externen Freigabeeingangs bis zum Schreiben der Schrittmaße für die Schrittmaßfahrten.

Aufträge während einer laufenden Positionierung

Die in der folgenden Tabelle aufgeführten Schreibaufträge werden, wenn Sie während einer Positionierung abgegeben werden, bis zum Ende der Positionierung zurückgehalten und erst bei dem dann folgenden Aufruf der Anweisung durchgeführt.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
34.0	PLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung plus
34.1	MLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung minus
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten
34.3	EDGE_ON	BOOL	FALSE	1 = Kantenerfassung ein
34.4	MSR_ON	BOOL	FALSE	1 = Längenmessung ein
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
35.2	DELDIST_EN	BOOL	FALSE	1 = Restweg löschen
35.3	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen rückgängig
35.6	REFPT_EN	BOOL	FALSE	1 = Bezugspunktkoordinate setzen
35.7	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.1	ZOFF_EN	BOOL	FALSE	1 = Nullpunktverschiebung setzen
36.4	DELDIAG_EN	BOOL	FALSE	1 = Diagnosepuffer löschen

Auftragsstatus

Den Status der Auftragsbearbeitung können Sie am Rückgabewert RET_VAL und am Tätigbit JOBBUSY im Kanal-DB ablesen. Den Status eines einzelnen Auftrags können Sie anhand der Anstoßbits, Fertigbits und Fehlerbits dieses Auftrags auswerten.

	RET_VAL	JOBBUSY	Anstoßbit _EN	Fertigbit _D	Fehlerbit _ERR
Auftrag aktiv	1	1	1	0	0
Auftrag fertig ohne Fehler	0	0	0	1	0
Auftrag fertig mit Fehler	-1	0	0	1	1
Schreibauftrag abgebrochen	-1	0	0	0	1

Fehlerklassen

Fehlerklasse 1: Betriebsfehler

Beschreibung

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt. Die Betriebsfehler führen zum Abbruch der Positionierung, außer bei Fehlernummer 9. Fehlernummer 9 führt zum Absteuern der Positionierung.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
2	**Software-Endschalter Ende überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
3	**Verfahrbereichsanfang überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
4	**Verfahrbereichsende überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
5	**Fehler beim Zieleinlauf**		ja
	Ursache	Zielbereich wurde innerhalb der Überwachungszeit nicht erreicht.	
6	**Stillstandsbereich verlassen**		ja
	Ursache	Der Istwert liegt außerhalb des Stillstandsbereichs.	
7	**Mitkopplung**		ja
	Ursache	Istwertänderung > ½ Stillstandsbereich in die falsche Richtung.	
8	**Fehlende oder zu geringe Istwertänderung**		ja
	Ursache	keine Istwertänderung oder eine Istwertänderung entgegen der Sollrichtung innerhalb der Überwachungszeit.	
9	**Ziel überfahren (FM 451)**		ja
	Ursache	Beim "Fliegendes Istwert setzen" wurde das Ziel überfahren.	
10	**Zielbereich überfahren**		ja
	Ursache	Zielbereich wurde nach Zieleinlauf überfahren.	
11	**Umschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Umschaltpunkt.	
12	**Abschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Abschaltpunkt/Umkehrpunkt.	
13	**Zielbereichsanfang fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Zielbereich.	
14	**Änderung größer als der halbe Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwertsprünge.	

Nr.	Bedeutung		Diagnosealarm
15	**Änderung größer als der Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwertsprünge.	
16	**Schrittmaß für Schrittmaßnummer 252 nicht übertragen (FM 451)**		ja
	Ursache	Das Schrittmaß wurde nicht übertragen.	
17	**Schrittmaß für Schrittmaßnummer 252 nicht anfahrbar (FM 451)**		ja
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Schrittmaß ist kleiner als die Umschalt- bzw. Abschaltdifferenz.	
18	**Falsches Schrittmaß für Schrittmaßnummer 252 (FM 451)**		ja
	Ursache	Das Schrittmaß liegt außerhalb des Arbeitsbereichs.	

Fehlerklasse 2: Bedienfehler

Beschreibung

Bedienfehler werden erkannt bei Änderung der Steuerungssignale im Nutzdatenbereich. Die Bedienfehler führen zum Absteuern der Positionierung.

Nr.	Bedeutung		Diagnosealarm
1	**Unzulässige Betriebsart**		nein
	Ursache	Die angewählte Betriebsart ist unzulässig.	
3	**Unzulässiger Schnittstellenauftrag**		nein
	Ursache	Das angewählte Signal ist mit dieser Betriebsart unzulässig.	
4	**Falscher Betriebsartenparameter**		nein
	Ursache	• Bei der Betriebsart "Tippen" ist die Geschwindigkeitsvorgabe ungleich der Eil- oder Schleichgeschwindigkeit. • Bei der Betriebsart "Schrittmaßfahrt" ist das Schrittmaß ungleich 1 bis 100 oder ungleich 252, 254 und 255.	
5	**Startfreigabe nicht vorhanden**		nein
	Ursache	Startfreigabe beim Starten nicht vorhanden.	
7	**Ziel / Zielbereich liegt außerhalb des Arbeitsbereichs**		nein
	Ursache	Vorgegebenes bzw. berechnetes Ziel liegt außerhalb der Software-Endschalter.	
8	**Achse nicht parametriert**		nein
	Ursache	Falsche oder keine Maschinendaten wurden für die Achse parametriert.	
9	**Achse nicht synchronisiert**		nein
	Ursache	Die Betriebsart "Schrittmaßfahrt" ist nur mit einer bereits synchronisierten Achse möglich.	
10	**Ziel / Wegstück nicht positionierbar**		nein
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Ziel ist kleiner als die Abschaltdifferenz.	

Nr.	Bedeutung		Diagnosealarm
17	**Referenzpunktfahrt nicht möglich**		nein
	Ursache	Es wurde ein SSI-Geber angeschlossen.	
18	**Schrittmaßfahrt relativ oder absolut nicht möglich**		nein
	Ursache	Das Schrittmaß ist ungültig.	
19	**Abschaltdifferenz nicht größer als ½ Zielbereich bei Schrittmaßnummer 255**		nein
	Ursache	Die Abschaltdifferenz für das Schrittmaß 255 ist kleiner als der halbe Zielbereich.	
20	**Fahrt in die vorgegebene Richtung unzulässig**		nein
	Ursache	Der Abstand zum Software-Endschalter ist zu gering.	

Fehlerklasse 4: Datenfehler

Beschreibung

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt. Die Datenfehler führen zu keiner Fehlerreaktion.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück / Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
10	**Fehlerhafte Nullpunktverschiebung**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m... +1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten-Aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen (fehlerfreien) Maschinendaten auf der Baugruppe vorhanden.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	

Nr.	Bedeutung		Diagnosealarm
29	**Unzulässige Bitcodierung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und einer Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereichs liegen.	
36	**Fehlerhafte Umschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
37	**Fehlerhafte Abschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Die Abschaltdifferenz muss kleiner als die Umschaltdifferenz sein.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind entweder keine Maschinendaten vorhanden oder sie sind nicht aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Einer der Aufträge "Istwert setzen" und "fliegendes Istwert setzen" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	

Fehlerklasse 5: Maschinendatenfehler

Beschreibung

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst. Die Maschinendatenfehler führen zu keiner Fehlerreaktion.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	
6	**Falscher minimaler Kantenabstand (FM 451)**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben.	
7	**Falsches Maßsystem**		ja
	Ursache	Der Wert für das Maßsystem liegt außerhalb des zulässigen Bereichs von 1 bis 4 und 6.	
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben.	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	

Nr.	Bedeutung		Diagnosealarm
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 4.	
11	**Falscher Weg pro Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Inkremente pro Geberumdrehung**		ja
14	**Falsche Anzahl Umdrehungen**		ja
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereiches von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m (je nach Auflösung). Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Geber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art der Referenzpunktfahrt**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 2 und 3 angegeben.	
19	**Falsche Zählrichtung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung Telegrammfehler im Parameter-DB auf "FALSE" gesetzt. Beim verwendeten Geber ist die Überwachung von Fehlimpulsen nicht möglich. Deaktivieren Sie den Parameter MON_PULSE.	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw. -1000 m (je nach Auflösung).	
22	**Falscher Software-Endschalter Ende**		ja
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Linearachse: Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
23	**Falsche Maximalgeschwindigkeit**		ja
	Ursache	Die nicht aufgelisteten Daten im Parameter-DB müssen 0 sein.	

Nr.	Bedeutung		Diagnosealarm
24	**Falscher Zielbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	
25	**Falsche Überwachungszeit**		ja
	Ursache	Der Wert für die Überwachungszeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 ms.	
26	**Falscher Stillstandsbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	
127	**Falsche Stillstandsgeschwindigkeit**		ja
	Ursache	Der Wert für die Stillstandsgeschwindigkeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 µm/min.	
128	**Falsche Ansteuerart**		ja
	Ursache	Sie haben für die Ansteuerart einen Wert außerhalb des zulässigen Bereiches von 1 bis 4 angegeben.	
129	**Falsche Startgeschwindigkeit für die Referenzpunktfahrt**		ja
	Ursache	Sie haben als Startgeschwindigkeit weder 0 noch 1 angegeben.	
130	**Falsche Umschaltdifferenz in Richtung +**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
131	**Falsche Umschaltdifferenz in Richtung –**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
132	**Falsche Abschaltdifferenz in Richtung +**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz plus, kleiner als ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
133	**Falsche Abschaltdifferenz in Richtung –**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz minus, kleiner als ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben eine Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Geber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Geber und Rundachse: Der Geber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Geber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Fehlerklasse 6: Schrittmaßtabellen-Fehler

Beschreibung

Die Schrittmaßtabellen-Fehler führen zu keiner Fehlerreaktion.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß in der Schrittmaßtabelle zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück / Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	

Fehlerklasse 15: Meldungen

Beschreibung

Die Meldungen führen zu keiner Fehlerreaktion.

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	
11	**Abstand zum Umschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
12	**Abstand zum Umkehrpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
14	**Abstand zum Abschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
15	**Abstand zum Zielbereichsanfang zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	

Fehlerklasse 128: Diagnosefehler

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	• Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen • Sicherung auf der Baugruppe ist defekt • Unterspannung • Masse-Drahtbruch • Kurzschluss (z. B. am angeschlossenen Geber)	
	Wirkung	• Die Positionierung wird auf allen Kanäle abgebrochen • Abschalten der Ausgänge • Löschen der Synchronisation bei Inkrementalgebern, wenn die Hilfsspannung für Geberversorgung fehlt • Die Baugruppe ist nicht parametriert. • Löschen der Startfreigabe	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
5	**Frontstecker fehlt (FM 451)**		ja
	Ursache	Fronstecker der Positionierbaugruppe ist nicht gesteckt.	
	Wirkung	• Externe Hilfsspannung 24 V fehlt • Baugruppe ist nicht betriebsbereit	
	Behebung	Frontstecker auf die Positionierbaugruppe stecken	
51	**Zeitüberwachung angesprochen (Watchdog)**		ja
	Ursache	• Starke Störeinflüsse auf die Baugruppe • Fehler in der Baugruppe	
	Wirkung	• Baugruppe wird rückgesetzt • Abschalten aller Ausgänge • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich bitte an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind. • Tauschen der Baugruppe	

Nr.	Bedeutung		Diagnosealarm
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale • Flankenfehler der Gebersignale • Maximale Eingangsfrequenz des Gebereingangs überschritten • Ausfall der Geberversorgung	
	Wirkung	• Die Positionierung wird abgebrochen • Abschalten aller Ausgänge • Löschen der Synchronisation bei Inkrementalgebern • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit der Parametrieroberfläche vorübergehend unter Verantwortung des Betreibers ausgeblendet werden. • Technische Daten der Baugruppe einhalten	
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen der Baugruppe und dem Absolutwertgeber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • Falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt • Monoflopzeit des Gebers größer als 64 µs	
	Wirkung	• Die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und Baugruppe	

Nr.	Bedeutung		Diagnosealarm
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• Die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben (Parametrierdialog) • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit der Parametrieroberfläche vorübergehend unter Verantwortung des Betreibers ausgeblendet werden.	

FM x52 Nockensteuern

CAM_INIT

Beschreibung

Die Anweisung CAM_INIT initialisiert die folgenden Daten im Kanal-DB:

- Die Steuersignale
- Die Rückmeldesignale
- Die Anstoß-, Fertig-, Fehlerbits der Aufträge
- Die Funktionsschalter und ihre Fertig- und Fehlerbits
- Die Auftragsverwaltung und die internen Puffer für die Anweisung CAM_CTRL (Seite 6176) bzw. CAM_CTRL_452 (Seite 6179)
- Die Auftragsverwaltung und die internen Puffer für die Anweisung CAM_MSRM_452 (Seite 6183) (nur FM 452)

Aufruf

Die Anweisung muss nach einem Anlauf, d. h. nach Einschalten der Versorgungsspannung der Baugruppe bzw. der CPU durchlaufen werden. Bauen Sie sie deshalb z. B. in den Neustart-OB OB 100 und den Ziehen/Stecken-OB OB 83 ein oder rufen Sie sie in der Initialisierungsphase Ihres Anwenderprogramms auf. Zuvor muss im OB 100 die Baugruppenadresse in den DB geschrieben werden. Somit ist sichergestellt, dass Ihr Anwenderprogramm nach einem Neustart der CPU oder einem Baugruppenanlauf nicht auf veraltete Daten zugreift.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Kanal-DB

Rückgabewerte

Die Anweisung liefert keinen Rückgabewert.

CAM_CTRL

Beschreibung

Mit der Anweisung CAM_CTRL können Sie Betriebsdaten aus der Baugruppe lesen, die Baugruppe initialisieren und während des Betriebs steuern. Dazu verwenden Sie Steuersignale, Rückmeldesignale, Funktionsschalter sowie Schreib- und Leseaufträge.

Arbeitsweise

Bei jedem Aufruf führt die Anweisung folgende Tätigkeiten aus:

- Rückmeldesignale lesen:
  Die Anweisung CAM_CTRL liest alle Rückmeldesignale der Baugruppe und trägt sie in den Kanal-DB ein. Da die Steuersignale und Aufträge erst anschließend bearbeitet werden, geben die Rückmeldesignale den Status der Baugruppe vor dem Aufruf der Anweisung wieder.
- Steuersignale schreiben:
  Die Steuersignale, die im Kanal-DB eingetragen sind, werden zur Baugruppe übertragen. Die Freigabe der Nockenbearbeitung wird jedoch so lange zurückgehalten, wie der Anstoß für einen Auftrag "Bezugspunkt setzen" oder "Nockendaten schreiben" gesetzt ist. Das (Wieder-) Einschalten der Nockenbearbeitung wird für diese Zeitspanne zurückgehalten.
- Auftragsverwaltung (Seite 6195):
  Aus den im Kanal-DB eingetragenen Anstoßbits für Aufträge wird der jeweils nächste Auftrag ausgeführt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden.

Bevor Sie die Anweisung aufrufen, tragen Sie alle Daten in den Kanal-DB ein, die für die Ausführung der beabsichtigten Funktionen erforderlich sind.

Verwendete Datenbausteine

- Kanal-DB (Seite 6184):
  Der Kanal-DB ist der Instanz-DB der Anweisung CAM_CTRL. Im Kanal-DB muss die Baugruppenadresse eingetragen sein. .
- Parameter-DB (Seite 6191):
  Wenn Sie Maschinen- oder Nockendaten über Aufträge schreiben oder lesen wollen, benötigen Sie einen Parameter-DB, dessen Nummer im Kanal-DB eingetragen sein muss. Die Größe des Parameter-DB muss für die vorhandene Nockenanzahl ausreichend sein.

Rückgabewerte

Die Anweisung liefert im Parameter RETVAL des Kanal-DB folgende Rückgabewerte:

RETVAL	BIE	Beschreibung
1	1	Mindestens 1 Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler: Datenfehler (DAT_ERR) oder Kommunikationsfehler (JOB_ERR) aufgetreten

Anlauf

Rufen Sie beim Anlauf der Baugruppe bzw. der CPU die Anweisung CAM_INIT (Seite 6175) auf. Dabei werden u. a. auch die Funktionsschalter zurückgesetzt.

Die Anweisung CAM_CTRL quittiert den Anlauf der Baugruppe. Während dieser Zeit sind RETVAL und JOBBUSY = 1.

Verhalten im Fehlerfall

Wenn bei einem Schreibauftrag fehlerhafte Daten geschrieben wurden, liefert die Baugruppe die Rückmeldung DATA_ERR = 1. Wenn bei einem Schreib- oder Leseauftrag ein Fehler bei der Kommunikation mit der Baugruppe auftritt, wird die Fehlerursache im Parameter JOB_ERR im Kanal-DB abgelegt.

- Fehler bei einem Schreibauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt. Bei allen noch anstehenden Schreibaufträgen wird ebenfalls das Anstoßbit zurückgenommen und das Fehlerbit _ERR gesetzt.
  Die anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.
- Fehler bei einem Leseauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt.
  Die noch anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.

CAM_DIAG

Beschreibung

Mit der Anweisung CAM_DIAG lesen Sie den Diagnosepuffer der Baugruppe aus und stellen ihn für eine Anzeige im B&B-System oder für eine programmierte Auswertung zur Verfügung.

Arbeitsweise

Die Anweisung liest den Diagnosepuffer dann, wenn über das Rückmeldesignal DIAG = 1 ein neuer Eintrag im Diagnosepuffer angezeigt wird. Nach dem Lesen des Diagnosepuffers wird DIAG von der Baugruppe auf 0 gesetzt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden. Ein zusätzlicher Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig. Für die vollständige Ausführung der Anweisung sind mindestens 2 Aufrufe (Zyklen) notwendig.

Verwendeter Datenbaustein

Diagnose-DB (Seite 6194):

De Diagnose-DB ist der Instanz-DB der Anweisung CAM_DIAG. Im Diagnose-DB muss die Baugruppenadresse eingetragen sein. Der Eintrag sollte im Anlauf-OB (OB 100) bzw. vor der ersten Verwendung erfolgen. Der neueste Eintrag des Diagnosepuffers wird in der Struktur DIAG[1] und der älteste Eintrag in der Struktur DIAG[4] eingetragen.

Rückgabewerte

Die Anweisung liefert im Parameter RETVAL des Diagnose-DB folgende Rückgabewerte:

RETVAL	BIE	Beschreibung
1	1	Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler

Aufträge

Sie können den Diagnosepuffer unabhängig von einem neuen Eintrag lesen, wenn Sie das Anstoßbit DIAGRD_EN setzen. Nach dem Lesen des Diagnosepuffers wird das Anstoßbit auf 0 gesetzt.

Anlauf

Die Anweisung führt keine Anlaufbearbeitung durch.

Verhalten im Fehlerfall

Bei einem fehlerhaften Auftrag finden Sie die Fehlerursache im Diagnose-DB im Parameter JOB_ERR.

Fehlerklassen

Die auftretenden Fehler sind in folgenden Fehlerklassen zusammengefasst:

- Fehlerklasse 1: Betriebsfehler (Seite 6198)
- Fehlerklasse 4: Datenfehler (Seite 6199)
- Fehlerklasse 5: Maschinendatenfehler (Seite 6200)
- Fehlerklasse 7: Nockendatenfehler (Seite 6203)
- Fehlerklasse 15: Meldungen (Seite 6204)
- Fehlerklasse 128: Diagnosefehler (Seite 6204)

CAM_CTRL_452

Beschreibung

Mit der Anweisung CAM_CTRL_452 können Sie Betriebsdaten aus der Baugruppe lesen, die Baugruppe initialisieren und während des Betriebs steuern. Dazu verwenden Sie Steuersignale, Rückmeldesignale, Funktionsschalter sowie Schreib- und Leseaufträge.

Arbeitsweise

Bei jedem Aufruf führt die Anweisung folgende Tätigkeiten aus:

- Rückmeldesignale lesen:
  Die Anweisung CAM_CTRL_452 liest alle Rückmeldesignale der Baugruppe und trägt sie in den Kanal-DB ein. Da die Steuersignale und Aufträge erst anschließend bearbeitet werden, geben die Rückmeldesignale den Status der Baugruppe vor dem Aufruf der Anweisung wieder.
- Steuersignale schreiben:
  Die Steuersignale, die im Kanal-DB eingetragen sind, werden zur Baugruppe übertragen. Die Freigabe der Nockenbearbeitung wird jedoch so lange zurückgehalten, wie der Anstoß für einen Auftrag "Bezugspunkt setzen" oder "Nockendaten schreiben" gesetzt ist. Das (Wieder-) Einschalten der Nockenbearbeitung wird für diese Zeitspanne zurückgehalten.
- Auftragsverwaltung (Seite 6197):
  Aus den im Kanal-DB eingetragenen Anstoßbits für Aufträge wird der jeweils nächste Auftrag ausgeführt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden.

Bevor Sie die Anweisung aufrufen, tragen Sie alle Daten in den Kanal-DB ein, die für die Ausführung der beabsichtigten Funktionen erforderlich sind.

Verwendete Datenbausteine

- Kanal-DB (Seite 6184):
  Im Kanal-DB muss die Baugruppenadresse eingetragen sein.
- Parameter-DB (Seite 6191):
  Wenn Sie Maschinen- oder Nockendaten über Aufträge schreiben oder lesen wollen, benötigen Sie einen Parameter-DB, dessen Nummer im Kanal-DB eingetragen sein muss. Die Größe des Parameter-DB muss für die vorhandene Nockenanzahl ausreichend sein.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Kanal-DB
RET_VAL	OUTPUT	INT	Rückgabewert

Rückgabewerte

Die Anweisung liefert im Parameter RET_VAL des Kanal-DB folgende Rückgabewerte:

RET_VAL	BIE	Beschreibung
1	1	Mindestens 1 Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler: Datenfehler (DAT_ERR) oder Kommunikationsfehler (JOB_ERR) aufgetreten

Anlauf

Rufen Sie beim Anlauf der Baugruppe bzw. der CPU die Anweisung CAM_INIT (Seite 6175) auf. Dabei werden u. a. auch die Funktionsschalter zurückgesetzt.

Die Anweisung CAM_CTRL_452 quittiert den Anlauf der Baugruppe. Während dieser Zeit sind RET_VAL und JOBBUSY = 1.

Verhalten im Fehlerfall

Wenn bei einem Schreibauftrag fehlerhafte Daten geschrieben wurden, liefert die Baugruppe die Rückmeldung DATA_ERR = 1. Wenn bei einem Schreib- oder Leseauftrag ein Fehler bei der Kommunikation mit der Baugruppe auftritt, wird die Fehlerursache im Parameter JOB_ERR im Kanal-DB abgelegt.

- Fehler bei einem Schreibauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt. Bei allen noch anstehenden Schreibaufträgen wird ebenfalls das Anstoßbit zurückgenommen und das Fehlerbit _ERR gesetzt.
  Die anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.
- Fehler bei einem Leseauftrag:
  Bei dem fehlerhaften Auftrag wird das Anstoßbit zurückgenommen und das Fehlerbit _ERR und das Fertigbit _D gesetzt.
  Die noch anstehenden Leseaufträge werden weiter bearbeitet. Dabei wird JOB_ERR für jeden Auftrag wieder neu gesetzt.

CAM_DIAG_452

Beschreibung

Mit der Anweisung CAM_DIAG_452 lesen Sie den Diagnosepuffer der Baugruppe aus und stellen ihn für eine Anzeige im B&B-System oder für eine programmierte Auswertung zur Verfügung.

Arbeitsweise

Die Anweisung liest den Diagnosepuffer dann, wenn über das Rückmeldesignal DIAG = 1 ein neuer Eintrag im Diagnosepuffer angezeigt wird. Nach dem Lesen des Diagnosepuffers wird DIAG von der Baugruppe auf 0 gesetzt.

Aufruf

Die Anweisung muss zyklisch aufgerufen werden. Ein zusätzlicher Aufruf in einem Alarm-OB ist nicht zulässig. Für die vollständige Ausführung der Anweisung sind mindestens 2 Aufrufe (Zyklen) notwendig.

Verwendeter Datenbaustein

Diagnose-DB (Seite 6194):

Im Diagnose-DB muss die Baugruppenadresse eingetragen sein. Der neueste Eintrag des Diagnosepuffers wird in der Struktur DIAG[1] und der älteste Eintrag in der Struktur DIAG[4] eingetragen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Diagnose-DB
RET_VAL	OUTPUT	INT	Rückgabewert

Rückgabewerte

Die Anweisung liefert im Parameter RET_VAL des Diagnose-DB folgende Rückgabewerte:

RET_VAL	BIE	Beschreibung
1	1	Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler

Aufträge

Sie können den Diagnosepuffer unabhängig von einem neuen Eintrag lesen, wenn Sie das Anstoßbit DIAGRD_EN setzen. Nach dem Lesen des Diagnosepuffers wird das Anstoßbit auf 0 gesetzt.

Anlauf

Die Anweisung führt keine Anlaufbearbeitung durch.

Verhalten im Fehlerfall

Bei einem fehlerhaften Auftrag finden Sie die Fehlerursache im Diagnose-DB im Parameter JOB_ERR.

Fehlerklassen

Die auftretenden Fehler sind in folgenden Fehlerklassen zusammengefasst:

- Fehlerklasse 1: Betriebsfehler (Seite 6198)
- Fehlerklasse 4: Datenfehler (Seite 6199)
- Fehlerklasse 5: Maschinendatenfehler (Seite 6200)
- Fehlerklasse 7: Nockendatenfehler (Seite 6203)
- Fehlerklasse 15: Meldungen (Seite 6204)
- Fehlerklasse 128: Diagnosefehler (Seite 6204)

CAM_MSRM_452

Beschreibung

Die Anweisung CAM_MSRM_452 setzen Sie ein, wenn Sie die Messdaten von Längenmessung oder Kantenerfassung einer FM 452 sofort im Prozessalarm-OB auswerten möchten.

Aufruf

Die Anweisung wird in einem Prozessalarm-OB (z. B. OB 40) aufgerufen.

Verwendeter Datenbaustein

Kanal-DB (Seite 6184):

Im Kanal-DB muss die Baugruppenadresse eingetragen sein. Der Eintrag sollte im Anlauf-OB (OB 100) erfolgen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DB_NO	INPUT	INT	Nummer des Kanal-DB
RET_VAL	OUTPUT	INT	Rückgabewert

Rückgabewerte

Die Anweisung liefert im Parameter RET_VAL des Kanal-DB folgende Rückgabewerte:

RET_VAL	BIE	Beschreibung
1	1	Auftrag aktiv
0	1	Kein Auftrag aktiv, kein Fehler
-1	0	Fehler

Messergebnisse und Statusinformationen

Die Messergebnisse und Statusinformationen stehen im Kanal-DB:

Adresse	Name	Typ	Anfangs-wert	Kommentar
112.0	BEG_VAL	DINT	L#0	Anfangswert
116.0	END_VAL	DINT	L#0	Endwert
120.0	LEN_VAL	DINT	L#0	Länge
56.0	JOB_ERR_M	INT	0	Kommunikationsfehler
58.0	JOBBUSY_M	BOOL	FALSE	Auftrag aktiv

Anlauf

Die Anweisung führt keine Anlaufbearbeitung durch.

Verhalten im Fehlerfall

Bei einem fehlerhaften Auftrag finden Sie die Fehlerursache im Kanal-DB im Parameter JOB_ERR_M.

Weitere Referenzen für FM x52 CAM V2

Kanal-Datenbaustein

Aufgabe

Der Kanal-DB ist die Datenschnittstelle zwischen dem Anwenderprogramm und dem Elektronischen Nockensteuerwerk FM x52. Er enthält und übernimmt alle Daten, die zur Steuerung und zum Betrieb der Baugruppe notwendig sind.

Aufbau

Der Kanal-DB ist in verschiedene Bereiche unterteilt:

Kanal-DB
Adressen* / Versionsschalter
Steuersignale
Rückmeldesignale
Funktionsschalter
Anstoßbits für Schreibaufträge
Anstoßbits für Leseaufträge
Fertigbits
Fehlerbits
Auftragsverwaltung für Anweisungen
Daten für Aufträge
* Die Baugruppenadresse können Sie in der Parametrieroberfläche eintragen.

Inhalt

Die folgende Tabelle zeigt den Inhalt des Kanal-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
Adressen / Versionsschalter				
0.0	MOD_ADDR (Eintragen!)	INT	0	Baugruppenadresse
2.0	CH_NO	INT	1	Kanalnummer (immer 1)
10.0	PARADBNO	INT	-1	Nummer des Parameter-DB -1 = DB nicht vorhanden
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 bis V4.0 1 = FM 452 oder FM 352 ab V5.0
Steuersignale				
15.2	DIR_M	BOOL	FALSE	1 = Simulation in negativer Richtung
15.3	DIR_P	BOOL	FALSE	1 = Simulation in positiver Richtung
15.4	CAM_EN	BOOL	FALSE	1 = Nockenbearbeitung freigeben
15.5	CNTC0_EN	BOOL	FALSE	1 = Zählfunktion der Zählnockenspur 0 freigeben
15.6	CNTC1_EN	BOOL	FLASE	1 = Zählfunktion der Zählnockenspur 1 freigeben
16.0	TRACK_EN	WORD	W#16#0	Nockenspuren 0 bis 15 freischalten Bit 0 = Spur 0
Rückmeldesignale				
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler
22.7	PARA	BOOL	FALSE	1 = Baugruppe ist parametriert
23.4	CAM_ACT	BOOL	FALSE	1 = Nockenbearbeitung läuft
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
25.1	MSR_DONE	BOOL	FALSE	1 = Längenmessung oder Kantenerfassung ist abgeschlossen
25.2	GO_M	BOOL	FALSE	1 = Achse fährt in negative Richtung
25.3	GO_P	BOOL	FALSE	1 = Achse fährt in positive Richtung
25.4	HYS	BOOL	FALSE	1 = Achse befindet sich innerhalb des Hysteresebereichs
25.5	FVAL_DONE	BOOL	FALSE	1 = Fliegendes Istwert setzen ausgeführt
26.0	ACT_POS	DINT	L#0	Aktuelle Position der Achse
30.0	TRACK_OUT	DWORD	DW#16#0	Aktuelle Spursignale der Spuren 0 bis 31 Bit 0 = Spur 0
Funktionsschalter				
34.0	EDGE_ON	BOOL	FALSE	1 = Kantenerfassung ein
34.1	SIM_ON	BOOL	FALSE	1 = Simulation ein
34.2	MSR_ON	BOOL	FALSE	1 = Längenmessung ein
34.3	REFTR_ON	BOOL	FALSE	1 = Referenzpunkt nachtriggern
34.4	SSW_OFF	BOOL	FALSE	1 = Software-Endschalter abgeschaltet
Anstoßbits für Schreibaufträge				
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
35.2	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen, Fliegendes Istwert setzen rückgängig
35.3	CAM1WR_EN	BOOL	FALSE	1 = Nockendaten 1 schreiben (Nocken 0 bis 15)
35.4	CAM2WR_EN	BOOL	FALSE	1 = Nockendaten 2 schreiben (Nocken 16 bis 31)
35.5	CAM3WR_EN	BOOL	FALSE	1 = Nockendaten 3 schreiben (Nocken 32 bis 47)
35.6	CAM4WR_EN	BOOL	FALSE	1 = Nockendaten 4 schreiben (Nocken 48 bis 63)
35.7	CAM5WR_EN	BOOL	FALSE	1 = Nockendaten 5 schreiben (Nocken 64 bis 79)
36.0	CAM6WR_EN	BOOL	FALSE	1 = Nockendaten 6 schreiben (Nocken 80 bis 95)
36.1	CAM7WR_EN	BOOL	FALSE	1 = Nockendaten 7 schreiben (Nocken 96 bis 111)
36.2	CAM8WR_EN	BOOL	FALSE	1 = Nockendaten 8 schreiben (Nocken 112 bis 127)
36.3	REFPT_EN	BOOL	FALSE	1 = Bezugspunktkoordinaten setzen
36.4	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.5	FVAL_EN	BOOL	FALSE	1 = Fliegend Istwert setzen
36.6	ZOFF_EN	BOOL	FALSE	1 = Nullpunktverschiebung setzen
36.7	CH01CAM_EN	BOOL	FALSE	1 = Einstellung Nockenflanken (1 Nocken) schreiben
37.0	CH16CAM_EN	BOOL	FALSE	1 = Einstellung für schnelles Nockenändern (16 Nocken) schreiben
Anstoßbits für Leseaufträge				
37.1	MDRD_EN	BOOL	FALSE	1 = Maschinendaten lesen
37.2	CAM1RD_EN	BOOL	FALSE	1 = Nockendaten 1 lesen (Nocken 0 bis 15)
37.3	CAM2RD_EN	BOOL	FALSE	1 = Nockendaten 2 lesen (Nocken 16 bis 31)
37.4	CAM3RD_EN	BOOL	FALSE	1 = Nockendaten 3 lesen (Nocken 32 bis 47)
37.5	CAM4RD_EN	BOOL	FALSE	1 = Nockendaten 4 lesen (Nocken 48 bis 63)
37.6	CAM5RD_EN	BOOL	FALSE	1 = Nockendaten 5 lesen (Nocken 64 bis 79)
37.7	CAM6RD_EN	BOOL	FALSE	1 = Nockendaten 6 lesen (Nocken 80 bis 95)
38.0	CAM7RD_EN	BOOL	FALSE	1 = Nockendaten 7 lesen (Nocken 96 bis 111)
38.1	CAM8RD_EN	BOOL	FALSE	1 = Nockendaten 8 lesen (Nocken 112 bis 127)

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
38.2	MSRRD_EN	BOOL	FALSE	1 = Messwerte lesen
38.3	CNTTRC_EN	BOOL	FALSE	1 = Zählwerte der Zählnockenspuren lesen
38.4	ACTPOS_EN	BOOL	FALSE	1 = Positions- und Spurdaten lesen
38.5	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
38.6	CAMOUT_EN	BOOL	FALSE	1 = Nocken- und Spurdaten lesen
Fertigbits für Funktionsschalter				
40.0	EDGE_D	BOOL	FALSE	1 = "Kantenerfassung einschalten" bzw. "Kantenerfassung ausschalten" abgeschlossen
40.1	SIM_D	BOOL	FALSE	1 = "Simulation einschalten" bzw. "Simulation ausschalten" abgeschlossen
40.2	MSR_D	BOOL	FALSE	1 = "Längenmessung einschalten" bzw. "Längenmessung ausschalten" abgeschlossen
40.3	REFTR_D	BOOL	FALSE	1 = "Referenzpunkt nachtriggern einschalten" bzw. "Referenzpunkt nachtriggern ausschalten" abgeschlossen
40.4	SSW_D	BOOL	FALSE	1 = "Software-Endschalter einschalten" bzw. "Software-Endschalter ausschalten" abgeschlossen
Fertigbits für Schreibaufträge				
41.0	MDWR_D	BOOL	FALSE	1 = Auftrag "Maschinendaten schreiben" abgeschlossen
41.1	MD_D	BOOL	FALSE	1 = Auftrag "Maschinendaten freigeben" abgeschlossen
41.2	AVALREM_D	BOOL	FALSE	1 = "Istwert setzen rückgängig" bzw. "Fliegendes Istwert setzen rückgängig" abgeschlossen
41.3	CAM1WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 1 schreiben" abgeschlossen
41.4	CAM2WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 2 schreiben" abgeschlossen
41.5	CAM3WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 3 schreiben" abgeschlossen
41.6	CAM4WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 4 schreiben" abgeschlossen
41.7	CAM5WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 5 schreiben" abgeschlossen
42.0	CAM6WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 6 schreiben" abgeschlossen
42.1	CAM7WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 7 schreiben" abgeschlossen
42.2	CAM8WR_D	BOOL	FALSE	1 = Auftrag "Nockendaten 8 schreiben" abgeschlossen
42.3	REFPT_D	BOOL	FALSE	1 = Auftrag "Referenzpunkt setzen" abgeschlossen
42.4	AVAL_D	BOOL	FALSE	1 = Auftrag "Istwert setzen" abgeschlossen
42.5	FVAL_D	BOOL	FALSE	1 = Auftrag "Fliegendes Istwert setzen" abgeschlossen

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
42.6	ZOFF_D	BOOL	FALSE	1 = Auftrag "Nullpunktverschiebung setzen" abgeschlossen
42.7	CH01CAM_D	BOOL	FALSE	1 = Auftrag "1 Nocken ändern" abgeschlossen
43.0	CH16CAM_D	BOOL	FALSE	1 = Auftrag "16 Nocken ändern" abgeschlossen (schnelles Nockenändern)
Fertigbits für Leseaufträge				
43.1	MDRD_D	BOOL	FALSE	1 = Auftrag "Maschinendaten lesen" abgeschlossen
43.2	CAM1RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 1 lesen" abgeschlossen
43.3	CAM2RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 2 lesen" abgeschlossen
43.4	CAM3RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 3 lesen" abgeschlossen
43.5	CAM4RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 4 lesen" abgeschlossen
43.6	CAM5RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 5 lesen" abgeschlossen
43.7	CAM6RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 6 lesen" abgeschlossen
44.0	CAM7RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 7 lesen" abgeschlossen
44.1	CAM8RD_D	BOOL	FALSE	1 = Auftrag "Nockendaten 8 lesen" abgeschlossen
44.2	MSRRD_D	BOOL	FALSE	1 = Auftrag "Messwerte lesen" abgeschlossen
44.3	CNTTRC_D	BOOL	FALSE	1 = Auftrag "Zählwerte der Zählnockenspuren lesen" abgeschlossen
44.4	ACTPOS_D	BOOL	FALSE	1 = Auftrag "Positions- und Spurdaten lesen" abgeschlossen
44.5	ENCVAL_D	BOOL	FALSE	1 = Auftrag "Aktuellen Geberwert lesen" abgeschlossen
44.6	CAMOUT_D	BOOL	FALSE	1 = Auftrag "Nocken- und Spurdaten lesen" abgeschlossen
Fehlerbits für Funktionsschalter				
46.0	EDGE_ERR	BOOL	FALSE	1 = Fehler bei "Kantenerfassung einschalten" bzw. "Kantenerfassung ausschalten"
46.1	SIM_ERR	BOOL	FALSE	1 = Fehler bei "Simulation einschalten" bzw. "Simulation ausschalten"
46.2	MSR_ERR	BOOL	FALSE	1 = Fehler bei "Längenmessung einschalten" bzw. "Längenmessung ausschalten"
46.3	REFTR_ERR	BOOL	FALSE	1 = Fehler bei "Referenzpunkt nachtriggern einschalten" bzw. "Referenzpunkt nachtriggern ausschalten"
46.4	SSW_ERR	BOOL	FALSE	1 = Fehler bei "Software-Endschalter einschalten" bzw. "Software-Endschalter ausschalten"
Fehlerbits für Schreibaufträge				
47.0	MDWR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten schreiben"

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
47.1	MD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten aktivieren"
47.2	AVALREM_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Istwert setzen rückgängig" bzw. "Fliegendes Istwert setzen rückgängig"
47.3	CAM1WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 1 schreiben"
47.4	CAM2WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 2 schreiben"
47.5	CAM3WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 3 schreiben"
47.6	CAM4WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 4 schreiben"
47.7	CAM5WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 5 schreiben"
48.0	CAM6WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 6 schreiben"
48.1	CAM7WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 7 schreiben"
48.2	CAM8WR_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 8 schreiben"
48.3	REFPT_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Bezugspunkt setzen"
48.4	AVAL_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Istwert setzen"
48.5	FVAL_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Fliegendes Istwert setzen"
48.6	ZOFF_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nullpunktverschiebung setzen"
48.7	CH01CAM_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "1 Nocken ändern"
49.0	CH16CAM_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "16 Nocken ändern" (schnelles Nockenändern)
Fehlerbits für Leseaufträge				
49.1	MDRD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Maschinendaten lesen"
49.2	CAM1RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 1 lesen"
49.3	CAM2RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 2 lesen"
49.4	CAM3RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 3 lesen"
49.5	CAM4RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 4 lesen"
49.6	CAM5RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 5 lesen"
49.7	CAM6RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 6 lesen"
50.0	CAM7RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 7 lesen"
50.1	CAM8RD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nockendaten 8 lesen"
50.2	MSRRD_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Messwerte lesen"
50.3	CNTTRC_ERR	BOOL	FALSE	1 = Fehler bei Auftrag "Zählwerte der Zählnockenspuren lesen"
50.4	ACTPOS_ERR	BOOL	FALSE	1 = Fehler bei Auftrag "Positions- und Spurdaten lesen"

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
50.5	ENCVAL_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Aktuellen Geberwert lesen"
50.6	CAMOUT_ERR	BOOL	FALSE	1 = Fehler beim Auftrag "Nocken- und Spurdaten lesen"
Auftragsverwaltung für die Anweisung CAM_CTRL bzw. CAM_CTRL_452				
52.0	JOB_ERR	INT	0	Kommunikationsfehler
54.0	JOBBUSY	BOOL	FALSE	1 = mindestens ein Auftrag läuft
54.1	JOBRESET	BOOL	FALSE	1 = alle Fehler- und Fertigbits zurücksetzen
Auftragsverwaltung für die Anweisung CAM_MSRM_452				
56.0	JOB_ERR_M	INT	0	Kommunikationsfehler
58.0	JOBBUSY_M	BOOL	FALSE	1 = Auftrag läuft
Daten für Auftrag "Nullpunktverschiebung"				
86.0	ZOFF	DINT	L#0	Nullpunktverschiebung
Daten für Auftrag "Istwert setzen"				
90.0	AVAL	DINT	L#0	Koordinate für "Istwert setzen"
Daten für Auftrag "Fliegendes Istwert setzen"				
94.0	FVAL	DINT	L#0	Koordinate für "Fliegendes Istwert setzen"
Daten für Auftrag "Bezugspunkt setzen"				
98.0	REFPT	DINT	L#0	Koordinate für "Bezugspunkt setzen"
Daten für Auftrag "Nockenflanken ändern"				
102.0	CAM_NO	INT	0	Nockennummer
104.0	CAM_START	DINT	L#0	Nockenanfang
108.0	CAM_END	DINT	L#0	Nockenende
Daten für Auftrag "Längenmessung / Kantenerfassung"				
112.0	BEG_VAL	DINT	L#0	Anfangswert
116.0	END_VAL	DINT	L#0	Endwert
120.0	LEN_VAL	DINT	L#0	Länge
Daten für Auftrag "Zählwerte lesen"				
124.0	CNT_TRC0	INT	0	Aktueller Zählwert für Zählnockenspur 0
126.0	CNT_TRC1	INT	0	Aktueller Zählwert für Zählnockenspur 1
Daten für Auftrag "Positions- und Spurdaten lesen"				
128.0	ACTPOS	DINT	L#0	Aktuelle Position
132.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
136.0	TRACK_ID	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31
Daten für Auftrag "Geberdaten lesen"				
140.0	ENCVAL	DINT	L#0	Geberwert
144.0	ZEROVAL	DINT	L#0	Zählerstand bei der letzten Nullmarke
148.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage
Daten für Auftrag "Nocken- und Spurdaten lesen"				
152.0	CAM_00_31	DWORD	DW#16#0	Nockenkennbits für Nocken 0 bis 31
156.0	CAM_32_63	DWORD	DW#16#0	Nockenkennbits für Nocken 32 bis 63
160.0	CAM_64_95	DWORD	DW#16#0	Nockenkennbits für Nocken 64 bis 95
164.0	CAM_96_127	DWORD	DW#16#0	Nockenkennbits für Nocken 96 bis 127

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
168.0	TRACK_ID1	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31
172.0	ACTPOS1	DINT	L#0	Aktuelle Position
Daten für Auftrag "Schnelles Nockenändern"				
176.0	C_QTY	BYTE	B#16#0	Anzahl der zu ändernden Nocken
177.0	DIS_CHECK	BOOL	FALSE	1 = Datenprüfung ausschalten
180.0	CAM	ARRAY [0...15] STRUCT		Hinweis: Nachfolgende Struktur muss für jeden zu ändernden Nocken ausgefüllt werden.
Relative Adresse				
+0.0	CAM_NO	BYTE	B#16#0	Nummer des zu ändernden Nockens
+1.0	C_EFFDIR	BOOL	FALSE	1 = Änderung der Wirkrichtung
+1.1	C_CBEGIN	BOOL	FALSE	1 = Änderung des Nockenanfangs auf den Wert CBEGIN (neuer Nockenanfang)
+1.2	C_CEND	BOOL	FALSE	1 = Änderung des Nockenendes / der Einschaltzeit auf den Wert CEND (neues Nockenende)
+1.3	C_LTIME	BOOL	FALSE	1 = Änderung der Vorhaltezeit auf den Wert LTIME (neue Vorhaltezeit)
+1.4	CAM_OFF	BOOL	FALSE	1 = Ausschalten des Nockens während der Nockendatenänderung
+1.5	EFFDIR_P	BOOL	FALSE	1 = neue Wirkrichtung positiv (plus)
+1.6	EFFDIR_M	BOOL	FALSE	1 = neue Wirkrichtung negativ (minus)
+2.0	CBEGIN	DINT	L#0	Neuer Nockenanfang
+6.0	CEND	DINT	L#0	Neues Nockenende / neue Einschaltzeit
+10.0	LTIME	INT	0	Neue Vorhaltezeit

Parameter-Datenbaustein

Aufgabe

Die Maschinen- und Nockendaten sind im Parameter-DB abgelegt. Die Parameter können vom Anwenderprogramm oder von einem B&B-System verändert werden. Die veränderten Daten können in die Parametrieroberfläche importiert und dort angezeigt werden. Die in der Parametrieroberfläche angezeigten Daten können Sie in einen Parameter-DB exportieren.

Zu einer Baugruppe kann es mehrere Sätze von Parametrierdaten geben (z. B. für verschiedene Rezepte), auf die Sie programmgesteuert umschalten können.

Aufbau

Parameter-DB

CAM_P016TYPE (UDT3)

Maschinendaten

Nockendaten der Nocken 0 bis 15

CAM_P032TYPE (UDT4)
Maschinendaten
Nockendaten der Nocken 0 bis 31
CAM_P064TYPE (UDT5)
Maschinendaten
Nockendaten der Nocken 0 bis 63
CAM_P0128TYPE (UDT6)
Maschinendaten
Nockendaten der Nocken 0 bis 127

Inhalt

Die folgende Tabelle zeigt den Inhalt des Parameter-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
Maschinendaten				
3.1	PI_MEND	BOOL	FALSE	• FM 352: 0 • FM 452: 1: Prozessalarm freigeben: Messende
3.2	PI_CAM	BOOL	FALSE	1: Prozessalarm freigeben: Nocken ein / aus
3.5	PI_MSTRT	BOOL	FALSE	• FM 352: 0 • FM 452: 1: Prozessalarm freigeben: Messbeginn
4.0	EDGEDIST	DINT	L#0	Minimaler Kantenabstand bei Kantenerfassung
8.0	UNITS	DINT	L#1	Maßsystem
12.0	AXIS_TYPE	DINT	L#0	0: Linearachse, 1: Rundachse
16.0	ENDROTAX	DINT	L#100 000	Ende der Rundachse
20.0	ENC_TYPE	DINT	L#1	Gebertyp, Telegrammlänge
24.0	DISP_REV	DINT	L#80 000	Weg pro Geberumdrehung
32.0	INC_REV	DINT	L#500	Inkremente pro Geberumdrehung
36.0	NO_REV	DINT	L#1024	Anzahl Geberumdrehungen
40.0	BAUDRATE	DINT	L#0	Baudrate
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
48.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage
52.0	RETR_TYPE	DINT	L#0	Art für "Referenzpunkt nachtriggern"
59.0	CNT_DIR	DINT	L#0	Zählrichtung: 0: normal, 1: invertiert
63.0	MON_WIRE	BOOL	TRUE	1: Überwachung Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1: Überwachung Telegrammfehler

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
63.2	MON_PULSE	BOOL	TRUE	1: Überwachung Fehlimpulse
64.0	SSW_STRT	DINT	L#-100 000 000	Software-Endschalter Anfang
68.0	SSW_END	DINT	L#100 000 000	Software-Endschalter Ende
76.0	C_QTY	DINT	L#0	Mengengerüst: 0, 1, 2, 3 = max. 16, 32, 64, 128 Nocken
80.0	HYS	DINT	L#0	Hysterese
84.0	SIM_SPD	DINT	L#0	Simulationsgeschwindigkeit
90.0	TRACK_OUT	WORD	W#16#0	Ansteuerung der Spurausgänge: 0 = Nockensteuerwerk, 1 = CPU; Bitnummer = Spurnummer
95.0	EN_IN_I3	BOOL	FALSE	Freigabe Eingang I3
95.1	EN_IN_I4	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I4
95.2	EN_IN_I5	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I5
95.3	EN_IN_I6	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I6
95.4	EN_IN_I7	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I7
95.5	EN_IN_I8	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I8
95.6	EN_IN_I9	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I9
95.7	EN_IN_I10	BOOL	FALSE	• FM 352: 0 • FM 452: Freigabe Eingang I10
99.0	SPEC_TRC0	BOOL	FALSE	1 = Spur 0 ist Zählnockenspur
99.1	SPEC_TRC1	BOOL	FALSE	1 = Spur 1 ist Zählnockenspur
99.2	SPEC_TRC2	BOOL	FALSE	1 = Spur 2 ist Bremsnockenspur
100.0	CNT_LIM0	DINT	L#2	Oberer Zählwert für Zählnockenspur 0
104.0	CNT_LIM1	DINT	L#2	Oberer Zählwert für Zählnockenspur 1
Nockendaten jeweils für Nocken 0 bis 15 / 0 bis 31 / 0 bis 63 / 0 bis 127				
108.0		STRUCT		(je Element 12 Byte Länge)
Relative Adresse				
+0.0	CAMVALID	BOOL	FALSE	1: Nocken gültig
+0.1	EFFDIR_P	BOOL	TRUE	1: Wirkrichtung positiv (plus)
+0.2	EFFDIR_M	BOOL	TRUE	1: Wirkrichtung negativ (minus)
+0.3	CAM_TYPE	BOOL	FALSE	0: Wegnocken, 1: Zeitnocken

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
+0.4	PI_SW_ON	BOOL	FALSE	1: Prozessalarm beim Einschalten
+0.5	PI_SW_OFF	BOOL	FALSE	1: Prozessalarm beim Ausschalten
+1.0	TRACK_NO	BYTE	B#16#0	Spurnummer
+2.0	CBEGIN	DINT	L#-100 000 000	Nockenanfang
+6.0	CEND	DINT	L#100 000 000	Nockenende / Einschaltzeit
+10.0	LTIME	INT	0	Vorhaltezeit

Diagnose-Datenbaustein

Aufgabe

Der Diagnose-DB ist die Datenablage für die Anweisung CAM_DIAG (Seite 6178) bzw. CAM_DIAG_452 (Seite 6181). Er enthält den von dieser Anweisung aufbereiteten Diagnosepuffer der Baugruppe.

Aufbau

Diagnose-DB
Baugruppenadresse
Interne Daten
Auftragsstatus
Anstoßbit
Aufbereiteter Diagnosepuffer

Inhalt

Die folgende Tabelle zeigt den Inhalt des Diagnose-DB.

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
0.0	MOD_ADDR (Eintragen!)	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Kommunikationsfehler
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten letzter Eintrag

Struktur des Diagnoseeintrags

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Parameter	Datentyp	Anfangswert	Beschreibung
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 4: Datenfehler 5: Maschinendatenfehler 7: Nockendatenfehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer 0 … 255
+6.0	CH_NO	INT	0	Kanalnummer (immer 1)
+8.0	CAMNO	INT	0	Nockennummer 0 bis 127 bei Fehlerklasse = Nockendatenfehler

Auftragsverwaltung (CAM_CTRL)

Aufträge

Der über die Steuer- und Rückmeldesignale hinausgehende Datenaustausch mit der Baugruppe wird über Aufträge abgewickelt.

Um einen Auftrag abzugeben, setzen Sie das entsprechende Anstoßbit im Kanal-DB und bei Schreibaufträgen noch die entsprechenden Daten. Rufen Sie dann die Anweisung CAM_CTRL (Seite 6176) auf, um den Auftrag auszuführen.

Wenn Sie die Baugruppe zentral einsetzen, wird ein Leseauftrag sofort ausgeführt. Wenn Sie die FM 352 dezentral einsetzen, kann ein Leseauftrag mehrere Zyklen benötigen.

Ein Schreibauftrag benötigt wegen der erforderlichen Quittungen der Baugruppe mindestens 3 Aufrufe (bzw. OB-Zyklen). Wenn Sie die FM 352 dezentral einsetzen, kann ein Schreibauftrag mehr als 3 Aufrufe benötigen.

Sie können mehrere Aufträge gleichzeitig abgeben, auch zusammen mit Steuersignalen. Bis auf den Auftrag zum Schreiben der Funktionsschalter werden die Aufträge in der Reihenfolge der Anstoßbits abgearbeitet, die im Kanal-DB festgelegt ist. Ist ein Auftrag fertig bearbeitet, wird das Anstoßbit zurückgenommen. Beim nächsten Aufruf der Anweisung wird der folgende Auftrag ermittelt und ausgeführt.

Zu jedem Auftrag gibt es neben dem Anstoßbit auch ein Fertigbit und ein Fehlerbit. Sie unterscheiden sich in der Endung des Namens:

Bit	Endung
Anstoßbit	_EN ("enable")
Fertigbit	_D ("done")
Fehlerbit	_ERR ("error")

Fertig- und Fehlerbits des Auftrags sollten Sie nach der Auswertung oder vor Abgabe dieses Auftrags auf 0 setzen.

Wenn Sie das Bit JOBRESET setzen, werden vor der Bearbeitung der anstehenden Aufträge alle Fertig- und Fehlerbits zurückgesetzt. Das Bit JOBRESET wird anschließend wieder auf 0 gesetzt.

Funktionsschalter

Die Funktionsschalter schalten Zustände der Baugruppe ein und aus. Ein Auftrag zum Schreiben der Funktionsschalter wird nur bei einer Änderung einer Schalterstellung ausgeführt. Die Stellung des Funktionsschalters bleibt nach der Ausführung des Auftrags erhalten.

Längenmessung und Kantenerfassung dürfen nicht gleichzeitig eingeschaltet sein. Deshalb sorgt die Anweisung CAM_CTRL dafür, dass beim Einschalten des einen Funktionsschalters der andere ausgeschaltet wird. Schalten Sie jedoch beide Funktionsschalter gleichzeitig ein (0 → 1), so wird die Längenmessung aktiviert.

Funktionsschalter und Aufträge können Sie bei einem Aufruf der Anweisung CAM_CTRL gleichzeitig verwenden.

Zu den Funktionsschaltern gibt es wie bei den Aufträgen Fertigbits mit der Namensendung _D und Fehlerbits mit der Namensendung _ERR.

Um die Fertig- und Fehlerbits der Funktionsschalter auswerten zu können, sollten Sie diese Bits 0 setzen, wenn Sie einen Funktionsschalter verändern.

Auftragsstatus

Den Status der Auftragsbearbeitung können Sie am Rückgabewert RETVAL und am Tätigbit JOBBUSY im Kanal-DB ablesen. Den Status eines einzelnen Auftrags können Sie anhand der Anstoß-, Fertig- und Fehlerbits dieses Auftrags auswerten.

	RETVAL	JOBBUSY	Anstoßbit _EN	Fertigbit _D	Fehlerbit _ERR
Auftrag aktiv	1	1	1	0	0
Auftrag fertig ohne Fehler	0	0	0	1	0
Auftrag fertig mit Fehler	-1	0	0	1	1
Schreibauftrag abgebrochen	-1	0	0	0	1

Auftragsverwaltung (CAM_CTRL_452)

Aufträge

Der über die Steuer- und Rückmeldesignale hinausgehende Datenaustausch mit der Baugruppe wird über Aufträge abgewickelt.

Um einen Auftrag abzugeben, setzen Sie das entsprechende Anstoßbit im Kanal-DB und bei Schreibaufträgen noch die entsprechenden Daten. Rufen Sie dann die Anweisung CAM_CTRL_452 (Seite 6179) auf, um den Auftrag auszuführen.

Ein Leseauftrag wird sofort ausgeführt. Ein Schreibauftrag benötigt wegen der erforderlichen Quittungen der Baugruppe mindestens 3 Aufrufe (bzw. OB-Zyklen).

Sie können mehrere Aufträge gleichzeitig abgeben, auch zusammen mit Steuersignalen. Bis auf den Auftrag zum Schreiben der Funktionsschalter werden die Aufträge in der Reihenfolge der Anstoßbits abgearbeitet, die im Kanal-DB festgelegt ist. Ist ein Auftrag fertig bearbeitet, wird das Anstoßbit zurückgenommen. Beim nächsten Aufruf der Anweisung wird der folgende Auftrag ermittelt und ausgeführt.

Zu jedem Auftrag gibt es neben dem Anstoßbit auch ein Fertigbit und ein Fehlerbit. Sie unterscheiden sich in der Endung des Namens:

Bit	Endung
Anstoßbit	_EN ("enable")
Fertigbit	_D ("done")
Fehlerbit	_ERR ("error")

Fertig- und Fehlerbits des Auftrags sollten Sie nach der Auswertung oder vor Abgabe dieses Auftrags auf 0 setzen.

Wenn Sie das Bit JOBRESET setzen, werden vor der Bearbeitung der anstehenden Aufträge alle Fertig- und Fehlerbits zurückgesetzt. Das Bit JOBRESET wird anschließend wieder auf 0 gesetzt.

Funktionsschalter

Die Funktionsschalter schalten Zustände der Baugruppe ein und aus. Ein Auftrag zum Schreiben der Funktionsschalter wird nur bei einer Änderung einer Schalterstellung ausgeführt. Die Stellung des Funktionsschalters bleibt nach der Ausführung des Auftrags erhalten.

Längenmessung und Kantenerfassung dürfen nicht gleichzeitig eingeschaltet sein. Deshalb sorgt die Anweisung CAM_CTRL_452 dafür, dass beim Einschalten des einen Funktionsschalters der andere ausgeschaltet wird. Schalten Sie jedoch beide Funktionsschalter gleichzeitig ein (0 → 1), so wird die Längenmessung aktiviert.

Funktionsschalter und Aufträge können Sie bei einem Aufruf der Anweisung CAM_CTRL_452 gleichzeitig verwenden.

Zu den Funktionsschaltern gibt es wie bei den Aufträgen Fertigbits mit der Namensendung _D und Fehlerbits mit der Namensendung _ERR.

Um die Fertig- und Fehlerbits der Funktionsschalter auswerten zu können, sollten Sie diese Bits 0 setzen, wenn Sie einen Funktionsschalter verändern.

Auftragsstatus

Den Status der Auftragsbearbeitung können Sie am Rückgabewert RET_VAL und am Tätigbit JOBBUSY im Kanal-DB ablesen. Den Status eines einzelnen Auftrags können Sie anhand der Anstoß-, Fertig- und Fehlerbits dieses Auftrags auswerten.

	RET_VAL	JOBBUSY	Anstoßbit _EN	Fertigbit _D	Fehlerbit _ERR
Auftrag aktiv	1	1	1	0	0
Auftrag fertig ohne Fehler	0	0	0	1	0
Auftrag fertig mit Fehler	-1	0	0	1	1
Schreibauftrag abgebrochen	-1	0	0	0	1

Fehlerklassen

Fehlerklasse 1: Betriebsfehler

Beschreibung

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
2	**Software-Endschalter Ende überfahren**		ja
3	**Verfahrbereichsanfang überfahren**		ja
4	**Verfahrbereichsende überfahren**		ja
13	**Fliegendes Istwertsetzen nicht ausführbar**		ja
	Ursache	Die Software-Endschalter liegen nach dem fliegenden Istwertsetzen außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen ist größer als ±100 m bzw. ±1000 m.	
	Wirkung	Achse nicht synchronisiert.	

Fehlerklasse 4: Datenfehler

Beschreibung

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt.

Nr.	Bedeutung		Diagnosealarm
10	**Fehlerhafte Nullpunktverschiebung**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten-Aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen (fehlerfreien) Maschinendaten auf der Baugruppe vorhanden	
21	**Fliegendes Istwertsetzen nicht zulässig**		nein
	Ursache	Es wurde versucht, "fliegendes Istwertsetzen" bei eingeschaltetem "Referenzpunkt nachtriggern" auszuführen.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0. Es wurde versucht, "Längenmessung" und "Kantenerfassung" gleichzeitig anzuwählen.	
28	**Referenzpunkt nachtriggern nicht zulässig**		nein
	Ursache	Es wurde versucht, "Referenzpunkt nachtriggern" bei anstehendem "fliegendes Istwertsetzen" auszuführen. Es wurde versucht, "Referenzpunkt nachtriggern" bei einem SSI-Geber auszuführen.	
29	**Unerlaubtes bitcodiertes Kommando**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
30	**Falsche Vorhaltezeit**		nein
31	**Falsche Nockennummer**		nein
	Ursache	Der Nocken ist nicht gültig Die Nockennummer ist nicht im Bereich 0...127.	

Nr.	Bedeutung		Diagnosealarm
32	**Falscher Nockenanfang**		nein
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Der Nockenanfang ist < 0 oder größer als das Ende der Rundachse.	
33	**Falsches Nockenende / falsche Einschaltzeit**		nein
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Das Nockenende ist < 0 oder größer als das Ende der Rundachse. Der Nocken ist nicht mindestens für die Dauer eines Impulses eingeschaltet. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereiches liegen.	
35	**Fehlerhafte Istwertvorgabe beim Istwertsetzen / fliegendes Istwertsetzen**		nein
	Ursache	Die Istwertvorgabe liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw ±1000 m. Die Software-Endschalter würden nach der Ausführung der Einstellung außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m) liegen. Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen wäre größer als ± 100 m bzw. ± 1000 m.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind keine Maschinendaten vorhanden. Auf der Achse sind keine Maschinendaten aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Eine der Einstellungen "Istwertsetzen" oder "fliegendes Istwertsetzen" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	
109	**Nockenbearbeitung läuft.**		nein
110	**Falsche Anzahl zu ändernder Nocken**		nein

Fehlerklasse 5: Maschinendatenfehler

Beschreibung

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	
6	**Falscher minimaler Kantenabstand**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben	

Nr.	Bedeutung		Diagnosealarm
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 10	
11	**Falscher Weg/Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Anzahl Inkremente/Geberumdrehung**		ja
14	**Falsche Anzahl Umdrehungen**		ja
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereiches von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m je nach Auflösung. Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Geber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art für "Referenzpunkt nachtriggern"**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 6 und 7 angegeben.	
19	**Falsche Richtungsanpassung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung auf Telegrammfehler im Parameter-DB auf "FALSE" gesetzt. Geber ermöglicht keine Überwachung von Fehlimpulsen (MON_PULSE deaktivieren)	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw. -1000 m (je nach Auflösung).	

Nr.	Bedeutung		Diagnosealarm
22	**Falscher Software-Endschalter Ende**		ja
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Linearachse: Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
144	**Falsches Mengengerüst**		ja
	Ursache	Sie haben für das Mengengerüst einen Wert ungleich 0 bis 3 angegeben.	
145	**Falsche Hysterese**		ja
	Ursache	Die Hysterese liegt außerhalb des Bereichs 0...65535 × Auflösung. Die Hysterese ist größer als ¼ × Arbeitsbereich bzw. ¼ × Rundachsenbereich.	
146	**Falsche Simulationsgeschwindigkeit**		ja
	Ursache	Die Simulationsgeschwindigkeit liegt außerhalb des Bereichs 1000 × AUFL bis 3×10^7 × AUFL oder ist größer als 5×10^8 µm/min. Die Simulationsgeschwindigkeit ist intern nicht einstellbar.	
147	**Falsche Spur**		ja
	Ursache	Die Ansteuerung einer Spur außerhalb 0...15 (Bit 0...15) wurde gewählt.	
148	**Falsche Anwahl der Freigabeeingänge**		ja
	Ursache	Sie wollten eine Spur außerhalb 3...10 (Bit 0...7) durch ein externes Signal freischalten.	
149	**Falsche Sonderspuranwahl**		ja
	Ursache	Sie wollten eine Spur außerhalb 0, 1 und 2 (Bit 0, 1 und 2) als Sonderspur definieren	
150	**Falscher oberer Zählwert Spur 0**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
151	**Falscher oberer Zählwert Spur 1**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben einen Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Geber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Geber und Rundachse: Der Geber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Geber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Fehlerklasse 7: Nockendatenfehler

Beschreibung

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Nr.	Bedeutung		Diagnosealarm
1	**Prozessalarm unzulässig**		ja
	Ursache	Sie wollen einen Prozessalarm bei einem Nocken mit einer Nockennummer > 7 angeben.	
2	**Falsche Spurnummer**		ja
	Ursache	Die Spurnummer liegt außerhalb des Bereichs 0 bis 31	
3	**Falscher Nockenanfang**		ja
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Der Nockenanfang ist < 0 und oder größer als das Ende der Rundachse.	
4	**Falsches Nockenende**		ja
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Der Nocken ist nicht mindestens 1 Impuls lang. Rundachse: Das Nockenende ist < 0 und oder größer als das Ende der Rundachse. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
5	**Falsche Einschaltzeit**		ja
	Ursache	Die Einschaltzeit ist < 0 µs. Der Maximalwert ist abhängig vom Mengengerüst.	
6	**Falsche Vorhaltezeit** [1]		ja
	Ursache	Die Vorhaltezeit ist < 0 µs. Der Maximalwert ist abhängig vom Mengengerüst.	
50	**Zu viele Nockensätze**		ja
	Ursache	Sie wollten mehr Nockensätze eingeben als bei diesem Mengengerüst möglich sind.	
51	**Achse in Betrieb**		ja
	Ursache	Sie wollten bei eingeschaltetem Nockensteuerwerk Nockensätze eingeben.	
52	**Achse nicht parametriert**		ja
	Ursache	Sie wollen Nockendaten eingeben, obwohl noch keine Maschinendaten aktiv sind.	

[1] Die Fehlermeldung kann auch dann auftreten, wenn Sie in Verbindung mit einem Absolutwertgeber (SSI) als Zählrichtung "invertiert" parametriert haben.

Fehlerklasse 15: Meldungen

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	

Fehlerklasse 128: Diagnosefehler

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	• Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen • Frontstecker fehlt • Kurzschluss, z. B. am angeschlossenen Geber	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spur-Ausgänge • Löschen der Synchronisation bei Inkrementalgebern • Die Baugruppe ist nicht parametriert (Rückmeldesignal PARA = 0).	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (Wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
5	**Frontstecker fehlt (FM 452)**		ja
	Ursache	Frontstecker nicht gesteckt	
	Wirkung	• Externe Hilfsspannung 24 V fehlt • Baugruppe ist nicht betriebsbereit	
	Behebung	Frontstecker aufstecken	
51	**Zeitüberwachung angesprochen (Watchdog)**		ja
	Ursache	• starke Störeinflüsse auf die Baugruppe • Fehler in der Baugruppe	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit. • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich bitte an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind. • Tauschen der Baugruppe	

Nr.	Bedeutung		Diagnosealarm
52	**Baugruppeninterne Versorgung ausgefallen**		ja
	Ursache	Fehler in der Baugruppe	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit.	
	Behebung	Tauschen der Baugruppe	
70	**Prozessalarm verloren**		ja
	Ursache	Ein Prozessalarmereignis wurde von der Baugruppe erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht vom Anwenderprogramm/der CPU verarbeitet wurde.	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spur-Ausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• Prozessalarm-OB in Anwenderprogramm einbinden • Busanschluss der Baugruppe überprüfen • Prozessalarm deaktivieren • Passen Sie Ihre Hardware und Software Ihren Prozessanforderungen an (z. B. schnellere CPU, Anwenderprogramm optimieren).	
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spur-Ausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit der Parametrieroberfläche vorübergehend unter Verantwortung des Betreibers ausgeblendet werden. • Technische Daten der Baugruppe einhalten	

Nr.	Bedeutung		Diagnosealarm
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen Baugruppe und dem Absolutwertgeber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spur-Ausgänge • der letzte korrekte Istwert bleibt bis zum Ende der nächsten korrekten SSI-Übertragung unverändert	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und Baugruppe	
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spur-Ausgänge • Löschen der Synchronisation	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit der Parametrieroberfläche vorübergehend unter Verantwortung des Betreibers ausgeblendet werden.	

FM 355 PID Control

PID_FM

Beschreibung PID_FM

Die Anweisung PID_FM dient dazu im laufenden Betrieb:

- Bedienparameter auf die FM 355 zu übertragen
- Reglerparameter auf die FM 355 zu übertragen
- Prozesswerte aus der FM 355 zu lesen

Bedienparameter sind alle Durchgangsparameter, die im Instanz-DB der Anweisung zwischen den Parametern op_par und cont_par liegen.

Reglerparameter sind alle Durchgangsparameter, die im Instanz-DB der Anweisung nach dem Parameter cont_par liegen.

Prozesswerte sind alle Ausgangsparameter der Anweisung nach dem Parameter out_par.

Die benötigten Daten werden in einem Instanz-DB auf der CPU abgelegt. Für jeden genutzten Reglerkanal benötigen Sie einen Instanz-DB. Nach Erzeugen eines Instanz-DB sind alle Durchgangsparameter auf FALSE gesetzt.

Aufruf

PID_FM wird normalerweise im Weckalarm-OB 35 aufgerufen und benötigt einen Initialisierungslauf, der durch Setzen des Parameters COM_RST = TRUE im Anlauf der CPU veranlasst wird. Ein Aufruf der Anweisung im Anlauf-OB ist möglich, aber nicht erforderlich. Nach dem Initialisierungslauf setzt PID_FM den Parameter COM_RST auf FALSE. Dies kann bei Einsatz der FM 355 in dezentraler Peripherie einige Aufrufzyklen dauern.

PID_FM muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Wenn Sie die Anweisung PID_FM als Multiinstanz-DB aufrufen, steht Ihnen keine Inbetriebnahmeoberfläche zur Verfügung. Sie müssen die FM 355 in diesem Fall über eine Beobachtungstabelle in Betrieb nehmen.

Anlauf

Beim Anlauf der CPU und beim STOP-RUN-Übergang der CPU werden die Parameter an der FM 355 mit den Parametern aus dem SDB der CPU überschrieben.

Verhalten im Fehlerfall

Der Ausgangsparameter RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn die Parameter READ_VAR und LOAD_PAR nicht zurückgesetzt werden.

Beim Aufruf der Anweisung PID_FM kann ein Peripheriezugriffsfehler (PZF) auftreten, wenn die FM 355 nicht gesteckt ist oder keine Spannungsversorgung hat. In diesem Fall geht die CPU in STOP, wenn kein OB 122 in der CPU geladen ist.

Siehe auch

Arbeitsweise PID_FM (Seite 6208)

Arbeitsweise PID_FM

Einfluss der Parameter LOAD_OP, READ_VAR und LOAD_PAR

Im Initialisierungslauf liest PID_FM die Parameter aus den Systemdaten und legt sie in seinem Instanz-DB ab. Instanz-DB und Systemdaten sind abgeglichen.

Die folgende Tabelle zeigt, wie die Funktion der Anweisung durch die Parameter LOAD_OP, READ_VAR und LOAD_PAR beeinflusst wird.

Parameter	TRUE	FALSE
LOAD_OP	Bedienparameter mit den Anweisungen WRREC und RDREC übertragen.	Bedienparameter über direkte Peripheriezugriffe im Multiplex-Verfahren übertragen
	READ_VAR wird TRUE gesetzt	READ_VAR wird nicht verändert
	Längere Laufzeit	Geringere Laufzeit
	Die Werte sind nach einem Zyklus der CPU oder FM wirksam. Der längere Zyklus ist maßgeblich.	Max. sind 3 Zyklen der CPU oder FM notwendig bis die Werte wirksam werden. Der längere Zyklus ist maßgeblich

Parameter	TRUE	FALSE
READ_VAR	Prozesswerte mit den Anweisungen WRREC und RDREC übertragen	Prozesswerte werden über direkte Peripheriezugriffe im Multiplex-Verfahren übertragen
	Längere Bearbeitungszeit	Geringere Bearbeitungszeit
	Alle Parameter werden aus der FM 355 gelesen	Folgende Parameter werden **nicht** aus der FM 355 gelesen • SP (Sollwert aus der FM) • ER (Regeldifferenz) • DISV (Störgröße) • LMN_A • LMN_B
	Folgende Parameter werden bei **jedem** Aufruf der Anweisung aktualisiert. • PV (Istwert) • LMN (Stellwert) • die Binäranzeigen	Folgende Parameter werden bei jedem **vierten** Aufruf der Anweisung aktualisiert. • PV (Istwert) • LMN (Stellwert) • die Binäranzeigen
	Im Anlauf der CPU übernimmt PID_FM folgende Werte aus der FM 355: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON Wenn diese Werte auf der CPU geändert wurden, werden Sie beim nächsten Anlauf der CPU mit den Werten aus der FM 355 überschrieben	Im Anlauf der CPU werden folgende Werte von der Anweisung gelesen, nicht aus der FM: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON
LOAD_PAR	Alle Reglerparameter werden von der CPU auf die FM 355 geladen. Dies ist sinnvoll, wenn die Regelparameter abhängig von Prozesszuständen geändert werden sollen.	Keine Übertragung der Reglerparameter

Nach erfolgreicher Datenübertragung werden die Parameter LOAD_OP, READ_VAR und LOAD_PAR wieder auf FALSE gesetzt.

Einsatz in dezentraler Peripherie

Bei Einsatz der FM 355 in dezentraler Peripherie können die Datenübertragung und das Rücksetzen der Parameter einige Aufrufzyklen dauern.

Besonderheiten bei Übertragung im Multiplex-Verfahren

Das Multiplexen der zu übertragenden Daten bei Zugriff auf die FM 355 über direkte Peripheriezugriffe wird über PID_FM gesteuert. Die Multiplexsteuerung durch PID_FM funktioniert nicht, wenn zwei Instanzen der Anweisung PID_FM auf die gleiche Kanalnummer einer Baugruppe zugreifen. Das Resultat sind falsche Parameter in der FM 355 (z. B. Sollwert und Handstellwert) und falsche Anzeigen an den Ausgangsparametern der Anweisung PID_FM.

Abspeichern der Parameter im EEPROM

Beim programmgesteuerten Umparametrieren (LOAD_PAR, LOAD_OP) der Reglerbaugruppe durch die Anweisung PID_FM vergrößert sich dessen Laufzeit. Die neuen Parameter sind immer sofort wirksam. Reglerparameter und der Bedienparameter SP_ER werden zusätzlich in einem nichtflüchtigen Speicher (EEPROM) abgelegt. Nach jedem Speichern der Parameter im EEPROM wird ein erneutes Abspeichern um 30 Minuten verzögert, da die Lebensdauer des EEPROMs durch die Anzahl der Schreibvorgänge begrenzt ist. Nach Wiederkehr der Versorgungsspannung ist das Abspeichern neuer Parameter im EEPROM sofort möglich. Ob die Umparametrierung der Reglerbaugruppe durch die Anweisung PID_FM stoßfrei erfolgt, hängt von der Wahl der Parameter ab.

Bearbeitungszeiten PID_FM

Randbedingungen			Bearbeitungszeit in	
READ_VAR	LOAD_OP	LOAD_PAR	CPU 314	CPU 414-2DP
FALSE	FALSE	FALSE	0,65 ms	0,077 ms
TRUE	FALSE	FALSE	2,85 ms	2,36 ms
TRUE	TRUE	FALSE	4,56 ms	4,48 ms
FALSE	FALSE	TRUE	3,75 ms	2,59 ms
TRUE	FALSE	TRUE	5,95 ms	5,15 ms
TRUE	TRUE	TRUE	7,66 ms	7,1 ms

Siehe auch

Beschreibung PID_FM (Seite 6207)

Eingangsparameter PID_FM

Tabelle 4-227

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässig sind die Werte 1-4.
PHASE	4.0	INT	0	Der Parameter PHASE kann mit dem Ausgangsparameter PHASE der Anweisungen TUN_EC oder TUN_ES (PID Self-Tuner) verschaltet werden. Im Dialog "Inbetriebnahme" kann dann der Phasenzustand des PID Self-Tuners im Klartext angezeigt werden. Dieser Parameter hat für das OP keine Bedeutung.

Siehe auch

Beschreibung PID_FM (Seite 6207)

Ausgangsparameter PID_FM

Tabelle 4-228

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	6.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
out_par	8.0	WORD	W#16#3130	Der Parameter out_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Ausgangsparameter, die von der Baugruppe gelesen werden, wenn READ_VAR = TRUE gesetzt wird.
SP	10.0	REAL	0.0	Am Ausgang "Sollwert" wird der effektiv wirkende Sollwert ausgegeben.
PV	14.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	18.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.
DISV	22.0	REAL	0.0	Am Ausgang "Störgröße" wird die effektiv wirkende Störgröße ausgegeben. Zulässig sind Werte von -100 bis 100 %
LMN	26.0	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Parameter LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100 bis 100 %
LMN_A	30.0	REAL	0.0	Am Ausgang "Stellwert A der Splitrangefunktion / Stellungsrückmeldung" wird beim kontinuierlichen Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Der Ausgang LMN_A kann nur zur ungefähren Anzeige einer entsprechenden simulierten Stellgröße verwendet werden. Der Startwert LMNRSVAL der simulierten Stellungsrückmeldung ist hierbei entsprechend zu parametrieren und wird beim Setzen von LMNRS_ON wirksam. Zulässig sind Werte von -100 bis 100 %
LMN_B	34.0	REAL	0.0	Am Ausgang "Stellwert B der Splitrangefunktion" wird beim kontinuierlichen Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100 bis 100 %
QH_ALM	38.0	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Alarm angesprochen" wird das Überschreiten der Grenze H_ALM gemeldet.
QH_WRN	38.1	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Warnung angesprochen" wird das Überschreiten der Grenze H_WRN gemeldet.
QL_WRN	38.2	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Warnung angesprochen" wird das Unterschreiten der Grenze L_WRN gemeldet.
QL_ALM	38.3	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Alarm angesprochen" wird das Unterschreiten der Grenze L_ALM gemeldet.
QLMN_HLM	38.4	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Stellwertes angesprochen" meldet die Überschreitung der oberen Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMN_LLM	38.5	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Stellwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QPARA_F	38.6	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang "Parametrierfehler" angezeigt. Diesen Parametrierfehler können Sie auch im Dialog "Online & Diagnose" der Baugruppe auslesen.
QCH_F	38.7	BOOL	FALSE	Der Ausgang "Kanalfehler" wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPARA_F = 1 oder QMOD_F = 1 gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	39.0	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "positive Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	39.1	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "negative Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	39.2	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Sollwertes angesprochen" meldet die Überschreitung der oberen Begrenzung.
QSP_LLM	39.3	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Sollwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung.
QLMNUP	39.4	BOOL	FALSE	Dies ist der Ausgang "Stellwertsignal Hoch". (Nur beim Schrittregler oder Impulsregler)
QLMNDN	39.5	BOOL	FALSE	Dies ist der Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QID	39.6	BOOL	FALSE	QID = TRUE zeigt an, dass eine Identifizierung läuft (nicht dass sie eingeschaltet ist). Nach dem Ende der Identifizierung kann das Identifizierungsergebnis über den Parameter IDSTATUS der Anweisung CH_DIAG ausgelesen werden.
QSPOPON	40.0	BOOL	FALSE	Am Ausgang "Sollwertbedienung eingeschaltet" wird angezeigt, ob der Sollwert durch die Anweisungen TUN_EC oder TUN_ES bedient wird. Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
QLMNSAFE	40.1	BOOL	FALSE	Ist der Ausgang "Sicherheitsbetrieb" gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNO-PON	40.2	BOOL	FALSE	Am Ausgang "Stellwertbedienung eingeschaltet " wird angezeigt, ob der Stellwert durch die Anweisungen TUN_EC oder TUN_ES bedient wird. Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
QLMNTRK	40.3	BOOL	FALSE	Der Ausgang "Nachführbetrieb" zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	40.4	BOOL	FALSE	Der Ausgang "Hand = 1; Automatik = 0" zeigt an, ob der Stellwert auf den externen Stellwert LMN_RE gesetzt wird (Hand = 1) oder nicht.
QLMNR_HS	40.5	BOOL	FALSE	Der Ausgang "oberes Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am oberen Anschlag befindet. QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNR_LS	40.6	BOOL	FALSE	Der Ausgang "unteres Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am unteren Anschlag befindet. QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
QLMNR_ON	40.7	BOOL	FALSE	Der Ausgang "Stellungsrückmeldung eingeschaltet" zeigt die eingestellte Betriebsart "Schrittregler mit Stellungsrückmeldung" bzw. "Schrittregler ohne Stellungsrückmeldung" an.
QFUZZY	41.0	BOOL	FALSE	Ist der Ausgangsparameter QFUZZY =1, arbeitet der Regler mit dem Fuzzy-Algorithmus.
QSPLEPV	41.1	BOOL	FALSE	Der Ausgang "Anzeige vom FUZZY-Regler: Sollwert < Istwert" wird bei eingeschaltetem Fuzzy-Regler gesetzt, wenn der Sollwert kleiner als der wirksame Istwert ist.
QSPR	41.2	BOOL	FALSE	Ist der Ausgang "Split-Range-Betrieb" gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QMAN_FC	41.4	BOOL	FALSE	Der Ausgang "QMAN_FC" wird in folgenden beiden Fällen gesetzt: Der Folgeregler ist in Handbetrieb und der Führungsregler wird auf den Istwert des Folgereglers nachgeführt. Der I-Anteil des Führungsreglers wird angehalten, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind oder weil der Folgeregler in Handbetrieb ist.
QPARA-BUB	41.7	BOOL	FALSE	Dieser Parameter wird von der FM gesetzt, wenn Bedienparameter über OP geändert wurden. Wenn READ_VAR = TRUE ist und wenn diese Anzeige von der FM gesetzt wird, dann liest die Anweisung PID_FM die Parameter SP_OP_ON, LMNOP_ON, SP_OP und LMN_OP aus der FM und legt sie im Instanz-DB ab. Die Anweisung übernimmt somit den Bedienzustand der FM. Nach dem Lesevorgang wird der Parameter auf FALSE gesetzt.
QMOD_F	42.0	BOOL	FALSE	Die Anweisung überprüft das erfolgreiche Schreiben und Lesen eines Datensatzes. Bei erkannten Fehlern wird der Ausgang "Baugruppenfehler" gesetzt. Fehlerursache kann sein: Eine falsche Baugruppenadresse am Parameter MOD_ADDR, eine falsche Kanalnummer am Parameter CHANNEL oder eine defekte Baugruppe.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter PID_FM

Tabelle 4-229

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
[1] Vorbelegungswerte der Baugruppe nach dem ersten Anlauf des PID_FM mit COM_RST = TRUE				
COM_RST	44.0	BOOL	FALSE	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung PID_FM einen Initialisierungslauf durch. Dabei werden die Reglerparameter (dies sind alle Parameter nach cont_par) aus der FM gelesen und im Instanz-DB abgelegt. Weiterhin findet eine Prüfung der Parameter MOD_ADDR und CHANNEL auf Zulässigkeit statt. Nach dem Initialisierungslauf wird der Parameter auf FALSE gesetzt.
LOAD_OP	44.1	BOOL	FALSE	Ist der Durchgangsparameter "Betriebsparameter in FM 355 laden" gesetzt, werden die Betriebsparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
READ_VAR	44.2	BOOL	FALSE	Ist der Durchgangsparameter "Variablen von FM 355 lesen" gesetzt, werden die Ausgangsparameter von der Baugruppe gelesen und der Durchgangsparameter rückgesetzt.
LOAD_PAR	44.3	BOOL	FALSE	Ist der Durchgangsparameter "Regelparameter in FM 355 laden" gesetzt, werden die Regelparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
op_par	46.0	WORD	W#16#3130[1]	Der Parameter op_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Bedienparameter, die zur Baugruppe übertragen werden, wenn LOAD_OP = TRUE gesetzt wird. Das Ende der Bedienparameter wird durch cont_par gekennzeichnet.
SP_RE	48.0	REAL	0.0	Ein externer Sollwert wird am Eingang "externer Sollwert" mit dem Regler verschaltet.
LMN_RE	52.0	REAL	0.0	Ein externer Stellwert wird am Eingang "externer Stellwert" mit dem Regler verschaltet. Zulässig sind Werte von -100.0...100.0 (%).
SP_OP_ON	56.0	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Sollwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
SAFE_ON	56.1	BOOL	FALSE	Ist der Eingang "Sicherheitsstellung einnehmen" gesetzt, wird ein Sicherheitswert als Stellwert übernommen. Hinweis: Die Stellsignalbearbeitung über LMNDN_OP, LMNUP_OP und LMNSOPON beim Schrittregler ist höher prior als der Sicherheitsstellwert.
LMNOP_ON	56.2	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Stellwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
LMNTRKON	56.3	BOOL	FALSE	Ist der Eingang "Nachführen (LMN über AE)" gesetzt, wird der Stellwert einem Analogeingang (AE) nachgeführt. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
LMN_REON	56.4	BOOL	FALSE	Ist der Eingang "externer Stellwert einschalten" gesetzt, wird der externe Stellwert LMN_RE als Stellwert übernommen.
LMNRHSRE	56.5	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "oberes Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRHSRE = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMNRLSRE	56.6	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "unteres Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRLSRE = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
LMNSO-PON	56.7	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignalbedienung einschalten". Ist das Bit gesetzt, werden die Signale LMNUP_OP und LMNDN_OP als Stellwertsignale übernommen. (Nur beim Schrittregler)
LMNUP_OP	57.0	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignal Hoch Bedienung". Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Hoch Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNDN_OP	57.1	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignal Tief Bedienung". Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Tief Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNRS_ON	57.3	BOOL	FALSE	Steht keine Stellungsrückmeldung zur Verfügung, so kann diese simuliert werden. Am Eingang "Simulation der Stellungsrückmeldung einschalten" wird die Funktion eingeschaltet. Auch die Anweisungen des PID Self-Tuners haben Zugriff auf diesen Parameter, da zur Optimierung zumindest ein simulierter Stellwert nötig ist, wenn ein Schrittregler ohne Stellungsrückmeldung parametriert wurde. Der simulierte Wert wird am Parameter LMN_A angezeigt. Beim Einschalten der Simulation wird der Wert des Parameters LMNRSVAL als Startwert gesetzt. ACHTUNG: Die Simulation weicht mit zunehmender Zeit immer stärker vor der wirklichen Stellungsrückmeldung ab. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung)
FUZID_ON	57.4	BOOL	FALSE	Am Eingang "Fuzzy-Identifikation einschalten" wird die Identifikation des Fuzzy-Algorithmus eingeschaltet.
SP_OP	58.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Sollwert Bedienung". Ist das Bit SP_OP_ON gesetzt, wird der Wert "Sollwert Bedienung" als Sollwert übernommen.
LMN_OP	62.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Stellwert Bedienung". Ist das Bit LMNOP_ON gesetzt, wird der Wert "Stellwert Bedienung" als Stellwert übernommen. Zulässig sind Werte von -100.0...100.0 (%)
LMNRSVAL	66.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Startwert der simulierten Stellungsrückmeldung". Am Parameter wird der Startwert der Simulation eingetragen. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100.0...100.0 (%)
cont_par	70.0	WORD	W#16#3130 [1]	Der Parameter cont_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Reglerparameter, die aus der FM gelesen und im Instanz-DB abgelegt werden, wenn COM_RST = TRUE ist, und die zur FM übertragen werden, wenn LOAD_PAR = TRUE gesetzt wird. Das Ende der Reglerparameter ist das Ende des Instanz-DB.
P_SEL	72.0	BOOL	TRUE [1]	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der Proportionalanteil ist eingeschaltet, wenn der Eingang "P-Anteil einschalten" gesetzt ist.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PFDB_SEL	72.1	BOOL	FALSE [1]	Im PID-Algorithmus lassen sich die P- und D-Anteile in die Rückführung legen. Der Proportionalanteil liegt in der Rückführung, wenn der Eingang "P-Anteil in Rückführung schalten" gesetzt ist.
MONER-SEL	72.2	BOOL	FALSE [1]	Der Regler besitzt einen Grenzwertmelder, der entweder für den Istwert oder die Regeldifferenz angewendet werden kann. Ist der Eingang "Überwachung: Istwert = 0, Regeldifferenz = 1" gesetzt, wird die Regeldifferenz überwacht.
D_EL_SEL	74.0	INT	0 [1]	Das D-Glied im PID-Algorithmus kann an einen separaten Eingang gelegt werden. Am Eingang "Eingang für D-Glied" wird dieser ausgewählt. 0: Regeldifferenz 1 bis 4: Analogeingang 1 bis 4 17: negativer Istwert, D-Anteil in der Rückführung
SP_HLM	76.0	REAL	100.0 [1]	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert obere Begrenzung" gibt die obere Begrenzung an. SP_HLM > SP_LLM
SP_LLM	80.0	REAL	0.0 [1]	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert untere Begrenzung" gibt die untere Begrenzung an. SP_LLM < SP_HLM
H_ALM	84.0	REAL	100.0 [1]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Alarm" gibt die oberste Grenze an. H_ALM > H_WRN
H_WRN	88.0	REAL	90.0 [1]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Warnung" gibt die zweite Grenze von oben an. Zulässig sind Werte von H_ALM bis L_WRN
L_WRN	92.0	REAL	10.0 [1]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Warnung" gibt die zweite Grenze von unten an. Zulässig sind Werte von H_WRN bis L_ALM
L_ALM	96.0	REAL	0.0 [1]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Alarm" gibt die unterste Grenze an. L_ALM < L_WRN
HYS	100.0	REAL	1.0 [1]	Zur Vermeidung von Flackern der Überwachungsanzeigen kann am Eingang "Hysterese" eine Hysterese parametriert werden. HYS ≥ 0.0
DEADB_W	104.0	REAL	0.0 [1]	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W ≥ 0.0
GAIN	108.0	REAL	1.0 [1]	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI	112.0	REAL	3000.0 [1]	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. Bei TI = 0 ist der Integrierer ausgeschaltet TI = 0.0 oder TI ≥ 0.5
TD	116.0	REAL	0.0 [1]	Der Eingang "Differenzierzeit" bestimmt das Zeitverhalten des Differenzierers. Bei TD = 0 ist der Differenzierer abgeschaltet. TD = 0.0 oder TD ≥ 1.0

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
TM_LAG	120.0	REAL	5.0 1)	Der Algorithmus des D-Anteils beinhaltet eine Verzögerung, die am Eingang "Verzögerungszeit des D-Anteils" parametriert werden kann. TM_LAG ≥ 0.5
LMN_SAFE	124.0	REAL	0.0 1)	Für den Stellwert kann ein Sicherheitswert am Eingang "Sicherheitsstellwert" parametriert werden. Zulässig sind Werte von -100.0 bis 100.0 (%)
LMN_HLM	128.0	REAL	100.0 1)	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von LMN_LLM bis 100.0 (%)
LMN_LLM	132.0	REAL	0.0 1)	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100.0 bis LMN_HLM (%)
MTR_TM	136.0	REAL	60.0 1)	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils von Anschlag zu Anschlag eingetragen. (Nur beim Schrittregler) Zulässig sind Werte von MTR_TM ≥ 0.001
PULSE_TM	140.0	REAL	0.2 1)	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden. (Nur beim Schrittregler oder Impulsregler) Zulässig sind Werte PULSE_TM ≥ 0.0
BREAK_TM	144.0	REAL	0.2 1)	Am Parameter "Mindestpausendauer" kann eine minimale Impulspausenlänge parametriert werden. (Nur beim Schrittregler oder Impulsregler) Zulässig sind Werte BREAK_TM ≥ 0.0

Hinweis

Mit LOAD_PAR = TRUE werden alle Regelparameter dauerhaft in das EEPROM der FM 355 geladen.

Mit LOAD_OP = TRUE wird von den Betriebsparametern nur der Sollwert SP_RE dauerhaft in das EEPROM der FM 355 geladen. Alle übrigen Betriebsparameter werden beim Anlauf der FM 355 mit den Werten 0 bzw. FALSE vorbelegt.

Das EEPROM der Baugruppe könnte durch zu häufige Schreibvorgänge zerstört werden. Um dies zu verhindern, wird von der Baugruppe nach dem Beschreiben des EEPROM ein erneutes Beschreiben um 30 Minuten verzögert.

Siehe auch

Beschreibung PID_FM (Seite 6207)

Blockschaltbilder PID_FM

Überblick

In den folgenden Abbildungen wird dargestellt, an welchen Stellen die Parameter der Anweisung PID_FM wirken.

Die Parameter wirken bei Dreikomponentenregler, Verhältnis-/Mischungsregler und Festwert- oder Kaskadenregler an den gleichen Stellen. Dies gilt auch für die Parameter, die gleichermaßen beim kontinuierlichen Regler, beim Regler mit Impulsausgang sowie beim Schrittregler vorhanden sind. Daher sind nicht alle Blockschaltbilder dargestellt und nicht in allen Bildern alle Parameter eingezeichnet.

Die Parameter der Anweisung PID_FM sind jedoch alle in den Bildern enthalten – bis auf die Parameter MOD_ADDR, CHANNEL, QMOD_F, QPARA_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_VAR, LOAD_OP.

Wirkung der Eingangsparameter

Regeldifferenzbildung beim Festwert- oder Kaskadenregler

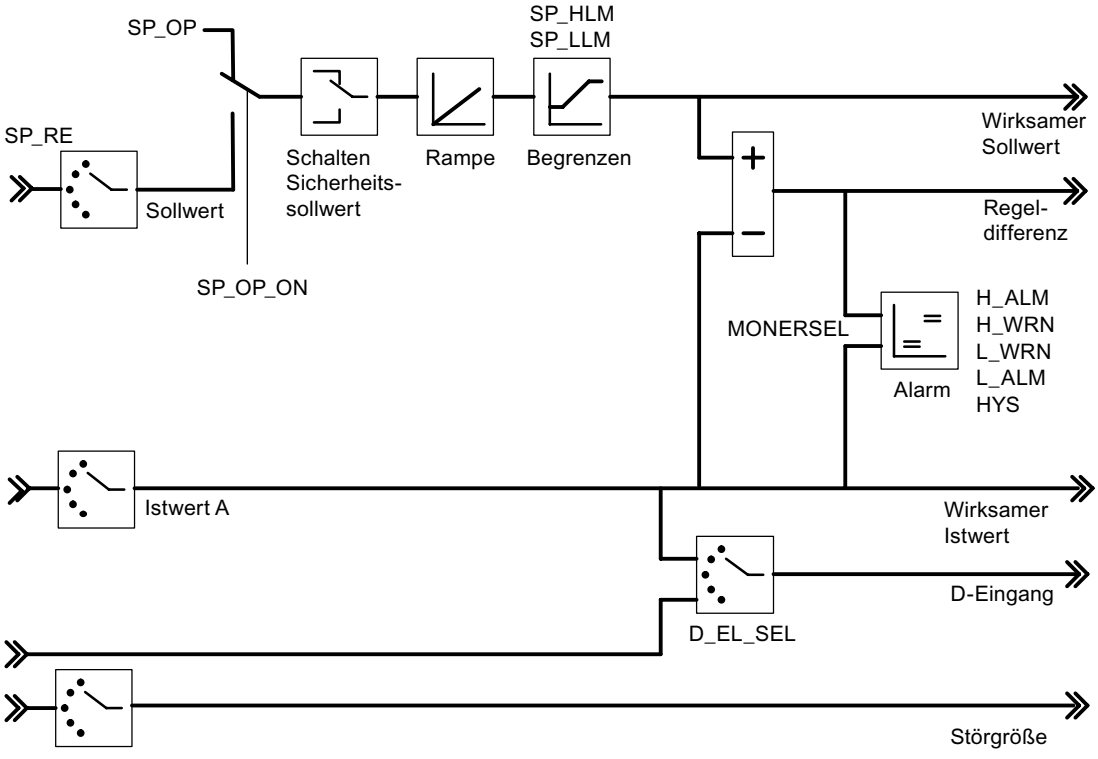

Blockschaltbild des Regelalgorithmus

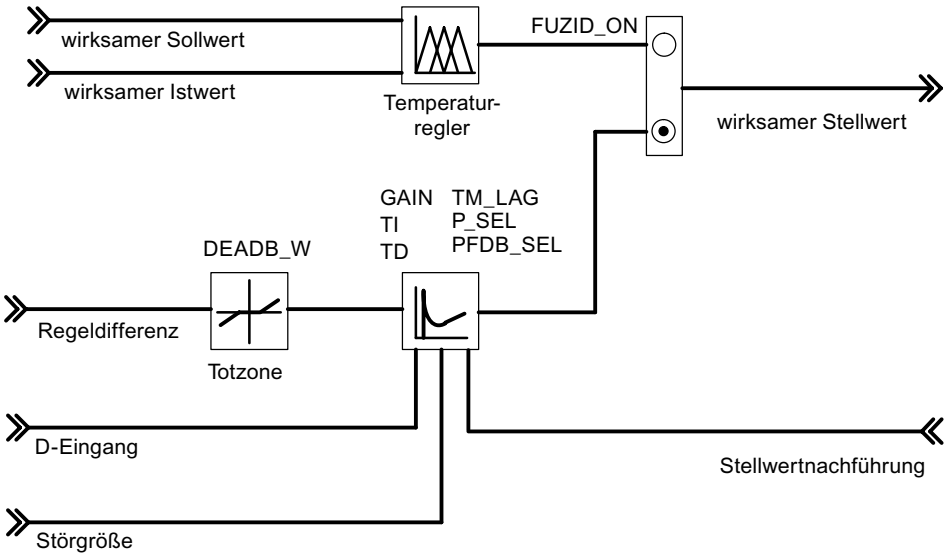

Reglerausgang des K-Reglers

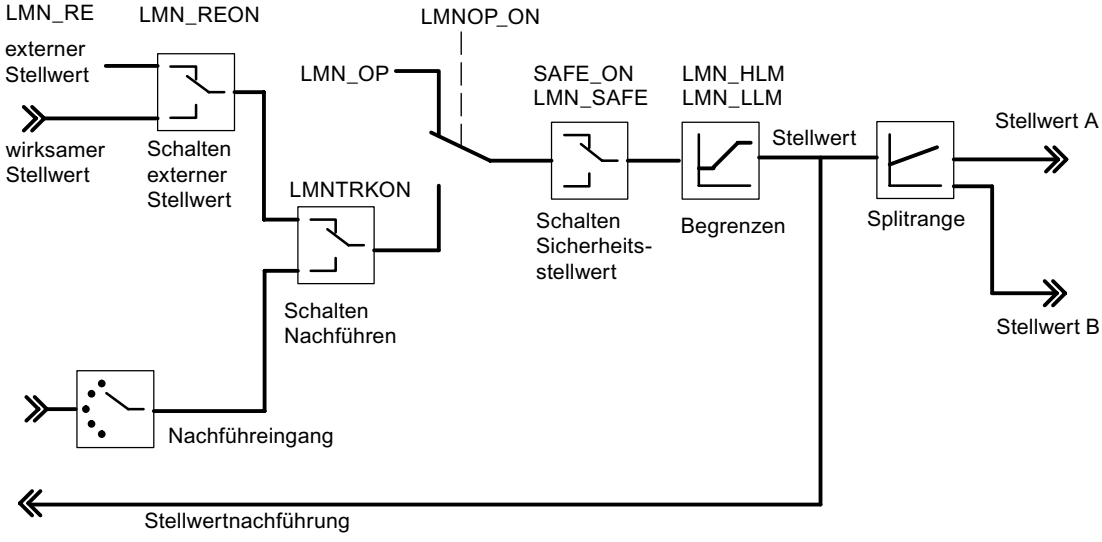

Reglerausgang des S-Reglers (Betriebsart Impulsregler)

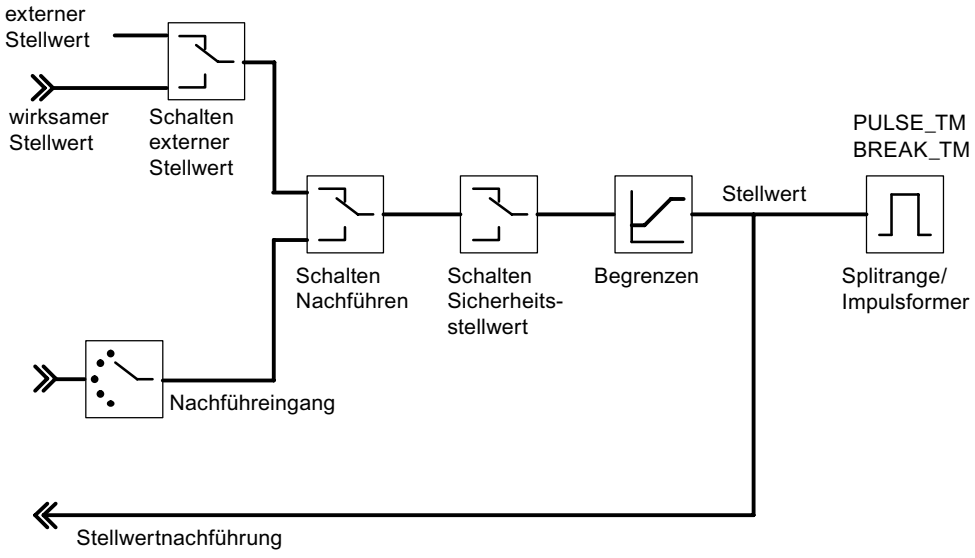

Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung)

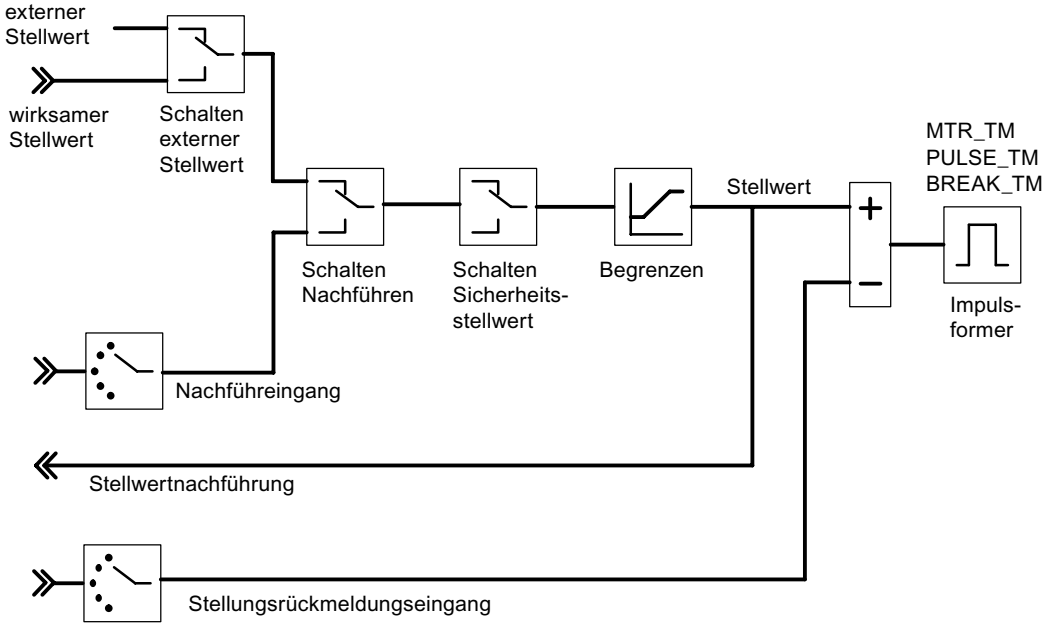

Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung)

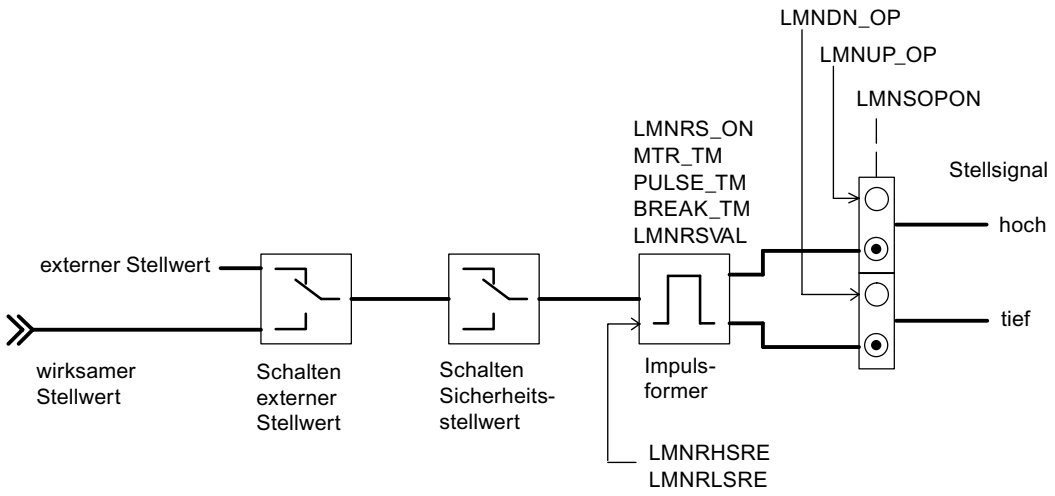

Erzeugung der Ausgangsparameter

In den folgenden Abbildungen wird dargestellt, an welchen Stellen in der Baugruppe die Ausgangsparameter der Anweisung PID_FM erzeugt werden.

Regeldifferenzbildung beim Festwert- oder Kaskadenregler

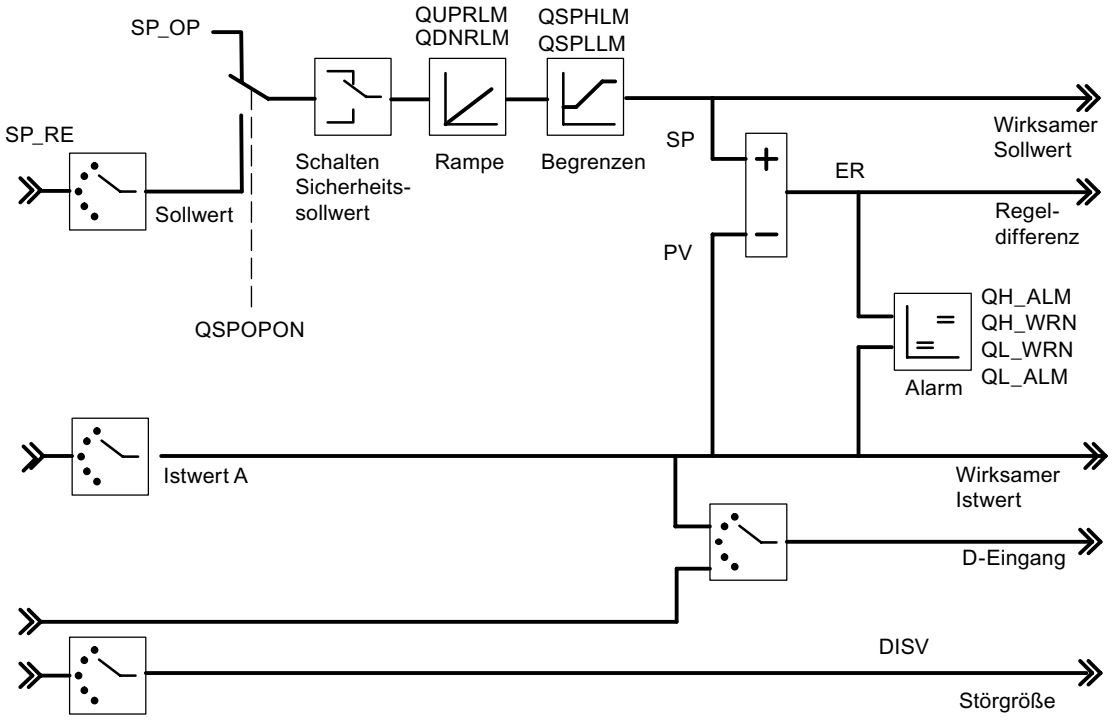

Blockschaltbild des Regelalgorithmus

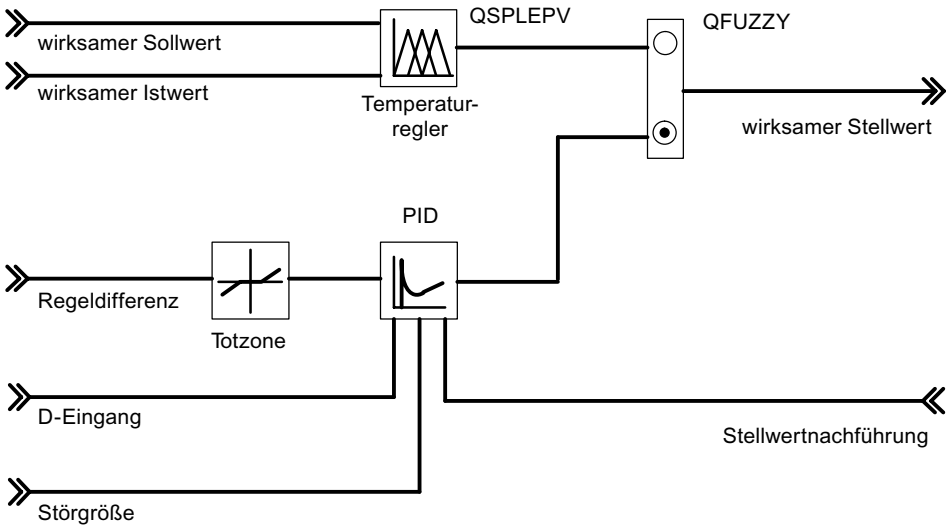

Reglerausgang des K-Reglers

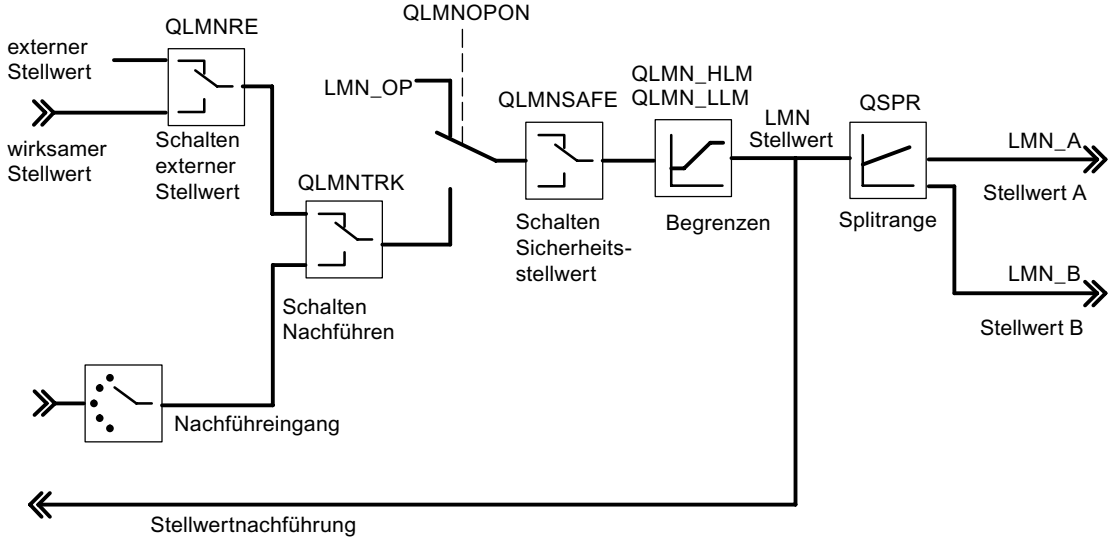

Reglerausgang des S-Reglers (Betriebsart Impulsregler)

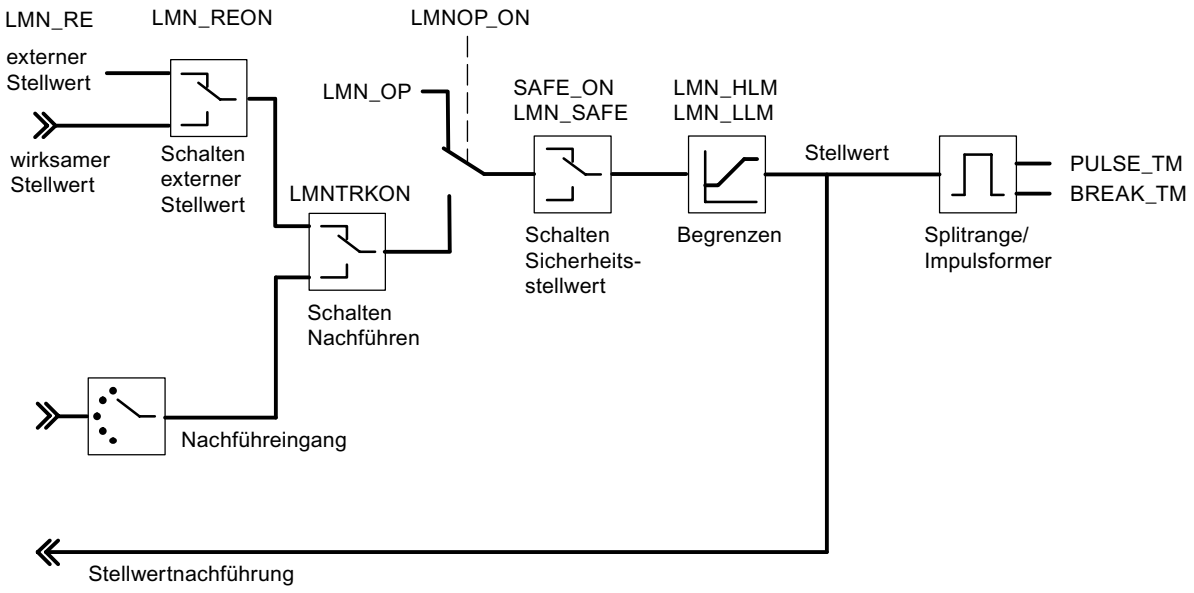

Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung)

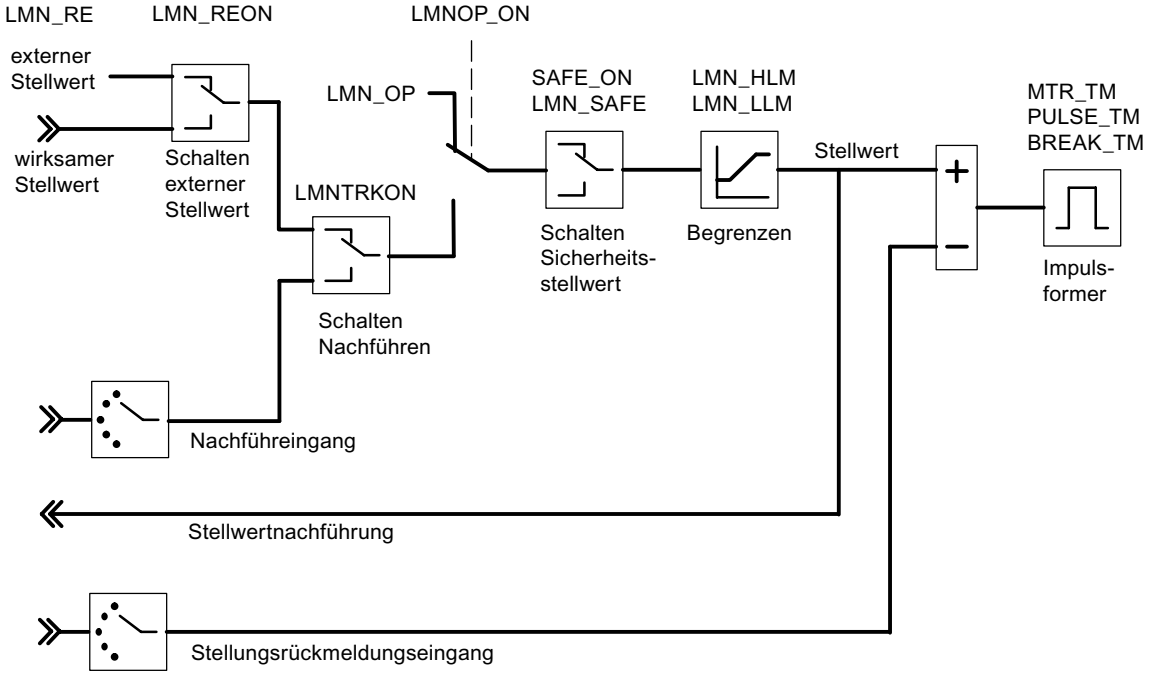

Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung)

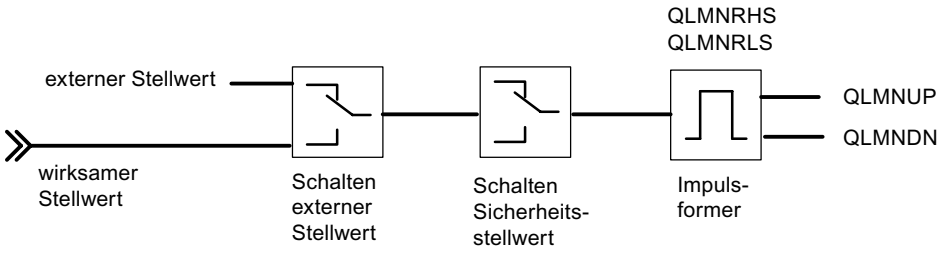

Siehe auch

Beschreibung PID_FM (Seite 6207)

FUZ_355

Beschreibung FUZ_355

Die Anweisung FUZ_355 dient dazu, die Reglerparameter des Fuzzy-Temperaturreglers aus der FM 355 auszulesen und zu schreiben.

Diese Funktion eignet sich für folgende Anwendungsfälle:

- Übertragen der durch Identifikation ermittelten Reglerparameter zur FM 355 nach Baugruppentausch
- Anpassen der FM 355 an unterschiedliche Regelstrecken

Hinweis

Die durch eine Identifikation von der FM 355 ermittelten Parameter dürfen Sie nicht ändern, da sie für die Regelstrecke optimiert sind.

Arbeitsweise

Mit den Parametern READ_PAR und LOAD_PAR steuern Sie die Arbeitsweise der Anweisung FUZ_355.

Wenn READ_PAR = TRUE, liest FUZ_355 die Parameter aller Temperaturrelger der FM 355 und legt sie im Instanz-DB ab. Nach erfolgreichem Auslesen der Temperaturreglerparameter setzt FUZ_355 den Parameter READ_PAR auf FALSE.

Wenn LOAD_PAR = TRUE, schreibt FUZ_355 die Parameter aller Temperaturregler der FM 355 aus dem Instanz-DB zur FM 355. Nach erfolgreicher Übertragung der Parameter setzt FUZ_355 den Parameter LOAD_PAR auf FALSE.

Aufruf

FUZ_355 benötigt keinen Initialisierungslauf.

Beim Einsatz der FM 355 in dezentraler Peripherie kann es einige Aufrufzyklen dauern, bis die Parameter READ_PAR und LOAD_PAR wieder auf FALSE gesetzt werden. Rufen Sie die

Anweisung FUZ_355 daher bedingt auf, solange READ_PAR = TRUE oder LOAD_PAR = TRUE.

FUZ_355 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Anlauf

Im Anlauf der CPU sollten Sie dem Parameter LOAD_PAR der Anweisung FUZ_355 setzen und danach den Baustein im zyklischen Programm bedingt aufrufen, solange LOAD_PAR = TRUE ist.

Eingangsparameter FUZ_355

Tabelle 4-230

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter FUZ_355

Tabelle 4-231

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	2.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
PARAFFUZ	4.0	WORD	0	Am Parameter PARAFFUZ wird ein durch die Anweisung FUZ_355 erzeugter Parametrierfehler wie folgt angezeigt: High Byte von PARAFFUZ = 01: es liegt ein Parametrierfehler vor. High Byte von PARAFFUZ = 00: es liegt kein Parametrierfehler vor. Im Low-Byte steht der Offset des Parameters, der den Parametrierfehler verursacht hat, gerechnet ab der statischen Variablen FUZ_PAR[1]. PARAFFUZ = W#16#0104 bedeutet z. B., dass der zweite Parameter fehlerhaft ist

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter FUZ_355

Tabelle 4-232

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
READ_PAR	6.0	BOOL	FALSE	Wenn der Parameter READ_PAR gesetzt wird, dann werden die Fuzzy-Parameter aus der Baugruppe ausgelesen und in statischen Variablen des Instanz-DBs abgelegt.
LOAD_PAR	6.1	BOOL	FALSE	Wenn der Parameter LOAD_PAR gesetzt wird, dann werden die Fuzzy-Parameter aus statischen Variablen des Instanz-DBs ausgelesen und zur Baugruppe übertragen.

FORCE355

Beschreibung FORCE355

Die Anweisung FORCE355 dient zur Simulation (Forcen) der Analog- und Digitaleingangswerte zur Unterstützung der Inbetriebsetzung.

Simulieren der Digitalwerte

Die Simulation der Werte für die Digitaleingänge eins bis acht wird eingeschaltet über die Schalter S_DION[i], wobei 1 ≤ i ≤ 8 ist. Die Simulationswerte geben Sie über die Parameter DI_SIM[i] vor. Wenn S_DION[i] = TRUE, wird statt des Werts am Digitaleingang i der Baugruppe der Wert DI_SIM[i] verwendet. Die LEDs I1 bis I8 zeigen auch bei Simulation immer den Zustand des zugehörigen Digitaleingangs an.

Simulieren der Analogwerte

Die Simulation der Analogwerte für die Kanäle eins bis vier wird eingeschaltet über die Schalter S_AION[i] bzw. S_PVON[i], wobei 1 ≤ i ≤ 4 ist.

Die Simulationswerte für die Kanäle eins bis vier geben Sie über die Parameter PV_SIM[i] vor.

Sie können die Simulationswerte an zwei Stellen wirksam werden lassen:

- S_AION[i] = TRUE (1 ≤ i ≤ 4)
  An Stelle des Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.
- S_PVON[i] = TRUE (1 ≤ i ≤ 4)
  An Stelle des aufbereiteten Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.

In der nachstehenden Abbildung ist dargestellt, an welcher Stelle der simulierte Analogwert wirksam wird.

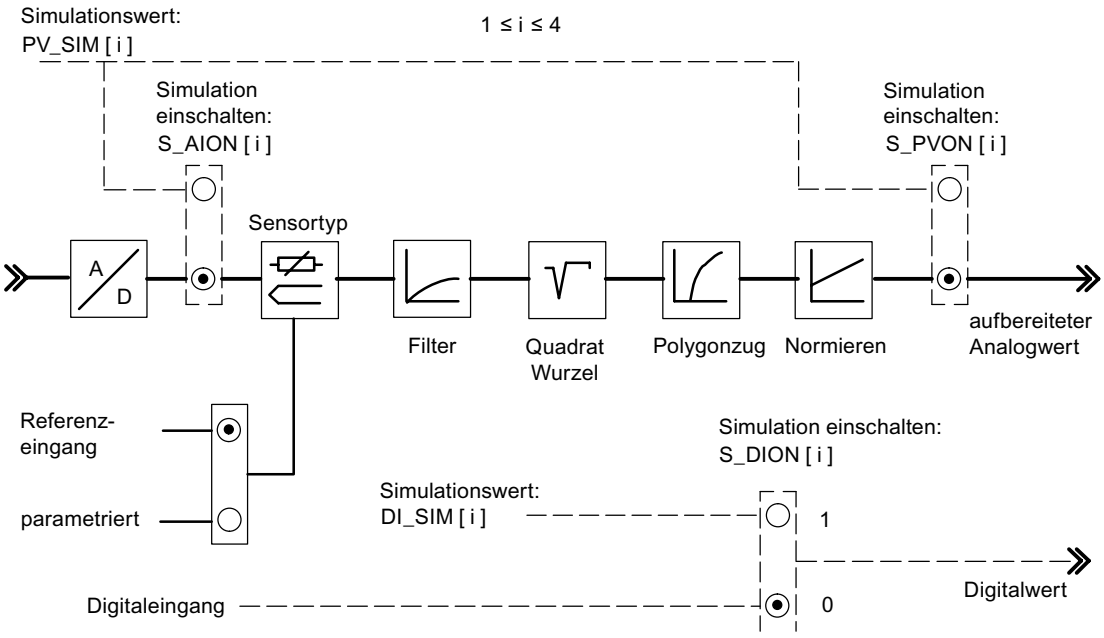

Das Einschalten und die Vorgabe der Simulationswerte (Forcen) erfolgen nicht über die Parametrieroberfläche. Daher sind die entsprechenden Schalter und Verbindungslinien gestrichelt gezeichnet.

Aufruf

FORCE355 benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

FORCE355 muss im selben OB aufgerufen werden, wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Anlaufverhalten

Bei Wiederanlauf der FM 355 nach Netz-Aus stehen die Simulationsschalter auf der FM 355 wieder auf FALSE.

Eingangsparameter FORCE355

Tabelle 4-233

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
S_AION	0.0	ARRAY [1..4] of BOOL	FALSE	Wenn z. B. der Schalter S_AION[1] auf TRUE gesetzt wird, dann wird an Stelle des Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.
S_PVON	2.0	ARRAY [1..4] of BOOL	FALSE	Wenn z. B. der Schalter S_PVON[1] auf TRUE gesetzt wird, dann wird an Stelle des aufbereiteten Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PV_SIM	4.0	ARRAY [1..4] of REAL	0.0	Am Eingang PV_SIM[1] wird z. B. der Simulationswert für den Analogeingang 1 vorgegeben. Falls S_PVON = TRUE ist, dann wird hier der aufbereitete Analogeingangswert vorgegeben. Falls S_PVON = FALSE und S_AION = TRUE sind, dann wird hier der Analogeingangswert in mA bzw. mV vorgegeben, der noch durch die Aufbereitungsfunktionen in einen aufbereiteten Wert gewandelt wird. Zulässig sind 0.0 bis 20.0 [mA], -1500 bis +10000 [mV] oder ein technischer Wertebereich
S_DION	20.0	ARRAY [1..8] of BOOL	FALSE	Wenn z. B. S_DION[1] auf TRUE gesetzt wird, dann wird als Digitalwert an Stelle des Digitaleingangs 1 der Baugruppe der Wert DI_SIM[1] verwendet.
DI_SIM	22.0	ARRAY [1..8] of BOOL	FALSE	Simulationswert für Digitaleingang
MOD_ADDR	24.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter FORCE355

Tabelle 4-234

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	26.0	WORD		RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

READ_355

Beschreibung READ_355

Die Anweisung READ_355 dient zum Auslesen der Digital- und Analogeingangswerte zur Unterstützung der Inbetriebsetzung.

Folgende Werte werden angezeigt.

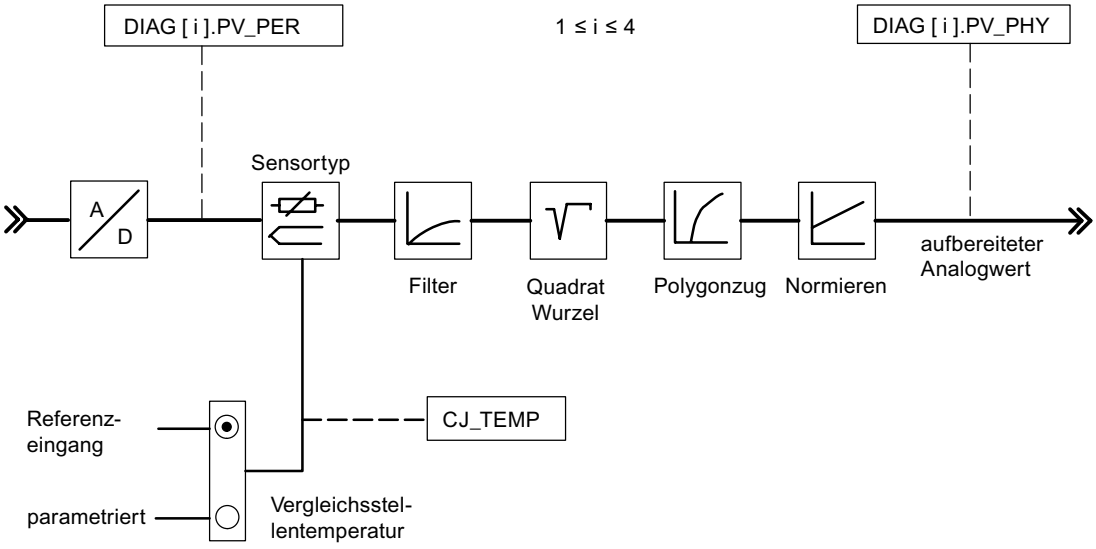

Aufruf

READ_355 benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

READ_355 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Eingangsparameter READ_355

Tabelle 4-235

Parameter	Adres-se	Daten-typ	Vorbe-legung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter READ_355

Tabelle 4-236

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CJ_TEMP	2.0	REAL	0.0	Am Parameter CJ_TEMP steht die an der Vergleichsstelle gemessene Vergleichsstellentemperatur in Grad C oder in Grad F (je nachdem, welche Temperatureinheit parametriert wurde). Wenn kein Sensortyp "Thermoelement" parametriert wurde oder wenn bei allen Analogeingängen die parametrierte Vergleichsstellentemperatur gewählt wurde, dann wird am Parameter CJ_TEMP 0.0 angezeigt.
STAT_DI	6.0	ARRAY [1..8] of BOOL	FALSE	An den Parametern STAT_DI[1] bis STAT_DI[8] werden die tatsächlichen Zustände der Digitaleingänge 1 bis 8 angezeigt, auch wenn diese simuliert werden.
DIAG[x].PV_PER	(Kanalnummer) x 8	ARRAY [1..4] of STRUCT	0.0	An den Parametern DIAG[1].PV_PER bis DIAG[4].PV_PER wird der Wert der Analogeingänge eins bis vier in der Einheit mA bzw. mV angezeigt. Falls über die Anweisung FORCE355 die Simulation des Analogeingangswertes eingeschaltet wurde, wird der simulierte Wert angezeigt.
DIAG[x].PV_PHY	(Kanalnummer) x 8 + 4	ARRAY [1..4] of STRUCT	0.0	An den Parametern DIAG[1].PV_PHY bis DIAG[4].PV_PHY wird der aufbereitete Analogeingangswert eins bis vier in physikalischer Einheit angezeigt. Falls über die Anweisung FORCE355 die Simulation des aufbereiteten physikalischen Analogwertes eingeschaltet wurde, wird der simulierte Wert angezeigt.
RET_VALU	40.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

CH_DIAG

Beschreibung CH_DIAG

Die Anweisung CH_DIAG dient dazu, zusätzliche kanalspezifische Diagnosevariablen aus der Baugruppe auszulesen.

Aufruf

CH_DIAG benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

CH_DIAG muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Eingangsparameter CH_DIAG

Tabelle 4-237

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässige Werte sind 1 - 4.

Siehe auch

Beschreibung CH_DIAG (Seite 6230)

Ausgangsparameter CH_DIAG (Seite 6231)

Parameter IDSTATUS (Seite 6232)

Blockschaltbilder CH_DIAG (Seite 6233)

Ausgangsparameter CH_DIAG

Tabelle 4-238

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP_R	4.0	REAL	0.0	Der Parameter ist bei eingestelltem Verhältnisregler mit dem Eingangswert des Sollwertes belegt.
PV_R	8.0	REAL	0.0	Der Parameter ist nur bei eingestelltem Verhältnisregler mit folgendem Wert belegt: (Istwert A - Sollwert Offset) / Istwert D
DIF_I	12.0	REAL	0.0	Am Parameter DIF_I wird die Eingangsgröße des D-Anteils angezeigt. Dies ist z. B. von Interesse, wenn als Eingangsgröße des D-Anteils ein Analogeingang parametriert ist.
TRACKPER	16.0	REAL	0.0	Am Parameter TRACKPER wird die Eingangsgröße angezeigt, auf die die Stellgröße nachgeführt wird, wenn der Regler auf Stellgrößennachführung geschaltet wird.
IDSTATUS	20.0	WORD	0.0	Am AUTOHOTSPOT wird der Zustand der Identifikation angezeigt.
LMN_P	22.0	REAL	0.0	Am Parameter LMN_P wird der P-Anteil der Stellgröße angezeigt.
LMN_I	26.0	REAL	0.0	Am Parameter LMN_I wird der I-Anteil der Stellgröße angezeigt.
LMN_D	30.0	REAL	0.0	Am Parameter LMN_D wird der D-Anteil der Stellgröße angezeigt.
RET_VALU	34.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Parameter IDSTATUS

Der Parameter IDSTATUS liefert Informationen über den Zustand der Identifikation und enthält die vier Hexadezimalwerte X, A, I und F.

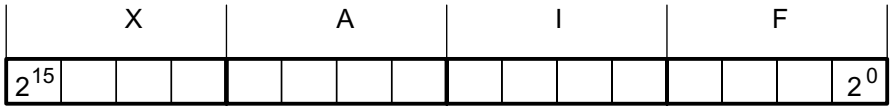

Wert	Beschreibung
X	ohne Bedeutung (immer 0)
A	Aktionsnummer • 0 = Handbetrieb (bzw. kein Regelbetrieb); • 2 = Regelung; • 4 = Optimierung eingeschaltet (FUZID_ON = true); • 6 = Übergangszustand vom Handbetrieb nach 2 oder 4;
I	Anzeige "Identifikation läuft" und "Parameter ermittelt, aber noch nicht im EEPROM abgelegt" • 0 = Identifikation läuft nicht, keine neuen Parameter ermittelt • 1 = Identifikation läuft, keine neuen Parameter ermittelt • 2 = Identifikation läuft nicht, neue Parameter ermittelt, aber noch nicht im EEPROM abgelegt • 3 = Identifikation läuft, neue Parameter ermittelt, aber noch nicht im EEPROM abgelegt
F	Fehlernummer • 0 = kein Fehler • 4 = Zu großer Sprung des Istwertes während der Identifikation • 5 = Verhältnis von Verzugszeit zu Streckenzeitkonstante zu groß oder stark nichtlineares Verhalten der Regelstrecke. • 6 = Temperaturgefälle oder -steigung bei Start der Identifikation zu groß. Strecke nicht ausreichend beruhigt

Blockschaltbilder CH_DIAG

Folgende Diagnosewerte der Regeldifferenz werden angezeigt.

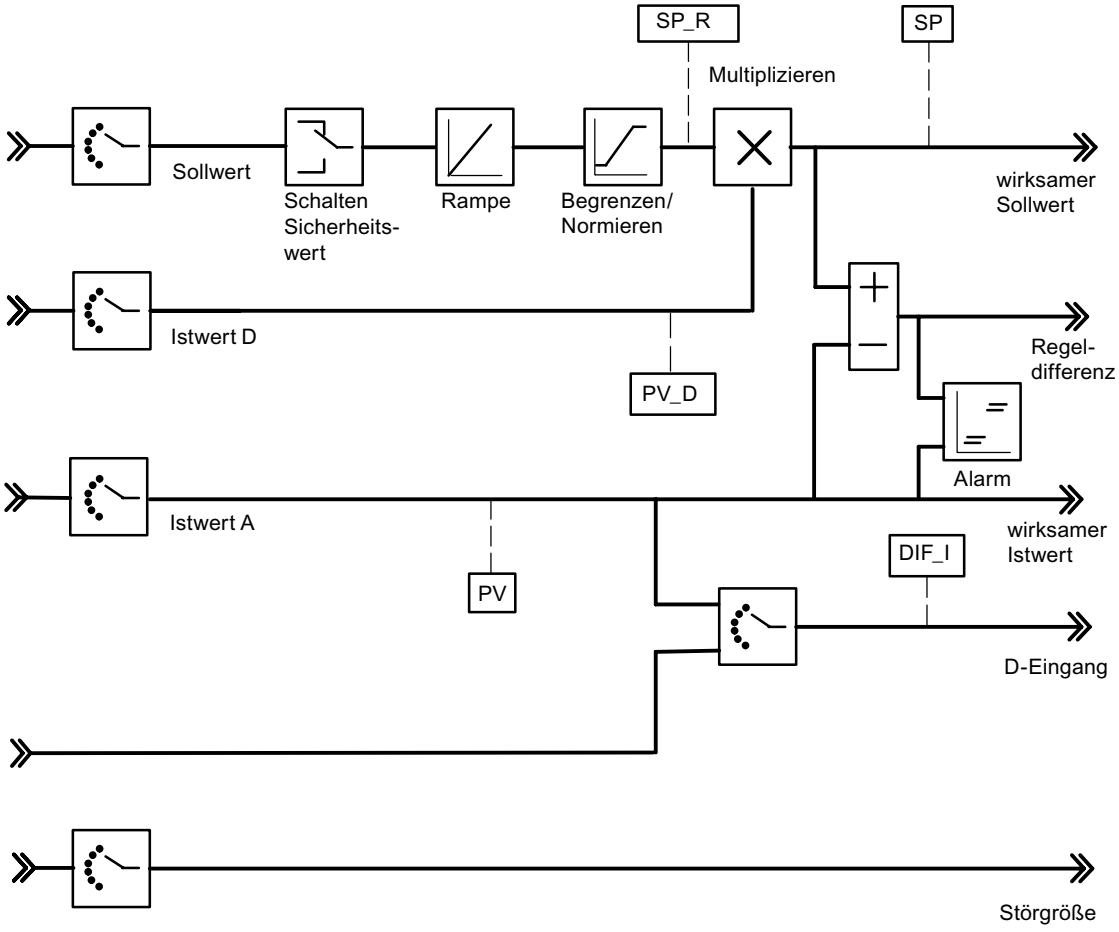

Folgende Werte des Regelalgorithmus werden angezeigt.

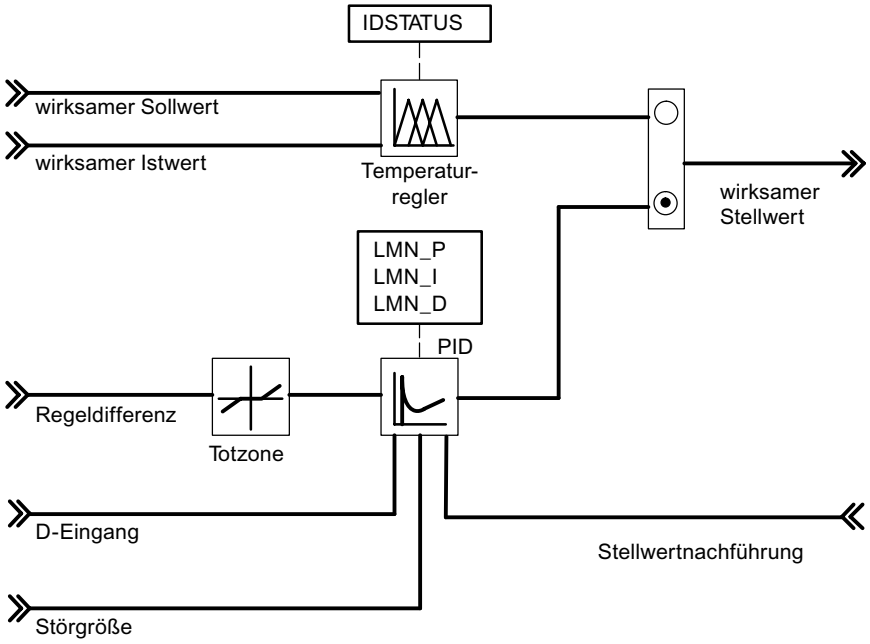

Folgende Werte des K- oder S-Reglers werden angezeigt.

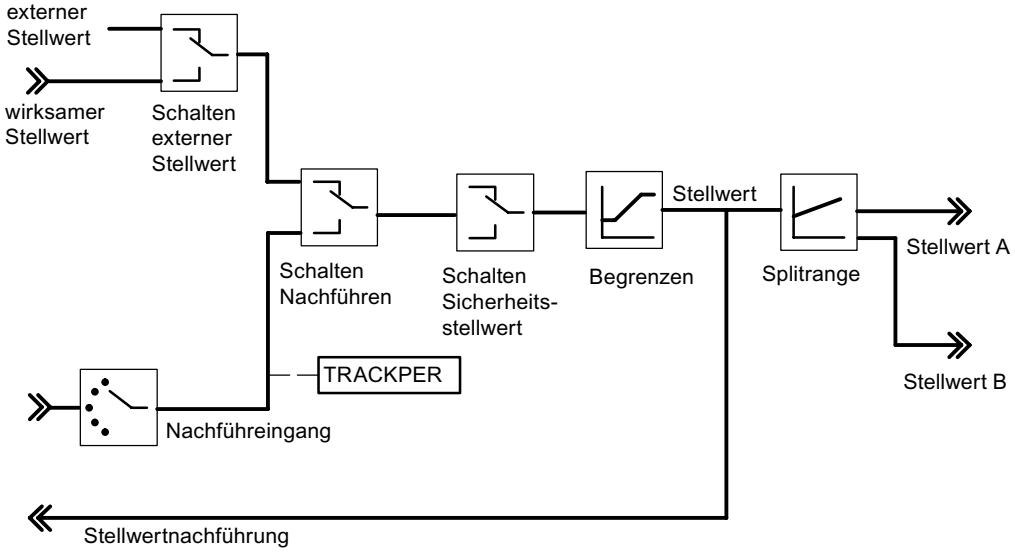

PID_PAR

Beschreibung PID_PAR

Die Anweisung PID_PAR dient zum Online-Ändern von Parametern, die nicht über die Anweisung PID_FM vorgegeben werden können. Welche Parameter Sie ändern können entnehmen Sie der Tabelle änderbare Parameter (Seite 6237).

Aufruf

PID_PAR benötigt einen Initialisierungslauf. Er wird automatisch angestoßen, wenn noch nie die Systemdaten (SDB Vorbelegungsdaten der FM355) von der Anweisung PID_PAR gelesen wurden. Sie können die Initialisierung mit COM_RST=TRUE auch selbst starten, was üblicherweise im OB100 gemacht wird, da nach STOP-RUN der CPU die Systemdaten auf die FM355 übertragen werden. Der Initialisierungsvorgang dauert mehrere Zyklen. Während der Initialisierung werden keine Daten über WRREC auf die FM 355 übertragen. Nach der Initialisierung setzt der Baustein automatisch den Parameter COM_RST zurück.

Laufzeit zu sparen, sollte PID_PAR nicht zyklisch aufgerufen werden sondern nur dann, wenn Parameter geändert werden sollen. COM_RST muss dann FALSE sein.

PID_PAR muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Bei Einsatz der FM 355 in dezentraler Peripherie kann es einige Aufrufzyklen dauern, bis die Parameter zur FM 355 übertragen worden sind. Solange die Übertragung noch nicht abgeschlossen ist, hat der Parameter BUSY den Wert TRUE. Sie sollten PID_PAR solange wiederholt aufrufen, bis BUSY = FALSE und RET_VALU = 0 ist.

Arbeitsweise

Im Initialisierungslauf liest PID_PAR die Parameter aus den Systemdaten und legt sie in statischen Variablen ab. PID_PAR ändert pro Aufruf eine dieser REAL-Variablen und eine dieser INT-Variablen. Am Eingangsparameter INDEX_R bzw. INDEX_I geben Sie die Indexnummer des Parameters an, den Sie ändern wollen. Am Eingangsparameter VALUE_R bzw. VALUE_I geben Sie die neuen Werte an. PID_PAR überträgt den kompletten Datensatz mit den geänderten Variablen an die FM 355.

Um weitere Parameter zu ändern, müssen Sie **denselben** Instanz-DB mehrfach nacheinander mit COM_RST = FALSE und mit unterschiedlichen Indexnummern aufrufen.

Der Parameter COM_RST ist ein Eingangsparameter, der von PID_PAR nicht zurückgesetzt wird.

Hinweis

Bitte beachten Sie, dass Parameter, die Sie mit PID_PAR geändert haben, beim Anlauf mit den Parametern aus den Systemdaten überschrieben werden.

Anlauf

Beim Anlauf der CPU und beim STOP-RUN-Übergang der CPU werden die Parameter an der FM 355 mit den Parametern aus dem SDB der CPU überschrieben.

Beispiel

Sie wollen im Betrieb die Hochlaufzeit der Rampe für die Führungsgröße ändern sowie abhängig vom Prozesszustand als Istwert verschiedene Analogeingangswerte verwenden.

- Rufen Sie im Anlauf der CPU PID_PAR auf mit COM_RST = TRUE.
- Zum Parametrieren der Hochlaufzeit der Rampe für die Führungsgröße auf 10.0 rufen Sie im laufenden Betrieb PID_PAR auf mit INDEX_R = 30, VALUE_R = 10.0 und INDEX_I = 0.
- Wenn Sie als Istwert den Analogeingangswert 4 der Baugruppe parametrieren wollen, rufen Sie im laufenden Betrieb PID_PAR auf mit INDEX_R = 0, INDEX_I = 50 und VALUE_I = 4.

Siehe auch

Änderbare Parameter (Seite 6237)

Eingangsparameter PID_PAR (Seite 6236)

Ausgangsparameter PID_PAR (Seite 6237)

Eingangsparameter PID_PAR

Tabelle 4-239

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL	TRUE	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung PID_PAR einen Initialisierungslauf durch. Dabei werden die Parameter aus den Systemdaten der CPU gelesen und im Instanz-DB abgelegt.
MOD_ADDR	2.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	4.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässig sind die Werte 1 bis 4.
INDEX_R	6.0	INT	0.0	• 0: kein Real-Parameter wird geändert • 1 bis 48: Jeder Wert entspricht einem änderbaren Real-Parameter.
VALUE_R	8.0	REAL	0.0	abhängig vom jeweiligen Parameter
INDEX_I	12.0	INT	0.0	• 0: kein INT-Parameter wird geändert. • 49 bis 61: Jeder Wert entspricht einem änderbaren INT-Parameter.
VALUE_I	14.0	INT	0.0	abhängig vom jeweiligen Parameter

Siehe auch

Beschreibung PID_PAR (Seite 6234)

Ausgangsparameter PID_PAR (Seite 6237)

Ausgangsparameter PID_PAR

Tabelle 4-240

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	16.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
BUSY	18.0	BOOL	FALSE	Wenn BUSY = TRUE, dann sind die Parameter noch nicht von der Baugruppe übernommen worden (bei dezentraler Peripherie). Der Aufruf der Anweisung PID_PAR sollte dann im nächsten Zyklus wiederholt werden.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Änderbare Parameter

Tabelle 4-241 Liste der mit PID_PAR zu ändernden REAL- und INT-Parameter auf einer FM 355

Datentyp	Beschreibung	Indexnummer
–	Kein Parameter angewählt	0
REAL	Filterzeitkonstante für Analogeingang	1
REAL	Messende (100%)	2
REAL	Messanfang (0%)	3
REAL	Polygonzug, Stützwert 1 Eingangsseite	4
REAL	Polygonzug, Stützwert 2 Eingangsseite	5
REAL	Polygonzug, Stützwert 3 Eingangsseite	6
REAL	Polygonzug, Stützwert 4 Eingangsseite	7
REAL	Polygonzug, Stützwert 5 Eingangsseite	8
REAL	Polygonzug, Stützwert 6 Eingangsseite	9
REAL	Polygonzug, Stützwert 7 Eingangsseite	10
REAL	Polygonzug, Stützwert 8 Eingangsseite	11
REAL	Polygonzug, Stützwert 9 Eingangsseite	12
REAL	Polygonzug, Stützwert 10 Eingangsseite	13
REAL	Polygonzug, Stützwert 11 Eingangsseite	14
REAL	Polygonzug, Stützwert 12 Eingangsseite	15
REAL	Polygonzug, Stützwert 13 Eingangsseite	16
REAL	Polygonzug, Stützwert 1 Ausgangsseite	17
REAL	Polygonzug, Stützwert 2 Ausgangsseite	18
REAL	Polygonzug, Stützwert 3 Ausgangsseite	19
REAL	Polygonzug, Stützwert 4 Ausgangsseite	20
REAL	Polygonzug, Stützwert 5 Ausgangsseite	21

Datentyp	Beschreibung	Indexnummer
REAL	Polygonzug, Stützwert 6 Ausgangsseite	22
REAL	Polygonzug, Stützwert 7 Ausgangsseite	23
REAL	Polygonzug, Stützwert 8 Ausgangsseite	24
REAL	Polygonzug, Stützwert 9 Ausgangsseite	25
REAL	Polygonzug, Stützwert 10 Ausgangsseite	26
REAL	Polygonzug, Stützwert 11 Ausgangsseite	27
REAL	Polygonzug, Stützwert 12 Ausgangsseite	28
REAL	Polygonzug, Stützwert 13 Ausgangsseite	29
REAL	Hochlaufzeit der Rampe für Führungsgröße	30
REAL	Sicherheitsführungsgröße oder Sicherh. Führungsgr.-Verh.	31
REAL	Offset für Sollwertverknüpfung (Verhältnis-/Mischungsregler)	32
REAL	Faktor für Istwert B (Dreikomponentenregler)	33
REAL	Faktor für Istwert C (Dreikomponentenregler)	34
REAL	Offset für Istwertverknüpfung (Dreikomponentenregler)	35
REAL	Faktor für Störgrößenverknüpfung	36
REAL	Arbeitspunkt	37
REAL	Aggressivität beim Fuzzy-Regler	38
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal A	39
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal A	40
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal A	41
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal A	42
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal B	43
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal B	44
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal B	45
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal B	46
REAL	Mindestimpulsdauer	47
REAL	Mindestpausendauer	48
INT	Auswahl der Führungsgröße SP bzw. SP_RE für den Regler 0: Sollwert SP_RE der Anweisung 1 bis 4: Analogeingangswert 1 bis 4 17 bis 20: Stellgröße (LMN) von Regler 1 bis 4	49
INT	Auswahl der Haupt-Regelgröße Istwert A für den Regler 0: Istwert A = 0.0 1 bis 4: Analogeingangswert 1 bis 4	50
INT	Auswahl der Hilfs-Regelgröße Istwert B für den Regler 0: Istwert B = 0.0 1 bis 4: Analogeingangswert 1 bis 4	51
INT	Auswahl der Hilfs-Regelgröße Istwert C für den Regler 0: Istwert C = 0.0 1 bis 4: Analogeingangswert 1 bis 4	52

Datentyp	Beschreibung	Indexnummer
INT	Auswahl der Hilfs-Regelgröße Istwert D für den Regler 0: Istwert D = 0.0 1 bis 4: Analogeingangswert 1 bis 4 17 bis 20: Stellgröße (LMN) von Regler 1 bis 4	53
INT	Auswahl der Störgröße DISV für den Regler 0: Störgröße = 0.0 1 bis 4: Analogeingangswert 1 bis 4	54
INT	Auswahl der Stellungsnachführung TRACK_PER für den Regler 0: Stellungsnachführung = 0.0 1 bis 4: Analogeingangswert 1 bis 4	55
INT	Auswahl der Stellungsnachführung LMNR_PER für den Regler 0: Stellungsnachführung = 0.0 1 bis 4: Analogeingangswert 1 bis 4	56
INT	Auswahl des Signals zur Umschaltung auf Sicherheitswert für den Stellwert des Reglers 0: Nur Vorgabe über Parameter SAFE_ON der Anweisung PID_FM 1 bis 8: Vorgabe über Parameter SAFE_ON der Anweisung PID_FM verodert mit Digitaleingang 1 bis 8	57
INT	Auswahl des Signals zur Umschaltung auf Nachführfunktion des Stellwertes des Reglers 0: Nur Vorgabe über Parameter LMNTRKON der Anweisung PID_FM 1 bis 8: Vorgabe über Parameter LMNTRKON der Anweisung PID_FM verodert mit Digitaleingang 1 bis 8	58
INT	Auswahl des Signals zur Umschaltung des Stellwertes des Reglers auf LMN_RE 0: Nur Vorgabe über Parameter LMN_REON der Anweisung PID_FM 1 bis 8: Vorgabe über Parameter LMN_REON der Anweisung PID_FM verodert mit Digitaleingang 1 bis 8	59
INT	Auswahl des oberen Anschlagsignals der Stellungsrückmeldung 0: Nur Vorgabe über Parameter LMNRHSRE der Anweisung PID_FM 1 bis 8: Vorgabe über Parameter LMNRHSRE der Anweisung PID_FM verodert mit Digitaleingang 1 bis 8	60
INT	Auswahl des unteren Anschlagsignals der Stellungsrückmeldung 0: Nur Vorgabe über Parameter LMNRLSRE der Anweisung PID_FM 1 bis 8: Vorgabe über Parameter LMNRLSRE der Anweisung PID_FM verodert mit Digitaleingang 1 bis 8	61

Siehe auch

Eingangsparameter PID_PAR (Seite 6236)

Ausgangsparameter PID_PAR (Seite 6237)

CJ_T_PAR

Beschreibung CJ_T_PAR

Die Anweisung CJ_T_PAR dient zum Online-Ändern der parametrierten Vergleichsstellentemperatur. Dies ist erforderlich, wenn eine Temperaturregelung mit mehreren FM 355 mit Thermoelementeingängen betrieben werden soll, ohne dass an jede FM 355 ein Pt 100 angeschlossen werden muss.

Aufruf

CJ_T_PAR benötigt einen Initialisierungslauf. Er muss hierzu einmalig im Anlauf der CPU mit Parameter COM_RST = TRUE aufgerufen werden.

CJ_T_PAR wird normalerweise zyklisch aufgerufen. COM_RST sollte dabei aus Laufzeitgründen FALSE sein. Der Parameter COM_RST ist ein Eingangsparameter, der nicht zurückgesetzt wird.

CJ_T_PAR muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355 zugreifen.

Bei Einsatz der FM 355 in dezentraler Peripherie kann es einige Aufrufzyklen dauern, bis der Parameter zur FM 355 übertragen worden ist. Solange die Übertragung noch nicht abgeschlossen ist, hat der Parameter BUSY den Wert TRUE. Sie sollten daher bei der Änderung von Parametern die Anweisung CJ_T_PAR solange wiederholt aufrufen, bis BUSY = FALSE ist.

Beispiel

Wenn bei einer Extruderregelung mit mehr als vier Heizzonen die Vergleichsstellentemperatur mit einer FM 355 gemessen wird, dann kann diese über READ_355 am Parameter CJ_TEMP ausgelesen werden und bei den anderen FM 355 über CJ_T_PAR parametriert werden.

Siehe auch

Eingangsparameter CJ_T_PAR (Seite 6241)

Ausgangsparameter CJ_T_PAR (Seite 6241)

Eingangsparameter CJ_T_PAR

Tabelle 4-242

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL	FALSE	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung CJ_T_PAR einen Initialisierungslauf durch. Dabei werden die Parameter aus den Systemdaten der CPU gelesen und im Instanz-DB abgelegt.
MOD_ADDR	2.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CJ_T	4.0	REAL	0.0	Am Parameter CJ_T kann die Vergleichsstellentemperatur vorgegeben werden. Der zulässige Wertebereich ist abhängig vom Sensortyp.

Siehe auch

Beschreibung CJ_T_PAR (Seite 6240)

Ausgangsparameter CJ_T_PAR (Seite 6241)

Ausgangsparameter CJ_T_PAR

Tabelle 4-243

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	8.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
BUSY	10.0	BOOL	FALSE	Wenn BUSY = TRUE, dann sind die Parameter noch nicht von der Baugruppe übernommen worden (bei dezentraler Peripherie). Der Aufruf der Anweisung PID_PAR sollte dann im nächsten Zyklus wiederholt werden.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

DB für HMI

Beschreibung DB 101 bis 104

Um bei CPU-Ausfall die FM 355 mit dem OP zu bedienen, benötigen Sie die Datenbausteine 101 bis 104. Über MPI können Sie maximal drei Verbindungen von der FM 355 zu OPs aufbauen.

Bedienen der FM 355 mit dem OP ist nur möglich im Zustand STOP der CPU oder bei CPU-Ausfall.

Beobachten der FM 355 mit dem OP ist immer möglich.

Die Variablenschnittstelle der FM 355 enthält vier Datenbausteine mit den Bausteinnummern 101 bis 104 für die Reglerkanäle 1 bis 4 (siehe folgendes Bild).

Hinweis

Der Inhalt der Datenbausteine 101 bis 104 spiegelt nicht automatisch den auf der FM 355 wirksamen Parameterwert wieder. Mit Hilfe des OP geänderte Parameter werden nur nach Setzen der Bedienbits LOAD_PAR bzw. LOAD_OP in die FM 355 übernommen.

Wenn Sie durch OP-Bedienung einen Parameter ändern, ohne das entsprechenden Bedienbit zu setzen, dann steht zwar der geänderte Parameterwert im Datenbaustein, die FM 355 arbeitet jedoch intern noch mit dem unveränderten Altwert des Parameters.

Nach Setzen der Bedienbits und Übernahme der Parameter in die FM 355 werden die Bedienbits LOAD_PAR bzw. LOAD_OP von der FM 355 zurückgesetzt.

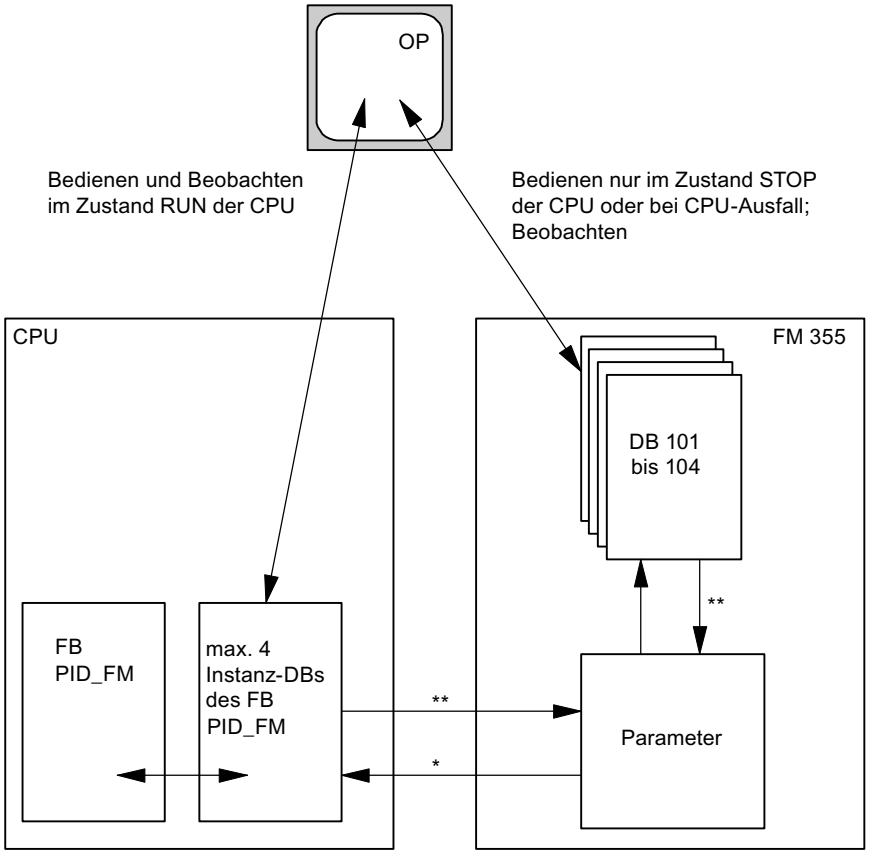

* gesteuert durch Parameter READ_VAR des Instanz-DB

** gesteuert durch die Parameter LOAD_OP und LOAD_PAR

Siehe auch

Beschreibung PID_FM (Seite 6207)

Eingangsparameter DB 101 bis 104

Tabelle 4-244

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP_HLM	0.0	REAL	100.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert obere Begrenzung" gibt die obere Begrenzung an. SP_HLM > SP_LLM
SP_LLM	4.0	REAL	0.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert untere Begrenzung" gibt die untere Begrenzung an. SP_LLM < SP_HLM
H_ALM	8.0	REAL	100.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Alarm" gibt die oberste Grenze an. H_ALM > H_WRN
H_WRN	12.0	REAL	90.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Warnung" gibt die zweite Grenze von oben an. Zulässig sind Werte von H_ALM bis L_WRN
L_WRN	16.0	REAL	10.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Warnung" gibt die zweite Grenze von unten an. Zulässig sind Werte von H_WRN bis L_ALM
L_ALM	20.0	REAL	0.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Alarm" gibt die unterste Grenze an. L_ALM < L_WRN
HYS	24.0	REAL	1.0	Zur Vermeidung von Flackern der Überwachungsanzeigen kann am Eingang "Hysterese" eine Hysterese parametriert werden. HYS ≥ 0.0
DEADB_W	28.0	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W ≥ 0.0
GAIN	32.0	REAL	1.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI	36.0	REAL	3000.0	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. Bei TI = 0 ist der Integrierer ausgeschaltet TI = 0.0 oder TI ≥ 0.5
TD	40.0	REAL	0.0	Der Eingang "Differenzierzeit" bestimmt das Zeitverhalten des Differenzierers. Bei TD = 0 ist der Differenzierer abgeschaltet. TD = 0.0 oder TD ≥ 1.0
TM_LAG	44.0	REAL	5.0	Der Algorithmus des D-Anteils beinhaltet eine Verzögerung, die am Eingang "Verzögerungszeit des D-Anteils" parametriert werden kann. TM_LAG ≥ 0.5
LMN_SAFE	48.0	REAL	0.0	Für den Stellwert kann ein Sicherheitswert am Eingang "Sicherheitsstellwert" parametriert werden. Zulässig sind Werte von -100.0 bis 100.0 (%).

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMN_HLM	52.0	REAL	100.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von LMN_LLM bis 100.0 (%).
LMN_LLM	56.0	REAL	0.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100.0 bis LMN_HLM (%).
MTR_TM	60.0	REAL	60.0	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils von Anschlag zu Anschlag eingetragen. (Nur beim Schrittregler) MTR_TM ≥ 0.001
PULSE_TM	64.0	REAL	0.2	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden. (Nur beim Schrittregler oder Impulsregler) PULSE_TM ≥ 0.0
BREAK_TM	68.0	REAL	0.2	Am Parameter "Mindestpausendauer" kann eine minimale Impulspausenlänge parametriert werden. (Nur beim Schrittregler oder Impulsregler) BREAK_TM ≥ 0.0
SP_RE	72.0	REAL	0.0	Ein externer Sollwert wird am Eingang "externer Sollwert" mit dem Regler verschaltet.
LMN_RE	76.0	REAL	0.0	Ein externer Stellwert wird am Eingang "externer Stellwert" mit dem Regler verschaltet.
LMNRSVAL	80.0	REAL	0.0	Vom Dialog Inbetriebnahme haben Sie Zugriff auf den Eingang "Startwert der simulierten Stellungsrückmeldung". Am Parameter wird der Startwert der Simulation eingetragen. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100.0 bis 100.0 (%).
SAFE_ON	84.0	BOOL	FALSE	Ist der Eingang "Sicherheitsstellung einnehmen" gesetzt, wird ein Sicherheitswert als Stellwert übernommen. Hinweis: Die Stellsignalbearbeitung über LMNDN_OP, LMNUP_OP und LMNSOPON beim Schrittregler ist höher prior als der Sicherheitsstellwert.
LMNTRKON	84.1	BOOL	FALSE	Ist der Eingang "Nachführen (LMN über AE)" gesetzt, wird der Stellwert einem Analogeingang (AE) nachgeführt. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
LMN_REON	84.2	BOOL	FALSE	Ist der Eingang "externer Stellwert einschalten" gesetzt, wird der externe Stellwert LMN_RE als Stellwert übernommen.
LMNRHSRE	84.3	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "oberes Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRHSRE = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)
LMNRLSRE	84.4	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "unteres Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRLSRE = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMNSOPON	84.5	BOOL	FALSE	Ist das Bit an dem Eingang "Stellwertsignalbedienung einschalten" gesetzt, werden die Signale LMNUP_OP und LMNDN_OP als Stellwertsignale übernommen. (Nur beim Schrittregler)
LMNUP_OP	84.6	BOOL	FALSE	Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Hoch Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNDN_OP	84.7	BOOL	FALSE	Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Tief Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
MONERSEL	85.0	BOOL	FALSE	Der Regler besitzt einen Grenzwertmelder, der entweder für den Istwert oder die Regeldifferenz angewendet werden kann. Ist der Eingang "Überwachung: Istwert = 0, Regeldifferenz = 1" gesetzt, wird die Regeldifferenz überwacht.
LMNRS_ON	85.1	BOOL	FALSE	Steht keine Stellungsrückmeldung zur Verfügung, so kann diese simuliert werden. Am Eingang "Simulation der Stellungsrückmeldung einschalten" wird die Funktion eingeschaltet. ACHTUNG: Die Simulation weicht mit zunehmender Zeit immer stärker vor der wirklichen Stellungsrückmeldung ab. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung)
FUZID_ON	85.2	BOOL	FALSE	Am Eingang "Fuzzy-Identifikation einschalten" wird die Identifikation des Fuzzy-Algorithmus eingeschaltet.
SPINT_EN	85.3	BOOL	FALSE	Der Eingang "Bedieneingang: extern = 0, intern = 1" bestimmt den Eingang, der als Sollwert an die Baugruppe übertragen wird. SPINT_EN = TRUE: SP_INT wird übertragen. SPINT_EN = FALSE: SP_RE wird übertragen.
P_SEL	85.4	BOOL	TRUE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der Proportionalanteil ist eingeschaltet, wenn der Eingang "P-Anteil einschalten" gesetzt ist.
PFDB_SEL	85.5	BOOL	FALSE	Im PID-Algorithmus lassen sich die P- und D-Anteile in die Rückführung legen. Der Proportionalanteil liegt in der Rückführung, wenn der Eingang "P-Anteil in Rückführung schalten" gesetzt ist.
D_EL_SEL	86.0	INT	0	Das D-Glied im PID-Algorithmus kann an einen separaten Eingang gelegt werden. Am Eingang "Eingang für D-Glied" wird dieser ausgewählt. 0: Regeldifferenz 1 bis 4: Analogeingang 1 bis 4 17: negativer Istwert

Siehe auch

Beschreibung PID_FM (Seite 6207)

Ausgangsparameter DB 101 bis 104

Tabelle 4-245

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP	94.0	REAL	0.0	Am Ausgang "Sollwert" wird der effektiv wirkende Sollwert ausgegeben.
PV	98.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	102.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.
DISV	106.0	REAL	0.0	Am Ausgang "Störgröße" wird die effektiv wirkende Störgröße ausgegeben. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN	110.0	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Parameter LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_A	114.0	REAL	0.0	Am Ausgang "Stellwert A der Splitrangefunktion / Stellungsrückmeldung" wird beim kontinuierlichen Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Beim Schrittregler ohne analoge Stellungsrückmeldung wird die simulierte Stellungsrückmeldung angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_B	118.0	REAL	0.0	Am Ausgang "Stellwert B der Splitrangefunktion" wird beim kontinuierlichen Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).
QH_ALM	122.0	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Alarm angesprochen" wird das Überschreiten der Grenze H_ALM gemeldet.
QH_WRN	122.1	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Warnung angesprochen" wird das Überschreiten der Grenze H_WRN gemeldet.
QL_WRN	122.2	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Warnung angesprochen" wird das Unterschreiten der Grenze L_WRN gemeldet.
QL_ALM	122.3	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Alarm angesprochen" wird das Unterschreiten der Grenze L_ALM gemeldet.
QLMN_HLM	122.4	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Stellwertes angesprochen" meldet die Überschreitung der oberen Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QLMN_LLM	122.5	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Stellwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QSPINTON	122.6	BOOL	FALSE	Der Ausgang "interner Sollwert eingeschaltet" zeigt an, dass SP_INT an die Baugruppe übertragen wurde.
QPARA_F	123.0	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang "Parametrierfehler" angezeigt. Diesen Parametrierfehler können Sie auch im Dialog "Online & Diagnose" der Baugruppe auslesen.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QCH_F	123.1	BOOL	FALSE	Der Ausgang "Kanalfehler" wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPARA_F = 1 oder QMOD_F = 1 gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	123.2	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "positive Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	123.3	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "negative Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	123.4	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Sollwertes angesprochen" meldet die Überschreitung der oberen Begrenzung.
QSP_LLM	123.5	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Sollwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung.
QSPOPON	123.6	BOOL	FALSE	Am Ausgang "Sollwertbedienung eingeschaltet" wird angezeigt, ob der Sollwert über das Dialog "Inbetriebnahme" bedient wird. Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
QLMNSAFE	123.7	BOOL	FALSE	Ist der Ausgang "Sicherheitsbetrieb" gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNOPON	124.0	BOOL	FALSE	Am Ausgang "Stellwertbedienung eingeschaltet" wird angezeigt, ob der Stellwert über das Dialog "Inbetriebnahme" bedient wird. Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
QLMNTRK	124.1	BOOL	FALSE	Der Ausgang "Nachführbetrieb" zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	124.2	BOOL	FALSE	Der Ausgang "Hand = 1; Automatik = 0" zeigt an, ob der Stellwert auf den externen Stellwert LMN_RE gesetzt wird (Hand = 1) oder nicht.
QLMNR_HS	124.3	BOOL	FALSE	Der Ausgang "oberes Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am oberen Anschlag befindet. QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)
QLMNR_LS	124.4	BOOL	FALSE	Der Ausgang "unteres Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am unteren Anschlag befindet. QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
QLMNR_ON	124.5	BOOL	FALSE	Der Ausgang "Stellungsrückmeldung eingeschaltet" zeigt die eingestellte Betriebsart "Schrittregler mit Stellungsrückmeldung" bzw. "Schrittregler ohne Stellungsrückmeldung" an.
QFUZZY	124.6	BOOL	FALSE	Ist der Ausgang "PID-Algorithmus = 0; Fuzzy = 1" gesetzt, arbeitet der Regler mit dem Fuzzy-Algorithmus.
QSPLEPV	124.7	BOOL	FALSE	Der Ausgang "Anzeige vom FUZZY-Regler: Sollwert < Istwert" wird bei eingeschaltetem Fuzzy-Regler gesetzt, wenn der Sollwert kleiner als der wirksame Istwert ist.
QSPR	125.0	BOOL	FALSE	Ist der Ausgang "Split-Range-Betrieb" gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QLMNUP	125.1	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Hoch". (Nur beim Schrittregler oder Impulsregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNDN	125.2	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QBACKUP	125.4	BOOL	FALSE	0= kein Backup-Zustand (CPU in RUN) 1= Backup-Zustand (CPU in STOP oder ausgefallen)
QID	125.5	BOOL	FALSE	QID = TRUE zeigt an, dass eine Identifizierung läuft (nicht dass sie eingeschaltet ist). Nach dem Ende der Identifizierung kann das Identifizierungsergebnis über den Parameter IDSTATUS der Anweisung CH_DIAG ausgelesen werden.
QMAN_FC	125.6	BOOL	FALSE	Der Regler ist ein Führungsregler, der durch Handbetrieb eines Folgereglers auf dessen Istwert nachgeführt wird oder dessen I-Anteil angehalten wird, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind.
RET_VALU	126.0	WORD	0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Beschreibung PID_FM (Seite 6207)

Durchgangsparameter DB 101 bis 104

Tabelle 4-246

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP_INT	128.0	REAL	0.0	Der Durchgangsparameter "interner Sollwert" dient zur Vorgabe eines Sollwertes mittels Bedien- und Beobachtungsfunktionen.
SP_OP	132.0	REAL	0.0	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Sollwert Bedienung". Ist das Bit SP_OP_ON gesetzt, wird der Wert "Sollwert Bedienung" als Sollwert übernommen.
LMN_OP	136.0	REAL	0.0	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Stellwert Bedienung". Ist das Bit LMNOP_ON gesetzt, wird der Wert "Stellwert Bedienung" als Stellwert übernommen. Zulässig sind Werte von -100.0 bis 100.0 (%).
SP_OP_ON	140.0	BOOL	FALSE	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Sollwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
LMNOP_ON	140.1	BOOL	FALSE	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Stellwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LOAD_PAR	140.2	BOOL	FALSE	Ist der Durchgangsparameter "Regelparameter in FM 355 laden" gesetzt, werden die Regelparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
LOAD_OP	140.3	BOOL	FALSE	Ist der Durchgangsparameter "Betriebsparameter in FM 355 laden" gesetzt, werden die Betriebsparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.

Siehe auch

Beschreibung DB 101 bis 104 (Seite 6241)

Beschreibung PID_FM (Seite 6207)

FM 355-2 Temperature Control

FMT_PID

Beschreibung FMT_PID

Die Anweisung FMT_PID dient dazu im laufenden Betrieb:

- Bedienparameter auf die FM 355-2 zu übertragen
- Reglerparameter auf die FM 355-2 zu übertragen
- Prozesswerte aus der FM 355-2 zu lesen

Bedienparameter sind alle statischen Variablen des Instanz-DB in der Struktur OP.

Reglerparameter sind alle statischen Variablen des Instanz-DB in der Struktur PAR.

Prozesswerte sind alle statischen Variablen des Instanz-DB in der Struktur OUT.

Die für den FMT_PID benötigten Daten werden in einem Instanz-DB auf der CPU abgelegt. Der FMT_PID liest programmgesteuert Daten von der FM 355-2 und schreibt Daten programmgesteuert zur FM 355-2.

Aufruf

Die Anweisung FMT_PID muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

FMT_PID wird im Weckalarm-OB aufgerufen. Er benötigt einen Initialisierungslauf, der durch Setzen des Parameters COM_RST = TRUE im Anlauf der CPU veranlasst wird. Nach dem Initialisierungslauf setzt FMT_PID den Parameter COM_RST auf FALSE.

Wenn Sie die Anweisung FMT_PID als Multiinstanz-DB aufrufen, steht Ihnen keine Inbetriebnahmeoberfläche zur Verfügung. Sie müssen die FM 355-2 in diesem Fall über eine Beobachtungstabelle oder mit der Anweisung FMT_TUN in Betrieb nehmen.

Aufruf in dezentraler Peripherie

Bei dezentralem Aufbau und gleichzeitigem Aufruf von FMT_PID und FMT_TUN ist folgendes zu beachten:

- LOAD_OP darf nicht gleichzeitig bei FMT_PID und FMT_TUN gesetzt sein.
- READ_OUT darf nicht gleichzeitig bei FMT_PID und FMT_TUN gesetzt sein.

Begründung: beide FBs greifen über dieselben Datensätze auf die FM 355-2 zu.

Daher muss sichergestellt werden, dass zur gleichen Zeit nur einer der beiden FBs einen Datensatz liest bzw. schreibt.

Anlauf

Beim Anlauf der CPU und beim STOP-RUN-Übergang der CPU werden die Parameter an der FM 355-2 mit den Parametern aus dem SDB der CPU überschrieben.

Verhalten im Fehlerfall

Der Ausgangsparameter RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC. RET_VALU kann ausgewertet werden, wenn die Parameter READ_PAR und LOAD_PAR nicht zurückgesetzt werden.

Beim Aufruf der Anweisung FMT_PID kann ein Peripheriezugriffsfehler auftreten, wenn die FM 355-2 nicht gesteckt ist oder keine Spannungsversorgung hat. In diesem Fall geht die CPU in STOP, wenn kein OB 122 in der CPU geladen ist.

Tritt ein Fehler beim Datensatz lesen/schreiben auf, wird QMOD_F und QCH_F gesetzt. Bei Parametrierfehler wird QPAR_F und QCH_F gesetzt.

Siehe auch

Blockschaltbilder (Seite 6261)

Arbeitsweise FMT_PID

Einfluss der Parameter LOAD_OP, READ_OUT, LOAD_PAR und READ_PAR

Im Initialisierungslauf liest FMT_PID die Parameter aus den Systemdaten und legt sie in seinem Instanz-DB ab. Instanz-DB und Systemdaten sind abgeglichen.

Das folgende Bild zeigt, wie die Funktion der Anweisung durch die Parameter LOAD_OP, READ_OUT, LOAD_PAR und READ_PAR beeinflusst wird.

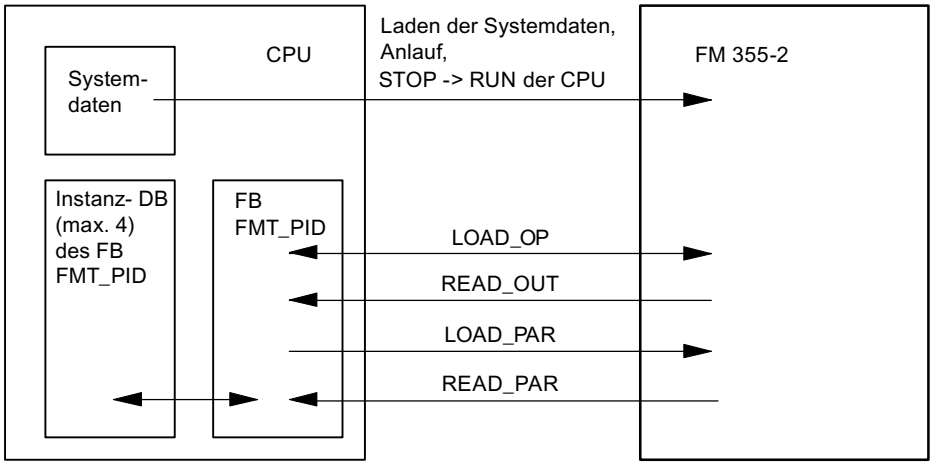

Die folgende Tabelle enthält die Detailinformationen in Abhängigkeit zu den Werten der Parameter.

Parameter	TRUE	FALSE
LOAD_OP	Die statischen Variablen der Struktur OP werden mit den Anweisungen WRREC und RDREC zyklisch zur FM 355-2 übertragen. Die Ausgangsparameter werden aus der FM 355-2 gelesen.	Die statischen Variablen der Struktur OP werden über Ein- und Ausgangsbereiche der Baugruppe übertragen. Nicht möglich, wenn zwei Instanzen der Anweisung FMT_PID auf die gleiche Kanalnummer einer Baugruppe zugreifen.
	READ_VAR wird TRUE gesetzt	READ_VAR wird nicht verändert
	In folgenden Situationen wird LOAD_OP automatisch TRUE: • wenn Sie einen der folgenden Parameter gesetzt haben: TUN_ST, TUN_CST, SAVE_PAR, UNDO_PAR und LOAD_PID • wenn Sie TUN_ON ändern • und wenn Sie den Sollwert SP_RE ändern.	
	Längere Laufzeit	Geringere Laufzeit
	Die Werte sind nach einem Zyklus der CPU oder FM wirksam. Der längere Zyklus ist maßgeblich.	Schon im zentralen Aufbau sind 3 bis 4 Zyklen der CPU oder FM notwendig bis die Werte wirksam werden. Der längere Zyklus ist maßgeblich
	Aller Bedienparameter werden übertragen	Nur die Bedienparameter bis einschließlich LMN_DN werden übertragen. Im Automatikbetrieb wird nur der Sollwert SP_RE übertragen. Im Handbetrieb wird nur der Handstellwert LMN_RE übertragen.

Parameter	TRUE	FALSE
READ_OUT	Die statischen Variablen der Struktur OUT werden zyklisch mit den Anweisungen WRREC und RDREC aus der FM 355-2 gelesen und in der Struktur OUT abgelegt.	Die statischen Variablen der Struktur OUT werden über Ein- und Ausgangsbereiche der Baugruppe übertragen und in der Struktur OUT abgelegt.
	In folgenden Situation wird READ_OUT automatisch TRUE: • während einer Optimierung QTUN_ON = TRUE • Wenn Sie den Sollwert SP_RE ändern	Nicht möglich, wenn zwei Instanzen der Anweisung FMT_PID auf die gleiche Kanalnummer einer Baugruppe zugreifen.
	Längere Bearbeitungszeit	Geringere Bearbeitungszeit
	Alle Parameter werden aus der FM 355 gelesen	Folgende Parameter werden **nicht** aus der FM 355 gelesen • SP (Sollwert aus der FM) • ER (Regeldifferenz) • DISV (Störgröße) • LMN_A • LMN_B • PHASE • STATUS_H • STATUS_C • STATUS_D • ZONE_TUN
READ_PAR	Alle statischen Variablen der Struktur PAR werden von der FM 355-2 gelesen und im Instanz-DB auf der CPU gespeichert.	Keine Übertragung der Reglerparameter zur CPU Sie können im Anwenderprogramm einzelne Parameter auf der CPU ändern.
LOAD_PAR	Alle statischen Variablen der Struktur PAR werden von der CPU auf die FM 355-2 geladen. Dies ist notwendig, wenn Sie die Regelparameter auf der CPU abhängig von Prozesszuständen geändert haben..	Keine Übertragung der Reglerparameter zur FM 355-2

Nach erfolgreicher Datenübertragung werden die Parameter LOAD_OP, READ_OUT, LOAD_PAR und READ_PAR wieder auf FALSE gesetzt.

Einsatz in dezentraler Peripherie

Bei Einsatz der FM 355-2 in dezentraler Peripherie können die Datenübertragung und das Rücksetzen der Parameter einige Aufrufzyklen dauern.

Bearbeitungszeiten FMT_PID

Beim programmgesteuerten Umparametrieren (LOAD_PAR, LOAD_OP) der FM 355-2 durch FMT_PID vergrößert sich dessen Laufzeit. Die neuen Parameter sind immer sofort wirksam.

Randbedingungen			Bearbeitungszeit in	
READ_OUT	LOAD_OP	LOAD_PAR READ_PAR	CPU 315-2-DP	CPU 416-2 DP
FALSE	FALSE	FALSE	0,65 ms	0.04 ms
TRUE	FALSE	FALSE	2.85 ms	0.30 ms
TRUE	TRUE	FALSE	4.56 ms	0.55 ms
FALSE	FALSE	TRUE	3.58 ms*)	0.30 ms
TRUE	FALSE	TRUE	5.8 ms*)	0.56 ms
TRUE	TRUE	TRUE	7.41 ms*)	0.82 ms
*) Wenn READ_PAR statt LOAD_PAR gesetzt ist, erhöht sich die Bearbeitungszeit bei der CPU 315-2 DP um 0.35 ms.				

Siehe auch

Blockschaltbilder (Seite 6261)

Eingangsparameter FMT_PID

Tabelle 4-247

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	0	Nummer des Reglerkanals, auf den sich der Instanz-DB bezieht. Gültig sind Werte von 0 bis 3

Siehe auch

Blockschaltbilder (Seite 6261)

Ausgangsparameter FMT_PID

Tabelle 4-248

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QMOD_F	4.0	BOOL	FALSE	Die Anweisung überprüft das erfolgreiche Schreiben und Lesen eines Datensatzes. Bei erkannten Fehlern wird der Ausgang QMOD_F gesetzt. Fehlerursache kann sein: eine falsche Baugruppenadresse am Parameter MOD_ADDR, eine falsche Kanalnummer am Parameter CHANNEL oder eine defekte Baugruppe.
RET_VALU	6.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter FMT_PID

Tabelle 4-249

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	8.0	BOOL	FALSE	Bei COM_RST = TRUE führt die Anweisung FMT_PID einen Initialisierungslauf durch und setzt COM_RST zurück. Dieser Initialisierungslauf ist bei jedem Anlauf der CPU zwingend notwendig.
LOAD_OP	8.1	BOOL	FALSE	Ist der Durchgangsparameter LOAD_OP gesetzt, werden die Bedienparameter in die Baugruppe geladen, die Ausgangsparameter gelesen und der Durchgangsparameter rückgesetzt.
READ_OUT	8.2	BOOL	FALSE	Ist der Durchgangsparameter READ_OUT gesetzt, werden die Ausgangsparameter von der Baugruppe gelesen, in der Struktur OUT des Instanz-DB abgelegt und der Durchgangsparameter rückgesetzt.
LOAD_PAR	8.3	BOOL	FALSE	Ist der Durchgangsparameter LOAD_PAR gesetzt, werden die Reglerparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
READ_PAR	8.4	BOOL	FALSE	Wenn der Parameter READ_PAR = TRUE ist, dann werden die Reglerparameter aus der FM 355-2 gelesen und in der Struktur PAR des Instanz-DB abgelegt. Anschließend wird der Durchgangsparameter zurückgesetzt.

Siehe auch

Blockschaltbilder (Seite 6261)

Statische Variablen FMT_PID.OP

Tabelle 4-250

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
vers_nr	10.0	WORD	W#16#3230	Der Parameter vers_nr darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Bedienparameter, die zur Baugruppe übertragen werden, wenn LOAD_OP = TRUE gesetzt wird.
SP_RE	12.0	REAL	0.0	Ein externer Sollwert wird am Eingang SP_RE mit dem Regler verschaltet.
LMN_RE	16.0	REAL	0.0	Ein externer Stellwert wird am Eingang LMN_RE mit dem Regler verschaltet. Zulässig sind Werte von -100.0...100.0 (%).
SAFE_ON	20.0	BOOL	FALSE	Ist SAFE_ON gesetzt, wird ein Sicherheitswert als Stellwert übernommen.
LMNTRKON	20.1	BOOL	FALSE	Ist LMNTRKON gesetzt, wird der Stellwert einem Analogeingang (AE) nachgeführt. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
LMN_REON	20.2	BOOL	FALSE	Ist LMN_REON gesetzt, wird der externe Stellwert LMN_RE als Stellwert übernommen (Handbetrieb).

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMNRHSRE	20.3	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" kann an einen Digitaleingang der FM 355-2 oder am Eingang LMNRHSRE verschaltet werden. LMNRHSRE = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler).
LMNRLSRE	20.4	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" kann an einen Digitaleingang der FM 355-2 oder am Eingang LMNRLSRE verschaltet werden. LMNRLSRE = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler).
LMNS_ON	20.5	BOOL	FALSE	Stellwertbedienung einschalten (nur beim Schrittregler; Handbetrieb).
LMN_UP	20.6	BOOL	FALSE	Stellwertsignal Hoch Bedienung (nur beim Schrittregler).
LMN_DN	20.7	BOOL	FALSE	Stellwertsignal Tief Bedienung (nur beim Schrittregler).
SAVE_PAR	22.0	BOOL	FALSE	Sichert die aktuellen Reglerparameter in den Ausgangsparametern der Anweisung FMT_TUN. • FMT_TUN.SAV_PFAC=PFAC_SP • FMT_TUN.SAV_GAIN =GAIN • FMT_TUN.SAV_TI =TI • FMT_TUN.SAV_TD = TD • FMT_TUN.SAV_D_F =D_F • FMT_TUN.SAV_CONZ = CON_ZONE • FMT_TUN.SAV_RATI =RATIOFAC • FMT_TUN.SAV_CZON=CONZ_ON • FMT_TUN.SAV_PSEL=P_SEL
UNDO_PAR	22.1	BOOL	FALSE	Die gesicherten Reglerparameter werden wieder zurückgeholt, auch im Automatikbetrieb. • PFAC_SP= FMT_TUN.SAV_PFAC • GAIN = FMT_TUN.SAV_GAIN • TI = FMT_TUN.SAV_TI • TD = FMT_TUN.SAV_TD • D_F = FMT_TUN.SAV_D_F • CON_ZONE = FMT_TUN.SAV_CONZ • RATIOFAC = FMT_TUN.SAV_RATI • CONZ_ON= FMT_TUN.SAV_CZON • P_SEL= FMT_TUN.SAV_PSEL
LOAD_PID	22.2	BOOL	FALSE	Lädt bei PID_ON=TRUE die PID-Parameter (PID_GAIN, PID_TI, PID_TD). Bei PID_ON=FALSE werden die PI-Parameter geladen (PI_GAIN, PI_TI). (auch im Automatikbetrieb)
TUN_ON	22.3	BOOL	FALSE	TUN_ON=TRUE versetzt die FM 355-2 in die Optimierbereitschaft (PHASE=1): Es wird der Stellwert gemittelt bis die Stellwertanregung aufgeschaltet wird.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
TUN_ST	22.4	BOOL	FALSE	Soll bei der Regleroptimierung am Arbeitspunkt der Sollwert konstant bleiben, wird durch TUN_ST=TRUE ein Stellwertsprung um TUN_DLMN aufgeschaltet (PHASE = 1 oder PHASE = 2)
TUN_CST	22.5	BOOL	FALSE	Startet die Kühloptimierung durch Aufschalten eines Stellwertsprungs um TUN_CLMN (PHASE = 1 oder PHASE = 2)

Siehe auch

Beschreibung FMT_PID (Seite 6249)

Blockschaltbilder (Seite 6261)

Statische Variablen FMT_PID.PAR

Tabelle 4-251

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
vers_nr	24.0	WORD	W#16#3230	Der Parameter vers_nr darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Reglerparameter, die mit READ_PAR = TRUE aus der FM gelesen und im Instanz-DB abgelegt werden bzw. mit LOAD_PAR = TRUE zur FM übertragen werden.
P_SEL	26.0	BOOL	TRUE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der Proportionalanteil ist eingeschaltet, wenn der Eingang P_SEL gesetzt ist.
MONERSEL	26.1	BOOL	FALSE	Der Regler besitzt einen Grenzwertmelder, der entweder für den Istwert oder die Regeldifferenz angewendet werden kann. Ist MONERSEL gesetzt, wird die Regeldifferenz überwacht.
PID_ON	26.2	BOOL	TRUE	Nach der Optimierung oder bei LOAD_PID =TRUE werden die PID-Parameter aktiviert. Bei PID_ON =FALSE werden die PI-Parameter genommen. Es kann jedoch sein, dass bei manchen Streckentypen trotz PID_ON = TRUE nur ein PI-Regler entworfen wird.
CONZ_ON	26.3	BOOL	FALSE	TRUE = Regelzone eingeschaltet FALSE = Regelzone ausgeschaltet
D_EL_SEL	28.0	INT	16	Das D-Glied im PID-Algorithmus kann an einen separaten Eingang gelegt werden. Am Parameter D_EL_SEL wird dieser ausgewählt. • D_EL_SEL = 0 bis D_EL_SEL = 3: Analogeingang 0 bis 3 • D_EL_SEL = 16: Regeldifferenz • D_EL_SEL = 17: negativer Istwert
SP_HLM	30.0	REAL	100.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Parameter SP_HLM gibt die obere Begrenzung an. SP_HLM > SP_LLM
SP_LLM	34.0	REAL	0.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Parameter SP_LLM gibt die untere Begrenzung an. SP_LLM < SP_HLM
H_ALM	38.0	REAL	100.0	Oberste Grenze für die Überwachung des Istwertes bzw. der Regeldifferenz. H_ALM > H_WRN

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
H_WRN	42.0	REAL	90.0	Zweite Grenze von oben für die Überwachung des Istwertes bzw. der Regeldifferenz. H_WRN > H_ALM
L_WRN	46.0	REAL	10.0	Zweite Grenze von unten für die Überwachung des Istwertes bzw. der Regeldifferenz. Zulässig sind Werte von H_WRN bis L_ALM.
L_ALM	50.0	REAL	0.0	Unterste Grenze für die Überwachung des Istwertes bzw. der Regeldifferenz. Zulässig sind Werte von L_ALM bis L_WRN.
HYS	54.0	REAL	1.0	Zur Vermeidung von Flackern der Überwachungsanzeigen kann eine Hysterese parametriert werden. HYS >= 0.0
DEADB_W	58.0	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Parameter DEADB_W bestimmt die Größe der Totzone. DEADB_W >= 0.0
PFAC_SP	62.0	REAL	1.0	PFAC_SP gibt den wirksamen P-Anteil bei Sollwertänderung an. Zulässig sind Werte von 0.0 bis 1.0.
GAIN	66.0	REAL	1.0	Der Parameter GAIN gibt die Reglerverstärkung an. Eine Invertierung des Regelsinns wird durch ein negatives Vorzeichen von GAIN erreicht.
TI	70.0	REAL	3000.0	Der Parameter TI (Nachstellzeit) bestimmt das Zeitverhalten des Integrierers. Bei TI = 0.0 ist der Integrierer ausgeschaltet. TI = 0.0 oder TI >= 0.5
TD	74.0	REAL	0.0	Der Parameter TD (Vorhaltezeit) bestimmt das Zeitverhalten des Differenzierers (D-Glied). Bei TD = 0.0 ist der Differenzierer abgeschaltet. TD = 0.0 oder TD >= 1.0
D_F	78.0	REAL	5.0	Die für das D-Glied wirksame Zeitkonstante TD/D_F wird intern auf ≥ Abtastzeit/2 begrenzt. Zulässig sind Werte von 5.0 bis 10.0.
LMN_SAFE	82.0	REAL	0.0	Für den Stellwert kann ein Sicherheitswert parametriert werden. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_HLM	86.0	REAL	100.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Parameter LMN_HLM gibt die obere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung). Zulässig sind Werte von LMN_LLM bis 100.0 (%).
LMN_LLM	90.0	REAL	0.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Parameter LMN_LLM gibt die untere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung). Zulässig sind Werte von -100.0 bis LMN_HLM (%).
MTR_TM	94.0	REAL	60.0	Laufzeit des Stellventils von Anschlag zu Anschlag (nur beim Schrittregler). MTR_TM >= 0.001
PULSE_TM	98.0	REAL	0.0	Minimale Impulslänge (nur beim Schrittregler oder Impulsregler)
BREAK_TM	102.0	REAL	0.0	Minimale Impulspausenlänge (nur beim Schrittregler oder Impulsregler).

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RATIOFAC	106.0	REAL	0.0	Verhältnis von Heizverstärkung/ Kühlverstärkung der Strecke. RATIOFAC <> 0.0 bewirkt: • LMN = LMN * RATIOFAC für LMN<0.0; • CON_ZONE 50% größer Zulässig sind Werte von 0.01 bis 100.0 oder 0.0.
CON_ZONE	110.0	REAL	100.0	Regelzonenbreite Wenn ER >= CON_ZONE, dann LMN = LMN_HLM. Wenn ER <= -CON_ZONE , dann LMN = LMN_LLM. Wenn RATIOFAC <> 0.0 und ER <= -CON_ZONE/RATIOFAC, dann LMN = LMN_LLM. CON_ZONE > 0.0
TUN_DLMN	114.0	REAL	20.0	Die Prozessanregung für die Regleroptimierung erfolgt durch einen Stellwertsprung um TUN_DLMN. Zulässig sind Werte von -100.0 bis 100.0 (%).
TUN_CLMN	118.0	REAL	-20.0	Stellwertsprung nach Start der Kühloptimierung durch TUN_CST. Zulässig sind Werte von -100.0 bis 100.0 (%).

Siehe auch

Blockschaltbilder (Seite 6261)

Statische Variablen FMT_PID.OUT

Tabelle 4-252

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
vers_nr	122.0	WORD	W#16#3230	Der Parameter vers_nr darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Ausgangsparameter, die von der Baugruppe gelesen werden, wenn READ_OUT = TRUE gesetzt wird.
PV	124.0	REAL	0.0	Effektiv wirkender Istwert.
LMN	128.0	REAL	0.0	Am Ausgang LMN wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Ausgang LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100.0 bis 100.0 (%).
QLMNSAFE	132.1	BOOL	FALSE	Ist der Ausgang QLMNSAFE gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNTRK	132.3	BOOL	FALSE	Der Ausgang QLMNTRK zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	132.4	BOOL	FALSE	Der Ausgang QLMN_RE zeigt an, ob LMN_REON gesetzt ist und der externe Stellwert LMN_RE als Stellwert übernommen wird.
QLMNR_HS	132.5	BOOL	FALSE	QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)
QLMNR_LS	132.6	BOOL	FALSE	QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler).

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNR_ON	132.7	BOOL	FALSE	QLMNR_ON = TRUE heißt: Schrittregler mit Stellungsrückmeldung.
QSTEPCON	133.0	BOOL	FALSE	Schrittregler / Impulsregler • QSTEPCON = 0: Impulsregler oder kontinuierlicher Regler • QSTEPCON = 1: Schrittregler
QSPR	133.1	BOOL	FALSE	Ist der Ausgang QSPR gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QMAN_FC	133.3	BOOL	FALSE	Der Regler ist ein Führungsregler, der durch Handbetrieb eines Folgereglers auf dessen Istwert nachgeführt wird oder dessen I-Anteil angehalten wird, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind.
QH_ALM	134.0	BOOL	FALSE	Oberste Grenze H_ALM durch Istwert oder Regeldifferenz überschritten.
QH_WRN	134.1	BOOL	FALSE	Zweite Grenze von oben H_WRN durch Istwert oder Regeldifferenz überschritten.
QL_WRN	134.2	BOOL	FALSE	Zweite Grenze von unten L_WRN durch Istwert oder Regeldifferenz unterschritten.
QL_ALM	134.3	BOOL	FALSE	Unterste Grenze L_ALM durch Istwert oder Regeldifferenz unterschritten.
QLMN_HLM	134.4	BOOL	FALSE	Der Ausgang QLMN_HLM meldet das Erreichen der oberen Stellwert-Begrenzung LMN_HLM. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung).
QLMN_LLM	134.5	BOOL	FALSE	Der Ausgang QLMN_LLM meldet das Erreichen der unteren Stellwert-Begrenzung LMN_LLM. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QPAR_F	134.6	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang QPAR_F angezeigt.
QCH_F	134.7	BOOL	FALSE	Der Ausgang QCH_F wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPAR_F = TRUE oder QMOD_F = TRUE gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	135.0	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang QUPRLM gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	135.1	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang QDNRLM gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	135.2	BOOL	FALSE	Der Ausgang QSP_HLM meldet das Erreichen der oberen Sollwert-Begrenzung SP_HLM.
QSP_LLM	135.3	BOOL	FALSE	Der Ausgang QSP_LLM meldet das Erreichen der unteren Sollwert-Begrenzung SP_LLM.
QLMNUP	135.4	BOOL	FALSE	Ausgang "Stellwertsignal Hoch". (Nur beim Schrittregler oder Impulsregler)
QLMNDN	135.5	BOOL	FALSE	Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QTUN_ON	135.6	BOOL	FALSE	QTUN_ON = TRUE zeigt an, dass eine Optimierung läuft.
PAR_ACT	135.7	BOOL	FALSE	Reglerparameter aktualisieren

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP	136.0	REAL	0.0	Effektiv wirkender Sollwert.
ER	140.0	REAL	0.0	Effektiv wirkende Regeldifferenz (SP- PV)
DISV	144.0	REAL	0.0	Effektiv wirkende Störgröße. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_A	148.0	REAL	0.0	Am Ausgang LMN_A wird beim kontinuierlichen oder Impuls-Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).)
LMN_B	152.0	REAL	0.0	Am Ausgang LMN_B wird beim kontinuierlichen oder Impuls-Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).
PHASE	156.0	INT	0	Zeigt den Ablauf der Regleroptimierung an. • PHASE = -1: Abbruch der Regleroptimierung z. B bei paralleler Optimierung mehrerer Kanäle. Der Wert steht wenige Zyklen an und dient als Abbruchmeldung zwischen der Anweisung FMT_PID und der Baugruppe. • PHASE = 0: kein Optimierbetrieb; Automatik- oder Handbetrieb • PHASE = 1: Optimierungsbereitschaft; Parameter prüfen, Warten auf Anregung, Messen der Abtastzeiten • PHASE = 2: Eigentliche Optimierung: Wendepunktsuche bei konstantem Stellwert. • PHASE = 3: Berechnen der Prozess-Parameter. Abspeichern der vor der Optimierung gültigen Reglerparameter. • PHASE = 4: Reglerentwurf • PHASE = 5: Nachführen des Reglers auf neue Stellgröße • PHASE = 6: Nachführen des Reglers auf neue Stellgröße • PHASE = 7: Überprüfung des Streckentyps, falls Streckentyp II oder III ermittelt wurde (nur bei Heizoptimierung).
STATUS_H	158.0	INT	0	AUTOHOTSPOT zeigt einen Diagnosewert über die Suche des Wendepunkts beim Heizvorgang an.
STATUS_C	160.0	INT	0	AUTOHOTSPOT zeigt einen Diagnosewert über die Suche des Wendepunkts beim Kühlvorgang an.
STATUS_D	162.0	INT	0	AUTOHOTSPOT zeigt einen Diagnosewert über den Reglerentwurf beim Heizvorgang.
ZONE_TUN	164.0	WORD	W#16#0	In der HEX-Darstellung entspricht jede der vier Stellen einem Kanal, in folgender Reihenfolge von links nach rechts: Kanal 0, 1, 2, 3. ZONE_TUN = 0 bedeutet, dass der angewählte Kanal nicht mit anderen Kanälen zusammen optimiert werden wird. Ein Wert <> 0000 zeigt - jeweils mit einer 1 - die Kanäle, die zusammen optimiert werden.

Siehe auch

Blockschaltbilder (Seite 6261)

Blockschaltbilder

Überblick

In den folgenden Abbildungen wird dargestellt, an welchen Stellen die Parameter der Anweisung FMT_PID wirken.

Bei Dreikomponentenreglern und Verhältnis-/Mischungsreglern wirken die Parameter an der gleichen Stelle wie bei Festwert- oder Kaskadenreglern. Dies gilt auch für die Parameter, die gleichermaßen bei kontinuierlichen Reglern, bei Reglern mit Impulsausgang sowie bei Schrittreglern vorhanden sind. Generell gilt, dass gleiche Schaltflächen auch gleiche Parameter beinhalten. Aus Gründen der Übersichtlichkeit werden daher nicht alle Strukturbilder dargestellt und nicht in allen Bildern alle Parameter eingezeichnet.

Die Parameter der Anweisung FMT_PID sind jedoch in den Bildern alle enthalten - bis auf die Parameter MOD_ADDR, CHANNEL, QMOD_F, QPAR_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_PAR, READ_OUT und LOAD_OP.

In den folgenden Bildern wird dargestellt, an welchen Stellen in der Baugruppe die Parameter des FMT_PID wirken.

Wirkung der Eingangsparameter

Das folgende Bild zeigt die Regeldifferenzbildung beim Festwert- oder Kaskadenregler

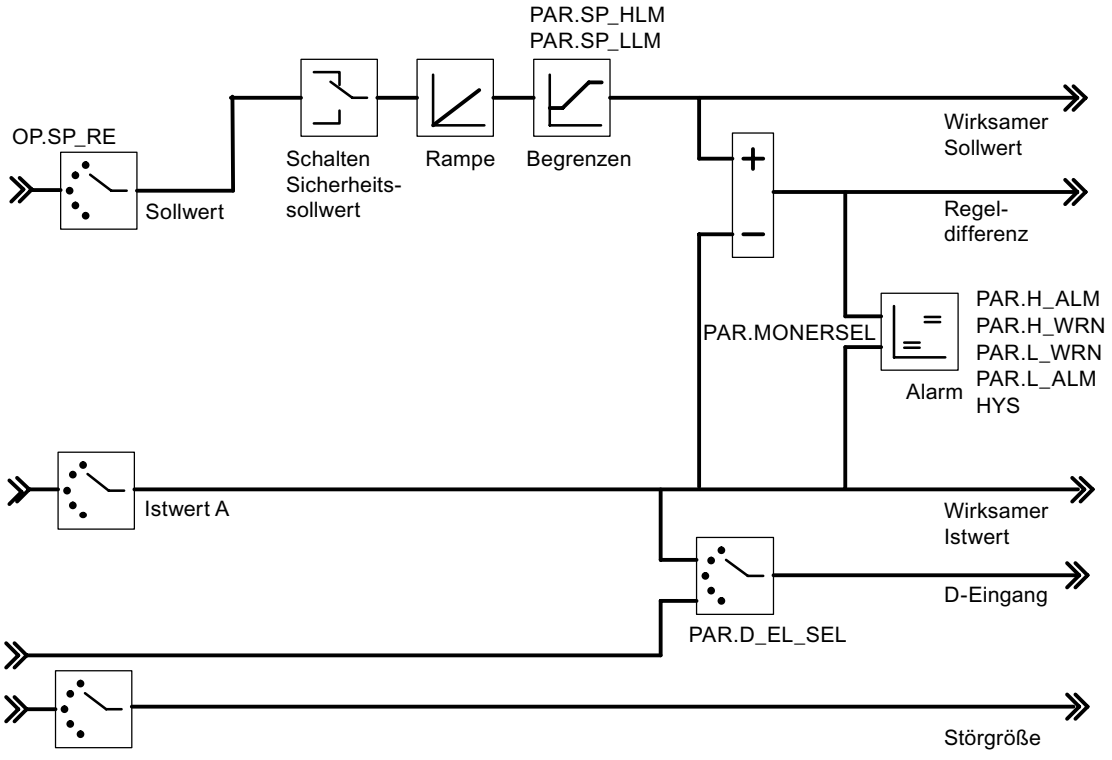

Das folgende Bild zeigt das Blockschaltbild des Regelalgorithmus

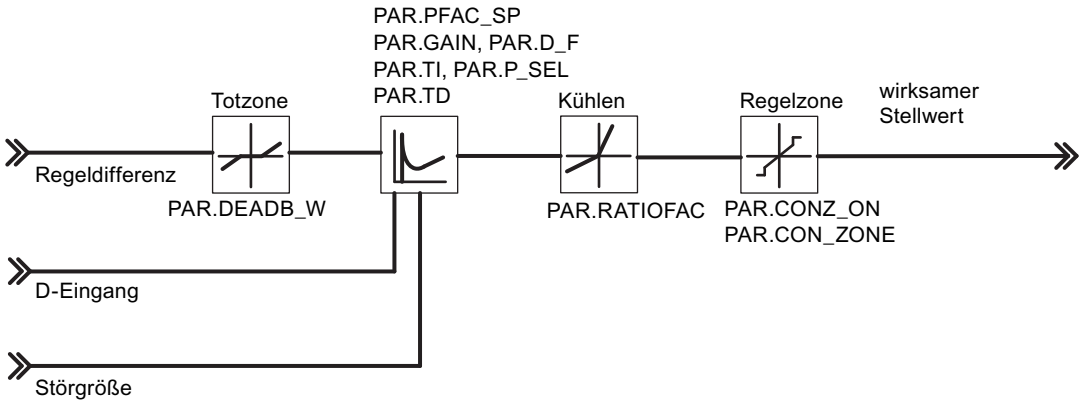

Das folgende Bild zeigt den Reglerausgang des kontinuierlichen Reglers(FM 355-2 C)

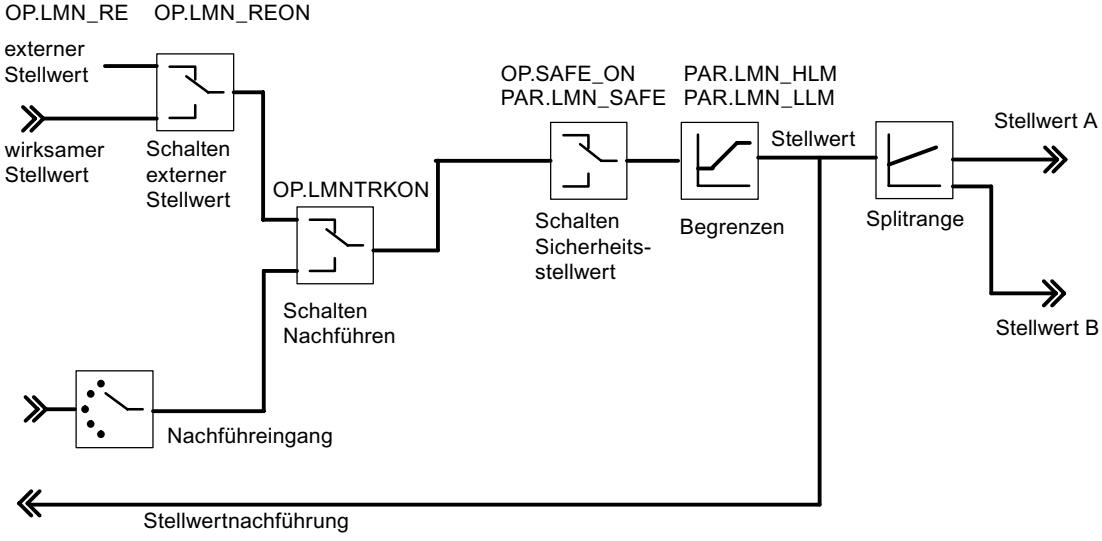

Das folgende Bild zeigt den Reglerausgang des Impulsreglers (FM 355-2 S)

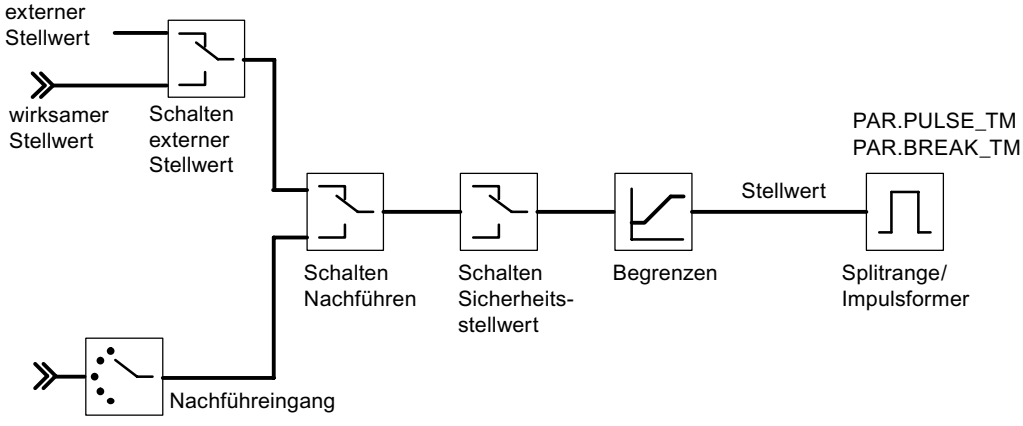

Das folgende Bild zeigt den Reglerausgang des Schrittreglers mit Stellungsrückmeldung (FM 355-2 S)

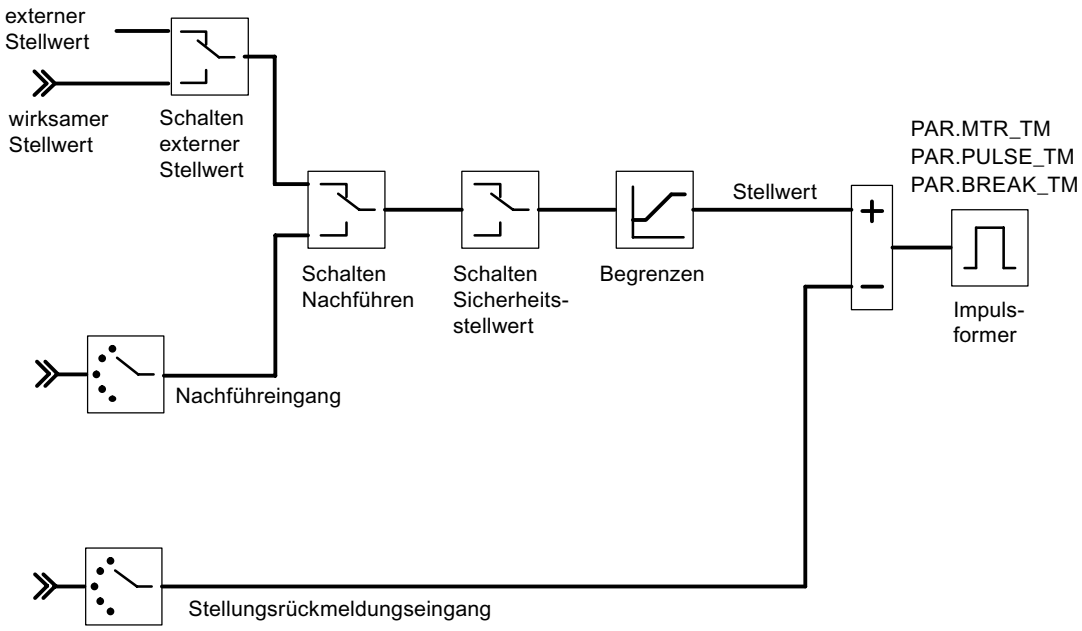

Das folgende Bild zeigt den Reglerausgang des Schrittreglers ohne Stellungsrückmeldung (FM 355-2 S)

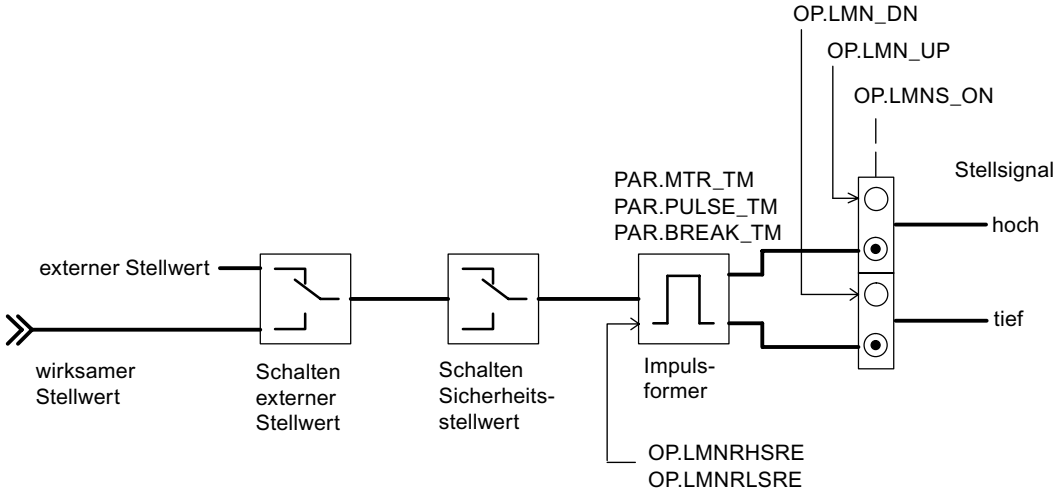

Erzeugung der Ausgangsparameter

In den folgenden Bildern wird dargestellt, an welchen Stellen in der Baugruppe die Ausgangsparameter der Anweisung FMT_PID erzeugt werden.

Das folgende Bild zeigt die Regeldifferenzbildung beim Festwert- oder Kaskadenregler

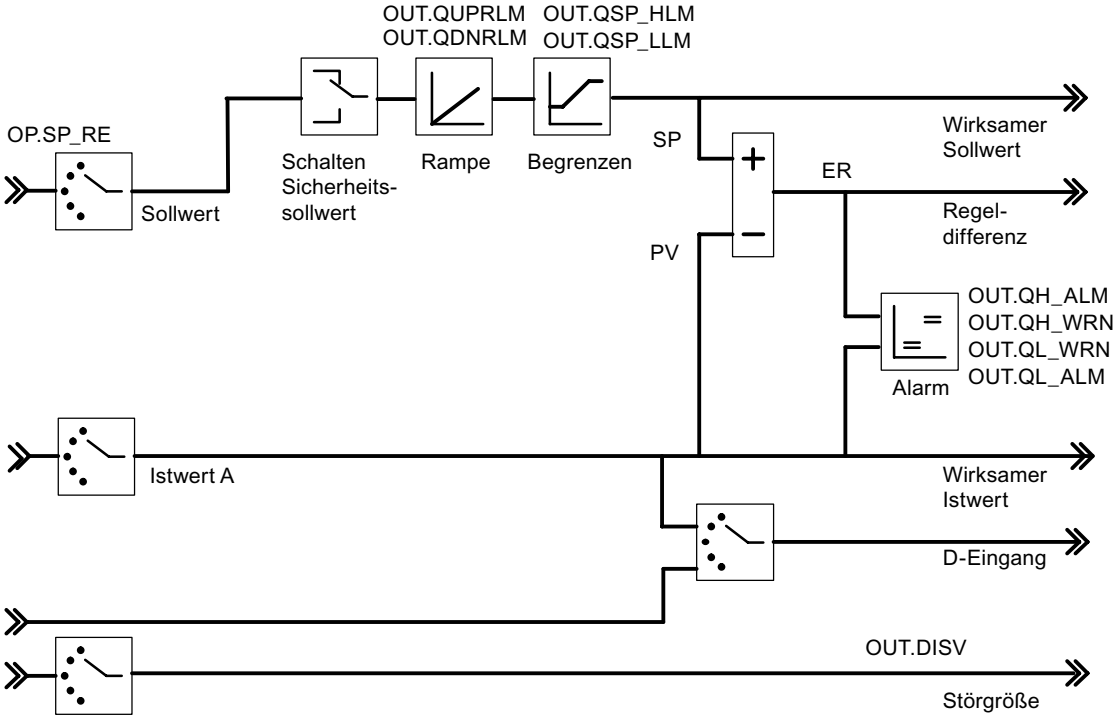

Das folgende Bild zeigt den Reglerausgang des kontinuierlichen Reglers (FM 355-2 C)

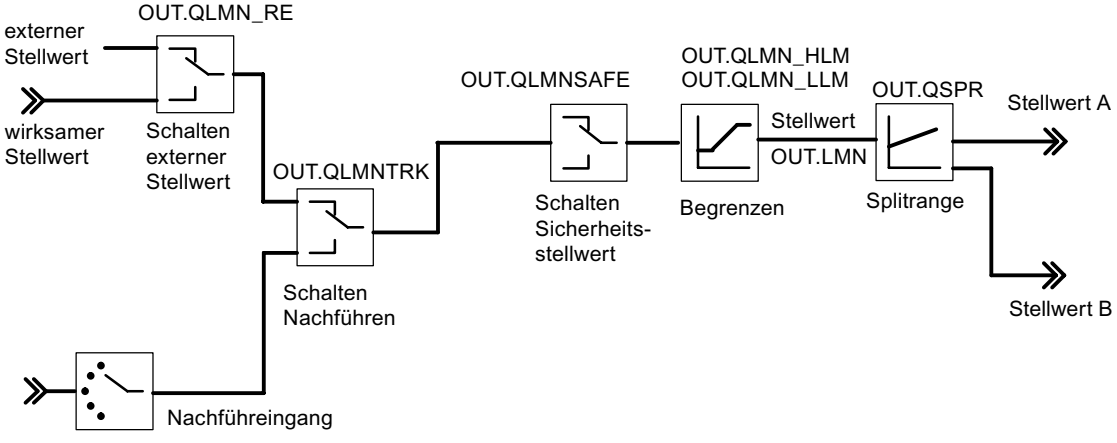

Das folgende Bild zeigt den Reglerausgang des Impulsreglers (FM 355-2 S)

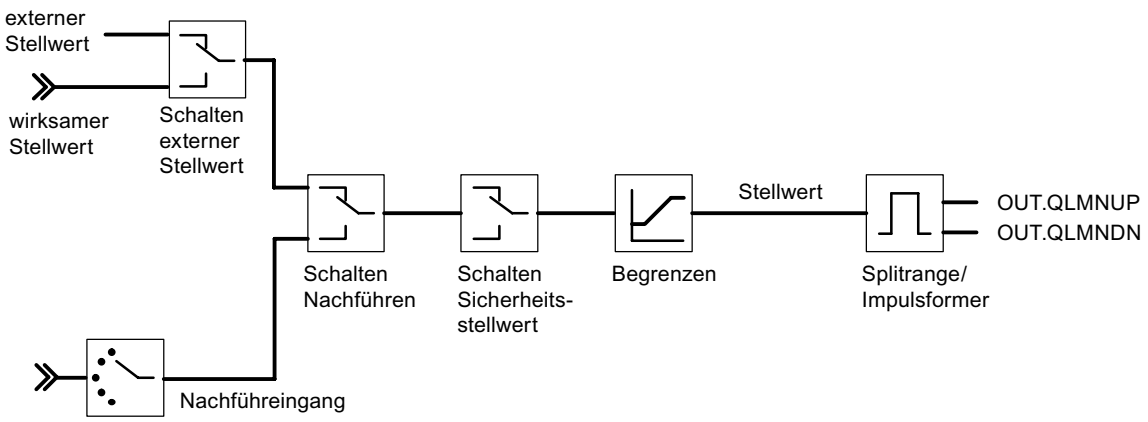

Das folgende Bild zeigt den Reglerausgang des Schrittreglers mit Stellungsrückmeldung (FM 355-2 S)

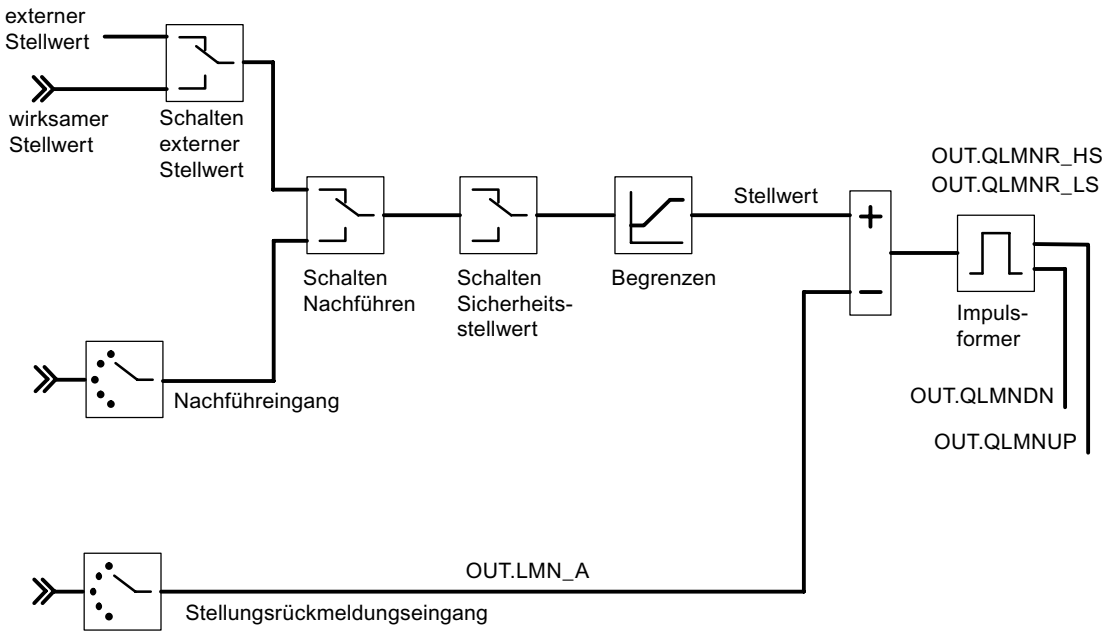

Das folgende Bild zeigt den Reglerausgang des Schrittreglers ohne Stellungsrückmeldung (FM 355-2 S)

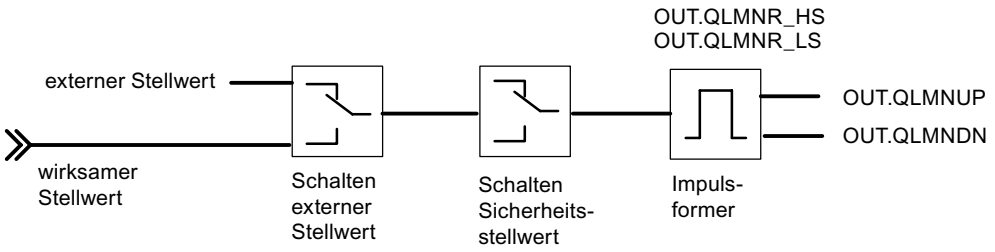

Siehe auch

Beschreibung FMT_PID (Seite 6249)

Parameter STATUS_H

STATUS_H zeigt einen Diagnosewert über die Suche des Wendepunktes beim Heizen.

Wenn ein Diagnosewert 2xxxx angezeigt wird, wird die Optimierung beendet, aber die Reglerparameter sind unsicher.

Wenn ein Diagnosewert 3xxxx angezeigt wird, wird die Optimierung abgebrochen (PHASE = 0 und TUN_ON = FALSE) und der Regler wechselt in die vorige Betriebsart..

STATUS_ H	Beschreibung	Abhilfe
0	Default bzw. keine oder noch keine neuen Reglerparameter	
10000	Optimierung beendet + geeignete Reglerparameter gefunden	
2xxxx	Optimierung beendet + Reglerparameter unsicher	
2xxx1	effektive Motorstellzeit >= 65% von Wendepunktzeit T_P_INF	Kompensation von TU und T_P_INF wird nicht mehr durchgeführt. Dadurch wird ein sanfterer Regler entworfen.
2xx2x	Wendepunkt nicht erreicht (nur bei Anregung über Sollwertsprung)	Falls Regler schwingt, Reglerparameter abschwächen oder Versuch wiederholen mit kleinerem TUN_DLMN.
2x1xx	Schätzfehler TU <= 3*Abtastzeit	Versuch wiederholen.
2x2xx	Schätzfehler schwer (TU < Abtastzeit)	Versuch wiederholen. Sonderfall reine PT1–Strecke: Versuch nicht wiederholen; evtl. Reglerparameter abschwächen.
2x3xx	Schätzfehler TU zu groß	Versuch wiederholen mit besseren Bedingungen.
21xxx	Schätzfehler N_PTN < 1	Versuch wiederholen mit besseren Bedingungen.
22xxx	Schätzfehler N_PTN > 10	Versuch wiederholen mit besseren Bedingungen.
3xxxx	Optimierung abgebrochen in Phase 1 durch fehlerhafte Parametrierung	
30001	Gleichzeitiges Setzen von TUN_ON und TUN_ST	Optimierung erneut starten.

STATUS_ H	Beschreibung	Abhilfe
30002	|TUN_DLMN| < 5 % oder Betrag der effektiven LMN-Änderung < 5 % • Übergang in Phase 99 (Phase 0 bei Paralleloptimierung) • Ausgabe LMN = LMN0	Entweder haben Sie für TUN_DLMN einen zu kleine Wert parametriert oder die Stellgröße war vor Optimierungsstart nahe an einer Stellgrenze. Dann muss verhindert werden, dass der Regler auf den neuen Sollwert einregelt und unnötig den stationären Arbeitspunkt verlässt (nicht möglich bei Paralleloptimierung). Sie sollten nun so vorgehen: • Evtl. Sollwert zurücknehmen und TUN_ON auf FALSE setzen • TUN_DLMN korrigieren • Stellwertgrenzen überprüfen, falls TUN_DLMN >= 5%. • Optimierung erneut starten.
30004	|TUN_DLMN| < 5 % oder effektive LMN-Änderung < 5 % Effektive Stellwertdifferenz begrenzt durch Splitrange-Grenzen und nicht durch Stellwertgrenzen. • Übergang in Phase 99 (Phase 0 bei Paralleloptimierung) • Ausgabe LMN = LMN0	Siehe STATUS_H=30002: Bedenken Sie außerdem, dass z. B. bei LMN_A < 5 % eine Heizoptimierung mit negativem TUN_DLMN nicht möglich ist (Grund: die Kühlleistung darf nicht verstellt werden).
30009	Sicherheitsbetrieb	Sicherheitsbetrieb beenden und Optimierung erneut starten.

Hinweis

Am Ende der Phase 1 wird STATUS_H = 0 rückgesetzt.

Wenn Sie die Optimierung in Phase 2 abbrechen, ist STATUS_H = 0. STATUS_D zeigt jedoch immer noch den Zustand der letzten Reglerberechnung an.

Siehe auch

Blockschaltbilder (Seite 6261)

Parameter STATUS_C

STATUS_C zeigt einen Diagnosewert über die Suche des Wendepunktes beim Kühlen.

Wenn ein Diagnosewert 2xxxx angezeigt wird, wird die Optimierung beendet, aber die Reglerparameter sind unsicher.

Wenn ein Diagnosewert 3xxxx angezeigt wird, wird die Optimierung abgebrochen (PHASE = 0 und TUN_ON = FALSE) und der Regler wechselt in die vorige Betriebsart.

STATUS_ C	Beschreibung	Abhilfe
0	Default bzw. keine oder noch keine neuen Reglerparameter	
10000	Optimierung beendet + geeignete Reglerparameter gefunden	
2xxxx	Optimierung beendet + Reglerparameter unsicher	
2x3xx	Schätzfehler TU zu groß	Versuch wiederholen mit besseren Bedingungen.
21xxx	Schätzfehler N_PTN < 1	Versuch wiederholen mit besseren Bedingungen.
22xxx	Schätzfehler N_PTN > 10	Versuch wiederholen mit besseren Bedingungen.
3xxxx	Optimierung abgebrochen in Phase 1 durch fehlerhafte Parametrierung	
30001	Gleichzeitiges Setzen von TUN_ON und TUN_CST	Optimierung erneut starten.
30002	Nur bei Schrittregler ohne analoge Stellungsrückmeldung: |TUN_CLMN| < 5 % oder Betrag der effektiven LMN-Änderung < 5 % • Übergang in Phase 0 • TUN_ON = FALSE	Entweder haben Sie für TUN_CLMN einen zu kleine Wert parametriert oder die Stellgröße war vor Optimierungsstart nahe an einer Stellgrenze. Dann muss verhindert werden, dass der Regler auf den neuen Sollwert einregelt und unnötig den stationären Arbeitspunkt verlässt (nicht möglich bei Paralleloptimierung). Sie sollten nun so vorgehen: • Evtl. Sollwert zurücknehmen und TUN_ON auf FALSE setzen. • TUN_CLMN korrigieren • Stellwertgrenzen überprüfen, falls |TUN_CLMN| >= 5% und das Vorzeichen richtig war. • Optimierung erneut starten.
30003	TUN_CLMN <= -5%, aber LMN_LLM > -5%. • Übergang in Phase 0 • TUN_ON = FALSE	Die untere Grenze LMN_LLM ist zu hoch (z.B. 0%) und verhindert daher die Ausgabe einer Kühlleistung. • Untere Grenze LMN_LLM korrigieren. • Optimierung erneut starten.
30004	TUN_CLMN > -5 % oder effektive LMN-Änderung > -5 % Effektive Stellwertdifferenz begrenzt durch Splitrange-Grenzen und nicht durch Stellwertgrenzen. • Übergang in Phase 0 • TUN_ON = FALSE	Siehe STATUS_C=30002: Bedenken Sie außerdem, dass z. B. bei LMN_A < 5 % eine Heizoptimierung mit negativem TUN_DLMN nicht möglich ist (Grund: die Kühlleistung darf nicht verstellt werden).
30008	TUN_CST aber keine Heizoptimierung zuvor • Übergang in Phase 0 • TUN_ON = FALSE	Zunächst Heizoptimierung durchführen
30009	Sicherheitsbetrieb	Sicherheitsbetrieb beenden und Optimierung erneut starten.

Siehe auch

Blockschaltbilder (Seite 6261)

Parameter STATUS_D

STATUS_D	Beschreibung
0	Es wurden keine Reglerparameter berechnet.
110	N_PTN <= 1.5 Streckentyp I schnell
121	N_PTN > 1.5 Streckentyp I
122	N_PTN = 1.9 Streckentyp I nach Phase 7 (zuvor N_PTN > 1.9)
200	N_PTN > 1.9 Streckentyp II (Übergangsbereich)
310	N_PTN >= 2.1 Streckentyp III schnell
320	N_PTN > 2.6 Streckentyp III

Hinweis

Wenn Sie die Optimierung in Phase 2 abbrechen, ist STATUS_H = 0. STATUS_D zeigt jedoch immer noch den Zustand der letzten Reglerberechnung an.

Je größer der Wert von STATUS_D, desto höher die Ordnung der Regelstrecke, desto größer das Verhältnis TU/TA und desto sanfter die Reglerparameter.

Siehe auch

Blockschaltbilder (Seite 6261)

FMT_PAR

Beschreibung FMT_PAR

Die Anweisung FMT_PAR dient zum Online-Ändern von weiteren Parametern, die über die Anweisung FMT_PID nicht vorgegeben werden können.

Aufruf

FMT_PAR muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

Um Laufzeit zu sparen, soll FMT_PAR nur dann mit LOAD=TRUE aufgerufen werden, wenn Parameter geändert werden sollen.

Arbeitsweise

Mit FMT_PAR können Sie pro Aufruf je einen REAL-Parameter und einen der INT-Parameter ändern.

Am AUTOHOTSPOT geben Sie die Indexnummer des Parameters an, den Sie ändern wollen. Am Eingangsparameter VALUE_R bzw. VALUE_I geben Sie die neuen Werte an.

Wenn Sie mehrere Parameter ändern wollen, müssen Sie denselben Instanz-DB mehrfach nacheinander mit LOAD_PAR = TRUE und mit unterschiedlichen Indexnummern aufrufen.

Bei Einsatz der FM 355-2 in dezentraler Peripherie kann es einige Aufrufzyklen dauern, bis die Parameter zur FM 355-2 übertragen worden sind. Solange die Übertragung noch nicht abgeschlossen ist, behält der Parameter LOAD_PAR den Wert TRUE. Rufen Sie daher bei der Änderung von Parametern die Anweisung FMT_PAR solange wiederholt auf, bis LOAD_PAR = FALSE ist.

Hinweis

Bitte beachten Sie, dass Parameter, die Sie mit Hilfe des FMT_PAR geändert haben, beim Anlauf der CPU mit den Parametern aus den Systemdaten überschrieben werden.

Beispiel

Sie wollen im Betrieb die Hochlaufzeit der Rampe für die Führungsgröße ändern sowie abhängig vom Prozesszustand als Istwert verschiedene Analogeingangswerte verwenden.

- Zum Parametrieren der Hochlaufzeit der Rampe für die Führungsgröße auf 10.0 rufen Sie im laufenden Betrieb FMT_PAR auf mit INDEX = 30 und VALUE_R = 10.0.
- Wenn Sie als Istwert den Analogeingangswert 4 der Baugruppe parametrieren wollen, rufen Sie im laufenden Betrieb den FMT_PAR auf mit INDEX = 50 und VALUE_I = 4.

Eingangsparameter FMT_PAR

Tabelle 4-253

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	0	Nummer des Reglerkanals, auf den sich der Instanz-DB bezieht. Gültig sind Werte von 0 bis 3.
INDEX	4.0	INT	0	0 bis 100
VALUE_R	6.0	REAL	0.0	abhängig vom jeweiligen Parameter
VALUE_I	10.0	INT	0	abhängig vom jeweiligen Parameter

Ausgangsparameter FMT_PAR

Tabelle 4-254

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	12.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisung WRREC.

Siehe auch

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter FMT_PAR

Tabelle 4-255

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LOAD_PAR	+14.0	BOOL	FALSE	Ist der Durchgangsparameter LOAD_PAR gesetzt, wird VALUE_R oder VALUE_I auf die Baugruppe geschrieben. INDEX gibt an, welcher Parameter überschrieben wird. Anschließend wird LOAD_PAR rückgesetzt.

Parameter INDEX

INDEX	Beschreibung	Datentyp
0	kein Parameter angewählt	-
1	Filterzeitkonstante für Analogeingang	REAL
2	Normierung Analogeingang Messbereichsende entspricht 100%	REAL
3	Normierung Analogeingang Messbereichsanfang entspricht 0%	REAL
4	Polygonzug, Stützwert 1 Eingangsseite	REAL
5	Polygonzug, Stützwert 2 Eingangsseite	REAL
6	Polygonzug, Stützwert 3 Eingangsseite	REAL
7	Polygonzug, Stützwert 4 Eingangsseite	REAL
8	Polygonzug, Stützwert 5 Eingangsseite	REAL
9	Polygonzug, Stützwert 6 Eingangsseite	REAL
10	Polygonzug, Stützwert 7 Eingangsseite	REAL
11	Polygonzug, Stützwert 8 Eingangsseite	REAL
12	Polygonzug, Stützwert 9 Eingangsseite	REAL
13	Polygonzug, Stützwert 10 Eingangsseite	REAL
14	Polygonzug, Stützwert 11 Eingangsseite	REAL
15	Polygonzug, Stützwert 12 Eingangsseite	REAL
16	Polygonzug, Stützwert 13 Eingangsseite	REAL
17	Polygonzug, Stützwert 1 Ausgangsseite	REAL
18	Polygonzug, Stützwert 2 Ausgangsseite	REAL
19	Polygonzug, Stützwert 3 Ausgangsseite	REAL
20	Polygonzug, Stützwert 4 Ausgangsseite	REAL
21	Polygonzug, Stützwert 5 Ausgangsseite	REAL
22	Polygonzug, Stützwert 6 Ausgangsseite	REAL
23	Polygonzug, Stützwert 7 Ausgangsseite	REAL
24	Polygonzug, Stützwert 8 Ausgangsseite	REAL
25	Polygonzug, Stützwert 9 Ausgangsseite	REAL
26	Polygonzug, Stützwert 10 Ausgangsseite	REAL
27	Polygonzug, Stützwert 11 Ausgangsseite	REAL

INDEX	Beschreibung	Datentyp
28	Polygonzug, Stützwert 12 Ausgangsseite	REAL
29	Polygonzug, Stützwert 13 Ausgangsseite	REAL
30	Hochlaufzeit der Rampe für Führungsgröße	REAL
31	Sicherheitssollwert	REAL
32	Offset für Sollwertverknüpfung (Verhältnis-/Mischungsregler)	REAL
33	Faktor für Istwert B (Dreikomponentenregler)	REAL
34	Faktor für Istwert C (Dreikomponentenregler)	REAL
35	Offset für Istwertverknüpfung (Dreikomponentenregler)	REAL
36	Faktor für Störgrößenverknüpfung	REAL
37	Arbeitspunkt	REAL
38	unverwendet	REAL
39	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal A	REAL
40	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal A	REAL
41	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal A	REAL
42	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal A	REAL
43	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal B	REAL
44	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal B	REAL
45	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal B	REAL
46	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal B	REAL
47	Mindestimpulsdauer	REAL
48	Mindestpausendauer	REAL
49	Auswahl der Führungsgröße SP bzw. SP_RE für den Regler • -1: Sollwert SP_RE der Anweisung • 0 bis 3: Analogeingangswert 0 bis 3 • 16 bis 19: Stellgröße (LMN) von Regler 0 bis 3 • 32 bis 35: Stellgröße A von Regler 0 bis 3 • 48 bis 51: Stellgröße B von Regler 0 bis 3	INT
50	Auswahl der Haupt-Regelgröße Istwert A für den Regler • -1: Istwert A = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
51	Auswahl der Hilfs-Regelgröße Istwert B für den Regler • -1: Istwert B = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
52	Auswahl der Hilfs-Regelgröße Istwert C für den Regler • -1: Istwert C = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
53	Auswahl der Hilfs-Regelgröße Istwert D für den Regler • -1: Istwert D = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3 • 16 bis 19: Stellgröße (LMN) von Regler 0 bis 3	INT

INDEX	Beschreibung	Datentyp
54	Auswahl der Störgröße DISV für den Regler • -1: Störgröße = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
55	Auswahl am Schalter Nachführeingang. Am Messpunkt MP TRACKPER kann der Wert per Tooltip abgelesen werden. • -1: Stellungsnachführung = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
56	Auswahl am Schalter Stellungsrückmeldungseingang bei Schrittregler mit Stellungsrückmeldung. • -1: Stellungsnachführung = 0.0 • 0 bis 3: Analogeingangswert 0 bis 3	INT
57	Auswahl des Signals zur Umschaltung auf Sicherheitswert für den Stellwert des Regler • -1: nur Vorgabe über Parameter SAFE_ON der Anweisung FMT_PID • 0 bis 7: Vorgabe über Parameter SAFE_ON der Anweisung FMT_PID verodert mit Digitaleingang 0 bis 7	INT
58	Auswahl des Signals zur Umschaltung auf Nachführfunktion des Stellwertes des Reglers • -1: nur Vorgabe über Parameter LMNTRKON der Anweisung FMT_PID • 0 bis 7: Vorgabe über Parameter LMNTRKON der Anweisung FMT_PID verodert mit Digitaleingang 0 bis 7	INT
59	Auswahl des Signals zur Umschaltung des Stellwertes des Reglers auf LMN_RE • -1: nur Vorgabe über Parameter LMN_REON der Anweisung FMT_PID • 0 bis 7: Vorgabe über Parameter LMN_REON der Anweisung FMT_PID verodert mit Digitaleingang 0 bis 7	INT
60	Auswahl des oberen Anschlagsignals der Stellungsrückmeldung • -1: nur Vorgabe über Parameter LMNRHSRE der Anweisung FMT_PID • 0 bis 7: Vorgabe über Parameter LMNRHSRE der Anweisung FMT_PID verodert mit Digitaleingang 0 bis 7	INT
61	Auswahl des unteren Anschlagsignals der Stellungsrückmeldung • -1: nur Vorgabe über Parameter LMNRLSRE der Anweisung FMT_PID • 0 bis 7: Vorgabe über Parameter LMNRLSRE der Anweisung FMT_PID verodert mit Digitaleingang 0 bis 7	INT

FMT_DS1

Beschreibung FMT_DS1

Für das Auslesen des Diagnosedatensatzes DS1 steht Ihnen die Anweisung FMT_DS1 zur Verfügung.

Arbeitsweise

Wenn der Parameter READ_DS1 = TRUE ist, dann wird der Diagnosedatensatz DS1 aus der FM 355-2 gelesen und in der Struktur DS1 des Instanz-DB abgelegt. Anschließend wird der Durchgangsparameter zurückgesetzt.

Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Aufruf

FMT_DS1 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

FMT_DS1 benötigt keinen Initialisierungslauf.

Um immer die aktuellen Diagnosewerte zu erhalten, setzen Sie im OB 82 den Parameter READ_DS1 zyklisch auf TRUE.

Nach erfolgreichem Lesen der Diagnosewerte setzt FMT_DS1 den Parameter READ_DS1 zurück.

Eingangsparameter FMT_DS1

Tabelle 4-256

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter FMT_DS1

Tabelle 4-257

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
DS1	2.0	STRUCT		Der Diagnosedatensatz DS1 besteht aus Bytes 0 bis 12. Die ersten 4 Bytes sind identisch mit dem Diagnosedatensatz DS0. AUTOHOTSPOT
RET_VALU	16.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisung RDREC.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

Durchgangsparameter FMT_DS1

Tabelle 4-258

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
READ_DS1	18.0	BOOL	FALSE	Wenn der Parameter READ_DS1 = TRUE ist, dann wird der Diagnosedatensatz DS1 aus der FM 355-2 gelesen und in der Struktur DS1 des Instanz-DB abgelegt. Anschließend wird der Durchgangsparameter zurückgesetzt.

Belegung des DS1

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS1. Die ersten 4 Bytes sind identisch mit dem Diagnosedatensatz DS0. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Zeitüberwachung angesprochen • EPROM-Fehler • ADU/DAU-Fehler • Analogeingang Hardwarefehler	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Externe Hilfsspannung fehlt • Parametrierung fehlerhaft • Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA) • Analogeingang Messbereichsunterschreitung • Analogeingang Messbereichsüberschreitung • Analogausgang Bürdenbruch • Analogausgang Kurzschluss	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe ab Byte 7	8:x:03
	4	Externe Hilfsspannung fehlt	24-V-Versorgung der FM 355-2 ausgefallen	8:x:04
	6	unbenutzt	-	-
	7	Parametrierung fehlerhaft	Die Baugruppe kann einen Parameter nicht verwerten. Grund: Parameter unbekannt oder unzulässige Kombination von Parametern.	8:x:07
1	0 ... 3	Baugruppenklasse	Ist immer mit 8 belegt.	-
	4	Kanalspezifische Diagnose	Wird gesetzt, wenn die Baugruppe zusätzliche Kanalinformation liefern kann und ein Kanalfehler vorhanden ist (siehe Byte 7 bis 12)	-

Byte	Bit	Bedeutung	Anmerkung		Ereignisnr.
2	3	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler		8:x:33
3	2	EPROM-Fehler	Baugruppe defekt		8:x:42
	4	ADU/DAU-Fehler	Baugruppe defekt		8:x:44
4	0 ... 7	Kanaltyp	Ist immer mit 75H belegt.		-
5	0 ... 7	Länge der Diagnoseinformation	Ist immer mit 8 belegt.		-
6	0 ... 7	Kanalanzahl	Ist immer mit 5 belegt (4 Regler + 1 Referenzkanal)		-
7	0 ... 4	Kanalfehlervektor	Jedem Kanal ist ein Bit zugeordnet (0...3; 4 für Referenzkanal).		-
8	0	Analogeingang Hardwarefehler	Kanalspezifische Diagnose Kanal 0		8:x:B0
	1	unbenutzt			8:x:B1
	2	Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA)			8:x:B2
	3	unbenutzt			8:x:B3
	4	Analogeingang Messbereichsunterschreitung			8:x:B4
	5	Analogeingang Messbereichsüberschreitung			8:x:B5
	6	Analogausgang Drahtbruch		Nur bei Stromausgang des K-Reglers	8:x:B6
	7	Analogausgang Kurzschluss		Nur bei Spannungsausgang des K-Reglers	8:x:B7
9	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 1		siehe oben
10	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 2		siehe oben
11	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 3		siehe oben
12	0 ... 5	siehe Byte 8	Diagnose für Referenzkanal		siehe oben

FMT_TUN

Beschreibung FMT_TUN

Mit der Anweisung FMT_TUN erhalten Sie zusätzliche Detail-Informationen (z. B. gesicherte Reglerparameter) während der Regleroptimierung.

Hinweis

Um eine Regleroptimierung durchzuführen ist jedoch die Anweisung FMT_PID ausreichend. Mit dieser Anweisung können Sie die Optimierung starten und die Status-Informationen beobachten.

Aufruf

FMT_TUN muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

FMT_TUN benötigt keinen Initialisierungslauf.

Setzen Sie den Parameter READ_OUT zyklisch auf TRUE, um immer die aktuellen Werte zu erhalten.

Nach erfolgreichem Lesen der Parameter setzt FMT_TUN den Parameter READ_OUT zurück.

Arbeitsweise

Nach dem Laden der Reglerparameter mit LOAD_PAR = TRUE oder der Bedienparameter mit LOAD_OP = TRUE sind die Parameter nicht sofort wirksam. Erst nach der Zykluszeit der FM355-2 (je nach Anzahl der Kanäle bis zu 500ms) stehen die Reglerparameter zum Rücklesen mit READ_PAR oder die Ausgangsparameter mit READ_OUT zur Verfügung. Dies gilt auch nach dem Laden der Systemdaten (CPU-Übertragung von STOP nach RUN) so wie beim Lesen der Ausgangsparameter über FMT_TUN.

Aufruf bei dezentralem Aufbau

Bei dezentralem Aufbau und gleichzeitigem Aufruf von FMT_PID und FMT_TUN ist folgendes zu beachten:

- LOAD_OP darf nicht gleichzeitig bei FMT_PID und FMT_TUN gesetzt sein.
- READ_OUT darf nicht gleichzeitig bei FMT_PID und FMT_TUN gesetzt sein.

Begründung: Beide Anweisungen greifen über dieselben Datensätze auf die FM355-2 zu. Daher muss sichergestellt werden, dass zur gleichen Zeit nur einer der beiden FBs einen Datensatz liest bzw. schreibt.

Eingangsparameter FMT_TUN

Tabelle 4-259

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	0	Nummer des Reglerkanals, auf den sich der Instanz-DB bezieht. Gültig sind Werte von 0 bis 3.

Ausgangsparameter FMT_TUN

Tabelle 4-260

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PI_GAIN	4.0	REAL	0.0	PI-Reglerparameter
PI_TI	8.0	REAL	0.0	PI-Reglerparameter
PID_GAIN	12.0	REAL	0.0	PID-Reglerparameter
PID_TI	16.0	REAL	0.0	PID-Reglerparameter
PID_TD	20.0	REAL	0.0	PID-Reglerparameter
SAV_PFAC	24.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.PFAC_SP
SAV_GAIN	28.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.GAIN
SAV_TI	32.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.TI
SAV_TD	36.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.TD
SAV_D_F	40.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.D_T
SAV_RATI	44.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.RATIOFAC
SAV_CONZ	48.0	REAL	0.0	Gespeicherter Reglerparameter FMT_PID.PAR.CON_ZONE
SAV_PSEL	52.0	BOOL	FALSE	Gespeicherter Reglerparameter FMT_PID.PAR.P_SEL
SAV_CZON	52.1	BOOL	FALSE	Gespeicherter Reglerparameter FMT_PID.PAR.CONZ_ON
RET_VALU	54.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisung RDREC.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

Durchgangsparameter FMT_TUN

Tabelle 4-261

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
READ_OUT	56.0	BOOL	FALSE	Ist der Durchgangsparameter READ_OUT gesetzt, werden die Ausgangsparameter von der FM 355-2 gelesen.

Statische Variablen FMT_TUN.OUT

Tabelle 4-262

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
vers_nr	58.0	WORD	W#16#3230	Der Parameter vers_nr darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Ausgangsparameter, die von der Baugruppe gelesen werden, wenn READ_OUT = TRUE gesetzt wird.
PV	60.0	REAL	0.0	Effektiv wirkender Istwert.
LMN	64.0	REAL	0.0	Am Ausgang LMN wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Ausgang LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100.0 bis 100.0 (%).
QLMNSAFE	68.1	BOOL	FALSE	Ist der Ausgang QLMNSAFE gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNTRK	68.3	BOOL	FALSE	Der Ausgang QLMNTRK zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	68.4	BOOL	FALSE	Der Ausgang QLMN_RE zeigt an, ob LMN_REON gesetzt ist und der externe Stellwert LMN_RE als Stellwert übernommen wird.
QLMNR_HS	68.5	BOOL	FALSE	QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)
QLMNR_LS	68.6	BOOL	FALSE	QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler).
QLMNR_ON	68.7	BOOL	FALSE	QLMNR_ON = TRUE heißt: Schrittregler mit Stellungsrückmeldung.
QSTEPCON	69.0	BOOL	FALSE	Schrittregler / Impulsregler 0 = Impulsregler oder kontinuierlicher Regler 1 = Schrittregler
QSPR	69.1	BOOL	FALSE	Ist der Ausgang QSPR gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QMAN_FC	69.3	BOOL	FALSE	Der Regler ist ein Führungsregler, der durch Handbetrieb eines Folgereglers auf dessen Istwert nachgeführt wird oder dessen I-Anteil angehalten wird, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind.
QH_ALM	70.0	BOOL	FALSE	Oberste Grenze H_ALM durch Istwert oder Regeldifferenz überschritten.
QH_WRN	70.1	BOOL	FALSE	Zweite Grenze von oben H_WRN durch Istwert oder Regeldifferenz überschritten.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QL_WRN	70.2	BOOL	FALSE	Zweite Grenze von unten L_WRN durch Istwert oder Regeldifferenz unterschritten.
QL_ALM	70.3	BOOL	FALSE	Unterste Grenze L_ALM durch Istwert oder Regeldifferenz unterschritten.
QLMN_HLM	70.4	BOOL	FALSE	Der Ausgang QLMN_HLM meldet das Erreichen der oberen Stellwert-Begrenzung LMN_HLM. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung).
QLMN_LLM	70.5	BOOL	FALSE	Der Ausgang QLMN_LLM meldet das Erreichen der unteren Stellwert-Begrenzung LMN_LLM. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QPAR_F	70.6	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang QPAR_F angezeigt.
QCH_F	70.7	BOOL	FALSE	Der Ausgang QCH_F wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPAR_F = 1 oder QMOD_F = 1 gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	71.0	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang QUPRLM gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	71.1	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang QDNRLM gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	71.2	BOOL	FALSE	Der Ausgang QSP_HLM meldet das Erreichen der oberen Sollwert-Begrenzung SP_HLM.
QSP_LLM	71.3	BOOL	FALSE	Der Ausgang QSP_LLM meldet das Erreichen der unteren Sollwert-Begrenzung SP_LLM.
QLMNUP	71.4	BOOL	FALSE	Ausgang "Stellwertsignal Hoch". (Nur beim Schrittregler oder Impulsregler)
QLMNDN	71.5	BOOL	FALSE	Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QTUN_ON	71.6	BOOL	FALSE	QTUN_ON = TRUE zeigt an, dass eine Optimierung läuft.
PAR_ACT	71.7	BOOL	FALSE	Reglerparameter aktualisieren
SP	72.0	REAL	0.0	Effektiv wirkender Sollwert.
ER	76.0	REAL	0.0	Regeldifferenz
DISV	80.0	REAL	0.0	Effektiv wirkende Störgröße. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_A	84.0	REAL	0.0	Am Ausgang LMN_A wird beim kontinuierlichen oder Impuls-Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).
LMN_B	88.0	REAL	0.0	Am Ausgang LMN_B wird beim kontinuierlichen oder Impuls-Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100.0 bis 100.0 (%).
PHASE	92.0	INT	0	Zeigt den Ablauf der Regleroptimierung an. *) 0..7

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
STATUS_H	94.0	INT	0	Diagnosewert für Wendepunktsuche bei der Heizoptimierung. Siehe Parameter FMT_PID.STATUS_H (Seite 6266)
STATUS_C	96.0	INT	0	Diagnosewert für Wendepunktsuche bei der Kühloptimierung. Siehe Parameter FMT_PID.STATUS_C (Seite 6267)
STATUS_D	98.0	INT	0	Diagnosewert für Reglerentwurf bei der Heizoptimierung. Siehe Parameter FMT_PID.STATUS_D (Seite 6269)
ZONE_TUN	100.0	WORD	W#16#0	In der HEX-Darstellung entspricht jede der vier Stellen einem Kanal, in folgender Reihenfolge von links nach rechts: Kanal 0, 1, 2, 3. ZONE_TUN = 0 bedeutet, dass der angewählte Kanal nicht mit anderen Kanälen zusammen optimiert werden wird. Ein Wert <> 0000 zeigt - jeweils mit einer 1 - die Kanäle, die zusammen optimiert werden.
GAIN_P	152.0	REAL	0.0	Identifizierte Prozessverstärkung (phys. Größe) /%. Beim Streckentyp I wird GAIN_P tendenziell zu klein geschätzt.
TU	156.0	REAL	0.0	Identifizierte Verzugszeit vom Prozess. TU ≥ 3*Abtastzeit
TA	160.0	REAL	0.0	Identifizierte Ausgleichszeit vom Prozess. Beim Streckentyp I wird TA tendenziell zu klein geschätzt.
KIG	164.0	REAL	0.0	Maximal möglicher Anstieg des Istwertes. (phys. Größe) /s GAIN_P = 0.01 * KIG * TA
N_PTN	168.0	REAL	0.0	Ordnung der Strecke. Auch "nicht ganzzahlige Werte" sind möglich. Zulässig sind Werte von 1.01 bis 10.0.
TM_LAG_P	172.0	REAL	0.0	Zeitkonstante eines PTN-Modells. (sinnvolle Werte nur für N_PTN>=2).
T_P_INF	176.0	REAL	0.0	Zeit von der Prozessanregung bis zum Wendepunkt.
P_INF	180.0	REAL	0.0	Istwertänderung von der Prozessanregung bis zum Wendepunkt.
LMN0	184.0	REAL	0.0	Stellwert zu Beginn der Optimierung Wird in Phase 1 ermittelt (Mittelwert) Zulässig sind Werte von -100.0 bis 100.0 (%).
PV0	188.0	REAL	0.0	Istwert zu Beginn der Optimierung Wird in Phase 1 ermittelt (Mittelwert).
PVDT0	192.0	REAL	0.0	Istwertsteigung zu Beginn der Optimierung Vorzeichen angepasst. (phys. Größe) /s
PVDT	196.0	REAL	0.0	Momentane Istwertsteigung Vorzeichen angepasst. (phys. Größe) /s
PVDT_MAX	200.0	REAL	0.0	Max. Istwertänderung (phys. Größe) /s Die Ableitung des Istwertes erreicht ein Maximum (Vorzeichen angepasst, immer > 0), wird verwendet zur Berechnung von TU und KIG.
NOI_PVDT	204.0	REAL	0.0	Rauschanteil in PVDT_MAX in % Je größer der Rauschanteil, desto ungenauer (sanfter) die Reglerparameter.
NOISE_PV	208.0	REAL	0.0	Absolutes Istwert-Rauschen (phys. Größe) /s Differenz zwischen maximalem und minimalem Istwert in Phase 1.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
FIL_CYC	212.0	INT	0	Anzahl der Zyklen des Mittelwertfilters Der Istwert wird über FIL_CYC Zyklen gemittelt. FIL_CYC wird bei Bedarf automatisch von 1 bis max. 1024 erhöht.
POI_CMAX	214.0	INT	0	Maximale Anzahl Zyklen nach Wendepunkt Diese Zeit wird genutzt, um bei Messrauschen einen weiteren (d. h. besseren) Wendepunkt zu finden. Erst dann wird die Optimierung beendet.
POI_CYCL	216.0	INT	0	Anzahl Zyklen nach Wendepunkt

Siehe auch

Parameter STATUS_H (Seite 6266)

Parameter STATUS_C (Seite 6267)

Parameter STATUS_D (Seite 6269)

FMT_PV

Beschreibung FMT_PV

Die Anweisung FMT_PV dient zum Lesen oder Schreiben von Prozesswerten (Analog- und Digitaleingangswerte) zur Unterstützung der Inbetriebsetzung.

Aufruf

FMT_PV muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

FMT_PV benötigt keinen Initialisierungslauf.

Setzen Sie den Parameter LOAD_PV zyklisch auf TRUE, falls Sie zyklisch simulierte Prozesswerte zur FM 355-2 schreiben wollen.

Setzen Sie den Parameter READ_PV zyklisch auf TRUE, um immer die aktuellen Prozesswerte zu erhalten.

Nach erfolgreichem Schreiben bzw. Lesen der Prozesswerte setzt FMT_PV den Parameter LOAD_PV bzw. READ_PV zurück.

Simulation der Analogwerte(LOAD_PV = TRUE)

Die Simulation der Analogwerte für die Kanäle 0 bis 3 wird eingeschaltet über die Schalter S_AION[i] bzw. S_PVON[i], wobei 0 ≤ i ≤ 3 ist. Im Bild unten ist dargestellt, an welcher Stelle der simulierte Analogwert wirksam wird.

Die Simulationswerte für die Kanäle 0 bis 3 geben Sie über die Parameter PV_SIM[i] vor.

Sie können die Simulationswerte an zwei Stellen wirksam werden lassen:

- S_AION[i] = TRUE (0 ≤ i ≤ 3)
  An Stelle des Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.
- S_PVON[i] = TRUE (0 ≤ i ≤ 3)
  An Stelle des aufbereiteten Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.

Simulation der Digitalwerte(LOAD_PV = TRUE)

Die Simulation der Werte für die Digitaleingänge 0 bis 7 wird eingeschaltet über den Schalter S_DION[i], wobei 0 ≤ i ≤ 7 ist.

Die Simulationswerte geben Sie über die Parameter DI_SIM[i] vor.

- S_DION[i] = TRUE (0 ≤ i ≤ 7)
  An Stelle des Werts des Digitaleingangs i der Baugruppe wird der Wert DI_SIM[i] verwendet.

Hinweis

Die LEDs I0 bis I7 zeigen auch bei Simulation immer den Zustand des zugehörigen Digitaleingangs an.

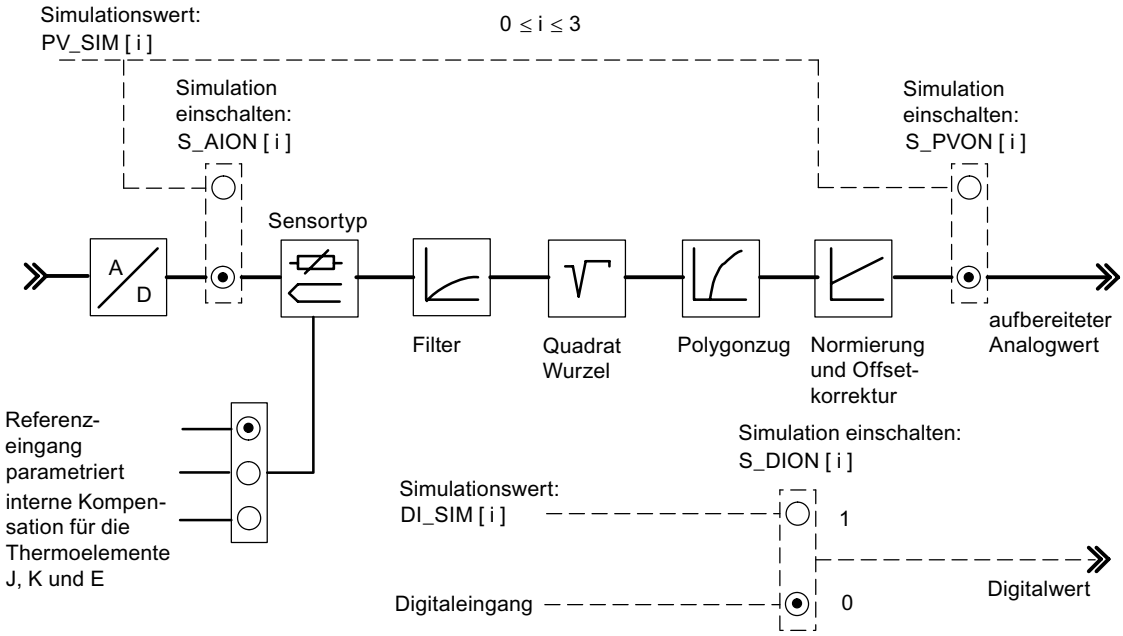

Hinweis

Das Einschalten und die Vorgabe der Simulationswerte (Forcen) erfolgt nicht über die Projektiersoftware. Daher sind die entsprechenden Schalter und Verbindungslinien gestrichelt gezeichnet.

Anzeigen der Prozesswerte (READ_PV = TRUE)

Folgende Werte werden angezeigt:

- An den Parametern STAT_DI[0] bis STAT_DI[7] wird der tatsächliche Zustand der Digitaleingänge 0 bis 7 angezeigt, auch wenn diese simuliert werden.
- An den Parametern DIAG[0].PV_PER bis DIAG[3].PV_PER wird der Wert der Analogeingänge 0 bis 3 in der Einheit mA bzw. mV angezeigt. Falls die Simulation des Analogeingangswertes eingeschaltet wurde, wird der simulierte Wert angezeigt.
- An den Parametern DIAG[0].PV_PHY bis DIAG[3].PV_PHY wird der aufbereitete Analogeingangswert 0 bis 3 in physikalischer Einheit angezeigt. Falls die Simulation des aufbereiteten physikalischen Analogwertes eingeschaltet wurde, wird der simulierte Wert angezeigt.

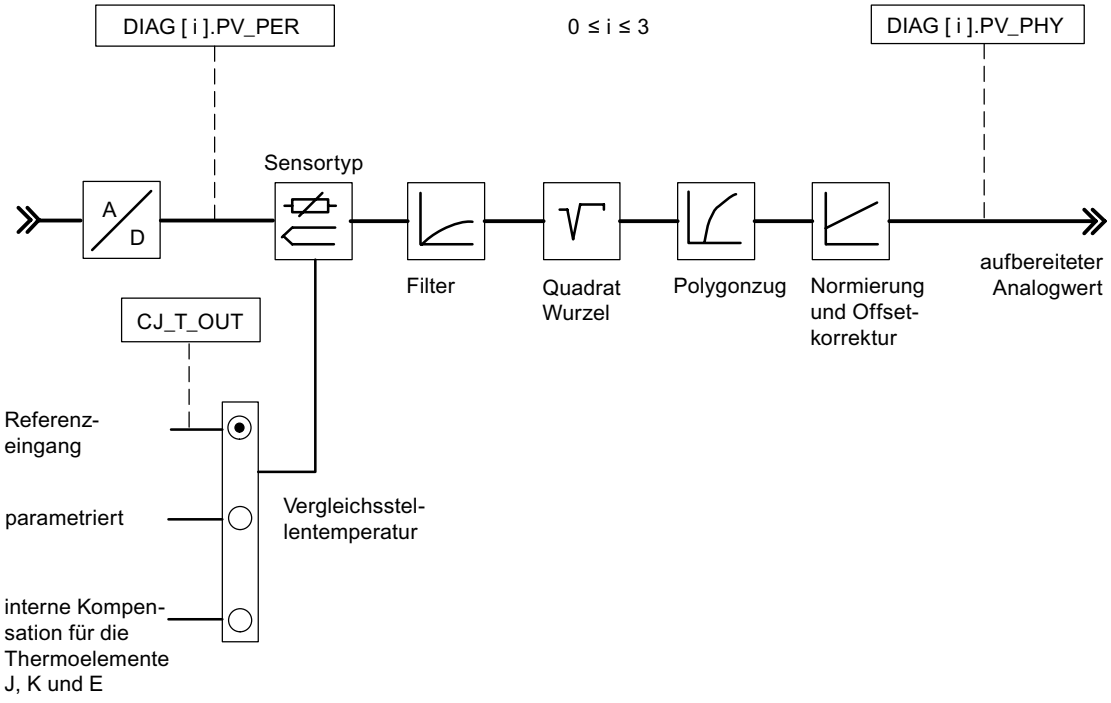

Hinweis

Die Aktualisierung der Prozesswerte kann sich bei Belastung der FM 355-2 gegenüber der Anweisung FMT_PID verzögern.

Anlauf

Bei Wiederanlauf der FM 355-2 nach Netz Aus stehen die Simulationsschalter auf der FM 355-2 wieder auf FALSE.

Eingangsparameter FMT_PV

Tabelle 4-263

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
S_AION	0.0	ARRAY [0..3] of BOOL	FALSE	Wenn z. B. der Schalter S_AION[1] auf TRUE gesetzt wird, dann wird an Stelle des Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.
S_PVON	2.0	ARRAY [0..3] of BOOL	FALSE	Wenn z. B. der Schalter S_PVON[1] auf TRUE gesetzt wird, dann wird an Stelle des aufbereiteten Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.
PV_SIM	4.0	ARRAY [0..3] of REAL	0.0	Am Eingang PV_SIM[1] wird z. B. der Simulationswert für den Analogeingang 1 vorgegeben. Falls S_PVON = TRUE ist, dann wird hier der aufbereitete Analogeingangswert vorgegeben. Falls S_PVON = FALSE und S_AION = TRUE sind, dann wird hier der Analogeingangswert in mA bzw. mV vorgegeben, der noch durch die Aufbereitungsfunktionen in einen aufbereiteten Wert gewandelt wird. 0.0 bis 20.0 [mA] oder -1500 bis +10000 [mV] oder technischer Wertebereich
S_DION	20.0	ARRAY [0..7] of BOOL	FALSE	Wenn z. B. S_DION[1] auf TRUE gesetzt wird, dann wird als Digitalwert an Stelle des Digitaleingangs 1 der Baugruppe der Wert DI_SIM[1] verwendet.
DI_SIM	22.0	ARRAY [0..7] of BOOL	FALSE	
MOD_ADDR	24.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Siehe auch

Ausgangsparameter FMT_PV (Seite 6286)

Durchgangsparameter FMT_PV (Seite 6286)

Ausgangsparameter FMT_PV

Tabelle 4-264

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
STAT_DI	26.0	ARRAY [0..7] of BOOL	FALSE	An den Parametern STAT_DI werden die Zustände der Digitaleingänge 0 bis 7 angezeigt.
DIAG[x].PV_PER	28+ (Kanalnummer x8)	ARRAY [0..3] of STRUCT	0.0	Am Parameter DIAG[x].PV_PER wird z. B. der Analogeingangswert der Baugruppe in der Einheit mA bzw. mV angezeigt.
DIAG[x].PV_PHY	32+ (Kanalnummer x8)	ARRAY [0..3] of STRUCT	0.0	Am Parameter DIAG[x].PV_PHY wird z. B. der aufbereitete Analogeingangswert der Baugruppe in physikalischer Einheit angezeigt.
RET_VALU	60.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter FMT_PV

Tabelle 4-265

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LOAD_PV	62.0	BOOL	FALSE	Ist der Durchgangsparameter LOAD_PV gesetzt, dann werden die Simulationswerte in PV_SIM und DI_SIM entsprechend der Schalterstellungen auf die FM 355-2 geschrieben und der Durchgangsparameter rückgesetzt.
READ_PV	62.1	BOOL	FALSE	Wenn der Parameter READ_PV = TRUE ist, dann wird die Vergleichsstellentemperatur von der Baugruppe gelesen. Anschließend wird der Durchgangsparameter zurückgesetzt.

Siehe auch

Eingangsparameter FMT_PV (Seite 6285)

Ausgangsparameter FMT_PV (Seite 6286)

FMT_CJ_T

Beschreibung FMT_CJ_T

Die Anweisung FMT_CJ_T dient zum Lesen der gemessenen Vergleichsstellentemperatur und zum Online-Ändern der parametrierten Vergleichsstellentemperatur. Dies ist erforderlich, wenn eine Temperaturregelung mit mehreren FM 355-2 mit Thermoelementeingängen betrieben werden soll, ohne dass an jede FM 355-2 ein Pt 100 angeschlossen werden muss.

Aufruf

FMT_PAR muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 355-2 zugreifen.

Bei Einsatz der FM 355-2 in dezentraler Peripherie kann es einige Aufrufzyklen dauern, bis der Parameter zur FM 355-2 übertragen worden ist. Solange die Übertragung noch nicht abgeschlossen ist, behält der Parameter LOAD_CJ den Wert TRUE. Rufen Sie daher bei der Änderung von Parametern den FMT_CJ_T so lange wiederholt auf, bis der Baustein LOAD_CJ = FALSE setzt.

Arbeitsweise

Wenn z. B. bei einer Extruderregelung mit mehr als vier Heizzonen die Vergleichsstellentemperatur mit einer FM 355-2 gemessen wird, kann diese mit READ_CJ = TRUE am Parameter CJ_T_OUT ausgelesen werden und bei den anderen FM 355-2 über den CJ_TEMP und LOAD_CJ parametriert werden.

Am Parameter CJ_T_OUT steht die an der Vergleichsstelle gemessene Vergleichsstellentemperatur in Grad C oder in Grad F (je nachdem, welche Temperatureinheit parametriert wurde).

Am Parameter CJ_T_OUT wird 0.0 angezeigt, wenn:

- kein Sensortyp "Thermoelement" parametriert wurde,
- bei allen Analogeingängen die parametrierte Vergleichsstellentemperatur gewählt wurde,
- bei allen Analogeingängen die interne Kompensation gewählt wurde.

Eingangsparameter FMT_CJ_T

Tabelle 4-266

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CJ_TEMP	2.0	REAL	0.0	Am Parameter CJ_TEMP kann die Vergleichsstellentemperatur vorgegeben werden.

Ausgangsparameter FMT_CJ_T

Tabelle 4-267

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CJ_T_OUT	6.0	REAL	0.0	Am Ausgang CJ_T_OUT wird die von der Baugruppe gemessene Vergleichsstellentemperatur angezeigt, falls ein Thermoelementeingang parametriert ist und die Vergleichsstellentemperatur nicht parametriert vorgegeben wird.
RET_VALU	10.0	WORD	W#16#0	RET_VALU enthält das 2te und 3te Byte vom Parameter STATUS der Anweisungen RDREC und WRREC.

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

Durchgangsparameter FMT_CJ_T

Tabelle 4-268

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LOAD_CJ	12.0	BOOL	FALSE	Ist der Durchgangsparameter LOAD_CJ gesetzt, wird die parametrierte Vergleichsstellentemperatur auf der Baugruppe mit dem Wert CJ_TEMP überschrieben und der Durchgangsparameter rückgesetzt.
READ_CJ	12.1	BOOL	FALSE	Wenn der Parameter READ_CJ = TRUE ist, dann wird die Vergleichsstellentemperatur von der Baugruppe gelesen. Anschließend wird der Durchgangsparameter zurückgesetzt.

FM 455 PID Control

PID_FM_455

Beschreibung PID_FM_455

Die Anweisung PID_FM_455 dient dazu im laufenden Betrieb:

- Bedienparameter auf die FM 455 zu übertragen
- Reglerparameter auf die FM 455 zu übertragen
- Prozesswerte aus der FM 455 zu lesen

Bedienparameter sind alle Durchgangsparameter, die im Instanz-DB der Anweisung zwischen den Parametern op_par und cont_par liegen.

Reglerparameter sind alle Durchgangsparameter, die im Instanz-DB der Anweisung nach dem Parameter cont_par liegen.

Prozesswerte sind alle Ausgangsparameter der Anweisung nach dem Parameter out_par.

Die benötigten Daten werden in einem Instanz-DB auf der CPU abgelegt. Für jeden genutzten Reglerkanal benötigen Sie einen Instanz-DB. Nach Erzeugen eines Instanz-DB sind alle Durchgangsparameter auf FALSE gesetzt.

Aufruf

PID_FM_455 wird normalerweise im Weckalarm-OB 35 aufgerufen und benötigt einen Initialisierungslauf, der durch Setzen des Parameters COM_RST = TRUE im Anlauf der CPU veranlasst wird. Ein Aufruf der Anweisung im Anlauf-OB ist möglich, aber nicht erforderlich. Nach dem Initialisierungslauf setzt PID_FM_455 den Parameter COM_RST auf FALSE.

PID_FM_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Wenn Sie die Anweisung PID_FM_455 als Multiinstanz-DB aufrufen, steht Ihnen keine Inbetriebnahmeoberfläche zur Verfügung. Sie müssen die FM 455 in diesem Fall über eine Beobachtungstabelle in Betrieb nehmen.

Anlauf

Beim Anlauf der CPU und beim STOP-RUN-Übergang der CPU werden die Parameter an der FM 455 mit den Parametern aus dem SDB der CPU überschrieben.

Verhalten im Fehlerfall

Der Ausgangsparameter RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen RD_REC und WR_REC. RET_VALU kann ausgewertet werden, wenn die Parameter READ_VAR und LOAD_PAR nicht zurückgesetzt werden.

Beim Aufruf der Anweisung PID_FM_455 kann ein Peripheriezugriffsfehler (PZF) auftreten, wenn die FM 455 nicht gesteckt ist oder keine Spannungsversorgung hat. In diesem Fall geht die CPU in STOP, wenn kein OB 122 in der CPU geladen ist.

Arbeitsweise PID_FM_455

Einfluss der Parameter LOAD_OP, READ_VAR und LOAD_PAR

Im Initialisierungslauf liest PID_FM_455 die Parameter aus den Systemdaten und legt sie in seinem Instanz-DB ab. Instanz-DB und Systemdaten sind abgeglichen.

Die folgende Tabelle zeigt, wie die Funktion der Anweisung durch die Parameter LOAD_OP, READ_VAR und LOAD_PAR beeinflusst wird.

Parameter	TRUE	FALSE
LOAD_OP	Übertragen der Bedienparameter von der CPU auf die FM 455 mit der Anweisung WR_REC	Übertragen der Bedienparameter von der CPU auf die FM 455 über direkte Peripheriezugriffe im Multiplex-Verfahren
	Längere Laufzeit	Geringere Laufzeit
	Die Werte sind nach einem Zyklus der CPU oder FM wirksam. Der längere Zyklus ist maßgeblich.	Max. sind 3 Zyklen der CPU oder FM notwendig bis die Werte wirksam werden. Der längere Zyklus ist maßgeblich
READ_VAR	Lesen der Prozesswerte aus der FM 455 mit der Anweisung RD_REC	Lesen der Prozesswerte aus der FM 455 über direkte Peripheriezugriffe im Multiplex-Verfahren
	Längere Bearbeitungszeit	Geringere Bearbeitungszeit
	Alle Parameter werden aus der FM 455 gelesen	Folgende Parameter werden **nicht** aus der FM 455 gelesen • SP (Sollwert aus der FM) • ER (Regeldifferenz) • DISV (Störgröße) • LMN_A • LMN_B
	Folgende Parameter werden bei **jedem** Aufruf der Anweisung aktualisiert. • PV (Istwert) • LMN (Stellwert) • die Binäranzeigen	Folgende Parameter werden bei jedem **vierten** Aufruf der Anweisung aktualisiert. • PV (Istwert) • LMN (Stellwert) • die Binäranzeigen
	Im Anlauf der CPU übernimmt PID_FM_455 folgende Werte aus der FM 455: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON Wenn diese Werte auf der CPU geändert wurden, werden Sie beim nächsten Anlauf der CPU mit den Werten aus der FM 455 überschrieben	Im Anlauf der CPU werden folgende Werte von der Anweisung gelesen, nicht aus der FM: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON
LOAD_PAR	Alle Reglerparameter werden von der CPU auf die FM 455 geladen. Dies ist sinnvoll, wenn die Regelparameter abhängig von Prozesszuständen geändert werden sollen.	Keine Übertragung der Reglerparameter

Nach erfolgreicher Datenübertragung werden die Parameter LOAD_OP, READ_VAR und LOAD_PAR wieder auf FALSE gesetzt.

Besonderheiten bei Übertragung im Multiplex-Verfahren

Das Multiplexen der zu übertragenden Daten bei Zugriff auf die FM 455 über direkte Peripheriezugriffe wird über PID_FM_455 gesteuert. Die Multiplexsteuerung durch PID_FM_455 funktioniert nicht, wenn zwei Instanzen der Anweisung PID_FM_455 auf die gleiche Kanalnummer einer Baugruppe zugreifen. Das Resultat sind falsche Parameter in der FM 455 (z. B. Sollwert und Handstellwert) und falsche Anzeigen an den Ausgangsparametern der Anweisung PID_FM_455.

Abspeichern der Parameter im EEPROM

Beim programmgesteuerten Umparametrieren (LOAD_PAR, LOAD_OP) der Reglerbaugruppe durch die Anweisung PID_FM_455 vergrößert sich dessen Laufzeit. Die neuen Parameter sind immer sofort wirksam und werden zusätzlich in einem nichtflüchtigen Speicher (EEPROM) abgelegt. Nach jedem Speichern der Parameter im EEPROM wird ein erneutes Abspeichern um 30 Minuten verzögert, da die Lebensdauer des EEPROMs durch die Anzahl der Schreibvorgänge begrenzt ist. Nach Wiederkehr der Versorgungsspannung ist das Abspeichern neuer Parameter im EEPROM sofort möglich. Ob die Umparametrierung der Reglerbaugruppe durch die Anweisung PID_FM_455 stoßfrei erfolgt, hängt von der Wahl der Parameter ab.

Bearbeitungszeiten des PID_FM_455

Randbedingungen			Bearbeitungszeit in
READ_VAR	LOAD_OP	LOAD_PAR	CPU 414-2DP
FALSE	FALSE	FALSE	0,077 ms
TRUE	FALSE	FALSE	2,36 ms
TRUE*)	TRUE	FALSE	4,48 ms
FALSE	FALSE	TRUE	2,59 ms
TRUE	FALSE	TRUE	5,15 ms
TRUE*)	TRUE	TRUE	7,1 ms
*) Wenn LOAD_OP = TRUE ist, dann wird READ_VAR von PID_FM_455 auch auf TRUE gesetzt.			

Eingangsparameter PID_FM_455

Tabelle 4-269

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässig sind die Werte 1-16
PHASE	4.0	INT	0	Der Parameter PHASE kann mit dem Ausgangsparameter PHASE der Anweisungen TUN_EC oder TUN_ES (PID Self-Tuner) verschaltet werden. Im Dialog "Inbetriebnahme" kann dann der Phasenzustand des PID Self-Tuners im Klartext angezeigt werden. Dieser Parameter hat für das OP keine Bedeutung.

Ausgangsparameter PID_FM_455

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	6.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
out_par	8.0	WORD	W#16 #3130	Der Parameter out_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Ausgangsparameter, die von der Baugruppe gelesen werden, wenn READ_VAR = TRUE gesetzt wird.
SP	10.0	REAL	0.0	Am Ausgang "Sollwert" wird der effektiv wirkende Sollwert ausgegeben.
PV	14.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	18.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.
DISV	22.0	REAL	0.0	Am Ausgang "Störgröße" wird die effektiv wirkende Störgröße ausgegeben. Zulässig sind Werte von -100 bis 100 %
LMN	26.0	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Parameter LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100 bis 100 %
LMN_A	30.0	REAL	0.0	Am Ausgang "Stellwert A der Splitrangefunktion / Stellungsrückmeldung" wird beim kontinuierlichen Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Der Ausgang LMN_A kann nur zur ungefähren Anzeige einer entsprechenden simulierten Stellgröße verwendet werden. Der Startwert LMNRSVAL der simulierten Stellungsrückmeldung ist hierbei entsprechend zu parametrieren und wird beim Setzen von LMNRS_ON wirksam. Zulässig sind Werte von -100 bis 100 %
LMN_B	34.0	REAL	0.0	Am Ausgang "Stellwert B der Splitrangefunktion" wird beim kontinuierlichen Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100 bis 100 %
QH_ALM	38.0	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Alarm angesprochen" wird das Überschreiten der Grenze H_ALM gemeldet.
QH_WRN	38.1	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Warnung angesprochen" wird das Überschreiten der Grenze H_WRN gemeldet.
QL_WRN	38.2	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Warnung angesprochen" wird das Unterschreiten der Grenze L_WRN gemeldet.
QL_ALM	38.3	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Alarm angesprochen" wird das Unterschreiten der Grenze L_ALM gemeldet.
QLMN_HLM	38.4	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Stellwertes angesprochen" meldet die Überschreitung der oberen Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMN_LLM	38.5	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Stellwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QPARA_F	38.6	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang "Parametrierfehler" angezeigt. Diesen Parametrierfehler können Sie auch im Dialog "Online & Diagnose" der Baugruppe auslesen.
QCH_F	38.7	BOOL	FALSE	Der Ausgang "Kanalfehler" wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPARA_F = 1 oder QMOD_F = 1 gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	39.0	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "positive Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	39.1	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "negative Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	39.2	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Sollwertes angesprochen" meldet die Überschreitung der oberen Begrenzung.
QSP_LLM	39.3	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Sollwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung.
QLMNUP	39.4	BOOL	FALSE	Dies ist der Ausgang "Stellwertsignal Hoch. (Nur beim Schrittregler oder Impulsregler)
QLMNDN	39.5	BOOL	FALSE	Dies ist der Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QID	39.6	BOOL	FALSE	QID = TRUE zeigt an, dass eine Identifizierung läuft (nicht dass sie eingeschaltet ist). Nach dem Ende der Identifizierung kann das Identifizierungsergebnis über den Parameter IDSTATUS der Anweisung CH_DIAG_455 ausgelesen werden.
QSPOPON	40.0	BOOL	FALSE	Am Ausgang "Sollwertbedienung eingeschaltet" wird angezeigt, ob der Sollwert durch das Konfigurationswerkzeug bedient wird. Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
QLMNSAFE	40.1	BOOL	FALSE	Ist der Ausgang "Sicherheitsbetrieb" gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNOPON	40.2	BOOL	FALSE	Am Ausgang "Stellwertbedienung eingeschaltet " wird angezeigt, ob der Stellwert durch das Konfigurations-werkzeug bedient wird. Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
QLMNTRK	40.3	BOOL	FALSE	Der Ausgang "Nachführbetrieb" zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	40.4	BOOL	FALSE	Der Ausgang "Hand = 1; Automatik = 0" zeigt an, ob der Stellwert auf den externen Stellwert LMN_RE gesetzt wird (Hand = 1) oder nicht.
QLMNR_HS	40.5	BOOL	FALSE	Der Ausgang "oberes Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am oberen Anschlag befindet. QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNR_LS	40.6	BOOL	FALSE	Der Ausgang "unteres Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am unteren Anschlag befindet. QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
QLMNR_ON	40.7	BOOL	FALSE	Der Ausgang "Stellungsrückmeldung eingeschaltet" zeigt die eingestellte Betriebsart "Schrittregler mit Stellungsrückmeldung" bzw. "Schrittregler ohne Stellungsrückmeldung" an.
QFUZZY	41.0	BOOL	FALSE	Ist der Ausgangsparameter QFUZZY = 1, arbeitet der Regler mit dem Fuzzy-Algorithmus.
QSPLEPV	41.1	BOOL	FALSE	Der Ausgang "Anzeige vom FUZZY-Regler: Sollwert < Istwert" wird bei eingeschaltetem Fuzzy-Regler gesetzt, wenn der Sollwert kleiner als der wirksame Istwert ist.
QSPR	41.2	BOOL	FALSE	Ist der Ausgang "Split-Range-Betrieb" gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QMAN_FC	41.4	BOOL	FALSE	Der Ausgang "QMAN_FC" wird in folgenden beiden Fällen gesetzt: • Der Folgeregler ist in Handbetrieb und der Führungsregler wird auf den Istwert des Folgereglers nachgeführt. • Der I-Anteil des Führungsreglers wird angehalten, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind oder weil der Folgeregler in Handbetrieb ist.
QPARABUB	41.7	BOOL	FALSE	Dieser Parameter wird von der FM gesetzt, wenn Bedienparameter über OP geändert wurden. Wenn READ_VAR = TRUE ist und wenn diese Anzeige von der FM gesetzt wird, dann liest die Anweisung PID_FM_455 die Parameter SP_OP_ON, LMNOP_ON, SP_OP und LMN_OP aus der FM und legt sie im Instanz-DB ab. Die Anweisung übernimmt somit den Bedienzustand der FM. Nach dem Lesevorgang wird der Parameter auf FALSE gesetzt.
QMOD_F	42.0	BOOL	FALSE	Die Anweisung überprüft das erfolgreiche Schreiben und Lesen eines Datensatzes. Bei erkannten Fehlern wird der Ausgang "Baugruppenfehler" gesetzt. Fehlerursache kann sein: Eine falsche Baugruppenadresse am Parameter MOD_ADDR, eine falsche Kanalnummer am Parameter CHANNEL oder eine defekte Baugruppe.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

Durchgangsparameter PID_FM_455

Tabelle 4-270

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
[1] Vorbelegungswerte der Baugruppe nach dem ersten Anlauf des PID_FM_455 mit COM_RST = TRUE				
COM_RST	44.0	BOOL	FALSE	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung PID_FM_455 einen Initialisierungslauf durch. Dabei werden die Reglerparameter (dies sind alle Parameter nach cont_par) aus der FM gelesen und im Instanz-DB abgelegt. Weiterhin findet eine Prüfung der Parameter MOD_ADDR und CHANNEL auf Zulässigkeit statt. Nach dem Initialisierungslauf wird der Parameter auf FALSE gesetzt.
LOAD_OP	44.1	BOOL	FALSE	Ist der Durchgangsparameter "Betriebsparameter in FM 455 laden" gesetzt, werden die Betriebsparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
READ_VAR	44.2	BOOL	FALSE	Ist der Durchgangsparameter "Variablen von FM 455 lesen" gesetzt, werden die Ausgangsparameter von der Baugruppe gelesen und der Durchgangsparameter rückgesetzt.
LOAD_PAR	44.3	BOOL	FALSE	Ist der Durchgangsparameter "Regelparameter in FM 455 laden" gesetzt, werden die Regelparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
op_par	46.0	WORD	W#16#3130 [1]	Der Parameter op_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Bedienparameter, die zur Baugruppe übertragen werden, wenn LOAD_OP = TRUE gesetzt wird. Das Ende der Bedienparameter wird durch cont_par gekennzeichnet.
SP_RE	48.0	REAL	0.0	Ein externer Sollwert wird am Eingang "externer Sollwert" mit dem Regler verschaltet.
LMN_RE	52.0	REAL	0.0	Ein externer Stellwert wird am Eingang "externer Stellwert" mit dem Regler verschaltet. Zulässig sind Werte von -100 bis 100 %.
SP_OP_ON	56.0	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Sollwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
SAFE_ON	56.1	BOOL	FALSE	Ist der Eingang "Sicherheitsstellung einnehmen" gesetzt, wird ein Sicherheitswert als Stellwert übernommen. Hinweis: Die Stellsignalbearbeitung über LMNDN_OP, LMNUP_OP und LMNSOPON beim Schrittregler ist höher prior als der Sicherheitsstellwert.
LMNOP_ON	56.2	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Stellwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
LMNTRKON	56.3	BOOL	FALSE	Ist der Eingang "Nachführen (LMN über AE)" gesetzt, wird der Stellwert einem Analogeingang (AE) nachgeführt. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
LMN_REON	56.4	BOOL	FALSE	Ist der Eingang "externer Stellwert einschalten" gesetzt, wird der externe Stellwert LMN_RE als Stellwert übernommen.
LMNRHSRE	56.5	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "oberes Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRHSRE = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMNRLSRE	56.6	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "unteres Anschlagsignal der Stellungsrückmeldung" verschaltet. LMNRLSRE = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
LMNSOPON	56.7	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignalbedienung einschalten". Ist LMNSOPON gesetzt, werden die Signale LMNUP_OP und LMNDN_OP als Stellwertsignale übernommen. (Nur beim Schrittregler)
LMNUP_OP	57.0	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignal Hoch Bedienung". Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Hoch Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNDN_OP	57.1	BOOL	FALSE	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Stellwertsignal Tief Bedienung". Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Tief Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNRS_ON	57.3	BOOL	FALSE	Steht keine Stellungsrückmeldung zur Verfügung, so kann diese simuliert werden. Am Eingang "Simulation der Stellungsrückmeldung einschalten" wird die Funktion eingeschaltet. Auch die Anweisungen des PID Self-Tuners haben Zugriff auf diesen Parameter, da zur Optimierung zumindest ein simulierter Stellwert nötig ist, wenn ein Schrittregler ohne Stellungsrückmeldung parametriert wurde. Der simulierte Wert wird am Parameter LMN_A angezeigt. Beim Einschalten der Simulation wird der Wert des Parameters LMNRSVAL als Startwert gesetzt. ACHTUNG: Die Simulation weicht mit zunehmender Zeit immer stärker vor der wirklichen Stellungsrückmeldung ab. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung)
FUZID_ON	57.4	BOOL	FALSE	Am Eingang "Fuzzy-Identifikation einschalten" wird die Identifikation des Fuzzy-Algorithmus eingeschaltet.
SP_OP	58.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Sollwert Bedienung". Ist das Bit SP_OP_ON gesetzt, wird der Wert "Sollwert Bedienung" als Sollwert übernommen.
LMN_OP[1]	62.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Durchgangsparameter "Stellwert Bedienung". Ist das Bit LMNOP_ON gesetzt, wird der Wert "Stellwert Bedienung" als Stellwert übernommen. Zulässig sind Werte von -100 bis 100 %.
LMNRSVAL	66.0	REAL	0.0	Die Anweisungen des PID Self-Tuners haben Zugriff auf den Eingang "Startwert der simulierten Stellungsrückmeldung". Am Parameter wird der Startwert der Simulation eingetragen. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100 bis 100 %.
cont_par	70.0	WORD	W#16#3130 [1]	Der Parameter cont_par darf vom Anwender nicht verändert werden. Er kennzeichnet den Anfang der Reglerparameter, die aus der FM gelesen und im Instanz-DB abgelegt werden, wenn COM_RST = TRUE ist, und die zur FM übertragen werden, wenn LOAD_PAR = TRUE gesetzt wird. Das Ende der Reglerparameter ist das Ende des Instanz-DB.
P_SEL	72.0	BOOL	TRUE [1]	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der Proportionalanteil ist eingeschaltet, wenn der Eingang "P-Anteil einschalten" gesetzt ist.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PFDB_SEL	72.1	BOOL	FALSE[1)]	Im PID-Algorithmus lassen sich die P- und D-Anteile in die Rückführung legen. Der Proportionalanteil liegt in der Rückführung, wenn der Eingang "P-Anteil in Rückführung schalten" gesetzt ist.
MONERSEL	72.2	BOOL	FALSE[1)]	Der Regler besitzt einen Grenzwertmelder, der entweder für den Istwert oder die Regeldifferenz angewendet werden kann. Ist der Eingang "Überwachung: Istwert = 0, Regeldifferenz = 1" gesetzt, wird die Regeldifferenz überwacht.
SDB_SEL	72.4	BOOL	FALSE	SDB_SEL=TRUE: Die Reglerparameter der Anweisung PID_FM_455 werden nach CPU STOP-RUN in der FMx55 von den SDB-Parametern nicht überschrieben.
D_EL_SEL	74.0	INT	0[1)]	Das D-Glied im PID-Algorithmus kann an einen separaten Eingang gelegt werden. Am Eingang "Eingang für D-Glied" wird dieser ausgewählt. • 0: Regeldifferenz • 1 bis 16: Analogeingang 1 bis 16 • 17: negativer Istwert, D-Anteil in der Rückführung
SP_HLM	76.0	REAL	100.0[1)]	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert obere Begrenzung" gibt die obere Begrenzung an. SP_HLM > SP_LLM
SP_LLM	80.0	REAL	0.0[1)]	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert untere Begrenzung" gibt die untere Begrenzung an. SP_LLM < SP_HLM
H_ALM	84.0	REAL	100.0[1)]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Alarm" gibt die oberste Grenze an. H_ALM > H_WRN
H_WRN	88.0	REAL	90.0[1)]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Warnung" gibt die zweite Grenze von oben an. Zulässig sind Werte von H_ALM bis L_WRN.
L_WRN	92.0	REAL	10.0[1)]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Warnung" gibt die zweite Grenze von unten an. Zulässig sind Werte von H_WRN bis L_ALM.
L_ALM	96.0	REAL	0.0[1)]	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Alarm" gibt die unterste Grenze an. L_ALM < L_WRN
HYS	100.0	REAL	1.0[1)]	Zur Vermeidung von Flackern der Überwachungsanzeigen kann am Eingang "Hysterese" eine Hysterese parametriert werden. HYS ≥ 0.0
DEADB_W	104.0	REAL	0.0[1)]	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W ≥ 0.0
GAIN	108.0	REAL	1.0[1)]	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI	112.0	REAL	3000.0[1)]	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. Bei TI = 0 ist der Integrierer ausgeschaltet TI = 0.0 oder TI ≥ 0.5

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
TD	116.0	REAL	0.0[1)]	Der Eingang "Differenzierzeit" bestimmt das Zeitverhalten des Differenzierers. Bei TD = 0 ist der Differenzierer abgeschaltet. TD = 0.0 oder TD ≥ 1.0
TM_LAG	120.0	REAL	5.0[1)]	Der Algorithmus des D-Anteils beinhaltet eine Verzögerung, die am Eingang "Verzögerungszeit des D-Anteils" parametriert werden kann. TM_LAG ≥ 0.5
LMN_SAFE	124.0	REAL	0.0[1)]	Für den Stellwert kann ein Sicherheitswert am Eingang "Sicherheitsstellwert" parametriert werden. Zulässig sind Werte von -100 bis 100 %.
LMN_HLM	128.0	REAL	100.0[1)]	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von LMN_LLM bis 100 %.
LMN_LLM	132.0	REAL	0.0[1)]	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100 bis LMN_HLM.
MTR_TM	136.0	REAL	60.0[1)]	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils von Anschlag zu Anschlag eingetragen. (Nur beim Schrittregler)
PULSE_TM	140.0	REAL	0.2[1)]	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden. (Nur beim Schrittregler oder Impulsregler)
BREAK_TM	144.0	REAL	0.2[1)]	Am Parameter "Mindestpausendauer" kann eine minimale Impulspausenlänge parametriert werden. (Nur beim Schrittregler oder Impulsregler)

Hinweis

Mit LOAD_PAR = TRUE werden alle Regelparameter dauerhaft in das EEPROM der FM 455 geladen.

Mit LOAD_OP = TRUE wird von den Betriebsparametern nur der Sollwert SP_RE dauerhaft in das EEPROM der FM 455 geladen. Alle übrigen Betriebsparameter werden beim Anlauf der FM 455 mit den Werten 0 bzw. FALSE vorbelegt.

Das EEPROM der Baugruppe könnte durch zu häufige Schreibvorgänge zerstört werden. Um dies zu verhindern, wird von der Baugruppe nach dem Beschreiben des EEPROM ein erneutes Beschreiben um 30 Minuten verzögert.

Blockschaltbilder PID_FM_455

Überblick

In den folgenden Abbildungen wird dargestellt, an welchen Stellen die Parameter der Anweisung PID_FM_455 wirken.

Die Parameter wirken bei Dreikomponentenregler, Verhältnis-/Mischungsregler und Festwert- oder Kaskadenregler an den gleichen Stellen. Dies gilt auch für die Parameter, die gleichermaßen beim kontinuierlichen Regler, beim Regler mit Impulsausgang sowie beim Schrittregler vorhanden sind. Daher sind nicht alle Blockschaltbilder dargestellt und nicht in allen Bildern alle Parameter eingezeichnet.

Die Parameter der Anweisung PID_FM_455 sind jedoch alle in den Bildern enthalten – bis auf die Parameter MOD_ADDR, CHANNEL, QMOD_F, QPARA_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_VAR, LOAD_OP.

Wirkung der Eingangsparameter

Regeldifferenzbildung beim Festwert- oder Kaskadenregler

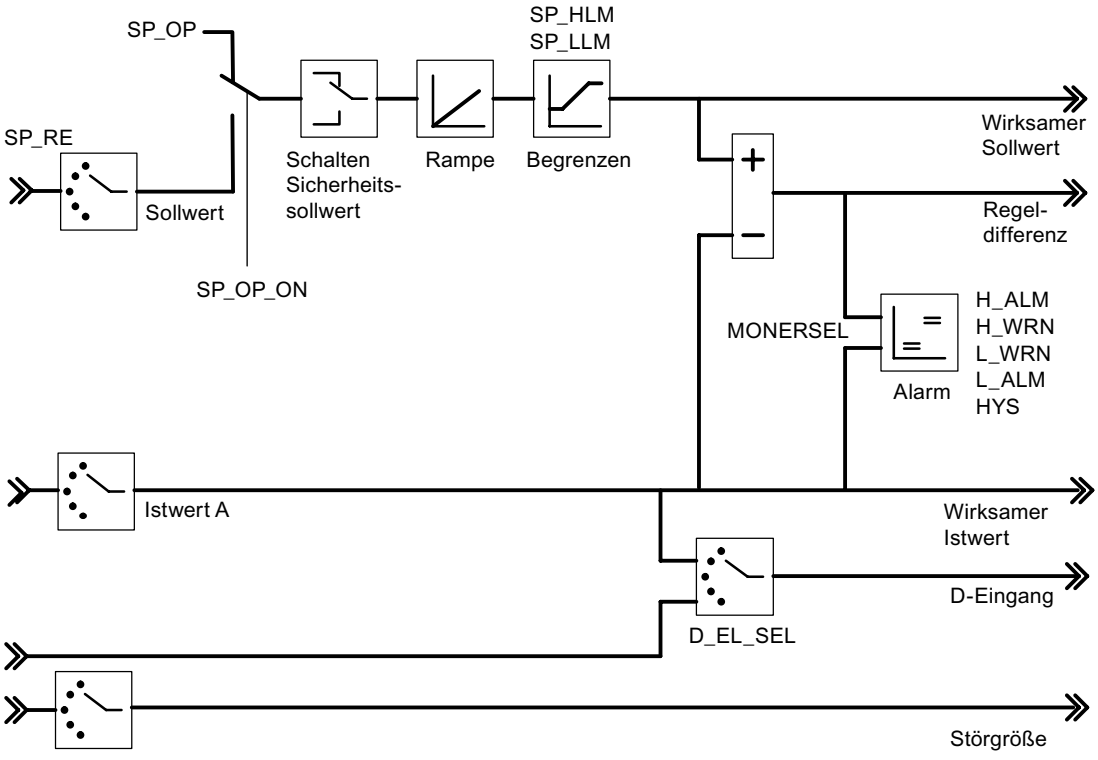

Blockschaltbild des Regelalgorithmus

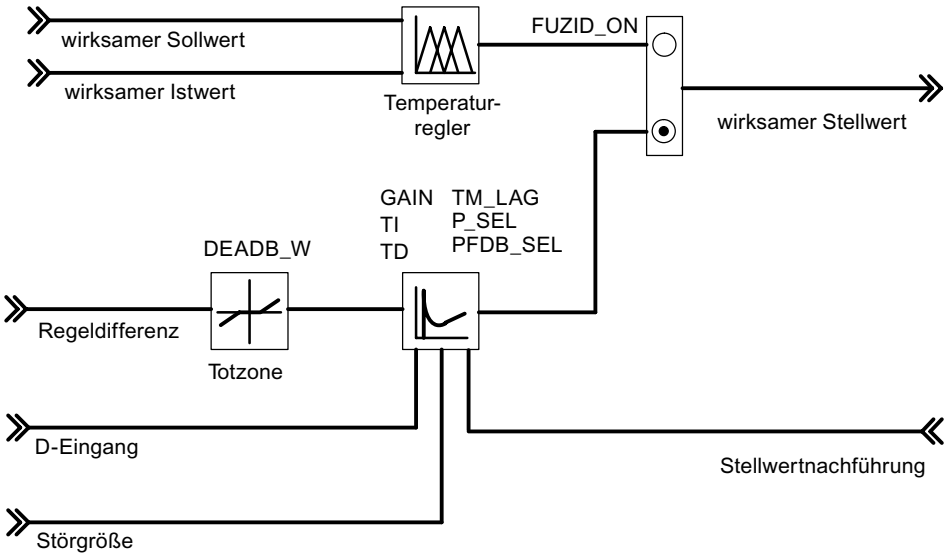

Reglerausgang des K-Reglers

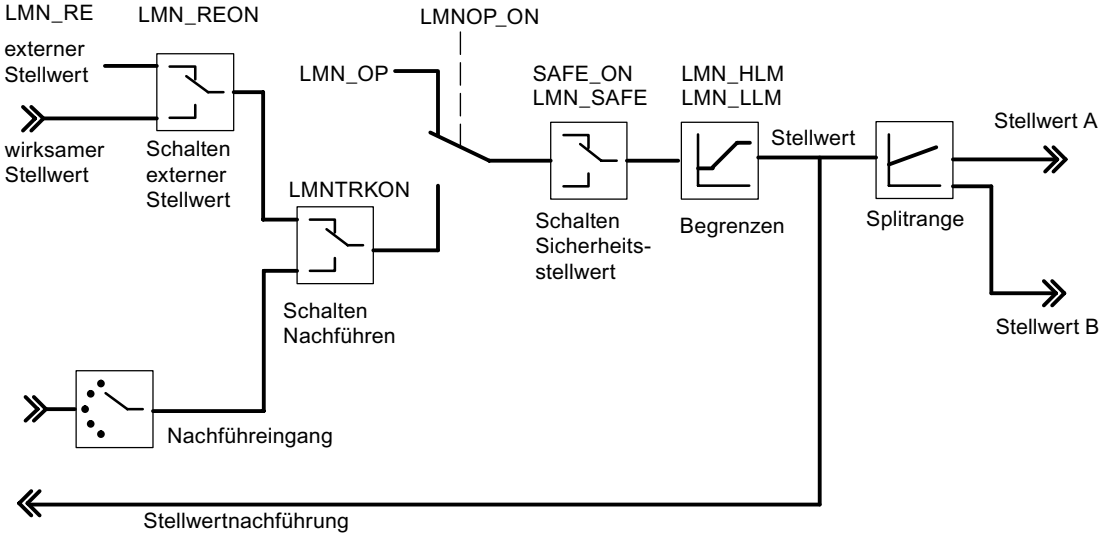

Reglerausgang des S-Reglers (Betriebsart Impulsregler)

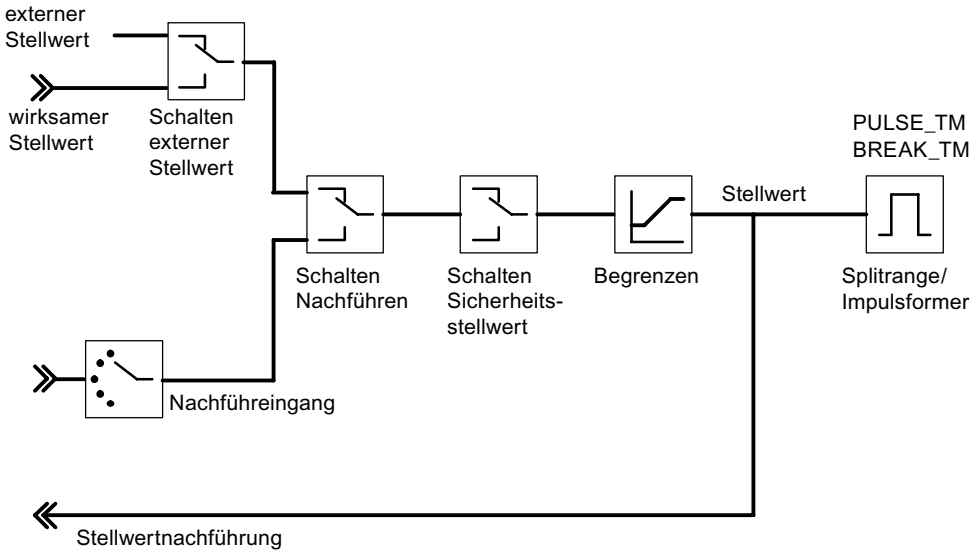

Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung)

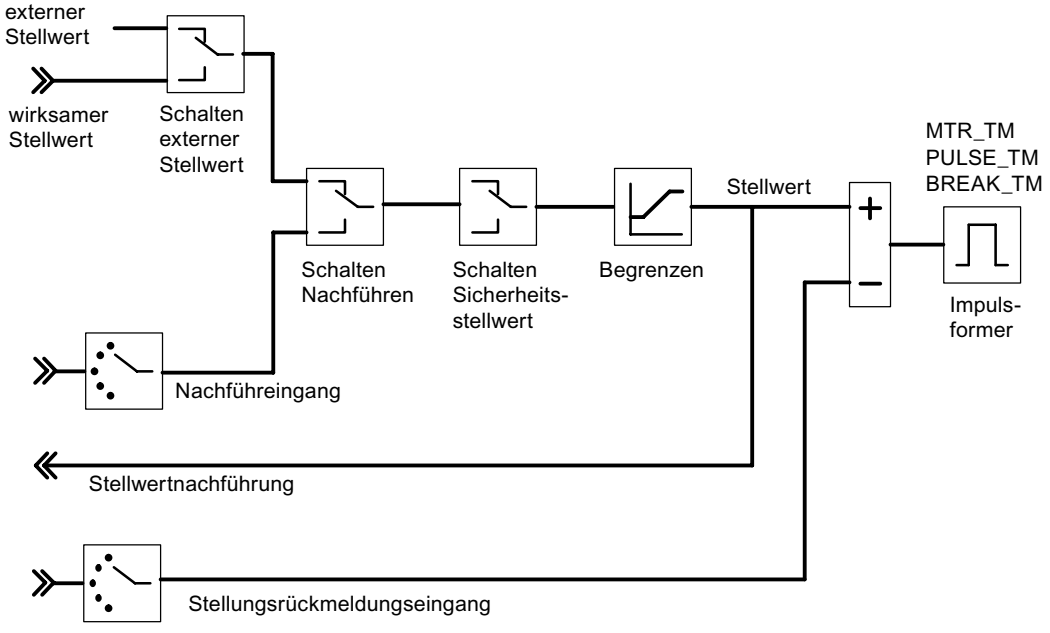

Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung)

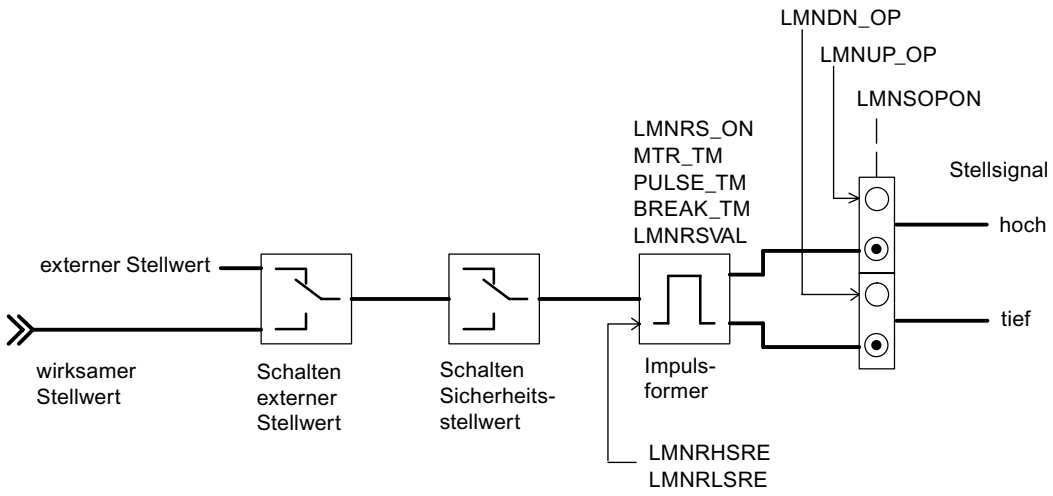

Erzeugung der Ausgangsparameter

In den folgenden Abbildungen wird dargestellt, an welchen Stellen in der Baugruppe die Ausgangsparameter der Anweisung PID_FM_455 erzeugt werden.

Regeldifferenzbildung beim Festwert- oder Kaskadenregler

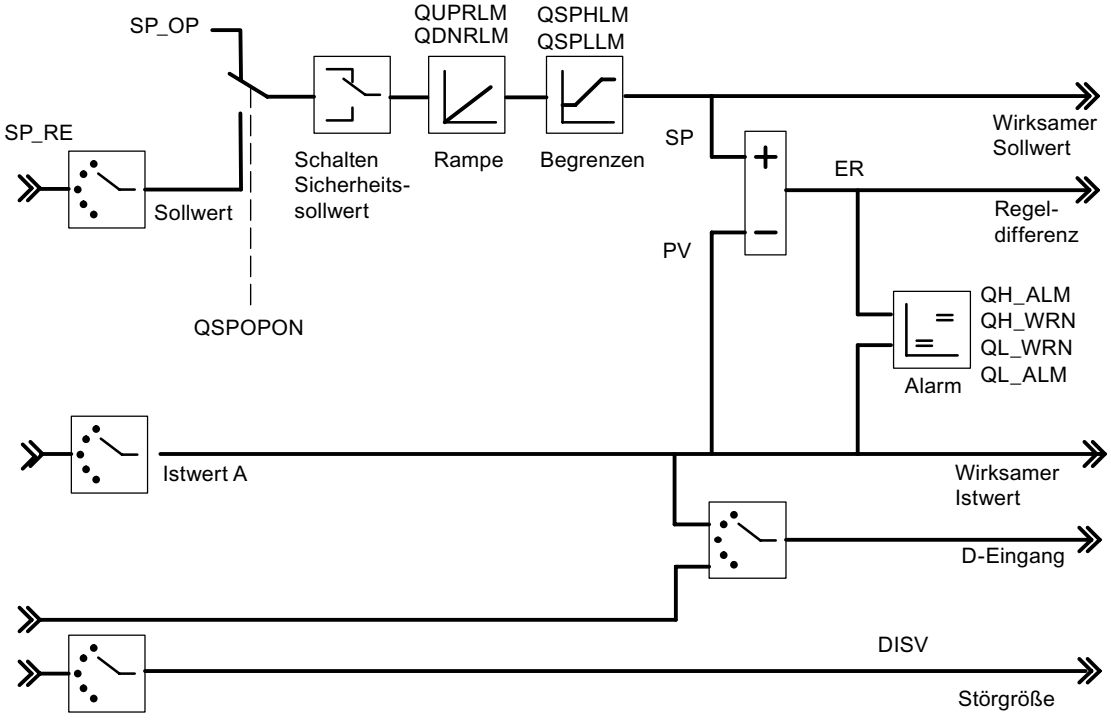

Blockschaltbild des Regelalgorithmus

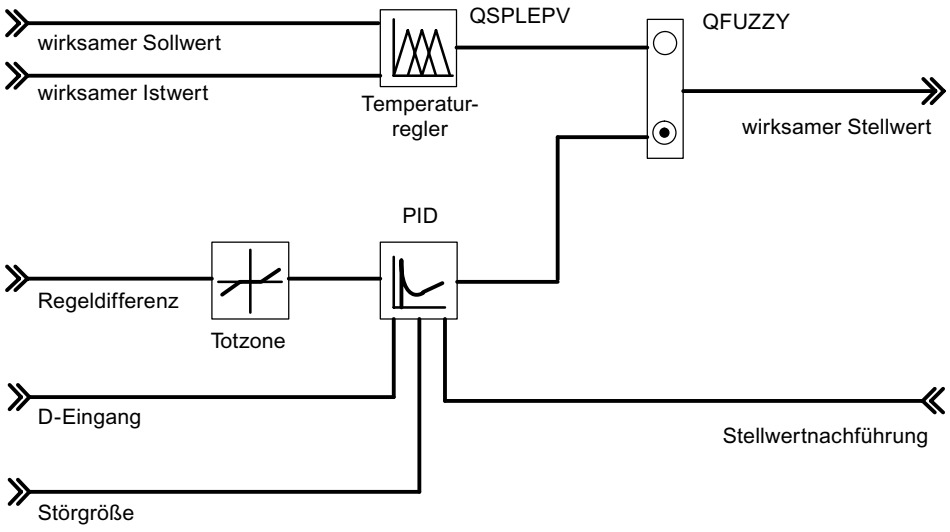

Reglerausgang des K-Reglers

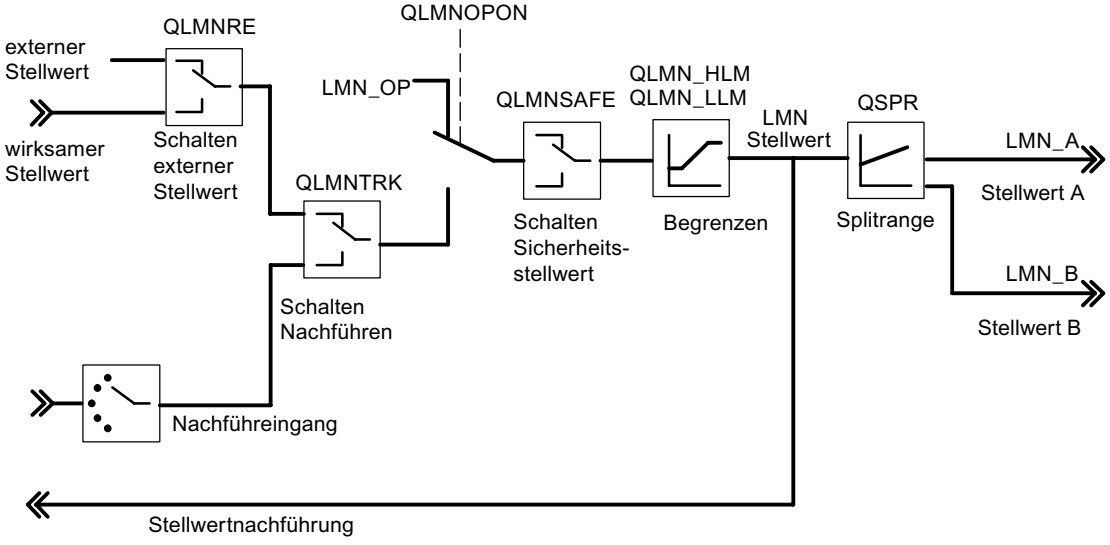

Reglerausgang des S-Reglers (Betriebsart Impulsregler)

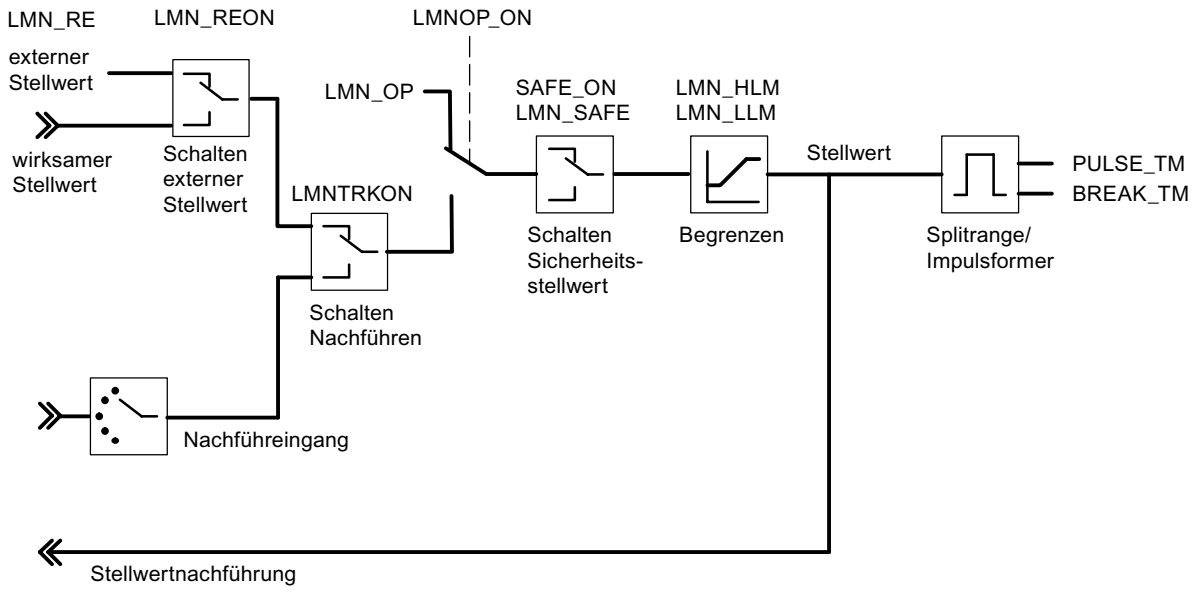

Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung)

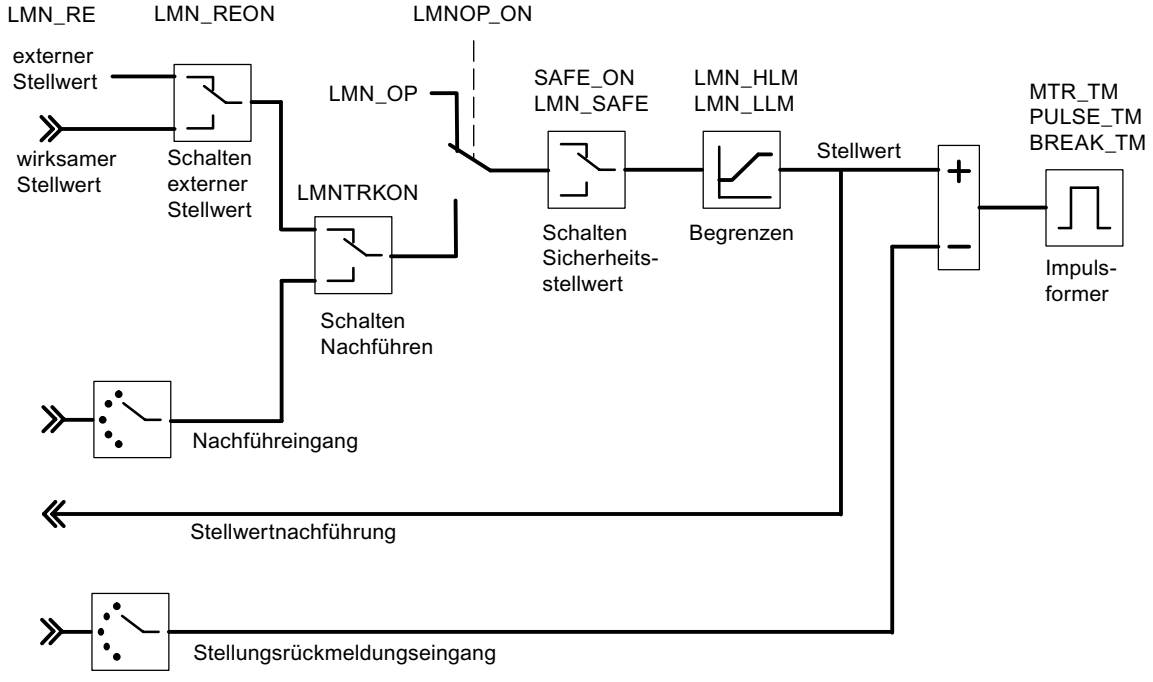

Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung)

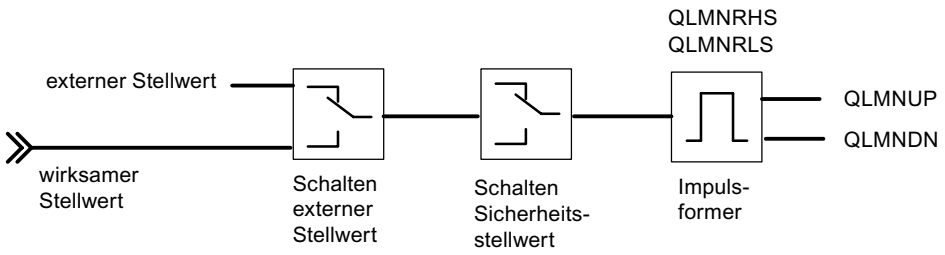

FUZ_455

Beschreibung FUZ_455

Die Anweisung FUZ_455 dient dazu, die Reglerparameter des Fuzzy-Temperaturreglers aus der FM 455 auszulesen und zu schreiben.

Diese Funktion eignet sich für folgende Anwendungsfälle:

- Übertragen der durch Identifikation ermittelten Reglerparameter zur FM 455 nach Baugruppentausch
- Anpassen der FM 455 an unterschiedliche Regelstrecken

Hinweis

Die durch eine Identifikation von der FM 455 ermittelten Parameter dürfen Sie nicht ändern, da sie für die Regelstrecke optimiert sind.

Arbeitsweise

Mit den Parametern READ_PAR und LOAD_PAR steuern Sie die Arbeitsweise der Anweisung FUZ_455.

Wenn READ_PAR = TRUE, liest FUZ_455 die Parameter aller Temperaturrelger der FM 455 und legt sie im Instanz-DB ab. Nach erfolgreichem Auslesen der Temperaturreglerparameter setzt FUZ_455 den Parameter READ_PAR auf FALSE.

Wenn LOAD_PAR = TRUE, schreibt FUZ_455 die Parameter aller Temperaturregler der FM 455 aus dem Instanz-DB zur FM 455. Nach erfolgreicher Übertragung der Parameter setzt FUZ_455 den Parameter LOAD_PAR auf FALSE.

Aufruf

FUZ_455 benötigt keinen Initialisierungslauf.

FUZ_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Anlauf

Im Anlauf der CPU sollten Sie dem Parameter LOAD_PAR der Anweisung FUZ_455 setzen und danach den Baustein im zyklischen Programm bedingt aufrufen, solange LOAD_PAR = TRUE ist.

Eingangsparameter FUZ_455

Tabelle 4-271

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	512	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter FUZ_455

Tabelle 4-272

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	2.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
PARAFFUZ	4.0	WORD	0	Am Parameter PARAFFUZ wird ein durch die Anweisung FUZ_455 erzeugter Parametrierfehler wie folgt angezeigt: High Byte von PARAFFUZ = 01: es liegt ein Parametrierfehler vor. High Byte von PARAFFUZ = 00: es liegt kein Parametrierfehler vor. Im Low-Byte steht der Offset des Parameters, der den Parametrierfehler verursacht hat, gerechnet ab der statischen Variablen FUZ_PAR[1]. PARAFFUZ = W#16#0104 bedeutet z. B., dass der zweite Parameter fehlerhaft ist

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

Durchgangsparameter FUZ_455

Tabelle 4-273

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
READ_PAR	6.0	BOOL	FALSE	Wenn der Parameter READ_PAR gesetzt wird, dann werden die Fuzzy-Parameter aus der Baugruppe ausgelesen und in statischen Variablen des Instanz-DBs abgelegt.
LOAD_PAR	6.1	BOOL	FALSE	Wenn der Parameter LOAD_PAR gesetzt wird, dann werden die Fuzzy-Parameter aus statischen Variablen des Instanz-DBs ausgelesen und zur Baugruppe übertragen.

FORCE455

Beschreibung FORCE455

Die Anweisung FORCE455 dient zur Simulation (Forcen) der Analog- und Digitaleingangswerte zur Unterstützung der Inbetriebsetzung.

Simulieren der Digitalwerte

Die Simulation der Werte für die Digitaleingänge 1 bis 16 wird eingeschaltet über die Schalter S_DION[i], wobei 1 ≤ i ≤ 16 ist. Die Simulationswerte geben Sie über die Parameter DI_SIM[i] vor. Wenn S_DION[i] = TRUE, wird statt des Werts am Digitaleingang i der Baugruppe der Wert DI_SIM[i] verwendet. Die LEDs I1 bis I16 zeigen auch bei Simulation immer den Zustand des zugehörigen Digitaleingangs an.

Simulieren der Analogwerte

Die Simulation der Analogwerte für die Kanäle 1 bis 16 wird eingeschaltet über die Schalter S_AION[i] bzw. S_PVON[i], wobei 1 ≤ i ≤ 16 ist.

Die Simulationswerte für die Kanäle 1 bis 16 geben Sie über die Parameter PV_SIM[i] vor.

Sie können die Simulationswerte an zwei Stellen wirksam werden lassen:

- S_AION[i] = TRUE (1 ≤ i ≤ 16)
  An Stelle des Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.
- S_PVON[i] = TRUE (1 ≤ i ≤ 16)
  An Stelle des aufbereiteten Werts vom Analogeingang i der Baugruppe wird der Wert PV_SIM[i] verwendet.

In der nachstehenden Abbildung ist dargestellt, an welcher Stelle der simulierte Analogwert wirksam wird.

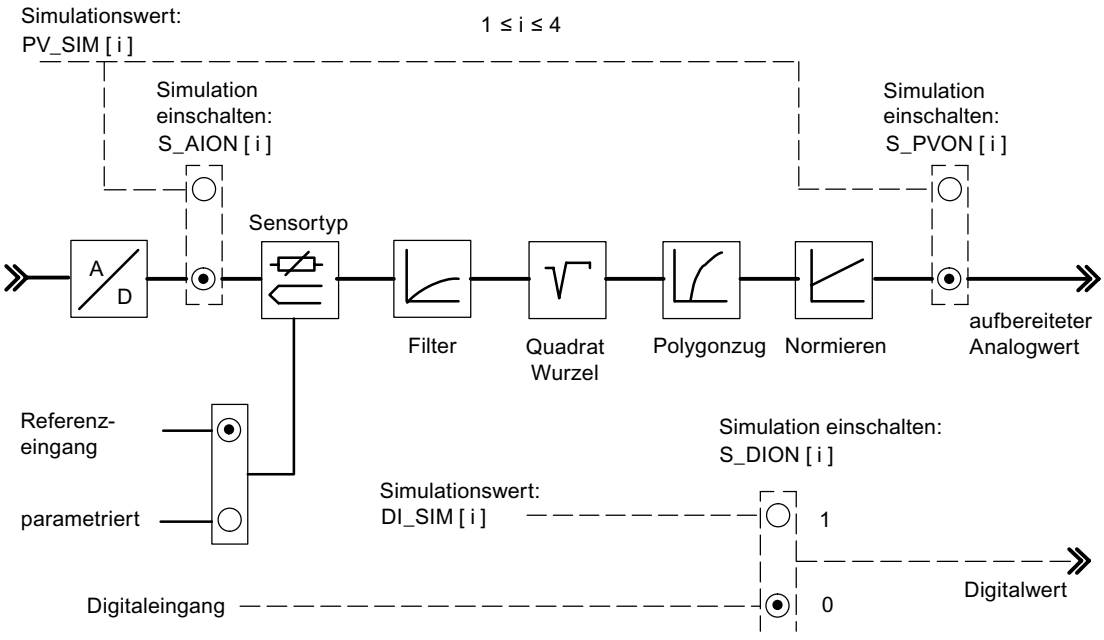

Das Einschalten und die Vorgabe der Simulationswerte (Forcen) erfolgen nicht über die Parametrieroberfläche. Daher sind die entsprechenden Schalter und Verbindungslinien gestrichelt gezeichnet.

Aufruf

FORCE455 benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

FORCE455 muss im selben OB aufgerufen werden, wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Anlaufverhalten

Bei Wiederanlauf der FM 455 nach Netz-Aus stehen die Simulationsschalter auf der FM 455 wieder auf FALSE.

Eingangsparameter FORCE455

Tabelle 4-274

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
S_AION	0.0	ARRAY [1..16] of BOOL	FALSE	Wenn z. B. der Schalter S_AION[1] auf TRUE gesetzt wird, dann wird an Stelle des Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.
S_PVON	2.0	ARRAY [1..16] of BOOL	FALSE	Wenn z. B. der Schalter S_PVON[1] auf TRUE gesetzt wird, dann wird an Stelle des aufbereiteten Analogeingangswerts 1 der Baugruppe der Wert PV_SIM[1] verwendet.
PV_SIM	4.0	ARRAY [1..16] of REAL	0.0	Am Eingang PV_SIM[1] wird z. B. der Simulationswert für den Analogeingang 1 vorgegeben. Falls S_PVON = TRUE ist, dann wird hier der aufbereitete Analogeingangswert vorgegeben. Falls S_PVON = FALSE und S_AION = TRUE sind, dann wird hier der Analogeingangswert in mA bzw. mV vorgegeben, der noch durch die Aufbereitungsfunktionen in einen aufbereiteten Wert gewandelt wird. Zulässig sind 0.0 bis 20.0 [mA], -1500 bis +10000 [mV] oder ein technischer Wertebereich.
S_DION	68.0	ARRAY [1..16] of BOOL	FALSE	Wenn z. B. S_DION[1] auf TRUE gesetzt wird, dann wird als Digitalwert an Stelle des Digitaleingangs 1 der Baugruppe der Wert DI_SIM[1] verwendet.
DI_SIM	70.0	ARRAY [1..16] of BOOL	FALSE	Simulationswert für Digitaleingang
MOD_ADDR	72.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter FORCE455

Tabelle 4-275

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	74.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

READ_455

Beschreibung READ_455

Die Anweisung READ_455 dient zum Auslesen der Digital- und Analogeingangswerte zur Unterstützung der Inbetriebsetzung.

Folgende Werte werden angezeigt.

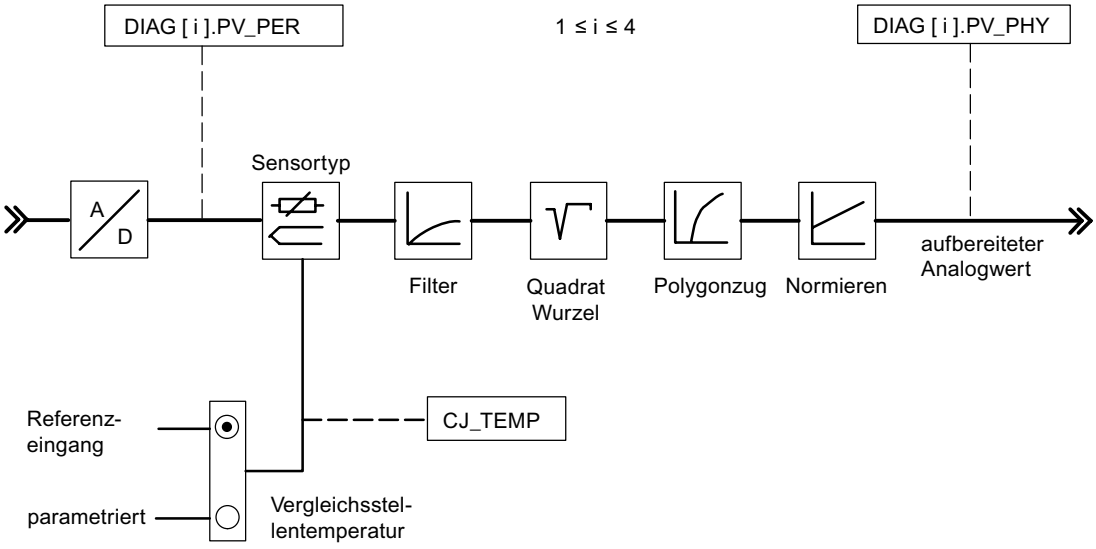

Aufruf

READ_455 benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

READ_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Eingangsparameter READ_455

Tabelle 4-276

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	512	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.

Ausgangsparameter READ_455

Tabelle 4-277

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
CJ_TEMP	2.0	REAL	0.0	Am Ausgang CJ_TEMP wird die von der Baugruppe gemessene Vergleichsstellentemperatur angezeigt, falls ein Thermoelementeingang parametriert ist und die Vergleichsstellentemperatur nicht parametriert vorgegeben wird.
STAT_DI	6.0	ARRAY [1..16] of BOOL	FALSE	An den Parametern STAT_DI werden die Zustände der Digitaleingänge 1 bis 16 angezeigt.
DIAG[x].PV_PER	(Kanalnummer) x 8	ARRAY [1..16] of STRUCT	0.0	Am Parameter DIAG[1].PV_PER wird z. B. der Analogeingangswert der Baugruppe in der Einheit mA bzw. mV angezeigt.
DIAG[x].PV_PHY	(Kanalnummer) x 8 + 4	ARRAY [1..16] of STRUCT	0.0	Am Parameter DIAG[1].PV_PHY wird z. B. der aufbereitete Analogeingangswert der Baugruppe in physikalischer Einheit angezeigt.
RET_VALU	136.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

CH_DIAG_455

Beschreibung CH_DIAG_455

Die Anweisung CH_DIAG_455 dient dazu, zusätzliche kanalspezifische Diagnosevariablen aus der Baugruppe auszulesen.

Aufruf

CH_DIAG_455 benötigt keinen Initialisierungslauf und wird normalerweise zyklisch aufgerufen.

CH_DIAG_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Eingangsparameter CH_DIAG_455

Tabelle 4-278

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	2.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässige Werte sind 1 - 16

Ausgangsparameter CH_DIAG_455

Tabelle 4-279

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP_R	4.0	REAL	0.0	Der Parameter ist bei eingestelltem Verhältnisregler mit dem Eingangswert des Sollwertes belegt.
PV_R	8.0	REAL	0.0	Der Parameter ist nur bei eingestelltem Verhältnisregler mit folgendem Wert belegt: (Istwert A - Sollwert Offset) / Istwert D
DIF_I	12.0	REAL	0.0	Am Parameter DIF_I wird die Eingangsgröße des D-Anteils angezeigt. Dies ist z. B. von Interesse, wenn als Eingangsgröße des D-Anteils ein Analogeingang parametriert ist.
TRACKPER	16.0	REAL	0.0	Am Parameter TRACKPER wird die Eingangsgröße angezeigt, auf die die Stellgröße nachgeführt wird, wenn der Regler auf Stellgrößennachführung geschaltet wird.
IDSTATUS	20.0	WORD	0.0	Am AUTOHOTSPOT wird der Zustand der Identifikation angezeigt
LMN_P	22.0	REAL	0.0	Am Parameter LMN_P wird der P-Anteil der Stellgröße angezeigt.
LMN_I	26.0	REAL	0.0	Am Parameter LMN_I wird der I-Anteil der Stellgröße angezeigt.
LMN_D	30.0	REAL	0.0	Am Parameter LMN_D wird der D-Anteil der Stellgröße angezeigt.
RET_VALU	34.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

Parameter IDSTATUS

Der Parameter IDSTATUS liefert Informationen über den Zustand der Identifikation und enthält die vier Hexadezimalwerte X, A, I und F.

X				A				I				F			
2^{15}															2^{0}

Wert	Beschreibung
X	ohne Bedeutung (immer 0)
A	Aktionsnummer • 0 = Handbetrieb (bzw. kein Regelbetrieb); • 2 = Regelung; • 4 = Optimierung eingeschaltet (FUZID_ON = true); • 6 = Übergangszustand vom Handbetrieb nach 2 oder 4;
I	Anzeige "Identifikation läuft" und "Parameter ermittelt, aber noch nicht im EEPROM abgelegt" • 0 = Identifikation läuft nicht, keine neuen Parameter ermittelt • 1 = Identifikation läuft, keine neuen Parameter ermittelt • 2 = Identifikation läuft nicht, neue Parameter ermittelt, aber noch nicht im EEPROM abgelegt • 3 = Identifikation läuft, neue Parameter ermittelt, aber noch nicht im EEPROM abgelegt
F	Fehlernummer • 0 = kein Fehler • 4 = Zu großer Sprung des Istwertes während der Identifikation • 5 = Verhältnis von Verzugszeit zu Streckenzeitkonstante zu groß oder stark nichtlineares Verhalten der Regelstrecke. • 6 = Temperaturgefälle oder -steigung bei Start der Identifikation zu groß. Strecke nicht ausreichend beruhigt

Blockschaltbilder CH_DIAG_455

Blockschaltbilder CH_DIAG_455

Folgende Diagnosewerte der Regeldifferenz werden angezeigt.

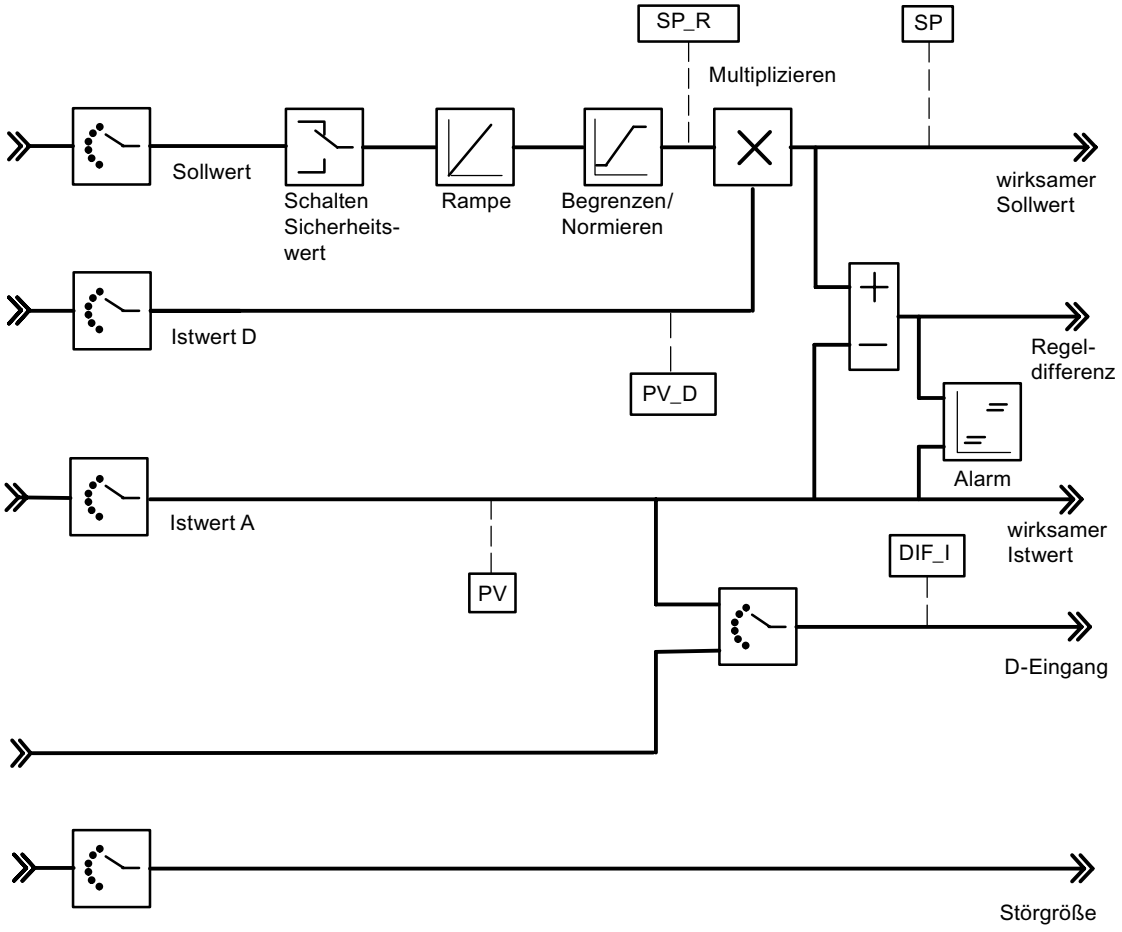

Folgende Werte des Regelalgorithmus werden angezeigt.

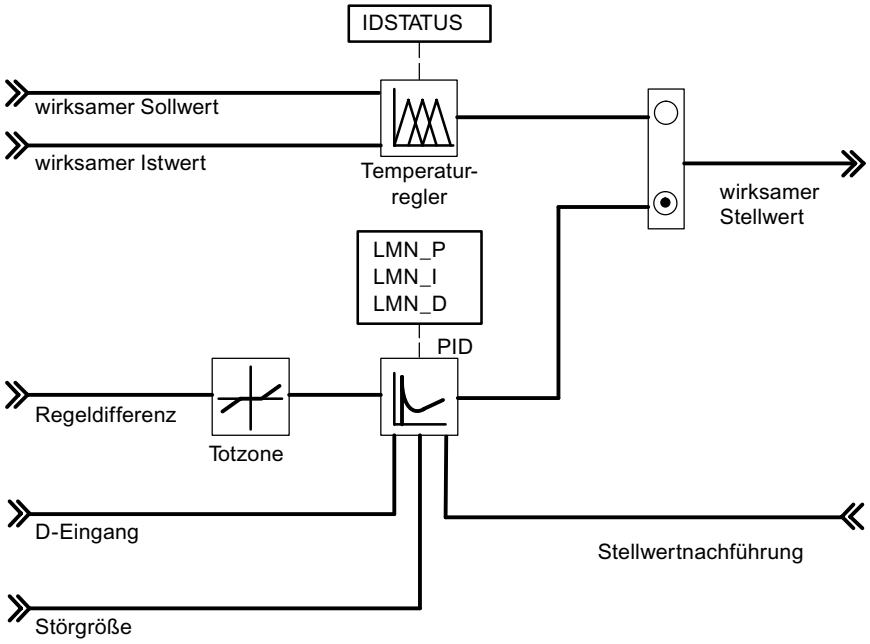

Folgende Werte des K- oder S-Reglers werden angezeigt.

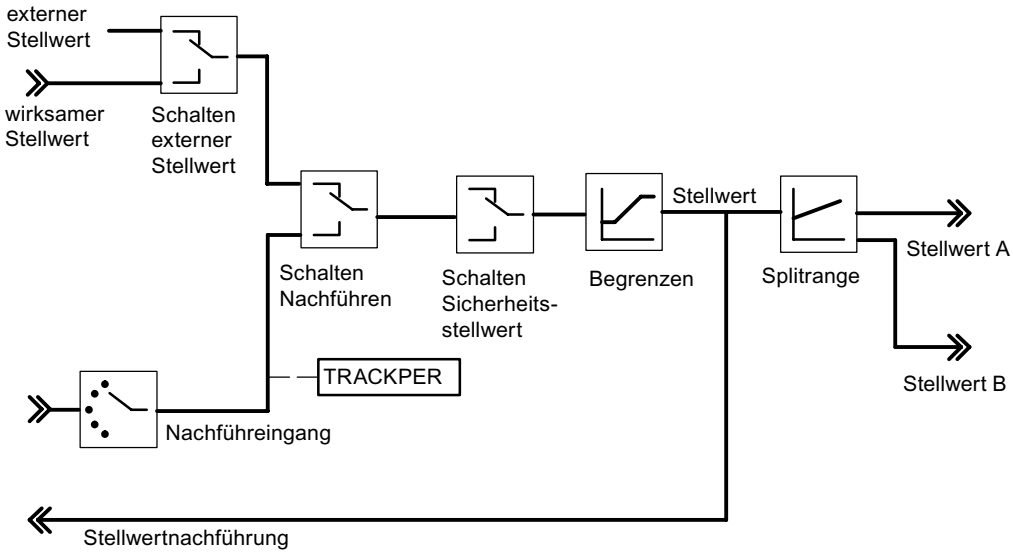

PID_PAR_455

Beschreibung PID_PAR_455

Die Anweisung PID_PAR_455 dient zum Online-Ändern von Parametern, die nicht über die Anweisung PID_FM_455 vorgegeben werden können. Welche Parameter Sie ändern können entnehmen Sie der Tabelle änderbare Parameter.

Aufruf

PID_PAR_455 benötigt einen Initialisierungslauf, der durch Setzen des Parameters COM_RST = TRUE veranlasst wird. Um Laufzeit zu sparen, sollte PID_PAR_455 nicht zyklisch aufgerufen werden sondern nur dann, wenn Parameter geändert werden sollen. COM_RST muss dann FALSE sein.

PID_PAR_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Arbeitsweise

Im Initialisierungslauf liest PID_PAR_455 die Parameter aus den Systemdaten und legt sie in statischen Variablen ab. PID_PAR_455 ändert pro Aufruf eine dieser REAL-Variablen und eine dieser INT-Variablen. Am Eingangsparameter INDEX_R bzw. INDEX_I geben Sie die Indexnummer des Parameters an, den Sie ändern wollen. Am Eingangsparameter VALUE_R bzw. VALUE_I geben Sie die neuen Werte an. PID_PAR_455 überträgt den kompletten Datensatz mit den geänderten Variablen an die FM 455.

Um weitere Parameter zu ändern, müssen Sie **denselben** Instanz-DB mehrfach nacheinander mit COM_RST = FALSE und mit unterschiedlichen Indexnummern aufrufen.

Der Parameter COM_RST ist ein Eingangsparameter, der von PID_PAR_455 nicht zurückgesetzt wird.

Hinweis

Bitte beachten Sie, dass Parameter, die Sie mit PID_PAR_455 geändert haben, beim Anlauf mit den Parametern aus den Systemdaten überschrieben werden.

Anlauf

Beim Anlauf der CPU und beim STOP-RUN-Übergang der CPU werden die Parameter an der FM 455 mit den Parametern aus dem SDB der CPU überschrieben.

Beispiel

Sie wollen im Betrieb die Hochlaufzeit der Rampe für die Führungsgröße ändern sowie abhängig vom Prozesszustand als Istwert verschiedene Analogeingangswerte verwenden.

- Rufen Sie im Anlauf der CPU PID_PAR_455 auf mit COM_RST = TRUE.
- Zum Parametrieren der Hochlaufzeit der Rampe für die Führungsgröße auf 10.0 rufen Sie im laufenden Betrieb PID_PAR_455 auf mit INDEX_R = 30, VALUE_R = 10.0 und INDEX_I = 0.
- Wenn Sie als Istwert den Analogeingangswert 4 der Baugruppe parametrieren wollen, rufen Sie im laufenden Betrieb PID_PAR_455 auf mit INDEX_R = 0, INDEX_I = 50 und VALUE_I = 4.

Eingangsparameter PID_PAR_455

Tabelle 4-280

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL	TRUE	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung PID_PAR_455 einen Initialisierungslauf durch. Dabei werden die Parameter aus den Systemdaten der CPU gelesen und im Instanz-DB abgelegt.
MOD_ADDR	2.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CHANNEL	4.0	INT	1	Am Eingang "Kanalnummer" wird die Nummer des Reglerkanals parametriert, auf den sich der Instanz-DB bezieht. Zulässig sind die Werte 1 - 16
INDEX_R	6.0	INT	0.0	• 0: kein Real-Parameter wird geändert • 1 bis 48: Jeder Wert entspricht einem änderbaren REAL-Parameter.
VALUE_R	8.0	REAL	0.0	abhängig vom jeweiligen Parameter
INDEX_I	12.0	INT	0.0	• 0: kein INT-Parameter wird geändert. • 49 bis 61: Jeder Wert entspricht einem änderbaren INT_Parameter.
VALUE_I	14.0	INT	0.0	abhängig vom jeweiligen Parameter

Ausgangsparameter PID_PAR_455

Tabelle 4-281

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	16.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
BUSY	18.0	BOOL	FALSE	Wenn BUSY = TRUE, dann sind die Parameter noch nicht von der Baugruppe übernommen worden. Der Aufruf der Anweisung PID_PAR_455 sollte dann im nächsten Zyklus wiederholt werden.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

Änderbare Parameter

Tabelle 4-282 Liste der mit PID_PAR_455 zu ändernden REAL- und INT-Parameter auf einer FM 455

Datentyp	Beschreibung	Indexnummer
-	Kein Parameter angewählt	0
REAL	Filterzeitkonstante für Analogeingang	1
REAL	Messende (100%)	2
REAL	Messanfang (0%)	3
REAL	Polygonzug, Stützwert 1 Eingangsseite	4
REAL	Polygonzug, Stützwert 2 Eingangsseite	5
REAL	Polygonzug, Stützwert 3 Eingangsseite	6
REAL	Polygonzug, Stützwert 4 Eingangsseite	7
REAL	Polygonzug, Stützwert 5 Eingangsseite	8
REAL	Polygonzug, Stützwert 6 Eingangsseite	9
REAL	Polygonzug, Stützwert 7 Eingangsseite	10
REAL	Polygonzug, Stützwert 8 Eingangsseite	11
REAL	Polygonzug, Stützwert 9 Eingangsseite	12
REAL	Polygonzug, Stützwert 10 Eingangsseite	13
REAL	Polygonzug, Stützwert 11 Eingangsseite	14
REAL	Polygonzug, Stützwert 12 Eingangsseite	15
REAL	Polygonzug, Stützwert 13 Eingangsseite	16
REAL	Polygonzug, Stützwert 1 Ausgangsseite	17
REAL	Polygonzug, Stützwert 2 Ausgangsseite	18
REAL	Polygonzug, Stützwert 3 Ausgangsseite	19
REAL	Polygonzug, Stützwert 4 Ausgangsseite	20
REAL	Polygonzug, Stützwert 5 Ausgangsseite	21
REAL	Polygonzug, Stützwert 6 Ausgangsseite	22
REAL	Polygonzug, Stützwert 7 Ausgangsseite	23
REAL	Polygonzug, Stützwert 8 Ausgangsseite	24
REAL	Polygonzug, Stützwert 9 Ausgangsseite	25
REAL	Polygonzug, Stützwert 10 Ausgangsseite	26
REAL	Polygonzug, Stützwert 11 Ausgangsseite	27
REAL	Polygonzug, Stützwert 12 Ausgangsseite	28
REAL	Polygonzug, Stützwert 13 Ausgangsseite	29
REAL	Hochlaufzeit der Rampe für Führungsgröße	30
REAL	Sicherheits-Führungsgröße oder Sicherheits-Führungsgrößen-Verhältnis	31
REAL	Offset für Sollwertverknüpfung (Verhältnis-/Mischungsregler)	32
REAL	Faktor für Istwert B (Dreikomponentenregler)	33
REAL	Faktor für Istwert C (Dreikomponentenregler)	34
REAL	Offset für Istwertverknüpfung (Dreikomponentenregler)	35
REAL	Faktor für Störgrößenverknüpfung	36
REAL	Arbeitspunkt	37

Datentyp	Beschreibung		Indexnummer
REAL	Aggressivität beim Fuzzy-Regler		38
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal A		39
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal A		40
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal A		41
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal A		42
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Eingangssignal B		43
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Eingangssignal B		44
REAL	Eckpunkte für Splitrangefunktion: Bereichsanfang Ausgangssignal B		45
REAL	Eckpunkte für Splitrangefunktion: Bereichsende Ausgangssignal B		46
REAL	Mindestimpulsdauer		47
REAL	Mindestpausendauer		48
INT	Auswahl der Führungsgröße SP bzw. SP_RE für den Regler		49
	0	Sollwert SP_RE der Anweisung	
	1 bis 16	Analogeingangswert 1 bis 16	
	17 bis 32	Stellgröße (LMN) von Regler 1 bis 16	
INT	Auswahl der Haupt-Regelgröße Istwert A für den Regler		50
	0:	Istwert A = 0.0	
	1 bis 16:	Analogeingangswert 1 bis 16	
INT	Auswahl der Hilfs-Regelgröße Istwert B für den Regler		51
	0	Istwert B = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
INT	Auswahl der Hilfs-Regelgröße Istwert C für den Regler		52
	0	Istwert C = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
INT	Auswahl der Hilfs-Regelgröße Istwert D für den Regler		53
	0	Istwert D = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
	17 bis 32	Stellgröße (LMN) von Regler 1 bis 16	
INT	Auswahl der Störgröße DISV für den Regler		54
	0	Störgröße = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
INT	Auswahl der Stellungsnachführung TRACK_PER für den Regler		55
	0	Stellungsnachführung = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
INT	Auswahl der Stellungsnachführung LMNR_PER für den Regler		56
	0	Stellungsnachführung = 0.0	
	1 bis 16	Analogeingangswert 1 bis 16	
INT	Auswahl des Signals zur Umschaltung auf Sicherheitswert für den Stellwert des Reglers		57
	0	Nur Vorgabe über Parameter SAFE_ON der Anweisung PID_FM_455	
	1 bis 16	Vorgabe über Parameter SAFE_ON der Anweisung PID_FM_455 verodert mit Digitaleingang 1 bis 16	

Datentyp	Beschreibung		Indexnummer
INT	Auswahl des Signals zur Umschaltung auf Nachführfunktion des Stellwertes des Reglers		58
	0	Nur Vorgabe über Parameter LMNTRKON der Anweisung PID_FM_455	
	1 bis 16	Vorgabe über Parameter LMNTRKON der Anweisung PID_FM_455 verodert mit Digitaleingang 1 bis 16	
INT	Auswahl des Signals zur Umschaltung des Stellwertes des Reglers auf LMN_RE		59
	0	Nur Vorgabe über Parameter LMN_REON der Anweisung PID_FM_455	
	1 bis 16	Vorgabe über Parameter LMN_REON der Anweisung PID_FM_455 verodert mit Digitaleingang 1 bis 16	
INT	Auswahl des oberen Anschlagsignals der Stellungsrückmeldung		60
	0	Nur Vorgabe über Parameter LMNRHSRE der Anweisung PID_FM_455	
	1 bis 16	Vorgabe über Parameter LMNRHSRE der Anweisung PID_FM_455 verodert mit Digitaleingang 1 bis 16	
INT	Auswahl des unteren Anschlagsignals der Stellungsrückmeldung		61
	0	Nur Vorgabe über Parameter LMNRLSRE der Anweisung PID_FM_455	
	1 bis 16	Vorgabe über Parameter LMNRLSRE der Anweisung PID_FM_455 verodert mit Digitaleingang 1 bis 16	

CJ_T_PAR_455

Beschreibung CJ_T_PAR_455

Die Anweisung CJ_T_PAR_455 dient zum Online-Ändern der parametrierten Vergleichsstellentemperatur. Dies ist erforderlich, wenn eine Temperaturregelung mit mehreren FM 455 mit Thermoelementeingängen betrieben werden soll, ohne dass an jede FM 455 ein Pt 100 angeschlossen werden muss.

Aufruf

CJ_T_PAR_455 benötigt einen Initialisierungslauf. Er muss hierzu einmalig im Anlauf der CPU mit Parameter COM_RST = TRUE aufgerufen werden.

CJ_T_PAR_455 wird normalerweise zyklisch aufgerufen. COM_RST sollte dabei aus Laufzeitgründen FALSE sein. Der Parameter COM_RST ist ein Eingangsparameter, der nicht zurückgesetzt wird.

CJ_T_PAR_455 muss im selben OB aufgerufen werden wie alle anderen Anweisungen, die auf dieselbe FM 455 zugreifen.

Beispiel

Wenn bei einer Extruderregelung mit mehr als 16 Heizzonen die Vergleichsstellentemperatur mit einer FM 455 gemessen wird, dann kann diese über READ_455 am Parameter CJ_TEMP ausgelesen werden und bei den anderen FM 455 über CJ_T_PAR_455 parametriert werden.

Eingangsparameter CJ_T_PAR_455

Tabelle 4-283

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	0.0	BOOL	–	Wenn der Parameter COM_RST = TRUE ist, dann führt die Anweisung CJ_T_PAR_455 einen Initialisierungslauf durch. Dabei werden die Parameter aus den Systemdaten der CPU gelesen und im Instanz-DB abgelegt.
MOD_ADDR	2.0	INT	256	An diesem Eingang steht die Baugruppenadresse, die sich aus der Konfigurierung mit STEP 7 ergibt.
CJ_T	4.0	REAL	0.0	Am Parameter CJ_T kann die Vergleichsstellentemperatur vorgegeben werden. Der zulässige Wertebereich ist abhängig vom Sensortyp.

Ausgangsparameter CJ_T_PAR_455

Tabelle 4-284

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
RET_VALU	8.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen WR_REC und RD_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.
BUSY	10.0	BOOL	FALSE	Wenn BUSY = TRUE, dann sind die Parameter noch nicht von der Baugruppe übernommen worden. Der Aufruf der Anweisung CJ_T_PAR_455 sollte dann im nächsten Zyklus wiederholt werden.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

LP_ZONE

Beschreibung LP_ZONE

Zur gleichmäßigen Auslastung der CPU und zur äquidistanten Bearbeitung der Anweisung ZONE bei einer Regleroptimierung ist es notwendig, den Regleraufrufverteiler LP_ZONE zu benutzen.

LP_ZONE sorgt dafür, dass immer nur eine Zone pro Aufruf des Weckalarm-OBs bearbeitet wird.

Die Abtastzeiten der Zonen werden von der Anweisung LP_ZONE berechnet und an die Instanzen der Anweisung ZONE weitergereicht.

Aufruf

LP_ZONE wird in einem Anlauf- oder Weckalarm-OB aufgerufen. Die Anweisungen LP_ZONE, ADM_ZONE und ZONE müssen alle in der gleichen Weckalarmebene aufgerufen werden, am besten sofort im Weckalarm-OB. Es dürfen keine eigenen Zeituntersetzungen für die Bausteine programmiert werden.

Nach dem Einfügen, Entfernen einer Zone oder bei einer Änderung der Aufrufreihenfolge benötigt LP_ZONE einen Initialisierungslauf. Er muss hierzu mit Parameter COM_RST = TRUE aufgerufen werden. Das COM_RST-Bit wird über die Schnittstelle LP_SCH an die einzelnen Zonen und deren Verwaltungsbaustein weitergegeben.

Die Zonennummern NBR_CALL der Instanzen der Anweisungen ZONE und ADM_ZONE und die maximale Zonennummer MNR_CALL der Anweisung LP_ZONE werden nach folgendem Mechanismus automatisch gesetzt:

Erster Durchlauf des Weckalarm-OBs nach Setzen von COM_RST der Anweisung LP_ZONE:

- LP_ZONE setzt den Aufrufzähler LP_SCH. CALL_CNT auf 1.
- ZONE und ADM_ZONE lesen den Aufrufzähler und schreiben den Wert auf ihren Durchgangsparameter NBR_CALL. ZONE erhöht den Aufrufzähler um 1. ADM_ZONE lässt ihn unverändert.

Zweiter Durchlauf des Weckalarm-OBs nach Setzen von COM_RST der Anweisung LP_ZONE:

- LP_ZONE liest den Aufrufzähler (Anzahl aller Zonen) und schreibt den Wert auf MNR_CALL.

Arbeitsweise

Mit jedem Aufruf der Anweisung LP_ZONE wird der Parameter LP_SCH.CALL_CNT um 1 hochgezählt. Wenn der Zähler die maximale Zonenzahl von MNR_CALL erreicht hat, wird er wieder auf 1 gesetzt. Beim Aufruf der Anweisung ZONE prüft diese zuerst den Zähler LP_SCH.CALL_CNT. Stimmt er überein mit seiner eigenen Zonennummer NBR_CALL, dann führt er sein Programm aus. Die Parameter MNR_CALL der Anweisung LP_ZONE und NBR_CALL der Anweisung ADM_ZONE und ZONE werden automatisch von den Bausteinen nach Neustart gesetzt.

Hinweis

Verschalten Sie den Durchgangsparameter LP_SCH der Anweisungen ZONE und ADM_ZONE auf die interne Struktur LP_SCH des Instanz-DBs der Anweisung LP_ZONE.

Verschalten Sie den Eingangsparameter OB_EXC_FREQ der Anweisung LP_ZONE mit dem temporären Parameter OBxx_EXC_FREQ des Weckalarm-OBs.

Beispiel

Die Aufrufreihenfolge im OB33

```
OB33
 ...
 call LP_ZONE, DB_LP_ZONE
```

```
call ADM_ZONE, DB_ADM_ZONE
call ZONE, DB_ZONE1
call ZONE, DB_ZONE2
call ZONE, DB_ZONE3
…
BE
```

führt zu folgendem zeitlichen Ablauf in der Zonenbearbeitung:

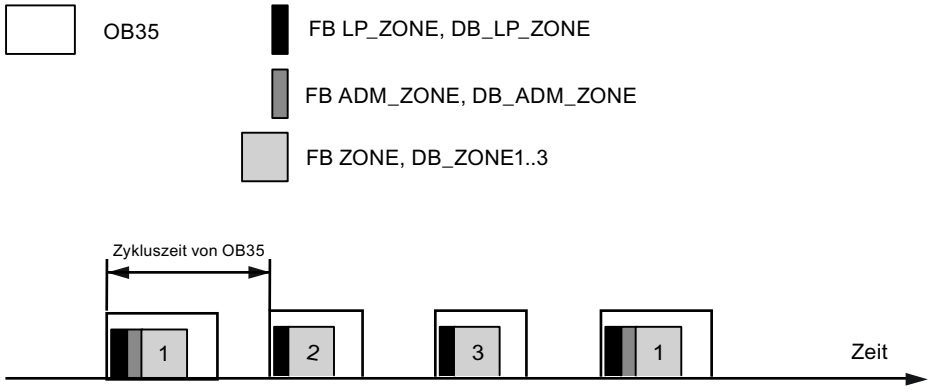

Eingangsparameter LP_ZONE

Tabelle 4-285

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
OB_EXC_FREQ	0.0	INT	0	Abtastzeit [msec]; Verschalten Sie den Parameter mit der entsprechenden lokalen Variablen des Weckalarm-OBs.

Ausgangsparameter LP_ZONE

Tabelle 4-286

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MNR_CALL	2.0	INT	0	Anzahl der Zonen. Wird nach Neustart automatisch bestimmt.

Durchgangsparameter LP_ZONE

Tabelle 4-287

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
COM_RST	4.0	BOOL	TRUE	Neustart; Sorgen Sie dafür, dass im OB100 dieses Bit gesetzt wird.

Statische Variablen LP_ZONE

Tabelle 4-288

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LP_SCH	6.0	STRUCT		Schnittstelle zwischen den Anweisungen LP_ZONE und ADM_ZONE bzw. ZONE

Danach folgen interne Parameter, die vom Anwender nicht benutzt werden dürfen.

ADM_ZONE

Beschreibung ADM_ZONE

Oft sind in der Praxis die Heiz-/Heizkühlzonen so konstruiert, dass relativ leicht Wärme von einer Zone in die andere fließen kann. In solchen Fällen müssen diese wärmegekoppelten Zonen zu einer Gruppe zusammengefasst werden. Die Anweisung ADM_ZONE verwaltet eine Gruppe von Heiz-Kühlzonen und übernimmt die folgenden Funktionen:

- Heizung aller Zonen ein- und ausschalten
- Regleroptimierung ein- und ausschalten
- Sammelanzeige eines Kanalfehlers
- Sammelanzeige eines Regleroptimierungsfehlers
- Sammelanzeige einer Regleroptimierungswarnung
- Steuerung des gemeinsamen Stellwertsprungs bei Regleroptimierung

Aufruf

Rufen Sie die Anweisungen LP_ZONE, ADM_ZONE und ZONE alle in der gleichen Weckalarmebene auf, am besten sofort im Weckalarm-OB. Es dürfen keine eigenen Zeituntersetzungen für die Bausteine programmiert werden. Rufen Sie ADM_ZONE nach LP_ZONE und vor ZONE auf.

Anlauf

Setzen Sie z.B. im OB100 oder im ersten Zyklus des Weckalarm-OBs das Neustartbit von ADM_ZONE. Die Heizungen an den einzelnen Zonen werden nach Neustart ausgeschaltet und eine laufende Regleroptimierung wird abgebrochen.

Arbeitsweise

Das eigentliche Programm wird nur bearbeitet, wenn der Aufrufzähler der Anweisung LP_ZONE den gleichen Wert besitzt wie die eigene Aufrufnummer: LP_SCH.CALL_CNT = NBR_CALL.

Heizung aller Zonen ein- und ausschalten

Eine Änderung des Schalters HEAT_OFF wird als Bedienung an die Schnittstelle ADMIN und damit nacheinander an die zu verwaltenden Zonen weitergeleitet. Wenn keine Bedienung vorliegt, wird das Ergebnis von den Zonen ADMIN.QHEAT_OFF auf den Durchgangsparameter HEAT_OFF übertragen. Nur wenn bei allen Zonen die Heizung abgeschaltet ist, wird der Sammelschalter HEAT_OFF auf TRUE gesetzt.

Eine Bedienung durch den Anwender am Schalter HEAT_OFF wird am Done-Parameter QHO_DONE angezeigt. Der Parameter steht defaultmäßig auf TRUE. Bei einer Bedienung wird er auf FALSE gesetzt. Wenn an allen zugehörigen Zonen die Heizung ein bzw. aus geschaltet ist, wird QHO_DONE wieder auf TRUE gesetzt.

Die Heizung an einer Zone ist dann ausgeschaltet, wenn gilt: cont_FM455.QLMN_RE = TRUE und cont_FM455.LMN = 0.0

Regleroptimierung aller Zonen ein- und ausschalten

Eine Änderung des Schalters TUN_ON wird als Bedienung an die Schnittstelle ADMIN und damit nacheinander an die zu verwaltenden Zonen weitergeleitet. Wenn keine Bedienung vorliegt, wird das Ergebnis von den Zonen ADMIN.QTUN_ON auf den Durchgangsparameter TUN_ON übertragen. Nur wenn bei allen Zonen die Regleroptimierung abgeschaltet ist, wird der Sammelschalter TUN_ON auf FALSE gesetzt.

Sammelanzeige eines Kanalfehlers

Das Bit QCH_F zeigt an, ob an mindestens einer Zone ein Kanalfehler der FM 455 vorliegt. Prüfen Sie an welchen Zonen ein Kanalfehler vorliegt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.

Sammelanzeige eines Regleroptimierungsfehlers

Das Bit QTUN_ERR zeigt an, ob in mindestens einem Instanz-DB der Anweisung ZONE STATUS_H = 3xxxx oder STATUS_C = 3xxxx. Prüfen Sie die entsprechenden Zonen und gehen Sie bei der Behebung vor wie bei AUTOHOTSPOT beschrieben.

Sammelanzeige einer Regleroptimierungswarnung

Das Bit QTUN_ERR zeigt an, ob in mindestens einem Instanz-DB der Anweisung ZONE STATUS_H = 2xxxx oder STATUS_C = 2xxxx. Prüfen Sie die entsprechenden Zonen und gehen Sie bei der Behebung vor wie bei AUTOHOTSPOT beschrieben.

Ablaufsteuerung bei einer Regleroptimierung

Der Administrationsbaustein sorgt dafür, dass bei der Regleroptimierung der Stellwertsprung an allen Zonen dieser Gruppe gleichzeitig erfolgt. Nach dem die einzelnen Zonen den Wendepunkt gefunden haben, gehen sie in Phase 7 über.

Hinweis

Verschalten Sie den Durchgangsparameter ADMIN der Anweisung ZONE auf die interne Struktur ADMIN des Instanz-DBs von ADM_ZONE.

ADM_ZONE wird immer in Verbindung mit LP_ZONE verwendet.

Ausgangsparameter ADM_ZONE

Tabelle 4-289

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QCH_F	0.0	BOOL	FALSE	Sammelmeldung: Kanalfehler
QTUN_ERR	0.1	BOOL	FALSE	Sammelmeldung: Fehler bei der Regleroptimierung
QTUN_WRN	0.2	BOOL	FALSE	Sammelmeldung: Warnung bei der Regleroptimierung
QHO_DONE	0.3	BOOL	TRUE	Rückmeldung einer HEAT_OFF Bedienung • 0: HEAT_OFF-Bedienung ist angestoßen • 1: HEAT_OFF-Bedienung ist an allen Zonen wirksam
NBR_CALL	2.0	INT	0	Aufrufnummer; wird nach Neustart von der Anweisung automatisch eingetragen.

Durchgangsparameter ADM_ZONE

Tabelle 4-290

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
HEAT_OFF	4.0	BOOL	TRUE	Heizung aus/ein; 1=aus
TUN_ON	4.1	BOOL	FALSE	Regleroptimierung ein/aus; 1=ein
COM_RST	4.2	BOOL	TRUE	Neustart; Sorgen Sie dafür, dass im OB100 dieses Bit gesetzt wird.
LP_SCH	6.0	STRUCT	P#P 0.0	Schnittstelle zwischen den Anweisungen LP_ZONE und ADM_ZONE bzw. ZONE

Statische Variablen ADM_ZONE

Tabelle 4-291

Parameter	Adres-se	Daten-typ	Vorbe-legung	Beschreibung
ADMIN	12.0	STRU CT		Schnittstelle zwischen den Anweisungen ADM_ZONE und ZONE

Danach folgen interne Parameter, die vom Anwender nicht benutzt werden dürfen.

ZONE

Beschreibung ZONE

Der Instanz-DB zur Anweisung ZONE enthält die Reglerdaten einer einzelnen Heiz- bzw. Heizkühlzone. Die Anweisung ZONE liest und schreibt Daten auf einen Kanal der FM 455.

Die Reglerfunktionen der FM 455 werden durch folgende Funktionen ergänzt:

- Regelzonenbetrieb
- Verhältnisfaktor für Heizkühlzonen
- Regleroptimierung

ZONE eignet sich dadurch besonders zum Regeln von Heiz- bzw. Heizkühlzonen eines Extruders.

Voraussetzung

Der Kanal der FM 455 ist als Impulsregler oder als kontinuierlicher Regler parametriert.

Die Anweisungen PID_FM_455 und PID_PAR_455 sind in das Anwenderprogramm kopiert.

Aufruf

Rufen Sie die Anweisungen LP_ZONE, ADM_ZONE und ZONE alle in der gleichen Weckalarmebene auf, am besten sofort im Weckalarm-OB. Es dürfen keine eigenen Zeituntersetzungen für die Bausteine programmiert werden. Wichtig ist, dass immer die zusammengehörigen Zonen in einem Block nach dem Aufruf der Anweisung ADM_ZONE stehen.

Anlauf

Setzen Sie im OB100 das Neustartbit von ZONE. Bei Anwendung von LP_ZONE geschieht dies automatisch.

Folgende Aktionen werden bei Anstoß der Initialisierungsroutine COM_RST=TRUE

durchgeführt:

- cont_FM455.SDB_SEL = TRUE: SDB-Reglerparameter cont_FM455.P_SEL...cont_FM455.BREAK_TM werden von der FM 455 nicht übernommen. Die in der FM wirksamen Parameter bleiben weiterhin gültig.
- cont_FM455.SP_OP_ON = FALSE: Dadurch ist Sollwert SP_RE wirksam.
- cont_FM455.LOAD_PAR = TRUE: Die Reglerparameter werden vom Instanz-DB zur FM 455 übertragen.

Arbeitsweise

ZONE enthält die Anweisungen PID_FM_455 und PID_PAR_455 als lokale Instanzen.

ZONE setzt bei jedem Aufruf die Parameter cont_FM455.LOAD_OP = TRUE und cont_FM455.READ_VAR = TRUE. Alle Bedienparameter (cont_FM455.SP_RE...cont_FM455.LMNRSVAL) und alle Prozesswerte (cont_FM455.SP...cont_FM455.QMOD_F) werden somit aktualisiert.

Sie können Reglerparameter (cont_FM455.P_SEL...cont_FM455.BREAK_TM) ändern, in dem Sie direkt den Wert am Instanz-DB auf der CPU ändern und das Bit cont_FM455.LOAD_PAR für einen Zyklus setzen. Sie benötigen keinen Hilfs-DB. Nach erfolgreicher Übertragung wird das Bit wieder zurückgesetzt. Nach Neustart der CPU werden die Reglerparameter nicht von den Reglerparametern des SDB überschrieben.

Wenn Sie ohne LP_ZONE arbeiten, müssen Sie den Parameter NBR_CALL=0 setzen.

Durch ZONE änderbare Parameter der Anweisung PID_FM_455

Tabelle 4-292 Die folgende Liste enthält die Parameter der Anweisung PID_FM_455, die von ZONE beeinflusst werden:

Parameter	Kommentar
cont_FM455.READ_VAR	Wird zyklisch auf TRUE gesetzt
cont_FM455.LOAD_OP	Wird zyklisch auf TRUE gesetzt
cont_FM455.LOAD_PAR	Wird bei einer Regleroptimierung und bei Änderung der Parameter CONZONE und RATIOFAC auf TRUE gesetzt.
cont_FM455.MOD_ADDR	Wird zyklisch auf den Eingang MOD_ADDR gesetzt.
cont_FM455.CHANNEL	Wird zyklisch auf den Eingang CHANNEL gesetzt.
cont_FM455.SDB_SEL	Wird bei Neustart auf TRUE gesetzt. Bei RUN-STOP Übergang der CPU werden die Reglerparameter (P_SEL...BREAK_TM) des SDBs von der FM 455 nicht übernommen.
cont_FM455.SP_OP_ON	Wird bei Neustart auf FALSE gesetzt, so dass SP_RE wirksam ist.
cont_FM455.SP_RE	Wird zyklisch auf SP_IN gesetzt
cont_FM455.GAIN cont_FM455.TI cont_FM455.TD cont_FM455.TM_LAG cont_FM455.D_EL_SEL cont_FM455.PFDB_SEL cont_FM455.LOAD_PAR	Werden gesetzt bei Regleroptimierung, Zurückladen der gesicherten Reglerparameter und Laden der PI/PID-Reglerparameter.

Parameter	Kommentar
cont_FM455.LMN_LLM cont_FM455.LOAD_PAR	Werden gesetzt bei Änderung von RATIOFAC und COOLZONE
cont_FM455.LMN_REON cont_FM455.LMN_RE cont_FM455.LOAD_OP	Werden zyklisch gesetzt

Durch ZONE änderbare Parameter der Anweisung PID_PAR_455

Tabelle 4-293 Die folgende Liste enthält die Parameter der Anweisung PID_PAR_455, die von ZONE beeinflusst werden:

Parameter	Kommentar
pidpar_FM455.COM_RST	Wird bei Neustart auf TRUE gesetzt
pidpar_FM455.MOD_ADDR	Wird zyklisch auf den Eingang MOD_ADDR gesetzt.
pidpar_FM455.CHANNEL	Wird zyklisch auf den Eingang CHANNEL gesetzt.
pidpar_FM455. INDEX_R	Wird auf 43 gesetzt bei Änderung von RATIOFAC und COOLZONE
pidpar_FM455. VALUE _R	Wird gesetzt bei Änderung von RATIOFAC und COOLZONE:

Eingangsparameter ZONE

Tabelle 4-294

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
MOD_ADDR	0.0	INT	512	Baugruppenadresse
CHANNEL	2.0	INT	1	Kanalnummer
LPSCH_EN	4.0	BOOL	TRUE	Wenn LPSCH_EN = TRUE, wird ZONE über die Anweisung LP_ZONE gesteuert. Der Durchgangsparameter LP_SCH muss mit der statischen Variable LP_SCH der Anweisung LP_ZONE verschaltet sein. Sonst geht die CPU mit "Bereichsfehler beim Lesen" in STOP. Wenn LPSCH_EN = FALSE, wird ZONE bei jedem Weckalarmzyklus bearbeitet. Der Durchgangsparameter LP_SCH muss nicht versorgt werden. Die Abtastzeit muss an der statischen Variable CYCLE eingetragen sein.
ADMIN_EN	4.1	BOOL	TRUE	Wenn ADMIN_EN = TRUE, werden mehrere Heiz-Kühlzonen über die Anweisung ADM_ZONE verwaltet. Der Durchgangsparameter ADMIN muss mit der statischen Variable ADMIN der Anweisung ADM_ZONE verschaltet sein. Sonst geht die CPU mit "Bereichsfehler beim Lesen" in STOP. Wenn ADMIN_EN = FALSE, arbeitet ZONE ohne die Anweisung ADM_ZONE. Der Durchgangsparameter ADMIN muss nicht versorgt werden.

Durchgangsparameter ZONE

Tabelle 4-295

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LP_SCH	6.0	STRUCT	P#P 0.0	Schnittstelle zwischen den Anweisungen LP_ZONE und ADM_ZONE bzw. ZONE
ADMIN	12.0	STRUCT	P#P 0.0	Schnittstelle zwischen den Anweisungen ADM_ZONE und ZONE

Statische Variablen ZONE

Tabelle 4-296

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
cont_FM455	18.0	FB PID_FM_455		Lokale Instanz der Anweisung PID_FM_455
pid-par_FM455	172.0	FB PID_PAR_455		Lokale Instanz der Anweisung PID_PAR_455
SP_IN	426.0	REAL	0.0	Der Eingang SP_IN dient zur Vorgabe eines Sollwertes.
CYCLE	430.0	REAL	0.1	Vorgabe der Abtastzeit für den PID-Algorithmus. Bei Anwendung der Anweisung LP_ZONE wird die Abtastzeit automatisch von LP_ZONE berechnet.
H_OFF_EN	434.0	BOOL	TRUE	TRUE= Schalter HEAT_OFF ist wirksam und beeinflusst die Parameter MAN_ON und MAN.
TUN_EN	434.1	BOOL	TRUE	Ist bei einer Regleroptimierung TUN_EN=0 werden keine neuen Reglerparameter berechnet. Vorsicht der Stellwertsprung wird trotzdem aufgeschaltet.
COOLZONE	434.2	BOOL	FALSE	COOLZONE = FALSE: reine Heizzone COOLZONE = TRUE: Heizkühlzone
PID_ON	434.3	BOOL	TRUE	Am Eingang PID_ON können Sie festlegen, ob der optimierte Regler als PI- oder als PID-Regler arbeiten soll. PID-Regler: PID_ON = TRUE PI-Regler: PID_ON = FALSE Es kann jedoch sein, dass bei manchen Streckentypen trotz PID_ON = TRUE nur ein PI-Regler entworfen wird.
QCH_F	434.4	BOOL	FALSE	QCH_F = TRUE: Bei einem der Aufrufe der lokalen Instanz der Anweisung PID_FM_455 (Parameter cont_FM455.QCH_F) ist ein Kanalfehler der FM 455 aufgetreten.
QTUN_ERR	434.5	BOOL	FALSE	QTUN_ERR = TRUE: Bei der Regleroptimierung ist ein Bedienfehler aufgetreten. STATUS_H oder STATUS_C = 3xxxx
QTUN_WRN	434.6	BOOL	FALSE	QTUN_WRN = TRUE: Bei der Regleroptimierung ist eine Warnmeldung aufgetreten. STATUS_H oder STATUS_C = 2xxxx

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QTUNSTCP	434.7	BOOL	FALSE	QTUNSTCP = TRUE: Die Optimierung wurde von kaltem Prozesszustand gestartet und nach der Optimierung soll auf den Arbeitspunkt geregelt werden. MAN_ON = FALSE; MAN = 0 (Heizung aus) SP_IN > PV0 +10 (Arbeitspunkt um 10 größer als Istwert beim Start der Optimierung)
QCONTINU	435.0	BOOL	TRUE	Zeigt an, ob die Zone bereit ist für die nächste Phase der Regleroptimierung.
PHASE	436.0	INT	0	Am Parameter PHASE wird die aktuelle Ablaufphase der Regleroptimierung angezeigt (0..7, 11..13).
STATUS_H	438.0	INT	0	Parameter STATUS_H zeigt einen Diagnosewert über die Suche des Wendepunktes beim Heizvorgang an.
STATUS_C	440.0	INT	0	Parameter STATUS_C zeigt einen Diagnosewert über die Suche des Wendepunktes beim Kühlvorgang an.
STATUS_D	442.0	INT	0	Parameter STATUS_D zeigt einen Diagnosewert über den Reglerentwurf beim Heizvorgang an.
PI_CON	444.0	STRUCT		PI Reglerparameter
GAIN	+0.0	REAL	0.0	PI Reglerverstärkung
TI	+4.0	REAL	0.0	PI Integrationszeit [s]
	=8.0	END_STRUCT		
PID_CON	452.0	STRUCT		PID Reglerparameter
GAIN		REAL	0.0	PID Reglerverstärkung
TI		REAL	0.0	PID Integrationszeit [s]
TD		REAL	0.0	PID Differenzierzeit [s]
	=12.0	END_STRUCT		
PAR_SAVE	464.0	STRUCT		In dieser Struktur werden die PID-Parameter gespeichert.
GAIN	+0.0	REAL	0.0	Reglerverstärkung
TI	+4.0	REAL	0.0	Integrationszeit [s]
TD	+8.0	REAL	0.0	Differenzierzeit [s]
TM_LAG	+12.0	REAL	0.0	Verzögerung des D-Anteils [s]
D_EL_SEL	+16.0	INT	0	Eingang für D-Glied
PFDB_SEL	+18.0	BOOL	FALSE	P-Anteil in Rückführung
CONZ_ON	+18.1	BOOL	FALSE	Regelzone einschalten
CON_ZONE	+20.0	REAL	100.0	Regelzonenbreite
RATIOFAC	+24.0	REAL	1.0	Verhältnis von Heiz-/ Kühlverstärkung der Strecke.
	=28.0	END_STRUCT		

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
HEAT_OFF	492.0	BOOL	FALSE	TRUE= Heizung ausschalten Sie müssen H_OFF_EN = TRUE setzen um HEAT_OFF wirksam werden zu lassen. • Beim Setzen von HEAT_OFF wird MAN_ON=TRUE und MAN=0.0. • Beim Rücksetzen von HEAT_OFF wird MAN_ON=FALSE.
TUN_ON	492.1	BOOL	FALSE	Regleroptimierung einschalten Setzen Sie TUN=TRUE nur für einem Zyklus. TUN_ON wird nach der Optimierung automatisch rückgesetzt.
CTUN_ON	492.2	BOOL	FALSE	Nur Kühloptimierung einschalten Setzen Sie CTUN =TRUE nur für einem Zyklus. CTUN _ON wird nach der Optimierung automatisch rückgesetzt.
COM_RST	492.3	BOOL	FALSE	Initialisierungsroutine wird angestoßen
TUN_LMN	494.0	REAL	80.0	Die Prozessanregung für die Regleroptimierung von kaltem Prozesszustand erfolgt durch einen Stellwertsprung um TUN_LMN.
TUN_DLMN	498.0	REAL	20.0	Die Prozessanregung für die Regleroptimierung am Arbeitspunkt erfolgt durch einen Stellwertsprung um TUN_DLMN.
TUN_CLMN	502.0	REAL	-10.0	Stellwert nach Start der Kühloptimierung.
STAT_TM	506.0	REAL	30.0	Zeit für Stationaritätsbedingung für Phase 1 bzw. 11 der Regleroptimierung Prozess gilt als stationär, wenn innerhalb der Zeit STAT_TM der Ist- bzw. Stellwert sich nicht mehr als um 2*STAT_D_PV bzw. 2*STATDLMN ändert.
STAT_DPV	510.0	REAL	1.0	Delta Istwert für Stationaritätsbedingung für Phase 1 bzw. 11 der Regleroptimierung Siehe STAT_TM
STATDLMN	514.0	REAL	4.0	Delta Stellwert für Stationaritätsbedingung für Phase 1 bzw. 11 der Regleroptimierung Siehe STAT_TM
RATIOFAC	518.0	REAL	1.0	Verhältnis von Heiz-/ Kühlverstärkung der Strecke. Stellwertuntergrenze und "Bereichanfang Eingangssignal von Stellwert B der Splitrangefunktion" werden automatisch angepasst.
CON_ZONE	522.0	REAL	100.0	Regelzone Wenn ER >= CON_ZONE, dann LMN = LMN_HLM. Wenn ER <= -CON_ZONE (bzw. -CON_ZONE /RATIOFAC), dann LMN = LMN_LLM.
MAN	526.0	REAL	0.0	Handwert Der Parameter wird bei H_OFF_EN=TRUE von HEAT_OFF beeinflusst.
MAN_ON	530.0	BOOL	TRUE	Hand/Automatik-Umschaltung Der Parameter wird bei H_OFF_EN=TRUE von HEAT_OFF beeinflusst.
CONZ_ON	530.1	BOOL	FALSE	TRUE = Regelzone eingeschaltet

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
UNDO_PAR	530.2	BOOL	FALSE	Kopiert die Reglerparameter der Struktur PAR_SAVE auf die Parameter: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG • cont_FM455.D_EL_SEL • cont_FM455.PFDB_SEL • CONZ_ON • CON_ZONE • RATIOFAC Danach werden die Parameter auf die FM 455 geladen. Die Funktion wird bei Automatikbetrieb nicht durchgeführt.
SAVE_PAR	530.3	BOOL	FALSE	Sichert die folgenden Reglerparameter in die Struktur PAR_SAVE: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG • cont_FM455.D_EL_SEL • cont_FM455.PFDB_SEL • CONZ_ON • CON_ZONE • RATIOFAC
LOAD_PID	530.4	BOOL	FALSE	PI/PID-Parameter laden: Kopiert bei PID_ON=TRUE die PID-Parameter der Struktur PID_CON auf die Parameter: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG Kopiert bei PID_ON=FALSE die PI-Parameter der Struktur PI_CON auf die Parameter: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD = 0.0 • cont_FM455.TM_LAG = 0.5 Danach werden die Parameter auf die FM455 geladen. Die Funktion wird bei Automatikbetrieb nicht durchgeführt.
BTUN_DPV	532	REAL	0.0	Istwertdifferenz für den automatischen Abbruch der Optimierung.
BTUN_TM	536	REAL	300.0	Zeit nach dem Stellwertsprung für den automatischen Abbruch der Optimierung.

Variablen zur Regleroptimierung ZONE

Tabelle 4-297

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
GAIN_P	540.0	REAL	0.0	Identifizierte Prozessverstärkung.
TU	544.0	REAL	0.0	Identifizierte Verzugszeit vom Prozess.
TA	548.0	REAL	0.0	Identifizierte Ausgleichszeit vom Prozess.
KIG	552.0	REAL	0.0	Maximal möglicher Anstieg des Istwertes beim Heizen.
KIG_C	556.0	REAL	0.0	Maximal möglicher Abstieg des Istwertes beim Kühlen.
N_PTN	560.0	REAL	0.0	Ordnung der Strecke. Auch "nicht ganzzahlige Werte" sind möglich.
TM_LAG_P	564.0	REAL	0.0	Zeitkonstante eines PTN-Modells. (sinnvolle Werte nur für N_PTN>=2).
T_P_INF	568.0	REAL	0.0	Zeit von der Prozessanregung bis zum Wendepunkt.
P_INF	572.0	REAL	0.0	Istwertänderung von der Prozessanregung bis zum Wendepunkt.
LMN0	576.0	REAL	0.0	Stellwert zu Beginn der Optimierung. Wird in Phase 1 ermittelt (Mittelwert).
PV0	580.0	REAL	0.0	Istwert zu Beginn der Optimierung. Wird in Phase 1 ermittelt (Mittelwert).
PVDT0	584.0	REAL	0.0	Istwertsteigung zu Beginn der Optimierung.
PVDT	588.0	REAL	0.0	Momentane Istwertsteigung
PVDT_MAX	592.0	REAL	0.0	Maximale Ableitung des Istwertes am Wendepunkt (Vorzeichen angepasst, immer > 0), wird verwendet zur Berechnung von TU und KIG.
NOI_PVDT	596.0	REAL	0.0	Rauschanteil in PVDT_MAX in %. Je größer der Rauschanteil, desto ungenauer (sanfter) die Reglerparameter.
NOISE_PV	600.0	REAL	0.0	Absolutes Istwert-Rauschen; Differenz zwischen maximalem und minimalem Istwert in Phase 1.
FIL_CYC	604.0	INT	1	Anzahl der Zyklen des Mittelwertfilters 1 ... 1024; Der Istwert wird über FIL_CYC Zyklen gemittelt. FIL_CYC wird bei Bedarf automatisch von 1 bis max. 1024 erhöht.
POI_CMAX	606.0	INT	0	Maximale Anzahl Zyklen nach Wendepunkt. Diese Zeit wird genutzt, um bei Messrauschen einen weiteren (d. h. besseren) Wendepunkt zu finden. Erst dann wird die Optimierung beendet.
POI_CYCL	608.0	INT	0	Anzahl Zyklen nach Wendepunkt.
NBR_CALL	610.0	INT	-1	Aufrufnummer; wird nach Neustart automatisch eingetragen.

Danach folgen interne Parameter, die vom Anwender nicht benutzt werden dürfen.

Parameter STATUS_H und STATUS_C

STATUS_H zeigt einen Diagnosewert über die Suche des Wendepunktes beim Heizen, STATUS_C beim Kühlen.

Wenn ein Diagnosewert 2xxxx angezeigt wird, wird QTUN_WRN = TRUE.

Wenn ein Diagnosewert 3xxxx angezeigt wird, wird QTUN_ERR = TRUE.

STATUS_ H STATUS_C	Beschreibung	Abhilfe
0	Default bzw. keine oder noch keine neuen Reglerparameter	
10000	Optimierung beendet + geeignete Reglerparameter gefunden	
2xxxx	Optimierung beendet + Reglerparameter unsicher	
2xx2x	Wendepunkt nicht erreicht (nur bei Optimierung von kaltem Prozesszustand in den Arbeitspunkt QTUNSTCP =TRUE)	Falls Regler schwingt, Reglerparameter abschwächen oder Versuch wiederholen mit kleinerer Stellwertdifferenz TUN_LMN.
2x1xx	Schätzfehler TU zu klein TU < 3*CYCLE	CYCLE verringern und Versuch wiederholen. Sonderfall reine PT1-Strecke: Versuch nicht wiederholen, Reglerparameter evtl. abschwächen.
2x2xx	Schätzfehler T_P_INF zu klein T_P_INF < 72.8*CYCLE	siehe 2x1xx
2x3xx	Schätzfehler TU zu klein und Schätzfehler T_P_INF zu klein	siehe 2x1xx
3xxxx	Optimierung abgebrochen. Es wurde kein Reglerparameter ermittelt.	
30002	Bei Heizoptimierung: Effektive Stellwertdifferenz < 5 % Bei Kühloptimierung: TUN_CLMN > -5 %	Stellwertdifferenz TUN_LMN, TUN_DLMN bzw. TUN_CLMN korrigieren.
30003	Bei Kühloptimierung ist Stellwertuntergrenze nicht korrekt: cont_FM455.LMN_LLM > -5.0	Zone als Heizkühlzone einstellen COOLZONE =TRUE und cont_FM455.LMN_LLM kontrollieren.
30004	Effektive Stellwertdifferenz begrenzt durch Splitrangegrenzen und nicht durch Stellwertgrenzen.	Stellwertgrenzen und Splitrangegrenzen kontrollieren. Stellwertdifferenz TUN_LMN, TUN_DLMN bzw. TUN_CLMN korrigieren.
30005	Die Abtastzeit CYCLE weicht um mehr als 5% von dem gemessenen Wert ab.	Vergleichen Sie CYCLE mit der Zykluszeit der Weckalarmebene und beachten Sie evt. Vorhandene Aufrufverteiler. Prüfen Sie die Auslastung der CPU. Eine zu stark ausgelastete CPU führt zu verlängerten Abtastzeiten, die nicht zu CYCLE passen.
30008	Es wird versucht eine Kühloptimierung durchzuführen, zuvor wurde jedoch keine Heizoptimierung gemacht (KIG=0.0).	Zunächst Heizoptimierung durchführen.
30009	Sicherheitsbetrieb: cont_FM455.QLMNSAFE = TRUE	Sicherheitsbetrieb beenden und Optimierung erneut starten.
30010	Eine Änderung der Heiz-/Kühlleistung bewirkt keine ausreichende Änderung der Temperatur.	Überprüfen Sie den Signalverlauf von Stell- und Zeitwert. Kontrollieren Sie, ob die Parameter BTUN_TM und BTUN_DPV plausibel sind.

Siehe auch

Ausgangsparameter ADM_ZONE (Seite 6326)

Parameter STATUS_D

STATUS_D	Beschreibung
0	Es wurden keine Reglerparameter berechnet
110	N_PTN <= 1.5 Streckentyp I schnell
121	N_PTN > 1.5 Streckentyp I
200	N_PTN > 1.9 Streckentyp II (Übergangsbereich)
310	N_PTN >= 2.1 Streckentyp III schnell
320	N_PTN > 2.6 Streckentyp III
111, 122, 201, 311, 321	Parameter wurden von Phase 7 korrigiert.

Hinweis

Am Ende der Phase 1 wird STATUS_H = 0 rückgesetzt. Wenn Sie die Optimierung in Phase 2 abbrechen, ist STATUS_H = 0. STATUS_D zeigt jedoch immer noch den Zustand der letzten Reglerberechnung an.

Je größer der Wert von STATUS_D, desto höher ist die Ordnung der Regelstrecke, desto größer ist das Verhältnis TU/TA und desto sanfter sind die Reglerparameter.

Parameter PHASE

PHASE	Beschreibung
0	kein Optimierbetrieb; Automatik- oder Handbetrieb
1	Stationaritätsprüfung; Parameter prüfen; Aufschalten eines konstanten Heiz-Stellwertes
2	Wendepunktsuche bei konstantem Stellwert
3 (1 Zyklus pro Anweisung ZONE)	Berechnen der Prozess-Parameter. Abspeichern der vor der Optimierung gültigen Reglerparameter
4 (1 Zyklus pro Anweisung ZONE)	Reglerentwurf
5 (1 Zyklus pro Anweisung ZONE)	Nachführen des Reglers auf neue Stellgröße
7	Überprüfung des Streckentyps (nur bei Heizoptimierung)
10	Kühloptimierung: Warten bis alle Zonen im Automatikbetrieb (Phase > 2) sind.
11	Kühloptimierung: Stationaritätsprüfung; Parameter prüfen; Aufschalten eines konstanten Kühl-Stellwertes
12	Kühloptimierung: Wendepunktsuche bei konstantem Stellwert
13	Kühloptimierung: Berechnen des Heizkühlverhältnisses

Phasen der Heizoptimierung bei Erstanlauf

In folgendem Bild sind die Phasen einer reinen Heizoptimierung von Umgebungstemperatur zum Arbeitspunkt dargestellt:

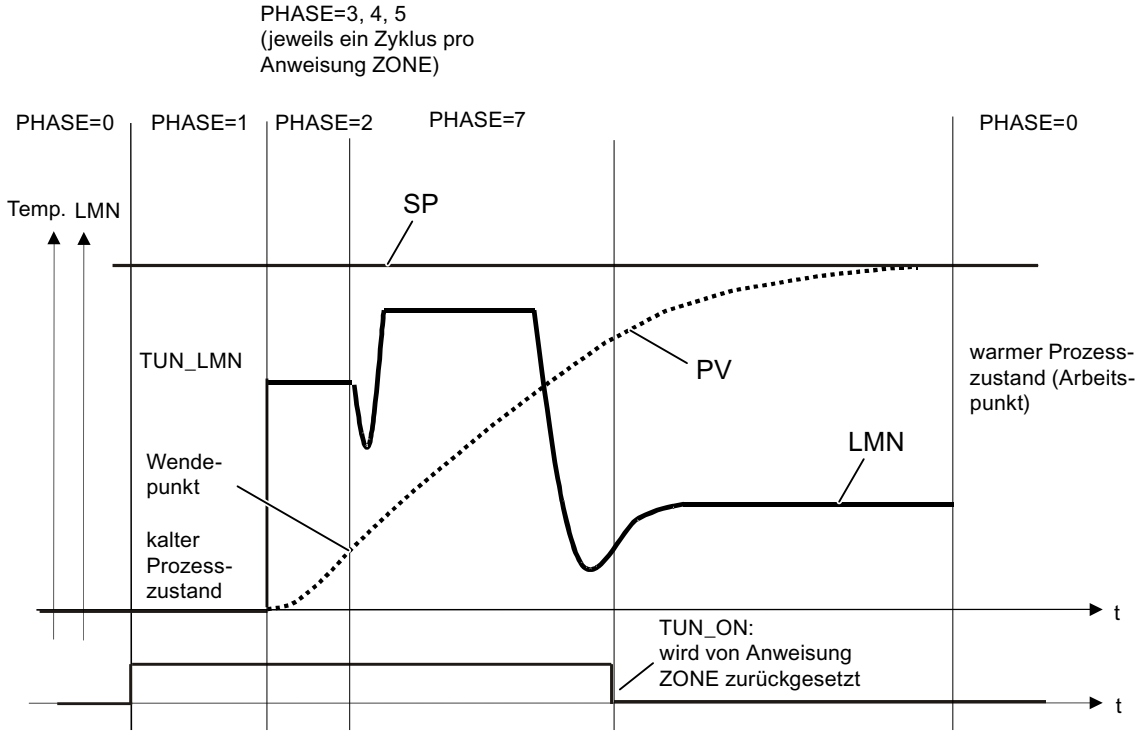

Phasen der Heizoptimierung im Arbeitspunkt

In folgendem Bild sind die Phasen einer reinen Heizoptimierung am Arbeitspunkt dargestellt:

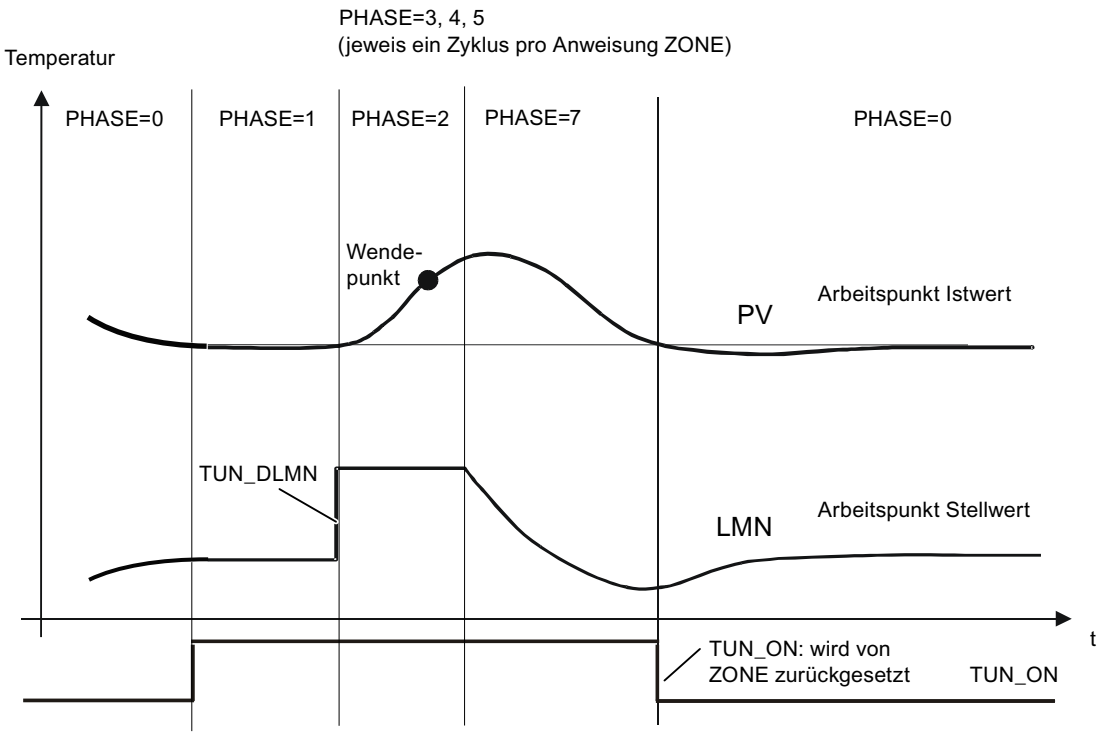

Am Ende der Optimierung, wenn die Anweisung wieder in Phase 0 wechselt und TUN_ON=FALSE setzt, können Sie an den Parametern STATUS_H, STATUS_C bzw. QTUN_WRN, QTUN_ERR erkennen, ob die Optimierung fehlerfrei durchlaufen wurde.

DB für HMI

Beschreibung DB 101 bis 116

Um bei CPU-Ausfall die FM 455 mit dem OP zu bedienen, benötigen Sie die Datenbausteine 101 bis 104. Über MPI können Sie maximal drei Verbindungen von der FM 455 zu OPs aufbauen.

Bedienen der FM 455 mit dem OP ist nur möglich im Zustand STOP der CPU oder bei CPU-Ausfall.

Beobachten der FM 455 mit dem OP ist immer möglich.

Die Variablenschnittstelle der FM 455 enthält 16 Datenbausteine mit den Bausteinnummern 101 bis 116 für die Reglerkanäle 1 bis 16 (siehe folgendes Bild).

Hinweis

Der Inhalt der Datenbausteine 101 bis 116 spiegelt nicht automatisch den auf der FM 455 wirksamen Parameterwert wieder. Mit Hilfe des OP geänderte Parameter werden nur nach Setzen der Bedienbits LOAD_PAR bzw. LOAD_OP in die FM 455 übernommen.

Wenn Sie durch OP-Bedienung einen Parameter ändern, ohne das entsprechende Bedienbit zu setzen, dann steht zwar der geänderte Parameterwert im Datenbaustein, die FM 455 arbeitet jedoch intern noch mit dem unveränderten Altwert des Parameters.

Nach Setzen der Bedienbits und Übernahme der Parameter in die FM 455 werden die Bedienbits LOAD_PAR bzw. LOAD_OP von der FM 455 zurückgesetzt.

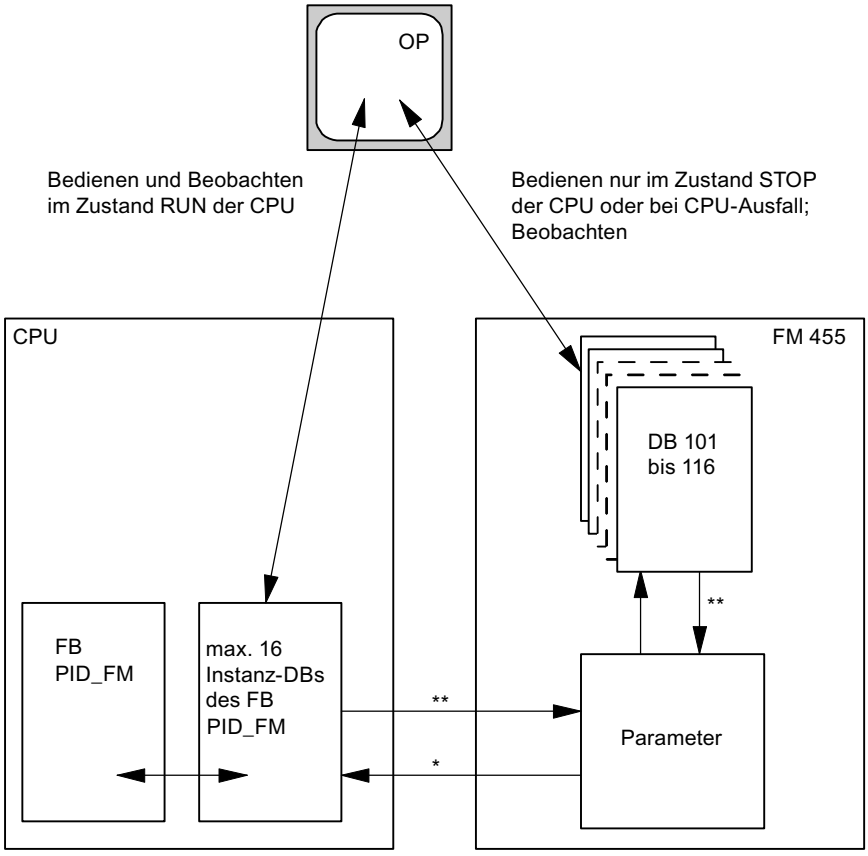

* gesteuert durch Parameter READ_VAR des Instanz-DB

** gesteuert durch die Parameter LOAD_OP und LOAD_PAR

Eingangsparameter DB 101 bis 116

Hinweis

Das EEPROM der Baugruppe könnte durch zu häufige Schreibvorgänge zerstört werden. Um dies zu verhindern, wird von der Baugruppe nach dem Beschreiben des EEPROM ein erneutes Beschreiben um 30 Minuten verzögert.

Tabelle 4-298

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
[1] Regelparameter; [2] Betriebsparameter				
SP_HLM[1]	0.0	REAL	100.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert obere Begrenzung" gibt die obere Begrenzung an. SP_HLM > SP_LLM
SP_LLM[1]	4.0	REAL	0.0	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Sollwert untere Begrenzung" gibt die untere Begrenzung an. SP_LLM < SP_HLM
H_ALM[1]	8.0	REAL	100.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Alarm" gibt die oberste Grenze an. H_ALM > H_WRN
H_WRN[1]	12.0	REAL	90.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "oberer Grenzwert Warnung" gibt die zweite Grenze von oben an. Zulässig sind Werte von H_ALM bis L_WRN.
L_WRN[1]	16.0	REAL	10.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Warnung" gibt die zweite Grenze von unten an. Zulässig sind Werte von H_WRN bis L_ALM.
L_ALM[1]	20.0	REAL	0.0	Für die Überwachung des Istwertes bzw. der Regeldifferenz können vier Grenzen parametriert werden. Der Eingang "unterer Grenzwert Alarm" gibt die unterste Grenze an. L_ALM < L_WRN
HYS[1]	24.0	REAL	1.0	Zur Vermeidung von Flackern der Überwachungsanzeigen kann am Eingang "Hysterese" eine Hysterese parametriert werden. HYS ≥ 0.0
DEADB_W[1]	28.0	REAL	0.0	Die Regeldifferenz wird über eine Totzone geführt. Der Eingang "Totzonenbreite" bestimmt die Größe der Totzone. DEADB_W ≥ 0.0
GAIN[1]	32.0	REAL	1.0	Der Eingang "Proportionalbeiwert" gibt die Reglerverstärkung an.
TI[1]	36.0	REAL	3000.0	Der Eingang "Integrationszeit" bestimmt das Zeitverhalten des Integrierers. Bei TI = 0 ist der Integrierer ausgeschaltet TI = 0.0 oder TI ≥ 0.5

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
TD[1)]	40.0	REAL	0.0	Der Eingang "Differenzierzeit" bestimmt das Zeitverhalten des Differenzierers. Bei TD = 0 ist der Differenzierer abgeschaltet. TD = 0.0 oder TD ≥ 1.0
TM_LAG[1)]	44.0	REAL	5.0	Der Algorithmus des D-Anteils beinhaltet eine Verzögerung, die am Eingang "Verzögerungszeit des D-Anteils" parametriert werden kann. TM_LAG ≥ 0.5
LMN_SAFE[1)]	48.0	REAL	0.0	Für den Stellwert kann ein Sicherheitswert am Eingang "Sicherheitsstellwert" parametriert werden. Zulässig sind Werte von -100 bis 100 %.
LMN_HLM[1)]	52.0	REAL	100.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert obere Begrenzung" gibt die obere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von LMN_LLM bis 100 %.
LMN_LLM[1)]	56.0	REAL	0.0	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Eingang "Stellwert untere Begrenzung" gibt die untere Begrenzung an. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100 bis LMN_HLM.
MTR_TM[1)]	60.0	REAL	60.0	Am Parameter "Motorstellzeit" wird die Laufzeit des Stellventils von Anschlag zu Anschlag eingetragen. (Nur beim Schrittregler) MTR_TM ≥ 0.001
PULSE_TM[1)]	64.0	REAL	0.2	Am Parameter "Mindestimpulsdauer" kann eine minimale Impulslänge parametriert werden. (Nur beim Schrittregler oder Impulsregler)
BREAK_TM[1)]	68.0	REAL	0.2	Am Parameter "Mindestpausendauer" kann eine minimale Impulspausenlänge parametriert werden. (Nur beim Schrittregler oder Impulsregler)
SP_RE[2)]	72.0	REAL	0.0	Ein externer Sollwert wird am Eingang "externer Sollwert" mit dem Regler verschaltet. technischer Wertebereich (physik. Größe)
LMN_RE[2)]	76.0	REAL	0.0	Ein externer Stellwert wird am Eingang "externer Stellwert" mit dem Regler verschaltet. Zulässig sind Werte von -100 bis 100 %.
LMNRSVAL[2)]	80.0	REAL	0.0	Vom Dialog Inbetriebnahme haben Sie Zugriff auf den Eingang "Startwert der simulierten Stellungsrückmeldung". Am Parameter wird der Startwert der Simulation eingetragen. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung) Zulässig sind Werte von -100 bis 100 %.
SAFE_ON[2)]	84.0	BOOL	FALSE	Ist der Eingang "Sicherheitsstellung einnehmen" gesetzt, wird ein Sicherheitswert als Stellwert übernommen. Hinweis: Die Stellsignalbearbeitung über LMNDN_OP, LMNUP_OP und LMNSOPON beim Schrittregler ist höher prior als der Sicherheitsstellwert.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
LMNTRKON[2)]	84.1	BOOL	FALSE	Ist der Eingang "Nachführen (LMN über AE)" gesetzt, wird der Stellwert einem Analogeingang (AE) nachgeführt. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
LMN_REON[2)]	84.2	BOOL	FALSE	Ist der Eingang "externer Stellwert einschalten" gesetzt, wird der externe Stellwert LMN_RE als Stellwert übernommen.
LMNRHSRE[2)]	84.3	BOOL	FALSE	Das Signal "Stellventil am oberen Anschlag" wird am Eingang "oberes Anschlagsignal der Stellungsrück-meldung" verschaltet. LMNRHSRE = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)
LMNRLSRE[2)]	84.4	BOOL	FALSE	Das Signal "Stellventil am unteren Anschlag" wird am Eingang "unteres Anschlagsignal der Stellungsrück-meldung" verschaltet. LMNRLSRE = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
LMNSOPON[2)]	84.5	BOOL	FALSE	Ist das Bit an dem Eingang "Stellwertsignalbedienung einschalten" gesetzt, werden die Signale LMNUP_OP und LMNDN_OP als Stellwertsignale übernommen. (Nur beim Schrittregler)
LMNUP_OP[2)]	84.6	BOOL	FALSE	Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Hoch Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
LMNDN_OP[2)]	84.7	BOOL	FALSE	Ist LMNSOPON gesetzt, wird der Wert am Eingang "Stellwertsignal Tief Bedienung" als Stellwertsignal übernommen. (Nur beim Schrittregler)
MONERSEL[1)]	85.0	BOOL	FALSE	Der Regler besitzt einen Grenzwertmelder, der entweder für den Istwert oder die Regeldifferenz angewendet werden kann. Ist der Eingang "Überwachung: Istwert = 0, Regeldifferenz = 1" gesetzt, wird die Regeldifferenz überwacht.
LMNRS_ON[2)]	85.1	BOOL	FALSE	Steht keine Stellungsrückmeldung zur Verfügung, so kann diese simuliert werden. Am Eingang "Simulation der Stellungsrückmeldung einschalten" wird die Funktion eingeschaltet. ACHTUNG: Die Simulation weicht mit zunehmender Zeit immer stärker vor der wirklichen Stellungsrückmeldung ab. (Nur beim Schrittregler ohne analoge Stellungsrückmeldung)
FUZID_ON[2)]	85.2	BOOL	FALSE	Am Eingang "Fuzzy-Identifikation einschalten" wird die Identifikation des Fuzzy-Algorithmus eingeschaltet.
SPINT_EN[2)]	85.3	BOOL	FALSE	Der Eingang "Bedieneingang: extern = 0, intern = 1" bestimmt den Eingang, der als Sollwert an die Baugruppe übertragen wird. SPINT_EN = TRUE: SP_INT wird übertragen. SPINT_EN = FALSE: SP_RE wird übertragen.
P_SEL[1)]	85.4	BOOL	TRUE	Im PID-Algorithmus lassen sich die PID-Anteile einzeln zu- und abschalten. Der Proportionalanteil ist eingeschaltet, wenn der Eingang "P-Anteil einschalten" gesetzt ist.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
PFDB_SEL[1]	85.5	BOOL	FALSE	Im PID-Algorithmus lassen sich die P- und D-Anteile in die Rückführung legen. Der Proportionalanteil liegt in der Rückführung, wenn der Eingang "PAnteil in Rückführung schalten" gesetzt ist.
D_EL_SEL[1]	86.0	INT	0	Das D-Glied im PID-Algorithmus kann an einen separaten Eingang gelegt werden. Am Eingang "Eingang für D-Glied" wird dieser ausgewählt. • 0: Regeldifferenz • 1 bis 16: Analogeingang 1 bis 16 • 17: negativer Istwert

Ausgangsparameter DB 101 bis 116

Tabelle 4-299

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP	94.0	REAL	0.0	Am Ausgang "Sollwert" wird der effektiv wirkende Sollwert ausgegeben.
PV	98.0	REAL	0.0	Am Ausgang "Istwert" wird der effektiv wirkende Istwert ausgegeben.
ER	102.0	REAL	0.0	Am Ausgang "Regeldifferenz" wird die effektiv wirkende Regeldifferenz ausgegeben.
DISV	106.0	REAL	0.0	Am Ausgang "Störgröße" wird die effektiv wirkende Störgröße ausgegeben. Zulässig sind Werte von -100 bis 100 %.
LMN	110.0	REAL	0.0	Am Ausgang "Stellwert" wird der effektiv wirkende Stellwert ausgegeben. Beim Schrittregler ohne analoge Stellungsrückmeldung wird am Parameter LMN der unbegrenzte P- + D-Anteil ausgegeben. Zulässig sind Werte von -100 bis 100 %.
LMN_A	114.0	REAL	0.0	Am Ausgang "Stellwert A der Splitrangefunktion / Stellungsrückmeldung" wird beim kontinuierlichen Regler der Stellwert A der Splitrangefunktion und beim Schrittregler mit analoger Stellungsrückmeldung die Stellungsrückmeldung angezeigt. Beim Schrittregler ohne analoge Stellungsrückmeldung wird die simulierte Stellungsrückmeldung angezeigt. Zulässig sind Werte von -100 bis 100 %.
LMN_B	118.0	REAL	0.0	Am Ausgang "Stellwert B der Splitrangefunktion" wird beim kontinuierlichen Regler der Stellwert B der Splitrangefunktion angezeigt. Zulässig sind Werte von -100 bis 100 %.
QH_ALM	122.0	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Alarm angesprochen" wird das Überschreiten der Grenze H_ALM gemeldet.
QH_WRN	122.1	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "oberer Grenzwert Warnung angesprochen" wird das Überschreiten der Grenze H_WRN gemeldet.
QL_WRN	122.2	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Warnung angesprochen" wird das Unterschreiten der Grenze L_WRN gemeldet.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QL_ALM	122.3	BOOL	FALSE	Der Istwert bzw. die Regelgröße wird auf vier Grenzwerte überwacht. Am Ausgang "unterer Grenzwert Alarm angesprochen" wird das Unterschreiten der Grenze L_ALM gemeldet.
QLMN_HLM	122.4	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Stellwertes angesprochen" meldet die Überschreitung der oberen Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QLMN_LLM	122.5	BOOL	FALSE	Der Stellwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Stellwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung. (Nicht beim Schrittregler ohne analoge Stellungsrückmeldung)
QSPINTON	122.6	BOOL	FALSE	Der Ausgang "interner Sollwert eingeschaltet" zeigt an, dass SP_INT an die Baugruppe übertragen wurde.
QPARA_F	123.0	BOOL	FALSE	Die Baugruppe überprüft Parameter auf Zulässigkeit. Ein Parametrierfehler wird am Ausgang "Parametrierfehler" angezeigt. Diesen Parametrierfehler können Sie auch im Dialog "Online & Diagnose" der Baugruppe auslesen.
QCH_F	123.1	BOOL	FALSE	Der Ausgang "Kanalfehler" wird gesetzt, wenn der Reglerkanal keine gültigen Ergebnisse liefern kann. Kanalfehler (z. B. Leitungsbruch) wird auch bei QPARA_F = 1 oder QMOD_F = 1 gesetzt. Wenn QCH_F = TRUE ist, dann ist die genaue Fehlerinformation im Diagnosedatensatz DS1 der Baugruppe auszulesen.
QUPRLM	123.2	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "positive Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertanstieg begrenzt.
QDNRLM	123.3	BOOL	FALSE	Der Sollwert wird auf positive und negative Steigung begrenzt. Ist der Ausgang "negative Steigungsbegrenzung des Sollwertes angesprochen" gesetzt, wird der Sollwertabstieg begrenzt.
QSP_HLM	123.4	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "obere Begrenzung des Sollwertes angesprochen" meldet die Überschreitung der oberen Begrenzung.
QSP_LLM	123.5	BOOL	FALSE	Der Sollwert wird immer auf eine obere und untere Begrenzung begrenzt. Der Ausgang "untere Begrenzung des Sollwertes angesprochen" meldet die Unterschreitung der unteren Begrenzung.
QSPOPON	123.6	BOOL	FALSE	Am Ausgang "Sollwertbedienung eingeschaltet" wird angezeigt, ob der Sollwert über das Dialog "Inbetriebnahme" bedient wird. Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
QLMNSAFE	123.7	BOOL	FALSE	Ist der Ausgang "Sicherheitsbetrieb" gesetzt, wird als Stellwert der Sicherheitsstellwert ausgegeben.
QLMNOPON	124.0	BOOL	FALSE	Am Ausgang "Stellwertbedienung eingeschaltet" wird angezeigt, ob der Stellwert über das Dialog "Inbetriebnahme" bedient wird. Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
QLMNTRK	124.1	BOOL	FALSE	Der Ausgang "Nachführbetrieb" zeigt an, ob der Stellwert über einen Analogeingang nachgeführt wird.
QLMN_RE	124.2	BOOL	FALSE	Der Ausgang "Hand = 1; Automatik = 0" zeigt an, ob der Stellwert auf den externen Stellwert LMN_RE gesetzt wird (Hand = 1) oder nicht.
QLMNR_HS	124.3	BOOL	FALSE	Der Ausgang "oberes Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am oberen Anschlag befindet. QLMNR_HS = TRUE heißt: Das Stellventil befindet sich am oberen Anschlag. (Nur beim Schrittregler)

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
QLMNR_LS	124.4	BOOL	FALSE	Der Ausgang "unteres Anschlagsignal der Stellungsrückmeldung" zeigt an, ob sich das Stellventil am unteren Anschlag befindet. QLMNR_LS = TRUE heißt: Das Stellventil befindet sich am unteren Anschlag. (Nur beim Schrittregler)
QLMNR_ON	124.5	BOOL	FALSE	Der Ausgang "Stellungsrückmeldung eingeschaltet" zeigt die eingestellte Betriebsart "Schrittregler mit Stellungsrückmeldung" bzw. "Schrittregler ohne Stellungsrückmeldung" an.
QFUZZY	124.6	BOOL	FALSE	Ist der Ausgang "PID-Algorithmus = 0; Fuzzy = 1" gesetzt, arbeitet der Regler mit dem Fuzzy-Algorithmus.
QSPLEPV	124.7	BOOL	FALSE	Der Ausgang "Anzeige vom FUZZY-Regler: Sollwert < Istwert" wird bei eingeschaltetem Fuzzy-Regler gesetzt, wenn der Sollwert kleiner als der wirksame Istwert ist.
QSPR	125.0	BOOL	FALSE	Ist der Ausgang "Split-Range-Betrieb" gesetzt, arbeitet der kontinuierliche Regler im Split-Range-Betrieb.
QLMNUP	125.1	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Hoch". (Nur beim Schrittregler oder Impulsregler)
QLMNDN	125.2	BOOL	FALSE	Ist der Ausgang "Stellwertsignal Tief". (Nur beim Schrittregler oder Impulsregler)
QBACKUP	125.4	BOOL	FALSE	• 0= kein Backup-Zustand (CPU in RUN) • 1= Backup-Zustand (CPU in STOP oder ausgefallen)
QID	125.5	BOOL	FALSE	QID = TRUE zeigt an, dass eine Identifizierung läuft (nicht dass sie eingeschaltet ist). Nach dem Ende der Identifizierung kann das Identifizierungsergebnis über den Parameter IDSTATUS der Anweisung CH_DIAG_455 ausgelesen werden.
QMAN_FC	125.6	BOOL	FALSE	Der Regler ist ein Führungsregler, der durch Handbetrieb eines Folgereglers auf dessen Istwert nachgeführt wird oder dessen I-Anteil angehalten wird, weil Sollwert oder Stellgröße des Folgereglers in der Begrenzung sind.
RET_VALU	126.0	WORD	0	RET_VALU enthält den Rückgabewert RET_VAL der Anweisungen RD_REC und WR_REC. RET_VALU kann ausgewertet werden, wenn über QMOD_F ein Fehler gemeldet wird.

Siehe auch

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

Durchgangsparameter DB 101 bis 116

Tabelle 4-300

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
1) Betriebsparameter				
SP_INT	128.0	REAL	0.0	Der Durchgangsparameter "interner Sollwert" dient zur Vorgabe eines Sollwertes mittels Bedien- und Beobachtungsfunktionen.

Parameter	Adresse	Datentyp	Vorbelegung	Beschreibung
SP_OP[1)]	132.0	REAL	0.0	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Sollwert Bedienung". Ist das Bit SP_OP_ON gesetzt, wird der Wert "Sollwert Bedienung" als Sollwert übernommen.
LMN_OP[1)]	136.0	REAL	0.0	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Stellwert Bedienung". Ist das Bit LMNOP_ON gesetzt, wird der Wert "Stellwert Bedienung" als Stellwert übernommen. Zulässig sind Werte von -100 bis 100 %.
SP_OP_ON [1)]	140.0	BOOL	FALSE	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Sollwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert SP_OP als Sollwert übernommen.
LMNOP_ON [1)]	140.1	BOOL	FALSE	Das Dialog "Inbetriebnahme" hat Zugriff auf den Durchgangsparameter "Stellwertbedienung einschalten". Ist das Bit gesetzt, wird der Wert LMN_OP als Stellwert übernommen.
LOAD_PAR	140.2	BOOL	FALSE	Ist der Durchgangsparameter "Regelparameter in FM 455 laden" gesetzt, werden die Regelparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.
LOAD_OP	140.3	BOOL	FALSE	Ist der Durchgangsparameter "Betriebsparameter in FM 455 laden" gesetzt, werden die Betriebsparameter in die Baugruppe geladen und der Durchgangsparameter rückgesetzt.

Hinweis

Das EEPROM der Baugruppe könnte durch zu häufige Schreibvorgänge zerstört werden. Um dies zu verhindern, wird von der Baugruppe nach dem Beschreiben des EEPROM ein erneutes Beschreiben um 30 Minuten verzögert.

4.2.4.4 300C Funktionen

ANALOG: Mit Analogausgang positionieren

Beschreibung

Zur Steuerung der Positionierfunktionen aus dem Anwenderprogramm verwenden Sie die Anweisung "ANALOG".

Ein fest zugeordneter Analogausgang steuert mit einer Spannung (Spannungssignal)zwischen ±10 V oder einem Strom (Stromsignal) von ±20 mA das Leistungsteil an.

- Nach Abschluss der Beschleunigungsphase (RAM_UP) wird das Ziel zunächst mit der Geschwindigkeit (V_{Soll}) angefahren.
- Am von der CPU berechneten Bremseinsatzpunkt wird die Verzögerung (RAMP_DN) bis zum Umschaltpunkt eingeleitet.
- Sobald der Umschaltpunkt erreicht ist, wird mit Schleichgeschwindigkeit ($V_{Schleich}$) weitergefahren.

- Am Abschaltpunkt wird der Antrieb abgeschaltet.
- Umschaltpunkt und Abschaltpunkt werden für jedes anzufahrende Ziel über die in den Parametern von Ihnen vorgegebenen Werte Umschaltdifferenz und Abschaltdifferenz festgelegt. Umschaltdifferenz und Abschaltdifferenz können für die Fahrtrichtung vorwärts (Richtung Plus) und rückwärts (Richtung Minus) unterschiedlich festgelegt werden.
- Die Fahrt ist beendet (WORKING = FALSE), wenn der Abschaltpunkt erreicht ist. Ab diesem Zeitpunkt kann eine neue Fahrt gestartet werden.
- Das vorgegebene Ziel ist erreicht (POS_RCD = TRUE), wenn der Lageistwert den Zielbereich erreicht hat. Verlässt der Lageistwert ohne Start einer neuen Fahrt wieder den Zielbereich, wird das Signal "Position erreicht" nicht wieder zurückgesetzt.

Wenn die Umschaltdifferenz kleiner als die Abschaltdifferenz ist, wird ab dem Bremseinsatzpunkt linear bis auf Geschwindigkeitssollwert 0 verzögert.

Grundparameter

Hier werden die für alle Betriebsarten identischen Parameter der Anweisung beschrieben. Die betriebsartenspezifischen Parameter sind bei den einzelnen Betriebsarten beschrieben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ANALOG":

Parameter	Deklaration	Datentyp	Adresse (Instanz DB)	Wertebereich	Vorbelegung	Beschreibung
LADDR	Input	WORD	0	CPU-spezifisch	W#16#0310	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben. Sind E- und A-Adresse nicht gleich, ist die kleinere der beiden Adressen anzugeben.
CHANNEL	Input	INT	2	0	0	Kanalnummer
STOP	Input	BOOL	4.4	TRUE/FALSE	FALSE	Fahrt Stoppen Mit STOP = TRUE kann die Fahrt vorzeitig beendet/unterbrochen werden.
ERR_A	Input	BOOL	4.5	TRUE/FALSE	FALSE	Sammelquittung Externfehler Mit ERR_A = TRUE werden Externfehler quittiert.

Parameter	Deklaration	Datentyp	Adresse (Instanz DB)	Wertebereich	Vorbelegung	Beschreibung
SPEED	Input	DINT	12	Schleich- geschwindigkeit bis 1 000 000 Impulse/s Höchstens bis zur parametrierten Maximal- geschwindigkeit	1000	Die Achse wird bis auf die Geschwindigkeit "V_{soll}" beschleunigt. Eine Änderung der Geschwindigkeit während der Fahrt ist nicht möglich.
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Fahrt läuft
ACT_POS	Output	DINT	18	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart
ERR	Output	WORD	24	Jedes Bit 0 oder 1	0	Externfehler: Bit 2 : Nullmarkenüberwachung Bit 11: Verfahrbereichsüberwachung (immer 1) Bit 12: Arbeitsbereichsüberwachung Bit 13: Istwertüberwachung Bit 14: Zieleinlaufüberwachung Bit 15: Zielbereichsüberwachung Restliche Bits reserviert
ST_ENBLD	Output	BOOL	26.0	TRUE/FALSE	TRUE	Die CPU setzt die Startfreigabe, wenn alle folgenden Bedingungen erfüllt sind: • STOP steht nicht an (STOP = FALSE) • Kein Externfehler steht an (ERR = 0) • Die Antriebsfreigabe ist gesetzt (DRV_EN = TRUE) • Keine Positionierung läuft (WORKING = FALSE)
ERROR	Output	BOOL	26.1	TRUE/FALSE	FALSE	Fehler beim Starten/Fortsetzen einer Fahrt
STATUS	Output	WORD	28.0	W#16#0000 bis W#16#FFFF	W#16#0000	Fehlernummer

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
ACCEL	STATIC	DINT	30	1 bis 100 000 Impulse/s^2	100	Beschleunigung Änderung während der Fahrt ist nicht möglich.
DECEL	STATIC	DINT	34	1 bis 100 000 Impulse/s^2	100	Verzögerung Änderung während der Fahrt ist nicht möglich.
CHGDIFF_P	STATIC	DINT	38	0 bis $+10^8$ Impulse	1000	Umschaltdifferenz plus: Die "Umschaltdifferenz plus" definiert den Umschaltpunkt, ab dem der Antrieb im Schleichgang in Vorwärtsrichtung fährt.
CUTOFF-DIFF_P	STATIC	DINT	42	0 bis $+10^8$ Impulse	100	Abschaltdifferenz plus: Die "Abschaltdifferenz plus" definiert den Abschaltpunkt, an dem der Antrieb aus dem Schleichgang in Vorwärtsrichtung abgeschaltet wird.
CHGDIFF_M	STATIC	DINT	46	0 bis $+10^8$ Impulse	1000	Umschaltdifferenz minus: Die "Umschaltdifferenz minus" definiert den Umschaltpunkt, ab dem der Antrieb im Schleichgang in Rückwärtsrichtung fährt.
CUTOFF-DIFF_M	STATIC	DINT	50	0 bis $+10^8$ Impulse	100	Abschaltdifferenz minus: Die "Abschaltdifferenz minus" definiert den Abschaltpunkt, an dem der Antrieb aus dem Schleichgang in Rückwärtsrichtung abgeschaltet wird.
PARA	STATIC	BOOL	54.0	TRUE/FALSE	FALSE	Achse parametriert
DIR	STATIC	BOOL	54.1	TRUE/FALSE	FALSE	Aktuelle/letzte Bewegungsrichtung FALSE = Vorwärts (Richtung Plus) TRUE = Rückwärts (Richtung Minus)
CUTOFF	STATIC	BOOL	54.2	TRUE/FALSE	FALSE	Antrieb im Abschaltbereich (ab dem Abschaltpunkt bis zum Start der nächsten Fahrt)
CHGOVER	STATIC	BOOL	54.3	TRUE/FALSE	FALSE	Antrieb im Umschaltbereich (ab Erreichen der Schleichgeschwindigkeit bis zum Start der nächsten Fahrt)
RAMP_DN	STATIC	BOOL	54.4	TRUE/FALSE	FALSE	Antrieb wird verzögert (vom Bremseinsatzpunkt bis zum Umschaltpunkt)

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
RAMP_UP	STATIC	BOOL	54.5	TRUE/FALSE	FALSE	Antrieb wird beschleunigt (vom Start bis zum Erreichen der Geschwindigkeit SPEED (V_{Soll}))
DIST_TO_GO	STATIC	DINT	56	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Restweg
LAST_TRG	STATIC	DINT	60	$-5x10^8$ bis $+5x10^8$ Impulse	0	Letztes/aktuelles Ziel • Schrittmaßfahrt absolut: Mit Start der Fahrt ist LST_TRG = aktuelles absolutes Ziel (TARGET). • Schrittmaßfahrt relativ: Mit Start der Fahrt ist LST_TRG = LAST_TRG der vorherigen Fahrt +/- angegebene Wegstrecke (TARGET).

Parameter für die Betriebsart "Tippen"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Tippen Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Tippen Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 1 = Tippen
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Fahrt läuft
ACT_POS	Output	DINT	18	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Referenzpunktfahrt"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Referenzpunktfahrt in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Referenzpunktfahrt in Richtung Minus (positive Flanke)

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 3 = "Referenzpunktfahrt"
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Fahrt läuft
SYNC	Output	BOOL	16.3	TRUE/FALSE	FALSE	SYNC = TRUE: Achse ist synchronisiert
ACT_POS	Output	DINT	18	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Schrittmaßfahrt relativ"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Fahren in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Fahren in Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 4 = Schrittmaßfahrt relativ
TARGET	Input	DINT	8	0 bis 10^9 Impulse	1000	Wegstrecke in Impulsen (nur positive Werte erlaubt)
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Fahrt läuft
POS_RCD	Output	BOOL	16.1	TRUE/FALSE	FALSE	Position erreicht
ACT_POS	Output	DINT	18	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Schrittmaßfahrt absolut"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
START	Input	BOOL	4.1	TRUE/FALSE	FALSE	Fahrt starten (positive Flanke)
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Fahren in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Fahren in Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 5 = Schrittmaßfahrt absolut

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
TARGET	Input	DINT	8	Linearachse: $-5x10^8$ bis $+5x10^8$ Rundachse: 0 bis Rund- achsenende -1	1000	Ziel in Impulsen
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Fahrt läuft
POS_RCD	Output	BOOL	16.1	TRUE/FALSE	FALSE	Position erreicht
ACT_POS	Output	DINT	18	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für den Auftrag "Bezugspunkt setzen"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
SYNC	Output	BOOL	16.3	TRUE/FALSE	FALSE	Achse ist synchronisiert

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_ID	STATIC	INT	78	1, 2	0	Auftrag, 1 = "Bezugspunkt setzen"
JOB_STAT	STATIC	WORD	80	W#16#0000 bis W#16#FFFF	W#16#0000	Auftragsfehler-Nummer
JOB_VAL	STATIC	DINT	82	$5x10^8$ bis $+5x10^8$ Impulse	0	Auftragsparameter Koordinate des Bezugspunktes

Parameter für den Auftrag "Restweg löschen"

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_ID	STATIC	INT	78	1, 2	0	Auftrag, 2 = "Restweg löschen"
JOB_STAT	STATIC	WORD	80	W#16#0000 bis W#16#FFFF	W#16#0000	Auftragsfehler-Nummer
JOB_VAL	STATIC	DINT	82	-	0	Einstellung beliebig

Parameter für die Funktion "Längenmessung"

Die Funktion wird über eine Flanke am Digitaleingang gestartet. Es gibt keine spezifischen Eingangs-Parameter.

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
MSR_DONE	OUT	BOOL	16.2	TRUE/FALSE	FALSE	Längenmessung beendet

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
BEG_VAL	STATIC	DINT	64	$-5x10^8$ bis $+5x10^8$ Impulse	0	Lageistwert Längenmessung Beginn
END_VAL	STATIC	DINT	68	$-5x10^8$ bis $+5x10^8$ Impulse	0	Lageistwert Längenmessung Ende
LEN_VAL	STATIC	DINT	72	0 bis 10^9 Impulse	0	Gemessene Länge

Parameter ERROR und STATUS

Fehler der Betriebsart (ERROR = TRUE)

Bei erkanntem Fehler wird der Ausgangsparameter ERROR auf TRUE gesetzt. Am Parameter STATUS wird die Fehlerursache angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#2002	Falsche Anweisung, Anweisung "ANALOG" verwenden
W#16#2004	Falsche Kanalnummer (CHANNEL). Stellen Sie als Kanalnummer "0" ein.
W#16#3001	Fahrauftrag wurde nicht angenommen, da Job im gleichen Aufruf fehlerhaft. Korrigieren Sie die Parameter vom entsprechenden JOB.
W#16#3002	Eine Änderung von Output _IN, während der Antrieb noch läuft, ist nicht erlaubt. Warten Sie, bis die laufende Positionierung beendet ist.
W#16#3003	Unbekannte Betriebsart (MODE_IN). Zulässig ist 1 (Tippen), 3 (Referenzpunktfahrt), 4 (Schrittmaßfahrt relativ) und 5 (Schrittmaßfahrt absolut).

Ereignisklasse Fehlercode	Erläuterung
W#16#3004	Es darf immer nur eine Startanforderung gleichzeitig gesetzt sein. Zulässige Startartanforderungen sind DIR_P oder DIR_M oder START.
W#16#3005	START nur in Betriebsart "Schrittmaßfahrt absolut" erlaubt. Starten Sie die Fahrt mit DIR_P oder DIR_M.
W#16#3006	DIR_P oder DIR_M bei Linearachse und Betriebsart "Schrittmaßfahrt absolut" nicht erlaubt. Starten Sie die Fahrt mit START.
W#16#3007	Achse nicht synchronisiert. "Schrittmaßfahrt absolut" ist nur bei synchronisierter Achse möglich.
W#16#3008	Arbeitsbereich verlassen. Fahrt nur per Tippen zurück in Richtung des Arbeitsbereichs erlaubt.
W#16#3101	Keine Startfreigabe, da die Achse nicht parametriert ist. Parametrieren Sie das Submodul "Positionieren" über HW-Konfig.
W#16#3102	Keine Startfreigabe, da keine Antriebsfreigabe gesetzt ist. Setzen Sie die "Antriebsfreigabe" an der Anweisung (DRV_EN=TRUE) .
W#16#3103	Keine Startfreigabe, da STOP gesetzt ist. Löschen Sie STOP an der Anweisung (STOP=FALSE)
W#16#3104	Keine Startfreigabe, da die Achse momentan positioniert (WORKING=TRUE). Warten Sie, bis die laufende Positionierung beendet ist.
W#16#3105	Keine Startfreigabe, da noch mindestens ein nicht quittierter externer Fehler ansteht. Beseitigen und quittieren Sie erst alle externen Fehler und starten sie dann die Fahrt neu.
W#16#3202	Geschwindigkeitsvorgabe SPEED falsch. Die Geschwindigkeitsvorgabe ist außerhalb des zulässigen Bereichs von Schleichgeschwindigkeit bis 1000000 Impulse/s. Aber höchstens bis zur parametrierten Maximalgeschwindigkeit.
W#16#3203	Die Beschleunigungsvorgabe ACCEL ist außerhalb des zulässigen Bereichs von 1 bis 100000 Impulse/s^2.
W#16#3204	Die Verzögerungsvorgabe DECEL ist außerhalb des zulässigen Bereichs von 1 bis 100000 Impulse/s^2.
W#16#3206	Die Geschwindigkeitsvorgabe SPEED muss größer/gleich der parametrierten Referenzierfrequenz sein.
W#16#3301	Um-/Abschaltdifferenz zu groß. Um-/Abschaltdifferenz maximal mit 10^8 vorgeben.
W#16#3304	Abschaltdifferenz zu klein. Die Abschaltdifferenz muss mindestens so groß sein wie der halbe Zielbereich.
W#16#3305	Umschaltdifferenz zu klein. Die Umschaltdifferenz muss mindestens so groß sein wie der halbe Zielbereich.
W#16#3401	Zielvorgabe außerhalb des Arbeitsbereichs. Bei einer Linearachse und Schrittmaßfahrt absolut muss die Zielvorgabe innerhalb der Softwareendschalter (inklusive) sein.
W#16#3402	Zielvorgabe falsch. Bei einer Rundachse muss die Zielvorgabe größer 0 und kleiner als das Rundachsenende sein.
W#16#3403	Wegangabe falsch. Das zu verfahrende Wegstück bei Schrittmaßfahrt relativ muss positiv sein.
W#16#3404	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss größer als $-5x10^8$ sein.
W#16#3405	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss kleiner als $5x10^8$ sein.
W#16#3406	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss innerhalb des Arbeitsbereichs (+/-halber Zielbereich) liegen.

Ereignisklasse Fehlercode	Erläuterung
W#16#3501	Verfahrbereich zu groß. Zielkoordinate + aktueller Restweg muss größer/gleich $-5x10^8$ sein.
W#16#3502	Verfahrbereich zu groß. Zielkoordinate + aktueller Restweg muss kleiner/gleich $5x10^8$ sein.
W#16#3503	Verfahrweg zu klein. Der Verfahrweg in Richtung Plus muss größer als die angegebene Abschaltdifferenz für die Richtung Plus sein
W#16#3504	Verfahrweg zu klein. Der Verfahrweg in Richtung Minus muss größer als die angegebene Abschaltdifferenz für die Richtung Minus sein
W#16#3505	Verfahrweg zu klein oder Endschalter bereits in Richtung Plus überfahren. Das letzte anfahrbare Ziel in Richtung Plus (Arbeitsbereich- oder Verfahrbereichsgrenze) liegt zu nahe an der aktuellen Position.
W#16#3506	Verfahrweg zu klein oder Endschalter bereits in Richtung Minus überfahren. Das letzte anfahrbare Ziel in Richtung Minus (Arbeitsbereich- oder Verfahrbereichsgrenze) liegt zu nahe an der aktuellen Position.

Auftragsfehler (JOB_ERR = TRUE)

Bei erkanntem Fehler wird der Ausgangsparameter JOB_ERR auf TRUE gesetzt. Am Parameter JOB_STAT wird die Fehlerursache angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#4001	Achse nicht parametriert. Parametrieren Sie das Submodul "Positionieren" über HW-Konfig.
W#16#4002	Auftrag nicht möglich, da noch eine Positionierung läuft. Warten Sie, bis WORKING = FALSE wird, und führen Sie den Auftrag erneut aus.
W#16#4004	Unbekannter Auftrag. Prüfen Sie die Auftragsnummer und führen Sie den Auftrag erneut aus.
W#16#4101	Bei einer Linearachse darf die Bezugspunktkoordinate nicht außerhalb der Arbeitsbereichsgrenzen liegen.
W#16#4102	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktueller Restweg noch größer/gleich $-5x10^8$ sein.
W#16#4103	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktueller Restweg noch kleiner/gleich $5x10^8$ sein.
W#16#4104	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktuelle Differenz zum Startpunkt der Fahrt noch größer/gleich $-5x10^8$ sein.
W#16#4105	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktuelle Differenz zum Startpunkt der Fahrt noch kleiner/gleich $5x10^8$sein.
W#16#4106	Bei einer Rundachse darf die Bezugspunktkoordinate nicht kleiner 0 und größer/gleich dem Rundachsenende sein.

Externfehler (ERR)

Von der Technologie werden Überwachungen bezüglich der Fahrt, des Verfahrbereichs und der angeschlossenen Peripherie durchgeführt. Voraussetzung ist, dass Sie die Überwachungen zuvor in den Parametriermasken "Antrieb", "Achse" und "Geber" eingeschaltet haben.

Bei Ansprechen der Überwachungen wird ein Externfehler gemeldet. Externfehler können unabhängig von gestarteten Funktionen jederzeit auftreten. Externfehler müssen Sie immer mit ERR_A = TRUE quittieren.

Die Externfehler werden am Parameter ERR (WORD) durch Setzen eines Bits angezeigt.

Überwachung	Fehlercode	Bit im ERR-WORD
Fehlimpuls (Nullmarke)	W#16#0004	2
Verfahrbereich	W#16#0800	11
Arbeitsbereich	W#16#1000	12
Istwert	W#16#2000	13
Zieleinlauf	W#16#4000	14
Zielbereich	W#16#8000	15

Systemfehler

Ein Systemfehler wird mit BIE = FALSE angezeigt. Ein Systemfehler wird ausgelöst durch Fehler beim Schreiben/Lesen des Instanz-DB oder durch Mehrfachaufruf der Anweisung.

DIGITAL: Mit Digitalausgang positionieren

Beschreibung

Die Anweisung "DIGITAL" verwenden Sie zur Steuerung der Positionierfunktionen aus dem Anwenderprogramm.

Vier dem Antrieb fest zugeordnete 24-V- Digitalausgänge steuern das Leistungsteil an (weitere Informationen hierzu entnehmen Sie der Dokumentation der verwendeten Hardware). Die Digitalausgänge steuern je nach parametrierter Ansteuerart die Richtung und die Geschwindigkeitsstufen (Eil-/Schleichgang).

Die Wegerfassung erfolgt über einen asymmetrischen 24-V-Inkrementalgeber mit zwei um 90 Grad phasenverschobenen Signalen.

- Zunächst wird das Ziel mit der Geschwindigkeit (V_{Eil}) angefahren.
- Am Umschaltpunkt wird auf Schleichgeschwindigkeit ($V_{Schleich}$) umgeschaltet.
- Am Abschaltpunkt wird der Antrieb abgeschaltet.
- Den Umschaltpunkt und Abschaltpunkt legen Sie über die Werte der Parameter CHGDIFF_P/M und CUTOFFDIFF_P/M fest. Umschaltdifferenz und Abschaltdifferenz können für die Fahrtrichtung vorwärts (Richtung Plus) und rückwärts (Richtung Minus) unterschiedlich definiert werden.
- Die Fahrt ist beendet (WORKING = FALSE), wenn der Abschaltpunkt erreicht ist. Ab diesem Zeitpunkt kann eine neue Fahrt gestartet werden.
- Das vorgegebene Ziel ist erreicht (POS_RCD = TRUE), wenn der Lageistwert den Zielbereich erreicht hat. Verlässt der Lageistwert ohne Start einer neuen Fahrt wieder den Zielbereich, wird das Signal "Position erreicht" nicht wieder zurückgesetzt.

Grundparameter:

Hier werden die für alle Betriebsarten identischen Parameter der Anweisung beschrieben. Die betriebsartenspezifischen Parameter sind bei den einzelnen Betriebsarten beschrieben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
LADDR	Input	WORD	0	CPU-spezifisch	W#16#0310	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben. Sind E- und A-Adresse nicht gleich, ist die kleinere der beiden Adressen anzugeben.
CHANNEL	Input	INT	2	0	0	Kanalnummer
STOP	Input	BOOL	4.4	TRUE/FALSE	FALSE	Fahrt Stoppen Mit STOP = TRUE kann die Fahrt vorzeitig beendet/unterbrochen werden.
ERR_A	Input	BOOL	4.5	TRUE/FALSE	FALSE	Sammelquittung Externfehler Mit ERR_A = TRUE werden Externfehler quittiert.
SPEED	Input	BOOL	12.0	TRUE/FALSE	FALSE	Zwei Geschwindigkeitsstufen für Eil-/Schleichgang TRUE=Eilgang FALSE=Schleichgang
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Fahrt läuft
ACT_POS	Output	DINT	16	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart
ERR	Output	WORD	22	Jedes Bit 0 oder 1	0	Externfehler • Bit2: Nullmarkenüberwachung • Bit11: Verfahrbereichsüberwachung (immer 1) • Bit12: Arbeitsbereichsüberwachung • Bit13: Istwertüberwachung • Bit14: Zieleinlaufsüberwachung • Bit15: Zielbereichsüberwachung Restliche Bits reserviert

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
ST_ENBLD	Output	BOOL	24.0	TRUE/FALSE	TRUE	Die CPU setzt die Startfreigabe, wenn alle folgenden Bedingungen erfüllt sind: • STOP steht nicht an (STOP = FALSE) • Kein Externfehler steht an (ERR = 0) • Die Antriebsfreigabe ist gesetzt (DRV_EN = TRUE) • Keine Positionierung läuft (WORKING = FALSE)
ERROR	Output	BOOL	24.1	TRUE/FALSE	FALSE	Fehler beim Starten/Fortsetzen einer Fahrt
STATUS	Output	WORD	26.0	W#16#0000 bis W#16#FFFF	W#16#0000	Fehlernummer

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
CHGDIFF_P	STATIC	DINT	28	0 bis $+10^8$ Impulse	1000	Umschaltdifferenz plus: Die "Umschaltdifferenz plus" definiert den Umschaltpunkt, an dem der Antrieb von Eilgang auf Schleichgang in Vorwärtsrichtung umschaltet.
CUTOFF-DIFF_P	STATIC	DINT	32	0 bis $+10^8$ Impulse	100	Abschaltdifferenz plus: Die "Abschaltdifferenz plus" definiert den Abschaltpunkt, an dem der Antrieb aus dem Schleichgang in Vorwärtsrichtung abgeschaltet wird.
CHGDIFF_M	STATIC	DINT	36	0 bis $+10^8$ Impulse	1000	Umschaltdifferenz minus: Die "Umschaltdifferenz minus" definiert den Umschaltpunkt, an dem der Antrieb von Eilgang auf Schleichgang in Rückwärtsrichtung umschaltet.
CUTOFF-DIFF_M	STATIC	DINT	40	0 bis $+10^8$ Impulse	100	Abschaltdifferenz minus: Die "Abschaltdifferenz minus" definiert den Abschaltpunkt, an dem der Antrieb aus dem Schleichgang in Rückwärtsrichtung abgeschaltet wird.

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
PARA	STATIC	BOOL	44.0	TRUE/FALSE	FALSE	Achse parametriert
DIR	STATIC	BOOL	44.1	TRUE/FALSE	FALSE	Aktuelle/letzte Bewegungsrichtung FALSE = Vorwärts (Richtung Plus) TRUE = Rückwärts (Richtung Minus)
CUTOFF	STATIC	BOOL	44.2	TRUE/FALSE	FALSE	Antrieb im Abschaltbereich (ab dem Abschaltpunkt bis zum Start der nächsten Fahrt)
CHGOVER	STATIC	BOOL	44.3	TRUE/FALSE	FALSE	Antrieb im Umschalt- bereich (ab Erreichen der Schleichgeschwindigkeit bis zum Start der nächsten Fahrt)
DIST_TO_GO	STATIC	DINT	46	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Restweg
LAST_TRG	STATIC	DINT	50	$-5x10^8$ bis $+5x10^8$ Impulse	0	Letztes/aktuelles Ziel • Schrittmaßfahrt absolut: Mit Start der Fahrt ist LST_TRG = aktuelles absolutes Ziel (TARGET). • Schrittmaßfahrt relativ: Mit Start der Fahrt ist LST_TRG = LAST_TRG der vorherigen Fahrt +/- angegebene Wegstrecke (TARGET).

Parameter für die Betriebsart "Tippen"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Tippen Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Tippen Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 1 = Tippen
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Fahrt läuft
ACT_POS	Output	DINT	16	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Referenzpunktfahrt"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Referenzpunktfahrt in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Referenzpunktfahrt in Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 3 = "Referenzpunktfahrt"
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Fahrt läuft
SYNC	Output	BOOL	14.3	TRUE/FALSE	FALSE	SYNC = TRUE: Achse ist synchronisiert
ACT_POS	Output	DINT	16	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Schrittmaßfahrt relativ"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Fahren in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Fahren in Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 4 = Schrittmaßfahrt relativ
TARGET	Input	DINT	8	0 bis 10^9 Impulse	1000	Wegstrecke in Impulsen (nur positive Werte erlaubt)
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Fahrt läuft
POS_RCD	Output	BOOL	14.1	TRUE/FALSE	FALSE	Position erreicht
ACT_POS	Output	DINT	16	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für die Betriebsart "Schrittmaßfahrt absolut"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Antriebsfreigabe
START	Input	BOOL	4.1	TRUE/FALSE	FALSE	Fahrt starten (positive Flanke)
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Fahren in Richtung Plus (positive Flanke)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Fahren in Richtung Minus (positive Flanke)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Betriebsart, 5 = Schrittmaßfahrt absolut
TARGET	Input	DINT	8	Linearachse: $-5x10^8$ bis $+5x10^8$ Rundachse: 0 bis Rund- achsenende -1	1000	Ziel in Impulsen
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Fahrt läuft
POS_RCD	Output	BOOL	14.1	TRUE/FALSE	FALSE	Position erreicht
ACT_POS	Output	DINT	16	$-5x10^8$ bis $+5x10^8$ Impulse	0	Aktueller Lageistwert
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Aktive/eingestellte Betriebsart

Parameter für den Auftrag "Bezugspunkt setzen"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
SYNC	Output	BOOL	14.3	TRUE/FALSE	FALSE	Achse ist synchronisiert

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_ID	STATIC	INT	68	1, 2	0	Auftrag, 1 = "Bezugspunkt setzen"

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_STAT	STATIC	WORD	70	W#16#0000 bis W#16#FFFF	W#16#0000	Auftragsfehler-Nummer
JOB_VAL	STATIC	DINT	72	$-5x10^8$ bis $+5x10^8$ Impulse	0	Auftragsparameter Koordinate des Bezugspunktes

Parameter für den Auftrag "Restweg löschen"

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_ID	STATIC	INT	68	1, 2	0	Auftrag, 2 = "Restweg löschen"
JOB_STAT	STATIC	WORD	70	0 bis FFFF hex	0	Auftragsfehler-Nummer
JOB_VAL	STATIC	DINT	72	-	0	Keine

Parameter für die Funktion "Längenmessung"

Die Funktion wird über eine Flanke am Digitaleingang gestartet. Es gibt keine spezifischen Eingangs-Parameter.

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
MSR_DONE	Output	BOOL	14.2	TRUE/FALSE	FALSE	Längenmessung beendet

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
BEG_VAL	STATIC	DINT	54	$-5x10^8$ bis $+5x10^8$ Impulse	0	Lageistwert Längenmessung Beginn
END_VAL	STATIC	DINT	58	$-5x10^8$ bis $+5x10^8$ Impulse	0	Lageistwert Längenmessung Ende
LEN_VAL	STATIC	DINT	62	0 bis 10^9 Impulse	0	Gemessene Länge

Parameter ERROR und STATUS

Bei erkanntem Fehler wird der Ausgangsparameter ERROR auf TRUE gesetzt. Am Parameter STATUS wird die Fehlerursache angezeigt.

STATUS Fehlercode	Erläuterung
W#16#2001	Falsche Anweisung, Anweisung "DIGITAL" verwenden.
W#16#2004	Falsche Kanalnummer (CHANNEL). Stellen Sie als Kanalnummer "0" ein.
W#16#3001	Fahrauftrag wurde nicht angenommen, da Job im gleichen Aufruf fehlerhaft. Korrigieren Sie die Parameter vom entsprechenden JOB.
W#16#3002	Eine Änderung von MODE_IN, während der Antrieb noch läuft, ist nicht erlaubt. Warten Sie, bis die laufende Positionierung beendet ist.
W#16#3003	Unbekannte Betriebsart (MODE_IN). Zulässig sind 1 (Tippen), 3 (Referenzpunktfahrt), 4 (Schrittmaßfahrt relativ) und 5 (Schrittmaßfahrt absolut).
W#16#3004	Es darf immer nur eine Startanforderung gleichzeitig gesetzt sein. Zulässige Startartanforderungen sind DIR_P oder DIR_M oder START.
W#16#3005	START nur in Betriebsart "Schrittmaßfahrt absolut" erlaubt. Starten Sie die Fahrt mit DIR_P oder DIR_M.
W#16#3006	DIR_P oder DIR_M bei Linearachse und Betriebsart "Schrittmaßfahrt absolut" nicht erlaubt. Starten Sie die Fahrt mit START.
W#16#3007	Achse nicht synchronisiert. "Schrittmaßfahrt absolut" ist nur bei synchronisierter Achse möglich.
W#16#3008	Arbeitsbereich verlassen. Fahrt nur per Tippen zurück in Richtung des Arbeitsbereichs erlaubt.
W#16#3101	Keine Startfreigabe, da die Achse nicht parametriert ist. Parametrieren Sie das Submodul "Positionieren" über die Hardware-Konfiguration.
W#16#3102	Keine Startfreigabe, da keine Antriebsfreigabe gesetzt ist. Setzen Sie die "Antriebsfreigabe" an der Anweisung (DRV_EN=TRUE).
W#16#3103	Keine Startfreigabe, da STOP gesetzt ist. Löschen Sie STOP an der Anweisung (STOP=FALSE).
W#16#3104	Keine Startfreigabe, da die Achse momentan positioniert (WORKING=TRUE). Warten Sie, bis die laufende Positionierung beendet ist.
W#16#3105	Keine Startfreigabe, da noch mindestens ein nicht quittierter externer Fehler ansteht. Beseitigen und quittieren Sie erst alle externen Fehler und Starten sie dann die Fahrt neu.
W#16#3201	Geschwindigkeitsvorgabe SPEED falsch. Beim Positionieren mit Digitalausgängen ist nur "Schleichgang" (0) und "Eilgang" (1) zulässig.
W#16#3301	Um-/Abschaltdifferenz zu groß. Um-/Abschaltdifferenz maximal gleich 10^8 vorgeben
W#16#3303	Umschaltdifferenz zu klein. Die Umschaltdifferenz muss größer/gleich der Abschaltdifferenz sein.
W#16#3304	Abschaltdifferenz zu klein. Die Abschaltdifferenz muss mindestens so groß sein wie der halbe Zielbereich.
W#16#3401	Zielvorgabe außerhalb des Arbeitsbereichs. Bei einer Linearachse und Schrittmaßfahrt absolut muss die Zielvorgabe innerhalb der Softwareendschalter (inklusive) sein.
W#16#3402	Zielvorgabe falsch. Bei einer Rundachse muss die Zielvorgabe größer "0" und kleiner als das Rundachsenende sein.
W#16#3403	Wegangabe falsch. Das zu verfahrende Wegstück bei Schrittmaßfahrt relativ muss positiv sein.

STATUS Fehlercode	Erläuterung
W#16#3404	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss größer als $-5x10^8$ sein.
W#16#3405	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss kleiner als $5x10^8$ sein.
W#16#3406	Wegangabe falsch. Die sich ergebende absolute Zielkoordinate muss innerhalb des Arbeitsbereichs (+/-halber Zielbereich) liegen.
W#16#3501	Verfahrweg zu groß. Zielkoordinate + aktueller Restweg muss größer/gleich $-5x10^8$ sein.
W#16#3502	Verfahrweg zu groß. Zielkoordinate + aktueller Restweg muss kleiner/gleich $5x10^8$ sein.
W#16#3503	Verfahrweg zu klein. Der Verfahrweg in Richtung Plus muss größer als die angegebene Abschaltdifferenz für die Richtung Plus sein.
W#16#3504	Verfahrweg zu klein. Der Verfahrweg in Richtung Minus muss größer als die angegebene Abschaltdifferenz für die Richtung Minus sein.
W#16#3505	Verfahrweg zu klein oder Endschalter bereits in Richtung Plus überfahren. Letztes anfahrbares Ziel in Richtung Plus (Arbeitsbereich- oder Verfahrbereichsgrenze) liegt zu nahe an der aktuellen Position.
W#16#3506	Verfahrweg zu klein oder Endschalter bereits in Richtung Minus überfahren. Letztes anfahrbares Ziel in Richtung Minus (Arbeitsbereich- oder Verfahrbereichsgrenze) liegt zu nahe an der aktuellen Position.

Auftragsfehler (JOB_ERR = TRUE)

Bei erkanntem Fehler wird der Parameter JOB_ERR auf TRUE gesetzt. Am Parameter JOB_STAT wird die Fehlerursache angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#4001	Achse nicht parametriert. Parametrieren Sie das Submodul "Positionieren" über die Hardware-Konfiguration.
W#16#4002	Auftrag nicht möglich, da noch eine Positionierung läuft. Aufträge sind nur ausführbar, wenn keine Positionierung läuft. Warten Sie, bis WORKING = FALSE wird, und führen Sie den Auftrag erneut aus.
W#16#4004	Unbekannter Auftrag. Prüfen Sie die Auftragsnummer und führen Sie den Auftrag erneut aus.
W#16#4101	Bei einer Linearachse darf die Bezugspunktkoordinate nicht außerhalb der Arbeitsbereichsgrenzen liegen.
W#16#4102	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktueller Restweg noch größer/gleich $-5x10^8$ sein.
W#16#4103	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktueller Restweg noch kleiner/gleich $5x10^8$ sein.
W#16#4104	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktuelle Differenz zum Startpunkt der Fahrt noch größer/gleich $-5x10^8$ sein.
W#16#4105	Bei einer Linearachse muss die angegebene Bezugspunktkoordinate + aktuelle Differenz zum Startpunkt der Fahrt noch kleiner/gleich $5x10^8$ sein.
W#16#4106	Bei einer Rundachse darf die Bezugspunktkoordinate nicht kleiner 0 und größer/gleich dem Rundachsenende sein.

Externfehler (ERR)

Von der Technologie werden Überwachungen bezüglich der Fahrt, des Verfahrbereichs und der angeschlossenen Peripherie durchgeführt. Voraussetzung ist, dass Sie die Überwachungen zuvor in den Parametriermasken "Antrieb", "Achse" und "Geber" eingeschaltet haben.

Bei Ansprechen der Überwachungen wird ein Externfehler gemeldet. Externfehler können unabhängig von gestarteten Funktionen jederzeit auftreten. Externfehler müssen Sie immer mit ERR_A = TRUE quittieren.

Die Externfehler werden am Parameter ERR (WORD) durch Setzen eines Bits angezeigt.

Überwachung	Fehlercode	Bit im ERR-WORD
Fehlimpuls (Nullmarke)	W#16#0004	2
Verfahrbereich	W#16#0800	11
Arbeitsbereich	W#16#1000	12
Istwert	W#16#2000	13
Zieleinlauf	W#16#4000	14
Zielbereich	W#16#8000	15

Systemfehler

Ein Systemfehler wird mit BIE = FALSE angezeigt. Ein Systemfehler wird ausgelöst durch Fehler beim Schreiben/Lesen des Instanz-DB oder Mehrfachaufruf der Anweisung.

COUNT: Zähler steuern

Beschreibung

Zur Steuerung des Zählers aus dem Anwenderprogramm verwenden Sie die Anweisung "COUNT".

Folgende Funktionalität steht Ihnen zur Verfügung:

- Starten/Stoppen des Zählers mit dem Softwaretor SW_GATE
- Freigabe/Steuern des Ausgangs DO
- Auslesen von Statusbits STS_CMP, STS_OFLW, STS_UFLW und STS_ZP
- Auslesen des aktuellen Zählerstands COUNTVAL
- Aufträge zum Lesen und Schreiben der internen Zählregister
- Auslesen der aktuellen Periodendauer TIMEVAL

Parameter

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
LADDR	Input	WORD	0	CPU-spezifisch	W#16#0300	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben. Sind E- und A-Adresse nicht gleich, ist die kleinere der beiden Adressen anzugeben.
CHANNEL	Input	INT	2	CPU 312C: 0 bis 1 CPU 313C: 0 bis 2 CPU 314C: 0 bis 3	0	Kanalnummer
SW_GATE	Input	BOOL	4.0	TRUE/FALSE	FALSE	Softwaretor zum Starten/ Stoppen des Zählers
CTRL_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Freigabe Ausgang
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Steuern Ausgang
JOB_REQ	Input	BOOL	4.3	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_ID	Input	WORD	6	W#16#0000 Auftrag ohne Funktion W#16#0001 Zählwert schreiben W#16#0002 Ladewert schreiben W#16#0004 Vergleichswert schreiben W#16#0008 Hysterese schreiben W#16#0010 Impulsdauer schreiben W#16#0082 Ladewert lesen W#16#0084 Vergleichswert lesen W#16#0088 Hysterese lesen W#16#0090 Impulsdauer lesen	W#16#0000	Auftragsnummer
JOB_VAL	Input	DINT	8	-2^{31} bis $+2^{31}$ -1	0	Wert für schreibende Aufträge.
STS_GATE	Output	BOOL	12.0	TRUE/FALSE	FALSE	Status internes Tor
STS_STRT	Output	BOOL	12.1	TRUE/FALSE	FALSE	Status Hardware-Tor (Starteingang)
STS_LTCH	Output	BOOL	12.2	TRUE/FALSE	FALSE	Status Latcheingang
STS_DO	Output	BOOL	12.3	TRUE/FALSE	FALSE	Status Ausgang

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
STS_C_DN	Output	BOOL	12.4	TRUE/FALSE	FALSE	Status Richtung rückwärts. Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf der Anweisung hat STS_C_DN den Wert FALSE.
STS_C_UP	Output	BOOL	12.5	TRUE/FALSE	FALSE	Status Richtung vorwärts Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf der Anweisung hat STS_C_UP den Wert TRUE
COUNTVAL	Output	DINT	14	-2^{31} bis $+2^{31}-1$	0	Aktueller Zählwert
LATCHVAL	Output	DINT	18	-2^{31} bis $+2^{31}-1$	0	Aktueller Latchwert
JOB_DONE	Output	BOOL	22.0	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	Output	BOOL	22.1	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_STAT	Output	WORD	24	0 bis W#16#FFFF	0	Auftragsfehler-Nummer

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Wenn Sie den über die Parametrieroberfläche einstellbaren Parameter "Verhalten des Ausgangs" auf "kein Vergleich" eingestellt haben, gilt:

- Der Ausgang wird wie ein normaler Ausgang geschaltet.
- Die Eingangsparameter CTRL_DO und SET_DO sind unwirksam.
- Die Statusbits STS_DO und STS_CMP (Status Vergleicher im IDB) bleiben rückgesetzt

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	Status Vergleicher. Wird mit RES_STS zurückgesetzt. Das Statusbit STS_CMP zeigt an, dass die Vergleichsbedingung des Komparators erfüllt ist oder erfüllt war. Mit STS_CMP wird auch angezeigt, dass der Ausgang gesetzt war (STS_DO = TRUE)
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	Status Überlauf Wird mit RES_STS zurückgesetzt.
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	Status Unterlauf Wird mit RES_STS zurückgesetzt.
STS_ZP	STATIC	BOOL	26.7	TRUE/FALSE	FALSE	Status Nulldurchgang Wird mit RES_STS zurückgesetzt. Wird nur gesetzt bei Zählen ohne Hauptzählrichtung. Zeigt Nulldurchgang an. Wird auch gesetzt, wenn der Zähler auf 0 gesetzt wird oder der Zähler ab Ladewert=0 zählt.
JOB_OVAL	STATIC	DINT	28	-2^{31} bis $+2^{31}$ -1	0	Ausgabewert für Leseaufträge.
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	Statusbits rücksetzen. Setzt die Status-Bits STS_CMP, STS_OFLW, STS_UFLW und STS_ZP zurück. Zum Rücksetzen der Statusbits werden zwei Aufrufe der Anweisung benötigt.

Parameter JOB_ERR und JOB_STAT

Ist ein Auftragsfehler aufgetreten, wird JOB_ERR = TRUE gesetzt. Die genaue Fehlerursache wird dann in JOB_STAT angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#0121	Vergleichswert zu klein.
W#16#0122	Vergleichswert zu groß.
W#16#0131	Hysterese zu klein.
W#16#0132	Hysterese zu groß.
W#16#0141	Impulsdauer zu klein.
W#16#0142	Impulsdauer zu groß.

Ereignisklasse Fehlercode	Erläuterung
W#16#0151	Ladewert zu klein .
W#16#0152	Ladewert zu groß.
W#16#0161	Zählstand zu klein .
W#16#0162	Zählstand zu groß.
W#16#01FF	Auftragsnummer ungültig.

Systemfehler

Ist ein Systemfehler aufgetreten, wird BIE = FALSE gesetzt.

Ereignisklasse Fehlercode	Erläuterung
W#16#8001	Falsche Betriebsart oder Parametrierfehler. Stellen Sie bei der Hardware-Konfiguration die richtige Betriebsart ein oder benutzen Sie die zur eingestellten Betriebsart passende Anweisung.
W#16#8009	Kanalnummer ungültig. Stellen Sie eine Kanalnummer ≤ 3 ein (CPU-spezifischer Wert).

FREQUENC: Frequenzmesser steuern

Beschreibung

Die Anweisung "FREQUENC" verwenden Sie, um den Frequenzmesser aus dem Anwenderprogramm zu bedienen.

Folgende Funktionalität steht Ihnen zur Verfügung:

- Starten/Stoppen mit dem Softwaretor SW_GATE
- Freigabe/Steuern des Ausgangs DO
- Auslesen der Statusbits STS_CMP, STS_OFLW und STS_UFLW
- Auslesen des aktuellen Frequenzwertes MEAS_VAL
- Aufträge zum Lesen und Schreiben der internen Frequenzmessregister

Parameter

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
LADDR	Input	WORD	0	CPU-spezifisch	W#16#0300	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben. Sind E- und A-Adresse nicht gleich, ist die kleinere der beiden Adressen anzugeben.
CHANNEL	Input	INT	2	CPU 312C: 0 bis 1 CPU 313C: 0 bis 2 CPU 314C: 0 bis 3	0	Kanalnummer
SW_GATE	Input	BOOL	4.0	TRUE/FALSE	FALSE	Softwaretor zum Starten/ Stoppen der Frequenzmessung
MAN_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Freigabe manuelle Steuerung des Ausgangs
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Steuern Ausgang
JOB_REQ	Input	BOOL	4.3	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_ID	Input	WORD	6	W#16#0000 = Auftrag ohne Funktion W#16#0001 = Untergrenze schreiben W#16#0002 = Obergrenze schreiben W#16#0004 = Integrationszeit schreiben W#16#0081 = Untergrenze lesen W#16#0082 = Obergrenze lesen W#16#0084 = Integrationszeit lesen	0	Auftragsnummer
JOB_VAL	Input	DINT	8	-2^{31} bis $+2^{31}-1$	0	Wert für schreibende Aufträge
STS_GATE	Output	BOOL	12.0	TRUE/FALSE	FALSE	Status internes Tor
STS_STRT	Output	BOOL	12.1	TRUE/FALSE	FALSE	Status Hardware-Tor (Starteingang)

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
STS_DO	Output	BOOL	12.2	TRUE/FALSE	FALSE	Status Ausgang
STS_C_DN	Output	BOOL	12.3	TRUE/FALSE	FALSE	Status Richtung rückwärts Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf der Anweisung hat STS_C_DN den Wert FALSE.
STS_C_UP	Output	BOOL	12.4	TRUE/FALSE	FALSE	Status Richtung vorwärts Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf der Anweisung hat STS_C_UP den Wert TRUE.
MEAS_VAL	Output	DINT	14	0 bis $+2^{31}$ -1	0	Aktueller Frequenzwert
COUNTVAL	Output	DINT	18	-2^{31} bis $+2^{31}$ -1	0	Aktueller Zählwert (Startet bei jedem Öffnen des Tors bei 0)
JOB_DONE	Output	BOOL	22.0	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	Output	BOOL	22.1	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_STAT	Output	WORD	24	W#16#0000 bis W#16#FFFF	W#16#0000	Auftragsfehler-Nummer

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Wenn Sie den über die Parametrieroberfläche einstellbaren Parameter "Verhalten des Ausgangs" auf "kein Vergleich" eingestellt haben, gilt:

- Der Ausgang wird wie ein normaler Ausgang geschaltet.
- Die Eingangsparameter MAN_DO und SET_DO sind unwirksam.
- Das Statusbit STS_DO bleibt rückgesetzt.

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten ohne die reservierten Parameter RESxx):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	Status Messende Wird mit RES_STS zurückgesetzt. Nach jedem abgelaufenen Zeitintervall wird der Messwert aktualisiert. Dabei wird das Messende einer Messung mit dem Statusbit STS_CMP gemeldet
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	Status Überlauf Wird mit RES_STS zurückgesetzt.
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	Status Unterlauf Wird mit RES_STS zurückgesetzt.
JOB_OVAL	STATIC	DINT	28	-2^{31} bis 2^{31} -1	0	Ausgabewert für Leseaufträge
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	Statusbits rücksetzen. Setzt die Status-Bits STS_CMP, STS_OFLW und STS_UFLW zurück. Zum Rücksetzen der Statusbits werden zwei Aufrufe der Anweisung benötigt.

Parameter JOB_ERR und JOB_STAT

Ist ein Auftragsfehler aufgetreten, wird JOB_ERR = TRUE gesetzt. Die genaue Fehlerursache wird dann in JOB_STAT angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#0221	Integrationszeit zu klein.
W#16#0222	Integrationszeit zu groß.
W#16#0231	Frequenzuntergrenze zu klein.
W#16#0232	Frequenzuntergrenze zu groß.
W#16#0241	Frequenzobergrenze zu klein.
W#16#0242	Frequenzobergrenze zu groß.
W#16#02FF	Auftragsnummer ungültig.

Systemfehler

Ist ein Systemfehler aufgetreten, wird BIE = FALSE gesetzt.

Ereignisklasse Fehlercode	Erläuterung
W#16#8001	Falsche Betriebsart oder Parametrierfehler. Stellen Sie bei der Hardware-Konfiguration die richtige Betriebsart ein oder benutzen Sie die zur eingestellten Betriebsart passende Anweisung
W#16#8009	Kanalnummer ungültig. Stellen Sie eine Kanalnummer ≤ 3 ein (CPU-spezifischer Wert).

PULSE: Pulsweitenmodulation steuern

Beschreibung

Zur Steuerung der Pulsweitenmodulation aus dem Anwenderprogramm verwenden Sie die Anweisung "PULSE".

Folgende Funktionalität steht Ihnen zur Verfügung:

- Starten/Stoppen mit dem Softwaretor SW_EN
- Freigabe/Steuern des Ausgangs DO
- Auslesen der Statusbits STS_EN, STS_STRT und STS_DO
- Eingeben des Ausgabewertes
- Aufträge zum Lesen und Schreiben der Register

Parameter

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Bedeutung
LADDR	Input	WORD	0	CPU-spezifisch	W#16#0300	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben. Sind E- und A-Adresse nicht gleich, ist die kleinere der beiden Adressen anzugeben.
CHANNEL	Input	INT	2	CPU 312C: 0 bis 1 CPU 313C: 0 bis 2 CPU 314C: 0 bis 3	0	Kanalnummer
SW_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Softwaretor zum Starten/Stoppen der Ausgabe
MAN_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Freigabe manuelle Steuerung des Ausgangs

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Bedeutung
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Steuern Ausgang
OUTP_VAL	Input	INT	6.0	in Promille: 0 bis 1000 als S7- Analog-wert: 0 bis 27648	0	Vorgabe des Ausgabewertes Geben Sie einen Ausgabe-wert > 1 000 bzw. 27648 vor, begrenzt die CPU diesen auf 1 000 bzw. 27648
JOB_REQ	Input	BOOL	8.0	TRUE/FALSE	FALSE	Auftragsanstoß (positive Flanke)
JOB_ID	Input	WORD	10	W#16#0000 = Auftrag ohne Funktion W#16#0001 = Pe-riodendauer schreiben W#16#0002 = Einschaltverzö-gerung schreiben W#16#0004 = Mindestimpuls-dauer schreiben W#16#0081 = Pe-riodendauer lesen W#16#0082 = Einschaltverzö-gerung lesen W#16#0084 = Mindestimpuls-dauer lesen	W#16#0000	Auftragsnummer
JOB_VAL	Input	DINT	12	-2^{31} bis$+2^{31}$ -1	0	Wert für schreibende Aufträ-ge.
STS_EN	Output	BOOL	16.0	TRUE/FALSE	FALSE	Status der Freigabe
STS_STRT	Output	BOOL	16.1	TRUE/FALSE	FALSE	Status Hardware-Tor (Start-eingang)
STS_DO	Output	BOOL	16.2	TRUE/FALSE	FALSE	Status Ausgang
JOB_DONE	Output	BOOL	16.3	TRUE/FALSE	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	Output	BOOL	16.4	TRUE/FALSE	FALSE	Auftrag fehlerhaft
JOB_STAT	Output	WORD	18	W#16#0000 bis W#16#FFFF	W#16#0000	Auftragsfehler-Nummer

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Nicht am Baustein verschaltete Parameter (Statische Lokaldaten):

Parameter	Deklaration	Datentyp	Adresse (Instanz-DB)	Wertebereich	Vorbelegung	Beschreibung
JOB_OVAL	Output	DINT	20	-2^{31} bis 2^{31} -1	0	Ausgabewert für Leseaufträge

Parameter JOB_ERR und JOB_STAT

Ist ein Auftragsfehler aufgetreten, wird JOB_ERR = TRUE gesetzt. Die genaue Fehlerursache wird dann in JOB_STAT angezeigt.

Ereignisklasse Fehlercode	Erläuterung
W#16#0411	Periodendauer zu klein.
W#16#0412	Periodendauer zu groß.
W#16#0421	Einschaltverzögerung zu klein.
W#16#0422	Einschaltverzögerung zu groß.
W#16#0431	Mindestimpulsdauer zu klein.
W#16#0432	Mindestimpulsdauer zu groß.
W#16#04FF	Auftragsnummer ungültig.

Systemfehler

Ist ein Systemfehler aufgetreten, wird BIE = FALSE gesetzt.

Ereignisklasse Fehlercode	Erläuterung
W#16#8001	Falsche Betriebsart oder Parametrierfehler. Stellen Sie bei der Hardware-Konfigurierung die richtige Betriebsart ein oder benutzen Sie die zur eingestellten Betriebsart passende Anweisung
W#16#8009	Kanalnummer ungültig. Stellen Sie eine Kanalnummer ≤ 3 ein (CPU-spezifischer Wert).

4.2.5 Kommunikation

4.2.5.1 Gemeinsame Parameter- und Fehlerinformationen

Gemeinsame Parameter der Anweisungen zur S7-Basiskommunikation

Parameter REQ

Der Eingangsparameter REQ (request to activate) ist ein pegelgetriggerter Steuerparameter. Er dient dazu, den Auftrag (die Datenübertragung bzw. den Verbindungsabbruch) anzustoßen:

- Wenn Sie die Anweisung zu einem Auftrag aufrufen, der momentan nicht aktiviert ist, so stoßen Sie mit REQ=1 den Auftrag an. Besteht zum Zeitpunkt des Erstaufrufs einer Anweisung noch keine Verbindung zum Kommunikationspartner, dann wird diese vor Beginn der Datenübertragung aufgebaut.
- Wenn Sie einen Auftrag angestoßen haben und dieser noch nicht abgeschlossen ist und Sie die Anweisung zum gleichen Auftrag erneut aufrufen, so wird REQ durch die Anweisung nicht ausgewertet.

Parameter REQ_ID (nur "X_SEND" und "X_RCV")

Der Eingangsparameter REQ_ID dient der Kennzeichnung Ihrer Sendedaten. Er wird vom Betriebssystem der sendenden CPU an die Anweisung "X_RCV (Seite 7078)" der CPU des Kommunikationspartners weitergereicht.

Sie benötigen auf Empfängerseite den Parameter REQ_ID,

- wenn Sie auf einer Sende-CPU mehrere Anweisungen "X_SEND (Seite 7077)" mit unterschiedlichen Parametern REQ_ID aufrufen, und die Daten an einen Kommunikationspartner übertragen.
- wenn Sie von mehreren Sende-CPUs Daten mit Hilfe der Anweisung "X_SEND (Seite 7077)" an einen Kommunikationspartner übertragen.

Sie können durch Auswertung von REQ_ID die Empfangsdaten in unterschiedlichen Speicherbereichen ablegen.

Parameter RET_VAL und BUSY

Die Anweisungen zur S7-Basiskommunikation sind asynchron laufende Anweisungen, d. h. die Abwicklung eines Auftrags erstreckt sich über mehrere Aufrufe. Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Parameter CONT

Der Eingangsparameter CONT (continue) ist ein Steuerparameter. Mit ihm bestimmen Sie, ob die Verbindung zum Kommunikationspartner nach Abschluss des Auftrags bestehen bleibt.

- Wenn Sie beim Erstaufruf CONT=0 wählen, wird die Verbindung nach Beendigung der Datenübertragung wieder abgebaut. Sie steht dann für den Datenaustausch zu einem neuen Kommunikationspartner wieder zur Verfügung.
  Mit dieser Vorgehensweise stellen Sie sicher, dass nur Verbindungsressourcen belegt sind, die aktuell benötigt werden.
- Wenn Sie beim Erstaufruf CONT=1 wählen, bleibt die Verbindung nach Beendigung der Datenübertragung bestehen.
  Diese Vorgehensweise bietet sich z. B. dann an, wenn Sie zyklisch zwischen zwei Stationen Daten austauschen.

Hinweis

Eine über CONT=1 aufgebaute Verbindung können Sie auch explizit mit "X_ABORT (Seite 7087)" bzw. mit "I_ABORT (Seite 7070)" abbrechen.

Fehlerinformationen der Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen

Anzeigen

Die in der folgenden Tabelle u. a. angegebenen "echten" Fehlerinformationen für die Anweisungen "X_SEND (Seite 7077)", "X_RCV (Seite 7078)", "X_GET (Seite 7083)", "X_PUT (Seite 7085)", "X_ABORT (Seite 7087)", "I_PUT (Seite 7068)", "I_GET (Seite 7066)", und "I_ABORT (Seite 7070)" können Sie wie folgt klassifizieren:

Fehlercode (W#16# ...)	Erläuterung
809x	Fehler auf der CPU, auf der die Anweisung abläuft
80Ax	Permanenter Kommunikationsfehler
80Bx	Fehler beim Kommunikationspartner
80Cx	Temporärer Fehler

Die folgende Tabelle enthält spezifische Fehlerinformationen für die Anweisungen.

Fehlercode (W#16# ...)	Erläuterung (allgemein)	Erläuterung (Anweisungs-spezifisch)
0000	Die Bearbeitung wurde fehlerfrei abgeschlossen.	"X_ABORT (Seite 7077)" und "I_ABORT (Seite 7070)": REQ=1, und die angegebene Verbindung ist nicht aufgebaut.
		"X_RCV (Seite 7078)": EN_DT=1 und RD=NIL

Fehlercode (W#16# ...)	Erläuterung (allgemein)	Erläuterung (Anweisungs-spezifisch)
00xy	-	"X_RCV (Seite 7078)" bei NDA=1 und RD<>NIL: RET_VAL enthält die Länge des empfangenen (bei EN_DT=0) bzw. des in RD kopierten Datenblocks (bei EN_DT=1).
		"X_GET (Seite 7083)": RET_VAL enthält die Länge des empfangenen Datenblocks.
		"I_GET (Seite 7066)": RET_VAL enthält die Länge des empfangenen Datenblocks.
7000	-	"X_SEND (Seite 7077)", "X_GET (Seite 7083)", "X_PUT (Seite 7085)", "X_ABORT (Seite 7087)", "I_GET (Seite 7066)", "I_PUT (Seite 7068)" und "I_ABORT (Seite 7070)": Aufruf mit REQ=0 (Aufruf ohne Bearbeitung), BUSY hat den Wert "0", es ist keine Datenübertragung aktiv.
		"X_RCV (Seite 7078)": EN_DT=0/1 und NDA=0
7001	Erstaufruf mit REQ=1: Datenübertragung wurde angestoßen; BUSY hat den Wert "1".	-
7002	Zwischenaufruf (REQ irrelevant): Datenübertragung ist bereits aktiv; BUSY hat den Wert "1".	"X_ABORT (Seite 7087)" und "I_ABORT (Seite 7070)": Zwischenaufruf mit REQ=1
8090	Angegebene Zieladresse des Kommunikationspartners ist ungültig, z. B. • falsche IOID • falsche Basisadresse vorhanden • falsche MPI-Adresse (> 126)	-

Fehlercode (W#16# ...)	Erläuterung (allgemein)	Erläuterung (Anweisungs-spezifisch)
8092	Fehler bei SD oder RD, z. B.: Die Adressierung des Lokaldatenbereichs ist nicht zulässig.	"X_SEND (Seite 7077)", z. B. • unzulässige Länge von SD • SD=NIL ist unzulässig
		"X_RCV (Seite 7078)", z. B. • Es wurden mehr Daten empfangen als in den durch RD vorgegebenen Bereich hineinpassen. • RD ist vom Datentyp BOOL, die Länge der empfangenen Daten ist jedoch größer als ein Byte.
		"X_GET (Seite 7083)" und "I_GET (Seite 7066)", z. B. • unzulässige Länge bei RD • Die Länge oder der Datentyp von RD stimmt mit den empfangenen Daten nicht überein. • RD=NIL ist unzulässig.
		"X_PUT (Seite 7085)" und "I_PUT (Seite 7068)", z. B. • unzulässige Länge bei SD • SD=NIL ist unzulässig
8095	Der Baustein wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.	
80A0	Fehler in empfangener Quittung	"X_PUT (Seite 7085)" und "I_PUT (Seite 7068)": Der in SD der sendenden CPU angegebene Datentyp wird vom Kommunikationspartner nicht unterstützt.
80A1	Kommunikationsprobleme: Aufruf der Anweisung nach Abbruch einer bestehenden Verbindung	"AS_DIAL (Seite 7093)": • Die Bearbeitung einer Kommunikations-Anweisung wurde vom lokalen TS Adapter abgelehnt, weil keine Fernverbindung durch "AS_DIAL" eingerichtet ist. • Während der Bearbeitung der Kommunikations-Anweisung ist die Fernverbindung zum entfernten TS Adapter abgebrochen.
80B0	Objekt ist nicht erreichbar, z. B. DB nicht geladen	möglich bei "X_GET (Seite 7083)" und "X_PUT (Seite 7085)" und "I_GET (Seite 7066)" und "I_PUT (Seite 7068)"
80B1	Fehler im ANY-Pointer. Die Länge des zu übertragenden Datenbereichs ist falsch.	-

Fehlercode (W#16# ...)	Erläuterung (allgemein)	Erläuterung (Anweisungs-spezifisch)
80B2	HW-Fehler: Baugruppe nicht vorhanden • Der projektierte Steckplatz ist nicht belegt. • Ist-Baugruppentyp ungleich Soll-Baugruppentyp • Dezentrale Peripherie ist nicht verfügbar. • Im zugehörigen SDB ist kein Eintrag für die Baugruppe vorhanden.	Möglich bei "X_GET (Seite 7083)" und "X_PUT (Seite 7085)" und "I_GET (Seite 7066)" und "I_PUT (Seite 7068)".
80B3	Daten dürfen entweder nur gelesen oder nur geschrieben werden, z. B. schreibgeschützter DB	Möglich bei "X_GET (Seite 7083)" und "X_PUT (Seite 7085)" und "I_GET (Seite 7066)" und "I_PUT (Seite 7068)".
80B4	Datentypfehler bei ANY-Pointer, oder ARRAY des angegebenen Datentyps nicht erlaubt.	"X_GET (Seite 7083)" und "X_PUT (Seite 7085)" und "I_GET (Seite 7066)" und "I_PUT (Seite 7068)": Der in VAR_ADDR angegebene Datentyp wird vom Kommunikationspartner nicht unterstützt.
80B5	Bearbeitung wegen unzulässigem Betriebszustand abgelehnt	Möglich bei "X_SEND (Seite 7077)". Bei "AS_DIAL (Seite 7093)": • Die Bearbeitung einer Kommunikations-Anweisung wurde vom lokalen TS Adapter abgelehnt, weil die Funktion DIAL von "AS_DIAL" noch nicht abgeschlossen ist. • Die Funktion DIAL des "AS_DIAL" wurde angefordert, obwohl bereits eine Fernverbindung für AS-AS-Fernkopplung eingerichtet ist.
80B6	In der empfangenen Quittung steht ein unbekannter Fehlercode.	-
80B7	Datentyp und/oder Länge der übertragenen Daten passen nicht zum Bereich in der Partner-CPU, in den geschrieben werden soll.	Möglich bei "X_PUT (Seite 7085)" und "I_PUT (Seite 7068)"
80B8	-	"X_SEND (Seite 7077)": "X_RCV (Seite 7078)" des Kommunikationspartners hat die Datenübernahme abgelehnt (RD=NIL).
80B9	-	"X_SEND (Seite 7077)": Der Datenblock wurde vom Kommunikationspartner identifiziert (Aufruf von "X_RCV (Seite 7078)" mit EN_DT=0), jedoch wegen Betriebszustand STOP noch nicht in das Anwenderprogramm übernommen.
80BA	Die Antwort des Kommunikationspartners passt nicht in das Kommunikationstelegramm.	-
80C0	Die angegebene Verbindung ist durch einen anderen Auftrag bereits belegt.	-

Fehlercode (W#16# ...)	Erläuterung (allgemein)	Erläuterung (Anweisungs-spezifisch)
80C1	Ressourcenengpass bei der CPU, auf der die Anweisung abläuft, z. B.: • Die maximale Anzahl verschiedener Sendeaufträge der Baugruppe wird bereits bearbeitet. • Die Verbindungsressource ist z. B. durch Empfang belegt.	-
80C2	Temporärer Ressourcenmangel des Kommunikationspartners, z. B.: • Der Kommunikationspartner bearbeitet momentan das Maximum an Aufträgen. • Die benötigten Betriebsmittel (Speicher etc.) sind belegt. • Zu wenig Arbeitsspeicher vorhanden. (Stoßen Sie einen Komprimiervorgang an.)	Bei "AS_DIAL (Seite 7093)": • Temporärer Ressourcenmangel auf der entfernten CPU. • Die entfernte CPU mit der MPI-Adresse ist nicht verfügbar oder existiert nicht.
80C3	Fehler beim Verbindungsaufbau, z. B.: • Die eigene S7-Station befindet sich nicht am MPI-Subnetz. • Sie haben die eigene Station am MPI-Subnetz adressiert. • Der Kommunikationspartner ist nicht mehr erreichbar. • Temporärer Ressourcenmangel des Kommunikationspartners	-

4.2.5.2 S7-Kommunikation

Übersicht über die Anweisungen zur S7-Kommunikation

Klassifikation

Für die S7-Kommunikation ist eine Verbindungsprojektierung erforderlich. Der Aufruf der integrierten Kommunikationsfunktionen erfolgt im Anwenderprogramm über Anweisungen.

Diese Anweisungen lassen sich in die folgenden Klassen einteilen:

- Anweisungen zum Datenaustausch
- Anweisungen zur Betriebszustands-Änderung

- Anweisungen zur Betriebszustands-Abfrage
- Je eine Anweisung zur Verbindungs-Abfrage

Hinweis

Falls Ihre S7-300-CPU keine Ethernet-Schnittstelle hat, benötigen Sie zum Ablauf der S7-300-Anweisungen einen SIMATIC NET CP (Seite 6770) des S7-300-Spektrums. Weitere Informationen hierzu finden Sie in der zugehörigen Dokumentation.

Anweisungen zum Datenaustausch

Kommunikations-Anweisungen zum Datenaustausch dienen dem Datenaustausch zwischen zwei Kommunikationspartnern. Abhängig davon, ob es nur auf der lokalen Baugruppe oder sowohl auf der lokalen als auch auf der remoten Baugruppe eine Kommunikations-Anweisung gibt, spricht man von einseitigem oder zweiseitigem Datenaustausch.

Anweisung	Kurzbeschreibung
"USEND (Seite 6398)" / "URCV (Seite 6400)"	Schnelles, unquittiertes Übertragen von Daten unabhängig von der zeitlichen Bearbeitung der Kommunikationsfunktion ("URCV") beim Kommunikationspartner (z. B. Betriebs- und Wartungsmeldungen). Das bedeutet, die Daten können beim Kommunikationspartner durch aktuellere Daten überschrieben werden.
"BSEND (Seite 6403)" / "BRCV (Seite 6406)"	Sicheres Übertragen eines Datenblocks zum Kommunikationspartner. Das heißt, die Datenübertragung ist erst dann abgeschlossen, wenn die Empfangsfunktion ("BRCV") im Kommunikationspartner die Daten übernommen hat.
"GET (Seite 6392)"	Programmgesteuertes Lesen von Variablen ohne zusätzliche Kommunikationsfunktion im Anwenderprogramm des Kommunikationspartners.
"PUT (Seite 6395)"	Programmgesteuertes Schreiben von Variablen ohne zusätzliche Kommunikationsfunktion im Anwenderprogramm des Kommunikationspartners.
"PRINT (Seite 6423)"	Daten an einen Drucker senden (nur S7-400: CP441).

Anweisungen zur Betriebszustands-Änderung

Mit Anweisungen zur Betriebszustands-Änderung steuern Sie den Betriebszustand eines remoten Geräts.

Bei den Anweisungen zur Betriebszustands-Änderung ist die Art des Datenaustauschs einseitig.

Anweisung	Kurzbeschreibung
"START (Seite 6429)"	Den NEUSTART einer S7-300/400-CPU auslösen, wenn sich diese im Betriebszustand STOP befindet.
"STOP (Seite 6431)"	STOP einer S7-300/400-CPU, wenn sich diese im Betriebszustand RUN, HALT oder Anlauf befindet
"RESUME (Seite 6433)"	Wiederanlauf einer S7-400-CPU auslösen, wenn sich diese im Betriebszustand STOP befindet.

Anweisungen zur Betriebszustands-Abfrage

Mit Anweisungen zur Betriebszustands-Abfrage können Sie Informationen über den Betriebszustand eines remoten Geräts empfangen.

Bei der Anweisung "STATUS" ist die Art des Datenaustauschs einseitig, bei der Anweisung "USTATUS (Seite 6437)" zweiseitig.

Anweisung	Kurzbeschreibung
"STATUS (Seite 6435)"	Liefert den Betriebszustand eines Kommunikationspartners (S7-400-CPU) auf Anwenderabfrage.
"USTATUS (Seite 6437)"	Empfängt den Betriebszustand einer S7-400-CPU bei deren Betriebszustandswechsel, falls das entsprechende Verbindungsattribut (Betriebszustandsmeldungen Senden) gesetzt ist.

Anweisung zur Verbindungs-Abfrage

Anweisung S7-400	Anweisung S7-300	Kurzbeschreibung
"CONTROL (Seite 6420)"	-	Abfrage des Zustands einer Verbindung, die zu einer Instanz einer Anweisung gehört.
-	"C_CNTRL (Seite 6409)"	Abfrage des Zustands einer Verbindung über die Verbindungs-ID.

Hinweis

Den Zustand der aktuellen Verbindung können Sie auch mit der Anweisung "C_DIAG (Seite 5748)" diagnostizieren (nur bei S7-400).

Datenkonsistenz

Definition

Die Größe des Datenbereichs, der nicht gleichzeitig durch konkurrierende Prozesse verändert werden kann, wird als konsistenter Datenbereich bezeichnet. Ein Datenbereich, der größer als der konsistente Datenbereich ist, kann somit in seiner Gesamtheit verfälscht werden.

Das heißt, ein in sich zusammengehöriger Datenbereich, der größer als der konsistente Datenbereich ist, kann zu einem Zeitpunkt teilweise aus neuen und aus alten konsistenten Datenblöcken bestehen.

Beispiel

Eine Inkonsistenz kann entstehen, wenn ein Kommunikations-Baustein z. B. durch einen Prozessalarm-OB mit höherer Priorität unterbrochen wird. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die teilweise bereits vom Kommunikations-Baustein verarbeitet wurden, stammen die übertragenen Daten:

- zum einen Teil aus der Zeit vor der Prozessalarm-Bearbeitung
- und zum anderen Teil aus der Zeit nach der Prozessalarm-Bearbeitung.

Das bedeutet, dass diese Daten inkonsistent (nicht zusammengehörig) sind.

Auswirkung

Wenn große Datenmengen konsistent übertragen werden sollen, dann darf die Übertragung nicht unterbrochen werden. Dadurch kann z. B. die Alarmreaktionszeit der CPU verlängert werden.

Datenkonsistenz bei SIMATIC

- Existiert im Anwenderprogramm eine Kommunikationsfunktion z. B. "BSEND (Seite 6403)" / "BRCV (Seite 6406)", welche auf gemeinsame Daten zugreift, kann der Zugriff auf diesen Datenbereich z. B. über den Parameter DONE selbst koordiniert werden. Die Datenkonsistenz der Kommunikationsbereiche, die lokal mit einem Kommunikationsbaustein übertragen wird, kann deshalb im Anwenderprogramm sichergestellt werden.
- Bei S7-Kommunikationsfunktionen z. B. "PUT (Seite 6395)" / "GET (Seite 6392)" bzw. Schreiben/Lesen über OP-Kommunikation muss bereits bei der Programmierung bzw. Projektierung die Größe der konsistenten Datenbereiche berücksichtigt werden, da im Anwenderprogramm des Zielgerätes (Server) kein Kommunikationsbaustein vorhanden ist, der die Kommunikationsdaten in das Anwenderprogramm einsynchronisiert.
- Bei der S7-300 werden die Kommunikationsdaten in Blöcken zu 32 Bytes im Zykluskontrollpunkt des Betriebssystems konsistent in den Anwenderspeicher kopiert. Für alle größeren Datenbereiche wird keine Datenkonsistenz garantiert. Ist eine definierte Datenkonsistenz gefordert, so dürfen die Kommunikationsdaten im Anwenderprogramm nicht größer als 32 Bytes sein (je nach Ausgabestand maximal 8 Byte).
- Bei der S7-400 werden im Gegensatz dazu die Kommunikationsdaten nicht im Zykluskontrollpunkt, sondern in festen Zeitscheiben während des Programmzyklusses bearbeitet. Systemseitig wird die Konsistenz einer Variable garantiert.
- Auf diese Kommunikationsbereiche kann dann, z. B. von einem OP oder von einer OS, mit den "PUT (Seite 6395)" / "GET (Seite 6392)"-Anweisungen bzw. Lesen/Schreiben von Variablen konsistent zugegriffen werden.

Hinweis

Weitere Hinweise zu Datenkonsistenz finden Sie bei der Beschreibung der einzelnen Anweisungen.

Gemeinsame Parameter der Anweisungen zur S7-Kommunikation

Klassifikation

Die Parameter der Anweisungen zur S7-Kommunikation lassen sich von ihrer Funktion her in die folgenden fünf Klassen einteilen:

1. Steuerparameter dienen der Aktivierung einer Anweisung.
2. Adressierungsparameter dienen der Adressierung des remoten Kommunikationspartners.

3. Sendeparameter zeigen auf diejenigen Datenbereiche, die zum remoten Partner gesendet werden sollen.
4. Empfangsparameter zeigen auf diejenigen Datenbereiche, in welche die vom remoten Partner empfangenen Daten eingetragen werden.
5. Zustandsparameter dienen zur Überwachung, ob die Anweisung ihre Aufgabe fehlerfrei beendet hat, bzw. zur Analyse der aufgetretenen Fehler.

Steuerparameter

Die Aktivierung des Datenaustauschs erfolgt nur dann, wenn beim Aufruf der Anweisung die zugehörigen Steuerparameter einen definierten Wert haben (z. B. gesetzt sind) bzw. wenn sich der Wert gegenüber dem letzten Aufruf in definierter Weise geändert hat (z. B. positive Flanke).

Hinweis

Erstaufruf bei S7-300

Sie müssen beim Erstaufruf den Parameter REQ mit FALSE belegen.

Adressierungsparameter

Parameter	Beschreibung
ID	Referenz auf die lokale Verbindungsbeschreibung (wird durch die Verbindungsprojektierung vorgegeben). Hinweis: Bei den Anweisungen der S7-Kommunikation ist die ID W#16#EEEE nicht erlaubt.
R_ID	Mit dem Parameter R_ID legen Sie die Zusammengehörigkeit einer Sende- und einer Empfangs-Anweisung fest: Der Parameter R_ID muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite übereinstimmen. Damit wird die Kommunikation mehrerer Anweisungspaare über dieselbe logische Verbindung ermöglicht. • R_ID muss in der Form DW#16#wxyzWXYZ angegeben werden. • Die über R_ID festgelegten Anweisungspaare einer logischen Verbindung müssen für diese Verbindung eindeutig sein.

Der Parameter PI_NAME ist nur bei den relevanten Anweisungen beschrieben (nur S7-400).

Hinweis

Adressierungsparameter ID und R_ID

S7-300: Sie können die Adressierungsparameter ID und R_ID zur Laufzeit umparametrieren. Die neuen Parameter werden mit jedem neuen Auftrag nach Abschluss des vorangegangenen Auftrags wirksam.

Mit folgenden Möglichkeiten können Sie Instanz-DBs und somit Arbeitsspeicher sparen:

1. Durch variable IDs können Sie mehrere Verbindungen über einen Dateninstanzbaustein nutzen.
2. Durch variable R_IDs können Sie mit einer Instanz mehrere Zusammengehörigkeiten von Sende- und Empfangs-Anweisungspaaren für einen Auftrag festlegen.
3. Sie können die Fälle 1 und 2 miteinander kombinieren.

Beachten Sie, dass die neuen Parameter erst nach Abschluss des vorangegangenen Auftrags gültig werden. Wenn Sie den Sendevorgang aktivieren, muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite der Parameter R_ID übereinstimmen.

S7-400: Die Adressierungsparameter ID und R_ID werden nur beim Erstaufruf der Anweisung oder beim Überladen der Instanz ausgewertet (die Aktualparameter oder die vordefinierten Werte aus der Instanz). Mit dem Erstaufruf wird somit die Kommunikationsbeziehung (Verbindung) zum remoten Partner festgeschrieben, und zwar bis zum nächsten Neustart (Warmstart) oder Kaltstart.

Zustandsparameter

Mit den Zustandsparametern überwachen Sie, ob die Anweisung ihre Aufgabe ordnungsgemäß beendet hat oder ob sie noch tätig ist. Darüber hinaus zeigen diese aufgetretene Fehler an.

Hinweis

Die Zustandsparameter sind nur einen Zyklus lang gültig, nämlich vom ersten Befehl, der dem Aufruf folgt, bis zum nächsten Aufruf. Daraus folgt, dass Sie diese Parameter nach jedem Durchlauf der Anweisung auswerten müssen.

Sende- und Empfangsparameter

Wenn Sie bei einer Anweisung nicht alle Sende- bzw. Empfangsparameter nutzen, so muss der erste unbenutzte Parameter jeweils ein NIL-Pointer sein, und die benutzten Parameter müssen lückenlos hintereinander stehen.

Hinweis

Erstaufruf bei S7-400

Beim Erstaufruf wird durch die ANY-Pointer die für den Auftrag maximal übertragbare Nutzdatenmenge festgeschrieben; es wird nämlich im Arbeitsspeicher der CPU ein Kommunikationsdatenpuffer zur Konsistenzsicherung der Daten angelegt. Dieser belegt bis zu 480 Byte Arbeitsspeicher. Es wird empfohlen, den Erstaufruf dann im Neustart- (Warmstart-) oder Kaltstart-OB durchzuführen, wenn der Baustein mit dem Aufruf der Anweisung nicht im Betriebszustand RUN der CPU nachgeladen wird.

Bei Folgeaufrufen können Sie eine beliebige Anzahl von Daten versenden/empfangen, jedoch höchstens so viele wie beim Erstaufruf.

Eine Ausnahme von dieser Regel bilden die Anweisungen "BSEND (Seite 6403)" und "BRCV (Seite 6406)". Mit ihnen können Sie bis zu 64 Kbyte pro Auftrag übertragen.

Bei den Anweisungen zur zweiseitigen Kommunikation

- muss die Anzahl der verwendeten SD_i-und RD_i-Parameter auf Sende- und Empfangsseite übereinstimmen
- müssen die Datentypen bei zusammengehörigen SD_i- und RD_i-Parametern auf Sende- und Empfangsseite übereinstimmen
- darf die Anzahl der über den Parameter SD_i zu sendenden Daten nicht größer sein als der Bereich, der durch den zugehörigen Parameter RD_i aufgespannt wird (gilt nicht für "BSEND (Seite 6403)" / "BRCV (Seite 6406)").

Wenn Sie gegen die genannten Regeln verstoßen, so wird Ihnen dies mit ERROR = 1 und STATUS = 4 angezeigt.

Nutzdatengröße

Bei den Anweisungen "USEND (Seite 6398)", "URCV (Seite 6400)", "GET (Seite 6392)" und "PUT (Seite 6395)" darf die Anzahl der zu übertragenden Daten eine bestimmte Nutzdatenlänge nicht überschreiten. Die maximale Nutzdatengröße ist abhängig von:

- verwendeter Anweisung und
- Kommunikationspartner.

Die garantierte Mindestgröße der Nutzdaten für eine Anweisung mit 1-4 Variablen können Sie folgender Tabelle entnehmen:

Anweisung	Partner: S7-300	Partner: S7-400
PUT / GET	160 Byte	400 Byte
USEND / URCV	160 Byte	440 Byte
BSEND / BRCV	32768/65534 Byte	65534 Byte

Beachten Sie die Einschränkungen zur Nutzdatengröße in den Technischen Daten der jeweiligen CPU.

Exakte Nutzdatengröße

Wenn die oben angegebene Nutzdatengröße nicht ausreicht, können Sie die exakte maximale Nutzdatenlänge in Byte wie folgt ermitteln:

- Zunächst lesen Sie aus der folgenden Tabelle die bei der Kommunikation gültige Datenblockgröße ab:

Eigene CPU	Remote CPU	Datenblockgröße in Byte
S7-300	beliebig	240 (S7-300)
S7-400	S7-300	240 (S7-400)
S7-400	S7-400	480

- Mit diesem Wert gehen Sie in die folgende Tabelle und lesen daraus die maximal mögliche Nutzdatenlänge in Byte als Summe der benutzten Parameter ab. Sie gilt für gerade Längen der Bereiche SD_i, RD_i, ADDR_i. Für jeden Bereich ungerader Länge reduziert sich die maximal mögliche Nutzdatenlänge um ein Byte.

		Anzahl benutzter Parameter SD_i, RD_i, ADDR_i			
Datenblockgröße	Anweisung	1	2	3	4
240 (S7-300)	PUT/GET/ USEND	160	-	-	-
240 (S7-300 via integrierte Schnittstelle)	PUT	212	-	-	-
	GET	222	-	-	-
	USEND	212	-	-	-
240 (S7-400)	PUT	212	196	180	164
	GET	222	218	214	210
	USEND	212	-	-	-
480	PUT	452	436	420	404
	GET	462	458	454	450
	USEND	452	448	444	440

Anlaufverhalten der Anweisungen zur S7-Kommunikation

Voraussetzungen

Im Folgenden wird für die S7-400 vorausgesetzt, dass

- die Verbindungsbeschreibungen (SDBs) auf den Baugruppen vorhanden sind
- die projektierten Verbindungen aufgebaut sind
- bei jeder Anweisung der Aktualparameter für ID mit der projektierten Verbindungs-ID übereinstimmt.

Neustart (Warmstart) und Kaltstart

Im Neustart (Warmstart) und Kaltstart werden alle Anweisungen in den Zustand NO_INIT versetzt. Die in den Instanz-DBs gespeicherten Aktualparameter bleiben unverändert.

Neustart (Warmstart) und Kaltstart bei Anweisungen zum zweiseitigen Datenaustausch

In der Regel führen bei Anweisungen zum zweiseitigen Datenaustausch nicht beide Baugruppen gleichzeitig einen Neustart (Warmstart) oder Kaltstart aus. Die Anpassung wird implizit durch die Anweisung nach folgenden Regeln ausgeführt.

- Empfangsbausteine ("URCV (Seite 6400)", "BRCV (Seite 6406)") verhalten sich wie folgt:
  - Hat die Anweisung einen Auftrag empfangen, aber zum Zeitpunkt des Neustarts (Warmstarts) oder Kaltstarts noch nicht quittiert, so generiert sie ein Sequenzabbruch-Telegramm ("BRCV (Seite 6406)") und verzweigt dann unmittelbar in den Zustand NO_INIT.
  - Bei "BRCV (Seite 6406)" kann der Fall eintreten, dass trotz gesendetem Sequenzabbruch noch ein weiteres Datensegment eintrifft. Dieses wird lokal verworfen.
  - Bei "URCV (Seite 6400)" erfolgt sofort der Übergang in den Zustand NO_INIT.
- Sendebausteine ("USEND (Seite 6398)", "BSEND (Seite 6403)") verhalten sich wie folgt:
  - Hat die Anweisung "BSEND (Seite 6403)" eine Auftragssequenz begonnen, die noch nicht abgeschlossen ist, so sendet sie im Neustart (Warmstart) oder Kaltstart einen Sequenzabbruch. Unmittelbar danach verzweigt sie in den Zustand NO_INIT. Eine später noch eintreffende Quittung wird lokal verworfen.
  - Hat die Anweisung "BSEND (Seite 6403)" zum Zeitpunkt der Neustart- (Warmstart-) oder Kaltstartanforderung bereits einen Sequenzabbruch gesendet oder empfangen, geht sie sofort in den Zustand NO_INIT.
  - In allen anderen Fällen und wenn die Anweisung nur Meldungen absetzt (z. B. "USEND (Seite 6398)") wird die lokale Bearbeitung abgebrochen, und die Anweisung verzweigt sofort in den Zustand NO_INIT.

Neustart (Warmstart) und Kaltstart bei Anweisungen zum einseitigen Datenaustausch

Es wird davon ausgegangen, dass nach dem Aufbau der Verbindungen der Server auf dem Kommunikationspartner funktionsfähig ist, d. h. jederzeit Aufträge bearbeiten oder Meldungen absetzen kann.

Anweisungen, die Aufträge absenden und Quittungen erwarten, zeigen folgendes Verhalten:

- Die momentane Bearbeitung wird abgebrochen, und es wird unmittelbar danach in den Zustand NO_INIT verzweigt. Eine später noch eintreffende Quittung für den vor dem Neustart (Warmstart) oder Kaltstart abgesendeten Auftrag wird lokal verworfen.
- Es kann bereits ein neuer Auftrag abgesetzt werden, bevor die noch ausstehende Quittung eingetroffen ist.

Anweisungen, die Meldungen absetzen oder empfangen, verhalten sich wie folgt:

- Die momentane Bearbeitung wird abgebrochen, und es wird unmittelbar in den Zustand NO_INIT verzweigt.
- Bei "USTATUS (Seite 6437)" werden Meldungen, die in den Zuständen NO_INIT und DISABLED eintreffen, lokal verworfen.

Wiederanlaufverhalten

Die Anweisungen der S7-Kommunikation werden nur im Neustart (Warmstart) oder Kaltstart in den Zustand NO_INIT versetzt. Daraus ergibt sich, dass sie sich im Wiederanlauf wie fortsetzbare Anwenderfunktionsbausteine verhalten.

Verhalten nach Urlöschen

Urlöschen führt immer zum Abbruch aller Verbindungen. Da nach Urlöschen als Anlaufart für das Anwenderprogramm nur ein Neustart (Warmstart) oder Kaltstart möglich ist, werden alle Anweisungen der S7-Kommunikation (sofern noch vorhanden) in den Zustand NO_INIT versetzt und initialisiert. Partnerbausteine in einer nicht urgelöschten Baugruppe gehen aufgrund des Verbindungsabbruchs in die Zustände IDLE oder ENABLED oder DISABLED.

Störverhalten der Anweisungen zur S7-Kommunikation

Einleitung

Im Folgenden wird das Störverhalten der Anweisungen zur S7-Kommunikation bei S7-400 beschrieben.

Verbindungsabbruch

Die den Anweisungs-Instanzen zugeordneten Verbindungen werden auf Abbruch überwacht.

Bei einem Verbindungsabbruch ist die Reaktion der Anweisung abhängig von ihrem inneren Zustand:

- Wird der Verbindungsabbruch in den Zuständen IDLE oder ENABLED erkannt, reagiert die Anweisung folgendermaßen:
  - Sie verzweigt in den Zustand ERROR und gibt über die Ausgangsparameter ERROR und STATUS die Fehlerkennung "Kommunikationsprobleme" aus.
  - Beim nächsten Aufruf nimmt sie wieder ihren ursprünglichen Zustand ein und überprüft erneut die Verbindung.
- Eine Anweisung, die sich nicht in den Zuständen IDLE oder DISABLED befindet, reagiert wie folgt:
  - Sie bricht ihre Bearbeitung ab, geht sofort oder beim nächsten Aufruf in den Zustand ERROR und gibt über die Ausgangsparameter ERROR und STATUS die Fehlerkennung "Kommunikationsprobleme" aus.
  - Beim nächsten Aufruf nimmt die Anweisung den Zustand IDLE, DISABLED oder ENABLED ein. In den Zuständen IDLE und ENABLED wird erneut die Verbindung überprüft.

Diese Vorgehensweise wird auch dann durchgeführt, wenn die Verbindung inzwischen wieder aufgebaut wurde.

Netzausfall

Ein gepufferter Netzausfall mit Wiederanlauf hat den Abbruch aller aufgebauten Verbindungen zur Folge. Für alle betroffenen Anweisungen gilt daher das oben Gesagte.

Bei einem gepufferten Netzausfall mit automatischem Neustart (Warmstart) oder Kaltstart gilt sowohl das zum Verbindungsabbruch als auch das zum Neustart (Warmstart) oder Kaltstart Gesagte.

Im Sonderfall des ungepufferten automatischen Neustarts (Warmstarts) oder Kaltstarts, bei dem nach Netzwiederkehr automatisch ein Urlöschen durchgeführt wird, verhalten sich die Anweisungen zur S7-Kommunikation wie im Kapitel "Anlaufverhalten der Anweisungen zur S7-Kommunikation (Seite 6388)" beschrieben.

Verhalten bei Betriebszustandsübergängen

Bei einem Betriebszustandsübergang zwischen den Zuständen STOP, ANLAUF, RUN und HALT bleibt die Anweisung in ihrem aktuellen Zustand (Ausnahme: Bei Neustart (Warmstart) oder Kaltstart wird der Zustand NO_INIT eingenommen.). Das gilt sowohl bei Anweisungen zur einseitigen als auch bei Anweisungen zur zweiseitigen Kommunikation.

Fehlerschnittstelle zum Anwenderprogramm

Tritt bei der Bearbeitung einer Anweisung ein Fehler auf, so erfolgt stets ein Übergang in den Zustand ERROR; gleichzeitig werden der Ausgangsparameter ERROR auf "1" gesetzt und in den Ausgangsparameter STATUS die zugehörige Fehlerkennung eingetragen. Diese Anzeigen können Sie in Ihrem Programm auswerten.

Beispiele für mögliche Fehler:

- Fehler beim Sammeln der Sendedaten
- Fehler beim Kopieren der Empfangsdaten in die Empfangsbereiche (z. B. Zugriff auf nicht vorhandenen DB)
- Die Länge des gesendeten Datenbereichs stimmt nicht mit der auf der Partner-Anweisung hinterlegten Länge für den Empfangsbereich überein.

GET: Daten aus einer remoten CPU lesen

Beschreibung

Mit Hilfe der Anweisung können Sie Daten aus einer remoten CPU auslesen, falls die Verbindung nicht über einen CP erfolgt.

- **S7-300:** Der Lesevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter ID, ADDR_1 und RD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern ID, ADDR_1 und RD_1 neue Werte zuweisen.
- **S7-400:** Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet. Dabei werden die relevanten Zeiger auf die auszulesenden Bereiche (ADDR_i) an die Partner-CPU gesendet.

Der remote Partner sendet die Dateninhalte zurück. Die empfangenen Daten werden beim nächsten Aufruf in die projektierten Empfangsbereiche (RD_i) kopiert. Sie müssen darauf achten, dass die über die Parameter ADDR_i und RD_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt. Eine erneute Aktivierung eines Lesevorgangs ist erst nach dem Abschluss des letzten möglich.

Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden. Wenn beim Lesen der Daten Zugriffsprobleme auftraten, oder die Datentypprüfung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR / STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
S7-300: ADDR_1 **S7-400:** ADDR_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10). Bei der Übertragung von Datenstrukturen (z.B. Struct, Array) muss an dem Parameter ADDR der Datentyp CHAR, BYTE, WORD oder DWORD verwendet werden.
S7-300: RD_1 **S7-400:** RD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der eigenen CPU, in der die gelesenen Daten abgelegt werden. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "GET" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Empfangsbereichszeigern RD_i bezüglich der Datenlänge oder des Datentyps.
1	8	Zugriffsfehler bei der Partner-CPU.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "GET" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) – Ist beim Erstaufruf möglich
1	27	**Nur bei S7-300:** Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie folgendes beachten:

Werten Sie den aktuell benutzten Teil des Empfangsbereichs RD_i vollständig aus, bevor Sie einen erneuten Auftrag aktivieren.

PUT: Daten in eine remote CPU schreiben

Beschreibung

Mit Hilfe der Anweisung können Sie Daten in eine remote CPU schreiben, falls die Verbindung nicht über einen CP erfolgt.

- **S7-300:** Der Sendevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter ID, ADDR_1 und SD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern ID, ADDR_1 und SD_1 neue Werte zuweisen.
- **S7-400:** Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet. Dabei werden die Zeiger auf die zu schreibenden Bereiche (ADDR_i) und die Daten (SD_i) an die Partner-CPU gesendet.

Der remote Partner legt die gesendeten Daten unter den mitgeführten Adressen ab und sendet eine Ausführungsquittung zurück. Sie müssen darauf achten, dass die über die Parameter ADDR_i und SD_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

Falls keine Fehler auftraten, wird dies beim nächsten Aufruf am Zustandsparameter DONE mit "1" angezeigt. Eine erneute Aktivierung eines Schreibvorgangs ist erst nach dem Abschluss des letzten möglich.

Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden. Wenn beim Schreiben der Daten Zugriffsprobleme auftraten, oder die Prüfung der Ausführungsquittung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PUT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
S7-300: ADDR_1 **S7-400:** ADDR_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10). Bei der Übertragung von Datenstrukturen (z.B. Struct, Array) muss an dem Parameter ADDR der Datentyp CHAR, BYTE, WORD oder DWORD verwendet werden.
S7-300: SD_1 **S7-400:** SD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der eigenen CPU, die die zu versendenden Daten enthalten. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für PUT spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.

ERROR	STATUS (dezimal)	Erläuterung
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Sendebereichszeigern SD_i bezüglich der Datenlänge oder des Datentyps
1	8	Zugriffsfehler bei der Partner-CPU
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "PUT" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut)
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich) – Ist beim Erstaufruf möglich
1	27	**Nur bei S7-300:** Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz bei S7-300:

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Sendebereich SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

Datenkonsistenz bei S7-400 und S7-300 über eine integrierte Schnittstelle:

Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) sind die zu sendenden Daten der Sendebereiche SD_i aus dem Anwenderprogramm kopiert. Sie können diese Bereiche nach dem Aufruf der Anweisung neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

Hinweis

Der gesamte Sendevorgang ist erst dann abgeschlossen, wenn der Zustandsparameter DONE den Wert "1" annimmt.

USEND: Daten unkoordiniert senden

Beschreibung

Die Anweisung "USEND" sendet Daten an eine remote Partner-Anweisung vom Typ "URCV (Seite 6400)". Der Sendevorgang verläuft ohne Koordination mit der Partner-Anweisung, d. h. die Datenübertragung erfolgt ohne Quittierung durch die Partner-Anweisung.

- **S7-300:** Der Sendevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter R_ID, ID und SD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID und SD_1 neue Werte zuweisen.
- **S7-400:** Der Sendevorgang erfolgt nach einer positiven Flanke am Steuereingang REQ. Die zu sendenden Daten werden durch die Parameter SD_1, ... SD_4 referenziert, wobei diese vier Sendeparameter nicht alle belegt sein müssen.

Sie müssen jedoch darauf achten, dass die über die Parameter SD_1 bis SD_4 / SD_1 und RD_1 bis RD_4 / RD_1 (bei der zugehörigen Partner-Anweisung "URCV (Seite 6400)") definierten Bereiche übereinstimmen in:

- Anzahl
- Länge und
- Datentyp.

Der Parameter R_ID muss bei beiden Anweisungen identisch sein.

Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit einer logischen "1" angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	
S7-300: SD_1 **S7-400:** SD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf i-ten Sendebereich. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten
1	4	Fehler in den Sendebereichszeigern SD_i bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "USEND" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • ein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).

ERROR	STATUS (dezimal)	Erläuterung
1	18	• R_ID existiert bereits in der Verbindung ID. • zusätzlich S7-300: Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich).
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich) – Ist beim Erstaufruf möglich
1	27	**Nur bei S7-300:** Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

S7-300: Um Datenkonsistenz zu gewährleisten, dürfen Sie den Sendebereich SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

S7-400 und S7-300 über eine integrierte Schnittstelle: Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) sind die zu sendenden Daten der Sendebereiche SD_i aus dem Anwenderprogramm kopiert. Sie können diese Bereiche nach dem Aufruf der Anweisung neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

Hinweis

Der gesamte Sendevorgang ist erst dann abgeschlossen, wenn der Zustandsparameter DONE den Wert "1" annimmt.

URCV: Daten unkoordiniert empfangen

Beschreibung

Die Anweisung "URCV" empfängt asynchron Daten von einer remoten Partner-Anweisung vom Typ "USEND (Seite 6398)" und kopiert sie in die projektierten Empfangsbereiche.

Die Anweisung ist empfangsbereit, wenn am Eingang EN_R eine logische "1" anliegt. Mit EN_R=0 kann ein laufender Auftrag abgebrochen werden.

- **S7-300:** Mit jeder positiven Flanke an EN_R werden die Parameter R_ID, ID und RD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID und RD_1 neue Werte zuweisen.
- **S7-400:** Die Empfangsdatenbereiche werden durch die Parameter RD_1, ... RD_4 referenziert.

Sie müssen beachten, dass die über die Parameter RD_i / RD_1 und SD_i / SD_1 (bei der zugehörigen Partner-Anweisung "USEND (Seite 6398)") definierten Bereiche übereinstimmen in:

- Anzahl
- Länge und
- Datentyp.

Der erfolgreiche Abschluss des Kopiervorgangs wird am Zustandsparameter NDR mit einer logischen "1" angezeigt. Der Parameter R_ID muss bei beiden Anweisungen identisch sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "URCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	
S7-300: RD_1 **S7-400:** RD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf den i-ten Empfangsbereich: Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	9	Overrun-Warnung: Ältere Empfangsdaten wurden von neueren Empfangsdaten überschrieben.
0	11	Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten
1	4	Fehler in den Empfangsbereichszeigern RD_i bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "URCV" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • Kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	18	• R_ID existiert bereits in der Verbindungs-ID • zusätzlich S7-300: Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.)
1	19	Die zugehörige Anweisung "USEND (Seite 6398)" sendet schneller Daten als diese von "URCV" in die Empfangsbereiche kopiert werden können.
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) – Ist beim Erstaufruf möglich
1	27	**Nur bei S7-300:** Für diesen Baustein existiert in der CPU kein Funktionscode.

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie Folgendes beachten:

- **S7-300:** Nachdem der Zustandsparameter NDR den Wert "1" angenommen hat, müssen Sie "URCV" mit dem Wert "0" an EN_R sofort erneut aufrufen. Damit wird gewährleistet, dass der Empfangsbereich nicht bereits überschrieben wird, bevor Sie ihn ausgewertet haben. Werten Sie den Empfangsbereich RD_1 vollständig aus, bevor Sie den Baustein wieder mit dem Wert "1" am Steuereingang EN_R aufrufen.
- **S7-400:** Nachdem der Zustandsparameter NDR den Wert "1" angenommen hat, befinden sich neue Empfangsdaten in Ihren Empfangsbereichen (RD_i). Ein erneuter Bausteinaufruf kann diese Daten mit neuen Empfangsdaten überschreiben. Wenn Sie dies verhindern wollen, müssen Sie "URCV" mit dem Wert "0" an EN_R so lange aufrufen (z. B. bei zyklischer Bausteinbearbeitung), bis Sie die Bearbeitung der Empfangsdaten beendet haben.

BSEND: Daten in Segmenten senden

Beschreibung

Die Anweisung "BSEND" sendet Daten an eine remote Partner-Anweisung vom Typ "BRCV (Seite 6406)". Bei diesem Datentransfer kann eine größere Datenmenge zwischen den Kommunikationspartnern transportiert werden, als dies mit allen anderen Kommunikations-Anweisungen für projektierte S7-Verbindungen möglich ist. Folgende Datenmengen können übertragen werden:

- 32768 Byte bei S7-300 über SIMATIC-Net-CPs
- 65534 Byte bei S7-400 und S7-300 über eine integrierte Schnittstelle

Arbeitsweise

Der zu sendende Datenbereich wird segmentiert. Jedes Segment wird einzeln an den Partner gesendet. Das letzte Segment wird vom Partner bereits bei seiner Ankunft quittiert, unabhängig vom zugehörigen Aufruf der Anweisung "BRCV (Seite 6406)".

- **S7-300:** Der Sendevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter R_ID, ID, SD_1 und LEN übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID, SD_1 und LEN neue Werte zuweisen. Zur Übertragung von segmentierten Daten muss die Anweisung zyklisch im Anwenderprogramm aufgerufen werden.
  Die Anfangsadresse und die maximale Länge der zu sendenden Daten werden durch SD_1 vorgegeben. Die Länge des Datenblocks legen Sie auftragsbezogen durch LEN fest.
- **S7-400 und S7-300 über eine integrierte Schnittstelle:** Die Aktivierung des Sendevorgangs erfolgt nach Aufruf der Anweisung und positiver Flanke am Steuereingang REQ. Das Senden der Daten aus dem Anwenderspeicher erfolgt asynchron zur Bearbeitung des Anwenderprogramms.
  Die Anfangsadresse der zu sendenden Daten wird durch SD_1 vorgegeben. Die Länge der Sendedaten legen Sie auftragsbezogen durch LEN fest. LEN ersetzt damit den Längenanteil von SD_1.

Der Parameter R_ID muss bei den zusammengehörenden Anweisungen identisch sein. Bei einer positiven Flanke am Steuereingang R wird ein laufender Sendevorgang abgebrochen. Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit "1" angezeigt.

Nach Abschluss eines Sendevorganges, kann erst wieder ein neuer Sendeauftrag bearbeitet werden, wenn die Zustandsparameter DONE oder ERROR den Wert "1" angenommen haben.

Aufgrund der asynchronen Datenübertragung kann ein erneutes Senden von Daten erst gestartet werden, wenn die vorhergehenden Daten durch Aufruf der Partner-Anweisung abgeholt wurden. Bis die Daten abgeholt wurden, wird beim Aufruf von "BSEND" der Statuswert "7" (s. u.) ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke
R	Input	BOOL	E, A, M, D, L	Steuerparameter reset, aktiviert Abbruch eines noch laufenden Datenaustauschs bei steigender Flanke
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384) Bei Kopplung über CP441 zu S5- oder Fremdgeräten enthält R_ID die Adressinformation des remoten Geräts. Näheres siehe CP441-Beschreibung.
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	E, A, M, D, L	

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SD_1	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf Sendebereich. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren. (z. B.: P#DB10.DBX5.0 Byte 10)
LEN	InOut	WORD	E, A, M, D, L	Länge des zu sendenden Datenblocks in Byte

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BSEND" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	• Kommunikationsprobleme, z. B. Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.
1	2	Negative Quittung von Partner-Anweisung. Die Funktion ist nicht ausführbar.
1	3	R_ID ist auf der durch ID vorgegebenen Kommunikationsverbindung nicht bekannt, oder der Empfangsbaustein wurde noch nie aufgerufen.
1	4	Fehler im Sendebereichszeiger SD_1 bezüglich der Datenlänge oder des Datentyps, oder am Parameter LEN wurde der Wert "0" übergeben.
1	5	Resetanforderung wurde ausgeführt.
1	6	Partner-Anweisung befindet sich im Zustand DISABLED (EN_R hat den Wert "0"). Überprüfen Sie zusätzlich die Eingangsparameter von BRCV auf Konsistenz zu BSEND.
1	7	Partner-Anweisung befindet sich in falschem Zustand Die Empfangs-Anweisungen wurde nach der letzten Datenübertragung nicht mehr aufgerufen.
1	8	Zugriff auf remotes Objekt im Anwenderspeicher wurde abgelehnt: Der Zielbereich beim zugehörigen "BRCV" ist zu klein. Beim zugehörigen BRCV wird ERROR = 1, STATUS = 4 oder ERROR = 1, STATUS = 10 gemeldet.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).

ERROR	STATUS (dezimal)	Erläuterung
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "BSEND" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	18	• R_ID existiert bereits in der Verbindung. • zusätzlich S7-300: Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.)
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) – Ist beim Erstaufruf möglich – Speicherengpass des CP
1	27	**Nur bei S7-300:** Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Um Datenkonsistenz zu gewährleisten, dürfen Sie den aktuell benutzten Teil des Sendebereichs SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

BRCV: Daten in Segmenten empfangen

Beschreibung

Die Anweisung "BRCV" empfängt Daten von einer remoten Partner-Anweisung vom Typ "BSEND (Seite 6403)". Nach jedem empfangenen Datensegment wird eine Quittung an die Partner-Anweisung geschickt, und der Parameter LEN wird aktualisiert.

Die Anweisung ist nach Aufruf mit dem Wert "1" am Steuereingang EN_R empfangsbereit. Mit EN_R=0 kann ein laufender Auftrag abgebrochen werden.

Die Anfangsadresse und die maximale Länge des Empfangsbereichs werden durch RD_1 vorgegeben. Die Länge des empfangenen Datenblocks wird in LEN angezeigt.

- **S7-300:** Mit jeder positiven Flanke an EN_R werden die Parameter R_ID, ID und RD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID und RD_1 neue Werte zuweisen. Zur Übertragung von segmentierten Daten muss die Anweisung zyklisch im Anwenderprogramm aufgerufen werden.
- **S7-400 und S7-300 über eine integrierte Schnittstelle:** Der Empfang der Daten aus dem Anwenderspeicher erfolgt asynchron zur Bearbeitung des Anwenderprogramms.

Der Parameter R_ID muss bei den zusammengehörenden Anweisungen identisch sein.

Der fehlerfreie Empfang sämtlicher Datensegmente wird am Zustandsparameter NDR mit dem Wert "1" angezeigt. Die empfangenen Daten bleiben unverändert bis zum nächsten Aufruf mit EN_R=1.

Wird die Anweisung während eines asynchronen Empfangs erneut aufgerufen, führt dies zur Ausgabe einer Warnung im Zustandsparameter STATUS; erfolgt der Aufruf mit EN_R=0, wird der Empfangsvorgang abgebrochen, und die Anweisung geht in seinen Grundzustand.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BRCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384) Bei Kopplung über CP441 zu S5- oder Fremdgeräten enthält R_ID die Adressinformation des remoten Geräts. Näheres siehe CP441-Beschreibung.
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
STATUS	Output	WORD	E, A, M, D, L	

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RD_1	InOut	ANY	S7-300: M, D S7-400: E, A, M, D, T, Z	Zeiger auf Empfangsbereich. Die Längenangabe gibt die maximale Länge des zu empfangenden Blocks vor. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren. (z. B.: P# DB10.DBX5.0 Byte 10).
LEN	InOut	WORD	E, A, M, D, L	Länge der bisher empfangenen Daten in Byte.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BRCV" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	17	Warnung: Anweisung empfängt asynchron Daten. Der Parameter LEN zeigt die Anzahl der bisher empfangenen Daten in Byte an.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • zusätzlich S7-300: Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.
1	2	Funktion nicht ausführbar (Protokollfehler)
1	4	Fehler im Empfangsbereichszeiger RD_1 bezüglich der Datenlänge oder des Datentyps. Der gesendete Datenblock ist länger als der Empfangsbereich.
1	5	Resetanforderung eingetroffen, unvollständige Übertragung.
1	8	Zugriffsfehler beim zugehörigen "BSEND (Seite 6403)": Nach dem Versenden des letzten gültigen Datensegments wird ERROR = 1 und STATUS = 4 oder ERROR = 1 und STATUS = 10 gemeldet.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zum "BRCV" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).

ERROR	STATUS (dezimal)	Erläuterung
1	18	• R_ID existiert bereits in der Verbindung. • zusätzlich S7-300: Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.)
1	20	• **S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode. • **S7-300** – Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten – Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) – Ist beim Erstaufruf möglich – Speicherengpass des CP
1	27	**Nur bei S7-300:** Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie Folgendes beachten: Werten Sie den aktuell benutzten Teil des Empfangsbereichs RD_1 vollständig aus, bevor Sie den Baustein wieder mit dem Wert 1 am Steuereingang EN_R aufrufen.

Sonderfall eines Datenempfangs (nur S7-400)

Falls die Empfänger-CPU mit einer empfangsbereiten "BRCV"-Anweisung (d. h. ein Aufruf mit EN_R =1 ist bereits erfolgt) in STOP geht, bevor die zugehörige Sende-Anweisung das erste Datensegment eines Auftrags abgeschickt hat, geschieht Folgendes:

- Die Daten des ersten Auftrags nach Übergang der Empfänger-CPU in STOP werden vollständig in den Empfangsbereich eingetragen.
- Die Partner-Anweisung "BSEND (Seite 6403)" erhält darüber eine positive Quittung.
- Weitere "BSEND (Seite 6403)"-Aufträge können von der Empfänger-CPU im STOP-Zustand nicht mehr angenommen werden.
- Solange sie sich im STOP-Zustand befindet, haben NDR und LEN den Wert "0".

Damit Ihnen die Information über die empfangenen Daten nicht verloren geht, müssen Sie bei der Empfänger-CPU einen Wiederanlauf durchführen und "BRCV" mit EN_R=1 aufrufen.

C_CNTRL: Verbindungszustand abfragen

Beschreibung

Mit der Anweisung ermitteln Sie **bei S7-300** den Zustand einer Verbindung. Nach Aufruf der Anweisung mit dem Wert "1" am Steuereingang EN_R wird der momentane Zustand der über ID adressierten Verbindung ermittelt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "C_CNTRL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
C_CONN	Output	BOOL	E, A, M, D, L	Zustand der zugehörigen Verbindung. Mögliche Werte: • 0: Verbindung abgebrochen oder nicht aufgebaut. • 1: Verbindung vorhanden.
C_STATUS	Output	WORD	E, A, M, D, L	Verbindungszustand: • W#16#0000: Verbindung ist nicht eingerichtet • W#16#0001: Verbindung wird gerade eingerichtet • W#16#0002:Verbindung ist eingerichtet • W#16#000F: Keine Daten für Verbindungsstatus verfügbar (z. B. beim CP-Anlauf) • W#16#00FF: Verbindung ist nicht projektiert

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL, ERROR und STATUS

Der Ausgangsparameter RET_VAL kann bei "C_CNTRL" die folgenden beiden Werte annehmen:

- 0000H: Bei der Ausführung ist kein Fehler aufgetreten.
- 8000H: Bei der Ausführung ist ein Fehler aufgetreten.

ERROR	STATUS (dezimal)	Erläuterung
1	10	Fehler beim Zugriff auf den CP, weil gerade ein anderer Auftrag läuft. Auftragswiederholung zu einem späteren Zeitpunkt.
1	27	Für diese Anweisung existiert in der CPU kein Funktionscode.

Hinweis

Auch dann, wenn im Ausgangsparameter RET_VAL der Wert 0000H angezeigt wird, sind die Ausgangsparameter ERROR und STATUS auszuwerten.

Weitere

GET_S: Daten aus einer remoten CPU lesen

Beschreibung

Mit Hilfe der Anweisung können Sie **bei S7-300** Daten aus einer remoten CPU auslesen. Dabei kann die Verbindung sowohl über die integrierte Schnittstelle als auch über einen CP erfolgen.

Der Lesevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter ID, ADDR_1 und RD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern ID, ADDR_1 und RD_1 neue Werte zuweisen.

Der remote Partner sendet die Dateninhalte zurück. Die empfangenen Daten werden beim nächsten Aufruf in die projektierten Empfangsbereiche (RD_1) kopiert. Sie müssen darauf achten, dass die über die Parameter ADDR_1 und RD_1 definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt. Eine erneute Aktivierung eines Lesevorgangs ist erst nach dem Abschluss des letzten möglich.

Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden. Wenn beim Lesen der Daten Zugriffsprobleme auftraten, oder die Datentypprüfung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET_S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR / STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
ADDR_1	InOut	ANY	M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen.
RD_1	InOut	ANY	M, D	Zeiger auf diejenigen Bereiche in der eigenen CPU, in der die gelesenen Daten abgelegt werden. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "GET_S" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.

ERROR	STATUS (dezimal)	Erläuterung
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Empfangsbereichszeigern RD_1 bezüglich der Datenlänge oder des Datentyps.
1	8	Zugriffsfehler bei der Partner-CPU.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "GET_S" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	• Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) • Ist beim Erstaufruf möglich
1	27	Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie Folgendes beachten:

Werten Sie den aktuell benutzten Teil des Empfangsbereichs RD_1 vollständig aus, bevor Sie einen erneuten Auftrag aktivieren.

PUT_S: Daten in eine remote CPU schreiben

Beschreibung

Mit Hilfe der Anweisung können Sie **bei S7-300** Daten in eine remote CPU schreiben. Dabei kann die Verbindung sowohl über die integrierte Schnittstelle als auch über einen CP erfolgen.

Der Sendevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter ID, ADDR_1 und SD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern ID, ADDR_1 und SD_1 neue Werte zuweisen.

Der remote Partner legt die gesendeten Daten unter den mitgeführten Adressen ab und sendet eine Ausführungsquittung zurück. Sie müssen darauf achten, dass die über die Parameter ADDR_1 und SD_1 definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

Falls keine Fehler auftraten, wird dies beim nächsten Aufruf am Zustandsparameter DONE mit "1" angezeigt. Eine erneute Aktivierung eines Schreibvorgangs ist erst nach dem Abschluss des letzten möglich.

Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden. Wenn beim Schreiben der Daten Zugriffsprobleme auftraten, oder die Prüfung der Ausführungsquittung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PUT_S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
ADDR_1	InOut	ANY	M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll.
SD_1	InOut	ANY	M, D	Zeiger auf diejenigen Bereiche in der eigenen CPU, die die zu versendenden Daten enthalten. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "PUT_S" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten.
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Sendebereichszeigern SD_1 bezüglich der Datenlänge oder des Datentyps
1	8	Zugriffsfehler bei der Partner-CPU
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB)
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "PUT_S" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut)
1	20	• Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) • Ist beim Erstaufruf möglich
1	27	Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Sendebereich SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

USEND_S: Daten unkoordiniert senden

Beschreibung

Die Anweisung "USEND_S" sendet **bei S7-300** Daten an eine remote Partner-Anweisung vom Typ "URCV_S (Seite 6418)". Der Sendevorgang verläuft ohne Koordination mit der Partner-Anweisung, d. h. die Datenübertragung erfolgt ohne Quittierung durch die Partner-Anweisung.

Der Sendevorgang erfolgt nach einer positiven Flanke an REQ. Mit jeder positiven Flanke an REQ werden die Parameter R_ID, ID und SD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID und SD_1 neue Werte zuweisen.

Sie müssen jedoch darauf achten, dass die über die Parameter SD_1 und RD_1 (bei der zugehörigen Partner-Anweisung "URCV_S (Seite 6418)") definierten Bereiche übereinstimmen in:

- Anzahl
- Länge und
- Datentyp.

Der Parameter R_ID muss bei beiden Anweisungen identisch sein.

Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit einer logischen "1" angezeigt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USEND_S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
SD_1	InOut	ANY	M, D	Zeiger auf den Sendebereich. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten
1	4	Fehler in den Sendebereichszeigern SD_1 bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "USEND_S" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • ein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	18	• R_ID existiert bereits in der Verbindung ID. • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich).

ERROR	STATUS (dezimal)	Erläuterung
1	20	• Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) • Ist beim Erstaufruf möglich
1	27	Für diese Anweisung existiert in der CPU kein Funktionscode.

Datenkonsistenz

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Sendebereich SD_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

URCV_S: Daten unkoordiniert empfangen

Beschreibung

Die Anweisung "URCV_S" empfängt **bei S7-300** asynchron Daten von einer remoten Partner-Anweisung vom Typ "USEND_S (Seite 6415)" und kopiert sie in die projektierten Empfangsbereiche.

Die Anweisung ist empfangsbereit, wenn am Eingang EN_R eine logische "1" anliegt. Mit EN_R=0 kann ein laufender Auftrag abgebrochen werden.

Mit jeder positiven Flanke an EN_R werden die Parameter R_ID, ID und RD_1 übernommen. Nach Abschluss eines Auftrags können Sie den Parametern R_ID, ID und RD_1 neue Werte zuweisen.

Sie müssen beachten, dass die über die Parameter RD_1 und SD_1 (bei der zugehörigen Partner-Anweisung "USEND_S (Seite 6415)") definierten Bereiche übereinstimmen in:

- Anzahl
- Länge
- Datentyp

Der erfolgreiche Abschluss des Kopiervorgangs wird am Zustandsparameter NDR mit einer logischen "1" angezeigt. Der Parameter R_ID muss bei beiden Anweisungen identisch sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "URCV_S":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
R_ID	Input	DWORD	E, A, M, D, L oder Konstante	Adressierungsparameter R_ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
RD_1	InOut	ANY	M, D	Zeiger auf den i-ten Empfangsbereich: Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (dezimal)	Erläuterung
0	9	Overrun-Warnung: Ältere Empfangsdaten wurden von neueren Empfangsdaten überschrieben.
0	11	Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote) • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP) • Verbindung zum Partner noch nicht aufgebaut • Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten
1	4	Fehler in den Empfangsbereichszeigern RD_1 bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).

ERROR	STATUS (dezimal)	Erläuterung
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "URCV_S" gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben. • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	18	• R_ID existiert bereits in der Verbindungs-ID • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.)
1	19	Die zugehörige Anweisung "USEND_S (Seite 6415)" sendet schneller Daten als diese von "URCV_S" in die Empfangsbereiche kopiert werden können.
1	20	• Maximale Anzahl paralleler Aufträge/Instanzen ist überschritten • Die Instanzen wurden bei CPU-RUN überladen (STOP-RUN-Übergang der CPU oder des CP ist erforderlich.) • Ist beim Erstaufruf möglich
1	27	Für diesen Baustein existiert in der CPU kein Funktionscode.

Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie Folgendes beachten:

Nachdem der Zustandsparameter NDR den Wert "1" angenommen hat, müssen Sie "URCV_S" mit dem Wert "0" an EN_R sofort erneut aufrufen. Damit wird gewährleistet, dass der Empfangsbereich nicht bereits überschrieben wird, bevor Sie ihn ausgewertet haben. Werten Sie den Empfangsbereich RD_1 vollständig aus, bevor Sie den Baustein wieder mit dem Wert "1" am Steuereingang EN_R aufrufen.

CONTROL: Verbindungszustand zur Instanz einer Anweisung abfragen

Beschreibung

Mit der Anweisung "CONTROL" ermitteln Sie **bei S7-400** den Zustand der Verbindung, die zu einer lokalen Kommunikations-Anweisungs-Instanz gehört.

Nach Aufruf der Anweisung mit dem Wert "1" am Steuereingang EN_R wird der momentane Zustand derjenigen Verbindung ermittelt, die zu der über I_DB selektierten Kommunikations-Anweisungs-Instanz gehört.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CONTROL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
I_DB	Input	WORD	E, A, M, D, L oder Konstante	Nummer des Instanz-DBs

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
OFFSET	Input	WORD	E, A, M, D, L oder Konstante	Offset des Datensatzes in Bytes im Multiinstanz-DB (falls kein Multiinstanz-DB vorliegt, ist hier "0" vorzugeben).
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
I_TYP	Output	BYTE	E, A, M, D, L	Kennung für den Bausteintyp, der zur selektierten Instanz gehört.
I_STATE	Output	BYTE	E, A, M, D, L	• = 0: Die zugehörige Anweisungs-Instanz wurde seit dem letzten Kaltstart, Neustart (Warmstart) oder Laden noch nie aufgerufen. • <> 0: Die zugehörige Anweisungs -Instanz wurde seit dem letzten Kaltstart, Neustart (Warmstart) oder Laden mindestens einmal aufgerufen.
I_CONN	Output	BOOL	E, A, M, D, L	Zustand der zugehörigen Verbindung. Mögliche Werte: • 0: Verbindung abgebrochen oder nicht aufgebaut. • 1: Verbindung vorhanden.
I_STATUS	Output	WORD	E, A, M, D, L	Zustandsparameter STATUS der abgefragten Kommunikations-Anweisungs-Instanz.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Ausgangsparameter I_TYP

Die folgende Tabelle erläutert, welche Kennung zu welchem Anweisungstyp gehört.

Anweisungstyp	Kennung (W#16#...)
USEND (Seite 6398)	00
URCV (Seite 6400)	01
BSEND (Seite 6403)	04
BRCV (Seite 6406)	05
GET (Seite 6392)	06
PUT (Seite 6395)	07
PRINT (Seite 6423)	08
START (Seite 6429)	0B
STOP (Seite 6431)	0C

Anweisungstyp	Kennung (W#16#...)
RESUME (Seite 6433)	0D
STATUS (Seite 6435)	0E
USTATUS (Seite 6437)	0F
ALARM (Seite 5650)	15
ALARM_8 (Seite 5643)	16
ALARM_8P (Seite 5645)	17
NOTIFY (Seite 5648)	18
AR_SEND (Seite 5671)	19
NOTIFY_8P (Seite 5640)	1A
(keine Anweisung vorhanden; I_DB oder OFFSET falsch)	FF

Parameter RET_VAL, ERROR und STATUS

Der Ausgangsparameter RET_VAL kann bei der Anweisung "CONTROL" die folgenden beiden Werte annehmen:

- 0000H: Bei der Ausführung ist kein Fehler aufgetreten.
- 8000H: Bei der Ausführung ist ein Fehler aufgetreten.

ERROR	STATUS (dezimal)	Erläuterung
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B.: Als Aktualparameter für I_TYP wurde ein Merkerbyte angegeben, das in der eingesetzten CPU nicht existiert.)
1	12	Beim Aufruf der Anweisung wurde: • am Parameter I_DB ein Instanz-DB angegeben, der nicht zu der Anweisung "CONTROL" gehört. • kein Instanz-DB, sondern ein Global-DB angegeben. • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).

Hinweis

Auch dann, wenn im Ausgangsparameter RET_VAL der Wert 0000H angezeigt wird, sind die Ausgangsparameter ERROR und STATUS auszuwerten.

PRINT: Daten an einen Drucker senden

Beschreibung PRINT

Beschreibung

Mit Hilfe der Anweisung "PRINT" können Sie **bei S7-400** Daten inklusive einer Formatierungsanweisung an einen remoten Drucker senden, z. B. mit Hilfe des CP 441.

Bei einer positiven Flanke am Steuereingang REQ werden die Formatbeschreibung (Parameter FORMAT) und die Daten (SD_i) an den über ID und PRN_NR selektierten Drucker gesendet.

Wenn Sie nicht alle vier Sendebereiche verwenden, müssen Sie darauf achten, dass der erste Bereich durch den Parameter SD_1, der zweite Bereich (falls vorhanden) durch SD_2, der dritte Bereich (falls vorhanden) durch SD_3 beschrieben wird.

Die erfolgreiche Durchführung des Auftrags wird am Zustandsparameter DONE mit "1" angezeigt, aufgetretene Fehler über die Zustandsparameter ERROR und STATUS.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PRINT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt • 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers
PRN_NR	InOut	BYTE	E, A, M, D, L	Druckernummer

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
FORMAT (Seite 6425)	InOut	STRING	E, A, M, D, L	Formatbeschreibung
SD_i (1≤ i ≤4)	InOut	ANY	M, D, T, Z	Zeiger auf den i-ten Sendebereich Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Hinweis:** Wenn der ANY-Pointer auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P# DB10.DBX5.0 Byte 10).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "PRINT" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).
1	2	Negative Quittung vom Drucker. Die Funktion ist nicht ausführbar.
1	3	PRN_NR ist auf der durch ID vorgegebenen Kommunikationsverbindung nicht bekannt.
1	4	Fehler im Durchgangsparameter FORMAT oder in den Sendebereichszeigern SD_i bezüglich der Datenlänge oder des Datentyps.
1	6	Der remote Drucker befindet sich im Zustand OFFLINE.
1	7	Der remote Drucker befindet sich in einem falschen Zustand (z. B. Paper out).
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde: • ein Instanz-DB, der nicht zu "PRINT" gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben. • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	13	Fehler im Durchgangsparameter FORMAT (Seite 6425).
1	20	**S7-400** Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

Anzahl übertragbarer Daten

Die Anzahl der an einen remoten Drucker zu übertragenden Daten darf eine maximale Länge nicht überschreiten. Diese maximale Datenlänge in Byte berechnet sich wie folgt:
`maxleng = 420 - format`

Dabei ist `format` die aktuelle Länge des Parameters FORMAT in Byte. Es ist unerheblich, ob Sie die zu druckenden Daten auf einen oder auf mehrere Sendebereiche aufteilen.

Parameter FORMAT

Beschreibung

Die Zeichenkette FORMAT enthält zu druckende Zeichen und Formatelemente. Sie hat folgenden Aufbau:

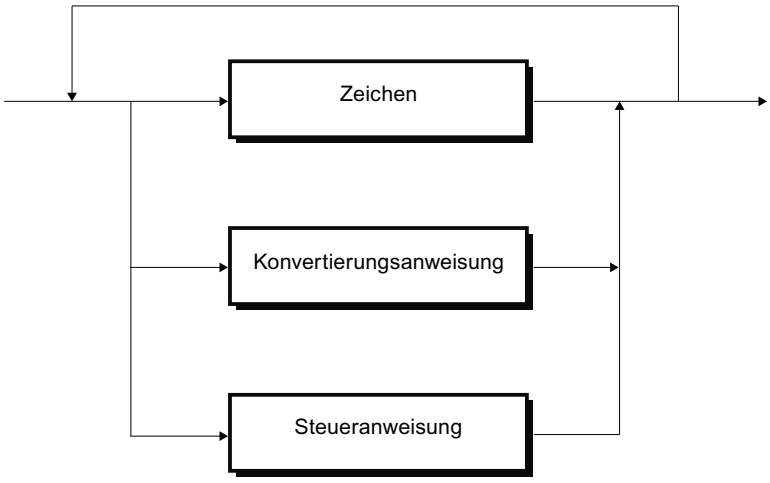

Bild 4-28 Aufbau Zeichen und Formatelemente

Für jeden zu druckenden Sendebereich SD_1 bis SD_4 muss genau eine Konvertierungsanweisung in FORMAT vorhanden sein. Die Konvertierungsanweisungen werden entsprechend ihrer Reihenfolge auf die Sendebereiche SD_i angewendet. Darüber hinaus können Zeichen und Anweisungen beliebig aufeinanderfolgen.

Zeichen

Zulässig sind:

- alle druckbaren Zeichen
- $$ (Dollar-Zeichen), $' (einfaches Anführungszeichen), $L und $l (line feed), $P und $p (page), $R und $r (carriage return), $T und $t (Tabulator)

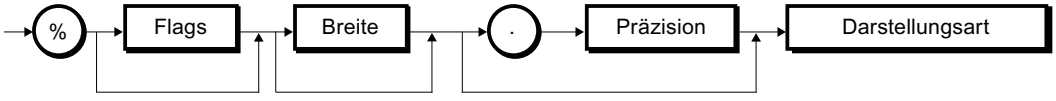

Bild 4-29 Syntaxdiagramm einer Konvertierungsanweisung

Element einer Konvertierungsanweisung	Bedeutung	
Flags	• Ohne: • -:	rechtsbündige Ausgabe linksbündige Ausgabe
Breite	• ohne: • n:	Ausgabe erfolgt in der Standarddarstellung Es werden genau n Zeichen ausgegeben. Bei rechtsbündiger Ausgabe werden ggf. Leerzeichen vorangestellt, bei linksbündiger Ausgabe werden sie hinten angestellt.
Präzision	Die Präzision ist nur bei den Darstellungsarten A, D, F und R (siehe folgende Tabelle) relevant.	
	• Ohne: • 0: • n:	Ausgabe erfolgt in der Standarddarstellung keine Ausgabe des Dezimalpunkts und von Nachkommastellen bei den Darstellungsarten F und R • bei F und R: Ausgabe des Dezimalpunkts und von n Nachkommastellen. • bei A und D (Datum): Anzahl der Stellen bei der Jahreszahl. Mögliche Werte: 2 und 4.
Darstellungsart	Die folgende Tabelle enthält • die möglichen Darstellungsarten • die für jede Darstellungsart möglichen Datentypen • für jede Darstellungsart die Standarddarstellung (die Druckerausgabe erfolgt in der Standarddarstellung, falls Sie im Parameter FORMAT keine Breite und keine Präzision angeben) und deren maximale Länge.	

Konvertierungsanweisung

Die folgende Tabelle zeigt mögliche Darstellungsarten in der Konvertierungsanweisung des Parameters FORMAT.

Darstellungsart	dafür mögliche Datentypen	Beispiel	Länge	Bemerkungen
A, a	DATE	25.07.1996	10	-
	DWORD			
C, c	CHAR	K	1	-
	BYTE	M	1	
	WORD	KL	2	
	DWORD	KLMN	4	
	ARRAY of CHAR	KLMNOP	Anzahl der Zeichen	
	ARRAY of BYTE			
D, d	DATE	1996-07-25	10	-
	DWORD			
F, f	REAL	0.345678	8	-
	DWORD			

Darstellungsart	dafür mögliche Datentypen	Beispiel	Länge	Bemerkungen
H, h	alle Datentypen incl. ARRAY of BYTE	je nach Datentyp	je nach Datentyp	hexadezimale Darstellung
I, i	INT	- 32 768	max. 6	-
	WORD	- 2 147 483 648	max. 11	
N, n	WORD	Textausgabe	-	Der zugehörige Sendebereich SD_i enthält einen Verweis (Nummer) auf einen zu druckenden Text. Der Text liegt auf der Baugruppe (z. B. CP 441), die eine druckbare Zeichenkette erzeugt. Wird unter der angegebenen Nummer kein Text gefunden, wird **** ausgegeben.
R, r	REAL	0.12E-04	8	-
	DWORD			
S, s	STRING	Textausgabe		-
T, t	TIME	2d_3h_10m_5s_250ms	max. 21	Im Fehlerfall wird **** ausgegeben.
	DWORD			
U, u	BYTE	255	max. 3	-
	WORD	65 535	max. 5	
	DWORD	4 294 967 295	max. 10	
X, x	BOOL	1	1	-
	BYTE	101 ..	8	
	WORD	101 ..	16	
	DWORD	101 ..	32	
Z, z	TIME_OF_DAY (TOD)	15:38:59.874	12	-

An den Stellen dieser Tabelle, an denen eine maximale Länge der Standarddarstellung angegeben ist, kann die tatsächliche Ausgabe auch kürzer sein.

Hinweis

Bei den Datenarten C und S hängt es vom eingesetzten Drucker ab,

- welche Zeichen druckbar sind
- was der Drucker bei nicht druckbaren Zeichen ausgibt, es sei denn, der Druckertreiber enthält eine Umwandlungstabelle für diese Zeichen.

Steueranweisung

Mit Hilfe der Steueranweisungen können Sie:

- die Zeichen % und \ drucken
- Druckereinstellungen ändern.

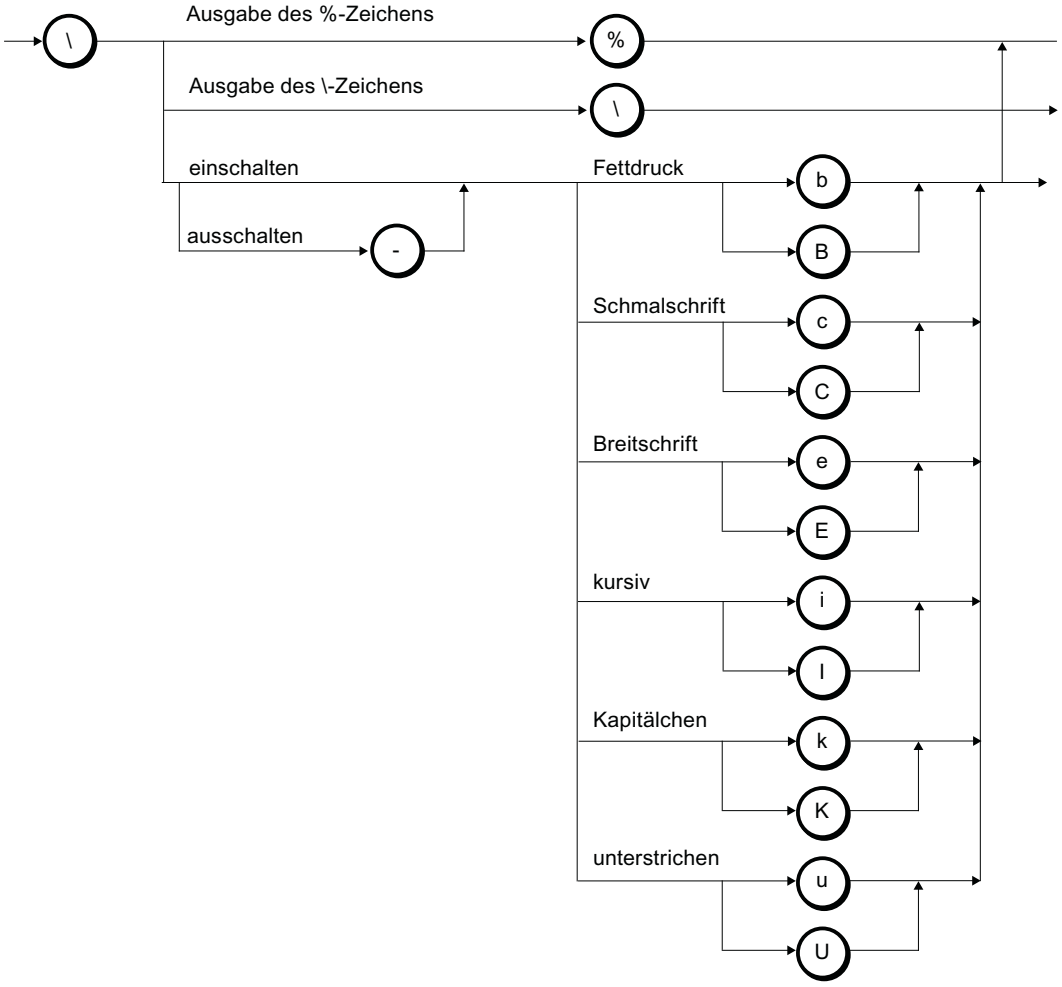

Bild 4-30 Syntaxdiagramm der Steueranweisung

Wird das Ausschalten z. B. eines Schrifttyps angefordert, obwohl dieser gar nicht eingeschaltet ist, oder soll eine Funktion durchgeführt werden, die der Drucker gar nicht kennt, wird die Steueranweisung ignoriert. Die folgende Tabelle enthält die Fehlerfälle beim Durchgangsparameter FORMAT.

Fehlerfall	Druckerausgabe
Konvertierungsanweisung nicht ausführbar	Entsprechend der (max.) Länge der Standarddarstellung bzw. der angegebenen Breite werden *-Zeichen ausgegeben.
Angegebene Breite zu gering	Bei den Darstellungsarten A, C, D, N, S, T, Z werden so viele Zeichen ausgegeben, wie es die angegebene Breite vorgibt. Bei allen anderen Darstellungsarten werden entsprechend der angegebenen Breite*-Zeichen ausgegeben.
Zu viele Konvertierungsanweisungen	Die Konvertierungsanweisungen, zu denen kein Sendebereichszeiger SD_i gehört, werden ignoriert.

Fehlerfall	Druckerausgabe
Zu wenig Konvertierungsanweisungen	Sendebereiche, zu denen keine Konvertierungsanweisung vorliegt, werden nicht ausgegeben.
Nicht definierte oder nicht unterstützte Konvertierungsanweisungen	Es wird ****** ausgegeben.
Unvollständige Konvertierungsanweisung	Es wird ****** ausgegeben.
Nicht definierte oder nicht unterstützte Steueranweisungen	Steueranweisungen, die nicht der Syntax des vorhergehenden Bildes entsprechen, werden ignoriert.

START: In einem remoten Gerät einen Neustart oder Kaltstart durchführen

Beschreibung

Bei einer positiven Flanke am Steuereingang REQ aktiviert die Anweisung "START" **bei S7-400** einen Neustart (Warmstart) oder einen Kaltstart in dem durch ID adressierten remoten Gerät.

Weitere Information zum Thema Warm- und Kaltstart finden Sie hier: Kalt- und Warmstart (http://support.automation.siemens.com/WW/view/de/34053759)

Folgende Bedingungen müssen zur Ausführung im remoten Gerät (falls dies eine CPU ist) vorliegen:

- Die CPU muss sich im Zustand STOP befinden.
- Der Betriebsartenschalter der CPU muss auf "RUN" stehen.

Nachdem dieses den Neustart (Warmstart) oder einen Kaltstart durchlaufen hat, geht es in den Betriebszustand RUN und sendet eine positive Ausführungsquittung. Mit der Auswertung der positiven Quittung wird der Zustandsparameter DONE auf "1" gesetzt. Aufgetretene Fehler werden über die Zustandsparameter ERROR und STATUS angezeigt.

Eine erneute Aktivierung eines Neustarts (Warmstarts) bzw. Kaltstarts in demselben remoten Gerät ist erst nach Abschluss der letzten Aktivierung möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "START":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert die Anweisung bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
PI_NAME	InOut	ANY	E, A, M, D, T, Z	Zeiger auf Speicherbereich, in dem der Name des zu startenden Programms (ASCII-Code) steht. Dieser Name darf aus maximal 32 Zeichen bestehen. Bei einem Standardsystem aus der S7-Familie muss er P_PROGRAM sein.
ARG	InOut	ANY	E, A, M, D, T, Z	Ausführungsargument. • Falls Sie ARG keinen Wert zuweisen, wird im remoten Gerät ein Neustart (Warmstart) durchgeführt. • Falls Sie den Wert "C" zuweisen, wird im remoten Gerät ein Kaltstart durchgeführt (falls das remote Gerät diese Anlaufart kennt).
IO_STATE	InOut	BYTE	E, A, M, D, L	Derzeit nicht relevant. Sie dürfen diesem Parameter derzeit keinen Wert zuweisen, wenn Ihr Kommunikationspartner ein Automatisierungssystem aus der S7-Familie ist.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für START spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	3	Der bei PI_NAME hinterlegte Programmname ist unbekannt.
1	4	Fehler bei den Zeigern PI_NAME oder ARG bezüglich der Datenlänge oder des Datentyps.

ERROR	STATUS (dezimal)	Erläuterung
1	7	Im Partnergerät ist kein Neustart durchführbar.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "START" gehört, angegeben • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	S7-400: Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

STOP: Remotes Gerät in den Betriebszustand STOP überführen

Beschreibung

Bei einer positiven Flanke am Steuereingang REQ aktiviert die Anweisung "STOP" **bei S7-400** die Überführung des durch ID adressierten remoten Geräts in den Betriebszustand STOP; der Betriebszustandsübergang ist möglich, wenn sich dieses in RUN, HALT oder Anlauf befindet.

Die erfolgreiche Durchführung des Auftrags wird am Zustandsparameter DONE mit "1" angezeigt, aufgetretene Fehler über die Zustandsparameter ERROR und STATUS.

Eine erneute Durchführung des beschriebenen Betriebszustandsübergangs in demselben remoten Gerät ist erst dann möglich, wenn der vorherige Anstoß von der Anweisung "STOP" abgeschlossen ist.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "STOP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert die Anweisung bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
PI_NAME	InOut	ANY	E, A, M, D	Zeiger auf Speicherbereich, in dem der Name des zu stoppenden Programms (ASCII-Code) steht. Dieser Name darf aus maximal 32 Zeichen bestehen. Bei einem Standardsystem aus der S7-Familie muss er P_PROGRAM sein.
IO_STATE	InOut	BYTE	E, A, M, D, L	Derzeit nicht relevant. Sie dürfen diesem Parameter derzeit keinen Wert zuweisen, wenn Ihr Kommunikationspartner ein Automatisierungssystem aus der S7-Familie ist.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "STOP" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	3	Der bei PI_NAME hinterlegte Programmname ist unbekannt.
1	4	Fehler beim Zeiger PI_NAME bezüglich der Datenlänge oder des Datentyps.
1	7	Das Partnergerät befindet sich bereits im Zustand STOP.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).

ERROR	STATUS (dezimal)	Erläuterung
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu STOP gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	S7-400: Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

RESUME: In einem remoten Gerät einen Wiederanlauf durchführen

Beschreibung

Bei einer positiven Flanke am Steuereingang REQ aktiviert die Anweisung "RESUME" **bei S7-400** einen Wiederanlauf in dem durch ID adressierten remoten Gerät. Folgende Bedingungen müssen vorliegen, falls das remote Gerät eine CPU ist:

- Die CPU muss sich im Zustand STOP befinden.
- Der Betriebsartenschalter der CPU muss auf RUN stehen.
- Sie müssen bei der Projektierung manuellen Wiederanlauf ermöglicht haben.
- Es darf kein Wiederanlaufhindernis vorliegen.

Nachdem dieses den Wiederanlauf durchgeführt hat, geht es in den Betriebszustand RUN und sendet eine positive Ausführungsquittung. Mit der Auswertung der positiven Quittung wird der Zustandsparameter DONE auf "1" gesetzt. Aufgetretene Fehler werden über die Zustandsparameter ERROR und STATUS angezeigt.

Eine erneute Aktivierung des Wiederanlaufs in demselben remoten Gerät ist erst nach Abschluss der letzten Aktivierung möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RESUME":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert die Anweisung bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
PI_NAME	InOut	ANY	E, A, M, D	Zeiger auf Speicherbereich, in dem der Name des zu startenden Programms (ASCII-Code) steht. Dieser Name darf aus maximal 32 Zeichen bestehen. Bei S7 muss er P_PROGRAM sein.
ARG	InOut	ANY	E, A, M, D, T, Z	Ausführungsargument. Derzeit nicht relevant. Sie dürfen diesem Parameter derzeit keinen Wert zuweisen, wenn Ihr Kommunikationspartner ein Automatisierungssystem aus der S7-Familie ist.
IO_STATE	InOut	BYTE	E, A, M, D, L	Derzeit nicht relevant. Sie dürfen diesem Parameter derzeit keinen Wert zuweisen, wenn Ihr Kommunikationspartner ein Automatisierungssystem aus der S7-Familie ist.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "RESUME" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	• Kommunikationsprobleme, z. B. Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	3	Der bei PI_NAME hinterlegte Programmname ist unbekannt.
1	4	Fehler bei den Zeigern PI_NAME oder ARG bezüglich der Datenlänge oder des Datentyps.
1	7	Wiederanlauf nicht durchführbar.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).

ERROR	STATUS (dezimal)	Erläuterung
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu RESUME gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	S7-400: Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

STATUS: Gerätestatus eines remoten Partners abfragen

Beschreibung

Mit Hilfe der Anweisung "STATUS" können Sie **bei S7-400** den Gerätestatus eines remoten Kommunikationspartners abfragen.

Bei einer positiven Flanke am Steuereingang REQ wird ein Auftrag an den remoten Partner gesendet. Die Rückantwort wird daraufhin ausgewertet, ob Probleme aufgetreten sind. Falls keine Fehler auftraten, wird der empfangene Zustand beim nächsten Aufruf in die Variablen der Parameter PHYS, LOG und LOCAL kopiert. Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt.

Eine erneute Abfrage bei demselben Kommunikationspartner ist erst nach Abschluss der letzten möglich.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "STATUS":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert die Anweisung bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler <> 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
PHYS	InOut	ANY	E, A, M, D	Physikalischer Zustand (Mindestlänge: ein Byte) Mögliche Werte: • 10H voll funktionsfähig. • 13H Service erforderlich.
LOG	InOut	ANY	E, A, M, D	Logischer Zustand (Mindestlänge: ein Byte) Möglicher Wert: • 00H Zustandswechsel erlaubt.
LOCAL	InOut	ANY	E, A, M, D	Betriebszustand, falls das Partnergerät eine S7-CPU ist (Mindestlänge: zwei Byte).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Durchgangsparameter LOCAL

Falls der Kommunikationspartner eine S7-CPU ist, beinhaltet der Durchgangsparameter LOCAL deren aktuellen Betriebszustand: Das erste Byte ist reserviert, das zweite Byte enthält eine Kennung für den Betriebszustand.

Betriebszustand	Zugehörige Kennung
STOP	00H
ANLAUF (Neustart)	01H
RUN	02H
ANLAUF (Wiederanlauf)	03H
HALT	04H
ANLAUF (Kaltstart)	06H
RUN	09H
ANKOPPELN	0BH
AUFDATEN	0CH

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für STATUS spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	11	Warnung: • Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist. • Der Auftrag wird bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).

ERROR	STATUS (dezimal)	Erläuterung
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler bei PHYS, LOG oder LOCAL bezüglich der Datenlänge oder des Datentyps.
1	8	Zugriff auf remotes Objekt wurde abgelehnt.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu STATUS gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut).
1	20	S7-400: Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

USTATUS: Unkoordiniertes Empfangen eines remoten Gerätestatus

Beschreibung

Die Anweisung "USTATUS" empfängt **bei S7-400** den Wechsel des Gerätestatus eines remoten Kommunikationspartners; dieser sendet seinen Status bei Änderung unaufgefordert, falls Sie dies projektiert haben.

Falls beim Aufruf am Steuereingang EN_R 1 anliegt und ein Telegramm des Partners vorliegt, wird beim nächsten Aufruf die Statusinformation in die Variablen der Parameter PHYS, LOG und LOCAL abgelegt. Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt.

Auf der von der Anweisung "USTATUS" benutzten Verbindung muss das Versenden von Betriebszustandsmeldungen freigegeben sein.

Hinweis

Pro Verbindung dürfen Sie nur eine Instanz der Anweisung "USTATUS" platzieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USTATUS":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L	Steuerparameter (enabled to receive), signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	Input	WORD	M, D oder Konstante	Adressierungsparameter ID Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 6384)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • 0: Auftrag wurde noch nicht gestartet oder läuft noch. • 1: Auftrag wurde erfolgreich abgeschlossen.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR STATUS	Output Output	BOOL WORD	E, A, M, D, L E, A, M, D, L	Zustandsparameter ERROR und STATUS, Fehleranzeige: • ERROR=0 STATUS hat den Wert: 0000H: weder Warnung noch Fehler 0000H: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=1 Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.
PHYS	InOut	ANY	E, A, M, D	Physikalischer Zustand (Mindestlänge: ein Byte) Mögliche Werte: • 10H voll funktionsfähig. • 13H Service erforderlich.
LOG	InOut	ANY	E, A, M, D	Logischer Zustand (Mindestlänge: ein Byte) Möglicher Wert: • 00H Zustandswechsel erlaubt.
LOCAL	InOut	ANY	E, A, M, D	Betriebszustand, falls das Partnergerät eine S7-CPU ist (Mindestlänge: zwei Byte).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Durchgangsparameter LOCAL

Falls der Kommunikationspartner eine S7-CPU ist, beinhaltet der Durchgangsparameter LOCAL deren aktuellen Betriebszustand: Das erste Byte ist reserviert, das zweite Byte enthält eine Kennung für den Betriebszustand.

Betriebszustand	Zugehörige Kennung
STOP	00H
ANLAUF (Neustart)	01H
RUN	02H
ANLAUF (Wiederanlauf)	03H
HALT	04H
ANLAUF (Kaltstart)	06H
RUN	09H
ANKOPPELN	0BH
AUFDATEN	0CH

Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "USTATUS" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

ERROR	STATUS (dezimal)	Erläuterung
0	9	Overrun-Warnung: Ein älterer Gerätestatus wurde von einem neueren Gerätestatus überschrieben.
0	11	Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z. B. • Verbindungsbeschreibung nicht geladen (lokal oder remote). • Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP).
1	4	Fehler bei PHYS, LOG oder LOCAL bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf der Anweisung wurde • ein Instanz-DB, der nicht zu "USTATUS" gehört, angegeben. • kein Instanz-DB, sondern ein Global-DB angegeben • kein Instanz-DB gefunden (Abhilfe: Laden Sie den zugehörigen Instanz-DB erneut)
1	18	Für die durch ID vorgegebene Verbindung gibt es bereits eine Instanz zu "USTATUS".
1	19	Die remote CPU sendet schneller Daten, als die Anweisung diese in das Anwenderprogramm übernehmen kann.
1	20	S7-400: Zu wenig Arbeitsspeicher vorhanden. Abhilfe: Reduzieren Sie den im Speicher befindlichen Programmcode.

4.2.5.3 Open User Communication

Übersicht

Open User Communication über Industrial Ethernet

Um mit anderen Kommunikationspartnern via Ethernet (TCP-, ISO-on-TCP-, UDP-Protokoll) per Anwenderprogramm Daten austauschen zu können, werden Ihnen die folgenden Anweisungen und UDTs zur Verfügung gestellt:

- Verbindungsorientierte Protokolle: TCP gemäß RFC 793, ISO on TCP gemäß RFC 1006:
  - UDT 65 "TCON_PAR (Seite 6450)" mit der Datenstruktur zur Verbindungsparametrierung
  - UDT 651 bis UDT 656 mit protokollspezifischen Vorbesetzungen
  - "TCON (Seite 6463)" zum Verbindungsaufbau
  - "TDISCON (Seite 6466)" zum Verbindungsabbau
  - "TSEND (Seite 6468)" zum Senden von Daten
  - "TRCV (Seite 6471)" zum Empfangen von Daten
- Verbindungsloses Protokoll: UDP gemäß RFC 768
  - UDT 65 "TCON_PAR (Seite 6450)" mit der Datenstruktur zur Parametrierung des lokalen Kommunikationszugangspunkts
  - UDT 657 mit protokollspezifischen Vorbesetzungen
  - UDT 66 "TCON_ADR (Seite 6452)" mit der Datenstruktur der Adressierungsparameter des remoten Partners
  - UDT 661 mit protokollspezifischen Vorbesetzungen
  - "TCON (Seite 6463)" zur Einrichtung des lokalen Kommunikationszugangspunkts
  - "TDISCON (Seite 6466)" zur Auflösung des lokalen Kommunikationszugangspunkts
  - "TUSEND (Seite 6475)" zum Senden von Daten
  - "TURCV (Seite 6478)" zum Empfangen von Daten

Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet

Verbindungsorientierte und verbindungslose Protokolle

Man unterscheidet in der Datenkommunikation zwischen folgenden beiden Protokollvarianten:

- Verbindungsorientierte Protokollvarianten:
  Diese bauen vor der Datenübertragung eine (logische) Verbindung zum Kommunikationspartner auf und bauen diese nach Abschluss der Datenübertragung ggf. wieder ab. Verbindungsorientierte Protokolle werden eingesetzt, wenn es bei der Datenübertragung insbesondere auf Sicherheit ankommt. Über eine physikalische Leitung können in der Regel mehrere logische Verbindungen bestehen.
  Bei den Anweisungen zur Open User Communication über Industrial Ethernet werden die folgenden verbindungsorientierten Protokolle unterstützt:
  - TCP gemäß RFC 793
  - ISO on TCP gemäß RFC 1006
- Verbindungslose Protokollvarianten:
  Diese arbeiten ohne Verbindung. Der Verbindungsauf- und der Verbindungsabbau zum remoten Partner entfallen also. Verbindungslose Protokolle übertragen die Daten unquittiert und damit ungesichert zum remoten Partner, d. h. es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird.
  Bei den Anweisungen zur Open User Communication über Industrial Ethernet wird das folgende verbindungslose Protokoll unterstützt:
  - UDP gemäß RFC 768 (Broadcast oder Singlecast, keine Unterstützung von Multicast)

Die Arbeitsweise der Anweisungen ist abhängig von der benutzten Protokollvariante. Darauf wird im Folgenden näher eingegangen.

Empfangsbereich

Dieser Begriff wird im Folgenden immer wieder verwendet. Man versteht darunter den Bereich, in den die Anweisung die empfangenen Daten einträgt.

Der Empfangsbereich ist durch die folgenden beiden Größen festgelegt:

- Zeiger auf den Beginn des Bereichs
- Länge des Bereichs

Die Länge des Bereichs wird abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) festgelegt.

Im ANY-Pointer sind die folgenden Datentypen zulässig: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME.

TCP

Bei der Datenübertragung werden weder Informationen zur Länge noch über Anfang und Ende einer Nachricht übertragen. Beim Senden ist dies unproblematisch, da der Sender weiß, wie viele Datenbytes er verschicken will. Der Empfänger hat jedoch keine Möglichkeit zu erkennen, wo eine Nachricht im Datenstrom endet und wo die nächste beginnt. Es wird daher empfohlen, den Empfangsbereich der Anweisung "TRCV (Seite 6471)" genau so groß zu wählen wie den Wert des Parameters LEN der Anweisung "TSEND (Seite 6468)" beim Kommunikationspartner (Anzahl der Bytes, die gesendet werden sollen).

- Datenempfang im Ad-hoc-Mode:
  Der Empfangsbereich ist identisch mit dem durch den Parameter DATA der Anweisung "TRCV (Seite 6471)" vorgegebenen Bereich.
  - Unmittelbar nach Empfang eines Datenblocks überträgt die Anweisung "TRCV (Seite 6471)" diesen in den Empfangsbereich und setzt NDR auf "1". Die maximale Länge beträgt 8192 Byte.
  - Falls Sie die Länge des Empfangsbereichs größer gewählt haben als die Länge der gesendeten Daten, kopiert die Anweisung "TRCV (Seite 6471)" die empfangenen Daten vollständig in den Empfangsbereich. Anschließend setzt sie NDR auf TRUE und beschreibt RCVD_LEN mit der Länge der empfangenen Daten.
  - Falls Sie die Länge des Empfangsbereichs kleiner gewählt haben als die Länge der gesendeten Daten, kopiert "TRCV (Seite 6471)" so viele Bytes in den Empfangsbereich, bis dieser voll ist. Anschließend setzt er NDR auf TRUE und beschreibt RCVD_LEN mit der Länge des Empfangsbereichs. Mit jedem weiteren Aufruf erhalten Sie einen weiteren Block der gesendeten Daten.
- Datenempfang mit vorgegebener Länge:
  - Der Empfangsbereich wird durch die Parameter DATA (Anfangsadresse des Empfangsbereichs) und LEN (Länge des Empfangsbereichs) von "TRCV (Seite 6471)" gebildet.
  - Falls Daten empfangen werden, die den Empfangsbereich nicht vollständig füllen, werden Ihnen diese Daten zunächst nicht zur Verfügung gestellt. Sie werden Ihnen erst dann zur Verfügung gestellt, wenn weitere Daten den Empfangsbereich vollständig gefüllt haben. Bitte beachten Sie, dass sich in diesem Fall Daten aus zwei unterschiedlichen Sendeaufträgen in ein und demselben Empfangsbereich befinden. Wenn Sie das Ende der ersten Nachricht bzw. den Anfang der zweiten Nachricht nicht feststellen können, haben sie keine Möglichkeit, die erste bzw. zweite Nachricht zu erkennen.
  - Falls Sie die Länge des Empfangsbereichs kleiner gewählt haben als die Länge der gesendeten Daten, kopiert "TRCV (Seite 6471)" so viele Bytes in den Empfangsbereich, bis dieser vollständig gefüllt ist.
  - Wenn "TRCV (Seite 6471)" den Empfangsbereich vollständig gefüllt hat, setzt er NDR auf TRUE und beschreibt RCVD_LEN mit dem Wert von LEN. Mit jedem weiteren Aufruf erhalten Sie damit einen weiteren Block der gesendeten Daten.

ISO on TCP

Bei der Datenübertragung werden Informationen zur Länge und zum Ende einer Nachricht übertragen.

Falls Sie den Empfangsbereich größer gewählt haben als die Länge der gesendeten Daten, kopiert die Anweisung "TRCV (Seite 6471)" die gesendeten Daten vollständig in den Empfangsbereich. Anschließend setzt sie NDR auf TRUE und beschreibt RCVD_LEN mit der Länge der gesendeten Daten.

Falls Sie den Empfangsbereich kleiner gewählt haben als die Länge der gesendeten Daten, kopiert "TRCV (Seite 6471)" keine Daten in den Empfangsdatenbereich, sondern liefert folgende Fehlerinformation: ERROR=1, STATUS=W#16#8088.

UDP

Sie führen keinen Verbindungsaufbau wie bei den Protokollen TCP und ISO on TCP durch. Daher müssen Sie beim Aufruf des Sendebausteins "TUSEND (Seite 6475)" einen Verweis auf die Adressparameter des Empfängers (IP-Adresse und Port-Nr.) angeben. Analog erhalten Sie nach Abschluss des Empfangsbausteins "TURCV (Seite 6478)" einen Verweis auf die Adressparameter des Senders (IP-Adresse und Port-Nr.).

- Damit Sie die Anweisungen "TUSEND (Seite 6475)" und "TURCV (Seite 6478)" nutzen können, müssen Sie zuvor sowohl auf der Sender- als auch auf der Empfängerseite die Anweisung "TCON (Seite 6463)" aufrufen, um den lokalen Kommunikationszugangspunkt einzurichten.
- Bei jedem Aufruf von "TUSEND (Seite 6475)" können Sie den remoten Partner durch Angabe seiner IP-Adresse und seiner Port-Nr. neu referenzieren.
- Bei der Datenübertragung werden Informationen zur Länge und zum Ende einer Nachricht übertragen.

Falls Sie den Empfangsbereich größer gewählt haben als die Länge der gesendeten Daten, kopiert "TURCV (Seite 6478)" die gesendeten Daten vollständig in den Empfangsbereich. Anschließend setzt er NDR auf TRUE und beschreibt RCVD_LEN mit der Länge der gesendeten Daten.

Falls Sie den Empfangsbereich kleiner gewählt haben als die Länge der gesendeten Daten, kopiert TURCV keine Daten in den Empfangsbereich, sondern liefert folgende Fehlerinformation: ERROR = 1, STATUS = W#16#8088.

Port-Nummern bei TCP und UDP

Die Adressierung einer Verbindung geschieht wie folgt:

- Bei TCP:
  - Aktive Verbindung: Über die remote IP-Adresse, den remoten Port und über den lokalen Port (Die Port-Nummer des lokalen Ports geben Sie entweder explizit vor, oder sie wird vom Betriebssystem vergeben.)
  - Passive Verbindung: Über den lokalen Port. Falls Sie den lokalen Port mehrfach benutzen wollen, müssen Sie entweder die remote IP-Adresse oder die remote IP-Adresse und den remoten Port mit vorgeben.
- Bei UDP: Über die remote IP-Adresse, den remoten Port und den lokalen Port (die Port-Nummer des lokalen Ports geben Sie explizit vor.)

Für den lokalen Port gelten die im Folgenden genannten Bereiche. Für den remoten Port gelten diese Bereiche nur dann, wenn der remote Partner eine S7-CPU ist. Andernfalls gelten für den remoten Port keine Einschränkungen.

- Für CPUs 31x-2 PN/DP bis einschl. FW-Stand V2.6 und CPUs 41x-3 PN/DP bis einschließlich FW-Stand V5.1:
  - Zulässige Port-Nummern in der Parametrierung (UDT 65): 2000 bis 5000
  - Bei Parametrierungen ohne explizite Vorgabe einer Port-Nummer vergibt das Betriebssystem einen Port aus dem dynamischen Nummernband zwischen 49152 und 65534.
- Für CPUs 31x-2 PN/DP ab FW-Stand V2.7 und CPUs 41x-3 PN/DP ab FW-Stand V5.2:
  - Zulässige Port-Nummern in der Parametrierung (UDT 65): 1 bis 49151
    Es wird empfohlen, sich auf die Port-Nummern 2000 bis 5000 zu beschränken, da Siemens sich vorbehält, Ports aus dem Nummernbereichen 1 bis 1999 und 5001 bis 49151 systemseitig zu verwenden.
  - Bei Parametrierungen ohne explizite Vorgabe einer Port-Nummer vergibt das Betriebssystem einen Port aus dem dynamischen Nummernband zwischen 49152 und 65534.
- Für CPUs 31x-2 PN/DP ab FW-Stand V3.2 und CPUs 41x-3 PN/DP ab FW-Stand V6.0 und CPUs 41x-5H PN/DP ab FW-Stand V6.0:
  - Die Mehrfachnutzung eines Ports ist möglich.

Die systemseitig reservierten Port-Nummern entnehmen Sie der folgenden Liste:

Protokoll	Port-Nummer	Dienst
-	0	(Dieser Port ist fest vergeben. Er darf nicht verwendet werden.)
TCP	20 und 21	FTP
TCP	25	SMTP
TCP	80	HTTP
TCP	102	RFC 1006
UDP	135	RPC-DCOM
UDP	161	SNMP REQUEST
TCP, UDP	34962	PNIO
TCP, UDP	34963	PNIO
TCP, UDP	34964	PNIO
UDP	65532	NTP
UDP	65533	NTP
UDP	65534	NTP
UDP	65535	NTP

Siehe dazu auch http://www.iana.org/assignments/port-numbers.

Hinweis

Verwenden Sie keine systemseitig reservierten Port-Nummern.

Identische Portnummern dürfen nicht für den aktiven und passiven Verbindungsaufbau in einem Gerät verwendet werden.

Parametrierung der Kommunikationsverbindungen bei TCP und ISO on TCP

Datenbaustein für Verbindungsparametrierung

Um die Kommunikationsverbindungen bei TCP und ISO on TCP zu parametrieren, legen Sie einen DB an, der die Datenstruktur aus dem UDT 65 "TCON_PAR" enthält. Diese Datenstruktur enthält die notwendigen Parameter, die Sie zum Aufbau der Verbindung benötigen. Für jede Verbindung benötigen Sie solch eine Datenstruktur, die Sie auch in einem globalen DB zusammenfassen können.

Der Verbindungsparameter CONNECT der Anweisung "TCON (Seite 6463)" enthält einen Verweis auf die Adresse der zugehörigen Verbindungsbeschreibung (z. B. P#DB100.DBX0.0 Byte 64).

Aufbau der Verbindungsbeschreibung (UDT 65)

Byte	Parameter	Datentyp	Startwert	Beschreibung
0 bis 1	block_length	WORD	W#16#40	Länge des UDT 65: 64 Bytes (fest)
2 bis 3	id	WORD	W#16#0001	Referenz auf diese Verbindung (Wertebereich: W#16#0001 bis W#16#0FFF) Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei ID angeben.
4	connection_type	BYTE	B#16#11	Protokollvariante: • B#16#11: TCP • B#16#12: ISO on TCP • B#16#01: TCP (Kompatibilitätsmode)
5	active_est	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus: • FALSE: passiver Verbindungsaufbau • TRUE: aktiver Verbindungsaufbau

Byte	Parameter	Datentyp	Startwert	Beschreibung
6	local_device_id	BYTE	B#16#02	• B#16#00: Kommunikation über CP 443-1EX (nur bei S7-400 und connection_type = B#16#12). Zulässige CPs: CP443-1EX4x, CP443-1EX20, CP443-1GX20, CP443-1EX30, CP443-1GX30 • B#16#01: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 1 (IF1) bei WinAC RTX (nur TCP) • B#16#02: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 315-2 PN/DP und 317-2 PN/DP • B#16#03: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 319-3 PN/DP • B#16#05: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 414-3 PN/DP, 416-3 PN/DP, 416-3F PN/DP und 41x-5H PN/DP (Rack 0) • B#16#06: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 2 (IF2) bei WinAC RTX (nur TCP) • B#16#0B: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 3 (IF3) bei WinAC RTX (nur TCP) • B#16#0F: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 4 (IF4) bei WinAC RTX (nur TCP) • B#16#15: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 41x-5H PN/DP (Rack 1)
7	local_tsap_id_len	BYTE	B#16#02	Verwendete Länge des Parameters local_tsap_id; mögliche Werte: • 0 oder 2, falls connection_type = B#16#01 (Aktive Seite: 0, passive Seite: 2) • 0 oder 2, falls connection_type = B#16#11 (Aktive Seite: 0 oder 2, passive Seite: 2) • 2 bis 16, falls connection_type = B#16#12
8	rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
9	rem_staddr_len	BYTE	B#16#00	Länge der Adresse des remoten Verbindungsendpunkts: • 0: unspezifiziert, d.h. Parameter rem_staddr ist irrelevant. • 4: gültige IP-Adresse im Parameter rem_staddr

Byte	Parameter	Datentyp	Startwert	Beschreibung
10	rem_tsap_id_len	BYTE	B#16#00	verwendete Länge des Parameters rem_tsap_id; mögliche Werte: • 0 oder 2, falls connection_type = B#16#01 Für die passive Seite ist nur der Wert B#16#00 zulässig. • 0 oder 2, falls connection_type = B#16#11 • 0 oder 2 bis 16, falls connection_type = B#16#12 (Aktive Seite: 2 bis 16; passive Seite: 0 oder 2 bis 16, 0 bedeutet unspezifiziert)
11	next_staddr_len	BYTE	B#16#00	verwendete Länge des Parameters next_staddr (nur relevant bei ISO on TCP)
12 bis 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#07 B#16#D0 B#16#00 ...	bei connection_type = • B#16#11: lokale Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)), local_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = B#16#00 • B#16#12: lokale TSAP-ID: Siehe unten. • B#16#01: lokale Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)), local_tsap_id[1] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = B#16#00 Hinweis: Wenn mehrere Verbindungen zu ein und demselben Kommunikationspartner bestehen, müssen Sie sicherstellen, dass jeder verwendete Wert von local_tsap_id auf Ihrer CPU eindeutig ist.
28 bis 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit 0 belegen.

Byte	Parameter	Datentyp	Startwert	Beschreibung
34 bis 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	IP-Adresse des remoten Verbindungsendpunkts, z. B. 192.168.0.1: bei connection_type = • B#16#1x: rem_staddr[1] = B#16#C0 (192), rem_staddr[2] = B#16#A8 (168), rem_staddr[3] = B#16#00 (0), rem_staddr[4] = B#16#01 (1), rem_staddr[5-6]= B#16#00 (reserviert) • B#16#01: rem_staddr[1] = B#16#01 (1), rem_staddr[2] = B#16#00 (0), rem_staddr[3] = B#16#A8 (168), rem_staddr[4] = B#16#C0 (192), rem_staddr[5-6]= B#16#00 (reserviert)
40 bis 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	bei connection_type = • B#16#11: remote Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)), rem_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, rem_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, rem_tsap_id[3-16] = B#16#00 • B#16#12: remote TSAP-ID: Siehe unten. • B#16#01: remote Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)), local_tsap_id[1] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = B#16#00
56 bis 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	bei local_device_id = • B#16#00: next_staddr[1]: Rack und Steckplatz des zugehörigen (lokalen) CP (Bits 0 bis 4: Steckplatz, Bits 5 bis 7: Racknummer) next_staddr[2-6]: B#16#00 • B#16#02, B#16#03, B#16#05: next_staddr[1-6]: B#16#00 • B#16#01, B#16#06, B#16#0B, B#16#0F: next_staddr[1-6]: B#16#00 (in diesem Fall ist next_staddr_len=B#16#00)
62 bis 63	spare	WORD	W#16#0000	Reserve. Belegen Sie diesen Parameter mit "0".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Länge und Aufbau der lokalen und der remoten TSAP-ID bei ISO on TCP

- Aktiver Verbindungsaufbau:
  - Remote TSAP-ID:
    Jede Zeichenfolge mit einer Länge zwischen 1 und 16 Byte ist erlaubt. Dabei kann jedes Zeichen einen Wert zwischen B#16#00 bis B#16#FF haben.
  - Lokale TSAP-ID:
    Bei integrierter IE-Schnittstelle ist jede Zeichenfolge mit einer Länge zwischen 1 und 16 Byte erlaubt. Dabei kann jedes Zeichen einen Wert zwischen B#16#00 bis B#16#FF haben.
- Passiver Verbindungsaufbau:
  - Remote TSAP-ID:
    Jede Zeichenfolge mit einer Länge zwischen 0 und 16 Byte ist erlaubt. Dabei kann jedes Zeichen einen Wert zwischen B#16#00 bis B#16#FF haben.
  - Lokale TSAP-ID:

loc_tsap_id_len	local_tsap_id[1]	local_tsap_id[2]	local_tsap_id[3 bis 16]
2	B#16#E0 (Verbindungstyp T-Verbindung)	0 (nur bei integrierter IE-Schnittstelle) oder Rack und Steckplatz der eigenen CPU (Bits 0 bis 4 Steckplatz, Bits 5 bis 7 Racknummer)	gibt es nicht
> 2	B#16#E0 (Verbindungstyp T-Verbindung)	0 (nur bei integrierter IE-Schnittstelle) oder Rack und Steckplatz der eigenen CPU (Bits 0 bis 4 Steckplatz, Bits 5 bis 7 Racknummer)	TSAP-Erweiterung
	nur bei integrierter IE-Schnittstelle: ein ASCII-Zeichen (B#16#20 bis B#16#7E)	irrelevant	TSAP-Erweiterung

CPU-Abhängigkeit der Protokollvarianten TCP und ISO on TCP

Welche der beiden Protokollvarianten TCP und ISO on TCP Sie auf welcher CPU einsetzen können, finden Sie in dem folgenden Abschnitt:

Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge (Seite 6453).

Die Anzahl der möglichen Verbindungen entnehmen Sie den Technischen Daten Ihrer CPU.

Verbindungsaufbau

Ein Kommunikationspartner A muss den aktiven Verbindungsaufbau anstoßen. Ein Kommunikationspartner B muss den passiven Verbindungsaufbau anstoßen. Wenn beide Kommunikationspartner ihren Verbindungsaufbau angestoßen haben, kann das Betriebssystem die Kommunikationsverbindung vollständig aufbauen.

In der Verbindungsparametrierung legen Sie fest, welcher Kommunikationspartner den Verbindungsaufbau aktiviert und welcher auf eine Anforderung des Kommunikationspartners hin einen passiven Verbindungsaufbau durchführt.

Bei UDP müssen beide Verbindungspartner den passiven Verbindungsaufbau anstoßen.

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Datenstruktur für Parametrierung des lokalen Kommunikationszugangspunkts

Um den lokalen Kommunikationszugangspunkt zu parametrieren, legen Sie einen DB an, der die Datenstruktur aus dem UDT 65 "TCON_PAR" enthält. Diese Datenstruktur enthält die notwendigen Parameter, die Sie zum Einrichten der Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems benötigen.

Der Parameter CONNECT der Anweisung "TCON (Seite 6463)" enthält einen Verweis auf die Adresse der zugehörigen Verbindungsbeschreibung (z. B. P#DB100.DBX0.0 Byte 64).

Aufbau der Verbindungsbeschreibung bei UDP (UDT 65)

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 bis 1	block_length	WORD	W#16#40	Länge des UDT 65: 64 Bytes (fest)
2 bis 3	id	WORD	W#16#0001	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems (Wertebereich: W#16#0001 bis W#16#0FFF) Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei ID angeben.
4	connection_type	BYTE	B#16#13	Protokollvariante: • B#16#13: UDP
5	active_est	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus: Diesen Parameter müssen Sie mit FALSE belegen.

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
6	local_device_id	BYTE	B#16#02	• B#16#01: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 1 (IF1) bei WinAC RTX • B#16#02: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 317-2 PN/DP • B#16#03: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 319-3 PN/DP • B#16#05: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 414-3 PN/DP, 416-3 PN/DP, 416-3F PN/DP und 41x-5H PN/DP (Rack 0) • B#16#06: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 2 (IF2) bei WinAC RTX • B#16#0B: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 3 (IF3) bei WinAC RTX • B#16#0F: Kommunikation über die IE-Schnittstelle auf Interface-Steckplatz 4 (IF4) bei WinAC RTX • B#16#15: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 41x-5H PN/DP (Rack 1)
7	local_tsap_id_len	BYTE	B#16#02	verwendete Länge des lokalen Parameters local_tsap_id (lokaler Port); 2 Byte
8	rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
9	rem_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
10	rem_tsap_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
11	next_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
12 bis 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#07 B#16#D0 B#16#00 ...	lokale Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)), local_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = B#16#00 (reserviert) Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id, den Sie auf Ihrer CPU verwenden, eindeutig ist.
28 bis 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.
34 bis 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.
40 bis 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
56 bis 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.
62 bis 63	spare	WORD	W#16#0000	Reserve. Belegen Sie diesen Parameter mit "0".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

CPU-Abhängigkeit der Protokollvariante UDP

Auf welcher CPU Sie die Protokollvariante UDP einsetzen können, finden Sie in folgendem Abschnitt:

Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge (Seite 6453)

Die Anzahl der möglichen Verbindungen zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems entnehmen Sie den Technischen Daten Ihrer CPU.

Einrichten des lokalen Kommunikationszugangspunkts

Jeder Kommunikationspartner muss unabhängig vom anderen seinen lokalen Kommunikationszugangspunkt einrichten, d. h. die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems aufbauen.

Bei UDP müssen beide Verbindungspartner den passiven Verbindungsaufbau anstoßen.

Struktur der Adressinformation des remoten Partners bei UDP

Übersicht

- Bei "TUSEND (Seite 6475)" übergeben Sie am Parameter ADDR die Adressinformation des Empfängers. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.
- Bei "TURCV (Seite 6471)" erhalten Sie am Parameter ADDR die Adresse des Absenders der empfangenen Daten. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.

Datenbaustein für die Adressinformation des remoten Partners

Sie müssen einen DB anlegen, der eine oder mehrere Datenstrukturen gemäß UDT 66 "TADDR_PAR" enthält.

Im Parameter ADDR von "TUSEND (Seite 6475)" übergeben Sie und am Parameter ADDR von "TURCV (Seite 6471)" erhalten Sie einen Zeiger auf die Adresse des zugehörigen remoten Partners (z. B. P#DB100.DBX0.0 Byte 8).

Aufbau der Adressinformation des remoten Partners (UDT 66)

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 bis 3	rem_ip_addr	ARRAY [1..4] of BYTE	B#16#00 ...	IP-Adresse des remoten Partners, z. B. 192.168.2.3: • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#02 (2) • rem_ip_addr[4] = B#16#03 (3)
4 bis 5	rem_port_nr	ARRAY [1..2] of BYTE	B#16#00 ...	remote Port-Nr. (mögliche Werte siehe: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)) • rem_port_nr[1] = high byte der Port-Nr. in hexadezimaler Darstellung • rem_port_nr[2] = low byte der Port-Nr. in hexadezimaler Darstellung
6 bis 7	spare	ARRAY [1..2] of BYTE	B#16#00 ...	Reserve. Belegen Sie diesen Parameter mit "0".

Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge

Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge

In der folgenden Tabelle wird beschrieben, welche Protokollvariante Sie auf welcher CPU einsetzen und welche Datenlängen Sie übertragen können.

Protokollvariante	Parameter "connection _type" in UDT 65	CPU	Datenlänge in Bytes bei Kommunikation über integrierte IE-Schnittstelle	Datenlänge in Bytes bei Kommunikation über CP
TCP	B#16#11	CPUs 31x-2 PN/DP ab Firmwarestand V2.4	1 bis 8192	-
		CPUs 31x-2 PN/DP ab Firmwarestand V3.1	1 bis 32767	
		CPUs 31x-3 PN/DP ab Firmwarestand V3.2	1 bis 32767	
		CPUs 41x-3 PN/DP	1 bis 32767	-
		CPUs 41x-5H PN/DP	1 bis 32767	-
	B#16#01	CPUs 31x-2 PN/DP	1 bis 1472	-

Protokollvariante	Parameter "connection _type" in UDT 65	CPU	Datenlänge in Bytes bei Kommunikation über integrierte IE-Schnittstelle	Datenlänge in Bytes bei Kommunikation über CP
ISO on TCP	B#16#12	CPUs 31x-2 PN/DP ab Firmwarestand V2.4	1 bis 8192	-
		CPUs 31x-2 PN/DP ab Firmwarestand V3.1	1 bis 32767	
		CPUs 31x-3 PN/DP ab Firmwarestand V3.2	1 bis 32767	
		CPUs 41x ab Firmwarestand V4.1 (ohne CPUs 41x-4H)	-	1 bis 1452
		CPUs 41x-3 PN/DP	1 bis 32767	1 bis 1452
		CPUs 41x-5H PN/DP	1 bis 32767	1 bis 1452
UDP	B#16#13	CPUs 31x-2 PN/DP ab Firmwarestand V2.4, CPUs 41x-3 PN/DP und 41x-5H PN/DP	1 bis 1472	-

Beispiele zur Parametrierung der Kommunikationsverbindungen

Beispiel 1: Kommunikation über ISO on TCP und CP 443-1 zwischen zwei S7-400-CPUs

Die beiden Kommunikationspartner sind zwei CPUs 414-2. Die Kommunikation wird über zwei CPs 443-1 abgewickelt.

Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 414-2 mit CP 443-1	Kommunikationspartner B: CPU 414-2 mit CP 443-1
Verbindungsaufbau	aktiv	passiv
IP-Adresse	192.168.4.14	192.168.4.16
Physikalische Adresse der CPU	Rack 0, Steckplatz 3	Rack 0, Steckplatz 4
Physikalische Adresse des zugehörigen CP	Rack 0, Steckplatz 6	Rack 1, Steckplatz 8
Lokale TSAP-ID (Hinweis: Die Codierung des eigentlichen TSAP zur Unterscheidung der Verbindung erfolgt ab dem 3. Byte.)	0xE0 03 54 43 50 2D 31	0xE0 04 54 43 50 2D 31

Die für den aktiven Verbindungsaufbau von Kommunikationspartner A relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#0414	Referenz auf diese Verbindung
connection_type	BYTE	B#16#12	Protokollvariante: ISO on TCP
active_est	BOOL	TRUE	Aktiver Verbindungsaufbau
local_device_id	BYTE	B#16#00	Kommunikation AS-intern über CP
local_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters local_tsap_id

Parameter	Datentyp	Wert im Beispiel	Beschreibung
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#01	verwendete Länge des Parameters next_staddr
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • local_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • local_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • local_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • local_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • local_tsap_id[8-16] = B#16#00	lokale TSAP-ID: 0xE0035443502D31
rem_staddr	ARRAY [1..6] of BYTE	"192.168.4.16" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#10 (16) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#04 • rem_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • rem_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • rem_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • rem_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • rem_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • rem_tsap_id[8-16] = B#16#00	remote TSAP-ID: 0xE0045443502D31
next_staddr	ARRAY [1..6] of BYTE	• next_staddr[1] = B#16#06 • next_staddr[2-6] = B#16#00	Rack = 0, Steckplatz = 6 (Bits 7 bis 5: Rack-Nr., Bits 4 bis 0: Steckplatz-Nr.)

Die für den passiven Verbindungsaufbau von Kommunikationspartner B relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#0416	Referenz auf diese Verbindung
connection_type	BYTE	B#16#12	Protokollvariante: ISO on TCP
active_est	BOOL	FALSE	Passiver Verbindungsaufbau
local_device_id	BYTE	B#16#00	Kommunikation AS-intern über CP
local_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters local_tsap_id
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#01	verwendete Länge des Parameters next_staddr
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#04 • local_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • local_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • local_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • local_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • local_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • local_tsap_id[8-16] = B#16#00	lokale TSAP-ID: 0xE0045443502D31
rem_staddr	ARRAY [1..6] of BYTE	"192.168.4.14" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#0E (14) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts

Parameter	Datentyp	Wert im Beispiel	Beschreibung
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • rem_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • rem_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • rem_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • rem_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • rem_tsap_id[8-16] = B#16#00	remote TSAP-ID: 0xE0035443502D31
next_staddr	ARRAY [1..6] of BYTE	• next_staddr[1] = B#16#28 • next_staddr[2-6] = B#16#00	Rack = 1, Steckplatz = 8 (Bits 7 bis 5: Rack-Nr., Bits 4 bis 0: Steckplatz-Nr.)

Beispiel 2: Kommunikation über TCP und integrierte PROFINET-Schnittstellen zwischen einer S7-400-CPU und einer S7-300-CPU

Der eine Kommunikationspartner ist eine CPU 416-3 PN/DP, der andere eine CPU 319-3 PN/DP. Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 416-3 PN/DP	Kommunikationspartner B: CPU 319-3 PN/DP
Verbindungsaufbau	aktiv	passiv
IP-Adresse	192.168.3.142	192.168.3.125
Lokale Port-Nr.	irrelevant	2005

Die für den aktiven Verbindungsaufbau von Kommunikationspartner A relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#0014	Referenz auf diese Verbindung
connection_type	BYTE	B#16#11	Protokollvariante: TCP
active_est	BOOL	TRUE	Aktiver Verbindungsaufbau
local_device_id	BYTE	B#16#05	Kommunikation über die integrierte Ethernet-Schnittstelle bei S7-400-CPUs
local_tsap_id_len	BYTE	B#16#00 (nur dieser Wert ist möglich)	Parameter local_tsap_id wird nicht verwendet
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr

Parameter	Datentyp	Wert im Beispiel	Beschreibung
rem_tsap_id_len	BYTE	B#16#02 (nur dieser Wert ist möglich)	verwendete Länge des Parameters rem_tsap_id
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.125" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#7D (125) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	"2005" • rem_tsap_id[1] = B#16#07 • rem_tsap_id[2] = B#16#D5 • rem_tsap_id[3-16] = B#16#00	Remote Port-Nr.: 2005 = W#16#07D5

Die für den passiven Verbindungsaufbau von Kommunikationspartner B relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#000F	Referenz auf diese Verbindung
connection_type	BYTE	B#16#11	Protokollvariante: TCP
active_est	BOOL	FALSE	Passiver Verbindungsaufbau
local_device_id	BYTE	B#16#03	Kommunikation über die integrierte Ethernet-Schnittstelle bei der CPU 319-3 PN/DP
local_tsap_id_len	BYTE	B#16#02 (nur dieser Wert ist möglich)	verwendete Länge des Parameters local_tsap_id
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#00 (nur dieser Wert ist möglich)	verwendete Länge des Parameters rem_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	"2005" • local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = B#16#00	Lokale Port-Nr.: 2005 = W#16#07D5
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.142" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#8E (142) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts

Beispiel 3: Kommunikation über UDP und integrierte PROFINET-Schnittstellen zwischen einer S7-300-CPU und einer S7-400-CPU

Der eine Kommunikationspartner ist eine CPU 319-3 PN/DP, der andere eine CPU 414-3 PN/DP. Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 319-3 PN/DP	Kommunikationspartner B: CPU 414-3 PN/DP
Sender/Empfänger	Sender	Empfänger
IP-Adresse	192.168.3.142	192.168.3.125
Lokale Port-Nr.	2004	2005

Die für den Sender (Kommunikationspartner A) relevanten Einträge im DB für die Parametrierung des lokalen Kommunikationszugangspunkts zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#0014	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems
connection_type	BYTE	B#16#13	Protokollvariante: UDP
active_est	BOOL	FALSE	Nur dieser Wert ist möglich bei der Protokollvariante UDP.
local_device_id	BYTE	B#16#03	Kommunikation über die integrierte Ethernet-Schnittstelle bei der CPU 319-3 PN/DP
local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D4 • local_tsap_id[3-16] = B#16#00	Lokale Port-Nr.: 2004 = W#16#07D4

Die für den Empfänger (Kommunikationspartner B) relevanten Einträge im DB für die Parametrierung des lokalen Kommunikationszugangspunkts zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#000F	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems
connection_type	BYTE	B#16#13	Protokollvariante: UDP
active_est	BOOL	FALSE	Nur dieser Wert ist möglich bei der Protokollvariante UDP.
local_device_id	BYTE	B#16#05	Kommunikation über die integrierte Ethernet-Schnittstelle bei S7-400-CPUs
local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = B#16#00	Lokale Port-Nr.: 2005 = W#16#07D5

Beim Aufruf von "TUSEND (Seite 6475)" auf dem Sender übergeben Sie in einem DB die folgenden Adressparameter des Empfängers:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
rem_ip_addr	ARRAY [1..4] of BYTE	• rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#7D (125)	IP-Adresse des Empfängers: 192.168.3.125
rem_port_nr	ARRAY [1..2] of BYTE	• rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D5	Port-Nr. des Empfängers: 2005 = W#16#07D5

Beim Aufruf von "TURCV (Seite 6478)" auf dem Empfänger erhalten Sie in einem DB die folgenden Adressparameter des Senders:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
rem_ip_addr	ARRAY [1..4] of BYTE	• rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#8E (142)	IP-Adresse des Senders: 192.168.3.142
rem_port_nr	ARRAY [1..2] of BYTE	• rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D4	Port-Nr. des Senders: 2004 = W#16#07D4

Beispiel 4: Kommunikation über ISO on TCP und integrierte PROFINET-Schnittstellen zwischen zwei S7-400-CPUs

Der eine Kommunikationspartner ist eine CPU 414-3 PN/DP ab Firmwarestand V5.0, der andere eine CPU 416-3 PN/DP mit Firmwarestand V5.0. Die Kommunikation wird über die integrierten PROFINET-Schnittstellen abgewickelt.

Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner.

Eigenschaft	Kommunikationspartner A: CPU 414-3 PN/DP	Kommunikationspartner B: CPU 416-3 PN/DP
Sender/Empfänger	Sender	Empfänger
Verbindungsaufbau	passiv	aktiv
IP-Adresse	192.168.0.14	192.168.0.16

Die folgende Tabelle zeigt alle Parametereinträge von Kommunikationspartner A im zugehörigen DB.

Parameter	Datentyp	Wert im Beispiel	Beschreibung
block_length	WORD	W#16#40	Länge des UDT65: 64 Bytes (fest)
id	WORD	W#16#0001	Referenz auf diese Verbindung

Parameter	Datentyp	Wert im Beispiel	Beschreibung
connection_type	BYTE	B#16#12	Protokollvariante: • B#16#12: ISO on TCP
active_est	BOOL	FALSE	Passiver Verbindungsaufbau
local_device_id	BYTE	B#16#05	Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 41x PN/DP
local_tsap_id_len	BYTE	B#16#03	Verwendete Länge des Parameters local_tsap_id
rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#03	Verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet (da Kommunikation über die integrierte Schnittstelle). Sie müssen ihn mit B#16#00 belegen.
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#01 • local_tsap_id[4-16] = B#16#00	lokale TSAP-ID: 0xE00301
rem_subnet_id	ARRAY [1..6] of BYTE	rem_subnet_id[1-6] = B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.
rem_staddr	ARRAY [1..6] of BYTE	"192.168.0.16" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#00 (0) • rem_staddr[4] = B#16#10 (16) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#01 • rem_tsap_id[4-16] = B#16#00	remote TSAP-ID: 0xE00301
next_staddr	ARRAY [1..6] of BYTE	next_staddr[1-6] = B#16#00	Dieser Parameter wird nicht verwendet (da Kommunikation über die integrierte Schnittstelle)
spare	WORD	W#16#0000	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.

Die folgende Tabelle zeigt alle Parametereinträge von Kommunikationspartner B im zugehörigen DB.

Parameter	Datentyp	Wert im Beispiel	Beschreibung
block_length	WORD	W#16#40	Länge des UDT65: 64 Bytes (fest)
id	WORD	W#16#0001	Referenz auf diese Verbindung
connection_type	BYTE	B#16#12	Protokollvariante: • B#16#12: ISO on TCP
active_est	BOOL	TRUE	Aktiver Verbindungsaufbau
local_device_id	BYTE	B#16#05	Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 41x PN/DP
local_tsap_id_len	BYTE	B#16#03	Verwendete Länge des Parameters local_tsap_id
rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#03	Verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet (da Kommunikation über die integrierte Schnittstelle). Sie müssen ihn mit B#16#00 belegen.
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#01 • local_tsap_id[4-16] = B#16#00	lokale TSAP-ID: 0xE00301
rem_subnet_id	ARRAY [1..6] of BYTE	rem_subnet_id[1-6] = B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.
rem_staddr	ARRAY [1..6] of BYTE	"192.168.0.14" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#00 (0) • rem_staddr[4] = B#16#0E (14) • rem_staddr[5-6] = B#16#00	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#01 • rem_tsap_id[4-16] = B#16#00	remote TSAP-ID: 0xE00301
next_staddr	ARRAY [1..6] of BYTE	next_staddr[1-6] = B#16#00	Dieser Parameter wird nicht verwendet (da Kommunikation über die integrierte Schnittstelle)
spare	WORD	W#16#0000	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit "0" belegen.

Datentypen

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

TCON: Kommunikationsverbindung aufbauen

Anzahl der möglichen Verbindungen

Die Anzahl der möglichen TCP- bzw. UDP-Verbindungen entnehmen Sie den Technischen Daten Ihrer CPU.

Verwendung bei TCP und ISO on TCP

Beide Kommunikationspartner rufen die Anweisung "TCON" zum Einrichten und Aufbauen der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationssendepunkt ist.

Nach dem Einrichten und Aufbauen der Verbindung wird diese automatisch von der CPU gehalten und überwacht.

Bei Verbindungsabbruch durch z. B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner die eingerichtete Verbindung wieder aufzubauen. Sie müssen "TCON" nicht erneut aufrufen.

Mit dem Aufruf der Anweisung "TDISCON" oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung müssen Sie "TCON" nochmals aufrufen.

Verwendung bei UDP

Beide Kommunikationspartner rufen die Anweisung "TCON" auf, um ihren lokalen Kommunikationszugangspunkt (lokalen Port) einzurichten. Dabei wird eine Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems eingerichtet. Es erfolgt kein Verbindungsaufbau zum remoten Partner.

Der lokale Zugangspunkt wird zum Senden und Empfangen von UDP-Telegrammen verwendet.

Bei der Parametrierung des lokalen Zugangspunktes (UDT65) bzw der Parametrierung des remoten Partners (UDT 66) müssen folgende Parameter gleich sein.

- bei UDP: Referenz auf die Verbindung und lokale Port-Nr.

Arbeitsweise

Die Anweisung "TCON" arbeitet asynchron, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag zum Einrichten und Aufbauen der Verbindung, indem Sie "TCON" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TCON" aktuell befindet bzw. wann die Verbindung eingerichtet bzw. aufgebaut (bei TCP und ISO on TCP) ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Dem Anweisung wurde kein (neuer) Auftrag erteilt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TCON":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter REQUEST, startet den Auftrag zum Aufbau der durch ID gegebenen Verbindung. Der Auftragsstart erfolgt bei steigender Flanke.
ID	Input	WORD	M, D oder Konstante	Referenz auf die aufzubauende Verbindung zum remoten Partner bzw. zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. • BUSY = 0: Der Auftrag ist beendet.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Auftragsbearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
CONNECT	InOut	ANY	D	Zeiger auf die zugehörige Verbindungsbeschreibung (UDT 65) Siehe auch: • Parametrierung der Kommunikationsverbindungen bei TCP und ISO on TCP (Seite 6445) • Parametrierung des lokalen Kommunikationszugangspunkts bei UDP (Seite 6450) Hinweis: Den Parameter CONNECT können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: CONN_DB.Conn1

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Verbindung konnte aufgebaut werden
0	7000	keine Auftragsbearbeitung aktiv
0	7001	Start der Auftragsbearbeitung, Verbindung wird aufgebaut
0	7002	Zwischenaufruf (REQ irrelevant), Verbindung wird aufgebaut
1	8086	Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.
1	8087	Maximale Anzahl der Verbindungen ist erreicht, keine weitere Verbindung möglich.
1	8089	Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.
1	809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht die Länge der Verbindungsbeschreibung (UDT65) hat.
1	809B	Die in der Verbindungsbeschreibung angegebene local_device_id passt nicht zur CPU.
1	80A0	Sammelfehler für die Fehlercodes W#16#80A1 und W#16#80A2
1	80A1	Verbindung bzw. Port ist bereits vom Anwender belegt
1	80A2	Lokaler oder remoter Port ist vom System belegt
1	80A3	Es wird versucht, eine bestehende Verbindung erneut aufzubauen.
1	80A4	IP-Adresse des remoten Verbindungsendpunkts ist ungültig, evtl. stimmt sie mit der eigenen IP-Adresse überein
1	80A7	Kommunikationsfehler: Sie haben nach einem nicht abgeschlossenen "TCON" einen "TDISCON (Seite 6466)" aufgerufen. Der Verbindungsaufbau wurde durch Aufruf eines "TDISCON (Seite 6466)" vorzeitig abgebrochen.
1	80B2	Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Schlüsselwort UNLINKED erzeugt wurde.

ERROR	STATUS (W#16#...)	Erläuterung
1	80B3	Inkonsistente Parametrierung: Sammelfehler für die Fehlercodes W#16#80A0 bis W#16#80A2, W#16#80A4, W#16#80B4 bis W#16#80B9
1	80B4	Sie haben bei der Protokollvariante ISO on TCP (connection_type = B#16#12) beim passiven Verbindungsaufbau (active_est = FALSE) eine oder mehrere der folgenden Bedingungen verletzt: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0 • Bei local_tsap_id_len >= B#16#03 ist local_tsap_id[1] ein ASCII-Zeichen • local_tsap_id[1] ist ein ASCII-Zeichen und local_tsap_id_len >= B#16#03
1	80B5	Fehler beim Parameter active_est (UDT 65) bei der Protokollvariante UDP
1	80B6	Parametrierfehler bzgl. des Parameters connection_type (UDT 65)
1	80B7	Fehler in einer der folgenden Parameter des UDT 65: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len
1	80B8	Parameter in der lokalen Verbindungsbeschreibung (UDT 65) und Parameter ID sind unterschiedlich
1	80C3	Alle Verbindungsressourcen sind belegt.
1	80C4	Temporärer Kommunikationsfehler: • Verbindung kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert • Die eingerichtete Verbindung wird momentan durch einen "TDISCON (Seite 6466)" entfernt. • Das H-System befindet sich im Ankoppeln und Aufdaten.
1	8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Parametrierung der Kommunikationsverbindungen bei TCP und ISO on TCP (Seite 6445)

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP (Seite 6450)

TDISCON: Kommunikationsverbindung abbauen

Verwendung bei TCP und ISO on TCP

Die Anweisung "TDISCON" baut eine Kommunikationsverbindung der CPU zu einem Kommunikationspartner ab.

Verwendung bei UDP

Die Anweisung "TDISCON" löst den lokalen Kommunikationszugangspunkt auf, d. h. die Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird abgebaut.

Arbeitsweise

Die Anweisung "TDISCON" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag zum Verbindungsabbau, indem Sie die Anweisung "TDISCON" mit REQ = 1 aufrufen.

Nach dem erfolgreichen Durchlauf der Anweisung "TDISCON" ist die bei der Anweisung "TCON (Seite 6463)" angegebene ID nicht mehr gültig und kann damit weder zum Senden noch zum Empfangen verwendet werden.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen (siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TDISCON":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z	Startet den Auftrag zum Abbauen der in der ID angegebenen Verbindung bei einer steigenden Flanke.
ID	Input	WORD	M, D oder Konstante	Referenz auf die mit "TCON (Seite 6463)" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF
DONE	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder noch in Bearbeitung • 1: Auftrag fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	Statusparameter mit folgenden Werten: • 0: Auftrag noch nicht gestartet oder bereits beendet • 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.
ERROR	Output	BOOL	E, A, M, D, L	Statusparameter ERROR: • 0: Kein Fehler • 1: Fehler aufgetreten
STATUS	Output	WORD	M, D	Status der Anweisung

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von TDISCON auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Verbindung erfolgreich beendet
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Beginn der Auftragsbearbeitung, Verbindung wird beendet.
0	7002	Verbindung wird beendet (REQ irrelevant).
1	8086	Der Parameter ID liegt außerhalb des zulässigen Adressbereichs.
1	80A3	Es wird versucht, eine nicht vorhandene Verbindung zu beenden.
1	80C4	Temporärer Kommunikationsfehler: Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut bzw. das H-System befindet sich im Ankoppeln und Aufdaten.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

TSEND: Daten über Kommunikationsverbindung senden

Beschreibung

Die Anweisung "TSEND" sendet Daten über eine bestehende Kommunikationsverbindung.

Arbeitsweise

"TSEND" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Sendevorgang, indem Sie "TSEND" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Der Ausgangsparameter DONE wird gesetzt, wenn die Daten vollständig zum Versenden an die lokale Schnittstelle übertragen wurden.

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TSEND" aktuell befindet.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TSEND" müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter REQUEST, startet den Sendeauftrag bei steigender Flanke. Die Daten werden aus dem mit DATA und LEN angegebenen Bereich übergeben.
ID	Input	WORD	M, D oder Konstante	Referenz auf die zugehörige Verbindung. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	INT	E, A, M, D, L	Maximale Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen Siehe auch: Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge (Seite 6453)
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Die Daten wurden vollständig zum Versenden an die lokale Schnittstelle übertragen.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
DATA	InOut	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Hinweis: Verwenden Sie als Sendebereich nicht ARRAY of BOOL. Hinweis: Den Parameter DATA können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Sendeauftrag wurde ohne Fehler abgeschlossen
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Start der Auftragsbearbeitung, Daten werden versendet Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
0	7002	Zwischenaufruf (REQ irrelevant), Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
1	8085	Parameter LEN hat den Wert "0" oder ist größer als der größte zulässige Wert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	Parameter LEN ist größer als der in DATA angegebene Speicherbereich
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich. • Schnittstelle wird neu initialisiert

ERROR	STATUS (W#16#...)	Erläuterung
1	80B3	Der parametrierte Verbindungstyp (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie "TUSEND (Seite 6475)".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Kommunikationspartner kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert bzw. die Verbindung wird gerade aufgebaut.
1	8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

TRCV: Daten über Kommunikationsverbindung empfangen

Beschreibung

Die Anweisung "TRCV" empfängt Daten über eine bestehende Kommunikationsverbindung.

Empfangsbereich

Man versteht darunter den Bereich, in den der Funktionsbaustein die empfangenen Daten einträgt.

Der Empfangsbereich ist durch die folgenden beiden Größen festgelegt:

- Zeiger auf den Beginn des Bereichs
- Länge des Bereichs

Die Länge des Bereichs wird abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) festgelegt.

Empfangsmodi von TRCV

In der folgenden Tabelle wird beschrieben, wie "TRCV" die empfangenen Daten in den Empfangsbereich einträgt.

Protokollvariante	Eintragen der Daten in den Empfangsbereich	Parameter connection_type	Wert des Parameters LEN
TCP	Ad-hoc-Mode	B#16#01, B#16#11	0
TCP	Datenempfang mit vorgegebener Länge	B#16#01, B#16#11	<> 0
ISO on TCP	protokollgesteuert	B#16#12	0 (empfohlen) oder <> 0

TCP / Ad-hoc-Mode

Den Ad-hoc-Mode gibt es nur bei der Protokollvariante TCP. Sie stellen den Ad-hoc-Mode ein, indem Sie dem Parameter LEN den Wert "0" zuweisen.

Der Empfangsbereich ist identisch mit dem durch DATA gebildeten Bereich. Es werden maximal 8192 Byte empfangen. Unmittelbar nach Empfang eines Datenblocks überträgt die Anweisung "TRCV" diesen in den Empfangsbereich und setzt NDR auf "1".

TCP / Datenempfang mit vorgegebener Länge

Indem Sie dem Parameter LEN einen von Null verschiedenen Wert zuweisen, stellen Sie den Datenempfang mit vorgegebener Länge ein.

Der Empfangsbereich wird durch die Parameter LEN und DATA gebildet. Sobald LEN Bytes empfangen wurden, überträgt TRCV diese in den Empfangsbereich und setzt NDR auf "1".

ISO on TCP / Protokollgesteuerte Datenübertragung

Bei der Protokollvariante ISO on TCP werden die Daten protokollgesteuert übertragen.

Der Empfangsbereich wird durch die Parameter LEN und DATA gebildet. Sobald sämtliche Daten eines Auftrags empfangen wurden, überträgt "TRCV" diese in den Empfangsbereich und setzt NDR auf "1".

Austausch von Daten des Datentyps STRING

Wenn Sie mit den Anweisungen "TSEND" und "TRCV" Daten des Datentyps STRING austauschen wollen, ist nur der Ad-hoc-Mode mit LEN=0 sinnvoll. Außerdem müssen Sie die Länge des Strings bei der Anweisung "TRCV" mindestens so groß wählen wie bei der Anweisung "TSEND". Wenn Sie gegen diese Regel verstoßen, kann dies zu Fehlern bei der weiteren Stringbearbeitung führen.

Wenn Sie als Empfangsbereich für eine gesendete Zeichenkette ein ARRAY of BYTE verwenden, enthält das erste Byte die vom Sender vorgegebene Maximallänge und das zweite Byte die aktuelle Länge der Zeichenkette.

Arbeitsweise

"TRCV" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Empfangsauftrag, indem Sie TRCV mit EN_R = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, NDR und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TRCV" aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

BUSY	NDR	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.

BUSY	NDR	ERROR	Beschreibung
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TRCV" sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter NDR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter enabled to receive: Mit EN_R = 1 wird "TRCV" empfangsbereit. Der Empfangsauftrag wird bearbeitet.
ID	Input	WORD	M, D oder Konstante	Referenz auf die zugehörige Verbindung. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	INT	E, A, M, D, L	Länge des Empfangsbereichs in Bytes Zur Bedeutung von LEN = 0 bzw. LEN <> 0 siehe oben (Empfangsmodi von "TRCV"). Zum Wertebereich siehe auch: Zusammenhang zwischen eingesetzter CPU und Protokollvariante (connection_type) und übertragbarer Datenlänge (Seite 6453).
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch • NDR = 1: Auftrag wurde erfolgreich abgeschlossen
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.
STATUS	Output	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
RCVD_LEN	Output	INT	E, A, M, D, L	Anzahl der tatsächlich empfangenen Daten in Bytes
DATA	InOut	ANY	E, A, M, D	Empfangsbereich (Definition siehe oben), enthält Adresse und Länge Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Hinweis: Verwenden Sie als Empfangsbereich nicht ARRAY of BOOL. Hinweis: Den Parameter DATA können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.
0	7000	Baustein nicht empfangsbereit
0	7001	Baustein ist empfangsbereit, Empfangsauftrag wurde aktiviert
0	7002	Zwischenaufruf, Empfangsauftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase schreibt die Anweisung Daten in den Empfangsbereich. Deshalb können im Fehlerfall inkonsistente Daten im Empfangsbereich stehen.
1	8085	Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben einen der Parameter LEN oder DATA gegenüber dem Erstaufruf geändert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	• Empfangsbereich ist zu klein • Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung wurde noch nicht aufgebaut. • Die angegebene Verbindung wird momentan abgebaut. Ein Empfangsauftrag über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu parametriert.
1	80B3	Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie "TURCV (Seite 6478)".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel

ERROR	STATUS (W#16#...)	Erläuterung
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zum Kommunikationspartner kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert bzw. die Verbindung wird gerade aufgebaut.
1	8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

TUSEND: Daten über Ethernet (UDP) senden

Beschreibung

Die Anweisung "TUSEND" sendet Daten über UDP an den über den Parameter ADDR adressierten remoten Partner.

⚠ WARNUNG

Datenübertragung über UDP

Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum remoten Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird.

Hinweis

Bei aufeinander folgenden Sendevorgängen an verschiedene Partner müssen Sie bei den Aufrufen von "TUSEND" lediglich den Parameter ADDR anpassen. Der erneute Aufruf der Anweisungen "TCON (Seite 6463)" und "TDISCON (Seite 6466)" hingegen entfällt.

Arbeitsweise

"TUSEND" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Sendevorgang, indem Sie "TUSEND" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TUSEND" aktuell befindet bzw. wann der Sendevorgang beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.

BUSY	DONE	ERROR	Beschreibung
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TUSEND" müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TUSEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter REQUEST, startet den Sendeauftrag bei steigender Flanke. Die Daten werden aus dem mit DATA und LEN angegebenen Bereich übergeben.
ID	Input	WORD	M, D oder Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	INT	E, A, M, D, L	Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen Wertebereich: 1 bis 1472
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DATA	InOut	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Hinweis: Den Parameter DATA können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable
ADDR	InOut	ANY	D	Zeiger auf die Adresse des Empfängers (z. B. P#DB100.DBX0.0 Byte 8) Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 6452) Hinweis: Den Parameter ADDR können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Sendeauftrag wurde ohne Fehler abgeschlossen
0	7000	Keine Auftragsbearbeitung aktiv
0	7001	Start der Auftragsbearbeitung, Daten werden versendet Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
0	7002	Zwischenaufruf (REQ irrelevant), Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.
1	8085	Parameter LEN hat den Wert "0" oder ist größer als der größte zulässige Wert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
0	8088	Parameter LEN ist größer als der in DATA angegebene Speicherbereich
1	8089	Parameter ADDR zeigt nicht auf einen Datenbaustein
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut. • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich. • Schnittstelle wird neu initialisiert

ERROR	STATUS (W#16#...)	Erläuterung
1	80A4	IP-Adresse des remoten Verbindungsendpunkts ist ungültig, evtl. stimmt sie mit der eigenen IP-Adresse überein.
1	80B3	• Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TSEND (Seite 6468)". • Parameter ADDR: ungültige Angaben für Port-Nr.
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel
1	80C4	Temporärer Kommunikationsfehler: • Die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert.
1	8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

TURCV: Daten über Ethernet (UDP) empfangen

Beschreibung

Die Anweisung "TURCV" empfängt Daten über UDP. Nach erfolgreichem Abschluss von "TURCV" wird Ihnen am Parameter ADDR die Adresse des remoten Partners, also des Senders, zur Verfügung gestellt.

⚠ WARNUNG

Datenübertragung über UDP gemäß RFC 768

Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum remoten Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird.

Arbeitsweise

"TURCV" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Empfangsauftrag, indem Sie "TURCV" mit EN_R = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, NDR und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich TURCV aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

BUSY	NDR	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Hinweis

Durch die asynchrone Arbeitsweise von "TURCV" sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter NDR den Wert TRUE annimmt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TURCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter enabled to receive: Mit EN_R = 1 wird "TURCV" empfangsbereit. Der Empfangsauftrag wird bearbeitet.
ID	Input	WORD	M, D oder Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	Input	INT	E, A, M, D, L	Länge des Empfangsbereichs in Bytes: 0 (empfohlen) bzw. 1 bis 1472 Siehe auch: Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)
NDR	Output	BOOL	E, A, M, D, L	Zustandsparameter NDR: • NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch • NDR = 1: Auftrag wurde erfolgreich abgeschlossen
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	Output	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
RCVD_LEN	Output	INT	E, A, M, D, L	Anzahl der tatsächlich empfangenen Daten in Bytes
DATA	InOut	ANY	E, A, M, D	Empfangsbereich (Definition siehe Arbeitsweise der Anweisungen zur Open User Communication über Industrial Ethernet (Seite 6441)) Die Adresse verweist auf: • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein Hinweis: Den Parameter DATA können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable
ADDR	InOut	ANY	D	Zeiger auf die Adresse des Senders (z. B. P#DB100.DBX0.0 Byte 8) Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 6452) Hinweis: Den Parameter ADDR können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.
0	7000	Baustein nicht empfangsbereit
0	7001	Baustein ist empfangsbereit, Empfangsauftrag wurde aktiviert
0	7002	Zwischenaufruf, Empfangsauftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase schreibt "TURCV" Daten in den Empfangsbereich. Deshalb können im Fehlerfall inkonsistente Daten im Empfangsbereich stehen.
1	8085	Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben einen der Parameter LEN oder DATA gegenüber dem Erstaufruf geändert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	• Empfangsbereich ist zu klein • Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich
1	8089	Parameter ADDR zeigt nicht auf einen Datenbaustein

ERROR	STATUS (W#16#...)	Erläuterung
1	80A1	Kommunikationsfehler: • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut. • Die angegebene Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Empfangsauftrag über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu parametriert.
1	80B3	Die parametrierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TRCV (Seite 6471)".
1	80C3	• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • interner Ressourcenmangel
1	80C4	Temporärer Kommunikationsfehler: Die Schnittstelle wird neu parametriert.
1	8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

IP_CONF: IP-Konfigurationsparameter ändern

Beschreibung IP_CONF

Beschreibung

Die Anweisung "IP_CONF" dient zur programmgesteuerten Konfiguration der integrierten PROFINET-Schnittstelle der CPU. Die bisher gültigen Konfigurationsdaten werden dabei überschrieben.

Sie können die folgenden Einstellungen der Schnittstellen-Konfiguration vornehmen:

- IP-Parameter: IP-Adresse, Subnetzmaske, Router-Adresse
- PROFINET IO-Gerätename (falls die CPU als PROFINET IO Device betrieben wird)

Die Konfigurationsdaten müssen Sie in einem Konfigurations-DB ablegen.

Die programmgesteuerte Einstellung der IP-Konfiguration mit der Anweisung "IP_CONF" können Sie alternativ zur Projektierung mit STEP 7 vornehmen. Sie wird jedoch nur dann wirksam, wenn Sie in der Hardware-Konfiguration explizit vorgegeben haben, dass die Vergabe von IP-Parametern über das Anwenderprogramm erfolgt.

Hinweis

Die Konfigurationsdaten eines Kommunikationsprozessors können alternativ zur Projektierung mit der Anweisung "IP_CONFIG" eingestellt werden (siehe SIMATIC NET CP).

Arbeitsweise

Die Anweisung "IP_CONF" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Übertragungsvorgang, indem Sie "IP_CONF" mit REQ = 1 aufrufen.

Zu jedem Zeitpunkt kann nur ein Auftrag aktiv sein.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET_VAL der asynchron arbeitenden Anweisungen (siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung aktuell befindet bzw. wann die Übertragung der Konfigurationsdaten beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen.
FALSE	FALSE	FALSE	Der Anweisung wurde kein (neuer) Auftrag erteilt.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "IP_CONF":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L, T, Z	Steuerparameter REQUEST. Startet die Anweisung bei steigender Flanke.
LADDR	Input	WORD	M, D oder Konstante	Diagnoseadresse der PROFINET-Schnittstelle
CONF_DB (Seite 6484)	Input	ANY	D	Zeiger auf die Konfigurationsdaten (zulässige Datentypen: BYTE, WORD, BLOCK_DB) Hinweis: Den Parameter CONF_DB können Sie auf die folgenden beiden Arten angeben: • absolut. Bsp.: P#DB13.DBX0.0 Byte 64 • symbolisch. Bsp.: DB_name.variable
DONE	Output	BOOL	E, A, M, D, L	Zustandsparameter DONE: • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	Output	BOOL	E, A, M, D, L	• 0: Der Auftrag ist beendet. • 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	Output	BOOL	E, A, M, D, L	Zustandsparameter ERROR: ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	Output	DWORD	E, A, M, D, L	Fehlerinformation
ERR_LOC	Output	DWORD	E, A, M, D, L	Fehlerquelle (field_id und subfield_type_id des Fehler verursachenden Parameterblocks)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter ERROR und STATUS

ERROR	STATUS (DW#16#..)	ERR_LOC*	Erläuterung
0	00000000	0	Auftragsbearbeitung ohne Fehler beendet
0	00700000	0	Keine Auftragsbearbeitung aktiv
0	00700100	0	Start der Auftragsbearbeitung
0	00700200	0	Zwischenaufruf (REQ irrelevant)
1	C08xyy00	0	Allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)
1	C0808000	0	LADDR ist ungültig
1	C0808100	0	LADDR ist der unterstützten PROFINET-Schnittstelle nicht zugewiesen
1	C0808200	0	Fehler im Parameter CONF_DB: Datentyp wird nicht unterstützt.
1	C0808300	0	Fehler im Parameter CONF_DB: Der Pointer zeigt auf einen nicht unterstützten Bereich.
1	C0808400	0	Fehler im Parameter CONF_DB: Falsche Länge des ANY-Pointers
1	C0808700	0	Fehler im Parameter CONF_DB: Inkonsistenz der Längenangabe bei den Konfigurationsdaten
1	C0808800	s, 0	field_type_id hat einen unzulässigen Wert
1	C0808900	s, 0	field_id hat einen unzulässigen Wert
1	C0808A00	s, 0	Falsche Anzahl bei subfield_cnt
1	C0808B00	s, t	subfield_id hat einen unzulässigen Wert
1	C0808C00	s, t	Teilfeld mehrfach benutzt
1	C0808D00	s, t	subfield_len hat einen falschen oder unzulässigen Wert
1	C0808E00	s, t	subfield_mode hat einen unzulässigen Wert
1	C0808F00	s, t	Im Teilfeld besteht ein Konflikt zu einem früheren Teilfeld.
1	C0809000	s, t	Die Parameter des Teilfelds sind schreibgeschützt. Z. B. erfolgte die Parametervorgabe per Projektierung oder der PNIO-Betrieb ist aktiv.
1	C0809400	s, t	Parameterwert im Teilfeld ist nicht definiert oder unzulässig
1	C080C200	0	Die Übertragung kann nicht durchgeführt werden (z. B. weil die Schnittstelle nicht erreichbar ist).

ERROR	STATUS (DW#16#..)	ERR_LOC*	Erläuterung
1	C080C300	0	Ungenügende Ressourcen (z. B. mehrfacher Aufruf von "IP_CONF" mit unterschiedlichen Parametern)
1	C080C400	0	Temporärer Kommunikationsfehler
* In der obigen Tabelle ist f die field_id und s die subfield_type_id des Fehler verursachenden Parameterblocks.			

Datenbaustein CONF_DB

Beschreibung

Im folgenden Bild ist dargestellt, wie die zu übertragenden Konfigurationsdaten im Konfigurations-DB abgelegt werden.

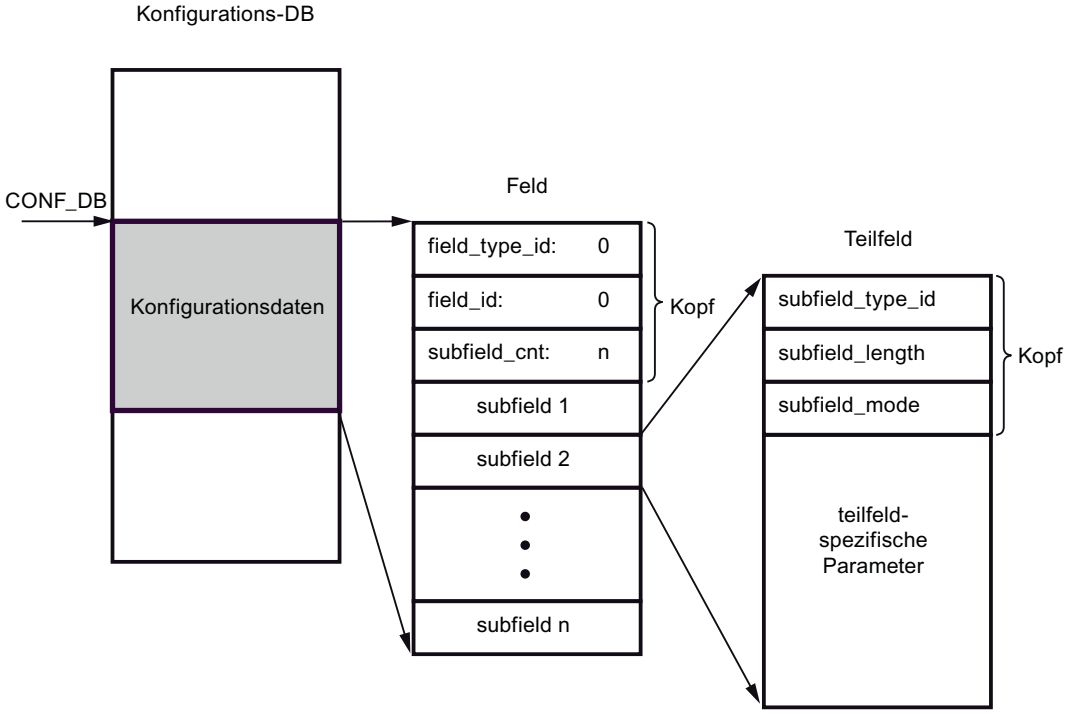

Die Konfigurationsdaten bestehen also aus genau einem Feld und mehreren Teilfeldern:

- Das Feld besteht aus einem Kopf und den Teilfeldern. Der Kopf wiederum besteht aus folgenden Elementen:
  - field_type_id (Datentyp INT): Null
  - field_id (Datentyp INT): Null
  - subfield_cnt (Datentyp INT): Anzahl der Teilfelder
- Jedes Teilfeld besteht wiederum aus einem Kopf (subfield_type_id, subfield_length, subfield_mode) und den teilfeldspezifischen Parametern. Jedes Teilfeld muss aus einer geraden Anzahl von Bytes bestehen.
  Für subfield_mode sind die folgenden Werte zulässig:
  - 1: permanente Gültigkeit der Konfigurationsdaten
  - 2: temporäre Gültigkeit der Konfigurationsdaten einschließlich Löschen bestehender permanenter Konfigurationsdaten

  Hinweis

  Derzeit ist genau ein Feld zulässig. Dessen Parameter field_type_id und field_id müssen den Wert Null haben. Weitere Felder mit anderen Werten für field_type_id und field_id bleiben zukünftigen Erweiterungen vorbehalten.

Zulässige Teilfelder

subfield_type_id	Name des Teilfelds	Erläuterung
30	SUB_IP_SUITE_IPV4	IP-Parameter: IP-Adresse, Subnetzmaske, Router-Adresse
40	SUB_NOS	PROFINET IO-Gerätename (Name of Station)

Typdefinitionen für das Teilfeld "SUB_IP_SUITE_IPV4"

Name		Datentyp	Startwert	Beschreibung
SUB_IP_SUITE_V4		Struct		
	Id	INT	30	subfield_type_id
	len	INT	18	subfield_length
	mode	INT	1	subfield_mode (1: permanent, 2: temporär)
	ipaddr_3	BYTE	b#16#C8	IP-Adresse high byte: 200
	ipaddr_2	BYTE	b#16#0C	IP-Adresse high byte: 12
	ipaddr_1	BYTE	b#16#01	IP-Adresse low byte: 1
	ipaddr_0	BYTE	b#16#90	IP-Adresse low byte: 144
	snmask_3	BYTE	b#16#FF	Subnetzmaske high byte: 255
	snmask_2	BYTE	b#16#FF	Subnetzmaske high byte: 255
	snmask_1	BYTE	b#16#FF	Subnetzmaske low byte: 255
	snmask_0	BYTE	b#16#00	Subnetzmaske low byte: 0
	router_3	BYTE	b#16#C8	Router high byte: 200
	router_2	BYTE	b#16#0C	Router high byte: 12

Name		Datentyp	Startwert	Beschreibung
	router_1	BYTE	b#16#01	Router low byte: 1
	router_0	BYTE	b#16#01	Router low byte: 1

Typdefinitionen für das Teilfeld "SUB_NOS"

Name		Datentyp	Startwert	Beschreibung
SUB_NOS		Struct		
	id	INT	40	subfield_type_id
	len	INT	246	subfield_length
	mode	INT	1	subfield_mode (1: permanent, 2: temporär)
	nos	ARRAY[1..240] of BYTE	0	Gerätename: Sie müssen das ARRAY ab dem ersten Byte belegen. Wenn das ARRAY länger ist als der zu vergebende Gerätename, müssen Sie nach dem eigentlichen Gerätenamen ein Nullbyte eintragen (konform zu IEC 61158-6-10). Andernfalls wird nos zurückgewiesen, und die Anweisung "IP_CONF (Seite 6481)" trägt den Fehlercode DW#16#C0809400 in STATUS ein. Wenn Sie das erste Byte mit Null belegen, wird der Gerätename gelöscht.

Der Gerätename unterliegt folgenden Einschränkungen:

- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein. Ein Namensbestandteil darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Der Bindestrich ist das einzige erlaubte Sonderzeichen.
- Der Gerätename darf nicht mit den Zeichen "-" und "." beginnen und auch nicht mit diesen Zeichen enden.
- Der Gerätename darf nicht mit Ziffern beginnen.
- Der Gerätename darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Gerätename darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).
- Die Zeichenfolge ".." innerhalb des Gerätenamens ist nicht zulässig.

Hinweis

Sie können das ARRAY "nos" auch kürzer als 240 Bytes anlegen, wobei die Mindestlänge 2 Bytes beträgt. In diesem Fall müssen Sie die Variable "len" (Länge des Teilfelds) entsprechend anpassen.

Datenaustausch mittels FETCH und WRITE

FW_TCP: Daten mittels FETCH und WRITE über TCP austauschen

Beschreibung

Mit der Anweisung "FW_TCP" werden die Dienste FETCH/WRITE über die TCP Verbindung zur Verfügung gestellt.

Aufruf

Da für die Dienste FETCH/WRITE jeweils eine separate TCP Verbindung benötigt wird, wird die Anweisung zweimal im OB1 aufgerufen:

- Der erste Aufruf mit Instanzdatenbaustein FETCH_TCP_DB stellt den Dienst FETCH über die TCP Verbindungen zur Verfügung.
- Der zweite Aufruf mit Instanzdatenbaustein WRITE_TCP_DB stellt die Funktion WRITE über die TCP Verbindung zur Verfügung.

Die TCP Verbindung wird nach Kalt-, Warmstart und Wiederanlauf der CPU sowie nach Änderung der Verbindungsparameter automatisch aufgebaut. Die Parameter für den Verbindungsaufbau sind im DB "ConnectParam" hinterlegt (Beschreibung: Siehe unten). Über den Eingangsparameter ENABLE besteht die Möglichkeit, den Verbindungsaufbau und -abbau manuell zu steuern. Wenn die Verbindungsparameter geändert werden, dann wird die Verbindung automatisch ab- und wieder aufgebaut.

In der folgenden Abbildung ist der Ablauf der Anweisung "FW_TCP", der im OB1 aufgerufen wird und die Dienste FETCH und WRITE über TCP Verbindungen realisiert, schematisch dargestellt.

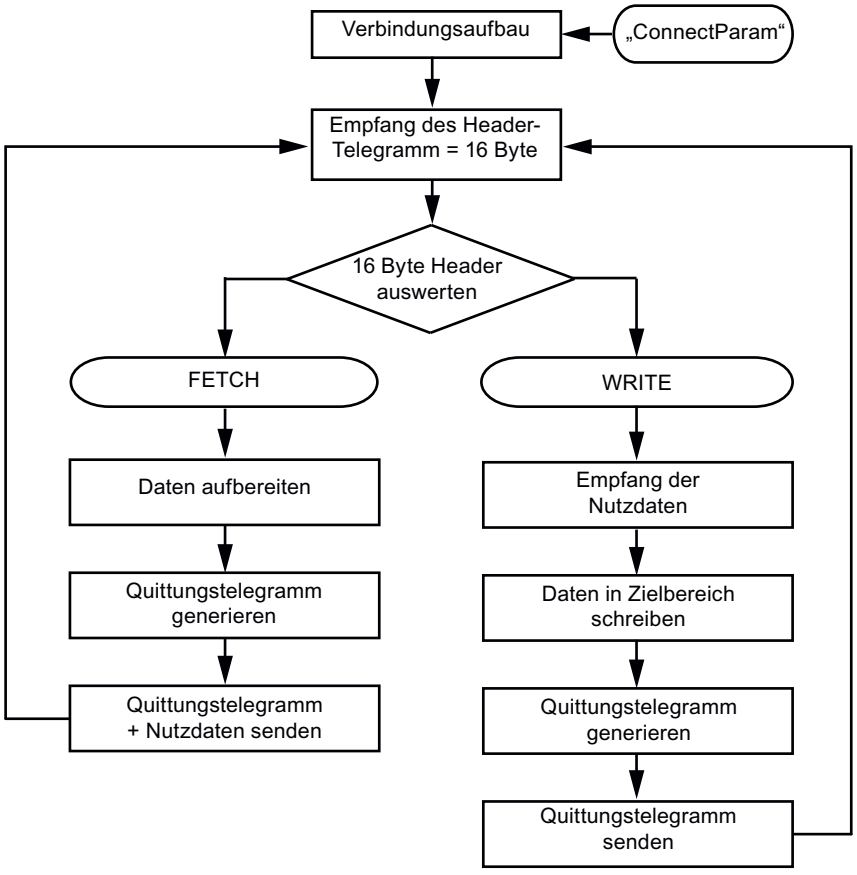

Verbindungsparameter des Instanzdatenbausteins "ConnectParam"

Folgende Verbindungsparameter werden vor dem Verbindungsaufbau überprüft:

- connection_type
- id
- local_tsap_id

Wenn einer dieser Parameter nicht korrekt definiert ist, dann wird die Kommunikationsverbindung nicht aufgebaut. Es werden keine Daten gesendet bzw. empfangen.

- Nachdem die Kommunikationsverbindung aufgebaut ist, wird ein 16 Byte langes Header-Telegramm der Dienste FETCH/WRITE empfangen. Das Header-Telegramm enthält die Information, ob ein FETCH- oder WRITE-Auftrag ansteht.
- Wenn ein FETCH-Auftrag vorliegt, dann werden die Daten, die vom Kommunikationspartner angefordert werden, im Instanz-Datenbaustein DB210 der Anweisung aufbereitet und das Quittungstelegramm generiert. Das Quittungstelegramm wird inklusive der aufbereiteten Daten gesendet.
- Wenn ein WRITE-Auftrag vorliegt, dann werden nach der Auswertung der 16 Bytes Header-Telegramm die Nutzdaten empfangen und in den Zielbereich geschrieben. Anschließend wird das Quittungstelegramm im Instanzdatenbaustein DB310 generiert und gesendet.

Die folgende Tabelle zeigt die Datenstruktur der Verbindungsparameter für eine TCP Verbindung. Diese ist 64 Byte lang.

Parameter	Datentyp	Startwert	Beschreibung
id	WORD	W#16#1	Verbindungs-ID: Der Wert des Parameters muss in folgendem Wertebereich liegen: id = W#16#0001 bis W#16#0FFF (1 bis 4095)
connection_type	BYTE	B#16#11	Verbindungstyp. Für TCP Verbindung gilt: • connection_type = B#16#11 (TCP native) oder • connection_type = B#16#1 (TCP native Kompatibilitätsmodus)
active_est	BOOL	FALSE	Passiver Verbindungsaufbau.
local_device_id	BYTE	B#16#3	Kennung für die lokale PN/IE-Schnittstelle (hier: CPU 319-3PN/DP).
local_tsap_id_len	BYTE	B#16#2	Verwendete Länge des Parameters local_tsap_id in Byte: • 0 oder 2, bei Verbindungstyp = 17 (TCP) • Für die aktive Seite ist nur der Wert "0" zulässig.
rem_subnet_id_len	BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
rem_staddr_len	BYTE	B#16#0	Länge der Adresse des Partner-Endpunkts (Parameter rem_staddr) in Byte. Da es sich um eine unspezifizierte Verbindung handelt, ist der Parameter irrelevant.
rem_tsap_id_len	BYTE	B#16#0	Verwendete Länge des Parameters rem_tsap_id in Byte: • 0 oder 2, bei Verbindungstyp = 17 (TCP). • Für die passive Seite ist nur der Wert 0 zulässig.
next_staddr_len	BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
local_tsap_id	ARRAY [1..16] of BYTE	B#16#7 B#16#D0 B#16#0 : : B#16#0	Lokaler Port für die TCP Verbindung. Mögliche Werte: 1...49151 (Empfohlene Werte: 2000...5000). • local_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung; • local_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung; • local_tsap_id[3-16] = irrelevant Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id innerhalb der CPU eindeutig ist.
rem_subnet_id	ARRAY [1..6] of BYTE	B#16#0 …	Dieser Parameter wird nicht verwendet.
rem_staddr	ARRAY [1..6] of BYTE	B#16#0	IP-Adresse des Partner-Endpunkts, z. B. für 192.168.2.3: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 2 • rem_staddr[4] = 3 • rem_staddr[5-6]= irrelevant Nicht relevant bei unspezifizierten Verbindungen mit passiven Verbindungsaufbau.

Parameter	Datentyp	Startwert	Beschreibung
rem_tsap_id	ARRAY [1..16] of BYTE	B#16#0	Port des Kommunikationspartners. Bei 17 (TCP) mögliche Werte: 1...49151 (Empfohlene Werte: 2000...5000). • rem_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung; • rem_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung; • rem_tsap_id[3-16] = irrelevant Nicht relevant bei unspezifizierten Verbindungen mit passiven Verbindungsaufbau.
next_staddr	ARRAY [1..6] of BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
spare	WORD	W#16#0	Dieser Parameter wird nicht verwendet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FW_TCP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ENABLE	Input	BOOL	E, A, M, D, L, T, Z	Mit diesem Parameter wird der Verbindungsaufbau bzw. -abbau initiiert: • 0 = Verbindung wird abgebaut • 1 = Verbindung wird aufgebaut
CONNECT	Input	ANY	D	ANY-Pointer auf den Datenbereich des DB211, welcher die Parameter für den Aufbau der TCP Verbindung enthält. Dieser Datenbereich muss mindestens 64 Byte groß sein.
ADDRMODE	Input	INT	E, A, M, D, L oder Konstante	Mit diesem Parameter wird festgelegt, wie die Adressierung der Daten des FETCH bzw. WRITE-Auftrags erfolgt (im S7- oder S5-Adressiermodus). • 0 = S7-Adressiermodus (Anfangsadresse der Daten wird als Byte Adresse interpretiert) • 1 = S5-Adressiermodus (Anfangsadresse der Daten wird als Wort Adresse interpretiert)
NDR	Output	BOOL	E, A, M, D, L	Parameter new data record: Mit diesem Parameter wird angezeigt, dass die Daten des WRITE-Auftrags erfolgreich übernommen und das Quittungstelegramm generiert und gesendet wurde.
ERROR	Output	BOOL	E, A, M, D, L	Dieser Parameter wird gesetzt, wenn: • ein Fehler beim Verbindungsaufbau bzw. -abbau aufgetreten ist • ein Fehler beim Senden bzw. Empfangen der Daten aufgetreten ist. • ein ungültiges FETCH/WRITE Header-Telegramm empfangen wurde.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MODE	Output	BYTE	E, A, M, D, L	Mit diesem Parameter wird angezeigt, ob ein FETCH- oder WRITE-Auftrag ausgeführt wird. • 0 = kein Auftrag aktiv • 1 = WRITE-Auftrag • 2 = FETCH-Auftrag
STATUS (Seite 6495)	Output	WORD	M, D	Im Parameter STATUS werden interne Kommunikationsstatusbits ausgegeben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

FW_IOT: Daten mittels FETCH und WRITE über ISO-on-TCP austauschen

Beschreibung

Mit der Anweisung "FW_IOT" werden die Dienste FETCH/WRITE über die ISO-on-TCP Verbindung zur Verfügung gestellt.

Aufruf

Da für die Dienste FETCH/WRITE jeweils eine separate ISO-on-TCP Verbindung benötigt wird, wird die Anweisung "FW_IOT" zweimal im OB1 aufgerufen.

- Der erste Aufruf mit Instanzdatenbaustein FETCH_IOT_DB stellt den Dienst FETCH über die ISO-on-TCP Verbindungen zur Verfügung.
- Der zweite Aufruf mit Instanzdatenbaustein WRITE_IOT_DB stellt den Dienst WRITE über die ISO-on-TCP Verbindung zur Verfügung.

Die ISO-on-TCP Verbindung wird nach Kalt-, Warmstart und Wiederanlauf der CPU sowie nach Änderung der Verbindungsparameter automatisch aufgebaut. Die Parameter für den Verbindungsaufbau sind im DB "ConnectParam" hinterlegt. Über den Eingangsparameter ENABLE hat der Anwender die Möglichkeit den Verbindungsaufbau und -abbau manuell zu steuern. Wenn die Verbindungsparameter geändert werden, dann wird die Verbindung automatisch ab- und wieder aufgebaut. Die folgende Tabelle zeigt die Datenstruktur der Verbindungsparameter für eine ISO-on-TCP Verbindung. Diese ist 64 Byte lang.

In der folgenden Abbildung ist der Ablauf der Anweisung "FW_IOT", der im OB1 aufgerufen wird und die Dienste FETCH und WRITE über ISO-on-TCP Verbindungen realisiert, schematisch dargestellt.

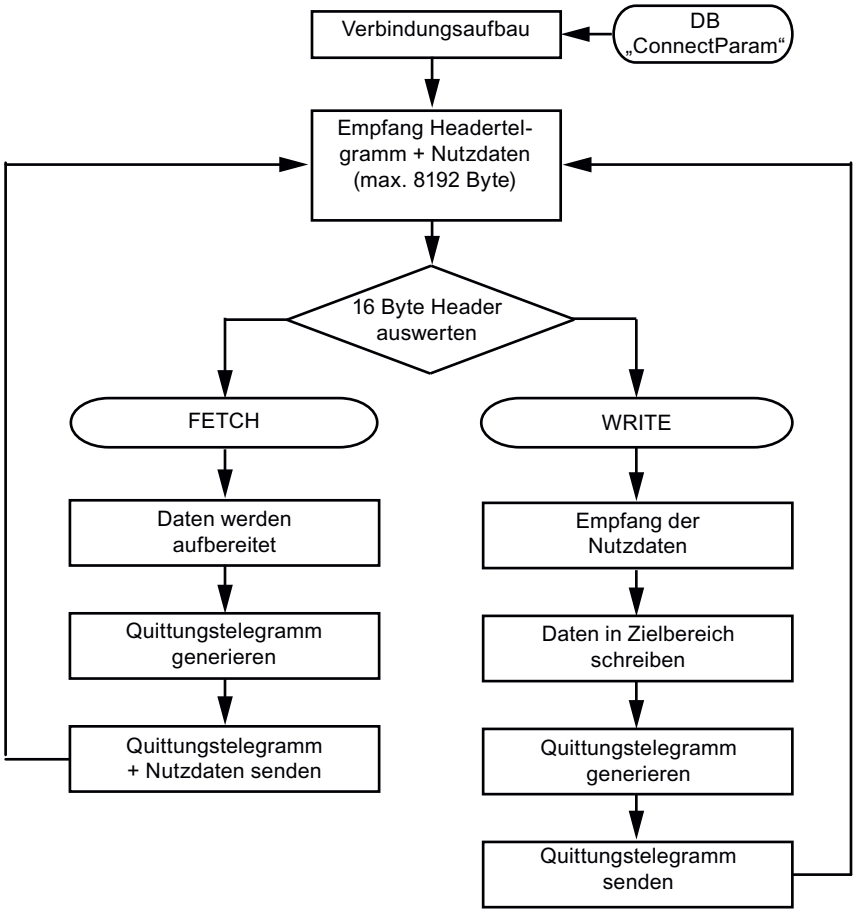

Verbindungsparameter des Instanzdatenbausteins "ConnectParam"

Folgende Verbindungsparameter werden vor dem Verbindungsaufbau überprüft:

- connection_type
- id
- local_tsap_id
- **local_tsap_id _len**

Wenn einer dieser Parameter nicht korrekt definiert ist, dann wird die Kommunikationsverbindung nicht aufgebaut. Es werden keine Daten gesendet bzw. empfangen.

- Nachdem die Kommunikationsverbindung aufgebaut ist, wird ein 16 Byte langes Header-Telegramm der Dienste FETCH/WRITE empfangen. Das Header-Telegramm enthält die Information, ob ein FETCH- oder WRITE-Auftrag ansteht.
- Wenn ein FETCH-Auftrag vorliegt, dann werden die Daten, die vom Kommunikationspartner angefordert werden, im Instanz-Datenbaustein FETCH_IOT_DB der Anweisung aufbereitet und das Quittungstelegramm generiert. Das Quittungstelegramm wird inklusive der aufbereiteten Daten gesendet.
- Wenn ein WRITE-Auftrag vorliegt, dann werden nach der Auswertung der 16 Bytes Header-Telegramm die Nutzdaten empfangen und in den Zielbereich geschrieben. Anschließend wird das Quittungstelegramm im Instanzdatenbaustein WRITE_IOT_DB generiert und gesendet.

Die folgende Tabelle zeigt die Datenstruktur der Verbindungsparameter für eine TCP Verbindung. Diese ist 64 Byte lang.

Parameter	Datentyp	Startwert	Beschreibung
id	WORD	W#16#1	Verbindungs-ID: Der Wert des Parameters muss in folgendem Wertebereich liegen: id = W#16#0001 bis W#16#0FFF (1 bis 4095)
connection_type	BYTE	B#16#12	Verbindungstyp. Für ISO-on-TCP Verbindung gilt: connection_type = B#16#12
active_est	BOOL	FALSE	Passiver Verbindungsaufbau.
local_device_id	BYTE	B#16#3	Kennung für die lokale PN/IE-Schnittstelle (hier: CPU 319-3PN/DP).
local_tsap_id_len	BYTE	B#16#2	Verwendete Länge des Parameters local_tsap_id in Byte: • 2 bis 16 bei Verbindungstyp = 18 (ISO-on-TCP)
rem_subnet_id_len	BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
rem_staddr_len	BYTE	B#16#0	Länge der Adresse des Partner-Endpunkts (Parameter rem_staddr) in Byte. Da es sich um eine unspezifizierte Verbindung handelt, ist der Parameter irrelevant.
rem_tsap_id_len	BYTE	B#16#0	Verwendete Länge des Parameters rem_tsap_id in Byte: • 2 bis 16 bei Verbindungstyp = 18 (ISO-on-TCP)
next_staddr_len	BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
local_tsap_id	ARRAY [1..16] of BYTE	B#16#7 B#16#D0 B#16#0 : : B#16#0	Lokale TSAP-ID für die ISO-on-TCP Verbindung. • local_tsap_id[1] = B#16#E0; • local_tsap_id[2] = Rack und Steckplatz der lokalen Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer); • local_tsap_id[3-16] = TSAP-Erweiterung, optional Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id innerhalb der CPU eindeutig ist.
rem_subnet_id	ARRAY [1..6] of BYTE	B#16#0 …	Dieser Parameter wird nicht verwendet.

Parameter	Datentyp	Startwert	Beschreibung
rem_staddr	ARRAY [1..6] of BYTE	B#16#0	IP-Adresse des Partner-Endpunkts, z. B. für 192.168.2.3: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 2 • rem_staddr[4] = 3 • rem_staddr[5-6]= irrelevant Nicht relevant bei unspezifizierten Verbindungen mit passiven Verbindungsaufbau.
rem_tsap_id	ARRAY [1..16] of BYTE	B#16#0	Port des Kommunikationspartners. Für Partner-TSAP-ID bei ISO-on-TCP: • rem_tsap_id[1] = B#16#E0; • rem_tsap_id[2] = Rack und Steckplatz des Partner-Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer); • rem_tsap_id[3-16] = TSAP-Erweiterung, optional Nicht relevant bei unspezifizierten Verbindungen mit passiven Verbindungsaufbau.
next_staddr	ARRAY [1..6] of BYTE	B#16#0	Dieser Parameter wird nicht verwendet.
spare	WORD	W#16#0	Dieser Parameter wird nicht verwendet.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FW_IOT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ENABLE	Input	BOOL	E, A, M, D, L, T, Z	Mit diesem Parameter wird der Verbindungsaufbau bzw. -abbau initiiert: • 0 = Verbindung wird abgebaut • 1 = Verbindung wird aufgebaut
CONNECT	Input	ANY	D	ANY-Pointer auf den Datenbereich des DB, welcher die Parameter für den Aufbau der TCP Verbindung enthält. Dieser Datenbereich muss mindestens 64 Byte groß sein.
ADDRMODE	Input	INT	E, A, M, D, L oder Konstante	Mit diesem Parameter wird festgelegt, wie die Adressierung der Daten des FETCH bzw. WRITE-Auftrags erfolgt (im S7- oder S5-Adressiermodus). • 0 = S7-Adressiermodus (Anfangsadresse der Daten wird als Byte Adresse interpretiert) • 1 = S5-Adressiermodus (Anfangsadresse der Daten wird als Wort Adresse interpretiert)

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDR	Output	BOOL	E, A, M, D, L	Parameter new data record: Mit diesem Parameter wird angezeigt, dass die Daten des WRITE-Auftrags erfolgreich übernommen und das Quittungstelegramm generiert und gesendet wurde.
ERROR	Output	BOOL	E, A, M, D, L	Dieser Parameter wird gesetzt, wenn: • ein Fehler beim Verbindungsaufbau bzw. -abbau aufgetreten ist • ein Fehler beim Senden bzw. Empfangen der Daten aufgetreten ist. • ein ungültiges FETCH/WRITE Header-Telegramm empfangen wurde.
MODE	Output	BYTE	E, A, M, D, L	Mit diesem Parameter wird angezeigt, ob ein FETCH- oder WRITE-Auftrag ausgeführt wird. • 0 = kein Auftrag aktiv • 1 = WRITE-Auftrag • 2 = FETCH-Auftrag
STATUS (Seite 6495)	Output	WORD	M, D	Im Parameter STATUS werden interne Kommunikationsstatusbits ausgegeben.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter STATUS

Aufbau des Parameters STATUS

Der Ausgangsparameter STATUS der Anweisungen "FW_TCP" und "FW_IOT" liefert Informationen zum Zustand der Kommunikation zwischen S7-Station und PC-Station.

Dieses Statuswort ist wie folgt aufgebaut:

Bit	Instanz-DB.FBStatus	Beschreibung
High-Byte		
0	NotConnected	keine Verbindung aufgebaut bzw. unterbrochen • FETCH/WRITE-Client hat den Verbindungsaufbau noch nicht initiiert • Fehler beim Verbindungsaufbau
1	ErrorConnect	Fehler beim Verbindungsaufbau (z. B.: Fehler in der Datenstruktur der Verbindungsparameter)
2	ErrorDisConnect	Fehler beim Verbindungsabbau
3	ErrorConType	fehlerhafter Verbindungstyp in der Datenstruktur der Verbindungsparameter
4	nicht belegt	
5	BusyConnect	Verbindungsaufbau läuft
6	BusyDisConnect	Verbindungsabbau läuft
7	nicht belegt	
Low-Byte		

Bit	Instanz-DB.FBStatus	Beschreibung
8	ErrorRec1	TCP Verbindung: • Fehler beim Empfangsauftrag • FETCH/WRITE Header-Telegramm konnte nicht empfangen werden ISO-on-TCP Verbindung • Fehler beim Empfangsauftrag • FETCH/WRITE Header-Telegramm inklusive der Nutzdaten konnte nicht empfangen werden
9	ErrorRec2	TCP Verbindung • Fehler beim Empfangsauftrag • Nutzdaten konnten nicht empfangen werden
10	ErrorSend	TCP und ISO-on-TCP Verbindung • Fehler beim Sendeauftrag • FETCH/WRITE Quittungstelegramm inklusive Nutzdaten konnte nicht gesendet werden
11	nicht belegt	
12	BusyRec1	Empfangsauftrag läuft
13	BusyRec2	Empfangsauftrag läuft
14	BusySend	Sendeauftrag läuft
15	ErrorHeader	undefiniertes FETCH/WRITE Header-Telegramm empfangen

Adressen und Variablen des Instanzdatenbaustein

Im Instanzdatenbaustein der Anweisung "FW_TCP" und "FW_IOT" wird der Fehlerstatus des Verbindungsaufbaus und -abbaus sowie der Fehlerstatus des Sende- und Empfangsauftrags gespeichert.

- Der Instanzdatenbaustein der Anweisung "FW_TCP" umfasst 8348 Byte.
- Der Instanzdatenbaustein der Anweisung "FW_IOT" umfasst 16544 Byte.

Die folgende Tabelle zeigt eine Übersicht der Adressen und Variablen des Instanzdatenbaustein, wo die Status gespeichert werden.

Variable	Datentyp	Adresse	Beschreibung
DoneError.STATUS_Connect	WORD	22.0	Status Verbindungsaufbau
DoneError.STATUS_REC_1	WORD	24.0	Status des ersten Empfangsauftrags
DoneError.STATUS_REC_2	WORD	26.0	Status des zweiten Empfangsauftrags (nur bei TCP)
DoneError.STATUS_SEND	WORD	28.0	Status des Sendeauftrags
DoneError.STATUS_DisConnect	WORD	30.0	Status Verbindungsabbau

Werte des Parameters STATUS

Die folgende Tabelle zeigt eine Übersicht der Statuswerte des Verbindungsaufbaus.

STATUS (W#16#...)	Beschreibung
0000	Verbindung konnte aufgebaut werden.
8086	Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.
8087	Maximale Anzahl der Verbindungen ist erreicht, keine weitere Verbindung möglich.
8089	Der Parameter CONNECT zeigt nicht auf einen DB.
809A	Der Parameter CONNECT zeigt auf ein Feld, das nicht die Länge der Verbindungsbeschreibung hat.
809B	Die in der Verbindungsbeschreibung angegebene local_device_id passt nicht zur CPU.
80A1	Verbindung bzw. Port ist bereits belegt.
80A2	Lokaler oder remoter Port ist vom System belegt.
80A3	Es wird versucht, eine nicht bestehende Verbindung abzubauen.
80A4	IP-Adresse des remoten Verbindungsendpunkts ist ungültig, eventuell stimmt sie mit der eigenen IP-Adresse überein.
80B2	Der Parameter CONNECT zeigt auf einen DB, der mit dem Schlüsselwort UNLINKED erzeugt wurde.
80B3	Inkonsistente Parametrierung
80B4	Sie haben bei der Protokollvariante ISO on TCP (connection_type = B#16#12) beim passiven Verbindungsaufbau (active_est = FALSE) eine oder mehrere der folgenden Bedingungen verletzt: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
80B6	Parametrierfehler bezüglich des Parameters connection_type.
80B7	Fehler in der Verbindungsbeschreibung
80C3	Alle Verbindungsressourcen sind belegt: • Ein anderer Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet. • Interner Ressourcenmangel
80C4	Temporärer Kommunikationsfehler: • Verbindung kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametriert. • Die eingerichtete Verbindung wird momentan durch die Anweisung "TDISCON" entfernt.
82xx	allgemeine Fehlerinformationen

4.2.5.4 Webserver

WWW: Anwenderdefinierte Webseiten synchronisieren

Beschreibung

Die Anweisung WWW initialisiert den Webserver der CPU bzw. synchronisiert anwenderdefinierte Webseiten mit dem Anwenderprogramm in der CPU.

Anwenderdefinierten Webseiten bilden zusammen mit dem Webserver der CPU die Möglichkeit, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen.

Mit Hilfe von Scriptanweisungen (z. B. Javascript) und HTML-Code in anwenderdefinierten Webseiten können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an die CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen. Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung WWW im Anwenderprogramm aufrufen.

Initialisierung

Anwenderdefinierte Webseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Die Datenbausteine mit codierten Webseiten werden Fragment-DBs genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin anwenderdefinierte Webseiten codiert sind. Durch die Anweisung "WWW" z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die anwenderdefinierten Webseiten über einen Webbrowser erreichbar.

Synchronisierung

Wenn das Anwenderprogramm mit den anwenderdefinierten Webseiten interagieren soll, muss die Anweisung WWW im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragendenden Webbrowser zusammenstellen und zurücksenden

In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer vom Webbrowser angeforderten Webseite dem Webserver mitgeteilt werden.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WWW":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
CTRL_DB	Input	BLOCK_DB	E, A, M, D, L oder Konstante	Datenbaustein, der die anwenderdefinierten Webseiten beschreibt (Web-Control-DB)
RET_VAL	Output	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten. Es stehen keine Webseitenanforderungen an, die vom Anwenderprogramm freigegeben werden müssen.
00xy	x: zeigt an, ob bei der Initialisierung des Web-Control-DBs (CTRL_DB) ein Fehler aufgetreten ist: x=0: keine Fehler aufgetreten. x=1: Fehler aufgetreten. Der Fehler ist im Byte "CTRL_DB.commandstate.last_error" des Web-Control-DBs kodiert, siehe Beschreibung zum Web-Control-DB. y: Nummer des anstehenden Requests. Mehrere Requests sind möglich (z. B. Request "0" und "1" stehen an: y="3". y="1": Request "0" y="2": Request "1" y="4": Request "2" y="8": Request "3"
803A	Der angegebene Web-Control-DB ist auf der CPU nicht vorhanden.
8081	Falsche Version oder falsches Format des Web-Control-DBs
80C1	Es sind keine Ressourcen vorhanden, um die Web-Applikation zu initialisieren, z. B. weil nur zwei oder vier Web-Applikationen laufen dürfen.

4.2.5.5 Weitere Kommunikation

4.2.5.6 Kommunikationsprozessor

Punkt-zu-Punkt

Überblick Freeport-Kommunikation

STEP 7 bietet erweiterte Anweisungen, mit denen die Freeport-Kommunikation über ein im Anwenderprogramm vorgegebenes Protokoll realisiert werden kann. Diese Anweisungen lassen sich in zwei Kategorien unterteilen:

- Konfigurationsanweisungen
- Kommunikationsanweisungen

Hinweis

CPU-Mengengerüste

Die Freeport- Anweisungen kommunizieren mit den Kommunikationsmodulen durch Lesen oder Schreiben von Datensätzen.
Beim Einsatz der Anweisungen ist daher das Mengengerüst der jeweiligen CPU für das Lesen und Schreiben von Datensätzen zu beachten.
Wenn mehrere Anweisungen auf einer CPU gleichzeitig Datensätze lesen oder schreiben sollen, so müssen diese Anweisungen evtl. durch das Anwenderprogramm zeitlich versetzt aufgerufen werden.

Konfigurationsanweisungen

Bevor das Anwenderprogramm die Freeport-Kommunikation starten kann, müssen die Kommunikationsschnittstelle und die Parameter zum Senden und Empfangen der Daten konfiguriert werden.

Die Schnittstellenkonfiguration und die Datenkonfiguration können für jedes CM in der Gerätekonfiguration oder mit den folgenden Anweisungen Ihres Anwenderprogramms durchgeführt werden:

- Port_Config (Seite 6504)
- Send_Config (Seite 6507)
- Receive_Config (Seite 6509)
- P3964_Config (Seite 6514)

ACHTUNG

Gerätekonfiguration <-> Konfigurationsanweisungen

Die Parameter der Gerätekonfiguration werden bei jedem Power On der CPU (Spannungswiederkehr) zum CM übertragen.

Die Parameter der Konfigurationsanweisungen werden entsprechend Ihrem Anwenderprogramm zum CM übertragen.

Es gibt keinen Abgleich zwischen den Parametern der Gerätekonfiguration und den Parametern der Konfigurationsanweisungen, d. h. die Parameter der Konfigurationsanweisungen werden nicht in die Gerätekonfiguration der CPU übernommen. Sie haben mit ihrem Anwenderprogramm die Kontrolle darüber, wann welche Parameter im CM wirksam sind.

Kommunikationsanweisungen

Mit den Anweisungen für die Freeport-Kommunikation kann das Anwenderprogramm Daten an die Kommunikationsschnittstellen senden und von diesen Daten empfangen. Die CMs senden und empfangen Daten an die bzw. von den Kommunikationsteilnehmern.

- Send_P2P (Seite 6516)
- Receive_P2P (Seite 6520)

Hinweis

Datenkonsistenz

- Falls die zu sendenden Daten konsistent übertragen werden sollen, dürfen diese nach dem Flankenanstieg des Parameters REQ bis zum Setzen von DONE durch die Anweisung Send_P2P nicht verändert werden.
- Falls die Empfangsdaten konsistent ausgelesen werden sollen, dürfen diese nur in dem Zyklus ausgewertet werden, in dem NDR = TRUE ist.

Mit zusätzlichen Anweisungen kann der Empfangspuffer zurückgesetzt und es können spezielle RS232-Signale abgefragt und gesetzt werden.

- Receive_Reset (Seite 6522)
- Signal_Get (Seite 6523)
- Signal_Set (Seite 6524)

Mit den folgenden Anweisungen können erweiterte Funktionen, sofern vom Modul unterstützt, gelesen bzw. geschrieben werden.

- Get_Features (Seite 6526)
- Set_Features (Seite 6527)

Alle Freeport-Anweisungen arbeiten asynchron. Die Anweisungen müssen deshalb solange aufgerufen werden, bis durch den Ausgangsparameter DONE die vollständige Ausführung angezeigt wird.

Mit Hilfe einer Abfragearchitektur kann das Anwenderprogramm den Sende- und Empfangsstatus feststellen. Send_P2P und Receive_P2P können gleichzeitig ausgeführt werden. Die Kommunikationsmodule puffern die Sende- und Empfangsdaten je nach Bedarf bis zu einer modulspezifischen maximalen Puffergröße.

Hinweis

Auflösung von Bitzeiten

Für verschiedene Parameter wird die Anzahl der Bitzeiten bei der konfigurierten Datenübertragungsrate angegeben. Durch Angabe des Parameters in Bitzeiten ist der Parameter unabhängig von der Datenübertragungsrate. Alle Parameter mit der Einheit Bitzeiten können mit einer maximalen Anzahl von 65535 angegeben werden.

Anwendung der Anweisungen

Die Freeport-Anweisungen müssen zyklisch aufgerufen werden, um nach empfangenen Daten oder auf das Übertragungsende eines Sendevorgangs abzufragen.

Abhängig von der Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.
Wurde der Auftrag mit DONE = TRUE bzw. NDR = TRUE durchlaufen, wurde der Auftrag fehlerfrei ausgeführt.

Hinweis

STATUS sichern

Die Parameter DONE, NDR, ERROR und STATUS stehen nur für einen Bausteindurchlauf an. Zur Anzeige des STATUS sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.

Master

Typische Sequenz für einen Master:

1. Die Anweisung Send_P2P veranlasst eine Übertragung zum CM.
   Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen.
2. Die Anweisung Send_P2P wird in aufeinander folgenden Zyklen ausgeführt, um den Status des Übertragungsvorgangs abzufragen.
3. Wenn die Anweisung Send_P2P am Ausgang DONE meldet, dass die Übertragung beendet ist, kann der Anwendercode den Empfang der Antwort vorbereiten.
4. Die Anweisung Receive_P2P wird wiederholt ausgeführt, um auf eine Antwort abzufragen. Wenn das CM Antwortdaten erfasst hat, kopiert die Anweisung Receive_P2P die Antwort in die CPU und meldet am Ausgang NDR, dass neue Daten empfangen wurden.
5. Das Anwenderprogramm kann die Antwort verarbeiten.
6. Zurück zu Schritt 1 und Wiederholung der Sequenz.

Slave

Typische Sequenz für einen Slave:

1. Das Anwenderprogramm führt die Anweisung Receive_P2P in jedem Zyklus aus.
2. Wenn das CM eine Anforderung empfangen hat, meldet die Anweisung Receive_P2P am Ausgang NDR, dass neue Daten bereit sind, und die Anforderung wird in die CPU kopiert.
3. Das Anwenderprogramm verarbeitet die Anforderung und erzeugt eine Antwort.
4. Mit der Anweisung Send_P2P wird die Antwort an den Master zurückgesendet.
5. Die Anweisung Send_P2P muss wiederholt ausgeführt werden, um sicherzustellen, dass der Sendevorgang stattfindet.
6. Zurück zu Schritt 1 und Wiederholung der Sequenz.

Der Slave muss dafür sorgen, dass Receive_P2P entsprechend oft aufgerufen wird, damit eine Übertragung vom Master empfangen werden kann, bevor dieser beim Warten auf eine

Antwort wegen Zeitüberschreitung den Vorgang abbricht. Um diese Aufgabe zu erfüllen, kann das Anwenderprogramm Receive_P2P aus einem Zyklus-OB heraus aufrufen, dessen Zykluszeit hinreichend kurz ist, um eine Übertragung vom Master vor dem Ablauf der Timeout-Einstellung zu empfangen. Wird die Zykluszeit für den OB so eingestellt, dass zwei Ausführungen innerhalb der Timeout-Einstellung des Masters erfolgen, kann das Anwenderprogramm alle Übertragungen ohne Verlust empfangen.

Gemeinsame Parameter für Freeport-Operationen

Tabelle 4-301 Gemeinsame Eingangsparameter der Freeport-Anweisungen

Parameter	Beschreibung
REQ	Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Eine weitere Flanke am REQ darf erst nach Auftragsabschluss (DONE bzw. ERROR) erzeugt werden. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen. Wenn Sie eine Freeport-Anweisung in Ihr Programm einfügen, werden Sie von STEP 7 aufgefordert, den Instanz-DB anzugeben (oder von STEP 7 einen entsprechenden Instanz-DB anlegen zu lassen). Verwenden Sie für jeden PtP-Anweisungsaufruf einen eindeutigen DB.
PORT	Eine Portadresse wird während der Konfiguration des Kommunikationsmoduls vergeben. Mit dem Parameter PORT wird der Anweisung die Zuordnung zu einem bestimmten Kommunikationsmodul mitgeteilt. Nach der Konfiguration kann ein symbolischer Name für den Standardport ausgewählt werden. Der zugewiesene CM-Portwert ist die Eigenschaft "HW-Kennung" der Gerätekonfiguration bei S7-1200/1500 bzw. die "Eingangsadresse" bei S7-300/400. Der symbolische Name des Ports wird in der Symboltabelle zugewiesen.

Die Ausgangsparameter DONE, NDR, ERROR und STATUS der Freeport-Anweisungen zeigen den Ausführungsstatus der Freeport-Funktionen an.

Tabelle 4-302 Ausgangsparameter DONE, NDR, ERROR und STATUS

Parameter	Datentyp	Standard	Beschreibung
DONE	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass die letzte Anforderung fehlerfrei abgeschlossen wurde; andernfalls FALSE.
NDR	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass neue Daten empfangen wurden; andernfalls FALSE.
ERROR	Bool	FALSE	Wird einen Zyklus lang auf TRUE gesetzt, um anzuzeigen, dass die letzte Anforderung mit Fehler abgeschlossen wurde, der entsprechende Fehlercode befindet sich in STATUS; andernfalls FALSE.
STATUS	Word	16#0000 oder 16#7000	Ergebniszustand: • Ist das Bit DONE oder NDR gesetzt, wird STATUS auf 0/16#7000 oder auf einen spezifischen Fehlercode gesetzt. • Ist das Bit ERROR gesetzt, wird am STATUS ein Fehlercode angezeigt. • Wird keines der oben aufgeführten Bits gesetzt, kann die Anweisung Statusergebnisse zurückgeben, die den aktuellen Zustand der Funktion beschreiben. Der Wert in STATUS ist bis zum nächsten Aufruf dieser Anweisung (mit ein und derselben Portadresse) gültig.

Tabelle 4-303 Durchgangsparameter COM_RST

Parameter	Datentyp	Standard	Beschreibung
COM_RST	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST wieder auf FALSE gesetzt. Hinweis: Sie müssen COM_RST im Anlauf auf TRUE setzen und sollten den Parameter danach nicht mehr ändern, d. h. beim Aufruf der Anweisung keinen Wert zuweisen. Nach Initialisierung des Instanz-DBs wird COM_RST von der Anweisung zurückgesetzt.

Hinweis

Beachten Sie, dass die Parameter DONE, NDR, ERROR und STATUS nur einen Zyklus lang gesetzt sind.

Tabelle 4-304 Gemeinsame Fehlercodes

Fehlercode	Beschreibung
16#0000	Kein Fehler
16#7000	Funktion nicht aktiv
16#7001	Erstaufruf nach dem Starten einer Anforderung.
16#7002	Folgeaufruf nach dem Starten einer Anforderung.
16#8x3A	Unzulässiger Pointer in Parameter x

Tabelle 4-305 Gemeinsame Fehlerklassen des Parameters STATUS

Beschreibung der Klasse	Fehlerklassen	Beschreibung
Portkonfiguration	16#81Ax	Für die Beschreibung häufiger Fehler in der Schnittstellenkonfiguration
Sendekonfiguration	16#81Bx	Für die Beschreibung von Fehlern in der Sendekonfiguration
Empfangskonfiguration	16#81Cx	Für die Beschreibung von Fehlern in der Empfangskonfiguration
Senden	16#81Dx	Für die Beschreibung von Laufzeitfehlern beim Senden
Empfangen	16#81Ex	Für die Beschreibung von Laufzeitfehlern beim Empfangen
RS232-Begleitsignale	16#81Fx	Für die Beschreibung von Fehlern in Verbindung mit der Signalverarbeitung

Port_Config: PtP-Kommunikationsport projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Port_Config (Port-Konfiguration) können Sie Parameter wie die Datenübertragungsrate über Ihr Programm im laufenden Betrieb ändern. Die im CM anstehenden Daten werden mit der Ausführung von Port_Config gelöscht.

Die Konfigurationsänderungen von Port_Config werden nicht in der CPU sondern auf dem CM gespeichert. Bei Spannungswiederkehr wird das CM mit den in der Gerätekonfiguration gespeicherten Daten parametriert.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
PROTOCOL	IN	UInt	Word	0	Protokoll • 0 = Freeport-Protokoll • 1 = Protokoll 3964(R)
BAUD	IN	UInt	Word	6	Datenübertragungsrate des Ports: • 1 = 300 Bit/s • 2 = 600 Bit/s • 3 = 1200 Bit/s • 4 = 2400 Bit/s • 5 = 4800 Bit/s • 6 = 9600 Bit/s • 7 = 19200 Bit/s • 8 = 38400 Bit/s • 9 = 57600 Bit/s • 10 = 76800 Bit/s • 11 = 115200 Bit/s

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
PARITY	IN	UInt	Word	1	Parität des Ports: • 1 = Keine Parität • 2 = Gerade Parität • 3 = Ungerade Parität • 4 = Mark-Parität • 5 = Space-Parität • 6 = Beliebig
DATABITS	IN	UInt	Word	1	Bits pro Zeichen: • 1 = 8 Datenbits • 2 = 7 Datenbits
STOPBITS	IN	UInt	Word	1	Stoppbits: • 1 = 1 Stoppbit • 2 = 2 Stoppbits
FLOWCTRL	IN	UInt	Word	1	Flusskontrolle: • 1 = Keine Flusskontrolle • 2 = XON/XOFF • 3 = Hardware-RTS immer EIN • 4 = Hardware-RTS geschaltet • 5 = Hardware-RTS immer EIN, DTR/DSR ignorieren
XON-CHAR	IN	Char		16#0011	Gibt das Zeichen an, das als XON-Zeichen dient. Dies ist typischerweise ein DC1-Zeichen (11H). Dieser Parameter wird nur ausgewertet, wenn die Software-Flusskontrolle aktiviert ist.
XOFF-CHAR	IN	Char		16#0013	Gibt das Zeichen an, das als XOFF-Zeichen dient. Dies ist typischerweise ein DC3-Zeichen (13H). Dieser Parameter wird nur ausgewertet, wenn die Software-Flusskontrolle aktiviert ist.
WAITIME	IN	UInt	Word	2000	Gibt an, wie lange nach dem Empfang eines XOFF-Zeichens auf ein XON-Zeichen gewartet werden soll bzw. wie lange nach CTS = OFF auf CTS = ON gewartet werden soll (0 bis 65535 ms). Dieser Parameter wird nur ausgewertet, wenn die Flusskontrolle aktiviert ist.
MODE	IN	USInt	Byte	0	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master; CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave; CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1]

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
LINE_PRE	IN	USInt	Byte	0	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) : Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V : Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.
BRK_DET	IN	USInt	Byte	0	Break-Erkennung Zulassig sind: • 0 = Break-Erkennung deaktiviert • 1 = Break-Erkennung aktiviert
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Ein Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Ein Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

[1)] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Send_Config: PtP-Sender projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Send_Config (Sendekonfiguration) können Sie im laufenden Betrieb Sendeparameter über Ihr Programm ändern (Bedingungen, die den Beginn und das Ende der zu sendenden Daten kennzeichnen). In einem CM anstehende Daten werden gelöscht, wenn Send_Config ausgeführt wird.

Die Konfigurationsänderungen von Send_Config werden nicht in der CPU sondern auf dem CM gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Aktiviert die Konfigurationsänderung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
RTSONDLY	IN	UInt	Word	0	Anzahl der Millisekunden, die nach Aktivierung von RTS gewartet werden soll, bevor eine Übertragung von Sende-Daten gestartet wird. Dieser Parameter ist nur gültig, wenn die Hardwareflusskontrolle aktiviert ist. Der gültige Bereich ist 0 bis 65535 ms. Der Wert 0 deaktiviert die Funktion.
RTSOFFDLY	IN	UInt	Word	0	Anzahl der Millisekunden, die nach der Übertragung von Sende-Daten gewartet werden soll, bevor RTS deaktiviert wird: Dieser Parameter ist nur gültig, wenn die Hardwareflusskontrolle aktiviert ist. Der gültige Bereich ist 0 bis 65535 ms. Der Wert 0 deaktiviert die Funktion.
BREAK	IN	UInt	Word	0	Dieser Parameter gibt an, dass vor dem Start jedes Telegramms für die angegebene Anzahl von Bitzeiten ein BREAK gesendet wird. Maximal sind 65535 Bitzeiten möglich. Der Wert 0 deaktiviert die Funktion.
IDLELINE	IN	UInt	Word	0	Dieser Parameter gibt an, dass vor dem Start jedes Telegramms für die angegebene Anzahl von Bitzeiten die Leitung im Leerlauf bleibt. Maximal sind 65535 Bitzeiten möglich. Der Wert 0 deaktiviert die Funktion.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
USR_END	IN	STRING[2]		0	Eingabe der Endezeichen. Es können maximal 2 Endezeichen projektiert werden. Es wird einschließlich der/des Endezeichen(s) gesendet, unabhängig von der parametrierten Telegrammlänge.
APP_END	IN	STRING[5]		0	Eingabe anzuhängende Zeichen. Es können maximal 5 Zeichen angehängt werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Receive_Config: PtP-Empfänger projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Receive_Config (Empfangskonfiguration) können Sie über Ihr Programm Empfangsparameter im laufenden Betreib ändern. Diese Anweisung konfiguriert die Bedingungen, die den Beginn und das Ende empfangener Daten kennzeichnen. In einem CM anstehende Daten werden gelöscht, wenn Receive_Config ausgeführt wird.

Die Konfigurationsänderungen von Receive_Config werden auf dem CM **nicht remanent** gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt. Die Anweisung Receive_Config muss deshalb aus dem Anwenderprogramm bei Spannungswiederkehr der CPU oder des Kommunikationsmoduls erneut aufgerufen werden, um die in der Gerätekonfiguration gespeicherten Parameter wieder zu überschreiben.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Aktiviert die Konfigurationsänderung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
RECEIVE_CONDITIONS	IN	Variant	Any	-	Die Datenstruktur von Receive_Conditions gibt die Anfangs- und Endbedingungen an, anhand derer Anfang und Ende eines Telegramms identifiziert werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Anfangsbedingungen für die Anweisung Receive_P2P

Die Anweisung Receive_P2P nutzt die in der Gerätekonfiguration oder von der Anweisung Receive_Config angegebene Konfiguration, um Anfang und Ende von Telegrammen der Freeport-Kommunikation zu ermitteln. Der Telegrammanfang wird von den Startbedingungen festgelegt. Der Telegrammanfang kann anhand von einer oder mehreren Startbedingungen ermittelt werden.
Ist sowohl Break als auch Idle Line aktiviert, so muss zunächst Break und dann zusätzlich Idle Line erfüllt sein. Danach genügt eine der weiteren Bedingungen (Startzeichen oder Anfangssequenz), um die Datenübertragung zu starten.
Die Startbedingung "Beliebiges Zeichen" kann nicht mit anderen Startbedingungen kombiniert werden.

Datentypstruktur des Parameters Receive_Conditions, Teil 1 (Startbedingungen)

Tabelle 4-306 Struktur von Receive_Conditions für Startbedingungen

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
START.START-COND	IN	Word	16#0002	Angabe der Startbedingung • 01H - Erkennung des Startzeichens • 02H - Beliebiges Zeichen • 04H - Erkennung eines Line Break • 08H - Erkennung einer Idle Line • 10H - Erkennung Anfangssequenz 1 • 20H - Erkennung Anfangssequenz 2 • 40H - Erkennung Anfangssequenz 3 • 80H - Erkennung Anfangssequenz 4 Die Startbedingungen können durch Addition der Werte kombiniert werden.
START.IDLE-TIME	IN	Word	16#0028	Die Anzahl der erforderlichen Bitzeiten im Leerlauf, damit ein neuer Telegrammanfang erkannt wird (Standardwert: W#16#28). Nur in Verbindung mit der Bedingung "Erkennung einer Idle Line". 0 bis FFFF
START.START-CHAR	IN	Byte	16#0002	Das Startzeichen für die Bedingung "Startzeichen". (Standardwert: B#16#2)
START.SEQ[1]. CTL	IN	Byte	0	Anfangssequenz 1, Vergleich für jedes Zeichen deaktivieren/aktivieren: (Standardwert: B#16#0) Dies sind die Aktivierungsbits für jedes Zeichen der Startzeichenfolge. • 01H - Zeichen 1 • 02H - Zeichen 2 • 04H - Zeichen 3 • 08H - Zeichen 4 • 10H - Zeichen 5 Wird ein Bit für ein bestimmtes Zeichen deaktiviert, bedeutet dies, dass an dieser Position der Zeichenfolge jedes Zeichen eine gültige Startzeichenfolge darstellt (z. B 1FH = alle 5 Zeichen werden interpretiert).
START.SEQ[1]. STR[1] .. START.SEQ[1]. STR.[5]	IN	Char[5]	0	Anfangssequenz 1, Startzeichen (5 Zeichen).
START.SEQ[2]. CTL	IN	Byte	0	Anfangssequenz 2, Vergleich für jedes Zeichen deaktivieren/aktivieren. Standardwert: B#16#0)
START.SEQ[2]. STR[1] .. START.SEQ[2]. STR.[5]	IN	Char[5]	0	Anfangssequenz 2, Startzeichen (5 Zeichen).
START.SEQ[3]. CTL	IN	Byte	0	Anfangssequenz 3, Vergleich für jedes Zeichen deaktivieren/aktivieren. Standardwert: B#16#0

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
START.SEQ[3]. STR[1] .. START.SEQ[3]. STR.[5]	IN	Char[5]	0	Anfangssequenz 3, Startzeichen (5 Zeichen).
START.SEQ[4]. CTL	IN	Byte	0	Anfangssequenz 4, Vergleich für jedes Zeichen deaktivieren/ aktivieren. Standardwert: B#16#0
START.SEQ[4]. STR[1] .. START.SEQ[4]. STR.[5]	IN	Char[5]	0	Anfangssequenz 4, Startzeichen (5 Zeichen),

Beispiel

Sehen Sie sich die folgende empfangene hexadezimal-codierten Daten an: "68 10 aa 68 bb 10 aa 16". Die konfigurierten Startzeichenfolgen finden Sie in der folgenden Tabelle. Startzeichenfolgen werden ausgewertet, nachdem das erste Zeichen 68H erfolgreich empfangen wurde. Nach erfolgreichem Empfang des vierten Zeichens (das zweite 68H) ist die Startbedingung 1 erfüllt. Wenn die Startbedingungen erfüllt sind, beginnt die Auswertung der Endebedingungen.

Die Verarbeitung der Startzeichenfolge kann aufgrund verschiedener Fehler bei Parität, Framing oder Zeitabständen zwischen den Zeichen beendet werden. Diese Fehler führen dazu, dass die Daten nicht empfangen werden, weil die Startbedingung nicht erfüllt wurde (es erfolgt eine Fehlermeldung).

Tabelle 4-307 Startbedingungen

Startbedingung	Erstes Zeichen	Erstes Zeichen +1	Erstes Zeichen +2	Erstes Zeichen +3	Erstes Zeichen +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Endebedingungen für die Anweisung Receive_P2P

Das Ende eines Telegramms wird durch das erste Auftreten einer oder mehrerer konfigurierter Endebedingungen festgelegt.

Sie können die Endebedingungen entweder in den Eigenschaften der Kommunikationsschnittstelle in der Gerätekonfiguration oder über die Anweisung Receive_Config konfigurieren. Nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls werden die Empfangsparameter (Start- und Endebedingungen) wieder auf die Einstellungen der Gerätekonfiguration gesetzt. Wenn das STEP 7 Anwenderprogramm Receive_Config ausführt, werden die Einstellungen auf die Parameter von Receive_Config gesetzt.

Datentypstruktur des Parameters Receive_Conditions, Teil 2 (Endebedingungen)

Tabelle 4-308 Struktur von Receive_Conditions für Endebedingungen

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
END.END-COND	IN	Word	0	Dieser Parameter gibt die Bedingung für das Telegrammende an: • 01H - Antwortzeit-Zeitüberschreitung • 02H - Nachrichten-Zeitüberschreitung • 04H - Zeichenverzugszeit • 08H - Maximale Telegrammlänge • 10H - Lese Nachrichtenlänge aus Nachricht (N+LEN+M) • 20H - Endesequenz • 40H - Feste Telegrammlänge
END.FIXLEN	IN	Word	1	Feste Telegrammlänge: Wird nur verwendet, falls die Endebedingung "Feste Telegrammlänge" ausgewählt ist. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.MAXLEN	IN	Word	1	Maximale Telegrammlänge: Wird nur verwendet, falls die Endebedingung "Maximale Telegrammlänge" ausgewählt ist. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.N	IN	Word	0	Byteposition des Längenfelds im Telegramm. Wird nur bei der Endebedingung N+LEN+M verwendet. 1 bis 4000 Byte (modulabhängig bis 4 kByte)
END.LENGTH-SIZE	IN	Word	0	Größe des Längenfeldes (1, 2 oder 4 Byte). Wird nur bei der Endebedingung N+LEN+M verwendet.
END.LENGTHM	IN	Word	0	Anzahl der Zeichen nach dem Längenfeld, die nicht im Wert des Längenfelds enthalten sind. Diese Angabe wird nur bei der Endebedingung N+LEN+M verwendet. 0 bis 255 Bytes
END.RCVTIME	IN	Word	200	Geben Sie an, wie lange nach dem Senden eines Telegramms auf das erste empfangene Zeichen gewartet werden soll. Die Empfangsanweisung wird mit einer Fehlermeldung beendet, wenn nicht innerhalb der angegebenen Zeit ein Zeichen erfolgreich empfangen wird. Diese Angabe wird nur bei der Bedingung "Antwortzeit-Überschreitung" verwendet. (0 bis 65535 ms). Hinweis: Dieser Parameter kann nicht als alleiniges Endekriterium sondern nur in Verbindung mit mindestens einer weiteren Endbedingung eingesetzt werden.
END.MSGTIME	IN	Word	200	Geben Sie an, wie lange nach dem Empfang des ersten Zeichens auf den vollständigen Empfang des gesamten Telegramms gewartet werden soll. Dieser Parameter wird nur verwendet, wenn die Bedingung "Nachrichten-Zeitüberschreitung" ausgewählt ist. (0 bis 65535 ms)
END.CHARGAP	IN	Word	12	Geben Sie die max. Anzahl der Bitzeiten zwischen den Zeichen an. Falls die Anzahl der Bitzeiten zwischen den Zeichen den angegebenen Wert überschreitet, ist die Endebedingung erfüllt. Diese Angabe wird nur bei der Bedingung "Zeichenverzugszeit" verwendet. (0 bis 65535 Bitzeiten)

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
END.SEQ.CTL	IN	Byte	0	Zeichenfolge 1, Vergleich für jedes Zeichen deaktivieren/aktivieren: Dies sind die Aktivierungsbits für jedes Zeichen der Endezeichenfolge. Zeichen 1 ist Bit 0, Zeichen 2 ist Bit 1, ..., Zeichen 5 ist Bit 4. Wird ein Bit für ein bestimmtes Zeichen deaktiviert, bedeutet dies, dass an dieser Position der Zeichenfolge jedes Zeichen eine Übereinstimmung darstellt.
END.SEQ.STR[1] .. END.SEQ.STR[5]	IN	Char[5]	0	Zeichenfolge 1, Startzeichen (5 Zeichen)

Tabelle 4-309 Allgemeine Parameter der Anweisung Receive_P2P

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
GENERAL.MBUF_SIZE	IN	Byte	255	Eingabe der Anzahl der Telegramme, die im Empfangspuffer des CM gepuffert werden sollen. Sind keine weiteren Bedingungen aktiv, die das Verhalten des Empfangspuffers beeinflussen (Überschreiben verhindern, Datenflusskontrolle), so werden nach Erreichen des Grenzwerts weitere Telegramme verworfen. (1 bis 255 Telegramme)
GENERAL.OW_PROT	IN	Byte	0	Aktiviert das Nicht-Überschreiben des gepufferten Telegramms, wenn der CM ein neues Telegramm empfängt und der Empfangspuffer des CM noch nicht ausgelesen wurde. Damit wird verhindert, dass bereits gepufferte Empfangstelegramme evtl. verloren gehen. • 0 - nicht aktiviert • 1 - aktiviert
GENERAL.CLR_MBUF	IN	Byte	0	Aktiviert das Löschen des Empfangspuffers im Anlauf der CPU. Ist dieser Parameter aktiviert, wird der Empfangspuffer beim CPU-Übergang von STOP nach RUN automatisch gelöscht. Im Empfangspuffer befinden sich dann nur Telegramme die nach dem CPU-Anlauf empfangen wurden. • 0 - nicht aktiviert • 1 - aktiviert

P3964_Config: Protokoll 3964(R) projektieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung P3964_Config (Protokollkonfiguration) können Sie Protokollparameter für 3964(R) wie Zeichenverzugszeit, Priorität und Blockcheck über ihr Programm im laufenden Betrieb ändern.

Die Konfigurationsänderungen von P3964_Config werden nicht in der CPU sondern auf dem CM gespeichert. Die in der Gerätekonfiguration gespeicherten Parameter werden werden nach Spannungswiederkehr der CPU oder des Kommunikationsmoduls wiederhergestellt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke an diesem Eingang.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BCC	IN	USInt	Byte	1	Aktiviert/deaktiviert die Verwendung des Blockchecks • 0 = ohne Blockcheck • 1 = mit Blockcheck
Priority	IN	USInt	Byte	1	Einstellung der Priorität • 0 = niedrige Priorität • 1 = hohe Priorität
CharacterDelayTime	IN	UInt	Word	16#00DC	Einstellung der Zeichenverzugszeit (abhängig von der eingestellten Datenübertragungsgeschwindigkeit) (Standardwert: 220 ms) 1 bis 65535 ms
AcknDelayTime	IN	UInt	Word	16#07D0	Einstellung der Quittungsverzugszeit (abhängig von der eingestellten Datenübertragungsgeschwindigkeit) (Standardwert: 2000 ms) 1 bis 65535 ms
BuildupAttempts	IN	USInt	Byte	16#0006	Einstellung der Anzahl Aufbauversuche (Standardwert: 6 Aufbauversuche) 1 bis 255

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RepetitionAttempts	IN	USInt	Byte	16#0006	Einstellung der Anzahl Übertragungsversuche (Standardwert: 6 Aufbauversuche) 1 bis 255
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Send_P2P: Daten senden

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Send_P2P (Punkt-zu-Punkt-Daten senden) startet die Übertragung der Daten und überträgt den Inhalt des zugewiesenen Puffers zum Kommunikationsmodul. Das Programm der CPU wird weiterhin ausgeführt, während das CM die Daten mit der Datenübertragungsrate sendet. Es darf zu jeder Zeit nur eine Sendeanweisung pro Kommunikationsmodul anstehen. Das CM gibt einen Fehler aus, wenn eine zweite Anweisung Send_P2P ausgeführt wird, während das CM bereits ein Telegramm sendet.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BUFFER	IN	Variant	Any	0	Dieser Parameter zeigt auf den Speicherbereich des Sendepuffers. **Hinweise:** • Boolsche Daten oder Boolesche Felder werden nicht unterstützt. • Liegt der Sendepuffer im optimierten Speicherbereich, dann beträgt die maximale zulässige Länge der gesendeten Daten 1024 Byte. Ausnahme: Arrays von Byte, Word oder DWord werden bis zu einer Länge von 4096 Byte unterstützt. • Ist der Sendepuffer ein String oder WString, dann wird der Inhalt des Strings ohne die aktuelle und maximale Länge übertragen. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 6519)"
LENGTH	IN	UInt	Word	0	Länge der zu übertragenden Daten in Byte. Mit LENGTH = 0 wird der im Parameter BUFFER adressierte Speicherbereich komplett übertragen. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 6519)"
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Send_P2P Mit 1 wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf 0 gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Parameter

Wenn eine Sendeanweisung in Bearbeitung ist, sind die Ausgänge DONE und ERROR im Zustand FALSE. Nach dem Ende einer Sendeanweisung wird einer der Ausgänge DONE oder ERROR für einen Zyklus auf TRUE gesetzt, um den Zustand der Sendeanweisung zu melden. Wenn ERROR im Zustand TRUE ist, kann der Fehlercode am Ausgang STATUS ausgewertet werden.

Die Anweisung gibt den Status 16#7001 aus, wenn die Kommunikationsschnittstelle die Sendedaten annimmt. Nachfolgende Ausführungen von Send_P2P geben den Wert 16#7002 aus, wenn das CM immer noch sendet. Nach dem Ende der Sendeanweisung gibt das CM den Status 16#0000 für die Sendeanweisung aus (sofern kein Fehler aufgetreten ist). Nachfolgende Ausführungen von Send_P2P mit REQ = 0 geben den Status 16#7000 (frei) aus.

Das folgende Diagramm zeigt die Beziehung zwischen den Ausgangswerten und REQ. Hierbei wird vorausgesetzt, dass die Anweisung zyklisch aufgerufen wird, um den Status des Sendevorgangs zu prüfen (dargestellt durch die STATUS-Werte).

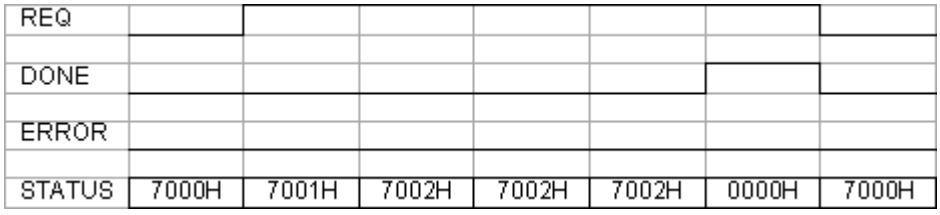

Die folgende Abbildung zeigt, wie die Parameter DONE und STATUS nur einen Zyklus lang gültig sind, wenn an der REQ-Leitung (einen Zyklus lang) ein Impuls anliegt, um die Sendeanweisung anzustoßen.

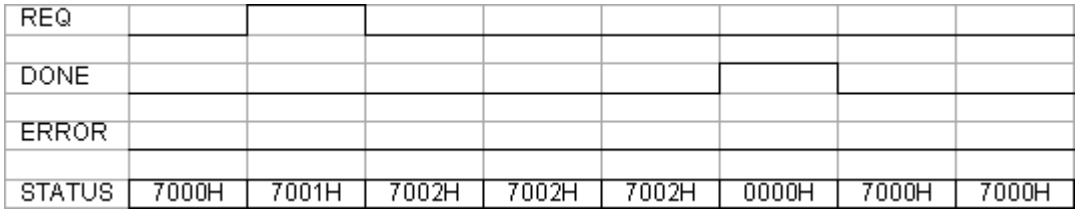

Die folgende Abbildung zeigt die Beziehung der Parameter DONE, ERROR und STATUS im Fehlerfall.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	80D1H	7000H	7000H

Die Werte DONE, ERROR und STATUS sind nur solange gültig, bis Send_P2P erneut mit dem gleichen Instanz-DB ausgeführt wird.

Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen

Interaktion der Parameter BUFFER und LENGTH für Send_P2P

Die Mindestdatengröße, die von der Anweisung Send_P2P gesendet werden kann, ist ein Byte.

Der Parameter BUFFER legt die Größe der zu sendenden Daten fest, falls beim Aufruf am Parameter LENTH eine "0" übergeben wird. Damit ist die Angabe einer Variable ausreichend.

Sie können für den Parameter BUFFER weder den Datentyp Bool noch Arrays vom Typ Bool verwenden. Wenn große Datenmengen übertragen werden, empfiehlt sich die Abbildung auf die Datentypen Array oder Struktur.

Tabelle 4-310 Parameter BUFFER

BUFFER	Beschreibung
Elementarer Datentyp	Beim Senden gilt: Der LENGTH-Wert muss die Bytezahl dieses Datentyps enthalten. Beispiel: Bei einem Word-Wert muss LENGTH zwei sein. Bei einem DWord- oder Real-Wert muss LENGTH vier sein.
Struktur	Beim optimized Speicher gilt: Die maximale erlaubte Länge des BUFFER ist 1024 Byte; ansonsten sind modulabhängig bis zu 4 kByte zulässig. Beim Senden gilt: Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge der Struktur enthalten, dann werden nur die ersten LENGTH Bytes der Struktur aus BUFFER gesendet.
Array	Beim optimized Speicher gilt: Falls der Array-Datentyp ungleich Byte, Word oder DWord ist, dann ist die maximale erlaubte Länge des Buffers 1024 Byte. Beim nicht optimized Speicher können modulabhängig bis zu 4 kByte übertragen werden, unabhängig von der Datenstruktur. Beim Senden gilt: Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge des Arrays enthalten, wobei es sich um ein Vielfaches der Bytezahl des Datenelements handeln muss. Beispiel: Der Parameter LENGTH eines Arrays vom Typ Word muss ein Vielfaches von zwei sein und bei einem Array vom Typ Real ein Vielfaches von vier. Wenn LENGTH angegeben wird, wird die Anzahl der Array-Elemente übertragen. Wenn beispielsweise BUFFER ein Array mit 15 DWord -Elementen (insgesamt 60 Bytes) enthält und Sie geben LENGTH = 20 an, werden die ersten fünf DWord -Elemente aus dem Array übertragen.
String	Der Parameter LENGTH enthält die Anzahl der zu sendenden Zeichen. Nur die Zeichen des String werden übertragen. Die Bytes mit der maximalen und der tatsächlichen Länge des String werden nicht gesendet.

Tabelle 4-311 Parameter LENGTH

LENGTH	Beschreibung
= 0	Der komplette Inhalt des über BUFFER spezifizierten Speicherbereichs wird übertragen. Falls BUFFER auf einen String zeigt, wird der komplette String-Inhalt übertragen, ohne die Bytes mit der maximalen und tatsächlichen Länge.
> 0	Es wird der Inhalt bis zur parametrierten Länge des über BUFFER spezifizierten Speicherbereichs übertragen.

Receive_P2P: Daten empfangen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Receive_P2P (Daten über eine Punkt-zu-Punkt-Kommunikation empfangen) prüft die Telegramme, die im CM empfangen wurden. Wenn ein Telegramm verfügbar ist, wird es vom CM zur CPU übertragen. Ein Empfangsfehler wird am Parameter STATUS angezeigt.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BUFFER	IN	Variant	Any	0	Dieser Parameter zeigt auf die Anfangsadresse des Empfangspuffers. Dieser Puffer muss groß genug sein, um die maximale Telegrammlänge zu empfangen. Hinweis: • Boolesche Daten oder Boolesche Felder werden nicht unterstützt. • Liegt der Empfangspuffer im optimierten Speicherbereich, dann beträgt die maximal zulässige Länge der empfangenen Daten 1024 Byte. Ausnahme: Arrays von Byte, Word oder DWord werden bis zu einer Länge von 4096 Byte unterstützt. • Ist der Empfangspuffer ein String oder WString, dann werden die empfangenen Daten in den Inhalt des Strings geschrieben und die aktuelle Länge des Strings wird entsprechend gesetzt. Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 6519)"
LENGTH	OUT	UInt	Word	0	Länge des empfangenen Telegramms in Byte Weitere Informationen unter "Verwendung der Parameter BUFFER und LENGTH bei Kommunikationsoperationen (Seite 6519)"
NDR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, wenn neue Daten bereit sind und die Anweisung fehlerfrei beendet wurde.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Falls ERROR im Zustand TRUE ist, kann der Fehlercode am Ausgang STATUS ausgewertet werden. Der STATUS-Wert liefert den Grund für die Beendigung der Empfangsoperation im CM.
Dies ist typischerweise ein positiver Wert, der angibt, dass die Empfangsoperation erfolgreich war und welches Telegrammendekriterium erkannt wurde.
Ist der STATUS-Wert negativ (das höchstwertige Bit des Hexadezimalwerts ist gesetzt), wurde die Empfangsoperation wegen einer Fehlerbedingung wie Paritäts-, Framing- oder Überlauffehler beendet.

Jedes Kommunikationsmodul kann eine modulspezifische Anzahl von Telegrammen puffern. Sind mehrere Telegramme im CM verfügbar, gibt die Anweisung Receive_P2P das älteste verfügbare Telegramm aus (FIFO).

Receive_Reset: Empfangspuffer löschen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Receive_Reset (Empfänger zurücksetzen) löscht den Empfangspuffer im CM.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Einen Zyklus lang TRUE bedeutet, dass die letzte Anforderung fehlerfrei ausgeführt wurde.
ERROR	OUT	Bool		FALSE	TRUE bedeutet, dass die letzte Anforderung mit Fehlern ausgeführt wurde. Ist dieser Ausgang TRUE, enthält Ausgang STATUS die zugehörigen Fehlercodes.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Signal_Get: Status lesen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Signal_Get (RS232-Signale abrufen) liest die aktuellen Zustände der RS232-Begeitsignale und zeigt sie an den entsprechenden Ausgängen der Anweisung an.

Hinweis

Einschränkung

- Diese Anweisung ist nur bei den CMs RS232 BA und RS232 HF anwendbar.
- Falls bei der Betriebsart RS232C eingestellt ist, ist diese Anweisung auch beim CM PtP (ET200SP) anwendbar.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Übertragung der Daten zum CM bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
NDR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, wenn die RS232-Begleitsignale gelesen wurden und die Anweisung fehlerfrei beendet wurde.
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))
DTR	OUT	Bool		FALSE	Datengerät bereit, Modul bereit (Ausgang)
DSR	OUT	Bool		FALSE	Datengerät bereit, Kommunikationsteilnehmer bereit (Eingang)
RTS	OUT	Bool		FALSE	Sendeanforderung, Modul sendebereit (Ausgang)
CTS	OUT	Bool		FALSE	Sendebereitschaft, Kommunikationsteilnehmer kann Daten empfangen (Eingang)
DCD	OUT	Bool		FALSE	Datenträgersignal erkannt, Signalpegel empfangen
RING	OUT	Bool		FALSE	Rufanzeige, Meldung eines eingehenden Rufs

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Signal_Set: Begleitsignale setzen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Signal_Set (RS232-Signale setzen) erlaubt es der RS232-Kommunikationssignale zu setzen.

Hinweis

Einschränkungen

- Diese Anweisung ist nur bei den CMs RS232 BA und RS232 HF anwendbar.
- Falls bei der Betriebsart RS232C eingestellt ist, ist diese Anweisung auch beim CM PtP (ET200SP) anwendbar.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
SIGNAL	IN	Byte		0	Auswahl des zu setzenden Signals (mehrere zulässig): • 01H = RTS • 02H = DTR • 04H = DSR (gilt nur für Schnittstellentyp DCE)
RTS	IN	Bool		FALSE	Sendeanforderung, Modul sendebereit Diesen Wert am Ausgang setzen (TRUE oder FALSE), Standardwert: FALSE
DTR	IN	Bool		FALSE	Datenendgerät bereit, Modul bereit Diesen Wert am Ausgang setzen (TRUE oder FALSE), Standardwert: FALSE
DSR	IN	Bool		FALSE	Datenendgerät bereit (gilt nur für Schnittstellentyp DCE), nicht verwendet.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
DONE	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Eine Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler ausgeführt wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Get_Features: Erweiterte Funktionen holen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Get_Features (Erweiterte Funktionen holen) können, sofern vom Modul unterstützt, Informationen über die Fähigkeit des Moduls zur CRC-Unterstützung und zur Generierung von Diagnosemeldungen abgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	PORT	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
NDR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, wenn neue Daten bereit sind und die Anweisung fehlerfrei beendet wurde
MODBUS_CRC	OUT	Bool		FALSE	Modbus CRC-Unterstützung
DIAG_ALARM	OUT	Bool		FALSE	Generierung von Diagnosemeldungen
SUPPLY_VOLT	OUT	Bool		FALSE	Diagnose für fehlende Versorgungsspannung L+ verfügbar
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Set_Features: Erweiterte Funktionen setzen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Set_Features (Erweiterte Funktionen auswählen) können, sofern vom Modul unterstützt, die CRC-Unterstützung und die Generierung von Diagnosemeldungen aktiviert werden.

Parameter

Parameter	Deklaration	Datentyp		Vorbelegung	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Die Anweisung zum Setzen erweiterter Funktionen wird bei steigender Flanke an diesem Eingang gestartet.
PORT	IN	PORT (UInt)	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
EN_MODBUS_CRC	IN	Bool		FALSE	Modbus CRC-Unterstützung aktivieren
EN_DIAG_ALARM	IN	Bool		FALSE	Generierung von Diagnosemeldungen aktivieren
EN_SUPPLY_VOLT	IN	Bool		FALSE	Diagnose für fehlende Versorgungsspannung L+ einschalten
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt.
DONE	OUT	Bool		FALSE	Eine Ausführung lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde
ERROR	OUT	Bool		FALSE	Einen Zyklus lang TRUE, nachdem die Anweisung mit Fehler beendet wurde
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6529))

Weitere Informationen zu den allgemeinen Parametern finden Sie unter "Gemeinsame Parameter für Freeport-Operationen (Seite 6503)".

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/ maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüberschreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichenverzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endesequenz"	-
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertragung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8088	Längenangabe größer als der im Empfangspuffer eingestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehlercode auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wählen Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		

Fehlercode	Beschreibung	Abhilfe
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).

Fehlercode	Beschreibung	Abhilfe
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.

Fehlercode	Beschreibung	Abhilfe
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)

Fehlercode	Beschreibung	Abhilfe
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.

Fehlercode	Beschreibung	Abhilfe
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

Fehlercode	Beschreibung	Abhilfe
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions

Fehlercode	Beschreibung	Abhilfe
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode	Beschreibung	Abhilfe
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)

Fehlercode	Beschreibung	Abhilfe
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG

Fehlercode	Beschreibung	Abhilfe
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.

Fehlercode	Beschreibung	Abhilfe
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.

Fehlercode	Beschreibung	Abhilfe
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.
16#8828 [1)]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1)]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1)]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1)]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1)]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8428 [1)]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG

Fehlercode	Beschreibung	Abhilfe
16#8455 [1]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.

Fehlercode	Beschreibung	Abhilfe
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

USS

Überblick über die USS-Kommunikation

USS-Kommunikation

Die USS-Anweisungen steuern den Betrieb von Antrieben, die das Protokoll der universellen seriellen Schnittstelle (USS) unterstützen. Mit den PtP-Kommunikationsmodulen können Sie über RS485-Verbindungen und den USS-Anweisungen mit mehreren Antrieben kommunizieren. Jeder RS485-Port kann bis zu 16 Antriebe betreiben.

Das USS-Protokoll nutzt ein Master/Slave-Netzwerk für die Kommunikation über einen seriellen Bus. Der Master verwendet einen Adressparameter, um Daten an einen ausgewählten Slave zu senden. Ein Slave selbst kann niemals senden, ohne dafür zuvor eine Anforderung erhalten zu haben. Die Kommunikation zwischen den einzelnen Slaves ist nicht möglich. Die USS-Kommunikation funktioniert im Halbduplex-Betrieb. Die folgende Abbildung zeigt ein Netzwerkdiagramm für eine Beispielanwendung mit 16 Antrieben.

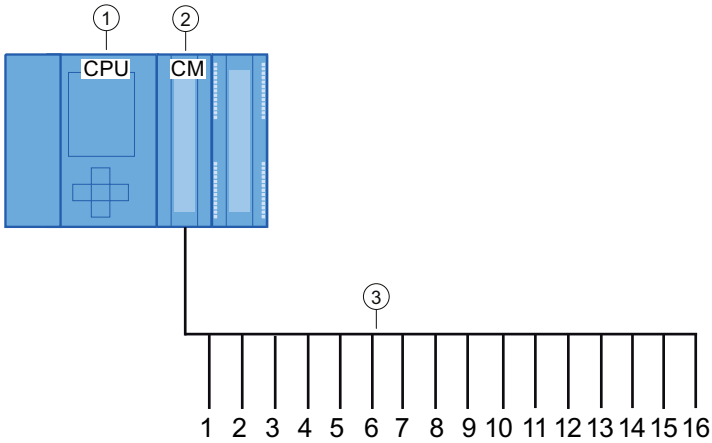

① CPU
② CM
③ USS-Antriebe an einem USS-Netz

Bild 4-31 Verschaltungsbeispiel mit S7-1500 Kommunikationsmodul

Hinweis

Kommunikation mit einem Antrieb über RS232

Prinzipiell können auch CM PtP RS232 BA und CM PtP RS232 HF zur Kommunikation mit einem Antrieb eingesetzt werden. An einem RS232-Port kann jedoch nur **ein** Antrieb angeschlossen werden.

Kommunikation mit einem Antrieb über RS422

Prinzipiell kann auch die RS422-Schnittstelle der CM PtP RS422/485 BA und CM PtP RS422/485 HF zur Kommunikation mit einem Antrieb eingesetzt werden. An einem RS422-Port kann jedoch nur **ein** Antrieb angeschlossen werden.

USS-Anweisungen in Ihrem Programm

- USS_Port_Scan: Die Anweisung USS_Port_Scan ermöglicht Ihnen die Kommunikation über ein Kommunikationsmodul mit bis zu 16 Antrieben über ein USS-Netzwerk (muss zyklisch aufgerufen werden).
  Es ist nur eine Anweisung USS_Port_Scan pro PtP-Kommunikationsport im Programm vorhanden, die die Übertragung zu allen Antrieben steuert.
- USS_Drive_Control: Die Anweisung USS_Drive_Control ermöglicht Ihnen, die Sendedaten von USS_Port_Scan für einen Antrieb aufzubereiten und dessen Empfangsdaten anzuzeigen.
  USS_Drive_Control konfiguriert die zu sendenden Daten und wertet Daten aus, die in einer vorherigen Anforderung durch USS_Port_Scan empfangen wurden.
- USS_Read_Param: Die Anweisung USS_Read_Param ermöglicht Ihnen, Parameter aus einem Antrieb auszulesen.
- USS_Write_Param: Die Anweisung USS_Write_Param ermöglicht Ihnen, Parameter in einem Antrieb zu ändern.

Voraussetzungen für den Einsatz des USS-Protokolls

Die vier USS-Anweisungen verwenden 2 FBs und 2 FCs zur Unterstützung des USS-Protokolls. Für jedes USS-Netzwerk wird ein Instanz-Datenbaustein für USS_Port_Scan und ein Instanz-Datenbaustein gemeinsam für alle Aufrufe von USS_Drive_Conrol verwendet.

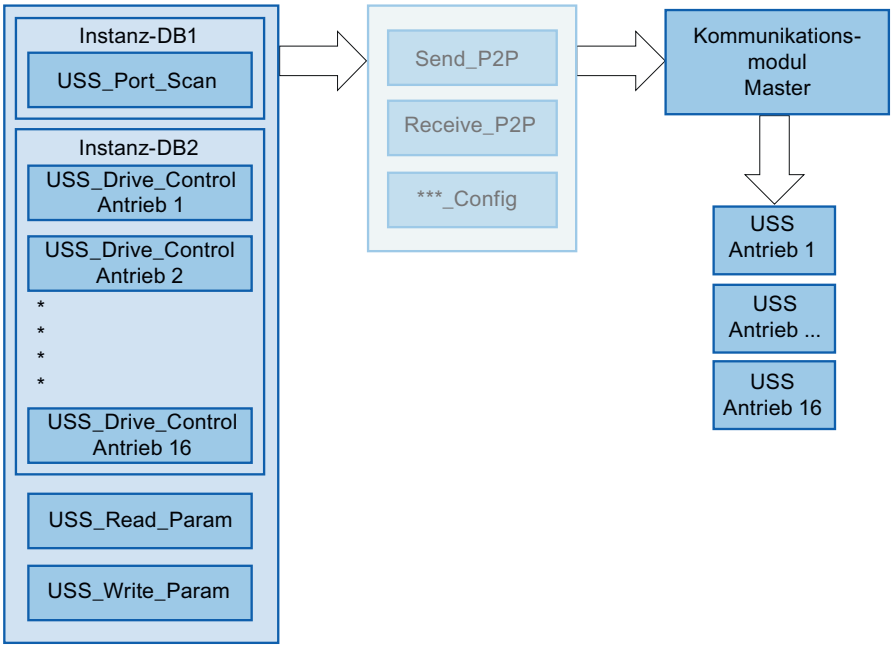

Bild 4-32 Programmablauf - USS

Alle Antriebe (max. 16), die an einen RS485-Port angeschlossen sind, sind Teil desselben USS-Netzwerks. Alle Antriebe, die an einen anderen RS485-Port angeschlossen sind, sind Teil eines anderen USS-Netzwerks. Jedes USS-Netzwerk wird mithilfe eines eindeutigen Instanz-Datenbausteins für alle Anweisungen USS_Drive_Control und eines weiteren Instanz-Datenbausteins für die Anweisung USS_Port_Scan verwaltet. Alle Anweisungen, die zu einem USS-Netzwerk gehören, müssen den Instanz-Datenbaustein für USS_Drive_Conrol gemeinsam nutzen. Die Anweisungen USS_Port_Scan, USS_Read_Param und

USS_Write_Param haben dafür den Parameter USS_DB, der mit dem (statischen) Parameter USS_DB des Instanz-DBs der Anweisung USS_Drive_Control verschaltet werden muss.

- Die Anweisungen USS_Drive_Control und USS_Port_Scan sind Funktionsbausteine (FB). Falls Sie die Anweisung USS_Drive_Control oder USS_Port_Scan in den Programmiereditor einfügen, werden Sie im Dialog "Aufrufoptionen" aufgefordert, einen DB für diesen FB zuzuweisen. Wenn es sich um die erste Anweisung USS_Drive_Control in diesem Programm für dieses USS-Netzwerk handelt, können Sie die DB-Standardzuweisung übernehmen (oder ggf. den Namen ändern), und der neue DB wird für Sie erstellt. Wenn es sich jedoch nicht um die erste Anweisung USS_Drive_Control für diesen Antrieb handelt, müssen Sie im Dialog "Aufrufoptionen" in der Klappliste den DB auswählen, der diesem USS-Netzwerk bereits zuvor zugewiesen wurde.
- Bei den Anweisungen USS_Write_Param und USS_Read_Param handelt es sich um Funktionen (FCs). Wenn Sie diese FCs im Editor einfügen, wird kein DB zugewiesen. Falls Sie diese FCs oder die Anweisung USS_Port_Scan im Editor einfügen, müssen Sie dem Eingang USS_DB dieser Anweisungen den Parameter USS_DB des jeweiligen Instanz-DB von USS_Drive_Control zuweisen. Doppelklicken Sie auf das Parameterfeld und klicken Sie dann auf das Symbol, um die verfügbaren DBs anzuzeigen. Geben Sie einen Punkt "." ein und wählen Sie dann aus der Klappliste den Parameter USS_DB aus.
- Die Funktion USS_Port_Scan steuert die Kommunikation zwischen der CPU und den Antrieben über den Punkt-zu-Punkt-RS485-Kommunikationsport. Bei jedem Aufruf dieser Funktion wird eine Kommunikation mit einem Antrieb bearbeitet. Ihr Programm muss diese Funktion schnell genug aufrufen, sodass die Antriebe keine Zeitüberschreitung melden. Um ein konstantes Zeitverhalten des Telegrammversands zu garantieren, sollten Sie diese Anweisung in einem Weckalarm-OB aufrufen.
- Die Anweisung USS_Drive_Control gibt Ihrem Programm Zugriff auf einen angegebenen Antrieb im USS-Netzwerk. Seine Ein- und Ausgänge entsprechen den Zuständen und den Bedienfunktionen des Antriebs. Sind 16 Antriebe im Netzwerk vorhanden, so muss USS_Drive_Control in Ihrem Programm 16-mal aufgerufen werden, also jeweils einmal für jeden Antrieb.
  Sie sollten die Anweisung USS_Drive_Control nur aus einem zyklischen OB aufrufen.
- Mit den Funktionen USS_Read_Param und USS_Write_Param werden die Betriebsparameter des Antriebs gelesen und geschrieben. Diese Parameter steuern die interne Funktionsweise des Antriebs. Eine Definition dieser Parameter finden Sie im Handbuch des Antriebs. Ihr Programm kann eine beliebige Anzahl dieser Funktionen enthalten, es kann jedoch immer nur eine Lese- oder Schreibanforderung für einen Antrieb aktiv sein. Sie dürfen die Funktionen USS_Read_Param und USS_Write_Param nur aus dem Zyklus-OB eines Hauptprogramms aufrufen.

ACHTUNG

Aufrufe der USS-Anweisungen

Rufen Sie USS_Drive_Control, USS_Read_Param und USS_Write_Param nur aus einem Zyklus-OB des Hauptprogramms auf. Die Anweisung USS_Port_Scan kann aus einem beliebigen OB aufgerufen werden, üblicherweise wird sie aus einem Weckalarm-OB aufgerufen.

Verwenden Sie die Anweisungen USS_Drive_Control, USS_Read_Param und USS_Write_Param nicht in einem OB mit einer höheren Priorität als die entsprechende Anweisung USS_Port_Scan. Fügen Sie beispielsweise nicht USS_Port_Scan in das Hauptprogramm und USS_Read_Param in einen Weckalarm-OB ein. Wird die Ausführung des USS_Port_Scan durch eine andere Anweisung unterbrochen, kann es zu unerwarteten Fehlern kommen.

Hinweis

Parameterkennungswert

Für die Antriebe müssen Sie die Verwendung von 4 PKW-Wörtern einrichten.

Zeit für die Kommunikation mit dem Antrieb berechnen

Die Kommunikation mit dem Antrieb läuft asynchron zum Zyklus der CPU ab. Die CPU durchläuft üblicherweise mehrere Zyklen, bevor die Kommunikation mit einem Antrieb beendet ist.

Damit die beim Antrieb eingestellte Zeitüberwachung nicht anspricht, müssen die Sendetelegramme zum Antrieb innerhalb dieser Zeitüberwachung gesendet werden. Dabei muss die Zahl der Wiederholversuche mit eingerechnet werden, falls zur Fertigstellung der Transaktion aufgrund von Kommunikationsfehlern mehrere Versuche notwendig sind. Standardmäßig werden für das USS Protokoll bei jeder Transaktion bis zu 2 Wiederholungen durchgeführt.
Der maximale Zeitabstand zwischen zwei Sendetelegrammen berechnet sich wie folgt:

N * (5 * Zykluszeit + Telegrammlaufzeit + Timeout des Empfangstelegramms max.) * (Anzahl Sendeversuche)

N	Anzahl der Antriebe in diesem Netzwerk
Faktor 5	Für das Senden und Empfangen von Telegrammen werden typischerweise 5 Zyklen benötigt.
Zykluszeit	Max. Zykluszeit des Weckalarm-OBs, in der die Anweisung USS_Port_Scan aufgerufen wird.
Telegrammlaufzeit	Telegrammlaufzeit = (Anzahl Zeichen pro Telegramm) * (11 Bit pro Zeichen) / (Datenübertragungsrate in Bit/s)
Anzahl Sendeversuche	Anzahl Wiederholversuche + 1

Timeout des Empfangstelegramms	RCVTIME (falls keine Antwort vom Antrieb empfangen wird)
Timeout des Empfangstelegramms max.	RCVTIME + MSGTIME (falls kurz vor Ablauf von RCVTIME eine unvollständige Antwort empfangen wird und die Überwachung von MSGTIME abläuft **oder** falls zum Ablauf von RCVTIME noch eine Antwort in Bearbeitung ist, wird der Timeout um die MSGTIME verlängert)

Für den "Timeout des Empfangstelegramms" gelten folgende Zeiten (ms):

Bit/s	115200	57600	38400	19200	9600	4800	2400	1200
Receive_Conditions.END.RCVTIME	25	29	33	56	72	100	100	100
Receive_Conditions.END.MSGTIME	25	29	33	56	72	124	240	460

Timeout des Empfangstelegramms max. = (Receive_Conditions.END.RCVTIME (0,072 s) + Receive_Conditions.END.MSGTIME (0,072 s))

Beispiel:

5 Antriebe
Datenübertragungsrate = 9600 Bit/s
28 Zeichen pro Telegramm
Zykluszeit = 0.020 s
Anzahl Wiederholversuche = 2

Zeitabstand = 5 * (5 * 0,02 s + 1 * 28 * 11 / 9600 s + 0,072 s + 0,072 s) * 3 = 3 s

Die Zeitüberwachung des Antriebs muss in diesem Fall auf ca. 3 Sekunden eingestellt werden.

USS_Port_Scan / USS_Port_Scan_31: Kommunikation über USS-Netzwerk

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Port_Scan_31

Der Einsatz der Anweisung USS_Port_Scan_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Port_Scan bearbeitet die Kommunikation über ein USS-Netzwerk für max. 16 Antriebe.

Die Anweisung USS_Port_Scan_31 bearbeitet die Kommunikation über ein USS-Netzwerk für max. 31 Antriebe.

STEP 7 erstellt automatisch den Instanz-DB, wenn Sie die Anweisung einfügen.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
PORT	IN	Port	Word	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BAUD	IN	DInt		9600	Datenübertragungsrate für die USS-Kommunikation Zulässig sind: • 1200 Bit/s • 2400 Bit/s • 4800 Bit/s • 9600 Bit/s • 19200 Bit/s • 38400 Bit/s • 57600 Bit/s • 115200 Bit/s
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
COM_RST	INOUT	---	Bool	FALSE	Initialisierung der Anweisung USS_Port_Scan / USS_Port_Scan_31 Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
ERROR	OUT	Bool		FALSE	Falls TRUE, weist dieser Ausgang darauf hin, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 6563)).

Es ist nur eine Anweisung USS_Port_Scan / USS_Port_Scan_31 pro PtP-Kommunikationsport im Programm vorhanden und jeder Aufruf dieser Anweisung steuert eine Übertragung zu oder von allen Antrieben dieses Netzwerks. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den gleichen Instanz-DB nutzen.

Ihr Programm muss die Anweisung USS_Port_Scan / USS_Port_Scan_31 so oft ausführen, dass kein Timeout im Antrieb auftritt (siehe Voraussetzungen für den Einsatz des USS-Protokolls (Seite 6545) Abschnitt "Zeit für die Kommunikation mit dem Antrieb berechnen").

Die Anweisung USS_Port_Scan / USS_Port_Scan_31 wird üblicherweise aus einem Weckalarm-OB aufgerufen, um Antriebs-Timeouts zu verhindern und die letzten USS-Datenaktualisierungen für Aufrufe von USS_Drive_Control / USS_Drive_Control_31 verfügbar zu haben.

USS_Port_Scan / USS_Port_Scan_31 Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von USS_Port_Scan / USS_Port_Scan_31, die in Ihrem Programm verwendet werden können.

Tabelle 4-312 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
MODE	USInt	4	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master; CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave; CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1)]
LINE_PRE	USInt	2	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) : Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V : Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.
BRK_DET	USInt	0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
RETRIES_MAX	SInt/Byte	2	Zahl der Wiederholversuche bei Auftreten von Kommunikationsfehlern. Mit diesem Parameter können Sie die Zahl der Wiederholversuche für das Senden eines Anforderungstelegramms setzen, falls das Antworttelegramm nicht innerhalb der eingestellten Zeit empfangen wird.

Variable	Datentyp	Standard	Beschreibung
EN_DI-AG_ALARM	Bool	0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
EN_SUP-PLY_VOLT	Bool	0	Diagnose für fehlende Versorgungsspannung L+ einschalten • 0 - nicht aktiviert • 1 - aktiviert

[1] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Die Version 2.5 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

Anweisungsversionen

USS_Port_Scan:

Die Version 2.5 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

USS_Port_Scan_31:

Die Version 1.2 ist funktional identisch zur Version 1.1 und wurde aufgrund interner Maßnahmen erhöht.

USS_Drive_Control / USS_Drive_Control_31: Daten für den Antrieb vorbereiten und anzeigen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Drive_Control_31

Der Einsatz der Anweisung USS_Drive_Control_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Drive_Control bereitet Sendedaten für maximal 16 Antriebe vor und wertet die Antwortdaten der Antriebe aus.

Die Anweisung USS_Drive_Control_31 bereitet Sendedaten für maximal 31 Antriebe vor und wertet die Antwortdaten der Antriebe aus.

Es muss für jeden Antrieb eine eigene Instanz der Anweisung verwendet werden, und es müssen alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, den gleichen Instanz-Datenbaustein verwenden. Sie müssen den DB-Namen eingeben, wenn Sie die erste Anweisung USS_Drive_Control / USS_Drive_Control_31

einfügen. Dann verweisen Sie auf diesen DB, der beim Einfügen der ersten Anweisung angelegt wurde.

STEP 7 erstellt automatisch den DB, wenn Sie die Anweisung einfügen.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RUN	IN	Bool		FALSE	Startbit des Antriebs: Ist dieser Parameter TRUE, so ermöglicht dieser Eingang den Betrieb des Antriebs mit der voreingestellten Drehzahl. Wenn RUN im Betrieb des Antriebs nach FALSE wechselt, läuft der Motor bis zum Stillstand aus. Dieses Verhalten unterscheidet sich von der Abschaltung der Spannungsversorgung (OFF2) und vom Bremsen des Motors (OFF3).
OFF2	IN	Bool		FALSE	Bit "Zum Stillstand auslaufen": Ist dieser Parameter FALSE, so veranlasst dieses Bit das Auslaufen des Antriebs, ohne zu bremsen.
OFF3	IN	Bool		FALSE	Schnelles Stoppbit: Ist dieser Parameter FALSE, so verursacht dieses Bit einen schnellen Halt durch Abbremsen des Antriebs.
F_ACK	IN	Bool		FALSE	Fehlerquittierungsbit: Mit diesem Bit wird das Fehlerbit eines Antriebs zurückgesetzt. Das Bit wird nach dem Löschen des Fehlers gesetzt und der Antrieb erkennt damit, dass der vorherige Fehler nicht mehr gemeldet werden muss.
DIR	IN	Bool		FALSE	Richtungssteuerung des Antriebs: Dieses Bit wird gesetzt, wenn der Antrieb in Vorwärtsrichtung laufen soll (wenn SPEED_SP positiv ist; siehe Tabelle "Interaktion der Parameter SPEED_SP und DIR").
DRIVE	IN	USInt	Byte	1	Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PZD_LEN	IN	USInt	Byte	2	Wortlänge: Dies ist die Anzahl der PZD-Datenwörter. Gültige Werte sind 2, 4, 6 oder 8 Wörter.
SPEED_SP	IN	Real		0.0	Drehzahlsollwert: Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Frequenz. Ein positiver Wert bedeutet, dass der Antrieb vorwärts läuft (wenn DIR wahr ist). Gültig ist der Bereich von 200,00 bis -200,00.
CTRL3	IN	Word		0	Steuerwort 3: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL4	IN	Word		0	Steuerwort 4: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL5	IN	Word		0	Steuerwort 5: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
CTRL6	IN	Word		0	Steuerwort 6: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL7	IN	Word		0	Steuerwort 7: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
CTRL8	IN	Word		0	Steuerwort 8: Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Sie müssen dies im Antrieb konfigurieren (optionaler Parameter).
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung USS_Drive_Control / USS_Drive_Control_31 Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
NDR	OUT	Bool		FALSE	Neue Daten bereit: Ist dieser Parameter TRUE, so meldet das Bit, dass am Ausgang Daten einer neuen Kommunikationsanforderung bereitstehen.
ERROR	OUT	Bool		FALSE	Fehler aufgetreten: Falls TRUE, weist dies darauf hin, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 6563)).
RUN_EN	OUT	Bool		FALSE	Betrieb freigegeben: Dieses Bit meldet, ob der Antrieb läuft.
D_DIR	OUT	Bool		FALSE	Antriebsrichtung: Dieses Bit meldet, ob der Antrieb vorwärts läuft. • FALSE – vorwärts • TRUE – rückwärts
INHIBIT	OUT	Bool		FALSE	Antrieb gesperrt: Dieses Bit meldet den Zustand des Sperrbits für den Antrieb. • FALSE – nicht gesperrt • TRUE – gesperrt
FAULT	OUT	Bool		FALSE	Antriebsfehler: Dieses Bit meldet, dass im Antrieb ein Fehler aufgetreten ist. Sie müssen die Störung beheben und Bit F_ACK setzen, um dieses Bit zu löschen.
SPEED	OUT	Real		0.0	Istwert Antriebsdrehzahl (skalierter Wert von STATUS 2 des Antriebs): Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Drehzahl.
STATUS1	OUT	Word		0	STATUS 1 des Antriebs Dieser Wert enthält feste Zustandsbits eines Antriebs.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
STATUS3	OUT	Word		0	STATUS 3 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS4	OUT	Word		0	STATUS 4 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS5	OUT	Word		0	STATUS 5 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS6	OUT	Word		0	STATUS 6 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS7	OUT	Word		0	STATUS 7 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.
STATUS8	OUT	Word		0	STATUS 8 des Antriebs Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.

Wenn die erste Ausführung von USS_Drive_Control / USS_Drive_Control_31 erfolgt, wird der von der USS-Adresse (Parameter DRIVE) angegebene Antrieb im Instanz-DB initialisiert. Nach der Initialisierung können nachfolgende Anweisungen USS_Port_Scan / USS_Port_Scan_31 die Kommunikation mit dem Antrieb an dieser Antriebsnummer beginnen.

Wenn Sie die Antriebsnummer ändern, muss die CPU zunächst in STOP und dann wieder in RUN versetzt werden, damit der Instanz-DB initialisiert wird. Die Eingangsparameter werden im USS-Sendepuffer konfiguriert und die Ausgänge werden, sofern vorhanden, aus einem "vorherigen" gültigen Antwortpuffer gelesen. USS_Drive_Control / USS_Drive_Control_31 konfiguriert nur die zu sendenden Daten und wertet Daten aus, die in einer vorherigen Anforderung empfangen wurden.

Sie können die Drehrichtung des Antriebs entweder über den Eingang D_IR (Bool) oder über das Vorzeichen (positiv oder negativ) am Eingang SPEED_SP (Real) steuern. Die folgende Tabelle erläutert, wie diese Eingänge zusammen funktionieren, um die Drehrichtung des Antriebs zu bestimmen, vorausgesetzt der Motor dreht vorwärts.

Tabelle 4-313 Interaktion der Parameter SPEED_SP und DIR

SPEED_SP	DIR	Drehrichtung des Antriebs
Wert > 0	0	Rückwärts
Wert > 0	1	Vorwärts
Wert < 0	0	Vorwärts
Wert < 0	1	Rückwärts

USS_Drive_Control / USS_Drive_Control_31 Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von USS_Drive_Control / USS_Drive_Control_31, die in Ihrem Programm verwendet werden können.

Tabelle 4-314 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
USS_DB. w_USSExtendedError	Word	16#0	USS Drive Extended Error Code - Antriebsspezifischer Wert Die Bedeutung der Fehlermeldung ist abhängig davon, welche Anweisung **zuvor** einen Fehler (ERROR = TRUE) gemeldet hat. Folgende Fälle werden unterschieden: • USS_Write_Param / USS_Write_Param_31: Die Bedeutung des Fehlercodes finden Sie in der Beschreibung ihres Antriebs. • USS_Read_Param / USS_Read_Param_31: Die Bedeutung des Fehlercodes finden Sie in der Beschreibung ihres Antriebs. • USS_Port_Scan / USS_Port_Scan_31: Nummer des von der Fehlermeldung betroffenen Antriebs.

Anweisungsversionen

USS_Drive_Control:

Die Version 2.0 ist funktional identisch zur Version 1.2 und wurde aufgrund interner Maßnahmen erhöht.

USS_Drive_Control_31:

Die Version 2.0 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

USS_Read_Param / USS_Read_Param_31: Daten aus dem Antrieb lesen

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Read_Param_31

Der Einsatz der Anweisung USS_Read_Param_31 ist nur auf einer S7-1500 CPU möglich.

Beschreibung

Die Anweisung USS_Read_Param liest einen Parameter aus einem von maximal 16 Antrieben.

Die Anweisung USS_Read_Param_31 liest einen Parameter aus einem von maximal 31 Antrieben.

Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den Instanzdatenbaustein der Anweisung USS_Drive_Control / USS_Drive_Control_31 nutzen. USS_Read_Param / USS_Read_Param_31 muss aus einem Zyklus-OB des Hauptprogramms aufgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		–	Mit einer positiven Flanke an REQ wird eine neue Leseanforderung gestellt.
DRIVE	IN	USInt	Byte	–	Adresse des Antriebs: DRIVE ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PARAM	IN	UInt		–	Parameternummer: PARAM gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich für diesen Parameter liegt zwischen 0 und 2047. Bei einigen Antrieben kann über das höchstwertige Byte des Parameters INDEX auf Parameterwerte größer als 2047 zugreifen werden. Weitere Informationen für den Zugriff auf einen erweiterten Bereich finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	IN	UInt		–	Parameterindex: INDEX gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höchstwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
DONE[1]	OUT	Bool		FALSE	Ist dieser Parameter TRUE, so steht am Ausgang VALUE der zuvor angeforderte Wert des Leseparameters an. Dieses Bit wird gesetzt, wenn die Anweisung USS_Drive_Control / USS_Drive_Control_31 die Leseantwort des Antriebs erkennt. Dieses Bit wird beim nächsten Aufruf von USS_Read_Param / USS_Read_Param_31 zurückgesetzt.
ERROR	OUT	Bool		FALSE	ERROR = TRUE: Es ist ein Fehler aufgetreten und der Ausgang STATUS ist gültig. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
STATUS	OUT	UInt		0	Fehlercode (siehe Fehlermeldungen (Seite 6563)).
VALUE	OUT	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Dies ist der Wert des Parameters, der gelesen wurde und er ist nur gültig, wenn das Bit DONE wahr ist.

[1] Das Bit DONE weist darauf hin, dass gültige Daten aus dem referenzierten Motorantrieb ausgelesen und an die CPU geliefert wurden. Es weist nicht darauf hin, dass die Anweisung in der Lage ist, sofort einen weiteren Parameter auszulesen. Eine leere Leseanforderung muss an den Motorantrieb gesendet und auch von der Anweisung quittiert werden, bevor der Parameterkanal zur Verwendung durch den jeweiligen Antrieb frei wird. Der sofortige Aufruf von USS_Read_Param / USS_Read_Param_31 oder USS_Write_Param / USS_Write_Param_31 für den spezifischen Motorantrieb führt zu dem Fehler 16#818A.

Anweisungsversionen

USS_Read_Param:

Die Version 1.5 ist funktional identisch zur Version 1.4 und wurde aufgrund interner Maßnahmen erhöht.

USS_Read_Param_31:

Die Version 1.1 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

USS_Write_Param / USS_Write_Param_31: Daten im Antrieb ändern

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Hinweis

Einsatz der Anweisung USS_Write_Param_31

Der Einsatz der Anweisung USS_Write_Param_31 ist nur auf einer S7-1500 CPU möglich.

Hinweis

Für EEPROM-Schreibanweisungen (EEPROM in einem USS-Antrieb) gilt:

Halten Sie die Anzahl der EEPROM-Schreiboperationen möglichst gering, um eine möglichst lange Lebensdauer des EEPROM zu gewährleisten.

Beschreibung

Die Anweisung USS_Write_Param ändert einen Parameter in einem von maximal 16 Antrieben.

Die Anweisung USS_Write_Param_31 ändert einen Parameter in einem von maximal 31 Antrieben.

Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den Instanzdatenbaustein des USS_Drive_Control / USS_Drive_Control_31 nutzen.

USS_Write_Param / USS_Write_Param_31 muss aus dem Zyklus-OB eines Hauptprogramms aufgerufen werden.

Parameter

Tabelle 4-315 Datentypen für die Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Mit einer positiven Flanke an REQ wird eine neue Schreibanforderung gestellt.
DRIVE	IN	USInt	Byte	–	Adresse des Antriebs: DRIVE ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.
PARAM	IN	UInt		–	Parameternummer: PARAM gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich für diesen Parameter liegt zwischen 0 und 2047. Bei einigen Antrieben kann über das höchstwertige Byte des Parameters INDEX auf Parameterwerte größer als 2047 zugreifen. Weitere Informationen für den Zugriff auf einen erweiterten Bereich finden Sie im Handbuch zu Ihrem Antrieb.
INDEX	IN	UInt		–	Parameterindex: INDEX gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höchstwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.
EEPROM	IN	Bool		–	Im EEPROM des Antriebs speichern: Wenn TRUE, wird die Transaktion eines Parameters zum Schreiben in den Antrieb im EEPROM des Antriebs gespeichert. Wenn FALSE, so wird der geschriebene Wert nur temporär gespeichert und geht beim nächsten Einschalten des Antriebs verloren.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/ WinAC		
VALUE	IN	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Wert des Parameters, in den geschrieben werden soll. Er muss bei der positiven Flanke von REQ gültig sein.
USS_DB	INOUT	USS_BASE		–	Den Parameter USS_DB müssen Sie mit dem (statischen) Parameter USS_DB des Instanz-DB verschalten, der erstellt und initialisiert wird, wenn eine Anweisung USS_Drive_Control / USS_Drive_Control_31 in Ihr Programm eingefügt wird.
DONE[1]	OUT	Bool		FALSE	Falls TRUE, wurde der Eingang VALUE in den Antrieb geschrieben. Dieses Bit wird gesetzt, wenn die Anweisung USS_Drive_Control / USS_Drive_Control_31 die Schreibantwort des Antriebs erkennt. Dieses Bit wird beim nächsten Aufruf von USS_Write_Param / USS_Write_Param_31 zurückgesetzt.
ERROR	OUT	Bool		FALSE	Falls TRUE, ist ein Fehler aufgetreten und der Ausgang STATUS ist gültig. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung USS_Port_Scan / USS_Port_Scan_31 gemeldet. Prüfen Sie ggf. zusätzlich den Wert der statischen Variablen USS_DB. w_USSExtendedError im Instanz-DB der Anweisung USS_Drive_Control / USS_Drive_Control_31.
STATUS	OUT	UInt		0	Fehlercode (siehe Fehlermeldungen (Seite 6563)).

[1] Das Bit DONE weist darauf hin, dass gültige Daten aus dem referenzierten Motorantrieb ausgelesen und an die CPU geliefert wurden. Es weist nicht darauf hin, dass die USS-Bibliothek in der Lage ist, sofort einen weiteren Parameter auszulesen. Eine leere Schreibanforderung muss an den Motorantrieb gesendet und auch von der Anweisung quittiert werden, bevor der Parameterkanal zur Verwendung durch den jeweiligen Antrieb frei wird. Der sofortige Aufruf von USS_Read_Param / USS_Read_Param_31 oder USS_Write_Param / USS_Write_Param_31 FC für den spezifischen Motorantrieb führt zu dem Fehler 0x818A.

Anweisungsversionen

USS_Write_Param:

Die Version 1.6 ist funktional identisch zur Version 1.5 und wurde aufgrund interner Maßnahmen erhöht.

USS_Write_Param_31:

Die Version 1.1 ist funktional identisch zur Version 1.0 und wurde aufgrund interner Maßnahmen erhöht.

Allgemeine Informationen zur Antriebseinrichtung

Voraussetzungen für die Antriebseinrichtung

- Für die Antriebe müssen Sie die Verwendung von 4 PKW-Wörtern (**P**arameter**k**ennungs**w**ert) einrichten.
- Konfigurieren Sie die Antriebe für 2, 4, 6 oder 8 PZD-Wörter (**P**ro**z**ess**d**atenbereich).
- Sorgen Sie dafür, dass die Anzahl der PZD-Wörter im Antrieb dem Eingang PZD_LEN der Anweisung USS_Drive_Control des Antriebs entspricht.
- Sorgen Sie dafür, dass die Datenübertragungsrate aller Antriebe dem Eingang BAUD der Anweisung USS_Port_Scan entspricht.
- Sorgen Sie dafür, dass der Antrieb für die USS-Kommunikation eingerichtet ist.
- Stellen Sie sicher, dass im Antrieb festgelegt ist, dass der Frequenzsollwert über die USS-Schnittstelle bereitgestellt wird.
- Stellen Sie sicher, dass die Antriebsadresse festgelegt ist (Bereiche: 1-16).
  Diese Adresse muss dem Eingang DRIVE am Baustein USS_Drive_Control des Antriebs entsprechen.
- Stellen Sie sicher, dass das RS485-Netzwerk ordnungsgemäß abgeschlossen ist.

SINAMICS V20-Antrieb anschließen und einrichten

Ein Anwendungsbeispiel zum Betrieb eines SINAMICS V20 an einer S7-1200 finden Sie im Internet. (http://support.automation.siemens.com/WW/view/de/63696870)

SINAMICS V20-Antrieb anschließen

Anschlussbeispiel eines SIEMENS G120(C)-Antriebs an einem USS-Netzwerk. Anschlussbeispiele für andere Antriebe finden im Handbuch des entsprechenden Antriebs.

Der Anschluss eines SINAMICS G120(C)-Antriebs an das USS-Netzwerk erfolgt über eine Steckverbindung. Der Anschluss ist kurzschlussfest und potenzialgetrennt.

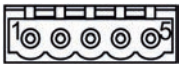

1 0 V Bezugspotential
2 RS485N, Empfangen und Senden (-)
3 RS485N, Empfangen und Senden (+)
4 Kabelschirm
5 Nicht verwendet

Bild 4-33 Anschluss USS

ACHTUNG
Unterschiedliche Bezugsspannungen Wenn Sie Geräte miteinander verbinden, die nicht die gleiche Bezugsspannung haben, kann dies unerwünschte Ströme im Verbindungskabel hervorrufen. Diese unerwünschten Ströme können Kommunikationsfehler verursachen oder Sachschaden in den Geräten hervorrufen. Stellen Sie sicher, dass alle Geräte, die Sie über ein Kommunikationskabel miteinander verbinden, entweder den gleichen Bezugsleiter im Stromkreis haben oder elektrisch getrennt sind, damit keine unerwünschten Ströme auftreten. Stellen Sie sicher, dass die Schirmung mit Masse oder Pin 1 des Bussteckers am Antrieb verbunden ist. Stellen Sie sicher, dass die Verdrahtungsklemme 2 (GND) des G120(C) mit Masse verbunden ist.

Falls der RS485-Master (z. B. S7-1200 CPU mit Kommunikkationsmoduls CM1241) über einen PROFIBUS-Stecker angeschlossen ist, verdrahten Sie die Busleitungen wie folgt:

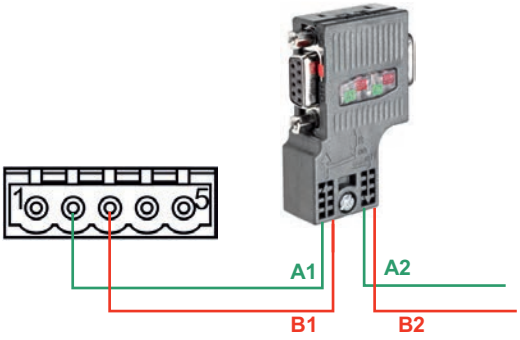

Bild 4-34 Anschluss Kommunikationsmodul

Handelt es sich beim RS485-Master um einen abschließenden Teilnehmer im Netz, oder handelt es sich um eine Punkt-zu-Punkt-Verbindung, müssen Sie die Klemmen A1 und B1 (nicht A2 und B2) des PROFIBUS-Steckers verwenden, weil diese die Abschlusseinstellungen ermöglichen (z. B. beim DP-Steckverbinder 6ES7972-0BB52-0XA0).

Falls der G120(C) als abschließender Teilnehmer im Netz konfiguriert ist, müssen Sie den Schalter für den Busabschlusswiderstand auf "ON" stellen.

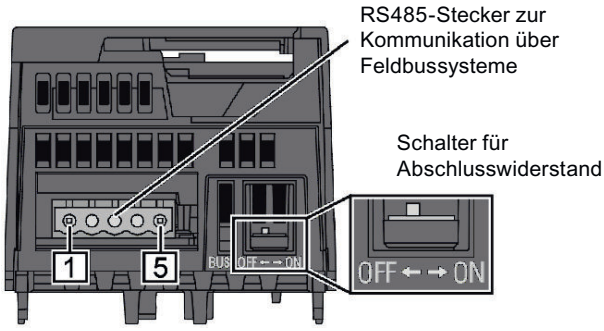

Bild 4-35 Anschluss abschließender Teilnehmer

Einrichten eines G120(C)-Antriebs

Bevor Sie einen Antrieb an die S7-1500 oder ET 200SP anschließen, müssen Sie sicherstellen, dass der Antrieb über folgende Systemparameter verfügt.

Schritt	Anweisung	Betriebsanleitung	
		G120 [1]	G120C [2]
1	Führen Sie eine Grundinbetriebnahme des Antriebs mit dem Operator Panel BOP-2 durch. Der Umrichter bietet unterschiedliche vordefinierte Belegungen (Makros) für seine Ein- und Ausgänge und die Feldbus-Schnittstelle. Wählen Sie im 9. Schritt der Grundinbetriebnahme (MAC PAR p15) das Makro 21 für die USS-Kommunikation. Damit sind auch folgende Parameter vorbelegt: • Datenübertragungsrate (p2020): 38400 Bit/s • Anzahl PZD (p2022): 2 • Anzahl PKW (p2023): variabel Hinweis: Die Grundinbetriebnahme kann auch mit der Inbetriebnahmesoftware STARTER oder mit SINAMICS Startdrive durchgeführt werden.	Kap. 4.4.3	Kap. 6.4.1
2	Legen Sie die USS-Adresse des Umrichters über die Adressschalter auf der Control Unit des G120 bzw. auf dem G120(C) fest. • Gültiger Adressbereich: 1 … 30 Hinweis: Sie können die USS-Adresse auch über Parameter p2021 oder über STARTER bzw. SINAMICS Startdrive einstellen.	Kap. 6.2.2.1	Kap. 8.4.2.1
	Bei den folgenden Schritten greifen Sie mit dem BOP-2 direkt auf Parameter zu, indem Sie deren Nummern wählen und deren Werte modifizieren.	Kap. 4.4.2	Kap. 6.4.2
3	Passen Sie die folgenden kommunikationsbezogenen Umrichterparameter Ihrer Anwendung an: • Datenübertragungsrate (p2020), sofern ≠ 38400 Bit/s (Achten Sie auf Übereinstimmung mit Parameter BAUD der Kommunikationsanweisung USS_Port_Scan.) • Anzahl PZD (p2022), sofern ≠ 2 (Achten Sie auf Übereinstimmung mit Parameter PZD_LEN der Kommunikationsanweisung USS_Drive_Control.) • Anzahl PKW (p2023) = 4 (Ändern Sie den über Makro 21 defaultmäßig auf "variabel" (127) eingestellten Wert auf "4" (die Anweisungen USS_Read_Param und USS_Write_Param erfordern dies.) • Feldbus SS Überwachungszeit [ms] (p2040)	Kap. 6.2.2.2	Kap. 8.4.2.1
4	Legen Sie die Quelle für den Drehzahlsollwert fest. • n_soll Ausw (p1000[0]) = 6 (Der Drehzahlsollwert wird über den USS-Bus bereitgestellt.)		

Schritt	Anweisung	Betriebsanleitung	
		G120 [1]	G120C [2]
5	Stellen Sie die Bezugsgröße für Drehzahl und Frequenz ein. • n_Bezug f_Bezug (p2000) = (6.00 min^{-1} bis 210000.00 min^{-1}) (Alle relativ angegebenen Drehzahlen oder Frequenzen beziehen sich auf diese Bezugsgröße. Die Bezugsgröße entspricht 100 % bzw. 4000_{hex} (Wort) oder 4000 0000_{hex} (Doppelwort). Dabei gilt: $f_{Bezugsgröße}$ (in Hz) = $n_{Bezugsgröße}$ (in ((min^{-1}) / 60) x Polpaarzahl))		
6	Übertragen Sie die Parameter in den nichtflüchtigen Speicher. • Par speichern (p0971) = 1		

1) G120 (http://support.automation.siemens.com/WW/view/de/62089662)

2) G120(C) (http://support.automation.siemens.com/WW/view/de/61462568)

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/ maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüberschreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichenverzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endesequenz"	-
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertragung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#8088	Längenangabe größer als der im Empfangspuffer eingestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehlercode auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wählen Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).

Fehlercode	Beschreibung	Abhilfe
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).

Fehlercode	Beschreibung	Abhilfe
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).

Fehlercode	Beschreibung	Abhilfe
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		

Fehlercode	Beschreibung	Abhilfe
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		

Fehlercode	Beschreibung	Abhilfe
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)

Fehlercode	Beschreibung	Abhilfe
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.

Fehlercode	Beschreibung	Abhilfe
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)

Fehlercode	Beschreibung	Abhilfe
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem

Fehlercode	Beschreibung	Abhilfe
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.

Fehlercode	Beschreibung	Abhilfe
16#8828 [1]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8428 [1]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %

Fehlercode	Beschreibung	Abhilfe
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

MODBUS (RTU)

Überblick über die Modbus RTU-Kommunikation

Modbus RTU-Kommunikation

Modbus RTU (Remote Terminal Unit) ist ein Standardprotokoll für die Kommunikation im Netzwerk und verwendet die RS232- oder RS422/485-Verbindung für die serielle Datenübertragung zwischen Modbus-Geräten im Netzwerk.

Modbus RTU nutzt ein Master/Slave-Netzwerk, in dem die gesamte Kommunikation von einem einzigen Master-Gerät ausgelöst wird, während die Slaves lediglich auf die Anforderung des Masters reagieren können. Der Master sendet eine Anforderung an eine Slave-Adresse und nur der Slave mit dieser Slave-Adresse antwortet auf den Befehl.

Sonderfall: Modbus-Slaveadresse 0 sendet ein Broadcast-Telegramm an alle Slaves (ohne Slave-Antwort).

Modbus-Funktionscodes

- Eine CPU, die als Modbus RTU-Master betrieben wird, kann Daten und E/A-Zustände in einem über eine Kommunikationsverbindung verbundenen Modbus RTU-Slave lesen und schreiben.
- Eine CPU, die als Modbus RTU-Slave betrieben wird, ermöglicht es einem über eine Kommunikationsverbindung verbundenen Modbus RTU-Master, Daten und E/A-Zustände in der eigenen CPU zu lesen und zu schreiben.

Tabelle 4-316 Funktionen zum Lesen von Daten: Dezentrale E/A und Programmdaten lesen

Modbus-Funktionscode	Funktionen zum Lesen von Daten aus dem Slave (Server) - Standard-adressierung
01	Ausgangsbits lesen: 1 bis 2000/1992[1)] Bits pro Anforderung
02	Eingangsbits lesen: 1 bis 2000/1992[1)] Bits pro Anforderung
03	Halteregister lesen: 1 bis 125/124[1)] Worte pro Anforderung
04	Eingangswörter lesen: 1 bis 125/124[1)] Worte pro Anforderung

1) bei erweiterter Adressierung

Tabelle 4-317 Funktionen zum Schreiben von Daten: Dezentrale E/A schreiben und Programmdaten ändern

Modbus-Funktionscode	Funktionen zum Schreiben von Daten in den Slave (Server) - Standard-adressierung
05	Ein Ausgangsbit schreiben: 1 Bit pro Anforderung
06	Ein Halteregister schreiben: 1 Wort pro Anforderung
15	Ein oder mehrere Ausgangsbits schreiben: 1 bis 1960 Bits pro Anforderung
16	Ein oder mehrere Halteregister schreiben: 1 bis 122 Worte pro Anforderung

- Die Modbus-Funktionscodes 08 und 11 bieten Diagnosemöglichkeiten für die Kommunikation mit dem Slave-Gerät.
- Modbus-Slaveadresse 0 sendet ein Broadcast-Telegramm an alle Slaves (ohne Slave-Antwort; für Funktionscodes 5, 6, 15, 16).

Tabelle 4-318 Stationsadressen im Modbus-Netzwerk

Station		Adresse
RTU-Station	Standardstationsadresse	1 bis 247 sowie 0 für Broadcast
	Erweiterte Stationsadresse	1 bis 65535 sowie 0 für Broadcast

Modbus-Speicheradressen

Die tatsächlich verfügbare Anzahl von Modbus-Speicheradressen (Ein-/Ausgangsadresen) richtet sich nach der CPU-Variante, nach dem verfügbaren Arbeitsspeicher.

Modbus RTU-Anweisungen in Ihrem Programm

- Modbus_Comm_Load: Eine Ausführung von Modbus_Comm_Load ist erforderlich, um PtP-Parameter wie Datenübertragungsgeschwindigkeit, Parität und Datenflusskontrolle einzurichten. Nachdem das Kommunikationsmodul für das Modbus RTU-Protokoll konfiguriert ist, kann er nur von der Anweisung Modbus_Master oder der Anweisung Modbus_Slave verwendet werden.
- Modbus_Master: Mit der Modbus-Master-Anweisung kann die CPU als Modbus-RTU-Mastergerät für die Kommunikation mit einem oder mehreren Modbus-Slavegeräten eingesetzt werden.
- Modbus_Slave: Mit der Modbus-Slave-Anweisung kann die CPU als Modbus-RTU-Slavegerät für die Kommunikation mit einem Modbus-Mastergerät eingesetzt werden.

Modbus_Comm_Load: Kommunikationsmodul für Modbus konfigurieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Modbus_Comm_Load konfiguriert ein Kommunikationsmodul für die Kommunikation über das Modbus-RTU-Protokoll). Wenn Sie die Anweisung Modbus_Comm_Load in Ihr Programm einfügen, wird automatisch ein Instanz-Datenbaustein zugewiesen.

Die Konfigurationsänderungen von Modbus_Comm_Load werden nicht in der CPU sondern auf dem CM gespeichert. Bei Spannungswiederkehr und Ziehen/Stecken wird das CM mit den

in der Gerätekonfiguration gespeicherten Daten parametriert. In diesen Fällen muss die Anweisung Modbus_Comm_Load erneut aufgerufen werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Startet die Anweisung bei steigender Flanke dieses Eingangs.
PORT	IN	Port	Laddr	0	Spezifiziert das Kommunikationsmodul, über das kommuniziert wird: • für S7-1500/S7-1200: "HW-Kennung" aus der Gerätekonfiguration. Der symbolische Name des Ports ist im Register "Systemkonstanten" der PLC-Variablentabelle zugewiesen und kann vor dort übernommen werden. • für S7-300/S7-400: “Eingangsadresse“ aus der Gerätekonfiguration. In den Systemen S7-300/400/WinAC wird dem Parameter PORT die in HWCN vergebene Eingangsadresse zugewiesen.
BAUD	IN	UDInt	DWord	9600	Auswahl der Datenübertragungsgeschwindigkeit Zulässige Werte sind: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 bit/s.
PARITY	IN	UInt	Word	0	Auswahl der Parität: • 0 – Keine • 1 – Ungerade • 2 – Gerade
FLOW_CTRL	IN	UInt	Word	0	Auswahl der Flusskontrolle: • 0 – (Standard) Keine Flusskontrolle • 1 – Hardware-Flusskontrolle mit RTS immer EIN (gilt nicht bei RS422/485-CMs) • 2 – Hardware-Flusskontrolle mit RTS geschaltet (gilt nicht bei RS422/485-CMs)
RTS_ON_DLY	IN	UInt	Word	0	Auswahl RTS-Einschaltverzögerung: • 0 – Keine Verzögerung von "RTS aktiv", bis das erste Zeichen des Telegramms gesendet wird. • 1 bis 65535 – Verzögerung in Millisekunden von "RTS aktiv", bis das erste Zeichen des Telegramms gesendet wird (gilt nicht bei RS422/485-CMs). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
RTS_OFF_DLY	IN	UInt	Word	0	Auswahl RTS-Ausschaltverzögerung: • 0 – Keine Verzögerung nach der Übertragung des letzten Zeichens bis "RTS inaktiv" • 1 bis 65535 – Verzögerung in Millisekunden von der Übertragung des letzten Zeichens bis "RTS inaktiv" (gilt nicht bei RS422/485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.
RESP_TO	IN	UInt	Word	1000	Antwort-Timeout: 5 ms bis 65535 ms - Zeit in Millisekunden, die Modbus_Master auf eine Antwort vom Slave wartet. Wenn der Slave während dieses Zeitraums nicht antwortet, wiederholt Modbus_Master die Anforderung oder beendet die Anforderung mit einem Fehler, wenn die angegebene Anzahl von Wiederholungen (siehe unten, Parameter RETRIES) gesendet wurde.
MB_DB	IN/OUT	MB_BASE		-	Ein Verweis auf den Instanz-Datenbaustein der Anweisungen Modbus_Master oder Modbus_Slave. Der Parameter MB_DB muss mit dem (statischen und daher an der Anweisung unsichtbaren) Parameter MB_DB der Anweisung Modbus_Master oder Modbus_Slave verschaltet werden.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Comm_Load Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DONE	OUT	Bool		FALSE	Das Bit DONE ist einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde.
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Der Fehlercode im Parameter STATUS ist nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		16#7000	Fehlercode (siehe Fehlermeldungen (Seite 6603))

Modbus_Comm_Load wird ausgeführt, um einen Port für das Modbus-RTU-Protokoll zu konfigurieren. Nachdem ein Port für das Modbus-RTU-Protokoll konfiguriert ist, kann er nur von den Anweisungen Modbus_Master oder Modbus_Slave verwendet werden.

Für die Konfiguration jedes Kommunikationsports, der für die Modbus-Kommunikation eingesetzt wird, muss eine Ausführung von Modbus_Comm_Load genutzt werden. Sie müssen jedem Port, den Sie nutzen, einen eindeutigen Instanz-DB Modbus_Comm_Load zuweisen. Führen Sie Modbus_Comm_Load nur dann erneut aus, wenn Kommunikationsparameter wie Datenübertragungsgeschwindigkeit oder Parität geändert werden müssen sowie ggf. nach Netzwiederkehr.

Wenn Sie z. B. Modbus_Master oder Modbus_Slave in Ihr Programm einfügen, wird der Anweisung ein Instanz-Datenbaustein zugewiesen. Sie müssen den Parameter MB_DB der Anweisung Modbus_Comm_Load mit dem Parameter MB_DB der Anweisung Modbus_Master bzw. Modbus_Slave verschalten.

Modbus_Comm_Load Datenbausteinvariablen

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von Modbus_Comm_Load, die in Ihrem Programm verwendet werden können.

Tabelle 4-319 Statische Variablen im Instanz-DB

Variable	Datentyp		Standard	Beschreibung
	S7-1200/1500	S7-300/400/WinAC		
ICHAR_GAP	Word		0	Maximale Zeichenverzugszeit zwischen den Zeichen. Dieser Parameter wird in Millisekunden angegeben und dient dazu, den erwarteten Zeitraum zwischen den empfangenen Zeichen zu erhöhen. Die entsprechende Anzahl Bitzeiten für diesen Parameter wird zum Modbus-Standardwert von 35 Bitzeiten (3,5 Zeichenzeiten) addiert.
RETRIES	Word		2	Anzahl der Wiederholungsversuche, die der Master durchführt, bevor der Fehlercode 0x80C8 für "Keine Antwort" zurückgegeben wird.
EN_SUPPLY_VOLT	Bool		0	Diagnose für fehlende Versorgungsspannung L+ einschalten
MODE	USInt	Byte	0	Betriebsart Zulässige Betriebsarten sind: • 0 = Vollduplex (RS232) • 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) • 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master, CM PtP (ET 200SP)) • 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave, CM PtP (ET 200SP)) • 4 = Halbduplex (RS485) Zweidraht-Betrieb [1)]
LINE_PRE	USInt	Byte	0	Vorbelegung der Empfangsleitung Zulässige Vorbelegungen sind: • 0 = "Keine" Vorbelegung [1)] • 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung): Bei dieser Vorbelegung ist Break-Erkennung möglich. Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • 2 = Signal R(A)=0 V, Signal R(B)=5 V: Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.

Variable	Datentyp		Stan-dard	Beschreibung
	S7-1200/1500	S7-300/400/WinAC		
BRK_DET	USInt	Byte	0	Break-Erkennung Zulässig sind: • 0 = Break-Erkennung deaktiviert • 1 = Break-Erkennung aktiviert
EN_DI-AG_ALARM	Bool		0	Diagnosemeldung aktivieren: • 0 - nicht aktiviert • 1 - aktiviert
STOP_BITS	USINT	Byte	1	Anzahl Stopp-Bits; • 1 = 1 Stopp-Bit, • 2 = 2 Stopp-Bits, • 0, 3 bis 255 = reserviert

[1] Erforderliche Einstellung beim Einsatz von PROFIBUS-Kabeln bei CM 1241 für RS485

Anweisungsversionen

Die Version 3.1 ist funktional identisch zur Version 3.0 und wurde aufgrund interner Maßnahmen erhöht.

Modbus_Master: Als Modbus-Master kommunizieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Die Anweisung Modbus_Master kommuniziert als Modbus-Master über einen Port, der mit der Anweisung Modbus_Comm_Load konfiguriert wurde. Wenn Sie die Anweisung Modbus_Master in Ihr Programm einfügen, wird automatisch ein Instanz-Datenbaustein zugewiesen. Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit dem (statischen) Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	FALSE = Keine Anforderung TRUE = Anforderung, Daten an den Modbus-Slave zu senden
MB_ADDR	IN	UInt	Word	-	Modbus-RTU-Stationsadresse: Standard-Adressierungsbereich (1 bis 247 sowie 0 für Broadcast) Erweiterter Adressierungsbereich (1 bis 65535 sowie 0 für Broadcast) Der Wert 0 ist für den Broadcast des Telegramms an alle Modbus Slaves reserviert. Für den Broadcast werden nur die Modbus-Funktionscodes 05, 06, 15 und 16 unterstützt.
MODE	IN	USInt	Byte	0	Auswahl Modus: Gibt die Art der Anforderung an (Lesen, Schreiben oder Diagnose). Weitere Informationen finden Sie in der Tabelle der Modbus-Funktionen unten.
DATA_ADDR	IN	UDInt	DWord	0	Anfangsadresse im Slave: Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll. Die gültigen Adressen finden Sie in der Tabelle der Modbus-Funktionen unten.
DATA_LEN	IN	UInt	Word	0	Datenlänge: Gibt die Anzahl der Bits oder Wörter an, auf die diese Anforderung zugreifen soll. Die gültigen Längen finden Sie in der Tabelle der Modbus-Funktionen unten.
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Master Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
DATA_PTR	IN/OUT	Variant	Any	-	Daten-Pointer: Zeigt auf die Merker- oder DB-Adresse für die zu schreibenden oder zu lesenden Daten. Ab Anweisungsversion V3.0: Der Parameter darf auf einen optimierten Speicherbereich zeigen. Im optimierten Speicherbereich ist ein Single-Element oder ein Array erlaubt mit folgenden Datentypen: Bool, Byte, Char, Word, Int, DWord, DInt, Real, USInt, UInt, UDInt, SInt, WChar. Jeder andere Datentyp führt zur Fehlermeldung 16#818C.
DONE	OUT	Bool		FALSE	Das Bit DONE ist einen Zyklus lang TRUE, nachdem die letzte Anforderung fehlerfrei ausgeführt wurde.
BUSY	OUT	Bool		-	• FALSE – Keine Auftrag für Modbus_Master aktiv • TRUE – Auftrag für Modbus_Master in Bearbeitung

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Der Fehlercode im Parameter STATUS ist nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 6603))

Variablen im Datenbaustein des Modbus-Master

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von Modbus_Master, die in Ihrem Programm verwendet werden können.

Tabelle 4-320 Statische Variablen im Instanz-DB

Variable	Datentyp	Standard	Beschreibung
Blocked_Proc_Timeout	Real	3.0	Zeitdauer (in Sekunden), die auf eine blockierte Modbus-Master-Instanz gewartet werden soll, bevor diese Instanz als AKTIV entfernt wird. Dies kann beispielsweise vorkommen, wenn eine Master-Anforderung ausgegeben wurde und das Programm dann aufhört, die Masterfunktion aufzurufen, bevor es die Anforderung vollständig beendet hat. Der Zeitwert muss größer als 0 und kleiner als 55 Sekunden sein, ansonsten tritt ein Fehler auf. Siehe hierzu auch Abschnitte "Regeln für die Kommunikation des Modbus-Master" und "Aufruf der Anweisung Modbus_Master mit verschienenen Parametereinstellungen".
Extended_Addressing	Bool	FALSE	Konfiguriert die Slave-Stationsadresse als einzelnes oder doppeltes Byte. • FALSE = Ein-Byte-Adresse; 0 bis 247 • TRUE = Zwei-Byte-Adresse (entspricht der erweiterten Adressierung); 0 bis 65535

Variable	Datentyp	Standard	Beschreibung
Compatibility_Mode [1]	Bool	FALSE	Kompatibilitätsmodus mit CP 341, CP 441-2 und ET 200S 1SI mit Treiber für Modbus RTU und mit ET 200S 1SI für Modbus. Der Standardwert ist 0. • FALSE = nach Modbus-Spezifikation, nicht kompatibel • TRUE = kompatibel – Bei FC1 und FC2 gilt: Die aus dem Empfangstelegramm gelesenen Daten werden wortweise in den adressierten Speicher der CPU geschrieben und dabei byteweise ausgetauscht. Falls die Anzahl zu übertragender Bits nicht ein Vielfaches von 16 ist, werden die nicht relevanten Bits im letzten Wort auf Null gesetzt. – Bei FC15 gilt: Die zu übertragenden Worte werden wortweise aus dem adressierten Speicher gelesen und byteweise in das Sendetelegramm geschrieben. Falls die Anzahl zu übertragender Bits nicht ein Vielfaches von 8 ist, werden die nicht relevanten Bits im letzten Byte unverändert aus dem adressierten Speicher gelesen und im Sendetelegram eingetragen.
MB_DB	MB_BA-SE	-	Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit diesem Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

[1] Die PtP-Kommunikationsmodule verhalten sich gemäß der Modbus Spezifikation. Um ein Verhalten wie bei CP 341, CP 441-2 und ET 200SP 1SI für Modbus bisher üblich zu erhalten, verwenden Sie den Parameter "Compatibility_Mode".

Ihr Programm kann in die Variablen Blocked_Proc_Timeout und Extended_Addressing Werte schreiben, um die Modbus-Master-Operationen zu steuern.

Regeln für die Kommunikation des Modbus-Master

- Modbus_Comm_Load muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung Modbus_Master mit diesem Port kommunizieren kann.
- Wenn ein Port als Modbus Master verwendet werden soll, darf dieser Port nicht von Modbus_Slave verwendet werden. Eine oder mehrere Instanzen von Modbus_Master [1] können mit diesem Port verwendet werden. Doch alle Ausführungen von Modbus_Master müssen den gleichen Instanz-DB für den Port verwenden.
- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss die Anweisung Modbus_Master auf abgeschlossene Aufträge abfragen (DONE, ERROR).
- Es wird empfohlen, alle Ausführungen von Modbus_Master für einen bestimmten Port aus einem Programmzyklus-OB aufzurufen. Modbus-Master-Anweisungen können nur in jeweils einem Programmzyklus oder in jeweils einer zyklischen/zeitgesteuerten Bearbeitungsebene ausgeführt werden. Sie dürfen nicht in verschiedenen Bearbeitungsebenen bearbeitet werden. Die Vorrangunterbrechung einer Modbus-Master-Anweisung durch eine andere Modbus-Master-Anweisung in einer Bearbeitungsebene mit höherer Priorität führt zu nicht ordnungsgemäßem Betrieb. Die Modbus-Master-Anweisungen dürfen nicht in Anlauf-, Diagnose- oder Zeitfehlerebenen bearbeitet werden.

[1)] Mit "Instanz von Modbus Master" ist hier ein Aufruf der Anweisung Modbus_Master mit gleicher Verschaltung mit einer Anweisung Modbus_Comm_Load und gleicher Einstellung für die Parameter MB_ADDR, MODE, DATA_ADDR und DATA_LEN gemeint.

Beispiel.

Modbus_Master wird aufgerufen mit MODE=0 und DATA_ADDR=10

Dieser Auftrag ist nun aktiv, bis er mit DONE=1 oder ERROR=1 beendet wird oder bis die am Parameter Blocked_Proc_Timeout parametrierte Zeitüberwachung abgelaufen ist. Wird nach Ablauf der Zeitüberwachung ein neuer Auftrag gestartet, bevor der vorherige Auftrag beendet ist, so wird der vorherige Auftrag ohne Fehlermeldung abgebrochen.

Falls nun, während dieser Auftrag läuft, ein zweiter Aufruf der Anweisung mit den gleichen Instanzdaten, aber anderen Einstellungen der Parameter MODE und DATA_ADDR erfolgt, so wird dieser zweite Aufruf mit ERROR=1 und STATUS=8200 beendet.

Aufruf der Anweisung Modbus_Master mit verschienenen Parametereinstellungen

Sollen mehrere Aufrufe der Anweisung Modbus_Master mit unterschiedlichen Einstellungen für MB_ADDR, MODE, DATA_ADDR oder DATA_LEN im Ihrem Programm platziert werden, müssen Sie sicherstellen, dass zu einer Zeit nur einer dieser Aufrufe aktiv ist, anderenfalls wird die Fehlermeldung 16#8200 (die Schnittstelle ist durch eine laufende Anfrage belegt) ausgegeben.

Falls ein Aufruf nicht fertig bearbeitet werden kann, wird die Zeitüberwachung über den Parameter Blocked_Proc_Timeout aktiv und beendet den laufenden Auftrag.

Parameter REQ

FALSE = Keine Anforderung; TRUE = Anforderung, Daten an den Modbus-Slave zu senden

Freigabe der angeforderten Übertragung. Dadurch wird der Inhalt des Puffers zur Punkt-zu-Punkt-Kommunikationsschnittstelle übertragen.

Über die Parameter DATA_ADDR und MODE wählen Sie den Modbus-Funktionscode aus.

DATA_ADDR (Modbus-Anfangsadresse im Slave): Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll.

Die Anweisung Modbus_Master nutzt statt eines Funktionscodeeingangs einen Eingang MODE. Die Kombination aus MODE und DATA_ADDR legt den Funktionscode fest, der im eigentlichen Modbus-Telegramm verwendet wird. Die folgende Tabelle zeigt die Zuordnung zwischen dem Parameter MODE, dem Modbus-Funktionscode und dem Modbus-Adressbereich in DATA_ADDR.

Tabelle 4-321 Modbus-Funktionen

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
0				Bits pro Anforderung			01	Ausgangsbits lesen:		
	1	bis	9999	1	bis	2000/1992 [1]		0	bis	9998

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
0				Bits pro Anforderung			02	Eingangsbits lesen:		
	10001	bis	19999	1	bis	2000/1992 [1]		0	bis	9998
0				Wörter pro Anforderung			03	Halteregister lesen:		
	40001	bis	49999	1	bis	125/124 [1]		0	bis	9998
	400001	bis	465535	1	bis	125/124 [1]		0	bis	65534
0				Wörter pro Anforderung			04	Eingangswörter lesen:		
	30001	bis	39999	1	bis	125/124 [1]		0	bis	9998
1				Bit pro Anforderung			05	Ein Ausgangsbit schreiben:		
	1	bis	9999	1				0	bis	9998
1				1 Wort pro Anforderung			06	Ein Halteregister schreiben:		
	40001	bis	49999	1				0	bis	9998
	400001	bis	465535	1				0	bis	65524
1				Bits pro Anforderung			15	Mehrere Ausgangsbits schreiben:		
	1	bis	9999	2	bis	1968/1960 [1]		0	bis	9998
1				Wörter pro Anforderung			16	Mehrere Halteregister schreiben:		
	40001	bis	49999	2	bis	123/122		0	bis	9998
	400001	bis	465534	2	bis	123/122 [1]		0	bis	65534
2 [2]				Bits pro Anforderung			15	Ein oder mehrere Ausgangsbits schreiben:		
	1	bis	9999	1	bis	1968/1960 [1]		0	bis	9998
2 [2]				Wörter pro Anforderung			16	Ein oder mehrere Halteregister schreiben:		
	40001	bis	49999	1	bis	123		0	bis	9998
	400001	bis	465535	1	bis	122 [1]		0	bis	65534
11	Beide Operanden DATA_ADDR und DATA_LEN von Modbus_Master werden bei dieser Funktion ignoriert.						11	Statuswort und Ereigniszähler der Slave-Kommunikation lesen. Das Statuswort zeigt beschäftigt an (0 – nicht beschäftigt, 0xFFFF - beschäftigt). Der Ereigniszähler wird bei jeder erfolgreichen Abarbeitung eines Telegramms inkrementiert.		
80				1 Wort pro Anforderung			08	Slave-Zustand über Daten-Diagnosecode 0x0000 prüfen (Prüfschleifentest – Slave gibt ein Echo der Anforderung zurück)		
	-			1				-		
81				1 Wort pro Anforderung			08	Slave-Ereigniszähler über Daten-Diagnosecode 0x000A zurücksetzen		
	-			1				-		

MODE	DATA_ADDR (Modbus-Adresse)			DATA_LEN (Länge der Daten)			Modbus-Funktionscode	Operation und Daten		
104 [3]				Wörter pro Anforderung			04	Eingangswörter lesen		
	0	bis	65535	1	bis	125/124 [1]		0	bis	65535
3 bis 10, 12 bis 79, 82 bis 103, 105 bis 255	-			-				Reserviert		

[1] Bei der erweiterten Adressierung, siehe Parameter Extended_Adressing, ist die maximale Datenlänge je nach Datentyp der Funktion um 1 Byte oder 1 Wort geringer.

[2] MODE 2 erlaubt das Schreiben von einem oder mehreren Ausgangsbits und einem oder mehreren Holding Register unter Verwendung der Modbus-Funktionen 15 und 16.
MODE 1 verwendet die Modbus-Funktionen 5 und 6, um 1 Ausgangsbit und 1 Holding Register zu schreiben und die Modbus-Funktionen 15 und 16, um mehrere Ausgangsbits und mehrere Holding Register zu schreiben.

[3] Bei S7-300/400/WinAC gilt: Reserviert

Parameter DATA_PTR

Der Parameter DATA_PTR zeigt auf die DB- oder M-Adresse, in die geschrieben oder aus der gelesen wird. Wenn Sie einen Datenbaustein verwenden, müssen Sie einen globalen Datenbaustein anlegen, der den Datenspeicher für Lese- und Schreibvorgänge auf Modbus-Slaves bereitstellt.

Hinweis

S7-1200/1500 - Der über DATA_PTR adressierte Datenbaustein muss die direkte Adressierung zulassen

Der Datenbaustein muss die direkte (absolute) und die symbolische Adressierung zulassen.

Hinweis

Verwendung von Funktionscode 5

Mit Funktionscode 5 kann ein einzelnes Bit gesetzt oder gelöscht werden.

Zum Setzen eines Bits muss im ersten Wort des über DATA_PTR adressierten DB- oder Merkerbereichs der Wert "16#FF00" vorgegeben werden.

- Bei S7-1200 kann zum Setzen eines Bits auch der Wert "16#0100" vorgegeben werden.
- Zum Rücksetzen eines Bits muss im ersten Wort des über DATA_PTR adressierten DB- oder Merkerbereichs der Wert "16#0000" vorgegeben werden.

Alle anderen Werte werden mit der ERROR = TRUE und STATUS = 16#8384 abgewiesen.

Datenbausteinstrukturen für den Parameter DATA_PTR

- Diese Datentypen gelten für das Lesen von Wörtern des Modbus-Adressbereichs (DATA_PTR) 30001 bis 39999, 40001 bis 49999 und 400001 bis 465535 sowie für das Schreiben von Wörtern des Modbus-Adressbereichs (Parameter DATA_PTR) 40001 bis 49999 und 400001 bis 465535.
  - Standard-Array der Datentypen WORD, UINT oder INT
  - Benannte Struktur vom Typ WORD, UINT oder INT, bei der jedes Element einen eindeutigen Namen und einen 16-Bit-Datentyp hat.
  - Benannte komplexe Struktur, bei der jedes Element einen eindeutigen Namen und einen 16- oder 32-Bit-Datentyp hat.
- Zum Lesen und Schreiben von Bits für den Modbus-Adressbereich (Parameter DATA_PTR) 00001 bis 09999 und zum Lesen von Bits von 10001 bis 19999.
  - Standardfeld aus Boole'schen Datentypen.
  - Benannte Boole'sche Struktur aus eindeutig benannten Boole'schen Variablen.
- Es ist zwar nicht erforderlich, doch empfehlenswert, dass jede Anweisung Modbus_Master einen eigenen getrennten Speicherbereich hat. Grund hierfür ist, dass die Möglichkeit der Datenzerstörung größer ist, wenn mehrere Anweisungen Modbus_Master in demselben Speicherbereich lesen und schreiben.
- Es ist nicht erforderlich, dass die Datenbereiche für DATA_PTR in demselben globalen Datenbaustein liegen. Sie können einen Datenbaustein mit mehreren Bereichen für Modbus-Lesevorgänge, einen Datenbaustein für Modbus-Schreibvorgänge oder einen Datenbaustein für jede Slave-Station anlegen.

Anweisungsversionen

Die Version 3.0 ist funktional identisch zur Version 2.4 und wurde aufgrund interner Maßnahmen erhöht.

Siehe auch

Modbus_Slave: Als Modbus-Slave kommunizieren (Seite 6590)

Modbus_Slave: Als Modbus-Slave kommunizieren

Hinweis

Einsatz mit CM1241

Der Einsatz dieser Anweisung mit einem CM1241 ist erst ab einem Firmware-Stand V2.1 des Moduls möglich.

Beschreibung

Mit der Anweisung Modbus_Slave kann Ihr Programm über einen CM (RS422/485 oder RS232) als Modbus-Slave kommunizieren. STEP 7 erstellt automatisch einen Instanz-DB, wenn Sie die Anweisung einfügen. Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit dem (statischen) Parameter MB_DB der Anweisung Modbus_Slave verschaltet werden.

Parameter

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
MB_ADDR	IN	UInt	Word	-	Stationsadresse des Modbus-Slaves: Standard-Adressierungsbereich (1 bis 247) Erweiterter Adressierungsbereich (0 bis 65535) Hinweis: 0 ist die Broadcast-Adresse
COM_RST	IN/OUT	---	Bool	FALSE	Initialisierung der Anweisung Modbus_Slave Mit TRUE wird die Anweisung initialisiert. Anschließend wird COM_RST von der Anweisung wieder auf FALSE gesetzt. Hinweis: Der Parameter ist nur für S7-300/400-Anweisungen verfügbar.
MB_HOLD_REG	IN/OUT	Variant	Any	-	Pointer auf den Modbus-Halteregister-DB: Bei dem Modbus-Halteregister kann es sich um den Speicherbereich der Merker oder um einen Datenbaustein handeln. Ab Anweisungsversion V4.0: Der Parameter muss auf einen Speicherbereich zeigen, der eine Länge von mindestens 16 Bit hat. Eine geringere Länge führt zur Fehlermeldung 16#8187. Dies betrifft Single-Elemente, Arrays, STRUCTs und UDTs. Z.B. führt ein Single Bool oder ein Array, das aus weniger als 16 Bool-Elementen besteht, zu der Fehlermeldung. Wenn die Länge kein Vielfaches von 16 Bit ist, können die Restbits am Ende des Speicherbereichs von der Anweisung Modbus_Slave weder gelesen noch geschrieben werden. Der Parameter darf auf einen optimierten Speicherbereich zeigen. Im optimierten Speicherbereich ist ein Single-Element oder ein Array erlaubt mit folgenden Datentypen: Bool, Byte, Char, Word, Int, DWord, DInt, Real, USInt, UInt, UDInt, SInt, WChar. Jeder andere Datentyp führt zur Fehlermeldung 16#818C.
NDR	OUT	Bool		FALSE	Neue Daten bereit: • FALSE – Keine neuen Daten • TRUE – Gibt an, dass neue Daten vom Modbus-Master geschrieben wurden Das Bit NDR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung ohne Fehler beendet wurde.

Parameter	Deklaration	Datentyp		Standard	Beschreibung
		S7-1200/1500	S7-300/400/WinAC		
DR	OUT	Bool		FALSE	Daten lesen: • FALSE – Keine Daten gelesen • TRUE – Gibt an, dass die Anweisung vom Modbus-Master empfangene Daten im Zielbereich abgelegt hat. Das Bit DR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung ohne Fehler beendet wurde.
ERROR	OUT	Bool		FALSE	Das Bit ERROR ist einen Zyklus lang TRUE, nachdem die letzte Anforderung mit Fehler beendet wurde. Wenn die Ausführung mit einem Fehler beendet wird, ist der Fehlercode im Parameter STATUS nur in dem einen Zyklus gültig, in dem ERROR = TRUE ist.
STATUS	OUT	Word		0	Fehlercode (siehe Fehlermeldungen (Seite 6603))

Die Funktionscodes der Modbus-Kommunikation (1, 2, 4, 5 und 15) können Bits und Wörter direkt im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge der CPU lesen und schreiben. Bei diesen Funktionscodes muss der Parameter MB_HOLD_REG als Datentyp größer als ein Byte definiert werden. Die folgende Tabelle zeigt die Beispielzuordnung der Modbus-Adressen zum Prozessabbild in der CPU.

Tabelle 4-322 Zuordnung der Modbus-Adressen zum Prozessabbild

Modbus-Funktionen						S7-1200			
Code	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
01	Bits lesen	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7
02	Bits lesen	Eingang	0	bis	8191	Prozessabbild der Eingänge	E0.0	bis	E1023.7
04	Wörter lesen	Eingang	0	bis	511	Prozessabbild der Eingänge	EW0	bis	EW1022
05	Bit schreiben	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7
15	Bits schreiben	Ausgang	0	bis	8191	Prozessabbild der Ausgänge	A0.0	bis	A1023.7

Tabelle 4-323 Zuordnung der Modbus-Adressen zum Prozessabbild

Modbus-Funktionen						S7-1500 / S7-300 / S7-400			
Funktionscode	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
01	Bits lesen	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6
02	Bits lesen	Eingang	0	bis	9998	Prozessabbild der Eingänge	E0.0	bis	E1249.6
04	Wörter lesen	Eingang	0	bis	9998	Prozessabbild der Eingänge	EW0	bis	EW19996

Modbus-Funktionen						S7-1500 / S7-300 / S7-400			
Funktionscode	Funktion	Datenbereich	Adressbereich			Datenbereich	CPU-Adresse		
05	Bit schreiben	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6
15	Bits schreiben	Ausgang	0	bis	9998	Prozessabbild der Ausgänge	A0.0	bis	A1249.6

Hinweis

Der verfügbare Adressbereich kann, abhängig vom Speicherausbau der CPU, auch kleiner sein.

Die Funktionscodes der Modbus-Kommunikation (3, 6, 16) nutzen ein Modbus-Halteregister, bei dem es sich um einen Adressbereich im Speicherbereich der Merker oder um einen Datenbaustein handeln kann. Der Typ des Halteregisters wird vom Parameter MB_HOLD_REG der Anweisung Modbus_Slave angegeben.

Hinweis

S7-1200/1500 - Typ des Datenbausteins MB_HOLD_REG

Ein Datenbaustein mit Modbus-Halteregister muss die direkte (absolute) und die symbolische Adressierung zulassen.

Tabelle 4-324 Diagnosefunktionen

Modbus-Diagnosefunktionen von Modbus_Slave der S7-1200		
Funktionscodes	Teilfunktion	Beschreibung
08	0000H	Echotest Abfragedaten ausgeben: Die Anweisung Modbus_Slave gibt einem Modbus-Master das Echo eines empfangenen Datenworts zurück.
08	000AH	Kommunikationsereigniszähler löschen: Die Anweisung Modbus_Slave löscht den Kommunikationsereigniszähler, der für Modbus-Funktion 11 verwendet wird.
11		Kommunikationsereigniszähler abrufen: Die Anweisung Modbus_Slave nutzt einen internen Kommunikationsereigniszähler, um die Anzahl erfolgreicher Modbus-Lese- und Modbus-Schreibanforderungen, die an den Modbus-Slave gesendet werden, zu erfassen. Der Zähler wird bei Funktion 8, Funktion 11 und Broadcast-Anforderungen nicht hochgezählt. Außerdem wird bei Anforderungen, die zu Kommunikationsfehlern führen (z. B. Paritäts- oder CRC-Fehler), nicht hochgezählt.

Die Anweisung Modbus_Slave unterstützt Broadcast-Schreibanforderungen von Modbus-Mastern, solange die Anforderungen den Zugriff auf gültige Adressen umfassen. Für Funktionscodes, die bei der Broadcast-Funktion nicht unterstützt werden, erzeugt Modbus_Slave den Fehlercode 16#8188.

Variablen des Modbus-Slave in der Anweisungsversion V3.0

Diese Tabelle zeigt die öffentlichen statischen Variablen im Instanz-Datenbaustein von Modbus_Slave, die in Ihrem Programm verwendet werden können.

Tabelle 4-325 Variablen des Modbus-Slave

Variable	Datentyp	Standard	Beschreibung
HR_Start_Offset	Word	0	Gibt die Anfangsadresse der Modbus-Halteregisters an (Standard = 0)
QB_Start	Word	0	Anfangsadresse des zulässigen schreibbaren Adressierungsbereichs der Ausgänge (Byte 0 bis 65535) Hinweis: Die Variable ist für S7-300, S7-400 und WinAC nicht verfügbar.
QB_Count	Word	0xFFFF	Anzahl der durch den Modbus-Master schreibbaren Ausgangs-Bytes. Hinweis: Die Variable ist für S7-300, S7-400 und WinAC nicht verfügbar.
Extended_Addressing	Bool	FALSE	Erweiterte Adressierung, konfiguriert die Slave-Adressierung als einzelnes oder doppeltes Byte (FALSE = einzelne Byte-Adresse, TRUE = doppelte Byte-Adresse)
Request_Count	Word	0	Die Anzahl aller von diesem Slave empfangenen Anforderungen
Slave_Message_Count	Word	0	Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen
Bad_CRC_Count	Word	0	Die Anzahl der empfangenen Anforderungen, die einen CRC-Fehler aufweisen
Broadcast_Count	Word	0	Die Anzahl der empfangenen Broadcast-Anforderungen
Exception_Count	Word	0	Modbus-spezifische Fehler, die mit einer Exception an den Master quittiert werden
Success_Count	Word	0	Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen ohne Protokollfehler
MB_DB	MB_BASE	-	Der Parameter MB_DB der Anweisung Modbus_Comm_Load muss mit diesem Parameter MB_DB der Anweisung Modbus_Master verschaltet werden.

Ihr Programm kann in die Variablen HR_Start_Offset und Extended_Addressing Werte schreiben und die Modbus-Slave-Operationen steuern. Die anderen Variablen können gelesen werden, um den Modbus-Zustand zu überwachen.

Regeln für die Modbus-Slave-Kommunikation

- Modbus_Comm_Load muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung Modbus_Slave über diesen Port kommunizieren kann.
- Wenn ein Port als Slave auf einen Modbus-Master reagieren soll, darf dieser Port nicht mit der Anweisung Modbus_Master programmiert werden.
- Nur eine Instanz von Modbus_Slave kann mit einem bestimmten Port verwendet werden, andernfalls kann unerwartetes Verhalten auftreten.

- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss den Kommunikationsprozess durch Abfrage der Anweisung Modbus_Slave auf abgeschlossene Sende- und Empfangsvorgänge steuern.
- Die Anweisung Modbus_Slave muss regelmäßig mit einer Häufigkeit ausgeführt werden, die eine zeitnahe Antwort auf eingehende Anforderungen eines Modbus-Masters ermöglicht. Es wird empfohlen, Modbus_Slave in jedem Zyklus aus einem Programmzyklus-OB auszuführen. Modbus_Slave kann aus einem Weckalarm-OB ausgeführt werden, doch ist dies nicht empfehlenswert, weil übermäßige Zeitverzögerungen im Interruptprogramm die Ausführung anderer Interruptprogramme temporär blockieren können.

Zeitsteuerung des Modbus-Signals

Modbus_Slave muss regelmäßig ausgeführt werden, um jede Anforderung des Modbus-Masters zu empfangen und entsprechend zu antworten. Die Häufigkeit der Ausführung von Modbus_Slave richtet sich nach dem vom Modbus-Master vorgegebenen Timeout-Wert für die Antwort. Dies ist in der nachstehenden Abbildung dargestellt.

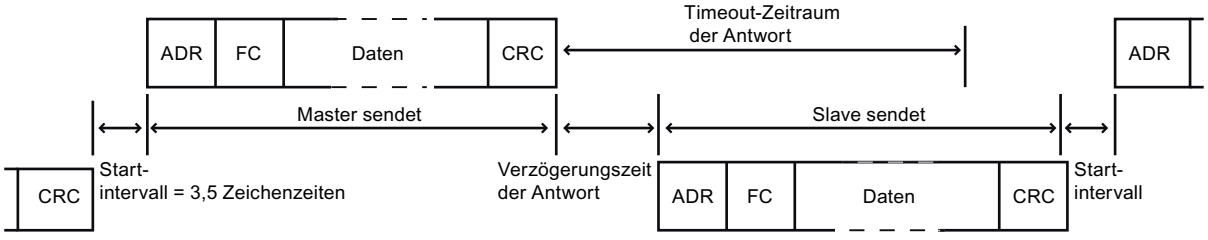

Der Timeout-Zeitraum der Antwort (RESP_TO) ist die Zeitdauer, die ein Modbus-Master auf den Beginn einer Antwort von einem Modbus-Slave wartet. Dieser Zeitraum wird nicht vom Modbus-Protokoll definiert, sondern von einem Parameter der Anweisung Modbus_Comm_Load. Da sowohl das Empfangen als auch das Senden eines Telegramms mehrere, mindestens aber je drei Aufrufe der Anweisung Modbus_Slave erfordert, sollten Sie Modbus_Slave mindestens zwölfmal während des Timeout-Zeitraums für die Antwort des Modbus-Masters ausführen, damit das Empfangen und Senden von Daten durch den Modbus-Slave doppelt so oft ausgeführt wird, wie durch den Timeout-Zeitraum vorgegeben.

HR_Start_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 40001 oder 400001. Diese Adressen entsprechen der Anfangsadresse des Halteregisters im Zielsystemspeicher. Sie können jedoch die Variable HR_Start_Offset konfigurieren, um eine andere Anfangsadresse als 40001 oder 400001 für das Modbus-Halteregister zu konfigurieren.

Die Adresse 0 im Empfangstelegramm entspricht der Anfangsadresse des Halteregisters im Zielsystemspeicher. Über die Variable HR_Start_Offset können Sie eine andere Anfangsadresse als 0 für das Modbus-Halteregister konfigurieren.

Sie können z. B. ein Halteregister mit Beginn an MW100 und einer Länge von 100 Wörtern konfigurieren. Mit HR_Start_Offset = 20 entspricht die Adresse 20 im Empfangstelegramm der

Anfangsadresse des Halteregisters im Zielspeicher (MW100). Jede Adresse im Empfangstelegramm unter 20 und über 119 führt zu einem Adressierungsfehler.

Tabelle 4-326 Beispiel für die Adressierung des Modbus-Halteregisters, falls DATA_PTR ein Zeiger auf MW100 mit der Länge 100 Worte ist

HR_Start_Offset	Adresse	Minimum	Maximum
0	Modbus-Adresse (Wort)	0	99
	S7-1500 Adresse	MW100	MW298
20	Modbus-Adresse (Wort)	20	119
	S7-1500 Adresse	MW100	MW298

HR_Start_Offset ist ein Wortwert, der die Anfangsadresse des Modbus-Halteregisters angibt und im Instanz-Datenbaustein Modbus_Slave gespeichert ist. Sie können diese öffentliche statische Variable über die Parameter-Klappliste auswählen, nachdem Sie Modbus_Slave in Ihr Programm eingefügt haben.

Wenn Sie beispielsweise Modbus_Slave in ein KOP-Netzwerk eingefügt haben, können Sie in ein vorheriges Netzwerk gehen und den Wert HR_Start_Offset mit dem Move-Befehl zuweisen. Der Wert muss vor der Ausführung von Modbus_Slave zugewiesen werden.

Modbus-Slave-Variable über den Standard-DB-Namen eingeben:

1. Positionieren Sie den Cursor im Parameterfeld OUT1 und geben Sie das Zeichen m ein.
2. Wählen Sie in der Klappliste den gewünschten Instanz-DB der Anweisung Modbus_Slave aus.
3. Positionieren Sie den Cursor rechts vom DB-Namen (nach den Anführungszeichen) und geben Sie einen Punkt ein.
4. Wählen Sie in der Klappliste "Modbus_Slave_DB.HR_Start_Offset" aus.

Anweisungsversionen

Die Version 4.0 ist funktional identisch zur Version 3.0 und wurde aufgrund interner Maßnahmen erhöht.

Telegrammaufbau

Extended_Addressing

Auf die Extended_Addressing-Variable wird in ähnlicher Weise zugegriffen wie auf die HR_Start_Offset-Referenz, außer dass es sich bei der Extended_Addressing-Variable um einen Booleschen Wert handelt.

Für die Modbus-Slave-Adressierung kann, mit Extended_Adressing = FALSE, ein einzelnes Byte (dies ist der Modbus-Standard) oder zwei Bytes konfiguriert werden (Extended_Adressing = TRUE). Die erweiterte Adressierung dient zur Adressierung von mehr als 247 Geräten in

einem einzigen Netzwerk. Mit Extended_Adressing = TRUE können Sie maximal 65535 Adressen ansprechen. Im Folgenden wird ein Modbus-Telegramm als Beispiel gezeigt.

Tabelle 4-327 Slave-Adresse mit einem Byte (Byte 0)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Anforderung	Slave-Adresse	Funktionscode	Anfangsadresse		Daten		
Gültige Antwort	Slave-Adresse	Funktionscode	Länge	Daten...			
Fehlermeldung	Slave-Adresse	0xxx	Exceptioncode				

Tabelle 4-328 Slave-Adresse mit zwei Bytes (Byte 0 und Byte 1)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anforderung	Slave-Adresse		Funktionscode	Anfangsadresse		Daten	
Gültige Antwort	Slave-Adresse		Funktionscode	Länge	Daten...		
Fehlermeldung	Slave-Adresse		0xxx	Exceptioncode			

Beschreibung der Telegramme

Der Datenverkehr zwischen Master und Slave bzw. zwischen Slave und Master beginnt mit der Slave-Adresse, gefolgt vom Funktionscode. Dann werden die Daten übertragen. Der Aufbau des Datenfeldes ist abhängig vom verwendeten Funktionscode. Am Ende des Telegramms wird die Prüfsumme (CRC) übertragen.

Funktionscode 1 - Diese Funktion ermöglicht das Lesen einzelner Ausgangsbits

Tabelle 4-329 FC 1 - Ausgangsbits lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktionscode 1	Anfangsadresse		Anzahl Ausgänge	
Gültige Antwort	Slave-Adresse	Funktionscode 1	Länge [1]	Ausgangsdaten [3)]		
Fehlermeldung	Slave-Adresse	0x81	Exceptioncode [2]	---		

[1] Länge: Falls die Anzahl der Ausgänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

[3] Die Ausgangsdaten können mehrere Bytes umfassen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 1	Anfangsadresse		Anzahl Ausgänge	
Gültige Ant-wort	Slave-Adresse		Funktions-code 1	Länge [1]	Ausgangsdaten		
Fehlermel-dung	Slave-Adresse		0x81	Exceptionc-ode [2]	---		

[1] Länge: Falls die Anzahl der Ausgänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

[3] Die Ausgangsdaten können mehrere Bytes umfassen

Funktionscode 2 - Diese Funktion ermöglicht das Lesen einzelner Eingangsbits

Tabelle 4-330 FC 2 - Eingangsbits lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 2	Anfangsadresse		Anzahl Eingänge	
Gültige Antwort	Slave-Adresse	Funktions-code 2	Länge [1]	Eingangsdaten		
Fehlermeldung	Slave-Adresse	0x82	Exceptioncode [2]	---		

[1] Länge: Falls die Anzahl der Eingänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 2	Anfangsadresse		Anzahl Eingänge	
Gültige Ant-wort	Slave-Adresse		Funktions-code 2	Länge [1]	Eingangsdaten		
Fehlermel-dung	Slave-Adresse		0x82	Exceptionc-ode [2]	---		

[1] Länge: Falls die Anzahl der Eingänge dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 3 - Diese Funktion ermöglicht das Lesen einzelner Register

Tabelle 4-331 FC 3 - Halteregister lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 3	Anfangsadresse		Anzahl Register	
Gültige Antwort	Slave-Adresse	Funktions-code 3	Länge [1]	Registerdaten		
Fehlermeldung	Slave-Adresse	0x83	Exceptioncode [2]	---		

[1] Länge: Anzahl Bytes

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 3	Anfangsadresse		AnzahlRegister	
Gültige Ant-wort	Slave-Adresse		Funktions-code 3	Länge [1]	Registerdaten		
Fehlermel-dung	Slave-Adresse		0x83	Exceptionc-ode [2]	---		

[1] Länge: Anzahl Bytes

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 4 - Diese Funktion ermöglicht das Lesen einzelner Register

Tabelle 4-332 FC 4 - Eingangswörter lesen

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 4	Anfangsadresse		Anzahl Eingangswörter	
Gültige Antwort	Slave-Adresse	Funktions-code 4	Länge [1]	Eingangsdaten		
Fehlermeldung	Slave-Adresse	0x84	Exceptioncode [2]	---		

[1] Länge: 2 * Anzahl Eingangswörter

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 4	Anfangsadresse		Anzahl Eingangswörter	
Gültige Ant-wort	Slave-Adresse		Funktions-code 4	Länge [1]	Eingangsdaten		
Fehlermel-dung	Slave-Adresse		0x84	Exceptionc-ode [2]	---		

[1] Länge: 2 * Anzahl Eingangswörter

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 5 - Mit dieser Funktion kann ein einzelnes Bit gesetzt oder gelöscht werden

Tabelle 4-333 FC 5 - Ein Ausgangsbit schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 5	Anfangsadresse		Wert	
Gültige Antwort	Slave-Adresse	Funktions-code 5	Länge	Wert		
Fehlermeldung	Slave-Adresse	0x85	Exceptioncode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 5	Anfangsadresse		Wert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 5	Länge	Wert		
Fehlermel-dung	Slave-Adresse		0x85	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 6 - Diese Funktion ermöglicht das Schreiben einzelner Register

Tabelle 4-334 FC 6 - Halteregister schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 6	Adresse		Register	
Gültige Antwort	Slave-Adresse	Funktions-code 6	Adresse		Register	
Fehlermeldung	Slave-Adresse	0x86	Exceptioncode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 6	Adresse		Register	
Gültige Ant-wort	Slave-Adresse		Funktions-code 6	Adresse		Register	
Fehlermel-dung	Slave-Adresse		0x86	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 8 - Diese Funktion dient der Überprüfung der Kommunikationsverbindung

Tabelle 4-335 FC 8 - Slave-Zustand

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 8	Diagnosecode		Testwert	
Gültige Antwort	Slave-Adresse	Funktions-code 8	Diagnosecode		Testwert	
Fehlermeldung	Slave-Adresse	0x88	Exceptioncode [1]	---		

[1] E-Code: 01 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 8	Diagnosecode		Testwert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 8	Diagnosecode		Testwert	
Fehlermel-dung	Slave-Adresse		0x88	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 03 oder 04

Funktionscode 11 - Diese Funktion ermöglicht das Lesen von 2 Byte "Status-Word" und ein 2 Byte "Event-Counter"

Tabelle 4-336 FC 11 - Ereigniszähler der Slave-Kommunikation

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Anfrage	Slave-Adresse	Funktions-code 11	---			
Gültige Antwort	Slave-Adresse	Funktions-code 11	Status		Ereigniszähler	
Fehlermeldung	Slave-Adresse	0x8B	Exceptioncode [1]	---		

[1] E-Code: 01 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Anfrage	Slave-Adresse		Funktions-code 11	---			
Gültige Ant-wort	Slave-Adresse		Funktions-code 11	Status		Ereigniszähler	
Fehlermel-dung	Slave-Adresse		0x8B	Exceptionc-ode [1]	---		

[1] E-Code: 01 oder 04

Funktionscode 15 - Diese Funktion ermöglicht das Schreiben mehrerer Bits

Tabelle 4-337 FC 15 - Ein/mehrere Ausgangsbits schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Anfrage	Slave-Adres-se	Funktions-code 15	Anfangsadresse		Anzahl Ausgangs-wörter		Bytezäh-ler [1]	Wert	
Gültige Ant-wort	Slave-Adres-se	Funktions-code 15	Anfangsadresse		Anzahl Ausgangs-wörter		---		
Fehlermel-dung	Slave-Adres-se	0x8F	Excepti-oncode [2]	---					

[1] Bytezähler: Falls die Anzahl der Bytes dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Anfrage	Slave-Adresse		Funktions-code 15	Anfangsadresse		Anzahl Ausgangswör-ter		Bytezäh-ler [1]	Wert	
Gültige Ant-wort	Slave-Adresse		Funktions-code 15	Anfangsadresse		Anzahl Ausgangswör-ter		---		
Fehlermel-dung	Slave-Adresse		0x8F	Exceptioncode [2]		---				

[1] Bytezähler: Falls die Anzahl der Bytes dividiert durch 8 einen Rest ergibt, muss die Anzahl Bytes um 1 erhöht werden.

[2] E-Code: 01 oder 02 oder 03 oder 04

Funktionscode 16 - Diese Funktion ermöglicht das Schreiben einzelner oder mehrerer Register

Tabelle 4-338 FC 16 - Ein/mehrere Halteregister schreiben

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Anfrage	Slave-Adresse	Funktionscode 16	Anfangsadresse		Anzahl Register		Bytezähler [1]	Wert	
Gültige Antwort	Slave-Adresse	Funktionscode 16	Anfangsadresse		Anzahl Register		---		
Fehlermeldung	Slave-Adresse	0x90	Exceptioncode [2]	---					

[1] Bytezähler: Anzahl der Register * 2

[2] E-Code: 01 oder 02 oder 03 oder 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Anfrage	Slave-Adresse		Funktionscode 16	Anfangsadresse		Anzahl Register		Bytezähler [1]	Wert	
Gültige Antwort	Slave-Adresse		Funktionscode 16	Anfangsadresse		Anzahl Register		---		
Fehlermeldung	Slave-Adresse		0x90	Exceptioncode [2]		---				

[1] Bytezähler: Anzahl der Register * 2

[2] E-Code: 01 oder 02 oder 03 oder 04

Fehlermeldungen

Übersicht der Fehlermeldungen - PtP

Die Fehlermeldungen werden jeweils am STATUS-Ausgang einer Anweisung bereitgestellt und können dort ausgewertet bzw. im Anwenderprogramm verarbeitet werden.

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
RECEIVE-Status und Fehlercodes		
16#0094	Telegrammende erkannt durch "Empfang einer festen/maximalen Telegrammlänge"	-
16#0095	Telegrammende erkannt durch "Nachrichten-Zeitüberschreitung"	-
16#0096	Telegrammende erkannt durch Ablauf der "Zeichenverzugszeit"	-
16#0097	Das Telegramm wurde abgebrochen, da die maximale Antwortzeit erreicht wurde.	-
16#0098	Telegrammende erkannt durch Erfüllung der "Lese Nachrichtenlänge aus Nachricht" Bedingungen	-
16#0099	Telegrammende erkannt durch Empfang der "Endesequenz"	-

Fehlercode	Beschreibung	Abhilfe
SEND-Status und Fehlercodes		
16#7000	Baustein im Leerlauf	-
16#7001	Erstaufruf für ein neues Telegramm: Datenübertragung angestoßen	-
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8085	Unzulässige Längenangabe	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8088	Längenangabe größer als der im Empfangspuffer eingestellte Bereich. Hinweis: Falls am Parameter BUFFER der Datentyp STRING spezifiziert wurde, erscheint dieser Fehlercode auch, falls die aktuelle Stringlänge kleiner ist als der am Parameter LENGTH angegebene Wert.	Ändern Sie den Bereich im Empfangspuffer oder wählen Sie eine geeignete Telegrammlänge, passend zum eingestellten Bereich im Empfangspuffer. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
RECEIVE-Status und Fehlercodes		
16#7002	Zwischenaufruf: Datenübertragung läuft	-
16#8088	Die Anzahl der empfangen Zeichen ist größer als am Parameter BUFFER spezifiziert.	Wählen Sie eine geeignete Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#8090	Konfigurationsfehler: Ungerade Anzahl Bytes bei WString	Wählen Sie eine gerade Anzahl Bytes.
Fehlermeldecodes der Sonderfunktionen		
16#818F	Falsche Einstellung für Parameternummer (nur bei USS)	Wählen Sie eine geeignete Parameternummer (PARAM). Zulässig sind: 0-2047
16#8190	Falsche Einstellung der CRC-Berechnung	Wählen Sie einen geeigneten Wert für die CRC-Berechnung. Zulässig sind: deaktiviert oder aktiviert. Prüfen Sie, ob das angesprochene Modul die CRC-Berechnung unterstützt.
16#8191	Falsche Einstellung des Diagnosealarms	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
16#8193	Das Modul unterstützt keine Versorgungsspannungsdiagnose L+.	Wählen Sie einen geeigneten Wert für "Diagnosealarm". Zulässig sind: Diagnosealarm deaktiviert oder Diagnosealarm aktiviert. Prüfen Sie, ob das angesprochene Modul das Generieren von Diagnosealarmen unterstützt.
Fehlermeldecodes der "Port-Konfiguration"		
16#81A0	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll (PROTOCOL).

Fehlercode	Beschreibung	Abhilfe
16#81A1	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit (BAUD).
16#81A2	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie einen geeigneten Wert für "Parität" (PARITY). Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#81A3	Das Modul unterstützt diese Anzahl Datenbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Datenbits" (DATABITS). Zulässig sind: • 7 (2) • 8 (1)
16#81A4	Das Modul unterstützt diese Anzahl Stoppbits nicht.	Wählen Sie einen geeigneten Wert für "Anzahl Stoppbits" (STOPBITS). Zulässig sind: • 1 (1) • 2 (2)
16#81A5	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie eine für das Modul zulässige Datenflusskontrolle (FLOWCTRL).
16#81A7	Unzulässiger Wert für XON oder XOFF	Wählen Sie geeignete Werte für XON (XONCHAR) und XOFF(XOFFCHAR). Zulässiger Wertebereich: 0...255
16#81AA	Unzulässige Betriebsart	Zulässige Betriebsarten sind: • Vollduplex (RS232) (0) • Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt) (1) / (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master) (2)/ (CM PtP (ET 200SP)) • Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave) (3) • Halbduplex (RS485) Zweidraht-Betrieb (4)

Fehlercode	Beschreibung	Abhilfe
16#81AB	Unzulässige Vorbelegung der Empfangsleitung	Zulässige Vorbelegungen sind: • "Keine" Vorbelegung (0) • Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) (1): Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)". • Signal R(A)=0 V, Signal R(B)=5 V (2): Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv).
16#81AC	Unzulässiger Wert für "Break-Erkennung"	Wählen Sie einen geeigneten Wert für "Break-Erkennung. Zulässig sind: • Break-Erkennung deaktiviert (0) • Break-Erkennung aktiviert (1).
16#81AF	Das Modul unterstützt dieses Protokoll nicht.	Wählen Sie ein für das Modul zulässiges Protokoll.
Fehlercodes der "Sendekonfiguration"		
16#81B5	Mehr als 2 Endezeichen -oder- Endesequenz > 5 Zeichen	Wählen Sie geeignete Werte für "Endezeichen" und "Endesequenz". Zulässig sind: • deaktiviert (0), • 1 (1) oder 2 (2) Endezeichen bzw. • deaktiviert (0), • 1 (1) bis 5 (5) Zeichen für die Endesequenz.
16#81B6	Send Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Sendekonfiguration gesendet wird.
Fehlercodes der "Empfangskonfiguration"		
16#81C0	Unzulässige Startbedingung	Wählen Sie eine geeignete Startbedingung. Zulässig sind: • Break vor Telegrammstart senden • Idle Line senden.
16#81C1	Unzulässige Endebedingung oder keine Endebedingung gewählt	Wählen Sie eine geeignete Endebedingung (siehe AUTOHOTSPOT).
16#81C3	Unzulässiger Wert für "Maximale Länge der Nachricht"	Wählen Sie einen geeigneten Wert für "Maximale Länge der Nachricht" (MAXLEN). Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)
16#81C4	Unzulässiger Wert für "Offset des Längenangabe in der Nachricht"	Wählen Sie einen geeigneten Wert für "Offset des Längenangabe in der Nachricht". Zulässiger Wertebereich (modulabhängig): 1-1024/2048/4096 (Byte)

Fehlercode	Beschreibung	Abhilfe
16#81C5	Unzulässiger Wert für "Größe des Längenfeldes"	Wählen Sie einen geeigneten Wert für "Größe des Längenfeldes" (LENGTHSIZE). Zulässiger Wertebereich in Byte: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Unzulässiger Wert für "Anzahl nicht zur Längenangabe zählender Zeichen"	Wählen Sie einen geeigneten Wert für "Anzahl nicht zur Längenangabe zählender Zeichen" (LENGTHM). Zulässiger Wertebereich: 0 bis 255 (Byte)
16#81C7	Summe aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" ist größer als die maximale Telegrammlänge	Wählen Sie geeignete Werte für "Offset in der Nachricht", "Größe des Längenfeldes" und "Anzahl nicht zählender Zeichen". Zulässiger Wertebereich: • Offset in der Nachricht (modulabhängig): 0-1024/2048/4096 (Byte) • Größe des Längenfeldes: 1, 2 oder 4 (Byte) • Anzahl nicht zählender Zeichen: 0-255 (Byte)
16#81C8	Unzulässiger Wert für "Antwort-Zeitüberschreitung"	Wählen Sie einen geeigneten Wert für "Antwort-Zeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)
16#81C9	Unzulässiger Wert für "Zeichenverzugszeit"	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit". Zulässiger Wertebereich: 1-65535 (Bitzeiten)
16#81CB	Telegrammende-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CC	Telegrammanfangs-Sequenz aktiviert aber keines der Zeichen ist für das Prüfen aktiviert	Aktivieren Sie eines oder mehrere Zeichen für die Prüfung.
16#81CD	Unzulässiger Wert für "Überschreiben verhindern"	Wählen Sie einen geeigneten Wert für "Überschreiben verhindern". Zulässig sind: • Überschreiben verhindern deaktiviert (0) oder • Überschreiben verhindern aktiviert (1)
16#81CE	Unzulässiger Wert für "Empfangspuffer im Anlauf löschen"	Wählen Sie einen geeigneten Wert für "Empfangspuffer im Anlauf löschen". Zulässig sind: • Empfangspuffer im Anlauf löschen deaktiviert (0) • Empfangspuffer im Anlauf löschen aktiviert (1)
SEND-Status und Fehlercodes		
16#81D0	Sendeanfrage empfangen während der Laufzeit eines Sendeauftrags	Stellen Sie sicher, dass zur Laufzeit eines Sendeauftrags keine weitere Sendeanfrage erfolgt.
16#81D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.

Fehlercode	Beschreibung	Abhilfe
16#81D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#81D3	Sendepufferüberlauf / Sendetelegramm zu groß	Wählen Sie eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
16#81D5	Übertragung abgebrochen, da Parametrierung geändert, Drahtbruch erkannt oder CPU in STOP	Prüfen Sie Parametrierung, Drahtbruch und Zustand der CPU.
16#81D6	Übertragung abgebrochen, da Endekennzeichen nicht empfangen	Prüfen Sie die Parametrierung der Endezeichen und das Telegramm des Kommunikationspartners.
16#81D7	Kommunikation zwischen Anwenderprogramm und Modul gestört	Prüfen Sie die Kommunikation (z. B. Übereinstimmung der Sequenznummer).
16#81D8	Übertragungsversuch abgewiesen, da Modul nicht parametriert	Parametrieren Sie das Modul.
16#81DF	Das Modul hat wegen einem der folgenden Gründe die Schnittstelle zum FB rückgesetzt: • Modul wurde neu gestartet • Modul wurde neu parametriert • CPU-STOP	—
Fehlercodes der Empfangskonfiguration		
16#81E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie die Funktion zum Empfangen im Anwenderprogramm häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#81E1	Telegramm abgebrochen: Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#81E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#81E3	Telegramm abgebrochen: Zeichenüberlauffehler	Firmware-Fehler: Bitte wenden Sie sich an den Customer Support.
16#81E4	Telegramm abgebrochen: Länge aus "Offset in der Nachricht + Größe des Längenfeldes + Anzahl nicht zählender Zeichen" größer als der Empfangspuffer	Überprüfen Sie die Einstellungen für Offset in der Nachricht, Größe des Längenfeldes und Anzahl nicht zählender Zeichen.
16#81E5	Telegramm abgebrochen: Break	Empfangsleitung zum Partner ist unterbrochen. Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
16#81E6	Maximale Anzahl "Gepufferte Empfangstelegramme" überschritten	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle oder erhöhen Sie die Anzahl gepufferter Telegramme.
16#81E7	Synchronisierungsfehler zwischen Modul und Receive_P2P	Stellen Sie sicher, das nicht verschiedene Instanzen des Receive_P2P auf dasselbe Modul zugreifen.
16#81E8	Telegramm abgebrochen: Zeichenverzugszeit abgelaufen, bevor das Nachrichtenende-Kriterium erkannt wurde	Das Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät nach, welches in die Übertragungsleitung eingeschaltet wird.

Fehlercode	Beschreibung	Abhilfe
16#81E9	Modbus CRC-Fehler (nur Kommunikationsmodule, die Modbus unterstützen)	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EA	Modbus-Telegramm zu kurz (nur Kommunikationsmodule, die Modbus unterstützen)	Unterschreitung der Minimallänge des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#81EB	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte) Überprüfen Sie die Parameter der Telegrammende-Erkennung.
Fehlercodes V24-Begleitsignale		
16#81F0	Das Modul unterstützt keine V24-Begleitsignale	Sie haben versucht, für ein Modul, das keine V24-Begleitsignale unterstützt, Begleitsignale einzustellen. Stellen Sie sicher, dass es sich um ein RS232-Modul handelt bzw. dass RS232-Modus (ET 200SP) eingestellt ist.
16#81F1	Keine Bedienung der V24-Begleitsignale	Falls Hardware-Datenflusskontrolle aktiv ist, können die V24-Begleitsignale nicht von Hand bedient werden.
16#81F2	Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DCE (data communication equipment) sein.
16#81F3	Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.	Überprüfen Sie den parametrierten Typ der Baugruppe. Der Baugruppentyp muss vom Typ DTE (data terminal equipment) sein.
16#81F4	Block-Header-Fehler (z. B. falscher Blocktyp oder falsche Blocklänge)	Überprüfen Sie den Instanz-DB und den Block-Header.
Fehlercodes der Empfangskonfiguration		
16#8201 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Geben Sie einen Zeiger auf einen der folgenden Datentypen ein: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions zeigt auf optimimierten Speicherbereich größer 1 kByte oder Receive_Conditions zeigt auf optimierten Speicherbereich und die Empfangslänge ist größer als der durch Receive_Conditions adressierte Bereich.	Geben Sie einen Zeiger auf einen Bereich folgender maximalen Länge ein: • Optimierter Speicherbereich: 1 kByte • Nicht optimierter Bereich: 4 kByte Hinweis: Falls der Zeiger auf einen optimierten Speicherbereich zeigt, senden Sie nicht mehr als 1 kByte.
16#8229 [1)]	Receive_Conditions ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Anzahl Bits ein Vielfaches von 8 sein.
Fehlercodes Allgemein		

Fehlercode	Beschreibung	Abhilfe
16#8280	Negative Quittung beim Lesen des Moduls	Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern RDREC.STATUS, sowie in der Beschreibung des SFB RDREC. • Überprüfen Sie die Eingabe am Parameter PORT • Setzen Sie vor dem 1. Aufruf den Parameter COM_RST.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Fehlercodes der Empfangskonfiguration		
16#82C1	Unzulässiger Wert für "Gepufferte Empfangstelegramme".	Wählen Sie einen geeigneten Wert für "Gepufferte Empfangstelegramme". Zulässiger Wertebereich: 1-255
16#82C2	Receive Configuration abgelehnt, da Protokoll 3964(R) ausgewählt	Stellen Sie sicher, dass bei eingestelltem Protokoll 3964(R) keine Empfangskonfiguration gesendet wird.
16#8301 [1)]	Receive_Conditions ist Zeiger auf unerlaubten Datentyp	Wählen sie einen zulässigen Datentyp. Zulässig sind: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Bereichslängenfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8324	Bereichsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8328	Einstellungsfehler beim Lesen eines Parameters	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
SEND-Status und Fehlercodes		
16#8328 [1)]	BUFFER ist Zeiger auf BOOL mit Bitanzahl ungleich n * 8	Falls Sie einen Zeiger auf BOOL verwenden, muss die Bitanzahl ein Vielfaches von 8 sein.
Fehlercodes der Empfangskonfiguration		
16#8332	Unzulässiger Datenbaustein am Parameter Receive_Conditions	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#833A	Die Bezeichnung des Datenbausteins am Parameter Receive_Conditions weißt auf einen nicht geladenen Datenbaustein.	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8351	Unzulässiger Datentyp	Überprüfen Sie die Eingabe am Parameter Receive_Conditions
16#8352 [1)]	Receive_Conditions zeigt nicht in einen Datenbaustein	Überprüfen Sie den Zeiger auf die Receive_Conditions
16#8353 [1)]	Receive_Conditions zeigt nicht auf eine Struktur vom Typ Receive_Conditions	Überprüfen Sie den Zeiger auf die Receive_Conditions
Fehlercodes 3964(R) Protokoll		
16#8380	Parametrierfehler: Unzulässiger Wert für "Zeichenverzugszeit".	Wählen Sie einen geeigneten Wert für "Zeichenverzugszeit" (CharacterDelayTime). Zulässiger Wertebereich: 1-65535 (ms)

Fehlercode	Beschreibung	Abhilfe
16#8381	Parametrierfehler: Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie einen geeigneten Wert für "Antwortzeitüberschreitung" (AcknDelayTime). Zulässiger Wertebereich: 1-65535 (ms)
16#8382	Parametrierfehler: Unzulässiger Wert für "Priorität".	Wählen Sie einen geeigneten Wert für "Priorität" (Priority). Zulässig sind: • Hoch (1) • Niedrig (0)
16#8383	Parametrierfehler: Unzulässiger Wert für "Blockcheck"	Wählen Sie einen geeigneten Wert für "Blockcheck" (BCC). Zulässig sind: • mit Blockcheck (1) • ohne Blockcheck (0)
16#8384	Parametrierfehler: Unzulässiger Wert für "Aufbauversuche".	Wählen Sie einen geeigneten Wert für "Aufbauversuche" (BuildupAttempts). Zulässiger Wertebereich: 1-255
16#8385	Parametrierfehler: Unzulässiger Wert für "Übertragungsversuche".	Wählen Sie einen geeigneten Wert für "Übertragungsversuche" (RepetitionAttempts). Zulässiger Wertebereich: 1-255
16#8386	Laufzeitfehler: Anzahl Aufbauversuche überschritten	Prüfen Sie das Schnittstellenkabel und die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion richtig parametriert ist.
16#8387	Laufzeitfehler: Anzahl Übertragungsversuche überschritten	Prüfen Sie das Schnittstellenkabel, die Übertragungsparameter und die Parametrierung des Kommunikationspartners.
16#8388	Laufzeitfehler: Fehler beim "Blockcheck-Zeichen" Der intern gebildete Wert des Blockcheck-Zeichens stimmt nicht mit dem vom Partner am Verbindungsende empfangenen Blockcheck-Zeichen überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#8389	Laufzeitfehler: Unzulässiges Zeichen empfangen während auf freien Empfangspuffer gewartet wurde	Der Sendewunsch des Kommunikationspartners (STX, 02H) wird erst dann mit DLE beantwortet, wenn der Empfangspuffer leer ist. Vorher darf (außer erneut STX) kein weiteres Zeichen empfangen werden. Korrektes Verhalten des Partnergerätes ggf. mit Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838A	Laufzeitfehler: Logischer Fehler während des Empfangs. Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Korrektes Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät prüfen, das in die Übertragungsleitung eingeschaltet wird.
16#838B	Laufzeitfehler: Zeichenverzugszeit überschritten	Partnergerät ist zu langsam oder gestört. Prüfen Sie dies ggf. mit einem Schnittstellentestgerät, welches in die Übertragungsleitung eingeschaltet wird.

Fehlercode	Beschreibung	Abhilfe
16#838C	Laufzeitfehler: Wartezeit auf freien Empfangspuffer gestartet	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#838D	Laufzeitfehler: Nach NAK startet die Telegrammwiederholung nicht innerhalb von 4 s	Überprüfen Sie den Kommunikationspartner. Ein eventuell fehlerhaft empfangenes Telegramm muss vom Partner innerhalb von 4 Sekunden wiederholt werden.
16#838E	Laufzeitfehler: In Ruhestellung wurden ein oder mehrere Zeichen (außer NAK und STX) empfangen.	Prüfen Sie das korrekte Verhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
16#838F	Laufzeitfehler: Initialisierungskonflikt - Beide Partner haben hohe Priorität eingestellt	Stellen Sie bei einem der Partner die Priorität auf "Niedrig"
16#8391	Parametrierfehler: 3964 Parametrierdaten abgelehnt, da Freeport eingestellt	Stellen Sie sicher, dass bei eingestelltem Protokoll Freeport keine 3964-Parametrierdaten gesendet werden.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - Modbus

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
Konfigurationsfehler der Schnittstelle - Modbus_Comm_Load		
16#8181	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie am Parameter BAUD eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#8182	Das Modul unterstützt diese Paritätseinstellung nicht.	Wählen Sie am Parameter PARITY einen geeigneten Wert für "Parität". Zulässig sind: • Keine (1) • Gerade (2) • Ungerade (3) • Mark (4) • Space (5) • Beliebig (6)
16#8183	Das Modul unterstützt diese Art der Datenflusskontrolle nicht.	Wählen Sie am Parameter FLOW_CTRL eine für das Modul zulässige Datenflusskontrolle.
16#8184	Unzulässiger Wert für "Antwortzeitüberschreitung".	Wählen Sie am Parameter RESP_TO einen geeigneten Wert für "Antwortzeitüberschreitung". Zulässiger Wertebereich: 1-65535 (ms)

Fehlercode	Beschreibung	Abhilfe
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS, oder WRREC.STATUS sowie der Beschreibung des SFB WRREC.
16#8282	Modul nicht verfügbar	Überprüfen Sie die Eingabe am Parameter PORT und stellen Sie sicher, dass das Modul erreichbar ist.
Konfigurationsfehler - Modbus_Slave		
16#8186	Unzulässige Slave-Adresse	Wählen Sie am Parameter MB_ADDR eine geeignete Slave-Adresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8187	Unzulässiger Wert am Parameter MB_HOLD_REG	Wählen Sie am Parameter MB_HOLD_REG einen geeigneten Wert für das Holding Register.
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#818C	Der Zeiger auf einen MB_HOLD_REG-Bereich muss ein Datenbaustein oder ein Merkerbereich sein.	Wählen Sie einen geeigneten Wert für den Zeiger auf den MB_HOLD_REG-Bereich.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS, sowie der Beschreibung des SFB WRREC.
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG

Fehlercode	Beschreibung	Abhilfe
16#8456 [1)]	Fehler beim Ausführen der Anweisung. Die Fehlerursache steht im statischen Parameter STATUS.	Ermitteln Sie den Wert des Parameter SFCSTATUS. Lesen Sie dessen Bedeutung in der Beschreibung zum SFC51, Parameter STATUS.
Konfigurationsfehler - Modbus_Master		
16#8180	Unzulässiger Wert für Parameter MB_DB	Der an der Anweisung Modbus_Comm_Load parametrierte Wert für MB_DB (Instanzdaten-DB) ist nicht zulässig. Überprüfen Sie die Verschaltung der Anweisung Modbus_Comm_Load und dessen Fehlermeldungen.
16#8186	Unzulässige Stationsadresse	Wählen Sie am Parameter MB_ADDR eine geeignete Stationsadresse. Zulässig sind: 1-247 bei Standardadressbereich; 1-65535 bei erweitertem Adressbereich (0 ist reserviert für Broadcast)
16#8188	Unzulässige Betriebsart oder Broadcast (MB_ADDR = 0) und Parameter MODE ≠ 1	Wählen Sie bei Betriebsart Broadcast für MODE den Wert 1 oder wählen Sie eine andere Betriebsart.
16#8189	Unzulässige Datenadresse	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818A	Unzulässige Längenangabe	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge. Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818B	Unzulässiger Wert für DATA_PTR	Wählen Sie am Parameter DATA_PTR einen geeigneten Wert für den Daten-Pointer (M- oder DB-Adresse). Siehe Beschreibung Modbus_Master (Seite 6583) im Infosystem
16#818C	Verschaltungsfehler des Parameters DATA_PTR	Überprüfen Sie die Verschaltung der Anweisung.
16#818D	Die Länge des mit DATA_PTR adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger DATA_PTR
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Send_P2P.WRREC.STATUS oder Receive_P2P.WRREC.STATUS oder Receive_Reset, sowie der Beschreibung des SFB WRREC.
Kommunikationsfehler - Modbus_Master und Modbus_Slave		

Fehlercode	Beschreibung	Abhilfe
16#80D1	Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
16#80D2	"Hardware RTS immer ON": Sendeauftrag abgebrochen da Wechsel von DSR = ON nach OFF	Überprüfen Sie den Kommunikationspartner. Stellen Sie sicher, dass DSR während der gesamten Übertragung ON ist.
16#80E0	Telegramm abgebrochen: Sendepufferüberlauf / Sendetelegramm zu groß	Rufen Sie im Anwenderprogramm die Anweisung häufiger auf oder parametrieren Sie eine Kommunikation mit Datenflusskontrolle.
16#80E1	Telegramm abgebrochen: Paritätsfehlers	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Datenübertragungsgeschwindigkeit, Parität und Stoppbitanzahl gleich eingestellt sind.
16#80E2	Telegramm abgebrochen: Zeichenrahmenfehler	Überprüfen Sie die Einstellungen für Startbit, Datenbits, Paritätsbit, Datenübertragungsgeschwindigkeit und Stoppbit(s).
16#80E3	Telegramm abgebrochen: Zeichenüberlauffehler	Überprüfen Sie die Anzahl der Daten im Telegramm des Kommunikationspartners.
16#80E4	Telegramm abgebrochen: Maximale Telegrammlänge erreicht	Wählen Sie beim Kommunikationspartner eine kleinere Telegrammlänge. Zulässig sind (modulabhängig): 1-1024/2048/4096 (Byte)
Kommunikationsfehler - Modbus_Master		
16#80C8	Der Slave antwortet nicht innerhalb der eingestellten Zeit	Überprüfen Sie die Datenübertragungsgeschwindigkeit, Parität und die Verdrahtung des Slave.
16#80C9	Der Slave antwortet nicht innerhalb der mittels Blocked_Proc_Timeout eingestellten Zeit.	Überprüfen Sie die Einstellung von Blocked_Proc_Timeout. Überprüfen Sie, ob das Modul mit der Anweisung Modbus_Comm_Load parametriert wurde. Eventuell muss das Modul nach Ziehen/Stecken oder nach Spannungswiederkehr über Modbus_Comm_Load nachparametriert werden.
16#8200	Die Schnittstelle ist durch eine laufende Anfrage belegt.	Wiederholen Sie den Auftrag zu einem späteren Zeitpunkt. Stellen Sie sicher, dass kein Auftrag mehr läuft, bevor Sie einen neuen Auftrag starten.
Protokollfehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Der Funktionscode wird nicht unterstützt bzw. wird für Broadcast nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Unzulässige Längenangabe im Anforderungstelegramm	Wählen Sie am Parameter DATA_LEN eine geeignete Datenlänge.
16#8383	Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Unzulässiger Datenwertfehler im Anforderungstelegramm	Überprüfen Sie den Datenwert im Anforderungstelegramm des Modbus-Master
16#8385	Der Diagnosewert wird vom Modbus-Slave nicht unterstützt (Funktionscode 08)	Der Modbus-Slave unterstützt nur die Diagnosewerte 16#0000 und 16#000A.

Fehlercode	Beschreibung	Abhilfe
Protokollfehler - Modbus_Master (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8380	CRC-Fehler	Checksummenfehler des Modbus-Telegramms. Überprüfen Sie den Kommunikationspartner.
16#8381	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Funktionscode wird nicht unterstützt.	Überprüfen Sie den Kommunikationspartner und stellen Sie sicher, dass ein gültiger Funktionscode gesendet wird.
16#8382	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Längenangabe	Wählen Sie eine geeignete Datenlänge.
16#8383	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Unzulässige Datenadresse im Anforderungstelegramm	Wählen Sie am Parameter DATA_ADDR einen geeigneten Wert für Datenadresse.
16#8384	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Datenwertfehler	Überprüfen Sie das Anforderungstelegramm an den Modbus-Slave.
16#8385	Antworttelegramm vom Modbus Slave mit Fehlermeldung: Der Diagnosewert wird vom Modbus-Slave nicht unterstützt	Modbus-Slaves unterstützen nur die Diagnosewerte 16#0000 und 16#000A.
16#8386	Der zurückgemeldete Funktionscode passt nicht zum angefragten Funktionscode.	Überprüfen Sie das Antworttelegramm und die Adressierung des Salve.
16#8387	Ein nicht angefragter Slave antwortet	Überprüfen Sie das Antworttelegramm des Slave. Überprüfen Sie die Adresseinstellungen der Slaves.
16#8388	Fehler in der Antwort des Slave auf eine Schreiben-Anfrage.	Überprüfen Sie das Antworttelegramm des Slave.
16#8828 [1)]	DATA_PTR zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger DATA_PTR
16#8852 [1)]	DATA_PTR ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger DATA_PTR
16#8853 [1)]	DATA_PTR ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger DATA_PTR
16#8855 [1)]	DATA_PTR zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger DATA_PTR
16#8856 [1)]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Master
Fehler - Modbus_Slave (nur Kommunikationsmodule, die Modbus unterstützen)		
16#8428 [1)]	MB_HOLD_REG zeigt auf eine Bitadresse ungleich n * 8	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8452 [1)]	MB_HOLD_REG ist kein Zeiger auf einen DB oder Merkerbereich	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG ist kein Zeiger vom Typ BOOL oder WORD	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8454 [1)]	Die Länge des mit MB_HOLD_REG adressierten Bereichs überschreitet die DB-Länge oder der adressierte Bereich ist zu klein für die Anzahl der zu lesenden oder zu schreibenden Daten.	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG zeigt auf einen schreibgeschützten DB	Überprüfen Sie den Zeiger MB_HOLD_REG
16#8456 [1)]	Fehler beim Aufruf des SFC51	Wiederholen Sie den Aufruf der Anweisung Modbus_Slave
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1)] Nur bei Anweisungen für S7-300/400 CPUs

Übersicht der Fehlermeldungen - USS

Fehlercode	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8180	Längenfehler in der Antwort des Antriebs	Prüfen Sie das Antworttelegramm des Antriebs.
16#8181	Datentypfehler	Wählen Sie einen geeigneten Datentyp. Zulässig sind: • Real • Wort • Doppelwort
16#8182	Datentypfehler: Auf die Anforderung "Wort" darf nicht "Doppelwort" oder "Real" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8183	Datentypfehler: Auf die Anforderung "Doppelwort" oder "Real" darf nicht "Wort" zurückgemeldet werden.	Prüfen Sie das Antworttelegramm des Antriebs.
16#8184	Checksummenfehler in der Antwort des Antriebs	Prüfen Sie den Antrieb und die Kommunikationsverbindung.
16#8185	Adressierungsfehler	Zulässiger Antriebsadressbereich: 1 bis 16
16#8186	Sollwertfehler	Zulässiger Sollwertbereich: -200 % bis +200 %
16#8187	Falsche Antriebsnummer zurückgemeldet	Prüfen Sie das Antworttelegramm des Antriebs.
16#8188	Unzulässige PZD-Länge	Zulässige PZD-Längen: 2, 4, 6, 8 Worte
16#8189	Das Modul unterstützt diese Datenübertragungsgeschwindigkeit nicht.	Wählen Sie eine für das Modul zulässige Datenübertragungsgeschwindigkeit.
16#818A	Eine andere Anfrage für diesen Antrieb ist gerade aktiv.	Wiederholen Sie den Parameter-Lese- oder -Schreibauftrag zu einem späteren Zeitpunkt.
16#818B	Der Antrieb antwortet nicht.	Überprüfen Sie den Antrieb
16#818C	Der Antrieb antwortet mit einer Fehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage. Prüfen Sie, ob die Anweisungen USS_Read_Param, USS_Write_Param oder USS_Port_Scan einen Fehler gemeldet haben. Falls ja, prüfen Sie den Wert der statischen Variablen USS_DB. w_USSExtendedError der Anweisung USS_Drive_Control.
16#818D	Der Antrieb antwortet mit einer Zugriffsfehlermeldung auf eine Parameteranfrage.	Prüfen Sie das Antworttelegramm des Antriebs. Prüfen Sie die Parameteranfrage.
16#818E	Der Antrieb wurde nicht initialisiert.	Überprüfen Sie das Anwenderprogramm und stellen Sie sicher, dass die Anweisung USS_Drive_Control für diesen Antrieb aufgerufen wird.
16#8280	Negative Quittung beim Lesen des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.RDREC.STATUS oder Send_Config.RDREC.STATUS oder Receive_Config.RDREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie in der Beschreibung des SFB RDREC.

Fehlercode	Beschreibung	Abhilfe
16#8281	Negative Quittung beim Schreiben des Moduls	Überprüfen Sie die Eingabe am Parameter PORT. Genauere Informationen zur Fehlerursache finden Sie in den statischen Parametern Port_Config.WRREC.STATUS oder Send_Config.WRREC.STATUS oder Receive_Config.WRREC.STATUS oder Send_P2P.RDREC.STATUS oder Receive_P2P.RDREC.STATUS, sowie der Beschreibung des SFB WRREC.
Fehlercodes Allgemein		
16#8FFF	Das Modul ist durch ein Reset kurzzeitig nicht betriebsbereit.	Anfrage wiederholen.

[1] Nur bei Anweisungen für S7-300/400 CPUs

PtP-Kopplung CP 340

P_SEND: Daten senden

Beschreibung

Die Anweisung P_SEND überträgt einen Datenblock von einem Datenbaustein, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN, zum CP 340. Die Anweisung P_SEND wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung P_SEND kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 340 abgebrochen und die Anweisung P_SEND in den Grundzustand versetzt. Daten, die der CP 340 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 340 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im Parameter STATUS (Seite 6632) wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_SEND ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 340 Die Basisadresse wird aus STEP 7 entnommen
DB_NO	INPUT	INT	Datenbausteinnummer Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8190 Sendedaten ab Datenbyte
LEN	INPUT	INT	Datenlänge 1 ≤ LEN ≤ 1024, Angabe in Anzahl Byte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6632) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6632) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung P_SEND arbeitet mit einem Instanz-DB I_SEND zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

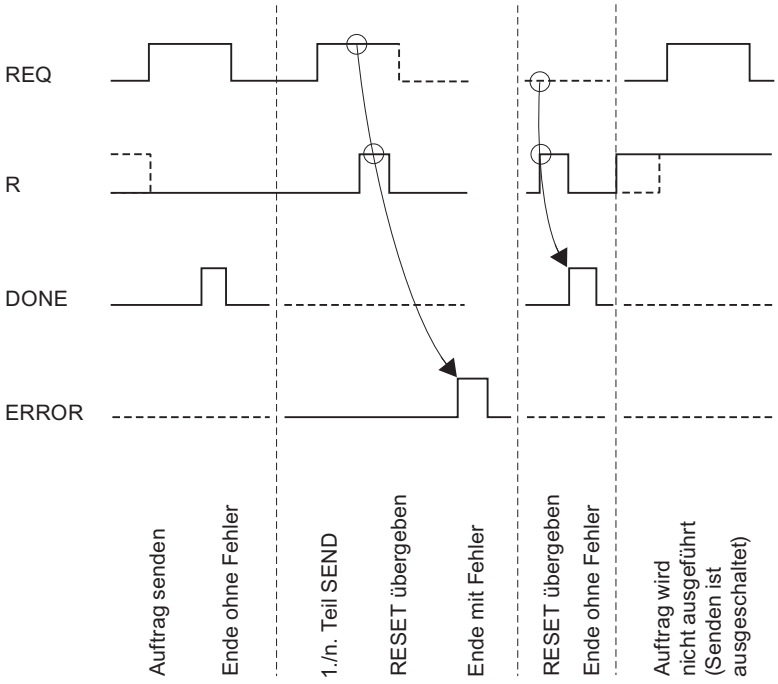

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regeln

Hinweis

Die Anweisung P_SEND hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN vom CP 340 bearbeitet werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_SEND abgeschlossen sein. Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zum CP 340 übertragen.

P_RCV: Daten empfangen

Beschreibung

Die Anweisung P_RCV überträgt Daten von dem CP 340 zu einem S7-Datenbereich, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN. Die Anweisung P_RCV wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von der CP 340 zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R abgebrochen werden. (Da es sporadisch vorkommen kann, dass nach Setzen des Parameters EN_R auf "1" die Übertagung nicht mehr anläuft, sollten Sie zum Abbrechen einer laufenden Übertragung parallel zum Setzen des EN_R auf "0" den Parameter R auf "1" setzen.) Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Übertragung über mehrere Aufrufe (Programmzyklen) laufen.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung P_RCV in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht. Bei erneutem Signalzustand "0" wird das abgebrochene Telegramm erneut, von Anfang an, empfangen.

Mit dem Parameter LADDR wird der anzusprechende CP 340 ausgewählt.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). ERROR zeigt einen aufgetretenen Fehler an. Im STATUS (Seite 6632) wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. NDR und ERROR/STATUS werden auch bei RESET des P_RCV ausgegeben (Parameter LEN == 16#00). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten lesen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 340 Die Basisadresse wird aus STEP 7 entnommen
DB_NO	INPUT	INT	Datenbausteinnummer Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt

Parameter	Deklaration	Datentyp	Beschreibung
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8190 Empfangsdaten ab Datenbyte
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6632) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 1 ≤ LEN ≤ 1024, Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6632) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung P_RCV arbeitet mit einem Instanz-DB I_RCV zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0E, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter NDR, LEN und ERROR je nach Eingangsbeschaltung von EN_R und R.

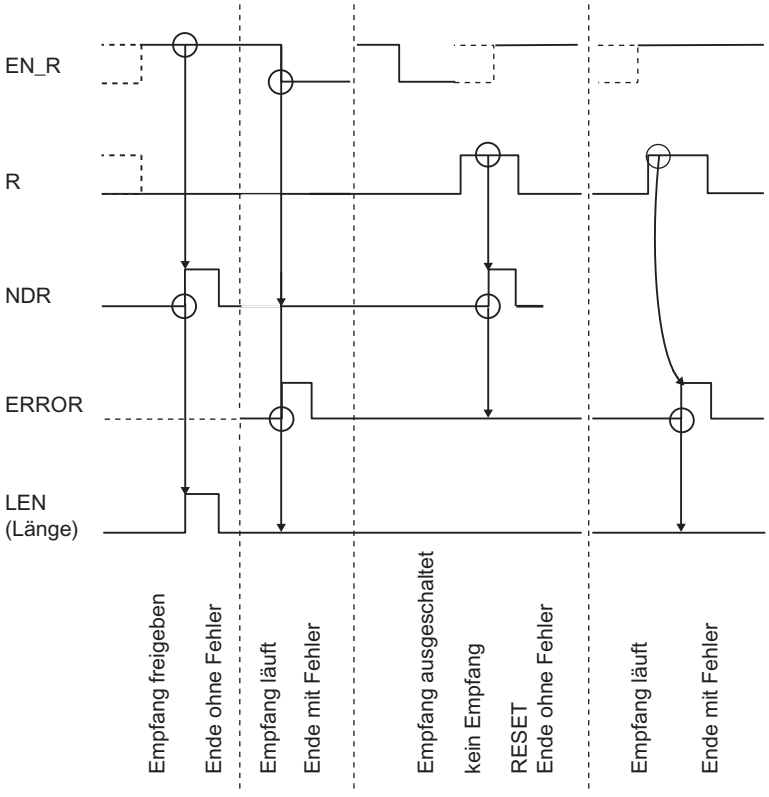

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Regeln

Hinweis

Die Anweisung P_RCV hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor nach einem Zustandsübergang der CPU von STOP nach RUN ein Auftrag vom CP 340 empfangen werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_RCV abgeschlossen sein.

P_PRINT: Meldetext mit bis zu 4 Variablen auf Drucker ausgeben

Beschreibung

Für die Druckerausgabe von Meldetexten steht Ihnen die Anweisung P_PRINT zur Verfügung. Die Anweisung P_PRINT überträgt z. B. eine Prozessmeldung zum CP 340. Der CP 340 protokolliert die Prozessmeldung auf dem angeschlossenen Drucker.

Die Anweisung P_PRINT überträgt einen Meldetext mit bis zu vier Variablen zum CP 340. Die Anweisung P_PRINT wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Über die Parameter DB_NO und DBB_NO sind die Pointer (Zeiger auf Datenbausteine) für den Formatstring und die vier Variablen erreichbar. Die Pointer müssen lückenlos, in einer bestimmten Reihenfolge, im parametrierten Datenbaustein (Pointer-DB) abgelegt sein (siehe Bild "Pointer-DB").

Mit einer positiven Flanke am Eingang REQ wird die Übertragung des Meldetextes angestoßen. Zuerst wird der Formatstring des Meldetextes übertragen. Danach werden die Variablen 1 bis 4 übergeben.

Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung P_PRINT kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 340 abgebrochen und die Anweisung P_PRINT in den Grundzustand versetzt. Daten, die der CP 340 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden von Druckaufträgen ausgeschaltet.

An dem Parameter LADDR wird die Adresse des anzusprechenden CP 340 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_PRINT ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Druckerausgabe gesperrt.

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 340 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Zeiger auf Pointer-DB: CPU-spezifisch, Null ist nicht erlaubt (Die Pointer auf Variablen und Formatstring sind im Pointer-DB in fester Reihenfolge hinterlegt (siehe vorheriges Bild.)
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8162 Pointer ab Datenbyte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6632) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6632) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich, Instanz-DB

Die Anweisung P_PRINT arbeitet mit einem Instanz-DB I_PRINT zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen.

Belegung im Datenbereich, Pointer-DB

Die Anweisung P_PRINT greift über die Parameter DB_NO und DBB_NO auf einen Pointer-DB zu, in dem die Pointer (Zeiger) auf die Datenbausteine mit den Meldetexten und Variablen in fester Reihenfolge hinterlegt sind. Der Pointer-DB muss von Ihnen erzeugt werden.

Das Bild zeigt den Aufbau des Pointer-DBs, der durch die Parameter DB_NO und DBB_NO der Anweisung P_PRINT angesprochen wird.

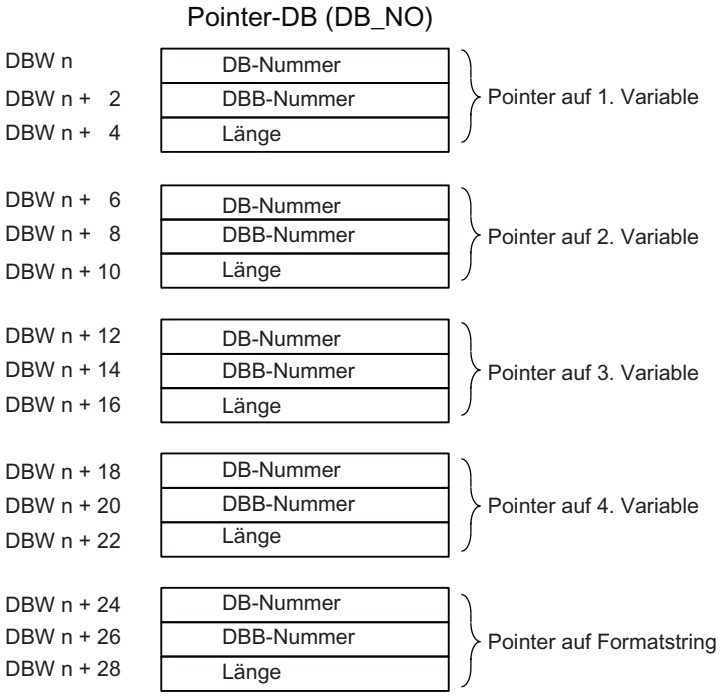

Zulässige DB-Nummer

Die zulässigen DB-Nummern sind CPU-spezifisch. Wird bei "Pointer auf Variable" als DB-Nummer der Wert 16#00 angegeben, dann wird diese Variable als nicht vorhanden interpretiert und der Pointer auf die nächste Variable oder auf den Formatstring gesetzt.

Ist bei "Pointer auf Formatstring" die DB-Nummer gleich dem Wert 16#00, dann wird der Druckauftrag abgebrochen und die Ereignisnummer 16#1E43 am Parameterausgang STATUS der Anweisung P_PRINT angezeigt.

Zulässige DBB-Nummer

Ab der parametrierten DBB-Nummer steht die Variable oder der Formatstring. Die Variablen dürfen eine maximale Länge von 32 Byte haben, der Formatstring eine maximale Länge von 150 Byte.

Wird die maximale Länge überschritten, dann wird der Druckauftrag abgebrochen und die Ereignisnummer 16#1E41 am Parameterausgang STATUS der Anweisung P_PRINT angezeigt.

Zulässige Länge

Die Längenangabe im Pointer-DB ist für die jeweilige Darstellungsart (Datentypen) sowie abhängig von der verwendeten Präzision anzupassen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

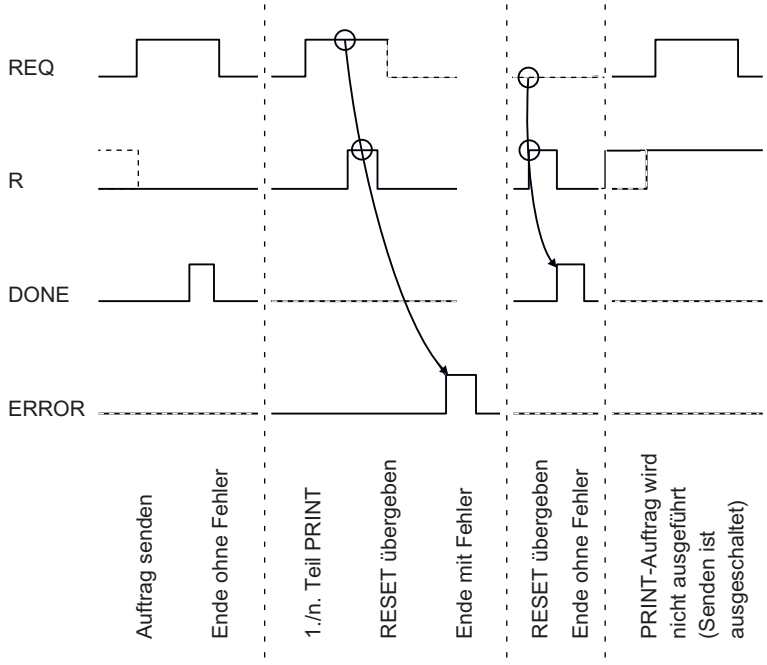

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Er muss nicht während der gesamten Übertragung Signalzustand "1" haben.

Regeln

Hinweis

Die Anweisung P_PRINT hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN vom CP 340 bearbeitet werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_PRINT abgeschlossen sein. Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zum CP 340 übertragen.

P_RESET: Empfangspuffer löschen

Beschreibung

Die Anweisung P_RESET löscht den kompletten Empfangspuffer des CP 340. Alle gespeicherten Telegramme werden verworfen. Ein zum Zeitpunkt des Aufrufs der Anweisung P_RESET eingehendes Telegramm wird gespeichert.

Mit einer positiven Flanke am Eingang REQ wird die Anweisung aktiviert. Der Auftrag kann über mehrere Aufrufe (Programmzyklen) laufen.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 340 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im Parameter STATUS (Seite 6632) wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
LADDR	INPUT	INT	Basisadresse des CP 340 Die Basisadresse wird aus STEP 7 entnommen.
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6632) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6632) die Fehlerinformation
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung P_RESET arbeitet mit einem Instanz-DB I_P_RESET zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ.

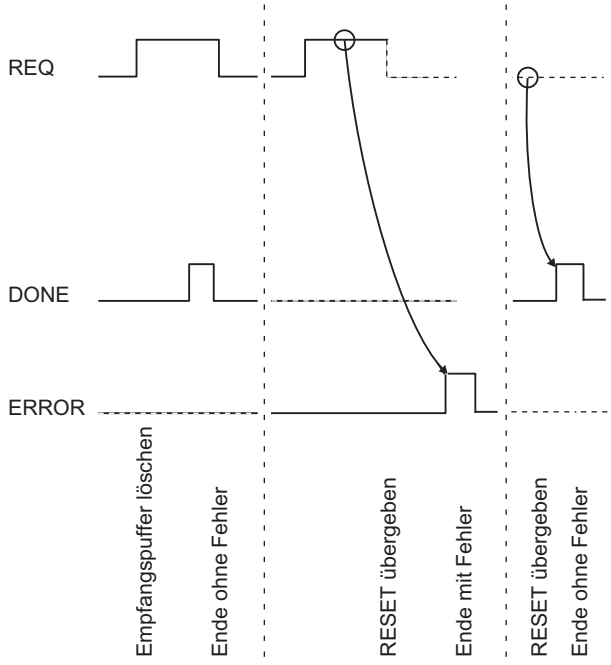

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regel

Hinweis

Die Anweisung P_RESET hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

V24_STAT_340 / V24_STAT: Begleitsignale an der RS232C-Schnittstelle lesen

Beschreibung

Die Anweisung V24_STAT_340 (CP 340) / V24_STAT (CP 341) liest vom CP 34x die RS 232C-Begleitsignale und stellt sie dem Anwender an den Bausteinparametern zur Verfügung. Die Anweisung V24_STAT_340 / V24_STAT wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Die RS 232C-Begleitsignale werden mit jedem Aufruf der Anweisung aktualisiert (zyklisches Pollen). Der CP 34x aktualisiert den Status der Ein-/Ausgänge in einem Zeitraster von 20 ms. Die Ein-/Ausgänge werden unabhängig davon laufend aktualisiert.

Das Binärergebnis BIE wird nicht beeinflusst. Die Anweisung gibt keine Fehlermeldung aus.

Mit dem Parameter LADDR wird der anzusprechende CP 34x ausgewählt.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 34x Die Basisadresse wird aus STEP 7 entnommen.
DTR_OUT	OUTPUT	BOOL	**D**ata **t**erminal **r**eady, CP 34x betriebsbereit (CP 34x-Ausgang)
DSR_IN	OUTPUT	BOOL	**D**ata **s**et **r**eady, Kommunkationspartner betriebsbereit (CP 34x-Eingang)
RTS_OUT	OUTPUT	BOOL	**R**equest **t**o **s**end, CP 34x sendebereit (CP 34x-Ausgang)
CTS_IN	OUTPUT	BOOL	**C**lear **t**o **s**end, Kommunikationspartner kann Daten vom CP 34x empfangen (Anwort auf RTS = ON des CP 34x) (CP 34x-Eingang)
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, Empfangssignalpegel (CP 34x-Eingang)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, Rufzeichen (CP 34x-Eingang)

Belegung im Datenbereich

Die Anweisung V24_STAT_340 / V24_STAT belegt keine Datenbereiche.

Regel

Hinweis

Zum Erkennen eines Signalwechsels ist eine Mindestimpulsdauer notwendig. Ausschlaggebende Größen sind die CPU-Zykluszeit, die Aktualisierungszeit auf dem CP 34x und die Reaktionszeit des Kommunikationspartners.

V24_SET_340 / V24_SET: Begleitsignale an der RS232C-Schnittstelle schreiben

Beschreibung

Der Anwender kann über die Parametereingänge der Anweisung V24_SET_340 (CP 340) / V24_SET (CP 341) die entsprechenden Schnittstellenausgänge setzen oder rücksetzen. Die Anweisung V24_SET_340 / V24_SET wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Das Binärergebnis BIE wird nicht beeinflusst. Die Anweisung gibt keine Fehlermeldung aus.

Mit dem Parameter LADDR wird der anzusprechende CP 34x ausgewählt.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 34x Die Basisadresse wird aus STEP 7 entnommen.
RTS	INPUT	BOOL	**R**equest **t**o **s**end, CP 34x sendebereit (CP 34x-Ausgang steuern)
DTR	INPUT	BOOL	**D**ata **t**erminal **r**eady, CP 34x betriebsbereit (CP 34x-Ausgang steuern)

Belegung im Datenbereich

Die Anweisung V24_SET_340 / V24_SET belegt keine Datenbereiche.

Parameter STATUS

Ereignisklassen

Ereignisklasse	Bedeutung
5	Fehler bei Bearbeitung eines CPU-Auftrags
7	Sendefehler
8	Empfangsfehler
30 ($1E_H$)	Fehler bei Kommunikation zwischen CP und CPU

Parameter STATUS

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0502	Auftrag ist in diesem Betriebszustand des CP nicht erlaubt (z. B. Geräteschnittstelle nicht parametriert).	Werten Sie den Diagnosealarm aus, und beheben Sie den Fehler entsprechend.
0505	**Nur bei Drucker-Treiber:** Systemdatenbaustein mit Meldetexten auf CP nicht vorhanden	Projektieren Sie die Meldetexte mit der Parametriersoftware, und führen Sie einen Neustart durch.
0506	**Nur bei Drucker-Treiber:** Meldetext nicht vorhanden	Projektieren Sie die Meldetexte mit der Parametriersoftware, und führen Sie einen Neustart durch.
0507	**Nur bei Drucker-Treiber:** Meldetext zu lang	Ändern Sie den Meldetext auf maximal 150 Zeichen (auf maximal 250 Zeichen bei der Verwendung von Variablen)
0508	**Nur bei Drucker-Treiber:** Zu viele Konvertierungsanweisungen	Sie haben mehr Konvertierungsanweisungen als Variablen projektiert. Die Konvertierungsanweisungen ohne dazugehörige Variable werden ignoriert.
0509	**Nur bei Drucker-Treiber:** Zu viele Variablen	Sie haben mehr Variablen als Konvertierungsanweisungen projektiert. Die Variablen ohne Konvertierungsanweisung werden nicht ausgegeben.
050A	**Nur bei Drucker-Treiber:** Unbekannte Konvertierungsanweisung	Überprüfen Sie die Konvertierungsanweisung. Nicht definierte oder nicht unterstützte Konvertierungsanweisungen werden im Ausdruck durch ****** ersetzt.
050B	**Nur bei Drucker-Treiber:** Unbekannte Steueranweisung	Überprüfen Sie die Steueranweisung. Nicht definierte oder nicht unterstützte Steueranweisungen werden ignoriert. Die Steueranweisung wird auch nicht als Text ausgegeben.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
050C	**Nur bei Drucker-Treiber:** Konvertierungsanweisung nicht ausführbar	Überprüfen Sie die Konvertierungsanweisung. Nicht ausführbare Konvertierungsanweisungen werden im Ausdruck entsprechend der definierten Breite und dem gültigen Rest der Konvertierungsanweisung bzw. der Standarddarstellung mit *–Zeichen ausgegeben.
050D	**Nur bei Drucker-Treiber:** Breite in Konvertierungsanweisung zu gering oder zu groß	Korrigieren Sie die angegebene Breite der Variable in der Konvertierungsanweisung anhand der maximal möglichen Zeichenanzahl der Variable bei textorientierten Darstellungsarten (A, C, D, S, T, Y, Z). Im Ausdruck wird nur die durch die Breite angegebene Anzahl Zeichen ausgegeben, der Rest wird abgeschnitten. Ansonsten werden entsprechend der Breite *–Zeichen ausgegeben.
050E	**Nur bei 3964(R) und ASCII-Treiber:** Ungültige Telegrammlänge	Die Telegrammlänge ist > 1024 Bytes. Der Rest des Telegramms (> 1024 Bytes) wird vom CP 340 noch empfangen und damit der erste Teil des Telegramms verworfen. Wählen Sie eine kleinere Telegrammlänge.
051B	**Nur bei Drucker-Treiber:** Präzision ungültig	Korrigieren Sie die angegebene Präzision in der Konvertierungsanweisung. Die Präzision wird immer durch einen Punkt eingeleitet zur Kennzeichnung und Abgrenzung zur Breite (z. B.: ".2" zur Ausgabe von Dezimalpunkt und 2 Nachkommastellen). Die Präzision ist nur bei den Darstellungsarten F, R, A und D relevant. Ansonsten wird sie ignoriert.
051C	**Nur bei Drucker-Treiber:** Variable ungültig (Variablenlänge falsch/falscher Typ)	Korrigieren Sie die angegebene Variable. Die bei der jeweiligen Darstellungsart möglichen Datentypen können Sie der entsprechenden Tabelle entnehmen.
051E	**Nur bei Drucker-Treiber:** Die mit diesem Auftrag gesendeten "Zeilenendesequenzen" (i.e.: $R / $L / $N) passen nicht (mehr) auf die (angefangene) Seite	Vergrößern Sie Ihre Seitenlänge, reduzieren Sie die Anzahl der Zeilen (bzw. der Zeilenvorschübe) oder verteilen Sie Ihren Ausdruck auf mehrere Seiten
0701	**Nur bei 3964(R):** Senden der ersten Wiederholung: • Beim Senden des Telegramms wurde ein Fehler erkannt oder • der Partner forderte durch ein negatives Quittungszeichen (NAK) eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der maximalen Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehlernummer gemeldet, die den Fehler beschreibt, der zuerst auftrat.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0702	**Nur bei 3964(R):** Fehler beim Verbindungsaufbau: Nachdem STX gesendet wurde, wurde NAK oder ein beliebiges Zeichen (außer DLE oder STX) empfangen.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0703	**Nur bei 3964(R):** Quittungsverzugszeit (QVZ) überschritten: Nach Senden von STX kam keine Antwort vom Partner innerhalb der Quittungsverzugszeit.	Partnergerät ist zu langsam oder nicht empfangsbereit, oder es liegt z. B ein Bruch der Sendeleitung vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0704	**Nur bei 3964(R):** Abbruch durch Partner: Während des laufenden Sendebetriebes wurden vom Partner ein oder mehrere Zeichen empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0705	**Nur bei 3964(R):** Negative Quittung während Senden	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0706	**Nur bei 3964(R):** Fehler bei Verbindungsende: • Das Telegramm wurde vom Partner am Ende mit NAK oder einem beliebigen Zeichen (außer DLE) abgelehnt oder • das Quittungszeichen (DLE) wurde zu früh empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0707	**Nur bei 3964(R):** Quittungsverzugszeit am Verbindungsende/Antwortüberwachungszeit nach Sendetelegramm überschritten: Nach Verbindungsabbau mit DLE ETX kam innerhalb der QVZ keine Antwort vom Partner.	Partnergerät ist zu langsam oder gestört. Dies ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0708	**Nur bei ASCII–Treiber und Drucker–Treiber:** Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
070B	**Nur bei 3964(R):** Initialisierungskonflikt ist nicht lösbar, weil beide Partner hochprior eingestellt sind.	Ändern Sie die Parametrierung.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
070C	**Nur bei 3964(R):** Initialisierungskonflikt ist nicht lösbar, weil beide Partner niederprior eingestellt sind.	Ändern Sie die Parametrierung.
0801	**Nur bei 3964(R):** Erwarten der ersten Wiederholung: Beim Empfangen eines Telegramms wurde ein Fehler erkannt und der CP forderte durch eine negative Quittierung (NAK) beim Partner eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der maximalen Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehlernummer gemeldet, die den Fehler beschreibt, der zuerst auftrat.
0802	**Nur bei 3964(R):** Fehler beim Verbindungsaufbau: • In Ruhestellung wurden ein oder mehrere beliebige Zeichen (außer NAK oder STX) empfangen oder • nach einem empfangenen STX wurden vom Partner weitere Zeichen gesendet, ohne die Antwort DLE abzuwarten. Nach Netz–EIN des Partners: • während der Partner eingeschaltet wird, empfängt der CP ein undefiniertes Zeichen.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0805	**Nur bei 3964(R):** Logischer Fehler während des Empfangs: Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Fehlverhalten des Partnergeräts mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0806	**Zeichenverzugszeit (ZVZ) überschritten:** • Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der ZVZ empfangen oder **Nur bei 3964(R):** • 1. Zeichen nach Senden von DLE beim Verbindungsaufbau wurde nicht innerhalb der ZVZ empfangen.	Partnergerät ist zu langsam oder gestört. Dies mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0807	**Telegrammlänge unzulässig:** Es wurde ein Telegramm mit der Länge 0 empfangen.	Der Empfang eines Telegramms mit Länge 0 ist kein Fehler. Überprüfen Sie, warum der Kommunikationspartner Telegramme ohne Nutzdaten sendet.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0808	**Nur bei 3964(R):** Fehler beim Blockprüfzeichen BCC: Der intern gebildete Wert des BCC stimmt nicht mit dem vom Partner am Verbindungsende empfangenen BCC überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0809	**Nur bei 3964(R):** Die Wiederholanzahl muss gleich eingestellt sein.	Parametrieren Sie beim Kommunikationspartner die gleiche Blockwartezeit wie am CP 340. Fehlverhalten des Kommunikationspartners ggf. mit Schnittstellengerät (FOXPG) nachweisen, das in die Übertragungsleitung eingeschaltet wird.
080A	Ein freier Empfangspuffer ist nicht vorhanden: Beim Empfang stand kein leerer Empfangspuffer zur Verfügung.	Die Anweisung P_RCV muss häufiger aufgerufen werden.
080C	Übertragungsfehler: • Ein Übertragungsfehler (Paritätsfehler, Stopbitfehler, Überlauffehler) wurde erkannt. **Nur bei 3964(R):** • Falls dies während des Sende– oder Empfangsbetriebes auftritt, werden Wiederholungen gestartet. • Wird in Ruhestellung ein gestörtes Zeichen empfangen, wird der Fehler sofort gemeldet, damit Störeinflüsse auf der Übertragungsleitung frühzeitig erkannt werden können. • Sind die SF–LED (rot) und die RxD–LED (grün) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Störungen auf der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind.
080D	BREAK: Empfangsleitung zum Partner ist unterbrochen.	Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
0810	**Nur bei ASCII-Treiber:** Paritätsfehler: Sind die SF-LED (rot) und die RxD-LED (grün) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0811	**Nur bei ASCII-Treiber:** Zeichenrahmenfehler: Sind die SF-LED (rot) und die RxD-LED (grün) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0812	**Nur bei ASCII-Treiber:** Nachdem der CP XOFF gesendet hat oder CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder entsorgen Sie den CP schneller.
0818	**Nur bei ASCII-Treiber:** DSR = OFF bzw. CTS = OFF	Vor oder während eines Sendevorgangs sind die Signale DSR bzw. CTS vom Partner auf "OFF" geschaltet worden. Überprüfen Sie die Steuerung der RS 232C-Begleitsignale beim Partner.
1E0D	Auftragsabbruch wegen Neustart, Wiederanlauf oder Reset	
1E0E	Statischer Fehler bei Aufruf des SFC RD_REC bzw. SFB RDREC. Der Returnwert RET_VAL des SFCs / SFBs wird Ihnen in der Variablen SFCERR bzw. SFCSTATUS im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR bzw. SFCSTATUS aus dem Instanz-DB.
1E0F	Statischer Fehler bei Aufruf des SFC WR_REC bzw. SFB RDREC. Der Returnwert RET_VAL des SFCs / SFBs wird Ihnen in der Variablen SFCERR bzw. SFCSTATUS im Instanz–DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR bzw. SFCSTATUS aus dem Instanz–DB.
1E41	Anzahl der am Parameter LEN der FBs angegebenen Bytes unzulässig	Halten Sie den Wertebereich von 1 bis 1024 Byte ein.
1E41	Anweisung P_PRINT: Anzahl der im Pointer–DB unter Länge angegebenen Bytes für Variable oder Formatstring ist unzulässig.	Halten Sie die zulässigen Längen ein: 32 Byte für Variablen, 150 Byte für Formatstring
1E43	Anweisung P_PRINT: Kein Pointer für den Formatstring vorhanden.	Geben Sie im Pointer–DB die Datenbaustein–Nr. und die Datenwort–Nr. für den Formatstring an.

Hinweis

Eine Fehlermeldung wird nur dann ausgegeben, wenn gleichzeitig das Bit ERROR (Auftrag abgebrochen mit Fehler) gesetzt ist. In jedem anderen Fall ist das STATUS-Wort Null.

Variable SFCERR bzw. SFCSTATUS aufrufen

Nähere Informationen zum aufgetretenen Fehler 14 (1E0EH) und 15 (1E0FH) der Ereignisklasse 30 erhalten Sie über die Variable SFCERR bzw. SFCSTATUS.

Die Variable SFCERR bzw. SFCSTATUS können Sie nur über einen symbolischen Zugriff auf den Instanz–DB der entsprechenden Anweisung laden.

Fehlermeldungen, die in der Variablen SFCERR eingetragen werden finden Sie bei den Systemfunktionen SFC 58 unter "WR_REC: Datensatz in Peripherie schreiben" und SFC 59 unter "RD_REC: Datensatz von Peripherie lesen".

Fehlermeldungen, die in der Variablen SFCSTATUS eingetragen werden finden Sie bei den Systemfunktionen SFB 52 unter "RDREC: Datensatz lesen" und SFB 53 unter "WRREC: Datensatz schreiben".

Siehe auch

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

PtP-Kopplung CP 341

P_SND_RK: Daten senden oder holen

P_SND_RK

Die Anweisung P_SND_RK unterscheidet folgende Einsatzfälle:

- P_SND_RK: Daten senden mit 3964(R) oder ASCII-Treiber (Seite 6638)
- P_SND_RK: Daten senden mit RK 512 (Seite 6641)
- P_SND_RK: Daten holen mit RK 512 (Seite 6645)

P_SND_RK: Daten senden mit 3964(R) oder ASCII-Treiber

Beschreibung

Die Anweisung P_SND_RK überträgt einen Datenblock von einem Datenbaustein, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN, zum CP 341. Die Anweisung P_SND_RK wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung P_SND_RK kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 341 abgebrochen und die Anweisung P_SND_RK in den Grundzustand versetzt. Daten, die der CP 341 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_SND_RK ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8190 Sendedaten ab Datenbyte
LEN	INPUT	INT	Datenlänge 1 ≤ LEN ≤ 4096, Angabe in Anzahl Byte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Hinweis

Die Parameter R_CPU_NO, R_TYP, R_NO, R_OFFSET, R_CF_BYT und R_CF_BIT haben bei der Prozedur 3964(R) und beim ASCII-Treiber keine Bedeutung und brauchen nicht versorgt werden. Der Parameter SF braucht ebenfalls nicht versorgt werden, da 'S' für Senden voreingestellt ist.

Belegung im Datenbereich

Die Anweisung P_SND_RK arbeitet mit einem Instanz-DB I_SND_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus den Variablen SFCERR bzw. SFCSTATUS entnehmen. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

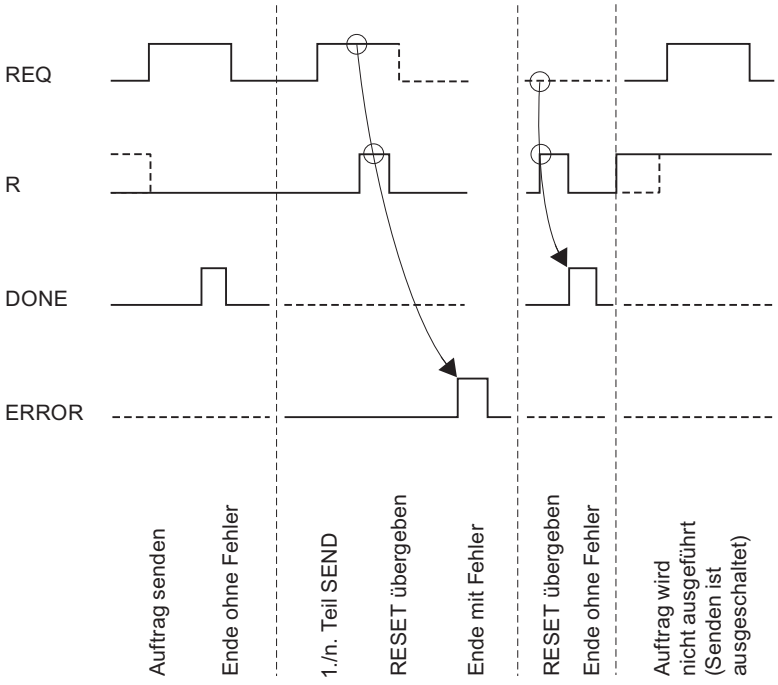

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regeln

Hinweis

Die Anweisung P_SND_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN vom CP 341 bearbeitet werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_SND_RK abgeschlossen sein. Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zum CP 341 übertragen.

P_SND_RK: Daten senden mit RK 512

Beschreibung

Die Anweisung P_SND_RK kann durch Einstellung des Parameters SF = 'S' Daten von einem S7-Datenbereich zu einem CP 341 übertragen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge (LEN) kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Als Quelle für zu sendende Daten wird nur der Bereich der Datenbausteine zugelassen. Die Quelle ist vollständig spezifiziert durch Angabe der Datenbausteinnummer (DB_NO) und des Offsets (DBB_NO) des ersten zu sendenden Datenbytes in diesem Datenbaustein.

Als Zielbereiche sind als Datentypen (R_TYP) Datenbausteine (DB) und erweiterte Datenbausteine (DX) zugelassen. Das Ziel ist vollständig spezifiziert durch die CPU-Nummer (R_CPU_NO, nur relevant bei Mehrprozessorkommunikation), den Datentyp (R_TYP: DB oder DX), die Datenbausteinnummer (R_NO) und den Offset (R_OFFSET), an den das erste Byte geschrieben werden soll.

Mit R_CF_BYT und R_CF_BIT wird das Koppelmerkerbyte und -bit auf der Partner-CPU festgelegt.

Die Anweisung P_SND_RK kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 341 abgebrochen und der Anweisung P_SND_RK in den Grundzustand versetzt. Daten, die der CP 341 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_SND_RK ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
SF	INPUT	CHAR	Auswahl für Daten senden oder Daten holen SF = 'S' (Senden) Defaultwert: 'S'
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt. Defaultwert: 0
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer von Quelle Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer von Quelle 0 ≤ DBB_NO ≤ 8190 Sendedaten ab Datenbyte
LEN	INPUT	INT	Datenlänge des zu sendenden Telegramms 1 ≤ LEN ≤ 4096, Angabe in Anzahl Byte, nur geradzahlige Werte sinnvoll
R_CPU_NO	INPUT	INT	CPU-Nr. der Partner-CPU 0 ≤ R_CPU_NO ≤ 4, nur bei Mehrprozessorbetrieb, Defaultwert: 1
R_TYP	INPUT	CHAR	Adresstyp auf Partner-CPU 'D': Datenbaustein 'X': erweiterter Datenbaustein
R_NO	INPUT	INT	Datenbausteinnummer auf Partner-CPU 0 ≤ R_NO ≤ 255
R_OFFSET	INPUT	INT	Datenbytenummer auf Partner-CPU 0 ≤ R_OFFSET ≤ 510,nur geradzahlige Werte
R_CF_BYT	INPUT	INT	Koppelmerkerbyte auf Partner-CPU 0 ≤ R_CF_BYTE ≤ 255 Defaultwert: 255 (bedeutet: ohne Koppelmerker)
R_CF_BIT	INPUT	INT	Koppelmerkerbit auf Partner-CPU 0 ≤ R_CF_BIT ≤ 7

Parameter	Deklaration	Datentyp	Beschreibung
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Angaben im Telegrammkopf

In der folgenden Tabelle sind die Angaben im Telegrammkopf des RK 512-Telegramms dargestellt.

Tabelle 4-339 Angaben im Telegrammkopf des RK 512-Telegramms für Auftrag "Daten senden"

Quelle auf Ihrem S7-Automatisierungssystem (lokale CPU)	zum Ziel, Partner-CPU	Telegrammkopf, Bytes		
		3/4 Befehlsart	5/6 Z-DBNR/Z-Offset	7/8 Anzahl in
Datenbaustein	Datenbaustein	AD	DB/DW	Wörtern
Datenbaustein	Erweiterter Datenbaustein	AD	DB/DW	Wörtern
Z-DBNR: Ziel-Datenbausteinnummer Z-Offset: Zielanfangsadresse DW: Offset in Worten				

Belegung im Datenbereich

Die Anweisung P_SND_RK arbeitet mit einem Instanz-DB I_SND_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Besonderheiten beim Daten senden

Beachten Sie die folgenden Besonderheiten beim "Daten senden":

- Mit RK 512 kann nur eine gerade Anzahl von Daten gesendet werden. Wenn Sie als Länge (LEN) eine ungerade Anzahl von Daten angeben, wird ein zusätzliches Füllbyte mit dem Wert "0" am Ende der Daten übertragen.
- Mit RK 512 kann nur ein gerader Offset angegeben werden. Bei Angabe eines ungeraden Offsets werden die Daten ab dem nächst kleineren geraden Offset beim Partner abgelegt. Beispiel: Offset ist 7, abgelegt werden die Daten ab Byte 6.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

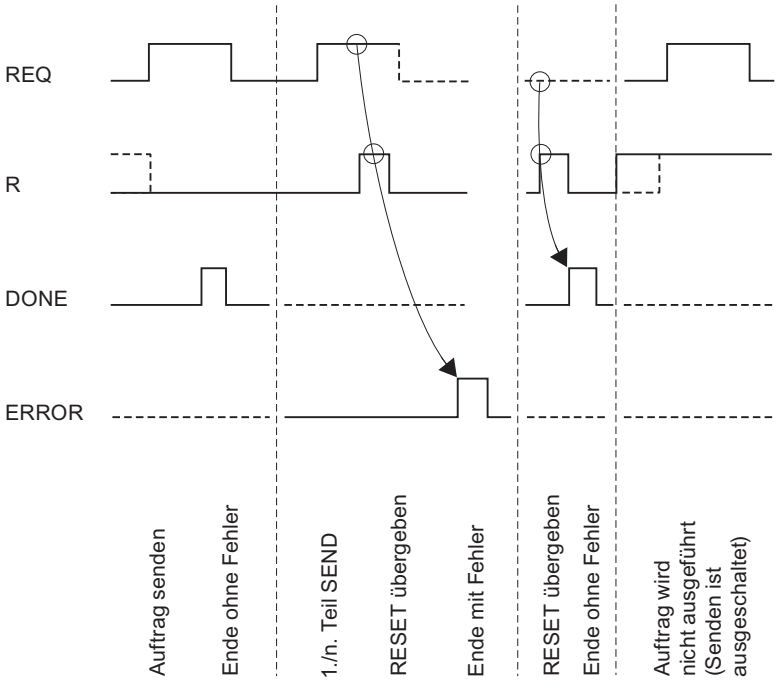

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regel

Hinweis

Die Anweisung P_SND_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

P_SND_RK: Daten holen mit RK 512

Beschreibung

Die Anweisung P_SND_RK kann durch Einstellung des Parameters SF = F Daten von einem remoten Kommunikationspartner holen und in einen S7-Datenbereich Ihres Automatisierungssystems ablegen.

Hinweis

Wenn Sie Daten von einem CP 341 holen, müssen Sie auf dem CP 341 immer eine Anweisung P_RCV_RK programmieren.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge (LEN) kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Von welchem Kommunikationspartner die Daten geholt werden, wird durch die Angabe der CPU-Nummer festgelegt (R_CPU_NO, nur relevant bei Mehrprozessorkommunikation). Als Quelle für die zu holenden Daten werden folgende Datentypen (R_TYP) zugelassen: Datenbausteine, erweiterte Datenbausteine, Merker, Eingänge, Ausgänge, Zähler und Zeiten. Die Quelle ist vollständig spezifiziert durch Angabe des Datentyps (R_TYP), der Datenbausteinnummer (R_NO, nur relevant bei Datenbausteinen und erweiterten Datenbausteinen) und des Offsets (R_OFFSET) des ersten zu sendenden Datenbytes in diesem Bereich.

Mit R_CF_BYT und R_CF_BIT wird das Koppelmerkerbyte und -bit auf der Partner-CPU festgelegt.

Als Zielbereiche sind nur Datenbausteine (DB) zugelassen. Das Ziel ist vollständig spezifiziert durch die Datenbausteinnummer (DB_NO) und den Offset (DBB_NO), an den das erste Byte geschrieben werden soll.

Die Anweisung P_SND_RK kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung vom CP 341 abgebrochen und die Anweisung P_SND_RK in den Grundzustand versetzt. Steht statisch Signalzustand "1" am Eingang R an, so ist das Holen ausgeschaltet.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_SND_RK ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
SF	INPUT	CHAR	Auswahl für Daten senden oder Daten holen SF = 'F' (Fetch = Holen)Defaultwert: 'S' (Senden)
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Holen gesperrt. Defaultwert: 0
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer vom Ziel Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer vom Ziel 0 ≤ DBB_NO ≤ 8190 Sendedaten ab Datenbyte
LEN	INPUT	INT	Datenlänge des zu holenden Telegramms 1 ≤ LEN ≤ 4096, Angabe in Anzahl Byte[1]
R_CPU_NO	INPUT	INT	CPU-Nr. der Partner-CPU 0 ≤ R_CPU_NO ≤ 4, nur bei Mehrprozessorbetrieb, Defaultwert: 1
R_TYP	INPUT	CHAR	Adresstyp auf Partner-CPU 'D': Datenbaustein 'X': Erweiterter Datenbaustein 'M': Merker 'E': Eingänge 'A': Ausgänge 'Z': Zähler 'T': Zeiten
R_NO	INPUT	INT	Datenbausteinnummer auf Partner-CPU 0 ≤ R_NO ≤ 255
R_OFFSET	INPUT	INT	Datenbytenummer auf Partner-CPU
R_CF_BYT	INPUT	INT	Koppelmerkerbyte auf Partner-CPU 0 ≤ CF_BYTE ≤ 255 Defaultwert: 255 (bedeutet: ohne Koppelmerker)
R_CF_BIT	INPUT	INT	Koppelmerkerbit auf Partner-CPU 0 ≤ CF_BIT ≤ 7
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Parameter R_TYP, R_NO und R_OFFSET

Quelle auf Partner-CPU	R_TYP	R_NO	R_OFFSET (in Byte) (Dieser Wert ist durch die Partner-CPU vorgegeben.)
Datenbaustein	'D'	0 - 255	0 - 510 (nur geradzahlige Werte sinnvoll!)
Erweiterter Datenbaustein	'X'	0 - 255	0 - 510 (nur geradzahlige Werte sinnvoll!)
Merker	'M'	irrelevant	0 - 255
Eingänge	'E'	irrelevant	0 - 255
Ausgänge	'A'	irrelevant	0 - 255
Zähler	'Z'	irrelevant	0 - 255
Zeiten	'T'	irrelevant	0 - 255

Angaben im Telegrammkopf

In der folgenden Tabelle sind die Angaben im Telegrammkopf des RK 512-Telegramms dargestellt.

Quelle auf Partner-CPU	zum Ziel, Ihr S7-Automatisierungssystem (lokale CPU)	Telegrammkopf, Bytes		
		3/4 Befehlsart	5/6 Q-DBNR/Q-Offset	7/8 Anzahl in
Datenbaustein	Datenbaustein	ED	DB/DW	Wörtern
Erweiterter Datenbaustein	Datenbaustein	EX	DB/DW	Wörtern
Merker	Datenbaustein	EM	Byteadresse	Bytes
Eingänge	Datenbaustein	EE	Byteadresse	Bytes
Ausgänge	Datenbaustein	EA	Byteadresse	Bytes
Zähler	Datenbaustein	EZ	Zählernummer	Wörtern
Zeiten	Datenbaustein	ET	Zeitnummer	Wörtern
Q-DBNR: Quell-Datenbausteinnummer Q-Offset: Quellanfangsadresse				

Belegung im Datenbereich

Die Anweisung P_SND_RK arbeitet mit einem Instanz-DB I_SND_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Besonderheiten bei (erweiterten) Datenbausteinen

Beachten Sie die folgende Besonderheiten beim "Daten holen" aus Datenbausteinen und erweiterten Datenbausteinen:

- Mit RK 512 kann nur eine gerade Anzahl von Daten geholt werden. Wenn Sie als Länge (LEN) eine ungerade Anzahl angeben, wird immer ein Byte mehr übertragen. Im Ziel-DB wird jedoch die korrekte Anzahl Daten eingetragen.
- Mit RK 512 kann nur ein gerader Offset angegeben werden. Bei Angabe eines ungeraden Offsets werden die Daten aus dem nächst kleineren geraden Offset beim Partner geholt. Beispiel: Offset ist 7, geholt werden die Daten ab Byte 6.

Besonderheiten bei Zeiten und Zählern

Wenn Sie vom Kommunikationspartner Zeiten oder Zähler holen, müssen Sie berücksichtigen, dass für jede Zeit bzw. Zähler 2 Bytes geholt werden. Wenn Sie z. B. 10 Zähler holen wollen, müssen Sie als Länge 20 eingeben.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

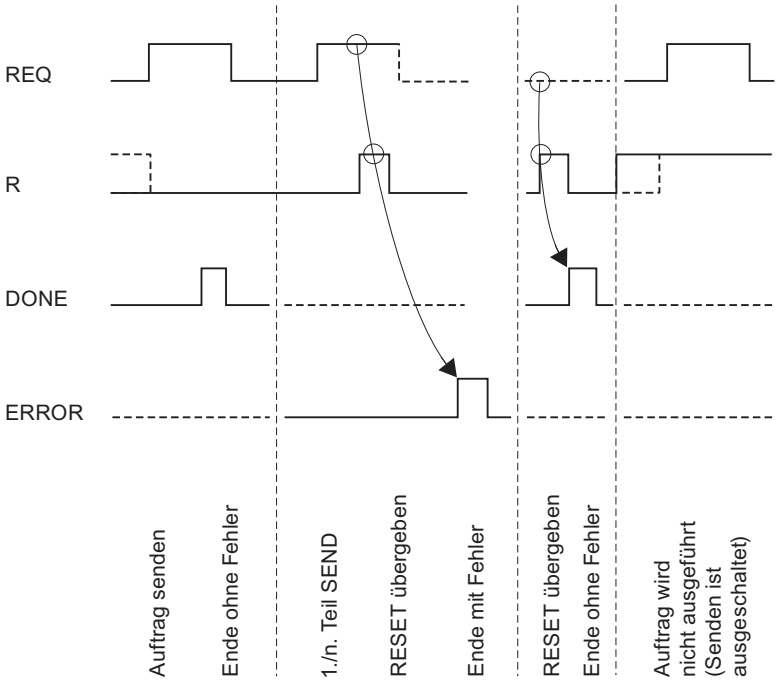

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regel

Hinweis

Die Anweisung P_SND_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

P_RCV_RK: Daten empfangen oder Daten bereitstellen

P_RCV_RK

Die Anweisung P_RCV_RK unterscheidet folgende Einsatzfälle:

- P_RCV_RK: Daten empfangen mit 3964(R) oder ASCII-Treiber (Seite 6649)
- P_RCV_RK: Daten empfangen mit RK 512 (Seite 6652)
- P_RCV_RK: Daten bereitstellen mit RK 512 (Seite 6656)

P_RCV_RK: Daten empfangen mit 3964(R) oder ASCII-Treiber

Beschreibung

Die Anweisung P_RCV_RK überträgt Daten von dem CP 341 zu einem S7-Datenbereich, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN. Die Anweisung P_RCV_RK wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von der CP 341 zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R abgebrochen werden. (Da es sporadisch vorkommen kann, dass nach Setzen des Parameters EN_R auf "1" die Übertagung nicht mehr anläuft, sollten Sie zum Abbrechen einer laufenden Übertragung parallel zum Setzen des EN_R auf "0" den Parameter R auf "1" setzen.) Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Übertragung über mehrere Aufrufe (Programmzyklen) laufen.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung P_RCV_RK in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht.

Mit dem Parameter LADDR wird der anzusprechende CP 341 ausgewählt.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert

0. NDR und ERROR/STATUS werden auch bei RESET der Anweisung P_RCV_RK ausgegeben (Parameter LEN == 16#00). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten lesen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8190 Empfangsdaten ab Datenbyte
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 1 ≤ LEN ≤ 4096, Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Hinweis

Die Parameter L_TYP, L_NO, L_OFFSET, L_CF_BYT und L_CF_BIT haben bei der Prozedur 3964(R) und beim ASCII-Treiber keine Bedeutung und brauchen nicht versorgt werden.

Belegung im Datenbereich

Die Anweisung P_RCV_RK arbeitet mit einem Instanz-DB I_RCV_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0E, können Sie einen genaueren Fehlerhinweis aus den Variablen SFCERR bzw. SFCSTATUS entnehmen. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter NDR, LEN und ERROR je nach Eingangsbeschaltung von EN_R und R.

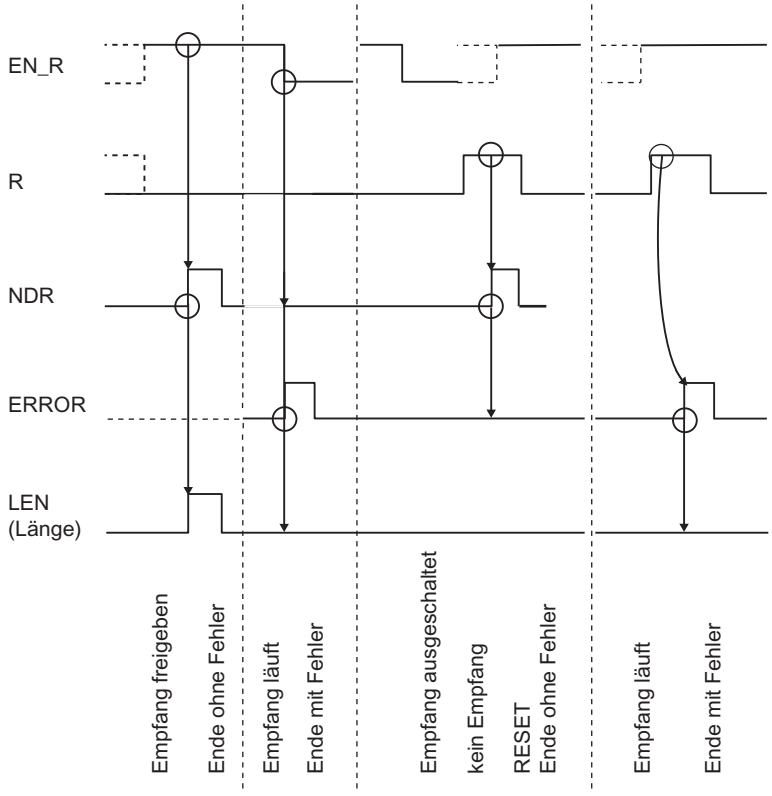

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Regeln

Hinweis

Die Anweisung P_RCV_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor nach einem Zustandsübergang der CPU von STOP nach RUN ein Auftrag vom CP 341 empfangen werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_RCV_RK abgeschlossen sein.

P_RCV_RK: Daten empfangen mit RK 512

Beschreibung

Die Anweisung P_RCV_RK überträgt Daten von dem CP 341 zu einem S7-Datenbereich. Die Anweisung P_RCV_RK wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von dem CP 341 zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R abgebrochen werden. Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Übertragung über mehrere Aufrufe (Programmzyklen) laufen.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Ist vom Kommunikationspartner als Datenziel "DB" angegeben, werden die Daten in den im RK 512-Telegrammkopf spezifizierten Datenbereich abgelegt. Dem Anwender werden mit Hilfe der Parameter (L_...) der Typ des Zielbereiches (L_TYP), die Ziel-Datenbausteinnummer (L_NO, nur relevant bei L_TYP = DB), der Offset im Zielbereich (L_OFFSET) sowie die Länge (LEN) der übertragenen Daten angezeigt. Ist vom Partner das Datenziel "DX" angegeben, werden die Daten in dem Datenbaustein (DB), der durch die Parameter DB_NO und DBB_NO spezifiziert ist, abgelegt.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung P_RCV_RK in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). Für einen Zyklus wird daraufhin an den Parametern L_TYP, L_NO und L_OFFSET angezeigt, wo die Daten abgelegt werden. Zusätzlich werden für einen Zyklus die Parameter L_CF_BYT und L_CF_BIT und die Länge LEN des entsprechenden Auftrags angezeigt.

Anwendung der Koppelmerker

Vor dem Empfang von Daten wird der im RK 512-Telegrammkopf angegebene Koppelmerker abgeprüft. Die Daten werden nur übertragen, wenn der Koppelmerker den Wert "0" hat. Nach Abschluss des Übertragungsvorganges wird der Koppelmerker von der Anweisung auf den Wert "1" gesetzt und an der Anweisung für einen Zyklus (NDR) der Koppelmerker ausgegeben.

Im Anwenderprogramm kann durch Auswertung des Koppelmerkers erkannt werden, dass die übertragenen Daten bearbeitet werden können. Sobald die Daten bearbeitet wurden, muss der Anwender den Koppelmerker wieder auf "0" setzen. Ein SEND-Auftrag vom Kommunikationspartner ist wieder möglich.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten empfangen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt. Defaultwert: 0
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen
DB_NO	INPUT	INT	Datenbausteinnummer der Empfangsdaten (Ziel) Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt (nur bei Datenziel DX relevant)
DBB_NO	INPUT	INT	Datenbytenummer der Empfangsdaten (Ziel) 0 ≤ DBB_NO ≤ 8190 Empfangsdaten ab Datenbyte (nur bei Datenziel DX relevant)
L_TYP [1]	OUTPUT	CHAR	Typ des Bereiches auf lokaler CPU (Ziel) 'D': Datenbaustein
L_NO [1]	OUTPUT	INT	Datenbausteinnummer auf lokaler CPU (Ziel) 0 ≤ L_NO ≤ 255
L_OFFSET [1]	OUTPUT	INT	Datenbytenummer auf lokaler CPU (Ziel) 0 ≤ L_OFFSET ≤ 510
L_CF_BYT [1]	OUTPUT	INT	Koppelmerkerbyte auf lokaler CPU 0 ≤ L_CF_BYTE ≤ 255255 bedeutet: ohne Koppelmerker
L_CF_BIT [1]	OUTPUT	INT	Koppelmerkerbit auf lokaler CPU 0 ≤ L_CF_BIT ≤ 7
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00;

Parameter	Deklaration	Datentyp	Beschreibung
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 0 ≤ LEN ≤ 4096,Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung P_RCV_RK arbeitet mit einem Instanz-DB I_RCV_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0E, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter NDR, LEN und ERROR je nach Eingangsbeschaltung von EN_R und R.

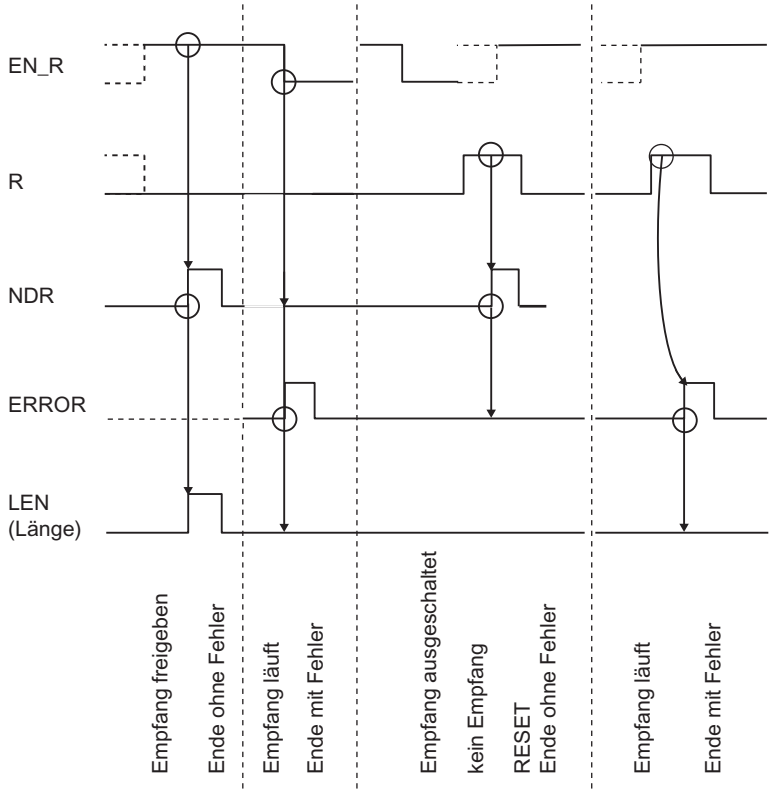

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Verhalten im Fehlerfall

ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. NDR und ERROR/STATUS werden auch bei RESET der Anweisung P_RCV_RK ausgegeben (Parameter LEN == 16#00). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Regel

Hinweis

Die Anweisung P_RCV_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

P_RCV_RK: Daten bereitstellen mit RK 512

Beschreibung

Der Aufruf der Anweisung P_RCV_RK ist notwendig, wenn der Kommunikationspartner einen Auftrag "Daten holen" (FETCH-Auftrag) durchführt.

Die Anweisung P_RCV_RK stellt Daten aus einem S7-Datenbereich für den CP 341 bereit. Die Anweisung P_RCV_RK wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten an den CP 341 bereitzustellen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R abgebrochen werden. Der abgebrochene Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Auftrag ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Übertragung über mehrere Aufrufe (Programmzyklen) laufen.

Der Typ des Quellbereiches (L_TYP), die Quell-Datenbausteinnummer (L_NO, nur relevant bei L_TYP = DB), der Offset im Quellbereich (L_OFFSET) sowie die Länge (LEN) der bereitzustellenden Daten wird aus dem ersten RK 512-Telegramm bestimmt. Die Anweisung wertet die Information aus diesem Telegramm aus und transferiert die geforderten Daten zum CP 341. Die Parameter DB_NO und DBB_NO haben bei der Anweisung P_RCV_RK keine Bedeutung.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung P_RCV_RK in den Grundzustand versetzt. Der Auftrag ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). Für einen Zyklus wird daraufhin an den Parametern L_TYP, L_NO und L_OFFSET angezeigt, von wo die Daten geholt wurden (mögliche Datentypen: Datenbausteine, Eingangsbytes, Ausgangsbytes, Zeiten und Zähler). Zusätzlich werden für einen Zyklus die

Parameter L_CF_BYT und L_CF_BIT und die Länge LEN des entsprechenden Auftrags angezeigt.

Hinweis

Wenn der Kommunikationspartner Zeiten bzw. Zähler vom CP 341 holt, ist die Länge auf maximal 32 Byte (16 Zeiten bzw. Zähler zu je 2 Byte) begrenzt.

Anwendung der Koppelmerker

Nach dem Empfang des Telegramms wird der im RK 512-Telegrammkopf angegebene Koppelmerker abgeprüft. Die Daten werden nur bereitgestellt, wenn der Koppelmerker den Wert "0" hat. Nach Abschluss des Übertragungsvorganges wird der Koppelmerker von der Anweisung auf den Wert "1" gesetzt und an der Anweisung für einen Zyklus (NDR) der Koppelmerker ausgegeben.

Im Anwenderprogramm kann durch Auswertung des Koppelmerkers erkannt werden, dass auf die bereitgestellten Daten wieder zugegriffen werden kann. Sobald die Daten bearbeitet wurden, muss der Anwender den Koppelmerker wieder auf "0" setzen. Ein FETCH-Auftrag vom Kommunikationspartner ist wieder möglich.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten bereitstellen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Bereitstellen gesperrt. Defaultwert: 0
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	irrelevant
DBB_NO	INPUT	INT	irrelevant
L_TYP [1]	OUTPUT	CHAR	Typ des Bereiches auf lokaler CPU (Quelle) 'D': Datenbaustein 'M': Merker 'E': Eingänge 'A': Ausgänge 'Z': Zähler 'T': Zeiten
L_NO [1]	OUTPUT	INT	Datenbausteinnummer auf lokaler CPU (Quelle) 0 ≤ L_NO ≤ 255 (nur relevant, wenn L_TYP = D)
L_OFFSET [1]	OUTPUT	INT	Datenbytenummer auf lokaler CPU (Quelle) 0 ≤ L_OFFSET ≤ 510 (abhängig vom Bereichstyp)
L_CF_BYT [1]	OUTPUT	INT	Koppelmerkerbyte auf lokaler CPU 0 ≤ CF_BYTE ≤ 255 255 bedeutet: ohne Koppelmerker

Parameter	Deklaration	Datentyp	Beschreibung
L_CF_BIT [1]	OUTPUT	INT	Koppelmerkerbit auf lokaler CPU 0 ≤ CF_BIT ≤ 7
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 0 ≤ LEN ≤ 4096, Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung P_RCV_RK arbeitet mit einem Instanz-DB I_RCV_RK zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0E, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter NDR, LEN und ERROR je nach Eingangsbeschaltung von EN_R und R.

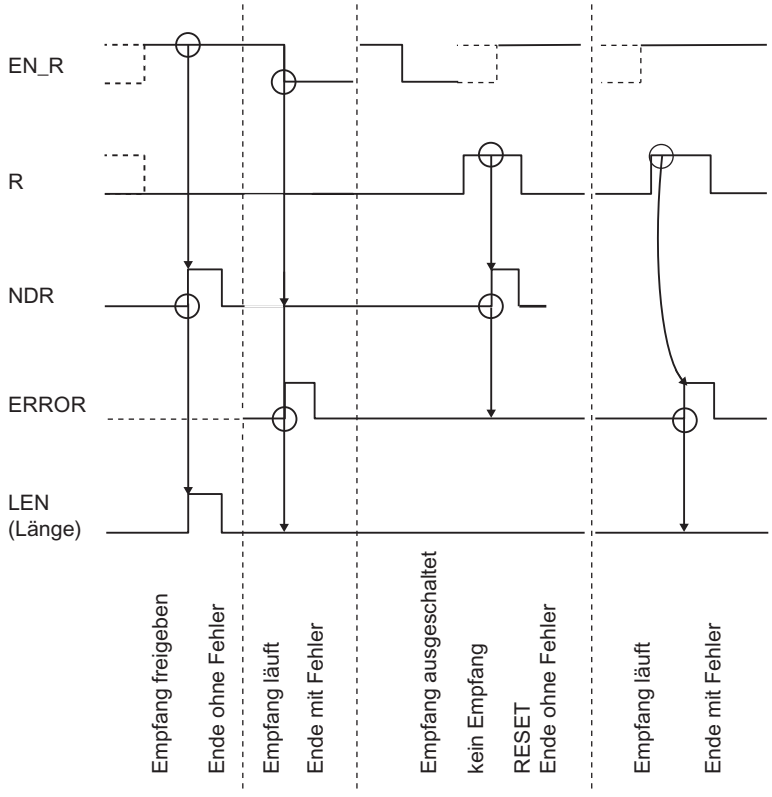

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Verhalten im Fehlerfall

ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. NDR und ERROR/STATUS werden auch bei RESET der Anweisung P_RCV_RK ausgegeben (Parameter LEN == 16#00). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Regel

Hinweis

Die Anweisung P_RCV_RK hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

P_PRT341: Meldetext mit bis zu 4 Variablen auf Drucker ausgeben

Beschreibung

Für die Druckerausgabe von Meldetexten steht Ihnen die Anweisung P_PRT341 zur Verfügung. Die Anweisung P_PRT341 überträgt z. B. eine Prozessmeldung zum CP 341. Der CP 341 protokolliert die Prozessmeldung auf dem angeschlossenen Drucker.

Arbeitsweise

Die Anweisung P_PRT341 überträgt einen Meldetext mit bis zu vier Variablen zum CP 341. Die Anweisung P_PRT341 wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Über die Parameter DB_NO und DBB_NO sind die Pointer (Zeiger auf Datenbausteine) für den Formatstring und die vier Variablen erreichbar. Die Pointer müssen lückenlos, in einer bestimmten Reihenfolge, im parametrierten Datenbaustein (Pointer-DB) abgelegt sein (siehe Bild "Pointer-DB").

Mit einer positiven Flanke am Eingang REQ wird die Übertragung des Meldetextes angestoßen. Zuerst wird der Formatstring des Meldetextes übertragen. Danach werden die Variablen 1 bis 4 übergeben.

Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung P_PRT341 kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 341 abgebrochen und die Anweisung P_PRT341 in den Grundzustand versetzt. Daten, die der CP 341 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden von Druckaufträgen ausgeschaltet.

An dem Parameter LADDR wird die Adresse des anzusprechenden CP 341 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung P_PRT341 ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Druckerausgabe gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 341 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Zeiger auf Pointer-DB: CPU-spezifisch, Null ist nicht erlaubt (Die Pointer auf Variablen und Formatstring sind im Pointer-DB in fester Reihenfolge hinterlegt.)
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8162 Pointer ab Datenbyte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6666) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6666) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich, Instanz-DB

Die Anweisung P_PRT341 arbeitet mit einem Instanz-DB I_PRINT zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR bzw. SFCSTATUS entnehmen.

Belegung im Datenbereich, Pointer-DB

Die Anweisung P_PRT341 greift über die Parameter DB_NO und DBB_NO auf einen Pointer-DB zu, in dem die Pointer (Zeiger) auf die Datenbausteine mit den Meldetexten und Variablen in fester Reihenfolge hinterlegt sind. Der Pointer-DB muss von Ihnen erzeugt werden.

Das Bild zeigt den Aufbau des Pointer-DBs, der durch die Parameter DB_NO und DBB_NO der Anweisung P_PRT341 angesprochen wird.

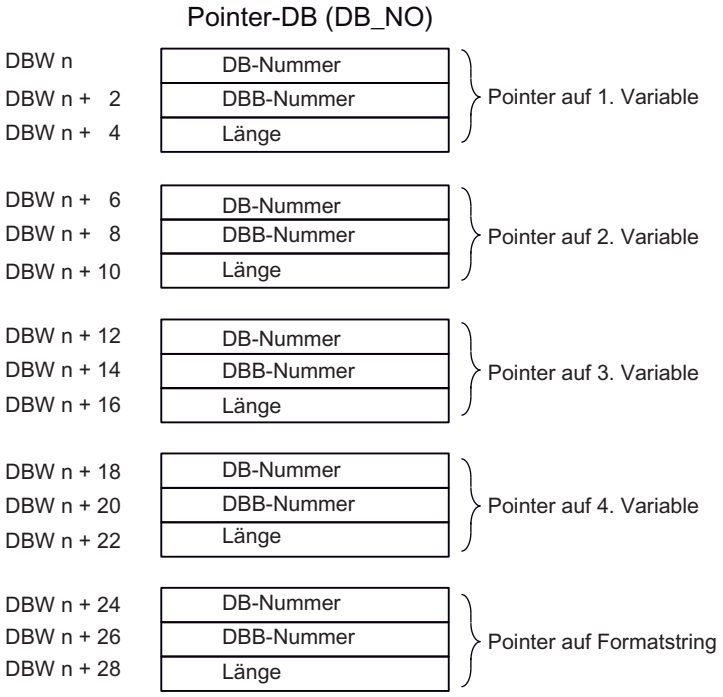

Zulässige DB-Nummer

Die zulässigen DB-Nummern sind CPU-spezifisch. Wird bei "Pointer auf Variable" als DB-Nummer der Wert 16#00 angegeben, dann wird diese Variable als nicht vorhanden interpretiert und der Pointer auf die nächste Variable oder auf den Formatstring gesetzt.

Ist bei "Pointer auf Formatstring" die DB-Nummer gleich dem Wert 16#00, dann wird der Druckauftrag abgebrochen und die Ereignisnummer 16#1E43 am Parameterausgang STATUS der Anweisung P_PRT341 angezeigt.

Zulässige DBB-Nummer

Ab der parametrierten DBB-Nummer steht die Variable oder der Formatstring. Die Variablen dürfen eine maximale Länge von 32 Byte haben, der Formatstring eine maximale Länge von 150 Byte.

Wird die maximale Länge überschritten, dann wird der Druckauftrag abgebrochen und die Ereignisnummer 16#1E41 am Parameterausgang STATUS der Anweisung P_PRT341 angezeigt.

Zulässige Länge

Die Längenangabe im Pointer-DB ist für die jeweilige Darstellungsart (Datentypen) sowie abhängig von der verwendeten Präzision anzupassen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

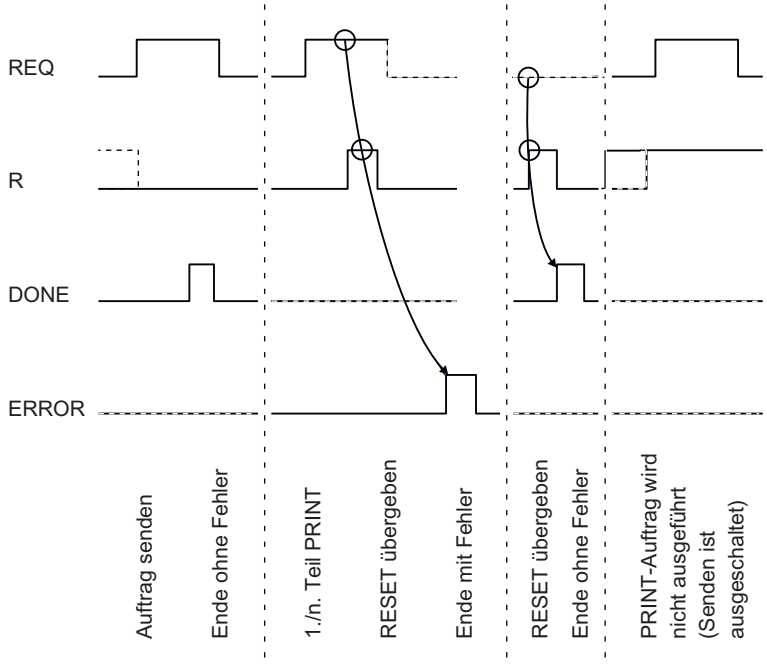

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Er muss nicht während der gesamten Übertragung Signalzustand "1" haben.

Regeln

Hinweis

Die Anweisung P_PRT341 hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN vom CP 341 bearbeitet werden kann, muss der CP-CPU-Anlaufmechanismus der Anweisung P_PRT341 abgeschlossen sein. Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zum CP 341 übertragen.

V24_STAT_340 / V24_STAT: Begleitsignale an der RS232C-Schnittstelle lesen

Beschreibung

Die Anweisung V24_STAT_340 (CP 340) / V24_STAT (CP 341) liest vom CP 34x die RS 232C-Begleitsignale und stellt sie dem Anwender an den Bausteinparametern zur Verfügung. Die Anweisung V24_STAT_340 / V24_STAT wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Die RS 232C-Begleitsignale werden mit jedem Aufruf der Anweisung aktualisiert (zyklisches Pollen). Der CP 34x aktualisiert den Status der Ein-/Ausgänge in einem Zeitraster von 20 ms. Die Ein-/Ausgänge werden unabhängig davon laufend aktualisiert.

Das Binärergebnis BIE wird nicht beeinflusst. Die Anweisung gibt keine Fehlermeldung aus.

Mit dem Parameter LADDR wird der anzusprechende CP 34x ausgewählt.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 34x Die Basisadresse wird aus STEP 7 entnommen.
DTR_OUT	OUTPUT	BOOL	**D**ata **t**erminal **r**eady, CP 34x betriebsbereit (CP 34x-Ausgang)
DSR_IN	OUTPUT	BOOL	**D**ata **s**et **r**eady, Kommunkationspartner betriebsbereit (CP 34x-Eingang)
RTS_OUT	OUTPUT	BOOL	**R**equest **t**o **s**end, CP 34x sendebereit (CP 34x-Ausgang)
CTS_IN	OUTPUT	BOOL	**C**lear **t**o **s**end, Kommunikationspartner kann Daten vom CP 34x empfangen (Anwort auf RTS = ON des CP 34x) (CP 34x-Eingang)
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, Empfangssignalpegel (CP 34x-Eingang)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, Rufzeichen (CP 34x-Eingang)

Belegung im Datenbereich

Die Anweisung V24_STAT_340 / V24_STAT belegt keine Datenbereiche.

Regel

Hinweis

Zum Erkennen eines Signalwechsels ist eine Mindestimpulsdauer notwendig. Ausschlaggebende Größen sind die CPU-Zykluszeit, die Aktualisierungszeit auf dem CP 34x und die Reaktionszeit des Kommunikationspartners.

V24_SET_340 / V24_SET: Begleitsignale an der RS232C-Schnittstelle schreiben

Beschreibung

Der Anwender kann über die Parametereingänge der Anweisung V24_SET_340 (CP 340) / V24_SET (CP 341) die entsprechenden Schnittstellenausgänge setzen oder rücksetzen. Die Anweisung V24_SET_340 / V24_SET wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Das Binärergebnis BIE wird nicht beeinflusst. Die Anweisung gibt keine Fehlermeldung aus.

Mit dem Parameter LADDR wird der anzusprechende CP 34x ausgewählt.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 34x Die Basisadresse wird aus STEP 7 entnommen.
RTS	INPUT	BOOL	**R**equest **t**o **s**end, CP 34x sendebereit (CP 34x-Ausgang steuern)
DTR	INPUT	BOOL	**D**ata **t**erminal **r**eady, CP 34x betriebsbereit (CP 34x-Ausgang steuern)

Belegung im Datenbereich

Die Anweisung V24_SET_340 / V24_SET belegt keine Datenbereiche.

Parameter STATUS

Ereignisklassen

Ereignisklasse	Bedeutung
0	Anlauf des CP
1	Hardwarefehler auf dem CP
2	Fehler bei der Initialisierung
3	Fehler bei der Parametrierung der Anweisungen
4	Vom CP erkannte Fehler im Datenverkehr CP - CPU
5	Fehler bei Bearbeitung eines CPU-Auftrags
6	Fehler bei Bearbeitung eines Partnerauftrags (nur bei RK 512)
7	Sendefehler
8	Empfangsfehler
9	Reaktionstelegramm mit Fehler oder Fehlertelegramm vom Koppelpartner empfangen
10 ($0A_H$)	Fehler beim Reaktionstelegramm des Partners, die vom CP erkannt wurden
30 ($1E_H$)	Fehler bei Kommunikation zwischen CP und CPU

Parameter STATUS

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0003	PtP–Parameter übernommen	-
0004	Parameter schon auf CP (Zeitstände gleich)	-
0007	Zustandsübergang CPU in STOP	-
0008	Zustandsübergang CPU in RUN/ANLAUF	-
0101	Fehler beim Test des Betriebssystem–EP-ROM des CP	CP defekt, CP muss getauscht werden.
0102	RAM–Test des CP fehlerhaft	
0103	Auftragsschnittstelle des CP defekt	
0110	Fehler in der CP–Firmware	Schalten Sie die Baugruppe aus und wieder ein. Tauschen Sie notfalls die Baugruppe.
020F	Beim Start der parametrierten Kommunikation wird eine unzulässige Parametrierung festgestellt. Die Schnittstelle konnte nicht parametriert werden.	Korrigieren Sie die unzulässige Parametrierung, und führen Sie einen Neustart durch.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0301	Quell–/Zieldatentyp nicht zulässig bzw. nicht vorhanden Bereich (Anfangsadresse, Länge) nicht zulässig DB nicht vorhanden bzw. nicht zulässig (z. B. DB 0) oder anderer Datentyp nicht vorhanden bzw. nicht zulässig Koppelmerkerbyte–Nummer ungültig oder Koppelmerkerbit–Nummer ungültig oder weder 'S' noch 'F' angewählt (bei der Anweisung P_SND_RK)	Parametrierung auf CPU und CP prüfen und evtl. korrigieren. **Nur bei RK 512:** Partner liefert unzulässige Parameter im Telegrammkopf. Parametrierung auf CPU und CP prüfen, evtl. Baustein einrichten. Entnehmen Sie den Auftragstabellen die zulässigen Datentypen. **Nur bei RK 512:** Partner liefert falsche Parameter im Telegrammkopf.
0403	Fehlerhafter, unbekannter oder nicht erlaubter Datentyp	Programm überprüfen, z. B. falsche Parametrierung der Anweisung.
0407	Fehler bei der Datenübertragung zwischen CPU und CP	Wird der Fehler ständig gemeldet, müssen Sie überprüfen, ob die von Ihnen im Anwenderprogramm aufgerufenen Anweisungen richtig parametriert sind. Wird der Fehler unmittelbar nach NETZ–EIN gemeldet, ist zu diesem Zeitpunkt noch keine Verbindung zur CPU aufgebaut. Bei der Prozedur 3964(R) und beim ASCII–Treiber wird die Datenübertragung vom empfangenden CP 341 wiederholt, bis die Daten zur CPU hin übertragen sind. Bei RK 512 wird der Auftrag negativ quittiert und muss im Anwenderprogramm wiederholt werden. Erfolgt die Meldung sporadisch während der laufenden Datenübertragung, kann die CPU zeitweise die Daten nicht übernehmen. Bei der Prozedur 3964(R) und beim ASCII–Treiber wird die Datenübertragung dann vom empfangenden CP 341 wiederholt, bis die Daten zur CPU hin übertragen sind. Bei RK 512 wird der Auftrag negativ quittiert und muss im Anwenderprogramm wiederholt werden. Zur Abhilfe sollten Sie in Ihrem Anwenderprogramm dann die Anweisung P_RCV_RK häufiger aufrufen.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0408	Fehler bei der Datenübertragung zwischen CPU und dem CP (Empfang)	
	• CPU ist temporär überlastet, Auftrag wird wiederholt	• Anzahl der Kommunikationsaufrufe reduzieren
	• Auf Datenbereich der CPU kann temporär nicht zugegriffen werden, z.B. weil Empfangsbaustein zu selten aufgerufen wird.	• Empfangsbaustein häufiger aufrufen
	• Auf Datenbereich der CPU kann temporär nicht zugegriffen werden, z.B. weil Empfangsbaustein zwischendurch gesperrt wird (EN=false).	• Überprüfen Sie, ob der Empfangsbaustein zu lange gesperrt wird
0409	Empfang von Daten nicht möglich. Fehler bei der Datenübertragung zwischen CPU und CP (Empfang). Empfang von Daten nicht möglich. Nach mehrmaligen Versuchen wurde Auftrag nach 10 s abgebrochen, weil	
	• Empfangsbaustein nicht aufgerufen wird.	• Überprüfen Sie Ihr Anwenderprogramm, ob der Empfangsbaustein durchlaufen wird.
	• Empfangsbaustein gesperrt ist.	• Überprüfen Sie, ob der Empfangsbaustein gesperrt ist.
	• auf Datenbereich der CPU nicht zugegriffen werden kann.	• Überprüfen Sie, ob der Datenbereich, in den die Daten übertragen werden sollen, vorhanden ist.
	• Datenbereich der CPU zu kurz ist.	• Überprüfen Sie die Länge des Datenbereichs
040A	Fehler bei der Datenübertragung zwischen CPU und CP. Die Datenübertragung wurde durch RESET abgebrochen, weil: • Ziel–DB nicht vorhanden ist • Ziel–DB zu kurz ist • RESET–Bit an der Anweisung gesetzt ist	Ziel–DB im Anwenderprogramm erzeugen bzw. vorhandenen Ziel–DB verlängern.
0501	Laufender Auftrag wurde durch Neuanlauf des CP abgebrochen.	Bei NETZ–EIN ist keine Hilfe möglich. Beim Umparametrieren des CP vom PG aus sollten Sie vor dem Beschreiben einer Schnittstelle darauf achten, dass von der CPU aus keine Aufträge mehr laufen.
0502	Auftrag ist in diesem Betriebszustand des CP nicht erlaubt (z. B. Geräteschnittstelle nicht parametriert).	Parametrieren Sie die Geräteschnittstelle.
0505	**Nur bei Drucker-Treiber:** Systemdatenbaustein mit Meldetexten auf CP nicht vorhanden	Projektieren Sie die Meldetexte mit der Parametriersoftware, und führen Sie einen Neustart durch.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0506	**Nur bei Drucker-Treiber:** Meldetext nicht vorhanden	Projektieren Sie die Meldetexte mit der Parametriersoftware, und führen Sie einen Neustart durch.
0507	**Nur bei Drucker-Treiber:** Meldetext zu lang	Ändern Sie den Meldetext auf maximal 150 Zeichen (auf maximal 250 Zeichen bei der Verwendung von Variablen)
0508	**Nur bei Drucker-Treiber:** Zu viele Konvertierungsanweisungen	Sie haben mehr Konvertierungsanweisungen als Variablen projektiert. Die Konvertierungsanweisungen ohne dazugehörige Variable werden ignoriert.
0509	**Nur bei Drucker-Treiber:** Zu viele Variablen	Sie haben mehr Variablen als Konvertierungsanweisungen projektiert. Die Variablen ohne Konvertierungsanweisung werden nicht ausgegeben.
050A	**Nur bei Drucker-Treiber:** Unbekannte Konvertierungsanweisung	Überprüfen Sie die Konvertierungsanweisung. Nicht definierte oder nicht unterstützte Konvertierungsanweisungen werden im Ausdruck durch ****** ersetzt.
050B	**Nur bei Drucker-Treiber:** Unbekannte Steueranweisung	Überprüfen Sie die Steueranweisung. Nicht definierte oder nicht unterstützte Steueranweisungen werden ignoriert. Die Steueranweisung wird auch nicht als Text ausgegeben.
050C	**Nur bei Drucker-Treiber:** Konvertierungsanweisung nicht ausführbar	Überprüfen Sie die Konvertierungsanweisung. Nicht ausführbare Konvertierungsanweisungen werden im Ausdruck entsprechend der definierten Breite und dem gültigen Rest der Konvertierungsanweisung bzw. der Standarddarstellung mit *–Zeichen ausgegeben.
050D	**Nur bei Drucker-Treiber:** Breite in Konvertierungsanweisung zu gering oder zu groß	Korrigieren Sie die angegebene Breite der Variable in der Konvertierungsanweisung anhand der maximal möglichen Zeichenanzahl der Variable bei textorientierten Darstellungsarten (A, C, D, S, T, Y, Z). Im Ausdruck wird nur die durch die Breite angegebene Anzahl Zeichen ausgegeben, der Rest wird abgeschnitten. Ansonsten werden entsprechend der Breite *–Zeichen ausgegeben.
050E	**Nur bei ASCII-Treiber:** Fehler beim Senden. Die parametrierten Endekennzeichen sind innerhalb der maximal zulässigen Länge nicht aufgetreten bzw. beim automatischen Anhängen wurde die maximal zulässige Sendelänge überschritten.	Ergänzen Sie die Endekennzeichen im Sendepuffer an der gewünschten Stelle oder wählen Sie beim automatischen Anhängen eine kleinere Telegrammlänge.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0514	Anfangsadressen zu hoch angegeben für gewünschten Datentyp oder Anfangsadresse bzw. DB/DX–Nr. zu niedrig.	Entnehmen Sie den Auftragstabellen die zulässigen Anfangsadressen und DB/DX–Nummern, die im Programm angegeben werden dürfen.
0515	**Nur bei RK 512:** Falsche Bitnummer beim Koppelmerker angegeben.	Erlaubte Bit–Nr.: 0 bis 7
0516	**Nur bei RK 512:** CPU–Nummer zu groß angegeben.	Erlaubte CPU–Nr.: keine, 0, 1, 2, 3 oder 4
0517	Übertragungslänge > 1 kByte ist für CP zu groß oder Länge für Schnittstellenparameter zu klein.	Spalten Sie den Auftrag auf mehrere Aufträge mit kleinerer Länge.
051A	**Nur bei RK 512:** Fehler beim Senden eines Befehlstelegramms Eine zugehörige Prozedurfehlernummer wurde unmittelbar zuvor im STATUS eingetragen.	Siehe Abhilfemaßnahmen bei vorangegangener Ereignisnummer.
051B	**Nur bei Drucker-Treiber:** Präzision ungültig	Korrigieren Sie die angegebene Präzision in der Konvertierungsanweisung. Die Präzision wird immer durch einen Punkt eingeleitet zur Kennzeichnung und Abgrenzung zur Breite (z. B.: ".2" zur Ausgabe von Dezimalpunkt und 2 Nachkommastellen). Die Präzision ist nur bei den Darstellungsarten F, R, A und D relevant. Ansonsten wird sie ignoriert.
051C	**Nur bei Drucker-Treiber:** Variable ungültig (Variablenlänge falsch/falscher Typ)	Korrigieren Sie die angegebene Variable. Die bei der jeweiligen Darstellungsart möglichen Datentypen können Sie der entsprechenden Tabelle entnehmen.
051E	**Nur bei Drucker-Treiber:** Die mit diesem Auftrag gesendeten "Zeilenendesequenzen" (i.e.: $R / $L / $N) passen nicht (mehr) auf die (angefangene) Seite	Vergrößern Sie Ihre Seitenlänge, reduzieren Sie die Anzahl der Zeilen (bzw. der Zeilenvorschübe) oder verteilen Sie Ihren Ausdruck auf mehrere Seiten
0601	Fehler im 1. Befehlsbyte (nicht 00 oder FFH)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0602	Fehler im 3. Befehlsbyte (nicht A, 0 oder E)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0603	Fehler im 3. Befehlsbyte bei Folgetelegrammen (Befehl nicht wie beim 1. Telegramm)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0604	Fehler im 4. Befehlsbyte (Befehlsbuchstabe falsch)	Prinzipieller Kopfaufbaufehler beim Partner bzw. eine Befehlskombination wurde verlangt, die beim CP nicht erlaubt ist. Kontrollieren Sie die zulässigen Befehle. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0605	Fehler im 4. Befehlsbyte beim Folgetelegramm (Befehl nicht wie beim 1. Telegramm)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0606	Fehler im 5. Befehlsbyte (DB–Nummer nicht zulässig)	Entnehmen Sie den Auftragstabellen die zulässigen DB–Nummern, Anfangsadressen oder Längen.
0607	Fehler im 5. bzw. 6. Befehlsbyte (Anfangsadresse zu hoch)	Entnehmen Sie den Auftragstabellen die zulässigen DB–Nummern, Anfangsadressen oder Längen.
0608	Fehler im 7. bzw. 8. Befehlsbyte (Länge unzulässig)	Entnehmen Sie den Auftragstabellen die zulässigen DB/DX–Nummern, Anfangsadressen oder Längen.
0609	Fehler im 9. und 10. Befehlsbyte (Koordinierungsmerker bei diesem Datentyp nicht zulässig oder Bitnummer zu hoch)	Prinzipieller Kopfaufbaufehler beim Partner. Entnehmen Sie den Auftragstabellen, wann ein Koordinierungsmerker erlaubt ist.
060A	Fehler im 10. Befehlsbyte (CPU–Nummer nicht erlaubt)	Prinzipieller Kopfaufbaufehler beim Partner
060B	SEND–Telegramm war länger/ kürzer als erwartet (es wurden mehr/weniger Daten empfangen, als im Telegrammkopf angekündigt).	Korrektur beim Partner erforderlich
060C	FETCH–Befehlstelegramm mit Nutzdaten empfangen.	Korrektur beim Partner erforderlich
060D	CP hat Telegramm empfangen bei unzulässiger Betriebsart:	
	• Die Empfangsverbindung zwischen der CPU und dem CP ist nicht oder noch nicht korrekt aufgebaut	• Überprüfen Sie, ob die angesprochene Verbindung (korrekt) parametriert ist
	• CP ist noch nicht vollständig angelaufen	• Diese Fehlermeldung kann nur im Anlauf des CP auftreten. Wiederholen Sie den Auftrag.
	• Empfangende CPU befindet sich im Betriebszustand STOP	• Versetzen Sie die CPU wieder in den Betriebszustand RUN und wiederholen Sie den Auftrag.
	• Angesprochene Schnittstelle wird gerade umparametriert	• Hierbei handelt es sich um einen temporären Fehler. Wiederholen Sie den Auftrag.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
060E	Synchronfehler des Partners • Neues (Folge–)Befehlstelegramm empfangen, bevor Reaktionstelegramm gesendet wurde. • 1. Befehlstelegramm erwartet und Folgetelegramm kam. • Folgebefehlstelegramm erwartet und 1. Telegramm kam.	Dieser Fehler kann nach Neustart des eigenen Automatisierungssystems bei langen Telegrammen oder bei Neustart des Partners gemeldet werden. In diesen Fällen handelt es sich um normales Anlaufverhalten der Anlage. Im laufenden Betrieb kann der Fehler auch infolge von vorausgegangenen Fehlerzuständen auftreten, die nur vom Partner erkannt wurden. Andernfalls müssen Sie von einem Fehlverhalten des Partnergerätes ausgehen. Evtl. tritt der Fehler nicht auf bei Aufträgen < 128 Byte.
060F	DB gesperrt durch Koordinierungsfunktion	Im eigenen Programm: Nach Bearbeitung der letzten Übertragungsdaten Koppelmerker rücksetzen. Im Partnerprogramm: Wiederholen des Auftrags
0610	Zu kurzes Telegramm empfangen (Länge < 4 Byte bei Folge– oder Reaktionstelegrammen oder Länge < 10 Byte bei Befehlstelegrammen)	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0611	Telegrammlänge und Längenangabe im Telegrammkopf stimmen nicht überein.	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0612	Fehler beim Senden des (Folge–) Reaktionstelegramms, eine zugehörige Prozedurfehlernummer wurde unmittelbar zuvor im STATUS eingetragen.	Siehe Abhilfemaßnahmen zu der unmittelbar zuvor im STATUS eingetragenen Fehlernummer.
0701	Senden der ersten Wiederholung: • Beim Senden des Telegramms wurde ein Fehler erkannt oder • der Partner forderte durch ein negatives Quittungszeichen (NAK) eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der max. Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehler–Nr. gemeldet, die den Fehler beschreibt, der zuerst auftrat.
0702	**Nur bei 3964(R):** Fehler beim Verbindungsaufbau: Nachdem STX gesendet wurde, wurde NAK oder ein beliebiges Zeichen (außer DLE oder STX) empfangen.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0703	**Nur bei 3964(R):** Quittungsverzugszeit (QVZ) überschritten: Nach Senden von STX kam keine Antwort vom Partner innerhalb der Quittungsverzugszeit.	Partnergerät ist zu langsam oder nicht empfangsbereit, oder es liegt z. B ein Bruch der Sendeleitung vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0704	**Nur bei 3964(R):** Abbruch durch Partner: Während des laufenden Sendebetriebes wurden vom Partner ein oder mehrere Zeichen empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0706	**Nur bei 3964(R):** Fehler bei Verbindungsende: • Das Telegramm wurde vom Partner am Ende mit NAK oder einem beliebigen Zeichen (außer DLE) abgelehnt oder • das Quittungszeichen (DLE) wurde zu früh empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0707	**Nur bei 3964(R):** Quittungsverzugszeit am Verbindungsende/Antwortüberwachungszeit nach Sendetelegramm überschritten: Nach Verbindungsabbau mit DLE ETX kam innerhalb der QVZ keine Antwort vom Partner.	Partnergerät ist zu langsam oder gestört. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0708	**Nur bei ASCII–Treiber und Drucker-Treiber:** Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
0709	Kein Verbindungsaufbau möglich, die Anzahl der erlaubten Aufbauversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion zwischen CPU und CP richtig parametriert ist.
070A	Die Daten konnten nicht übertragen werden, die erlaubte Anzahl der Übertragungsversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter.
0801	Erwarten der ersten Wiederholung: Beim Empfangen eines Telegrammes wurde ein Fehler erkannt und der CP forderte durch eine negative Quittierung (NAK) beim Partner eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der max. Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehler–Nr. gemeldet, die den Fehler beschreibt, der zuerst auftrat.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0802	**Nur bei 3964(R):** Fehler beim Verbindungsaufbau: • In Ruhestellung wurden ein oder mehrere beliebige Zeichen (außer NAK oder STX) empfangen oder • nach einem empfangenen STX wurden vom Partner weitere Zeichen gesendet, ohne die Antwort DLE abzuwarten. Nach Netz–EIN des Partners: • während der Partner eingeschaltet wird, empfängt der CP ein undefiniertes Zeichen.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0805	**Nur bei 3964(R):** Logischer Fehler während des Empfangs: Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Fehlverhalten des Partnergeräts mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0806	Zeichenverzugszeit (ZVZ) überschritten: • Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der ZVZ empfangen oder Nur bei 3964(R): • 1. Zeichen nach Senden von DLE beim Verbindungsaufbau wurde nicht innerhalb der ZVZ empfangen.	Partnergerät ist zu langsam oder gestört. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0808	**Nur bei 3964(R):** Fehler beim Blockprüfzeichen BCC: Der intern gebildete Wert des BCC stimmt nicht mit dem vom Partner am Verbindungsende empfangenen BCC überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
080A	Ein freier Empfangspuffer ist nicht vorhanden: Beim Empfang stand kein leerer Empfangspuffer zur Verfügung.	Die Anweisung P_RCV_RK muss häufiger aufgerufen werden.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
080C	Übertragungsfehler: • Ein Übertragungsfehler (Paritätsfehler, Stopbitfehler, Überlauffehler) wurde erkannt. **Nur bei 3964(R):** • Wird in Ruhestellung ein gestörtes Zeichen empfangen, wird der Fehler sofort gemeldet, damit Störeinflüsse auf der Übertragungsleitung frühzeitig erkannt werden können. **Nur bei RK 512 und 3964(R):** • Falls dies während des Sende– oder Empfangsbetriebes auftritt, werden Wiederholungen gestartet.	Störungen auf der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind.
080D	BREAK: Empfangsleitung zum Partner ist unterbrochen.	Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
0815	Einstellung der Übertragungsversuche des CP und des Kommunikationspartners stimmen nicht überein.	Parametrieren Sie beim Partner die gleiche Anzahl der Übertragungsversuche wie am CP. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0816	Die Länge eines empfangenen Telegramms war länger als die maximal vereinbarte Länge.	Korrektur beim Partner erforderlich.
0818	**Nur bei ASCII–Treiber:** DSR = OFF bzw. CTS = OFF	Vor oder während eines Sendevorgangs sind die Signale DSR bzw. CTS vom Partner auf "OFF" geschaltet worden. Überprüfen Sie die Steuerung der RS 232C–Begleitsignale beim Partner.
0902	**Nur bei RK 512:** Speicherzugriffsfehler beim Partner (Speicher nicht vorhanden) **Bei SIMATIC S5 als Partner:** • Falscher Bereich beim Anzeigenwort oder • Datenbereich nicht vorhanden (außer DB/DX) oder • Datenbereich zu kurz (außer DB/DX)	Kontrollieren Sie, ob der gewünschte Datenbereich beim Partner vorhanden und ausreichend groß ist bzw. prüfen Sie die Parameter des aufgerufenen Systemfunktionsbausteins. Kontrollieren Sie die angegebene Länge am Systemfunktionsbaustein.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
0903	**Nur bei RK 512:** DB/DX Zugriffsfehler beim Partner (DB/DX nicht vorhanden oder zu kurz) **Bei SIMATIC S5 als Partner:** • DB/DX nicht vorhanden oder • DB/DX zu kurz oder • DB/DX–Nr. unzulässig Beim FETCH–Auftrag zulässiger Quellbereich überschritten	Kontrollieren Sie, ob der gewünschte Datenbereich beim Partner vorhanden und ausreichend groß ist bzw. prüfen Sie die Parameter des aufgerufenen Systemfunktionsbausteins. Kontrollieren Sie die angegebene Länge am Systemfunktionsbaustein.
0904	**Nur bei RK 512:** Partner meldet "Auftragsart nicht erlaubt".	Fehlerhaftes Partnerverhalten, da vom CP niemals ein Systembefehl ausgegeben wird.
0905	**Nur bei RK 512:** Fehler beim Partner bzw. bei SIMATIC S5 als Partner: • Quell–/Zieltyp nicht zulässig oder • Speicherfehler im Partner–AG oder • Fehler bei Verständigung CP/CPU beim Partner oder • Partner–AG ist im STOP–Zustand	Prüfen, ob der Partner den gewünschten Datentyp übertragen kann. Aufbau der Hardware beim Partner prüfen. Schalter des Partner–AG in RUN–Stellung bringen.
0908	**Nur bei RK 512:** Partner erkennt Synchronfehler: Telegrammreihenfolge ist gestört.	Dieser Fehler tritt auf bei Neustart des eigenen AG oder des Partners. Es handelt sich dabei um ein normales Anlaufverhalten der Anlage. Sie brauchen nichts beheben. Bei laufendem Betrieb ist der Fehler auch in Folge von vorausgegangenen Fehlern denkbar. Andernfalls können Sie von einem Fehlverhalten des Partners ausgehen.
0909	**Nur bei RK 512:** DB/DX beim Partner gesperrt durch Koordinierungsmerker	Im Partnerprogramm: Nach Bearbeitung der letzten Übertragungsdaten den Koordinierungsmerker wieder rücksetzen! Im Programm: Auftrag wiederholen!
090A	**Nur bei RK 512:** Fehler im Telegrammkopf, die vom Partner erkannt werden: 3. Befehlsbyte im Kopf falsch	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
090B	**Nur bei RK 512:** Fehler im Telegrammkopf: 1. oder 4. Befehlsbyte im Kopf falsch	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
090C	**Nur bei RK 512:** Partner erkennt falsche Telegrammlänge (Gesamtlänge).	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
090D	**Nur bei RK 512:** Bisher erfolgte noch kein Neustart beim Partner.	Beim Partner–AG Neustart durchführen bzw. Betriebsartenwahlschalter am CP in Stellung RUN bringen.
090E	**Nur bei RK 512:** Unbekannte Fehlernummer im Reaktionstelegramm empfangen.	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A01	**Nur bei RK 512:** Synchronfehler des Partners, da • Reaktionstelegramm ohne Auftrag • Reaktionstelegramm empfangen, bevor Folgetelegramm gesendet wurde • nach Senden eines ersten Telegramms Folge–Reaktionstelegramm empfangen wurde • nach Senden eines Folgetelegramms erstes Reaktionstelegramm empfangen wurde	Dieser Fehler wird nach Neustart des eigenen AS bei langen Telegrammen oder bei Neustart des Partners gemeldet. Es handelt sich dabei um ein normales Anlaufverhalten der Anlage, Sie müssen nichts beheben. Im laufenden Betrieb kann der Fehler auch auftreten infolge von vorausgegangenen Fehlerzuständen, die nur vom Partner erkannt wurden. Andernfalls können Sie von einem Fehlverhalten des Partnergerätes ausgehen. Evtl. tritt der Fehler nicht auf bei Aufträgen < 128 Byte.
0A02	Nur bei RK 512: Fehler im Aufbau des empfangenen Reaktionstelegramms (1. Byte nicht 00 oder FF)	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A03	Nur bei RK 512: Empfangenes Reaktionstelegramm hat zu viele oder zu wenig Daten.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A04	Nur bei RK 512: Reaktionstelegramm auf SEND–Auftrag kam mit Daten.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A05	Nur bei RK 512: Innerhalb der Überwachungszeit kam kein Reaktionstelegramm vom Partner.	Ist der Partner ein sehr langsames Gerät? Oft wird dieser Fehler auch infolge eines vorangegangenen Fehlers angezeigt. Beispielsweise können Prozedurempfangsfehler (Ereignisklasse 8) angezeigt werden, nachdem ein FETCH–Telegramm gesendet wurde. Grund: Das Reaktionstelegramm konnte wegen Störungen nicht empfangen werden, die Überwachungszeit verstreicht. Evtl. tritt dieser Fehler auch auf, wenn beim Partner ein Neustart durchgeführt wurde, bevor er das zuletzt erhaltene FETCH–Telegramm beantworten konnte.
1E0D	Auftragsabbruch wegen Neustart, Wiederanlauf oder Reset	
1E0E	Statischer Fehler bei Aufruf des SFC RD_REC. Der Returnwert RET_VAL des SFCs wird Ihnen in der Variablen SFCERR im Instanz–DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz–DB.

Fehler-code (W#16#...)	Beschreibung	Abhilfe
1E0F	Statischer Fehler bei Aufruf des SFC WR_REC. Der Returnwert RET_VAL des SFCs wird Ihnen in der Variablen SFCERR im Instanz–DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz–DB.
1E41	Anzahl der am Parameter LEN der Anweisungen angegebenen Bytes unzulässig	Halten Sie den Wertebereich von 1 bis 4096 Byte ein.
1E42	Anweisung P_PRINT_RK: Anzahl der im Pointer–DB unter Länge angegebenen Bytes für Variable oder Formatstring ist unzulässig.	Halten Sie die zulässigen Längen ein: 32 Byte für Variablen, 150 Byte für Formatstring
1E43	Anweisung P_PRINT_RK: Kein Pointer für den Formatstring vorhanden.	Geben Sie im Pointer–DB die Datenbaustein–Nr. und die Datenwort–Nr. für den Formatstring an.

Hinweis

Eine Fehlermeldung wird nur dann ausgegeben, wenn gleichzeitig das Bit ERROR (Auftrag abgebrochen mit Fehler) gesetzt ist. In jedem anderen Fall ist das STATUS–Wort Null.

Variable SFCERR bzw. SFCSTATUS aufrufen

Nähere Informationen zum aufgetretenen Fehler 14 (1E0EH) und 15 (1E0FH) der Ereignisklasse 30 erhalten Sie über die Variable SFCERR bzw. SFCSTATUS.

Die Variable SFCERR bzw. SFCSTATUS können Sie nur über einen symbolischen Zugriff auf den Instanz–DB der entsprechenden Anweisung laden.

Fehlermeldungen, die in der Variablen SFCERR eingetragen werden finden Sie bei den Systemfunktionen SFC 58 unter "WR_REC: Datensatz in Peripherie schreiben" und SFC 59 unter "RD_REC: Datensatz von Peripherie lesen".

Fehlermeldungen, die in der Variablen SFCSTATUS eingetragen werden finden Sie bei den Systemfunktionen SFB 52 unter "RDREC: Datensatz lesen" und SFB 53 unter "WRREC: Datensatz schreiben".

Siehe auch

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

RDREC: Datensatz lesen (Seite 5456)

WRREC: Datensatz schreiben (Seite 5458)

PtP-Kopplung CP 440

SEND_440: Daten senden

Beschreibung

Die Anweisung SEND_440 überträgt einen Datenblock von einem Datenbaustein, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN, zum CP 440. Die Anweisung SEND_440 wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung SEND_440 kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zum CP 440 abgebrochen und die Anweisung SEND_440 in den Grundzustand versetzt. Daten, die der CP 440 bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 440 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung SEND_440 ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 440 Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer 0 ≤ DBB_NO ≤ 8190 Sendedaten ab Datenbyte; Offset ist CPU-spezifisch

Parameter	Deklaration	Datentyp	Beschreibung
LEN	INPUT	INT	Datenlänge 1 ≤ LEN ≤ 400, Angabe in Anzahl Byte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6686) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6686) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung SEND_440 arbeitet mit einem Instanz-DB I_SEND_440 zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall (STATUS == W#16#1E0F) können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

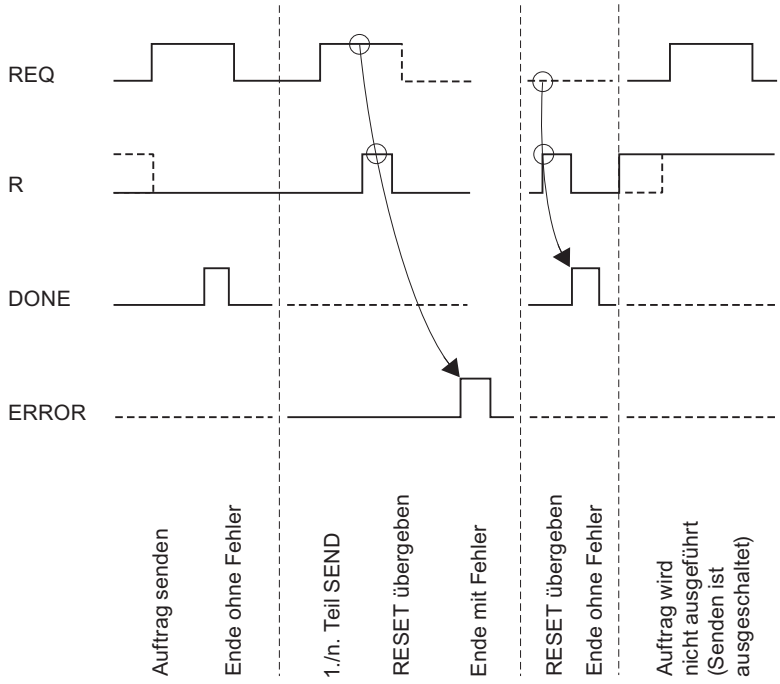

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regeln

Hinweis

Die Anweisung SEND_440 hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

RECV_440: Daten empfangen

Beschreibung

Die Anweisung RECV_440 überträgt Daten von dem CP 440 zu einem S7-Datenbereich, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN. Die Anweisung RECV_440 wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen. .

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von der CP 440 zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R abgebrochen werden. Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange der Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Übertragung über mehrere Aufrufe (Programmzyklen) laufen.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung RECV_440 in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange der Signalzustand "1" am Parameter R ansteht.

Mit dem Parameter LADDR wird der anzusprechende CP 440 ausgewählt.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. NDR und ERROR/STATUS werden auch bei RESET der Anweisung RECV_440 ausgegeben (Parameter LEN == 16#00). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten lesen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt.
LADDR	INPUT	INT	Basisadresse des CP 440 Die Basisadresse wird in STEP 7 aus der Konfigurationstabelle entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer Offset ist CPU-spezifisch
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6686) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 1 ≤ LEN ≤ 400, Angabe in Anzahl Byte

Parameter	Deklaration	Datentyp	Beschreibung
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6686) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung RECV_440 arbeitet mit einem Instanz-DB I_RECV_440 zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall (STATUS == W#16#1E0E) können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter NDR, LEN und ERROR je nach Eingangsbeschaltung von EN_R und R.

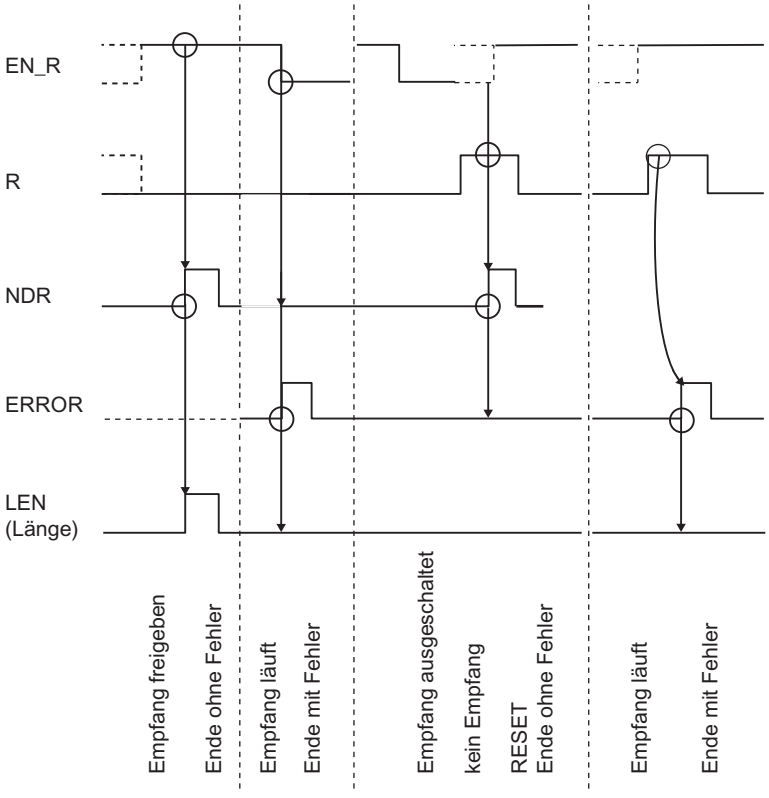

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Regel

Hinweis

Die Anweisung RECV_440 hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

RES_RECV: Empfangspuffer löschen

Beschreibung

Die Anweisung RES_RECV löscht den kompletten Empfangspuffer des CP 440. Alle gespeicherten Telegramme werden verworfen. Ein zum Zeitpunkt des Aufrufs der Anweisung RES_RECV eingehendes Telegramm wird gespeichert.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Anweisung aktiviert. Der Auftrag kann über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung RES_RECV kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Der Löschvorgang wird damit abgebrochen und die Anweisung RES_RECV in den Grundzustand versetzt.

Mit dem Parameter LADDR wird die Adresse des anzusprechenden CP 440 angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. Im STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung RES_RECV ausgegeben. Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse des CP 440 Die Basisadresse wird aus STEP 7 entnommen.
DONE[1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6686) enthält die Fehlerinformation.
STATUS[1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6686) die Fehlerinformation.
[1] Der Parameter steht bis zum nächsten Aufruf der Anweisung zur Verfügung!			

Belegung im Datenbereich

Die Anweisung RES_RECV arbeitet mit einem Instanz-DB I_RES_RECV zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1E0F, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen.

Zeitablaufdiagramm

Im folgenden Bild finden Sie das Verhalten der Parameter DONE und ERROR je nach Eingangsbeschaltung von REQ und R.

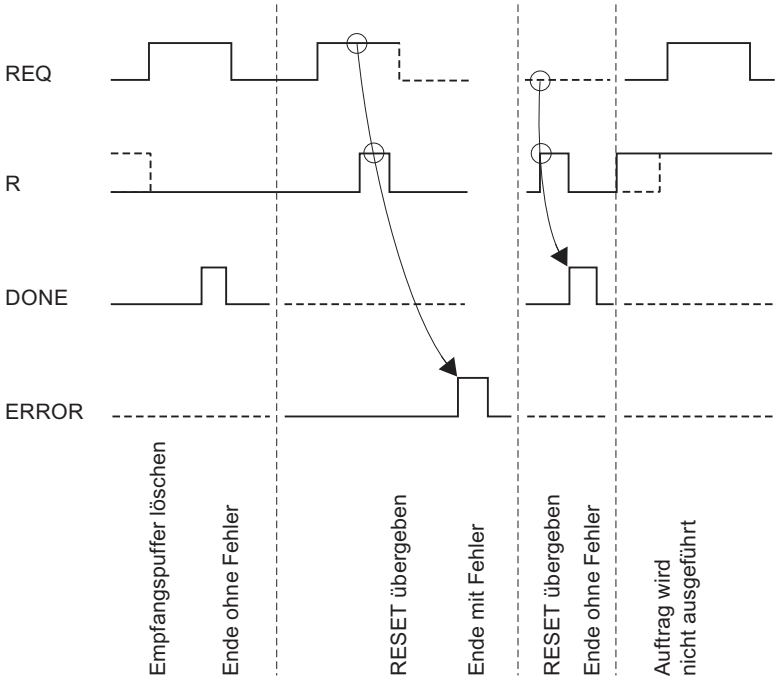

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regel

Hinweis

Die Anweisung RES_RECV hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Parameter STATUS

Ereignisklassen

Ereignisklasse	Bedeutung
0 (0_H)	Anlauf des CP
2 (02_H)	Fehler bei der Initialisierung

Ereignisklasse	Bedeutung
5 (05_H)	Fehler bei Bearbeitung eines CPU-Auftrags
7 (07_H)	Sendefehler
8 (08_H)	Empfangsfehler
11 ($0B_H$)	Warnungen
30 ($1E_H$)	Fehler bei Kommunikation zwischen CP und CPU
128 (80_H)	Baugruppenfehler

Parameter STATUS

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0001	Init des CP abgeschlossen	-
0003	PtP–Parameter übernommen	-
0004	Parameter schon auf CP (Zeitstände gleich)	-
0007	Zustandsübergang CPU in STOP	-
0008	Zustandsübergang CPU in RUN/AN-LAUF	-
0201	Keine Parameter vorhanden Parametrierspeicher leer oder unbekannter Inhalt	Laden Sie die Parameter der Schnittstelle.
0204	Unzulässiger Zeichenrahmen	Korrigieren Sie die unzulässige Parametrierung, laden Sie die Parameter auf die Baugruppe und führen Sie einen Neustart durch.
0205	Unzulässige Baudrate	Korrigieren Sie die unzulässige Parametrierung, laden Sie die Parameter auf die Baugruppe und führen Sie einen Neustart durch.
020F	Beim Start der parametrierten Kommunikation wird eine unzulässige Parametrierung festgestellt. Die Schnittstelle konnte nicht parametriert werden.	Korrigieren Sie die unzulässige Parametrierung, und führen Sie einen Neustart durch.
0212	Unzulässige • Überwachungszeit • Zeichenverzugszeit Nur bei 3964(R): • Quittungsverzugszeit	Korrigieren Sie die unzulässige Parametrierung, laden Sie die Parameter auf die Baugruppe und führen Sie einen Neustart durch.
0501	Laufender Auftrag wurde durch Neuanlauf des CP abgebrochen.	Bei NETZ–EIN ist keine Hilfe möglich. Beim Umparametrieren des CP vom PG aus sollten Sie vor dem Beschreiben einer Schnittstelle darauf achten, dass von der CPU aus keine Aufträge mehr laufen.
0502	Auftrag ist in diesem Betriebszustand des CP nicht erlaubt (z. B. Geräteschnittstelle nicht parametriert).	Parametrieren Sie die Geräteschnittstelle.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
050E	• Ungültige Telegrammlänge oder	• Die Telegrammlänge ist > 400 Bytes. Wählen Sie eine kleinere Telegrammlänge oder
	• die parametrierten Endekennzeichen sind innerhalb der maximal zulässigen Länge nicht aufgetreten.	• ergänzen Sie die Endekennzeichen im Sendepuffer an der gewünschten Stelle.
0517	Übertragungslänge > 400 Byte ist für den CP zu groß	Spalten Sie den Auftrag in mehrere Aufträge mit kleinerer Länge auf.
051D	Sende–/Empfangsauftrag abgebrochen durch • Reset des Kommunikationsbausteins • Umparametrierung	Wiederholen Sie den Aufruf des Kommunikationsbausteins.
0522	Ein neuer SEND–Auftrag wurde gestartet, obwohl der alte Auftrag noch nicht abgeschlossen ist.	Starten Sie den neuen SEND–Auftrag erst, wenn der alte Auftrag mit DONE oder ERROR abgeschlossen ist.
0701	Nur bei 3964(R): Senden der ersten Wiederholung: • Beim Senden des Telegramms wurde ein Fehler erkannt oder • der Partner forderte durch ein negatives Quittungszeichen (NAK) eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der maximalen Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehlernummer gemeldet, die den Fehler beschreibt, der zuerst auftrat.
0702	Nur bei 3964(R): Fehler beim Verbindungsaufbau: Nachdem STX gesendet wurde, wurde NAK oder ein beliebiges Zeichen (außer DLE oder STX) empfangen.	Untersuchen Sie das Fehlverhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät, das in die Übertragungsleitung eingeschaltet wird.
0703	Nur bei 3964(R): Quittungsverzugszeit (QVZ) überschritten: Nach Senden von STX kam keine Antwort vom Partner innerhalb der Quittungsverzugszeit.	Das Partnergerät ist zu langsam oder nicht empfangsbereit, oder es liegt z. B ein Bruch der Sendeleitung vor. Weisen Sie das Fehlverhalten des Partnergeräts ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0704	Nur bei 3964(R): Abbruch durch Partner: Während des laufenden Sendebetriebes wurden vom Partner ein oder mehrere Zeichen empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Weisen Sie dies ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0705	Nur bei 3964(R): Negative Quittung während Senden	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen oder es liegt ein Fehlverhalten des Partnergeräts vor. Weisen Sie dies ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0706	Nur bei 3964(R): Fehler bei Verbindungsende: • Das Telegramm wurde vom Partner am Ende mit NAK oder einem beliebigen Zeichen (außer DLE) abgelehnt oder • das Quittungszeichen (DLE) wurde zu früh empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Weisen Sie dies ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0707	Nur bei 3964(R): Quittungsverzugszeit am Verbindungsende/Antwortüberwachungszeit nach Sendetelegramm überschritten: Nach Verbindungsabbau mit DLE ETX kam innerhalb der QVZ keine Antwort vom Partner.	Das Partnergerät ist zu langsam oder gestört. Weisen Sie dies ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0708	Nur bei ASCII–Treiber: Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
0709	Nur bei 3964(R): Kein Verbindungsaufbau möglich, die Anzahl der erlaubten Aufbauversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter. Überprüfen Sie auch beim Partner, ob die Empfangsfunktion zwischen CPU und CP richtig parametriert ist.
070A	Nur bei 3964(R): Die Daten konnten nicht übertragen werden, die erlaubte Anzahl der Übertragungsversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter.
070B	Nur bei 3964(R): Initialisierungskonflikt ist nicht lösbar, weil beide Partner hochprior eingestellt sind.	Ändern Sie die Parametrierung.
070C	Nur bei 3964(R): Initialisierungskonflikt ist nicht lösbar, weil beide Partner niederprior eingestellt sind.	Ändern Sie die Parametrierung.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0801	Nur bei 3964(R): Erwarten der ersten Wiederholung: Beim Empfangen eines Telegramms wurde ein Fehler erkannt und der CP forderte durch eine negative Quittierung (NAK) beim Partner eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der maximalen Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehlernummer gemeldet, die den Fehler beschreibt, der zuerst auftrat.
0802	Nur bei 3964(R): Fehler beim Verbindungsaufbau: • In Ruhestellung wurden ein oder mehrere beliebige Zeichen (außer NAK oder STX) empfangen oder • nach einem empfangenen STX wurden vom Partner weitere Zeichen gesendet, ohne die Antwort DLE abzuwarten. Nach Netz–EIN des Partners: • während der Partner eingeschaltet wird, empfängt der CP ein undefiniertes Zeichen.	Weisen Sie das Fehlverhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0805	Nur bei 3964(R): Logischer Fehler während des Empfangs: Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Weisen Sie das Fehlverhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0806	Zeichenverzugszeit (ZVZ) überschritten: • Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der ZVZ empfangen oder Nur bei 3964(R): • 1. Zeichen nach Senden von DLE beim Verbindungsaufbau wurde nicht innerhalb der ZVZ empfangen.	Das Partnergerät ist zu langsam oder gestört. Weisen Sie dies ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0807	Telegrammlänge unzulässig: Es wurde ein Telegramm mit der Länge 0 empfangen.	Der Empfang eines Telegramms mit Länge 0 ist kein Fehler. Überprüfen Sie, warum der Kommunikationspartner Telgramme ohne Nutzdaten sendet.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0808	Nur bei 3964(R): Fehler beim Blockprüfzeichen BCC: Der intern gebildete Wert des BCC stimmt nicht mit dem vom Partner am Verbindungsende empfangenen BCC überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Weisen Sie das Fehlverhalten des Partnergerätes ggf. mit einem Schnittstellentestgerät nach, das in die Übertragungsleitung eingeschaltet wird.
0809	Nur bei 3964(R): Wartezeit auf Blockwiederholung abgelaufen	Parametrieren Sie beim Kommunikationspartner die gleiche Blockwartezeit wie am CP 440. Weisen Sie das Fehlverhalten des Kommunikationspartners ggf. mit einem Schnittstellengerät nach, das in die Übertragungsleitung eingeschaltet wird.
080A	Ein freier Empfangspuffer ist nicht vorhanden: Beim Empfang stand kein leerer Empfangspuffer zur Verfügung.	Die Anweisung RECV_440 muss häufiger aufgerufen werden.
080C	Übertragungsfehler: • Ein Übertragungsfehler (Paritätsfehler, Stopbitfehler, Überlauffehler) wurde erkannt. Nur bei 3964(R): • Wird in Ruhestellung ein gestörtes Zeichen empfangen, wird der Fehler sofort gemeldet, damit Störeinflüsse auf der Übertragungsleitung frühzeitig erkannt werden können. Nur bei 3964(R): • Falls dies während des Sende– oder Empfangsbetriebes auftritt, werden Wiederholungen gestartet.	Störungen auf der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind.
080D	BREAK: Empfangsleitung zum Partner ist unterbrochen.	Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
080E	Empfangspufferüberlauf bei nicht freigegebener Flusskontrolle.	Die Anweisung zum Empfangen muss im Anwenderprogramm häufiger aufgerufen werden oder eine Kommunikation mit Flusskontrolle parametriert werden.
0810	Paritätsfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind.
0811	Zeichenrahmenfehler	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0812	Nur bei ASCII–Treiber: Nachdem der CP XOFF gesendet hat oder CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder entsorgen Sie die Daten im CP schneller.
0814	Nur bei ASCII–Treiber: Ein Telegramm oder mehrere Telegramme sind verloren gegangen, da ohne Flusskontrolle gearbeitet wurde.	Arbeiten Sie soweit wie möglich mit Flusskontrolle. Nutzen Sie den gesamten Empfangspuffer. Stellen Sie bei den Grundparametern den Parameter "Reaktion auf CPU STOP" auf "Weiterarbeit".
0816	Die Länge eines empfangenen Telegramms war länger als die maximale vereinbarte Länge.	Korrektur beim Partner erforderlich.
0820	Statischer Fehler bei Aufruf des SFC RD_REC. Der Returnwert RET_VAL des SFCs wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Analysieren Sie die Variable SFCERR aus dem Instanz-DB.
0B01	Empfangspuffer zu mehr als 2/3 gefüllt	Rufen Sie den Empfangsbaustein häufiger auf, um einen Überlauf des Empfangspuffers zu vermeiden.
8000	Baugruppen–Firmware nicht vorhanden	Führen Sie ein Firmware–Update durch.
1E0D	• Auftragsabbruch wegen Neustart, Wiederanlauf, Reset oder	• Wiederholen Sie den Auftrag.
	• weil Baugruppe nicht vorhanden ist.	• Kontrollieren Sie, ob Baugruppe gesteckt ist.
1E0E	Statischer Fehler bei Aufruf des SFC RD_REC. Der Returnwert RET_VAL des SFCs wird Ihnen in der Variablen SFCERR im Instanz–DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz–DB.
1E0F	Statischer Fehler bei Aufruf des SFC WR_REC. Der Returnwert RET_VAL des SFCs wird Ihnen in der Variablen SFCERR im Instanz–DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz–DB.
1E41	Anzahl der am Parameter LEN der Anweisungen angegebenen Bytes unzulässig.	Halten Sie den Wertebereich von 1 bis 400 Bytes ein.

Hinweis

Eine Fehlermeldung wird nur dann ausgegeben, wenn gleichzeitig das Bit ERROR (Auftrag abgebrochen mit Fehler) gesetzt ist. In jedem anderen Fall ist das STATUS–Wort Null.

Variable SFCERR aufrufen

Nähere Informationen zum aufgetretenen Fehler 14 (1E0EH) und 15 (1E0FH) der Ereignisklasse 30 erhalten Sie über die Variable SFCERR.

Die Variable SFCERR können Sie aus dem Instanz–DB der entsprechenden Anweisung laden.

Fehlermeldungen, die in der Variablen SFCERR eingetragen werden finden Sie bei den Systemfunktionen SFC 58 unter "WR_REC: Datensatz in Peripherie schreiben" und SFC 59 unter "RD_REC: Datensatz von Peripherie lesen".

Siehe auch

WR_REC: Datensatz in Peripherie schreiben (Seite 5494)

RD_REC: Datensatz von Peripherie lesen (Seite 5490)

PtP-Kopplung CP 441

Anweisungen für Kommunikation

Einleitung

Der CP 441 kommuniziert nicht über eigene Bausteine, sondern über Anweisungen des Automatisierungssystems S7-400.

Das Automatisierungssystem S7-400 stellt Ihnen eine Reihe von Anweisungen zur Verfügung, die im Anwenderprogramm die Kommunikation zwischen CPU und dem Kommunikationsprozessor CP 441 anstoßen und steuern. Die Anweisungen sind fest im Speicher der CPU hinterlegt.

Anweisungen des Automatisierungssystems S7-400

Die folgende Tabelle zeigt die Anweisungen des Automatisierungssystems S7-400, die für die Kommunikation zwischen CPU und CP 441 zur Verfügung stehen.

Tabelle 4-340 Anweisungen des Automatisierungssystems S7-400

SFB	Bedeutung
BSEND	Mit der Anweisung BSEND können Sie Daten aus einem S7-Datenbereich an einen Kommunikationspartner mit fester Zielangabe senden.
BRCV	Mit der Anweisung BRCV können Sie Daten von einem Kommunikationspartner empfangen und in einen S7-Datenbereich übergeben.
FETCH	Nur bei RK 512: Mit der Anweisung FETCH können Sie Daten von einem Kommunikationspartner holen.
PUT	Nur bei RK 512: Mit der Anweisung PUT können Sie Daten an einen Kommunikationspartner mit dynamisch änderbarer Zielangabe senden.

SFB	Bedeutung
PRINT	Mit der Anweisung PRINT können Sie einen Meldetext mit bis zu vier Variablen auf einen Drucker ausgeben.
STATUS	Mit der Anweisung STATUS können Sie den Status einer Kommunikationsverbindung abfragen.

Weitere Informationen

Eine ausführliche Beschreibung der Anweisungen des Automatisierungssystems S7-400 und deren Parameter finden Sie unter "Übersicht über die Anweisungen zur S7-Kommunikation".

Siehe auch

Übersicht über die Anweisungen zur S7-Kommunikation (Seite 6381)

V24_STAT_441: Begleitsignale an der RS232C-Schnittstelle lesen

Beschreibung

Die Anweisung V24_STAT_441 liest von einer Schnittstelle des CP 441 die RS 232C-Begleitsignale und stellt sie dem Anwender an den Bausteinparametern zur Verfügung. Die Anweisung V24_STAT_441 wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Die RS 232C-Begleitsignale werden mit jedem Aufruf der Anweisung aktualisiert (zyklisches Pollen). Die Schnittstelle wählen Sie aus, indem Sie am der Anweisung V24_STAT_441 als ID die "Lokale ID" einer der Verbindungen angeben, die über diese Schnittstelle laufen.

Das Binärergebnis BIE wird nicht beeinflusst.

Hinweis

Eine positive Eingangsspannung an den RS232 C Eingangssignalen DSR, CTS, DCD und RI wird jeweils auf den Signalzustand "1" der Eingangssignale DSR_IN, CTS_IN, DCD_IN und RI_IN der Anweisung V24_STAT_441 abgebildet.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	VAR_INPUT	BOOL	aktiviert mit steigender Flanke eine Übertragung
ID	VAR_INPUT	WORD	eindeutige Kommunikationsverbindung zu einem Kommunikationspartner

Parameter	Deklaration	Datentyp	Beschreibung
NDR	VAR_OUTPUT	BOOL	steigende Flanke signalisiert, dem Anwenderprogramm stehen neue Empfangsdaten zur Verfügung
ERROR	VAR_OUTPUT	BOOL	steigende Flanke signalisiert Fehler
STATUS	VAR_OUTPUT	WORD	enthält detaillierte Fehleraussage oder Warnung
DTR_OUT	OUTPUT	BOOL	**D**ata **t**erminal **r**eady, CP 441 betriebsbereit (CP 441-Ausgang)
DSR_IN	OUTPUT	BOOL	**D**ata **s**et **r**eady, Kommunikationspartner betriebsbereit (CP 441-Eingang)
RTS_OUT	OUTPUT	BOOL	**R**equest **t**o **s**end, CP 441 sendebereit[1] (CP 441-Ausgang)
CTS_IN	OUTPUT	BOOL	**C**lear **t**o **s**end, Kommunikationspartner kann Daten vom CP 441 empfangen (Antwort auf RTS = ON des CP 441) (CP 441-Eingang)
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, Empfangssignalpegel (CP 441-Eingang)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, Rufzeichen (CP 441-Eingang)

Beispiel

Tabelle 4-341 Beispiel für den Aufruf der Anweisung V24_STAT_441

AWL

```
        CALL FB 5, DB55
                 REQ                 := DB30.DBX10.0
                 ID                  := W#16#1000
                 NDR                 := DB30.DBX10.1
                 ERROR               := DB30.DBX10.2
                 STATUS              := DB30.DBW20
                 DTR_OUT             := DB30.DBX30.0
                 DSR_IN              := DB30.DBX30.1
                 RTS_OUT             := DB30.DBX30.2
                 CTS_IN              := DB30.DBX30.3
                 DCD_IN              := DB30.DBX30.4
                 RI_IN               := DB30.DBX30.5
```

V24_SET_441: Begleitsignale an der RS232C-Schnittstelle schreiben

Beschreibung

Der Anwender kann über die Parametereingänge der Anweisung V24_SET_441 die entsprechenden Schnittstellenausgänge von einer Schnittstelle des CP 441 setzen oder rücksetzen. Die Anweisung V24_SET_441 wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Die Schnittstelle wählen Sie aus, indem Sie am der Anweisung V24_SET_441 als ID die "Lokale ID" einer der Verbindungen angeben, die über diese Schnittstelle laufen.

Das Binärergebnis BIE wird nicht beeinflusst.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	VAR_INPUT	BOOL	aktiviert mit steigender Flanke eine Übertragung
ID	VAR_INPUT	WORD	eindeutige Kommunikationsverbindung zu einem Kommunikationspartner
DONE	VAR_OUTPUT	BOOL	signalisiert mit steigender Flanke fehlerfreien Abschluss des Auftrages
ERROR	VAR_OUTPUT	BOOL	steigende Flanke signalisiert Fehler
STATUS	VAR_OUTPUT	WORD	enthält detaillierte Fehleraussage oder Warnung
RTS	INPUT	BOOL	Request to send, CP 441 sendebereit (CP 441-Ausgang steuern)
DTR	INPUT	BOOL	Data terminal ready, CP 441 betriebsbereit (CP 441-Ausgang steuern)

Beispiel

Tabelle 4-342 Beispiel für den Aufruf der Anweisung V24_SET_441

AWL

```
        CALL FB 6, DB56
                 REQ         := DB40.DBX10.0
                 ID          := W#16#1000
                 DONE        := DB40.DBX10.1
                 ERROR       := DB40.DBX10.2
                 STATUS      := DB40.DBW20
                 RTS         := DB40.DBX30.2
```

```
AWL
                    DTR         := DB40.DBX30.0
```

Fehlermeldebereich SYSTAT

Fehlerbereich des SYSTAT

Der Fehlermeldebereich SYSTAT ist ein Fehlerbereich, der für jede Schnittstelle vorhanden ist. Im SYSTAT werden alle Fehler/Ereignisse eingetragen, die bei der Datenübertragung auf einer Schnittstelle auftreten können.

Hinweis

Da der Auftrag STATUS asynchron zu den übrigen auf einer Schnittstelle laufenden Aufträgen abläuft, kann den Fehlermeldungen kein SFB mit einer bestimmten R_ID zugeordnet werden. Das heißt, es kann zwar angezeigt werden, welche Fehler auf einer Schnittstelle aufgetreten sind, nicht aber von welchem SFB-Aufruf (R_ID-Nummer) der Fehler ausgelöst wurde.

Fehler/Ereignisse

Die SYSTAT - Meldungen werden beim Aufruf des SFB STATUS in die Bytes 2 bis 15 des Parameters LOCAL eingetragen. Zusätzlich zu dem Fehlerbyte (Byte 2) werden die ersten sechs Fehler/Ereignisse angezeigt. Dabei ist das Fehlerereignis 1 das älteste.

Treten weitere Fehlerereignisse auf, können diese nicht gemeldet werden, falls die "alten" Einträge zuvor nicht gelöscht wurden. Daher muss der Fehlermeldebereich rechtzeitig gelöscht werden. Dies geschieht durch Aufruf des SFB STATUS.

Die Fehler/Ereignisse werden wie folgt abgelegt:

- Byte 0 — Betriebszustand des CP (02H für RUN, 05H für Defekt)
- Byte 1 — Reserviert
- Byte 2 — Bit 0 -F Fehler im SYSTAT eingetragen
  Bit 1 -U Fehlerüberlauf
  Bit 2 -B Break
- Byte 3 — Reserviert
- Byte 4/5 — Ereignis 1
- Byte 6/7 — Ereignis 2
- Byte 8/9 — Ereignis 3
- Byte 10/11 — Ereignis 4
- Byte 12/13 — Ereignis 5
- Byte 14/15 — Ereignis 6

Ereignisklassen

Ereignisklasse	Bedeutung
1	Hardwarefehler auf dem CP
2	Fehler bei der Initialisierung
3	Fehler bei der Parametrierung der Anweisungen
4	Vom CP erkannte Fehler im Datenverkehr CP - CPU
5	Fehler bei Bearbeitung eines CPU-Auftrags
6	Fehler bei Bearbeitung eines Partnerauftrags (nur bei RK 512)
7	Sendefehler
8	Empfangsfehler
9	Reaktionstelegramm mit Fehler oder Fehlertelegramm vom Koppelpartner empfangen
10 ($0A_H$)	Fehler beim Reaktionstelegramm des Partners, die vom CP erkannt wurden

Ereignisnummern

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0101	Fehler beim Test des Betriebssystem-EP-ROM des CP	CP defekt, CP muss getauscht werden.
0102	RAM-Test des CP fehlerhaft	
0103	Auftragsschnittstelle des CP defekt	
0104	Kein Schnittstellenmodul gesteckt	Stecken Sie ein für den CP geeignetes Schnittstellenmodul.
0105	• Parameterspeicher defekt • Schnittstellenmodul nach Parametrierung gezogen	Tauschen Sie den CP aus bzw. stecken Sie ein für den CP geeignetes Schnittstellenmodul.
0110	Fehler in der CP-Firmware	Schalten Sie die Baugruppe aus und wieder ein. Tauschen Sie notfalls die Baugruppe.
0201	Keine Parameter vorhanden Parameterspeicher leer oder unbekannter Inhalt	Laden Sie die Parameter der Schnittstelle.
0208	Parametrierung und Schnittstellenmodul passen nicht zusammen.	Bezeichnung des Schnittstellenmoduls mit der Parametrierung überprüfen.
020F	Beim Start der parametrierten Kommunikation wird eine unzulässige Parametrierung festgestellt. Die Schnittstelle konnte nicht parametriert werden.	Korrigieren Sie die unzulässige Parametrierung, und führen Sie einen Neustart durch.
0210	Summenbaudrate überschritten	Reduzieren Sie die Baudrate von beiden Schnittstellen so, dass die Summenbaudrate nicht überschritten wird.
0211	Summenbaudrate wieder im erlaubten Bereich	Die Summenbaudrate wurde zuvor überschritten. Sie befindet sich jetzt wieder im erlaubten Bereich

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0301	Quell-/Zieldatentyp nicht zulässig bzw. nicht vorhanden Bereich (Anfangsadresse, Länge) nicht zulässig DB nicht vorhanden bzw. nicht zulässig (z. B. DB 0) oder anderer Datentyp nicht vorhanden bzw. nicht zulässig	Parametrierung auf CPU und CP prüfen und evtl. korrigieren. **Nur bei RK 512:** Partner liefert unzulässige Parameter im Telegrammkopf. Parametrierung auf CPU und CP prüfen, evtl. Baustein einrichten. Entnehmen Sie den Auftragstabellen die zulässigen Datentypen. **Nur bei RK 512:** Partner liefert falsche Parameter im Telegrammkopf.
0302	Bereich zu kurz	Parametrierung auf CPU und CP prüfen, evtl. Baustein/Bereich prüfen. **Nur bei RK 512:** Partner liefert falsche Parameter im Telegrammkopf.
0303	Kein Zugriff auf Bereich möglich	Parametrierung auf CPU und CP prüfen. Entnehmen Sie den Auftragstabellen die zulässigen Anfangsadressen und Längen. **Nur bei RK 512:** Partner liefert falsche Parameter im Telegrammkopf.
0401	CP kann keine Aufträge annehmen (Überlastung).	Reduzieren Sie die gleichzeitig in Ihrem Anwenderprogramm aufgerufenen Aufträge an den CP.
0402	Auftragsart kann vom CP nicht bearbeitet werden.	Überprüfen Sie, ob die von Ihnen im Anwenderprogramm aufgerufenen Systemfunktionsbausteine für den CP zugelassen sind.
0403	Fehlerhafter, unbekannter oder nicht erlaubter Datentyp	Programm überprüfen, z. B. falsche Parametrierung des SFBs.
0407	Fehler bei der Datenübertragung zwischen CPU und CP. Empfang von Daten nicht möglich, weil auf den Zieldatenbereich der CPU nicht zugegriffen werden kann oder der Zeilbereich der CPU nicht vorhanden oder zu kurz ist. Schreiben von Daten in den Zielbereich bzw. Lesen von Daten aus dem Quelldatenbereich der CPU im CPU-Stop wegen entsprechender Parametrierung nicht zulässig.	Überprüfen Sie den Zieldatenbereich auf der CPU. Überprüfen Sie die Parametrierung "Reaktion auf CPU-Stop".
0408	• **Nur bei 3964R und ASCII Treiber** • Temporärer Fehler bei der Datenübertragung zwischen CPU und CP (Empfang). Auftrag wird wiederholt, weil die CPU temporär überlastet ist, der Empfangsbaustein (BRCV) zu selten aufgerufen wird oder der Empfangsbaustein zwischenzeitlich gesperrt wird.	• Anzahl der Kommunikations-Aufrufe reduzieren. Empfangsbaustein häufiger aufrufen. Überprüfen Sie, ob der Empfangsbaustein zu lange gesperrt wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0409	• **Nur bei 3964R und ASCII Treiber** • Fehler bei der Datenübertragung zwischen CPU und CP. Empfang von Daten nicht möglich. Nach mehrmaligen Versuchen (siehe (04)07H) wurde der Auftrag nach 10s abgebrochen, weil der Empfangsbaustein (BRCV) nicht aufgerufen wird oder gesperrt ist.	• Überprüfen Sie, ob der Empfangsbaustein aufgerufen wird oder gesperrt ist.
040B	Fehler bei der Datenübertragung zwischen CPU und CP weil • keine Verbindung projektiert wurde • über projektierte Verbindung nicht empfangen werden kann	• Projektieren Sie in "NetPro" die Verbindung • Geben Sie in "NetPro" unter "Objekteigenschaften Verbindung" als Kommunikationsrichtung ein: – 2: Partner → Lokal oder – 3: Lokal ↔ Partner
0501	Laufender Auftrag wurde durch Neuanlauf des CP abgebrochen.	Bei NETZ-EIN ist keine Hilfe möglich. Beim Umparametrieren des CP vom PG aus sollten Sie vor dem Beschreiben einer Schnittstelle darauf achten, dass von der CPU aus keine Aufträge mehr laufen.
0502	Auftrag ist in diesem Betriebszustand des CP nicht erlaubt (z. B. Geräteschnittstelle nicht parametriert).	Parametrieren Sie die Geräteschnittstelle.
0503	Uhrzeit fehlerhaft, falsches Format	Überprüfen Sie die Uhrzeit-Parameter.
0505	**Nur bei Drucker-Treiber:** Systemdatenbaustein mit Meldetexten auf CP nicht vorhanden	Projektieren Sie den Meldetext mit der Parametriersoftware, und führen Sie einen Neustart durch.
0506	**Nur bei Drucker-Treiber:** Meldetext nicht vorhanden	Projektieren Sie die Meldetexte mit der Parametriersoftware, und führen Sie einen Neustart durch.
0507	**Nur bei Drucker-Treiber:** Meldetext zu lang	Ändern Sie den Meldetext auf maximal 150 Zeichen (auf maximal 250 Zeichen bei der Verwendung von Variablen)
0508	**Nur bei Drucker-Treiber:** Zu viele Konvertierungsanweisungen	Sie haben mehr Konvertierungsanweisungen als Variablen projektiert. Die Konvertierungsanweisungen ohne dazugehörige Variable werden ignoriert.
0509	**Nur bei Drucker-Treiber:** Zu viele Variablen	Sie haben mehr Variablen als Konvertierungsanweisungen projektiert. Die Variablen ohne Konvertierungsanweisung werden nicht ausgegeben.
050A	**Nur bei Drucker-Treiber:** Unbekannte Konvertierungsanweisung	Überprüfen Sie die Konvertierungsanweisung. Nicht definierte oder nicht unterstützte Konvertierungsanweisungen werden im Ausdruck durch ****** ersetzt.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
050B	**Nur bei Drucker-Treiber:** Unbekannte Steueranweisung	Überprüfen Sie die Steueranweisung. Nicht definierte oder nicht unterstützte Steueranweisungen werden ignoriert. Die Steueranweisung wird auch nicht als Text ausgegeben.
050C	**Nur bei Drucker-Treiber:** Konvertierungsanweisung nicht ausführbar	Überprüfen Sie die Konvertierungsanweisung. Nicht ausführbare Konvertierungsanweisungen werden im Ausdruck entsprechend der definierten Breite und dem gültigen Rest der Konvertierungsanweisung bzw. der Standarddarstellung mit *-Zeichen ausgegeben.
050D	**Nur bei Drucker-Treiber:** Breite in Konvertierungsanweisung zu gering oder zu groß	Korrigieren Sie die angegebene Breite der Variable in der Konvertierungsanweisung anhand der maximal möglichen Zeichenanzahl der Variable bei textorientierten Darstellungsarten (A, C, D, S, T, Y, Z). Im Ausdruck wird nur die durch die Breite angegebene Anzahl Zeichen ausgegeben, der Rest wird abgeschnitten. Ansonsten werden entsprechend der Breite *-Zeichen ausgegeben.
050E	**Nur bei ASCII-Treiber:** Fehler beim Senden. Die parametrierten Endekennzeichen sind innerhalb der maximal zulässigen Länge nicht aufgetreten bzw. beim automatischen Anhängen wurde die maximal zulässige Sendelänge überschritten.	Ergänzen Sie die Endekennzeichen im Sendepuffer an der gewünschten Stelle oder wählen Sie beim automatischen Anhängen eine kleinere Telegrammlänge.
050F	Anzahl der gleichzeitig bearbeitbaren Aufträge zu groß	Ändern Sie Ihr STEP 7-Programm, so dass weniger Aufträge gleichzeitig laufen.
0510	Bereich ist belegt (Ressource).	Wiederholen Sie den Auftrag.
0511	Länge für diese Auftragsart nicht zugelassen.	Teilen Sie die zu übertragenden Daten auf mehrere Aufträge auf.
0512	**Nur bei RK 512:** Die am SFB angegebenen Quell- und Zielparameter passen nicht zueinander.	Entnehmen Sie den Auftragstabellen die zulässigen Werte.
0513	Fehler beim Datentyp (DB ...): Unbekannter Datentyp oder Datentyp nicht erlaubt (z. B. DE) **Nur bei RK 512:** Die am SFB angegebenen Quell- und Zieldatentypen passen nicht zueinander.	Entnehmen Sie den Auftragstabellen die zulässigen Datentypen und deren Kombinationen.
0514	Anfangsadressen zu hoch angegeben für gewünschten Datentyp oder Anfangsadresse bzw. DB/DX-Nr. zu niedrig.	Entnehmen Sie den Auftragstabellen die zulässigen Anfangsadressen und DB/DX-Nummern, die im Programm angegeben werden dürfen.
0515	**Nur bei RK 512:** Falsche Bitnummer beim Koordinierungsmerker angegeben.	Erlaubte Bit-Nr.: 0 bis 7

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0516	**Nur bei RK 512:** CPU-Nummer zu groß angegeben.	Erlaubte CPU-Nr.: keine, 1, 2, 3 oder 4
0517	Fehler beim Empfangen. Empfangstelegramm länger als 4 kByte bzw. länger als die parametrierte "feste Empfangslänge" oder Empfangstelegramm passt nicht in den Zieldatenbereich.	Reduzieren Sie die Telegrammlänge Ihres Koppelpartners oder verlängern Sie Ihren Empfangs-DB.
0518	Übertragungslänge beim Senden ist zu groß (> 4 kByte).	Nur bei RK 512: Entnehmen Sie den Auftragstabellen die zulässigen Längen. Spalten Sie den Auftrag auf in mehrere Aufträge mit kleinerer Länge.
0519	CP in falscher Betriebsart bei AG-Auftrag	Überprüfen Sie, ob die angesprochene Schnittstelle parametriert ist.
051A	**Nur bei RK 512:** Fehler beim Senden eines Befehlstelegramms Eine zugehörige Prozedurfehlernummer wurde unmittelbar zuvor im STATUS eingetragen.	Siehe Abhilfemaßnahmen bei vorangegangener Ereignisnummer.
051B	**Nur bei Drucker-Treiber:** Präzision ungültig	Korrigieren Sie die angegebene Präzision in der Konvertierungsanweisung. Die Präzision wird immer durch einen Punkt eingeleitet zur Kennzeichnung und Abgrenzung zur Breite (z. B.: ".2" zur Ausgabe von Dezimalpunkt und 2 Nachkommastellen). Die Präzision ist nur bei den Darstellungsarten F, R, A und D relevant. Ansonsten wird sie ignoriert.
051C	**Nur bei Drucker-Treiber:** Variable ungültig (Variablenlänge falsch/falscher Typ)	Korrigieren Sie die angegebene Sendevariable.
0601	Fehler im 1. Befehlsbyte (nicht 00 oder FFH)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0602	Fehler im 3. Befehlsbyte (nicht A, 0 oder E)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0603	Fehler im 3. Befehlsbyte bei Folgetelegrammen (Befehl nicht wie beim 1. Telegramm)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0604	Fehler im 4. Befehlsbyte (Befehlsbuchstabe falsch)	Prinzipieller Kopfaufbaufehler beim Partner bzw. eine Befehlskombination wurde verlangt, die beim CP nicht erlaubt ist. Kontrollieren Sie die zulässigen Befehle. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0605	Fehler im 4. Befehlsbyte beim Folgetelegramm (Befehl nicht wie beim 1. Telegramm)	Prinzipieller Kopfaufbaufehler beim Partner. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0606	Fehler im 5. Befehlsbyte (DB-Nummer nicht zulässig)	Entnehmen Sie den Auftragstabellen die zulässigen DB-Nummern, Anfangsadressen oder Längen.
0607	Fehler im 5. bzw. 6. Befehlsbyte (Anfangsadresse zu hoch)	Entnehmen Sie den Auftragstabellen die zulässigen DB-Nummern, Anfangsadressen oder Längen.
0608	Fehler im 7. bzw. 8. Befehlsbyte (Länge unzulässig)	Entnehmen Sie den Auftragstabellen die zulässigen DB/DX-Nummern, Anfangsadressen oder Längen.
0609	Fehler im 9. und 10. Befehlsbyte (Koordinierungsmerker bei diesem Datentyp nicht zulässig oder Bitnummer zu hoch)	Prinzipieller Kopfaufbaufehler beim Partner. Entnehmen Sie den Auftragstabellen, wann ein Koordinierungsmerker erlaubt ist.
060A	Fehler im 10. Befehlsbyte (CPU-Nummer nicht erlaubt)	Prinzipieller Kopfaufbaufehler beim Partner
060B	SEND-Telegramm war länger/ kürzer als erwartet (es wurden mehr/weniger Daten empfangen, als im Telegrammkopf angekündigt).	Korrektur beim Partner erforderlich
060C	GET-Befehlstelegramm mit Nutzdaten empfangen.	Korrektur beim Partner erforderlich
060D	CP hat Telegramm empfangen während unzulässiger Betriebsart. • Die Empfangsverbindung zwischen der CPU und dem CP ist nicht oder nicht korrekt aufgebaut • CP ist noch nicht vollständig angelaufen • Angesprochene Schnittstelle wird gerade umparametriert	• Überprüfen Sie, ob die angesprochene Verbindung (korrekt) parametriert ist. • Diese Fehlermeldung kann nur im Anlauf des CP auftreten. Wiederholen Sie den Auftrag • Hierbei handelt es sich um einen temporären Fehler. Wiederholen Sie den Auftrag.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
060E	Synchronfehler des Partners • Neues (Folge-)Befehlstelegramm empfangen, bevor Reaktionstelegramm gesendet wurde. • 1. Befehlstelegramm erwartet und Folgetelegramm kam. • Folgebefehlstelegramm erwartet und 1. Telegramm kam.	Dieser Fehler kann nach Neustart des eigenen AG bei langen Telegrammen oder bei Neustart des Partners gemeldet werden. In diesen Fällen handelt es sich um normales Anlaufverhalten der Anlage. Im laufenden Betrieb kann der Fehler auch infolge von vorausgegangenen Fehlerzuständen auftreten, die nur vom Partner erkannt wurden. Andernfalls müssen Sie von einem Fehlverhalten des Partnergerätes ausgehen. Evtl. tritt der Fehler nicht auf bei Aufträgen < 128 Byte.
060F	DB gesperrt durch Koordinierungsfunktion	Im eigenen Programm: Nach Bearbeitung der letzten Übertragungsdaten Empfangsbaustein mit "EN" freigeben. Im Partnerprogramm: Wiederholen des Auftrags
0610	Zu kurzes Telegramm empfangen (Länge < 4 Byte bei Folge- oder Reaktionstelegrammen oder Länge < 10 Byte bei Befehlstelegrammen)	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0611	Telegrammlänge und Längenangabe im Telegrammkopf stimmen nicht überein.	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0612	Fehler beim Senden des (Folge-) Reaktions-telegramms, eine zugehörige Prozedur-fehlernummer wurde unmittelbar zuvor im STATUS eingetragen.	Siehe Abhilfemaßnahmen zu der unmittelbar zuvor im STATUS eingetragenen Fehlernummer.
0701	Senden der ersten Wiederholung: • Beim Senden des Telegramms wurde ein Fehler erkannt oder • der Partner forderte durch ein negatives Quittungszeichen (NAK) eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der max. Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehler-Nr. gemeldet, die den Fehler beschreibt, der zuerst auftrat.
0702	Fehler beim Verbindungsaufbau: • Nachdem STX gesendet wurde, wurde NAK oder ein beliebiges Zeichen (außer DLE oder STX) empfangen oder • die Antwort erfolgte zu früh oder • Initialisierungskonflikt ist aufgetreten	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0703	Quittungsverzugszeit (QVZ) überschritten: Nach Senden von STX kam keine Antwort vom Partner innerhalb der Quittungsverzugszeit.	Partnergerät ist zu langsam oder nicht empfangsbereit oder es liegt z. B ein Bruch der Sendeleitung vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0704	Abbruch durch Partner: Während des laufenden Sendebetriebes wurden vom Partner ein oder mehrere Zeichen empfangen	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0705	Negative Quittung während Senden	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0706	Fehler bei Verbindungsende: • Das Telegramm wurde vom Partner am Ende mit NAK oder einem beliebigen Zeichen (außer DLE) abgelehnt oder • das Quittungszeichen (DLE) wurde zu früh empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen oder es liegt ein Fehlverhalten des Partnergeräts vor. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0707	Quittungsverzugszeit am Verbindungsende/Antwortüberwachungszeit nach Sendetelegramm überschritten: Nach Verbindungsabbau mit DLE ETX kam innerhalb der QVZ keine Antwort vom Partner.	Partnergerät ist zu langsam oder gestört. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0708	Nur bei ASCII-Treiber und Drucker-Treiber: Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder Offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
0709	Kein Verbindungsaufbau möglich, die Anzahl der erlaubten Aufbauversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter.
070A	Die Daten konnten nicht übertragen werden, die erlaubte Anzahl der Übertragungsversuche wurde überschritten.	Überprüfen Sie das Schnittstellenkabel oder die Übertragungsparameter.
0801	Erwarten der ersten Wiederholung: Beim Empfangen eines Telegramms wurde ein Fehler erkannt und der CP forderte durch eine negative Quittierung (NAK) beim Partner eine Wiederholung an.	Eine Wiederholung ist kein Fehler, jedoch kann sie ein Hinweis sein, dass Störungen auf der Übertragungsleitung auftreten oder ein Fehlverhalten des Partnergerätes vorliegt. Wenn nach der max. Wiederholungsanzahl das Telegramm immer noch nicht übertragen werden konnte, wird eine Fehler-Nr. gemeldet, die den Fehler beschreibt, der zuerst auftrat.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0802	**Fehler beim Verbindungsaufbau:** • In Ruhestellung wurden ein oder mehrere beliebige Zeichen (außer NAK oder STX) empfangen oder • nach einem empfangenen STX wurden vom Partner weitere Zeichen gesendet, ohne die Antwort DLE abzuwarten. **Nach NETZ-EIN des Partners:** • während der Partner eingeschaltet wird, empfängt der CP ein undefiniertes Zeichen.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0805	Logischer Fehler während des Empfangs: Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Fehlverhalten des Partnergeräts mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0806	Zeichenverzugszeit (ZVZ) überschritten: • Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der ZVZ empfangen oder • 1. Zeichen nach Senden von DLE beim Verbindungsaufbau wurde nicht innerhalb der ZVZ empfangen.	Partnergerät ist zu langsam oder gestört. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0808	Fehler beim Blockprüfzeichen BCC (nur bei RK 512 mit Prozedur 3964R und Prozedur 3964R) Der intern gebildete Wert des BCC stimmt nicht mit dem vom Partner am Verbindungsende empfangenen BCC überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes der Ereignisklasse 8/ Ereignisnummer 12 zu beobachten sein. Fehlerverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
080A	Ein freier Empfangspuffer ist nicht vorhanden: Nach Empfang von STX stand der Prozedur beim Verbindungsaufbau und einer weiteren Wartezeit kein leerer Empfangspuffer zur Verfügung.	Die Anweisung zum Empfangen muss im Anwenderprogramm häufiger aufgerufen werden.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
080C	Übertragungsfehler: Ein Übertragungsfehler (Paritätsfehler, Stopbitfehler, Überlauffehler) wurde erkannt. Wird in Ruhestellung ein gestörtes Zeichen empfangen, wird der Fehler sofort gemeldet, damit Störeinflüsse auf der Übertragungsleitung frühzeitig erkannt werden können. **Nur bei RK 512 und 3964(R):** Falls dies während des Sende- oder Empfangsbetriebes auftritt, werden Wiederholungen gestartet.	Störungen auf der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stopbitanzahl gleich eingestellt sind.
080D	BREAK Die Verbindungsleitung (Empfangsleitung) zum Partnergerät ist unterbrochen.	Stellen Sie die Verbindung zwischen den Geräten her oder schalten Sie das Partnergerät ein. Prüfen Sie bei TTY-Betrieb, ob im Ruhezustand Linienstrom fließt.
0812	**Nur bei ASCII-Treiber:** Nachdem der CP XOFF gesendet hat oder CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder entsorgen Sie den CP schneller.
0815	Einstellung der Übertragungsversuche des CP und des Kommunikationspartners stimmt nicht überein	Parametrieren Sie beim Partner die gleiche Anzahl der Übertragungsversuche wie am CP. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0816	• Die Länge eines empfangenen Telegramms war länger als die maximal vereinbarte Länge oder • die Länge des parametrierten Empfangsfachs (nur bei CP 441) ist zu kurz.	• Korrektur beim Partner erforderlich oder • Länge des Empfangsfachs (nur bei CP 441) vergrößern.
0818	**Nur bei ASCII-Treiber:** DSR = OFF bzw. CTS = OFF	Vor oder während eines Sendevorgangs sind die Signale DSR bzw. CTS vom Partner auf "OFF" geschaltet worden. Überprüfen Sie die Steuerung der RS 232C-Begleitsignale beim Partner.
0902	**Nur bei RK 512:** Speicherzugriffsfehler beim Partner (Speicher nicht vorhanden) Bei SIMATIC S5 als Partner: • Falscher Bereich beim Anzeigenwort oder • Datenbereich nicht vorhanden (außer DB/DX) oder • Datenbereich zu kurz (außer DB/DX)	Kontrollieren Sie, ob der gewünschte Datenbereich beim Partner vorhanden und ausreichend groß ist bzw. prüfen Sie die Parameter des aufgerufenen Systemfunktionsbausteins. Kontrollieren Sie die angegebene Länge am Systemfunktionsbaustein.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0903	**Nur bei RK 512:** DB/DX Zugriffsfehler beim Partner (DB/DX nicht vorhanden oder zu kurz) Bei SIMATIC S5 als Partner: • DB/DX nicht vorhanden oder • DB/DX zu kurz oder • DB/DX-Nr. unzulässig Beim GET-Auftrag zulässiger Quellbereich überschritten	Kontrollieren Sie, ob der gewünschte Datenbereich beim Partner vorhanden und ausreichend groß ist bzw. prüfen Sie die Parameter des aufgerufenen Systemfunktionsbausteins. Kontrollieren Sie die angegebene Länge am Systemfunktionsbaustein.
0904	**Nur bei RK 512:** Partner meldet "Auftragsart nicht erlaubt".	Fehlerhaftes Partnerverhalten, da vom CP 441 niemals ein Systembefehl ausgegeben wird.
0905	**Nur bei RK 512:** Fehler beim Partner bzw. bei SIMATIC S5 als Partner: • Quell-/Zieltyp nicht zulässig oder • Speicherfehler im Partner-AG oder • Fehler bei Verständigung CP/CPU beim Partner oder • Partner-AG ist im STOP-Zustand	Prüfen, ob der Partner den gewünschten Datentyp übertragen kann. Aufbau der Hardware beim Partner prüfen. Schalter des Partner-AG in RUN-Stellung bringen.
0908	**Nur bei RK 512:** Partner erkennt Synchronfehler: Telegrammreihenfolge ist gestört.	Dieser Fehler tritt auf bei Neustart des eigenen AG oder des Partners. Es handelt sich dabei um ein normales Anlaufverhalten der Anlage. Sie brauchen nichts beheben. Bei laufendem Betrieb ist der Fehler auch in Folge von vorausgegangenen Fehlern denkbar. Andernfalls können Sie von einem Fehlverhalten des Partners ausgehen.
0909	**Nur bei RK 512:** DB/DX beim Partner gesperrt durch Koordinierungsmerker	Im Partnerprogramm: nach Bearbeitung der letzten Übertragungsdaten den Koordinierungsmerker wieder rücksetzen! Im Programm: Auftrag wiederholen!
090A	**Nur bei RK 512:** Fehler im Telegrammkopf, die vom Partner erkannt werden: 3. Befehlsbyte im Kopf falsch	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
090B	**Nur bei RK 512:** Fehler im Telegrammkopf: 1. oder 4. Befehlsbyte im Kopf falsch	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
090C	**Nur bei RK 512:** Partner erkennt falsche Telegrammlänge (Gesamtlänge).	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
090D	**Nur bei RK 512:** Bisher erfolgte noch kein Neustart beim Partner.	Beim Partner-AG Neustart durchführen bzw. Betriebsartenwahlschalter am CP oder CPU in Stellung RUN bringen.
090E	**Nur bei RK 512:** Unbekannte Fehlernummer im Reaktionstelegramm empfangen.	Prüfen Sie, ob der Fehler von Störungen oder vom Fehlverhalten des Partners herrührt. Dies mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A01	**Nur bei RK 512:** Synchronfehler des Partners, da • Reaktionstelegramm ohne Auftrag • Reaktionstelegramm empfangen, bevor Folgetelegramm gesendet wurde • nach Senden eines ersten Telegramms Folge-Reaktionstelegramm empfangen wurde • nach Senden eines Folgetelegramms erstes Reaktionstelegramm empfangen wurde	Dieser Fehler wird nach Neustart des eigenen AS bei langen Telegrammen oder bei Neustart des Partners gemeldet. Es handelt sich dabei um ein normales Anlaufverhalten der Anlage, Sie müssen nichts beheben. Im laufenden Betrieb kann der Fehler auch auftreten infolge von vorausgegangenen Fehlerzuständen, die nur vom Partner erkannt wurden. Andernfalls können Sie von einem Fehlverhalten des Partnergerätes ausgehen. Evtl. tritt der Fehler nicht auf bei Aufträgen < 128 Byte.
0A02	**Nur bei RK 512:** Fehler im Aufbau des empfangenen Reaktionstelegramms (1. Byte nicht 00 oder FF)	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A03	**Nur bei RK 512:** Empfangenes Reaktionstelegramm hat zu viele oder zu wenig Daten.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A04	**Nur bei RK 512:** Reaktionstelegramm auf SEND-Auftrag kam mit Daten.	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0A05	**Nur bei RK 512:** Innerhalb der Überwachungszeit kam kein Reaktionstelegramm vom Partner.	Ist der Partner ein sehr langsames Gerät? Oft wird dieser Fehler auch infolge eines vorangegangenen Fehlers angezeigt. Beispielsweise können Prozedurempfangsfehler (Ereignisklasse 8) angezeigt werden, nachdem ein GET-Telegramm gesendet wurde. Grund: Das Reaktionstelegramm konnte wegen Störungen nicht empfangen werden, die Überwachungszeit verstreicht. Evtl. tritt dieser Fehler auch auf, wenn beim Partner ein Neustart durchgeführt wurde, bevor er das zuletzt erhaltene GET-Telegramm beantworten konnte.
0A06	**Nur bei RK 512:** Empfangenes Reaktionstelegramm nach GET-Auftrag hat zu wenig Daten	Fehlverhalten des Partnergerätes ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.

Modbus Slave (RTU)

Modbus-Anweisungen

Einleitung

Für Modbus Master sind keine separaten Modbus-Anweisungen für die Kommunikation erforderlich.

Anweisungen des CP 341

Die Datenübergabe zwischen CP 341 und CPU erfolgt mit den Anweisungen P_SND_RK und P_RCV_RK.

Tabelle 4-343 Anweisungen des CP 341

Anweisung	Bedeutung
P_RCV_RK	Mit der Anweisung P_RCV_RK können Sie Daten von einem Kommunikationspartner empfangen und in einem Datenbaustein hinterlegen oder Daten für den Kommunikationspartner bereitstellen.
P_SND_RK	Mit der Anweisung P_SND_RK können Sie den gesamten Bereich oder einen Teilbereich eines Datenbausteins an einen Kommunikationspartner senden oder Daten vom Kommunikationspartner holen.

Anweisungen des Automatisierungssystems S7-400

Die Datenübergabe zwischen CP 441 und CPU erfolgt mit den Anweisungen BSEND und BRCV.

Tabelle 4-344 Anweisungen des Automatisierungssystems S7-400

SFB	Bedeutung
BSEND	Mit der Anweisung BSEND können Sie Daten aus einem S7-Datenbereich an einen Kommunikationspartner mit fester Zielangabe senden.
BRCV	Mit der Anweisung BRCV können Sie Daten von einem Kommunikationspartner empfangen und in einen S7-Datenbereich übergeben.

MODB_341: Modbus-Slave-Anweisung für CP 341

Beschreibung

Mit der Anweisung MODB_341 und dem dazugehörenden Treiber wird eine Kommunikationsverbindung zwischen einem Modbus-Master-Steuerungssystem und der Kommunikationsbaugruppe CP 341 als "modbusfähiges" Slavesystem ermöglicht.

Arbeitsweise

Im Anwenderprogramm ist die Anweisung MODB_341 im zyklischen Programm aufzurufen. Die Anweisung benutzt den zugehörigen Instanz-DB IDB_MODB_341 als Arbeitsbereich.

Die Anweisung MODB_341 initialisiert den CP und führt die Modbus-Funktionen aus, die der Treiber nicht selbständig durchführen kann.

Nach jedem Neustart oder Wiederanlauf der CPU muss eine Initialisierung der Anweisung MODB_341 durchgeführt werden. Die Initialisierung wird mit einer steigenden Flanke am Eingang CP_START aktiviert. Die Anweisung löscht zunächst den Instanz-DB, liest die Operandenbereiche E, A, M, T, Z mit der Systemanweisung SZL_LESEN aus der CPU aus und legt diese im Instanz-DB ab. Dadurch können die Schreib-Anforderungen vom Modbus-Master-System auf Bereichs-Überschreitung geprüft werden.

Mit einem SEND-Auftrag wird dem CP die Nummer des Instanz-DBs und der bisher erfolgreiche Verlauf der Initialisierung mitgeteilt.

Sobald der SEND-Auftrag ohne Fehler beendet ist, wird der Ausgang CP_START_OK gesetzt und die Initialisierung der Anweisung ist abgeschlossen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	Int	Basisadresse des CP Belegung aus HW Konfig übernehmen
START_TIMER	INPUT	Timer	Timer für Überwachungszeit Start
START_TIME	INPUT	S5Time	Zeitwert Überwachungszeit Start
OB_MASK	INPUT	BOOL	Peripheriezugriffsfehler maskieren, Alarme verzögern FALSE: Zugriffsfehler auf die Peripherie werden nicht maskiert TRUE: Zugriffsfehler auf die Peripherie werden maskiert und die Alarme verzögert
CP_START	INPUT	BOOL	Anweisung Initialisierung starten
CP_START_FM	INPUT	BOOL	Die Initialisierung wird mit der steigenden Flanke von CP_START aktiviert.
CP_START_NDR	OUTPUT	BOOL	Info: Schreib-Anforderung vom CP
CP_START_OK	OUTPUT	BOOL	Anlauf ohne Fehler beendet TRUE: Vor Ablauf der Überwachungszeit konnte der Initialisierungsauftrag fehlerfrei ausgeführt werden.
CP_START_ERROR	OUTPUT	BOOL	Anlauf mit Fehler abgebrochen TRUE: Auch nach Ablauf der Überwachungszeit konnte der Initialisierungsauftrag nicht fehlerfrei ausgeführt werden.

Parameter	Deklaration	Datentyp	Beschreibung
ERROR_NR	OUTPUT	Word	Fehlernummer Belegung siehe Parameter ERROR_NR, ERROR_INFO (Seite 6714)
ERROR_INFO	OUTPUT	Word	Fehlerzusatzinfo Belegung siehe Parameter ERROR_NR, ERROR_INFO (Seite 6714)

Verhalten im Fehlerfall

Wird der SEND-Auftrag mit Fehler beendet, wird CP_START zurückgesetzt und CP_START_ERROR gesetzt.

Wurde die Initialisierung mit Fehler beendet, ist keine Modbus-Kommunikation möglich. Alle Anforderungen vom Modbus-Master-System werden mit einem Exception-Code-Telegramm beantwortet.

Nach Netz-Ein benötigt der CP für den Hardware- und Speichertest mehrere Sekunden bis er betriebsbereit ist. Die während dieser Zeit von der Anweisung MODB_341 durchgeführten Initialisierungsversuche werden mit Fehler beendet. Deshalb wiederholt die Anweisung ihren Initialisierungsauftrag während der Überwachungszeit mehrfach.

Kann innerhalb der parametrierten START-TIME des START-TIMERS die Initialisierung fehlerfrei beendet werden, wird CP_START_OK gesetzt. Konnte auch nach Ablauf der Überwachungszeit der Initialisierungsauftrag nicht fehlerfrei ausgeführt werden, wird CP_START_ERROR gesetzt.

MODB_441: Modbus-Slave-Anweisung für CP 441

Beschreibung

Mit der Anweisung MODB_441 und dem dazugehörenden Treiber wird eine Kommunikationsverbindung zwischen einem Modbus-Master-Steuerungssystem und der Kommunikationsbaugruppe CP 441-2 als "modbusfähiges" Slavesystem ermöglicht.

Arbeitsweise

Im Anwenderprogramm ist die Anweisung MODB_441 im zyklischen Programm aufzurufen. Die Anweisung benutzt den zugehörigen Instanz-DB IDB_MODB_441 als Arbeitsbereich.

Die Modbus-Funktionscodes 01, 02, 03, 04, 06, 16 werden direkt vom CP bearbeitet. Bei den Funktionscodes 05, 15 übernimmt die Anweisung den bitweisen Eintrag der Daten in den SIMATIC-Speicherbereich.

Die Anweisung MODB_441 initialisiert den CP und führt die Modbus-Funktionen aus, die der Treiber nicht selbständig durchführen kann.

Nach jedem Neustart oder Wiederanlauf der CPU muss eine Initialisierung der Anweisung MODB_441 durchgeführt werden. Die Initialisierung wird mit einer steigenden Flanke am Eingang CP_START aktiviert. Die Anweisung löscht zunächst den Instanz-DB, liest die Operandenbereiche E, A, M, T, Z mit der Systemanweisung SZL_LESEN aus der CPU aus

und legt diese im Instanz-DB ab. Dadurch können die Schreib-Anforderungen vom Modbus-Master-System auf Bereichs-Überschreitung geprüft werden.

Mit einem SEND-Auftrag wird dem CP die Nummer des Instanz-DB´s und der bisher erfolgreiche Verlauf der Initialisierung mitgeteilt.

Sobald der SEND-Auftrag ohne Fehler beendet ist, wird der Ausgang CP_START_OK gesetzt und die Initialisierung der Anweisung ist abgeschlossen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
ID	INPUT	Word	Verbindungs-ID Verbindungs-ID aus Verbindungsprojektierung
START_TIMER	INPUT	Timer	Timer für Überwachungszeit Start
START_TIME	INPUT	S5Time	Zeitwert Überwachungszeit Start
STATUS_TIMER	INPUT	Timer	Timer für SYSTAT lesen
STATUS_TIME	INPUT	S5Time	Zeitintervall für SYSTAT lesen
OB_MASK	INPUT	BOOL	Peripheriezugriffsfehler maskieren, Alarme verzögern FALSE: Zugriffsfehler auf die Peripherie werden nicht maskiert. TRUE: Zugriffsfehler auf die Peripherie werden maskiert und die Alarme verzögert.
CP_START	INPUT	BOOL	Anweisung Initialisierung starten
CP_START_FM	INPUT	BOOL	Die Initialisierung wird mit der steigenden Flanke von CP_START aktiviert
CP_START_NDR	OUTPUT	BOOL	Info: Schreib-Anforderung vom CP
CP_START_OK	OUTPUT	BOOL	Anlauf ohne Fehler beendet TRUE: Vor Ablauf der Überwachungszeit konnte der Initialisierungsauftrag fehlerfrei ausgeführt werden.
CP_START_ERROR	OUTPUT	BOOL	Anlauf mit Fehler abgebrochen TRUE: Auch nach Ablauf der Überwachungszeit konnte der Initialisierungsauftrag nicht fehlerfrei ausgeführt werden.
ERROR_NR	OUTPUT	Word	Fehlernummer Belegung siehe Parameter ERROR_NR, ERROR_INFO (Seite 6714)
ERROR_INFO	OUTPUT	Word	Fehlerzusatzinfo Belegung siehe Parameter ERROR_NR, ERROR_INFO (Seite 6714)

Verhalten im Fehlerfall

Wird der SEND-Auftrag mit Fehler beendet, wird CP_START zurückgesetzt und CP_START_ERROR gesetzt.

Wurde die Initialisierung mit Fehler beendet, ist keine Modbus-Kommunikation möglich. Alle Anforderungen vom Modbus-Master-System werden mit einem Exception-Code-Telegramm beantwortet.

Nach Netz-Ein benötigt der CP für den Hardware- und Speichertest mehrere Sekunden bis er betriebsbereit ist. Die während dieser Zeit von der Anweisung MODB_441 durchgeführten Initialisierungsversuche werden mit Fehler beendet. Deshalb wiederholt die Anweisung ihren Initialisierungsauftrag während der Überwachungszeit mehrfach.

Kann innerhalb der parametrierten START-TIME des START-TIMERS die Initialisierung fehlerfrei beendet werden, wird CP_START_OK gesetzt. Konnte auch nach Ablauf der Überwachungszeit der Initialisierungsauftrag nicht fehlerfrei ausgeführt werden, wird CP_START_ERROR gesetzt.

Parameter ERROR_NR, ERROR_INFO

Fehlerklassen

Fehlernummern von ... bis	Beschreibung	Bedeutung
1 ... 9	Fehler bei der Initialisierung Anweisung und CP	Bei den Fehlernummern 1...9 wurde die Initialisierung mit Fehler beendet. Der Parameter CP_START_ERROR ist 1. Es ist keine MODBUS-Kommunikation zum Master-System möglich.
10 ... 19	Fehler bei der Bearbeitung eines Funktionscodes	Bei den Fehlernummern 10...19 ist bei der Bearbeitung eines Funktionscodes ein Fehler aufgetreten. Der CP hat dem Kommunikations- Anweisung einen unzulässigen Bearbeitungsauftrag gesendet. Der Fehler wird ebenfalls an den Treiber gemeldet. Nachfolgende Bearbeitungs-Aufträge werden weiter bearbeitet.
90 ... 99	Sonstige Fehler	Es ist ein Verarbeitungsfehler aufgetreten. Der Fehler wird nicht an den Treiber gemeldet. Nachfolgende Bearbeitungs-Aufträge werden weiter bearbeitet.

Parameter ERROR_NR, ERROR_INFO

ERROR_NR (dezimal)	ERROR_INFO	Fehlertext	Abhilfe
0	0	Kein Fehler	
1	SFC 51->RET_VAL	Fehler beim Lesen der SZL mit der SFC 51.	RET_VAL in ERROR_INFO analysieren, Ursache beseitigen.
2		TimeOut bei CP-Initialisierung oder Fehler bei CP-Initialisierung	Prüfen Sie, ob für die CP-Schnittstelle als Protokoll "MODBUS Slave" parametriert wurde. Analysieren Sie ERROR_INFO.
	P_SND_RK -> STATUS	Fehler beim P_SND_RK-Auftrag	Überprüfen Sie an der Kommunikations-Anweisung die Baugruppenadresse LADDR.
	SFB 12->STATUS	Fehler beim BSEND-Auftrag	Überprüfen Sie am Kommunikations- Anweisung die Verbindungs-ID.
10	Bearbeitungs-Code	Unzulässige Bearbeitungsfunktion vom Treiber an die Kommunikations- Anweisung übergeben.	Neuanlauf des CP (Netz_EIN)
11	Start-Adresse	Unzulässige Start-Adresse vom Treiber an die Kommunikations- Anweisung übergeben.	MODBUS-Adresse vom MODBUS-Master-System kontrollieren.
12	Register-Anzahl	Unzulässige Register-Anzahl vom Treiber an die Kommunikations- Anweisung übergeben: Register-Anzahl = 0.	Register-Anzahl vom MODBUS-Master-System kontrollieren, gegebenenfalls Neuanlauf des CP (Netz_EIN)
13	Register-Anzahl	Unzulässige Register-Anzahl vom Treiber an die Kommunikations- Anweisung übergeben: Register-Anzahl > 128.	Register-Anzahl vom MODBUS-Master-System kontrollieren, gegebenenfalls Neuanlauf des CP (Netz_EIN)
14	Merker M -End-Adresse	Zugriffsversuch auf den SIMATIC-Speicherbereich "Merker" über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist CPU-Typ-abhängig.	MODBUS-Start-Adresse bzw. Zugriffslänge im MODBUS-Master-System verringern.
15	Ausgänge A -End-Adresse	Zugriffsversuch auf den SIMATIC-Speicherbereich "Ausgänge" über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist CPU-Typ-abhängig.	MODBUS-Start-Adresse bzw. Zugriffslänge im MODBUS-Master-System verringern.
16	Zeiten T -End-Adresse	Zugriffsversuch auf den SIMATIC-Speicherbereich "Zeiten (Timer)" über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist CPU-Typ-abhängig.	MODBUS-Start-Adresse bzw. Zugriffslänge im MODBUS-Master-System verringern.

ERROR_NR (dezimal)	ERROR_INFO	Fehlertext	Abhilfe
17	Zähler Z -End-Adresse	Zugriffsversuch auf den SIMATIC-Speicherbereich "Zähler (Counter)" über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist CPU-Typ-abhängig.	MODBUS-Start-Adresse bzw. Zugriffslänge im MODBUS-Master-System verringern.
18	0	Unzulässiger SIMATIC-Speicherbereich vom Treiber an die Kommunikations- Anweisung übergeben.	Gegebenenfalls Neuanlauf des CP (Netz_EIN)
19		Fehler beim Zugriff auf die SIMATIC-Peripherie.	Überprüfen Sie , ob die benötigte Peripherie vorhanden und fehlerfrei ist.
90	P_SND_RK -> STATUS	Fehler beim Senden einer Quittungsmeldung mit der Anweisung P_SND_RK an den Treiber	STATUS-Information analysieren
	SFB 12 -> STATUS	Fehler beim Senden einer Quittungsmeldung mit dem SFB 12 BSEND an den Treiber	STATUS-Information analysieren
91	SFB 22 -> STATUS	Fehler beim Lesen des SYSTAT mit dem SFB 22 (STATUS)	STATUS-Information analysieren
92	P_RCV_RK -> STATUS	Fehler beim Ausführen eines RECEIVE/ FETCH-Aufrufs mit der Anweisung P_RCV_RK	P_RCV_RK-STATUS analysieren

Fehlermeldebereich SYSTAT

Ereignisklassen

Ereignisklasse	Bedeutung
5	Fehler bei Bearbeitung eines CPU-Auftrags
8	Empfangsfehler
14 ($0E_H$)	Allgemeiner Verarbeitungsfehler

Parameter STATUS

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0518	Übertragungslänge beim Senden ist zu groß (> 4 kByte), oder die Übertragungslänge beim SEND ist zu klein.	Kommunikations- Anweisung überprüfen, gegebenenfalls neu laden.
0806	Zeichenverzugszeit ZVZ überschritten	Fehler beim Partnergerät oder Störungen auf der Übertragungsleitung beseitigen.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
080C	Bei einem Zeichen wurde ein Übertragungsfehler erkannt (Paritätsfehler, Overrunfehler, Stoppbitfehler (Frame)).	Prüfen Sie, ob Störungen auf die Übertragungsleitung einwirken. Ändern Sie gegebenenfalls. den Systemaufbau bzw. die Leitungsverlegung. Prüfen Sie, ob die Protokollparameter Baudrate, Datenbitanzahl, Parität, Stopbitanzahl bei CP und Koppelpartner gleich eingestellt sind.
080D	BREAK Die Empfangsleitung zum Partnergerät ist unterbrochen.	Stellen Sie die Verbindung zwischen den Geräten her oder schalten Sie das Partnergerät ein. Prüfen Sie bei TTY-Betrieb, ob im Ruhezustand Linienstrom fließt. Überprüfen und ändern Sie gegebenenfalls. bei einer RS422/485(X27)-Verbindung die Vorbelegung der 2-Draht Empfangsleitung R(A), R(B).
0830	Broadcast bei diesem Funktionscode nicht erlaubt.	Das MODBUS-Master-System darf Broadcast nur bei den dafür freigegebenen Funktionscodes verwenden.
0831	Empfangener Funktionscode nicht erlaubt.	Dieser Funktionscode kann bei diesem Treiber nicht verwendet werden.
0832	Maximale Anzahl von Bits oder Register überschritten, oder Bitanzahl nicht durch 16 teilbar bei Zugriff auf SIMATIC-Speicherbereich Zeiten oder Zähler.	Maximale Anzahl von Bits auf 2040, maximale Anzahl von Registern auf 127 beschränken. Zugriff auf SIMATIC-Zeiten/Zähler nur im 16-Bit-Raster.
0833	Bitanzahl oder Registeranzahl bei Funktionscode FC 15/16 und Telegrammelement byte_count passen nicht zusammen.	Bitanzahl, Registeranzahl oder byte_count korrigieren.
0834	Unzulässige Bit-Codierung für "Bit setzen"/Bit rücksetzen" erkannt.	Bei FC 05 nur die Codierungen 0000Hex oder FF00Hex verwenden.
0835	Unzulässiger Diagnostic-Subcode (≠ 0000Hex) bei Funktionscode FC 08 "Loop Back Test" erkannt.	Bei FC 08 nur den Subcode 0000Hex verwenden.
0836	Der intern gebildete Wert der CRC-16-Prüfsumme stimmt nicht mit der empfangenen CRC-Prüfsumme überein.	Bildung der CRC-Prüfsumme beim MODBUS-Master-System überprüfen.
0837	Telegramm-Ablauf-Fehler: Das MODBUS-Master-System sendete ein neues Anforderungstelegramm, ehe vom Treiber das letzte Antworttelegramm übertragen wurde.	Erhöhen Sie den TimeOut auf das Slave-Antworttelegramm beim MODBUS-Master-System.
0E01	Fehler bei der Initialisierung des Treiberspezifischen SCC-Prozesses.	Treiber neu parametrieren und laden.
0E02	Fehler bei Anlauf des Treibers: Falscher SCC-Prozess aktiv (SCC-Treiber). Der Treiber ist mit diesem SCC-Treiber nicht funktionsfähig.	Treiber neu parametrieren und laden.
0E03	Fehler bei Anlauf des Treibers: Falscher Datenübergabe-Prozess aktiv (Schnittstelle zu den SFBs). Der Treiber ist mit diesem Datenübergabe-Prozess nicht funktionsfähig.	Treiber neu parametrieren und laden.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E04	Fehler bei Anlauf des Treibers: Unzulässiges Schnittstellenmodul. Der Treiber kann mit dem parametrierten Schnittstellenmodul nicht betrieben werden.	Parametrierung überprüfen und korrigieren.
0E05	Fehler bei Treiber-Dongle: Es ist kein Dongle gesteckt oder der gesteckte Dongle ist fehlerhaft. Der Treiber ist nicht betriebsbereit.	Überprüfen Sie, ob im CP ein Treiber-Dongle gesteckt ist. Wenn ein Dongle gesteckt ist, ersetzen Sie den fehlerhaften Dongle durch einen korrekten Dongle.
0E06	Fehler bei Treiber-Dongle: Der Dongle hat keinen gültigen Inhalt. Der Treiber ist nicht betriebsbereit.	Besorgen Sie sich bei Ihrer Siemens Niederlassung, von der Sie den Treiber erhalten haben, einen korrekten Dongle.
0E10	Interner Fehler Prozedur: Default-Zweig im Sende-Automat.	Neuanlauf des CP (Netz_EIN)
0E11	Interner Fehler Prozedur: Default-Zweig im Empfangs-Automat.	Neuanlauf des CP (Netz_EIN)
0E12	Interner Fehler Aktiv-Automat: Default-Zweig	Neuanlauf des CP (Netz_EIN)
0E13	Interner Fehler Passiv-Automat: Default-Zweig	Neuanlauf des CP (Netz_EIN)
0E20	Die Anzahl der Datenbits bei dieser Kopplung muss 8 betragen. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E21	Der parametrierte Multiplikationsfaktor für die Zeichenverzugszeit ist nicht im Bereich von 1 bis 10. Der Treiber arbeitet mit der Standardeinstellung von 1.	Korrigieren Sie die Parametrierung des Treibers.
0E22	Die parametrierte Betriebsart des Treibers ist unzulässig. Es ist "Normal"- oder "Störungsunterdrückungs"-Betrieb anzugeben. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E23	Als Slave-Adresse wurde ein unzulässiger Wert parametriert. Slave-Adresse 0 ist nicht zulässig. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E24	Es wurden unzulässige Grenzen für schreibende Zugriffe parametriert. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E25	Es wurde eine unzulässige "von/bis"-Kombination bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 01, 05, 15" parametriert. (Bereich Merker, Ausgänge, Zeiten, Zähler). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E26	Es wurde eine unzulässige "von/bis"-Kombination bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 02" parametriert. (Bereich Merker, Eingänge). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E27	Es wurde eine Überlappung bei den "von/bis"-Kombinationen bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 01, 05, 15" parametriert. (Bereich Merker, Ausgänge, Zeiten, Zähler). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E28	Es wurde eine Überlappung bei den "von/bis"-Kombinationen bei der Eingabe der Bereiche "Umsetzung der Modbus-Adressierung bei FC 02" parametriert. (Bereich Merker, Eingänge). Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E2E	Beim Lesen der Schnittstellenparameter-Datei ist ein Fehler aufgetreten. Der Treiber ist nicht betriebsbereit.	Neuanlauf des CP (Netz_EIN).
0E30	Interner Fehler Übergabe an CPU: Unerwartete Quittung Passiv.	Kann bei sporadischem Auftreten ignoriert werden.
0E31	Time Out bei Übergabe von Daten an die CPU.	Schnittstelle CP-CPU überprüfen.
0E32	Fehler bei Übergabe von Daten mit RCV an die CPU: Ursache analysieren - Detailfehler steht vor diesem Eintrag im SYSTAT.	Schnittstelle CP-CPU überprüfen.
0E33	Interner Fehler bei Datenübergabe CP-CPU: Unzulässiger Automatenzustand	Schnittstelle CP-CPU überprüfen.
:		
0E38	Beim Zugriff auf einen der SIMATIC-Bereiche "Merker, Ausgang, Zeiten, Zähler, Eingang" mit dem Funktionscode FC 01 oder FC 02 ist ein Fehler aufgetreten: Z. B. Eingang nicht vorhanden, oder Leseversuch über das Bereichsende hinaus.	Prüfen Sie, ob der angesprochene SIMATIC-Bereich vorhanden ist und ob nicht versucht wurde, über das Bereichsende hinaus zuzugreifen.
0E39	Beim Zugriff auf den SIMATIC-Bereich "Datenbaustein" mit dem Funktionscode FC 03, 04, 06, 16 ist ein Fehler aufgetreten: Datenbaustein nicht vorhanden oder zu kurz.	Prüfen Sie, ob der angesprochene Datenbaustein vorhanden und ausreichend lang ist.
0E3A	Bei der Ausführung eines Schreibauftrags mit dem Funktionscode FC 05, 15 ist ein Fehler aufgetreten: Instanz-Datenbaustein der MODBUS- Anweisung nicht vorhanden oder zu kurz.	Prüfen Sie, ob der an der MODBUS-Kommunikations- Anweisung parametrierte Instanz-DB vorhanden und ausreichend lang ist.
0E3B	TimeOut (warten auf Quittung) bei der Ausführung eines Schreibauftrags durch die Modbus-Kommunikations- Anweisung.	Verbindungsprojektierung sowie Schnittstelle CP-CPU überprüfen (SFB BSEND), gegebenenfalls MODBUS-Kommunikations- Anweisung neu laden.
0E3C	Unzulässiger Auftrag bei diesem Treiber.	Nur SFB SEND, STATUS (nur bei CP 441-2) zulässig
0E50	Bei den wortorientierten SIMATIC-Bereichen Zeiten/Zähler ist die sich aus der Modbus-Adresse ergebende Rest-Bitnummer ≠ 0.	Nur MODBUS-Adressen verwenden, bei denen sich gültige Bit-Nummern ergeben.
0E51	Die empfangene Modbus-Adresse liegt ausserhalb der parametrierten "von/bis"-Bereiche.	Als Adressangabe im Anforderungstelegramm nur Adressen verwenden, die vorher bei der Parametrierung definiert wurden.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E52	• Überschreitung einer SIMATIC-Bereichsgrenze bei Zugriffsversuch durch das Modbus-Master-System: Resultierende DB-Nr. < 1 • Schreib-Zugriff auf einen nicht freigegebenen Bereich (Parametrierung) • Schreibzugriff auf den Instanz-DB der Kommunikations- Anweisung	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.
0E53	Überschreitung einer SIMATIC-Bereichsgrenze bei Zugriffsversuch durch das Modbus-Master-System. Z. B. Überlauf bei der Bildung der resultierenden DB-Nr. (> 65535)	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.
0E54	Zugriff über das parametrierte Bereichsende hinaus, oder Zugriff über das SIMATIC-Bereichsende hinaus.	Zugriffs-Bereich auf gültige SIMATIC-Speicherbereiche beschränken.
0E55	Ein Schreib-Zugriff auf diesen SIMATIC-Speicherbereich ist nicht erlaubt.	Schreibende Zugriffe nur auf die SIMATIC-Datenbereiche Merker, Ausgänge durchführen.
0E56	Ein Betrieb der Kopplung ist nicht möglich, da die Kommunikations- Anweisung nicht läuft.	Rufen Sie die MODBUS-Kommunikations- Anweisung im STEP 7-Anwenderprogramm zyklisch auf. Führen Sie gegebenenfalls nochmal eine Initialisierung der Kommunikations- Anweisung durch.
0E57	Bei der Abarbeitung des Modbus-Funktionscodes ist in der Kommunikations- Anweisung ein Fehler aufgetreten.	Analysieren Sie die genaue Ursache.

ET 200S Serielle Schnittstelle

ET 200S 1SI für 3964(R) und ASCII

S_RCV: Daten empfangen

Beschreibung

Die Anweisung S_RCV überträgt Daten von der Baugruppe ET 200S 1SI zu einem S7-Datenbereich, spezifiziert durch die Parameter DB_NO und DBB_NO. Die Anweisung S_RCV wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von der Baugruppe ET 200S 1SI zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R und zusätzlich mit Signalzustand "1" am Parameter R abgebrochen werden. (Da es sporadisch vorkommen kann, dass nach Setzen des Parameters EN_R auf "1" die Übertagung nicht mehr anläuft, sollten Sie zum Abbrechen einer laufenden Übertragung parallel zum Setzen des EN_R auf "0" den Parameter R auf "1" setzen.) Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung S_RCV in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht. Bei erneutem Signalzustand "0" wird das abgebrochene Telegramm erneut, von Anfang an, empfangen.

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). ERROR zeigt einen aufgetretenen Fehler an. In STATUS wird bei einem Fehler die entsprechende Fehlernummer angezeigt. Ist der Empfangspuffer mehr als 2/3 voll, enthält STATUS nach jedem Aufruf von S_RCV eine Warnung. Liegen keine Fehler oder Warnungen vor, hat STATUS den Wert "0".

NDR und ERROR/STATUS werden auch dann ausgegeben, wenn die Anweisung S_RCV zurückgesetzt wird (Parameter LEN == 16#00) (siehe Zeitablaufdiagramm). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten lesen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer (Empfangsdaten) **S7-1500**: 0 ≤ DBB_NO und DBB_NO + LEN ≤ Größe des DB **S7-300/400**: 0 ≤ DBB_NO und DBB_NO + LEN < 8190

Parameter	Deklaration	Datentyp	Beschreibung
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 1 ≤ LEN ≤ 224 Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter NDR; ERROR, LEN und STATUS stehen nach korrektem Empfangsauftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_RCV arbeitet mit einem Instanz-DB I_RCV zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Anlauf

Der Parameter COM_RST der Anweisung S_RCV dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

Zeitablaufdiagramm

Das folgende Bild zeigt das Verhalten der Parameter NDR, LEN und ERROR je nachdem, wie die Eingänge EN_R und R verdrahtet sind.

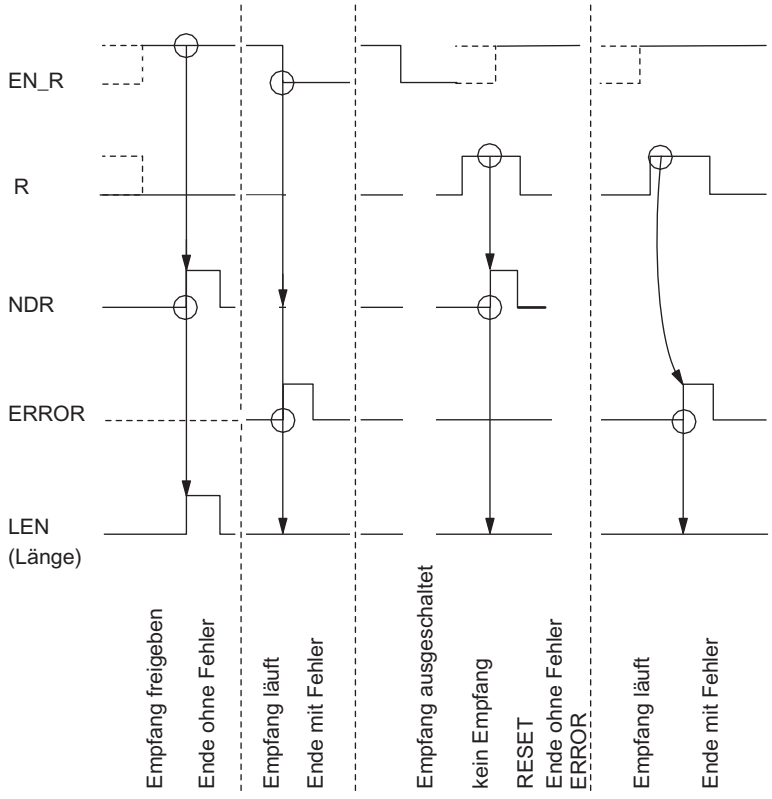

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Regeln

Hinweis

Die Anweisung S_RCV hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN von der Baugruppe ET 200S 1SI empfangen werden kann, muss der CPU-Anlaufmechanismus der Baugruppe ET 200S für die Anweisung S_RCV abgeschlossen sein.

S_SEND: Daten senden

Beschreibung

Die Anweisung S_SEND überträgt einen Datenblock von einem Datenbaustein, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN, zur Baugruppe ET 200S 1SI. Die Anweisung S_SEND wird zur Datenübertragung im Zyklus statisch (ohne Bedingungen) oder **alternativ** in einem zeitgesteuerten Programm aufgerufen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung S_SEND kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zur Baugruppe ET 200S 1SI abgebrochen und die Anweisung S_SEND in den Grundzustand versetzt. Daten, die die Baugruppe ET 200S 1SI bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. In STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung S_SEND ausgegeben (siehe Zeitablaufdiagramm). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer (Sendedaten) **S7-1500**: 0 ≤ DBB_NO und DBB_NO + LEN ≤ Größe des DB **S7-300/400**: 0 ≤ DBB_NO und DBB_NO + LEN < 8190

Parameter	Deklaration	Datentyp	Beschreibung
LEN	INPUT	INT	Datenlänge 1 ≤ LEN ≤ 224 Angabe in Anzahl Byte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter DONE, ERROR und STATUS stehen nach korrektem Sendeauftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_SEND arbeitet mit einem Instanz-DB I_SEND zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Anlauf

Der Parameter COM_RST der Anweisung S_SEND dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

Zeitablaufdiagramm

Das folgende Bild zeigt das Verhalten der Parameter DONE und ERROR je nachdem, wie die Eingänge REQ und R verdrahtet sind.

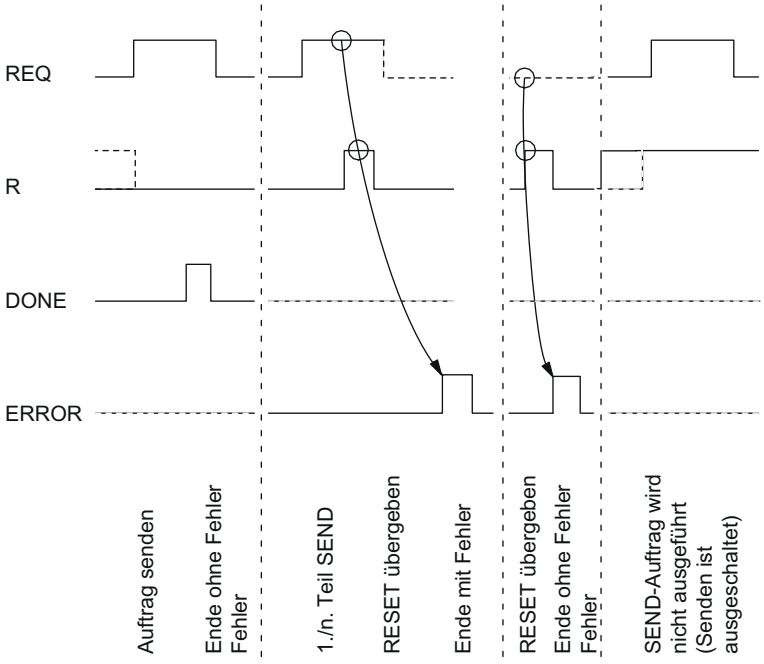

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regeln

Hinweis

Die Anweisung S_SEND hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN von der Baugruppe ET 200S 1SI bearbeitet werden kann, muss der CPU-Anlaufmechanismus der Baugruppe ET 200S für die Anweisung S_SEND abgeschlossen sein (siehe oben). Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zur Baugruppe ET 200S 1SI übertragen.

S_VSTAT: Begleitsignale an der RS 232C-Schnittstelle lesen

Beschreibung

Die Anweisung S_VSTAT liest die RS-232C-Begleitsignale der Baugruppe ET 200S 1SI und stellt sie dem Anwender in den Bausteinparametern zur Verfügung. Die Anweisung S_VSTAT wird statisch (ohne Bedingungen) für die Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm aufgerufen.

Arbeitsweise

Die RS-232C-Begleitsignale werden mit jedem Aufruf der Anweisung aktualisiert (zyklisches Pollen).

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
DONE [1]	OUTPUT	BOOL	Zeigt an, dass die Anweisung beendet ist (Ausgang ET 200S 1SI)
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
DTR_OUT [1]	OUTPUT	BOOL	Data terminal ready, ET 200S 1SI ist betriebsbereit. (Ausgang ET 200S 1SI)
DSR_IN [1]	OUTPUT	BOOL	Data set ready, Kommunikationspartner ist betriebsbereit. (Eingang ET 200S 1SI)
RTS_OUT [1]	OUTPUT	BOOL	Request to send, ET 200S 1SI ist sendebereit. (Ausgang ET 200S 1SI)

Parameter	Deklaration	Datentyp	Beschreibung
CTS_IN [1]	OUTPUT	BOOL	Clear to send, Kommunikationspartner kann Daten von der Baugruppe ET 200S 1SI empfangen (Antwort auf RTS = ON der ET 200S 1SI) (Eingang ET 200S 1SI)
DCD_IN [1]	OUTPUT	BOOL	Data carrier detect, Empfangssignalpegel (Eingang ET 200S 1SI)
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Diese Parameter stehen nach korrektem Auftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_VSTAT arbeitet mit einem Instanz-DB I_STAT zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Zum Erkennen eines Signalwechsels ist eine Mindestimpulsdauer notwendig. Ausschlaggebende Größen sind die CPU-Zykluszeit, die Aktualisierungszeit auf der Baugruppe ET 200S 1SI und die Reaktionszeit des Kommunikationspartners.

Anlauf

Der Parameter COM_RST der Anweisung S_VSTAT dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

S_VSET: Begleitsignale an der RS 232C-Schnittstelle schreiben

Beschreibung

Sie können die Schnittstellenausgänge über die entsprechenden Parametereingänge der Anweisung S_VSET setzen und rücksetzen. Die Anweisung S_VSET wird im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
RTS	INPUT	BOOL	Request to send, ET 200S 1SI ist sendebereit. (Ausgang ET 200S 1SI steuern)
DTR	INPUT	BOOL	Data terminal ready, ET 200S 1SI ist betriebsbereit. (Ausgang ET 200S 1SI steuern)
DONE [1]	OUTPUT	BOOL	(Ausgang ET 200S 1SI)
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Diese Parameter stehen nach korrektem Auftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_VSET arbeitet mit einem Instanz-DB I_SET zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Anlauf

Der Parameter COM_RST der Anweisung S_VSET dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

S_XON: Datenflusskontrolle über XON/XOFF parametrieren

Beschreibung

Mit der Anweisung S_XON können Sie zusätzliche Parameter einstellen, wenn die Baugruppe für die Datenflusskontrolle XON/XOFF parametriert wurde.

Hinweis

Die Anweisung S_XON muss vor den Anweisungen S_SEND, S_RCV, S_VSET und S_VSTAT aufgerufen werden und ihren Auftrag beendet haben (DONE = TRUE).

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
XON	INPUT	BYTE	XON-Zeichen 0 bis $7F_H$ (7 Datenbits) 0 bis FF_H (8 Datenbits) Voreinstellung: 11 (DC1)
XOFF	INPUT	BYTE	XOFF-Zeichen 0 bis $7F_H$ (7 Datenbits) 0 bis FF_H (8 Datenbits) Voreinstellung: 13 (DC3)
WAIT_FOR_XON	INPUT	TIME	Wartezeit für XON nach XOFF 20 ms bis 10 min 55 s 350 ms Voreinstellung: 2 s
DONE[1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00

Parameter	Deklaration	Datentyp	Beschreibung
ERROR[1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS[1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter DONE, ERROR und STATUS stehen nach korrektem Auftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_XON arbeitet mit einem Instanz-DB I_XON zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen.

Anlauf

Der Parameter COM_RST der Anweisung S_XON dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

S_RTS: Datenflusskontrolle über RTS/CTS parametrieren

Beschreibung

Mit der Anweisung S_RTS können Sie zusätzliche Parameter einstellen, wenn die Baugruppe für die Datenflusskontrolle RTS/CTS parametriert wurde.

Hinweis

Die Anweisung S_RTS muss vor den Anweisungen S_SEND, S_RCV, S_VSET und S_VSTAT aufgerufen werden und ihren Auftrag beendet haben (DONE = TRUE).

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S SI Die Basisadresse wird aus STEP 7 entnommen.
WAIT_FOR_CTS	INPUT	TIME	Wartezeit für CTS = ON 20 ms bis 10 min 55 s 350 ms Voreinstellung: 2 s
DONE[1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00
ERROR[1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS[1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung

[1] Die Parameter DONE, ERROR und STATUS stehen nach korrektem Auftrag **einen** CPU-Zyklus lang zur Verfügung!

Belegung im Datenbereich

Die Anweisung S_RTS arbeitet mit einem Instanz-DB I_RTS zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen.

Anlauf

Der Parameter COM_RST der Anweisung S_RTS dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

S_V24: Datenflusskontrolle über automatische Bedienung der RS 232C-Begleitsignale parametrieren

Beschreibung

Mit der Anweisung S_V24 können Sie zusätzliche Parameter einstellen, wenn die Baugruppe für automatische Bedienung der RS-232C-Begleitsignale parametriert ist.

Hinweis

Die Anweisung S_V24 muss vor den Anweisungen S_SEND, S_RCV, S_VSET und S_VSTAT aufgerufen werden und ihren Auftrag beendet haben (DONE = TRUE).

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
TIME_RTS_OFF	INPUT	TIME	Zeit, die nach der Übertragung ablaufen muss, bevor RTS ausgeschaltet wird. 0 ms bis 10 min 55 s 350 ms Voreinstellung: 10 ms
DATA_WAIT_TIME	INPUT	TIME	Zeit, die darauf gewartet wird, dass der Partner CTS = ON setzt, nachdem RTS gesetzt wurde. 0 ms bis 10 min 55 s 350 ms Voreinstellung: 10 ms

Parameter	Deklaration	Datentyp	Beschreibung
DONE[1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00
ERROR[1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6735) enthält die Fehlerinformation.
STATUS[1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6735) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter DONE, ERROR und STATUS stehen nach korrektem Auftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_V24 arbeitet mit einem Instanz-DB I_V24 zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen.

Anlauf

Der Parameter COM_RST der Anweisung S_V24 dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

Parameter STATUS

Ereignisklassen

Ereignisklasse	Bedeutung
2	Fehler bei der Initialisierung
5	Fehler bei Bearbeitung eines CPU-Auftrags
7	Sendefehler
8	Empfangsfehler
11 ($0B_H$)	Warnung
30 ($1E_H$)	Fehler bei Kommunikation zwischen Baugruppe und CPU

Parameter STATUS

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0201	Keine (gültige) Parametrierung vorhanden.	Versorgen Sie die Baugruppe mit korrekten Parametern. Überprüfen Sie ggf. die ordnungsgemäße Installation der Anlage.
0502	Auftrag ist in diesem Betriebszustand der Baugruppe ET 200S 1SI nicht erlaubt (z. B. Geräteschnittstelle nicht parametriert).	Das Sendetelegramm ist länger als 224 Byte. Der Sendeauftrag wurde von der Baugruppe ET 200S 1SI abgebrochen. Wählen Sie eine kleinere Telgrammlänge.
050E	Ungültige Telegrammlänge	Das Sendetelegramm ist länger als 224 Byte. Der Sendeauftrag wurde von der Baugruppe ET 200S 1SI abgebrochen. Wählen Sie eine kleinere Telegrammlänge.
0550	Parameteraktualisierungsauftrag ungültig für aktuelle Datenflusskontrolle der Baugruppe ET 200S 1SI.	Entweder ändern Sie die Parameter der Anweisungen (S_XON, S_RTS, S_V24) im AS-Programm oder Sie ändern die Datenflusskontrolle der Baugruppe ET 200S 1SI in der Hardware-Konfiguration, so dass sich beide entsprechen.
0551	Rahmenablauffehler bei der Kommunikation zwischen der Baugruppe ET 200S 1SI und dem Automatisierungssystem. Der Fehler ist beim Übertragen eines empfangenen Telegramms der Baugruppe ET 200S 1SI im Automatisierungssystem aufgetreten.	Die Baugruppe und das Automatisierungssystem haben die Übertragung abgebrochen. Wiederholen Sie den Empfangsauftrag; die Baugruppe ET 200S 1SI sendet die empfangene Meldung erneut.
0702	Nur bei 3964(R): Fehler beim Verbindungsaufbau: Nachdem STX gesendet wurde, wurde NAK oder ein beliebiges Zeichen (außer DLE oder STX) empfangen.	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0703	Nur bei 3964(R): Quittungsverzugszeit (QVZ) überschritten: Nach Senden von STX kam keine Antwort vom Partner innerhalb der Quittungsverzugszeit.	Partnergerät ist zu langsam oder nicht empfangsbereit, oder es liegt z. B ein Bruch der Sendeleitung vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0704	Nur bei 3964(R): Abbruch durch Partner: Während des laufenden Sendebetriebs wurden vom Partner ein oder mehrere Zeichen empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z. B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0705	Nur bei 3964(R): Negative Quittung beim Senden	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z.B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0706	Nur bei 3964(R): Fehler bei Verbindungsende: • Das Telegramm wurde vom Partner am Ende mit NAK oder einem beliebigen Zeichen (außer DLE) abgelehnt oder • das Quittungszeichen (DLE) wurde zu früh empfangen.	Prüfen Sie, ob der Partner ebenfalls Fehler anzeigt, da evtl. nicht alle Sendedaten angekommen sind (z.B. Bruch in der Sendeleitung) oder schwere Störungen vorliegen, oder es liegt ein Fehlverhalten des Partnergeräts vor. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0707	Nur bei 3964(R): Quittungsverzugszeit am Verbindungsende/ Antwortüberwachungszeit nach Sendetelegramm überschritten: Nach Verbindungsabbau mit DLE ETX kam innerhalb der QVZ keine Antwort vom Partner.	Partnergerät ist zu langsam oder gestört. Dies ggf. mit Schnittstellentestgerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
0708	Nur bei ASCII-Treiber: Die Wartezeit auf XON bzw. CTS = ON ist abgelaufen.	Der Kommunikationspartner ist gestört, zu langsam oder offline geschaltet. Überprüfen Sie den Kommunikationspartner oder ändern Sie ggf. die Parametrierung.
070B	Nur bei 3964(R): Initialisierungskonflikt ist nicht lösbar, weil beide Partner hochprior eingestellt sind.	Ändern Sie die Parametrierung.
070C	Nur bei 3964(R): Initialisierungskonflikt ist nicht lösbar, weil beide Partner niederprior eingestellt sind.	Ändern Sie die Parametrierung.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0802	Nur bei 3964(R): Fehler beim Verbindungsaufbau: • In Ruhestellung wurden ein oder mehrere beliebige Zeichen (außer NAK oder STX) empfangen oder • Nach einem empfangenen STX wurden vom Partner weitere Zeichen gesendet, ohne die Antwort DLE abzuwarten. Nach Netz-EIN des Partners: • Während der Partner eingeschaltet wird, empfängt die Baugruppe ein undefiniertes Zeichen.	Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0805	Nur bei 3964(R): Logischer Fehler während des Empfangs: Nach Empfang von DLE wurde ein weiteres beliebiges Zeichen empfangen (außer DLE, ETX).	Prüfen Sie, ob der Partner DLE im Telegrammkopf und im Datenstring immer verdoppelt bzw. der Verbindungsabbau mit DLE ETX vorgenommen wird. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0806	Zeichenverzugszeit (ZVZ) überschritten: • Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der ZVZ empfangen oder Nur bei 3964(R): • 1. Zeichen nach Senden von DLE beim Verbindungsaufbau wurde nicht innerhalb der ZVZ empfangen.	Partnergerät ist zu langsam oder gestört. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0807	Nur bei 3964(R): Telegrammlänge unzulässig: Es wurde ein Telegramm mit der Länge 0 empfangen.	Der Empfang eines Telegramms mit Länge 0 ist kein Fehler. Überprüfen Sie, warum der Kommunikationspartner Telegramme ohne Nutzdaten sendet.
0808	Nur bei 3964(R): Fehler beim Blockprüfzeichen BCC: Der intern gebildete Wert des BCC stimmt nicht mit dem vom Partner am Verbindungsende empfangenen BCC überein.	Prüfen Sie, ob die Verbindung stark gestört ist, in diesem Fall werden auch gelegentlich Fehlercodes zu beobachten sein. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
0809	Nur bei 3964(R): Die Wiederholanzahl muss gleich eingestellt sein.	Parametrieren Sie beim Kommunikationspartner die gleiche Blockwartezeit wie an der Baugruppe. Fehlverhalten des Kommunikationspartners ggf. mit Schnittstellengerät nachweisen, das in die Übertragungsleitung eingeschaltet wird.
080A	Ein freier Empfangspuffer ist nicht vorhanden: Beim Empfang stand kein leerer Empfangspuffer zur Verfügung.	Die Anweisung S_RCV muss häufiger aufgerufen werden.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
080C	Übertragungsfehler: • Ein Übertragungsfehler (Paritätsfehler, Stoppbitfehler, Überlauffehler) wurde erkannt. Nur bei 3964(R): • Falls dies während des Sende- oder Empfangsbetriebs auftritt, werden Wiederholungen gestartet. • Wird in Ruhestellung ein gestörtes Zeichen empfangen, wird der Fehler sofort gemeldet, damit Störeinflüsse in der Übertragungsleitung frühzeitig erkannt werden können. • Ist die SF-LED (rot) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung der beiden Kommunikationspartner vor.	Störungen in der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind.
080D	BREAK: Empfangsleitung zum Partner ist unterbrochen.	Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
0810	Nur bei ASCII-Treiber: Paritätsfehler: • Ist die SF-LED (rot) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0811	Nur bei ASCII-Treiber: Zeichenrahmenfehler: • Ist die SF-LED (rot) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0812	Nur bei ASCII-Treiber: Nachdem die Baugruppe XOFF gesendet hat oder CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder lesen Sie die Daten der Baugruppe schneller.
0818	Nur bei ASCII-Treiber: DSR = OFF bzw. CTS = OFF	Vor oder während eines Sendevorgangs sind die Signale DSR bzw. CTS vom Partner auf "OFF" geschaltet worden. Überprüfen Sie die Steuerung der RS-232C-Begleitsignale beim Partner.
0850	Länge des Empfangstelegramms größer als 224 Byte bzw. größer als die parametrierte Telegrammlänge	Telegrammlänge des Partners anpassen
0B01	Empfangspuffer ist mehr als 2/3 voll	
1E0D	"Auftragsabbruch wegen Neustart, Wiederanlauf oder Reset"	

Fehlercode (W#16#...)	Beschreibung	Abhilfe
1E0E	Statischer Fehler bei Aufruf der SFC DPRD_DAT. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E0F	Statischer Fehler bei Aufruf der SFC DPWR_DAT. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E10	Statischer Fehler bei Aufruf der SFC RD_LGADR. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E11	Statischer Fehler bei Aufruf der SFC RDSYSST. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E20	Parameter außerhalb des Bereichs.	Ändern Sie den Eingang der Anweisung, so dass er im gültigen Bereich liegt.
1E41	Anzahl der am Parameter LEN der Anweisungen angegebenen Bytes unzulässig	Halten Sie den Wertebereich von 1 bis 224 Byte ein.

Auswerten der Variable SFCERR

Nähere Informationen zu den aufgetretenen Fehlern (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H und (1E) 11_H, der Ereignisklasse 30 erhalten Sie über die Variable SFCERR.

Die Variable SFCERR können Sie aus dem Instanz-DB der entsprechenden Anweisung laden.

Die Fehlermeldungen, die in der Variablen SFCERR eingetragen werden, finden Sie bei den Systemfunktionen "DPRD_DAT" und SFC15 "DPWR_DAT" im Referenzhandbuch *Systemsoftware für S7 300/400, System- und Standardfunktionen*.

ET 200S 1SI für Modbus

Senden und Empfangen als Modbus Master

Beschreibung

Die Datenübergabe zwischen Baugruppe und CPU erfolgt mittels der Anweisungen S_SEND und S_RCV. Die Anweisung S_SEND wird mit einer Flanke am Eingang REQ aktiviert, wenn Daten ausgegeben werden sollen. Die Anweisung S_RCV wird mit EN_R=1 empfangsbereit gemacht. Ein S_RCV ist bei allen lesenden Funktionscodes notwendig.

Senden und Empfangen mit dem Modbus-Master-Protokoll

Für die Ausführung eines Modbus-Master-Auftrags müssen die Anweisungen S_SEND und S_RCV aktiviert werden. Die Anweisung S_SEND wird mit einer Flanke am Eingang REQ aktiviert, wenn Daten an die Baugruppe ausgegeben werden sollen. Die Anweisung S_RCV wird mit EN_R=1 für den Empfang von Daten von der Baugruppe vorbereitet. Ein S_RCV ist bei allen lesenden Funktionscodes notwendig. Das folgende Bild zeigt das Gesamtverhalten der Parameter von S_SEND und S_RCV, wenn ein Modbus-Auftrag ausgeführt wird.

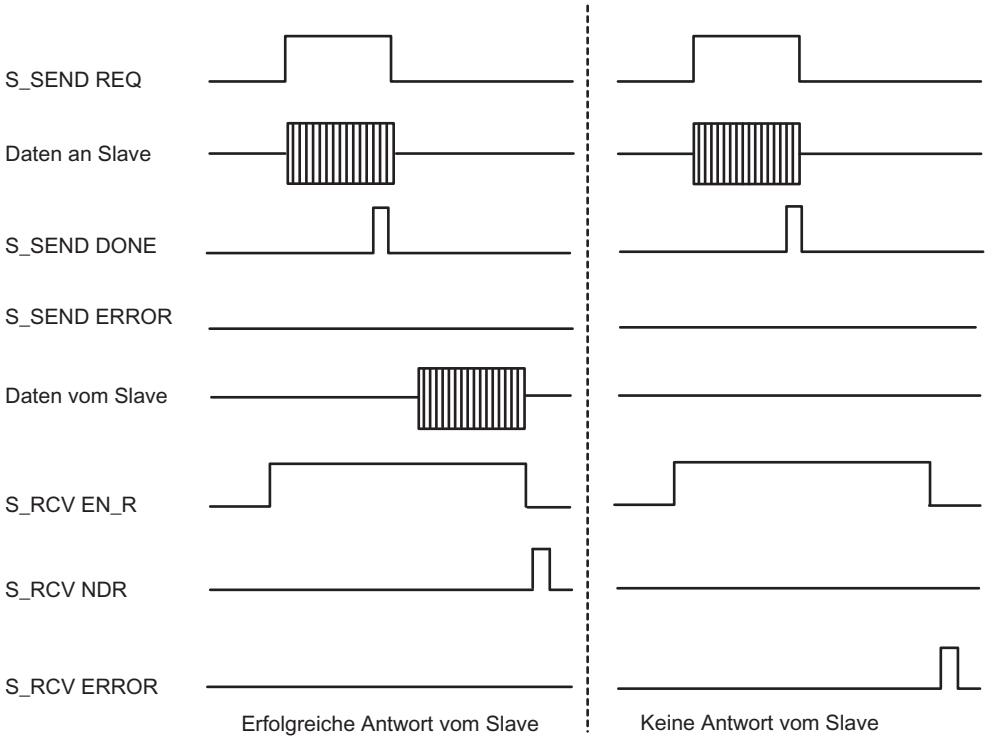

Hinweis

Abfolge von Senden und Empfangen

Da die Schnittstelle zwischen dem Anwenderprogramm und der Schnittstellenbaugruppe halbduplex arbeitet, müssen Sie folgendes beachten:

Nach einem positiv quittierten Modbus-Master-Leseauftrag müssen Sie zuerst mit der Anweisung S_RCV die Empfangsdaten von der Schnittstellenbaugruppe abholen, bevor Sie einen neuen Modbus-Master-Sendeauftrag starten.

S_RCV: Daten empfangen

Beschreibung

Die Anweisung S_RCV überträgt Daten von der Baugruppe ET 200S 1SI zu einem S7-Datenbereich, spezifiziert durch die Parameter DB_NO und DBB_NO. Die Anweisung S_RCV wird zur Datenübertragung im Zyklus oder alternativ in einem zeitgesteuerten Programm statisch (ohne Bedingungen) aufgerufen.

Arbeitsweise

Mit (statisch) Signalzustand "1" am Parameter EN_R wird die Überprüfung, ob Daten von der Baugruppe ET 200S 1SI zu lesen sind, freigegeben. Eine laufende Übertragung kann mit Signalzustand "0" am Parameter EN_R und zusätzlich mit Signalzustand "1" am Parameter R abgebrochen werden. (Da es sporadisch vorkommen kann, dass nach Setzen des Parameters EN_R auf "1" die Übertagung nicht mehr anläuft, sollten Sie zum Abbrechen einer laufenden Übertragung parallel zum Setzen des EN_R auf "0" den Parameter R auf "1" setzen.) Der abgebrochene Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange Signalzustand "0" am Parameter EN_R ansteht. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Erkennt die Anweisung Signalzustand "1" am Parameter R, dann wird der momentane Übertragungsauftrag abgebrochen und die Anweisung S_RCV in den Grundzustand versetzt. Der Empfang ist ausgeschaltet, solange Signalzustand "1" am Parameter R ansteht. Bei erneutem Signalzustand "0" wird das abgebrochene Telegramm erneut, von Anfang an, empfangen.

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Der Ausgang NDR zeigt "Auftrag fertig ohne Fehler/Daten übernommen" an (alle Daten gelesen). ERROR zeigt einen aufgetretenen Fehler an. In STATUS wird bei einem Fehler die entsprechende Fehlernummer angezeigt. Ist der Empfangspuffer mehr als 2/3 voll, enthält STATUS nach jedem Aufruf von S_RCV eine Warnung. Liegen keine Fehler oder Warnungen vor, hat STATUS den Wert "0".

NDR und ERROR/STATUS werden auch dann ausgegeben, wenn die Anweisung S_RCV zurückgesetzt wird (Parameter LEN == 16#00) (siehe Zeitablaufdiagramm). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
EN_R	INPUT	BOOL	Freigabe für Daten lesen
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Empfang gesperrt.

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Empfangs-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer (Empfangsdaten) **S7-1500**: 0 ≤ DBB_NO und DBB_NO + LEN ≤ Größe des DB **S7-300/400**: 0 ≤ DBB_NO und DBB_NO + LEN < 8190
NDR [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler, Daten übernommen Parameter STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6749) enthält die Fehlerinformation.
LEN [1]	OUTPUT	INT	Länge des empfangenen Telegramms 1 ≤ LEN ≤ 224 Angabe in Anzahl Byte
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6749) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter NDR; ERROR, LEN und STATUS stehen nach korrektem Empfangsauftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_RCV arbeitet mit einem Instanz-DB I_RCV zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR entnehmen. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Anlauf

Der Parameter COM_RST der Anweisung S_RCV dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

Zeitablaufdiagramm

Das folgende Bild zeigt das Verhalten der Parameter NDR, LEN und ERROR je nachdem, wie die Eingänge EN_R und R verdrahtet sind.

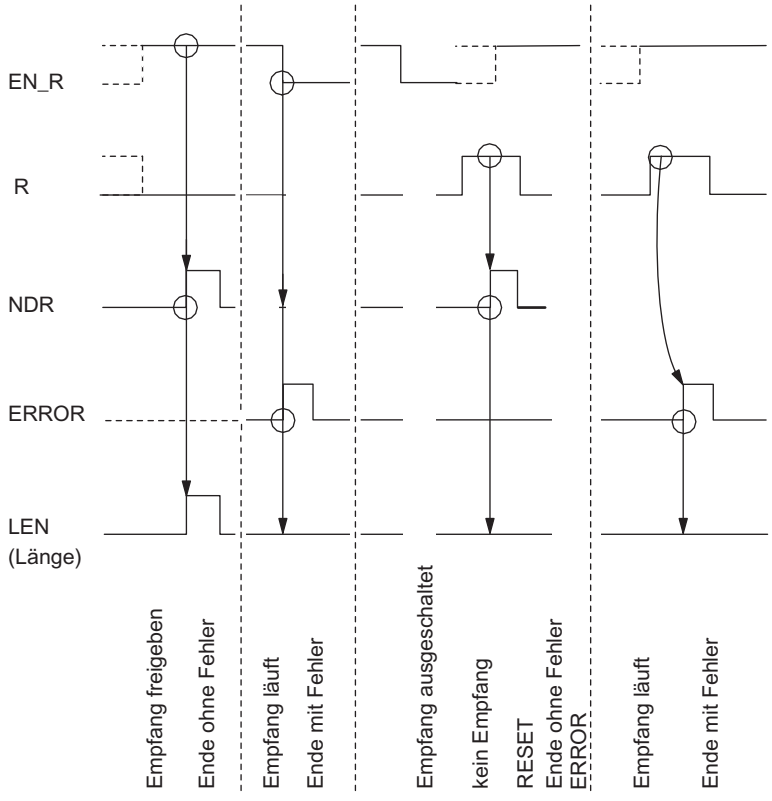

Hinweis

Der Eingang EN_R ist statisch auf "1" zu legen. Während des gesamten Empfangsauftrages muss der Parameter EN_R mit dem VKE "1" (Verknüpfungsergebnis) versorgt werden.

Regeln

Hinweis

Die Anweisung S_RCV hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN von der Baugruppe ET 200S 1SI empfangen werden kann, muss der CPU-Anlaufmechanismus der Baugruppe ET 200S für die Anweisung S_RCV abgeschlossen sein.

S_SEND: Daten senden

Beschreibung

Die Anweisung S_SEND überträgt einen Datenblock von einem Datenbaustein, spezifiziert durch die Parameter DB_NO, DBB_NO und LEN, zur Baugruppe ET 200S 1SI. Die Anweisung S_SEND wird zur Datenübertragung im Zyklus statisch (ohne Bedingungen) oder **alternativ** in einem zeitgesteuerten Programm aufgerufen.

Arbeitsweise

Mit einer positiven Flanke am Eingang REQ wird die Übertragung der Daten angestoßen. Je nach Datenmenge kann eine Datenübertragung über mehrere Aufrufe (Programmzyklen) laufen.

Die Anweisung S_SEND kann mit Signalzustand "1" am Parametereingang R im Zyklus aufgerufen werden. Es wird damit die Übertragung zur Baugruppe ET 200S 1SI abgebrochen und die Anweisung S_SEND in den Grundzustand versetzt. Daten, die die Baugruppe ET 200S 1SI bereits erhalten hat, werden noch an den Kommunikationspartner gesendet. Steht statisch Signalzustand "1" am Eingang R an, so ist das Senden ausgeschaltet.

An dem Parameter LADDR wird die Adresse der anzusprechenden Baugruppe ET 200S 1SI angegeben.

Der Ausgang DONE zeigt "Auftragsende ohne Fehler" an. ERROR zeigt einen aufgetretenen Fehler an. In STATUS wird bei einem Fehler die entsprechende Ereignisnummer angezeigt. Ist kein Fehler aufgetreten, hat STATUS den Wert 0. DONE und ERROR/STATUS werden auch bei RESET der Anweisung S_SEND ausgegeben (siehe Zeitablaufdiagramm). Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand "1".

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
REQ	INPUT	BOOL	Auftragsanstoß bei positiver Flanke
R	INPUT	BOOL	Auftragsabbruch Laufender Auftrag wird abgebrochen. Senden gesperrt.
LADDR	INPUT	INT	Basisadresse der Baugruppe ET 200S 1SI Die Basisadresse wird aus STEP 7 entnommen.
DB_NO	INPUT	INT	Datenbausteinnummer Sende-DB-Nr.: CPU-spezifisch, Null ist nicht erlaubt
DBB_NO	INPUT	INT	Datenbytenummer (Sendedaten) **S7-1500**: 0 ≤ DBB_NO und DBB_NO + LEN ≤ Größe des DB **S7-300/400**: 0 ≤ DBB_NO und DBB_NO + LEN < 8190
LEN	INPUT	INT	Datenlänge 1 ≤ LEN ≤ 224 Angabe in Anzahl Byte
DONE [1]	OUTPUT	BOOL	Auftrag fertig ohne Fehler Parameter STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Auftrag abgebrochen mit Fehler Parameter STATUS (Seite 6749) enthält die Fehlerinformation.
STATUS [1]	OUTPUT	WORD	Spezifikation des Fehlers Bei ERROR == 1 steht im Parameter STATUS (Seite 6749) die Fehlerinformation.
COM_RST	IN_OUT	BOOL	Neustart der Anweisung
[1] Die Parameter DONE, ERROR und STATUS stehen nach korrektem Sendeauftrag **einen** CPU-Zyklus lang zur Verfügung!			

Belegung im Datenbereich

Die Anweisung S_SEND arbeitet mit einem Instanz-DB I_SEND zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Hinweis

Ausnahme: Im Fehlerfall, STATUS == W#16#1Exx, können Sie einen genaueren Fehlerhinweis aus der Variablen SFCERR. Diese Fehlervariable kann nur über einen symbolischen Zugriff auf den Instanz-DB geladen werden.

Anlauf

Der Parameter COM_RST der Anweisung S_SEND dient dazu, der Anweisung einen Anlauf mitzuteilen.

Setzen Sie den Parameter COM_RST im Anlauf-OB auf 1.

Rufen Sie die Anweisung im zyklischen Betrieb auf, ohne den Parameter COM_RST zu setzen oder rückzusetzen.

Wenn der Parameter COM_RST gesetzt ist,

- ermittelt die Anweisung Informationen über das Modul ET 200S 1SI (Anzahl Byte im Peripheriebereich, in dezentraler Peripherie oder nicht).
- setzt sich die Anweisung zurück und beendet dabei evtl. einen zuvor (vor dem letzten Stopp-Übergang der CPU) begonnenen Auftrag.

Wenn die Anweisung die Information über das Modul ET 200S 1SI ermittelt hat, setzt sie den Parameter COM_RST selbst zurück.

Zeitablaufdiagramm

Das folgende Bild zeigt das Verhalten der Parameter DONE und ERROR je nachdem, wie die Eingänge REQ und R verdrahtet sind.

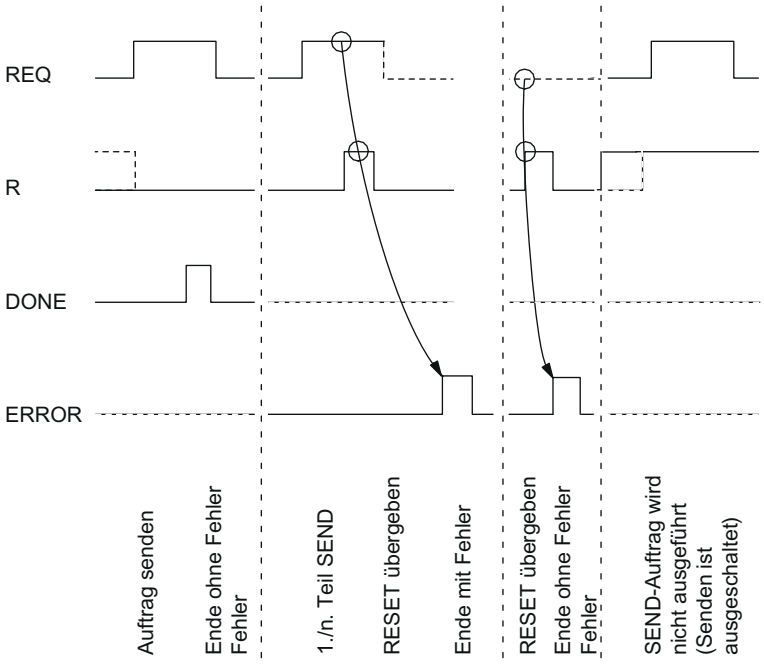

Hinweis

Der Eingang REQ ist flankengetriggert. Es genügt am Eingang REQ eine positive Flanke. Es muss nicht während der gesamten Übertragung das VKE (Verknüpfungsergebnis) auf "1" sein.

Regeln

Hinweis

Die Anweisung S_SEND hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Hinweis

Bevor ein angestoßener Auftrag nach einem Zustandsübergang der CPU von STOP nach RUN von der Baugruppe ET 200S 1SI bearbeitet werden kann, muss der CPU-Anlaufmechanismus der Baugruppe ET 200S für die Anweisung S_SEND abgeschlossen sein (siehe oben). Ein in der Zwischenzeit angestoßener Auftrag geht nicht verloren. Er wird nach Abschluss der Anlaufkoordinierung zur Baugruppe ET 200S 1SI übertragen.

S_MODB: Modbus-Slave-Anweisung für ET 200S 1SI

Beschreibung

Für die Ausführung eines Modbus-Slave-Auftrags muss die Anweisung S_MODB zyklisch im Anwenderprogramm aktiviert werden. Die Anweisung S_MODB empfängt den Auftrag von der seriellen Schnittstellenbaugruppe ET200S Modbus/USS, führt den Auftrag aus und gibt die Antwort an die Baugruppe zurück. Die Kommunikation zwischen der CPU und der Baugruppe wird mittels der Anweisungen S_SEND und S_RCV durchgeführt, die von S_MODB aufgerufen werden.

Arbeitsweise

Nach jedem Wiederanlauf der CPU muss das Anwenderprogramm eine Initialisierung der Modbus-Kommunikations-Anweisung durchführen. Die Initialisierung wird durch eine steigende Flanke am Eingang CP_START aktiviert. Die Anweisung zeichnet die Größen der Operandenbereiche E, A, M, T und Z der CPU im Instanz-Datenbaustein der Anweisung auf. Nach erfolgreichem Abschluss der Initialisierung setzt die Anweisung den Ausgang CP_START_OK.

Ein Initialisierungsfehler wird vom Ausgang CP_START_ERROR angegeben. In diesem Fall ist keine Modbus-Kommunikation möglich und alle Aufträge vom Modbus-Master werden mit einer Ausnahmecode-Meldung beantwortet.

S_MODB nutzt eine Modbus-Datenumsetzungstabelle, die sich im Datenbaustein befindet, um die Modbus-Adressen auf die SIMATIC S7-Speicherbereiche abzubilden.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
LADDR	INPUT	INT	Basisadresse der seriellen Schnittstellenbaugruppe ET 200S 1SI. Die Basisadresse wird aus STEP 7 entnommen.
START_TIMER	INPUT	TIMER	Zeit für Initialisierungs- Timeout: CPU-Zeit, die die Anweisung einsetzt, um zu verhindern, dass sich die Initialisierung aufhängt.
START_TIME	INPUT	S5TIME	Zeitwert für Initialisierungs-Timeout: Maximale Zeit, die Initialisierung der Anweisung dauern darf
DB_No	INPUT	BLOCK_DB	Nummer des Datenbausteins, der die Modbus-Umsetzungstabelle enthält: Siehe Datenbereiche in der SIMATIC CPU
OB_MASK	INPUT	BOOL	Maskiert E/A-Zugriffsfehler und verzögert Alarme: TRUE ermöglicht, dass Zugriffsfehler auf nicht vorhandene E/A nur dem Modbus-Master gemeldet werden
CP_START	INPUT	BOOL	Start Initialisierung der Anweisung
CP_START_FM	INPUT	BOOL	Die Initialisierung wird durch eine steigende Flanke am Eingang START aktiviert: Speicherelement, in dem die Anweisung den vorherigen Wert von CP_START speichert
CP_NDR	OUTPUT	BOOL	Auftragstelegramm empfangen
CP_START_OK	OUTPUT	BOOL	Initialisierung fehlerfrei beendet: Die Initialisierung wurde fehlerfrei beendet, bevor die Überwachungszeit abgelaufen ist.
CP_START_ERROR	OUTPUT	BOOL	Initialisierung mit Fehler beendet: Die Initialisierung konnte nicht fehlerfrei beendet werden, auch nicht, nachdem die Überwachungszeit abgelaufen war.
ERROR_NR	OUTPUT	WORD	Fehlernummer Siehe Diagnose
ERROR_INFO	OUTPUT	WORD	Zusätzliche Fehlerinformationen Siehe Diagnose

Belegung im Datenbereich

Die Anweisung S_MODB arbeitet zusammen mit einem Instanz-DB IDB_MODB, dessen Nummer im Aufruf angegeben wird.

Es ist nicht möglich, auf die Daten im Instanz-DB zuzugreifen.

Verhalten im Fehlerfall

Mit dem Eingangsparameter OB_MASK kann die Modbus-Anweisung angewiesen werden, E/A-Zugriffsfehler zu maskieren. Bei einem Schreibzugriff auf nicht vorhandene Peripherie geht die CPU nicht in den Betriebszustand STOP und ruft keinen Fehler-OB auf. Der Zugriffsfehler wird von der Anweisung erkannt und die Anweisung wird mit einer Fehlerantwort an den Modbus-Master beendet.

Parameter STATUS (Modbus / USS)

Ereignisklassen

Ereignisklasse	Bedeutung
2	Fehler bei der Initialisierung
5	Fehler bei Bearbeitung eines CPU-Auftrags
8	Empfangsfehler
14 ($0E_H$)	Allgemeine Verarbeitungsfehler: Parametrierung, Verarbeitung eines Auftrags S_SEND. Empfangsauswertung, Empfang Ausnahmecodemeldung
30 ($1E_H$)	Fehler bei Kommunikation zwischen Baugruppe und CPU

Parameter STATUS

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0201	Keine (gültige) Parametrierung vorhanden.	Versorgen Sie die Baugruppe mit korrekten Parametern. Überprüfen Sie ggf. die ordnungsgemäße Installation der Anlage.
0502	Auftrag unzulässig in diesem Betriebszustand der seriellen Schnittstellenbaugruppe ET 200S Modbus/USS (Beispiel: Geräteschnittstelle ist nicht parametriert).	Werten Sie den Diagnosealarm aus, und beheben Sie den Fehler entsprechend.
050E	Ungültige Telegrammlänge	Das Sendetelegramm ist länger als 224 Byte. Der Sendeauftrag wurde von der Baugruppe ET 200S Modbus/USS abgebrochen. Wählen Sie eine kleinere Telegrammlänge.
0530	Modbus-Master-Sendeauftrag abgelehnt, da die Antwort des Koppelpartners auf einen vorherigen lesenden Modbus-Master-Sendeauftrag noch nicht abgerufen wurde.	Nach einem erfolgreichen lesenden Modbus-Master-Sendeauftrag müssen Sie zuerst die Antwort des Koppelpartners aus dem Modul auslesen, bevor Sie einen neuen Modbus-Master-Sendeauftrag starten.
0551	Rahmenablauffehler bei der Kommunikation zwischen der seriellen Schnittstellenbaugruppe ET 200S Modbus/USS und der CPU. Der Fehler ist beim Übertragen eines empfangenen Telegramms der seriellen Schnittstellenbaugruppe ET 200S SI in der CPU aufgetreten.	Die Baugruppe und die CPU haben die Übertragung abgebrochen. Wiederholen Sie den Empfangsauftrag. Die serielle Schnittstellenbaugruppe ET 200S Modbus/USS sendet die empfangene Meldung erneut.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0806	Zeichenverzugszeit überschritten. Zwei aufeinanderfolgende Zeichen wurden nicht innerhalb der Zeichenverzugszeit empfangen.	Partnergerät ist zu langsam oder gestört. Fehlverhalten des Partnergeräts ggf. mit Schnittstellentestgerät (FOXPG) untersuchen, das in die Übertragungsleitung eingeschaltet wird.
080A	Überlauf des Empfangspuffers beim Master während des Empfangs des Antworttelegramms.	Prüfen Sie die Protokolleinstellungen des Slave.
080C	Ein Übertragungsfehler (Paritätsfehler, Stoppbitfehler, Überlauffehler) wurde erkannt.	Störungen auf der Übertragungsleitung verursachen Telegrammwiederholungen und erniedrigen dadurch den Nutzdatendurchsatz. Die Gefahr eines nicht erkannten Fehlers steigt. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung. Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind.
080D	BREAK: Empfangsleitung zum Partner ist unterbrochen.	Stellen Sie die Verbindung wieder her oder schalten Sie den Partner ein.
0810	Paritätsfehler: Ist die SF-LED (rot) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0811	Zeichenrahmenfehler: Ist die SF-LED (rot) eingeschaltet, liegt eine Unterbrechung der Verbindungsleitung (Leitungsbreak) der beiden Kommunikationspartner vor.	Überprüfen Sie die Verbindungsleitung der Kommunikationspartner bzw. überprüfen Sie, ob bei beiden Geräten Baudrate, Parität und Stoppbitanzahl gleich eingestellt sind. Ändern Sie Ihren Systemaufbau bzw. die Leitungsverlegung.
0812	Nachdem die serielle Schnittstelle CTS auf OFF gestellt hat, wurden weitere Zeichen empfangen.	Parametrieren Sie den Kommunikationspartner neu oder entsorgen Sie die serielle Schnittstelle schneller.
0830	Master: Es wurde ein Anforderungstelegramm gesendet und die Antwortüberwachungszeit ist abgelaufen, ohne dass der Beginn eines Antworttelegramms erkannt wurde. Slave: Broadcast bei diesem Funktionscode nicht zulässig.	Prüfen Sie, ob die Übertragungsleitung unterbrochen ist (evtl. Schnittstellenanalyse erforderlich). Prüfen Sie, dass in der Baugruppe und beim Kommunikationspartner die gleichen Einstellungen für die Protokollparameter Übertragungsgeschwindigkeit, Anzahl Datenbits, Parität und Anzahl Stoppbits vorhanden sind. Prüfen Sie, ob der Wert für die Antwortüberwachungszeit in PtP_PARAM groß genug ist. Prüfen Sie, ob die angegebene Slave-Adresse vorhanden ist. Das Modbus-Mastersystem darf Broadcast nur bei den dafür freigegebenen Funktionscodes verwenden.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0831	Master: Das erste Zeichen im Antworttelegramm des Slave unterscheidet sich von der Slave-Adresse, die im Anforderungstelegramm gesendet wurde (für den Normalbetrieb). Slave: Empfangener Funktionscode nicht zulässig.	Der falsche Slave hat geantwortet. Prüfen Sie, ob die Übertragungsleitung unterbrochen ist (evtl. Schnittstellenanalyse erforderlich). Dieser Funktionscode kann für diesen Treiber nicht verwendet werden.
0832	Maximale Anzahl Bits oder Register überschritten oder Anzahl Bits kann nicht durch 16 dividiert werden, wenn auf die SIMATIC-Speicherbereiche Zeiten oder Zähler zugegriffen wird.	Begrenzen Sie die maximale Anzahl Bits auf 2040 und die maximale Anzahl Register auf 127. Zugriff auf SIMATIC-Zeiten/-Zähler nur in 16-Bit-Abständen.
0833	Bit- oder Registeranzahl bei Funktionscode FC 15/16 und Telegrammelement byte_count passen nicht zusammen.	Korrigieren Sie die Bit-/Registeranzahl oder byte_count.
0834	Unzulässige Bit-Codierung für "Bit setzen"/Bit rücksetzen" erkannt.	Verwenden Sie für FC05 nur die Codierungen 0000Hex oder FF00Hex.
0835	Unzulässiger Diagnose-Subcode (ungleich 0000Hex) bei Funktionscode FC 08 "Loop Back Test" erkannt.	Verwenden Sie für FC08 nur den Subcode 0000Hex.
0836	Der intern gebildete Wert der CRC-16-Prüfsumme stimmt nicht mit der empfangenen CRC-Prüfsumme überein.	Überprüfen Sie die Bildung der CRC-Püfsumme beim Modbus-Mastersystem.
0837	Telegrammablauffehler: Das Modbus-Mastersystem sendete ein neues Anforderungstelegramm ehe vom Treiber das letzte Antworttelegramm übertragen wurde.	Erhöhen Sie das Timeout auf das Slave-Antworttelegramm beim Modbus-Mastersystem.
0850	Länge des Empfangtelegramms größer als 224 Byte bzw. größer als die parametrierte Telegrammlänge.	Telegrammlänge des Partners anpassen.
0E20	Die Anzahl der Datenbits bei dieser Kopplung muss 8 betragen. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.
0E21	Der parametrierte Multiplikationsfaktor für die Zeichenverzugszeit ist nicht im Bereich von 1 bis 10. Der Treiber arbeitet mit der Standardeinstellung 1.	Korrigieren Sie die Parametrierung des Treibers.
0E22	Die parametrierte Betriebsart des Treibers ist unzulässig. Es ist "Normalbetrieb" oder "Störunterdrückungsbetrieb" anzugeben. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E23	**Master**: Es wurde ein unzulässiger Wert für die Antwortüberwachungszeit eingestellt: gültige Werte liegen zwischen 50 und 655000 ms. Der Treiber ist nicht betriebsbereit. **Slave**: Es wurde ein unzulässiger Wert für die Slave-Adresse eingestellt. Die Slave-Adresse 0 ist unzulässig. Der Treiber ist nicht betriebsbereit.	Korrigieren Sie die Parametrierung des Treibers. Korrigieren Sie die Parametrierung des Treibers.
0E2E	Beim Lesen der Schnittstellenparameterdatei ist ein Fehler aufgetreten. Der Treiber ist nicht betriebsbereit.	Starten Sie den Master (Mains_ON) neu.
0E40	Der für LEN bei S_SEND angegebene Wert ist zu klein.	Mindestlänge sind 2 Byte.
0E41	Der für LEN bei S_SEND angegebene Wert ist zu klein. Eine größere Länge ist für den übertragenen Funktionscode erforderlich.	Die Mindestlänge für diesen Funktionscode beträgt 6 Byte.
0E42	Der übertragene Funktionscode ist unzulässig.	Verwenden Sie nur zulässige Funktionscodes.
0E43	Slave-Adresse 0 (= Broadcast) unzulässig bei diesem Funktionscode.	Verwenden Sie die Slave-Adresse 0 nur bei geeigneten Funktionscodes.
0E44	Der Wert der übertragenen "Bitanzahl" liegt nicht im Bereich 1 bis 2040.	Die "Bitanzahl" muss sich im Bereich 1 bis 2040 befinden.
0E45	Der Wert der übertragenen "Registeranzahl" liegt nicht im Bereich 1 bis 127.	Die "Registeranzahl" muss sich im Bereich 1 bis 127 befinden.
0E46	Funktionscode 15 oder 16: Die Werte der übertragenen "Bitanzahl" bzw. "Registeranzahl" liegen nicht im Bereich 1 bis 2040 bzw. 1 bis 127.	Die "Bitanzahl" bzw. "Registeranzahl" muss im Bereich 1 bis 2040 bzw. 1 bis 127 liegen.
0E47	Funktionscode 15 oder 16: LEN für S_SEND entspricht nicht der übertragenen "Bitanzahl" bzw. "Registeranzahl". LEN ist zu klein.	Erhöhen Sie LEN für SEND, bis eine ausreichende Menge Anwenderdaten an die Baugruppe übertragen wird. Eine größere Menge Benutzerdaten muss wegen der "Bitanzahl" bzw. "Registeranzahl" zur Baugruppe übertragen werden.
0E48	Funktionscode 5: Der in Quell-DB SEND angegebene Code für "Bit setzen" (FF00H) bzw. "Bit löschen" (0000H) ist falsch.	Die einzigen zulässigen Codes sind "Bit setzen" (FF00H) "Bit löschen" oder 0000H.
0E49	Funktionscode 8: Der in Quell-DB SEND angegebene Code für "Diagnosecode" ist falsch.	Der einzige zulässige Code ist "Diagnosecode" 0000H.
0E4A	Die Länge für diesen Funktionscode ist größer als die maximale Länge.	Im Handbuch finden Sie die maximalen Längenangaben zu jedem Funktionscode.
0E50	Master hat Antwort empfangen, ohne zu senden.	Ein Slave oder ein weiterer Master ist im Netz. Prüfen Sie, ob die Übertragungsleitung unterbrochen ist (evtl. Schnittstellenanalyse erforderlich).

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E51	Funktionscode falsch: Der im Antworttelegramm empfangene Funktionscode unterscheidet sich vom gesendeten Funktionscode.	Prüfen Sie das Slave-Gerät.
0E52	Byte-Unterlauf: Anzahl empfangener Zeichen ist kleiner als sich aus dem Byte-Zähler des Antworttelegramms ergeben würde bzw. ist kleiner als für diesen Funktionscode erwartet.	Prüfen Sie das Slave-Gerät.
0E53	Byte-Überlauf: Anzahl empfangener Zeichen ist größer als sich aus dem Byte-Zähler des Antworttelegramms ergeben würde bzw. ist größer als für diesen Funktionscode erwartet.	Prüfen Sie das Slave-Gerät.
0E54	Bytezähler falsch: Der im Antworttelegramm empfangene Bytezähler ist zu klein.	Prüfen Sie das Slave-Gerät.
0E55	Bytezähler falsch: Der im Antworttelegramm empfangene Bytezähler ist falsch.	Prüfen Sie das Slave-Gerät.
0E56	Echo falsch: Die vom Slave geechoten Daten des Antworttelegramms (Bitanzahl, ...) unterscheiden sich von den im Anforderungstelegramm gesendeten Daten.	Prüfen Sie das Slave-Gerät.
0E57	CRC-Prüfung falsch: Bei der Prüfung der CRC-16-Prüfsumme des Antworttelegramms vom Slave ist ein Fehler aufgetreten.	Prüfen Sie das Slave-Gerät.
0E61	Antworttelegramm mit Ausnahmecode 01: Unzulässige Funktion	Siehe Handbuch des Slave-Geräts.
0E62	Antworttelegramm mit Ausnahmecode 02: Unzulässige Datenadresse	Siehe Handbuch des Slave-Geräts.
0E63	Antworttelegramm mit Ausnahmecode 03: Unzulässiger Datenwert	Siehe Handbuch des Slave-Geräts.
0E64	Antworttelegramm mit Ausnahmecode 04: Ausfall in zugehörigem Gerät	Siehe Handbuch des Slave-Geräts.
0E65	Antworttelegramm mit Ausnahmecode 05: Quittierung	Siehe Handbuch des Slave-Geräts.
0E66	Antworttelegramm mit Ausnahmecode 06: Belegt, Telegramm zurückgewiesen	Siehe Handbuch des Slave-Geräts.

Fehlercode (W#16#...)	Beschreibung	Abhilfe
0E67	Antworttelegramm mit Ausnahmecode 07: Negative Quittierung	Siehe Handbuch des Slave-Geräts.
1E0D	"Auftragsabbruch wegen Neustart, Wiederanlauf oder Reset"	
1E0E	Statischer Fehler bei Aufruf der SFC DP_RDDAT. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E0F	Statischer Fehler bei Aufruf der SFC DP_WRDAT. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E10	Statischer Fehler bei Aufruf der SFC RD_LGADR. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E11	Statischer Fehler bei Aufruf der SFC RDSYSST. Der Returnwert RET_VAL der SFC wird Ihnen in der Variablen SFCERR im Instanz-DB zur Auswertung zur Verfügung gestellt.	Laden Sie die Variable SFCERR aus dem Instanz-DB.
1E20	Parameter außerhalb des Bereichs.	Geben Sie für die Anweisung einen Parameter im zulässigen Bereich ein.
1E41	Anzahl der am Parameter LEN der Anweisungen angegebenen Bytes unzulässig	Halten Sie den Wertebereich von 1 bis 256 Byte ein.

Auswerten der Variable SFCERR

Nähere Informationen zu den aufgetretenen Fehlern (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H und (1E) 11_H der Ereignisklasse 30 erhalten Sie über die Variable SFCERR.

Die Variable SFCERR können Sie aus dem Instanz-DB der entsprechenden Anweisung laden.

Die Fehlermeldungen, die in der Variablen SFCERR eingetragen werden, finden Sie bei den Systemfunktionen "DPRD_DAT: Konsistente Daten eines DP-Normslaves lesen (Seite 5496)" und SFC15 "DPWR_DAT: Konsistente Daten eines DP-Normslaves schreiben (Seite 5499)" und RD_LGADR: Sämtliche logischen Adressen einer Baugruppe ermitteln (Seite 5793).

Parameter ERROR_NR, ERROR_INFO

Fehlerklassen der Diagnose

Fehlernummern von ... bis	Beschreibung	Bedeutung
1 ... 9	Fehler bei der Initialisierung Anweisung und CP	Bei den Fehlernummern 1...9 wurde die Initialisierung mit Fehler beendet. Der Parameter CP_START_ERROR ist 1. Es ist keine MODBUS-Kommunikation zum Master-System möglich.
10 ... 19	Fehler bei der Bearbeitung eines Funktionscodes	Bei den Fehlernummern 10...19 ist bei der Bearbeitung eines Funktionscodes ein Fehler aufgetreten. Der CP hat der Kommunikations- Anweisung einen unzulässigen Bearbeitungsauftrag gesendet. Der Fehler wird ebenfalls an den Treiber gemeldet. Nachfolgende Bearbeitungs-Aufträge werden weiter bearbeitet.
90 ... 99	Sonstige Fehler	Es ist ein Verarbeitungsfehler aufgetreten. Der Fehler wird nicht an den Treiber gemeldet. Nachfolgende Bearbeitungs-Aufträge werden weiter bearbeitet.

Fehlernummern

ERROR_NR (dezimal)	ERROR_INFO	Fehlertext	Abhilfe
0	0	Kein Fehler.	
1	SFC51->RET_VAL	Fehler beim Lesen der SZL mit der SFC51.	RET_VAL in ERROR_INFO analysieren, Ursache beseitigen.
2	S_SEND->STATUS, S_RCV->STATUS	Timeout bei Baugruppeninitialisierung oder Fehler bei Baugruppeninitialisierung (Fehler in Auftrag S_SEND).	Prüfen Sie, ob für diese Schnittstelle als Protokoll "MODBUS Slave" parametriert wurde. Prüfen Sie, ob die in der Kommunikations- Anweisung angegebene "ID" korrekt ist. Analysieren Sie ERROR_INFO.
11	Startadresse	Unzulässige Startadresse vom Treiber an die Kommunikations-Anweisung übergeben.	Prüfen Sie die Modbus-Adresse des Modbus-Mastersystems.
12	Registeranzahl	Unzulässige Registeranzahl vom Treiber an die Kommunikations-Anweisung übergeben. Registeranzahl = 0.	Prüfen Sie die Registeranzahl des Modbus-Mastersystems, starten Sie ggf. die Baugruppe neu (Mains_ON).
13	Registeranzahl	Unzulässige Registeranzahl vom Treiber an die Kommunikations-Anweisung übergeben: Registeranzahl > 128.	Prüfen Sie die Registeranzahl des Modbus-Mastersystems, starten Sie ggf. die Baugruppe neu (Mains_ON).

ERROR_NR (dezimal)	ERROR_INFO	Fehlertext	Abhilfe
14	Merker M – Endadresse	Zugriffsversuch auf den SIMATIC-Speicherbereich der Merker über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist vom Typ der CPU abhängig.	Verringern Sie die Modbus-Startadresse bzw. die Zugriffslänge im Modbus-Mastersystem.
15	Ausgänge A – Endadresse Eingänge E – Endadresse	Zugriffsversuch auf den SIMATIC-Speicherbereich der Ausgänge über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist vom Typ der CPU abhängig.	Verringern Sie die Modbus-Startadresse bzw. die Zugriffslänge im Modbus-Mastersystem.
16	Zeiten T – Endadresse	Zugriffsversuch auf den SIMATIC-Speicherbereich der Zeiten über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist vom Typ der CPU abhängig.	Verringern Sie die Modbus-Startadresse bzw. die Zugriffslänge im Modbus-Mastersystem.
17	Zähler Z – Endadresse	Zugriffsversuch auf den SIMATIC-Speicherbereich der Zähler über das Bereichsende hinaus. Achtung: Die Bereichslänge in der SIMATIC-CPU ist vom Typ der CPU abhängig.	Verringern Sie die Modbus-Startadresse bzw. die Zugriffslänge im Modbus-Mastersystem.
18	0	Unzulässiger SIMATIC-Speicherbereich vom Treiber an die Kommunikations- Anweisung übergeben.	Starten Sie ggf. die Baugruppe neu (Mains_ON).
19		Fehler beim Zugriff auf die SIMATIC-Peripherie.	Prüfen Sie, ob die benötigte Peripherie vorhanden und fehlerfrei ist.
20	DB#	DB ist nicht vorhanden.	Nehmen Sie den DB in Ihr Projekt auf.
21	DB#	DB-Länge ungültig	Erhöhen Sie die DB-Länge.
22	DB#	DB# liegt unter dem DB-Mindestwert.	Ändern Sie den DB-Mindestwert.
23	DB#	DB# liegt über dem DB-Maximalwert.	Ändern Sie den DB-Maximalwert.
24	Merkeradresse	Merker unterhalb der Mindestgrenze.	Ändern Sie die Mindestgrenzen der Merker in Umsetzungs-DB.
25	Merkeradresse	Merker oberhalb der Maximalgrenze.	Ändern Sie die Höchstgrenzen der Merker in Umsetzungs-DB.
26	Ausgangsadresse	Ausgang unterhalb der Mindestgrenze.	Ändern Sie die Mindestgrenzen der Ausgänge in Umsetzungs-DB.
27	Ausgangsadresse	Ausgang oberhalb der Maximalgrenze.	Ändern Sie die Höchstgrenzen der Ausgänge in Umsetzungs-DB.

ERROR_NR (dezimal)	ERROR_INFO	Fehlertext	Abhilfe
90	S_SEND-> STATUS	Fehler bei der Übertragung eines Quittierungs-telegramms an den Treiber mit S_SEND.	Analysieren Sie die STATUS-Informationen.
94	S_RCV->STATUS	Fehler beim Lesen von SYSTAT mit S_RCV (STATUS).	Analysieren Sie die STATUS-Informationen.

ET 200S 1SI für USS

S_USST: Daten an einen USS-Slave senden

Beschreibung

Die Anweisung S_USST wickelt die Übertragung der Netzdaten (PZD- und evtl. PKW-Daten) an die Slaves je nach der verwendeten Netzdatenstruktur ab.

Arbeitsweise

Die Anweisung nimmt die Parametrierung des aktuellen Slaves aus der Polling-Liste (Parametrierungs-DB) und sendet die Daten aus dem Netzdaten-DB. Sie wertet das Kommunikationssteuerwort des aktuellen Slaves aus (Anstoß eines PKW-Auftrags bzw. Quittierung eines Parameteränderungsberichts), vervollständigt die USS-Sendedaten und überträgt sie an den Sendepuffer des Kommunikationsprozessor-DB. Schließlich stößt sie die Übertragung der Netzdaten an den Slave mittels der Anweisung S_SEND an.

Erkennt die Anweisung einen Parametrierfehler im Parametrierungs-DB, dann wird ein Fehlersignal in Byte Pafe 2 des Netzdaten-DB gespeichert.

Die Anweisung wird einmal pro Zyklus des Automatisierungssystems aufgerufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DBPA	INPUT	INT	Bausteinnummer des Parametrierungs-DB CPU-spezifisch (Null nicht zulässig)
SYPA	INPUT	INT	Anfangsadresse der Systemparameter im Parametrierungs-DB 0 <= SYPA <= 8174
SLPA	INPUT	INT	Anfangsadresse der Slave-Parameter im Parametrierungs-DB 0 <= SLPA <= 8184

Programmaufbau von S_USST

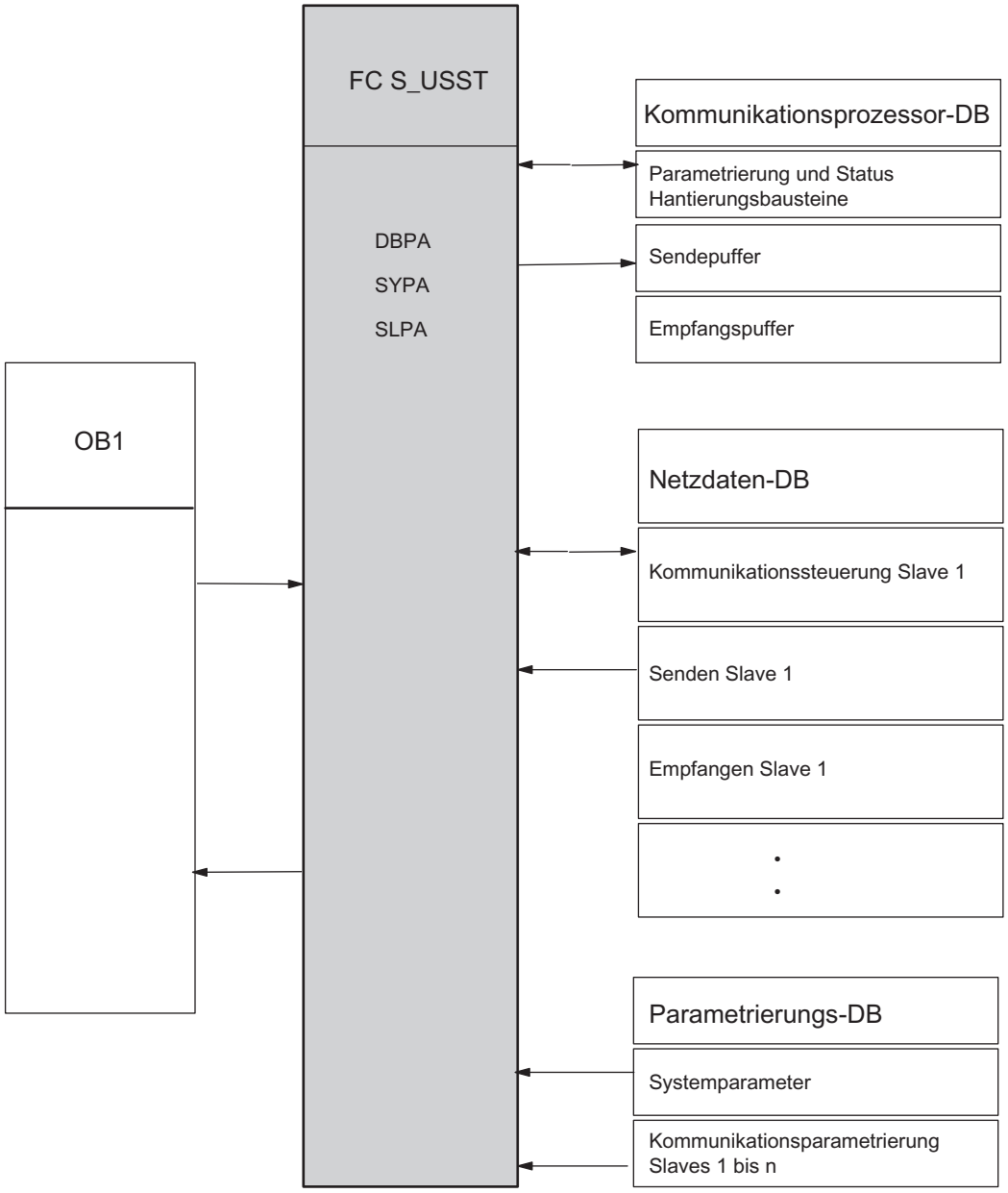

S_USSR: Daten von einem USS-Slave empfangen

Beschreibung

Die Anweisung S_USSR wickelt den Empfang der Netzdaten (PZD- und evtl. PKW-Daten) von den Slaves je nach der verwendeten Netzdatenstruktur ab.

Arbeitsweise

Die Anweisung nimmt die Parametrierung des aktuellen Slaves aus der Polling-Liste (Parametrierungs-DB) und wertet das Statuswort des Bausteins TRANSMIT aus.

Wurde der aktuelle Auftrag fehlerfrei beendet (Bit 9 = 0 im Kommunikations-Statuswort des Netzdaten-DB), werden die eingehenden Daten vom Empfangspuffer des Kommunikationsprozessor-DB in den Netzdaten-DB übertragen und ausgewertet. Anschließend wird das Kommunikations-Statuswort im Netzdaten-DB aktualisiert.

Wurde der aktuelle Auftrag nicht fehlerfrei beendet (Bit 9 = 1 im Kommunikations-Statuswort des Netzdaten-DB), werden die Daten des aktuellen Slaves nicht vom Empfangspuffer des Kommunikationsprozessor-DB übernommen. Die Anweisung gibt dies im Kommunikations-Statuswort des Netzdaten-DB an und trägt die Fehlerursache ins Kommunikations-Fehlerwort ein.

Erkennt der Baustein einen Parametrierfehler im Parametrierungs-DB, dann wird ein Fehlersignal in Byte Pafe 1 des Netzdaten-DB gespeichert.

Die Anweisung wird einmal pro Zyklus des Automatisierungssystems aufgerufen.

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DBPA	INPUT	INT	Bausteinnummer des Parametrierungs-DB CPU-spezifisch (Null nicht zulässig)
SYPA	INPUT	INT	Anfangsadresse der Systemparameter im Parametrierungs-DB 0 <= SYPA <= 8174
SLPA	INPUT	INT	Anfangsadresse der Slave-Parameter im Parametrierungs-DB 0 <= SLPA <= 8184

Programmaufbau von S_USSR

Die folgende Abbildung zeigt den Programmaufbau von S_USSR.

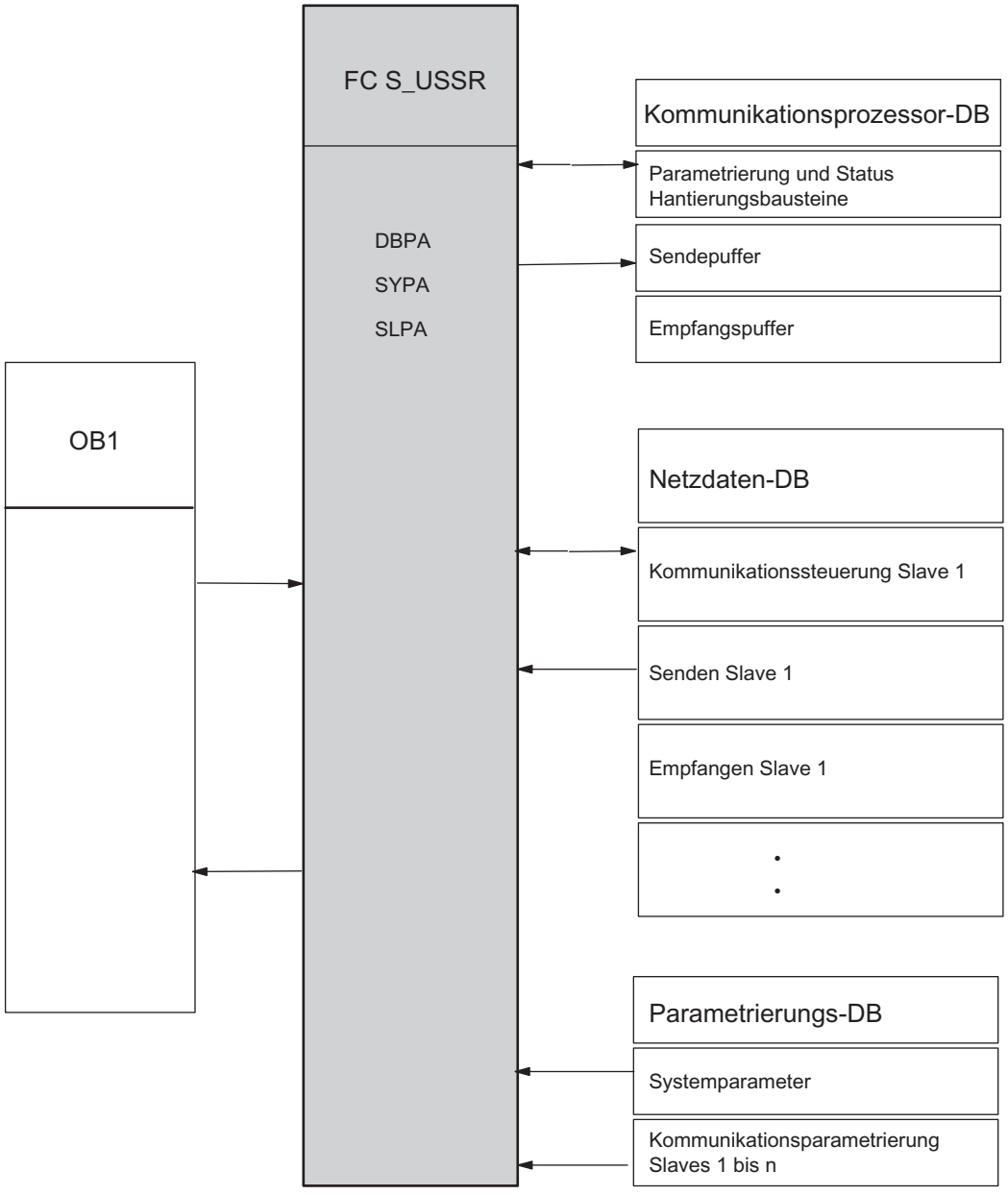

Regel

Hinweis

Die Parameter der Anweisung U_USST entsprechen den Parametern der Anweisung S_USSR. Die beiden Anweisungen greifen auf die gleiche Parametrierung (System- und Slave-Parameter) im Parametrierungs-DB zu und müssen deshalb identisch parametriert werden.

S_USSI: USS initialisieren

Beschreibung

Die Anweisung S_USSI ist eine optionale Funktion.

Wird diese Anweisung bei Anlauf des S7-Systems aufgerufen, werden die Datenbausteine Kommunikationsprozessor-DB, Netzdaten-DB und Parametrierungs-DB, die für die Kommunikation erforderlich sind, generiert. Der DBPA wird zusätzlich vorbesetzt. Die Anweisung S_USSI eignet sich für die Generierung und Vorbesetzung der genannten Datenbereiche nur bei gleicher Netzdatenstruktur für alle Slaves.

Hinweis

Wenn Sie S_USSI nicht verwenden, müssen Sie DBW 2 im Kommunikationsprozessor-DB mit einem Wert vorbelegen (z. B. 20000).

Hinweis

S_USSI an S7-1500

Beachten Sie folgendes, falls Sie die Anweisung S_USSI der dezentralen Peripherie ET 200S 1SI an einer S7-1500 einsetzen.

Der Parameter ANZ zeigt auch im Fehlerfall den Wert 0.

Prüfen Sie, falls die CPU durch die Anweisung S_USSI in STOP geschaltet wird, alle beim Parameter ANZ aufgelisteten Fehlermöglichkeiten.

Arbeitsweise

Beim Aufruf prüft die Anweisung zunächst die Plausibilität ihrer Parametrierung für Slaveanzahl, Netzdatenstruktur, Start-Teilnehmernummer und PKW-Wiederholungen. Erkennt der Baustein dabei einen Fehler, werden Generierung und Vorbesetzung der Datenbausteine nicht ausgeführt. Die CPU geht in den Betriebszustand STOP und der Anwender erhält eine Fehlermeldung über das Fehlerbyte der Anweisung S_USSI. Nach Behebung des Parametrierfehlers müssen alle bereits generierten Datenbausteine vor dem Neustart gelöscht werden.

Nach der Plausibilitätsprüfung prüft der Baustein, ob die zu generierenden Datenbausteine bereits vorhanden sind:

- Sind die zu generierenden Datenbausteine noch nicht vorhanden, werden sie erzeugt und der DBPA wird vorbesetzt.
- Sind die zu generierenden Datenbausteine bereits vorhanden, wird die jeweilige Bausteinlänge geprüft. Ist der DB lang genug, dann wird die Vorbesetzung des Parametrierungs-DB neu erstellt und der Inhalt des Netzdaten-DB und des Kommunikationsprozessor-DB wird gelöscht. Ist ein DB zu kurz, geht die CPU in den Betriebszustand STOP. Der Anwender erkennt den fehlerhaften DB im Anzeigenbyte der Anweisung S_USSI. Zur Behebung des Fehlers müssen die drei Datenbausteine vollständig gelöscht werden. Beim nachfolgenden Neustart werden sie dann neu erzeugt und der Parametrierungs-DB wird vorbesetzt.

Die Anweisung S_USSI muss einmal während des Systemanlaufs aufgerufen werden (OB100).

Parameter

Parameter	Deklaration	Datentyp	Beschreibung
SANZ	INPUT	INT	Anzahl der Slaves mit gleicher Netzdatenstruktur (Systemparameter im DBPA) 1 ≤ SANZ ≤ 31
TNU1	INPUT	INT	Anfangsteilnehmernummer (Stationsnummer) 0 ≤ TNU1 ≤ 31
PKW	INPUT	INT	PKW, Anzahl Anzahl der Wörter der PKW-Schnittstelle 0, 3 oder 4
PZD	INPUT	INT	PZD, Anzahl Anzahl der Wörter der PZD-Schnittstelle 0 ≤ PZD ≤ 16
DBND	INPUT	INT	Nummer Netzdaten-DB CPU-spezifisch (Null nicht zulässig) Zulässiger Bereich: 60000 bis 60999
DBPA	INPUT	INT	Nummer Parametrierungs-DB CPU-spezifisch (Null nicht zulässig) Zulässiger Bereich: 60000 bis 60999
DBCP	INPUT	INT	Nummer Kommunikationsprozessor-DB CPU-spezifisch (Null nicht zulässig) Zulässiger Bereich: 60000 bis 60999
WDH	INPUT	INT	Anzahl zulässiger Wiederholungen eines PKW-Auftrags 0 ≤ WDH ≤ 32767
ANZ	OUTPUT	BYTE	Fehlerbyte 0: kein Fehler 1: Anzahl Slaves zu groß 2: unzulässige Angaben für Netzdatenstruktur 3: Parametrierungs-DB: Unzulässige Nummer oder DB zu kurz 4: Netzdaten-DB: Unzulässige Nummer oder DB zu kurz 5: Fehler Stationsnummer 6: Kommunikationsprozessor-DB: Unzulässige Nummer oder DB zu kurz 7: frei 8: Wiederholungszähler: falscher Wert

Netzdaten-DB

Beschreibung

Der Netzdaten-DB bildet die Schnittstelle zwischen dem Kommunikations- und dem Steuerungsprogramm. Der Anwender muss diesen Baustein in ausreichender Länge "leer" bereitstellen.

Arbeitsweise

Diese Datenbausteine können entweder mit der Anweisung S_USSI beim CPU-Anlauf erzeugt und vorbesetzt (nur DBPA) oder manuell eingegeben werden.

Nur die Sendedaten für einen Slave werden in den dem Slave vom Steuerungsprogramm zugeordneten Sendepuffer des Netzdaten-DB eingetragen. Die Antwortdaten vom Slave werden aus dem entsprechenden Empfangspuffer übernommen (nach Auswertung von Bit 9 im Kommunikations-Steuerwort). Statusworte erlauben die Kontrolle der Kommunikation, das Steuerwort den gezielten Anstoß eines Parametrierauftrags.

Die Kommunikationsschnittstelle enthält für jeden Slave einmal folgende Daten:

- Slavebezogene Kommunikationsdaten (Kommunikationssteuerung, Verfolgung, 6 Datenworte)
- Puffer für den laufenden PKW-Auftrag (nur wenn ein PKW-Bereich vorhanden ist)
- Sendepuffer für Netzdaten (maximal 20 Datenworte)
- Empfangspuffer für Netzdaten (maximal 20 Datenworte)

Die Länge der Sende- und Empfangspuffer richten sich nach der gewählten Netzdatenstruktur. Bei fehlender PKW-Schnittstelle entfällt der Puffer für den aktuellen PKW-Auftrag.

Die Gesamtlänge des benötigten Netzdaten-DB ist abhängig von der Anzahl Slaves und der verwendeten Netzdatenstruktur.

Anzahl Datenworte je Slave = 2 x (PKW + PZD) + PKW + 6

mit PKW = 0, 3 oder 4 und 0 <= PZD <= 16

Beispiel: Ein Antrieb mit einem PKW-Bereich von 3 Worten und einem PZD-Bereich von 2 Worten belegt folglich 19 Datenworte im Netzdaten-DB.

Der Netzdaten-DB ist bei 31 Slaves und maximaler Netzdatenlänge 1550 Datenworte lang. DBW0 ist reserviert.

Slave-Datenbelegung im Netzdaten-DB mit 4 Worten im PKW-Bereich und 0 bis 16 Worten im PZD-Bereich

DBWn	Intern	Kommunikationssteuerung
DBWn+2	Kommunikations-Steuerwort (KSTW)	
DBWn+4	Kommunikations-Statuswort	Kommunikationsverfolgung
DBWn+6	Kommunikations-Fehlerwort	Fehlerstatus
DBW n+8	Intern	PKW-Versuchszähler
DBW n+10	Pafe 1-Byte, Pafe 2-Byte	Parameterfehler

DBW n+12	Parameter-ID	PKE	Puffer für aktuellen PKW-Auftrag	
DBW n+14	Index	IND		
DBW n+16	Parameterwert 1	PWE1		
DBW n+18	Parameterwert 2	PWE2		
DBW n+20	Parameter-ID	PKE	PKW-Bereich	Sendepuffer
DBW n+22	Index	IND		
DBW n+24	Parameterwert 1	PWE1		
DBW n+26	Parameterwert 2	PWE2		
DBW n+28	Steuerwort (STW)	PZD1	PZD-Bereich (max. 16 Worte PZD)	
DBW n+30	Hauptsollwert (HSW)	PZD2		
DBW n+32	Sollwert/Zusatzsteuerwort	PZD3		
DBW n+34	Sollwert/Zusatzsteuerwort	PZD4		
...	...			
DBW n+58	Sollwert/Zusatzteuerwort	PZD16	PKW-Bereich	Empfangspuffer
DBW n+60	Parameter-ID	PKE		
DBW n+62	Index	IND		
DBW n+64	Parameterwert 1	PWE1		
DBW n+66	Parameterwert 2	PWE2		
DBW n+68	Statuswort (ZSW)	PZD1	PZD-Bereich (max. 16 Worte PZD)	
DBW n+70	Hauptistwert (HIW)	PZD2		
DBW n+72	Istwert/Zusatzstatuswort	PZD3		
DBW n+74	Istwert/Zusatzstatuswort	PZD4		
...	...			
DBW n+98	Istwert/Zusatzstatuswort	PZD16		
	•			
(n = 2,4,6...)	•			

Hinweis

Gibt es keinen PKW-Bereich, dann sind der Puffer für aktuelle PKW-Aufträge und der PKW-Bereich im Sendepuffer nicht vorhanden.

Kommunikations-Steuerwort KSTW (DBW n)

Die Bits im Kommunikations-Steuerwort koordinieren das Anwenderprogramm und die Anweisung S_USST.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Anstoß PKW-Auftrag
  Bit 0 wird vom Anwender gesetzt, wenn ein neuer PKW-Auftrag im Sendepuffer steht und bearbeitet werden soll. Das Bit wird durch die Anweisung zurückgesetzt, wenn der PKW-Auftrag übernommen wurde.
- Bit 1: Annahme Parameteränderungsbericht
  Bit 1 wird vom Anwender gesetzt, wenn der Parameteränderungsbericht angenommen wurde. Das Bit wird durch die Anweisung zurückgesetzt, um die Übernahme zu quittieren. Der Slave führt nach dieser Quittierung die unterbrochene Bearbeitung des laufenden Auftrags fort oder überträgt den nächsten Parameteränderungsbericht.

Kommunikations-Statuswort (DBW n+4)

Die Bits im Kommunikations-Statuswort werden von den Anweisungen S_USST und S_USSR gesetzt.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: PKW-Auftrag in Bearbeitung
  Bit 0 wird von der Anweisung S_USST gesetzt, wenn der PKW-Auftrag angenommen wurde und die Parameter-ID (PKE) eine gültige Auftrags-ID enthält. Das Bit wird durch die Anweisung S_USSR zurückgesetzt, wenn der PKW-Auftrag ausgeführt wurde (mit oder ohne Fehler) oder wenn die PKW-Schnittstelle gestört ist.
- Bit 1: PKW-Auftrag fertig ohne Fehler
  Bit 1 wird von der Anweisung S_USSR gesetzt, wenn ein PKW-Auftrag fehlerfrei ausgeführt wurde. Die Antwort ist dem Empfangspuffer zu entnehmen. Das Bit wird von der Anweisung S_USST zurückgesetzt, wenn ein neuer PKW-Auftrag angestoßen wird.

  Hinweis

  Die PKW-Aufträge für den Slave werden in der Reihenfolge der Polling-Liste (DBPA) abgearbeitet. Es ist pro Slave jeweils nur ein Auftrag aktiv. Ist mehr als ein Slave in der Polling-Liste eingetragen, sind die Antwortdaten für einen neuen PKW-Auftrag nur bei einer steigenden Flanke von Bit 1 (bzw. Bit 2) verfügbar.

- Bit 2: PKW-Auftrag fertig mit Fehler
  Bit 2 wird von der Anweisung S_USSR bei Antwort-ID im PKE gesetzt. Die Fehlernummer steht im PWE der Slave-Antwort. Das Bit wird von der Anweisung S_USST zurückgesetzt, wenn ein neuer PKW-Auftrag angestoßen wird.

  Hinweis

  Der vom Anwender zuletzt übertragene PKW-Auftrag wird nach der Bearbeitung in der Sendeschnittstelle gespeichert. Die Übertragung an den Slave wird wiederholt, bis ein neuer Auftrag eingegeben wird. Dadurch können zusätzliche Antworten im Anwenderprogramm erforderlich sein, wenn der Status-PKW-Auftrag mit Fehler (Bit 2) und PKW-Schnittstellenfehler (Bit 4) beendet wird.

- Bit 3: PKW-Auftrags-ID ungültig.
  Bit 3 wird von der Anweisung S_USST gesetzt, wenn die Auftrags-ID 15 im PKE festgestellt wird oder wenn der Index 255 in die Auftrags-ID 4 eingetragen wird. Das Bit wird von der Anweisung S_USST zurückgesetzt, wenn der nächste PKW-Auftrag mit gültiger Auftrags-ID im PKE angestoßen wird.
- Bit 4: PKW-Schnittstelle mit Fehler (Zählerüberlauf).
  Bit 4 wird von der Anweisung S_USSR gesetzt, wenn der PKW-Auftrag nicht innerhalb einer parametrierbaren Anzahl von Auftragswiederholungen (Parameter WDH im Parametrierungs-DB) vom Slave beantwortet wird oder bei Antwort-ID 8 im PKE. Das Bit wird von der Anweisung S_USSR zurückgesetzt, wenn ein neuer PKW-Auftrag angestoßen und ordnungsgemäß ausgeführt wird.
- Bit 5: Antwortdaten enthalten Parameteränderungsbericht.
  Bit 5 wird von der Anweisung S_USSR gesetzt, wenn ein Parameteränderungsbericht vom Slave vorhanden ist (Antwort-ID 9 bis 12 und Umschaltbit 11 invertiert). Das Bit wird von der Anweisung S_USST zurückgesetzt, wenn der Anwender den Parameteränderungsbericht quittiert (Kommunikations-Steuerwort, Bit 1).
- Bit 6: Betriebsstörung beim Slave.
  Bit 6 wird von der Anweisung S_USSR gesetzt und zurückgesetzt. Die Anweisung wertet das Statuswort (Bit 3) des Slave aus.
- Bit 7: Warnung vom Slave steht an.
  Bit 7 wird von der Anweisung S_USSR gesetzt und zurückgesetzt. Die Anweisung wertet das Statuswort (Bit 7) des Slave aus.
- Bit 8: Steuerung vom Automatisierungssystem gefordert.
  Bit 8 wird von der Anweisung S_USSR gesetzt und zurückgesetzt. Die Anweisung wertet das Statuswort (Bit 9) und das Steuerwort (Bit 10) aus.
- Bit 9: Sammelfehler Kommunikation.
  Bit 9 wird von der Anweisung S_USSR gesetzt und zurückgesetzt. Die Anweisung wertet die Rückmeldungen der Standardbausteine S_SEND und S_RCV aus und prüft das empfangene Telegramm in Bezug auf ADR, STX, BCC und LGE. Die Anweisung meldet hier zusätzlich die Überschreitung der Telegrammüberwachungszeit.

Hinweis

Die Empfangsdaten aus dem Netzdaten-DB sind nur gültig, wenn Bit 9 = 0.

Aufbau des Kommunikations-Fehlerworts (DBW n+6)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Adressierungsfehler (ADR)
- Bit 3: Telegrammstart nicht erkannt (erstes Zeichen kein STX)
- Bit 4: falsches Blockprüfzeichen (BCC)

- Bit 6: falsche Telegrammlänge (LGE)
  Die Bits 0, 3, 4 und 6 werden von der Anweisung S_USSR gesetzt, wenn bei der Prüfung des empfangenen Telegramms ein Fehler erkannt wird (ADR, STX, BCC, LGE).
- Bit 7: Telegrammüberwachungszeit abgelaufen
  Bit 7 wird von der Anweisung S_USSR gesetzt, wenn die Zeit zwischen dem Senden des Telegramms vom Master an den Slave und die Ankunft der Antwort vom Slave die vom Programm berechnete zulässige Zeit überschreitet (Telegrammüberwachungszeit).

Die übrigen Bits werden nicht verwendet.

Pafe 1-Byte

Fehlermeldung von der Anweisung S_USSR, Parametrierfehler im Parametrierungs-DB

- Wert 0: Kein Fehler
- Wert 1: Falsche Daten für PKW / PZD

Pafe 2-Byte

Fehlermeldung von der Anweisung S_USST, Parametrierfehler im Parametrierungs-DB

- Wert 0: Kein Fehler
- Wert 1: Falsche Daten für PKW / PZD

Parameter-ID PKE im Sendepuffer

Der Anwender muss die Parameternummer (Bits 0 bis 10) und die Auftrags-ID (Bits 12 bis 15) versorgen. Das Umschaltbit für den Parameteränderungsbericht (Bit 11) wird von den Anweisungen S_USSR und S_USST maskiert.

Parametrierungs-DB

Beschreibung

Der Parametrierungs-DB enthält die für die Steuerung der Kommunikation notwendigen Programmparameter. Der Anwender muss diesen Baustein erzeugen und der Konfiguration des Kommunikationssystems entsprechend vorbesetzen (S_USSI oder manuell). Die Slaves am Bus werden in der Reihenfolge des Eintrags im DBPA bearbeitet (Polling-Liste).

Ein Slave kann auch mehrmals in den Parametrierungs-DB eingetragen werden, wodurch seine Priorität effektiv erhöht wird.

Arbeitsweise

Die Länge des Parametrierungs-DB ist abhängig von der Anzahl n der in einem Buszyklus anzusprechenden Slaves.
Anzahl Datenworte des Parametrierungs-DB = (n x 4) + 5.

Für jede Slavekommunikation werden 4 Datenworte benötigt, für die Systemparameter werden einmalig 4 Datenworte belegt. DBW0 ist reserviert.

Belegung des Datenbausteins

DBW 0	Frei	Systemparameter
DBW 2	DBCP	
DBW 4	SANZ	
DBW 6	SLAV	
DBW 8	WDH	
DBW 10	Anzahl PKW, Anzahl PZD	Kommunikation Parametersatz Slave 1
DBW 12	TUN	
DBW 14	DBND	
DBW 16	KSTW	
DBW 18	Anzahl PKW, Anzahl PZD	Kommunikation Parametersatz Slave 2
DBW 20	TUN	
DBW 22	DBND	
DBW 24	KSTW	
	Anzahl PKW, Anzahl PZD	Kommunikation Parametersatz Slave n
	TUN	
	DBND	
DBW (n x 8 + 8)	KSTW	

Systemparameter

DBCP	Bausteinnummer des Kommunikationsprozessor-DB
SANZ	Anzahl aller Slave-Parametersätze im Parametrierungs-DB. Sollen einzelne Slaves in einem Buszyklus häufiger als andere angesprochen werden, so müssen ihre Slaveparameter mehrmals in den Parametrierungs-DB eingetragen werden. Der Systemparameter SANZ muss entsprechend angepasst werden.
SLAV	Laufende Nummer des aktuellen Slaves. Wird von der FC S_USST und der FC S_USSR zur Ermittlung des aktuellen Parametersatzes benötigt. Dieses Datenwort muss mit 1 vorbesetzt werden. Dies wird von der FC S_USSI durchgeführt, sofern diese verwendet wird.
WDH	Anzahl der zulässigen Wiederholungen eines PKW-Auftrags (Wertebereich: 0 bis 32767). Wird der aktuelle PKW-Auftrag nicht innerhalb der Satznummer beendet, wird die PKW-Schnittstelle als gestört gemeldet.

Slave-Kommunikationsparametrierung

Anzahl PKW, Anzahl PZD	Definition der Netzdatenstruktur Linkes Byte: Anzahl Worte für PKW-Bereich (0, 3, 4) Rechtes Byte: Anzahl Worte für PZD-Bereich (0 bis 16) Davon abweichende Angaben werden als Parametrierfehler erkannt (von den FCs S_USST und S_USSR) und in das Pafe 1-Byte, Pafe 2-Byte des Netzdaten-DB eingetragen.
TUN	Teilnehmernummer, die der am Antrieb eingestellten Busadresse entspricht (0 bis 31).

DBND	Bausteinnummer des Netzdaten-DB.
KSTW	Adresse des Kommunikations-Steuerworts KSTW für den Slave im Netzdaten-DB.

Kommunikationsprozessor-DB

Beschreibung

Über diesen Datenbaustein wird der Datenverkehr zwischen der CPU und der seriellen Schnittstellenbaugruppe ET 200S Modbus/USS abgewickelt. Der Anwender muss diesen Baustein in ausreichender Länge bereitstellen. Der Kommunikationsprozessor-DB muss mindestens 50 Worte lang sein (DBW 0 bis 98).

Aufbau des Kommunikationsprozessor-DBs

DBW 0	Kommunikationsstatus		SENDEN und EMPFANGEN
DBW 2	Maximale Anzahl Zyklen beim Warten auf Empfang	Zykluszähler für Timeout-Bildung beim Warten auf Empfang	FC17
DBW 4	Gemessene Startpause		FC17
DBW 6	Dauer des letzten Zyklus (OB1_MIN_CYCLE)		FC17, OB1
DBW 8	Sendetelegrammlänge (LEN)		SENDEN
DBB10	Frei		
DBB 11 : : DBB 54	Sendepuffer		Telegramm an Baugruppe senden (Länge richtet sich nach der Netzdatenstruktur des aktuellen Slaves)
DBB 55 : : DBB 98	Empfangspuffer		Telegramm von der Baugruppe empfangen (Länge richtet sich nach der Netzdatenstruktur des aktuellen Slaves)

Kommunikationsstatus DBW0

Bit Nr.		
0	Eingang REQ für S_SEND	Dieses Bit wird zurückgesetzt, wenn Bit 8 gesetzt wird.
1	Eingang R für S_SEND	Dieses Bit wird zyklisch von S_USST zurückgesetzt.
2	Ausgang DONE von S_SEND	
3	Ausgang ERROR von S_SEND	
4	Eingang EN_R für S_RCV	Dieses Bit wird zyklisch von S_USSR gesetzt.
5	Eingang R für S_RCV	Dieses Bit wird zyklisch von S_USSR zurückgesetzt.
6	Ausgang NDR von S_RCV	

Bit Nr.		
7	Ausgang ERROR von S_RCV	
8	Auftrag in Bearbeitung (Bit DONE von S_SEND gespeichert)	Dieses Bit wird von S_USST gesetzt und zurückgesetzt.

Dauer des letzten Zyklus DBW6

Mit diesem Parameter misst S_USST die Antwortzeit eines Slaves. Das Anwenderprogramm muss die Zykluszeit des Automatisierungssystems (OB1_MIN_CYCLE) vor jedem Aufruf von S_USST in diesen Parameter kopieren.

SIMATIC NET-CPs

Industrial Ethernet

Einführung Ethernet

Allgemeine Hinweise

Verfügbare Anweisungen

Nachfolgende Liste gibt die bei der Lieferung verwendeten Nummern der Programmbausteine (Anweisungen) an. Die Nummern können von Ihnen geändert werden.

Beachten Sie, dass Sie unterschiedliche Anweisungen für S7-300 und S7-400 (getrennte Bibliotheken) verwenden müssen.

Kommunikationsdienst / Funktionsbereich	Typ der Anweisung	Bibliothek	
		CP 300	CP 400
SEND / RECEIVE (Offene Kommunikationsdienste)	AG_SEND	x	x
	AG_RECV	x	x
	AG_LSEND	x [2)]	x
	AG_LRECV	x [2)]	x
	AG_LOCK	x	x
	AG_UNLOCK	x	x
	AG_CNTEX	x	x [3)]
	AG_CNTRL	x	x [3)]
Programmierte Kommunikationsverbindungen	IP_CONFIG	x	x

Kommunikationsdienst / Funktionsbereich	Typ der Anweisung	Bibliothek	
		CP 300	CP 400
S7-Kommunikation	BSEND	x	
	BRCV	x	
	PUT	x	
	GET	x	
	USEND	x	
	URCV	x	
	C_CNTRL	x	
FTP	FTP_CMD	x	x
PROFINET IO	PNIO_SEND	x	
	PNIO_RECV	x	
	PNIO_RW_REC	x	
	PNIO_ALARM	x	

1) Der PN_InOut / PN_InOut_Fast wird zusammen mit dem Engineering-Werkzeug SIMATIC iMap geliefert.

2) nicht für aktuelle CPs zu verwenden und nicht mehr Lieferbestandteil der aktuellen Bibliothek SIMATIC_NET_CP

3) abhängig vom CP-Typ

Welche Anweisungsversion verwenden ?

Die folgenden Beschreibungen enthalten auch Hinweise auf abweichende Verhaltensweisen bei unterschiedlichen Anweisungsversionen. Bitte achten Sie auf die Versionskennzeichnungen der von Ihnen verwendeten Anweisungen.

Eine jederzeit aktuelle Ausgabe der Versionshistorie für die SIMATIC NET Anweisungen finden Sie im Internet unter der folgenden Beitrags-ID:

9836605 (http://support.automation.siemens.com/WW/view/de/9836605)

Die mit der Automation Workbench installierten globalen Bibliotheken enthalten die zum Zeitpunkt der Workbench-Freigabe aktuellen Anweisungsversionen.

Hinweis

Es wird empfohlen, für alle Baugruppentypen immer die aktuellen Anweisungsversion zu verwenden.

Informationen über die aktuellen Anweisungsversionen sowie die aktuellen Anweisungen zum Download finden Sie bei unserem Customer Support im Internet:

http://support.automation.siemens.com/WW/view/de/8797900

Bei älteren Baugruppentypen setzt diese Empfehlung voraus, dass Sie den für diesen Baugruppentyp aktuellen Firmware-Stand verwenden.

Anweisungen im Ersatzteilfall

Unter Ersatzteilfall wird hier der Austausch einer Baugruppe gegen eine andere Baugruppe, mit eventuell neuerem Ausgabestand verstanden.

Hinweis

Beachten Sie, dass Sie im Ersatzteilfall im Anwenderprogramm nur die für den projektierten CP-Typ zugelassenen Anweisungen verwenden.

Es wird empfohlen, für alle Baugruppentypen immer die aktuellen Versionen der Anweisungen zu verwenden. Bei älteren Baugruppentypen setzt diese Empfehlung voraus, dass Sie den für diesen Baugruppentyp aktuellen Firmware-Stand verwenden.

Weitere Informationen zum Ersatzteilfall finden Sie bei unserem Customer Support im Internet.

Die spezifischen Gerätehandbücher der S7-CPs geben Auskunft über die Kompatibilität der S7-CPs und der zugehörenden Anweisungen.

Siehe auch

Programmierhandbuch "Funktionen (FC) und Funktionsbausteine (FB) für SIMATIC NET S7-CPs" (http://support.automation.siemens.com/ww/view/de/30564821)

Aufruf der Anweisungen parametrieren

Bevor die Anweisungen im Detail beschrieben werden, folgen an dieser Stelle einige allgemeine Hinweise zum Aufruf und zur Parametrierung der Anweisungen.

Allgemeine Angaben können hier zu folgenden, bei allen Anweisungen vorhandenen Parametergruppen gemacht werden:

- Parameter zur CP- und Verbindungszuordnung (Eingangsparameter)
- Parameter zur Angabe eines CPU-Datenbereichs (Eingangsparameter)
- Statusinformationen (Ausgangsparameter)

Kommunikations-Anweisungen für S7-300 aufrufen

ACHTUNG
Es ist nicht zulässig, die Kommunikations-Anweisungen für S7-300 in mehreren Ablaufebenen aufzurufen! Wenn Sie beispielsweise eine Kommunikations-Anweisung im OB1 und im OB35 aufrufen, dann könnte die Bearbeitung der Anweisung durch den jeweils höherprioren OB unterbrochen werden. Wenn Sie die Anweisungen in mehreren OBs aufrufen, müssen Sie programmtechnisch dafür sorgen, dass eine laufende Kommunikations-Anweisung nicht durch eine andere Kommunikations-Anweisung unterbrochen wird, beispielsweise über System-Anweisungen Alarme sperren/freigeben.

Parameter zur Angabe eines CPU-Datenbereichs (Eingangsparameter)

Angabe des Datenbereichs in der CPU

Beim Aufruf einer Anweisung übergeben Sie die Adresse und Länge des Datenbereiches in der CPU, in dem Nutzdaten bereitgestellt oder abgelegt werden sollen oder weitere Parametrierinformationen enthalten sein können

Zur Adressierung dieses Bereichs wird der Datentyp ANY-Zeiger verwendet.

Statusinformationen (Ausgangsparameter)

Statusanzeigen auswerten

Für die Statusauswertung sind im Anwenderprogramm die Parameter auszuwerten:

- DONE bzw. NDR
  Diese Parameter (DONE bei Sendaufträgen bzw. NDR bei Empfangsaufträgen) melden den (positiven) Abschluss einer Auftragsausführung.
- ERROR
  Meldet, wenn der Auftrag nicht fehlerfrei ausgeführt werden konnte.
- STATUS
  Der Parameter liefert Detailinformation zur Auftragsausführung. Statusanzeigen können bereits während der Auftragsausführung geliefert werden (DONE=0 und ERROR=0).

Hinweis

Beachten Sie, daß die Statusanzeigen DONE, NDR, ERROR, STATUS bei jedem Aufruf einer Anweisung aktualisiert werden.

Statusanzeigen beim CP-Anlauf

Bei einem Neu-/Wiederanlauf des Ethernet-CP werden die Ausgabeparameter der Anweisung wie folgt zurückgesetzt:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS =
  - 8180_H bei AG_RECV / AG_LRECV
  - 8181_H bei AG_SEND / AG_LSEND / AG_SSEND

Anweisungen für die Offenen Kommunikationsdienste (SEND/RECEIVE-Schnittstelle)

Anweisungen für die SEND/RECEIVE-Schnittstelle

Übersicht

Für die Übertragung von Daten an der SEND/RECEIVE-Schnittstelle stehen folgende Anweisungen zur Verfügung:

Anweisung	verwendbar bei [1]		Bedeutung
S7-300	S7-400		
AG_SEND	x	x	für Daten senden
AG_RECV	x	x	für Daten empfangen
AG_LSEND		x	für Daten senden
AG_LRECV		x	für Daten empfangen
AG_SSEND		x	für Daten senden
AG_SRECV		x	für Daten empfangen

1) Anmerkungen zu den Anweisungen bei S7-300 und S7-400

- für S7-300 gilt:
  - Bei den aktuellen Ausgabeständen der Ethernet-CPs werden ausschließlich AG_SEND und AG_RECV verwendet; die Datenlänge kann hier bis zu 8192 Byte betragen.
  - In S7–300-CPs (bis 6GK7 343–1EX10–0XE0 mit Firmware V2.2) müssen Sie auf TCP-Verbindungen statt AG_RECV die Anweisung AG_LRECV verwenden.
- für S7-400 gilt:
  - Bei AG_SEND / AG_RECV ist die Datenlänge pro Auftrag auf <=240 Byte beschränkt. Längere Datensätze (bis zu 8192 Byte) können mit Hilfe der Anweisungen AG_LSEND oder AG_LRECV übertragen werden.
  - AG_SSEND und AG_SRECV dienen der beschleunigten Übertragung von Daten, indem eine optimierte Kommunikation zwischen CPU und CP in der S7−Station verwendet wird. Die schnelle Kommunikation bleibt ohne Auswirkung auf die LAN−Kommunikation. Die beiden Anweisungen werden ab STEP 7 V5.4 SP3 unterstützt.
  - In der S7–400 kann auf TCP-Verbindungen nicht AG_RECV verwendet werden, sondern nur die Anweisung AG_LRECV oder AG_SRECV.

Weitere Informationen

Bitte informieren Sie sich zu dem von Ihnen verwendeten S7-CP im gerätespezifischen Handbuch bzgl. des unterstützten Datenbereiches. Eine Übersicht über die Versionen der Anweisungen finden Sie in der SIMATIC NET-Bausteinhistorie.

Anwendung

Die folgende Darstellung verdeutlicht die Anwendung der hier beschriebenen Anweisungen für den bidirektionalen Datentransfer über eine projektierte Verbindung.

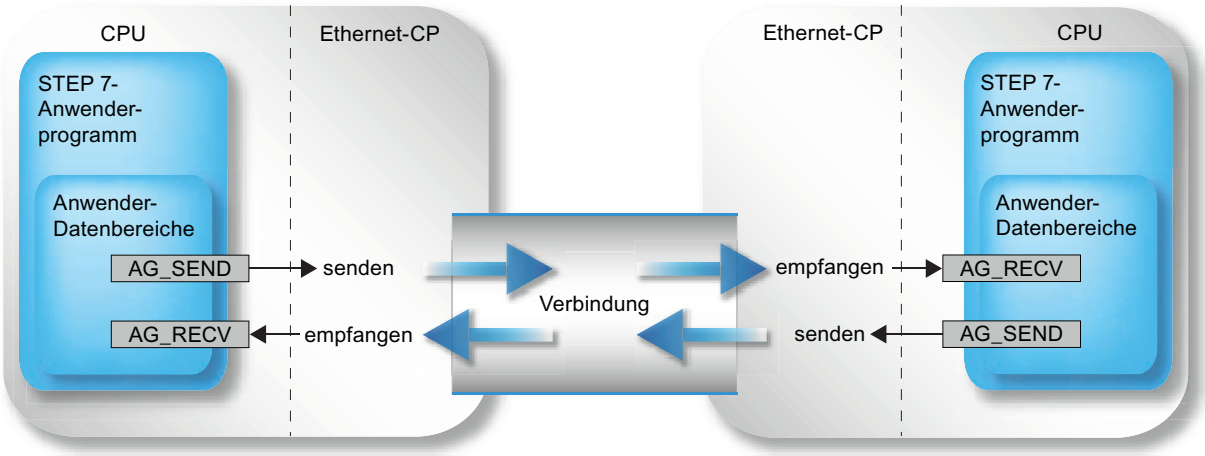

Angabe des Datenbereichs in der CPU

Beim Aufruf einer Anweisung übergeben Sie Adresse und Länge des Datenbereiches in der CPU. Beachten Sie bitte, dass die maximale Länge des Datenbereichs vom verwendeten Typ und der Version der Anweisung abhängig ist.

- AG_SEND und AG_RECV
  Mit diesen Anweisungen können bis zur Version V3.0 maximal 240 Byte gesendet oder empfangen werden. Die aktuellen Versionen lassen bei S7-300 einen Datenbereich bis zu 8192 Byte zu. Bei S7-400 muss für die Übertragung größerer Datenbereiche weiterhin AG_LSEND / AG_LRECV verwendet werden.
- AG_LSEND / AG_LRECV
  Bei den CPs der S7-400 sowie bei früheren Ausgabeständen der S7-300 können größere Datenbereiche nur über AG_LSEND bzw. AG_LRECV übertragen werden. Bitte informieren Sie sich in der Produktinformation des CP bzgl. des unterstützten Datenbereiches.
- AG_SSEND / AG_SRECV
  Mit CPs der S7−400, die PROFINET-Kommunikation zusammen mit CPUs ab Ausgabestand 5.1 unterstützen, können Daten über AG_SSEND bzw. AG_SRECV mit einer höheren Übertragungsgeschwindigkeit übertragen werden (gilt nicht für den CP 443−1 Advanced 6GK7 443–1EX41–0XE0).
  Welche CP-Typen von CPUs ab Ausgabestand 5.1 unterstützt werden, können Sie dem Gerätehandbuch Ihres CP entnehmen (Kapitel "Voraussetzungen für den Einsatz").

Die folgende Tabelle gibt an, welche Grenzwerte bei den verschiedenen Verbindungstypen gelten.

ANWEISUNG	ISO-Transport	ISO-on-TCP	TCP	UDP
AG_LSEND (S7-400) AG_SEND (S7-300)	8192 Byte	8192 Byte	8192 Byte	2048 Byte
AG_SEND (S7-400)	240 Byte	240 Byte	240 Byte	240 Byte

ANWEISUNG	ISO-Transport	ISO-on-TCP	TCP	UDP
AG_LRECV (S7-400) AG_RECV (S7-300)	8192 Byte	8192 Byte	8192 Byte	2048 Byte
AG_RECV (S7-400)	240 Byte	240 Byte	240 Byte	240 Byte
AG_SSEND (S7-400) AG_SRECV (S7-400)	1452 Byte	1452 Byte	1452 Byte	1452 Byte

Hinweis

Beachten Sie zur Länge des übertragbaren Datenbereichs bei älteren Ausgabeständen der Ethernet-CPs bitte die Angaben in der Produktinformation / Gerätehandbuches des von Ihnen verwendeten Ethernet-CPs.

Anwendung ohne Auftragsheader

Bei spezifizierter Verbindung sind Adress- und Auftragsparameter durch die Verbindungsprojektierung festgelegt. Das Anwenderprogramm stellt daher nur die Nutzdaten im UDP-Datenbereich beim Senden mit AG_SEND / AG_LSEND / AG_SSEND bereit, bzw. empfängt diese mit AG_RECV / AG_LRECV / AG_SRECV.

Anwendung mit Auftragsheader

Freie UDP-Verbindungen erfordern einen Auftragsheader im Anwender-Datenbereich.

Entnehmen Sie der folgenden Darstellung die Struktur des Auftragspuffers und die Bedeutung und Platzierung (high byte / low byte) der Parameter im Auftragsheader.

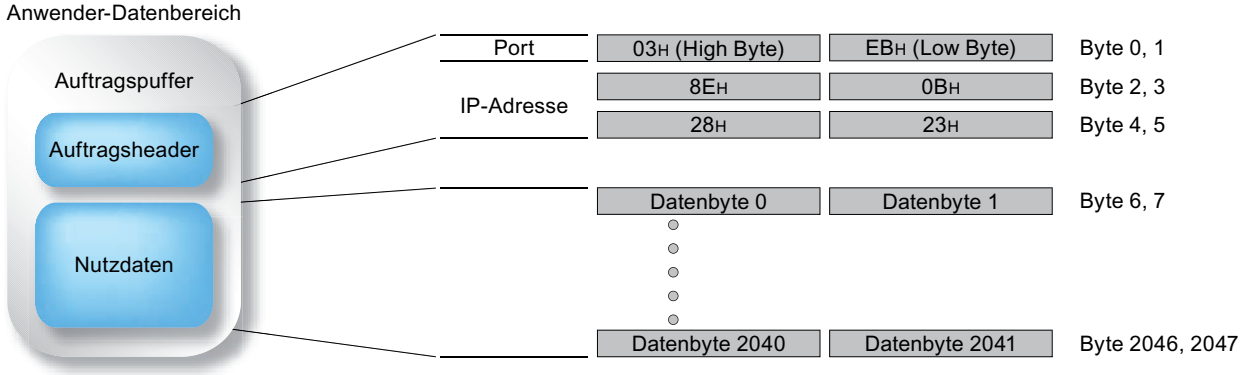

Bild 4-36 Senden und Empfangen über eine freie UDP-Verbindung per Programm

- Im Bild (Eingaben sind hexadezimal) wird beispielhaft von folgender IP-Adresse ausgegangen: 142.11.40.35;
- Für die Port-Adresse 1003 ist z. B. einzutragen: für High Byte: 03_H; für Low Byte: EB_H.
- Der Anwender-Datenbereich kann bis zu 2048 Byte umfassen. Es können bis zu 2042 Byte Nutzdaten übertragen werden. 6 Byte sind für den Auftragsheader reserviert.
  Beachten Sie, dass beim Aufruf der Anweisung die angegebene Datenlänge (Parameter LEN) den Header und die Nutzdaten umfassen muss!

Aufrufparameter erst nach Auftragsbestätigung verändern

Hinweis

Die Aufrufparameter an der Aufrufschnittstelle der Anweisung AG_SEND bzw. AG_RECV dürfen Sie nach dem Auftragsanstoß erst dann wieder verändern, nachdem die Anweisung die Auftragsausführung mit DONE=1 oder mit ERROR=1 bestätigt hat.

Wird dies nicht beachtet, kann es sein, dass die Auftragsausführung mit Fehler abgebrochen wird.

Statusanzeige an der Aufrufschnittstelle der Anweisung; Besonderheiten bei AG_SEND / AG_RECV (S7-300)

Das folgende Verhalten gilt nur für die S7-300 mit Programmbausteinen ab Version V4.0.

Bei AG_SEND sowie AG_RECV erhalten Sie in folgenden Betriebsfällen die nachfolgend genannten Anzeigen:

- CP befindet sich in STOP.
- Verbindung ist nicht projektiert.
- Verbindung ist nicht aufgebaut.
- Verbindung ist abgebrochen.

Anzeigen:

- AG_SEND:
  DONE=0; ERROR=1; Status=8183_H
- AG_RECV:
  DONE=0; ERROR=0; Status=8180_H
  oder
  DONE=0; ERROR=1; Status=8183_H

Siehe auch

Allgemeine Hinweise (Seite 6770)

AG_SEND / AG_LSEND / AG_SSEND

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet)

Beschreibung

AG_SEND / AG_LSEND / AG_SSEND übergeben Daten an den Ethernet-CP zur Übertragung über eine projektierte Verbindung.

Der angegebene Datenbereich kann ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn der gesamte Anwenderdatenbereich über Ethernet gesendet werden konnte.

Anmerkung:
Alle folgenden Angaben gelten, soweit nicht anders angegeben, gleichermaßen für AG_SEND, AG_LSEND und AG_SSEND.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

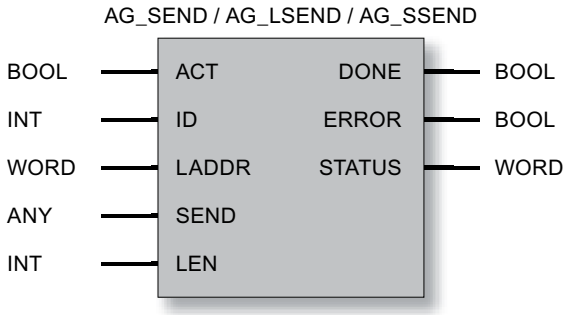

Hinweis

Falls Sie die Anweisung AG_SSEND nutzen wollen, dann müssen Sie in der Verbindungsprojektierung bei den Verbindungseigenschaften die Betriebsart "SPEED SEND/ RECV" auswählen.

Siehe auch

Parameter bei AG_SEND / AG_LSEND / AG_SSEND (IE) (Seite 6779)

Parameter DONE, ERROR, STATUS (Seite 6780)

AG_SEND / AG_LSEND (PROFIBUS) (Seite 6922)

Programmierhandbuch "Funktionen (FC) und Funktionsbausteine (FB) für SIMATIC NET S7-CPs" (http://support.automation.siemens.com/ww/view/de/30564821)

Parameter bei AG_SEND / AG_LSEND / AG_SSEND (IE)

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisungen AG_SEND / AG_LSEND / AG_SSEND:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL	0,1	Beim Aufruf der Anweisung mit ACT = 1 werden LEN Bytes aus dem mit dem Parameter SEND angegebenen Datenbereich gesendet. Beim Aufruf der Anweisung mit ACT = 0 werden die Statusanzeigen DONE, ERROR und STATUS aktualisiert.
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	Im Parameter ID wird die Verbindungsnummer der Verbindung angegeben.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
SEND	INPUT	ANY		Angabe von Adresse und Länge Die Adresse des Datenbereichs verweist alternativ auf: • Merkerbereich • Datenbausteinbereich
LEN	INPUT	INT	bei ISO-Transport und ISO-on-TCP / TCP: 1,2,...8192 bzw. bis "Längenangabe beim Parameter SEND") bei UDP: 1,2,...2048 bzw. bis "Längenangabe beim Parameter SEND")	Anzahl der Bytes, die mit dem Auftrag aus dem Datenbereich gesendet werden sollen. Die Angabe kann im Bereich von 1 bis "Längenangabe beim Parameter SEND" liegen. • Typ der Anweisung beachten: – bei S7-300 Mit den aktuellen Versionen des AG_SEND lassen sich bis zu 8192 Byte (2048 Byte bei UDP) übertragen. – bei S7-400 Mit dem AG_SEND ist der Datenbereich generell auf maximal 240 Byte beschränkt. Bei S7-400 ist zu beachten: • Erhöhte Performance bei kurzen Datensätzen: Die Übertragung von Datensätzen bis zu 240 Byte erfolgt mit erhöhter Performance! Dies gilt unabhängig vom verwendeten Typ (AG_SEND / AG_LSEND). • Mit dem AG_SSEND ist der Datenbereich auf maximal 1452 Byte beschränkt.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
DONE	OUTPUT	BOOL	0: Auftrag läuft 1: Auftrag ausgeführt	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Solange DONE = 0 ist, kann kein weiterer Auftrag angestoßen werden. DONE wird vom CP bei Auftragsannahme auf 0 gesetzt.
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6780)
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe Parameter DONE, ERROR, STATUS (Seite 6780)

Siehe auch

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Seite 6777)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-345 Anzeigen AG_SEND / AG_LSEND / AG_SSEND

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fertig ohne Fehler.
0	0	0000_H	Kein Auftrag in Bearbeitung.
0	0	8181_H	Auftrag läuft.
0	1	7000_H	Die Anzeige ist nur bei S7-400 möglich: Die Anweisung wurde mit ACT=0 aufgerufen; der Auftrag wird jedoch nicht bearbeitet.
0	1	8183_H	Die Projektierung fehlt oder der ISO-/TCP-Dienst im Ethernet-CP ist noch nicht gestartet.
0	1	8184_H	• Unzulässiger Datentyp für den Parameter SEND angegeben. • Systemfehler (Der Quelldatenbereich ist fehlerhaft).
0	1	8185_H	Parameter LEN größer als Quell-Bereich SEND.

DONE	ERROR	STATUS	Bedeutung
0	1	8186_H	Parameter ID ungültig. • ID != 1,2....16 (S7-300) • ID != 1,2....64.(S7-400)
0	1	8302_H	Keine Empfangsressourcen bei Ziel-Station. Empfänger-Station kann empfangene Daten nicht schnell genug verarbeiten bzw. hat keine Empfangsressourcen bereitgestellt.
0	1	8304_H	Die Verbindung ist nicht aufgebaut. Der Sendeauftrag sollte erst nach einer Wartezeit >100 ms erneut abgesetzt werden.
0	1	8311_H	Zielstation ist unter der angegebenen Ethernet-Adresse nicht erreichbar.
0	1	8312_H	Ethernet-Fehler im CP.
0	1	$8F22_H$	Quell-Bereich ungültig, z. B.: Bereich im DB nicht vorhanden Parameter LEN < 0
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	Falsche DB-Nummer
0	1	$8F3A_H$	Bereich nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F44_H$	Der Zugriff auf einen in der Anweisungsbearbeitung zu lesenden Parameter ist gesperrt.
0	1	$8F7F_H$	Interner Fehler, z. B. unzulässige ANY-Referenz z.B. Parameter LEN = 0 .
0	1	8090_H	• Baugruppe mit dieser Baugruppen-Anfangsadresse nicht vorhanden; • Die verwendete Anweisung passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Anweisungen für S7-300 und S7-400 zu verwenden).
0	1	8091_H	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	1	8092_H	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben. (nur bei S7-400)
0	1	$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut. (bei neueren CPU-Ausgabeständen)
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Die Längenangabe (im Parameter LEN) ist falsch.
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	1	$80C4_H$	Kommunikationsfehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll
0	1	$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Seite 6777)

AG_RECV / AG_LRECV / AG_SRECV

AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet)

Beschreibung

Die Anweisung AG_RECV / AG_LRECV / AG_SRECV übernimmt vom Ethernet-CP die über eine projektierte Verbindung übertragenen Daten.

Der für die Datenübernahme angegebene Datenbereich kann ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn die Daten vom Ethernet-CP übernommen werden konnten.

Anmerkung:
Alle folgenden Angaben gelten, soweit nicht anders angegeben, gleichermaßen für AG_RECV und AG_LRECV / AG_SRECV.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

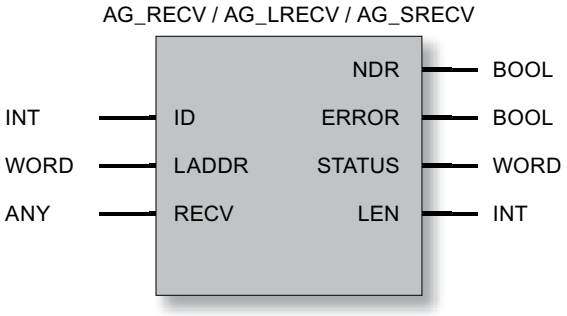

Hinweis

Falls Sie die Anweisung AG_SRECV nutzen wollen, dann müssen Sie in der Verbindungsprojektierung bei den Verbindungseigenschaften die Betriebsart "SPEED SEND/ RECV" auswählen.

Siehe auch

AG_RECV / AG_LRECV (PROFIBUS) (Seite 6929)

Parameter bei AG_RECV / AG_LRECV / AG_SRECV (IE) (Seite 6783)

Parameter DONE, ERROR, STATUS (IE) (Seite 6784)

Parameter bei AG_RECV / AG_LRECV / AG_SRECV (IE)

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisungen AG_RECV / AG_LRECV / AG_SRECV:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	Im Parameter ID wird die Verbindungsnummer der ISO-Transport-Verbindung angegeben.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
RECV	INPUT	ANY		Angabe von Adresse und Länge Die Adresse des Datenbereiches verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Beachten Sie für die Längenangabe: Die Übertragung von Datensätzen bis zu 212 Byte erfolgt mit erhöhter Performance, wenn Sie die Längenangabe am Parameter RECV ebenfalls auf 212 Byte beschränken! Beachten Sie beim AG_SRECV: Beim AG_SRECV müssen Sie RECV immer auf die maximale Empfangspufferlänge von 1452 Byte setzen. Andernfalls kann es in bestimmten Fällen zu folgendem Fehler kommen: NDR=0; ERROR=1; STATUS=8185_H
NDR	OUTPUT	BOOL	0: - 1: neue Daten	Der Parameter zeigt an, ob neue Daten übernommen wurden. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Seite 6782)
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und STATUS siehe AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Seite 6782)

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und ERROR siehe AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Seite 6782)
LEN	OUTPUT	INT	**bei ISO-Transport und ISO-on-TCP:** 1,2,...8192 **bei UDP:** 1,2,...2048	Gibt die Anzahl der Bytes an, die vom Ethernet-CP in den Datenbereich übernommen wurden. Typ der Anweisung beachten: • bei S7-300 Mit den aktuellen Versionen des AG_RECV lassen sich bis zu 8192 Byte (2048 Byte bei UDP) übertragen. • bei S7-400 Mit dem AG_RECV ist der Datenbereich generell auf max. 240 Byte beschränkt. Mit dem AG_SRECV ist der Datenbereich auf max. 1452 Byte beschränkt.

Parameter DONE, ERROR, STATUS (IE)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus NDR, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-346 Anzeigen AG_RECV / AG_LRECV / AG_SRECV

NDR	ERROR	STATUS	Bedeutung
1	0	0000_H	Neue Daten übernommen.
0	0	8180_H	Es liegen noch keine Daten vor (nicht bei AG_SRECV).
0	0	8181_H	Auftrag läuft.
0	1	8183_H	• Die Projektierung fehlt; • Der ISO-Transport-Dienst im Ethernet-CP ist noch nicht gestartet; • Die Verbindung ist nicht aufgebaut.
0	1	8184_H	• Unzulässiger Datentyp für den Parameter RECV angegeben; • Systemfehler.
0	1	8185_H	Ziel-Puffer (RECV)ist zu klein.

NDR	ERROR	STATUS	Bedeutung
0	1	8186_H	Parameter ID ungültig. ID != 1,2....16 (S7-300). ID != 1,2....64.(S7-400)
0	1	8304_H	Die Verbindung ist nicht aufgebaut. Der Empfangsauftrag sollte erst nach einer Wartezeit >100 ms erneut abgesetzt werden.
0	1	$8F23_H$	Quell-Bereich ungültig. z. B.: Bereich im DB nicht vorhanden.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. aktuellen Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. aktuellen Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummernfehler
0	1	$8F3A_H$	Zielbereich nicht geladen (DB).
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F45_H$	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F7F_H$	Interner Fehler, z.B. unzulässige ANY-Referenz.
0	1	8090_H	• Baugruppe mit dieser Baugruppen-Anfangsadresse nicht vorhanden oder CPU in STOP • Die verwendete Anweisung passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Anweisungen für S7-300 und S7-400 zu verwenden).
0	1	8091_H	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	1	8092_H	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben. (nur bei S7-400)
0	1	$80A0_H$	Negative Quittung beim Lesen von Baugruppe.
0	1	$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Mögliche Ursachen: • Der Zielbereich ist ungültig. • Der Zielbereich ist zu klein. • Der Zielbereich für die Empfangsdaten wurde nicht ausreichend bemessen. Abhilfe: Führen Sie einen weiteren Empfangsaufruf mit maximaler Empfangspuffergröße durch. Dies gilt unabhängig vom Verbindungstyp (Uni-/Multi-/Broadcast) und von der Gerätefamilie (S7-300 / S7-400).
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	1	$80C4_H$	Kommunikationsfehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll
0	1	$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Seite 6782)

Anweisungen zur Zugriffskoordinierung bei FETCH/WRITE

Anweisungen zur Zugriffskoordinierung bei FETCH/WRITE

Übersicht

Für die Funktion FETCH/WRITE stehen folgende Anweisungen für die Zugriffskoordinierung zur Verfügung:

Anweisung	verwendbar bei		Bedeutung
S7-300	S7-400		
AG_LOCK	x	x	Sperren des externen Datenzugriffes mittels FETCH/WRITE.
AG_UNLOCK	x	x	Freigeben des externen Datenzugriffes mittels FETCH/WRITE.

Projektierung beachten

Wenn Sie AG_LOCK und AG_UNLOCK verwenden, dann müssen Sie für CPs in S7-400 Stationen folgende Angaben in der Projektierung vornehmen:

- Unter "Eigenschaften > Adressen"
  Es ist die Option "Adresseinstellung für LOCK/UNLOCK" zu wählen, sofern diese Auswahlmöglichkeit besteht.

Arbeitsweise

Mit diesen Anweisungen haben Sie die Möglichkeit, den Zugriff auf Systemspeicherbereiche so zu koordinieren, dass keine inkonsistenten Daten erzeugt und übertragen werden. Die Kontrolle geht hierbei vom Anwenderprogramm in der S7-CPU aus, das über den AG_LOCK-Aufruf einen externen FETCH/WRITE-Zugriff bei Bedarf sperren kann. Nach einer bestimmten Zeit oder nachdem der eigene Schreib-/Lesezugriff abgeschlossen ist, kann mit einem AG_UNLOCK-Auftrag der externe Zugriff wieder freigegeben werden.

Zusätzlich kann ausgenutzt werden, dass die Zugriffssperre nur auf der im Aufruf angegebene FETCH/WRITE-Verbindung gilt. Falls mehrere FETCH/WRITE-Verbindungen projektiert werden, können diese dann beispielsweise gezielt für bestimmte Systemspeicherbereiche genutzt werden und es kann eine entsprechend selektive Zugriffskoordination realisiert werden.

Die folgende Ablaufdarstellung zeigt den üblichen zeitlichen Ablauf einer mit AG_LOCK und AG_UNLOCK gesteuerten Speicherzugriffskoordination im Anwenderprogramm.

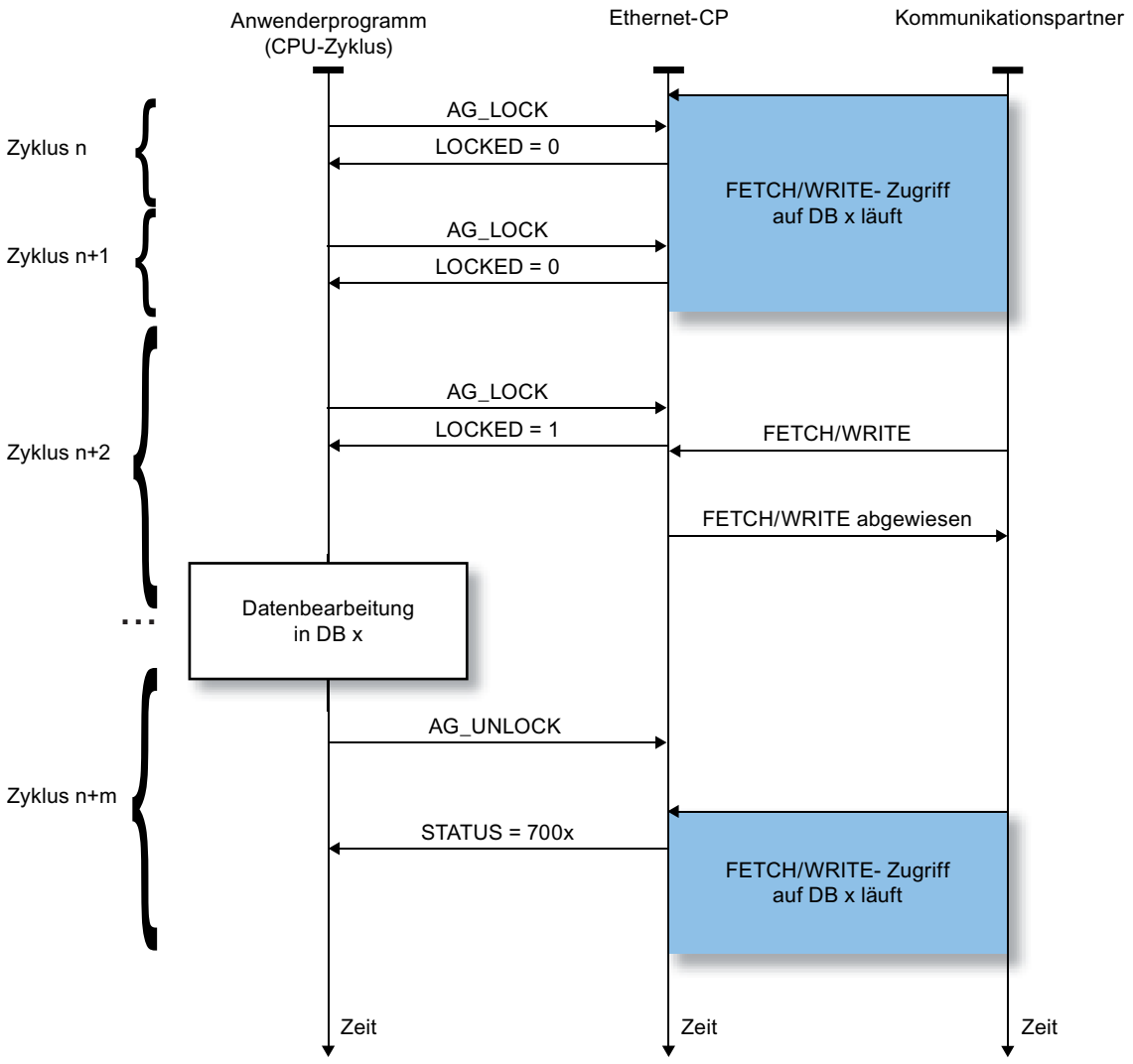

Der Sperrauftrag muss zunächst im Anwenderprogramm über die Anzeige im Rückgabeparameter LOCKED überwacht werden. Solange LOCKED=0 angezeigt wird, muss von einem noch laufenden externen FETCH/WRITE-Zugriff ausgegangen werden.

Mit LOCKED=1 wird angezeigt, dass die Sperre aktiv ist; Daten können jetzt vom Anwenderprogramm aus verändert werden.

Die Statusanzeige wird bei jedem Aufruf der Anweisung aktualisiert.

AG_LOCK

AG_LOCK

Beschreibung

Mit Hilfe der Anweisung AG_LOCK wird der Datenaustausch mittels FETCH bzw. WRITE über die durch den Parameter ID selektierte Verbindung gesperrt. Der Ausgang LOCKED zeigt an, ob das Sperren erfolgreich war oder nicht. War das Sperren nicht erfolgreich, muss der Auftrag in einem folgenden CPU-Zyklus erneut angestoßen werden.

Der Ausgang STATUS gibt den Zustand des CPs für diese Verbindung an.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

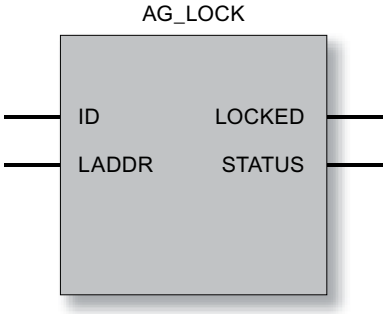

Siehe auch

Parameter bei AG_LOCK (Seite 6788)

Parameter STATUS (Seite 6789)

Parameter bei AG_LOCK

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für AG_LOCK:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ID	INPUT	INT	1,2...16 bei S7-300 1,2...64 bei S7-400	Im Parameter ID wird die Verbindungsnummer der Verbindung angegeben.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
LOCKED	OUTPUT	BOOL	0: (noch) nicht gesperrt 1: gesperrt	Anzeige des Zustands für die auf der angegebenen FETCH/WRITE-Verbindung angeforderte Zugriffssperre.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung siehe Parameter STATUS (Seite 6789)

Siehe auch

AG_LOCK (Seite 6788)

Parameter STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeigen STATUS.

Tabelle 4-347 Anzeigen AG_LOCK

STATUS	Bedeutung
7000_H	CP bearbeitet keinen Auftrag
7001_H	FETCH läuft
7002_H	WRITE läuft
8183_H	FETCH/WRITE für diese Verbindung nicht projektiert (nur bei S7-400)
8186_H	ID-Nummer nicht im erlaubten Bereich (z.B. 1...64 bei S7-400 Industrial Ethernet CPs)
$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut (bei neueren CPU-Ausgabeständen).
$80B0_H$	Baugruppe kennt den Datensatz nicht.
$80B1_H$	Die Längenangabe (im Parameter LEN) ist falsch.
$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
$80C0_H$	Datensatz kann nicht gelesen werden.
$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
$80C2_H$	Es liegt ein Auftragsstau vor.
$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
$80C4_H$	Kommunikationsfehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll.
$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_LOCK (Seite 6788)

AG_UNLOCK

AG_UNLOCK

Beschreibung

Mit Hilfe der Anweisung AG_UNLOCK geben Sie den externen Zugriff auf Anwenderspeicherbereiche der S7-CPU frei. Mit FETCH bzw. WRITE ist anschließend der Zugriff über die durch den Parameter ID selektierte Verbindung möglich.

Vorausgegangen ist eine Zugriffssperre durch AG_LOCK.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

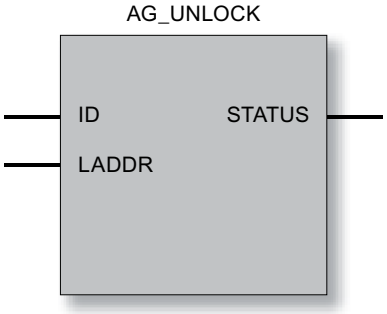

Arbeitsweise

Um die Verbindung wieder freizugeben, muss die Anweisung das LOCK-Anforderungsbit wieder zurücksetzen. Die Anweisung zeigt außerdem den aktuellen Status mit Fehlermeldungen an.

Siehe auch

Parameter STATUS (Seite 6791)

Parameter bei AG_UNLOCK (Seite 6791)

Parameter bei AG_UNLOCK

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für AG_UNLOCK:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ID	INPUT	INT	1,2...16 bei S7-300 1,2...64 bei S7-400	Im Parameter ID wird die Verbindungsnummer der Verbindung angegeben. (siehe Projektierung)
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse ausgegeben. Geben Sie diese Adresse hier an.
STATUS	OUTPUT	WORD		Statusanzeige: Zur Bedeutung siehe Parameter STATUS (Seite 6791)

Siehe auch

AG_UNLOCK (Seite 6790)

Parameter STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige STATUS.

Tabelle 4-348 Anzeigen AG_UNLOCK

STATUS	Bedeutung
7000_H	CP bearbeitet keinen Auftrag
7001_H	FETCH läuft
7002_H	WRITE läuft
8183_H	FETCH/WRITE für diese Verbindung nicht projektiert (nur bei S7-400)
8186_H	ID-Nummer nicht im erlaubten Bereich (z.B. 1...64 bei S7-400 Industrial Ethernet CPs)
$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut (bei neueren CPU-Ausgabeständen).
$80B0_H$	Baugruppe kennt den Datensatz nicht.
$80B1_H$	Die Längenangabe (im Parameter LEN) ist falsch.
$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
$80C0_H$	Datensatz kann nicht gelesen werden.
$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
$80C2_H$	Es liegt ein Auftragsstau vor.
$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
$80C4_H$	Kommunikationsfehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll.
$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_UNLOCK (Seite 6790)

Anweisungen zur Verbindungs-Diagnose

AG_CNTEX

AG_CNTEX

Bedeutung und Arbeitsweise

Mit der Anweisung AG_CNTEX besteht die Möglichkeit, Verbindungen zu diagnostizieren und Geräte per Ping-Kommando im Netz anzusprechen. Bei Bedarf können Sie über AG_CNTEX einen erneuten Verbindungsaufbau initialisieren.

Hinweis

AG_CNTRL und AG_CNTEX

Der Programmbaustein AG_CNTEX bietet Erweiterungen gegenüber dem Programmbaustein AG_CNTRL.

Sämtliche Funktionen des AG_CNTRL sind im AG_CNTEX enthalten und an der Schnittstelle im Anwenderprogramm identisch nutzbar.

Folgende Aktionen können über parametrierbare Kommandos ausgeführt werden:

- Auslesen von Verbindungsinformationen
  Sie können anhand von Status-Informationen für alle oder einzelne Verbindungen des CP entscheiden, ob ein Rücksetzen von Verbindungen sinnvoll ist.
- Rücksetzen von projektierten Verbindungen
  Sie können einzelne Verbindungen oder alle Verbindungen eines CPs zurückzusetzen.
- Aktive Verbindung abbrechen und erneut aufbauen
- Im CP konfigurierte Verbindungstypen auslesen (Erweiterung gegenüber AG_CNTRL)
- PING-Kommando senden (Erweiterung gegenüber AG_CNTRL)
  Sie können prüfen, ob ein bestimmter Teilnehmer im Netz erreichbar ist.

Die Kommandos des Programmbausteins AG_CNTEX sind nur für SEND/RECV–Verbindungen zulässig, die auf die Protokolle ISO / RFC / TCP / UDP aufsetzen.

Hinweis

Verfügbarkeit in der Bausteinbibliothek

Falls der Programmbaustein AG_CNTEX in der Bausteinbibliothek SIMATIC_NET_CP noch nicht verfügbar ist, installieren Sie die aktuelle SIMATIC NET Bausteinbibliothek; Sie finden diese unter folgender Beitrags-ID im Internet:

8797900 (http://support.automation.siemens.com/WW/view/de/8797900)

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

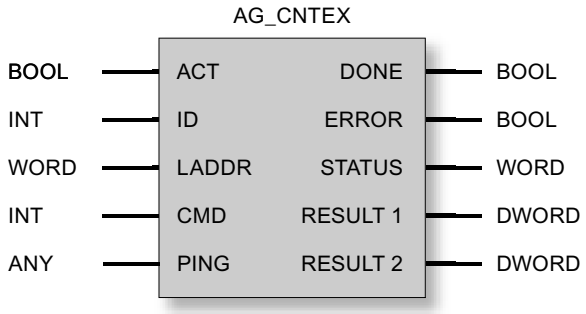

Siehe auch

Arbeitsweise der PING-Funktion (Seite 6793)

Parameter bei AG_CNTEX (Seite 6794)

Parameter DONE, ERROR, STATUS (Seite 6796)

Kommandos und Auftragsergebnisse - AG_CNTEX (Seite 6797)

Arbeitsweise der PING-Funktion

Arbeitsweise / Aufruffolge

Mit dem PING-Kommando CMD=8 veranlassen Sie den CP, 4 aufeinander folgende PING-Anforderungen im Netz an die im Auftrag angegebene IP-Adresse abzusetzen. Das PING-Echo wird vom CP innerhalb der Zeitspanne erwartet, die Sie im PING-Auftragsblock vorgeben.

Der CP registriert die Antwortzeiten und legt diese in den Parametern RESULT 1/2 ab.

Die beiden Parameter RESULT 1/2 fragen Sie mittels PING-Kommando CMD=9 ab. Sobald die 4 PING-Anforderungen beantwortet sind oder deren vorgegebene Überwachungszeit überschritten wurde, wird die Ausführung im Parameter DONE=1 bestätigt. Das PING-Ergebnis können Sie dann innerhalb von maximal 30 Sekunden abfragen; danach werden die RESULT-Einträge ungültig.

Hinweis

PING setzt projektierte Verbindung voraus

Das PING-Kommando setzt voraus, dass mindestens **eine** Verbindung für die SEND/RECEIVE-Schnittstelle (TCP, ISO-on-TCP, ISO-Transport, UDP) projektiert ist.

Mehrere PING-Anforderungen gleichzeitig

Sie können bis zu 4 PING-Anforderungen gleichzeitig an unterschiedliche IP-Adressen absetzen. Dazu müssen Sie jeweils den selben Instanz-DB für die PING-Anforderungen verwenden. Weitere PING-Anforderungen sind erst nach Abarbeitung mindestens einer der laufenden PING-Anforderungen möglich.

Wenn zu viele PING-Anforderungen gleichzeitig abgesetzt werden, dann erfolgt entsprechende Fehlermeldung (Parameter STATUS = $828A_H$).

Wann sind PING-Anforderungen abgearbeitet?

PING-Anforderungen gelten als abgearbeitet, wenn eine der folgenden Bedingungen erfüllt ist:

- Das PING-Ergebnis wurde ausgelesen;
- Das PING-Ergebnis wurde zwar nicht ausgelesen aber nach Bereitstellung des PING-Ergebnisses sind 30 Sekunden abgelaufen.

Siehe auch

AG_CNTEX (Seite 6792)

Kommandos und Auftragsergebnisse - AG_CNTEX (Seite 6797)

Parameter bei AG_CNTEX

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Funktion AG_CNTEX:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL	0, 1	Der Aufruf des FB muss mit ACT=1 erfolgen. Beim Aufruf mit ACT=0 erfolgt kein Funktionsaufruf und der Baustein wird sofort wieder verlassen.
ID	INPUT	INT	• 1, 2, .., n, oder • 0	Im Parameter ID wird die Verbindungsnummer der Verbindung angegeben. Die Verbindungsnummer ist der Projektierung zu entnehmen. n ist die maximale Anzahl Verbindungen und abhängig vom Produkt (S7-300 oder S7-400). Bei einem Aufruf, der alle Verbindungen anspricht, muss als ID 0 angegeben werden. Dies betrifft: • die Funktionen CN_STATUS_ALL (CMD 3) und CN_RESET_ALL (CMD 4) • PING-Kommando mit CMD 8 bzw CMD 9
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse ausgegeben. Geben Sie diese Adresse hier an.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CMD	INPUT	INT		Kommando an den FB AG_CNTEX
PING	INPUT	ANY		Verweist auf einen Datenblock (z.B. DB), der die Datenstruktur für das PING-Kommando enthält. Der Datenblock enthält die IP-Adresse sowie optional die Angaben zur Zeitüberwachung und zur Anzahl der in der PING-Anforderung zu übertragenden Bytes. Datenstruktur siehe unten
DONE	OUTPUT	BOOL	0: Auftrag ist noch in Bearbeitung bzw. noch nicht angestoßen 1: Auftrag ausgeführt	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Parameter DONE, ERROR, STATUS (Seite 6796) Hinweis: bei DONE=1 kann RESULT ausgewertet werden
ERROR	OUTPUT	BOOL	0: kein Fehler 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Parameter DONE, ERROR, STATUS (Seite 6796)
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Parameter DONE, ERROR, STATUS (Seite 6796)
RESULT1	OUTPUT	DWORD		Rückmeldung gemäß Kommando an den AG_CNTEX.
RESULT2	OUTPUT	DWORD		Rückmeldung Teil 2 gemäß Kommando an den AG_CNTEX.

PING-Datenblock

Der Parameter PING verweist auf einen Datenblock mit folgender Datenstruktur:

Parameter	Datentyp	Wertebereich	Hinweis
IP-Adresse	ARRAY [1..4] of Byte		
TIMEOUT	INT	1..60000 ms	Angabe optional; Default-Wert = 1000 ms
Size	INT	1..1000 Byte	Angabe optional; Default-Wert = 32 Byte

Siehe auch

AG_CNTEX (Seite 6792)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Zusätzlich sind die Kommando-Ergebnisse in den Parametern RESULT1/2 gemäß Kommandos und Auftragsergebnisse - AG_CNTEX (Seite 6797) auszuwerten.

Tabelle 4-349 Anzeigen AG_CNTEX

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Es wurde ein Auftrag (CMD) erfolgreich an den CP übertragen (z.B. RESET) bzw. es wurde ein Status erfolgreich vom CP gelesen. Die Parameter RESULT1/2 können ausgewertet werden.
0	0	0000_H	Es erfolgte noch kein Bausteinaufruf bzw. der Programmbaustein wurde mit ACT=0 aufgerufen.
0	0	8181_H	Auftrag läuft Der Bausteinaufruf ist mit den gleichen Parametern zu wiederholen, bis DONE oder ERROR gemeldet werden.
0	1	8183_H	Die Projektierung fehlt oder der Dienst im Ethernet-CP ist noch nicht gestartet.
0	1	8184_H	Systemfehler oder falscher Parametertyp. Ursache kann sein: • Datentyp des ANY-Pointers für den Parameter PING ist nicht richtig. • Der ANY-Pointer verweist auf eine ungerade Bitadresse.
0	1	8186_H	Der Parameter ID ist ungültig. Die zulässige ID ist abhängig vom gewählten Kommando.
0	1	8187_H	Der Parameter CMD ist ungültig.
0	1	8090_H	Mögliche Bedeutungen: • Eine Baugruppe mit dieser Baugruppen-Anfangsadresse ist nicht vorhanden; • Der verwendete Programmbaustein passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Programmbausteine für S7-300 und S7-400 zu verwenden); • Die Funktion wird von dieser Baugruppe nicht unterstützt.
0	1	8091_H	Die Baugruppen-Anfangsadresse liegt nicht auf Doppel-Wort-Raster.
0	1	8092_H	Die Baugruppen-Anfangsadresse ist falsch.
0	1	$80B0_H$	Die Baugruppe kennt den Datensatz nicht.
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut. Im H-System befindet sich die zugehörende CPU im Betriebszustand STOP.
0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	1	$80C4_H$	Kommunikationsfehler Der Fehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll.
0	1	8286_H	Der Wert für die Angabe "Timeout" im PING-Datenblock liegt außerhalb des gültigen Wertebereichs.
0	1	8287_H	Die im PING-DB angegebene IP-Adresse ist reserviert und daher unzulässig.

DONE	ERROR	STATUS	Bedeutung
0	1	8288_H	Die Anzeige tritt nur beim Kommando PING-Ergebnisanforderung auf. Mögliche Bedeutungen: • Die IP-Adresse wurde vom PING-Kommando nicht erreicht (beispielsweise weil der CP in den STOP-Modus gesetzt wurde, nachdem das Kommando PING-Anforderung abgesetzt wurde); • Das PING-Ergebnis wurde bereits ausgelesen; • Das PING-Ergebnis wurde nicht innerhalb der maximalen Zeitspanne von 30 Sekunden ausgelesen.
0	1	8289_H	Das Datenvolumen für die PING-Anforderung hat den zulässigen Bereich überschritten (maximal 1000 Bytes; siehe Datenstruktur für das PING-Kommando)
0	1	$828A_H$	Es befinden sich bereits 4 PiING-Anforderungen in Bearbeitung. Neue Anforderungen sind erst wieder nach Abarbeitung der bestehenden Anforderungen möglich.
0	1	$828B_H$	Für die angegebene IP-Adresse befindet sich bereits eine PING-Anforderung in Bearbeitung. Verwenden Sie das Kommando PING-Ergebnisanforderung um die laufende Bearbeitung abzuschließen.

Siehe auch

AG_CNTEX (Seite 6792)

Kommandos und Auftragsergebnisse - AG_CNTEX

Kommandos und Auswertung der Auftragsergebnisse

Entnehmen Sie den folgenden Tabellen die möglichen Kommandos und die in den Parametern RESULT1/2 auswertbaren Ergebnisse.

Hinweis

Kommandoauswertung bei älteren CP-Typen oder Firmware-Versionen

Die nachfolgend beschriebenen Kommandos werden bei aktuellen CP-Typen oder Firmware-Versionen unterstützt. Beachten Sie weitergehende Hinweise im Internet unter folgender Beitrags-ID:

http://support.automation.siemens.com/WW/view/de/33414377 (http://support.automation.siemens.com/ww/view/de/33414377)

Tabelle 4-350 Kommandos an den FC AG_CNTRL

CMD	Bedeutung		
0	NOP – no operation Der Baustein wird ohne Auftrag an den CP durchlaufen.		
	RESULT (für CMD = 0)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Ablauf ohne Fehler
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung			
1	CN_STATUS – connection status Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	$0000\ 000*_H$	Bits 0-3: Anzeigen für die Sende-Richtung (ausgeschlossene Werte: 0x2)	
			Bit 0	Verbindungstyp • 0: keine Sende- und Empfangsverbindung • 1: Verbindung für Sende- und Empfangsaufträge reserviert
			Bit 1	Status aktueller Auftrag • 0: Kein Sendeauftrag in Bearbeitung • 1: Sendeauftrag in Bearbeitung
			Bits 2+3	vorangegangener Auftrag: • 00: Keine Informationen zum vorangegangenen Sendeauftrag verfügbar • 01: vorangegangener Sendeauftrag positiv abgeschlossen • 10: vorangegangener Sendeauftrag negativ abgeschlossen

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 00*0_H	Bits 4−7: Anzeigen für die Empfangs−Richtung (ausgeschlossene Werte: 0x2)	
			Bit 4	Verbindungstyp • 0: keine Sende- und Empfangsverbindung • 1: Verbindung für Sende- und Empfangsaufträge reserviert
			Bit 5	Status aktueller Auftrag • 0: Kein Empfangsauftrag in Bearbeitung • 1: Empfangsauftrag in Bearbeitung
			Bits 6+7	vorangegangener Auftrag: • 00: Keine Informationen zum vorangegangenen Empfangsauftrag verfügbar • 01: vorangegangener Empfangsauftrag positiv abgeschlossen • 10: vorangegangener Empfangsauftrag negativ abgeschlossen

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 0*00_H	Bits 8-11: Anzeigen für FETCH/WRITE (ausgeschlossene Werte:0x3,0x7,0x8,0xB,0xF)	
			Bit 8	Verbindungstyp: • 0: keine FETCH-Verbindung • 1: Verbindung für FETCH-Aufträge reserviert
			Bit 9	Verbindungstyp: • 0: keine WRITE -Verbindung • 1: Verbindung für WRITE -Aufträge reserviert
			Bit 10	Auftragsstatus (FETCH/WRITE): • 0: Auftragsstatus OK • 1: Auftragsstatus NOT OK diese Kennung wird bei den folgenden Fällen gesetzt: – Der Auftrag wurde von der CPU negativ quittiert – Der Auftrag konnte nicht zur CPU weitergeleitet werden, weil sich die Verbindung im Zustand "LOCKED" befand. – Der Auftrag wurde abgelehnt, weil der FETCH/WRITE-Header nicht korrekt aufgebaut war.
			Bit 11	Status FETCH/WRITE-Auftrag • 0: es läuft kein Auftrag • 1: es läuft ein Auftrag vom LAN

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 *000$_H$	Bits 12-15: Allgemeine CP-Informationen (ausgeschlossene Werte:0x3,0xB)	
			Bit 12 + 13	Information zum Verbindungszustand: (nur für SEND/RECV-Verbindungen verfügbar, die auf die Protokolle ISO/RFC/TCP aufsetzen; bei UDP werden die entsprechenden internen Informationen ausgegeben) • 00: Verbindung ist abgebaut • 01: Verbindungsaufbau läuft • 10: Verbindungsabbau läuft • 11: Verbindung ist aufgebaut
			Bit 14	CP-Information: • 0: CP in STOP • 1: CP in RUN
			Bit 15	Reset-Kennung • 0: Durch den AG_CNTEX wurde noch kein Verbindungs-Reset durchgeführt bzw. die Reset-Kennung wurde zurückgenommen. • 1: Durch den AG_CNTEX wurde ein Verbindungs-Reset durchgeführt.
	RESULT1	**** 0000$_H$		Bits 16-31: Reserviert 0 – reserviert für spätere Erweiterungen
	RESULT2	0000 0000$_H$		- reserviert für spätere Erweiterungen -

CMD	Bedeutung		
2	CN_RESET – connection reset Dieses Kommando setzt die mit ID gewählte Verbindung zurück. Der CP ist über den Parameter LADDR ausgewählt. Das Rücksetzen der Verbindung bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung). Zum Zeitpunkt des Verbindungsabbruchs empfangene aber im Anwenderprogramm noch nicht entgegengenommene Daten werden gelöscht. Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.		
	RESULT (für CMD = 2)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau wurden angestoßen.
		$0000\ 0002_H$	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
3	CN_STATUS_ALL – all connections status Dieses Kommando liefert in den Parametern RESULT1/2 (insgesamt 8 Byte Sammelinformation) den Verbindungsstatus aller Verbindungen (aufgebaut/abgebaut). Der Parameter ID muss auf "0" gesetzt sein (wird auf 0 überprüft). Der CP ist über den Parameter LADDR ausgewählt. Detailinformationen einer abgebauten oder nicht projektierten Verbindung erhält man bei Bedarf über einen erneuten, auf die Verbindung gezielten Status-Aufruf mit CMD=1.		
	RESULT (für CMD = 3)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$**** ****_H$ 32 Bit mit folgender Gültigkeit: • Bei S7-400: Bit 0-31 für Verbindung 1 - 32 • Bei S7-300: Bit 0-15 für Verbindung 1 - 16	Für jeweilige Verbindung: • 0 – Verbindung abgebaut / nicht projektiert • 1 – Verbindung aufgebaut
	RESULT2	$**** ****_H$ 32 Bit mit folgender Gültigkeit: • Bei S7-400: Bit 0-31 für Verbindung 33 - 64	Für jeweilige Verbindung: • 0 – Verbindung abgebaut / nicht projektiert • 1 – Verbindung aufgebaut

CMD	Bedeutung		
4	CN_RESET_ALL – all connections reset Dieses Kommando setzt alle Verbindungen zurück. Der Parameter ID muss auf "0" gesetzt sein (wird auf 0 überprüft). Der CP ist über den Parameter LADDR ausgewählt. Das Rücksetzen der Verbindungen bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung). Zum Zeitpunkt des Verbindungsabbruchs empfangene aber im Anwenderprogramm noch nicht entgegengenommene Daten werden gelöscht. Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.		
	RESULT (für CMD = 4)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau aller Verbindungen wurden angestoßen.
	RESULT1	$0000\ 0002_H$	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
5	CN_CLEAR_RESET – Rücksetzen der Reset-Kennung Dieses Kommando setzt die Reset-Kennung (Bit 15 in RESULT1) für die mit ID gewählte Verbindung zurück. Der CP ist über den Parameter LADDR ausgewählt. Dieser Auftrag wird auch automatisch beim Lesen des Verbindungsstatus ausgeführt (CMD=1); der hier beschriebene separat absetzbare Auftrag wird daher nur in Sonderfällen benötigt.		
	RESULT (für CMD = 5)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Clear-Auftrag wurde erfolgreich an den CP übertragen.
	RESULT1	$0000\ 0002_H$	Der Clear-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
6	CN_DISCON − connection disconnect Dieses Kommando setzt die Verbindung zurück, die mit ID und LADDR ausgewählt wurde. Das Rücksetzen der Verbindung wird durch einen Verbindungsabbruch bewerkstelligt. Eventuell im Stack gespeicherte Daten gehen ohne Hinweis verloren. Im Anschluss erfolgt kein automatischer Aufbau der Verbindung. Die Verbindung kann durch den Control-Auftrag CN_STARTCON wieder aufgebaut werden. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftragsergebnis entnehmen können.		
	RESULT (für CMD = 6)		Bedeutung
	Parameter	Hex-Wert/Bereich	
	RESULT1	$0000\ 0001_H$	Der Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch wurde eingeleitet.
	RESULT1	$0000\ 0002_H$	Der Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
7	CN_STARTCON − start connection Dieses Kommando baut eine Verbindung auf, die mit ID und LADDR ausgewählt und zuvor mit dem Control-Auftrag CN_DISCON abgebrochen wurde. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftragsergebnis entnehmen können.		
	RESULT (für CMD = 7)		Bedeutung
	Parameter	Hex-Wert/Bereich	
	RESULT1	$0000\ 0001_H$	Der Auftrag zum Verbindungsaufbau wurde erfolgreich an den CP übertragen. Der Verbindungsaufbau wurde eingeleitet.
	RESULT1	$0000\ 0002_H$	Der Auftrag zum Verbindungsaufbau konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
8	PING_REQUEST - PING-Anforderung senden Dieses Kommando sendet eine PING-Anforderung an den CP. Der CP veranlasst daraufhin 4 PING-Echo-Anforderungen an die angegebene IP-Adresse.		
	RESULT (für CMD = 8)		Bedeutung
	Parameter	Hex-Wert/Bereich	
	RESULT1	$0000\ 0001_H$	Die Ping-Anforderung wurde erfolgreich dem CP zugestellt.
	RESULT2	$0000\ 0002_H$	Die Ping-Anforderung konnte dem CP nicht erfolgreich zugestellt werden, da der entsprechende Dienst auf dem CP nicht erreichbar war. Mögliche Ursache kann beispielsweise sein: CP in STOP-Modus.

CMD	Bedeutung		
9	PING_RESULT - Ping-Ergebnis abfragen Dieses Kommando sendet eine PING-Ergebnisabfrage an den CP. Der CP übergibt die Ergebnisse der 4 durchgeführten PING-Echo-Anforderungen im Parameter RESULT. Der Aufruf ist erst dann erfolgreich, wenn seitens des CP die 4 Ping-Echo-Anforderungen abgeschlossen wurden.		
	RESULT (für CMD = 9)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	**** ****$_H$	1. Wort: Antwortzeit in ms auf die 1. PING-Echo-Anforderung. 2. Wort: Antwortzeit in ms auf die 2. PING-Echo-Anforderung. Beispiel: 0005 FFFF$_H$ Echo 1 -> empfangen nach 5 ms Echo 2 -> kein Echo innerhalb der vorgegebenen Überwachungszeit.
	RESULT2	**** ****$_H$	1. Wort: Antwortzeit in ms auf die 3. PING-Echo-Anforderung. 2. Wort: Antwortzeit in ms auf die 4. PING-Echo-Anforderung. Beispiel: 0002 3456$_H$ Echo 3 -> empfangen nach 2 ms Echo 4 -> empfangen nach 13398 ms
	Wertebereiche für Datenworte in RESULT1 / RESULT 2:		
		0000$_H$	nicht verwendet
		0001$_H$ ... EA60$_H$	Antwortzeit in ms 0001$_H$ = 1 ms EA60$_H$ = 60000 ms
		EA61$_H$ ... FFFE$_H$	nicht verwendet
		FFFF$_H$	Timeout: Kein Echo innerhalb der vorgegeben Überwachungszeit.

CMD	Bedeutung		
10	CONN_TYPE - Verbindungstyp Dieses Kommando fordert den CP auf, den aktuellen Verbindungstyp für die angegebene Verbindungs-ID anzugeben.		
	RESULT (für CMD = 10)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	0000 000*$_H$	Die Anforderung liefert für die möglichen Verbindungstypen folgende Werte: 0: keine Verbindung eingerichtet 1: UDP-Verbindung 2: SMTP-Verbindung 3: TCP-Verbindung 4: Freie UDP-Verbindung 5: FTP-Verbindung 6: ISO-Transportverbindung 7: ISO-on-TCP-Verbindung
	RESULT2	0000 0000$_H$	Default

Siehe auch

Beachten Sie weitere Hinweise im FAQ unter folgender Beitrags-ID: http://support.automation.siemens.com/WW/view/de/33414377 (http://support.automation.siemens.com/ww/view/de/33414377)

Siehe auch

AG_CNTEX (Seite 6792)

Arbeitsweise der PING-Funktion (Seite 6793)

AG_CNTRL zur Verbindungsdiagnose

AG_CNTRL

Beschreibung

Mit der Anweisung AG_CNTRL besteht die Möglichkeit, Verbindungen zu diagnostizieren. Bei Bedarf können Sie über die Anweisung einen erneuten Verbindungsaufbau initialisieren.

Hinweis

AG_CNTRL und AG_CNTEX

Die Anweisung AG_CNTEX bietet Erweiterungen gegenüber der Anweisung AG_CNTRL.

Sämtliche Funktionen des AG_CNTRL sind in AG_CNTEX enthalten und an der Schnittstelle im Anwenderprogramm identisch nutzbar.

Folgende Aktionen können über parametrierbare Kommandos ausgeführt werden:

- Auslesen von Verbindungsinformationen
  Sie können anhand von Status-Informationen für alle oder einzelne Verbindungen des CP entscheiden, ob ein Rücksetzen von Verbindungen sinnvoll ist.
- Rücksetzen von projektierten Verbindungen
  Sie können einzelne Verbindungen oder alle Verbindungen eines CPs zurückzusetzen.
- Aktive Verbindung abbrechen und erneut aufbauen

Die Kommandos der Anweisung AG_CNTRL sind nur für SEND/RECV–Verbindungen zulässig, die auf die Protokolle ISO / RFC / TCP / UDP aufsetzen.

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

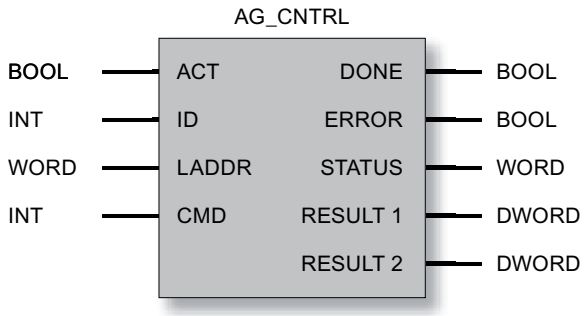

Weitere Informationen

FAQ unter Beitrags-ID 33414377 (http://support.automation.siemens.com/WW/view/de/33414377)

Siehe auch

Parameter bei AG_CNTRL (Seite 6808)
Parameter DONE, ERROR, STATUS (Seite 6809)
Kommandos und Auftragsergebnisse AG_CNTRL (Seite 6810)

Parameter bei AG_CNTRL

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für AG_CNTRL:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL	0, 1	Der Aufruf des AG_UNLOCK muss mit ACT=1 erfolgen. Beim Aufruf mit ACT=0 erfolgt kein Funktionsaufruf und die Anweisung wird sofort wieder verlassen.
ID	INPUT	INT	• 1, 2, .., n, oder • 0	Im Parameter ID wird die Verbindungsnummer der Verbindung angegeben. Die Verbindungsnummer ist der Projektierung zu entnehmen. n ist die maximale Anzahl Verbindungen und abhängig vom Produkt (S7-300 oder S7-400). Bei einem Aufruf, der alle Verbindungen anspricht, (_ALL-Funktion mit CMD 3 bzw. 4), muss als ID 0 angegeben werden.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
CMD	INPUT	INT		Kommando an den AG_CNTRL.
DONE	OUTPUT	BOOL	0: Auftrag ist noch in Bearbeitung bzw. noch nicht angestoßen 1: Auftrag ausgeführt	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6809) Hinweis: bei DONE=1 kann RESULT ausgewertet werden
ERROR	OUTPUT	BOOL	0: kein Fehler 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6809)
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe Parameter DONE, ERROR, STATUS (Seite 6809)
RESULT1	OUTPUT	DWORD		Rückmeldung gemäß Kommando an den AG_CNTRL.
RESULT2	OUTPUT	DWORD		nur bei S7-400 auszuwerten: Rückmeldung Teil 2 gemäß Kommando an den AG_CNTRL.

Siehe auch

AG_CNTRL (Seite 6807)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Zusätzlich sind die Kommando-Ergebnisse in den Parametern RESULT1/2 gemäß "Kommandos und Auftragsergebnisse AG_CNTRL (Seite 6810)" auszuwerten.

Tabelle 4-351 Anzeigen AG_CNTRL

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Es wurde ein Auftrag (CMD) erfolgreich an den CP übertragen (z.B. RESET) bzw. es wurde ein Status erfolgreich vom CP gelesen. Die Parameter RESULT1/2 können ausgewertet werden.
0	0	0000_H	Es erfolgte noch kein Aufruf der Anweisung bzw. die Anweisung wurde mit ACT=0 aufgerufen.
0	0	8181_H	Auftrag läuft Der Aufruf der Anweisung ist mit den gleichen Parametern zu wiederholen, bis DONE oder ERROR gemeldet werden.
0	1	8183_H	Die Projektierung fehlt oder der Dienst im Ethernet-CP ist noch nicht gestartet.
0	1	8186_H	Der Parameter ID ist ungültig. Die zulässige ID ist abhängig vom gewählten Kommando; siehe Parameter CMD in Kommandos und Auftragsergebnisse AG_CNTRL (Seite 6810)
0	1	8187_H	Der Parameter CMD ist ungültig.
0	1	8188_H	Sequenzfehler bei der ACT-Steuerung (Hinweis: diese Anzeige tritt in der Produktversion des CP / der Firmware nicht auf).
0	1	8189_H	Die genutzte CP-Version / Firmware unterstützt den AG_CNTRL nicht. Die Anzeige wird bei Aufruf auf einem CP3431-EX20 mit Firmware ab V1.3.9 gesetzt; bei anderen CP-Typen wird stattdessen die Anzeige 80B0H abgesetzt. Hinweis: Der AG_CNTRL in der Version V1.0 wird von den CPs ab CP 343-1 EX21/GX21 unterstützt; bei diesen tritt diese Anzeige nicht auf.
0	1	8090_H	• Eine Baugruppe mit dieser Baugruppen-Anfangsadresse ist nicht vorhanden. oder • Die verwendete Anweisung passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Anweisungen für S7-300 und S7-400 zu verwenden). oder • Die Funktion wird von dieser Baugruppe nicht unterstützt.
0	1	8091_H	Die Baugruppen-Anfangsadresse liegt nicht auf Doppel-Wort-Raster.
0	1	$80B0_H$	Die Baugruppe kennt den Datensatz nicht.
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut. Im H-System befindet sich die zugehörende CPU im Betriebszustand STOP.
0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.

DONE	ERROR	STATUS	Bedeutung
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	1	$80C4_H$	Kommunikationsfehler Der Fehler tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll.
0	1	$80D2_H$	Die Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_CNTRL (Seite 6807)

Kommandos und Auftragsergebnisse AG_CNTRL

Kommandos und Auswertung der Auftragsergebnisse

Entnehmen Sie den folgenden Tabellen die möglichen Kommandos und die in den Parametern RESULT1/2 auswertbaren Ergebnisse.

Tabelle 4-352 Kommandos an AG_CNTRL

CMD	Bedeutung		
0	NOP – no operation Der AG_CNTRL wird ohne Auftrag an den CP durchlaufen.		
	RESULT (für CMD = 0)		Bedeutung
	Parameter	Hex-Wert/Bereich	
	RESULT1	$0000\ 0001_H$	Ablauf ohne Fehler
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung			
1	CN_STATUS – connection status Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 000*$_H$	Bits 0-3: Anzeigen für die Sende-Richtung (ausgeschlossene Werte: 0x2)	
			Bit 0	Verbindungstyp • 0: keine Sende- und Empfangsverbindung • 1: Verbindung für Sende- und Empfangsaufträge reserviert
			Bit 1	Status aktueller Auftrag • 0: Kein Sendeauftrag in Bearbeitung • 1: Sendeauftrag in Bearbeitung
			Bits 2+3	vorangegangener Auftrag: • 00: Keine Informationen zum vorangegangenen Sendeauftrag verfügbar • 01: vorangegangener Sendeauftrag positiv abgeschlossen • 10: vorangegangener Sendeauftrag negativ abgeschlossen

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 00*0_H	Bits 4−7: Anzeigen für die Empfangs−Richtung (ausgeschlossene Werte: 0x2)	
			Bit 4	Verbindungstyp • 0: keine Sende- und Empfangsverbindung • 1: Verbindung für Sende- und Empfangsaufträge reserviert
			Bit 5	Status aktueller Auftrag • 0: Kein Empfangsauftrag in Bearbeitung • 1: Empfangsauftrag in Bearbeitung
			Bits 6+7	vorangegangener Auftrag: • 00: Keine Informationen zum vorangegangenen Empfangsauftrag verfügbar • 01: vorangegangener Empfangsauftrag positiv abgeschlossen • 10: vorangegangener Empfangsauftrag negativ abgeschlossen

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 0*00_H	Bits 8-11: Anzeigen für FETCH/WRITE (ausgeschlossene Werte:0x3,0x7,0x8,0xB,0xF)	
			Bit 8	Verbindungstyp: • 0: keine FETCH-Verbindung • 1: Verbindung für FETCH-Aufträge reserviert
			Bit 9	Verbindungstyp: • 0: keine WRITE -Verbindung • 1: Verbindung für WRITE -Aufträge reserviert
			Bit 10	Auftragsstatus (FETCH/WRITE): • 0: Auftragsstatus OK • 1: Auftragsstatus NOT OK diese Kennung wird bei den folgenden Fällen gesetzt: – Der Auftrag wurde von der CPU negativ quittiert – Der Auftrag konnte nicht zur CPU weitergeleitet werden, weil sich die Verbindung im Zustand "LOCKED" befand. – Der Auftrag wurde abgelehnt, weil der FETCH/WRITE-Header nicht korrekt aufgebaut war.
			Bit 11	Status FETCH/WRITE-Auftrag • 0: es läuft kein Auftrag • 1: es läuft ein Auftrag vom LAN

CMD	Bedeutung			
1	CN_STATUS – connection status **(Fortsetzung für CMD=1)** Dieses Kommando liefert den Status der mit ID ausgewählten Verbindung. Der CP ist über den Parameter LADDR ausgewählt. Sollte das Bit 15 (Reset-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht dem Auftrag CN_CLEAR_RESET - siehe CMD = 5).			
	RESULT (für CMD = 1)			**Wert/Bedeutung**
	Parameter	**Hex-Wert/Bereich**	**Bit**	
	RESULT1	0000 *000$_H$	Bits 12-15: Allgemeine CP-Informationen (ausgeschlossene Werte:0x3,0xB)	
			Bit 12 + 13	Information zum Verbindungszustand: (nur für SEND/RECV-Verbindungen verfügbar, die auf die Protokolle ISO/RFC/TCP aufsetzen; bei UDP werden die entsprechenden internen Informationen ausgegeben) • 00: Verbindung ist abgebaut • 01: Verbindungsaufbau läuft • 10: Verbindungsabbau läuft • 11: Verbindung ist aufgebaut
			Bit 14	CP-Information: • 0: CP in STOP • 1: CP in RUN
			Bit 15	Reset-Kennung • 0: Durch den AG_CNTRL wurde noch kein Verbindungs-Reset durchgeführt bzw. die Reset-Kennung wurde zurückgenommen. • 1: Durch die Control-Anweisung wurde ein Verbindungs-Reset durchgeführt.
	RESULT1	**** 0000$_H$		Bits 16-31: Reserviert 0 – reserviert für spätere Erweiterungen
	RESULT2	0000 0000$_H$		- reserviert für spätere Erweiterungen -

CMD	Bedeutung		
2	CN_RESET – connection reset Dieses Kommando setzt die mit ID gewählte Verbindung zurück. Der CP ist über den Parameter LADDR ausgewählt. Das Rücksetzen der Verbindung bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung). Zum Zeitpunkt des Verbindungsabbruchs empfangene aber im Anwenderprogramm noch nicht entgegengenommene Daten werden gelöscht. Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.		
	RESULT (für CMD = 2)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau wurden angestoßen.
		$0000\ 0002_H$	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
3	CN_STATUS_ALL – all connections status Dieses Kommando liefert in den Parametern RESULT1/2 (insgesamt 8 Byte Sammelinformation) den Verbindungsstatus aller Verbindungen (aufgebaut/abgebaut). Der Parameter ID muss auf "0" gesetzt sein (wird auf 0 überprüft). Der CP ist über den Parameter LADDR ausgewählt. Detailinformationen einer abgebauten oder nicht projektierten Verbindung erhält man bei Bedarf über einen erneuten, auf die Verbindung gezielten Status-Aufruf mit CMD=1.		
	RESULT (für CMD = 3)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$**** ****_H$	32 Bit: Verbindung 1 - 32 • 0 – Verbindung abgebaut / nicht projektiert • 1 – Verbindung aufgebaut
	RESULT2	$**** ****_H$	32 Bit: Verbindung 33 - 64 • 0 – Verbindung abgebaut / nicht projektiert • 1 – Verbindung aufgebaut

CMD	Bedeutung		
4	CN_RESET_ALL – all connections reset: Dieses Kommando setzt alle Verbindungen zurück. Der Parameter ID muss auf "0" gesetzt sein (wird auf 0 überprüft). Der CP ist über den Parameter LADDR ausgewählt. Das Rücksetzen der Verbindungen bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung). Zum Zeitpunkt des Verbindungsabbruchs empfangene aber im Anwenderprogramm noch nicht entgegengenommene Daten werden gelöscht. Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.		
	RESULT (für CMD = 4)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau aller Verbindungen wurden angestoßen.
	RESULT1	$0000\ 0002_H$	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
5	CN_CLEAR_RESET – Rücksetzen der Reset-Kennung Dieses Kommando setzt die Reset-Kennung (Bit 15 in RESULT1) für die mit ID gewählte Verbindung zurück. Der CP ist über den Parameter LADDR ausgewählt. Dieser Auftrag wird auch automatisch beim Lesen des Verbindungsstatus ausgeführt (CMD=1); der hier beschriebene separat absetzbare Auftrag wird daher nur in Sonderfällen benötigt.		
	RESULT (für CMD = 5)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Clear-Auftrag wurde erfolgreich an den CP übertragen.
	RESULT1	$0000\ 0002_H$	Der Clear-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
6	CN_DISCON − connection disconnect Dieses Kommando setzt die Verbindung zurück, die mit ID und LADDR ausgewählt wurde. Das Rücksetzen der Verbindung wird durch einen Verbindungsabbruch bewerkstelligt. Eventuell im Stack gespeicherte Daten gehen ohne Hinweis verloren. Im Anschluss erfolgt kein automatischer Aufbau der Verbindung. Die Verbindung kann durch den Control−Auftrag CN_STARTCON wieder aufgebaut werden. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftragsergebnis entnehmen können.		
	RESULT (für CMD = 6)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch wurde eingeleitet.
	RESULT1	$0000\ 0002_H$	Der Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

CMD	Bedeutung		
7	CN_STARTCON − start connection Dieses Kommando baut eine Verbindung auf, die mit ID und LADDR ausgewählt und zuvor mit dem Control −Auftrag CN_DISCON abgebrochen wurde. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftrags-ergebnis entnehmen können.		
	RESULT (für CMD = 6)		**Bedeutung**
	Parameter	**Hex-Wert/Bereich**	
	RESULT1	$0000\ 0001_H$	Der Auftrag zum Verbindungsaufbau wurde erfolgreich an den CP übertragen. Der Verbindungsaufbau wurde einge-leitet.
	RESULT1	$0000\ 0002_H$	Der Auftrag zum Verbindungsaufbau konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z. B. CP in STOP).
	RESULT2	$0000\ 0000_H$	Default

Weitere Informationen

Beachten Sie weitere Hinweise im FAQ unter folgender Beitrags-ID: 33414377 (http://support.automation.siemens.com/WW/view/de/33414377)

Siehe auch

AG_CNTRL (Seite 6807)

Anweisungen für PROFINET IO

Anweisungen für PROFINET IO

Übersicht

Für die zyklische Übertragung von Daten an der PROFINET IO-Schnittstelle stehen die nachfolgend genannten Anweisungen zur Verfügung. Je nach Verwendung des CP als PROFINET IO-Controller oder als PROFINET IO-Device in einer S7-Station unterscheidet sich die Bedeutung der Anweisungen.

Anweisung	verwendbar bei		Bedeutung
S7-300	S7-400		
PNIO_SEND	x	-	Abhängig von der Betriebsart des CP: • Beim PROFINET IO-Controller Prozess-Ausgangsdaten zu den PROFINET IO-Devices senden. • Beim PROFINET IO-Device Prozess-Eingangsdaten zum PROFINET IO-Controller weiterleiten.
PNIO_RECV	x	-	Abhängig von der Betriebsart des CP: • Beim PROFINET IO-Controller Prozess-Eingangsdaten von den PROFINET IO-Devices empfangen. • Beim PROFINET IO-Device Prozess-Ausgangsdaten vom PROFINET IO-Controller empfangen.

Für CPs im Parallelbetrieb von PROFINET IO−Controller und IO-Device stehen die Anweisungen ab Version 2.0 zur Verfügung.

Für die azyklische Datenübertragung (Datensätze, Alarminformationen) an der PROFINET IO-Schnittstelle stehen die nachfolgend genannten Anweisungen zur Verfügung. Beide Anweisungen sind nur im PROFINET IO-Controller-Betrieb nutzbar.

Anweisung	verwendbar bei		Bedeutung
S7-300	S7-400		
PNIO_RW_REC	x	-	• Datensatz lesen (von einem PROFINET IO-Device) • Datensatz schreiben (an ein PROFINET IO-Device)
PNIO_ALARM	x	-	Alarminformationen von den PROFINET IO-Devices empfangen

Anweisungen zur zyklischen Übertragung von Nutzdaten

PNIO_SEND

PNIO_SEND

Beschreibung

Die Anweisung PNIO_SEND wird für die Datenübergabe in den CP-Betriebsarten PROFINET IO-Controller oder PROFINET IO-Device verwendet.

- Betrieb als PROFINET IO-Controller
  Die Anweisung übergibt die Prozessdaten (Ausgänge) eines angegebenen Ausgabebereichs an den CP zur Weiterleitung an PROFINET IO-Devices und liefert als Statusanzeige den IO Consumer Status (IOCS) der Ausgänge von den PROFINET IO-Devices.
- Betrieb als PROFINET IO-Device
  Die Anweisung liest die vorverarbeiteten Prozess-Eingänge der CPU im PROFINET IO-Device und transferiert sie zum PROFINET IO-Controller (projektierte E-Adressen). Zusätzlich liefert die Anweisung als Statusanzeige den IO Consumer Status (IOCS) des PROFINET IO-Controllers.

Die vorverarbeiteten Prozessdaten werden in einem DB oder Merkerbereich bereitgestellt.

Für CPs im Parallelbetrieb von PROFINET IO-Controller- und IO-Device steht die Anweisung ab Version 2.0 zur Verfügung. Über den zusätzlichen Parameter MODE stellen Sie ein, für welche Betriebsart die Anweisung aufgerufen wird.

Aufrufschnittstelle (Version 2.0)

Aufrufschnittstelle in FUP-Darstellung

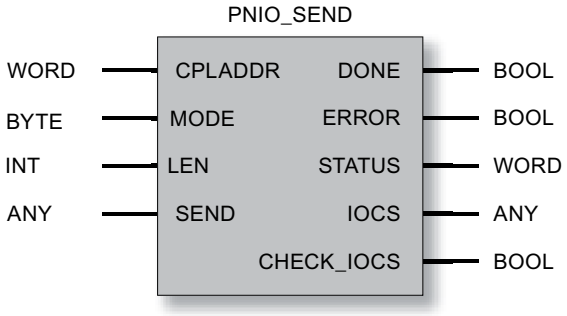

Siehe auch

Allgemeines Verhalten der Anweisungen für PROFINET IO (Seite 6832)

Datenkonsistenz (Seite 6833)

Parameter bei PNIO_SEND (Seite 6820)
Parameter DONE, ERROR, STATUS (IE) (Seite 6824)

Parameter bei PNIO_SEND

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung PNIO_SEND:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	-	Baugruppen-Anfangsadresse
MODE (Parameter ab Version 2.0)	INPUT	BYTE	Werte für X: • $0Y_H$: Statusbits werden in IOCS übertragen. • $8Y_H$: Beschränkung auf Sammelmeldung in CHECK_IOCS; keine Statusbits in IOCS. Werte für Y: • $X0_H$: – IO-Controller-Betrieb – IO-Device-Betrieb (wenn kein Parallelbetrieb) Es besteht Kompatibilität mit der Anweisung in der Version 1.0. • $X1_H$: IO-Device-Betrieb (bei Parallelbetrieb)	Angabe in der Form XY (hexadezimal): • X stellt die Übertragung von Statusangaben ein. • Y gibt die Betriebsart des CP als IO-Controller oder IO-Device an. Hinweise zur Kompatibilität: • Die Anweisung in der Version 1.0 kann weiterhin verwendet werden, wenn der CP nicht parallel als IO-Controller und als IO-Device betrieben wird. • Die Anweisung in der Version ab 2.0 verhält sich mit MODE=0 wie die Anweisung in der Version 1.0.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
SEND	IN_OUT	ANY (als VAR-TYPE ist nur BYTE zugelassen)	Die Adresse des Datenbereiches verweist alternativ auf: • Merkerbereich • Datenbausteinbereich	Angabe von Adresse und Länge **IO-Controller-Betrieb:** Die Länge sollte der projektierten Gesamtlänge der dezentralen Peripherie entsprechen, wobei Adresslücken mit übertragen werden. Die Länge kann auch kürzer als die Gesamtlänge der dezentralen Peripherie sein, beispielsweise wenn die Anweisung mehrmals in 1 OB aufgerufen wird. Er muss jedoch bei mindestens einem Aufruf die Gesamtlänge haben. **IO-Device-Betrieb:** Die Datenstruktur ergibt sich aus der Reihenfolge der Steckplätze der am PROFINET IO-Controller-Strang für dieses PROFINET IO-Device projektierten Eingangsmodule und deren Länge ohne Adresslücken. (Beachten sie hierzu weitergehende Erläuterungen oder Beispiele zu Ihrem CP im gerätespezifischen Teil B dieses Handbuches) Hinweise: • Die Anweisung beginnt die Übertragung der Daten bei Adresse 0, unabhängig davon, wie Sie die Adressen projektiert haben (unabhängig von der kleinsten projektierten Adresse). • Die Angabe eines Peripherie-Bereichs ist nicht erlaubt, da Sie zunächst den IOCS auf GOOD prüfen müssen, bevor Daten in die Peripherie übernommen werden dürfen.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
LEN	INPUT	INT	Wert > 0 Die maximale Gesamtlänge der zu übertragenden Datenbereiche entnehmen Sie bitte dem gerätespezifischen Teil B des Gerätehandbuchs unter Kapitel "Leistungsdaten". Sie kann für Controller- bzw. Device-Betrieb unterschiedlich sein.	Länge des zu übertragenden Datenbereichs in Byte. Die Übertragung der Daten beginnt zwingend mit Adresse 0, unabhängig von der Projektierung. Beachten Sie, dass die IO-Adresse "0" mit der Länge=1 berücksichtigt wird. **IO-Controller-Betrieb:** • Hier muss die größte projektierte Adresse der Devices angegeben werden. Die einzelnen Bereiche werden nicht zusammengefasst. Bei mehrmaligem Aufruf der Anweisung kann LEN auch kleiner als die größte Adresse sein. Zumindest bei einem Aufruf sollte die größte Adresse angegeben werden (vgl. Parameter "SEND"). • Die Daten werden in der Reihenfolge der logischen Adressen übertragen (wie bei PROFIBUS DP). **IO-Device-Betrieb:** • Die Daten werden in der Reihenfolge der Steckplätze so übertragen, wie die Eingangsmodule am PROFINET IO-Controller-Strang für dieses PROFINET IO-Device projektiert sind. Hinweis: Sie müssen für die Konsistenz zwischen der hier programmierten Längenangabe und der Projektierung des PROFINET IO-Controllers sorgen. Beim Device wird die gesamte Datenbereichslänge inklusive eventueller Lücken übertragen.
DONE	OUTPUT	BOOL	0: - 1: neue Daten übernommen	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
ERROR	OUTPUT	BOOL	0: -1: Fehler	Fehleranzeige
STATUS	OUTPUT	WORD	-	Statusanzeige

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CHECK_IOCS	OUTPUT	BOOL	0: alle IOCS auf GOOD 1: mindestens ein IOCS auf BAD	Hilfsbit, das anzeigt, ob es erforderlich ist, den IOCS-Statusbereich auszuwerten.
IOCS	OUTPUT	ANY (als VAR-TYPE ist nur BYTE zugelassen)	Die Adresse des Datenbereiches verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Länge: Den Maximalwert entnehmen Sie bitte dem gerätespezifischen Teil B dieses Handbuches unter Kapitel "Leistungsdaten". Sie kann für Controller- bzw. Device-Betrieb unterschiedlich sein.	Pro Byte Nutzdaten wird ein Status-Bit übertragen. Die Längenangabe hängt von der Länge im Parameter LEN ab (pro Byte ein Bit) = (Länge LEN + 7/8) Controller-Betrieb: Entsprechend dem Parameter SEND werden Adresslücken mit übertragen. Adresslücken werden mit Status GOOD übertragen. Device-Betrieb: Adresslücken werden nicht mit übertragen. Die Anweisung beginnt die Übertragung des Status für Adresse 0. Hinweis: Die Mindestlänge des ANY-Pointers ist (Länge LEN + 7/8)

Hinweis

Beachten Sie, dass alle Ausgangsparameter erst dann ausgewertet werden dürfen, wenn die Anweisung entweder DONE = 1 oder ERROR = 1 signalisiert.

Hinweis

Sie müssen davon ausgehen, dass der gelieferte IOCS Status nicht zeitsynchron zu den Daten (SEND Parameter) kommt, sondern um einen Anwenderprogramm-Zyklus verzögert. Das heißt: Anwenderdaten und IOCS sind nicht konsistent.

Siehe auch

PNIO_SEND (Seite 6819)

Parameter DONE, ERROR, STATUS (IE)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Welche System-Anweisungen genutzt werden und für die Fehlerauswertung relevant sind, können Sie über den Eigenschaftendialog der hier beschriebenen Anweisung im Register "Aufrufe" anzeigen lassen.

Tabelle 4-353 Anzeigen PNIO_SEND

DONE	ERROR	STATUS	Bedeutung
0	0	8180_H	• Datenübergabe läuft; oder • der CP ist im Betriebszustand STOP.
0	0	8181_H	Baugruppe unterstützt nicht die Version 2.0. der Anweisung Abhilfe: Version 1.0 benutzen.
1	0	0000_H	Neue Daten fehlerfrei übergeben.
0	1	8183_H	• PROFINET IO Projektierung fehlt; oder • falsche CPLADDR; oder • der CP ist im Betriebszustand STOP. oder • Beschaltung von MODE passt nicht zur Baugruppen−Projektierung oder falsche Beschaltung mit MODE > 1 Bei Device-Betrieb zusätzlich: • Die Verbindung zwischen PROFINET IO-Controller und PROFINET IO-Device ist unterbrochen, oder • PROFINET IO-Controller nicht erreichbar oder • Gesamtlängen (Projektierung und Parameter LEN) sind nicht konsistent.
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp.
0	1	8185_H	Der Parameter LEN ist größer als der Quell-Bereich SEND oder der Zielpuffer (IOCS) ist zu klein.
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).

DONE	ERROR	STATUS	Bedeutung
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F3A_H$	Zielbereich ist nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F44_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu lesenden Parameter ist gesperrt.
0	1	$8F45_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu schreibenden Parameter ist gesperrt.
0	1	$8F7F_H$	Interner Fehler. z. B. Unzulässige ANY-Referenz.
0	1	8090_H	Baugruppe mit dieser Adresse ist nicht vorhanden.
0	1	$80A0_H$	Negative Quittung beim Lesen von der Baugruppe.
0	1	$80A1_H$	Negative Quittung beim Schreiben zur Baugruppe.
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	• Die angegebene Datensatzlänge ist falsch. oder • Der CP geht in den Betriebszustand STOP.
0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)

Siehe auch

PNIO_SEND (Seite 6819)

PNIO_RECV

PNIO_RECV

Beschreibung

Die Anweisung PNIO_RECV wird für die Datenübernahme in den CP-Betriebsarten PROFINET IO-Controller oder PROFINET IO-Device verwendet.

- Betrieb als PROFINET IO-Controller
  Die Anweisung übernimmt die Prozessdaten von PROFINET IO-Devices (Eingänge des Controllers) sowie den IO Provider Status (IOPS) von den PROFINET IO-Devices in die angegeben Eingabebereiche.
- Betrieb als PROFINET IO-Device
  Die Anweisung übernimmt die vom PROFINET IO-Controller übertragenen Daten (projektierte A-Adressen) sowie den IO Provider Status (IOPS) des PROFINET IO-Controllers und schreibt sie in die für die Prozess-Ausgänge reservierten Datenbereiche der CPU im PROFINET IO-Device.

Für CPs im Parallelbetrieb von PROFINET IO−Controller− und IO-Device steht die Anweisung ab Version 2.0 zur Verfügung. Über den zusätzlichen Parameter MODE stellen Sie ein, für welche Betriebsart die Anweisung aufgerufen wird.

Aufrufschnittstelle (Version 2.0)

Aufrufschnittstelle in FUP-Darstellung

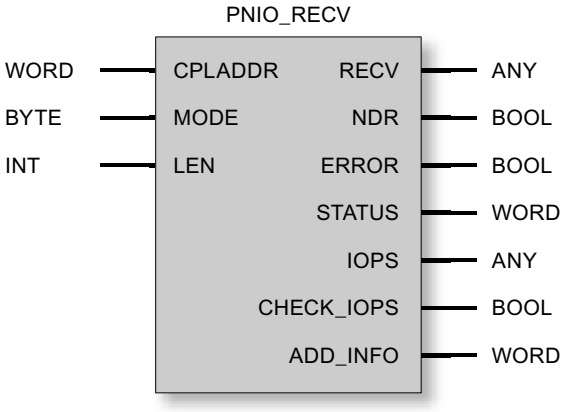

Siehe auch

Allgemeines Verhalten der Anweisungen für PROFINET IO (Seite 6832)

Datenkonsistenz (Seite 6833)

Parameter bei PNIO_RECV (Seite 6827)

Parameter NDR, ERROR, STATUS (IE) (Seite 6831)

Parameter bei PNIO_RECV

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung PNIO_RECV:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	-	Baugruppen-Anfangsadresse
MODE (Parameter ab Version 2.0)	INPUT	BYTE	Werte für X: • $0Y_H$: Statusbits werden in IOCS übertragen. • $8Y_H$: Beschränkung auf Sammelmeldung in CHECK_IOCS; keine Statusbits in IOCS. Werte für Y: • $X0_{H:}$ – IO-Controller-Betrieb – IO-Device-Betrieb (wenn kein Parallelbetrieb) Es besteht Kompatibilität mit der Anweisung in der Version 1.0 • $X1_H$: IO-Device−Betrieb (bei Parallelbetrieb)	Angabe in der Form XY (hexadezimal): • X stellt die Übertragung von Statusangaben ein. • Y gibt die Betriebsart des CP als IO-Controller oder IO-Device an. Hinweise zur Kompatibilität: • Die Anweisung in der Version 1.0 kann weiterhin verwendet werden, wenn der CP nicht parallel als IO-Controller und als IO-Device betrieben wird. • Die Anweisung in der Version ab 2.0 verhält sich mit MODE=0 wie die Anweisung in der Version 1.0.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
RECV	IN_OUT	ANY (als VAR-TYPE ist nur BYTE zugelassen)	Die Adresse des Datenbereiches verweist alternativ auf: • Merkerbereich • Datenbausteinbereich	Angabe von Adresse und Länge IO-Controller-Betrieb: Die Länge sollte der projektierten Gesamtlänge der dezentralen Peripherie entsprechen, wobei Adresslücken mit übertragen werden. Die Länge kann auch kürzer als die Gesamtlänge der dezentralen Peripherie sein, beispielsweise wenn die Anweisung mehrmals in 1 OB aufgerufen wird. Er muss jedoch bei mindestens einem Aufruf die Gesamtlänge haben. IO-Device-Betrieb: Die Datenstruktur ergibt sich aus der Reihenfolge der Steckplätze der am PROFINET IO-Controller-Strang für dieses PROFINET IO-Device projektierten Ausgangsmodule und deren Länge ohne Adresslücken. Hinweise: • Die Anweisung beginnt die Übertragung der Daten bei Adresse 0, unabhängig davon, wie Sie die Adressen projektiert haben (unabhängig von der kleinsten projektierten Adresse). • Die Angabe eines Peripherie-Bereichs ist nicht erlaubt, da Sie zunächst den IOPS auf GOOD prüfen müssen, bevor Daten in die Peripherie übernommen werden dürfen.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
LEN	INPUT	INT	Wert > 0 Die maximale Gesamtlänge der zu übertragenden Daten entnehmen Sie bitte dem Teil B des Gerätehandbuchs unter Kapitel "Leistungsdaten". Sie kann für Controller- bzw. Device-Betrieb unterschiedlich sein.	Länge des zu übertragenden Datenbereichs in Byte. Die Übertragung der Daten beginnt zwingend mit Adresse 0, unabhängig von der Projektierung. Beachten Sie, dass die IO-Adresse "0" mit der Länge=1 berücksichtigt wird. IO-Controller-Betrieb: • Hier muss die größte projektierte Adresse der Devices angegeben werden. Die einzelnen Bereiche werden nicht zusammengefasst. Bei mehrmaligem Aufruf der Anweisung kann LEN auch kleiner als die größte Adresse sein. Zumindest bei einem Aufruf sollte die größte Adresse angegeben werden (vgl. Parameter "RECV"). • Die Daten werden in der Reihenfolge der logischen Adressen übertragen (wie bei PROFIBUS DP). IO-Device-Betrieb: • Die Daten werden in der Reihenfolge der Steckplätze so übertragen, wie die Eingangsmodule am PROFINET IO-Controller-Strang für dieses PROFINET IO-Device projektiert sind. • Hinweis: Sie müssen für die Konsistenz zwischen der hier programmierten Längenangabe und der Projektierung des PROFINET IO-Controllers sorgen. Beim Device wird die gesamte Datenbereichslänge inklusive eventueller Lücken übertragen.
NDR	OUTPUT	BOOL	0: - 1: Daten übernommen	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige
STATUS	OUTPUT	WORD	-	Statusanzeige
CHECK_ IOPS	OUTPUT	BOOL	0: alle IOPS auf GOOD 1: mindestens ein IOPS auf BAD	Hilfsbit, das anzeigt, ob es erforderlich ist, den IOPS-Statusbereich auszuwerten.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
IOPS	OUTPUT	ANY (als VAR-TYPE ist nur BYTE zugelassen)	Die Adresse des Datenbereiches verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Länge: Den Maximalwert entnehmen Sie bitte dem Teil B des Gerätehandbuchs unter Kapitel "Leistungsdaten". Sie kann für Controller- bzw. Device-Betrieb unterschiedlich sein.	Pro Byte Nutzdaten wird ein Status-Bit übertragen. Die Längenangabe hängt von der Länge im Parameter RECV ab (pro Byte ein Bit) = (Länge LEN + 7/ 8) Controller-Betrieb: Entsprechend dem Parameter RECV werden Adresslücken mit übertragen. Adresslücken werden mit Status GOOD übertragen. Device-Betrieb: Adresslücken werden nicht mit übertragen. Die Anweisung beginnt die Übertragung des Status für Adresse 0. Hinweis: • Die Mindestlänge des ANY-Pointer ist (Länge LEN + 7/ 8)
ADD_INFO	OUTPUT	WORD	Zusätzliche Diagnose-Information Im Controller-Betrieb: • 0: Kein Alarm • >0: Anzahl anstehender Alarme Im Device-Betrieb ist der Parameter immer = 0.	Parameter-Erweiterung Hinweis: Der Parameter ADD_INFO wird auch dann aktualisiert, wenn im PROFINET IO-Controller keine INPUT-Adressen konfiguriert sind. In diesem Fall wird die Anweisung PNIO_RECV mit einer Länge LEN > 0 (z. B. LEN = 1 Byte) aufgerufen. Er überträgt dann eine Adresslücke von 1 Byte. Die Parameter−Erweiterung ist nutzbar für CPs ab folgendem Firmware−Stand (FW): • CP 343−1 (EX30) ab FW V2.0 • CP 343−1 Lean (CX10) ab FW V2.0 • CP 343−1 Advanced (GX30) ab FW V1.0 Bei älteren Firmwareversionen ist der Parameter reserviert.

Hinweis

Beachten Sie, dass alle Ausgangsparameter erst dann ausgewertet werden dürfen, wenn die Anweisung entweder NDR = 1 oder ERROR = 1 signalisiert.

Siehe auch

PNIO_RECV (Seite 6826)

Parameter NDR, ERROR, STATUS (IE)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus NDR, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Welche System-Anweisungen genutzt werden und für die Fehlerauswertung relevant sind, können Sie über den Eigenschaftendialog der hier beschriebenen Anweisung im Register "Aufrufe" anzeigen lassen.

Tabelle 4-354 Anzeigen PNIO_RECV

NDR	ERROR	STATUS	Bedeutung
0	0	8180_H	• Datenübernahme läuft; oder • der CP ist im Betriebszustand STOP.
0	0	8181_H	Baugruppe unterstützt nicht die Version 2.0 der Anweisung. Abhilfe: Version 1.0 benutzen.
1	0	0000_H	Neue Daten fehlerfrei übernommen.
0	1	8183_H	• PROFINET IO-Projektierung fehlt; oder • falsche CPLADDR; oder • der CP ist im Betriebszustand STOP. oder • Beschaltung von MODE passt nicht zur Baugruppen−Projektierung oder falsche Beschaltung mit MODE > 1. Bei Device-Betrieb zusätzlich: • Die Verbindung zwischen PROFINET IO-Controller und PROFINET IO-Device ist unterbrochen oder • PROFINET IO-Controller nicht erreichbar oder • Gesamtlängen (Projektierung und Parameter LEN) sind nicht konsistent
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp.
0	1	8185_H	Ziel-Puffer (RECV oder IOCS) ist zu klein.
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).

NDR	ERROR	STATUS	Bedeutung
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F3A_H$	Zielbereich ist nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F44_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu lesenden Parameter ist gesperrt.
0	1	$8F45_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu schreibenden Parameter ist gesperrt.
0	1	$8F7F_H$	Interner Fehler. z. B. Unzulässige ANY-Referenz.
0	1	8090_H	Baugruppe mit dieser Adresse ist nicht vorhanden.
0	1	$80A0_H$	Negative Quittung beim Lesen von der Baugruppe.
0	1	$80A1_H$	Negative Quittung beim Schreiben zur Baugruppe.
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	• Die angegebene Datensatzlänge ist falsch. oder • Der CP geht in den Betriebszustand STOP.
0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)

Siehe auch

PNIO_RECV (Seite 6826)

Allgemeines Verhalten der Anweisungen für PROFINET IO

IO Consumer Status (IOCS) und IO Provider Status (IOPS)

Bei beiden Kommunikationspartnern - CPU/CP einerseits und IO-Device andererseits - gibt es jeweils eine Statusinformation GOOD oder BAD zu den Daten. Diese Statusinformation wird parallel mit den Daten übertragen. Der Status des Partners, der die Daten sendet, heißt IOPS (IO Provider Status), der Status des empfangenden Partners heißt IOCS (IO Consumer Status).

Der Status IOPS und der Status IOCS sind nicht zwingend identisch. Es kann beispielsweise sein, dass sich die S7-300 CPU im Stop-Zustand befindet (Output Disable bzw. keine PROFINET IO-Anweisungen laufen). In diesem Fall übermittelt der CP als PROFINET IO-Controller den Status BAD zu den IO-Devices.

Zusammenhang zwischen Aufruf der Anweisung und IO-Daten

- Betrieb als PROFINET IO-Controller
  Der CP als PROFINET IO-Controller überwacht nicht den zyklischen Aufruf der PNIO_SEND/RECV-Anweisungen. Wenn die Anweisungn nicht aufgerufen werden, gelten die letzten übertragenen IO-Daten und IOCS/IOPS-Daten.
- Betrieb als PROFINET IO-Device
  PNIO_SEND und PNIO_RECV besitzen jeweils einen eigenen Watchdog. In Abhängigkeit von der CPU-Zykluszeit wird die Verbindung zum PROFINET IO-Controller abgebaut, falls nach der Initialisierungsphase einer der beiden Anweisungen nicht mehr aufgerufen wird.

Datenübertragung optimieren (nur bei Betrieb als PROFINET IO-Controller)

Es ist möglich, die Anweisungen mit einer Länge (Parameter LEN) aufzurufen, die kleiner ist, als die projektierte Gesamtlänge der E/A-Daten am PNIO-Strang.

Dies können Sie so ausnutzen, dass zeitkritische Daten in jedem CPU-Zyklus und unkritische Daten hingegen nicht in jedem Zyklus übertragen werden.

Beispiel:
Übertragen Sie beispielsweise in jedem Zyklus nur den ersten Bereich der Daten (zeitkritische Daten) und in jedem zweiten Zyklus die Gesamtlänge der projektierten E/A-Daten. Dazu müssen Sie die zeitkritischen Daten in der Projektierung in den unteren Bereich (ab E/A-Adresse 0) legen.

Datenkonsistenz

Es wird immer der gesamte Eingangs- bzw. Ausgangsdatenbereich des PROFINET IO-Controller komplett und damit konsistent übertragen.

- Betrieb als PROFINET IO-Controller
  Davon unabhängig haben Sie über die Längenangabe im Aufruf der Anweisung die Möglichkeit, einen kleineren als den projektierten Eingangs- bzw. Ausgangsdatenbereich konsistent zu lesen bzw. auszugeben.

Anmerkung: Beachten Sie jedoch, dass in Bezug auf die "IO-Nutzdaten" innerhalb eines PROFINET IO-Systems nur die Daten-Konsistenz innerhalb der einzelnen IO-Slots garantiert werden kann. Dies ist unabhängig davon, dass für die hier beschriebenen Anweisungen eine konsistente Datenübergabe zwischen CPU und IO-Controller gewährleistet wird.

Aufruf der Anweisung

Um die Datenkonsistenz zu sichern, dürfen Sie jedoch nur auf die IO-Daten zugreifen, wenn die Anweisung fehlerfrei beendet wurde (Output-Parameter NDR = TRUE). Zusätzlich müssen Sie prüfen, ob der Status IOCS bzw. IOPS zu den Daten = GOOD ist.

Beispiel

Im Normalfall (abhängig von der Gesamtlänge der IO-Daten) wird die Anweisung über mehrere Anwenderprogramm-Zyklen laufen, bis die Anzeige DONE/NDR = 1 meldet.

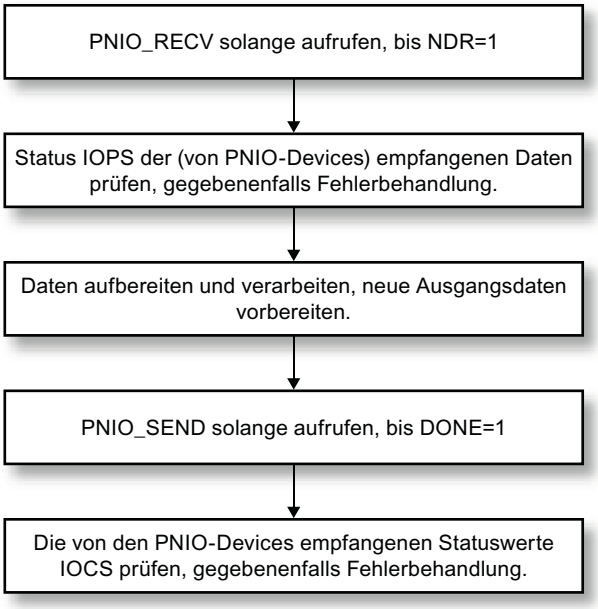

Anmerkung: Der Anwenderprogramm-Zyklus und der Zyklus des IO-Datenaustauschs zwischen PROFINET IO-Controller und PROFINET IO-Devices sind voneinander unabhängig.

Ersatzwerte

Betriebsfälle

Die Aufschaltung von Ersatzwerten wird für die beiden folgenden Betriebsfälle unterstützt:

- Ersatzwerte im Anlauf (Betriebszustandswechsel der CPU von STOP nach RUN)
- Ersatzwerte bei Störungen (Ziehen/Stecken oder Stationsausfall/-wiederkehr)

Ersatzwerte im Anlauf

Sie können die Ausgänge mit Ersatzwerten initialisieren, indem Sie im Anlauf-OB einen Merker ("Anlauf"-Merker) setzen. Im zyklischen Betrieb (OB1) werten Sie dann diesen "Anlauf"-Merker aus um gegebenenfalls die PNIO_SEND-Anweisung mit den Initialisierungswerten aufzurufen.

Ersatzwerte bei Störungen (nur bei Betrieb als PROFINET IO-Controller)

Im Fehlerfall (Device/Modul ist ausgefallen) können Sie über die Abfrage der Statusinformationen IOCS / IOPS Status ermitteln, welche Module ausgefallen sind. Daraufhin haben Sie die Möglichkeit, Ersatzwerte aufzuschalten.

PNIO_RW_REC

PNIO_RW_REC

Beschreibung

RW_REC dient im PROFINET IO-Controller-Betrieb sowohl der Funktion "Datensatz lesen" als auch der Funktion "Datensatz schreiben". RW_REC kann zu einem Zeitpunkt nur eine der beiden Funktionen ausführen. Die Funktion "Datensatz lesen" oder "Datensatz schreiben" wird über den Parameter WRITE_REC gesteuert.

Beispiel: Das Anlagenkennzeichen und das Ortskennzeichen können dem CP über die Funktion "Datensatz schreiben" mitgeteilt werden (sofern diese Parameter nicht schon in STEP 7 im Eigenschaftendialog des CP eingestellt wurden). Hierfür wird der Maintenance-Datensatz "IM1" mit dem Index AFF1H benutzt.

Details zu den unterstützten Datensätzen sowie deren Aufbau können Sie unter der folgenden Internet-Adresse abrufen:

`http://support.automation.siemens.com/WW/view/de/19289930`

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung:

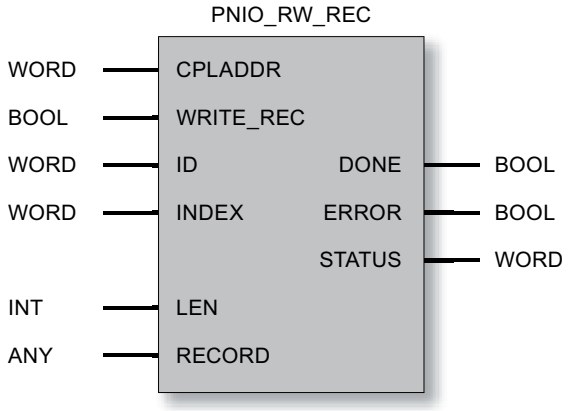

Siehe auch

Parameter DONE, ERROR, STATUS (IE) (Seite 6837)

Parameter bei PNIO_RW_REC (Seite 6836)

Parameter bei PNIO_RW_REC

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung PNIO_RW_REC:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	-	Baugruppen-Anfangsadresse
WRITE_REC	INPUT	BOOL	0: Datensatz lesen 1: Datensatz schreiben	Auftragstyp; Der Parameter darf über die Laufzeit der Anweisung nicht verändert werden.
ID	INPUT	WORD		Logische Adresse der PROFINET IO-Komponente (Baugruppe bzw. Modul). Bei einer Ausgabebaugruppe muss Bit 15 gesetzt werden. (Bsp. für Ausgangs-Adresse 5: ID:=DW#16#8005). Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
INDEX	INPUT	WORD	Siehe Herstellerinformation, welche Datensatznummern von der Baugruppe bzw. dem Modul unterstützt werden.	Datensatznummer, die der Anwender lesen oder schreiben möchte.
DONE	OUTPUT	BOOL	0: - 1: Datensatz erfolgreich übertragen	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige
STATUS	OUTPUT	WORD	0: kein Fehler anderer Wert: Fehler (siehe "Parameter DONE, ERROR, STATUS (IE) (Seite 6837)")	Statusanzeige

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
LEN	IN_OUT	INT	Die Maximallänge ist 480 Byte.	• Datensatz lesen: Reiner OUTPUT-Parameter; nach erfolgreichem Lesen wird hier die Länge des gelesenen Datensatzes angegeben; sonst 0. • Datensatz schreiben: Reiner INPUT-Parameter; Länge des zu schreibenden Datensatzes wird hier vom Anwender angegeben. Die Länge muss zur Definition des Datensatzes passen.
RECORD	IN_OUT	ANY (als VARTYPE sind BYTE, WORD und DWORD zugelassen)	Die Adresse des Datenbereichs verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Die Länge des ANY-Pointers muss größer oder gleich der Definition des Datensatzes sein.	• Datensatz lesen: Reiner OUTPUT-Parameter; nach erfolgreichem Lesen werden hier die Daten des Datensatzes abgelegt. Wenn die Länge des ANY-Pointers zu klein ist, werden soviel Daten wie möglich übertragen. • Datensatz schreiben: Reiner INPUT-Parameter; die zu schreibenden Daten des Datensatzes werden hier vom Anwender abgelegt. Die Länge des ANY-Pointers muss mindestens so groß sein, wie der Parameter LEN vorgibt.

Siehe auch

PNIO_RW_REC (Seite 6835)

Parameter DONE, ERROR, STATUS (IE)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben im Referenzhandbuch "STEP 7 - System- und Standardfunktionen für S7-300 und S7-400". Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Tabelle 4-355 Anzeigen PNIO_RW_REC

DONE	ERROR	STATUS	Bedeutung
0	0	8180_H	Datenübergabe läuft
1	0	0000_H	Datensatz erfolgreich übertragen

DONE	ERROR	STATUS	Bedeutung
0	1	8183_H	• Fehlende PROFINET IO-Controller-Projektierung, • Falsche CPLADDR oder • CP im Betriebszustand STOP
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp
0	1	8185_H	Ziel-Puffer (RECORD) ist zu klein
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz)
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz)
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. aktiven Datenbaustein
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. aktiven Datenbaustein
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer
0	1	$8F3A_H$	Zielbereich nicht geladen (DB)
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich
0	1	$8F44_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu lesenden Parameter ist gesperrt
0	1	$8F45_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu schreibenden Parameter ist gesperrt
0	1	$8F7F_H$	Interner Fehler. z. B. unzulässige ANY-Referenz
0	1	8090_H	Baugruppe mit dieser Adresse ist nicht vorhanden
0	1	$80A0_H$	Negative Quittung beim Lesen von der Baugruppe
0	1	$80A1_H$	Negative Quittung beim Schreiben zur Baugruppe
0	1	$80A3_H$	Allgemeiner PROFINET IO-Context-Management-Fehler
0	1	$80A9_H$	PROFINET IO-Device oder Baugruppe meldet einen unzulässigen Typ
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht
0	1	$80B1_H$	• Die angegebene Datensatzlänge ist falsch oder • Der CP geht in den Betriebszustand STOP
0	1	$80B2_H$	Die logische Adresse oder der projektierte Steckplatz ist nicht belegt
0	1	$80B4_H$	PROFINET IO-Device oder Baugruppe meldet einen Zugriff auf unzulässigen Bereich
0	1	$80B6_H$	PROFINET IO-Device oder Baugruppe verweigert den Zugriff
0	1	$80B8_H$	Die Baugruppe meldet einen unzulässigen Parameter
0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung
0	1	$80C2_H$	Es liegt ein Auftragsstau vor
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)

Siehe auch

PNIO_RW_REC (Seite 6835)

PNIO_Alarm

PNIO_ALARM

Beschreibung

Die Anweisung PNIO_ALARM dient der Alarmauswertung durch einen als PROFINET IO-Controller betriebenen CP 343-1 und sollte in dessen Anwenderprogramm aufgerufen werden, wenn in PNIO_RECV der Parameter ADD_INFO ungleich 0 ist. Nach vollständiger und fehlerfreier Übertragung aller OUTPUT-Parameter des PNIO_ALARM werden die empfangenen Alarme automatisch quittiert.

Die Alarme werden in der zeitlichen Reihenfolge ihrer Meldung ins Anwenderprogramm weitergegeben. Ältere, dem Anwenderprogramm noch nicht signalisierte Alarme, die durch neuere Alarme hinfällig werden, werden durch neue Alarme nicht gelöscht.

Hinweis

Solange PNIO_ALARM noch nicht aufgerufen wurde, werden die Alarme CP-intern automatisch quittiert.

Wenn PNIO_ALARM im Anwenderprogramm (wenigstens) einmal aufgerufen wurde, dann muss die Anweisung auch weiterhin aufgerufen werden, um anstehende Alarme zu quittieren. Dies ist der Fall, wenn die Anweisung PNIO_RECV im Parameter ADD_INFO einen Wert ungleich "0" meldet.

Wenn PNIO_ALARM nach ein- oder mehrmaligem Aufruf im Anwenderprogramm nicht mehr aufgerufen wird, dann werden Alarme nicht quittiert und es ist nicht sichergestellt, dass das IO-Abbild korrekt aktualisiert wird. Dies kann z.B. der Fall nach einem Stationswiederkehr-Alarm sein. Die Notwendigkeit des Aufrufens der Anweisung PNIO_ALARM kann nur durch einen Neustart des CP (Spannung AUS) zurückgesetzt werden.

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

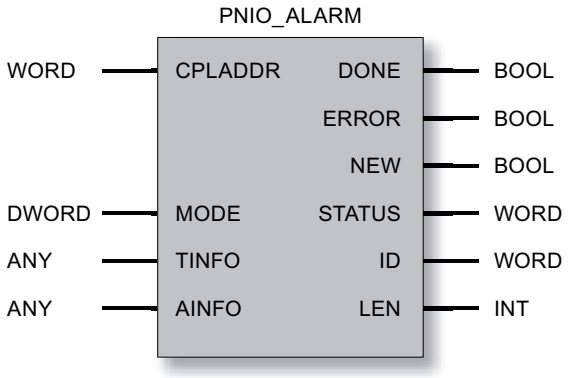

Siehe auch

Parameter DONE, NEW, ERROR, STATUS (IE) (Seite 6842)

Parameter bei PNIO_ALARM (Seite 6840)

Parameter bei PNIO_ALARM

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung PNIO_ALARM:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	-	Anfangsadresse der Fehler auslösenden Baugruppe
DONE	OUTPUT	BOOL	0: - 1: Alarminformation erfolgreich übertragen	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Bei DONE = 1 muss zusätzlich der Parameter NEW geprüft werden.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige
NEW	OUTPUT	BOOL	0: Datenübertragung läuft oder kein neuer Alarm 1: Neuen Alarm empfangen und quittiert	Bei DONE = 1 und NEW = 1 wird hier ein neu empfangener Alarm signalisiert.
STATUS	OUTPUT	WORD	0: kein Fehler anderer Wert: Fehler (siehe Tabelle 1–14)	Statusanzeige

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ID	OUTPUT	WORD		Logische Anfangsadresse der Alarm auslösenden PNIO-Komponente (Baugruppe bzw. Modul). Bei einer Ausgabebaugruppe wird Bit 15 gesetzt (Beispiel für Ausgangs-Adresse 5: ID:=DW#16#8005). Bei einer Mischbaugruppe wird die kleinere der beiden Adressen angegeben.
LEN	OUTPUT	INT		Länge der empfangenen Alarminformation (AINFO)
MODE	IN_OUT	DWORD	0	Reserviert
TINFO	IN_OUT	ANY (als VARTYPE ist BYTE, WORD und DWORD zugelassen)	Die Adresse des Datenbereichs verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Die Länge des ANY-Pointers muss >= 32 Byte sein.	(task information) Zielbereich für die Alarm-Verwaltungsinformation. Die Fehler-OB-Startinformation (OB-Header = Byte 0...19 von TINFO) wird von der CP-Firmware - soweit möglich - nachgebildet. Siehe auch 1)
AINFO	IN_OUT	ANY (als VARTYPE sind BYTE, WORD und DWORD zugelassen)	Die Adresse des Datenbereichs verweist alternativ auf: • Merkerbereich • Datenbausteinbereich Die Länge des ANY-Pointers muss größer oder gleich der max. zu erwartenden Alarmzusatzinformation sein, maximal 1432 Byte (siehe Parameter LEN)	(alarm information) Zielbereich für Kopfinformation und Alarmzusatzinformation. Wenn der ANY-Pointer AINFO zu klein ist, dann wird die Information abgeschnitten. Siehe auch 1)

1) Referenzhandbuch "STEP 7 - System- und Standardfunktionen für S7-300 und S7-400", Alarm empfangen mit der Anweisung "RALRM"

Siehe auch

PNIO_ALARM (Seite 6839)

Parameter DONE, NEW, ERROR, STATUS (IE)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, NEW, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben im Referenzhandbuch "STEP 7 - System- und Standardfunktionen für S7-300 und S7-400". Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Tabelle 4-356 Anzeigen PNIO_ALARM

DONE	NEW	ERROR	STATUS	Bedeutung
0	0	0	8180_H	Datenübergabe läuft
1	1	0	0000_H	Alarmdaten erfolgreich übertragen und Alarm quittiert
1	0	0	0000_H	Keine Alarmdaten vorhanden
0	0	1	8183_H	• Fehlende PROFINET IO-Controller-Projektierung, • Falsche CPLADDR oder • CP im Betriebszustand STOP
0	0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp
0	0	1	8185_H	Ziel-Puffer (TINFO oder AINFO) ist zu klein
0	0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz)
0	0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz)
0	0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters
0	0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters
0	0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters
0	0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters
0	0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. aktiven Datenbaustein
0	0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. aktiven Datenbaustein
0	0	1	$8F32_H$	Parameter enthält zu große DB-Nummer
0	0	1	$8F3A_H$	Zielbereich nicht geladen (DB)
0	0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich
0	0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich
0	0	1	$8F44_H$	Der Zugriff auf einen zu lesenden Parameter in der Bearbeitung der Anweisung ist gesperrt.
0	0	1	$8F45_H$	Der Zugriff auf einen zu schreibenden Parameter in der Bearbeitung der Anweisung ist gesperrt.
0	0	1	$8F7F_H$	Interner Fehler. z. B. unzulässige ANY-Referenz
0	0	1	8090_H	Baugruppe mit dieser Adresse ist nicht vorhanden
0	0	1	$80A0_H$	Negative Quittung beim Lesen von der Baugruppe

DONE	NEW	ERROR	STATUS	Bedeutung
0	0	1	$80A1_H$	Negative Quittung beim Schreiben zur Baugruppe
0	0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht
0	0	1	$80B1_H$	• Die angegebene Datensatzlänge ist falsch oder • Der CP geht in den Betriebszustand STOP
0	0	1	$80C0_H$	Der Datensatz kann nicht gelesen werden
0	0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung
0	0	1	$80C2_H$	Es liegt ein Auftragsstau vor
0	0	1	$80C3_H$	Betriebsmittel (Speicher) belegt
0	0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)

Siehe auch

PNIO_ALARM (Seite 6839)

PROFIenergy

PROFIenergy

Die PROFIenergy-Funktionen unter PROFINET dienen dem Energie-Management von Anlagen. Hierzu gehören das geplante oder spontane Abschalten einzelner Feldgeräte, Aggregate oder Anlagenteile zum Zweck der Energieeinsparung. Das Abschalten erfolgt in nichtproduktiven Zeiten oder Produktionspausen. Energie- und Diagnosedaten können aus Geräten, die in das Energiekonzept eingebunden sind und diese Funktionen unterstützen, ausgelesen werden.

PROFIenergy-Controller

Die Befehle zum Abschalten gibt die übergeordnete Steuerung aus, bei PROFINET IO der IO-Controller.

Bei SIMATIC S7-300 kann eine S7-300-CPU mit PROFIenergy-Funktion oder der CP PROFIenergy-Controller sein.

PROFIenergy-Devices

Die Befehle des PROFIenergy-Controllers werden von den IO-Devices mit PROFIenergy-Funktion verarbeitet, um angeschlossene Geräte im Feld abzuschalten.

Im Kontext der PROFIenergy-Programmbausteine wird ein IO-Device mit PROFIenergy-Funktion als PROFIenergy-Device bezeichnet. Bei SIMATIC S7-300 kann eine S7-300-CPU mit PROFIenergy-Funktion oder der CP PROFIenergy-Device sein.

I-Devices

Bei SIMATIC S7 kann ein intelligentes Device (I-Device) selbst unterlagerte PROFIenergy-Devices haben. In diesem Fall hat das I-Device auch die Funktion eines PROFIenergy-Controllers.

Energiesparmodi und PE_MODE_ID

Viele Geräte unterstützen nur die Betriebszustände "Betriebsbereit" (Spannung EIN) und "Pause" (Spannung AUS). Abgestufte Energiesparzustände mit unterschiedlichen Energiesparmodi können für Geräte, welche dies unterstützen, oder für Aggregategruppen in der Steuerung des PROFIenergy-Device festgelegt werden. Mit PROFIenergy können diese verschiedenen Zustände der Energieaufnahme den abzuschaltenden Geräten im Feld zugewiesen werden.

Die unterschiedlichen Zustände der Energieaufnahme werden "Energiesparmodi" genannt. Für jeden einzelnen Energiesparmodus wird eine definierte "PE_MODE_ID" festgelegt.

Programmierung der Eigenschaften der Energiesparmodi

Die Details der Energiesparmodi (angesprochenes Feldgerät, Pausendauer usw.) werden im Anwenderprogramm der CPU des PROFIenergy-Device programmiert.

PROFIenergy-Programmbausteine für den CP 300

Realisierung der PROFIenergy-Funktionen bei S7-300

Bei der SIMATIC S7-300 werden die PROFIenergy-Funktionen der CPU durch Programmbausteine für den IO-Controller und das IO-Device bereitgestellt.

Beachten Sie, dass S7-300-CPU und CP 300 unterschiedliche PROFIenergy-Programmbausteine verwenden.

PROFIenergy-Spezifikation

Die Funktionen der PROFIenergy-Programmbausteine für den CP 300 basieren auf der folgenden Spezifikation der PROFIBUS Nutzerorganisation e.V. (PNO):

Common Application Profile PROFIenergy, Technical Specification for PROFINET, Version 1.0, January 2010, Order No. 3.802

PROFIenergy-Programmbausteine für den CP 300

Die PROFIenergy-Programmbausteine werden vom Anwenderprogramm der CPU aufgerufen. Folgende PROFIenergy-Programmbausteine stehen für die PROFIenergy-Funktionen des CP 300 zur Verfügung:

- CP 300 als IO-Controller:
  - PE_START_END_CP
    Programmbaustein zum Einleiten und Beenden von Pausen zur Energieeinspeisung sowie für die Vorgabe definierter Energiesparmodi für das PROFIenergy-Device.
  - PE_CMD_CP
    Programmbaustein zum Einleiten und Beenden von Pausen zur Energieeinspeisung, für die Vorgabe definierter Energiesparmodi und zum Abfragen von Energiemesswerten aus dem PROFIenergy-Device.

  Die beiden Programmbausteine können alternativ eingesetzt werden. Gegenüber PE_START_END_CP hat PE_CMD_CP einen erweiterten Funktionsumfang für die Einbindung von Energiemesswerten.
  Für jedes PROFIenergy-Device muss der Programmbaustein separat aufgerufen werden.

  - PE_DS3_WRITE_ET200_CP
    Gehört nicht zu den PROFIenergy-Funktionsbausteinen, ergänzt aber die PROFIenergy-Funktionen für eine ET 200S.
    Mit dem PE_DS3_WRITE_ET200_CP werden die Einstellungen für das Schaltverhalten von bis zu 8 Steckplätzen (hier: Powermodule) der ET 200S festgelegt.
- CP 300 als IO-Device:
  - PE_I_DEV_CP
    Empfängt alle PROFIenergy-Befehle und ermöglicht dem Anwenderprogramm die Durchführung der PROFIenergy-Funktionen.
    Stellt die Antworttelegramme des IO-Device dem IO-Controller zur Verfügung.
    PE_I_DEV_CP wird zyklisch vom Anwenderprogramm des IO-Device aufgerufen.
  - Ergänzende Programmbausteine (FC 0...FC 8) für PE_I_DEV_CP:
    Diese FCs stellen die Antwortdaten für PE_I_DEV_CP bereit. Die FCs müssen im Anwenderprogramm aufgerufen und mit PE_I_DEV_CP verknüpft werden.

  Wenn das PROFIenergy-Device ein I-Device ist und selbst unterlagerte PROFIenergy-Devices hat, dann werden in der CPU des I-Device PE_START_END_CP bzw. PE_CMD_CP für die unterlagerten PROFIenergy-Devices aufgerufen.

System- und Programm-Bausteine zur Übertragung von Datensätzen

Die PROFIenergy-Befehle und Statusinformationen zwischen IO-Controller und IO-Device werden über das Lesen und Schreiben von Datensätzen ausgetauscht. Dies wird über die Programmbausteine RDREC und RWREC realisiert.

Die PROFIenergy-Datensätze sind nachfolgend bei den Antwortdaten der einzelnen Programmbausteine beschrieben.

Hinweis

Bausteinaufrufe

PE_START_END_CP, PE_CMD_CP, PE_I_DEV_CP und PE_DS3_WRITE_ET200_CP dürfen nicht gleichzeitig aufgerufen werden. Erst wenn einer dieser Programmbausteine "Kein Fehler" (VALID = 1) oder "Fehler" (ERROR = 1) gemeldet hat, darf der nächste aufgerufen werden.

Der Programmbaustein PNIO_RW_REC darf ebenfalls nicht gleichzeitig mit PE_START_END_CP, PE_CMD_CP, PE_I_DEV_CP oder PE_DS3_WRITE_ET200_CP aufgerufen werden.

PE_START_END_CP

Bedeutung und Aufruf - PE_START_END_CP

Bedeutung und Arbeitsweise

PE_START_END_CP kann alternativ zum PE_CMD_CP eingesetzt werden.

PE_START_END_CP wird im IO-Controller verwendet. Er löst im zugeordneten PROFIenergy-Device eine Energiesparpause aus oder beendet eine Pause.

Der Programmbaustein kann vorzugsweise bei IO-Controllern eingesetzt werden, an deren zugeordneten IO-Devices nur Feldgeräte angeschlossen sind, bei denen keine Energiedaten ausgelesen werden können oder sollen.

Die Energiesparmodi werden im Anwenderprogramm der CPU des IO-Device projektiert. Der tatsächlich eingenommene Energiesparmodus wird nach Ausführung von PE_START_END_CP vom IO-Device zurückgemeldet und am Parameter PE_MODE_ID ausgegeben.

Mit dem Parameter Pause_Time wird dem IO-Device die Dauer der Energiesparpause vorgegeben. Im IO-Device wird über den Programmbaustein PE_I_DEV_CP geprüft, ob die vorgegebene Pausendauer ausreichend groß ist und umgesetzt werden kann.

Ablauf

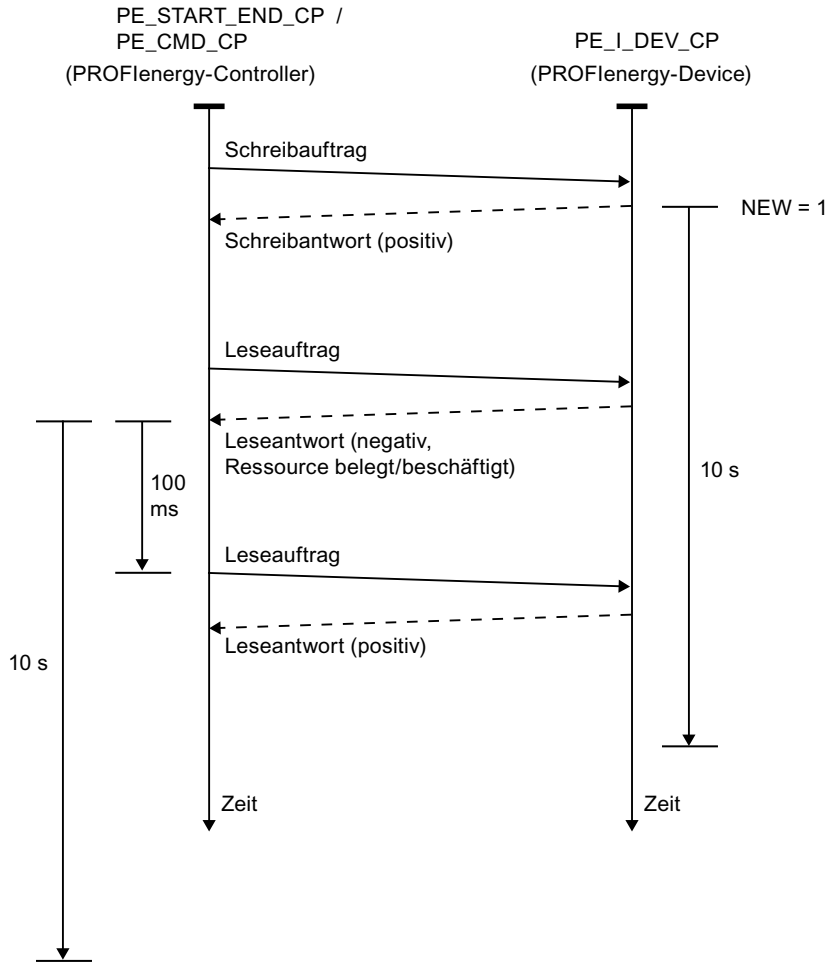

Bild 4-37 Ablaufdiagramm der Schreib-/Leseaufträge von PE_START_END_CP und PE_CMD_CP

PE_START_END_CP versendet mittels WRREC einen PROFIenergy-Befehl als Schreibauftrag an das IO-Device. Anschließend wartet PE_START_END_CP auf die Quittung vom IO-Device. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit dem Programmbaustein RDREC gelesen.

Solange keine Quittung vom IO-Device eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt.

Die Antwortdaten des IO-Device werden mit dem RDREC gelesen.

Aufrufschnittstelle in FUP-Darstellung

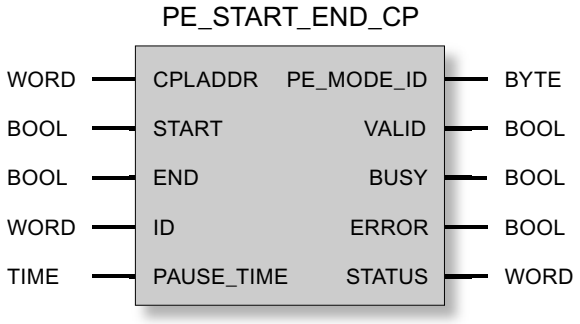

Erläuterung der Formalparameter von PE_START_END_CP

Erläuterung der Formalparameter von PE_START_END_CP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	E, A, M, D, L, Konst.	Baugruppen-Anfangsadresse des CP
START	INPUT	BOOL	• 1 = Befehl aktiv • 0 = Befehl nicht aktiv	Eine steigende Flanke aktiviert den Befehl "Start_Pause" (Energiesparpause einleiten)
END	INPUT	BOOL	• 1 = Befehl aktiv • 0 = Befehl nicht aktiv	Eine steigende Flanke aktiviert den Befehl "End_Pause" (Beenden der Energiesparpause)
ID	INPUT	WORD		Logische Adresse des Ziel-PROFIenergy-Device
PAUSE_ TIME	INPUT	TIME	T#-24D_20H_31M_23S_648MS bis T#24D_20H_31M_23S_647MS	IEC-Zeit in Schritten von 1 ms, Ganzzahl mit Vorzeichen
PE_MODE_ ID	OUTPUT	BYTE	• 00_h: Spannung AUS (Pause) • 01_h...FE_h: Projektierbar • FF_h: Betriebsbereit	ID des Energiesparmodus, der vom IO-Device nach der Ausführung des Befehls angenommen wurde.
VALID	OUTPUT	BOOL	0: - 1: Bearbeitung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
BUSY	OUTPUT	BOOL	0: Bearbeitung abgeschlossen, abgebrochen oder noch nicht begonnen 1: Bearbeitung läuft	Anzeige des Bearbeitungs-Status des Programmbausteins

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit dem Parameter STATUS siehe Anzeigen von PE_START_END_CP (Seite 6849).
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit dem Parameter ERROR siehe Anzeigen von PE_START_END_CP (Seite 6849).

Siehe auch

Antwortdaten (Seite 6855)

Anzeigen von PE_START_END_CP

Anzeigen von PE_START_END_CP

PE_START_END_CP basiert auf dem Programmbaustein PNIO_RW_REC und liefert alle Anzeigen von PNIO_RW_REC, siehe Anzeigen des Bausteins PNIO_RW_REC.

Zusätzlich werden die nachfolgenden PROFIenergy-spezifischen Anzeigen ausgegeben. Die Fehleranzeigen von STATUS gelten nur in Zusammenhang mit ERROR = 1.

Tabelle 4-357 Spezifische Anzeigen von PE_START_END_CP

STATUS	Bedeutung
Baustein-spezifische Fehler	
8080_h	Steigende Flanke gleichzeitig bei START und END
8081_h	Längenkonflikt bei CMD_PARAM und CMD_PARAM_LEN
PROFIenergy-spezifische Fehler	
$FE01_h$	Ungültige Service_Request_ID
$FE02_h$	Ungültige Request_Reference
$FE03_h$	Ungültiger CMD_MODIFIER
$FE04_h$	Ungültige Angabe zur Datenstruktur eines Befehls (Data_Structure_Identifier_RQ) im Telegramm für den schreibenden PROFIenergy-Datensatz
$FE05_h$	Ungültige Angabe zur Datenstruktur eines Befehls (Data_Structure_Identifier_RS) im Telegramm für den lesenden PROFIenergy-Datensatz
$FE06_h$	Energiesparmodus (PE_Mode_ID) nicht unterstützt
$FE07_h$	Antwort länger als max. Transferlänge
$FE08_h$	Ungültige Anzahl an Befehlen
$FE09_h$	Ungültiger Block Type (siehe Telegramm-Header)
$FE0A_h$	Ungültige Block Length (siehe Telegramm-Header)
$FE0B_h$	Ungültige Block Version (siehe Telegramm-Header)
$FE50_h$	Kein passender Energiesparmodus (PE_Mode_ID)
$FE51_h$	Wert für PAUSE_TIME nicht unterstützt
$FE52_h$	PE_Mode_ID nicht unterstützt

Details zu den Parametern der PROFIenergy-spezifischen Fehler finden Sie im Kapitel AUTOHOTSPOT.

PE_CMD_CP

Bedeutung und Aufruf - PE_CMD_CP

Bedeutung und Arbeitsweise

PE_CMD_CP kann alternativ zum PE_START_END_CP eingesetzt werden.

PE_CMD_CP wird im IO-Controller verwendet und löst im zugeordneten PROFIenergy-Device eine Energiesparpause aus oder beendet eine Pause. Zusätzlich kann PE_CMD_CP weitere Informationen und Energiemesswerte aus einem IO-Device auslesen.

Der Programmbaustein kann vorzugsweise bei IO-Controllern eingesetzt werden, an deren zugeordneten IO-Devices Feldgeräte angeschlossen sind, von denen Energiemesswerte ausgelesen werden sollen.

Ein Ablaufdiagramm der Schreib-/Leseaufträge von PE_CMD_CP finden Sie im Kapitel AUTOHOTSPOT.

Den einzelnen Befehlen, die mit dem Programmbaustein an das IO-Device übertragen werden, sind definierte "Service_Request_IDs" zugeordnet. Die Service_Request_IDs 01...05 und 16 werden am Parameter CMD zugewiesen.

Über den Parameter CMD_MODIFIER werden die beiden Befehle 04 (Query_Modes) und 16 (Query_Measurement) näher spezifiziert.

Der Parameter CMD_PARA weist einigen Befehlen über einen Any-Pointer die Werte für bestimmte Parameter zu. Der Parameter CMD_PARA_LEN legt die Länge dieser Parameter fest.

Der Parameter RESPONSE_DATA zeigt auf den Datenbereich der Antwortdaten des IO-Device.

Aufrufschnittstelle in FUP-Darstellung

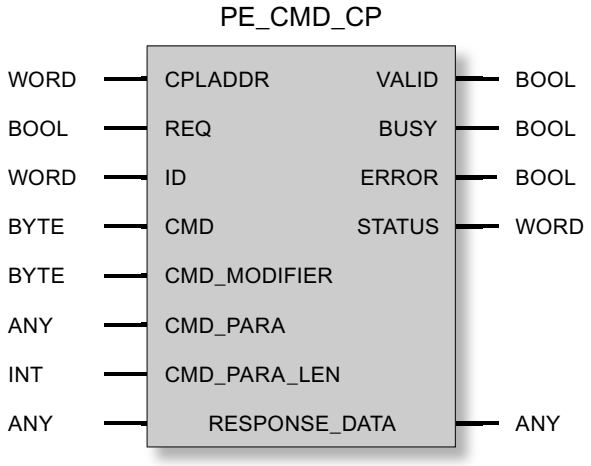

Siehe auch

Erläuterung der Formalparameter von PE_CMD_CP (Seite 6851)

Erläuterung der Formalparameter von PE_CMD_CP

Erläuterung der Formalparameter von PE_CMD_CP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	E, A, M, D, L, Konst.	Baugruppen-Anfangsadresse des CP
REQ	INPUT	BOOL		Beginnt mit steigender Flanke den Transfer der PROFIenergy-Befehle.
ID	INPUT	WORD		Logische Adresse des Ziel-PROFIenergy-Device
CMD	INPUT	BYTE	• 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identity • 16: Query_Measurement	Service_Request_ID des PROFIenergy-Befehls. Die Bedeutung der Befehle finden Sie unterhalb dieser Tabelle.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CMD_ MODIFIER	INPUT	BYTE	Für "Start_Pause": 00 Für "End_Pause": 00 Für "Query_Modes": • 01: List_Energy_Saving _Modes • 02: Get_Mode Für "PEM_Status": 00 Für "PE_Identity": 00 Für "Query_Measurement": • 01: Get_Measurement_List • 02: Get_Measurement_Values	Modifier des PROFIenergy-Befehls, Bedeutung: • Befehl "Query_Modes" – Modifier 01: Liest alle unterstützten Energiesparmodi (PE_Mode_ID). – Modifier 02 Liest die Parameter der selektierten PE_Mode_ID. • Befehl "Query_Measurement" – Modifier 01: Liest die projektierten Measurement_IDs. – Modifier 02: Liest die Messwerte der selektierten Measurement_IDs. Informationen zu den Parametern finden Sie im Kapitel AUTOHOTSPOT im Abschnitt des jeweiligen Befehls. Modifier 00 bedeutet "keine Optionen".
CMD_PARA	INPUT	ANY		Any-Pointer auf Parameter für Befehle • Für Befehl 01 Start_Pause: "Pause_Time" • Für Befehl 02 End_Pause: Irrelevant • Für Befehl 03 Query_Modes: – Für Modifier 01: Irrelevant – Für Modifier 02 Get_Mode: "PE_Mode_ID" • Für Befehl 04 PEM_Status: Irrelevant • Für Befehl 05 PE_Identity: Irrelevant • Für Befehl 16 Query_Measurement: – Für Modifier 01: Irrelevant – Für Modifier 02 Get_Measurement_Values: ANY-Pointer auf Datenstruktur mit Parametern "Count" und "Measurement_IDs" Informationen zu den Parametern finden Sie im Abschnitt AUTOHOTSPOT im Abschnitt des jeweiligen Befehls. Der komplette Datenbereich des zu schreibenden Datensatzes (Service_Data_Request) wird eingetragen. Max. Länge: 234 Byte
CMD_PA-RA_LEN	INPUT	INT		Tatsächliche Länge der Parameter in CMD_PARA. Max. Länge: 234 Byte

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
RESPONSE _DATA	INOUT	ANY		Pointer auf die Adresse der Antwortdaten des IO-Device (komplettes Telegramm inkl. Block Header) **Hinweis:** Wenn der Bereich zu klein gewählt wurde, wird nur die projektierte Byte-Anzahl gespeichert.
VALID	OUTPUT	BOOL	0: - 1: Bearbeitung erfolgreich beendet	Der Zustandsparameter des Programmbausteins zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
BUSY	OUTPUT	BOOL	0: Bearbeitung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung läuft	Anzeige des Bearbeitungs-Status des Programmbausteins
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit dem Parameter STATUS siehe AUTOHOTSPOT.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit dem Parameter ERROR siehe AUTOHOTSPOT.

Service_Request_IDs und Bedeutung der PROFIenergy-Befehle

Die PROFIenergy-Befehle mit Service_Request_ID 01...05 und 16 haben folgende Bedeutung:

- **01 = Start_Pause**
  Befehl zum Start einer Energiesparpause.
  Das IO_Device wählt den projektierten Energiesparmodus aus. Der Energiesparmodus wird dem Controller in den Antwortdaten zurückgemeldet.
- **02 = End_Pause**
  Befehl zum Beenden einer Energiesparpause
- **03 = Query_Modes**
  Fragt die im IO-Device projektierten Energiesparmodi mit allen zugehörigen Zeit- und Energie-Informationen ab.
  Über den Parameter CMD_MODIFIER werden die abgefragten Informationen detailliert:
  - **List_Energy_Saving_Modes**
    Liest alle unterstützten PROFIenergy-Modi des IO-Device.
  - **Get_Mode**
    Liest die Daten des selektierten PROFIenergy-Modus.
- **04 = PEM_Status**
  Abfrage des tatsächlich eingenommenen Energiesparmodus des Feldgeräts oder der Aggregategruppe.

- **05 = PE_Identity**
  Fragt die vom IO-Device unterstützten PROFIenergy-Dienste ab.
- **16 = Query_Measurement**
  Fragt Energiedaten des IO-Device ab.
  Über den Parameter CMD_MODIFIER werden die abgefragten Informationen detailliert:
  - **Get_Measurement_List**
    Liest alle im IO-Device projektierten Measurement_IDs.
  - **Get_Measurement_Values**
    Liest die Energiemesswerte der selektierten Measurement_IDs.

Befehle für verschiedene Geräteklassen

Die Geräte, welche sich in PROFIenergy-Konzepte einbinden lassen, können in drei Klassen unterteilt werden, die vom IO-Controller mit verschieden Befehlen angesprochen werden:

- IO-Module, Aktoren, Motorstarter
  Unterstützte Befehle:
  - Start_Pause, End_Pause
  - Query_Modes, PEM_Status, PE_Identify
- Messgeräte für elektrische Größen
  Unterstützte Befehle:
  - Query_Measurement
- Frequenzumrichter
  Unterstützte Befehle:
  - Start_Pause, End_Pause
  - Query_Modes, PEM_Status, PE_Identify
  - Query_Measurement
    Von Frequenzumrichtern können Daten elektrischer Größen, welche diese Geräte erfassen, abgefragt werden.

Anzeigen von PE_CMD_CP

Anzeigen von PE_CMD_CP

PE_CMD_CP basiert auf dem Programmbaustein PNIO_RW_REC und liefert alle Anzeigen von PNIO_RW_REC, siehe Anzeigen des Bausteins PNIO_RW_REC.

Zusätzlich werden die nachfolgenden PROFIenergy-spezifischen Anzeigen ausgegeben. Die Fehleranzeigen von STATUS gelten nur in Zusammenhang mit ERROR = 1.

Tabelle 4-358 Spezifische Anzeigen von PE_CMD_CP

STATUS	Bedeutung
Baustein-spezifische Fehler	
8081_h	Längenkonflikt bei CMD_PARAM und CMD_PARAM_LEN

STATUS	Bedeutung
PROFIenergy-spezifische Fehler	
$FE01_h$	Ungültige Service_Request_ID
$FE02_h$	Ungültige Request_Reference
$FE03_h$	Ungültiger CMD_MODIFIER
$FE04_h$	Ungültige Angabe zur Datenstruktur eines Befehls (Data_Structure_Identifier_RQ) im Telegramm für den schreibenden PROFIenergy-Datensatz
$FE05_h$	Ungültige Angabe zur Datenstruktur eines Befehls (Data_Structure_Identifier_RS) im Telegramm für den lesenden PROFIenergy-Datensatz
$FE06_h$	Energiesparmodus (PE_Mode_ID) nicht unterstützt
$FE07_h$	Antwort länger als max. Transferlänge
$FE08_h$	Ungültige Anzahl an Befehlen
$FE09_h$	Ungültiger Block Type (siehe Telegramm-Header)
$FE0A_h$	Ungültige Block Length (siehe Telegramm-Header)
$FE0B_h$	Ungültige Block Version (siehe Telegramm-Header)
$FE50_h$	Kein passender Energiesparmodus (PE_Mode_ID)
$FE51_h$	Wert für PAUSE_TIME nicht unterstützt
$FE52_h$	PE_Mode_ID nicht unterstützt

Details zu den Parametern der PROFIenergy-spezifischen Fehler finden Sie im Kapitel AUTOHOTSPOT.

Antwortdaten

Struktur der Antwortdaten

Die nachfolgenden Tabellen zeigen die Struktur des Datensatzes ($80A0_h$) der Antwortdaten von PE_START_END_CP und PE_CMD_CP.

Die nachfolgende Tabelle zeigt in der Übersicht den Aufbau des Datensatzes der zurückgelieferten Antwortdaten gemäß der PROFIenergy-Spezifikation. Die Zusammensetzung des Bereichs "Service Data Response" wird anschließend für die einzelnen PROFIenergy-Befehle beschrieben.

Tabelle 4-359 Struktur der Antwortdaten

Blockdefinitionen	Attribute	Wert	Datentyp	Beschreibung
Block Header	BlockType	0801_h	Unsigned16	
	BlockLength		Unsigned16	Telegrammlänge (ohne die Felder "BlockType" und "BlockLength")
	BlockVersionHigh	01_h	Unsigned8	
	BlockVersionLow	00_h	Unsigned8	

Blockdefinitionen	Attribute	Wert	Datentyp	Beschreibung
Response Header	Service_Request_ID	01_h...FF_h	Unsigned8	01_h: Start_Pause 02_h: End_Pause 03_h: Query_Modes 04_h: PEM_Status 05_h: PE_Identify 06_h...09_h: - Reserviert - 10_h: Query_Measurement 11_h...CF: - Reserviert - $D0_h$...FF_h: Herstellerspezifisch
	Request_Reference	01_h...FF_h	Unsigned8	Identifikationsnummer der Anfrage (in der Antwort des IO-Device gespiegelt)
Service Header Response	Status	01_h...FF_h	Unsigned8	00_h: - Reserviert - 01_h: Fertig 02_h: Fertig mit Fehler 03_h: Daten unvollständig 04_h...CF_h: - Reserviert - $D0_h$...FF_h: Abhängig von der Service_Request_ID
	Data_Structure_Identifier_RS	01_h...FF_h	Unsigned8	00_h: - Reserviert - 01_h...FF_h: Datenstruktur abhängig von der Service_Request_ID FF_h: error
Service Data Response				Antwortdaten des IO-Device Abhängig vom jeweiligen PROFIenergy-Befehl (nachfolgend beschrieben)

Bedeutung von "Service Data Request" und "Service Data Response"

Die nachfolgenden Abschnitte erläutern für die einzelnen PROFIenergy-Befehle die Parameterwerte bei den Anfragen des IO-Controllers an das IO-Device (Service Data Request) und die Struktur der Antwortdaten des IO-Device (Service Data Response).

- **Service Data Request**
  Parameterwerte bei der Anfrage des IO-Controllers
- **Service Data Response**
  Struktur der Antwortdaten des IO-Device

PROFIenergy-Befehl "Start_Pause"

- Service Data Request
  - CMD = 01
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 04
  - CMD_PARA = Any-Pointer auf Wert für "Pause_Time" (Datentyp "TIME")
    IEC-Zeit in Schritten von 1 ms, Ganzzahl mit Vorzeichen
    Wert: T#-24D_20H_31M_23S_648MS bis T#24D_20H_31M_23S_647MS
- Service Data Response

Parameter	Wert	Datentyp
PE_Mode_ID *	01_h...FF_h	Unsigned8
- Reserviert -	00_h	Unsigned8

* Identifikationsnummer des Energiesparmodus

PROFIenergy-Befehl "End_Pause"

- Service Data Request
  - CMD = 02
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevant
- Service Data Response

Parameter	Wert	Datentyp
Time_to_operate *		Unsigned32

* Erwartete Zeit, um das PROFIenergy-Device auf "Betriebsbereit" (ready_to_operate) umzuschalten

PROFIenergy-Befehl "Query_Modes" – List_Energy_Saving_Modes

- Service Data Request
  - CMD = 03
  - CMD_MODIFIER = 01
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevant
- Service Data Response

Parameter	Wert	Datentyp
Number_of_PE_Mode_IDs *	01_h	Unsigned8
PE_Mode_IDs		Unsigned8-Array of Number_of_PE_Mode_IDs (Unique ID for mode)

* Anzahl der Energiesparmodi

PROFIenergy-Befehl "Query_Modes" – Get_Mode

- Service Data Request
  - CMD = 03
  - CMD_MODIFIER = 02
  - CMD_PARA_LEN = 01
  - CMD_PARA = Any-Pointer auf Wert für PE_MODE_ID (unsigned8)
- Service Data Response

Parameter	Wert	Datentyp
PE_Mode_ID	01_h...FF_h	Unsigned8
PE_Mode_Attributes *	00_h...01_h	Unsigned8
Time_min_Pause		Unsigned32
Time_to_Pause		Unsigned32
Time_to_operate		Unsigned32
Time_min_length_of_stay		Unsigned32
Time_max_length_of_stay		Unsigned32
Mode_Power_Consumption		Float32
Energy_Consumption_to_pause		Float32
Energy_Consumption_to_operate		Float32

* Kodierung von Bit 0:
0 = Nur statische Zeit- und Energie-Messwerte verfügbar.
1 = Dynamische Zeit- und Energie-Messwerte verfügbar.
Bit 1...7: Reserviert

PROFIenergy-Befehl "PEM_Status"

- Service Data Request
  - CMD = 04
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevant
- Service Data Response

Parameter	Wert	Datentyp
PE_Mode_ID_Source *	00_h 01_h...FE_h FF_h	Unsigned8
PE_Mode_ID_Destination *	00_h 01_h...FE_h FF_h	Unsigned8
Time_to_operate		Unsigned32
Remaining_time_to_destination		Unsigned32
Mode_Power_Consumption		Float32
Energy_Consumption_to_Destination		Float32
Energy_Consumption_to_operate		Float32

* Mögliche Werte für "PE_Mode_ID_Source" und "PE_Mode_ID_Destination":
00_h: PE_Power_off (Spannungsversorgung AUS)
01_h...FE_h: Frei projektierbar
FF_h: PE_Ready_to_operate (Betriebsbereit)

PROFIenergy-Befehl "PE_Identify"

- Service Data Request
  - CMD = 05
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevant
- Service Data Response

Parameter	Wert	Datentyp
Count *	6	Unsigned8
Start_Pause **	01_h	Unsigned8
End_Pause	02_h	Unsigned8
Query_Modes	03_h	Unsigned8
PEM_Status	04_h	Unsigned8
PE_Identify	05_h	Unsigned8
Query_Measurement ***	10_h	Unsigned8

* Anzahl der unterstützten PROFIenergy-Befehle

** Service_Request_ID des ersten unterstützten PROFIenergy-Befehls

*** Service_Request_ID des letzten unterstützten PROFIenergy-Befehls

PROFIenergy-Befehl "Query_Measurement" – Get_Measurement_List

- Service Data Request
  - CMD = 16
  - CMD_MODIFIER = 01
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevant
- Service Data Response

Parameter	Wert	Datentyp
Count *		Unsigned8
- Reserviert -		Unsigned8
Measurement_ID **		Unsigned16
Accuracy_Domain [1]		Unsigned8
Accuracy_Class [2]		Unsigned8
Range [3]		Float32
…		
Measurement_ID ***		Unsigned16
Accuracy_Domain [1]		Unsigned8
Accuracy_Class [2]		Unsigned8
Range [3]		Float32

* Anzahl der Measurement_IDs

** Erste unterstützte Measurement_ID

*** Letzte unterstützte Measurement_ID

[1] Genauigkeitsdomäne (Bereich 1...4):
0 = Reserviert
1 = Prozent des Messbereichs
2 = Prozent des aktuellen Messwerts
3 = Genauigkeit gemäß IEC 61557-12
4 = Genauigkeit gemäß EN 50470-3 Kapitel 8

[2] Genauigkeitsklasse (Bereich 1...15):
0 = Reserviert
1 (0,01%) ... 15 (>20%)

[3] Messbereich, falls Accuracy_Domain = 1; andernfalls undefiniert

PROFIenergy-Befehl "Query_Measurement" – Get_Measurement_Values

- Service Data Request
  - CMD = 16
  - CMD_MODIFIER = 02
  - CMD_PARA_LEN = Länge der Datenstruktur in Byte
  - CMD_PARA = Any-Pointer auf Datenstruktur mit folgendem Aufbau:

Parameter	Wert	Datentyp
Count *		Unsigned8
- Reserviert -		Unsigned8
Measurement_ID **		Unsigned16
…		
Measurement_ID ***		Unsigned16

* Anzahl der Measurement_IDs

** Erster abgefragter Messwert

*** Letzter abgefragter Messwert

- Service Data Response

Parameter	Wert	Datentype
Count *		Unsigned8
- Reserviert -		Unsigned8
Length_of_Structure	0002_h...$FFFF_h$	Unsigned16
Measurement_Data_Structure_ID	1 = Einfacher Wert	Unsigned8
Measurement_ID **	00_h...FF_h	Unsigned16
Status_of_Measurement_Value	1 = Gültig 2 = Nicht verfüg-bar 3 = Zeitweise nicht verfügbar	Unsigned8
Transmission_Data_Type		Float32
End_of_demand		Unsigned32 oder Unsigned16
Length_of_Structure		Unsigned16
Measurement_Data_Structure_ID		Unsigned8
Measurement_ID ***		Unsigned16
Status_of_Measurement_Value		Unsigned8
Transmission_Data_Type		Float32
End_of_demand		Unsigned32 oder Unsigned16

* Anzahl der Measurement_IDs

** Erster abgefragter Messwert

*** Letzter abgefragter Messwert

Siehe auch

Erläuterung der Formalparameter von PE_START_END_CP (Seite 6848)

PE_I_DEV_CP

Bedeutung und Aufruf - PE_I_DEV_CP

Bedeutung und Arbeitsweise

Der Programmbaustein PE_I_DEV_CP wird im PROFIenergy-Device verwendet und wickelt dort die PROFIenergy-Befehle des IO-Controllers ab. Die vom IO-Controller gesendeten PROFIenergy-Datensätze ($80A0_h$) werden von der CP-Firmware an PE_I_DEV_CP übermittelt. Die PROFIenergy-Daten des IO-Device werden von PE_I_DEV_CP über den PROFIenergy-Datensatz ($80A0_h$) wiederum dem IO-Controller als Antwort zur Verfügung gestellt.

Die Antwortdaten von PE_I_DEV_CP werden von den ergänzenden Funktionen FC 0 bis FC 8 generiert, siehe Kapitel AUTOHOTSPOT.

Aufrufschnittstelle in FUP-Darstellung

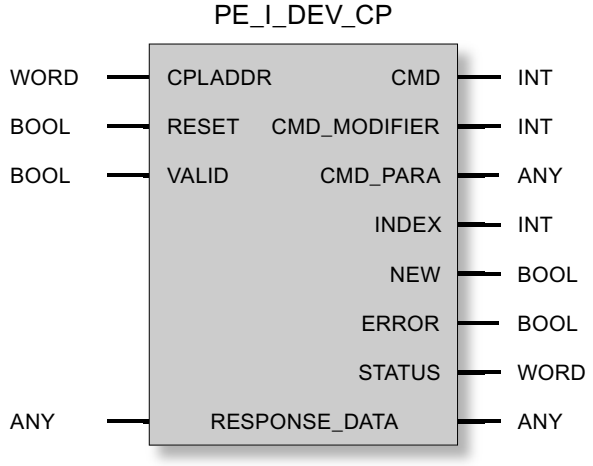

Erläuterung der Formalparameter von PE_I_DEV_CP

Erläuterung der Formalparameter von PE_I_DEV_CP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	E, A, M, D, L, Konst.	Baugruppen-Anfangsadresse des CP
RESET	INPUT	BOOL		Setzt die Bearbeitung des Programmbausteins zurück. NEW wird auf 0 gesetzt.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
VALID	INPUT	BOOL		Wenn die Antwortdaten in den betreffenden Speicherbereich des IO-Device geschrieben wurden, muss durch das Anwenderprogramm VALID = 1 gesetzt werden. Anschließend stellt der Programmbaustein die Daten dem IO-Controller zur Verfügung. NEW wird auf 0 gesetzt.
CMD	OUTPUT	INT	• 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identity • 16: Query_Measurement	Service-ID des PROFIenergy-Befehls
CMD_ MODIFIER	OUTPUT	INT	• Modifier für Start_Pause: 00 • Modifier für End_Pause: 00 • Query_Modes, Modifier: – 01 (List_Energy_ Saving_Modes) – 02 (Get_Mode) • Modifier für PEM_Status: 00 • Modifier für PE_Identity: 00 • Query_Measurement, Modifier: – 01 (Get_Measurement _List) – 02 (Get_Measurement _Values)	Modifier der PROFIenergy-Befehle Bedeutung der Modifier für Befehle: • Befehl "Query_Modes", Modifier: – 01 (List_Energy_Saving_Modes): Liest alle unterstützten PROFIenergy-Modi – 02 (Get_Mode): Liest die Daten des selektierten PROFIenergy-Modus • Befehl "Query_Measurement", Modifier: – 01 (Get_Measurement_List): Liest alle projektierten Measurement_IDs. – 02 (Get_Measurement_Values): Liest die Messwerte der selektierten Measurement_IDs.
CMD_PARA	OUTPUT	ANY		Any-Pointer auf Parameter für die folgenden Befehls-Modifier (vgl. Parameter CMD_MODIFIER): • Für "Get_Mode": PE_Mode_ID (ID des Energiesparmodus)Länge = 1 • Für "Get_Measurement_Values":Messwerte der Measurement_IDs Länge = max. 236 Byte (komplettes Telegramm des Controller-Befehls ohne Header)
INDEX	OUTPUT	INT		Nummer des PROFIenergy-Datensatzes ($80A0_h$)

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
NEW	OUTPUT	BOOL	0: Bearbeitung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung läuft	Anzeigen des Bearbeitungs-Status des Programmbausteins
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit dem Parameter STATUS siehe Anzeigen von PE_I_DEV_CP (Seite 6864).
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit dem Parameter ERROR siehe Anzeigen von PE_I_DEV_CP (Seite 6864).
RESPONSE _DATA	INOUT	ANY	Siehe "Antwortdaten" des Programmbausteins	Zeiger auf den Datenbereich der Antwort des IO-Device (komplettes Antwort-Telegramm inklusive Header). Der Datenbereich muss mit dem Datenbereich der ergänzenden Programmbausteine FC 0 - FC 8 (Parameter "DATA_ERRORRSP") übereinstimmen. Empfohlene Größe: Mind. 244 Byte. Wenn der Datenbereich zu klein ist, werden nur die Daten der projektierten Bytes übertragen.

Anzeigen von PE_I_DEV_CP

Anzeigen von PE_I_DEV_CP

PE_I_DEV_CP basiert auf dem Programmbaustein PNIO_RW_REC und liefert alle Anzeigen von PNIO_RW_REC, siehe Anzeigen des Bausteins PNIO_RW_REC.

Ergänzende Programmbausteine für PE_I_DEV_CP

Übersicht der FCs

Funktion

Die ergänzenden Programmbausteine FC 0 bis FC 8 unterstützen das Bereitstellen der Antwortdaten, welche PE_I_DEV_CP dem Controller zur Verfügung stellt:

- Für die Antwortdaten jedes PROFIenergy-Befehls gibt es einen separaten Programmbaustein (FC 1 - FC 8).
- FC 0 generiert eine gemeinsame negative Antwort für alle PROFIenergy-Befehle.

Die FCs werden im Anwenderprogramm aufgerufen. In STEP 7 V5.5 werden sie in der Standard-Bibliothek im Ordner "PROFIenergy" zur Verfügung gestellt.

Die FCs besitzen einige gemeinsame Parameter und jeweils individuelle Parameter. Einige der gemeinsamen Parameter der FCs werden mit Parametern von PE_I_DEV_CP verschaltet. An manchen der individuellen Eingangsparametern der FCs werden die Antwortdaten anwenderseitig als Klartext eingegeben bzw. im Speicherbereich des IO-Device hinterlegt.

Übersicht der FCs

Folgende ergänzende Programmbausteine werden zur Verfügung gestellt:

Tabelle 4-360 Übersicht der ergänzenden FCs

Nummer	Name
FC 0	PE_ERROR_RSP
FC 1	PE_START_RSP
FC 2	PE_END_RSP
FC 3	PE_LIST_MODES_RSP
FC 4	PE_GET_MODE_RSP
FC 5	PE_PEM_STATUS_RSP
FC 6	PE_IDENTIFY_RSP
FC 7	PE_MEASUREMENT_LIST_RSP
FC 8	PE_MEASUREMENT_VALUE_RSP

Verschaltung der FCs mit PE_I_DEV_CP

Verschaltung der FCs mit dem Programmbaustein PE_I_DEV_CP

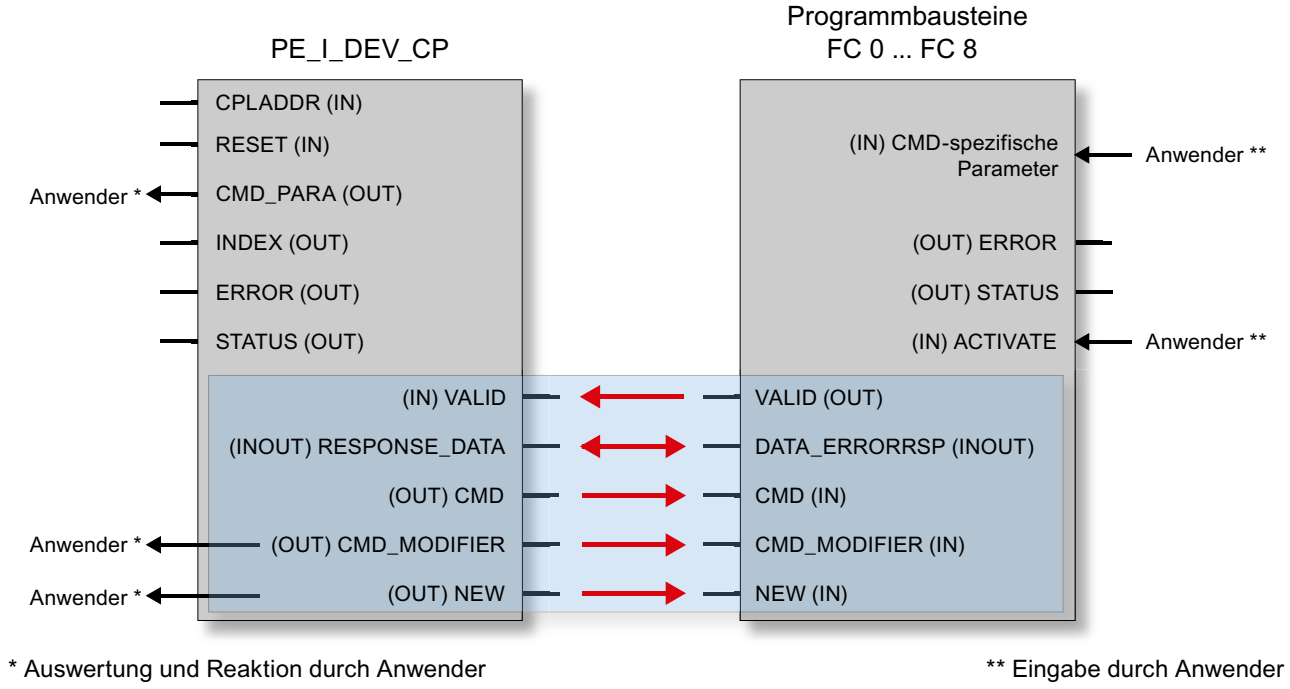

* Auswertung und Reaktion durch Anwender

** Eingabe durch Anwender

Bild 4-38 Verschaltung der FCs mit PE_I_DEV_CP

Hinweis

Verschaltung der Programmbausteine ist zwingend erforderlich

PE_I_DEV_CP muss an den hellblau hinterlegten Parametern, die durch rote Pfeile den entsprechenden Parametern der FCs zugeordnet sind, mit FC 0...FC 8 verschaltet werden.

Gemeinsame Parameter der FCs

Gemeinsame Parameter der ergänzenden Programmbausteine FC 0 - FC 8

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Eingangsparameter				
ACTIVATE	INPUT	BOOL		Veranlasst den Baustein mit steigender Flanke, die Eingangsparameter in den Datenbereich "DATA_ERRORRSP" zu kopieren. Wird danach vom Baustein zurückgesetzt. Muss innerhalb von 10 Sekunden, nachdem an PE_I_DEV_NEW eine positive Flanke erkannt wurde, vom Anwender gesetzt werden.
PE_I_DEV_NEW	INPUT	BOOL		Muss mit dem Ausgangsparameter NEW von PE_I_DEV_CP verschaltet werden. Der Baustein wird nur bearbeitet, wenn 1 ansteht.
CMD	INPUT	INT		Muss mit dem Ausgangsparameter CMD von PE_I_DEV_CP verschaltet werden.
CMD_MODIFIER	INPUT	INT		Muss mit dem Ausgangsparameter CMD_MODIFIER von PE_I_DEV_CP verschaltet werden.
Ausgangsparameter				
DATA_ERRORRSP	OUTPUT	ANY		Zeiger auf den Datenbereich, in dem die Antwortdaten abgelegt werden (komplettes Antwort-Telegramm inklusive Header). Muss mit dem Ausgangsparameter RESPONSE_DATA von PE_I_DEV_CP verschaltet werden. Empfohlene Größe: Mind. 244 Byte.
VALID	OUTPUT	BOOL	0: - 1: Kein Fehler	Wird vom Baustein gesetzt. Muss mit dem Eingangsparameter VALID von PE_I_DEV_CP verschaltet werden.
ERROR	OUTPUT	BOOL	0: Kein Fehler 1: Fehler	Fehleranzeige
STATUS	OUTPUT	WORD	0: Kein Fehler	Statusanzeige $80B1_h$: Fehler bei der ANY-Angabe (z. B. falscher Bereich)

Individuelle Parameter der FCs

Individuelle Parameter von FC 0 bis FC 8

Nachfolgend finden Sie die Beschreibung der individuellen Parameter der FCs.

PE_ERROR_RSP

Generiert eine negative Antwort, falls der anfordernde PROFIenergy-Befehl generell oder temporär nicht unterstützt wird. Die negative Antwort ist unabhängig vom anfordernden Befehl.

Tabelle 4-361 Individuelle Parameter von FC 0 PE_ERROR_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ERROR_CODE	INPUT	BYTE		Fehlernummer

PE_START_RSP

Leitet eine Energiesparpause ein. Generiert die Antwort auf den Befehl "Start_Pause". Gibt zurück, welchen Energiesparmodus das Gerät einnimmt.

Tabelle 4-362 Individuelle Parameter von FC 1 PE_START_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
PE_Mode_ID	INPUT	BYTE		ID des Energiesparmodus, den das Gerät oder die Aggregategruppe einnimmt.

Rückmeldung der PE_Mode_ID des Energiesparmodus, den das Feldgerät oder die Aggregategruppe eingenommen hat.

PE_END_RSP

Generiert die Antwort auf den Befehl "End_Pause".

Tabelle 4-363 Individuelle Parameter von FC 2 PE_END_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Time_to_Operate	INPUT	DWORD		Zeit, die benötigt wird, um vom aktuellen Energiesparmodus nach "Betriebsbereit" (Ready_to_Operate) zu wechseln.

PE_LIST_MODES_RSP

Generiert die Antwort auf den Befehl "Query_Modes" > Modifier "List_Modes" (Liste der unterstützen Energiesparmodi).

Die IDs der Energiesparmodi sind im Anwenderprogramm festzulegen.

Tabelle 4-364 Individuelle Parameter von FC 3 PE_LIST_MODES_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Number_of_PE_Mode_IDs	INPUT	BYTE		Anzahl der unterstützen Energiesparmodi
PE_Mode_ID	INPUT	ANY	• 00_h • 01_h...FE_h • FF_h	Zeiger auf den Bereich, in dem die Energiesparmodi hinterlegt sind. Als Anwender müssen Sie hier die IDs der Energiesparmodi hinterlegen. Eine Energiesparmodus-ID wird im Format Unsigned8 projektiert. Zulässiger Bereich: 1 bis 254 Byte.

Wenn die Geräte einer Gruppe bei verschieden langen Pausen unterschiedlich reagieren sollen, dann können Sie hierfür unterschiedliche Energiesparmodi (PE_Mode) einrichten. Den unterschiedlichen Energiesparmodi weisen Sie eine unterschiedliche PE_Mode_ID zu.

Mögliche Werte für "PE_Mode_ID":

- 00_h: PE_Power_off (Spannungsversorgung AUS)
- 01_h...FE_h: Frei projektierbar bzw. herstellerspezifisch
- FF_h: PE_Ready_to_operate (Betriebsbereit)

PE_GET_MODE_RSP

Generiert die Antwort auf den Befehl "Query_Modes" > Modifier "Get_Mode".

Tabelle 4-365 Individuelle Parameter von FC 4 PE_GET_MODE_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
PE_Mode_ID	INPUT	BYTE		Aktuell verwendete Energiesparmodus-ID
Time_Min_Pause *	INPUT	Unsigned32		Min. Pausendauer für diesen PE-Energiesparmodus. Es ist die Summe der drei Parameter: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay
Time_to_Pause *	INPUT	Unsigned32		Zeitdauer von der START-Flanke bis zum Erreichen des angeforderten Energiesparmodus
Time_to_operate *	INPUT	Unsigned32		Max. Einschaltdauer bis PE_ready_to_operate Time_to_operate kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer MAX-Wert sein oder vom PE-Device dynamisch berechnet werden.
Time_min_length_of_stay *	INPUT	Unsigned32		Minimale Zeitdauer, die das PE-Device in diesem PE_Mode verbleiben muss.
Time_max_length_of_stay *	INPUT	Unsigned32		Maximale Zeitdauer, die das PE-Device in diesem PE_Mode verbleiben kann.
Mode_Power _Consumption **	INPUT	Float32		Energieverbrauch im aktuellen PE_Mode [kW]

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Energy_Consumption _to_pause **	INPUT	Float32		Energieverbrauch von PE_ready_to_operate bis zum aktuellen PE_Mode [kWh]
Energy_Consumption _to_operate **	INPUT	Float32		Energieverbrauch vom aktuellen PE_Mode bis PE_ready_to_operate [kWh]

* Das PROFIenergy Profile spezifiziert kein ungültiges Zeitformat.
Wenn die Zeitdauer unbegrenzt ist, kann der max. Wert $FFFFFFFF_h$ angegeben werden.
Wenn die Zeitdauer null ist, kann 00_h verwendet werden.

** Wenn ein Energieverbrauchswert nicht definiert ist, kann 0,0 (Float32) angegeben werden.

PE_PEM_STATUS_RSP

Generiert die Antwort auf den Befehl "PEM_STATUS".

Tabelle 4-366 Individuelle Parameter von FC 5 PE_PEM_STATUS_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
PE_Mode_ID_Source	INPUT	BYTE	• 00_h • 01_h...FE_h • FF_h	ID des tatsächlich eingenommenen Energiesparmodus
PE_Mode_ID_Destination	INPUT	BYTE	• 00_h • 01_h...FE_h • FF_h	ID des vom Controller vorgegebenen Energiesparmodus
Time_to_operate *	INPUT	Unsigned32		Max. Einschaltdauer bis PE_ready_to_operate Time_to_operate kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer MAX-Wert sein oder vom PE-Device dynamisch berechnet werden.
Remaining_time_to _destination *	INPUT	Unsigned32		Optional: verbleibende Zeit bis zum angeforderten PE_Mode. Dynamischer Wert oder statischer MAX-Wert
Mode_Power _Consumption **	INPUT	Float32		Energieverbrauch im aktuellen PE_Mode [kW]
Energy_Consumption _to_Destination **	INPUT	Float32		Energieverbrauch bis zum angeforderten PE_Mode [kWh]
Energy_Consumption _to_operate **	INPUT	Float32		Energieverbrauch vom aktuellen PE_Mode bis PE_ready_to_operate [kWh]

* Das PROFIenergy Profile spezifiziert kein ungültiges Zeitformat.
Wenn die Zeitdauer unbegrenzt ist, kann der max. Wert $FFFFFFFF_h$ angegeben werden.
Wenn die Zeitdauer null ist, kann 00_h verwendet werden.

** Wenn ein Energieverbrauchswert nicht definiert ist, kann 0,0 (Float32) angegeben werden.

Mögliche Werte für "PE_Mode_ID_Source" und "PE_Mode_ID_Destination":

- 00_h: PE_Power_off (Spannungsversorgung AUS)
- 01_h...FE_h: Frei projektierbar bzw. herstellerspezifisch
- FF_h: PE_Ready_to_operate (Betriebsbereit)

PE_IDENTIFY_RSP

Generiert die Antwort auf den Befehl "PE_Identify".

Als Anwender müssen Sie angeben, welche PROFIenergy-Befehle unterstützt werden.

Tabelle 4-367 Individuelle Parameter von FC 6 PE_IDENTIFY_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Count	INPUT	BYTE	0...6	Anzahl der unterstützten PROFIenergy-Befehle
Start_Pause	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt
End_Pause	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt
Query_Modes	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt
PEM_Status	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt
PEM_Identify	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt
Query_Measurement	INPUT	BOOL	0...1	• 1: Befehl wird unterstützt • 0: Befehl wird nicht unterstützt

PE_MEASUREMENT_LIST_RSP

Generiert die Antwort auf den Befehl "Query_Measurement" > Modifier "Get_Measurement_List".

Tabelle 4-368 Individuelle Parameter von FC 7 PE_MEASUREMENT_LIST_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Count	INPUT	BYTE		Anzahl der unterstützten Messwert-IDs (Measurement_ID)
Measurement_List	INPUT	ANY		Zeiger auf den Datenbereich mit den unterstützten Messwert-IDs. Als Anwender müssen Sie in diesem Datenbereich die Messwert-IDs hinterlegen. Pro Telegramm können maximal 29 Messwert-IDs übertragen werden. Zum Aufbau des Arrays siehe Kapitel AUTO-HOTSPOT > "Query_Measurement" – Get_Measurement_List.

PE_MEASUREMENT_VALUE_RSP

Generiert die Antwort auf den Befehl "Query_Measurement" > Modifier "Get_Measurement_Values".

Tabelle 4-369 Individuelle Parameter von FC 8 PE_MEASUREMENT_VALUE_RSP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
Count	INPUT	BYTE		Anzahl der unterstützten Measurement_Values
Measurement_Values	INPUT	ANY		Zeiger auf den Datenbereich der Messwerte (Measurement_Values). Als Anwender müssen Sie die Messwerte in diesem Datenbereich hinterlegen. Pro Telegramm können maximal 116 Messwerte übertragen werden. Zum Aufbau des Arrays siehe Kapitel AUTOHOTSPOT > "Query_Measurement" – Get_Measurement_List.

PE_DS3_WRITE_ET200S_CP

Bedeutung und Aufruf - PE_DS3_WRITE_ET200_CP

Bedeutung und Arbeitsweise

In der CPU des CP 300 als PROFIenergy-Controller wird PE_DS3_WRITE_ET200_CP eingesetzt, um Einstellungen des Schaltverhaltens von Powermodulen an eine ET 200S zu übertragen. Das Schaltverhalten für bis zu 8 Steckplätze (hier: Powermodule) kann übertragen werden.

PE_DS3_WRITE_ET200_CP ist kein PROFIenergy-Programmbaustein.

Aufrufschnittstelle in FUP-Darstellung

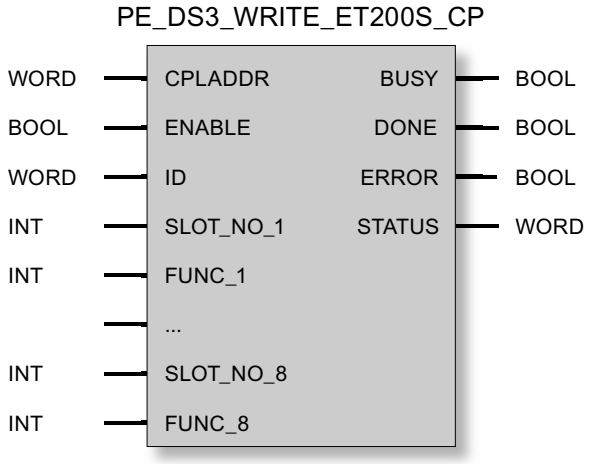

Erläuterung der Formalparameter von PE_DS3_WRITE_ET200S_CP

Erläuterung der Formalparameter von PE_DS3_WRITE_ET200_CP

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD	E, A, M, D, L, Konst.	Baugruppen-Anfangsadresse des CP
ENABLE	INPUT	BOOL		Beginnt mit steigender Flanke die Bearbeitung des Programmbausteins.
ID	INPUT	WORD		Logische Adresse des Header-Steckplatzes des IO-Device
SLOT_NO_1	INPUT	INT		Steckplatznummer des ersten Powermoduls
FUNC_1	INPUT	INT		Festlegung des Schaltverhaltens für das Powermodul bezüglich Start bzw. Ende der Energiesparpausen • 0 (FALSE) – PAUSE_START: Kein Einfluss (Powermodul bleibt eingeschaltet) – PAUSE_STOP: Schaltet Powermodul wieder ein. • 1 (TRUE) – PAUSE_START: Schaltet Powermodul aus. – PAUSE_STOP: Schaltet Powermodul wieder ein.
...	INPUT	INT		
...	INPUT	INT		
SLOT_NO_8	INPUT	INT		Steckplatznummer des achten Powermoduls
FUNC_8	INPUT	INT		Siehe "FUNC_1"
BUSY	OUTPUT	BOOL	0: Bearbeitung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung läuft	Anzeige des Bearbeitungs-Status des Programmbausteins
DONE	OUTPUT	BOOL	0: - 1: Datensatz erfolgreich übertragen	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige
STATUS	OUTPUT	WORD		Statusanzeige

Zur Bedeutung von DONE, ERROR und STATUS siehe Anzeigen von PE_DS3_WRITE_ET200S_CP (Seite 6873).

Anzeigen von PE_DS3_WRITE_ET200S_CP

Anzeigen von PE_DS3_WRITE_ET200_CP

PE_DS3_WRITE_ET200_CP basiert auf dem Programmbaustein PNIO_RW_REC und liefert alle Anzeigen von PNIO_RW_REC, siehe Anzeigen des Bausteins PNIO_RW_REC.

Anweisungen für programmierte Verbindungen

Übersicht

Zusammenspiel von Programmierung und Projektierung

Verbindungen an der SEND/RECEIVE-Schnittstelle sowie die IP-Konfiguration eines CP projektieren Sie entweder per STEP 7 oder konfigurieren diese per Anwenderprogramm zur Laufzeit der S7-Station. Eine Mischform dieser Varianten ist innerhalb eines CPs nicht möglich!

Prinzip der programmierten Konfiguration

Über eine Anweisung, die im Anwenderprogramm aufgerufen wird, können Konfigurationsdaten für Kommunikationsverbindungen und die IP-Konfiguration (Systemdaten für CP) an den CP übertragen werden.

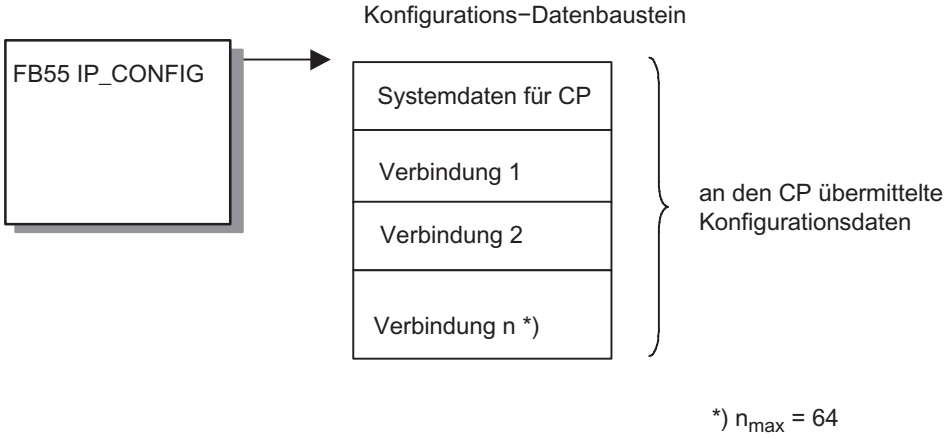

Der Konfigurations-DB kann jederzeit in den CP geladen werden. Die vorher aktuellen Verbindungen und Konfigurationsdaten (IP-Adresse, Subnetz-Maske, Default-Router, NTP-Uhrzeit-Server und weitere Parameter) werden hierbei überschrieben.

Der Ethernet-CP erkennt anhand der Projektierdaten, dass die Kommunikationsverbindungen über das Anwenderprogramm eingerichtet werden sollen.

Hinweis

Die Funktionen können nur ausgeführt werden, wenn der Baugruppen-Zugriffsschutz als "Nicht gesperrt" projektiert wurde: Siehe Eigenschaftendialog des CP, Register "Optionen" (nicht bei jedem CP).

Weiterhin muss die Option "IP-Adresse im Anwenderprogramm erstellen" aktiviert sein (siehe Eigenschaftendialog des CP oder der Ethernet-Schnittstelle des CP, Register "IP-Konfiguration").

Der Ethernet-CP erkennt anhand der Projektierdaten, dass die Kommunikationsverbindungen über das Anwenderprogramm eingerichtet werden sollen.

Hinweis

Sobald das Anwenderprogramm über die Anweisung IP_CONFIG die Verbindungsdaten übergibt, schaltet die CPU den CP kurzzeitig in STOP. Der CP übernimmt die Systemdaten (inklusive IP-Adresse) und die neuen Verbindungsdaten und arbeitet diese im Anlauf ab (RUN).

Mengengerüst

In der Anweisung IP_CONFIG können maximal 64 Verbindungen angegeben werden. Maßgebend ist jedoch die maximale Anzahl Verbindungen, die der von Ihnen genutzte CP-Typ unterstützt.

Besonderheiten / Einschränkungen

- Konsistenzprüfung nur in der Projektierung
  Die Verbindungsprojektierung ist mit Konsistenzprüfungen verbunden, die bei der programmierten Konfiguration nicht oder nur bedingt möglich sind!
- Verbindungsprojektierung beim Partner erforderlich
  Bei der Projektierung spezifizierter Verbindungen wird bei der Projektierung implizit die Verbindung für den Partner angelegt; bei der programmierten Konfiguration ist dies nicht möglich! Hier müssen für die Partner entsprechende Verbindungen konfiguriert werden.
- Projektierung des IP-Zugriffschutzes
  Über den IP-Zugriffschutz besteht die Möglichkeit, die Kommunikation über den CP der lokalen S7-Station auf Partner mit ganz bestimmten IP-Adressen einzuschränken. Diese Parametrierung gilt auch für programmierte Kommunikationsverbindungen. Sie müssen den IP-Zugriffschutz in der Projektierung entweder ausschalten (= Voreinstellung) oder die Kommunikationspartner autorisieren.
- DHCP / DNS wird unterstützt
  Die IP-Adressierung ist bei der programmierten Konfiguration auch über DHCP (und DNS für den Mail-Dienst) möglich.
  Die Nutzung eines DHCP-Servers wird in diesem Fall in der Anweisung IP_CONFIG definiert (nicht in der Projektierung).

- Keine Verbindungsinformationen beim Hochladen
  Beim Hochladen der S7-Stationsdaten in die Projektierung sind die Daten der programmierten Konfiguration nicht enthalten.
- Verbindungsprojektierung bei CPs mit mehreren Schnittstellen
  Vergewissern Sie sich bei CPs mit mehreren Schnittstellen (z.B. mit Gigabit-Schnittstelle) im Gerätehandbuch des CP, ob die Verbindungsprojektierung für beide Schnittstellen unterstützt wird.
- PROFINET IO nicht gleichzeitig möglich
  Bei Geräten, die als PROFINET IO-Controller oder IO-Device betrieben werden sollen, ist die Verbindungseinrichtung über den hier beschriebenen FB55 nicht möglich.
- Keine Verwendung der Anweisung IP_CONFIG bei CP-Einsatz mit hochverfügbaren S7-Verbindungen
  Wenn Sie hochverfügbare S7-Verbindungen über den CP konfigurieren, dürfen Sie die Anweisung IP_CONFIG nicht zur IP-Konfiguration des CP verwenden.

Siehe auch

IP_CONFIG für programmierte Kommunikationsverbindungen (Seite 6875)

Konfigurations-Datenbaustein (Seite 6881)

IP_CONFIG

IP_CONFIG für programmierte Kommunikationsverbindungen

Beschreibung

Die Anweisung IP_CONFIG überträgt die in einem Datenbaustein (Konfigurations-DB) spezifizierten Verbindungsdaten an den CP. Der Konfigurations-DB enthält sämtliche Verbindungsdaten, um die Verbindungen für die SEND/RECEIVE-Schnittstelle eines Ethernet-CP einzurichten.

Diese Variante der programmierten Kommunikationsverbindungen können Sie alternativ zur Verbindungsprojektierung mit STEP 7 einsetzen.

Je nach Größe des Konfigurations-DB erfolgt die Übertragung zum CP in mehreren Segmenten. Sie müssen daher die Anweisung so lange erneut aufrufen, bis die Anweisung mit dem DONE-Bit=1 die vollständige Übertragung signalisiert.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

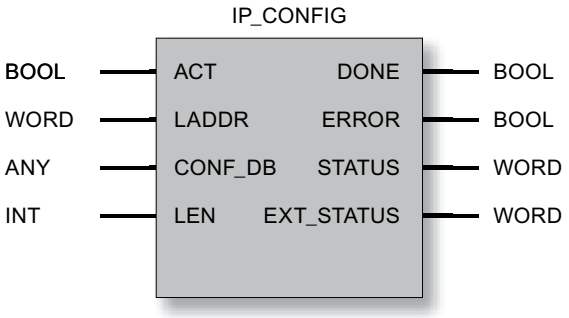

Hinweis

Keine Verwendung der Anweisung IP_CONFIG bei CP-Einsatz mit hochverfügbaren S7-Verbindungen

Wenn Sie hochverfügbare S7-Verbindungen über den CP konfigurieren, dürfen Sie die Anweisung IP_CONFIG nicht zur IP-Konfiguration des CP verwenden.

Hinweis

Auf mögliche Doppeladressierung achten

Wenn Sie die Anweisung IP_CONFIG einsetzen sollten Sie besonders auf die eindeutige Vergabe der IP-Adressen achten. Beim Erkennen einer doppelten Adresse kann es vorkommen, dass der CP am Netz nicht aktiv wird.

Siehe auch

Übersicht (Seite 6873)

Konfigurations-Datenbaustein (Seite 6881)

Parameter bei IP_CONFIG (Seite 6877)

Reservierte Port-Nummern - IP_CONFIG (Seite 6878)

Parameter DONE, ERROR, STATUS (Seite 6878)

Parameter bei IP_CONFIG

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Aufrufschnittstelle der Anweisung "IP_CONFIG":

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL, TIMER, COUNTER	0,1	Beim Aufruf der Anweisung mit ACT = 1 wird der DBxx an den CP gesendet. Beim Aufruf der Anweisung mit ACT = 0 werden nur die Statusanzeigen DONE, ERROR und STATUS aktualisiert.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
CONF_DB	INPUT	ANY		Der Parameter zeigt auf die Anfangsadresse des Konfigurations-Datenbereiches in einem Datenbaustein (Datentyp: Byte).
LEN	INPUT	INT		Längenangabe in Byte für den Konfigurations-Datenbereich.
DONE	OUTPUT	BOOL	0: - 1: Auftrag mit Datenübertragung abgeschlossen.	Der Parameter zeigt an, ob der Konfigurations-Datenbereich vollständig übertragen wurde. Beachten Sie, dass die Anweisung je nach Größe des Konfigurations-Datenbereiches mehrfach angestoßen (in mehreren Zyklen) werden muss, bis die Anzeige DONE=1 den Abschluss signalisiert. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe nachfolgende Tabelle.
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe nachfolgende Tabelle.
STATUS	OUTPUT	WORD	siehe nachfolgende Tabelle	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe nachfolgende Tabelle.
EXT_Status	OUTPUT	WORD		Der Parameter zeigt bei einer fehlerhaften Auftragsausführung an, welcher Parameter im Konfigurations-DB als Fehlerursache erkannt wurde. High Byte: Index des Parameterblockes Low Byte: Index des Subblockes innerhalb des Parameterblockes

Siehe auch

IP_CONFIG für programmierte Kommunikationsverbindungen (Seite 6875)

Reservierte Port-Nummern - IP_CONFIG

Reservierte Port-Nummern

Die folgenden lokalen Port-Nummern sind reserviert; verwenden Sie diese nicht bei der Verbindungsprojektierung.

Tabelle 4-370 Reservierte Port-Nummern

Protokoll	Port-Nummer	Dienst
TCP	20, 21	FTP
TCP	25	SMTP
TCP	80	HTTP
TCP	102	RFC1006
TCP	135	RPC-DCOM
TCP	502	ASA Application Protocol
UDP	161	SNMP_REQUEST
UDP	34964	PN IO
UDP	65532	NTP
UDP	65533	NTP
UDP	65534	NTP
UDP	65535	NTP

Siehe auch

IP_CONFIG für programmierte Kommunikationsverbindungen (Seite 6875)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Tabelle 4-371 Anzeigen IP_CONFIG

DONE	ERROR	STATUS	Bedeutung
Allgemeine Anzeigen zur Auftragsausführung			
1	0	0000_H	Auftrag fertig ohne Fehler
0	0	8181_H	Auftrag läuft
Fehler, die an der Schnittstelle zwischen CPU und CP erkannt wurden.			
0	1	$80A4_H$	• Kommunikationsfehler am K-Bus oder • Datenfehler: Es ist nicht eingestellt, dass die Konfiguration über das Anwenderprogramm erfolgt.

DONE	ERROR	STATUS	Bedeutung
0	1	$80B1_H$	Die Anzahl der zu sendenden Daten überschreitet die für diesen Dienst zulässige Obergrenze. (Obergrenze = 16 kByte)
0	1	$80C4_H$	Kommunikationsfehler Der Fehler kann temporär auftreten; daher ist eine Wiederholung im Anwenderprogramm sinnvoll.
0	1	$80D2_H$	Projektierungsfehler Die eingesetzte Baugruppe unterstützt diesen Dienst nicht.
Fehler, die bei Auswertungen des IP_CONFIG in der CPU oder an der Schnittstelle zwischen CPU und CP erkannt wurden.			
0	1	8183_H	Der CP lehnt die angeforderte Datensatznummer ab.
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp. (Datentyp des ANY-Pointer CONF_DB nicht ok) (Aktuell wird nur der Datentyp Byte akzeptiert)
0	1	8185_H	Der Wert des Parameters LEN ist größer als der CONF_DB abzüglich des reservierten Header (4 Byte) oder die Längenangabe ist falsch.
0	1	8186_H	Unzulässiger Parameter erkannt Der ANY-Pointer CONF_DB verweist nicht auf einen Datenbaustein.
0	1	8187_H	Ungültiger Zustand des IP_CONFIG Eventuell wurden Daten im Header des CONF_DB überschrieben.
Weitere Fehler, die an der Schnittstelle zwischen CPU und CP erkannt wurden.			
0	1	$8A01_H$	Die Statusanzeige im gelesenen Datensatz ist ungültig (Wert ist >= 3).
0	1	$8A02_H$	Es läuft kein Auftrag auf dem CP. Der IP_CONFIG hat jedoch eine Quittung für den gelaufenen Auftrag erwartet.
0	1	$8A03_H$	Es läuft kein Auftrag auf dem CP und der CP ist nicht bereit. Der IP_CONFIG hat einen ersten Auftrag für Datensatz-Lesen angestoßen.
0	1	$8A04_H$	Es läuft kein Auftrag auf dem CP und der CP ist nicht bereit. Der IP_CONFIG hat jedoch eine Quittung für den gelaufenen Auftrag erwartet.
0	1	$8A05_H$	Es läuft ein Auftrag, eine Quittierung ist jedoch noch nicht erfolgt. Der IP_CONFIG hat jedoch einen ersten Auftrag für Datensatz-Lesen angestoßen.
0	1	$8A06_H$	Ein Auftrag ist fertig. Der IP_CONFIG hat jedoch einen ersten Auftrag für Datensatz-Lesen angestoßen.
Fehler, die bei Auswertungen des IP_CONFIG im CP erkannt wurden.			
0	1	$8B01_H$	Kommunikationsfehler Der DB konnte nicht übertragen werden.
0	1	$8B02_H$	Parameterfehler Doppelter Parameterblock
0	1	$8B03_H$	Parameterfehler Der Subblock im Parameterblock ist nicht erlaubt.
0	1	$8B04_H$	Parameterfehler Die Länge, die an der Anweisung angegeben wurde, stimmt nicht mit der Länge der Parameterblöcke / Subblöcke überein.
0	1	$8B05_H$	Parameterfehler Die Länge des Parameterblockes ist ungültig.
0	1	$8B06_H$	Parameterfehler Die Länge des Subblockes ist ungültig.
0	1	$8B07_H$	Parameterfehler Die ID des Parameterblockes ist ungültig

DONE	ERROR	STATUS	Bedeutung
0	1	$8B08_H$	Parameterfehler Die ID des Subblockes ist ungültig
0	1	$8B09_H$	Systemfehler Die Verbindung existiert nicht
0	1	$8B0A_H$	Datenfehler Der Inhalt des Subblockes ist nicht korrekt.
0	1	$8B0B_H$	Strukturfehler Ein Subblock ist doppelt aufgeführt.
0	1	$8B0C_H$	Datenfehler Im Parameterblock sind nicht alle erforderlichen Parameter enthalten.
0	1	$8B0D_H$	Datenfehler Der CONF_DB enthält keinen Parameterblock für Systemdaten.
0	1	$8B0E_H$	Datenfehler / Strukturfehler Der Typ des CONF_DB ist ungültig.
0	1	$8B0F_H$	Systemfehler Der CP hat zu wenig Ressourcen, um den CONF_DB vollständig bearbeiten zu können.
0	1	$8B10_H$	Datenfehler Es ist nicht eingestellt, dass die Konfiguration über das Anwenderprogramm erfolgt.
0	1	$8B11_H$	Datenfehler Der angegebene Typ des Parameterblockes ist ungültig.
0	1	$8B12_H$	Datenfehler Es wurden zu viele Verbindungen angegeben (entweder insgesamt oder zu viele für einen bestimmten Typ; es ist beispielsweise nur eine E-Mail Verbindung möglich).
0	1	$8B13_H$	CP-interner Fehler
0	1	$8B14_H$	Die aktive Schutzstufe lässt die Aktion zur Änderung nicht zu.
Weitere Fehler, die an den Programmschnittstellen innerhalb der CPU erkannt wurden (System-Anweisungsfehler).			
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	Der Parameter liegt im schreibgeschützten 1. aktuellen Datenbaustein.
0	1	$8F31_H$	Der Parameter liegt im schreibgeschützten 2. aktuellen Datenbaustein.
0	1	$8F32_H$	Der Parameter enthält eine zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummernfehler
0	1	$8F3A_H$	Der Zielbereich wurde nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F44_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu lesenden Parameter ist gesperrt.

DONE	ERROR	STATUS	Bedeutung
0	1	$8F45_H$	Der Zugriff auf einen in der Bearbeitung der Anweisung zu schreibenden Parameter ist gesperrt.
0	1	$8F7F_H$	Interner Fehler Es wurde z. B. eine unzulässige ANY-Referenz erkannt.

Siehe auch

IP_CONFIG für programmierte Kommunikationsverbindungen (Seite 6875)

Konfigurations-Datenbaustein

Konfigurations-Datenbaustein (CONF_DB)

Bedeutung

Der Konfigurations-Datenbaustein (CONF_DB) enthält sämtliche Verbindungsdaten und Konfigurationsdaten (IP-Adresse, Subnetz-Maske, Default-Router, NTP-Uhrzeit-Server und weitere Parameter) für einen Ethernet-CP. Der Konfigurations-Datenbaustein wird mit der Anweisung IP_CONFIG an den CP übergeben.

Aufbau / Datenstruktur

Sie können der folgenden Darstellung entnehmen:

- Strukturierung durch Parameterblöcke und Subblöcke
  - Jede Verbindung sowie besondere Systemdaten werden durch einen identisch aufgebauten Parameterblock beschrieben.
  - Einzelne Parameter werden durch Subblöcke typisiert.
- Offset-Bereich
  Der CONF_DB kann über einen Offset-Bereich an beliebiger Stelle innerhalb eines Datenbausteines beginnen. Die Adresse (bzw. der Offset) muss lediglich geradzahlig sein.

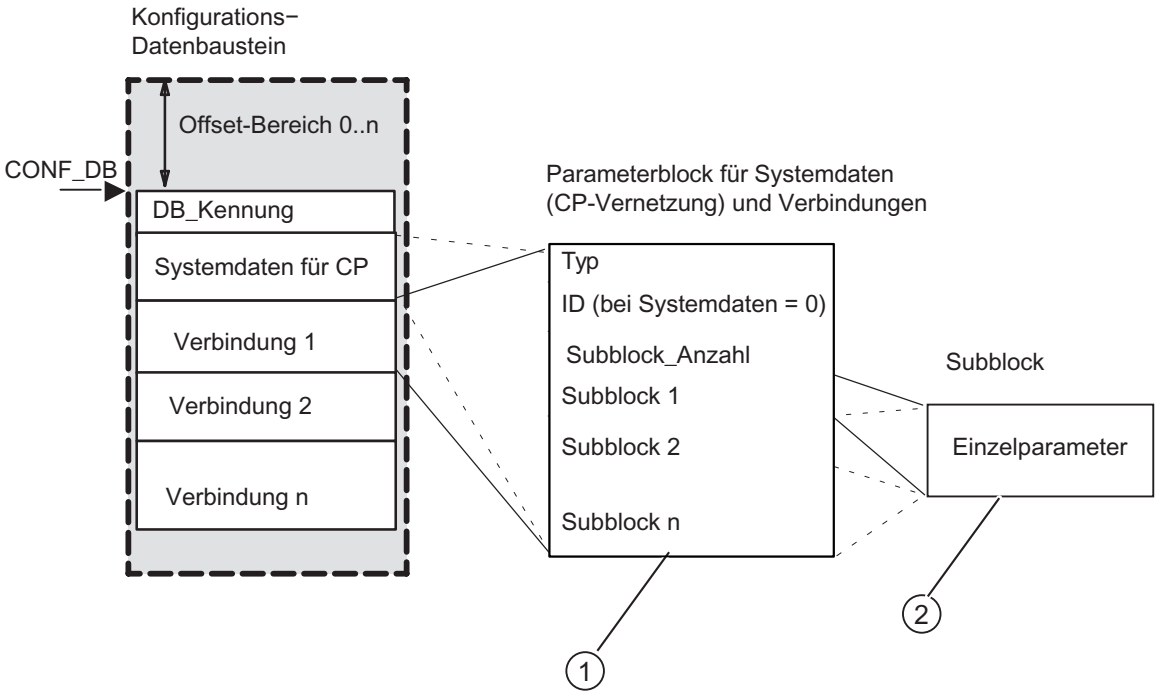

① Parameterblöcke sind nachfolgend beschrieben in Parameterblock für Systemdaten (IP-Konfiguration) (Seite 6882)

② Subblocktypen sind nachfolgend beschrieben in Parameterblöcke für Verbindungstypen (Seite 6886)

Parameterblock für Systemdaten (IP-Konfiguration)

Bedeutung

Sie finden nachfolgend den für die IP-Konfiguration des CP relevanten Parameterblock für Systemdaten und die darin anzugebenden Subblöcke.

Je nach Anwendungsfall werden nicht alle Subblock-Typen benötigt - entsprechende Angaben finden Sie in der Tabelle.

Aufbau

Bei CPs mit mehreren Schnittstellen gilt der nachfolgend beschriebene Aufbau für den Parameterblock nur für die PROFINET-Schnittstelle.

• **Typ = 0**
• **ID = 0**
• Subblock_Anzahl = n
• Subblock 1
• Subblock 2
• Subblock n ...

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
1	SUB_IP_V4	Lokale IP-Adresse	++
2	SUB_NETMASK	-	++
8	SUB_DEF_ROUTER	-	+
4	SUB_DNS_SERV_ADDR *)	Dieser Subblock kann bis zu 0-4 mal vorkommen. Der erste Eintrag ist der Primary DNS Server.	+
14	SUB_DHCP_ENABLE	0: kein DHCP 1: DHCP	+
15	SUB_CLIENT_ID	- Anmerkung: Nur sinnvoll wenn SUB_DHCP_ENABLE = 1	+
30**)	SUB_DEVICE_NAME	Gerätename nach PROFINET IO-Konvention Geben Sie einen Gerätenamen an, um das Gerät für Analyse- und Diagnosezwecke im Netz individuell kenntlich zu machen.	+
*) Der Subblocktyp wird nur für E-Mail-Verbindungen benutzt. **) ID wird nur bei bestimmten CP-Typen unterstützt. ***) ++ = zwingend; + = optional			

Siehe auch

Subblock-Typen (Seite 6883)

Subblock-Typen

Je nach Parameterblock werden unterschiedliche Parameter benötigt. Jeder Parameter wird durch einen Subblock beschrieben. Welche Subblöcke benötigt werden, können Sie den Beschreibungen zu den Systemdaten und zu den Verbindungstypen in den vorhergehenden Kapiteln entnehmen.

Jeder Subblock besteht aus dem spezifischen Parameterabschnitt sowie einem Header (4 Byte).

Beispiel

Der folgende Auszug aus einem CONF_DB zeigt den Aufbau eines Subblockes am Beispiel des Subblock-Typs SUB_NETMASK.

Adresse	Name	Typ	Anfangswert	Kommentar
+14.0	Sub_Block_2	STRUCT		// Subblock 2 Typ SUB_NET-MASK
+0.0	Sub_Block_ID	INT	2	// Subblock ID
+2.0	Sub_Block_Len	INT	8	// Gesamtlänge des Subblocks in Byte
+4.0	Parameter	STRUCT		Parameterbereich SUB_NET-MASK
+0.0	Wert_1	BYTE	B#16#FF	
+1.0	Wert_2	BYTE	B#16#FF	
+2.0	Wert_3	BYTE	B#16#FF	
+3.0	Wert_4	BYTE	B#16#0	
=4.0	END_	STRUCT		
=8.0	END_	STRUCT		

Folgende Subblock-Typen stehen insgesamt zur Verfügung:

Subblock ID [1)]	Subblock-Typ	Subblock-Länge (in Byte)	Bedeutung des Parameters
1	SUB_IP_V4	4 + 4	IP-Adresse gemäß IPv4
2	SUB_NETMASK	4 + 4	Subnetzmaske
3	SUB_DNS_NAME	Länge DNS Name + 4	DNS Name
4	SUB_DNS_SERV_ADDR	4 + 4	DNS Server Adresse.
8	SUB_DEF_ROUTER	4 + 4	IP-Adresse des Default Router
9	SUB_LOC_PORT	2 + 4	Lokaler Port
10	SUB_REM_PORT	2 + 4	Ferner Port, auch für E-MAIL Verbindungen
11	SUB_LOC_TSAP	Tsap-Länge + 4	Lokaler TSAP
12	SUB_REM_TSAP	Tsap-Länge + 4	Ferner TSAP
13	SUB_EMAIL_SENDER	Länge der Absender E–Mail Adresse + 4	E-Mail Adresse des Absenders
14	SUB_DHCP_ENABLE	2 + 4	IP-Adresse von einem DHCP-Server beziehen. • Wertebereich: 0 = kein DHCP 1 = DHCP (optional)
15	SUB_CLIENT_ID	Länge der Client ID + 4	(optional)
18	SUB_CONNECT_NAME	Länge des Namens + 4	Name der Verbindung. Mögliche Zeichen sind: a...z, A...Z, 0...9, -, _

Subblock ID [1)]	Subblock-Typ	Subblock-Länge (in Byte)	Bedeutung des Parameters
19	SUB_LOC_MODE	1 + 4	Lokale Betriebsart der Verbindung • Wertebereich: 0x00 = SEND/RECV 0x01 = FTP Protokoll (nur bei TCP-Verb.) 0x10 = S5-Adressierungsmodus bei FETCH/WRITE *) 0x20 = SPEED SEND/RECV (nur beim CP 443-1 Advanced erlaubt) 0x80 = FETCH *) 0x40 = WRITE *) Die Default-Einstellung bei Verzicht auf den Parameter ist SEND/RECV. Anmerkung: FETCH / WRITE erfordern die Einstellung passiver Verbindungsaufbau (siehe unter SUB_CON_ESTABL).
20	SUB_REM_MODE	1 + 4	Einstellung der Betriebsart beim Kommunikationspartner. (wird derzeit nicht unterstützt)
21	SUB_KBUS_ADR	5	KBUS-Adresse der CPU (nur relevant für S7–400)
22	SUB_CON_ESTABL	1 + 4	Typ des Verbindungsaufbaus. Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von dieser S7-Station aus aktiv oder passiv erfolgen soll. • Wertebereich: 0 = Passiv 1 = Aktiv
23	SUB_ADDR_IN_DATA-BLOCK	1 + 4	Freie UDP-Verbindung wählen. Der ferne Teilnehmer wird vom Anwenderprogramm beim AG_SEND Aufruf im Auftragsheader des Auftragspuffers eingetragen. Damit kann jeder beliebige Teilnehmer am Ethernet/LAN/WAN erreicht werden. • Wertebereich: 1 = Freie UDP-Verbindung 0 = sonstige Der Parameter ist nur bei der UDP-Verbindung sinnvoll.
24	SUB_NTP_SERVER	4 + 4	Der Subblock definiert einen NTP-Server, von dem der CP seine Uhrzeit über das NTP-Protokoll beziehen kann. Für den Fall, dass ein oder mehrere NTP-Server definiert werden, können bis zu 4 Subblocks der ID 24 definiert werden. Die Subblocks der ID 24 dürfen nur in den Systemparameterblock Typ 0 / ID 0 eingebaut werden.

Subblock ID [1]	Subblock-Typ	Subblock-Länge (in Byte)	Bedeutung des Parameters
30	SUB_DEVICE_NAME	Länge des Namens + 4	Gerätename nach PROFINET IO-Konvention Der Gerätename muss den DNS-Konventionen genügen, d. h. • Beschränkung auf 127 Zeichen insgesamt (Buchstaben, Ziffern, Bindestrich oder Punkt) • Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf max. 63 Zeichen lang sein. • Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Der Bindestrich ist das einzige erlaubte Sonderzeichen. • Der Gerätename darf nicht mit den Zeichen "-" oder "." beginnen und auch nicht mit diesem Zeichen enden. • Der Gerätename darf nicht mit Ziffern beginnen. • Der Gerätename darf nicht die Form n.n.n.n haben (n = 0...999). • Der Gerätename darf nicht mit der Zeichenfolge "port-xyz-" beginnen (x,y,z = 0...9).
1) Hinweis: nicht aufgeführte ID-Nummern werden derzeit noch nicht genutzt.			

Parameterblöcke für Verbindungstypen

Allgemeines

Sie erfahren nachfolgend, welche Werte in die Parameterblöcke einzutragen sind und welche Subblöcke zu den jeweiligen Verbindungstypen zu verwenden sind.

Je nach Anwendungsfall werden nicht alle Subblock-Typen benötigt - entsprechende Angaben finden Sie ebenfalls in der Tabelle.

Verbindungs-ID

Von besonderer Bedeutung ist der ID-Parameter, der jedem Verbindungsparameterblock neben der Typkennung vorangestellt wird.

Bei programmierten Verbindungen können Sie diese ID innerhalb des zulässigen Wertebereiches frei vergeben. Sie müssen diese ID dann an der Aufrufschnittstelle der FCs für die SEND/RECV-Schnittstelle zur Identifizierung der Verbindung verwenden.

Wertebereiche für die Verbindungs-ID:

- S7-400: 1,2...64
- S7-300: 1,2...16

Siehe auch

Subblock-Typen (Seite 6883)

Parameterblock für TCP-Verbindung

Aufbau

Hinterlegen Sie die Parameter im Parameterblock für TCP-Verbindungen wie folgt:

• **Typ = 1 -> ①**
• **ID = Verbindungs-ID -> ②**
• Subblock_Anzahl = n
• Subblock 1
• Subblock 2
• Subblock n ...

Legende:
① Kennung für den Verbindungstyp
② frei zu vergebene Verbindungsreferenz; im AG_SEND / AG_RECV anzugeben.
Wertebereich für die Verbindungs-ID:
bei S7-400: 1, 2...64
bei S7-300: 1,2...16

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
1	SUB_IP_V4	IP-Adresse des Partners	++ *)
9	SUB_LOC_PORT	-	++
10	SUB_REM_PORT	-	++ **)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	Bei CPs für S7-300 ist dieser Wert fest auf 2 eingestellt und muß daher nicht angegeben werden.	++ (bei S7-400)
22	SUB_CON_ESTABL	-	++
*) optional bei passiver Verbindung. ***) ++ = zwingend; + = optional			

Siehe auch

Subblock-Typen (Seite 6883)

Parameterblock für UDP- Verbindung

Aufbau

Hinterlegen Sie die Parameter im Parameterblock für UDP-Verbindungen wie folgt:

• **Typ = 2 -> ①**
• **ID = Verbindungs-ID -> ②**
• Subblock_Anzahl = n
• Subblock 1
• Subblock 2
• Subblock n ...

Legende:
① Kennung für den Verbindungstyp
② frei zu vergebene Verbindungsreferenz; im AG_SEND / AG_RECV anzugeben.
Wertebereich für die Verbindungs-ID:
bei S7-400: 1, 2...64
bei S7-300: 1,2...16

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
1	SUB_IP_V4	IP-Adresse des Partners	++
9	SUB_LOC_PORT	-	++
10	SUB_REM_PORT	-	++
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	Bei CPs für S7-300 ist dieser Wert fest auf 2 eingestellt und muss daher nicht angegeben werden.	++ (bei S7-400)
23	SUB_ADDR_IN_DATABLOCK	Falls über diesen Parameter die Option "Freie UDP-Verbindung" gewählt wird, entfallen die Parameter SUB_IP_V4 und SUB_REM_PORT.	+
***) ++ = zwingend; + = optional			

Siehe auch

Subblock-Typen (Seite 6883)

Parameterblock für ISO-on-TCP Verbindung

Aufbau

Hinterlegen Sie die Parameter im Parameterblock für ISO-on-TCP-Verbindungen wie folgt:

• **Typ = 3 -> ①**
• **ID = Verbindungs-ID -> ②**
• Subblock_Anzahl = n
• Subblock 1
• Subblock 2
• Subblock n ...

Legende:
① Kennung für den Verbindungstyp
② frei zu vergebene Verbindungsreferenz; im AG_SEND / AG_RECV anzugeben.
Wertebereich für die Verbindungs-ID:
bei S7-400: 1, 2...64
bei S7-300: 1,2...16

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
1	SUB_IP_V4	IP-Adresse des Partners	++ *)
11	SUB_LOC_TSAP	-	++
12	SUB_REM_TSAP	-	++ *)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	Bei CPs für S7-300 ist dieser Wert fest auf 2 eingestellt und muß daher nicht angegeben werden.	++ (bei S7-400)
22	SUB_CON_ESTABL	-	++
*) optional bei passiver Verbindung. ***) ++ = zwingend; + = optional			

Siehe auch

Subblock-Typen (Seite 6883)

Parameterblock für E-Mail-Verbindung

Bedeutung

Für das Senden von E-Mails ist grundsätzlich eine E-Mail-Verbindung pro Advanced-CP einzurichten. Mit der E-Mail-Verbindung ist der Mailserver festgelegt, über den sämtliche vom Advanced-CP gesendeten E-Mails zugestellt werden.

Aufbau

Hinterlegen Sie die Parameter im Parameterblock für E-Mail-Verbindungen wie folgt:

• **Typ = 4 -> ①**
• **ID = Verbindungs-ID -> ②**
• Subblock_Anzahl = n
• Subblock 1
• Subblock 2
• Subblock n ...

Legende:
① Kennung für den Verbindungstyp
② frei zu vergebene Verbindungsreferenz; im AG_SEND / AG_RECV anzugeben.
Wertebereich für die Verbindungs-ID:
bei S7-400: 1, 2...64
bei S7-300: 1,2...16

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
1	SUB_IP_V4	IP-Adresse des Mailservers, über den die E-Mails gesendet werden. Die IP-Adresse kann absolut oder symbolisch angegeben werden. Die symbolische Angabe setzt voraus, dass dem Advanced-CP die Adresse des Domain Name Servers (DNS) bekannt ist. Ein entsprechender Eintrag ist bei der Projektierung des Advanced-CP in HW Konfig vorzunehmen; nähere Angaben hierzu finden Sie dort in der Online-Hilfe.	++ / + *)
3	SUB_DNS_NAME	DNS Name des E-MAIL Servers	++ / + *)
13	SUB_EMAIL_SENDER	Absender E-Mail Adresse	++
18	SUB_CONNECT_NAME	-	+

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
21	SUB_KBUS_ADR	Bei CPs für S7-300 ist dieser Wert fest auf 0 eingestellt und muß daher nicht angegeben werden.	++ (bei S7-400)
22	SUB_CON_ESTABL	-	++
*) Die Parameter SUB_IP_V4 und SUB_DNS_NAME schließen sich hier gegenseitig aus; ausschließlich einer von beiden Parametern ist anzugeben. ***) ++ = zwingend; + = optional			

Hinweis

Mail-Server-Ports sind "well known ports" und müssen nicht zwingend angegeben werden.

Siehe auch

Subblock-Typen (Seite 6883)

Parameterblock für FTP-Verbindung

Bedeutung

Für die Abwicklung einer FTP-Auftragssequenz zwischen der S7-Station als FTP-Client und einem FTP-Server muss der Advanced-CP eine Verbindung zur S7-CPU einrichten. Diese Verbindung bezeichnen wir hier als FTP-Verbindung.

Bei den FTP-Verbindungen handelt es sich um TCP-Verbindungen, die über den Parameter SUB_LOC_MODE auf die Betriebsart "FTP" eingestellt werden.

Aufbau

Hinterlegen Sie die Parameter im Parameterblock für FTP-Verbindungen wie folgt:

• **Typ = 1 -> ①**
• **ID = Verbindungs-ID -> ②**
• Subblock_Anzahl = n
• Subblock 1

• Subblock 2
• Subblock n ...

Legende:

① Kennung für den Verbindungstyp

② frei zu vergebene Verbindungsreferenz; im AG_SEND / AG_RECV anzugeben.
Wertebereich für die Verbindungs-ID:
bei S7-400: 1, 2...64
bei S7-300: 1,2...16

Verwendbare Subblöcke

Subblock		Parameter	
ID	Typ	Besonderheiten / Hinweise	Anwendung ***)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	hier: 0x01 = FTP Protokoll	++
21	SUB_KBUS_ADR	Bei CPs für S7-300 ist dieser Wert fest auf 0 eingestellt und muß daher nicht angegeben werden.	++ (bei S7-400)
***) ++ = zwingend; + = optional			

Siehe auch

Subblock-Typen (Seite 6883)

Anweisungen für FTP-Dienste

FTP_CMD für FTP-Dienste

Migration von FC 40-44 zum FB 40

Vergleich des Programmbausteins FTP_CMD mit den älteren Funktionen FC40...44

Sämtliche CPs mit FTP-Funktionalität unterstützen die Funktionen FC40...44. Bestehende Anwenderprogramme können also unverändert verwendet werden.

Wenn Sie von den FTP-Funktionen FC40...44 auf den FTP_CMD umstellen wollen, dann müssen Sie Ihr Anwenderprogramm ändern.

Die folgende Tabelle zeigt Ihnen, mit welchen Kommandos des FTP_CMD die Funktionen der FC40...44 umgesetzt werden.

- Übereinstimmungen sind mit "X" markiert.
- Fehlende Entsprechungen sind mit "-" gekennzeichnet.

	Kommandos des Parameters "CMD" des FTP_CMD					
Ältere FTP-Funktionen FC40...44	CMD = 1	CMD = 2	CMD = 3	CMD = 4	CMD = 5	CMD = 6 CMD = 7 CMD = 17
FC40	X [1]					
FC41		X [2]				
FC42			X [3]			
FC43				X [4]		
FC44					X	
						-
[1]...[4] Die Parameter von FC 40...43 und CMD 1...4 (FTP_CMD) sind nicht identisch. (siehe nachfolgende Tabelle)						

Die entsprechenden Parameter, die in den Funtionen FC40...FC43 bzw. in den Kommandos des FTP_CMD die jeweilige Funktion spezifizieren, sind in folgender Tabelle aufgelistet.

Parameter des FC			Parameter im FTP_CMD (mit CMD 1...4)	
FC40:	LOGIN	→	CMD = 1:	NAME_STR
FC41:	FILE_NAME	→	CMD = 2:	NAME_STR
FC42:	FILE_NAME	→	CMD = 3:	NAME_STR
FC43:	FILE_NAME	→	CMD = 4:	NAME_STR
FC40...43:	BUFFER_DB_NR	→	entfällt (durch Instanz-DB ersetzt)	

Siehe auch

Übersicht FTP_CMD (Seite 6893)

Übersicht FTP_CMD

Bedeutung

Mit der Anweisung FTP_CMD können Sie FTP-Verbindungen aufbauen und Dateien von und zu einem FTP-Server übertragen.

Mit der Anweisung FTP_CMD ergeben sich folgende Vorteile:

- Vereinfachung im Anwenderprogramm durch Abwicklung über Kommando-Variable anstelle unterschiedlicher Funktionsaufrufe;
- Zusätzliche Funktion "APPEND"
  "APPEND" ermöglicht, Daten an eine bestehende Datei anzuhängen.
- Zusätzliche Funktion "RETR_PART"
  "RETR_PART" ermöglicht, Datenbereiche selektiv aus einer Datei zu lesen.
- Zusätzliche Funktion "CONNECT_TLS_PRIVATE"
  "CONNECT_TLS_PRIVATE" ermöglicht den Aufbau von gesichterten SSL-Verbindungen
- Die Funktion AG_SEND (FC5) wird für die Abwicklung nicht benötigt.

Gültigkeit

Der FB40 kann ab folgenden Baugruppentypen eingesetzt werden:

- Ab CP 343-1 Advanced (GX30*/GX31)
- Ab CP 443-1 Advanced (GX20*/GX30)
  *) Funktion "CONNECT-TLS-PRIVATE" nicht nutzbar.

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

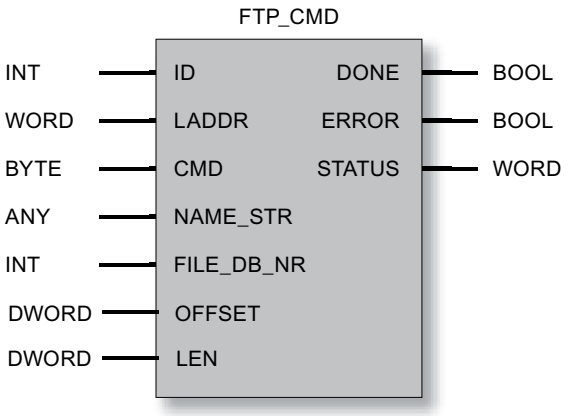

Aufgerufene Systemfunktionen

Folgende Systemfunktionen werden vom Programmbaustein FTP_CMD aus aufgerufen:

SFC 1, SFC 20, SFC 24, SFC 58, SFC 59

Hinweis

Beachten Sie, dass die FTP-Client-Dienste von alten SIMATIC S7-300-CPUs, zum Beispiel CPU 312 oder CPU 315-1AF01, nicht ausgeführt werden können, da diese den SFC 24 nicht unterstützen.

Siehe auch

Aufbau und Verwendung des File-DB (Seite 6903)

Ausgangsparameter und Statusinformationen FTP_CMD (Seite 6899)

Eingangsparameter FTP_CMD (Seite 6895)

Migration von FC 40-44 zum FB 40 (Seite 6892)

Eingangsparameter FTP_CMD

Erläuterung der Eingangsparameter

Sie müssen jeden FTP-Bausteinaufruf mit folgenden Eingangsparametern versorgen:

Tabelle 4-372 Formalparameter des FB40 (FTP_CMD) - Eingangsparameter

Parameter	Deklaration	Typ	Wertebereich	Bedeutung / Bemerkung
ID	INPUT	INT	Bei S7-300: 1, 2...16 Bei S7-400: 1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Beim Aufruf eines FC-Bausteines übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des ADVANCED-CP. Die Baugruppen-Anfangsadresse des ADVANCED-CP können Sie in der Projektierung des ADVANCED-CP unter "Eigenschaften>Adressen>Eingänge" entnehmen.
CMD	INPUT	BYTE	Siehe Tabelle unten -FTP-Kommandos im Parameter"CMD"	FTP-Kommandos, die mit Aufruf des FTP_CMD ausgeführt werden. Weitere Informationen finden Sie im Anschluss an die Tabelle. Wenn ein Kommando nicht von der CP-Firmware unterstützt wird, dann wird eine Fehlermeldung mit STATUS = $8F6B_H$ ausgegeben. Beispiele für FTP-Kommandos: • RETRIEVE: B#16#3 • CONNECT_TLS_PRIVATE: B#16#11

Parameter	Deklaration	Typ	Wertebereich	Bedeutung / Bemerkung
NAME_STR	INPUT	ANY	als VARTYPE sind nur "BYTE" zugelassen.	Die angegebene Adresse verweist auf einen Datenbausteinbereich. Anzugeben sind hier Adresse und Länge des Datenbereichs, in dem die Zielangaben eingetragen sind. • Bei CMD = 1, 17: Bei diesem Kommando spezifiziert der Parameter "NAME_STR" den mit der FTP-Verbindung anzusprechenden FTP-Server mit folgenden Attributen: - IP-Adresse des FTP-Servers - Benutzername - Passwort für das Login Diese Werte sind als drei aufeinander folgende Strings im Zielbereich des ANY-Pointers anzugeben. • Bei CMD = 2, 3, 4, 6, 7: Bei diesem Kommando spezifiziert der Parameter "NAME_STR" den Dateinamen auf dem FTP-Server, also die Datenquelle bzw. das Datenziel. Der Dateiname ist als String im Zielbereich des ANY-Pointers anzugeben. • Bei CMD = 5: Parameter nicht relevant Beispiele für Inhalte finden Sie weiter unten.
FILE_DB_NR	INPUT	INT		Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB. Der Parameter ist nur bei CMD = 2, 3, 6 und 7 relevant.

Parameter	Deklaration	Typ	Wertebereich	Bedeutung / Bemerkung
OFFSET	INPUT	DWORD		Nur bei CMD = 7: Offset in Byte, ab dem die Datei gelesen werden soll.
LEN	INPUT	DWORD		Nur bei CMD = 7: Teillänge in Byte, die ab dem in "OFFSET" angegebenen Wert gelesen werden. Besonderheiten: • Bei Angabe von "DW#16#FFFFFFFF" wird der verfügbare Rest der Datei gelesen. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET > Länge der Original-Datei: Die Länge der Ziel-Datei wird in diesem Fall wie folgt im Parameter ACT_LENGTH im File-DB angezeigt: 0 Byte in der CPU. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET + LEN > Länge der Original-Datei (und LEN ≠ 0xFFFFFFFF): Die Länge der Ziel-Datei wird in diesem Fall wie folgt im Parameter ACT_LENGTH im File-DB angezeigt: Verfügbare Bytes ab "OFFSET". Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt.

FTP-Kommandos im Parameter "CMD"

Entnehmen Sie der folgenden Tabelle, welche Bedeutung die Kommandos des Parameters "CMD" haben und welche Input-Parameter jeweils zu versorgen sind. Die Parameter ID und LADDR müssen zur Identifikation der Verbindung immer gesetzt sein.

CMD	Relevante Input-Parameter (zusätzlich zu ID und LADDR)	Bedeutung / Handhabung
0 (NOOP)	-	Die aufgerufene Anweisung führt keine Aktionen aus. Die Statusanzeigen werden bei dieser Parameterversorgung wie folgt gesetzt: • DONE=1; ERROR=0; STATUS=0
1 (CONNECT)	NAME_STR	Mit diesem Kommando baut der FTP-Client eine FTP-Verbindung zu einem FTP-Server auf (Port 21). Die Verbindung steht unter der hier zugewiesenen Verbindungs-ID für alle weiteren FTP-Kommandos zur Verfügung. Daten werden dann mit dem für diesen Benutzer angegebenen FTP-Server ausgetauscht.
2 (STORE)	NAME_STR FILE_DB_NR	Mit diesem Funktionsaufruf wird ein Datenbaustein (File-DB) vom FTP-Client (S7-CPU) zum FTP-Server übertragen. Achtung: Falls die Datei (File-DB) auf dem FTP-Server schon vorhanden ist, wird diese überschrieben.

CMD	Relevante Input-Parameter (zusätzlich zu ID und LADDR)	Bedeutung / Handhabung
3 (RETRIEVE)	NAME_STR FILE_DB_NR	Mit diesem Funktionsaufruf wird eine Datei vom FTP-Server zum FTP-Client (S7-CPU) übertragen. Achtung: Falls der Datenbaustein (File-DB) beim FTP-Client schon eine Datei enthält, wird diese überschrieben.
4 (DELETE)	NAME_STR	Mit diesem Funktionsaufruf löschen Sie eine Datei auf dem FTP-Server.
5 (QUIT)	Keine weiteren	Mit diesem Funktionsaufruf bauen Sie die über die ID benannte FTP-Verbindung ab.
6 (APPEND)	NAME_STR FILE_DB_NR	Ähnlich wie "STORE" speichert das Kommando "APPEND" (anhängen) eine Datei auf dem FTP-Server. Bei "APPEND" wird die Datei auf dem FTP-Server aber nicht überschrieben, sondern der neue zu speichernde Inhalt wird an die Datei angehängt. Falls die Datei auf dem FTP-Server nicht vorhanden ist, wird sie angelegt.
7 (RETR_PART)	NAME_STR FILE_DB_NR OFFSET LEN	Mit dem Kommando "RETR_PART" (Teillänge lesen) können Sie einen Ausschnitt einer Datei vom FTP-Server anfordern. Bei sehr großen Dateien können Sie damit das Lesen auf den Teil beschränken, den Sie gerade benötigen. Dazu müssen Sie die Struktur dieser Datei kennen. Geben Sie den gewünschten Ausschnitt der Datei mit Hilfe der zwei Parameter "OFFSET" und "LEN" am FB 40 an.
17 (CONNECT_TLS_PRIVATE)	NAME_STR	Mit dem Kommando "CONNECT_TLS_PRIVATE" richtet der FTP-Client eine SSL-gesicherte FTP-Verbindung (FTPS) zum FTP-Server (Port 21) ein. Die Daten der Kontrollverbindung als auch die der Datenverbindung sind sicher. Die Verbindung steht unter der hier zugewiesenen Verbindungs-ID für alle weiteren FTP-Kommandos zur Verfügung. Daten werden dann mit dem für diesen Benutzer angegebenen FTP-Server ausgetauscht. Voraussetzungen: • Die Option "Security" muss beim CP aktiviert sein. • Für die SSL-gesicherte FTP-Verbindung müssen Zertifikate in der CP-Projektierung geliefert werden.

Beispiele für Inhalte des Parameters "NAME_STR"

Der Parametersatz hat folgende Inhalte:

Tabelle 4-373 Inhalt des Parametersatzes für CMD = 1, 17

Relative Adresse [2]	Name	Typ [1]	Beispiel	Bedeutung
0.0	ip_address	STRING[100]	’142.11.25.135’	IP-Adresse des FTP-Servers
102.0	username	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server

Relative Adresse 2)	Name	Typ 1)	Beispiel	Bedeutung
136.0	password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
1) Angegeben ist die jeweils maximal mögliche String-Länge 2) Die angegebenen Werte beziehen sich auf die unter "Typ" angegebenen String-Längen.				

Tabelle 4-374 Inhalt des Parametersatzes für CMD = 2, 3, 4, 6, 7

Relative Adresse 2)	Name	Typ 1)	Beispiel	Bedeutung
170.0	filename	STRING[220] bzw. STRING[212]	’anlage1/kessel2/druck.dat’	Dateiname der Ziel- bzw. Quelldatei Hinweis: Bei CMD=7 (RETR_PART) ist die maximale Länge des Dateinamens auf 212 Zeichen begrenzt.
1) Angegeben ist die jeweils maximal mögliche String-Länge 2) Die angegebenen Werte beziehen sich auf die unter "Typ" angegebenen String-Längen.				

Siehe auch

Übersicht FTP_CMD (Seite 6893)

Ausgangsparameter und Statusinformationen FTP_CMD

Einleitung

Für die Statusauswertung sind im Anwenderprogramm die Parameter auszuwerten:

Tabelle 4-375 Formalparameter des FB40 (FTP_CMD) - Ausgangsparameter

Parameter	Deklaration	Typ	Wertebereich	Bedeutung / Bemerkung
DONE	OUTPUT	BOOL	0: - 1: Auftrag ausgeführt	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Der Parameter meldet, wenn der Auftrag nicht fehlerfrei ausgeführt werden konnte.
STATUS	OUTPUT	WORD	siehe nachfolgende Tabelle	Statusanzeige Der Parameter liefert Detailinformation zur Auftragsausführung.

Die Parameter DONE, ERROR und STATUS werden bei jedem Bausteinaufruf aktualisiert.

Beispiel

Während einer Auftragsausführung liefert der FTP_CMD die Anzeigen:

- DONE=0
- ERROR=0
- STATUS=8181_H

Bedeutung: Der Auftrag läuft noch.

Statusanzeigen auswerten

Hinweis

Beachten Sie für die Einträge mit der Codierung 8FxxH unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Tabelle 4-376 FB 40: Bedeutung des Parameters STATUS in Zusammenhang mit DONE und ERROR

DONE	ERROR	STATUS	Bedeutung
0	0	0000_H	Kein Auftrag in Bearbeitung.
1	0	0000_H	Auftrag fertig ohne Fehler.
0	0	8181_H	Auftrag läuft. Bei Daueranzeige von 8181_H: Der CP ist nicht für den FTP_CMD freigegeben (ein für die Firmware-Version unzulässiges Kommando CMD 6, CMD 7 oder CMD 17 wurde aufgerufen.)
0	1	8090_H	• Eine Baugruppe mit dieser Baugruppen-Anfangsadresse ist nicht vorhanden. • Der verwendete Baustein passt nicht zur verwendeten Systemfamilie (für S7-300 und S7-400 sind unterschiedliche Bausteine zu verwenden).
0	1	8091_H	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster
0	1	8092_H	Typangabe im ANY-Pointer ungleich Byte
0	1	$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut (bei neueren CPU-Ausgabeständen). Dies kann beispielsweise begründet sein durch: • eine fehlende Projektierung von Verbindungen • eine Überschreitung der maximalen Anzahl parallel betreibbarer CPs
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Ziel-Bereich ist ungültig; beispielsweise Ziel-Bereich > 240 Byte.
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut (bei älteren CPU-Ausgabeständen). (bei neueren CPU-Ausgabeständen siehe $80A4_H$)
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.

DONE	ERROR	STATUS	Bedeutung
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)
0	1	$80D2_H$	Baugruppen-Anfangsadresse ist falsch.
0	1	8183_H	Die Projektierung entspricht nicht den Auftragsparametern.
0	1	8184_H	Unzulässiger Datentyp für den Parameter NAME_STR angegeben.
0	1	8186_H	Parameter ID ungültig. ID = 1, 2....64
0	1	8302_H	
0	1	$8F22_H$	Quell-Bereich ungültig; beispielsweise: Bereich im DB nicht vorhanden
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z.B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters
0	1	$8F28_H$	Ausrichtungfehler beim Lesen eines Parameters
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummernfehler
0	1	$8F3A_H$	Bereich nicht geladen (DB)
0	1	$8F50_H$	File-DB DB 0 oder DB nicht vorhanden
0	1	$8F51_H$	File-DB-Datenbereich größer angegeben als vorhanden
0	1	$8F52_H$	File-DB im schreibgeschützten Speicher
0	1	$8F53_H$	File-DB max. Länge < akt. Länge
0	1	$8F54_H$	File-DB enthält keine gültigen Daten.
0	1	$8F55_H$	Header-Status-Bit: Locked
0	1	$8F56_H$	Das NEW-Bit im File-DB-Header wurde nicht zurückgesetzt
0	1	$8F57_H$	FTP-Client hat nicht das Schreibrecht auf den File-DB, sondern der FTP-Server (Header-Status-Bit: WriteAccess).
0	1	$8F60_H$	Ungültige Benutzer-Daten, beispielsweise ungültige IP Adresse des FTP-Servers
0	1	$8F61_H$	FTP-Server nicht erreichbar
0	1	$8F62_H$	Mögliche Bedeutungen: • Auftrag wird vom FTP-Server nicht unterstützt oder zurückgewiesen • Der FTP-Server unterstützt keine SSL-gesicherten Verbindungen.
0	1	$8F63_H$	Dateitransfer wurde vom FTP-Server abgebrochen
0	1	$8F64_H$	Fehler auf der FTP-Control-Verbindung; Daten konnten nicht gesendet oder empfangen werden; Die FTP-Control-Verbindung muss nach einem solchen Fehler erneut aufgebaut werden.
0	1	$8F65_H$	Fehler auf der FTP-Daten-Verbindung; Daten konnten nicht gesendet oder empfangen werden. Der Auftrag muss erneut aufgerufen werden. Der Fehler kann beispielsweise bei der Funktion RETRIEVE (CMD=3) dadurch hervorgerufen werden, dass die angesprochene Datei auf dem FTP-Server bereits geöffnet ist.
0	1	$8F66_H$	Fehler beim Lesen/Schreiben der Daten von/zur CPU (beispielsweise DB nicht vorhanden oder zu klein)

DONE	ERROR	STATUS	Bedeutung
0	1	$8F67_H$	Fehler im FTP-Client auf dem CP; beispielsweise beim Versuch, mehr als die maximal mögliche Anzahl an FTP-Verbindungen zu öffnen.
0	1	$8F68_H$	Der Auftrag wurde vom FTP-Client zurückgewiesen. Der Fehler kann beispielsweise bei der Funktion RETRIEVE (CMD=3) dadurch hervorgerufen werden, dass der Wert für den Parameter MAX_LENGTH im File-DB-Header zu klein gewählt wurde.
0	1	$8F69_H$	Die FTP-Verbindung befindet sich in einem falschen Zustand, z.B.: • Es erfolgt ein Aufruf der Verbindung ohne vorhergehenden Verbindungsabbau (bei gleicher Verbindungs-ID); • Es erfolgt ein Verbindungsabbau auf eine bereits abgebaute Verbindung; • Es wurde ein STORE-Kommando auf eine nicht aufgebaute Verbindung abgesetzt.
0	1	$8F6A_H$	Die Verbindung konnte wegen temporärem Ressourcenengpass nicht aufgebaut werden. Abhilfe: Bausteinaufruf wiederholen.
0	1	$8F6B_H$	Mögliche Ursachen: • Falscher Wert für den Parameter CMD • Ein Kommando des FTP_CMD wird nicht unterstützt. Mögliche Ursache: Falsche Firmware des CP Abhilfe: Firmware-Update
0	1	$8F6C_H$	Im Parameter OFFSET wurde ein Wert > 7FFF FFF_H angegeben.
0	1	$8F6D_H$	Der FTP-Client unterstützt keine SSL-gesicherte Verbindung.
0	1	$8F6F_H$	Mögliche Ursachen: • Im Zertifikat enthält die Zeitangabe "notBefore" einen ungültigen Wert. • Das Zertifikat ist ungültig: der Eintrag "notBefore" enthält eine nach der aktuellen Zeit liegende Zeitangabe.
0	1	$8F70_H$	Mögliche Ursachen: • Im Zertifikat enthält die Zeitangabe "notAfter" einen ungültigen Wert. • Das Zertifikat istabgelaufen: der Eintrag "notAfter" enthält eine vor der aktuellen Zeit liegende Zeitangabe.
0	1	$8F71_H$	Das Aussteller-Zertifikat eines nicht vertrauenswürdigen Zertifikats konnte nicht gefunden werden
0	1	$8F72_H$	Das ursprüngliche CA-Zertifikat ist ungültig. Entweder handelt es sich nicht um ein CA-Zertifikat oder dessen Erweiterungen sind nicht konsistent mit dem beabsichtigten Zweck.
0	1	$8F73_H$	Das ursprüngliche CA-Zertifikat ist für den angegebenen Zweck als nicht vertrauenswürdig gekennzeichnet
0	1	$8F74_H$	Es sind sonstige Fehler bei der Verifizierung eines Zertifikats aufgetreten.
0	1	$8F7F_H$	Interner Fehler; beispielsweise unzulässige ANY-Referenz

Siehe auch

Übersicht FTP_CMD (Seite 6893)

Aufbau und Verwendung des File-DB

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten

- Abschnitt 1: File-DB-Header (besitzt feste Länge (20 Byte) und Struktur)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

File-DB-Header für FTP-Server-Betrieb

Anmerkung: Der hier beschriebene File-DB-Header ist weitgehend identisch zu dem für den Client-Betrieb beschriebenen File-DB-Header. Die Unterschiede beziehen sich auf die Parameter

- WRITE_ACCESS
- FTP_REPLY_CODE

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten ("Datei existiert nicht"). • 1: Der File-DB enthält gültige Nutzdaten ("Datei existiert").	Das FTP-Kommando dele setzt EXIST=0; Das FTP-Kommando store setzt EXIST=1;
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos store und retr setzen während der Bearbeitung LOCKED=1. Für einen Schreibvorgang aus dem Anwenderprogramm gilt: Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes LOCKED setzen bzw. zurücksetzen. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen; wenn =0 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen; wenn =0 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen

Parameter	Typ	Wert / Bedeutung	Versorgung
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muss nach dem Lesen der Daten NEW=0 setzen, um ein erneutes store zu ermöglichen oder um das File mittels FTP-Kommando dele löschen zu können.
WRITE_ACCESS	BOOL	0: Der FTP-Client auf dem PG/PC hat kein Schreibrecht für die File-DBs in der S7-CPU. 1: Der FTP-Client auf dem PG/PC hat Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungwert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereiches (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Der Parameter ist im FTP-Serverbetrieb ohne Bedeutung.	Wird vom FTP-Server auf "0" gesetzt.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung des Files. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wird die Funktion "Uhrzeitweiterleitung" genutzt, entspricht der Eintrag der weitergeleiteten Zeit. Wird die Funktion "Uhrzeitweiterleitung" nicht genutzt, wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des IT-CP (Initialisierungswert ist 1.1.1994 0.0 Uhr).

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

File-DB-Header für FTP-Client-Betrieb

Anmerkung: Der hier beschriebene File-DB Header ist weitgehend identisch zu dem für den Server-Betrieb beschriebenen File-DB-Header. Die Unterschiede beziehen sich auf die Parameter:

- WRITE_ACCESS
- FTP_REPLY_CODE

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten ("Datei existiert nicht"). • 1: Der File-DB enthält gültige Nutzdaten ("Datei existiert").	Das FTP-Kommando dele setzt EXIST=0; Das FTP-Kommando store setzt EXIST=1;
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos store und retr setzen während der Bearbeitung LOCKED=1. Für einen Schreibvorgang aus dem Anwenderprogramm gilt: Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes LOCKED setzen bzw. zurücksetzen. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen; wenn =0 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen; wenn =0 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muß nach dem Lesen der Daten NEW=0 setzen, um ein erneutes Kommando retr zu ermöglichen.

Parameter	Typ	Wert / Bedeutung	Versorgung
WRITE_ACCESS	BOOL	0: Das Anwenderprogramm (FTP-Client Bausteine) hat Schreibrecht für die File-DBs in der S7-CPU. 1: Das Anwenderprogramm (FTP-Client Bausteine) hat kein Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungwert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereiches (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Vorzeichenlose Zahl (16 Bit), die den letzten Reply-Code von FTP als Binärwert enthält. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Wird vom FTP-Client bei der FTP-Kommandobearbeitung aktualisiert.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung des Files. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wird die Funktion "Uhrzeitweiterleitung" genutzt, entspricht der Eintrag der weitergeleiteten Zeit. Wird die Funktion "Uhrzeitweiterleitung" nicht genutzt, wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des IT-CP (Initialisierungswert ist 1.1.1994 0.0 Uhr).

Datenbaustein FILE_DB_HEADER als Vorlage

Bedeutung

Für das Anlegen eines File-DB-Header ist der Datentyp FILE_DB_HEADER vordefiniert.

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

Gehen Sie wie folgt vor:

1. Legen Sie bei der PLC-CPU, in der Sie das Anwenderprogramm mit den FTP-Anweisungen erstellen, einen Datenbaustein vom Typ "Global-DB" an.
2. Selektieren Sie die Zeile, die Sie als Startzeile für den File-DB verwenden werden.
3. Wählen Sie über die Klappliste in der Spalte "Datentyp" ein Strukturelement vom Typ FILE_DB_HEADER aus.

Ergebnis: Es wird eine Datenstruktur mit der für den File-DB benötigte Header-Struktur angelegt.

Hinweis

Funktion "Neuen Baustein hinzufügen" - Typauswahl

Beim Anlegen neuer Datenbausteine wird unter dem Eintrag "Typ" in der Klappliste auch der Bausteintyp "FILE_DB_HEADER" angeboten. Verwenden Sie diese Auswahlmöglichkeit nicht! Der so erzeugte Datenbaustein enthält lediglich die Header-Struktur und kann nicht um den benötigten Bereich zur Ablage von Nutzdaten erweitert werden.

Datenbaustein FILE_DB_HEADER - Beispiel und Vorlage für den File-DB-Header

In der Deklarationsansicht erkennen Sie folgende Struktur:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	bit08	BOOL	FALSE	reserviert
+0.1	bit09	BOOL	FALSE	reserviert
+0.2	bit10	BOOL	FALSE	reserviert
+0.3	bit11	BOOL	FALSE	reserviert
+0.4	bit12	BOOL	FALSE	reserviert
+0.5	bit13	BOOL	FALSE	reserviert
+0.6	bit14	BOOL	FALSE	reserviert
+0.7	bit15	BOOL	FALSE	reserviert
+1.0	EXIST	BOOL	FALSE	if TRUE: FileDB content is valid data
+1.1	LOCKED	BOOL	FALSE	it TRUE: FileDB is locked caused by changes of the content
+1.2	NEW	BOOL	FALSE	if TRUE: FileDB content is new and may not be overwritten
+1.3	WRITE_ACCESS	BOOL	FALSE	if TRUE: Ftp-Server of the IT-CP has write access, else Ftp-Server
+1.4	bit04	BOOL	FALSE	reserviert
+1.5	bit05	BOOL	FALSE	reserviert
+1.6	bit06	BOOL	FALSE	reserviert
+1.7	bit07	BOOL	FALSE	reserviert

Adresse	Name	Typ	Anfangswert	Kommentar
+2.0	ACT_LENGTH	DINT	L#0	actual size of the content in bytes (not including the header of 20 bytes)
+6.0	MAX_LENGTH	DINT	L#0	max. size of the content in bytes (not including the header of 20 bytes)
+10.0	FTP_REPLY_CODE	INT	0	last reply code from the remote FTP-Server
+12.0	DATE_TIME	DATE_AND_TIME	DT#00-1-1-0:0:0.000	date and time of last change of the content of the FileDB
=20.0		END_STRUCT		

Unterschiede bei den Betriebsarten

File-DB-Header für FTP-Client-Betrieb

Der hier beschriebene File-DB-Header ist in den Betriebsarten für FTP-Client-Betrieb und FTP-Server-Betrieb weitgehend identisch. Die Unterschiede beziehen sich auf die Parameter

- WRITE_ACCESS
- FTP_REPLY_CODE

Beachten Sie hierzu die unterschiedlichen Beschreibungen der Parameter:

- Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Server-Betrieb (Seite 6903)
- Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb (Seite 6906)

Anweisungen für ERPC-CP

LOGICAL_TRIGGER für den logischen Trigger

Bedeutung der Anweisung

Für die Nutzung eines logischen Triggers für die ERPC-Kommunikation steht die Anweisung LOGICAL_TRIGGER zur Verfügung.

Um einen logischen Trigger zu starten, rufen Sie die Anweisung LOGICAL_TRIGGER im Anwenderprogramm der CPU im OB1 auf.

Für den Aufruf der Anweisung LOGICAL_TRIGGER werden weitere Bausteine benötigt:

- Ein automatisch generierter Instanz-DB
- Ein Datenbaustein "CONF_DB"
  In diesem Konfigurations-DB befinden sich die Konfigurationsdaten des logischen Triggers. Den Konfigurations-DB müssen Sie im STEP 7-Projekt anlegen und projektieren. Wenn Sie mehrere logische Trigger aufrufen wollen, dann müssen Sie mehrere Konfigurations-DBs bereitstellen.

Sie können die Nummern der Anweisung LOGICAL_TRIGGER und des Instanz-DB ändern.

Gültigkeit

Die Anweisung LOGICAL_TRIGGER ist mit folgenden Baugruppentypen verwendbar:

- CP 343-1 ERPC

Aufruf

Aufrufschnittstelle in FUP-Darstellung

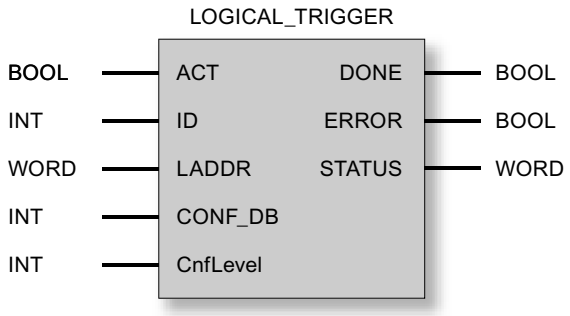

Siehe auch

Der Konfigurations-Datenbaustein (Seite 6916)

Arbeitsweise LOGICAL_TRIGGER (Seite 6911)

Erläuterung der Formalparameter - LOGICAL_TRIGGER (Seite 6912)

Anzeigen der Anweisung LOGICAL_TRIGGER (Seite 6914)

Arbeitsweise LOGICAL_TRIGGER

Arbeitsweise

Die folgende Tabelle zeigt die Schritte, die an einem Trigger-Aufruf durch das Anwenderprogramm der CPU beteiligt sind.

Schritt	Bedeutung
1	Die Anweisung LOGICAL_TRIGGER wird an der vorgesehenen Stelle im Anwenderprogramm der CPU mit zugehörigem Instanz-DB und dem projektierten Konfigurations-Datenbaustein CONF_DB aufgerufen. • Wenn die Anweisung LOGICAL_TRIGGER mit ACT = 1 aufgerufen wird, dann werden die aktuellen Trigger-Daten gelesen und an die CP-Firmware gesendet. • Wenn die Anweisung LOGICAL_TRIGGER mit ACT = 0 aufgerufen wird, dann werden die Statusanzeigen DONE, ERROR und STATUS aktualisiert.
2	Die Anweisung LOGICAL_TRIGGER liest die aktuellen Trigger-Daten ein.
3	Die Anweisung LOGICAL_TRIGGER bildet die PDU, die mit den aktuellen Trigger-Daten an die CP-Firmware gesendet wird.

Schritt	Bedeutung
4	Die CP-Firmware bildet das Datentelegramm und übergibt es der ERPC-Applikation.
5	Die ERPC-Applikation sendet das Datentelegramm an den ERP-Teilnehmer (ERP-System oder MES).

Siehe auch

LOGICAL_TRIGGER für den logischen Trigger (Seite 6910)

Erläuterung der Formalparameter - LOGICAL_TRIGGER

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Aufrufschnittstelle der Anweisung LOGICAL_TRIGGER:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL	0	Beim Aufruf der Anweisung LOGICAL_TRIGGER mit ACT = 0 werden die Statusanzeigen DONE, ERROR und STATUS aktualisiert.
			1	Beim Aufruf der Anweisung LOGICAL_TRIGGER mit ACT = 1 werden die aktuellen Trigger-Daten eingelesen und an den CP gesendet.
ID	INPUT	INT		Trigger-ID Dieser Wert identifiziert den in der ILS-Workbench projektierten logischen Trigger.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP mit STEP 7 wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle von HW Konfig ausgegeben. Geben Sie diese Adresse hier an.
CONF_DB	INPUT	INT		In diesem Datenbaustein befinden sich die Konfigurationsdaten der projektierten logischen Trigger.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CnfLevel	INPUT	INT	0: Transportquittung 1: Ende-zu-Ende-Quittung	Quittiermodus Die jeweilige Quittung finden Sie über den STATUS-Wert in den Anzeigen der Anweisung LOGICAL_TRIGGER. • 0 = Transportquittung (STATUS = 0000_H) Der Auftrag wird als erfolgreich gemeldet, sobald die Daten an die ERPC-Applikation übergeben wurden. Das muss jedoch nicht unbedingt bedeuten, dass das Datentelegramm an den ERP-Teilnehmer (ERP-System oder MES) gesendet worden ist oder dass nicht noch nachträglich ein Fehler von der ERPC-Applikation festgestellt werden kann. • 1 = Ende-zu-Ende-Quittung (STATUS = 0001_H) Der Auftrag wird erst quittiert, nachdem die ERPC-Applikation die Daten überprüft hat. Über die Variable "TriggerResponse" des Konfigurations-DB (DB_CONF) wird gemeldet, ob der ERP-Teilnehmer erreichbar war oder ob sich die ERPC-Applikation im Store-and-Forward-Modus befindet. Die Ende-zu-Ende-Quittung führt zu einer längeren Auftragslaufzeit als die Transportquittung.
DONE	OUTPUT	BOOL	0: Auftrag läuft 1: Auftrag abgeschlossen	Der Parameter zeigt an, ob der Auftrag zur Übertragung des Konfigurations-Datenbereichs fehlerfrei abgewickelt wurde. DONE wird vom CP bei Auftragsannahme auf 0 gesetzt. Solange DONE = 0 ist, kann kein weiterer Auftrag ausgelöst werden. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Tabelle "Anzeigen Anweisung LOGICAL_TRIGGER".
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe Tabelle "Anzeigen Anweisung LOGICAL_TRIGGER".
STATUS	OUTPUT	WORD	Siehe Tabelle "Anzeigen Anweisung LOGICAL_TRIGGER".	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe Tabelle "Anzeigen Anweisung LOGICAL_TRIGGER".

Siehe auch

LOGICAL_TRIGGER für den logischen Trigger (Seite 6910)

Anzeigen der Anweisung LOGICAL_TRIGGER (Seite 6914)

Anzeigen der Anweisung LOGICAL_TRIGGER

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Tabelle 4-377 Anzeigen Anweisung LOGICAL_TRIGGER

DONE	ERROR	STATUS	Bedeutung
Anzeigen zur Auftragsausführung			
1	0	0000_H	Auftrag fertig ohne Fehler. Der logische Trigger wurde erfolgreich abgeschlossen.
1	0	0001_H	Auftrag fertig ohne Fehler. Die Datenbank ist nicht erreichbar (Store-and-forward-Modus).
0	0	8181_H	Auftrag läuft.
0	1	7000_H	Die Anweisung LOGICAL_TRIGGER wurde mit ACT = 0 aufgerufen. Der Auftrag wird jedoch nicht bearbeitet. Rufen Sie die Anweisung LOGICAL_TRIGGER mindestens einmal mit ACT = 1 auf.
Anzeigen zu Projektierung und Ablauf des logischen Triggers			
0	1	$80D2_H$	Der eingesetzte CP unterstützt keine ERPC-Kommunikation (falscher CP-Typ).
0	1	8183_H	Der eingesetzte CP unterstützt keine ERPC-Kommunikation (falscher CP-Typ).
0	1	8187_H	Ungültiger Zustand der Anweisung LOGICAL_TRIGGER (unbekannter LOGICAL_TRIGGER_STATE). Rufen Sie die Anweisung LOGICAL_TRIGGER erneut auf.
0	1	$8A01_H$	Die Anzahl der projektierten logischen Trigger ist gleich 0.
0	1	$8A02_H$	Für diesen logischen Trigger ist keine Projektierung im Konfigurations-DB. Prüfen Sie die ILS-Workbench-Projektierung.
0	1	$8A03_H$	Die Struktur des Konfigurations-DB ist nicht korrekt. Der "header identifier" hat nicht den richtigen Wert. Korrigieren Sie im Konfigurations-DB den Wert der Variablen "ident" (siehe Gerätehandbuch ERPC-CP).
0	1	$8A04_H$	Die Struktur des Konfigurations-DB ist nicht korrekt. Laden Sie die ILS-Workbench-Projektierung erneut in den CP, legen Sie den (die) Konfigurations-DB neu an und projektieren Sie diesen (siehe Gerätehandbuch ERPC-CP).
0	1	$8A05_H$	Der projektierte Konfigurations-DB ist nicht in der CPU vorhanden.
0	1	$8A06_H$	Bei einem Folgeaufruf wurde ein noch laufender Trigger mit einer anderen ID aufgerufen. Prüfen Sie die "ID" in der aufgerufenen Anweisung LOGICAL_TRIGGER.
0	1	$8A08_H$	Die Projektierungsdaten im Konfigurations-DB sind nicht oder noch nicht vollständig vorhanden. Wenn der Fehler nur im Anlauf der S7-Station auftritt, dann kann die Ursache sein, dass die Projektierungsdaten des logischen Triggers noch nicht vollständig an den Konfigurations-DB übertragen worden sind. Wenn der Fehler weiterhin auftritt, dann prüfen Sie die Projektierung der ERPC-Symbole.
0	1	$8A09_H$	Im Konfigurations-DB wurde ein unbekannter Fehler gemeldet.
0	1	$8A0A_H$	Der logische Trigger kann nicht gestartet werden, da gerade eine neue Trigger-Konfiguration geladen wird.
0	1	$8A0B_H$	Fehler bei der Ermittlung des Zeitstempels des aktuellen Datensatzes (CPU-Daten)

DONE	ERROR	STATUS	Bedeutung
0	1	$8A0C_H$	Der Konfigurations-DB wurde mit der Eigenschaft "Unlinked" erzeugt. Korrigieren Sie die Objekteigenschaften der Anweisung LOGICAL_TRIGGER.
0	1	$8A0D_H$	Fehler beim Eingangsparameter CONF_DB der Anweisung LOGICAL_TRIGGER. Der Parameter hat den Wert "0" oder ist größer als die für die CPU maximal zulässige DB-Nummer.
0	1	$8A0E_H$	Die übergebene Trigger-ID liegt nicht im zulässigen Bereich von 1...16. Korrigieren Sie den Wert im Aufruf der Anweisung LOGICAL_TRIGGER im Anwenderprogramm.
0	1	$8A0F_H$	Der vorgegebene Quittiermodus (CnfLevel) ist ungültig. Korrigieren Sie den Wert im Aufruf der Anweisung LOGICAL_TRIGGER im Anwenderprogramm.
0	1	$8Bxx_H$	Fehler beim Umkopieren aktueller Variablenwerte in die PDU des logischen Triggers. Die letzten beiden Stellen (xx) liefern die Variablennummer. Prüfen Sie die Projektierung des betroffenen Symbols in der Symboltabelle der CPU und in der Liste der ERPC-Symbole im Eigenschaftendialog des CP.
0	1	$8C01_H$	Die interne Statusanzeige der Anweisung LOGICAL_TRIGGER ist ungültig. Laden Sie die ILS-Workbench-Projektierung erneut in den CP, legen Sie den (die) Konfigurations-DB neu an und projektieren Sie diesen (siehe Gerätehandbuch ERPC-CP).
0	1	$8C02_H$	Der Rückgabewert der Ende-zu-Ende-Quittung ist ungültig. Laden Sie die ILS-Workbench-Projektierung erneut in den CP, legen Sie den (die) Konfigurations-DB neu an und projektieren Sie diesen (siehe Gerätehandbuch ERPC-CP).
0	1	$8C03_H$	Der logische Trigger enthält mehr als 255 Variablen.
0	1	$8C06_H$	Fehler beim Datensatz lesen.
0	1	$8D03_H$	Bei einer Datenbank-Aktion meldet die Firmware einen Timeout.
0	1	$8D04_H$	Die Datenbank-Applikation meldet einen allgemeinen Fehler bei der Quittierung der aktuellen Aktion.
0	1	$8E01_H$	Der projektierte Konfigurations-DB in der CPU ist nicht groß genug. Ändern Sie die Größe des Konfigurations-DB.
0	1	$8E06_H$	Es wurde noch keine Verbindung zum logischen Trigger aufgebaut.
0	1	$8EXX_H$	Bei diesem Status-Code mit Werten für XX_H im Bereich von $02_H..FF_H$ handelt es sich um Abbildungen einer internen Trigger-Response. Falls solche Werte auftreten, sind diese für Service-Zwecke relevant.

Siehe auch

LOGICAL_TRIGGER für den logischen Trigger (Seite 6910)

Erläuterung der Formalparameter - LOGICAL_TRIGGER (Seite 6912)

Der Konfigurations-Datenbaustein

Bereitstellung des Konfigurations-Datenbausteins "CONF_DB"

Wenn Sie die ERPC-Funktion "Logischer Trigger" verwenden, dann müssen Sie in STEP 7 einen Datenbaustein (DB) für die Konfigurationsdaten des logischen Triggers anlegen und in den Aufrufparametern der Anweisung LOGICAL_TRIGGER angeben. Die Anweisung LOGICAL_TRIGGER greift auf den CONF_DB zu. Für das Anwenderprogramm hat der CONF_DB keine weitere Bedeutung.

Programmierung des Konfigurations-Datenbausteins

Zur Identifizierung des neu angelegten DB müssen Sie den DB öffnen und in den ersten beiden freien Zeilen den "header identifier" und die DB-Größe festlegen.

Öffnen Sie in STEP 7 den DB und projektieren Sie die ersten zwei freien Zeilen mit den Variablen "ident" und "data" folgendermaßen:

Adresse	Name	Typ	Anfangswert	Kommentar (*optional*)
*)		STRUCT *)		
*)	ident	DWORD	DW#16#45525043	*header identifier*
*)	data	array[1..2048]		*DB-Größe (siehe nachfolgenden Warnhinweis)*
*)		Byte		
*)		END_STRUCT *)		
*) Werte werden programmseitig eingetragen				

Hinweis

DB-Größe

Für die DB-Größe werden 2 048 Byte empfohlen. Sollte sich während der Inbetriebnahme zeigen, dass dieser Wert nicht ausreicht, dann vergrößern Sie ihn. Ein zu kleiner Wert wird von der Anweisung LOGICAL_TRIGGER mit einem Fehler und dem STATUS "$8A05_H$" gemeldet.

Siehe auch

LOGICAL_TRIGGER für den logischen Trigger (Seite 6910)

PROFIBUS

Einführung PROFIBUS

Allgemeine Hinweise zu den Anweisungen für PROFIBUS CPs

Verfügbare Anweisungen

Nachfolgende Liste gibt die symbolischen Namen der Prorammbausteine (Anweisungen) an, die bei der Lieferung verwendet werden. Die symbolischen Namen können von Ihnen nicht geändert werden.

Kommunikationsdienst / Funktionsbereich	Anweisung (symbolischer Name) 1)	Bibliothek	
		CP 300	CP 400
PROFIBUS DP	DP_SEND	x	x
	DP_RECV	x	x
	DP_DIAG	x	x
	DP_CTRL	x	x
SEND / RECEIVE (Offene Kommunikationsdienste)	AG_SEND		x
	AG_RECV		x
	AG_LSEND		x
	AG_LRECV		x
S7-Kommunikation	BSEND		x
	BRCV		x
	PUT		x
	GET		x
	USEND		x
	URCV		x
	C_CNTRL		x

1) Anmerkung:
Die folgenden Beschreibungen enthalten auch Hinweise auf abweichende Verhaltensweisen bei unterschiedlichen Versionen der Anweisungen. Bitte achten Sie auf die Versionskennzeichnungen der von Ihnen verwendeten Anweisungen.
Die mit der Automation Workbench installierten SIMATIC Manager globalen Bibliotheken enthalten die zum Zeitpunkt der STEP 7-Freigabe aktuellen Versionen der Anweisungen.

Welche Bausteinversion verwenden ?

Die folgenden Beschreibungen enthalten auch Hinweise auf abweichende Verhaltensweisen bei unterschiedlichen Versionen der Anweisungen. Bitte achten Sie auf die Versionskennzeichnungen der von Ihnen verwendeten Anweisungen.

Hinweis

Es wird empfohlen, für alle Baugruppentypen immer die aktuellen Versionen der Anweisungen zu verwenden.

Informationen über die aktuellen Versionen der Anweisungen sowie die aktuellen Anweisungen zum Download finden Sie bei unserem Customer Support im Internet:

http://support.automation.siemens.com/WW/view/de/8797900

Bei älteren Baugruppentypen setzt diese Empfehlung voraus, dass Sie den für diesen Baugruppentyp aktuellen Firmware-Stand verwenden.

Anweisungen im Ersatzteilfall

Unter Ersatzteilfall wird hier der Austausch einer Baugruppe gegen eine andere Baugruppe, mit eventuell neuerem Ausgabestand verstanden.

Hinweis

Beachten Sie bitte, dass Sie im Ersatzteilfall im Anwenderprogramm nur die für den projektierten CP-Typ zugelassenen Anweisungen verwenden.

Es wird empfohlen, für alle Baugruppentypen immer die aktuellen Versionen der Anweisungen zu verwenden. Bei älteren Baugruppentypen setzt diese Empfehlung voraus, dass Sie den für diesen Baugruppentyp aktuellen Firmware-Stand verwenden.

Weitere Informationen zum Ersatzteilfall finden Sie bei unserem Customer Support im Internet.

http://support.automation.siemens.com/WW/view/de/7806643

Die Gerätehandbücher geben Auskunft über die Kompatibilität der S7-CPs und der zugehörenden Anweisungen.

Aufrufe der Anweisungen parametrieren

Bevor die Anweisungen im Detail beschrieben werden, folgen an dieser Stelle einige allgemeine Hinweise zum Aufruf und zur Parametrierung der Anweisungen.

Allgemeine Angaben können hier zu folgenden, bei allen Anweisungen vorhandenen Parametergruppen gemacht werden:

- Parameter zur CP- und Verbindungszuordnung (Eingangsparameter)
- Parameter zur Angabe eines CPU-Datenbereichs (Eingangsparameter)
- Statusinformationen (Ausgangsparameter)

Kommunikations-Anweisungen für S7-300 aufrufen

ACHTUNG
Es ist nicht zulässig, die Kommunikations-Anweisungen für S7-300 in mehreren Ablaufebenen aufzurufen! Wenn Sie beispielsweise eine Kommunikations-Anweisung in OB1 und in OB35 aufrufen, könnte die Bearbeitung der Anweisung durch den jeweils höherprioren OB unterbrochen werden. Wenn Sie die Anweisungen in mehreren OBs aufrufen, müssen Sie programmtechnisch dafür sorgen, dass eine laufende Kommunikations-Anweisung nicht durch eine andere Kommunikations-Anweisungen unterbrochen wird (beispielsweise über System-Anweisungen Alarme sperren/freigeben).

Parameter zur Angabe eines CPU-Datenbereichs (Eingangsparameter)

Angabe des Datenbereiches in der CPU

Beim Aufruf einer Anweisung übergeben Sie die Adresse und Länge des Datenbereichs in der CPU, in dem Nutzdaten bereitgestellt oder abgelegt werden sollen oder weitere Parametrierinformationen enthalten sein können.

Zur Adressierung dieses Bereiches wird der Datentyp ANY-Zeiger verwendet.

Statusinformationen (Ausgangsparameter)

Statusanzeigen auswerten

Für die Statusauswertung sind im Anwenderprogramm die Parameter auszuwerten:

- DONE bzw. NDR
  Diese Parameter (DONE bei Sendaufträgen bzw. NDR bei Empfangsaufträgen) melden den (positiven) Abschluss einer Auftragsausführung.
- ERROR
  Meldet, wenn der Auftrag nicht fehlerfrei ausgeführt werden konnte.
- STATUS
  Der Parameter liefert Detailinformation zur Auftragsausführung. Statusanzeigen können bereits während der Auftragsausführung geliefert werden (DONE=0 und ERROR=0).

Hinweis

Beachten Sie, daß die Statusanzeigen DONE, NDR, ERROR, STATUS bei jedem Aufruf der Anweisung aktualisiert werden.

Statusanzeigen beim CP-Anlauf

Bei einem Neu-/Wiederanlauf des Ethernet-CP werden die Ausgabeparameter der Anweisung wie folgt zurückgesetzt:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS = 8180H bzw. 8181H

Anweisungen für die Offenen Kommunikationsdienste (SEND/RECEIVE-Schnittstelle)

Anweisungen für FDL-Verbindungen (SEND/RECEIVE-Schnittstelle)

Übersicht

Für die Übertragung von Daten über projektierte FDL-Verbindungen stehen folgende Anweisungen für die SEND/RECEIVE-Schnittstelle zur Verfügung:

Anweisung	verwendbar bei [1]		Bedeutung
S7-300	S7-400		
AG_SEND	x	x	für Daten Senden
AG_RECV	x	x	für Daten empfangen
AG_LSEND		x	für Daten Senden
AG_LRECV		x	für Daten empfangen

1) Anmerkungen zu den Anweisungen bei S7-300 und S7-400

Um die Kompatibilität von PROFIBUS und Industrial Ethernet an der Schnittstelle im Anwenderprogramm zu gewährleisten, können die Anweisungen AG_LSEND und AG_LRECV bei PROFIBUS alternativ zu AG_SEND bzw. AG_RECV verwendet werden. Es gibt keine Unterschiede an der Schnittstelle und im Verhalten. Bei PROFIBUS können Sie aber auch mit diesen Anweisungen, die bei Industrial Ethernet für die Übertragung langer Datensätze bestimmt sind, nur Datenmengen bis max. 240 Byte übertragen.

Voraussetzung ist, dass Typ und Version der Anweisungen für den verwendeten CP-Typ zugelassen sind.

Bei den S7-CPs für S7-300 werden ausschließlich die Anweisungen AG_SEND und AG_RECV verwendet; bei Industrial Ethernet auch für die Übertragung langer Datensätze.

Die Gerätehandbücher geben Auskunft über die Kompatibilität der S7-CPs und der zugehörenden Anweisungen. Eine Übersicht über die Versionen der Anweisungen finden Sie in der Dokumentations- und Bausteinhistorie.

Anwendung

Die folgende Darstellung verdeutlicht die Anwendung der Anweisungen AG_SEND / AG_LSEND und AG_RECV / AG_LRECV für den bidirektionalen Datentransfer über eine projektierte FDL-Verbindung.
Bei bestimmten Verbindungstypen ist im Anwenderdatenbereich ein Auftragsheader vorzusehen.

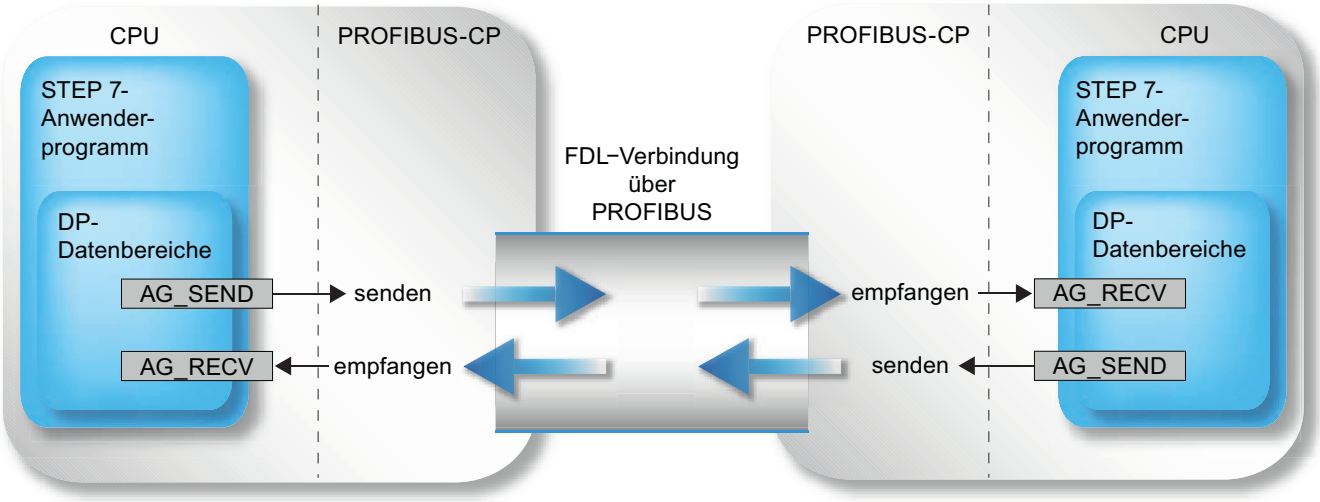

Bild 4-39 AG_SEND und AG_RECV bei beiden Kommunikationspartnern verwenden

Anwendung ohne Auftragsheader

Bei spezifizierter FDL-Verbindung sind Adress- und Auftragsparameter durch die Verbindungsprojektierung festgelegt. Das Anwenderprogramm stellt daher nur die Nutzdaten im FDL-Datenbereich beim Senden mit AG_SEND / AG_LSEND bereit, bzw. empfängt diese mit AG_RECV / AG_LRECV.

Es können bis zu 240 Byte Nutzdaten übertragen werden. Diese Angabe gilt bei PROFIBUS für AG_SEND und AG_LSEND.

Anwendung mit Auftragsheader

Folgende Verbindungstypen erfordern einen Auftragsheader im FDL-(Anwender-)Datenbereich:

- Unspezifizierte FDL-Verbindung mit freiem Layer 2 Zugang
- FDL-Verbindung mit Broadcast
- FDL-Verbindung mit Multicast

Entnehmen Sie der folgenden Darstellung die Struktur des Auftragspuffers und die Bedeutung und Platzierung der Parameter im Auftragsheader.

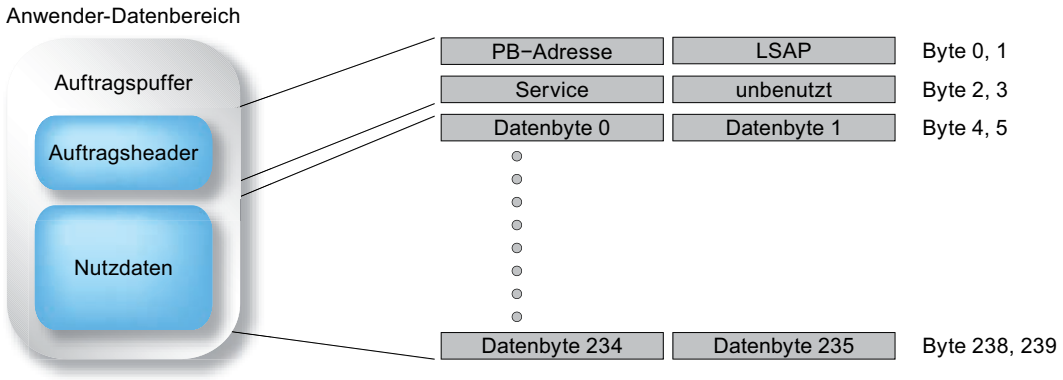

Bild 4-40 Senden und Empfangen über eine FDL-Verbindung mit Broadcast-Adressierung per Programm

Der Anwender-Datenbereich kann bis zu 240 Byte umfassen. Es können bis zu 236 Byte Nutzdaten übertragen werden. 4 Byte sind für den Auftragsheader reserviert.

Beachten Sie, dass die beim Aufruf der Anweisung angegebene Datenlänge (Parameter LEN) den Header und die Nutzdaten umfassen muss!

AG_SEND / AG_LSEND

AG_SEND / AG_LSEND (PROFIBUS)

Beschreibung

Die Anweisung AG_SEND / AG_LSEND übergibt Daten an den PROFIBUS-CP zur Übertragung über eine projektierte FDL-Verbindung.

Der angegebene Datenbereich kann ein PA-Bereich, ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn der gesamte Anwenderdatenbereich über PROFIBUS gesendet werden konnte.

Anmerkung:
Alle folgenden Angaben gelten, soweit nicht anders angegeben, gleichermaßen für AG_SEND und AG_LSEND.

Aufruf

Aufrufschnittstelle in FUP-Darstellung

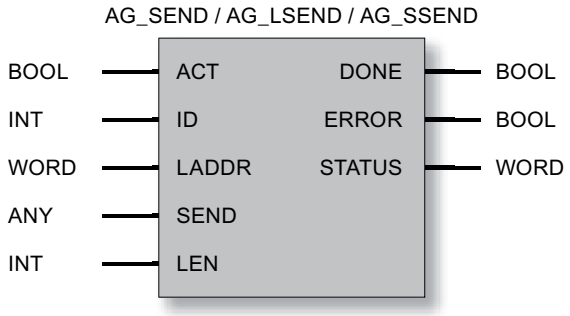

Aufrufe mit Auftragsheader

Entnehmen Sie der folgenden Tabelle, für welche Verbindungstypen und Auftragsarten Parameter im Auftragsheader zu versorgen sind.

Der Auftragsheader liegt im FDL-(Anwender-)Datenbereich. Er belegt dort die ersten 4 Byte und muss bei der Längenangabe im Parameter LEN hinzugerechnet werden. Die maximale Nutzdatenlänge reduziert sich daher bei Aufträgen mit Auftragsheader auf 236 Byte.

Tabelle 4-378 Versorgung des Auftrags-Header im Anwender-Datenbereich

Parameter	FDL-Verbindungstyp		
unspezifiziert: freie Layer 2 [2]	**Broadcast**	**Multicast**	
PB-Adresse	Adresse der Zielstation Wertebereich: 0..126 je nach Teilnehmer / 127 für Broadcast/Multicast	bei AG_SEND ohne Bedeutung; Bereich aber zu reservieren.	bei AG_SEND ohne Bedeutung; Bereich aber zu reservieren.
LSAP	LSAP der Zielstation Wertebereich: 0..62 je nach Teilnehmer / 63 für Broadcast	ohne Bedeutung; Bereich aber zu reservieren.	ohne Bedeutung; Bereich aber zu reservieren.
Service [1]	SDA (Send Data with Acknowledge): Wert: 00_H SDN (Send Data with No Acknowledge): Wert: 01_H	ohne Bedeutung; Bereich aber zu reservieren.	ohne Bedeutung; Bereich aber zu reservieren.

[1] für Broadcast und Multicast ist nur der Service SDN möglich.

[2] Die Angaben zu Broadcast und Multicast in dieser Spalte sind nur für den Fall relevant, dass eine unspezifizierte FDL-Verbindung für Broadcast oder Multicast verwendet wird. Bei einer projektierten FDL-Verbindung (empfohlene Anwendung) mit Broadcast oder Multicast als Verbindungspartner werden die Adressparameter entsprechend der Projektierung automatisch zugewiesen.

Siehe auch

Parameter bei AG_SEND / AG_LSEND (PB) (Seite 6924)

Parameter DONE, ERROR, STATUS (PB) (Seite 6925)

Beispiel für AG_SEND (PB) (Seite 6927)

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Seite 6777)

Parameter bei AG_SEND / AG_LSEND (PB)

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung AG_SEND / AG_LSEND:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ACT	INPUT	BOOL	0,1	Beim Aufruf der Anweisung mit ACT = 1 werden LEN Bytes aus dem mit dem Parameter SEND angegebenen Datenbereich gesendet. Beim Aufruf der Anweisung mit ACT = 0 werden die Statusanzeigen DONE, ERROR und STATUS aktualisiert.
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	Im Parameter ID wird die Verbindungsnummer der FDL-Verbindung angegeben.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
SEND	INPUT	ANY (als VAR-TYPE sind nur zugelassen: BYTE, WORD und DWORD		Angabe von Adresse und Länge Die Adresse des Datenbereiches verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich Bei Aufruf mit Auftragsheader enthält der FDL-Datenbereich den Auftragsheader und die Nutzdaten.
LEN	INPUT	INT	1,2,...240 (bzw. bis "Längenangabe beim Parameter SEND")	Anzahl der Bytes, die mit dem Auftrag aus dem FDL-Datenbereich gesendet werden sollen. Die Angabe kann im Bereich von 1 bis "Längenangabe beim Parameter SEND" liegen. Bei Aufruf mit Auftragsheader setzt sich die Längenangabe aus dem Auftragsheader (4 Byte) + Nutzdaten (1..236 Byte) zusammen. Es gilt daher LEN >= 4 !
DONE	OUTPUT	BOOL	0: - 1: neue Daten	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe nachfolgende Tabelle.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe nachfolgende Tabelle.
STATUS	OUTPUT	WORD	siehe nachfolgende Tabelle	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe nachfolgende Tabelle.

Siehe auch

AG_SEND / AG_LSEND (PROFIBUS) (Seite 6922)

Parameter DONE, ERROR, STATUS (PB)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung 8FxxH unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-379 Anzeigen AG_SEND

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fertig ohne Fehler.
0	0	0000_H	Kein Auftrag in Bearbeitung.
0	0	8181_H	Auftrag läuft.
0	1	7000_H	Die Anzeige ist nur bei S7-400 möglich: Die Anweisung wurde mit ACT=0 aufgerufen; der Auftrag wird jedoch nicht bearbeitet.
0	1	8183_H	Die Projektierung fehlt oder der FDL-Dienst im PROFIBUS-CP ist noch nicht gestartet.
0	1	8184_H	Mögliche Ursachen: • Unzulässiger Datentyp für den Parameter SEND angegeben. • FDL-Verbindung ohne Auftragspuffer: Systemfehler. • FDL-Verbindung mit Auftragspuffer: Parameter LEN<4 oder unzulässiger Parameter im Auftragsheader (bei freiem Layer 2 Zugang).
0	1	8185_H	Parameter LEN größer als Quell-Bereich SEND.
0	1	8186_H	Parameter ID ungültig. ID != 1,2....15,16.
0	1	8301_H	SAP bei Ziel-Station nicht aktiviert.

DONE	ERROR	STATUS	Bedeutung
0	1	8302_H	keine Empfangsressourcen bei Ziel-Station, Empfänger-Station kann empfangene Daten nicht schnell genug verarbeiten bzw. hat keine Empfangsressourcen bereitgestellt.
0	1	8303_H	Der PROFIBUS-Service (SDA-Send Data with Acknowledge) wird auf diesem SAP von der Ziel-Station nicht unterstützt. Die Anzeige kann auch temporär auftreten, wenn Verbindungen oder Netzübergänge "im RUN" geladen werden.
0	1	8304_H	Die FDL-Verbindung ist nicht aufgebaut.
0	1	8311_H	Die Zielstation ist unter der angegebenen PROFIBUS-Adresse nicht erreichbar oder der benutzte Service ist für die angegebene PROFIBUS-Adresse nicht möglich.
0	1	8312_H	PROFIBUS-Fehler im CP : z. B. Buskurzschluss, eigene Station nicht am Ring,.
0	1	8315_H	Mögliche Ursachen: • Interner Parameterfehler bei einer FDL-Verbindung mit Auftragsheader: Parameter LEN<4 oder unzulässiger Parameter im Auftragsheader (bei freiem Layer 2 Zugang). • Busstörung Mögliche zusätzliche Bedeutung: • Die Fehleranzeige kann auch bei Busstörungen auftreten (beispielsweise bei physikalischen Störungen aufgrund fehlerhafter Leistungsabschlüsse oder unterschiedlicher Einstellungen der Übertragungsgeschwindigkeit bei den Teilnehmern).
0	1	$8F22_H$	Quell-Bereich ungültig. z. B.: Bereich im DB nicht vorhanden Parameter LEN < 0
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummer Fehler.
0	1	$8F3A_H$	Bereich nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F44_H$	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F7F_H$	Interner Fehler. z. B. unzulässige ANY-Referenz z. B. Parameter LEN = 0 .
0	1	8090_H	• Eine Baugruppe mit dieser Baugruppen-Anfangsadresse ist nicht vorhanden. • Die verwendete Anweisung passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Anweisungen für S7-300 und S7-400 zu verwenden).
0	1	8091_H	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	1	8092_H	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben. (nur bei S7-400)
0	1	$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut. (bei neueren CPU-Ausgabeständen). Dies kann beispielsweise begründet sein durch: • eine fehlende Projektierung von Verbindungen; • eine Überschreitung der maximalen Anzahl parallel betreibbarer CPs (Angaben hierzu siehe CP-Gerätehandbuch).
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.

DONE	ERROR	STATUS	Bedeutung
0	1	$80B1_H$	Der Ziel-Bereich ist ungültig. Die Anzahl der zu sendenden Daten überschreitet die für diesen Dienst zulässige Obergrenze (z. B. Ziel-Bereich > 240 Bytes).
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.(bei älteren CPU-Ausgabeständen; sonst $80A4_H$; weitere Angaben siehe dort)
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung..
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.
0	1	$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_SEND / AG_LSEND (PROFIBUS) (Seite 6922)

Beispiel für AG_SEND (PB)

Beispiel für Aufruf und Parameterauswertung

Nachfolgend finden Sie ein ablauffähiges Beispiel für den Aufruf und die Parameterauswertung einer Anweisung AG_SEND.

Zur hier gewählten Anweisung "AE_460_1", in der der Sendeaufruf erfolgt, gehört der unten aufgeführte OB100; der OB100 stellt das ACT-Bit beim Anlauf der CPU korrekt ein.

Zur korrekten Funktion muss ein DB100 mit der Größe von mindestens 240 Byte geladen sein.

Das Programm setzt einen CP auf Adresse 256, sowie eine projektierte Verbindung vom Typ ISO-Transport / ISO-on-TCP / TCP- oder FDL mit der ID=1 voraus (bitte ggf. anpassen !).

```
//-------------------------------------------------------------------
---
FUNCTION FC 100: VOID
TITLE = SENDE_DEMO
AUTHOR : Tester
FAMILY : S7300
NAME : FC5_Demo
VERSION : 1.0
//-------------------------------------------------------------------
---
BEGIN
    CALL FC 5 (
    ACT        := M100.0,
    ID            := 1,
    LADDR        := W#16#100,
    SEND        := P#DB100.dbx0.0 BYTE 240,
    LEN        := 240,
    DONE        := M100.1,
    ERROR    := M100.2,
    STATUS    := MW102 );
//-------------------------------------------------------------------
---
    R M100.0;          //   Reset Parameter ACT für alle weiteren
    SET;               //   Aufrufe von AG_SEND
    U M100.1;          //
    SPB done;          //   Test auf DONE = TRUE
    SET;               //
    U M100.2;          //
    SPB err;           //   Test auf ERROR = TRUE
//-------------------------------------------------------------------
---
    BEA;               //   Weder DONE noch ERROR sind gesetzt; der
                       //   Auftrag läuft noch.
//-------------------------------------------------------------------
---
done:    S M100.0;     //   Auftrag ist ohne Fehler beendet. Setze
    BEA;               //   ACT = TRUE, damit der folgende Aufruf den
                            neuen Auftrag anstossen kann.
//-------------------------------------------------------------------
---
```

```
err:    NOP 1;          //   Ein Fehler ist aufgetreten. Hier kann
    NOP 1;              //   eine Auswertung des Statuswortes
    S M100.0;           //   erfolgen. Auf jeden Fall ACT auf TRUE
    BEA;                //   setzen, damit ein neuer Sendeauftrag
                             angestossen werden kann, falls der Fehler
                             verschwindet.
//-------------------------------------------------------------------
---
END_FUNCTION
ORGANIZATION_BLOCK OB100
TITLE         = Init_for_FC100
FAMILY:    S7300
NAME:        SENDE_DEMO_INIT
VERSION:    1.0
VAR_TEMP
OB1_System: array [1..20] of byte;
END_VAR
//-------------------------------------------------------------------
---
BEGIN                   //
    SET                 //
    S M100.0            //   Initialisiere den Parameter ACT
END_ORGANIZATION_BLOCK
```

AG_RECV / AG_LRECV

AG_RECV / AG_LRECV (PROFIBUS)

Beschreibung

Die Anweisung AG_RECV / AG_LRECV übernimmt vom PROFIBUS-CP die über eine projektierte FDL-Verbindung übertragenen Daten.

Der für die Datenübernahme angegebene Datenbereich kann ein PA-Bereich, ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn die Daten vom PROFIBUS-CP übernommen werden konnten.

Anmerkung:

Alle folgenden Angaben gelten, soweit nicht anders angegeben, gleichermaßen für AG_RECV und AG_LRECV.

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

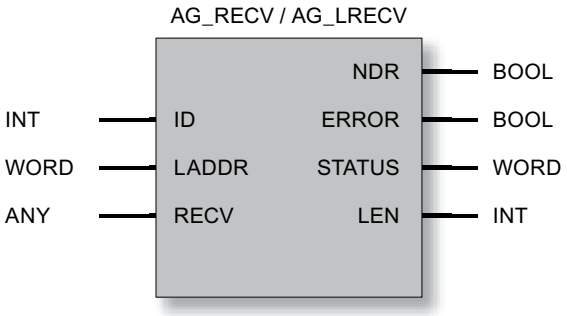

Aufrufe mit Auftrags-Header

Tabelle 4-380 Rückgabeparameter im Auftragsheader im FDL-(Anwender-)Datenbereich

Parameter	FDL-Verbindungstyp		
unspezifiziert: freie Layer 2)	Broadcast	Multicast	
PB-Adresse	Adresse des Senders Wertebereich: 0..126 je nach Teilnehmer		
LSAP	LSAP des Senders Wertebereich: 0..63 je nach Teilnehmer		
Service	SDN-Anzeige (Send Data with No Acknowledge - Indication): Wert: 01_H oder SDA-Anzeige (Send Data with Acknowledge - Indication): Wert: 00_H	SDN-Anzeige (Send Data with No Acknowledge - Indication): Wert: $7F_H$	SDN-Anzeige (Send Data with No Acknowledge - Indication): Wert: $7F_H$

Siehe auch

Parameter bei AG_RECV / AG_LRECV (PB) (Seite 6931)

Parameter DONE, ERROR, STATUS (PB) (Seite 6932)

AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Seite 6782)

Parameter bei AG_RECV / AG_LRECV (PB)

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung AG_RECV / AG_LRECV:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ID	INPUT	INT	1,2...16 (S7-300) 1,2...32 (S7-400)	Im Parameter ID wird die Verbindungsnummer der FDL-Verbindung angegeben.
LADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
RECV	INPUT	ANY (als VARTYPE sind nur zugelassen: BYTE, WORD und DWORD		Angabe von Adresse und Länge Die Adresse des FDL-Datenbereiches verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich Bei Aufruf mit Auftragsheader enthält der FDL-Datenbereich den Auftragsheader und die Nutzdaten.
LEN	OUTPUT	INT	1,2,...240	Gibt die Anzahl der Bytes an, die vom PROFIBUS-CP in den FDL-Datenbereich übernommen wurden. Bei Aufruf mit Auftragsheader setzt sich die Längenangabe aus dem Auftragsheader (4 Byte) + Nutzdaten (1..236 Byte) zusammen. Es gilt daher LEN >= 4 !
NDR	OUTPUT	BOOL	0: - 1: neue Daten	Der Parameter zeigt an, ob neue Daten übernommen wurden. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe nachfolgende Tabelle.
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und STATUS siehe nachfolgende Tabelle.
STATUS	OUTPUT	WORD	siehe nachfolgende Tabelle	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und ERROR siehe nachfolgende Tabelle.

Siehe auch

AG_RECV / AG_LRECV (PROFIBUS) (Seite 6929)

Parameter DONE, ERROR, STATUS (PB)

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus NDR, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-381 Anzeigen AG_RECV / AG_LRECV

NDR	ERROR	STATUS	Bedeutung
1	0	0000_H	Neue Daten übernommen.
0	0	8180_H	• Es liegen noch keine Daten vor. Die Projektierung fehlt oder der FDL-Dienst im PROFIBUS-CP ist noch nicht gestartet (tritt anstelle der Anzeige 0,1,8183_H auf !).
0	0	8181_H	Auftrag läuft.
0	1	8183_H	Die Projektierung fehlt oder der FDL-Dienst im PROFIBUS-CP ist noch nicht gestartet.
0	1	8184_H	• Unzulässiger Datentyp für den Parameter RECV angegeben. • Systemfehler.
0	1	8185_H	Ziel-Puffer (RECV)ist zu klein.
0	1	8186_H	Parameter ID ungültig. ID != 1,2....15,16.
0	1	8303_H	Der PROFIBUS-Service (SDA-SendDatawithAcknowledge) wird auf diesem SAP nicht unterstützt. Die Anzeige kann auch temporär auftreten, wenn Verbindungen oder Netzübergänge "im RUN" geladen werden.
0	1	8304_H	Die FDL-Verbindung ist nicht aufgebaut.
0	1	$8F23_H$	Quell-Bereich ungültig. z. B.: Bereich im DB nicht vorhanden.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummer Fehler.
0	1	$8F3A_H$	Zielbereich nicht geladen (DB).
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F45_H$	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.

NDR	ERROR	STATUS	Bedeutung
0	1	$8F7F_H$	Interner Fehler. z. B. Unzulässige ANY-Referenz.
0	1	8090_H	• Eine Baugruppe mit dieser Baugruppen-Anfangsadresse ist nicht vorhanden. • Die verwendete Anweisung passt nicht zur verwendeten Systemfamilie (es sind unterschiedliche Anweisungen für S7-300 und S7-400 zu verwenden).
0	1	8091_H	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	1	8092_H	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben. (nur bei S7-400)
0	1	$80A0_H$	Negative Quittung beim Lesen von Baugruppe.
0	1	$80A4_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut. (bei neueren CPU-Ausgabeständen). Dies kann beispielsweise begründet sein durch: • eine fehlende Projektierung von Verbindungen; • eine Überschreitung der maximalen Anzahl parallel betreibbarer CPs (Angaben hierzu siehe CP-Gerätehandbuch).
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Mögliche Ursachen: • Der Zielbereich ist ungültig. • Der Zielbereich ist zu klein. • Der Zielbereich für die Empfangsdaten wurde nicht ausreichend bemessen. Abhilfe: Führen Sie einen weiteren Empfangsaufruf mit maximaler Empfangspuffergröße durch. Dies gilt unabhängig vom Verbindungstyp (Uni-/Multi-/Broadcast) und von der Gerätefamilie (S7-300 / S7-400).
0	1	$80B2_H$	Die K-Busverbindung zwischen CPU und CP ist nicht aufgebaut.
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	80C3H	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll).
0	1	$80D2_H$	Baugruppen-Anfangsadresse ist falsch.

Siehe auch

AG_RECV / AG_LRECV (PROFIBUS) (Seite 6929)

Anweisungen für DP (Dezentrale Peripherie) bei S7-300

Anweisungen für die DP-Betriebsart S7-300

Übersicht

Für die DP-Betriebsarten DP-Master und DP-Slave stehen folgende Anweisungen für S7-300 zur Verfügung:

Anweisung	verwendbar bei		Bedeutung
	DP-Master	DP-Slave	
DP_SEND	X	X	für Daten Senden
DP_RECV	X	X	für Daten empfangen
DP_DIAG	X	-	für Diagnosefunktionen vom DP-Master aus
DP_CTRL	X	-	für Steuerfunktionen

Anwendung

Die folgende Darstellung verdeutlicht die Anwendung der Anweisungen DP_SEND und DP_RECV beim DP-Master und beim DP-Slave.

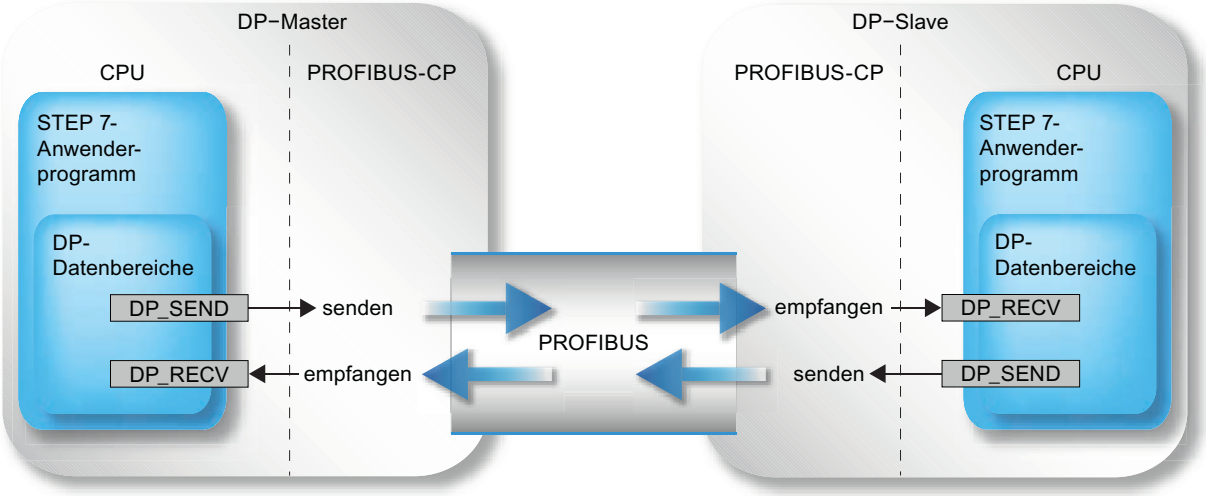

Bild 4-41 Verwendung der Anweisungen DP_SEND und DP_RECV bei DP-Master und DP-Slave

DP_SEND

DP_SEND

Beschreibung

Die Anweisung DP_SEND überträgt Daten zum PROFIBUS-CP. Je nach Betriebsart des PROFIBUS-CP hat DP_SEND folgende Bedeutung:

- Bei Verwendung im DP-Master

Die Anweisung übergibt die Daten eines angegebenen DP-Ausgabebereiches an den PROFIBUS-CP zur Ausgabe an die Dezentrale Peripherie.

- Bei Verwendung im DP-Slave

Die Anweisung übergibt die Eingangsdaten des DP-Slave an den PROFIBUS-CP zur Übertragung an den DP-Master

Der angegebene Datenbereich kann ein PA-Bereich, ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn der gesamte DP-Datenbereich vom PROFIBUS-CP übernommen werden konnte.

Zum Starten des DP-Masters ist genau ein Aufruf von DP-SEND oder DP-RECV in der Aufruffolge voranzustellen. Für diesen Erstaufruf gilt:

Wird zur Initialisierung der DP-SEND verwendet, wird der dabei übergebene Datenbereich nicht übernommen und "0" an die Slaves gesendet. Erst der zweite Bausteinaufruf sendet die übergebenen Nutzdaten.

Aufrufschnittstelle

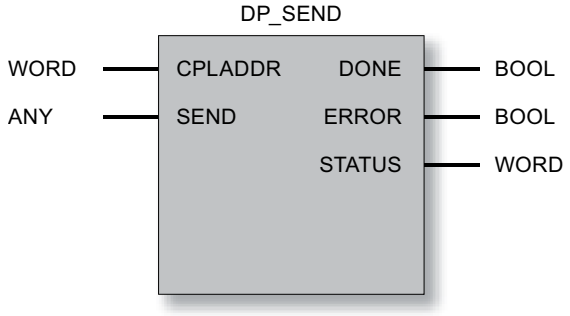

Siehe auch

Parameter bei DP_SEND (Seite 6936)

Parameter DONE, ERROR, STATUS (Seite 6937)

Parameter bei DP_SEND

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung DP_SEND:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
SEND	INPUT	ANY (als VARTYPE sind nur zugelassen: bei DP_SEND ab V3: BYTE bei DP_SEND bis V2.x: BYTE, WORD und DWORD)		Angabe von Adresse und Länge Die Adresse des DP-Datenbereichs verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich Die Länge ist einzustellen für • DP-Master: 1...21600 • DP-Slave: 1...240
DONE	OUTPUT	BOOL	0: - 1: neue Daten	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6937).
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6937).
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe Parameter DONE, ERROR, STATUS (Seite 6937).

Siehe auch

DP_SEND (Seite 6935)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-382 Anzeigen DP_SEND

DONE	ERROR	STATUS	Bedeutung
0	0	8180_H	• Anlauf: Der DP-Dienst wurde gestartet aber die Datenübernahme ist noch nicht möglich. • Normalbetrieb Die Datenübergabe läuft. • DP ist nicht gestartet wegen folgender Situation: – CP-STOP oder – "keine Parametrierung (tritt hier anstelle der Anzeige 0,1,8183_H auf)
1	0	0000_H	Neue Daten fehlerfrei übergeben.
0	1	8183_H	Projektierung fehlt oder DP-Dienst im PROFIBUS-CP noch nicht gestartet.
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp.
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummernfehler.
0	1	$8F3A_H$	Zielbereich nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F44_H$	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F45_H$	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F7F_H$	Interner Fehler. z. B. Unzulässige ANY-Referenz.

DONE	ERROR	STATUS	Bedeutung
0	1	8090_H	Baugruppe mit dieser Adresse nicht vorhanden.
0	1	8091_H	Logische Basisadresse nicht auf Doppel-Wort-Raster.
0	1	$80A1_H$	Negative Quittung beim Schreiben zur Baugruppe.
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Die Anzahl der zu sendenden Daten überschreitet die für diesen Dienst zulässige Obergrenze (gilt für den DP-Master- und DP-Slave-Betrieb).
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)
0	1	$80D2_H$	Logische Basisadresse falsch.

Siehe auch

DP_SEND (Seite 6935)

DP_RECV

DP_RECV

Beschreibung

Die Anweisung DP_RECV empfängt Daten über PROFIBUS. Je nach Betriebsart des PROFIBUS-CP hat DP_RECV folgende Bedeutung:

- Bei Verwendung im DP-Master
  DP_RECV übernimmt die Prozessdaten der dezentralen Peripherie sowie eine Statusinformation in einen angegebenen DP-Eingabebereich.
- Bei Verwendung im DP-Slave
  DP_RECV übernimmt die vom DP-Master übertragenen Ausgangsdaten in den an der Anweisung angegebenen DP-Datenbereich.

Der für die Datenübernahme angegebene Datenbereich kann ein PA-Bereich, ein Merkerbereich oder ein Datenbausteinbereich sein.

Eine fehlerfreie Ausführung wird signalisiert, wenn der gesamte DP-Dateneingabebereich vom PROFIBUS-CP übergeben werden konnte.

Beachten Sie, dass die Anweisung DP_RECV beim DP-Slave im Anwenderprogramm mindestens einmal erfolgreich aufgerufen werden muss, falls für diesen DP-Slave Ausgangsdaten projekiert wurden. Bitte beachten Sie die Angaben im Gerätehandbuch.

Zum Starten des DP-Masters ist genau ein Aufruf von DP-SEND oder DP-RECV in der Aufruffolge voranzustellen. Für diesen Erstaufruf gilt:

- Wird zur Initialisierung der DP-RECV verwendet, werden die empfangenen Daten nicht übernommen. Erst der zweite Bausteinaufruf liefert die empfangenen Nutzdaten.

Zusatzaufgabe: Statusbyte eintragen

Die Anweisung DP_RECV hat folgende zusätzliche Aufgabe:

- Aktualisieren des DP-Statusbyte DPSTATUS. DP_RECV übernimmt damit Aufgaben für die DP_Diagnose.
  Falls keine Empfangsdaten projektiert sind, muss der DP_RECV zur Aktualisierung des Statusbyte DPSTATUS mit der Länge 1 aufgerufen werden (gilt nur beim DP-Master). Beachten Sie auch die Angaben im Gerätehandbuch.
- Freigeben der Stationsliste (siehe DP_DIAG (Seite 6944)).

Aufrufschnittstelle

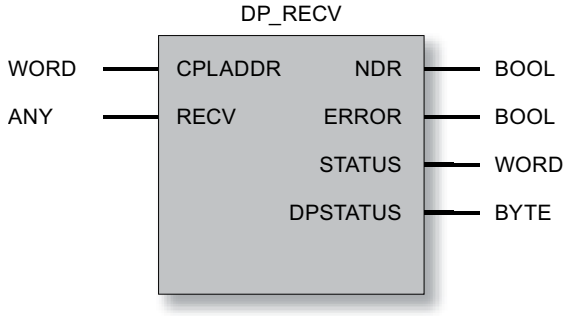

Siehe auch

Parameter bei DP_RECV (Seite 6940)

Parameter NDR, ERROR, STATUS (Seite 6941)

Parameter DPSTATUS - DP_RECV (Seite 6942)

Parameter bei DP_RECV

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung DP_RECV:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
RECV	INPUT	ANY (als VARTYPE sind nur zugelassen: bei DP_RECV ab V3: BYTE bei DP_RECV bis V2.x: BYTE, WORD und DWORD)		Angabe von Adresse und Länge Die Adresse des DP-Datenbereiches verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich Die Länge ist einzustellen für: • DP-Master: 1...2160 • DP-Slave: 1...240 • DP-Master; nur Statusbyte lesen: 1 (siehe auch CP-Gerätehandbuch)
NDR	OUTPUT	BOOL	0: - 1: neue Daten übernommen	Der Zustandsparameter zeigt an, ob neue Daten übernommen wurden. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Parameter NDR, ERROR, STATUS (Seite 6941).
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und STATUS siehe Parameter NDR, ERROR, STATUS (Seite 6941)
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und ERROR siehe Parameter NDR, ERROR, STATUS (Seite 6941)
DPSTATUS	OUTPUT	Byte	Codierung siehe nachfolgend unter DPSTATUS	DP-Statusanzeige

Siehe auch

DP_RECV (Seite 6938)

Parameter NDR, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus NDR, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

NDR	ERROR	STATUS	Bedeutung
0	0	8180_H	• Anlauf: Der DP-Dienst wurde gestartet aber die Datenübernahme ist noch nicht möglich. • Normalbetrieb Die Datenübergabe läuft. • DP ist nicht gestartet wegen folgender Situation: – CP-STOP oder – "keine Parametrierung" (tritt hier anstelle der Anzeige 0,1,8183_H auf).
1	0	0000_H	Neue Daten fehlerfrei übernommen.
0	1	8183_H	Projektierung fehlt oder DP-Dienst im PROFIBUS-CP noch nicht gestartet.
0	1	8184_H	Systemfehler bzw. unzulässiger Parametertyp.
0	1	$8F22_H$	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).
0	1	$8F23_H$	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz).
0	1	$8F24_H$	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein.
0	1	$8F31_H$	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein.
0	1	$8F32_H$	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	DB-Nummernfehler.
0	1	$8F3A_H$	Zielbereich nicht geladen (DB).
0	1	$8F42_H$	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	1	$8F44_H$	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F45_H$	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.
0	1	$8F7F_H$	Interner Fehler. z. B. Unzulässige ANY-Referenz.
0	1	8090_H	Baugruppe mit dieser Adresse nicht vorhanden.

NDR	ERROR	STATUS	Bedeutung
0	1	8091_H	Logische Basisadresse nicht auf Doppel-Wort-Raster.
0	1	$80A0_H$	Negative Quittung beim Lesen von der Baugruppe.
0	1	$80B0_H$	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	Die Anzahl der zu sendenden Daten überschreitet die für diesen Dienst zulässige Obergrenze (gilt für den DP-Master- und DP-Slave-Betrieb).
0	1	$80C0_H$	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$	Betriebsmittel (Speicher) belegt.
0	1	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll).
0	1	$80D2_H$	Logische Basisadresse falsch.

Siehe auch

DP_RECV (Seite 6938)

Parameter DPSTATUS - DP_RECV

DPSTATUS

Die Codierung des Ausgabeparameters DP-Status ist für die Betriebsarten DP-Master-Betrieb und DP-Slave-Betrieb unterschiedlich.

DP-Master-Betrieb

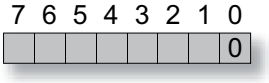

Tabelle 4-383 Bedeutung der Bits in DPSTATUS im DP-Master-Betrieb

Bit	Bedeutung
7	nicht belegt
6	Dieses Bit wird nicht gesetzt. Bitte beachten Sie auch die Angaben im Gerätehandbuch.
5,4	Werte für DPSTATUS des DP-Masters: 00 RUN 01 CLEAR 10 STOP (wird auf den Betriebszustand OFFLINE abgebildet) 11 OFFLINE Bitte beachten Sie auch die Angaben im Gerätehandbuch.
3	Wert 1: Zyklische Synchronisation ist eingeschaltet
2	Wert 0: keine neuen Diagnosedaten vorhanden Wert 1: Diagnoseliste auswerten ist sinnvoll; mindestens 1 Station hat neue Diagnosedaten

Bit	Bedeutung
1	Wert 0: alle DP-Slaves in der Datentransferphase Wert 1: Stationsliste auswerten ist sinnvoll
0	DP Betrieb Wert 0: DP-Master-Betrieb Die anderen Bits sind mit der angegebenen Bedeutung nur gültig, wenn dieses Bit nicht gesetzt ist.

DP-Slave-Betrieb

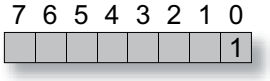

Tabelle 4-384 Bedeutung der Bits in DPSTATUS im DP-Slave-Betrieb

Bit	Bedeutung
7-5	nicht belegt
4	Dieses Bit wird nicht gesetzt. Bitte beachten Sie auch die Angaben im Gerätehandbuch.
3	Dieses Bit wird nicht gesetzt. Bitte beachten Sie auch die Angaben im Gerätehandbuch.
2	Wert 1: Der DP-Master 1 ist im Zustand CLEAR. Der DP-Slave empfängt in den für die Ausgänge bestimmten DP-Daten für alle Daten den Wert 0. Es besteht kein Einfluss auf die Sendedaten.
1	Wert 1: Die Konfigurierung / Parametrierung ist noch nicht erfolgreich beendet.
0	Wert 1: DP Slave-Betrieb. **Die anderen Bits sind mit der angegebenen Bedeutung nur gültig, wenn dieses Bit gesetzt ist.**

Hinweis

Beachten Sie bitte, dass DPSTATUS erst ausgewertet werden darf, wenn der Rückgabeparameter NDR=1 gesetzt ist.

Siehe auch

DP_RECV (Seite 6938)

DP_DIAG

DP_DIAG

Beschreibung

Die Anweisung DP_DIAG wird zum Anfordern von Diagnoseinformationen verwendet. Es werden folgende Auftragsarten unterschieden:

- DP-Stationsliste anfordern;
- DP-Diagnoseliste anfordern;
- DP-Einzelstatus anfordern;
- Eingangs- / Ausgangsdaten eines DP-Slave azyklisch lesen;
- Ältere DP-Einzeldiagnose lesen;
- DP-Betriebszustand lesen.
- DP-Betriebszustand für AG-/CP-STOP lesen;
- Aktuellen Zustand des DP-Slave lesen.

Diagnosedaten können durch Angabe einer Stationsadresse Slave-spezifisch angefordert werden.

Für die Übergabe der Diagnosedaten an die CPU ist ein Speicherbereich in der CPU vorzusehen und im Aufruf anzugeben. Dieser Speicherbereich kann ein Datenbausteinbereich oder ein Merkerbereich sein. Im Auftrag ist die maximale Länge des verfügbaren Speicherbereiches anzugeben.

Hinweis

Die Anweisung DP_DIAG ist nur in der DP-Betriebsart mit DP-Master sinnvoll.

Ausschluss

Solange die Anweisung läuft, darf sie nicht mit neuen Auftragsdaten versorgt werden.

Ausnahme: DP-Stationsliste oder DP-Diagnoseliste anfordern.

Aufrufschnittstelle

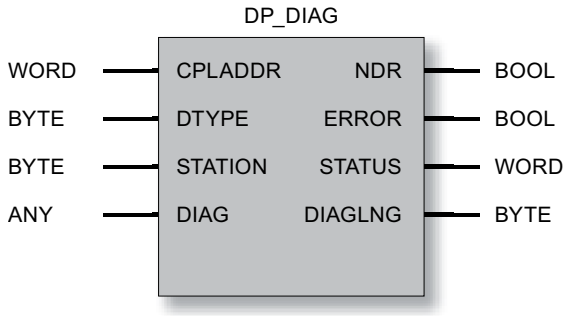

Siehe auch

Parameter bei DP_DIAG (Seite 6945)
Auftragsarten DP_DIAG (Seite 6947)
Ringpuffer für Diagnosedaten - DP_DIAG (Seite 6949)
Parameter NDR, ERROR, STATUS (Seite 6949)

Parameter bei DP_DIAG

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung DP_DIAG:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
DTYPE	INPUT	BYTE	0: Stationsliste 1: Diagnoseliste 2: Aktuelle Diagnose 3: Ältere Diagnose 4: Betriebszustand lesen 5: Betriebszustand für CPU-STOP lesen 6: Betriebszustand für CP-STOP lesen 7: Eingangsdaten (azyklisch) lesen 8: Ausgangsdaten (azyklisch) lesen 10: Aktuellen Zustand des DP-Slave lesen	Diagnosetyp
STATION	INPUT	BYTE		Stationsadresse des DP-Slaves

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
DIAG	INPUT	ANY (als VAR-TYPE sind nur zugelassen: BYTE, WORD und DWORD)	Die Länge ist einzustellen von 1...240	Angabe von Adresse und Länge Adresse des Datenbereiches. Verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich
NDR	OUTPUT	BOOL	0: - 1: neue Daten	Der Zustandsparameter zeigt an, ob neue Daten übernommen wurden. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Parameter NDR, ERROR, STATUS (Seite 6949)
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und STATUS siehe Parameter NDR, ERROR, STATUS (Seite 6949)
STATUS	OUTPUT	WORD	siehe Liste	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern NDR und ERROR siehe Parameter NDR, ERROR, STATUS (Seite 6949)
DIAGLNG	OUTPUT	BYTE	siehe Liste	Enthält die tatsächliche Länge (in Byte) der vom PROFIBUS-CP bereitgestellten Daten, unabhängig von der im Parameter DIAG angegebenen Puffergröße. Bei den Auftragsarten mit DTYPE 4, 5 und 6 gilt: DIAGLNG wird hier immer mit dem Wert "1" belegt. Der im Parameter DIAG zurückgegebene Wert ist für die Auswertung in diesen Fällen nicht relevant. In diesen Fällen ist der relevante Wert im Parameter STATUS enthalten.

Siehe auch

DP_DIAG (Seite 6944)

Auftragsarten DP_DIAG

Auftragsarten

Zulässige bzw. sinnvolle Auftragsangaben ergeben sich gemäß nachfolgender Übersicht aus der Angabe für DTYPE, STATION und DIAGLNG.

Tabelle 4-385 Auftragsarten für DP_DIAG

DTYPE	Entspricht Auftrag	Parameter STATION	DIAGLNG	Quittungscode (enthalten im Parameter STATUS; angegeben in Tabelle "Anzeigen DP_DIAG")
0	DP-Stationsliste lesen	---	- wird nicht berücksichtigt -	Mit der DP-Stationsliste erhalten Sie im Anwenderprogramm Informationen zum Zustand und zur Verfügbarkeit von DP-Slaves. Die Information in der DP-Stationsliste bezieht sich auf sämtliche durch Projektierung dem DP-Master zugeordnete DP-Slaves.
1	DP-Diagnoseliste lesen	---	- wird nicht berücksichtigt -	Mit der DP-Diagnoseliste erhalten Sie im CPU-Programm die Information, bei welchen DP-Slaves neue Diagnosedaten vorliegen.
2	Aktuelle DP-Einzeldiagnose lesen	1...126	>=6	Mit der aktuellen DP-Einzeldiagnose erhalten Sie im CPU-Programm die aktuellen Diagnosedaten eines DP-Slave.
3	Ältere DP-Einzeldiagnose lesen	1...126	>=6	Mit der älteren DP-Einzeldiagnose erhalten Sie im CPU-Programm die älteren Diagnosedaten eines DP-Slave. Diese Daten sind im PROFIBUS-CP gespeichert und werden nach dem Prinzip "Last in first out" im Ringpuffer gelesen. Zur Erläuterung des Ringpuffers siehe unten. Damit können bei schnellen Änderungen der DP-Slave-Diagnosedaten diese DP-Slaves erfasst und im CPU-Programm des DP-Masters ausgewertet werden.
4	Mit DP-CTRL-Auftrag (CTYPE=4) angeforderten Betriebszustand lesen		=1	Mit diesem Auftrag kann der DP-Betriebszustand gelesen werden, der mittels DP-CTRL-Auftrag (CTYPE=4) zuvor gesetzt wurde. Anmerkung: Der ausgelesene Betriebszustand muss nicht dem aktuellen Betriebszustand des CP entsprechen. Folgende Betriebszustände sind möglich: : • RUN • CLEAR • STOP (wird auf den Betriebszustand OFFLINE abgebildet) *) • OFFLINE

DTYPE	Entspricht Auftrag	Parameter STATION	DIAGLNG	Quittungscode (enthalten im Parameter STATUS; angegeben in Tabelle "Anzeigen DP_DIAG")
5	DP-Betriebszustand für CPU-STOP lesen		=1	Mit diesem Auftrag ermitteln Sie, in welchen DP-Betriebszustand der PROFIBUS-CP bei CPU-STOP geht: • RUN • CLEAR • STOP (wird auf den Betriebszustand OFFLINE abgebildet) *) • OFFLINE Als DEFAULT-Wert bei CPU-STOP geht der PROFIBUS-CP in den DP-Betriebszustand CLEAR.
6	DP-Betriebszustand für CP-STOP lesen		=1	Mit diesem Auftrag ermitteln Sie, in welchen DP-Betriebszustand der PROFIBUS-CP bei CP-STOP geht : • STOP (wird auf den Betriebszustand OFFLINE abgebildet) *) • OFFLINE Als DEFAULT-Wert bei CP-STOP geht der PROFIBUS-CP in den DP-Betriebszustand OFFLINE.
7	Eingangsdaten lesen	1...126	>=1	Mit diesem Auftrag lesen Sie als DP-Master (Klasse 2) die Eingangsdaten eines DP-Slave. Dieser Vorgang wird auch als "Shared Input" bezeichnet.
8	Ausgangsdaten lesen	1...126	>=1	Mit diesem Auftrag lesen Sie als DP-Master (Klasse 2) die Ausgangsdaten eines DP-Slave. Dieser Vorgang wird auch als "Shared Output" bezeichnet.
10	Aktuellen DP-Slave-Zustand lesen	1...126	>=0	Mit diesem Auftrag lesen Sie den aktuellen Betriebszustand des DP-Slave aus. Folgende Zustände sind möglich: • Der DP-Master tauscht zyklisch Daten mit dem DP-Slave aus. • Der DP-Master liest zyklisch die Eingangsdaten des DP-Slave. • Der DP-Master liest zyklisch die Ausgangsdaten des DP-Slave. • Der DP-Master bearbeitet diesen DP-Slave momentan nicht zyklisch.

*) Der Betriebszustand STOP wird bei den aktuellen Baugruppen (ab Baugruppentyp DA02) nicht mehr unterstützt.

Siehe auch

DP_DIAG (Seite 6944)

Ringpuffer für Diagnosedaten - DP_DIAG

Ringpuffer für Diagnosedaten

Die folgende Darstellung zeigt das Prinzip des Lesens bei der Auftragsart "ältere DP-Einzeldiagnose lesen". Mit dem 1. Lesezugriff wird die **jüngste ältere Diagnose** gelesen.

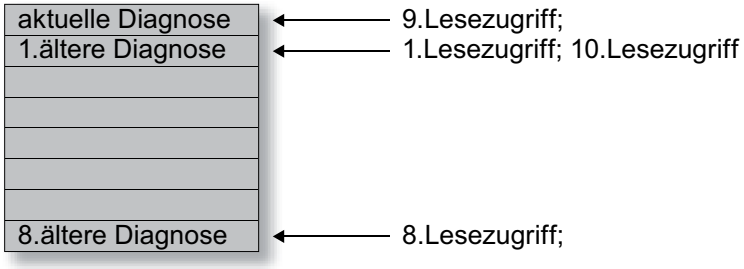

Bild 4-42 Ringpuffer für Diagnosedaten

Mit dem Auslesen einer aktuellen Diagnose wird der Lesezeiger auf 1. ältere Diagnose zurückgesetzt.

Siehe auch

DP_DIAG (Seite 6944)

Parameter NDR, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus NDR, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System- Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-386 Anzeigen DP_DIAG

NDR	ERROR	STATUS	Möglich bei DTYPE	Bedeutung
0	0	8181_H	2-10	Auftrag läuft. Der DP-Master ist nicht gestartet wegen • CP-STOP oder • keine Parametrierung (tritt hier anstelle der Anzeige 0,1,8183_H auf).
0	0	8182_H	0	Anstoß nicht sinnvoll. Der DP-Master ist nicht gestartet wegen • CP-STOP oder • keine Parametrierung (tritt hier anstelle der Anzeige 0,1,8183_H auf).
0	0	8182_H	1	Keine neue Diagnose vorhanden. Der DP-Master ist nicht gestartet wegen • CP-STOP oder • keine Parametrierung (tritt hier anstelle der Anzeige 0,1,8183_H auf).
1	0	0000_H	0, 1 und 4-9	Auftrag abgeschlossen ohne Fehler . Hinweis: Bei DTYPE 2,3 und 10 wird die fehlerfreie Ausführung mit einem Statuscode ungleich "0" signalisiert. Entsprechend finden Sie nachfolgend für eine fehlerfreie Ausführung detaillierte Statuscodes für den Bereich: 82XXH Bei fehlerbehafteter Ausführung erhalten Sie Statuscodes in den folgenden Bereichen: 80XXH, 83XXH, 8FXXH
1	0	8222_H	7,8	Auftrag abgeschlossen ohne Fehler. Die Länge der gelesenen DP-Slave-Daten ist ungleich der Datenlänge, die der DP-Master anhand der Modulliste des DP-Slave in der CP-Datenbasis erwartet.
1	0	8227_H	7,8	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : es sind keine Daten vorhanden.
1	0	8231_H	4,5,6	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Betriebszustand ist bereits "RUN"
1	0	8232_H	4,5,6	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Betriebszustand ist bereits "CLEAR"
1	0	8233_H	4,5,6	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Betriebszustand ist bereits in STOP Der Zustand STOP wird auf den Zustand OFFLINE abgebildet (hier Anzeige 8234_H). Bitte beachten Sie auch die Angaben im Gerätehandbuch.
1	0	8234_H	4,5,6	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Betriebszustand ist bereits "OFFLINE"

NDR	ERROR	STATUS	Möglich bei DTYPE	Bedeutung
1	0	$823A_H$	2,3,7,8	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : es wurden 241 oder 242 Byte Daten gelesen. Bereitgestellt werden 240 Byte.
1	0	8241_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der angegebene DP-Slave wurde nicht projektiert.
1	0	8243_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : in der Modulliste des CP-Slave in der CP-Datenbasis sind nur Leer -Module enthalten.
1	0	8245_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Slave ist im Zustand "Eingangsdaten zyklisch lesen".
1	0	8246_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Slave ist im Zustand "Ausgangsdaten zyklisch lesen".
1	0	8248_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Anmerkung: Dies ist für die genannten Diagnosetypen die Standardanzeige, sofern keine zu signalisierenden Besonderheiten vorliegen.
1	0	8249_H	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Slave ist deaktiviert wegen eines Wechsel des DP-Betriebszustandes (z. B. CP-Schalter auf STOP).
1	0	$824A_H$	2,3,10	Auftrag abgeschlossen ohne Fehler. Hinweis-Meldung : der DP-Slave ist deaktiviert aufgrund eines DP_CTRL-Auftrages im CPU-Programm.
0	1	8090_H	0-10	Logische Basisadresse der Baugruppe ist ungültig
0	1	$80B0_H$	0-10	Die Baugruppe kennt den Datensatz nicht oder befindet sich im RUN --> STOP Übergang.
0	1	$80B1_H$	0-10	Angegebene Datensatzlänge ist falsch
0	1	$80C0_H$	0-10	Datensatz kann nicht gelesen werden
0	1	$80C1_H$	0-10	Der angegebene Datensatz ist gerade in Bearbeitung
0	1	$80C2_H$	0-10	Es liegt ein Auftragsstau vor
0	1	$80C3_H$	0-8	Betriebsmittel (Speicher) belegt
0	1	$80C4_H$	0-10	Kommunikationsfehler
0	1	$80D2_H$	0-10	Logische Basisadresse falsch
0	1	8183_H	0-10	DP Master nicht projektiert.
0	1	8184_H	0-8	Systemfehler bzw. unzulässiger Parametertyp.
0	1	8311_H	>=2	Parameter DTYPE außerhalb des Wertebereichs .
0	1	8313_H	2,3,7,8,10	Parameter STATION außerhalb des Wertebereichs.
0	1	8321_H	>=2	Der DP-Slave stellt keine gültigen Daten zur Verfügung.
0	1	8326_H	7,8	Der DP-Slave stellt mehr als 242 Byte Daten zur Verfügung. Der PROFIBUS-CP unterstützt maximal 242 Byte.
0	1	8335_H	7,8	Der PROFIBUS-CP ist im PROFIBUS-Status: "Station nicht im Ring".
0	1	8341_H	2,3,7,8,10	Der angegebene Slave wurde nicht projektiert
0	1	8342_H	7,8	Der DP-Slave mit der im Parameter STATION angegebenen PROFIBUS-Adresse ist nicht erreichbar.
0	1	8349_H	7,8	Der DP-Master ist im Zustand OFFLINE.
0	1	$8F22_H$	0-10	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz)

NDR	ERROR	STATUS	Möglich bei DTYPE	Bedeutung
0	1	$8F23_H$	0-10	Bereichslängenfehler beim Schreiben eines Parameters (z. B. DB zu kurz)
0	1	$8F24_H$	0-10	Bereichsfehler beim Lesen eines Parameters
0	1	$8F25_H$	0-10	Bereichsfehler beim Schreiben eines Parameters
0	1	$8F28_H$	0-10	Ausrichtungsfehler beim Lesen eines Parameters
0	1	$8F29_H$	0-10	Ausrichtungsfehler beim Schreiben eines Parameters
0	1	$8F30_H$	0-10	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein
0	1	$8F31_H$	0-10	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein
0	1	$8F32_H$	0-10	Parameter enthält zu große DB-Nummer
0	1	$8F33_H$	0-10	DB-Nummernfehler
0	1	$8F3A_H$	0-10	Bereich nicht geladen (DB)
0	1	$8F42_H$	0-10	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich
0	1	$8F43_H$	0-10	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich
0	1	$8F44_H$	0-10	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt
0	1	$8F45_H$	0-10	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt
0	1	$8F7F_H$	0-10	Interner Fehler. z. B. Unzulässige ANY-Referenz

Siehe auch

DP_DIAG (Seite 6944)

DP_CTRL

DP_CTRL

Beschreibung

Die Anweisung DP_CTRL übergibt Steueraufträge an den PROFIBUS-CP. Durch Angabe eines Auftragsblockes (Parameter CONTROL) wird der Steuerauftrag näher spezifiziert.

Es werden folgende Auftragsarten unterschieden:

- Global Control azyklisch / zyklisch;
- Ältere Diagnose löschen;
- Aktuellen DP-Betriebszustand setzen;
- DP-Betriebszustand für AG-/CP-STOP setzen;
- Ein-/Ausgangsdaten zyklisch lesen;
- Bearbeitungsmodus des DP-Slave setzen.

Es gibt Einschränkungen bezüglich der hier genannten Auftragsarten (bitte beachten Sie hierzu die Angaben im Gerätehandbuch).

Hinweis

Die Anweisung DP_CTRL ist nur in der DP-Betriebsart mit DP-Master sinnvoll.

Anschluss

Solange die Anweisung läuft, darf sie nicht mit neuen Auftragsdaten versorgt werden.

Aufrufschnittstelle

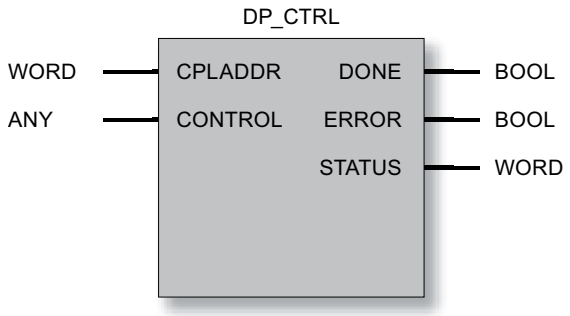

Siehe auch

Parameter bei DP_CTRL (Seite 6954)

Auftragsarten DP_CTRL (Seite 6956)

Command Mode und Group Select - DP_CTRL (Seite 6959)

Parameter DONE, ERROR, STATUS (Seite 6960)

Parameter bei DP_CTRL

Erläuterung der Formalparameter

Die folgende Tabelle erläutert alle Formalparameter für die Anweisung DP_CTRL:

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
CPLADDR	INPUT	WORD		Baugruppen-Anfangsadresse Bei der Konfiguration des CP wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
CONTROL	INPUT	ANY (als VARTYPE sind nur zugelassen: BYTE, WORD und DWORD)	Die Länge ist einzustellen von 1...240	Angabe von Adresse und Länge des CONTROL-Auftragsblockes Adresse des Datenbereichs. Verweist alternativ auf: • PA-Bereich • Merkerbereich • Datenbausteinbereich Die Länge muss mindestens so groß wie die Parameteranzahl gewählt werden.
DONE	OUTPUT	BOOL	0: - 1: Auftrag fehlerfrei ausgeführt.	Zeigt an, ob der Auftrag gesendet und fehlerfrei abgeschlossen wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6960)
ERROR	OUTPUT	BOOL	0: - 1: Fehlerfall	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe Parameter DONE, ERROR, STATUS (Seite 6960)
STATUS	OUTPUT	WORD	siehe nachfolgende Tabelle 'Anzeigen'	Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe Parameter DONE, ERROR, STATUS (Seite 6960)

Aufbau des CONTROL-Auftragsblockes

Der Kontrollauftrag hat folgende Struktur:

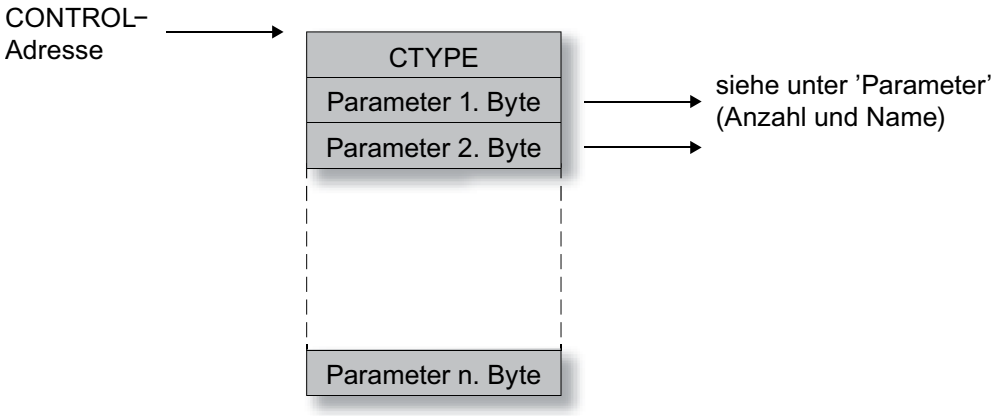

Beispiel für den Auftragsblock

Mit einem Auftragsblock nach unten stehendem Muster wird ein zyklischer Global Control-Auftrag SYNC und UNFREEZE für die Gruppe 4 und 5 ohne die Option Autoclear abgesetzt.

DB 14

Byte 0	01H	CTYPE
Byte 1	24H	Command Mode
Byte 2	18H	Group Select
Byte 3	00H	Autoclear

Die im ANY-Pointer angegebene Länge muss mindestens 4 sein (im Aufrufbeispiel gewählt sind 30).

Siehe auch

DP_CTRL (Seite 6952)

Auftragsarten DP_CTRL

Auftragsarten

Zulässige bzw. sinnvolle Auftragsangaben ergeben sich gemäß nachfolgender Übersicht aus der Angabe für CTYPE sowie den Angaben im Auftragsblock.

CTYPE	Entspricht Auftrag	Parameter im Auftragsblock		Beschreibung
Name	Anzahl			
0	Global Control anstoßen	1. Byte: Command Mode 2. Byte: Group Select (siehe im Anschluss an diese Tabelle)	2	Es wird ein einmaliger Global Control Auftrag an die mit Group Select selektierten DP-Slaves gesendet. Im Parameter Command Mode werden folgende Global Control Aufträge bestimmt : • SYNC • UNSYNC • FREEZE • UNFREEZE • CLEAR - wird nicht unterstützt (bitte beachten Sie auch die Angaben im Gerätehandbuch) Es ist möglich, mehr als einen Auftrag im Parameter Command Mode anzugeben.

CTYPE	Entspricht Auftrag	Parameter im Auftragsblock		Beschreibung
1*)	Zyklischen Global Control anstoßen	1. Byte: Command Mode 2. Byte: Group Select 3. Byte: Autoclear (siehe im Anschluss an diese Tabelle)	3	Es erfolgt der Anstoß an den PROFIBUS-CP, dass dieser zyklisch Global Control Aufträge an die mit Group Select selektierten DP-Slaves sendet. Der Parameter Autoclear wird nur beim Global Control Auftrag SYNC ausgewertet. Wenn mindestens ein DP-Slave in der gewählten Gruppe nicht in der Datentransferphase ist, wird bei Autoclear = 1 zusätzlich der CLEAR-Modus eingeschaltet. Das heißt, die Ausgangsdaten der DP-Slaves werden auf "0" gesetzt. Im Parameter Command Mode können folgende Global Control Aufträge eingeschaltet werden : • SYNC • FREEZE • CLEAR (CLEAR-Bit = 1) - wird nicht unterstützt (bitte beachten Sie auch die Angaben im Gerätehandbuch) bzw. ausgeschaltet werden : • UNSYNC • UNFREEZE • UNCLEAR (CLEAR-Bit = 0) Es ist möglich, mehr als einen Auftrag im Parameter Command Mode anzugeben. Um einen laufenden zyklischen Global Control Auftrag zu beenden, muss ein erneuter Global Control Auftrag (zyklisch oder azyklisch) durchgeführt werden. Um den im Command Mode eingestellten Auftrag zu beenden, muss der entsprechende Auftrag ausgeschaltet werden. Z. B. wird der SYNC-Auftrag durch einen UNSYNC-Auftrag ausgeschaltet.
3	Ältere DP-Einzeldiagnose löschen	1. Byte: Slv_Adresse 1..126 127 = alle Slaves	1	Es werden die älteren, im PROFIBUS-CP gespeicherten DP-Enzeldiagnosedaten für einen oder alle DP-Slaves gelöscht.

CTYPE	Entspricht Auftrag	Parameter im Auftragsblock		Beschreibung
4	Aktuellen DP-Betriebszustand setzen	1. Byte: RUN = 00_H CLEAR = 01_H OFFLINE = 03_H RUN mit AUTOCLEAR = 04_H RUN ohne AUTOCLEAR = 04_H	1	Mit diesem Auftrag kann der DP-Betriebszustand wie folgt gesetzt werden: • RUN • CLEAR • OFFLINE Der Parameter AUTOCLEAR bedeutet, dass der DP-Master Klasse 1 selbständig in den Betriebszustand CLEAR geht, wenn folgende Bedingung erfüllt ist: mindestens einer der DP-Slave, mit denen der DP-Master Klasse 1 Datenaustausch betreiben will, ist nicht im Datentransfer. Mit dem Parameter RUN ohne AUTOCLEAR wird AUTOCLEAR zurückgesetzt. Hinweise: Der Betriebszustand STOP = 02_H wird bei den aktuellen Baugruppen nicht mehr unterstützt (ab Baugruppentyp DA02). Eine Angabe STOP = 02_H wird auf den Betriebszustand OFFLINE abgebildet.
5	DP-Betriebszustand für CPU-STOP setzen	1. Byte: RUN = 00_H CLEAR = 01_H OFFLINE = 03_H	1	Mit diesem Auftrag wird festgelegt, in welchen DP-Betriebszustand der PROFIBUS-CP bei CPU-STOP geht : • RUN • CLEAR • OFFLINE Als DEFAULT-Wert bei CPU- STOP geht der PROFIBUS-CP in den DP-Betriebszustand CLEAR. Der gesetzte Betriebszustand bleibt bei einem CP-Zustandswechsel von RUN --> STOP --> RUN erhalten. Hinweise: Der Betriebszustand STOP = 02_H wird bei den aktuellen Baugruppen (ab Baugruppentyp DA02) nicht mehr unterstützt. Eine Angabe STOP = 02_H wird auf den Betriebszustand OFFLINE abgebildet.
6	DP-Betriebszustand für CP-STOP setzen	1. Byte: OFFLINE=03_H	1	Mit diesem Auftrag wird festgelegt, in welchen DP-Betriebszustand der PROFIBUS-CP bei CP-STOP geht : • OFFLINE Als DEFAULT-Wert bei CP-STOP geht der PROFIBUS-CP in den DP-Betriebszustand OFFLINE. Der gesetzte Betriebszustand bleibt bei einem CP-Zustandswechsel von RUN --> STOP --> RUN erhalten. Hinweise: Der Betriebszustand STOP = 02_H wird bei den aktuellen Baugruppen (ab Baugruppentyp DA02) nicht mehr unterstützt. Eine Angabe STOP = 02_H wird auf den Betriebszustand OFFLINE abgebildet.
7 *)	Zyklisches Lesen der Eingangsdaten (DP-Master Klasse 2)	1. Byte: Slave-Adresse 1..125	1	Dieser Auftrag wird nicht unterstützt. Beachten Sie auch die Angaben im Gerätehandbuch.

CTYPE	Entspricht Auftrag	Parameter im Auftragsblock		Beschreibung
8 *)	Zyklisches Lesen der Ausgangsdaten (DP-Master Klasse 2)	1. Byte: Slave-Adresse 1..125	1	Dieser Auftrag wird nicht unterstützt. Beachten Sie auch die Angaben im Gerätehandbuch.
9	Zyklisches Bearbeiten des DP-Slave vom DP-Master (Klasse 1 /Klasse 2) beenden	1. Byte: Slave-Adresse 1..125	1	Mit diesem Auftrag wird das zyklische Lesen der Eingangsdaten oder Ausgangsdaten des adressierten DP-Slave oder der Datentransfer (DP-Master Klasse 1) beendet. Anschließend wird der DP-Slave nicht mehr vom PROFIBUS-CP als DP-Master (Klasse 2) bearbeitet **Dieser Vorgang deaktiviert den DP-Slave.**
10	Zyklisches Bearbeiten als DP-Master (Klasse 1) aufnehmen	1. Byte: Slave-Adresse 1..125	1	Der PROFIBUS-CP als DP-Master (Klasse 1) parametriert den adressierten DP-Slave und nimmt den zyklischen Datentransfer auf (Ausgänge schreiben / Eingänge lesen). **Dieser Vorgang aktiviert den DP-Slave.**

*) Dieser CTYPE wird bei den aktuellen Baugruppen (ab Baugruppentyp DA02) nicht mehr unterstützt.

Siehe auch

DP_CTRL (Seite 6952)

Command Mode und Group Select - DP_CTRL

Aufbau von Command Mode

Für die Aufträge Global Control zyklisch und azyklisch geben Sie im Parameter Command Mode die Betriebsmodi für die Ein- und Ausgangsdaten an.

Hierbei bedeutet jeweils:

1 = aktiviert
0 = nicht aktiviert

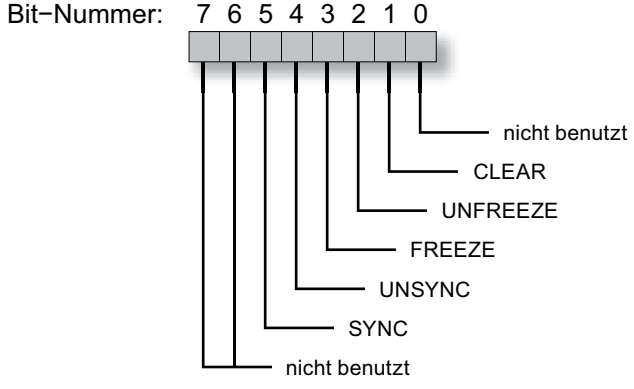

Aufbau von Group Select

Mit dem Parameter Group Select legen Sie fest, auf welche Gruppe der im Command Mode angegebene Steuerauftrag angewendet werden soll. Group Select belegt das 2. Byte im Kontrollauftrag. Jedes Bit definiert eine mögliche DP-Slavegruppe.

Hierbei bedeutet jeweils:

1 = zugeordnet
0 = nicht zugeordnet

Bit−Nummer:	7	6	5	4	3	2	1	0
Gruppe:	8	7	6	5	4	3	2	1

Siehe auch

DP_CTRL (Seite 6952)

Parameter DONE, ERROR, STATUS

Anzeigen

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus den Parametern DONE, ERROR und STATUS.

Hinweis

Beachten Sie für die Einträge mit der Codierung $8Fxx_H$ unter STATUS auch die Angaben zum Ausgangsparameter RET_VAL in den Beschreibungen der referenzierten System-Programmbausteine.

Welche System-Programmbausteine genutzt werden und für die Fehlerauswertung relevant sind, können Sie über STEP 7 abfragen.

Tabelle 4-387 Anzeigen DP_CTRL

DONE	ERROR	STATUS	Möglich bei CTYPE	Bedeutung
0	0	8181_H	0..10	Auftrag läuft. Der DP-Master ist nicht gestartet wegen: • CP-STOP oder • "keine Parametrierung" Anmerkung: Die beschriebene Anzeige tritt anstelle einer der folgenden, weiter unten beschriebenen Anzeigen auf: 0,1,8183_H 0,1, 8333_H 0,1, 8334_H
1	0	0000_H	0..10	**Auftrag fertig ohne Fehler.**
1	0	8214_H	0,1	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : zyklischer Global Control Auftrag wird als azyklischer Global Control -Auftrag gesendet
1	0	8215_H	0,1	**Auftrag abgeschlossen ohne Fehler.** Die in der selektierten Gruppe angesprochenen DP-Slaves sind alle deaktiviert.
1	0	8219_H	0,1	**Auftrag abgeschlossen ohne Fehler.** Es wurde versucht, einen bereits laufenden zyklischen Global Control noch einmal abzusetzen. Der Global Control läuft unverändert weiter.
1	0	8228_H	0,1	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : die in selektierten Gruppen angesprochenen DP-Slaves besitzen keine Eingangsmodule.
1	0	8229_H	0,1	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : die in selektierten Gruppen angesprochenen DP-Slaves besitzen keine Ausgangsmodule.
1	0	8231_H	4,5,6	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "RUN"
1	0	8232_H	4,5,6	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "CLEAR"
1	0	8233_H	4,5,6	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "STOP"
1	0	8234_H	4,5,6	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "OFFLINE"
1	0	8235_H	4	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "RUN" mit eingeschaltetem AUTOCLEAR
1	0	8236_H	4	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Betriebszustand ist bereits "RUN" mit ausgeschaltetem AUTOCLEAR
1	0	8241_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der angegebene DP-Slave wurde nicht projektiert.
1	0	8243_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Slave ist bereits deaktiviert, da in der Modulliste des DP-Slave in der CP-Datenbasis nur Leer-Module enthalten sind .

DONE	ERROR	STATUS	Möglich bei CTYPE	Bedeutung
1	0	8245_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Slave ist bereits im Zustand "Eingangsdaten zyklisch lesen"
1	0	8246_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Slave ist bereits im Zustand "Ausgangsdaten zyklisch lesen"
1	0	8248_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung: in der Modulliste des DP-Slaves in der CP-Datenbasis sind Eingangs-, Ausgangs-, oder Ein/Ausgangsmodule enthalten.
1	0	8249_H	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung: der Slave ist wegen eines Wechsels des DP-Betriebszustandes deaktiviert.
1	0	$824A_H$	7-10	**Auftrag abgeschlossen ohne Fehler.** Hinweis-Meldung : der DP-Slave ist bereits deaktiviert auf Grund eines DP_CTRL Auftrags im CPU-Programm.
0	1	8090_H	0..10	Baugruppe mit dieser Adresse nicht vorhanden.
0	1	8091_H	0..10	Logische Adresse nicht auf Doppel-Wort-Raster.
0	1	$80B0_H$	0..10	Baugruppe kennt den Datensatz nicht.
0	1	$80B1_H$	0..10	Angegebene Datensatzlänge ist falsch.
0	1	$80C0_H$	0..10	Datensatz kann nicht gelesen werden.
0	1	$80C1_H$	0..10	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	$80C2_H$	0..10	Es liegt ein Auftragsstau vor.
0	1	$80C3_H$		Betriebsmittel (Speicher) belegt.
0	1	8183_H	0..10	DP-Master ist nicht projektiert... Anmerkung: Wenn sich der DP-Master im Betriebszustand "STOP" befindet, dann kann auch der Status 8181_H ausgegeben werden.
0	1	8184_H		Systemfehler bzw. unzulässiger Parametertyp...
0	1	8311_H	0..10	Parameter CTYPE außerhalb des Wertebereichs
0	1	8312_H	0..10	Länge des Bereiches im Parameter CONTROL ist zu klein.
0	1	8313_H	3,7,8,9, 10	Parameter Slave-Adresse außerhalb des Wertebereichs.
0	1	8315_H	0,1	Alle DP-Slaves der im global Control angegebenen Gruppe sind deaktiviert (tritt bei leerer Gruppe immer auf).
0	1	8317_H	8	Die Länge der projektierten Ausgangsdaten ist größer als der projektierte Empfangsbereich des DP-Slave. Eine Aktivierung des Slave-Modus "Lese Ausgangsdaten" ist nicht möglich.
0	1	8318_H	0,1,4,5,6	Parameter 1. Byte des Auftragsdatenblocks liegt außerhalb des Wertebereichs. Bei GLOBAL CONTROL wurde CLEAR mit SYNC verwendet oder ein GLOBAL CONTROL mit gesetztem CLEAR an die Gruppe 0 gesendet.
0	1	$831A_H$	0,1	Mindestens ein DP-Slave beherrscht FREEZE nicht.
0	1	$831B_H$	0,1	Mindestens ein DP-Slave beherrscht SYNC nicht.
0	1	8333_H	0,1	Der Auftrag ist im DP-Betriebszustand "STOP" nicht erlaubt. Anmerkung: Wenn kein DP-Master projektiert ist, dann kann auch der Status 8181_H ausgegeben werden.

DONE	ERROR	STATUS	Möglich bei CTYPE	Bedeutung
0	1	8334_H	0, 1	Der Auftrag ist im DP-Betriebszustand "OFFLINE" nicht erlaubt. Anmerkung: Wenn kein DP-Master projektiert ist, dann kann auch der Status 8181_H ausgegeben werden.
0	1	8335_H	0, 1	Der PROFIBUS-CP ist im PROFIBUS-Status: "Station nicht im Ring".
0	1	8339_H	0, 1	Mindestens ein DP-Slave der selektierten Gruppe befindet sich nicht in der Datentransferphase.
0	1	$833C_H$	1	Zyklischer Global Control darf im Modus "Plc <-> CP freilaufend" nicht benutzt werden. Dieser Fehler tritt beim CP 3425 nicht auf, da dieser Modus dort nicht möglich ist (Datenübertragung erfolgt immer über PBUS Datensätze).
0	1	8341_H	7-10	Der angegebene DP-Slave wurde nicht projektiert.
0	1	8183_H	0..10	DP Master nicht projektiert.
0	1	8184_H	-	Systemfehler bzw. unzulässiger Parametertyp.
0	1	$8F22_H$	0..10	Bereichslängenfehler beim Lesen eines Parameters (z. B. DB zu kurz).
0	1	$8F23_H$	0..10	Bereichslängenfehler beim Schreiben eines Parameters.
0	1	$8F24_H$	0..10	Bereichsfehler beim Lesen eines Parameters.
0	1	$8F25_H$	0..10	Bereichsfehler beim Schreiben eines Parameters.
0	1	$8F28_H$	0..10	Ausrichtungsfehler beim Lesen eines Parameters.
0	1	$8F29_H$	0..10	Ausrichtungsfehler beim Schreiben eines Parameters.
0	1	$8F30_H$	0..10	Der Parameter liegt im schreibgeschützten 1. aktuellen Datenbaustein.
0	1	$8F31_H$	0..10	Der Parameter liegt im schreibgeschützten 2. aktuellen Datenbaustein.
0	1	$8F32_H$	0..10	Parameter enthält zu große DB-Nummer.
0	1	$8F33_H$	0..10	DB-Nummernfehler.
0	1	$8F3A_H$	0..10	Bereich nicht geladen (DB).
0	1	$8F42_H$	0..10	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
0	1	$8F43_H$	0..10	Quittungsverzug beim Schreiben des Parameters in den Peripheriebereich.
0	1	$8F44_H$	0..10	Der Zugriff auf einen in der Bearbeitung der Anweisung zu lesenden Parameter ist gesperrt.
0	1	$8F45_H$	0..10	Der Zugriff auf einen in der Bearbeitung der Anweisung zu schreibenden Parameter ist gesperrt.
0	1	$8F7F_H$	0..10	Interner Fehler. z. B. Unzulässige ANY-Referenz.
0	1	$80C4_H$	0..10	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll).
0	1	$80D2_H$	0..10	Logische Basisadresse falsch.

Siehe auch

DP_CTRL (Seite 6952)

SIMATIC NET-CM/CP

S7-1500-CM/CP

Industrial Ethernet

Anweisungen für FTP-Dienste

FTP_CMD für FTP-Dienste

Übersicht FTP_CMD

Bedeutung

Mit der Anweisung FTP_CMD können Sie FTP-Verbindungen aufbauen und Dateien von und zu einem FTP-Server übertragen.

Hinweis

Bausteinversionen

Die Version V2.x des FTP_CMD können Sie in einer Station nur zusammen mit einer CPU V2.x und einem CP V2.x verwenden.

Sobald die Station eine CPU V1.x oder einen CP V1.x enthält, müssen Sie den FTP_CMD in der älteren Version V1.x verwenden (bspw. V1.4). Schalten Sie hierzu die Version der Bibliothek "SIMATIC NET CP" auf V3.4. Sie können dann eine ältere Version des Bausteins auswählen.

Der Datentransfer ist über FTP oder FTPS (gesichterte SSL-Verbindungen) möglich.

Hinweis

FTPS: Zertifikate abgleichen

FTPS erfordert den Abgleich der Zertifikate zwischen FTP-Server und FTP-Client. Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich. Importieren Sie im Zertifikatsmanager das Zertifikat des FTP-Servers als vertrauenswürdiges Zertifikat.

Funktionsweise

Die Anweisung FTP_CMD verweist auf einen Auftragsblock (ARG), in dem das FTP-Kommando spezifiziert wird. Je nach Typ des FTP-Kommandos (CMD) verwendet dieser Auftragsblock unterschiedliche Datenstrukturen zur Parametrierung. Für diese unterschiedlichen Strukturen stehen jeweils passende Datentypen (UDTs) zur Verfügung.

Die folgende Darstellung zeigt die Aufrufstruktur:

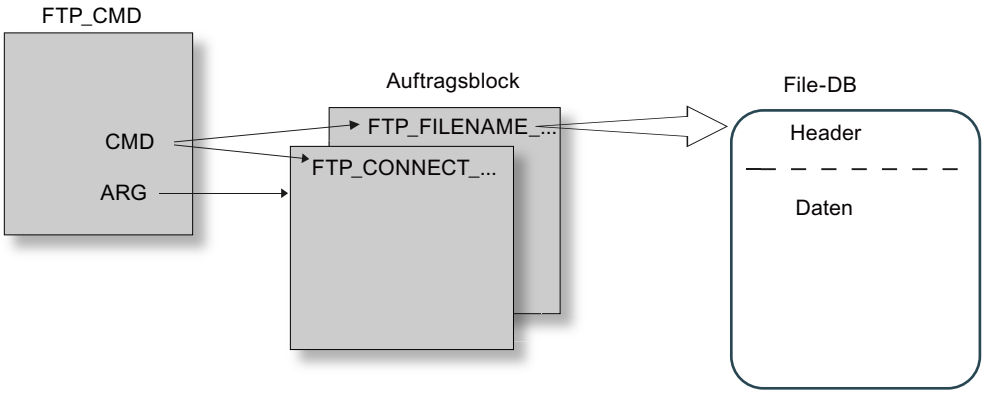

Auftragsblöcke

Für die Auftragsblöcke werden folgende Datenstrukturen verwendet:

- Verbindungsaufbau
  Für den Verbindungsaufbau stehen unterschiedliche Datenstrukturen für folgende Zugriffsarten zur Verfügung:
  - FTP_CONNECT_IPV4: Verbindungsaufbau mit IP-Adressen gemäß IPv4
  - FTP_CONNECT_IPV6: Verbindungsaufbau mit IP-Adressen gemäß IPv6
  - FTP_CONNECT_NAME: Verbindungsaufbau mit Server-Namen (DNS)
- Datentransfer
  Für den Datentransfer stehen zwei unterschiedliche Datenstrukturen zur Verfügung:
  - FTP_FILENAME: Datenstruktur für den Zugriff auf eine vollständige Datei
  - FTP_FILENAME_PART: Datenstruktur für den lesenden Zugriff auf einen Datenbereich

Datenübertragung im File_DB

Der Datentransfer erfolgt über Datenbausteine, die einen Header für Auftragsdaten sowie den Bereich für die Nutzdaten enthalten. Der Datenbaustein wird im Auftragspuffer angegeben.

Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen, um den FTP-Zugriff zu ermöglichen:

- Deaktivieren Sie bei allen als File-DB verwendeten Datenbausteinen das Attribut "Optimierter Bausteinzugriff".
- Nur bei Verwendunng einer CPU V1.x und eines CP V1.1.x:
  Deaktivieren Sie in den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz" die Option "PUT/GET-Kommunikation deaktivieren" (PUT/GET muss freigegeben sein).

Gültigkeit

Die Anweisung FTP_CMD kann mit folgenden Baugruppentypen eingesetzt werden:

- CP 1543-1

Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung

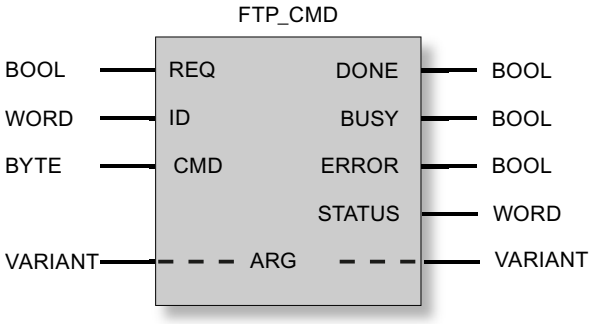

Siehe auch

Ausgangsparameter und Statusinformationen FTP_CMD (Seite 6974)

Eingangsparameter FTP_CMD (Seite 6966)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb (Seite 6977)

Eingangsparameter FTP_CMD

Erläuterung der Eingangsparameter

Sie versorgen die Anweisung FTP_CMD mit folgenden Eingangsparametern:

Tabelle 4-388 Formalparameter der Anweisung FTP_CMD - Eingangsparameter

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung / Bemerkung
REQ	Input	BOOL	E, A, M, D, L	Startet den Sendeauftrag bei einer steigenden Flanke.
ID *	INPUT	INT	1, 2 ... 64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.

Parameter	Deklaration	Datentyp	Speicherbereich	Bedeutung / Bemerkung
CMD *	INPUT	BYTE	Siehe nachfolgende Tabelle "Kommandos".	FTP-Kommando, das beim Aufruf der Anweisung ausgeführt werden soll. Wertebereiche für die FTP-Kommandotypen finden Sie im Anschluss an die Tabelle. Hinweis: Das hier angegebene FTP-Kommando muss identisch im Auftragsblock (Parameter ARG) angegeben werden. Wenn ein Kommando nicht von der CP-Firmware unterstützt wird, dann wird eine Fehlermeldung mit STATUS = $8F6B_H$ ausgegeben.
ARG *	INPUT	VARIANT	Siehe nachfolgende Tabelle "Kommandos".	Auftragsblock Verweist auf den Datenbereich mit den zum FTP-Kommando passenden Ausführungsparametern. Abhängig vom FTP-Kommando werden spezifische Datentypen (UDT) verwendet. Die UDTs werden nachfolgend angegeben. Für den hier anzugebenden Zeiger ist der Datentyp ANY nicht zulässig!

* Die Werte der Eingangsparameter "ID" und "CMD" überschreiben den Wert des Eingangsparameters "ARG".

FTP-Kommandos im Parameter "CMD"

Entnehmen Sie der folgenden Tabelle, welche Bedeutung die Kommandos des Parameters "CMD" haben und welche UDTs Sie zur Versorung der Auftragsblöcke verwenden.

Tabelle 4-389 Kommandotypen

CMD (Kommandotyp)	Relevante Auftragsblöcke / UDT	Bedeutung / Handhabung
0 (NOOP)	*	Der aufgerufene FC führt keine Aktionen aus. Die Statusanzeigen werden bei dieser Parameterversorgung wie folgt gesetzt: DONE=1; ERROR=0; STATUS=0
1 (CONNECT)	FTP_CONNECT_IPV4 FTP_CONNECT_IPV6 FTP_CONNECT_NAME	FTP-Verbindungsaufbau Mit diesem Kommando baut der FTP-Client eine FTP-Verbindung zu einem FTP-Server auf (Port 21). Die Verbindung steht unter der hier zugewiesenen Verbindungs-ID für alle weiteren FTP-Kommandos zur Verfügung. Daten werden dann mit dem für diesen Benutzer angegebenen FTP-Server ausgetauscht.
2 (STORE)	FTP_FILENAME	Mit diesem Funktionsaufruf wird ein Datenbaustein (File-DB) vom FTP-Client (S7-CPU) zum FTP-Server übertragen. Achtung: Falls die Datei (File-DB) auf dem FTP-Server schon vorhanden ist, wird diese überschrieben.
3 (RETRIEVE)	FTP_FILENAME	Mit diesem Funktionsaufruf wird eine Datei vom FTP-Server zum FTP-Client (S7-CPU) übertragen. Achtung: Falls der Datenbaustein (File-DB) beim FTP-Client schon eine Datei enthält, wird diese überschrieben.

CMD (Kommandotyp)	Relevante Auftragsblöcke / UDT	Bedeutung / Handhabung
4 (DELETE)	FTP_FILENAME	Mit diesem Funktionsaufruf löschen Sie eine Datei auf dem FTP-Server.
5 (QUIT)	*	Mit diesem Funktionsaufruf bauen Sie die über die ID benannte FTP-Verbindung ab.
6 (APPEND)	FTP_FILENAME	Ähnlich wie "STORE" speichert das Kommando "APPEND" (anhängen) eine Datei auf dem FTP-Server. Bei "APPEND" wird die Datei auf dem FTP-Server aber nicht überschrieben, sondern der neue zu speichernde Inhalt wird an die Datei angehängt. Falls die Datei auf dem FTP-Server nicht vorhanden ist, wird sie angelegt.
7 (RETR_PART)	FTP_FILENAME_PART	Mit dem Kommando "RETR_PART" (Teillänge lesen) können Sie einen Ausschnitt einer Datei vom FTP-Server anfordern. Bei sehr großen Dateien können Sie damit das Lesen auf den Teil beschränken, den Sie gerade benötigen. Dazu müssen Sie die Struktur dieser Datei kennen. Geben Sie den gewünschten Ausschnitt der Datei mithilfe der zwei Parameter "OFFSET" und "LEN" am FB 40 an.

* Bei den Kommandotypen 0 (NOOP) und 5 (QUIT) muss ein beliebiger Auftragsblock (UDT) angegeben werden. Dieser wird nicht ausgewertet.

Siehe auch

Übersicht FTP_CMD (Seite 6964)

Auftragsblöcke für FTP_CMD

Bedeutung

Sie versorgen die Anweisung FTP_CMD über den Parameter ARG mit einem Auftragsblock. Die Struktur ist abhängig vom FTP-Kommandotyp. Indem Sie die vorgegebenen Datentypen (UDT) verwenden, erkennt die Anweisung den Typ des Auftragsblocks. Nachfolgend werden die jeweiligen Datentypen (UDT) für folgende Auftragsblöcke angegeben:

- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4
- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6
- FTP-Verbindungsaufbau mit Server-Name
- Schreib- und Lesezugriff sowie übrige FTP-Kommandos
- FTP-Kommando RETR_PART

Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4 wird die folgende Datenstruktur verwendet.

Tabelle 4-390 FTP_CONNECT_IPV4

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPCmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 6966). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverIPaddr	IP_V4	ADDR(1) ... ADDR(4)	IP-Adresse des FTP-Servers als Array[1..4] of Byte, wobei jeweils 1 Byte einen Block der Adresse spezifiziert. Bsp.: ADDR(1) spezifizieren den ersten Adressblock (das erste Byte der Adresse).

Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6 wird die folgende Datenstruktur verwendet.

Tabelle 4-391 FTP_CONNECT_IPV6

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPCmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 6966). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverIPaddr	IP_V6	ADDR(1) ... ADDR(16)	IP-Adresse des FTP-Servers als Array[1..16] of Byte, wobei jeweils 2 Byte einen Block der Adresse spezifizieren. Bsp.: ADDR(1) + ADDR(2) spezifizieren den ersten Adressblock.

Auftragsblock für FTP-Verbindungsaufbau mit Server-Name

Für den FTP-Verbindungsaufbau mit Angabe eines Server-Namens wird die folgende Datenstruktur verwendet. Der Server-Name wird über DNS einer IP-Adresse zugewiesen.

Tabelle 4-392 FTP_CONNECT_NAME

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	1	FTP-Kommando "CONNECT" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 6966). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
UserName	STRING[32]	’benutzer’	Benutzername für das Login auf dem FTP-Server
Password	STRING[32]	’passwort’	Passwort für das Login auf dem FTP-Server
FTPserverName	STRING[254]		IP-Adresse des FTP-Servers

Auftragsblock für Schreib- und Lesezugriff sowie übrige FTP-Kommandos

Für die FTP-Kommandos store, retrieve, delete und append wird die folgende Datenstruktur verwendet.

Tabelle 4-393 FTP_FILENAME

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	2, 3, 4, 6	FTP-Kommando "STORE / RETRIEVE / DELETE / APPEND" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 6966). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
DataBlockNumber	UINT		Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB.
LenFilename	UINT	0...1000	Der Parameter "LenFilename" zur Angabe der Gesamtlänge des Dateinamens wird nicht ausgewertet. Statt dessen wird die Längenangabe im String des Parameters "Filename" ausgewertet.
Filename	ARRAY[0..3] OF STRING[220]		Dateiname der Ziel- bzw. Quelldatei.

Auftragsblock für das FTP-Kommando RETR_PART

Für das FTP-Kommando RETR_PART wird die folgende Datenstruktur verwendet.

Tabelle 4-394 FTP_FILENAME_PART

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
InterfaceID	HW_ANY		Baugruppen-Anfangsadresse Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP. Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter: "Eigenschaften>Adressen>Eingänge"
ID	CONN_OUC	1, 2...64	Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.
ConnectionType	BYTE	0	Verbindungstyp "FTP"
ActiveEstablishment	BOOL	TRUE	
FTPcmd	BYTE	7	FTP-Kommando "RETR_PART" FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird.I. Wertebereiche für die Kommandotypen finden Sie im Kapitel Eingangsparameter FTP_CMD (Seite 6966). Hinweis: Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.
CertIndex	BYTE	0 = FTP 1 = FTPS	WählenSie hier zwischen den Protokolltypen FTP oder FTPS. Hinweis für FTPS: Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.
Offset	DWORD		Offset in Byte, ab dem die Datei gelesen werden soll.
Length	DWORD		Teillänge in Byte, die ab dem in "OFFSET" angegebenen Wert gelesen werden. Besonderheiten: • Bei Angabe von "DW#16#FFFFFFFF" wird der verfügbare Rest der Datei gelesen. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET > Länge der Original-Datei: Länge der Ziel-Datei (ACT_LENGTH im File-DB): 0 Byte in der CPU. Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. • Wenn OFFSET + LEN > Länge der Original-Datei (und LEN ≠ 0xFFFFFFFF): Länge der Ziel-Datei (ACT_LENGTH im File-DB): Verfügbare Bytes ab "OFFSET". Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt.

Parameter	Typ	Wertebereich	Bedeutung / Bemerkung
DataBlockNumber	UINT		Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB.
LenFilename	UINT	0...1000	Der Parameter "LenFilename" zur Angabe der Gesamtlänge des Dateinamens wird nicht ausgewertet. Statt dessen wird die Längenangabe im String des Parameters "Filename" ausgewertet.
Filename	ARRAY[0..3] OF STRING[220]		Dateiname der Ziel- bzw. Quelldatei.

Siehe auch

Übersicht FTP_CMD (Seite 6964)

Ausgangsparameter und Statusinformationen FTP_CMD

Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "FTP_CMD" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

BUSY	DONE	ERROR	Beschreibung
1	-	-	Der Auftrag wird bearbeitet.
0	1	0	Der Auftrag wurde erfolgreich durchgeführt.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben.
0	0	0	Kein neuer Auftrag wurde zugewiesen.

Statusanzeigen auswerten

Hinweis

Beachten Sie für die Einträge mit der Codierung 8FxxH unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET_VAL".

Tabelle 4-395 FTP_CMD: Bedeutung des Parameters STATUS in Zusammenhang mit DONE und ERROR

DONE	ERROR	STATUS	Bedeutung
0	0	0000_H	Es ist kein Auftrag in Bearbeitung.
1	0	0000_H	Der Auftrag ist fertig ohne Fehler.
0	0	7001_H	Der Auftrag wurde erstmalig angestoßen (BUSY=1)
0	0	7002_H	Der Auftrag läuft noch (BUSY=1)
0	0	$80C4_H$	Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)
0	0	8183_H	Die Projektierung entspricht nicht den Auftragsparametern.
0	1	8401_H	unbekannter Fehler mögliche Ursachen: • Auf der Verbindung wurde ein Timeout erkannt. • Der FTP-Server hat die Verbindung abgebrochen. Abhilfe: • QUIT und CONNECT-Kommandos erneut senden um die Verbindung wieder einzurichten.
0	1	8402_H	Verbindung befindet sich in fehlerhaftem Zustand. Das Timeout der Verbindung kann überschritten sein oder der FTP-Server kann die Verbindung abgebaut haben. Abhilfe: Senden Sie erneut die Kommandos QUIT und CONNECT um die Verbindung wieder aufzubauen.
0	1	8403_H	Login ist fehlgeschlagen.
0	1	8404_H	FTP-Server ist nicht erreichbar.
0	1	8405_H	Übertragung ist fehlgeschlagen.
0	1	8406_H	Timeout bei aktuellem Vorgang
0	1	8407_H	Datei im FTP-Server wurde nicht gefunden.
0	1	8408_H	Übertragung nicht möglich.
0	1	8409_H	Datei konnte nicht geholt werden.
0	1	8410_H	Das Setzen des TCP Port für die Datenverbindung ist fehlgeschlagen.
0	1	8411_H	Offset-Angabe passt nicht.
0	1	8412_H	Fehler beim Wechsel der Verzeichnisangabe
0	1	8413_H	Fehler beim Daten Empfangen
0	1	8414_H	Fehler beim Daten Senden.
0	1	8415_H	CMD-Angabe (Kommandotyp) wurde vom Client abgewiesen.
0	1	8416_H	Verbindung wurde vom FTP-Server geschlossen.
0	1	8418_H	Fehler in den Nutzdaten. Mögliche Ursachen: • Dateiname ist leer. • Datenlänge ist "0" • usw.
0	1	8419_H	Es ist keine Socket Ressource zum Öffnen einer Datenverbindung vorhanden.
0	1	8420_H	Es ist keine Socket Ressource zum Öffenen einer Controlverbindung vorhanden.

DONE	ERROR	STATUS	Bedeutung
0	1	8421_H	Fehler beim Öffnen des zu lesenden File-DB.
0	1	8422_H	Fehler beim Öffnen des zu beschreibenden File-DB.
0	1	8423_H	Der Verbindungsaufbau zum FTP Server ist fehlgeschlagen.
0	1	8424_H	interner Fehler
0	1	8425_H	Formatfehler beim Domain-Namen
0	1	8426_H	Es stehen zu viele DNS-Anfragen an.
0	1	8427_H	Der angegebene DNS-Server konnte den spezifizierten Domain-Namen niocht zuordnen.
0	1	8428_H	Es ist keine Verbindungs-Ressource verfügbar.
0	1	8429_H	Unbekannte Kanal-ID
0	1	8430_H	Der File-DB ist zu kurz.
0	1	8431_H	Fehler beim Schreiben in den File-DB.
0	1	8432_H	Fehler beim Lesen aus dem File-DB.
0	1	8433_H	Fehler beim Zugriff auf den File-DB.
0	1	8434_H	Aktion wurde abgebrochen.
0	1	8435_H	Kanal wird zurückgesetzt.
0	1	8436_H	Unerwartete Server Antwort
0	1	8437_H	Zertifikat konnte nicht verifiziert werden.
0	1	8438_H	Unbekannter Fehler aufgetreten
0	1	8439_H	Das FTP-Kommando führt zu einem Fehler. Die Ursache ist beim FTP-Server zu suchen (REST-Kommando).
0	1	8440_H	Der FTP-Server unterstützt das geforderte SSL-Protokoll nicht.
0	1	8446_H	Nachdem das FTP-Passwort an den FTP-Server gesendet wurde,wurde ein unerwarteter Code vom FTP-Server zurückgegeben.
0	1	8451_H	Beim Versuch, den Übertragungsmodus von binär auf ASCII umzustellen, wurde ein Fehler gemeldet.
0	1	8455_H	Eine Speicheranforderung im CM/CP ist fehlgeschlagen.
0	1	8460_H	Bei der SSL/TLS-Abwicklung ist ein Problem aufgetreten.
0	1	8469_H	Schnittstellen-Fehler Die angegebene Ausgangs-Schnittstelle konnte nicht genutzt werden. Abhilfe: Stellen Sie ein, welche Schnittstelle für ausgehende Verbindungen genutzt werden soll.
0	1	8475_H	Das SSL-Zertifikat oder der SSH md5 fingerprint wurde als nicht vertrauenswürdig erachtet.
0	1	8476_H	Vom FTP-Server wurde nichts empfangen. Im aktuellen Zustand muss von einem fehlerhaften Verhalten ausgegangen werden.
0	1	8477_H	Die angegebene "Crypto engine" (kryptografisches Modul) wurde nicht gefunden.
0	1	8478_H	Der Vorgang schlug fehl, die gewählte SSL-"Crypto engine" (kryptografisches Modul) als Standard zu setzen.
0	1	8480_H	Es ist ein Problem mit dem Zertifikat des FTP-Client aufgetreten.
0	1	8481_H	Die angegebene Ziffer konnte nicht verwendet werden.
0	1	8482_H	Der FTP-Server verwendet eine Codierung die nicht unterstützt wird.
0	1	8484_H	Die maximale Dateigröße wurde überschritten.

DONE	ERROR	STATUS	Bedeutung
0	1	8485_H	Der File-DB wurde während der Sendebearbeitung verändert oder der File-DB ist nicht korrekt aufgebaut.
0	1	8489_H	Daten konnten nicht gesendet werden. Es ist nicht genügend Speicherkapazität für den Vorgang auf dem FTP-Server verfügbar.
0	1	8492_H	Die Datei existiert bereits. Die Datei wird nicht überschrieben.
0	1	8496_H	Beim Lesen des SSL CA-Zertifikats ist ein Problem aufgetreten.
0	1	8497_H	Ein unerwarteter Fehler ist bei der SSH-Sitzung aufgetreten.
0	1	8498_H	Es war nicht möglich, die SSL-Verbindung abzubauen.
0	1	8499_H	Der Socket ist nicht bereit zum Senden/Empfangen. Warten Sie, bis die Bereitschaft besteht und versuchen Sie es erneut.
0	1	8501_H	Die Überprüfung des SSL-Zertifikates vom FTP-Server ist fehlgeschlagen.
0	1	8507_H	Während der aktiven FTP-Sitzung ist im Rahmen des Verbindungsaufbaus beim Warten auf den FTP-Server ein timeout aufgetreten.
0	1	$8F55_H$	Header-Status-Bit: Locked
0	1	$8F56_H$	Das NEW-Bit im File-DB-Header wurde nicht zurückgesetzt
0	1	$8F6B_H$	Mögliche Ursachen: • Falscher Wert für den Parameter CMD Werte von 0 bis 15 sind zugelassen. • Ein Kommando des FB 40 wird nicht unterstützt. Mögliche Ursache: Falsche Firmware des CP Abhilfe: Firmware-Update (bei älteren CPs statt des FB 40 die Funktionen FC 40...FC 44 benutzen.)
0	1	$8F7F_H$	Interner Fehler; beispielsweise unzulässige ANY-Referenz

Siehe auch

Übersicht FTP_CMD (Seite 6964)

Aufbau und Verwendung des File-DB - FTP-Client-Betrieb

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten als "Array [..] of Byte" oder "Array [..] of Char" (besitzt variable Länge und Struktur)

Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen, um den FTP-Zugriff zu ermöglichen:

- In den Projektierungsdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Deaktivieren Sie bei allen als File-DB verwendeten Datenbausteinen das Attribut "Optimierter Bausteinzugriff".

File-DB-Header für FTP-Client-Betrieb

Der hier beschriebene File-DB Header ist identisch zu dem für den Server-Betrieb beschriebenen File-DB-Header.

Parameter	Typ	Wert / Bedeutung	Versorgung
EXIST	BOOL	Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält. Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1. • 0: Der File-DB enthält keine gültigen Nutzdaten (Datei existiert nicht). • 1: Der File-DB enthält gültige Nutzdaten (Datei existiert).	Das FTP-Kommando "DELETE" setzt EXIST=0. Das FTP-Kommando "STORE" setzt EXIST=1.
LOCKED	BOOL	Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB. • 0: Auf den File-DB kann zugegriffen werden. • 1: Der File-DB ist gesperrt.	Die FTP-Kommandos "STORE" und "RETRIEVE" setzen während der Bearbeitung LOCKED=1, wenn das Bit zuvor auf 0 stand. Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes ebenfalls LOCKED setzen bzw. zurücksetzen. Dadurch wird zur Konsistenzsicherung eine gegenseitige Verriegelung zwischen Anwenderprogramm und FTP-Abwicklung bewirkt. Empfehlung zur Vorgehensweise im Anwenderprogramm: 1. LOCKED-Bit prüfen (wenn = 0) 2. WRITEACCESS-Bit=0 setzen 3. LOCKED-Bit prüfen (wenn = 0) 4. LOCKED-Bit=1 setzen 5. Daten schreiben 6. LOCKED-Bit=0 setzen

Parameter	Typ	Wert / Bedeutung	Versorgung
NEW	BOOL	Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. • 0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert. • 1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.	FTP-Kommando store setzt nach der Bearbeitung NEW=1 Das Anwenderprogramm in der S7-CPU muss nach dem Lesen der Daten NEW=0 setzen, um ein erneutes Kommando "RETRIEVE" zu ermöglichen.
WRITE_ACCESS	BOOL	• 0: Das Anwenderprogramm hat Schreibrecht für die File-DBs in der S7-CPU. • 1: Das Anwenderprogramm hat kein Schreibrecht für die File-DBs in der S7-CPU.	Das Bit wird bei der DB-Projektierung auf einen Initialisierungwert gesetzt. Empfehlung: Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.
ACT_LENGTH	DINT	Aktuelle Länge des Nutzdatenbereiches. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.
MAX_LENGTH	DINT	Maximale Länge des Nutzdatenbereichs (Länge des gesamten DB abzüglich 20 Byte Header).	Die maximale Länge sollte bei der DB-Projektierung festgelegt werden. Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.
FTP_REPLY_CODE	INT	Vorzeichenlose Zahl (16 Bit), die den letzten Reply-Code von FTP als Binärwert enthält. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Wird vom FTP-Protokoll-Abwickler bei der FTP-Kommandobearbeitung des Servers aktualisiert.
DATE_TIME	DATE_AND_TIME	Datum und Zeit der letzten Änderung des Files. Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.	Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert. Wenn die Funktion "Uhrzeitweiterleitung" genutzt wird, dann entspricht der Eintrag der weitergeleiteten Zeit. Wenn die Funktion "Uhrzeitweiterleitung" nicht genutzt wird, dann wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des CP (Initialisierungswert: 01.01.1994 00:00h).

Siehe auch

Datenbaustein FILE_DB_HEADER als Vorlage - FTP-Client-Betrieb (Seite 6980)

Datenbaustein FILE_DB_HEADER als Vorlage - FTP-Client-Betrieb

Bedeutung

Der hier beschriebene File-DB-Header ist in den Betriebsarten für FTP-Client-Betrieb und FTP-Server-Betrieb identisch.

Für das Anlegen eines File-DB-Header ist der Datentyp FILE_DB_HEADER vordefiniert.

Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

Gehen Sie wie folgt vor:

1. Legen Sie in der CPU, in der Sie das Anwenderprogramm mit den FTP-Anweisungen erstellen, einen Datenbaustein vom Typ "Global-DB" an.
2. Öffnen Sie den Eigenschaftendialog des neu angelegten Datenbausteins (Kontextmenü), deaktivieren Sie das Attribut "Optimierter Bausteinzugriff" und schließen Sie den Eigenschaftendialog wieder.
3. Selektieren Sie im Bausteineditor des DB die Zeile, die Sie als Startzeile für den File-DB verwenden werden.
4. Wählen Sie in der Spalte "Datentyp" ein Strukturelement vom Typ "FILE_DB_HEADER" oder bzw. geben Sie den Typ "FILE_DB_HEADER" über die Tastatur ein.
   Ergebnis: Es wird eine Datenstruktur mit der für den File-DB benötigte Header-Struktur angelegt.
5. Setzen Sie den Parameter "WRITE_ACCESS" auf "true", um den Zugriff zu ermöglichen.
6. Geben Sie einen Wert für die Länge der Nutzdaten am Parameter "MAX_LENGTH" ein, vgl. nächster Punkt.
7. Legen Sie darunter für die zu übertragenden Nutzdaten ein Datenfeld vom Typ "Array [..] of Byte" oder "Array [..] of Char" an.
   Die Größe des Feldes muss der Angabe von MAX_LENGTH im Header entsprechen.

Hinweis

Funktion "Neuen Baustein hinzufügen" - Typauswahl

Beim Anlegen neuer Datenbausteine wird unter dem Eintrag "Typ" in der Klappliste auch der Bausteintyp "FILE_DB_HEADER" angeboten. Verwenden Sie diese Auswahlmöglichkeit nicht! Der so erzeugte Datenbaustein enthält lediglich die Header-Struktur und kann nicht um den benötigten Bereich zur Ablage von Nutzdaten erweitert werden.

Datenbaustein FILE_DB_HEADER - Beispiel und Vorlage für den File-DB-Header

In der Deklarationsansicht erkennen Sie folgende Struktur:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	bit08	BOOL	FALSE	reserviert
+0.1	bit09	BOOL	FALSE	reserviert
+0.2	bit10	BOOL	FALSE	reserviert
+0.3	bit11	BOOL	FALSE	reserviert
+0.4	bit12	BOOL	FALSE	reserviert
+0.5	bit13	BOOL	FALSE	reserviert
+0.6	bit14	BOOL	FALSE	reserviert
+0.7	bit15	BOOL	FALSE	reserviert
+1.0	EXIST	BOOL	FALSE	wenn TRUE: File-DB enthält gültige Daten
+1.1	LOCKED	BOOL	FALSE	wenn TRUE: Der File-DB ist blockiert aufgrund einer inhaltlichen Änderung.
+1.2	NEW	BOOL	FALSE	wenn TRUE: Der Inhalt des File-DB ist neu und darf nicht überschrieben werden.
+1.3	WRITE_ACCESS	BOOL	FALSE	wenn TRUE: Der FTP-Server hat Schreib-Zugriff.
+1.4	bit04	BOOL	FALSE	reserviert
+1.5	bit05	BOOL	FALSE	reserviert
+1.6	bit06	BOOL	FALSE	reserviert
+1.7	bit07	BOOL	FALSE	reserviert
+2.0	ACT_LENGTH	DINT	L#0	Aktuelle Länge des Inhalts in Byte (nicht eingeschlossen 20 Byte für den Header)
+6.0	MAX_LENGTH	DINT	L#0	Maximale Länge des Inhalts in Byte (nicht eingeschlossen 20 Byte für den Header)
+10.0	FTP_REPLY_CODE	INT	0	Wird vom FTP-Protokoll-Abwickler bei der FTP-Kommandobearbeitung aktualisiert, letzte Antwortinformation vom FTP-Server.
+12.0	DATE_TIME	DATE_AND_TIME	DT#00-1-1-0:0:0.000	Datum und Zeitangabe für die letzte Änderung am Inhalt des File-DB.
=20.0		END_STRUCT		

S7-1200-CM/CP

Telecontrol

Telecontrol-Anweisungen

TC_CON: Verbindung über das GSM-Netz aufbauen

Bedeutung

Die Anweisung TC_CON ermöglicht einer S7-1200 mit CP 1242-7 den Aufbau einer Verbindungen der folgenden Typen:

- ISO-ON-TCP
  Verbindungspartner ist ein CP 1242-7.
  ISO-ON-TCP-Verbindungen werden nur in der Betriebsart "GPRS direkt" verwendet.
- UDP
  Der Verbindungspartner ist beliebig.
- SMS
  Verbindungspartner ist ein SMS-Client.
- Telecontrol-Verbindung
  Verbindungspartner ist entweder ein Telecontrol-Server oder eine andere Station, die über den Telecontrol-Server erreichbar ist.

Ein TC_CON baut genau eine Verbindung auf. Abhängig von der Betriebsart des CP 1242-7 und dem verwendeten Protokoll werden pro CP maximal 3 bis 5 parallele Verbindungen mit eindeutigen IDs (siehe unten) unterstützt. Die maximal mögliche Anzahl paralleler Verbindungen entnehmen Sie den Leistungsdaten des CP.

Der Parameter CONNECT verwendet zur Verbindungsbeschreibung einen Datenbaustein (DB) mit der Struktur eines Systemdatentyps (SDT).

Der gewünschte Verbindungstyp wird über einen jeweils verbindungsspezifischen SDT "TCON_..." (siehe unten) definiert. Für jeden der oben genannten Verbindungstypen muss einer der folgenden SDTs parametriert werden:

- TCON_IP_RFC für ISO-ON-TCP-Verbindungen
- TCON_IP_V4 für UDP-Verbindungen
- TCON_PHONE für SMS-Verbindungen
- TCON_WDC für Telecontrol-Verbindungen

Im Parameter "ActiveEstablished" dieser SDTs wird unter anderem auch festgelegt, ob es sich um einen aktiven oder passiven Verbindungaufbau handelt.

Zur Parametrierung dieser SDTs siehe TCON_...: SDTs für den Telecontrol-Verbindungsaufbau (Seite 6998).

Der Parameter ID referenziert die GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Diese muss aus STEP 7 übernommen werden.

Aufrufschnittstelle in FUP-Darstellung

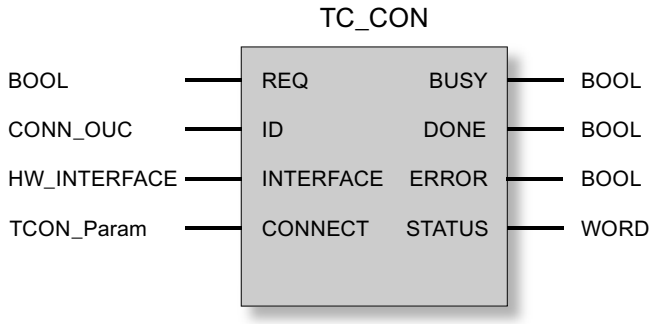

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_CON.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	$1...07FF_h$	Referenz auf die jeweilige Verbindung. Die ID wird vergeben. Der Wert von ID wird auch vom jeweiligen Systemdatentyp (SDT) des Parameters CONNECT benötigt.
INTERFACE	INPUT	HW_INTERFACE		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
CONNECT	INOUT	TCON_Param	Siehe auch "TCON_...: SDTs für den Telecontrol-Verbindungsaufbau"	Referenz auf einen Datenbaustein für den Verbindungsaufbau. Die SDTs vom Typ TCON_IP_RFC, TCON_IP_V4, TCON_PHONE oder TCON_WDC geben die Struktur des passenden Datenbausteins für die jeweilige Verbindung vor. Beachten Sie bei den SDTs den Parameter "ActiveEstablished" (aktiver / passiver Verbindungsaufbau).
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1. In folgenden Fällen kann der Zustand BUSY = 1 länger andauern:

- Bei aktiven ISO-ON-TCP-Verbindungen, wenn der Partner nicht erreichbar ist.
- Bei passiven Verbindungen, wenn kein Telegramm empfangen wird.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	8087_H	Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich

DONE	ERROR	STATUS	Bedeutung
0	1	$80E3_H$	Die ID wird bereits für eine andere Verbindung verwendet. Das heißt, bei TC_SEND, TC_RECV oder TC_DISCON ist BUSY derzeit TRUE. Die Statusanzeige wird auch ausgegeben, wenn EN_R bei TC_RECV dauerhaft TRUE ist. Dies bewirkt, dass TC_RECV größtenteils aufgerufen ist. Abhilfe für diesen Fall: Schalten Sie EN_R aus, bevor TC_CON oder TC_DISCON aufgerufen wird. EN_R darf erst wieder eingeschaltet werden, wenn TC_CON fehlerfrei ausgeführt wurde.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80E8_H$	Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter. In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.
0	1	$80EB_H$	Anfrage vorübergehend zurückgewiesen (TC_CON wurde bereits mit der selben Zieladresse aufgerufen.)
0	1	$80EC_H$	Öffnen des Listener Port konnte nicht durchgeführt werden: Prüfen Sie die Verbindungsparameter.
0	1	$80F2_H$	Der CP befindet sich in der falschen Betriebsart: • Telecontrol-Verbindungen sind nur in der Betriebsart "Telecontrol" erlaubt. • ISO-ON-TCP-Verbindungen sind nur in der Betriebsart "GPRS direkt" erlaubt.
0	1	$80F3_H$	Kein freier Verbindungsendpunkt zum Senden von Daten: • Nutzen Sie weniger Verbindungen oder • Nutzen Sie weniger passive Verbindungen oder • Schalten Sie NTP aus. Beachten Sie die maximale Anzahl paralleler Verbindungen des CP 1242-7.
0	1	$80F4_H$	Verbindungsendpunkt kann nicht erzeugt werden: Wiederholen Sie den Aufruf. Prüfen Sie ggf. die Verbindungsparameter.
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: Verbindungsaufbau durch TC_CON fehlgeschlagen. Wiederholen Sie den Bausteinaufruf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format, ungültiger Wert oder Rufnummer im TCON_Phone größer als 20 Zeichen) Prüfen Sie die Projektierung des SDT "TC_CON...".

TC_DISCON: Verbindung über das GSM-Netz abbauen

Bedeutung

Die Anweisung TC_DISCON bewirkt bei einer S7-1200 mit CP 1242-7 den Abbau einer ISO-ON-TCP-, UDP, SMS- oder Telecontrol-Verbindung, die mit der Anweisung TC_CON aufgebaut wurde.

Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

TC_DISCON baut die Verbindung zum Telecontrol-Server nur logisch ab. Auf TCP/IP-Ebene bleibt die Verbindung erhalten.

Wenn die Verbindung zum Telecontrol-Server physikalisch abgebaut werden soll, dann projektieren Sie die Verbindung in STEP 7 unter der Parametergruppe "Telecontrol-Server"

als "Temporäre Verbindung". Temporäre Stationen bauen die Verbindung nach dem Senden der Daten automatisch ab.

Hinweis

Abbruch der Bearbeitung weiterer Programmbausteine durch TC_DISCON

Ein Aufruf von TC_DISCON beendet die Bearbeitung von TC_CON-, TC_SEND- und TC_RECV-Bausteinen, die mit derselben Verbindungs-ID (Parameter "ID") und Schnittstelle (Parameter "INTERFACE") aufgerufen wurden. Diese Bausteine melden anschließend ERROR.

Kein Aufruf von TC_DISCON, wenn TC_CON "Error = 1" anzeigt.

Wenn TC_CON "ERROR" anzeigt, dann ist die Verbindung nicht aufgebaut. TC_DISCON darf in diesem Fall nicht aufgerufen werden.

Wenn TC_DISCON in diesem Fall doch aufgerufen wird, dann wird die Verbindungs-ID ("ID") kurrzeitig noch reserviert und ein unmittelbar daraufhin aufgerufener TC_CON würde ERROR und STATUS 80E3 anzeigen.

Der Parameter ID referenziert die GPRS-Verbindung. Die ID muss innerhalb der CPU eindeutig sein und gleich derjenigen ID sein, die bei TC_CON verwendet wird.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Aufrufschnittstelle in FUP-Darstellung

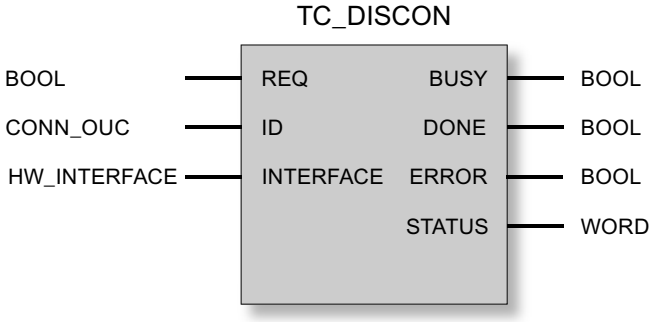

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_DISCON

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	1...07FF$_h$	Referenz auf die jeweilige Verbindung

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
INTERFACE	INPUT	HW_INTER-FACE		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Die Anweisung wurde noch nicht aufgerufen.

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Hinweis

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)

DONE	ERROR	STATUS	Bedeutung
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E4_H$	Unbekannte ID: Keine Verbindung mit dieser ID ist durch TC_CON aufgebaut.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen oder • Verbindung vom entfernten Partner abgebaut.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie die Projektierung des SDT "TC_CON...".

TC_SEND: Daten über das GSM-Netz senden

Bedeutung

Die Anweisung TC_SEND ermöglicht das Senden von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- UDP-Verbindungen
- SMS-Verbindungen
  Das Versenden von SMS wird nur unterstützt, wenn dies in der STEP 7-Projektierung des CP eingerichtet wurde.
- Telecontrol-Verbindungen

Hinweis

SMS an mehrere Empfänger senden

Wenn Sie eine identische SMS an mehrere Empfänger senden wollen, dann müssen Sie für jeden Empfänger eine Verbindung aufbauen.

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC_CON verwendeten Wert für ID entsprechen.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Die Anzahl zu sendender Daten wird mit dem Parameter LEN angegeben.

Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projektierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL.

Die Zieladresse (Verbindungspartner) für die zu sendenden Daten wird in der Anweisung TC_CON projektiert.

Aufrufschnittstelle in FUP-Darstellung

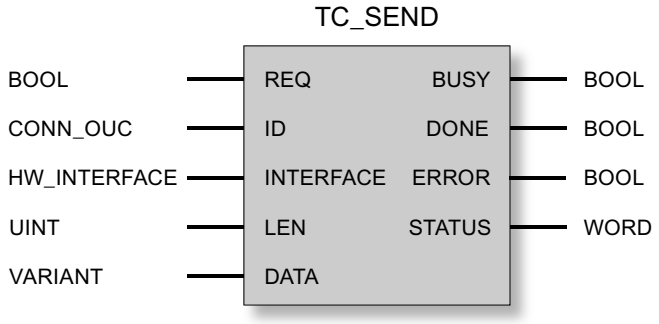

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_SEND.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
ID	INPUT	CONN_OUC	$1...07FF_h$	Referenz auf die jeweilige Verbindung
INTERFACE	INPUT	HW_INTERFACE		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
LEN	INPUT	UINT	1...2048	Anzahl der Bytes der zu sendenden Daten. Der Wert muss ≥ 1 und ≤ 2048 sein. Der Wert sollte der Größe des Bereichs von DATA entsprechen.
DATA	INOUT	VARIANT		Adressverweis auf den zu übertragenden Datenbereich der CPU *
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. ** Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

** Nach dem Senden eines Telegramms setzt TC_SEND DONE = 1. Beachten Sie folgendes Verhalten:
Der Ausfall einer ISO-on-TCP-Verbindungen wird beim Sender erst nach 1 bis 2 Minuten festgestellt. Die übertragenen Daten können verloren sein, obwohl TC_SEND beim Sender DONE = 1 gesetzt hat.
Wenn eine ISO-on-TCP-Verbindung nach dem Empfang eines Telegramms abgebrochen wird, bevor TC_RECV gestartet wurde, dann können die übertragenen Daten verloren sein, auch wenn TC_SEND beim Sender DONE = 1 gesetzt hat.

Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung sendet diejenigen Daten als SMS-Text, auf welche der Zeiger vom Typ VARIANT des Parameters DATA referenziert.

Wenn von DATA auf einen Operanden des Datentyps STRING für SMS-Texte referenziert wird, dann werden die ersten zwei Bytes mit Längenangaben des Strings mit übertragen.

Eine Möglichkeit für die korrekte Textdarstellung von zu sendenden SMS ist die Umwandlung des Text-Strings in ein Array of BYTE oder Array of CHAR mithilfe der Konvertierungsfunktion Strg_TO_Chars. Strg_TO_Chars wird am Parameter EN mit dem Ausgangsparameter ENO von TC_SEND verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS *	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)

DONE	ERROR	STATUS *	Bedeutung
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E0_H$	Interner Fehler Wenn Sie Telegramme direkt an den Telecontrol-Server schicken (Betriebsart "Telecontrol"), dann stellen Sie sicher, dass die Sendezykluszeit ≥ 1 Sekunde ist.
0	1	$80E1_H$	Zeitüberschreitung: • Erhöhen Sie in der Projektierung des CP 1242-7 den Wert der "Verbindungsüberwachungszeit" oder • Überprüfen Sie den Verbindungspartner.
0	1	$80E4_H$	Unbekannte ID: Rufen Sie zuerst TC_CON auf.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80E7_H$	Zu sendende Daten nicht vollständig übertragen: Wiederholen Sie den Auftrag.
0	1	$80E8_H$	Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter. In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.
0	1	$80E9_H$	Verbindungsabbau durch entfernten Partner: Überprüfen Sie den Verbindungspartner. Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.
0	1	$80EA_H$	Fehlermeldung vom entfernten Partner: • Überprüfen Sie den Verbindungspartner. Aktivieren Sie beim Kommunikationspartner die Anweisung "TC_RECV". • Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.
0	1	$80EF_H$	Senden der SMS konnte nicht durchgeführt werden: • Überprüfen Sie, ob die Zieladresse (Rufnummer des Zielteilnehmers) existiert. • Überprüfen Sie, ob die gesteckte SIM-Karte das Senden von SMS unterstützt. • Überprüfen Sie die Länge des versendeten SMS-Texts. SMS-Texte > 160 Zeichen werden nicht gesendet. • Stellen Sie sicher, dass beim Anlegen des Datenbausteins TCON_PHONE die Option "Standard" für den Bausteinzugriff gewählt wurde.
0	1	$80F1_H$	Das Senden von SMS ist in der STEP 7-Projektierung des CP nicht aktiviert: Aktivieren Sie in der Projektierung des CP die Option "SMS ermöglichen".
0	1	$80F4_H$	Verbindungsendpunkt kann nicht erzeugt werden: Überprüfen Sie den Verbindungspartner.
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen. oder • Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert): Prüfen Sie die Projektierung des SDT "TC_CON...".

* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

TC_RECV: Daten über das GSM-Netz empfangen

Bedeutung

Die Anweisung TC_RECV ermöglicht das Empfangen von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- SMS-Verbindungen
  Zum Empfangen von SMS muss die Rufnummer des Absenders in der STEP 7-Projektierung des empfangenden CP projektiert sein (Autorisierte Rufnummern). Der Absender muss die CLIP-Funktion unterstützen.
  Die Rufnummer des Verbindungspartners muss im SDT "TCON_PHONE" eingetragen sein. Weck-SMS werden ausgefiltert.
- Telecontrol-Verbindungen

Hinweis

SMS von verschiedenen Sendern empfangen

Wenn Sie SMS von verschiedenen Sendern empfangen wollen, dann stehen Ihnen hierfür zwei Alternativen zur Wahl:

- Sie müssen mehrere Verbindungen (TC_CON, TC_RECV, TC_DISCON) projektieren.
  oder
- Sie dürfen bei nur einer einzigen projektierten Verbindung in dem erforderlichen Datenbaustein "TCON_PHONE" im Parameter "PhoneNumber" keine Rufnummer eingeben. Dies wird beim Empfangen von Nachrichten als Platzhalter für alle autorisierten Verbindungspartner interpretiert.

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC_CON verwendeten Wert für ID entsprechen.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC_CON für INTERFACE verwendet wird.

Die maximale Anzahl der Empfangsdaten wird mit dem Parameter LEN angegeben.

Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projektierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL. Die empfangenen Daten werden so interpretiert, als hätte der sendende Partner die gleichen Datentypen verwendet.

Der für die Verbindungsbeschreibung des TC_RECV verwendete DB (Systemdatentyp) muss sich von einem für TC_SEND verwendeten DB unterscheiden.

Speichern von SMS

Empfangene SMS werden im CP 1242-7 (25 Speicherplätze) und auf der SIM-Karte (variable Anzahl an Speicherplätze) remanent gespeichert.

- Nach dem Lesen der SMS durch TC_RECV wird die SMS aus ihrem Speicherplatz gelöscht.
- Wenn alle Speicherplätze belegt sind und eine neue SMS empfangen wird, wird die älteste SMS gelöscht.

Aufrufschnittstelle in FUP-Darstellung

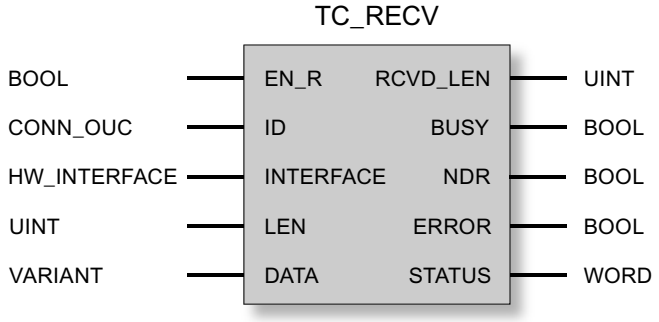

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_RECV

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
EN_R	INPUT	BOOL	0: Datenempfang gesperrt 1: Datenempfang ermöglicht	Ermöglichen / Sperren des Empfangs von Daten. • Bausteinversion 1.1: Nach dem Setzen von 1 auf 0 ist der Baustein inaktiv. • Bausteinversion 1.0: Nach dem Setzen von 1 auf 0 empfängt der Programmbaustein noch einmal Daten (bis DONE = 0 und ERROR = 0). Beachten Sie die Hinweise zur Statusanzeige 80E3 beim TC_CON.
ID	INPUT	CONN_OUC	$1...07FF_h$	Referenz auf die jeweilige Verbindung
INTERFACE	INPUT	HW_INTER-FACE		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
LEN	INPUT	UINT	1...2048	(Mindest-) Azahl der Bytes der zu empfangenden Daten, maximal 2048
DATA	INOUT	VARIANT		Adressverweis auf Empfangs-Datenbereich der CPU *

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
RCVD_LEN	OUTPUT	UINT		Byte-Anzahl der empfangenen Daten
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung referenziert den empfangenen SMS-Text mit dem Zeiger vom Typ VARIANT des Parameters DATA auf den Datenbereich der CPU.

Wenn von DATA auf einen Operanden des Datentyps STRING für den SMS-Text referenziert wird, dann werden die ersten zwei Bytes des SMS-Texts als Längenangaben des Datentyps STRING und nicht als SMS-Text interpretiert.

Eine Möglichkeit für die korrekte Textdarstellung von zu empfangenden SMS ist die Umwandlung eines Array of BYTE oder Array of CHAR in einen Text-String mithilfe der Konvertierungsfunktion Chars_TO_Strg. Chars_TO_Strg wird am Parameter EN mit dem Ausgangsparameter ENO von TC_RECV verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS *	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	$80A3_H$	• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen. • Es wird versucht, eine nicht vorhandene Verbindung zu beenden.
0	1	$80E0_H$	Interner Fehler
0	1	8086_H	Nicht erlaubter Wert für ID
0	1	$80E4_H$	Unbekannte ID: Rufen Sie zuerst TC_CON auf.
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80F5_H$	Ungültiger Verbindungsendpunkt: • Verbindungsaufbau durch TC_CON fehlgeschlagen. oder • Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie die Projektierung des SDT "TC_CON...".

* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

TC_CONFIG: Projektierungsdaten auf CP übertragen

Bedeutung

Mit der Anweisung TC_CONFIG können die in STEP 7 projektierten Parameter eines S7-1200-Mobilfunk-CP geändert werden. Die projektierten Werte werden nicht remanent überschrieben. Die überschriebenen Werte bleiben gültig bis zum erneuten Aufruf von TC_CONFIG oder bis zum nächsten Anlauf der Station (Kaltstart durch Spannung AUS → EIN).

Um die Funktion nutzen zu können, müssen in der STEP 7-Basisprojektierung des CP bereits projektierte Werte vorhanden sein.

Wenn die STEP 7-Projektierungsdaten des CP dauerhaft geändert werden sollen, dann muss die Anweisung nach jedem Anlauf der Station (Kaltstart) neu aufgerufen werden oder ein geändertes Projekt muss in die Station geladen werden.

Der Parameter CONFIG zeigt auf den Speicherbereich mit den Projektierungsdaten. Die Projektierungsdaten werden in einem Datenbaustein (DB) gespeichert. Die Struktur des DB wird durch den Systemdatentyp (SDT) IF_CONF vorgegeben.

Diejenigen Projektierungsdaten, die im CP geändert werden sollen, werden im IF_CONF als Blöcke "IF_CONF_..." für die einzelnen Parameter nach Bedarf zusammengestellt.

Parameter, die sich durch die Anweisung nicht ändern sollten, werden im IF_CONF nicht eingetragen. Sie behalten dann den in STEP 7 projektierten Wert.

Details zur Parametrierung von IF_CONF enthält der Abschnitt IF_CONF: SDT für Telecontrol-Projektierungsdaten (Seite 7002).

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP.

Aufrufschnittstelle in FUP-Darstellung

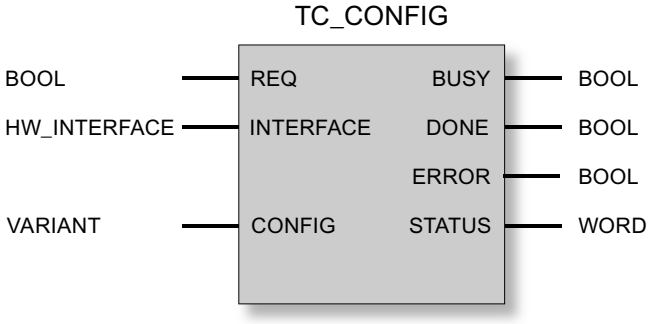

Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC_CONFIG

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
REQ	INPUT	BOOL	0, 1	Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert. Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht.
INTERFACE	INPUT	HW_INTERFACE (WORD)		Referenz auf die Schnittstelle des lokalen CP 1242-7
CONFIG	INOUT	VARIANT	Siehe auch "IF_CONF: SDT für Telecontrol-Projektierungsdaten"	Referenz auf den Speicherbereich mit der Zusammenstellung der zu ändernden Projektierungsdaten
ENO	OUTPUT	BOOL	0: Fehler 1: Fehlerfrei	Freigabeausgang Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.
BUSY	OUTPUT	BOOL	0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen 1: Bearbeitung der Anweisung läuft	Anzeige des Bearbeitungs-Status der Anweisung

Parameter	Deklaration	Datentyp	Wertebereich	Beschreibung
DONE	OUTPUT	BOOL	0: - 1: Bearbeitung der Anweisung erfolgreich beendet	Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.
ERROR	OUTPUT	BOOL	0: - 1: Fehler	Fehleranzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.
STATUS	OUTPUT	WORD		Statusanzeige Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.

Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

BUSY	DONE	ERROR	Bedeutung
0	0	0	Kein Auftrag in Bearbeitung

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

DONE	ERROR	STATUS	Bedeutung
1	0	0000_H	Auftrag fehlerfrei ausgeführt
0	0	7000_H	Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)
0	0	7001_H	Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)
0	0	7002_H	Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)
0	1	$80E0_H$	Interner Fehler (bspw. kein Wert in der Basisprojektierung eines Parameters vorhanden)
0	1	$80E6_H$	Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)
0	1	$80EB_H$	Anfrage vorübergehend zurückgewiesen (der CP wird momentan von STEP 7 konfiguriert.)
0	1	$80F6_H$	Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert) Prüfen Sie den SDT "IF_CONF".
0	1	$80F7_H$	Falsche ID in den Parameterblöcken der Projektierungsdaten: Prüfen Sie den SDT "IF_CONF".

TCON_...: SDTs für den Telecontrol-Verbindungsaufbau

Systemdatentypen TCON_... für Telecontrol-Verbindungen

Die vier nachfolgend beschriebenen Systemdatentypen TCON_... werden unter anderem für die Verbindungsbeschreibung von Telecontrol-Verbindungen des CP 1242-7 verwendet.

TCON_PHONE wird auch bei Mobilfunk-CPs der S7-1200 für die Beschreibung einer Verbindung der Open User Communication zur Übertragung von SMS verwendet (TCON / TDISCON / TSEND oder TSEND_C).

Um für den CP 1242-7 den Aufbau einer Telecontrol-Verbindung mithilfe der Anweisung TC_CON zu projektieren, wird der Parameter CONNECT der Anweisung für die Verbindungsbeschreibung verwendet.

Die Verbindungsbeschreibung ist durch die Struktur eines Systemdatentyps (SDT) vorgegeben. Die Struktur des jeweiligen SDT enthält die notwendigen Parameter, die zum Aufbau der Verbindung mit dem entfernten Kommunikationspartner erforderlich sind.

Für unterschiedliche Verbindungstypen, die vom jeweiligen entfernten Kommunikationspartner abhängen, werden folgende SDTs verwendet:

- TCON_IP_RFC für ISO-on-TCP-Verbindungen zu IPv4-Stationen mit CP 1242-7
- TCON_IP_V4 für UDP-Verbindungen zu IPv4-Stationen (nur Senden)
- TCON_PHONE für Verbindungen zu SMS-Clients
- TCON_WDC für Verbindungen zu Telecontrol-Servern oder Stationen, die über den Telecontrol-Server erreichbar sind.

Die Parametrierung der Verbindungsbeschreibung wird in einem Datenbaustein des gleichen Typs wie der SDT vorgenommen.

Anlegen eines DB vom Typ TCON_...

Die Datentypen der jeweiligen DBs müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen eines TCON_...-DB folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB einen SDT an, indem Sie den Namen vergeben und in die Zelle des Datentyps den gewünschten Typ (bspw. "TCON_IP_RFC") eintippen.
   Der SDT mit seinen Parametern (siehe unten) wird angelegt.
3. Projektieren Sie die Parameter, die nachfolgend für jeden SDT-Typ beschrieben sind.

Reservierte Bits werden nicht angezeigt.

Systemdatentyp TCON_IP_RFC für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur bei ISO-on-TCP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt. Der CP muss für die Betriebsart "GPRS direkt" projektiert sein.

Tabelle 4-396 Parameter von TCON_IP_RFC

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0C	Protokoll-Variante 12 (C$_h$): ISO-on-TCP-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau • 1: Aktiver Verbindungsaufbau
6 ... 7	-	-	-	- reserviert -
8 ... 11	RemoteAddress	IP_V4		IP-Adresse der Verbindungspartner
	ADDR	Array [1...4] of Byte		IP-Adresse des jeweiligen Verbindungspartners
12 ... 13	RemoteTSelector	TSelector		Entfernter T-Selektor
	TSelLen	UINT		Länge des entfernten T-Selektors "RemoteTSelector"
14 ... 45	TSel	Array [1...32] of Byte	beliebig	Entfernter Transport-Selektor der Verbindung • Bei "ActiveEstablished" = 1: Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner). • Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)
46 ... 47	LocalTSelector	TSelector		Lokaler T-Selektor
	TSelLen	UINT		Länge des lokalen T-Selektors "LOCAL_TSel"
48 ... 79	TSel	Array [1...32] of Byte	beliebig	Lokaler Transport-Selektor der Verbindung • Bei "ActiveEstablished" = 1: Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner). • Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)

Systemdatentyp TCON_IP_V4 für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur zum Senden über UDP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt.

Wenn empfangen werden soll, dann muss ActiveEstablished = 0 gesetzt werden.

Tabelle 4-397 Parameter von TCON_IP_V4

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0B	Protokoll-Variante 11 (B_h): UDP-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau Einstellung zum Senden und Empfangen von Daten. • 1: Aktiver Verbindungsaufbau Einstellung nur zum Senden von Daten.
6 ... 9	RemoteAddress	IP_V4		IP-Adresse des Verbindungspartners
	ADDR	Array [1...4] of Byte		Die vier Bytes (ADDR[1] ... ADDR[4]) spezifizieren die vier Blöcke der IP-Adresse.
10 ... 11	RemotePort	UINT	1...65535	IP-Port des Verbindungspartners Nicht relevant, wenn ActiveEstablished = 0.
12 ... 13	LocalPort	UINT	1...65535	Lokaler IP-Port ("0" ist nicht erlaubt.) Nicht relevant, wenn ActiveEstablished = 1.

Systemdatentyp TCON_PHONE für SMS-Verbindungen

Hinweis

Autorisierte Rufnummern

Voraussetzung dafür, dass der CP eine SMS akzeptiert, ist die Autorisierung des sendenden Kommunikationspartners mithilfe seiner Rufnummer. Diese Rufnummern werden in STEP 7 beim CP in der Liste "Autorisierte Rufnummern" projektiert.

SMS-Text

- Auf programmierte SMS-Texte für zu sendende SMS wird über den Parameter DATA der Anweisung TC_SEND zugegriffen.
- Der Text einer empfangenen SMS wird dem Adressbereich der CPU über den Parameter DATA der Anweisung TC_RECV zugewiesen.

Tabelle 4-398 Parameter von TCON_PHONE

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die lokale Schnittstelle (Telecontrol-Schnittstelle bzw. Ethernet-Schnittstelle des CP > "HW-Kennung")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale Mobilfunk-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0E	Protokoll-Variante 14 (E$_h$): SMS-Verbindung
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus (nicht relevant für den CP 1242-7): • 0: Passiver Verbindungsaufbau (hier nicht relevant) • 1: Aktiver Verbindungsaufbau
6...7	-	-	-	- reserviert -
8 ... 31	PhoneNumber	STRING[22]		Rufnummer des Verbindungspartners Erlaubte Werte: Plus-Zeichen (+) und Ziffern Achten Sie auf die genaue Zeichenfolge der vom Netzbetreiber zugewiesenen Ländervorwahl der betreffenden Rufnummer ("+"-Zeichen oder Nullen). Ohne Eintrag des Parameters PhoneNumber wird kein Verbindungspartner spezifiziert und SMS können von allen autorisierten Verbindungspartnern empfangen werden. Beachten Sie beim Anlauf: Ohne Eintrag liefert TC_RECV zunächst die älteste empfangene SMS.

Systemdatentyp TCON_WDC für Verbindungen zu Telecontrol-Servern oder entfernten Stationen

Sie können die Verbindung mit dem Telecontrol-Server, welcher der S7-1200 zugeordnet ist, oder zu einer entfernten Station, die über den Telecontrol-Server erreichbar ist, mit dem TCON_WDC projektieren. Die Adressdaten des Telecontrol-Servers, der dem CP zugeordnet ist, finden Sie in STEP 7 im Register "Telecontrol-Schnittstelle > Betriebsart" des CP. Der Telecontrol-Server oder eine entfernte Station wird über den Host-Namen oder die IP-Adresse adressiert.

Im Parameter "RemoteWdcAddress" des TCON_WDC wird die Zugangs-ID des Verbindungspartners angegeben.

Tabelle 4-399 Parameter von TCON_WDC

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	InterfaceID	HW_ANY		Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein. Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.
4	ConnectionType	BYTE	W#16#0F	Protokoll-Variante 15 (F$_h$): Telecontrol-Verbindung über IP-Adressse
5	ActiveEstablished	BOOL		Kennung für die Art des Verbindungsaufbaus: • 0: Passiver Verbindungsaufbau • 1: Aktiver Verbindungsaufbau
6 ... 7	-	-	-	- reserviert -
8 ... 11	RemoteWdcAddress	DWORD		Angabe der Zugangs-ID (hex). Die Zugangs-ID ist abhängig vom Verbindungspartner. • Verbindung zu einem entfernten CP: Die Zugangs-ID wird gebildet aus: – STEP 7-Projektnummer – Stationsnummer – Steckplatz Wenn die entfernte Station mehrere GPRS-CPs hat und der Weg nicht festgelegt werden soll, dann muss das letzte Byte für den Steckplatz gleich 0 gesetzt werden. Die Zugangs-ID finden Sie im STEP 7-Projekt unter der Parametergruppe "CP-Authentifizierung" des CP. • Verbindung zum Telecontrol-Server: Zugangs-ID = 0 • Um nur in das Prozessabbild des CP zu schreiben: Zugangs-ID = DW#16#FEEDDADA

IF_CONF: SDT für Telecontrol-Projektierungsdaten

Aufbau des Systemdatentyps IF_CONF für die Anweisung TC_CONFIG

Der Parameter CONFIG der Anweisung TC_CONFIG referenziert den Speicherbereich mit den zu ändernden Projektierungsdaten des CP. Die in einem Datenbaustein abgelegten Projektierungsdaten werden als Struktur vom Systemdatentyp (SDT) IF_CONF beschrieben.

Um die Funktion nutzen zu können, müssen in der STEP 7-Basisprojektierung des CP bereits projektierte Werte vorhanden sein.

IF_CONF setzt sich aus einem Header und nachfolgenden Blöcken zusammen, die den Parametern oder Parameterbereichen des CP in den Geräteeigenschaften des STEP 7-Projekts entsprechen.

Die zu ändernden Projektierungsdaten des CP werden als IF_CONF-Blöcke zusammengestellt. Nicht zu ändernde Parameter werden in der IF_CONF-Struktur nicht berücksichtigt und bleiben so, wie sie im STEP 7-Projekt konfiguriert wurden.

Anlegen des DB und der IF_CONF-Strukturen

Die Parameter des CP können Sie innerhalb des IF_CONF-DB in einer oder in mehreren Strukturen mit jeweils einem oder mehreren Blöcken anlegen.

Die Datentypen der jeweiligen Blöcke müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen von IF_CONF folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB eine Struktur an (Datentyp "Struct").
   Den Name können Sie frei festlegen.
3. Fügen Sie unter dieser Struktur einen Header ein, indem Sie den Namen des Headers vergeben und in die Zelle des Datentyps "IF_CONF_Header" eintippen.
   Der Header der Struktur mit seinen drei Parametern (siehe unten) wird angelegt.
4. Legen Sie einen Block für den ersten zu ändernden Parameter an, indem Sie den gewünschten Datentyp (bspw. "IF_CONF_APN") in die Zelle des Datentyps eintippen.
5. Wiederholen Sie den letzten Schritt für all diejenigen Parameter, die Sie mithilfe der Anweisung TC_CONFIG im CP ändern wollen.
6. Aktualisieren Sie abschließend im Header die Blockanzahl im Parameter "subfieldCnt".

Header von IF_CONF

Tabelle 4-400 IF_CONF_Header

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	fieldType	UINT		Blocktyp: Muss immer 0 sein.
2 ... 3	fieldId	UINT		Block-ID: Muss immer 0 sein.
4 ... 5	subfieldCnt	UINT		Gesamtanzahl der in der Struktur enthaltenen Blöcke

Allgemeine Parameter der Parameterblöcke

Jeder Block enthält folgende allgemeine Parameter:

- Id
  Dieser Parameter kennzeichnet den jeweiligen Block und darf nicht verändert werden.
- Length
  Dieser Parameter gibt die Länge des Blocks an. Der Wert dient nur Informationszwecken. Blöcke mit Strings und / oder Arrays haben eine variable Länge. Durch versteckte Bytes kann die tatsächliche Länge von Blöcken größer als die Summe der angezeigten Parameter sein.
- Mode
  Für diesen Parameter sind die folgenden Werte zulässig:

Tabelle 4-401 Werte von "Mode"

Wert	Bedeutung
1	Permanente Gültigkeit der Projektierungsdaten Nicht relevant beim CP
2	Temporäre Gültigkeit der Projektierungsdaten einschließlich Löschen vorhandener permanenter Projektierungsdaten Die permanenten Projektierungsdaten werden durch die Parameterblöcke von IF_CONF ersetzt.

Block für den Parameterbereich "GPRS-Zugang"

Tabelle 4-402 IF_CONF_APN

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	4	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 174
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
AccesspointGPRS	STRING [98]		APN: Name des Zugangspunkts des GSM-Netzwerkbetreibers zum Internet
AccesspointUser	STRING [42]		APN-Benutzername
AccesspointPassword	STRING [22]		APN-Passwort

Block für den Parameterbereich "CP-Identifikation"

Tabelle 4-403 IF_CONF_Login

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	5	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 54
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)

Parameter	Datentyp	Anfangswert	Beschreibung
ModemName	STRING [22]		Zugangs-ID Der Wert ist nicht parametrierbar.
ModemPassword	STRING [22]		Telecontrol-Passwort (max. 20 Zeichen) Das Passwort ist beim CP 1242-7 (6GK7 242-7KX30-0XE0) nicht über den SDT änderbar.

Block für den Parameterbereich "Telecontrol-Server-Zugang"

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit einem über DNS auflösbaren Namen adressiert wird. Wenn der Telecontrol-Server mit seiner IP-Addresse adressiert wird, dann wird der Block "IF_CONF_TCS_IP_V4" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-404 IF_CONF_TCS_Name

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	6	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 266
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
TcsName	-	-	- reserviert -
	STRING [254]		Durch DNS auflösbarer Name des Telecontrol-Servers
RemotePort	UINT		Port des Telecontrol-Servers
Rank	UINT		Priorität des Servers [1, 2] 1 = Erster Telecontrol-Server, 2 = Zweiter Telecontrol-Server

Block für den Parameterbereich "Telecontrol-Server-Zugang"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit seiner IP-Addresse adressiert wird. Wenn der Telecontrol-Server mit seinem DNS-Namen adressiert wird, dann wird der Block "IF_CONF_TCS_Name" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-405 IF_CONF_TCS_IP_v4

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	7	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
RemoteAddress	IP_V4		IP-Adresse des Telecontrol-Servers

Parameter	Datentyp	Anfangswert	Beschreibung
RemotePort	UINT		Port des Telecontrol-Servers
Rank	UINT		Priorität des Servers [1, 2] 1 = Erster Telecontrol-Server, 2 = Zweiter Telecontrol-Server

Block für den Parameterbereich "Betriebsart"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Betriebsart" und "Modem-Einstellungen".

Tabelle 4-406 IF_CONF_GPRS_Mode

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	8	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 10
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
GPRSmode	UINT		Betriebsart des CP: • 0 = Telecontrol • 1 = GPRS direkt
TemporaryStation	BOOL		Bit 0: Temporäre Verbindung Bei aktivierter Option baut der CP eine Verbindung nur zeitweilig zum Senden von Daten auf. Nach erfolgreichem Übertragen der Telegramme baut der CP die Verbindung wieder ab. • 1: aktiviert (temporäre Verbindung) • 0: deaktiviert (permanente Verbindung)
SMS_Enabled	BOOL		Bit 1: SMS ermöglichen Die Aktivierung der Option ermöglicht der S7-Station das Versenden von SMS. • 1: aktiviert (SMS ermöglicht) • 0: deaktiviert (keine SMS)

Block für den Parameter "SMSC"

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 4-407 IF_CONF_SMS_Provider

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	10	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 28

Parameter	Datentyp	Anfangswert	Beschreibung
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
SMSProvider	STRING [20]		Teilnehmernummer der SMS-Zentrale (SMSC) des GSM-Netzwerkbetreibers, mit dem der Mobilfunk-Vertrag für diese Station abgeschlossen wurde.

Block für den Parameter "PIN"

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 4-408 IF_CONF_PIN

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	11	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 16
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
Pin	STRING [8]		PIN der im CP gesteckten SIM-Karte Der Parameter ist nicht relevant, wenn die PIN richtig projektiert wurde. Bei falsch projektierter PIN kann die richtige PIN hiermit eingegeben werden.

Block für Überwachungszeiten

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Keep-Alive-Wartezeit" und "Betriebsart" des CP.

Tabelle 4-409 IF_CONF_TC_Timeouts

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	12	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 12
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
KeepAliveTimeout	-	-	- Reserviert - (nicht parametrierbar)

Parameter	Datentyp	Anfangswert	Beschreibung
SendTimeout	UINT		Verbindungsüberwachungszeit: Überwachungszeit der Verbindung zum Kommunikationspartner (Sekunden) Relevant in den Betriebsarten "Telecontrol" und "GPRS direkt"
RedialTimeout	UINT		Verbindungsaufbau-Verzögerung: Basiswert für die Wartezeit bis zum nächsten Verbindungsaufbauversuch nach erfolglosem Verbindungsaufbau. Nach jeweils 3 Versuchen wird der Basiswert bis max. 900 s verdoppelt. Wertebereich: 10...600 s. Bsp.: Basiswert 20 ergibt folgende Wählintervalle: dreimal 20 s, dreimal 40 s, dreimal 80 s etc. bis max. 900 s. Wenn ein zweiter Telecontrol-Server / Router projektiert ist, dann versucht der CP beim 4. Mal, sich mit dem zweiten Partner zu verbinden. Wenn der zweite Partner auch nicht erreichbar ist, dann versucht der CP beim 7. Mal, sich wieder mit dem ersten Partner zu verbinden. Nicht relevant für SMS-Verbindungen

Block für den Parameterbereich "Weckberechtigung"

Tabelle 4-410 IF_CONF_WakeupList

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	13	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 246
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
WakeupPhone [1...10]	ARRAY [1...10] of STRING [22]		Rufnummer des zum Wecken autorisierten Teilnehmers Der Stern (*) am Ende einer Rufnummer dient als Platzhalter für Durchwahlnummern.

Block für den Parameterbereich "Bevorzugte GSM-Netze"

Tabelle 4-411 IF_CONF_PrefProvider

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	14	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 46
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
Provider [1...5]	ARRAY [1...5] of STRING [6]		Alternative GSM-Netze mit Priorität 1 bis 5, in die sich der CP bevorzugt einwählt. Bis zu 5 Netze sind projektierbar. Nr. 1 mit höchster Priorität, Nr. 5 mit niedrigster Priorität. Eingabe des Public Land Mobile Network (PLMN) des Netzwerkbetreibers, bestehend aus Mobile Country Code (MCC) und Mobile Network Code (MNC). Beispiel (Testnetz der Siemens AG): 26276

Block für den Parameterbereich "DNS-Konfiguration"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Tabelle 4-412 IF_CONF_DNS

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	16	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
DNS_IP [1]	IP_V4		IP-Adresse des 1. Domain Name System Servers
DNS_IP [2]	IP_V4		IP-Adresse des 2. Domain Name System Servers

Block für den Parameterbereich "Uhrzeitsynchronisation"

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

Tabelle 4-413 IF_CONF_NTP

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	17	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 24
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
NTP_IP [1]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 1
...	...		(IP-Adresse von NTP-Server 2...3)
NTP_IP [4]	ARRAY [1...4] of IP_V4		IP-Adresse von NTP-Server 4

Block zur Aktivierung / Deaktivierung von TeleService-Benutzern

Der SDT kann nur beim CP 1242-7 (6GK7 242-7KX30-0XE0) zur Konfiguration zur Laufzeit verwendet werden.

SDT zur Aktivierung oder Deaktivierung von TeleService-Benutzern, die bereits im STEP 7-Projekt des CP projektiert sind. In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen" > "TeleService-Benutzerverwaltung".

Tabelle 4-414 IF_CONF_GPRS_UserList

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	19	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 506
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
GPRS_User [1...10]	ARRAY [1...10] of GPRS_User		TeleService-Benutzer Nr. 1 bis max. Nr. 10

Der Array wird gebildet aus den Parametersätzen für die TeleService-Benutzer ("GPRS_User" [1...n]).

Tabelle 4-415 GPRS_User [n] (Parameter für TeleService-Benutzer)

Parameter	Datentyp	Anfangswert	Beschreibung
UserName [n]	STRING [22]		TeleService-Benutzername
Password [n]	STRING [22]		- Der String muss leer sein! -
Diag_Allowed [n]	BOOL		- Reserviert - (nicht parametrierbar)
Teleserv_Allowed [n]	BOOL		Aktivierung des TeleService-Benutzers • 0 = Benutzer ist deaktiviert • 1 = Benutzer ist aktiviert
FW_Load_Allowed [n]	BOOL		- Reserviert - (nicht parametrierbar)

Block für die Parametrierung des TeleService-Zugangs (DNS-Name des Servers)

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-416 IF_CONF_TS_Name

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	20	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 266
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)
ts_name	String [254]		Durch DNS auflösbarer Name ders TeleService-Servers
RemotePort	UINT		Port der Engineering-Station
Rank	UINT		Priorität des Servers [1] oder [2] 1 = Server 1, 2 = Server 2

Block für die Parametrierung des TeleService-Zugangs (IP-Adresse des Servers)

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 4-417 IF_CONF_TS_IF_V4

Parameter	Datentyp	Anfangswert	Beschreibung
Id	UINT	21	Kennung des Parameterblocks
Length	UINT		Länge des Parameterblocks in Byte: 14
Mode	UINT		Gültigkeit (1: permanent, 2: temporär)

Parameter	Datentyp	Anfangswert	Beschreibung
RemoteAddress	IP_V4		IP-Adresse des TeleService-Servers
RemotePort	UINT		Port des TeleService-Servers
Rank	UINT		Priorität des Servers [1] oder [2] 1 = Server 1, 2 = Server 2

S7-300/400-CP

CP 443-1 OPC UA

Programmbausteine für den OPC UA-Client

Übersicht der Programmbausteine für die OPC UA-Client-Funktion

Für die OPC UA-Client-Funktion des CP stehen die nachfolgend aufgeführten Funktionsbausteine in der Bibliothek "SIMATIC NET CP" zur Verfügung. Verwenden Sie die Bausteine in der CPU.

Einige Bausteine benötigen spezielle benutzerdefinierte PLC-Datentypen (UDTs), die Sie nachfolgend unterhalb des jeweiligen Programmbausteins finden. Die UDTs stehen als vorgefertigte Datentypen zur Verfügung. Legen Sie hierzu unter dem Eintrag "PLC-Datentypen" der Station einen neuen Datentyp an und tippen Sie im Feld "Datentyp" den Namen des UDT ein, beispielsweise "UA_SessionConnectInfo". Die jeweils benötigten Parameter werden dann automatisch in der Deklarationstabelle des UDT angelegt.

- UA_Connect
  Baut eine Verbindung für eine Sitzung mit einem OPC UA-Server auf.
  - UA_SessionConnectInfo
    Enthält die Verbindungsinformationen für den Bausteinparameter "SessionConnectInfo".
  - UA_UserIdentityToken
    Enthält die Daten der Benutzerauthentifizierung für den Bausteinparameter "UserIdentityToken".
- UA_NamespaceGetIndex
  Holt den Namensraumindex einer Namensraum-URI.
- UA_NodeGetHandleList
  Registriert Node-Ids am verbundenen Server und holt die Node-Handles in Form einer Liste.
  - UA_NodeID
    Enthält die Parameter zur Identifizierung des Ziel-Knotens für den Bausteinparameter "NodeID".
- UA_NodeReleaseHandleList
  Gibt am Server die Node-Handles der verwendeten Liste frei.
- UA_Disconnect
  Baut die Verbindung einer aktuellen Sitzung mit einem OPC UA-Server ab.

- UA_ReadList
  Liest mithilfe der Liste der Node-Handles die Daten aus Knoten des verbundenen Servers.
  - UA_NodeAdditionalInfo
    Spezifiziert das Item-Attribut und den Indexbereich für den Bausteinparameter "NodeAddInfos".
  - UA_IndexRange
    Legt den Start- und End-Index für den Bausteinparameter "IndexRange" fest.
  - UA_AnyPointer
    Verweist auf den Speicherbereich der CPU, in dem die Prozessdaten für den Bausteinparameter "Variables" abgelegt werden.
  - UA_TimeStamps
    Enthält die Zeitstempel der Elemente des Datenbereichs des Bausteinparameters "Variables".
- UA_WriteList
  Schreibt mithilfe der Liste der Node-Handles Daten in Knoten des verbundenen Servers.
  - UA_NodeAdditionalInfo
    Spezifiziert das Item-Attribut und den Indexbereich für den Bausteinparameter "NodeAddInfos".
  - UA_IndexRange
    Legt den Start- und End-Index für den Bausteinparameter "IndexRange" fest.
  - UA_AnyPointer
    Verweist auf den Speicherbereich der CPU, in dem die Prozessdaten für den Bausteinparameter "Variables" abgelegt sind.

Zusätzlich benötigte Systemfunktionen (SFC)

Für die vollständige Funktion der Programmbausteine für den OPC UA-Client werden folgende Systemfunktionen benötigt:

- BLKMOV (SFC20)
- TIME_TCK (SFC64)

UA_Connect benötigt für seine Funktion zusätzlich:

- LGC_GADR (SFC49)
- RDSYSST (SFC51)

Anlegen der SFCs

Legen Sie die Systemfunktionen an, indem Sie im Bausteinverzeichnis der CPU einen Organisationsbaustein öffnen, bswp. den OB1.

Klappen Sie im Bausteinkatalog, der mit dem Öffnen des OB1 eingeblendet wird, die entsprechende Bausteingruppe auf.

Sie finden die SFCs in folgenden Bausteinordnen:

- BLKMOV (SFC20): Gruppe "Einfache Anweisungen" > Ordner "Verschieben"
- TIME_TCK (SFC64): Gruppe "Erweiterte Anweisungen" > Ordner "Datum und Uhrzeit"

- LGC_GADR (SFC 49): Gruppe "Erweiterte Anweisungen" > Ordner "Adressierung"
- RDSYSST (SFC 51): Gruppe "Erweiterte Anweisungen" > Ordner "Diagnose"

Ziehen Sie die SFCs in das Netzwerk des Organisationsbausteins und versorgen Sie die Parameter der SFCs.

Aufruf und Zusammenspiel der Programmbausteine

Hinweis

Allgemeine Hinweise zu den Programmbausteinen und Instanz-DBs

Legen Sie die Funktionsbausteine mit "Standard-Zugriff" an. Lassen Sie in den jeweils zugeordneten Instanz-Datenbausteinen die Option "Remanenz" deaktiviert.

Jeder Programmbaustein muss zyklisch solange aufgerufen werden, bis der Zustandsparameter "Done" oder "Error" auf 1 wechselt.

Innerhalb einer Sitzung kann gleichzeitig nur ein einziger Baustein aufgerufen werden. Verschiedene Bausteine können innerhalb einer Sitzung nur nacheinander bearbeitet werden.

Mit UA_Connect bauen Sie vom CP als OPC UA-Client eine Verbindung zu einem UA-Server auf und öffnen damit eine Sitzung. Der Baustein muss für jede Verbindung zu einem UA-Server aufgerufen werden. Bei Bedarf für mehr als eine Verbindung zu einem UA-Server kann der Baustein mehrmals mit unterschiedlichen Instanz-Datenbausteinen (IDBs) aufgerufen werden. Sie können dies beispielsweise nutzen, um über eine Verbindung zu lesen und über eine zweite Verbindung zu schreiben oder um über mehrere Verbindungen unterschiedliche Daten zu lesen bzw. zu schreiben.

Für Verbindungen zu mehreren UA-Servern muss der Bausteine UA_Connect mehrmals mit unterschiedlichen Eingangsparametern aufgerufen werden. Der CP kann als UA-Client Verbindungen mit bis zu fünf UA-Servern aufbauen.

Zur Abfrage des Namensraumindex einer Namensraum-URI des UA-Servers rufen Sie innerhalb einer Sitzung einen UA_NamespaceGetIndex auf. Zur Abfrage mehrerer Namensraum-URIs können Sie den Baustein mehrmals mit unterschiedlichen Eingangsparametern aufrufen.

Zum Anlegen einer Handle-Liste als Vorbereitung der Lese- und Schreib-Dienste rufen Sie UA_NodeGetHandleList auf. Der Baustein wird für jeden Ziel-Knoten separat mit den entsprechenden Angaben am Eingangsparameter "NodeIDs" aufgerufen.

Anschließend können Sie über die Bausteine UA_ReadList und UA_WriteList die Daten der Items lesen und schreiben. Die beiden Bausteine können Sie innerhalb einer Sitzung beliebig oft aufrufen.

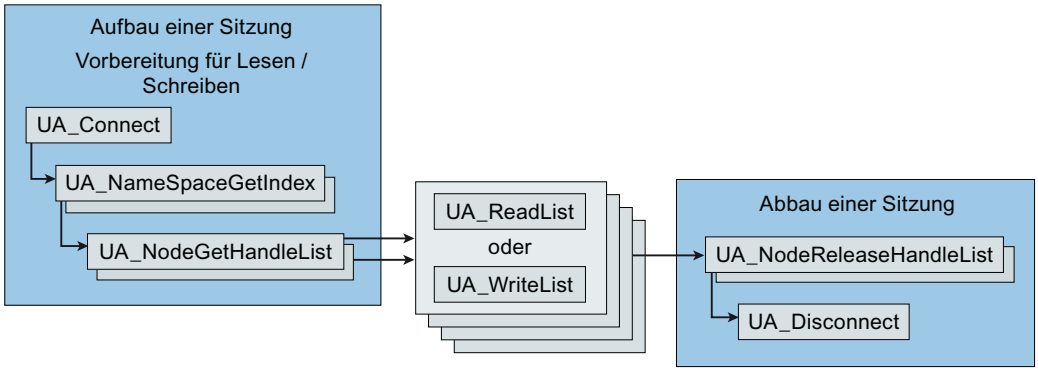

Bild 4-43 Aufruf und Zusammenspiel der Client-Programmbausteine

Wenn die Verbindung nicht mehr benötigt wird, dann geben Sie die Handles am Server über den Baustein UA_NodeReleaseHandleList wieder frei. Die Handles werden damit am Server wieder gelöscht.

Die Verbindung wird mit UA_Disconnect abgebaut und die Sitzung damit beendet.

Siehe auch

Zeitliche Überwachung der Bausteine (Seite 7014)

Zeitliche Überwachung der Bausteine

Parameter für das Zeitverhalten der Bausteine

Für die Steuerung und die Überwachung des zeitlichen Verhaltens der Programmbausteine stehen die folgenden drei Parameter zur Verfügung:

- Timeout
  Eingangsparameter an allen Bausteinen für den OPC UA-Client
- SessionTimeout
  Eingangsparameter am Baustein UA_Connect
- MonitorConnection
  Eingangsparameter am Baustein UA_Connect

Da die Parameter Einfluss auf den Ablauf der OPC UA-Kommunikation insgesamt haben, werden nachfolgend Hinweise zu diesen Parametern vor die allgemeine Bausteinbeschreibung gesetzt.

Timeout

Mit diesem Eingangsparameter überwachen Sie jeden Aufruf eines Bausteins. Wenn ein Bausteinaufruf nicht innerhalb der jeweils projektierten maximal zulässigen Zeit abgeschlossen werden kann, wird der Zustandsparameter "Error" auf 1 gesetzt und die Bearbeitung abgebrochen.

SessionTimeout

Mit diesem Eingangsparameter überwachen Sie die Dauer einer Sitzung ohne Datenverkehr, ohne dass der Baustein "UA_Disconnect" zum Abbau einer Verbindung aufgerufen wird.

Der minimale Wert von "SessionTimeout" beträgt generell 30 Sekunden. Bei Überschreitung des Werts wird die Verbindung abgebaut.

Der Parameter erlaubt einem Server, gebundene Ressourcen abzubauen, wenn der Client die Sitzung für längere Zeit nicht nutzt.

Eine ausreichend lange Zeit gewährleistet dagegen, dass eine Sitzung nicht sofort abgebaut wird, wenn Daten beispielsweise aufgrund von Netzwerkstörungen nicht sofort übertragen werden können.

Der Wert von "SessionTimeout" sollte in der Regel größer sein als die Werte von "Timeout" bei den verwendeten Bausteinen "UA_Connect".

MonitorConnection

Mit diesem Eingangsparameter wird der Zustand einer Verbindung ohne Datenverkehr geprüft. Nach Ablauf dieser Verbindungsüberwachungszeit wird ein Telegramm zur Prüfung des Verbindungszustandes mit dem Server gesendet.

Damit bei fehlendem Datenverkehr und dem Ansprechen der Verbindungsüberwachungszeit nicht gleich die Sitzung über den Parameter "SessionTimeout" abgebrochen wird, wird empfohlen, einen Wert für "MonitorConnection" zu wählen, der kleiner ist als die Hälfte von "SessionTimeout".

Siehe auch

Programmbausteine für den OPC UA-Client (Seite 7011)

UA_Connect

Funktion

Funktion des Bausteins

Mit dem Baustein UA_Connect bauen Sie vom CP als OPC UA-Client eine Verbindung zu einem UA-Server auf und öffnen damit eine Sitzung.

Als Zieladresse geben Sie am Parameter "ServerEndpointUrl" die URL des UA-Servers an.

Die Angaben zu den Verbindungsinformationen am Parameter "SessionConnectInfo" hinterlegen Sie in einem Datenbaustein.

UA_Connect benötigt zusätzlich die Systemfunktionen LGC_GADR (SFC 49) und RDSYSST (SFC 51).

Siehe auch

Parameter (Seite 7016)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-418 Parameter des Bausteins UA_Connect

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ServerEndpointUrl	IN	STRING	Max. 254 Zeichen	Adresse (URL) des Verbindungspartners (Server). Nur IPv4-Adressen sind zugelassen.
SessionConnectInfo	IN	UDT	Siehe "Bedeutung"	Verbindungsinformationen, siehe UASession-ConnectInfo (Seite 7030).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
ConnectionHdl	OUT	DWORD	1 .. 5	Eindeutige Kennzeichnung einer aufgebauten Verbindung Sie wird von den anderen Bausteinen als Eingangsparameter benötigt.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036). Hinweis: Wenn der Fehler-Code B000F002 ausgegeben wird, dann ist die OPC UA-Sitzung noch nicht aufgebaut. Rufen Sie den Baustein erneut auf.

UA_NamespaceGetIndex

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NamespaceGetIndex holen Sie den Namensraumindex der Namensraum-URI des verbundenen UA-Servers.

Siehe auch

Parameter (Seite 7017)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-419 Parameter des Bausteins UA_NamespaceGetIndex

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus. Solange Execute = 1 ist, können die anderen Eingangsparameter nicht verändert werden.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NamespaceUri	IN	STRING	Max. 254 Zeichen	Angabe der Adresse (Namensraum-URI) des Ziel-Servers Beispiel für den Namensraum der CPU-Symbole in der S7-400 mit CP 443-1 OPC UA: http://www.siemens.com/simatic-classic-s7-opcua
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
NamespaceIndex	OUT	WORD	0 .. 65535	Ausgabe des Namensraum-Index des am Parameter "NamespaceUri" angegebenen Namensraums. Die Namensraum-Indices 0 und 1 sind von der OPC Foundation reserviert und spezifiziert.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).

UA_NodeGetHandleList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NodeGetHandleList registrieren Sie NodeIDs am verbundenen UA-Server.

Der UA-Server liefert die Handles zurück. Diese werden vom Baustein am Parameter "NodeHdls" ausgegeben. Auf die Items der Node-Handles greifen Sie mit den Bausteinen "UA_ReadList" und "UA_WriteList" an deren Eingangsparameter "NodeHdls" zu.

Die NodeIDs des Ziel-Servers, die Sie am Eingangsparameter "NodeIDs" angeben, müssen Ihnen bekannt sein. Diese legen Sie in einem UDT "UANodeIDs" ab.

Die Anzahl der Ziel-Knoten am Eingangsparameter "NodeIDCount" ist immer 1, daher müssen Sie den Baustein für jede NodeID separat aufrufen.

Siehe auch

Parameter (Seite 7019)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-420 Parameter des Bausteins UA_NodeGetHandleList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeIDCount	IN	WORD	1	Anzahl der Elemente im Datenbereich von "NodeIDs". Die Anzahl muss 1 sein.
NodeIDs	IN	ARRAY of UDT	Siehe "Bedeutung"	Struktur der Parameter des Ziel-Knotens des OPC UA-Servers. Die Anzahl der NodeIDs (1) muss identisch sein mit der Anzahl am Ausgangsparameter "NodeHdls". Zum Aufbau des UDT siehe UANodeID (Seite 7032).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeHdls	OUT	ARRAY of DWORD	0 .. 4 294 967 295	Array der Handles, die der OPC UA-Server als Antwort auf die Anfrage des Client vergibt. Die Handles dienen der eindeutigen Identifikation des Items im Server. Die Handles sind solange gültig, bis sie durch Aufruf des Bausteins UA_NodeReleaseHandle-List aufgelöst, d. h. ungültig werden. Wenn eine Sitzung abgebaut wird, löst der CP alle registrierten Node-Handles dieser Sitzung auf. Der Parameter wird für den Eingangsparameter "NodeHdls" der Bausteine "UA_ReadList" und "UA_WriteList" verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält die Fehlernummern für alle Elemente des Datenbereichs von "NodeIDs". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).

UA_NodeReleaseHandleList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_NodeReleaseHandleList geben Sie am verbundenen UA-Server die Node Handles der aktuellen Sitzung frei. Die Liste wird damit gelöscht.

Der Eingangsparameter "NodeHdls" referenziert den Datenbaustein, in den die Handles vom Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" geschrieben wurden.

Die Anzahl der Handles am Eingangsparameter "NodeHdlCount" ist immer 1, daher müssen Sie den Baustein pro Handle separat aufrufen.

Siehe auch

Parameter (Seite 7021)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-421 Parameter des Bausteins UA_NodeReleaseHandleList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung / Bemerkung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im Datenbereich von "NodeHdls". Die Anzahl muss 1 sein.
NodeHdls	IN	ARRAY of DWORD	0 .. 4 294 967 295	Array der IDs der aufzulösenden Handles. Bei erfolgreicher Bausteinbearbeitung werden die Handles am Server aufgelöst und damit ungültig.
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung / Bemerkung
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehler-Codes für alle Elemente des Datenbereichs von "NodeHdls".

UA_Disconnect

Funktion

Funktion des Bausteins

Mit dem Baustein UA_Disconnect bauen Sie eine bestehende Verbindung zwischen dem CP als OPC UA-Client und einem UA-Server ab. Die bestehende Sitzung wird damit beendet.

Siehe auch

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-422 Parameter des Bausteins UA_Disconnect

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).

UA_ReadList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_ReadList lesen Sie Daten aus dem verbundenen UA-Server.

Die zu lesenden Attribute programmieren Sie in einem UDT "UANodeAdditionalInfo", der vom Eingangsparameter "NodeAddInfos" referenziert wird.

Die am Eingangsparameter "NodeHdls" benötigten Handles werden aus dem Array des Ausgangsparameters "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.

Die Anzahl der NodeIDs am Eingangsparameter "NodeIDCount" ist 1, daher müssen Sie den Baustein für jeden Ziel-Knoten separat aufrufen.

Siehe auch

Parameter (Seite 7024)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-423 Parameter des Bausteins UA_ReadList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im zu lesenden Zielbereich. Die Anzahl muss 1 sein.
NodeHdls	IN	ARRAY of DWORD	1	Array der Handles der Ziel-Knoten (Items) des OPC UA-Servers. Die Handles werden aus dem Array des Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.
NodeAddInfos	IN	ARRAY of UDT	Siehe "Bedeutung"	Spezifiziert das zu lesende Attribut der Items und den Indexbereich. Zum Aufbau des UDT siehe UANodeAdditionalInfo (Seite 7033).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Variables	IN_OUT	UDT	Siehe "Bedeutung"	Verweis auf die zu lesenden Daten. Zum Aufbau des UDT siehe UAAnyPointer (Seite 7034). Zu den unterstützten Datentypen der Prozessdaten siehe Kapitel Datentypen (Seite 7026).
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehlernummern für alle Elemente des Datenbereichs von "Variables". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).
TimeStamps	OUT	UDT	Siehe "Bedeutung"	Enthält Zeitstempel für alle Elemente des Datenbereichs von "Variables". Zum Aufbau des UDT siehe UATimeStamp (Seite 7034).

UA_WriteList

Funktion

Funktion des Bausteins

Mit dem Baustein UA_WriteList schreiben Sie Daten in den verbundenen UA-Server.

Die zu schreibenden Attribute programmieren Sie in einem UDT "UANodeAdditionalInfo", der vom Eingangsparameter "NodeAddInfos" referenziert wird.

Die am Eingangsparameter "NodeHdls" benötigten Handles werden aus dem Array des Ausgangsparameters "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.

Die Anzahl der NodeIDs am Eingangsparameter "NodeIDCount" ist 1, daher müssen Sie den Baustein für jeden Ziel-Knoten separat aufrufen.

Siehe auch

Parameter (Seite 7025)

Programmbausteine für den OPC UA-Client (Seite 7011)

Parameter

Bausteinparameter

Die folgende Tabelle erläutert die Formalparameter des Programmbausteins.

Tabelle 4-424 Parameter des Bausteins UA_WriteList

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
Execute	IN	BOOL	0, 1	Eine steigende Flanke 0 → 1 am Parameter löst die Bearbeitung des Bausteins aus.
ConnectionHdl	IN	DWORD	1 .. 5	Von UA_Connect gelieferte Verbindungskennzeichnung
NodeHdlCount	IN	WORD	1	Anzahl der Elemente im zu schreibenden Zielbereich. Die Anzahl muss 1 sein.

Parameter	Deklaration	S7-Datentyp	Wertebereich	Bedeutung
NodeHdls	IN	ARRAY of DWORD	1	Array der Handles der Ziel-Knoten (Items) des OPC UA-Servers. Die Handles werden aus dem Array des Ausgangsparameter "NodeHdls" des Bausteins "UA_NodeGetHandleList" gelesen.
NodeAddInfos	IN	ARRAY of UDT	Siehe "Bedeutung"	Spezifiziert das zu schreibende Attribut der Items und den Indexbereich. Zum Aufbau des UDT siehe UANodeAdditionalInfo (Seite 7033).
Timeout	IN	TIME	5000 .. 120000	Maximale Zeitspanne für den Aufbau der Verbindung in Millisekunden. Bei Überschreitung des Werts wird die Bausteinbearbeitung mit der Fehlernummer B0007001 abgebrochen. Bei Eingaben außerhalb des zulässigen Bereichs wird der voreingestellte Wert von 60000 (60 Sekunden) verwendet.
Variables	IN_OUT	UDT	Siehe "Bedeutung"	Verweis auf die zu schreibenden Daten. Zum Aufbau des UDT siehe UAAnyPointer (Seite 7034). Zu den unterstützten Datentypen der Prozessdaten siehe Kapitel Datentypen (Seite 7026).
Done	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Bausteinbearbeitung abgebrochen, noch nicht abgeschlossen oder noch nicht begonnen • 1: Bausteinbearbeitung ohne Fehler abgeschlossen
Busy	OUT	BOOL	0, 1	Zustandsparameter zur Bausteinbearbeitung • 0: Baustein nicht in Bearbeitung • 1: Baustein momentan in Bearbeitung
Error	OUT	BOOL	0, 1	Fehleranzeige • 0: Kein Fehler • 1: Ein Fehler ist aufgetreten. Siehe Parameter "ErrorID".
ErrorID	OUT	DWORD	Siehe "Bedeutung"	Ausgabe der Fehlernummer, wenn "Error" = 1. Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).
NodeErrorIDs	OUT	ARRAY of DWORD	Siehe "Bedeutung"	Enthält Fehlernummern für alle Elemente des Datenbereichs von "Variables". Zur Bedeutung der Nummern siehe Kapitel Fehlernummern (Seite 7036).

Datentypen

Datentypen und Speicherbereiche

Der CP 443-1 OPC UA unterstützt die nachfolgend aufgeführten Datentypen für Prozessdaten.

Die entsprechenden Symbole in der CPU sind für die jeweils aufgeführten Operandenbereiche zulässig. Die Speicherbereiche haben folgende Abkürzungen. Die Kürzel in Klammern sind die Abkürzungen in Abhängigkeit der eingestellten Mnemonik (Deutsch/Englisch).

- E = Prozessabbild der Eingänge (E/I)
- A = Prozessabbild der Ausgänge (A/Q)
- PE = Peripheriebereich Eingänge (PE/PI)
- PA = Peripheriebereich Ausgänge (PA/PQ)
- M = Merker
- DB = Datenbaustein / Systemdatenbaustein / Instanzdatenbaustein

Peripheriebereiche werden nur vom OPC UA-Server unterstützt.

Tabelle 4-425 Datentypen

Größe [bit], Typ	Bereich	Datentyp			Anmerkung	Operandenbereich der CPU
		SIMATIC S7	IEC 61131-3	OPC UA		
1	0 .. 255	BOOL	BOOL	Boolean	0 = false 1 .. 255 = true	E, A, PE, PA, M, DB
Zahlen						
8, UInt8	0 .. 255	BYTE	BYTE, USINT	Byte		E, A, PE, PA, M, DB
8, UInt8	0 .. 255	CHAR	CHAR	Byte	ASCII-Zeichen 32 .. 255	E, A, PE, PA, M, DB
16, UInt16	0 .. 65535	WORD	WORD, UINT	UInt16		E, A, PE, PA, M, DB
16, Int16	-32768 .. 32767	INT	INT	Int16		E, A, PE, PA, M, DB
32, UInt32	0 .. 4294967295	DWORD	DWORD, UDINT	UInt32		E, A, PE, PA, M, DB
32, Int32	2147483648 .. 2147483647	DINT	DINT	Int32		E, A, PE, PA, M, DB
32, Float	±1.175 495e-38 .. 3.402823e+38	REAL	REAL	Float		DB
Zeit und Datum						
16, UInt16	0 .. 999	S5TIME	TIME	UInt16	SIMATIC-Zeit Zur Codierung siehe Online-Hilfe von STEP 7.	DB
16, UInt16	D#1990-1-1 .. D#2168-12-31 (0 .. 65378)	DATE	DATE	UInt16	IEC-Datum [d]	DB
32, UInt32	TOD#0:0:0.0 .. TOD#23:59:59.999 (0 .. 86399999)	TIME_OF_DAY (TOD)	TIME_OF_DAY	UInt32	Uhrzeit [ms]	DB

Größe [bit], Typ	Bereich	Datentyp			Anmerkung	Operandenbereich der CPU
		SIMATIC S7	IEC 61131-3	OPC UA		
32, UInt32	-24D_20H_31M_23S_648MS .. T#24D_20H_31M_23S_647MS (-2147483648 .. 2147483647)	TIME	TIME	UInt32	IEC-Zeit [ms]	DB
Zusammengesetzte Datentypen						
64	1990-1-1-0:0:0.0 .. 2089-12-31-23:59:59.999	DATE_AND_TIME	DT DATE_AND_TIME	UInt32+ UInt32	Datum und Uhrzeit Zur Codierung siehe Online-Hilfe von STEP 7.	DB
254 * 8		STRING *	STRING	String	Max. 254 Byte Nutzdaten	DB
		ARRAY *			Multidimensionales Feld aus elementaren oder zusammengesetzten Datentypen	DB
		UDT *			Anwenderdefinierter Datentyp für spezifische Parametergruppen	DB

* Zur Codierung siehe unten.

Die generelle Beschreibung zur Codierung der Datentypen finden Sie in der Online-Hilfe von STEP 7.

Besonderheiten bei zusammengesetzten Datentypen

Bei zusammengesetzten Datentypen, die von der OPC UA-Funktion des CP verwendet werden, gelten besondere Codierungs-Vorschriften, die nachfolgend beschrieben sind.

STRING

Ein STRING kann max. 254 Zeichen enthalten.

Es sind ASCII-Zeichen aus dem Bereich 32 (0x20) .. 255 (0xFF) erlaubt.

Davon sind folgende Zeichen nicht erlaubt: 127 (0x7F), 129 (0x81), 141 (0x8D), 143 (0x8F), 144 (0x90), 157 (0x9D)

Besonderheiten für den OPC UA-Server

Die Zeichen werden im OPC UA-Server des CP im Format UTF-8 codiert.

Besonderheiten für den OPC UA-Client

Bei Prozessdaten mit dem Datentyp STRING, die vom UDT "UAAnyPointer" referenziert werden, wird als Datentyp "2" (BYTE) angegeben.

Wählen Sie als Wiederholfaktor im UAAnyPointer immer den maximalen Wert von 256 (0x100): 254 für die Nutzdaten + 2 für den Header.

ARRAY

Besonderheiten für den OPC UA-Client

Die Client-Programmbausteine unterstützen nur die Übertragung von eindimensionalen Arrays aus dem Speicherbereich "DB".

Die Übertragung des Datentyps "ARRAY of BOOL" wird nicht von den Client-Programmbausteinen unterstützt.

Bei einem ARRAY of STRING muss bei den Grenzen der Dimension jedes String-Elements "n" mal die maximale Länge (256) angegeben werden, wobei "n" die Anzahl der Strings im Array ist.

Besonderheiten für den OPC UA-Server

Der Datentyp "ARRAY of BOOL" ist in allen Fällen lesbar.

Der Datentyp "ARRAY of BOOL" ist nur dann schreibbar, wenn die Länge ein Vielfaches von 8 ist.

UDT

Den Aufbau der UDTs, die von den Programmbausteinen für die OPC UA-Client-Funktion benötigt werden, finden Sie bei der Beschreibung des jeweiligen Bausteins, siehe beispielsweise AUTOHOTSPOT.

Parametertypen

Die Parametertypen "TIMER" und "COUNTER" werden nur für den OPC UA-Server des CP unterstützt.

UDTs für den CP 443-1 OPC UA

UASessionConnectInfo

UASessionConnectInfo

Entnehmen Sie der folgenden Tabelle die Bedeutung der Verbindungsinformationen für den Parameter "SessionConnectInfo" des Bausteins UA_Connect.

Tabelle 4-426 Parameter des UDT UASessionConnectInfo

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SessionName	STRING	0 .. 254 Zeichen	Name der Sitzung Wenn der String leer bleibt, wird folgendes als Sitzungsname systemseitig eingetragen: <Connection>+<Dezimalwert von "ConnectionHdl">
ApplicationName	STRING	0 .. 254 Zeichen	Der Parameter wird durch STEP 7 mit dem in HW Konfig projektierten Applikationsnamen des CP (OPC UA-Client) belegt.
SecurityMsgMode	WORD	0 .. 3	Security-Verfahren • 0 = Bestmögliches Verfahren • 1 = Kein Security-Verfahren • 2 = Authentifizieren • 3 = Authentifizieren und verschlüsseln
SecurityPolicy	WORD	0 .. 4	Security-Profil • 0 = Bestmögliches Security-Profil • 1 = Kein Security-Profil • 2 = Basic128Rsa15 • 3 = Basic256 • 4 = Basic256Sha256
CertificateStore	STRING	0 .. 254 Zeichen	Zertifikatspeicher des CP (OPC UA-Client) Der Parameter wird durch STEP 7 eingesetzt.
ClientCertificate-Name	STRING	0 .. 254 Zeichen	Name des Client-Zertifikats Der Parameter wird durch STEP 7 eingesetzt.
ServerUri	STRING	0 .. 254 Zeichen	Server-Adresse (URI), die auch im Server-Zertifikat hinterlegt ist. Der Parameter darf nur leer gelassen werden für Sitzungen ohne Sercurity-Verfahren. Beispiel: urn:<ApplicationName>:GUID

Parameter	S7-Datentyp	Wertebereich	Bedeutung
CheckServerCertificate	BOOL	0, 1	Abgleich (Prüfung) des vom Server gesendeten Zertifikats mit dem Server-Zertifikat, das im Zertifikatspeicher des Client-CP gespeichert ist: • 0 = Keine Prüfung • 1 = Prüfung **Hinweis:** Die Einstellungen dieses Parameters werden von den Security-Einstellungen in der Projektierung überschrieben. Wenn Sie die Prüfung aktivieren (1), aber die Security-Funktionen in der Projektierung des CP deaktiviert sind, dann wird der Verbindungsaufbau mit einer Fehlermeldung abgebrochen. Empfehlung: Lassen Sie den Parameter in der Voreinstellung (0), da die Security-Funktionen über die Projektierung des CP aktiviert werden.
TransportProfile	WORD	1	1 = UATP_UATcp Nur dieses Transportprofil gemäß der PLCopen-Spezifikation wird unterstützt.
UserIdentityToken	UDT	Siehe "Bedeutung"	Daten der Benutzerauthentifizierung, siehe UAUserIdentityToken (Seite 7032).
VendorSpecificParameter	WORD	Siehe "Bedeutung"	Eingabe der logischen Adresse des CP 443-1 OPC UA. Sie finden diese im STEP 7-Eigenschaftendialog des CP als Eingangsadresse im Register "Adressen".
SessionTimeout	TIME	30000 .. 86 400 000	Maximale Zeitspanne, über die eine Sitzung ohne Datenverkehr aufrecht erhalten wird (Millisekunden). Bei Überschreitung des Werts wird die Sitzung (Verbindung) abgebrochen. In diesem Fall müssen Sie die Verbindung über den Aufruf von UA_Connect neu aufbauen. Bei Programmierung von Werten außerhalb des zulässigen Wertebereichs (30 Sekunden ... 1 Tag) wird der Wert 1200000 (20 Minuten) verwendet.
MonitorConnection	TIME	5000 .. 86 400 000	Verbindungsüberwachungszeit (Millisekunden) Zeitspanne ohne Datenverkehr, nach welcher der Client durch Versenden eines Telegramms die Verbindung zum Server prüft. Voreinstellung: 15000 (15 Sekunden)
LocaleIDs	ARRAY[1..5] of String[6]	Beispiele: en-US, de-DE, zh-CHS ...	Optionale Sprach- und Regionenkennung gemäß RFC 3066. 0 = keine oder unbekannte LocaleID.

UAUserIdentityToken

UAUserIdentityToken

Entnehmen Sie der folgenden Tabelle die Bedeutung der Benutzerauthentifizierungsdaten, welche den Parameter "UserIdentityToken" des UDT "UASessionConnectInfo" versorgen.

Tabelle 4-427 Parameter des UDT UAUserIdentityToken

Parameter	S7-Datentyp	Wertebereich	Bedeutung
UserIdentity TokenType	WORD	0, 1	0 = Keine Authentifizierung 1 = Authentifizierung Wenn der Kommunikationspartner (Server) eine Authentifizierung über Benutzername und Passwort verlangt, dann setzen Sie diesen Parameter auf 1 und die beiden nachfolgenden entsprechend den Anforderungen des Servers.
TokenParam1	STRING	1 .. 254 Zeichen	Benutzername
TokenParam2	STRING	1 .. 254 Zeichen	Passwort

UANodeID

UANodeID

Entnehmen Sie der folgenden Tabelle die Bedeutung der Parameter von UANodeID zur Identifizierung des Ziel-Knotens im OPC UA-Server. UANodeID versorgt den Parameter "NodeIDs" des Bausteins UA_NodeGetHandleList".

Tabelle 4-428 Parameter des UDT UANodeID

Parameter	S7-Datentyp	Wertebereich	Bedeutung
NamespaceIndex	WORD	0 .. 65535	Index des Namensraums des Servers
Identifier	STRING	Max. 254	Spezifiziert die NodeID im Namensraum-Index.
IdentifierType	WORD	1 .. 2	Legt das Format und den Geltungsbereich (i. d. R. der Server) der NodeID fest. Unterstützte Typen: • 1: UAIdentifierType_String String-Identifier unterscheiden Groß-/Kleinschreibung. • 2: UAIdentifierType_Numeric Numerischer Identifier

UANodeAdditionalInfo

UANodeAdditionalInfo

Spezifiziert das Attribut des Items und den Indexbereich für den Bausteinparameter "NodeAddInfos".

Tabelle 4-429 Parameter des UDT UANodeAdditionalInfo

Parameter	S7-Datentyp	Wertebereich	Bedeutung
AttributeID	WORD	13	Attribut der Items. Nur Attribut 13 (UAAI_Value) für den Wert des Items wird unterstützt.
IndexRangeCount	WORD	0 .. 1	Anzahl der Indexbereiche Der Wert ist bei elementaren Datentypen unerheblich. Für Items mit Datentyp ARRAY gilt: • 0: Ein einziger Index. Das gesamte Array wird gelesen/geschrieben. • 1: Ein durch "IndexRange" definierter Teilbereich eines Arrays.
IndexRange	UDT	Siehe "Bedeutung"	Bereich des Arrays für den Parameter IndexRangeCount. Zum Aufbau des UDT siehe UAIndexRange (Seite 7033).

UAIndexRange

UAIndexRange

Legt den Start- und End-Index für den Parameter "IndexRange" des UDT "UANodeAdditionalInfo" fest.

Tabelle 4-430 Parameter des UDT UAIndexRange

Parameter	Datentyp	Wertebereich	Bedeutung
StartIndex	WORD	0 .. 65535	Index, ab dem gelesen wird.
EndIndex	WORD	0 .. 65535	Index, bis zu dem gelesen wird.

Regeln:

Wenn IndexRangeCount = 1 verwendet wird, gelten folgende Regeln für die Programmierung des Indexbereichs:

- StartinIndex und EndIndex müssen zugewiesen werden.
- Für den Zugriff auf mehr als ein Element muss der StartIndex kleiner als der EndIndex sein. Nichtbeachtung führt zu einem Fehler mit der Nummer 80360000.

- Wenn Sie ein einzelnes Element des Arrays lesen wollen, dann geben Sie für StartIndex und EndIndex den gleichen Wert an.
- Um auf alle Elemente des Arrays zugreifen zu können, müssen StartIndex und EndIndex abhängig von der Gesamtzahl der Elemente des Arrays zugewiesen werden.
  Angaben außerhalb der Array-Größe führen zu einem Fehler mit der Nummer 80370000.

Beispiel für die Zuweisung von "StartIndex" und "EndIndex" bei Arrays:

Aus einem Array mit der Größe von 10 Elementen (Nr. 1 .. 10) sollen die Elemente 3 bis 5 gelesen werden. Die beiden Indices werden wie folgt programmiert:

- StartIndex = 2 (Element Nr. 3)
- EndIndex = 4 (Element Nr. 5)

UATimeStamp

UATimeStamp

Der UDT enthält die Zeitstempel der Elemente des Datenbereichs in Millisekunden. UATimeStamp versorgt den Parameter "TimeStamps" des Bausteins "UA_ReadList".

Tabelle 4-431 Parameter des UDT UATimeStamp

Parameter	S7-Datentyp	Wertebereich	Bedeutung
TimestampL	WORD	Siehe unten	Niederwertiges Wort (Bit 0..15)
TimestampH	WORD	Siehe unten	Höherwertiges Wort (Bit 16..31)

Wertebereich des kompletten Doppelworts:

TOD#0:0:0.0 .. TOD#23:59:59.999 (0 .. 86 399 999 Millisekunden)

UAAnyPointer

UAAnyPointer

Der UDT "UAAnyPointer" wird vom Parameter "Variables" des Programmbausteins referenziert.

Der UDT verweist auf den Speicherbereich der Prozessdaten in der CPU. Prozessdaten in den Operandenbereichen "Eingänge" oder "Ausgänge" müssen hierfür in einen Datenbaustein kopiert werden.

Legen Sie den UDT in einem Datenbaustein ab.

Tabelle 4-432 Parameter des UDT UAAnyPointer

Parameter	S7-Datentyp	Wertebereich	Bedeutung
SyntaxID	WORD	10	Der Wert für die Syntax-ID ist immer 10.
DataType	WORD	Siehe "Bedeutung"	Datentyp der Ziel-Knoten. Zum Wertebereich siehe unten, Tabelle "Codierung des Datentyps".
RepetitionFactor	WORD	0 .. 65535	Wiederholfaktor Für weitere Informationen siehe STEP 7-Online-Hilfe.
DB_Number	WORD	0 .. 65535	Nummer des Datenbausteins (DB) Geben Sie alternativ die Nummer eines DB oder einen Speicherbereich an. Wenn Sie einen DB angeben, dann geben Sie unter "MemArea" eine Null ein.
MemArea	WORD	Siehe "Bedeutung"	Speicherbereich Geben Sie alternativ die Nummer eines DB oder einen Speicherbereich an. Wenn Sie einen Speicherbereich angeben, dann geben Sie unter "DB_Number" eine Null ein. Zum Wertebereich des Speicherbereichs siehe unten, Tabelle "Codierung des Speicherbereichs".
ByteOffset	WORD	Abhängig vom Speicherbereich, siehe "Bedeutung"	Byte-Offset im angegebenen Speicherbereich, ab dem auf die Daten zugegriffen wird.
BitOffset	WORD	0 .. 7	Bit-Offset im angegebenen Speicherbereich

Datentyp

Die folgende Tabelle erläutert die Codierung des Datentyps im Parameter "DataType" des UDT UAAnyPointer.

Tabelle 4-433 Codierung des Datentyps

Hexadezimal-Code	S7-Datentyp	Beschreibung
b#16#01	BOOL	Bit
b#16#02	BYTE	Byte (8 Bits)
b#16#03	CHAR	Zeichen (8 Bits)
b#16#04	WORD	Wort (16 Bits)
b#16#05	INT	Ganzzahl (16 Bits)
b#16#06	DWORD	Wort (32 Bits)
b#16#07	DINT	Ganzzahl (32 Bits)
b#16#08	REAL	Gleitpunktzahl (32 Bits)
b#16#09	DATE	Datum
b#16#0A	TIME_OF_DAY (TOD)	Uhrzeit
b#16#0B	TIME	Zeit

Hexadezimal-Code	S7-Datentyp	Beschreibung
b#16#0C	S5TIME	Datentyp S5TIME
b#16#0E	DATE_AND_TIME (DT)	Datum und Zeit (64 Bits)
b#16#02 *	STRING	Zeichenkette

* Der String verwendet den untergeordneten Datentyp BYTE. Beachten Sie die Besonderheiten im Kapitel Datentypen (Seite 7026).

Speicherbereich

Die folgende Tabelle erläutert die Codierung des Speicherbereichs im Parameter "MemArea" des UDT UAAnyPointer.

Tabelle 4-434 Codierung des Speicherbereichs

Hexadezimal-Code	Bereich	Beschreibung
b#16#80	P	Speicherbereich der Peripherie
b#16#81	E	Speicherbereich der Eingänge
b#16#82	A	Speicherbereich der Ausgänge
b#16#83	M	Speicherbereich der Merker
b#16#84	DB	Datenbaustein

Fehlernummern

Auswertung der Ausgangsparameter

Die Ausgangsparameter Done (BOOL), Error (BOOL) und ErrorID (DWORD) der OPC UA-Client-Programmbausteine werden bei jedem Bausteinaufruf aktualisiert. Werten Sie die Zustände der Ausgangsparameter im Anwenderprogramm aus.

Beispiel:

Nach dem Aufruf liefert der Programmbaustein UA_Connect folgende Werte an den drei Ausgangsparametern:

- Done = 1
- Error = 0
- ErrorID = 00000000

Bedeutung: Der Auftrag wurde erfolgreich abgeschlossen.

Bedeutung der Fehlernummern (hexadezimal) von ErrorID

Tabelle 4-435 Bedeutung von ErrorID in Zusammenhang mit Done und Error

Done	Error	ErrorID (hex)	Name	Bedeutung
1	0	00000000		Die Bausteinbearbeitung wurde erfolgreich durchgeführt.
CP-spezifische Fehlernummern				

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	B0006001		Fehlende positive Flanke zum Aufruf des Programmbausteins
0	1	B0007001		Zeitüberschreitung
0	1	B0008004		Ungültiger NodeIDCount
0	1	B0008086		Ungültiges Node-Handle
0	1	B0008087		Die maximale Anzahl an Verbindungen (5) ist erreicht.
0	1	B0008090		Falsche logische Adresse. Verbindungsaufbau abgebrochen.
0	1	B0008093		Die logische Adresse kann nicht verarbeitet werden. Eine unzulässige Baugruppe wird verwendet.
0	1	B00080A1		Das Node-Handle wurde nicht für diese Verbindung eingerichtet.
0	1	B00080C3		Die Verbindungsressourcen der CPU sind vorübergehend ausgeschöpft.
0	1	B0009085		Der maximale Speicherplatz für Items (64000 Byte) des OPC UA-Client ist überschritten.
0	1	B000800A		Der OPC UA-Client wurde in der Projektierung nicht aktiviert.
0	1	B0008723		Der Lese-/Schreibauftrag referenziert einen nicht existierenden Datenbereich in einem DB.
0	1	B000F001		In einem Zyklus wurden zwei Funktionsbausteine aufgerufen.
0	1	B000F002		OPC UA-Sitzung noch nicht aufgebaut. Rufen Sie den Baustein UA_Connect erneut auf.
0	1	B000F003		Kommunikationsfehler
0	1	B000F004		Ungültiger Wert für den Parameter "SyntaxID" im UDT "UAAnyPointer"
0	1	B000F005		Ungültiger Datentyp in UAAnyPointer
0	1	B000F006		Ungültiger Wert für den Parameter "RepetitionFactor" im UDT "UAAnyPointer"
0	1	B000F007		Ungültiger Operandenbereich von UAAnyPointer
0	1	B000F010		Ungültiger Operandenbereich in einem Lese- oder Schreibauftrag: Arrays von Eingängen, Arrays von Ausgängen, Arrays von Merkern, Arrays von Lokaldaten und Arrays von Lokaldaten des Vorgänger-Bausteins sind nicht erlaubt.
0	1	B000F011		Der Bit-Offset muss Null sein.
0	1	B0010001		Falsches Datenpaket in UA_Connect
0	1	B0010002		Ungültige URL eines Server-Endpunkts
0	1	B0010003		Ungültiger Wert für Timeout
0	1	B0010004		Ungültiger Wert für SessionTimeout
0	1	B0010005		Ungültiger Wert für MonitorConnection
0	1	B0010006		Ungültiges Transportprofil
0	1	B0010007		Ungültige Server-URI
0	1	B0010008		Ungültiges Security-Verfahren
0	1	B0010009		Ungültiges Security-Profil
0	1	B001000A		Ungültiger Wert für UserIdentityToken
0	1	B001000B		Ungültiger Applikationsname
0	1	B0010101		Ungültiges Format der Server-Endpunkt-URL. Es muss mit "opc.tcp://" beginnen.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	B0010102		Der Pfad der Server-Endpunkt-URL wird nicht unterstützt.
0	1	B0010103		Der Client darf sich nicht mit dem Server des eigenen CP verbinden.
0	1	B0010104		Ungültiger Port
0	1	B0010105		Ungültige Adresse. DNS wird nicht unterstützt.
0	1	B0020001		Falsches Datenpaket in UA_NameSpaceGetIndex
0	1	B0030001		Falsches Datenpaket in UA_NodeGetHandleList
0	1	B0030002		Ungültiger NodeID-Identifier
0	1	B0030003		Überschreitung der max. Anzahl an Items (64000)
0	1	B0030004		Nicht unterstütztes Attribut "ValueRank" einer Variable. Die ValueRank-Werte 1 (eindimensionales Array) und -1 (skalar) werden unterstützt.
0	1	B0040001		Falsches Datenpaket in UA_NodeReleaseHandleList
0	1	B0050001		Falsches Datenpaket in UA_Disconnect
0	1	B0050002		UA_Disconnect kann nicht aufgerufen werden, da keine Verbindung besteht.
0	1	B0060001		Falsches Datenpaket in UA_ReadList
0	1	B0060002		Die Länge der zu lesenden Daten überschreitet das Maximum von 64000.
0	1	B0060003		Ungültiger Bereich von IndexRangeCount
0	1	B0060004		Ungültiger IndexRange. StartIndex darf nicht größer sein als EndIndex und EndIndex darf nicht größer sein als die Array-Länge.
0	1	B0060005		Ungültige AttributeID
0	1	B0060006		Der gelesene Datenbereich ist zu klein. Korrigieren Sie die Angaben im UDT "UAIndexRange".
0	1	B0060007		Ungültige String-Länge. Der Wert (Nutzdaten) muss 254 sein.
0	1	B0060008		Ungültiger Wert des Knotens. Der Wert kann nicht in S7-Variable eingelesen werden.
0	1	B0060010		Ungültiger Operandenbereich in einem Lese- oder Schreibauftrag: Arrays von Eingängen, Arrays von Ausgängen, Arrays von Merkern, Arrays von Lokaldaten und Arrays von Lokaldaten des Vorgänger-Bausteins sind nicht erlaubt.
0	1	B0070001		Falsches Datenpaket in UA_WriteList
0	1	B0070002		Die Länge der zu schreibenden Daten überschreitet das Maximum.
0	1	B0070003		Ungültiger Bereich von IndexRangeCount
0	1	B0070004		Ungültiger IndexRange. StartIndex darf nicht größer sein als EndIndex und EndIndex darf nicht größer sein als die Array-Länge.
0	1	B0070005		Ungültige AttributeID
0	1	B0070006		Der Datenbereich ist außerhalb des zugelassenen Bereichs.
0	1	B0070009		Ungültiges Zeichen im String
Fehlernummern nach IEC 61131-3 für OPC UA				
0	1	002D0000	GoodSubscriptionTransferred	The subscription was transferred to another session.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	002E0000	GoodCompletes Asynchronously	The processing will complete asynchronously.
0	1	002F0000	GoodOverload	Sampling has slowed down due to resource limitations.
0	1	00300000	GoodClamped	The value written was accepted but was clamped.
0	1	00960000	GoodLocalOverride	The value has been overridden.
0	1	00A20000	GoodEntryInserted	The data or event was successfully inserted into the historical database.
0	1	00A30000	GoodEntryReplaced	The data or event field was successfully replaced in the historical database.
0	1	00A50000	GoodNoData	No data exists for the requested time range or event filter.
0	1	00A60000	GoodMoreData	The data or event field was successfully replaced in the historical database.
0	1	00A70000	GoodCommunicationEvent	The communication layer has raised an event.
0	1	00A80000	GoodShutdownEvent	The system is shutting down.
0	1	00A90000	GoodCallAgain	The operation is not finished and needs to be called again.
0	1	00AA0000	GoodNonCriticalTimeout	A non-critical timeout occurred.
0	1	00BA0000	GoodResultsMayBe Incomplete	The server should have followed a reference to a node in a remote server but did not. The result set may be incomplete.
0	1	406C0000	UncertainReferenceOutOf Server	One of the references to follow in the relative path references to a node in the address space in another server.
0	1	408F0000	UncertainNoCommunication LastUsableValue	Communication to the data source has failed. The variable value is the last value that had a good quality.
0	1	40900000	UncertainLastUsableValue	Whatever was updating this value has stopped doing so.
0	1	40910000	UncertainSubstituteValue	The value is an operational value that was manually overwritten.
0	1	40920000	UncertainInitialValue	The value is an initial value for a variable that normally receives its value from another variable.
0	1	40930000	UncertainSensorNotAccurate	The value is at one of the sensor limits.
0	1	40940000	UncertainEngineeringUnits Exceeded	The value is outside of the range of values defined for this parameter.
0	1	40950000	UncertainSubNormal	The value is derived from multiple sources and has less than the required number of Good sources.
0	1	40A40000	UncertainDataSubNormal	The value is derived from multiple values and has less than the required number of Good values.
0	1	40BC0000	UncertainReferenceNot Deleted	The server was not able to delete all target references.
0	1	40C00000	UncertainNotAllNodes Available	The list of references may not be complete because the underlying system is not available.
0	1	80010000	BadUnexpectedError	An unexpected error occurred
0	1	80020000	BadInternalError	An internal error occurred as a result of a programming or configuration error.
0	1	80030000	BadOutOfMemory	Not enough memory to complete the operation.
0	1	80040000	BadResourceUnavailable	An operating system resource is not available
0	1	80050000	BadCommunicationError	A low level communication error occurred.
0	1	80060000	BadEncodingError	Encoding halted because of invalid data in the objects being serialized.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80070000	BadDecodingError	Decoding halted because of invalid data in the stream.
0	1	80080000	BadEncodingLimitsExceeded	The message encoding/decoding limits imposed by the stack have been exceeded.
0	1	80B80000	BadRequestTooLarge	The resources (memory) of the CPU are temporarily occupied
0	1	80B90000	BadResponseTooLarge	The response message size exceeds limits set by the client.
0	1	80B90000	BadUnknownResponse	An unrecognized response was received from the server.
0	1	800A0000	BadTimeout	The operation timed out.
0	1	800B0000	BadServiceUnsupported	The server does not support the requested service.
0	1	800C0000	BadShutdown	The operation was cancelled because the application is shutting down.
0	1	800D0000	BadServerNotConnected	The operation could not complete because the client is not connected to the server.
0	1	800E0000	BadServerHalted	The server has stopped and cannot process any requests.
0	1	800F0000	BadNothingToDo	There was nothing to do because the client passed a list of operations with no elements.
0	1	80100000	BadTooManyOperations	The request could not be processed because it specified too many operations.
0	1	80110000	BadDataTypeIdUnknown	The extension object cannot be (de)serialized because the data type id is not recognized.
0	1	80120000	BadCertificateInvalid	The certificate provided as a parameter is not valid.
0	1	80130000	BadSecurityChecksFailed	The certificate provided as a parameter is not valid.
0	1	80140000	BadCertificateTimeInvalid	The Certificate has expired or is not yet valid.
0	1	80150000	BadCertificateIssuer TimeInvalid	An Issuer Certificate has expired or is not yet valid.
0	1	80160000	BadCertificateHost NameInvalid	The HostName used to connect to a Server does not match a HostName in the Certificate.
0	1	80170000	BadCertificateUriInvalid	The URI specified in the Application Description does not match the URI in the Certificate.
0	1	80180000	BadCertificateUseNotAllowed	The Certificate may not be used for the requested operation.
0	1	80190000	BadCertificateIssuerUse NotAllowed	The Issuer Certificate may not be used for the requested operation.
0	1	801A0000	BadCertificateUntrusted	The Certificate is not trusted.
0	1	801B0000	BadCertificateRevocation Unknown	It was not possible to determine if the Certificate has been revoked.
0	1	801C0000	BadCertificateIssuer RevocationUnknown	It was not possible to determine if the Issuer Certificate has been revoked.
0	1	801D0000	BadCertificateRevoked	The Certificate has been revoked.
0	1	801E0000	BadCertificateIssuerRevoked	The Issuer Certificate has been revoked.
0	1	801F0000	BadUserAccessDenied	User does not have permission to perform the requested operation.
0	1	80200000	BadIdentityTokenInvalid	The user identity token is not valid.
0	1	80210000	BadIdentityTokenRejected	The user identity token is valid but the server has rejected it.
0	1	80220000	BadSecureChannelIdInvalid	The specified secure channel is no longer valid.
0	1	80230000	BadInvalidTimestamp	The timestamp is outside the range allowed by the server.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80240000	BadNonceInvalid	The nonce does appear to be not a random value or it is not the correct length.
0	1	80250000	BadSessionIdInvalid	The session id is not valid.
0	1	80260000	BadSessionClosed	The session was closed by the client.
0	1	80270000	BadSessionNotActivated	The session cannot be used because ActivateSession has not been called.
0	1	80280000	BadSubscriptionIdInvalid	The subscription id is not valid.
0	1	802A0000	BadRequestHeaderInvalid	The header for the request is missing or invalid.
0	1	802B0000	BadTimestampsTo ReturnInvalid	The timestamps to return parameter is invalid.
0	1	802C0000	BadRequestCancelled ByClient	The request was cancelled by the client.
0	1	80310000	BadNoCommunication	Communication with the data source is defined, but not established, and there is no last known value available.
0	1	80320000	BadWaitingForInitialData	Waiting for the server to obtain values from the underlying data source.
0	1	80330000	BadNodeIdInvalid	The syntax of the node id is not valid.
0	1	80340000	BadNodeIdUnknown	The node id refers to a node that does not exist in the server address space.
0	1	80350000	BadAttributeIdInvalid	The attribute is not supported for the specified Node.
0	1	80360000	BadIndexRangeInvalid	The syntax of the index range parameter is invalid.
0	1	80370000	BadIndexRangeNoData	No data exists within the range of indexes specified.
0	1	80380000	BadDataEncodingInvalid	The data encoding is invalid.
0	1	80390000	BadDataEncoding Unsupported	The server does not support the requested data encoding for the node.
0	1	803A0000	BadNotReadable	The access level does not allow reading or subscribing to the Node.
0	1	803B0000	BadNotWritable	The access level does not allow writing to the Node.
0	1	803C0000	BadOutOfRange	The value was out of range.
0	1	803D0000	BadNotSupported	The requested operation is not supported.
0	1	803E0000	BadNotFound	A requested item was not found or a search operation ended without success.
0	1	803F0000	BadObjectDeleted	The object cannot be used because it has been deleted.
0	1	80400000	BadNotImplemented	Requested operation is not implemented.
0	1	80410000	BadMonitoringModeInvalid	The monitoring mode is invalid.
0	1	80420000	BadMonitoredItemIdInvalid	The monitoring item id does not refer to a valid monitored item.
0	1	80430000	BadMonitoredItem FilterInvalid	The monitored item filter parameter is not valid.
0	1	80440000	BadMonitoredItem FilterUnsupported	The server does not support the requested monitored item filter.
0	1	80450000	BadFilterNotAllowed	A monitoring filter cannot be used in combination with the attribute specified.
0	1	80460000	BadStructureMissing	A mandatory structured parameter was missing or null.
0	1	80470000	BadEventFilterInvalid	The event filter is not valid.
0	1	80480000	BadContentFilterInvalid	The content filter is not valid.
0	1	80490000	BadFilterOperandInvalid	The operand used in a content filter is not valid.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	804A0000	BadContinuation PointInvalid	The continuation point provide is longer valid.
0	1	804B0000	BadNoContinuationPoints	The operation could not be processed because all continuation points have been allocated.
0	1	804C0000	BadReferenceTypeIdInvalid	The operation could not be processed because all continuation points have been allocated.
0	1	804D0000	BadBrowseDirectionInvalid	The browse direction is not valid.
0	1	804E0000	BadNodeNotInView	The node is not part of the view.
0	1	804F0000	BadServerUriInvalid	The ServerUri is not a valid URI.
0	1	80500000	BadServerNameMissing	No ServerName was specified
0	1	80510000	BadDiscoveryUrlMissing	No DiscoveryUrl was specified.
0	1	80520000	BadSempahoreFileMissing	The semaphore file specified by the client is not valid.
0	1	80530000	BadRequestTypeInvalid	The security token request type is not valid.
0	1	80540000	BadSecurityModeRejected	The security mode does not meet the requirements set by the Server.
0	1	80550000	BadSecurityPolicyRejected	The security policy does not meet the requirements set by the Server.
0	1	80560000	BadTooManySessions	The server has reached its maximum number of sessions.
0	1	80570000	BadUserSignatureInvalid	The user token signature is missing or invalid.
0	1	80580000	BadApplicationSignature Invalid	The signature generated with the client certificate is missing or invalid.
0	1	80590000	BadNoValidCertificates	The client did not provide at least one software certificate that is valid and meets the profile requirements for the server.
0	1	805A0000	BadRequestCancelled ByRequest	The request was cancelled by the client with the Cancel service.
0	1	805B0000	BadParentNodeIdInvalid	The parent node id does not to refer to a valid node.
0	1	805C0000	BadReferenceNotAllowed	The reference could not be created because it violates constraints imposed by the data model.
0	1	805D0000	BadNodeIdRejected	The requested node id was reject because it was either invalid or server does not allow node ids to be specified by the client.
0	1	805F0000	BadNodeClassInvalid	The node class is not valid.
0	1	80600000	BadBrowseNameInvalid	The browse name is invalid.
0	1	80610000	BadBrowseNameDuplicated	The browse name is not unique among nodes that share the same relationship with the parent.
0	1	80620000	BadNodeAttributesInvalid	The node attributes are not valid for the node class.
0	1	80630000	BadTypeDefinitionInvalid	The type definition node id does not reference an appropriate type node.
0	1	80640000	BadSourceNodeIdInvalid	The source node id does not reference a valid node.
0	1	80650000	BadTargetNodeIdInvalid	The target node id does not reference a valid node.
0	1	80660000	BadDuplicateReference NotAllowed	The reference type between the nodes is already defined.
0	1	80670000	BadInvalidSelfReference	The server does not allow this type of selfreference on this node.
0	1	80680000	BadReferenceLocalOnly	The reference type is not valid for a reference to a remote server.
0	1	80690000	BadNoDeleteRights	The server will not allow the node to be deleted.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	806A0000	BadServerIndexInvalid	The server index is not valid.
0	1	806B0000	BadViewIdUnknown	The view id does not refer to a valid view node.
0	1	806D0000	BadTooManyMatches	The requested operation has too many matches to return.
0	1	806E0000	BadQueryTooComplex	The requested operation requires too many resources in the server.
0	1	806F0000	BadNoMatch	The requested operation has no match to return.
0	1	80700000	BadMaxAgeInvalid	The max age parameter is invalid.
0	1	80710000	BadHistoryOperationInvalid	The history details parameter is not valid.
0	1	80720000	BadHistoryOperation Unsupported	The server does not support the requested operation.
0	1	80730000	BadWriteNotSupported	The server not does support writing the combination of value, status and timestamps provided.
0	1	80740000	BadTypeMismatch	The value supplied for the attribute is not of the same type as the attribute's value.
0	1	80750000	BadMethodInvalid	The method id does not refer to a method for the specified object.
0	1	80760000	BadArgumentsMissing	The client did not specify all of the input arguments for the method.
0	1	80770000	BadTooManySubscriptions	The server has reached its maximum number of subscriptions.
0	1	80780000	BadTooManyPublish Requests	The server has reached the maximum number of queued publish requests.
0	1	80790000	BadNoSubscription	There is no subscription available for this session.
0	1	807A0000	BadSequenceNumber Unknown	The sequence number is unknown to the server.
0	1	807B0000	BadMessageNotAvailable	The requested notification message is no longer available.
0	1	807C0000	BadInsufficientClientProfile	The Client of the current Session does not support one or more Profiles that are necessary for the Subscription.
0	1	80BF0000	BadStateNotActive	The sub-state machine is not currently active.
0	1	807D0000	BadTcpServerTooBusy	The server cannot process the request because it is too busy.
0	1	807E0000	BadTcpMessageTypeInvalid	The type of the message specified in the header invalid.
0	1	807F0000	BadTcpSecureChannel Unknown	The SecureChannelId and/or TokenId are not currently in use.
0	1	80800000	BadTcpMessageTooLarge	The size of the message specified in the header is too large.
0	1	80810000	BadTcpNotEnough Resources	There are not enough resources to process the request.
0	1	80820000	BadTcpInternalError	An internal error occurred.
0	1	80830000	BadTcpEndpointUrlInvalid	The Server does not recognize the QueryString specified.
0	1	80840000	BadRequestInterrupted	The request could not be sent because of a network interruption.
0	1	80850000	BadRequestTimeout	Timeout occurred while processing the request.
0	1	80860000	BadSecureChannelClosed	The secure channel has been closed.
0	1	80870000	BadSecureChannelToken Unknown	The token has expired or is not recognized.
0	1	80880000	BadSequenceNumberInvalid	The sequence number is not valid.
0	1	80890000	BadConfigurationError	There is a problem with the configuration that affects the usefulness of the value.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	808A0000	BadNotConnected	The variable should receive its value from another variable, but has never been configured to do so.
0	1	808B0000	BadDeviceFailure	There has been a failure in the device/data source that generates the value that has affected the value.
0	1	808C0000	BadSensorFailure	There has been a failure in the sensor from which the value is derived by the device/data source.
0	1	808D0000	BadOutOfService	The source of the data is not operational.
0	1	808E0000	BadDeadbandFilterInvalid	The dead band filter is not valid.
0	1	80970000	BadRefreshInProgress	This Condition refresh failed, a Condition refresh operation is already in progress.
0	1	80980000	BadConditionAlreadyDisabled	This condition has already been disabled.
0	1	80990000	BadConditionDisabled	Property not available, this condition is disabled.
0	1	809A0000	BadEventIdUnknown	The specified event id is not recognized.
0	1	809B0000	BadNoData	No data exists for the requested time range or event filter.
0	1	809D0000	BadDataLost	Data is missing due to collection started/stopped/lost.
0	1	809E0000	BadDataUnavailable	Expected data is unavailable for the requested time range due to an unmounted volume an off-line archive or tape or similar reason for temporary unavailability.
0	1	809F0000	BadEntryExists	The data or event was not successfully inserted because a matching entry exists.
0	1	80A00000	BadNoEntryExists	The data or event was not successfully updated because no matching entry exists.
0	1	80A10000	BadTimestampNotSupported	The client requested history using a timestamp format the server does not support (i. e. requested ServerTimestamp when server only supports SourceTimestamp).
0	1	80AB0000	BadInvalidArgument	One or more arguments are invalid.
0	1	80AC0000	BadConnectionRejected	Could not establish a network connection to remote server.
0	1	80AD0000	BadDisconnect	The server has disconnected from the client.
0	1	80AE0000	BadConnectionClosed	The network connection has been closed.
0	1	80AF0000	BadInvalidState	The operation cannot be completed because the object is closed uninitialized or in some other invalid state.
0	1	80B00000	BadEndOfStream	Cannot move beyond end of the stream.
0	1	80B10000	BadNoDataAvailable	No data is currently available for reading from a non-blocking stream.
0	1	80B20000	BadWaitingForResponse	The asynchronous operation is waiting for a response.
0	1	80B30000	BadOperationAbandoned	The asynchronous operation was abandoned by the caller.
0	1	80B40000	BadExpectedStreamToBlock	The stream did not return all data requested (possibly because it is a non-blocking stream).
0	1	80B50000	BadWouldBlock	Non-blocking behavior is required and the operation would block.
0	1	80B60000	BadSyntaxError	A value had an invalid syntax.
0	1	80B70000	BadMaxConnections Reached	The operation could not be finished because all available connections are in use.
0	1	80BB0000	BadEventNot Acknowledgeable	The event cannot be acknowledged.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	80BD0000	BadInvalidTimestamp Argument	The defined timestamp to return was invalid.
0	1	80BE0000	BadProtocolVersion Unsupported	The applications do not have compatible protocol versions.
0	1	80C10000	BadFilterOperatorInvalid	An unrecognized operator was provided in a filter.
0	1	80C20000	BadFilterOperator Unsupported	A valid operator was provided, but the server does not provide support for this filter operator.
0	1	80C30000	BadFilterOperandCount Mismatch	The number of operands provided for the filter operator was less than expected for the operand provided.
0	1	80C40000	BadFilterElementInvalid	The referenced element is not a valid element in the content filter.
0	1	80C50000	BadFilterLiteralInvalid	The referenced literal is not a valid value.
0	1	80C90000	BadViewTimestampInvalid	The view timestamp is not available or not supported.
0	1	80CA0000	BadViewParameterMismatch	The view parameters are not consistent with each other.
0	1	80CB0000	BadViewVersionInvalid	The view version is not available or not supported.
0	1	80CC0000	BadConditionAlready Enabled	This condition has already been enabled.
0	1	80CD0000	BadDialogNotActive	The dialog condition is not active.
0	1	80CF0000	BadConditionBranch AlreadyAcked	The condition branch has already been acknowledged.
0	1	80D00000	BadConditionBranch AlreadyConfirmed	The condition branch has already been confirmed.
0	1	80D10000	BadConditionAlreadyShel-ved	The condition has already been shelved.
0	1	80D20000	BadConditionNotShelved	The condition is not currently shelved.
0	1	80D30000	BadShelvingTimeOutOfRan-ge	The shelving time not within an acceptable range.
0	1	80D40000	BadAggregateListMismatch	The requested number of Aggregates does not match the requested number of NodeIds.
0	1	80D50000	BadAggregateNotSupported	The requested Aggregate is not support by the server.
0	1	80D60000	BadAggregateInvalidInputs	The aggregate value could not be derived due to invalid data inputs.
0	1	80DB0000	BadTooManyMonitoredItems	The request could not be processed because there are too many monitored items in the subscription.
0	1	80D70000	BadBoundNotFound	No data found to provide upper or lower bound value.
0	1	80D80000	BadBoundNotSupported	The server cannot retrieve a bound for the variable.
0	1	00D90000	GoodDataIgnored	The request specifies fields which are not valid for the EventType or cannot be saved by the historian.
0	1	80DA0000	BadAggregateConfiguration Rejected	The aggregate configuration is not valid for specified node.
Fehlernummern von PLCOpen für OPC UA				
0	1	A0000001	PLCopenUA_Bad_FW_ PermanentError	Internal, permanent error.
0	1	A0000002	PLCopenUA_Bad_FW_ TempError	Temp. Error; FB could retry to reach FW.
0	1	A0000003	PLCopenUA_Bad_ ConnectionError	Connection could not be established.

Done	Error	ErrorID (hex)	Name	Bedeutung
0	1	A0000004	PLCopenUA_Bad_ HostNotFound	The requested hostname could not be found.
0	1	A0000005	PLCopenUA_Bad_ AlreadyConnected	Connection was already established.
0	1	A0000006	PLCopenUA_Bad_ SecurityFailed	Connection failed due to security setup.
0	1	A0000007	PLCopenUA_Bad_ Suspended	Connection is suspended.
0	1	A0000008	PLCopenUA_Bad_ ConnectionInvalidHdl	Provided ConnectionHdl is not known.
0	1	A0000009	PLCopenUA_Bad_ NSNotFound	A namespace with the requested name cannot be found on server.
0	1	A000000A	PLCopenUA_Bad_ ResultTooLong	Target PLC variable is too short for retrieved data.
0	1	A000000B	PLCopenUA_Bad_ InvalidType	Invalid or unsupported Type.
0	1	A000000C	PLCopenUA_Bad_ NodeInvalidHdl	Provided NodeHdl is not known.
0	1	A000000D	PLCopenUA_Bad_ MethodInvalidHdl	Provided MethodHdl is not known.
0	1	A000000E	PLCopenUA_Bad_ ReadFailed	Read failed for unknown reason.
0	1	A000000F	PLCopenUA_Bad_ WriteFailed	Write failed for unknown reason.
0	1	A0000010	PLCopenUA_Bad_ CallFailed	Method Call failed for unknown reason.
0	1	A0000011	PLCopenUA_Bad_ InParamFailed	Method Call Input parameter conversion failed.
0	1	A0000012	PLCopenUA_Bad_ OutParamFailed	Method Call Output parameter conversion failed. ATTENTI-ON: this means the MethodCall was executed successfully but the returned values could not be converted.
0	1	A0000013	PLCopenUA_Bad_ SubscriptionInvalidHdl	Provided SubscriptionHdl is not known.
0	1	A0000014	PLCopenUA_Bad_ MonitoredItemInvalidHdl	Provided MonitoredItemHdl is not known.

4.2.5.7 300C Funktionen

ASCII, 3964(R)

SEND_PTP: Daten senden (ASCII, 3964(R))

Beschreibung

Mit der Anweisung "SEND_PTP" senden Sie einen Datenblock aus einem Datenbaustein.

- Die Aktivierung des Sendevorgangs erfolgt nach Aufruf des Bausteins und positiver Flanke am Steuereingang REQ.
- Der Bereich der zu sendenden Daten wird durch SD_1 (DB-Nummer und Anfangsadresse) vorgegeben, die Länge des Datenblocks durch LEN.
- Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie sie mit R(Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird ein laufender Sendevorgang abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.
- Mit LADDR geben Sie die E-/A-Adresse Ihres Submoduls an, die Sie bei der Hardware-Konfiguration festgelegt haben.
- Es wird entweder DONE auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERRORauf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.

Wurde der Auftrag mit DONE=TRUE durchlaufen, bedeutet das:

- Bei Verwendung des ASCII-Treibers: Die Daten wurden an den Kommunikationspartner gesendet. Nicht sichergestellt ist, ob die Daten auch vom Kommunikationspartner empfangen wurden.
- Bei Verwendung der Prozedur 3964(R): Die Daten wurden an den Kommunikationspartner gesendet und von diesem positiv quittiert. Nicht sichergestellt ist, ob die Daten auch an die Partner-CPU übergeben wurden.

Im STATUS zeigt die CPU bei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an. DONE bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben.

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird die Anweisung ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Instanz-DB

Die Anweisung "SEND_PTP" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND_PTP":

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
REQ	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Request: Aktiviert den Datenaustausch bei positiver Flanke.
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset. Auftrag wird abgebrochen. Senden gesperrt.
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E-/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben.
DONE	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): • FALSE: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • TRUE: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SD_1	InOut	ANY	CPU-spezifisch	0	Sendeparameter: Hier geben Sie folgende Werte an: • Nummer des DB, aus dem die Daten gesendet werden sollen. • Datenbytenummer, ab der die Daten gesendet werden sollen. Mögliche Werte: 0 bis 8190 Z. B.: DB 10 ab Byte 2 -> DB10.DBB2 Hinweis: Beachten Sie, dass der Parameter SD_1 bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
LEN	InOut	INT	1 bis 1024	1	Hier geben Sie die Länge des zu sendenden Datenblocks in Byte an (Die Länge wird hier indirekt angegeben.).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datenkonsistenz

Die Datenkonsistenz ist auf 206 Byte begrenzt. Für die konsistente Datenübertragung von mehr als 206 Byte müssen Sie Folgendes beachten:

Beschreiben Sie den aktuell benutzten Teil des Sendebereichs SD_1 erst dann wieder, wenn der Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert TRUE annimmt.

RCV_PTP: Daten empfangen (ASCII, 3964(R))

Beschreibung

Mit der Anweisung "RCV_PTP" empfangen Sie Daten und legen sie in einen Datenbaustein ab.

- Die Anweisung ist nach Aufruf mit dem Wert TRUE am Steuereingang EN_Rempfangsbereit. Eine laufende Übertragung können Sie mit Signalzustand FALSE am Parameter EN_R abbrechen. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange der Signalzustand FALSE am Parameter EN_R ansteht.
- Der Empfangsbereich wird durch RD_1 (DB-Nummer und Anfangsadresse) vorgegeben, die Länge des Datenblocks durch LEN.

- Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie ihn mit R(Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird die laufende Übertragung abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Empfangsauftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.
- Mit LADDR geben Sie die E-/A-Adresse Ihres Submoduls an, die Sie bei der Hardware-Konfiguration festgelegt haben.
- Es wird entweder NDR auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERROR auf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.
- Im STATUS zeigt die CPU bei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an.
- NDR bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben (Parameter LEN == 16#00).

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird die Anweisung ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Instanz-DB

Die Anweisung "RCV_PTP" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV_PTP":

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
EN_R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Enable to receive: Empfangsfreigabe
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset: Auftrag wird abgebrochen.
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E-/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben.
NDR	Output	BOOL	TRUE/FALSE	FALSE	Auftrag fertig ohne Fehler, Daten übernommen • FALSE: Auftrag wurde noch nicht gestartet oder läuft noch. • TRUE: Auftrag wurde erfolgreich abgeschlossen.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.
RD_1	InOut	ANY	CPU-spezifisch	0	Empfangsparameter: Hier geben Sie an: • Nummer des DB, in den die empfangenen Daten abgelegt werden. • Datenbytenummer, ab der die empfangenen Daten abgelegt werden sollen. Mögliche Werte: 0 bis 8190 Z. B.: DB 20 ab Byte 5 -> DB20.DBB5 Hinweis: Beachten Sie, dass der Parameter RD_1 bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
LEN	InOut	INT	0 bis 1024	0	Ausgabe der Datenlänge (Anzahl Bytes)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datenkonsistenz

Die Datenkonsistenz ist auf 206 Byte begrenzt. Für die konsistente Datenübertragung von mehr als 206 Byte müssen Sie Folgendes beachten:

Greifen Sie auf den Empfangs-DB erst wieder zu, wenn die Daten komplett empfangen wurden (NDR = TRUE). Sperren Sie den Empfangs-DB danach solange (EN_R = FALSE), bis Sie die Daten bearbeitet haben.

RES_RCVB: Empfangspuffer zurücksetzen (ASCII, 3964(R))

Beschreibung

Mit der Anweisung "RES_RECV" löschen Sie den kompletten Empfangspuffer der Baugruppe. Alle gespeicherten Telegramme werden verworfen. Ein zum Zeitpunkt des Aufrufs der Anweisung "RES_RCVB" eingehendes Telegramm wird gespeichert.

- Die Aktivierung des Auftrags erfolgt nach Aufruf der Anweisung und positiver Flanke am Steuereingang REQ. Der Auftrag kann über mehrere Aufrufe (Programmzyklen) laufen.
- Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie sie mit R (Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird der Löschvorgang abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.
- Mit LADDR geben Sie die E-/A-Adresse Ihres Submoduls an, die Sie bei der Hardware-Konfiguration festgelegt haben.
- Es wird entweder DONE auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERROR auf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.
- Im STATUS zeigt die CPUbei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an.
- DONE bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben.

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Instanz-DB

Die Anweisung "RES_RCVB" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RES_RCVB":

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
REQ	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Request: Aktiviert den Auftrag bei positiver Flanke
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset: Auftrag wird abgebrochen.

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E-/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben.
DONE	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): • FALSE: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • TRUE: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

RK 512

SEND_RK: Daten senden (RK 512)

Beschreibung

Mit der Anweisung SEND_RK senden Sie einen Datenblock aus einem Datenbaustein.

- Die Aktivierung des Sendevorgangs erfolgt nach Aufruf der Anweisung und positiver Flanke am Steuereingang REQ.
- Der Bereich der zu sendenden Daten wird durch SD_1 (DB-Nummer und Anfangsadresse) vorgegeben, die Länge des Datenblocks durch LEN.

- An der Anweisung geben Sie ebenfalls den Empfangsbereich beim Partner an. Diese Information wird von der CPU in den Telegrammkopf eingetragen und an den Partner übertragen.
- Das Ziel wird angegeben durch die CPU-Nummer R_CPU (nur relevant bei Mehrprozessorkommunikation), den Datentyp R_TYPE (Datenbausteine (DB) und erweiterte Datenbausteine (DX)), die Datenbausteinnummer R_DBNO und den Offset R_OFFSET, an den das erste Byte geschrieben werden soll.
- Mit R_CF_BYT und R_CF_BIT legen Sie das Koppelmerkerbyte und -bit auf der Partner-CPU fest.
- Mit dem Parameter SYNC_DB bestimmen Sie den DB, in dem die gemeinsamen Daten aller von Ihnen verwendeten Anweisungen für die Initialisierung im Anlauf und Synchronisation hinterlegt werden. Die DB-Nummer muss für alle in Ihrem Anwenderprogramm verwendeten Anweisungen identisch sein.
- Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie sie mit R(Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird der laufende Sendevorgang abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.
- Mit LADDR geben Sie die E-/A-Adresse Ihres Submoduls an, die Sie bei der Hardware-Konfiguration festgelegt haben.
- Es wird entweder DONE auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERROR auf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.
- Wurde der Auftrag mit DONE = TRUE durchlaufen, wurden die Daten an den Kommunikationspartner gesendet, von diesem positiv quittiert und die Daten an die Partner-CPU übergeben.
- Im STATUS zeigt die CPU bei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an.
- DONE bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben.

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Instanz-DB

Die Anweisung "SEND_RK" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Besonderheiten bei dem Senden von Daten

Beachten Sie die folgenden Besonderheiten beim "Daten senden":

- Mit RK 512 kann nur eine gerade Anzahl von Daten gesendet werden. Wenn Sie als Länge (LEN) eine ungerade Anzahl von Daten angeben, wird ein zusätzliches Füllbyte mit dem Wert "0" am Ende der Daten übertragen.
- Mit RK 512 kann nur ein gerader Offset angegeben werden. Bei Angabe eines ungeraden Offsets werden die Daten ab dem nächst kleineren geraden Offset beim Partner abgelegt. Beispiel: Offset ist "7", abgelegt werden die Daten ab Byte "6".

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND_RK":

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SYNC_DB	Input	INT	CPU- spezifisch	0	Nummer des DB, in dem die gemeinsamen Daten zur Synchronisation der RK-Anweisungen abgelegt werden (Mindestlänge 240 Byte).
REQ	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Request: Aktiviert den Auftrag bei positiver Flanke
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset: Auftrag wird abgebrochen.
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E-/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben.
R_CPU	Input	INT	0 bis 4	1	CPU-Nummer der Partner-CPU (nur bei Mehrprozessorbetrieb)
R_TYPE	Input	CHAR	'D', 'X'	'D'	Adresstyp auf Partner-CPU (nur Großbuchstaben erlaubt) 'D': Datenbaustein 'X': erweiterter Datenbaustein
R_DBNO	Input	INT	0 bis 255	0	Datenbausteinnummer auf Partner-CPU
R_OFFSET	Input	INT	0 bis 510 (nur gerad-zahlige Werte)	0	Datenbytenummer auf Partner-CPU
R_CF_BYT	Input	INT	0 bis 255	255	Koppelmerkerbyte auf Partner-CPU (255: Bedeutet: ohne Koppelmerker)
R_CF_BIT	Input	INT	0 bis 7	0	Koppelmerkerbit auf Partner-CPU
DONE	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): • FALSE: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • TRUE: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.
SD_1	Input	ANY	CPU-spezifisch	0	Sendeparameter: Hier geben Sie an: • Nummer des DB, aus dem die Daten gesendet werden sollen. • Datenbytenummer, ab der die Daten gesendet werden sollen. Mögliche Werte: 0 bis 8190 Z. B.: DB 10 ab Byte 2 -> DB10.DBB2 Hinweis: Beachten Sie, dass der Parameter SD_1 bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
LEN	Input	INT	1 bis 1024	1	Hier geben Sie die Länge des zu sendenden Datenblocks in Byte an (Die Länge wird hier indirekt angegeben.).

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Angaben im Telegrammkopf

In der folgenden Tabelle sind die Angaben im Telegrammkopf des RK 512-Telegramms dargestellt.

Quelle auf Ihrem S7- Automatisierungssystem (lokale CPU)	zum Ziel, Partner-CPU	Telegrammkopf, Bytes		
		3/4 Befehlsart	5/6 Z-DBNR[1] / Z-Offset[2]	7/8 Anzahl in
Datenbaustein	Datenbaustein	AD	DB/DW[3]	Wörtern
Datenbaustein	Erweiterter Datenbaustein	AD	DB/DW[3]	Wörtern

[1] Z-DBNR: Ziel-Datenbausteinnummer
[2] Z-Offset: Zielanfangsadresse
[3] DW: Offset in Worten

Datenkonsistenz

Die Datenkonsistenz ist auf 128 Byte begrenzt. Für die konsistente Datenübertragung von mehr als 128 Byte müssen Sie Folgendes beachten:

Beschreiben Sie den aktuell benutzten Teil des Sendebereichs SD_1 erst dann wieder, wenn der Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert TRUE annimmt.

FETCH_RK: Daten holen (RK 512)

Beschreibung

Mit der Anweisung "FETCH_RK" holen Sie vom Partner einen Datenblock und legt die Daten in einen Datenbaustein ab.

- Die Aktivierung des Sendevorgangs erfolgt nach Aufruf des Bausteins und positiver Flanke am Steuereingang REQ.
- Der Bereich, in den die geholten Daten abgelegt werden, wird durch RD_1 (DB-Nummer und Anfangsadresse) vorgegeben, die Länge des Datenblocks durch LEN.
- An der Anweisung geben Sie ebenfalls an, aus welchem Bereich beim Partner die Daten geholt werden. Diese Information wird von der CPU in den RK512-Telegrammkopf eingetragen und an den Partner übertragen.
- Der Bereich beim Partner wird angegeben durch die CPU-Nummer R_CPU (nur relevant bei Mehrprozessorkommunikation), den Datentyp R_TYPE (Datenbausteine, erweiterte Datenbausteine, Merker, Eingänge, Ausgänge, Zähler und Zeiten), die Datenbausteinnummer R_DBNO (nur relevant bei Datenbausteinen und erweiterten Datenbausteinen) und den Offset R_OFFSET, aus dem das erste Byte geholt werden soll.
- Mit R_CF_BYT und R_CF_BIT werden das Koppelmerkerbyte und das Koppelmerkerbit auf der Partner-CPU festgelegt.

- Mit dem Parameter SYNC_DB bestimmen Sie den DB, in dem die gemeinsamen Daten aller von Ihnen verwendeten Anweisungen für die Initialisierung im Anlauf und Synchronisation hinterlegt werden. Die DB-Nummer muss für alle in Ihrem Anwenderprogramm verwendeten Anweisungen identisch sein.
- Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie sie mit R (Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird die laufende Übertragung abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.
- Mit LADDR geben Sie die E/A-Adresse Ihres Submoduls an, die Sie bei der Hardware-Konfiguration festgelegt haben.
- Es wird entweder DONE auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERROR auf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.
- Im STATUS zeigt die CPU bei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an.
- DONE bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben.

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird die Anweisung ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen. Wenn Daten von Ihrer CPU geholt werden, müssen Sie auf Ihrer CPU eine Anweisung "SERVE_RK (Seite 7062)" programmieren.

Instanz-DB

Die Anweisung "FETCH_RK" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Besonderheiten bei (erweiterten) Datenbausteinen

Beachten Sie die folgenden Besonderheiten beim "Daten holen" aus Datenbausteinen und erweiterten Datenbausteinen:

- Mit RK 512 kann nur eine gerade Anzahl von Daten geholt werden. Wenn Sie als Länge (LEN) eine ungerade Anzahl angeben, wird immer ein Byte mehr übertragen. Im Ziel-DB wird jedoch die korrekte Anzahl Daten eingetragen.
- Mit RK 512 kann nur eine gerader Offset angegeben werden. Bei Angabe eines ungeraden Offsets werden die Daten aus dem nächst kleineren geraden Offset beim Partner geholt. Beispiel: Offset ist 7, geholt werden die Daten ab Byte 6.

Besonderheiten bei Zeiten und Zählern

Wenn Sie vom Kommunikationspartner Zeiten oder Zähler holen, müssen Sie berücksichtigen, dass für jede Zeit bzw. Zähler 2 Bytes geholt werden. Wenn Sie z. B. 10 Zähler holen wollen, müssen Sie als Länge 20 eingeben.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FETCH_RK".

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SYNC_DB	Input	INT	CPU-spezifisch	0	Nummer des DB, in dem die gemeinsamen Daten zur Synchronisation der RK-Anweisungen abgelegt werden (Mindestlänge 240 Byte).
REQ	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Request: Aktiviert den Auftrag bei positiver Flanke
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset: Auftrag wird abgebrochen.
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E-/A-Adresse Ihres Submoduls, die Sie bi der Hardware-Konfiguration festgelegt haben.
R_CPU	Input	INT	0 bis 4	1	CPU-Nr. der Partner-CPU (nur bei Mehrprozessorbetrieb)
R_TYPE	Input	CHAR	'D', 'X', 'M', 'E', 'A', 'Z', 'T'	'D'	Adresstyp auf Partner-CPU • 'D': Datenbaustein • 'X': Erweiterter Datenbaustein • 'M': Merker • 'E': Eingänge • 'A': Ausgänge • 'Z': Zähler • 'T': Zeiten
R_DBNO	Input	INT	0 bis 255	0	Datenbausteinnummer auf Partner-CPU
R_OFFSET	Input	INT	Siehe Tabelle: "Parameter für Datenquelle (Partner CPU)"	0	Datenbytenummer auf Partner-CPU
R_CF_BYT	Input	INT	0 bis 255	255	Koppelmerkerbyte auf Partner-CPU (255: Bedeutet: ohne Koppelmerker)
R_CF_BIT	Input	INT	0 bis 7	0	Koppelmerkerbit auf Partner-CPU
DONE	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): • FALSE: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • TRUE: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.
RD_1	Input	ANY	CPU-spezifisch	0	Empfangsparameter: Hier geben Sie an: • Nummer des DB, in den die geholten Daten abgelegt werden. • Datenbytenummer, ab der die geholten Daten abgelegt werden. Mögliche Werte: 0 bis 8190 Z. B.: DB 10 ab Byte 2 -> DB10.DBB2 Hinweis: Beachten Sie, dass der Parameter RD_1 bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPUs unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm.
LEN	Input	INT	1 bis 1024	1	Hier geben Sie die Länge des zu holenden Datenblocks in Byte an (Die Länge wird hier indirekt angegeben.). Pro Zeit und Zähler muss als Länge zwei Byte angegeben werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter für Datenquelle (Partner-CPU)

In der folgenden Tabelle sind die übertragbaren Datentypen dargestellt.
Der Wert "R_OFFSET" ist durch die Partner-CPU vorgegeben.

Quelle auf Partner-CPU	R_TYPE	R_NO	R_OFFSET (in Byte)
Datenbaustein	'D'	0 - 255	0 - 510, nur geradzahlige Werte sinnvoll
Erweiterter Datenbaustein	'X'	0 - 255	0 - 510, nur geradzahlige Werte sinnvoll
Merker	'M'	irrelevant	0 - 255
Eingänge	'E'	irrelevant	0 - 255
Ausgänge	'A'	irrelevant	0 - 255
Zähler	'Z'	irrelevant	0 - 255
Zeiten	'T'	irrelevant	0 - 255

Angaben im Telegrammkopf

In der folgenden Tabelle sind die Angaben im Telegrammkopf des RK 512-Telegramms dargestellt.

Quelle auf Partner-CPU	zum Ziel, Ihr S7- Automatisierungssystem (lokale CPU)	Telegrammkopf, Bytes		
		3/4 Befehlsart	5/6 Q-DBNR[1] / Q-Offset[2]	7/8 Anzahl in
Datenbaustein	Datenbaustein	ED	DB/DW	Wörtern
Erweiterter Datenbaustein	Datenbaustein	EX	DB/DW	Wörtern
Merker	Datenbaustein	EM	Byteadresse	Bytes
Eingänge	Datenbaustein	EE	Byteadresse	Bytes
Ausgänge	Datenbaustein	EA	Byteadresse	Bytes
Zähler	Datenbaustein	EZ	Zählernummer	Wörtern
Zeiten	Datenbaustein	ET	Zeitnummer	Wörtern

[1] Q-DBNR: Quell-Datenbausteinnummer
[2] Q-Offset: Quellanfangsadresse

Datenkonsistenz

Die Datenkonsistenz ist auf 128 Byte begrenzt. Für die konsistente Datenübertragung von mehr als 128 Byte müssen Sie Folgendes beachten:

Sie dürfen den aktuell benutzten Teil des Empfangsbereichs RD_1 erst dann wieder beschreiben, wenn der Übertragungsvorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert TRUE annimmt.

SERVE_RK: Daten empfangen und bereitstellen (RK 512)

Beschreibung

Die Anweisung "SERVE_RK" verwenden Sie zum Empfangen und Bereitstellen von Daten.

- Empfangen von Daten: Die Daten werden in den im RK 512-Telegrammkopf vom Partner spezifizierten Datenbereich abgelegt. Der Aufruf der Anweisung ist notwendig, wenn der Kommunikationspartner einen Auftrag "Daten senden" (SEND-Auftrag) durchführt.
- Bereitstellen von Daten: Die Daten werden aus dem im RK 512-Telegrammkopf vom Partner spezifizierten Datenbereich geholt. Der Aufruf der Anweisung ist notwendig, wenn der Kommunikationspartner einen Auftrag "Daten holen" (FETCH-Auftrag) durchführt.

Die Anweisung ist nach Aufruf mit dem Wert TRUE am Steuereingang EN_R bereit. Eine laufende Übertragung können Sie mit Signalzustand FALSE am Parameter EN_R abbrechen. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet. Der Empfang ist ausgeschaltet, solange der Signalzustand FALSE am Parameter EN_R ansteht.

Mit dem Parameter "SYNC_DB" bestimmen Sie den DB, in dem die gemeinsamen Daten aller von Ihnen verwendeten Anweisungen für die Initialisierung im Anlauf und Synchronisation hinterlegt werden. Die DB-Nummer muss für alle in Ihrem Anwenderprogramm verwendeten Anweisungen identisch sein.

Damit die Anweisung den Auftrag bearbeiten kann, müssen Sie sie mit R (Reset)=FALSE aufrufen. Bei einer positiven Flanke am Steuereingang R wird die laufende Übertragung abgebrochen und die Anweisung in den Grundzustand versetzt. Ein abgebrochener Auftrag wird mit einer Fehlermeldung (STATUS-Ausgang) beendet.

Mit LADDR geben Sie die E-/A-Adresse Ihres Submoduls an, die Sie in HW-Konfig festgelegt haben.

Es wird entweder NDR auf TRUE gesetzt, wenn der Auftrag ohne Fehler beendet wurde oder ERROR auf TRUE gesetzt, wenn der Auftrag mit Fehler beendet wurde.

Die CPU zeigt mit NDR=TRUE für einen Aufruf der Anweisung an den Parametern L_TYPE, L_DBNO und L_OFFSET an, wo die Daten abgelegt bzw. von wo die Daten geholt wurden. Zusätzlich werden für einen Aufruf die Parameter L_CF_BYT und L_CF_BIT und die Länge LEN des entsprechenden Auftrags angezeigt.

Im STATUS zeigt die CPU bei einem Fehler bzw. bei einer Warnung die entsprechende Ereignisnummer an.

NDR bzw. ERROR/STATUS werden auch bei RESET der Anweisung (R=TRUE) ausgegeben (Parameter LEN == 16#00).

Bei einem aufgetretenen Fehler wird das Binärergebnis BIE rückgesetzt. Wird der Baustein ohne Fehler beendet, hat das Binärergebnis den Zustand TRUE.

Hinweis

Die Anweisung hat keine Parameterprüfung, bei falscher Parametrierung kann die CPU in den Zustand STOP verzweigen.

Instanz-DB

Die Anweisung "SERVE_RK" arbeitet mit einem Instanz-DB zusammen. Die DB-Nummer wird beim Aufruf mitgegeben. Ein Zugriff auf die Daten im Instanz-DB ist nicht zulässig.

Anwendung der Koppelmerker

Über einen Koppelmerker können Sie SEND- und FETCH-Aufträge Ihres Kommunikationspartners sperren und freigeben. So können Sie verhindern, dass Daten, die noch nicht bearbeitet wurden, überschrieben oder gelesen werden.

Sie können für jeden Auftrag einen Koppelmerker festlegen.

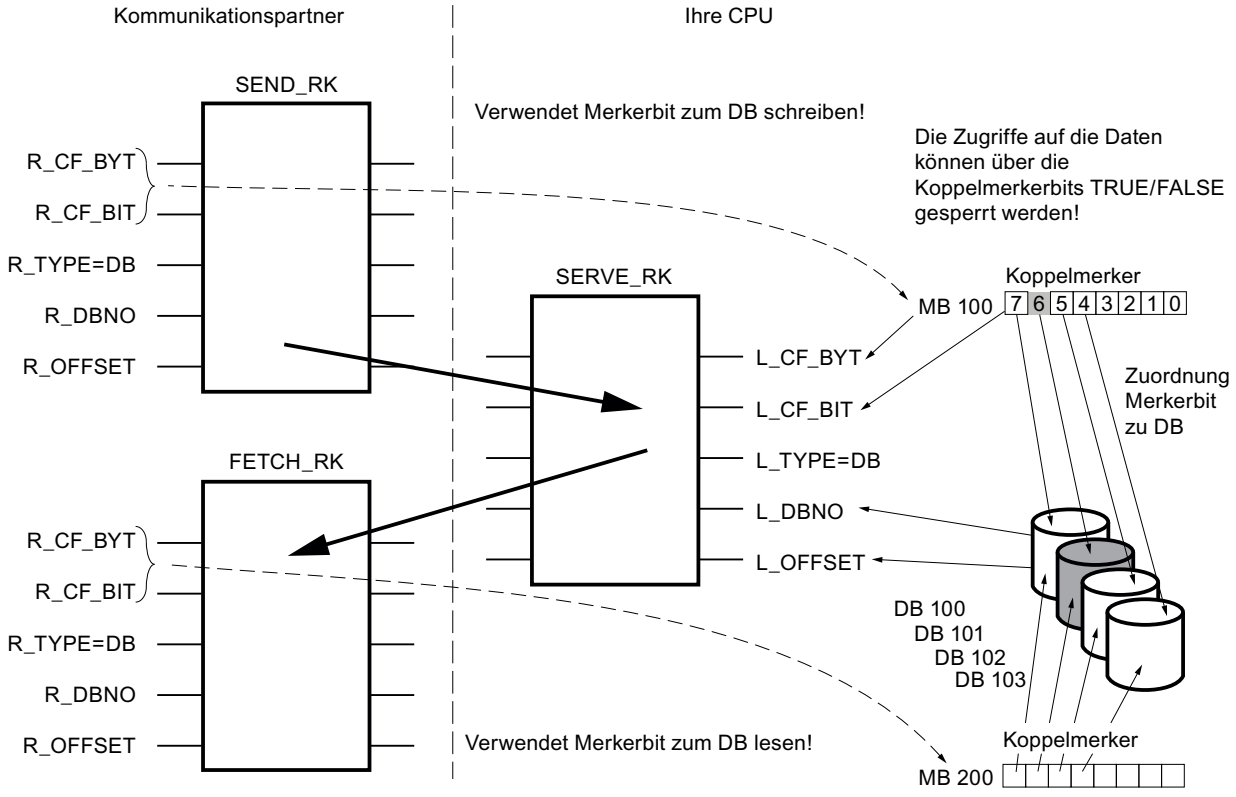

Beispiel: SEND_RK mit Koppelmerker:

In dem Beispiel sendet der Kommunikationspartner Daten in den DB 101 auf Ihrer CPU

1. Setzen Sie auf Ihrer CPU den Koppelmerker 100.6 auf FALSE.
2. Geben Sie beim Kommunikationspartner am SEND-Auftrag den Koppelmerker 100.6 (Parameter R_CF_BYT, R_CF_BIT) an.
   Der Koppelmerker wird im RK 512-Telegrammkopf an Ihre CPU übertragen.
   Vor der Bearbeitung des Auftrags überprüft Ihre CPU den im RK 512-Telegrammkopf angegebenen Koppelmerker. Der Auftrag wird nur bearbeitet, wenn der Koppelmerker auf Ihrer CPU den Wert FALSE hat. Hat der Koppelmerker den Wert TRUE, wird im Reaktionstelegramm die Fehlermeldung "32 hex" an den Kommunikationspartner gesendet.
   Nachdem die Daten in den DB101 übertragen worden sind, wird der Koppelmerker 100.6 auf Ihrer CPU von SERVE auf den Wert TRUE gesetzt und Koppelmerker-Byte und Bit an SERVE für einen Aufruf (wenn NDR =TRUE ist) ausgegeben.
3. Im Anwenderprogramm können Sie durch Auswertung des Koppelmerkers (Koppelmerker 100.6 =TRUE) erkennen, dass der Auftrag beendet ist und die übertragenen Daten bearbeitet werden können.
4. Nachdem Sie die Daten in Ihrem Anwenderprogramm bearbeitet haben, müssen Sie den Koppelmerker 100.6 wieder auf FALSE setzen. Erst dann kann Ihr Koppelpartner den Auftrag wieder ohne Fehler ausführen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SERVE_RK".

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
SYNC_DB	Input	INT	CPU-spezifisch	0	Nummer des DB in dem die gemeinsamen Daten zur Synchronisation der RK-Anweisungen abgelegt werden (Mindestlänge 240 Byte).
EN_R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Enable to receive: Auftragsfreigabe
R	Input	BOOL	TRUE/FALSE	FALSE	Steuerparameter Reset: Auftrag wird abgebrochen.
LADDR	Input	WORD	CPU-spezifisch	W#16#03FF	E/A-Adresse Ihres Submoduls, die Sie bei der Hardware-Konfiguration festgelegt haben.
NDR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter New Data Ready (Der Parameter ist nur einen Aufruf lang gesetzt.): • FALSE: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • TRUE: Auftrag wurde erfolgreich ausgeführt.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt.): Auftrag fertig mit Fehler

Parameter	Deklaration	Datentyp	Wertebereich	Vorbelegung	Beschreibung
STATUS	Output	WORD	W#16#0000 bis W#16#FFFF	W#16#0000	Zustandsparameter (Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren) STATUS hat in Abhängigkeit vom ERROR-Bit folgende Bedeutung: • ERROR=FALSE: – STATUS hat den Wert W#16#0000: Weder Warnung noch Fehler – STATUS hat den Wert<> W#16#0000: Warnung, STATUS liefert detaillierte Auskunft. • ERROR=TRUE: Es liegt ein Fehler vor, STATUS liefert detaillierte Auskunft über die Art des Fehlers.
L_TYPE	Output	CHAR	'D' 'D', 'M', 'E', 'A', 'Z', 'T',	' '	**Daten empfangen:** Typ des Zielbereichs auf lokaler CPU (nur Großbuchstaben erlaubt): • 'D': Datenbaustein **Daten bereitstellen:** Typ des Quell-Bereiches auf lokaler CPU (nur Großbuchstaben erlaubt): • 'D': Datenbaustein • 'M': Merker • 'E': Eingänge • 'A': Ausgänge • 'Z': Zähler • 'T': Zeiten Der Parameter ist nur einen Aufruf lang gesetzt.
L_DBNO	Output	INT	CPU-spezifisch	0	Datenbausteinnummer auf lokaler CPU. Der Parameter ist nur einen Aufruf lang gesetzt.
L_OFFSET	Output	INT	0 - 510	0	Datenbytenummer auf lokaler CPU. Der Parameter ist nur einen Aufruf lang gesetzt.
L_CF_BYT	Output	INT	0 bis 255	0	Koppelmerkerbyte auf lokaler CPU. Der Parameter ist nur einen Aufruf lang gesetzt. (255: Bedeutet: ohne Koppelmerker)
L_CF_BIT	Output	INT	0 bis 7	0	Koppelmerkerbit auf lokaler CPU. Der Parameter ist nur einen Aufruf lang gesetzt.
LEN	InOut	INT	0 bis 1024	0	Länge des Telegramms, Anzahl in Byte (Der Parameter ist nur einen Aufruf lang gesetzt.)

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datenkonsistenz

Die Datenkonsistenz ist auf 128 Byte begrenzt. Für die konsistente Datenübertragung von mehr als 128 Byte müssen Sie Folgendes beachten:

Benutzen Sie die Koppelmerkerfunktion. Greifen Sie auf die Daten erst wieder zu, wenn die Daten komplett übertragen wurden (Auswertung des für diesen Auftrag festgelegten Koppelmerkers; Koppelmerker steht für einen Aufruf an der Anweisung an, wenn NDR = TRUE). Setzen Sie den Koppelmerker erst wieder auf FALSE, wenn Sie die Daten bearbeitet haben.

4.2.5.8 Kommunikation mit iSlave/iDevice

I_GET: Daten eines Kommunikationspartners innerhalb der eigenen S7-Station lesen

Beschreibung

Mit der Anweisung "I_GET" lesen Sie Daten aus einem Kommunikationspartner, der innerhalb der eigenen S7-Station liegt. Der Kommunikationspartner kann im Zentralgerät, in einem Erweiterungsgerät oder dezentral platziert sein. Bitte beachten Sie, dass Sie dezentral platzierte Kommunikationspartner der eigenen CPU zugeordnet haben. Auf dem Kommunikationspartner gibt es keine zugehörige Anweisung.

Die Aktivierung des Empfangsvorgangs erfolgt nach Aufruf der Anweisung mit REQ=1. Anschließend rufen Sie die Anweisung so lange auf, bis der Datenempfang mit BUSY=0 angezeigt wird. RET_VAL enthält dann die Länge des empfangenen Datenblocks in Byte.

Sie müssen darauf achten, dass der über den Parameter RD definierte Empfangsbereich (auf der empfangenden CPU) mindestens so lang ist wie der über den Parameter VAR_ADDR definierte Lesebereich (beim Kommunikationspartner). Darüber hinaus müssen die Datentypen bei RD und VAR_ADDR übereinstimmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "I_GET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
CONT	Input	BOOL	E, A, M, D, L	Steuerparameter "continue" Siehe auch: AUTOHOTSPOT

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
IOID	Input	BYTE	E, A, M, D, L, P oder Konstante	Kennung des Adressbereichs der Partnerbaugruppe: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L, P oder Konstante	Logische Adresse der Partnerbaugruppe Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
VAR_ADDR	Input	ANY	E, A, M, D	Referenz auf den Bereich in der Partner-CPU, der gelesen werden soll. Sie müssen einen Datentyp wählen, der vom Kommunikationspartner unterstützt wird.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode. Tritt kein Fehler auf, enthält RET_VAL die Länge des in den Empfangsbereich RD kopierten Datenblocks in Byte als positive Zahl.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Empfangsvorgang ist noch nicht abgeschlossen. • BUSY=0: Der Empfangsvorgang ist abgeschlossen, bzw. es ist kein Empfangsvorgang aktiv.
RD	Output	ANY	E, A, M, D	Referenz auf den Empfangsbereich (receive data area). Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. Der Empfangsbereich RD muss mindestens so lang sein wie der Lesebereich VAR_ADDR beim Kommunikationspartner. Außerdem müssen die Datentypen bei RD und VAR_ADDR übereinstimmen. Die maximale Größe des Empfangsbereichs ist 76 Bytes.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird die von "I_GET" aufgebaute Verbindung abgebaut. Ob die bereits empfangenen Daten, die in einem Zwischenspeicher des Betriebssystems stehen, verloren gehen, hängt von der Art des anschließenden Anlaufs ab:

- Bei einem Wiederanlauf (nicht bei S7-300 und bei S7-400H) werden diese Daten in den durch RD festgelegten Bereich kopiert.
- Bei einem Neustart (Warmstart) oder Kaltstart werden diese Daten verworfen.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für die Datenübertragung mit "I_GET" keine Auswirkung: Die Daten werden auch im Betriebszustand STOP gelesen.

Datenkonsistenz

Die Daten werden konsistent empfangen.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

I_PUT: Daten eines Kommunikationspartners innerhalb der eigenen S7-Station schreiben

Beschreibung

Mit der Anweisung "I_PUT" schreiben Sie Daten in einen Kommunikationspartner, der innerhalb der eigenen S7-Station liegt. Der Kommunikationspartner kann im Zentralgerät, in einem Erweiterungsgerät oder dezentral platziert sein. Bitte beachten Sie, dass Sie dezentral platzierte Kommunikationspartner der eigenen CPU zugeordnet haben. Auf dem Kommunikationspartner gibt es keine zugehörige Anweisung.

Der Sendevorgang erfolgt nach Aufruf der Anweisung mit 1-Pegel am Steuereingang REQ.

Sie müssen darauf achten, dass der über den Parameter SD definierte Sendebereich (auf der sendenden CPU) die gleiche Länge hat wie der über den Parameter VAR_ADDR definierte Empfangsbereich (beim Kommunikationspartner). Darüber hinaus müssen die Datentypen bei SD und VAR_ADDR übereinstimmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "I_PUT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
CONT	Input	BOOL	E, A, M, D, L	Steuerparameter "continue" Siehe auch: AUTOHOTSPOT
IOID	Input	BYTE	E, A, M, D, L, P oder Konstante	Kennung des Adressbereichs der Partnerbaugruppe: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L, P oder Konstante	Logische Adresse der Partnerbaugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
VAR_ADDR	Input	ANY	E, A, M, D, L	Referenz auf den Bereich in der Partner-CPU, in den geschrieben werden soll. Sie müssen einen Datentyp wählen, der vom Kommunikationspartner unterstützt wird.
SD	Input	ANY	E, A, M, D	Referenz auf den Bereich in der eigenen CPU, der die zu versendenden Daten enthält. Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. SD muss die gleiche Länge haben wie der Parameter VAR_ADDR des Kommunikationspartners. Außerdem müssen die Datentypen bei SD und VAR_ADDR übereinstimmen. Die maximale Größe des Sendebereichs ist 76 Bytes.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Sendevorgang ist noch nicht abgeschlossen. • BUSY=0: Der Sendevorgang ist abgeschlossen, bzw. es ist kein Sendevorgang aktiv.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird die von "I_PUT" aufgebaute Verbindung abgebaut. Die Daten können nicht mehr gesendet werden. Falls die Sendedaten zum Zeitpunkt des Betriebszustandsübergangs bereits in den internen Puffer kopiert waren, wird der Pufferinhalt verworfen.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für die Datenübertragung mit "I_PUT" keine Auswirkung. Die gesendeten Daten werden dennoch geschrieben.

Datenkonsistenz

Die Daten werden konsistent versendet.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

I_ABORT: Verbindung zu Kommunikationspartner innerhalb der eigenen S7-Station abbrechen

Beschreibung

Mit der Anweisung "I_ABORT" brechen Sie die Verbindung zu einem innerhalb der eigenen S7-Station liegenden Kommunikationspartner ab, die mit "I_GET (Seite 7066)" oder "I_PUT (Seite 7068)" aufgebaut wurde. Falls der zu "I_GET (Seite 7066)" oder "I_PUT (Seite 7068)" gehörige Auftrag abgeschlossen ist (BUSY = 0), sind nach dem Aufruf von "I_ABORT" die dafür belegten Verbindungsressourcen auf beiden Seiten wieder freigegeben.

Falls der zu "I_GET (Seite 7066)" oder "I_PUT (Seite 7068)" gehörige Auftrag nicht abgeschlossen ist (BUSY = 1), müssen Sie nach Abschluss des Verbindungsabbruchs die zugehörige Anweisung erneut mit REQ = 0 und CONT = 0 aufrufen und BUSY = 0 abwarten. Erst dann sind alle belegten Verbindungsressourcen wieder freigegeben.

Sie können "I_ABORT" nur auf derjenigen Seite aufrufen, auf der "I_PUT (Seite 7068)" oder "I_GET (Seite 7066)" abläuft (d.h. auf Client-Seite).

Die Aktivierung des Verbindungsabbruchs erfolgt nach Aufruf der Anweisung mit REQ=1.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "I_ABORT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
IOID	Input	BYTE	E, A, M, D, L, P oder Konstante	Kennung des Adressbereichs der Partnerbaugruppe: • B#16#54 = Peripherie Eingang (PE) • B#16#55 = Peripherie Ausgang (PA) Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben.
LADDR	Input	WORD	E, A, M, D, L, P oder Konstante	Logische Adresse der Partnerbaugruppe. Bei einer Mischbaugruppe ist die kleinere der beiden Adressen anzugeben.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Verbindungsabbruch ist noch nicht abgeschlossen. • BUSY=0: Der Verbindungsabbruch ist abgeschlossen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird ein über "I_ABORT" angestoßener Verbindungsabbruch zu Ende bearbeitet.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für den Verbindungsabbruch mit "I_ABORT" keine Auswirkung: Die Verbindung wird abgebrochen.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

4.2.5.9 PROFINET CBA

Informationen zu PN_IN, PN_OUT und PN_DP

Hinweis

Das Betriebssystem aktualisiert defaultmäßig sowohl die PROFINET CBA-Komponente als auch die DP-Verschaltungen am Zykluskontrollpunkt. Falls Sie jedoch diese automatischen Aktualisierungen bei der Projektierung abgeschaltet haben (z. B. um das Zeitverhalten der CPU besser beeinflussen zu können), müssen Sie die Aktualisierungen selbst vornehmen. Dies geschieht durch Aufruf der Anweisungen "PN_IN (Seite 7074)", "PN_OUT (Seite 7075)" und "PN_DP (Seite 7076)" zu geeigneten Zeitpunkten. Die genannten Aktualisierungen können bei der Projektierung nur gemeinsam abgeschaltet werden.

Schattenspeicher

Der Interface-DB ist die Anwenderprogramm-Schnittstelle der PROFINET CBA-Komponente. Um zu gewährleisten, dass seine Ein- und Ausgänge beim Ablauf der zugehörigen Anwenderprogrammteile konsistent sind, gibt es zu jedem Interface-DB einen vom Betriebssystem verwalteten identisch aufgebauten Speicherbereich, den so genannten Schattenspeicher. Sie können in Ihrem Programm nur auf den Interface-DB zugreifen, wobei andere (externe) PROFINET CBA-Komponenten nur auf den Schattenspeicher zugreifen können. Dadurch wird ein Zugriffskonflikt auf die Ein- und Ausgänge des Interface-DB ausgeschlossen.

Aktualisierung der PROFINET CBA-Komponente

Aus der oben beschriebenen Realisierung der Datenkonsistenz mittels Schattenspeicher ergibt sich, dass die Aktualisierung der PROFINET CBA-Komponente aus folgenden zwei Aktionen besteht:

- Kopieren der Eingänge des Schattenspeichers in die Eingänge des Interface-DB vor der Bearbeitung Ihres zur PROFINET CBA-Komponente gehörenden Programms
- Kopieren der Ausgänge des Interface-DB in die Ausgänge des Schattenspeichers nach der Bearbeitung Ihres zur PROFINET CBA-Komponente gehörenden Programms

Die Aktualisierung der PROFINET CBA-Komponente erfolgt bei CPUs mit integrierter PROFINET-Schnittstelle entweder durch das Betriebssystem oder durch die Anweisungen "PN_IN (Seite 7074)" und "PN_OUT (Seite 7075)" (je nachdem, wie Sie beim Erstellen der PROFINET CBA-Komponente projektiert haben.). Im folgenden Bild ist die Aktualisierung durch die Anweisungen "PN_IN (Seite 7074)" und "PN_OUT (Seite 7075)" schematisch dargestellt.

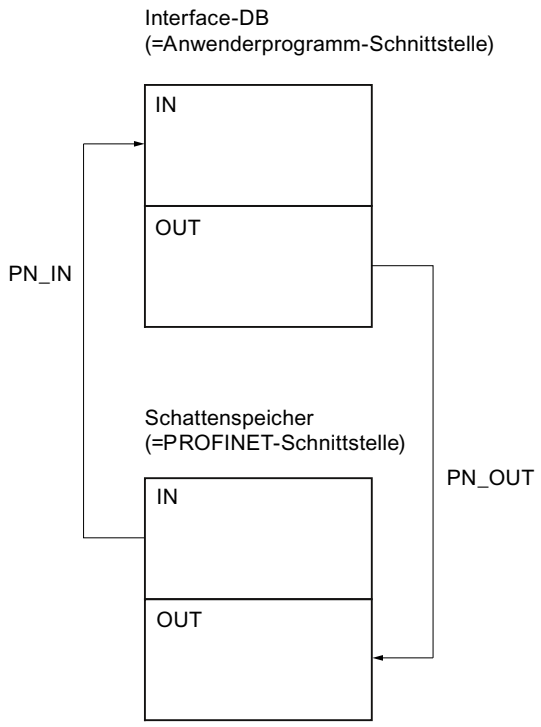

Die systemseitige Aktualisierung erfolgt stets am Zykluskontrollpunkt.

Falls Sie die Aktualisierung hingegen über die Anweisungen "PN_IN (Seite 7074)" und "PN_OUT (Seite 7075)" vornehmen, dann rufen Sie "PN_IN (Seite 7074)" zu Beginn des OB, in dem das zur PROFINET CBA-Komponente gehörende Programm liegt, auf, und "PN_OUT (Seite 7075)" am Ende dieses OB. Dieser Sachverhalt ist im folgenden Bild am Beispiel des OB 30 dargestellt.

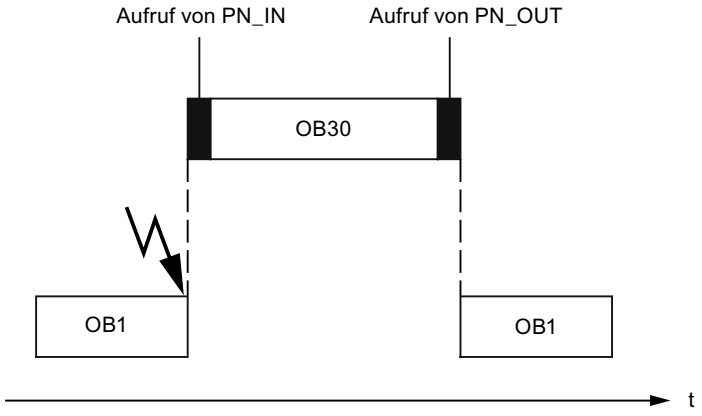

Hinweis

(mehrere PROFINET-Interface-DBs auf einem Gerät):

Bei Mutifunktionskomponenten gibt es die Möglichkeit, mit einem Auftrag (DBNO=0) sämtliche Interface-DBs zu aktualisieren (ähnlich der Aktualisierung am Zykluskontrollpunkt, lediglich per Anweisung getriggert). Tritt bei der Aktualisierung eines Interface-DB ein Fehler auf, wird die Aktualisierung der anderen Interface-DBs fortgeführt. Ein negativer RET_VAL bezieht sich auf einen der aktualisierten Interface-DBs.

Aktualisierung der DP-Verschaltungen

Die Aktualisierung der DP-Verschaltungen erfolgt entweder durch das Betriebssystem oder durch die Anweisung "PN_DP (Seite 7076)", je nachdem, wie Sie beim Erstellen der PROFINET CBA-Komponente projektiert haben.

PN_IN: Eingänge der Anwenderprogramm-Schnittstelle aktualisieren

Beschreibung

Mit der Anweisung "PN_IN" kopieren Sie die unter PROFINET CBA erhaltenen Eingangsdaten aus dem Schattenspeicher der PROFINET CBA-Komponente in den zugehörigen Interface-DB. Nach Abschluss der Anweisung stehen Ihrer Applikation die aktuellen Eingangsdaten zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PN_IN":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DBNO	Input	WORD	E, A, M, D, L oder Konstante	DB-Nr. des Interface-DB (DBNO=0: Aktualisierung sämtlicher PROFINET CBA-Interface-DBs)
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8001	PROFINET CBA-Projektierung ist nicht vorhanden oder ungültig.
8002	DB-Nr. stimmt nicht mit der Komponentenprojektierung überein.
8004	DB-Nr. stimmt mit der Komponentenprojektierung überein, der DB ist jedoch nicht geladen.

Fehlercode (W#16#...)	Erläuterung
8005	Der Interface-DB wurde mit dem Schlüsselwort UNLINKED compiliert, d. h. er befindet sich nur im Ladespeicher, nicht aber im Arbeitsspeicher.
8006	Der Interface-DB ist schreibgeschützt in der CPU.
80B1	Längenfehler beim Lesen bzw. Schreiben. Die Komponentenprojektierung passt nicht zum geladenen DB.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

PN_OUT: Ausgänge der Anwenderprogramm-Schnittstelle aktualisieren

Beschreibung

Mit der Anweisung "PN_OUT" kopieren Sie die in Ihrer Applikation erzeugten Ausgangsdaten aus dem Interface-DB der PROFINET CBA-Komponente in den zugehörigen Schattenspeicher. Nach Abschluss der Anweisung stehen anderen PROFINET CBA-Komponenten die aktuellen Ausgangsdaten zur Verfügung.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PN_OUT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DBNO	Input	WORD	E, A, M, D, L oder Konstante	DB-Nr. des Interface-DB (DBNO=0: Aktualisierung sämtlicher PROFINET-Interface-DBs)
RET_VAL	Return	INT	E, A, M, D, L	Fehlerinformation

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Es ist kein Fehler aufgetreten.
8001	PROFINET CBA-Projektierung ist nicht vorhanden oder ungültig.
8002	DB-Nr. stimmt nicht mit der Komponentenprojektierung überein.
8004	DB-Nr. stimmt mit der Komponentenprojektierung überein, der DB ist jedoch nicht geladen.
8005	Der Interface-DB wurde mit dem Schlüsselwort UNLINKED compiliert, d. h. er befindet sich nur im Ladespeicher, nicht aber im Arbeitsspeicher.
8006	Der Interface-DB ist schreibgeschützt in der CPU.
80B1	Längenfehler beim Lesen bzw. Schreiben. Die Komponentenprojektierung passt nicht zum geladenen DB.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

PN_DP: DP-Verschaltungen aktualisieren

Beschreibung

Mit der Anweisung "PN_DP" aktualisieren Sie alle

- Verschaltungen zwischen PROFINET CBA-Komponenten am lokalen PROFIBUS und
- Verschaltungen mit zyklischer Übertragung zwischen PROFINET CBA-Komponenten am lokalen PROFIBUS und externen PROFINET CBA-Komponenten. Diese Verschaltungen sind Netz übergreifend (zwischen Industrial Ethernet und PROFIBUS DP).

Arbeitsweise

"PN_DP" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich ggf. über mehrere Aufrufe. Sie starten die Aktualisierung der DP-Verschaltungen, indem Sie "PN_DP" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370).

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PN_DP":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	REQ=1: Aktualisierung der DP-Verschaltungen anstoßen
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	BUSY=1: Die Aktualisierung der DP-Verschaltungen ist noch nicht abgeschlossen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Parameter RET_VAL

Fehlercode (W#16#...)	Erläuterung
0000	Der Auftrag wurde fehlerfrei durchgeführt.
7000	Erstaufruf mit REQ=0. Es wird keine Aktualisierung der DP-Verschaltungen angestoßen. BUSY hat den Wert "0".
7001	(nur relevant bei S7-400:) Erstaufruf mit REQ=1. BUSY hat den Wert "1".

Fehlercode (W#16#...)	Erläuterung
7002	(nur relevant bei S7-400:) Zwischenaufruf (REQ irrelevant). Die Aktualisierung der DP-Verschaltungen ist noch nicht abgeschlossen. BUSY hat den Wert "1".
8001	PROFINET CBA-Projektierung ist nicht vorhanden oder ungültig.
8095	Sie haben eine weitere Aktualisierung der DP-Verschaltungen in einer höheren Prioritätsklasse angestoßen. Die Aktualisierung in der Prioritätsklasse mit niedriger Priorität (durch das Betriebssystem oder eine "PN_DP"-Bearbeitung) ist aber noch in Bearbeitung.
8xyy	allgemeine Fehlerinformation Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

4.2.5.10 MPI-Kommunikation

X_SEND: Daten an Kommunikationspartner außerhalb der eigenen S7-Station senden

Beschreibung

Mit der Anweisung "X_SEND" senden Sie Daten an einen außerhalb der eigenen S7-Station liegenden Kommunikationspartner. Der Datenempfang beim Kommunikationspartner erfolgt über die Anweisung "X_RCV (Seite 7078)". Der Sendevorgang erfolgt nach Aufruf der Anweisung mit REQ=1.

Sie müssen darauf achten, dass der über den Parameter SD definierte Sendebereich (auf der sendenden CPU) kleiner oder gleich dem über den Parameter RD definierten Empfangsbereich (beim Kommunikationspartner) ist. Falls SD vom Datentyp BOOL ist, muss auch RD vom Datentyp BOOL sein.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "X_SEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
CONT	Input	BOOL	E, A, M, D, L	Steuerparameter "continue" Siehe auch: AUTOHOTSPOT
DEST_ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter "destination ID". Er enthält die projektierte MPI-Adresse des Kommunikationspartners.
REQ_ID	Input	DWORD	E, A, M, D, L oder Konstante	Auftragskennung zur Identifizierung der Daten beim Kommunikationspartner.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SD	Input	ANY	E, A, M, D	Referenz auf den Sendebereich. Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. Die maximale Größe des Sendebereichs ist 76 Bytes.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Sendevorgang ist noch nicht abgeschlossen. • BUSY=0: Der Sendevorgang ist abgeschlossen, bzw. es ist kein Sendevorgang aktiv.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datenkonsistenz

Die Daten werden konsistent versendet.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

X_RCV: Daten von Kommunikationspartner außerhalb der eigenen S7-Station empfangen

Beschreibung

Mit der Anweisung "X_RCV" empfangen Sie Daten, die ein oder mehrere außerhalb der eigenen S7-Station liegende Kommunikationspartner mit der Anweisung "X_SEND (Seite 7077)" gesendet haben.

Mit der Anweisung "X_RCV"

- können Sie feststellen, ob zum aktuellen Zeitpunkt gesendete Daten bereitstehen. Diese wurden vom Betriebssystem gegebenenfalls in eine interne Warteschlange eingereiht.
- können Sie den ältesten Datenblock, der in der Warteschlange bereitsteht, in einen von Ihnen vorgegebenen Empfangsbereich kopieren.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "X_RCV":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_DT	Input	BOOL	E, A, M, D, L	Steuerparameter "enable data transfer". Mit dem Wert 0 überprüfen Sie, ob mindestens ein Datenblock bereitsteht. Der Wert "1" bewirkt das Umkopieren des ältesten in der Warteschlange vorhandenen Datenblocks in den Bereich des Arbeitsspeichers, den Sie durch RD vorgegeben haben.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode. Tritt kein Fehler auf, enthält RET_VAL • bei EN_DT=0/1 und NDA=0:W#16#7000. In diesem Fall steht kein Datenblock in der Warteschlange. • bei EN_DT=0 und NDA=1 die Länge des ältesten in der Warteschlange eingetragenen Datenblocks in Byte als positive Zahl. • bei EN_DT=1 und NDA=1 die Länge des in den Empfangsbereich RD kopierten Datenblocks in Byte als positive Zahl.
REQ_ID	Output	DWORD	E, A, M, D, L	Auftragskennung derjenigen Anweisung "X_SEND (Seite 7077)", deren gesendete Daten in der Warteschlange an erster Stelle stehen, d. h. die ältesten Daten in der Warteschlange. Falls kein Datenblock in der Warteschlange steht, enthält REQ_ID den Wert "0".

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDA	Output	BOOL	E, A, M, D, L	Zustandsparameter "new data arrived". NDA=0: • In der Warteschlange ist kein Datenblock vorhanden. NDA=1: • In der Warteschlange ist mindestens ein Datenblock vorhanden. (Aufruf von "X_RCV" mit EN_DT=0). • Der älteste Datenblock in der Warteschlange wurde in das Anwenderprogramm kopiert (Aufruf von "X_RCV" mit EN_DT=1).
RD	Output	ANY	E, A, M, D	Referenz auf den Empfangsbereich (receive data area). Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. Falls Sie den ältesten in der Warteschlange vorhandenen Datenblock verwerfen wollen, belegen Sie RD mit dem Wert NIL. Die maximale Größe des Empfangsbereichs ist 76 Bytes.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Datenempfang anzeigen mit EN_DT=0

Sobald Daten eines Kommunikationspartners angekommen sind, werden diese vom Betriebssystem in die Warteschlange eingereiht, und zwar in der Reihenfolge ihres Empfangs.

Wenn Sie überprüfen wollen, ob mindestens ein Datenblock bereitsteht, rufen Sie "X_RCV" mit EN_DT=0 auf und werten den Ausgangsparameter NDA aus:

- NDA=0 bedeutet, in der Warteschlange ist kein Datenblock eingetragen. REQ_ID ist irrelevant, RET_VAL enthält W#16#7000.
- NDA=1 bedeutet, in der Warteschlange steht mindestens ein Datenblock zur Abholung bereit.
  In diesem Fall werten Sie zusätzlich die Ausgangsparameter RET_VAL und ggf. REQ_ID aus. RET_VAL enthält die Länge des Datenblocks in Byte, REQ_ID die Auftragskennung des Sendebausteins. Falls in der Warteschlange mehrere Datenblöcke bereitstehen, gehören REQ_ID und RET_VAL zum ältesten in der Warteschlange liegenden Datenblock.

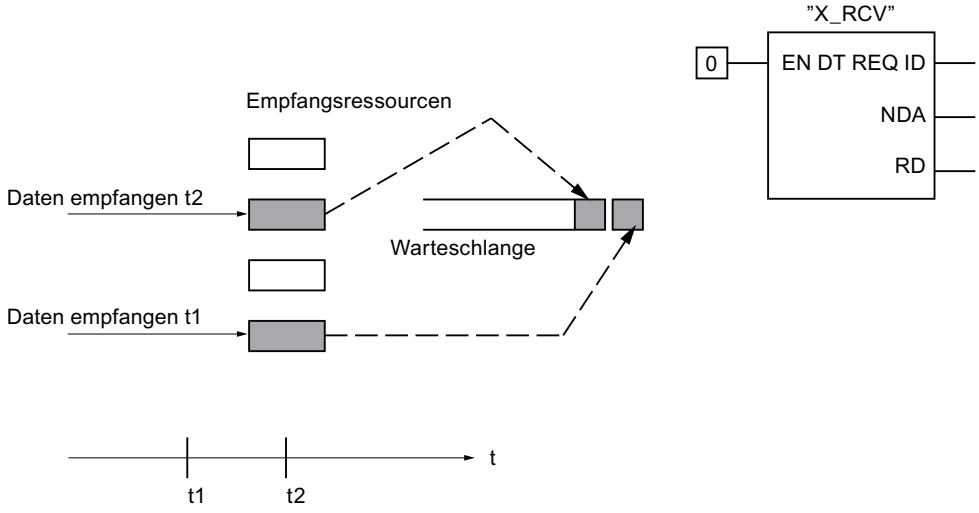

Daten in Empfangsbereich übernehmen mit EN_DT=1

Wenn Sie "X_RCV" mit EN_DT=1 aufrufen, wird der älteste in der Warteschlange vorhandene Datenblock in den durch RD vorgegebenen Bereich des Arbeitsspeichers kopiert. RD muss größer oder gleich dem über den Parameter SD definierten Sendebereich der zugehörigen Anweisung "X_SEND (Seite 7077)" sein. Falls dessen Eingangsparameter SD vom Datentyp BOOL ist, muss auch RD vom Datentyp BOOL sein. Wenn Sie die empfangenen Daten in unterschiedlichen Bereichen ablegen wollen, können Sie REQ_ID ermitteln (Anweisungs-Aufruf mit EN_DT=0) und im Folgeaufruf (mit EN_DT=1) RD geeignet wählen. Trat beim Kopieren kein Fehler auf, dann steht in RET_VAL die Länge des kopierten Datenblocks in Byte, und es erfolgt eine positive Quittung an den Sender.

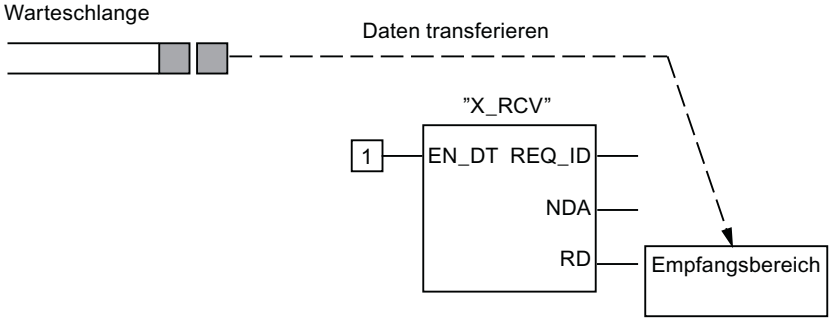

Daten verwerfen

Falls Sie die Daten nicht übernehmen wollen, belegen Sie RD mit dem Wert NIL. In diesem Fall erhält der Sender eine negative Quittung (RET_VAL der zugehörigen Anweisung "X_SEND (Seite 7077)" hat den Wert W#16#80B8), in RET_VAL bei der Anweisung "X_RCV" wird "0" eingetragen.

Datenkonsistenz

Nach Aufruf mit EN_DT=1 und RETVAL=W#16#00xy befinden sich neue Daten im Empfangsbereich RD. Ein weiterer Bausteinaufruf kann diese Daten überschreiben. Wenn Sie dies verhindern wollen, dürfen Sie "X_RCV" nicht mehr mit gleichem Empfangsbereich RD aufrufen, bevor Sie die empfangenen Daten ausgewertet haben.

Betriebszustandsübergang nach STOP

Beim Übergang in den Betriebszustand STOP

- werden alle neu ankommenden Aufträge negativ quittiert.
- gilt für die bereits angekommenen Aufträge: Alle in der Empfangswarteschlange eingetragenen Aufträge werden negativ quittiert.
  - Bei einem anschließenden Neustart (Warmstart) oder Kaltstart werden alle Datenblöcke verworfen.
  - Bei einem anschließenden Wiederanlauf (nicht bei S7-300 und bei S7-400H) wird der zum ältesten Auftrag gehörende Datenblock in das Anwenderprogramm übernommen, falls Sie ihn vor Übergang in den Betriebszustand STOP abgefragt haben (Aufruf von "X_RCV" mit EN_DT=0). Andernfalls wird er verworfen.
    Alle anderen Datenblöcke werden verworfen.

Verbindungsabbruch

Beim Abbruch einer Verbindung wird ein in der Empfangswarteschlange bereits eingetragener Auftrag, der zu dieser Verbindung gehört, verworfen.

Ausnahme: Falls dieser Auftrag der älteste Auftrag in der Warteschlange ist und Sie ihn durch Aufruf von "X_RCV" mit EN_DT=0 bereits erkannt haben, können Sie ihn mit EN_DT=1 in den Empfangsbereich übernehmen.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

X_GET: Daten aus Kommunikationspartner außerhalb der eigenen S7-Station lesen

Beschreibung

Mit der Anweisung "X_GET" lesen Sie Daten aus einem Kommunikationspartner, der außerhalb der eigenen S7-Station liegt. Auf dem Kommunikationspartner gibt es keine zugehörige Anweisung.

Die Aktivierung des Lesevorgangs erfolgt nach Aufruf der Anweisung mit REQ=1. Anschließend rufen Sie "X_GET" so lange auf, bis der Datenempfang mit BUSY=0 angezeigt wird. RET_VAL enthält dann die Länge des empfangenen Datenblocks in Byte.

Sie müssen darauf achten, dass der über den Parameter RD definierte Empfangsbereich (auf der empfangenden CPU) mindestens so lang ist wie der über den Parameter VAR_ADDR definierte Lesebereich (beim Kommunikationspartner). Darüber hinaus müssen die Datentypen bei RD und VAR_ADDR übereinstimmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "X_GET":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
CONT	Input	BOOL	E, A, M, D, L	Steuerparameter "continue" Siehe auch: AUTOHOTSPOT
DEST_ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter "destination ID". Er enthält die MPI-Adresse des Kommunikationspartners. Diese haben Sie projektiert.
VAR_ADDR	Input	ANY	E, A, M, D	Referenz auf den Bereich in der Partner-CPU, der gelesen werden soll. Sie müssen einen Datentyp wählen, der vom Kommunikationspartner unterstützt wird.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode. Tritt kein Fehler auf, enthält RET_VAL die Länge des in den Empfangsbereich RD kopierten Datenblocks in Byte als positive Zahl.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Empfangsvorgang ist noch nicht abgeschlossen. • BUSY=0: Der Empfangsvorgang ist abgeschlossen, bzw. es ist kein Empfangsvorgang aktiv.
RD	Output	ANY	E, A, M, D	Referenz auf den Empfangsbereich (receive data area). Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. Der Empfangsbereich RD muss mindestens so lang sein wie der Lesebereich VAR_ADDR beim Kommunikationspartner. Außerdem müssen die Datentypen bei RD und VAR_ADDR übereinstimmen. Die maximale Größe des Empfangsbereichs ist 76 Bytes.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird die von "X_GET" aufgebaute Verbindung abgebaut. Ob die bereits empfangenen Daten, die in einem Zwischenspeicher des Betriebssystems stehen, verloren gehen, hängt von der Art des anschließenden Anlaufs ab:

- Bei einem Wiederanlauf (nicht bei S7-300 und bei S7-400H) werden diese Daten in den durch RD festgelegten Bereich kopiert.
- Bei einem Neustart (Warmstart) oder Kaltstart werden diese Daten verworfen.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für die Datenübertragung mit "X_GET" keine Auswirkung: Die Daten werden auch im Betriebszustand STOP gelesen.

Datenkonsistenz

Die Daten werden konsistent empfangen.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

X_PUT: Daten in Kommunikationspartner außerhalb der eigenen S7-Station schreiben

Beschreibung

Mit der Anweisung "X_PUT" schreiben Sie Daten in einen Kommunikationspartner, der außerhalb der eigenen S7-Station liegt. Auf dem Kommunikationspartner gibt es keine zugehörige Anweisung.

Die Aktivierung des Schreibvorgangs erfolgt nach Aufruf der Anweisung mit REQ=1. Anschließend rufen Sie "X_PUT" so lange auf, bis der Quittungsempfang mit BUSY=0 angezeigt wird.

Sie müssen darauf achten, dass der über den Parameter SD definierte Sendebereich (auf der sendenden CPU) die gleiche Länge hat wie der über den Parameter VAR_ADDR definierte Empfangsbereich (beim Kommunikationspartner). Darüber hinaus müssen die Datentypen bei SD und VAR_ADDR übereinstimmen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "X_PUT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
CONT	Input	BOOL	E, A, M, D, L	Steuerparameter "continue" Siehe auch: AUTOHOTSPOT.
DEST_ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter "destination ID". Er enthält die MPI-Adresse des Kommunikationspartners. Diese haben Sie projektiert.
VAR_ADDR	Input	ANY	E, A, M, D	Referenz auf den Bereich in der Partner-CPU, in den geschrieben werden soll. Sie müssen einen Datentyp wählen, der vom Kommunikationspartner unterstützt wird.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
SD	Input	ANY	E, A, M, D	Referenz auf den Bereich in der eigenen CPU, der die zu versendenden Daten enthält. Folgende Datentypen sind erlaubt: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME,S5_TIME, DATE_AND_TIME sowie Arrays der genannten Datentypen mit Ausnahme von BOOL. SD muss die gleiche Länge haben wie der Parameter VAR_ADDR des Kommunikationspartners. Außerdem müssen die Datentypen bei SD und VAR_ADDR übereinstimmen. Die maximale Größe des Sendebereichs ist 76 Bytes.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Sendevorgang ist noch nicht abgeschlossen. • BUSY=0: Der Sendevorgang ist abgeschlossen, bzw. es ist kein Sendevorgang aktiv.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird die von "X_PUT" aufgebaute Verbindung abgebaut. Die Daten können nicht mehr gesendet werden. Falls die Sendedaten zum Zeitpunkt des Betriebszustandsübergangs bereits in den internen Puffer kopiert waren, wird der Pufferinhalt verworfen.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für die Datenübertragung mit "X_PUT" keine Auswirkung: Die gesendeten Daten werden dennoch geschrieben.

Datenkonsistenz

Die Daten werden konsistent versendet.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

X_ABORT: Bestehende Verbindung zu Kommunikationspartner außerhalb der eigenen S7-Station abbrechen

Beschreibung

Mit der Anweisung "X_ABORT" brechen Sie die Verbindung zu einem außerhalb der eigenen S7-Station liegenden Kommunikationspartner ab, die mit einer der Anweisungen "X_SEND (Seite 7077)", "X_GET (Seite 7083)" oder "X_PUT (Seite 7085)" aufgebaut wurde.

- Falls der zu "X_SEND (Seite 7077)", "X_GET (Seite 7083)" oder "X_PUT (Seite 7085)" gehörige Auftrag abgeschlossen ist (BUSY=0), sind nach dem Aufruf der Anweisung "X_ABORT" die dafür belegten Verbindungsressourcen auf beiden Seiten wieder freigegeben.
- Falls der zu "X_SEND (Seite 7077)", "X_GET (Seite 7083)" oder "X_PUT (Seite 7085)" gehörige Auftrag nicht abgeschlossen ist (BUSY=1), müssen Sie nach Abschluss des Verbindungsabbruchs die zugehörige Anweisung erneut mit REQ=0 und CONT=0 aufrufen und BUSY=0 abwarten. Erst dann sind alle belegten Verbindungsressourcen wieder freigegeben.

Sie können "X_ABORT" nur auf derjenigen Seite aufrufen, auf der "X_SEND (Seite 7077)", "X_PUT (Seite 7085)" oder "X_GET (Seite 7083)" abläuft. Die Aktivierung des Verbindungsabbruchs erfolgt nach Aufruf der Anweisung mit REQ=1.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "X_ABORT":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter "request to activate" Siehe auch: AUTOHOTSPOT
DEST_ID	Input	WORD	E, A, M, D, L oder Konstante	Adressierungsparameter "destination ID". Er enthält die MPI-Adresse des Kommunikationspartners. Diese haben Sie projektiert.
RET_VAL	Return	INT	E, A, M, D, L	Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert den zugehörigen Fehlercode.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Verbindungsabbruch ist noch nicht abgeschlossen. • BUSY=0: Der Verbindungsabbruch ist abgeschlossen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Betriebszustandsübergang nach STOP

Wechselt die CPU in den Betriebszustand STOP, wird ein über "X_ABORT" angestoßener Verbindungsabbruch zu Ende bearbeitet.

Betriebszustandsübergang des Kommunikationspartners nach STOP

Ein Wechsel der CPU des Kommunikationspartners in den Betriebszustand STOP hat für den Verbindungsabbruch mit "X_ABORT" keine Auswirkung: Die Verbindung wird abgebrochen.

Parameter RET_VAL

Siehe auch:

- AUTOHOTSPOT
- Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 171)

Siehe auch

Unterschied zwischen synchron und asynchron arbeitenden Anweisungen (Seite 4370)

4.2.5.11 TeleService

PG_DIAL: Fernverbindung zu PG/PC aufbauen

Beschreibung PG_DIAL

Beschreibung

Die Anweisung "PG_DIAL" überträgt eine Telefonnummer und eine Event-ID zu einem TS Adapter. Der TS Adapter baut mit Hilfe der angegebenen Telefonnummer eine Fernverbindung zu einem PG/PC auf. Die Event-ID wird zum PG/PC übertragen und einer wartenden Anwendung übergeben.

Wurde die Event-ID (Parameter EVENT_ID (Seite 7091)) der Anwendung erfolgreich übergeben, empfängt der TS Adapter eine Quittung, die an die Anweisung "PG_DIAL" weitergeleitet wird. Daraufhin wird die Bearbeitung von "PG_DIAL" beendet und dem Aufrufer von "PG_DIAL" der Status übermittelt. Die Anwendung auf dem PG/PC ist dafür verantwortlich, die Fernverbindung zu trennen.

Tritt während der Bearbeitung ein Fehler auf, der zum Abbruch der Bearbeitung führt, wird der Fehlercode dem Aufrufer von "PG_DIAL" übergeben. Eine eventuell aufgebaute Fernverbindung wird vom TS Adapter beendet.

Aufruf von "PG_DIAL"

Sie können die Anweisung "PG_DIAL" im Zyklus oder in einem zeitgesteuerten Programm aufrufen. Beim Aufruf von "PG_DIAL" ist die Angabe eines Instanz-DBs zwingend erforderlich.

Zur Bearbeitung einer Bausteinfunktion sind mehrere Aufrufe von "PG_DIAL" erforderlich. Der Aufruf von "PG_DIAL" in einer "Warteschleife" ist deswegen nicht sinnvoll. Das Ende der Bausteinfunktion wird mit BUSY = 0 angezeigt.

Abbruch der Kommunikationsverbindung

Wechselt die CPU in den Betriebszustand STOP, während "PG_DIAL" aktiv ist, wird die Kommunikationsverbindung zum TS Adapter abgebrochen.

Die Kommunikationsverbindung zum TS Adapter geht auch verloren, wenn am MPI-Bus schwere Kommunikationsprobleme auftreten oder beispielsweise die CPU ausgeschaltet wird. In diesen Fällen werden die Telefonnummer und Event-ID, die bereits zum TS Adapter übertragen wurden, von diesem nicht verworfen. Der TS Adapter baut die Fernverbindung zum PG/PC auf und übergibt die Event-ID. Die Quittung, die vom PG/PC zum TS Adapter übertragen wird, wird jedoch vom TS Adapter verworfen.

Versucht ein Anwenderprogramm, eine Fernverbindung aufzubauen, während sich der TS Adapter in dem oben beschriebenen Zustand befindet, so wird "PG_DIAL" mit dem Rückgabewert W#16#B10A beendet. Das Anwenderprogramm kann den Aufbau einer Fernverbindung dann zu einem späteren Zeitpunkt wiederholen. Den Rückgabewert W#16#B10A erhalten Sie auch, wenn verschiedene CPUs gleichzeitig versuchen, über denselben TS Adapter eine Fernverbindung aufzubauen.

ACHTUNG
Ändern von Anwenderprogrammen Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf Aufrufe von "PG_DIAL" auswirken, nur im Betriebszustand STOP der CPU ändern. Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "PG_DIAL" enthalten. Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen geraten. Nach der Übertragung der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

Datenkonsistenz

Die Eingangsparameter des Funktionsbausteins werden beim ersten Aufruf von "PG_DIAL" in einen internen Puffer kopiert. Ändern Sie diese Daten nicht, bevor der erste Aufruf abgeschlossen ist (Rückgabewert W#16#7001), da sonst inkonsistente Daten übertragen werden können.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PG_DIAL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MPI_TS_ADAPTER	Input	INT	E, A, M, D, L oder Konstante	MPI-Adresse des TS Adapters
PHONE_NO (Seite 7090)	Input	ANY	D	Referenz auf einen Daten-String mit einer maximalen Länge von 31 Zeichen
EVENT_ID (Seite 7091)	Input	ANY	E, A, M, D	Referenz auf ein Byte-Array mit einer maximalen Länge von 16 Zeichen
BUSY (Seite 7091)	Output	BOOL	E, A, M, D, L	• BUSY=1: Der Aufbau der Fernverbindung ist noch nicht abgeschlossen. • BUSY=0: Die Bearbeitung von "PG_DIAL" wurde beendet.
STATUS (Seite 7091)	Output	INT	E, A, M, D, L	Rückgabewert von "PG_DIAL"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Virtuelle Bausteinparameter EN und ENO und das BIE-Bit

Die virtuellen Bausteinparameter EN und ENO treten nur bei der Einbindung von "PG_DIAL" in der KOP oder FUP Darstellung des STEP 7-Editors auf. Sie sind eng mit dem BIE-Bit (Binärergebnis) des Statusworts verknüpft.

- Eingangsparameter EN
  Der Zustand des Verknüpfungsergebnisses (VKE) wird während dem Bausteinaufruf im BIE-Bit gespeichert.
- Ausgangsparameter ENO
  Bei fehlerfreier Bearbeitung von "PG_DIAL" wird der Zustand vom BIE-Bit, der beim Bausteinaufruf vorlag, am Ende des Bausteinaufrufs wieder hergestellt.
  Bei der Ausgabe einer Fehlermeldung über den Ausgangsparameter STATUS wird das BIE-Bit unmittelbar vor dem Verlassen von "PG_DIAL" auf "0" gesetzt.

Parameter PHONE_NO

Beschreibung

Der Eingangsparameter PHONE_NO gibt die Telefonnummer an, zu der eine Fernverbindung aufgebaut werden soll. Geben Sie die gesamte Telefonnummer einschließlich Landeskennzahl, Vorwahl- und Anschlussnummer an. Der String wird ohne Änderungen an das Modem übergeben. Wenn Sie Zeichen verwenden, die keine Ziffern sind, müssen Sie sich vergewissern, dass das eingesetzte Modem diese Zeichen unterstützt.

Zum Aufbauen einer Fernverbindung verwendet der TS Adapter folgende parametrierte Werte:

- Standort: Wahlverfahren, Amtskennzahl
- Rufeinstellungen: "Vor dem Wählen auf Freizeichen warten", "Anzahl der Wahlwiederholungen" und "Wartezeit nach erfolglosem Wahlversuch"

Parameter EVENT_ID

Beschreibung

Der Eingangsparameter EVENT_ID gibt die Ereignis-ID an. Diese wird mittels "PG_DIAL", den TS Adapter und die Anwendung TeleService auf dem PG/PC transparent vom Anwenderprogramm des Automatisierungssystems zum Anwenderprogramm auf dem PG/PC übertragen. Sie können die Event-ID beliebig strukturieren und beliebige Informationen vom Automatisierungssystem zum PG/PC übermitteln.

Wird beim Aufruf von "PG_DIAL" eine Länge kleiner als 16 übergeben, werden die restlichen Bytes des Arrays mit B#16#00 aufgefüllt.

Parameter BUSY

Beschreibung

Die Anweisung "PG_DIAL" ist eine asynchron laufende Anweisung, d.h. die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Ist der Ausgangsparameter BUSY = 1, wird im Ausgangsparameter STATUS (Seite 7091) der interne Status von "PG_DIAL" ausgegeben.
- Ist die Bearbeitung abgeschlossen, wird dies mit dem Ausgangsparameter BUSY = 0 mitgeteilt. Der Ausgangsparameter STATUS (Seite 7091) gibt dann an, ob der Auftrag fehlerfrei (STATUS = W#16#0000) oder mit Fehler abgeschlossen wurde.

Parameter STATUS

Beschreibung

Die Rückgabewerte von "PG_DIAL" können folgendermaßen klassifiziert werden:

- W#16#0000: "PG_DIAL" wurde erfolgreich beendet
- W#16#7xxx: Status von "PG_DIAL"
- W#16#8xxx: beim internen Aufruf eines Kommunikations-Anweisung oder der Anweisung "BLKMOV" wurde ein Fehler gemeldet
- W#16#9xxx: Parameterfehler beim Aufruf von "PG_DIAL"
- W#16#Bxxx: vom TS Adapter wurde ein Fehler gemeldet

Die folgende Tabelle zeigt die Rückgabewerte von "PG_DIAL" mit Ausnahme der Fehlercodes der eingesetzten Kommunikations-Anweisungen.

Rückgabewert (W#16#...)	Erläuterung	Hinweise
0000	Die Bearbeitung von "PG_DIAL" wurde fehlerfrei abgeschlossen.	
7000	"PG_DIAL" wurde zurückgesetzt (die Kommunikation zum TS Adapter wurde zwischenzeitlich abgebaut).	Rufen Sie "PG_DIAL" erneut auf.
7001	"PG_DIAL" ist aktiv (Erstaufruf, BUSY = 1). Die Funktion wurde gerade gestartet.	
7002	"PG_DIAL" ist aktiv (Folgeaufruf, BUSY = 1). Die Bearbeitung der Funktion ist noch nicht abgeschlossen.	
8xxx bzw. 8zxx	Die Bearbeitung von "PG_DIAL" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen abgeschlossen. Wenn die Fehlermeldung von der Anweisung "BLKMOV" stammt, dann bedeutet • z = 2: Fehler beim Kopieren des Parameters PHONE_NO in den internen Puffer • z = 3: Fehler beim Kopieren des Parameters EVENT_ID in den internen Puffer	Weitere Informationen finden Sie bei den Fehlerinformationen der Kommunikationsanweisungen: AUTOHOTSPOT
9001	Länge von PHONE_NO = 0 oder > 31	Die Telefonnummer muss mind. 1 Zeichen und max. 31 Zeichen umfassen.
9002	Länge von EVENT_ID = 0 oder > 16	Die Event-ID muss mind. 1 Zeichen und max. 16 Zeichen umfassen.
B000	Die Fernverbindung zum PG/PC konnte aufgebaut werden. Bei TeleService hat sich jedoch kein Programm für die übergebene Event-ID angemeldet.	
B10A	Im TS Adapter ist bereits ein Auftrag hinterlegt, der vom TS Adapter noch bearbeitet wird.	Rufen Sie "PG_DIAL" erneut auf.
B10B	Nach dem Aufbau der Fernverbindung wurde die Verbindung vom Partner abgebaut, bevor die Event-ID übergeben werden konnte.	Rufen Sie "PG_DIAL" erneut auf.
B206	Die Fernverbindung zum PG/PC konnte nicht aufgebaut werden.	Überprüfen Sie die Parametrierung des Modems im TS Adapter (lokal und entfernt).
B302	Der TS Adapter ist direkt mit einem PG/PC verbunden (Direktanschluss).	Beenden Sie die Kommunikation im Direktanschluss.
B305	Es ist noch eine Fernverbindung zu einem PG/PC aufgebaut.	Rufen Sie "PG_DIAL" zu einem späteren Zeitpunkt erneut auf.

AS_DIAL: Fernverbindung zur AS aufbauen

Beschreibung AS_DIAL

Definition

Lokales und entferntes Automatisierungssystem

- Lokal: Das S7-Automatisierungssystem, von dem die Initiative zum Aufbau der Fernverbindung ausgeht, wird als lokal bezeichnet.
- Entfernt: Das S7-Automatisierungssystem, zu dem die Fernverbindung aufgebaut werden soll, wird als entfernt bezeichnet.

Beschreibung

Mit der Anweisung "AS_DIAL" können Sie eine Fernverbindung von einem lokalen S7-Automatisierungssystem zu einem entfernten S7-Automatisierungssystem aufbauen und anschließend Prozessdaten austauschen. Die Prozessdaten können Sie mit den Anweisungen "X_GET (Seite 7083)", "X_PUT (Seite 7085)" und "X_SEND (Seite 7077)" austauschen.

Die Anweisung "AS_DIAL" verfügt über zwei Funktionen:

- **Funktion DIAL:** Aufbau einer Fernverbindung zu einem entfernten TS Adapter. Diese Funktion fordern Sie mit dem Eingangsparameter REQ_DIAL (Seite 7096) an.
- **Funktion HANGUP:** Beenden einer bestehenden Fernverbindung zu einem entfernten TS Adapter. Diese Funktion fordern Sie mit dem Eingangsparameter REQ_HANGUP (Seite 7096) an.

Es kann immer nur eine der Funktionen aktiv sein. Durch Anfordern der Funktion HANGUP wird die Funktion DIAL abgebrochen.

Tritt während der Bearbeitung ein Fehler auf, der zum Abbruch der Bearbeitung führt, wird der Fehlercode dem Aufrufer von "AS_DIAL" übergeben. Eine eventuell aufgebaute Fernverbindung wird vom TS Adapter beendet.

Aufruf von "AS_DIAL"

Sie können "AS_DIAL" im Zyklus oder in einem zeitgesteuerten Programm aufrufen. Wenn Sie "AS_DIAL" in unterschiedlichen Ablaufebenen aufrufen, dann müssen Sie für eine gegenseitige Aufrufverriegelung sorgen.

- Beim Aufruf von "AS_DIAL" ist die Angabe eines Instanz-DBs zwingend erforderlich. Für eine Fernverbindung müssen Sie immer die gleiche Instanz verwenden.
  Ausnahme: Wenn die lokale CPU über mehrere lokale TS Adapter kommuniziert, dann müssen Sie jedem TS Adapter eine eigene Instanz zuordnen.
- Zur Bearbeitung einer Bausteinfunktion sind mehrere Aufrufe von "AS_DIAL" erforderlich. Der Aufruf von "AS_DIAL" in einer "Warteschleife" ist deswegen nicht sinnvoll. Das Ende der Bausteinfunktion wird mit BUSY = 0 angezeigt.
- Zur Kommunikation mit dem TS Adapter benutzt "AS_DIAL" ebenfalls Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen.
  Stellen sie daher im lokalen Anwenderprogramm sicher, dass während des Auf- oder Abbaus einer Fernverbindung durch "AS_DIAL" keine Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen ("X_PUT (Seite 7085)", "X_GET (Seite 7083)", "X_SEND (Seite 7077)", "X_RCV (Seite 7078)", "X_ABORT (Seite 7087)") aktiv ist, der mit DEST_ID = "MPI-Adresse des lokalen TS Adapters" gestartet wurde.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AS_DIAL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ADDR_TS_ ADAPTER	Input	INT	E, A, M, D, L oder Konstante	Eingangsparameter der Funktion DIAL und HANGUP: MPI-Adresse des TS Adapters am lokalen MPI-Netz
PHONE_NO (Seite 7095)	Input	ANY	D	Eingangsparameter der Funktion DIAL: Telefonnummer des Modems vom entfernten Automatisierungssystem (max. 31 Zeichen)
LOGIN	Input	ANY	D	Eingangsparameter der Funktion DIAL: Benutzername für die Legitimation beim entfernten TS Adapter (max. 8 Zeichen)
PASSWORD	Input	ANY	D	Eingangsparameter der Funktion DIAL: Kennwort für die Legitimation beim entfernten TS Adapter (max. 8 Zeichen)
ADDR_CPU	Input	INT	E, A, M, D, L oder Konstante	Eingangsparameter der Funktion DIAL: MPI-Adresse der CPU am entfernten MPI-Netz, zu der die Verbindung hergestellt werden soll
REQ_DIAL (Seite 7096)	Input	BOOL	E, A, M, D	Eingangsparameter zur Anforderung der Funktion DIAL: Aufbau einer Fernverbindung zu einem entfernten TS Adapter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ_HANGUP (Seite 7096)	Input	BOOL	E, A, M, D	Eingangsparameter zur Anforderung der Funktion HANGUP: Abbau einer bestehenden Fernverbindung zu einem entfernten TS Adapter
STATUS (Seite 7097)	Output	INT	E, A, M, D, L	Rückgabewert von "AS_DIAL"
BUSY (Seite 7097)	Output	BOOL	E, A, M, D, L	• BUSY = 1: Die Funktion DIAL oder HANGUP ist noch aktiv, d. h. der Aufbau/Abbau der Fernverbindung ist noch nicht abgeschlossen. • BUSY = 0: Bearbeitung der Anweisung "AS_DIAL" ist beendet. Es ist keine Funktion mehr aktiv.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Virtuelle Bausteinparameter EN und ENO und das BIE-Bit

Die virtuellen Bausteinparameter EN und ENO treten nur bei der Einbindung von "AS_DIAL" in der KOP- oder FUP-Darstellung auf.

- Eingangsparameter EN: Der Zustand des Verknüpfungsergebnisses (VKE) wird während dem Bausteinaufruf im BIE-Bit gespeichert.
- Ausgangsparameter ENO:
  Bei fehlerfreier Bearbeitung von "AS_DIAL" wird der Zustand vom BIE-Bit, der beim Bausteinaufruf vorlag, am Ende des Bausteinaufrufs wieder hergestellt.
  Bei der Ausgabe einer Fehlermeldung über den Ausgangsparameter STATUS (Seite 7097) wird das BIE-Bit unmittelbar vor dem Verlassen von "AS_DIAL" auf "0" gesetzt.

Parameter PHONE_NO

Beschreibung

Der Eingangsparameter PHONE_NO gibt die Telefonnummer an, zu der eine Fernverbindung aufgebaut werden soll. Geben Sie die gesamte Telefonnummer einschließlich Landeskennzahl, Vorwahl- und Anschlussnummer an. Der String wird ohne Änderungen an das Modem übergeben. Wenn Sie Zeichen verwenden, die keine Ziffern sind, müssen Sie sich vergewissern, dass das eingesetzte Modem diese Zeichen unterstützt.

Zum Aufbauen einer Fernverbindung verwendet der TS Adapter folgende parametrierte Werte:

- Standort: Wahlverfahren, Amtskennzahl
- Rufeinstellungen: "Vor dem Wählen auf Freizeichen warten", "Anzahl der Wahlwiederholungen" und "Wartezeit nach erfolglosem Wahlversuch"

Parameter REQ_DIAL

Beschreibung

Die Funktion DIAL kann nur dann angefordert werden, wenn beim letzten Aufruf von "AS_DIAL" BUSY (Seite 7097) = 0 gemeldet wurde. Mit dem Eingangsparameter REQ_DIAL starten Sie die Funktion DIAL. Der Eingangsparameter REQ_DIAL ist pegelgetriggert.

REQ_DIAL stößt die Funktion DIAL der Anweisung "AS_DIAL" an. Die Anweisung "AS_DIAL" überprüft die Parameter auf Plausibilität und überträgt diese zum lokalen TS Adapter.

Wenn während der Ausführung der Funktion DIAL ein Fehler festgestellt wird, dann führt dies zum Abbruch der Bearbeitung der Funktion DIAL. "AS_DIAL" führt automatisch die Funktion HANGUP (Seite 7096) aus und gibt anschließend die Fehlermeldung im Rückgabewert STATUS (Seite 7097) aus.

Ergebnis: Mit BUSY (Seite 7097) = 0 wird Ihnen im Rückgabewert STATUS (Seite 7097) das Ergebnis des Verbindungsaufbaus angezeigt.

Parameter REQ_HANGUP

Beschreibung

Mit dem Eingangsparameter REQ_HANGUP starten Sie die Funktion HANGUP. Die Funktion HANGUP beendet eine Fernverbindung zum entfernten TS Adapter, die vorher durch die Funktion DIAL (Seite 7096) aufgebaut wurde.

Der Eingangsparameter REQ_HANGUP ist pegelgetriggert. Die Funktion HANGUP kann auch dann gestartet werden, wenn bereits die Funktion DIAL (Seite 7096) gestartet wurde.

Das Ende der Funktion HANGUP wird Ihnen über den Ausgangsparameter BUSY (Seite 7097) = 0 mitgeteilt. Der Ausgangsparameter STATUS (Seite 7097) liefert dann das Ergebnis der Funktionsbearbeitung.

Wert	Bedeutung
W#16#0000	Die Fernverbindung wurde durch die Funktion HANGUP regulär beendet.
W#16#3007	Die Fernverbindung wurde vom lokalen TS Adapter beendet, weil ein Fehler aufgetreten ist.
W#16#3008	Die Fernverbindung ist spontan zusammengebrochen oder der entfernte TS Adapter hat aufgelegt.

Parameter BUSY

Beschreibung

Die Anweisung "AS_DIAL" ist eine asynchron laufende Anweisung, das heißt, die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Ist BUSY = 1, dann wird im Ausgangsparameter STATUS (Seite 7097) der interne STATUS (Seite 7097) von "AS_DIAL" ausgegeben. Es ist mindestens ein weiterer Aufruf von "AS_DIAL" erforderlich.
- Ist BUSY = 0, dann ist die Bearbeitung abgeschlossen. Der Ausgangsparameter STATUS (Seite 7097) gibt dann an, ob der Auftrag fehlerfrei oder mit Fehler abgeschlossen wurde.

Parameter STATUS

Beschreibung

Die Rückgabewerte von "AS_DIAL" können folgendermaßen klassifiziert werden:

- W#16#0000: "AS_DIAL" wurde ohne Fehler beendet
- W#16#3xxx: letzter Zustand der Fernverbindung im TS Adapter
- W#16#7xxx: Status von "AS_DIAL"
- W#16#8xxx: beim internen Aufruf einer Kommunikations-Anweisung ("X_ABORT (Seite 7087)", "X_GET (Seite 7083)", "X_PUT (Seite 7085)", "X_RCV (Seite 7078)" und "X_SEND (Seite 7077)") wurde ein Fehler gemeldet.
- W#16#9xxx: Parameterfehler beim Aufruf von "AS_DIAL"
- W#16#Bxxx: Fehler beim Verbindungsaufbau

Die folgende Tabelle zeigt die Rückgabewerte von "AS_DIAL" mit Ausnahme der Fehlercodes der eingesetzten Kommunikations-Anweisungen:

Rückgabewert (W#16#...)	Erläuterung	Hinweise
0000	Es ist keine Funktion aktiv. Die Bearbeitung von "AS_DIAL" wurde fehlerfrei abgeschlossen.	Diese Meldung erhalten Sie nur ein einziges Mal nach erfolgreichem Abschluss der jeweiligen Funktion.
3007	Der lokale TS Adapter hat aufgelegt.	Wird nur von der Funktion HANGUP gezeigt.
3008	Die Fernverbindung ist spontan zusammengebrochen oder der entfernte TS Adapter hat aufgelegt.	Wird nur von der Funktion HANGUP gezeigt.
7000	Es ist keine Funktion aktiv.	Es wurde weder eine Funktion bearbeitet oder über die Eingangsparameter angefordert.
7001	Funktion DIAL ist aktiv (Erst-Aufruf, BUSY =1). Die Funktion wurde gerade gestartet.	Bis zum Ende der Funktion (BUSY = 0) erfolgt keine Auswertung des Eingangsparameters REQ_DIAL.

Rückgabewert (W#16#...)	Erläuterung	Hinweise
7002	Funktion DIAL ist aktiv (Folge-Aufruf, BUSY = 1). Die Bearbeitung der Funktion ist noch nicht abgeschlossen.	Bis zum Ende der Funktion (BUSY = 0) erfolgt keine Auswertung des Eingangsparameters REQ_DIAL.
7701	Funktion HANGUP ist aktiv (Erst-Aufruf). Die Funktion wurde gerade gestartet.	Bis zum Ende der Funktion (BUSY = 0) erfolgt keine Auswertung der Eingangsparameter REQ_DIAL und REQ_HANGUP.
7702	Funktion HANGUP ist aktiv (Folge-Aufruf). Die Bearbeitung der Funktion ist noch nicht abgeschlossen.	Bis zum Ende der Funktion (BUSY = 0) erfolgt keine Auswertung der Eingangsparameter REQ_DIAL und REQ_HANGUP.
8xxx	Fehlermeldung: Die Bearbeitung der aktiven Bausteinfunktion wurde abgebrochen wegen Fehlermeldung (RET_VAL < 0) einer bausteinintern aufgerufenen Anweisung.	Weitere Informationen finden Sie bei den Fehlerinformationen der Kommunikationsanweisungen: AUTOHOTSPOT.
9001	Länge von PHONE_NO = 0 oder > 31	Die Telefonnummer muss mind. 1 Zeichen und max. 31 Zeichen umfassen.
9003	Länge von LOGIN > 8	Der Benutzername für die Legitimation beim entfernten TS Adapter darf max. 8 Zeichen betragen.
9004	Länge von PASSWORD > 8	Das Kennwort für die Legitimation beim entfernten TS Adapter darf max. 8 Zeichen betragen.
B100	TS Adapter kann zur Zeit keine Fernverbindung aufbauen.	Der TS Adapter kann momentan keine Verbindung zum Modem aufrufen. Rufen Sie "AS_DIAL" erneut auf.
B10A	Im TS Adapter ist bereits eine Event-ID einer "PG_DIAL"-Funktion hinterlegt oder es ist eine "AS_DIAL"-Funktion aktiv, die noch übertragen werden soll.	Rufen Sie "AS_DIAL" erneut auf.
B10B	Nach dem erfolgreichen Aufbau der Fernverbindung wurde die Fernverbindung wieder getrennt.	Rufen Sie "AS_DIAL" erneut auf.
B206	Die Fernverbindung zum entfernten TS Adapter konnte nicht aufgebaut werden.	Überprüfen Sie die Parametrierung des Modems im TS Adapter (lokal und entfernt).
B20A	Der Typ des entfernten TS Adapters unterstützt keine AS-AS-Fernkopplung.	Verwenden Sie einen TS Adapter, der die AS-AS-Fernkopplung unterstützt.
B20B	Die Anweisung "AS_DIAL" und die Version des TS Adapters sind inkompatibel.	Verwenden Sie einen TS Adapter, der die AS-AS-Fernkopplung unterstützt.
B20C	Der entfernte TS Adapter wird nicht in das MPI-Netz aufgenommen. Ursache: Netz-Parameterfehler	Überprüfen Sie die Parametrierung der Netzparameter im entfernten TS Adapter.
B20D	Die Version des entfernten TS Adapters unterstützt keine AS-AS-Fernkopplung.	Verwenden Sie einen TS Adapter, der die AS-AS-Fernkopplung unterstützt.
B20E	Die Datenübertragung zum entfernten TS Adapter wurde abgebrochen. Ursache: Übertragungsfehler	Rufen Sie "AS_DIAL" erneut auf.
B20F	Der entfernte TS Adapter hat nicht zurückgerufen.	Überprüfen Sie die Parametrierung des Zugriffsschutzes (Rückrufnummer) und Modems im entfernten TS Adapter.

Rückgabewert (W#16#...)	Erläuterung	Hinweise
B210	Die Datenübertragung zum entfernten TS Adapter ist gestört. Ursache: Timeout	Rufen Sie "AS_DIAL" erneut auf.
B252	Der entfernte TS Adapter lehnt die Auftragsbearbeitung ab, da die Legitimation fehlt.	Sie haben keinen Benutzernamen und Kennwort beim Aufruf von "AS_DIAL" angegeben.
B253	Der entfernte TS Adapter lehnt die Legitimation ab. Ursache: unbekannter Benutzername	Überprüfen Sie die Aufrufparameter von "AS_DIAL" in Ihrem Anwenderprogramm.
B254	Der entfernte TS Adapter lehnt die Legitimation ab. Ursache: falsches Kennwort	Überprüfen Sie die Aufrufparameter von "AS_DIAL" in Ihrem Anwenderprogramm.
B302	Der TS Adapter ist direkt mit einem PG/PC verbunden (Direktanschluss).	Beenden Sie die Kommunikation im Direktanschluss.
B305	Vom lokalen TS Adapter ist noch eine Fernverbindung zu einem PG/PC aufgebaut.	Rufen Sie "AS_DIAL" zu einem späteren Zeitpunkt erneut auf.

SMS_SEND: SMS Nachricht senden

Beschreibung SMS_SEND

Beschreibung

Die Anweisung "SMS_SEND" überträgt eine Telefonnummer, eine Servicecenternummer und eine SMS Nachricht zu einem TS Adapter. Der TS Adapter übergibt diese Daten mittels GSM-Kommandos an ein Funkmodem.

Nach dem Senden der SMS Nachricht empfängt der TS Adapter eine Quittung, die an die Anweisung "SMS_SEND" weitergeleitet wird. Daraufhin wird die Bearbeitung von "SMS_SEND" beendet und dem Aufrufer der Status übermittelt. Dieser Status ist nur eine Quittung über das Senden der SMS, keine Empfangsbestätigung.

Tritt während der Bearbeitung ein Fehler auf, der zum Abbruch der Bearbeitung führt, wird der Fehlercode dem Aufrufer von "SMS_SEND" übergeben.

Aufruf von "SMS_SEND"

Sie können die Anweisung "SMS_SEND" im Zyklus oder in einem zeitgesteuerten Programm aufrufen.

Beim Aufruf von "SMS_SEND" ist die Angabe eines Instanz-DBs zwingend erforderlich.

Zur Bearbeitung einer Bausteinfunktion sind mehrere Aufrufe von "SMS_SEND" erforderlich. Der Aufruf von "SMS_SEND" in einer "Warteschleife" ist deswegen nicht sinnvoll. Das Ende der Bausteinfunktion wird mit BUSY = 0 angezeigt.

Abbruch der Kommunikationsverbindung

Wechselt die CPU in den Betriebszustand STOP, während "SMS_SEND" aktiv ist, wird die Kommunikationsverbindung zum TS Adapter abgebrochen. Die Kommunikationsverbindung zum TS Adapter geht auch verloren, wenn am MPI-Bus schwere Kommunikationsprobleme auftreten oder beispielsweise die CPU ausgeschaltet wird.

In diesen Fällen werden die Telefonnummer, Servicecenternummer und SMS Nachricht, die bereits zum TS Adapter übertragen wurden, von diesem nicht verworfen. Der TS Adapter übergibt die Daten an das Funkmodem. Die Quittung, die vom Funkmodem zum TS Adapter übertragen wird, wird jedoch vom TS Adapter verworfen.

Versucht ein Anwenderprogramm auf der CPU eine SMS zu senden, während sich der TS Adapter in dem oben beschriebenen Zustand befindet, so wird "SMS_SEND" mit dem Rückgabewert W#16#B10A beendet. Das Anwenderprogramm kann das Senden einer SMS dann zu einem späteren Zeitpunkt wiederholen. Den Rückgabewert W#16#B10A erhalten Sie auch, wenn verschiedene CPUs gleichzeitig versuchen, über denselben TS Adapter eine SMS zu senden.

ACHTUNG
Ändern des STEP 7-Anwenderprogramms Sie dürfen die Teile Ihres STEP 7-Anwenderprogramms, die sich unmittelbar auf Aufrufe von "SMS_SEND" auswirken, nur im Betriebszustand STOP der CPU ändern. Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "SMS_SEND" enthalten. Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen ("X_PUT (Seite 7085)", "X_GET (Seite 7083)", "X_SEND (Seite 7077)", "X_RCV (Seite 7078)", "X_ABORT (Seite 7087)") geraten. Nach der Übertragung der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

Datenkonsistenz

Die Eingangsparameter des Funktionsbausteins werden beim ersten Aufruf von "SMS_SEND" in einen internen Puffer kopiert. Ändern Sie diese Daten nicht, bevor der erste Aufruf abgeschlossen ist (Rückgabewert W#16#7001), da sonst inkonsistente Daten übertragen werden können.

Übergabe der PIN an das Funkmodem

Wenn Sie eine SIM-Karte verwenden, bei der die PIN-Überprüfung eingeschaltet ist, dann muss die PIN in der Initialisierungsphase dem Funkmodem übergeben werden. Dies ist möglich über den Initstring des TS Adapters, den Sie mittels TeleService parametrieren können. Parametrieren Sie den Initstring wie folgt (Beispiel für PIN = 4711):

- AT+CPIN="4711";AT&F.....

Der TS Adapter sendet nach Spannungswiederkehr einmalig den String "AT+CPIN="4711" mit der PIN für die im Funkmodem eingesetzte SIM-Karte zu der angeschlossenen Komponente. Anschließend wird das Funkmodem mit "AT&F..." initialisiert.

Achtung: Eine fehlerhafte PIN wird nicht bei der Initialisierung gemeldet, sondern erst beim Senden einer SMS über die Rückmeldung von "SMS_SEND".

Senden eines Fax

Durch Voranstellen einer speziellen Nummer des Netzanbieters (z. B. "99") vor die eigentliche Nummer wird die SMS vom Netzanbieter in ein Fax umgewandelt und an ein Faxgerät gesendet. Dies ist eine Leistung des Netzanbieters, keine Funktion des TS Adapters.

Die Fax-Nummer wird mit dem Nummerzusatz des Netzanbieters am Parameter PHONE_NO eingeben.

Beispiel für ein Netz mit der Faxnummer 07214711:

- PHONE_NO = '9907214711'

Senden einer E-Mail

Durch Wahl einer speziellen Nummer des Netzanbieters (z. B. "8000") und Voranstellen einer E-Mail-Adresse vor die SMS-Nachricht kann die SMS an eine E-Mail-Adresse gesendet werden. Dies ist eine Leistung des Netzanbieters, keine Funktion des TS Adapters.

Die entsprechende Nummer übergeben Sie am Parameter PHONE_NO (Seite 7102).

Die E-Mail-Adresse wird in der SMS-Nachricht dem eigentlichen Nachrichtentext vorangestellt und durch ein Trennzeichen von diesem getrennt. Die maximale Länge des SMS-Nutztextes verkürzt sich dadurch um die E-Mail-Adresse + Trennzeichen. Trennzeichen sind z. B. Leerzeichen (" ") oder Doppelpunkte (":").

Statt des Zeichens "@" wird von manchen Netzanbietern ein "*" verlangt.

Beispiel:

- Gesendet werden soll der Text 'CPU in STOP' an die E-Mail-Adresse "Surname*provider.com".
- PHONE_NO = 8000
- MESSAGE = Surname*provider.comCPU in STOP

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SMS_SEND":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
MPI_TS_ ADAPTER	Input	INT	E, A, M, D, L oder Konstante	MPI-Adresse des TS Adapters
PHONE_NO (Seite 7102)	Input	ANY	D	Referenz auf einen Daten-String mit einer maximalen Länge von 31 Zeichen
SCENTER_NO (Seite 7102)	Input	ANY	D	Referenz auf einen Daten-String mit einer maximalen Länge von 20 Zeichen
MESSAGE (Seite 7103)	Input	ANY	D	Referenz auf einen Daten-String mit einer maximalen Länge von 160 Zeichen

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BUSY (Seite 7103)	Output	BOOL	E, A, M, D, L	• BUSY=1: Das Senden der SMS ist noch nicht abgeschlossen • BUSY=0: Die Bearbeitung von "SMS_SEND" wurde beendet
STATUS (Seite 7103)	Output	INT	E, A, M, D, L	Rückgabewert von "SMS_SEND"

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Virtuelle Parameter EN und ENO und das BIE-Bit

Die virtuellen Bausteinparameter EN und ENO treten nur bei der Einbindung von "SMS_SEND" in der KOP- oder FUP-Darstellung auf.

- Eingangsparameter EN:
  Der Zustand des Verknüpfungsergebnisses (VKE) wird während des Bausteinaufrufs im BIE-Bit gespeichert.
- Ausgangsparameter ENO:
  Bei fehlerfreier Bearbeitung von "SMS_SEND" wird der Zustand vom BIE-Bit, der beim Bausteinaufruf vorlag, am Ende des Bausteinaufrufs wieder hergestellt.
  Bei der Ausgabe einer Fehlermeldung über den Ausgangsparameter STATUS wird das BIE-Bit unmittelbar vor dem Verlassen von "SMS_SEND" auf "0" gesetzt.

Parameter PHONE_NO

Beschreibung

Der Eingangsparameter PHONE_NO gibt die Telefonnummer an, zu der eine SMS gesendet werden soll. Geben Sie die gesamte Telefonnummer einschließlich Landeskennzahl, Vorwahl- und Anschlussnummer an. Der String wird ohne Änderungen an das Funkmodem übergeben. Wenn Sie Zeichen verwenden, die keine Ziffern sind, müssen Sie sich vergewissern, dass das eingesetzte Funkmodem diese Zeichen unterstützt.

Parameter SCENTER_NO

Beschreibung

Der Eingangsparameter SCENTER_NO gibt die Servicecenternummer an, über die die SMS verschickt werden soll. Der String wird ohne Änderungen an das Funkmodem übergeben.

Parameter MESSAGE

Beschreibung

Der Eingangsparameter MESSAGE gibt die SMS-Nachricht an. Diese wird mittels der Anweisung "SMS_SEND", dem TS Adapter und dem GSM-Funkmodem zum Netzwerk-Provider übertragen.

Parameter BUSY

Beschreibung

Die Anweisung "SMS_SEND" ist eine asynchron laufende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Ist der Ausgangsparameter BUSY = 1, wird im Ausgangsparameter STATUS (Seite 7103) der interne Status von "SMS_SEND" ausgegeben.
- Ist die Bearbeitung abgeschlossen, wird dies mit dem Ausgangsparameter BUSY = 0 mitgeteilt. Der Ausgangsparameter STATUS (Seite 7103) gibt dann an, ob der Auftrag fehlerfrei (STATUS = W#16#0000) oder mit Fehler abgeschlossen wurde.

Parameter STATUS

Beschreibung

Die Rückgabewerte von "SMS_SEND" können folgendermaßen klassifiziert werden:

- W#16#0000: "SMS_SEND" wurde erfolgreich beendet.
- W#16#7xxx: Status von "SMS_SEND"
- W#16#8xxx: beim internen Aufruf eines Kommunikations-Anweisungen oder der Anweisung "BLKMOV" wurde ein Fehler gemeldet.
- W#16#9xxx: Parameterfehler beim Aufruf von "SMS_SEND"
- W#16#Bxxx: vom TS Adapter wurde ein Fehler gemeldet.
- W#16#Cxxx: vom Funkmodem wurde ein Fehler gemeldet.

Die folgende Tabelle zeigt die Rückgabewerte von "SMS_SEND" mit Ausnahme der Fehlercodes der eingesetzten Kommunikations-Anweisungen sowie der Funkmodems.

Rückgabewert (W#16#...)	Erläuterung	Hinweise
0000	Die Bearbeitung von "SMS_SEND" wurde fehlerfrei abgeschlossen.	
7000	"SMS_SEND" wurde zurückgesetzt (die Kommunikation zum TS Adapter wurde zwischenzeitlich abgebaut).	Rufen Sie "SMS_SEND" erneut auf.
7001	"SMS_SEND" ist aktiv (Erst-Aufruf, BUSY = 1). Die Funktion wurde gerade gestartet.	

Rückgabewert (W#16#...)	Erläuterung	Hinweise
7002	"SMS_SEND" ist aktiv (Folgeaufruf, BUSY = 1). Die Bearbeitung der Funktion ist noch nicht abgeschlossen.	
8xxx bzw. 8zxx	Die Bearbeitung von "SMS_SEND" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen oder der Anweisung "BLKMOV" abgeschlossen. Wenn die Fehlermeldung von "BLKMOV" stammt, dann bedeutet • z = 2 : Fehler beim Kopieren des Parameters PHONE_NO in den internen Puffer • z = 3 : Fehler beim Kopieren des Parameters SCENTER_NO in den internen Puffer • z = 4 : Fehler beim Kopieren des Parameters MESSAGE in den internen Puffer	Ausführliche Informationen entnehmen Sie der Beschreibung der Anweisung "BLKMOV" sowie den Fehlerinformationen der Kommunikations-Anweisungen: AUTOHOTSPOT
9001	Länge von PHONE_NO = 0 oder > 31	Die Telefonnummer muss mind. 1 Zeichen und max. 31 Zeichen umfassen.
9002	Länge von SCENTER_NO = 0 oder > 20	Die Servicecenternummer muss mind. 1 Zeichen und max. 20 Zeichen umfassen.
9003	Länge der MESSAGE = 0 oder > 160	Die SMS-Nachricht muss mind. 1 Zeichen und darf max. 160 Zeichen umfassen.
B10A	Im TS Adapter ist bereits ein Auftrag hinterlegt, der vom TS Adapter noch bearbeitet wird.	Rufen Sie "SMS_SEND" erneut auf.
B301	Es ist noch eine Fernverbindung zu einem PG/PC aufgebaut.	Rufen Sie "SMS_SEND" zu einem späteren Zeitpunkt erneut auf.
B302	Der TS Adapter ist direkt mit einem PG/PC verbunden (Direktanschluss)	Verbinden Sie den TS Adapter mit einem GSM Funkmodem.
B303	Der TS Adapter ist weder mit einem Funkmodem, noch mit einem PG/PC verbunden.	Verbinden Sie den TS Adapter mit einem GSM Funkmodem.
B304	Die Schnittstelle zum Funkmodem ist im Moment nicht bereit, eine SMS zu verschicken.	Rufen Sie "SMS_SEND" zu einem späteren Zeitpunkt erneut auf.
B614	Das angeschlossene Modem hat innerhalb der Überwachungszeit nicht auf ein SMS-spezifisches Kommando geantwortet.	Überprüfen Sie, ob Sie ein GSM Funkmodem angeschlossen haben. Rufen Sie "SMS_SEND" zu einem späteren Zeitpunkt erneut auf.

Rückgabewert (W#16#...)	Erläuterung	Hinweise
B615	Der TS Adapter hat eine unspezifische Fehlermeldung des angeschlossenen Modems empfangen.	Überprüfen Sie, ob Sie ein GSM Funkmodem angeschlossen haben.
Cxxx	Die Bearbeitung von "SMS_SEND" wurde mit einem Fehlercode des GSM-Modems abgebrochen. xxx: Fehlernummer	Eine Liste der Fehlernummern finden Sie im Handbuch zu Ihrem Funkmodem oder auch in den GSM-Standards GSM 04.11, GSM 03.40 und GSM 07.05. Beispiele: • C136 = Sim-Karte im GSM-Modem fehlt • C137 = PIN für GSM-Modem nicht korrekt (siehe Übergabe der PIN an das Funkmodem) • C001 = Falsche Rufnummer des Service Center

AS_MAIL: E-Mail übertragen

Beschreibung AS_MAIL

Benötigte Anweisungen

Die Anweisung "AS_MAIL" überträgt mittels Simple Mail Transfer Protocol (SMTP) eine E-Mail von einer CPU zu einem Mail-Server. "AS_MAIL" benötigt zum Ablauf die folgenden Anweisungen:

- "S_COMP: Zeichenketten vergleichen (Seite 5436)"
- "FIND: Zeichen in Zeichenkette finden (Seite 5449)"
- "INSERT: Zeichen in Zeichenkette einfügen (Seite 5447)"
- "LEFT: Linke Zeichen einer Zeichenkette lesen (Seite 5444)"
- "LEN: Länge einer Zeichenkette ermitteln (Seite 5443)"
- "RIGHT: Rechte Zeichen einer Zeichenkette lesen (Seite 5445)"

Beschreibung

Die Anweisung "AS_MAIL" ist eine asynchron arbeitende Anweisung, d.h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Beim Aufruf der Anweisung "AS_MAIL" ist die Angabe eines Instanz-DBs zwingend erforderlich.

Sie initialisieren den Baustein durch einen Aufruf mit COM_RST = 1.

Sie starten das Versenden einer E-Mail mit einem Flankenwechsel von "0" auf "1" am Parameter REQ.

Über die Ausgangsparameter "BUSY", "DONE", "ERROR" und die Ausgangsparameter "STATUS" und "SFC_STATUS" wird der Zustand des Auftrags angezeigt. Dabei entspricht

"SFC_STATUS" dem Ausgangsparameter "STATUS" der aufgerufenen Kommunikations-Bausteine bzw. der Anweisung "BLKMOV".

Die Ausgangsparameter DONE, ERROR, STATUS und SFC_STATUS werden jeweils nur einen Zyklus lang angezeigt, wenn der Zustand des Ausgangsparameters BUSY von "1" auf "0" wechselt.

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung "AS_MAIL" aktuell befindet bzw. wann das Senden der E-Mail beendet ist.

BUSY	DONE	ERROR	Beschreibung
1	Irrelevant	Irrelevant	Der Auftrag ist in Bearbeitung.
0	1	0	Der Auftrag wurde erfolgreich abgeschlossen.
0	0	1	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache entnehmen Sie den Parametern STATUS und SFC_STATUS.
0	0	0	Der Anweisung "AS_MAIL" wurde kein (neuer) Auftrag erteilt.

Wechselt die CPU in den Betriebszustand STOP, während "AS_MAIL" aktiv ist, so wird die Kommunikationsverbindung zum Mail-Server abgebrochen. Die Kommunikationsverbindung zum Mail-Server geht auch verloren, wenn am Industrial Ethernet-Bus Kommunikationsprobleme auftreten. In diesen Fällen wird das Versenden der E-Mail unterbrochen und sie erreicht ihre Empfänger nicht.

ACHTUNG

Ändern von Anwenderprogrammen

Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf die Aufrufe von "AS_MAIL" auswirken, nur ändern:

- wenn die CPU im Betriebszustand "STOP" ist oder
- wenn keine Mail gesendet wird (REQ = 0 und BUSY = 0).

Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "AS_MAIL" oder Aufrufe des Instanz-DBs von "AS_MAIL" enthalten.

Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den TPC/IP-Kommunikationsfunktionen über Industrial Ethernet geraten.

Nach dem Übertragen der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen oder den Parameter COM_RST der Anweisung "AS_MAIL" setzen.

Datenkonsistenz

Der Eingangsparameter ADDR_MAIL_SERVER der Anweisung wird bei jedem Anstoß zum Versenden einer E-Mail von der Anweisung "AS_MAIL" neu übernommen. Bei einer Änderung während des Betriebs wird der "neue" Wert erst bei einem erneuten Anstoß einer E-Mail wirksam.

Die Parameter WATCH_DOG_TIME, TO_S, CC, FROM, SUB, TEXT, ATTACHMENT, sowie ggf. USERNAME und PASSWORD werden hingegen während des Betriebs der Anweisung "AS_MAIL" von diesem übernommen und dürfen somit erst geändert werden, wenn der Auftrag beendet wurde (BUSY = 0)

Parametrierung des TS Adapter IE

Am TS Adapter IE müssen Sie die Parameter für ausgehende Rufe so eingeben, dass der TS Adapter IE eine Verbindung zum Einwahlserver Ihres Internet Service Providers aufbaut.

Wenn Sie für Verbindungsaufbau "bei Bedarf" einstellen, dann wird die Verbindung erst dann aufgebaut, wenn eine Mail versendet werden soll.

Bei einer analogen Modemverbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "AS_MAIL":

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
COM_RST	Input / Output	BOOL	E, A, M, D, L	Ein-/Ausgangsparameter "COMPLETE" RESTART: Der Baustein hat eine Initialisierungsroutine, die bearbeitet wird, wenn der Eingang COM_RST gesetzt ist. Der Parameter COM_RST der Anweisung "AS_MAIL" muss auf "1" gesetzt werden: • einmalig vor dem ersten Aufruf und • immer dann, wenn der zugehörige Instanz-DB neu geladen wird. Nach der Initialisierung setzt die Anweisung "AS_MAIL" den Parameter COM_RST zurück.
REQ	Input	BOOL	E, A, M, D, L	Steuerparameter REQUEST: aktiviert das Versenden einer E-Mail bei steigender Flanke.
ADDR_MAIL_ SERVER	Input	DWORD	E, A, M, D, L	Eingangsparameter IP-Adresse des Mail-Servers: Anzugeben als Datenwort im HEX-Format, z.B.: IP-Adresse = 192.168.0.200. ADDR_MAIL_SERVER = DW#16#C0A800C8, wobei: • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 und • 200 = 16#C8 entspricht.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
WATCH_DOG_ TIME	Input	TIME	E, A, M, D, L	Eingangsparameter max. Zeitspanne: In der vorgegebenen Zeit (WATCH_DOG_TIME) sollte die Anweisung "AS_MAIL" eine Verbindung aufbauen. Wird diese Zeit überschritten, so wird der Baustein mit einem Fehler beendet. Die Zeit, bis der Baustein beendet und der Fehler ausgegeben wird, kann die WATCH_DOG_TIME überschreiten, da der Verbindungsabbau ebenso Zeit beansprucht. Zu Anfang sollten Sie eine Zeit von 2 Minuten einstellen. Bei der ISDN-Telefonverbindung kann diese Zeit deutlich kleiner gewählt werden.
USERNAME	Input	ANY	D	Eingangsparameter Benutzername: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Benutzername zwingend erforderlich.
PASSWORD	Input	ANY	D	Eingangsparameter Passwort: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Passwort zwingend erforderlich.
TO_S	Input	ANY	D	Eingangsparameter Empfängeradressen: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).
CC	Input	ANY	D	Eingangsparameter CC-Empfängeradressen: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).
FROM	Input	ANY	D	Eingangsparameter Absenderadresse: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).
SUB	Input	ANY	D	Eingangsparameter Betreff der E-Mail: Referenz auf eine Variable vom Typ STRING mit einer maximalen Länge von 240 Zeichen.
TEXT	Input	ANY	D	Eingangsparameter Text der E-Mail: Referenz auf einen Daten-String mit einer maximalen Länge von 240 Zeichen.
ATTACHMENT	Input	ANY	E, A, M, D, L	Eingangsparameter Anhang der E-Mail: Referenz auf ein Byte-/Wort-/Doppelwort-Feld mit einer maximalen Länge von 65534 Bytes.
BUSY	Output	BOOL	E, A, M, D, L	• BUSY = 1: Das Senden der E-Mail ist noch nicht abgeschlossen. • BUSY = 0: Die Bearbeitung von "AS_MAIL" wurde beendet.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
DONE	Output	BOOL	E, A, M, D, L	• DONE = 0: Auftrag ist noch nicht gestartet oder wird noch ausgeführt. • DONE = 1: Auftrag wurde fehlerfrei ausgeführt.
ERROR	Output	BOOL	E, A, M, D, L	ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS und SFC_STATUS liefern detaillierte Auskunft über die Art des Fehlers.
STATUS (Seite 7110)	Output	INT	E, A, M, D, L	Ausgangs-/Zustandsparameter STATUS: Rückgabewert bzw. Fehlerinformation der Anweisung "AS_MAIL".
SFC_STATUS (Seite 7110)	Output	INT	E, A, M, D, L	Ausgangs-/Zustandsparameter "SFC_STATUS": Fehlerinformation der aufgerufenen Kommunikations-Bausteine bzw. der Anweisung "BLKMOV".

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 243)".

Hinweis

Hinweise zu den Parametern der Anweisung

- Wird in dem Baustein, in dem "AS_MAIL" aufgerufen wird, die Zuweisung der Parameter USERNAME und PASSWORD entfernt, neu eingefügt oder geändert, so wird eine solche Änderung erst dann wirksam, wenn der zugehörige Instanz DB erneut geladen und der Parameter COM_RST gesetzt wird. Andernfalls bleibt der Zeiger auf den entsprechenden Parameter im Instanz-DB erhalten.
- Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "AS_MAIL" keine Syntaxprüfung der Parameter TO_S, CC und FROM durch. Bitte beachten Sie die korrekte Syntax der Parameter.
- Die optionalen Parameter CC, TEXT und ATTACHMENT werden nur mit der E-Mail versendet, wenn sie in dem Baustein, in dem die Anweisung "AS_MAIL" aufgerufen wird, zugewiesen werden. Wird eine solche Zuweisung aus der Aufrufschnittstelle der Anweisung "AS_MAIL" entfernt, so wird diese Änderung erst wirksam, wenn der zugehörige Instanz DB erneut geladen und der Parameter COM_RST gesetzt wird. Andernfalls bleibt der Zeiger auf den entsprechenden Parameter im Instanz-DB erhalten.
  Bitte beachten Sie, dass bei einer Zuweisung der Parameter leere Strings unzulässig sind.
- Zeigt der Parameter ATTACHMENT auf ein Array in einem Datenbaustein und wird dieses Array in seiner Größe verändert und gespeichert, so muss an der Aufrufschnittstelle der Anweisung "AS_MAIL" der ANY-Pointer des Parameters ATTACHMENT neu eingetragen werden.

SMTP-Authentisierung

Unter Authentisierung versteht man ein Verfahren zur Sicherstellung einer Identität z.B. durch eine Passwortabfrage.

Die Anweisung "AS_MAIL" unterstützt das SMTP-Authentisierungsverfahren AUTH-LOGIN, das von den meisten Mail-Servern gefordert wird. Informationen über die

Authentisierungsverfahren Ihres Mail-Servers entnehmen Sie bitte dem Handbuch des Mail-Servers oder der Website Ihres Internet Service Providers.

Um das Authentisierungsverfahren AUTH-LOGIN zu nutzen benötigt die Anweisung "AS_MAIL" den Benutzernamen, mit dem er sich beim Mail-Server anmelden kann. Dieser Benutzername entspricht dem Nutzernamen mit dem Sie ein Mailkonto auf Ihrem Mail-Server eingerichtet haben. Er wird über den Parameter USERNAME der Anweisung "AS_MAIL" bekannt gegeben.

Des Weiteren benötigt die Anweisung "AS_MAIL" zur Anmeldung das dazugehörige Passwort. Dieses Passwort entspricht dem Passwort, das Sie beim Einrichten Ihres Mailkontos vergeben haben. Es wird über den Parameter PASSWORD der Anweisung "AS_MAIL" bekannt gegeben.

Der Benutzername und das Passwort werden jeweils unverschlüsselt (BASE64-kodiert) an den Mailserver übertragen.

Wird im DB kein Benutzername angegeben, so wird das Authentisierungsverfahren AUTH-LOGIN auch nicht verwendet. Die E-Mail wird dann ohne Authentisierung versendet.

Parameter STATUS und SFC_STATUS

Beschreibung

Die Rückgabewerte der Anweisung "AS_MAIL" können folgendermaßen klassifiziert werden:

- W#16#0000: "AS_MAIL" wurde erfolgreich beendet
- W#16#7xxx: Status von "AS_MAIL"
- W#16#8xxx: beim internen Aufruf eines Kommunikations-Bausteins, beim Aufruf der Anweisung "BLKMOV" oder vom Mail-Server wurde ein Fehler gemeldet.

Die folgende Tabelle zeigt die Rückgabewerte von "AS_MAIL" mit Ausnahme der Fehlercodes der intern aufgerufenen Kommunikations-Bausteine sowie der Anweisung "BLKMOV".

Rückgabewert STATUS (W#16#...):	Rückgabewert SFC_STATUS (W#16#...):	Erläuterung	Hinweise
0000	-	Die Bearbeitung von "AS_MAIL" wurde fehlerfrei abgeschlossen.	Ein fehlerfreies Beenden von "AS_MAIL" bedeutet nicht, dass die versendete E-Mail ankommt (siehe unten - Hinweis Punkt 1)
7001		"AS_MAIL" ist aktiv (BUSY = 1).	
7002	7002	"AS_MAIL" ist aktiv (BUSY = 1).	
8xxx	xxxx	Die Bearbeitung von "AS_MAIL" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen oder der Anweisung "BLKMOV" abgeschlossen.	Ausführliche Informationen zur Auswertung des Parameters SFC_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisungen "BLKMOV".

Rückgabewert STATUS (W#16#...):	Rückgabewert SFC_STATUS (W#16#...):	Erläuterung	Hinweise
800z	xxxx	Die Fehlermeldung stammt von der Anweisung "BLKMOV" und bedeutet: • z = 1: Fehler beim Kopieren des Parameters TO_S in den internen Puffer • z = 2: Fehler beim Kopieren des Parameters CC in den internen Puffer • z = 3: Fehler beim Kopieren des Parameters FROM in den internen Puffer • z = 4: Fehler beim Kopieren des Parameters SUB in den internen Puffer • z = 5: Fehler beim Kopieren des Parameters TEXT in den internen Puffer • z = 6: Fehler beim Kopieren des Parameters ATTACHMENT in den internen Puffer • z = 7: Fehler beim Kopieren des Parameters USERNAME in den internen Puffer • z = 8: Fehler beim Kopieren des Parameters PASSWORD in den internen Puffer	
8010	xxxx	Fehler beim Verbindungsaufbau.	Möglicherweise wurde nach dem Laden des Instanz-DBs der COM_RST nicht gesetzt.
8011	xxxx	Fehler beim Senden der Daten.	Weitere Informationen zur Auswertung des SFC_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TSEND (Seite 6468)".
8012	xxxx	Fehler beim Empfangen der Daten.	Weitere Informationen zur Auswertung des SFC_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TRCV (Seite 6471)".
8013	xxxx	Fehler beim Verbindungsaufbau.	Weitere Informationen zur Auswertung des SFC_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TCON (Seite 6463)" und "TDISCON (Seite 6466)".

Rückgabewert STATUS (W#16#...):	Rückgabewert SFC_STATUS (W#16#...):	Erläuterung	Hinweise
8014	-	Aufbau einer Verbindung nicht möglich.	Möglicherweise haben Sie eine fehlerhafte Mail-Server-IP-Adresse (ADDR_MAIL_SERVER) oder eine zu kleine Zeitspanne (WATCH_DOG_TIME) für den Verbindungsaufbau eingegeben. Es besteht auch die Möglichkeit, dass die CPU keine Verbindung zum Netzwerk hat oder die CPU-Konfiguration fehlerhaft ist.
82xx, 84xx, bzw. 85xx	-	Die Fehlermeldung stammt vom Mail-Server und entspricht bis auf die "8" der Fehlernummer des SMTP-Protokolls. In den nachfolgenden Spalten sind einige Fehlercodes aufgeführt, die auftreten können:	Siehe Punkt 2 im Hinweis.
8450	-	Aktion nicht ausgeführt: Mailbox nicht verfügbar/nicht erreichbar.	Versuchen Sie es später noch einmal.
8451	-	Aktion abgebrochen: Lokaler Fehler in der Verarbeitung	Versuchen Sie es später noch einmal.
8500	-	Syntax-Fehler: Fehler nicht erkannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann auch sein, dass der E-Mail-Server das Authentisierungs-verfahren LOGIN nicht unterstützt.	Überprüfen Sie die Parameter des "AS_MAIL". Versuchen Sie, eine E-Mail ohne Authentisierung zu versenden. Ersetzen Sie dazu den Parameter USERNAME durch einen leeren String.
8501	-	Syntax-Fehler: Parameter oder Argument falsch	Möglicherweise haben Sie eine fehlerhafte Adresse in TO_S oder CC eingegeben.
8502	-	Befehl unbekannt bzw. nicht implementiert.	Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" oder "." vergessen.
8535	-	SMTP-Authentisierung unvollständig.	Eventuell haben Sie einen fehlerhaften Benutzernamen oder ein fehlerhaftes Passwort eingegeben.
8550	-	Mail-Server kann nicht erreicht werden, Sie haben keine Zugriffsrechte.	Möglicherweise haben Sie einen fehlerhaften Benutzernamen bzw. Passwort eingegeben oder der Mail-Server unterstützt Ihr LOGIN nicht. Eine weitere Fehlerursache könnte eine fehlerhafte Eingabe des Domainnamens nach dem "@" in TO_S bzw. CC sein.
8552	-	Aktion abgebrochen: Überschreitung der zugewiesenen Speichergröße	Versuchen Sie es später noch einmal.
8554	-	Übertragung misslungen.	Versuchen Sie es später noch einmal.

Hinweis

Statusfehler

1. Eine fehlerhafte Eingabe der Empfängeradressen erzeugt keinen Statusfehler der Anweisung "AS_MAIL". Das Versenden der E-Mail an weitere Empfänger, auch wenn diese richtig eingegeben wurden, ist in diesem Fall nicht sichergestellt.
2. Weitere Informationen zum SMTP-Fehlercode und weitere Fehlercodes finden Sie im SMTP-Protokoll im Internet bzw. in der Fehlerdokumentation des Mail-Servers. Darüber hinaus können Sie die letzte, vom Mail-Server gesendete Fehlermeldung in Textform in Ihrem Instanz-DB im Parameter BUFFER1 einsehen. Dort finden Sie unter "Daten" die zuletzt von der Anweisung "AS_MAIL" versendeten Daten.

Aufrufbeispiel für "AS_MAIL"

Beschreibung

In einem Datenbaustein werden die Daten hinterlegt die mittels ANY-Pointer der Anweisung "AS_MAIL" bekannt gegeben werden. In diesem Beispiel wird dafür der DB 2 verwendet.

Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "AS_MAIL" keine Syntaxprüfung der Parameter TO_S, CC und FROM durch.

Parameter TO_S, CC und FROM

Die Parameter TO_S, CC und FROM sind ANY-Pointer auf Strings mit z. B. folgendem Inhalt:

- TO: <wenna@mydomain.com>, <ruby@mydomain.com>,
- CC: <admin@mydomain.com>, <judy@mydomain.com>,
- FROM: <admin@mydomain.com>

Folgende Regeln sind bei der Eingabe der Parameter zu beachten:

- Die Zeichen "TO:", "CC:" und "FROM:" müssen eingegeben werden.
- Vor jeder Adresse muss ein Leerzeichen und eine spitze Klammer auf "<" eingegeben werden.
- Nach jeder Adresse muss eine spitze Klammer zu ">" eingegeben werden.
- Nach jeder Adresse in TO_S und CC muss ein Komma eingegeben werden.
- Unter FROM darf nur eine E-Mail Adresse eingetragen werden, bei dieser darf kein Komma am Ende stehen.

Initialisierung / Anlauf im OB 100

Der Parameter COM_RST der Anweisung "AS_MAIL" muss vor dem ersten Aufruf der Anweisung
gesetzt werden, bzw. immer nach einem erneuten Laden des zugehörigen Instanz-DBs.

In dem folgenden Beispiel ist DB 1 der Instanz-DB der Anweisung "AS_MAIL".

SET		//VKE setzen
R	DB1.REQ	//REQ wird zurückgesetzt, so dass die Anweisung "AS_MAIL" nicht unmittelbar nach dem Setzen von COM_RST beginnt, eine E-Mail zu versenden.
S	DB1.COM_RST	//COM_RST wird gesetzt, dies muss vor dem ersten Aufruf der Anweisung "AS_MAIL" zur Initialisierung einmal durchgeführt werden, bzw. immer nach einem erneuten Laden des zugehörigen Instanz-DBs.

Zyklischer Aufruf

Dieses Aufrufbeispiel für die Anweisung "AS_MAIL" ist zyklisch aufzurufen (z.B. im OB1). Den Parameter REQ müssen Sie im Anwenderprogramm mittels "Variable Steuern" auf "1" setzen.

Sie können diese Zeilen in eine AWL-Quelle kopieren und übersetzen. Voraussetzung dafür ist, dass Sie die Bausteine DB 1 und DB 2 in die Variablentabelle des Projektes eingetragen haben.

Aufrufbeispiel für "AS_MAIL"	Erläuterung
DATA_BLOCK "DB2"	//Globaler Datenbaustein mit den Daten für die Anweisung "AS_MAIL"
TITLE =	
VERSION : 1.0	
STRUCT	
TO_S : STRING [240] := 'TO: <wenna@mydomain.com>, <ruby@mydomain.com>,';	//Empfängeradresse
CC : STRING [240] := 'CC: <admin@mydomain.com>, <judy@mydomain.com>,';	//CC Empfängeradresse
FROM : STRING [60] := 'FROM: <admin@mydomain.com>';	//Absenderadresse
SUB : STRING [60] := 'Status plant 7';	//Betreff
TEXT : STRING [240] := 'Fault in plant 7 sector 2';	//Text
USERNAME : STRING [60] := 'admin';	//Benutzername für LOGIN-Authentifizierung
PASSWORD : STRING [60] := 'test';	//Passwort für LOGIN-Authentifizierung
ATTACHMENT : ARRAY [1 .. 178] OF BYTE;	//Anhang
END_STRUCT ;	
BEGIN	
END_DATA_BLOCK	
DATA_BLOCK "DB1"	//Instanz-DB der Anweisung "AS_MAIL"
TITLE =	
AUTHOR : TELESERV	

Aufrufbeispiel für "AS_MAIL"	Erläuterung
FAMILY : TELESERV	
VERSION : 1.0	
"AS_MAIL"	
BEGIN	
END_DATA_BLOCK	
FUNCTION_BLOCK FB 1	
TITLE =Example AS_MAIL	//Der Parameter COM_RST der Anweisung "AS_MAIL" muss vor dem ersten Aufruf von "AS_MAIL" gesetzt werden, bzw. immer nach einem erneuten Laden des zugehörigen Instanz DBs. Die Aufrufbedingung wird während der Bearbeitung des Anwenderprogramms gesetzt.
VERSION : 1.0	
VAR	
CallCondition: BOOL ;	
END_VAR	
BEGIN	
NETWORK	
CALL AS_MAIL, DB1 (	//Aufruf des AS_MAIL
REQ := # CallCondition,	//Auftragsanstoß bei steigender Flanke
ADR_MAIL_SERVER := DW#16#C0A800C8,	//entspricht der IP-Adresse 192.168.0.200 im HEX-Format
WATCH_DOG_TIME := T#2M,	//Zeitspanne von 2 Min. in der der Versand abgeschlossen sein sollte
USERNAME := DB2.USERNAME,	//Zeiger auf Benutzername des Absenders. Wird DB2.USERNAME weggelassen, d.h. es wird kein Parameter zugewiesen, so wird das Authentifizierungsverfahren LOGIN nicht verwendet.
PASSWORD := DB2.PASSWORD,	//Zeiger auf Kennwort
TO_S := DB2.TO_S,	//Zeiger auf Empfängeradresse
CC := DB2.CC,	//Zeiger auf CC-Empfängeradresse (optional). Wird DB2.CC weggelassen d.h. nicht zugewiesen, so wird die E-Mail nicht an die CC-Empfänger gesendet.
FROM := DB2.FROM,	//Zeiger auf Absenderadresse
SUB := DB2.SUB,	//Zeiger auf den Mailbetreff
TEXT := DB2.TEXT,	//Zeiger auf den Mailtext (optional). Wird DB2.TEXT weggelassen, d.h. nicht zugewiesen, so wird die E-Mail ohne Text versendet.
ATTACHMENT := DB2.ATTACHMENT,	//Zeiger auf den Mailanhang (optional). Beispiel: 101 Bytes im Mail-Anhang - ATTACHMENT := P#DB2.DBX974.0 BYTE 101 (Bsp. bezieht sich auf obigen DB). //Wird DB2.ATTACHMENT weggelassen, d.h. nicht zugewiesen, so wird die E-Mail ohne Anhang versendet.

Aufrufbeispiel für "AS_MAIL"	Erläuterung
BUSY := M46.0,	//Ausgangs-/Zustandsparameter BUSY
DONE := M46.1,	//Ausgangs-/Zustandsparameter DONE
ERROR := M46.2,	//Ausgangs-/Zustandsparameter ERROR
STATUS := MW48,	//Ausgangs-/Zustandsparameter STATUS
SFC_STATUS := MW50	//Ausgangs-/Zustandsparameter SFC_STATUS
);	
	//Prüfung der Aufrufergebnisse auf das Ende der Bearbeitung
U M 46.0;	//ist BUSY ==0?
SPB End;	//nein, BUSY ==1, "AS_MAIL" bearbeitet den Auftrag noch
U M 46.1;	//Wurde die Bearbeitung des; "AS_MAIL" fehlerfrei beendet?
SPB End;	
NOP 0;	//Nein, bei der Bearbeitung ist ein Fehler aufgetreten, BUSY ==0, BIE ==0
	//hier ggf. Fehlerauswertung ergänzen, STATUS, SFC_STATUS
End:BE;	
END_FUNCTION_BLOCK	

Hinweis

Optionale Parameter

Für die optionalen Parameter USERNAME, PASSWORD, CC, TEXT und ATTACHMENT gilt:

Wird eine Zuweisung entfernt bzw. neu eingefügt, so wird eine solche Änderung erst dann gültig, wenn der zugehörige Instanz-DB (im Beispiel DB 1) neu geladen und der Parameter "COM_RST" gesetzt wird.

Programmiereditor 5

5.1 Übersicht über den Programmiereditor

Funktion des Programmiereditors

Der Programmiereditor ist die integrierte Entwicklungsumgebung für das Programmieren von Funktionen, Funktionsbausteinen und Organisationsbausteinen. Er bietet maximale Unterstützung bei der Programmerstellung und bei der Fehlersuche.

Je nach verwendeter CPU, Programmiersprache und Bausteintyp können Aussehen und Funktionalität des Programmiereditors variieren.

Aufbau des Programmiereditors

Das folgende Bild zeigt die Komponenten des Programmiereditors am Beispiel von KOP:

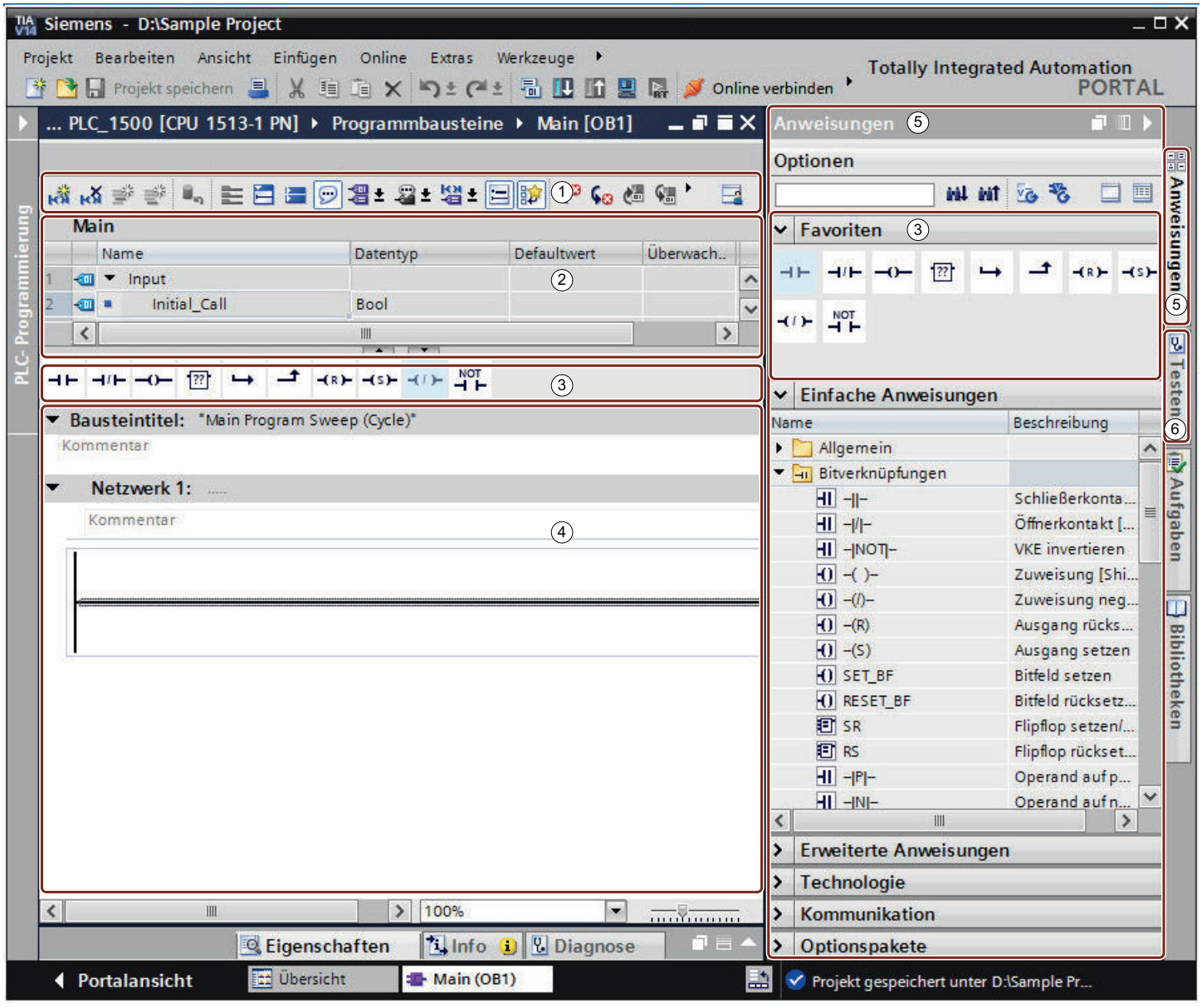

① Funktionsleiste
② Bausteinschnittstelle
③ Palette "Favoriten" in der Task Card "Anweisungen" und Favoriten im Programmierfenster
④ Programmierfenster
⑤ Task Card "Anweisungen"
⑥ Task Card "Testen"

Funktionsleiste

Über die Funktionsleiste haben Sie Zugriff auf die wichtigsten Funktionen des Programmiereditors, z. B.:

- Absolute Operanden ein- und ausblenden
- Favoriten ein- und ausblenden
- Zu Syntaxfehlern springen
- Bausteinaufrufe aktualisieren
- Programmstatus ein- und ausblenden

Je nach verwendeter Programmiersprache können die zur Verfügung stehenden Funktionen der Funktionsleiste variieren.

Bausteinschnittstelle

Die Bausteinschnittstelle enthält die Deklarationen lokaler Variablen, die nur innerhalb des Bausteins verwendet werden. Die verfügbaren Abschnitte variieren abhängig vom Bausteintyp.

Favoriten

Sie haben die Möglichkeit, für häufig verwendete Anweisungen Favoriten anzulegen. Diese Favoriten werden in der Task Card "Anweisungen" in der Palette "Favoriten" angezeigt. Zusätzlich können Sie die Favoriten über die Funktionsleiste des Programmiereditors auch im Programmierfenster anzeigen. Dadurch können Sie auch dann auf Ihre Favoriten zugreifen, wenn die Task Card "Anweisungen" nicht sichtbar ist.

Programmierfenster

Das Programmierfenster ist der Arbeitsbereich des Programmiereditors. Hier können Sie den Programmcode eingeben. Je nach verwendeter Programmiersprache kann das Programmierfenster in Aussehen und Funktionalität variieren.

Task Card "Anweisungen"

Die Task Card "Anweisungen" ermöglicht Ihnen einen einfachen Zugriff auf alle Anweisungen, die Sie für die Erstellung Ihres Programms verwenden können. Die Anweisungen sind thematisch in unterschiedliche Paletten gegliedert. Wenn ein Anweisungsprofil aktiv ist, können die angebotenen Anweisungen variieren. Für jede Anweisung erhalten Sie eine kurze Beschreibung. Wenn eine Anweisung in mehreren Versionen verfügbar ist, werden die Versionen in einer Klappliste aufgelistet und die aktuell ausgewählte Version wird im Programm verwendet. Sie können bei Bedarf eine andere Anweisungsversion auswählen. Beachten Sie dabei die Hinweise unter "Grundlagen zu Anweisungsversionen (Seite 7137)".

Über die Funktionsleiste der Task Card haben Sie Zugriff auf die folgenden Funktionen:

- Anweisungen nach unten oder nach oben durchsuchen
  Sie haben die Möglichkeit, die Paletten nach bestimmten Anweisungen zu durchsuchen.
- Projekt und Anweisungen hochrüsten
  Wenn Sie noch mit einem Projekt arbeiten, das mit TIA Portal V14 erstellt wurde, stehen Ihnen nur die Anweisungsversionen aus der V14 zur Verfügung. Über die Schaltfläche "Projekt und Anweisungen hochrüsten" können Sie ganz einfach Ihr Projekt auf TIA Portal V14 SP1 hochrüsten. Anschließend enthält die Task Card "Anweisungen" auch die neuen Anweisungsversionen von TIA Portal V14 SP1.

  Hinweis

  Beachten Sie die folgenden Hinweise:

  - Ihr Programm wird durch das Hochrüsten nicht geändert, d. h. es enthält noch die alten Anweisungsversionen aus der V14. Um die neuen Anweisungsversionen zu verwenden, müssen Sie Ihr Programm aktualisieren.
  - Die Schaltfläche "Projekt und Anweisungen hochrüsten" ist nur aktiv, wenn Sie noch mit einem TIA Portal V14-Projekt arbeiten. Sobald Sie Ihr Projekt auf TIA Portal V14 SP1 hochgerüsten, wird die Schaltfläche inaktiv.

- Programm in der aktuellen CPU aktualisieren
  Aktualisiert das Programm der CPU auf die neuen Anweisungsversionen.
- Palettenmodus ändern
  Sie können zwischen Einzel- und Mehrpalettenmodus wählen.
  Siehe auch: Palettenmodus ändern
- Spaltenüberschriften ein-/ausblenden
  Sie können die Überschriften zu den Spalten in den Paletten ein- oder ausblenden. Die Anordnung der Spalten können Sie ändern, indem Sie auf eine Spaltenüberschrift klicken und die Spalte per Drag & Drop verschieben.

Task Card "Testen"

In der Task Card "Testen" können Sie Einstellungen vornehmen, die sich auf die Fehlersuche mithilfe des Programmstatus auswirken. Die Funktionen der Task Card "Testen" stehen Ihnen nur im Online-Modus zur Verfügung. Sie enthält folgende Paletten, die abhängig von der ausgewählten CPU und der eingestellten Programmiersprache des Bausteins angezeigt werden:

- CPU-Bedienpanel
  Im CPU-Bedienpanel können Sie den Betriebszustand der CPU umschalten.
- Haltepunkte
  Bausteine, die Sie in einer der textuellen Programmiersprachen AWL oder SCL erstellt haben, können Sie im Einzelschrittmodus testen. Dazu setzen Sie Haltepunkte in den Programmcode.
  In der Palette "Haltepunkte" finden Sie alle Haltepunkte, die Sie gesetzt haben, und können diese aktivieren, löschen, zu einzelnen Haltepunkten navigieren oder die Aufrufumgebung für den Haltepunkt festlegen.
- AS-Register
  In dieser Palette können Sie die Werte für die AS-Register und die Akkumulatoren ablesen.

- Kettensteuerung
  In dieser Palette legen Sie die Betriebsart für das Testen von Ablaufketten für GRAPH-Bausteine fest.
- Testeinstellungen
  In dieser Palette legen Sie die Testeinstellungen für GRAPH-Bausteine fest.
- Aufrufumgebung
  In dieser Palette legen Sie die Aufrufumgebung für den Baustein fest.
- Aufrufhierarchie
  In dieser Palette können Sie die Aufrufhierarchie der Bausteine verfolgen. Sie sehen die Aufrufhierarchie nur während des Beobachtens von Bausteinen.

Siehe auch

Übersicht über die Bausteinschnittstelle (Seite 7263)

Fläche des Programmierfensters vergrößern (Seite 7127)

Nach Anweisungen suchen (Seite 7133)

Anweisungsprofile verwenden (Seite 7141)

5.2 Tastaturbedienung im Programmiereditor

Im Editor navigieren

Funktion	Tastenkombination
Task Card "Anweisungen" öffnen	<Strg+Shift+C>
Task Card "Testen" öffnen	<Strg+Shift+O>
Neuen Baustein hinzufügen	<Strg+N>
Baustein/PLC-Datentyp öffnen	<F7>
Alle Netzwerke erweitern	<Alt+F11>
Alle Netzwerke reduzieren	<Alt+F12>
Zur nächsten Verwendungsstelle des selektierten Bausteins oder Operanden navigieren	<Strg+Shift+G>
Zur vorangehenden Verwendungsstelle des selektierten Bausteins oder Operanden navigieren	<Strg+Shift+F>
Zum nächsten Lese-/Schreibzugriff navigieren	<Alt+F8>
Zum vorangehenden Lese-/Schreibzugriff navigieren	<Alt+F9>
Zur Definition des selektierten Bausteins oder Operanden navigieren	<Strg+Shift+D>

Im Programmcode navigieren (KOP/FUP)

Funktion	Selektiertes Objekt	Tastenkombination
Zwischen den Objekten im Netzwerk navigieren	Objekt im Netzwerk	Pfeiltasten
Zum ersten Element des Netzwerks navigieren	Objekt im Netzwerk	<Pos1>
Zum letzten Element des Netzwerks navigieren	Objekt im Netzwerk	<Ende>
Zum nächsten Element des Netzwerks navigieren	Objekt im Netzwerk	<Tab
Zum vorherigen Element des Netzwerks navigieren	Objekt im Netzwerk	<Shift+Tab>
Netzwerk einfügen	Beliebig	<Strg+R>

Im Programmcode navigieren (AWL/SCL)

Funktion	Position der Einfügemarke	Tastenkombination
Im Programmcode navigieren	Zeile	Pfeiltasten
Ein Wort nach rechts/links	Zeile	<Strg+Pfeiltasten>
Zum Zeilenanfang	Zeile	<Pos1>
Zum Zeilenende	Zeile	<Ende>
Zum Anfang eines Codeabschnitts	Zeile	<Strg+Pos1>
Zum Ende eines Codeabschnitts	Zeile	<Strg+Ende>
Zum nächsten Netzwerk (nur AWL)	Netzwerktitel	<Pfeil-nach-unten>

Funktion	Position der Einfügemarke	Tastenkombination
Zum nächsten Netzwerk (nur AWL)	Zeile	<Tab> Wiederholen Sie die Tastenkombination bis die Einfügemarke im nächsten Netzwerk steht.
Zum vorherigen Netzwerk (nur AWL)	Netzwerktitel	<Pfeil-nach-oben>
Zum vorherigen Netzwerk (nur AWL)	Zeile	<Shift+Tab> Wiederholen Sie die Tastenkombination bis die Einfügemarke im vorherigen Netzwerk steht.
Netzwerk einfügen	Beliebig	<Strg+R>

Anweisungen einfügen (KOP)

Funktion	Selektiertes Objekt	Tastenkombination
Schließerkontakt einfügen	Strompfad	<Shift+F2>
Öffnerkontakt einfügen	Strompfad	<Shift+F3>
Leerbox einfügen	Strompfad	<Shift+F5>
Zuweisung einfügen	Strompfad	<Shift+F7>
"Verzweigung öffnen" einfügen	Strompfad	<Shift+F8>
"Verzweigung schließen" einfügen	Strompfad	<Shift+F9>

Anweisungen einfügen (FUP)

Funktion	Selektiertes Objekt	Tastenkombination
UND-Verknüpfung einfügen	Netzwerk, Ein- oder Ausgang	<Shift+F2>
ODER-Verknüpfung einfügen	Netzwerk, Ein- oder Ausgang	<Shift+F3>
Leerbox einfügen	Netzwerk	<Shift+F5>
Zuweisung einfügen	Netzwerk, Ein- oder Ausgang	<Shift+F7>
"Verzweigung öffnen" einfügen	Verbindungslinie zwischen zwei Boxen	<Shift+F8>
VKE invertieren	Netzwerk, Ein- oder Ausgang	<Strg+Shift+4>
Eingang einfügen	Netzwerk, Ein- oder Ausgang	<Strg+Shift+3>

Operanden eingeben (KOP/FUP/GRAPH)

Funktion	Selektiertes Objekt	Tastenkombination
Eingabefeld für den ersten Operanden der Anweisung aktivieren	Anweisung	<Return> Oder <Beliebige Buchstaben/ Zahlen> Bei <Return> wird ein Eingabefeld geöffnet, bei Buchstaben oder Zahlen werden diese ins Eingabefeld übernommen.
Eingabefeld für den Operanden aktivieren	Operand	<F2>
Operand löschen	Operand	<Entf>
Variable definieren	Operand	<Strg+Shift+I>
Variable umverdrahten	Operand	<Strg+Shift+P>
Variable umbenennen	Operand	<Strg+Shift+T>
Operanden eingeben	Eingabefeld für Operanden	<Beliebige Buchstaben/ Zahlen>
Eingabe des Operanden bestätigen	Eingabefeld für Operanden	<Return>
Autovervollständigung öffnen	Eingabefeld für Operanden	<Strg+I>
Aktuelle Änderung verwerfen	Eingabefeld für Operanden	<ESC> Das Eingabefeld wird deaktiviert und der vorherige Inhalt wieder hergestellt.

Anweisungen bearbeiten (AWL/SCL)

Funktion	Selektiertes Objekt	Tastenkombination
Zeile einrücken (nur SCL)	Zeile	<Tab> oder <Strg+R>
Zeile ausrücken (nur SCL)	Zeile	<Shift+Tab> oder <Strg+Shift+R>
Selektierten Text automatisch formatieren (nur SCL)		<Strg+Shift+W>
Dialog "Aufrufoptionen" öffnen	Einfügemarke hinter einem Bausteinaufruf	<Return>
Variable definieren	Operand	<Strg+Shift+I>
Variable umverdrahten	Operand	<StrgShift+P>
Variable umbenennen	Operand	<Strg+Shift+T>
Parameterliste erweitern/reduzieren (nur SCL)	Operand	<Strg+Shift+Leertaste>
Codeabschnitt auf-/zuklappen	Einfügemarke innerhalb des Codeabschnitts	<Strg+Shift+Num+> <Strg+Shift+Num->

Funktion	Selektiertes Objekt	Tastenkombination
Alle Codeabschnitte auf-/zuklappen	Beliebig	<Strg+Shift+Num*> <Strg+Shift+Num/>
Autovervollständigung öffnen	Beliebig	<Strg+I> oder <Strg+Leertaste>
Lesezeichen setzen/löschen		<Strg+Shift+M>
Zum nächsten Lesezeichen		<Strg+Shift+6>
Zum vorherigen Lesezeichen		<Strg+Shift+5>
Markierung auskommentieren	Zeile	<Strg+Shift+Y>
Kommentar entfernen	Zeile	<Strg+Shift+U>

Programmierfenster von GRAPH

Funktion	Bereich	Tastenkombination
Eine Seite nach oben/unten	Navigation, Einzelschrittansicht, Kettenansicht, permanente Anweisungen	<Bild-nach-oben>/ <Bild-nach-unten>
In der Navigation navigieren	Navigation	<Pfeil-nach-oben> <Pfeil-nach-unten>
Objekt aufklappen/zuklappen	Navigation	<+> oder <Pfeil-nach-rechts>/ <-> oder <Pfeil-nach-links>
Zwischen Einzelschrittansicht und Kettenansicht wechseln, wenn ein Schritt oder eine Transition selektiert ist	Navigation	<Return>
Zwischen Navigation und Arbeitsbereich wechseln	Navigation, Einzelschrittansicht, Kettenansicht, permanente Anweisungen	<ALT+F6>
Zum ersten Element in einem Netzwerk	Einzelschrittansicht	<Pos1>
Zum letzten Element in einem Netzwerk	Einzelschrittansicht	<Ende>
Zum Interlock wechseln	Einzelschrittansicht	<Strg+Pos1>
Zur Transition wechseln	Einzelschrittansicht	<Strg+Ende>
In der Struktur navigieren	Kettenansicht	Pfeiltasten
Zum ersten Schritt	Kettenansicht	<Pos1> oder <Strg+Pos1>
Zum letzten Schritt	Kettenansicht	<Ende> oder <Strg+Ende>
Verzweigung öffnen	Kettenansicht	<Shift+F8>
Verzweigung schließen	Kettenansicht	<Shift+F9>
Kettenende einfügen	Kettenansicht	<Shift+F7>
Sprung einfügen	Kettenansicht	<Shift+F12>
Schritt und Transition einfügen	Kettenansicht	<Shift+F5>
Element löschen	Kettenansicht	<Entf>

Funktion	Bereich	Tastenkombination
Zum ersten editierbaren Element	permanente Anweisungen	<Pos1>
Zum nächsten editierbaren Element	permanente Anweisungen	<Tab>
Zum letzten editierbaren Element	permanente Anweisungen	<Ende>
Zum vorherigen editierbaren Element	permanente Anweisungen	<Shift+Tab>
Springe zum Anfang der Zelle "Aktion"	Aktionen	<Pos1>
Springe zum Ende der Zelle "Aktion"	Aktionen	<Ende>
Neue Aktion einfügen	Aktionen	<Return>

Programmierfenster von SCL

Funktion	Bereich	Tastenkombination
Bereich aufklappen	Bereichsübersicht, Bereich im Programmierfenster	<Strg+Shift+Num+>
Bereich zuklappen	Bereichsübersicht, Bereich im Programmierfenster	<Strg+Shift+Num->
Alle Bereiche aufklappen	Bereichsübersicht, Programmierfenster	<Strg+Shift+Num*>
Alle Bereiche zuklappen	Bereichsübersicht, Programmierfenster	<Strg+Shift+Num/>

Programm beobachten

Funktion	Tastenkombination
Haltepunkt setzen/löschen (AWL, SCL)	<Strg+Shift+F9>
Haltepunkt überspringen (AWL SCL)	<Strg+Shift+F10>
In einen unterlagerten Baustein hineinspringen (AWL SCL)	<Strg+Shift+F11>
In den aufrufenden Baustein zurückspringen (AWL, SCL)	<Strg+Shift+F12>
Programm bis zur Markierung (Cursor-Position) ausführen (AWL SCL)	<Strg+F3>
Programmstatus anzeigen (AWL, SCL, PLC-Variablentabelle, Beobachtungstabelle und Datenbaustein)	<Strg+T>
Alle Haltepunkte aktivieren (AWL, SCL)	<Strg+Shift+F2>
Alle Haltepunkte deaktivieren (AWL, SCL)	<Strg+Shift+F3>
Steuern auf 0 (KOP, FUP)	<Strg+Shift+9>
Steuern auf 1 (KOP, FUP)	<Strg+Shift+1>
Operand steuern (KOP, FUP)	<Strg+Shift+2>

5.3 Fläche des Programmierfensters vergrößern

Einführung

Wenn alle Komponenten der Anwendung eingeblendet sind, ist die Fläche des Programmierfensters relativ klein. Daher kann es vorkommen, dass Sie den Arbeitsbereich bei umfangreichem Programmcode häufig verschieben müssen. Um dies zu vermeiden, können Sie die Anzeige folgender Komponenten der Anwendung und des Programmiereditors ausblenden oder minimieren:

- Projektnavigation
- Task Cards
- Bausteinschnittstelle
- Favoriten
- Kommentare
- Netzwerke

Hinweis

Sie können für die Task Cards, die Projektnavigation und das Inspektorfenster zusätzlich die Option "Automatisch reduzieren" verwenden. Dadurch klappen diese Fenster automatisch zu, wenn Sie sie nicht benötigen.

Siehe auch: Arbeitsbereich maximieren und minimieren

Projektnavigation aus- und einblenden

Über die Projektnavigation greifen Sie auf alle Bereiche des Projekts zu. Während der Programmerstellung können Sie die Projektnavigation ausblenden, um die Fläche des Programmierfensters zu vergrößern.

Um die Projektnavigation aus- oder einzublenden, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie im Menü "Ansicht" das Optionskästchen "Projektnavigation" oder klicken Sie auf die Schaltfläche "Reduzieren" in der Titelleiste der Projektnavigation, um die Projektnavigation auszublenden.
2. Aktivieren Sie im Menü "Ansicht" das Optionskästchen "Projektnavigation" oder klicken Sie auf die Schaltfläche "Erweitern" in der Titelleiste der Projektnavigation, um die Projektnavigation einzublenden.

Task Cards öffnen und schließen

Die Task Cards befinden sich am rechten Rand des Programmierfensters.

Um die Task Cards zu schließen oder zu öffnen, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie im Menü "Ansicht" das Optionskästchen "Task Card" oder klicken Sie auf die Schaltfläche "Reduzieren" in der Titelleiste der Task Cards, um die Task Cards zu schließen.
2. Aktivieren Sie im Menü "Ansicht" das Optionskästchen "Task Card" oder klicken Sie auf die Schaltfläche "Erweitern" in der Titelleiste der Task Cards, um die Task Cards zu öffnen.

Bausteinschnittstelle aus- und einblenden

Die Bausteinschnittstelle eines Bausteins befindet sich im oberen Bereich des Programmiereditors. Während der Programmerstellung kann sie aus- und bei Bedarf wieder eingeblendet werden.

Um die Bausteinschnittstelle aus- oder einzublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie im unteren Bereich der Schnittstelle innerhalb des Fensterteilers auf den Pfeil-nach-oben oder Pfeil-nach-unten.

Favoriten ein- und ausblenden

Um die Favoriten im Programmiereditor ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Kommentare ein- und ausblenden

In einem Baustein können Sie einen Kommentar zum Baustein oder für jedes Netzwerk eingeben. Diese beiden Kommentararten werden unterschiedlich ein- und ausgeblendet.

Um einen Bausteinkommentar ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Zeile mit dem Bausteintitel auf das Dreieck am Anfang der Zeile.

Um Netzwerkkommentare ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf "Netzwerkkommentare ein/aus".

Hinweis

Je nach verwendeter Programmiersprache können die zur Verfügung stehenden Kommentare variieren.

Netzwerke auf- und zuklappen

Einige Programmiersprachen verwenden Netzwerke. Diese Netzwerke können Sie je nach Bedarf auf- oder zuklappen.

Um einzelne Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Netzwerk aufklappen möchten, klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel. Wenn Sie ein Netzwerk zuklappen möchten, klicken Sie auf den Pfeil-nach-unten vor dem Netzwerktitel.

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf "Alle Netzwerke öffnen" oder auf "Alle Netzwerke schließen".

Hinweis

Netzwerke werden nicht in jeder Programmiersprache verwendet.

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

5.4 Einstellen der Mnemonik

Bausteine können Sie in deutscher oder internationaler Mnemonik programmieren. Wenn Sie das TIA Portal zum ersten Mal öffnen, ist internationale Mnemonik voreingestellt. Sie können die Mnemonik jederzeit ändern.

Vorgehen

Um die Mnemonik einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Wählen Sie in der Gruppe "Allgemeine Einstellungen" die gewünschte Mnemonik aus.
   Die Mnemonik wird in allen Bausteinen umgestellt.

5.5 Symbolische und absolute Operanden anzeigen

Für die Darstellung der Operanden im Programmiereditor gibt es folgende Möglichkeiten:

- Symbolische Darstellung
  Im Programm werden die symbolischen Operanden angezeigt. Die dazugehörigen absoluten Adressen werden in Tooltips eingeblendet, wenn der Mauszeiger längere Zeit über dem Operanden verweilt.
- Absolute Darstellung
  Im Programm werden die absoluten Adressen angezeigt. Die dazugehörigen symbolischen Operanden werden in Tooltips eingeblendet.
- Symbolische und absolute Darstellung
  Im Programm werden symbolische Operanden und absolute Adressen angezeigt. Diese Einstellung wirkt sich nur auf Bausteine aus, die mit KOP, FUP, AWL und GRAPH programmiert wurden.

Hinweis

Beachten Sie für GRAPH-Bausteine, dass keine absoluten Operanden angezeigt werden, wenn Sie im Kontextmenü der Spalte "Aktion" die Option "Absolute Operanden anzeigen" deaktiviert haben.

Sie können die Darstellung der Operanden zentral für alle neuen Bausteine des Projekts oder für einzelne Bausteine einstellen.

Operandendarstellung für neue Bausteine einstellen

Um die Operandendarstellung für alle neuen Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Wählen Sie in der Klappliste "Operandendarstellung" die gewünschte Darstellung aus.
   Die ausgewählte Darstellung wird für alle neuen Bausteine im Programm verwendet.

Operandendarstellung für einen einzelnen Baustein einstellen

Um die Darstellung der Operanden zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Absolute/symbolische Operanden".
   Mit jedem Klick auf die Schaltfläche erhalten Sie eine andere Darstellung und das Symbol der Schaltfläche wird geändert.

Oder:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste des Programmiereditors auf den kleinen Pfeil neben der Schaltfläche "Absolute/symbolische Operanden".
   Eine Klappliste wird geöffnet.
3. Wählen Sie die gewünschte Darstellung aus der Klappliste.
   Das Symbol der Schaltfläche wird geändert.

Siehe auch

Grundlagen zu Operanden (Seite 93)

5.6 Nach Anweisungen suchen

Sie haben die Möglichkeit in der Task Card "Anweisungen" nach bestimmten Anweisungen zu suchen, um sie dann in Ihr Programm einzufügen. Beachten Sie bei der Eingabe von Suchbegriffen folgende Regeln:

- Zwischen Groß- und Kleinschreibung wird nicht unterschieden.
- Teile eines gesuchten Begriffs werden bei der Suche berücksichtigt.
- Es können keine Platzhalterzeichen wie z. B. "*" und "?" verwendet werden.
- Falls ein Anweisungsname Unterstriche enthält, wird die Anweisung auch dann bei der Suche gefunden, wenn Sie den Unterstrich nicht eingeben.
- Bei der Suche werden die Texte in den Spalten "Name" und "Beschreibung" berücksichtigt.

Voraussetzung

- Ein Baustein ist geöffnet.
- Die Task Card "Anweisungen" ist geöffnet.

Vorgehen

Um in der Task Card "Anweisungen" nach einer bestimmten Anweisung zu suchen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Startpunkt der Suche, wenn Sie ab einer bestimmten Stelle suchen möchten. Wenn Sie nichts selektieren, beginnt die Suche entweder oben oder unten in der Task Card, abhängig von der Art der Suche, die Sie wählen.
2. Geben Sie in der Funktionsleiste der Task Card einen Suchbegriff in das Textfeld ein.
3. Klicken Sie auf die Schaltfläche "Suche abwärts", wenn Sie die Task Card von oben nach unten durchsuchen möchten.
4. Oder klicken Sie auf die Schaltfläche "Suche aufwärts", wenn Sie die Task Card von unten nach oben durchsuchen möchten.
   Die erste gefundene Übereinstimmung mit dem Suchbegriff wird als Ergebnis angezeigt. Wenn Sie die Suche fortsetzen möchten, klicken Sie erneut auf die Schaltfläche "Suche abwärts" oder "Suche aufwärts". Falls keine Übereinstimmungen gefunden werden können, erhalten Sie eine entsprechende Meldung.

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

5.7 Mehrsprachige Projekttexte in Bausteinen bearbeiten

Einführung

Ein Baustein kann folgende übersetzbaren Projekttexte enthalten:

- Titel des Bausteins und der Netzwerke
- Kommentare zum Baustein und zu den Netzwerken
- Kommentare in der Bausteinschnittstelle
- Kommentare zu verwendeten Operanden
- Freie Kommentare

Um diese Texte in eine andere Sprache zu übersetzen, müssen Sie diese Sprache als Projektsprache aktivieren. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Beachten Sie, dass die Texte auch nach der Übersetzung die Länge von 32767 Unicode-Zeichen nicht überschreiten dürfen.

Voraussetzung

- Mehrere Projektsprachen sind aktiviert.
- Ein Baustein ist geöffnet und enthält mindestens einen übersetzbaren Projekttext.
- Der Baustein trägt keinen Know-how-Schutz.

Vorgehen

Um einen Text in allen Projektsprachen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie im geöffneten Baustein das gewünschte Element. Die folgende Tabelle gibt einen Überblick über die Selektionsmöglichkeiten:

Selektiertes Element	Angezeigte Projekttexte
Palette "Bausteintitel"	Wenn Sie die Palette "Bausteintitel" selektieren, können Sie folgende Projekttexte bearbeiten: • Titel des Bausteins und der Netzwerke • Kommentare zum Baustein und zu den Netzwerken • Freie Kommentare im gesamten Baustein • Kommentare in der Bausteinschnittstelle Bei Multiinstanzen oder Parameterinstanzen werden nur die Kommentare von Elementen angezeigt, die Sie instanzspezifisch geändert haben. Diese Kommentare können Sie daran erkennen, dass sie nicht gegraut dargestellt sind. Grau dargestellte Kommentare können Sie nur im zugehörigen Funktionsbaustein anwählen und bearbeiten.
Netzwerkpalette	Wenn Sie eine oder mehrere Netzwerkpaletten selektieren, können Sie folgende Projekttexte bearbeiten: • Titel und Kommentare der selektierten Netzwerke • Freie Kommentare in den Netzwerken
Sprachelement	Wenn Sie ein oder mehrere Sprachelemente (z. B. Boxen in KOP/FUP) oder einen Teil des Programmcodes in AWL oder SCL selektieren, können Sie folgende Projekttexte bearbeiten: • Kommentare zu verwendeten Operanden, auch zu globalen PLC-Variablen • Bei Bausteinaufrufen: Kommentare zu den Bausteinparametern, die mit Aktualparametern versorgt sind • Freie Kommentare am Sprachelement
Schritt oder Transition	Wenn Sie in einem GRAPH-Baustein einen Schritt oder eine Transition selektieren, können Sie folgende Projekttexte bearbeiten: • Name des Schrittes • Name der Transition

2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das untergeordnete Register "Texte".
   Im Register "Texte" werden die Texte in den aktiven Projektsprachen und, soweit vorhanden, deren Übersetzungen angezeigt.
4. Sie können die Übersetzungen direkt in die Tabelle im Register "Texte" bearbeiten.
5. Um die Übersetzungen mit einem externen Editor zu bearbeiten, können Sie die angezeigten Texte über die Schaltflächen "Projekttexte exportieren/importieren" in das Format OOXML exportieren.

Hinweis

Alle Projekttexte in der globalen Tabelle "Projekttexte" bearbeiten

Sie können die Übersetzungen für Projekttexte auch in der globale Tabelle "Projekttexte" bearbeiten. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte". Sie enthält alle übersetzbaren Texte des gesamten Projekts.

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

5.8 Anweisungsversionen verwenden

5.8.1 Grundlagen zu Anweisungsversionen

Einführung

Die Anweisungen, die Ihnen zur Programmierung des Anwenderprogramms zur Verfügung stehen, werden in Systembibliotheken verwaltet. Wenn durch eine Aktualisierung eine neue Version einer Systembibliothek installiert wird, werden dadurch eventuell auch neuere Versionen der Anweisungen dieser Systembibliothek installiert.

Wenn es für eine Anweisung mehrere Versionen gibt, werden diese in der Task Card "Anweisungen" hinter der jeweiligen Anweisung in einer Klappliste aufgelistet. Falls die Anweisungsversionen ausgeblendet sind, können Sie diese über das Kontextmenü der Spaltenüberschrift wieder einblenden. Wählen Sie in der Klappliste die Version einer Anweisung, die im Programm verwendet werden soll. Wenn Sie keine Versionen auswählen, werden die neuesten Versionen verwendet.

Hinweis

Beachten Sie folgende Hinweise:

- Innerhalb eines Geräts können Sie eine Anweisung immer nur in der gleichen Version verwenden.
- Falls Sie eine Anweisung in einem know-how-geschützten Baustein in einer Version ungleich der im Projekt eingestellten Version verwenden, können Sie den know-how-geschützten Baustein ohne Passwort übersetzen, wenn die Schnittstellen der Anweisungsversionen identisch sind. Beim nächsten Öffnen des know-how-geschützten Bausteins müssen Sie den Aufruf für die Anweisung aktualisieren. Der entsprechende Aufruf ist rot gekennzeichnet. Durch das Übersetzen des gesamten Programms wird der Aufruf automatisch vom System aktualisiert.
- Wenn Sie die Version einer Anweisung ändern, von der andere Anweisungen abhängen, werden die Versionen der abhängigen Anweisungen ebenfalls geändert.
- Falls Sie für eine Anweisung eine Version auswählen, die für die verwendete CPU nicht lauffähig ist, wird die Anweisung gegraut dargestellt. Das heißt, Sie können die Anweisung in dieser Version nicht mit Ihrer CPU verwenden.
- Der Systembaustein in der Projektnavigation zeigt in seinen Eigenschaften die Bausteinversionsnummer an und diese muss nicht mit der Anweisungsversionsnummer der zugehörigen Anweisung in der Task Card übereinstimmen.
- Wenn Sie die Version einer Anweisung in der Task Card ändern, müssen Sie den zugehörigen Systembaustein in der Projektnavigation übersetzen, bevor dessen Bausteinversionsnummer aktualisiert wird.

Änderungen in den Versionen

Neue Versionen können Haupt- oder Nebenversionen sein. Neue Hauptversionen, z. B. 2.0 oder 3.0, enthalten größere Änderungen. Daher können neue Hauptversionen Änderungen der Bausteinschnittstelle zur Folge haben. Neue Nebenversionen, z. B. 1.3 oder 1.4, enthalten kleinere Änderungen oder Fehlerbehebungen.

Verwendung von Anweisungsversionen

Sie können innerhalb eines Geräts entscheiden, welche Version einer Anweisung Sie verwenden möchten. Wenn Sie für eine Anweisung eine andere Version auswählen, wird für alle Verwendungsstellen dieser Anweisung in Ihrem Programm die neue Version festgelegt. Im Programm werden diese Anweisungen mit einem roten Rahmen gekennzeichnet. Anschließend müssen Sie Ihr Programm in das Gerät laden, damit die neue Anweisungsversion verwendet wird.

Wenn Sie Ihr Projekt mit einer neueren TIA Portal Version öffnen, können Sie innerhalb der Task Card "Anweisungen" Ihr Projekt und die Anweisungen auf die neue TIA Portal Version hochrüsten. Dadurch stehen Ihnen für einige Anweisungen neuere Versionen zur Verfügung. Diese ersetzen jedoch nicht automatisch die alten Versionen in Ihrem Programm. Um die neuen Anweisungsversionen in Ihrem Programm zu verwenden, müssen Sie das Programm explizit aktualisieren.

Siehe auch

Projekt und Anweisungen hochrüsten (Seite 7138)

Programm der aktuellen CPU aktualisieren (Seite 7139)

5.8.2 Projekt und Anweisungen hochrüsten

Sie können ein Projekt und die Versionen der Anweisungen in einem Schritt auf eine neuere TIA Portal Version hochrüsten. Ihr Programm wird dadurch nicht verändert.

Hinweis

Keine Rückwärtskompatibilität

Projekte, die mit einer aktuellen Version des TIA Portals gespeichert werden, sind aufgrund der erweiterten Funktionalität gegenüber älteren Versionen nicht rückwärts kompatibel. Beim Hochrüsten wird ein neues Projekt in der aktuellen Projektversion gespeichert. Das Ursprungsprojekt wird nicht überschrieben und kann weiterhin mit einer kompatiblen älteren Version des TIA Portals bearbeitet werden.

Vorgehen

Um ein Projekt und die Anweisungen hochzurüsten, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen Codebaustein.
2. Öffnen Sie die Task Card "Anweisungen".

3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Projekt und Anweisungen hochrüsten".
   Der Dialog "Projektkompatibilität" wird geöffnet.
4. Um mit dem Hochrüsten fortzufahren, klicken Sie auf "OK".

Ergebnis

Das Projekt und die Anweisungen werden hochgerüstet. Die Schaltfläche "Projekt und Anweisungen hochrüsten" wird inaktiv und die Funktion kann nicht erneut ausgeführt werden.

Als nächsten Schritt können Sie Ihr Programm aktualisieren, damit die neuen Anweisungsversionen verwendet werden.

Siehe auch

Grundlagen zu Anweisungsversionen (Seite 7137)

Programm der aktuellen CPU aktualisieren (Seite 7139)

5.8.3 Programm der aktuellen CPU aktualisieren

Wenn in Ihrem Projekt neue Anweisungsversionen vorhanden sind, können Sie das Programm einer CPU aktualisieren, damit die neuen Anweisungsversionen verwendet werden. Dies kann zum Beispiel nach dem Hochrüsten Ihres Projekts auf eine neue TIA Portal Version der Fall sein. Beachten Sie, dass Sie Ihr Programm nach dem Aktualisieren übersetzen müssen.

Vorgehen

Um das Programm in der aktuellen CPU zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf die CPU, für die Sie das Programm aktualisieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Programm aktualisieren".
   Der Dialog "Programm in der aktuellen CPU aktualisieren" wird geöffnet.
3. Um die Aktualisierung durchzuführen, klicken Sie auf "OK".
4. Übersetzen Sie Ihr Programm, um die Aktualisierung abzuschließen.

Oder:

1. Öffnen Sie einen Codebaustein.
2. Öffnen Sie die Task Card "Anweisungen".
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Programm in der aktuellen CPU aktualisieren".
   Der Dialog "Programm in der aktuellen CPU aktualisieren" wird geöffnet.
4. Um die Aktualisierung durchzuführen, klicken Sie auf "OK".
5. Übersetzen Sie Ihr Programm, um die Aktualisierung abzuschließen.

Siehe auch

Grundlagen zu Anweisungsversionen (Seite 7137)
Projekt und Anweisungen hochrüsten (Seite 7138)

5.9 Anweisungsprofile verwenden

5.9.1 Grundlagen zu Anweisungsprofilen

Einführung

Das TIA Portal stellt Ihnen für die Programmierung Ihres Anwenderprogramms eine Vielzahl von Anweisungen zur Verfügung. Wenn Sie jedoch nicht alle Anweisungen verwenden möchten, haben Sie die Möglichkeit, die nicht benötigten Anweisungen auszufiltern. Zu diesem Zweck können Sie Anweisungsprofile erstellen und innerhalb eines solchen Profils genau festlegen, welche Anweisungen in der Task Card "Anweisungen" angeboten werden sollen. Sie können innerhalb eines Projekts mehrere Anweisungsprofile anlegen, jedoch kann immer nur ein Profil aktiv sein. Über globale Bibliotheken können Sie Anweisungsprofile mit anderen Anwendern austauschen.

Hinweis

Beachten Sie folgende Hinweise:

- Wenn in einem Baustein Anweisungen verwendet werden, die durch das aktive Profil nicht zugelassen sind, erhalten Sie einen Fehler beim Übersetzen des Bausteins. Dies könnte z. B. durch einen Baustein passieren, den Sie aus einer Bibliothek in Ihr Programm ziehen.
- Anweisungen eines Profils, die durch die aktuell installierten Produkte nicht unterstützt werden, werden beim nächsten Bearbeiten des Profils aus dem Profil gelöscht. Wenn Sie dieses Profil in ein Engineering System übertragen, in dem diese Anweisungen durch die installierten Produkte unterstützt werden, sind die Anweisungen wieder im Profil vorhanden, jedoch deaktiviert. Sie können diese Anweisungen bei Bedarf jederzeit aktivieren.
- Wenn Sie Änderungen am aktiven Profil vornehmen, müssen Sie die Bausteine im Projekt neu übersetzen. Dies ist auch beim Deaktivieren und Löschen des aktiven Profils oder beim Aktivieren eines Profils erforderlich.

Siehe auch

Neues Anweisungsprofil erzeugen (Seite 7141)

Anweisungsprofil öffnen und bearbeiten (Seite 7143)

Anweisungsprofil aktivieren und deaktivieren (Seite 7144)

Anweisungsprofil löschen (Seite 7145)

5.9.2 Neues Anweisungsprofil erzeugen

Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

Vorgehen

Um ein neues Anweisungsprofil zu erzeugen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neues Anweisungsprofil hinzufügen".
   Der Anweisungsprofil-Editor wird geöffnet und das neue Anweisungsprofil wird angezeigt. Für das neue Anweisungsprofil sind alle Anweisungen aktiviert.
2. Bearbeiten Sie das neue Anweisungsprofil gemäß Ihren Anforderungen.

Sie können das neue Anweisungsprofil bei Bedarf umbenennen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das neue Anweisungsprofil.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen Namen für das neue Anweisungsprofil ein.

Hinweis

Wenn Sie das erste Anweisungsprofil erzeugen, wird dieses als aktives Profil verwendet. Übersetzen Sie in diesem Fall alle Bausteine im Projekt. Falls bereits weitere Anweisungsprofile vorhanden sind, müssen Sie das neue Profil explizit aktivieren, damit es als aktives Profil verwendet wird. Das aktive Profil erkennen Sie in der Projektnavigation am Symbol.

Siehe auch

Grundlagen zu Anweisungsprofilen (Seite 7141)

Anweisungsprofil öffnen und bearbeiten (Seite 7143)

Anweisungsprofil aktivieren und deaktivieren (Seite 7144)

Anweisungsprofil löschen (Seite 7145)

5.9.3 Anweisungsprofil öffnen und bearbeiten

Nach dem Öffnen eines Anweisungsprofils können Sie es folgendermaßen bearbeiten:

- Anweisungen aktivieren und deaktivieren
  Sie können für jede Anweisung festlegen, ob sie innerhalb des Anweisungsprofils zugelassen ist.

  Hinweis

  Beachten Sie, dass zwischen einigen Anweisungen Abhängigkeiten bestehen. Dadurch werden evtl. mit einer Aktion mehrere Anweisungen aktiviert oder deaktiviert. Sie können am Symbol des Optionskästchens erkennen, in welchen Ordnern Anweisungen deaktiviert sind.

- Anweisungsversionen aktivieren und deaktivieren
  Einige Anweisungen sind mit verschiedenen Versionen vorhanden. Wenn eine Anweisung mehrere Versionen besitzt, können Sie für jede Version festlegen, ob sie innerhalb des Anweisungsprofils zugelassen ist.
- Bausteinnummer ändern
  Wenn eine Anweisung systemintern ein Funktionsbaustein (FB) oder eine Funktion (FC) ist, erhält die Anweisung vom System eine bestimmte Bausteinnummer. Sie haben die Möglichkeit, diese Bausteinnummer durch eine eigene Bausteinnummer zu ersetzen. Für einige Anweisungen gibt es innerhalb einer Version mehrere Implementierungen. Bei solchen Anweisungen können Sie die Bausteinnummern nur für die einzelnen Implementierungen ändern.

  Hinweis

  Wenn die Anweisung aus dem Anweisungsprofil im Programm verwendet wird und die festgelegte Bausteinnummer im Anwenderprogramm bereits durch einen anderen Baustein belegt ist, wird die festgelegte Bausteinnummer der Anweisung durch eine freie Bausteinnummer ersetzt.

Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

Anweisungsprofile öffnen

Um ein Anweisungsprofil zu öffnen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf das Anweisungsprofil, das Sie bearbeiten möchten.
   Das Anweisungsprofil wird im Anweisungsprofil-Editor geöffnet.

Anweisungsprofile bearbeiten

Um ein Anweisungsprofil im Anweisungsprofil-Editor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Gerätefamilie" das Gerät, für das Sie das Anweisungsprofil bearbeiten möchten.
2. Wählen Sie in der Klappliste "Sprache" die Programmiersprache, für die Sie das Anweisungsprofil bearbeiten möchten.
3. Deaktivieren Sie die Anweisungen bzw. Anweisungsversion, die Sie aus dem Anweisungsprofil ausschließen möchten. Wenn Sie einen Ordner deaktivieren, werden alle unterlagerten Anweisungen deaktiviert.
4. Aktivieren Sie die Anweisungen bzw. Anweisungsversionen, die Sie innerhalb des Anweisungsprofils erlauben möchten.
5. Vergeben Sie bei Bedarf eigene Bausteinnummern.

   Hinweis

   Für CPUs der Baureihe S7-1200/1500 können Sie Nummern bis 65535 vergeben. Für CPUs der Baureihe S7300/400 finden Sie die Beschränkungen der Nummernbänder im jeweiligen Handbuch der CPU.

Hinweis

Wenn Sie das aktive Profil ändern, ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

Siehe auch

Grundlagen zu Anweisungsprofilen (Seite 7141)

Neues Anweisungsprofil erzeugen (Seite 7141)

Anweisungsprofil aktivieren und deaktivieren (Seite 7144)

Anweisungsprofil löschen (Seite 7145)

Grundlagen zu Anweisungsversionen (Seite 7137)

5.9.4 Anweisungsprofil aktivieren und deaktivieren

Damit die Filterung der Anweisungen eines Anweisungsprofils berücksichtigt werden kann, muss das Anweisungsprofil zuerst aktiviert werden. Sie können das aktive Anweisungsprofil jederzeit wieder deaktivieren, anschließend enthält die Task Card "Anweisungen" wieder den Standardumfang an Anweisungen.

Hinweis

Es ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

Anweisungsprofil aktivieren

Um ein Anweisungsprofil zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisungsprofil aktivieren".
   Das selektierte Anweisungsprofil ist jetzt das aktive Profil. Anweisungen können nur noch gemäß den Einstellungen dieses Profils verwendet werden.

Anweisungsprofil deaktivieren

Um das aktive Anweisungsprofil zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisungsprofil deaktivieren".
   Es ist kein Anweisungsprofil aktiv, in der Task Card "Anweisungen" werden wieder alle Anweisungen angezeigt und können verwendet werden.

Siehe auch

Grundlagen zu Anweisungsprofilen (Seite 7141)

Neues Anweisungsprofil erzeugen (Seite 7141)

Anweisungsprofil öffnen und bearbeiten (Seite 7143)

Anweisungsprofil löschen (Seite 7145)

5.9.5 Anweisungsprofil löschen

Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

Vorgehen

Um ein Anweisungsprofil zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Wenn Sie das aktive Profil löschen, ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

Ergebnis

Das selektierte Anweisungsprofil wird gelöscht. Wenn das gelöschte Anweisungsprofil das aktive Profil war, ist kein Profil mehr aktiv, sondern es wird in der Task Card "Anweisungen" wieder der Standardumfang an Anweisungen angeboten.

Siehe auch

Grundlagen zu Anweisungsprofilen (Seite 7141)

Neues Anweisungsprofil erzeugen (Seite 7141)

Anweisungsprofil öffnen und bearbeiten (Seite 7143)

Anweisungsprofil aktivieren und deaktivieren (Seite 7144)

5.10 Autovervollständigung verwenden

5.10.1 Grundlagen zur Autovervollständigung

Funktion

Im Programmierfenster des Programmiereditors können Sie die Autovervollständigung verwenden, um während des Programmierens bequem auf vorhandene Variablen oder Anweisungen zuzugreifen. Autovervollständigung bedeutet dabei, dass Sie in einem Dialog eine kontextabhängige Liste erhalten, aus der Sie die gewünschten Variablen oder Anweisungen auswählen können.

Siehe auch

Autovervollständigung in grafischen Programmiersprachen verwenden (Seite 7147)

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

5.10.2 Autovervollständigung in grafischen Programmiersprachen verwenden

Variablen mithilfe der Autovervollständigung einfügen

Um in grafischen Programmiersprachen Variablen mithilfe der Autovervollständigung einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Operanden der Anweisung, dem Sie eine Variable zuweisen möchten.
   Das Eingabefeld für den Operanden wird geöffnet. Neben dem Eingabefeld erscheint die Schaltfläche für die Autovervollständigung.
2. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur diejenigen lokalen und globalen Variablen, Datenbausteine und Multiinstanzen, die für den Operanden zulässig sind. Mit <Esc> können Sie die Autovervollständigung jederzeit verlassen.

3. Wählen Sie die gewünschte Variable aus der Liste aus. Sie können die Liste bei Bedarf auch filtern:
   - Geben Sie z. B. die Anfangsbuchstaben der Variablen oder Anweisung ein, die Sie einfügen möchten. Mit jedem eingegebenen Buchstaben wird die Autovervollständigung weiter gefiltert. Wenn es keine Variable oder Anweisung gibt, die mit den eingegebenen Buchstaben beginnt, bleibt die Autovervollständigung bei der letzten Übereinstimmung stehen.
   - Geben Sie # ein, um auf die lokalen Variablen aus der Bausteinschnittstelle zuzugreifen.
   - Geben Sie " ein, um auf die globalen Variablen zuzugreifen.
   - Geben Sie % ein, um auf absolute Adressen zuzugreifen.

   Falls es sich bei einer Variablen um eine strukturierte Variable, einen Datenbaustein oder eine Multiinstanz handelt, wird am Ende der Zeile ein Pfeil angezeigt. Klicken Sie auf den Pfeil, um die untergeordneten Elemente angezeigt zu bekommen. Auf diese Weise können Sie bis zur letzten Ebene navigieren. Falls für den Operanden eine Struktur als Datentyp zulässig ist, können Sie aus der Liste "Kein Eintrag" auswählen. Dadurch wird die komplette Struktur dem Operanden als Variable zugewiesen. Um zur vorherigen Ebene zurückzukehren, verwenden Sie die <Rücktaste>.
4. Drücken Sie die Taste <Return>, um die Variable zu übernehmen.

Siehe auch

Grundlagen zur Autovervollständigung (Seite 7147)

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

5.10.3 Autovervollständigung in textuellen Programmiersprachen verwenden

Variablen und Anweisungen mithilfe der Autovervollständigung einfügen

Um in textuellen Programmiersprachen Variablen und Anweisungen mithilfe der Autovervollständigung einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Anfangsbuchstaben der Variable oder Anweisung ein, die Sie einfügen möchten. Sie können bei Bedarf auch direkt nach der Art der Variablen filtern:
   - Geben Sie # ein, um auf die lokalen Variablen aus der Bausteinschnittstelle zuzugreifen.
   - Geben Sie " ein, um auf die globalen Variablen zuzugreifen.
   - Geben Sie % ein, um auf absolute Adressen zuzugreifen.

   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur diejenigen lokalen und globalen Variablen, Datenbausteine, Multiinstanzen und Anweisungen, die an der aktuellen Position zulässig sind. Mit <Esc> können Sie die Autovervollständigung jederzeit verlassen.
2. Geben Sie weitere Buchstaben der Variablen oder Anweisung ein, die Sie einfügen möchten. Mit <Enter> oder <Tab> übernehmen Sie die Variable oder Anweisung und schließen die Autovervollständigung.
   Mit jedem eingegebenen Buchstaben wird die Autovervollständigung weiter gefiltert. Wenn es keine Variable oder Anweisung gibt, die mit den eingegebenen Buchstaben beginnt, enthält die Autovervollständigung lediglich die bisher passenden Treffer.
3. Wählen Sie die gewünschte Variable oder Anweisung aus der Liste aus.
   Falls es sich bei einer Variablen um eine strukturierte Variable, einen Datenbaustein oder eine Multiinstanz handelt, wählen Sie zunächst die Variable, den Datenbaustein oder die Multiinstanz aus der Autovervollständigung aus und übernehmen Sie die Auswahl mit <Enter>. Um die weiteren Bestandteile der Struktur, des Datenbausteins oder der Multiinstanz auszuwählen, geben Sie einen Punkt ein. Anschließend öffnet sich die Autovervollständigung erneut und Sie können den nächsten Bestandteil auswählen.
4. Drücken Sie die Taste <Return>, um die Variable zu übernehmen.

Siehe auch

Grundlagen zur Autovervollständigung (Seite 7147)

Autovervollständigung in grafischen Programmiersprachen verwenden (Seite 7147)

5.11 Allgemeine Einstellungen für die PLC-Programmierung

5.11.1 Übersicht über die allgemeinen Einstellungen

Überblick

Die folgende Tabelle zeigt die allgemeinen Einstellungen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Ansicht	Operandendarstellung	Darstellung der Operanden im Programmiereditor. Sie können zwischen den folgenden Optionen wählen: • Symbolisch und absolut • Symbolisch • Absolut (nur für KOP, FUP, AWL und GRAPH)
	Variableninformationen	Im Programmiereditor werden zusätzliche Informationen für die verwendeten Variablen angezeigt. Wenn Sie die Option "Variableninformationen mit Hierarchie" auswählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt.
	Position der Variableninformation (KOP/FUP)	Legt die Position fest, an der die Variableninformationen angezeigt werden sollen. Sie können die Variableninformationen entweder gesammelt unterhalb eines KOP-/FUP-Netzwerks anzeigen lassen oder direkt am jeweiligen Operanden.
	mit Netzwerkkommentaren	Netzwerkkommentare werden angezeigt.
Übersetzung	Aktualparameter löschen beim Schnittstellenabgleich	Aktualparameter werden gelöscht, wenn der zugehörige Formalparameter im aufgerufenen Baustein gelöscht wurde und Sie die Funktion "Bausteinaufruf aktualisieren" ausführen oder den Baustein übersetzen.

Gruppe	Einstellung	Beschreibung
Voreinstellung für neue Bausteine	IEC-Prüfung für Codebausteine	Die Kompatibilität von Operanden in Vergleichsoperationen und arithmetischen Operationen wird nach den Regeln der IEC geprüft. Nicht kompatible Operanden müssen Sie explizit konvertieren. Siehe auch: Übersicht über die gültigen Datentypen (Seite 243)
	"DB erreichbar aus OPC UA" für neue Datenbausteine setzen	Bei aktiviertem Optionskästchen wird das Attibut „DB erreichbar aus OPC UA“ für neue globale Datenbausteine gesetzt. Damit ist der Datenbaustein als ganzes Objekt aus OPC UA erreichbar. Die einzelnen Variablen des Datenbaustein können anschließend individuell für OPC UA freigegeben oder verriegelt werden. Siehe auch: AUTOHOTSPOT
	Instanz-Datenbausteinelemente mit der Zugriffsart "Standard" remanent setzen.	Bei aktiviertem Optionskästchen wird das Attribut „Remanent“ für neu angelegte Variablen gesetzt. Die Einstellung wirkt sich in folgenden Arten von Datenbausteinen aus: • S7-1200/S7-1500: Instanz-Datenbausteine ohne optimiertem Zugriff - die Einstellung wirkt auf alle Variablen des Datenbausteins. • S7-1200/S7-1500: Instanz-Datenbausteine mit optimiertem Zugriff - die Einstellung wirkt auf alle Variablen, mit dem Attribut "Remanenz: im IDB setzen". **Hinweis** Bei anderen Arten von Datenbausteinen können Sie keine Voreinstellung für die Remanenz festlegen. • S7-300/S7-400: Variablen in Instanz-Datenbausteinen und globalen Datenbausteinen sind per Voreinstellung remanent. Wenn Sie die Remanenzeinstellung ändern, wirkt sich die Änderung auf alle Variablen im Baustein aus. • S7-1200/S7-1500: Variablen in Instanz-Datenbausteinen mit optimiertem Zugriff sind per Voreinstellung nicht remanent. Sie

Gruppe	Einstellung	Beschreibung
		können die Remanenz für jede Variable individuell setzen. Siehe auch: Remanenz einstellen (Seite 7338)
Weitere Einstellungen	Netzwerktitel automatisch setzen	Setzt den Titel eines Netzwerks anhand des Kommentars des Ausgangsparameters der ersten schreibenden Anweisung im Netzwerk. Siehe auch: Netzwerktitel einfügen (Seite 7435)
	Autovervollständigung anzeigen	Autovervollständigung wird angezeigt.
	Mnemonik	Deutsche oder internationale Darstellung der Operationen und Operanden
Laden ohne Reinitialisierung	Speicherreserve	Definiert die Größe der Reserve im Standardspeicher, die für Schnittstellenerweiterungen genutzt werden kann. Siehe auch: Bausteine laden für S7-1200/1500 (Seite 7871)
Bausteinschnittstelle/Datenbausteinelemente	"Erreichbar aus HMI/OPC UA" für neue Elemente und ARRAY-Datenbausteinelemente setzen	Aktiviert die Option "Erreichbar aus HMI" für neue Variablen in der Bausteinschnittstelle und in Datenbausteinen. Die Verwendung dieser Option ist besonders beim Arbeiten mit großen Datenmengen in ARRAY-Datenbausteinen sinnvoll. Siehe auch: Grundlagen zur Programmierung von Datenbausteinen (Seite 7319)
	"Schreibbar aus HMI/OPC UA" für neue Elemente und ARRAY-Datenbausteinelemente setzen	Bei aktiviertem Optionskästchen wird das Attibut "Schreibbar aus HMI/OPC UA“ für neue Elemente in der Bausteinschnittstelle und in Datenbausteinen gesetzt. Die Verwendung dieser Option ist besonders beim Arbeiten mit großen Datenmengen in ARRAY-Datenbausteinen sinnvoll. Siehe auch: Grundlagen zur Programmierung von Datenbausteinen (Seite 7319)

Siehe auch

Übersicht über die Bausteinschnittstelle (Seite 7263)

Einstellungen ändern (Seite 7153)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

Grundlagen zum Bausteinzugriff (Seite 55)

IEC-Prüfung aktivieren oder deaktivieren (Seite 509)

Autovervollständigung verwenden (Seite 7147)

5.11.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die allgemeinen Einstellungen (Seite 7150)

5.12 Syntaxfehler im Programm beheben

5.12.1 Grundlagen zu Syntaxfehlern

Syntaxfehler

Als Syntaxfehler werden z. B. folgende Fehler bezeichnet:

- Fehlende Trennzeichen oder Verwendung von zu vielen Trennzeichen
- Falsche Schreibweise von Schlüsselwörtern
- Falsche Schreibweise von Sprungmarken
- Nicht zur eingestellten Mnemonik passende Schreibweise (z. B. "I2.3" statt "E2.3")
- Verwendung von Schlüsselwörtern als Operanden

Kennzeichnung von Syntaxfehler

Syntaxfehler werden mit roter Unterstreichung oder mit rotem Text hervorgehoben.

Durch die Kennzeichnung können Sie fehlerhafte Eingaben auf einen Blick erkennen und von Fehler zu Fehler springen, um sie zu beheben. Zusätzlich werden Syntaxfehler im Inspektorfenster im Register "Info" mit einer Fehlermeldung aufgelistet.

Siehe auch

Syntaxfehler im Programm finden (Seite 7154)

5.12.2 Syntaxfehler im Programm finden

Vorgehen

Um die Syntaxfehler im Programm zu finden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Position im Programm, ab der Sie nach Fehlern suchen möchten.
2. Klicken Sie in der Funktionsleiste auf "Gehe zu nächstem Fehler".
   Die erste fehlerhafte Stelle nach der selektieren Position wird markiert.

Sie können über die Funktionsleiste mit "Gehe zu nächstem Fehler" und "Gehe zu vorherigem Fehler" alle fehlerhaften Stellen im Baustein auffinden und korrigieren.

Oder:

1. Öffnen Sie die Fehlerliste im Inspektorfenster unter "Info > Syntax".
   Alle Syntaxfehler werden in der Tabelle mit einer kurzen Fehlerbeschreibung aufgelistet.
2. Wenn vorhanden, klicken Sie auf das blaue Fragezeichen neben dem Fehlertext, um zusätzliche Hinweise zur Fehlerbehebung zu erhalten.
3. Doppelklicken Sie auf den Fehler, den Sie korrigieren möchten.
   Der entsprechende Fehler wird markiert.

Siehe auch

Grundlagen zu Syntaxfehlern (Seite 7154)

5.13 Programmiersprache umschalten

5.13.1 Regeln zum Umschalten der Programmiersprache

Regeln

Beachten Sie folgende Regeln, wenn Sie die Programmiersprache für einen Baustein umstellen möchten:

- Alle CPU-Familien:
  - Sie können nur ganze Bausteine umschalten. Einzelne Netzwerke können nicht umgeschaltet werden.
  - Bausteine, die in den Programmiersprachen SCL oder GRAPH programmiert werden, können Sie nicht umschalten. Bei GRAPH-Bausteinen können Sie aber bei der Sprache der Netzwerke zwischen KOP und FUP wechseln.
- S7-300/400:
  - Sie können zwischen den Programmiersprachen KOP, FUP und AWL umschalten.
  - Sie können Netzwerke in einem Baustein mit einer anderen Programmiersprache erstellen und anschließend in den gewünschten Baustein kopieren.
  - Einzelne Netzwerke des Bausteins, die nicht umgeschaltet werden können, werden in der Originalsprache angezeigt.
- S7-1200/1500:
  - Sie können zwischen den Programmiersprachen KOP und FUP umschalten.
- S7-1500:
  - Sie können innerhalb von KOP- und FUP-Bausteinen AWL-Netzwerke erstellen. Das Kopieren zwischen AWL und KOP/FUP ist jedoch nicht möglich.

5.13.2 Programmiersprache umschalten

Vorgehen

Um die Programmiersprache zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
4. Wählen Sie in der Klappliste "Sprache" die neue Programmiersprache aus.
5. Bestätigen Sie Ihre Auswahl mit "OK".

Siehe auch

Regeln zum Umschalten der Programmiersprache (Seite 7156)

Software Units verwenden

6

6.1 Grundlagen zu Software Units

6.1.1 Einführung zu Software Units

Programmieren mit Software Units

Mithilfe von Software Units haben Sie die Möglichkeit, Ihr Anwenderprogramm in einzelne Programmeinheiten zu unterteilen, die Sie unabhängig voneinander bearbeiten und laden können. Dazu erhalten Sie in der Projektnavigation einen neuen Ordner "Software Units", in dem Sie Ihre Software Units erstellen und programmieren können.

Das folgende Bild zeigt den Ordner "Software Units" in der Projektnavigation:

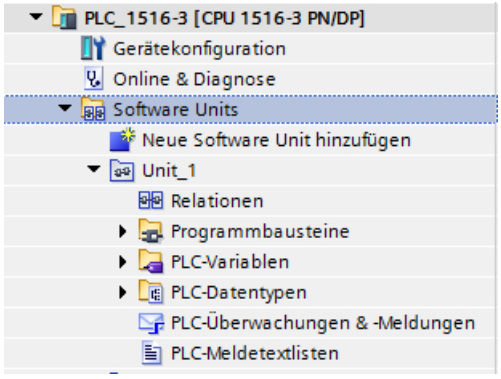

Jede Software Unit enthält die folgenden Hauptelemente:

- Relationen: Über die Relationentabelle können Sie Zugriffe aus Ihrer Software Unit heraus auf folgende Objekte einrichten:
  - Veröffentlichte Bausteine und PLC-Datentypen anderer Software Units
  - Globale Datenbausteine außerhalb der Software Unit
  - Technologieobjekte
- Ordner "Programmbausteine": In diesem Ordner erstellen Sie die Programmbausteine (Organisationsbausteine, Funktionsbausteine, Funktionen, Datenbausteine).
- Ordner "PLC-Variablen": In diesem Ordner erstellen Sie PLC-Variablentabellen, PLC-Variablen und Anwenderkonstanten.
- Ordner "PLC-Datentypen": In diesem Ordner erstellen Sie die PLC-Datentypen.
- PLC-Überwachungen & -Meldungen: Öffnet den Meldungs- und Überwachungseditor
- PLC-Meldetextlisten: Öffnet den Textlisten-Editor

Das Erstellen des Anwenderprogramms innerhalb von Software Units ist eine Erweiterung der bisherigen Programmiermöglichkeiten. Das bedeutet, dass Sie ganz flexibel entscheiden können, welche Programmteile Sie innerhalb und welche Sie außerhalb von Software Units erstellen möchten.

Hinweis

Beachten Sie bei der Verwendung von Software Units die folgenden Hinweise:

- Sie können Software Units mit allen Ausprägungen der S7-1500-CPU mit einem Firmware-Stand ab V2.6 verwenden.
- Pro CPU können Sie bis zu 255 Software Units verwenden.
- Für alle Programmbausteine einer Software Unit ist das Attribut "Optimierter Bausteinzugriff" aktiviert und kann nicht deaktiviert werden.
- Wenn Sie für ein Gerät eine Online-Sicherung erstellt haben, das Software Units unterstützt, können Sie diese Sicherung nicht für ein Gerät verwenden, das keine Software Units unterstützt.
- Die Programminformationen enthalten Informationen über alle Bausteine, unabhängig davon, ob sie in einer Software Unit erstellt wurden oder nicht.
- Die Datenbausteine und Variablen aus Software Units sind auch in einem Geräte-Proxy sichtbar und können somit exportiert und importiert werden. Dabei spielt es keine Rolle, ob die Datenbausteine veröffentlicht oder unveröffentlicht sind.
- OPC UA-Client-Schnittstelle: Konfigurations-Datenbausteine, die aus OPC UA-Client-Schnittstellen vom System generiert werden ("<Name der Client-Schnittstelle_Configuration>"), werden immer außerhalb der Software Units erstellt. Wenn Sie darauf zugreifen möchten, können Sie eine Relation zu diesen Client-Schnittstellen anlegen. Dabei verwenden Sie als Relationstyp die Option "Datenbaustein außerhalb der Software Unit".

Programmstruktur

Unabhängig davon, ob Sie ein bereits bestehendes Programm in Software Units aufteilen möchten oder ein neues Projekt beginnen, sollten Sie zu Beginn die Programmstruktur planen. Eine sinnvolle Aufteilung Ihres Anwenderprogramms in Software Units kann z. B. anhand der folgenden Kriterien erfolgen:

- Funktionale und technologische Einheiten
  In Software Units können Sie funktionale oder technologische Einheiten bilden, die Sie unabhängig voneinander bearbeiten und laden können. Falls die Bausteine eines vorhandenen Programms bereits in Gruppen unterteilt waren, können diese Gruppen möglicherweise sinnvolle Software Units ergeben.
- Zu erwartende Änderungsfrequenz
  Eine andere Möglichkeit zur Aufteilung ist die zu erwartende Änderungsfrequenz von Programmteilen. Damit möglichst voneinander unabhängig ladbare Einheiten entstehen, erstellen Sie eigene Software Units für Programmteile mit unterschiedlichen Änderungsintervallen. So können Sie Bausteine mit Basisfunktionen, die selten geändert werden müssen, in einer Software Unit erstellen und Bausteine, die für die Inbetriebnahme häufig geändert werden müssen, in anderen Software Units. Diese Aufteilung kann auch die Fehlersuche erleichtern.

- Bearbeiter
  Wenn Sie Ihr Programm nach Bearbeitern aufteilen, kann dies den Abstimmungsaufwand erheblich reduzieren. Um die Zusammenarbeit der Projektbearbeiter weiter zu optimieren, verwenden Sie Software Units in Verbindung mit Multiuser Engineering.
- Weitestmögliche Kapselung der Software Units
  Damit Sie die Software Units unabhängig voneinander bearbeiten und laden können, sollten möglichst keine Querbeziehungen zwischen den Software Units untereinander bestehen. Setzen Sie daher veröffentlichte Programmelemente und Relationen nur dann ein, wenn dies wirklich notwendig ist. Versuchen Sie so, die einzelnen Software Units möglichst zu kapseln.

Sie können die einzelnen Kriterien selbstverständlich auch mischen oder eigene Kriterien für die Aufteilung erstellen. Allerdings sollten Sie dabei die Kapselung der Software Units als ein Hauptkriterium ansehen, um den größtmöglichen Nutzen aus den Software Units zu ziehen.

Siehe auch: Mögliche Programmstrukturen mit Software Units (Seite 7165)

Software Units für die gemeinsame Inbetriebnahme nutzen

Durch die Strukturierung Ihres Anwenderprogramms in Software Units vereinfachen Sie auch die Inbetriebnahme des Anwenderprogramms auf einer PLC durch mehrere Anwender. Eine solche Strukturierung und die Aufteilung in unterschiedliche Verantwortungsbereiche ermöglicht, dass Änderungen an verschiedenen Software Units unabhängig voneinander auf die PLC geladen werden können. Hierbei wird vom TIA Portal sichergestellt, dass Änderungen von anderen Anwendern nicht überschrieben werden. In den meisten Fällen entfällt dadurch die Synchronisation des Anwenderprogramms über die PLC und der Ladevorgang auf die PLC wird beschleunigt. Beachten Sie dabei die Voraussetzungen für das Laden von Software Units.

Weitere Informationen finden Sie in den folgenden Kapiteln:

- Software Units laden (Seite 7193)
- Einführung in Multiuser Commissioning

Datenzugriffe

In Software Units können Sie die folgenden Programmelemente anlegen:

- Bausteine (OBs, FBs, FCs und DBs)
- PLC-Variablentabellen und PLC-Variablen
- PLC-Datentypen

Um von außerhalb der eigenen Software Unit auf diese Programmelemente zugreifen zu können, müssen sie veröffentlicht werden. Dies ist jedoch nicht für Organisationsbausteine und PLC-Variablen möglich. Auf sie kann nur innerhalb der Software Unit zugegriffen werden, in der sie definiert sind.

In Beobachtungs- und Forcetabellen können Sie auf alle Daten der PLC zugreifen, unabhängig von ihrer Position oder ihres Veröffentlichungsstatus.

Um auf die FBs, FCs, globale DBs und PLC-Datentypen von anderen Software Units zuzugreifen, müssen diese veröffentlicht sein und es muss zusätzlich eine Relation zu der entsprechenden Software Unit bestehen. Über Relationen können Sie auch auf globale

Datenbausteine und Technologieobjekte zugreifen, die außerhalb von Software Units vorhanden sind.

Das folgende Bild zeigt die möglichen Datenzugriffe zwischen Programmen innerhalb und außerhalb von Software Units:

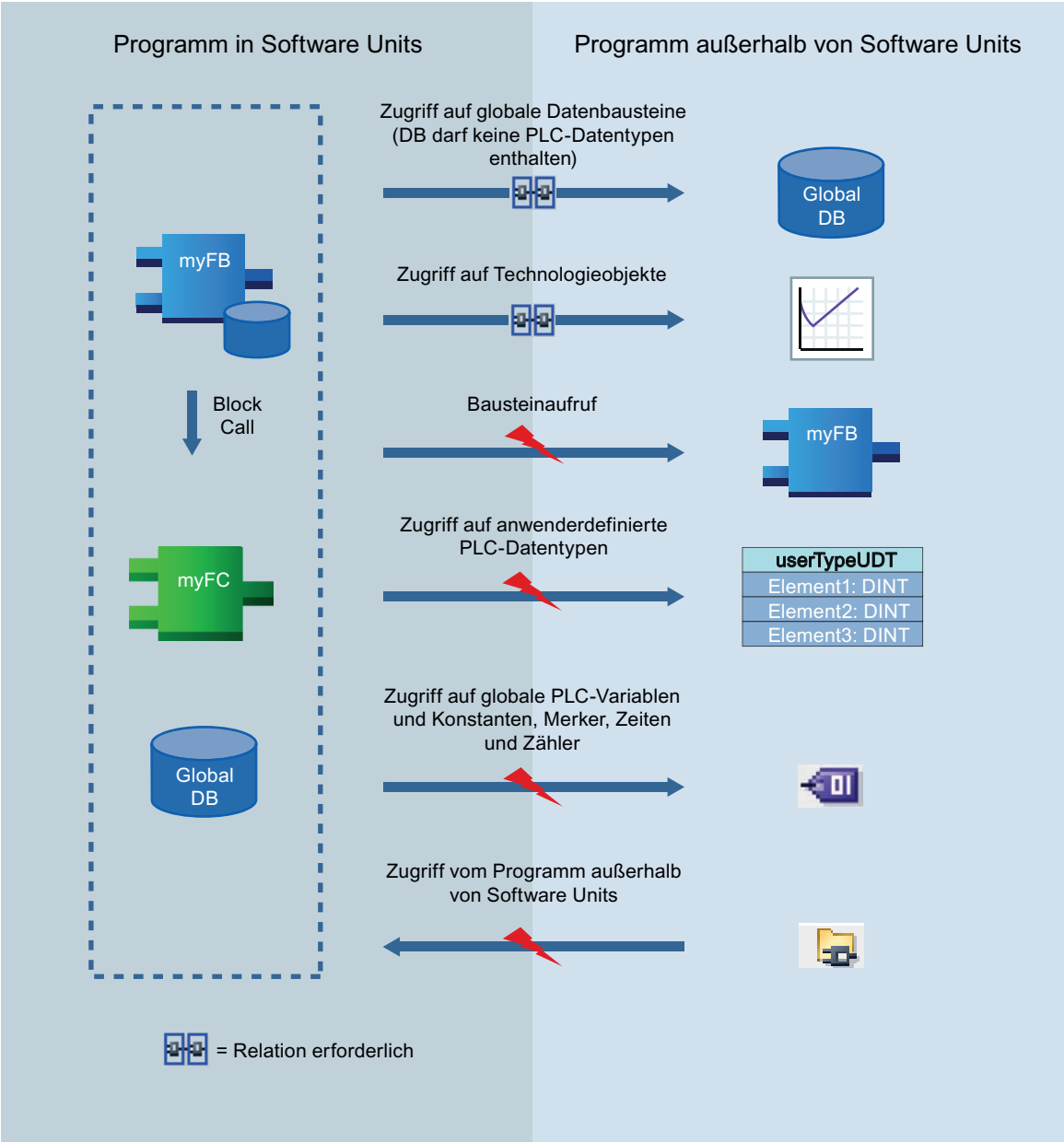

Siehe auch

PLC-Programm in Software Units erstellen (Seite 7163)

6.1.2 PLC-Programm in Software Units erstellen

Die allgemeinen Programmierkonzepte bleiben bei der Programmerstellung in Software Units gleich. Sie erstellen Ihr Anwenderprogramm wie gewohnt im Programmiereditor. Dabei haben Sie die gleiche Unterstützung wie außerhalb von Software Units, z. B. Drag & Drop von Anweisungen aus der Task Card "Anweisungen" und die Autovervollständigung beim Eingeben von Datentypen oder Operandennamen.

Besonderheiten bei der Programmerstellung in Software Units

Beachten Sie bei der Programmierung innerhalb von Software Units die folgenden Punkte:

- Für alle Programmbausteine einer Software Unit ist das Attribut "Optimierter Bausteinzugriff" aktiviert und kann nicht deaktiviert werden. Das bedeutet auch, dass beim Verschieben von Bausteinen von außerhalb von Software Units in eine Software Unit dieses Attribut aktiviert sein muss. Verwenden Sie daher auch keine Anweisungen, deren Parameter eine absolute Adressierung benötigen, wie z. B. "PUT" oder "GET".
- Programmteile, die Sie außerhalb einer Software Unit erstellen, haben keinen Zugriff auf die Programmteile, die Sie innerhalb einer Software Unit erstellen. Und umgekehrt können Bausteine, die Sie innerhalb einer Software Unit programmiert haben, nicht auf Bausteine außerhalb von Software Units zugreifen.
- Sie können jedoch gezielt einer Software Unit Zugriff auf bestimmte Programmelemente außerhalb von Software Units erlauben. So können Sie z. B. auf globale Datenbausteine und Technologieobjekte zugreifen, die außerhalb von Software Units existieren.
- Der Programmcode innerhalb der Software Units ist Teil des gesamten Anwenderprogramms. Daher dürfen z. B. Bausteinnamen CPU-weit nur einmal vorkommen.
- Innerhalb von Software Units können Sie nur symbolisch programmieren.
- Innerhalb von Software Units können Sie keine fehlersicheren Bausteine (F-Bausteine) verwenden.
- Mit Ausnahme von Organisationsbausteinen (OBs) werden alle Bausteine automatisch nummeriert. Die automatisch vergebenen Bausteinnummern werden nicht angezeigt, Sie können sie jedoch in den Eigenschaften eines Bausteins sehen.
- Die Parametertypen "TIMER", "COUNTER", "BLOCK_FB" und "BLOCK_FC" sind in der Bausteinschnittstelle nicht zulässig.
- Merker, die Zeitfunktion "Timer" und die Zählfunktion "Counter" sind in der PLC-Variablentabelle nicht zulässig. Verwenden Sie stattdessen IEC-Zeiten und -Zähler.
- Sie können keine System- und Taktmerker verwenden. Im Siemens Industry Online Support stehen Ihnen fertige Funktionen zur Verfügung, die Sie alternativ verwenden können.
  Siehe auch: https://support.industry.siemens.com/cs/ww/de/view/109479728 (https://support.industry.siemens.com/cs/ww/de/view/109479728)
- Software Units können getrennt bearbeitet und geladen werden. Falls dadurch Bausteine der gleichen Software Unit von mehreren Bearbeitern geändert werden, erhalten Sie beim Laden die Möglichkeit, Ihre Daten zu synchronisieren.

- Systemkonstanten werden in der Software Unit erzeugt, in der sich das zugehörige Objekt befindet. Sie können nicht veröffentlicht werden, dadurch ist kein Zugriff von außerhalb der gleichen Software Unit möglich.
- Beim Verwenden von Anweisungen, die systemintern als Funktion oder Funktionsbaustein umgesetzt sind, werden diese Systemfunktionen bzw. Systemfunktionsbausteine im Ordner "Systembausteine" außerhalb der Software Units erstellt. Die Einzel-Instanz des Datenbausteins eines Systemfunktionsbaustein wird dagegen unveröffentlicht im Ordner "Systembausteine" der entsprechenden Software Unit angelegt. Wenn Sie den Datenbaustein verschieben und darauf zugreifen möchten, ist es möglicherweise erforderlich, dass Sie den Zugriff auf "Veröffentlicht" ändern und eine Relation zu diesem Datenbaustein erstellen.
- Die Verwendung von Software Units ändert nichts an den Zugriffsmöglichkeiten für Variablen über HMI bzw. OPC UA. Diese defnieren Sie weiterhin in der Bausteinschnittstelle für lokale Variablen oder in der PLC-Variablentabelle für globale Variablen.

Programmiersprachen

Sie können in Software Units die folgenden Programmiersprachen verwenden:

- KOP
- FUP
- SCL
- GRAPH
- PRODIAG

In den Programmiersprachen KOP, FUP, SCL und GRAPH gibt es keine Unterschiede zur Verwendung außerhalb von Software Units. Beachten Sie bei PRODIAG jedoch die folgenden Besonderheiten bei der Verwendung innerhalb von Software Units:

- Anwenderdefinierte Textlisten, die in überwachungsspezifischen Texten referenziert werden, können Sie in einer beliebigen Software Unit ablegen, jedoch nicht außerhalb von Software Units.
- Jede Software Unit hat einen eigenen ProDiag-OB, welchem eine Nummer >= 250 zugeordnet wird.
- Die globalen Variablenüberwachungen an einem ProdDiag-FB können nur Operanden derselben Software Unit überwachen.
- Ein ProDiag-FB enthält nur Bausteinüberwachungen, die sich auf Parameter aus derselben Software Unit beziehen.
- Sie können ein ProDiag-FB nicht veröffentlichen, daher ist ein Zugriff von anderen Software Units aus nicht möglich.
- Die Erstwerterfassung für einen ProDiag-FB berücksichtigt bei globalen Überwachungen und Aufrufen nur schreibende Zugriffe innerhalb der Software Unit, in welcher sich der ProDiag-FB befindet.

PLC-Meldetextlisten

Jede Software Unit enthält in der Projektnavigation einen Eintrag "PLC-Meldetextlisten". Das heißt, Sie können für jede Software Unit gezielt die entsprechenden Textlisten anlegen. Textlisten sind immer veröffentlicht, Sie können daher von anderen Software Units aus darauf zugreifen, ohne zuvor Relationen angelegt zu haben. Ein Zugriff von außerhalb von Software Units ist dagegen nicht möglich.

In jeder Software Unit werden nur die eigenen Textlisten angezeigt. Um Textlisten zu erstellen oder zu löschen, müssen Sie jeweils den Textlisten-Editor der entsprechenden Software Unit oder der CPU öffnen. Textlistennamen dürfen CPU-weit nur einmal vorkommen.

PLC-Überwachungen & -Meldungen

Jede Software Unit enthält einen eigenen ProDiag-Übersichtseditor, den Sie in der Projektnavigation über den Eintrag "PLC-Überwachungen & -Meldungen" öffnen können. In diesem Editor können Sie für eine Software Unit Überwachungen hinzufügen. Die Überwachungen, die Sie in einer Software Unit anlegen, werden nicht in anderen Software Units angezeigt. Der ProDiag-Übersichtseditor außerhalb von Software Units zeigt hingegen alle Überwachungen einer PLC an, also auch die Überwachungen innerhalb der Software Units.

Sie können die angelegten Überwachungen in eine Excel-Tabelle exportieren und danach wieder in das TIA Portal importieren. Wenn Sie den Export im ProDiag-Übersichtseditor einer Software Unit durchführen, werden nur die Überwachungen dieser Software Unit exportiert. Um alle Überwachungen einer PLC zu exportieren, starten Sie den Exportvorgang im ProDiag-Übersichtseditor außerhalb der Software Units. Die Excel-Tabelle erhält eine zusätzliche Spalte, in der die Software Unit angegeben wird, in der sich die Überwachung befindet.

Siehe auch:

- ProDiag-Überwachungen anzeigen (Seite 8276)
- ProDiag-Überwachungen und Eigenschaften eines ProDiag-FB exportieren und importieren (Seite 8287)

Siehe auch

Einführung zu Software Units (Seite 7159)

Mögliche Programmstrukturen mit Software Units (Seite 7165)

6.1.3 Mögliche Programmstrukturen mit Software Units

Im Folgenden wird anhand von zwei stark vereinfachten Beispielen gezeigt, wie Sie ein Anwenderprogramm in Software Units strukturieren und dabei Querbezüge zwischen den einzelnen Software Units weitestgehend reduzieren können.

Beide Beispielprogramme steuern eine Produktionsanlage, die aus zwei Stationen besteht. Die Stationen führen verschiedene Bearbeitungsschritte an Werkstücken durch und erledigen standardisierte Aufgaben, z. B. das Addieren der produzierten Stückzahlen.

Programmstruktur - Beispiel 1

Im ersten Beispiel erledigt die zentrale Unit "UnitMain" zentrale Steuerungsaufgaben. Sie enthält einen Organisationsbaustein zur zyklischen Programmbearbeitung und einen Funktionsbaustein "LineControl", der die beiden Stationen koordiniert.

Die beiden Stationen werden in je einer Software Unit abgebildet. Die Programmteile in den beiden Software Units sind so weit gekapselt, dass sie von zwei verschiedenen Programmierern bearbeitet werden können.

Die "UnitStandard" hat die Funktion einer Bibliothek, die allgemeine Funktionalität zur Verfügung stellt. Im Beispiel enthält sie den FB "PartCount" zum Zählen der Stückzahlen. Dieser FB wird von jeder Station einmal instanziiert. In der Praxis wird eine solche "Standard-Software Unit" möglicherweise eine große Zahl mehrfach zu verwendender Programmteile enthalten, die beliebig aufgerufen werden können.

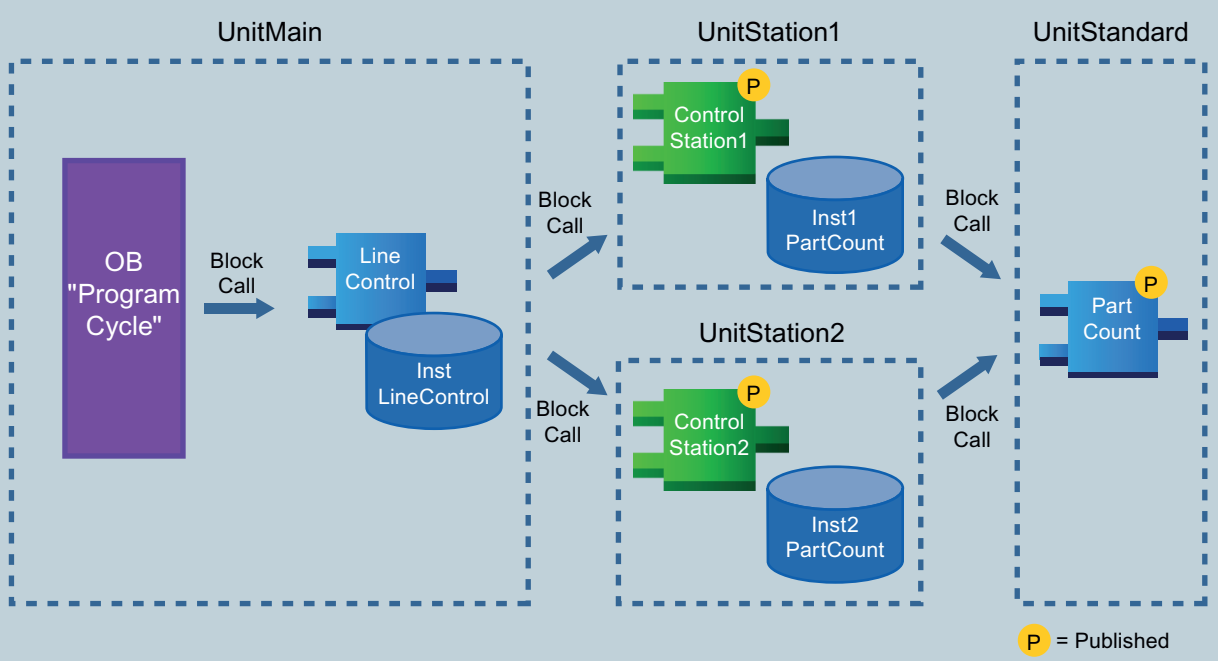

Erforderliche Relationen

Relationsquelle	Relationsziel
UnitMain	UnitStation1
	UnitStation2
UnitStation1	UnitStandard
UnitStation2	

Programmstruktur - Beispiel 2

Auch im zweiten Beispiel wird jede Station gekapselt in einer eigenen Software Unit gesteuert. Im zweiten Beispiel hat jedoch jede Station einen eigenen zyklischen Organisationsbaustein. Wenn ein Programm mehrere Organisationsbausteine enthält, werden diese vom Betriebssystem nacheinander, in der Reihenfolge ihrer OB-Nummern, aufgerufen.

Wie im ersten Beispiel gibt es auch hier die "UnitStandard", die als Bibliothek für mehrfach verwendete Programmteile genutzt wird. Hier enthält sie neben der FB "PartCount" auch einen PLC-Datentyp (UDT), der die Produktionsdaten eines Werkstücks abbildet. Der PLC-Datentyp kann in der Schnittstelle jedes Programmbausteins verwendet werden und ermöglichst so die standardisierte Ablage der Werkstückdaten.

Zur zentralen Datenablage und -verarbeitung wird "UnitGlobalMachineData" verwendet. Sie enthält einen globalen Datenbaustein, in den jede Software Unit ihre Werkstückdaten ablegen kann.

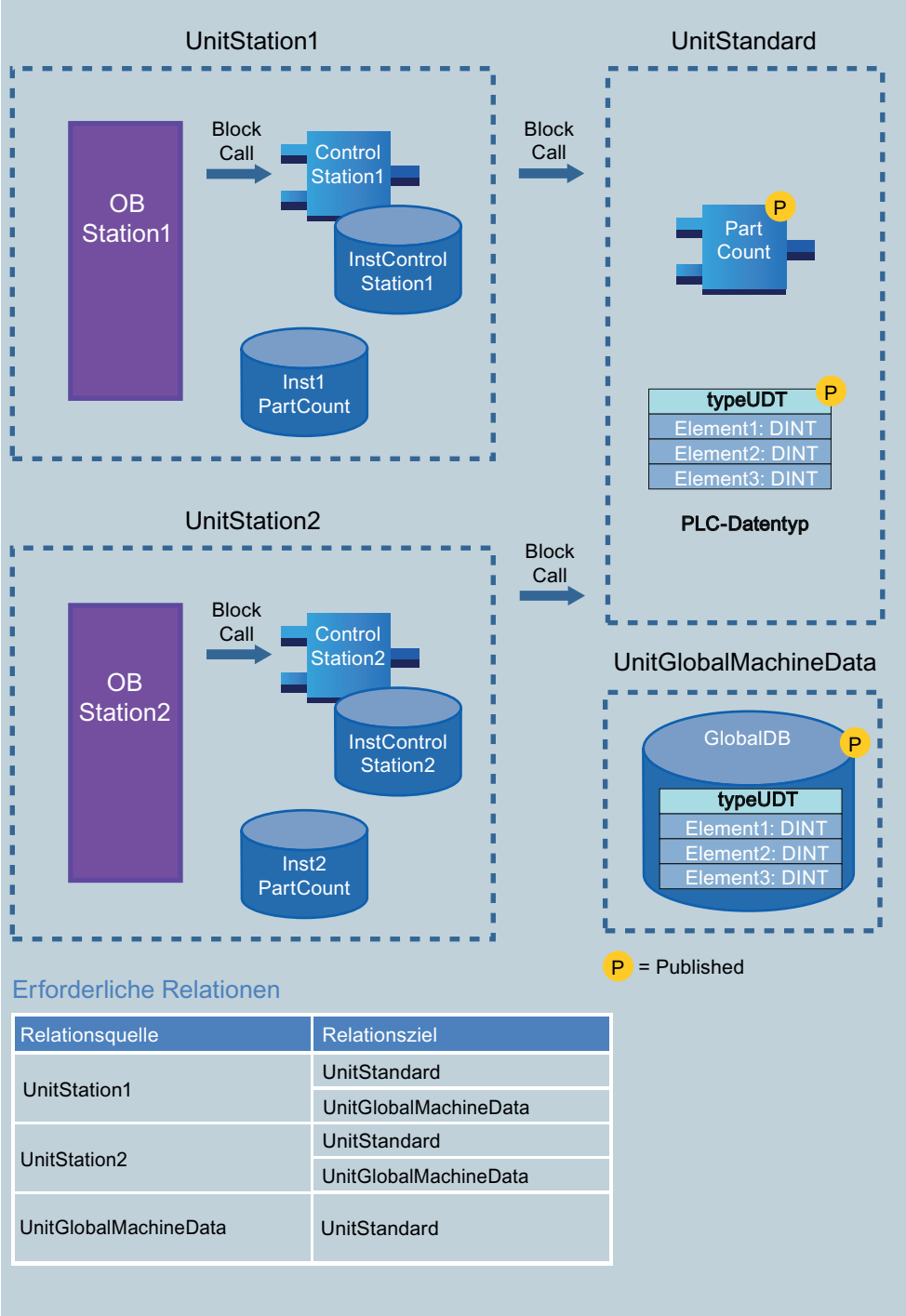

Erforderliche Relationen

Relationsquelle	Relationsziel
UnitStation1	UnitStandard
	UnitGlobalMachineData
UnitStation2	UnitStandard
	UnitGlobalMachineData
UnitGlobalMachineData	UnitStandard

Siehe auch

Einführung zu Software Units (Seite 7159)

PLC-Programm in Software Units erstellen (Seite 7163)

6.2 Software Units anlegen und verwalten

6.2.1 Software Units anlegen

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Vorgehen

Um eine neue Software Unit anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Eintrag "Neue Software Unit hinzufügen".
   Der Dialog "Neue Software Unit hinzufügen" wird geöffnet.
2. Geben Sie einen Namen für die neue Software Unit ein.
3. Um weitere Eigenschaften für die neue Software Unit einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
4. Geben Sie alle gewünschten Eigenschaften ein.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die neue Software Unit wird erzeugt. Sie können der Software Unit nun weitere Programmelemente hinzufügen und die notwendigen Relationen erstellen.

Siehe auch

Einführung zu Software Units (Seite 7159)

Software Units umbenennen (Seite 7169)

Software Units kopieren und einfügen (Seite 7170)

Software Units löschen (Seite 7171)

Eigenschaften einer Software Unit anzeigen und ändern (Seite 7172)

6.2.2 Software Units umbenennen

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Vorgehen

Um eine Software Unit umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Software Unit, die Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
   In der Projektnavigation wird der Name der Software Unit in ein Eingabefeld geändert.
3. Geben Sie den neuen Namen der Software Unit ein.
4. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.

Ergebnis

Der Name der Software Unit wird an allen Verwendungsstellen im Programm geändert.

Siehe auch

Software Units anlegen (Seite 7169)

Software Units kopieren und einfügen (Seite 7170)

Software Units löschen (Seite 7171)

Eigenschaften einer Software Unit anzeigen und ändern (Seite 7172)

6.2.3 Software Units kopieren und einfügen

Sie können neue Software Units auch dadurch anlegen, dass Sie bestehende Software Units kopieren und die Kopie einfügen. Sie haben dabei die folgenden Möglichkeiten:

- Software Unit kopieren und innerhalb der gleichen CPU einfügen
- Software Unit kopieren und in eine andere CPU innerhalb des gleichen Projekts einfügen
- Software Unit eines Referenzprojekts kopieren und in das geöffnete Projekt einfügen
- Software Unit aus einer TIA Portal-Instanz kopieren und in eine andere TIA Portal-Instanz einfügen

Beim Einfügen von kopierten Software Units mit identischen Namen wie bereits bestehende Software Units kommt es zu Namenskonflikten, die folgendermaßen behandelt werden:

- Einfügen der kopierten Software Unit in die gleiche CPU:
  Die Kopie der Software Unit erhält einen Namen, der um eine Nummer erweitert ist. Wenn z. B. Software Unit "SoftwareUnit_1" kopiert wird, ist ein möglicher Name der Kopie "SoftwareUnit_2". Dabei wird als Nummer kein fortlaufendes Nummernband verwendet, sondern immer die kleinste freie Nummer. Die Kopie von Software Unit "SoftwareUnit_1" kann also auch den Namen "SoftwareUnit_25" erhalten, falls keine niedrigere Nummer frei ist.
- Einfügen der kopierten Software Unit in eine andere CPU:
  Ein Dialog wird geöffnet, in dem Sie auswählen können, ob die Software Unit mit dem gleichen Namen ersetzt oder ob die kopierte Software Unit mit einem neuen Namen eingefügt werden soll.

Software Units kopieren

Um eine Software Unit zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Software Unit, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
   Eine Kopie der Software Unit befindet sich nun in der Zwischenablage und kann entweder in die gleiche oder eine andere CPU eingefügt werden.

Software Units einfügen

Um eine Software Unit einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Ordnerstruktur für die CPU, in die Sie die kopierte Software Unit einfügen möchten.
2. Klicken Sie mit der rechten Maustaste auf den Ordner "Software Units".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   - Falls Sie die Software Unit in die gleiche CPU einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   - Falls Sie die Software Unit in eine andere CPU einfügen und es eine Software Unit mit diesem Namen schon gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Siehe auch

Software Units anlegen (Seite 7169)

Software Units umbenennen (Seite 7169)

Software Units löschen (Seite 7171)

Eigenschaften einer Software Unit anzeigen und ändern (Seite 7172)

6.2.4 Software Units löschen

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Vorgehen

Um eine Software Unit zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Software Units" mit der rechten Maustaste auf die Software Unit, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".
   Die Software Unit wird aus dem Projekt gelöscht.

Siehe auch

Software Units anlegen (Seite 7169)

Software Units umbenennen (Seite 7169)

Software Units kopieren und einfügen (Seite 7170)

Eigenschaften einer Software Unit anzeigen und ändern (Seite 7172)

6.2.5 Eigenschaften einer Software Unit anzeigen und ändern

Sie können für Software Units die folgenden Eigenschaften anzeigen und ändern:

- Name
- Autor
- Kommentar

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Vorgehen

Um die Eigenschaften einer Software Unit anzuzeigen und zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Software Unit, deren Eigenschaften Sie anzeigen oder bearbeiten möchten.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet und die Eigenschaften werden angezeigt.
3. Ändern Sie bei Bedarf die gewünschte Eigenschaft.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Siehe auch

Software Units anlegen (Seite 7169)

Software Units umbenennen (Seite 7169)

Software Units kopieren und einfügen (Seite 7170)

Software Units löschen (Seite 7171)

6.2.6 Software Unit-Elemente in Bibliotheken verwalten

Sie haben die Möglichkeit, Software Units in Bibliotheken zu verwalten, um sie zusammen mit ihren enthaltenen Programmelementen bei Bedarf wiederverwenden zu können.Software Units können Sie als Kopiervorlagen in die Projektbibliothek oder in eine globale Bibliothek einfügen.

Sie können beim Erstellen von Kopiervorlagen zwischen den folgenden Möglichkeiten wählen:

- Sie selektieren eine oder mehrere Software Units und erzeugen daraus separate Kopiervorlagen.
- Sie selektieren mehrere Software Untis und erzeugen daraus eine einzelne Kopiervorlage, die alle selektierten Software Units enthält.

Die in Software Units enthaltenen Bausteine und PLC-Datentypen können Sie auch unabhängig von ihrer Software Unit in die Projektblibliothek oder eine globale Bibliothek einfügen. Dabei gibt es keine Unterschiede zur Verwendung von Bausteinen und PLC-Datentypen, die nicht innerhalb von Software Units programmiert wurden. So können sie z. B. ebenfalls als Kopiervorlagen oder als Typen in der Projektbibliothek oder in globalen Bibliotheken eingefügt und dann im Projekt wiederverwendet werden. Falls der Baustein oder der PLC-Datentyp im Projekt bereits vorhanden ist, erhalten Sie eine entsprechende Meldung. In dieser Meldung können Sie auswählen, ob das schon existierende Objekt in den ausgewählten Ordner verschoben werden oder am aktuellen Ort bleiben soll.

Aus einer oder mehreren Software Units separate Kopiervorlagen erzeugen

Um aus einer oder mehreren Software Units separate Kopiervorlagen zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektbibliothek oder eine globale Bibliothek in der Task Card "Bibliotheken".
2. Selektieren Sie die gewünschten Software Units in der Projektnavigation.
3. Ziehen Sie die Software Units per Drag&Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".

Oder:

1. Selektieren Sie die gewünschten Software Units in der Projektnavigation.
2. Kopieren Sie die Software Units in die Zwischenablage und fügen Sie sie in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen" ein.

Die Software Units werden jeweils als separate Kopiervorlagen in die Projektibliothek oder globale Bibliothek eingefügt.

Aus mehreren Software Units eine einzelne Kopiervorlage erzeugen

Um aus mehreren Software Units eine einzelne Kopiervorlage zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektbibliothek oder eine globale Bibliothek in der Task Card "Bibliotheken".
2. Kopieren Sie die Software Units, die Sie als Kopiervorlagen anlegen möchten, in die Zwischenablage.

3. Klicken Sie in der Bibliothek mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner.
4. Wählen Sie im Kontextmenü den Befehl "Als einzelne Kopiervorlage einfügen".

Oder:

1. Selektieren Sie die gewünschten Software Units in der Projektnavigation.
2. Drücken Sie die Taste <Alt> und ziehen Sie die Elemente per Drag&Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".

Die Elemente werden als eine einzelne Kopiervorlage in die Bibliothek eingefügt. Die einzelne Kopiervorlage enthält alle gewählten Elemente.

6.3 Programmelemente einfügen

6.3.1 Bausteine anlegen

Sie können innerhalb einer Software Unit die folgenden Bausteinarten verwenden:

- Organisationsbausteine (OB)
- Funktionen (FC)
- Funktionsbausteine (FB)
- Datenbausteine (DB)

Um die Bausteine zu programmieren, können Sie die Programmiersprachen KOP, FUP, SCL, GRAPH und PRODIAG einsetzen.

Die Nummern für FBs, FCs und DBs werden immer automatisch vergeben und können nicht geändert werden. Die Bausteinnummern für OBs können Sie dagegen entsprechend des jeweils verfügbaren Nummernbands selbst vergeben. In einer Software Unit werden die Bausteinnummern in der Projektnavigation nicht angezeigt. Sie können aber die Eigenschaften eines Bausteins öffnen, um seine Nummer anzuzeigen.

Beachten Sie, dass alle Beschränkungen für Organisationsbausteine auch innerhalb von Software Units erhalten bleiben. Das bedeutet, dass Sie nur die maximale Anzahl eines Organisationsbausteintyps verwenden können, unabhängig davon, ob die Organisationsbausteine innerhalb oder außerhalb von Software Units vorhanden sind.

Wenn Sie auf einen Baustein einer anderen Software Unit zugreifen möchten, ist es notwendig, dass für diesen Baustein unter "Zugriff" die Option "Veröffentlicht" aktiviert ist. Beachten Sie dabei die folgenden Besonderheiten:

- Organisationsbausteine:
  Für Organisationsbausteine steht die Eigenschaft "Zugriff" nicht zur Verfügung. Auf sie kann nicht von einer anderen Software Unit aus zugegriffen werden.
- ProDiag-Funktionsbausteine:
  Für ProDiag-Funktionsbausteine ist die Option "Veröffentlicht" immer deaktiviert. Das bedeutet, dass Sie auch darauf nicht von einer anderen Software Unit aus zugreifen können.

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Vorgehen

Um in einer Software Unit einen neuen Baustein anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Software Unit, der Sie einen neuen Baustein hinzufügen möchten.
2. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
3. Klicken Sie auf die entsprechende Schaltfläche für die gewünschte Bausteinart.
4. Geben Sie einen Namen für den neuen Baustein ein.

5. Wählen Sie die Sprache für den neuen Baustein aus.
6. Aktivieren Sie das Optionskästchen "Veröffentlicht", wenn Sie von anderen Software Units aus darauf zugreifen möchten.
7. Um weitere Eigenschaften des neuen Bausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
8. Geben Sie alle gewünschten Eigenschaften ein.
9. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
10. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Der neue Baustein wird erzeugt. Sie finden den Baustein in der Projektnavigation innerhalb der entsprechenden Software Unit im Ordner "Programmbausteine".

Siehe auch

PLC-Variablen deklarieren (Seite 7176)

PLC-Datentypen anlegen (Seite 7178)

6.3.2 PLC-Variablen deklarieren

Innerhalb von Software Units können Sie auf die gleiche Art und Weise mit PLC-Variablen und globalen Konstanten arbeiten wie außerhalb von Software Units. Beachten Sie jedoch die folgenden Besonderheiten:

- Die Variablentabelle "Alle Variablen anzeigen" ist in einer Software Unit nicht verfügbar.
- Die Variablentabelle "Alle Variablen anzeigen" außerhalb der Software Units zeigt auch die Variablen und globalen Konstanten der Software Units an. Die Variablen und globalen Konstanten der Software Units können hier auch bearbeitet werden.
- Variablen und globale Konstanten einer Software Unit können nur innerhalb dieser Software Unit verwendet werden. Ein Zugriff auf die Variablen und globalen Konstanten einer Software Unit von außerhalb dieser Software Unit ist nicht erlaubt. Der Zugriff über HMI und OPC UA ist dadurch jedoch nicht eingeschränkt.
- S5-Zeiten und S5-Zähler sind als Datentypen für Operanden nicht zulässig.

- Die Verwendung von Merkern ist nicht zulässig.
- Sie können auch in Software Units PLC-Variablentabellen exportieren, um sie mit externen Applikationen weiterzubearbeiten. Ebenso können Sie PLC-Variablentabellen, die extern bearbeitet wurden, in Software Units importieren. Beim Import und Export werden folgende Formate unterstützt:
  - XLSX (Excel)
  - XML
  - SDF

  Im Format "XLSX" wird eine zusätzliche Spalte erzeugt, in der die Software Unit angegeben wird, aus der die PLC-Variablentabelle exportiert wurde. Die beiden anderen Formate enthalten keine Information über die exportierende Software Unit. Beachten Sie, dass Sie PLC-Variabeln aus einer Excel-Datei mit der zusätzlichen Spalte für die Software Unit nur in andere Software Units importieren können, nicht aber in PLC-Variabentabellen außerhalb von Software Units.

PLC-Variablentabelle anlegen

Um eine neue PLC-Variablentabelle innerhalb einer Software Unit anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" innerhalb der Software Unit, der Sie eine PLC-Variablentabelle hinzufügen möchten.
2. Doppelklicken Sie auf den Eintrag "Neue Variablentabelle hinzufügen".
   Eine neue PLC-Variablentabelle mit dem Standardnamen "Variablentabelle_x" wird angelegt.
3. Selektieren Sie die PLC-Variablentabelle in der Projektnavigation.
4. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
5. Geben Sie einen CPU-weit eindeutigen Namen ein.
   Eine neue PLC-Variablentabelle wird angelegt. In dieser können Sie Variablen und Konstanten deklarieren.

Variablen deklarieren

Um PLC-Variablen innerhalb einer Software Unit zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablentabelle, die sich innerhalb einer Software Unit befindet.
2. Öffnen Sie das Register "Variablen".
3. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
4. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
   Eine dem Datentyp entsprechende Adresse wird automatisch ergänzt.
5. Optional: Klicken Sie in der Spalte "Adresse" auf die Pfeiltaste und geben Sie im nachfolgenden Dialog ein Operandenkennzeichen, einen Operandentyp, eine Adresse und eine Bitnummer ein.

6. Optional: Deaktivieren Sie die Optionen für HMI und OPC UA.
7. Optional: Tragen Sie in der Spalte "Kommentar" einen Kommentar ein.
8. Wiederholen Sie die Schritte 3 - 7 für alle erforderlichen Variablen.

Globale Konstanten deklarieren

Um globale Konstanten innerhalb einer Software Unit zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablentabelle, die sich innerhalb einer Software Unit befindet.
2. Öffnen Sie das Register "Anwenderkonstanten".
   Die Konstantentabelle wird geöffnet.
3. Tragen Sie in der Spalte "Name" einen Konstantennamen ein.
4. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
5. Tragen Sie in der Spalte "Wert" einen Konstantenwert ein, der für den gewählten Datentyp zulässig ist.
6. Optional: Tragen Sie in der Spalte "Kommentar" einen Kommentar ein, falls Sie Anmerkungen zu der Konstante haben.
7. Wiederholen Sie die Schritte 3 - 6 für alle weiteren Konstanten, die Sie deklarieren möchten.

Ergebnis

Die PLC-Variablentabelle, die PLC-Variable oder die globale Konstante wurde angelegt und können genauso bearbeiten werden wie außerhalb einer Software Unit. Allerdings ist ein Zugriff auf die Variablen und Konstanten nur von innerhalb der eigenen Software Unit aus möglich. Sie können aber Variablen in eine Beobachtungstabelle oder Forcetabelle einfügen, um sie zu beobachten oder zu forcen.

Siehe auch

Bausteine anlegen (Seite 7175)

PLC-Datentypen anlegen (Seite 7178)

PLC-Variablen deklarieren (Seite 7382)

PLC-Variablen exportieren und importieren (Seite 7402)

6.3.3 PLC-Datentypen anlegen

Sie können innerhalb einer Software Unit ebenfalls mit PLC-Datentypen arbeiten. Wenn Sie von anderen Software Units darauf zugreifen möchten, können Sie die Eigenschaft "Veröffentlicht" setzen und eine Relation zu der entsprechenden Software Unit einrichten. Außerhalb von Software Units können Sie diesen PLC-Datentyp nicht verwenden.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Datentypen" innerhalb einer Software Unit.
2. Doppelklicken Sie im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Der Dialog "Neuen PLC-Datentyp hinzufügen" wird geöffnet.
3. Geben Sie einen Namen für den neuen PLC-Datentyp ein.
4. Aktivieren Sie das Optionskästchen "Veröffentlicht", wenn Sie von anderen Software Units aus darauf zugreifen möchten.
5. Um weitere Eigenschaften des neuen PLC-Datentyps einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
6. Geben Sie alle gewünschten Eigenschaften ein.
7. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der PLC-Datentyp nach dem Erstellen gleich geöffnet werden soll.
8. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Der PLC-Datentyp wurde angelegt und Sie können ihn genauso bearbeiten wie außerhalb einer Software Unit.

Siehe auch

Bausteine anlegen (Seite 7175)

PLC-Variablen deklarieren (Seite 7176)

PLC-Datentypen anlegen (Seite 7412)

6.3.4 Bausteinaufrufe einfügen

Sie können innerhalb von Software Units Bausteinaufrufe genauso verwenden wie außerhalb von Software Units. Beachten Sie jedoch die folgenden Besonderheiten:

- Innerhalb einer Software Unit können Sie Bausteine unabhängig von ihrem Veröffentlichungszustands aufrufen. Das heißt, dass der aufgerufene Baustein nicht veröffentlicht sein muss.
- Bei Bausteinaufrufen zwischen zwei Software Units muss zwischen den Software Units eine Relation bestehen und der aufgerufene Baustein muss veröffentlicht sein. Dabei muss der aufrufende Baustein Zugriff auf den aufgerufenen Baustein und auf den zugehörigen Instanz-Datenbaustein erhalten. Der Instanz-Datenbaustein muss ebenso Zugriff auf seinen Funktionsbaustein haben. Achten Sie dabei darauf, dass sich keine zirkulären Relationen ergeben.

- Aufrufe zwischen Bausteinen von Software Units und Bausteinen, die außerhalb von Software Units liegen, sind nicht möglich.
- Sie können einen Instanzdatenbaustein nicht veröffentlichen, wenn der zugehörige Funktionsbaustein nicht veröffentlicht ist.

Aufruf einer Funktion (FC) einfügen

Um den Aufruf einer Funktion (FC) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das gewünschte Netzwerk.

Aufruf eines Funktionsbausteins (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in den Programmcode.
   Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
     Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie im Eingabefeld "ProDiag-FB" den Überwachungen einen ProDiag-Funktionsbaustein zu.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
3. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter können Sie anschließend versorgen.

Hinweis

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt, jedoch nicht veröffentlicht. Sie können ihn bei Bedarf nachträglich veröffentlichen.

Siehe auch

Bausteinaufrufe (Seite 62)

6.3.5 Programmelemente kopieren oder verschieben

Sie können die folgenden Programmelemente kopieren oder verschieben:

- Bausteine
- PLC-Variablentabellen und Variablen
- PLC-Datentypen
- Textlisten

Dabei sind die folgenden Kopier- oder Verschiebevorgänge möglich:

- Kopieren oder Verschieben eines Programmelements von einer Software Unit zu einer anderen Software Unit
  Alle gesetzten Eigenschaften des Programmelements bleiben erhalten.
- Kopieren oder Verschieben eines Programmelements von einer Software Unit in einen Bereich außerhalb von Software Units
  Bei Bausteinen und PLC-Datentypen wird die Eigenschaft "Zugriff" entfernt. Für Bausteine ist die automatische Nummerierung eingestellt.
- Kopieren oder Verschieben eines Programmelements von einem Bereich außerhalb von Software Units in eine Software Unit
  Sie können nur Bausteine in eine Software Unit einfügen und dort bearbeiten, die automatisch nummeriert werden und für die das Attribut "Optimierter Bausteinzugriff" aktiviert ist. Bausteine, die diese Voraussetzungen nicht erfüllen, können Sie zwar einfügen, sie werden jedoch nicht unterstützt und können daher nicht bearbeitet werden. Bei unterstützten Bausteinen und PLC-Datentypen wird die Eigenschaft "Zugriff" ergänzt und das Optionskästchen "Veröffentlicht" wird deaktiviert.

Beachten Sie, dass beim Verschieben per Drag & Drop in eine andere CPU immer ein Kopieren erfolgt. Das Objekt wird dabei an der Ursprungsstelle nicht gelöscht.

Bausteine kopieren oder verschieben

Um einen Baustein zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf den Programmbausteinordner, in den Sie den kopierten Baustein einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   - Falls Sie den Baustein in die gleiche CPU einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   - Falls Sie den Baustein in eine andere CPU einfügen und es bereits einen Baustein mit diesem Namen gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Alternativ können Sie den Baustein auch per Drag & Drop mit gedrückter <Strg>-Taste kopieren.

Um einen Baustein zu verschieben, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Baustein per Drag & Drop an die neue Position.

PLC-Variablentabellen und Variablen kopieren oder verschieben

Um PLC-Variablentabellen zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PLC-Variablentabelle, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die PLC-Variablentabelle einfügen möchten.
   Sie können die PLC-Variablentabelle in die gleiche oder eine andere Software Unit einfügen oder außerhalb einer Software Unit in den Ordner "PLC-Variablen" einer CPU.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   – Falls Sie die PLC-Variablentabelle in die gleiche CPU einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   – Falls Sie die PLC-Variablentabelle in eine andere CPU einfügen und es bereits eine PLC-Variablentabelle mit diesem Namen gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Alternativ können Sie die PLC-Variablentabelle auch per Drag & Drop mit gedrückter <Strg>-Taste kopieren.

Um Variablentabellen zu verschieben, gehen Sie folgendermaßen vor:

1. Ziehen Sie die PLC-Variablentabelle an die neue Position.

Um Variablen zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable, die Sie kopieren möchten.
   Sie können auch mehrere Variablen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Variable anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die Variable einfügen möchten.
   Sie können die Variablen in jede PLC-Variablentabelle einfügen, unabhängig davon, ob sich diese in der gleichen oder einer anderen Software Unit oder außerhalb einer Software Unit befindet.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   Falls Sie die Variablen in die gleiche PLC-Variablentabelle einfügen oder in eine PLC-Variablentabelle, in der es bereits Variablen mit dem gleichen Namen gibt, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.

Alternativ können Sie die Variablen auch per Drag & Drop mit gedrückter <Strg>-Taste kopieren.

Um Variablen zu verschieben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable, die Sie verschieben möchten.
   Sie können auch mehrere Variablen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Variable anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

3. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die Variablen einfügen möchten.
   Sie können die Variablen in jede PLC-Variablentabelle einfügen, unabhängig davon, ob sich diese in der gleichen oder einer anderen Software Unit oder außerhalb einer Software Unit befindet.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   Falls Sie die Variablen in die gleiche PLC-Variablentabelle einfügen oder in eine PLC-Variablentabelle, in der es bereits Variablen mit dem gleichen Namen gibt, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.

Alternativ können Sie die Variablen auch per Drag & Drop verschieben.

PLC-Datentypen kopieren oder verschieben

Um einen PLC-Datentyp zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den PLC-Datentyp, den Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf den Ordner "PLC-Datentypen", in den Sie den kopierten PLC-Datentyp einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   - Falls Sie den PLC-Datentyp in die gleiche CPU einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   - Falls Sie den PLC-Datentyp in eine andere CPU einfügen und es bereits einen PLC-Datentyp mit diesem Namen gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Alternativ können Sie den PLC-Datentyp auch per Drag & Drop mit gedrückter <Strg>-Taste kopieren.

Um einen PLC-Datentyp zu verschieben, gehen Sie folgendermaßen vor:

1. Ziehen Sie den PLC-Datentyp per Drag & Drop an die neue Position.

Textlisten kopieren oder verschieben

Um Textlisten zu kopieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PLC-Meldetextlisten" in der CPU oder Software Unit, für die Sie eine Textliste kopieren möchten.
   Alle Textlisten, die der CPU oder Software Unit zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
2. Klicken Sie mit der rechten Maustaste auf die Textliste, die Sie kopieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Kopieren".

4. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die Textliste einfügen möchten.
5. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   – Falls Sie die Textliste in die gleiche CPU einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   – Falls Sie Textliste in eine andere CPU einfügen und es bereits eine Textliste mit diesem Namen gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Alternativ können Sie die Textliste auch per Drag & Drop mit gedrückter <Strg>-Taste kopieren.

Um Textlisten zu verschieben, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PLC-Meldetextlisten" in der CPU oder Software Unit, für die Sie eine Textliste verschieben möchten.
   Alle Textlisten, die der CPU oder Software Unit zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
2. Ziehen Sie die Textliste, die Sie verschieben möchten, per Drag & Drop an die neue Position.

6.4 Relationen erstellen

6.4.1 Grundlagen zu Relationen

Funktion von Relationen

Mithilfe von Software Units können Sie Ihr Anwenderprogramm in einzelne Programmeinheiten unterteilen, die Sie einzeln bearbeiten, übersetzen und laden können. Über Relationen können Sie Verbindungen schaffen, die Ihnen den Zugriff auf die folgenden Objekte außerhalb der eigenen Software Unit erlauben:

- Funktionsbausteine (FB), Funktionen (FC)
- Globale Datenbausteine (DB)
- PLC-Datentypen
- Technologieobjekte

Beachten Sie bei der Verwendung von Relationen die folgenden Besonderheiten:

- Eine Software Unit kann keine Relation zu sich selbst erstellen.
- Eine Relation kann immer nur in eine Richtung erstellt werden. Das heißt, eine Software Unit kann keine Relation zu einer anderen Software Unit erstellen, wenn von dieser bereits eine Relation zurück besteht.
  Beispiel: Software Unit A hat eine Relation zur Software Unit B. Software Unit B kann keine weitere Relation zur Software Unit A herstellen und auch nicht auf die Programmelemente von Software Unit A zugreifen.
- Relationen, die zu zirkulären Referenzen führen würden, können nicht erstellt werden.

Relationen können Sie in der Relationentabelle anlegen, ändern und löschen.

Zugriff auf Funktionsbausteine (FB) und Funktionen (FC)

Sie können von einer Software Unit aus auf die Funktionsbausteine und Funktionen anderer Software Units zugreifen, wenn diese veröffentlicht sind und eine Relation zu der Software Unit besteht. In der zugreifenden Software Unit können Sie dazu einen neuen Datenbaustein als Instanz des Funktionsbausteins der anderen Software Unit anlegen. Alternativ können Sie auch den zugehörigen Instanz-Datenbaustein eines FB veröffentlichen, um darauf zuzugreifen.

Zugriffe auf einen Funktionsbaustein oder eine Funktion einer Software Unit von einem Baustein aus, der nicht in einer Software Unit erstellt wurde, ist nicht möglich. Ebenso kann ein Funktionsbaustein oder eine Funktion innerhalb einer Software Unit nicht auf einen Baustein außerhalb einer Software Unit zugreifen. Außerhalb einer Software Unit sind die Bausteine, die Sie direkt im Ordner "Programmbausteine" unterhalb der CPU erstellt haben.

Eine Ausnahme bilden Systemfunktionen und Systembausteine, auf sie können Sie wie gewohnt zugreifen.

Zugriff auf globale Datenbausteine (DB)

Sie haben die folgenden Möglichkeiten auf globale Datenbausteine zuzugreifen, die sich nicht in der eigenen Software Unit befinden:

- Sie können aus einer Software Unit heraus auf die globalen Datenbausteine außerhalb von Software Units zugreifen, wenn Sie eine entsprechende Relation erstellt haben. Für den Zugriff auf globale Datenbausteine gelten die folgenden Regeln:
  - Der Zugriff auf einen globalen Datenbaustein, der PLC-Datentypen enthält, ist nicht zulässig.
  - Der Zugriff auf einen nicht optimierten Datenbaustein ist nicht zulässig.
  - Der Zugriff auf globale ARRAY-Datenbausteine ist nicht zulässig.
  - Der Zugriff auf einen globalen fehlersicheren Datenbaustein (F-DB) ist nicht zulässig.
- Auf die globalen Datenbausteine einer Software Unit kann von einer anderen Software Unit aus zugegriffen werden, wenn die Datenbausteine veröffentlicht sind und eine Relation zu der Software Unit besteht. Ein Zugriff auf die globalen Datenbausteine einer Software Unit von Bausteinen aus, die nicht in Software Units abgelegt sind, ist jedoch nicht möglich.

Zugriff auf PLC-Datentypen

Sie können von einer Software Unit aus auf die PLC-Datentypen anderer Software Units zugreifen, wenn diese veröffentlicht sind und eine Relation zu der Software Unit besteht. Ein Zugriff auf die PLC-Datentypen einer Software Unit von Bausteinen aus, die nicht in Software Units abgelegt sind, ist jedoch nicht möglich.

Bausteine und die von ihnen in der Bausteinschnitte in den Abschnitten "Input", "Output", "InOut" und "Static" verwendeten PLC-Datentypen müssen den gleichen Veröffentlichungsstatus haben. Wenn also ein Baustein „Motor“ veröffentlicht ist und den PLC-Datentyp „UDT1“ der gleichen Software Unit verwendet, muss auch der PLC-Datentyp „UDT1“ veröffentlicht sein.

Zugriff auf Technologieobjekte

Sie können aus einer Software Unit heraus auf Technologieobjekte zugreifen, indem Sie in der Software Unit eine Relation zu dem entsprechenden Objekt erstellen.

Siehe auch

Programmelemente veröffentlichen (Seite 7187)

Relationen einrichten (Seite 7188)

6.4.2 Programmelemente veröffentlichen

Damit andere Software Units auf die Funktionsbausteine (FBs), Funktionen (FCs), globalen Datenbausteine (DBs) und PLC-Datentypen einer Software Unit zugreifen können, müssen diese Programmelemente veröffentlicht sein. Falls dies beim Anlegen dieser Elemente nicht erfolgt ist, können Sie das jederzeit ändern. Sie können aber auch für ein Element die Veröffentlichung wieder zurücknehmen. Dann ist ein Zugriff von außen nicht mehr möglich.

Für die folgenden Programmelemente können Sie den Veröffentlichungsstatus nicht ändern:

- Organisationsbausteine (OBs)
- PLC-Variablentabellen
- ProDiag-Funktionsbausteine

Diese Programmelemente sind immer unveröffentlicht. Ein Zugriff von außerhalb der eigenen Software Unit ist nicht möglich.

Sie können den Veröffentlichungsstatus für ein Programmelement an den folgenden Stellen ändern:

- In den Eigenschaften des Programmelements
- Im Übersichtsfenster
- In der Portalansicht

In der Detailansicht erhalten Sie für jedes Programmelement den Veröffentlichungsstatus angezeigt.

Veröffentlichungsstatus in den Eigenschaften des Programmelements ändern

Um den Veröffentlichungsstatus für ein Programmelement über seine Eigenschaften zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Software Units" und navigieren Sie zu dem Programmelement, für das Sie den Veröffentlichungsstatus ändern möchten.
2. Klicken Sie mit der rechten Maustaste auf das Programmelement.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Programmelements wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf die Gruppe "Allgemein", falls diese nicht angezeigt wird.
5. Aktivieren oder deaktivieren Sie unter "Zugriff" das Optionskästchen "Veröffentlicht".
6. Bestätigen Sie Ihre Eingaben mit "OK".
   Der Veröffentlichungsstatus für das selektierte Programmelement wird geändert.

Veröffentlichungsstatus im Übersichtsfenster ändern

Um den Veröffentlichungsstatus für ein Programmelement im Übersichtsfenster zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Software Units" und navigieren Sie zu der Software Unit des Programmelements, dessen Veröffentlichungsstatus Sie ändern möchten.
2. Selektieren Sie entweder den Ordner "Programmbausteine" oder "PLC-Datentypen", abhängig vom Typ des Programmelements.
3. Öffnen Sie das Übersichtsfenster entweder über die Schaltfläche "Maximiert/minimiert die Übersicht" oder über das Menü "Ansicht > Übersicht".
4. Öffnen Sie das Register "Details".
   In der Spalte "Veröffentlicht" wird für jedes Element der Veröffentlichungsstatus angezeigt.
5. Aktivieren oder deaktivieren Sie in der Spalte "Veröffentlicht" das Optionskästchen des Programmelements, das Sie ändern möchten.
6. Alternativ können Sie mehrere Programmelemente selektieren und den Veröffentlichungsstatus gleichzeitig für alle selektierten Elemente über den Kontextmenübefehl "Veröffentlichung" ändern.

Veröffentlichungsstatus in der Portalansicht ändern

Um den Veröffentlichungsstatus für ein Programmelement in der Portalansicht zu ändern, gehen Sie folgendermaßen vor:

1. Wechseln Sie in die Portalansicht.
2. Klicken Sie auf "PLC-Programmierung".
3. Klicken Sie auf "Alle Objekte anzeigen".
4. Öffnen Sie das Register "Details".
   In der Spalte "Veröffentlicht" wird für jedes Element der Veröffentlichungsstatus angezeigt.
5. Aktivieren oder deaktivieren Sie in der Spalte "Veröffentlicht" das Optionskästchen des Programmelements, das Sie ändern möchten.

Siehe auch

Grundlagen zu Relationen (Seite 7185)

Relationen einrichten (Seite 7188)

6.4.3 Relationen einrichten

Sie haben die Möglichkeit, Relationen zu anderen Software Units, globalen Datenbausteinen und Technologieobjekte einzurichten. Beachten Sie, dass Sie nur auf die Objekte einer anderen Software Unit zugreifen können, für die die Eigenschaft "Veröffentlicht" gesetzt ist. Sie können Relationen jederzeit ändern oder löschen.

Voraussetzung

Der Ordner "Software Units" in der Projektnavigation ist geöffnet.

Relationen erzeugen

Um eine neue Relation zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Software Unit, für die Sie eine Relation erzeugen möchten.
2. Doppelklicken Sie auf "Relationen".
   Die Relationentabelle wird geöffnet.
3. Klicken Sie auf "<Neue Relation hinzufügen>".
4. Wählen Sie in der Spalte "Relationstyp" den Typ der Relation aus.
5. Wählen Sie in der Spalte "Verwendbares Element" das Element aus, auf das Sie zugreifen möchten, oder geben Sie es ein.
   Die Relation wurde erzeugt und Sie können auf das Element zugreifen.

Relationen ändern

Um eine Relation zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Software Unit, für die Sie eine Relation ändern möchten.
2. Doppelklicken Sie auf "Relationen".
   Die Relationentabelle wird geöffnet.
3. Ändern Sie in der Spalte "Verwendbares Element" bei Bedarf das Element, auf das Sie zugreifen möchten.

Relationen löschen

Um eine Relation zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Software Unit, für die Sie eine Relation löschen möchten.
2. Doppelklicken Sie auf "Relationen".
   Die Relationentabelle wird geöffnet.
3. Selektieren Sie die Relationen, die Sie löschen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Löschen" oder drücken Sie die Taste <Entf>.
5. Bestätigen Sie die Sicherheitsabfrage mit "Ja".
   Die selektierten Relationen werden gelöscht.

Siehe auch

Grundlagen zu Relationen (Seite 7185)

Programmelemente veröffentlichen (Seite 7187)

6.5 Software Units übersetzen und laden

6.5.1 Grundlagen zum Übersetzen und Laden von Software Units

Einführung

Software Units sind Programmeinheiten, die Sie unabhängig vom übrigen Programmcode übersetzen und laden können. Gibt es Abhängigkeiten zwischen Software Units, erhalten Sie beim Übersetzen eine entsprechende Meldung. Nach dem erfolgreichen Übersetzen können Sie Software Units in ein Gerät oder auf eine Memory Card laden. Dabei kann der Ladevorgang nur durchgeführt werden, wenn die folgenden Voraussetzungen gegeben sind:

- Im Ladeumfang ist kein Programmelement enthalten, das es mit gleichem Namen oder gleicher Nummer an einer anderen Position auf dem Gerät oder der Memory Card gibt.
- Wenn eine Software Unit Relationen zu Objekten außerhalb der Software Unit besitzt und diese Objekte seit ihrem letzten Laden geändert wurden, müssen diese Objekte im Ladeumfang enthalten sein. Dies betrifft veröffentlichte Bausteine anderer Software Units, aber auch globale Datenbausteine und Technologieobjekte außerhalb von Software Units.
- Im Ladeumfang ist kein Systembaustein vorhanden, den es mit anderer Nummer oder Version bereits auf dem Gerät oder der Memory Card gibt.
- Im Ladeumfang befinden sich keine Organisationsbausteine mit Nummern, die im Gerät oder auf der Memory Card für Organisationsbausteine anderen Typs verwendet werden.
- Wenn sich im Ladeumfang ein Organisationsbaustein mit einem bestimmten Startereignis befindet und es weitere Software Units mit Organisationsbausteinen mit dem gleichen Startereignis gibt, müssen diese weiteren Software Units ebenfalls im Ladeumfang enthalten sein. Dies betrifft z. B. den Zyklus-OB und den Anlauf-OB.
- Die maximale Anzahl von Organisationsbausteinen eines bestimmten Typs wird durch den Ladevorgang im Gerät oder auf der Memory Card nicht überschritten. Wenn Sie den Ladevorgang dennoch durchführen möchten, laden Sie die gesamte CPU in das Gerät oder auf die Memory Card. Dadurch stellen Sie die Konsistenz zwischen Projekt und Gerät oder Memory Card wieder her.
- Das Laden aller Software Units über den Ordner "Software Units" ist nur möglich, wenn sich im Gerät oder auf der Memory Card keine Software Units befinden, die im Projekt nicht vorhanden sind. Ansonsten würde diese Software Unit gelöscht werden. Wenn Sie den Ladevorgang dennoch durchführen möchten, laden Sie die gesamte CPU in das Gerät oder auf die Memory Card. Dadurch stellen Sie die Konsistenz zwischen Projekt und Gerät oder Memory Card wieder her.
- Alle auf dem Gerät oder der Memory Card vorhandenen PLC-Datentypen sind mit dem gleichen Stand auch im Offline-Projekt vorhanden.
- Im Gerät oder auf der Memory Card befindet sich bereits die Hardwarekonfiguration für das Gerät. Wenn das noch nicht der Fall ist, müssen Sie die gesamte CPU laden.

Wenn keine Abhängigkeiten zu anderen Programmteilen bestehen, befinden sich nur die Software Units im Ladeumfang, für die Sie den Ladevorgang angestoßen haben. Das

bedeutet, dass es beim restlichen Programm Unterschiede zwischen den Programmelementen im Projekt und auf dem Gerät oder der Memory Card geben kann.

Hinweis

Prüfsumme bei einzeln geladenen Software Units

Zur Prüfung der Identität bzw. Integrität von PLC-Programmen wird beim Übersetzen einer CPU im Projekt eine Prüfsumme erzeugt, die mit ins Gerät oder auf die Memory Card geladen wird. Mit der Anweisung "GetChecksum" können Sie diese Prüfsumme zur Laufzeit im Programm auslesen. Wenn Sie dagegen einzelne Software Units in ein Gerät oder auf eine Memory Card laden, dann wird die Prüfsumme auf "0x0000000000000000" zurückgesetzt. Der Grund dafür ist, dass die Prüfsumme im Projekt nicht mehr zur Prüfsumme des Geräts passt. Beim nächsten Laden des gesamten PLC-Programms erhalten Sie wieder eine gültige Prüfsumme. Beachten Sie dies bei der Verwendung von „GetChecksum".

Software Units im Betriebszustand "RUN" laden

Für das Laden von Software Units im Betriebszustand "RUN" gelten die gleichen Regeln wie beim Laden der verschiedenen Programmelemente außerhalb von Software Units. Beachten Sie hierbei die Beschreibungen auf den Seiten "Bausteine im Betriebszustand "RUN" in Gerät laden" der verschiedenen CPU-Familien im Informationssystem.

Software Units in Verbindung mit OPC UA

Wenn Sie die OPC UA-Kommunikation in Verbindung mit den Software Units verwenden möchten, dann beachten Sie beim Übersetzen Ihres Anwenderprogramms die folgenden Besonderheiten:

- Beim Kompilieren der Client-Schnittstellen entstehen die relevanten Datenbausteine für OPC UA. Diese werden allerdings außerhalb der Software Units abgelegt. Verschieben Sie sie daher per Drag & Drop in die gewünschten Software Units.
- Beim Kompilieren der Client-Schnittstellen entstehen zudem neue PLC-Datentypen im Ordner "Systemdatentypen" außerhalb der Software Units. Verschieben Sie auch diese in den Ordner "PLC-Datentypen" der entsprechenden Software Units.
- Wenn Sie den Namen von Lese- Schreib- oder Methodenlisten ändern, entstehen neue PLC-Datentypen. Diese müssen Sie ebenfalls wieder in die entsprechende Software Units verschieben.
- Wenn Sie neue Knoten anlegen oder den Namen von bestehenden Knoten in den Listen ändern, muss die entsprechende Client-Schnittstelle neu übersetzt werden, damit die Namen in den Datenbausteinen und PLC-Datentypen in den Software Units aktualisiert werden.

Siehe auch

Software Units übersetzen (Seite 7192)

Grundlagen zum Übersetzen und Laden von PLC-Programmen (Seite 7863)

6.5.2 Software Units übersetzen

Um Software Units zu übersetzen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Übersetzen der Programmelemente eines Geräts. Dies schließt die Programmelemente der Software Units mit ein.
- Übersetzen der Programmelemente aller Software Units eines Geräts.
- Übersetzen der Programmelemente einzelner Software Units.
- Übersetzen der Bausteine im Ordner "Programmbausteine" einer Software Unit

Abhängig davon, welche Möglichkeit Sie verwenden, erhalten Sie unterschiedliche Optionen für den Übersetzungsvorgang:

- Hardware und Software (nur Änderungen)
- Hardware (nur Änderungen)
- Hardware (komplett übersetzen)
- Software (nur Änderungen)
- Software (komplett übersetzen)
- Software (Speicherreserve zurücksetzen)

Wenn eine Software Unit auf Programmelemente einer anderen Software Unit zugreift, kann es beim Übersetzung zu Inkonsistenzen kommen. Übersetzen Sie in einem solchen Fall auch die abhängige Software Unit.

Übersetzen der Programmelemente eines Geräts

Um die Programmelemente eines Geräts einschließlich der Software Units zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, dessen Programmelemente Sie übersetzen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

Übersetzen der Programmelemente aller Software Units eines Geräts

Um die Programmelemente aller Software Units eines Geräts zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation den Ordner "Software Units".
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

Übersetzen der Programmelemente einzelner Software Units

Um die Programmelemente einzelner Software Units zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation im Ordner "Software Units" die Software Units, deren Programmelemente Sie übersetzen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

Übersetzen der Bausteine im Ordner "Programmbausteine" einer Software Unit

Um die Bausteine einer Software Unit zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation im Ordner "Programmbausteine" innerhalb einer Software Unit die Bausteine, die Sie übersetzen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

Ergebnis

Die Programmelemente werden übersetzt. Ob das Übersetzen erfolgreich durchgeführt wurde, können Sie im Inspektorfenster unter "Info > Übersetzen" kontrollieren.

Siehe auch

Grundlagen zum Übersetzen und Laden von Software Units (Seite 7190)

6.5.3 Software Units laden

6.5.3.1 Software Units in ein Gerät laden

Sie können eine, mehrere oder alle Software Units in einem Ladevorgang in ein Gerät laden. Um alle Software Units auf einmal zu laden, darf es auf dem Gerät keine Software Unit geben, die nicht im Offline-Projekt vorhanden ist. In diesem Fall können Sie entweder die gesamte CPU oder einzelne Software Units laden.

Voraussetzung

Auf dem Gerät ist die Hardwarekonfiguration vorhanden.

Eine oder mehrere Software Units in ein Gerät laden

Um eine oder mehrere Software Units in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Software Units" in der Projektnavigation.
2. Selektieren Sie die Software Units, die Sie laden möchten.

3. Wählen Sie im Kontextmenü im Untermenü "Laden in Gerät" einen der beiden folgenden Befehle:
   – "Software (nur Änderungen)": Diese Option lädt nur die selektierten Software Units. Wenn jedoch an anderen Software Units Änderungen vorgenommen wurden, die zu Inkonsistenzen mit den selektierten Units führen, müssen diese Software Units ebenfalls geladen werden. In diesem Fall kann der Ladevorgang nicht durchgeführt werden und Sie müssen das Laden mit der Option „Software (mit Software Unit-Relationen)“ neu starten.
   – "Software (mit Software Unit-Relationen)": Diese Option lädt die selektierten Software Units zusammen mit den Software Units, zu denen Relationen vorhanden sind. Dabei werden auch die Änderungen in den über Relationen verbundenen Software Units geladen, die nicht zu Inkonsistenzen mit den selektierten Units führen.

   Wenn bereits eine Online-Verbindung definiert wurde, werden die Projektdaten bei Bedarf übersetzt und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Führen Sie dann Schritt 4 aus.
4. Stellen Sie alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzen", damit die Software Units geladen werden. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
7. Wenn der Ladevorgang nicht im Betriebszustand "RUN" ausgeführt wurde und die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, wählen Sie in der Klappliste "Baugruppen nach dem Ladevorgang starten" den Eintrag "Baugruppe starten".
8. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Alle Software Units in ein Gerät laden

Um alle Software Units in einem Ladevorgang in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Software Units" in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen)". Wenn bereits eine Online-Verbindung definiert wurde, werden die Projektdaten bei Bedarf übersetzt und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen. Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Führen Sie dann Schritt 3 aus.
3. Stellen Sie alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzen", damit die Software Units geladen werden. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Wenn der Ladevorgang nicht im Betriebszustand "RUN" ausgeführt wurde und die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, wählen Sie in der Klappliste "Baugruppen nach dem Ladevorgang starten" den Eintrag "Baugruppe starten".
7. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Die Software Units werden in das Gerät geladen. Wenn die Änderungen weitere Objekte betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Objekte, die nur online im Gerät existieren, werden gelöscht. Beachten Sie, dass es durch das Laden einzelner Software Units zu Inkonsistenzen zwischen dem Anwenderprogramm im Gerät und dem Anwenderprogramm im Offline-Projekt kommen kann.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

6.5.3.2 Software Units von einem Gerät laden

Sie können entweder alle oder einzelne Software Units von einem Gerät laden oder die Software Units zusammen mit allen Projektdaten. Um die Software Units gemeinsam mit den Projektdaten von einem Gerät zu laden, beachten Sie die Beschreibung auf der Seite "Projektdaten von einem Gerät laden".

Alle Software Units von einem Gerät laden

Um alle Software Units von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Software Units laden möchten.
   Siehe auch: "Online verbinden und Online-Verbindung trennen" im Informationssystem
2. Selektieren Sie in der Projektnavigation den Ordner "Software Units" des Geräts, von dem Sie Software Units laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Einzelne Software Units von einem Gerät laden

Um einzelne Software Units von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Software Units laden möchten.
   Siehe auch: "Online verbinden und Online-Verbindung trennen" im Informationssystem
2. Selektieren Sie in der Projektnavigation die Software Units, die Sie vom Gerät laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Ergebnis

Die Software Units werden von dem Gerät in das Projekt geladen. Sie können sie bearbeiten und erneut übersetzen und zurück in das Gerät laden.

6.5.3.3 Software Units auf eine Memory Card laden

Sie haben die folgenden Möglichkeiten, Software Units zusammen mit den übrigen Projektdaten auf eine Memory Card laden:

- Projektdaten per Drag & Drop auf eine Memory Card laden
- Projektdaten auf eine Memory Card schreiben

Voraussetzung

Eine Memory Card wird angezeigt.

Vorgehen

Um Software Units zusammen mit den übrigen Projektdaten auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation die Projektdaten, die Sie laden möchten, auf die Memory Card. Beachten Sie, dass Sie den Ordner "Programmbausteine" nicht als Ziel verwenden können.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden".
4. Der Ladevorgang wird durchgeführt.

Oder:

1. Selektieren Sie in der Projektnavigation die Projektdaten, die Sie laden möchten.
2. Klicken Sie mit der rechten Maustaste auf die Selektion und wählen Sie im Kontextmenü den Befehl "Kopieren". Alternativ können Sie auch die Tastenkombination <Strg+C> verwenden.
3. Klicken Sie mit der rechten Maustaste auf die Memory Card und wählen Sie im Kontextmenü den Befehl "Einfügen". Alternativ können Sie auch die Tastenkombination <Strg+V> verwenden.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Oder:

1. Selektieren Sie in der Projektnavigation die Projektdaten, die Sie laden möchten.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben".
   Der Dialog "Memory Card auswählen" wird geöffnet.
3. Wählen Sie eine Memory Card aus, die mit der CPU kompatibel ist.
   Im unteren Bereich des Dialogs wird eine Schaltfläche mit einem grünen Häkchen aktiv.

4. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

6.5.3.4 Software Units von einer Memory Card laden

Sie können Software Units, die sich auf einer Memory Card befinden, zusammen mit den übrigen Projektdaten wieder ins Projekt laden. Dabei haben Sie die folgenden Möglichkeiten:

- Projektdaten der Memory Card als neue Station laden
  Mit dieser Option laden Sie die Projektdaten einer Memory Card als neue Station in Ihr Projekt.
- Projektdaten der Memory Card in ein vorhandenes Gerät laden
  Mit dieser Option laden Sie die Projektdaten einer Memory Card in ein vorhandenes Gerät im Projekt.

Voraussetzung

- Ein Projekt ist geöffnet.
- Die Memory Card wird angezeigt.

Als neue Station laden

Um die Projektdaten einer Memory Card als neue Station in das Projekt zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Memory Card, von der Sie die Projektdaten laden möchten.
2. Wählen Sie im Menü "Online" den Befehl "Laden des Geräts als neue Station (Hardware und Software)".

Oder:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf das Projekt.

Oder:

1. Klicken Sie mit der rechten Maustaste auf die Memory Card.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Klicken Sie mit der rechten Maustaste auf das Projekt.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

In ein vorhandenes Gerät laden

Um die Projektdaten einer Memory Card in ein vorhandenes Gerät zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf ein Gerät im Projekt oder kopieren Sie die Memory Card und fügen die Daten in ein Gerät ein.
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet.
2. Kontrollieren Sie die Meldungen im Dialog "Vorschau für das Laden von Gerät" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

6.6 Software Units vergleichen

6.6.1 Grundlagen zum Software Unit-Vergleich

Offline/Online- und Offline/Offline-Vergleich von Software Units

Der Vergleich von Software Units funktioniert im Prinzip genauso wie der Vergleich anderer Projektdaten auch. So erhalten Sie z. B. ebenfalls einen einfachen Offline/Online-Vergleich, wenn Sie eine Online-Verbindung herstellen. Dabei werden vergleichsfähige Objekte in der Projektnavigation mit Symbolen gekennzeichnet, die das Vergleichsergebnis repräsentieren. Dabei gelten die folgenden Regeln:

- Die Software Units werden als identisch angezeigt, wenn ihre Eigenschaften und alle unterlagerten Objekte identisch sind.
- Die Software Units werden als unterschiedlich angezeigt, wenn ihre Eigenschaften unterschiedlich sind.
- Für eine Software Unit wird das Ausrufezeichen angezeigt, wenn die unterlagerten Objekte unterschiedlich, ihre Eigenschaften aber identisch sind.

Darüber hinaus können Sie einen weitergehenden Offline/Online- und Offline/Offline-Vergleich im Vergleichseditor durchführen.

Wenn Sie an einem Gerät einen Offline/Online- oder einen Offline/Offline-Vergleich starten, werden auch die Software Units in den Vergleich mit einbezogen. Im Vergleichseditor werden die Software Units sowohl in der linken, als auch in der rechten Vergleichstabelle angezeigt. Im Bereich "Eigenschaftsvergleich" des Vergleichseditors sehen Sie unter "Quelldaten > Software Unit" die Prüfsummen der verglichenen Objekte.

Beim Vergleich werden die Software Units anhand ihrer Namen einander zugeordnet und ihre Objekte miteinander verglichen. Objekte mit gleichem Namen in verschiedenen Software Units werden dabei nicht miteinander verglichen. Das bedeutet, dass wenn z. B. im Offline-Projekt der Baustein "Block_1" in der Software Unit "Unit1" enthalten ist und online im Gerät in der Software Unit "Unit2", dann werden sie als unterschiedliche Objekte betrachtet.

Für nicht identische Objekte können Sie Aktionen ausführen, um die Objekte zu synchronisieren. Wenn Sie die Aktion für eine Software Unit festlegen, kann es vorkommen, dass durch die Synchronisation mehrere Objekte mit dem gleichen Namen entstehen würden. Bei einem Offline/Online-Vergleich erhalten Sie einen Auswahldialog, in dem Sie entscheiden können, ob die bereits vorhandenen Objekte gelöscht oder umbenannt werden sollen. Bei einem Offline/Offline-Vergleich hingegen wird die Aktion abgebrochen und Sie erhalten eine entsprechende Meldung angezeigt.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Bausteine anlegen und verwalten 7

7.1 Bausteine anlegen

7.1.1 Bausteinordner

Funktion

In der Projektnavigation finden Sie einen Ordner "Programmbausteine", in dem Sie folgende Bausteine anlegen und verwalten können:

- Organisationsbausteine (OB)
- Funktionsbausteine (FB)
- Funktionen (FC)
- Datenbausteine (DB)

Zusätzlich wird innerhalb des Ordners "Programmbausteine" ein Unterordner "Systembausteine" mit einem weiteren Unterordner "Programmressourcen" erstellt, wenn Sie zum ersten Mal eine Anweisung in Ihr Programm ziehen, die systemintern ein Funktionsbaustein ist. Dabei wird der Instanz-Datenbaustein des systeminternen Funktionsbausteins in den Ordner "Programmressourcen" eingefügt. Sie können solche Instanz-Datenbausteine aus dem Ordner "Programmressourcen" kopieren, umbenennen und löschen. Der Ordner "Programmressourcen" wird zusammen mit dem Ordner "Systembausteine" gelöscht, wenn er keine Bausteine mehr enthält.

Für jedes Gerät wird automatisch ein Zyklus-OB erzeugt und in den Ordner "Programmbausteine" eingefügt.

Hinweis

Verschieben von Bausteinen aus dem Ordner "Programmressourcen" oder in den Ordner "Programmressourcen"

Sie können Bausteine zwischen dem Ordner "Programmressourcen" und anderen Ordnern verschieben. Beachten Sie beim Verschieben von Bausteinen aus dem Ordner "Programmressourcen" oder in den Ordner "Programmressourcen" jedoch die folgenden Hinweise:

- Verschieben von Bausteinen aus dem Ordner "Programmressourcen" in einen beliebigen anderen Ordner: Beim nächsten Übersetzen des Anwenderprogramms werden die verschobenen Instanz-Datenbausteine wieder in den Ordner "Programmressourcen" verschoben.
- Verschieben von beliebigen Bausteinen in den Ordner "Programmbausteine": Bausteine im Ordner "Programmressourcen", die für den Ablauf des Anwenderprogramms nicht erforderlich sind, werden beim nächsten Übersetzungsvorgang entfernt. Dadurch kann es sein, dass Ihr verschobener Baustein gelöscht wird.

Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 7203)

Datenbausteine anlegen (Seite 7204)

Organisationsbausteine anlegen (Seite 7202)

Bausteine aus Bibliotheken verwenden (Seite 7205)

7.1.2 Organisationsbausteine anlegen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Organisationsbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Organisationsbaustein (OB)".
3. Wählen Sie den Typ des neuen Organisationsbausteins aus.
4. Geben Sie einen Namen für den neuen Organisationsbaustein ein.
5. Geben Sie die Eigenschaften des neuen Organisationsbausteins ein.
6. Um weitere Eigenschaften des neuen Organisationsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
7. Geben Sie alle gewünschten Eigenschaften ein.
8. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Organisationsbaustein nach dem Erstellen gleich geöffnet werden soll.
9. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Der neue Organisationsbaustein wird erzeugt. Sie finden den Organisationsbaustein in der Projektnavigation im Ordner "Programmbausteine". Einige Organisationsbausteine können Sie nach dem Anlegen im Inspektorfenster oder in der Gerätesicht weiter parametrieren. Ob der neu angelegte Organisationsbaustein zusätzliche Parameter besitzt, können Sie in der Beschreibung des Organisationsbausteins nachlesen.

Siehe auch

Organisationsbausteine (OB) (Seite 39)

Bausteinordner (Seite 7201)

Funktionen und Funktionsbausteine anlegen (Seite 7203)

Datenbausteine anlegen (Seite 7204)

Bausteine aus Bibliotheken verwenden (Seite 7205)

Bausteintitel eingeben (Seite 7211)

Bausteinkommentar eingeben (Seite 7212)

7.1.3 Funktionen und Funktionsbausteine anlegen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um eine Funktion (FC) oder einen Funktionsbaustein (FB) anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktionsbaustein (FB)" oder "Funktion (FC)".
3. Geben Sie einen Namen für den neuen Baustein ein.
4. Geben Sie die Eigenschaften des neuen Bausteins ein.
5. Um weitere Eigenschaften des neuen Bausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
6. Geben Sie alle gewünschten Eigenschaften ein.
7. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
8. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Der neue Baustein wird erzeugt. Sie finden den Baustein in der Projektnavigation im Ordner "Programmbausteine".

Siehe auch

Funktionsbausteine (FB) (Seite 41)

Funktionen (FC) (Seite 40)

Grundlagen zum Bausteinzugriff (Seite 55)

Bausteinordner (Seite 7201)

Organisationsbausteine anlegen (Seite 7202)
Datenbausteine anlegen (Seite 7204)
Bausteine aus Bibliotheken verwenden (Seite 7205)
Bausteintitel eingeben (Seite 7211)
Bausteinkommentar eingeben (Seite 7212)

7.1.4 Datenbausteine anlegen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Datenbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Wählen Sie den Typ des Datenbausteins aus. Folgende Möglichkeiten stehen zur Verfügung:
   - Um einen globalen Datenbaustein anzulegen, wählen Sie den Listeneintrag "Global-DB".
   - Um einen ARRAY-Datenbaustein anzulegen, wählen Sie den Listeneintrag "ARRAY-DB".
   - Um einen Instanz-Datenbaustein anzulegen, wählen Sie aus der Liste den Funktionsbaustein aus, dem Sie den Instanz-Datenbaustein zuordnen möchten. In der Liste werden nur Funktionsbausteine angeboten, die zuvor für die CPU erstellt wurden.
   - Um einen Datenbaustein basierend auf einem PLC-Datentyp anzulegen, wählen Sie aus der Liste den PLC-Datentyp aus. In der Liste werden nur PLC-Datentypen angeboten, die zuvor für die CPU erstellt wurden.
   - Um einen Datenbaustein basierend auf einem Systemdatentyp anzulegen, wählen Sie aus der Liste den Systemdatentyp aus. In der Liste werden nur Systemdatentypen angeboten, die zuvor in Programmbausteinen der CPU eingefügt wurden.
4. Geben Sie einen Namen für den Datenbaustein ein.
5. Geben Sie die Eigenschaften des neuen Datenbausteins ein.
6. Wenn Sie als "Typ" einen ARRAY-DB ausgewählt haben, geben Sie den ARRAY-Datentyp und die obere ARRAY-Grenze ein.
   Die obere ARRAY-Grenze können Sie jederzeit im Eigenschaftsfenster des erzeugten Bausteins ändern. Der ARRAY-Datentyp ist nachträglich nicht mehr änderbar.
7. Wenn Sie als "Typ" einen Baustein angewählt haben, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.

8. Um weitere Eigenschaften des neuen Datenbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
9. Geben Sie alle gewünschten Eigenschaften ein.
10. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
11. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Der neue Datenbaustein wird erzeugt. Sie finden den Datenbaustein in der Projektnavigation im Ordner "Programmbausteine".

Hinweis

Remanenz von ARRAY-Datenbausteinen

ARRAY-Datenbausteine oder deren Komponenten können nicht remanent gesetzt werden.

Siehe auch

Grundlagen zur Programmierung von Datenbausteinen (Seite 7319)

Globale Datenbausteine (DB) (Seite 42)

Instanz-Datenbausteine (Seite 53)

Systemdatentypen (Seite 364)

Bausteine aus Bibliotheken verwenden (Seite 7205)

7.1.5 Bausteine aus Bibliotheken verwenden

Sie haben die Möglichkeit, Bausteine in der Projektbibliothek oder in einer globalen Bibliothek abzulegen, um sie in Ihrem Anwenderprogramm mehrfach zu verwenden. Dabei können Sie die Bausteine als Kopiervorlagen oder als Typen einfügen.

Siehe auch: Grundlagen zu Bibliotheken

Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Für globale Bibliotheken ist kein Schreibschutz festgelegt.

Bausteine als Kopiervorlagen zur Projektbibliothek oder einer globalen Bibliothek hinzufügen

Um der Projektbibliothek oder einer globalen Bibliothek neue Bausteine als Kopiervorlagen hinzuzufügen, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
2. Ziehen Sie den Baustein, den Sie der Bibliothek hinzufügen möchten, per Drag & Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen" der Projektbibliothek oder einer globalen Bibliothek. Lassen Sie die linke Maustaste erst los, wenn unter dem Mauszeiger ein kleines Plussymbol erscheint.

Oder:

1. Kopieren Sie den Baustein, den Sie als Kopiervorlage hinzufügen möchten.
2. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Bausteine als Typen zur Projektbibliothek oder einer globalen Bibliothek hinzufügen

Um der Projektbibliothek oder einer globalen Bibliothek neue Bausteine als Typen hinzuzufügen, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek.
2. Ziehen Sie den Baustein, den Sie als Typ hinzufügen möchten, mit Drag & Drop in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" der Projektbibliothek. Lassen Sie die linke Maustaste erst los, wenn unter dem Mauszeiger ein kleines Plussymbol erscheint.
3. Ziehen Sie bei Bedarf den Baustein aus dem Ordner "Typen" der Projektbibliothek in den Ordner "Typen" einer globalen Bibliothek. Oder kopieren Sie den Baustein in der Projektbibliothek und fügen ihn in eine globale Bibliothek ein.

Oder:

1. Kopieren Sie den Baustein, den Sie als Typ hinzufügen möchten.
2. Maximieren Sie die Projektbibliothek.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".
5. Ziehen Sie bei Bedarf den Baustein aus dem Ordner "Typen" der Projektbibliothek in den Ordner "Typen" einer globalen Bibliothek. Oder kopieren Sie den Baustein in der Projektbibliothek und fügen ihn in eine globale Bibliothek ein.

Bausteine der Projektbibliothek oder einer globalen Bibliothek verwenden

Um einen Baustein aus der Projektbibliothek oder einer globalen Bibliothek in Ihrem Projekt zu verwenden, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek oder die globale Bibliothek, sodass Sie den Baustein sehen können, den Sie verwenden möchten.
2. Ziehen Sie den Baustein per Drag & Drop in den Bausteinordner der CPU. Nicht erlaubte Einfügestellen werden dadurch gekennzeichnet, dass sich der Mauszeiger in einen durchgestrichenen Kreis verwandelt.

Hinweis

Wenn Sie von einem Typ aus einer globalen Bibliothek eine Instanz ableiten, wird der Typ zusätzlich in die Projektbibliothek eingefügt. Die Instanz ist anschließend nur mit dem Typ in der Projektbibliothek verbunden.

7.1.6 Bausteine kopieren und einfügen

7.1.6.1 Grundlagen zum Kopieren und Einfügen von Bausteinen

Funktion

Sie können neue Bausteine auch dadurch anlegen, dass Sie bestehende Bausteine kopieren und die Kopie einfügen. Beachten Sie dabei folgende Prinzipien beim Kopieren in CPUs der gleichen Gerätefamilie:

- Organisationsbausteine (OBs), Funktionen (FCs), Funktionsbausteine (FBs) und globale Datenbausteine (DBs) können Sie uneingeschränkt kopieren.
- Instanz-Datenbausteine können Sie nur für den gleichen Funktionsbaustein kopieren, da Sie die Zuordnung zum Funktionsbaustein nicht nachträglich ändern können. Die Zuordnung wird jedoch aufgehoben, wenn Sie den Instanz-Datenbaustein in eine andere CPU kopieren. Falls es dort einen Funktionsbaustein mit dem gleichen Namen gibt, wird der Instanz-Datenbaustein diesem Funktionsbaustein zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein in die andere CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.

Die unterschiedlichen Gerätefamilien unterstützen zum Teil unterschiedliche Bausteine, vor allem bei den Organisationsbausteinen. Aber auch Funktionsbausteine und Funktionen können auf den verschiedenen Geräten mit unterschiedlichen Zugriffsarten programmiert

werden. Daher werden nicht auf allen Geräten alle Bausteine unterstützt. Beachten Sie beim Kopieren in eine andere Gerätefamilie folgende Prinzipien:

- Kopieren in eine S7-1200-CPU:
  - Organisationsbausteine mit der Zugriffsart "optimiert" können in eine S7-1200 kopiert werden. Falls der kopierte OB-Typ von der S7-1200-CPU unterstützt wird, behält der kopierte OB die Eigenschaften seines Ereignisses. Sie müssen ihn jedoch neu übersetzen.
  - Organisationsbausteine mit der Zugriffsart "standard" können Sie zwar auf eine S7-1200 kopieren, sie werden von der CPU jedoch nicht unterstützt.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "optimiert" können in eine S7-1200 kopiert werden. Sie müssen anschließend neu übersetzt werden.
  - Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.
- Kopieren in eine S7-1500-CPU:
  - Organisationsbausteine mit der Zugriffsart "optimiert" können in eine S7-1500 kopiert werden. Falls der kopierte OB-Typ von der S7-1500-CPU unterstützt wird, behält der kopierte OB die Eigenschaften seines Ereignisses. Sie müssen ihn jedoch neu übersetzen. Nicht unterstützte OB-Typen erhalten ein Parkverbotssymbol.
  - Organisationsbausteine mit der Zugriffsart "standard" können in eine S7-1500 kopiert werden. Wenn der OB aus einer S7-300/400-CPU stammt, erhält er das Standardereignis des entsprechenden OB-Typs. Wenn der OB aus einer S7-1200/1500-CPU stammt, behält er die Eigenschaften seines Ereignisses, er muss jedoch neu übersetzt werden.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "optimiert" können in eine S7-1500 kopiert werden. Sie müssen anschließend neu übersetzt werden.
  - Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.
- Kopieren in S7-300/400-CPUs:
  - Organisationsbausteine können zwischen S7-300 und S7-400 beliebig kopiert werden.
  - Organisationsbausteine aus S7-1200/1500-CPUs können Sie zwar in S7-300/400-CPUs kopieren, sie werden von der Ziel-CPU jedoch nicht unterstützt.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) können zwischen S7-300 und S7-400 beliebig kopiert werden.

- Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) aus S7-1200/1500-CPUs können Sie zwar in S7-300/400-CPUs kopieren, sie werden von der Ziel-CPU jedoch nicht unterstützt.
- Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.
- Funktionsbausteine (FBs) und Funktionen (FCs), die mit KOP oder FUP programmiert sind und SCL-Netzwerke enthalten, können Sie zwar in S7-300/400-CPUs kopieren, sie werden von der Ziel-CPU jedoch nicht unterstützt.

Nicht unterstützte Bausteine erkennen Sie in der Projektnavigation am Parkverbotssymbol. Bausteine mit einem Parkverbotssymbol können Sie nicht bearbeiten, sondern nur erneut als Kopierquellen nutzen.

Hinweis

Beim Kopieren von Bausteinen zwischen unterschiedlichen Gerätefamilien ist es möglich, dass der kopierte Baustein neu kompiliert werden muss. Dies betrifft auch das Kopieren von Bausteinen zwischen CPUs und Software-Controller. Falls der Baustein einen Know-how-Schutz besitzt, ist ein erneutes Kompilieren nur mit korrektem Passwort möglich.

Kopierte Daten

Beim Einfügen werden alle Daten des Bausteins kopiert und an die Kopie weiter gegeben. Zu diesen Daten gehören:

- Variablen der Bausteinschnittstelle
- Alle Netzwerke
- Kommentare in allen vorhandenen Übersetzungen
- Im Baustein definierte Meldungen
- Der gesamte Programmcode des kopierten Bausteins mit darin befindlichen Aufrufanweisungen.
  Die folgenden Objekte werden jedoch nicht kopiert:
  - Aufgerufene Bausteine und zugehörige Instanz-Datenbausteine
  - Globale Variablen, die im kopierten Baustein referenziert werden
  - Die im kopierten Baustein verwendeten PLC-Datentypen zur Deklaration der Bausteinparameter und lokalen Variablen
  - Die im kopierten Baustein verwendeten Technologieobjekte

Vermeidung von Namenskonflikten beim Einfügen

Beim Einfügen von kopierten Bausteinen mit identischen Namen wie bereits bestehende Bausteine werden folgende Mechanismen angewendet, um Namenskonflikte zu vermeiden:

- Einfügen des kopierten Bausteins in die gleiche CPU:
  Die Kopie des Bausteins erhält einen Namen, der um eine Nummer erweitert ist. Wenn z. B. Baustein "A" kopiert wird, ist ein möglicher Name der Kopie "A_1". Dabei wird als Nummer kein fortlaufendes Nummernband verwendet, sondern immer die kleinste freie Nummer. Die Kopie von Baustein "A" kann also auch den Namen "A_25" erhalten, falls keine niedrigere Nummer frei ist.
- Einfügen des kopierten Bausteins in eine andere CPU:
  Ein Dialog wird geöffnet, in dem Sie auswählen können, ob der Baustein mit dem gleichen Namen ersetzt oder ob der kopierte Baustein mit einer Duplikatskennung (Name_Nummer) eingefügt werden soll.

Hinweis

Nummernkonflikte

Es können jedoch Nummernkonflikte auftreten, wenn der eingefügte Baustein die gleiche Bausteinnummer wie ein bereits vorhandener Baustein hat. Die Bausteinnummer wird beim Einfügen nicht automatisch verändert. Dies kann z. B. Auswirkungen auf Bausteinaufrufe haben. Prüfen Sie daher beim Kopieren von Bausteinen die Bausteinnummer sorgfältig und korrigieren Sie doppelte Bausteinnummern manuell über die Eigenschaften des Bausteins. Doppelte Bausteinnummern führen zu einem Übersetzungsfehler.

Siehe auch

Bausteine kopieren (Seite 7210)

Bausteine einfügen (Seite 7211)

7.1.6.2 Bausteine kopieren

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Baustein zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Ergebnis

Eine Kopie des Bausteins befindet sich nun in der Zwischenablage und kann entweder in die gleiche oder eine andere CPU eingefügt werden.

Siehe auch

Grundlagen zum Kopieren und Einfügen von Bausteinen (Seite 7207)

Bausteine einfügen (Seite 7211)

7.1.6.3 Bausteine einfügen

Voraussetzung

Sie haben einen Baustein kopiert.

Vorgehen

Um einen kopierten Baustein und seine Daten in eine CPU einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Ordnerstruktur für die CPU, in die Sie den kopierten Baustein einfügen möchten.

   Hinweis

   Beachten Sie, dass Sie den kopierten Baustein nur in eine CPU einfügen können, die die Programmiersprache und den Typ des Bausteins unterstützt.

2. Klicken Sie mit der rechten Maustaste auf den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".
   - Falls Sie den Baustein in die CPU des Originalbausteins einfügen, wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   - Falls Sie den Baustein in eine andere CPU einfügen und es einen Baustein mit diesem Namen schon gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Siehe auch

Grundlagen zum Kopieren und Einfügen von Bausteinen (Seite 7207)

Bausteine kopieren (Seite 7210)

7.1.7 Bausteintitel eingeben

Der Bausteintitel ist die Überschrift des Bausteins. Er ist nicht identisch mit dem Bausteinnamen, der beim Anlegen des Bausteins vergeben wurde. Die Länge des Bausteintitels ist auf eine Zeile beschränkt. Sie können den Bausteintitel für geöffnete und geschlossene Bausteine eingeben.

Voraussetzung

Ein Codebaustein ist vorhanden.

Bausteintitel für geöffnete Bausteine eingeben

Um in einem geöffneten Baustein den Bausteintitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie im Programmiereditor in die Titelzeile des Bausteins.
2. Geben Sie den Bausteintitel ein.

Bausteintitel für geschlossene Bausteine eingeben

Um in einem geschlossenen Baustein den Bausteintitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Information".
4. Geben Sie im Eingabefeld "Titel" den Bausteintitel ein.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Organisationsbausteine anlegen (Seite 7202)

Funktionen und Funktionsbausteine anlegen (Seite 7203)

Bausteinkommentar eingeben (Seite 7212)

7.1.8 Bausteinkommentar eingeben

Mithilfe des Bausteinkommentars können Sie den gesamten Codebaustein dokumentieren. Sie können z. B. den Zweck des Bausteins angeben oder auf Besonderheiten hinweisen. Sie können den Bausteinkommentar für geöffnete und geschlossene Bausteine eingeben.

Hinweis

Der Bausteinkommentar sollte 32767 Unicode-Zeichen nicht überschreiten.

Voraussetzung

Ein Codebaustein ist vorhanden.

Bausteinkommentar für geöffnete Bausteine eingeben

Um in einem geöffneten Baustein einen Bausteinkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den kleinen Pfeil vor dem Bausteintitel.
   Aus dem Pfeil-nach-rechts wird ein Pfeil-nach-unten und der Kommentarbereich wird angezeigt.
2. Klicken Sie im Kommentarbereich auf "Kommentar".
   Die Textstelle "Kommentar" wird markiert.
3. Geben Sie den Bausteinkommentar ein.

Bausteinkommentare für geschlossene Bausteine eingeben

Um für einen geschlossenen Baustein einen Bausteinkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Information".
4. Geben Sie im Eingabefeld "Kommentar" den Bausteinkommentar ein.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Organisationsbausteine anlegen (Seite 7202)

Funktionen und Funktionsbausteine anlegen (Seite 7203)

Bausteintitel eingeben (Seite 7211)

7.2 Bausteineigenschaften festlegen

7.2.1 Grundlagen zu Bausteineigenschaften

Bausteineigenschaften

Jeder Baustein besitzt bestimmte Eigenschaften, die Sie anzeigen und bearbeiten können. Diese Eigenschaften haben z. B. folgende Aufgaben:

- Identifikation des Bausteins
- Anzeigen des Speicherbedarfs und des Übersetzungsstatus des Bausteins
- Anzeigen der Zeitstempel
- Anzeigen der Referenzinformationen
- Einstellen des Zugriffschutzes

Siehe auch

Übersicht über die Bausteineigenschaften (Seite 7215)

Zeitstempel von Bausteinen (Seite 7219)

Bausteineigenschaften anzeigen und bearbeiten (Seite 7222)

Einstellen der Mnemonik (Seite 7130)

7.2.2 Übersicht über die Bausteineigenschaften

Übersicht

Die Eigenschaften der Bausteine sind baustein- und CPU-spezifisch. Daher sind nicht alle Eigenschaften für alle Bausteine und in allen CPU-Familien verfügbar. Die folgende Tabelle zeigt eine Übersicht der Bausteineigenschaften:

Gruppe	Eigenschaft	Beschreibung
Eigenschaften	Name	Innerhalb der Station eindeutiger Bausteinname
	Konstantenname	Name der Konstante, die für den OB in die PLC-Variablentabelle eingefügt wird
	Typ	Bausteintyp (nicht änderbar)
	Nummer	Bausteinnummer
	Ereignisklasse	Ereignisklasse eines OBs (nicht änderbar)
	Sprache	Programmiersprache des Bausteins
	Sprache in Netzwerken	Sprache, die zur Programmierung von Bedingungen in GRAPH-Bausteinen verwendet wird.
	Version	Version des GRAPH-Bausteins
	Teilprozessabbild-Nummer	Anzeige der Teilprozessabbilder, die dem Organisationsbaustein zugeordnet sind (nicht änderbar)
	ARRAY-Datentyp	Datentyp eines ARRAY-Datenbausteins (nicht änderbar)
	ARRAY-Grenze	Obere Grenze eines ARRAY-Datenbausteins Möglichkeiten zur Adressierung von ARRAY-Datenbausteins bietet die Task Card "Anweisungen" im Abschnitt "Verschieben".
Information	Titel	Bausteintitel
	Kommentar	Bausteinkommentar
	Version	Versionsnummer des Bausteins
	Familie	Name der Bausteinfamilie
	Autor	Name des Autors, Firmenname, Abteilungsname oder andere Namen
	Anwenderdefinierte ID	Vom Anwender erzeugte ID
Zeitstempel	Baustein	Erstell- und Änderungszeitpunkt des Bausteins (nicht änderbar)
	Schnittstelle	Änderungszeitpunkt für die Schnittstelle des Bausteins (nicht änderbar)
	Code	Änderungszeitpunkt für Code (nicht änderbar)
	Daten	Änderungszeitpunkt für Daten (nicht änderbar)
	Laderelevant	Zeitpunkt der letzten laderelevanten Änderung (nicht änderbar)

Gruppe	Eigenschaft	Beschreibung
Übersetzung	Status	Details zum letzten Kompiliervorgang (nicht änderbar)
	Längen	Details zu den Bausteinlängen (nicht änderbar)
	Simulation	Zeigt an, ob der Baustein mit SIMATIC S7-PLCSIM Advanced simuliert werden kann. Voraussetzung dazu ist, dass in den Projekteigenschaften die Option "Beim Übersetzen von Bausteinen Simulierbarkeit unterstützen" aktiviert ist. Siehe auch: Eigenschaften des Projekts anzeigen
	Bibliothekskonformität	Zeigt an, ob der Baustein in der Bibliothek mit Know-how-Schutz verwendet werden kann.
Schutz	Schutz	Know-how-Schutz und Kopierschutz für den Baustein einstellen Siehe auch: Schützen von Bausteinen (Seite 7239)

Gruppe	Eigenschaft	Beschreibung
Attribute	IEC-Prüfung	Die Kompatibilität von Operanden in Vergleichsoperationen und arithmetischen Operationen wird gemäß IEC 61131 geprüft. Nicht kompatible Operanden müssen Sie explizit konvertieren. Siehe auch: Übersicht über die Datentypkonvertierung (Seite 507)
	Lokale Fehlerbehandlung im Baustein	Fehlerbehandlung innerhalb des Bausteins mit der Anweisung GET_ERROR bzw. GET_ERR_ID (nicht änderbar). Siehe auch: Programmablauffehler behandeln (Seite 143)
	Optimierter Bausteinzugriff	Bei Bausteinen mit optimiertem Zugriff enthält die Variablendeklaration nur die symbolischen Namen der Datenelemente. Die Adressen werden vom System automatisch optimiert und verwaltet. Die Performance der CPU wird gesteigert, Zugriffsfehler, z. B. aus SIMATIC-HMI, können nicht auftreten. Siehe auch: Bausteine mit optimiertem Zugriff (Seite 55)
	Multiinstanzfähig	Der Baustein kann mit der Aufrufoption "Multiinstanz" aufgerufen werden. Dieses Attribut kann nicht geändert werden.
	Erweiterte Statusinformation erstellen	Ermöglicht das Beobachten aller Variablen in einem SCL-Baustein. Der Speicherbedarf des Programms und die Ablaufzeiten erhöhen sich jedoch durch diese Option.
	ARRAY-Grenzen prüfen	Prüft zur Laufzeit eines SCL-Bausteins, ob Feldindizes in dem für ein ARRAY deklarierten Bereich liegen. Wenn ein Feldindex außerhalb des zulässigen Bereichs liegt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.
	ENO automatisch setzen/ ENO automatisch für SCL-Bausteine und SCL-Netzwerke setzen	Prüft zur Laufzeit eines SCL-Bausteins, ob Fehler bei der Bearbeitung bestimmter Anweisungen auftreten. Wenn ein Laufzeitfehler auftritt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.
	Speicherplatzminimierten DB erzeugen	Erzeugt Instanz-Datenbausteine für GRAPH-Bausteine der S7-300 und S7-400 in minimiertem Format. Diese Option reduziert den Speicherbedarf des GRAPH-FBs erheblich, allerdings erhalten Sie nur eingeschränkte Programmstatus-Informationen.
	Schritte überspringen	Wenn in einem GRAPH-Baustein die Transitionen vor und nach einem Schritt gleichzeitig gültig werden, wird der Schritt nicht aktiv und damit übersprungen.
	Quittierpflicht bei Supervisionsfehlern	Tritt während des Betriebs eines GRAPH-Bausteins ein Supervisionsfehler auf, muss er quittiert werden, bevor das Programm weiter ausgeführt wird.
	Permanente Bearbeitung aller Interlocks im Handbetrieb	Bewirkt, dass in einem GRAPH-Baustein Interlock-Bedingungen im Handbetrieb permanent überprüft werden.
	Erstwerterfassung	Der Signalzustand von Operanden bzw. das Ergebnis von Vergleichern aus einem Schritt oder einer Tran-

Gruppe	Eigenschaft	Beschreibung
	(Nur ab Bausteinversion V4.0 aktivierbar)	sition wird in den statischen Parametern "CRIT_LOC", "CRIT_LOC_ERR", "CRIT" und "CRIT_FLT" der GRAPH-Bausteinschnittstelle gespeichert.
	Sperre der Betriebsartenwahl	Verhindert die Auswahl einer Betriebsart eines GRAPH-Bausteins.
	Name des Erweiterungsbausteins	Legt den Erweiterungsbaustein fest, über den Sie auf die unsichtbaren Parameter in der Bausteinschnittstelle des GRAPH-Funktionsbausteins zugreifen können.
	Datenbaustein im Gerät schreibgeschützt	Zeigt an, ob der Datenbaustein im Zielsystem schreibgeschützt ist und während des Programmablaufs nicht überschrieben werden kann. Die Option ist nur für Datenbausteine verfügbar.
	Nur im Ladespeicher ablegen	Bei Aktivierung wird der Datenbaustein nur im Ladespeicher abgelegt, belegt keinen Platz im Arbeitsspeicher und wird nicht in das Programm eingebunden. Möglichkeiten zum Hantieren der Datenbausteine bietet die Task Card "Anweisungen" in der Palette "Erweiterte Anweisungen > Datenbausteinfunktionen". Die Option ist nur für Datenbausteine verfügbar.
	DB erreichbar aus OPC UA	Bei aktiviertem Optionskästchen ist der Datenbaustein als ganzes Objekt aus OPC UA erreichbar. Die einzelnen Variablen des Datenbaustein können anschließend individuell für OPC UA freigegeben oder verriegelt werden. Die Option ist nur für Datenbausteine verfügbar.
	Startinformation	Bei S7-1500-CPUs legen Sie hier die Struktur der Startinformation des Organisationsbausteins fest: entweder wie bei S7-300 und S7-400-CPUs oder optimierte Startinformation.
	Priorität	Zeigt bei Organisationsbausteinen die eingestellte Priorität an. Ob Sie die Priorität ändern können, hängt von der verwendeten CPU-Familie und von der Art des Organisationsbausteins ab.
	Parameterversorgung über Register	Ermöglicht in einem AWL-Baustein der S7-1500 die Parameterversorgung über Register. Dadurch können die Anweisungen "Bausteine bedingt aufrufen" (CC) und "Bausteine unbedingt aufrufen" (UC) im Baustein verwendet werden.
	Rücklesbarkeit aktivieren	Ermöglicht es ihnen, einzelne Parameter des Bausteins als "rücklesbar" zu kennzeichnen. Die Funktion "Variablen zurücklesen" ist relevant, wenn der Baustein in einem CFC-Plan genutzt wird.
	Bausteindarstellung	Legt fest, wie der Baustein in einem CFC-Plan dargestellt wird. Weitere Informationen finden Sie in der Hilfe zu CFC auf der Seite "Darstellungsvarianten des Bausteinsymbols im CFC-Plan".

Gruppe	Eigenschaft	Beschreibung
Uhrzeitalarm	Uhrzeitalarm	Parameter des Uhrzeitalarm-OB: aktiv (ja oder nein), Ausführung, Startdatum und Uhrzeit, Lokalzeit bzw. Systemzeit
Weckalarm	Weckalarm	Zykluszeit und Phasenverschiebung des Weckalarm-OB
Startereignisse	Startereignisse	Anzeige der Startereignisse des Prozessalarm-OB
Taktsynchronität	Taktsynchronität	Parameter des Taktsynchronalarm-OB ein: Applikationszyklus, Automatische Einstellung (ja oder nein), Verzögerungszeit. Außerdem wird das PROFINET IO-System bzw. DP-Mastersystem angezeigt, dessen IO-Devices bzw. DP-Slaves dem Taktsynchronalarm-OB zugeordnet sind.
Laden ohne Reinitialisierung	Speicherreserve	Größe der Reserve im Standardspeicher, die für Schnittstellenerweiterungen genutzt werden kann. Die Anzahl der aktuell verfügbaren Bytes wird in Klammern angegeben. Diese Angabe wird bei jedem Übersetzungsvorgang aktualisiert.
	Laden ohne Reinitialisierung für remanente Variablen aktivieren	Ermöglicht die Definition einer Reserve im remanenten Speicher.
	Reserve im remanenten Speicher	Größe der Reserve im remanenten Speicher, die für Schnittstellenerweiterungen genutzt werden kann. Die Anzahl der aktuell verfügbaren Bytes wird in Klammern angegeben. Diese Angabe wird bei jedem Übersetzungsvorgang aktualisiert.

Siehe auch

Grundlagen zu Bausteineigenschaften (Seite 7214)

Zeitstempel von Bausteinen (Seite 7219)

Bausteineigenschaften anzeigen und bearbeiten (Seite 7222)

Grundlagen zum Bausteinzugriff (Seite 55)

Allgemeine Einstellungen für die PLC-Programmierung (Seite 7150)

7.2.3 Zeitstempel von Bausteinen

Einführung

Bausteine erhalten verschiedene Zeitstempel, an denen Sie erkennen können, wann der Baustein erstellt und wann er zuletzt geändert wurde. Diese Zeitstempel werden auch für die automatische Konsistenzprüfung vor einem Übersetzungsvorgang verwendet.

Zeitstempel für Codebausteine

Für Codebausteine (OBs, FBs, FCs) werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Code/Daten: Datum der letzten Änderung
- Laderelevant: Datum der letzten Änderung

Beim Übersetzen wird ein Zeitstempelkonflikt angezeigt, wenn der Zeitstempel für den aufrufenden Baustein älter ist als der Zeitstempel für die Schnittstelle des aufgerufenen Bausteins.

Die Zeitstempel für Codebausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung des Bausteins entspricht immer dem Zeitstempel der Schnittstelle oder des Codes, abhängig davon, welcher Bereich zuletzt geändert wurde.
- Schnittstelle: Der Zeitstempel der Schnittstelle wird bei jeder Änderung der Schnittstelle aktualisiert. Auch wenn Sie die Änderung der Schnittstelle manuell rückgängig machen, z. B. den Namen erneut ändern, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Code/Daten: Der Zeitstempel des Codes wird bei jeder Änderung des Bausteincodes aktualisiert. Auch wenn Sie die Änderung des Codes manuell rückgängig machen, z. B. eine Anweisung wieder entfernen, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Laderelevant: Der Zeitstempel für "Laderelevant" wird bei jeder Änderung eines Codebausteins geändert. Diese Änderungen können den Code bzw. die Daten oder die Schnittstelle betreffen.

Zeitstempel für globale Datenbausteine

Für globale Datenbausteine werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Daten: Datum der letzten Änderung
- Laderelevant: Datum der letzten Änderung

Beim Übersetzen eines globalen Datenbausteins, der auf einem PLC-Datentyp basiert, wird ein Zeitstempelkonflikt angezeigt, wenn der Zeitstempel des globalen Datenbausteins älter als der Zeitstempel für den verwendeten PLC-Datentyp ist.

Die Zeitstempel für globale Datenbausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines globalen Datenbausteins entspricht immer des Zeitstempels für die Schnittstelle und der Daten.
- Schnittstelle und Daten: Die Zeitstempel der Schnittstelle und der Daten werden bei jeder Änderung des globalen Datenbausteins aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. eine Variable wieder entfernen, ist das eine Änderung, die die Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, werden die Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Laderelevant: Der Zeitstempel für "Laderelevant" wird bei jeder Änderung eines globalen Datenbausteins geändert. Diese Änderungen können die Daten oder die Schnittstelle betreffen.

Zeitstempel für Instanz-Datenbausteine

Für Instanz-Datenbausteine werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Daten: Datum der letzten Änderung
- Laderelevant: Datum der letzten Änderung

Beim Übersetzen eines Instanz-Datenbausteins wird ein Zeitstempelkonflikt angezeigt, wenn die Zeitstempel für die Schnittstellen des Instanz-Datenbausteins und des Funktionsbausteins nicht identisch sind.

Die Zeitstempel für Instanz-Datenbausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines Instanz-Datenbausteins entspricht immer des Zeitstempels für die Schnittstelle und der Daten.
- Schnittstelle und Daten: Die Zeitstempel der Schnittstelle und der Daten werden bei jeder Änderung des Instanz-Datenbausteins aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. die Remanenz einer Variablen wieder aufheben, ist das eine Änderung, die die Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, werden die Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Laderelevant: Der Zeitstempel für "Laderelevant" wird bei jeder Änderung eines Instanz-Datenbausteins geändert. Diese Änderungen können die Daten oder die Schnittstelle betreffen.

Zeitstempel für PLC-Datentypen

Für PLC-Datentypen werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Laderelevant: Datum der letzten Änderung

Die Zeitstempel für PLC-Datentypen werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines PLC-Datentyps entspricht immer dem Zeitstempel der Schnittstelle.
- Schnittstelle: Der Zeitstempel der Schnittstelle wird bei jeder Änderung des PLC-Datentyps aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. Inhalte des PLC-Datentyps wieder löschen, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Laderelevant: Der Zeitstempel für "Laderelevant" wird bei jeder Änderung eines PLC-Datentyps geändert.

Siehe auch

Grundlagen zu Bausteineigenschaften (Seite 7214)

Übersicht über die Bausteineigenschaften (Seite 7215)

Bausteineigenschaften anzeigen und bearbeiten (Seite 7222)

Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

7.2.4 Bausteineigenschaften anzeigen und bearbeiten

Die Eigenschaften der Bausteine sind baustein- und CPU-spezifisch. Daher sind nicht alle Eigenschaften für alle Bausteine und in allen CPU-Familien verfügbar. Eigenschaften, die Sie nur anzeigen können, sind schreibgeschützt.

Eigenschaften eines geschlossenen Bausteins anzeigen und bearbeiten

Um die Eigenschaften eines geschlossenen Bausteins anzuzeigen oder zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Baustein, dessen Eigenschaften Sie anzeigen oder bearbeiten möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf eine Gruppe, deren Eigenschaften Sie anzeigen oder bearbeiten möchten.
5. Ändern Sie die entsprechende Eigenschaft.
6. Bestätigen Sie Ihre Eingaben mit "OK".

Eigenschaften eines geöffneten Bausteins anzeigen und bearbeiten

Um die Eigenschaften eines geöffneten Bausteins anzuzeigen oder zu bearbeiten, gehen Sie folgendermaßen vor:

1. Aktivieren Sie im Menü "Ansicht" die Option "Inspektorfenster".
   Das Inspektorfenster wird geöffnet.
2. Klicken Sie auf das Register "Eigenschaften".
   Die Eigenschaften des Bausteins werden im Register "Eigenschaften" des Inspektorfensters angezeigt.
3. Klicken Sie in der Bereichsnavigation auf eine Gruppe, deren Eigenschaften Sie anzeigen oder bearbeiten möchten.
4. Ändern Sie die entsprechende Eigenschaft.

Ergebnis

Die Eigenschaften des Bausteins werden geändert. Die Änderungen werden jedoch erst zusammen mit dem Projekt gespeichert.

Siehe auch

Grundlagen zu Bausteineigenschaften (Seite 7214)

Übersicht über die Bausteineigenschaften (Seite 7215)

Zeitstempel von Bausteinen (Seite 7219)

7.3 Bausteine verwalten

7.3.1 Bausteine öffnen

Sie können sowohl die Bausteine im Projekt (Offline-Bausteine) als auch die Bausteine im Gerät (Online-Bausteine) öffnen.

Um einen Offline-Baustein zu öffnen, haben Sie folgende Möglichkeiten:

- Offline-Baustein direkt öffnen
  Sie können einen Baustein direkt öffnen, wenn Sie den entsprechenden Bausteinordner in der Projektnavigation oder im Übersichtsfenster geöffnet haben.
- Offline-Baustein suchen und öffnen
  Sie können innerhalb eines Projekts, eines Geräts, des Ordners "Programmbausteine" und im Programmiereditor nach Bausteinen suchen und diese dann öffnen.

Beachten Sie beim Öffnen von Online-Bausteinen die folgenden Besonderheiten:

- Sie können Online-Bausteine nicht bearbeiten.
- Durch das gemeinsame parallele Arbeiten auf einer CPU ist es möglich, dass ein weiterer Anwender einen Ladevorgang auf die gewählte CPU ausführt. Dadurch kann es vorkommen, dass derOnline- Baustein, den Sie offen haben, durch den Ladevorgang gelöscht wird, wenn er nur im Gerät vorhanden ist. In diesem Fall wird der Online-Baustein geschlossen und Sie erhalten eine Meldung im Inspektorfenster.

Offline-Baustein direkt öffnen

Um einen Offline-Baustein direkt zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner mit dem Baustein, den Sie öffnen möchten, in der Projektnavigation oder im Übersichtsfenster.
2. Doppelklicken Sie auf den Baustein, den Sie öffnen möchten.
   Der Baustein wird im Programmiereditor geöffnet.

Offline-Baustein suchen und öffnen

Um nach einem Offline-Baustein zu suchen und ihn dann zu öffnen, gehen Sie folgendermaßen vor:

1. Sie können den Suchvorgang auf die folgenden beiden Arten starten:
   - Wählen Sie in der Projektnavigation im Kontextmenü des Projekts, eines Geräts oder des Ordners "Programmbaustein" den Befehl "Öffne Baustein/PLC-Datentyp" oder drücken Sie die Taste <F7>.
   - Drücken Sie im geöffneten Programmiereditor die Taste <F7>.

   Der Dialog "Öffne Baustein/PLC-Datentyp" wird geöffnet.
2. Geben Sie den Namen, die Adresse oder den Typ des Bausteins ein, den Sie suchen. Mit jedem eingegebenen Buchstaben wird die Bausteinliste weiter gefiltert. Wenn es keinen Baustein gibt, der mit Ihrer Eingabe übereinstimmt, wird die Bausteinliste geschlossen. Sie können die vollständige Bausteinliste jederzeit mit einem Klick auf die Schaltfläche rechts neben dem Textfeld einblenden. Beachten Sie jedoch, dass dann keine Filterung erfolgt. Möchten Sie wieder nach Ihren Eingaben filtern, klicken Sie erneut auf die Schaltfläche.
3. Klicken Sie in der Bausteinliste auf den Baustein, den Sie öffnen möchten.
   Der Baustein wird im Programmiereditor geöffnet und in der Projektnavigation als Selektion angezeigt.

Online-Baustein öffnen

Um einen Online-Baustein zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Online-Zugänge" in der Projektnavigation.
2. Klicken Sie auf das Pfeilsymbol links neben der Schnittstelle, um alle Objekte sichtbar zu machen, die unterhalb der Schnittstelle angeordnet sind.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer aktualisieren" unterhalb der Schnittstelle.
   Alle Geräte, die über diese Schnittstelle erreichbar sind, werden angezeigt.
4. Öffnen Sie den Ordner des Geräts, das den Baustein enthält, den Sie öffnen möchten.
5. Öffnen Sie den Ordner "Programmbausteine".
   Alle Bausteine des Geräts werden angezeigt.
6. Doppelklicken Sie auf den Baustein, den Sie öffnen möchten.
   Der Baustein wird im Programmiereditor geöffnet.

Siehe auch

Bausteine speichern (Seite 7226)

Bausteine schließen (Seite 7226)

Bausteine umbenennen (Seite 7227)

Bausteine offline löschen (Seite 7227)

Bausteine online löschen (Seite 7228)

Know-how-geschützte Bausteine öffnen (Seite 7256)

7.3.2 Bausteine speichern

Bausteine werden immer zusammen mit dem Projekt gespeichert. Auch fehlerhafte Bausteine können gespeichert werden. Dadurch kann die Fehlerbehebung zu jedem beliebigen Zeitpunkt erfolgen.

Vorgehen

Um einen Baustein zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern" oder "Speichern unter".
   Siehe auch: Projekte speichern

Siehe auch

Bausteine öffnen (Seite 7224)

Bausteine schließen (Seite 7226)

Bausteine umbenennen (Seite 7227)

Bausteine offline löschen (Seite 7227)

Bausteine online löschen (Seite 7228)

7.3.3 Bausteine schließen

Vorgehen

Um einen Baustein zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Programmiereditors auf die Schaltfläche "Schließen".

Hinweis

Beachten Sie, dass der Baustein beim Schließen nicht gespeichert wird.

Siehe auch

Bausteine öffnen (Seite 7224)

Bausteine speichern (Seite 7226)

Bausteine umbenennen (Seite 7227)

Bausteine offline löschen (Seite 7227)

Bausteine online löschen (Seite 7228)

7.3.4 Bausteine umbenennen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um den Namen eines Bausteins zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
   In der Projektnavigation wird der Bausteinname in ein Eingabefeld geändert.
3. Geben Sie den neuen Namen des Bausteins ein.
4. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.

Ergebnis

Der Name des Bausteins wird an allen Verwendungsstellen im Programm geändert.

Siehe auch

Bausteine öffnen (Seite 7224)
Bausteine speichern (Seite 7226)
Bausteine schließen (Seite 7226)
Bausteine offline löschen (Seite 7227)
Bausteine online löschen (Seite 7228)

7.3.5 Bausteine offline löschen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Baustein zu löschen, der offline vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den Baustein, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".
   Der Baustein wird offline aus dem Projekt gelöscht.

Hinweis

Beachten Sie beim Löschen von Organisationsbausteinen, dass ihnen Ereignisse zugeordnet sein können. Wenn Sie solche Organisationsbausteine löschen, kann das Programm nicht auf parametrierte Ereignisse reagieren.

Siehe auch

Bausteine öffnen (Seite 7224)

Bausteine speichern (Seite 7226)

Bausteine schließen (Seite 7226)

Bausteine umbenennen (Seite 7227)

Bausteine online löschen (Seite 7228)

Bausteine laden für S7-1200/1500 (Seite 7871)

7.3.6 Bausteine online löschen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Baustein zu löschen, der online vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den Baustein, den Sie im Gerät löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Dialog "Löschen" wird geöffnet.
3. Aktivieren Sie das Optionsfeld "Löschen aus Gerät".
4. Klicken Sie auf "Ja".
   Der Baustein wird online aus dem Gerät gelöscht.

Siehe auch

Bausteine öffnen (Seite 7224)

Bausteine speichern (Seite 7226)

Bausteine schließen (Seite 7226)

Bausteine umbenennen (Seite 7227)

Bausteine offline löschen (Seite 7227)

7.3.7 CPU-Datenbausteine löschen

Sie können CPU-Datenbausteine sowohl offline als auch online löschen.

CPU-Datenbausteine offline löschen

Um einen CPU-Datenbaustein zu löschen, der offline im Projekt vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den CPU-Datenbaustein, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".
   Der CPU-Datenbaustein wird offline aus dem Projekt gelöscht.

CPU-Datenbaustein online löschen

Um einen CPU-Datenbaustein zu löschen, der online vorhanden ist, gehen Sie folgendermaßen vor:

1. Sellen Sie eine Online-Verbindung zum Gerät her, das den CPU-Datenbaustein enthält, den Sie löschen möchten.
2. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den CPU-Datenbaustein, den Sie im Gerät löschen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Dialog "Löschen" wird geöffnet.
4. Aktivieren Sie das Optionsfeld "Löschen aus Gerät".
5. Klicken Sie auf "Ja".
   Der CPU-Datenbaustein wird online aus dem Gerät gelöscht.

Siehe auch

CPU-Datenbausteine (Seite 54)

7.4 Externe Quelldateien verwenden

7.4.1 Grundlagen zur Verwendung von externen Quelldateien

Funktion

Für die textuellen Programmiersprachen AWL und SCL haben Sie die Möglichkeit, den Programmcode in einem beliebigen ASCII-Editor einzugeben und als externe Quelldatei zu speichern. Sie können dabei z. B. folgende Arbeiten durchführen:

- Variablen deklarieren
- Bausteineigenschaften festlegen
- Bausteine programmieren

Hinweis

Damit im TIA Portal aus der externen Quelldatei Bausteine generiert werden können, muss die Quelldatei mit einer der folgenden Zeichencodierungen gespeichert werden:

- ANSI
- UTF-8 mit BOM (Byte Order Mark)
- UTF-16 mit BOM, Little- und Big-Endian-Format
- UTF-32 mit BOM, Little- und Big-Endian-Format
- UCS2 im Little- und Big-Endian-Format

Diese externen Quelldateien können Sie in Ihr Projekt importieren und daraus Bausteine generieren. Auf diese Weise können Sie mehrere Bausteine aus einer Quelldatei erzeugen. Beachten Sie beim Generieren von Bausteinen aus einer Quelldatei folgende Besonderheiten:

- Wenn ein Baustein mit dem gleichen symbolischen Namen bereits im Projekt vorhanden ist, wird der Baustein im Projekt überschrieben.
- Wenn ein Baustein in der Quelldatei mit seiner absoluten Bausteinnummer statt mit einem symbolischen Namen programmiert wurde und diese Nummer durch einen Baustein im Projekt bereits vergeben ist, erhält der neu generierte Baustein den nächst freien symbolischen Namen.
- Wenn Sie in der externen Quelldatei die Zugriffsart für einen Baustein nicht explizit definiert haben, wird die Zugriffsart für den Baustein abhängig von der CPU-Familie gesetzt:
  - Bausteine, die für eine CPU der Baureihe S7-1200/1500 generiert werden, erhalten als Voreinstellung die Zugriffsart "optimiert".
  - Bausteine, die für eine CPU der Baureihe S7-300/400 generiert werden, erhalten als Voreinstellung die Zugriffsart "standard".

  Eine Ausnahme bilden Organisationsbausteine, die in einem solchen Fall unabhängig von der CPU-Familie als Voreinstellung immer die Zugriffsart "standard" erhalten. Sie haben die Möglichkeit den Bausteinzugriff bei Bedarf manuell zu ändern.
- Eventuell werden nicht alle Kommentare aus der Quelldatei in den Baustein übernommen.

- Wenn Sie in der externen Quelldatei absolute Adressierung verwenden, wird bei der Generierung des Bausteins für jede absolute Adresse eine symbolische Variable erzeugt. Die Namen dieser Variablen setzen sich zusammen aus "Tag_" und einem Zeitstempel. Dadurch ergeben sich eventuell relativ lange Variablennamen, die Sie jedoch bei Bedarf manuell ändern können.
- Wenn Sie in der externen Quelldatei Anweisungen in einer anderen Version als im Zielgerät verwenden, kann dies zu Übersetzungsfehler führen. Korrigieren Sie in diesem Fall die entsprechenden Anweisungen und starten Sie den Übersetzungsvorgang erneut. Alternativ können Sie für das Zielgerät eine andere Version auswählen.
- CPUs der Baureihe S7-300/400 akzeptieren beim Import Organisationsbausteine mit beliebigem Namen. Wenn Sie einen OB mit Standardnamen importieren (z. B. CYCL_EXC oder CYC_INT5), wird automatisch die korrekte OB-Klasse zugewiesen. Wenn Sie einen OB mit einem unbekannten Namen importieren, wird ein OB 0 angelegt. Diese Nummer können Sie nach dem Import ändern und so die gewünschte OB-Klasse zuweisen. Um die OB-Nummer zu ändern, wählen Sie im Kontextmenü den Befehl "Eigenschaften".
- Beim Import einer SCL-Quelle erfolgt keine automatische Überprüfung der Quelle. Somit können daraus generierte Bausteine Fehler enthalten, die erst bei der nächsten Übersetzung durch Fehlermeldungen angezeigt werden.

Zusätzlich haben Sie auch die Möglichkeit, bestehende Bausteine als externe Quelldateien zu speichern. Dabei können Sie entweder nur den Programmcode des ausgewählten Bausteins in einer Quelldatei speichern oder zusätzlich den Programmcode aller vom ausgewählten Baustein abhängigen Bausteinen und PLC-Datentypen. Beim Speichern von mehreren Bausteinen in eine Quelldatei müssen alle Bausteine die gleiche Programmiersprache verwenden und keiner der Bausteine darf einen Know-how-Schutz besitzen.

Siehe auch

Regeln zur Programmierung von externen Quelldateien (Seite 7231)

Baustein als externe Quelldatei speichern (Seite 7233)

Externe Quelldateien einbinden (Seite 7235)

Externe Quelldateien öffnen und bearbeiten (Seite 7235)

Bausteine aus externen Quelldateien generieren (Seite 7236)

7.4.2 Regeln zur Programmierung von externen Quelldateien

Eine externe Quelldatei besteht prinzipiell aus fortlaufendem Text. Damit sie sich in Bausteine übersetzen lässt, müssen jedoch bestimmte Strukturen und Syntaxvorschriften eingehalten werden.

Syntaxregeln

Die Syntax der Anweisungen in den externen Quelldateien ist der beim Erstellen des Anwenderprogramms im Programmiereditor mit AWL oder SCL sehr ähnlich. Beachten Sie jedoch folgende zusätzlichen Syntaxregeln:

- Bausteinaufruf
  Übergeben Sie im ASCII-Editor bei einem Bausteinaufruf die Parameter in definierter Reihenfolge. Ansonsten stimmen möglicherweise die Kommentarzuordnungen dieser Zeilen nicht überein.
  Geben Sie die Parameter in Klammern an. Die einzelnen Parameter werden dabei durch ein Komma voneinander getrennt.
- Groß- oder Kleinschreibung
  Der Programmiereditor berücksichtigt allgemein keine Groß- oder Kleinschreibung. Ausgenommen davon sind Sprungmarken. Bei Eingabe von Zeichenketten (Datentyp "STRING") muss ebenfalls auf Groß- und Kleinschreibung geachtet werden.
  Schlüsselwörter werden in Großbuchstaben dargestellt. Beim Übersetzen wird die Groß- und Kleinschreibung jedoch nicht berücksichtigt, sodass Sie Schlüsselwörter in Groß- oder Kleinbuchstaben oder gemischt angeben können.
- Semikolon
  Kennzeichnen Sie das Ende jeder Anweisung und jeder Variablendeklaration mit einem Semikolon. Sie können mehrere Anweisungen pro Zeile eingeben.
- Schrägstriche
  Beginnen Sie jeden Kommentar mit zwei Schrägstrichen (//) und beenden Sie jede Kommentareingabe mit der Taste <Return>.
- Verwendung von String-Konstanten
  Um bei der Verwendung von String-Konstanten Übersetzungsfehler zu vermeiden, geben Sie die Texte in der Sprache des Zielprojekts ein. Für CPUs der Baureihe S7-1200/1500 können Sie über den Präfix "WString#" explizit den Datentyp WSTRING verwenden:
  ```
  Operand := WString#'<String-Konstante>';
  ```

Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 7230)

Baustein als externe Quelldatei speichern (Seite 7233)

Externe Quelldateien einbinden (Seite 7235)

Externe Quelldateien öffnen und bearbeiten (Seite 7235)

Bausteine aus externen Quelldateien generieren (Seite 7236)

7.4.3 Baustein als externe Quelldatei speichern

Sie haben die folgenden Möglichkeiten AWL- und SCL-Bausteine als externe Quelldateien zu speichern:

- Baustein als Text kopieren
  Beachten Sie, dass bei geschützten Bausteinen nur die Informationen in die Quelldatei übernommen werden, die Sie auch ohne Passwort sehen können.
- Externe Quelldatei aus Bausteinen generieren
  Sie können entweder nur den Programmcode des ausgewählten Bausteins in eine externe Quelldatei speichern oder zusätzlich den Programmcode aller vom ausgewählten Baustein abhängigen Bausteinen und PLC-Datentypen. Dabei müssen alle Bausteine die gleiche Programmiersprache verwenden und keiner der Bausteine darf einen Know-how-Schutz besitzen.

Hinweis

Beachten Sie bei der Verwendung der tabellarischen Bausteinschnittstelle, dass nicht alle Initialwerte mit in die Quelle exportiert werden, z. B. bei PLC-Datentypen.

Baustein als Text kopieren

Um einen Baustein als Text zu kopieren und in einer externen Quelldatei zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf den Baustein, den Sie in einer externen Quelldatei speichern möchten.
2. Wählen Sie im Kontextmenü den Befehl "Als Text kopieren".
3. Öffnen Sie einen externen Texteditor.
4. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
5. Speichern Sie die Datei mit einer der folgenden Dateinamenerweiterungen:
   - ".scl", wenn Sie eine Quelldatei für einen SCL-Baustein erzeugen möchten
   - ".awl", wenn Sie eine Quelldatei für einen AWL-Baustein erzeugen möchten
   - ".DB", wenn Sie eine Quelldatei für einen Datenbaustein erzeugen möchten
   - ".UDT", wenn Sie eine Quelldatei für einen PLC-Datentyp erzeugen möchten

Externe Quelldatei aus Bausteinen generieren

Um aus AWL- oder SCL-Bausteinen externe Quelldateien zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation oder im Übersichtsfenster die AWL- oder SCL-Bausteine, aus denen Sie eine externe Quelldatei generieren möchten.

   Hinweis

   Selektieren Sie nur Bausteine mit der gleichen Programmiersprache und ohne Know-how-Schutz.

2. Wählen Sie im Kontextmenü den Befehl "Quelle aus Bausteinen generieren > Nur selektierte Bausteine", um nur den ausgewählten Baustein in einer externen Quelldatei zu speichern. Oder wählen Sie den Befehl "Quelle aus Bausteinen generieren > Einschließlich abhängige Bausteine", wenn Sie auch den Programmcode der abhängigen Bausteinen und referenzierten PLC-Datentypen in der externen Quelldatei speichern möchten.
   Der Dialog "Speichern unter" wird geöffnet.
3. Geben Sie einen Pfad und einen Namen für die externe Quelle an.
4. Klicken Sie auf "OK".

Sie können auch einen geöffneten AWL- oder SCL-Baustein als externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Wählen Sie im Programmiereditor in der Funktionsleiste in der Klappliste "Quelle aus Baustein generieren" den Befehl "Quelle aus diesem Baustein generieren", um nur den geöffneten Baustein in einer externen Quelldatei zu speichern. Oder wählen Sie den Befehl "Quelle aus diesem Baustein und allen abhängigen Bausteinen generieren", wenn Sie auch den Programmcode der abhängigen Bausteinen und referenzierten PLC-Datentypen in der externen Quelldatei speichern möchten.
   Der Dialog "Speichern unter" wird geöffnet.
2. Geben Sie einen Pfad und einen Namen für die externe Quelle an.
3. Klicken Sie auf "OK".

Ergebnis

Abhängig von der gewählten Option wird entweder nur der Baustein oder der Baustein zusammen mit den abhängigen Bausteinen und PLC-Datentypen als externe Quelldatei gespeichert. Diese Quelldatei können Sie im TIA Portal in ein Projekt einbinden und daraus erneut Bausteine generieren. Beachten Sie aber, dass Sie AWL-Quelldateien nur in S7-300/400/1500-CPUs verwenden können.

Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 7230)

Regeln zur Programmierung von externen Quelldateien (Seite 7231)

Externe Quelldateien einbinden (Seite 7235)

Externe Quelldateien öffnen und bearbeiten (Seite 7235)

Bausteine aus externen Quelldateien generieren (Seite 7236)

7.4.4 Externe Quelldateien einbinden

Voraussetzung

- Eine externe Quelldatei ist vorhanden und entspricht den Syntax- und Strukturregeln.
- Der Ordner "Externe Quellen" in der Projektnavigation ist geöffnet.

Vorgehen

Um eine externe Quelldatei einzubinden, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen".
   Der Dialog "Öffnen" wird geöffnet.
2. Navigieren Sie zur vorhandenen externen Quelldatei und selektieren Sie diese.
3. Bestätigen Sie Ihre Auswahl mit "Öffnen".
   Falls es bereits eine externe Quelldatei mit diesem Namen im Projekt gibt, wird der Dialog "Externe Datei hinzufügen" geöffnet. Fahren Sie dann mit Schritt 4 fort.
4. Wählen Sie aus, ob die neue externe Quelle umbenannt und eingefügt oder ob die vorhandene externe Quelle ersetzt werden soll.
5. Klicken Sie auf "OK".

Ergebnis

Die neue Quelldatei wird in den Ordner "Externe Quellen" eingefügt.

Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 7230)

Regeln zur Programmierung von externen Quelldateien (Seite 7231)

Baustein als externe Quelldatei speichern (Seite 7233)

Externe Quelldateien öffnen und bearbeiten (Seite 7235)

Bausteine aus externen Quelldateien generieren (Seite 7236)

7.4.5 Externe Quelldateien öffnen und bearbeiten

Wenn Sie Dateien mit den Dateinamenerweiterungen "awl" und "scl" mit einem Editor verknüpfen, haben Sie die Möglichkeit, externe Quelldateien mit diesen Formaten direkt zu öffnen und zu bearbeiten. Als Editor sollten Sie Notepad verwenden, da Sie mit anderen Texteditoren möglicherweise nicht gleichzeitig mehrere Quellen öffnen können.

Dadurch müssen Sie die externen Quelldateien nach einer Bearbeitung nicht erneut einbinden.

Dateien mit den Dateinamenerweiterungen "awl" und "scl" mit einem Editor verknüpfen

Um die Dateien mit den Dateinamenerweiterungen "awl" und "scl" mit einem Editor zu verknüpfen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Windows Explorer.
2. Klicken Sie mit der rechten Maustaste auf eine AWL-Datei.
3. Wählen Sie im Kontextmenü "Eigenschaften".
   Der Dialog "Eigenschaften" wird geöffnet.
4. Klicken Sie im Register "Allgemein" im Bereich "Dateityp" auf "Ändern".
   Der Dialog "Öffnen mit" wird geöffnet.
5. Wählen Sie den Texteditor, den Sie mit dem Dateityp "awl" verknüpfen möchten.
6. Bestätigen Sie Ihre Auswahl mit "OK".
7. Schließen Sie den Dialog "Eigenschaften" mit "OK".
8. Wiederholen Sie die Schritte 2 bis 7 mit einer SCL-Datei.

Externe Quelldatei öffnen und bearbeiten

Um eine externe Quelldatei zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Externe Quellen".
2. Klicken Sie doppelt auf die externe Quelldatei, die Sie öffnen möchten.
   Die externe Quelldatei wird im verknüpften Editor geöffnet und kann bearbeitet werden.

Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 7230)

Regeln zur Programmierung von externen Quelldateien (Seite 7231)

Baustein als externe Quelldatei speichern (Seite 7233)

Externe Quelldateien einbinden (Seite 7235)

Bausteine aus externen Quelldateien generieren (Seite 7236)

7.4.6 Bausteine aus externen Quelldateien generieren

Voraussetzung

- Der Ordner "Externe Quellen" in der Projektnavigation ist geöffnet.
- Eine externe Quelldatei ist vorhanden.
- Die externe Quelldatei wurde mit einer zulässigen Zeichencodierung gespeichert.

Vorgehen

Um aus einer externen Quelldatei die Bausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die externe Quelldatei, aus der Sie die Bausteine generieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Bausteine aus Quelle generieren".
3. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
4. Bestätigen Sie die Sicherheitsabfrage mit "Ja".

Ergebnis

Die Bausteine der externen Quelldatei werden generiert und in der Projektnavigation in den Ordner "Programmbausteine" eingefügt. Im Fehlerfall finden Sie Informationen zu den aufgetretenen Fehlern im Inspektorfenster. Diese Informationen beziehen sich jedoch auf die externe Quelldatei und nicht auf den generierten Baustein.

Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 7230)

Regeln zur Programmierung von externen Quelldateien (Seite 7231)

Baustein als externe Quelldatei speichern (Seite 7233)

Externe Quelldateien einbinden (Seite 7235)

Externe Quelldateien öffnen und bearbeiten (Seite 7235)

Bausteineigenschaften festlegen (Seite 7214)

Bausteine schützen 8

8.1 Schützen von Bausteinen

Einführung

Sie haben die folgenden Möglichkeit, Ihre Bausteine schützen:

- Know-how-Schutz:
  Mithilfe des Know-how-Schutzes können Sie Bausteine des Typs OB, FB, FC und globale Datenbausteine mit einem Passwort vor unbefugtem Zugriff schützen.
- Kopierschutz:
  Für S7-1200/1500-CPUs können Sie einen Kopierschutz für Codebausteine einrichten, der die Ausführung des Bausteins an die CPU oder die Memory Card mit der festgelegten Seriennummer bindet. Bei der Verwendung des Kopierschutzes für einen Baustein ist es sinnvoll, den Baustein auch mit einem Know-how-Schutz zu versehen, damit der Kopierschutz nur mit Passwort aufgehoben werden kann.
- Schreibschutz:
  Für Codebausteine können Sie einen Schreibschutz vergeben, der ein unbeabsichtigtes Ändern des Bausteins verhindert.

Beachten Sie für den Know-how-Schutz die folgende Besonderheiten:

- Instanz-Datenbausteine können Sie nicht manuell schützen, sondern sie sind vom Know-how-Schutz des zugeordneten FB abhängig. Das heißt, wenn Sie für einen know-how-geschützten FB einen Instanz-Datenbaustein erzeugen, erhält auch der Instanz-Datenbaustein den Know-how-Schutz. Dies ist unabhängig davon, ob Sie den Instanz-Datenbaustein explizit anlegen oder ob er durch einen Bausteinaufruf erzeugt wird.
- Bei globalen Datenbausteinen und bei Instanz-Datenbausteinen wirkt sich der Know-how-Schutz als Schreibschutz aus.
- ARRAY-Datenbausteine können Sie nicht mit einem Know-how-Schutz versehen.
- Der Ladespeicherbedarf kann bei know-how-geschützten Bausteinen höher sein.

Bei einem know-how-geschützten Baustein sind lediglich die folgenden Daten ohne korrektes Passwort lesbar:

- Übergabeparameter Input, Output, InOut, Return, Static
- Bausteintitel
- Bausteinkommentar
- Bausteineigenschaften
- Variablen von globalen Datenbausteinen ohne Angabe der Verwendungsstelle

Außerdem sind die folgenden Aktionen mit einem know-how-geschützten Baustein durchführbar:

- Kopieren und Löschen
- Aufrufen in einem Programm
- Offline/Online-Vergleich
- Laden

Der Code des Bausteins ist dagegen vor unbefugtem Lesen und Verändern geschützt.

Hinweis

Beachten Sie folgende Hinweise:

- S7-1200 Version 1.0 und S7-300/400 (nur GRAPH- und SCL-Bausteine): Wenn Sie einen know-how-geschützten Baustein in ein Gerät laden, werden keine Wiederherstellungsinformationen mitgeladen. Das bedeutet, dass Sie einen know-how-geschützten Baustein auch mit dem korrekten Passwort nicht wieder öffnen können, wenn Sie ihn vom Gerät laden.
- Bei einem Vergleich zwischen der Offline- und der Online-Version von know-how-geschützten Bausteinen werden nur die nicht geschützten Daten verglichen.
- Ohne Passwort ist kein weiterer Zugriff auf den Baustein möglich.
- Wenn Sie einen know-how-geschützten Baustein in eine Bibliothek einfügen, erhält auch die entstehende Kopiervorlage den Know-how-Schutz.
- In den Bausteineigenschaften finden Sie unter "Übersetzung > Bibliothekskonformität" einen Hinweis darauf, ob der Baustein als geschütztes Bibliothekselement verwendet werden kann. Dazu darf der Baustein keine Variablen aus den Operandenbereiche Ausgang (A), Eingang (E), Merker (M), Zeitfunktion (T) oder Zählfunktion (Z) verwenden und nicht auf Datenbausteine zugreifen.
- Querverweise zu verwendeten Variablen, Merkern, Ein- und Ausgängen in know-how-geschützten Bausteinen werden nur dann angezeigt, wenn die betreffenden Bausteine vor der Erzeugung der Querverweisliste durch die Eingabe des korrekten Passworts entsperrt wurden.
- Automatisches Umnummerieren und manuelles Umnummerieren ohne Passwort von know-how-geschützten Bausteinen ist nur für CPUs der Baureihe S7-1500 und S7-1200 (ab V4) möglich. Der Know-how-Schutz muss zudem mit einer TIA Portal-Version ab V13 SP1 gesetzt worden sein. Sind für einen know-how-geschützten Baustein diese Voraussetzungen nicht gegeben, ist bei einer Änderung der Nummer die ladbare Binärkomponente des Bausteins nicht mehr aktuell. In diesem Fall muss der Baustein vor dem Laden in ein Gerät neu übersetzt werden, was bei einem know-how-geschützten Baustein nur mit korrektem Passwort möglich ist. Beachten Sie dies vor allem, wenn Sie einen know-how-geschützten Baustein in ein anderes Gerät kopieren möchten, in dem es bereits einen Baustein mit dieser Nummer gibt.
- Geben Sie ein Projekt, das know-how-geschützte Bausteine enthält, immer als Projekt- oder Bibliotheksarchiv weiter. Dadurch stellen Sie sicher, dass der Know-how-Schutz nicht umgangen werden kann.
- Für einen know-how-geschützten geöffneten Baustein können Sie die Einstellungen zum Know-how-Schutz nicht ändern.

Verwenden eines Passwort-Providers

Sie haben die Möglichkeit, das TIA Portal über einen Passwort-Provider an eine externe Passwortverwaltung anzubinden. Dazu ist ein Projekt ab der Produktversion V14 SP1 notwendig.

Wenn Sie einen Passwort-Provider verwenden, können Sie beim Setzen des Know-how-Schutzes, des Kopier- oder Schreibschutzes das Passwort aus einer Liste von Passwörtern auswählen. In der Liste werden nur die Namen der Passwörter angezeigt, die Passwörter selbst sind nicht sichtbar. Beim Öffnen des Bausteins verbindet sich das TIA Portal mit dem Passwort-Provider und holt sich das entsprechende Passwort. Ist der Passwort-Provider zu diesem Zeitpunkt nicht verfügbar, können Sie den Baustein nur dann vollständig öffnen, wenn die manuelle Eingabe als Rückfallstrategie festgelegt und das entsprechende Passwort bekannt ist. Der Name des Passworts reicht hier nicht aus. Abhängig von der gewählten Sicherheitseinstellung im Passwort-Provider können Sie das Passwort für einen know-how-geschützten Baustein ändern oder den Schutz wieder aufheben.

Die Verwendung eines Passwort-Providers erhöht die Sicherheit für Ihren Programmcode, da es nicht notwendig ist, Passwörter für know-how-geschützte Bausteine an andere Bearbeiter weiterzugeben. Stattdessen können alle Bearbeiter den gleichen Passwort-Provider anbinden, um Zugriff auf die Bausteine zu erhalten. Falls ein Bearbeiter das Projekt verlässt, ist ein Passwortwechsel nicht notwendig.

Bei Problemen mit angebundenen Passwort-Providern können Sie die Aktivitäten der Passwort-Provider aufzeichnen und die Aufzeichnung an den Customer Support zur Analyse weiterleiten.

Hinweis

Sicherheitshinweis zum Bausteinschutz

Wird ein Baustein mit einem manuell eingegebenen Passwort geschützt, ist stets die manuelle Eingabe zur Bearbeitung des Bausteins oder dessen Schutzeinstellungen notwendig. Dies gilt auch dann, wenn zu einem späteren Zeitpunkt ein Password-Provider ein textuell gleiches Passwort enthält.

Siehe auch

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.2 Passwort-Provider anbinden

Als Alternative zur manuellen Passworteingabe kann ein Passwort-Provider an das TIA Portal angebunden werden. Bei der Verwendung eines Passwort-Providers können Sie aus einer Liste von verfügbaren Passwörtern ein Passwort auswählen. Beim Öffnen eines know-how-geschützten Bausteins verbindet sich das TIA Portal mit dem Passwort-Provider und holt sich das entsprechende Passwort.

Um einen Passwort-Provider anzubinden, müssen Sie ihn installieren und aktivieren. Zusätzlich ist eine Einstellungsdatei erforderlich, in der die Verwendung eines Passwort-Providers festgelegt ist.

Passwort-Provider installieren und aktivieren

Um einen Passwort-Provider zu installieren und zu aktivieren, gehen Sie folgendermaßen vor:

1. Installieren Sie den gewünschten Passwort-Provider entsprechend der Installationsanleitung des Passwort-Providers.
2. Öffnen Sie das TIA Portal.
3. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
4. Wählen Sie in der Bereichsnavigation die Gruppe "Passwort-Provider > Verfügbare Passwort-Provider".
   Der neu installierte Passwort-Provider wird in der Tabelle der verfügbaren Passwort-Provider angezeigt.
5. Aktivieren Sie das Kontrollkästchen in der Spalte "Status", um den Passwort-Provider zu verwenden.
   Der installierte Passwort-Provider ist nun aktiv und kann verwendet werden.
6. Stellen Sie die bevorzugte Passworteingabe so ein, dass der Passwort-Provider verwendet wird.
   Siehe auch: Bevorzugte Passwortvergabe festlegen (Seite 7248)

Einstellungsdatei für die Nutzung von Passwort-Providern erzeugen

Um eine Einstellungsdatei für die Nutzung von Passwort-Providern zu erzeugen, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine XML-Datei mit unten genanntem Inhalt. Verwenden Sie die Kodierung "UTF-8".
2. Legen Sie im Setting-Tag mit dem Namen "EnablePasswordPreference" fest, ob ein Anwender zwischen manueller Passwortvergabe und der Passwortvergabe über einen Passwort-Provider umschalten kann.
3. Legen Sie im Setting-Tag mit dem Namen "PasswordPreference" fest, welche Art der Passwortvergabe verwendet werden soll.
4. Speichern Sie die XML-Datei.
5. Speichern Sie die Datei mit dem Namen "CorporateSettings.xml" in folgendem Verzeichnis auf dem Computer:
   C:\ProgramData\Siemens\Automation\Portal V15_1\CorporateSettings\

Inhalt der Einstellungsdatei

Die Einstellungsdatei muss folgenden Inhalt besitzen:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<Document>
   <Settings.Settings ID="0">
       <ObjectList>
          <Settings.General ID="1" AggregationName="General">
             <ObjectList>
                <Settings.PasswordProviders ID="3" AggregationeName="PasswordProviders">
                   <AttributeList>
                      <EnablePasswordPreference>
                         <Value>true</Value>
                      </EnablePasswordPreference>
                      <DefaultPasswordPreference>
                         <Value>Advanced</Value>
                      </DefaultPasswordPreference>
                   </AttributeList>
                </Settings.PasswordProviders>
             </ObjectList>
          </Settings.General>
       </ObjectList>
    </Settings.Settings>
</Document>
```

Die Einstellungen für den Passwort-Provider legen Sie Innerhalb des Elements "Settings.PasswordProviders" fest:

- Für "EnablePasswordPreference" sind die folgenden Werte möglich:
  - "true": Der Anwender kann im TIA Portal zwischen manueller Passwortvergabe und Passwortvergabe über Passwort-Provider umschalten.
  - "false": Das Umschalten im TIA Portal zwischen manueller Passwortvergabe und Passwortvergabe über Passwort-Provider ist nicht erlaubt.

  Die Standardeinstellung ist "false".
- Für "DefaultPasswordPreference" sind die folgenden Werte möglich:
  - "Manual": Die Vergabe von Passwörtern im TIA Portal erfolgt manuell.
  - "Advanced": Die Vergabe von Passwörtern im TIA Portal erfolgt über einen Passwort-Provider.
  - "AdvancedWithManualFallback": Die Vergabe von Passwörtern im TIA Portal erfolgt über einen Passwort-Provider. Wenn der Passwort-Provider nicht verfügbar ist, kann über eine manuelle Passworteingabe auf den geschützten Baustein zugegriffen werden.

  Die Standardeinstellung ist "Manual".
  Wenn die Datei "CorporateSettings.xml" bereits vorhanden ist und bereits andere Einstellungen enthält, ergänzen Sie das Element "Settings.PasswortProviders" innerhalb des Elements "ObjectList>.

Siehe auch

Schützen von Bausteinen (Seite 7239)
Verfügbare Passwort-Provider anzeigen (Seite 7245)
Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)
Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)
Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)
Know-how-Schutz für Bausteine einrichten (Seite 7255)
Know-how-geschützte Bausteine öffnen (Seite 7256)
Know-how-geschützte Bausteine drucken (Seite 7258)
Passwort ändern (Seite 7260)
Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.3 Verfügbare Passwort-Provider anzeigen

Sie können sich die Passwort-Provider anzeigen lassen, die im TIA Portal verfügbar sind.

Voraussetzung

Die Passwortvergabe über Passwort-Provider wurde über eine Einstellungsdatei festgelegt.

Vorgehen

Um die verfügbaren Passwort-Provider anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Passwort-Provider > Verfügbare Passwort-Provider".
   Die verfügbaren Passwort-Provider werden in der Übersichtstabelle angezeigt.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.4 Aktivitäten der Passwort-Provider aufzeichnen

Bei Problemen im Zusammenhang mit angebundenen Passwort-Providern können Sie die Aktivitäten der Passwort-Provider aufzeichnen. Dabei werden keine Passwörter aufgezeichnet. Die Aufzeichnung können Sie dann an den Customer Support zur Analyse weiterleiten.

Voraussetzung

Ein Passwort-Provider ist angebunden und aktiviert.

Aufzeichnung der Aktivitäten der Passwort-Provider aktivieren

Um die Aufzeichnung zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Passwort-Provider > Analyse der Passwort-Provider".
3. Aktivieren Sie das Optionskästchen "Aktivitäten des Passwort-Providers aufzeichnen".
   Das Textfeld für den Speicherort der Aufzeichnungsdatei und die Schaltfläche "Durchsuchen" werden aktiv.
4. Geben Sie einen Speicherort für die Aufzeichnungsdatei ein oder wählen Sie ihn über die Schaltfläche "Durchsuchen" aus.

Aufzeichnung der Aktivitäten der Passwort-Provider deaktivieren

Um die Aufzeichnung zu deaktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Passwort-Provider > Analyse der Passwort-Provider".
3. Deaktivieren Sie das Optionskästchen "Aktivitäten des Passwort-Providers aufzeichnen".
   Das Textfeld für den Speicherort der Aufzeichnungsdatei und die Schaltfläche "Durchsuchen" werden inaktiv.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)
Know-how-geschützte Bausteine drucken (Seite 7258)
Passwort ändern (Seite 7260)
Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.5 Art der Passwortvergabe festlegen

Das TIA Portal bietet Ihnen bei der Art der Passwortvergabe für Bausteine beim Know-how-Schutz, beim Kopierschutz und beim Schreibschutz die folgenden Möglichkeiten:

- "Manuelle Passworteingabe"
  Bei der manuellen Passworteingabe geben Sie das Passwort direkt in den Dialog für die Passwortabfrage ein.
- "Auswahl über Passwort-Provider"
  Bei der Passworteingabe über einen externen Passwort-Provider können Sie das Passwort über einen Namen aus der Liste der verfügbaren Passwörter auswählen. Falls der Passwort-Provider nicht verfügbar ist, ist das Setzen des Schutzes nicht möglich. Auch ein späterer Zugriff auf den Baustein ist nur dann möglich, wenn die Verbindung zum Passwort-Provider besteht.
- "Auswahl über Passwort-Provider mit manueller Eingabe als Rückfallstrategie für den Zugriff"
  Wie die Passworteingabe über einen externen Passwort-Provider. Wenn der Passwort-Provider nicht verfügbar ist und Sie das Passwort kennen, können Sie jedoch den Baustein durch eine manuelle Eingabe des Passworts zur Bearbeitung öffnen.

Die manuelle Passworteingabe ist die Standardeinstellung.

Voraussetzung

Damit die Passwortvergabe über einen externen Passwort-Provider genutzt werden kann, müssen die folgenden Voraussetzungen erfüllt sein:

- Die Passwortvergabe über Passwort-Provider wurde über eine Einstellungsdatei festgelegt.
- In der Einstellungsdatei wurde der Wechsel der Art der Passwortvergabe erlaubt.
- Ein Passwort-Provider ist angebunden.

Vorgehen

Um die Art der Passwortvergabe festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Passwort-Provider > Passwortvergabe".
3. Aktivieren Sie die gewünschte Art der Passwortvergabe.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)
Know-how-Schutz für Bausteine einrichten (Seite 7255)
Know-how-geschützte Bausteine öffnen (Seite 7256)
Know-how-geschützte Bausteine drucken (Seite 7258)
Passwort ändern (Seite 7260)
Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.6 Kopierschutz für Bausteine einrichten und entfernen

Sie können für S7-1200/1500-CPUs einen Kopierschutz einrichten, mit dem das Ausführen des Bausteins an eine bestimmte CPU oder an eine bestimmte Memory Card gebunden wird. Nur wenn der Baustein sich auf dem Gerät mit der festgelegten Seriennummer befindet, kann er ausgeführt werden. Wenn Sie die Seriennummer durch einen Ladevorgang automatisch einfügen möchten, müssen Sie ein Kopierschutz-Passwort festlegen. Dieses Passwort wird bei einem Ladevorgang des Bausteins abgefragt. Sie können das Passwort ändern, ohne dafür den Know-how-Schutz des Bausteins aufzuheben. Bei einem Wechsel der Kopierschutzeinstellungen oder beim Aufheben des Kopierschutzes bleibt das Passwort bis zum nächsten Übersetzungsvorgang erhalten.

Wenn Sie einen solchen Kopierschutz für einen Baustein einrichten, ist es wichtig, diesen Baustein auch mit einem Know-how-Schutz zu versehen. Ansonsten kann jeder den Kopierschutz zurücksetzen.

Hinweis

S7-1500 und S7-1200 V2.2 und höher: Wenn Sie einen Baustein mit Kopierschutz in ein Gerät laden, das mit der festgelegten Seriennummer nicht übereinstimmt, wird der gesamte Ladevorgang zurückgewiesen. Das bedeutet, dass auch Bausteine ohne Kopierschutz nicht geladen werden.

Kopierschutz einrichten

Um den Kopierschutz für einen Baustein einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Kopierschutz einrichten möchten.
2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Wählen Sie im Bereich "Kopierschutz" aus der Klappliste entweder den Eintrag "An Seriennummer der CPU binden" oder den Eintrag "An Seriennummer der Memory Card binden".
4. Geben Sie entweder die Seriennummer direkt ein oder aktivieren Sie die Option "Seriennummer eingefügt beim Laden in ein Gerät oder eine Memory Card", wenn die Seriennummer beim Ladevorgang automatisch eingefügt werden soll.
   Fahren Sie mit Schritt 7 fort, wenn Sie die Seriennummer direkt eingegeben haben. Wenn die Seriennummer beim Ladevorgang automatisch eingefügt werden soll, müssen Sie zusätzlich die Schritte unter 5 und 6 ausführen.
5. Klicken Sie auf "Passwort definieren".
   Der Dialog "Schutz definieren" wird geöffnet.
6. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in die Felder "Neues Passwort" und "Passwort bestätigen" das Passwort ein.
7. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das Passwort aus.

8. Bestätigen Sie die Eingaben mit "OK".
   Im Statusfeld erhalten Sie die Information, dass ein Kopierschutz-Passwort definiert ist.
9. Im Bereich "Know-how-Schutz" können Sie nun den Know-how-Schutz für den Baustein einrichten, falls der Baustein noch keinen Know-how-Schutz besitzt.

Kopierschutz entfernen

Um den Kopierschutz wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Kopierschutz entfernen möchten.

   Hinweis

   Wenn der Baustein einen Know-how-Schutz besitzt, müssen Sie den Baustein mit dem korrekten Passwort öffnen, um die Einstellungen zum Kopierschutz ändern zu können.

2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Wählen Sie im Bereich "Kopierschutz" aus der Klappliste den Eintrag "Keine Bindung".

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.7 Schreibschutz für Bausteine einrichten und entfernen

Sie können einen Schreibschutz für Bausteine des Typs OB, FB, FC einrichten, um ein unbeabsichtigtes Ändern zu verhindern. Bausteine mit einem Schreibschutz können nur lesend geöffnet werden, die Bausteineigenschaften können Sie jedoch weiterhin bearbeiten.

Beachten Sie, dass der Schreibschutz kein Know-how-Schutz ist, da Sie den Schreibschutz jederzeit wieder entfernen können. Wenn ein Baustein schreibgeschützt ist, können Sie nicht zusätzlich den Know-how-Schutz einrichten. Entfernen Sie den Schreibschutz für den Baustein, wenn Sie ihn know-how-schützen möchten.

Schreibschutz einrichten

Um den Schreibschutz für einen Baustein einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Schreibschutz einrichten möchten.
2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Klicken Sie im Bereich "Schreibschutz" auf "Passwort definieren".
   Der Dialog "Schutz definieren" wird geöffnet.
4. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in die Felder "Neues Passwort" und "Passwort bestätigen" das Passwort ein.
5. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das Passwort aus.
6. Bestätigen Sie die Eingaben mit "OK".
   Im Statusfeld erhalten Sie die Information, dass ein Schreibschutz-Passwort definiert ist. Der Text der Schaltfläche ändert sich zu "Passwort ändern".
7. Klicken Sie auf das Optionskästchen "Schreibschutz".
   Der Dialog "Zugriffsschutz" wird geöffnet.
8. Geben Sie das korrekte Passwort ein oder wählen Sie das korrekte Passwort aus, falls Sie einen Passwort-Provider verwenden.
   Der Schreibschutz wird aktiviert und ab dem nächsten Öffnen des Bausteins aktiv.

Schreibschutz deaktivieren

Um den Schreibschutz zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Schreibschutz deaktivieren möchten.
2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Klicken Sie im Bereich "Schreibschutz" auf das Optionskästchen "Schreibschutz".
   Der Dialog "Zugriffsschutz" wird geöffnet.
4. Geben Sie das korrekte Passwort ein oder wählen Sie das korrekte Passwort aus, falls Sie einen Passwort-Provider verwenden
   Der Schreibschutz für den Baustein wird deaktiviert und der Baustein kann ab dem nächsten Öffnen wieder bearbeitet werden. Das Passwort bleibt jedoch erhalten, sodass Sie den Schreibschutz jederzeit wieder aktivieren können.

Schreibschutz aktivieren

Um den Schreibschutz wieder zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Schreibschutz wieder aktivieren möchten.
2. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Schutz".
3. Klicken Sie im Bereich "Schreibschutz" auf das Optionskästchen "Schreibschutz".
   Der Dialog "Zugriffsschutz" wird geöffnet.
4. Geben Sie das korrekte Passwort ein oder wählen Sie das korrekte Passwort aus, falls Sie einen Passwort-Provider verwenden.
   Der Schreibschutz für den Baustein wird erneut gesetzt und ist ab dem nächsten Öffnen des Bausteins aktiv.

Schreibschutz-Passwort ändern

Um das Schreibschutz-Passwort zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie das Schreibschutz-Passwort ändern möchten.
2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Klicken Sie auf die Schaltfläche "Passwort ändern".
   Der Dialog "Schutz ändern" wird geöffnet.
4. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in das Feld "Altes Passwort" das alte Passwort und in das Feld "Neues Passwort" das neue Passwort ein. Geben Sie das neue Passwort zusätzlich in das Feld "Passwort bestätigen" ein.
5. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das neue Passwort aus.
6. Bestätigen Sie die Eingaben mit "OK".
   Das Schreibschutz-Passwort für den Baustein wird geändert.

Hinweis

Das Ändern des Schreibschutz-Passworts für den Baustein ist immer möglich. Dies ist unabhängig davon, ob der Schreibschutz aktiv ist oder nicht.

Schreibschutz entfernen

Um den Schreibschutz für einen Baustein zu entfernen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des Bausteins, für den Sie den Schreibschutz entfernen möchten.
2. Wählen Sie in den Bausteineigenschaften in der Bereichsnavigation den Eintrag "Schutz".
3. Klicken Sie auf die Schaltfläche "Passwort ändern".
   Der Dialog "Schutz ändern" wird geöffnet.

4. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in das Feld "Altes Passwort" das aktuelle Passwort ein.
5. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das aktuelle Passwort aus.
6. Klicken Sie auf "Entfernen".
   Der Schreibschutz für den Baustein wird entfernt. Im Gegensatz zum Deaktivieren des Schreibschutzes wird auch das Passwort gelöscht.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.8 Know-how-Schutz für Bausteine einrichten

Sie können in Ihrem Projekt für die Bausteine in den Geräten einen Know-how-Schutz einrichten.

Vorgehen

Um einen Know-how-Schutz für Bausteine einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Bausteine ohne Know-how-Schutz, die Sie schützen möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Schutz definieren" wird geöffnet.
3. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in den Feldern "Neues Passwort" und "Passwort bestätigen" das Passwort ein.
4. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das Passwort aus.
5. Bestätigen Sie die Eingaben mit "OK".
6. Schließen Sie den Dialog "Schutz definieren" mit "OK".

Ergebnis

Die ausgewählten Bausteine werden mit einem Know-how-Schutz versehen. In der Projektnavigation werden know-how-geschützte Bausteine mit einem Schloss markiert. Das eingegebene Passwort ist für alle ausgewählten Bausteine gültig.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.9 Know-how-geschützte Bausteine öffnen

Wenn als Art der Passwortvergabe die Verwendung eines Passwort-Providers eingestellt ist, verbindet sich das TIA Portal beim Öffnen eines geschützten Bausteins mit dem Passwort-Provider. Bei erfolgreicher Verbindung ist eine Passworteingabe nicht notwendig. Ist der Passwort-Provider nicht verfügbar, können Sie den Baustein nur dann vollständig öffnen, wenn Sie die manuelle Eingabe als Rückfallstrategie festgelegt hatten und das entsprechende Passwort kennen. Der Name des Passworts reicht hier nicht aus.

Sie können mehrere know-how-geschützte Bausteine nur dann gemeinsam öffnen, wenn sie mit dem gleichen Passwort geschützt werden.

Voraussetzung

- Als Art der Passwortvergabe ist "Manuelle Passworteingabe" eingestellt.
- Als Art der Passwortvergabe ist "Auswahl über Passwort-Provider mit manueller Eingabe als Rückfallstrategie für den Zugriff" eingestellt und es besteht keine Verbindung zum Passwort-Provider.

Vorgehen

Um einen know-how-geschützten Baustein zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie doppelt auf den Baustein, den Sie öffnen möchten.
   Der Dialog "Zugriffsschutz" wird geöffnet.
2. Geben Sie das Passwort für den know-how-geschützten Baustein ein.
3. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn Sie das korrekte Passwort eingegeben haben, wird der know-how-geschützte Baustein geöffnet. Der Know-how-Schutz des Bausteins wird jedoch nicht aufgehoben. Wenn Sie den Baustein z. B. kopieren oder in eine Bibliothek einfügen, sind auch die Kopien know-how-geschützt.

Nach dem Öffnen des Bausteins können Sie den Programmcode und die Bausteinschnittstelle des Bausteins so lange bearbeiten, bis Sie den Baustein oder das TIA Portal schließen. Beim nächsten Öffnen des Bausteins muss das Passwort wieder eingegeben werden. Wenn Sie den Dialog "Zugriffsschutz" mit "Abbrechen" schließen, wird der Baustein zwar geöffnet, aber der Code des Bausteins wird nicht angezeigt und Sie können den Baustein nicht bearbeiten.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)
Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)
Know-how-Schutz für Bausteine einrichten (Seite 7255)
Know-how-geschützte Bausteine drucken (Seite 7258)
Passwort ändern (Seite 7260)
Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.10 Know-how-geschützte Bausteine drucken

Sie können know-how-geschützte Bausteine nur dann vollständig ausdrucken, wenn der Baustein mit dem korrekten Passwort geöffnet wurde. Drucken Sie einen geschlossenen Baustein oder wurde der Baustein nicht mit dem korrekten Passwort geöffnet, werden nur die nicht geschützten Daten des Bausteins gedruckt.

Vorgehen

Um einen know-how-geschützten Baustein vollständig zu drucken, gehen Sie folgendermaßen vor:

1. Öffnen Sie den know-how-geschützten Baustein, den Sie drucken möchten.
   Siehe auch: Know-how-geschützte Bausteine öffnen (Seite 7256)
2. Wählen Sie im Menü "Projekt" den Befehl "Drucken".
   Der Dialog "Drucken" wird geöffnet.
3. Wählen Sie im Feld "Name" den Drucker aus.
4. Klicken Sie auf "Erweitert", um die Windows-Einstellungen des Druckers zu bearbeiten.
5. Wählen Sie in der Klappliste "Dokument-Information" das Dokumentations-Informations-Set aus, das für das Rahmen-Layout verwendet werden soll.
6. Wählen Sie unter "Objekte/Bereich drucken aus", ob Sie alle Objekt bzw. den gesamten Bereich oder nur eine Auswahl drucken möchten.
7. Wählen Sie unter "Eigenschaften" den Druckumfang aus.
   - Wählen Sie "Alle", um den kompletten Baustein zu drucken.
   - Wählen Sie "Sichtbar", um alle auf dem Bildschirm sichtbaren Informationen innerhalb des Bausteins zu drucken.
   - Wählen Sie "Kompakt", um den Baustein in einer gekürzten Form zu drucken.
8. Klicken Sie auf "Vorschau", um vorab eine Druckvorschau zu erzeugen.
   Im Arbeitsbereich wird eine Druckvorschau erstellt.
9. Klicken Sie auf "Drucken", um den Ausdruck zu starten.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Passwort ändern (Seite 7260)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.11 Passwort ändern

Wenn Sie einen Passwort-Provider verwenden, ist es von den Vorgaben des Passwort-Providers abhängig, ob Sie das Passwort für know-how-geschützte Bausteine ändern können.

Vorgehen

Um das Passwort zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die know-how-geschützten Bausteine, für die Sie das Passwort ändern möchten.

   Hinweis

   Wenn Sie für mehrere Bausteine das Passwort gleichzeitig ändern möchten, müssen die selektierten Bausteine das gleiche Passwort haben.

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Schutz ändern" wird geöffnet.
3. Wenn Sie die manuelle Passworteingabe nutzen, tragen Sie in das Feld "Altes Passwort" das alte Passwort und in das Feld "Neues Passwort" das neue Passwort ein. Geben Sie das neue Passwort zusätzlich in das Feld "Passwort bestätigen" ein.
4. Wenn Sie einen Passwort-Provider nutzen, wählen Sie zunächst den Passwort-Provider und anschließend das neue Passwort aus.
5. Schließen Sie den Dialog "Schutz ändern" mit "OK".

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Know-how-Schutz für Bausteine entfernen (Seite 7261)

8.12 Know-how-Schutz für Bausteine entfernen

Wenn Sie einen Passwort-Provider verwenden, ist es von den Vorgaben des Passwort-Providers abhängig, ob Sie den Know-how-Schutz eines Bausteins entfernen können.

Vorgehen

Um den Know-how-Schutz für Bausteine zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Bausteine, bei denen Sie den Know-how-Schutz entfernen möchten.

   Hinweis

   Wenn Sie für mehrere Bausteine den Know-how-Schutz gleichzeitig entfernen möchten, müssen die selektierten Bausteine das gleiche Passwort haben.

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Schutz ändern" wird geöffnet.
3. Wenn Sie die manuelle Passworteingabe nutzen, geben Sie in das Feld "Altes Passwort" das Passwort für den Know-how-Schutz des Bausteins ein. Wenn Sie einen Passwort-Provider verwenden, ist dies nicht notwendig.
4. Klicken Sie auf die Schaltfläche "Entfernen".

Ergebnis

Der Know-how-Schutz wird für die ausgewählten Bausteine aufgehoben.

Siehe auch

Schützen von Bausteinen (Seite 7239)

Passwort-Provider anbinden (Seite 7242)

Verfügbare Passwort-Provider anzeigen (Seite 7245)

Aktivitäten der Passwort-Provider aufzeichnen (Seite 7246)

Art der Passwortvergabe festlegen (Seite 7248)

Kopierschutz für Bausteine einrichten und entfernen (Seite 7250)

Schreibschutz für Bausteine einrichten und entfernen (Seite 7252)

Know-how-Schutz für Bausteine einrichten (Seite 7255)

Know-how-geschützte Bausteine öffnen (Seite 7256)

Know-how-geschützte Bausteine drucken (Seite 7258)

Passwort ändern (Seite 7260)

Bausteinschnittstelle deklarieren 9

9.1 Übersicht über die Bausteinschnittstelle

Einführung

Die Schnittstelle enthält die Deklarationen lokaler Variablen und Konstanten, die innerhalb des Bausteins verwendet werden. Die Variablen unterteilen sich in zwei Gruppen:

- Bausteinparameter, die die Schnittstelle des Bausteins für den Aufruf im Programm bilden.
- Lokaldaten, die zum Speichern von Zwischenergebnissen dienen.

Durch die Variablendeklaration bestimmen Sie die Aufrufschnittstelle eines Bausteins im Programm sowie die Namen und Datentypen der Variablen und Konstanten, die Sie im Baustein verwenden möchten.

Die Schnittstelle von Funktionsbausteinen definiert darüber hinaus auch die Struktur der Instanzen, die dem Funktionsbaustein zugeordnet werden.

Tabellarische oder textuelle Darstellung der Bausteinschnittstelle

Standardmäßig wird die Schnittstelle in der tabellarischen Ansicht angezeigt. In SCL kann die Schnittstelle auch in einer textuellen Sicht dargestellt werden.

Siehe auch:

Tabellarische oder textuelle Darstellung der Bausteinschnittstelle wählen (Seite 7271)

Bausteinparameter

Die folgende Tabelle zeigt die Typen von Bausteinparametern:

Typ	Abschnitt	Funktion	Verfügbar in
Eingangsparameter	Input	Parameter, deren Werte der Baustein liest.	Funktionen, Funktionsbausteinen und einigen Arten von Organisationsbausteinen
Ausgangsparameter	Output	Parameter, deren Werte der Baustein schreibt.	Funktionen und Funktionsbausteinen
Durchgangsparameter	InOut	Parameter, deren Werte der Baustein beim Aufruf liest und nach der Bearbeitung wieder in denselben Parameter schreibt.	Funktionen und Funktionsbausteinen
Rückgabewert	Return	Wert, der an den aufrufenden Baustein zurückgeliefert wird.	Funktionen

Je nach Art des geöffneten Bausteins können weitere Abschnitte angezeigt werden.

Lokaldaten

Die folgende Tabelle zeigt die Typen von Lokaldaten:

Typ	Abschnitt	Funktion	Verfügbar in
Temporäre Lokaldaten	Temp	Variablen, die zum Speichern von temporären Zwischenergebnissen dienen. Temporäre Daten bleiben nur für einen Zyklus erhalten. Wenn Sie temporäre Lokaldaten verwenden, müssen Sie sicherstellen, dass die Werte innerhalb des Zyklus geschrieben werden, in dem Sie sie lesen wollen. Ansonsten sind die Werte zufällig.	Funktionen, Funktionsbausteinen und Organisationsbausteinen **Hinweis**: In Instanzdatenbausteinen werden temporäre Lokaldaten nicht angezeigt.
Statische Lokaldaten	Static	Variablen, die zum Speichern von statischen Zwischenergebnissen im Instanz-Datenbaustein dienen. Statische Daten bleiben so lange erhalten, bis sie neu geschrieben werden, auch über mehrere Zyklen hinweg. Auch die Namen der Bausteine, die in einem Codebaustein als Multiinstanz aufgerufen werden, werden in den statischen Lokaldaten abgelegt.	Funktionsbausteinen
Konstante	Constant	Konstanten mit deklariertem symbolischem Namen, die innerhalb des Bausteins verwendet werden.	Funktionen, Funktionsbausteinen und Organisationsbausteinen **Hinweis**: In Instanzdatenbausteinen werden lokale Konstanten nicht angezeigt.

Siehe auch

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu Konstanten (Seite 102)

Parameterübergabe beim Bausteinaufruf (Seite 78)

Einstellwerte während der Inbetriebnahme nutzen (Seite 7365)

9.2 Regeln zur Deklaration der Bausteinschnittstelle

9.2.1 Allgemeine Regeln zur Deklaration der Bausteinschnittstelle

Verwenden von Bausteinparametern

Für die Verwendung der Bausteinparameter innerhalb des Bausteins gelten die folgenden Regeln:

- Eingangsparameter dürfen nur gelesen werden.
- Ausgangsparameter dürfen nur geschrieben werden.
- Durchgangsparameter dürfen gelesen und geschrieben werden.

Hinweis

Maximale Anzahl an Parametern pro Baustein

Die maximal zulässige Anzahl an Formalparametern ist für alle gängigen Anwendungsszenarien ausreichend. Die genaue Anzahl hängt u. a. von den gewählten Datentypen und Deklarationsabschnitten ab.

Bei Überschreitung der Maximalmenge erhalten Sie während des Übersetzungslaufs eine Meldung. In diesem Fall können Sie mehrere Parameter in einem PLC-Datentyp (UDT) oder in einem globalen Datenbaustein (DB) zusammenfassen und diesen als Bausteinparameter übergeben.

Zuordnen von Defaultwerten zu Bausteinparametern

Sie können bestimmten Parametern in der Schnittstelle von Funktionsbausteinen Defaultwerte zuordnen. Die Möglichkeit der Zuordnung ist abhängig vom Deklarationsabschnitt und Datentyp des Parameters. Wenn Sie keinen Defaultwert eintragen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert.

Die folgende Tabelle zeigt, welchen Parametern ein Defaultwert zugeordnet werden kann:

Parameterart	Abschnitt	Zuordung eines Defaultwerts möglich		
		Elementare Datentypen	Strukturierte Datentypen	Parametertypen
Eingangsparameter	Input	X	X	-
Ausgangsparameter	Output	X	X	-
Durchgangsparameter	InOut	X	- (1)	-
Statische Lokaldaten	Static	X	X	-
Temporäre Lokaldaten	Temp	-	-	-
Konstanten	Constant	X	-	-

(1) Ausnahme: In Bausteinen mit optimiertem Zugriff haben Sie unter bestimmten Bedingungen die Möglichkeit, PLC-Datentypen als Defaultwerte zu verwenden.

Hinweis

Weitere Informationen zu zulässigen Zeichen in Variablennamen finden Sie im Siemens Industry Online Support in folgendem Beitrag:

Wann müssen Bezeichner bzw. Operanden in STEP 7 (TIA Portal) in "Anführungszeichen" verwendet werden?

https://support.industry.siemens.com/cs/ww/de/view/10947785 (https://support.industry.siemens.com/cs/ww/de/view/109477857)

Empfehlungen für eine projektweit einheitliche Benennung von Variablen finden Sie im Programmierstyleguide:

https://support.industry.siemens.com/cs/ww/de/view/109478084 (https://support.industry.siemens.com/cs/ww/de/view/81318674)

Siehe auch

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Vordefinierte Aktualparameter deklarieren (Seite 7285)

9.2.2 Gültige Datentypen in der Bausteinschnittstelle

Gültige Datentypen in der Bausteinschnittstelle in S7-300/400

Die folgende Tabelle zeigt, welche Datentypen Sie den Parametern in den einzelnen Abschnitten der Schnittstelle zuweisen können.

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING DT	Parameterty-pen	VOID	POINTER	ANY
Organisationsbaustein						
Temp	X	X	-	-	-	X
Constant	X	X (3)	-	-	-	-
Funktionsbaustein						
Input	X	X	X	-	X	X
Output	X	X	-	-	-	-
InOut	X	X (1)	-	-	X	X
Static	X	X	-	-	-	-
Temp	X	X	-	-	-	X
Constant	X	X (3)	-	-	-	-
Funktion						

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING DT	Parameterty-pen	VOID	POINTER	ANY
Input	X	X (1)	X	-	X	X
Output	X	X (1)	-	-	X	X
InOut	X	X (1)	-	-	X	X
Temp	X	X	-	-	-	X
Return	X	X	-	X	X	X (2)
Constant	X	X (3)	-	-	-	-

(1) STRING kann nur in der Standardlänge von 254 Zeichen definiert werden.
(2) In SCL ist ANY nicht als Funktionswert zulässig.
(3) Konstanten mit den Datentypen ARRAY oder STRUCT sind nicht zulässig.

Siehe auch

Parameterübergabe beim Bausteinaufruf (Seite 78)

9.2.3 Gültige Datentypen in der Bausteinschnittstelle

Gültige Datentypen in der Bausteinschnittstelle in S7-1200

Die folgende Tabelle zeigt, welche Datentypen Sie in den Parametern in den einzelnen Abschnitten der Schnittstelle zuweisen können.

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING / WSTRING DT	ARRAY [*]	VOID	VARIANT
Organisationsbaustein					
Temp	X	X	-	-	X
Constant	X	X (1) (2)	-	-	-
Funktionsbaustein					
Input	X	X	-	-	X
Output	X	X	-	-	-
InOut	X	X (1)	X (4)	-	X
Static	X	X	-	-	-
Temp	X	X	-	-	X
Constant	X	X (1) (2)	-	-	-
Funktion					
Input	X	X (1)	X (4)	-	X
Output	X	X (1)	X (4)	-	X

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING / WSTRING DT	ARRAY [*]	VOID	VARIANT
InOut	X	X (1)	X (4)	-	X
Temp	X	X	-	-	X
Return	X	X (3)	-	X	-
Constant	X	X (1) (2)	-	-	-

(1) In diesen Abschnitten können Sie für STRING und WSTRING keine Längendeklarationen angeben. STRINGs haben hier immer die Standardlänge 254, WSTRINGs die Standardlänge 16832. Eine Deklaration im Format MyString[3] wäre nicht zulässig. WSTRING ist in diesen Abschnitten nur in Bausteinen mit optimiertem Zugriff zulässig.

(2) Konstanten mit den Datentypen ARRAY oder STRUCT sind nicht zulässig.

(3) Funktionswerte vom Datentyp WSTRING dürfen nicht länger als 1022 Zeichen sein.

(4) ARRAY[*] ist ab Firmware V4.2 in Bausteinen mit optimiertem Zugriff verfügbar.

Gültige Datentypen in der Bausteinschnittstelle in S7-1500

Die folgende Tabelle zeigt, welche Datentypen Sie in den Parametern in den einzelnen Abschnitten der Schnittstelle zuweisen können.

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING / WSTRING DT	ARRAY[*]	Parameterty-pen	VOID	DB_ANY	POINTER	ANY	VARIANT	REF_TO
Organisationsbaustein										
Temp	X	X	-	- (4)	-	X	-	X (3)	X	X
Constant	X	X (1) (5)	-	-	-	-	-	-	-	-
Funktionsbaustein										
Input	X	X	-	X	-	X	X	X	X	-
Output	X	X	-	-	-	X	-	-	-	-
InOut	X	X (1)	X(7)(8)	- (4)	-	X	X	X	X	-
Static	X	X	-	-	-	X	-	-	-	-
Temp	X	X	-	- (4)	-	X	-	X (3)	X	X
Constant	X	X (1) (5)	-	-	-	-	-	-	-	-
Funktion										
Input	X	X (1)	X(7)	X	-	X	X	X	X	X
Output	X	X (1)	X(7)	-	-	X	X	X	X	X
InOut	X	X (1)	X(7)	- (4)	-	X	X	X	X	-
Temp	X	X	-	- (4)	-	X	-	X (3)	X	X
Return	X	X (6)	-	-	X	X	X	x (2)	-	X

Abschnitt	Standard-Datentypen	ARRAY STRUCT STRING / WSTRING DT	ARRAY[*]	Parametertypen	VOID	DB_ANY	POINTER	ANY	VARIANT	REF_TO
Constant	X	X (1) (5)	-	-	-	-	-	-	-	-
(1) In diesen Abschnitten können Sie für STRING und WSTRING keine Längendeklarationen angeben. STRINGs haben hier immer die Standardlänge 254, WSTRINGs die Standardlänge 16832. Eine Deklaration im Format MyString[3] wäre nicht zulässig. WSTRING ist in diesen Abschnitten nur in Bausteinen mit optimiertem Zugriff zulässig. (2) In SCL ist ANY nicht als Funktionswert zulässig. (3) ANY kann nur in Bausteinen mit Standardzugriff im Abschnitt "Temp" verwendet werden. (4) Der Parametertyp "INSTANCE" ist als einzige Ausnahme in den Abschnitten "TEMP" und "InOut" zulässig. (5) Konstanten mit den Datentypen ARRAY oder STRUCT sind nicht zulässig. (6) Funktionswerte vom Datentyp WSTRING dürfen nicht länger als 1022 Zeichen sein. (7) ARRAY[*] ist ab Firmware V2.0 in Bausteinen mit optimiertem Zugriff verfügbar. (8) ARRAY[*] ist nur verfügbar, wenn das Attribut "Im IDB setzen" nicht gesetzt ist.										

Siehe auch

Parameterübergabe beim Bausteinaufruf (Seite 78)

9.2.4 Remanenz von lokalen Variablen

Einführung

Funktionsbausteine speichern ihre Daten in einer Instanz. Um den Verlust dieser Daten bei Spannungsausfall zu vermeiden, können Sie Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Die Möglichkeiten zur Einstellung der Remanenz sind abhängig von der eingestellten Zugriffsart des Funktionsbausteins.

Remanenzverhalten in Bausteinen mit Standardzugriff

In Bausteinen mit Standardzugriff können Sie das Remanenzverhalten einzelner Variablen nicht einstellen. Sie können sie lediglich in der zugeordneten Instanz als remanent definieren. Alle darin enthaltenen Variablen gelten dann als remanent.

Remanenz bei optimiertem Bausteinzugriff

In Datenbausteinen mit optimiertem Zugriff können Sie das Remanenzverhalten einzelner Variablen bestimmen.

Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb der Struktur können Sie keine individuelle Remanenzeinstellung vornehmen.

Im Abschnitt "InOut" können Sie keine remanenten Variablen strukturierten Datentyps anlegen. Durchgangsparameter mit strukturiertem Datentyp, z. B. ARRAY, STRUCT, oder STRING, sind immer nicht-remanent.

Folgende Einstellungen stehen zur Verfügung:

- Remanent
  Die Werte der Variablen oder der Struktur stehen auch nach Spannungsausfall zur Verfügung.
- Nicht remanent
  Die Werte der Variablen oder der Struktur gehen bei Spannungsausfall verloren.
- Im IDB setzen
  Die Remanenz kann im Instanz-Datenbaustein eingestellt werden. Die Einstellung, die im Instanz-Datenbaustein vorgenommen wird, gilt dann jedoch zentral für alle Variablen, die mit "Im IDB setzen" markiert sind.

Siehe auch

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Grundlagen zum Bausteinzugriff (Seite 55)

9.3 Tabellarische oder textuelle Darstellung der Bausteinschnittstelle wählen

Tabellarische oder textuelle Darstellung der Bausteinschnittstelle

Standardmäßig wird die Schnittstelle in der tabellarischen Ansicht angezeigt. In SCL kann die Schnittstelle auch in einer textuellen Sicht dargestellt werden.

Die Darstellung der Schnittstelle eines SCL-Bausteins können Sie zentral einstellen. Wenn Sie als Voreinstellung die textuelle Darstellung wählen, wird die Schnittstelle aller künftig neu erzeugten Bausteine in der textuellen Sicht dargestellt.

Folgende Elemente können nur in der textuellen Ansicht verwendet werden:

- Kommentare zwischen den Abschnitten oder am Beginn eines Abschnitts
- Mehrere Abschnitte eines Typs (z. B. VAR_INPUT)

Hinweis

Fehlersichere Bausteine und Systembausteine

Bei fehlersicheren Bausteinen und Systembausteinen wird die Schnittstelle ausschließlich in der tabellarischen Darstellung angezeigt.

Darstellung der Schnittstelle für neue Bausteine einstellen

Um die Darstellung der Schnittstelle für neue Bausteine einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > SCL".
3. Wählen Sie unter "Schnittstelle > Voreinstellung für neue Bausteine" die gewünschte Ansicht der Schnittstelle aus.

Wenn Sie neue Bausteine anlegen, wird deren Schnittstelle in der gewählten Ansicht dargestellt.

Hinweis

Darstellung der Schnittstelle nach dem Import eines Bausteins

Nachdem dem Import eines Bausteins wird seine Schnittstelle in der tabellarischen Ansicht dargestellt.

Siehe auch

Aufbau der Bausteinschnittstelle (Seite 7272)

Aufbau der textuellen Bausteinschnittstelle (Seite 7290)

9.4 Bausteinschnittstelle deklarieren

9.4.1 Aufbau der Bausteinschnittstelle

Aufbau der Bausteinschnittstelle

Das folgende Bild zeigt den Aufbau der Bausteinschnittstelle. Die Anzahl der Spalten und Abschnitte variiert je nach Bausteinart.

	Name	Datentyp	Defaultwert	Remanenz
	Input			
	My_Input1	Bool	false	Nicht remanent
	Output			
	My_Output1	Bool	1	Remanent
	InOut			
	<Hinzufügen>			
	Static			
	Temp			
	Constant			
	My_Constant1	Int	22	

Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie und der Art des geöffneten Objekts.

Spalte	Erläuterung
	Symbol, das Sie anklicken können, um ein Element per Drag & Drop in ein Programm zu ziehen und dort als Operand zu verwenden.
Name	Name des Elements.
Datentyp	Datentyp des Elements.
Offset	Relative Adresse einer Variablen innerhalb des Bausteins. Diese Spalte ist nur in Bausteinen mit Standardzugriff sichtbar. Hinweis: Viele Anweisungen aus den SIMATIC-Systembibliotheken haben die Eigenschaft "optimierter Bausteinzugriff" und belegen somit keine festen Speicheradressen. Für diese Anweisungen wird kein Offset angezeigt, auch wenn Sie sie als Multiinstanz in einem Baustein mit Standardzugriff verwenden.

Spalte	Erläuterung
Defaultwert	Wert, mit dem Sie bestimmte Variablen in der Schnittstelle des Codebausteins vorbelegen können, oder Wert einer lokalen Konstante. Die Angabe des Defaultwerts ist für Variablen optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Der Defaultwert einer Variablen wird im zugehörigen Instanz-Datenbaustein als Startwert übernommen. Diese übernommenen Werte können Sie im Instanz-Datenbaustein mit instanzspezifischen Startwerten ersetzen. Konstanten haben immer den in der Bausteinschnittstelle deklarierten Defaultwert. Sie werden in Instanz-Datenbausteinen nicht angezeigt und können dort nicht mit instanzspezifischen Werten belegt werden.
Remanenz	Kennzeichnet eine Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten. Diese Spalte ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff sichtbar.
Sichtbar in HMI Engineering	Zeigt an, ob eine Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.
Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
Einstellwert	Kennzeichnet eine Variable als Einstellwert. Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Die Spalte ist nur in der Schnittstelle von Funktionsbausteinen vorhanden.
Überwachung	Zeigt an, ob für die Variable eine Überwachung zur Prozessdiagnose angelegt wurde.
Kommentar	Kommentar zur Dokumentation des Elements.

Siehe auch

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Gültige Datentypen in der Bausteinschnittstelle (Seite 7267)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu Konstanten (Seite 102)

Tabellarische oder textuelle Darstellung der Bausteinschnittstelle wählen (Seite 7271)

Parameterübergabe beim Bausteinaufruf (Seite 78)

Einstellwerte während der Inbetriebnahme nutzen (Seite 7365)

9.4.2 Lokale Variablen und Konstanten in der Bausteinschnittstelle deklarieren

Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

Vorgehen

Um eine Variable oder Konstante zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstelle den gewünschten Deklarationsabschnitt aus. Eine Konstante müssen Sie im Abschnitt "Constant" deklarieren.
2. Tragen Sie in der Spalte "Name" einen Namen für das Element ein.
3. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
4. Tragen Sie für Konstanten einen Wert in der Spalte "Defaultwert" ein.
5. Optional: Ändern Sie die Eigenschaften, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.

Ergebnis

Das Element wird angelegt.

Syntaxprüfung

Nach jeder Eingabe wird eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält, lässt sich das Programm allerdings nicht übersetzen.

Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, werden die Aufrufe des Bausteins im Programm möglicherweise inkonsistent. Die Aufrufstellen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, müssen Sie die inkonsistenten Bausteinaufrufe manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Siehe auch

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)
Variablen (Seite 100)
Konstanten (Seite 102)
Variablen vom Datentyp ARRAY deklarieren (Seite 7277)
Variablen vom Datentyp STRUCT deklarieren (Seite 7279)
Variablen vom Datentyp STRING und WSTRING deklarieren (Seite 7280)
Variablen basierend auf einem PLC-Datentyp deklarieren (Seite 7281)
Überlagerte Variablen deklarieren (Seite 7281)
Multiinstanzen deklarieren (Seite 7282)
Parameterinstanzen deklarieren (Seite 7283)
Grundlagen zu Startwerten (Seite 7336)
Remanenz von lokalen Variablen (Seite 7269)
Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

9.4.3 Lokale Variable im Programmiereditor deklarieren

Voraussetzung

Der Programmiereditor ist geöffnet.

Vorgehen

Um eine lokale Variable zu deklarieren, gehen Sie folgendermaßen vor:

1. Fügen Sie eine Anweisung in Ihr Programm ein.
   Die Zeichenfolgen "<???>", "<??.?>" oder "..." stellen Operandenplatzhalter dar.
2. Ersetzen Sie einen Operandenplatzhalter mit dem Namen der zu erstellenden Variablen.
3. Selektieren Sie den Namen des Elements.
   Wenn Sie mehrere Elemente deklarieren wollen, selektieren Sie die Namen aller zu deklarierenden Elemente.
4. Wählen Sie im Kontextmenü den Befehl "Variable definieren".
   Der Dialog "Variable definieren" wird geöffnet. Er zeigt eine Deklarationstabelle, in die bereits der Name des Elements eingetragen ist.

5. Um eine lokale Variable zu deklarieren, wählen Sie einen der folgenden Abschnitte aus:
   - Local In
   - Local Out
   - Local InOut
   - Local Static
   - Local Temp
6. Geben Sie in den weiteren Spalten Datentyp und Kommentar ein.
7. Klicken Sie auf die Schaltfläche "Definieren", um die Eingabe abzuschließen.

Ergebnis

Die Deklaration wird direkt in die Bausteinschnittstelle geschrieben und ist innerhalb des gesamten Bausteins gültig.

Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, werden die Aufrufe des Bausteins im Programm möglicherweise inkonsistent. Die Aufrufstellen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, müssen Sie die inkonsistenten Bausteinaufrufe manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Siehe auch

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Grundlagen zu Startwerten (Seite 7336)

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Remanenz von lokalen Variablen (Seite 7269)

Tabelle der Bausteinschnittstelle bearbeiten (Seite 7287)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

9.4.4 Variablen vom Datentyp ARRAY deklarieren

Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Klicken Sie in der Spalte "Datentyp" auf die Schaltfläche für die Datentypauswahl.
   Eine Liste der zulässigen Datentypen wird geöffnet.
4. Wählen Sie den Datentyp "Array" aus.
   Der Dialog "Array" wird geöffnet.
5. Geben Sie im Eingabefeld "Datentyp" den Datentyp der ARRAY-Elemente an.
6. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.
   Beispiel für ein eindimensionales ARRAY:
   `Array[0..3] of Bool`
   Beispiel für ein dreidimensionales ARRAY:
   `Array[0..3, 0..15, 0..33]of Bool`
   Beispiel für ein eindimensionales ARRAY mit variablen Grenzen:
   `Array[*]of Bool`
   Beispiel für ein dreidimensionales ARRAY mit variablen Grenzen:
   `Array[*, *, *]of Bool`
   Beispiel für ein eindimensionales ARRAY mit lokalen Konstanten als ARRAY-Grenzen
   `Array[#My_local_const1..#My_local_const2]of Bool`
   Beispiel für ein eindimensionales ARRAY mit globalen Konstanten als ARRAY-Grenzen
   `Array["My_global_const1".."My_global_const1"]of Bool`
7. Bestätigen Sie die Eingabe.
8. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.

Die Variable wird angelegt, bleibt jedoch zugeklappt. Um das ARRAY aufzuklappen, klicken Sie auf das Dreieck vor der Variablen. Beachten Sie, dass ARRAYs mit mehr als 10000 Elementen aus Gründen der Übersichtlichkeit nicht aufklappbar sind.

Hinweis

Verfügbarkeit von ARRAY[*]

ARRAY[*] steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 in optimierten Bausteinen zur Verfügung. In Funktionen (FC) können Sie ARRAY[*] in allen Deklarationsabschnitten nutzen. In Funktionsbausteinen (FB) können Sie ARRAY[*] nur als Durchgangsparameter im Abschnitt "InOut" deklarieren.

Defaultwerte von ARRAY-Elementen eingeben

Um die einzelnen Elemente eines ARRAYs mit Defaultwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Defaultwert" ein.

ARRAYs kommentieren

Um einen Kommentar für das gesamte ARRAY anzugeben, tragen Sie diesen in die oberste Zeile der ARRAY-Deklaration ein. Der Kommentar wird als Voreinstellung für alle unterlagerten ARRAY-Elemente verwendet.

Um die einzelnen Elemente eines ARRAYs mit spezifischen Kommentaren zu versehen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Kommentar" ein.

ARRAYs im erweiterten Modus anzeigen

In der Anzeige "Erweiterter Modus" werden alle ARRAYs per Voreinstellung aufgeklappt dargestellt.

Um den erweiterten Modus einzuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Erweiterter Modus" in der Funktionsleiste.

Hinweis

Einschränkung in der Anzeige "Erweiterter Modus"

Abhängig der Größe des Arbeitsspeichers Ihres Computers gelten für den erweiterten Modus folgende Maximalgrenzen:

- Arbeitsspeicher < 8 GB: Es werden maximal 10.000 Zeilen angezeigt.
- Arbeitsspeicher >= 8 GB: Es werden maximal 40.000 Zeilen angezeigt.

Siehe auch

ARRAY (Seite 296)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Remanenz von lokalen Variablen (Seite 7269)

9.4.5 Variablen vom Datentyp STRUCT deklarieren

Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Tragen Sie Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
   Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
4. Fügen Sie in der leeren Zeile das erste Strukturelement ein.
   Nach dem Element wird erneut eine leere Zeile eingefügt.
5. Wählen Sie einen Datentyp für das Strukturelement aus.
6. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.
7. Wiederholen Sie Schritt 4 bis 6 für alle weiteren Strukturelemente.
   Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
8. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.

Hinweis

S7-1500: Maximal 252 Strukturen in einem Datenbaustein

In einem Datenbaustein sind maximal 252 Strukturen zulässig. Wenn Sie weitere Strukturen benötigen, müssen Sie Ihr Programm umstrukturieren. Beispielsweise können Sie die Strukturen in mehreren globalen Datenbausteinen anlegen.

Siehe auch

Deklaration von STRUCT (Seite 7294)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Remanenz von lokalen Variablen (Seite 7269)

9.4.6 Variablen vom Datentyp STRING und WSTRING deklarieren

Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp STRING oder WSTRING zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Tragen Sie in der Spalte "Datentyp" "STRING" oder "WSTRING" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
4. Optional: Geben Sie die Maximallänge der Zeichenkette mithilfe von eckigen Klammern nach dem Schlüsselwort STRING oder WSTRING an. Wenn Sie keine Maximallänge angeben, hat der STRING die Standardlänge von 254 Zeichen.
   Beispiel für einen WSTRING der Maximallänge 4:
   `WSTRING[4]`
   Beispiel für einen STRING, dessen Maximallänge durch eine lokale Konstante definiert wird:
   `STRING[#My_local_const1]`
   Beispiel für einen STRING, dessen Maximallänge durch eine globale Konstante definiert wird:
   `STRING["My_global_const1"]`

Ergebnis

Die Variable vom Datentyp STRING oder WSTRING wird angelegt.

Siehe auch

Deklaration von STRING und WSTRING (Seite 7300)

9.4.7 Variablen basierend auf einem PLC-Datentyp deklarieren

Voraussetzung

In der aktuellen CPU ist ein PLC-Datentyp deklariert.

Vorgehen

Um eine Variable basierend auf einem PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
   Die Variable wird angelegt.
3. Wenn Sie bei der Dekaration des PLC-Datentyps bereits Defaultwerte oder Kommentare für die Variablen innerhalb eines PLC-Datentyps definiert haben, werden diese grau angezeigt. Sie können diese Werte in der Bausteinschnittstelle ändern.
   Die geänderten Werte werden schwarz dargestellt und gelten nur für die spezifische Verwendungsstelle.

Hinweis

Wenn Sie PLC-Datentypen ändern oder löschen, die in der Bausteinschnittstelle verwendet werden, wird die Schnittstelle inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie das Programm neu übersetzen.

Siehe auch: Bausteinschnittstelle aktualisieren (Seite 7314)

9.4.8 Überlagerte Variablen deklarieren

Einführung

Um auf Datenbereiche innerhalb einer deklarierten Variablen zuzugreifen, können Sie die deklarierten Variablen mit einer weiteren Deklaration überlagern. Sie haben so die Möglichkeit, eine bereits deklarierte Variable mit einem anderen Datentyp anzusprechen. Sie können z. B. die einzelnen Bits einer Variablen vom Datentyp WORD mit einem ARRAY of BOOL ansprechen.

Variable überlagern

Um eine Variable mit einem neuen Datentyp zu überlagern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Wählen Sie in der Schnittstelle die Variable aus, die Sie mit einem neuen Datentyp überlagern möchten.

3. Klicken Sie in der Funktionsleiste auf "Zeile hinzufügen".
   Nach der zu überlagernden Variable wird eine Zeile hinzugefügt. Die überlagernde Variable muss in der Zeile direkt nach der überlagerten Variable deklariert werden.
4. Geben Sie in der Spalte "Name" einen Variablennamen ein.
5. Tragen Sie in der Spalte "Datentyp" den Eintrag "AT" ein. Sie werden dabei von der Autovervollständigung unterstützt.
   Der Eintrag in der Spalte "Name" wird um folgenden Zusatz ergänzt:
   `"AT<Name der überlagerten Variable>"`
6. Klicken Sie erneut auf die Schaltfläche zur Datentypauswahl und wählen Sie den Datentyp für die neue Variable aus.
   Die Variable wird angelegt. Sie zeigt auf dieselben Daten wie die überlagerte Variable, interpretiert diese jedoch mit dem neuen Datentyp.

Überlagerung entfernen

Um die Überlagerung einer Variablen wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Bausteinschnittstlle die überlagernde Variable aus, die Sie entfernen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Die Überlagerung wird entfernt.

Siehe auch

Variablen mit AT überlagern (Seite 116)

9.4.9 Multiinstanzen deklarieren

Voraussetzung

Der aufzurufende Funktionsbaustein ist in der Projektnavigation vorhanden und multiinstanzfähig.

Multiinstanz deklarieren

Um einen aufzurufenden Funktionsbaustein als Multiinstanz zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle des aufrufenden Funktionsbausteins.
2. Geben Sie im Abschnitt "Static" in der Spalte "Name" eine Bezeichnung für den Bausteinaufruf ein.
3. Geben Sie in der Spalte "Datentyp" einen symbolischen Namen für den aufzurufenden Funktionsbaustein an.

Hinweis

Der Programmiereditor deklariert die Multiinstanz selbst, wenn Sie in einem Netzwerk einen Bausteinaufruf programmieren und im nachfolgenden Dialog "Aufrufoptionen" angeben, dass Sie den Baustein als Multiinstanz aufrufen wollen.

Multiinstanzen als ARRAY deklarieren

Um ein ARRAY von Multiinstanzen zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle des aufrufenden Funktionsbausteins.
2. Geben Sie im Abschnitt "Static" in der Spalte "Name" eine Bezeichnung für das ARRAY ein.
3. Wählen Sie in der Spalte "Datentyp" den Datentyp ARRAY aus.
   Der Dialog "Array" wird geöffnet.
4. Geben Sie im Eingabefeld "Datentyp" den Namen des Funktionsbausteins an, für den eine Instanz definiert werden soll. Der Bausteinname muss in Anführungszeichen stehen.
5. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.
   Beispiel für ein eindimensionales ARRAY:[0..3]
   Beispiel für ein dreidimensionales ARRAY:[0..3, 0..15, 0..33]
6. Bestätigen Sie die Eingabe.

Siehe auch

Bausteinschnittstelle aktualisieren (Seite 7314)

Deklaration von Instanzen (Seite 7301)

9.4.10 Parameterinstanzen deklarieren

Einführung

Die folgende Tabelle zeigt die Möglichkeiten zur Deklaration und Verwendung von Parameterinstanzen in der Bausteinschnittstelle:

Deklaration	Baustein-art	Verwendung
InOut	FB FC	Die Instanz, die Sie als Aktualparameter übergeben, können Sie anschließend innerhalb des Funktionsbausteins aufrufen.
Input	FC	Sie können auf die Daten der Instanz, die Sie als Aktualparameter übergeben, lesend zugreifen. Sie können die Instanz jedoch nicht aus dem Baustein heraus aufrufen.
Output	FC	Sie können auf die Daten der Instanz, die Sie als Aktualparameter übergeben, schreibend zugreifen. Sie können die Instanz jedoch nicht aus dem Baustein heraus aufrufen.

Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

Vorgehen

Um eine Parameterinstanz zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den gewünschten Abschnitt der Bausteinschnittstelle (siehe Tabelle).
2. Geben Sie in der Spalte "Name" eine Bezeichnung für den Durchgangsparameter ein, an den die Instanz übergeben wird.
3. Geben Sie in der Spalte "Datentyp" den Namen des Funktionsbausteins an, für den eine Instanz definiert werden soll. Der Bausteinname muss in Anführungszeichen stehen.

Hinweis

Parameterinstanz automatisch deklarieren

Der Programmiereditor deklariert die Parameterinstanz selbst, wenn Sie in einem Netzwerk einen Bausteinaufruf programmieren und im nachfolgenden Dialog "Aufrufoptionen" angeben, dass Sie den Baustein als Parameterinstanz aufrufen wollen.

Siehe auch

Parameterinstanzen (Seite 70)

Deklaration von Instanzen (Seite 7301)

9.4.11 Parameter beim Bausteinaufruf verstecken

Einführung

Sie haben die Möglichkeit, Bausteinparameter beim Aufruf des Bausteins in KOP oder FUP zu verstecken. Versteckte Parameter sind zunächst nicht sichtbar, können jedoch über einen kleinen Pfeil am unteren Rand der Aufrufbox eingeblendet werden.

Um einen Parameter zu verstecken, stehen zwei Optionen zur Verfügung:

- Verstecken
  Der Parameter wird immer versteckt.
- Verstecken, wenn kein Parameter zugewiesen ist
  Der Parameter wird versteckt, solange er nicht beschaltet ist. Wenn sie einen Aktualparameter zuweisen, wird der Parameter an der Aufrufbox angezeigt.

Vorgehen

Um zu definieren, ob ein Bausteinparameter beim Aufruf angezeigt oder versteckt werden soll, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Selektieren Sie in der Bausteinschnittstelle einen Parameter.
3. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
4. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
5. Wählen Sie unter "Sichtbarkeit bei Bausteinaufrufen in KOP/FUP" einer der folgenden Optionen:
   - "Anzeigen"
   - "Verstecken"
   - "Verstecken, wenn kein Parameter zugewiesen ist"

Ergebnis

Der Parameter wird beim Bausteinaufruf in KOP oder FUP entsprechend der Einstellung angezeigt oder versteckt.

Wird beim Bausteinaufruf ein Syntaxfehler entdeckt, kann der Parameter nicht versteckt werden.

Siehe auch

Versteckte Parameter beschalten (Seite 7467)

Vordefinierte Aktualparameter deklarieren (Seite 7285)

9.4.12 Vordefinierte Aktualparameter deklarieren

Einführung

Sie können bereits bei der Deklaration eines Bausteinparameters angeben, welcher Aktualparameter beim Bausteinaufruf verwendet werden soll.

Dies kann nützlich sein, wenn Sie Programmbausteine als Bibliothekselemente verwenden und Informationen über die zu verwendenden Aktualparameter zusammen mit dem Bibliothekselement ablegen wollen.

Darüber hinaus ist es in KOP und FUP möglich, Parameter, die einen gültigen vordefinierten Aktualparameter haben, beim Bausteinaufruf zu verstecken. Versteckte Parameter sind zunächst nicht sichtbar, können jedoch über einen kleinen Pfeil am unteren Rand der Aufrufbox eingeblendet werden.

Um einen vordefinierten Parameter zu verstecken, stehen zwei Optionen zur Verfügung:

- Der Parameter wird immer versteckt.
- Der Parameter wird versteckt, solange der vordefinierte Aktualparameter zugewiesen ist. Wenn jedoch ein anderer Aktualparameter zugewiesen wird, wird der Parameter sichtbar.

Voraussetzung

Der Parameter befindet sich im Abschnitt "Input", "Output" oder "InOut".

Vorgehen

Um einen Aktualparameter vorzudefinieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Selektieren Sie in der Bausteinschnittstelle einen Parameter.
3. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
4. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
5. Geben Sie im Eingabefeld "Vordefinierter Aktualparameter" den gewünschten Aktualparameter ein.
6. Wählen Sie optional unter "Sichtbarkeit bei Bausteinaufrufen in KOP/FUP" die Option "Verstecken".
7. Wählen Sie als zusätzliche Option "Anzeigen, wenn der zugewiesene Aktualparameter nicht mit dem vordefinierten Aktualparameter identisch ist". Diese Option ist nur verfügbar, wenn die Option "Verstecken" zuvor aktiviert wurde.

Ergebnis

- Ein Aktualparameter ist vordefiniert. Wenn Sie den Programmbaustein als Bibliothekselement abspeichern, enthält er somit auch die Information über den zu verwendenden Aktualparameter.
- Wenn das Bibliothekselement im Programm verwendet wird, wird geprüft, ob der von Ihnen vordefinierte Aktualparameter adressiert werden kann. Ist dies der Fall, wird er automatisch als Aktualparameter eingesetzt.
- Wird der Aktualparameter im Programm nicht gefunden, wird ein Syntaxfehler gemeldet und sie müssen die Parameterversorgung manuell durchführen. Parameter mit ungültiger Versorgung werden nicht versteckt.

Siehe auch

Parameter beim Bausteinaufruf verstecken (Seite 7284)

9.4.13 Tabelle der Bausteinschnittstelle bearbeiten

9.4.13.1 Tabellenzeilen einfügen

Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

9.4.13.2 Tabellenzeilen hinzufügen

Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

9.4.13.3 Variablen löschen

Vorgehen

Um Variablen zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variable. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

9.4.13.4 Aufeinander folgende Zellen automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

Voraussetzung

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.
   Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.
   Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

9.4.13.5 Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
5. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.

9.4.13.6 Variablen mit externen Editoren bearbeiten

Um einzelne Variablen in externen Tabelleneditoren, z. B. in Excel, zu bearbeiten, können Sie sie per Copy & Paste exportieren oder importieren.

Hinweis

Einschränkung beim Anzeigen und Kopieren von ARRAY-Elementen

Abhängig von der Größe des Arbeitsspeichers Ihres Computers gelten bei der Anzeige und beim Kopieren von ARRAYs aus der Bausteinschnittstelle in eine Excel-Tabelle folgende Maximalgrenzen:

- Arbeitsspeicher < 8 GB: Es werden maximal 10.000 Elemente angezeigt und kopiert.
- Arbeitsspeicher >= 8 GB: Es werden maximal 40.000 Elemente angezeigt und kopiert.

Voraussetzung

Die Bausteinschnittstelle und ein externer Editor sind geöffnet.

Vorgehen

Um einzelne Variablen in einen externen Editor zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Um alle Elemente strukturierter Datentypen einzublenden, klicken Sie auf die Schaltfläche "Erweiterter Modus".
2. Selektieren Sie eine oder mehrere Variablen.
3. Wählen Sie im Kontextmenü den Befehl "Kopieren".
4. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
5. Bearbeiten Sie die Variablen nach Bedarf.
6. Kopieren Sie die Variablen im externen Editor.
7. Selektieren Sie die Variablen im externen Editor.
8. Wechseln Sie wieder in die Bausteinschnittstelle.
9. Wählen Sie im Kontextmenü den Befehl "Einfügen".

9.5 Bausteinschnittstelle in der textuellen Ansicht deklarieren

9.5.1 Aufbau der textuellen Bausteinschnittstelle

Aufbau der textuellen Bausteinschnittstelle

Das folgende Bild zeigt den Aufbau der textuellen Bausteinschnittstelle, so wie sie nach dem Anlegen eines neuen Funktionsbausteins dargestellt wird:

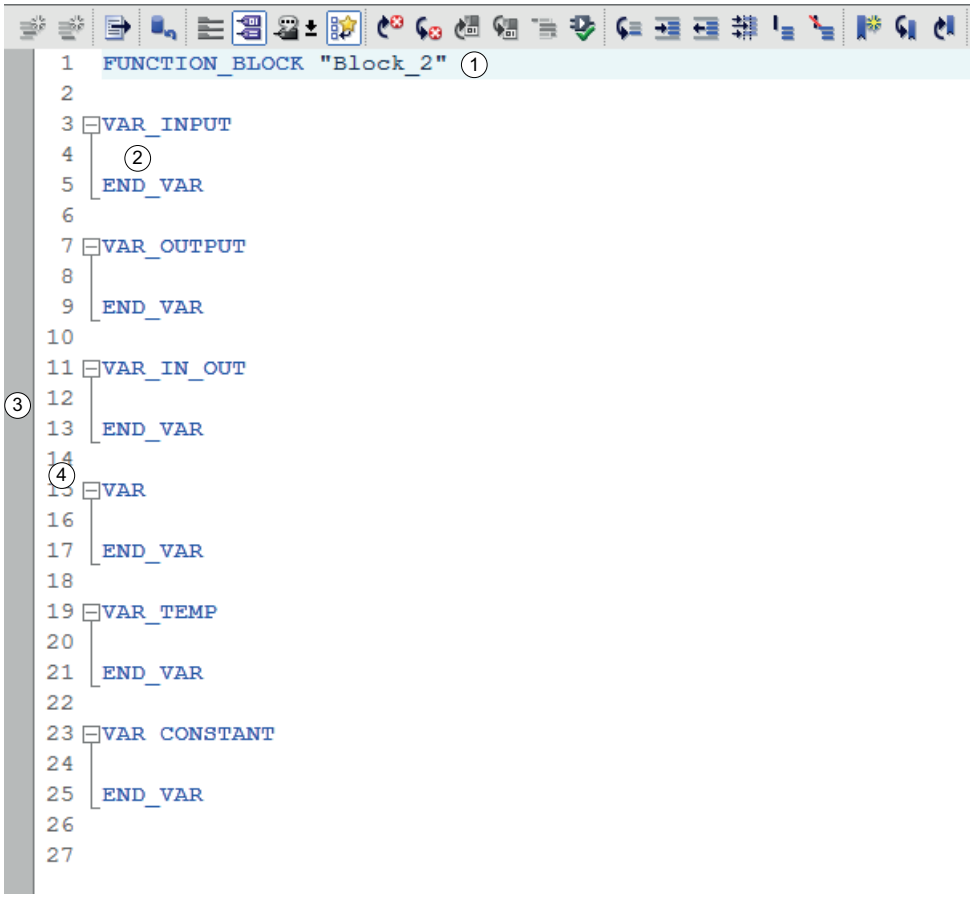

Die textuelle Bausteinschnittstelle besteht aus folgenden Bereichen:

Bereich	Bedeutung
① Bausteindeklaration	Die Bausteindeklaration definiert den Name und die Art des Bausteins.
② Deklarationsabschnitte	In den Deklarationsabschnitten deklarieren Sie die Parameter und Lokaldaten des Bausteins. In der leeren Bausteinschnittstelle sind die Deklarationsabschnitte bereits eingetragen. Sie können die Abschnitte jedoch ganz beliebig verschieben, kopieren oder löschen.

Bereich	Bedeutung
③ Seitenleiste	In der Seitenleiste können Sie Lesezeichen setzen.
④ Zeilennummern	Links neben den Deklarationsabschnitten werden die Zeilennummern angezeigt.

Siehe auch

Tabellarische oder textuelle Darstellung der Bausteinschnittstelle wählen (Seite 7271)

9.5.2 Bausteindeklaration und Rückgabewert

Bausteindeklaration

Die textuelle Bausteinschnittstelle beginnt mit der Bausteindeklaration. Die Bausteindeklaration müssen Sie nicht selbst eingeben. Sie ist im Template der Schnittstelle bereits enthalten.

Rückgabewert

Bei Funktionen (FC) geben Sie nach der Bausteindeklaration den Rückgabewert der Funktion angeben. Der Rückgabewert ist der Wert, der an den aufrufenden Baustein zurückgeliefert wird. Wenn Sie den Wert "VOID" angeben, liefert die Funktion keinen Rückgabewert.

Syntax

Für die Bausteindeklaration und den Rückgabewert wird die folgende Syntax verwendet:

Bausteinart	Syntax	Beispiel
Funktion (FC)	FUNCTION <Name> [:Datentyp Rückgabewert>] <Deklaration>	`FUNCTION "My_Function" : Int` `FUNCTION "My_Function" : Void`
Funktionsbaustein (FB)	FUNCTION_BLOCK <Name> <Deklaration>	`FUNCTION_BLOCK "My_FunctionBlock"`
Organisationsbaustein (OB)	ORGANIZATION_BLOCK <Name> <Deklaration>	`ORGANIZATION_BLOCK "My_OrganizationBlock"`

Siehe auch

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.3 Deklarationsabschnitte

Deklarationsabschnitte

Die textuelle Bausteinschnittstelle gliedert sich in unterschiedliche Deklarationsabschnitte, die durch Schlüsselwortpaare gekennzeichnet sind. Je nach Bausteinart sind unterschiedliche Abschnitte zulässig.

Die Reihenfolge der Deklarationsabschnitte ist beliebig. Ein Abschnitt kann mehrmals in der Bausteinschnittstelle vorkommen.

Für die Deklarationsabschnitte wird die folgende Syntax verwendet:

Deklarationsabschnitt	Syntax
Eingangsparameter	VAR_INPUT [<ATTRIBUT>] <Deklaration> END_VAR
Ausgangsparameter	VAR_OUTPUT [<ATTRIBUT>] <Deklaration> END_VAR
Durchgangsparameter	VAR_IN_OUT [<ATTRIBUT>] <Deklaration> END_VAR
Temporäre Lokaldaten	VAR_TEMP <Deklaration> END_VAR
Statische Lokaldaten	VAR [<ATTRIBUT>] <Deklaration> END_VAR
Konstante	VAR CONSTANT <Deklaration> END_VAR

Informationen zur Bedeutung der einzelnen Deklarationsabschnitte finden Sie unter:

Übersicht über die Bausteinschnittstelle (Seite 7263)

Attribute für Deklarationsabschnitte

Optional können Sie den Abschnitten Attribute zuweisen. Die Attribute gelten für alle Variablen, die in diesem Abschnitt deklariert sind. Wenn ein Abschnitt mehrmals vorkommt, können Sie für jedes Vorkommen andere Attribute verwenden.

Die folgenden Attribute stehen zur Verfügung:

Attribut	Bedeutung
RETAIN	Die Variablen in diesem Abschnitt sind remanent, das heißt, ihre Werte stehen nach Spannungsausfall weiterhin zur Verfügung.
NON_RETAIN	Die Variablen in diesem Abschnitt sind nicht remanent, das heißt, ihre Werte gehen nach Spannungsausfall verloren.
DB_SPECIFIC	Die Remanenz kann im Instanz-Datenbaustein eingestellt werden

Wenn Sie kein Attribut angeben, sind die Daten des Deklarationsabschnitts nicht remanent.

Siehe auch

Remanenz von lokalen Variablen (Seite 7269)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.4 Variablendeklaration und Initialisierung

Deklaration und Initialisierung von Variablen oder Konstanten

In den einzelnen Deklarationsabschnitten deklarieren Sie die lokalen Variablen und Konstanten, die innerhalb des Bausteins verwendet werden.

Eine Variablendeklaration besteht aus einem symbolischen Namen und einem Datentyp. Optional können Sie einen Defaultwert zur Initialisierung der Variablen angeben.

Eine Konstantendeklaration enthält neben dem symbolischen Namen und dem Datentyp immer einen Wert. Konstantendeklarationen sind nur im Abschnitt "VAR CONSTANT" zulässig.

Variablen mit AT überlagern

Um auf Datenbereiche innerhalb einer deklarierten Variablen zuzugreifen, können Sie die deklarierte Variable mit einer weiteren Deklaration überlagern. Sie haben so die Möglichkeit, eine bereits deklarierte Variable mit einem anderen Datentyp anzusprechen. Sie können z. B. die einzelnen Bits einer Variablen vom Datentyp WORD mit einem ARRAY of BOOL ansprechen.

Während eine AT-Deklaration in der Tabellenansicht immer direkt hinter der referenzierten Variable stehen muss, kann sie in der textuellen Ansicht an einer beliebigen Position stehen.

AT-Deklarationen können nicht initialisiert werden. Beachten Sie, dass AT-Deklarationen grundsätzlich nur in nicht-optimierten Bausteinen genutzt werden können.

Syntax

Für die Deklaration von Variablen und Konstanten wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele
Variable	<Name> : <Datentyp> [:= <Wert>];	`myBit : BOOL;` `myBit : BOOL := true;`

Konstante	<Name> : <Datentyp> := <Wert>;	`PI : REAL := 3.141592;`
		`myInt: INT := INT#16#7FFF;`
		`myString: STRING := 'hello';`

AT-Deklaration	<Name> AT <Name der referenzierten Variable> : <Datentyp>;	`myReferenceToVar2 AT Var_2 : Int;`

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

Allgemeine Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Variablen mit AT überlagern (Seite 116)

Überlagerte Variablen deklarieren (Seite 7281)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

Konstanten (Seite 102)

9.5.5 Deklaration von STRUCT

Deklaration von STRUCT

Der Datentyp STRUCT repräsentiert eine Datenstruktur, die sich aus einer festen Anzahl von Komponenten unterschiedlicher Datentypen zusammensetzt.

Eine Struktur kann auch auf einem PLC-Datentypen (UDT) basieren. Dazu geben Sie den PLC-Datentypen als Datentyp der Struktur an. Ein Tooltip zeigt Ihnen die einzelnen Elemente des PLC-Datentyps an.

Syntax

Für die Deklaration eines STRUCT wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele

STRUCT	<Name> : Struct
	<Elementname> : <Datentyp> [:= <Wert>];
	<Elementname> : <Datentyp> [:= <Wert>];
	...
	END_Struct := (<Initialisierungsliste>);
		`myStruct : Struct`
		`mem_1 : Int;`
		`mem_2 : Int;`
		`mem_3 : BOOL;`
		`END_Struct := (6,0, TRUE) ;`

STRUCT auf Basis eines PLC-Datentyps (UDT)	<Name> : "<UDT-Name>" := (<Initialisierungsliste>);	`myStruct : "myType" := (6,0, TRUE);`

Hinweis

S7-1500: Maximal 252 Strukturen in einem Datenbaustein

In einem Datenbaustein sind auf CPUs der Baureihe S7-1500 maximal 252 Strukturen zulässig. Wenn Sie weitere Strukturen benötigen, müssen Sie Ihr Programm umstrukturieren. Beispielsweise können Sie die Strukturen in mehreren globalen Datenbausteinen anlegen.

Initialisierung von STRUCT

Die Initialisierung von STRUCT ist optional. Geben Sie zur Initialisierung einen Wert direkt nach der Variablendeklaration ein oder verwenden Sie eine Initialisierungsliste.

Eine Initialisierungsliste steht nach dem Schlüsselwort END_STRUCT in runden Klammern und wird mit := eingeleitet. Wenn die Struktur unterlagerte Strukturen enthält, verwenden Sie zur Initialisierung geschachtelte Klammern. Wenn sie versehentlich beide Möglichkeiten der Initialisierung nutzen, wird der Wert aus der Initialisierungsliste verwendet.

Für die Initialisierungsliste wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele	Kommentar

Initialisierung von STRUCT	<Name> : Struct
	<Elementname> : <Datentyp> [:= <Wert>];
	...
	END_Struct
		`myStruct : Struct`
		`mem_1 : Int := 1;`
		`mem_2 : Int := 2;`
		`mem_3 : BOOL := FALSE`
		`End_Struct`
			Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 1
			// mem_2 := 2
			// mem_3 := FALSE

Initialisierungsliste für STRUCT	:= (<Wert>,<Wert>,<Wert>...)	`myStruct : Struct`
		`mem_1 : Int;`
		`mem_2 : Int;`
		`mem_3 : BOOL`
		`End_Struct:=(2,0,TRUE);`
			Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 2
			// mem_2 := 0
			// mem_3 := TRUE

Deklaration	Syntax	Beispiele	Kommentar

Initialisierungsliste für geschachteltes STRUCT	:= (<Wert>,(<Wert>,<Wert>...))	`myStruct : Struct`
		`mem_1 : Int;`
		`mem_2 : Int;`
		`mem_3 : Struct`
		`mem_4 : BOOL;`
		`mem_5 : BOOL;`
		`End_Struct;`
		`End_Struct:=(2,0, (TRUE,TRUE));`
			Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 2
			// mem_2 := 0
			// mem_4 := TRUE
			// mem_5 := TRUE

Initialisierungsliste für STRUCT mit Angabe des Elementnamens	:= (<Elementname> := <Wert>), (<Elementname> := <Wert>)...	`myStruct : Struct`
		`mem_1 : Int;`
		`mem_2 : Int;`
		`End_Struct:=(mem_2:=55);`
			Die Strukturelemente werden wie folgt initialisiert:
			// mem_2 := 55

Initialisierungsliste für PLC-Datentyp (UDT)	:= (<Wert>,<Wert>,<Wert>...)	`myStruct : "myType" := (2,0,TRUE);`	Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 2
			// mem_2 := 0
			// mem_3 := TRUE

Initialisierungsliste für geschachtelten PLC-Datentyp (UDT)	:= (<Wert>,(<Wert>,<Wert>...))	`myStruct : "myType" := (2,0, (TRUE,TRUE);`	Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 2
			// mem_2 := 0
			// mem_3 := TRUE
			// mem_4 := TRUE

Initialisierungsliste für PLC-Datentyp (UDT) mit Angabe des Elementnamens	:= (<Elementname> := <Wert>), (<Elementname> := <Wert>)...	`myStruct : "myType" := (mem_1:=22,mem_2:=55);`	Die Strukturelemente werden wie folgt initialisiert:
			// mem_1 := 22
			// mem_2 := 55

Siehe auch

Variablen vom Datentyp STRUCT deklarieren (Seite 7279)

Datenstruktur STRUCT (Anonyme Strukturen) (Seite 291)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.6 Deklaration von ARRAY

Deklaration von ARRAY

Der Datentyp ARRAY repräsentiert eine Datenstruktur, die aus einer festen Anzahl von Elementen des gleichen Datentyps besteht.

Für die Deklaration eines ARRAY wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele
ARRAY	<Name> : ARRAY [unterer Grenzwert..oberer Grenzwert] OF <Datentyp> := [<Initialisierungsliste>];	`MyARRAY_1 : ARRAY[0..7] OF BOOL;` `MyARRAY_1 : ARRAY[0..7] OF BOOL := [1,1,0,0,0,1,0,0];`
ARRAY mit variablen Grenzen	<Name> : ARRAY [*] OF <Datentyp>;	`MyARRAY_1 : ARRAY[*] OF INT;` `MyARRAY_2 : ARRAY[*, *, *] OF INT;`

ARRAY mit lokalen Konstanten als Grenzen	<Name> : ARRAY [#<Konstantenname>..#<Konstantenname>] OF <Datentyp> := [<Initialisierungsliste>];	`MyARRAY_1 : ARRAY[#LocConst1..#LocConst2] OF INT;`
		`MyARRAY_2 : ARRAY[1..#LocConst] OF INT;`
		`MyARRAY_3 : ARRAY[1..#LocConst] OF INT := [1,1,0,0];`

ARRAY mit globalen Konstanten als Grenzen	<Name> : ARRAY ["<Konstantenname>".."<Konstantenname>"] OF <Datentyp> := [<Initialisierungsliste>];	`MyARRAY_1 : ARRAY["GlobConst1".."GlobConst2"] OF INT;`
		`MyARRAY_2 : ARRAY[1.."GlobConst", 2..5,#1..#u] OF INT;`
		`MyARRAY_3 : ARRAY[1.."GlobConst"] OF INT:= [1,1,0,0];`

ARRAY OF STRUCT	<Name> : ARRAY[unterer Grenzwert..oberer Grenzwert] OF Struct
	<Elementname> : <Datentyp>;
	<Elementname> : <Datentyp>;
	...
	END_Struct := [<Initialisierungsliste>];
		`MyARRAY_1 : Array[0..1] OF Struct`
		`mem_1 : Int;`
		`mem_2 : Int;`
		`END_STRUCT := [ (2,4),(22,44) ];`

ARRAY OF UDT	<Name> : ARRAY[unterer Grenzwert..oberer Grenzwert] OF "<UDT_ Name>" := [<Initialisierungsliste>];	`MyARRAY_1 : Array[0..1] OF MyType := [ (2,4),(22,44) ];`

Initialisierung von ARRAY

Die Initialisierung eines ARRAY ist optional. Sie verwenden dazu eine Initialisierungsliste. Sie steht nach der Deklaration in eckigen Klammern und wird mit := eingeleitet. Die Liste enthält durch Komma getrennte Werte für alle Elemente des ARRAY. Sie können maximal 10000 Elemente initialisieren.

Optional können Sie Wiederholungsfaktoren in runden Klammern verwenden, um mehrere aufeinanderfolgende Elemente mit demselben Wert zu initialisieren. Der Wiederholungsfaktor muss eine positive Ganzzahl sein.

Bei ARRAY OF STRUCT haben Sie optional auch die Möglichkeit, elementweise zu initialisieren. Dabei weisen Sie zentral allen Strukturelementen eines Typs einen Wert zu.

Für die Initialisierungsliste wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele	Kommentar

Initialisierungsliste	:= [<Wert>,<Wert>,<Wert>...]	`myArray : ARRAY[0..2] OF BOOL := [1,1,0];`	Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0] := 1
			// myArray[1] := 1
			// myArray[2] := 0

Initialisierungsliste mit Wiederholungsfaktor	:= [<Wert>,<Wiederholungsfaktor>(Wert),<Wert>...]	`myArray : ARRAY[1..2, 1..3] OF INT := [9,8,3(10),6];`	Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[1,1] := 9
			// myArray[1,2] := 8
			// myArray[1,3] := 10
			// myArray[2,1] := 10
			// myArray[2,2] := 10
			// myArray[2,3] := 6

Initialisierungsliste für ARRAY of STRUCT	:= [(<Werteliste Element 1>), (<Werteliste Element 2>)...]	`myArray : Array[0..1] OF Struct`
		`  Element1 : Int;`
		`  Element2 : Int;`
		`END_Struct := [ (2,4), (22,44) ];`
			Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0].Element1 := 2
			// myArray[0].Element2 := 4
			// myArray[1].Element1 := 22
			// myArray[1].Element2 := 44

Initialisierungsliste für ARRAY of STRUCT mit Angabe des Elementnamens	:= [(<Elementname> := <Wert>), (<Elementname> := <Wert>)...]	`myArray : Array[0..1] OF Struct`
		`  Element1 : Int;`
		`  Element2 : Int;`
		`END_Struct := [ (Element1 := 2 , Element2:=4), (Element1 := 22, Element2:=44) ];`
			Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0].Element1 := 2
			// myArray[0].Element2 := 4
			// myArray[1].Element1 := 22
			// myArray[1].Element2 := 44

Deklaration	Syntax	Beispiele	Kommentar

Initialisierungsliste für ARRAY of STRUCT mit Wiederholungsfaktor	:= [<Wiederholungsfaktor>(Werteliste Element 1),<Wiederholungsfaktor>(Werteliste Element 2),...]	`myArray : Array[0..1, 0..1] OF Struct`
		`  Element_x : Int;`
		`  Element_y : Int;`
		`End_Struct:=[2((1, 11)), 2((2, 22))];`
			Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0,0].Element_x:= 1
			// myArray[0,0].Element_y:= 11
			// myArray[0,1].Element_x:= 1
			// myArray[0,1].Element_y:= 11
			// myArray[1,0].Element_x:= 2
			// myArray[1,0].Element_y:= 22
			// myArray[1,1].Element_x:= 2
			// myArray[1,1].Element_y:= 22

Initialisierungsliste für ARRAY of UDT	:= [(Werteliste Element 1>), (<Werteliste Element 2>)...]	`myArray : Array[0..1] OF MyType := [ (2,4),(22,44) ];`	Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0].UDT-Element1 := 2
			// myArray[0].UDT-Element2 := 4
			// myArray[1].UDT-Element1 := 22
			// myArray[1].UDT-Element2 := 44

Initialisierungsliste für ARRAY of UDT mit Angabe des Elementnamens	:= [(<Elementname> := <Wert>), (<Elementname> := <Wert>)...]	`myArray : Array[0..1] OF MyType := [ (UDT-Element1 := 2,UDT-Element2:=4),(UDT-Element1 := 22,UDT-Element2:=44) ];`	Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0].UDT-Element1 := 2
			// myArray[0].UDT-Element2 := 4
			// myArray[1].UDT-Element1 := 22
			// myArray[1].UDT-Element2 := 44

Initialisierungsliste für ARRAY of UDT mit Wiederholungsfaktor	:= [<Wiederholungsfaktor>(Werteliste Element 1),<Wiederholungsfaktor>(Werteliste Element 2),...]	`myArray : Array[0..1] OF myType :=[2((((),1))];`	Die ARRAY-Elemente werden wie folgt initialisiert:
			// myArray[0].UDT-Element1 := 0
			// myArray[0].UDT-Element2 := 1
			// myArray[1].UDT-Element1 := 0
			// myArray[1].UDT-Element2 := 1

Siehe auch

ARRAY (Seite 296)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.7 Deklaration von STRING und WSTRING

Deklaration von STRING und WSTRING

Die Datentypen STRING und WSTRING speichern mehrere Zeichen in einer Zeichenkette. In der Zeichenkette sind alle Zeichen des ASCII-Codes zugelassen. Die Zeichen werden in einfachen Anführungszeichen angegeben.

Optional können Sie einen Defaultwert zur Initialisierung der Zeichenkette angeben.

Die Maximallänge der Zeichenkette kann bei der Deklaration eines Operanden mithilfe von eckigen Klammern nach dem Schlüsselwort STRING bzw. WSTRING angegeben werden (z. B. STRING[4]). Zur Deklaration der Maximallänge können Sie einen absoluten Wert eingeben oder lokale bzw. globale Konstanten verwenden.

Wenn die Angabe der Maximallänge weggelassen wird, wird für den jeweiligen Operanden die Standardlänge von 254 Zeichen eingestellt.

Syntax

Für die Deklaration von STRING und WSTRING wird die folgende Syntax verwendet:

Deklaration	Syntax	Beispiele
STRING	`<Name> : STRING [:= <Wert>];`	`myString: STRING;` `myString: STRING := 'hello';`
WSTRING	`<Name> : WSTRING [:= <Wert>];`	`myWstring: WSTRING;` `myWstring_var: WSTRING := 'helloWorld';`

STRING mit definierter Maximallänge	`<Name> : STRING[[Konstante]];`	`myString: STRING[10];`
		`myString: STRING["globConst"];`
		`myString: STRING[#locConst];`

Siehe auch

STRING (Seite 277)

WSTRING (Seite 280)

Variablen vom Datentyp STRING und WSTRING deklarieren (Seite 7280)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.8 Deklaration von Instanzen

Deklaration von Multiinstanzen

Multiinstanzen deklarieren Sie im Abschnitt "VAR...END_VAR" der textuellen Bausteinschnittstelle. Sie setzen dabei einfach den Namen des aufgerufenen Bausteins als Datentyp ein.

Multiinstanzen können auch als ARRAY angelegt werden.

Siehe auch:

Multiinstanzen (Seite 67)

Multiinstanzen deklarieren (Seite 7282)

Deklaration von Parameterinstanzen

Parameterinstanzen deklarieren Sie am besten als Durchgangsparameter im Abschnitt "VAR_IN_OUT...END_VAR" der textuellen Bausteinschnittstelle. Sie sind jedoch in Funktionen (FC) auch als Eingangs- oder Ausgangsparameter möglich. Sie setzen dabei einfach den Namen des aufgerufenen Bausteins als Datentyp ein.

Siehe auch:

Parameterinstanzen (Seite 70)

Parameterinstanzen deklarieren (Seite 7283)

Instanz	Syntax	Beispiele	Bedeutung

Multiinstanz	VAR
	<Multiinstanzname>:"<Bausteinname>";
	END_VAR
		`VAR`
		`instFB : "MyFB";`
		`END_VAR`
			"instFB" ist eine Multiinstanz des Bausteins "MyFB".

ARRAY von Multiinstanzen	VAR
	<Multiinstanzname>: ARRAY [n...m] of "<Bausteinname>";
	END_VAR
		`VAR`
		`instArray : ARRAY [0…4] of "MyFB";`
		`END_VAR`
			"instArray" ist ein Array von Multiinstanzen des Bausteins "MyFB".

Parameterinstanz	VAR
	<Parameterinstanzname>:"<Bausteinname>";
	END_VAR
		`VAR_IN_OUT`
		`instParam:"MyFB";`
		`END_VAR`
			"instParam" ist eine Parameterinstanz des Bausteins "MyFB".

Siehe auch

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.9 Attribute für Variablen

Attribute für Variablen

Optional können Sie den einzelnen Variablen Attribute zuweisen. Wenn Sie einer Variablen nicht explizit ein Attribut zuweisen, besitzt sie die vorbelegten Attributwerte. Die Voreinstellung einiger Attribute können Sie in den allgemeinen Einstellungen zur PLC-Programmierung ändern.

Folgende Attribute stehen zur Verfügung:

Attribut	Bedeutung	Vorbelegung
Sichtbar in HMI Engineering	Zeigt an, ob eine Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.	TRUE
Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.	TRUE
Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.	FALSE
Einstellwert	Kennzeichnet eine Variable als Einstellwert. Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Die Spalte ist nur in der Schnittstelle von Funktionsbausteinen vorhanden.	FALSE

Syntax

Für Attribute wird die folgende Syntax verwendet:

Attribut	Syntax	Beispiele
Sichtbar in HMI Engineering	<Name> {EXTERNALVISIBLE:=<Wert>}: <Datentyp>;	`myInt {EXTERNALVISIBLE:= 'False'} : INT;`
Erreichbar aus HMI / OPC UA	<Name> {EXTERNALACCESSIBLE:=<Wert>}: <Datentyp>;	`myInt {EXTERNALACCESSIBLE:= 'False'} : INT;`
Schreibbar aus HMI / OPC UA	<Name> {EXTERNALWRITABLE:=<Wert>}: <Datentyp>;	`myInt {EXTERNALWRITABLE:= 'False'} : INT;`
Einstellwert	<Name> {S7_SETPOINT:=<Wert>}: <Datentyp>;	`myInt {S7_SETPOINT:= 'False'} : INT;`

Siehe auch

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

Allgemeine Einstellungen für die PLC-Programmierung (Seite 7150)

9.5.10 Kommentare

Kommentare

Sie haben verschiedene Möglichkeiten, eine Variable in der textuellen Bausteinschnittstelle zu kommentieren:

- Zeilenkommentar
  Ein Zeilenkommentar wird mit "//" eingeleitet und erstreckt sich bis zum Ende der Zeile.
- Kommentarabschnitt
  Ein Kommentarabschnitt wird mit "(*" eingeleitet und durch "*)" abgeschlossen. Er kann sich über mehrere Zeilen erstrecken.

Für Kommentare gelten folgende Regeln:

- Kommentare können nicht ineinander geschachtelt werden.
- Mehrsprachige Kommentare sind in der textuellen Bausteinschnittstelle nicht möglich.
- Zwischen der Bausteindeklaration und dem ersten Deklarationsabschnitt sind Kommentare nicht zulässig.

Syntax

Für Kommentare wird die folgende Syntax verwendet:

Attribut	Syntax	Beispiele
Zeilenkommentar	<Deklaration>; //<Kommentar>	`MyInt : INT; // das ist mein Kommentar`

Kommentarabschnitt	<Deklaration>; (*<Kommentar>*)	`MyInt : INT; (* das sind meine Kommentare:`
		`1. Kommentar`
		`2. Kommentar*)`

Siehe auch

Mehrsprachige Kommentare in der Bausteinschnittstelle bearbeiten (Seite 7312)

Beispiel einer textuellen Bausteinschnittstelle (Seite 7306)

9.5.11 Textuelle Bausteinschnittstelle bearbeiten

Bei der Bearbeitung der textuellen Bausteinschnittstelle können Sie dieselben Bearbeitungsfunktionen nutzen, wie bei der Bearbeitung des Programmcodes.

Autovervollständigung nutzen

1. Geben Sie die Syntax der Deklaration über die Tastatur ein.
   Während der Eingabe öffnet sich die Autovervollständigung. Sie bietet alle Syntaxelemente an, die an der aktuellen Position zulässig sind.
2. Wählen Sie aus der Autovervollständigung das gewünschte Element aus.

Siehe auch: SCL-Anweisungen manuell eingeben (Seite 7643)

Zeilen automatisch formatieren

1. Selektieren Sie den Text, den Sie formatieren möchten, oder setzen Sie die Einfügemarke in die entsprechende Zeile.
2. Wählen Sie in der Funktionsleiste des Programmiereditors die Schaltfläche "Selektierten Text automatisch formatieren".
   Alle markierten Zeilen werden formatiert, sofern sie syntaktisch korrekt sind.

Siehe auch: SCL-Code formatieren (Seite 7635)

Einzelne Zeilen ein- oder ausrücken

1. Klicken Sie in die Zeilen, die Sie ein- oder ausrücken möchten.
2. Wählen Sie in der Funktionsleiste des Programmiereditors die Schaltfläche "Text einrücken" oder "Text ausrücken".
   Die Breite der Einrückung können Sie über "Extras > Einstellungen" definieren.

Siehe auch: SCL-Code formatieren (Seite 7635)

Codeabschnitte auf- und zuklappen

1. Klicken Sie auf das Minus-Zeichen in der Gliederungsanzeige.
   Der Codeabschnitt wird zugeklappt.
2. Klicken Sie auf das Plus-Zeichen in der Gliederungsanzeige.
   Der Codeabschnitt wird wieder aufgeklappt.

Siehe auch: Codeabschnitte auf- und zuklappen (Seite 7637)

Lesezeichen verwenden

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen setzen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Lesezeichen setzen/löschen".

Hinweis

Navigation zwischen Schnittstelle und Programmcode

Wenn Sie sowohl in der Schnittstelle als auch im Programmocde Lesezeichen gesetzt haben, können Sie über die Befehle "Gehe zu nächstem/vorangehendem Lesezeichen" nicht zwischen den beiden Fenstern navigieren.

Siehe auch: Lesezeichen verwenden (Seite 7638)

Variablen zwischen textueller und tabellarischer Schnittstelle kopieren oder verschieben

Sie können Variablen zwischen einer textuellen Schnittstelle und einer tabellarische Schnittstelle kopieren oder verschieben.

1. Markieren Sie eine Variable oder mehrere Variablen eines Deklarationsabschnitts.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren" oder "Ausschneiden".
3. Setzen Sie den Mauszeiger an die Position, an der Sie die Variable einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Hinweis

Kopieren oder Verschieben von strukturierten Datentypen mit Initialisierungen

Beim Kopieren von geschachtelten strukturierten Datentypen (z. B. ARRAY of STRUCT) von einer textuellen Schnittstelle in eine tabellarische Schnittstelle werden die Defaultwerte der unterlagerten Strukturen nicht mitkopiert oder verschoben.

9.5.12 Standard-Schnittstelle eines Organisationsbausteins wiederherstellen

Einführung

Organisationsbausteine haben definierte Schnittstellenparameter, die die Startinformationen des OB enthalten. Falls Sie diese vordefinierten Parameter in der textuellen Bausteinschnittstelle ändern oder löschen, haben Sie bei optimierten Organisationsbausteinen die Möglichkeit, die definierte Standard-Schnittstelle wiederherzustellen.

Vorgehen

Um die Standard-Schnittstelle eines Organisationsbausteins wiederherzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Kontextmenü den Befehl "Standard-Schnittstelle wiederherstellen".

Ergebnis

- Gelöschte oder geänderte OB-Parameter werden wiederhergestellt.
- Parameter, die Sie manuell hinzugefügt haben, werden nicht gelöscht, sondern bleiben erhalten.

9.5.13 Beispiel einer textuellen Bausteinschnittstelle

Beispiel

Das folgende Bild zeigt ein Beispiel einer textuellen Bausteinschnittstelle:

```
1   FUNCTION_BLOCK "MyFB"
2
3   VAR_INPUT
4     myIn1 : Int := 5; //Declaration of an Input tag with initialization
5   END_VAR
6
7   VAR_INPUT RETAIN
8     myIn2 {EXTERNALACCESSIBLE := 'False'; EXTERNALVISIBLE := 'False'} :
9     //Declaration of an Input tag with attributes
10  END_VAR
11
12  VAR_OUTPUT RETAIN
13    myOut : Array[0..2] OF Byte; //Declaration of an ARRAY
14  END_VAR
15
16  VAR_IN_OUT
17
18  END_VAR
19
20  VAR
21    myStatic : "FB_SCL"; //Declaration of a multi-instance
22  END_VAR
23
24  VAR_TEMP
25
26  END_VAR
27
28  VAR CONSTANT
29    myConstant : Byte := 44; // Declaration of a constant
30  END_VAR
31
```

9.6 Eigenschaften von lokalen Variablen und Konstanten bearbeiten

9.6.1 Eigenschaften von lokalen Variablen und Konstanten

Eigenschaften

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von lokalen Variablen und Konstanten:

Gruppe	Eigenschaft	Beschreibung
Allgemein	Name	Name des Elements
	Datentyp	Datentyp des Elements
	Defaultwert	Wert, mit dem Sie bestimmte Variablen in der Schnittstelle des Codebausteins vorbelegen können, oder Wert einer lokalen Konstante. Die Angabe des Defaultwerts ist für Variablen optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Der Defaultwert einer Variablen wird in der zugehörigen Instanz als Startwert übernommen. Diese übernommenen Werte können Sie mit instanzspezifischen Startwerten ersetzen.
	Kommentar	Kommentar zum Element

Gruppe	Eigenschaft	Beschreibung
Attribute	Remanenz	Kennzeichnet die Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten. Dieses Attribut ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff verfügbar.
	Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
	Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
	Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI /OPC UA zur Laufzeit auf diese Variable zugreifen kann. Beachten Sie jedoch, dass Sie mit dem Attribut "Erreichbar aus HMI / OPC UA" keinen allgemeinen Zugriffsschutz für die Variable realisieren können. Der lesende oder schreibende Zugriff durch andere Applikationen kann möglich sein, auch wenn das Attribut nicht aktiviert ist.
	Sichtbarkeit bei Bausteinaufrufen in KOP/FUP	Anzeigen: Der Parameter ist bei Bausteinaufrufen in KOP oder FUP immer sichtbar.
		Verstecken: Der Parameter wird bei Bausteinaufrufen in KOP oder FUP immer versteckt. Die Aufrufbox verfügt dann über einen kleinen Pfeil am unteren Rand. Durch Klicken auf diesen Pfeil können Sie den Parameter einblenden und beschalten.
		Verstecken, wenn kein Parameter zugewiesen ist: Der Parameter wird bei Bausteinaufrufen in KOP oder FUP versteckt, solange er nicht beschaltet ist. Wenn sie einen Aktualparameter zuweisen, wird der Parameter an der Aufrufbox angezeigt.
	Vordefinierter Aktualparameter	Definiert einen Parameter, der beim Bausteinaufruf als Aktualparameter verwendet werden soll.
	Anzeigen, wenn der zugewiesene Aktualparameter nicht mit dem vordefinierten Aktualparameter identisch ist	Der Parameter wird bei Bausteinaufrufen in KOP oder FUP versteckt, solange der vordefinierte Aktualparameter zugewiesen ist. Wenn jedoch ein anderer Aktualparameter zugewiesen wird, wird der Parameter sichtbar. Diese Option ist nur aktiv, wenn Sie die Option "Verstecken" angewählt und einen vordefinierten Aktualparameter definiert haben.

Gruppe	Eigenschaft	Beschreibung
Anwenderdefinierte Attribute	CFC_Configurable	Parametrierbar Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.
	CFC_ForTest	Für Test Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.
	CFC_Visible	Sichtbar Kennzeichnet, ob ein Parameter in CFC sichtbar ist.
	CFC_Interconnectable	Verschaltbar Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.
	CFC_EnableTagReadback	Rücklesbarkeit für Variablen aktivieren Kennzeichnet, ob ein Parameter für die Funktion "Plan rücklesen" in CFC relevant ist.
	CFC_EnumerationTexts	Enumerationstexte Ordnet einen Parameter einer Enumeration in CFC zu.
	CFC_EngineeringUnit	Engineering Einheit Ordnet einen Parameter einer Einheit in CFC zu.
	CFC_LowLimit	Untergrenze Definiert den unteren Grenzwert für den Parameter in CFC.
	CFC_HighLimit	Obergrenze Definiert den oberen Grenzwert für den Parameter in CFC.

Siehe auch

Remanenz von lokalen Variablen (Seite 7269)

Eigenschaften von lokalen Variablen und Konstanten ändern (Seite 7310)

Schlüsselwörter (Seite 93)

Parameter beim Bausteinaufruf verstecken (Seite 7284)

Vordefinierte Aktualparameter deklarieren (Seite 7285)

9.6.2 Eigenschaften von lokalen Variablen und Konstanten ändern

Eigenschaften eines Elements in der tabellarischen Bausteinschnittstelle bearbeiten

Um die Eigenschaften eines Elements in der Bausteinschnittstelle zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Selektieren Sie in der Tabelle das gewünschte Element.
3. Ändern Sie die Einträge in den Spalten.

Eigenschaften mehrerer Elemente in der tabellarischen Bausteinschnittstelle bearbeiten

Einige Eigenschaften können Sie auch für mehrere selektierte Elemente gleichzeitig setzen oder rücksetzen.

Um eine dieser Eigenschaften für mehrere Elemente umzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Halten Sie die <Strg>-Taste gedrückt.
3. Selektieren Sie in der gewünschten Spalte nacheinander die Tabellenzellen, deren Werte sie ändern möchten.
4. Wählen Sie im Kontextmenü die Befehle "<Eigenschaft> setzen" oder "<Eigenschaft> rücksetzen".

Eigenschaften im Eigenschaftsfenster bearbeiten

Um Eigenschaften einer einzelnen Variable oder Konstante zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable oder die Konstante.
   Die Eigenschaften des Elements werden im Inspektorfenster angezeigt.
2. Ändern Sie die Einträge im Inspektorfenster.

Variablen direkt im Programmiereditor umbenennen

Um eine oder mehrere Elemente umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Elemente im Programm.
2. Wählen Sie im Kontextmenü den Befehl "Variable umbenennen".
   Der Dialog "Variable umbenennen" wird geöffnet. Er zeigt eine Deklarationstabelle mit den selektierten Elementen.
3. Ändern Sie die Einträge in der Spalte "Name".
4. Bestätigen Sie die Eingabe mit der Schaltfläche "Ändern".

Datentyp oder Kommentar im Programmiereditor bearbeiten

Um den Datentyp oder den Kommentar einer Variablen im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Namen der Variablen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umverdrahten".
   Der Dialog "Variable umverdrahten" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Ändern Sie den Eintrag in den Spalten "Datentyp" oder "Kommentar".
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

Auswirkung im Programm

Bei einer Änderung des Namens, des Datentyps oder der Adresse einer Variablen oder Konstanten wird automatisch jede Verwendungsstelle der Variablen im Programm aktualisiert.

Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, wird das Programm möglicherweise inkonsistent. Die Inkonsistenzen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, werden die inkonsistenten Aufrufe rot markiert. Sie müssen die Inkonsistenzen dann manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Siehe auch

Übersicht über die Bausteinschnittstelle (Seite 7263)

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Remanenz von lokalen Variablen (Seite 7269)

Grundlagen zu Startwerten (Seite 7336)

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Bausteinschnittstelle aktualisieren (Seite 7314)

Tabelle der Bausteinschnittstelle bearbeiten (Seite 7287)

Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

9.6.3 Mehrsprachige Kommentare in der Bausteinschnittstelle bearbeiten

Sie können die Kommentare aus der Bausteinschnittstelle in allen Projektsprachen bearbeiten, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Beachten Sie, dass die Texte auch nach der Übersetzung die Länge von 32767 Unicode-Zeichen nicht überschreiten dürfen.

In der textuellen Bausteinschnittstelle sind mehrsprachige Kommentare nicht möglich.

Voraussetzung

- Sie haben mehrere Projektsprachen aktiviert.
- Die Bausteinschnittstelle ist geöffnet und enthält mindestens einen Kommentar.
- Der Baustein trägt keinen Know-how-Schutz.

Vorgehen

Um einen Kommentar in allen Projektsprachen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie den Kommentar an, dessen Übersetzung Sie bearbeiten wollen.
   Wenn Sie die Übersetzungen mehrerer Kommentare gleichzeitig bearbeiten wollen, markieren Sie alle gewünschten Kommentare.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das untergeordnete Register "Texte".
   Im Register "Texte" werden die markierten Kommentare in den aktiven Projektsprachen und, soweit vorhanden, die Übersetzungen der Kommentare angezeigt.
4. Sie können die Übersetzungen direkt in die Tabelle im Register "Texte" bearbeiten.
5. Um die Übersetzungen mit einem externen Editor zu bearbeiten, können Sie die angezeigten Texte über die Schaltflächen "Projekttexte exportieren/importieren" in das Format OOXML exportieren.

Hinweis

Alle Projekttexte in der globalen Tabelle "Projekttexte" bearbeiten

Sie können die Übersetzungen für die einzelnen Projektsprachen auch in der globale Tabelle "Projekttexte" bearbeiten. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte". Sie enthält alle übersetzbaren Texte des gesamten Projekts.

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

9.7 Bausteinschnittstelle aktualisieren

Einführung

Wenn Sie PLC-Datentypen oder Multiinstanzen ändern oder löschen, die in der Bausteinschnittstelle verwendet werden, wird die Schnittstelle inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie die Schnittstelle aktualisieren.

Sie haben zwei Möglichkeiten, die Bausteinschnittstelle zu aktualisieren:

- Explizites Aktualisieren der Bausteinschnittstelle.
  Die verwendeten PLC-Datentypen und Multiinstanzen werden aktualisiert. Die zum Baustein gehörenden Instanz-Datenbausteine werden dabei nicht implizit aktualisiert.
- Implizites Aktualisieren während des Übersetzens.
  Alle verwendeten PLC-Datentypen und Multiinstanzen sowie die zugehörigen Instanz-Datenbausteine werden aktualisiert.

Explizites Aktualisieren der Bausteinschnittstelle

Um die Bausteinschnittstelle explizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Wählen Sie im Kontextmenü den Befehl "Aktualisieren".

Implizites Aktualisieren während des Übersetzens

Um alle Verwendungen von PLC-Datentypen und Multiinstanzen sowie die Instanz-Datenbausteine während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Variablen basierend auf einem PLC-Datentyp deklarieren (Seite 7281)

Grundlagen zu Startwerten (Seite 7336)

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Eigenschaften von lokalen Variablen und Konstanten (Seite 7308)

Remanenz von lokalen Variablen (Seite 7269)

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Multiinstanzen deklarieren (Seite 7282)

Tabelle der Bausteinschnittstelle bearbeiten (Seite 7287)
Regeln zur Deklaration der Bausteinschnittstelle (Seite 7265)

9.8 Bausteinschnittstelle erweitern

Beschreibung

Um PLC-Programme, die bereits in Betrieb genommen wurden und fehlerfrei auf einer Anlage laufen, nachträglich zu ändern, bieten CPUs der Baureihe S7-1500 und die meisten CPUs der Baureihe S7-1200 ab V4 die Möglichkeit, die Schnittstellen von Funktionsbausteinen im laufenden Betrieb zu erweitern.

Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

Funktionsweise

Grundsätzlich enthält jeder Funktionsbaustein per Voreinstellung eine Speicherreserve. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass Sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die Werte bereits geladener Variablen nicht beeinflusst, der laufende Betrieb wird somit nicht gestört.

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie zusätzlich die Möglichkeit, das Speicherlayout einzelner oder mehrerer Bausteine in einem Schritt zu überarbeiten. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

Voraussetzungen

Die Funktion "Laden ohne Reinitialisierung" steht zur Verfügung, wenn folgende Voraussetzungen erfüllt sind:

- Das Projekt liegt im Format "TIA Portal V12" oder einer höheren Version vor.
- Sie arbeiten mit einer CPU, die das "Laden ohne Reinitialisierung" unterstützt.
- Die Bausteine wurden in KOP, FUP, AWL oder SCL erstellt.
- Die Bausteine wurden vom Anwender erstellt, d.h. es sind keine mitgelieferten Bausteine.
- Es handelt sich um Bausteine mit optimiertem Zugriff.

Grundsätzliche Schritte

Um die Schnittstelle eines Funktionsbausteins zu erweitern und den Baustein anschließend ohne Reinitialisierung zu laden, gehen Sie folgendermaßen vor:

1. Alle Bausteine haben per Voreinstellung eine Speicherreserve von 100 Byte. Bei Bedarf können Sie die Größe der Speicherreserve anpassen.
2. Aktivieren Sie die Speicherreserve.
3. Erweitern Sie die Bausteinschnittstelle.

4. Übersetzen Sie den Baustein.
5. Laden Sie den Baustein wie gewohnt in die CPU.

Detaillierte Beschreibungen der einzelnen Schritte finden Sie im Kapitel "Bausteine laden (S7-1200/1500) ".

Hinweis

Die Funktion "Laden ohne Reinitialisierung" steht Ihnen nur in CPUs der Baureihen S7-1500 und S7-1200 V4 in vollem Umfang zur Verfügung.

Grundsätzlich haben Sie jedoch in allen CPU-Familien folgende Möglichkeiten, die Schnittstelle von Funktionsbausteinen zu erweitern und neu deklarierte Variablen rückwirkungsfrei zu laden:

- Sie können im Abschnitt "Temp" neue Variablen hinzufügen und diese laden, ohne den Prozess zu beeinflussen.
- Sie können im Abschnitt "InOut" neue Variablen strukturierten Datentyps anlegen und diese laden, ohne den Prozess zu beeinflussen.

Datenbausteine programmieren 10

10.1 Grundlagen zur Programmierung von Datenbausteinen

Ein Datenbaustein (DB) dient dazu, Werte zu speichern, die während des Programmablaufs geschrieben werden.

Im Gegensatz zum Codebaustein enthält der Datenbaustein nur Variablendeklarationen. Er enthält keine Netzwerke oder Anweisungen. Die Variablendeklarationen definieren die Struktur des Datenbausteins.

Arten von Datenbausteinen

Es gibt zwei Arten von Datenbausteinen:

- Globale Datenbausteine
  Der globale Datenbaustein ist keinem Codebaustein zugeordnet. Sie können von jedem Codebaustein aus auf die Werte eines globalen Datenbausteins zugreifen. Ein globaler Datenbaustein enthält ausschließlich statische Variablen.
  Die Struktur des globalen Datenbausteins ist frei definierbar. Sie deklarieren die Datenelemente, die der globale Datenbaustein enthalten soll, in der Deklarationstabelle für Datenbausteine.
- Instanz-Datenbausteine
  Der Instanz-Datenbaustein ist einem Funktionsbaustein (FB) unmittelbar zugeordnet. Die Struktur eines Instanz-Datenbausteins ist nicht frei definierbar, sondern wird durch die Schnittstellendeklaration des Funktionsbausteins bestimmt. Der Instanz-Datenbaustein enthält genau die Bausteinparameter und Variablen, die dort deklariert wurden.
  Sie können jedoch im Instanz-Datenbaustein instanzspezifische Werte definieren, z. B. Startwerte für die deklarierten Variablen.

ARRAY-Datenbausteine (S7-1500)

ARRAY-Datenbausteine sind globale Datenbausteine, die aus einem ARRAY bestehen. Dieses ARRAY kann auf einem beliebigen Datentypen basieren. Es kann z. B. ein ARRAY eines PLC-Datentyps (UDT) sein. Neben dem ARRAY enthält der DB keine weiteren Elemente. ARRAY-Datenbausteine erleichtern durch ihre flache Struktur den Zugriff auf die ARRAY-Elemente sowie deren Übergabe an aufgerufene Bausteine.

ARRAY-Datenbausteine verwenden Sie, um ARRAYs indirekt zu adressieren: Sie können zur Laufzeit ermitteln, welcher ARRAY-DB gelesen oder geschrieben wird, und wie groß er ist.

Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben".

Hinweis

Remanenz von ARRAY-Datenbausteinen

ARRAY-Datenbausteine oder deren Komponenten können nicht remanent gesetzt werden.

Siehe auch:

Beispiel zur Verwendung von ARRAY-Datenbausteinen (Seite 44)

PLC-Datentypen als Vorlage für globale Datenbausteine

PLC-Datentypen können als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen. Sie erstellen nur einmal die Struktur als PLC-Datentyp und erzeugen anschließend die benötigten Datenbausteine durch Zuweisung des PLC-Datentyps.

Systemdatentypen als Vorlage für globale Datenbausteine

Systemdatentypen können ebenfalls als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen. Systemdatentypen haben bereits eine fest definierte Struktur. Sie fügen den Systemdatentyp nur einmal in das Programm ein und erzeugen anschließend weitere Datenbausteine mit derselben Struktur durch Zuweisung des Systemdatentyps.

Zugriffsarten

Es gibt zwei unterschiedliche Arten, auf Datenwerte in Datenbausteinen zuzugreifen:

- Datenbausteine mit optimiertem Zugriff (nur S7-1200)
  Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins. Auf Datenwerte in diesen Bausteinen greifen Sie über den symbolischen Namen zu.
  Für ARRAY-Datenbausteine ist das Attribut "Optimierter Bausteinzugriff" immer aktiviert.
- Datenbausteine mit Standardzugriff (alle CPU-Familien)
  Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins. Auf Datenwerte in diesen Bausteinen können Sie über den symbolischen Namen oder die Adresse zugreifen.
  Datenbausteine mit Standardzugriff sind an WORD-Grenzen ausgerichtet. Das bedeutet, sie belegen immer ein WORD (16 Bit) oder ein Vielfaches davon. Falls erforderlich werden dem Datenbaustein beim Übersetzen automatisch Variablen hinzugefügt, die den Baustein bis zur nächsten WORD-Grenze auffüllen.
  ARRAY-Datenbausteine mit Standardzugriff sind nicht möglich.

Remanenz der Datenwerte

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie die Datenwerte in einem remanenten Speicherbereich ablegen.

Siehe auch

Datenbausteine anlegen (Seite 7204)
Globale Datenbausteine (DB) (Seite 42)
Instanz-Datenbausteine (Seite 53)
Bausteine mit optimiertem Zugriff (Seite 55)

10.2 Aufbau der Deklarationstabelle für Datenbausteine

Aufbau der Deklarationstabelle für Datenbausteine

Das folgende Bild zeigt den Aufbau der Deklarationstabelle für Datenbausteine. Die Darstellung variiert je nach Baustein- und Zugriffsart.

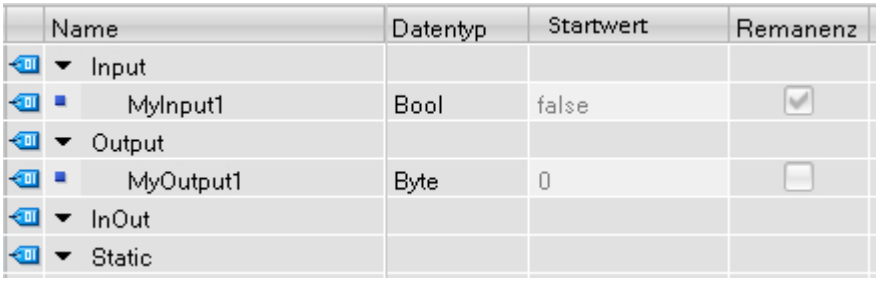

Name	Datentyp	Startwert	Remanenz
Input			
MyInput1	Bool	false	☑
Output			
MyOutput1	Byte	0	☐
InOut			
Static			

Darstellung instanzspezifischer Werte

In Instanz-Datenbausteinen können Sie bereits definierte Werte aus der Schnittstelle des zugeordneten Funktionsbausteins übernehmen oder instanzspezifische Startwerte definieren. Werte, die aus dem Funktionsbaustein übernommen sind, werden gegraut dargestellt. Die gegrauten Werte können Sie durch instanzspezifische Werte ersetzen. Werte, die bereits instanzspezifisch geändert wurden, sind nicht gegraut.

Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie.

Spalte	Erläuterung
	Symbol, das Sie anklicken können, um die Variable zu verschieben oder zu kopieren. Sie können sie z. B. per Drag & Drop in ein Programm ziehen und dort als Operand verwenden.
Name	Name der Variablen.
Datentyp	Datentyp der Variablen.
Offset	Relative Adresse der Variablen. Diese Spalte ist nur in Datenbausteinen mit Standardzugriff sichtbar. Hinweis: Viele Anweisungen aus den SIMATIC-Systembibliotheken haben die Eigenschaft "optimierter Bausteinzugriff" und belegen somit keine festen Speicheradressen. Für diese Anweisungen wird kein Offset angezeigt, auch wenn Sie sie als Multiinstanz in einem Baustein mit Standardzugriff verwenden.
Defaultwert	Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde. Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt.

Spalte	Erläuterung
Startwert	Wert, den die Variable beim Anlauf einnehmen soll. Beim Anlegen des Datenbausteins werden die in einem Codebaustein definierten Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie hier mit instanzspezifischen Startwerten ersetzen. Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt.
Beobachtungswert	Aktueller Datenwert in der CPU. Die Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und Sie die Schaltfläche "Beobachten" anwählen.
Momentaufnahme	Zeigt Werte an, die aus dem Gerät geladen wurden.
Remanenz	Kennzeichnet die Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.
Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.
Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
Einstellwert	Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Nach der Inbetriebnahme können die Werte dieser Variablen als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.
Überwachung	Zeigt an, ob für die Variable eine Überwachung zur Prozessdiagnose angelegt wurde.
Kommentar	Kommentar zur Dokumentation der Variablen.

Siehe auch

Eigenschaften von Variablen in Datenbausteinen bearbeiten (Seite 7341)

Einstellwerte während der Inbetriebnahme nutzen (Seite 7365)

Datenbausteine anlegen (Seite 7204)

Datenstruktur von globalen Datenbausteinen erstellen (Seite 7330)

Grundlagen zu Startwerten (Seite 7336)

Startwerte definieren (Seite 7336)

Remanenz einstellen (Seite 7338)

10.3 Datenbausteine anlegen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen Datenbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Wählen Sie den Typ des Datenbausteins aus. Folgende Möglichkeiten stehen zur Verfügung:
   - Um einen globalen Datenbaustein anzulegen, wählen Sie den Listeneintrag "Global-DB".
   - Um einen ARRAY-Datenbaustein anzulegen, wählen Sie den Listeneintrag "ARRAY-DB".
   - Um einen Instanz-Datenbaustein anzulegen, wählen Sie aus der Liste den Funktionsbaustein aus, dem Sie den Instanz-Datenbaustein zuordnen möchten. In der Liste werden nur Funktionsbausteine angeboten, die zuvor für die CPU erstellt wurden.
   - Um einen Datenbaustein basierend auf einem PLC-Datentyp anzulegen, wählen Sie aus der Liste den PLC-Datentyp aus. In der Liste werden nur PLC-Datentypen angeboten, die zuvor für die CPU erstellt wurden.
   - Um einen Datenbaustein basierend auf einem Systemdatentyp anzulegen, wählen Sie aus der Liste den Systemdatentyp aus. In der Liste werden nur Systemdatentypen angeboten, die zuvor in Programmbausteinen der CPU eingefügt wurden.
4. Geben Sie einen Namen für den Datenbaustein ein.
5. Geben Sie die Eigenschaften des neuen Datenbausteins ein.
6. Wenn Sie als "Typ" einen ARRAY-DB ausgewählt haben, geben Sie den ARRAY-Datentyp und die obere ARRAY-Grenze ein.
   Die obere ARRAY-Grenze können Sie jederzeit im Eigenschaftsfenster des erzeugten Bausteins ändern. Der ARRAY-Datentyp ist nachträglich nicht mehr änderbar.
7. Wenn Sie als "Typ" einen Baustein angewählt haben, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
8. Um weitere Eigenschaften des neuen Datenbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
9. Geben Sie alle gewünschten Eigenschaften ein.
10. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
11. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Der neue Datenbaustein wird erzeugt. Sie finden den Datenbaustein in der Projektnavigation im Ordner "Programmbausteine".

Hinweis

Remanenz von ARRAY-Datenbausteinen

ARRAY-Datenbausteine oder deren Komponenten können nicht remanent gesetzt werden.

Siehe auch

Globale ARRAY-Datenbausteine (DB) (Seite 43)

Grundlagen zur Programmierung von Datenbausteinen (Seite 7319)

Globale Datenbausteine (DB) (Seite 42)

Instanz-Datenbausteine (Seite 53)

Systemdatentypen (Seite 364)

Bausteine aus Bibliotheken verwenden (Seite 7205)

10.4 Datenbausteine aktualisieren

Einführung

Änderungen in der Schnittstelle eines Funktionsbausteins oder eines PLC-Datentyps können dazu führen, dass zugehörige Datenbausteine inkonsistent werden. Diese Inkonsistenzen werden in der Deklarationstabelle und an der Aufrufstelle des Bausteins rot markiert. Um die Inkonsistenzen zu beheben, müssen Sie die Datenbausteine aktualisieren.

Sie haben drei Möglichkeiten, Datenbausteine zu aktualisieren:

- Explizites Aktualisieren in der Deklarationstabelle für Datenbausteine.
  Der Datenbaustein wird aktualisiert. Änderungen aus der Schnittstelle des zugeordneten Funktionsbausteins und Änderungen an verwendeten PLC-Datentypen werden übernommen.
- Explizites Aktualisieren im Programmiereditor.
  Die Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden auch die zugehörigen Instanz-Datenbausteine angepasst.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen und die zugehörigen Instanz-Datenbausteine werden aktualisiert.

Explizites Aktualisieren in der Deklarationstabelle für Datenbausteine

Um einen einzelnen Datenbaustein explizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Wählen Sie im Kontextmenü "Schnittstelle aktualisieren".

Explizites Aktualisieren im Programmiereditor

Um in einem Baustein alle Bausteinaufrufe oder einen bestimmten Aufruf zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf die Anweisung mit dem Bausteinaufruf.
3. Wählen Sie im Kontextmenü den Befehl "Aktualisieren".
4. Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.
5. Wenn Sie den Bausteinaufruf aktualisieren möchten, klicken Sie auf "OK". Um die Aktualisierung abzubrechen, klicken Sie auf "Abbrechen".

Implizites Aktualisieren während des Übersetzens

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen sowie die Instanz-Datenbausteine während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen).

Siehe auch

Eigenschaften von Variablen in Instanz-Datenbausteinen ändern (Seite 7344)

10.5 Datenbausteine erweitern

Beschreibung

Um PLC-Programme, die bereits in Betrieb genommen wurden und fehlerfrei auf einer Anlage laufen, nachträglich zu ändern, bieten CPUs der Baureihe S7-1500 und die meisten CPUs der Baureihe S7-1200 ab V4 die Möglichkeit, globale Datenbausteine im laufenden Betrieb zu erweitern.

Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

Funktionsweise

Grundsätzlich enthält jeder Datenbaustein per Voreinstellung eine Speicherreserve. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die Werte bereits geladener Variablen nicht beeinflusst.

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie zusätzlich die Möglichkeit, das Speicherlayout einzelner oder mehrerer Bausteine in einem Schritt zu überarbeiten. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

Voraussetzungen

Die Funktion "Laden ohne Reinitialisierung" steht zur Verfügung, wenn folgende Voraussetzungen erfüllt sind:

- Das Projekt liegt im Format "TIA Portal V12" oder einer höheren Version vor.
- Sie arbeiten mit einer CPU , die das "Laden ohne Reinitialisierung" unterstützt.
- Die Bausteine wurden in KOP, FUP, AWL oder SCL erstellt.
- Die Bausteine wurden vom Anwender erstellt, d.h., es sind keine mitgelieferten Bausteine.
- Es handelt sich um Bausteine mit optimiertem Zugriff.

Grundsätzliche Schritte

Um einen Datenbaustein zu erweitern und den Baustein anschließend ohne Reinitialisierung zu laden, führen Sie folgende Schritte aus.

1. Alle Bausteine haben per Voreinstellung eine Speicherreserve von 100 Byte. Bei Bedarf können Sie die Größe der Speicherreserve anpassen.
2. Aktivieren Sie die Speicherreserve.
3. Erweitern Sie die Bausteinschnittstelle.

4. Übersetzen Sie den Baustein.
5. Laden Sie den Baustein wie gewohnt in die CPU.

Verweis

Detaillierte Beschreibungen der einzelnen Schritte finden Sie im Kapitel "Bausteine laden (S7-1200/1500) ".

10.6 Datenstruktur von globalen Datenbausteinen erstellen

10.6.1 Variablen von elementarem Datentyp deklarieren

Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

Hinweis

Die Struktur von Instanz-Datenbausteinen und von Datenbausteinen, die auf einem PLC-Datentyp basieren, können Sie nicht direkt ändern, da die Strukturen dieser Bausteine durch den Funktionsbaustein oder den PLC-Datentyp definiert werden.

Der Typ des Datenbausteins ist in den Bausteineigenschaften eingetragen.

Vorgehen

Um eine Variable elementaren Datentyps zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Klicken Sie in der Spalte "Datentyp" auf die Schaltfläche für die Datentypauswahl. Eine Liste der zulässigen Datentypen wird geöffnet.
3. Wählen Sie den gewünschten Datentyp aus.
4. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.
5. Wiederholen Sie die Schritte 1 bis 4 für alle zu deklarierenden Variablen.

Siehe auch

Bausteineigenschaften anzeigen und bearbeiten (Seite 7222)

Variablen vom Datentyp ARRAY deklarieren (Seite 7331)

Variablen vom Datentyp STRUCT deklarieren (Seite 7332)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

Startwerte definieren (Seite 7336)

Remanenz einstellen (Seite 7338)

Online- und Diagnosefunktionen für Datenbausteine (Seite 7352)

10.6.2 Variablen vom Datentyp ARRAY deklarieren

Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den Datentyp "Array" ein. Sie werden dabei von der Autovervollständigung unterstützt.
   Der Dialog "Array" wird geöffnet.
3. Geben Sie im Eingabefeld "Datentyp" den Datentyp der Array-Elemente an.
4. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.
   Beispiel für ein eindimensionales ARRAY:
   `ARRAY [0..3] of Bool`
   Beispiel für ein dreidimensionales ARRAY:
   `ARRAY[0..3, 0..15, 0..33] of Bool`
   Beispiel für ein eindimensionales ARRAY mit lokalen Konstanten als ARRAY-Grenzen
   `ARRAY[#My_local_const1..#My_local_const2] of Bool`
   Beispiel für ein eindimensionales ARRAY mit globalen Konstanten als ARRAY-Grenzen
   `ARRAY["My_global_const1".."My_global_const1"] of Bool`
5. Bestätigen Sie die Eingabe.
6. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.

Die Variable wird angelegt, bleibt jedoch zugeklappt. Um das ARRAY aufzuklappen, klicken Sie auf das Dreieck vor der Variablen. Beachten Sie, dass ARRAYs mit mehr als 10000 Elementen aus Gründen der Übersichtlichkeit nicht aufklappbar sind.

Startwerte von ARRAY-Elementen eingeben

Um die einzelnen Elemente eines ARRAYs mit Startwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein.

ARRAYs kommentieren

Um einen Kommentar für das gesamte ARRAY anzugeben, tragen Sie diesen in die oberste Zeile der ARRAY-Deklaration ein. Der Kommentar wird als Voreinstellung für alle unterlagerten ARRAY-Elemente verwendet.

Um die einzelnen Elemente eines ARRAYs mit spezifischen Kommentaren zu versehen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Kommentar" ein.

ARRAYs im erweiterten Modus anzeigen

In der Anzeige "Erweiterter Modus" werden alle ARRAYs per Voreinstellung aufgeklappt dargestellt.

Um den erweiterten Modus einzuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Erweiterter Modus" in der Funktionsleiste.

Hinweis

Einschränkung in der Anzeige "Erweiterter Modus"

Abhängig der Größe des Arbeitsspeichers Ihres Computers gelten für den erweiterten Modus folgende Maximalgrenzen:

- Arbeitsspeicher < 8 GB: Es werden maximal 10.000 Zeilen angezeigt.
- Arbeitsspeicher >= 8 GB: Es werden maximal 40.000 Zeilen angezeigt.

Siehe auch

Startwerte definieren (Seite 7336)

Remanenz einstellen (Seite 7338)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

ARRAY (Seite 296)

Autovervollständigung verwenden (Seite 7147)

10.6.3 Variablen vom Datentyp STRUCT deklarieren

Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
   Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
3. Fügen Sie in der leeren Zeile das erste Strukturelement ein.
   Nach dem Element wird erneut eine leere Zeile eingefügt.
4. Wählen Sie einen Datentyp für das Strukturelement aus.
5. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.
6. Wiederholen Sie Schritt 4 bis 7 für alle weiteren Strukturelemente.
   Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
7. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.

Startwerte von Strukturelementen eingeben

Um die einzelnen Elemente einer Struktur mit Startwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp STRUCT.
   Die Struktur wird aufgeklappt und die einzelnen Strukturelemente werden in eigenen Zeilen dargestellt.
   Beachten Sie, dass sehr große Strukturen aus Gründen der Übersichtlichkeit nicht aufklappbar sind.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein.

Hinweis

S7-1500: Maximal 252 Strukturen in einem Datenbaustein

In einem Datenbaustein sind maximal 252 Strukturen zulässig. Wenn Sie weitere Strukturen benötigen, müssen Sie Ihr Programm umstrukturieren. Beispielsweise können Sie die Strukturen in mehreren globalen Datenbausteinen anlegen.

Siehe auch

Grundlagen zu STRUCT (Seite 291)
Startwerte definieren (Seite 7336)
Remanenz einstellen (Seite 7338)
Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)
Autovervollständigung verwenden (Seite 7147)

10.6.4 Variablen vom Datentyp STRING deklarieren

Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp STRING zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Schnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Tragen Sie Sie in der Spalte "Datentyp" "STRING" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
4. Optional: Geben Sie die Maximallänge der Zeichenkette mithilfe von eckigen Klammern nach dem Schlüsselwort STRING an. Wenn Sie keine Maximallänge angeben, hat der STRING die Standardlänge von 254 Zeichen.
   Beispiel für einen STRING der Maximallänge 4:
   `STRING[4]`
   Beispiel für einen STRING, dessen Maximallänge durch eine lokale Konstante definiert wird:
   `STRING[#My_local_const1]`
   Beispiel für einen STRING, dessen Maximallänge durch eine globale Konstante definiert wird:
   `STRING["My_global_const1"]`

Ergebnis

Die Variable vom Datentyp STRING wird angelegt.

10.6.5 Variablen basierend auf einem PLC-Datentyp deklarieren

Voraussetzung

- Ein globaler Datenbaustein ist geöffnet.
- In der aktuellen CPU ist ein PLC-Datentyp deklariert.

Vorgehen

Um eine Variable basierend auf einem PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
3. Die Variable wird angelegt.
4. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Tabelle angezeigt werden.
   Wenn Sie bei der Dekaration des PLC-Datentyps bereits Defaultwerte oder Kommentare für die Variablen innerhalb eines PLC-Datentyps definiert haben, werden diese grau angezeigt. Sie können diese Werte in der Bausteinschnittstelle ändern.
   Die geänderten Werte werden schwarz dargestellt und gelten nur für die spezifische Verwendungsstelle.

Hinweis

Wenn Sie PLC-Datentypen ändern oder löschen, die im Datenbaustein verwendet werden, wird der Datenbaustein inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie das Programm neu übersetzen.

Siehe auch

Übersicht über die Bausteinschnittstelle (Seite 7263)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

Autovervollständigung verwenden (Seite 7147)

10.7 Startwerte definieren

10.7.1 Grundlagen zu Startwerten

Definition "Startwert"

Der Startwert einer Variablen ist ein von Ihnen definierter Wert, den die Variable nach einem Anlauf der CPU einnehmen soll.

Eine Sonderstellung nehmen die remanenten Variablen ein. Ihre Werte nehmen nur nach einem "Kaltstart" den definierten Startwert ein. Nach einem "Neustart (Warmstart)" bleiben ihre Werte erhalten und werden nicht auf den Startwert zurückgesetzt.

Definition "Defaultwert"

Die Struktur von Datenbausteinen kann von übergeordneten Elementen abgeleitet sein:

- Ein Instanz-Datenbaustein basiert z. B. auf der Schnittstelle eines übergeordneten Codebausteins.
- Ein globaler Datenbaustein kann auf einem vordefinierten PLC-Datentyp basieren.

In diesem Fall können Sie im übergeordneten Element einen Defaultwert für jede Variable definieren. Beim Anlegen des Datenbausteins werden diese Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie anschließend im Datenbaustein mit instanzspezifischen Startwerten ersetzen.

Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt.

Siehe auch

Startwerte definieren (Seite 7337)

Aufbau der Deklarationstabelle für Datenbausteine (Seite 7322)

Lokale Variablen und Konstanten in der Bausteinschnittstelle deklarieren (Seite 7274)

Werte aus dem Online-Programm als Startwerte übernehmen (Seite 7368)

10.7.2 Startwerte definieren

Startwerte definieren

Um Startwerte für die Variablen eines Datenbausteins festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
   Die Spalte "Defaultwert" zeigt die Defaultwerte, die für die Variablen in der Schnittstelle eines übergeordneten Codebausteins oder in einem PLC-Datentyp definiert wurden.
2. Um alle Elemente strukturierter Datentypen einzublenden, klicken Sie auf die Schaltfläche "Erweiterter Modus".
3. Tragen Sie in die Spalte "Startwert" die gewünschten Startwerte ein. Der Wert muss dem Datentyp der Variablen entsprechen und darf den Wertebereich des Datentyps nicht überschreiten.
   Die Startwerte sind definiert. Die Variable nimmt beim Anlauf den definierten Wert ein, sofern sie nicht als remanent deklariert wurde.

Eine Variable auf den Defaultwert zurücksetzen

Um eine Variable, für die Sie einen Startwert definiert haben, wieder auf den Defaultwert zurückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen geänderten Wert.
2. Löschen Sie den Wert.
   Der Defaultwert wird eingetragen. Er wird gegraut dargestellt.

Alle Variablen auf den Defaultwert zurücksetzen

Um alle Variablen, für die Sie einen Startwert definiert haben, wieder auf den Defaultwert zurückzusetzen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Funktionsleiste das Symbol "Startwerte zurücksetzen".
   Die Defaultwerte werden in die Spalte "Startwert" übertragen. Schreibgeschützte Startwerte werden nicht überschrieben.

Siehe auch

Grundlagen zu Startwerten (Seite 7336)

Werte aus dem Online-Programm als Startwerte übernehmen (Seite 7368)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

10.8 Remanenz einstellen

10.8.1 Remanenz von Variablen in Datenbausteinen

Remanenzverhalten

Um Datenverlust bei Spannungsausfall zu vermeiden, können Sie Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Die Möglichkeiten zur Einstellung der Remanenz sind abhängig von der Art des Datenbausteins und der eingestellten Art des Bausteinzugriffs.

Siehe auch

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 7338)

Remanenz in einem globalen Datenbaustein einstellen (Seite 7339)

10.8.2 Remanenz in einem Instanz-Datenbaustein einstellen

Einführung

In einem Instanz-Datenbaustein hängt die Editierbarkeit des Remanenzverhaltens von der Zugriffsart des übergeordneten Funktionsbausteins ab:

- Funktionsbaustein mit Standardzugriff
  Sie können die Instanzdaten gemeinsam als remanent oder nicht-remanent definieren. Individuelle Remanenzeinstellungen für einzelne Variablen sind nicht möglich.
- Funktionsbaustein mit optimiertem Zugriff
  Sie können im Instanz-Datenbaustein die Remanenzeinstellungen der Variablen definieren, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind. Auch bei diesen Variablen können Sie das Remanenzverhalten nicht für jede Variable individuell einstellen. Die Remanenzeinstellung wirkt sich auf alle Variablen aus, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind.

Remanenz bei Standardzugriff einstellen

Um in Datenbausteinen mit Standardzugriff die Remanenz aller Variablen zentral einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen werden als remanent definiert.
3. Um die Remanenzeinstellung für alle Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen werden als nicht remanent definiert.

Remanenz bei optimiertem Zugriff einstellen

Um in Datenbausteinen mit optimiertem Zugriff das Remanenzverhalten der Variablen, die mit "Im IDB setzen" markiert sind, einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind, werden als remanent definiert.
3. Um die Remanenzeinstellung für die Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind, werden als nicht remanent definiert.

Siehe auch

Grundlagen zum Bausteinzugriff (Seite 55)

Remanenz von Variablen in Datenbausteinen (Seite 7338)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

10.8.3 Remanenz in einem globalen Datenbaustein einstellen

Einführung

In einem globalen Datenbaustein hängt die Editierbarkeit des Remanenzverhaltens von der Zugriffsart ab:

- Globaler Datenbaustein mit Standardzugriff
  Sie können die Daten gemeinsam als remanent oder nicht-remanent definieren.
  Individuelle Remanenzeinstellungen für einzelne Variablen sind nicht möglich.
- Globaler Datenbaustein mit optimiertem Zugriff
  Sie können die Remanenzeinstellungen der Variablen individuell definieren. Bei Variablen mit strukturierten Datentypen werden die Remanenzeinstellungen auf alle Variablenelemente übertragen.

Hinweis

Remanenz von ARRAY-Datenbausteinen

ARRAY-Datenbausteine oder deren Komponenten können nicht remanent gesetzt werden.

Remanenz bei Standardzugriff einstellen

Um in Datenbausteinen mit Standardzugriff die Remanenz aller Variablen zentral einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen werden als remanent definiert.
3. Um die Remanenzeinstellung für alle Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Alle Variablen werden als nicht remanent definiert.

Remanenz bei optimiertem Zugriff einstellen

Um in Datenbausteinen mit optimiertem Zugriff die Remanenz der Variablen individuell einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" für die Variable, für die Sie ein Remanenzverhalten einstellen wollen.
   Die selektierte Variable wird als remanent definiert.
3. Um die Remanenzeinstellung für die Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.
   Die selektierten Variablen werden als nicht remanent definiert.

Siehe auch

Grundlagen zum Bausteinzugriff (Seite 55)

Remanenz von Variablen in Datenbausteinen (Seite 7338)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

10.9 Eigenschaften von Variablen in Datenbausteinen bearbeiten

10.9.1 Eigenschaften von Variablen in Datenbausteinen

Eigenschaften

Die Eigenschaften einer Variable werden im Inspektorfenster angezeigt, wenn Sie die Variable in der Deklarationstabelle oder an einer Verwendungsstelle im Programmiereditor selektieren.

Hinweis

Eigenschaften von ARRAY-Elementen

Wenn Sie in Ihrem Programm ARRAY-Zugriffe mit variablem Index in der Form `#Program_Alarm_1[#myIndex]` verwenden, steht zum Zeitpunkt der Programmierung noch nicht fest, welches konkrete ARRAY-Element zur Laufzeit verwendet wird. Wenn Sie einen solchen ARRAY-Zugriff im Programmiereditor selektieren, werden deshalb im Inspektorfenster immer die Eigenschaften des ersten ARRAY-Elements angezeigt.

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von Variablen in Datenbausteinen:

Gruppe	Eigenschaft	Beschreibung
Allgemein	Name	Name der Variablen.
	Datentyp	Datentyp der Variablen.
	Defaultwert	Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde. Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt.
	Startwert	Wert, den die Variable beim Anlauf der CPU einnehmen soll. Beim Anlegen des Datenbausteins werden die in einem Codebaustein definierten Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie hier mit instanzspezifischen Startwerten ersetzen. Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt.
	Kommentar	Kommentar zur Variablen.

Gruppe	Eigenschaft	Beschreibung
Attribute	Remanenz	Kennzeichnet die Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten. Dieses Attribut ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff verfügbar.
	Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
	Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
	Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI /OPC UA zur Laufzeit auf diese Variable zugreifen kann. Beachten Sie jedoch, dass Sie mit dem Attribut "Erreichbar aus HMI / OPC UA" keinen allgemeinen Zugriffsschutz für die Variable realisieren können. Der lesende oder schreibende Zugriff durch andere Applikationen kann möglich sein, auch wenn das Attribut nicht aktiviert ist.
	Sichtbarkeit bei Bausteinaufrufen in KOP/FUP	Anzeigen: Der Parameter ist bei Bausteinaufrufen in KOP oder FUP immer sichtbar.
		Verstecken: Der Parameter wird bei Bausteinaufrufen in KOP oder FUP immer versteckt. Die Aufrufbox verfügt dann über einen kleinen Pfeil am unteren Rand. Durch Klicken auf diesen Pfeil können Sie den Parameter einblenden und beschalten.
		Verstecken, wenn kein Parameter zugewiesen ist: Der Parameter wird bei Bausteinaufrufen in KOP oder FUP versteckt, solange er nicht beschaltet ist. Wenn sie einen Aktualparameter zuweisen, wird der Parameter an der Aufrufbox angezeigt.
	Vordefinierter Aktualparameter	Definiert einen Parameter, der beim Bausteinaufruf als Aktualparameter verwendet werden soll.
	Anzeigen, wenn der zugewiesene Aktualparameter nicht mit dem vordefinierten Aktualparameter identisch ist	Der Parameter wird bei Bausteinaufrufen in KOP oder FUP versteckt, solange der vordefinierte Aktualparameter zugewiesen ist. Wenn jedoch ein anderer Aktualparameter zugewiesen wird, wird der Parameter sichtbar. Diese Option ist nur aktiv, wenn Sie die Option "Verstecken" angewählt und einen vordefinierten Aktualparameter definiert haben.

Gruppe	Eigenschaft	Beschreibung
Anwenderdefinierte Attribute	CFC_Visible	Sichtbar Kennzeichnet, ob ein Parameter im CFC sichtbar ist.
	CFC_Configurable	Parametrierbar Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.
	CFC_ForTest	Für Test Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.
	CFC_Interconnectable	Verschaltbar Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.
	CFC_EnableTagReadback	Rücklesbarkeit für Variablen aktivieren Kennzeichnet, ob ein Parameter für die Funktion "Plan rücklesen" in CFC relevant ist.
	CFC_EnumerationTexts	Enumerationstexte Ordnet einen Parameter einer Enumeration in CFC zu.
	CFC_EngineeringUnit	Engineering Einheit Ordnet einen Parameter einer Einheit in CFC zu.
	CFC_LowLimit	Untergrenze Definiert den unteren Grenzwert für den Parameter in CFC.
	CFC_HighLimit	ObergrenzeDefiniert den oberen Grenzwert für den Parameter in CFC.

Siehe auch

Eigenschaften von Variablen in Instanz-Datenbausteinen ändern (Seite 7344)

Eigenschaften von Variablen in globalen Datenbausteinen ändern (Seite 7345)

Remanenz einstellen (Seite 7338)

Startwerte definieren (Seite 7336)

10.9.2 Eigenschaften von Variablen in Instanz-Datenbausteinen ändern

Instanzspezifische Variableneigenschaften

Für die Definition der Variableneigenschaften gibt zwei Möglichkeiten:

- Die Variableneigenschaften werden aus der Schnittstelle des zugeordneten Funktionsbausteins übernommen.
  Eigenschaften, die aus dem Funktionsbaustein übernommen sind, werden in den Spalten der Deklarationstabelle gegraut dargestellt. Die Eigenschaften "Name" und "Datentyp" werden immer übernommen.
- Sie definieren instanzspezifische Eigenschaften.
  Einige Eigenschaften können Sie instanzspezifisch ändern. Ein änderbarer Wert ist z. B. der Kommentar. Eigenschaften die instanzspezifisch geändert wurden, sind in den Spalten der Deklarationstabelle nicht gegraut. Die instanzspezifischen Änderungen bleiben erhalten, auch wenn die Schnittstelle des übergeordneten Funktionsbausteins geändert und der Instanz-Datenbaustein anschließend aktualisiert wird.

Eigenschaften eines Elements in der Deklarationstabelle bearbeiten

Um die Eigenschaften eines Elements zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Selektieren Sie in der Tabelle das gewünschte Element.
3. Ändern Sie die Einträge in den Spalten.

Eigenschaften mehrerer Elemente in der Deklarationstabelle bearbeiten

Einige Eigenschaften können Sie auch für mehrere selektierte Elemente gleichzeitig setzen oder rücksetzen.

Um eine dieser Eigenschaften für mehrere Elemente umzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Halten Sie die STRG-Taste gedrückt.
3. Selektieren Sie in der gewünschten Spalte nacheinander die Tabellenzellen, deren Werte sie ändern möchten.
4. Wählen Sie im Kontextmenü die Befehle "<Eigenschaft> setzen" oder "<Eigenschaft> rücksetzen".

Eigenschaften im Eigenschaftsfenster bearbeiten

Um die Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Das Eigenschaftsfenster wird geöffnet. Es zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".

3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

Einzelne Eigenschaften auf den Vorbelegungswert zurücksetzen

Um einzelne Variableneigenschaften auf den Wert zurückzusetzen, der im Funktionsbaustein als Vorbelegungswert definiert wurde, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen instanzspezifisch geänderten Wert.
2. Löschen Sie den Wert.
   Der instanzspezifische Wert wird gelöscht und der Vorbelegungswert aus der Schnittstelle des Funktionsbausteins wird eingetragen. Der Vorbelegungswert wird gegraut dargestellt.

Siehe auch

Datenbausteine aktualisieren (Seite 7326)

Eigenschaften von Variablen in Datenbausteinen (Seite 7341)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

10.9.3 Eigenschaften von Variablen in globalen Datenbausteinen ändern

Einführung

Für die Definition der Variableneigenschaften gibt zwei Möglichkeiten:

- Die Variableneigenschaften werden aus dem PLC-Datentyp übernommen.
  Eigenschaften, die aus dem PLC-Datentyp übernommen sind, werden in den Spalten der Deklarationstabelle gegraut dargestellt. Die Eigenschaften "Name" und "Datentyp" werden immer übernommen.
- Sie definieren spezifische Eigenschaften.
  Einige Eigenschaften können Sie im globalen Datenbaustein ändern. Änderbare Werte sind z. B. der "Kommentar" oder "Sichtbar in HMI". Eigenschaften die geändert wurden, sind in den Spalten der Deklarationstabelle nicht gegraut. Die Änderungen bleiben erhalten, auch wenn der PLC-Datentyp geändert und der globale Datenbaustein anschließend aktualisiert wird.

Eigenschaften eines Elements in der Deklarationstabelle bearbeiten

Um die Eigenschaften eines Elements zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Selektieren Sie in der Tabelle das gewünschte Element.
3. Ändern Sie die Einträge in den Spalten.

Eigenschaften mehrerer Elemente in der Deklarationstabelle bearbeiten

Einige Eigenschaften können Sie auch für mehrere selektierte Elemente gleichzeitig setzen oder rücksetzen.

Um eine dieser Eigenschaften für mehrere Elemente umzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Halten Sie die STRG-Taste gedrückt.
3. Selektieren Sie in der gewünschten Spalte nacheinander die Tabellenzellen, deren Werte sie ändern möchten.
4. Wählen Sie im Kontextmenü die Befehle "<Eigenschaft> setzen" oder "<Eigenschaft> rücksetzen".

Eigenschaften im Eigenschaftsfenster bearbeiten

Um die Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Das Eigenschaftsfenster wird geöffnet. Es zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".
3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

Einzelne Eigenschaften auf den Vorbelegungswert zurücksetzen

Um einzelne Variableneigenschaften auf den Wert zurückzusetzen, der im PLC-Datenyp als Vorbelegungswert definiert wurde, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen geänderten Wert.
2. Löschen Sie den Wert.
   Der Vorbelegungswert aus dem PLC-Datentyp wird eintragen. Der Vorbelegungswert wird gegraut dargestellt.

Siehe auch

Eigenschaften von Variablen in Datenbausteinen (Seite 7341)

Deklarationstabelle für Datenbausteine bearbeiten (Seite 7348)

10.9.4 Mehrsprachige Kommentare im Datenbaustein bearbeiten

Sie können die Kommentare aus einem Datenbaustein in allen Projektsprachen bearbeiten, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Beachten Sie, dass die Texte auch nach der Übersetzung die Länge von 32767 Unicode-Zeichen nicht überschreiten dürfen.

Voraussetzung

- Sie haben mehrere Projektsprachen aktiviert.
- Der Datenbaustein ist geöffnet und enthält mindestens einen Kommentar.
- Der Baustein trägt keinen Know-how-Schutz.

Vorgehen

Um einen Kommentar in allen Projektsprachen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie den Kommentar an, dessen Übersetzung Sie bearbeiten wollen.
   Wenn Sie die Übersetzungen mehrerer Kommentare gleichzeitig bearbeiten wollen, markieren Sie alle gewünschten Kommentare.
   In Instanzdatenbausteinen können Sie nur Kommentare anwählen, die Sie instanzspezifisch geändert haben. Diese Kommentare können Sie daran erkennen, dass sie nicht gegraut dargestellt sind. Grau dargestellte Kommentare können Sie nur im zugehörigen Funktionsbaustein anwählen und bearbeiten.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das untergeordnete Register "Texte".
   Im Register "Texte" werden die markierten Kommentare in den aktiven Projektsprachen und, soweit vorhanden, die Übersetzungen der Kommentare angezeigt.
4. Sie können die Übersetzungen direkt in die Tabelle im Register "Texte" bearbeiten.
5. Um die Übersetzungen mit einem externen Editor zu bearbeiten, können Sie die angezeigten Texte über die Schaltflächen "Projekttexte exportieren/importieren" in das Format OOXML exportieren.

Hinweis

Alle Projekttexte in der globalen Tabelle "Projekttexte" bearbeiten

Sie können die Übersetzungen für die einzelnen Projektsprachen auch in der globale Tabelle "Projekttexte" bearbeiten. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte". Sie enthält alle übersetzbaren Texte des gesamten Projekts.

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

10.10 Deklarationstabelle für Datenbausteine bearbeiten

10.10.1 Tabellenzeilen einfügen

Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

10.10.2 Tabellenzeilen hinzufügen

Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

10.10.3 Variablen löschen

Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

Vorgehen

Um eine Variable zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variablen. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Die Struktur von Instanz-Datenbausteinen oder von globalen Datenbausteinen, die auf einem PLC-Datentyp basieren, können Sie nicht direkt ändern, da die Strukturen dieser Bausteine im übergeordneten Objekt definiert werden.

Der Typ des Datenbausteins ist in den Bausteineigenschaften eingetragen.

Siehe auch: Bausteineigenschaften anzeigen und bearbeiten (Seite 7222)

10.10.4 Aufeinander folgende Zellen automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

Voraussetzung

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.
   Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.
   Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

10.10.5 Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
5. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.

10.10.6 Variablen mit externen Editoren bearbeiten

Um einzelne Variablen in externen Tabelleneditoren, z. B. in Excel, zu bearbeiten, können Sie sie per Copy & Paste exportieren oder importieren.

Hinweis

Einschränkung beim Anzeigen und Kopieren von ARRAY-Elementen

Abhängig der Größe des Arbeitsspeichers Ihres Computers gelten bei der Anzeige und beim Kopieren von ARRAYs aus dem Datenbaustein-Editor in eine Excel-Tabelle folgende Maximalgrenzen:

- Arbeitsspeicher < 8 GB: Es werden maximal 10.000 Elemente angezeigt und kopiert.
- Arbeitsspeicher >= 8 GB: Es werden maximal 40.000 Elemente angezeigt und kopiert.

Voraussetzung

Der Datenbaustein und ein externer Editor sind geöffnet.

Vorgehen

Um einzelne Variablen per Drag & Drop zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Um alle Elemente strukturierter Datentypen einzublenden, klicken Sie auf die Schaltfläche "Erweiterter Modus".
2. Selektieren Sie eine oder mehrere Variablen.
3. Wählen Sie im Kontextmenü den Befehl "Kopieren".

4. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
5. Bearbeiten Sie die Variablen nach Bedarf.
6. Kopieren Sie die Variablen im externen Editor.
7. Wechseln Sie wieder in die Deklarationstabelle.
8. Wählen Sie im Kontextmenü den Befehl "Einfügen".

10.11 Online- und Diagnosefunktionen für Datenbausteine

10.11.1 Datenbausteine im Online-Betrieb bearbeiten

Einführung

Wenn Sie einen Datenbaustein in die CPU laden, gibt es von dem Baustein zwei Versionen:

- Eine Offline-Version im Projekt auf Ihrem Programmiergerät
- Eine Online-Version auf der CPU
  Datenbausteine können entweder nur im Ladespeicher, nur im Arbeitsspeicher oder in beiden Speicherbereichen liegen.

Startwerte und Aktualwerte von Variablen

Die Werte der Variablen in der Offline- und Online-Version eines Datenbausteins unterscheiden sich, da die Variablen im Online-Baustein vom Anwenderprogramm zur Laufzeit verändert werden können.

Wenn Sie einen Datenbaustein deklarieren, belegen Sie die Variablen mit einem Startwert. Der Startwert ist der Wert, den die Variable annehmen soll, wenn der Datenbaustein zum ersten Mal aufgerufen wird. Die Variable wird also mit dem Startwert "initialisiert".

Die Startwerte der Variablen werden in der Offline-Version des Bausteins deklariert. Wird der Datenbaustein in die CPU geladen, wird er mit den Startwerten in den Ladespeicher übertragen. Von dort wird der Datenbaustein in den Arbeitsspeicher der CPU geladen. Beim ersten Laden des Bausteins werden die Variablen im Arbeitsspeicher mit den deklarierten Startwerten initialisiert. Nun können sie durch das Anwenderprogramm geändert werden. Die Werte der Variablen im Arbeitsspeicher nennt man "Aktualwerte".

Geänderte Datenbausteine laden

Wenn Sie nach dem ersten Laden Änderungen an einem Datenbaustein durchführen, bearbeiten Sie die Offline-Version des Bausteins. Je nach vorliegenden Randbedingungen hat das anschließende Laden des Datenbausteins unterschiedliche Auswirkungen auf die Aktualwerte.

- Änderungen im Betriebszustand "RUN" laden:
  - Wenn Sie keine Änderungen an der Struktur des Datenbausteins gemacht haben, bleiben die Aktualwerte aller Variablen erhalten. Das Programm wird "stoßfrei" weiter bearbeitet. Eventuelle Änderungen an den Startwerten im Offline-Datenbaustein werden in den Ladespeicher geschrieben. Sie werden erst beim nächsten Wechsel von "STOP" nach "RUN" in die CPU übertragen.
  - Wenn Sie die Struktur des Datenbausteins geändert haben, werden alle Variablen mit ihren Startwerten neu initialisiert. Das Initialisieren der Variablen kann zu ungewünschten Effekten im Prozess führen, weil der Prozess und das Programm möglicherweise nicht mehr synchron sind.
    Dies können Sie vermeiden, indem Sie die Speicherreserve des Bausteins nutzen. Die Speicherreserve ermöglicht es Ihnen, Datenbausteine im "RUN" ohne Reinitialisierung zu laden. Sie können so "stoßfrei" nachladen, auch wenn Sie die Struktur des Datenbausteins geändert und z.B. neue Variablen hinzugefügt haben.
    Siehe auch: Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)
- Änderungen im Betriebszustand "STOP" laden:
  - Die Startwerte von nicht-remanenten Variablen werden in den Ladespeicher der CPU gelegt. Beim nächsten Wechsel von "STOP" nach "RUN" läuft das Programm mit den neuen Startwerten an.
  - Remanente Variablen werden nicht reinitialisiert. Ihre Aktualwerte bleiben nach einem Ladevorgang im "STOP" erhalten.
    Um auch die remanenten Variablen beim Laden in "STOP" zu reinitialisieren, selektieren Sie die zu ladenden Bausteine in der Projektnavigation und wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)". Wählen Sie anschließend im Menü "Online" den Befehl "PLC-Programm ins Gerät laden und zurücksetzen". Die Online-Bausteine werden gelöscht und mit den neuen Bausteinen überschrieben. Dadurch werden alle Variablen, auch die remanenten Variablen, neu initialisiert.

Aktualwerte mit Online- und Diagnosefunktionen bearbeiten

Um Aktualwerte direkt im Online-Betrieb zu beeinflussen, ohne Datenbausteine neu laden zu müssen, können Sie die Online- und Diagnosefunktionen des TIA Portals nutzen.

Siehe auch: Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

Datenbausteine von einem Gerät laden

Sie können Datenbausteine auch wieder von einem Gerät laden. Dies ist sinnvoll, wenn z.B. im Servicefall das ursprünglich Offline-Projekt nicht vorhanden ist. Beachten Sie, dass die beiden CPU-Familien beim Laden von Bausteinen von einem Gerät unterschiedlich mit den Werten der Variablen umgehen:

- S7-1200/1500
  CPUs der Baureihe S7-1200 und S7-1500 legen die Startwerte, mit denen ein Datenbaustein in die CPU geladen wurde, im Ladespeicher ab. Wenn Sie einen Datenbaustein von einer S7-1200/1500-CPU laden, werden diese Startwerte aus dem Ladespeicher wieder in den Offline-Datenbaustein übernommen. Auch wenn Sie zur Laufzeit Werteänderungen im Ladespeicher über die Anweisung "WRIT_DBL" vorgenommen haben, werden die ursprünglich in das Gerät geladenen Werte in den Offline-Datenbaustein übernommen.
- S7-300/400
  CPUs der Baureihe S7-300/400 legen die Startwerte, mit denen ein Datenbaustein in die CPU geladen wurde, ebenfalls im Ladespeicher ab. Sie können die Werte jedoch nicht aus dem Ladespeicher rücklesen. Wenn Sie einen Datenbaustein von einem Gerät laden, werden die aktuellen Beobachtungswerte aus der CPU geladen und in die Spalte "Startwerte" im Offline-Datenbaustein eingetragen.

Siehe auch

Bausteine laden für S7-1200/1500 (Seite 7871)

10.11.2 Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen

Übersicht der Funktionen

Im Datenbaustein-Editor stehen Ihnen verschiedene Möglichkeiten zur Verfügung, Datenbausteine im Online-Betrieb zu bearbeiten. Diese Funktionen greifen direkt auf die Aktualwerte der Variablen im Online-Programm zu. Aktualwerte sind die Werte, die die Variablen zum aktuellen Zeitpunkt während der Programmbearbeitung im Arbeitsspeicher der CPU annehmen.

Die folgende Tabelle gibt eine Übersicht über die Online- und Diagnosefunktionen. Detaillierte Beschreibungen zu den einzelnen Funktionen finden Sie in den nachfolgenden Kapiteln.

Schalt-fläche	Funktion	Beschreibung	S7-300/400	S7-1200/1500
	Variablen online beobachten (Seite 7355)	Zeigt die Aktualwerte, die die Variablen aktuell in der CPU einnehmen.	X	X
-	Einzelne Aktualwerte steuern (Seite 7356)	Steuert einzelne Variablen in der Deklarationstabelle sofort und einmalig auf bestimmte Werte. Die CPU verwendet dann diese Werte als Aktualwerte im Online-Programm.	X	X

Schaltfläche	Funktion	Beschreibung	S7-300/400	S7-1200/1500
	Momentaufnahme der Aktualwerte anlegen (Seite 7359)	Legt die Aktualwerte, die zum aktuellen Zeitpunkt anstehen, als Momentaufnahme im Offline-Projekt ab. Die Momentaufnahme erfasst immer die Aktualwerte aller Variablen des Datenbausteins.	X	X
	Aktualwerte mit einer Momentaufnahme reinitialisieren (Seite 7361)	Lädt die Momentaufnahme als Aktualwerte in die CPU. Die CPU verwendet dann diese Werte als Aktualwerte im Online-Programm.	-	S7-1200 ab V4.1 S7-1500 ab V1.7
	Momentaufnahmen als Startwerte kopieren (Seite 7363)	Kopiert im Offline-Programm die Momentaufnahme in die Startwerte. Beim nächsten Wechsel von STOP nach RUN läuft das Programm mit den neuen Startwerten an. Sie können alle Startwerte oder nur die Startwerte der als "Einstellwert" markierten Variablen kopieren.	X	X
	Aktualwerte mit Startwerten reinitialisieren (Seite 7357)	Reinitialisiert die Aktualwerte von Variablen. Die Startwerte werden dabei direkt in den Arbeitsspeicher der CPU geschrieben. Sie können alle Aktualwerte oder nur die Aktualwerte der als "Einstellwert" markierten Variablen kopieren.	X	X

Siehe auch

Einstellwerte im Online-Programm initialisieren (Seite 7367)

Werte aus dem Online-Programm als Startwerte übernehmen (Seite 7368)

Datenbausteine im Online-Betrieb bearbeiten (Seite 7352)

Einstellwerte während der Inbetriebnahme nutzen (Seite 7365)

Startwerte definieren (Seite 7336)

10.11.3 Variablen beobachten

Sie können die Aktualwerte, die die Variablen aktuell in der CPU einnehmen, direkt in der Deklarationstabelle beobachten.

Voraussetzung

- Eine Online-Verbindung besteht.
- Der Datenbaustein ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Der Datenbaustein ist geöffnet.

Vorgehen

Um die Variablen zu beobachten, gehen Sie folgendermaßen vor:

1. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle beobachten" klicken. Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.
   Siehe auch: Aufbau der Deklarationstabelle für Datenbausteine (Seite 7322)
2. Beenden Sie die Beobachtung, indem Sie erneut auf die Schaltfläche "Alle beobachten" klicken.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.4 Variablen steuern

Sie können eine einzelne Variable im Datenbaustein auf einen bestimmten Wert steuern. Die CPU verwendet dann diesen Wert als Aktualwert im Online-Programm.

> ⚠ GEFAHR
>
> **Gefahr beim Verändern von Variablenwerten**
>
> Ein Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!
>
> Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Variablen steuern.

Voraussetzung

- Eine Online-Verbindung zur CPU besteht.
- Der Datenbaustein, dessen Variablen Sie steuern möchten, ist offline und online identisch.
- Der Datenbaustein ist geöffnet.

Vorgehen

Um eine einzelne Variable im Datenbaustein zu steuern, gehen Sie folgendermaßen vor:

1. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle beobachten" klicken. Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.
2. Doppelklicken Sie auf die zu steuernde Variable.
   - Bei booleschen Variablen öffnet sich der Dialog "Wert umschalten". Bestätigen Sie diesen Dialog mir "OK".
     Wenn Sie in diesem Dialog die Option "Diesen Hinweis zukünftig nicht mehr anzeigen" wählen, können Sie den Wert zukünftig durch Doppelklick umschalten.
   - Bei nicht-booleschen Variablen öffnet sich der Dialog "Wert steuern". Geben Sie im Eingabefeld "Steuerwert" den gewünschten Wert ein und bestätigen Sie mit "OK". Falls eine Momentaufnahme existiert, ist der Wert bereits als Voreinstellung eingetragen.

Ergebnis

Beim Ausführen des Steuerauftrags nimmt die Variable einmalig den angegebenen Wert an. Die Ausführung des Auftrags erfolgt sofort und ist nicht an den nächsten Zykluskontrollpunkt gebunden.

Siehe auch

Einführung zum Steuern von Variablen (Seite 8106)

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.5 Startwerte als Aktualwerte laden

Sie können die Startwerte aus dem Offline-Programm als Aktualwerte in die CPU laden. Die Startwerte werden dabei direkt in den Arbeitsspeicher der CPU geladen. Die Variablen im Online-Baustein werden reinitialisiert und die CPU verwendet die neuen Werte als Aktualwerte im Online-Programm. Es wird nicht zwischen remanenten und nicht-remanenten Werten unterschieden. Sie können alle Aktualwerte oder nur die Aktualwerte der als "Einstellwert" markierten Variablen kopieren.

So können Sie den gesteuerten Prozess schnell in einen definierten Zustand bringen.

Folgende Arten von Datenbausteinen können Sie nicht reinitialisieren:

- Datenbausteine, die Teil eines Referenzprojekts sind
- Fehlersichere Datenbausteine
- Von der CPU erzeugte Datenbausteine
- Datenbausteine mit dem Attribut "Nur im Ladespeicher ablegen"
- Datenbausteine, die nur Online vorhanden sind

Diese Funktion können Sie sowohl im Betriebszustand "RUN" als auch im Betriebszustand "STOP" ausführen.

⚠ GEFAHR

Gefahr beim Verändern von Variablenwerten

Ein Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Falls die zu übertragende Datenmenge zu groß ist, werden die Werte möglicherweise in mehreren Zyklen übertragen. Wenn das Programm auf Variablen zugreift, bevor alle Werte komplett übertragen wurden, besteht die Gefahr, dass inkonsistente Wertkombinationen entstehen und weiterverarbeitet werden.

Auch das Kopieren der Werte elementarer Datentypen kann möglicherweise über mehrere Zyklen erfolgen. Diese Werte sind so lange potenziell ungültig, bis sie komplett übertragen wurden. Wenn das Programm auf diese Werte zugreift, bevor sie komplett übertragen wurden, können gefährliche Zustände eintreten.

Um zu verhindern, dass gefährliche Zustände eintreten, beachten Sie folgende Hinweise:

- Stellen Sie sicher, dass die Anlage sich in einem sicheren Zustand befindet, bevor Sie die Aktualwerte überschreiben.
- Stellen Sie sicher, dass das Programm die betroffenen Daten während der Übertragung weder lesend noch schreibend bearbeitet.
- Verwenden Sie gegebenenfalls ersatzweise die Funktion "Variable steuern" in der Beobachtungstabelle oder im DB-Editor.

Voraussetzung

- Sie verwenden eine S7-1200 ab V4.1 oder eine S7-1500 ab V1.7.
- Eine Online-Verbindung zur CPU besteht.
- Die Datenbausteine, deren Aktualwerte Sie reinitialisieren wollen, sind offline und online identisch.

Aktualwerte eines geöffneten Bausteins reinitialisieren

Um alle Aktualwerte oder nur die Einstellwerte eines Bausteins mit Startwerten zu reinitialisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Klicken Sie auf die Schaltfläche "Startwerte als Aktualwerte laden > Alle Werte".

Oder:

1. Öffnen Sie den Datenbaustein.
2. Klicken Sie auf die Schaltfläche "Startwerte als Aktualwerte laden > Nur Einstellwerte".

Aktualwerte mehrerer Bausteine reinitialisieren

Um alle Aktualwerte mehrerer Bausteine in einem Schritt zu reinitialisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie mehrere Bausteine oder den gesamten Bausteinordner in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Startwerte als Aktualwerte laden".

Ergebnis

Die Aktualwerte im Online-Programm werden mit den Startwerten reinitialisiert.

Die maximale Anzahl initialisierbarer Variablen ist CPU-abhängig. Wenn die Anzahl der Variablen zu hoch ist, werden Sie mit einer Meldung darüber informiert. In diesem Fall können Sie die Variablen in eine Beobachtungstabelle einfügen und über die Funktion "Steuern" in der Beobachtungstabelle initialisieren.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.6 Momentaufnahme der Aktualwerte anlegen

Sie können die Aktualwerte der Variablen eines oder mehrerer Datenbausteine als Momentaufnahme ablegen. Aktualwerte sind die Werte, die die Variablen zum aktuellen Zeitpunkt während der Programmbearbeitung im Arbeitsspeicher der CPU annehmen.

Für Datenbausteine, die nur im Ladespeicher abgelegt werden, und nicht in das Programm eingebunden sind, kann keine Momentaufnahme angelegt werden.

Grundsätzlich haben Sie zum Anlegen einer Momentaufnahme folgende Möglichkeiten:

- Momentaufnahme eines geöffneten Datenbausteins erzeugen
- Momentaufnahme mehrerer selektierter Datenbausteine erzeugen

Darüber hinaus wird beim Laden eines Bausteins oder eines Programms aus dem Gerät automatisch eine Momentaufnahme erzeugt.

⚠ VORSICHT
Erzeugen der Momentaufnahme
Die Werte in der Momentaufnahme können möglicherweise aus mehreren Zyklen stammen.

Voraussetzung

- Eine Online-Verbindung zur CPU besteht.
- Die Datenbausteine, für die Sie eine Momentaufnahme anlegen möchten, sind offline und online identisch.

Vorgehen

Um eine Momentaufnahme eines geöffneten Datenbausteins anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Klicken Sie auf die Schaltfläche "Momentaufnahme der Aktualwerte".

Um eine Momentaufnahme mehrerer selektierter Datenbausteine anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Bausteine in der Projektnavigation.
   Sie können die Bausteine einzeln selektieren, oder Geräte, Gruppen oder Ordner in der Projektnavigation markieren, die Datenbausteine enthalten.
2. Wählen Sie im Kontextmenü den Befehl "Momentaufnahme der Aktualwerte" oder wählen Sie den Menübefehl "Online > Momentaufnahme der Aktualwerte".

Ergebnis

- Die aktuellen Beobachtungswerte werden in die Spalte "Momentaufnahme" übernommen.
- Wenn der Vorgang beendet ist, wird eine Meldung im Inspektorfenster ausgegeben.
- Der Zeitstempel der Momentaufnahme wird über der Deklarationstabelle angezeigt.

Hinweis

Strukturänderungen am Datenbaustein

Wenn Sie nach dem Anlegen einer Momentaufnahme die Struktur des Datenbausteins ändern, werden möglicherweise einzelne Werte in der Momentaufnahme ungültig. Wenn Sie beispielsweise den Datentyp einer Variablen ändern, wird der Wert dieser Variable ungültig. In der Spalte "Momentaufnahme" wird für diese Variable kein Wert angezeigt.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.7 Momentaufnahmen als Aktualwerte laden

Sie können die Momentaufnahmen aus dem Offline-Programm als Aktualwerte in die CPU laden. Die Werte aus der Momentaufnahme werden dabei direkt in den Arbeitsspeicher der CPU geladen. Die Variablen im Online-Baustein werden reinitialisiert und die CPU verwendet die neuen Werte als Aktualwerte im Online-Programm.

So können Sie den gesteuerten Prozess schnell in einen definierten Zustand bringen.

Folgende Arten von Datenbausteinen können Sie nicht mit Momentaufnahmen reinitialisieren:

- Datenbausteine, die Teil eines Referenzprojekts sind
- Fehlersichere Datenbausteine
- Von der CPU erzeugte Datenbausteine
- Datenbausteine mit dem Attribut "Nur im Ladespeicher ablegen"
- Datenbausteine, die nur Online vorhanden sind

⚠ GEFAHR

Gefahr beim Verändern von Variablenwerten

Ein Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

- Stellen Sie sicher, dass die Anlage sich in einem sicheren Zustand befindet, bevor Sie die Aktualwerte überschreiben.
- Stellen Sie sicher, dass das Programm die betroffenen Daten während der Übertragung weder lesend noch schreibend bearbeitet.
- Verwenden Sie gegebenenfalls ersatzweise die Funktion "Variable steuern" in der Beobachtungstabelle oder im DB-Editor.

Abhängigkeiten vom Betriebszustand der CPU

Diese Funktion können Sie sowohl im Betriebszustand "RUN" als auch im Betriebszustand "STOP" ausführen. Die folgende Tabelle zeigt die Reaktionen der CPU in den unterschiedlichen Betriebszuständen:

Aktion	Systemreaktion	Folgen für das Online-Programm
Aktualwerte im Betriebszustand "RUN" überschreiben	Die Werte aller Variablen des DB im laufenden Programm werden überschrieben. Es wird nicht zwischen remanenten und nicht-remanenten Werten unterschieden.	Das Ändern der Aktualwerte kann zu Inkonsistenzen zwischen dem Programm und dem tatsächlichen Prozess führen. Falls die zu übertragende Datenmenge zu groß ist, werden die Werte möglicherweise in mehreren Zyklen übertragen. Wenn das Programm auf Variablen zugreift, bevor alle Werte komplett übertragen wurden, besteht die Gefahr, dass inkonsistente Wertkombinationen entstehen und weiterverarbeitet werden. Auch das Kopieren der Werte elementarer Datentypen kann möglicherweise über mehrere Zyklen erfolgen. Diese Werte sind so lange potenziell ungültig, bis sie komplett übertragen wurden. Wenn das Programm auf diese Werte zugreift, bevor sie komplett übertragen wurden, können gefährliche Zustände eintreten.
Aktualwerte im Betriebszustand "STOP" überschreiben	Es werden lediglich die Aktualwerte der remanenten Variablen mit der Momentaufnahme überschrieben. Nicht-remanente Variablen werden beim Übergang von "STOP" auf "RUN" mit ihren Startwerten initialisiert. Die Werte aus der Momentaufnahme bleiben dabei unberücksichtigt.	Da nur die remanenten Daten aus der Momentaufnahme übertragen werden, besteht die Gefahr, dass inkonsistente Wertkombinationen entstehen und weiterverarbeitet werden.

Voraussetzung

- Sie verwenden eine S7-1200 ab V4.1 oder eine S7-1500 ab V1.7.
- Eine Online-Verbindung zur CPU besteht.
- Die Datenbausteine, deren Aktualwerte Sie reinitialisieren wollen, sind offline und online identisch.
- Eine Momentaufnahmen der Datenbausteine ist angelegt.

Aktualwerte eines geöffneten Bausteins reinitialisieren

Um alle Aktualwerte eines Bausteins mit Momentaufnahmen zu reinitialisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Klicken Sie auf die Schaltfläche "Momentaufnahmen als Aktualwerte laden".

Aktualwerte mehrerer Bausteine reinitialisieren

Um die Aktualwerte mehrerer Bausteine in einem Schritt zu reinitialisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie mehrere Bausteine oder den gesamten Bausteinordner in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Momentaufnahmen als Aktualwerte laden".

Ergebnis

Die Aktualwerte im Online-Programm werden mit den Werten aus der Momentaufnahme reinitialisiert. Variablen, für die keine gültige Momentaufnahme existiert, werden mit ihren Startwerten reinitialisiert.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.8 Momentaufnahme in die Startwerte kopieren

Sie können im Offline-Programm die Momentaufnahme in die Startwerte kopieren. Beim nächsten Wechsel von "STOP" nach "RUN" läuft das Programm mit den neuen Startwerten an.

Sie können alle Startwerte, die Startwerte der remanenten Variablen oder nur die Startwerte der als "Einstellwert" markierten Variablen kopieren.

Beachten Sie, dass immer die Werte aus der Momentaufnahme kopiert werden. Dabei wird nicht geprüft, ob alle Werte aus demselben Zyklus stammen.

Schreibgeschützte Startwerte werden nicht überschrieben.

Grundsätzlich haben Sie zur Übernahme der Werte folgende Möglichkeiten:

- Übernehmen der Werte eines geöffneten Datenbausteins
  In einem geöffneten Datenbaustein können Sie alle Werte oder nur die Werte der als "Einstellwert" markierten Variablen als Startwerte übernehmen.
- Übernehmen der Werte mehrerer Bausteine in der Projektnavigation
  In der Projektnavigation können Sie wahlweise alle Werte, alle Einstellwerte oder alle remanenten Werte als Startwerte übernehmen.

Voraussetzung

Eine Momentaufnahme für den Datenbaustein ist angelegt.

Vorgehen

Um in einem Datenbaustein die Momentaufnahmen als Startwerte zu übernehmen, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen Datenbaustein.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Momentaufnahmen als Startwerte kopieren > Alle Werte".

Oder:

1. Öffnen Sie einen Datenbaustein.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Momentaufnahmen als Startwerte kopieren > Nur Einstellwerte".

Um die Beobachtungswerte mehrerer Datenbausteine in der Projektnavigation zu übernehmen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bausteine in der Projektnavigation.
2. Wählen Sie anschließend im Kontextmenü einen der folgenden Befehle:
   - "Momentaufnahmen als Startwerte kopieren > Alle Werte"
   - "Momentaufnahmen als Startwerte kopieren > Nur Einstellwerte"
   - "Momentaufnahmen als Startwerte kopieren > Nur Remanenzwerte"

Ergebnis

Die Werte werden aus der Spalte "Momentaufnahme" in die Spalte "Startwert" übernommen. Die neuen Startwerte werden im Offline-Programm gespeichert.

Hinweis

Werte einzelner Variablen übernehmen

Sie können auch die Werte einzelner Variablen aus der Spalte "Momentaufnahme" in die Spalte "Startwerte" übernehmen. Kopieren Sie dazu die Werte mithilfe der Befehle "Kopieren" und "Einfügen" aus dem Kontextmenü und fügen Sie sie in die Spalte "Startwert" ein. Beachten Sie, dass dabei nur die Werte kopiert werden, die momentan im sichtbaren Bereich der Tabelle liegen.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.9 Einstellwerte während der Inbetriebnahme nutzen

10.11.9.1 Grundlagen zum Anpassen von Einstellwerten während der Inbetriebnahme

Einführung

Bei der Inbetriebnahme einer Anlage müssen Datenwerte häufig justiert werden, um das Programm optimal an die Rahmenbedingungen vor Ort anzupassen. Zu diesem Zweck können Sie die Online- und Diagnosefunktionen der Deklarationstabelle für Datenbausteine nutzen.

Um die Funktionen einzusetzen, definieren Sie zunächst bestimmte Variablen im Programm als "Einstellwerte". Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen.

Die folgende Tabelle gibt eine Übersicht über die Funktionen zum Justieren von Werten bei der Inbetriebnahme. Detaillierte Beschreibungen zu den einzelnen Funktionen finden Sie in den nachfolgenden Kapiteln.

Darüber hinaus gibt es auch die allgemeinen Funktionen zum Beobachten und Steuern des Datenbausteins:

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

Schaltfläche	Funktion	Beschreibung
	Einstellwerte im Betriebszustand "RUN" initialisieren (Seite 7367)	Diese Funktion ermöglicht es Ihnen, die Werte einzelner Variablen online zu ändern, um so schnell die optimalen Variablenwerte zu ermitteln.
	Werte aus dem Online-Programm als Startwerte in das Offline-Programm übernehmen (Seite 7368)	Wenn Sie die optimalen Variablenwerte ermittelt haben, können Sie diese als Startwerte in das Offline-Programm übernehmen. So sorgen Sie dafür, dass das Programm nach dem nächsten Laden mit den optimierten Werten anläuft.

Siehe auch

Daten als einstellbare Werte kennzeichnen (Seite 7365)

10.11.9.2 Daten als einstellbare Werte kennzeichnen

Sie können bestimmte Variablen im Programm als "Einstellwerte" kennzeichnen. Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen.

Regeln

Sie können Variablen in den folgenden Bausteinarten als "Einstellwert" kennzeichnen:

- In Funktionsbausteinen (FB), jedoch nur im Abschnitt "Static"
- In globalen Datenbausteinen (DB)
- In PLC-Datentypen (UDT)
  Bei PLC-Datentypen wird die Einstellung jedoch nur wirksam, wenn der UDT im Abschnitt "Static" eines Funktions- oder Datenbausteins verwendet wird.

In folgenden Bausteinarten ist die Definition von Einstellwerten nicht möglich:

- In Datenbausteinen, die auf einem PLC-Datentyp basieren, und in Instanz-Datenbausteinen. Diese erben die Einstellung vom übergeordneten FB oder UDT.
- Variablen in ARRAY-Datenbausteinen können Sie nicht als "Einstellwert" kennzeichnen.
- An der Aufrufstelle einer Multiinstanz können Sie Variablen ebenfalls nicht als "Einstellwert" kennzeichnen. Sie müssen die Einstellung in der Schnittstelle des Funktionsbausteins vornehmen, der als Multiinstanz aufgerufen wird.
- In know-how-geschützten Bausteinen können Sie die Kennzeichnung als "Einstellwert" nicht ändern. Dazu müssen Sie zunächst den Know-how-Schutz entfernen.

Voraussetzung

Ein Funktionsbaustein, ein globaler Datenbaustein oder ein PLC-Datentyp (UDT) ist geöffnet.

Vorgehen

Um eine Variable als "Einstellwert" zu kennzeichnen, gehen Sie folgendermaßen vor:

1. Wählen Sie eine Variable aus dem Abschnitt "Static" aus.
2. Aktivieren Sie das Optionskästchen in der Spalte "Einstellwert".
   - Das übergeordnete Element einer Struktur oder eines PLC-Datentyps können Sie nicht als "Einstellwert" definieren. Sie müssen die Einstellung für die unterlagerten Elemente einzeln vornehmen.
   - Bei ARRAYs können Sie nur das übergeordnete Element als "Einstellwert" kennzeichnen. Die unterlagerten Elemente erben die Einstellung.
   - Bei ARRAYs of STRUCT können Sie nur die Elemente unterhalb der ersten Struktur als "Einstellwerte" kennzeichnen. Die Elemente weiterer Strukturen erben die Einstellung.

Ergebnis

Die Variablen sind als Einstellwerte markiert. Während der Inbetriebnahme können diese Variablen online initialisiert werden. Die CPU muss dazu nicht in den Betriebszustand "STOP" gesetzt werden, sondern kann im Betriebszustand "RUN" bleiben. Außerdem können die aktuellen Variablenwerte als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

Online- und Diagnosefunktionen für Datenbausteine (Seite 7352)

Startwerte definieren (Seite 7336)

10.11.9.3 Einstellwerte im Online-Programm initialisieren

Grundlagen zum Initialisieren von Einstellwerten

Sie können alle Variablen, die als "Einstellwert" markiert sind, im Online-Programm mit neuen Werten initialisieren. Dabei werden die Startwerte aus dem Offline-Programm in das Online-Programm geladen. Die CPU bleibt im Betriebszustand "RUN". Alle Variablen, die als Einstellwert markiert sind, werden am nächsten Zykluskontrollpunkt einmalig initialisiert. Dies gilt sowohl für remanente als auch für nicht-remanente Variablen. Die Programmbearbeitung wird anschließend mit den neuen Variablenwerten fortgesetzt.

Die maximale Anzahl initialisierbarer Variablen ist CPU-abhängig: Bei CPUs der Baureihe S7-300/400 können Sie bis maximal 35 Einstellwerte initialisieren, bei CPUs der Baureihe S7-1200/1500 liegt die Maximalmenge bei 200 Einstellwerten. Abhängig von den verwendeten Datentypen ist die Maximalmenge jedoch niedriger.

Wenn die Anzahl der markierten Einstellwerte zu hoch ist, werden Sie mit einer Meldung darüber informiert. In diesem Fall können Sie die Variablen in eine Beobachtungstabelle einfügen und über die Funktion "Steuern" in der Beobachtungstabelle initialisieren.

⚠ GEFAHR

Gefahr beim Verändern von Variablenwerten

Ein Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Einstellwerte neu initialisieren.

Voraussetzung

- Eine Online-Verbindung zur CPU ist vorhanden.
- Die Struktur des Datenbausteins ist Offline und Online identisch.
- Eine oder mehrere Variablen sind als "Einstellwert" markiert.

Vorgehen

Um alle Einstellwerte des Datenbausteins zu initialisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen globalen Datenbaustein oder einen Instanz-Datenbaustein.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein. Die Startwerte müssen dem angegebenen Datentyp entsprechen.
3. Klicken Sie auf die Schaltfläche "Startwerte als Aktualwerte laden > Nur Einstellwerte".

Ergebnis

Die Einstellwerte im Online-Programm werden am nächsten Zykluskontrollpunkt mit den Startwerten aus dem Offline-Programm initialisiert.

Siehe auch

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

10.11.9.4 Werte aus dem Online-Programm als Startwerte übernehmen

Um Variablenwerte aus dem Online-Programm als Startwerte in das Offline-Programm zu übernehmen, erzeugen Sie zunächst eine Momentaufnahme der Variablenwerte aus dem Online-Programm. Diese können Sie anschließend in das Offline-Progamm übernehmen. Beachten Sie, dass immer die Werte aus der Momentaufnahme kopiert werden. Dabei wird nicht geprüft, ob alle Werte aus demselben Zyklus stammen.

Schreibgeschützte Startwerte werden nicht überschrieben.

Grundsätzlich haben Sie zur Übernahme der Werte folgende Möglichkeiten:

- Übernehmen der Einstellwerte eines geöffneten Datenbausteins
- Übernehmen der Einstellwerte mehrerer Bausteine in der Projektnavigation

Voraussetzung

- Eine Online-Verbindung zur CPU besteht.
- Mindestens ein Datenbaustein ist in die CPU geladen.

Vorgehen

Um in einem Datenbaustein alle Einstellwerte als Startwerte zu übernehmen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle beobachten" klicken. Die Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.

3. Klicken Sie in der Funktionsleiste auf "Momentaufnahme der Aktualwerte".
   Die Aktualwerte werden in die Spalte "Momentaufnahme" übernommen.
4. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Momentaufnahmen als Startwerte kopieren > Nur Einstellwerte".

Die Werte werden aus der Spalte "Momentaufnahme" in die Spalte "Startwerte" übernommen.

Um die Beobachtungswerte mehrerer Datenbausteine in der Projektnavigation zu übernehmen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bausteine in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Momentaufnahme der Aktualwerte".
   Die Aktualwerte aller markierten Bausteine werden in die Spalte "Momentaufnahme" übernommen.
   Wenn der Vorgang beendet ist, wird eine Meldung im Inspektorfenster ausgegeben.
3. Wählen Sie anschließend im Kontextmenü den Befehl "Momentaufnahme als Startwerte übernehmen > Nur Einstellwerte".

Die Werte werden aus der Spalte "Momentaufnahme" in die Spalte "Startwert" übernommen.

Ergebnis

Die neuen Startwerte werden im Offline-Programm gespeichert.

Hinweis

Werte einzelner Variablen übernehmen

Sie können auch die Werte einzelner Variablen, die nicht zuvor als Einstellwert markiert wurden, aus der Spalte "Momentaufnahme" in die Spalte "Startwerte" übernehmen. Kopieren Sie dazu die Werte mithilfe der Befehle "Kopieren" und "Einfügen" aus dem Kontextmenü und fügen Sie sie in die Spalte "Startwert" ein. Beachten Sie, dass dabei nur die Werte kopiert werden, die momentan im sichtbaren Bereich der Tabelle liegen.

Siehe auch

Grundlagen zu Startwerten (Seite 7336)

Startwerte definieren (Seite 7337)

Übersicht über die Online-und Diagnosefunktionen in Datenbausteinen (Seite 7354)

Online- und Diagnosefunktionen für Datenbausteine (Seite 7352)

PLC-Variablen deklarieren 11

11.1 Übersicht über PLC-Variablentabellen

Einführung

PLC-Variablentabellen enthalten die Definitionen der CPU-weit gültigen PLC-Variablen und symbolischen Konstanten. Für jede im Projekt angelegte CPU wird automatisch eine PLC-Variablentabelle angelegt. Sie können selbst weitere Variablentabellen anlegen, um Variablen und Konstanten darin zu sortieren und zu gruppieren.

In der Projektnavigation gibt es für jede CPU des Projekts einen Ordner "PLC-Variablen". Darin sind folgende Tabellen enthalten:

- Tabelle "Alle Variablen"
- Standard-Variablentabelle
- Optional: Weitere anwenderdefinierte Variablentabellen

Alle Variablen

Die Tabelle "Alle Variablen" zeigt eine Übersicht über alle PLC-Variablen, Anwenderkonstanten und Systemkonstanten der CPU. Diese Tabelle kann nicht gelöscht oder verschoben werden.

Standard-Variablentabelle

Für jede CPU des Projekts gibt es eine Standard-Variablentabelle. Sie kann nicht gelöscht, umbenannt oder verschoben werden. Die Standard-Variablentabelle enthält PLC-Variablen, Anwenderkonstanten und Systemkonstanten. Sie können alle PLC-Variablen in der Standard-Variablentabelle deklarieren, oder nach Bedarf weitere anwenderdefinierte Variablentabellen anlegen.

Anwenderdefinierte Variablentabellen

Sie können mehrere anwenderdefinierte Variablentabellen für jede CPU anlegen, um Variablen nach Ihren Bedürfnissen zu gruppieren. Anwenderdefinierte Variablentabellen können Sie umbenennen, in Gruppen zusammenfassen oder löschen. Anwenderdefinierte Variablentabellen können PLC-Variablen und Anwenderkonstanten enthalten.

Siehe auch

Aufbau der PLC-Variablentabellen (Seite 7373)
Variablen (Seite 100)
Grundlagen zu Konstanten (Seite 102)

11.2 Aufbau der PLC-Variablentabellen

Einführung

Jede PLC-Variablentabelle enthält je ein Register für Variablen und Anwenderkonstanten. Die Standard-Variablentabelle und die Tabelle "Alle Variablen" enthalten zusätzlich ein Register "Systemkonstanten".

Aufbau des Registers "Variablen"

Im Register "Variablen" deklarieren Sie die globalen PLC-Variablen, die Sie im Programm benötigen. Das folgende Bild zeigt den Aufbau des Registers. Die Anzahl der angezeigten Spalten kann variieren.

	Name	Datentyp	Adresse	Remanenz
	Motor1	Bool	%Q3.1	☐
	Motor2	Bool	%M3.2	☑
	Control	Bool	%I3.3	☐

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Anzahl der angezeigten Spalten kann variieren. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

Spalte	Erläuterung
	Symbol, das Sie anklicken können, um die Variable per Drag & Drop in ein Programm zu ziehen und dort als Operand zu verwenden.
Name	CPU-weit eindeutiger Name der Variablen.
Datentyp	Datentyp der Variablen.
Adresse	Adresse der Variablen.
Remanenz	Kennzeichnet die Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.
Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Operandenauswahl von HMI per Voreinstellung sichtbar ist.
Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.
Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
Überwachung	Zeigt an, ob für die Variable eine Überwachung zur Prozessdiagnose angelegt wurde.
Beobachtungswert	Aktueller Datenwert in der CPU. Die Spalte ist nur dann eingeblendet, wenn eine Online-Verbindung besteht und Sie die Schaltfläche "Alle beobachten" anwählen.
Variablentabelle	Zeigt an, in welcher Variablentabelle die Variablendeklaration enthalten ist. Diese Spalte ist nur in der Tabelle "Alle Variablen" enthalten.
Kommentar	Kommentar zur Dokumentation der Variablen.

Aufbau der Register "Anwenderkonstanten" und "Systemkonstanten"

Im Register "Anwenderkonstanten" definieren Sie CPU-weit gültige symbolische Konstanten. Im Register "Systemkonstanten" werden vom System benötigte Konstanten angezeigt. Systemkonstanten können z. B. Hardware-Kennungen sein, die zur Identifikation von Baugruppen dienen.

Das folgende Bild zeigt den Aufbau der beiden Register. Die Anzahl der gezeigten Spalten kann variieren.

	Name	Datentyp	Wert	Kommentar
	Const_1	Bool	true	
	Const_2	Byte	12	
	Const_3	Bool	false	
	Const_4	Real	1.0	

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

Spalte	Erläuterung
	Symbol, das Sie anklicken können, um die Variable per Drag & Drop in ein Netzwerk zu ziehen und dort als Operand zu verwenden.
Name	CPU-weit eindeutiger Name der Konstanten.
Datentyp	Datentyp der Konstanten.
Wert	Wert der Konstanten.
Variablentabelle	Zeigt an, in welcher Variablentabelle die Konstantendeklaration enthalten ist. Diese Spalte ist nur in der Tabelle "Alle Variablen" enthalten.
Kommentar	Kommentar zur Dokumentation der Konstanten.

Siehe auch

Hardware-Datentypen (Seite 366)

Variablen (Seite 100)

Grundlagen zu Konstanten (Seite 102)

Übersicht über PLC-Variablentabellen (Seite 7371)

Tabellenspalten ein- und ausblenden (Seite 7402)

11.3 Regeln für PLC-Variablen

11.3.1 Zulässige Namen von PLC-Variablen

Zulässige Zeichen

Für die Namen von PLC-Variablen gelten folgende Regeln:

- Buchstaben, Ziffern, Sonderzeichen sind zulässig.
- Die Verwendung reservierter Schlüsselwörter wird nicht empfohlen.
- Anführungszeichen als Teil des Variablennamens sind nicht zulässig.

Eindeutige Variablennamen

Die Namen der PLC-Variablen müssen CPU-weit eindeutig sein, auch wenn sich die Variablen in unterschiedlichen Variablentabellen einer CPU befinden. Einen Namen, der bereits für einen Baustein, eine andere PLC-Variable oder eine Konstante innerhalb der CPU vergeben ist, können Sie nicht für eine neue PLC-Variable verwenden. Groß- und Kleinschreibung wird bei der Prüfung der Eindeutigkeit nicht unterschieden.

Wenn Sie einen bereits vergebenen Namen erneut eintragen, wird automatisch an den zweiten Namen eine fortlaufende Nummer angehängt. Wenn Sie z. B. zwei Mal den Namen "Motor" eingeben, wird der zweite Eintrag in "Motor(1)" umgewandelt.

Eindeutige Tabellennamen

Auch die Namen der PLC-Variablentabellen müssen CPU-weit eindeutig sein. Beim Anlegen von anwenderdefinierten PLC-Variablentabellen wird automatisch ein eindeutiger Name vorgeschlagen.

Hinweis

Weitere Informationen zu zulässigen Zeichen in Variablennamen finden Sie im Siemens Industry Online Support in folgenden Beiträgen:

Wann müssen Bezeichner bzw. Operanden in STEP 7 (TIA Portal) in "Anführungszeichen" verwendet werden?

https://support.industry.siemens.com/cs/ww/de/view/10947785 (https://support.industry.siemens.com/cs/ww/de/view/109477857)

Siehe auch

Variablen (Seite 100)

Schlüsselwörter (Seite 93)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

PLC-Variablendeklaration eingeben (Seite 7382)

11.3.2 Zulässige Adressen und Datentypen von PLC-Variablen

Die Adressen von PLC-Variablen bestehen aus der Angabe des Operandenbereichs und der Adresse innerhalb dieses Bereichs.

Die Adressen müssen CPU-weit eindeutig sein. Wenn Sie eine Adresse eintragen, die bereits einer anderen Variablen zugewiesen ist, wird die Adresse an beiden Stellen gelb markiert und eine Fehlermeldung ausgegeben.

Operandenbereiche

Die folgende Tabelle zeigt die möglichen Operandenbereiche. Die verfügbaren Datentypen sind abhängig von der verwendeten CPU:

Operandenbereich		Erläuterung	Datentyp	Format	Adressbereich:		
Internationale Mnemonik	Deutsche Mnemonik				S7-1200	S7-300/400	S7-1500
I	E	Eingangsbit	BOOL	I x.y E x.y	0.0..1023.7	0.0..65535.7	0.0..32767.7
I	E	Eingang (64 Bit)	LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, PLC-Datentyp	I x.0 E x.0	-	-	0.0..32760.0
IB	EB	Eingangsbyte	BYTE, CHAR, SINT, USINT, PLC-Datentyp	IB x EB y	0..1023	0..65535	0..32767
IW	EW	Eingangswort	WORD, INT, UINT, DATE, WCHAR, S5TIME, PLC-Datentyp	IW x EW y	0..1022	0..65534	0..32766
ID	ED	Eingangsdoppelwort	DWORD, DINT, UDINT, REAL, TIME, TOD, PLC-Datentyp	ID x ED y	0..1020	0..65532	0..32764
Q	A	Ausgangsbit	BOOL	Q x.y A x.y	0.0..1023.7	0.0..65535.7	0.0..32767.7

Operandenbereich		Erläuterung	Datentyp	Format	Adressbereich:		
Internationale Mnemonik	Deutsche Mnemonik				S7-1200	S7-300/400	S7-1500
Q	A	Ausgang (64 Bit)	LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, PLC-Datentyp	Q x.0 A x.0	-	-	0.0..32760.0
QB	AB	Ausgangsbyte	BYTE, CHAR, SINT, USINT, PLC-Datentyp	QB x AB y	0..1023	0..65535	0..32767
QW	AW	Ausgangswort	WORD, INT, UINT, DATE, WCHAR, S5TIME, PLC-Datentyp	QW x AW y	0..1022	0..65534	0..32766
QD	AD	Ausgangsdoppelwort	DWORD, DINT, UDINT, REAL, TIME, TOD, PLC-Datentyp	QD x AD y	0..1020	0..65532	0..32764
M	M	Merkerbit	BOOL	M x.y	0.0..8191.7	0.0..65535.7	0.0..16383.7
M	M	Merker (64 Bit)	LREAL	M x.0	0.0..8184.0	-	0.0..16376.0
M	M	Merker (64 Bit)	LWORD, LINT, ULINT, LTIME, LTOD, LDT	M x.0	-	-	0.0..16376.0
MB	MB	Merkerbyte	BYTE, CHAR, SINT, USINT	MB x	0..8191	0..65535	0..16383
MW	MW	Merkerwort	WORD, INT, UINT, DATE, WCHAR, S5TIME	MW x	0..8190	0..65534	0..16382
MD	MD	Merkerdoppelwort	DWORD, DINT, UDINT, REAL, TIME, TOD	MD x	0..8188	0..65532	0..16380
T	T	Zeitfunktion (nur bei S7-300/400)	Timer	T n	-	0..65535	0..2047
C	Z	Zählfunktion (nur bei S7-300/400)	Counter	Z n C n	-	0..65535	0..2047

Adressen

Die folgende Tabelle zeigt die möglichen Adressen der Variablen:

Datentyp	Adresse	Beispiel
BOOL	Variablen mit Datentyp BOOL werden mit einer Bytenummer und einer Bitnummer adressiert. Die Nummerierung der Bytes beginnt für jeden Operandenbereich bei 0. Die Nummerierung der Bits geht von 0 bis 7.	A 1.0
BYTE, CHAR, SINT, USINT	Variablen mit Datentyp BYTE, CHAR, SINT, USINT werden mit einer Bytenummer adressiert.	MB 1
WORD, INT, UINT, DATE, WCHAR, S5TIME	Variablen mit Datentyp WORD, INT, UINT, DATE, S5TIME bestehen aus zwei Bytes. Sie werden mit der Nummer des niedrigeren Bytes adressiert.	EW 1
DWORD, DINT, UDINT, REAL, TIME, TOD	Variablen mit Datentyp DWORD, DINT, UDINT, REAL, TIME, TOD bestehen aus vier Bytes. Sie werden mit der Nummer des niedrigeren Bytes adressiert.	AD 1
LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL	Variablen mit Datentyp LWORD, LINT, ULINT, LTIME, LTOD, LDT,LREAL bestehen aus acht Bytes. Sie werden mit der Nummer des niedrigeren Bytes ind der Bitnummer 0 adressiert.	E 1.0

Verwendete Mnemonik

Die Adressen, die Sie in der PLC-Variablentabelle eingeben, werden automatisch an die eingestellte Mnemonik angepasst.

Siehe auch

Einstellen der Mnemonik (Seite 7130)

Variablen (Seite 100)

Zulässige Namen von PLC-Variablen (Seite 7375)

Übersicht über die gültigen Datentypen (Seite 243)

PLC-Variablendeklaration eingeben (Seite 7382)

11.4 PLC-Variablentabellen anlegen und verwalten

11.4.1 PLC-Variablentabelle anlegen

Sie können mehrere anwenderdefinierten PLC-Variablentabellen in einer CPU anlegen. Jede Variablentabelle muss einen CPU-weit eindeutigen Namen haben.

Voraussetzung

Die Projektansicht ist geöffnet.

Vorgehen

Um eine neue PLC-Variablentabelle anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
2. Doppelklicken Sie auf den Eintrag "Neue Variablentabelle hinzufügen".
   Eine neue PLC-Variablentabelle mit dem Standardnamen "Variablentabelle_x" wird angelegt.
3. Selektieren Sie die PLC-Variablentabelle in der Projektnavigation.
4. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
5. Geben Sie einen CPU-weit eindeutigen Namen ein.

Ergebnis

Eine neue PLC-Variablentabelle wird angelegt. In dieser können Sie Variablen und Konstanten deklarieren.

Siehe auch

Übersicht über PLC-Variablentabellen (Seite 7371)

Aufbau der PLC-Variablentabellen (Seite 7373)

PLC-Variablen deklarieren (Seite 7382)

Globale Konstanten deklarieren (Seite 7391)

PLC-Variablen exportieren und importieren (Seite 7402)

11.4.2 PLC-Variablentabellen gruppieren

Sie können anwenderdefinierte Variablentabellen einer CPU in Gruppen zusammenfassen. Die Standard-Variablentabelle und die Tabelle "Alle Variablen" können Sie jedoch nicht in eine Gruppe verschieben.

Voraussetzung

Im Ordner "PLC-Variablen" der CPU sind mehrere anwenderdefinierte Variablentabellen enthalten.

Vorgehen

Um mehrere PLC-Variablentabellen in einer Gruppe zusammenzufassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
2. Wählen Sie den Menübefehl "Einfügen > Gruppe".
   Eine neue Gruppe mit dem Standardnamen "Gruppe_x" wird eingefügt.
3. Selektieren Sie die neu eingefügte Gruppe in der Projektnavigation.
4. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
5. Geben Sie der neuen Gruppe einen CPU-weit eindeutigen Namen.
6. Ziehen Sie die zu gruppierenden Tabellen per Drag & Drop in die neue Gruppe.

Ergebnis

Die Variablentabellen werden in der neuen Gruppe zusammengefasst.

Siehe auch

Übersicht über PLC-Variablentabellen (Seite 7371)

Aufbau der PLC-Variablentabellen (Seite 7373)

PLC-Variablen deklarieren (Seite 7382)

Globale Konstanten deklarieren (Seite 7391)

11.4.3 PLC-Variablentabelle öffnen

Vorgehen

Um die PLC-Variablentabelle einer CPU zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
2. Doppelklicken Sie auf die PLC-Variablentabelle im Ordner.
3. Wählen Sie in der rechten oberen Ecke das gewünschte Register.

Ergebnis

Die zur CPU gehörende PLC-Variablentabelle wird geöffnet. Sie können die benötigten Variablen und Konstanten deklarieren.

Siehe auch

Übersicht über PLC-Variablentabellen (Seite 7371)
Aufbau der PLC-Variablentabellen (Seite 7373)
PLC-Variablen deklarieren (Seite 7382)
Globale Konstanten deklarieren (Seite 7391)

11.5 PLC-Variablen deklarieren

11.5.1 PLC-Variablendeklaration eingeben

11.5.1.1 Variablen in der PLC-Variablentabelle deklarieren

Voraussetzung

Das Register "Variablen" einer PLC-Variablentabelle ist geöffnet.

Vorgehen

Um PLC-Variablen zu definieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
   Eine dem Datentyp entsprechende Adresse wird automatisch ergänzt.
3. Optional: Klicken Sie in der Spalte "Adresse" auf die Pfeiltaste und geben Sie im nachfolgenden Dialog ein Operandenkennzeichen, einen Operandentyp, eine Adresse und eine Bitnummer ein.
4. Optional: Deaktivieren Sie die Optionen für HMI und OPC UA.
5. Optional: Tragen Sie in der Spalte "Kommentar" einen Kommentar ein.
6. Wiederholen Sie die Schritte 1 - 5 für alle erforderlichen Variablen.
   Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)

Syntaxprüfung

Nach jeder Eingabe wird automatisch eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält und die Variable im Programm verwendet wird, lässt sich das Programm allerdings nicht übersetzen.

Siehe auch

PLC-Variablen exportieren und importieren (Seite 7402)

Autovervollständigung verwenden (Seite 7147)

Zulässige Namen von PLC-Variablen (Seite 7375)

PLC-Variablen im Programmiereditor deklarieren (Seite 7383)

PLC-Variablentabellen bearbeiten (Seite 7400)

Aufbau der PLC-Variablentabellen (Seite 7373)

11.5.1.2 PLC-Variablen im Programmiereditor deklarieren

Voraussetzung

- Der Programmiereditor ist geöffnet.

Vorgehen

Um Operanden als globale PLC-Variablen zu deklarieren, gehen Sie folgendermaßen vor:

1. Fügen Sie eine Anweisung in Ihr Programm ein.
   Die Zeichenfolgen "<???>", "<??.?>" oder "..." stellen Operandenplatzhalter dar.
2. Ersetzen Sie einen Operandenplatzhalter mit dem Namen der zu erstellenden PLC-Variablen.
3. Selektieren Sie den Variablennamen.
   Wenn Sie mehrere PLC-Variablen deklarieren wollen, selektieren Sie die Namen aller zu deklarierenden Variablen.
4. Wählen Sie im Kontextmenü den Befehl "Variable definieren".
   Der Dialog "Variable definieren" wird geöffnet. Er zeigt eine Deklarationstabelle, in die bereits der Variablenname eingetragen ist.
5. Klicken Sie in der Spalte "Abschnitt" auf die Pfeiltaste und wählen Sie einen der folgenden Einträge:
   - Global Memory
   - Global Input
   - Global Output
   - Global Timer
   - Global Counter
6. Geben Sie in den weiteren Spalten Adresse, Datentyp und Kommentar ein.
   Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)
7. Falls die CPU mehrere PLC-Variablentabellen enthält, können Sie in der Spalte "PLC-Variablentabelle" eingeben, in welche Tabelle die Variable eingefügt werden soll. Wenn Sie in der Spalte keine Angabe machen, wird die neue Variable in die Standard-Variablentabelle eingefügt.
8. Klicken Sie auf die Schaltfläche "Definieren", um die Eingabe abzuschließen.

Ergebnis

Die Variablendeklaration wird in die PLC-Variablentabelle geschrieben und ist für alle Bausteine in der CPU gültig.

Siehe auch

Zulässige Namen von PLC-Variablen (Seite 7375)

Variablen in der PLC-Variablentabelle deklarieren (Seite 7382)

Autovervollständigung verwenden (Seite 7147)

11.5.1.3 Zur Gerätesicht navigieren

Sie können aus einer PLC-Variablentabelle heraus in den Geräte- und Netzwerkeditor navigieren und den konfigurierten Ein- oder Ausgang in der Gerätesicht anzeigen.

Voraussetzung

- Die Variable ist als Ein- oder Ausgang in einer PLC-Variablentabelle deklariert.
- Im Hardware- und Netzwerkeditor ist eine Ein- oder Ausgabebaugruppe konfiguriert.

Vorgehen

Um eine PLC-Variable in der Gerätesicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable in der PLC-Variablentabelle.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu Gerätesicht".

Der Geräte- und Netzwerkeditor wird geöffnet und der Ein- oder Ausgang, für den die Variable deklariert wurde, wird angezeigt.

11.5.2 Remanenz von PLC-Variablen einstellen

11.5.2.1 Remanenzverhalten von PLC-Variablen

Remanente PLC-Variablen

Jede CPU verfügt über einen Speicherbereich, dessen Inhalt auch nach dem Ausschalten der Versorgungsspannung erhalten bleibt. Dieser Bereich wird als remanenter Speicherbereich bezeichnet.

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie bestimmte PLC-Variablen in diesem Speicherbereich ablegen. Die Remanenzeinstellung von PLC-Variablen legen Sie in der PLC-Variablentabelle (S7-1200/S7-1500) oder in den CPU-Eigenschaften (S7-300/S7-400) fest.

Je nach CPU-Familie kann der remanente Speicherbereich unterschiedliche Arten von PLC-Variablen aufnehmen. Die folgende Tabelle gibt eine Übersicht über die Möglichkeiten der unterschiedlichen CPUs:

CPU-Typ	Remanente Merker	Remanente SIMATIC-Zeiten	Remanente SIMATIC-Zähler
Baureihe S7-300/400	✓	✓	✓
Baureihe S7-1200	✓	-	-
Baureihe S7-1500	✓	✓	✓

Siehe auch

Remanenzverhalten von PLC-Variablen einstellen (Seite 7385)

Remanenzverhalten von PLC-Variablen einstellen (Seite 7386)

11.5.2.2 Remanenzverhalten von PLC-Variablen einstellen

Einführung

Für die CPU-Familien S7-1200/S7-1500 legen Sie die Remanenz für PLC-Variablen in der PLC-Variablentabelle fest. Die Remanenzeinstellung einer Variablen erkennen Sie am gesetzten Häkchen in der Spalte "Remanenz" der PLC-Variablentabelle.

Voraussetzung

Das Register "Variablen" der PLC-Variablentabelle ist geöffnet.

Vorgehen

Um die Breite des remanenten Speicherbereichs für PLC-Variablen einr CPU S7-1200/S7-1500 zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz".
   Der Dialog "Remanenter Speicher" wird geöffnet.
2. Legen Sie die Breite des remanenten Speicherbereichs fest:
   - Tragen Sie die Anzahl der remanenten Merkerbytes ein.
     Beispiel: Wenn Sie "2" eingeben, sind die Merkerbytes MB 0 und MB 1 remanent.
   - Tragen Sie die Anzahl der remanenten Zeiten ein. Jeder S7-Timer belegt zwei Byte.
     Beispiel: Wenn Sie "2" eingeben, sind die S7-Timer T 0 und T 1 remanent.
   - Tragen Sie die Anzahl der remanenten Zähler ein. Jeder S7-Zähler belegt zwei Byte.
   - Beispiel: Wenn Sie "2" eingeben, sind die S7-Zähler C 0 und C 1 remanent.
3. Klicken Sie auf die Schaltfläche "OK".

Ergebnis

Die Breite des remanenten Speicherbereichs wird definiert. Alle Variablen, deren Adressen in diesem Speicherbereich liegen, werden als remanent gekennzeichnet. In der Spalte "Remanenz" der Variablentabelle wird automatisch ein Häkchen für alle Variablen gesetzt, die im remanenten Speicherbereich liegen.

Siehe auch

Remanenzverhalten von PLC-Variablen (Seite 7384)

11.5.2.3 Remanenzverhalten von PLC-Variablen einstellen

Einführung

Für die CPU-Familien S7-300/400 legen Sie die Remanenz für PLC-Variablen in den CPU-Einstellungen fest.

Vorgehen

Um die Breite des remanenten Speicherbereichs für PLC-Variablen einer CPU S7-300/400 zu definieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die CPU.
2. Wählen Sie den Menübefehl "Eigenschaften".
3. Wählen Sie den Abschnitt "Remanenz".
   - Tragen Sie die Anzahl der remanenten Merkerbytes ein.
     Beispiel: Wenn Sie "2" eingeben, sind die Merkerbytes MB 0 und MB 1 remanent.
   - Tragen Sie die Anzahl der remanenten Zeiten ein. Jeder S7-Timer belegt zwei Byte.
     Beispiel: Wenn Sie "2" eingeben, sind die S7-Timer T 0 und T 1 remanent.
   - Tragen Sie die Anzahl der remanenten Zähler ein. Jeder S7-Zähler belegt zwei Byte.
     Beispiel: Wenn Sie "2" eingeben, sind die S7-Zähler C 0 und C 1 remanent.

Ergebnis

Die Breite des remanenten Speicherbereichs wird definiert. Alle Variablen, die in diesem Speicherbereich liegen, werden als remanent deklariert.

Siehe auch

Remanenzverhalten von PLC-Variablen (Seite 7384)

11.6 PLC-Variablen für Ein- und Ausgänge in Strukturen zusammenfassen

11.6.1 Wissenswertes zu strukturierten PLC-Variablen

Nutzen von strukturierten PLC-Variablen (S7-1200 ab V4/S7-1500)

Um Ihr Programm übersichtlicher zu gestalten, können Sie mehrere Ein- oder Ausgangsadressen in einer übergeordneten PLC-Variable zusammenfassen. Die übergeordnete PLC-Variable stellt eine Struktur dar, die mehrere logisch zusammengehörige Ein- oder Ausgänge enthält. Beim Bausteinaufruf übergeben Sie die übergeordnete Variable und benötigen somit nur einen Ein- oder Ausgangsparameter für alle zusammengehörenden Ein- oder Ausgänge.

Funktionsweise

Um eine strukturierte PLC-Variable zu erstellen, definieren Sie zunächst einen PLC-Datentypen (UDT). Darin deklarieren Sie die benötigten Datenelemente und legen deren Namen und Datentypen fest.

Anschließend wechseln Sie in die PLC-Variablentabelle und legen dort die übergeordnete PLC-Variable an. Als Datentyp für die Variable wählen Sie Ihren PLC-Datentypen aus. Das System reserviert nun ab der Startadresse der übergeordneten Variable eine bestimmte Anzahl von Ein- oder Ausgangsadressen. Die Anzahl der reservierten Adressen hängt von der Länge Ihres PLC-Datentyps ab.

Wenn Sie einen Baustein aufrufen, der die reservierten Ein- oder Ausgänge zur Programmbearbeitung benötigt, übergeben Sie die übergeordnete Variable als Bausteinparameter.

Im Programmcode können Sie die einzelnen PLC-Variablen wie Strukturelemente adressieren.

Eine detaillierte Beschreibung der einzelnen Handlungsschritte finden Sie in den nachfolgenden Kapiteln.

Anwendungsbeispiel

Sie können strukturierte PLC-Variablen einsetzen, um die Ein- oder Ausgänge eines Funktionsmoduls zu gruppieren. Das folgende Bild zeigt die schematische Darstellung eines Motors. Für jeden der drei Eingänge wurde eine Komponente im PLC-Datentyp "Datatype_Motor" angelegt.

Die Speicherbereiche der deklarierten Variablen dürfen sich nicht überlappen. Im Beispiel sehen Sie, dass die Komponente "Speed" vom Datentyp "Integer" ist und deshalb an einer Wortadresse beginnen muss. Aus diesem Grund wurde das erste Eingangswort mit der Füll-Variable "Dummy" aufgefüllt. "Speed" liegt somit auf dem zweiten Eingangswort.

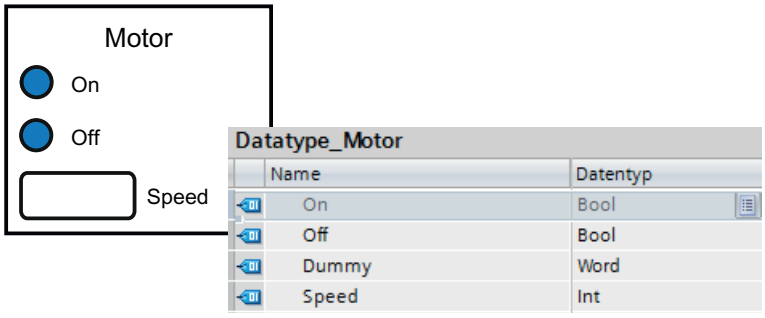

Das folgende Bild zeigt die übergeordnete PLC-Variable "Motor", die auf dem Datentyp "Datatype_Motor" beruht. Durch die Deklaration von "Motor" werden die Adressen EW0 und EW1 auf der Eingabebaugruppe reserviert.

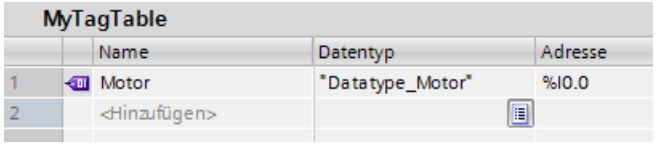

Das folgende Bild zeigt die Übergabe der PLC-Variable "Motor" als Eingangsparameter des Bausteins "Motor_Control".

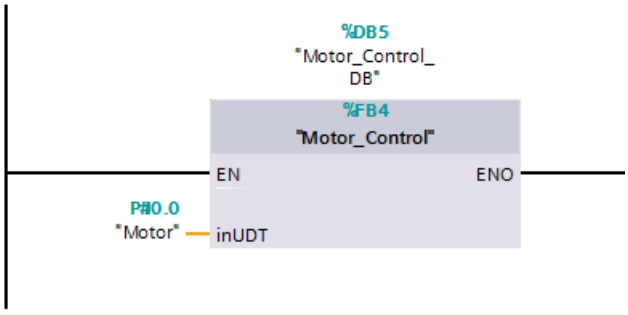

Im Baustein "Motor_Control" können Sie die einzelnen Komponenten der Variable adressieren.

Adressierung	Erläuterung
"Motor"	Adressierung der übergeordneten PLC-Variable.
"Motor".On	Adressierung einer Komponente einer strukturierten PLC-Variable.
"Motor".On:P	Adressierung eines Peripherieeingangs oder -ausgangs (PE oder PA).

Regeln zur Verwendung strukturierter PLC-Variablen

Beachten Sie bei der Erstellung und Verwendung strukturierter PLC-Variablen folgende Regeln:

- Strukturierte PLC-Variablen sind in den Operandenbereichen "Eingänge" und "Ausgänge" verwendbar.
- Im Merkerbereich sind strukturierte PLC-Variablen nicht zulässig.
- Aus HMI können strukturierte PLC-Variablen nicht adressiert werden.

Beachten Sie bei der Erstellung des PLC-Datentyps, der als Basis für eine PLC-Variable dienen soll, folgende Regeln:

- Die Speicherbereiche der einzelnen Elemente dürfen sich nicht überlappen.
  Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)
- Fassen Sie Ein- und Ausgänge nicht in einem PLC-Datentypen zusammen, sondern erzeugen Sie verschiedene PLC-Datentypen für Ein- und Ausgänge.
- Fassen Sie Ein- oder Ausgänge aus verschiedenen Baugruppen nicht in einem PLC-Datentyp zusammen, da nicht gewährleistet ist, dass die Prozessabbilder der Baugruppen synchron aktualisiert werden.
- In den unterlagerten PLC-Datentypen sind alle Datentypen mit Ausnahme von "STRING" und "WSTRING" zulässig.

Siehe auch

Strukturierte PLC-Variablen anlegen (Seite 7389)

Globale Variablen adressieren (Seite 108)

11.6.2 Strukturierte PLC-Variablen anlegen

Regeln

Beachten Sie bei der Erstellung strukturierter PLC-Variablen folgende Regeln:

- Verwenden Sie getrennte PLC-Datentypen für die Operandenbereiche "Eingänge" und "Ausgänge".
- Im Merkerbereich sind strukturierte PLC-Variablen nicht zulässig.
- Fassen Sie Ein- oder Ausgänge aus verschiedenen Baugruppen nicht in einem PLC-Datentyp zusammen, da nicht gewährleistet ist, dass die Prozessabbilder der Baugruppen synchron aktualisiert werden.

Vorgehen

Um eine strukturierte PLC-Variable zu erstellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie in dem PLC-Datentypen alle benötigten Komponenten. Alle Datentypen mit Ausnahme von "STRING" und "WSTRING" sind zulässig.
3. Selektieren Sie in der Projektnavigation den PLC-Datentypen und wählen Sie aus dem Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".
   Der PLC-Datentyp wird übersetzt und kann anschließend in der PLC-Variablentabelle verwendet werden.
   Auch wenn Sie Änderungen an bestehenden PLC-Datentypen durchführen, müssen Sie das Programm erneut übersetzen. Dadurch werden alle Verwendungsstellen des PLC-Datentyps aktualisiert.
4. Öffnen Sie eine PLC-Variablentabelle innerhalb derselben CPU.
5. Deklarieren Sie eine neue Variable oder wählen Sie eine bestehende Variable aus.
6. Wählen Sie in der Spalte "Datentyp" den PLC-Datentypen aus und weisen Sie ihn der PLC-Variable zu.
   Die PLC-Variable erhält die Struktur des PLC-Datentyps. Eine passende Adresse wird automatisch vergeben. Strukturierte PLC-Variablen beginnen immer an Wortadressen.
   In der Tabelle wird nur das oberste Strukturelement ohne seine Unterelemente dargestellt.

Hinweis

Belegungsregeln und Defaultwerte

- Beachten Sie bei der Deklaration des PLC-Datentyps, dass die Speicherbereiche der einzelnen Variablen sich nicht überlappen dürfen. Zum Beispiel müssen Variablen vom Datentyp "Integer" an einer Wortgrenze beginnen. Fügen Sie, wenn nötig, "Füll-Variablen" ein, um Überlappungen zu vermeiden.
  Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 7376)
- Es ist nicht möglich, Defaultwerte für die einzelnen Komponenten zu vergeben. Werte, die Sie in die Spalte "Defaultwert" eintragen, werden nicht ausgewertet. Variablen der Datentypen "DT" und "DTL" können deshalb möglicherweise ungültige Werte enthalten.

Siehe auch

Wissenswertes zu strukturierten PLC-Variablen (Seite 7387)

Globale Variablen adressieren (Seite 108)

PLC-Variablen deklarieren (Seite 7382)

11.7 Globale Konstanten deklarieren

11.7.1 Regeln für globale Anwenderkonstanten

Zulässige Zeichen

Namen von gobalen Konstanten dürfen aus folgenden Zeichen bestehen:

- Buchstaben, Ziffern, Sonderzeichen sind zulässig.
- Anführungszeichen sind nicht zulässig.

Eindeutige Konstantennamen

Die Namen der globalen Konstanten müssen CPU-weit eindeutig sein, auch wenn sich die Konstanten in unterschiedlichen Variablentabellen einer CPU befinden. Einen Namen, der bereits für einen Baustein, eine PLC-Variable oder eine andere Konstante innerhalb der CPU vergeben ist, können Sie nicht für eine neue Konstante verwenden. Groß- und Kleinschreibung wird bei der Prüfung der Eindeutigkeit nicht unterschieden.

Wenn Sie einen bereits vergebenen Namen erneut eintragen, wird automatisch an den zweiten Namen eine fortlaufende Nummer angehängt. Wenn Sie z. B. zwei Mal den Namen "Motor" eingeben, wird der zweite Eintrag in "Motor(1)" umgewandelt.

Zulässige Datentypen

Für Konstanten sind alle von der CPU unterstützten Datentypen zulässig, mit Ausnahme der strukturierten Datentypen.

Zulässige Werte

Als Konstantenwert können Sie einen beliebigen Wert aus dem Wertebereich des angegebenen Datentyps wählen. Angaben zu den Wertebereichen finden Sie im Kapitel "Datentypen".

Siehe auch: Übersicht über die gültigen Datentypen (Seite 243)

Siehe auch

Grundlagen zu Konstanten (Seite 102)

Globale Konstanten deklarieren (Seite 7392)

Konstanten (Seite 102)

11.7.2 Regeln für globale Systemkonstanten

Definition

Systemkonstanten sind CPU-weit eindeutige, globale Konstanten, die vom System benötigt und automatisch angelegt werden. Systemkonstanten können z. B. der Adressierung und Identifizierung von Hardware-Objekten dienen.

Regeln

Systemkonstanten werden automatisch beim Stecken von Komponenten in der Geräte- oder Netzsicht vergeben und in die Standard-Variablentabelle eingetragen (Register "Systemkonstanten"). Für jede Baugruppe, aber auch für jedes Submodul wird eine Systemkonstante angelegt. So erhält z. B. auch ein integrierter Zähler eine Systemkonstante. Systemkonstanten bestehen aus einem symbolischen Namen sowie einer numerischen HW-Kennung und sind nicht änderbar.

Namen von Systemkonstanten

Die Namen der Systemkonstanten sind hierarchisch aufgebaut. Sie bestehen aus maximal vier Hierarchieebenen, die jeweils durch eine Tilde "~" getrennt sind. Somit können Sie anhand des Namens den "Pfad" zu dem betreffenden Hardware-Modul erkennen.

Beispiel

Eine Systemkonstante mit dem Namen "`Local~PROFINET_interface_1~Port_1`" bezeichnet Port 1 der PROFINET-Schnittstelle 1 der lokalen CPU.

Siehe auch

Anweisungen zur Adressumwandlung (Seite 2849)

11.7.3 Globale Konstanten deklarieren

Einführung

Konstanten deklarieren Sie im Register "Anwenderkonstanten" einer PLC-Variablentabelle. Bei der Deklaration ist für jede Konstante die Eingabe eines symbolischen Namens, eines Datentyps und eines festen Werts erforderlich. Als Konstantenwert können Sie einen beliebigen Wert aus dem Wertebereich des angegebenen Datentyps wählen. Angaben zu den Wertebereichen finden Sie im Kapitel "Datentypen".

Siehe auch: Übersicht über die gültigen Datentypen (Seite 243)

Vorgehen

Um Konstanten zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablentabelle.
2. Öffnen Sie das Register "Anwenderkonstanten".
   Die Konstantentabelle wird geöffnet.
3. Tragen Sie in der Spalte "Name" einen Konstantennamen ein.
4. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
5. Tragen Sie in der Spalte "Wert" einen Konstantenwert ein, der für den gewählten Datentyp zulässig ist.
6. Tragen Sie in der Spalte "Kommentar" einen Kommentar ein, falls Sie Anmerkungen zu der Konstante haben. Die Eingabe eines Kommentars ist optional.
7. Falls Sie weitere Konstanten deklarieren wollen, setzen Sie die Einfügemarke in die nächste Zeile und wiederholen Sie die Schritte 3 bis 6.

Syntaxprüfung

Nach jeder Eingabe wird automatisch eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält und die Konstante im Programm verwendet wird, lässt sich das Programm allerdings nicht übersetzen.

Siehe auch

PLC-Variablentabelle öffnen (Seite 7380)

Tabellenzeilen in der PLC-Variablentabelle einfügen (Seite 7400)

Aufbau der PLC-Variablentabellen (Seite 7373)

Regeln für globale Anwenderkonstanten (Seite 7391)

Autovervollständigung verwenden (Seite 7147)

Konstanten (Seite 102)

11.8 Eigenschaften bearbeiten

11.8.1 Eigenschaften von PLC-Variablen bearbeiten

11.8.1.1 Eigenschaften von PLC-Variablen

Überblick

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von PLC-Variablen. Die Darstellung der Eigenschaften kann abhängig vom CPU-Typ variieren.

Gruppe	Eigenschaft	Beschreibung
Allgemein	Name	Innerhalb der CPU eindeutiger Name.
	Datentyp	Datentyp der Variablen.
	Adresse	Adresse der Variablen.
	Remanent	Zeigt an, ob die Variable im remanenten Speicherbereich liegt.
	Kommentar	Kommentar zur Variablen.
Verlauf	Erstellt am	Erstellzeitpunkt der Variablen (nicht änderbar).
	Zuletzt geändert am	Änderungszeitpunkt der Variablen (nicht änderbar).
Verwendung	Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
	Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.
	Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.

Siehe auch

Eigenschaften von PLC-Variablen bearbeiten (Seite 7394)

11.8.1.2 Eigenschaften von PLC-Variablen bearbeiten

Eigenschaften in einer PLC-Variablentabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf die PLC-Variablentabelle, in der sich die Variablen befinden.
   Die PLC-Variablentabelle wird geöffnet.
2. Ändern Sie die Einträge in den Spalten.

Adressen im Programmiereditor bearbeiten

Um die Adresse einer Variable im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variabnlennamen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umverdrahten".
   Der Dialog "Variable umverdrahten" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Tragen Sie in der Spalte "Adresse" die neue Adresse ein.
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

Namen im Programmiereditor bearbeiten

Um den Namen einer Variablen im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variablennamen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umbenennen".
   Der Dialog "Variable umbenennen" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Tragen Sie in der Spalte "Name" den neuen Namen ein.
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

Auswirkung im Programm

Bei einer Änderung des Namens, des Datentyps oder der Adresse einer Variablen wird automatisch jede Verwendungsstelle der Variablen im Programm aktualisiert.

Siehe auch

Eigenschaften von PLC-Variablen (Seite 7394)

11.8.2 Eigenschaften von globalen Konstanten bearbeiten

11.8.2.1 Eigenschaften von globalen Konstanten

Überblick

Folgende Tabelle gibt eine Übersicht über die Eigenschaften von Konstanten:

Gruppe	Eigenschaft	Beschreibung
Allgemein	Name	Innerhalb der Tabelle eindeutiger Name
	Datentyp	Datentyp der Konstanten
	Wert	Wert, den Sie für die Konstante definiert haben. Dieser Wert muss kompatibel zum angegebenen Datentyp sein. Siehe auch: Übersicht über die gültigen Datentypen (Seite 243)
	Kommentar	Kommentar zur Konstanten
Verlauf	Erstellt am	Erstellzeitpunkt der Konstanten (nicht änderbar)
	Zuletzt geändert am	Änderungszeitpunkt der Konstanten (nicht änderbar)

11.8.2.2 Eigenschaften von globalen Konstanten bearbeiten

Eigenschaften in einer PLC-Variablentabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Konstanten zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf die PLC-Variablentabelle, in der sich die Konstanten befinden.
   Die PLC-Variablentabelle wird geöffnet.
2. Öffnen Sie das Register "Anwenderkonstanten".
3. Ändern Sie die Einträge in den Spalten "Name", "Datentyp", "Wert" oder "Kommentar".

Auswirkung im Programm

Bei einer Änderung des Namens, Datentyps oder Werts einer Konstanten wird automatisch jede Verwendungsstelle der Konstanten im Programm aktualisiert.

11.8.3 Mehrsprachige Kommentare in der PLC-Variablentabelle bearbeiten

Sie können die Kommentare aus einer PLC-Variablentabelle in allen Projektsprachen bearbeiten, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Beachten Sie, dass die Texte auch nach der Übersetzung die Länge von 32767 Unicode-Zeichen nicht überschreiten dürfen.

Voraussetzung

- Sie haben mehrere Projektsprachen aktiviert.
- Eine PLC-Variablentabelle ist geöffnet und enthält mindestens einen Kommentar.
- Der Baustein trägt keinen Know-how-Schutz.

Vorgehen

Um einen Kommentar in allen Projektsprachen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie den Kommentar an, dessen Übersetzung Sie bearbeiten wollen.
   Wenn Sie die Übersetzungen mehrerer Kommentare gleichzeitig bearbeiten wollen, markieren Sie alle gewünschten Kommentare.
   Kommentare eines PLC-Datentyps (UDT) sind in der PLC-Variablentabelle nicht anwählbar. Diese Kommentare können Sie nur im PLC-Datentyp selbst bearbeiten.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das untergeordnete Register "Texte".
   Im Register "Texte" werden die markierten Kommentare in den aktiven Projektsprachen und, soweit vorhanden, die Übersetzungen der Kommentare angezeigt.
4. Sie können die Übersetzungen direkt in die Tabelle im Register "Texte" bearbeiten.
5. Um die Übersetzungen mit einem externen Editor zu bearbeiten, können Sie die angezeigten Texte über die Schaltflächen "Projekttexte exportieren/importieren" in das Format OOXML exportieren.

Hinweis

Alle Projekttexte in der globalen Tabelle "Projekttexte" bearbeiten

Sie können die Übersetzungen für die einzelnen Projektsprachen auch in der globale Tabelle "Projekttexte" bearbeiten. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte". Sie enthält alle übersetzbaren Texte des gesamten Projekts.

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

11.9 PLC-Variablen beobachten

11.9.1 PLC-Variablen beobachten

Sie können die Datenwerte, die die Variablen aktuell in der CPU annehmen, direkt in der PLC-Variablentabelle beobachten.

Voraussetzung

Eine Online-Verbindung zur CPU besteht oder ist möglich.

Vorgehen

Um die Datenwerte zu beobachten, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablentabelle.
2. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle beobachten" klicken.
   - Falls noch keine Online-Verbindung zur CPU besteht, wird diese aufgebaut.
   - Das Beobachten wird mit der Triggereinstellung "Permanent" gestartet.
   - Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.
   - Das Symbol für das Forcen von Variablen wird angezeigt, falls eine Variable gerade geforct wird.
3. Beenden Sie die Beobachtung, indem Sie erneut auf die Schaltfläche "Alle beobachten" klicken.

Hinweis

Bearbeiten von PLC-Variablen während des Beobachtens von Variablen

Wenn das Beobachten von Variablen gestartet ist und die PLC-Variablentabelle bearbeitet wird, z.B. durch Hinzufügen neuer Variablen, so wird das Beobachten nach Abschluss des Bearbeitens erneut gestartet.

Hinweis

Zusätzlich haben Sie die Möglichkeit, PLC-Variablen in eine Beobachtungs- oder Forcetabelle zu kopieren und sie dort zu beobachten, zu steuern oder zu forcen.

Siehe auch

Aufbau der PLC-Variablentabellen (Seite 7373)

Einführung zum Testen mit der Beobachtungstabelle (Seite 8084)

Einführung zum Testen mit der Forcetabelle (Seite 8115)
Einträge in der PLC-Variablentabelle kopieren (Seite 7400)

11.10 PLC-Variablentabellen bearbeiten

11.10.1 Tabellenzeilen in der PLC-Variablentabelle einfügen

Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

11.10.2 Einträge in der PLC-Variablentabelle kopieren

Sie können PLC-Variablen innerhalb einer Tabelle oder in andere Tabellen kopieren.

Vorgehen

Um eine Variable zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable, die Sie kopieren möchten.
   Sie können auch mehrere Variablen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Variable anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die Variable einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Oder

1. Markieren Sie die Variable.
2. Halten Sie die linke Maustaste gedrückt.
3. Drücken Sie gleichzeitig die <Strg>-Taste.
4. Ziehen Sie die Variable an das Ziel.

Ergebnis

- Die Variable wird an das Ziel kopiert.
- Bei einem Namenskonflikt wird der Variablenname automatisch um eine Nummer erweitert. Aus "Variable" wird zum Beispiel "Variable(1)".
- Alle weiteren Eigenschaften der Variablen bleiben gleich.

11.10.3 Einträge in der PLC-Variablentabelle löschen

Vorgehen

Um Elemente zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit dem zu löschenden Element. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

11.10.4 Zeilen in PLC-Variablentabellen sortieren

Sie können die Zeilen in Tabellen alphanumerisch nach Name, Datentyp oder Adresse sortieren.

Vorgehen

Um Tabellenzeilen zu sortieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die Spalte aus, nach der Sie sortieren möchten.
2. Klicken Sie in den Spaltenkopf.
   Die Spalte wird aufsteigend sortiert.
   Ein Pfeil-nach-oben zeigt die Sortierreihenfolge an.
3. Um die Sortierreihenfolge zu ändern, klicken Sie auf den Pfeil.
   Die Spalte wird absteigend sortiert. Ein Pfeil-nach-unten zeigt die Sortierreihenfolge an.
4. Um die ursprüngliche Anordnung wiederherzustellen, klicken Sie ein drittes Mal auf den Spaltenkopf.

11.10.5 Zellen in der PLC-Variablentabelle automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor_1".

Wenn Sie Zellen in der Spalte "Adresse" automatisch ausfüllen, werden die Adressen abhängig vom angegebenen Datentyp erhöht.

Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.
   Der Mauszeiger wird zu einem Fadenkreuz.

3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.
   Die Zellen werden automatisch ausgefüllt. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog, in dem Sie angeben können, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

11.10.6 Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
5. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.

11.10.7 PLC-Variablen exportieren und importieren

11.10.7.1 Grundlagen zum Exportieren und Importieren von PLC-Variablen

Einführung

Sie können PLC-Variablentabellen exportieren, um sie mit externen Applikationen weiterzubearbeiten. Ebenso können Sie PLC-Variablentabellen, die extern bearbeitet wurden, in das TIA Portal importieren.

Beim Import und Export werden folgende Formate unterstützt:

- XLSX (Excel)
- XML
- SDF

Beim Export können Sie angeben, welche Elemente Sie exportieren wollen:

- Variablen und / oder Konstanten
- Alle Elemente oder nur die Elemente, die tatsächlich im Programm auf der aktuellen CPU verwendet werden.

Überschreiben bestehender PLC-Variablen und Konstanten beim Import

Beim Import werden bereits bestehende Einträge überschrieben, wenn sie denselben Namen haben wie zu importierende Einträge.

Verbindung zu bestehenden Objekten

Wenn im Projekt bereits Verweise auf PLC-Variablen oder Konstanten vorhanden sind, werden diese Verweise beim Import automatisch aktualisiert. Die Aktualisierung wird auf Basis des Namens der PLC-Variablen und Konstanten durchgeführt.

Siehe auch

Format der Exportdatei (*.xlsx) (Seite 7403)

Exportieren von PLC-Variablen (Seite 7405)

Importieren von PLC-Variablen (Seite 7406)

11.10.7.2 Format der Exportdatei (*.xlsx)

Einführung

Beim Export von PLC-Variablentabellen wird ein standardisiertes XSLX-Format erzeugt, das Sie mit externen Tabelleneditoren bearbeiten können.

Dieses Format wird auch beim Import von Tabellen erwartet.

Format der Exportdatei

Der Name des Blatts lautet immer "PLCTags". In diesem Blatt können die dargestellten Spalten enthalten sein. Die Reihenfolge der Spalten kann variieren. Es müssen nicht alle Spalten vorhanden sein, jedoch darf die Tabelle keine zusätzlichen Spalten enthalten. Fehlende Werte werden beim Import durch die Angabe <no value> gekennzeichnet.

Die Benennungen der Spaltentitel sind ebenfalls fest definiert und werden immer in Englisch erwartet.

Die folgende Tabelle gibt an, welche Inhalte in den einzelnen Spalten erwartet werden:

Element	Erläuterung
Name	Name der Variablen
Path	Gruppe und Name der PLC-Variablentabelle

Element	Erläuterung
Data Type	Die Notation des Datentyps entspricht der in der PLC-Variablentabelle verwendeten Notation.
Logical Address	Die Adresse kann in deutscher oder internationaler Mnemonik angegeben sein.
Comment	Frei wählbarer Kommentar
Hmi Visible	Als Wert wird TRUE oder FALSE erwartet.
Hmi Accessible	Als Wert wird TRUE oder FALSE erwartet.
Hmi Writeable	Als Wert wird TRUE oder FALSE erwartet.

Siehe auch

Grundlagen zum Exportieren und Importieren von PLC-Variablen (Seite 7402)

Exportieren von PLC-Variablen (Seite 7405)

Importieren von PLC-Variablen (Seite 7406)

11.10.7.3 Format der Exportdatei (*.xml)

Einführung

Beim Export von PLC-Variablentabellen wird ein standardisiertes xml-Format erzeugt. Dieses Format wird auch beim Import von xml-Dateien erwartet.

Beispiel einer xml-Exportdatei

Das folgende Beispiel zeigt eine PLC-Variablentabelle, die in das Format *.xml exportiert wurde:

```
1  <?xml version='1.0' encoding='utf-8'?>
2  <Tagtable name='Default tag table'>
3    <!-- exported by 'Totally Integrated Automation Portal' version 15.0.1.0 ®
     2004-2016 Siemens AG - all rights reserved -->
4    <Tag type='Bool' hmiVisible='True' hmiWriteable='True' hmiAccessible='True'
     retain='False' remark='' addr='%I0.0'>myTag</Tag>
5    <Constant type='' remark='' value=''>myConstant</Constant>
6  </Tagtable>
```

11.10.7.4 Format der Exportdatei (*.sdf)

Einführung

Beim Export von PLC-Variablentabellen wird ein standardisiertes sdf-Format erzeugt. Dieses Format wird auch beim Import von sdf-Dateien erwartet.

Beispiel einer sdf-Exportdatei

Das folgende Beispiel zeigt eine PLC-Variablentabelle, die in das Format *.sdf exportiert wurde:

```
"Tag_1","%I0.0","Bool","True","True","False","Comment","","True"
"Tag_2","%I0.1","Bool","True","True","False","","","True"
"Const_1","CONSTANT","Bool","","","","Comment1","TRUE",""
"Const_2","CONSTANT","Bool","","","","","TRUE",""
```

11.10.7.5 Exportieren von PLC-Variablen

Voraussetzung

Eine PLC-Variablentabelle ist geöffnet.

Vorgehen

Um PLC-Variablen und Konstanten zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der PLC-Variablentabelle auf die Schaltfläche "Exportieren".
   Der Dialog "Export" wird geöffnet.
2. Klicken Sie unter "Pfad der Exportdatei" auf die Auswahlschaltfläche.
   Der Dialog "Speichern unter" öffnet sich.
3. Geben Sie den Namen der Exportdatei an und wählen Sie den Pfad, in dem sie abgelegt werden soll.
4. Wählen Sie als Dateityp *.xlsx, *.sdf oder *.xml aus.
5. Klicken Sie auf die Schaltfläche "Speichern".
6. Wählen Sie nun die Exportoptionen aus:
   - Wählen Sie aus, ob Sie Variablen und/oder Konstanten exportieren wollen.
   - Wählen Sie aus, ob Sie alle Elemente der Tabelle exportieren wollen oder nur diejenigen, die tatsächlich im Programm auf der aktuellen CPU verwendet werden.
7. Klicken Sie auf die Schaltfläche "OK".

Ergebnis

Die Exportdatei wird erzeugt. Fehler und Warnungen, die während des Exports aufgetreten sind, werden im Inspektorfenster im Register "Info" angezeigt.

Siehe auch

Grundlagen zum Exportieren und Importieren von PLC-Variablen (Seite 7402)

Format der Exportdatei (*.xlsx) (Seite 7403)

Importieren von PLC-Variablen (Seite 7406)

11.10.7.6 Importieren von PLC-Variablen

Voraussetzung

Eine PLC-Variablentabelle ist geöffnet.

Vorgehen

Um PLC-Variablen zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Importieren".
   Der Dialog "Import" wird geöffnet.
2. Wählen Sie die Datei aus, die Sie importieren möchten.
3. Geben Sie an, ob Sie Variablen und/oder Konstanten importieren wollen.
4. Klicken Sie auf die Schaltfläche "OK".

Ergebnis

Die Variablen aus der Importdatei werden in das TIA Portal importiert.

Dabei werden sie in die PLC-Variablentabelle eingefügt, die den gleichen Namen hat wie die Importdatei. Bereits bestehende Einträge werden überschrieben, wenn sie denselben Namen haben wie zu importierende Einträge. Falls es noch keine PLC-Variablentabelle mit dem Namen der Importdatei im Projekt gibt, wird sie neu erstellt und die Variablen werden eingefügt.

Wenn im Projekt bereits Verweise auf PLC-Variablen oder Konstanten vorhanden sind, werden diese Verweise beim Import automatisch aktualisiert. Die Aktualisierung wird auf Basis des Namens der PLC-Variablen und Konstanten durchgeführt.

Fehler und Warnungen, die während des Imports aufgetreten sind, werden im Importprotokoll angezeigt.

Siehe auch

Grundlagen zum Exportieren und Importieren von PLC-Variablen (Seite 7402)

Format der Exportdatei (*.xlsx) (Seite 7403)

Exportieren von PLC-Variablen (Seite 7405)

11.10.7.7 Einzelne PLC-Variablen mit externen Editoren bearbeiten

Um einzelne PLC-Variablen in externen Editoren außerhalb des TIA-Portals zu bearbeiten, können Sie sie per Copy & Paste exportieren und wieder importieren. Strukturierte Variablen können Sie allerdings nicht in einen Editor kopieren.

Voraussetzung

Eine PLC-Variablentabelle und ein externer Editor sind geöffnet.

Vorgehen

Um einzelne PLC-Variablen zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere PLC-Variablen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
4. Bearbeiten Sie die Variablen nach Bedarf.
5. Kopieren Sie die Variablen im externen Editor.
6. Wechseln Sie wieder in die PLC-Variablentabelle.
7. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Hinweis

Sie haben zusätzlich die Möglichkeit, PLC-Variablen als Massendaten zu exportieren oder zu importieren.

Siehe auch: PLC-Variablen exportieren und importieren (Seite 7402)

PLC-Datentypen (UDT) deklarieren 12

12.1 Aufbau der Deklarationstabelle für PLC-Datentypen

Aufbau der Deklarationstabelle für PLC-Datentypen

Das folgende Bild zeigt den Aufbau der Deklarationstabelle für PLC-Datentypen.

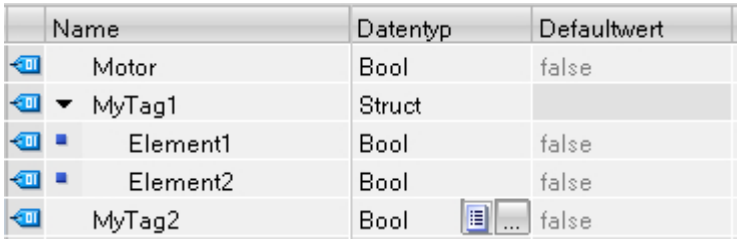

Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie.

Spalte	Erläuterung
	Symbol, das Sie anklicken können, um die Variable zu verschieben oder zu kopieren.
Name	Name der Variablen.
Datentyp	Datentyp der Variablen.
Defaultwert	Wert, mit dem Sie die Variable bei der Deklaration des PLC-Datentyps vorbelegen. Die Angabe des Defaultwerts ist optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert..
Sichtbar in HMI Engineering	Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.
Erreichbar aus HMI / OPC UA	Zeigt an, ob HMI / OPC UA zur Laufzeit auf diese Variable zugreifen kann.
Schreibbar aus HMI / OPC UA	Zeigt an, ob die Variable aus HMI / OPC UA zur Laufzeit geschrieben werden kann.
Einstellwert	Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Nach der Inbetriebnahme können die Werte dieser Variablen als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.
Kommentar	Kommentar zur Dokumentation der Variablen.

Siehe auch

PLC-Datentypen anlegen (Seite 7412)
Tabellenspalten ein- und ausblenden (Seite 7423)
Grundlagen zu PLC-Datentypen (UDT) (Seite 286)
Einstellwerte während der Inbetriebnahme nutzen (Seite 7365)

12.2 PLC-Datentypen in der Projektnavigation

Ordner "PLC-Datentypen"

Die Projektnavigation enthält einen Ordner "PLC-Datentypen", in dem Sie folgende PLC-Datentypen (UDT) anlegen und verwalten können:

- Selbst erzeugte PLC-Datentypen
  PLC-Datentypen, die Sie selbst erzeugen, liegen direkt im Ordner und können dort geöffnet, gelöscht oder umbenannt werden.
- Systemdatentypen
  Vom System erzeugte PLC-Datentypen liegen im Unterordner "Systemdatentypen". Dieser Ordner wird erstellt, wenn Sie zum ersten Mal einen Datenbaustein erzeugen, der auf einem Systemdatentypen basiert, oder eine Anweisung in das Programm ziehen, die einen Systemdatentypen verwendet. Wenn der Systemdatentyp nicht mehr im Programm verwendet wird, wird er während des Übersetzungslaufs automatisch gelöscht. Der Ordner "Systemdatentypen" wird gelöscht, wenn er keine Datentypen mehr enthält.

Hinweis

Sichtbarkeit von Systemdatentypen in der Projektnavigation

Beachten Sie, dass nicht alle Systemdatentypen in der Projektnavigation sichtbar sind. Interne Systemdatentypen, wie z. B. "IEC_COUNTER" werden nicht im Ordner "Systemdatentypen" angezeigt.

Verschieben oder Kopieren von PLC-Datentypen

Sie können PLC-Datentypen zwischen dem Ordner "Systemdatentypen" und anderen Ordnern verschieben oder kopieren. Beachten Sie dabei jedoch die folgenden Regeln:

- Verschieben oder Kopieren von beliebigen PLC-Datentypen in den Ordner "Systemdatentypen":
  PLC-Datentypen im Ordner "Systemdatentypen", die für den Ablauf des Anwenderprogramms nicht erforderlich sind, werden beim nächsten Übersetzungsvorgang entfernt. Dadurch kann es sein, dass Ihr verschobener PLC-Datentyp gelöscht wird.
- Verschieben von Datentypen aus dem Ordner "Systemdatentypen":
  Systemdatentypen können aus dem Ordner "Systemdatentypen" in den darüberliegenden Ordner "PLC-Datentypen" verschoben aber nicht kopiert werden. Datentypen, die nicht im Ordner "Systemdatentypen" liegen, werden jedoch nicht vom System verwaltet. Sie werden also nicht automatisch gelöscht, wenn Sie im Programm nicht mehr verwendet werden.

Siehe auch

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

PLC-Datentypen anlegen (Seite 7412)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

12.3 PLC-Datentypen anlegen

Voraussetzung

Der Ordner "PLC-Datentypen" ist in der Projektnavigation geöffnet.

Vorgehen

Um einen PLC-Datentyp anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".
   Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Selektieren Sie den PLC-Datentyp und wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen Namen für den PLC-Datentyp ein.

Der neue PLC-Datentyp wird erzeugt. Sie finden den PLC-Datentyp in der Projektnavigation im Ordner "PLC-Datentypen".

PLC-Datentyp kommentieren

Um für einen PLC-Datentyp einen Kommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den PLC-Datentyp mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des PLC-Datentyps wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Information".
4. Geben Sie im Eingabefeld "Kommentar" den Kommentar ein.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

12.4 PLC-Datentypen löschen

Voraussetzung

Der PLC-Datentyp, den Sie löschen möchten, ist nicht geöffnet.

Vorgehen

Um einen PLC-Datentyp zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Datentypen".
2. Selektieren Sie den zu löschenden PLC-Datentyp. Sie können auch mehrere PLC-Datentypen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste den ersten und den letzte Datentyp anklicken.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Wenn Sie einen PLC-Datentyp löschen, werden die Bausteine, die den Datentyp verwenden, inkonsistent. Diese Inkonsistenzen werden im verwendenden Baustein rot markiert. Um die Inkonsistenzen zu beheben, müssen Sie die Bausteine aktualisieren.

Siehe auch

Bausteinschnittstelle aktualisieren (Seite 7314)

Datenbausteine aktualisieren (Seite 7326)

Grundlagen zu PLC-Datentypen (UDT) (Seite 286)

12.5 PLC-Datentypen umnummerieren

Aus Performancegründen werden PLC-Datentypen intern mit Nummern verarbeitet. Falls es zu Nummernkonflikten kommt, werden diese automatisch aufgelöst. Dies ist jedoch nicht für einen PLC-Datentypen möglich, der von einem know-how-geschützten Baustein verwendet wird. Durch das Ändern der Nummer des PLC-Datentyps muss der Baustein neu übersetzt werden, was bei einem know-how-geschützten Baustein zu einer Passwortabfrage führt. Dies können Sie umgehen, wenn Sie sich für Ihre PLC-Datentypen ein eigenes Nummernschema einrichten. Verwenden Sie dabei Nummern größer 5000.

Vorgehen

Um die voreingestellte Nummer eines PLC-Datentyps zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektbibliothek in der Task Card "Bibliotheken".
2. Ziehen Sie den übersetzbaren PLC-Datentypen per Drag & Drop in den Ordner "Typen".
   Der Dialog "Typ hinzufügen" wird geöffnet.
3. Geben Sie die Eigenschaften des neuen Typen ein.
4. Bestätigen Sie mit "OK".
5. Klicken Sie den PLC-Datentypen in der Projektbibliothek mit der rechten Maustaste an und wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".
6. Bestätigen Sie die Auswahl der Instanz mit "OK".
   Die Bibliotheksansicht wird geöffnet.
7. Schließen Sie die Bibliotheksansicht.
   Hinter dem Namen des PLC-Datentyps steht jetzt der Zusatz "im Test".
8. Klicken Sie den PLC-Datentypen mit der rechten Maustaste an und wählen Sie im Kontextmenü den Befehl "Eigenschaften".
9. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
   Die Nummer des PLC-Datentyps ist jetzt editierbar.
10. Ändern Sie die Nummer des PLC-Datentyps.
11. Bestätigen Sie mit "OK".
12. Klicken Sie den PLC-Datentypen in der Projektbibliothek mit der rechten Maustaste an und wählen Sie im Kontextmenü den Befehl "Version freigeben".
    Der PLC-Datentyp hat eine neue Nummer. Die vergebene Nummer bleibt auch erhalten, wenn die Typisierung des PLC-Datentyps aufgehoben wird.

12.6 Struktur von PLC-Datentypen programmieren

12.6.1 Variablen von elementarem Datentyp deklarieren

Voraussetzung

Ein PLC-Datentyp ist geöffnet.

Vorgehen

Um eine Variable zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
3. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.
4. Wiederholen Sie die Schritte 1 bis 3 für alle zu deklarierenden Variablen.

Siehe auch

Deklarationstabelle für PLC-Datentypen bearbeiten (Seite 7422)

12.6.2 Variablen vom Datentyp ARRAY deklarieren

Voraussetzung

Ein PLC-Datentyp ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den Datentyp "Array" ein. Sie werden dabei von der Autovervollständigung unterstützt.
   Der Dialog "Array" wird geöffnet.
3. Geben Sie im Eingabefeld "Datentyp" den Datentyp der Array-Elemente an.

4. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.
   Beispiel für ein eindimensionales ARRAY:
   `[0..3]`
   Beispiel für ein dreidimensionales ARRAY:
   `[0..3, 0..15, 0..33]`
5. Bestätigen Sie die Eingabe.
6. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.

Die Variable wird angelegt, bleibt jedoch zugeklappt. Um das ARRAY aufzuklappen, klicken Sie auf das Dreieck vor der Variablen. Beachten Sie, dass ARRAYs mit mehr als 10000 Elementen aus Gründen der Übersichtlichkeit nicht aufklappbar sind.

Defaultwerte von ARRAY-Elementen eingeben

Um die einzelnen Elemente eines ARRAYs mit Defaultwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Defaultwert" ein.

ARRAYs kommentieren

Um einen Kommentar für das gesamte ARRAY anzugeben, tragen Sie diesen in die oberste Zeile der ARRAY-Deklaration ein. Der Kommentar wird als Voreinstellung für alle unterlagerten ARRAY-Elemente verwendet.

Um die einzelnen Elemente eines ARRAYs mit spezifischen Kommentaren zu versehen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.
   Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Kommentar" ein.

ARRAYs im erweiterten Modus anzeigen

In der Anzeige "Erweiterter Modus" werden alle ARRAYs per Voreinstellung aufgeklappt dargestellt.

Um den erweiterten Modus einzuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Erweiterter Modus" in der Funktionsleiste.

Hinweis

Einschränkung in der Anzeige "Erweiterter Modus"

Abhängig der Größe des Arbeitsspeichers Ihres Computers gelten für den erweiterten Modus folgende Maximalgrenzen:

- Arbeitsspeicher < 8 GB: Es werden maximal 10.000 Zeilen angezeigt.
- Arbeitsspeicher >= 8 GB: Es werden maximal 40.000 Zeilen angezeigt.

Siehe auch

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

ARRAY (Seite 296)

Autovervollständigung verwenden (Seite 7147)

12.6.3 Variablen vom Datentyp STRUCT deklarieren

Voraussetzung

Ein PLC-Datentyp ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
   Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
3. Fügen Sie in der leeren Zeile das erste Strukturelement ein.
   Nach dem Element wird erneut eine leere Zeile eingefügt.
4. Wählen Sie einen Datentyp für das Strukturelement aus.
5. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der angezeigt werden.
6. Wiederholen Sie Schritt 3 bis 5 für alle weiteren Strukturelemente.
   Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
7. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.

Siehe auch

Grundlagen zu STRUCT (Seite 291)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

Autovervollständigung verwenden (Seite 7147)

12.6.4 Variablen basierend auf einem anderen PLC-Datentyp deklarieren

Voraussetzung

- Ein globaler Datenbaustein ist geöffnet.
- In der aktuellen CPU ist ein PLC-Datentyp deklariert.

Vorgehen

Um eine Variable basierend auf einem anderen PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
   Die Variable wird angelegt.
3. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Tabelle angezeigt werden.
   Wenn Sie bei der Dekaration des PLC-Datentyps bereits Defaultwerte oder Kommentare für die Variablen innerhalb eines PLC-Datentyps definiert haben, werden diese grau angezeigt. Sie können diese Werte hier ändern.
   Die geänderten Werte werden schwarz dargestellt und gelten nur für die spezifische Verwendungsstelle.

Hinweis

Wenn Sie PLC-Datentypen ändern oder löschen, die im Datenbaustein verwendet werden, wird der Datenbaustein inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie das Programm neu übersetzen.

Siehe auch

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

Autovervollständigung verwenden (Seite 7147)

12.7 Eigenschaften von Variablen in PLC-Datentypen bearbeiten

12.7.1 Eigenschaften von Variablen in PLC-Datentypen

Eigenschaften

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von Variablen in PLC-Datentypen:

Gruppe	Eigenschaft	Beschreibung
Allgemein	Name	Name der Variablen.
	Datentyp	Datentyp der Variablen.
	Defaultwert	Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde. Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt.
	Startwert	In PLC-Datentypen nicht relevant
	Kommentar	Kommentar zur Variablen.
Attribute	Remanenz	In PLC-Datentypen nicht relevant
Anwenderdefinierte Attribute	CFC_Visible	Sichtbar Kennzeichnet, ob ein Parameter im CFC sichtbar ist.
	CFC_Configurable	Parametrierbar Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.
	CFC_ForTest	Für Test Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.
	CFC_Interconnectable	Verschaltbar Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.
	CFC_EnableTagReadback	Rücklesbarkeit für Variablen aktivieren Kennzeichnet, ob ein Parameter für die Funktion "Plan rücklesen" in CFC relevant ist.
	CFC_Enumeration-Texts	Enumerationstexte Ordnet einen Parameter einer Enumeration in CFC zu.
	CFC_EngineeringUnit	Engineering Einheit Ordnet einen Parameter einer Einheit in CFC zu.
	CFC_LowLimit	Untergrenze Definiert den unteren Grenzwert für den Parameter in CFC.
	CFC_HighLimit	Obergrenze Definiert den oberen Grenzwert für den Parameter in CFC.

Siehe auch

Eigenschaften von Variablen in PLC-Datentypen ändern (Seite 7420)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 7409)

12.7.2 Eigenschaften von Variablen in PLC-Datentypen ändern

Allgemeine Eigenschaften in der Deklarationstabelle bearbeiten

Um allgemeine Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den PLC-Datentyp.
2. Ändern Sie die Einträge in den Spalten.

Detaillierte Eigenschaften im Eigenschaftsfenster bearbeiten

Um detaillierte Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Das Inspektorfenster zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".
3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

Siehe auch

Bausteinschnittstelle aktualisieren (Seite 7314)

Datenbausteine aktualisieren (Seite 7326)

12.7.3 Mehrsprachige Kommentare im PLC-Datentyp bearbeiten

Sie können die Kommentare aus dem PLC-Datentyp in allen Projektsprachen bearbeiten, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Beachten Sie, dass die Texte auch nach der Übersetzung die Länge von 32767 Unicode-Zeichen nicht überschreiten dürfen.

Voraussetzung

- Sie haben mehrere Projektsprachen aktiviert.
- Der PLC-Datentyp ist geöffnet und enthält mindestens einen Kommentar.
- Der Datentyp trägt keinen Know-how-Schutz.

Vorgehen

Um einen Kommentar in allen Projektsprachen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie den Kommentar an, dessen Übersetzung Sie bearbeiten wollen.
   Wenn Sie die Übersetzungen mehrerer Kommentare gleichzeitig bearbeiten wollen, markieren Sie alle gewünschten Kommentare.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das untergeordnete Register "Texte".
   Im Register "Texte" werden die markierten Kommentare in den aktiven Projektsprachen und, soweit vorhanden, die Übersetzungen der Kommentare angezeigt.
4. Sie können die Übersetzungen direkt in die Tabelle im Register "Texte" bearbeiten.
5. Um die Übersetzungen mit einem externen Editor zu bearbeiten, können Sie die angezeigten Texte über die Schaltflächen "Projekttexte exportieren/importieren" in das Format OOXML exportieren.

Hinweis

Alle Projekttexte in der globalen Tabelle "Projekttexte" bearbeiten

Sie können die Übersetzungen für die einzelnen Projektsprachen auch in der globale Tabelle "Projekttexte" bearbeiten. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte". Sie enthält alle übersetzbaren Texte des gesamten Projekts.

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

12.8 Deklarationstabelle für PLC-Datentypen bearbeiten

12.8.1 Tabellenzeilen einfügen

Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

12.8.2 Tabellenzeilen hinzufügen

Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

12.8.3 Variablen löschen

Vorgehen

Um Elemente zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit dem zu löschenden Element. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Bausteinschnittstelle aktualisieren (Seite 7314)

Datenbausteine aktualisieren (Seite 7326)

12.8.4 Aufeinander folgende Zellen automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

Voraussetzung

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.
   Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.
   Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

12.8.5 Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
5. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.

KOP-Programme erstellen 13

13.1 Grundlagen zu KOP

13.1.1 Programmiersprache KOP

Übersicht über die Programmiersprache KOP (Kontaktplan)

KOP ist eine grafische Programmiersprache. Die Darstellung ist Stromlaufplänen nachempfunden.

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält am linken Rand eine Stromschiene, von der die Strompfade ausgehen. Auf den Strompfaden werden die Abfragen der binären Signale in Form von Kontakten angeordnet. Durch serielle Anordnung der Elemente auf einem Strompfad entsteht eine Reihenschaltung, durch Anordnung auf parallelen Zweigen entsteht eine Parallelschaltung. Komplexe Funktionen werden durch Boxen dargestellt.

Beispiel für Netzwerke in KOP

Das folgende Bild zeigt ein KOP-Netzwerk mit zwei Schließerkontakten, einem Öffnerkontakt und einer Spule:

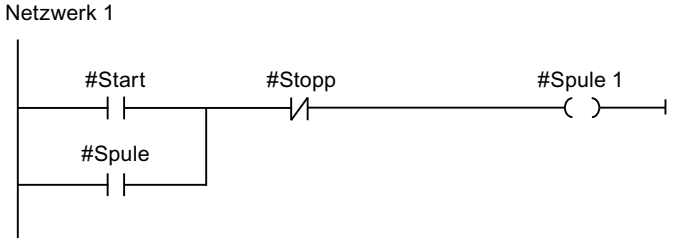

13.1.2 Übersicht über die KOP-Elemente

KOP-Elemente

Ein KOP-Programm besteht aus einzelnen Elementen, die Sie an der Stromschiene eines Netzwerks in Reihe oder parallel zueinander anordnen können. Die meisten Programmelemente müssen mit Variablen versorgt werden.

Von der Stromschiene geht mindestens ein Strompfad aus. Die Programmierung des Netzwerks beginnt am linken Rand des Strompfads. Die Stromschiene können Sie durch mehrere Strompfade und Zweige erweitern.

Das folgende Bild zeigt beispielsweise Elemente eines KOP-Netzwerks:

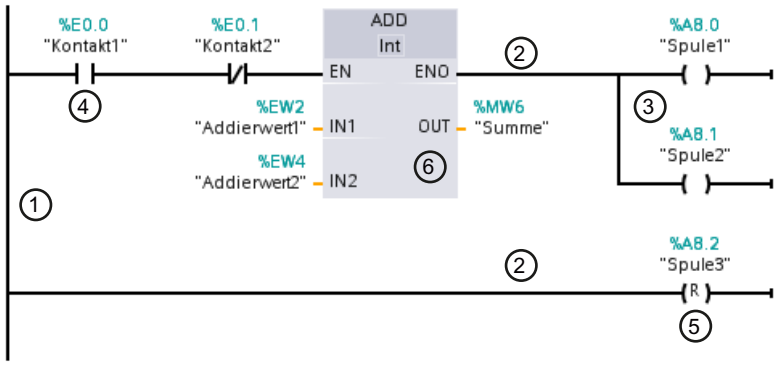

1) Stromschiene

2) Strompfad

3) Zweig

4) Kontakt

5) Spule

6) Box

Stromschiene

Jedes KOP-Netzwerk besteht aus einer Stromschiene, die mindestens einen Strompfad enthält. Ein Netzwerk kann durch Einfügen weiterer Strompfade erweitert werden. In den einzelnen Strompfaden können Sie mithilfe von Zweigen Parallelschaltungen programmieren.

Kontakte

Kontakte dienen dazu, eine stromleitende Verbindung zwischen zwei Elementen herzustellen oder zu unterbrechen. Der Strom wird von links nach rechts weitergeleitet. Mit Kontakten können Sie den Signalzustand oder den Wert eines Operanden abfragen und abhängig vom Ergebnis den Stromfluss kontrollieren.

Die folgenden Arten von Kontakten stehen Ihnen in einem KOP-Programm zur Verfügung:

- Schließerkontakt:
  Schließerkontakte leiten den Strom weiter, wenn der Signalzustand eines angegebenen Binäroperanden "1" ist.
- Öffnerkontakt:
  Öffnerkontakte leiten den Strom weiter, wenn der Signalzustand eines angegebenen Binäroperanden "0" ist.
- Kontakt mit Zusatzfunktion:
  Kontakte mit Zusatzfunktion leiten den Strom weiter, wenn eine bestimmte Bedingung erfüllt ist. Mit diesen Kontakten können Sie auch eine zusätzliche Funktion, z.B. eine Flankenauswertung und einen Vergleich durchführen.

Spulen

Mit Spulen können Sie Binäroperanden steuern. Spulen können einen Binäroperanden abhängig vom Signalzustand des Verknüpfungsergebnisses setzen oder rücksetzen.

Die folgenden Arten von Spulen stehen Ihnen in einem KOP-Programm zur Verfügung:

- Einfache Spulen:
  Einfache Spulen setzen einen Binäroperanden, wenn Strom in die Spule fließt. Beispiel für eine einfache Spule ist die Anweisung "Zuweisung".
- Spulen mit Zusatzfunktion:
  Diese Spulen weisen neben der Auswertung des Verknüpfungsergebnisses weitere Funktionalitäten auf. Beispiele für Spulen mit Zusatzfunktion sind die Spulen zur Flankenauswertung und Programmsteuerung.

Boxen

Boxen sind KOP-Elemente mit komplexen Funktionen. Eine Ausnahme dabei ist die Leerbox. Die Leerbox dient als Platzhalter, in dem Sie die gewünschte Anweisung auswählen können.

Die folgenden Arten von Boxen stehen Ihnen in einem KOP-Programm zur Verfügung:

- Boxen ohne EN-/ENO-Mechanismus:
  Eine Box wird abhängig vom Signalzustand an den Box-Eingängen ausgeführt. Der Fehlerstatus der Bearbeitung kann nicht abgefragt werden.
- Boxen mit EN-/ENO-Mechanismus:
  Eine Box wird nur dann ausgeführt, wenn der Freigabeeingang "EN" den Signalzustand "1" führt. Bei ordnungsgemäßer Bearbeitung der Box führt der Freigabeausgang "ENO" den Signalzustand "1". Wenn während der Bearbeitung ein Fehler auftritt, wird der Freigabeausgang "ENO" zurückgesetzt.

Aufrufe von Codebausteinen werden im Netzwerk auch als Boxen mit EN-/ENO-Mechanismus dargestellt.

Siehe auch

Regeln für die Verwendung von KOP-Elementen (Seite 7439)

EN-/ENO-Mechanismus (Seite 150)

13.2 Einstellungen zu KOP

13.2.1 Übersicht über die Einstellungen zu KOP

Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Schriftart	Schriftgröße	Schriftgröße im Programmiereditor
Ansicht	Layout	Kompakt oder weit Ändert den vertikalen Abstand zwischen Operanden und anderen Objekten (z. B. Operanden und Kontakt). Die Änderung wird erst nach dem erneuten Öffnen eines Bausteins sichtbar.
Operandenfeld	Maximale Breite	Maximale Anzahl der Zeichen, die horizontal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.
	Maximale Höhe	Maximale Anzahl der Zeichen, die vertikal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.

Siehe auch

Einstellungen ändern (Seite 7428)

13.2.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die Einstellungen zu KOP (Seite 7428)

13.3 Arbeiten mit Netzwerken

13.3.1 Netzwerke verwenden

Funktion

Das Anwenderprogramm wird im Baustein innerhalb von Netzwerken erstellt. Damit ein Codebaustein programmiert werden kann, muss er mindestens ein Netzwerk enthalten. Um eine bessere Übersichtlichkeit des Anwenderprogramms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

Sie haben die Möglichkeit, in KOP- und FUP-Bausteinen auch Netzwerke für die textuellen Programmiersprachen SCL und AWL einzufügen und dann Anweisungen in diesen Programmiersprachen zu verwenden. Abhängig von der eingesetzten CPU können Sie Netzwerke für die folgenden Programmiersprachen einfügen:

- S7-300/400: AWL-Netzwerke
- S7-1200: SCL-Netzwerke
- S7-1500: AWL- und SCL-Netzwerke

Der Programmiereditor passt sich immer an das Netzwerk an, das aktuell den Fokus besitzt. Das bedeutet, dass Ihnen z. B. bei der Programmierung eines SCL-Netzwerks die SCL-Anweisungen, aber auch die SCL-Funktionen zur Verfügung stehen.

Hinweis

In SCL-Netzwerken innerhalb von KOP- oder FUP-Bausteinen können Sie die Anweisung "Springen (GOTO)" nicht verwenden.

EN-/ENO-Mechanismus in Bausteinen mit unterschiedlichen Netzwerksprachen

Sie können auch in Bausteinen mit unterschiedlichen Netzwerksprachen den EN-/ENO-Mechanismus einsetzen. Jede Programmiersprache bildet den Fehlerstatus unterschiedlich ab:

- SCL besitzt eine Variable "ENO", die den Fehlerstatus speichert und die Sie abfragen können. Der direkte Zugriff auf diese Variable ist nur mit SCL möglich.
- KOP/FUP/AWL haben keine spezielle Variable für "ENO". Sie können den Fehlerstatus aber für AWL aus dem BIE-Bit auslesen und für KOP/FUP über die RET-Spule abfragen.

Für das Auslesen des Fehlerstatus für den gesamten Baustein gelten die folgenden Regeln:

- Das letzte Netzwerk im Baustein ist ein KOP-/FUP-Netzwerk:
  Falls Sie keine RET-Spule verwenden, ist der Fehlerstatus standardmäßig "TRUE".
- Das letzte Netzwerk im Baustein ist ein AWL-Netzwerk:
  Das BIE-Bit bestimmt den Fehlerstatus. Das BIE-Bit kann in AWL-Netzwerken durch das BIE-Register bearbeitet werden.
- Das letzte Netzwerk im Baustein ist ein SCL-Netzwerk:
  Die Variable "ENO" bestimmt den Fehlerstatus des Bausteins.

Siehe auch

Netzwerke einfügen (Seite 7431)

Netzwerktitel einfügen (Seite 7435)

Netzwerkkommentar eingeben (Seite 7436)

In Netzwerken navigieren (Seite 7437)

13.3.2 Netzwerke einfügen

In einen KOP- oder FUP-Baustein können Sie entweder ein Netzwerk der gleichen Programmiersprache oder ein Netzwerk für eine textuelle Programmiersprache einfügen. Beachten Sie jedoch, dass nicht jeder CPU-Typ alle textuellen Sprachen unterstützt.

Voraussetzung

Ein KOP- oder FUP-Baustein ist geöffnet.

KOP- oder FUP-Netzwerk einfügen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

AWL- oder SCL-Netzwerk einfügen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü entweder den Befehl "AWL-Netzwerk einfügen" oder "SCL-Netzwerk einfügen".

Ergebnis

Es wird ein neues, leeres Netzwerk in der entsprechenden Programmiersprache in den Baustein eingefügt.

Siehe auch

Netzwerke verwenden (Seite 7430)

Netzwerke selektieren (Seite 7432)

Netzwerke kopieren und einfügen (Seite 7433)

Netzwerke löschen (Seite 7433)

Netzwerke reduzieren und erweitern (Seite 7434)

Netzwerktitel einfügen (Seite 7435)

Netzwerkkommentar eingeben (Seite 7436)

In Netzwerken navigieren (Seite 7437)

13.3.3 Netzwerke selektieren

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.
   Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

Siehe auch

Netzwerke verwenden (Seite 7430)

Netzwerke einfügen (Seite 7431)

Netzwerke kopieren und einfügen (Seite 7433)

Netzwerke löschen (Seite 7433)

Netzwerke reduzieren und erweitern (Seite 7434)

Netzwerktitel einfügen (Seite 7435)

Netzwerkkommentar eingeben (Seite 7436)

In Netzwerken navigieren (Seite 7437)

13.3.4 Netzwerke kopieren und einfügen

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

Netzwerke verwenden (Seite 7430)
Netzwerke einfügen (Seite 7431)
Netzwerke selektieren (Seite 7432)
Netzwerke löschen (Seite 7433)
Netzwerke reduzieren und erweitern (Seite 7434)
Netzwerktitel einfügen (Seite 7435)
Netzwerkkommentar eingeben (Seite 7436)
In Netzwerken navigieren (Seite 7437)

13.3.5 Netzwerke löschen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Netzwerke verwenden (Seite 7430)
Netzwerke einfügen (Seite 7431)
Netzwerke selektieren (Seite 7432)
Netzwerke kopieren und einfügen (Seite 7433)
Netzwerke reduzieren und erweitern (Seite 7434)
Netzwerktitel einfügen (Seite 7435)
Netzwerkkommentar eingeben (Seite 7436)
In Netzwerken navigieren (Seite 7437)

13.3.6 Netzwerke reduzieren und erweitern

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

Alle Netzwerke auf- und zuklappen

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

Siehe auch

Netzwerke verwenden (Seite 7430)
Netzwerke einfügen (Seite 7431)
Netzwerke selektieren (Seite 7432)
Netzwerke kopieren und einfügen (Seite 7433)
Netzwerke löschen (Seite 7433)
Netzwerktitel einfügen (Seite 7435)

Netzwerkkommentar eingeben (Seite 7436)

In Netzwerken navigieren (Seite 7437)

13.3.7 Netzwerktitel einfügen

Der Netzwerktitel ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitels ist auf eine Zeile beschränkt. Sie können den Titel entweder manuell eingeben oder automatisch setzen. Beim automatischen Setzen können Sie dies für einzelne Netzwerke vornehmen oder über die Einstellungen festlegen, dass der Netzwerktitel grundsätzlich automatisch gesetzt werden soll.

Für das automatische Einfügen des Netzwerktitels wird der Kommentar des Operanden einer der folgenden Anweisungen im Netzwerk ausgewertet:

- Zuweisung
- Ausgang setzen
- Ausgang rücksetzen

Dabei wird die Anweisung verwendet, die im Netzwerk zuerst steht.

Der Netzwerktitel wird nur dann automatisch eingefügt, wenn die folgenden Bedingungen erfüllt sind:

- Das Netzwerk besitzt noch keinen Titel.
- Der Operand der Anweisung, die für den Kommentar verwendet wird, besitzt einen Kommentar.

Hinweis

Beachten Sie die folgenden Einschränkungen beim automatischen Einfügen des Netzwerktitels:

- Wenn Sie den Kommentar des Operanden nachträglich ändern, wird der Netzwerktitel nicht angepasst.
- Wenn Sie den Operanden der Anweisung wechseln, wird der Netzwerktitel nicht angepasst.
- Der Netzwerktitel wird nur durch die oben genannten schreibenden Anweisungen gesetzt.
- Ist der Operand vom Datentyp Array, wird der Kommentar des Arrays verwendet und nicht die Kommentare der Array-Elemente.
- Kommentare ungültiger Operanden werden nicht berücksichtigt.

Netzwerktitel manuell eingeben

Um einen Netzwerktitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelzeile des Netzwerks.
2. Geben Sie den Netzwerktitel ein.

Netzwerktitel automatisch setzen

Um festzulegen, dass die Netzwerktitel grundsätzlich automatisch gesetzt werden sollen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Aktivieren Sie in der Gruppe "Weitere Einstellungen" das Optionskästchen "Netzwerktitel automatisch setzen".
   Die Titel der Netzwerke werden ab diesem Zeitpunkt automatisch gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Um einen einzelnen Netzwerktitel automatisch zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste eines Netzwerks mit der rechten Maustaste auf "Netzwerk <Nummer des Netzwerks>".
2. Wählen Sie im Kontextmenü den Befehl "Netzwerktitel automatisch setzen".
   Der Titel des selektierten Netzwerks wird anhand des Kommentars des Operanden gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Siehe auch

Netzwerke verwenden (Seite 7430)

Netzwerke einfügen (Seite 7431)

Netzwerke selektieren (Seite 7432)

Netzwerke kopieren und einfügen (Seite 7433)

Netzwerke löschen (Seite 7433)

Netzwerke reduzieren und erweitern (Seite 7434)

Netzwerkkommentar eingeben (Seite 7436)

In Netzwerken navigieren (Seite 7437)

13.3.8 Netzwerkkommentar eingeben

Mithilfe von Netzwerkkommentaren können Sie die Programminhalte der einzelnen Netzwerke erläutern. Sie können z. B. die Funktion des Netzwerks beschreiben oder auf Besonderheiten hinweisen.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um einen Netzwerkkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel.
2. Wenn der Kommentarbereich nicht sichtbar ist, klicken Sie in der Funktionsleiste auf die Schaltfläche "Netzwerkkommentare ein/aus".
   Der Kommentarbereich wird angezeigt.
3. Klicken Sie im Kommentarbereich auf "Kommentar".
   Die Textstelle "Kommentar" wird markiert.
4. Geben Sie den Netzwerkkommentar ein.

Siehe auch

Netzwerke verwenden (Seite 7430)

Netzwerke einfügen (Seite 7431)

Netzwerke selektieren (Seite 7432)

Netzwerke kopieren und einfügen (Seite 7433)

Netzwerke löschen (Seite 7433)

Netzwerke reduzieren und erweitern (Seite 7434)

Netzwerktitel einfügen (Seite 7435)

In Netzwerken navigieren (Seite 7437)

13.3.9 In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".
   Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

Siehe auch

Netzwerke verwenden (Seite 7430)

Netzwerke einfügen (Seite 7431)

Netzwerke selektieren (Seite 7432)

Netzwerke kopieren und einfügen (Seite 7433)

Netzwerke löschen (Seite 7433)

Netzwerke reduzieren und erweitern (Seite 7434)

Netzwerktitel einfügen (Seite 7435)

Netzwerkkommentar eingeben (Seite 7436)

13.4 KOP-Elemente einfügen

13.4.1 Regeln für die Verwendung von KOP-Elementen

Regeln

Beachten Sie beim Einfügen von KOP-Elementen folgende Regeln:

- Jedes KOP-Netzwerk muss einen Abschluss in Form einer Spule oder einer Box haben. Folgende KOP-Elemente dürfen Sie jedoch nicht als Abschluss eines Netzwerks verwenden:
  - Vergleicherboxen
  - Anweisungen für positive oder negative Flankenauswertung
- Der Ausgangspunkt des Zweigs für einen Boxanschluss muss immer die Stromschiene sein. Im Zweig vor der Box dürfen jedoch logische Verknüpfungen oder andere Boxen vorhanden sein.
- In Parallelzweigen mit Vorverknüpfungen können nur Kontakte eingefügt werden. Dabei macht der Kontakt zum Negieren des Verknüpfungsergebnisses (-|NOT|-) eine Ausnahme. Der Kontakt zum Negieren des Verknüpfungsergebnisses sowie Spulen und Boxen können in Parallelzweigen verwendet werden, wenn diese direkt von der Stromschiene ausgehen.
- Öffner- und Schließerkontakte können nicht mit Konstanten belegt werden, z. B. mit TRUE oder FALSE. Verwenden Sie stattdessen Operanden vom Datentyp BOOL.
- Pro Netzwerk kann nur eine Sprunganweisung eingefügt werden.
- Pro Netzwerk kann nur eine Sprungmarke eingefügt werden.
- Anweisungen mit positiver oder negativer Flankenauswertung dürfen nicht direkt am linken Rand des Strompfads angeordnet werden, da diese eine Vorverknüpfung voraussetzen.

Platzierungsregeln für S7-1200/1500 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

Anweisung		Vorverknüpfung notwendig
Mnemonik	Name	
SET_BF	Bitfeld setzen	Nein
RESET_BF	Bitfeld rücksetzen	Nein
JMP	Springen bei VKE = 1	Nein
JMPN	Springen bei VKE = 0	Ja
JMP_LIST	Sprungliste definieren	Nein
SWITCH	Sprungverteiler	Nein
RET	Zurück springen	Nein

Platzierungsregeln für S7-300/400 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

Anweisung		Vorverknüpfung notwendig
Mnemonik	Name	
S	Ausgang setzen	Ja
R	Ausgang rücksetzen	Ja
SI	Zeit als Impuls starten	Ja
SV	Zeit als verlängerten Impuls starten	Ja
SE	Zeit als Einschaltverzögerung starten	Ja
SS	Zeit als speichernde Einschaltverzögerung starten	Ja
SA	Zeit als Ausschaltverzögerung starten	Ja
SZ	Zähleranfangswert setzen	Ja
ZV	Vorwärts zählen	Ja
ZR	Rückwärts zählen	Ja
JMP	Springen bei VKE = 1	Nein
JMPN	Springen bei VKE = 0	Ja
RET	Zurück springen	Nein
OPN	Globalen Datenbaustein öffnen	Nein
OPNI	Instanz-Datenbaustein öffnen	Nein
CALL	Baustein aufrufen	Nein
SAVE	VKE im BIE-Bit speichern	Nein
MCRA	MCR-Bereich aktivieren	Nein
MCRD	MCR-Bereich deaktivieren	Nein
MCR<	MCR-Bereiche öffnen	Nein
MCR>	MCR-Bereiche schließen	Nein

Hinweis

Aufruf von Bausteinen mit binären Eingängen in S7-300/400 CPUs

Der folgende FAQ beschreibt die Besonderheiten beim Aufruf von Bausteinen mit binären Eingängen:

FAQ 13988019: Was müssen Sie bei der Programmierung von verschachtelten FB/FC-Aufrufen mit binären Eingängen beachten? (Seite 7441)

Siehe auch

Übersicht über die KOP-Elemente (Seite 7425)

FAQ 13988019 (https://support.industry.siemens.com/cs/de/de/view/13988019)

EN-/ENO-Mechanismus (Seite 150)

13.4.2 Unzulässige Verschaltungen in KOP

Stromfluss von rechts nach links

Es können keine Zweige programmiert werden, die einen Stromfluss in umgekehrter Richtung verursachen könnten.

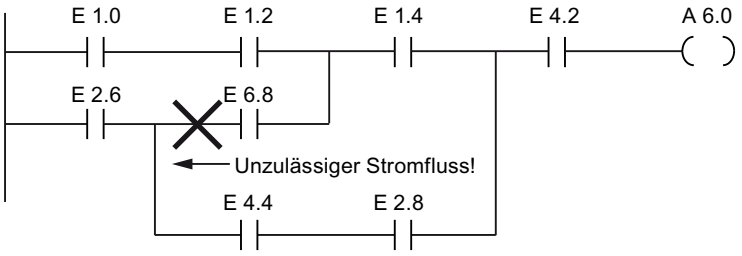

Kurzschluss

Es können keine Zweige programmiert werden, die einen Kurzschluss verursachen.

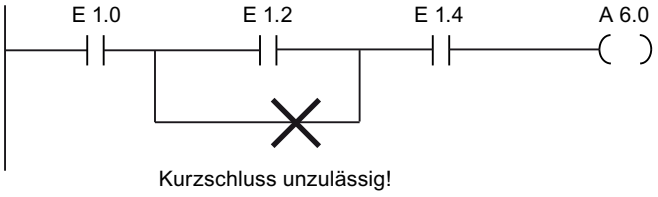

Verknüpfungen

Für Verknüpfungen gelten folgende Regeln:

- Nur boolesche Eingänge sind vorverknüpfbar.
- Nur der erste boolesche Ausgang ist weiter verknüpfbar.
- Es existiert nur ein vollständiger Verknüpfungspfad pro Netzwerk. Nicht zusammenhängende Pfade können verbunden werden.

Siehe auch

Regeln für die Verwendung von KOP-Elementen (Seite 7439)

13.4.3 KOP-Elemente über Task Card "Anweisungen" einfügen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein KOP-Element mithilfe der Task Card "Anweisungen" in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie zu dem KOP-Element, das Sie einfügen möchten.
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Oder:

1. Markieren Sie die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
2. Öffnen Sie die Task Card "Anweisungen".
3. Doppelklicken Sie auf das einzufügende Element.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Ergebnis

Das ausgewählte KOP-Element wird mit Platzhaltern für die Parameter eingefügt.

Siehe auch

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

Parameterinstanzen (Seite 70)

13.4.4 KOP-Elemente mithilfe einer Leerbox einfügen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein KOP-Element mithilfe einer Leerbox in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke der Leerbox. Eine Klappliste wird geöffnet.
5. Wählen Sie das gewünschte KOP-Element aus der Klappliste.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Ergebnis

Die Leerbox wird zum entsprechenden KOP-Element geändert. Für die Parameter werden Platzhalter eingefügt.

Siehe auch

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

Parameterinstanzen (Seite 70)

13.4.5 Datentyp eines KOP-Elements auswählen

13.4.5.1 Auswahl eines Datentyps

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie eine dieser Anweisungen im Programm verwenden, müssen Sie an der konkreten Programmstelle ein für die Anweisung zulässiger Datentyp festlegen. Bei einigen Anweisungen müssen Sie den Datentypen für die Eingänge und für die Ausgänge separat auswählen.

Hinweis

Der zulässige Datentyp (BOOL) für die Variablen am Freigabeeingang EN und am Freigabeausgang ENO ist vom System vorgegeben und kann nicht geändert werden.

Die für eine Anweisung zulässigen Datentypen sind in der Klappliste der Anweisung aufgeführt. Durch die Auswahl eines Eintrags aus der Klappliste bestimmen Sie den Datentyp der Anweisung. Wenn der Datentyp eines Operanden vom Datentyp der Anweisung abweicht und nicht implizit konvertiert werden kann, wird der Operand rot angezeigt und ein Rollout mit der entsprechenden Fehlermeldung eingeblendet.

Datentypauswahl von mathematischen Anweisungen

Einige mathematische Anweisungen bieten Ihnen die Möglichkeit, den Datentyp automatisch passend zu den Datentypen der Operanden einstellen zu lassen. Diese Anweisungen besitzen in der Klappliste für die Datentypauswahl zusätzlich zu den eigentlichen Datentypen noch den Eintrag "Auto". Wenn Sie diesen Eintrag auswählen und den ersten Operanden beschalten, wird der Datentyp des Operanden als Datentyp für die Anweisung verwendet. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)", z. B. "Auto (Real)". Wenn Sie weitere Operanden beschalten, wird der automatisch eingestellte Datentyp der Anweisung nach folgenden Kriterien angepasst:

- Sie versorgen alle weiteren Operanden mit Variablen des gleichen Datentyps:
  Der Datentyp der Anweisung wird nicht geändert.
- Sie versorgen alle weiteren Operanden mit Variablen, deren Datentyp kleiner ist als der Datentyp der Anweisung:
  Der Datentyp der Anweisung wird nicht geändert. Für die Operanden mit dem kleineren Datentyp wird ggf. eine implizite Konvertierung durchgeführt.
- Sie versorgen einen weiteren Operanden mit einer Variablen, deren Datentyp größer ist als der Datentyp der Anweisung:
  Der Datentyp der Anweisung wird auf den größeren Datentyp geändert. Für Operanden, die von dem neu eingestellten Datentyp der Anweisung abweichen, erfolgt ggf. eine implizite Konvertierung.

Jede Datentypänderung eines Operanden kann eine Änderung des Datentyps der Anweisung nach sich ziehen. Dadurch werden möglichweise auch weitere Operanden implizit konvertiert.

Operanden, für die eine implizite Konvertierung durchgeführt wird, werden mit einem grauen Quadrat gekennzeichnet.

Hinweis

Beachten Sie auch die Informationen zur Datentypkonvertierung für Ihr Gerät und hier insbesondere die Hinweise zur IEC-Prüfung.

Siehe auch: Datentypkonvertierung (Seite 507)

Siehe auch

Datentyp einer Anweisung festlegen (Seite 7445)

13.4.5.2 Datentyp einer Anweisung festlegen

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie solche Anweisungen in Ihr Programm einfügen, müssen Sie für diese Anweisungen an der konkreten Programmstelle den Datentyp festlegen.

Datentyp über die Klappliste festlegen

Um den Datentyp einer Anweisung über die Klappliste festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Klicken Sie auf das Dreieck in der oberen Ecke der Klappliste.
   Die Klappliste wird geöffnet und die Datentypen angezeigt, die für die Anweisung zulässig sind.
3. Wählen Sie aus der Klappliste einen Datentyp aus.
   Der ausgewählte Datentyp wird angezeigt.
4. Falls die Anweisung über zwei Klapplisten verfügt, wählen Sie in der linken Klappliste den Datentyp für die Eingänge der Anweisung und in der rechten Klappliste den Datentyp für die Ausgänge der Anweisung.

Datentyp durch Zuweisen von Variablen festlegen

Um den Datentyp einer Anweisung durch Zuweisen von Variablen festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an, deren Datentyp als Datentyp der Anweisung übernommen werden soll.
   Der Datentyp der Variablen wird in der Klappliste angezeigt.
3. Falls sowohl für die Eingänge als auch für die Ausgänge der Anweisung Datentypen festgelegt werden müssen, geben Sie an einem Eingang und an einem Ausgang jeweils eine gültige Variable an. Die am Eingang angegebene Variable bestimmt den Datentyp der Eingänge und die Variable am Ausgang bestimmt den Datentyp der Ausgänge der Anweisung.

Datentyp von mathematischen Anweisungen automatisch festlegen

Um für mathematische Anweisungen den Datentyp automatisch festlegen zu lassen, gehen Sie folgendermaßen vor:

1. Fügen Sie die mathematische Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Wählen Sie in der Klappliste den Eintrag "Auto".
3. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an.
   Der Datentyp der Variablen wird als Datentyp der Anweisung übernommen. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)".

Siehe auch: Auswahl eines Datentyps (Seite 7444)

Siehe auch

Auswahl eines Datentyps (Seite 7444)

13.4.6 Favoriten in KOP verwenden

13.4.6.1 KOP-Elemente zu den Favoriten hinzufügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

KOP-Elemente aus den Favoriten entfernen (Seite 7448)

Übersicht über den Programmiereditor (Seite 7117)

13.4.6.2 KOP-Elemente mithilfe von Favoriten einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

KOP-Elemente aus den Favoriten entfernen (Seite 7448)

Übersicht über den Programmiereditor (Seite 7117)

13.4.6.3 KOP-Elemente aus den Favoriten entfernen

Voraussetzung

Ein Codebaustein ist geöffnet.

Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

KOP-Elemente zu den Favoriten hinzufügen (Seite 7446)

KOP-Elemente mithilfe von Favoriten einfügen (Seite 7447)

Übersicht über den Programmiereditor (Seite 7117)

13.4.7 Bausteinaufrufe in KOP einfügen

13.4.7.1 Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FC) und Funktionsbausteine (FB) per Drag & Drop aus der Projektnavigation einfügen. Wenn Sie Funktionsbausteine aus anderen Funktionsbausteinen aufrufen, können Sie sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Voraussetzung

- Ein Netzwerk ist vorhanden.
- Der Baustein, der aufgerufen werden soll, ist vorhanden.

Aufruf einer Funktion (FC) einfügen

Um den Aufruf einer Funktion (FC) per Drag & Drop in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das gewünschte Netzwerk.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das gewünschte Netzwerk. Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
     Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie im Eingabefeld "ProDiag-FB" den Überwachungen einen ProDiag-Funktionsbaustein zu.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
3. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter können Sie anschließend versorgen.

Siehe auch: Parameterübergabe beim Bausteinaufruf (Seite 78)

Hinweis

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Instanztyp ändern (Seite 7453)

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

Bausteinaufrufe (Seite 62)

13.4.7.2 Bausteinaufrufe in KOP aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben die folgenden Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren aller inkonsistenten Bausteinaufrufe im Programmiereditor.
  Die inkonsistenten Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Explizites Aktualisieren eines Bausteinaufrufs im Programmiereditor.
  Der Dialog "Schnittstellenabgleich" wird angezeigt. In diesem Dialog haben Sie die Möglichkeit, die Beschaltung der Operanden der neuen Schnittstelle zu ändern. Anschließend wird der inkonsistente Aufruf dieses Bausteins aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert. Beachten Sie, dass Sie beim Aufruf von Funktionen (FCs) vor dem nächsten Übersetzungsvorgang alle neuen Formalparameter mit Aktualparametern versorgen müssen.

Alle inkonsistenten Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein alle Bausteinaufrufe zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Einen bestimmten Bausteinaufruf im Programmiereditor aktualisieren

Um einen bestimmten Bausteinaufruf im Programmiereditor zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf, den Sie aktualisieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Bausteinaufruf aktualisieren".
   Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden Ihnen die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.

4. Ändern Sie bei Bedarf die Beschaltung der Operanden. Dazu haben Sie die folgenden Möglichkeiten:
   - Sie können einen Operanden entweder per Drag & Drop oder über Ausschneiden/ Kopieren und Einfügen aus der alten Schnittstelle in die neue Schnittstelle übernehmen.
   - Sie können einen Operanden löschen.
   - Sie können einen Operanden umbenennen.
   - Sie können über die Autovervollständigung einen neuen Operanden angeben.
5. Klicken Sie auf "OK", um den Bausteinaufruf zu aktualisieren. Wenn Sie die Aktualisierung abbrechen möchten, klicken Sie auf "Abbrechen".

Hinweis

Beachten Sie, dass Ihnen der Befehl "Bausteinaufruf aktualisieren" nur zur Verfügung steht, wenn Sie nicht zuvor alle Bausteinaufrufe im Editor über den Befehl "Inkonsistente Bausteinaufrufe aktualisieren" aktualisiert haben.

Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Bausteinaufrufe per Drag & Drop einfügen (Seite 7449)

Instanztyp ändern (Seite 7453)

Bausteinaufrufe (Seite 62)

13.4.7.3 Bausteinaufruf ändern

Sie haben die Möglichkeit, für einen Bausteinaufruf den aufgerufenen Baustein zu ändern. Beachten Sie jedoch, dass dabei keine neuen Instanz-Datenbausteine erzeugt werden, z. B. beim Wechsel von einer Funktion (FC) zu einem Funktionsbaustein (FB).

Vorgehen

Um den aufgerufenen Baustein eines Bausteinaufrufs zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb des Bausteinaufrufs auf den Namen des aufgerufenen Bausteins und drücken Sie die Taste <F2>. Oder klicken Sie doppelt auf den Namen des aufgerufenen Bausteins.
   Ein Eingabefeld wird geöffnet, der Name des aktuell aufgerufenen Bausteins ist markiert.
2. Geben Sie den Namen des Bausteins ein, den Sie aufrufen möchten, oder wählen Sie einen Baustein in der Autovervollständigung aus.
3. Wenn Sie einen FB aufrufen möchten, erstellen Sie ggf. einen neuen Instanz-Datenbaustein und geben Sie ihn als Operand an.

13.4.7.4 Instanztyp ändern

Instanztyp

Funktionsbausteine können Sie auf verschiedene Arten aufrufen:

- Als Einzelinstanz
- Als Multiinstanz
- Als Parameterinstanz

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Sie können einen einmal definierten Instanztyp jederzeit ändern.

Voraussetzung

Das Anwenderprogramm enthält einen Bausteinaufruf.

Vorgehen

Um den Instanztyp eines Funktionsbausteins zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Codebaustein und selektieren Sie den Bausteinaufruf.
2. Wählen Sie im Kontextmenü den Befehl "Instanz ändern".
   Der Dialog "Aufrufoptionen" wird geöffnet.

3. Klicken Sie auf die Schaltfläche "Einzel-Instanz", "Multi-Instanz" oder "Parameter-Instanz".
   - Wenn Sie als Instanztyp "Einzel-Instanz" wählen, geben Sie einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
   - Wenn Sie als Instanztyp "Multi-Instanz" wählen, geben Sie im Textfeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie als Instanztyp "Parameter-Instanz" wählen, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Hinweis

Die vorherigen Instanzen werden nicht automatisch gelöscht.

Siehe auch

Bausteinaufrufe per Drag & Drop einfügen (Seite 7449)

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Instanzen (Seite 64)

Bausteinaufrufe (Seite 62)

13.4.8 Komplexe KOP-Anweisungen einfügen

13.4.8.1 Anweisung "Berechnen" verwenden

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um die Anweisung "Berechnen" zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Mathematische Funktionen > CALCULATE".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.
   Die Anweisung "Berechnen" wird mit einem Platzhalterausdruck und Fragezeichen für den Datentyp eingefügt.
4. Geben Sie den Datentyp für die Berechnung ein.

5. Geben Sie die Operanden für die Berechnung ein.

   Hinweis

   Die Berechnung wird mit den Eingängen der Anweisung "Berechnen" durchgeführt. Wenn Sie Konstanten verwenden möchten, müssen Sie daher auch für die Konstanten entsprechende Eingänge einfügen.

6. Klicken Sie auf die Schaltfläche "Anweisung 'Berechnen' bearbeiten", um den Platzhalterausdruck durch den korrekten Ausdruck zu ersetzen.
   Der Dialog "Anweisung 'Berechnen' bearbeiten" wird geöffnet.
7. Geben Sie in das Textfeld "OUT:= " den gewünschten Ausdruck ein.

   Hinweis

   Im Bereich "Beispiel" finden Sie ein Beispiel für einen gültigen Ausdruck und mögliche Anweisungen, die Sie verwenden können.

   Um z. B. mithilfe des Pythagoras einen Wert zu ermitteln, geben Sie "OUT := SQRT (SQR (IN1) + SQR (IN2))" ein.

8. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

CALCULATE: Berechnen (Seite 752)

13.4.9 Freie Kommentare verwenden

13.4.9.1 Grundlagen zur Verwendung von freien Kommentaren in KOP

Einführung

Freie Kommentare ermöglichen das Kommentieren des Quellcodes für grafische Programmiersprachen ähnlich wie Zeilenkommentare für textuelle Sprachen.

Freie Kommentare können für folgende Elemente verwendet werden:

- Boxen
- Spulen

Programmcode auskommentieren

Um in den Programmiersprachen KOP und FUP einzelne Anweisungen oder ganze Programmteile zu deaktivieren, müssen Sie parallel zu der Anweisung oder in Reihe zu einem Anweisungsblock ein Bit setzen. Das Auskommentieren wie in AWL und SCL ist in KOP und FUP nicht möglich.

Beispiele zum Auskommentieren von Programmcode in allen Programmiersprachen finden Sie im Siemens Industry Online Support unter folgender FAQ ID: 109482004 (https://support.industry.siemens.com/cs/ww/de/view/109482004)

Siehe auch

Freie Kommentare einfügen (Seite 7456)

Freie Kommentare bearbeiten (Seite 7456)

Freie Kommentare löschen (Seite 7458)

13.4.9.2 Freie Kommentare einfügen

Voraussetzung

Ein Netzwerk mit Anweisungen ist vorhanden.

Vorgehen

Um einen freien Kommentar zu einer Anweisung hinzuzufügen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie ggf. in der Funktionsleiste die Schaltfläche "Freie Kommentare ein/aus".
2. Klicken Sie mit der rechten Maustaste auf die Anweisung, für die Sie einen freien Kommentar einfügen möchten.
3. Wählen Sie den Befehl "Kommentar einfügen" aus dem Kontextmenü.
   Eine Kommentarbox mit einem Standardkommentar wird geöffnet. Die Kommentarbox ist über einen Pfeil mit der zugehörigen Anweisung verbunden.
4. Geben Sie in die Kommentarbox den gewünschten Kommentar ein.

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 7455)

Freie Kommentare bearbeiten (Seite 7456)

Freie Kommentare löschen (Seite 7458)

13.4.9.3 Freie Kommentare bearbeiten

Einführung

Freie Kommentare können folgendermaßen bearbeitet werden:

- Kommentartext ändern
- Platzierung oder Größe der Kommentarbox ändern

- Kommentar an ein anderes Element anhängen
- Freie Kommentare ein- und ausblenden

Kommentartext ändern

Um den Text von freien Kommentaren zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Kommentarbox.
2. Geben Sie den gewünschten Text ein.

Platzierung der Kommentarbox ändern

Um die Platzierung der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox und halten Sie die Maustaste gedrückt.
2. Ziehen Sie die Kommentarbox an die gewünschte Stelle.

Größe der Kommentarbox ändern

Um die Größe der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox.
2. Ziehen Sie die Kommentarbox am Anfasser in der unteren rechten Ecke auf die gewünschte Größe.

Kommentar an ein anderes Element anhängen

Um einen freien Kommentar an ein anderes Element zu hängen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Pfeilspitze, die die Kommentarbox mit der Anweisung verbindet, und halten Sie die Maustaste gedrückt.
2. Ziehen Sie den Pfeil auf das Element, an das Sie den Kommentar anhängen möchten. Mögliche Einfügestellen werden mit einem grünen Quadrat gekennzeichnet.
3. Lassen Sie die Maustaste los.

Freie Kommentare ein- und ausblenden

Um freie Kommentare ein- und auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Freie Kommentare ein/aus".

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 7455)

Freie Kommentare einfügen (Seite 7456)

Freie Kommentare löschen (Seite 7458)

13.4.9.4 Freie Kommentare löschen

Vorgehen

Um einen freien Kommentar zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den freien Kommentar, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 7455)
Freie Kommentare einfügen (Seite 7456)
Freie Kommentare bearbeiten (Seite 7456)

13.5 KOP-Elemente bearbeiten

13.5.1 KOP-Elemente selektieren

Sie können mehrere einzelne Elemente oder alle Elemente eines Netzwerks selektieren.

Voraussetzung

KOP-Elemente sind vorhanden.

Mehrere einzelne KOP-Elemente selektieren

Um mehrere einzelne KOP-Elemente zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie die KOP-Elemente an, die Sie selektieren möchten.
3. Lassen Sie die Taste <Strg> los.

Alle KOP-Elemente eines Netzwerks selektieren

Um alle KOP-Elemente eines Netzwerks zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Netzwerk, dessen Elemente Sie selektieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

Siehe auch

KOP-Elemente kopieren (Seite 7460)

KOP-Elemente ausschneiden (Seite 7460)

KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)

KOP-Elemente ersetzen (Seite 7462)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)

Eingänge und Ausgänge entfernen (Seite 7464)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

KOP-Elemente löschen (Seite 7465)

13.5.2 KOP-Elemente kopieren

Voraussetzung

Ein KOP-Element ist vorhanden.

Vorgehen

Um ein KOP-Element zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Ergebnis

Das KOP-Element wird kopiert und in der Zwischenablage gespeichert.

Siehe auch

KOP-Elemente selektieren (Seite 7459)
KOP-Elemente ausschneiden (Seite 7460)
KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)
KOP-Elemente ersetzen (Seite 7462)
Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)
Eingänge und Ausgänge entfernen (Seite 7464)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)
KOP-Elemente löschen (Seite 7465)

13.5.3 KOP-Elemente ausschneiden

Voraussetzung

Ein KOP-Element ist vorhanden.

Ausschneiden

Um ein KOP-Element auszuschneiden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

Ergebnis

Das KOP-Element wird ausgeschnitten und in der Zwischenablage gespeichert.

Siehe auch

KOP-Elemente selektieren (Seite 7459)
KOP-Elemente kopieren (Seite 7460)
KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)
KOP-Elemente ersetzen (Seite 7462)
Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)
Eingänge und Ausgänge entfernen (Seite 7464)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)
KOP-Elemente löschen (Seite 7465)

13.5.4 KOP-Elemente aus der Zwischenablage einfügen

Voraussetzung

Ein KOP-Element ist vorhanden.

Vorgehen

Um ein KOP-Element aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie ein KOP-Element oder schneiden Sie ein KOP-Element aus.
2. Klicken Sie mit der rechten Maustaste auf die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

KOP-Elemente selektieren (Seite 7459)
KOP-Elemente kopieren (Seite 7460)
KOP-Elemente ausschneiden (Seite 7460)
KOP-Elemente ersetzen (Seite 7462)
Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)
Eingänge und Ausgänge entfernen (Seite 7464)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)
KOP-Elemente löschen (Seite 7465)

13.5.5 KOP-Elemente ersetzen

Sie können KOP-Elemente ganz einfach gegen andere KOP-Elemente des gleichen Typs austauschen. Dies hat den Vorteil, dass die Parameter bestehen bleiben und nicht erneut eingegeben werden müssen. Sie können z. B. Schließer und Öffner oder RS-FlipFlop und SR-FlipFlop gegeneinander austauschen.

Voraussetzung

Ein Netzwerk mit mindestens einem KOP-Element ist vorhanden.

Vorgehen

Um ein KOP-Element durch ein anderes KOP-Element zu ersetzen, gehen Sie folgendermaßen vor:

1. Markieren Sie das KOP-Element, das Sie ersetzen möchten.
2. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke des KOP-Elements.
   Eine Klappliste wird geöffnet.
3. Wählen Sie das KOP-Element aus der Klappliste, durch das Sie das aktuelle KOP-Element ersetzen möchten.

Siehe auch

KOP-Elemente selektieren (Seite 7459)

KOP-Elemente kopieren (Seite 7460)

KOP-Elemente ausschneiden (Seite 7460)

KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)

Eingänge und Ausgänge entfernen (Seite 7464)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

KOP-Elemente löschen (Seite 7465)

13.5.6 Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen

Einführung

Durch zusätzliche Eingänge können Sie KOP-Elemente erweitern, die kommutative arithmetische Anweisungen durchführen. Solche Elemente sind z. B. die Anweisungen "Addieren" (ADD) und "Multiplizieren" (MUL). Durch zusätzliche Ausgänge können Sie die Boxen der Anweisungen "Wert kopieren" (MOVE) und "Demultiplexen" (DEMUX) erweitern.

Voraussetzung

Ein KOP-Element ist vorhanden, das das Einfügen von zusätzlichen Eingängen bzw. Ausgängen erlaubt.

Zusätzlichen Eingang einfügen

Um die Box eines KOP-Elements durch einen zusätzlichen Eingang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Eingang des KOP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Eingang einfügen".
   Die Box des KOP-Elements wird durch einen zusätzlichen Eingang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.
   Die Box des KOP-Elements wird durch einen zusätzlichen Eingang erweitert.

Zusätzlichen Ausgang einfügen

Um die Box eines KOP-Elements durch einen zusätzlichen Ausgang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Ausgang des KOP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Ausgang einfügen".
   Die Box des KOP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.
   Die Box des KOP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Siehe auch

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente selektieren (Seite 7459)

KOP-Elemente kopieren (Seite 7460)

KOP-Elemente ausschneiden (Seite 7460)

KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)

KOP-Elemente ersetzen (Seite 7462)

Eingänge und Ausgänge entfernen (Seite 7464)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

KOP-Elemente löschen (Seite 7465)

13.5.7 Eingänge und Ausgänge entfernen

Einführung

Eingänge oder Ausgänge, die Sie für eine Anweisung zusätzlich eingefügt haben, können wieder entfernt werden.

Voraussetzung

Ein KOP-Element ist vorhanden, das Sie durch zusätzliche Ein- oder Ausgänge erweitert haben.

Eingang entfernen

Um einen Eingang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eingang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Eingang des KOP-Elements wird entfernt.

Ausgang entfernen

Um einen Ausgang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ausgang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Ausgang des KOP-Elements wird entfernt.

Siehe auch

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente selektieren (Seite 7459)

KOP-Elemente kopieren (Seite 7460)

KOP-Elemente ausschneiden (Seite 7460)

KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)

KOP-Elemente ersetzen (Seite 7462)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

KOP-Elemente löschen (Seite 7465)

13.5.8 KOP-Elemente löschen

Voraussetzung

Ein KOP-Element ist vorhanden.

Vorgehen

Um ein KOP-Element zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

KOP-Elemente einfügen (Seite 7439)
KOP-Elemente selektieren (Seite 7459)
KOP-Elemente kopieren (Seite 7460)
KOP-Elemente ausschneiden (Seite 7460)
KOP-Elemente aus der Zwischenablage einfügen (Seite 7461)
KOP-Elemente ersetzen (Seite 7462)
Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 7462)
Eingänge und Ausgänge entfernen (Seite 7464)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 153)

13.6 Operanden in KOP-Anweisungen beschalten

13.6.1 Operanden beschalten

Beim Einfügen eines KOP-Elements werden die Zeichenfolgen "<???>", "<??.?>" und "..." als Platzhalter für die Parameter eingefügt. Die rot dargestellten Zeichenfolgen "<???>" und "<??.?>" kennzeichnen Parameter, die beschaltet werden müssen. Die schwarz dargestellte Zeichenfolge "..." kennzeichnet Parameter, die beschaltet werden können. "<??.?>" steht für boolesche Platzhalter. Zudem können KOP-Elemente die folgenden unterschiedlichen Anschlüsse für die Parameter besitzen:

- Orangefarbene Anschlüsse: An diesem Anschluss können nur Variablen oder Konstanten verschaltet werden.
- Schwarze Anschlüsse: Boolescher Ein- oder Ausgang, an dem auch andere Elemente vorverschaltet oder nachgeschaltet werden können.

Hinweis

Wenn Sie den Mauszeiger über den Platzhalter bewegen, wird der erwartete Datentyp angezeigt.

Voraussetzung

Ein KOP-Element ist vorhanden.

Vorgehen

Um die Parameter eines KOP-Elements zu beschalten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Platzhalter des Parameters.
   Ein Eingabefeld wird geöffnet, der Platzhalter ist markiert.
2. Geben Sie den entsprechenden Parameter ein.

   Hinweis

   Wenn Sie die absolute Adresse eines Parameters eingeben, der bereits definiert ist, wird diese Absolutadresse in den symbolischen Namen des Parameters geändert, sobald Sie die Eingabe bestätigen. Haben Sie den Parameter noch nicht definiert, wird in der PLC-Variablentabelle eine neue Variable mit dieser Absolutadresse und dem Standardnamen "Tag_<n>" eingetragen. Bestätigen Sie Ihre Eingabe, wird die Absolutadresse gegen den symbolischen Namen "Tag_<n>" ausgetauscht.

3. Bestätigen Sie den Parameter mit der Eingabetaste.
4. Wenn Sie den Parameter bisher nicht definiert haben, können Sie dies über das Kontextmenü direkt im Programmiereditor erledigen.
   Siehe auch:
   PLC-Variablen im Programmiereditor deklarieren (Seite 7383)
   Lokale Variablen im Programmiereditor deklarieren (Seite 7275)

Oder per Drag & Drop aus der PLC-Variablentabelle:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" oder öffnen Sie die PLC-Variablentabelle.
2. Wenn Sie die PLC-Variablentabelle geöffnet haben, ziehen Sie das Symbol aus der ersten Spalte der gewünschten Variablen an die entsprechende Stelle in Ihrem Programm. Falls Sie die PLC-Variablentabelle nicht geöffnet haben, öffnen Sie die Detailansicht. Ziehen Sie die gewünschte Variable aus der Detailansicht an die entsprechende Stelle in Ihrem Programm.

Oder per Drag & Drop aus der Bausteinschnittstelle:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ziehen Sie den gewünschten Operanden aus der Bausteinschnittstelle in das Anweisungsfenster.

Ergebnis

- Wenn die Syntax fehlerfrei ist, wird der Parameter schwarz dargestellt. Der Editor springt zum nächsten Platzhalter.
- Wenn in der Syntax ein Fehler vorhanden ist, wird das Eingabefeld nicht verlassen und es wird eine entsprechende Fehlermeldung in der Statuszeile angezeigt. Wenn Sie die Eingabetaste erneut drücken, wird das Eingabefeld geschlossen und die fehlerhafte Eingabe wird rot und kursiv dargestellt.

Siehe auch

Operanden verwenden und adressieren (Seite 93)

Autovervollständigung verwenden (Seite 7147)

13.6.2 Versteckte Parameter beschalten

Einführung

Wenn eine Anweisung versteckte Parameter enthält, verfügt die Anweisungsbox über einen kleinen Pfeil am unteren Rand. Die versteckten Parameter erkennen Sie an der weißen Schriftfarbe.

Sie können die versteckten Parameter jederzeit einblenden und beschalten.

Versteckte Parameter ein- oder ausblenden

Um versteckte Parameter ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten am unteren Rand der Anweisungsbox, um versteckte Parameter einzublenden.
2. Klicken Sie auf den Pfeil-nach-oben am unteren Rand der Anweisungsbox, um versteckte Parameter auszublenden.

Versteckte Parameter beschalten

Um verstecke Parameter zu beschalten, gehen Sie folgendermaßen vor:

1. Beschalten Sie die versteckten Parameter wie normale sichtbare Parameter.
   Durch das Beschalten des Parameters kann sich dessen Sichtbarkeit ändern:
   - Der Parameter wird sichtbar, wenn Sie in den Eigenschaften die Option "Verstecken, wenn kein Parameter zugewiesen ist" angewählt haben.
   - Der Parameter bleibt versteckt, wenn Sie in den Eigenschaften die Option "Verstecken" angewählt haben.

Siehe auch

Parameter beim Bausteinaufruf verstecken (Seite 7284)

13.6.3 Variableninformationen ein- oder ausblenden

Einführung

Sie können die folgenden Informationen zu den verwendeten Variablen im Programmiereditor einblenden:

- Name der Variablen
- Adresse der Variablen
- Einfache oder hierarchische Kommentare zur Dokumentation der Variablen

Die Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablentabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden, werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet. Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete Bausteine wieder einblenden.

Wenn Sie die Anzeige von Variableninformationen mit hierarchischen Kommentaren wählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt. Die Anzeige erfolgt dabei in Klammern hinter dem Kommentar der Variablen, die

Kommentare der einzelnen Ebenen werden jeweils durch einen Punkt getrennt. Falls auf einer Strukturebene für eine Variable kein Kommentar vorhanden ist, entfällt er in der Anzeige, was Sie an zwei aufeinander folgenden Punkten erkennen können.

Sie können die Variableninformationen entweder gesammelt unterhalb eines KOP-/FUP-Netzwerks anzeigen lassen oder direkt am jeweiligen Operanden. Wie beim Ein- und Ausblenden der Variableninformationen können Sie dies über die Einstellungen für alle Bausteine einstellen oder speziell für einen geöffneten Baustein ändern.

Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Variableninformationen" entweder die Option "Einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
4. Legen Sie über die Klappliste "Position der Variableninformation (KOP/FUP)" fest, ob die Variableninformationen unterhalb eines KOP-/FUP-Netzwerks oder am Operanden angezeigt werden soll.
   Die Variableninformationen werden für alle geöffneten Bausteine eingeblendet und an der gewählten Position angezeigt. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls eingeblendet.
5. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Variableninformationen" die Option "Ausblenden".
   Die Variableninformationen werden für alle geöffneten Bausteine ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls ausgeblendet.

Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." entweder die Option "Variableninformationen einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
2. Um die Position der Variableninformationen festzulegen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Position der Variableninformation (KOP/FUP)". Mit jedem Klick auf die Schaltfläche ändert sich die Position. Alternativ können Sie auf den kleinen Pfeil neben der Schaltfläche "Position der Variableninformation (KOP/FUP)" klicken, um eine Klappliste zu öffnen. Wählen Sie dann die Position aus der Klappliste aus.
   Die Variableninformationen werden eingeblendet und an der gewählten Position angezeigt.
3. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." die Option "Variableninformationen ausblenden". Die Variableninformationen werden ausgeblendet.

13.7 Zweige in KOP

13.7.1 Grundlagen zu Zweigen in KOP

Definition

Um mit der Programmiersprache Kontaktplan (KOP) Parallelschaltungen zu programmieren, verwenden Sie Zweige. Zweige werden in den Hauptstrompfad eingefügt. Sie können in den Zweig mehrere Kontakte einfügen und damit eine Parallelschaltung von Reihenschaltungen erreichen. Auf diese Weise können Sie komplexe Kontaktpläne programmieren.

Das folgende Bild zeigt ein Beispiel für die Verwendung von Zweigen:

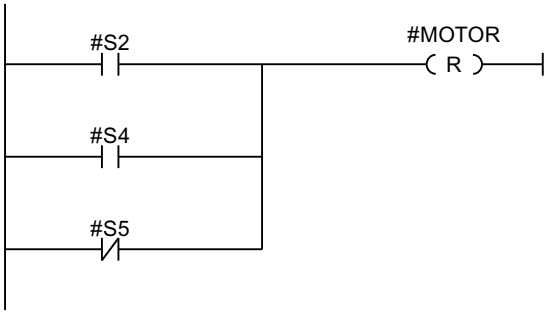

MOTOR führt Signal 1, wenn eine der folgenden Bedingungen erfüllt ist:

- An S2 oder S4 steht Signal 1 an
- An S5 steht Signal 0 an.

Siehe auch

Regeln für Zweige in KOP (Seite 7471)

Zweige in KOP-Netzwerke einfügen (Seite 7472)

Zweige in KOP-Netzwerke schließen (Seite 7473)

Zweige in KOP-Netzwerke löschen (Seite 7473)

13.7.2 Regeln für Zweige in KOP

Regeln

Für parallele Verzweigungen gelten folgende Regeln:

- Das Einfügen eines Parallelzweigs ist nur dann möglich, wenn im Hauptzweig bereits ein KOP-Element existiert.
- Parallelzweige werden nach unten geöffnet oder direkt an die Stromschiene angeschlossen und nach oben geschlossen.

- Parallelzweige werden hinter dem markierten KOP-Element geöffnet.
- Parallelzweige werden hinter dem markierten KOP-Element geschlossen.
- Um einen Parallelzweig zu löschen, müssen Sie alle KOP-Elemente dieses Zweiges löschen. Beim Entfernen des letzten KOP-Elements der Verzweigung wird der Rest der Verzweigung ebenfalls entfernt.
- Das Einfügen einer Spule in eine Parallelschaltung ist nur möglich, wenn die Parallelschaltung direkt an der Stromschiene beginnt.

Siehe auch

Grundlagen zu Zweigen in KOP (Seite 7471)

Zweige in KOP-Netzwerke einfügen (Seite 7472)

Zweige in KOP-Netzwerke löschen (Seite 7473)

Zweige in KOP-Netzwerke schließen (Seite 7473)

13.7.3 Zweige in KOP-Netzwerke einfügen

Sie können in einem Netzwerk mehrere Zweige erstellen.

Voraussetzung

- Ein Netzwerk ist vorhanden.
- Das Netzwerk enthält Elemente.

Vorgehen

Um einen neuen Zweig in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.
   Falls Sie den neuen Zweig direkt an die Stromschiene setzen möchten, ziehen Sie das Element auf die Stromschiene.

Siehe auch

Grundlagen zu Zweigen in KOP (Seite 7471)

Regeln für Zweige in KOP (Seite 7471)

Zweige in KOP-Netzwerke löschen (Seite 7473)

13.7.4 Zweige in KOP-Netzwerke schließen

Zweige müssen an geeigneten Stellen wieder geschlossen werden. Wenn erforderlich, werden die Zweige so angeordnet, dass Überkreuzungen von Zweigen vermieden werden.

Voraussetzung

Ein Zweig ist vorhanden.

Vorgehen

Um einen offenen Zweig zu schließen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den offenen Zweig.
2. Drücken und halten Sie die linke Maustaste.
   Eine gestrichelte Linie wird eingeblendet, sobald Sie den Mauszeiger bewegen.
3. Ziehen Sie die gestrichelte Linie auf eine geeignete Stelle im Netzwerk. Erlaubte Verbindungen werden durch grüne Linien angezeigt.
4. Lassen Sie die linke Maustaste los.

Siehe auch

Grundlagen zu Zweigen in KOP (Seite 7471)

Regeln für Zweige in KOP (Seite 7471)

13.7.5 Zweige in KOP-Netzwerke löschen

Voraussetzung

Ein Zweig ist vorhanden.

Vorgehen

Um einen Zweig zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die den Zweig mit dem Hauptzweig verbindet.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zu Zweigen in KOP (Seite 7471)

Regeln für Zweige in KOP (Seite 7471)

Zweige in KOP-Netzwerke einfügen (Seite 7472)

13.8 Kreuzungen in KOP

13.8.1 Grundlagen zu Kreuzungen in KOP

Definition

Eine Kreuzung ist eine Stelle in einem KOP-Netzwerk, an der ein Zweig geschlossen und gleichzeitig ein weiterer Zweig geöffnet wird.

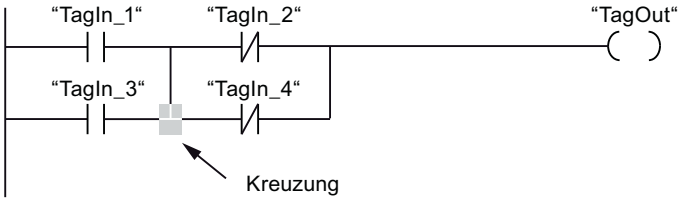

"TagOut" erhält Signal 1, wenn folgende beiden Bedingungen erfüllt sind:

- "TagIn_1" oder "TagIn_3" führt Signal 1
- "TagIn_2" oder "TagIn_4" führt Signal 0

13.8.2 Kreuzungen einfügen

Sie können Kreuzungen in ein KOP-Netzwerk einfügen, indem Sie Verbindungen zwischen dem Hauptzweig und einem zusätzlichen Zweig oder zwischen verschiedenen Zweigen erstellen.

Voraussetzung

Ein Zweig ist vorhanden.

Vorgehen

Um in ein KOP-Netzwerk eine Kreuzung einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".
3. Ziehen Sie das Element per Drag & Drop hinter den vorhandenen Zweig.
4. Fügen Sie ein beliebiges Element in den offenen Zweig ein.
5. Klicken Sie hinter dem eingefügten Element auf den Pfeil des geöffneten Zweigs.

6. Halten Sie die linke Maustaste gedrückt und ziehen Sie die gestrichelte Verbindungslinie auf den Hauptzweig.
7. Lassen Sie die linke Maustaste los.

Siehe auch

Kreuzungen umhängen (Seite 7475)

Kreuzungen löschen (Seite 7476)

Zweige in KOP-Netzwerke einfügen (Seite 7472)

13.8.3 Kreuzungen umhängen

Voraussetzung

Eine Kreuzung ist vorhanden.

Vorgehen

Um eine Kreuzung umzuhängen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die in den jeweiligen Zweigen die Kreuzungen definiert.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Öffnen Sie die Task Card "Anweisungen".
4. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".
5. Ziehen Sie das Element per Drag & Drop an die Stelle im Netzwerk, an der Sie die neue Kreuzung einfügen möchten.
6. Klicken Sie auf den Pfeil des geöffneten Zweigs.
7. Halten Sie die linke Maustaste gedrückt und ziehen Sie die gestrichelte Verbindungslinie auf den Nebenzweig, in den Sie die neue Kreuzung einfügen möchten.
8. Lassen Sie die linke Maustaste los.

Siehe auch

Kreuzungen einfügen (Seite 7474)

Kreuzungen löschen (Seite 7476)

13.8.4 Kreuzungen löschen

Voraussetzung

Eine Kreuzung ist vorhanden.

Vorgehen

Um eine Kreuzung zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die in den jeweiligen Zweigen die Kreuzungen definiert.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Kreuzungen einfügen (Seite 7474)
Kreuzungen umhängen (Seite 7475)

13.9 Strompfade in KOP

13.9.1 Grundlagen zu Strompfaden in KOP

Verwendung von Strompfaden

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält am linken Rand eine Stromschiene, von der ein oder mehrere Strompfade ausgehen können. Auf den Strompfaden werden die Abfragen der binären Signale in Form von Kontakten angeordnet. Durch serielle Anordnung der Elemente auf einem Strompfad entsteht eine Reihenschaltung, durch Anordnung auf parallelen Zweigen entsteht eine Parallelschaltung. Ein Strompfad wird von einer Spule oder einer Box abgeschlossen, in die das Verknüpfungsergebnis geschrieben wird.

Das folgende Bild zeigt ein Beispiel für die Verwendung von mehreren Strompfaden innerhalb eines Netzwerks:

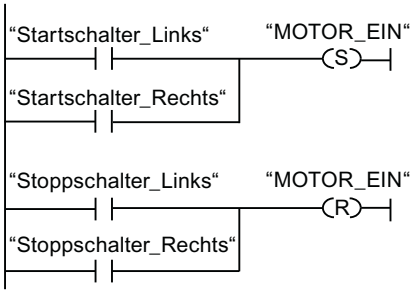

Regeln

Beachten Sie folgende Regeln bei der Verwendung von mehreren Strompfaden:

- Verbindungen zwischen Strompfaden sind nicht erlaubt.
- Es ist nur eine Sprunganweisung pro Netzwerk zulässig. Die Platzierungsregeln für Sprunganweisungen bleiben dabei erhalten.

Abarbeitung von Strompfaden

Die Abarbeitung von Strompfaden und Netzwerken erfolgt von oben nach unten und von links nach rechts. Das bedeutet, dass zuerst die erste Anweisung im ersten Strompfad des ersten Netzwerks bearbeitet wird. Anschließend werden alle Anweisungen dieses Strompfads bearbeitet. Danach kommen alle weiteren Strompfade des ersten Netzwerks an die Reihe. Erst wenn alle Strompfade abgearbeitet wurden, wird das nächste Netzwerk bearbeitet.

Unterschiede zwischen Zweigen und Strompfaden

Der Unterschied zwischen Zweigen und Strompfaden besteht darin, dass die Strompfade eigenständige Zweige sind, die auch in einem anderen Netzwerk stehen können. Zweige hingegen erlauben die Programmierung einer Parallelschaltung.

Siehe auch

Strompfad einfügen (Seite 7478)

Strompfad löschen (Seite 7478)

13.9.2 Strompfad einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Ein Netzwerk ist vorhanden.

Vorgehen

Um einen neuen Strompfad in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie eine beliebige Spule an der Stromschiene ein.
   Ein neuer Strompfad wird eingefügt und die Spule wird am Ende des Strompfads platziert.
2. Fügen Sie weitere Anweisungen in den neuen Strompfad ein.

Siehe auch

Grundlagen zu Strompfaden in KOP (Seite 7477)

Strompfad löschen (Seite 7478)

13.9.3 Strompfad löschen

Voraussetzung

Ein Strompfad ist vorhanden.

Vorgehen

Um einen Strompfad zu löschen, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen um den Strompfad. Achten Sie darauf, dass Sie dabei alle Anweisungen des Strompfads selektieren. Alternativ können Sie auch mit gedrückter <Shift>-Taste die erste und letzte Anweisung des Strompfads selektieren.
2. Klicken Sie mit der rechten Maustaste auf eine der Anweisungen im Strompfad.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zu Strompfaden in KOP (Seite 7477)
Strompfad einfügen (Seite 7478)

13.10 Programmierbeispiele KOP

13.10.1 Beispiel für das Steuern eines Förderbands

Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Drucktaster, S1 für START und S2 für STOP. Am Ende des Förderbands befinden sich ebenfalls zwei Drucktaster, S3 für START und S4 für STOP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.

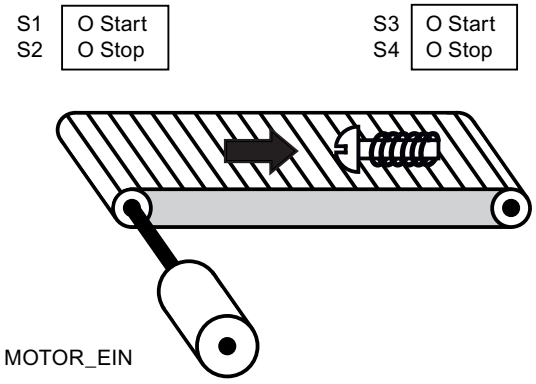

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
Starttaster_Links (S1)	Input	BOOL	Starttaster am linken Rand des Förderbands
Stoptaster_Links (S2)	Input	BOOL	Stoptaster am linken Rand des Förderbands
Starttaster_Rechts (S3)	Input	BOOL	Starttaster am rechten Rand des Förderbands
Stoptaster_Rechts (S4)	Input	BOOL	Stoptaster am rechten Rand des Förderbands
MOTOR_EIN	Output	BOOL	Motor des Förderbands einschalten

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Lösung der Aufgabe:

Netzwerk 1:

Der Motor des Förderbands wird eingeschaltet, wenn der Starttaster "S1" oder "S3" betätigt wird.

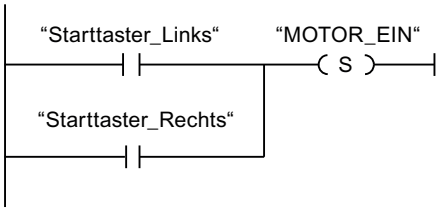

Netzwerk 2:

Der Motor des Förderbands wird ausgeschaltet, wenn der Stoptaster "S2" oder "S4" betätigt wird.

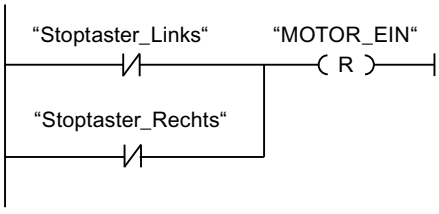

Siehe auch

Grundlagen zu KOP (Seite 7425)

Einstellungen zu KOP (Seite 7428)

Arbeiten mit Netzwerken (Seite 7430)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Zweige in KOP (Seite 7471)

Kreuzungen in KOP (Seite 7474)

Strompfade in KOP (Seite 7477)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7482)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Beispiel für das Berechnen einer Gleichung (Seite 7487)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7490)

13.10.2 Beispiel für das Erfassen der Richtung eines Förderbands

Erfassen der Richtung eines Förderbandes

Die erkannte Laufrichtung des Bandes wird mit einem Pfeil RECHTS oder einem Pfeil LINKS angezeigt. Läuft weiteres Fördergut von rechts an LS1 oder von links an LS2 heran, wird zunächst der angezeigte Pfeil ausgeschaltet, bis nach dem Durchlaufen beider Lichtschranken die Laufrichtung wieder neu festgestellt wurde und der entsprechende Pfeil angezeigt werden kann. Für die Lösung der Aufgabe sind noch 2 Flankenmerker notwendig, die den Signalwechsel an den zwei Lichtschranken von "0" auf "1" erkennen.

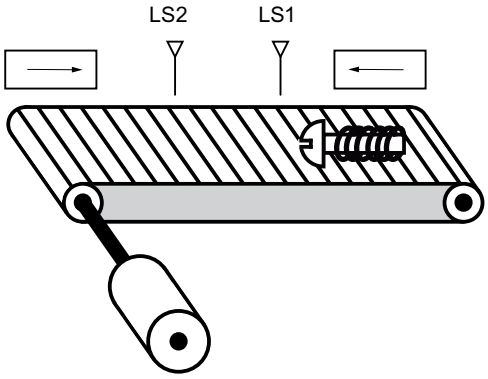

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
RECHTS	Output	BOOL	Anzeige bei einer Bewegung nach rechts
LINKS	Output	BOOL	Anzeige bei einer Bewegung nach links
TM1	Input	BOOL	Flankenmerker 1
TM2	Input	BOOL	Flankenmerker 2

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Lösung der Aufgabe:

Netzwerk 1:

Wenn an der Lichtschranke "LS1" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalflanke) und gleichzeitig der Signalzustand an "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links.

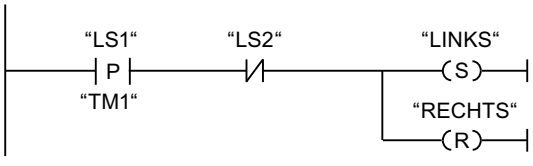

Netzwerk 2:

Wenn an der Lichtschranke "LS2" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalflanke) und gleichzeitig der Signalzustand an "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts.

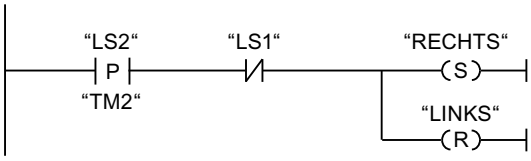

Siehe auch

Grundlagen zu KOP (Seite 7425)

Einstellungen zu KOP (Seite 7428)

Arbeiten mit Netzwerken (Seite 7430)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Zweige in KOP (Seite 7471)

Kreuzungen in KOP (Seite 7474)

Strompfade in KOP (Seite 7477)

Beispiel für das Steuern eines Förderbands (Seite 7480)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Beispiel für das Berechnen einer Gleichung (Seite 7487)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7490)

13.10.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbandes 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, von der sie auf einen LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

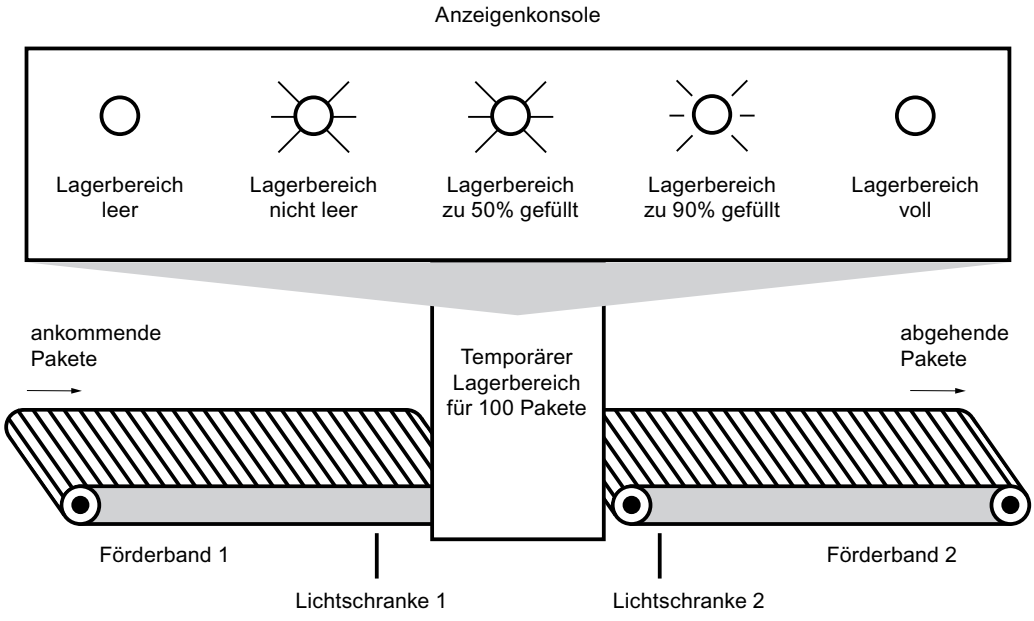

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
RESET	Input	BOOL	Zähler rücksetzen
LOAD	Input	BOOL	Den aktuellen Zählerwert auf den Wert des Parameters PV anpassen.
MAX LAGERFUELL-MENGE	Input	INT	Maximal mögliche Anzahl der Pakete im Lager

Name	Deklaration	Datentyp	Beschreibung
PAKETANZAHL	Output	INT	Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)
LB_PAKETE	Output	BOOL	Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELLMENGE" ist.
LAG_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich leer
LAG_NICHT_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich nicht leer
LAG_50%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 50 % voll
LAG_90%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 90 % voll
LAG_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich voll
VOLUME_50	Input	INT	Vergleichswert: 50 Pakete
VOLUME_90	Input	INT	Vergleichswert: 90 Pakete
VOLUME_100	Input	INT	Vergleichswert: 100 Pakete

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Aktivierung der Anzeigeleuchten:

Netzwerk 1:

Wenn ein Paket zum Lagebereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "MAX LAGERFUELLMENGE" gesetzt. Solange der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELLMENGE" ist, liefert die Variable "LB_PAKETE" den Signalzustand "1".

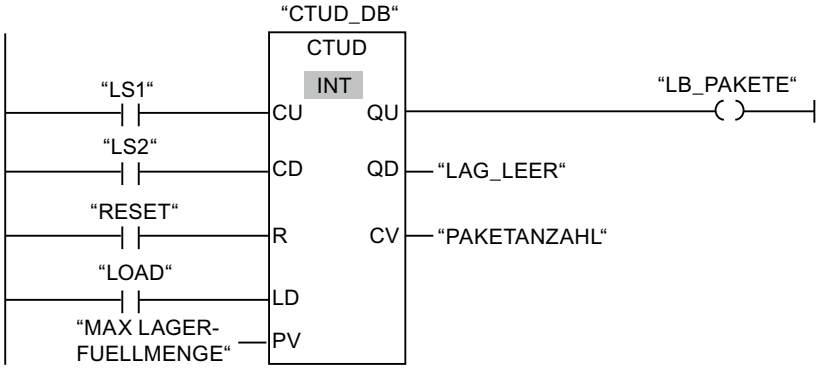

Netzwerk 2:

Solange sich Pakete im Lagerbereich befinden, wird die Variable "LAG_NICHT_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.

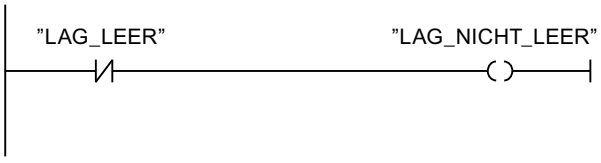

Netzwerk 3:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50 % gefüllt" ein.

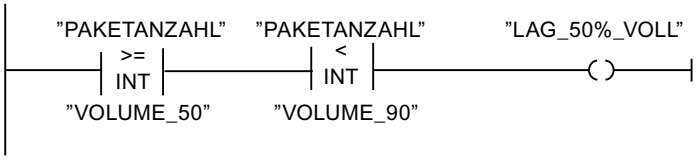

Netzwerk 4:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90 % gefüllt" ein.

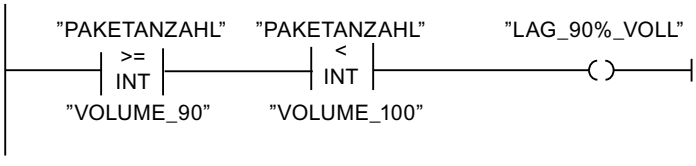

Netzwerk 5:

Wenn die Anzahl der Pakete im Lagerbereich 100 erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein.

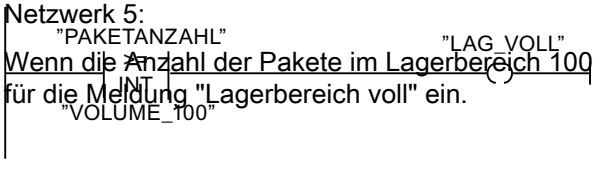

Siehe auch

Grundlagen zu KOP (Seite 7425)
Einstellungen zu KOP (Seite 7428)
Arbeiten mit Netzwerken (Seite 7430)
KOP-Elemente einfügen (Seite 7439)
KOP-Elemente bearbeiten (Seite 7459)
Operanden in KOP-Anweisungen beschalten (Seite 7466)
Zweige in KOP (Seite 7471)
Kreuzungen in KOP (Seite 7474)
Strompfade in KOP (Seite 7477)
Beispiel für das Steuern eines Förderbands (Seite 7480)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7482)
Beispiel für das Berechnen einer Gleichung (Seite 7487)
Beispiel für das Kontrollieren der Raumtemperatur (Seite 7490)

13.10.4 Beispiel für das Berechnen einer Gleichung

Berechnen einer Gleichung

Das folgende Programmbeispiel zeigt Ihnen, wie Sie mit der Programmiersprache KOP Gleichungen des folgenden Typs lösen können:

X= ((A + B) x C) / D

Grundsätzlich können Sie dies mithilfe von mathematischen Operationen umsetzen. Dafür stehen Ihnen die Anweisungen "ADD", "MUL" und "DIV" zur Verfügung. Setzen Sie eine CPU der Baureihe S7-1200 oder S7-1500 ein, können Sie alternativ die Anweisung "CALCULATE" verwenden. Für diese Anweisung stehen Ihnen abhängig vom gewählten Datentyp unterschiedliche mathematische Operationen zur Verfügung, die Sie miteinander kombinieren können.

Umsetzung mithilfe der Anweisungen "ADD", "MUL" und "DIV"

Es soll die folgende Gleichung berechnet werden:

RESULT = ((A + B) x 15) / E

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Datentyp	Kommentar
A	INT	Erster Wert der Addition
B	INT	Zweiter Wert der Addition
C	INT	Erstes Zwischenergebnis
15	INT	Multiplikator

Name	Datentyp	Kommentar
D	INT	Zweites Zwischenergebnis
E	INT	Divisor
RESULT	INT	Endergebnis

Das folgende Netzwerk zeigt die KOP-Programmierung zur Berechnung der Gleichung:

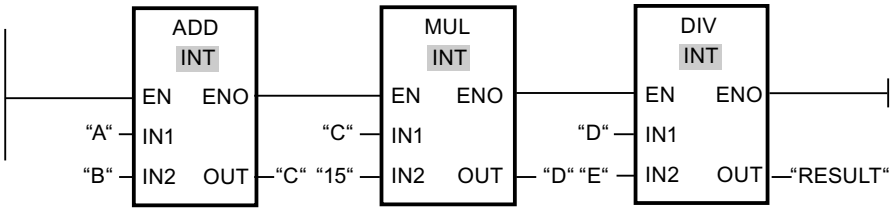

Der Wert des Operanden "A" wird mit dem Wert des Operanden "B" addiert. Die Summe wird im Operanden "C" gespeichert. Der Wert des Operanden "C" wird mit "15" multipliziert. Das Ergebnis der Multiplikation wird im Operanden "D" gespeichert. Anschließend wird der im Operanden "D" gespeicherte Wert durch den Wert des Operanden "E" dividiert. Das Endergebnis wird im Operanden "RESULT" gespeichert.

Umsetzung mithilfe der Anweisung "CALCULATE" (nur S7-1200/1500)

Es soll die folgende Gleichung berechnet werden:

RESULT = ((5 + 10) x 4) / 6

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Kommentar	Wert
A	Input	INT	Erster Wert der Addition	5
B	Input	INT	Zweiter Wert der Addition	10
C	Input	INT	Multiplikator	4
D	Input	INT	Divisor	6
RESULT	Output	INT	Endergebnis	10

Zur Programmierung der Gleichung mithilfe der Anweisung "CALCULATE", gehen Sie folgendermaßen vor:

1. Ziehen Sie die Anweisung "CALCULATE" aus der Task Card "Anweisungen" in ein KOP-Netzwerk.
2. Wählen Sie aus der Klappliste "<???>" den Datentyp INT für die Anweisung aus.
3. Verschalten Sie die in der Bausteinschnittstelle deklarierten Variablen mit den Ein- bzw. Ausgängen der Anweisungsbox.

4. Klicken Sie das "Taschenrechner"-Symbol am rechten oberen Rand der Anweisungsbox an, um die zu berechnende Gleichung einzutragen.
   Der Dialog "Anweisung Berechnen bearbeiten" wird geöffnet.
5. Geben Sie in das Feld "OUT:=" folgenden Ausdruck ein:
   ((IN1 + IN2) * IN4) / IN3
   Die Gleichung wird in der Anweisungsbox dargestellt.

Das folgende Netzwerk zeigt das Ergebnis in der Programmiersprache KOP:

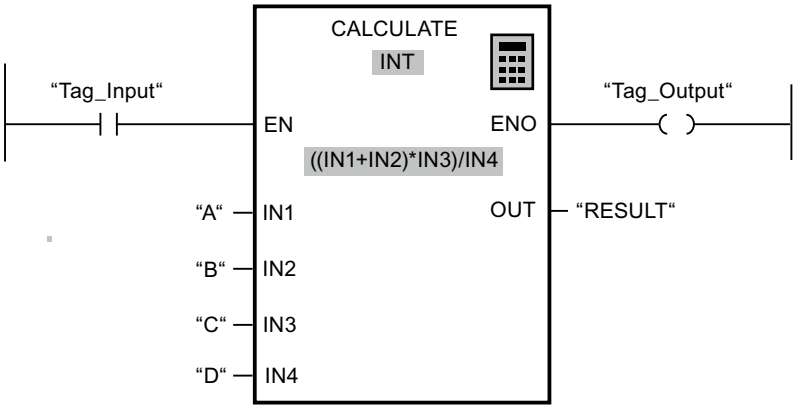

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "A" wird mit dem Wert des Operanden "B" addiert. Die Zwischensumme wird mit "C" multipliziert und anschließend durch den Wert des Operanden "D" dividiert. Das Endergebnis wird im Operanden "RESULT" gespeichert.

Siehe auch

Grundlagen zu KOP (Seite 7425)

Einstellungen zu KOP (Seite 7428)

Arbeiten mit Netzwerken (Seite 7430)

KOP-Elemente einfügen (Seite 7439)

KOP-Elemente bearbeiten (Seite 7459)

Operanden in KOP-Anweisungen beschalten (Seite 7466)

Zweige in KOP (Seite 7471)

Kreuzungen in KOP (Seite 7474)

Strompfade in KOP (Seite 7477)

Beispiel für das Steuern eines Förderbands (Seite 7480)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7482)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7490)

13.10.5 Beispiel für das Kontrollieren der Raumtemperatur

Kontrollieren der Raumtemperatur

In einem Kühlraum muss die Temperatur unter null Grad Celsius gehalten werden. Eventuelle Temperaturschwankungen werden mithilfe eines Sensors kontrolliert. Wenn die Temperatur über null Grad Celsius steigt, wird das Kühlsystem für eine voreingestellte Zeit eingeschaltet. Während des Kühlens leuchtet eine Anzeige "Kühlsystem gestartet".

Das Kühlsystem und die Anzeigeleuchte werden ausgeschaltet, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Sensor meldet eine Senkung der Temperatur unter null Grad Celsius.
- Die voreingestellte Kühlzeit ist abgelaufen.
- Der Drucktaster "STOP" wird betätigt.

Wenn die voreingestellte Kühlzeit abgelaufen ist und die Temperatur im Kühlraum immer noch zu hoch ist, kann das Kühlsystem mithilfe des Drucktasters "RESET" erneut gestartet werden.

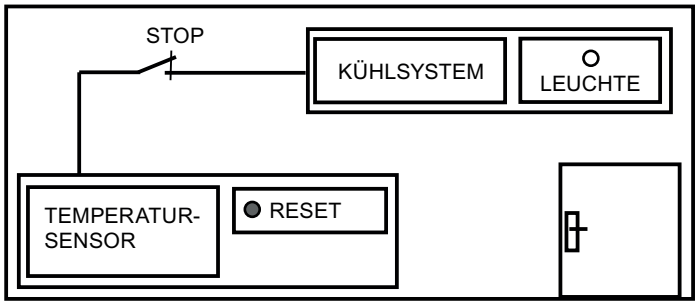

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Kommentar
Sensor	Input	BOOL	Signal des Temperatur-sensors
RESET	Input	BOOL	Neustart
STOP	Input	BOOL	Das Kühlsystem wird ausgeschaltet.
MaxKuehlzeit	-	TIME	Voreingestellte Kühlzeit Diese Variable ist im Da-tenbaustein "DB_Kueh-len" definiert.
AktKuehlzeit	-	TIME	Aktuell abgelaufene Kühlzeit Diese Variable ist im Da-tenbaustein "DB_Kueh-len" definiert.
Kuehlsystem	Output	BOOL	Das Kühlsystem wird eingeschaltet.

Name	Deklaration	Datentyp	Kommentar
Leuchte	Output	BOOL	Die Anzeigeleuchte für die Meldung "Kühlsystem gestartet" wird eingeschaltet.
TempVariable	Temp	BOOL	Temporäre Variable Diese Variable speichert den Signalzustand der IEC-Zeit TP.

Das folgende Netzwerk zeigt die KOP-Programmierung zum Kontrollieren der Raumtemperatur:

Netzwerk 1:

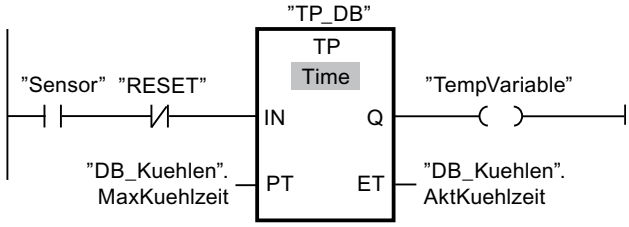

Netzwerk 2:

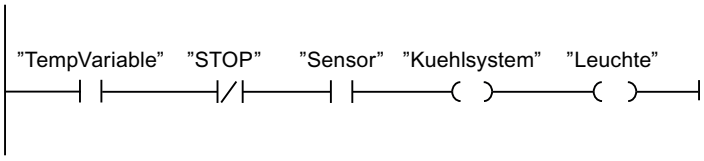

Wenn die Temperatur im Kühlraum höher als null Grad Celsius wird, wechselt der Signalzustand am Operanden "Sensor" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke am Eingang IN der Zeitfunktion wird die voreingestellte Kühlzeit gestartet und die "TempVariable" erhält den Signalzustand "1". Der Signalzustand "1" der "TempVariable" bewirkt im Netzwerk 2, dass das Kühlsystem sowie die Anzeigeleuchte eingeschaltet werden. Die Ausgänge "Sensor", "Kuehlsystem" und "Leuchte" müssen im Netzwerk 2 programmiert werden, da am Ausgang Q der Zeitfunktion nur eine Spule programmiert werden kann.

Sinkt die Temperatur im Kühlraum unter null Grad Celsius, wechselt der Signalzustand des Sensors wieder auf "0". Dadurch werden das Kühlsystem und die Anzeigeleuchte ausgeschaltet.

Wenn der Sensor keine Senkung der Temperatur meldet, werden das Kühlsystem und die Anzeigeleuchte spätestens nach dem Ablauf der voreingestellten Kühlzeit ausgeschaltet. Der Kühlprozess kann in diesem Fall mithilfe des Drucktasters "RESET" neu gestartet werden. Durch das Drücken und Loslassen des Drucktaster wird eine neue positive Signalflanke am Eingang IN erzeugt und dadurch das Kühlsystem neu gestartet.

Das Kühlsystem und die Anzeigeleuchte können jederzeit mithilfe des Drucktasters "STOP" ausgeschaltet werden.

Siehe auch

Grundlagen zu KOP (Seite 7425)
Einstellungen zu KOP (Seite 7428)
Arbeiten mit Netzwerken (Seite 7430)
KOP-Elemente einfügen (Seite 7439)
KOP-Elemente bearbeiten (Seite 7459)
Operanden in KOP-Anweisungen beschalten (Seite 7466)
Zweige in KOP (Seite 7471)
Kreuzungen in KOP (Seite 7474)
Strompfade in KOP (Seite 7477)
Beispiel für das Steuern eines Förderbands (Seite 7480)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7482)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7484)
Beispiel für das Berechnen einer Gleichung (Seite 7487)

FUP-Programme erstellen 14

14.1 Grundlagen zu FUP

14.1.1 Programmiersprache FUP

Übersicht über die Programmiersprache FUP (Funktionsplan)

FUP ist eine grafische Programmiersprache. Die Darstellung ist elektronischen Schaltkreissystemen nachempfunden.

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält ein oder mehrere Verknüpfungspfade. Die Abfragen der binären Signale werden durch Boxen miteinander verknüpft. Zur Darstellung der Logik werden die von der booleschen Algebra bekannten grafischen Logiksymbole verwendet.

Beispiel für Netzwerke in FUP

Das folgende Bild zeigt ein FUP-Netzwerk mit UND- und ODER-Boxen und einer Zuweisung:

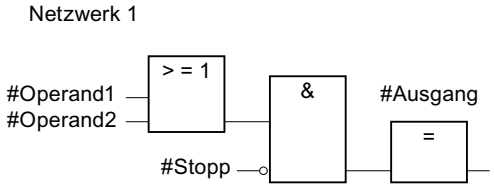

Siehe auch

Arbeiten mit Netzwerken (Seite 7498)

14.1.2 Übersicht über die FUP-Elemente

FUP-Elemente

Ein FUP-Programm besteht aus einzelnen Elementen, die über den binären Signalfluss miteinander verbunden sind. Die meisten Programmelemente müssen mit Variablen versorgt werden.

Das Programmieren in einem FUP-Netzwerk verläuft von links nach rechts.

Das folgende Bild zeigt beispielsweise Elemente eines FUP-Netzwerks:

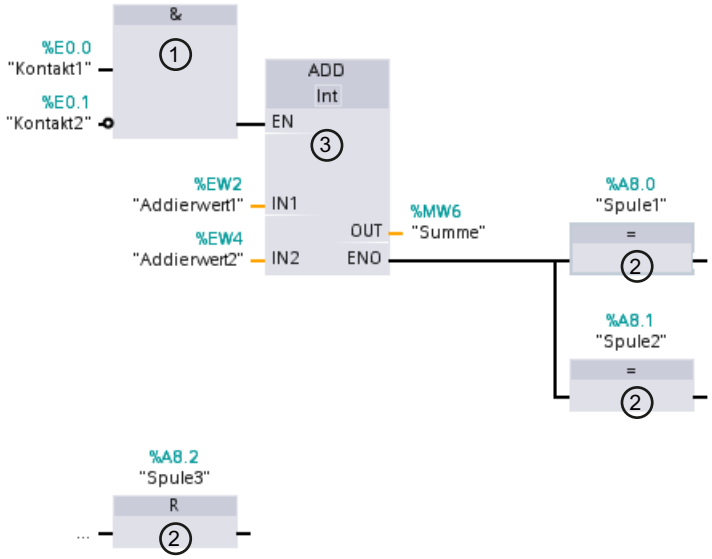

1) Binäre Funktion

2) Einfache Box

3) Komplexe Box

Binäre Funktionen

Mit binären Funktionen können Sie Binäroperanden abfragen und deren Signalzustände verknüpfen. Beispiele für binäre Funktionen sind die Anweisungen "UND-Verknüpfung", "ODER-Verknüpfung" und "EXKLUSIV ODER-Verknüpfung".

Einfache Boxen

Mit einfachen Boxen können Sie Binäroperanden steuern, Flanken auswerten oder Sprungfunktionen im Programm ausführen. Einfache Boxen haben im Allgemeinen nur einen einzigen Eingang.

Komplexe Boxen

Komplexe Boxen stellen Programmelemente mit komplexen Funktionen dar. Eine Ausnahme dabei ist die Leerbox. Die Leerbox dient als Platzhalter, in dem Sie die gewünschte Anweisung auswählen können.

Die folgenden Arten von Boxen stehen Ihnen in einem FUP-Programm zur Verfügung:

- Komplexe Boxen ohne EN-/ENO-Mechanismus:
  Eine Box wird unabhängig vom Signalzustand an den Box-Eingängen ausgeführt. Der Fehlerstatus der Bearbeitung kann nicht abgefragt werden.
- Komplexe Boxen mit EN-/ENO-Mechanismus:
  Eine Box wird nur dann ausgeführt, wenn der Freigabeeingang "EN" den Signalzustand "1" führt. Bei ordnungsgemäßer Bearbeitung der Box führt der Freigabeausgang "ENO" den Signalzustand "1". Wenn während der Bearbeitung ein Fehler auftritt, wird der Freigabeausgang "ENO" zurückgesetzt.
  Wenn der Freigabeeingang EN nicht verschaltet ist, wird die Box immer ausgeführt.

Aufrufe von Codebausteinen werden im Netzwerk auch als komplexe Boxen mit EN-/ENO-Mechanismus dargestellt.

Siehe auch

EN-/ENO-Mechanismus (Seite 150)

14.2 Einstellungen zu FUP

14.2.1 Übersicht über die Einstellungen zu FUP

Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie vornehmen können:

Gruppe	Einstellung	Beschreibung
Schriftart	Schriftgröße	Schriftgröße im Programmiereditor
Ansicht	Layout	Kompakt oder weit Ändert den vertikalen Abstand zwischen Operanden und anderen Objekten (z. B. Operanden und Kontakt). Die Änderung wird erst nach dem erneuten Öffnen eines Bausteins sichtbar.
Operandenfeld	Maximale Breite	Maximale Anzahl der Zeichen, die horizontal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.
	Maximale Höhe	Maximale Anzahl der Zeichen, die vertikal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.

Siehe auch

Einstellungen ändern (Seite 7496)

14.2.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die Einstellungen zu FUP (Seite 7496)

14.3 Arbeiten mit Netzwerken

14.3.1 Netzwerke verwenden

Funktion

Das Anwenderprogramm wird im Baustein innerhalb von Netzwerken erstellt. Damit ein Codebaustein programmiert werden kann, muss er mindestens ein Netzwerk enthalten. Um eine bessere Übersichtlichkeit des Anwenderprogramms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

Sie haben die Möglichkeit, in KOP- und FUP-Bausteinen auch Netzwerke für die textuellen Programmiersprachen SCL und AWL einzufügen und dann Anweisungen in diesen Programmiersprachen zu verwenden. Abhängig von der eingesetzten CPU können Sie Netzwerke für die folgenden Programmiersprachen einfügen:

- S7-300/400: AWL-Netzwerke
- S7-1200: SCL-Netzwerke
- S7-1500: AWL- und SCL-Netzwerke

Der Programmiereditor passt sich immer an das Netzwerk an, das aktuell den Fokus besitzt. Das bedeutet, dass Ihnen z. B. bei der Programmierung eines SCL-Netzwerks die SCL-Anweisungen, aber auch die SCL-Funktionen zur Verfügung stehen.

Hinweis

In SCL-Netzwerken innerhalb von KOP- oder FUP-Bausteinen können Sie die Anweisung "Springen (GOTO)" nicht verwenden.

EN-/ENO-Mechanismus in Bausteinen mit unterschiedlichen Netzwerksprachen

Sie können auch in Bausteinen mit unterschiedlichen Netzwerksprachen den EN-/ENO-Mechanismus einsetzen. Jede Programmiersprache bildet den Fehlerstatus unterschiedlich ab:

- SCL besitzt eine Variable "ENO", die den Fehlerstatus speichert und die Sie abfragen können. Der direkte Zugriff auf diese Variable ist nur mit SCL möglich.
- KOP/FUP/AWL haben keine spezielle Variable für "ENO". Sie können den Fehlerstatus aber für AWL aus dem BIE-Bit auslesen und für KOP/FUP über die RET-Spule abfragen.

Für das Auslesen des Fehlerstatus für den gesamten Baustein gelten die folgenden Regeln:

- Das letzte Netzwerk im Baustein ist ein KOP-/FUP-Netzwerk:
  Falls Sie keine RET-Spule verwenden, ist der Fehlerstatus standardmäßig "TRUE".
- Das letzte Netzwerk im Baustein ist ein AWL-Netzwerk:
  Das BIE-Bit bestimmt den Fehlerstatus. Das BIE-Bit kann in AWL-Netzwerken durch das BIE-Register bearbeitet werden.
- Das letzte Netzwerk im Baustein ist ein SCL-Netzwerk:
  Die Variable "ENO" bestimmt den Fehlerstatus des Bausteins.

Siehe auch

Netzwerktitel einfügen (Seite 7502)

Netzwerkkommentar eingeben (Seite 7504)

In Netzwerken navigieren (Seite 7505)

14.3.2 Netzwerke einfügen

In einen KOP- oder FUP-Baustein können Sie entweder ein Netzwerk der gleichen Programmiersprache oder ein Netzwerk für eine textuelle Programmiersprache einfügen. Beachten Sie jedoch, dass nicht jeder CPU-Typ alle textuellen Sprachen unterstützt.

Voraussetzung

Ein KOP- oder FUP-Baustein ist geöffnet.

KOP- oder FUP-Netzwerk einfügen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

AWL- oder SCL-Netzwerk einfügen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü entweder den Befehl "AWL-Netzwerk einfügen" oder "SCL-Netzwerk einfügen".

Ergebnis

Es wird ein neues, leeres Netzwerk in der entsprechenden Programmiersprache in den Baustein eingefügt.

Siehe auch

Netzwerktitel einfügen (Seite 7502)

Netzwerkkommentar eingeben (Seite 7504)

In Netzwerken navigieren (Seite 7505)

14.3.3 Netzwerke selektieren

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.
   Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

Siehe auch

Netzwerke verwenden (Seite 7498)

Netzwerke einfügen (Seite 7499)

Netzwerktitel einfügen (Seite 7502)

Netzwerkkommentar eingeben (Seite 7504)

In Netzwerken navigieren (Seite 7505)

14.3.4 Netzwerke kopieren und einfügen

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

Netzwerke verwenden (Seite 7498)
Netzwerke einfügen (Seite 7499)
Netzwerke selektieren (Seite 7500)
Netzwerktitel einfügen (Seite 7502)
Netzwerkkommentar eingeben (Seite 7504)
In Netzwerken navigieren (Seite 7505)

14.3.5 Netzwerke löschen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Netzwerke verwenden (Seite 7498)
Netzwerke einfügen (Seite 7499)
Netzwerke selektieren (Seite 7500)
Netzwerke kopieren und einfügen (Seite 7500)
Netzwerktitel einfügen (Seite 7502)
Netzwerkkommentar eingeben (Seite 7504)
In Netzwerken navigieren (Seite 7505)

14.3.6 Netzwerke reduzieren und erweitern

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

Alle Netzwerke auf- und zuklappen

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

Siehe auch

Netzwerke verwenden (Seite 7498)

Netzwerke einfügen (Seite 7499)

Netzwerke selektieren (Seite 7500)

Netzwerke kopieren und einfügen (Seite 7500)

Netzwerke löschen (Seite 7501)

Netzwerktitel einfügen (Seite 7502)

Netzwerkkommentar eingeben (Seite 7504)

In Netzwerken navigieren (Seite 7505)

14.3.7 Netzwerktitel einfügen

Der Netzwerktitel ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitels ist auf eine Zeile beschränkt. Sie können den Titel entweder manuell eingeben oder automatisch setzen. Beim automatischen Setzen können Sie dies für einzelne Netzwerke vornehmen oder über die Einstellungen festlegen, dass der Netzwerktitel grundsätzlich automatisch gesetzt werden soll.

Für das automatische Einfügen des Netzwerktitels wird der Kommentar des Operanden einer der folgenden Anweisungen im Netzwerk ausgewertet:

- Zuweisung
- Ausgang setzen
- Ausgang rücksetzen

Dabei wird die Anweisung verwendet, die im Netzwerk zuerst steht.

Der Netzwerktitel wird nur dann automatisch eingefügt, wenn die folgenden Bedingungen erfüllt sind:

- Das Netzwerk besitzt noch keinen Titel.
- Der Operand der Anweisung, die für den Kommentar verwendet wird, besitzt einen Kommentar.

Hinweis

Beachten Sie die folgenden Einschränkungen beim automatischen Einfügen des Netzwerktitels:

- Wenn Sie den Kommentar des Operanden nachträglich ändern, wird der Netzwerktitel nicht angepasst.
- Wenn Sie den Operanden der Anweisung wechseln, wird der Netzwerktitel nicht angepasst.
- Der Netzwerktitel wird nur durch die oben genannten schreibenden Anweisungen gesetzt.
- Ist der Operand vom Datentyp Array, wird der Kommentar des Arrays verwendet und nicht die Kommentare der Array-Elemente.
- Kommentare ungültiger Operanden werden nicht berücksichtigt.

Netzwerktitel manuell eingeben

Um einen Netzwerktitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelzeile des Netzwerks.
2. Geben Sie den Netzwerktitel ein.

Netzwerktitel automatisch setzen

Um festzulegen, dass die Netzwerktitel grundsätzlich automatisch gesetzt werden sollen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Aktivieren Sie in der Gruppe "Weitere Einstellungen" das Optionskästchen "Netzwerktitel automatisch setzen".
   Die Titel der Netzwerke werden ab diesem Zeitpunkt automatisch gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Um einen einzelnen Netzwerktitel automatisch zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste eines Netzwerks mit der rechten Maustaste auf "Netzwerk <Nummer des Netzwerks>".
2. Wählen Sie im Kontextmenü den Befehl "Netzwerktitel automatisch setzen".
   Der Titel des selektierten Netzwerks wird anhand des Kommentars des Operanden gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Siehe auch

Netzwerke verwenden (Seite 7498)

Netzwerke einfügen (Seite 7499)

Netzwerke selektieren (Seite 7500)

Netzwerke kopieren und einfügen (Seite 7500)

Netzwerke löschen (Seite 7501)

Netzwerke reduzieren und erweitern (Seite 7502)

Netzwerkkommentar eingeben (Seite 7504)

In Netzwerken navigieren (Seite 7505)

14.3.8 Netzwerkkommentar eingeben

Mithilfe von Netzwerkkommentaren können Sie die Programminhalte der einzelnen Netzwerke erläutern. Sie können z. B. die Funktion des Netzwerks beschreiben oder auf Besonderheiten hinweisen.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um einen Netzwerkkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel.
2. Wenn der Kommentarbereich nicht sichtbar ist, klicken Sie in der Funktionsleiste auf die Schaltfläche "Netzwerkkommentare ein/aus".
   Der Kommentarbereich wird angezeigt.
3. Klicken Sie im Kommentarbereich auf "Kommentar".
   Die Textstelle "Kommentar" wird markiert.
4. Geben Sie den Netzwerkkommentar ein.

Siehe auch

Netzwerke verwenden (Seite 7498)

Netzwerke einfügen (Seite 7499)

Netzwerke selektieren (Seite 7500)

Netzwerke kopieren und einfügen (Seite 7500)

Netzwerke löschen (Seite 7501)

Netzwerke reduzieren und erweitern (Seite 7502)

Netzwerktitel einfügen (Seite 7502)

In Netzwerken navigieren (Seite 7505)

14.3.9 In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".
   Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

Siehe auch

Netzwerke verwenden (Seite 7498)

Netzwerke einfügen (Seite 7499)

Netzwerke selektieren (Seite 7500)

Netzwerke kopieren und einfügen (Seite 7500)

Netzwerke löschen (Seite 7501)

Netzwerke reduzieren und erweitern (Seite 7502)

Netzwerktitel einfügen (Seite 7502)
Netzwerkkommentar eingeben (Seite 7504)

14.4 FUP-Elemente einfügen

14.4.1 Regeln für die Verwendung von FUP-Elementen

Regeln

Beachten Sie beim Einfügen von FUP-Elementen folgende Regeln:

- Ein FUP-Netzwerk kann aus mehreren Elementen bestehen. Alle Elemente eines Verknüpfungspfads müssen nach IEC 61131-3 untereinander verbunden sein.
- An Boxen mit binären Verknüpfungen (z. B. UND, ODER) können Standard-Boxen (Flipflops, Zähler, Zeiten, Rechenoperationen usw.) als Ausgang angefügt werden. Ausgenommen von dieser Regelung sind Vergleicherboxen.
- Nur boolesche Eingänge einer Anweisung sind vorverknüpfbar.
- Nur der unterste boolesche Ausgang einer Anweisung ist weiter verknüpfbar.
- Die Beschaltung des Freigabeeingangs EN bzw. des Freigabeausgangs ENO von Boxen kann erfolgen, ist aber nicht zwingend erforderlich.
- Binäre Verknüpfungen können nicht mit Konstanten belegt werden, z. B. mit TRUE oder FALSE. Verwenden Sie stattdessen Variablen mit dem Datentyp BOOL.
- Pro Netzwerk kann nur eine Sprunganweisung eingefügt werden.
- Pro Netzwerk kann nur eine Sprungmarke eingefügt werden.
- Anweisungen mit positiver oder negativer Flankenauswertung dürfen nicht direkt am linken Rand des Netzwerks angeordnet werden, da diese eine Vorverknüpfung voraussetzen.

Platzierungsregeln für S7-1200/1500 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

Anweisung		Vorverknüpfung notwendig
Mnemonik	Name	
SET_BF	Bitfeld setzen	Nein
RESET_BF	Bitfeld rücksetzen	Nein
JMP	Springen bei VKE = 1	Nein
JMPN	Springen bei VKE = 0	Ja
JMP_LIST	Sprungliste definieren	Nein
SWITCH	Sprungverteiler	Nein
RET	Zurück springen	Nein

Platzierungsregeln für S7-300/400 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

Anweisung		Vorverknüpfung notwendig
Mnemonik	Name	
S	Ausgang setzen	Ja
R	Ausgang rücksetzen	Ja
SI	Zeit als Impuls starten	Ja
SV	Zeit als verlängerten Impuls starten	Ja
SE	Zeit als Einschaltverzögerung starten	Ja
SS	Zeit als speichernde Einschaltverzögerung starten	Ja
SA	Zeit als Ausschaltverzögerung starten	Ja
SZ	Zähleranfangswert setzen	Ja
ZV	Vorwärts zählen	Ja
ZR	Rückwärts zählen	Ja
JMP	Springen bei VKE = 1	Nein
JMPN	Springen bei VKE = 0	Ja
RET	Zurück springen	Nein
OPN	Globalen Datenbaustein öffnen	Nein
OPNI	Instanz-Datenbaustein öffnen	Nein
CALL	Baustein aufrufen	Nein
SAVE	VKE im BIE-Bit speichern	Nein
MCRA	MCR-Bereich aktivieren	Nein
MCRD	MCR-Bereich deaktivieren	Nein
MCR<	MCR-Bereiche öffnen	Nein
MCR>	MCR-Bereiche schließen	Nein

Hinweis

Aufruf von Bausteinen mit binären Eingängen in S7-300/400 CPUs

Der folgende FAQ beschreibt die Besonderheiten beim Aufruf von Bausteinen mit binären Eingängen:

FAQ 13988019: Was müssen Sie bei der Programmierung von verschachtelten FB/FC-Aufrufen mit binären Eingängen beachten? (https://support.industry.siemens.com/cs/de/de/view/13988019)

14.4.2 FUP-Elemente über Task Card "Anweisungen" einfügen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein FUP-Element mithilfe der Task Card "Anweisungen" in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie zu dem FUP-Element, das Sie einfügen möchten.
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Oder:

1. Markieren Sie die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
2. Öffnen Sie die Task Card "Anweisungen".
3. Doppelklicken Sie auf das einzufügende Element.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Ergebnis

Das ausgewählte FUP-Element wird mit Platzhaltern für die Parameter eingefügt.

Siehe auch

Regeln für die Verwendung von FUP-Elementen (Seite 7507)

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

Parameterinstanzen (Seite 70)

14.4.3 FUP-Elemente mithilfe einer Leerbox einfügen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein FUP-Element mithilfe einer Leerbox in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke der Leerbox. Eine Klappliste wird geöffnet.
5. Wählen Sie das gewünschte FUP-Element aus der Klappliste.
   Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie festlegen, ob die Daten des eingefügten Elements in einer Einzel-, Multi- oder Parameterinstanz gespeichert werden sollen. Wenn Sie "Einzelinstanz" auswählen, finden Sie den neuen Instanz-Datenbaustein nach dem Erstellen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Multiinstanzen finden Sie in der Bausteinschnittstelle im Abschnitt "Static" und Parameterinstanzen im Abschnitt "InOut".

Ergebnis

Die Leerbox wird zum entsprechenden FUP-Element geändert. Für die Parameter werden Platzhalter eingefügt.

Siehe auch

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

Parameterinstanzen (Seite 70)

14.4.4 Datentyp eines FUP-Elements auswählen

14.4.4.1 Auswahl eines Datentyps

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie eine dieser Anweisungen im Programm verwenden, müssen Sie an der konkreten Programmstelle ein für die Anweisung zulässiger Datentyp festlegen. Bei einigen Anweisungen müssen Sie den Datentypen für die Eingänge und für die Ausgänge separat auswählen.

Hinweis

Der zulässige Datentyp (BOOL) für die Variablen am Freigabeeingang EN und am Freigabeausgang ENO ist vom System vorgegeben und kann nicht geändert werden.

Die für eine Anweisung zulässigen Datentypen sind in der Klappliste der Anweisung aufgeführt. Durch die Auswahl eines Eintrags aus der Klappliste bestimmen Sie den Datentyp der Anweisung. Wenn der Datentyp eines Operanden vom Datentyp der Anweisung abweicht und nicht implizit konvertiert werden kann, wird der Operand rot angezeigt und ein Rollout mit der entsprechenden Fehlermeldung eingeblendet.

Datentypauswahl von mathematischen Anweisungen

Einige mathematische Anweisungen bieten Ihnen die Möglichkeit, den Datentyp automatisch passend zu den Datentypen der Operanden einstellen zu lassen. Diese Anweisungen besitzen in der Klappliste für die Datentypauswahl zusätzlich zu den eigentlichen Datentypen noch den Eintrag "Auto". Wenn Sie diesen Eintrag auswählen und den ersten Operanden beschalten, wird der Datentyp des Operanden als Datentyp für die Anweisung verwendet. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)", z. B. "Auto (Real)". Wenn Sie weitere Operanden beschalten, wird der automatisch eingestellte Datentyp der Anweisung nach folgenden Kriterien angepasst:

- Sie versorgen alle weiteren Operanden mit Variablen des gleichen Datentyps:
  Der Datentyp der Anweisung wird nicht geändert.
- Sie versorgen alle weiteren Operanden mit Variablen, deren Datentyp kleiner ist als der Datentyp der Anweisung:
  Der Datentyp der Anweisung wird nicht geändert. Für die Operanden mit dem kleineren Datentyp wird ggf. eine implizite Konvertierung durchgeführt.
- Sie versorgen einen weiteren Operanden mit einer Variablen, deren Datentyp größer ist als der Datentyp der Anweisung:
  Der Datentyp der Anweisung wird auf den größeren Datentyp geändert. Für Operanden, die von dem neu eingestellten Datentyp der Anweisung abweichen, erfolgt ggf. eine implizite Konvertierung.

Jede Datentypänderung eines Operanden kann eine Änderung des Datentyps der Anweisung nach sich ziehen. Dadurch werden möglichweise auch weitere Operanden implizit konvertiert.

Operanden, für die eine implizite Konvertierung durchgeführt wird, werden mit einem grauen Quadrat gekennzeichnet.

Hinweis

Beachten Sie auch die Informationen zur Datentypkonvertierung für Ihr Gerät und hier insbesondere die Hinweise zur IEC-Prüfung.

Siehe auch: Datentypkonvertierung (Seite 507)

Siehe auch

Datentyp einer Anweisung festlegen (Seite 7512)

14.4.4.2 Datentyp einer Anweisung festlegen

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie solche Anweisungen in Ihr Programm einfügen, müssen Sie für diese Anweisungen an der konkreten Programmstelle den Datentyp festlegen.

Datentyp über die Klappliste festlegen

Um den Datentyp einer Anweisung über die Klappliste festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Klicken Sie auf das Dreieck in der oberen Ecke der Klappliste.
   Die Klappliste wird geöffnet und die Datentypen angezeigt, die für die Anweisung zulässig sind.
3. Wählen Sie aus der Klappliste einen Datentyp aus.
   Der ausgewählte Datentyp wird angezeigt.
4. Falls die Anweisung über zwei Klapplisten verfügt, wählen Sie in der linken Klappliste den Datentyp für die Eingänge der Anweisung und in der rechten Klappliste den Datentyp für die Ausgänge der Anweisung.

Datentyp durch Zuweisen von Variablen festlegen

Um den Datentyp einer Anweisung durch Zuweisen von Variablen festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an, deren Datentyp als Datentyp der Anweisung übernommen werden soll.
   Der Datentyp der Variablen wird in der Klappliste angezeigt.
3. Falls sowohl für die Eingänge als auch für die Ausgänge der Anweisung Datentypen festgelegt werden müssen, geben Sie an einem Eingang und an einem Ausgang jeweils eine gültige Variable an. Die am Eingang angegebene Variable bestimmt den Datentyp der Eingänge und die Variable am Ausgang bestimmt den Datentyp der Ausgänge der Anweisung.

Datentyp von mathematischen Anweisungen automatisch festlegen

Um für mathematische Anweisungen den Datentyp automatisch festlegen zu lassen, gehen Sie folgendermaßen vor:

1. Fügen Sie die mathematische Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Wählen Sie in der Klappliste den Eintrag "Auto".
3. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an.
   Der Datentyp der Variablen wird als Datentyp der Anweisung übernommen. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)".

Siehe auch: Auswahl eines Datentyps (Seite 7511)

Siehe auch

Auswahl eines Datentyps (Seite 7511)

14.4.5 Favoriten in FUP verwenden

14.4.5.1 FUP-Elemente zu den Favoriten hinzufügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

FUP-Elemente aus den Favoriten entfernen (Seite 7515)

Übersicht über den Programmiereditor (Seite 7117)

14.4.5.2 FUP-Elemente mithilfe von Favoriten einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

FUP-Elemente aus den Favoriten entfernen (Seite 7515)

14.4.5.3 FUP-Elemente aus den Favoriten entfernen

Voraussetzung

Ein Codebaustein ist geöffnet.

Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

FUP-Elemente zu den Favoriten hinzufügen (Seite 7513)

FUP-Elemente mithilfe von Favoriten einfügen (Seite 7514)

Übersicht über den Programmiereditor (Seite 7117)

14.4.6 Bausteinaufrufe in FUP einfügen

14.4.6.1 Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FC) und Funktionsbausteine (FB) per Drag & Drop aus der Projektnavigation einfügen. Wenn Sie Funktionsbausteine aus anderen Funktionsbausteinen aufrufen, können Sie sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Voraussetzung

- Ein Netzwerk ist vorhanden.
- Der Baustein, der aufgerufen werden soll, ist vorhanden.

Aufruf einer Funktion (FC) einfügen

Um den Aufruf einer Funktion (FC) per Drag & Drop in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das gewünschte Netzwerk.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das gewünschte Netzwerk. Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
     Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie im Eingabefeld "ProDiag-FB" den Überwachungen einen ProDiag-Funktionsbaustein zu.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
3. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter können Sie anschließend versorgen.

Siehe auch: Parameterübergabe beim Bausteinaufruf (Seite 78)

Hinweis

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Instanztyp ändern (Seite 7520)

Einzelinstanzen (Seite 65)

Multiinstanzen (Seite 67)

14.4.6.2 Bausteinaufrufe in FUP aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben die folgenden Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren aller inkonsistenten Bausteinaufrufe im Programmiereditor.
  Die inkonsistenten Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Explizites Aktualisieren eines Bausteinaufrufs im Programmiereditor.
  Der Dialog "Schnittstellenabgleich" wird angezeigt. In diesem Dialog haben Sie die Möglichkeit, die Beschaltung der Operanden der neuen Schnittstelle zu ändern. Anschließend wird der inkonsistente Aufruf dieses Bausteins aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert. Beachten Sie, dass Sie beim Aufruf von Funktionen (FCs) vor dem nächsten Übersetzungsvorgang alle neuen Formalparameter mit Aktualparametern versorgen müssen.

Alle inkonsistenten Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein alle Bausteinaufrufe zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Einen bestimmten Bausteinaufruf im Programmiereditor aktualisieren

Um einen bestimmten Bausteinaufruf im Programmiereditor zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf, den Sie aktualisieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Bausteinaufruf aktualisieren".
   Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden Ihnen die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.

4. Ändern Sie bei Bedarf die Beschaltung der Operanden. Dazu haben Sie die folgenden Möglichkeiten:
   - Sie können einen Operanden entweder per Drag & Drop oder über Ausschneiden/ Kopieren und Einfügen aus der alten Schnittstelle in die neue Schnittstelle übernehmen.
   - Sie können einen Operanden löschen.
   - Sie können einen Operanden umbenennen.
   - Sie können über die Autovervollständigung einen neuen Operanden angeben.
5. Klicken Sie auf "OK", um den Bausteinaufruf zu aktualisieren. Wenn Sie die Aktualisierung abbrechen möchten, klicken Sie auf "Abbrechen".

Hinweis

Beachten Sie, dass Ihnen der Befehl "Bausteinaufruf aktualisieren" nur zur Verfügung steht, wenn Sie nicht zuvor alle Bausteinaufrufe im Editor über den Befehl "Inkonsistente Bausteinaufrufe aktualisieren" aktualisiert haben.

Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Bausteinaufrufe per Drag & Drop einfügen (Seite 7516)

Instanztyp ändern (Seite 7520)

Bausteinaufrufe (Seite 62)

14.4.6.3 Bausteinaufruf ändern

Sie haben die Möglichkeit, für einen Bausteinaufruf den aufgerufenen Baustein zu ändern. Beachten Sie jedoch, dass dabei keine neuen Instanz-Datenbausteine erzeugt werden, z. B. beim Wechsel von einer Funktion (FC) zu einem Funktionsbaustein (FB).

Vorgehen

Um den aufgerufenen Baustein eines Bausteinaufrufs zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb des Bausteinaufrufs auf den Namen des aufgerufenen Bausteins und drücken Sie die Taste <F2>. Oder klicken Sie doppelt auf den Namen des aufgerufenen Bausteins.
   Ein Eingabefeld wird geöffnet, der Name des aktuell aufgerufenen Bausteins ist markiert.
2. Geben Sie den Namen des Bausteins ein, den Sie aufrufen möchten, oder wählen Sie einen Baustein in der Autovervollständigung aus.
3. Wenn Sie einen FB aufrufen möchten, erstellen Sie ggf. einen neuen Instanz-Datenbaustein und geben Sie ihn als Operand an.

14.4.6.4 Instanztyp ändern

Instanztyp

Funktionsbausteine können Sie auf verschiedene Arten aufrufen:

- Als Einzelinstanz
- Als Multiinstanz
- Als Parameterinstanz

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Sie können einen einmal definierten Instanztyp jederzeit ändern.

Voraussetzung

Das Anwenderprogramm enthält einen Bausteinaufruf.

Vorgehen

Um den Instanztyp eines Funktionsbausteins zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Codebaustein und selektieren Sie den Bausteinaufruf.
2. Wählen Sie im Kontextmenü den Befehl "Instanz ändern".
   Der Dialog "Aufrufoptionen" wird geöffnet.

3. Klicken Sie auf die Schaltfläche "Einzel-Instanz", "Multi-Instanz" oder "Parameter-Instanz".
   - Wenn Sie als Instanztyp "Einzel-Instanz" wählen, geben Sie einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
   - Wenn Sie als Instanztyp "Multi-Instanz" wählen, geben Sie im Textfeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie als Instanztyp "Parameter-Instanz" wählen, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Hinweis

Die vorherigen Instanzen werden nicht automatisch gelöscht.

Siehe auch

Bausteinaufrufe per Drag & Drop einfügen (Seite 7516)

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Instanzen (Seite 64)

14.4.7 Komplexe FUP-Anweisungen einfügen

14.4.7.1 Anweisung "Berechnen" verwenden

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um die Anweisung "Berechnen" zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Mathematische Funktionen > CALCULATE".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.
   Die Anweisung "Berechnen" wird mit einem Platzhalterausdruck und Fragezeichen für den Datentyp eingefügt.
4. Geben Sie den Datentyp für die Berechnung ein.

5. Geben Sie die Operanden für die Berechnung ein.

   Hinweis

   Die Berechnung wird mit den Eingängen der Anweisung "Berechnen" durchgeführt. Wenn Sie Konstanten verwenden möchten, müssen Sie daher auch für die Konstanten entsprechende Eingänge einfügen.

6. Klicken Sie auf die Schaltfläche "Anweisung 'Berechnen' bearbeiten", um den Platzhalterausdruck durch den korrekten Ausdruck zu ersetzen.
   Der Dialog "Anweisung 'Berechnen' bearbeiten" wird geöffnet.
7. Geben Sie in das Textfeld "OUT:= " den gewünschten Ausdruck ein.

   Hinweis

   Im Bereich "Beispiel" finden Sie ein Beispiel für einen gültigen Ausdruck und mögliche Anweisungen, die Sie verwenden können.

   Um z. B. mithilfe des Pythagoras einen Wert zu ermitteln, geben Sie "OUT := SQRT (SQR (IN1) + SQR (IN2))" ein.

8. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

CALCULATE: Berechnen (Seite 1111)

14.4.8 Freie Kommentare verwenden

14.4.8.1 Grundlagen zur Verwendung von freien Kommentaren in FUP

Einführung

Freie Kommentare ermöglichen das Kommentieren des Quellcodes für grafische Programmiersprachen ähnlich wie Zeilenkommentare für textuelle Sprachen.

Freie Kommentare können für alle nicht-binären Boxen verwendet werden.

Programmcode auskommentieren

Um in den Programmiersprachen KOP und FUP einzelne Anweisungen oder ganze Programmteile zu deaktivieren, müssen Sie parallel zu der Anweisung oder in Reihe zu einem Anweisungsblock ein Bit setzen. Das Auskommentieren wie in AWL und SCL ist in KOP und FUP nicht möglich.

Beispiele zum Auskommentieren von Programmcode in allen Programmiersprachen finden Sie im Siemens Industry Online Support unter folgender FAQ ID: 109482004 (https://support.industry.siemens.com/cs/ww/de/view/109482004)

Siehe auch

Freie Kommentare einfügen (Seite 7523)

Freie Kommentare bearbeiten (Seite 7523)

Freie Kommentare löschen (Seite 7525)

14.4.8.2 Freie Kommentare einfügen

Voraussetzung

Ein Netzwerk mit Anweisungen ist vorhanden.

Vorgehen

Um einen freien Kommentar zu einer Anweisung hinzuzufügen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie ggf. in der Funktionsleiste die Schaltfläche "Freie Kommentare ein/aus".
2. Klicken Sie mit der rechten Maustaste auf die Anweisung, für die Sie einen freien Kommentar einfügen möchten.
3. Wählen Sie den Befehl "Kommentar einfügen" aus dem Kontextmenü.
   Eine Kommentarbox mit einem Standardkommentar wird geöffnet. Die Kommentarbox ist über einen Pfeil mit der zugehörigen Anweisung verbunden.
4. Geben Sie in die Kommentarbox den gewünschten Kommentar ein.

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 7522)

Freie Kommentare bearbeiten (Seite 7523)

Freie Kommentare löschen (Seite 7525)

14.4.8.3 Freie Kommentare bearbeiten

Einführung

Freie Kommentare können folgendermaßen bearbeitet werden:

- Kommentartext ändern
- Platzierung oder Größe der Kommentarbox ändern
- Kommentar an ein anderes Element anhängen
- Freie Kommentare ein- und ausblenden

Kommentartext ändern

Um den Text von freien Kommentaren zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Kommentarbox.
2. Geben Sie den gewünschten Text ein.

Platzierung der Kommentarbox ändern

Um die Platzierung der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox und halten Sie die Maustaste gedrückt.
2. Ziehen Sie die Kommentarbox an die gewünschte Stelle.

Größe der Kommentarbox ändern

Um die Größe der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox.
2. Ziehen Sie die Kommentarbox am Anfasser in der unteren rechten Ecke auf die gewünschte Größe.

Kommentar an ein anderes Element anhängen

Um einen freien Kommentar an ein anderes Element zu hängen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Pfeilspitze, die die Kommentarbox mit der Anweisung verbindet, und halten Sie die Maustaste gedrückt.
2. Ziehen Sie den Pfeil auf das Element, an das Sie den Kommentar anhängen möchten. Mögliche Einfügestellen werden mit einem grünen Quadrat gekennzeichnet.
3. Lassen Sie die Maustaste los.

Freie Kommentare ein- und ausblenden

Um freie Kommentare ein- und auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Freie Kommentare ein/aus".

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 7522)

Freie Kommentare einfügen (Seite 7523)

Freie Kommentare löschen (Seite 7525)

14.4.8.4 Freie Kommentare löschen

Vorgehen

Um einen freien Kommentar zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den freien Kommentar, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 7522)
Freie Kommentare einfügen (Seite 7523)
Freie Kommentare bearbeiten (Seite 7523)

14.5 FUP-Elemente bearbeiten

14.5.1 FUP-Elemente selektieren

Sie können mehrere einzelne Elemente oder alle Elemente eines Netzwerks selektieren.

Voraussetzung

FUP-Elemente sind vorhanden.

Mehrere einzelne FUP-Elemente selektieren

Um mehrere einzelne FUP-Elemente zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie die FUP-Elemente an, die Sie selektieren möchten.
3. Lassen Sie die Taste <Strg> los.

Alle FUP-Elemente eines Netzwerks selektieren

Um alle FUP-Elemente eines Netzwerks zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Netzwerk, dessen Elemente Sie selektieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

Siehe auch

FUP-Elemente kopieren (Seite 7526)

FUP-Elemente ausschneiden (Seite 7527)

FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)

FUP-Elemente ersetzen (Seite 7529)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

FUP-Elemente löschen (Seite 7532)

14.5.2 FUP-Elemente kopieren

Voraussetzung

Ein FUP-Element ist vorhanden.

Vorgehen

Um ein FUP-Element zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Ergebnis

Das FUP-Element wird kopiert und in der Zwischenablage gespeichert.

Siehe auch

FUP-Elemente selektieren (Seite 7526)
FUP-Elemente ausschneiden (Seite 7527)
FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)
FUP-Elemente ersetzen (Seite 7529)
Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)
Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)
FUP-Elemente löschen (Seite 7532)

14.5.3 FUP-Elemente ausschneiden

Voraussetzung

Ein FUP-Element ist vorhanden.

Ausschneiden

Um ein FUP-Element auszuschneiden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

Ergebnis

Das FUP-Element wird ausgeschnitten und in der Zwischenablage gespeichert.

Siehe auch

FUP-Elemente selektieren (Seite 7526)
FUP-Elemente kopieren (Seite 7526)
FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)
FUP-Elemente ersetzen (Seite 7529)
Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)
Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)
FUP-Elemente löschen (Seite 7532)

14.5.4 FUP-Elemente aus der Zwischenablage einfügen

Voraussetzung

Ein FUP-Element ist vorhanden.

Vorgehen

Um ein FUP-Element aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie ein FUP-Element oder schneiden Sie ein FUP-Element aus.
2. Klicken Sie mit der rechten Maustaste auf die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

FUP-Elemente selektieren (Seite 7526)
FUP-Elemente kopieren (Seite 7526)
FUP-Elemente ausschneiden (Seite 7527)
FUP-Elemente ersetzen (Seite 7529)
Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)
Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)
FUP-Elemente löschen (Seite 7532)

14.5.5 FUP-Elemente ersetzen

Sie können FUP-Elemente ganz einfach gegen andere FUP-Elemente des gleichen Typs austauschen. Dies hat den Vorteil, dass die Parameter bestehen bleiben und nicht erneut eingegeben werden müssen. Sie können z. B. ODER und UND, RS-FlipFlop und SR-FlipFlop, Vergleichsfunktionen oder Sprunganweisungen gegeneinander austauschen.

Voraussetzung

Ein Netzwerk mit mindestens einem FUP-Element ist vorhanden.

Vorgehen

Um ein FUP-Element durch ein anderes FUP-Element zu ersetzen, gehen Sie folgendermaßen vor:

1. Markieren Sie das FUP-Element, das Sie ersetzen möchten.
   Wenn zu dem gewählten FUP-Element kompatible Elemente vorhanden sind, erscheint in der oberen rechten Ecke des Elements ein Dreieck.
2. Bewegen Sie den Mauszeiger über das Dreieck des FUP-Elements.
   Eine Klappliste wird geöffnet.
3. Wählen Sie das FUP-Element aus der Klappliste, durch das Sie das aktuelle FUP-Element ersetzen möchten.

Siehe auch

FUP-Elemente selektieren (Seite 7526)

FUP-Elemente kopieren (Seite 7526)

FUP-Elemente ausschneiden (Seite 7527)

FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

FUP-Elemente löschen (Seite 7532)

14.5.6 Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen

Einführung

Sie können einige FUP-Elemente, die arithmetische oder binäre Operationen durchführen, durch zusätzliche Eingänge erweitern. Solche Elemente sind z. B. die Anweisungen "Addieren" (ADD), "Multiplizieren" (MUL), UND oder ODER. Durch zusätzliche Ausgänge können Sie die Boxen der Anweisungen "Wert kopieren" (MOVE) und "Demultiplexen" (DEMUX) erweitern.

Der Name der neuen Ein- und Ausgänge setzt sich aus der Art des eingefügten Elements und einer fortlaufenden Nummer zusammen. Der Name eines neuen Eingangs lautet z. B. "IN2", der Name eines neuen Ausgangs z. B. "OUT2".

Voraussetzung

Ein FUP-Element ist vorhanden, das das Einfügen von zusätzlichen Eingängen bzw. Ausgängen erlaubt.

Zusätzlichen Eingang einfügen

Um die Box eines FUP-Elements durch einen zusätzlichen Eingang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Eingang des FUP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Eingang einfügen".
   Die Box des FUP-Elements wird durch einen zusätzlichen Eingang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.
   Die Box des FUP-Elements wird durch einen zusätzlichen Eingang erweitert.

Zusätzlichen Ausgang einfügen

Um die Box eines FUP-Elements durch einen zusätzlichen Ausgang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Ausgang des FUP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Ausgang einfügen".
   Die Box des FUP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Ausgang der Anweisungsbox.
   Die Box des FUP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente selektieren (Seite 7526)

FUP-Elemente kopieren (Seite 7526)

FUP-Elemente ausschneiden (Seite 7527)

FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)

FUP-Elemente ersetzen (Seite 7529)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

FUP-Elemente löschen (Seite 7532)

14.5.7 Eingänge und Ausgänge einer Anweisung entfernen

Einführung

Eingänge oder Ausgänge, die Sie für eine Anweisung zusätzlich eingefügt haben, können wieder entfernt werden.

Voraussetzung

Ein FUP-Element ist vorhanden, das Sie durch zusätzliche Ein- oder Ausgänge erweitert haben.

Eingang entfernen

Um einen Eingang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eingang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Eingang des FUP-Elements wird entfernt.

Ausgang entfernen

Um einen Ausgang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ausgang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Ausgang des FUP-Elements wird entfernt.

Siehe auch

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente selektieren (Seite 7526)

FUP-Elemente kopieren (Seite 7526)

FUP-Elemente ausschneiden (Seite 7527)

FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)

FUP-Elemente ersetzen (Seite 7529)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)

EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

FUP-Elemente löschen (Seite 7532)

14.5.8 FUP-Elemente löschen

Voraussetzung

Ein FUP-Element ist vorhanden.

Vorgehen

Um ein FUP-Element zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

FUP-Elemente einfügen (Seite 7507)
FUP-Elemente selektieren (Seite 7526)
FUP-Elemente kopieren (Seite 7526)
FUP-Elemente ausschneiden (Seite 7527)
FUP-Elemente aus der Zwischenablage einfügen (Seite 7528)
FUP-Elemente ersetzen (Seite 7529)
Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 7529)
Eingänge und Ausgänge einer Anweisung entfernen (Seite 7531)
EN-/ENO-Mechanismus in KOP und FUP aktivieren und deaktivieren (Seite 157)

14.6 Operanden in FUP-Anweisungen beschalten

14.6.1 Operanden beschalten

Beim Einfügen eines FUP-Elements werden die Zeichenfolgen "<???>", "<??.?>" und "..." als Platzhalter für die Parameter eingefügt. Die rot dargestellten Zeichenfolgen "<???>" und "<??.?>" kennzeichnen Parameter, die beschaltet werden müssen. Die schwarz dargestellte Zeichenfolge "..." kennzeichnet Parameter, die beschaltet werden können. "<??.?>" steht für boolesche Platzhalter. Zudem können FUP-Elemente die folgenden unterschiedlichen Anschlüsse für die Parameter besitzen:

- Orangefarbene Anschlüsse: An diesem Anschluss können nur Variablen oder Konstanten verschaltet werden.
- Schwarze Anschlüsse: Boolescher Ein- oder Ausgang, an dem auch andere Elemente vorverschaltet oder nachgeschaltet werden können.

Hinweis

Wenn Sie den Mauszeiger über den Platzhalter bewegen, werden die möglichen Datentypen in einem Tooltip angezeigt.

Voraussetzung

Ein FUP-Element ist vorhanden.

Vorgehen

Um die Parameter eines FUP-Elements zu beschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Platzhalter des Parameters.
   Ein Eingabefeld wird geöffnet.
2. Geben Sie den entsprechenden Parameter ein, z. B. eine PLC-Variable, eine lokale Variable oder eine Konstante.

   Hinweis

   Wenn Sie die absolute Adresse eines Parameters eingeben, der bereits definiert ist, wird diese Absolutadresse in den symbolischen Namen des Parameters geändert, sobald Sie die Eingabe bestätigen. Haben Sie den Parameter noch nicht definiert, wird in der PLC-Variablentabelle eine neue Variable mit dieser Absolutadresse und dem Standardnamen "Tag_1" eingetragen. Bestätigen Sie Ihre Eingabe, wird die Absolutadresse gegen den symbolischen Namen "Tag_1" ausgetauscht.

3. Bestätigen Sie den Parameter mit der Eingabetaste.
4. Wenn Sie den Parameter bisher nicht definiert haben, können Sie dies über das Kontextmenü direkt im Programmiereditor erledigen.
   Siehe auch: "PLC-Variablen im Programmiereditor deklarieren (Seite 7383)".

Oder per Drag & Drop aus der PLC-Variablentabelle:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" und öffnen Sie die PLC-Variablentabelle.
2. Wenn Sie die PLC-Variablentabelle geöffnet haben, ziehen Sie die gewünschte Variable an die entsprechende Stelle in Ihrem Programm. Falls Sie die PLC-Variablentabelle nicht geöffnet haben, öffnen Sie die Detailansicht. Ziehen Sie die gewünschte Variable aus der Detailansicht an die entsprechende Stelle in Ihrem Programm.

Oder per Drag & Drop aus der Bausteinschnittstelle:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ziehen Sie den gewünschten Operanden aus der Bausteinschnittstelle an die entsprechende Stelle in Ihrem Programm.

Ergebnis

- Wenn die Syntax fehlerfrei ist, wird der Parameter schwarz dargestellt.
- Wenn in der Syntax ein Fehler vorhanden ist, wird das Eingabefeld nicht verlassen und es wird im Inspektorfenster im Register "Info > Syntax" eine entsprechende Fehlermeldung angezeigt.

Siehe auch

Operanden verwenden und adressieren (Seite 93)

Autovervollständigung verwenden (Seite 7147)

14.6.2 Versteckte Parameter beschalten

Einführung

Wenn eine Anweisung versteckte Parameter enthält, verfügt die Anweisungsbox über einen kleinen Pfeil am unteren Rand. Die versteckten Parameter erkennen Sie an der weißen Schriftfarbe.

Sie können die versteckten Parameter jederzeit einblenden und beschalten.

Versteckte Parameter ein- oder ausblenden

Um versteckte Parameter ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten am unteren Rand der Anweisungsbox, um versteckte Parameter einzublenden.
2. Klicken Sie auf den Pfeil-nach-oben am unteren Rand der Anweisungsbox, um versteckte Parameter auszublenden.

Versteckte Parameter beschalten

Um verstecke Parameter zu beschalten, gehen Sie folgendermaßen vor:

1. Beschalten Sie die versteckten Parameter wie normale sichtbare Parameter.
   Durch das Beschalten des Parameters kann sich dessen Sichtbarkeit ändern:
   - Der Parameter wird sichtbar, wenn Sie in den Eigenschaften die Option "Verstecken, wenn kein Parameter zugewiesen ist" angewählt haben.
   - Der Parameter bleibt versteckt, wenn Sie in den Eigenschaften die Option "Verstecken" angewählt haben.

Siehe auch

Parameter beim Bausteinaufruf verstecken (Seite 7284)

14.6.3 Variableninformationen ein- oder ausblenden

Einführung

Sie können die folgenden Informationen zu den verwendeten Variablen im Programmiereditor einblenden:

- Name der Variablen
- Adresse der Variablen
- Einfache oder hierarchische Kommentare zur Dokumentation der Variablen

Die Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablentabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden, werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet. Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete Bausteine wieder einblenden.

Wenn Sie die Anzeige von Variableninformationen mit hierarchischen Kommentaren wählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt. Die Anzeige erfolgt dabei in Klammern hinter dem Kommentar der Variablen, die Kommentare der einzelnen Ebenen werden jeweils durch einen Punkt getrennt. Falls auf einer Strukturebene für eine Variable kein Kommentar vorhanden ist, entfällt er in der Anzeige, was Sie an zwei aufeinander folgenden Punkten erkennen können.

Sie können die Variableninformationen entweder gesammelt unterhalb eines KOP-/FUP-Netzwerks anzeigen lassen oder direkt am jeweiligen Operanden. Wie beim Ein- und Ausblenden der Variableninformationen können Sie dies über die Einstellungen für alle Bausteine einstellen oder speziell für einen geöffneten Baustein ändern.

Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Variableninformationen" entweder die Option "Einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
4. Legen Sie über die Klappliste "Position der Variableninformation (KOP/FUP)" fest, ob die Variableninformationen unterhalb eines KOP-/FUP-Netzwerks oder am Operanden angezeigt werden soll.
   Die Variableninformationen werden für alle geöffneten Bausteine eingeblendet und an der gewählten Position angezeigt. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls eingeblendet.
5. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Variableninformationen" die Option "Ausblenden".
   Die Variableninformationen werden für alle geöffneten Bausteine ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls ausgeblendet.

Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." entweder die Option "Variableninformationen einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
2. Um die Position der Variableninformationen festzulegen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Position der Variableninformation (KOP/FUP)". Mit jedem Klick auf die Schaltfläche ändert sich die Position. Alternativ können Sie auf den kleinen Pfeil neben der Schaltfläche "Position der Variableninformation (KOP/FUP)" klicken, um eine Klappliste zu öffnen. Wählen Sie dann die Position aus der Klappliste aus.
   Die Variableninformationen werden eingeblendet und an der gewählten Position angezeigt.
3. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." die Option "Variableninformationen ausblenden".
   Die Variableninformationen werden ausgeblendet.

14.7 Zweige in FUP

14.7.1 Grundlagen zu Zweigen in FUP

Definition

Sie können mit der Programmiersprache Funktionsplan (FUP) Parallelzweige programmieren. Dazu verwenden Sie Zweige, die Sie zwischen den Boxen einfügen. Sie können in den Zweig weitere Boxen einfügen und auf diese Weise komplexe Funktionspläne programmieren.

Das folgende Bild zeigt ein Beispiel für die Verwendung von Zweigen:

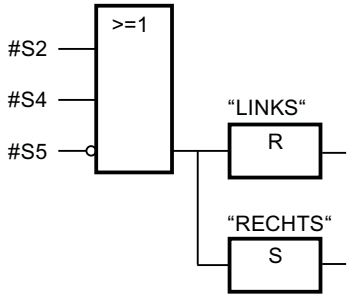

Siehe auch

Regeln für Zweige in FUP (Seite 7537)

Zweige in FUP-Netzwerke einfügen (Seite 7538)

Zweige in FUP-Netzwerke löschen (Seite 7538)

14.7.2 Regeln für Zweige in FUP

Regeln

Für die Verwendung von Zweigen in FUP gelten folgende Regeln:

- Zweige werden nach unten geöffnet.
- Zweige können nur zwischen zwei FUP-Elementen eingefügt werden.
- Um einen Zweig zu löschen, müssen Sie alle FUP-Elemente löschen, einschließlich des Zweigs selbst.
- Wenn Sie die Verbindung zwischen zwei Zweigen löschen, werden die FUP-Elemente des aufgebrochenen Zweiges frei im Netzwerk platziert.

Siehe auch

Grundlagen zu Zweigen in FUP (Seite 7537)
Zweige in FUP-Netzwerke einfügen (Seite 7538)
Zweige in FUP-Netzwerke löschen (Seite 7538)

14.7.3 Zweige in FUP-Netzwerke einfügen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um einen neuen Zweig in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Abzweig".
3. Ziehen Sie das Element per Drag & Drop an der gewünschten Stelle auf eine Verbindungslinie zwischen zwei Boxen.

Siehe auch

Regeln für Zweige in FUP (Seite 7537)
Grundlagen zu Zweigen in FUP (Seite 7537)
Zweige in FUP-Netzwerke löschen (Seite 7538)

14.7.4 Zweige in FUP-Netzwerke löschen

Voraussetzung

Ein Zweig ist vorhanden.

Vorgehen

Um einen Zweig zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die den Zweig mit dem Hauptzweig verbindet.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Der Zweig wird gelöscht. Am gelöschten Zweig angeschlossene Boxen werden frei im Netzwerk platziert.

Siehe auch

Regeln für Zweige in FUP (Seite 7537)

Grundlagen zu Zweigen in FUP (Seite 7537)

Zweige in FUP-Netzwerke einfügen (Seite 7538)

14.8 Verknüpfungspfade in FUP

14.8.1 Grundlagen zu Verknüpfungspfaden in FUP

Verwendung von Verknüpfungspfaden

Das Anwenderprogramm wird in einem oder mehreren Netzwerken abgebildet. Innerhalb der Netzwerke können sich ein oder mehrere Verknüpfungspfade befinden, auf denen die binären Signale in Form von Boxen angeordnet sind.

Das folgende Bild zeigt ein Beispiel für die Verwendung von mehreren Verknüpfungspfaden innerhalb eines Netzwerks:

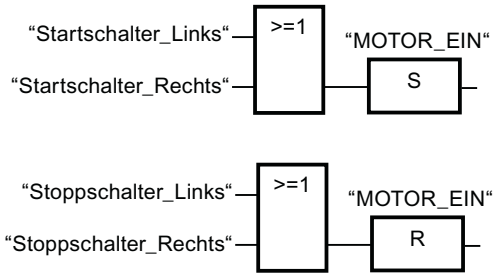

Regeln

Beachten Sie folgende Regeln bei der Verwendung von Verknüpfungspfaden:

- Verbindungen zwischen Verknüpfungspfaden sind nicht erlaubt.
- Es ist nur eine Sprunganweisung pro Netzwerk zulässig. Die Platzierungsregeln für Sprunganweisungen bleiben dabei erhalten.

Abarbeitung von Verknüpfungspfaden

Die Abarbeitung von Verknüpfungspfaden und Netzwerken erfolgt von oben nach unten und von links nach rechts. Das bedeutet, dass zuerst die erste Anweisung im ersten Verknüpfungspfad im ersten Netzwerk bearbeitet wird. Anschließend werden alle Anweisungen dieses Verknüpfungspfads bearbeitet. Danach kommen alle weiteren Verknüpfungspfade des ersten Netzwerks an die Reihe. Erst wenn alle Verknüpfungspfade abgearbeitet wurden, wird das nächste Netzwerk bearbeitet.

Bei der Verwendung von Sprüngen wird die reguläre Abarbeitung der Verknüpfungspfade umgangen und es wird die Anweisung am Sprungziel bearbeitet.

Unterschiede zwischen Zweigen und Verknüpfungspfaden

Der Unterschied zwischen Zweigen und Verknüpfungspfaden besteht darin, dass die Verknüpfungspfade eigenständige Zweige sind, die auch in einem anderen Netzwerk stehen können. Zweige hingegen erlauben die Programmierung einer Parallelschaltung und haben eine gemeinsame Vorverknüpfung.

Siehe auch

Verknüpfungspfad einfügen (Seite 7541)

Verknüpfungspfad löschen (Seite 7541)

14.8.2 Verknüpfungspfad einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Ein Netzwerk ist vorhanden.

Vorgehen

Um einen neuen Verknüpfungspfad in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie eine beliebige Anweisung so in ein Netzwerk ein, dass sie keine Verbindung zu vorhandenen Anweisungen besitzt.
   Ein neuer Verknüpfungspfad wird eingefügt.
2. Fügen Sie eine Zuweisung am Ende des neuen Verknüpfungspfads ein.
3. Fügen Sie weitere Anweisungen in den neuen Verknüpfungspfad ein.

Siehe auch

Grundlagen zu Verknüpfungspfaden in FUP (Seite 7540)

Verknüpfungspfad löschen (Seite 7541)

14.8.3 Verknüpfungspfad löschen

Voraussetzung

Ein Verknüpfungspfad ist vorhanden.

Vorgehen

Um einen Verknüpfungspfad zu löschen, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen um den Verknüpfungspfad. Achten Sie darauf, dass Sie dabei alle Anweisungen des Verknüpfungspfads selektieren. Alternativ können Sie auch mit gedrückter <Shift>-Taste die erste und letzte Anweisung des Verknüpfungspfads selektieren.
2. Klicken Sie mit der rechten Maustaste auf eine der Anweisungen im Verknüpfungspfad.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zu Verknüpfungspfaden in FUP (Seite 7540)

Verknüpfungspfad einfügen (Seite 7541)

14.9 Programmierbeispiele FUP

14.9.1 Beispiel für das Steuern eines Förderbands

Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Drucktaster, S1 für START und S2 für STOP. Am Ende des Förderbands befinden sich ebenfalls zwei Drucktaster, S3 für START und S4 für STOP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.

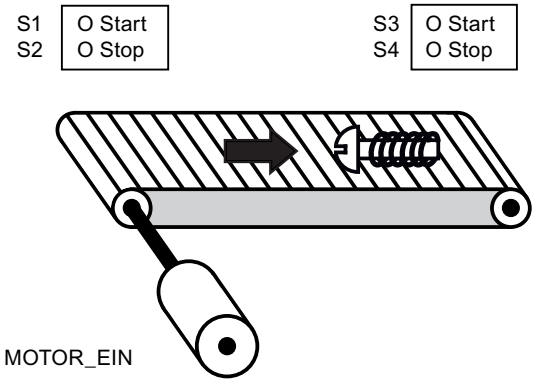

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
Starttaster_Links (S1)	Input	BOOL	Starttaster am linken Rand des Förderbands
Stoptaster_Links (S2)	Input	BOOL	Stoptaster am linken Rand des Förderbands
Starttaster_Rechts (S3)	Input	BOOL	Starttaster am rechten Rand des Förderbands
Stoptaster_Rechts (S4)	Input	BOOL	Stoptaster am rechten Rand des Förderbands
MOTOR_EIN	Output	BOOL	Motor des Förderbands einschalten

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Lösung der Aufgabe:

Netzwerk 1:

Der Motor des Förderbands wird eingeschaltet, wenn der Starttaster "S1" oder "S3" betätigt wird.

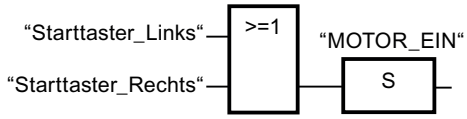

Netzwerk 2:

Der Motor des Förderbands wird ausgeschaltet, wenn der Stoptaster "S2" oder "S4" betätigt wird.

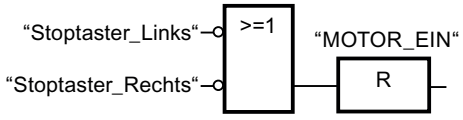

Siehe auch

Grundlagen zu FUP (Seite 7493)

Einstellungen zu FUP (Seite 7496)

Arbeiten mit Netzwerken (Seite 7498)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zweige in FUP (Seite 7537)

Verknüpfungspfade in FUP (Seite 7540)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Beispiel für das Berechnen einer Gleichung (Seite 7551)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

14.9.2 Beispiel für das Erfassen der Richtung eines Förderbands

Erfassen der Richtung eines Förderbands

Die erkannte Laufrichtung des Bandes wird mit einem Pfeil RECHTS oder einem Pfeil LINKS angezeigt. Läuft weiteres Fördergut von rechts an LS1 oder von links an LS2 heran, wird zunächst der angezeigte Pfeil ausgeschaltet, bis nach dem Durchlaufen beider Lichtschranken die Laufrichtung wieder neu festgestellt wurde und der entsprechende Pfeil angezeigt werden kann. Für die Lösung der Aufgabe sind noch 2 Flankenmerker notwendig, die den Signalwechsel an den zwei Lichtschranken von "0" auf "1" erkennen.

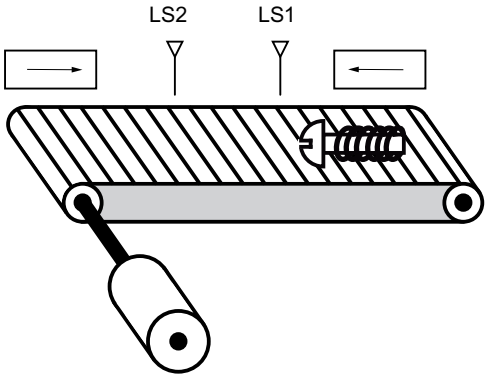

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
RECHTS	Output	BOOL	Anzeige bei einer Bewegung nach rechts
LINKS	Output	BOOL	Anzeige bei einer Bewegung nach links
TM1	Input	BOOL	Flankenmerker 1
TM2	Input	BOOL	Flankenmerker 2

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Lösung der Aufgabe:

Netzwerk 1:

Wenn an der Lichtschranke "LS1" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalflanke) und gleichzeitig der Signalzustand an "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links.

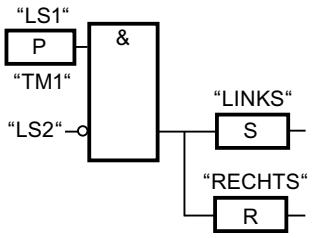

Netzwerk 2:

Wenn an der Lichtschranke "LS2" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalflanke) und gleichzeitig der Signalzustand an "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts.

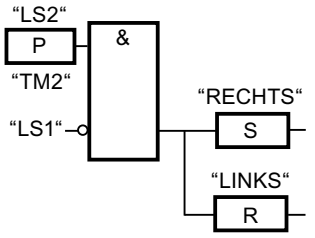

Siehe auch

Grundlagen zu FUP (Seite 7493)

Einstellungen zu FUP (Seite 7496)

Arbeiten mit Netzwerken (Seite 7498)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zweige in FUP (Seite 7537)

Verknüpfungspfade in FUP (Seite 7540)

Beispiel für das Steuern eines Förderbands (Seite 7543)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Beispiel für das Berechnen einer Gleichung (Seite 7551)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

14.9.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, von der sie auf einen LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

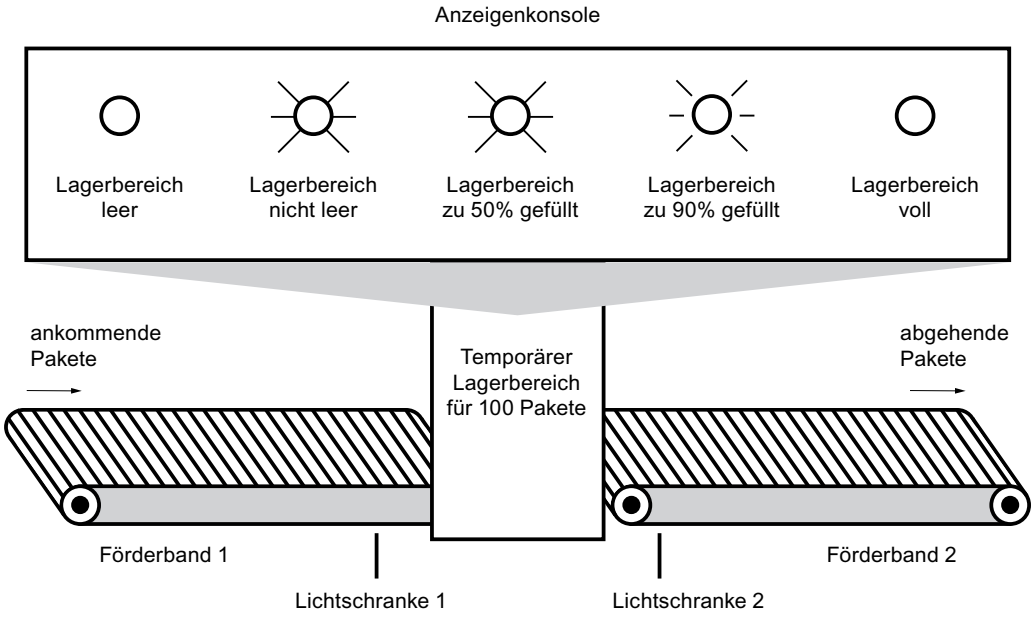

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
RESET	Input	BOOL	Zähler rücksetzen
LOAD	Input	BOOL	Den aktuellen Zählerwert auf den Wert des Parameters PV anpassen.
MAX LAGERFUELL-MENGE	Input	INT	Maximal mögliche Anzahl der Pakete im Lager

Name	Deklaration	Datentyp	Beschreibung
PAKETANZAHL	Output	INT	Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)
LB_PAKETE	Output	BOOL	Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELL-MENGE" ist.
LAG_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich leer
LAG_NICHT_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich nicht leer
LAG_50%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 50 % voll
LAG_90%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 90 % voll
LAG_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich voll
VOLUME_50	Input	INT	Vergleichswert: 50 Pakete
VOLUME_90	Input	INT	Vergleichswert: 90 Pakete
VOLUME_100	Input	INT	Vergleichswert: 100 Pakete

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Aktivierung der Anzeigeleuchten:

Netzwerk 1:

Wenn ein Paket zum Lagebereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "MAX LAGERFUELLMENGE" gesetzt. Solange der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELLMENGE" ist, liefert die Variable "LB_PAKETE" den Signalzustand "1".

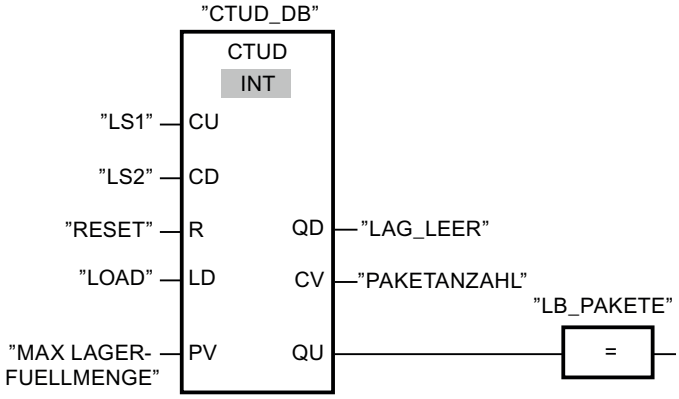

Netzwerk 2:

Solange sich Pakete im Lagerbereich befinden, wird die Variable "LAG_NICHT_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.

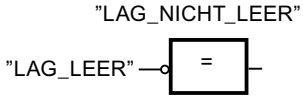

Netzwerk 3:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50 % gefüllt" ein.

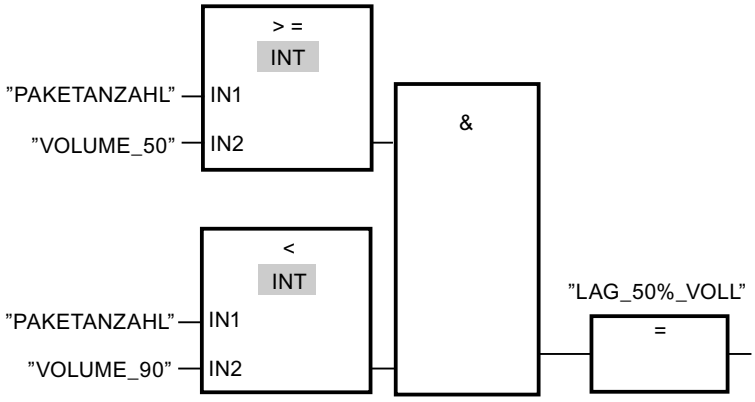

Netzwerk 4:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90 % gefüllt" ein.

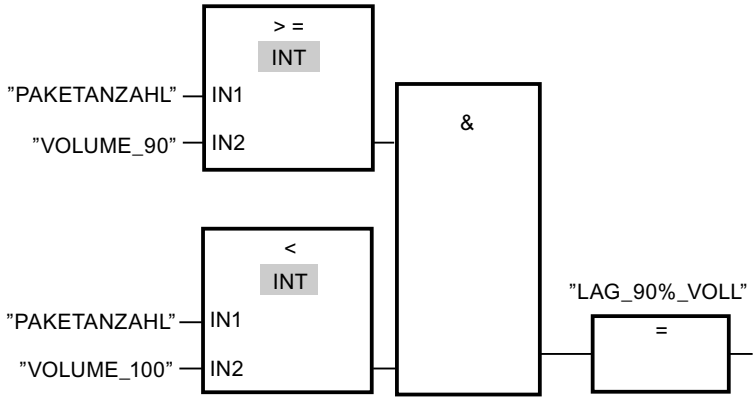

Netzwerk 5:

Wenn die Anzahl der Pakete im Lagerbereich 100 erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein.

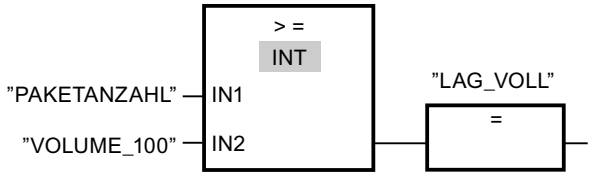

Siehe auch

Grundlagen zu FUP (Seite 7493)

Einstellungen zu FUP (Seite 7496)

Arbeiten mit Netzwerken (Seite 7498)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zweige in FUP (Seite 7537)

Verknüpfungspfade in FUP (Seite 7540)

Beispiel für das Steuern eines Förderbands (Seite 7543)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Beispiel für das Berechnen einer Gleichung (Seite 7551)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

14.9.4 Beispiel für das Berechnen einer Gleichung

Berechnen einer komplexen Gleichung

Das folgende Programmbeispiel zeigt Ihnen, wie Sie mit der Programmiersprache FUP Gleichungen des folgenden Typs lösen können:

X= ((A + B) x C) / D

Grundsätzlich können Sie dies mithilfe von mathematischen Operationen umsetzen. Dafür stehen Ihnen die Anweisungen "ADD", "MUL" und "DIV" zur Verfügung. Setzen Sie eine CPU der Baureihe S7-1200 oder S7-1500 ein, können Sie alternativ die Anweisung "CALCULATE" verwenden. Für diese Anweisung stehen Ihnen abhängig vom gewählten Datentyp unterschiedliche mathematische Operationen zur Verfügung, die Sie miteinander kombinieren können.

Umsetzung mithilfe der Anweisungen "ADD", "MUL" und "DIV"

Es soll die folgende Gleichung berechnet werden:

RESULT = ((A + B) x 15) / E

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Datentyp	Kommentar
A	INT	Erster Wert der Addition
B	INT	Zweiter wert der Addition
C	INT	Erstes Zwischenergebnis
15	INT	Multiplikator
D	INT	Zweites Zwischenergebnis
E	INT	Divisor
RESULT	INT	Endergebnis

Das folgende Netzwerk zeigt die FUP-Programmierung zur Berechnung der Gleichung:

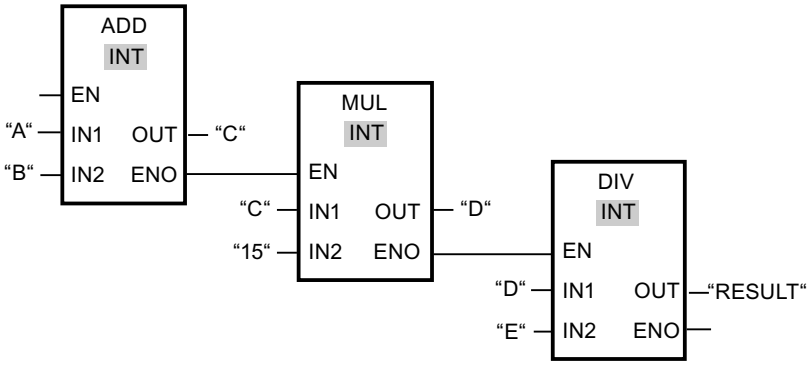

Der Wert des Operanden "A" wird mit dem Wert des Operanden "B" addiert. Die Summe wird im Operanden "C" gespeichert. Der Wert des Operanden "C" wird mit "15" multipliziert. Das Ergebnis der Multiplikation wird im Operanden "D" gespeichert. Anschließend wird der im Operanden "D" gespeicherte Wert durch den Wert des Operanden "E" dividiert. Das Endergebnis wird im Operanden "RESULT" gespeichert.

Umsetzung mithilfe der Anweisung "CALCULATE" (nur S7-1200/1500)

Es soll die folgende Gleichung berechnet werden:

RESULT = ((5 + 10) x 4) / 6

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Kommentar	Wert
A	Input	INT	Erster Wert der Addition	5
B	Input	INT	Zweiter Wert der Addition	10
C	Input	INT	Multiplikator	4
D	Input	INT	Divisor	6
RESULT	Output	INT	Endergebnis	10

Zur Programmierung der Gleichung mithilfe der Anweisung "CALCULATE", gehen Sie folgendermaßen vor:

1. Ziehen Sie die Anweisung "CALCULATE" aus der Task Card "Anweisungen" in ein FUP-Netzwerk.
2. Wählen Sie aus der Klappliste "<???>" den Datentyp INT für die Anweisung aus.
3. Verschalten Sie die in der Bausteinschnittstelle deklarierten Variablen mit den Ein- bzw. Ausgängen der Anweisungsbox.
4. Klicken Sie das "Taschenrechner"-Symbol am rechten oberen Rand der Anweisungsbox an, um die zu berechnende Gleichung einzutragen.
   Der Dialog "Anweisung Berechnen bearbeiten" wird geöffnet.
5. Geben Sie in das Feld "OUT:=" folgenden Ausdruck ein:
   ((IN1 + IN2) * IN4) / IN3
   Die Gleichung wird in der Anweisungsbox dargestellt.
   Das folgende Netzwerk zeigt das Ergebnis in der Programmiersprache FUP:

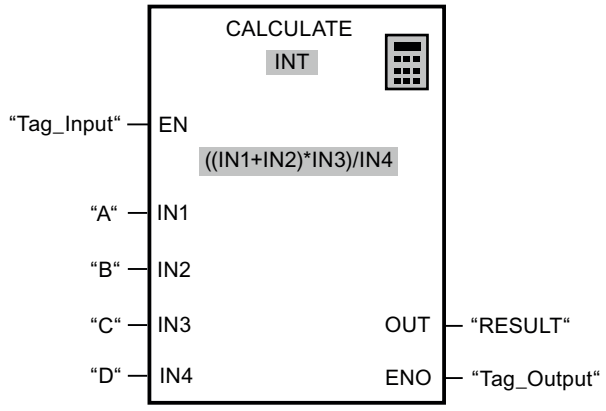

Wenn der Eingang "Tag_Input" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Der Wert des Operanden "A" wird mit dem Wert des Operanden "B" addiert. Die Zwischensumme wird mit "C" multipliziert und anschließend durch den Wert des Operanden "D" dividiert. Das Endergebnis wird im Operanden "RESULT" gespeichert.

Siehe auch

Grundlagen zu FUP (Seite 7493)
Einstellungen zu FUP (Seite 7496)
Arbeiten mit Netzwerken (Seite 7498)
FUP-Elemente einfügen (Seite 7507)
FUP-Elemente bearbeiten (Seite 7526)
Operanden in FUP-Anweisungen beschalten (Seite 7533)
Zweige in FUP (Seite 7537)
Verknüpfungspfade in FUP (Seite 7540)
Beispiel für das Steuern eines Förderbands (Seite 7543)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)
Beispiel für das Kontrollieren der Raumtemperatur (Seite 7553)

14.9.5 Beispiel für das Kontrollieren der Raumtemperatur

Kontrollieren der Raumtemperatur

In einem Kühlraum muss die Temperatur unter null Grad Celsius gehalten werden. Eventuelle Temperaturschwankungen werden mithilfe eines Sensors kontrolliert. Wenn die Temperatur über null Grad Celsius steigt, wird das Kühlsystem für eine voreingestellte Zeit eingeschaltet. Während des Kühlens leuchtet eine Anzeige "Kühlsystem gestartet".

Das Kühlsystem und die Anzeigeleuchte werden ausgeschaltet, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Sensor meldet eine Senkung der Temperatur unter null Grad Celsius.
- Die voreingestellte Kühlzeit ist abgelaufen.
- Der Drucktaster "Stop" wird betätigt.

Wenn die voreingestellte Kühlzeit abgelaufen ist und die Temperatur im Kühlraum immer noch zu hoch ist, kann das Kühlsystem mithilfe des Drucktasters "RESET" erneut gestartet werden.

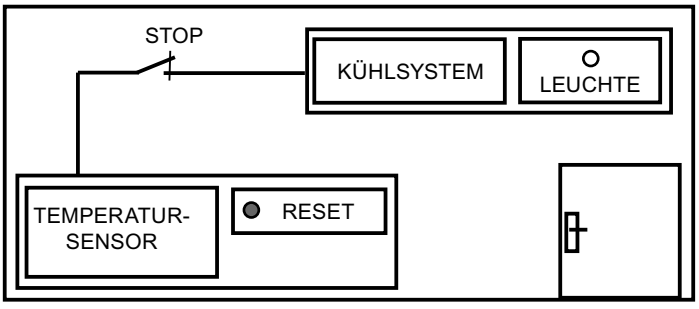

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Kommentar
Sensor	Input	BOOL	Signal des Temperatursensors
RESET	Input	BOOL	Neustart
STOP	Input	BOOL	Das Kühlsystem wird ausgeschaltet.
MaxKuehlzeit	-	TIME	Voreingestellte Kühlzeit Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert.
AktKuehlzeit	-	TIME	Aktuell abgelaufene Kühlzeit Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert.
Kuehlsystem	Output	BOOL	Das Kühlsystem wird eingeschaltet.
Leuchte	Output	BOOL	Die Anzeigeleuchte für die Meldung "Kühlsystem gestartet" wird eingeschaltet.
TempVariable	Temp	BOOL	Temporäre Variable Diese Variable speichert den Signalzustand der IEC-Zeit TP.

Das folgende Netzwerk zeigt die FUP-Programmierung zum Kontrollieren der Raumtemperatur:

Netzwerk 1:

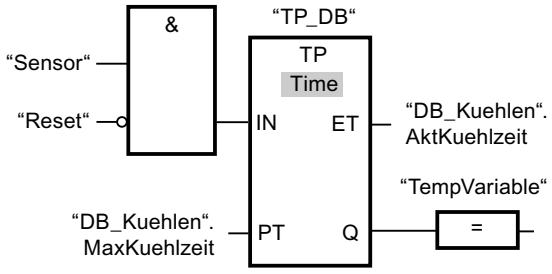

Netzwerk 2:

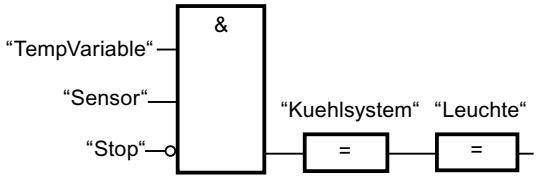

Wenn die Temperatur im Kühlraum höher als null Grad Celsius wird, wechselt der Signalzustand am Operanden "Sensor" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke am Eingang IN der Zeitfunktion wird die voreingestellte Kühlzeit gestartet und die "TempVariable" erhält den Signalzustand "1". Der Signalzustand "1" der "TempVariable" bewirkt im Netzwerk 2, dass das Kühlsystem sowie die Anzeigeleuchte eingeschaltet werden. Die Ausgänge "Sensor", "Kuehlsystem" und "Leuchte" müssen im Netzwerk 2 programmiert werden, da am Ausgang Q der Zeitfunktion nur eine Spule programmiert werden kann.

Sinkt die Temperatur im Kühlraum unter null Grad Celsius, wechselt der Signalzustand des Sensors wieder auf "0". Dadurch werden das Kühlsystem und die Anzeigeleuchte ausgeschaltet.

Wenn der Sensor keine Senkung der Temperatur meldet, werden das Kühlsystem und die Anzeigeleuchte spätestens nach dem Ablauf der voreingestellten Kühlzeit ausgeschaltet. Der Kühlprozess kann in diesem Fall mithilfe des Drucktasters "RESET" neu gestartet werden. Durch das Drücken und Loslassen des Drucktasters wird eine neue positive Signalflanke am Eingang IN erzeugt und dadurch das Kühlsystem neu gestartet.

Das Kühlsystem und die Anzeigeleuchte können jederzeit mithilfe des Drucktasters "STOP" ausgeschaltet werden.

Siehe auch

Grundlagen zu FUP (Seite 7493)

Einstellungen zu FUP (Seite 7496)

Arbeiten mit Netzwerken (Seite 7498)

FUP-Elemente einfügen (Seite 7507)

FUP-Elemente bearbeiten (Seite 7526)

Operanden in FUP-Anweisungen beschalten (Seite 7533)

Zweige in FUP (Seite 7537)

Verknüpfungspfade in FUP (Seite 7540)

Beispiel für das Steuern eines Förderbands (Seite 7543)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7545)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7547)

Beispiel für das Berechnen einer Gleichung (Seite 7551)

AWL-Programme erstellen 15

15.1 Grundlagen zu AWL

15.1.1 Programmiersprache AWL

Übersicht über die Programmiersprache AWL (Anweisungsliste)

AWL ist eine textbasierende Programmiersprache, mit der Sie Codebausteine programmieren können.

Das AWL-Programm wird in Netzwerke gegliedert. Jedes Netzwerk kann eine oder mehrere Zeilen enthalten. Die Nummerierung der Zeilen fängt in jedem Netzwerk mit 1 an und wird mit jeder neuen Zeile aufsteigend fortgesetzt. In den Zeilen eines Netzwerks werden die einzelnen AWL-Anweisungen programmiert, wobei pro Zeile nur eine AWL-Anweisung angegeben werden kann. Jede Anweisung stellt eine Arbeitsvorschrift für die CPU dar. Die CPU führt die Anweisungen von oben nach unten aus.

Das folgende Beispiel zeigt die AWL-Programmierung eines Netzwerks:

AWL	Erläuterung
`U "Tag_Input_1"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen`
`U "Tag_Input_2"`	`// Signalzustand des Operanden auf "1" abfragen und mit dem aktuellen VKE nach UND verknüpfen`
`S "Tag_Output"`	`// Operanden auf "1" setzen, wenn das VKE "1" ist`

15.1.2 Datenaustausch in AWL

Datenaustausch S7-300/400

Das Austauschen von Informationen zwischen verschiedenen Speicherbereichen erfolgt in AWL für CPUs der Baureihe S7-300/400 über Akkumulatoren. Die Akkumulatoren sind besondere Register im Prozessor, die als Zwischenspeicher dienen. Alle S7-300/400 CPUs besitzen zwei Zwischenspeicher, nämlich den Akkumulator 1 (AKKU 1) und den Akkumulator 2 (AKKU 2). Die S7-400 CPUs und die S7-318 CPU haben zusätzlich noch zwei weitere Zwischenspeicher, die Akkumulatoren 3 (AKKU 3) und 4 (AKKU 4).

Datenaustausch S7-1500

CPUs der Baureihe S7-1500 besitzen keine Register mehr im Prozessor, können aber die Adressregister, AKKU 1, AKKU 2, das Statuswort und die Datenbausteinregister DB und DI emulieren. Der Datenaustausch erfolgt primär über die Bausteinschnittstelle, globale Datenbausteine oder über PLC-Variablen. Werden dennoch die Akkumulatoren für den Datenaustausch verwendet, verlangsamt sich dadurch die Programmbearbeitung.

Beachten Sie bei der Programmierung in AWL für S7-1500 die folgenden grundsätzlichen Hinweise in Bezug auf die Register:

- Die Inhalte der Register, der Akkumulatoren und des Statusworts stehen nur innerhalb von AWL-Netzwerken zur Verfügung. Wenn ein KOP- oder FUP-Netzwerk auf ein AWL-Netzwerk folgt, können Sie aus dem KOP- oder FUP-Netzwerk heraus nicht auf die Registerinhalte zugreifen, die zuvor in AWL gesetzt wurden. In einem nachfolgenden AWL-Netzwerk stehen die Registerinhalte jedoch wieder zur Verfügung.
  Eine Ausnahme bildet das VKE-Bit: Es wird bei einem Sprachwechsel auf "undefiniert" gesetzt und steht in nachfolgenden Netzwerken nicht mehr zur Verfügung.
- Die Werte aus Registern, Akkumulatoren und dem Statuswort werden grundsätzlich nicht an aufgerufene Bausteine übergeben. Eine Ausnahme besteht nur bei den Anweisungen "CC" und "UC". Wenn Sie "UC" oder "CC" verwenden und dem aufgerufenen Baustein Parameter über Register, Statuswort oder Akkumulatoren übergeben möchten, müssen Sie in den Eigenschaften des aufgerufenen Bausteins die Option "Parameterversorgung über Register" aktivieren. Beachten Sie, dass diese Option nur für AWL-Bausteine mit Standardzugriff zur Verfügung steht und der Baustein keine Formalparameter haben darf. Wenn die Option aktiviert ist, können Sie Registerinhalte zwischen Bausteinen übergeben. Eine Ausnahme bildet auch hier das VKE-Bit: Es wird beim Verlassen des Bausteins auf "undefiniert" gesetzt und steht nach einem Bausteinaufruf nicht mehr zur Verfügung.
- Nach jedem Zugriff auf einen Datenbaustein mit Angabe einer vollqualifizierten Adresse (z. B. %DB10.DBW10) wird das Datenbausteinregister DB auf "0" gesetzt. Ein nachfolgender teilqualifizierter Zugriff führt beim Übersetzen zu einem Fehler.
- Um eine Fehleraussage an den aufrufenden Baustein zu übergeben, können Sie das BIE-Bit nutzen. Die Fehleraussage müssen Sie zunächst im aufgerufenen Baustein mit der Anweisung "SAVE" im BIE-Bit ablegen. Anschließend können Sie das BIE-Bit im aufrufenden Baustein lesen.
- Wenn Sie in S7-1500 einen lokalen Formalparameter aus der Bausteinschnittstelle symbolisch adressieren (z .B. mit der Anweisung L #myIn), greifen Sie immer auf den Datenbaustein zu, den Sie beim Bausteinaufruf als Instanz angegeben haben. Die Anweisungen AUF DI, L AR2, +AR2, TDB, TAR ändern zwar den Inhalt des DI- bzw. Adressregisters, die Register werden jedoch bei der Adressierung von lokalen Formalparametern nicht mehr ausgewertet.

Laden und Transferieren

Beim Datenaustausch wird die Richtung des Informationsflusses gekennzeichnet. Wenn Daten von einem Speicherbereich in den Akkumulator 1 übertragen werden, spricht man vom Laden der Daten. Die Datenübertragung vom Akkumulator 1 in einen Speicherbereich wird als "Transferieren" bezeichnet. Zum Laden und Transferieren werden in AWL Lade- und Transferanweisungen verwendet.

Das Laden und Transferieren ist vor allem für das Bearbeiten von Digitalwerten notwendig. Wenn Sie z. B. zwei Werte addieren möchten, müssen Sie diese in die Zwischenspeicher

laden und die entsprechende Addieranweisung ausführen. Zuerst wird der erste zu addierende Wert in Akkumulator 1 geladen. Mit dem Laden des zweiten Summanden wird der erste Wert, der sich im Akkumulator 1 befindet, in den Akkumulator 2 verschoben. Nach der Ausführung der Addieranweisung (z. B. I+) wird die Summe im Akkumulator 1 gespeichert. Mithilfe einer Transferanweisung kann die Summe in einen beliebigen Operanden übertragen werden.

Alle Akkumulatoren haben eine Länge von 32 Bits und sind byteweise organisiert. Die Datenübertragung zwischen den Speicherbereichen und dem Akkumulator 1 kann byte-, wort-, oder doppeltwortweise erfolgen.

Datenaustausch zwischen den Akkumulatoren

Zwischen den einzelnen Akkumulatoren können Daten ausgetauscht werden. Dazu dienen die Akkumulatoranweisungen "TAK", "PUSH" und "POP". Diese ermöglichen die Übertragung von Daten zwischen den verfügbaren Akkumulatoren und das Tauschen von Bytes im Akkumulator 1. Beachten Sie, dass Ihnen für die S7-1500 die Akkumulatoranweisungen "ENT" und "LEAVE" nicht mehr zur Verfügung stehen.

Siehe auch

Parameterversorgung über Register beim Bausteinaufruf in AWL (Seite 7559)

Laden (Seite 5000)

Transferieren (Seite 5008)

Akkumulator (Seite 5099)

15.1.3 Parameterversorgung über Register

15.1.3.1 Parameterversorgung über Register beim Bausteinaufruf in AWL

Bausteinaufrufe können in AWL über folgende Anweisungen realisiert werden:

- CALL: Baustein aufrufen
- UC: Baustein unbedingt aufrufen
- CC: Baustein bedingt aufrufen

Beim Bausteinaufruf werden üblicherweise Werte zwischen dem aufgerufenen und dem aufrufenden Baustein übergeben. Die Möglichkeiten des Datenaustauschs sind in den verschiedenen CPU-Familien unterschiedlich.
Siehe auch: Datenaustausch in AWL (Seite 7557)

Vom aufrufenden an den aufgerufenen Baustein übergebene Werte

Die folgende Tabelle zeigt, welche Register, Akkumulatoren und Statusbits direkt nach einem Bausteinaufruf im aufgerufenen Baustein, vor der Bearbeitung des ersten Netzwerks, zur Verfügung stehen:

	S7-300/400		S7-1500	
	UC / CC	CALL	UC / CC mit Option "Parameter-versorgung über Re-gister"	CALL
Statusbit OS	0	0	0	0
Statusbit OV	W	W	W	0
Statusbit A0	W	W	W	0
Statusbit A1	W	W	W	0
BIE	W	W	W	1
VKE	W	W	W	U
Akku 1	W	W	W	U
Akku 2	W	W	W	U
Adressregister 1	W	W	W	0
Adressregister 2	W	Call FC: W Call FB: Instanz-Offset	W	Call FC: 0 Call FB: Instanz-Offset
DB-Register	W	W	W	U
DI-Register	W	Call FC: W Call FB: Instanz-DB Nummer	W	Call FC: U Call FB: Instanz-DB Nummer

Legende:
0 = Register wird beim Bausteinwechsel auf "0" gesetzt.
1 = Register wird beim Bausteinwechsel auf "1" gesetzt.
W = Registerinhalt kann im aufrufenden Baustein gesetzt und im aufgerufenen Baustein gelesen werden.
U = Registerinhalt wird nicht an den aufgerufenen Baustein übergeben. Der Inhalt ist undefiniert, der Zugriff führt zu einem Fehler beim Übersetzen.
Instanz-Offset = Offset im Instanz-DB
Instanz-DB Nummer = Nummer des Instanz-DB

Vom aufgerufenen an den aufrufenden Baustein zurückgegebene Werte

Die folgende Tabelle zeigt, welche Register, Akkumulatoren und Statusbits im aufrufenden Baustein direkt nach der Bearbeitung des Bausteinaufrufs zur Verfügung stehen:

	S7-300/400		S7-1500	
	UC / CC	CALL	UC / CC mit Option "Parameter-versorgung über Re-gister"	CALL
Statusbit OS	0	0	0	0
Statusbit OV	W	W	W	-
Statusbit A0	W	W	W	-
Statusbit A1	W	W	W	-
BIE	W	W	W	W
VKE	W	W1	W	U

	S7-300/400		S7-1500	
Akku 1	W	W1	W	-
Akku 2	W	W1	W	-
Adressregister 1	W	W1	W	-
Adressregister 2	W	Call FC: W Call FB: -	W	-
DB-Register	-	X	-	-
DI-Register	-	X	-	-

Legende:
0 = Register wird beim Bausteinwechsel auf "0" gesetzt.
1 = Register wird beim Bausteinwechsel auf "1" gesetzt.
W = Registerinhalt kann im aufgerufenen Baustein gesetzt und im aufrufenden Baustein gelesen werden.
W1 = Der zurückgegebene Wert kann durch die Ausgangs- und Durchgangsparameter des aufgerufenen Bausteins verändert werden.
- = Registerinhalt wird nicht an den aufrufenden Baustein zurückgegeben, der Inhalt wird durch den Bausteinaufruf nicht verändert.
U = Registerinhalt wird nicht an den aufgerufenen Baustein übergeben. Der Inhalt ist undefiniert, der Zugriff führt zu einem Fehler beim Übersetzen.
X = Registerinhalt wird nicht an den aufrufenden Baustein zurückgegeben. Der Wert kann jedoch implizit durch die Parameter des aufgerufenen Bausteins verändert werden.

Siehe auch

Parameterversorgung über Register aktivieren (Seite 7561)

15.1.3.2 Parameterversorgung über Register aktivieren

Voraussetzung

- Die Option "Optimierter Bausteinzugriff" ist deaktiviert.
- Die Bausteinschnittstelle enthält keine Parameter.

Vorgehen

Um die Parameterversorgung über Register zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Baustein, für den Sie die Parameterversorgung über Register ermöglichen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
4. Der Eigenschaftsdialog des Bausteins wird geöffnet.

5. Klicken Sie in der Bereichsnavigation auf "Attribute".
6. Aktivieren Sie die Option "Parameterversorgung über Register".
7. Bestätigen Sie Ihre Eingaben mit "OK".

Siehe auch

Parameterversorgung über Register beim Bausteinaufruf in AWL (Seite 7559)
Datenaustausch in AWL (Seite 7557)

15.2 Einstellungen zu AWL

15.2.1 Übersicht über die Einstellungen zu AWL

Übersicht

Die folgende Tabelle zeigt die Einstellungen, die Sie für AWL vornehmen können:

Gruppe	Einstellung	Beschreibung
Ansicht	Zeilenumbruch bei Return	Fügt einen Zeilenumbruch ein, wenn Sie die Eingabetaste drücken.

Siehe auch

Einstellungen ändern (Seite 7563)

15.2.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die Einstellungen zu AWL (Seite 7563)

15.3 Arbeiten mit Netzwerken

15.3.1 Netzwerke verwenden

Funktion

Das Anwenderprogramm wird im Baustein innerhalb von Netzwerken erstellt. Damit ein Codebaustein programmiert werden kann, muss er mindestens ein Netzwerk enthalten. Um eine bessere Übersichtlichkeit des Anwenderprogramms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

Siehe auch

Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke kopieren und einfügen (Seite 7566)

Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerktitel einfügen (Seite 7568)

Netzwerkkommentar eingeben (Seite 7569)

In Netzwerken navigieren (Seite 7570)

15.3.2 Netzwerke einfügen

Voraussetzung

Ein Baustein ist geöffnet.

Vorgehen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

Ergebnis

Ein neues, leeres Netzwerk wird in den Baustein eingefügt.

Siehe auch

Netzwerke verwenden (Seite 7564)
Netzwerke selektieren (Seite 7565)
Netzwerke kopieren und einfügen (Seite 7566)
Netzwerke löschen (Seite 7566)
Netzwerke reduzieren und erweitern (Seite 7567)
Netzwerktitel einfügen (Seite 7568)
Netzwerkkommentar eingeben (Seite 7569)
In Netzwerken navigieren (Seite 7570)

15.3.3 Netzwerke selektieren

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.
   Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

Siehe auch

Netzwerke verwenden (Seite 7564)
Netzwerke einfügen (Seite 7564)
Netzwerke kopieren und einfügen (Seite 7566)
Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerktitel einfügen (Seite 7568)

Netzwerkkommentar eingeben (Seite 7569)

In Netzwerken navigieren (Seite 7570)

15.3.4 Netzwerke kopieren und einfügen

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

Netzwerke verwenden (Seite 7564)

Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerktitel einfügen (Seite 7568)

Netzwerkkommentar eingeben (Seite 7569)

In Netzwerken navigieren (Seite 7570)

15.3.5 Netzwerke löschen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Netzwerke verwenden (Seite 7564)
Netzwerke einfügen (Seite 7564)
Netzwerke selektieren (Seite 7565)
Netzwerke kopieren und einfügen (Seite 7566)
Netzwerke reduzieren und erweitern (Seite 7567)
Netzwerktitel einfügen (Seite 7568)
Netzwerkkommentar eingeben (Seite 7569)
In Netzwerken navigieren (Seite 7570)

15.3.6 Netzwerke reduzieren und erweitern

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

Alle Netzwerke auf- und zuklappen

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

Siehe auch

Netzwerke verwenden (Seite 7564)
Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke kopieren und einfügen (Seite 7566)

Netzwerke löschen (Seite 7566)

Netzwerktitel einfügen (Seite 7568)

Netzwerkkommentar eingeben (Seite 7569)

In Netzwerken navigieren (Seite 7570)

15.3.7 Netzwerktitel einfügen

Der Netzwerktitel ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitels ist auf eine Zeile beschränkt. Sie können den Titel entweder manuell eingeben oder automatisch setzen. Beim automatischen Setzen können Sie dies für einzelne Netzwerke vornehmen oder über die Einstellungen festlegen, dass der Netzwerktitel grundsätzlich automatisch gesetzt werden soll.

Für das automatische Einfügen des Netzwerktitels wird der Kommentar des Operanden einer der folgenden Anweisungen im Netzwerk ausgewertet:

- Zuweisung
- Ausgang setzen
- Ausgang rücksetzen

Dabei wird die Anweisung verwendet, die im Netzwerk zuerst steht.

Der Netzwerktitel wird nur dann automatisch eingefügt, wenn die folgenden Bedingungen erfüllt sind:

- Das Netzwerk besitzt noch keinen Titel.
- Der Operand der Anweisung, die für den Kommentar verwendet wird, besitzt einen Kommentar.

Hinweis

Beachten Sie die folgenden Einschränkungen beim automatischen Einfügen des Netzwerktitels:

- Wenn Sie den Kommentar des Operanden nachträglich ändern, wird der Netzwerktitel nicht angepasst.
- Wenn Sie den Operanden der Anweisung wechseln, wird der Netzwerktitel nicht angepasst.
- Der Netzwerktitel wird nur durch die oben genannten schreibenden Anweisungen gesetzt.
- Ist der Operand vom Datentyp Array, wird der Kommentar des Arrays verwendet und nicht die Kommentare der Array-Elemente.
- Kommentare ungültiger Operanden werden nicht berücksichtigt.

Netzwerktitel manuell eingeben

Um einen Netzwerktitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelzeile des Netzwerks.
2. Geben Sie den Netzwerktitel ein.

Netzwerktitel automatisch setzen

Um festzulegen, dass die Netzwerktitel grundsätzlich automatisch gesetzt werden sollen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Aktivieren Sie in der Gruppe "Weitere Einstellungen" das Optionskästchen "Netzwerktitel automatisch setzen".
   Die Titel der Netzwerke werden ab diesem Zeitpunkt automatisch gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Um einen einzelnen Netzwerktitel automatisch zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste eines Netzwerks mit der rechten Maustaste auf "Netzwerk <Nummer des Netzwerks>".
2. Wählen Sie im Kontextmenü den Befehl "Netzwerktitel automatisch setzen".
   Der Titel des selektierten Netzwerks wird anhand des Kommentars des Operanden gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Siehe auch

Netzwerke verwenden (Seite 7564)

Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke kopieren und einfügen (Seite 7566)

Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerkkommentar eingeben (Seite 7569)

In Netzwerken navigieren (Seite 7570)

15.3.8 Netzwerkkommentar eingeben

Mithilfe von Netzwerkkommentaren können Sie die Programminhalte der einzelnen Netzwerke erläutern. Sie können z. B. die Funktion des Netzwerks beschreiben oder auf Besonderheiten hinweisen.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um einen Netzwerkkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel.
2. Wenn der Kommentarbereich nicht sichtbar ist, klicken Sie in der Funktionsleiste auf die Schaltfläche "Netzwerkkommentare ein/aus".
   Der Kommentarbereich wird angezeigt.
3. Klicken Sie im Kommentarbereich auf "Kommentar".
   Die Textstelle "Kommentar" wird markiert.
4. Geben Sie den Netzwerkkommentar ein.

Siehe auch

Netzwerke verwenden (Seite 7564)

Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke kopieren und einfügen (Seite 7566)

Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerktitel einfügen (Seite 7568)

In Netzwerken navigieren (Seite 7570)

15.3.9 In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".
   Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

Siehe auch

Netzwerke verwenden (Seite 7564)

Netzwerke einfügen (Seite 7564)

Netzwerke selektieren (Seite 7565)

Netzwerke kopieren und einfügen (Seite 7566)

Netzwerke löschen (Seite 7566)

Netzwerke reduzieren und erweitern (Seite 7567)

Netzwerktitel einfügen (Seite 7568)

Netzwerkkommentar eingeben (Seite 7569)

15.4 AWL-Anweisungen einfügen

15.4.1 Regeln zu AWL

Regeln

Bei der inkrementellen Eingabe von AWL-Anweisungen müssen Sie folgende Regeln beachten:

- Eine Anweisung besteht aus der Angabe einer optionalen Sprungmarke, der Operation, dem Operanden und einem optionalen Kommentar. Beispiel:
  `M001: U E1.0 //Kommentar`
- Jede Anweisung steht in einer eigenen Zeile.
- Sie können je Baustein maximal 999 Netzwerke eingeben.
- Bei der Eingabe von Operationen oder absoluten Adressen wird nicht zwischen Groß- und Kleinschreibung unterschieden.
- Wenn Sie bei Bausteinaufrufen Unterstützung möchten, müssen Sie den aufgerufenen Baustein vor dem aufrufenden Baustein programmieren.

Unzulässige Anweisungen werden rot gekennzeichnet. Ein Rollout und das Inspektorfenster geben Hinweise auf die Fehlerbehebung.

Sprünge

Achten Sie darauf, dass bei Sprungoperationen das Sprungziel immer der Beginn einer Verknüpfungskette ist und sich innerhalb einer anderen Klammerebene befindet (gilt nicht für die CPU 318-2). Das Sprungziel darf sich nicht innerhalb einer Verknüpfungskette oder einer Klammerebene befinden.

Siehe auch

AWL-Anweisungen manuell eingeben (Seite 7575)

AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)

Kommentare einfügen (Seite 7590)

Operanden einfügen (Seite 7579)

Favoriten in AWL verwenden (Seite 7581)

Bausteinaufrufe in AWL einfügen (Seite 7583)

15.4.2 Anweisungen der globalen Bibliothek "Long Functions"

Beschreibung

Die folgende Tabelle zeigt Ihnen, welche Anweisungen in der globalen Bibliothek "Long Functions" enthalten sind:

Anweisung	Erklärung
ABS_LINT	Absolutwert bilden mit dem Datentyp LINT
ABS_LREAL	Absolutwert bilden mit dem Datentyp LREAL
ACOS_LREAL	Arcuscosinuswert bilden mit dem Datentyp LREAL
ADD_LINT	Addieren mit dem Datentyp LINT
ADD_LREAL	Addieren mit dem Datentyp LREAL
ADD_ULINT	Addieren mit dem Datentyp ULINT
AND_LWORD	Und verknüpfen mit dem Datentyp LWORD
ASIN_LREAL	Arcussinuswert bilden mit dem Datentyp LREAL
ATAN_LREAL	Arcustangenswert bilden mit dem Datentyp LREAL
COS_LREAL	Cosinuswert bilden mit dem Datentyp LREAL
DINT_to_LINT	Datentypkonvertierung von DINT nach LINT
DIV_LINT	Dividieren mit dem Datentyp LINT
DIV_LREAL	Dividieren mit dem Datentyp LREAL
DIV_ULINT	Dividieren mit dem Datentyp ULINT
EQ_LINT	Auf Gleich vergleichen mit dem Datentyp LINT
EQ_LREAL	Auf Gleich vergleichen mit dem Datentyp LREAL
EQ_LWORD	Auf Gleich vergleichen mit dem Datentyp LWORD
EQ_ULINT	Auf Gleich vergleichen mit dem Datentyp ULINT
EXP_LREAL	Exponentialwert bilden mit dem Datentyp LREAL
GE_LINT	Auf Größer gleich vergleichen mit dem Datentyp LINT
GE_LREAL	Auf Größer gleich vergleichen mit dem Datentyp LREAL
GE_ULINT	Auf Größer gleich vergleichen mit dem Datentyp ULINT
GT_LINT	Auf Größer als vergleichen mit dem Datentyp LINT
GT_LREAL	Auf Größer als vergleichen mit dem Datentyp LREAL
GT_ULINT	Auf Größer als vergleichen mit dem Datentyp ULINT
LE_LINT	Auf Kleiner gleich vergleichen mit dem Datentyp LINT
LE_LREAL	Auf Kleiner gleich vergleichen mit dem Datentyp LREAL
LE_ULINT	Auf Kleiner gleich vergleichen mit dem Datentyp ULINT
LINT_to_DINT	Datentypkonvertierung von LINT nach DINT
LINT_to_LREAL	Datentypkonvertierung von LINT nach LREAL

Anweisung	Erklärung
LINT_to_ULINT	Datentypkonvertierung von LINT nach ULINT
LN_LREAL	Natürlichen Logarithmus bilden mit dem Datentyp LREAL
LREAL_to_LINT	Datentypkonvertierung von LREAL nach LINT
LREAL_to_REAL	Datentypkonvertierung von LREAL nach REAL
LREAL_to_ULINT	Datentypkonvertierung von LREAL nach ULINT
LT_LINT	Auf Kleiner als vergleichen mit dem Datentyp LINT
LT_LREAL	Auf Kleiner als vergleichen mit dem Datentyp LREAL
LT_ULINT	Auf Kleiner als vergleichen mit dem Datentyp ULINT
LWORD_to_ULINT	Datentypkonvertierung von LWORD nach ULINT
LWORD_to_WORD	Datentypkonvertierung von LWORD nach WORD
MOVE_LINT	Wert kopieren mit dem Datentyp LINT
MOVE_LREAL	Wert kopieren mit dem Datentyp LREAL
MOVE_LWORD	Wert kopieren mit dem Datentyp LWORD
MOVE_ULINT	Wert kopieren mit dem Datentyp ULINT
MUL_LINT	Multiplizieren mit dem Datentyp LINT
MUL_LREAL	Multiplizieren mit dem Datentyp LREAL
MUL_ULINT	Multiplizieren mit dem Datentyp ULINT
NE_LINT	Auf Ungleich vergleichen mit dem Datentyp LINT
NE_LREAL	Auf Ungleich vergleichen mit dem Datentyp LREAL
NE_LWORD	Auf Ungleich vergleichen mit dem Datentyp LWORD
NE_ULINT	Auf Ungleich vergleichen mit dem Datentyp ULINT
NEG_LINT	Zweierkomplement erzeugen mit dem Datentyp LINT
NEG_LREAL	Zweierkomplement erzeugen mit dem Datentyp LREAL
OR_LWORD	ODER verknüpfen mit dem Datentyp LWORD
RDN_LREAL	Zahl runden mit dem Datentyp LREAL
REAL_to_LREAL	Datentypkonvertierung von REAL nach LREAL
SIN_LREAL	Sinuswert bilden mit dem Datentyp LREAL
SQR_LREAL	Quadrat bilden mit dem Datentyp LREAL
SQRT_LREAL	Quadratwurzel bilden mit dem Datentyp LREAL
SUB_LINT	Subtrahieren mit dem Datentyp LINT
SUB_LREAL	Subtrahieren mit dem Datentyp LREAL
SUB_ULINT	Subtrahieren mit dem Datentyp ULINT
TAN_LREAL	Tangenswert bilden mit dem Datentyp LREAL
TRUNC_LREAL	Ganzzahl erzeugen mit dem Datentyp LREAL
UDINT_to_ULINT	Datentypkonvertierung von UDINT nach ULINT
ULINT_to_LINT	Datentypkonvertierung von ULINT nach LINT
ULINT_to_LREAL	Datentypkonvertierung von ULINT nach LREAL
ULINT_to_LWORD	Datentypkonvertierung von ULINT nach LWORD

Anweisung	Erklärung
ULINT_to_UDINT	Datentypkonvertierung von ULINT nach UDINT
WORD_to_LWORD	Datentypkonvertierung von WORD nach LWORD
XOR_LWORD	EXCLUSIV ODER mit dem Datentyp LWORD

Siehe auch

AWL-Anweisungen manuell eingeben (Seite 7575)

AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)

Operanden einfügen (Seite 7579)

15.4.3 AWL-Anweisungen manuell eingeben

Beim Erstellen eines leeren Netzwerks, automatisch oder manuell, wird ein zweizeiliges Eingabefeld innerhalb des Netzwerks erzeugt. In dieses Eingabefeld werden die AWL-Anweisungen eingegeben.

Hinweis

S7-1500: Anweisungen aus der Task Card "Anweisungen", die emulierte Register und Akkumulatoren verwenden, können als Operanden keine Variablen mit 64 Bit Länge verwenden, z. B. LInt. Wenn Sie 64-Bit-Variablen einsetzen möchten, verwenden Sie die Anweisungen aus der globalen Bibliothek "Long Functions".

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um AWL-Anweisungen in ein Netzwerk einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in einem Netzwerk in die erste leere Zeile des Eingabefelds.
2. Geben Sie eine Anweisung ein.
3. Geben Sie ein Leerzeichen.
4. Geben Sie den Operanden ein.

   Hinweis

   Sie können einen bereits definierten Operanden auch per Drag & Drop aus der PLC-Variablentabelle oder der Detailansicht einfügen.

5. Geben Sie einen Zeilenkommentar ein. Dieser Schritt ist optional.

6. Bestätigen Sie die Anweisung mit der Eingabetaste.
7. Wiederholen Sie die Schritte 2 bis 6 so lange, bis Sie alle benötigten Anweisungen eingegeben haben.

Ergebnis

Nach dem Abschluss der Anweisung mit der Eingabetaste führt der Programmiereditor folgende Aktionen durch:

- Syntaxprüfung: Fehlerhafte Eingaben werden nach der Prüfung rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Register "Syntax" des Info-Fensters.
- Änderung von Klein- und Großschreibung: Anweisungen werden in Großbuchstaben geändert.
- Wenn Sie bei symbolischer Programmierung absolute Adressen eingegeben haben, werden diese in symbolische Namen geändert. Falls kein symbolischer Name vorhanden ist, wird "Tag_< Nummer>" eingefügt.
- Bausteinaufruf: Notwendige Parameter werden eingefügt und angezeigt.
- In das Eingabefeld wird eine weitere Zeile eingefügt.

Siehe auch

Anweisungen der globalen Bibliothek "Long Functions" (Seite 7573)

Regeln zu AWL (Seite 7572)

AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)

Operanden einfügen (Seite 7579)

Favoriten in AWL verwenden (Seite 7581)

Bausteinaufrufe in AWL einfügen (Seite 7583)

Kommentare einfügen (Seite 7590)

15.4.4 AWL-Anweisungen über Task Card "Anweisungen" einfügen

Beim Erstellen eines leeren Netzwerks, automatisch oder manuell, wird ein zweizeiliges Eingabefeld innerhalb des Netzwerks erzeugt. In dieses Eingabefeld werden die AWL-Anweisungen eingegeben.

Hinweis

S7-1500: Anweisungen aus der Task Card "Anweisungen", die emulierte Register und Akkumulatoren verwenden, können als Operanden keine Variablen mit 64 Bit Länge verwenden, z. B. LInt. Wenn Sie 64-Bit-Variablen einsetzen möchten, verwenden Sie die Anweisungen aus der globalen Bibliothek "Long Functions".

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um AWL-Anweisungen mithilfe der Task Card "Anweisungen" in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Zum Einfügen der Anweisung wählen Sie einen der folgenden Schritte:
   - Navigieren Sie zu dem AWL-Element, das Sie einfügen möchten, und ziehen Sie es per Drag & Drop an die gewünschte Stelle im Netzwerk.
   - Markieren Sie die Stelle im Netzwerk, an die Sie das Element einfügen möchten, und doppelklicken Sie dann auf das einzufügende Element.
3. Geben Sie die notwendigen Operanden ein.

   Hinweis

   Sie können einen bereits definierten Operanden auch per Drag & Drop aus der PLC-Variablentabelle, einem Datenbaustein, der Detailansicht oder der Bausteinschnittstelle einfügen.

4. Geben Sie einen Zeilenkommentar ein. Dieser Schritt ist optional.
5. Wiederholen Sie die Schritte 2 bis 4 so lange, bis Sie alle benötigten Anweisungen eingegeben haben.

Ergebnis

Nach dem Abschluss der Anweisung mit der Eingabetaste führt der Programmiereditor folgende Aktionen durch:

- Syntaxprüfung: Fehlerhafte Eingaben werden nach der Prüfung rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Info-Fenster.
- Änderung von Klein- und Großschreibung: Anweisungen werden in Großbuchstaben geändert.
- Wenn Sie bei symbolischer Programmierung absolute Adressen eingegeben haben, werden diese in symbolische Namen geändert. Falls kein symbolischer Name vorhanden ist, wird "Tag_< Nummer>" eingefügt.
- Bausteinaufruf: Notwendige Parameter werden eingefügt und angezeigt.
- In das Eingabefeld wird eine weitere Zeile eingefügt.

Siehe auch

Anweisungen der globalen Bibliothek "Long Functions" (Seite 7573)

Regeln zu AWL (Seite 7572)

AWL-Anweisungen manuell eingeben (Seite 7575)

Operanden einfügen (Seite 7579)

Favoriten in AWL verwenden (Seite 7581)

Bausteinaufrufe in AWL einfügen (Seite 7583)

Kommentare einfügen (Seite 7590)

15.4.5 Datentyp einer AWL-Anweisung auswählen

15.4.5.1 Auswahl eines Datentyps

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie eine dieser Anweisungen im Programm verwenden, müssen Sie an der konkreten Programmstelle einen für die Anweisung zulässiger Datentyp festlegen.

Die für eine Anweisung zulässigen Datentypen sind in der Klappliste der Anweisung aufgeführt. Durch die Auswahl eines Eintrags aus der Klappliste bestimmen Sie den Datentyp der Anweisung. Wenn der Datentyp eines Operanden vom Datentyp der Anweisung abweicht und nicht implizit konvertiert werden kann, wird der Operand rot angezeigt und ein Rollout mit der entsprechenden Fehlermeldung eingeblendet.

15.4.5.2 Datentyp einer Anweisung festlegen

Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie solche Anweisungen in Ihr Programm einfügen, müssen Sie für diese Anweisungen an der konkreten Programmstelle den Datentyp festlegen.

Vorgehen

Um den Datentyp einer Anweisung über die Klappliste festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
   In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Klicken Sie auf das Dreieck in der oberen Ecke der Klappliste.
   Die Klappliste wird geöffnet und die Datentypen angezeigt, die für die Anweisung zulässig sind.
3. Wählen Sie aus der Klappliste einen Datentyp aus.
   Der ausgewählte Datentyp wird angezeigt.

15.4.6 Operanden einfügen

Sie können die Operanden einer AWL-Anweisung entweder manuell eingeben oder per Drag & Drop aus der PLC-Variablentabelle, einem Datenbaustein, der Detailansicht oder der Bausteinschnittstelle übernehmen.

Hinweis

S7-1500: Anweisungen aus der Task Card "Anweisungen", die emulierte Register und Akkumulatoren verwenden, können als Operanden keine Variablen mit 64 Bit Länge verwenden, z. B. LInt. Wenn Sie 64-Bit-Variablen einsetzen möchten, verwenden Sie die Anweisungen aus der globalen Bibliothek "Long Functions".

Vorgehen

Um den Operanden einer AWL-Anweisung manuell einzugeben, gehen Sie folgendermaßen vor:

1. Setzen Sie den Cursor rechts neben die Operation.
2. Geben Sie ein Leerzeichen und anschließend den Operanden ein.

   Hinweis

   Wenn Sie bei symbolischer Programmierung die absolute Adresse eines Parameters eingeben, der bereits definiert ist, wird diese Absolutadresse in den symbolischen Namen des Parameters geändert, sobald Sie die Eingabe bestätigen. Haben Sie den Parameter noch nicht definiert, wird in der PLC-Variablentabelle eine neue Variable mit dieser Absolutadresse und dem Standardnamen "Tag_1" eingetragen. Bestätigen Sie Ihre Eingabe, wird die Absolutadresse gegen den symbolischen Namen "Tag_1" ausgetauscht.

3. Bestätigen Sie den Parameter mit der Eingabetaste.
4. Wenn Sie den Parameter bisher nicht definiert haben, können Sie dies über das Kontextmenü direkt im Programmiereditor erledigen.
   Siehe auch: "Variablen direkt im Programmiereditor deklarieren (Seite 7383)"

Oder per Drag & Drop aus der PLC-Variablentabelle:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" oder öffnen Sie die PLC-Variablentabelle.
2. Wenn Sie die PLC-Variablentabelle geöffnet haben, ziehen Sie das Symbol aus der ersten Spalte der gewünschten Variablen an die entsprechende Stelle in Ihrem Programm. Falls Sie die PLC-Variablentabelle nicht geöffnet haben, öffnen Sie die Detailansicht. Ziehen Sie die gewünschte Variable aus der Detailansicht an die entsprechende Stelle in Ihrem Programm.

Oder per Drag & Drop aus der Bausteinschnittstelle:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ziehen Sie den gewünschten Operanden aus der Bausteinschnittstelle in das Anweisungsfenster.

Siehe auch

Anweisungen der globalen Bibliothek "Long Functions" (Seite 7573)

Regeln zu AWL (Seite 7572)

AWL-Anweisungen manuell eingeben (Seite 7575)

AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)

Favoriten in AWL verwenden (Seite 7581)

Bausteinaufrufe in AWL einfügen (Seite 7583)

Kommentare einfügen (Seite 7590)

Operanden verwenden und adressieren (Seite 93)

Autovervollständigung verwenden (Seite 7147)

15.4.7 Variableninformationen ein- oder ausblenden

Einführung

Unabhängig von der absoluten oder symbolischen Darstellung der Operanden können Sie einfache oder hierarchische Kommentare zur Dokumentation der Variablen ein- bzw. ausblenden. Diese Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablentabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden, werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet.

Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete Bausteine wieder einblenden.

Wenn Sie die Anzeige von Variableninformationen mit hierarchischen Kommentaren wählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt. Die Anzeige erfolgt dabei in Klammern hinter dem Kommentar der Variablen, die Kommentare der einzelnen Ebenen werden jeweils durch einen Punkt getrennt. Falls auf einer Strukturebene für eine Variable kein Kommentar vorhanden ist, entfällt er in der Anzeige, was Sie an zwei aufeinander folgenden Punkten erkennen können.

Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".

3. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Variableninformationen" entweder die Option "Einblenden" oder die Option "Variableninformationen mit Hierarchie", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
4. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Variableninformationenen" die Option "Ausblenden".
   Die Variableninformationen werden für alle geöffneten Bausteine ein- oder ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen abhängig von der gewählten Einstellung ein- oder ausgeblendet.

Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." entweder die Option "Variableninformationen einblenden" oder die Option "Variableninformationen mit Hierarchie", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
2. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." die Option "Variableninformationen ausblenden".
   Die Variableninformationen werden ein- oder ausgeblendet.

15.4.8 Favoriten in AWL verwenden

15.4.8.1 AWL-Elemente zu den Favoriten hinzufügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

AWL-Anweisungen mithilfe von Favoriten einfügen (Seite 7582)

AWL-Anweisungen aus den Favoriten entfernen (Seite 7583)

Übersicht über den Programmiereditor (Seite 7117)

15.4.8.2 AWL-Anweisungen mithilfe von Favoriten einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

AWL-Elemente zu den Favoriten hinzufügen (Seite 7581)

AWL-Anweisungen manuell eingeben (Seite 7575)

AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)

AWL-Anweisungen aus den Favoriten entfernen (Seite 7583)

15.4.8.3 AWL-Anweisungen aus den Favoriten entfernen

Voraussetzung

Ein Codebaustein ist geöffnet.

Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

AWL-Elemente zu den Favoriten hinzufügen (Seite 7581)

AWL-Anweisungen mithilfe von Favoriten einfügen (Seite 7582)

Übersicht über den Programmiereditor (Seite 7117)

15.4.9 Bausteinaufrufe in AWL einfügen

15.4.9.1 Bausteinaufrufe manuell einfügen

Sie können Aufrufe für Funktionen (FCs) und Funktionsbausteine (FBs) einfügen. Funktionsbausteine können Sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Voraussetzung

Ein Netzwerk ist vorhanden.

Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf in ein AWL-Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "CALL <Name der aufzurufenden Funktion>" ein.
   Ersetzen Sie dabei "<Name der aufzurufenden Funktion>" durch den Namen der Funktion, die Sie aufrufen möchten.
2. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.
   - Wenn die Funktion bereits existiert, werden alle Parameter der Funktion in den Aufruf eingefügt.
   - Wenn die Funktion noch nicht existiert, müssen Sie die Parameter selbst eingeben.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "CALL <Name des aufzurufenden Funktionsbausteins>[, <Name des Instanz-Datenbausteins>]" ein.
   Ersetzen Sie dabei "<Name des aufzurufenden Funktionsbausteins>" mit dem Namen des Funktionsbausteins, den Sie aufrufen möchten. Die Eingabe des Instanz-Datenbausteins ist optional.
2. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.
   Wenn Sie den Instanz-Datenbaustein nicht angegeben haben oder der eingegebene Instanz-Datenbaustein nicht vorhanden ist, wird der Dialog "Aufrufoptionen" geöffnet. Führen Sie in diesem Fall die Schritte 3 und 4 aus.
3. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Namen in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Beim Aufruf bestehender Funktionen und bestehender Funktionsbausteine mit Instanz-Datenbaustein werden die Parameter der Funktion bzw. des Funktionsbausteins in den Aufruf eingefügt. Diese Parameter müssen Sie anschließend versorgen.

Siehe auch: Parameterübergabe beim Bausteinaufruf (Seite 78)

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe per Drag & Drop einfügen (Seite 7585)

Bausteinaufrufe in AWL aktualisieren (Seite 7586)

Instanztyp ändern (Seite 7588)

Parameterversorgung über Register beim Bausteinaufruf in AWL (Seite 7559)

15.4.9.2 Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FCs) und Funktionsbausteine (FBs) per Drag & Drop aus der Projektnavigation einfügen. Funktionsbausteine können Sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Voraussetzung

- Ein Netzwerk ist vorhanden.
- Die aufzurufende Funktion (FC) oder der aufzurufende Funktionsbaustein (FB) ist vorhanden.

Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf per Drag & Drop in ein AWL-Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das Netzwerk.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das Netzwerk.
   Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Instanz-Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
3. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter müssen Sie anschließend versorgen.

Siehe auch: Parameterübergabe beim Bausteinaufruf (Seite 78)

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7583)

Bausteinaufrufe in AWL aktualisieren (Seite 7586)

Instanztyp ändern (Seite 7588)

15.4.9.3 Bausteinaufrufe in AWL aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben die folgenden Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren aller inkonsistenten Bausteinaufrufe im Programmiereditor.
  Die inkonsistenten Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Explizites Aktualisieren eines Bausteinaufrufs im Programmiereditor.
  Der Dialog "Schnittstellenabgleich" wird angezeigt. In diesem Dialog haben Sie die Möglichkeit, die Beschaltung der Operanden der neuen Schnittstelle zu ändern. Anschließend wird der inkonsistente Aufruf dieses Bausteins aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden entfernt, wenn sie nicht beschaltet sind.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert. Beachten Sie, dass Sie beim Aufruf von Funktionen (FCs) vor dem nächsten Übersetzungsvorgang alle neuen Formalparameter mit Aktualparametern versorgen müssen.

Alle inkonsistenten Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein alle Bausteinaufrufe zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Einen bestimmten Bausteinaufruf im Programmiereditor aktualisieren

Um einen bestimmten Bausteinaufruf im Programmiereditor zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf, den Sie aktualisieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Bausteinaufruf aktualisieren".
   Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden Ihnen die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.

4. Ändern Sie bei Bedarf die Beschaltung der Operanden. Dazu haben Sie die folgenden Möglichkeiten:
   - Sie können einen Operanden entweder per Drag & Drop oder über Ausschneiden/ Kopieren und Einfügen aus der alten Schnittstelle in die neue Schnittstelle übernehmen.
   - Sie können einen Operanden löschen.
   - Sie können einen Operanden umbenennen.
   - Sie können über die Autovervollständigung einen neuen Operanden angeben.
5. Klicken Sie auf "OK", um den Bausteinaufruf zu aktualisieren. Wenn Sie die Aktualisierung abbrechen möchten, klicken Sie auf "Abbrechen".

Hinweis

Beachten Sie, dass Ihnen der Befehl "Bausteinaufruf aktualisieren" nur zur Verfügung steht, wenn Sie nicht zuvor alle Bausteinaufrufe im Editor über den Befehl "Inkonsistente Bausteinaufrufe aktualisieren" aktualisiert haben.

Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7583)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7585)

Instanztyp ändern (Seite 7588)

Bausteinaufrufe (Seite 62)

15.4.9.4 Instanztyp ändern

Instanztyp

Funktionsbausteine können Sie auf verschiedene Arten aufrufen:

- Als Einzelinstanz
- Als Multiinstanz
- Als Parameterinstanz

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Sie können einen einmal definierten Instanztyp jederzeit ändern.

Voraussetzung

Das Anwenderprogramm enthält einen Bausteinaufruf.

Vorgehen

Um den Instanztyp eines Funktionsbausteins zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Codebaustein und selektieren Sie den Bausteinaufruf.
2. Wählen Sie im Kontextmenü den Befehl "Instanz ändern".
   Der Dialog "Aufrufoptionen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "Einzel-Instanz", "Multi-Instanz" oder "Parameter-Instanz".
   - Wenn Sie als Instanztyp "Einzel-Instanz" wählen, geben Sie einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
   - Wenn Sie als Instanztyp "Multi-Instanz" wählen, geben Sie im Textfeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
   - Wenn Sie als Instanztyp "Parameter-Instanz" wählen, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Hinweis

Die vorherigen Instanzen werden nicht automatisch gelöscht.

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7583)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7585)

Bausteinaufrufe in AWL aktualisieren (Seite 7586)

Instanzen (Seite 64)

15.4.10 Kommentare einfügen

Programmcode kommentieren

Sie haben verschiedene Möglichkeiten, um AWL-Programme zu kommentieren:

- Zeilenkommentar
  Ein Zeilenkommentar wird mit "//" eingeleitet und erstreckt sich bis zum Ende der Zeile.
- Kommentarabschnitt innerhalb einer Zeile
  Ein Kommentarabschnitt wird mit "(*" eingeleitet und durch "*)" abgeschlossen. Ein Kommentarabschnitt kann an jeder Stelle in der Zeile stehen, aber er kann sich nicht über mehrere Zeilen erstrecken.

Voraussetzung

Ein Netzwerk ist vorhanden.

Zeilenkommentare einfügen

Um in eine Anweisungszeile einen Zeilenkommentar einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie am Ende der Anweisungszeile "//" ein.
2. Geben Sie den Kommentar ein.

Kommentarabschnitt innerhalb einer Zeile einfügen

Um innerhalb einer Anweisungszeile einen Kommentarabschnitt einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie an einer beliebigen Stelle in der Anweisungszeile "(*" ein.
2. Geben Sie hinter "(*" den Kommentar ein.
3. Geben Sie hinter dem Kommentar "*)" ein.

Eine oder mehrere Zeilen durch Kommentieren deaktivieren

Um Programmcode durch Kommentieren zu deaktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie auskommentieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Markierung auskommentieren".
3. In die markierten Zeilen wird am Zeilenbeginn "//" eingefügt. Der nachfolgende Code wird als Kommentar interpretiert. Werden Zeilen deaktiviert, die bereits einen Zeilenkommentar enthalten, wird ebenfalls "//" eingefügt. Werden die Zeilen anschließend wieder aktiviert, bleiben ursprüngliche Kommentare erhalten.

Kommentarzeilen aktivieren

Um auskommentierte Zeilen wieder als Code zu aktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie aktivieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Kommentar entfernen".
3. Die Kennzeichnung "//" für Zeilenkommentare am Beginn der Zeile wird entfernt.

Ergebnis

Kommentare werden mit einer Hervorhebung formatiert.

Siehe auch

Regeln zu AWL (Seite 7572)
AWL-Anweisungen manuell eingeben (Seite 7575)
AWL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7576)
Operanden einfügen (Seite 7579)
Favoriten in AWL verwenden (Seite 7581)
Bausteinaufrufe in AWL einfügen (Seite 7583)

15.5 AWL-Anweisungen bearbeiten

15.5.1 AWL-Anweisungen selektieren

Sie können einzelne Anweisungen oder alle Anweisungen eines Netzwerks selektieren.

Voraussetzung

AWL-Anweisungen sind vorhanden.

Einzelne AWL-Anweisungen selektieren

Um einzelne AWL-Anweisungen zu selektieren, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke vor das erste Zeichen, das Sie markieren möchten.
2. Drücken und halten Sie die linke Maustaste.
3. Bewegen Sie den Mauszeiger hinter das letzte Zeichen, das Sie markieren möchten.
4. Lassen Sie die linke Maustaste los.

Alle AWL-Anweisungen eines Netzwerks selektieren

Um alle Anweisungen eines Netzwerks zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Netzwerk, dessen Anweisungen Sie selektieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

Siehe auch

AWL-Anweisungen kopieren (Seite 7592)

AWL-Anweisungen ausschneiden (Seite 7593)

AWL-Anweisungen aus der Zwischenablage einfügen (Seite 7594)

AWL-Anweisungen löschen (Seite 7594)

15.5.2 AWL-Anweisungen kopieren

Voraussetzung

Eine AWL-Anweisung ist vorhanden.

Vorgehen

Um eine AWL-Anweisung zu kopieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die AWL-Anweisung, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Ergebnis

Die AWL-Anweisung wird kopiert und in der Zwischenablage gespeichert.

Siehe auch

AWL-Anweisungen selektieren (Seite 7592)
AWL-Anweisungen ausschneiden (Seite 7593)
AWL-Anweisungen aus der Zwischenablage einfügen (Seite 7594)
AWL-Anweisungen löschen (Seite 7594)

15.5.3 AWL-Anweisungen ausschneiden

Voraussetzung

Eine AWL-Anweisung ist vorhanden.

Vorgehen

Um eine AWL-Anweisung auszuschneiden, gehen Sie folgendermaßen vor:

1. Markieren Sie die AWL-Anweisung, die Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

Ergebnis

Die AWL-Anweisung wird ausgeschnitten und in der Zwischenablage gespeichert.

Siehe auch

AWL-Anweisungen selektieren (Seite 7592)
AWL-Anweisungen kopieren (Seite 7592)
AWL-Anweisungen aus der Zwischenablage einfügen (Seite 7594)
AWL-Anweisungen löschen (Seite 7594)

15.5.4 AWL-Anweisungen aus der Zwischenablage einfügen

Voraussetzung

Eine AWL-Anweisung ist vorhanden.

Vorgehen

Um eine AWL-Anweisung aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie eine AWL-Anweisung oder schneiden Sie eine AWL-Anweisung aus.
2. Klicken Sie auf die Stelle im Netzwerk, an der Sie die AWL-Anweisung einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

AWL-Anweisungen selektieren (Seite 7592)
AWL-Anweisungen kopieren (Seite 7592)
AWL-Anweisungen ausschneiden (Seite 7593)
AWL-Anweisungen löschen (Seite 7594)

15.5.5 AWL-Anweisungen löschen

Voraussetzung

Eine AWL-Anweisung ist vorhanden.

Vorgehen

Um eine AWL-Anweisung zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die AWL-Anweisung, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

AWL-Anweisungen selektieren (Seite 7592)
AWL-Anweisungen kopieren (Seite 7592)
AWL-Anweisungen ausschneiden (Seite 7593)
AWL-Anweisungen aus der Zwischenablage einfügen (Seite 7594)

15.6 Programmierbeispiele AWL

15.6.1 Beispiel für das Steuern eines Förderbands

Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Drucktaster, S1 für START und S2 für STOP. Am Ende des Förderbands befinden sich ebenfalls zwei Drucktaster, S3 für START und S4 für STOP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.

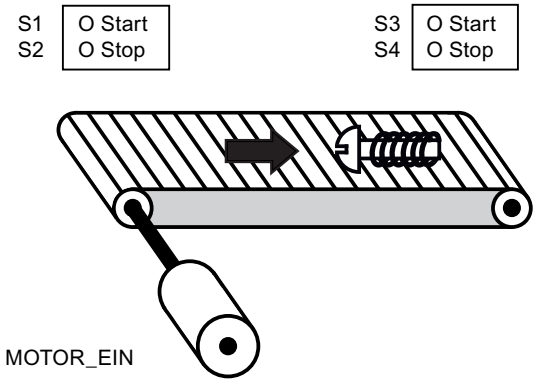

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Operand	Deklaration	Datentyp	Beschreibung
Starttaster_Links (S1)	Input	BOOL	Starttaster am linken Rand des Förderbands
Stoptaster_Links (S2)	Input	BOOL	Stoptaster am linken Rand des Förderbands
Starttaster_Rechts (S3)	Input	BOOL	Starttaster am rechten Rand des Förderbands
Stoptaster_Rechts (S4)	Input	BOOL	Stoptaster am rechten Rand des Förderbands
MOTOR_EIN	Output	BOOL	Motor des Förderbands einschalten

Das folgende AWL-Programm zeigt die Umsetzung der Aufgabe:

```
AWL                 Erläuterung
O #S1               // Starttaster S1 auf "1" abfragen.
O #S3               // Starttaster S3 auf "1" abfragen.
S #MOTOR_EIN        // Wenn einer der Starttaster (S1 oder S3) den Signalzustand "1"
                    liefert, wird der Motor des Förderbands eingeschaltet.
O #S2               // Stoptaster S2 auf "1" abfragen.
```

AWL	Erläuterung
O #S4	// Stoptaster S4 auf "1" abfragen.
R #MOTOR_EIN	// Wenn einer der Stoptaster (S2 oder S4) den Signalzustand "1" liefert, wird der Motor des Förderbands ausgeschaltet.

Siehe auch

Grundlagen zu AWL (Seite 7557)
Einstellungen zu AWL (Seite 7563)
Arbeiten mit Netzwerken (Seite 7564)
AWL-Anweisungen einfügen (Seite 7572)
AWL-Anweisungen bearbeiten (Seite 7592)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7596)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7598)
Beispiel für das Berechnen einer Gleichung (Seite 7600)
Beispiel für das Heizen eines Ofens (Seite 7602)
Beispiel für eine Schrittkette (Seite 7604)

15.6.2 Beispiel für das Erfassen der Richtung eines Förderbands

Erfassen der Richtung eines Förderbands

Die erkannte Laufrichtung des Bandes wird mit einem Pfeil RECHTS oder einem Pfeil LINKS angezeigt. Läuft weiteres Fördergut von rechts an LS1 oder von links an LS2 heran, wird zunächst der angezeigte Pfeil ausgeschaltet, bis nach dem Durchlaufen beider Lichtschranken die Laufrichtung wieder neu festgestellt wurde und der entsprechende Pfeil angezeigt werden kann. Für die Lösung der Aufgabe sind noch 2 Flankenmerker notwendig, die den Signalwechsel an den zwei Lichtschranken von "0" auf "1" erkennen.

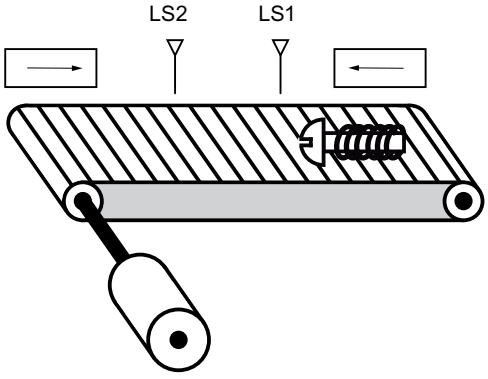

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
S1	Input	BOOL	Lichtschranke 1
S2	Input	BOOL	Lichtschranke 2
TM1	Input	BOOL	Flankenmerker 1
TM2	Input	BOOL	Flankenmerker 2
RECHTS	Output	BOOL	Anzeige für eine Bewegung nach rechts
LINKS	Output	BOOL	Anzeige für eine Bewegung nach links

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

```
AWL                 Erläuterung
U #S1               // Lichtschranke "S1" auf "1" abfragen
FP #TM1             // Positive Flanke abfragen
UN #S2              // Lichtschranke "S2" auf "0" abfragen
S #LINKS            // Wenn an der Lichtschranke "S1" ein Wechsel des Signalzustands
                    von "0" auf "1" auftritt (positive Flanke) und gleichzeitig der
                    Signalzustand der Lichtschranke "S2" "0" ist, dann bewegt sich
                    der Gegenstand auf dem Förderband nach links.
                    // Die Anzeige für eine Bewegung nach links wird eingeschaltet.
U #S2               // Lichtschranke "S2" auf "1" abfragen
FP #TM2             // Positive Flanke abfragen
UN #S1              // Lichtschranke "S1" auf "0" abfragen
S #RECHTS           // Wenn an der Lichtschranke "S2" ein Wechsel des Signalzustands
                    von "0" auf "1" auftritt (positive Flanke) und gleichzeitig der
                    Signalzustand an "S1" "0" ist, dann bewegt sich der Gegenstand
                    auf dem Förderband nach rechts.
                    // Die Anzeige für eine Bewegung nach rechts wird eingeschaltet.
UN #S1              // Lichtschranke "S1" auf "0" abfragen
UN #S2              // Lichtschranke "S2" auf "0" abfragen
R #LINKS            // Die Anzeige für eine Bewegung nach links wird ausgeschaltet,
                    wenn der Signalzustand an den beiden Lichtschranken "0" ist.
R #RECHTS           // Die Anzeige für eine Bewegung nach rechts wird ausgeschaltet,
                    wenn der Signalzustand an den beiden Lichtschranken "0" ist.
```

Siehe auch

Grundlagen zu AWL (Seite 7557)

Einstellungen zu AWL (Seite 7563)

Arbeiten mit Netzwerken (Seite 7564)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Beispiel für das Steuern eines Förderbands (Seite 7595)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7598)
Beispiel für das Berechnen einer Gleichung (Seite 7600)
Beispiel für das Heizen eines Ofens (Seite 7602)
Beispiel für eine Schrittkette (Seite 7604)

15.6.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete vom temporären Lagerbereich zu einer Laderampe, von der sie auf einen LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

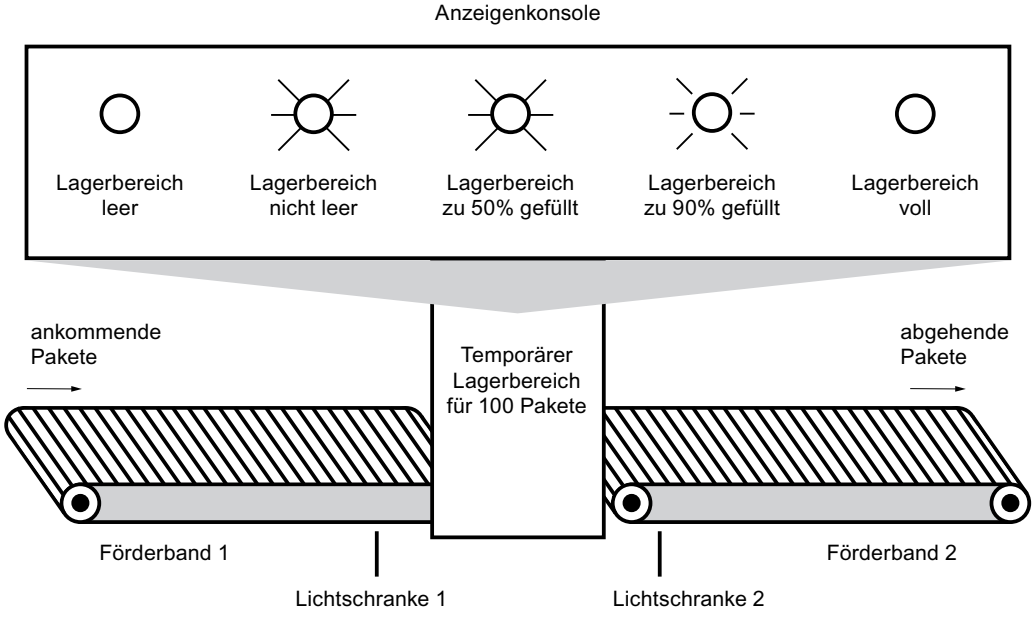

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Datentyp	Adresse	Beschreibung
PACKAGECOUNT	COUNTER	Z1	Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)

Name	Abschnitt	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
LAG_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich leer
LAG_NICHT_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich nicht leer
LAG_50%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 50% voll
LAG_90%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 90% voll
LAG_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich voll

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

```
 AWL                 Erläuterung
U #LS1               // Lichtschranke "LS1" auf "1" abfragen.
ZV "PACKAGECOUNT"    // Bei einer positiven Signalflanke an der Lichtschranke "LS1"
                     wird der Zählwert des Zählers "PACKAGECOUNT" um eins erhöht.
U #LS2               // Lichtschranke "LS2" auf "1" abfragen.
ZR "PACKAGECOUNT"    // Bei einer positiven Signalflanke an der Lichtschranke "LS2"
                     wird der Zählwert des Zählers "PACKAGECOUNT" um eins verringert.
UN "PACKAGECOUNT"    // Zählwert auf "0" abfragen.
= #LAG_LEER          // Bei einem Zählwert "0" wird die Anzeigeleuchte "Lagerbereich
                     leer" eingeschaltet.
U "PACKAGECOUNT"     // Zählwert auf "1" abfragen.
= #LAG_NICHT_LEER    // Bei einem Zählerstand größer "0" wird die Anzeigeleuchte "La-
                     gerbereich nicht leer" eingeschaltet.
L 50                 // Vergleichswert "50" in Akkumulator 1 laden.
L "PACKAGECOUNT"     // Vergleichswert in Akkumulator 2 verschieben.
                     // Aktuellen Zählwert in Akkumulator 1 laden.
<=I                  // Werte vergleichen
= #"LAG_50%_VOLL"    // Bei einem Zählwert größer oder gleich "50" wird die Anzeige-
                     leuchte "Lagerbereich zu 50% voll" eingeschaltet.
L 90                 // Zählwert in Akkumulator 2 verschieben.
                     // Vergleichswert "90" in Akkumulator 1 laden.
>=I                  // Werte vergleichen
= #"LAG_90%_VOLL"    // Bei einem Zählwert größer oder gleich "90" wird die Anzeige-
                     leuchte "Lagerbereich zu 90% voll" eingeschaltet.
```

```
 AWL                 Erläuterung
L "PACKAGECOUNT"     // Aktuellen Zählwert in Akkumulator 1 laden.
L 100                // Zählwert in Akkumulator 2 verschieben.
                     // Vergleichswert "100" in Akkumulator 1 laden.
>=I                  // Werte vergleichen
= #LAG_VOLL          // Bei einem Zählwert größer oder gleich "100" wird die Anzeige-
                     leuchte "Lagerbereich voll" eingeschaltet.
```

Siehe auch

Grundlagen zu AWL (Seite 7557)

Einstellungen zu AWL (Seite 7563)

Arbeiten mit Netzwerken (Seite 7564)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Beispiel für das Steuern eines Förderbands (Seite 7595)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7596)

Beispiel für das Berechnen einer Gleichung (Seite 7600)

Beispiel für das Heizen eines Ofens (Seite 7602)

Beispiel für eine Schrittkette (Seite 7604)

15.6.4 Beispiel für das Berechnen einer Gleichung

Berechnen einer Gleichung

Das folgende Programmbeispiel zeigt, wie Sie mit drei arithmetischen Operationen die folgende Gleichung berechnen können:

RESULT = ((A + B) x C) /D

Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablentabelle:

Name	Datentyp	Kommentar
A	INT	Erster Wert der Addition
B	INT	Zweiter Wert der Addition
C	INT	Multiplikator
D	INT	Divisor
RESULT	INT	Endergebnis

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

AWL	Erläuterung
L "A"	// Wert des Operanden "A" in Akkumulator 1 laden
L "B"	// Wert des Operanden "A" in Akkumulator 2 verschieben
	// Wert des Operanden "B" in Akkumulator 1 laden
+I	// Werte der Akkumulatoren 1 und 2 addieren
	// Summe im Akkumulator 1 speichern
L "C"	// Summe in Akkumulator 2 verschieben
	// Wert des Operanden "C" in Akkumulator 1 laden
*I	// Werte der Akkumulatoren 1 und 2 multiplizieren
	// Produkt im Akkumulator 1 speichern
L "D"	// Produkt in Akkumulator 2 verschieben
	// Wert des Operanden "D" in Akkumulator 1 laden
/I	// Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren
	// Ergebnis im Akkumulator 1 speichern
T "RESULT"	// Ergebnis in den Operanden "RESULT" übertragen

Siehe auch

Grundlagen zu AWL (Seite 7557)
Einstellungen zu AWL (Seite 7563)
Arbeiten mit Netzwerken (Seite 7564)
AWL-Anweisungen einfügen (Seite 7572)
AWL-Anweisungen bearbeiten (Seite 7592)
Beispiel für das Steuern eines Förderbands (Seite 7595)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7596)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7598)
Beispiel für das Heizen eines Ofens (Seite 7602)
Beispiel für eine Schrittkette (Seite 7604)

15.6.5 Beispiel für das Heizen eines Ofens

Heizen eines Ofens

Das folgende Bild zeigt einen Ofen, der mit einem Startschalter eingeschaltet wird. Wenn der Startschalter gedrückt wird, wird der Heizvorgang angestoßen. Die Heizdauer wird durch digitale Vorwählschalter festgelegt. Die Angabe der Heizdauer erfolgt dabei in Sekunden im BCD-Format.

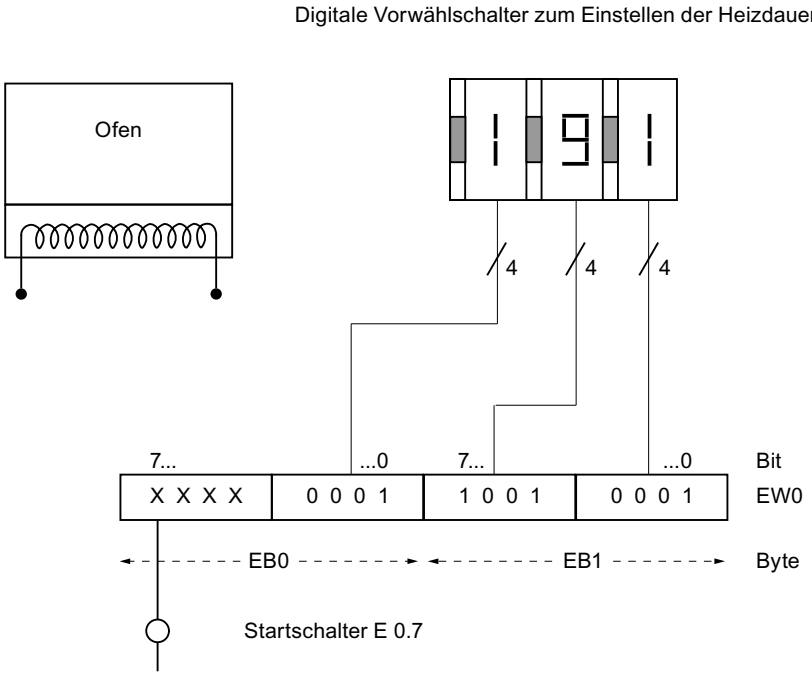

Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablentabelle:

Name	Datentyp	Adresse	Kommentar
DURATION	WORD	EW0	Heizdauer in Sekunden • E1.0 bis E1.3: Vorwählschalter für Einer • E1.4 bis E1.7: Vorwählschalter für Zehner • E0.0 bis E0.3: Vorwählschalter für Hunderter
HEATING	TIMER	T1	Zeit, die mit der voreingestellten Heizdauer gestartet wird.

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der Bausteinschnittstelle des Codebausteins:

Name	Abschnitt	Datentyp	Kommentar
START	Input	BOOL	Startschalter
START_HEATING	Output	BOOL	Beginn des Heizvorgangs

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

```
 AWL                 Erläuterung
U "HEATING"          // Abfrage, ob die Zeit gestartet ist
= #START_HEATING     // Heizvorgang anstoßen
BEB                  // Bei VKE=1 Bearbeitung des Bausteins beenden
                     // Dadurch wird ein Neustart der Zeit "HEATING" verhindert, wenn
                     der Startschalter gedrückt wird
L "DURATION"         // Heizdauer in den Akkumulator 1 laden
UW W#16#0FFF         // Eingangsbits E0.4 bis E0.7 auf "0" zurücksetzen.
OW W#16#4000         // Sekunden in Bits E1.2 und E1.3 des Akkumulators 1 einstellen
U #START             // Startschalter auf "1" abfragen
SV "HEATING"         // Bei einer positiven Signalflanke am Startschalter Heizvor-
                     gang als verlängerten Impuls starten
```

Siehe auch

Grundlagen zu AWL (Seite 7557)

Einstellungen zu AWL (Seite 7563)

Arbeiten mit Netzwerken (Seite 7564)

AWL-Anweisungen einfügen (Seite 7572)

AWL-Anweisungen bearbeiten (Seite 7592)

Beispiel für das Steuern eines Förderbands (Seite 7595)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7596)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7598)

Beispiel für das Berechnen einer Gleichung (Seite 7600)

Beispiel für eine Schrittkette (Seite 7604)

15.6.6 Beispiel für eine Schrittkette

Schrittkette programmieren

Das folgende Bild zeigt eine Station zum Entpalettieren von Glasbehältern. Die Paletten werden auf einem Förderband zur Station transportiert. Wenn eine Palette mit Glasbehältern in der Station eintrifft, wird das Förderband angehalten und ein Greifer aus seiner Grundposition (P0) zur Position über der Palette (P2) bewegt. Befindet sich der Greifer über der Palette, werden die Greifklemmen geöffnet und der Greifer wird gesenkt. Die aktuelle Position des Greifers und der Zustand der Greiferklemmen werden durch Sensoren erfasst. Der beschriebene Ablauf der Greiferbewegung wird in diesem Beispiel durch eine Schrittkette umgesetzt. Die weiteren Schritte zum Entnehmen der Flaschen und deren Transport auf einem weiteren Förderband können zusätzlich programmiert werden.

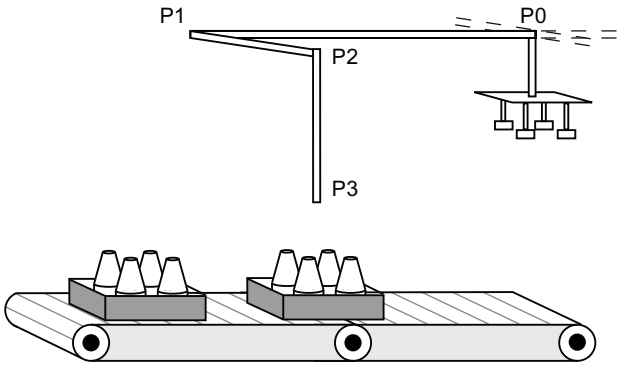

Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablentabelle:

Name	Datentyp	Kommentar
NUMBER	INT	Schrittnummer
Tag_Error	BOOL	Operand, der gesetzt wird, wenn die Schrittnummer größer 3 ist oder einer der Schritte nicht ausgeführt wurde.

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der Bausteinschnittstelle des Codebausteins:

Name	Abschnitt	Datentyp	Kommentar
POS_0	Input	BOOL	Greifer in Grundstellung (P0)
POS_1	Input	BOOL	Greifer in Position 1 (P1)
POS_2	Input	BOOL	Greifer in Position 2 (P2)
GRIPPER_OPEN	Input	BOOL	Greiferklemmen offen
OUT_POS_1	Output	BOOL	Greifer zur Position 1 bewegen

Name	Abschnitt	Datentyp	Kommentar
OUT_POS_2	Output	BOOL	Greifer zur Position 2 bewegen
OUT_GRIPPER	Output	BOOL	Greiferklemmen öffnen
OUT_POS_3	Output	BOOL	Greifer zur Position 3 bewegen

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

```
AWL                          Erläuterung
L "NUMBER"                   // Schrittnummer in Akkumulator 1 laden.
SPL END                      // Anfang der Sprungliste
SPA POSITION_0               // Bei einem Wert "0" im Akkumulator 1 zur Sprungmarke
                             "POSITION_0" springen.
SPA POSITION_1               // Bei einem Wert "1" im Akkumulator 1 zur Sprungmarke
                             "POSITION_1" springen.
SPA POSITION_2               // Bei einem Wert "2" im Akkumulator 1 zur Sprungmarke
                             "POSITION_2" springen.
SPA POSITION_3               // Bei einem Wert "3" im Akkumulator 1 zur Sprungmarke
                             "POSITION_3" springen.
END: SPA ERROR               // Ende der Sprungliste
                             // Bei einer Schrittnummer größer als 3 zur Sprungmar-
                             ke "ERROR" springen.
POSITION_0: U #POS_0         // Sprungmarke "POSITION_0"
                             // Abfrage, ob der Greifer in Grundstellung (P0) ist.
= #OUT_POS_1                 // Bei erfüllter Bedingung Ausgang "OUT_POS_1" setzen
                             und Greifer zur Position 1 (P1) bewegen.
SPBN ERROR                   // Bei VKE "0" zur Sprungmarke "ERROR" springen.
SPB NEXT                     // Bei VKE "1" zur Sprungmarke "NEXT" springen.
POSITION_1: U #POS_1         // Sprungmarke "POSITION_1"
                             // Abfrage, ob der Greifer in der Position 1 (P1) ist.
= #OUT_POS_2                 // Bei erfüllter Bedingung Ausgang "OUT_POS_2" setzen
                             und Greifer zur Position 2 (P2) bewegen.
SPBN ERROR                   // Bei VKE "0" zur Sprungmarke "ERROR" springen.
SPB NEXT                     // Bei VKE "1" zur Sprungmarke "NEXT" springen.
POSITION_2: U #POS_2         // Sprungmarke "POSITION_2"
                             // Abfrage, ob der Greifer in der Position 2 (P2) ist.
= #OUT_GRIPPER               // Bei erfüllter Bedingung Ausgang "OUT_GRIPPER" set-
                             zen und Greiferklemmen öffnen.
SPBN ERROR                   // Bei VKE "0" zur Sprungmarke "ERROR" springen.
SPB NEXT                     // Bei VKE "1" zur Sprungmarke "NEXT" springen.
POSITION_3: U #POS_2         // Sprungmarke "POSITION_3"
                             // Abfrage, ob der Greifer in der Position 2 (P2) ist.
U #GRIPPER_OPEN              // Abfrage, ob die Greiferklemmen offen sind
= #OUT_POS_3                 // Bei erfüllter Bedingung Ausgang "OUT_POS_3" setzen
                             und Greifer zur Position 3 (P3) senken.
SPBN ERROR                   // Bei VKE "0" zur Sprungmarke "ERROR" springen.
SPB NEXT                     // Bei VKE "1" zur Sprungmarke "NEXT" springen.
```

AWL	Erläuterung
NEXT: INC 1	// Sprungmarke "NEXT"
	// Schrittnummer im Akkumulator 1 um eins erhöhen.
T "NUMBER"	// Schrittnummer in den Operanden "NUMBER" transferieren.
L 3	// Aktuelle Schrittnummer in Akkumulator 2 verschieben.
	// Den Wert 3 in Akkumulator 1 laden.
>I	// Abfrage, ob die aktuelle Schrittnummer größer als 3 ist.
SPB RESET_NUMBER	// Bei einem Abfrageergebnis "1" zur Sprungmarke "RESET_NUMBER" springen und die Programmbearbeitung dort fortsetzen.
BEA	// Baustein beenden
RESET_NUMBER: L 0	// Sprungmarke "RESET_NUMBER"
	// Den Wert "0" in Akkumulator 1 laden.
T "NUMBER"	// Operanden "NUMBER" (Schrittnummer) den Wert "0" zuweisen.
BEA	// Baustein beenden
ERROR: NOT	// Sprungmarke "ERROR"
= "Tag_Error"	// Negiertes VKE dem Operanden "Tag_Error" zuweisen.
BEA	// Baustein beenden

Siehe auch

Grundlagen zu AWL (Seite 7557)
Einstellungen zu AWL (Seite 7563)
Arbeiten mit Netzwerken (Seite 7564)
AWL-Anweisungen einfügen (Seite 7572)
AWL-Anweisungen bearbeiten (Seite 7592)
Beispiel für das Steuern eines Förderbands (Seite 7595)
Beispiel für das Erfassen der Richtung eines Förderbands (Seite 7596)
Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 7598)
Beispiel für das Berechnen einer Gleichung (Seite 7600)
Beispiel für das Heizen eines Ofens (Seite 7602)

SCL-Programme erstellen 16

16.1 Grundlagen zu SCL

16.1.1 Programmiersprache SCL

Programmiersprache SCL

SCL (Structured Control Language) ist eine höhere Programmiersprache, die sich an PASCAL orientiert. Die Sprache basiert auf DIN EN-61131-3 (international IEC 1131-3).

Die Norm standardisiert die Programmiersprachen für speicherprogrammierbare Steuerungen. Die Programmiersprache SCL erfüllt den in dieser Norm definierten PLCopen Basis Level der Sprache ST (Strukturierter Text).

Sprachelemente

SCL enthält neben den typischen Elementen der SPS, wie Eingängen, Ausgängen, Zeiten, oder Merkern, auch Sprachelemente höherer Programmiersprachen:

- Ausdrücke
- Wertzuweisungen
- Operatoren

Programmsteuerung

SCL bietet komfortable Anweisungen zur Programmsteuerung, mit denen Sie z. B. Programmverzweigungen, -schleifen oder -sprünge realisieren können.

Anwendungsbereich

SCL ist damit für folgende Anwendungsbereiche besonders geeignet:

- Datenverwaltung
- Prozessoptimierung
- Rezepturverwaltung
- mathematische/statistische Aufgaben

16.1.2 Ausdrücke

Beschreibung

Ausdrücke werden zur Laufzeit des Programms berechnet und liefern einen Wert zurück. Ein Ausdruck besteht aus Operanden (z. B. aus Konstanten, Variablen oder Funktionsaufrufen) und optional aus Operatoren (z. B. *, /, + oder -). Ausdrücke können durch Operatoren miteinander verknüpft oder ineinander geschachtelt sein.

Auswertungsreihenfolge

Die Auswertung des Ausdrucks geschieht in einer bestimmten Reihenfolge, die durch folgende Faktoren festgelegt ist:

- Priorität der beteiligten Operatoren
- Links-Rechts-Reihenfolge
- Klammerung

Verwenden Sie Funktionsaufrufe, die globale oder statische Variablen beeinflussen, nicht innerhalb eines Ausdrucks.

Arten von Ausdrücken

Abhängig vom Operator werden folgende Ausdrucksarten unterschieden:

- Arithmetische Ausdrücke
  Arithmetische Ausdrücke bestehen entweder aus einem numerischen Wert oder verknüpfen zwei Werte oder Ausdrücke mit arithmetischen Operatoren.
- Vergleichsausdrücke
  Vergleichsausdrücke vergleichen die Werte zweier Operanden und liefern einen booleschen Wert. Das Ergebnis ist TRUE, falls der Vergleich erfüllt ist, und FALSE, falls er nicht erfüllt ist.
- Logische Ausdrücke
  Logische Ausdrücke verknüpfen zwei Operanden durch logische Operatoren (AND, OR, XOR) oder negieren Operanden (NOT).

Verwendung von Ausdrücken

Das Ergebnis eines Ausdrucks können Sie unterschiedlich verwenden:

- als Wertzuweisung zu einer Variablen
- als Bedingung für eine Kontrollanweisung
- als Parameter für den Aufruf eines Bausteins oder einer Anweisung

Siehe auch

Operatoren und Operatorenrangfolge (Seite 7618)

Arithmetische Ausdrücke (Seite 7609)

Vergleichsausdrücke (Seite 7612)

Logische Ausdrücke (Seite 7617)

16.1.3 Arithmetische Ausdrücke

Beschreibung

Arithmetische Ausdrücke bestehen entweder aus einem numerischen Wert oder verknüpfen zwei Werte oder Ausdrücke mit arithmetischen Operatoren.

Arithmetische Operatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Wenn an der Operation zwei Operanden beteiligt sind, dann wird der Datentyp des Ergebnisses nach folgenden Kriterien bestimmt:

- Wenn die beiden Operanden Ganzzahlen mit Vorzeichen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Ganzzahl (z. B. INT + DINT = DINT).
- Wenn die beiden Operanden Ganzzahlen ohne Vorzeichen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Ganzzahl (z. B. USINT + UDINT = UDINT).
- Wenn ein Operand eine Ganzzahl mit Vorzeichen und der andere Operand eine Ganzzahl ohne Vorzeichen ist, erhält das Ergebnis den nächst größeren Datentyp mit Vorzeichen, der die Ganzzahl ohne Vorzeichen abdeckt (z. B. SINT + USINT = INT).
  Eine Operation mit solchen Operanden lässt sich nur ausführen, wenn keine IEC-Prüfung eingestellt ist.
- Wenn ein Operand eine Ganzzahl und der andere Operand eine Gleitpunktzahl ist, erhält das Ergebnis den Datentyp der Gleitpunktzahl (z. B. INT + REAL = REAL).
- Wenn die beiden Operanden Gleitpunktzahlen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Gleitpunktzahl (z. B. REAL + LREAL = LREAL).
- Den Datentyp des Ergebnisses einer Operation an der Operanden von den Datentypgruppen "Zeiten" und "Datum und Uhrzeit" beteiligt sind, können Sie der Tabelle im Abschnitt "Datentypen von arithmetischen Ausdrücken" entnehmen.
  Bei eingestellter IEC-Prüfung können Sie keine Datentypen aus den Datentypgruppen "Zeiten" und "Datum und Uhrzeit" verwenden.

Datentypen von arithmetischen Ausdrücken

Die folgende Tabelle zeigt die Datentypen, die Sie in arithmetischen Ausdrücken verwenden können:

Operation	Operator	1. Operand	2. Operand	Ergebnis
Potenz	**	Ganzzahl/Gleitpunkt-zahl	Ganzzahl/Gleitpunktzahl	Gleitpunktzahl
Unäres Plus	+	Ganzzahl/Gleitpunkt-zahl TIME, LTIME	-	Ganzzahl/Gleitpunkt-zahl TIME, LTIME

Operation	Operator	1. Operand	2. Operand	Ergebnis
Unäres Minus	-	Ganzzahl/Gleitpunkt-zahl TIME, LTIME	-	Ganzzahl/Gleitpunkt-zahl TIME, LTIME
Multiplikation	*	Ganzzahl/Gleitpunkt-zahl	Ganzzahl/Gleitpunktzahl	Ganzzahl/Gleitpunkt-zahl
		TIME, LTIME	Ganzzahl	TIME, LTIME
Division	/	Ganzzahl/Gleitpunkt-zahl	Ganzzahl/Gleitpunkt-zahl (ungleich 0)	Ganzzahl/Gleitpunkt-zahl
		TIME, LTIME	Ganzzahl	TIME, LTIME
Modulo-Funktion	MOD	Ganzzahl	Ganzzahl	Ganzzahl
Addition	+	Ganzzahl/Gleitpunkt-zahl	Ganzzahl/Gleitpunktzahl	Ganzzahl/Gleitpunkt-zahl
		TIME	TIME	TIME
		TIME	DINT	TIME
		LTIME	TIME, LTIME	LTIME
		LTIME	LINT	LTIME
		TOD	TIME	TOD
		TOD	DINT	TOD
		LTOD	TIME, LTIME	LTOD
		LTOD	LINT	LTOD
		DATE	LTOD	DTL
		DATE	TOD	• S7-300/400: DT • S7-1200/1500: DTL
		DT	TIME	DT
		LDT	TIME, LTIME	LDT
		DTL	TIME, LTIME	DTL

Operation	Operator	1. Operand	2. Operand	Ergebnis
Subtraktion	-	Ganzzahl/Gleitpunkt-zahl	Ganzzahl/Gleitpunktzahl	Ganzzahl/Gleitpunkt-zahl
		TIME	TIME	TIME
		TIME	DINT	TIME
		LTIME [1]	TIME, LTIME	LTIME
		LTIME	LINT	LTIME
		TOD	TIME	TOD
		TOD	DINT	TOD
		TOD	TOD	TIME
		LTOD	TIME, LTIME	LTOD
		LTOD	LINT	LTOD
		LTOD	LTOD	LTIME
		DATE	DATE	• S7-300/400/1200: TIME • S7-1500: LTIME
		DT	TIME	DT
		DT	DT	TIME
		LDT	TIME, LTIME	LDT
		DTL	TIME, LTIME	DTL
		DTL	DTL	• S7-1200: TIME • S7-1500: LTIME
[1] Kombinationen zwischen Nanosekunden und Millisekunden sind innerhalb von Ausdrücken nicht möglich.				

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Beispiel

Das folgende Beispiel zeigt einen arithmetischen Ausdruck:

```
SCL
"MyTag1":= "MyTag2" * "MyTag3";
```

Siehe auch

Ausdrücke (Seite 7608)

Operatoren und Operatorenrangfolge (Seite 7618)

16.1.4 Vergleichsausdrücke

Beschreibung

Vergleichsausdrücke vergleichen die Werte oder die Datentypen zweier Operanden und liefern einen booleschen Wert. Das Ergebnis ist TRUE, falls der Vergleich erfüllt ist, und FALSE, falls er nicht erfüllt ist.

Vergleichsperatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Der Datentyp des Ergebnisses ist immer BOOL.

Die folgenden Regeln müssen bei der Bildung von Vergleichsausdrücken beachtet werden:

- Innerhalb folgender Datentypgruppen sind alle Variablen vergleichbar:
  - Ganzzahlen/Gleitpunktzahlen
  - Binärzahlen
  - Zeichenfolgen
- Bei folgenden Datentypen/Datentypgruppen sind nur typgleiche Variablen vergleichbar:
  - TIME, LTIME
  - Datum und Uhrzeit
  - PLC-Datentypen
  - ARRAY
  - STRUCT
  - Variable, auf die der ANY zeigt
  - Variable, auf die der VARIANT zeigt
- Der Vergleich von STRINGs erfolgt nach der Zeichencodierung in dem in Windows eingestellten Zeichensatz. Der Vergleich von WSTRINGs erfolgt nach der Zeichencodierung in UTF-16. Für den Vergleich werden die Länge der Variablen und der numerische Wert jedes Zeichens herangezogen.
- S5TIME-Variablen sind als Vergleichsoperanden nicht erlaubt. Eine explizite Konvertierung von S5TIME nach TIME oder LTIME ist notwendig.

Vergleich von Gleitpunktzahlen

Bei Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

Die speziellen Bitmuster von ungültigen Gleitpunktzahlen (NaN), die bei undefinierten Ergebnissen entstehen (z. B. Wurzel aus -1) sind nicht vergleichbar. D.h., hat einer der beiden Operanden den Wert NaN, so liefern sowohl der Vergleichsausdruck "==: Gleich" als auch "<>: Ungleich" als Ergebnis FALSE.

Vergleich von Zeichenketten

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche "==" von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'AA'	1
'Hallo Welt'	'HalloWelt'	0
'AA'	'aa'	0
'aa'	'aaa'	0

Die folgende Tabelle zeigt Beispiele für Vergleiche "<>" von Zeichenketten:

<Operand1>	<Operand2>	VKE der Anweisung
'AA'	'aa'	1
'Hallo Welt'	'HalloWelt'	1
'AA'	'AA'	0
'aa'	'aaa'	1

Sie können auch einzelne Zeichen einer Zeichenkette vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird z. B. das zweite Zeichen der Zeichenkette "MyString" verglichen.

Vergleich von Zeiten, Datum und Uhrzeit

Bitmuster von ungültigen Zeiten, Datum und Uhrzeiten, z. B. DT#2015-13-33-25:62:99.999_999_999, sind nicht vergleichbar. D.h., hat einer der beiden Operanden einen ungültigen Wert, so liefern sowohl die Anweisung "==: Gleich" als auch "<>: Ungleich" als Ergebnis FALSE.

Es können nicht alle Zeiten direkt miteinander verglichen werden, wie z. B. S5TIME. In diesem Fall werden sie implizit in eine andere Zeit umgewandelt, damit sie verglichen werden können, z. B. in TIME.

Wenn Sie Datum und Uhrzeiten unterschiedlicher Datentypen miteinander vergleichen wollen, dann wird der Wert des kleineren Datum- bzw. Uhrzeit-Datentyps implizit in den größeren Datum- bzw. Uhrzeit-Datentyp konvertiert. So werden z. B. die beiden Datum- und Uhrzeit-Datentypen DATE und DTL auf Basis von DTL verglichen.

Wenn die impliziten Konvertierungen fehlschlagen, dann liefert der Vergleich als Ergebnis FALSE.

Vergleich von Variablen vom Datentyp WORD mit Variablen vom Datentyp S5TIME

Beim Vergleich einer Variablen vom Datentyp WORD mit einer Variablen vom Datentyp S5TIME werden beide Variablen in den Datentyp TIME konvertiert. Die WORD-Variable wird dabei als S5TIME-Wert interpretiert. Falls eine der beiden Variablen nicht konvertiert werden kann, wird der Vergleich nicht durchgeführt und das Ergebnis ist FALSE. Bei erfolgreicher Konvertierung wird der Vergleich abhängig von der gewählten Vergleichsanweisung durchgeführt.

Vergleich von Strukturen

Hinweis

Verfügbarkeit des Vergleichs von Strukturen

Die Möglichkeit Strukturen zu vergleichen, steht Ihnen bei einer CPU der Baureihe S7-1200 ab der Firmware Version >= 4.2 und bei einer CPU der Baureihe S7-1500 ab der Firmware Version >= 2.0 zur Verfügung.

Sie können die Werte zweier strukturierter Operanden miteinander vergleichen, wenn die beiden Variablen vom gleichen Struktur-Datentyp sind. Bei Vergleichen von Strukturen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein. Eine Ausnahme bilden Vergleiche, bei denen einer der beiden Operanden ein VARIANT oder ein ANY ist. Wenn der Datentyp zur Zeit der Programmerstellung noch nicht bekannt ist, dann können Sie den VARIANT verwenden. In diesem Fall können Sie den Operanden auch mit einer strukturierten Variable eines beliebigen Datentyps vergleichen. Sie können auch zwei Variablen vom Datentyp VARIANT oder ANY miteinander vergleichen.

Wählen Sie für den Vergleich von Strukturen den Datentyp VARIANT aus der Klappliste der Anweisungsbox aus. Variablen aus folgenden Datentypen sind möglich:

- PLC-Datentyp (UDT)
- STRUCT (Die Struktur vom Datentyp STRUCT muss entweder in einem PLC-Datentyp enthalten sein oder die beiden zu vergleichenden Strukturen sind zwei Elemente von einem ARRAY of STRUCT. Instanzdatenbausteine und Variablen von anonymen Strukturen sind nicht zulässig.)
- Variable, auf die der ANY zeigt
- Variable, auf die der VARIANT zeigt

Damit Sie zwei Variablen vom Datentyp ARRAY miteinander vergleichen können, müssen folgende Voraussetzungen erfüllt sein:

- Die Elemente müssen jeweils den gleichen Datentyp haben.
- Die beiden ARRAYs müssen die gleiche Dimension haben.
- Alle Dimensionen müssen die gleiche Anzahl an Elementen haben. Die genauen ARRAY-Grenzen müssen nicht miteinander übereinstimmen.

Hinweis

ARRAY of BOOL

Wenn Sie zwei Operanden vom Datentyp ARRAY of BOOL miteinander vergleichen und die Anzahl der Elemente nicht durch 8 teilbar ist, dann werden auch die Füll-Bits verglichen. Das kann das Vergleichsergebnis beeinflussen.

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen auf "==: Gleich":

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit der Variable vom Datentyp A)		Wert der Variable	1
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Die folgende Tabelle zeigt ein Beispiel für einen Vergleich von Strukturen auf "<>: Ungleich":

<Operand1>			**<Operand2>**			**VKE der Anweisung**
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	Variable vom Datentyp B <PLC-Datentyp>		Wert der Variable	1
	BOOL	FALSE		BOOL	TRUE	
	INT	2		INT	3	

<Operand1>			<Operand2>			VKE der Anweisung
Variable vom Datentyp A <PLC-Datentyp>		Wert der Variable	VARIANT (versorgt mit Variable vom Datentyp A)		Wert der Variable	0
	BOOL	FALSE		BOOL	FALSE	
	INT	2		INT	2	

Datentypen von Vergleichsausdrücken

Die folgende Tabelle zeigt die Datentypen/Datentypgruppen, die Sie in Vergleichsausdrücken verwenden können:

Operation	Operator	1. Operand	2. Operand	Ergebnis
Vergleich auf gleich, ungleich	=, <>	Ganzzahl/Gleitpunktzahl	Ganzzahl/Gleitpunktzahl	BOOL
		Bitfolgen	Bitfolgen	BOOL
		Zeichenfolge	Zeichenfolge	BOOL
		TIME, LTIME	TIME, LTIME	BOOL
		Datum und Uhrzeit	Datum und Uhrzeit	BOOL
		VARIANT/ANY	VARIANT/ANY	BOOL
		Beliebiger Datentyp (Muss aber dem Datentyp entsprechen, mit dem der VARIANT versorgt wird.)	VARIANT/ANY	BOOL
		VARIANT/ANY	Beliebiger Datentyp	BOOL
		PLC-Datentyp	PLC-Datentyp	BOOL
		ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen	ARRAY of <Datentyp> mit festen und variablen ARRAY-Grenzen	BOOL
		STRUCT	STRUCT	BOOL

Operation	Operator	1. Operand	2. Operand	Ergebnis
Vergleich auf kleiner, kleiner-gleich, größer, größer-gleich	<, <=, >, >=	Ganzzahl/Gleitpunktzahl	Ganzzahl/Gleitpunktzahl	BOOL
		Bitfolgen (nur S7-1200/1500)	Bitfolgen (nur S7-1200/1500)	BOOL
		Zeichenfolge	Zeichenfolge	BOOL
		TIME, LTIME	TIME, LTIME	BOOL
		Datum und Uhrzeit	Datum und Uhrzeit	BOOL

Beispiel

Das folgende Beispiel zeigt einen Vergleichsausdruck:

```
SCL
IF a > b THEN c:= a;
IF A > 20 AND B < 20 THEN C:= TRUE;
IF A<>(B AND C) THEN C:= FALSE;
```

Hinweis

Der Vergleich bei STRING und DT wird bei S7-300/400 intern durch erweiterte Anweisungen durchgeführt. Folgende Operanden sind für diese Funktionen nicht zulässig:

- Parameter eines FC
- Durchgangsparameter eines FB vom Typ STRUCT oder ARRAY

Hinweis

Vergleich der beiden Hardware-Datentypen HW_IO und HW_DEVICE

Wenn Sie die beiden Datentypen vergleichen möchten, dann müssen Sie zuerst eine Variable vom Datentyp HW_ANY im Abschnitt "Temp" der Bausteinschnittstelle anlegen und dann den LADDR (vom Datentyp HW_DEVICE) in die Variable umkopieren. Anschließend ist ein Vergleich von HW_ANY und HW_IO möglich.

Siehe auch

Ausdrücke (Seite 7608)

Operatoren und Operatorenrangfolge (Seite 7618)

16.1.5 Logische Ausdrücke

Beschreibung

Logische Ausdrücke verknüpfen zwei Operanden durch logische Operatoren (AND, OR, XOR) oder negieren Operanden (NOT).

Logische Operatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Das Ergebnis eines logischen Ausdrucks ist vom Datentyp BOOL, wenn beide Operanden vom Datentyp BOOL sind. Wenn mindestens einer der beiden Operanden eine Bitfolge ist, ist das Ergebnis ebenfalls eine Bitfolge und wird durch den Typ des höchstwertigen Operanden bestimmt. Wenn Sie zum Beispiel einen Operanden vom Typ BYTE mit einem Operanden vom Typ WORD verknüpfen, ist das Ergebnis vom Typ WORD.

Um einen Operanden vom Typ BOOL mit einer Bitfolge zu verknüpfen, müssen Sie ihn zunächst durch explizite Konvertierung in eine Bitfolge umwandeln.

Datentypen von logischen Ausdrücken

Die folgende Tabelle zeigt die Datentypen, die Sie in logischen Ausdrücken verwenden können:

Operation	Operator	1. Operand	2. Operand	Ergebnis
Negation (Einerkomplement erzeugen)	NOT	BOOL	-	BOOL
		Bitfolge	-	Bitfolge
Und-Verknüpfung	AND oder &	BOOL	BOOL	BOOL
		Bitfolge	Bitfolge	Bitfolge
Oder-Verknüpfung	OR	BOOL	BOOL	BOOL
		Bitfolge	Bitfolge	Bitfolge
Exklusiv-Oder-Verknüpfung	XOR	BOOL	BOOL	BOOL
		Bitfolge	Bitfolge	Bitfolge

Beispiel

Das folgende Beispiel zeigt einen logischen Ausdruck:

SCL

```
IF "MyTag1" AND NOT "MyTag2" THEN c := a;
MyTag := ALPHA OR BETA;
```

Siehe auch

Ausdrücke (Seite 7608)

Operatoren und Operatorenrangfolge (Seite 7618)

Programmierbeispiele SCL (Seite 7676)

16.1.6 Operatoren und Operatorenrangfolge

Operatoren und ihre Auswertungsreihenfolge

Ausdrücke können durch Operatoren miteinander verknüpft oder ineinander geschachtelt sein.

Die Reihenfolge der Auswertung von Ausdrücken hängt von der Rangfolge der Operatoren und der Klammerung ab. Grundsätzlich gelten folgende Regeln:

- Arithmetische Operatoren werden vor Vergleichsoperatoren und diese vor logischen Operatoren ausgewertet.
- Gleichrangige Operatoren werden nach ihrem Auftreten von links nach rechts ausgewertet.
- Wertzuweisungen werden von rechts nach links ausgewertet.
- Operationen in Klammern werden zuerst ausgewertet.

Die folgende Tabelle gibt eine Übersicht über die Operatoren und ihre Rangfolge:

Operator	Operation	Rangfolge
Arithmetische Ausdrücke		
+ (Seite 7609)	Unäres Plus	2
- (Seite 7609)	Unäres Minus	2
** (Seite 7609)	Potenz	3
* (Seite 7609)	Multiplikation	4
/ (Seite 7609)	Division	4
MOD (Seite 7609)	Modulo-Funktion	4
+ (Seite 7609)	Addition	5
- (Seite 7609)	Subtraktion	5
+= (Seite 7609), -= (Seite 7609), *= (Seite 7609), /= (Seite 7609)	Kombinierte Wertzuweisungen	11
Vergleichsausdrücke		
< (Seite 7612)	Kleiner	6
> (Seite 7612)	Größer	6
<= (Seite 7612)	Kleiner gleich	6
>= (Seite 7612)	Größer gleich	6
= (Seite 7612)	Gleich	7
<> (Seite 7612)	Ungleich	7
Logische Ausdrücke		
NOT (Seite 7617)	Negation	3
AND (Seite 7617) oder & (Seite 7617)	Boolesches Und	8
XOR (Seite 7617)	Exklusives Oder	9
OR (Seite 7617)	Boolesches Oder	10
Referenzausdrücke		
REF (Seite 320)	Referenzieren	
^ (Seite 324)	Dereferenzieren	1
?= (Seite 329)	Zuweisungsversuch	11
Weitere Operationen		
( ) (Seite 7608)	Klammerung	1
:= (Seite 7620)	Zuweisung	11

16.1.7 Wertzuweisungen

Definition

Mit einer Wertzuweisung weisen Sie einer Variablen den Wert eines Ausdrucks zu. Auf der linken Seite der Zuweisung steht die Variable, die den Wert des rechts stehenden Ausdrucks übernimmt.

Als Ausdruck kann auch der Name einer Funktion angegeben werden. Die Funktion wird durch die Wertzuweisung aufgerufen und liefert ihren Funktionswert an die links stehende Variable zurück.

Der Datentyp der Wertzuweisung wird durch den Datentyp der links stehenden Variablen definiert. Der Datentyp des rechts stehenden Ausdrucks muss zu diesem Datentyp passen.

Sie haben die folgenden Möglichkeiten, Wertzuweisungen zu programmieren:

- Einfache Wertzuweisungen
  Bei einfachen Wertzuweisungen wird ein Ausdruck oder eine Variable nur einer einzigen Variablen zugewiesen:
  Beispiel: `a := b;`
- Mehrfache Wertzuweisungen
  Bei mehrfachen Wertzuweisungen können mehrere Zweisungen mit einer Anweisung durchgeführt werden.
  Beispiel: `a := b := c;`
  Dies entspricht der folgenden Schreibweise:
  ```
  b := c;
  a := b;
  ```
- Kombinierte Wertzuweisungen
  Bei kombinierten Wertzuweisungen können Sie die Operatoren "+", "-", "*" und "/" mit dem Zuweisungsoperator kombinieren:
  Beispiel: `a += b;`
  Dies entspricht der folgenden Schreibweise:
  ```
  a := a + b;
  ```
  Sie können kombinierte Wertzuweisungen auch mehrfach zuweisen:
  ```
  a += b += c *= d;
  ```
  Dies entspricht der folgenden Abfolge von Zuweisungen:
  ```
  c := c * d;
  b := b + c;
  a := a + b;
  ```

Wertzuweisungen für Datentyp STRUCT oder PLC-Datentypen

Eine gesamte Struktur ist einer anderen Struktur zuweisbar, wenn die Strukturen identisch aufgebaut sind und sowohl die Datentypen als auch die Namen der Strukturkomponenten übereinstimmen.

Sie können einem einzelnen Strukturelement eine Variable, einen Ausdruck oder ein anderes Strukturelement zuweisen.

Wertzuweisungen für Datentyp ARRAY

Ein gesamtes ARRAY ist einem anderen ARRAY zuweisbar, wenn sowohl die Datentypen der ARRAY-Elemente als auch die ARRAY-Grenzen übereinstimmen.

Sie können einem einzelnen ARRAY-Element eine Variable, einen Ausdruck oder ein anderes ARRAY-Element zuweisen.

Wertzuweisungen für Datentyp STRING

Ein gesamter STRING ist einem anderen STRING zuweisbar.

Sie können einem einzelnen STRING-Element ein anderes STRING-Element zuweisen.

Wertzuweisung für Datentyp WSTRING (S7-1200/1500)

Ein gesamter WSTRING ist einem anderen WSTRING zuweisbar.

Sie können einem einzelnen WSTRING-Element ein anderes WSTRING-Element zuweisen.

Wertzuweisungen für Datentyp ANY

Eine Variable mit dem Datentyp ANY können Sie nur folgenden Objekten zuweisen:

- Eingangparametern oder temporären Lokaldaten von FBs, die ebenfalls den Datentyp ANY haben.
- Temporären Lokaldaten von FCs, die ebenfalls den Datentyp ANY haben.

Beachten Sie, dass Sie mit dem Zeiger ANY nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen können.

Wertzuweisungen für Datentyp POINTER

In SCL können Sie POINTER in Wertzuweisungen nicht verwenden.

Beispiele

Die folgende Tabelle zeigt Beispiele für einfache Wertzuweisungen:

SCL	
`"MyTag1" := "MyTag2";`	`(* Zuweisung einer Variablen*)`
`"MyTag1" := "MyTag2" * "MyTag3";`	`(* Zuweisung eines Ausdrucks*)`
`"MyTag" := "MyFC"();`	`(* Aufruf einer Funktion, die ihren Funktionswert der Variablen "MyTag" zuweist*)`
`#MyStruct.MyStructElement := "MyTag";`	`(* Zuweisung einer Variablen an ein Strukturelement*)`
`#MyArray[2] := "MyTag";`	`(* Zuweisung einer Variablen an ein ARRAY-Element*)`
`"MyTag" := #MyArray[1,4];`	`(* Zuweisung eines ARRAY-Elements an eine Variable*)`

SCL

```
#MyString[2] := #MyOtherString[5];          (* Zuweisung eines STRING-Elements an
                                             ein anderes STRING-Element*)
```

Die folgende Tabelle zeigt Beispiele für mehrfache Wertzuweisungen:

SCL

```
"MyTag1" := "MyTag2" := "MyTag3";           (* Zuweisung einer Variablen*)
"MyTag1" := "MyTag2" := "MyTag3" * "My-     (* Zuweisung eines Ausdrucks*)
Tag4";
"MyTag1" := "MyTag2" := "MyTag3 :=          (* Aufruf einer Funktion, die ihren Funk-
"MyFC"();                                   tionswert den Variablen "MyTag1", "My-
                                            Tag1" und "MyTag1" zuweist*)
#MyStruct.MyStructElement1 := #MySt-        (* Zuweisung einer Variablen an zwei
ruct.MyStructElement2 := "MyTag";           Strukturelemente*)
#MyArray[2] := #MyArray[32] := "MyTag";     (* Zuweisung einer Variablen an zwei AR-
                                            RAY-Elemente*)
"MyTag1" := "MyTag2" := #MyArray[1,4];      (* Zuweisung eines ARRAY-Elements an zwei
                                            Variablen*)
#MyString[2] := #MyString[3]:= #MyO-        (* Zuweisung eines STRING-Elements an
therString[5];                              zwei STRING-Elemente*)
```

Die folgende Tabelle zeigt Beispiele für kombinierte Wertzuweisungen:

SCL

```
"MyTag1" += "MyTag2";                       (* "MyTag1" und "MyTag2" werden addiert
                                            und das Ergebnis der Addition wird "My-
                                            Tag1" zugewiesen.*)
"MyTag1" -= "MyTag2" += "MyTag3";           (* "MyTag2" und "MyTag3" werden addiert.
                                            Das Ergebnis der Addition wird "MyTag2"
                                            zugewiesen und anschließend von "MyTag1"
                                            subtrahiert. Das Ergebnis wird "MyTag1"
                                            zugewiesen.*)
#MyArray[2] += #MyArray[32] += "MyTag";     (* Das ARRAY-Element "MyArray[32]" wird
                                            mit "MyTag" addiert. Das Ergebnis wird
                                            "MyArray[32]" zugewiesen. Anschließend
                                            werden die beiden ARRAY-Elemente addiert
                                            und das Ergebnis wird dem ARRAY-Element
                                            "MyArray[2]" zugewiesen. Die Datentypen
                                            müssen dabei kompatibel sein.*)
#MyStruct.MyStructElement1 /= #MySt-        (* Das Strukturelement "MyStructElement2"
ruct.MyStructElement2 *= "MyTag";           wird mit "MyTag" multipliziert. Das Ergeb-
                                            nis wird "MyStructElement2" zugewiesen.
                                            Anschließend wird das Strukturelement
                                            "MyStructElement1" durch "MyStructEle-
                                            ment2" dividiert und das Ergebnis wird
                                            "MyStructElement1" zugewiesen. Die Daten-
                                            typen müssen dabei kompatibel sein.*)
```

Siehe auch

Operatoren und Operatorenrangfolge (Seite 7618)

Operanden adressieren (Seite 108)

16.1.8 Mit Gleitpunktzahlen (REAL und LREAL) in SCL rechnen

Darstellung der Genauigkeit von Gleitpunktzahlen

Der Datentyp REAL z. B. wird mit einer Genauigkeit von 6 Nachkommastellen im Programm angegeben und berechnet. Bei der Berechnung von Gleitpunktzahlen (REAL und LREAL) ist dabei generell zu beachten, dass diese Genauigkeit für jeden einzelnen Rechenschritt gilt.

Beim Addieren und Subtrahieren von Gleitpunktzahlen werden die Exponenten angeglichen. Damit sind die Basis und die Exponenten beim Addieren und Subtrahieren gleich und es werden nur die Mantissen addiert. Weitere Informationen zum Aufbau einer Gleitpunktzahl finden Sie unter "Siehe auch".

Programmierbeispiel

Im folgenden Programmierbeispiel sollen Sie eine Rechnung ausführen, bei der zwei Operanden vom Datentyp REAL addiert und einer subtrahiert werden soll. Im nächsten Rechenschritt wird die Konstante 1 durch das vorherige Ergebnis geteilt. Erstellen Sie dazu einen globalen Datenbaustein, in dem Sie Ihre Operanden deklarieren und eine Funktion, in der Sie die Rechenoperationen programmieren.

Rechenformeln

```
y = a + b - c
Z = 1/y
```

Die Operanden werden mit folgenden Werten hinterlegt:

Operand	Wert	REAL-Wert
a	100 000 000	$1.000000*10^8$
b	1	$1.000000*10^0$
c	100 000 000	$1.000000*10^8$

Vorgehensweise

Legen Sie den Datenbaustein "DB_GlobalData" an:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Geben Sie den Namen "DB_GlobalData" an.
4. Wählen Sie als Typ des Datenbausteins "Global-DB" aus.

5. Klicken Sie auf "OK".
6. Legen Sie folgende Variablen im Datenbaustein an und geben Sie die entsprechenden Startwerte ein:

DB_GlobalData			
	Name	Data type	Start value
1	Static		
2	a	Real	1.0E+8
3	b	Real	1.0
4	c	Real	1.0E+8

Der Startwert der beiden Variablen beträgt 100000000.0 und wird entsprechend des Datentyps REAL in 1.0E+8 umgewandelt.

Legen Sie eine SCL-Funktion an und benennen Sie sie "FC_Calculate".

1. Deklarieren Sie die Bausteinschnittstelle wie folgt:

FC_Calculate		
	Name	Data type
1	Input	
2	Output	
3	InOut	
4	Temp	
5	z	Real
6	y	Real
7	Constant	
8	Return	
9	FC_Calculate	Void

2. Schreiben Sie folgende Formeln in Ihren Programmcode und verbinden Sie sich Online um das Ergebnis zu sehen:

SCL

```
#y := "DB_GlobalData".a + "DB_GlobalData".b - "DB_GlobalData".c;
#z := 1/#y;
```

#y	0.000000E+000
"Global data".a	1.000000E+008
"Global data".b	1.000000E+000
"Global data".c	1.000000E+008

Wie Sie sehen, ist das Ergebnis am Operanden #y = 0, obwohl eigentlich die Zahl 1 als Ergebnis erwartet wird.

Dieses falsche Ergebnis kommt folgendermaßen zu stande:

1. Im ersten Rechenschritt werden die Operanden a + b addiert. Die REAL-Werte der beiden Operanden (a = $1.000000*10^8$ und b = $1.000000*10^0$) sehen nach dem Angleichen der Exponenten wie folgt aus:
   a = $1.000000*10^8$ und b = $0.00000001*10^8$. Die letzten beiden Stellen der zweiten Zahl (Operand b) werden abgeschnitten, da sie aufgrund der Genauigkeit von 6 Nachkommastellen nicht mehr darstellbar sind. Es wird also eine 0 statt einer 1 zum Operanden a addiert.
2. Im zweiten Rechenschritt wird der Operand c vom Ergebnis des vorangegangenen Rechenschrittes subtrahiert (Zwischenergebnis = $1.000000*10^8$ - c = $1.000000*10^8$ ergibt $0.000000e^0$).
3. Wenn Sie nun im nächsten Rechenschritt den Operanden z berechnen, dann versuchen Sie durch Null zu dividieren.

	#z	16#7F800000
	#y	0.000000E+000

1. Möglichkeit zur Abhilfe

Um den oben genannten Fall zu umgehen, können Sie einfach Ihre Rechenformel umstellen. Schreiben Sie stattdessen die Formel folgendermaßen:

Rechenformeln

```
y = a - c + b
Z = 1/y
```

Da in diesem Fall nach dem ersten Rechenschritt (Operand a - c) das Ergebnis $0.000000e^0$ vorliegt, führt im zweiten Schritt der Rechnung (Zwischenergebnis + b) die Addition der REAL-Zahl zum richtigen Ergebnis (y = $0.000000*10^0$ + $1.000000*10^0$ = $1.000000*10^0$).

	#y	1.000000E+000
	"DB_GlobalData".a	1.000000E+008
	"DB_GlobalData".c	1.000000E+008
	"DB_GlobalData".b	1.000000E+000
	#z	1.000000E+000
	#y	1.000000E+000

Es ist empfehlenswert, zu überprüfen, wie die Berechnung am sinnvollsten durchzuführen ist, bevor Sie eine Berechnung programmieren.

2. Möglichkeit zur Abhilfe

Verwenden Sie zur Berechnung der oben genannten Formeln statt des Datentyps REAL den Datentyp LREAL. Da der Datentyp mit einer Genauigkeit von 15 Nachkommastellen verarbeitet wird, kommt es erst gar nicht zu diesem Problem.

1. Legen Sie im globalen Datenbaustein "DB_GlobalData" drei neue Variablen mit den gleichen Werten und jeweils mit dem Datentyp LREAL an.

DB_GlobalData

	Name	Data type	Start value
1	Static		
2	a	Real	1.0E+8
3	a_LREAL	LReal	100000000.0
4	b	Real	1.0
5	b_LREAL	LReal	1.0
6	c	Real	1.0E+8
7	c_LREAL	LReal	100000000.0

2. Deklarieren Sie in der Bausteinschnittstelle des FC "FC_Calculate" ebenfalls zwei neue Variablen mit dem Datentyp LREAL.

FC_Calculate

	Name	Data type
1	Input	
2	Output	
3	InOut	
4	Temp	
5	z_LREAL	LReal
6	z	Real
7	y_LREAL	LReal
8	y	Real
9	Constant	
10	Return	
11	FC_Calculate	Void

3. Verwenden Sie für die Formeln in Ihrem Programmcode die neuen LREAL-Variablen und verbinden Sie sich Online um das Ergebnis zu sehen.

SCL

```
#y_LREAL := "DB_GlobalData".a_LREAL + "DB_GlobalData".b_LREAL -
"DB_GlobalData".c_LREAL;
#z_LREAL := 1/#y_LREAL;
```

▼	#y_LREAL	1.00000000000000E+000
	"DB_GlobalData".a_LREAL	1.00000000000000E+008
	"DB_GlobalData".b_LREAL	1.00000000000000E+000
	"DB_GlobalData".c_LREAL	1.00000000000000E+008
▼	#z_LREAL	1.00000000000000E+000
	#y_LREAL	1.00000000000000E+000

Siehe auch

REAL (Seite 264)

LREAL (Seite 265)

Ungültige Gleitpunktzahlen (Seite 266)

16.1.9 Mit Konstanten in SCL rechnen

Interpretation von typisierten und untypisierten Konstanten

Konstanten sind Daten, die einen festen Wert besitzen, der sich zur Programmlaufzeit nicht ändern kann. Konstanten können während der Programmbearbeitung von verschiedenen Programmelementen ausgelesen, aber nicht überschrieben werden. Für den Wert einer Konstante gibt es je nach Datentyp und Datenformat definierte Schreibweisen. Grundsätzlich wird zwischen der typisierten und der untypisierten Schreibweise unterschieden.

Innerhalb von mathematischen Funktionen empfehlen wir, typisierte und untypisierte Konstanten nicht zu mischen, da es in diesem Fall zu ungewollten impliziten Konvertierungen und damit zu falschen Werten kommen kann.

Programmierbeispiel

Im folgenden Programmierbeispiel sehen Sie eine Rechenoperation mit einer typisierten und einer untypisierten Konstanten.

1. Legen Sie einen SCL-Funktionsbaustein an und benennen ihn "FB_MathsFunctions".
2. Deklarieren Sie in der Bausteinschnittstelle im Abschnitt "Temp" die Variable "Variable_DINT".

FB_MathsFunctions		
	Name	Data type
1	Input	
2	Output	
3	InOut	
4	Static	
5	Temp	
6	Variable_DINT	DInt
7	Constant	

3. Schreiben Sie folgenden Programmcode:
   Variable_DINT := INT#1 +50000;
   In dieser Rechenoperation soll die typisierte Konstante INT#1 und die untypisierte Konstante 50000 addiert werden. Die untypisierte Konstante 50000 wird in der Software gelb unterringelt um Ihnen zu signalisieren, dass der Konstantenwert außerhalb des zulässigen Bereichs des Datentyps INT liegt.

```
1
2  #Variable_DINT := INT#1 + 50000;
```

Gehen Sie Online um das Ergebnis zu sehen.

1. Übersetzen Sie den SCL-Funktionsbaustein "FB_MathsFunctions", indem Sie mit einem rechten Mausklick den Befehl "Übersetzen > Software (nur Änderungen)" ausführen.
2. Laden Sie den Baustein mit dem Befehl "Laden in Gerät > Software (nur Änderungen)".
3. Verbinden Sie sich online und beobachten Sie Ihren Baustein.

#Variable_DINT	-15535

Der Datentyp der typisierten Konstante legt den Datentyp der Addition fest. Das bedeutet, dass die Addition im Datentypbereich INT ausgeführt wird. Im ersten Schritt wird die untypisierte Konstante 50000 implizit in den Datentyp INT konvertiert. Die Konvertierung führt allerdings zu einem negativen Wert (-15536). Dieser Wert wird anschließend zur typisierten Konstante (INT#1) addiert. Das Ergebnis lautet -15535. Da die Variable, in die das Ergebnis der Addition geschrieben werden soll, mit dem Datentyp DINT deklariert ist, wird die Zahl -15535 implizit in den Datentyp DINT konvertiert und in die Variable "Variable_DINT" geschrieben. Das Ergebnis bleibt jedoch negativ.

1. Möglichkeit zur Abhilfe

Eine Möglichkeit, dieses unerwünschte Ergebnis zu vermeiden, besteht darin, beide Konstanten zu typisieren. Wenn Sie die beiden Konstanten typisieren, dann bestimmt der längere Datentyp die Rechenoperation.

1. Schreiben Sie im SCL-Funktionsbaustein "FB_MathsFunctions" folgenden Programmcode:

```
3
4  #Variable_DINT := INT#1 + DINT#50000;
```

   In dieser Rechenoperation soll die typisierte Konstante INT#1 und die typisierte Konstante DINT#50000 addiert werden.

Gehen Sie Online um das Ergebnis zu sehen.

1. Übersetzen Sie den SCL-Funktionsbaustein "FB_MathsFunctions", indem Sie mit einem rechten Mausklick den Befehl "Übersetzen > Software (nur Änderungen)" ausführen.
2. Laden Sie den Baustein mit dem Befehl "Laden in Gerät > Software (nur Änderungen)".
3. Verbinden Sie sich online und beobachten Sie Ihren Baustein.

#Variable_DINT	50001

   Die Konstante INT#1 wird in den Datentyp DINT konvertiert und die Addition der beiden Konstanten wird im Datentypbereich DINT ausgeführt.

2. Möglichkeit zur Abhilfe

Eine weitere Möglichkeit, dieses unerwünschte Ergebnis zu vermeiden, besteht darin, beide Konstanten nicht zu typisieren. Wenn Sie die beiden Konstanten nicht typisieren, dann werden diese als der auf der aktuellen CPU breitestmögliche Datentyp interpretiert. Das bedeutet, dass auf einer CPU der Baureihe S7-1500 die beiden Konstanten als Datentyp LINT interpretiert werden.

1. Schreiben Sie im SCL-Funktionsbaustein "FB_MathsFunctions" folgenden Programmcode:

```
1
2  #Variable_DINT := 1 + 50000;
```

   In dieser Rechenoperation soll die untypisierte Konstante 1 und die untypisierte Konstante 50000 addiert werden.

Gehen Sie Online um das Ergebnis zu sehen.

1. Übersetzen Sie den SCL-Funktionsbaustein "FB_MathsFunctions", indem Sie mit einem rechten Mausklick den Befehl "Übersetzen > Software (nur Änderungen)" ausführen.
2. Laden Sie den Baustein mit dem Befehl "Laden in Gerät > Software (nur Änderungen)".
3. Verbinden Sie sich online und beobachten Sie Ihren Baustein.

	#Variable_DINT	50001

Die Konstanten 1 und 50000 werden als Datentyp LINT interpretiert und das Ergebnis der Addition wird wieder in den Datentyp DINT konvertiert.

Siehe auch

Grundlagen zu Konstanten (Seite 102)

Übersicht über die gültigen Datentypen (Seite 243)

16.2 Einstellungen zu SCL

16.2.1 Übersicht über die Einstellungen zu SCL

Übersicht

Die folgenden Tabellen zeigen die Einstellungen, die Sie für SCL vornehmen können:

Editoreinstellungen

Gruppe	Einstellung	Beschreibung
Ansicht	Schlüsselworthervorhebung	Schreibweise, mit der Schlüsselwörter der Programmiersprache dargestellt werden. Sie können zwischen Groß- und Kleinschreibung oder einer den Konventionen der Programmiersprache Pascal entsprechenden Schreibweise wählen.
	Aktualparameter linksbündig ausrichten	Richtet die Aktualparameter bei einem Bausteinaufruf linksbündig aus. Wirkt sich nur aus, wenn bei den Einstellungen unter "Allgemein > Skript-/Texteditoren > Einrückung" die Option "Smart" aktiviert ist.

Voreinstellung für neue Bausteine

Wenn Sie Bausteine neu anlegen, werden die folgenden Einstellungen als Standardwerte gesetzt. Sie können diese zu einem späteren Zeitpunkt innerhalb der Bausteineigenschaften ändern.

Gruppe	Einstellung	Beschreibung
Übersetzen	Erweiterte Statusinformation erstellen	Ermöglicht das Beobachten aller Variablen in einem Baustein. Der Speicherbedarf des Programms und die Ablaufzeiten erhöhen sich jedoch durch diese Option.
	ARRAY-Grenzen prüfen[1)]	Prüft zu Laufzeit, ob Feldindizes in dem für ein ARRAY deklarierten Bereich liegen. Wenn ein Feldindex den zulässigen Bereich überschreitet, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.
	ENO automatisch setzen	Prüft zur Laufzeit, ob Fehler bei der Bearbeitung bestimmter Anweisungen auftreten. Wenn ein Laufzeitfehler auftritt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.
Bausteinschnittstelle	Ansicht der Bausteinschnittstelle	Legt fest, ob die Bausteinschnittstelle neu angelegter Bausteine in der Tabellenansicht oder der textuellen Ansicht angezeigt wird.

[1)] Für CPUs der Baureihen S7-300/400 gilt: Bei Überschreitung der ARRAY-Grenzen wird der Freigabeausgang ENO auf FALSE gesetzt.

Für CPUs der Baureihen S7-1200/1500 gilt: Bei Überschreitung der ARRAY-Grenzen wird der Freigabeausgang ENO nicht auf FALSE gesetzt. Siehe "ARRAY-Komponenten adressieren (Seite 303)" zu Möglichkeiten der Fehlerabfrage.

Siehe auch

Einstellungen ändern (Seite 7632)

EN-/ENO-Mechanismus (Seite 150)

16.2.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die Einstellungen zu SCL (Seite 7631)

16.3 Das Programmierfenster von SCL

16.3.1 Übersicht über das Programmierfenster

Funktion

Das Programmierfenster ist der Arbeitsbereich, in dem Sie das SCL-Programm eingeben.

Das folgende Bild zeigt das Programmierfenster von SCL:

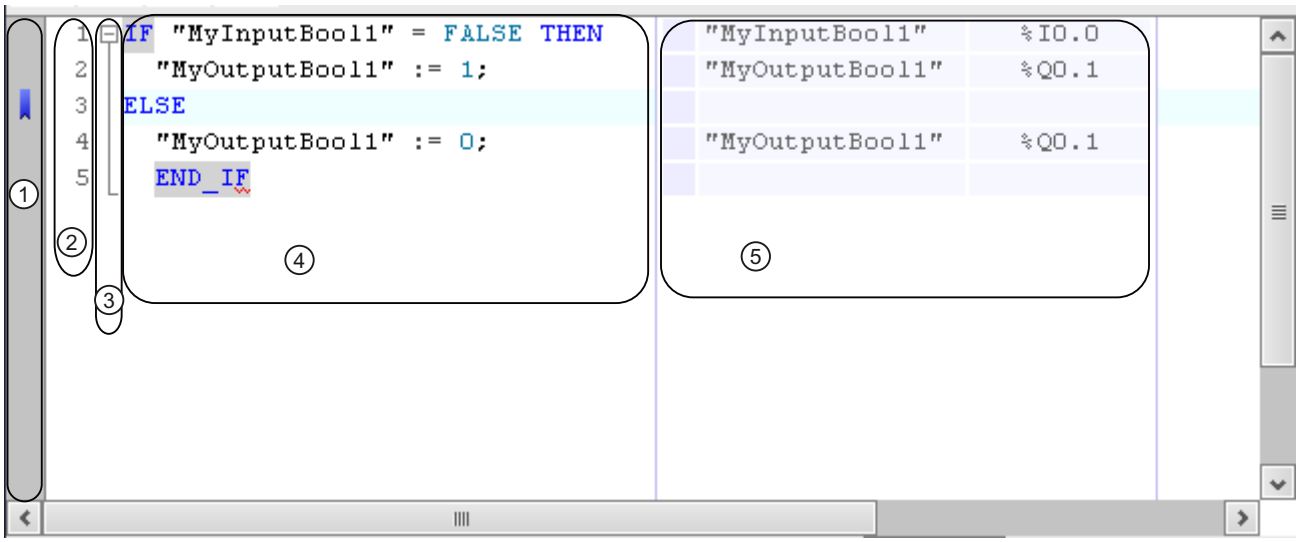

Das Programmierfenster besteht aus folgenden Bereichen:

Bereich	Bedeutung
① Seitenleiste	In der Seitenleiste können Sie Lesezeichen und Haltepunkte setzen.
② Zeilennummern	Links neben dem Programmcode werden die Zeilennummern angezeigt.
③ Gliederungsanzeige	Die Gliederungsanzeige markiert zusammenhängende Codeabschnitte.
④ Codebereich	Im Codebereich bearbeiten Sie das SCL-Programm.
⑤ Anzeige der absoluten Operanden	Diese Tabelle zeigt die Zuordnung der symbolischen Operanden zu absoluten Adressen.

Siehe auch

Anpassen des Programmierfensters (Seite 7634)

SCL-Code formatieren (Seite 7635)

Codeabschnitte auf- und zuklappen (Seite 7637)

Zu Definitionen navigieren (Seite 7637)

Lesezeichen verwenden (Seite 7638)

16.3.2 Anpassen des Programmierfensters

Einführung

Die Darstellung des Programmierfensters und des Programmcodes können Sie auf folgende Weise anpassen:

- Einstellen der Schriftart, -größe und -farbe
- Einstellen der Tabulatorweite
- Anzeige der Zeilennummern
- Ein- oder Ausblenden der absoluten Operanden

Schriftart, -größe und -farbe einstellen

Um Schriftart, -größe und -farbe einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Stellen Sie die gewünschte Schriftart und -größe ein oder wählen Sie Schriftfarben für die einzelnen Sprachelemente aus.

Tabulatorweite einstellen

Um das Programm übersichtlicher zu gestalten, werden Zeilen syntaxgerecht eingerückt. Die Breite der Einrückung definieren Sie über die Tabulatorweite.

Um die Tabulatorweite einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Stellen Sie die Tabulatorweite ein.

Zeilennummern anzeigen

Um die Zeilennummern anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Aktivieren Sie die Option "Zeilennummern anzeigen".

Ein- und Ausblenden der absoluten Operanden

Die Zuordnung von symbolischen und absoluten Operanden können Sie bei Bedarf in einer Tabelle neben dem Programmcode einblenden.

Um die Anzeige der absoluten Operanden ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Absolute/symbolische Operanden". Die Anzeige der absoluten Operanden wird eingeblendet.
2. Um die Anzeige zu verschieben, klicken Sie auf die Tabelle und ziehen sie bei gedrückter Maustaste an die gewünschte Position.
3. Um die Breite der Tabelle zu verändern, klicken Sie auf den rechten oder linken Tabellenrand und ziehen Sie ihn bei gedrückter Maustaste nach rechts oder links.

Siehe auch

Übersicht über das Programmierfenster (Seite 7633)

SCL-Code formatieren (Seite 7635)

Codeabschnitte auf- und zuklappen (Seite 7637)

Zu Definitionen navigieren (Seite 7637)

Lesezeichen verwenden (Seite 7638)

16.3.3 SCL-Code formatieren

Einführung

Um das Programm übersichtlicher zu gestalten, können Sie einzelne Zeilen manuell ein- oder ausrücken oder Codeabschnitte formatieren. Beachten Sie beim Formatieren von Codeabschnitten die folgenden Hinweise:

- Die Art der Formatierung richtet sich nach den allgemeinen Einstellungen für Einrückungen, jedoch wird immer mindestens die Zeile bzw. der Abschnitt eingerückt. Haben Sie die Einstellung "Smart" gewählt, werden auch unnötige Leerzeichen innerhalb der SCL-Anweisung entfernt.
- Nur syntaktisch korrekte Codeabschnitte können formatiert werden.
- Wenn Sie die Einfügemarke in die erste oder letzte Zeile einer Anweisung für die Programmsteuerung setzen, z. B. bei einer IF-Anweisung in die Zeile mit dem "IF", wird die gesamte Anweisung formatiert.
- Wenn Sie Text selektieren, wird nur der selektierte Text formatiert.

Zusätzlich können Sie einstellen, dass die Aktualparameter von Bausteinaufrufen generell linksbündig ausgerichtet werden sollen. Dazu muss als Option für Einrückungen "Smart" eingestellt sein.

Einzelne Zeilen ein- oder ausrücken

Um einzelne Zeilen ein- oder auszurücken, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Zeile, die Sie ein- oder ausrücken möchten.
2. Wählen Sie in der Funktionsleiste des Programmiereditors die Schaltfläche "Text einrücken" oder "Text ausrücken".

Hinweis

Die Breite der Einrückung können Sie über "Extras > Einstellungen" definieren.

Codeabschnitte formatieren

Um Codeabschnitte zu formatieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Text, den Sie formatieren möchten, oder setzen Sie die Einfügemarke in die entsprechende Zeile.
2. Wählen Sie in der Funktionsleiste des Programmiereditors die Schaltfläche "Selektierten Text automatisch formatieren".

Aktualparameter von Bausteinaufrufen linksbündig ausrichten

Um die Aktualparameter von Bausteinaufrufen linksbündig ausrichten zu lassen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein > Skript-/Texteditoren".
3. Aktivieren Sie unter "Einrückung" die Option "Smart".
4. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > SCL (Structured Control Language)".
5. Aktivieren Sie unter "Ansicht" das Optionskästchen "Aktualparameter linksbündig ausrichten".
   Für neu eingefügte Bausteinaufrufe werden die Aktualparameter linksbündig ausgerichtet. Um die Aktualparameter von vorhandenen Bausteinaufrufen linksbündig auszurichten, können Sie die Bausteinaufrufe selektieren und in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Selektierten Text automatisch formatieren" klicken.

Siehe auch

Übersicht über das Programmierfenster (Seite 7633)

Anpassen des Programmierfensters (Seite 7634)

Codeabschnitte auf- und zuklappen (Seite 7637)

Zu Definitionen navigieren (Seite 7637)

Lesezeichen verwenden (Seite 7638)

16.3.4 Codeabschnitte auf- und zuklappen

Einführung

SCL-Anweisungen können sich über mehrere Zeilen erstrecken. Beispiele hierfür sind Programmsteuerungsanweisungen oder Bausteinaufrufe.

Solche zusammengehörende Anweisungen werden wie folgt gekennzeichnet:

- Eine Gliederungsanzeige zwischen der Zeilennummer und dem Programmcode markiert den gesamten Codeabschnitt.
- Wenn Sie das öffnende Schlüsselwort markieren, wird automatisch auch das schließende Schlüsselwort markiert.

Um die Übersichtlichkeit zu verbessern, können Sie in der Gliederungsanzeige zusammengehörende Codeabschnitte auf- oder zuklappen. Die gewählte Gliederungsansicht bleibt beim Schließen des Bausteins oder des Projekts erhalten, sodass die Codeabschnitte beim nächsten Öffnen des Bausteins wieder genauso angezeigt werden wie beim Schließen.

Vorgehen

Um Codeabschnitte auf- und zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Minus-Zeichen in der Gliederungsanzeige.
   Der Codeabschnitt wird zugeklappt.
2. Klicken Sie auf das Plus-Zeichen in der Gliederungsanzeige.
   Der Codeabschnitt wird wieder aufgeklappt.

Siehe auch

Übersicht über das Programmierfenster (Seite 7633)

Anpassen des Programmierfensters (Seite 7634)

SCL-Code formatieren (Seite 7635)

Zu Definitionen navigieren (Seite 7637)

Lesezeichen verwenden (Seite 7638)

16.3.5 Zu Definitionen navigieren

Sie können in Ihrem Programmcode Elemente verwenden, die in anderen Editoren definiert wurden, z. B. Variablen oder Datenbausteine (DBs) von aufgerufenen Funktionsbausteinen (FBs). Um sich die Definitionsstellen in den entsprechenden Editoren anzusehen, können Sie gezielt zu diesen Stellen navigieren.

Vorgehen

Um zu der Definition eines Code-Elements zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Code-Element.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Definition".
   Der Editor, in dem das Code-Element definiert wurde, wird geöffnet und die Definitionsstelle wird angezeigt.

Oder:

1. Drücken und halten Sie die Taste <Strg>.
2. Bewegen Sie den Mauszeiger über Ihren Programmcode.
   Bewegt sich der Mauszeiger über ein Code-Element, dessen Definition angezeigt werden kann, wird es unterstrichen dargestellt und der Name des Elements wird zu einem Link.
3. Klicken Sie auf den Link.
   Der Editor, in dem das Code-Element definiert wurde, wird geöffnet und die Definitionsstelle wird angezeigt.

Siehe auch

Übersicht über das Programmierfenster (Seite 7633)

Anpassen des Programmierfensters (Seite 7634)

SCL-Code formatieren (Seite 7635)

Codeabschnitte auf- und zuklappen (Seite 7637)

Lesezeichen verwenden (Seite 7638)

16.3.6 Lesezeichen verwenden

16.3.6.1 Grundlagen zu Lesezeichen

Funktion

Mit Lesezeichen können Sie Programmstellen in umfangreichen Programmen markieren, um sie für eine spätere Überarbeitung leichter aufzufinden. Lesezeichen werden in der Seitenleiste des Programmierfensters angezeigt. Sie können über Menübefehle zwischen mehreren Lesezeichen innerhalb eines Bausteins navigieren.

Lesezeichen werden mit dem Projekt gespeichert und stehen somit allen Bearbeitern des Bausteins zur Verfügung. Sie werden jedoch nicht mit in ein Gerät geladen.

Beim Bausteinvergleich werden Lesezeichen nicht ausgewertet.

Siehe auch

Lesezeichen setzen (Seite 7639)
Zwischen Lesezeichen navigieren (Seite 7639)
Lesezeichen löschen (Seite 7640)

16.3.6.2 Lesezeichen setzen

Voraussetzung

Der SCL-Baustein ist geöffnet.

Vorgehen

Um ein Lesezeichen zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in der Seitenleiste in die gewünschte Zeile.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Setzen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen setzen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Lesezeichen setzen/löschen".

Oder:

1. Halten Sie die Taste <Strg> gedrückt.
2. Klicken Sie in der Seitenleiste in die Zeile, in der Sie das Lesezeichen setzen möchten.

Ergebnis

Ein Lesezeichen wird in den Programmcode gesetzt.

Siehe auch

Grundlagen zu Lesezeichen (Seite 7638)
Zwischen Lesezeichen navigieren (Seite 7639)
Lesezeichen löschen (Seite 7640)

16.3.6.3 Zwischen Lesezeichen navigieren

Voraussetzung

In einem Baustein sind mehrere Lesezeichen gesetzt.

Vorgehen

Um zwischen Lesezeichen zu navigieren, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke in den Programmcode.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Gehe zu > Nächstem Lesezeichen" oder "Gehe zu > Vorangehendem Lesezeichen".

Oder:

1. Setzen Sie die Einfügemarke in den Programmcode.
2. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Gehe zu nächstem Lesezeichen" oder "Gehe zu vorangehendem Lesezeichen".

Oder:

1. Klicken Sie in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Nächstes" oder "Lesezeichen > Vorangehendes".

Ergebnis

Die Zeile mit dem Lesezeichen wird hervorgehoben.

Siehe auch

Grundlagen zu Lesezeichen (Seite 7638)

Lesezeichen setzen (Seite 7639)

Lesezeichen löschen (Seite 7640)

16.3.6.4 Lesezeichen löschen

Sie können einzelne Lesezeichen oder alle Lesezeichen aus dem Baustein bzw. der CPU löschen.

Einzelne Lesezeichen löschen

Um ein einzelnes Lesezeichen zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Zeile, in der Sie das Lesezeichen löschen möchten, mit der rechten Maustaste in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Löschen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen löschen möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Lesezeichen > Löschen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen löschen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Lesezeichen setzen/löschen".

Alle Lesezeichen aus dem Baustein löschen

Um alle Lesezeichen aus dem Baustein zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Alle aus Baustein löschen".

Oder:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Lesezeichen > Alle aus Baustein löschen".

Siehe auch

Grundlagen zu Lesezeichen (Seite 7638)
Lesezeichen setzen (Seite 7639)
Zwischen Lesezeichen navigieren (Seite 7639)

16.4 SCL-Anweisungen eingeben

16.4.1 Regeln zu SCL-Anweisungen

Anweisungen in SCL

SCL kennt folgende Arten von Anweisungen:

- Wertzuweisungen
  Wertzuweisungen dienen dazu, einer Variablen einen konstanten Wert, das Ergebnis eines Ausdrucks, oder den Wert einer anderen Variablen zuzuweisen.
- Anweisungen zur Programmsteuerung
  Anweisungen zur Programmsteuerung dienen dazu, Programmverzweigungen, -schleifen oder -sprünge zu realisieren.
- Weitere Anweisungen aus der Task Card "Anweisungen"
  Die Task Card "Anweisungen" bietet eine breite Auswahl von Standardanweisungen, die Sie in Ihrem SCL-Programm verwenden können.
- Bausteinaufrufe
  Bausteinaufrufe dienen dazu, Unterprogramme, die in andere Bausteine ausgelagert sind, zu bearbeiten und deren Ergebnis weiter zu verarbeiten.

Regeln

Bei der Eingabe von SCL-Anweisungen müssen Sie folgende Regeln beachten:

- Anweisungen können sich über mehrere Zeilen erstrecken.
- Jede Anweisung endet mit einem Semikolon (;).
- Groß- und Kleinschreibung wird nicht unterschieden.
- Kommentare dienen lediglich der Dokumentation des Programms. Sie beeinträchtigen den Programmablauf nicht.

Beispiele

Die folgenden Beispiele zeigen die verschiedenen Anweisungsarten:

SCL

```
// Beispiel für eine Wertzuweisung
"MyTag":= 0;
// Beispiel für einen Bausteinaufruf
"MyDB"();
// Beispiel für eine Anweisung zur Programmsteuerung
WHILE "Counter" < 10 DO
    "MyTAG" := "MyTag" + 2;
END_WHILE;
```

Siehe auch

Grundlagen zu SCL (Seite 7607)

16.4.2 SCL-Anweisungen manuell eingeben

Voraussetzung

Ein SCL-Baustein ist geöffnet.

Vorgehen

Um SCL-Anweisungen einzugeben, gehen Sie folgendermaßen vor:

1. Geben Sie die Syntax der Anweisung über die Tastatur ein.
   Bei der Eingabe werden Sie durch die Autovervollständigung unterstützt. Sie bietet alle Anweisungen und Operanden an, die an der aktuellen Position zulässig sind.
2. Wählen Sie aus der Autovervollständigung die gewünschte Anweisung oder den gewünschten Operanden aus.
   Wenn Sie eine Anweisung auswählen, die die Angabe von Operanden erfordert, werden Platzhalter für die Operanden ins Programm eingefügt. Die Platzhalter für die Operanden sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
3. Ersetzen Sie den Platzhalter durch einen Operanden.
4. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Operanden.

Hinweis

Sie können einen bereits definierten Operanden auch per Drag & Drop aus der PLC-Variablentabelle oder der Bausteinschnittstelle in das Programm ziehen. Um dabei einen bereits eingefügten Operanden zu ersetzen, lassen Sie den Mauszeiger vor dem Loslassen der Maustaste kurz über dem zu ersetzenden Operanden stehen. Dadurch wird der Operand selektiert und beim Loslassen der Maustaste durch den neuen Operanden ersetzt.

Ergebnis

Die Anweisung wird eingefügt.

Der Programmiereditor führt eine Syntaxprüfung durch. Dabei werden fehlerhafte Eingaben rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Inspektorfenster.

Siehe auch

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

Parameterliste erweitern und reduzieren (Seite 7670)

Programmcode mit Strukturelementen umschließen (Seite 7645)

Datentypkonvertierungen bei S7-1200 (Seite 507)

16.4.3 SCL-Anweisungen über Task Card "Anweisungen" einfügen

Die Task Card "Anweisungen" bietet eine breite Auswahl von Anweisungen, die Sie in Ihrem SCL-Programm verwenden können. Auch die SCL-spezifischen Anweisungen zur Programmsteuerung sind über die Task Card "Anweisungen" verfügbar.

Voraussetzung

Ein SCL-Baustein ist geöffnet.

Vorgehen

Um SCL-Anweisungen mithilfe der Task Card "Anweisungen" in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Zum Einfügen der Anweisung wählen Sie einen der folgenden Schritte:
   - Navigieren Sie zu der SCL-Anweisung, die Sie einfügen möchten, und ziehen Sie sie per Drag & Drop an die gewünschte Zeile im Programmcode. Die Einfügestelle wird durch ein grünes Rechteck gekennzeichnet.
   - Markieren Sie die Stelle im Programmcode, an die Sie die Anweisung einfügen möchten, und doppelklicken Sie dann auf die einzufügende Anweisung.

   Die Anweisung wird ins Programm eingefügt. Die Platzhalter für die Operanden sind gelb hinterlegt. Ein helleres Gelb kennzeichnet dabei die optionalen Parameter, die Sie nicht unbedingt verschalten müssen. Ein dunkleres Gelb kennzeichnet die Pflichtparameter, die Sie verschalten müssen. Der erste Platzhalter ist selektiert.
3. Ersetzen Sie den Platzhalter durch einen Operanden. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablentabelle per Drag & Drop auf den Platzhalter ziehen.
4. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Operanden.

Ergebnis

Die Anweisung wird eingefügt.

Der Programmiereditor führt eine Syntaxprüfung durch. Dabei werden fehlerhafte Eingaben rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Inspektorfenster.

Siehe auch

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

Parameterliste erweitern und reduzieren (Seite 7670)

Programmcode mit Strukturelementen umschließen (Seite 7645)

Datentypkonvertierungen bei S7-1200 (Seite 507)

16.4.4 Programmcode mit Strukturelementen umschließen

Sie haben die Möglichkeit, Ihren Programmcode mit den folgenden Strukturelementen zu umschließen:

- Bereiche:
  - REGION
- Kontrollstrukturen
  - IF ... THEN
  - CASE ... OF ...
  - FOR ... TO ... DO ...
  - WHILE ... DO ...
- Kommentarabschnitte
  - (* *)

Vorgehen

Um Ihren Programmcode mit einem Strukturelement zu umschließen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Programmcode, den Sie mit einem Strukturelement umschließen möchten.
2. Klicken Sie mit der rechten Maustaste auf den selektierten Programmcode.
3. Wählen Sie im Kontextmenü unter "Umschließen mit" das gewünschte Strukturelement aus.
4. Alternativ können Sie das Strukturelement auch in den Favoriten oder in der Task Card "Anweisungen" in der Palette "Einfache Anweisungen > Programmsteuerung" auswählen.

Siehe auch

SCL-Anweisungen manuell eingeben (Seite 7643)

SCL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7644)

16.4.5 Arbeiten mit Bereichen

16.4.5.1 Bereiche verwenden

Funktion

In SCL können Sie Ihren Programmcode in Bereiche strukturieren. Dies verbessert die Übersichtlichkeit Ihres Programms, da Sie zusammengehörende Programmteile zusammenfassen und die Bereiche je nach Bedarf auf- und zuklappen können. Für eine schnelle Navigation in den Bereichen steht Ihnen eine Bereichsübersicht zur Verfügung. Die Bereichsübersicht wird neben dem Programmierfenster angezeigt und kann bei Bedarf ein- oder ausgeblendet werden.

Das folgende Bild zeigt die Bereichsübersicht und das Programmierfenster:

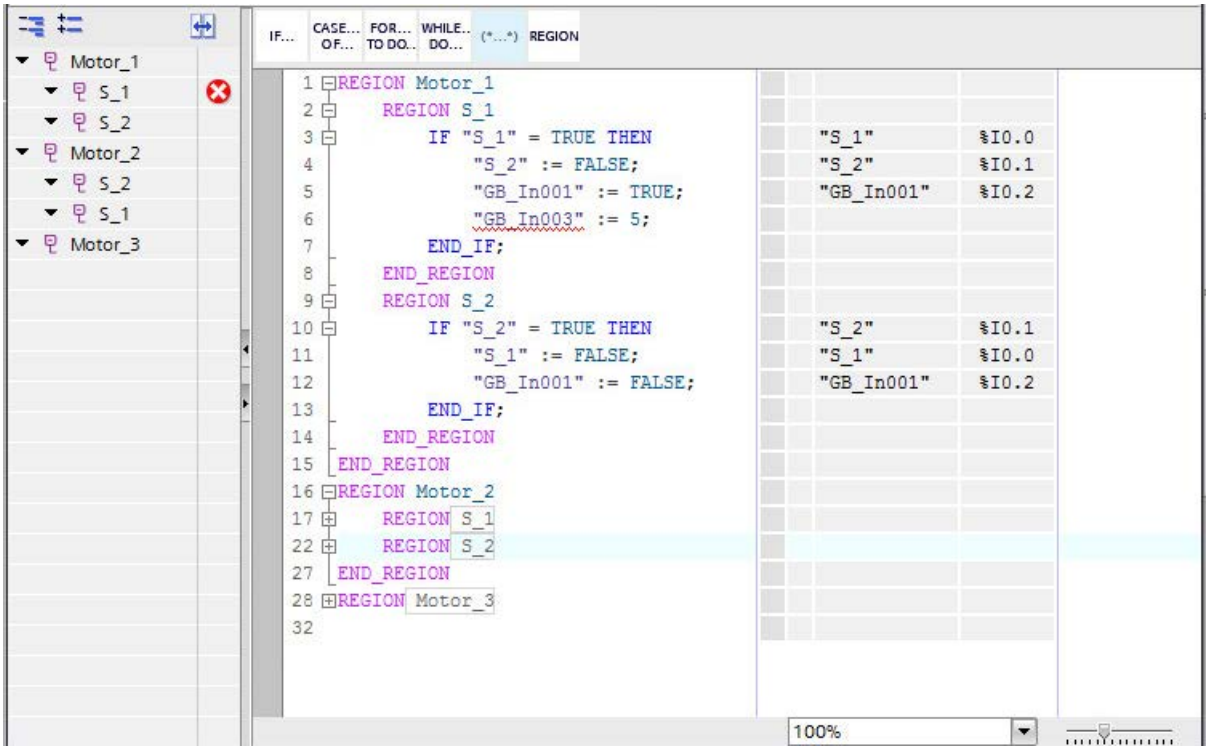

Die Bereichsübersicht zeigt zusätzlich, in welchen Bereichen noch Syntaxfehler vorhanden sind. Dadurch können Sie schnell zu fehlerhaften Codestellen navigieren und die Fehler beheben.

Um einen Programmteil in einen Bereich zusammenzufassen, umschließen Sie ihn mit den Schlüsselwörtern "REGION" und "END_REGION". Abhängig von Ihren Einstellungen bei der Schlüsselworthervorhebung kann die Schreibweise der Schlüsselwörter auch "region" und "end_region" oder "Region" und "End_Region" sein. Die Schlüsselwörter werden automatisch in die Schreibweise umgewandelt, die der aktuellen Einstellung entspricht. Um zusammengehörige "REGION"- und "END_REGION"-Schlüsselwörter zu finden, klicken Sie entweder auf "REGION" oder "END_REGION". Beide Schlüsselwörter werden dadurch farblich hervorgehoben.

Sie können jedem Bereich einen Namen geben. Dies hat die folgende Vorteile:

- Der Name ist auch im zugeklappten Zustand eines Bereichs sichtbar. Dadurch behalten Sie jederzeit den Überblick über Ihren Programmcode.
- Sie können die Bereiche in der Bereichsübersicht leichter identifizieren und zu einzelnen Bereichen navigieren.

Die Angabe eines Namens ist optional.

Syntax

Verwenden Sie die folgende Syntax, um Programmteile in Bereiche zusammenzufassen:

```
REGION <Name>
     <Anweisungen>
END_REGION
```

Beachten Sie bei der Verwendung von Bereichen die folgenden Hinweise:

- Die Schlüsselwörter "REGION" und "END_REGION" müssen dabei an Anfang einer Zeile stehen. Lediglich Leerzeichen sind vor den Schlüsselwörtern zulässig.
- Alle Zeichen nach dem Schlüsselwort "END_REGION" werden als Kommentar betrachtet und haben daher keine Auswirkungen auf den Programmablauf.
- Alle Anweisungen müssen innerhalb eines Bereichs abgeschlossen werden.
- Innerhalb von CASE-Anweisungen können die Anweisungen mit Bereichen umschlossen werden. Die Konstanten, welche für den Vergleich der CASE-Anweisung genutzt werden, müssen jedoch außerhalb der Bereiche stehen. Die Bereiche dürfen nicht mehrere Zweige umspannen.
  Das folgende Beispiel zeigt eine erlaubte Verwendung von Bereichen innerhalb einer CASE-Anweisung:
  ```
  CASE <Expression> OF
    <Constant1> : REGION <Instructions1> END_REGION
    <Constant2> : REGION <Instructions2> END_REGION
  ELSE REGION <Instructions0> END_REGION
  END_CASE
  ```
  Die folgenden Beispiele zeigen nicht erlaubte Verwendungen von Bereichen innerhalb von CASE-Anweisungen:
  ```
  CASE <Expression> OF
    REGION <Constant1> : <Instructions1> END_REGION
    <Constant2> : REGION <Instructions2> END_REGION
  ELSE <Instructions0>
  END_CASE
  CASE <Expression> OF
    <Constant1> : <Instructions1> REGION
    <Constant2> : END_REGION <Instructions2>
  ELSE <Instructions0>
  END_CASE
  ```
  In diesen Fälle werden die Konstanten innerhalb eines Bereiches als Sprungmarken interpretiert, die mit GOTO erreicht werden können, die aber nicht von den CASE-Anweisungen berücksichtigt werden.

Sie haben die Möglichkeit, Bereiche auch zu verschachteln. Achten Sie jedoch darauf, dass Bereiche innerhalb anderer Bereiche vollständig abgeschlossen sind:

```
REGION <Name>
     <Anweisungen>
     REGION <Name>
           <Anweisungen>
     END_REGION
     <Anweisungen>
     REGION <Name>
          REGION
               <Anweisungen>
          END_REGION
          REGION <Name>
               <Anweisungen>
          END_REGION
     END_REGION
END_REGION
```

Siehe auch

Bereiche einfügen (Seite 7648)

Bereiche kopieren und einfügen (Seite 7650)

In Bereichen navigieren (Seite 7651)

Bereiche löschen (Seite 7653)

16.4.5.2 Bereiche einfügen

Sie haben die folgenden Möglichkeiten, Bereiche in Ihren Programmcode einzufügen:

- Sie können Bereiche manuell über die Tastatur einfügen.
- Sie können Bereiche über die Task Card "Anweisungen" einfügen. Dabei können Sie entweder leere Bereiche einfügen oder bestehenden Programmcode mit Bereichen umschließen.
- Sie können Bereiche über die Favoriten einfügen. Dabei können Sie entweder leere Bereiche einfügen oder bestehenden Programmcode mit Bereichen umschließen.

Bereiche manuell über die Tastatur einfügen

Um in Ihren Programmcode einen Bereich manuell über die Tastatur einzufügen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke an die Position, an der Sie einen Bereich einfügen möchten.
2. Geben Sie das Schlüsselwort "REGION" ein.

   Hinweis

   Abhängig von Ihren Einstellungen bei der Schlüsselworthervorhebung kann die Schreibweise der Schlüsselwörter auch "region" und "end_region" oder "Region" und "End_Region" sein. Die Schlüsselwörter werden automatisch in die Schreibweise umgewandelt, die der aktuellen Einstellung entspricht.

3. Geben Sie einen Namen für den Bereich ein. Dieser Schritt ist optional.
4. Setzen Sie die Einfügemarke an das Ende des Bereichs.
5. Geben Sie das Schlüsselwort "END_REGION" ein.
   Der entsprechende Programmcode wird in einem Bereich zusammengefasst, den Sie nun auf- und zuklappen können. Der neue Bereich ist in der Bereichsübersicht sichtbar. Falls Sie keinen Namen vergeben haben, wird er in der Bereichsübersicht mit dem Namen "Ohne Namen" angezeigt.

Bereiche über die Task Card "Anweisungen" einfügen

Um in Ihren Programmcode einen Bereich über die Task Card "Anweisungen" einzufügen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke an die Position, an der Sie einen leeren Bereich einfügen möchten, oder selektieren Sie den Programmcode, den Sie mit einem Bereich umschließen möchten.
2. Öffnen Sie die Task Card "Anweisungen".
3. Navigieren Sie zu "Programmsteuerung > REGION".
4. Ziehen Sie die Anweisung "REGION" per Drag & Drop an die Position in Ihrem Programmcode, an der Sie einen Bereich einfügen möchten, oder fügen Sie die Anweisung "REGION" per Doppelklick ein.
   Es wird entweder ein leerer Bereich eingefügt oder der selektierte Programmcode wird mit einem Bereich umschlossen. Der neue Bereich ist in der Bereichsübersicht sichtbar, dabei wird der Platzhalter "_name_" als Name verwendet.
5. Ersetzen Sie den Platzhalter "_name_" durch den gewünschten Namen oder löschen Sie den Platzhalter, falls Sie keinen Namen vergeben möchten. Falls Sie keinen Namen vergeben haben, wird der neue Bereich in der Bereichsübersicht mit dem Namen "Ohne Namen" angezeigt.

Bereiche über die Favoriten einfügen

Um in Ihren Programmcode einen Bereich über die Favoriten einzufügen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke an die Position, an der Sie einen leeren Bereich einfügen möchten, oder selektieren Sie den Programmcode, den Sie mit einem Bereich umschließen möchten.
2. Klicken Sie in der Palette "Favoriten" oder in der Favoritenleiste im Programmiereditor auf die Anweisung "REGION".
   Es wird entweder ein leerer Bereich eingefügt oder der selektierte Programmcode wird mit einem Bereich umschlossen. Der neue Bereich ist in der Bereichsübersicht sichtbar, dabei wird der Platzhalter "_name_" als Name verwendet.
3. Ersetzen Sie den Platzhalter "_name_" durch den gewünschten Namen oder löschen Sie den Platzhalter, falls Sie keinen Namen vergeben möchten. Falls Sie keinen Namen vergeben haben, wird der neue Bereich in der Bereichsübersicht mit dem Namen "Ohne Namen" angezeigt.

Siehe auch

Bereiche verwenden (Seite 7646)

Bereiche kopieren und einfügen (Seite 7650)

In Bereichen navigieren (Seite 7651)

Bereiche löschen (Seite 7653)

16.4.5.3 Bereiche kopieren und einfügen

Sie können Bereiche und ihre Inhalte auch kopieren und wieder in Ihren Programmcode einfügen.

Vorgehen

Um Bereiche und ihre Inhalte zu kopieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bereichsübersicht.
2. Klicken Sie mit der rechten Maustaste auf den Bereich, den Sie kopieren möchten, und wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Setzen Sie die Einfügemarke an die Position in Ihrem Programmcode, an der Sie den Bereich einfügen möchten.
4. Fügen Sie den Bereich entweder über den Kontextmenübefehl "Einfügen" oder über die Tastenkombination <Strg+V> in Ihren Programmcode ein.
   Der kopierte Bereich wird zusammen mit seinem Inhalt aufgeklappt in Ihren Programmcode eingefügt. Zusätzlich wird der Bereich in die Bereichsübersicht eingefügt.
5. Vergeben Sie für den eingefügten Bereich bei Bedarf einen neuen Namen.

Oder:

1. Selektieren Sie den Bereich im Programmfenster. Der Bereich kann dabei auf- oder zugeklappt sein.
2. Kopieren Sie den Bereich entweder über den Kontextmenübefehl "Kopieren" oder über die Tastenkombination <Strg+C>.
3. Setzen Sie die Einfügemarke an die Position in Ihrem Programmcode, an der Sie den Bereich einfügen möchten.
4. Fügen Sie den Bereich entweder über den Kontextmenübefehl "Einfügen" oder über die Tastenkombination <Strg+V> in Ihren Programmcode ein.
   Der kopierte Bereich wird zusammen mit seinem Inhalt aufgeklappt in Ihren Programmcode eingefügt. Zusätzlich wird der Bereich in die Bereichsübersicht eingefügt.
5. Vergeben Sie für den eingefügten Bereich bei Bedarf einen neuen Namen.

Siehe auch

Bereiche verwenden (Seite 7646)

Bereiche einfügen (Seite 7648)

In Bereichen navigieren (Seite 7651)

Bereiche löschen (Seite 7653)

16.4.5.4 In Bereichen navigieren

Zum schnellen Navigieren in den Bereichen steht Ihnen die Bereichsübersicht zur Verfügung. Die Bereichsübersicht zeigt Ihnen alle Bereiche an, die in Ihrem Programmcode vorhanden sind. Zusätzlich bietet Sie Ihnen die folgenden Funktionen:

- Anzeige der Bereiche in der Bereichsübersicht und im Programmierfenster synchronisieren:
  Sie können festlegen, ob sich das Auf- und Zuklappen von Bereichen entweder nur auf die Bereichsübersicht bzw. das Programmierfenster oder auf beides auswirkt.
- Einzelnen Bereich auf- oder zuklappen:
  Sie können einzelne Bereiche entweder in der Bereichsübersicht oder im Programmierfenster auf- oder zuklappen. Wenn Sie die Synchronisation eingeschaltet haben, wird der Bereich in beiden Fenstern auf- oder zugeklappt.
- Alle Bereiche auf- oder zuklappen
  Sie können alle Bereiche entweder in der Bereichsübersicht oder im Programmierfenster auf einmal auf- oder zuklappen. Wenn Sie die Synchronisation eingeschaltet haben, werden die Bereiche in beiden Fenstern auf- oder zugeklappt.

Anzeige der Bereiche in der Bereichsübersicht und im Programmierfenster synchronisieren

Um die Synchronisation der Anzeige für die Bereiche in der Bereichsübersicht und im Programmierfenster festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bereichsübersicht.
2. Um die Synchronisation der Anzeige einzuschalten, klicken Sie in der Bereichsübersicht in der Funktionsleiste auf die Schaltfläche "Synchronisation ein/aus".
   Die Schaltfläche für die Synchronisation wird als Aktiv angezeigt.
3. Um die Synchronisation der Anzeige auszuschalten, klicken Sie in der Bereichsübersicht in der Funktionsleiste auf die Schaltfläche "Synchronisation ein/aus".
   Die Schaltfläche für die Synchronisation wird als Inaktiv angezeigt.

Einzelnen Bereich auf- oder zuklappen

Um einen einzelnen Bereich auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bereichsübersicht.
2. Klicken Sie in der Bereichsübersicht mit der rechten Maustaste auf den Bereich, den Sie auf- oder zuklappen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Aufklappen" oder "Zuklappen".

Oder:

1. Öffnen Sie die Bereichsübersicht.
2. Selektieren Sie in der Bereichsübersicht den Bereich, den Sie auf- oder zuklappen möchten.
3. Drücken Sie die Tastenkombinationen <Strg+Shift+Num+> für Aufklappen oder <Strg+Shift +Num-> für Zuklappen.

Oder:

1. Setzen Sie im Programmierfenster die Einfügemarke in den Bereich, den Sie auf- oder zuklappen möchten.
2. Drücken Sie die Tastenkombinationen <Strg+Shift+Num+> für Aufklappen oder <Strg+Shift +Num-> für Zuklappen.

Abhängig von der Synchronisationseinstellung wird der Bereich entweder in der Bereichsübersicht oder im Programmierfenster oder in beiden Fenstern auf- oder zugeklappt.

Alle Bereiche auf- oder zuklappen

Um alle Bereiche gemeinsam auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bereichsübersicht.
2. Um alle Bereiche aufzuklappen, klicken Sie in der Bereichsübersicht in der Funktionsleiste auf die Schaltfläche "Alle aufklappen".
3. Um alle Bereiche zuzuklappen, klicken Sie in der Bereichsübersicht in der Funktionsleiste auf die Schaltfläche "Alle zuklappen".

Oder:

1. Öffnen Sie die Bereichsübersicht.
2. Selektieren Sie in der Bereichsübersicht einen beliebigen Bereich.
3. Um alle Bereiche aufzuklappen, drücken Sie die Tastenkombination <Strg+Shift+Num*>.
4. Um alle Bereiche zuzuklappen, drücken Sie die Tastenkombination <Strg+Shift+Num/>.

Oder:

1. Setzen Sie die Einfügemarke in das Programmierfenster.
2. Um alle Bereiche aufzuklappen, drücken Sie die Tastenkombination <Strg+Shift+Num*>.
3. Um alle Bereiche zuzuklappen, drücken Sie die Tastenkombination <Strg+Shift+Num/>.

Abhängig von der Synchronisationseinstellung werden entweder alle Bereiche in der Bereichsübersicht oder im Programmierfenster oder in beiden Fenstern auf- oder zugeklappt.

Siehe auch

Bereiche verwenden (Seite 7646)

Bereiche einfügen (Seite 7648)

Bereiche kopieren und einfügen (Seite 7650)

Bereiche löschen (Seite 7653)

16.4.5.5 Bereiche löschen

Sie können Bereiche jederzeit wieder aus Ihrem Programmcode entfernen.

Vorgehen

Um einen Bereich zu löschen, gehen Sie folgendermaßen vor:

1. Entfernen Sie in Ihrem Programmcode das Schlüsselwort "REGION" und den Namen des Bereichs, falls Sie einen Namen vergeben hatten.
2. Entfernen Sie in Ihrem Programmcode das zugehörige Schlüsselwort "END_REGION". Der Bereich wird aus dem Programmcode und der Bereichsübersicht gelöscht.

Hinweis

Sobald Sie entweder das Schlüsselwort "REGION" oder "END_REGION" aus dem Programmcode entfernt haben, erkennen Sie das jeweils zugehörige Schlüsselwort daran, dass es mit einer roten Linie für einen Syntaxfehler markiert wird.

Siehe auch

Bereiche verwenden (Seite 7646)

Bereiche einfügen (Seite 7648)

Bereiche kopieren und einfügen (Seite 7650)

In Bereichen navigieren (Seite 7651)

16.4.6 Datentyp einer SCL-Anweisung festlegen

16.4.6.1 Grundlagen zu den Datentypen von SCL-Anweisungen

Einführung

Die SCL-Anweisungen, die Sie für die Programmierung Ihrer Bausteine einsetzen, nutzen bestimmte Datentypen für die Ermittlung des Funktionswerts. Dabei gibt es SCL-Anweisungen, für die nur ein ganz bestimmter Datentyp verwendet werden kann. Bei diesen Anweisungen können Sie den Datentyp nicht ändern. Die Mehrheit der SCL-Anweisungen kann jedoch mit unterschiedlichen Datentypen arbeiten. Bei diesen Anweisungen werden folgende beide Arten unterschieden:

- Anweisungen, bei denen der Datentyp des Funktionswerts durch den Datentyp der Eingangsparameter bestimmt wird. Dies ist bei den meisten Anweisungen der Fall.
- Anweisungen, bei denen ein Datentyp voreingestellt ist. Dies betrifft die Anweisungen, die in der unten stehenden Tabelle aufgelistet sind.

Bei der zweiten Gruppe müssen Sie den voreingestellten Datentyp ändern, falls dieser vom Datentyp des verwendeten Eingangsparameters abweicht. Grundsätzlich können Sie den Datentyp mit der folgenden Syntax ändern:

```
_<Datentyp>
```

SCL-Anweisungen mit voreingestelltem Datentyp

Die folgende Tabelle zeigt die SCL-Anweisungen mit voreingestellten Datentypen:

Anweisung	Voreingestellter Datentyp
CEIL	DINT
DECO	DWORD
FLOOR	DINT
NORM_X	REAL
PEEK	BYTE
SCALE_X	INT
TRUNC	DINT
CONCAT	STRING
T_DIFF	TIME

Siehe auch

Datentyp einer SCL-Anweisung ändern (Seite 7655)

Beispiel zum Ändern des Datentyps einer SCL-Anweisung (Seite 7656)

16.4.6.2 Datentyp einer SCL-Anweisung ändern

Vorgehen

Um eine SCL-Anweisung einzufügen und anschließend ihren Datentyp zu ändern, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
2. Geben Sie die Operanden für die Anweisung an.
   Der Datentyp des Funktionswerts wird entsprechend den Eingangsparametern festgelegt oder es wird der voreingestellte Datentyp der Anweisung verwendet.
3. Ergänzen Sie den Anweisungsnamen um "_<Datentyp>".
   "<Datentyp>" ist dabei der gewünschte Datentyp der Anweisung.

Siehe auch

Grundlagen zu den Datentypen von SCL-Anweisungen (Seite 7654)

Beispiel zum Ändern des Datentyps einer SCL-Anweisung (Seite 7656)

16.4.6.3 Datentyp von IEC-Zeiten und IEC-Zählern ändern

IEC-Zeiten und IEC-Zähler sind systeminterne Funktionsbausteine, die einen Instanz-Datenbaustein benötigen. Den Instanz-Datenbaustein können Sie entweder als Einzel- oder Multiinstanz anlegen. Der Datentyp des Instanz-Datenbausteins richtet sich nach der zugehörigen Anweisung. Für CPUs der Baureihen S7-1200 und S7-1500 können Sie die Anweisungen jedoch auch mit unterschiedlichen Datentypen ausführen, um sie Ihren Bedürfnissen anzupassen.

Wenn der neu eingestellte Datentyp des Instanz-Datenbausteins nicht mit dem Datentyp der Eingangsparameter übereinstimmt, erfolgt eine implizite Konvertierung, falls dies möglich ist. Wenn die Konvertierung nicht möglich ist, erhalten Sie eine Fehlermeldung.

Vorgehen

Um den Datentyp des Instanz-Datenbausteins von IEC-Zeiten oder IEC-Zählern zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, in dem Sie die IEC-Zeit oder den IEC-Zähler aufrufen.
   Abhängig vom Instanztyp des Instanz-Datenbausteins befindet sich ein grün umrandetes Rechteck vor (Multiinstanz) oder nach (Einzelinstanz) dem Namen des Instanz-Datenbausteins.
2. Klicken Sie auf das grün umrandete Rechteck.
   Eine Klappliste mit den möglichen Datentypen für den Instanz-Datenbaustein wird geöffnet.
3. Wählen Sie den gewünschten Datentyp.

16.4.6.4 Beispiel zum Ändern des Datentyps einer SCL-Anweisung

Ändern des voreingestellten Datentyps der Anweisung "Decodieren" (DECO)

Wenn Sie die Anweisung "Decodieren" in das Programm einfügen, ist der Datentyp DWORD bereits voreingestellt.

```
"Tag_Result" := DECO(IN := "Tag_Value");
```

Um den Datentyp anschließend von DWORD zu BYTE zu ändern, ändern Sie den Programmcode folgendermaßen:

```
"Tag_Result_BYTE" := DECO_BYTE(IN := "Tag_Value");
```

Siehe auch

Grundlagen zu den Datentypen von SCL-Anweisungen (Seite 7654)

Datentyp einer SCL-Anweisung ändern (Seite 7655)

16.4.7 Variableninformationen ein- oder ausblenden

Einführung

Unabhängig von der absoluten oder symbolischen Darstellung der Operanden können Sie einfache oder hierarchische Kommentare zur Dokumentation von globalen Variablen ein- bzw. ausblenden. Diese Informationen werden aus der PLC-Variablentabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden, werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet.

Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete Bausteine wieder einblenden.

Wenn Sie die Anzeige von Variableninformationen mit hierarchischen Kommentaren wählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt. Die Anzeige erfolgt dabei in Klammern hinter dem Kommentar der Variablen, die Kommentare der einzelnen Ebenen werden jeweils durch einen Punkt getrennt. Falls auf einer Strukturebene für eine Variable kein Kommentar vorhanden ist, entfällt er in der Anzeige, was Sie an zwei aufeinander folgenden Punkten erkennen können.

Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".

3. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Variableninformationen" entweder die Option "Einblenden" oder die Option "Variableninformationen mit Hierarchie", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
4. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Variableninformationenen" die Option "Ausblenden".
   Die Variableninformationen werden für alle geöffneten Bausteine ein- oder ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen abhängig von der gewählten Einstellung ein- oder ausgeblendet.

Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." entweder die Option "Variableninformationen einblenden" oder die Option "Variableninformationen mit Hierarchie", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
2. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." die Option "Variableninformationen ausblenden". Die Variableninformationen werden ein- oder ausgeblendet.

16.4.8 Favoriten in SCL verwenden

16.4.8.1 SCL-Anweisungen zu den Favoriten hinzufügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

SCL-Anweisungen mithilfe von Favoriten einfügen (Seite 7658)

SCL-Anweisungen aus den Favoriten entfernen (Seite 7659)

16.4.8.2 SCL-Anweisungen mithilfe von Favoriten einfügen

Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

SCL-Anweisungen zu den Favoriten hinzufügen (Seite 7657)

SCL-Anweisungen aus den Favoriten entfernen (Seite 7659)

16.4.8.3 SCL-Anweisungen aus den Favoriten entfernen

Voraussetzung

Ein Codebaustein ist geöffnet.

Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

Übersicht über den Programmiereditor (Seite 7117)

SCL-Anweisungen zu den Favoriten hinzufügen (Seite 7657)

SCL-Anweisungen mithilfe von Favoriten einfügen (Seite 7658)

16.4.9 Bausteinaufrufe in SCL einfügen

16.4.9.1 Grundlagen zum Bausteinaufruf in SCL

Aufruf von Funktionsbausteinen

Syntax eines Aufrufs

Für den Aufruf eines Funktionsbausteins als Einzel- oder Multiinstanz wird die folgende Syntax verwendet:

- Einzelinstanz:
  - Wenn der Funktionsbaustein aus dem Projekt stammt:
    `<DBName> (Parameterliste)`
  - Wenn der Funktionsbaustein aus der Task Card "Anweisungen" stammt:
    `<DBName>.<Anweisungsname> (Parameterliste)`
    oder
    `<Anweisungsname> (Parameterliste)`
- Multiinstanz
  `<#Instanz-Name> (Parameterliste)`

Aufruf als Einzel-/ oder Multiinstanz

Funktionsbausteine können Sie entweder als Einzelinstanz oder als Multiinstanz aufrufen.

- Aufruf als Einzelinstanz
  Der aufgerufene Funktionsbaustein speichert seine Daten in einem eigenen Datenbaustein.
- Aufruf als Multiinstanz
  Der aufgerufene Funktionsbaustein speichert seine Daten im Instanz-Datenbaustein des aufrufenden Funktionsbausteins.

Weitere Informationen zu den Aufrufarten finden Sie unter "Siehe auch".

Rekursive Bausteinaufrufe

Sie haben die Möglichkeit, einen Baustein rekursiv aufzurufen. Das bedeutet, dass ein Baustein sich selbst wieder aufrufen kann. Beachten Sie dabei, dass die Aufruftiefe auf 24 beschränkt ist und dass keine Multiinstanzen verwendet werden können.

Parameterliste

Wenn Sie aus einem SCL-Baustein heraus einen anderen Codebaustein aufrufen, können Sie die Formalparameter des aufgerufenen Bausteins mit Aktualparametern versorgen.

Die Angabe der Parameter hat die Form einer Wertzuweisung. Durch diese Wertzuweisung weisen Sie den Parametern, die Sie im aufgerufenen Baustein definiert haben, einen Wert (Aktualparameter) zu.

Die Formalparameter des aufgerufenen Codebausteins werden in einer Klammer direkt nach dem Aufruf aufgelistet. Eingangs- und Durchgangsparameter haben die Zuweisungskennung ":=", Ausgangsparameter haben die Zuweisungskennung "=>". Ein Platzhalter hinter dem Parameter zeigt den erforderlichen Datentyp und die Art des Parameters an.

Regeln zur Parameterversorgung

Zur Parameterversorgung gelten folgende Regeln:

- Als Aktualparameter können Konstanten, Variablen und Ausdrücke verwendet werden.
- Die Reihenfolge der Zuweisungen ist beliebig.
- Datentyp von Formal- und Aktualparameter stimmen überein. Dabei können Sie für den Aktualparameter auch Datentypen verwenden, für die eine implizite Datentypkonvertierung in den Datentyp des Formalparameters möglich ist.
- Die einzelnen Zuweisungen sind durch Komma getrennt.
- Wenn der aufgerufene Baustein nur einen Parameter besitzt, genügt die Angabe des Aktualparameters in der Klammer. Der Formalparameter muss nicht mit angegeben werden.

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

Beispiele zum Aufruf eines Funktionsbausteins in SCL (Seite 7663)

Bausteinaufrufe (Seite 62)

Operanden verwenden und adressieren (Seite 93)

Aufruf von Funktionen

Syntax eines Aufrufs

Für den Aufruf einer Funktion wird die folgende Syntax verwendet:

```
<Funktionsname> (Parameterliste);     //Standardaufruf
<Operand>:=<Funktionsname> (Parameterliste); // Aufruf in einem
Ausdruck
```

Funktionswert

Funktionen, die einen Rückgabewert liefern, können in SCL in jedem Ausdruck anstelle eines Operanden verwendet werden. Deshalb wird der Rückgabewert in SCL auch als "Funktionswert" bezeichnet.

Die Aufrufmöglichkeiten von Funktionen hängen in SCL davon ab, ob die Funktion einen Funktionswert an den aufrufenden Baustein zurückgibt.

Der Funktionswert wird im Parameter RET_VAL definiert. Ist der Parameter RET_VAL vom Datentyp VOID, liefert die Funktion keinen Wert an den aufrufenden Baustein zurück. Wenn

der Parameter RET_VAL einen anderen Datentyp hat, liefert die Funktion einen Funktionswert dieses Datentyps zurück.

Für den Parameter RET_VAL sind in SCL alle Datentypen zulässig mit Ausnahme von ANY, ARRAY, STRUCT und VARIANT sowie der Parametertypen TIMER und COUNTER.

Aufrufmöglichkeiten

Es gibt zwei Möglichkeiten des Funktionsaufrufs in SCL:

- Standardaufruf für Funktionen mit und ohne Funktionswert
  Beim Standardaufruf stehen die Ergebnisse der Funktion als Ausgangs- und Durchgangsparameter zur Verfügung.
- Aufruf in einem Ausdruck für Funktionen mit Funktionswert
  Funktionen, die einen Funktionswert liefern, können in jedem Ausdruck anstelle eines Operanden verwendet werden, z. B. in einer Wertzuweisung.
  Die Funktion berechnet den Funktionswert, der den gleichen Namen trägt wie die Funktion, und gibt ihn an den aufrufenden Baustein zurück. Dort ersetzt der Wert den Funktionsaufruf. Nach dem Aufruf stehen die Ergebnisse der Funktion als Funktionswert oder als Ausgangs- und Durchgangsparameter zur Verfügung.

Rekursive Bausteinaufrufe

Sie haben die Möglichkeit, einen Baustein rekursiv aufzurufen. Das bedeutet, dass ein Baustein sich selbst wieder aufrufen kann. Beachten Sie dabei, dass die Aufruftiefe auf 24 beschränkt ist und dass keine Multiinstanzen verwendet werden können.

Parameterliste

Wenn Sie aus einem SCL-Baustein heraus einen anderen Codebaustein aufrufen, müssen Sie die Formalparameter des aufgerufenen Bausteins mit Aktualparametern versorgen.

Die Angabe der Parameter hat die Form einer Wertzuweisung. Durch diese Wertzuweisung weisen Sie den Parametern, die Sie im aufgerufenen Baustein definiert haben, einen Wert (Aktualparameter) zu.

Die Formalparameter des aufgerufenen Codebausteins werden in einer Klammer direkt nach dem Aufruf aufgelistet. Eingangs- und Durchgangsparameter haben die Zuweisungskennung ":=", Ausgangsparameter haben die Zuweisungskennung "=>". Ein grau hinterlegter Platzhalter hinter dem Parameter zeigt den erforderlichen Datentyp und die Art des Parameters an.

Regeln zur Parameterversorgung

Zur Parameterversorgung von Funktionen gelten folgende Regeln:

- Alle Parameter der Funktion müssen versorgt werden.
- Die Reihenfolge der Zuweisungen ist beliebig.
- Als Aktualparameter können Konstanten, Variablen und Ausdrücke verwendet werden.

- Datentyp von Formal- und Aktualparameter stimmen überein. Dabei können Sie für den Aktualparameter auch Datentypen verwenden, für die eine implizite Datentypkonvertierung in den Datentyp des Formalparameters möglich ist.
- Die einzelnen Zuweisungen sind durch Komma getrennt.
- Wenn der aufgerufene Baustein nur einen Parameter besitzt, genügt die Angabe des Aktualparameters in der Klammer. Der Formalparameter muss nicht mit angegeben werden.
- Beim Aufruf von Funktionen in SCL können Sie den Freigabemechanismus über EN nicht nutzen. Verwenden Sie stattdessen eine IF-Anweisung, um Funktionen bedingt aufzurufen.

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

Beispiele zum Aufruf von Funktionen in SCL (Seite 7664)

Bausteinaufrufe (Seite 62)

Operanden verwenden und adressieren (Seite 93)

Beispiele zum Aufruf eines Funktionsbausteins in SCL

Aufruf als Einzelinstanz

Das folgende Beispiel zeigt den Aufruf eines FB als Einzelinstanz:

SCL

```
// Aufruf als Einzelinstanz
"MyDB" (MyInput:=10, MyInout:= "Tag1");
```

Ergebnis

Nach der Ausführung des Aufrufs steht der für den Durchgangsparameter "MyInout" ermittelte Wert in "Tag1" im Datenbaustein "MyDB" zur Verfügung.

Aufruf als Multiinstanz

Das folgende Beispiel zeigt den Aufruf eines FB als Multiinstanz:

SCL

```
// Aufruf als Multiinstanz
#MyFB (MyInput:= 10, MyInout:= "Tag1");
```

Ergebnis

Nach der Ausführung des Bausteins "#MyFB" steht der für den Durchgangsparameter "MyInout" ermittelte Wert in "Tag1" im Datenbaustein des aufrufenden Codebausteins zur Verfügung.

Siehe auch

Aufruf von Funktionsbausteinen (Seite 7660)

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

Bausteinaufrufe (Seite 62)

Operanden verwenden und adressieren (Seite 93)

Beispiele zum Aufruf von Funktionen in SCL

Standardaufruf

Das folgende Beispiel zeigt einen Standard-Funktionsaufruf:

SCL

```
// Standard-Funktionsaufruf
"MyFC" (MyInput := 10, MyInOut := "Tag1");
```

Ergebnis

Nach der Ausführung des Bausteins "MyFC" steht der für den Durchgangsparameter "MyInOut" ermittelte Wert in "Tag1" im aufrufenden Baustein zur Verfügung und muss dort weiter verarbeitet werden.

Aufruf in einer Wertzuweisung

Das folgende Beispiel zeigt einen Funktionsaufruf in einer Wertzuweisung:

SCL

```
(*Aufruf in einer Wertzuweisung, für "MyFC" wurde ein Funktionswert
definiert *)
#MyOperand := "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 := 8.9,
MyInOut := "Tag1");
```

Ergebnis

Der Funktionswert von "MyFC" wird an "#MyOperand" übertragen.

Aufruf in einem arithmetischen Ausdruck

Das folgende Beispiel zeigt einen Funktionsaufruf in einem arithmetischen Ausdruck:

```
SCL
(*Aufruf in einem arithmetischen Ausdruck, für "MyFC" wurde ein
Funktionswert definiert *)
#MyOperand := "Tag2" + "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 :=
8.9);
```

Ergebnis

Der Funktionswert von "MyFC" wird zu "Tag2" addiert und das Ergebnis an "MyOperand" übertragen.

Siehe auch

Aufruf von Funktionen (Seite 7661)

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

16.4.9.2 Bausteinaufrufe manuell einfügen

Sie können Aufrufe für Funktionen (FCs) und Funktionsbausteine (FBs) einfügen. Funktionsbausteine können Sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Funktionsnamen ein.
2. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.
3. Wählen Sie im Kontextmenü den Befehl "Alle Parameter anzeigen" oder drücken Sie die Tastenkombination <Strg+Shift+Leertaste>. Alternativ können Sie auch in der Funktionsleiste auf die Schaltfläche "Erweitert/reduziert die Parameterliste von Bausteinaufrufen" klicken.
   Die Syntax für den Funktionsaufruf einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.

4. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablentabelle per Drag & Drop auf den Platzhalter ziehen.
5. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter. Mit <Shift + TAB> können Sie zum vorherigen Platzhalter navigieren.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Namen des Funktionsbausteins ein.
2. Geben Sie eine öffnende Klammer "(" ein.
   Der Dialog "Aufrufoptionen" wird geöffnet.
3. Bestätigen Sie Ihre Eingaben mit "OK".
   Die Syntax für den Aufruf des Funktionsbausteins einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
4. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
5. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
6. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablentabelle per Drag & Drop auf den Platzhalter ziehen.
7. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter. Mit <Shift + TAB> können Sie zum vorherigen Platzhalter navigieren.

Ergebnis

Der Bausteinaufruf wird eingefügt.

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe aktualisieren (Seite 7669)

Parameterliste erweitern und reduzieren (Seite 7670)

Grundlagen zum Bausteinaufruf in SCL (Seite 7660)

Bausteinaufrufe (Seite 62)

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

16.4.9.3 Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FCs) und Funktionsbausteine (FBs) per Drag & Drop aus der Projektnavigation einfügen.

Funktionsbausteine können Sie entweder als Einzelinstanz, Multiinstanz oder Parameterinstanz aufrufen.

Siehe auch: Grundlagen zu Instanzen (Seite 64)

Voraussetzung

Die aufzurufende Funktion (FC) oder der aufzurufende Funktionsbaustein (FB) ist vorhanden.

Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf per Drag & Drop einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das Programm.
   Die Syntax für den Funktionsaufruf einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
2. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablentabelle per Drag & Drop auf den Platzhalter ziehen.
3. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter. Mit <Shift + TAB> können Sie zum vorherigen Platzhalter navigieren.

Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) per Drag & Drop einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das Programm.
   Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
3. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
4. Bestätigen Sie Ihre Eingaben mit "OK".
   Die Syntax für den Aufruf des Funktionsbausteins einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
5. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablentabelle per Drag & Drop auf den Platzhalter ziehen.
6. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter. Mit <Shift + TAB> können Sie zum vorherigen Platzhalter navigieren.

Ergebnis

Der Bausteinaufruf wird eingefügt.

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

Siehe auch

Bausteinaufrufe aktualisieren (Seite 7669)

Parameterliste erweitern und reduzieren (Seite 7670)

Grundlagen zum Bausteinaufruf in SCL (Seite 7660)

Bausteinaufrufe (Seite 62)

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 7149)

16.4.9.4 Bausteinaufrufe aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben die folgenden Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren aller inkonsistenten Bausteinaufrufe im Programmiereditor.
  Die inkonsistenten Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt. Beachten Sie jedoch, dass die Parameter bei Funktionsbausteinen (FBs) ausgeblendet werden und die Versorgung der Parameter über den entsprechenden Instanz-Datenbaustein (DB) erfolgt. Bei Bedarf können Sie die Parameter über den Kontextmenübefehl "Alle Parameter anzeigen" einblenden.
  - Gelöschte Parameter werden nicht entfernt. Erweitern Sie gegebenenfalls die Parameterliste, um gelöschte Parameter manuell zu entfernen.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Explizites Aktualisieren eines Bausteinaufrufs im Programmiereditor.
  Der inkonsistente Aufruf dieses Bausteins wird an allen Aufrufstellen aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt.
  - Gelöschte Parameter werden nicht entfernt. Erweitern Sie gegebenenfalls die Parameterliste, um gelöschte Parameter manuell zu entfernen.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert. Beachten Sie, dass Sie vor dem Übersetzungsvorgang gelöschte Parameter manuell entfernen und beim Aufruf von Funktionen alle neuen Formalparameter mit Aktualparametern versorgen müssen.

Alle inkonsistenten Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein alle Bausteinaufrufe zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf "Inkonsistente Bausteinaufrufe aktualisieren".
   Alle inkonsistenten Aufrufe werden aktualisiert. Versorgen Sie gegebenenfalls neue Formalparameter von Funktionen (FCs) mit Aktualparametern.

Einen bestimmten Bausteinaufruf im Programmiereditor aktualisieren

Um einen bestimmten Bausteinaufruf im Programmiereditor zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf, den Sie aktualisieren möchten.

3. Wählen Sie im Kontextmenü den Befehl "Bausteinaufruf aktualisieren".
4. Falls Parameter ergänzt wurden, geben Sie die Werte für die neuen Bausteinparameter an.

Hinweis

Beachten Sie, dass Ihnen der Befehl "Bausteinaufruf aktualisieren" nur zur Verfügung steht, wenn Sie nicht zuvor alle Bausteinaufrufe im Editor über den Befehl "Inkonsistente Bausteinaufrufe aktualisieren" aktualisiert haben.

Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

Parameterliste erweitern und reduzieren (Seite 7670)

16.4.10 Parameterliste erweitern und reduzieren

Wenn Sie in SCL Bausteine aufrufen oder Anweisungen einfügen, die systemintern Funktionsbausteine sind, wird die Syntax einschließlich der Parameterliste mit den Platzhaltern für die Aktualparameter in das SCL-Programm eingefügt. Um die Übersichtlichkeit des Programmcodes zu erhöhen, wird die Parameterliste um die unbeschalteten optionalen Parameter reduziert, wenn Sie andere Anweisungen bearbeiten. Sie können Sie jederzeit wieder erweitern. Genauso können Sie die Parameterliste aber auch explizit reduzieren, wenn Sie mit der Beschaltung der Parameter fertig sind.

Parameterliste erweitern

Um die Parameterliste zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Bausteinaufruf oder die Anweisung.
2. Wählen Sie im Kontextmenü den Befehl "Parameterliste erweitern" oder verwenden Sie die Tastenkombination <Strg+Shift+Leertaste>.
   Die Parameterliste wird wieder vollständig angezeigt.

Parameterliste reduzieren

Um die Parameterliste zu reduzieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Bausteinaufruf oder die Anweisung.
2. Wählen Sie im Kontextmenü den Befehl "Parameterliste reduzieren" oder verwenden Sie die Tastenkombination <Strg+Shift+Leertaste>.
   Alle unbeschalteten optionalen Parameter werden ausgeblendet.

Siehe auch

SCL-Anweisungen manuell eingeben (Seite 7643)

SCL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 7644)

Bausteinaufrufe manuell einfügen (Seite 7665)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7667)

16.4.11 Pragmas einfügen

Sie haben die Möglichkeit, Ihren Programmcode durch Pragmas bestimmten Gültigkeiten zuzuordnen. Dazu umschließen Sie den Programmcode mit Pragma-Schlüsselwörtern und vergeben mindestens einen Parameter. Optional können Sie weitere Parameter vergeben, um den Inhalt zu definieren. Beachten Sie, dass das Zeichen "$" in String-Konstanten als Escape-Sequenz dient und daher nicht als normales Zeichen für den Parameternamen verwendet werden kann.

Die allgemeine Syntax für Pragmas sieht folgendermaßen aus:

```
{PRAGMA_BEGIN 'Param1', 'Param2', ... , 'ParamN'}
      //Programmcode
{PRAGMA_END}
```

Pragmas können auch ineinander verschachtelt werden. Achten Sie jedoch darauf, dass Pragmas innerhalb anderer Pragmas vollständig abgeschlossen sind. Pragmas können bei Bedarf auf- und zugeklappt werden.

Hinweis

Die Auswertung von Pragmas erfolgt zurzeit nur in SiVArc. Dazu muss der Parameter "SIVARC" und ein weiterer Parameter vergeben werden.

Weitere Informationen finden Sie in der Hilfe zu SiVArc.

Vorgehen

Um ein Pragma einzufügen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke an die Position, an der Sie ein Pragma einfügen möchten.
2. Geben Sie "{ PRAGMA_BEGIN" ein.

3. Geben Sie mindestens ein Parameter in einfachen Anführungszeichen ein.
4. Geben Sie beliebig viele weitere Parameter in einfachen Anführungszeichen ein. Dieser Schritt ist optional.
5. Geben Sie "}" ein.
6. Setzen Sie die Einfügemarke an das Ende des Programmcodes, den Sie mit dem Pragma umschließen möchten.
7. Geben Sie "{ PRAGMA_END}" ein.
   Der entsprechende Programmcode wird in einem Pragma zusammengefasst.

16.4.12 Kommentare einfügen

Programmcode kommentieren

Sie haben verschiedene Möglichkeiten, um SCL-Programme zu kommentieren:

- Zeilenkommentar
  Ein Zeilenkommentar wird mit "//" eingeleitet und erstreckt sich bis zum Ende der Zeile.
- Kommentarabschnitt
  Ein Kommentarabschnitt wird mit "(*" eingeleitet und durch "*)" abgeschlossen. Er kann sich über mehrere Zeilen erstrecken.

Zeilenkommentar einfügen

Um einen Zeilenkommentar einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "//" an der Stelle ein, an der Sie den Kommentar platzieren wollen. Dies muss nicht der Zeilenbeginn sein.
2. Geben Sie den Kommentartext ein.

Kommentarabschnitt einfügen

Um einen Kommentarabschnitt einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "(*" an der Stelle ein, an der Sie den Kommentar platzieren wollen. Dies muss nicht der Zeilenbeginn sein.
2. Geben Sie den Kommentartext ein.
3. Schließen Sie den Kommentar mit "*)" ab.

Eine oder mehrere Zeilen durch Kommentieren deaktivieren

Um Programmcode durch Kommentieren zu deaktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie auskommentieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Markierung auskommentieren".
   In die markierten Zeilen wird am Zeilenbeginn "//" eingefügt. Der nachfolgende Code wird als Kommentar interpretiert. Werden Zeilen deaktiviert, die bereits einen Zeilenkommentar enthalten, wird ebenfalls "//" eingefügt. Werden die Zeilen anschließend wieder aktiviert, bleiben ursprüngliche Kommentare erhalten.

Kommentarzeilen aktivieren

Um auskommentierte Zeilen wieder als Code zu aktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie aktivieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Kommentar entfernen".
   Die Kennzeichnung "//" für Zeilenkommentare am Beginn der Zeile wird entfernt.

Beispiel

Der folgende Code enthält Kommentarabschnitte und Zeilenkommentare

```
(*******************************************************************************
  Hier kann eine Beschreibung der folgenden Anweisungen stehen
*******************************************************************************)
IF "MyVal1" > 0 THEN //Keine Division durch 0
    "MyReal" := "MyVal2" (* Eingabewert *) / "MyVal1" (* Messwert *);
END_IF;
//Datentypkonvertierung
"MyInt" := REAL_TO_INT("MyReal");
```

Siehe auch

(*...*): Kommentarabschnitt einfügen (Seite 1921)

16.5 SCL-Anweisungen bearbeiten

16.5.1 Anweisungen selektieren

Sie können einzelne Anweisungen oder alle Anweisungen eines Bausteins selektieren.

Voraussetzung

Ein SCL-Baustein ist geöffnet.

Einzelne Anweisungen selektieren

Um einzelne Anweisungen zu selektieren, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke vor das erste Zeichen, das Sie markieren möchten.
2. Drücken und halten Sie die linke Maustaste.
3. Bewegen Sie den Mauszeiger hinter das letzte Zeichen, das Sie markieren möchten.
4. Lassen Sie die linke Maustaste los.

Alle Anweisungen eines Programms selektieren

Um alle Anweisungen zu selektieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

Hinweis

Wenn Sie das öffnende Schlüsselwort einer Anweisung markieren, wird automatisch auch das schließende Schlüsselwort markiert.

16.5.2 Anweisungen kopieren, ausschneiden und einfügen

Anweisung kopieren

Um eine Anweisung zu kopieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Anweisung ausschneiden

Um eine Anweisung zu auszuschneiden, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

Anweisung aus der Zwischenablage einfügen

Um eine Anweisung aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie eine Anweisung oder schneiden Sie eine Anweisung aus.
2. Klicken Sie auf die Stelle, an der Sie die Anweisung einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

16.5.3 Anweisungen löschen

Voraussetzung

Ein SCL-Baustein ist geöffnet

Vorgehen

Um eine Anweisung zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

16.6 Programmierbeispiele SCL

16.6.1 Beispiel für das Steuern eines Förderbands

Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Drucktaster, S1 für START und S2 für STOP. Am Ende des Förderbands befinden sich ebenfalls zwei Drucktaster, S3 für START und S4 für STOP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.

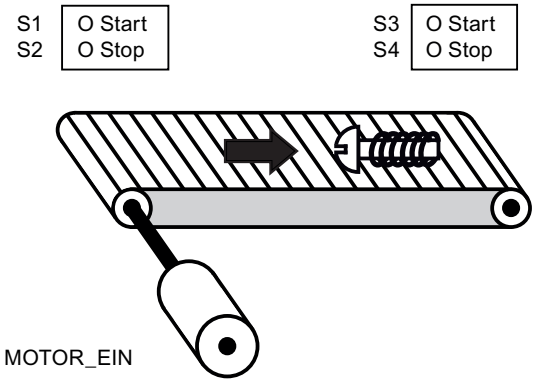

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Operand	Deklaration	Datentyp	Beschreibung
Starttaster_Links (S1)	Input	BOOL	Starttaster am linken Rand des Förderbands
Stoptaster_Links (S2)	Input	BOOL	Stoptaster am linken Rand des Förderbands
Starttaster_Rechts (S3)	Input	BOOL	Starttaster am rechten Rand des Förderbands
Stoptaster_Rechts (S4)	Input	BOOL	Stoptaster am rechten Rand des Förderbands
MOTOR_EIN	Output	BOOL	Motor des Förderbands einschalten
MOTOR_AUS	Output	BOOL	Motor des Förderbands ausschalten

Das folgende SCL-Programm zeigt die Umsetzung der Aufgabe:

SCL

```
IF "Starttaster_Links_S1" OR "Starttaster_Rechts_S3" THEN
"MOTOR_EIN" := 1;
"MOTOR_AUS" := 0;
END_IF;
IF "Stoptaster_Links_S2" OR "Stoptaster_Rechts_S4" THEN
"MOTOR_EIN" := 0;
"MOTOR_AUS" := 1;
END_IF;
```

Wenn der Starttaster "Starttaster_Links_S1" oder "Starttaster_Rechts_S3" betätigt wird, wird der Motor des Förderbands eingeschaltet. Wenn der Stoptaster "Stoptaster_Links_S2" oder "Stoptaster_Rechts_S4" betätigt wird, wird der Motor des Förderbands ausgeschaltet.

Siehe auch

Logische Ausdrücke (Seite 7617)

Grundlagen zu SCL (Seite 7607)

Einstellungen zu SCL (Seite 7631)

Das Programmierfenster von SCL (Seite 7633)

SCL-Anweisungen eingeben (Seite 7642)

SCL-Anweisungen bearbeiten (Seite 7674)

16.6.2 Beispiel für das Erfassen der Richtung eines Förderbands

Erfassen der Richtung eines Förderbands

Die erkannte Laufrichtung des Bandes wird mit einem Pfeil RECHTS oder einem Pfeil LINKS angezeigt. Läuft weiteres Fördergut von rechts an LS1 oder von links an LS2 heran, wird zunächst der angezeigte Pfeil ausgeschaltet, bis nach dem Durchlaufen beider Lichtschranken die Laufrichtung wieder neu festgestellt wurde und der entsprechende Pfeil angezeigt werden kann. Für die Lösung der Aufgabe sind noch 2 Flankenmerker notwendig, die den Signalwechsel an den zwei Lichtschranken von "0" auf "1" erkennen.

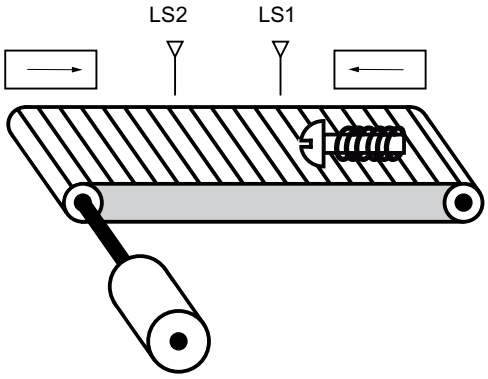

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
Lichtschranke LS1	Input	BOOL	Lichtschranke 1
Lichtschranke LS2	Input	BOOL	Lichtschranke 2
RECHTS	Output	BOOL	Anzeige für eine Bewegung nach rechts
LINKS	Output	BOOL	Anzeige für eine Bewegung nach links
Hilfsmerker LS1	Input	BOOL	Flankenmerker 1
Hilfsmerker LS2	Input	BOOL	Flankenmerker 2

Das folgende SCL-Programm zeigt die Umsetzung des Beispiels:

SCL

```
// Programmcode für den Linkslauf
IF "Lichtschranke LS1" = 1 AND "Hilfsmerker LS2" = 0 THEN
          "Hilfsmerker LS1" := 1; // Hilfsmerker für LS1 setzen
          "LINKS" := 0; // Pfeilanzeige links ausschalten
          "RECHTS" := 0; // Pfeilanzeige rechts ausschalten
END_IF;

IF "Hilfsmerker LS1" = 1 AND "Lichtschranke LS2" = 1 THEN // Das Förderband läuft nach links
```

SCL

```
          "LINKS" := 1;
          "RECHTS" := 0;
END_IF;

IF "LINKS" = 1 AND "Lichtschranke LS2" = 0 THEN // Hilfsmerker für LS1 zurücksetzen
          "Hilfsmerker LS1" := 0;
END_IF;
```

SCL

```
// Programmcode für den Rechtslauf
IF "Lichtschranke LS2" = 1 AND "Hilfsmerker LS1" = 0 THEN
          "Hilfsmerker LS2" := 1; // Hilfsmerker für LS2 setzen
          "LINKS" := 0; // Pfeilanzeige links ausschalten
          "RECHTS" := 0; // Pfeilanzeige rechts ausschalten
END_IF;

IF "Hilfsmerker LS2" = 1 AND "Lichtschranke LS1" = 1 THEN // Das Förderband läuft nach rechts
          "LINKS" := 0;
          "RECHTS" := 1;
END_IF;

IF "RECHTS" = 1 AND "Lichtschranke LS1" = 0 THEN // Hilfsmerker für LS2 zurücksetzen
          "Hilfsmerker LS2" := 0;
END_IF;
```

Wenn an der Lichtschranke "LS1" der Signalzustand "1" auftritt und gleichzeitig der Signalzustand an der Lichtschranke "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links. Wenn an der Lichtschranke "LS2" der Signalzustand "1" auftritt und gleichzeitig der Signalzustand an der Lichtschranke "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts. Die Anzeigen für eine Bewegung nach links oder rechts werden ausgeschaltet, wenn der Signalzustand an den beiden Lichtschranken "0" ist.

Siehe auch

Logische Ausdrücke (Seite 7617)

Grundlagen zu SCL (Seite 7607)

Einstellungen zu SCL (Seite 7631)

Das Programmierfenster von SCL (Seite 7633)

SCL-Anweisungen eingeben (Seite 7642)

SCL-Anweisungen bearbeiten (Seite 7674)

16.6.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, von der sie auf einen LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

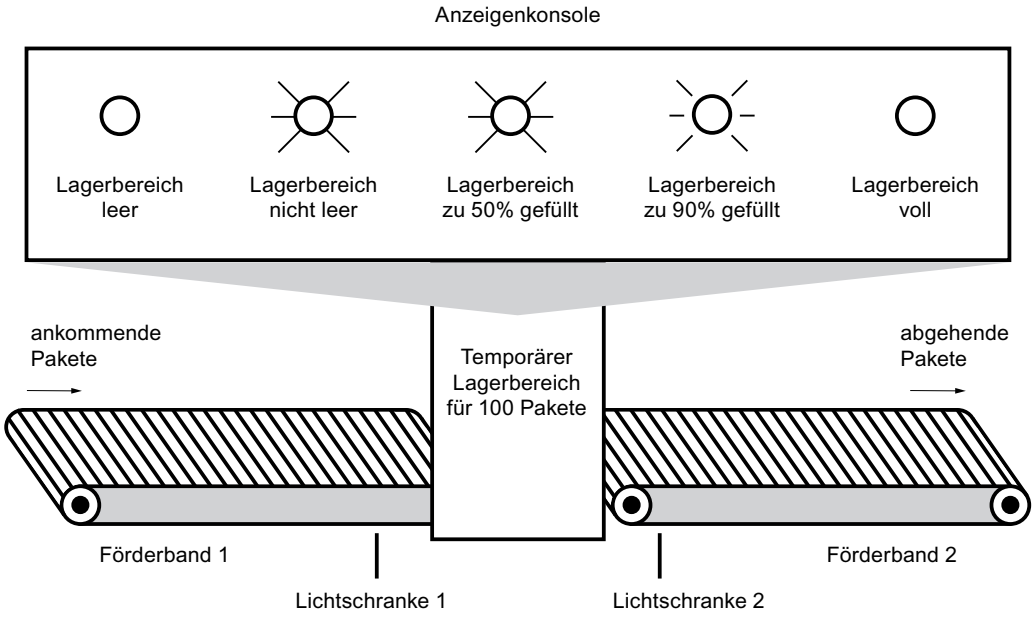

Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Name	Deklaration	Datentyp	Beschreibung
LS1	Input	BOOL	Lichtschranke 1
LS2	Input	BOOL	Lichtschranke 2
RESET	Input	BOOL	Zähler rücksetzen
LOAD	Input	BOOL	Den aktuellen Zählerwert auf den Wert des Parameters PV anpassen.
MAX LAGERFUELL-MENGE	Input	INT	Maximal mögliche Anzahl der Pakete im Lager

Name	Deklaration	Datentyp	Beschreibung
PAKETANZAHL	Output	INT	Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)
LB_PAKETE	Output	BOOL	Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELLMENGE" ist.
LAG_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich leer
LAG_NICHT_LEER	Output	BOOL	Anzeigeleuchte: Lagerbereich nicht leer
LAG_50%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 50 % voll
LAG_90%_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich zu 90 % voll
LAG_VOLL	Output	BOOL	Anzeigeleuchte: Lagerbereich voll
VOLUME_50	Input	INT	Vergleichswert: 50 Pakete
VOLUME_90	Input	INT	Vergleichswert: 90 Pakete
VOLUME_100	Input	INT	Vergleichswert: 100 Pakete

Das folgende SCL-Programm zeigt die Umsetzung des Beispiels:

Wenn ein Paket zum Lagerbereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "MAX LAGERFUELLMENGE" gesetzt. Solange der aktuelle Zählwert größer oder gleich dem Wert der Variablen "MAX LAGERFUELLMENGE" ist, liefert die Variable "LB_PAKETE" den Signalzustand "1".

SCL

```
"VOLUME_50" := 5; // Vorbesetzen des Vergleichswerts auf 50 Pakete (für den Test nur 5
Pakete)
```

SCL

```
"VOLUME_90" := 9; // Vorbesetzen des Vergleichswerts auf 90 Pakete (für den Test nur 9 Pakete)
"VOLUME_100" := 10; // Vorbesetzen des Vergleichswerts auf 100 Pakete (für den Test nur 10 Pakete)
"MAX LAGERFUELLMENGE" := 10; // Vorbesetzen des maximalen Lagerbestands auf 100 Pakete (für den Test nur 10 Pakete)

"IEC_Counter_0_DB".CTUD(CU := "LS1",
                        CD := "LS2",
                        R := "RESET",
                        LD := "LOAD",
                        PV := "MAX LAGERFUELLMENGE",
                        QU => "LB_PAKETE",
                        QD => "LAG_LEER",
                        CV => "PAKETANZAHL");
```

Solange sich Pakete im Lagerbereich befinden, wird die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.

SCL

```
"LAG_NICHT_LEER" := NOT "LAG_LEER"
```

Wenn die Anzahl der Pakete im Lagerbereich kleiner 50% ist, schalten sich die Anzeigeleuchten für die Meldung "Lagerbereich zu 50% gefüllt", " Lagerbereich zu 90% gefüllt" und "Lagerbereich voll" aus.

SCL

```
IF "PAKETANZAHL" < "VOLUME_50" THEN
"LAG_50%_VOLL" := 0;
"LAG_90%_VOLL" := 0;
"LAG_VOLL" := 0;
END_IF;
```

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50% ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50% gefüllt" ein.

SCL

```
IF "PAKETANZAHL" >= "VOLUME_50" AND "PAKETANZAHL" <= "VOLUME_90" THEN
"LAG_50%_VOLL" := 1;
"LAG_90%_VOLL" := 0;
"LAG_VOLL" := 0;
END_IF;
```

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90% ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90% gefüllt" ein. Die Anzeigenleuchte für 50% voll bleibt ebenfalls an.

SCL

```
IF "PAKETANZAHL" >= "VOLUME_90" AND "PAKETANZAHL" < "VOLUME_100" THEN
```

SCL

```
"LAG_50%_VOLL" := 1;
"LAG_90%_VOLL" := 1;
"LAG_VOLL" := 0;
END_IF;
```

Wenn die Anzahl der Pakete im Lagerbereich 100% erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein. Die Anzeigenleuchten für 50% und 90% voll bleiben ebenfalls an.

SCL

```
IF "PAKETANZAHL" >= "VOLUME_100" THEN
"LAG_50%_VOLL" := 1;
"LAG_90%_VOLL" := 1;
"LAG_VOLL" := 1;
END_IF;
```

Siehe auch

Grundlagen zu SCL (Seite 7607)

Einstellungen zu SCL (Seite 7631)

Das Programmierfenster von SCL (Seite 7633)

SCL-Anweisungen eingeben (Seite 7642)

SCL-Anweisungen bearbeiten (Seite 7674)

GRAPH-Programme erstellen 17

17.1 Grundlagen zu GRAPH

17.1.1 Programmiersprache GRAPH

Übersicht

GRAPH ist eine grafische Programmiersprache zum Erstellen von Ablaufsteuerungen. Sequenzielle Abläufe können übersichtlich und schnell mithilfe von Ablaufketten programmiert werden. Der Prozess wird dabei in Einzelschritte mit überschaubarem Funktionsumfang zerlegt und in den Ablaufketten organisiert. In den einzelnen Schritten werden die auszuführenden Aktionen festgelegt. Die Übergänge zwischen den Schritten bilden die Transitionen. Sie enthalten Bedingungen zum Weiterschalten in den nächsten Schritt.

Bausteine der Ablaufsteuerung

Eine Ablaufsteuerung steuert den Prozess in einer vorgegebenen Reihenfolge und in Abhängigkeit von bestimmten Bedingungen. Die Komplexität der Ablaufsteuerung wird von der Automatisierungsaufgabe bestimmt. Es gehören immer mindestens drei Bausteine zu einer Ablaufsteuerung:

- GRAPH-Funktionsbaustein
  Im GRAPH-Funktionsbaustein beschreiben Sie die einzelnen Schritte und Transitionen der Ablaufsteuerung innerhalb einer oder mehrerer Ablaufketten.
- Instanz-Datenbaustein
  Der Instanz-Datenbaustein enthält die Daten und Parameter der Ablaufsteuerung. Er ist dem GRAPH-Funktionsbaustein zugeordnet und kann vom System automatisch generiert werden.
- Aufrufender Codebaustein
  Damit der GRAPH-Funktionsbaustein innerhalb des Zyklus ausgeführt werden kann, muss er von einem übergeordneten Codebaustein aufgerufen werden. Dieser Baustein kann ein Organisationsbaustein (OB), eine Funktion (FC) oder ein anderer Funktionsbaustein (FB) sein. Der GRAPH-Funktionsbaustein wird dabei immer als Einzelinstanz aufgerufen.

Das folgende Bild zeigt die Zusammenarbeit der Bausteine in einer Ablaufsteuerung:

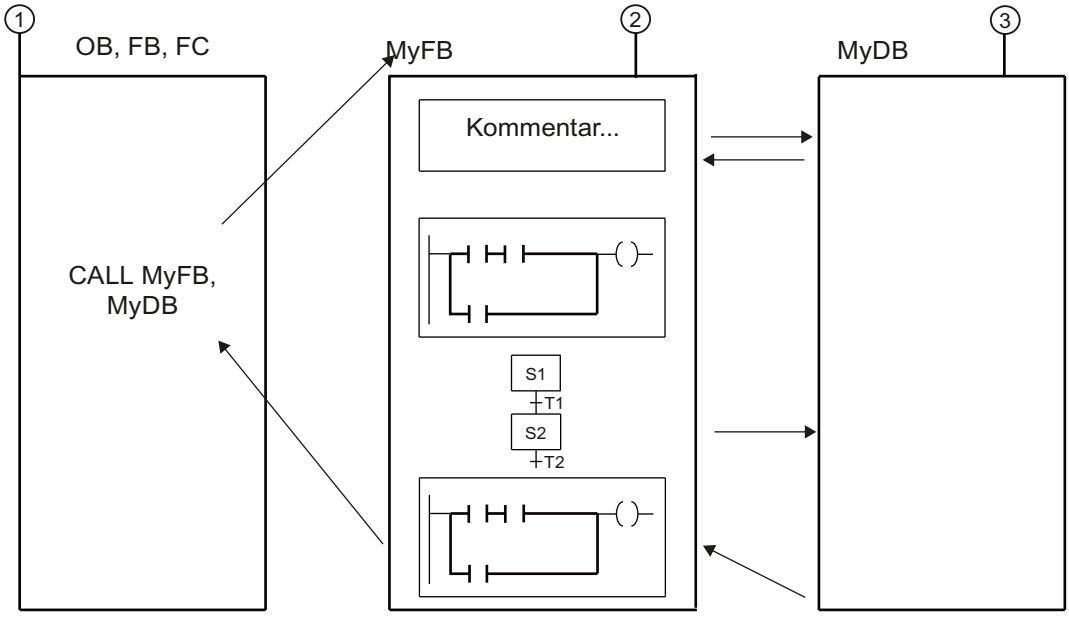

① Aufrufender Baustein

② GRAPH-Funktionsbaustein

③ Instanz-Datenbaustein

Der Zyklus eines GRAPH-Funktionsbausteines wird definiert durch den Zyklus des aufrufenden Bausteines. In jedem Zyklus werden im GRAPH-Funktionsbaustein zuerst die vorausgeschalteten permanenten Anweisungen ausgeführt. Als nächstes werden die Aktionen der aktiven Schritte abgearbeitet. Zuletzt werden die nachgeschalteten permanenten Anweisungen ausgeführt.

Hinweis

Permanente Anweisungen werden in jedem Zyklus ausgeführt, auch wenn kein Schritt aktiv ist.

Siehe auch

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.2 Ablaufketten

17.1.2.1 Grundlagen zu Ablaufketten

Einführung

Innerhalb eines GRAPH-Funktionsbausteins können Sie Ihr Programm in Form von Ablaufketten programmieren. Dabei können die Ablaufketten entweder unabhängige Aufgaben bearbeiten oder Sie zerlegen eine komplexe Aufgabe in mehrere Ablaufketten. Falls die Ablaufketten unabhängige Aufgaben bearbeiten, werden sie im Programmablauf parallel bearbeitet. Wenn Sie mehrere Ablaufketten dazu nutzen, eine komplexe Aufgabe in kleinere Segmente zu zerlegen, müssen Sie die zusammengehörenden Ablaufketten mit Sprüngen verknüpfen.

In einer Ablaufkette unterteilen Sie das Programm in einzelne Schritte. Die Schritte werden im einfachen Fall linear nacheinander bearbeitet. Durch den Einsatz von Alternativ- oder Simultanzweigen können Sie jedoch auch Ablaufketten mit komplexeren Strukturen erzeugen. Die Programmbearbeitung beginnt immer mit einem Schritt, den Sie als Initialschritt definiert haben. Eine Ablaufkette kann einen oder mehrere Initialschritte haben. Die Initialschritte können an beliebiger Stelle in der Ablaufkette stehen.

Wenn ein Schritt aktiv ist, werden seine Aktionen ausgeführt. Es können auch mehrere Schritte gleichzeitig aktiv sein, z. B. innerhalb einer Simultanverzweigung. Während ein Schritt aktiv ist, werden ständig die Bedingungen der nachfolgenden Transition geprüft. Sobald alle Bedingungen erfüllt sind und kein Supervisionsfehler ansteht, schaltet die Transition in den nachfolgenden Schritt. Dieser wird nun zum aktiven Schritt. Sie können eine Ablaufkette entweder mit einem Sprung oder einem Kettenende abschließen. Ein Sprung kann dabei zu einem beliebigen Schritt der gleichen oder einer anderen Ablaufkette erfolgen. Dies ermöglicht eine zyklische Bearbeitung der Ablaufkette.

Bearbeitungsprinzip einer Ablaufkette

Die Bearbeitung einer Ablaufkette beginnt damit, dass der oder die Initialschritte der Kette aktiv werden. Mehrere Initialschritte sind nur in Simultanverzweigungen zulässig. Solange ein Schritt aktiv ist, werden seine Aktionen ausgeführt. Dabei werden die Interlockbedingungen für die einzelnen Aktionen berücksichtigt. Nachdem alle möglichen Aktionen ausgeführt wurden, wird zunächst geprüft, ob ein Supervisionsfehler ansteht. Ist dies nicht der Fall und ist auch die nachfolgende Transition erfüllt, wird der nächste Schritt der Kette aktiv. Steht ein Supervisionsfehler an oder ist die Transition nicht erfüllt, bleibt der Schritt aktiv, bis der Fehler behoben bzw. die Transition erfüllt ist. Am Ende einer Ablaufkette können Sie durch Sprünge eine zyklische Bearbeitung der Ablaufkette ermöglichen oder mit einem Kettenende die Ablaufkette beenden.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Struktur einer Ablaufkette programmieren (Seite 7752)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.2.2 Elemente einer Ablaufkette

Übersicht

Die folgende Tabelle zeigt die möglichen Elemente einer Ablaufkette mit ihren Symbolen:

Symbol	Name
	Schritt und Transition (Seite 7692)
	Schritt (Seite 7692)
	Transition (Seite 7692)
	Simultanzweig
	Alternativzweig
	Verzweigung schließen
	Sprung zu Schritt
	Kettenende

Simultanzweig

Mithilfe von Simultanzweigen können Sie UND-Verzweigungen programmieren. Das bedeutet, dass Sie mit einer Transition mehrere Schritte aktiv setzen, deren Aktionen dann ausgeführt werden. Darum beginnen Simultanzweige immer mit einem Schritt.

Die nachfolgenden Transitionen der Simultanzweige befinden sich auf dem Hauptzweig, wobei die verschiedenen Simultanzweige an verschiedenen Stellen des Hauptzweigs geschlossen werden können. Beachten Sie, dass Zweige, die Sie in einer Transition zusammenführen, erst dann in den nächsten Schritt schalten, wenn alle Zweige komplett abgearbeitet sind.

In einer Ablaufkette können Sie bis zu 249 Simultanzweige programmieren.

Das folgende Bild zeigt die grafische Darstellung eines Simultanzweigs:

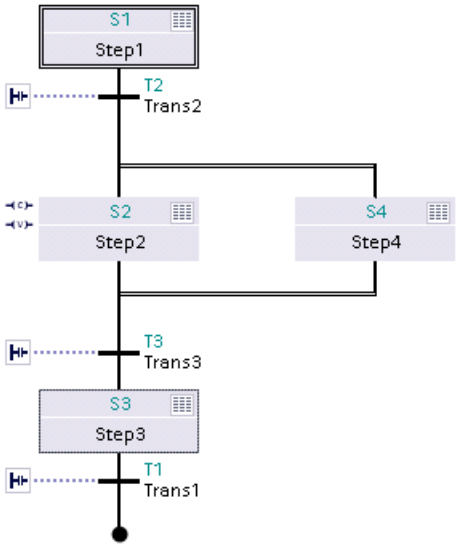

Alternativzweig

Mithilfe von Alternativzweigen können Sie ODER-Verzweigungen programmieren. Das bedeutet, dass Sie nach einem Schritt Zweige einfügen, die mit einer Transition beginnen. Abhängig davon, welche Transition zuerst erfüllt ist, wird der entsprechende Zweig abgearbeitet. Sind mehrere Transitionen gleichzeitig erfüllt, ist es von der eingestellten Betriebsart abhängig, welcher Zweig bearbeitet wird:

- Automatischer und halbautomatischer Betrieb: Die am weitesten links stehende Transition erhält die höchste Priorität und der entsprechende Zweig wird abgearbeitet.
- Handbetrieb: Die Transition, die zuerst erfüllt wurde, erhält die höchste Priorität und der entsprechende Zweig wird abgearbeitet.

In einer Ablaufkette können Sie bis zu 125 Alternativzweige programmieren.

Das folgende Bild zeigt die grafische Darstellung eines Alternativzweigs:

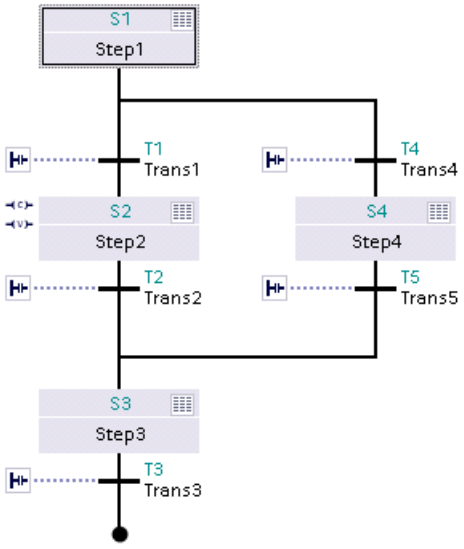

Verzweigung schließen

Mit dem Element "Verzweigung schließen" können Sie Simultan- oder Alternativzweige zu ihrem Elternzweig hin schließen. Dies ist notwendig, wenn Sie die Verzweigung nicht mit einem Sprung oder einem Kettenende abschließen möchten.

Sprung zu Schritt

Mit einem Sprung können Sie die Programmausführung mit einem beliebigen Schritt innerhalb des GRAPH-Funktionsbausteins fortsetzen. Sprünge können am Ende des Hauptzweiges, eines Alternativzweiges oder eines Simultanzweiges eingefügt werden, sodass eine zyklische Bearbeitung der Ablaufkette möglich ist. Sprung und Sprungziel werden in der Ablaufkette als Pfeile dargestellt, wobei beim Sprungziel die Absprungtransition und beim Absprung der Zielschritt angegeben ist.

Hinweis

Vermeiden Sie Sprünge von einer Transition auf einen unmittelbar vorausgehenden Schritt. Wenn Sie einen solchen Sprung dennoch benötigen, fügen Sie einen Leerschritt mit einer Transition ohne Bedingung ein.

Das folgende Bild zeigt die grafische Darstellung eines Sprungs:

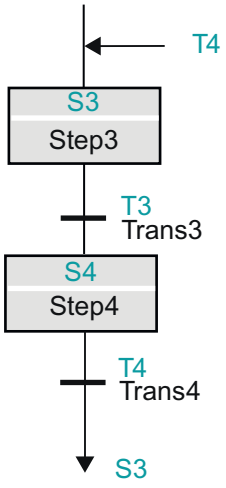

Kettenende

Sie können die Ablaufkette oder eine Verzweigung mit dem Element "Kettenende" abschließen. Dadurch wird die Ablaufkette bzw. die Verzweigung beendet. In einer Simultanverzweigung muss dem Kettenende jedoch eine Transition vorangehen. Das Element "Kettenende" wird in der Ablaufkette mit einem schwarzen Punkt dargestellt.

Das folgende Bild zeigt die grafische Darstellung eines Kettenendes:

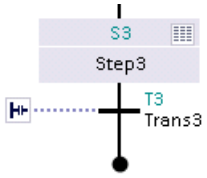

Hinweis

Werden alle Zweige einer Ablaufkette mit einem Kettenende abgeschlossen, kann die Ablaufkette entweder durch den Parameter "INIT_SQ" oder in der Task Card "Testen" in der Palette "Kettensteuerung" mit der Schaltfläche "Initialisieren" neu gestartet werden.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Struktur einer Ablaufkette programmieren (Seite 7752)

Programmiersprache GRAPH (Seite 7685)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.3 Schritte und Transitionen

17.1.3.1 Grundlagen zu Schritten

Einführung

Schritte dienen dazu, komplexe Automatisierungsaufgaben in übersichtliche Teilaufgaben zu untergliedern, die anschließend mithilfe von Aktionen gelöst werden können. Die einzelnen Schritte werden in Ablaufketten organisiert, sodass bei Programmablauf jeder Schritt in der festgelegten Reihenfolge abgearbeitet werden kann. Jeder Schritt muss einen eindeutigen Namen und eine eindeutige Nummer erhalten.

Die Aktionen eines Schritts werden ausgeführt, wenn der Schritt aktiv wird. Dazu muss ein Schritt entweder ein Initialschritt sein oder die ihm vorausgehende Transition erfüllt werden. Ein Schritt kann auch durch eine ereignisabhängige Aktion aufgerufen werden. Zusätzlich können Sie in einem Schritt Interlock- und Supervisionsbedingungen programmieren.

Sie können frei definieren, mit welchem Schritt die Bearbeitung der Ablaufkette beginnen soll. Dazu legen Sie einen beliebigen Schritt als Initialschritt fest. In Simultanzweigen können Sie mehrere Schritte als Initialschritte festlegen. Im zyklischen Betrieb einer Ablaufkette wird der Initialschritt wie jeder andere Schritt behandelt, solange keine Initialisierung der Ablaufkette über den Parameter "INIT_SQ" erfolgt. Wenn die Ablaufkette neu initialisiert wird, werden alle Initialschritte aktiviert, wenn eine positive Signalflanke an "INIT_SQ" anliegt. Alle anderen Schritte werden deaktiviert. Initialschritte erhalten in der Navigation und in der Kettenansicht einen Rahmen, sodass Sie sie leicht erkennen können.

Ein Schritt, in dem Sie keine Aktionen programmieren, ist ein Leerschritt. Da ein Leerschritt keine Aktionen bearbeiten muss, schaltet die Ablaufkette sofort zur nachfolgenden Transition weiter. Auf diese Weise können Sie zwei aufeinander folgende Transitionen programmieren.

Sie können den Namen und die Nummer eines Schritts jederzeit ändern. Zudem haben Sie die Möglichkeit Schritte mithilfe von Kommentaren zu dokumentieren.

Siehe auch

Elemente eines Schritts (Seite 7693)

Grundlagen zu Transitionen (Seite 7694)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Struktur einer Ablaufkette programmieren (Seite 7752)

Ablaufketten (Seite 7687)

Aktionen (Seite 7695)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.3.2 Elemente eines Schritts

In einem Schritt können Sie die folgenden Komponenten programmieren:

- Interlock
- Supervision
- Aktionen

Interlock

Über einen Interlock programmieren Sie Bedingungen, von denen die Ausführung der einzelnen Aktionen abhängt. Aktionen eines Schritts, die mit dem Interlock verknüpft sind, werden nur bei erfüllter Interlockbedingung ausgeführt. Ist die Interlockbedingung nicht erfüllt, liegt eine Störung vor. Für diesen Fall können Sie Meldungen festlegen, die dann angezeigt werden. Die Störung hat jedoch keinen Einfluss auf das Weiterschalten in den nächsten Schritt. Beachten Sie auch, dass beim Deaktivieren des Schritts der Interlock automatisch aufgehoben wird. Optional können Sie festlegen, dass der Bediener eine Interlockmeldung quittieren muss, um die Meldung zu schließen.

Sie haben auch die Möglichkeit, im Interlock keine Bedingungen zu programmieren, aber den Interlock trotzdem mit Aktionen zu verknüpfen. In diesem Fall ist der Interlock immer erfüllt.

Supervision

Über eine Supervision programmieren Sie Bedingungen, mit denen Sie einen ganzen Schritt überwachen können. Ist die Bedingung erfüllt, liegt eine Störung vor und der Schritt bleibt aktiv. Das heißt, die Ablaufkette schaltet in diesem Fall nicht in den nächsten Schritt weiter, sondern sie bleibt im aktiven Schritt stehen, bis die Störung behoben wurde. Auch bei der Supervision können Sie Meldungen festlegen und die Quittierpflicht aktivieren.

Aktionen

In den Schritten werden Aktionen programmiert, die z. B. Ein- oder Ausgänge steuern, andere Schritte der Ablaufkette aktivieren bzw. deaktivieren oder Bausteine aufrufen. Aktionen enthalten somit Anweisungen zur Prozesssteuerung.

Aktionen werden in Standardaktionen und ereignisabhängige Aktionen unterschieden. Standardaktionen werden ausgeführt, solange der Schritt aktiv ist. Ereignisabhängige Aktionen sind die Aktionen, die Sie mit einem Ereignis verknüpft haben.

Siehe auch

Grundlagen zu Schritten (Seite 7692)

Grundlagen zu Transitionen (Seite 7694)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Struktur einer Ablaufkette programmieren (Seite 7752)

Ablaufketten (Seite 7687)

Aktionen (Seite 7695)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.3.3 Grundlagen zu Transitionen

Einführung

Transitionen stehen zwischen den Schritten und enthalten die Bedingungen zum Weiterschalten in den nächsten Schritt. Das heißt, die Ablaufkette schaltet nur dann zu den nachfolgenden Schritten, wenn die Bedingungen der Transition erfüllt sind. Dabei werden die zur Transition gehörenden Schritte deaktiviert und die nachfolgenden Schritte aktiviert. Sind die Bedingungen nicht erfüllt, bleiben die zur Transition gehörenden Schritte aktiv. Jede Transition muss einen eindeutigen Namen und eine eindeutige Nummer erhalten. Sie können die Bedingungen einer Transition entweder in KOP oder in FUP programmieren.

Eine Transition, in der Sie keine Bedingungen programmieren, ist eine Leertransition. In diesem Fall schaltet die Ablaufkette sofort zu den nachfolgenden Schritten weiter.

Sie können den Namen und die Nummer einer Transition ändern.

Siehe auch

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Elemente eines Schritts (Seite 7693)

Struktur einer Ablaufkette programmieren (Seite 7752)

Ablaufketten (Seite 7687)

Aktionen (Seite 7695)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4 Aktionen

17.1.4.1 Grundlagen zu Aktionen

Einführung

Mit Aktionen können Sie z. B. Ein- oder Ausgänge steuern, andere Schritte der Ablaufkette aktivieren bzw. deaktivieren oder Bausteine aufrufen. Aktionen werden entweder ausgeführt, wenn der zugehörige Schritt aktiv wird (Standardaktionen) oder wenn ein mit der Aktion verbundenes Ereignis auftritt (ereignisabhängige Aktionen). In Aktionen können Sie auch Zeiten und Zähler verwenden.

Die folgende Tabelle zeigt Beispiele für die Programmierung von Aktionen:

Interlock	Ereignis	Kennung	Aktion	Beschreibung
		N	MyTag1	Aktion, bestehend aus Anweisung und Operand ohne Interlock
-(C)-		N	MyTag2	Aktion, bestehend aus Anweisung und Operand mit Interlock
	S1	S	MyTag3	Aktion, bestehend aus Ereignis, Anweisung und Operand ohne Interlock

Aktionen werden in der Reihenfolge von "oben" nach "unten" ausgeführt, wenn der Schritt aktiv ist oder das Ereignis auftritt.

Das folgende Bild zeigt ein Beispiel für die Reihenfolge der Bearbeitung von Aktionen in einem Schritt:

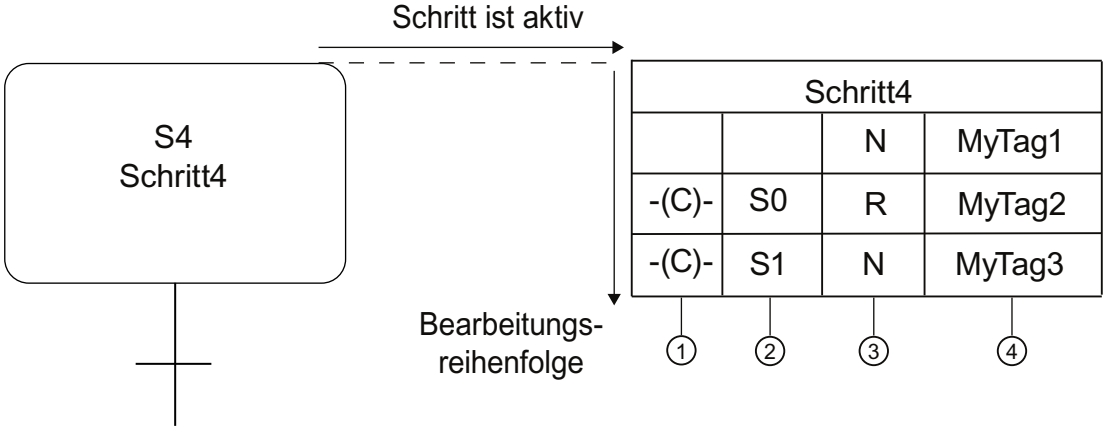

① Interlock

② Optionales Ereignis

③ Kennung

④ Operand, Zuweisung oder Bausteinaufruf

Eine Aktion besteht aus folgenden Komponenten:

- Interlock (optional)
  Sie können eine Aktion mit einem Interlock verknüpfen, um die Ausführung der Aktion zu beeinflussen.
- Ereignis (optional)
  Ein Ereignis definiert, wann eine Aktion ausgeführt werden soll. Für einige Kennungen müssen Sie ein Ereignis angeben.
- Kennung (obligatorisch)
  Die Kennung definiert die Art der auszuführenden Aktion, z. B. das Setzen oder Rücksetzen eines Operanden.
- Aktion (obligatorisch)
  Die Aktion bestimmt den Operanden, mit dem die Aktion ausgeführt wird.

Siehe auch

Standardaktionen (Seite 7696)

Ereignisabhängige Aktionen (Seite 7698)

Bausteinaufrufe (Seite 7701)

Zeiten (Seite 7702)

Zähler (Seite 7704)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Aktionen programmieren (Seite 7773)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4.2 Standardaktionen

Einführung

Standardaktionen werden ausgeführt, solange ein Schritt aktiv ist. Die folgende Tabelle zeigt die möglichen Standardaktionen mit ihren Kennungen:

Kennung	Datentyp des Operanden	Bedeutung
N - Setzen solange Schritt aktiv	BOOL FB, FC, SFB, SFC	Solange der Schritt aktiv ist, ist der Signalzustand des Operanden "1". Solange der Schritt aktiv ist, wird der angegebene Baustein aufgerufen Der Schritt gilt auch in dem Zyklus als aktiv, in dem das S1 Ereignis auftritt.
S - Auf 1 setzen	BOOL	Sobald der Schritt aktiv ist, wird der Operand auf "1" gesetzt und bleibt anschließend auf "1".
R - Auf 0 setzen	BOOL	Sobald der Schritt aktiv ist, wird der Operand auf "0" gesetzt und bleibt anschließend auf "0".
D - Einschaltverzögerung	BOOL und Time/DWORD	n Sekunden nach der Schrittaktivierung wird der Operand auf "1" gesetzt und bleibt für die Dauer der Schrittaktivierung auf "1". Das gilt nicht, wenn die Dauer der Schrittaktivierung kürzer als n Sekunden ist. Die Zeit können Sie entweder als Konstante oder als PLC-Variable vom Datentyp TIME/DWORD angeben.
L - Setzen für limitierte Zeitdauer	BOOL und Time/DWORD	Wenn der Schritt aktiv ist, wird der Operand für n Sekunden auf "1" gesetzt. Danach wird der Operand zurückgesetzt. Der Operand wird auch dann zurückgesetzt, wenn die Schrittaktivierungszeit kleiner als die Zeitdauer ist. Die Zeit können Sie entweder als Konstante oder als PLC-Variable vom Datentyp TIME/DWORD angeben.

Alle diese Standardaktionen können Sie mit einem Interlock verknüpfen. Ist eine Aktion mit einem Interlock verknüpft, wird die Aktion nur ausgeführt, wenn die Bedingungen des Interlocks erfüllt sind.

Die folgende Tabelle zeigt einige Beispiele für die Verwendung von Zeitkonstanten:

Kennung	Aktion	Beschreibung
D	"MyTag", T#2s	2 Sekunden nach Schrittaktivierung wird der Operand "MyTag" auf "1" gesetzt und bleibt für die Dauer der Schrittaktivierung auf "1". Das gilt nicht, wenn die Dauer der Schrittaktivierung kürzer als 2 Sekunden ist. Mit Schrittdeaktivierung wird der Operand zurückgesetzt (nicht speichernd).
L	"MyTag", T#20s	Ist der Schritt aktiv, wird der Operand "MyTag" für 20 Sekunden auf "1" gesetzt. Danach wird der Operand zurückgesetzt (nicht speichernd). Der Operand wird auch dann zurückgesetzt, wenn die Schrittaktivierungszeit kleiner als 20 Sekunden ist.

Siehe auch

Grundlagen zu Aktionen (Seite 7695)

Ereignisabhängige Aktionen (Seite 7698)

Bausteinaufrufe (Seite 7701)

Zeiten (Seite 7702)

Zähler (Seite 7704)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Aktionen programmieren (Seite 7773)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4.3 Ereignisabhängige Aktionen

Einführung

Sie haben die Möglichkeit eine Aktion mit einem Ereignis zu verknüpfen, um die Ausführung der Aktion von bestimmten Bedingungen abhängig zu machen. Für Aktionen mit den Kennungen "D", "L" und "TF" ist die Verknüpfung mit Ereignissen nicht möglich. Ist eine Aktion mit einem Ereignis verknüpft, wird der Signalzustand des Ereignisses durch Flankenauswertung erfasst. Das bedeutet, dass die Anweisungen nur in dem Zyklus ausgeführt werden, in dem das Ereignis auftritt.

Die folgende Tabelle zeigt die Ereignisse, die Sie mit Aktionen verknüpfen können:

Ereignis	Signalauswertung	Beschreibung
S1	Steigende Flanke	Schritt wird aktiviert (Signalzustand = "1")
S0	Fallende Flanke	Schritt wird deaktiviert (Signalzustand = "0")
V1	Steigende Flanke	Supervision ist erfüllt, d. h. Störung tritt auf (Signalzustand = "1")
V0	Fallende Flanke	Supervision ist nicht mehr erfüllt, d. h. Störung ist behoben (Signalzustand = "0")
L0	Steigende Flanke	Interlock ist erfüllt, d. h. Störung ist behoben (Signalzustand = "1")
L1	Fallende Flanke	Interlock ist nicht erfüllt, d. h. Störung tritt auf (Signalzustand = "0")
A1	Steigende Flanke	Eine Meldung wird quittiert.
R1	Steigende Flanke	Eine Registrierung kommt.

Die Signalzustände eines Ereignisses können zum Programmieren weiterer Aktionen verwendet werden. Dadurch können Sie nicht nur einzelne Schritte, sondern die gesamte Ablaufsteuerung überwachen und beeinflussen.

Aktionen, für die Sie die Kennungen "ON" (Schritt aktivieren) und "OFF" (Schritt deaktivieren) verwenden, müssen immer mit einem Ereignis verknüpft werden. Das Ereignis bestimmt den Zeitpunkt der Aktivierung bzw. der Deaktivierung des Schritts. Falls ein Schritt innerhalb eines Zyklus sowohl aktiviert als auch deaktiviert wird, hat das Deaktivieren die höhere Priorität.

Aktionen, für die Sie die Ereignisse "S1", "V1", "A1" oder "R1" verwenden, können Sie zusätzlich mit einem Interlock verbinden. Dadurch werden sie nur ausgeführt, wenn die Bedingungen des Interlocks erfüllt sind.

Das folgende Bild zeigt ein Beispiel für einen Schritt, der ereignisabhängige Aktionen enthält:

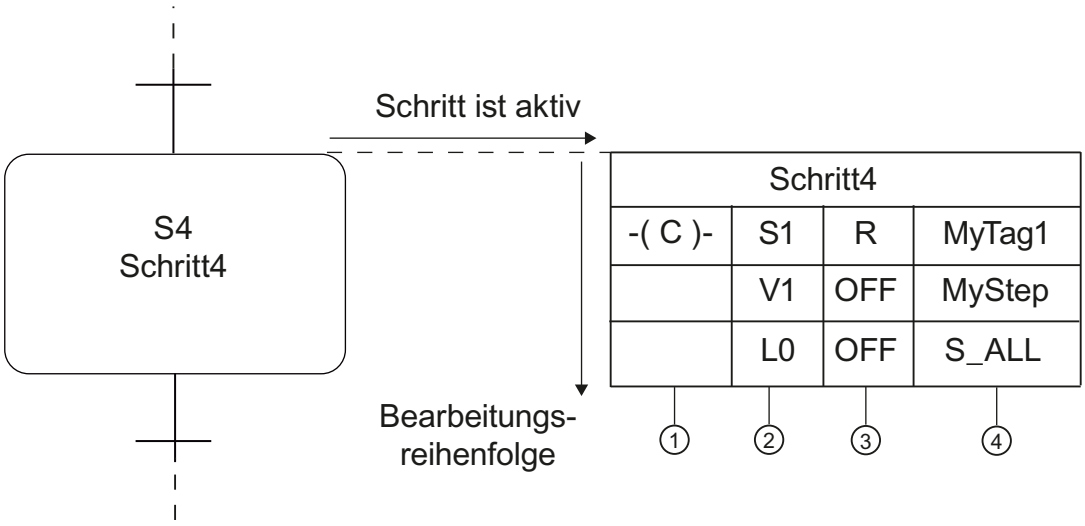

Sobald der Schritt "Schritt4" aktiv wird und der Interlock erfüllt ist, wird "MyTag1" auf "0" gesetzt und bleibt danach auf "0". Wenn ein Supervisionsfehler auftritt, wird der Schritt "MyStep" deaktiviert. Wenn ein Interlockfehler auftritt, werden alle Schritte deaktiviert.

Registrierung

Eine Registrierung ist ein Ereignis, das außerhalb des Bausteins angestoßen wird und die über eine positive Signalflanke an einem der Eingangsparameter "REG_S" oder "REG_EF" abgefragt werden. Wenn die Registrierung über den Eingangsparameter "REG_S" erfolgt, dann wird das Ereignis nur an den aktiven Schritt weitergegeben, der am Ausgangsparameter "S_NO" angezeigt wird. Wenn die Registrierung über den Eingangsparameter "REG_EF" erfolgt, dann wird das Ereignis an alle momentan aktiven Schritte weitergegeben.

Beispiele für die Signalauswertung von Ereignissen

Das folgende Bild zeigt die Signalauswertung für einen Schritt:

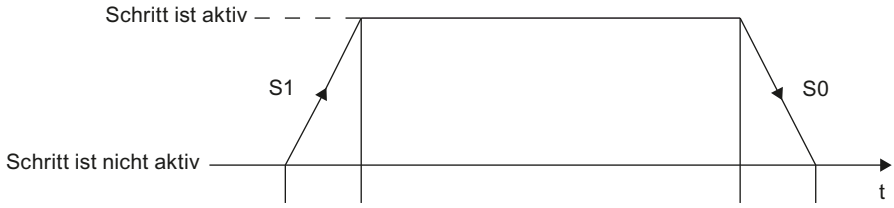

- S1: Schritt wird aktiv
- S0: Schritt wird inaktiv

Das folgende Bild zeigt die Signalauswertung für eine Supervision:

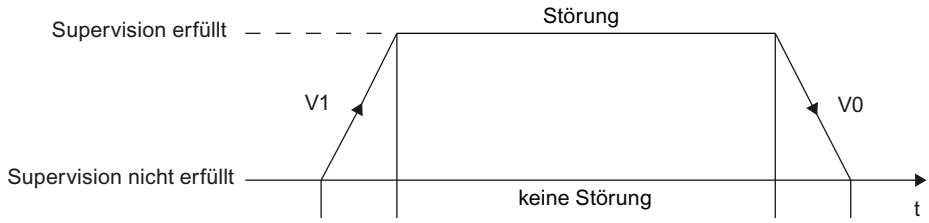

- V1: Supervisionsfehler tritt auf
- V0: Supervisionsfehler ist behoben

Das folgende Bild zeigt die Signalauswertung für einen Interlock:

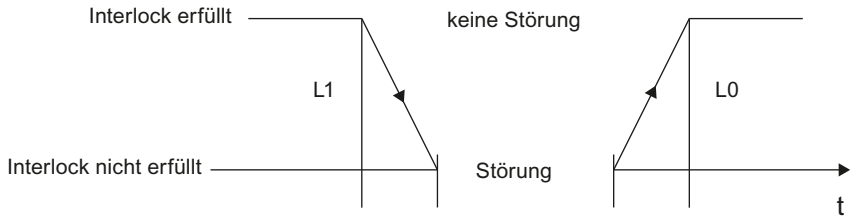

- L1: Interlock nicht mehr erfüllt (Störung kommt)
- L0: Interlock erfüllt (Störung geht)

Das folgende Bild zeigt die Signalauswertung für eine Meldung und die Registrierung:

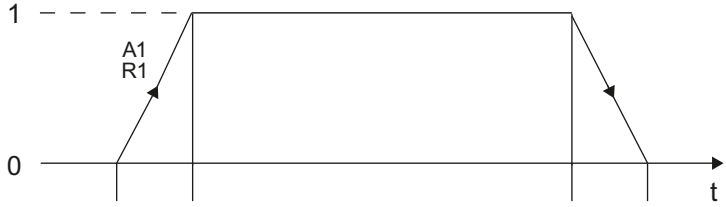

- A1: Eine Meldung wird quittiert
- R1: Eine Registrierung kommt (steigende Flanke am Eingang REG_EF/REG_S)

Siehe auch

Grundlagen zu Aktionen (Seite 7695)
Standardaktionen (Seite 7696)
Bausteinaufrufe (Seite 7701)
Zeiten (Seite 7702)
Zähler (Seite 7704)
Programmiersprache GRAPH (Seite 7685)
Permanente Anweisungen (Seite 7705)
Bedingungen (Seite 7706)
Interlock- und Supervisionsmeldungen (Seite 7706)

Aktionen programmieren (Seite 7773)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4.4 Bausteinaufrufe

Einführung

Innerhalb von Aktionen können Sie andere Funktionsbausteine und Funktionen aufrufen, in denen Sie bestimmte Teilaufgaben durchführen. Dadurch können Sie Ihr Programm besser strukturieren.

Aufruf von Funktionsbausteinen (FB)

Für den Aufruf eines Funktiongsbausteins verwenden Sie folgende Syntax:
`CALL "<FBName>", "<DBName>" (Parameterliste)`

Die folgende Tabelle zeigt ein Beispiel für den Aufruf eines Funktionsbausteins:

Interlock	Ereignis	Kennung	Aktion
		N	CALL "FB_KOP", "FB_KOP_DB" (MyInOut := "MyTag2")

Über die Anweisung "CALL" wird der Funktionsbaustein "FB_KOP" aufgerufen. Die Anweisung "CALL" ist stets mit der Kennung "N" verbunden. Hinter dem Namen des aufgerufenen Bausteins steht der Name des Datenbausteins, der die Daten und Parameter des Funktionsbausteins enthält. In der Parameterliste wird dem Parameter "MyInOut" die Variable "MyTag2" zugewiesen.

Aufruf von Funktionen (FC)

Für den Aufruf einer Funktion verwenden Sie die folgende Syntax:
`CALL "<FCName>" (Parameterliste)`

Die folgende Tabelle zeigt ein Beispiel für den Aufruf einer Funktion:

Inter-lock	Ereignis	Kennung	Aktion
		N	CALL "FC_KOP" (MyInput := "MyTag1")

Über die Anweisung "CALL" wird die Funktion "FC_KOP" aufgerufen. In der Parameterliste wird dem Parameter "MyInput" die Variable "MyTag1" zugewiesen.

Siehe auch

Bausteinaufrufe (Seite 62)

Bausteinaufrufe manuell einfügen (Seite 7800)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7801)

Bausteinaufrufe in GRAPH aktualisieren (Seite 7803)
Grundlagen zu Aktionen (Seite 7695)
Standardaktionen (Seite 7696)
Ereignisabhängige Aktionen (Seite 7698)
Zeiten (Seite 7702)
Zähler (Seite 7704)
Programmiersprache GRAPH (Seite 7685)
Permanente Anweisungen (Seite 7705)
Bedingungen (Seite 7706)
Interlock- und Supervisionsmeldungen (Seite 7706)
Aktionen programmieren (Seite 7773)
Ablaufketten (Seite 7687)
Schritte und Transitionen (Seite 7692)
Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4.5 Zeiten

Einführung

Innerhalb von Aktionen können Sie Zeiten verwenden. Bis auf "TF" sind alle Zeiten von einem Ereignis abhängig, das den Aktivierungszeitpunkt der Zeit bestimmt. Die Zeit "TF" wird durch den Schritt selbst aktiviert.

Bei der Verwendung der Zeiten "TL", "TD" und "TF" müssen Sie die Zeitdauer angeben. Verwenden Sie dazu die folgende Syntax:

```
<Zeitdauer> = EWy, AWy, MWy, LWy, DBWy oder DIWy; Variable vom Typ
S5TIME oder WORD; S5T#Konstante
```

y = 0 ... 65534

Sie können die Konstanten auch als normale Zeitangaben eingeben, z. B. als "5s".

Die folgende Tabelle zeigt die Zeiten, die Sie in Aktionen verwenden können:

Ereignis	Kennung	Beschreibung
S1, S0, L1, L0, V1, V0, A1, R1	TL	Verlängerter Impuls: Sobald das definierte Ereignis auftritt, startet die Zeit. Während der angegebenen Zeitdauer hat der Zeitstatus den Signalzustand "1". Nach Ablauf der Zeit erhält der Zeitstatus den Signalzustand "0".
S1, S0, L1, L0, V1, V0, A1, R1	TD	Speichernde Einschaltverzögerung: Sobald das definierte Ereignis auftritt, startet die Zeit. Während der angegebenen Zeitdauer hat der Zeitstatus den Signalzustand "0". Nach Ablauf der Zeit erhält der Zeitstatus den Signalzustand "1".
S1, S0, L1, L0, V1, V0, A1, R1	TR	Zeit anhalten und rücksetzen: Sobald das definierte Ereignis auftritt, wird die Zeit gestoppt. Zeitstatus und Zeitwert werden auf "0" zurückgesetzt.
-	TF	Ausschaltverzögerung: Sobald der Schritt aktiviert wird, wird der Zeitstatus auf "1" gesetzt. Mit Deaktivierung des Schritts läuft die Zeit und erst nach Ablauf der Zeit ist der Zeitstatus "0".

Siehe auch

Grundlagen zu Aktionen (Seite 7695)

Standardaktionen (Seite 7696)

Ereignisabhängige Aktionen (Seite 7698)

Bausteinaufrufe (Seite 7701)

Zähler (Seite 7704)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Aktionen programmieren (Seite 7773)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.4.6 Zähler

Einführung

Innerhalb von Aktionen können Sie Zähler einsetzen. Zähler benötigen immer ein Ereignis, um den Aktivierungszeitpunkt des Zählers festzulegen. Das bedeutet, der Zähler wird dann aktiviert, wenn das zugehörige Ereignis eintritt. Aktionen, für die Sie die Ereignisse S1, V1, A1 oder R1 verwenden, können Sie zusätzlich mit einem Interlock verbinden. Dadurch werden sie nur ausgeführt, wenn die Bedingungen des Interlocks erfüllt sind.

Die folgende Tabelle zeigt die Zähler, die Sie in Aktionen verwenden können:

Ereignis	Kennung	Datentyp des Operanden	Bedeutung
S1, S0, L1, L0, V1, V0, A1, R1	CS	COUNTER	Zähleranfangswert setzen: Sobald das definierte Ereignis auftritt, wird der Zähler auf den angegebenen Zählwert gesetzt. Den Zählwert können Sie als Variable oder Konstante vom Datentyp WORD angeben (C#0 bis C#999).
S1, S0, L1, L0, V1, V0, A1, R1	CU	COUNTER	Vorwärts zählen: Sobald das definierte Ereignis auftritt, zählt der Zähler um "1" vorwärts. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Flanke nicht mehr erhöht.
S1, S0, L1, L0, V1, V0, A1, R1	CD	COUNTER	Rückwärts zählen: Sobald das definierte Ereignis auftritt, zählt der Zähler um "1" rückwärts. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Flanke nicht mehr verringert.
S1, S0, L1, L0, V1, V0, A1, R1	CR	COUNTER	Zähler rücksetzen: Sobald das definierte Ereignis auftritt, wird der Zähler auf "0" zurückgesetzt.

Siehe auch

Grundlagen zu Aktionen (Seite 7695)

Standardaktionen (Seite 7696)

Ereignisabhängige Aktionen (Seite 7698)

Bausteinaufrufe (Seite 7701)

Zeiten (Seite 7702)

Programmiersprache GRAPH (Seite 7685)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Aktionen programmieren (Seite 7773)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.5 Permanente Anweisungen

Einführung

Über permanente Anweisungen haben Sie die Möglichkeit Programmcode zu erstellen, der entweder vor oder nach der Bearbeitung der Ablaufkette ausgeführt wird. Dadurch können Sie Bedingungen und Bausteinaufrufe programmieren, die unabhängig von der Ablaufkette in jedem Zyklus bearbeitet werden.

Permanente Anweisungen können Sie mit den Programmiersprachen KOP und FUP erstellen, wobei Sie maximal 32 KOP- oder FUP-Anweisungen in einer permanenten Anweisung programmieren können. Innerhalb eines GRAPH-Funktionsbausteins können Sie bis zu 250 vorausgeschaltete und 250 nachgeschaltete permanente Anweisungen verwenden. Das Verknüpfungsergebnis der Bedingungen übergeben Sie einer Zuweisung, für die auch die Speicherfunktionen "Setzen" und "Rücksetzen" zur Verfügung stehen. Der verwendete Operand (z. B. ein Merker) kann in Transitions-, Interlock-, Supervisions- und anderen permanenten Netzwerken abgefragt werden.

Siehe auch

Ansicht "Vorausgeschaltete permanente Anweisungen" (Seite 7741)

Ansicht "Nachgeschaltete permanente Anweisungen" (Seite 7746)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.6 Bedingungen

Einführung

Bedingungen sind binäre Zustände des Prozesses, die als KOP- oder FUP-Elemente (z. B. Schließer, Öffner, Und-Verknüpfung, Oder-Verknüpfung, Vergleicher) im Kontaktplan bzw. im Funktionsplan entsprechend der booleschen Logik miteinander verknüpft werden können. Das Verknüpfungsergebnis (VKE) kann einzelne Aktionen eines Schritts, den ganzen Schritt, das Weiterschalten zum nächsten Schritt oder die gesamte Ablaufkette beeinflussen.

Mit Bedingungen programmieren Sie folgende Bestandteile Ihres GRAPH-Programms:

- Permanente Anweisungen
- Interlocks
- Supervisionen
- Transitionen

Je Netzwerk können Sie bis zu 32 Anweisungen programmieren, an denen maximal 32 Operanden verschaltet sind. Sie können sich zusätzliche Informationen über die verwendeten Variablen anzeigen lassen, indem Sie die Variableninformationen einblenden.

Siehe auch

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Interlock- und Supervisionsmeldungen (Seite 7706)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.7 Interlock- und Supervisionsmeldungen

Einführung

Über Interlocks und Supervisionen können Sie Störungen aufdecken. Um diese Störungen schnell beheben zu können, haben Sie die Möglichkeit Meldungen festzulegen, die die Störungen ausführlich beschreiben. Dazu aktivieren Sie zuerst die Meldungsgenerierung und passen anschließend die Meldungstexte an Ihre Bedürfnisse an. Zusätzlich können Sie festlegen, dass die Meldungen quittiert werden müssen.

Siehe auch

Meldungen festlegen (Seite 7806)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

17.1.8 Bausteinschnittstelle eines GRAPH-Funktionsbausteins

17.1.8.1 Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins

Schnittstellenparametersätze

In den Einstellungen "Extras > Einstellungen > PLC-Programmierung > GRAPH" können Sie zwischen den folgenden Schnittstellenparametersätzen wählen, die Ihnen eine unterschiedliche Menge an Ein- und Ausgangsparametern zur Verfügung stellen:

- Minimaler Schnittstellenparametersatz
  Der minimale Schnittstellenparametersatz enthält nur den Eingangsparameter "INIT_SQ" und keine Ausgangsparameter.
- Standard-Schnittstellenparametersatz
  Der Standard-Schnittstellenparametersatz ermöglicht den Betrieb der Ablaufkette in verschiedenen Betriebsarten und gibt die Möglichkeit, Meldungen zu quittieren.
- Maximaler Schnittstellenparametersatz
  Der maximale Schnittstellenparametersatz stellt zusätzliche Diagnosemöglichkeiten zur Verfügung.

Sie können in allen Parametersätzen einzelne Parameter manuell löschen oder hinzufügen.

Hinweis

Beachten Sie bei der Auswahl auch den verfügbaren Speicherplatz in Ihrer CPU, da mit der Anzahl der Parameter auch der Speicherbedarf für den GRAPH-Funktionsbaustein und den zugehörigen Instanz-Datenbaustein steigt.

Siehe auch:

Eingangsparameter (Seite 7710)

Ausgangsparameter (Seite 7714)

Speicherplatzmodelle für den Instanz-Datenbaustein (S7-300/400)

Für CPUs der Baureihe S7-300/400 können Sie zusätzlich das Speicherplatzmodell des Instanz-Datenbausteins festlegen. Sie haben folgende Möglichkeiten:

- Standard-Datenbaustein
- Speicherplatzminimierter Datenbaustein

Über die Auswahl des Speichermodells definieren Sie die Anzahl der statischen Parameter. Grundsätzlich werden bei beiden Speichermodellen Informationen zu den einzelnen Schritten und Transitionen als statische Parameter abgelegt. Für jeden Schritt und jede Transition wird eine eigene Struktur mit detaillierten Informationen angelegt. In einem Standard-Datenbaustein erhalten Sie dabei ausführlichere Informationen als in einem speicherplatzminimierten Datenbaustein, dafür benötigt ein Standard-Datenbaustein jedoch deutlich mehr Speicherplatz.

In einem speicherplatzminimierten Datenbaustein gelten folgende Einschränkungen:

- Einige Elemente der Schritt- und Transitionsstrukturen sind nicht verfügbar.
- Die Option "Schritte überspringen" ist nicht verfügbar.
- Es sind keine Meldungen verfügbar.
- Schritt- und Transitionsnummern müssen fortlaufend sein. Ist dies nicht der Fall, werden die Schritte und Transitionen beim Übersetzen automatisch umnummeriert.
- Es werden keine Daten für die Kriterienanalyse erzeugt.

Siehe auch: Statische Parameter (Seite 7718)

Zugriff auf die Schnittstellenparameter

Grundsätzlich können Sie bei einer S7-1500-CPU auf die GRAPH-internen Schnittstellenparameter nicht schreibend zugreifen. Die folgenden Komponenten der Struktur "RT_DATA > MOP" im Abschnitt "Static", die die Betriebsarten verwalten, erlauben jedoch einen schreibenden Zugriff:

- AUTO
- MAN
- TAP
- TOP
- LOCK
- SUP
- OFF
- INIT
- HALT
- TMS_HALT
- OPS_ZERO
- S_PREV
- S_NEXT

- S_ON
- S_OFF
- T_PUSH
- ACK
- IL_PERM
- T_PERM
- ILP_MAN

Diese Parameter reagieren auf eine steigende Signalflanke. Vermeiden Sie für S7-300/400-CPUs ebenfalls den schreibenden Zugriff auf andere als die genannten Parameter, da sonst die fehlerfreie Bearbeitung der Ablaufkette nicht sichergestellt ist.

Siehe auch: Weitere statische Parameter (Seite 7721)

Zurücksetzen von Registrierungsanforderungen

Bei der Registrierung von Fehlern und Störungen über den Eingangsparameter "REG_EF" gibt es zwischen den CPU-Familien folgende Unterschiede beim Zurücksetzen der Registrierungsanforderung:

- S7-300/400
  Die Registrierungsanforderung bleibt solange erhalten bis in einem Schritt ein Interlock- oder Supervisionsfehler auftritt. Erst dann wird das Ereignis "R1" ausgelöst, die damit verknüpfte Aktion ausgeführt und die Registrierungsanforderung zurückgesetzt.
- S7-1500
  Die Registrierungsanforderung wird immer zurückgesetzt, unabhängig vom Auftreten eines Interlock- oder Supervisionsfehlers im aktuellen Zyklus.

Siehe auch

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Aktionen (Seite 7695)

Schritte und Transitionen (Seite 7692)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

17.1.8.2 Eingangsparameter

Übersicht über die Eingangsparameter eines GRAPH-Funktionsbausteins

Die Eingangsparameter eines GRAPH-Funktionsbausteins reagieren auf eine steigende Signalflanke. Die folgende Tabelle zeigt eine Übersicht über die Eingangsparameter eines GRAPH-Funktionsbausteins:

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
OFF_SQ	BOOL	OFF_SEQUENCE: Ablaufkette ausschalten, d.h. alle Schritte deaktivieren Typ: Anforderung	-	x	x
INIT_SQ	BOOL	INIT_SEQUENCE: Initialschritte aktivieren Ablaufkette rücksetzen Typ: Anforderung	x	x	x
ACK_EF	BOOL	ACKNOWLEDGE_ERROR_FAULT: Quittieren einer Störung, Weiterschalten erzwingen Typ: Anforderung	-	x	x
REG_EF	BOOL	REGISTRATE_ERROR_FAULT: Alle Fehler und Störungen registrieren Typ: Anforderung	-	-	x
ACK_S	BOOL	ACKNOWLEDGE_STEP: In Ausgangsparameter "S_NO" angezeigten Schritt quittieren Typ: Anforderung	-	-	x
REG_S	BOOL	REGISTRATE_STEP: In Ausgangsparameter "S_NO" angezeigten Schritt registrieren Typ: Anforderung	-	-	x
HALT_SQ	BOOL	HALT_SEQUENCE: Ablaufkette anhalten/wieder aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
HALT_TM	BOOL	HALT_TIMES: Alle Schrittaktivierungszeiten und zeitabhängige Operationen (L und D) der Ablaufkette anhalten/wieder aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
ZERO_OP	BOOL	ZERO_OPERANDS: Alle Operanden mit der Kennung N, D, L in aktiven Schritten auf Null zurücksetzen und CALL-Anweisungen in Aktionen nicht ausführen/Operanden und CALL-Anweisungen wieder aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
EN_IL	BOOL	ENABLE_INTERLOCKS: Interlock deaktivieren (Ablaufkette verhält sich wie bei erfülltem Interlock)/wieder aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
EN_SV	BOOL	ENABLE_SUPERVISIONS: Supervision deaktivieren (Ablaufkette verhält sich wie bei nicht erfüllter Supervision)/wieder aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
EN_ACKREQ	BOOL	ENABLE_ACKNOWLEDGE_REQUIRED: Quittierpflicht aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
EN_SSKIP	BOOL	ENABLE_STEP_SKIPPING: Schritt überspringen aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
DISP_SACT	BOOL	DISPLAY_ACTIVE_STEPS: Nur aktive Schritte anzeigen Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an DISP_SEF, DISP_SALL	-	-	x
DISP_SEF	BOOL	DISPLAY_STEPS_WITH_ERROR_OR_FAULT: Nur fehlerhafte und gestörte Schritte anzeigen Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an DISP_SACT, DISP_SALL	-	-	x
DISP_SALL	BOOL	DISPLAY_ALL_STEPS: Alle Schritte anzeigen Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an DISP_SACT, DISP_SEF	-	-	x
S_PREV	BOOL	PREVIOUS_STEP:Automatischer Betrieb: Rückwärtsblättern durch die derzeit aktiven Schritte, Anzeige der Schrittnummer in Ausgangsparameter "S_NO" Handbetrieb: vorhergehenden Schritt in "S_NO" anzeigen (kleinere Nummer) Typ: Anforderung	-	x	x

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
S_NEXT	BOOL	NEXT_STEP: Automatischer Betrieb: Vorwärtsblättern durch die derzeit aktiven Schritte, Anzeige der Schrittnummer in Ausgangsparameter "S_NO" Handbetrieb: nächsten Schritt in "S_NO" anzeigen (größere Nummer) Typ: Anforderung	-	x	x
SW_AUTO	BOOL	SWITCH_MODE_AUTOMATIC: Betriebsarten-Umschaltung: Automatischer Betrieb Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an SW_TAP, SW_TOP, SW_MAN	-	x	x
SW_TAP	BOOL	SWITCH_MODE_TRANSITION: Betriebsarten-Umschaltung: Halbautomatischer Betrieb Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an SW_AUTO, SW_TOP, SW_MAN	-	x	x
SW_TOP	BOOL	SWITCH_MODE_TRANSITION_OR_PUSH: Betriebsarten-Umschaltung: Automatischer oder halbautomatischer Betrieb Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an SW_AUTO, SW_TAP, SW_MAN	-	x	x
SW_MAN	BOOL	SWITCH_MODE_MANUAL: Betriebsarten-Umschaltung: Handbetrieb, es wird kein selbständiger Ablauf angestoßen Typ: Zustand, das Zurücksetzen erfolgt durch eine positive Flanke an SW_AUTO, SW_TAP, SW_TOP	-	x	x
S_SEL	INT	STEP_SELECT: Schrittnummer für Ausgangsparameter "S_NO" anwählen im Handbetrieb, aktivieren/deaktivieren mit "S_ON", "S_OFF" Typ: N/A	-	x	x
S_SELOK	BOOL	STEP_SELECT_OK: Wert in "S_SEL" für Ausgangsparameter "S_NO" übernehmen Typ: Anforderung	-	-	x
S_ON	BOOL	STEP_ON: Handbetrieb: Angezeigten Schritt aktivieren Typ: Anforderung	-	x	x
S_OFF	BOOL	STEP_OFF: Handbetrieb: Angezeigten Schritt deaktivieren Typ: Anforderung	-	x	x

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
T_PREV	BOOL	PREVIOUS_TRANSITION: Vorherige gültige Transition in Ausgangsparameter "T_NO" anzeigen Typ: Anforderung	-	-	x
T_NEXT	BOOL	NEXT_TRANSITION: Nächste gültige Transition in Ausgangsparameter "T_NO" anzeigen Typ: Anforderung	-	-	x
T_PUSH	BOOL	PUSH_TRANSITION: Transition schaltet, wenn Bedingung erfüllt und "T_PUSH" (Flanke) Voraussetzung: Automatischer Betrieb oder Handbetrieb Typ: Anforderung	-	x	x
EN_LMODE (nur S7-1500 ab Bausteinversion V2.0)	BOOL	ENABLE_LEARNING_MODE: Lernmodus aktivieren Typ: Zustand, das Zurücksetzen erfolgt durch eine erneute positive Flanke	-	-	x
RESET_CRIT (nur S7-1500; GRAPH-FB ab Bausteinversion V5.0)	BOOL	RESET_CRITERIA: Erstwerte von Interlocks und Transitionen zurücksetzen Typ: Anforderung	-	-	x

Siehe auch

Erstwerte zurücksetzen (Seite 7811)

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Ausgangsparameter (Seite 7714)

Statische Parameter (Seite 7718)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

17.1.8.3 Ausgangsparameter

Übersicht über die Ausgangsparameter eines GRAPH-Funktionsbausteins

Die folgende Tabelle zeigt eine Übersicht über die Ausgangsparameter eines GRAPH-Funktionsbausteins:

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
S_NO	INT	STEP_NUMBER: Anzeige Schrittnummer	-	x	x
S_MORE	BOOL	MORE_STEPS: Weitere Schritte sind aktiv	-	x	x
S_ACTIVE	BOOL	STEP_ACTIVE: Angezeigter Schritt ist aktiv	-	x	x
S_TIME	TIME	STEP_TIME: Schritt-Aktivierungszeit	-	-	x
S_TIMEOK	TIME	STEP_TIME_OK: Schritt-Aktivierungszeit fehlerfrei	-	-	x
S_CRITLOC (S7-300/400, bzw. S7-1500 ab Bausteinversion V4.0)	DWORD	STEP_CRITERIA_INTERLOCK: Interlock-Kriterienbits	-	-	x
S_CRITLOCERR (S7-300/400, bzw. S7-1500 ab Bausteinversion V4.0)	DWORD	S_CRITERIA_IL_LAST_ERROR: Interlock-Kriterienbits bei Ereignis L1	-	-	x
S_CRITSUP (nur S7-300/400)	DWORD	STEP_CRITERIA_SUPERVISION: Supervisions-Kriterienbits	-	-	x
S_STATE	WORD	STEP_STATE: Schrittzustandsbits	-	-	x
T_NO	INT	TRANSITION: Gültige Transitionsnummer	-	-	x

Parameter	Datentyp	Beschreibung	Minimaler Parameter-satz	Standard-Pa-rametersatz	Maximaler Pa-rametersatz
T_MORE	BOOL	MORE_TRANSITIONS: Weitere gültige Transitionen zur Anzeige verfügbar	-	-	x
T_CRIT (S7-300/400, bzw. S7-1500 ab Bausteinversion V4.0)	DWORD	TRANSITION_CRITERIA: Status der max. 32 KOP/FUP-Elemente der Transitions-Kriterienbits im aktuellen Bearbeitungszyklus Weitere Informationen zur Reihenfolge der Kriterienbits im DWORD finden Sie hier: Beispiel für die Reihenfolge der Auflistung (Seite 7812)	-	-	x
T_CRITOLD (S7-300/400, bzw. S7-1500 ab Bausteinversion V4.0)	DWORD	T_CRITERIA_LAST_CYCLE: Status der max. 32 KOP/FUP-Elemente der Transitions-Kriterienbits vom letzten Zyklus Weitere Informationen zur Reihenfolge der Kriterienbits im DWORD finden Sie hier: Beispiel für die Reihenfolge der Auflistung (Seite 7812)	-	-	x
T_CRITFLT (S7-300/400, bzw. S7-1500 ab Bausteinversion V4.0)	DWORD	T_CRITERIA_LAST_FAULT: Status der max. 32 KOP/FUP-Elemente der Transitions-Kriterienbits bei Ereignis V1 aus dem Bearbeitungszyklus, als der Supervisionsfehler im vorherigen Schritt aufgetreten ist. Weitere Informationen zur Reihenfolge der Kriterienbits im DWORD finden Sie hier: Beispiel für die Reihenfolge der Auflistung (Seite 7812)	-	-	x
ERROR	BOOL	INTERLOCK_ERROR: Interlockfehler (beliebiger Schritt)	-	-	x
FAULT	BOOL	SUPERVISION_FAULT: Supervisionsfehler (beliebiger Schritt)	-	-	x

Parameter	Datentyp	Beschreibung	Minimaler Parameter-satz	Standard-Parametersatz	Maximaler Parametersatz
ERR_FLT	BOOL	IL_ERROR_OR_SV_FAULT: Sammelstörung	-	x	x
SQ_ISOFF	BOOL	SEQUENCE_IS_OFF: Ablaufkette ist ausgeschaltet (kein Schritt aktiv)	-	-	x
SQ_HALTED	BOOL	SEQUENCE_IS_HALTED: Ablaufkette angehalten	-	-	x
TM_HALTED	BOOL	TIMES_ARE_HALTED: Zeiten angehalten	-	-	x
OP_ZEROED	BOOL	OPERANDS_ARE_ZEROED: Operanden zurückgesetzt	-	-	x
IL_ENABLED	BOOL	INTERLOCK_IS_ENABLED: Interlock wird berücksichtigt	-	-	x
SV_ENABLED	BOOL	SUPERVISION_IS_ENABLED: Supervision wird berücksichtigt	-	-	x
ACKREQ_ENABLED	BOOL	ACKNOWLEDGE_REQUIRED_IS_ENABLED: Quittierpflicht ist aktiviert	-	-	x
SSKIP_ENABLED	BOOL	STEP_SKIPPING_IS_ENABLED: Schritt überspringen ist aktiviert	-	-	x
SACT_DISP	BOOL	ACTIVE_STEPS_WERE_DISPLAYED: Nur aktive Schritte in "S_NO" anzeigen	-	-	x
SEF_DISP	BOOL	STEPS_WITH_ERROR_FAULT_WERE_DISPLAYED: Nur fehlerhafte und gestörte Schritte in "S_NO" anzeigen	-	-	x
SALL_DISP	BOOL	ALL_STEPS_WERE_DISPLAYED: Alle Schritte in "S_NO" anzeigen	-	-	x
AUTO_ON	BOOL	AUTOMATIC_IS_ON: Anzeige Betriebsart Automatik	-	x	x

Parameter	Datentyp	Beschreibung	Minimaler Parametersatz	Standard-Parametersatz	Maximaler Parametersatz
TAP_ON	BOOL	T_AND_PUSH_IS_ON: Anzeige Halbautomatischer Betrieb	-	x	x
TOP_ON	BOOL	T_OR_PUSH_IS_ON: Anzeige Halbautomatischer Betrieb	-	x	x
MAN_ON	BOOL	MANUAL_IS_ON: Anzeige Handbetrieb	-	x	x
LMODE_ENABLED (nur S7-1500 ab Bausteinversion V2.0)	BOOL	LEARNING_MODE_IS_ENABLED: Lernmodus aktiviert·	-	-	x

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Eingangsparameter (Seite 7710)

Statische Parameter (Seite 7718)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

17.1.8.4 Statische Parameter

Statische Parameter eines Schritts

Übersicht über die statischen Parameter eines Schritts

Die folgende Tabelle zeigt die Parameter, die für jeden Schritt des GRAPH-Funktionsbausteins in der Bausteinschnittstelle im Abschnitt "Static" angelegt werden:

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
S1	Schritt wird aktiviert	BOOL	x	x
L1	Interlock geht	BOOL	x	x
V1	Supervision kommt	BOOL	x	x
R1	Reserviert	BOOL	x	x
A1	Fehler wird quittiert	BOOL	x	x
S0	Schritt wird deaktiviert	BOOL	x	x
L0	Interlock kommt	BOOL	x	x
V0	Supervision geht	BOOL	x	x
X	Schritt ist aktiv	BOOL	x	x
LA	Interlock ist nicht erfüllt	BOOL	x	x
VA	Supervision steht an	BOOL	x	x
RA	Reserviert	BOOL	x	x
AA	Reserviert	BOOL	x	x
SS	Systemintern	BOOL	x	x
LS	Direktes Ergebnis des programmierten Interlocks	BOOL	x	x
VS	Direktes Ergebnis der programmierten Supervision	BOOL	x	x
SNO	Anwenderschrittnummer	INT	-	x
T	Gesamte Schrittaktivierungszeit	TIME	x	x
U	Ungestörte Schrittaktivierungszeit	TIME	-	x
T_MAX (nur S7-1500 ab Bausteinversion V2.0)	Maximale Schrittaktivierungszeit	TIME	-	x
T_WARN (nur S7-1500 ab Bausteinversion V2.0)	Warnungszeit	TIME	-	x
CRIT_LOC (S7-300/400) (S7-1500 ab Bausteinversion V4.0)	Status der max. 32 KOP/FUP-Elemente in Interlock im aktuellen Bearbeitungszyklus	DWORD	-	x

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
CRIT_LOC_ERR (nur S7-300/400) (S7-1500 ab Bausteinversion V4.0)	Kopie von CRIT_LOC, wenn Interlock geht	DWORD	-	x
CRIT_SUP (nur S7-300/400)	Status der max. 32 KOP/FUP-Elemente in Supervision im aktuellen Bearbeitungszyklus	DWORD	-	x
SM	Systemintern	BOOL	-	x
LP (nur S7-300/400)	Systemintern	BOOL	x	x
LN (nur S7-300/400)	Systemintern	BOOL	x	x
VP (nur S7-300/400)	Systemintern	BOOL	x	x
VN (nur S7-300/400)	Systemintern	BOOL	x	x
H_IL_ERR	Systemintern	BYTE	-	x
H_SV_FLT	Systemintern	BYTE	-	x
RESERVED (nur S7-300/400)	Reserve	DWORD	-	x

Zugriff auf die Komponenten

Da die Werte der Parameter den Ablauf der Kette beeinflussen, wird ein schreibender Zugriff nicht empfohlen. Lesender Zugriff erfolgt über den Schrittnamen gefolgt vom Namen des Strukturelements. In Bedingungen bzw. Aktionen sind damit z. B. folgende Zugriffe möglich:

- TROCKNEN.X
- TROCKNEN.T

Siehe auch

Erstwerte zurücksetzen (Seite 7811)

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Eingangsparameter (Seite 7710)

Ausgangsparameter (Seite 7714)

Statische Parameter einer Transition (Seite 7720)

Weitere statische Parameter (Seite 7721)
Schnittstellenparametersatz festlegen (Seite 7734)
Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)
Programmiersprache GRAPH (Seite 7685)
Ablaufketten (Seite 7687)
Schritte und Transitionen (Seite 7692)
Aktionen (Seite 7695)
Permanente Anweisungen (Seite 7705)
Bedingungen (Seite 7706)
Interlock- und Supervisionsmeldungen (Seite 7706)

Statische Parameter einer Transition

Übersicht über die statischen Parameter einer Transition

Die folgende Tabelle zeigt die Parameter, die für jede Transition des GRAPH-Funktionsbausteins in der Bausteinschnittstelle im Abschnitt "Static" angelegt werden:

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
TV	Transition ist gültig	BOOL	x	x
TT	Transition ist erfüllt	BOOL	x	x
TS	Transition schaltet	BOOL	x	x
CF_IV (nur S7-300/400)	Eintrag "CRIT_FLT" ist ungültig	BOOL	-	x
TNO	Anzeige der anwenderdefinierten Transitionsnummer	INT	-	x
CRIT (nur S7-300/400) (S7-1500 ab Bausteinversion V4.0)	Status der max. 32 KOP/FUP-Elemente der Transition im aktuellen Bearbeitungszyklus	DWORD	-	x
CRIT_OLD (nur S7-300/400) (S7-1500 ab Bausteinversion V4.0)	Status der max. 32 KOP/FUP-Elemente der Transition im vorherigen Bearbeitungszyklus Welches Element der Transition hat letztendlich die Weiterschaltung veranlasst.	DWORD	-	x
CRIT_FLT (nur S7-300/400) (S7-1500 ab Bausteinversion V4.0)	Kopie von "CRIT", wenn ein Fehler auftritt	DWORD	-	x

Zugriff auf die Komponenten

Da die Werte der Parameter den Ablauf der Kette beeinflussen, wird ein schreibender Zugriff nicht empfohlen. Lesender Zugriff erfolgt über den Transitionsnamen gefolgt vom Namen des Strukturelements. In Bedingungen bzw. Aktionen sind damit z. B. folgende Zugriffe möglich:

- VENTIL_EIN.TT
- VENTIL_EIN.CRIT

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Eingangsparameter (Seite 7710)

Ausgangsparameter (Seite 7714)

Statische Parameter eines Schritts (Seite 7718)

Weitere statische Parameter (Seite 7721)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Programmiersprache GRAPH (Seite 7685)

Ablaufketten (Seite 7687)

Schritte und Transitionen (Seite 7692)

Aktionen (Seite 7695)

Permanente Anweisungen (Seite 7705)

Bedingungen (Seite 7706)

Interlock- und Supervisionsmeldungen (Seite 7706)

Weitere statische Parameter

Übersicht

Bei einer S7-1500-CPU befinden sich die folgenden statischen Parameter in der Struktur "RT_DATA".

Zu den statischen Parametern gehören neben den Schritt- und Transitionsparametern auch die folgenden Informationen:

- Interne Parameter "VERSION", "S_DISPLAY", "S_SEL_OLD", "S_DISPIDX", "T_DISPIDX"
- Komponente "MOP >Betriebsart"
- Komponente "SQ_FLAGS > Kettenmerker"
- Komponente "Zeittakte"

Hinweis

In einem GRAPH-spezifischen Instanz-Datenbaustein sind weitere Parameter enthalten. Diese dienen jedoch lediglich der internen Verwendung. Es wird nicht empfohlen, diese Parameter im Programm zu verwenden.

Interne Parameter

Die folgende Tabelle zeigt die internen Parameter, die sich im Abschnitt "Static" befinden. Sie dienen zur Verwaltung der Schritt- und Transitionsnummern:

Komponente	Beschreibung	Datentyp	Länge in Byte	Speicherplatzminimierter DB	Standard-DB
VERSION	Bausteinversion	STRING[10]	2	x	x
S_DISPLAY	Kopie des Ausgangsparameters "S_NO"	INT	2	x	x
S_SEL_OLD	Schrittnummer aus dem Eingangsparameter "S_SEL" aus dem letzten CPU-Zyklus	INT	2	x	x
S_DISPIDX	Schrittindex für den Schritt in Ausgangsparameter "S_NO"	BYTE	1	x	x
T_DISPIDX	Transitionsindex für die Transition in Ausgangsparameter "T_NO"	BYTE	1	x	x

Betriebsart

Die folgende Tabelle zeigt die Komponenten der Struktur "MOP" im Abschnitt "Static", die die Betriebsarten verwalten:

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
AUTO	Wechseln in den Automatikbetrieb	BOOL	x	x
MAN	Wechseln in den Handbetrieb	BOOL	x	x
TAP	Wechseln in die Betriebsart "Tippen"	BOOL	x	x
TOP	Wechseln in die Betriebsart "Automatik" oder "Weiterschalten"	BOOL	x	x
ACK_S	Anforderung: Schritt quittieren, der am Ausgangsparameter "S_NO" anliegt	BOOL	x	x
REG_S	Anforderung: Schritt registrieren, der am Ausgangsparameter "S_NO" anliegt	BOOL	x	x

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
T_PREV	Anforderung: Vorherige gültige Transition am Ausgangsparameter "T_NO" ausgeben	BOOL	x	x
T_NEXT	Anforderung: Nächste gültige Transition am Ausgangsparameter "T_NO" ausgeben	BOOL	x	x
LOCK	Interlockbehandlung einschalten/ausschalten	BOOL	x	x
SUP	Supervisionbehandlung einschalten/ausschalten	BOOL	x	x
ACKREQ	Anforderung: Schritt quittieren, der an Ausgangsparameter "S_NO" anliegt	BOOL	x	x
SSKIP	Zustand: "Schritt überspringen" aktiviert	BOOL	x	x
OFF	Ablaufkette deaktivieren	BOOL	x	x
INIT	Initialschritte aktivieren	BOOL	x	x
HALT	Kette anhalten/weiterlaufen lassen	BOOL	x	x
TMS_HALT	Zeiten anhalten/weiterlaufen lassen	BOOL	x	x
OPS_ZERO	Aktionen abschalten/zuschalten	BOOL	x	x
SACT_DISP	Zustand: Nur aktive Schritte anzeigen	BOOL	x	x
SEF_DISP	Zustand: Nur fehlerhafte und gestörte Schritte anzeigen	BOOL	x	x
SALL_DISP	Zustand: Alle Schritte anzeigen	BOOL	x	x
S_PREV	Betriebsart "Automatik": vorhergehenden gleichzeitig aktiven Schritt in Ausgangsparameter "S_NO" anzeigen Betriebsart "Hand": nächst kleinere Schrittnummer anzeigen	BOOL	x	x
S_NEXT	Betriebsart "Automatik": nächsten gleichzeitig aktiven Schritt in Ausgangsparameter "S_NO" anzeigen Betriebsart "Hand": nächst größere Schrittnummer anzeigen	BOOL	x	x
S_SELOK	Anforderung: Schrittnummer, die am Eingangsparameter "S_SEL" anliegt, am Ausgangsparameter "S_NO" ausgeben	BOOL	x	x
S_ON	Betriebsart "Hand": angezeigten Schritt aktivieren	BOOL	x	x
S_OFF	Betriebsart "Hand": angezeigten Schritt deaktivieren	BOOL	x	x

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
T_PUSH	Betriebsart "Tippen": Transition schaltet, wenn die erfüllte Bedingung mit "T_PUSH" übernommen wird	BOOL	x	x
REG	Anforderung: Alle Interlock- und Supervisionsfehler registrieren	BOOL	x	x
ACK	Quittierung der Störung	BOOL	x	x
IL_PERM	Permanente Interlockbearbeitung	BOOL	x	x
T_PERM	Permanente Transitionsbearbeitung	BOOL	x	x
ILP_MAN	Permanente Interlockbearbeitung bei Betriebsart "Hand"	BOOL	x	x
LMODE (nur S7-1500, ab Bausteinversion V2.0)	Lernmodus ist aktiv	BOOL	-	-
RESET_CRIT(nur S7-1500; ab Bausteinversion V5.0)	RESET_CRITERIA: Erstwerte von Interlocks und Transitionen zurücksetzenTyp: Anforderung	BOOL	-	-

Zeittakte (nur S7-300/400)

Die folgende Tabelle zeigt die Komponenten der Struktur "TICKS" im Abschnitt "Static", die die Informationen zur Berechnung der Aktivierungszeit verwalten. Diese Informationen können nur systemintern ausgewertet werden.

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
DELTA	Zeitdifferenz zwischen zwei Zyklen	TIME	x	x
OLD	10 ms Taktzählerstand im letzten Zyklus	TIME	x	x
NEW	10 ms Taktzählerstand in diesem Zyklus	TIME	x	x

Kettenmerker

Die folgende Tabelle zeigt die Komponenten der Struktur "SQ_FLAGS" im Abschnitt "Static", die den internen Kettenzustand überwachen:

Komponente	Beschreibung	Datentyp	Speicherplatzminimierter DB	Standard-DB
ERR_FLT	Sammelstörung	BOOL	x	x
ERROR	Interlockfehler	BOOL	x	x
FAULT	Supervisionsfehler	BOOL	x	x
RT_FAIL	Laufzeitfehler	BOOL	x	x
NO_SNO	Selektierter Schritt existiert nicht	BOOL	x	x
NF_OFL	Zu viele zu aktivierende oder zu deaktivierende Schritte	BOOL	x	x
SA_OFL	Zu viele aktive Schritte	BOOL	x	x
TV_OFL	Zu viele gültige Transitionen	BOOL	x	x
MSG_OFL	Zu viele Meldungen	BOOL	x	x
NO_SWI	Nicht Schalten in diesem Zyklus	BOOL	x	x
CYC_OP	Zyklische Bearbeitung der Ablaufkette nach Initialisierung	BOOL	x	x
AS_MSG	Meldungen zur Laufzeit durch Anweisung aktiviert oder deaktiviert	BOOL	x	x
AS_SEND	Meldungen von "WR_USMSG" senden oder nur in Diagnosepuffer eintragen	BOOL	x	x
SQ_BUSY	Interner Flankenmerker für Vorgangsbearbeitung	BOOL	x	x
SA_BUSY	Interner Flankenmerker für Vorgangsbearbeitung	BOOL	x	x
AS_SIG	Flankenmerker für Meldungen von "Alarm_S" und "Alarm_SQ"	BOOL	x	x

Zugriff auf die Komponenten

Da die Werte der Parameter den Ablauf der Kette beeinflussen, wird ein schreibender Zugriff nicht empfohlen. Lesender Zugriff ist jedoch möglich.

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Eingangsparameter (Seite 7710)

Ausgangsparameter (Seite 7714)

Statische Parameter eines Schritts (Seite 7718)

Statische Parameter einer Transition (Seite 7720)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Programmiersprache GRAPH (Seite 7685)
Ablaufketten (Seite 7687)
Schritte und Transitionen (Seite 7692)
Aktionen (Seite 7695)
Permanente Anweisungen (Seite 7705)
Bedingungen (Seite 7706)
Interlock- und Supervisionsmeldungen (Seite 7706)

17.1.8.5 Standardverhalten von GRAPH erweitern

Grundlagen zum Erweitern des Standardverhaltens von GRAPH

Einführung

GRAPH-Funktionsbausteine enthalten in der Bausteinschnitte im Abschnitt "Static" sowohl sichtbare als auch unsichtbare interne Parameter. Sichtbare Parameter sind:

- Interner Datenbereich "RT_DATA"
- Transitionsstruktur "Trans1"
- Schrittstruktur "Step1"

Unsichtbare Parameter sind:

- Beschreibungs- und Status-Arrays

Bei Bedarf können Sie, wie in STEP 7 Klassik, auch auf die unsichtbaren Parameter zugreifen. Dafür muss der GRAPH-Baustein ab der Version 4.0 verwendet werden. Zusätzlich zum GRAPH-Funktionsbaustein ist ein SCL-Funktionsbaustein als Erweiterung notwendig. Zugriffe auf die einzelnen Arrays innerhalb des Erweiterungsbausteins erfolgen über Offsets, die in der "RT_DATA"-Struktur verfügbar sind.

Arrays

Über den Erweiterungsbaustein für GRAPH können Sie unter anderem auf die folgenden Arrays zugreifen:

Array	Beschreibung
SINI	Feld der Initialschritte
LSTT	Zuordnung von Absprungtransition zu Einsprungschritt
ATAJ	Zuordnung aller Alternativtransitionen einer Alternativzusammenführung
ATAB	Zuordnung aller Alternativtransitionen einer Alternativverzweigung
PSTT	Zuordnung von direktem Vorgängerschritt zu Transition
NSTT	Zuordnung von direktem Nachfolgeschritt zu Transition
ASSJ	Zuordnung aller Simultanschritte einer Simultanzusammenführung
ASSB	Zuordnung aller Simultanschritte einer Simultanverzweigung

Array	Beschreibung
PTTS	Zuordnung von direkter Vorgängertransition zu Schritt
NTTS	Zuordnung von direkter Nachfolgetransition zu Schritt
SW_SQTS	Zuordnung von Schritt zu Sequenzteil
SWITCH	Daten für gesichertes Schaltverhalten
TVX	Feld der gültigen Transitionen
TTX	Feld der erfüllten Transitionen
TSX	Feld der schaltenden Transitionen
S00X	Feld der zu deaktivierenden Schritte (vorletzter Zyklus)
SOFFX	Feld_1 der zu deaktivierende Schritte
SONX	Feld_1 der zu aktivierende Schritte
SAX	Feld der aktiven Schritte
SERRX	Feld der gestörten Schritte
SMX	Feld der markierten Schritte
S0X	Feld_2 der zu deaktivierende Schritte
S1X	Feld_2 der zu aktivierende Schritte

Erweiterungs-Funktionsbaustein erstellen und einem GRAPH-FB zuordnen

Sie können einen Erweiterungsbaustein erstellen und diesen dem GRAPH-Funktionsbaustein zuordnen, um auf die GRAPH-internen Parameter zuzugreifen. Der Erweiterungsbaustein kann dabei als Einzel- oder als Multiinstanz verwendet werden.

Voraussetzung

Das aktuelle Projekt ist ein V14 SP1-Projekt.

Vorgehen

Um einen Erweiterungsbaustein zu erstellen und einem GRAPH-FB zuzuordnen, gehen Sie folgendermaßen vor:

1. Erstellen Sie einen neuen Funktionsbaustein (FB) in der Programmiersprache "GRAPH".
2. Öffnen Sie die Bausteineigenschaften des neuen GRAPH-FB.
3. Stellen Sie unter "Allgemein > Version" die Version des GRAPH-FB auf >= V4.0.

   Hinweis

   Hinweis zur Projektkompatibilität

   Falls Ihr aktuelles Projekt noch kein V14 SP1-Projekt ist, erhalten Sie eine entsprechende Meldung angezeigt. Um über den Erweiterungsbaustein auf die GRAPH-internenParameter zugreifen zu können, muss Ihr Projekt auf ein V14 SP1-Projekt hochgerüstet werden. Nach dem Hochrüsten können Sie das Projekt jedoch nicht mehr mit einer TIA Portal Version kleiner als V14 SP1 bearbeiten.

4. Erstellen Sie einen neuen Funktionsbaustein (FB) in der Programmiersprache "SCL". Sie können diesen FB beliebig benennen. Dieser FB dient als Erweiterungsbaustein.

5. Tragen Sie im Erweiterungsbaustein in der Bausteinschnittstelle im Abschnitt "InOut" die folgenden Durchgangsparameter ein:
   - "io_RT_Data", Datentyp "G7_RTDataPlus_V4" (z. B. für die Bausteinversion V4.0)
   - "io_G7T", Datentyp "Array[*] of G7_TransitionPlus_V4" (z. B. für die Bausteinversion V4.0)
   - "io_G7S", Datentyp "Array[*] of G7_StepPlus_V4" (z. B. für die Bausteinversion V4.0)
   - "io_G7Arrays", Datentyp "Array[*] of USInt"
6. Öffnen Sie die Bausteineigenschaften des GRAPH-FB.
7. Tragen Sie unter "Attribute > Name des Erweiterungsbausteins" den Namen des SCL-Funktionsbausteins ein.
8. Instantiieren Sie den Erweiterungsbaustein entweder als Einzel- oder Multiinstanz:
   - Einzelinstanz:
     Erstellen Sie einen Instanz-Datenbaustein vom Typ des Erweiterungsbausteins. Tragen Sie im GRAPH-FB in der Bausteinschnittstelle im Abschnitt "InOut" einen Parameter ein, der als Datentyp den Erweiterungsbaustein hat.

     Hinweis

     Wenn Sie einen Erweiterungsbaustein für mehrere GRAPH-FBs verwenden möchten, tragen Sie den Erweiterungsbaustein in den Attributen der entsprechenden GRAPH-FBs ein. Erzeugen Sie jedoch für jeden GRAPH-FB einen eigenen Instanz-Datenbaustein des Erweiterungsbausteins. Um für die verschiedenen GRAPH-FBs unterschiedliches Verhalten zu definieren, können Sie den Abschnitt "Static" in der Bausteinschnittstelle des Erweiterungsbausteins nutzen.

   - Multiinstanz:
     Tragen Sie im GRAPH-FB in der Bausteinschnittstelle im Abschnitt "Static" einen Parameter ein, der als Datentyp den Erweiterungsbaustein hat.
9. Rufen Sie den GRAPH-FB in einem beliebigen anderen Baustein auf, dabei wird automatisch der Instanz-Datenbaustein für den GRAPH-FB erzeugt.
10. Wenn Sie den Erweiterungsbaustein als Einzelinstanz instantiiert haben, versorgen Sie den entsprechenden Parameter des Bausteinaufrufs, den Sie im Abschnitt "InOut" des GRAPH-FB definiert haben, mit dem Instanz-Datenbaustein des Erweiterungsbausteins.

Ergebnis

Der Erweiterungsbaustein ist soweit vorbereitet und Sie können innerhalb von ihm auf die GRAPH-internen Parameter zugreifen. Die einzelnen Beschreibungs- und Status-Arrays sind im Sammel-Array "io_G7Arrays" enthalten. Zugriffe auf sie erfolgen über die Offsets. Ein solcher Zugriff kann z. B. folgendermaßen aussehen:

```
#t_NumberOfActiveSteps :=
#io_G7Arrays[#io_RT_DATA.OFFSETS.SAX_OFFSET] - 1;
```

17.2 Einstellungen zu GRAPH

17.2.1 Übersicht über die Einstellungen zu GRAPH

Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie vornehmen können:

Gruppe	Bereich	Einstellung	Beschreibung
Ansicht		Länge von Schritt- und Transitionsnamen	Legt fest, ob Schritt- und Transitionsnamen in einer oder zwei Zeilen angezeigt werden sollen. Mit der Einstellung "Zwei Zeilen anzeigen" sind auch längere Namen sichtbar.
Laden		Kette vor Laden des DB ausschalten	Vor dem Laden des Instanz-Datenbausteins werden alle aktiven Schritte deaktiviert.
Voreinstellung für Zeitüberwachungen		Aktivierungszeit des Schritts ohne Störungen (U)	Aktivierungszeit für die Zeitüberwachung eines Schritts durch die Anweisung "Größer ununterbrochene Schrittaktivierungszeit" (CMP>U).
		Aktivierungszeit des Schritts (T)	Aktivierungszeit für die Zeitüberwachung eines Schritts durch die Anweisung "Größer Schrittaktivierungszeit" (CMP>T).
		Maximale Schrittaktivierungszeit (T_MAX)	Legt die maximale Zeit fest, die seit der Aktivierung eines Schritts vergangen sein darf. Diese Zeit können Sie mithilfe der Anweisung "Größer maximale Schrittaktivierungszeit" (CMP>T_MAX) überwachen. Sie können diesen Wert auch vom System ermitteln lassen. Siehe auch: Grundlagen zur Schrittzeit-Überwachung (Seite 7784) Diese Einstellung ist nur für CPUs der Baureihe S7-1500 relevant.
		Warnungszeit (T_WARN)	Legt die Zeitdauer fest, nach deren Ablauf seit der Aktivierung eines Schritts eine Warnung ausgegeben werden soll. Diese Zeit können Sie mithilfe der Anweisung "Größer Warnungszeit" (CMP>T_WARN) überwachen. Sie können diesen Wert auch vom System ermitteln lassen. Siehe auch: Grundlagen zur Schrittzeit-Überwachung (Seite 7784) Diese Einstellung ist nur für CPUs der Baureihe S7-1500 relevant.

Gruppe	Bereich	Einstellung	Beschreibung
Voreinstellung für neue Bausteine	Editor	Sprache in Netzwerken	Auswahl der Programmiersprache für das Programmieren der Bedingungen.
	Schnittstelle	Minimale Schnittstellenparameter	Fügt beim Anlegen von neuen GRAPH-Funktionsbausteinen den minimalen Parametersatz ein.
		Standard-Schnittstellenparameter	Fügt beim Anlegen von neuen GRAPH-Funktionsbausteinen den Standard-Parametersatz ein.
		Maximale Schnittstellenparameter	Fügt beim Anlegen von neuen GRAPH-Funktionsbausteinen die maximale Anzahl von Parametern in die Bausteinschnittstelle ein.
	Übersetzen	Speicherplatzminimierten DB erzeugen	Erzeugt beim Übersetzen einen Instanz-Datenbaustein mit einem reduzierten Satz von statischen Parametern. Dies führt jedoch zu Funktionseinschränkungen. Sie können diese Einstellung für einzelne Bausteine nachträglich wieder ändern. Siehe auch: Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)
	Ketteneigenschaften	Schritte überspringen	Überspringt Schritte, bei denen die nachfolgende Transition bereits bei der Aktivierung des Schritts erfüllt ist. Der Schritt wird in diesem Fall nicht aktiv. Sie können diese Einstellung für einzelne Bausteine nachträglich ändern.
		Quittierpflicht bei Supervisionsfehlern	Legt fest, dass Supervisionsfehler quittiert werden müssen, bevor die Ablaufkette weiterbearbeitet wird.
		Permanente Bearbeitung aller Interlocks im Handbetrieb	Erlaubt die Bearbeitung aller Interlocks im Handbetrieb.
		Sperre der Betriebsartenwahl	Verhindert den manuellen Wechsel der Betriebsart über die Task Card "Testen". Sie können diese Einstellung für einzelne Bausteine nachträglich ändern.
		ENO automatisch setzen	Zeigt beim Testen mit dem Programmstatus für CPUs der Baureihe S7-1500 den zusätzlichen Operanden "ENO" an.
	Meldungseigenschaften	Meldungen aktivieren	Aktiviert für neue GRAPH-Funktionsbausteine die Interlock- und Supervisionsmeldungen. Sie können diese Einstellung für einzelne Bausteine nachträglich ändern.
Synchronisation		Vorhergehende Transition erfüllt	Markiert beim Auffinden möglicher Synchronisationspunkte die Schritte, deren vorhergehende Transition erfüllt ist. Die nachfolgende Transition ist dabei nicht erfüllt.
		Interlock erfüllt	Markiert beim Auffinden möglicher Synchronisationspunkte die Schritte, deren Interlock erfüllt ist. Die nachfolgende Transition ist dabei erfüllt.

Siehe auch

Übersicht über die Bausteineigenschaften (Seite 7215)
Einstellungen ändern (Seite 7731)
Programmiersprache in Netzwerken festlegen (Seite 7733)
Schnittstellenparametersatz festlegen (Seite 7734)
Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)
Einstellungen für interne Parameter ändern (Seite 7737)

17.2.2 Einstellungen ändern

Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Ändern Sie die Einstellungen.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Siehe auch

Übersicht über die Einstellungen zu GRAPH (Seite 7729)
Programmiersprache in Netzwerken festlegen (Seite 7733)
Schnittstellenparametersatz festlegen (Seite 7734)
Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)
Einstellungen für interne Parameter ändern (Seite 7737)

17.2.3 GRAPH-Versionen

17.2.3.1 Übersicht über die GRAPH-Versionen

Überblick

Die folgende Tabelle zeigt die zur Verfügung stehenden Versionen für GRAPH-Funktionsbausteine und die darin enthaltene Funktionalität:

Version	Beschreibung
V1.0	Beinhaltet die Basisfunktionalität für GRAPH-Funktionsbausteine.
V2.0	Stellt Ihnen zusätzlich zur Basisfunktionalität den Lernmodus zum Ermitteln der Werte von "T_MAX" und "T_Warn" zur Verfügung.
V3.0 *	Beinhaltet die Funktionalität von V1.0 und V2.0 und erweitert zusätzlich den Funktionsumfang der PLC-Code-Anzeige und verbessert die Bearbeitung von Meldungen.
V4.0 *	Voraussetzung für die Aktivierung der Erstwerterfassung. Es gibt zwei neue statische Parameter für "STEPx" und drei neue Parameter für "Transx". Diese werden benötigt, um den Signalzustand von Operanden bzw. das Ergebnis von Vergleichern aus einem Schritt oder einer Transition zu speichern. Erweitern des Standardverhaltens von GRAPH.
V5.0 *	Voraussetzung für: • Das Übersetzen der GRAPH-Schritt- und GRAPH-Transitionsnamen • Das Zurücksetzen der Erstwerte von Interlocks und Transitionen
V6.0 *	Ermöglicht mehrsprachige Unterkategorien für Interlock- und Supervisionsfehler und GRAPH-Warnungen.

* Nur für S7-1500 ab Firmware V2.0 verfügbar

In Projekten, die mit dem TIA Portal V15.1 angelegt werden, wird der GRAPH-Funktionsbaustein automatisch mit der Version 6.0 erstellt.

Siehe auch

Version des GRAPH-Funktionsbausteins festlegen (Seite 7732)

17.2.3.2 Version des GRAPH-Funktionsbausteins festlegen

Sie können für einen GRAPH-Funktionsbaustein verschiedene Versionen verwenden. Abhängig von der gewählten Version stehen Ihnen unterschiedliche Funktionen zur Verfügung. Sie können die Version entweder über die Projektnavigation oder im Inspektorfenster festlegen.

Version des GRAPH-Funktionsbausteins über die Projektnavigation festlegen

Um die Version für einen GRAPH-Funktionsbaustein festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den GRAPH-Funktionsbaustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
4. Wählen Sie im Bereich "Baustein" in der Klappliste "Version" die Version aus, die Sie verwenden möchten.
5. Bestätigen Sie Ihre Auswahl mit "OK".

Version des GRAPH-Funktionsbausteins im Inspektorfenster festlegen

Um die Version für einen GRAPH-Funktionsbaustein im Inspektorfenster festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein, dessen Version Sie ändern möchten.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Allgemein".
4. Wählen Sie im Bereich "Baustein" in der Klappliste "Version" die Version aus, die Sie verwenden möchten.

Siehe auch

Übersicht über die GRAPH-Versionen (Seite 7732)

17.2.4 Programmiersprache in Netzwerken festlegen

Für GRAPH-Funktionsbausteine können Sie die Programmiersprache des Bausteins nicht wechseln. Sie können aber die Programmiersprache der Netzwerke für Bedingungen festlegen. Dabei können Sie zwischen KOP und FUP wählen. Sie können die Programmiersprache entweder in der Projektnavigation oder im Inspektorfenster festlegen.

Programmiersprache in Netzwerken in der Projektnavigation festlegen

Um die Programmiersprache in Netzwerken in der Projektnavigation festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den GRAPH-Funktionsbaustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".

4. Wählen Sie im Bereich "Baustein" in der Klappliste "Sprache in Netzwerken" die Programmiersprache aus, die Sie verwenden möchten.
5. Bestätigen Sie Ihre Auswahl mit "OK".

Programmiersprache in Netzwerken im Inspektorfenster festlegen

Um die Programmiersprache in Netzwerken im Inspektorfenster festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein, dessen Programmiersprache Sie ändern möchten.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Allgemein".
4. Wählen Sie im Bereich "Baustein" in der Klappliste "Sprache in Netzwerken" die Programmiersprache aus, die Sie verwenden möchten.

Siehe auch

Übersicht über die Einstellungen zu GRAPH (Seite 7729)

Einstellungen ändern (Seite 7731)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Einstellungen für interne Parameter ändern (Seite 7737)

17.2.5 Schnittstellenparametersatz festlegen

Sie können den Schnittstellenparametersatz für GRAPH-Funktionsbausteine folgendermaßen festlegen:

- Global für alle neuen GRAPH-Funktionsbausteine
- Lokal für einen geöffneten GRAPH-Funktionsbaustein

Die lokale Einstellung hat Priorität vor der globalen Einstellung.

Hinweis

Sie können in der Bausteinschnittstelle jederzeit einzelne Ein- oder Ausgangsparameter manuell hinzufügen oder löschen.

Schnittstellenparametersatz global für alle neuen GRAPH-Funktionsbausteine festlegen

Um den Schnittstellenparametersatz global für alle neuen GRAPH-Funktionsbausteine festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Wenn Sie den minimalen Schnittstellenparametersatz verwenden möchten, aktivieren Sie die Option "Minimale Schnittstellenparameter".
4. Wenn Sie den Standard-Schnittstellenparametersatz verwenden möchten, aktivieren Sie die Option "Standard-Schnittstellenparameter".
5. Wenn Sie den maximalen Schnittstellenparametersatz verwenden möchten, aktivieren Sie die Option "Maximale Schnittstellenparameter".

Schnittstellenparametersatz lokal für einen geöffneten GRAPH-Funktionsbaustein festlegen

Um den Schnittstellenparametersatz für einen geöffneten GRAPH-Funktionsbaustein festzulegen, gehen Sie folgendermaßen vor:

1. Wenn Sie den minimalen Schnittstellenparametersatz verwenden möchten, wählen Sie im Menü "Bearbeiten > Schnittstellenparameter" den Befehl "Minimale Schnittstellenparameter".
2. Wenn Sie den Standard-Schnittstellenparametersatz verwenden möchten, wählen Sie im Menü "Bearbeiten > Schnittstellenparameter" den Befehl "Standard-Schnittstellenparameter".
3. Wenn Sie den maximalen Schnittstellenparametersatz verwenden möchten, wählen Sie im Menü "Bearbeiten > Schnittstellenparameter" den Befehl "Maximale Schnittstellenparameter".

Die Bausteinschnittstelle des GRAPH-Funktionsbausteins wird angepasst.

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Übersicht über die Einstellungen zu GRAPH (Seite 7729)

Einstellungen ändern (Seite 7731)

Programmiersprache in Netzwerken festlegen (Seite 7733)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

Einstellungen für interne Parameter ändern (Seite 7737)

17.2.6 Speicherplatzmodell des Datenbausteins festlegen

Sie können das Speicherplatzmodell für GRAPH-Funktionsbausteine folgendermaßen festlegen:

- Global für alle neuen GRAPH-Funktionsbausteine
- Lokal für einen geöffneten GRAPH-Funktionsbaustein

Die lokale Einstellung hat Priorität vor der globalen Einstellung.

Hinweis

GRAPH-spezifische Parameter, die im minimierten Datenbaustein nicht zur Verfügung stehen, können Sie nicht manuell in die Bausteinschnittstelle einfügen.

Speicherplatzmodell global für alle neuen GRAPH-Funktionsbausteine festlegen

Um das Speicherplatzmodell global für alle neuen GRAPH-Funktionsbausteine festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Wenn Sie einen speicherplatzminimierten Datenbaustein verwenden möchten, aktivieren Sie die Option "Speicherplatzminimierten DB erzeugen".
4. Wenn Sie einen Datenbaustein mit dem Standard-Speicherplatzmodell verwenden möchten, deaktivieren Sie die Option "Speicherplatzminimierten DB erzeugen".

Speicherplatzmodell lokal für einen geöffneten GRAPH-Funktionsbaustein festlegen

Um das Speicherplatzmodell für einen geöffneten GRAPH-Funktionsbaustein festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
2. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Attribute".
3. Wenn Sie einen speicherplatzminimierten Datenbaustein verwenden möchten, aktivieren Sie in der Gruppe "Übersetzen" die Option "Speicherplatzminimierten DB erzeugen".
4. Wenn Sie einen Datenbaustein mit dem Standard-Speicherplatzmodell verwenden möchten, deaktivieren Sie in der Gruppe "Übersetzen" die Option "Speicherplatzminimierten DB erzeugen".

Die Schnittstelle des GRAPH-Bausteins wird angepasst.

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Übersicht über die Einstellungen zu GRAPH (Seite 7729)

Einstellungen ändern (Seite 7731)

Programmiersprache in Netzwerken festlegen (Seite 7733)

Schnittstellenparametersatz festlegen (Seite 7734)

Einstellungen für interne Parameter ändern (Seite 7737)

17.2.7 Einstellungen für interne Parameter ändern

Sie können die internen Parameter eines GRAPH-Bausteins als sichtbar und erreichbar aus HMI deklarieren. Für CPUs der Baureihe S7-1500 können Sie die internen Parameter zusätzlich als remanent deklarieren. Wenn Sie die internen Parameter als remanent deklarieren, bleiben deren Werte auch nach einem Betriebszustandswechsel von "STOP" nach "RUN“, z. B. nach einem Stromausfall, erhalten. Dies hat zur Folge, dass die Schrittkette im aktuellen Zustand bleibt und nach dem Wechsel nach „RUN“ an der Stelle weiterläuft, an der sie unterbrochen wurde.

Parameter, die Sie in der Schnittstelle eines GRAPH-Bausteins selbst deklariert haben, sind von diesen Einstellungen nicht betroffen.

Interne Parameter als sichtbar und erreichbar aus HMI deklarieren

Um alle internen Parameter eines GRAPH-Bausteins als sichtbar und erreichbar aus HMI zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Baustein, für den Sie die Einstellung ändern möchten.
2. Aktivieren Sie im Menü "Bearbeiten" das Optionskästchen "Interne Parameter sichtbar/ erreichbar aus HMI". Deaktivieren Sie das Optionskästchen, wenn Sie die Einstellung zurücksetzen möchten.

Interne Parameter als remanent deklarieren (S7-1500)

Um alle internen Parameter eines GRAPH-Bausteins als remanent zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Baustein, für den Sie die Einstellung ändern möchten.
2. Aktivieren Sie im Menü "Bearbeiten" das Optionskästchen "Interne Parameter remanent". Deaktivieren Sie das Optionskästchen, wenn Sie die Einstellung zurücksetzen möchten.

Hinweis

Besonderheiten für S7-1500-CPUs mit einem Firmware-Stand kleiner V1.5

Wenn Sie die internen Parameter eines GRAPH-Bausteins als remanent deklarieren, dann hat dies zur Folge, dass die Ablaufsteuerung im aktuellen Zustand bleibt und nach dem Wechsel nach "RUN" an der Stelle weiterläuft, an der sie unterbrochen wurde. Beachten Sie, dass bei Aktivierung des neuen Parameters "Interne Parameter remanent" in einem GRAPH-Baustein, Supervisionen nicht genutzt werden sollten.

Daten im Standardbereich remanent setzen (S7-300/400)

Um die Daten innerhalb des Standardbereichs remanent zu setzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Instanzsdatenbaustein des GRAPH-Bausteins.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf "Attribute".
5. Aktivieren Sie das Optionskästchen "Daten im Standardbereich remanent setzen".
6. Bestätigen Sie Ihre Eingaben mit "OK".

Siehe auch

Übersicht über die Einstellungen zu GRAPH (Seite 7729)

Einstellungen ändern (Seite 7731)

Programmiersprache in Netzwerken festlegen (Seite 7733)

Schnittstellenparametersatz festlegen (Seite 7734)

Speicherplatzmodell des Datenbausteins festlegen (Seite 7736)

17.3 Das Programmierfenster von GRAPH

17.3.1 Übersicht über das Programmierfenster

Funktion

Das Programmierfenster ist der Bereich, in dem Sie das GRAPH-Programm erstellen. Es enthält neben dem eigentlichen Arbeitsbereich eine Navigation, um Ihnen einen einfachen Zugriff auf alle Bereiche des GRAPH-Funktionsbausteins zu ermöglichen.

Im Programmierfenster können Sie die folgenden Aufgaben durchführen:

- Programmieren von vorausgeschalteten und nachgeschalteten permanenten Anweisungen
- Programmieren der Ablaufketten
- Festlegen der Interlock- und Supervisionsmeldungen

Abhängig davon, was Sie programmieren möchten, können Sie zwischen den folgenden Ansichten umschalten:

- Vorausgeschaltete permanente Anweisungen (Seite 7741)
- Kettenansicht (Seite 7742)
- Einzelschrittansicht (Seite 7744)
- Nachgeschaltete permanente Anweisungen (Seite 7746)
- Meldungsansicht (Seite 7747)

Der Arbeitsbereich und die angebotenen Anweisungen und Favoriten variieren je nach Ansicht.

Aufbau des Programmierfensters

Das folgende Bild zeigt das Programmierfenster von GRAPH in der Kettenansicht:

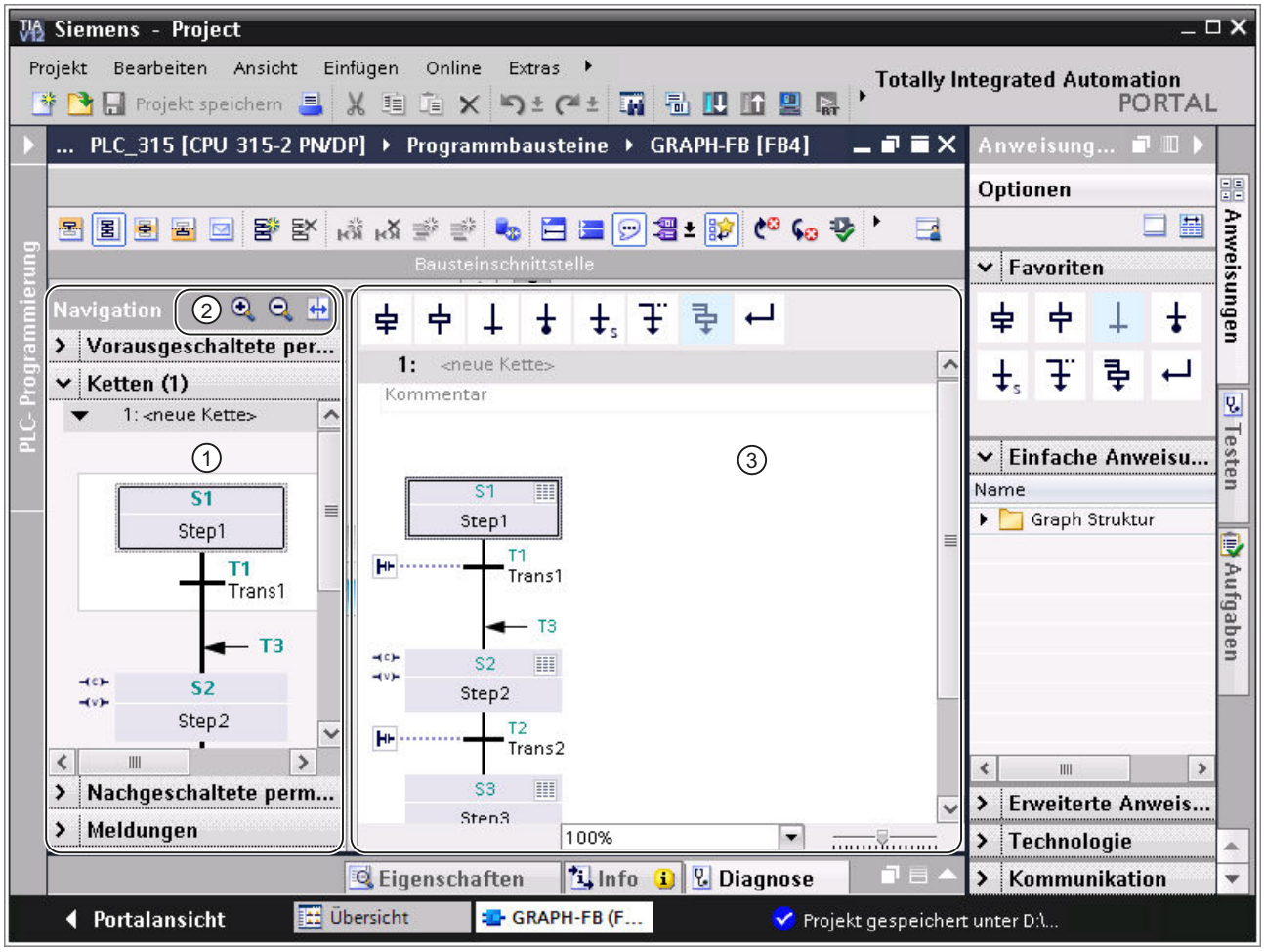

① Navigation

② Funktionsleiste der Navigation

③ Arbeitsbereich

Navigation

Die Navigation enthält die folgenden Paletten, über die Sie folgende Ansichten öffnen können:

- Vorausgeschaltete permanente Anweisungen
- Kettenansicht
- Nachgeschaltete permanente Anweisungen
- Meldungsansicht

Zusätzlich zeigt die Navigation grafische Übersichten über die permanenten Anweisungen und Ablaufketten und bietet Ihnen über das Kontextmenü grundlegende Bearbeitungsmöglichkeiten.

Mit einem Doppelklick auf einen Schritt in der Navigation können Sie zwischen der Ketten- und der Einzelschrittansicht wechseln.

Funktionsleiste der Navigation

Über die Funktionsleiste der Navigation können Sie folgende Aktionen durchführen:

- Elemente innerhalb der Navigation vergrößern oder verkleinern
  Mithilfe der Vergrößern- und Verkleinern-Schaltflächen können Sie die grafischen Elemente der permanenten Anweisungen und der Ketten innerhalb der Navigation vergrößern und verkleinern. Sie können den Zoomfaktor für jede Palette getrennt einstellen. Beim Schließen des GRAPH-Funktionsbausteins wird die Änderung des Zoomfaktors verworfen. Wenn Sie den Zoomfaktor erhalten möchten, können Sie dies über die Schaltfläche "Anordnung merken" erreichen.
- Navigation synchronisieren
  Wenn die Schaltfläche aktiv ist, werden Navigation und Arbeitsbereich miteinander synchronisiert, sodass immer die gleichen Elemente angezeigt werden. Wenn Sie die Schaltfläche deaktivieren, können Sie in Navigation und Arbeitsbereich verschiedene Objekte anzeigen.

Arbeitsbereich

Im Arbeitsbereich programmieren Sie die einzelnen Komponenten Ihrer Ablaufkette. Dabei können Sie das GRAPH-Programm in unterschiedlichen Ansichten anzeigen. Sie können diese Ansichten über eine Zoom-Funktion skalieren.

17.3.2 Ansicht "Vorausgeschaltete permanente Anweisungen"

Funktion der Ansicht "Vorausgeschaltete permanente Anweisungen"

In der Ansicht "Vorausgeschaltete permanente Anweisungen" können Sie Anweisungen programmieren, die vor der Bearbeitung der Ablaufkette ausgeführt werden.

Aufbau der Ansicht "Vorausgeschaltete permanente Anweisungen"

Das folgende Bild zeigt ein Beispiel für die Ansicht "Vorausgeschaltete permanente Anweisungen" mit FUP als Programmiersprache:

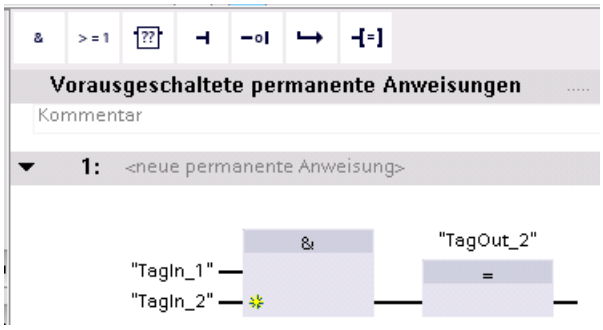

Siehe auch

Permanente Anweisungen (Seite 7705)

Übersicht über das Programmierfenster (Seite 7739)

Kettenansicht (Seite 7742)

Einzelschrittansicht (Seite 7744)

Ansicht "Nachgeschaltete permanente Anweisungen" (Seite 7746)

17.3.3 Kettenansicht

Funktion der Kettenansicht

Die Kettenansicht zeigt Ihnen übersichtlich die Struktur Ihrer Ablaufkette und ermöglicht Ihnen sie durch die folgenden Elemente zu erweitern:

- Schritte
- Transitionen
- Sprünge
- Verzweigungen
- Kettenenden

Zusätzlich können Sie Schritte und Transitionen per Mausklick erweitern, um sich die Aktionen eines Schritts und die Bedingungen einer Transition anzeigen zu lassen oder sie zu bearbeiten.

Aufbau der Kettenansicht

Das folgende Bild zeigt ein Beispiel für die Kettenansicht:

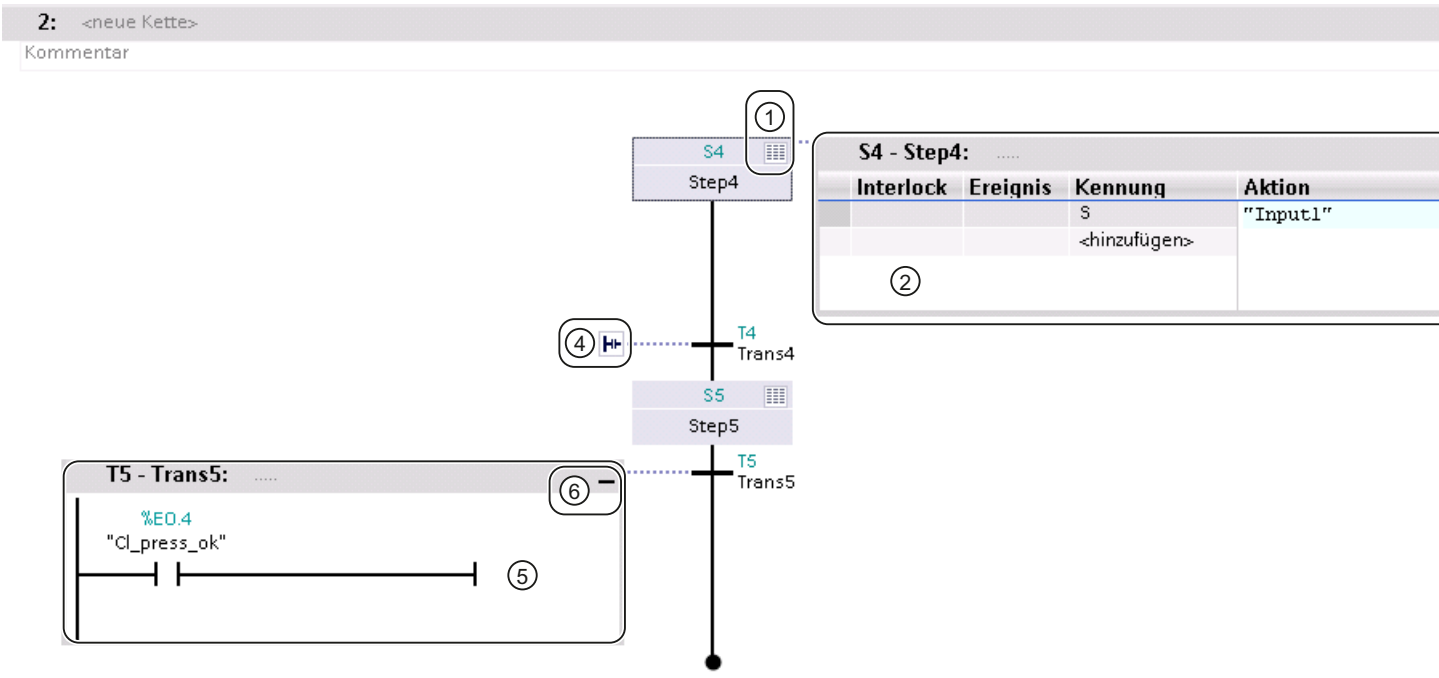

① Schaltfläche zum Erweitern oder Reduzieren eines Schritts

② Schritt in der erweiterten Ansicht

③ Schaltfläche zum Reduzieren eines Schritts

④ Schaltfläche zum Erweitern einer Transition

⑤ Transition in der erweiterten Ansicht

⑥ Schaltfläche zum Reduzieren einer Transition

Siehe auch

Übersicht über das Programmierfenster (Seite 7739)

Ansicht "Vorausgeschaltete permanente Anweisungen" (Seite 7741)

Einzelschrittansicht (Seite 7744)

Ansicht "Nachgeschaltete permanente Anweisungen" (Seite 7746)

17.3.4 Einzelschrittansicht

Funktion der Einzelschrittansicht

Die Einzelschrittansicht bietet Ihnen die Möglichkeit, die folgenden Elemente eines Schritts zu programmieren:

- Interlock
- Supervision
- Aktionen
- Transition

Außerdem können Sie den Titel des Schritts festlegen und einen Kommentar zum Schritt eingeben.

Hinweis

Die Aktionen des Schritts und die zugehörige Transition können Sie zusätzlich auch in der Kettenansicht programmieren.

Aufbau der Einzelschrittansicht

Das folgende Bild zeigt einen Schritt in der Einzelschrittansicht:

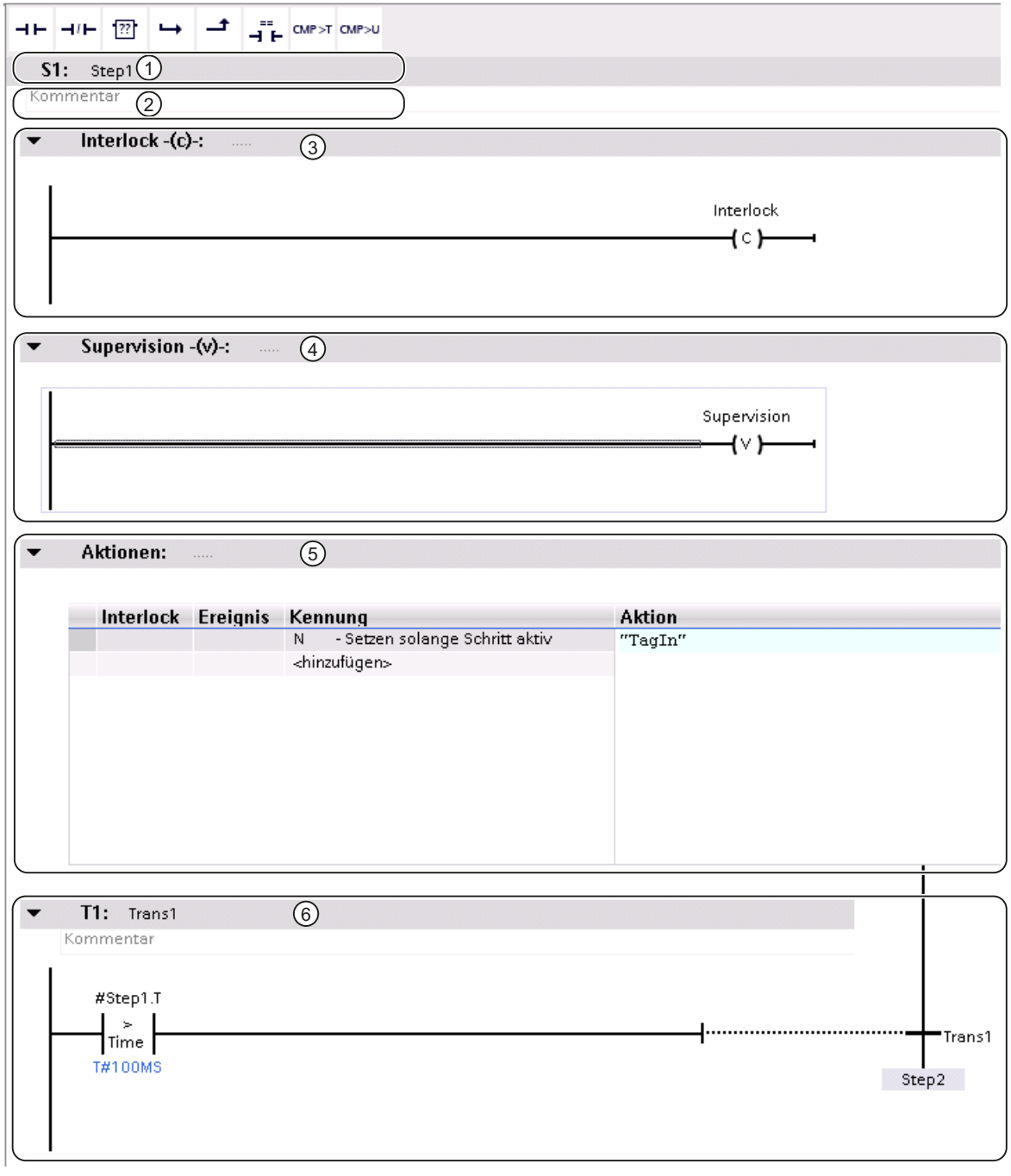

① Titel des Schritts

② Kommentar zum Schritt

③ Interlock

④ Supervision

⑤ Aktionen

⑥ Transition

Siehe auch

Elemente eines Schritts (Seite 7693)

Übersicht über das Programmierfenster (Seite 7739)

Ansicht "Vorausgeschaltete permanente Anweisungen" (Seite 7741)

Kettenansicht (Seite 7742)

Ansicht "Nachgeschaltete permanente Anweisungen" (Seite 7746)

17.3.5 Ansicht "Nachgeschaltete permanente Anweisungen"

Funktion der Ansicht "Nachgeschaltete permanente Anweisungen"

In der Ansicht "Nachgeschaltete permanente Anweisungen" können Sie Anweisungen programmieren, die nach der Bearbeitung der Ablaufkette ausgeführt werden.

Aufbau der Ansicht "Nachgeschaltete permanente Anweisungen"

Das folgende Bild zeigt ein Beispiel für die Ansicht "Nachgeschaltete permanente Anweisungen" mit KOP als Programmiersprache:

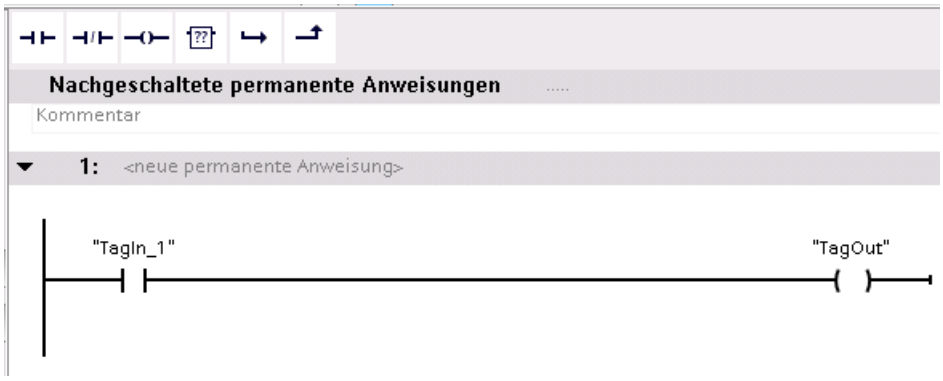

Siehe auch

Permanente Anweisungen (Seite 7705)

Übersicht über das Programmierfenster (Seite 7739)

Ansicht "Vorausgeschaltete permanente Anweisungen" (Seite 7741)

Kettenansicht (Seite 7742)

Einzelschrittansicht (Seite 7744)

17.3.6 Meldungsansicht

Funktion der Meldungsansicht

In der Meldungsansicht können Sie die folgenden Aufgaben durchführen:

- Meldungsanzeige aktivieren
- Freigabekriterien der Kategorie festlegen
- Anzeigeklassen auswählen
- Kategorie und Unterkategorien für Interlocks auswählen
- Kategorie und Unterkategorien für Supervisionen auswählen
- Kategorie und Unterkategorien für GRAPH-Warnungen auswählen

Aufbau der Meldungsansicht

Das folgende Bild zeigt die Meldungsansicht eines GRAPH-Funktionsbausteins:

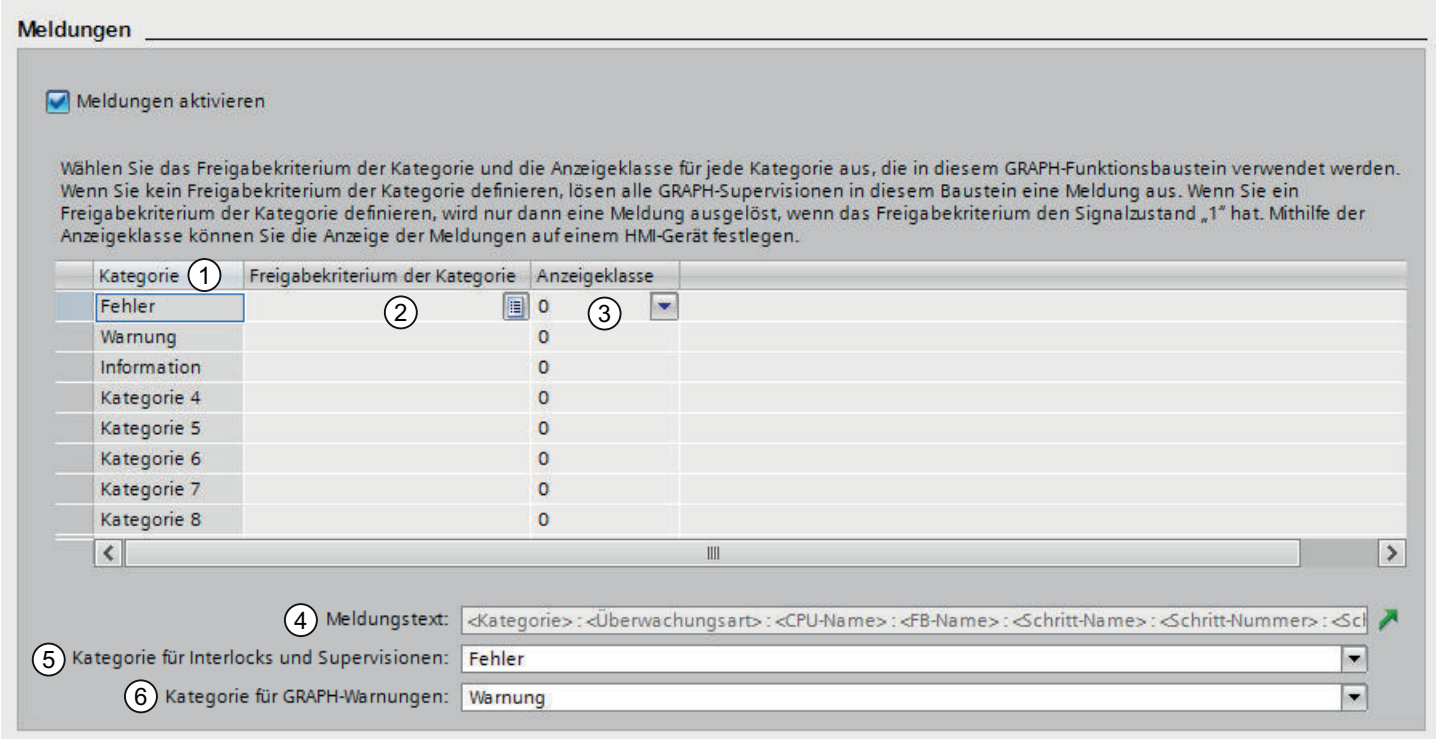

Nummerierung	Erklärung	Weiterführende Informationen
1	Mithilfe der Kategorien können Sie die auftretenden Störungen in unterschiedliche Dringlichkeitsstufen einteilen, wie z. B. Fehler, Warnung, etc.	Weitere Informationen zu den Kategorien finden Sie hier: Kategorien (Seite 8224)
2	Mithilfe des Freigabekriteriums der Kategorie können Sie für den ProDiag-Funktionsbaustein und für jede einzelne darin enthaltene Kategorie ein Freigabekriterium (globale boolesche Variable) definieren und damit die Überwachungen dieser Kategorie aktivieren bzw. deaktivieren.	Weitere Informationen zum Freigabekriterium finden Sie hier: Freigabekriterium der Kategorie verwenden (Seite 8218)
3	Mithilfe der Anzeigeklasse können Sie definieren, welche Überwachungen an einem HMI-Gerät angezeigt werden sollen.	Weitere Informationen zu den Anzeigeklassen finden Sie hier: Anzeigeklassen einstellen (Seite 8220)
4	Meldetextstruktur der GRAPH-Überwachungen Mithilfe des grünen Pfeils können Sie direkt in die die globalen ProDiag-Überwachungseinstellungen springen, wo Sie die Meldetextstruktur festlegen.	Weitere Informationen zum Erstellen der Meldetextstruktur finden Sie hier: Aufbau einer GRAPH-Überwachungsmeldung (Seite 8236)
5	Für Interlocks können Sie eine Kategorie und zwei Unterkategorien festlegen. Sie können aus maximal 8 Kategorien auswählen.	Weitere Informationen finden Sie hier: Interlocks programmieren (Seite 7782)
6	Für Supervisionen können Sie eine Kategorie und zwei Unterkategorien festlegen. Sie können aus maximal 8 Kategorien auswählen.	Weitere Informationen finden Sie hier: Supervisionen programmieren (Seite 7783)
7	Für GRAPH-Warnungen, die von der Anweisung "T_WARN" ausgegeben werden, können Sie eine Kategorie und zwei Unterkategorien festlegen. Sie können aus maximal 8 Kategorien auswählen.	Weitere Informationen finden Sie hier: Schrittzeit-Überwachung verwenden (Seite 7784)

Siehe auch

Meldungen festlegen (Seite 7806)

Übersicht über das Programmierfenster (Seite 7739)

Ansicht "Vorausgeschaltete permanente Anweisungen" (Seite 7741)

Kettenansicht (Seite 7742)

Einzelschrittansicht (Seite 7744)

Ansicht "Nachgeschaltete permanente Anweisungen" (Seite 7746)

17.4 Favoriten in GRAPH verwenden

17.4.1 GRAPH-Elemente zu den Favoriten hinzufügen

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in einer der folgenden Ansichten geöffnet:
  - Vorausgeschaltete permanente Anweisungen
  - Kettenansicht
  - Einzelschrittansicht
  - Nachgeschaltete permanente Anweisungen
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im GRAPH-Programmierfenster angezeigt.

Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im GRAPH-Programmierfenster.

Hinweis

Um die Favoriten zusätzlich auch im Programmierfenster anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

GRAPH-Elemente mithilfe von Favoriten einfügen (Seite 7750)

GRAPH-Elemente aus den Favoriten entfernen (Seite 7751)

17.4.2 GRAPH-Elemente mithilfe von Favoriten einfügen

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in einer der folgenden Ansichten geöffnet:
  - Vorausgeschaltete permanente Anweisungen
  - Kettenansicht
  - Einzelschrittansicht
  - Nachgeschaltete permanente Anweisungen
- Favoriten sind vorhanden.

Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

Hinweis

Um die Favoriten zusätzlich auch im Programmierfenster anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

GRAPH-Elemente zu den Favoriten hinzufügen (Seite 7749)

GRAPH-Elemente aus den Favoriten entfernen (Seite 7751)

17.4.3 GRAPH-Elemente aus den Favoriten entfernen

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in einer der folgenden Ansichten geöffnet:
  - Vorausgeschaltete permanente Anweisungen
  - Kettenansicht
  - Einzelschrittansicht
  - Nachgeschaltete permanente Anweisungen
- Favoriten sind vorhanden.

Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Favoritenbereich mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

Hinweis

Um die Favoriten zusätzlich auch im Programmierfenster anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

Siehe auch

GRAPH-Elemente zu den Favoriten hinzufügen (Seite 7749)

GRAPH-Elemente mithilfe von Favoriten einfügen (Seite 7750)

17.5 Struktur einer Ablaufkette programmieren

17.5.1 Ablaufketten einfügen

Wenn Sie einen GRAPH-Funktionsbaustein öffnen, ist in der Navigation die Palette "Ketten" aufgeklappt. In einem neuen GRAPH-Funktionsbaustein ist bereits eine Ablaufkette mit einem Schritt und einer Transition vorhanden. Sie können jederzeit weitere Ablaufketten einfügen.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist geöffnet.

Vorgehen

Um eine neue Ablaufkette einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Navigation mit der rechten Maustaste auf die Titelleiste der Palette "Ketten" oder auf die Titelleiste einer bereits vorhandenen Ablaufkette.
2. Wählen Sie im Kontextmenü den Befehl "Kette einfügen".
   Die neue Ablaufkette wird nach der selektierten Ablaufkette eingefügt. Ist keine Ablaufkette selektiert, wird sie vor der ersten Ablaufkette eingefügt.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten umbenennen (Seite 7752)

Ablaufketten kommentieren (Seite 7753)

Elemente in eine Ablaufkette einfügen (Seite 7754)

Ablaufketten beenden (Seite 7755)

Ablaufketten löschen (Seite 7756)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.2 Ablaufketten umbenennen

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Vorgehen

Um eine Ablaufkette umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Kettenansicht in die Titelzeile der Ablaufkette.
   Standardmäßig ist der Platzhalter "<neue Kette>" vorhanden.
2. Geben Sie den neuen Namen der Ablaufkette ein.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten einfügen (Seite 7752)

Ablaufketten kommentieren (Seite 7753)

Elemente in eine Ablaufkette einfügen (Seite 7754)

Ablaufketten beenden (Seite 7755)

Ablaufketten löschen (Seite 7756)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.3 Ablaufketten kommentieren

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Vorgehen

Um einen Kommentar zu einer Ablaufkette einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Kettenansicht in den Kommentarbereich der Ablaufkette.
   Standardmäßig ist der Platzhalter "Kommentar" vorhanden.
2. Geben Sie einen Kommentar zur Ablaufkette ein.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten einfügen (Seite 7752)

Ablaufketten umbenennen (Seite 7752)

Elemente in eine Ablaufkette einfügen (Seite 7754)

Ablaufketten beenden (Seite 7755)

Ablaufketten löschen (Seite 7756)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.4 Elemente in eine Ablaufkette einfügen

Voraussetzung

Eine Ablaufkette ist in der Kettenansicht geöffnet.

Vorgehen

Um in eine Ablaufkette Elemente einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Stelle in der Ablaufkette, an der Sie ein neues Element einfügen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element, das Sie einfügen möchten. Sie erhalten nur die Elemente zur Auswahl, die an der selektierten Stelle zulässig sind.

Oder:

1. Öffnen Sie in der Task Card "Anweisungen" die Palette "Einfache Anweisungen".
2. Öffnen Sie den Ordner "Graph Struktur".
3. Ziehen Sie das Element, das Sie einfügen möchten, per Drag & Drop an die gewünschte Stelle in der Ablaufkette. Beachten Sie dabei, dass nicht alle Elemente an allen Stellen eingefügt werden können. Erlaubte Stellen werden durch ein kleines grünes Quadrat angezeigt.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten einfügen (Seite 7752)

Ablaufketten umbenennen (Seite 7752)

Ablaufketten kommentieren (Seite 7753)

Ablaufketten beenden (Seite 7755)

Ablaufketten löschen (Seite 7756)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.5 Ablaufketten beenden

Jede Ablaufkette muss entweder mit einem Sprung zu einem anderen Schritt in der eigenen oder einer anderen Ablaufkette oder mit dem Element "Kettenende" beendet werden. Mit einem Sprung zu einem Schritt ermöglichen Sie eine zyklische Bearbeitung der Ablaufkette.

Voraussetzung

Eine Ablaufkette ist in der Kettenansicht geöffnet.

Vorgehen

Um eine Kette zu beenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Doppelpfeil am Ende der Ablaufkette.
2. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element "Kettenende".

Oder:

1. Klicken Sie mit der rechten Maustaste auf den Doppelpfeil am Ende der Ablaufkette.
2. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element "Sprung". Eine Übersicht mit allen vorhandenen Schritten des GRAPH-Funktionsbausteins wird geöffnet.
3. Wählen Sie einen Schritt als Sprungziel aus.

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten einfügen (Seite 7752)

Ablaufketten umbenennen (Seite 7752)

Ablaufketten kommentieren (Seite 7753)

Elemente in eine Ablaufkette einfügen (Seite 7754)

Ablaufketten löschen (Seite 7756)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.6 Ablaufketten löschen

Sie können Ablaufketten jederzeit wieder löschen. Allerdings muss mindestens eine Ablaufkette im GRAPH-Funktionsbaustein verbleiben. Beachten Sie beim Löschen von Ablaufketten, dass Sie damit möglicherweise auch Sprungziele löschen.

Vorgehen

Um eine Ablaufkette zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Navigation mit der rechten Maustaste auf die Titelleiste der Ablaufkette.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Grundlagen zu Ablaufketten (Seite 7687)

Elemente einer Ablaufkette (Seite 7688)

Ablaufketten einfügen (Seite 7752)

Ablaufketten umbenennen (Seite 7752)

Ablaufketten kommentieren (Seite 7753)

Elemente in eine Ablaufkette einfügen (Seite 7754)

Ablaufketten beenden (Seite 7755)

Schritte und Transition verwenden (Seite 7756)

Verzweigungen verwenden (Seite 7767)

Sprünge verwenden (Seite 7770)

17.5.7 Schritte und Transition verwenden

17.5.7.1 Schritte und Transitionen einfügen

Wenn Sie einen neuen GRAPH-Funktionsbaustein in Ihr Projekt einfügen, enthält er bereits eine Ablaufkette mit einem Schritt und einer Transition. Sie können weitere Schritte und Transitionen in die Ablaufkette einfügen.

Beachten Sie beim Einfügen von Schritten und Transitionen folgende Regeln:

- Schritte und Transitionen müssen sich immer abwechseln. Das bedeutet, dass Sie keine zwei Schritte oder zwei Transitionen nacheinander einfügen können.
- Daraus folgt, dass Sie Schritte und Transitionen nur am Ende eines Zweigs oder an Stellen, an denen ein Schritt oder eine Transition fehlt, einfügen können.
- Ein Simultanzweig muss immer mit einem Schritt beginnen.

- Ein Alternativzweig muss immer mit einer Transition beginnen.
- Sie können einen Schritt und eine Transition gemeinsam als Paar einfügen. Da hier sichergestellt ist, dass dem Schritt wieder eine Transition folgt, ist dies an jeder beliebigen Stelle in der Ablaufkette möglich.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Schritt einfügen

Um einen Schritt einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Ablaufkette mit der rechten Maustaste auf eine Stelle, an der das Einfügen eines Schritts zulässig ist.
2. Wählen Sie aus dem Kontextmenü den Befehl "Element einfügen > Schritt".

Oder:

1. Ziehen Sie das Element "Schritt" per Drag & Drop aus den Favoriten oder der Task Card "Anweisungen" an die entsprechende Stelle der Ablaufkette.

Transition einfügen

Um eine Transition einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Ablaufkette mit der rechten Maustaste auf eine Stelle, an der das Einfügen einer Transition zulässig ist.
2. Wählen Sie aus dem Kontextmenü den Befehl "Element einfügen > Transition".

Oder:

1. Ziehen Sie das Element "Transition" per Drag & Drop aus den Favoriten oder der Task Card "Anweisungen" an die entsprechende Stelle der Ablaufkette.

Schritt und Transition gemeinsam als Paar einfügen

Um einen Schritt und eine Transition gemeinsam als Paar einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Ablaufkette mit der rechten Maustaste auf die Stelle, an der Sie den Schritt und die Transition einfügen möchten.
2. Wählen Sie aus dem Kontextmenü den Befehl "Element einfügen > Schritt und Transition".

Oder:

1. Ziehen Sie das Element "Schritt und Transition" per Drag & Drop aus den Favoriten oder der Task Card "Anweisungen" an die entsprechende Stelle der Ablaufkette.

Siehe auch

Schritte und Transitionen (Seite 7692)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte überspringen (Seite 7764)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.2 Initialschritt festlegen und abwählen

Wenn Sie einen neuen GRAPH-Funktionsbaustein in Ihr Projekt einfügen, enthält er bereits eine Ablaufkette mit einem Schritt und einer Transition. Dieser Schritt ist standardmäßig als Initialschritt festgelegt. Das bedeutet, dass die Bearbeitung der Ablaufkette mit diesem Schritt beginnt. Sie können jederzeit einen anderen Schritt als Initialschritt definieren. In Simultanzweigen sind auch mehrere Initialschritte zulässig. Die maximale Anzahl der Initialschritte beträgt jedoch 250. Sie können einen Initialschritt nachträglich wieder durch Abwählen zu einem normalen Schritt machen.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Initialschritt festlegen

Um einen Schritt als Initialschritt festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Schritt, den Sie als Initialschritt festlegen möchten.
2. Aktivieren Sie im Kontextmenü das Optionskästchen "Initialschritt".

Initialschritt abwählen

Um einen Initialschritt abzuwählen und wieder als normalen Schritt zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Schritt, den Sie als Initialschritt festlegen möchten.
2. Deaktivieren Sie im Kontextmenü das Optionskästchen "Initialschritt".

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte überspringen (Seite 7764)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.3 Schritte und Transitionen neu nummerieren

Wenn Sie Schritte und Transitionen in Ihre Ablaufkette einfügen, erhalten Sie als Nummer den nächst freien Wert. Sie können diese Nummer nachträglich ändern. Dabei haben Sie die folgenden beiden Möglichkeiten:

- Sie vergeben für die Schritte und Transitionen, die Sie umnummerieren möchten, manuell neue Nummern. Beachten Sie dabei, dass Sie keine Nummern doppelt vergeben dürfen.
- Sie führen die Umnummerierung programmgestützt durch. Dadurch können Sie alle Schritte und Transitionen eines Zweigs, einer Ablaufkette oder eines Bausteins automatisch umnummerieren lassen. Beachten Sie dabei folgende Unterschiede:
  - Umnummerieren innerhalb eines Zweigs:
    Innerhalb des Zweigs werden ab dem gewählten Schritt oder der gewählten Transition alle folgenden Schritte oder Transitionen umnummeriert. Die Umnummerierung endet mit dem Beginn eines neuen Zweigs.
  - Umnummerieren innerhalb einer Ablaufkette:
    Innerhalb der Ablaufkette werden alle Schritte oder Transitionen umnummeriert. Der erste Schritt oder die erste Transition der Ablaufkette erhalten die von Ihnen eingegebene Startnummer. Beachten Sie, dass bei der Umnummerierung gegebenenfalls auch Schritte in anderen Ablaufketten umnummeriert werden, falls sich ansonsten doppelte Nummern ergeben würden.
  - Umnummerieren innerhalb eines Bausteins:
    Innerhalb des GRAPH-Bausteins werden alle Schritte oder Transitionen umnummeriert. Der erste Schritt oder die erste Transition innerhalb der ersten Ablaufkette des Bausteins erhalten die von Ihnen eingegebene Startnummer.

Sie können auch nur die Schritte oder nur die Transitionen umnummerieren lassen.

Hinweis

Beachten Sie jedoch, dass Ihnen diese Option nicht für GRAPH-Funktionsbausteine zur Verfügung steht, für die Sie das Attribut "Speicherplatzminimierten DB erzeugen" aktiviert haben.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Schritte und Transitionen manuell umnummerieren

Um Schritte oder Transitionen manuell umzunummerieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Schritt, dessen Nummer Sie ändern möchten.
2. Klicken Sie auf die Nummer des Schritts.
3. Geben Sie die neue Nummer ein.

Schritte und Transitionen programmgestützt innerhalb eines Zweigs umnummerieren

Um Schritte oder Transitionen programmgestützt innerhalb eines Zweigs umzunummerieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Schritt oder die Transition, ab der Sie den Zweig umnummerieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umnummerieren".
   Der Dialog "Umnummerieren" wird geöffnet.
3. Wenn Sie die Schritte umnummerieren möchten, aktivieren Sie das Optionskästchen "Schritte" und geben Sie die Nummer ein, die der ausgewählte Schritt erhalten soll.
4. Wenn Sie die Transitionen umnummerieren möchten, aktivieren Sie das Optionskästchen "Transitionen" und geben Sie die Nummer ein, die die Transition nach dem ausgewählten Schritt erhalten soll.
5. Wählen Sie im Bereich "Umnummerieren in" das Optionsfeld "Zweig".
6. Bestätigen Sie mit "OK".

Schritte und Transitionen programmgestützt innerhalb einer Ablaufkette umnummerieren

Um Schritte oder Transitionen programmgestützt innerhalb einer Ablaufkette umzunummerieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen Schritt oder eine Transition innerhalb der Ablaufkette, für die Sie die Umnummerierung vornehmen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umnummerieren".
   Der Dialog "Umnummerieren" wird geöffnet.
3. Wenn Sie die Schritte umnummerieren möchten, aktivieren Sie das Optionskästchen "Schritte" und geben Sie die Nummer ein, die der erste Schritt der Ablaufkette erhalten soll.
4. Wenn Sie die Transitionen umnummerieren möchten, aktivieren Sie das Optionskästchen "Transitionen" und geben Sie die Nummer ein, die die erste Transition der Ablaufkette erhalten soll.
5. Wählen Sie im Bereich "Umnummerieren in" das Optionsfeld "Kette".
6. Bestätigen Sie mit "OK".

Schritte und Transitionen programmgestützt innerhalb eines GRAPH-Funktionsbausteins umnummerieren

Um Schritte oder Transitionen programmgestützt innerhalb eines GRAPH-Funktionsbausteins umzunummerieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen Schritt oder eine Transition innerhalb des GRAPH-Funktionsbausteins, für den Sie die Umnummerierung vornehmen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umnummerieren".
   Der Dialog "Umnummerieren" wird geöffnet.
3. Wenn Sie die Schritte umnummerieren möchten, aktivieren Sie das Optionskästchen "Schritte" und geben Sie die Nummer ein, die der erste Schritt des GRAPH-Funktionsbausteins erhalten soll.
4. Wenn Sie die Transitionen umnummerieren möchten, aktivieren Sie das Optionskästchen "Transitionen" und geben Sie die Nummer ein, die die erste Transition des GRAPH-Funktionsbausteins erhalten soll.
5. Wählen Sie im Bereich "Umnummerieren in" das Optionsfeld "Baustein".
6. Bestätigen Sie mit "OK".

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte überspringen (Seite 7764)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.4 Schritte und Transitionen umbenennen

Wenn Sie Schritte und Transitionen in Ihre Ablaufkette einfügen, erhalten Sie als Namen zunächst "Step<Schrittnummer>" und "Trans<Schrittnummer>". Sie können diese Namen ändern, um die Übersichtlichkeit Ihres Programms zu erhöhen. Beachten Sie dabei, dass Sie keine Namen doppelt vergeben dürfen.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Schritt umbenennen

Um einen Schritt umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Schritt, dessen Namen Sie ändern möchten.
2. Klicken Sie auf den Schrittnamen.
3. Geben Sie einen neuen Namen ein.

Transition umbenennen

Um eine Transition umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie doppelt auf den Transitionsnamen, den Sie ändern möchten.
2. Geben Sie einen neuen Namen ein.

Siehe auch

Schritte und Transitionen (Seite 7692)
Schritte und Transitionen einfügen (Seite 7756)
Initialschritt festlegen und abwählen (Seite 7758)
Schritte und Transitionen neu nummerieren (Seite 7759)
Schritte kommentieren (Seite 7762)
Schritte und Transitionen kopieren und einfügen (Seite 7763)
Schritte überspringen (Seite 7764)
Schritte und Transitionen löschen (Seite 7765)

17.5.7.5 Schritte kommentieren

Sie haben die Möglichkeit, Schritte zu kommentieren. Damit können Sie z. B. den Zweck des Schritts angeben oder auf Besonderheiten hinweisen.

Voraussetzung

Ein Schritt ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um einen Schritt zu kommentieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Kommentarbereich auf "Kommentar".
   Der Text "Kommentar" wird gelöscht und Sie erhalten eine Einfügemarke.
2. Geben Sie den Kommentar ein.

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte überspringen (Seite 7764)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.6 Schritte und Transitionen kopieren und einfügen

Sie können Schritte und Transitionen kopieren und in die gleiche oder eine andere Ablaufkette einfügen. Dadurch können Sie den Programmieraufwand verringern, wenn ein neuer Schritt ähnliche Aufgaben durchführen soll wie ein bereits vorhandener Schritt. Beachten Sie jedoch folgende Besonderheiten:

- Auch beim Kopieren von Schritten und Transitionen muss die Abfolge von Schritt und Transition erhalten bleiben. Daher werden Platzhalter eingefügt ("Missing Step", "Missing Transition"), falls nach dem Einfügevorgang zwei Schritte oder zwei Transitionen aufeinander folgen.
- Es erfolgt ein Umbenennen und Umnummerieren des eingefügten Elements, um doppelte Namen und Nummern zu vermeiden.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Schritte und Transitionen kopieren

Um einen Schritt oder eine Transition zu kopieren, gehen Sie folgendermaßen vor:

1. Um einen Schritt zu kopieren, klicken Sie mit der rechten Maustaste auf den Schritt, den Sie kopieren möchten. Um eine Transition zu kopieren, klicken Sie mit der rechten Maustaste auf die Transition, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Schritte und Transitionen einfügen

Um einen Schritt oder eine Transition einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Stelle der Ablaufkette, an der Sie den Schritt oder die Transition einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Einfügen".
3. Ersetzen Sie ggf. die Platzhalter durch gültige Elemente.

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte überspringen (Seite 7764)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.7 Schritte überspringen

Sie haben die Möglichkeit, das Ausführen von Schritten zu verhindern, bei denen die vorangehende und nachfolgende Transition gleichzeitig gültig werden. Beachten Sie bei CPUs der Baureihe S7-300/400 jedoch, dass die Option nicht zusammen mit der Option "Speicherplatzminimierten DB erzeugen" verwendet werden kann.

Vorgehen

Um Schritte zu überspringen, für die die vorangehende und nachfolgende Transition gleichzeitig gültig werden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Aktivieren Sie unter "Ketteneigenschaften" die Option "Schritte überspringen".

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte und Transitionen löschen (Seite 7765)

17.5.7.8 Schritte und Transitionen löschen

Sie können Schritte und Transitionen wieder aus einer Ablaufkette entfernen. Beachten Sie jedoch folgende Besonderheiten:

- Auch beim Löschen von Schritten und Transitionen muss die Abfolge von Schritt und Transition erhalten bleiben. Daher werden Platzhalter eingefügt ("Missing Step", "Missing Transition"), falls nach dem Löschen zwei Schritte oder zwei Transitionen aufeinander folgen.
- Sie können nicht alle Schritte einer Ablaufkette löschen, da eine Ablaufkette immer mindestens einen Schritt enthalten muss.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Vorgehen

Um einen Schritt oder eine Transition zu löschen, gehen Sie folgendermaßen vor:

1. Um einen Schritt zu löschen, klicken Sie mit der rechten Maustaste auf den Schritt, den Sie löschen möchten. Um eine Transition zu löschen, klicken Sie mit der rechten Maustaste auf die Transition, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Ersetzen Sie ggf. die Platzhalter durch gültige Elemente.

Siehe auch

Schritte und Transitionen (Seite 7692)

Schritte und Transitionen einfügen (Seite 7756)

Initialschritt festlegen und abwählen (Seite 7758)

Schritte und Transitionen neu nummerieren (Seite 7759)

Schritte und Transitionen umbenennen (Seite 7761)

Schritte kommentieren (Seite 7762)

Schritte und Transitionen kopieren und einfügen (Seite 7763)

Schritte überspringen (Seite 7764)

17.5.7.9 Übersetzungen der Schritt- und Transitionsnamen anzeigen und editieren

Sie können sich die Namen der Schritte und Transitionen in allen Projektsprachen anzeigen lassen, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben und deren Übersetzungen bereits vorliegen. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Wenn Sie den Namen des Schritts oder der Transition direkt im Programmiereditor ändern und Sie keine Änderungen der Namen im Register "Texte" vorgenommen haben, dann wird

der Name in allen Projektsprachen automatisch angepasst. Das ist zum Beispiel von Vorteil, wenn Sie sprachneutrale Schritt- und Transitionsnamen verwenden.

Voraussetzung

Sie haben mindestens einen Schritt und eine Transition in einem GRAPH-FB der Version 5.0 angelegt und der GRAPH-FB ist geöffnet.

Vorgehen

Um sich den Namen eines Schritts oder einer Transition in allen Projektsprachen anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Klicken Sie den gewünschten Schritt oder die Transition an, deren Übersetzungen Sie sich anzeigen lassen wollen.
   Wenn Sie sich die Übersetzungen mehrerer Schritte oder Transitionen gleichzeitig anzeigen lassen wollen, dann müssen Sie die entsprechenden Schritte und Transitionen markieren.
   Wenn Sie sich die Übersetzungen aller Schritte und Transitionen gleichzeitig anzeigen lassen wollen, dann klicken Sie außerhalb der Schrittkette in den GRAPH-FB.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Öffnen Sie das Register "Texte".
   Wenn Ihnen die Namen der Schritte und Transitionen nur in der aktuellen Oberflächensprache angezeigt werden, dann haben Sie noch keine Projektsprachen ausgewählt bzw. es sind noch keine Übersetzungen vorhanden. Sie können die Übersetzungen entweder direkt in die Tabelle im Register "Texte" eintragen oder Sie können die Übersetzungen für die einzelnen Projektsprachen in die globale Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

Ergebnis

Im untergeordneten Register "Texte" werden Ihnen die Namen der markierten Schritte und Transitionen in den gewünschten Projektsprachen und, soweit vorhanden, die Übersetzungen der Namen angezeigt.

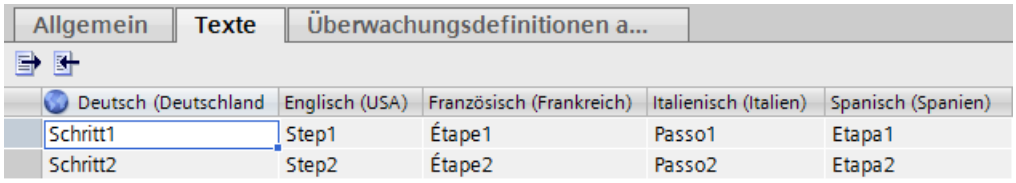

Deutsch (Deutschland	Englisch (USA)	Französisch (Frankreich)	Italienisch (Italien)	Spanisch (Spanien)
Schritt1	Step1	Étape1	Passo1	Etapa1
Schritt2	Step2	Étape2	Passo2	Etapa2

17.5.8 Verzweigungen verwenden

17.5.8.1 Alternativzweige erstellen

Voraussetzung

Eine Ablaufkette ist in der Kettenansicht geöffnet.

Vorgehen

Um einen Alternativzweig zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Schritt, nach dem Sie einen Alternativzweig einfügen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" den Befehl "Alternativzweig öffnen".

Oder:

1. Öffnen Sie in der Task Card "Anweisungen" die Palette "Einfache Anweisungen".
2. Öffnen Sie den Ordner "Graph Struktur".
3. Ziehen Sie das Element "Alternativzweig" per Drag & Drop an die gewünschte Stelle in der Ablaufkette. Erlaubte Stellen werden durch ein grünes Quadrat angezeigt.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Simultanzweig erstellen (Seite 7767)

Verzweigungen schließen (Seite 7768)

Verzweigungen löschen (Seite 7769)

17.5.8.2 Simultanzweig erstellen

Voraussetzungen

Eine Ablaufkette ist in der Kettenansicht geöffnet.

Vorgehen

Um einen Simultanzweig zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Transition, nach der Sie einen Simultanzweig einfügen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" den Befehl "Simultanzweig öffnen".

Oder:

1. Öffnen Sie in der Task Card "Anweisungen" die Palette "Einfache Anweisungen".
2. Öffnen Sie den Ordner "Graph Struktur".
3. Ziehen Sie das Element "Simultanzweig" per Drag & Drop an die gewünschte Stelle in der Ablaufkette. Erlaubte Stellen werden durch ein grünes Quadrat angezeigt.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Alternativzweige erstellen (Seite 7767)

Verzweigungen schließen (Seite 7768)

Verzweigungen löschen (Seite 7769)

17.5.8.3 Verzweigungen schließen

Sie können Simultan- und Alternativzweige folgendermaßen schließen:

- Sie können sie mit ihrem Elternzweig zusammenführen.
- Sie können Sie mit einem Kettenende schließen.
- Sie können Sie mit einem Sprung schließen.

In einem Simultanzweig können Sie ein Kettenende oder einen Sprung allerdings nur nach einer Transition einfügen.

Voraussetzung

Ein Alternativ- oder Simultanzweig ist vorhanden.

Zweige mit dem Elternzweig zusammenführen

Um einen offenen Zweig mit seinem Elternzweig zusammenzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Doppelpfeil am Ende des Zweigs.
2. Drücken und halten Sie die linke Maustaste.
   Eine gestrichelte Linie wird eingeblendet, sobald Sie den Mauszeiger bewegen.
3. Ziehen Sie die gestrichelte Linie auf eine geeignete Stelle im Netzwerk. Erlaubte Stellen werden durch ein grünes Quadrat angezeigt.
4. Lassen Sie die linke Maustaste los.

Zweige mit einem Kettenende schließen

Um einen Zweig mit einem Kettenende zu schließen, gehen Sie folgendermaßen vor:

1. Fügen Sie in einem Simultanzweig bei Bedarf eine Transition am Ende des Zweigs ein.
2. Klicken Sie mit der rechten Maustaste auf den Doppelpfeil am Ende des Zweigs.
3. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element "Kettenende".

Zweige mit einem Sprung schließen

Um einen Zweig mit einem Sprung zu schließen, gehen Sie folgendermaßen vor:

1. Fügen Sie in einem Simultanzweig bei Bedarf eine Transition am Ende des Zweigs ein.
2. Klicken Sie mit der rechten Maustaste auf den Doppelpfeil am Ende des Zweigs.
3. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element "Sprung". Eine Übersicht mit allen vorhandenen Schritten des GRAPH-Funktionsbausteins wird geöffnet.
4. Wählen Sie einen Schritt als Sprungziel aus.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Alternativzweige erstellen (Seite 7767)

Simultanzweig erstellen (Seite 7767)

Verzweigungen löschen (Seite 7769)

17.5.8.4 Verzweigungen löschen

Voraussetzung

Ein Alternativ- oder Simultanzweig ist vorhanden.

Vorgehen

Um eine Verzweigung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Verzweigung, die Sie löschen möchten.
2. Klicken Sie mit der rechten Maustaste auf die selektierte Verzweigung.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Alternativzweige erstellen (Seite 7767)

Simultanzweig erstellen (Seite 7767)

Verzweigungen schließen (Seite 7768)

17.5.9 Sprünge verwenden

17.5.9.1 Sprünge einfügen

Sie können Sprünge nur am Ende einer Ablaufkette oder eines Zweigs einfügen. In einem Simultanzweig muss dazu das letzte Element eine Transition sein.

Voraussetzung

Eine Ablaufkette ist in der Kettenansicht geöffnet.

Vorgehen

Um einen Sprung in eine Ablaufkette oder einen Zweig einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie in einem Simultanzweig bei Bedarf eine Transition am Ende des Zweigs ein.
2. Klicken Sie mit der rechten Maustaste auf den Doppelpfeil am Ende der Ablaufkette oder des Zweigs.
3. Wählen Sie im Kontextmenü im Untermenü "Element einfügen" das Element "Sprung". Eine Übersicht mit allen vorhandenen Schritten des GRAPH-Funktionsbausteins wird geöffnet.
4. Wählen Sie einen Schritt als Sprungziel aus.

Ergebnis

Der Sprung wird eingefügt. Sprung und Sprungziel werden in der Ablaufkette als Pfeile dargestellt, wobei beim Sprungziel die Absprungtransition und beim Absprung der Zielschritt angezeigt wird.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Sprungziel ändern (Seite 7770)

Zum Sprungziel navigieren (Seite 7771)

Sprünge löschen (Seite 7772)

17.5.9.2 Sprungziel ändern

Sie können das Ziel eines Sprungs nachträglich ändern.

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.
- Ein Sprung ist vorhanden.

Vorgehen

Um das Sprungziel nachträglich zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie doppelt auf die Schrittnummer des aktuellen Sprungziels.
   Eine Übersicht mit allen vorhandenen Schritten des GRAPH-Funktionsbausteins wird geöffnet.
2. Wählen Sie einen Schritt als neues Sprungziel aus.

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Sprünge einfügen (Seite 7770)

Zum Sprungziel navigieren (Seite 7771)

Sprünge löschen (Seite 7772)

17.5.9.3 Zum Sprungziel navigieren

Sie können von einem Sprung aus zu dem Schritt navigieren, den Sie als Sprungziel festgelegt haben.

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.
- Ein Sprung ist vorhanden.

Vorgehen

Um von einem Sprung zu dem Schritt zu navigieren, der als Ziel festgelegt ist, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Sprung.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Sprungziel".

Siehe auch

Elemente einer Ablaufkette (Seite 7688)

Sprünge einfügen (Seite 7770)

Sprungziel ändern (Seite 7770)

Sprünge löschen (Seite 7772)

17.5.9.4 Sprünge löschen

Voraussetzung

- Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.
- Ein Sprung ist vorhanden.

Vorgehen

Um einen Sprung zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Sprung, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Elemente einer Ablaufkette (Seite 7688)
Sprünge einfügen (Seite 7770)
Sprungziel ändern (Seite 7770)
Zum Sprungziel navigieren (Seite 7771)

17.6 Aktionen programmieren

17.6.1 Aktionen einfügen

Einführung

Sie können Aktionen festlegen, die bei der Bearbeitung des Schritts abgearbeitet werden. Um Aktionen zu kommentieren oder Bausteinaufrufe anzupassen, können Sie eine mehrzeilige Eingabe aktivieren. Standardmäßig erfolgt eine einzeilige Eingabe. Sie können Aktionen entweder in der Einzelschrittansicht oder in der Kettenansicht einfügen.

Für einen besseren Überblick über Ihre programmierten Aktionen können Sie sich für die Ereignisse und Kennungen Beschreibungen anzeigen lassen.

In Aktionen können Sie einem Operanden den Wert eines anderen Operanden oder eines mathematischen Ausdrucks oder das Ergebnis einer Anweisung zuweisen. Verwenden Sie dabei die folgende Syntax:

- Wert eines Operanden zuweisen: A:=B
- Wert eines mathematischen Ausdrucks zuweisen: A:=B<Operator>C
  Sie können die folgenden Operatoren verwenden: + (Addition), - (Subtraktion), * (Multiplikation), / (Division), % (Modulo), AND (Und-Verknüpfung), OR (Oder-Verknüpfung), XOR (Exklusiv-Oder-Verknüpfung)
- Ergebnis einer Anweisung zuweisen: A:=<Anweisung>(B)
  Die verfügbaren Anweisungen entnehmen Sie der Task Card "Anweisungen".

Beachten Sie, dass die Datentypen der Operanden zueinander kompatibel sein müssen.

Aktionen in der Einzelschrittansicht einfügen

Um eine Aktion in der Einzelschrittansicht einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Einzelschrittansicht.
2. Falls die Aktionen zugeklappt sind, klicken Sie auf den kleinen Pfeil vor "Aktionen".
   Aus dem Pfeil-nach-rechts wird ein Pfeil-nach-unten und die Tabelle mit den Aktionen wird angezeigt.
3. Wenn Sie die mehrzeilige Eingabe aktivieren möchten, klicken Sie mit der rechten Maustaste in die Zeile der neuen Aktion und aktivieren Sie im Kontextmenü den Befehl "Mehrzeilige Eingabe zulassen".
4. Wenn Sie die neue Aktion mit einem Interlock verknüpfen möchten, klicken Sie in die Zelle der Spalte "Interlock" und wählen in der Klappliste den Eintrag "-(C)- - Interlock".
5. Wenn Sie die neue Aktion mit einem Ereignis verknüpfen möchten, klicken Sie in die Zelle der Spalte "Ereignis" und wählen in der Klappliste das entsprechende Ereignis.

6. Klicken Sie in die Zelle der Spalte "Kennung" und wählen in der Klappliste die Kennung der neuen Aktion.
7. Geben Sie in der Spalte "Aktion" die Aktion an, die ausgeführt werden soll. Sie haben folgende Möglichkeiten:
   – Sie können die Platzhalter mit den Operanden oder Werten ersetzen, die Sie für die Aktion verwenden möchten. Dabei können Sie sie auch per Drag & Drop oder über die Autovervollständigung einfügen.
   – Sie können Anweisungen aus der Task Card "Anweisungen" verwenden. Ziehen Sie sie dazu aus der Task Card per Drag & Drop in die Spalte "Aktion".
   – Sie können Bausteine per Drag & Drop aus der Projektnavigation in die Spalte "Aktion" ziehen, um sie aufzurufen.

Hinweis

Beim Aufruf von datentypabhängigen Anweisungen müssen Sie einen Datentyp angeben. Sie können den Datentyp entweder manuell eingeben oder aus einer Klappliste auswählen, in dem Sie auf das kleine grüne Rechteck klicken, das unter dem vorhandenen Datentyp angezeigt wird.

Aktionen in der Kettenansicht einfügen

Um eine Aktion in der Kettenansicht einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Kettenansicht.
2. Erweitern Sie den Schritt, in den Sie Aktionen einfügen möchten. Klicken Sie dazu entweder auf die Schaltfläche in der rechten oberen Ecke des Schritts oder selektieren Sie den Schritt und drücken <Enter>.
3. Wenn Sie die mehrzeilige Eingabe aktivieren möchten, klicken Sie mit der rechten Maustaste in die Zeile der neuen Aktion und aktivieren Sie im Kontextmenü den Befehl "Mehrzeilige Eingabe zulassen".
4. Wenn Sie die neue Aktion mit einem Interlock verknüpfen möchten, klicken Sie in die Zelle der Spalte "Interlock" und wählen in der Klappliste den Eintrag "-(C)- - Interlock".
5. Wenn Sie die neue Aktion mit einem Ereignis verknüpfen möchten, klicken Sie in die Zelle der Spalte "Ereignis" und wählen in der Klappliste das entsprechende Ereignis.

6. Klicken Sie in die Zelle der Spalte "Kennung" und wählen in der Klappliste die Kennung der neuen Aktion.
7. Geben Sie in der Spalte "Aktion" die Aktion an, die ausgeführt werden soll. Sie haben folgende Möglichkeiten:
   - Sie können die Platzhalter mit den Operanden oder Werten ersetzen, die Sie für die Aktion verwenden möchten. Dabei können Sie sie auch per Drag & Drop oder über die Autovervollständigung einfügen.
   - Sie können Anweisungen aus der Task Card "Anweisungen" verwenden. Ziehen Sie sie dazu aus der Task Card per Drag & Drop in die Spalte "Aktion".
   - Sie können Bausteine per Drag & Drop aus der Projektnavigation in die Spalte "Aktion" ziehen, um sie aufzurufen.

Hinweis

Beim Aufruf von datentypabhängigen Anweisungen müssen Sie einen Datentyp angeben. Sie können den Datentyp entweder manuell eingeben oder aus einer Klappliste auswählen, in dem Sie auf das kleine grüne Rechteck klicken, das unter dem vorhandenen Datentyp angezeigt wird.

Beschreibung der Ereignisse anzeigen

Um die Beschreibung der Ereignisse anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die Aktionentabelle.
2. Aktivieren Sie im Kontextmenü den Befehl "Beschreibung der Ereignisse anzeigen".

Beschreibung der Kennungen anzeigen

Um die Beschreibung der Kennungen anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die Aktionentabelle.
2. Aktivieren Sie im Kontextmenü den Befehl "Beschreibung der Kennungen anzeigen".

Siehe auch

Aktionen ändern (Seite 7778)

Aktionen kopieren, ausschneiden und einfügen (Seite 7779)

Aktionen verschieben (Seite 7780)

Aktionen löschen (Seite 7781)

Aktionen (Seite 7695)

17.6.2 Datentyp einer GRAPH-Anweisung festlegen

17.6.2.1 Grundlagen zu den Datentypen von GRAPH-Anweisungen

Einführung

Die Verwendung von GRAPH-Anweisungen kann über die folgenden beiden Mechanismen erfolgen:

- Als Bausteinaufruf
  Bei Anweisungen, die Sie als Bausteinaufruf in Ihr Programm einfügen, können Sie den Datentyp über eine Klappliste auswählen.
- Als GRAPH-interne Funktion
  GRAPH-interne Funktionen weisen Sie direkt einer Variablen zu. Dabei muss der Operand einen Datentyp haben, den die Funktion unterstützt. Eine Übersicht über die zulässigen Datentypen der GRAPH-internen Funktionen bietet die folgende Tabelle.

GRAPH-interne Funktionen

Die folgende Tabelle zeigt die GRAPH-internen Funktionen und ihre Datentypen:

Funktion	S7-300/400	S7-1500
BCD_TO_INT	WORD	WORD
BCD_TO_DINT	DWORD	DWORD
INT_TO_BCD	INT	INT
DINT_TO_BCD	DINT	DINT
INT_TO_DINT	INT	INT
DINT_TO_REAL	DINT	DINT
ROUND	REAL	REAL, LREAL
TRUNC	REAL	REAL, LREAL
CEIL	REAL	REAL, LREAL
FLOOR	REAL	REAL, LREAL
NEG	INT, DINT, REAL	SINT, INT, DINT, REAL, LINT, LREAL
ABS	REAL	SINT, INT, DINT, LINT, REAL, LREAL
SQR	REAL	REAL, LREAL
SQRT	REAL	REAL, LREAL
LN	REAL	REAL, LREAL
EXP	REAL	REAL, LREAL
SIN	REAL	REAL, LREAL
COS	REAL	REAL, LREAL
TAN	REAL	REAL, LREAL
ASIN	REAL	REAL, LREAL
ACOS	REAL	REAL, LREAL
ATAN	REAL	REAL, LREAL

Funktion	S7-300/400	S7-1500
NOT	WORD, DWORD	USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, BYTE, WORD, DWORD, LWORD
RLDA	DWORD	-
RRDA	DWORD	-
SWAP	WORD, DWORD	WORD, DWORD, LWORD
AND	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
OR	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
XOR	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
SHL	WORD, DWORD	BYTE, USINT, SINT, WORD, UINT, INT, DWORD, UDINT, DINT, LWORD, ULINT, LINT
SHR	WORD, DWORD	BYTE, WORD, DWORD, USINT, UINT, UDINT, SINT, INT, DINT, LWORD, ULINT, LINT
ROL	DWORD	BYTE, WORD, DWORD, LWORD
ROR	DWORD	BYTE, WORD, DWORD, LWORD
MOD	DINT	USINT, UINT, UDINT, SINT, INT, DINT, ULINT, LINT
-: Funktion nicht verfügbar		

17.6.2.2 Datentyp einer GRAPH-Anweisung ändern

GRAPH-Anweisungen, die Sie über einen Bausteinaufruf in Ihr Programm einfügen, werden mit einem voreingestellten Datentyp eingefügt. Diesen Datentyp können Sie bei Bedarf ändern. Änderbare Datentypen sind mit einem grünen Rechteck unter dem Anfangsbuchstaben des Datentyps gekennzeichnet. Bei den GRAPH-internen Funktionen können Sie den Operanden ändern.

Datentyp einer aufgerufenen Anweisung ändern

Um den Datentyp einer aufgerufenen Anweisung zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Datentyp der Anweisung.
   Die Klappliste für die Datentypauswahl wird geöffnet.
2. Klicken Sie auf den Datentyp, den Sie verwenden möchten.

Datentyp einer GRAPH-internen Funktion ändern

Um den Datentyp einer GRAPH-internen Funktion zu ändern, gehen Sie folgendermaßen vor:

1. Geben Sie als Operand einen Wert oder eine Variable ein. Beachten Sie dabei den zulässigen Wertebereich für den Operanden.

Oder:

1. Ziehen Sie mit gedrückter rechter Maustaste eine Variable vom zulässigen Datentyp über den Operanden.
2. Warten Sie, bis der aktuelle Operand eine grüne Umrandung aufweist.
3. Lassen Sie die Maustaste los.

17.6.3 Aktionen ändern

Sie können für Aktionen folgende Änderungen vornehmen:

- Aktion mit einem Interlock verknüpfen oder die Verknüpfung zu einem Interlock aufheben
- Aktion mit einem Ereignis verknüpfen oder die Verknüpfung mit einem Ereignis aufheben oder ändern
- Aktionskennung ändern
- Aktion ändern, z. B. Operanden tauschen oder eine andere Anweisung verwenden

Beachten Sie, dass bestimmte Aktionskennungen die Verwendung von Ereignissen erfordern und dass Sie bei diesen Kennungen die Verknüpfung zu einem Ereignis nicht aufheben können.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um eine Aktion zu ändern, gehen Sie folgendermaßen vor:

1. Wenn Sie eine Aktion mit einem Interlock verknüpfen möchten, klicken Sie in die Zelle der Spalte "Interlock" und wählen in der Klappliste den Eintrag "-(C)- - Interlock".
2. Wenn Sie die Verknüpfung mit einem Interlock aufheben möchten, klicken Sie in die Zelle der Spalte "Interlock" und wählen in der Klappliste den Eintrag " - Keine Bedingung".
3. Wenn Sie die Aktion mit einem Ereignis verknüpfen oder das verknüpfte Ereignis ändern möchten, klicken Sie in die Zelle der Spalte "Ereignis" und wählen in der Klappliste das entsprechende Ereignis.
4. Wenn Sie die Verknüpfung mit einem Ereignis aufheben möchten, klicken Sie in die Zelle der Spalte "Ereignis" und wählen in der Klappliste den Eintrag " - Kein Ereignis". Wenn die Aktionskennung ein Ereignis erfordert, wird die Zelle rot hinterlegt und "<???>" wird angezeigt. Weisen Sie in diesem Fall wieder ein Ereignis zu oder wählen Sie eine andere Aktionskennung.

5. Wenn Sie die Aktionskennung ändern möchten, klicken Sie in die Zelle der Spalte "Kennung" und wählen in der Klappliste die neue Kennung der Aktion.
6. Wenn Sie die Aktion ändern möchten, klicken Sie in die Aktion in der Spalte "Aktion" und nehmen eine der folgenden Änderungen vor:
   - Ändern Sie den oder die Operanden der Aktion.
   - Um den Datentyp eines Operanden zu ändern, klicken Sie auf das kleine grüne Rechteck und wählen in der Klappliste den neuen Datentyp aus.
   - Um eine andere Anweisung zu verwenden, löschen Sie die bestehende Aktion und ziehen eine andere Anweisung aus der Task Card "Anweisungen" in die Spalte "Aktion". Geben Sie die Operanden ein.
   - Um einen anderen Baustein aufzurufen, löschen Sie die bestehende Aktion und ziehen per Drag & Drop einen Baustein aus der Projektnavigation in die Spalte "Aktion", um ihn aufzurufen.

Siehe auch

Aktionen einfügen (Seite 7773)

Aktionen kopieren, ausschneiden und einfügen (Seite 7779)

Aktionen verschieben (Seite 7780)

Aktionen löschen (Seite 7781)

Aktionen (Seite 7695)

17.6.4 Aktionen kopieren, ausschneiden und einfügen

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Aktionen kopieren

Um eine Aktion zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die erste Zelle der Aktion, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Aktionen ausschneiden

Um eine Aktion auszuschneiden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die erste Zelle der Aktion, die Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

Aktionen aus der Zwischenablage einfügen

Um eine Aktion aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Stelle, an der Sie die Aktion einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

Aktionen einfügen (Seite 7773)

Aktionen ändern (Seite 7778)

Aktionen verschieben (Seite 7780)

Aktionen löschen (Seite 7781)

Aktionen (Seite 7695)

17.6.5 Aktionen verschieben

Sie können Aktionen innerhalb der Aktionentabelle verschieben.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um eine Aktion innerhalb der Aktionentabelle zu verschieben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die erste Zelle der Aktion, die Sie verschieben möchten, und halten Sie die Maustaste gedrückt.
   Die Aktion wird selektiert.
2. Ziehen Sie die Aktion per Drag & Drop an die gewünschte Position.

Siehe auch

Aktionen einfügen (Seite 7773)

Aktionen ändern (Seite 7778)

Aktionen kopieren, ausschneiden und einfügen (Seite 7779)

Aktionen löschen (Seite 7781)

Aktionen (Seite 7695)

17.6.6 Aktionen löschen

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um eine Aktion zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die erste Zelle der Aktion, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Aktionen einfügen (Seite 7773)
Aktionen ändern (Seite 7778)
Aktionen kopieren, ausschneiden und einfügen (Seite 7779)
Aktionen verschieben (Seite 7780)
Aktionen (Seite 7695)

17.7 Bedingungen programmieren

17.7.1 Interlocks programmieren

Sie können bis zu 32 Anweisungen in Ihrem Interlocknetzwerk verwenden, an denen maximal 32 Operanden verschaltet sind. In KOP können die Anweisungen sowohl in einen Strompfad als auch in die Stromschiene eingefügt werden.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um Interlocks zu programmieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Interlocknetzwerk, indem Sie auf den kleinen Pfeil vor "Interlock -(c)-" klicken.
2. Öffnen Sie die Task Card "Anweisungen".
3. Navigieren Sie zu der Anweisung, die Sie einfügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop an die gewünschte Stelle im Netzwerk.
5. Klicken Sie in der Aktionstabelle in der Spalte "Interlock" in die Zelle der Aktion, die Sie mit dem Interlock verknüpfen möchten, und wählen Sie in der Klappliste den Eintrag "-(C)- - Interlock".

Oder:

1. Öffnen Sie das Interlocknetzwerk, indem Sie auf den kleinen Pfeil vor "Interlock -(c)-" klicken.
2. Öffnen Sie die Task Card "Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu "KOP/FUP > Allgemein > Leerbox".
4. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
5. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.
6. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur die zulässigen Elemente.
7. Wählen Sie das gewünschte Element aus der Klappliste.
8. Klicken Sie in der Aktionstabelle in der Spalte "Interlock" in die Zelle der Aktion, die Sie mit dem Interlock verknüpfen möchten, und wählen Sie in der Klappliste den Eintrag "-(C)- - Interlock".

Siehe auch

Bedingungen (Seite 7706)

Supervisionsbedingungen einfügen (Seite 7783)

Transitionen programmieren (Seite 7789)

Permanente Anweisungen programmieren (Seite 7790)

Variableninformationen ein- oder ausblenden (Seite 7798)

Programmiersprache in Netzwerken festlegen (Seite 7733)

17.7.2 Supervisionen programmieren

17.7.2.1 Supervisionsbedingungen einfügen

Sie können bis zu 32 Anweisungen in Ihrem Supervisionsnetzwerk verwenden, an denen maximal 32 Operanden verschaltet sind. In KOP können die Anweisungen sowohl in einen Strompfad als auch in die Stromschiene eingefügt werden.

Hinweis

Für CPUs der Baureihe S7-1500 können Sie eine Schrittzeit-Überwachung programmieren. Dazu stehen Ihnen die beiden Anweisungen "Größer maximale Schrittaktivierungszeit" (CMP>T_MAX), "Größer Warnungszeit" (CMP>T_WARN) und eine Warnungsspule (-(w)-) zur Verfügung. Die Vergleichswerte für die beiden Anweisungen können Sie entweder manuell festlegen oder über den Lernmodus vom System ermitteln lassen.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um Supervisionen zu programmieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Supervisionsnetzwerk, indem Sie auf den kleinen Pfeil vor "Supervision -(v)-" klicken.
2. Öffnen Sie die Task Card "Anweisungen".
3. Navigieren Sie zu der Anweisung, die Sie einfügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop an die gewünschte Stelle im Netzwerk.

Oder:

1. Öffnen Sie das Supervisionsnetzwerk, indem Sie auf den kleinen Pfeil vor "Supervision -(v)-" klicken.
2. Öffnen Sie die Task Card "Anweisungen".

3. Navigieren Sie in der Palette "Einfache Anweisungen" zu "KOP/FUP > Allgemein > Leerbox".
4. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
5. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.
6. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur die zulässigen Elemente.
7. Wählen Sie das gewünschte Element aus der Klappliste.

Siehe auch

Bedingungen (Seite 7706)

Interlocks programmieren (Seite 7782)

Transitionen programmieren (Seite 7789)

Permanente Anweisungen programmieren (Seite 7790)

Variableninformationen ein- oder ausblenden (Seite 7798)

Programmiersprache in Netzwerken festlegen (Seite 7733)

17.7.2.2 Schrittzeit-Überwachung verwenden

Grundlagen zur Schrittzeit-Überwachung

Einführung

Sie können mithilfe von Supervisionsbedingungen die Schrittaktivierungszeit überwachen, d. h. die Zeit, die seit der Aktivierung des Schritts vergangen ist. Dazu können Sie die maximale Zeitdauer "T_MAX" festlegen und diese als Vergleichswert verwenden. Um den Vergleich auszuführen, steht Ihnen die Anweisung "Größer maximale Schrittaktivierungszeit" (CMP>T_MAX) zur Verfügung. Wenn die maximale Schrittaktivierungszeit überschritten wird, wird ein Supervisionsfehler ausgelöst und eine Supervisionsmeldung ausgegeben, falls Sie dies aktiviert haben. Um eine Meldung zu erhalten, bevor es zum Supervisionsfehler kommt, können Sie eine Warnungszeit "T_WARN" festlegen und diese ebenfalls über eine Supervisionsbedingung überwachen. Dazu können Sie die Anweisung "Größer Warnungszeit" (CMP>T_WARN) und eine Warnungsspule verwenden.

Als Werte für "T_MAX" und "T_WARN" werden zunächst die Werte verwendet, die in den Einstellungen unter "PLC-Programmierung > GRAPH > Voreinstellung für Zeitüberwachungen" festgelegt sind. Sie haben die folgenden Möglichkeiten, die Zeiten für "T_MAX" und "T_WARN" zu ändern:

- Manuell über den Dialog "Zeit definieren" (Seite 7786)
- Über den Lernmodus der Schnittstelle (Seite 7787)
- Über den Lernmodus beim Testen (Seite 8048)

Wenn Sie den Lernmodus verwenden, wird die Aktivierungszeit der Schritte beim Programmablauf gemessen. Je länger Sie den Lernmodus aktiviert lassen, umso genauer kann die Aktivierungszeit ermittelt werden. Die Zeiten für "T_MAX" und "T_WARN" ergeben sich aus den Summen der ermittelten Zeiten und additiven Schwellenwerten, die Sie festlegen können.

Wenn Sie die Zeiten über den Lernmodus beim Testen ermitteln lassen, können Sie die gelernten Zeiten bei Bedarf wieder zurücksetzen lassen. Dadurch erhalten "T_MAX" und "T_WARN" im Instanzdatenbaustein wieder ihre Defaultwerte.

Hinweis

Für GRAPH-Bausteine mit einer Version kleiner 2.0 steht Ihnen der Lernmodus für das Ermitteln von "T_MAX" und "T_WARN" nicht zur Verfügung. Sie können die Version in den Eigenschaften des GRAPH-Bausteins unter "Allgemein > Baustein" ändern.

Siehe auch

Schrittzeit-Überwachung programmieren (Seite 7785)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2162)

CMP>T_WARN: Größer Warnungszeit (Seite 2164)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2174)

CMP>T_WARN: Größer Warnungszeit (Seite 2176)

Schrittzeit-Überwachung programmieren

Voraussetzung

- Für den GRAPH-Funktionsbaustein ist die Meldungsanzeige aktiviert.
  Siehe auch: Meldungsanzeige aktivieren und deaktivieren (Seite 7806)
- Für den GRAPH-Funktionsbaustein sind die Meldungstexte definiert.
  Siehe auch: Meldungen bearbeiten (Seite 7806)
- Der GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.

Vorgehen

Um eine Schrittzeit-Überwachung zu programmieren, die eine Supervision und eine Warnung enthält, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Supervisionsnetzwerk, indem Sie auf den kleinen Pfeil vor "Supervision -(V)-" klicken.
2. Öffnen Sie die Task Card "Anweisungen".
3. Fügen Sie die Anweisung "CMP>T_MAX" in den Strompfad mit der Supervisionsspule "-(V)-" ein.
4. Fügen Sie die Anweisung "CMP>T_WARN" in einen neuen Strompfad ein.

5. Fügen Sie die Anweisung "-(W)-" in den neuen Strompfad ein.
6. Legen Sie die Zeiten für "T_MAX" und "T_WARN" entweder manuell oder über den Lernmodus fest.

Ergebnis

Ist die Schrittaktivierungszeit größer als die Zeit, die für "T_WARN" festgelegt wurde, wird eine Warnung angezeigt. Überschreitet sie auch die Zeit von "T_MAX", wird die Supervisionsmeldung ausgegeben.

Siehe auch

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Zeiten für T_MAX und T_WARN manuell festlegen (Seite 7786)

Zeiten für T_MAX und T_WARN über den Lernmodus der Schnittstelle ermitteln (Seite 7787)

Zeiten für T_MAX und T_WARN beim Testen ermitteln (Seite 8048)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2162)

CMP>T_WARN: Größer Warnungszeit (Seite 2164)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2174)

CMP>T_WARN: Größer Warnungszeit (Seite 2176)

Zeiten für T_MAX und T_WARN manuell festlegen

Voraussetzung

- Der GRAPH-Funktionsbaustein ist in der Einzelschrittansicht geöffnet.
- Für den Schritt ist eine Schrittzeit-Überwachung programmiert.

Vorgehen

Um die Zeiten für "T_MAX" und "T_WARN" manuell festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Supervisionsnetzwerk.
2. Wählen Sie im Kontextmenü den Befehl "Zeit definieren".
   Der Dialog "Zeit definieren" wird geöffnet.
3. Geben Sie die Zeiten für "T_MAX" und "T_WARN" ein.
4. Bestätigen Sie Ihre Eingaben mit "OK".
   Die eingegebenen Zeiten werden in der Bausteinschnittstelle als Defaultwerte für die statischen Schrittparameter "T_MAX" und "T_WARN" übernommen.

Siehe auch

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Zeiten für T_MAX und T_WARN über den Lernmodus der Schnittstelle ermitteln (Seite 7787)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2162)

CMP>T_WARN: Größer Warnungszeit (Seite 2164)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2174)

CMP>T_WARN: Größer Warnungszeit (Seite 2176)

Zeiten für T_MAX und T_WARN über den Lernmodus der Schnittstelle ermitteln

Sie können die Zeiten für "T_MAX" und "T_WARN" direkt über den Bausteinaufruf des GRAPH-Funktionsbausteins ermitteln lassen. Beachten Sie jedoch die folgenden Einschränkungen gegenüber der Verwendung des Lernmodus beim Testen, den Sie in der Task Card "Testen" starten:

- Die ermittelten Werte werden nur im Online-Instanzdatenbaustein gespeichert.
- Sie können die gelernten Zeiten nicht zurücksetzen lassen.

Damit "T_MAX" und "T_WARN" größer sind als die gelernten Aktivierungszeiten, können Sie additive Schwellenwerte definieren, die zu den gelernten Zeiten addiert werden. Geben Sie für die Warnung einen kleineren Wert als für die Supervision ein, damit die Warnung vor der Supervision angezeigt wird.

Voraussetzung

Der GRAPH-Funktionsbaustein wird im Programm aufgerufen.

Additive Schwellenwerte festlegen

Um additive Schwellenwerte festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie im Baustein, der den GRAPH-Funktionsbaustein aufruft, die Anweisung "MOVE" ein.
2. Geben Sie am Parameter "IN" den additiven Schwellenwert für die Supervision als Prozentwert ein.
3. Geben Sie am Parameter "OUT1" den statischen Parameter "RT_DATA > TRESHOLD_SUP" des Instanzdatenbausteins ein.
4. Fügen Sie eine weitere Anweisung "MOVE" ein.
5. Geben Sie am Parameter "IN" den additiven Schwellenwert für die Warnung als Prozentwert ein.
6. Geben Sie am Parameter "OUT1" den statischen Parameter "RT_DATA > TRESHOLD_WARN" des Instanzdatenbausteins ein.
   Die additiven Schwellenewerte werden bei der Berechnung von "T_MAX" und "T_WARN" im Lernmodus berücksichtigt.

Lernmodus aktivieren

Um den Lernmodus über die Schnittstelle zu aktivieren, gehen Sie folgendermaßen vor:

1. Stellen Sie den Schnittstellenparametersatz auf "Maximale Schnittstellenparameter" oder fügen Sie in die Bausteinschnittstelle des GRAPH-Funktionsbausteins manuell die beiden Parameter ein:
   - Eingangsparameter "EN_LMODE"
   - Ausgangsparameter "LMODE_ENABLED"
2. Öffnen Sie den aufrufenden Baustein.
3. Verknüpfen Sie den Eingangsparameter "EN_LMODE" mit einer Variablen.
4. Laden Sie den Baustein ins Gerät.
5. Kicken Sie in der Funktionsleiste auf "Beobachten ein/aus", um das Testen mit Programmstatus für den aufrufenden Baustein zu starten.
6. Klicken Sie mit der rechten Maustaste auf die Variable, die Sie mit dem Eingangsparameter "EN_LMODE" verknüpft haben.
7. Wählen Sie im Kontextmenü den Befehl "Steuern > Steuern auf 1", um eine steigende Flanke für den Eingangsparameter "EN_LMODE" zu erhalten.
   Der Signalszustand von "LMODE_ENABLED" wird auf "1" gesetzt und der Lernmodus wird aktiviert.
8. Warten Sie, bis jeder Schritt im GRAPH-Funktionsbaustein mindestens einmal aktiv war.
9. Klicken Sie erneut mit der rechten Maustaste auf die Variable, die Sie mit dem Eingangsparameter "EN_LMODE" verknüpft haben.
10. Wählen Sie im Kontextmenü den Befehl "Steuern > Steuern auf 0", um den Eingangsparameter "EN_LMODE" zurückzusetzen.
11. Klicken Sie erneut mit der rechten Maustaste auf die Variable, die Sie mit dem Eingangsparameter "EN_LMODE" verknüpft haben.
12. Wählen Sie im Kontextmenü den Befehl "Steuern > Steuern auf 1", um eine steigende Flanke für den Eingangsparameter "EN_LMODE" zu erhalten.
    Der Lernmodus wird deaktiviert. Die gelernten Zeiten werden in den Online-Instanzdatenbaustein des GRAPH-Funktionsbausteins übernommen.

Siehe auch

Grundlagen zur Schrittzeit-Überwachung (Seite 7784)

Schrittzeit-Überwachung programmieren (Seite 7785)

Zeiten für T_MAX und T_WARN manuell festlegen (Seite 7786)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2162)

CMP>T_WARN: Größer Warnungszeit (Seite 2164)

CMP>T_MAX: Größer maximale Schrittaktivierungszeit (Seite 2174)

CMP>T_WARN: Größer Warnungszeit (Seite 2176)

17.7.3 Transitionen programmieren

Sie können bis zu 32 Anweisungen in Ihrem Transitionsnetzwerk verwenden, an denen maximal 32 Operanden verschaltet sind. In KOP können die Anweisungen sowohl in einen Strompfad als auch in die Stromschiene eingefügt werden. Sie können Transitionen entweder in der Einzelschrittansicht oder in der Kettenansicht programmieren.

Transitionen in der Einzelschrittansicht programmieren

Um Transitionen in der Einzelschrittansicht zu programmieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Einzelschrittansicht.
2. Öffnen Sie das Transitionsnetzwerk, indem Sie auf den kleinen Pfeil vor "<Transitionsnummer> - <Transitionsname>" klicken.
3. Öffnen Sie die Task Card "Anweisungen".
4. Navigieren Sie zu der Anweisung, die Sie einfügen möchten.
5. Ziehen Sie die Anweisung per Drag & Drop an die gewünschte Stelle im Netzwerk.

Oder:

1. Öffnen Sie das Transitionsnetzwerk, indem Sie auf den kleinen Pfeil vor "<Transitionsnummer> - <Transitionsname>" klicken.
2. Öffnen Sie die Task Card "Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu "KOP/FUP > Allgemein > Leerbox".
4. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
5. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.
6. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur die zulässigen Elemente.
7. Wählen Sie das gewünschte Element aus der Klappliste.

Transitionen in der Kettenansicht programmieren

Um Transitionen in der Kettenansicht zu programmieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Kettenansicht.
2. Erweitern Sie die Transition, die Sie programmieren möchten. Klicken Sie dazu entweder auf die Schaltfläche neben der Transition oder selektieren Sie die Transition und drücken <Enter>.
3. Öffnen Sie die Task Card "Anweisungen".
4. Navigieren Sie zu der Anweisung, die Sie einfügen möchten.
5. Ziehen Sie die Anweisung per Drag & Drop an die gewünschte Stelle im Netzwerk.

Oder:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "KOP/FUP > Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.
5. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur die zulässigen Elemente.
6. Wählen Sie das gewünschte Element aus der Klappliste

Siehe auch

Bedingungen (Seite 7706)

Interlocks programmieren (Seite 7782)

Supervisionsbedingungen einfügen (Seite 7783)

Permanente Anweisungen programmieren (Seite 7790)

Variableninformationen ein- oder ausblenden (Seite 7798)

Programmiersprache in Netzwerken festlegen (Seite 7733)

17.7.4 Permanente Anweisungen programmieren

17.7.4.1 Arbeiten mit Netzwerken

Netzwerke verwenden

Funktion

Permanente Anweisungen werden in einem GRAPH-Funktionsbaustein innerhalb von Netzwerken programmiert. Damit eine permanente Anweisung programmiert werden kann, muss mindestens ein Netzwerk vorhanden sein. Um eine bessere Übersichtlichkeit des Programms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

Siehe auch

Netzwerke einfügen (Seite 7791)

Netzwerke selektieren (Seite 7791)

Netzwerke kopieren und einfügen (Seite 7792)

Netzwerke löschen (Seite 7793)
Netzwerke reduzieren und erweitern (Seite 7793)
Netzwerktitel einfügen (Seite 7794)
In Netzwerken navigieren (Seite 7796)

Netzwerke einfügen

Voraussetzung

Ein Baustein ist geöffnet.

Vorgehen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

Ergebnis

Ein neues, leeres Netzwerk wird in den Baustein eingefügt.

Siehe auch

Netzwerke verwenden (Seite 7790)
Netzwerke selektieren (Seite 7791)
Netzwerke kopieren und einfügen (Seite 7792)
Netzwerke löschen (Seite 7793)
Netzwerke reduzieren und erweitern (Seite 7793)
Netzwerktitel einfügen (Seite 7794)
In Netzwerken navigieren (Seite 7796)

Netzwerke selektieren

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.
   Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

Siehe auch

Netzwerke verwenden (Seite 7790)

Netzwerke einfügen (Seite 7791)

Netzwerke kopieren und einfügen (Seite 7792)

Netzwerke löschen (Seite 7793)

Netzwerke reduzieren und erweitern (Seite 7793)

Netzwerktitel einfügen (Seite 7794)

In Netzwerken navigieren (Seite 7796)

Netzwerke kopieren und einfügen

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Siehe auch

Netzwerke verwenden (Seite 7790)
Netzwerke einfügen (Seite 7791)
Netzwerke selektieren (Seite 7791)
Netzwerke löschen (Seite 7793)
Netzwerke reduzieren und erweitern (Seite 7793)
Netzwerktitel einfügen (Seite 7794)
In Netzwerken navigieren (Seite 7796)

Netzwerke löschen

Voraussetzung

Ein Netzwerk ist vorhanden.

Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Siehe auch

Netzwerke verwenden (Seite 7790)
Netzwerke einfügen (Seite 7791)
Netzwerke selektieren (Seite 7791)
Netzwerke kopieren und einfügen (Seite 7792)
Netzwerke reduzieren und erweitern (Seite 7793)
Netzwerktitel einfügen (Seite 7794)
In Netzwerken navigieren (Seite 7796)

Netzwerke reduzieren und erweitern

Voraussetzung

Ein Netzwerk ist vorhanden.

Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

Alle Netzwerke auf- und zuklappen

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

Siehe auch

Netzwerke verwenden (Seite 7790)

Netzwerke einfügen (Seite 7791)

Netzwerke selektieren (Seite 7791)

Netzwerke kopieren und einfügen (Seite 7792)

Netzwerke löschen (Seite 7793)

Netzwerktitel einfügen (Seite 7794)

In Netzwerken navigieren (Seite 7796)

Netzwerktitel einfügen

Der Netzwerktitel ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitels ist auf eine Zeile beschränkt. Sie können den Titel entweder manuell eingeben oder automatisch setzen. Beim automatischen Setzen können Sie dies für einzelne Netzwerke vornehmen oder über die Einstellungen festlegen, dass der Netzwerktitel grundsätzlich automatisch gesetzt werden soll.

Für das automatische Einfügen des Netzwerktitels wird der Kommentar des Operanden einer der folgenden Anweisungen im Netzwerk ausgewertet:

- Zuweisung
- Ausgang setzen
- Ausgang rücksetzen

Dabei wird die Anweisung verwendet, die im Netzwerk zuerst steht.

Der Netzwerktitel wird nur dann automatisch eingefügt, wenn die folgenden Bedingungen erfüllt sind:

- Das Netzwerk besitzt noch keinen Titel.
- Der Operand der Anweisung, die für den Kommentar verwendet wird, besitzt einen Kommentar.

Hinweis

Beachten Sie die folgenden Einschränkungen beim automatischen Einfügen des Netzwerktitels:

- Wenn Sie den Kommentar des Operanden nachträglich ändern, wird der Netzwerktitel nicht angepasst.
- Wenn Sie den Operanden der Anweisung wechseln, wird der Netzwerktitel nicht angepasst.
- Der Netzwerktitel wird nur durch die oben genannten schreibenden Anweisungen gesetzt.
- Ist der Operand vom Datentyp Array, wird der Kommentar des Arrays verwendet und nicht die Kommentare der Array-Elemente.
- Kommentare ungültiger Operanden werden nicht berücksichtigt.

Netzwerktitel manuell eingeben

Um einen Netzwerktitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelzeile des Netzwerks.
2. Geben Sie den Netzwerktitel ein.

Netzwerktitel automatisch setzen

Um festzulegen, dass die Netzwerktitel grundsätzlich automatisch gesetzt werden sollen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Aktivieren Sie in der Gruppe "Weitere Einstellungen" das Optionskästchen "Netzwerktitel automatisch setzen".
   Die Titel der Netzwerke werden ab diesem Zeitpunkt automatisch gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Um einen einzelnen Netzwerktitel automatisch zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste eines Netzwerks mit der rechten Maustaste auf "Netzwerk <Nummer des Netzwerks>".
2. Wählen Sie im Kontextmenü den Befehl "Netzwerktitel automatisch setzen".
   Der Titel des selektierten Netzwerks wird anhand des Kommentars des Operanden gesetzt, wenn die oben genannten Bedingungen erfüllt sind.

Siehe auch

Netzwerke verwenden (Seite 7790)

Netzwerke einfügen (Seite 7791)

Netzwerke selektieren (Seite 7791)

Netzwerke kopieren und einfügen (Seite 7792)

Netzwerke löschen (Seite 7793)

Netzwerke reduzieren und erweitern (Seite 7793)

In Netzwerken navigieren (Seite 7796)

In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".
   Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

Siehe auch

Netzwerke verwenden (Seite 7790)

Netzwerke einfügen (Seite 7791)

Netzwerke selektieren (Seite 7791)

Netzwerke kopieren und einfügen (Seite 7792)

Netzwerke löschen (Seite 7793)

Netzwerke reduzieren und erweitern (Seite 7793)

Netzwerktitel einfügen (Seite 7794)

17.7.4.2 Permanente Anweisungen einfügen

Wenn Sie eine Palette für vorausgeschaltete oder nachgeschaltete permanente Anweisungen aufklappen, enthält sie standardmäßig mindestens ein leeres Netzwerk. Sie können bis zu 32 Anweisungen je Netzwerk verwenden, an denen maximal 32 Operanden verschaltet sind. In KOP können die Anweisungen sowohl in einen Strompfad als auch in die Stromschiene eingefügt werden. Innerhalb eines GRAPH-Funktionsbausteins können Sie bis zu 250 vorausgeschaltete und 250 nachgeschaltete permanente Anweisungen verwenden.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist in der Kettenansicht geöffnet.

Vorgehen

Um permanente Anweisungen zu programmieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Navigation entweder die Palette "Vorausgeschaltete permanente Anweisungen" oder die Palette "Nachgeschaltete permanente Anweisungen".
2. Klicken Sie auf ein Netzwerk.
   Im Arbeitsbereich wird die Ansicht für vorausgeschaltete oder nachgeschaltete permanente Anweisungen geöffnet.
3. Öffnen Sie die Task Card "Anweisungen".
4. Navigieren Sie zu der Anweisung, die Sie einfügen möchten.
5. Ziehen Sie die Anweisung per Drag & Drop an die gewünschte Stelle im Netzwerk.

Oder:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.
5. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.
   Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur die zulässigen Elemente.
6. Wählen Sie das gewünschte Element aus der Klappliste.

Siehe auch

Bedingungen (Seite 7706)

Interlocks programmieren (Seite 7782)

Supervisionsbedingungen einfügen (Seite 7783)

Transitionen programmieren (Seite 7789)

Variableninformationen ein- oder ausblenden (Seite 7798)

Programmiersprache in Netzwerken festlegen (Seite 7733)

17.7.5 Variableninformationen ein- oder ausblenden

Einführung

Für permanente Anweisungen, Interlocks, Supervisionen und Transitionen können Sie die folgenden Informationen zu den verwendeten Variablen im Programmierfenster einblenden:

- Name der Variablen
- Adresse der Variablen
- Einfache oder hierarchische Kommentare zur Dokumentation der Variablen

Die Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablentabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden, werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet. Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete Bausteine wieder einblenden.

Wenn Sie die Anzeige von Variableninformationen mit hierarchischen Kommentaren wählen, werden bei strukturierten Variablen auch die Kommentare der übergeordneten Strukturebenen angezeigt. Die Anzeige erfolgt dabei in Klammern hinter dem Kommentar der Variablen, die Kommentare der einzelnen Ebenen werden jeweils durch einen Punkt getrennt. Falls auf einer Strukturebene für eine Variable kein Kommentar vorhanden ist, entfällt er in der Anzeige, was Sie an zwei aufeinander folgenden Punkten erkennen können.

Sie können die Variableninformationen entweder gesammelt unterhalb eines KOP-/FUP-Netzwerks anzeigen lassen oder direkt am jeweiligen Operanden. Wie beim Ein- und Ausblenden der Variableninformationen können Sie dies über die Einstellungen für alle Bausteine einstellen oder speziell für einen geöffneten Baustein ändern.

Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Variableninformationen" entweder die Option "Einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.

4. Legen Sie über die Klappliste "Position der Variableninformation (KOP/FUP)" fest, ob die Variableninformationen unterhalb eines KOP-/FUP-Netzwerks oder am Operanden angezeigt werden soll.
   Die Variableninformationen werden für alle geöffneten Bausteine eingeblendet und an der gewählten Position angezeigt. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls eingeblendet.
5. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Variableninformationen" die Option "Ausblenden".
   Die Variableninformationen werden für alle geöffneten Bausteine ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen für diese Bausteine ebenfalls ausgeblendet..

Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wenn Sie die Variableninformationen einblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." entweder die Option "Variableninformationen einblenden" oder die Option "Variableninformationen mit hierarchischen Kommentaren", abhängig davon, ob Sie einfache oder hierarchische Kommentare anzeigen möchten.
2. Um die Position der Variableninformationen festzulegen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Position der Variableninformation (KOP/FUP)". Mit jedem Klick auf die Schaltfläche ändert sich die Position. Alternativ können Sie auf den kleinen Pfeil neben der Schaltfläche "Position der Variableninformation (KOP/FUP)" klicken, um eine Klappliste zu öffnen. Wählen Sie dann die Position aus der Klappliste aus.
   Die Variableninformationen werden eingeblendet und an der gewählten Position angezeigt.
3. Wenn Sie die Variableninformationen ausblenden möchten, wählen Sie in der Klappliste "Blendet Variableninformationen ein." die Option "Variableninformationen ausblenden".
   Die Variableninformationen werden ausgeblendet.

Siehe auch

Interlocks programmieren (Seite 7782)

Supervisionsbedingungen einfügen (Seite 7783)

Transitionen programmieren (Seite 7789)

Permanente Anweisungen programmieren (Seite 7790)

17.8 Bausteinaufrufe in GRAPH einfügen

17.8.1 Bausteinaufrufe manuell einfügen

Sie können in Aktionen und in permanenten Anweisungen andere Bausteine aufrufen.

Bausteinaufrufe manuell in eine Aktion einfügen

Um einen Bausteinaufruf manuell in eine Aktion einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein in der Einzelschrittansicht.
2. Klicken Sie in der Spalte "Aktion" in die Zelle der Aktion.
3. Geben Sie "CALL <Name des aufzurufenden Bausteins>" ein und bestätigen Sie Ihre Eingabe mit der Eingabetaste.
4. Ersetzen Sie dabei "<Name des aufzurufenden Bausteins>" mit dem Namen des Bausteins, den Sie aufrufen möchten.
   Wenn der aufzurufende Baustein ein Funktionsbaustein ist, wird der Dialog "Aufrufoptionen" geöffnet. Führen Sie in diesem Fall die Schritte 5 und 6 aus.
5. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
6. Bestätigen Sie Ihre Eingabe mit "OK".

Bausteinaufruf manuell in eine permanente Anweisung einfügen

Um einen Bausteinaufruf manuell in eine permanente Anweisung einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie entweder die Ansicht für "Vorausgeschaltete permanente Anweisungen" oder für "Nachgeschaltete permanente Anweisungen".
2. Klicken Sie mit der rechten Maustaste auf die Stelle im Netzwerk, an der Sie den Bausteinaufruf einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Leerbox einfügen".
4. Klicken Sie auf "??".
   Ein Textfeld wird geöffnet.

5. Geben Sie den Namen des Bausteins ein, den Sie aufrufen möchten, oder öffnen Sie die Autovervollständigung und wählen den Baustein aus.
   Wenn der aufzurufende Baustein ein Funktionsbaustein ist, wird der Dialog "Aufrufoptionen" geöffnet. Führen Sie in diesem Fall die Schritte 6 und 7 aus.
6. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
7. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
8. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Grundlagen zu Instanzen (Seite 64)

Bausteinaufrufe (Seite 7701)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7801)

Bausteinaufrufe in GRAPH aktualisieren (Seite 7803)

17.8.2 Bausteinaufrufe per Drag & Drop einfügen

Sie können in Aktionen und in permanenten Anweisungen andere Bausteine aufrufen.

Bausteinaufrufe per Drag & Drop in eine Aktion einfügen

Um einen Bausteinaufruf per Drag & Drop in eine Aktion einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein in der Einzelschrittansicht.
2. Ziehen Sie den Baustein, den Sie aufrufen möchten, aus der Projektnavigation in der Spalte "Aktion" in die Zelle der Aktion.
   Wenn der aufzurufende Baustein ein Funktionsbaustein ist, wird der Dialog "Aufrufoptionen" geöffnet. Führen Sie in diesem Fall die Schritte 3 und 4 aus.

3. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Bestätigen Sie Ihre Eingabe mit "OK".

Bausteinaufrufe per Drag & Drop in eine permanente Anweisung einfügen

Um einen Bausteinaufruf per Drag & Drop in eine permanente Anweisung einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie entweder die Ansicht für "Vorausgeschaltete permanente Anweisungen" oder für "Nachgeschaltete permanente Anweisungen".
2. Ziehen Sie den Baustein, den Sie aufrufen möchten, aus der Projektnavigation an die Stelle des Netzwerks der permanenten Anweisung, an der Sie den Bausteinaufruf einfügen möchten.
   Wenn der aufzurufende Baustein ein Funktionsbaustein ist, wird der Dialog "Aufrufoptionen" geöffnet. Führen Sie in diesem Fall die Schritte 3 und 4 aus.
3. Geben Sie im Dialog an, ob Sie den Baustein als Einzel-, Multi- oder Parameterinstanz aufrufen wollen.
   - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
   - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
   - Wenn Sie auf die Schaltfläche "Parameter-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen des Durchgangsparameters (InOut) an, an dem die Instanz zur Laufzeit übergeben werden soll.
4. Wenn Sie einen Baustein aufrufen, der Überwachungen enthält, ordnen Sie den Überwachungen einen ProDiag-Funktionsbaustein zu.
5. Bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Grundlagen zu Instanzen (Seite 64)

Bausteinaufrufe (Seite 7701)

Bausteinaufrufe manuell einfügen (Seite 7800)

Bausteinaufrufe in GRAPH aktualisieren (Seite 7803)

17.8.3 Bausteinaufrufe in GRAPH aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben die folgenden Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren aller inkonsistenten Bausteinaufrufe im Programmiereditor.
  Die inkonsistenten Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt. Beachten Sie jedoch, dass die Parameter bei Funktionsbausteinen (FBs) nicht angezeigt werden und die Versorgung der Parameter über den entsprechenden Instanz-Datenbaustein (DB) erfolgt.
  - Gelöschte Parameter werden nicht entfernt.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Explizites Aktualisieren eines Bausteinaufrufs im Programmiereditor.
  Der inkonsistente Aufruf dieses Bausteins wird aktualisiert. Dabei werden die folgenden Aktionen durchgeführt:
  - Neue Parameter werden ergänzt. Beachten Sie jedoch, dass die Parameter bei Funktionsbausteinen (FBs) nicht angezeigt werden und die Versorgung der Parameter über den entsprechenden Instanz-Datenbaustein (DB) erfolgt.
  - Gelöschte Parameter werden nicht entfernt.
  - Umbenannte Parameter erhalten den neuen Parameternamen.
- Implizites Aktualisieren während des Übersetzens.
  Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert. Beachten Sie, dass Sie beim Aufruf von Funktionen (FCs) vor dem Übersetzungsvorgang alle neuen Formalparameter mit Aktualparametern versorgen müssen.

Alle inkonsistenten Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein alle Bausteinaufrufe zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Einen bestimmten Bausteinaufruf im Programmiereditor aktualisieren

Um einen bestimmten Bausteinaufruf im Programmiereditor zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf, den Sie aktualisieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Bausteinaufruf aktualisieren".
4. Falls für eine Funktion (FC) Parameter ergänzt wurden, geben Sie die Werte für die neuen Bausteinparameter an.

Hinweis

Beachten Sie, dass Ihnen der Befehl "Aktualisieren" nur zur Verfügung steht, wenn Sie nicht zuvor alle Bausteinaufrufe im Editor über den Befehl "Inkonsistente Bausteinaufrufe aktualisieren" aktualisiert haben.

Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Siehe auch

Bausteinaufrufe (Seite 7701)

Bausteinaufrufe manuell einfügen (Seite 7800)

Bausteinaufrufe per Drag & Drop einfügen (Seite 7801)

17.8.4 Bausteinaufruf ändern

Sie haben die Möglichkeit, für einen Bausteinaufruf den aufgerufenen Baustein zu ändern. Beachten Sie jedoch, dass dabei keine neuen Instanz-Datenbausteine erzeugt werden, z. B. beim Wechsel von einer Funktion (FC) zu einem Funktionsbaustein (FB).

Vorgehen

Um den aufgerufenen Baustein eines Bausteinaufrufs zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb des Bausteinaufrufs auf den Namen des aufgerufenen Bausteins und drücken Sie die Taste <F2>. Oder klicken Sie doppelt auf den Namen des aufgerufenen Bausteins.
   Ein Eingabefeld wird geöffnet, der Name des aktuell aufgerufenen Bausteins ist markiert.
2. Geben Sie den Namen des Bausteins ein, den Sie aufrufen möchten, oder wählen Sie einen Baustein in der Autovervollständigung aus.
3. Wenn Sie einen FB aufrufen möchten, erstellen Sie ggf. einen neuen Instanz-Datenbaustein und geben Sie ihn als Operand an.

17.9 Meldungen festlegen

17.9.1 Meldungsanzeige aktivieren und deaktivieren

Sie können die Meldungsanzeige global für neue GRAPH-Funktionsbausteine oder lokal für einzelne Bausteine aktivieren oder deaktivieren. Die lokale Einstellung eines Bausteins hat Vorrang vor der globalen Einstellung.

Meldungsanzeige global aktivieren oder deaktivieren

Um die Meldungsanzeige global für neue GRAPH-Funktionsbausteine zu aktivieren oder zu deaktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Wenn Sie die Meldungsanzeige aktivieren möchten, aktivieren Sie im Bereich "Voreinstellung für neue Bausteine > Meldungseigenschaften" das Optionskästchen "Meldungen aktivieren". Wenn Sie die Meldungsanzeige deaktivieren möchten, deaktivieren Sie das Optionskästchen "Meldungen aktivieren".

Meldungsanzeige lokal aktivieren oder deaktivieren

Um die Meldungsanzeige für einen vorhandenen GRAPH-Funktionsbaustein zu aktivieren oder zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Meldungsansicht für den GRAPH-Funktionsbaustein.
2. Wenn Sie die Meldungsanzeige aktivieren möchten, aktivieren Sie im Bereich "Meldungen" das Optionskästchen "Meldungen aktivieren". Wenn Sie die Meldungsanzeige deaktivieren möchten, deaktivieren Sie das Optionskästchen "Meldungen aktivieren".

Siehe auch

Interlock- und Supervisionsmeldungen (Seite 7706)

Meldungen bearbeiten (Seite 7806)

17.9.2 Meldungen bearbeiten

Die Meldung für Interlocks, Supervisionen und GRAPH-Warnungen (Schrittzeit-Überwachungen) legen Sie in den globalen Überwachungseinstellungen fest. Die Kategorie und die Unterkategorien für Interlocks, Supervisionen und GRAPH-Warnungen weisen Sie dagegen in der Meldungsansicht des GRAPH-Funktionsbaustein zu.

Zusätzlich können Sie innerhalb eines GRAPH-Schritts für Interlocks und Supervisionen schrittspezifische Meldungstexte anlegen.

Hinweis

Die Schrittzeit-Überwachung steht Ihnen nur für GRAPH-Funktionsbausteine der S7-1500 zur Verfügung.

Meldungen bearbeiten

Um die Meldungen für alle GRAPH-Funktionsbausteine zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Gemeinsame Daten > Überwachungseinstellungen".
   Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Meldetexte > GRAPH-Überwachungen".
3. Ziehen Sie per Drag & Drop die gewünschten Meldetextfelder aus dem Feld "Unterstützte Meldetextfelder" in den Textrahmen "Meldetext".
   Die Reihenfolge der einzelnen Textfelder können Sie jederzeit verändern.
4. Wenn Sie ein Textfeld wieder löschen möchten, dann klicken Sie auf das entsprechende Textfeld mit einem rechten Mausklick und wählen Sie aus dem Kontextmenü "Löschen" aus.
5. Wählen Sie ein gewünschtes Trennzeichen für die Textfelder aus.

oder:

1. Öffnen Sie die Meldungsansicht für den GRAPH-Funktionsbaustein.
2. Klicken Sie auf den grünen Pfeil "Gehe zu Überwachungseinstellungen" rechts neben dem Meldungstext.
3. Ab hier gilt die gleiche Vorgehensweise wie ab Schritt 3 oben beschrieben.

Kategorien und Unterkategorien auswählen

Um die Kategorien und Unterkategorien für Interlocks, Supervisionen und GRAPH-Warnungen auszuwählen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein.
2. Öffnen Sie die Meldungsansicht des GRAPH-Funktionsbausteins.
3. Wählen Sie aus den Klapplisten für die Kategorien oder die Unterkategorien jeweils den gewünschten Eintrag für Interlocks, Supervisionen und für die GRAPH-Warnungen aus.

Interlockmeldung eines einzelnen Schrittes bearbeiten

Um die Interlockmeldung eines einzelnen Schrittes zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Schritt in der Einzelschrittansicht.
2. Selektieren Sie den Interlock "-(C)-".
3. Öffnen Sie das Inspektorfenster und wechseln Sie ggf. in das Register "Eigenschaften".
4. Geben Sie in das Textfeld "Meldungstext" den gewünschten Meldetext ein.
   Dieser Text wird mit dem Meldetextfeld "<Step specific field>" an das Ende der GRAPH-Meldung angehängt.

Supervisionsmeldung eines einzelnen Schrittes bearbeiten

Um die Supervisionsmeldung eines einzelnen Schrittes zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Schritt in der Einzelschrittansicht.
2. Selektieren Sie die Supervision "-(v)-".
3. Öffnen Sie das Inspektorfenster und wechseln Sie ggf. in das Register "Eigenschaften".
4. Geben Sie in das Textfeld "Meldungstext" den gewünschten Meldetext ein.
   Dieser Text wird mit dem Meldetextfeld "<Step specific field>" an das Ende der GRAPH-Meldung angehängt. Bei einer S7-1500 wird der Text auch als schrittspezifischer Text für die Schrittzeit-Überwachung verwendet, falls diese als Supervisionsbedingung genutzt wird.

Siehe auch

Interlock- und Supervisionsmeldungen (Seite 7706)

Meldungsanzeige aktivieren und deaktivieren (Seite 7806)

Aufbau des Meldetexteditors (Seite 8233)

Meldetexte anlegen (Seite 8240)

17.10 Erstwerte erfassen

17.10.1 Übersicht über die Erstwerterfassung

Voraussetzung

Damit Sie die Erstwerterfassung aktivieren können, müssen Sie im GRAPH-Funktionsbaustein eine Version >= 4.0 einstellen.

Weitere Informationen finden Sie hier: Erstwerterfassung aktivieren (Seite 7810)

Beschreibung

Sie haben die Möglichkeit, den Ablauf des Anwenderprogramms auf einem HMI-Gerät zu verfolgen. Dabei werden die Daten und Werte auf dem Bediengerät stetig mit der CPU abgeglichen und aktualisiert. Für jeden GRAPH-Funktionsbaustein sehen Sie immer den aktuellen Programmstatus auf dem Bediengerät. Die Aktualwerte der Signalzustände von booleschen Operanden und die Ergebnisse von Vergleichern in Interlocks und Transitionen werden ab dem Zeitpunkt der Aktivierung der Erstwerterfassung kontinuierlich aufgezeichnet, auch wenn kein Fehler auftritt.

Die Signalzustände werden pro Zyklus im GRAPH-Instanzdatenbaustein gespeichert. Es können max. 32 Signalzustände pro Interlock bzw. pro Transition aufgezeichnet werden. Die Aktualwerte werden in den statischen Parametern CRIT_LOC (Schritt) bzw. CRIT (Transition) in einem DWORD gespeichert und jeder einzelne Signalzustand belegt ein Bit. Sobald ein Interlock-Fehler auftritt, werden die Werte des Parameters CRIT_LOC in den Parameter CRIT_LOC_ERR kopiert. Sobald ein Supervision-Fehler auftritt, werden die Werte des Parameters CRIT in der nachfolgenden Transition in den Parameter CRIT_FLT kopiert. Diese Erstwerte werden bis zum Auftreten eines neuen Fehlers im aktiven Schritt nicht mehr verändert.

Mithilfe der PLC-Code-Anzeige können Sie sich die Aktualwerte anzeigen lassen. Sobald ein Fehler auftritt, können Sie sich zusätzlich auch die Erstwerte anzeigen lassen und die fehlerhaften Operanden sofort erkennen (Kriterienanalyse). Sie können zwischen der Anzeige der Aktualwerte und der Anzeige der Erstwerte hin und her schalten.

Weitere Informationen zum Visualisieren auf dem HMI-Gerät finden Sie hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen"

Damit Ihnen die genannten statischen Parameter der Schritt- bzw. Transitionsstrukturen angezeigt werden, müssen Sie den maximalen Schnittstellenparametersatz für die Bausteinschnittstelle auswählen.

Weitere Informationen zum Schnittstellenparametersatz finden Sie hier: Schnittstellenparametersatz festlegen (Seite 7734)

Weitere Informationen zu den statischen Parametern eines GRAPH-Funktionsbausteins finden Sie hier: Statische Parameter (Seite 7718)

Unterstützte Anweisungen

Für die Erstwerterfassung werden alle Anweisunge unterstützt, die bei einem GRAPH-Funktionsbaustein für Transitionen und Interlocks unterstützt werden.

Bei den Bitverknüpfungen wird der Zustand des Operanden und bei den Vergleichern das Ergebnis des Vergleichs gespeichert.

Aufzeichnung der Aktualwerte bzw. Erstwerte

Pro Programmzyklus wird jeweils das Interlock der aktiven Schritte und die darauffolgenden Transitionen aufgezeichnet. Wenn Sie die Bausteineigenschaft "Permanente Bearbeitung aller Interlocks im Handbetrieb" oder die Testeinstellungen "Alle Interlocks bearbeiten" bzw. "Alle Transititionen bearbeiten" aktivieren, dann werden pro Programmzyklus alle Interlocks und alle Transitionen der GRAPH-Ablaufkette aufgezeichnet. Das kann möglicherweise zu einer verlängerten Zykluszeit führen.

Auswertungsreihenfolge auf dem HMI-Gerät in der PLC-Code-Anzeige

Die Aufzeichnung der Signalzustände im jeweiligen DWORD bzw. die Auflistung der Operanden in der Symboltabelle der PLC-Code-Anzeige findet in einer fest definierten Reihenfolge statt. Dabei wird z. B. in einem KOP-Interlock immer an der Stromschiene links oben begonnen und endet rechts unten.

Hinweis

Auswertungsreihenfolge bei der Programmierung beachten

Auch bei der Kriterienanalyse auf dem HMI-Gerät wird diese feste Reihenfolge bei der Auflistung der fehlerhaften Operanden in der Symboltabelle eingehalten. Beachten Sie diese Auswertungsreihenfolge bei der Programmierung des Interlocks bzw. der Transition.

Weitere Informationen zur Reihenfolge der Auflistung finden Sie hier: Beispiel für die Reihenfolge der Auflistung (Seite 7812)

17.10.2 Erstwerterfassung aktivieren

Vorgehen

Um die Erstwerterfassung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den GRAPH-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des GRAPH-Funktionsbausteins wird geöffnet.
2. Wählen Sie im Register "Allgemein > Baustein" eine Version >= V4.0 aus.
3. Klicken Sie das Register "Attribute" an.
4. Aktivieren Sie die "Erstwerterfassung".

5. Klicken Sie auf "OK".
6. Übersetzen und laden Sie das Anwenderprogramm.

Ergebnis

Sobald das Anwenderprogramm übersetzt wurde, beginnt die Aufzeichnung der Erstwerte.

17.10.3 Erstwerte zurücksetzen

Beschreibung

Jeder GRAPH-Funktionsbaustein enthält in seinem Baustein-Parametersatz für jeden einzelnen Schritt die beiden systeminternen statischen Parameter "H_IL_ERR" und "H_SV_FLT". Wenn Sie die Option "Erstwerterfassung" in den Bausteineigenschaften des GRAPH-Funktionsbausteins aktiviert haben, dann werden bei einem Verriegelungsfehler beim Parameter "H_IL_ERR" (Interlock) das Bit 2 und bei einem Überwachungsfehler beim Parameter "H_SV_FLT" (Transition) das Bit 5 jeweils auf TRUE gesetzt, sobald ein entsprechendes Ereignis im aktiven Schritt vorliegt. Dadurch haben Sie die Möglichkeit, die Erstwerte auf Ihrem HMI-Gerät mit der PLC-Code-Anzeige zu visualisieren.

Ab Bausteinversion >= 5.0 eines GRAPH-Funktionsbausteins haben Sie die Möglichkeit, mithilfe des Eingangsparameters "RESET_CRIT" den Signalzustand der beiden statischen Parameter "H_IL_ERR" und "H_SV_FLT" nach der Kriterienanalyse wieder auf FALSE zurück zu setzen. Diese Möglichkeit sollten Sie vor allem dann nutzen, wenn Sie die Programmierung von Interlocks und Transitionen verändern und die dadurch veralteten Erstwerte nicht mehr benötigen. Die Signalzustände werden durch die Umprogrammierung nicht automatisch wieder auf FALSE zurückgesetzt.

Vorgehen

Um die Signalzustände der beiden systeminternen statischen Parameter zurück zu setzen, gehen Sie folgendermaßen vor:

1. Überprüfen Sie, dass der GRAPH-Funktionsbaustein eine Version >= 5.0 hat.
2. Maximalen Bausteinparametersatz auswählen:
   - Wenn Sie einen neuen GRAPH-Funktionsbaustein anlegen wollen, dann wählen Sie zuerst in den allgemeinen Einstellungen für GRAPH den maximalen Bausteinparametersatz aus.
   - Wenn Sie bei einem bereits angelegten GRAPH-Funktionsbaustein den Parametersatz verändern möchten, dann öffnen Sie den Baustein, klicken Sie in die Bausteinschnittstelle und wählen Sie in der Menüleiste "Bearbeiten > Schnittstellenparameter > Maximale Schnittstellenparameter" aus. Oder legen Sie den Parameter RESET_CRIT manuell im Abschnitt IN an.
3. Verschalten Sie beim Aufruf des GRAPH-Funktionsbausteins den Eingangsparameter "RESET_CRIT" in Ihrem Anwenderprogramm.

Ergebnis

Sobald am Eingangsparameter "RESET_CRIT" eine positive Signalflanke ansteht, werden die Signalzustände der beiden systeminternen statischen Parameter "H_IL_ERR" und "H_SV_FLT" in allen programmierten GRAPH-Schritten auf FALSE zurückgesetzt und die PLC-Code-Anzeige auf Ihrem HMI-Gerät zeigt kein veraltetes Ereignis mehr an.

Siehe auch

Grundlagen zur Bausteinschnittstelle eines GRAPH-Funktionsbausteins (Seite 7707)

Übersicht über die Erstwerterfassung (Seite 7809)

Eingangsparameter (Seite 7710)

Statische Parameter eines Schritts (Seite 7718)

17.10.4 Beispiel für die Reihenfolge der Auflistung

Die Aufzeichnung der Aktual- bzw. Erstwerte und die Auflistung der Operanden in der Symboltabelle der PLC-Code-Anzeige unterliegen der gleichen fest definierten Reihenfolge.

Anhand der folgenden Beispiele sehen Sie, wie die Aufzeichnung der Signalzustände innerhalb eines KOP-Interlocks und die Auflistung der Operanden auf dem HMI-Gerät funktionieren.

Aufzeichnung aller Signalzustände

Die Signalzustände werden in der folgenden Reihenfolge in den statischen Parameter CRIT_LOC, bzw. in den Ausgangsparameter T_CRIT geschrieben:

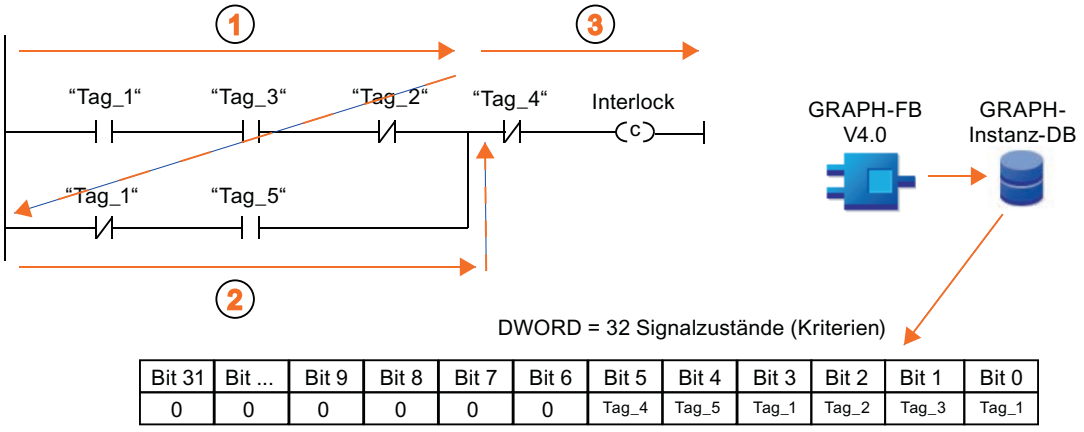

Bit 31	Bit ...	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	Tag_4	Tag_5	Tag_1	Tag_2	Tag_3	Tag_1

Auf dem HMI-Gerät werden alle Operanden genau nach dieser Reihenfolge in der Symboltabelle der PLC-Code-Anzeige aufgelistet:

Symbolname	Operand	Kommentar
Tag_1	z. B. %M2xy	Automatikbetrieb / Handbetrieb
Tag_3		Motor einschalten
Tag_2		Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster
Tag_4		Lichtschranke

Auflistung der Operanden im Fehlerfall

Wenn nun im Interlock Fehler auftreten, dann werden die fehlerhaften Operanden farblich markiert und in der Symboltabelle werden nur noch die fehlerhaften Operanden aufgelistet:

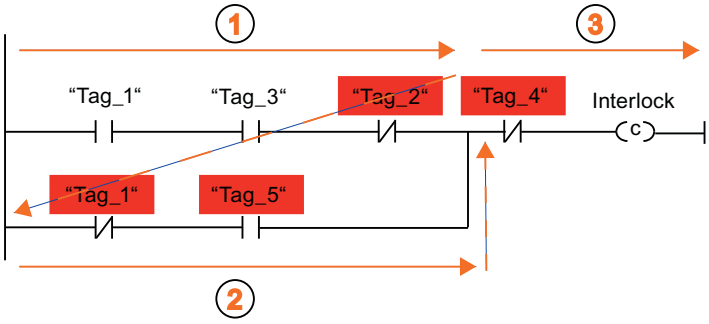

Auflistung der fehlerhaften Operanden in der Symboltabelle der PLC-Code-Anzeige:

Symbolname	Operand	Kommentar
Tag_2	z. B. %M2xy	Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster
Tag_4		Lichtschranke

Optimierte Programmierung des Interlocks im Fehlerfall

Damit Sie schnell auf den ersten Blick erkennen können, dass die Lichtschranke für den Fehler verantwortlich ist, ist es in diesem Beispiel von Vorteil, den Operanden "Tag_4: Lichtschranke" an erster Stelle zu programmieren:

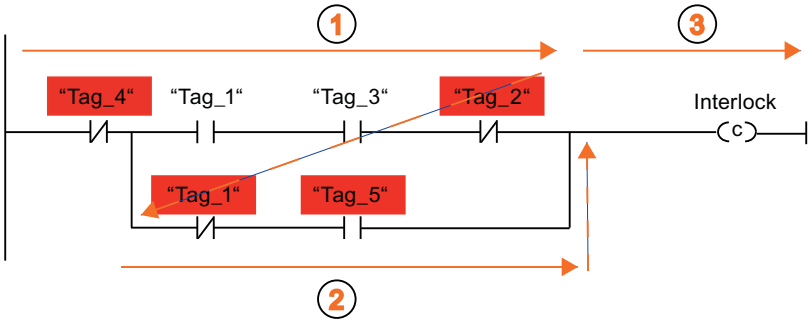

Damit wird die Information zur offenen Lichtschranke in der ersten Zeile der Symboltabelle dargestellt:

Symbolname	Operand	Kommentar
Tag_4	z. B. %M2xy	Lichtschranke
Tag_2		Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster

Meldungen projektieren 18

18.1 Einführung in die Meldungsprojektierung

Übersicht

Meldungen ermöglichen Ihnen, Fehler bei der Prozessbearbeitung in den Automatisierungssystemen schnell zu erkennen, genau zu lokalisieren und zu beheben. Stillstandszeiten einer Anlage können so wesentlich verkürzt werden.

Bevor Meldungen ausgegeben werden können, müssen sie projektiert werden.

Sie können ereignisabhängige Meldungen mit zugeordneten Meldetexten und Meldeattributen anlegen, bearbeiten, übersetzen und auf Anzeigegeräten ausgeben.

In der nachfolgenden Tabelle sind die Meldungsarten mit einer kurzen Beschreibung ihrer Funktion aufgelistet.

Meldungsart	Beschreibung
Programmmeldungen	Programmmeldungen dienen zum Melden von programmsynchronen Ereignissen und sind jeweils einem Baustein zugeordnet. Sie werden im Programmiereditor erstellt und im Meldungseditor bearbeitet.
Systemdiagnosemeldungen	Systemdiagnosemeldungen sind konfigurationsabhängige Baugruppenereignisse und werden in der Hardwareprojektierung aktiviert bzw. deaktiviert. Sie können im Meldungseditor nur angesehen, aber nicht bearbeitet werden.
Anwenderdiagnosemeldungen	Mit Anwenderdiagnosemeldungen können Sie einen Anwendereintrag in den Diagnosepuffer schreiben und eine entsprechende Meldung senden. Sie sind einer CPU zugeordnet. Sie werden im Meldungseditor erstellt und können hier bearbeitet werden.

18.2 Vergabe der Meldungsnummern

Nummernvergabe

Die Meldungen sind durch eine CPU-weit eindeutige Nummer identifiziert. Dadurch entfällt das Neuübersetzen nach dem Kopieren von kompletten Programmen. Eine Ausnahme bildet das Kopieren einzelner Bausteine. Hier ist eine Neuübersetzung notwendig, um die geänderte Meldungsnummer (ID) in das Programm einzubinden.

18.3 Bestandteile einer Meldung

Übersicht

Wie eine Meldung angezeigt wird, ist abhängig vom Meldeverfahren, vom verwendeten Meldebaustein und vom Anzeigegerät.

Mögliche Bestandteile sind in der folgenden Tabelle aufgelistet:

Bestandteil	Beschreibung
Zeitstempel	Wird beim Auftreten des Meldeereignisses im Automatisierungssystem erzeugt.
Meldungszustand	Möglich sind: gekommen, gekomen mit Quittierung, gegangen, gegangen ohne Quittierung, gegangen mit Quittierung.
Begleitwert	Einigen Meldungen kann ein Begleitwert beigefügt werden, der von der verwendeten Meldeanweisung ausgewertet werden kann.
Abbild	Bei einem Systemabsturz können die angefallenen Meldungen im Nachhinein auf HMI-Geräten angezeigt werden.
Meldungsnummer	CPU-weit eindeutige Nummer (ID), die vom System vergeben wird und eine Meldung identifiziert.
Meldetexte	Werden vom Anwender projektiert.

18.4 Verfügbare Meldebausteine

S7-300/400: Übersicht über die Meldebausteine

Sie haben die Auswahl zwischen den folgenden Meldebausteinen:

- ALARM
- ALARM_8
- ALARM_8P
- NOTIFY
- ALARM_S
- ALARM_SQ
- AR_SEND (zum Senden von Archiven; keine Projektierung von Meldetexten und Meldeattributen möglich)
- NOTIFY_8P
- ALARM_DQ
- ALARM_D
- ALARM_T

S7-300/400: Wann setzen Sie welchen Meldebaustein ein?

Die folgende Tabelle hilft Ihnen bei der Entscheidung, welchen Meldebaustein Sie für Ihren Anwendungsfall auswählen. Die Auswahl des Meldebausteins ist abhängig von den folgenden Faktoren:

- Der Anzahl der im Baustein zur Verfügung stehenden Kanäle und damit die Anzahl der Signale, die je Bausteinaufruf überwacht werden.
- Der Möglichkeit, Meldungen zu quittieren.
- Der Möglichkeit, Begleitwerte mitzugeben.
- Den Anzeigegeräten, die eingesetzt werden sollen.
- Dem Mengengerüst der CPU.

Melde-baustein	Kanäle	Quittie-rung	Begleit-werte	Besonderheiten
ALARM *)	1	ja	bis zu 10	Sendet bei kommender oder gehender Flanke je eine Meldung
ALARM_8 *)	8	ja	-	Sendet bei kommender oder gehender Flanke eines oder mehrerer Signale eine Meldung
ALARM_8P *)	8	ja	bis zu 10	wie ALARM_8
NOTIFY	1	nein	bis zu 10	wie ALARM

Melde-baustein	Kanäle	Quittie-rung	Begleit-werte	Besonderheiten
NOTIFY_8P	8	nein	bis zu 10	wie ALARM
AR_SEND	1	-	-	Dient zum Senden eines Archivs; keine Projektierung von Meldetexten und Meldeattributen möglich
ALARM_SQ	1	ja	1	Mit jedem Bausteinaufruf und einer Signaländerung zum vorhergehenden Bausteinaufruf, wird eine Meldung erzeugt
ALARM_S	1	nein	1	wie ALARM_SQ
ALARM_DQ	1	ja	1	wie ALARM_SQ
ALARM_D	1	nein	1	wie ALARM_SQ
ALARM_T	-	-	-	Für Systemtestzwecke reserviert
*) Wenn das quittierungsgetriggerte Melden aktiviert ist, wird die Meldung nicht in der Meldungsanzeige angezeigt.				

S7-1500: Übersicht über die Meldebausteine

Der folgende Meldebaustein steht zur Verfügung:

- Program_Alarm

Die Auswahl der entsprechenden Meldeklasse entscheidet, ob der Baustein quittierpflichtig oder nicht quittierpflichtig ist. Standardmäßig ist eingestellt, dass der Baustein nicht quittierpflichtig ist.

Melde-baustein	Kanäle	Quittie-rung	Begleit-werte	Besonderheiten
Program_Alarm	1	möglich	bis zu 10	Sendet bei kommender oder gehender Flanke je eine Meldung mit Zeitstempel

18.5 Meldungstyp und Meldungen

Mit Hilfe der Programmiereditoren können Sie in verschiedenen Arbeitsgängen zunächst einen Meldungstyp (d.h. einen FB als Vorlage für Instanz-DBs) und später Meldungen (d.h. Instanz-DBs aus der Vorlage des Meldungstyps) erstellen.

Der meldefähige Baustein kann ein FB oder ein Instanz-DB sein

- Bei einem FB erstellen Sie einen Meldungstyp, der als Vorlage für Meldungen dient. Alle Eingaben, die Sie für den Meldungstyp machen, werden automatisch für daraus abgeleitete Meldungen übernommen. Wenn Sie dem FB einen Instanz-DB zuordnen, werden nach der Vorlage des Meldungstyps automatisch Meldungen zum Instanz-DB generiert und Meldungsnummern zugeordnet.
- Bei einem Instanz-DB können Sie die Meldungen, die aus dem Meldungstyp generiert wurden, instanzspezifisch modifizieren, sofern sie nicht verriegelt sind.

Der sichtbare Unterschied ist, dass bei den Meldungen Meldungsnummern zugeordnet sind, bei einem Meldungstyp dagegen nicht. Die Meldungstypen und die zugehörigen -Instanzen werden im Meldungseditor untereinander angeordnet. Standardmäßig ist die Tabelle, die die Meldungsinstanzen enthält, zugeklappt.

Ändern der Daten beim Meldungstyp

Wenn Sie nachträglich Daten beim Meldungstyp ändern, so werden diese Änderungen automatisch an den Instanzen übernommen.
Ausnahmen: Sie haben diese Daten zuvor an der Instanz geändert bzw. nachträglich am Meldungstyp Daten verriegelt oder entriegelt.

Hinweis

Wenn Sie Instanzen in ein anderes Programm kopieren, ohne dass Sie den Meldungstyp mitkopieren, wird die Instanz nicht vollständig angezeigt. Kopieren Sie in diesem Fall den Meldungstyp mit in das neue Programm.

Zurücksetzen von instanzspezifischen Daten auf den Wert des Meldungstyps

Wurden Attribute oder Texte an der Meldungsinstanz überschrieben, wird im Meldungseditor ein Typsymbol neben dem Attribut angezeigt. Für jedes Attribut kann bestimmt werden, ob wieder der Wert des Meldungstyps verwendet werden soll. In diesem Fall wird kein Typsymbol angezeigt.

Siehe auch

Meldungstyp bearbeiten (Seite 7828)

Instanz-DB anlegen und Meldungsinstanzen bearbeiten (Seite 7828)

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

Meldeklassen anlegen (Seite 7853)
Meldeklassen bearbeiten (Seite 7854)
Arbeiten mit dem Meldungseditor (Seite 7825)

18.6 Formalparameter, Meldungsdatentypen und Meldebausteine

Formalparameter als Meldenummerneingang (S7-300/400)

Für jede Programmmeldung oder Gruppe von Programmmeldungen benötigen Sie in Ihrem Programm einen Formalparameter (Name der Meldung), den Sie als STATIC-Parameter in der Variablenübersicht Ihres Programms angeben. Der Formalparameter wird als Meldenummerneingang benutzt und bildet die Basis für eine Meldung.

Versorgung der Formalparameter mit dem passenden Datentyp (S7-300/400)

Dem Formalparameter muss entsprechend dem verwendeten Meldebausteins ein Meldungsdatentyp zugewiesen werden.

Meldungsdatentypen und zugehörige Meldebausteine

Die folgende Tabelle zeigt die Meldungsdatentypen mit den zugehörigen Meldebausteinen und deren Eigenschaften. Die Werte der Datentypen haben die gleichen Bezeichnungen wie die Meldebausteine (Ausnahme: siehe Tabelle) und sind mit dem Präfix "C_" versehen.

SIMATIC-System	Datentyp	Meldebaustein	Eigenschaften
S7-400	C_Alarm_8	ALARM_8	8 Kanäle, mit Quittierung, keine Begleitwerte
S7-400	C_Alarm_8p	ALARM_8P	8 Kanäle, mit Quittierung, bis zu 10 Begleitwerte pro Kanal
S7-400	C_Notify	NOTIFY	1 Kanal, keine Quittierung, bis zu 10 Begleitwerte
S7-400	C_Alarm	ALARM	1 Kanal, mit Quittierung, bis zu 10 Begleitwerte
S7-300/400	C_Alarm_s	ALARM_S	1 Kanal, keine Quittierung, bis zu 1 Begleitwert
S7-300/400	C_Alarm_s	ALARM_SQ	1 Kanal, mit Quittierung, bis zu 1 Begleitwert
S7-400	C_Ar_Send	AR_SEND	dient zum Senden eines Archivs
S7-400	C_Notify_8p	NOTIFY_8P	8 Kanäle, keine Quittierung, bis zu 10 Begleitwerte
S7-300/400	C_Alarm_s	ALARM_DQ	1 Kanal, mit Quittierung, bis zu 1 Begleitwert
S7-300/400	C_Alarm_s	ALARM_D	1 Kanal, keine Quittierung möglich, bis zu 1 Begleitwert
S7-1500	Program_Alarm	Program_Alarm	1 Kanal, Quitierung durch Parametrierung möglich, bis zu 10 Begleitwerte

18.7 Aufbau des Meldungseditors

Aufbau des Meldungseditors

Das folgende Bild zeigt exemplarisch die Komponenten des Meldungseditors:

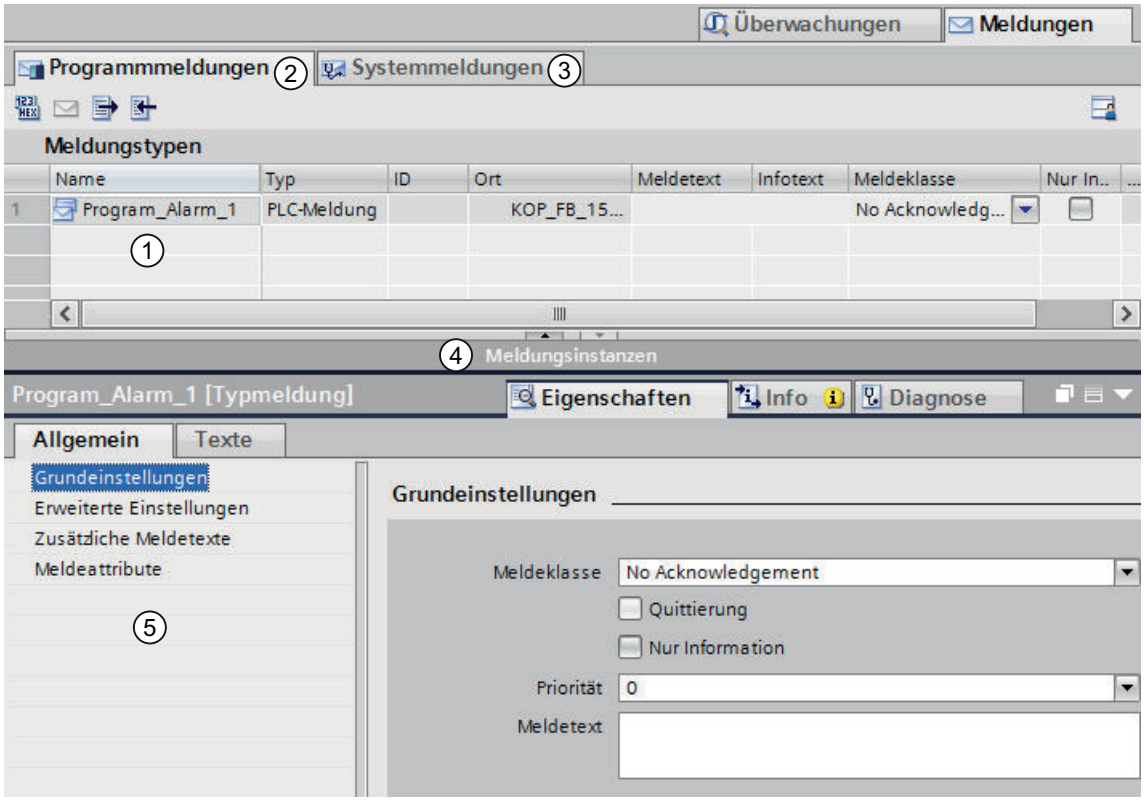

1		Tabellarische Darstellung der Meldungen im Arbeitsbereich, siehe: Arbeiten mit dem Meldungseditor (Seite 7825)
2		Register "Programmmeldungen": Programmmeldungen können hier bearbeitet werden. Siehe Grundlagen zum Arbeitsbereich
3		Register "Systemmeldungen": Systemmeldungen können nur angesehen, aber nicht bearbeitet werden. Die Konfiguration dieser Meldungen findet in den Eigenschaften des Geräts (Kontextmenübefehl "Gehe zu Gerät") sowie in der Projektnavigation unter "Gemeinsame Daten > Systemdiagnoseeinstellungen" statt.
4		Die Tabelle "Meldungsinstanzen" enthält die vom Meldungstyp (FB) abgeleiteten Instanzen (Instanz-DBs). Standardmäßig ist die Tabelle zugklappt.
5		Inspektorfenster
		Register "Anwenderdiagnosemeldungen": Anwenderdiagnosemeldungen können hier angelegt und bearbeitet werden. Bei S7-1500 CPUs ist das Register nicht verfügbar.
		Anordnung merken: Die aktuelle Anordnung wird beim erneuten Öffnen der PLC-Meldungen wiederhergestellt.

Sie können die notwendigen Parameter, Texte und Attribute sowohl in der Tabelle als auch im Inspektorfenster eingeben bzw. ändern. Sie können die Spalten der Tabelle auf- oder absteigend sortieren.

18.8 Arbeiten mit dem Meldungseditor

Nachfolgend sind einige grundsätzliche Vorgehensweise zum Arbeiten mit dem Meldungseditor aufgeführt.

Meldungsinstanzen anzeigen/verbergen

Beim ersten Öffnen des Meldungseditors ist die Tabelle, die die Meldungsinstanzen enthält, zugeklappt. Die Teilung zwischen den beiden Tabellen können Sie mit der Maus verändern. Klicken Sie dazu mit der Maus zwischen die beiden Bereiche und verändern Sie bei gedrückter Maustaste durch einer Verschieben des Teilers nach links oder rechts die Aufteilung. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

Tabellen auf- oder absteigend sortieren

Um eine Tabelle nach einer Spalte auf- oder absteigend zu sortieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Tabellenkopf einer Spalte, wenn Sie diese aufsteigend sortieren möchten.
2. Klicken Sie erneut auf den Tabellenkopf derselben Spalte, um die Spalte absteigend zu sortieren.
3. Klicken Sie ein drittes Mal auf den Tabellenkopf derselben Spalte, um die Sortierung wieder aufzuheben.

Zellen kopieren/einfügen

Sie können eine oder mehrere Zellen selektieren und an anderer Stelle wieder einfügen. Die Daten werden überprüft, und wenn diese gültig sind, ist der Kontextmenübefehl "Kopieren" aktiviert. An der einzufügenden Stelle werden die Daten erneut geprüft, und wenn diese gültig sind, ist der Kontextmenübefehl "Einfügen" aktiviert. Wenn die Daten ungültig sind, sind die Kontextmenübefehle gegraut.

18.9 Meldungen anlegen und bearbeiten

18.9.1 Programmmeldungen anlegen

Voraussetzung

Sie haben einen Funktionsbaustein angelegt.

Vorgehen

Um eine Programmmeldung anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Ordner "Programmbausteine" in der Projektnavigation den Funktionsbaustein (FB) aus, für den Sie eine Programmmeldung anlegen wollen und öffnen Sie diesen Baustein durch Doppelklick.
2. Klicken Sie in der Task Card "Anweisungen", auf "Erweiterte Anweisungen" > "Meldungen" und ziehen Sie den gewünschten Meldebaustein (z. B. "Alarm_S") in das Anweisungsfenster bzw. Netzwerk (je nach Bausteinsprache).
   Ergebnis: Im Anweisungsteil des FBs werden die Eingänge des aufgerufenen Meldebausteins, hier also des ALARM_S-Bausteins, angezeigt.
3. Füllen Sie die Bausteinschnittstelle aus. Für jeden Meldebaustein, der im FB aufgerufen wird, müssen Sie im aufrufenden FB Variablen deklarieren. Tragen Sie dazu beispielsweise folgende Variablen ein:
   - beim Parameter "STATIC" einen Namen für den Meldebausteineingang, z. B. "Meld01" und den Datentyp (C_Alarm_s).
   - Weisen Sie der Variable "EV_ID" den von Ihnen vergebenen Namen für den Meldebausteineingang, hier also "Meld01", und ID_16#eeee zu.

   Hinweis

   Wenn Sie statt eines in der CPU vorhandenen Meldebausteins einen FB mit Multiinstanzen aufrufen, in dem auch Meldungen projektiert sind, müssen Sie im aufrufenden Baustein ebenfalls die Meldungen des FBs mit Multiinstanzen projektieren.
4. Wiederholen Sie die Schritte 2. bis 3. für alle Aufrufe von Meldebausteinen in diesem FB.

18.9.2 Programmmeldungen im Meldungseditor bearbeiten

Voraussetzung

Sie haben eine Programmmeldung angelegt.

Vorgehen

Um Programmmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen". Der Meldungseditor wird geöffnet.
2. Geben Sie die gewünschten Texte und Attribute in die entsprechenden Spalten ein.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.9.3 Programmmeldungen im Programmiereditor bearbeiten

Voraussetzung

Sie haben eine Programmmeldung angelegt.

Vorgehen

Um Programmmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Markieren Sie die entsprechende Zeile in der Bausteinschnittstelle.
2. Wechseln Sie in das Register "Meldung" im Inspektorfenster.
3. Geben Sie die gewünschten Texte und Attribute in die entsprechenden Felder ein.

18.9.4 Programmmeldungen löschen

Vorgehen

Um eine Programmmeldung zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, der die zu löschende Meldung enthält.
2. Entfernen Sie im Anweisungsfenster bzw. Netzwerk (je nach Bausteinsprache) den beim Anlegen eingefügten Meldebaustein.
3. Markieren Sie in der Bausteinschnittstelle die entsprechende Zeile und wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die Meldung wird glöscht.

18.9.5 Meldungstyp bearbeiten

Vorgehen

Um einen Meldungstyp zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gewünschten Meldebaustein.
2. Tragen Sie die gewünschten Texte ein bzw. wählen Sie die gewünschten Attribute aus. Falls Sie einen mehrkanaligen Meldebaustein (z. B. "ALARM_8") selektiert haben, können Sie jeder Submeldung eigene Texte und zum Teil eigene Attribute zuweisen.

   Hinweis

   Um Submeldungen zu bearbeiten, müssen Sie sich im Meldungseditor befinden.

3. Wenn die Texte bzw. Attribute an der Instanz nicht änderbar sein sollen, verriegeln Sie diese im Meldungstyp.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.9.6 Instanz-DB anlegen und Meldungsinstanzen bearbeiten

Voraussetzung

Sie müssen zuvor einen FB angelegt und in diesem mindestens eine Meldung angelegt haben.

Vorgehen

Um einem Meldungstyp Instanz-Datenbausteine (DBs) zuzuweisen und die Meldungen für diese DBs instanzspezifisch zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Neuen Baustein hinzufügen", klicken Sie im aufgeblendeten Dialog auf die Schaltfläche "Datenbaustein (DB)" und wählen Sie aus der Klappliste bei "Typ" den Funktionsbaustein (Meldungstyp) aus, dem Sie den Instanz-Datenbaustein zuordnen möchten.
2. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen", um die Meldungsprojektierung zu öffnen.
3. Tragen Sie die gewünschten Änderungen für die jeweilige Meldungsinstanz ein.

Ergebnis

Damit ist die Meldungsprojektierung für den angelegten Instanz-DB abgeschlossen.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.9.7 Anwenderdiagnosemeldungen anlegen

Anwenderdiagnosemeldungen sind einer CPU zugeordnet. Sie werden im Meldungseditor erstellt und bearbeitet.

Voraussetzung

Sie haben einen Funktionsbaustein angelegt.

Vorgehen

Um eine Anwenderdiagnosemeldung anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Ordner "Programmbausteine" in der Projektnavigation den Funktionsbaustein (FB) aus, für den Sie eine Anwenderdiagnosemeldung anlegen wollen und öffnen Sie diesen Baustein durch Doppelklick.
2. Fügen Sie im Anweisungsfenster des FBs eine Projektierung der gehenden und der kommenden Anwenderdiagnosemeldung ein, indem Sie den Meldebaustein "WR_USMSG" zweimal (jeweils im Zusammenhang mit einer negativen und einer positiven Flanke) aufrufen.
3. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen", um den Meldungseditor zu öffnen.

4. Wählen Sie im Meldungseditor das Register "Anwenderdiagnosemeldungen".
5. Klicken Sie in die Tabelle und wählen Sie im Kontextmenü den Befehl "Neue Meldung einfügen".

Ergebnis

Sie haben eine Anwenderdiagnosemeldung angelegt.

18.9.8 Anwenderdiagnosemeldungen bearbeiten

Voraussetzung

Sie haben eine Anwenderdiagnosemeldung angelegt.

Der Meldungseditor ist geöffnet.

Vorgehen

Um eine Anwenderdiagnosemeldung zu bearbeiten, gehen Sie folgendermaßen vor:

1. Geben Sie die gewünschten Texte und Attribute in die entsprechenden Spalten ein.

18.9.9 Anwenderdiagnosemeldungen löschen

Sie können eine markierte Meldung löschen. Die Texte, die Sie zu dieser Meldung projektiert haben, werden gelöscht, und die Meldenummer wird freigegeben.

Vorgehen

Um eine Anwenderdiagnosemeldung zu löschen, gehen Sie folgendermaßen vor:

1. Entfernen Sie im Anweisungsfenster des FBs die beim Anlegen eingefügten Meldebausteine.
2. Markieren Sie im Meldungseditor die entsprechende Zeile in der Tabelle und wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die Meldung wurde gelöscht. In der Tabelle wird sie nicht mehr angezeigt.

18.9.10 Begleitwerte in Meldungen einfügen

Um Meldungen aktuelle Informationen, z. B. aus dem Prozess, mitzugeben, können Sie Begleitwerte an beliebigen Stellen eines Meldungstextes einfügen.

Vorgehen

Um einen Begleitwert in eine Meldung einzufügen, gehen Sie folgendermaßen vor:

Alarm-SFB der S7-400 (z.B. Alarm, Alarm_8P)

Es gibt 10 Begleitwerte, deren jeweilige maximale Länge 12 Bytes beträgt. Sie sind nicht intern adressierbar.

1. Stellen Sie einen Block mit folgendem Aufbau zusammen:
   @<Nr. des Begleitwertes><Elementtyp>%<Formatangabe>@.
2. Fügen Sie diesen Block an den Stellen im Meldungstext ein, an denen der Begleitwert angezeigt werden soll.

Alarm-SFC der S7-300/400 (z.B. Alarm_S)

Es gibt einen Begleitwert, dessen maximale Länge 12 Bytes beträgt. Er ist intern mit Hilfe des Index und des Elementtyps adressierbar.

1. Stellen Sie einen Block mit folgendem Aufbau zusammen:
   @<Index><Elementtyp>%<Formatangabe>@.
   Index: Index im Begleitwert. Der Index wird interpretiert als Array Index (mit 1 startend).
   Der Byte-Offset wird berechnet mit fogender Formel:
   Offset = ((Index - 1) * Datenbreite (Elementtyp))
   Verwenden Sie nach Möglichkeit als Elemettyp nur Y,W,X, R. Der Elementtyp wird hier nur zur Ermittlung der Datenbreite benötigt. Die Art der Darstellung wird mit der Formatangabe festgelegt.
2. Fügen Sie diesen Block an den Stellen im Meldungstext ein, an denen der Begleitwert angezeigt werden soll.

Siehe auch

Aufbau von Begleitwerten (Seite 7831)

Beispiele für Begleitwerte (Seite 7833)

18.9.11 Aufbau von Begleitwerten

Begleitwerte bestehen aus den folgenden Bestandteilen:

Elementtyp

Hiermit wird der Datentyp des Begleitwertes eindeutig projektiert:

Elementtyp	Datentyp
Y	BYTE
W	WORD
X	DWORD
I	Integer
D	DINT
B	BOOL
C	CHAR
R	REAL
O	LREAL

Der Elementtyp macht nur den von der AS übertragenen Datentyp eindeutig. Er wird nicht als Casting Operator verwendet.

Formatangabe

Bestimmen Sie das Ausgabeformat für den Begleitwert auf dem Anzeigegerät. Die Formatangabe wird durch das Zeichen "%" eingeleitet. Für Meldungstexte gibt es folgende feste Formatangaben:

Formatangabe	Beschreibung
%[i]X	Hexadezimalzal mit i Stellen
%[i]u	Dezimalzahl ohne Vorzeichen mit i Stellen
%[i]d	Dezimalzahl mit Vorzeichen mit i Stellen
%[i]b	Binärzahl mit i Stellen
%[i][.y]f	Gleitpunktzahl mit Vorzeichen mit y Stellen nach dem Dezimalpunkt und i Gesamtstellen
%[i]s	Zeichenkette (ANSI String) mit i Stellen Zeichen werden gedruckt bis zum ersten 0 Byte (00Hex).
%t#<Name der Textliste>	Zugriff auf Textliste

Ist die Stellenanzahl [i] zu klein, so wird der Wert trotzdem in voller Länge ausgegeben.

Ist die Stellenanzahl [i] zu groß, so wird vor dem Wert eine passende Anzahl Füllzeichen ausgegeben.

Ist die Stellenanzahl [i] zu groß und wird (bei Dezimal-/Gleitpunktzahlen) ein Vorzeichen ausgegeben, so verringert sich die Anzahl der Füllzeichen um eine Stelle. Gleiches gilt für das Punkt- bzw. Kommazeichen innerhalb von Gleitpunktzahlen.

Beispiel: Wert=1234.567 Format=%8.3f Ausgabe=1234.567; d.h. es wird kein Füllzeichen ausgegeben, obwohl nur 7 Ziffern vorhanden sind.

Hinweis

Bitte beachten Sie, dass Sie "[i]" optional angeben können, wobei Sie die eckigen Klammern bei der Eingabe weglassen müssen.

Siehe auch

Begleitwerte in Meldungen einfügen (Seite 7831)

Beispiele für Begleitwerte (Seite 7833)

18.9.12 Beispiele für Begleitwerte

Beispiele für Begleitwerte:

@1I%6d@: Der Wert aus Begleitwert 1 wird als Dezimalzahl mit maximal 6 Stellen dargestellt.

@2R%6f@: Der Wert "5.4" zum Beispiel aus Begleitwert 2 wird als Festpunktzahl "5.4" dargestellt (drei führende Leerzeichen).

@2R%2f@: Der Wert "5.4" zum Beispiel aus Begleitwert 2 wird als Festpunktzahl "5.4" dargestellt (bei zu kleiner Stellenzahl wird nicht abgeschnitten).

@1W%t#Textbib1@: Begleitwert 1 vom Datentyp WORD ist der Index, mit dem in der Textbibliothek Textbib1 der einzusetzende Text referenziert wird.

Beispiele für Formatangabe und ausgegebene Werte:

Wert=255 Format=%5X Ausgabe="16#000FF"

Wert=123 Format=%5u Ausgabe=" 123"

Wert=-200 Format=%4d Ausgabe="-200"

Wert=19 Format=%8b Ausgabe="2#00010011"

Wert=1234.567 Format=%8.3f Ausgabe="1234.567"; d.h. es wird kein Leerzeichen ausgegeben, obwohl nur 7 Ziffern vorhanden sind

Hinweis

Möchten Sie einem der ALARM_S-Bausteine mehr als einen Begleitwert übergeben, können Sie einen Array mit maximal 12 Byte Länge übergeben. Dies können z. B. maximal 12 Byte oder Char, maximal 6 Word oder Int bzw. maximal 3 DWord, Real oder DInt sein.

Siehe auch

Begleitwerte in Meldungen einfügen (Seite 7831)

Aufbau von Begleitwerten (Seite 7831)

18.9.13 Begleitwerte löschen

Sie können Begleitwerte löschen, indem Sie im Meldungstext den Platzhalter löschen, der einen Begleitwert repräsentiert.

Vorgehen:

Um Begleitwerte zu löschen, gehen Sie folgendermaßen vor:

1. Suchen Sie den Platzhalter im Meldungstext, der dem Begleitwert, den Sie löschen wollen, entspricht.
   Der Platzhalter beginnt mit dem Zeichen "@", enthält dann den Index und Elementtyp, an dem Sie den Begleitwert erkennen können, sowie eine Formatangabe, und endet mit dem Zeichen "@".
2. Löschen Sie den Platzhalter, den Sie gefunden haben, aus dem Meldungstext heraus.

18.10 Meldungen anlegen und bearbeiten

18.10.1 Programmmeldungen anlegen

Voraussetzung

Sie haben einen Funktionsbaustein angelegt.

Vorgehen

Um eine Programmmeldung anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Ordner "Programmbausteine" in der Projektnavigation den Funktionsbaustein (FB) aus, für den Sie eine Programmmeldung anlegen wollen und öffnen Sie diesen Baustein durch Doppelklick.
2. Fügen Sie im Anweisungsfenster bzw. Netzwerk (je nach Bausteinsprache) des FBs den Aufruf für den ausgewählten Meldebaustein ein ("Program_Alarm"). Dieser befindet sich in der Task Card "Anweisungen", unter "Erweiterte Anweisungen" > "Meldungen". Die Bausteinschnittstelle wird im Bereich "Static" automatisch mit den notwendigen Informationen versorgt.
3. Ändern Sie ggf. den Meldungsnamen im aufgeblendeten Dialog.
   Ergebnis: Im Anweisungsteil des FBs werden die Eingangsvariablen des aufgerufenen Meldebausteins, hier also des Program_Alarm-Bausteins, angezeigt.
4. Wiederholen Sie die Schritte 2 bis 3 für alle Aufrufe von Meldebausteinen in diesem FB.

Beachten Sie, dass Sie beim Anlegen von Meldungen bei S7-1500 CPUs die Meldeklasse angeben müssen. Diese spezifiziert, ob die Meldung quittierpflichtig ist oder nicht. Nicht-quittierpflichtige Meldungen können auch nur zu Informationszwecken dienen.

Hierdurch ergeben sich drei Meldungsarten:

- Die Meldung muss quittiert werden und hat den Staus "kommend" oder "gehend".
- Die Meldung muss nicht quittiert werden und hat den Staus "kommend" oder "gehend".
- Die Meldung dient nur zu Informationszwecken und hat keinen Status. Das Häkchen in der Spalte "Nur Information" muss gesetzt sein.

Hinweis

Die Standardeinstellung ist "ohne Quittierung".

Siehe auch

Meldeklassen anlegen (Seite 7853)

18.10.2 Programmmeldungen im Meldungseditor bearbeiten

Voraussetzung

Sie haben eine Programmmeldung angelegt.

Vorgehen

Um Programmmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen". Der Meldungseditor wird geöffnet.
2. Geben Sie die gewünschten Texte und Attribute in die entsprechenden Spalten ein.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.10.3 Programmmeldungen im Programmiereditor bearbeiten

Voraussetzung

Sie haben eine Programmmeldung angelegt.

Vorgehen

Um Programmmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Markieren Sie die entsprechende Zeile in der Bausteinschnittstelle.
2. Wechseln Sie in das Register "Meldung" im Inspektorfenster.
3. Geben Sie die gewünschten Texte und Attribute in die entsprechenden Felder ein.

Meldebaustein "Program_Alarm"

Wenn Sie den Meldebaustein "Program_Alarm" verwenden, wird Ihnen im Inspektorfenster das zusätzliche Register "Konfiguration" angezeigt. An den leeren Eingangskanälen SD_1 bis SD_10 können Sie zusätzliche Begleitwerte eintragen, die z.B. nicht in Texten vorkommen.

Klicken Sie hierzu auf das Auswahlfeld neben dem Parameter und wählen Sie den gewünschten Wert aus.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.10.4 Programmmeldungen löschen

Vorgehen

Um eine Programmmeldung zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, der die zu löschende Meldung enthält.
2. Markieren Sie in der Bausteinschnittstelle die entsprechende Zeile und wählen Sie im Kontextmenü den Befehl "Löschen".
3. Entfernen Sie den Meldebaustein aus dem Anweisungsfenster bzw. Netzwerk (je nach Bausteinsprache).

Ergebnis

Die Meldung wird gelöscht.

18.10.5 Meldungstyp bearbeiten

Vorgehen

Um einen Meldungstyp zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gewünschten Meldebaustein.
2. Tragen Sie die gewünschten Texte ein bzw. wählen Sie die gewünschten Attribute aus.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.10.6 Instanz-DB anlegen und Meldungsinstanzen bearbeiten

Voraussetzung

Sie müssen zuvor einen FB angelegt und in diesem mindestens eine Meldung angelegt haben.

Vorgehen

Um einem Meldungstyp Instanz-Datenbausteine (DBs) zuzuweisen und die Meldungen für diese DBs instanzspezifisch zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Neuen Baustein hinzufügen", klicken Sie im aufgeblendeten Dialog auf die Schaltfläche "Datenbaustein (DB)" und wählen Sie aus der Klappliste bei "Typ" den Funktionsbaustein (Meldungstyp) aus, dem Sie den Instanz-Datenbaustein zuordnen möchten.
2. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen", um die Meldungsprojektierung zu öffnen.
3. Tragen Sie die gewünschten Änderungen für die jeweilige Meldungsinstanz ein.

Ergebnis

Damit ist die Meldungsprojektierung für den angelegten Instanz-DB abgeschlossen.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.10.7 Begleitwerte in Meldungen einfügen

Um Meldungen aktuelle Informationen, z. B. aus dem Prozess, mitzugeben, können Sie Begleitwerte an verschiedenen Stellen eines Meldungstextes einfügen.

Vorgehen über das Kontextmenü

Um einen Begleitwert in eine Meldung einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie an der gewünschten Stelle das Kontextmenü und wählen über den entsprechenden Menübefehl aus, ob Sie ein Schlüsselwort, eine Variable oder einen Eintrag aus einer Textliste einfügen möchten.
2. Tragen Sie im aufgeblendeten Dialog die gewünschten Parameter ein und bestätigen Sie Ihre Eingaben.

Hinweis

Sie können Begleitwerte sowohl im Meldungseditor als auch im Bausteineditor projektieren. Wenn Sie im Meldungseditor arbeiten, sollten Sie den Baustein, für den Sie die Meldung projektieren, im Bausteineditor geöffnet haben. Andernfalls können im Auswahlfeld "Variable" nur bereits projektierte Variablen verschaltet bzw. angezeigt, aber keine neuen Variablen hinzugefügt bzw. erstellt werden.

Wenn Sie im Meldungseditor Begleitwerte eingeben, müssen Sie sich im Bearbeitungsmodus befinden (klicken Sie zwei Mal in das gewünschte Feld), um das benötigte Kontextmenü anzeigen zu lassen.

Ergebnis

Der Begleitwert wird in die Meldung integriert.

Bei Auftreten eines Syntaxfehlers wird der Text rot hinterlegt und muss korrigiert werden. Nicht jede Kombination aus Variable und Formatierung ist möglich.

Beim Eingeben von Variablen und Textlisteneinträgen werden die Felder "Adresse" und Meldeparameter automatisch ausgefüllt. Sie werden erst beim erneuten Öffnen des Dialogs angezeigt. Bei geöffnetem Bausteineditor wird der Begleitwert am Eingangskanal SD_1 bis SD_10 automatisch hinzugefügt.

Hinweis

Wenn Sie verschiedene Projektsprachen eingestellt haben, werden die Begleitwerte in alle Projektsprachen kopiert. Änderungen, die Sie ggf. vornehmen, werden automatisch in allen Projektsprachen übernommen. Stellen Sie beim Übersetzen der Texte sicher, dass die gleiche Reihenfolge in allen Sprachen eingehalten wird!

Das Kopieren von eingebetteten Daten zwischen verschiedenen Program_Alarm-Bausteinen führt zu Fehlern, sofern nicht beide Bausteine über eine identische Anordnung von SD-Variablen des gleichen Datentyps und Formats verfügen.

Vorgehen ohne Kontextmenü

Sie können Begleitwerte an beliebigen Stellen eines Meldungstextes oder Infotextes einfügen. Gehen Sie dazu folgendermaßen vor:

1. Fügen Sie Ihren "Program_Alarm"-Baustein in Ihr SPS-Programm ein.
2. Belegen die den SD-Eingang des "Program_Alarm"-Bausteins mit der Variablen, deren Wert Sie im "Program_Alarm"-Meldtext oder im zugehörigen Infotext anzeigen möchten.
3. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen" > "Programmmeldungen".
4. Wählen Sie Ihren neuen "Program_Alarm" aus und fügen Sie den Verweis auf den gewünschten Begleitwert in der Form "@SD-Nummer%Formatangabe@" an der gewünschten Stelle im Meldetext oder im Infotext ein.

Hinweis

Wenn Sie verschiedene Projektsprachen eingestellt haben, werden die Begleitwerte nicht in alle Projektsprachen kopiert. Sie müssen die Begleitwerte in der apatierten klassischen Schreibweise für jede Sprache neu einfügen. Ein Vorgabe, dass die Begleitwerte in jeder Sprache in der gleichen Reihenfolge stehen müssen, gibt es hier nicht.

Siehe auch

Beispiele für Begleitwerte (Seite 7842)

Einstellungen beim Eingeben von Begleitwerten (Seite 7840)

Besonderheiten bei Begleitwerten (Seite 7841)

Verwendung der Begleitwerte (Seite 8255)

18.10.8 Einstellungen beim Eingeben von Begleitwerten

Sie können Begleitwerte sowohl im Meldungseditor als auch im Bausteineditor in Meldungen einfügen. Wenn Sie im Meldungseditor arbeiten, sollten Sie den Baustein, für den Sie die Meldung projektieren, im Bausteineditor geöffnet haben. Dadurch werden Ihnen im Auswahlfeld "Variable" die möglichen Parameter angezeigt, die Sie dann übernehmen können. Bei geschlossenem Bausteineditor werden Ihnen nur die bereits verwendeten Parameter angezeigt.

Schlüsselwort eingeben

In der Klappliste werden Ihnen die möglichen Schlüsselwörter zur Auswahl angeboten, die Sie im Text definieren können.

Textlisteneintrag als Begleitwert eingeben

In der Klappliste wählen Sie die gewünschte Textliste aus. Im Auswahlfeld "Variable" wählen Sie die Variable aus, die den Index für die Textliste zur Laufzeit des Programms bereithält.

Variable als Begleitwert eingeben

Im Auswahlfeld "Variable" wählen Sie die gewünschte Variable aus. Die Adresse und der Meldeparameter werden vom Programm automatisch ergänzt. Im Abschnitt "Format" müssen Sie nun den für die spezifizierte Variable passenden Anzeigetyp, die minimale Länge und ggf. die Anzahl der Nachkommastellen definieren.

Hinweis

Mit einer älteren Version des TIA Portals erstellte Projekte können noch die manuelle Schreibweise bei eingebetteten Begleitwerten enthalten. Diese entspricht der Syntax @<Nummer des Begleitwerts>%<Formatangabe>@. Informationen zum Parameter "Formatangabe" finden Sie unter Aufbau von Begleitwerten (Seite 7831).

Es wird empfohlen, bei neuen Projekten nur die oben beschriebenen Dialoge zum Eingeben von Begleitwerten zu verwenden.

Wenn Sie in Ihrem Programm ARRAY-Zugriffe mit variablem Index in der Form #Program_Alarm_1[#myIndex] verwenden, steht zum Zeitpunkt der Programmierung noch nicht fest, welches konkrete ARRAY-Element zur Laufzeit verwendet wird. Wenn Sie einen solchen ARRAY-Zugriff im Programmiereditor selektieren, werden deshalb im Inspektorfenster immer die Eigenschaften des ersten ARRAY-Elements angezeigt.

Siehe auch

Schlüsselwörter für Meldetexte (Seite 7844)

Begleitwerte in Meldungen einfügen (Seite 7838)

18.10.9 Besonderheiten bei Begleitwerten

Hochrüsten älterer Projekte

Mit einer älteren Version des TIA Portals erstellte Projekte können eingebettete Begleitwerte enthalten, die in den verschiedenen Projektsprachen unterschiedlich strukturiert sind. Die Begleitwerte können beispielsweise eine andere Reihenfolge haben oder fehlen in einer anderen Sprache. Beim Hochrüsten eines solchen Projekts wird die Reihenfolge der Begleitwerte vereinheitlicht. Sie richtet sich nach der im Ursprungsprojekt eingestellten Referenzsprache. In diesem Fall ist es empfehlenswert, die übersetzten Texte prüfen und korrigieren zu lassen.

Importieren einer übersetzten Textliste < V14

Es kann vorkommen, dass Sie in einer älteren Version eine Textliste für Meldungen exportiert, übersetzt und wieder importiert haben. Wenn Sie in einem aktuellen Projekt versuchen, eine alte Datei (z. B. wegen Namensgleichheit) zu importieren, wird dies wegen Dateninkompatibilität abgewiesen.

18.10.10 Beispiele für Begleitwerte

Beispiele für Begleitwerte, die über das Kontextmenü eingegeben werden:

Schlüsselwort:

<Schlüsselwort: CpuName>

Der Name der CPU wird als Begleitwert eingetragen.

Variable:

<Variable: #Temperature>

Der Wert der Variable "#Temperature" wird an dieser Stelle im Meldungstext angezeigt.

Textlisteneintrag:

<Textliste: USER_1: "tag_1">

Der Inhalt der Variable "tag_1" wird als Index für die Textliste "USER_1" benutzt. Der Eintrag aus der Textliste wird an dieser Stelle im Meldungstext angezeigt.

Beispiele für Begleitwerte, die ohne Kontextmenü eingegeben werden:

@1%6d@: Der Wert aus Begleitwert 1 (SD_1) wird als Dezimalzahl mit maximal 6 Stellen dargestellt.

@2%6f@: Der Wert "5.4" zum Beispiel aus Begleitwert 2 (SD_2) wird als Festpunktzahl "5.4" dargestellt (drei führende Leerzeichen).

@2%2f@: Der Wert "5.4" zum Beispiel aus Begleitwert 2 (SD_2) wird als Festpunktzahl "5.4" dargestellt (bei zu kleiner Stellenzahl wird nicht abgeschnitten).

@10%t#Textbib1@: Der Begleitwert 10 (SD_10) wird mit dem aus der Textbibliothek "Textbib1" einzusetzenden Text dargestellt.

Beispiele für Formatangabe und ausgegebene Werte:

Wert=255 Format=%5X Ausgabe="16#000FF"

Wert=123 Format=%5u Ausgabe=" 123"

Wert=-200 Format=%4d Ausgabe="-200"

Wert=19 Format=%8b Ausgabe="2#00010011"

Wert=1234.567 Format=%8.3f Ausgabe="1234.567"; d.h. es wird kein Leerzeichen ausgegeben, obwohl nur 7 Ziffern vorhanden sind

Siehe auch

Begleitwerte in Meldungen einfügen (Seite 7838)

18.10.11 Begleitwerte löschen

Vorgehen:

Um Begleitwerte zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Programmiereditor den das Register "Konfiguration".
2. Wählen Sie den gewünschten Begleitwert aus und löschen ihn.
3. Löschen Sie den Begleitwert ebenfalls aus den Meldetexten im Meldungseditor.

Oder

1. Selektieren Sie im Programmiereditor den gewünschten Eingangskanal.
2. Löschen Sie den Begleitwert.
3. Löschen Sie den Begleitwert ebenfalls aus den Meldetexten im Meldungseditor.

18.11 Texte und Attribute

18.11.1 Schlüsselwörter für Meldetexte

Übersicht über die Schlüsselwörter für Meldetexte

Sie können die folgenden Schlüsselwörter in Ereignistexten, Infotexten oder Zusatztexten auswählen, die vom Engineering-System durch ihre realen Werte ersetzt werden.

PLC-Name

Name der projektierten PLC.

Beispiel: <Schlüsselwort: CpuName> = "PLC_1"

Name des Instanz-DBs

Ort der Meldungsinstanz, üblicherweise ein Datenbaustein.

Beispiel: <Schlüsselwort: Instance> = "Block_3_DB"

FB-Verschachtelung zur Meldung

Wenn Sie im Projekt geschachtelte Bausteine verwenden, wird der Variablenpfad vom meldungsdefinierenden Baustein durch alle Multiinstanzen hinweg angezeigt, der vom Funktionsbaustein, von dem der Datenbaustein erzeugt wurde, hinführt.

Beispiel: <Schlüsselwort: Path> = "myFB2\myFB1"

Hinweis

Bei nicht geschachtelten Bausteinen wird das Schlüsselwort "Path" durch einen Leerstring ersetzt.

Name der Meldung

Name der Meldung.

Beispiel: <Schlüsselwort: Name> = "myAlarm"

Hinweis

Schlüsselwörter können nicht in Textlisten eingegeben werden.

18.11.2 Texte eingeben

Sie können die Texte für die Meldungsinstanzen manuell eingeben oder Sie können Defaultwerte benutzen, die Sie in den Texten am Meldungstyp eingeben. Klicken Sie dazu in die entsprechende Zeile und geben Sie den Text ein. Wenn Sie den Text gegen Überschreiben sichern wollen, klicken Sie die entsprechende Option in der Spalte an. Die Texte können Zeilenumbrüche enthalten.

Hinweis

Verwendung des Zeichens @ in einem Textfeld

Bitte verwenden Sie das Zeichen @ nur für die Deklaration der Begleitwerte, z. B. @4%6d@. Wenn Sie das Zeichen @ darüber hinaus im Freitext eines Textfelds verwenden, dann kann das unter Umständen dazu führen, dass die im Freitext enthaltenen Begleitwerte in der Meldung nicht aufgelöst werden können.

Textvorlage aus dem Meldungstyp

Sämtliche Texte aus dem Meldungstyp stehen Ihnen bei der Erstellung der Meldungstexte als Vorlage zur Verfügung. Enthält also der Meldungstyp bereits einen allgemeinen Text, so beinhalten alle Instanzen dieses Meldungstyps dieselben Attribute und Texte. Bei Bedarf können Sie diese an der Meldungsinstanz abändern.

Ereignistext

Ereignistexte können auf allen Anzeigegeräten angezeigt werden.

Infotext

Der Infotext ist ein Text, der für bestimmte Anzeigegeräte zusätzlich angegeben werden kann.

Zusatztexte

Zusatztexte sind Texte, die von bestimmten Anzeigegeräten angezeigt werden können.

18.11.3 Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen

Sie haben die Möglichkeit, Texte und Attribute am Meldungstyp zu verriegeln, damit diese an den Meldungsinstanzen nicht verändert werden können. Ab der Version V13 SP1 werden die Verriegelungssymbole standardmäßig nicht angezeigt und bei neu erzeugten Meldungen automatisch gesetzt.

Verriegeln der Daten beim Meldungstyp

Zur Erleichterung der Erstellung der Meldungen nutzen Sie die Meldungstypen als Vorlage.

- Wenn Sie Daten (Attribute und Texte) für den Meldungstyp eingeben, können Sie festlegen, ob sie verriegelt werden sollen oder nicht. Bei verriegelten Daten wird das Symbol eines geschlossenen Kettenglieds neben das Eingabefeld gesetzt. Bei nicht verriegelten Attributen ist das Kettenglied offen.
- Im Meldungstyp verriegelte Daten können Sie in den instanzspezifischen Meldungen nicht mehr ändern. Die Daten werden nur angezeigt.
- Wenn Sie dennoch Änderungen benötigen, müssen Sie wieder zum Meldungstyp zurück gehen, die Verriegelung wieder aufheben, und die Änderungen dort durchführen.

Hinweis

Wenn Sie Meldungsinstanzen im Meldungseditor verändert haben, bei denen die Texte/ Attribute im Meldungstyp nicht verriegelt sind, so erscheint ein Typsymbol vor der entsprechenden Spalte. Wenn Sie auf dieses Symbol klicken, werden die geänderten Texte/ Attribute wieder auf den Wert des Meldungstyps zurückgesetzt.

Siehe auch

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.11.4 Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden

Verriegelungssymbole anzeigen/ausblenden

Vorgehen

Um die Verriegelungssymbole anzuzeigen bzw. auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Meldungen", klicken Sie in das Optionskästchen "Verriegelungssymbole anzeigen" und setzen bzw. entfernen Sie das Häkchen. Das Setzen des Häkchens bewirkt, dass die Verriegelungssymbole angezeigt werden, das Entfernen des Häkchens blendet die Symbole aus.

Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

Hinweis

Standardmäßig ist die Option deaktiviert.

Verriegelung für neue Programmmeldungen setzen/entfernen

Vorgehen

Um die Verriegelung für neue Programmmeldungen zu setzen bzw. zu entfernen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Meldungen", klicken Sie in das Optionskästchen "Verriegelungen beim Erstellen einer neuen Programmmeldung setzen" und setzen bzw. entfernen Sie das Häkchen.

Hinweis

Standardmäßig ist die Option aktiviert.

18.11.5 Texte verriegeln

Option "Verriegelt" beim Meldungstyp

Sie können Texte nur an Meldungstypen verriegeln. Als Kennzeichen für die Verriegelung dient das Symbol neben dem Eingabefeld. Verriegelte Texte sind bei allen Meldungsinstanzen, die vom betrachteten Meldungstyp abgeleitet sind, schreibgeschützt.

Voraussetzung

Die Verriegelungssymbole sind eingeschaltet.

Hinweis

Die Reihenfolge der nachstehenden Vorgehensweise ist beliebig.

Verriegeln von Texten

Um Texte zu verriegeln, gehen Sie folgendermaßen vor:

- Beginnen Sie mit der Bearbeitung der Meldungstypen.
- Klicken Sie links neben dem Eingabefeld, das Sie verriegeln wollen, das Symbol an.
  Ergebnis: Das Symbol ändert sich in ein geschlossenes Kettenglied.

Entriegeln von Texten

Um die Verriegelung von Texten aufzuheben, gehen Sie folgendermaßen vor:

- Beginnen Sie mit der Bearbeitung des Meldungstypen.
- Klicken Sie links neben dem Eingabefeld, dessen Verriegelung Sie aufheben, auf das Symbol.
  Ergebnis: Das Symbol ändert sich in ein geöffnetes Kettenglied.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

Texte eingeben (Seite 7845)

18.11.6 Attribute verriegeln

Verriegeln von Attributen beim Meldungstyp

Sie können Attribute nur an Meldungstypen verriegeln. Als Kennzeichen für die Verriegelung dient das Symbol neben dem Eingabefeld. Verriegelte Attribute sind bei allen Meldungsinstanzen, die vom betrachteten Meldungstyp abgeleitet sind, schreibgeschützt.

Voraussetzung

Die Verriegelungssymbole sind eingeschaltet.

Hinweis

Die Reihenfolge der nachstehenden Vorgehensweise ist beliebig.

Verriegeln von Attributen

Um Attribute zu verriegeln, gehen Sie folgendermaßen vor:

- Beginnen Sie mit der Bearbeitung der Meldungstypen.
- Klicken Sie in der Tabelle auf das Symbol links neben dem Eingabefeld, das Sie verriegeln wollen.
  Ergebnis: Das Symbol ändert sich in ein geschlossenes Kettenglied.

Entriegeln von Attributen

Um Attribute zu entriegeln, gehen Sie folgendermaßen vor:

- Beginnen Sie mit der Bearbeitung des Meldungstypen.
- Klicken Sie in der Tabelle auf das Symbol links neben dem Eingabefeld, das Sie entriegeln wollen.
  Ergebnis: Das Symbol ändert sich in ein offenes Kettenglied.

Siehe auch

Grundlagen zum Verriegeln/Entriegeln von Texten und Attributen (Seite 7845)

Verriegelungen für neue Meldungen und Verriegelungssymbole anzeigen/ausblenden (Seite 7846)

18.12 Textlisten für Meldungen

18.12.1 Grundlagen zu Textlisten für Meldungen

Sie können bestehende Textlisten (anwenderdefinierte und systemdefinierte Textlisten) Ihren Anforderungen anpassen und Texte und Attribute editieren. Die Texte können Sie dann in die gewünschte(n) Projektsprache(n) übersetzen.

Wird ein meldender Baustein, der eine Textliste referenziert, in ein anderes Programm kopiert, so müssen Sie die zugehörigen Textlisten ebenfalls kopieren, bzw. eine andere Textliste mit gleichem Namen und Inhalt anlegen oder den Begleitwert im Meldetext ändern.

Hinweis

Beachten Sie, dass die Namen von Textlisten keine der folgenden Sonderzeichen enthalten dürfen:

\ / : * ? " < > | @

Wenn Sie auf eine Textliste referenzieren, die eines dieser Zeichen enthält, so wird ein Syntaxfehler erkannt. Der Text rot hinterlegt und muss korrigiert werden.

Detaillierte Informationen zu Textlisten erhalten Sie im Kapitel "Mit Textlisten arbeiten".

Siehe auch

Grundlagen zum Exportieren und Importieren von Meldetexten (Seite 7858)

18.12.2 Textlisten für Meldungen bearbeiten

Voraussetzung

- Sie haben eine anwenderdefinierte Textliste angelegt.

Vorgehen

Um anwenderdefinierte Textlisten zu bearbeiten gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "PLC-Meldetextlisten" unterhalb des Geräts. Der Textlisteneditor wird geöffnet.
2. Wählen Sie in der Tabelle die Textliste aus, die Sie bearbeiten möchten.
3. Ändern Sie die gewünschten Werte.
   Sie können die zuvor angelegten Texte ändern oder neue Texte in eine bestehende Textliste einfügen.

Hinweis

Beachten Sie, dass Textlisten für Meldungen keine der folgenden Sonderzeichen enthalten dürfen:

\ / : * ? " < > | @

Wenn Sie auf eine Textliste referenzieren, die eines dieser Zeichen enthält, so wird ein Syntaxfehler erkannt. Der Text rot hinterlegt und muss korrigiert werden.

18.12.3 Texte aus Textlisten in Meldungen integrieren

In eine Meldung können Sie Texte aus verschiedenen Textlisten integrieren. Die Texte können frei platziert werden, so ist auch die Verwendung in fremdsprachigen Meldungen gewährleistet.

Vorgehen S7-300/400

Um Texte aus Textlisten in Meldungen zu integrieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen". Der Meldungseditor wird geöffnet.
2. Geben Sie an der Stelle in der Meldung, an welcher der Text aus der Textliste erscheinen soll, einen Begleitwert vom Format @[Index][Datentyp]%t#[Textliste]@ ein. Der Datentyp muss vom Typ WORD (W) sein.

   Hinweis

   S7-300/400: [Index] = z. B. 1, wobei 1 der erste Begleitwert der Meldung ist.

Vorgehen S7-1500

Um Texte aus Textlisten in Meldungen zu integrieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "PLC-Überwachungen & -Meldungen". Wählen Sie das Register "Meldungen". Der Meldungseditor wird geöffnet.
2. Gehen Sie an die Stelle in der Meldung, an welcher der Text aus der Textliste erscheinen soll, und selektieren Sie im Kontextmenü die Option zum Einfügen einer Textliste.
3. In der Klappliste wählen Sie die gewünschte Textliste aus. Wählen Sie dann im Auswahlfeld "Variable" die Variable aus, die im Automatisierungssystem den Index des Textlisteneintrags enthalten wird.

   Hinweis

   Wenn Sie im Meldungseditor arbeiten, sollten Sie den Baustein, für den Sie die Meldung projektieren, im Bausteineditor geöffnet haben.

   Wenn Sie im Meldungseditor Begleitwerte eingeben, müssen Sie sich im Bearbeitungsmodus befinden (klicken Sie zwei Mal in das gewünschte Feld), um das benötigte Kontextmenü anzeigen zu lassen.

Siehe auch

Aufbau von Begleitwerten (Seite 7831)

18.12.4 Beispiel für das Integrieren von Texten aus Textlisten in Meldungen

S7-300/400: Projektierter Meldetext: Druck ist @2W%t#USER_1@.

S7-1500: Projektierter Meldetext: Druck ist <Textliste: USER_1: "tag_1">.

Textliste mit dem Namen "USER_1":

Index	deutsch	englisch
1734	zu hoch	too high
1735	zu niedrig	too low

Annahme: Der Begleitwert der Variable "tag_1" ist mit dem Wert 1734 versorgt.

Ergebnis: Folgender Meldetext wird zur Laufzeit am Bediengerät angezeigt (wenn dort die Projektsprache "deutsch" eingestellt ist) : Druck ist zu hoch.

Annahme: Der Begleitwert der Variable "tag_1" ist mit dem Wert 1735 versorgt.

Ergebnis: Folgender Meldetext wird zur Laufzeit am Bediengerät angezeigt (wenn dort die Projektsprache "deutsch" eingestellt ist) : Druck ist zu niedrig.

Siehe auch

Texte aus Textlisten in Meldungen integrieren (Seite 7851)

18.13 Meldeklassen

18.13.1 Meldeklassen anlegen

Meldeklassen sind frei projektierbar. Sie können im Meldeklasseneditor angelegt und bearbeitet werden. Im Meldungseditor kann dann einer Meldung eine Meldeklasse zugeordnet werden.

Vom System sind bereits zwei Meldeklassen, "Acknowledgement" (für Meldungen mit Quittierung) und "No Acknowledgement" (für Meldungen ohne Quittierung) angelegt. Bei diesen kann nur der Anzeigename verändert werden.

Insgesamt können 16 Meldeklassen angelegt werden, denen Sie die entsprechende Priorität zuweisen können Die Eigenschaft "Mit Quittierung" der Meldeklassen gibt die Quittierpflicht der zugeordneten Meldungen vor.

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um eine Meldeklasse anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Meldeklassen".
   Der Meldeklasseneditor wird geöffnet.
2. Wählen Sie im Kontextmenü den Befehl "Neue Meldeklasse einfügen".
   Ergebnis: Eine neue Meldeklasse mit einem automatisch vergebenen Namen wurde angelegt.
3. Vergeben Sie ggf. in der Spalte "Name" einen anderen Namen für die neue Meldeklasse.
   Der hier vergebene Name ist sprachneutral.
4. Vergeben Sie ggf. in der Spalte "Anzeigename" einen Anzeigennamen. Dieser Name ist übersetzbar.
5. Geben Sie in der Spalte "Mit Quittierung" an, ob die Meldungen dieser Meldeklasse quittierpflichtig sind oder nicht.
6. Stellen Sie ggf. eine Priorität ein. Dies ist nur bei anwenderdefinierten Meldeklassen möglich.
   Die Priorität wird für den Meldungstyp übernommen.

Hinweis

Beachten Sie, dass Sie beim Anlegen von Meldungen bei S7-1500 CPUs die Meldeklasse angeben müssen. Diese spezifiziert, ob die Meldung quittierpflichtig ist oder nicht.

S7-1500: Programmalarme, bei denen die Option „"Nur Information“" ausgewählt ist, sollten möglichst kurzzeitig gesetzt werden! Damit wird eine Mehrfacharchivierung (Meldearchiv) beim Download nach Programmalarmtextänderungen vermieden.

18.13.2 Meldeklassen bearbeiten

Sie können jederzeit die Einstellungen (Name, Anzeigename, Priorität oder Quittierung) einer Meldeklasse ändern, auch wenn dieser Meldeklasse bereits Meldungen zugeordnet sind. Die Änderungen werden automatisch in die Meldungen übernommen (falls die Einstellungen an der Meldung nicht überschrieben wurden).

Meldeklassen kopieren

Um Meldeklassen zu kopieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile mit der Meldeklasse, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

Die kopierte Meldeklasse wird unter einem neuen Namen an die letzte Stelle der Tabelle abgelegt.

Der Name der neuen Meldeklasse setzt sich folgendermaßen zusammen:

<alter Name>_<Nr.>

Nr.: kleinste freie natürliche Zahl

Hinweis

- Beachten Sie, dass Sie beim Anlegen von Meldungen bei S7-1500 CPUs die Meldeklasse angeben müssen. Diese spezifiziert, ob die Meldung quittierpflichtig ist oder nicht.
- Bei vom System angelegten Meldeklassen können Sie nur den Anzeigenamen verändern.

18.13.3 Meldeklassen löschen

Da maximal 16 Meldeklassen angelegt werden können, kann es notwendig sein, nicht mehr benötigte Meldeklassen zu löschen.

Vorgehen

Zum Löschen einer Meldeklasse gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Meldeklassen".
   Der Meldeklasseneditor wird geöffnet.
2. Selektieren Sie die gewünschte Meldeklasse und klicken Sie in der Funktionsleiste auf das Symbol "Löschen".

Ergebnis

Die Meldeklasse wurde gelöscht.

Hinweis

Vom System erzeugte Meldklassen können nicht gelöscht werden.

18.13.4 Priorität der Meldeklasse

Bei Programmmeldungen und Anwenderdiagnosemeldungen können Sie für jeden Meldungstyp eine Priorität (0 bis 16) festlegen. Die Priorität einer Meldeklasse wird üblicherweise im Meldeklasseneditor spezifiziert bzw. geändert, sie kann aber nachträglich im Meldungseditor für einzelne Meldungen geändert werden. Der der Meldung zugeordnete Baustein darf nicht schreibgeschützt sein! Wenn Sie neue Meldungen anlegen, ist die Priorität standardmäßig auf "0" eingestellt.

Ändern der Priorität beim Meldungstyp und bei Meldungen

Wenn Sie nachträglich Daten beim Meldungstyp ändern, so werden diese Änderungen automatisch an den Instanzen übernommen. Ausnahme: Sie haben diese Daten zuvor an der Instanz geändert.

Die nachfolgende Tabelle listet die möglichen Aktionen und ihre Auswirkungen auf:

Ausgangszustand	Aktion	Ergebnis
Es existiert kein Meldungstyp.	Der Meldungstyp wird erzeugt.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität am Meldungstyp im Meldungseditor in der Spalte "Priorität" sind identisch.
Der Meldungstyp existiert, die Priorität ist unverändert.	Die Meldeklasse beim Meldungstyp wird geändert.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität am Meldungstyp im Meldungseditor in der Spalte "Priorität" sind identisch.
Der Meldungstyp existiert, die Priorität ist unverändert.	Die Priorität der Meldeklasse wird im Meldeklasseneditor geändert.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität am Meldungstyp im Meldungseditor in der Spalte "Priorität" sind identisch.
Der Meldungstyp existiert, die Priorität ist unverändert.	Die Meldeklasse wird im Meldeklasseneditor gelöscht.	Die Priorität des Meldungstyps wird auf "0" gesetzt und als "ungültig" markiert.
Der Meldungstyp existiert, die Priorität ist unverändert.	Die Priorität wird beim Meldungstyp geändert.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität am Meldungstyp im Meldungseditor in der Spalte "Priorität" sind unterschiedlich.
Der Meldungstyp existiert, die Priorität beim Meldungstyp wurde geändert.	Die Meldeklasse der Instanz wird geändert.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität an der Instanz im Meldungseditor in der Spalte "Priorität" sind unterschiedlich.

Ausgangszustand	Aktion	Ergebnis
Die Meldung (Instanz) existiert, die Priorität beim Meldungstyp wurde geändert.	Die Priorität der Meldeklasse wird geändert.	Die Prioriät der Meldeklasse im Meldeklasseneditor und die Priorität an der Instanz im Meldungseditor in der Spalte "Priorität" sindunterschiedlich.
Die Meldung (Instanz) existiert, die Priorität beim Meldungstyp wurde geändert.	Die Meldeklasse wird im Meldeklasseneditor gelöscht.	Die Priorität beim Meldungstyp wird angezeigt, die Meldeklasse wird als "ungültig" markiert.
Die Meldung (Instanz) existiert, die Priorität beim Meldungstyp wurde geändert.	Die Priorität beim Meldungstyp wird auf den gleichen Wert wie die Priorität der Meldeklasse geändert.	Die Prioriät der Meldeklasse und in der Spalte "Priorität" im Meldungseditor sind identisch, aber nicht gekoppelt (*).
Die Meldung (Instanz) existiert, die Priorität beim Meldungstyp wurde geändert.	Die Priorität beim Meldungstyp wird auf dir Priorität der Meldeklasse zurückgesetzt.	Die Prioriät der Meldeklasse und in der Spalte "Priorität" im Meldungseditor sind identisch.

Wenn Sie die Priorität nachträglich an der Meldung ändern, wird der Wert mit einem Sternchen (*) gekennzeichnet.

Sie können geänderte Werte jederzeit wieder auf die Priorität der Meldeklasse zurücksetzen. Wählen Sie hierzu in der Klappliste den Eintrag "Auf den Wert des Meldungstyps zurücksetzen".

18.13.5 Meldeklassen exportieren und importieren

18.13.5.1 Grundlagen zum Exportieren und Importieren von Meldeklassen

Beschreibung

Es ist möglich, die Meldeklassen eines Projekts zu exportieren und in ein anderes Projekt zu importieren. Dies hat den Vorteil, dass Meldeklassen nicht für jedes Projekt separat angelegt werden müssen.

Die Meldeklassen werden in eine Datendatei (*.dat) gespeichert.

18.13.5.2 Meldeklassen exportieren

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um die Meldeklassen zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Ordner "Gemeinsame Daten" und doppelklicken Sie auf den Eintrag "Meldeklassen".
2. Klicken Sie im Meldeklasseneditor auf das Symbol "Export der Meldeklasseneinstellungen" in der Funktionsleiste. Der Dialog "Speichern unter" wird geöffnet.
3. Wählen Sie den Pfad aus, in dem die Exportdatei abgelegt werden soll.
4. Klicken Sie auf die Schaltfläche "Speichern".

Ergebnis

Am ausgewählten Speicherort wird eine .dat-Datei angelegt.

18.13.5.3 Meldeklassen importieren

Voraussetzung

Das Projekt, in das die Meldeklassen importiert werden sollen, ist geöffnet.

Vorgehen

Um die Meldeklassen zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Ordner "Gemeinsame Daten" und doppelklicken Sie auf den Eintrag "Meldeklassen".
2. Klicken Sie im Meldeklasseneditor auf das Symbol "Import der Meldeklasseneinstellungen" in der Funktionsleiste. Der Dialog "Öffnen" wird geöffnet.
3. Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist.
4. Wählen Sie die gewünschte .dat-Datei aus.
5. Klicken Sie auf die Schaltfläche "Öffnen".

Hinweis

Das Symbol "Importieren" ist gegraut, wenn das geöffnete Projekt schreibgeschützt ist (z. B. Referenzprojekt).

Wenn es während des Imports zu Fehlern kommt, wird eine Meldung angezeigt und der Vorgang wird abgebrochen.

Ergebnis

Die Meldeklassen werden angezeigt.

18.14 Meldetexte exportieren und importieren

18.14.1 Grundlagen zum Exportieren und Importieren von Meldetexten

Einführung

Sie können Programmmeldetexte vom Meldungstyp in ein standardisiertes XLSX-Format exportieren und mit externen Tabelleneditoren bearbeiten (z. B. übersetzen). Ebenso können Sie diese Meldetexte, die mit externen Tabelleneditoren bearbeitet wurden, in das TIA Portal importieren.

Format der Exportdatei

Die Benennungen der Spaltentitel sind fest definiert und sollten nicht verändert werden. Die Reihenfolge der Spalten innerhalb des Tabellenblatts kann variieren. Die Anzahl der Spalten ist abhängig von der beim Export getroffenen Auswahl. Jede Meldung erhält eine separate Zeile in der Datei.

Die folgende Tabelle gibt an, welche Inhalte in den einzelnen Spalten erwartet werden:

<table>
<tr><th>Spalte</th><th colspan="2">Erläuterung</th><th>Übersetzungsrelevant</th></tr>
<tr><td>Location</td><td colspan="2">Name des Bausteins, dem die Meldung zugeordnet ist.</td><td>nein</td></tr>
<tr><td>Alarm name</td><td colspan="2">Name der Meldung</td><td>nein</td></tr>
<tr><td>"Alarm text" - English (United States) / [en-US] / Event text</td><td>Ereignistext der Meldung</td><td rowspan="3">Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein.
Wenn ein Begleitwert im Meldetext vorhanden ist, wird dem Text bei S7-1500 ein Ausdruck mit einer Referenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der Begleitwert einem Meldetext zugeordnet.</td><td>ja</td></tr>
<tr><td>"Info text" - English (United States) / [en-US] / Info text</td><td>Infotext der Meldung</td><td>ja</td></tr>
<tr><td>"Additional text" - English (United States) / [en-US] / Additional text</td><td>Zusatztext der Meldung</td><td>ja</td></tr>
<tr><td>FieldInfo (nur für S7-1500)</td><td colspan="2">Gibt an, ob Begleitwerte im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt.
Beispiel für Begleitwerte bei S7-1500:
Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;>
Textliste: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;></td><td>nein</td></tr>
</table>

Beispiel für S7-300/400

Das folgende Bild zeigt beispielhaft einen Auszug aus einer Exportdatei für S7-300/400.

	A	B	C	D
1	Location	Alarm name	"Alarm text" - Englisch (USA) / [en-US] / Event text	"Alarm text" - Deutsch (Deutschland) / [de-DE]
2	FB1	BoilerAlarm	Boiler temperature is @1R%6.2f@!	Heizertemperatur ist @1R%6.2f@!
3	FB1	MotorAlarm	Motor @1W%t#MotorNames@ is running at @2W%	Motor @1W%t#MotorNames@ läuft mit @2W%

Hinweis

Beachten Sie, dass im obenstehenden Beispiel die Schlüsselwörter (eingeschlossen in $$ bzw. @) ebenfalls nicht zu übersetzen sind.

Beispiel für S7-1500

Das folgende Bild zeigt beispielhaft einen Auszug aus einer Exportdatei für S7-1500.

	A	B	C	D	
1	Location	Alarm name	"Alarm text" - English (United States) / [en-US] /	"Meldetext" - German (Germany)	Fieldl
2	FB1	BoilerAlarm	Boiler temperature is <field ref="0" />!	Heizertemperatur ist <field ref="0	<ref i
3	FB1	MotorAlarm	Motor <field ref="0" /> is running at <field ref="1	Motor <field ref="0" /> läuft mit <	<ref i
4					

Übersetzen der Meldetexte in einem externen Tabelleneditor

Die Texte werden in der jeweiligen Spalte mit der für die zu übersetzende Sprache angegebenen Sprachkennung übersetzt. Stellen Sie beim Übersetzen der Texte sicher, dass die gleiche Reihenfolge der Begleitwerte in allen Sprachen eingehalten wird, da die Texte ansonsten nicht mehr importiert werden können!

Siehe auch

Meldetexte exportieren (Seite 7859)

Meldetexte importieren (Seite 7860)

Besonderheiten bei Begleitwerten (Seite 7841)

18.14.2 Meldetexte exportieren

Voraussetzung

Der Meldungseditor ist geöffnet.

Vorgehen

Um Meldetexte zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Register "Programmmeldungen" auf das Symbol "Exportieren" in der Funktionsleiste. Der Dialog "Meldetexte exportieren" wird geöffnet.
2. Wählen Sie den Pfad aus, in dem die Exportdatei abgelegt werden soll.
3. Wählen Sie die gewünschten Sprachen aus.
4. Wählen Sie gegebenenfalls aus, ob Ereignistexte, Infotexte und/oder Zusatztexte ebenfalls exportiert werden sollen.
5. Klicken Sie auf die Schaltfläche "OK".

Hinweis

- Im Export-Dialog werden alle im TIA Portal ausgewählten Projektsprachen angezeigt. Weitere Projektsprachen können Sie im Projektspracheneditor hinzufügen, damit sie in dieser Auswahl erscheinen. Standardmäßig sind alle verfügbaren Projektsprachen ausgewählt; Sprachen, die für den aktuellen Übersetzungsprozess keine Rolle spielen, können Sie wieder abwählen.
- Exportieren Sie die Daten nur in einen Bereich, der mit entsprechenden Zugriffsmechanismen geschützt ist.

Siehe auch

Grundlagen zum Exportieren und Importieren von Meldetexten (Seite 7858)

18.14.3 Meldetexte importieren

Voraussetzung

Der Meldungseditor ist geöffnet.

Vorgehen

Um Meldetexte zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Register "Programmmeldungen" auf das Symbol "Importieren" in der Funktionsleiste. Der Dialog "Meldetexte importieren" wird geöffnet.
2. Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist und klicken Sie auf die Schaltfläche "Importieren".
3. Wählen Sie im nachfolgenden Dialog die gewünschte(n) Sprache(n) aus.
4. Klicken Sie auf die Schaltfläche "OK".

Hinweis

- Sie müssen beim Importieren nicht dieselben Einstellungen verwenden wie beim Exportieren.
- Importieren Sie nur Dateien aus vertrauenswürdigen Quellen.

Siehe auch

Grundlagen zum Exportieren und Importieren von Meldetexten (Seite 7858)

PLC-Programme übersetzen und laden 19

19.1 Grundlagen zum Übersetzen und Laden von PLC-Programmen

PLC-Programme übersetzen und laden

Damit das von Ihnen erstellte PLC-Programm im Automatisierungssystem ablauffähig wird, müssen Sie die Programmdaten, die Sie offline erzeugt haben, zunächst übersetzen und dann in ein Gerät laden. Sie können die Programmdaten in Geräte und auf Memory Cards laden.

Übersetzen des PLC-Programms

Beim Übersetzen werden die Bausteine immer vollständig übersetzt, um das Programm konsistent zu halten.

Weiter Informationen zum Übersetzen des PLC-Programms finden Sie hier: Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

Laden des PLC-Programms

Beim erstmaligen Laden werden die Programmdaten vollständig geladen. Bei weiteren Ladevorgängen werden nur noch Änderungen geladen.

Beim Laden in Gerät werden alle Änderungen an Bausteinen, die PLC-Datentypen und PLC-Variablen mitgeladen, um ein konsistentes Laden sicher zu stellen.

Hinweis

Laden in Gerät

- Wenn bei den nachfolgenden Beschreibungen zum Laden von "Bausteinen" gesprochen wird, ist damit das gesamte PLC-Programm inklusive der PLC-Datentypen und der PLC-Variablen gemeint!

Beim Ladevorgang werden alle Informationen, die zur Rekonstruktion des Programms notwendig sind, auch symbolische Informationen, wie z. B. Namen und Kommentare für Code- und Datenbausteine, in bis zu drei projektierten Projektsprachen mitgeladen. Wenn Sie die Projektsprache ändern, ist daher ein neuer Ladevorgang notwendig.

Die symbolischen Informationen werden nicht in den Arbeitsspeicher, sondern in den Ladespeicher geladen.

Nach dem Laden der Daten aus einem Gerät stehen Ihnen auch die symbolischen Informationen wieder in Ihrem Programm zur Verfügung, dadurch erhöht sich die Lesbarkeit Ihres Programmcodes. Beachten Sie jedoch, dass Sie durch das Laden in und vom Gerät nicht die Ablage der Daten in einem Offline-Projekt ersetzen können, da Sie durch dieses

Vorgehen z. B. Beobachtungstabellen oder die Mehrsprachigkeit von Projekten nicht abbilden können.

Die Daten aus know-how-geschützten Bausteinen können Sie nach dem Laden aus einem Gerät nur mit korrektem Passwort vollständig anzeigen lassen.

Ab dem TIA Portal V15 werden die PLC-Variablen beim Laden entsprechend den offline konfigurierten Variablentabellen abgelegt.

Weiter Informationen zum Laden des PLC-Programms finden Sie hier:

Einführung zum Laden von Bausteinen (Seite 7871) (S7-1200/1500)

Einführung zum Laden von Bausteinen (Seite 7897) (S7-300/400)

Einführung zum Laden von PLC-Variablen (Seite 7914) (S7-1200/1500)

Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

Einführung zum Laden von Bausteinen (Seite 7897)

Einführung zum Laden von Bausteinen (Seite 7871)

Einführung zum Laden von PLC-Variablen (Seite 7914)

19.2 Bausteine übersetzen

19.2.1 Grundlagen zum Übersetzen von Bausteinen

Einführung

Damit die CPU das Anwenderprogramm bearbeiten kann, muss das Programm zunächst übersetzt werden. Nach jeder Änderung müssen Sie Ihr Programm erneut übersetzen.

Bei der Übersetzung finden folgende Vorgänge statt:

- Das Anwenderprogramm wird nach Syntaxfehlern durchsucht.
- Nicht benötigte Anweisungen werden aus dem Anwenderprogramm entfernt.
- Alle Bausteinaufrufe innerhalb der übersetzten Bausteine werden überprüft. Bei Änderungen an den Schnittstellen von aufgerufenen Bausteinen werden im Register "Übersetzen" des Info-Fensters Fehler angezeigt. Diese Fehler müssen Sie zunächst beheben.
- Im Anwenderprogramm müssen die Nummern von Bausteinen eindeutig sein. Falls mehrere Bausteine die gleichen Nummern haben, erfolgt daher bei der Übersetzung eine automatische Umnummerierung der Bausteine mit Nummernkonflikten. In folgenden Fällen wird ein Baustein nicht umnummeriert:
  - Der Baustein wurde entweder einzeln oder als Teil einer Multiselektion für die Übersetzung selektiert.
  - In den Eigenschaften des Bausteins ist für die Nummernvergabe "manuell" eingestellt.

  Nummernkonflikte, die nicht durch automatisches Umnummerieren aufgelöst werden können, müssen Sie manuell korrigieren. Beachten Sie hierzu die Meldungen im Inspektorfenster.

  Hinweis

  Umnummerieren von know-how-geschützten Bausteinen

  Automatisches Umnummerieren und manuelles Umnummerieren ohne Passwort von know-how-geschützten Bausteinen ist nur für CPUs der Baureihe S7-1500 und S7-1200 (ab V4) möglich. Der Know-how-Schutz muss zudem mit einer TIA Portal-Version ab V13 SP1 gesetzt worden sein.

- Anschließend wird das Anwenderprogramm in einen von der CPU lesbaren Code übersetzt.

Übersetzungswege

Die Übersetzung können Sie in folgenden Fenstern oder Editoren starten:

- Übersetzen von Bausteinen in der Projektnavigation
  Dient dem Übersetzen einzelner Bausteine oder dem gleichzeitigen Übersetzen mehrerer oder aller Bausteine im Ordner "Programmbausteine".
- Übersetzen von Bausteinen im Programmiereditor
  Dient dem Übersetzen eines einzelnen geöffneten Bausteins.
- Übersetzen von Bausteinen in der Aufruf- oder Abhängigkeitsstruktur
  Dient dem Übersetzen einzelner Bausteine.
  Siehe auch: Aufrufstruktur (Seite 7969), Abhängigkeitsstruktur (Seite 7977)

Übersetzungsoptionen

Wenn Sie Bausteine über die Projektnavigation übersetzen, erhalten Sie weitere Optionen:

- Software (nur Änderungen)
  Es werden alle Programmänderungen der selektierten Bausteine übersetzt. Wenn Sie einen Bausteinordner selektiert haben, werden alle Programmänderungen an den im Ordner enthaltenen Bausteinen übersetzt.
- Software (Bausteine komplett übersetzen)
  Alle Bausteine werden komplett übersetzt. Dies empfiehlt sich bei der ersten Übersetzung und bei größeren Änderungen.
- Software (Speicherreserve zurücksetzen)
  Alle Variablen, die im Reservebereich der Schnittstellen der selektierten Bausteine deklariert sind, werden in den Standardbereich der Schnittstelle verschoben. Die Speicherreserve wird frei für weitere Schnittstellenerweiterungen.

  Hinweis

  Diese Option steht Ihnen nur bei CPUs der Baureihe S7-1500 und S7-1200 ab V4 zur Verfügung.

Konsistenzprüfung

Wenn Sie die Schnittstellen aufgerufener Bausteine oder verwendeter PLC-Datentypen ändern, kann es zu Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen oder zwischen den PLC-Datentypen und den globalen Datenbausteinen, die diese PLC-Datentypen verwenden, kommen.

Um solche Inkonsistenzen im Anwenderprogramm zu vermeiden, wird vom System vor jedem Übersetzungsvorgang eine automatische Konsistenzprüfung durchgeführt. Dabei werden die

Zeitstempel verglichen und abhängig von diesem Vergleichsergebnis wird der Übersetzungsvorgang durchgeführt oder abgebrochen:

- Das erfolgreiche Übersetzen des aufrufenden Bausteins ist nur möglich, wenn die Zeitstempel der Schnittstellen der aufgerufenen Bausteine älter sind als der Zeitstempel des aufrufenden Bausteins.
- Das erfolgreiche Übersetzen eines globalen Datenbausteins, der auf einem PLC-Datentyp basiert, ist nur möglich, wenn der Zeitstempel des globalen Datenbausteins neuer als der Zeitstempel des verwendeten PLC-Datentyps ist.
- Das erfolgreiche Übersetzen eines Instanz-Datenbausteins ist nur möglich, wenn die Zeitstempel für die Schnittstellen des Instanz-Datenbausteins und des zugeordneten Funktionsbausteins identisch sind.

Wenn der Übersetzungsvorgang abgebrochen wird, erhalten Sie eine Meldung im Inspektorfenster. Aktualisieren Sie dann zunächst die Bausteinaufrufe in den jeweiligen Bausteinen und die PLC-Datentypen in den globalen Datenbausteinen und starten Sie anschließend den Übersetzungsvorgang erneut. Die Konsistenzprüfung ermittelt auch know-how-geschützte Bausteine, für die der Übersetzungsvorgang nicht durchgeführt werden kann. Auch hierzu finden Sie die entsprechenden Meldungen im Inspektorfenster.

Wenn Sie gleich einen Ladevorgang statt eines Übersetzungsvorgangs starten, werden die ausgewählten Bausteine automatisch übersetzt und die Bausteinaufrufe und globalen Datenbausteine werden implizit aktualisiert. Dabei gibt es folgende Unterschiede bei den CPU-Familien:

- S7-1200/1500: Bei einem Ladevorgang werden alle betroffenen Bausteine mitgeladen, sodass keine Inkonsistenzen entstehen können.
- S7-300/400: Nur der ausgewählte Baustein wird geladen.

Siehe auch

Bausteine in der Projektnavigation übersetzen (Seite 7867)

Bausteine im Programmiereditor übersetzen (Seite 7869)

Übersetzungsfehler beheben (Seite 7870)

Zeitstempel von Bausteinen (Seite 7219)

Bausteinaufrufe in KOP aktualisieren (Seite 7450)

Bausteinaufrufe in FUP aktualisieren (Seite 7517)

Bausteinaufrufe in AWL aktualisieren (Seite 7586)

19.2.2 Bausteine in der Projektnavigation übersetzen

In der Projektnavigation können Sie einen, mehrere oder alle Bausteine übersetzen.

Für CPUs der Baureihen S7-1500 und S7-1200 V4 können Sie mit einem Übersetzungsvorgang zusätzlich das Speicherlayout von Bausteinen mit Speicherreserve

rücksetzen. Die Beschreibung der Speicherreserve finden Sie im Kapitel "Bausteine laden (S7-1200/1500) > Bausteinänderungen ohne Reinitialisierung laden".

Voraussetzung

Die Projektnavigation ist eingeblendet.

Einen oder mehrere Bausteine in der Projektnavigation übersetzen

Um einen oder mehrere Bausteine in der Projektnavigation zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Selektieren Sie die Bausteine, die Sie übersetzen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".

Alle Bausteine in der Projektnavigation übersetzen

Um alle Bausteine des Ordners "Programmbausteine" in der Projektnavigation zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Sie können für die Übersetzung zwei verschiedene Optionen wählen:
   - Wenn Sie nur die Änderungen seit der letzten Übersetzung übersetzen möchten, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".
   - Wenn Sie die Bausteine vollständig übersetzen möchten, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

Speicherlayout rücksetzen (S7-1500/S7-1200 V4)

Um das Speicherlayout von Bausteinen rückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gesamten Ordner "Programmbausteine" oder einzelne Bausteine innerhalb des Ordners.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Speicherreserve zurücksetzen)".

Ergebnis

Wenn die Konsistenzprüfung erfolgreich war, wird der Code für die Bausteine generiert. Vom System erzeugte und nicht mehr benötigte Instanz-Datenbausteine werden gelöscht.

Ob die Übersetzung erfolgreich war, können Sie an der Meldung im Inspektorfenster unter "Info > Übersetzen" erkennen.

Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

Bausteine im Programmiereditor übersetzen (Seite 7869)

Übersetzungsfehler beheben (Seite 7870)

Syntaxfehler im Programm finden (Seite 7154)

19.2.3 Bausteine im Programmiereditor übersetzen

Hinweis

Beachten Sie, dass der Baustein auch dann neu übersetzt wird, wenn Sie keine Änderungen vorgenommen haben und dass dadurch die Zeitstempel des Bausteins verändert werden.

Voraussetzung

Der zu übersetzende Baustein ist geöffnet.

Vorgehen

Um einen Baustein im Programmiereditor zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Anweisungsfenster des Programmiereditors.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen".

Ergebnis

Der Code für den Baustein wird generiert. Vom System erzeugte und nicht mehr benötigte Instanz-Datenbausteine werden gelöscht.

Ob die Übersetzung erfolgreich war, können Sie an der Meldung im Inspektorfenster unter "Info > Übersetzen" erkennen.

Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

Bausteine in der Projektnavigation übersetzen (Seite 7867)

Übersetzungsfehler beheben (Seite 7870)

19.2.4 Übersetzungsfehler beheben

Im Inspektorfenster unter "Info > Übersetzen" erkennen Sie, ob eine Übersetzung erfolgreich war oder ob Fehler im Programm erkannt worden sind. Wenn Fehler auftreten, müssen Sie diese beheben und anschließend die Übersetzung erneut starten.

Vorgehen

Um Fehler nach einer Übersetzung zu beheben, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Fehlerliste im Inspektorfenster unter "Info > Übersetzen".
2. Wenn vorhanden, klicken Sie auf das blaue Fragezeichen neben dem Fehlertext, um Hinweise zur Fehlerbehebung zu erhalten.
3. Doppelklicken Sie auf den Fehler, den Sie korrigieren möchten.
   Der entsprechende Fehler wird markiert.
4. Korrigieren Sie den Fehler.
5. Starten Sie die Übersetzung erneut.

Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 7865)

Bausteine im Programmiereditor übersetzen (Seite 7869)

Bausteine in der Projektnavigation übersetzen (Seite 7867)

19.3 Bausteine laden für S7-1200/1500

19.3.1 Einführung zum Laden von Bausteinen

Bausteine in Gerät laden

Damit die CPU das Anwenderprogramm bearbeiten kann, müssen die Programmdaten zunächst übersetzt und anschließend in das Gerät geladen werden.

Sie können die Programmdaten in Geräte und auf Memory Cards laden.

Das Laden in und vom Gerät kann über verschiedene Wege gestartet werden:

- über die Befehle im Menü "Online"
- über das Kontextmenü
- über die Schaltfläche "Laden" in der Toolbar

Das Laden kann in der Projektnavigation von folgenden Objekten aus gestartet werden:

- von Geräten
- vom Ordner "Programmbausteine"
- von einzelnen Bausteinen
- vom Ordner "PLC-Datentypen"
- vom Ordner "PLC-Variablen"

Abhängig vom selektierten Objekt und dem jeweils ausgewählten Befehl können Sie folgende Komponenten über das Menü "Online" laden:

- Laden in Gerät
  Sowohl die Hardware-Konfiguration als auch die Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen.
- Erweitertes Laden in Gerät
  Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen. Beim erweiterten Laden können Sie zusätzlich noch ein Zielgerät und die Einstellungen für ein Zielgerät auswählen.
- PLC-Programm ins Gerät laden und zurücksetzen
  Es wird das PLC-Programm inklusive aller Bausteine, PLC-Datentypen und PLC-Variablen in das Ziel geladen und alle Werte auf ihre Startwerte zurückgesetzt. Beachten Sie, dass dies auch für die remanenten Werte der Fall ist. Die CPU geht dabei in "STOP".
- Anwenderprogramm auf Memory Card laden
  Das gesamte Programm wird auf eine gesteckte Memory Card geladen.

Abhängig vom selektierten Objekt und dem jeweils ausgewählten Befehl können Sie folgende Komponenten über das Kontextmenü laden:

- Hardware und Software (nur Änderungen)
  Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen.
- Hardwarekonfiguration
  Es wird nur die Hardware-Konfiguration in das Ziel geladen.
- Software (nur Änderungen)
  Es werden nur die Objekte in das Ziel geladen, die online und offline unterschiedlich sind.
- Software (komplett laden)
  Die Software wird komplett in das Ziel geladen.

Hinweis

Keine gültige Hardwarekonfiguration vorhanden

Falls beim Laden von Software keine gültige Hardwarekonfiguration auf der CPU gefunden wird, so wird die vorhandene Hardware beim "Laden in Gerät" mitgeladen.

Beachten Sie bitte, dass in diesem Fall beim Laden Hardwarekomponenten geladen werden, obwohl Sie den Befehl zum Laden von Software ausgeführt haben!

Hinweis

Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, werden bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine übersetzt und geladen.

Hinweis

S7-1500

Bei CPUs der Baureihe S7-1500 befindet sich der Ladespeicher auf der SIMATIC Memory Card. Daher ist für den Betrieb der CPU eine gesteckte SIMATIC Memory Card zwingend erforderlich.

Bausteine von Gerät laden

Sie können die Bausteine eines Gerätes in Ihr Projekt laden. Dies ist z. B. notwendig, um Bausteine zu bearbeiten, die nur in diesem Gerät vorhanden sind. Sie haben die Möglichkeit entweder alle vorhandenen Bausteine (Organisationsbausteine, Funktionsbausteine, Funktionen, Datenbausteine) und globalen PLC-Variablen oder einzelne Bausteine in das Projekt geladen.

Bausteine auf oder von einer Memory Card laden

Memory Cards sind steckbare Speicherkarten, die z. B. bei einer CPU der Baureihe S7-1200 dazu dienen, den Ladespeicher eines Geräts zu ersetzen. Bei CPUs der Baureihe S7-1500 enthalten sie den Ladespeicher. Für Geräte der S7-1200- und S7-1500-Baureihe können Sie dafür ausschließlich die SIMATIC Memory Card von Siemens verwenden.

Damit Sie eine Memory Card als Ladespeicher nutzen können, müssen Sie das Anwenderprogramm oder einzelne Bausteine auf eine Memory Card laden. Umgekehrt können Sie auch Bausteine von einer Memory Card wieder ins Projekt laden.

Hinweis

S7-1200

Beachten Sie beim Laden auf oder von einer Memory Card folgende Hinweise:

- Wenn die CPU noch kein Programm enthält und Sie eine leere Memory Card in die CPU stecken, wird das Programm aus dem PG/PC auf die Memory Card und nicht in die CPU geladen.
- Wenn Sie vor dem Anlauf der CPU eine leere Memory Card in die CPU stecken, wird das Programm, das auf der CPU vorhanden ist, automatisch auf die Memory Card übertragen. Das Programm auf der CPU wird anschließend gelöscht.
- Wenn Sie vor dem Anlauf der CPU eine Memory Card mit einem Programm in die CPU stecken, die ebenfalls ein Programm enthält, wird das Programm auf der Memory Card und nicht das Programm der CPU ausgeführt. Das Programm auf der CPU wird gelöscht.

GRAPH-Funktionsbausteine laden

Wenn Sie einen GRAPH-Funktionsbaustein zusammen mit seinem Instanz-Datenbaustein laden, beginnt die Bearbeitung der Ablaufkette wieder mit dem Initialschritt. Dadurch können Probleme bei der Synchronisation der Ablaufkette mit dem Prozess auftreten. Sie können diese Probleme vermeiden, indem Sie die Ablaufkette vor dem Laden ausschalten.

Bausteinänderungen ohne Reinitialisierung laden

Häufig ergibt sich die Notwendigkeit, ein PLC-Programm, das bereits in Betrieb genommen wurde und fehlerfrei auf einer Anlage läuft, nachträglich zu ändern oder zu erweitern. Dabei soll der laufende Betrieb möglichst nicht beeinträchtigt werden.

S7-1500 bietet deshalb die Möglichkeit, die Schnittstellen von Funktions- oder Datenbausteinen im laufenden Betrieb zu erweitern und die geänderten Bausteine zu laden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen. Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

Auswirkungen eines Ladevorgangs auf die Variablenwerte eines Datenbausteins

Wenn Datenbausteine im Betriebszustand STOP in ein Gerät geladen werden, wirkt sich der nächste Übergang des Geräts nach RUN auf die Aktualwerte der Variablen folgendermaßen aus:

- Variablen, die nicht als remanent gekennzeichnet sind, erhalten ihre definierten Startwerte.
- Remanente Variablen der S7-1200 behalten ihre Werte nur, wenn folgende Bedingungen erfüllt sind:
  - Sie haben den Datenbaustein über "Laden in Gerät > Software (nur Änderungen)" geladen.
  - Sie haben die Struktur des Datenbausteins nicht geändert.

  Ansonsten erhalten auch die remanenten Variablen ihre definierten Startwerte.
- Remanente Variablen der S7-1500 behalten ihre Werte nur, wenn folgende Bedingungen erfüllt sind:
  - Sie haben den Datenbaustein über "Laden in Gerät > Software (nur Änderungen)" geladen.
  - Sie haben die Struktur des Datenbausteins nicht oder innerhalb der Speicherreserve geändert.

  Ansonsten erhalten auch die remanenten Variablen ihre definierten Startwerte.

Bausteine laden mit Synchronisierung

Im Rahmen des Team Engineering ist es möglich, dass mehrere Anwender mit mehreren Engineering Systemen parallel an einem Projekt arbeiten und auf eine CPU S7-1500 zugreifen. Um die Konsistenz innerhalb des gemeinsamen bearbeiteten Projektes sicher zu stellen, ist es notwendig, die jeweils geänderten Daten vor dem Laden zu synchronisieren, damit es nicht zu ungewollten Überschreibungen kommt.

Wenn also beim Laden Unterschiede zwischen der Online- und der Offline-Datenhaltung innerhalb des gemeinsamen Projektes festgestellt werden, die von einem anderen Engineering System verursacht wurden, wird beim Ladevorgang automatisch eine Synchronisierung der zu ladenden Daten angeboten.

In diesem Fall werden im Dialog "Synchronisierung" die zu synchronisierenden Daten mit dem aktuellen Status (Online-Offline-Vergleich) und den möglichen Aktionen angezeigt.

Folgende Optionen für eine Synchronisierung stehen zur Verfügung:

Anwendungsfall	Empfehlung	Synchronisierung
Ein oder mehrere Bausteine sind auf der CPU (online) neuer als im Engineering System (offline).	Diese Bausteine sollten vor dem Laden zuerst von der CPU in das Engineering System geladen werden.	Automatische Synchronisierung ist möglich: Die Bausteine im Engineering System werden vor dem Laden aktualisiert.
Ein oder mehrere Bausteine sind neu erstellt und nur in der CPU (online) vorhanden.	Diese Bausteine sollten vor dem Laden zuerst von der CPU in das Engineering System geladen werden.	Automatische Synchronisierung ist möglich: Die neuen Bausteine werden vor dem Laden im Engineering System hinzugefügt.

Anwendungsfall	Empfehlung	Synchronisierung
Ein oder mehrere Bausteine auf der CPU wurden gelöscht.	Diese Bausteine sollten vor dem Laden im Engineering System ebenfalls gelöscht werden.	Automatische Synchronisierung ist nicht möglich. Die auf der CPU gelöschten Bausteine sollten im Offline-Projekt im Engineering System manuell gelöscht werden.
Ein oder mehrere Bausteine sind auf der CPU und im Engineering System unterschiedlich. Dieser Fall tritt dann auf, wenn ein anderer Anwender Bausteine verändert und bereits in die CPU geladen hat, an denen auch Sie Korrekturen vorgenommen haben.	Diese Bausteine mit konkurrierenden Änderungen müssen manuell angepasst werden. Entscheiden Sie in diesem Fall, welche Änderungen Sie übernehmen wollen. Sollen die Bausteine auf der CPU erhalten bleiben, so sollten Sie diese Bausteine vor dem Laden von der CPU in Ihr Engineering System übernehmen. Sollen die von Ihnen geänderten Bausteine übernommen werden, können Sie mit dem Laden ohne Synchronisierung fortfahren.	Automatische Synchronisierung ist nicht möglich: Die betroffenen Bausteine auf der CPU oder im Engineering System müssen manuell angepasst werden. Eine der vorhandenen Baustein-Versionen (online oder offline) wird dabei überschrieben.
Es werden Unterschiede in der Hardware-Konfiguration auf der CPU (online) und im Engineering System (offline) festgestellt.	Unterschiede in der Hardware-Konfiguration müssen manuell angepasst werden. Entscheiden Sie in diesem Fall, welche Hardware-Konfiguration Sie übernehmen wollen. Soll die bereits auf der CPU vorhandene Hardware-Konfiguration erhalten bleiben, so sollten Sie diese vor dem Laden in Ihr Engineering System übernehmen. Soll die von Ihnen geänderte Hardware-Konfiguration übernommen werden, können Sie mit dem Laden ohne Synchronisierung fortfahren.	Automatische Synchronisierung ist nicht möglich: Die Hardware-Konfiguration muss manuell angepasst werden. Eine der vorhandenen Hardware-Konfigurationen (online oder offline) wird dabei überschrieben.

Über den Befehl "Laden in Gerät erzwingen" können Sie Bausteine auch ohne Synchronisierung laden, falls gewünscht.

I&M-Daten in PROFINET-IO-Geräte und ihre Module laden

Beim Laden in ein Gerät können Sie zusätzlich zu den oben genannten Komponenten I&M-Daten in PROFINET-IO-Geräte und ihre Module laden.

19.3.2 Bausteine im Betriebszustand "RUN" in Gerät laden

Grundlagen zum Laden von Bausteinen im Betriebszustand "RUN"

Wenn Sie geänderte Bausteine in ein Gerät laden, dann ist es nicht immer notwendig, dass das Gerät in den Betriebszustand "STOP" wechselt. Bei einem Ladevorgang prüft das Engineering System daher, ob das Gerät vor dem Laden gestoppt werden muss. Das Ergebnis dieser Prüfung wird im Dialog "Vorschau Laden" angezeigt.

Wenn ein Wechsel in den Betriebszustand "STOP" notwendig ist, können Sie den Ladevorgang erst nach Stoppen der CPU fortsetzen.

Wenn die Voraussetzungen erfüllt sind, können Sie ein geändertes Programm oder Programmteile auch im Betriebszustand "RUN" in eine CPU laden.

Bei sehr komplexen Programmen werden möglicherweise Mengengerüste überschritten, die ein Laden im "RUN" verhindern. In diesem Fall müssen Sie zunächst die Randbedingungen für ein Laden im "RUN" herstellen.

Tipps:

- Verwenden Sie eine Memory Card mit ausreichender Kapazität.
- Wählen Sie eine CPU mit ausreichendem Arbeitsspeicher.
- Reduzieren Sie gegebenenfalls die Anzahl der zu ladenden Objekte (Bausteine, Konstanten, Variablen, Datentypen).
  Wenn Sie nicht alle Objekte auf einmal laden können, gehen Sie in mehreren Schritten vor und laden Sie jeweils eine geringere Anzahl von Objekten.

Hinweis

Durch einen Ladevorgang im Betriebszustand "RUN" werden keine Aktualparameter überschrieben. Änderungen an den Aktualparametern erfolgen erst beim nächsten Betriebszustandswechsel von "STOP" nach "RUN".

Laden von Änderungen im "RUN"

Die folgende Tabelle zeigt, welche Programm- oder Konfigurationsänderungen auch im Betriebszustand "RUN" geladen werden können, sortiert nach CPU-Familien und unter Berücksichtigung der Firmware-Versionen der CPUs.

Erläuterungen zur Tabelle:

- "RUN": Änderung ist nicht nur im Betriebszustand STOP, sondern auch im Betriebszustand "RUN" in die CPU ladbar.
- "RUN (< 57)": Bis zu 56 neue oder geänderte Objekte/Bausteine kann die CPU in einem Programmzyklus integrieren. Wenn Sie mehr Objekte/Bausteine laden, werden sie in mehreren aufeinander folgenden Programmzyklen integriert. Wenn Sie alle Objekte/Bausteine konsistent laden wollen, dann müssen Sie die CPU in den Betriebszustand "STOP" setzen. Bei S7-300-CPUs mit Parametriermöglichkeit "Prozessbetrieb/Testbetrieb" variiert diese Anzahl je nach Einstellung.
- "RUN (Init)": Änderung ist im Betriebszustand "RUN" ladbar; geladene Datenbausteine werden reinitialisiert.
- "STOP": Änderung ist nur im Betriebszustand "STOP" ladbar.
- "STOP (Reset)": Änderung ist nur im Betriebszustand "STOP" ladbar; alle Daten einschließlich der remanenten Daten werden zurückgesetzt.

	S7-300	S7-400	S7-1200 ab V4.0	S7-1500
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**			
Geänderte Eigenschaften von HW-Komponenten. Dazu gehören auch Änderungen bei den Kommentaren in der HW-Konfiguration.	STOP	STOP	STOP	STOP
Hinzugefügte HW-Komponenten	STOP	STOP	STOP	STOP
Neue/geänderte Textlisten (Meldungen)	**RUN**	**RUN**	STOP	**RUN (ab V1.1)**
Geänderte Kommentare (neu, geändert, gelöscht) mit Ausnahme der Kommentare in der HW-Konfiguration	-	-	**RUN**	**RUN**
Anzahl Bausteine synchron laden	**RUN (<17)**	**RUN (<57)**	**RUN (<21)**	**RUN (alle)**
PLC-Programm ins Gerät laden und zurücksetzen	STOP (Reset)	STOP (Reset)	STOP (Reset)	STOP (Reset)
Neuer OB	**RUN**	**RUN**	STOP	**RUN**
Geänderter OB: Codeänderungen	**RUN**	**RUN**	**RUN**	**RUN**
OB mit geänderten Eigenschaften (z. B. Zykluszeit-Änderung)	**RUN**	**RUN**	STOP	**RUN**
Gelöschter OB	**RUN**	**RUN**	STOP	**RUN**
Neuer FB/FC/DB/Anwenderdatentyp (UDT)	**RUN**	**RUN**	**RUN**	**RUN**
Gelöschter FB/FC/DB/Anwenderdatentyp (UDT)	**RUN**	**RUN**	**RUN**	**RUN**
Geänderter FB/FC: Codeänderung	**RUN**	**RUN**	**RUN**	**RUN**
Geänderter FB/FC: Schnittstellenänderung*	STOP	STOP	**RUN**	**RUN**
Geänderter DB: Eigenschaft geändert (Attribut "Nur im Ladespeicher ablegen" geändert)	STOP	STOP	**RUN (Init)**	**RUN (Init)**
Geänderter DB (Speicherreserve nicht aktiviert): Name/Typ von Variablen geändert, Variablen hinzugefügt oder gelöscht **	**RUN (Init)**	**RUN (Init)**	**RUN (Init)**	**RUN (Init)**
Geänderter DB (Speicherreserve aktiviert): Neue Variablen hinzugefügt**	-	-	**RUN**	**RUN**
Geänderter Anwenderdatentyp (UDT)	STOP	STOP	**RUN (Init)**	**RUN (Init)**
Neue PLC-Variablen hinzugefügt (Zeiten, Zähler, Merker)	**RUN**	**RUN**	**RUN**	**RUN**
Geänderte Remanenzeinstellungen (Zeiten, Zähler, Merker, DB-Bereich)	STOP	STOP	STOP	STOP
Motion Control Technologieobjekte: Änderungen MC-Servo Zeittakt, Änderung von freilaufend auf zyklisch (und umgekehrt). Änderungen an der HW-Schnittstelle des TOs	-	-	-	STOP

* Falls die Schnittstellenänderung zu strukturellen Änderungen am Instanz-DB führt, siehe "Geänderter DB..."

** Zur Auswirkung des Ladens von Datenbaustein-Änderungen auf den Datenbaustein-Inhalt siehe Abschnitt "Laden von Datenbausteinen in die CPU"

Laden von Änderungen im "RUN" mit früheren CPU-Firmware-Versionen

Die folgende Tabelle zeigt, welche Änderungen Sie im Betriebszustand "RUN" laden können, und zwar für frühere CPU-Firmware-Versionen.

	S7-1200 V1.0 - 2.1	S7-1200 V2.2 - V3.0
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**	
Geänderte Eigenschaften von HW-Komponenten. Dazu gehören auch Änderungen bei den Kommentaren in der HW-Konfiguration.	STOP	STOP
Hinzugefügte HW-Komponenten	STOP	STOP
Neue/geänderte Textlisten (Meldungen)	STOP	STOP
Geänderte Kommentare (neu, geändert, gelöscht) mit Ausnahme der Kommentare in der HW-Konfiguration.	STOP	**RUN**
Anzahl Bausteine synchron laden	STOP	**RUN (<11)**
PLC-Programm ins Gerät laden und zurücksetzen	STOP (Reset)	STOP (Reset)
Neuer OB	STOP	STOP
Geänderter OB: Codeänderungen	STOP	**RUN**
OB mit geänderten Eigenschaften (z. B. Zykluszeit-Änderung)	STOP	STOP
Gelöschter OB	STOP	STOP
Neuer FB/FC/DB/Anwenderdatentyp (UDT)	STOP	**RUN**
Gelöschter FB/FC/DB/Anwenderdatentyp (UDT)	STOP	**RUN**
Geänderter FB/FC: Codeänderung	STOP	**RUN**
Geänderter FB/FC: Schnittstellenänderung	STOP	STOP
Geänderter DB: Eigenschaft geändert (Attribut "Nur im Ladespeicher ablegen" geändert)	STOP	STOP
Geänderter DB (Speicherreserve nicht aktiviert): Name/Typ von Variablen geändert, Variablen hinzugefügt oder gelöscht	STOP	STOP
Geänderter Anwenderdatentyp (UDT)	STOP	STOP
Neue PLC-Variablen hinzugefügt (Zeiten, Zähler, Merker)	STOP	STOP
Geänderte Remanenzeinstellungen (Zeiten, Zähler, Merker, DB-Bereich)	STOP	STOP

Laden von Datenbausteinen in die CPU

Je nach den vorliegenden Randbedingungen hat das Laden von neuen oder geänderten Datenbausteinen Einfluss auf die Aktualwerte im Datenbaustein:

Neue Datenbausteine laden	Aktualwerte in den neuen Datenbausteinen sind auf Startwerte gesetzt.
Strukturell geänderte Datenbausteine laden (Speicherreserve nicht aktiviert)	Aktualwerte hinzugefügter Variablen in den strukturell geänderten Datenbausteinen sind auf Startwerte gesetzt.
Strukturell geänderte Datenbausteine laden (Speicherreserve aktiviert)	Folgende Aktualwerte bleiben erhalten: • Aktualwerte von Variablen außerhalb der Speicherreserve • Aktualwerte von nicht geänderten Variablen innerhalb der Speicherreserve. Aktualwerte hinzugefügter Variablen innerhalb der Speicherreserve sind auf Startwerte gesetzt.
Einfach geänderten Datenbaustein laden (keine Strukturänderung)	Aktualwerte bleiben erhalten.

Weitere Informationen

Weitere Informationen zum Laden von Bausteinerweiterungen ohne Reinitialisierung und zum Laden von geänderten Werten bei Datenbausteinen finden Sie unter "Siehe auch".

Siehe auch

Bausteine vom Programmiereditor in Gerät laden (Seite 7879)

Bausteine aus der Projektnavigation in Gerät laden (Seite 7881)

19.3.3 Bausteine vom Programmiereditor in Gerät laden

Voraussetzung

Der zu ladende Baustein ist geöffnet.

Vorgehen

Um einen Baustein vom Programmiereditor aus in das Gerät zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Anweisungsfenster des Programmiereditors.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden".
   - Sie können Ihre bevorzugt verwendeten Verbindungsparameter unter "Extras > Einstellungen > Online & Diagnose" als Voreinstellung speichern. Beim erstmaligen Laden mit neuen Verbindungsparametern erhalten Sie automatisch eine Abfrage, ob diese als Voreinstellung gespeichert werden sollen. Klicken Sie auf "Ja", um die aktuellen Verbindungsparameter als Voreinstellung zu speichern.
   - Sie haben die Möglichkeit, sich alle kompatiblen Teilnehmer anzeigen zu lassen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken. Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

Hinweis

Aktionen

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

Hinweis

Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, können Sie bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine laden. Wählen Sie dazu die Aktion "Konsistentes Laden" aus.

---

4. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzten", damit der Baustein geladen wird. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
5. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Baugruppe starten".
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Der Code für den Baustein wird in das Gerät geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Vorhandene CPU-Datenbausteine bleiben dabei jedoch erhalten. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

Siehe auch

Bausteine aus der Projektnavigation in Gerät laden (Seite 7881)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 7875)

19.3.4 Bausteine aus der Projektnavigation in Gerät laden

In der Projektnavigation können Sie einen, mehrere oder alle Bausteine in ein Gerät laden.

Einen oder mehrere Bausteine aus der Projektnavigation in Gerät laden

Um einen oder mehrere Bausteine aus der Projektnavigation heraus in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Selektieren Sie die Bausteine, die Sie laden möchten.

3. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen)".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden".
   - Sie können Ihre bevorzugt verwendeten Verbindungsparameter unter "Extras > Einstellungen > Online & Diagnose" als Voreinstellung speichern. Beim erstmaligen Laden mit neuen Verbindungsparametern erhalten Sie ebenfalls eine diesbezügliche Abfrage. Beantworten Sie diese mit "Ja", um die aktuellen Verbindungsparameter als Voreinstellung zu speichern.
   - Sie haben die Möglichkeit, sich alle kompatiblen Teilnehmer anzeigen zu lassen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken. Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzten", damit der Baustein geladen wird. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
7. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Alle Bausteine aus der Projektnavigation in Gerät laden

Um alle Bausteine des Ordners "Programmbausteine" aus der Projektnavigation heraus in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Wählen Sie im Kontextmenü das Untermenü "Laden in Gerät".
3. Wenn Sie nur die Änderungen seit dem letzten Ladevorgang laden möchten, wählen Sie die Option "Software (nur Änderungen)". Wenn alle Bausteine komplett geladen und alle Werte auf ihre Startwerte zurückgesetzt werden sollen, wählen Sie die Option "PLC-Programm ins Gerät laden und zurücksetzen".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie haben die Möglichkeit, sich alle kompatiblen Teilnehmer anzeigen zu lassen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken. Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
5. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzten", damit der Baustein geladen wird. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Baugruppe starten".
7. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Der Code für die Bausteine wird in das Gerät geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

Siehe auch

Bausteine vom Programmiereditor in Gerät laden (Seite 7879)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 7875)

19.3.5 Bausteine von Gerät laden

Sie können entweder alle oder nur einzelne Bausteine von einem Gerät in Ihr Projekt laden.

Hinweis

Beachten Sie folgende Hinweise:

- Beachten Sie, dass beim Laden von einzelnen Bausteinen keine Variablen und keine anderen notwendigen Bausteine mitgeladen werden, auf die Sie evtl. verweisen. Während des Ladevorgangs werden Referenzen auf Variablen und Bausteine anhand der Namen möglichst wieder zugeordnet. Prüfen Sie nach dem Ladevorgang, ob diese Zuordnungen korrekt sind.
- Beim Laden von Gerät in ein leeres Projekt werden evtl. vorhandene Ordnerstrukturen für Bausteine und für Objekte aus Bibliotheken nicht geladen.
- S7-1500: Beim Laden von GRAPH-Funktionsbausteinen von einem Gerät in Ihr Projekt werden die schrittspezifischen Meldungstexte für die Interlock- und Supervisionsmeldungen nicht mitgeladen.
- S7-1500: Die Startwerte, die Sie mithilfe der Anweisung "WRIT_DBL: In Datenbaustein im Ladespeicher schreiben" verändert haben, gehen beim Ausführen der Aktion "Laden von Gerät" wieder verloren.

Voraussetzung

Die Online- und Offline-Version der zu ladenden Bausteine ist unterschiedlich oder die Bausteine existieren nur online.

Alle Bausteine von Gerät laden

Um alle Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.
   Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation den Ordner des Geräts, von dem Sie Bausteine laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Einzelne Bausteine von Gerät laden

Um einzelne Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.
   Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation die Bausteine, die Sie vom Gerät laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Ergebnis

Die Bausteine werden von dem Gerät in das Projekt geladen. Sie können sie normal bearbeiten und erneut übersetzen und zurück in das Gerät laden.

19.3.6 Bausteine auf eine Memory Card laden

Voraussetzung

- Die Memory Card ist als Programmkarte gekennzeichnet.
- Der Ordner "Programmbausteine" der Memory Card ist geöffnet.

Vorgehen

Um Bausteine auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine" des Geräts.
2. Selektieren Sie die Bausteine, die Sie auf die Memory Card laden möchten.
3. Ziehen Sie die Bausteine in der Projektnavigation auf den Ordner "Programmbausteine" der Memory Card. Alternativ können Sie die Bausteine kopieren und in die Memory Card einfügen.
   Falls notwendig, werden die Bausteine übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
5. Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzen", damit der Baustein geladen wird. Der Dialog "Ergebnisse Laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
7. Klicken Sie auf die Schaltfläche "Fertig stellen".

Oder:

1. Selektieren Sie in der Projektnavigation die Bausteine, die Sie laden möchten.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben".
   Der Dialog "Memory Card auswählen" wird geöffnet.
3. Wählen Sie eine Memory Card aus, die mit der CPU kompatibel ist.
   Im unteren Bereich des Dialogs wird eine Schaltfläche mit einem grünen Häkchen aktiv.
4. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.

5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzen", damit der Baustein geladen wird. Der Dialog "Ergebnisse Laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
7. Klicken Sie auf die Schaltfläche "Fertig stellen".

Ergebnis

Die Bausteine werden auf die Memory Card geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese ebenfalls auf die Memory Card geladen. Bausteine, die nur auf der Memory Card existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine auf der Memory Card werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

Siehe auch

Bausteine von einer Memory Card laden (Seite 7887)

19.3.7 Bausteine von einer Memory Card laden

Sie können immer nur alle Bausteine von einer Memory Card in Ihr Projekt laden.

Voraussetzung

Die Memory Card wird angezeigt.

Siehe auch: Auf Memory Cards zugreifen

Vorgehen

Um Bausteine von einer Memory Card in das Projekt zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf den Ordner des Geräts im Projekt. Alternativ können Sie die Memory Card kopieren und in das Gerät einfügen.
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden von Gerät".

Siehe auch

Bausteine auf eine Memory Card laden (Seite 7886)

19.3.8 Ablaufkette vor dem Laden eines GRAPH-DB ausschalten

Sie können das Ausschalten der Ablaufkette vor dem Laden eines Instanz-Datenbausteins entweder global festlegen oder während des Ladevorgangs.

Ablaufkette global ausschalten

Um die Ablaufkette global für jeden Ladevorgang eines Instanz-Datenbausteins auszuschalten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".
   Bei zukünftigen Ladevorgängen wird die Ablaufkette vor dem Laden des Instanz-Datenbausteins ausgeschaltet.

Ablaufkette während des Ladevorgangs ausschalten

Um die Ablaufkette während des Ladevorgangs auszuschalten, gehen Sie folgendermaßen vor:

1. Laden Sie den GRAPH-Funktionsbaustein in das Gerät.
   Während des Ladevorgangs erscheint der Dialog "Vorschau laden". Hier werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen. Wenn der Instanz-Datenbaustein zusammen mit dem GRAPH-Funktionsbaustein geladen werden muss, wird Ihnen im Dialog "Vorschau laden" die Aktion "Kette vor Laden des DB ausschalten" angeboten.
2. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".

19.3.9 Bausteinerweiterungen ohne Reinitialisierung laden

19.3.9.1 Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung

Beschreibung

Um PLC-Programme, die bereits in Betrieb genommen wurden und fehlerfrei auf einer Anlage laufen, nachträglich zu ändern, bieten CPUs der Baureihe S7-1200 ab V4 und S7-1500 die Möglichkeit, die Schnittstellen von Funktions- oder Datenbausteinen im laufenden Betrieb zu erweitern.

Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

Funktionsweise

Grundsätzlich enthält jeder Funktions- oder Datenbaustein per Voreinstellung eine Speicherreserve, die Sie für nachträgliche Schnittstellenänderungen verwenden können. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass Sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die Werte bereits geladener Variablen nicht beeinflusst.

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie zusätzlich die Möglichkeit, das Speicherlayout einzelner oder mehrerer Bausteine in einem Schritt zurückzusetzen. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

Voraussetzungen

Die Funktion "Laden ohne Reinitialisierung" steht zur Verfügung, wenn folgende Voraussetzungen erfüllt sind:

- Das Projekt liegt im Format "TIA Portal V12" oder einer höheren Version vor.
- Sie arbeiten mit einer CPU der Baureihe S7-1200 ab V4 oder S7-1500.
- Die Bausteine wurden in KOP, FUP, AWL oder SCL erstellt.
- Die Bausteine wurden vom Anwender erstellt, d. h. es sind keine mitgelieferten Bausteine.
- Es handelt sich um Bausteine mit "optimiertem Zugriff".

Folgende Bausteine bieten nicht die Möglichkeit, Bausteinerweiterungen ohne Reinitialisierung zu laden:

- GRAPH-Bausteine
- ARRAY-Datenbausteine
- ProDiag-Bausteine

Hinweis

Einschränkung bei Verwendung der Speicherreserve

Es ist nicht möglich ist, den Inhalt eines globalen Datenbausteins einem strukturell gleichen Datenbaustein z. B. über eine Move-Box zuzuweisen, wenn einer der beiden DBs eine Speicherreserve enthält.

Grundsätzliche Schritte

Um die Schnittstelle eines Funktionsbausteins oder einen Datenbaustein zu erweitern und den Baustein anschließend ohne Reinitialisierung zu laden, führen Sie folgende Schritte aus. Eine detaillierte Beschreibung der einzelnen Schritte finden Sie in den nachfolgenden Kapiteln:

1. Alle Bausteine haben per Voreinstellung eine Speicherreserve von 100 Byte im nicht-remanenten Speicher. Bei Bedarf können Sie die Größe der Speicherreserve anpassen oder zusätzlich eine Speicherreserve im remanenten Speicher definieren.
2. Aktivieren Sie die Speicherreserve.
3. Erweitern Sie die Bausteinschnittstelle.
4. Übersetzen Sie den Baustein.
5. Laden Sie den Baustein wie gewohnt in die CPU.

Übersicht der Einstellungen zur Speicherreserve

Die Übersicht der Projektnavigation zeigt die Einstellungen zur Speicherreserve für alle Programmbausteine auf einen Blick:

Um die Übersicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Klicken Sie auf die Schaltfläche "Maximiert/minimiert die Übersicht".

Siehe auch

Einstellwerte im Online-Programm initialisieren (Seite 7367)

Speicherreserve für Bausteinerweiterungen einstellen (Seite 7891)

Speicherreserve aktivieren (Seite 7892)

Speicherreserve deaktivieren (Seite 7893)

Bausteinschnittstelle oder Datenbaustein erweitern (Seite 7894)

Speicherreserve zurücksetzen (Seite 7895)

19.3.9.2 Speicherreserve für Bausteinerweiterungen einstellen

Einführung

Grundsätzlich enthält jeder Funktions- oder Datenbaustein per Voreinstellung eine Reserve von 100 Byte. Diese Reserve liegt nicht im remanenten Speicherbereich des Bausteins. Eine Reserve im remanenten Speicher wird per Voreinstellung nicht vorgesehen, da der remanente Speicher auf der CPU begrenzt ist.

Sie können den voreingestellten Wert von 100 Byte, der für alle neu angelegten Bausteine im Projekt gilt, ändern. Darüber hinaus können Sie die Größe der Speicherreserve individuell für einzelne Bausteine ändern und für einzelne Bausteine eine Reserve im remanenten Speicher definieren.

Größe der Speicherreserve für neue Bausteine einstellen

Um die Größe der Speicherreserve für alle neu angelegten Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Tragen Sie in der Gruppe "Speicherreserve für das Laden ohne Reinitialisierung" im Eingabefeld "Speicherreserve" die gewünschte Anzahl von Bytes ein, die für eine spätere Erweiterung der Bausteinschnittstelle vorgesehen werden soll.
   Neu erstellte Funktionsbausteine und globale Datenbausteine erhalten nun eine Speicherreserve in der angegebenen Größe. Bereits existierende Bausteine werden von der Änderung nicht beeinflusst.

   Hinweis

   Die Einstellung gilt nur für Bausteine, die die Funktion "Laden ohne Reinitialisierung" unterstützen. Andere Bausteine werden von der Einstellung nicht beeinflusst.

Größe der Speicherreserve für einen bestehenden Baustein einstellen

Um die Größe der Speicherreserve für einen bestehenden Baustein einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Baustein in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Dialog "Eigenschaften" wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die Gruppe "Laden ohne Reinitialisierung".

4. Tragen Sie im Eingabefeld "Speicherreserve" die gewünschte Anzahl von Bytes ein.
5. Um eine Reserve im remanenten Speicher zu definieren, aktivieren Sie das Optionskästchen "Laden ohne Reinitialisierung für remanente Variablen aktivieren" und tragen im Eingabefeld "Reserve im remanenten Speicher" die gewünschte Anzahl von Bytes ein.

Hinweis

Beachten Sie folgende Hinweise:

- Wenn Sie die Speicherreserve für den aktuellen Baustein bereits aktiviert haben, können Sie die Größe der Speicherreserve nicht mehr ändern.
- Datenelemente werden im Speicher nach bestimmten Regeln angeordnet, die einen möglichst schnellen Zugriff während des Programmablaufs ermöglichen. Aus diesem Grund reduziert sich der verfügbare Speicher in manchen Fällen durch ein neu hinzugefügtes Datenelement um mehr als die tatsächliche Größe des Datenelements. Es ist z. B. möglich, dass sich der verfügbare Speicherbereich durch das Hinzufügen eines Elements vom Typ "WORD" und eines Elements vom Typ "BYTE" um vier Byte reduziert, da alle Datenelemente an einer WORD-Grenze ausgerichtet werden.

Siehe auch

Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)

19.3.9.3 Speicherreserve aktivieren

Einführung

Grundsätzlich enthält jeder Funktions- oder Datenbaustein per Voreinstellung eine Speicherreserve. Die Speicherreserve wird jedoch zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass Sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt.

Voraussetzungen

- Der Baustein ist aktuell übersetzt.
- Den Baustein enthält eine Speicherreserve.

Vorgehen

Um die Speicherreserve für einen Funktionsbaustein oder globalen Datenbaustein zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Wählen Sie in der Funktionsleiste die Schaltfläche "Aktualwerte behalten".

Ergebnis

- Ab jetzt werden alle neu deklarierten Variablen in der Speicherreserve abgelegt und können somit ohne Rückwirkungen auf den laufenden Prozess geladen werden.
- Bereits bestehende Variablen können nicht mehr gelöscht werden. Ihre Eigenschaften können nicht mehr geändert werden, mit Ausnahme der Eigenschaften "Defaultwert", "Startwert" und "Kommentar".
- Lokale Konstanten, die als ARRAY-Grenzen verwendet werden, lassen sich nicht mehr ändern, da sich eine Änderung des Konstantenwerts auf das Speicherlayout des Bausteins auswirken könnte.

Siehe auch

Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)

19.3.9.4 Speicherreserve deaktivieren

Einführung

Um das Speicherlayout eines Bausteins zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, können Sie die Speicherreserve deaktivieren. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve bleibt bestehen und wird wieder frei für weitere Schnittstellenerweiterungen. Sie hat dann wieder die Größe, die in den Bausteineigenschaften definiert ist.

Hinweis

Beachten Sie, dass der Baustein neu übersetzt und geladen werden muss, nachdem Sie die Speicherreserve deaktiviert haben. Ein "Laden ohne Reinitialisierung" ist nach dem Deaktivieren der Speicherreserve nicht möglich.

Voraussetzung

Die Speicherreserve des Bausteins ist aktiviert.

Vorgehen

Um die Speicherreserve für einen Funktionsbaustein oder globalen Datenbaustein zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Wählen Sie in der Funktionsleiste die Schaltfläche "Aktualwerte behalten".

Ergebnis

Alle Variablen, die bisher in der Speicherreserve lagen, werden aus dem Reservebereich in den regulären Bereich verschoben. Dadurch muss der Baustein neu übersetzt und geladen werden. Beim Laden werden die Werte der Variablen in der CPU neu initialisiert.

Siehe auch

Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)

19.3.9.5 Bausteinschnittstelle oder Datenbaustein erweitern

Einführung

Nachdem Sie die Speicherreserve eines Funktionsbausteins oder eines globalen Datenbausteins aktiviert haben, können Sie neue Variablen einfügen.

Voraussetzung

Die Speicherreserve ist aktiviert.

Vorgehen

Um zusätzliche Variablen zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie einen Deklarationsabschnitt, z. B. "Input", "Output", "InOut" oder "Static".
2. Deklarieren Sie darin wie gewohnt eine oder mehrere Variablen und tragen Sie deren Eigenschaften in die Spalten ein. In der Spalte "Remanenz" eines FB können Sie nur zwischen den Einstellungen "Remanent" und "Nicht remanent" wählen. Die Einstellung "Im IDB setzten" steht in der Speicherreserve nicht zur Verfügung.

   Hinweis

   Es ist nicht möglich, neue Variablen innerhalb einer Variable mit strukturiertem Datentyp hinzuzufügen. Sie können z. B. innerhalb einer bestehenden Struktur keine neuen Strukturelemente deklarieren.

   Sie können jedoch innerhalb der Speicherreserve neue Strukturen anlegen.

3. Übersetzen Sie die geänderten Bausteine, z. B. indem Sie sie in der Projektnavigation anwählen und im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen) wählen.
   Beim Übersetzen werden die neu deklarierten Variablen in die Speicherreserve eingefügt. Wenn die Speicherreserve nicht groß genug ist, wird der Übersetzungsvorgang abgebrochen und Sie werden mit einer Fehlermeldung darüber informiert.
4. Laden Sie die geänderten Bausteine, z. B. indem Sie sie in der Projektnavigation anwählen und im Kontextmenü den Befehl "Laden > Software (nur Änderungen)" wählen.
   Wenn Sie Bausteinerweiterungen laden, werden nur die neu eingefügten Variablen mit den definierten Startwerten initialisiert. Die bereits Online existierenden Variablen werden nicht neu initialisiert.

Siehe auch

Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)

19.3.9.6 Speicherreserve zurücksetzen

Einführung

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie die Möglichkeit, die Speicherreserve einzelner oder mehrerer Bausteine in einem Schritt zurückzusetzen. Dabei werden alle Variablen aus der Speicherreserve in den regulären Speicherbereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

Voraussetzung

- Die Speicherreserve des Bausteins ist aktiviert.
- Die Speicherreserve enthält Variablen.

Vorgehen

Um die Speicherreserve eines oder mehrerer Bausteine zurückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gesamten Ordner "Programmbausteine" oder einzelne Bausteine innerhalb des Ordners.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Speicherreserve zurücksetzen)".

Ergebnis

Alle Variablen in den selektierten Bausteinen, die bisher in der Speicherreserve lagen, werden aus dem Reservebereich in den regulären Bereich verschoben. Der Baustein wird neu übersetzt. Beim nächsten Laden werden die Variablen neu initialisiert. Die eingestellte Speicherreserve des Bausteins bleibt erhalten und ist weiterhin aktiv.

⚠ GEFAHR

Reinitialisierung beim nächsten Laden

Wenn Sie die Speicherreserve zurücksetzen, werden beim nächsten Laden alle Variablen, die zuvor in der Speicherreserve lagen, neu initialisiert. Das betrifft sowohl die remanenten als auch die nicht-remanenten Variablen.

Das Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann schwere Sach- und Personenschäden verursachen! Es ist deshalb unbedingt erforderlich, das Programm nach Zurücksetzen der Speicherreserve ausführlich in einer Testumgebung zu überprüfen, bevor Sie es in den laufenden Betrieb übernehmen.

Siehe auch

Grundlagen zum Laden von Bausteinerweiterungen ohne Reinitialisierung (Seite 7889)

19.4 Bausteine laden für S7-300/400

19.4.1 Einführung zum Laden von Bausteinen

Bausteine in Gerät laden

Damit die CPU das Anwenderprogramm bearbeiten kann, müssen die Programmdaten zunächst übersetzt und anschließend in das Gerät geladen werden.

Sie können die Programmdaten in Geräte und auf Memory Cards laden.

Das Laden in und vom Gerät kann über verschiedene Wege gestartet werden:

- über die Befehle im Menü "Online"
- über das Kontextmenü
- über die Schaltfläche "Laden" in der Toolbar

Das Laden kann in der Projektnavigation von folgenden Objekten aus gestartet werden:

- von Geräten
- vom Ordner "Programmbausteine"
- von einzelnen Bausteinen
- vom Ordner "PLC-Datentypen"

Abhängig vom selektierten Objekt und dem jeweils ausgewählten Befehl können Sie folgende Komponenten über das Menü "Online" laden:

- Laden in Gerät
  Sowohl die Hardware-Konfiguration als auch die Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen.
- Erweitertes Laden in Gerät
  Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen. Beim erweiterten Laden können Sie zusätzlich noch ein Zielgerät und die Einstellungen für ein Zielgerät auswählen.
- PLC-Programm ins Gerät laden und zurücksetzen
  Es wird das PLC-Programm inklusive aller Bausteine und PLC-Datentypen in das Ziel geladen und alle Werte auf ihre Startwerte zurückgesetzt. Beachten Sie, dass dies auch für die remanenten Werte der Fall ist. Die CPU geht dabei in "STOP".
- Anwenderprogramm auf Memory Card laden
  Das gesamte Programm wird auf eine gesteckte Memory Card geladen.

Abhängig vom selektierten Objekt und dem jeweils ausgewählten Befehl können Sie folgende Komponenten über das Kontextmenü laden:

- Hardware und Software (nur Änderungen)
  Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen, falls Unterschiede zwischen den Online- und Offline-Versionen bestehen.
- Hardwarekonfiguration
  Es wird nur die Hardware-Konfiguration in das Ziel geladen.
- Software (nur Änderungen)
  Es werden nur die Objekte in das Ziel geladen, die online und offline unterschiedlich sind.
- Software (komplett laden)
  Die Software wird komplett in das Ziel geladen.

Hinweis

Keine gültige Hardwarekonfiguration vorhanden

Falls beim Laden von Software keine gültige Hardwarekonfiguration auf der CPU gefunden wird, so wird die vorhandene Hardware beim "Laden in Gerät" mitgeladen.

Beachten Sie bitte, dass in diesem Fall beim Laden Hardwarekomponenten geladen werden, obwohl Sie den Befehl zum Laden von Software ausgeführt haben!

Hinweis

Laden von mehr als 16 Bausteinen im Betriebszustand RUN

Falls beim Laden in RUN mehr Bausteine geladen werden sollen, als die CPU-Kapazität für einen Ladevorgang zulässt, können die zu ladenden Bausteine asynchron über mehrere Zyklen hinweg geladen werden. Wenn diese Fall auftritt, können Sie diese Option im Dialog "Vorschau Laden" aus der Klappliste auswählen.

Beachten Sie bitte, dass erst nach Abschluss eines Ladens über mehrere Zyklen hinweg wieder ein konsistenter Zustand auf der CPU erreicht wird.

Hinweis

Laden von know-how-geschützten Bausteinen

Wenn Sie einen know-how-geschützten Baustein in ein Gerät laden, werden keine Wiederherstellungsinformationen mitgeladen. Das bedeutet, dass Sie einen know-how-geschützten Baustein nicht wieder öffnen können, wenn Sie ihn vom Gerät laden.

Hinweis

Laden von CPU S7-300/400

Beachten Sie beim Laden von Gerät, dass Alarm-Informationen eventuell verloren gehen können und dies beim Übersetzen zu Fehlern führen kann.

PLC-Programme von Gerät laden

Sie können die auf einem Gerät geladenen Programmdaten wieder in Ihr Projekt laden.

Dazu gehören die vorhandenen Bausteine und PLC-Datentypen.

Dies ist z. B. notwendig, um Bausteine zu bearbeiten, die nur in diesem Gerät vorhanden sind. Sie haben die Möglichkeit entweder alle vorhandenen Bausteine (Organisationsbausteine, Funktionsbausteine, Funktionen, Datenbausteine) und die PLC-Datentypen oder einzelne Bausteine in das Projekt zu laden.

Hinweis

Laden von CPU S7-300/400

Beachten Sie beim Laden von Gerät, dass Alarm-Informationen eventuell verloren gehen können und dies beim Übersetzen zu Fehlern führen kann.

Bausteine auf oder von einer Memory Card laden

Memory Cards sind steckbare Speicherkarten, die z. B. dazu dienen, den Speicher einer CPU zu erweitern und Daten gegen Spannungsausfall zu sichern. Sie können abhängig von der verwendeten CPU verschiedene Typen von Speicherkarten verwenden. Beachten Sie für weitere Informationen zu den zulässigen Memory Cards für Ihre CPU im Kapitel "Arbeitsweise für S7-300/400 CPU" unter "Zusatzinformationen zu Konfigurationen".

GRAPH-Funktionsbausteine laden

Wenn Sie einen GRAPH-Funktionsbaustein zusammen mit seinem Instanz-Datenbaustein laden, beginnt die Bearbeitung der Ablaufkette wieder mit dem Initialschritt. Dadurch können Probleme bei der Synchronisation der Ablaufkette mit dem Prozess auftreten. Sie können diese Probleme vermeiden, indem Sie die Ablaufkette vor dem Laden ausschalten.

Auswirkungen eines Ladevorgangs auf die Variablenwerte eines Datenbausteins

Wenn Datenbausteine im Betriebszustand STOP in ein Gerät geladen werden, wirkt sich der nächste Übergang des Geräts nach RUN auf die Aktualwerte der Variablen folgendermaßen aus:

- Variablen von nicht remanenten Datenbausteinen erhalten ihre definierten Startwerte.
- Variablen von remanenten Datenbausteine behalten ihre Werte nur, wenn folgende Bedingungen erfüllt sind:
  - Sie haben den Datenbaustein über "Laden in Gerät > Software (nur Änderungen)" geladen.
  - Sie haben die Struktur des Datenbausteins nicht geändert.

  Ansonsten erhalten auch die Variablen von remanenten Datenbausteinen ihre definierten Startwerte.

I&M-Daten in PROFINET-IO-Geräte und ihre Module laden

Beim Laden in ein Gerät können Sie zusätzlich zu den oben genannten Komponenten I&M-Daten in PROFINET-IO-Geräte und ihre Module laden.

19.4.2 Bausteine im Betriebszustand "RUN" in Gerät laden

Grundlagen zum Laden von Bausteinen im Betriebszustand RUN

Wenn Sie geänderte Bausteine in ein Gerät laden, ist es nicht immer notwendig, dass das Gerät in den Betriebszustand STOP wechselt. Bei einem Ladevorgang prüft das Engineering System, ob das Gerät vor dem Laden gestoppt werden muss. Das Ergebnis dieser Prüfung wird im Dialog "Vorschau Laden" angezeigt.

Wenn ein Wechsel in den Betriebszustand STOP notwendig ist, können Sie den Ladevorgang erst nach dem Stoppen der CPU fortsetzen.

Wenn die Voraussetzungen erfüllt sind, können Sie ein geändertes Programm oder Programmteile auch im Betriebszustand RUN in eine CPU laden.

Grundsätzlich gibt es folgende Einschränkungen bei einem Ladevorgang im Betriebszustand RUN:

- Abhängig vom verwendeten Gerät können die Anzahl und die Art der Bausteine beschränkt sein, die Sie im Betriebszustand RUN laden können.
- Sie haben die Option, Bausteine im RUN über mehrere Zyklen hinweg ins Gerät zu laden. Dies ist nur möglich, wenn Sie den Befehl "Laden > Software (nur Änderungen)" verwenden.
- Das Laden aller Bausteine über "PLC-Programm ins Gerät laden und zurücksetzen" ist nicht möglich.
- Wenn Sie die Hardware-Konfiguration des Geräts geändert haben, das als Ziel des Ladevorgangs verwendet werden soll, ist das Laden im Betriebszustand RUN nicht möglich.

Hinweis

Durch einen Ladevorgang im Betriebszustand RUN werden keine Aktualparameter überschrieben. Änderungen an den Aktualparametern erfolgen erst beim nächsten Betriebszustandswechsel von STOP nach RUN.

Laden von Änderungen im Betriebszustand RUN

Die folgende Tabelle zeigt, welche Programm- oder Konfigurationsänderungen im Betriebszustand RUN geladen werden können, sortiert nach CPU-Familien und unter Berücksichtigung der Firmware-Versionen der CPUs

Erläuterungen zur Tabelle:

- RUN: Änderung ist nicht nur im Betriebszustand STOP, sondern auch im Betriebszustand RUN in die CPU ladbar.
- RUN (< 57): Bis zu 56 neue oder geänderte Objekte/Bausteine kann die CPU in einem Programmzyklus integrieren. Wenn Sie mehr Objekte/Bausteine laden, werden sie in mehreren aufeinander folgenden Programmzyklen integriert. Wenn Sie alle Objekte/Bausteine konsistent laden wollen, dann müssen Sie die CPU in den Betriebszustand STOP setzen. Bei S7-300-CPUs mit Parametriermöglichkeit "Prozessbetrieb/Testbetrieb" variiert diese Anzahl je nach Einstellung.
- RUN (Init): Änderung ist im Betriebszustand RUN ladbar; geladene Datenbausteine werden reinitialisiert.
- STOP: Änderung ist nur im Betriebszustand STOP ladbar.
- STOP (Reset): Änderung ist nur im Betriebszustand STOP ladbar; alle Daten, einschließlich der remanenten Daten, werden zurückgesetzt.

	S7-300	**S7-400**	**S7-1200 ab V4.0**	**S7-1500**
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**			
Geänderte Eigenschaften von HW-Komponenten. Dazu gehören auch Änderungen bei den Kommentaren in der HW-Konfiguration.	STOP	STOP	STOP	STOP
Hinzugefügte HW-Komponenten	STOP	STOP	STOP	STOP
Neue/geänderte Textlisten (Meldungen)	**RUN**	**RUN**	STOP	**RUN (ab V1.1)**
Geänderte Kommentare (neu, geändert, gelöscht) mit Ausnahme der Kommentare in der HW-Konfiguration	-	-	**RUN**	**RUN**
Anzahl Bausteine synchron laden	**RUN (<17)**	**RUN (<57)**	**RUN (<21)**	**RUN (alle)**
PLC-Programm ins Gerät laden und zurücksetzen	STOP (Reset)	STOP (Reset)	STOP (Reset)	STOP (Reset)
Neuer OB	**RUN**	**RUN**	STOP	**RUN**

	S7-300	S7-400	S7-1200 ab V4.0	S7-1500
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**			
Geänderter OB: Codeänderungen	**RUN**	**RUN**	**RUN**	**RUN**
OB mit geänderten Eigenschaften (z. B. Zykluszeit-Änderung)	**RUN**	**RUN**	STOP	**RUN**
Gelöschter OB	**RUN**	**RUN**	STOP	**RUN**
Neuer FB/FC/DB/Anwenderdatentyp (UDT)	**RUN**	**RUN**	**RUN**	**RUN**
Gelöschter FB/FC/DB/Anwenderdatentyp (UDT)	**RUN**	**RUN**	**RUN**	**RUN**
Geänderter FB/FC: Codeänderung	**RUN**	**RUN**	**RUN**	**RUN**
Geänderter FB/FC: Schnittstellenänderung*	STOP	STOP	**RUN**	**RUN**
Geänderter DB: Eigenschaft geändert (Attribut "Nur im Ladespeicher ablegen" geändert)	STOP	STOP	**RUN (Init)**	**RUN (Init)**
Geänderter DB (Speicherreserve nicht aktiviert): Name/Typ von Variablen geändert, Variablen hinzugefügt oder gelöscht **	**RUN (Init)**	**RUN (Init)**	**RUN (Init)**	**RUN (Init)**
Geänderter DB (Speicherreserve aktiviert): Neue Variablen hinzugefügt**	-	-	**RUN**	**RUN**
Geänderter Anwenderdatentyp (UDT)	STOP	STOP	**RUN (Init)**	**RUN (Init)**
Neue PLC-Variablen hinzugefügt (Zeiten, Zähler, Merker)	**RUN**	**RUN**	**RUN**	**RUN**
Geänderte Remanenzeinstellungen (Zeiten, Zähler, Merker, DB-Bereich)	STOP	STOP	STOP	STOP
Motion Control Technologieobjekte: Änderungen MC-Servo Zeittakt, Änderung von freilaufend auf zyklisch (und umgekehrt). Änderungen an der HW-Schnittstelle des TOs	-	-	-	STOP

* Falls die Schnittstellenänderung zu strukturellen Änderungen am Instanz-DB führt, siehe "Geänderter DB..."

** Zur Auswirkung des Ladens von Datenbaustein-Änderungen auf den Datenbaustein-Inhalt siehe Abschnitt "Laden von Datenbausteinen in die CPU".

Laden von Änderungen im RUN mit früheren CPU-Firmware-Versionen

Die folgende Tabelle zeigt, welche Änderungen Sie im Betriebszustand RUN laden können, und zwar für frühere CPU-Firmware-Versionen.

	S7-1200 V1.0 - 2.1	S7-1200 V2.2 - V3.0
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**	
Geänderte Eigenschaften von HW-Komponenten. Dazu gehören auch Änderungen bei den Kommentaren in der HW-Konfiguration.	STOP	STOP
Hinzugefügte HW-Komponenten	STOP	STOP

	S7-1200 V1.0 - 2.1	S7-1200 V2.2 - V3.0
Aktion/Art der Änderung	**Laden möglich im Betriebszustand ...**	
Neue/geänderte Textlisten (Meldungen)	STOP	STOP
Geänderte Kommentare (neu, geändert, gelöscht) mit Ausnahme der Kommentare in der HW-Konfiguration.	STOP	**RUN**
Anzahl Bausteine synchron laden	STOP	**RUN (<11)**
PLC-Programm ins Gerät laden und zurücksetzen	STOP (Reset)	STOP (Reset)
Neuer OB	STOP	STOP
Geänderter OB: Codeänderungen	STOP	**RUN**
OB mit geänderten Eigenschaften (z. B. Zykluszeit-Änderung)	STOP	STOP
Gelöschter OB	STOP	STOP
Neuer FB/FC/DB/Anwenderdatentyp (UDT)	STOP	**RUN**
Gelöschter FB/FC/DB/Anwenderdatentyp (UDT)	STOP	**RUN**
Geänderter FB/FC: Codeänderung	STOP	**RUN**
Geänderter FB/FC: Schnittstellenänderung	STOP	STOP
Geänderter DB: Eigenschaft geändert (Attribut "Nur im Ladespeicher ablegen" geändert)	STOP	STOP
Geänderter DB (Speicherreserve nicht aktiviert): Name/Typ von Variablen geändert, Variablen hinzugefügt oder gelöscht	STOP	STOP
Geänderter Anwenderdatentyp (UDT)	STOP	STOP
Neue PLC-Variablen hinzugefügt (Zeiten, Zähler, Merker)	STOP	STOP
Geänderte Remanenzeinstellungen (Zeiten, Zähler, Merker, DB-Bereich)	STOP	STOP

Laden von Datenbausteinen in die CPU

Je nach den vorliegenden Randbedingungen hat das Laden von neuen oder geänderten Datenbausteinen Einfluss auf die Aktualwerte im Datenbaustein:

Neue Datenbausteine laden	Aktualwerte in den neuen Datenbausteinen sind auf die Startwerte gesetzt.
Strukturell geänderte Datenbausteine laden (Speicherreserve nicht aktiviert)	Aktualwerte hinzugefügter Variablen in den strukturell geänderten Datenbausteinen sind auf die Startwerte gesetzt.

Strukturell geänderte Datenbausteine laden (Speicherreserve aktiviert)	Folgende Aktualwerte bleiben erhalten: • Aktualwerte von Variablen außerhalb der Speicherreserve • Aktualwerte von nicht geänderten Variablen innerhalb der Speicherreserve. Aktualwerte hinzugefügter Variablen innerhalb der Speicherreserve sind auf die Startwerte gesetzt.
Einfach geänderten Datenbaustein laden (keine Strukturänderung)	Aktualwerte bleiben erhalten.

Weitere Informationen

Weitere Informationen zum Laden von Bausteinerweiterungen ohne Reinitialisierung und zum Laden von geänderten Werten bei Datenbausteinen finden Sie unter "Siehe auch".

Siehe auch

Bausteine vom Programmiereditor in Gerät laden (Seite 7904)

Bausteine aus der Projektnavigation in Gerät laden (Seite 7906)

19.4.3 Bausteine vom Programmiereditor in Gerät laden

Voraussetzung

Der zu ladende Baustein ist geöffnet.

Vorgehen

Um einen Baustein im Programmiereditor in das Gerät zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Anweisungsfenster des Programmiereditors.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden".
   - Sie können Ihre bevorzugt verwendeten Verbindungsparameter unter "Extras > Einstellungen > Online & Diagnose" als Voreinstellung speichern. Beim erstmaligen Laden mit neuen Verbindungsparametern erhalten Sie automatisch eine Abfrage, ob diese als Voreinstellung gespeichert werden sollen. Klicken Sie auf "Ja", um die aktuellen Verbindungsparameter als Voreinstellung zu speichern.
   - Sie haben die Möglichkeit, sich alle kompatiblen Teilnehmer anzeigen zu lassen, indem Sie die entsprechende Option aktivieren und auf den Befehl "Suche starten" klicken. Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wenn das Gerät bereits online verbunden war, werden die Projektdaten bei Bedarf übersetzt und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv

Hinweis

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

Hinweis

Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, können Sie bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine laden. Wählen Sie dazu die Aktion "Konsistentes Laden" aus.

4. Klicken Sie auf "Laden".
   Falls es Synchronisierungsbedarf gibt, wird vom System automatisch der Dialog "Synchronisierung" angezeigt. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Synchronisieren notwendige Aktionen vorgeschlagen. Sie haben die Möglichkeit, diese Aktionen auszuführen oder das Laden ohne Synchronisierung zu erzwingen, indem Sie auf "Laden in Gerät erzwingen" klicken. Falls Sie die vorgeschlagenen Aktionen ausgeführt haben, werden Sie gefragt, ob Sie mit dem Laden fortfahren wollen. Klicken Sie auf "Laden fortsetzten", damit der Baustein geladen wird. Der Dialog "Ergebnisse laden" wird daraufhin geöffnet und zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
5. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Der Code für den Baustein wird in das Gerät geladen. Wenn Sie "Konsistentes Laden" ausgewählt haben, werden die Bausteine, auf die sich die Änderungen auswirken, übersetzt und ebenfalls in das Gerät geladen.

Ob der Ladevorgang erfolgreich war, erkennen Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein".

Siehe auch

Bausteine aus der Projektnavigation in Gerät laden (Seite 7906)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 7900)

19.4.4 Bausteine aus der Projektnavigation in Gerät laden

In der Projektnavigation können Sie einen, mehrere oder alle Bausteine in ein Gerät laden.

Einen oder mehrere Bausteine aus der Projektnavigation in Gerät laden

Um einen oder mehrere Bausteine aus der Projektnavigation in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Selektieren Sie die Bausteine, die Sie laden möchten.

3. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wenn das Gerät bereits online verbunden war, werden die Projektdaten bei Bedarf übersetzt und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

   Hinweis

   Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, können Sie bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine laden. Wählen Sie dazu die Aktion "Konsistentes Laden" aus.

5. Klicken Sie auf "Laden".
   Die Bausteine werden geladen und der Dialog "Ergebnisse laden" wird geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Alle Bausteine aus der Projektnavigation in Gerät laden

Um alle Bausteine des Ordners "Programmbausteine" aus der Projektnavigation heraus in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Wählen Sie im Kontextmenü das Untermenü "Laden in Gerät".

3. Wenn Sie nur die Änderungen seit dem letzten Ladevorgang laden möchten, wählen Sie die Option "Software (nur Änderungen)". Wenn alle Bausteine komplett geladen und alle Werte auf ihre Startwerte zurückgesetzt werden sollen, wählen Sie die Option "PLC-Programm ins Gerät laden und zurücksetzen".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.
     Siehe auch: Online verbinden und Online-Verbindung trennen
   - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

   Hinweis

   Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, können Sie bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine laden. Wählen Sie dazu die Aktion "Konsistentes Laden" aus.

5. Klicken Sie auf "Laden".
   Die Bausteine werden geladen und der Dialog "Ergebnisse laden" wird geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Der Code für die Bausteine wird in das Gerät geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, erkennen Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein".

Siehe auch

Bausteine vom Programmiereditor in Gerät laden (Seite 7904)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 7900)

19.4.5 Bausteine von Gerät laden

Sie können entweder alle oder nur einzelne Bausteine von einem Gerät in Ihr Projekt laden.

Hinweis

SCL-Bausteine und GRAPH-Funktionsbausteine können zwar von einer S7-300/400 geladen werden, sind jedoch nach dem Laden nicht editierbar.

Voraussetzung

Die Online- und Offline-Version der zu ladenden Bausteine ist unterschiedlich oder die Bausteine existieren nur online.

Alle Bausteine von einem Gerät laden

Um alle Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.
   Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation den Ordner des Geräts, von dem Sie Bausteine laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Einzelne Bausteine von einem Gerät laden

Um einzelne Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.
   Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation die Bausteine, die Sie vom Gerät laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".
   Der Ladevorgang wird durchgeführt.

Ergebnis

Die Bausteine werden von dem Gerät in das Projekt geladen. Sie können die Bausteine bearbeiten, übersetzen und zurück in das Gerät laden.

19.4.6 Bausteine auf eine Memory Card laden

Voraussetzung

Der Ordner "Programmbausteine" der Memory Card ist geöffnet.

Vorgehen

Um Bausteine auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine" des Geräts.
2. Selektieren Sie die Bausteine, die Sie auf die Memory Card laden möchten.
3. Ziehen Sie die Bausteine in der Projektnavigation auf den Ordner "Programmbausteine" der Memory Card. Alternativ können Sie die Bausteine kopieren und in die Memory Card einfügen.
   Falls notwendig, werden die Bausteine übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

   Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

5. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Oder:

1. Selektieren Sie in der Projektnavigation die Bausteine, die Sie laden möchten.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben".
   Der Dialog "Memory Card auswählen" wird geöffnet.
3. Wählen Sie eine Memory Card aus, die mit der CPU kompatibel ist.
   Im unteren Bereich des Dialogs wird eine Schaltfläche mit einem grünen Haken aktiv.
4. Klicken Sie auf die Schaltfläche mit dem grünen Haken.
   Falls notwendig, werden die Projektdaten übersetzt. Anschließend wird der Dialog "Vorschau Laden" geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
5. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
6. Klicken Sie auf die Schaltfläche "Laden".
   Der Ladevorgang wird durchgeführt.

Ergebnis

Die ausgewählten Bausteine werden auf die Memory Card geladen. Ob der Ladevorgang erfolgreich war, erkennen Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein".

Siehe auch

Bausteine von einer Memory Card laden (Seite 7911)

19.4.7 Bausteine von einer Memory Card laden

Sie können immer nur alle Bausteine von einer Memory Card in Ihr Projekt laden. Beachten Sie, dass beim Laden von einer Memory Card keine symbolischen Informationen mitgeladen werden.

Voraussetzung

Die Memory Card wird angezeigt.

Siehe auch: Auf Memory Cards zugreifen

Vorgehen

Um Bausteine von einer Memory Card in das Projekt zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf den Ordner des Geräts im Projekt. Alternativ können Sie die Memory Card kopieren und in das Gerät einfügen.
   Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden von Gerät".

Siehe auch

Bausteine auf eine Memory Card laden (Seite 7910)

19.4.8 Ablaufkette vor dem Laden eines GRAPH-DB ausschalten

Sie können das Ausschalten der Ablaufkette vor dem Laden eines Instanz-Datenbausteins entweder global festlegen oder während des Ladevorgangs.

Ablaufkette global ausschalten

Um die Ablaufkette global für jeden Ladevorgang eines Instanz-Datenbausteins auszuschalten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".
   Bei zukünftigen Ladevorgängen wird die Ablaufkette vor dem Laden des Instanz-Datenbausteins ausgeschaltet.

Ablaufkette während des Ladevorgangs ausschalten

Um die Ablaufkette während des Ladevorgangs auszuschalten, gehen Sie folgendermaßen vor:

1. Laden Sie den GRAPH-Funktionsbaustein in das Gerät.
   Während des Ladevorgangs erscheint der Dialog "Vorschau laden". Hier werden Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen. Wenn der Instanz-Datenbaustein zusammen mit dem GRAPH-Funktionsbaustein geladen werden muss, wird Ihnen im Dialog "Vorschau laden" die Aktion "Kette vor Laden des DB ausschalten" angeboten.
2. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".

19.5 PLC-Variablen laden

19.5.1 Einführung zum Laden von PLC-Variablen

Einführung

PLC-Variablen können, ebenso wie Bausteine, in ein Gerät geladen und von einem Gerät wieder in ein Projekt hochgeladen werden. Wenn Sie in der Projektnavigation eine PLC-Variablentabelle selektieren, steht Ihnen der Befehl "Laden in Gerät > Software (nur Änderungen)" zur Verfügung.

Sie können das Laden sowohl über den Befehl im Menü "Online" als auch über das Kontextmenü oder über die Schaltfläche "Laden" in der Funktionsleiste ausführen.

Wenn Sie das Laden in Gerät von einem anderen Objekt aus starten, z. B. über den Bausteinordner, werden die Variablentabellen immer konsistent mitgeladen.

Diese Funktionalität wird von folgenden Geräten unterstützt:

- CPU S7-1200
- CPU S7-1500

Bei einer S7-1500 ab Firmwarestand V2.5 wird zusätzlich zu den PLC-Variablen auch die Information in die CPU geladen, zu welcher PLC-Variablentabelle die Variablen gehören.

Bei Geräten, die diese Funktionalität nicht unterstützen, steht der Befehl nicht zur Verfügung.

Regeln für das Laden in Gerät von PLC-Variablentabellen (nur S7-1500 ab V2.5)

Offline vorhandene Variablentabellen können in der vorgegebenen Struktur in die CPU geladen werden.

Dabei gelten folgende Regeln:

- Für das Laden von PLC-Variablen gelten die gleichen Voraussetzungen und Vorgehensweisen wie für das Laden von Bausteinen.
- Falls beim Laden Inkonsistenzen bei PLC-Variablen festgestellt werden, wird der Synchronisierungsdialog angezeigt. Um die Konsistenz innerhalb eines gemeinsamen bearbeiteten Projektes sicher zu stellen, ist es notwendig, die jeweils geänderten Daten vor dem Laden zu synchronisieren, damit es nicht zu ungewollten Überschreibungen kommt. In diesem Fall werden die zu synchronisierenden Daten mit dem aktuellen Status (Offline/Online-Vergleich) und den möglichen Aktionen angezeigt. Über den Vergleichseditor können Sie nachvollziehen, welche Variablen sich in welcher Variablentabelle geändert haben. Dabei werden die Variablen immer namensbasiert verglichen. Im Konfliktfall muss der jeweilige Benutzer entscheiden, welche Variablen er behalten und laden möchte.

Siehe auch:

- PLC-Variablen vergleichen (Seite 7951)
- Bausteine aus der Projektnavigation in Gerät laden (Seite 7881)
- Bausteine auf eine Memory Card laden (Seite 7886)

Regeln für das Laden von Gerät für PLC-Variablentabellen (nur S7-1500 ab V2.5)

Online vorhandene Variablentabellen können in der vorgegebenen Struktur in das Projekt geladen werden.

Dabei gelten folgende Regeln:

- Für das Hochladen von PLC-Variablen gelten die gleichen Voraussetzungen und Vorgehensweisen wie für das Hochladen von Bausteinen.
- Für das Laden von Gerät gibt es bei PLC-Variablentabellen folgende Besonderheiten:
  - Variablen, die online und offline mit identischem Namen vorhanden sind, werden offline überschrieben.
  - Variablen, die nur offline vorhanden sind, bleiben unverändert erhalten.
  - Variablen, die nur online vorhanden sind, werden offline neu angelegt.
  - Variablen, die online in einer anderen Variablentabelle vorhanden sind, werden in die jeweils hochgeladene Variablentabelle eingefügt und aus der anderen Variablentabelle gelöscht.

Siehe auch:

- Bausteine von Gerät laden (Seite 7884)
- Bausteine von einer Memory Card laden (Seite 7887)

Beispiel für das Laden von Gerät für PLC-Variablentabellen (nur S7-1500 ab V2.5)

Das nachfolgende Beispiel verdeutlicht das Systemverhalten beim Hochladen von Variablentabellen, wenn diese Online/Offline unterschiedlich sind:

- Die Variable "Tag_A" ist in der Variablentabelle 1 nur offline vorhanden.
- Die Variable "Tag_A" ist in der Variablentabelle 2 nur online vorhanden.

Resultat beim Hochladen von Variablentabelle 1:

- "Tag_A" ist in der Variablentabelle 1 nur offline vorhanden und bleibt deshalb in der Variablentabelle 1 nach dem Hochladen unverändert erhalten.

Resultat beim Hochladen von Variablentabelle 2:

- "Tag_A" ist in der Variablentabelle 2 nur online vorhanden und wird deshalb nach dem Hochladen in der Variablentabelle 2 hinzugefügt und in der Variablentabelle 1 gelöscht.

Siehe auch

PLC-Variablen vergleichen (Seite 7951)

Bausteine aus der Projektnavigation in Gerät laden (Seite 7881)

Bausteine von Gerät laden (Seite 7884)

Bausteine von einer Memory Card laden (Seite 7887)

Bausteine auf eine Memory Card laden (Seite 7886)

Online/Offline-Sichtbarkeit von PLC-Variablen (Seite 7916)

19.5.2 Online/Offline-Sichtbarkeit von PLC-Variablen

Einführung

PLC-Variablentabellen können, ebenso wie Bausteine, in ein Gerät geladen und von einem Gerät wieder in ein Projekt hochgeladen werden.

Abhängig davon, ob Variablentabellen nur offline oder auch online vorhanden sind, werden diese im TIA Portal unterschiedlich angezeigt.

Sichtbarkeit von PLC-Variablentabellen (nur S7-1500 ab V2.5)

Die PLC-Variablentabellen werden im TIA Portal wie folgt angezeigt:

Anzeige von PLC-Variablentabellen	Sichtbarkeit	mögliche Aktionen
Offline-Ansicht in der Projektnavigation	Offline-Daten werden permanent angezeigt, sofern vorhanden. Nach einem Laden von Gerät sind die Online-Daten auch offline sichtbar. Es können jedoch trotzdem Unterschiede vorhanden sein, da nur offline vorhandene Variablen nach dem Laden von Gerät erhalten bleiben.	Das Öffnen und das Bearbeiten im Editor sind möglich. Das Laden in und von Gerät sowie in und von Memory Cards ist möglich.
Online-Ansicht in der Projektnavigation	Online-Daten werden im Online-Modus angezeigt. Online-Daten werden auch offline angezeigt, falls es keine entsprechende Offline-Variablentabelle gibt.	Der Editor wird schreibgeschützt geöffnet, ein Bearbeiten ist nicht möglich. Sie können sich jedoch informieren, welche Variablen vorhanden sind.
Bei den Erreichbaren Teilnehmern	Nach dem Laden in Gerät.	Der Editor wird schreibgeschützt geöffnet, ein Bearbeiten ist nicht möglich. Sie können sich jedoch informieren, welche Variablen vorhanden sind.
Auf lokalen Memory Cards	Nach dem Laden auf die Memory Card.	Der Editor wird schreibgeschützt geöffnet, ein Bearbeiten ist nicht möglich. Sie können sich jedoch informieren, welche Variablen vorhanden sind.
Im Vergleichseditor	Sofern eine oder mehrere Variablentabellen offline und/oder online vorhanden sind, können Vergleichsoperationen durchgeführt werden.	Das Öffnen des Vergleichseditors ist möglich und die Vergleichsergebnisse für die ausgewählten Variablen werden namensbasiert angezeigt.

Weiterführende Informationen, wie Sie den Vergleichseditor zum Vergleich von PLC-Variablen nutzen können, finden Sie hier: PLC-Variablen vergleichen (Seite 7951)

Siehe auch

PLC-Variablen vergleichen (Seite 7951)

Einführung zum Laden von PLC-Variablen (Seite 7914)

PLC-Programme vergleichen 20

20.1 Grundlagen zum Vergleichen von PLC-Programmen

20.1.1 Einführung zum Vergleichen von PLC-Programmen

Funktion

Sie haben die Möglichkeit, folgende Objekte eines PLC-Programms miteinander zu vergleichen, um mögliche Unterschiede zu ermitteln:

- Codebausteine mit anderen Codebausteinen
- Datenbausteine mit anderen Datenbausteinen
- PLC-Variablen einer PLC-Variablentabelle mit den PLC-Variablen einer anderen PLC-Variablentabelle
- PLC-Datentypen mit anderen PLC-Datentypen

Der Vergleich von Variablen erfolgt über die Variablennamen.

Vergleichsarten und Vergleichsstufen

Prinzipiell können Sie zwei unterschiedliche Vergleichsarten einsetzen:

- Offline/Online-Vergleich
  Die Objekte im Projekt werden mit den Objekten des entsprechenden Online-Geräts verglichen. Dazu ist eine Online-Verbindung zu dem Gerät notwendig.
- Offline/Offline-Vergleich
  Beim Offline/Oflfine-Vergleich können Sie die Objekte zweier Geräte im aktuell geöffneten Projekt vergleichen. Als Vergleichsgerät kann auch ein Gerät aus einem Referenzprojekt oder einer Bibliothek auf die rechte Drop-Fläche gezogen werden. Sie können nur Geräte innerhalb einer TIA Portal-Instanz miteinander vergleichen.

Beachten Sie, dass Sie nicht beliebig viele Vergleiche zur gleichen Zeit durchführen können, sondern nur einen Vergleich pro Vergleichsart (Offline/Online, Offline/Offline).

Abhängig davon, in welcher Tiefe Sie die Objekte vergleichen möchten, können Sie zwischen den folgenden Vergleichsstufen wählen:

- Vergleichseditor:
  Wenn Sie einen Vergleich an einem Gerät starten, dann öffnet sich der Vergleichseditor, in dem das Vergleichsergebnis angezeigt wird.
- Detailvergleich
  Für einige Vergleichsobjekte können Sie im Vergleichseditor einen Detailvergleich starten, bei dem die verglichenen Objekte nebeneinander in je einer eigenen Instanz des Programmiereditors geöffnet werden. Falls Unterschiede bestehen, werden diese hervorgehoben. Für Bausteine, PLC-Variablen und PLC-Datentypen können Sie den Detailvergleich auch direkt aus der Projektnavigation starten. Beachten Sie jedoch, dass Sie für SCL-Bausteine nur dann einen Detailvergleich starten können, wenn bei beiden Bausteinen die Art der Bausteinschnittstelle identisch ist.

Die folgende Tabelle zeigt eine Übersicht, welche Vergleichsart und -stufe Sie für die einzelnen Objekte einsetzen können:

Objekt	Offline/Online		Offline/Offline	
Vergleichseditor	Detailvergleich	Vergleichseditor	Detailvergleich	
KOP-Baustein	X	X	X	X
FUP-Baustein	X	X	X	X
AWL-Baustein[1]	X	X	X	X
SCL-Baustein	X	X[3 4 7]	X	X[7]
GRAPH-Baustein[2]	X	X[4]	X	X[5]
Globaler Datenbaustein	X	X	X	X
Instanz-Datenbaustein	X	X	X	X
PLC-Variablen	X[6]	X[6]	X	X
PLC-Datentyp	X[4]	X[4]	X	X

Legende:

X: verfügbar
-: nicht verfügbar

[1]: AWL ist für S7-1200 nicht verfügbar

[2]: GRAPH ist für S7-1200 nicht verfügbar

[3]: nicht für S7-1200 kleiner Version 2.0

[4]: nicht für S7-300/400

[5]: nur vergleichbar mit einem GRAPH-Baustein aus der gleichen CPU-Familie

[6]: nur für S7-1500 ab Version 2.5

[7]: die Art der Bausteinschnittstelle muss identisch sein

Hinweis

Beachten Sie folgende Hinweise:

- Für know-how-geschützte Bausteine können Sie keinen Detailvergleich durchführen.
- Wenn im Detailvergleich die Unterschiede nur in den Datentypen von lokalen Variablen bestehen, wobei offline ein Alarmdatentyp (C_ALARM C_ALARM_S C_ALARM_8 C_ALARM_8P C_ALARM_T C_AR_SEND C_NOTIFY C_NOTIFY_8P) und online ein DWORD ist, wird dieser Unterschied nicht als Unterschied gekennzeichnet.
- Für Typen und Kopiervorlagen aus Bibliotheken können Sie keinen Detailvergleich durchführen.

PLC-Programme anhand von Prüfsummen vergleichen

Neben den genannten Vergleichsmöglichkeiten bietet das TIA Portal die Möglichkeit des einfachen Prüfsummenvergleichs. Das System ermittelt automatisch eine Prüfsumme für das komplette PLC-Programm. Anhand der Prüfsumme können Sie leicht feststellen, ob zwei PLC-Programme identisch sind.

Siehe auch: PLC-Programme anhand von Prüfsummen vergleichen (Seite 7921)

Siehe auch

Vergleich von Bausteinen und PLC-Datentypen (Seite 7919)

Vergleich von PLC-Variablen (Seite 7920)

Detailvergleich von Bausteinen durchführen (Seite 7925)

20.1.2 Vergleich von Bausteinen und PLC-Datentypen

Zuordnungskriterien

Beim Vergleich werden die zu vergleichenden Bausteine und PLC-Datentypen anhand folgender Kriterien einander zugeordnet:

- Bausteine:
  - Offline/Online-Vergleich: Adressen, z. B. FB100 oder DB100
  - Offline/Offline-Vergleich: Symbolische Namen der Bausteine
- PLC-Datentypen: PLC-Datentypen werden anhand ihres Namens einander zugeordnet

Vergleich

Beim Vergleich von Bausteinen und PLC-Datentypen werden für bestimmte Eigenschaften Prüfsummen gebildet, die als Basis für den Vergleich dienen. Im Vergleichseditor sind diese Eigenschaften teilweise in Kategorien zusammengefasst. Für CPUs der Baureihe S7-300/400 stehen beim Offline/Online-Vergleich keine Prüfsummen zur Verfügung, hier werden die Zeitstempel ausgewertet. Grundsätzlich werden die Zeitstempel bei einem Offline/Online-Vergleich bei allen CPUs als eigene Kategorie als Information ausgegeben.

Nach dem Ausführen eines Vergleichs können Sie über Aktionen festlegen, wie mit den Unterschieden umgegangen werden soll. Zusätzlich können Sie für Bausteine einen Detailvergleich starten. Dabei werden die verglichenen Versionen eines Bausteins nebeneinander geöffnet und die Unterschiede hervorgehoben. Für Bausteine, PLC-Variablen und PLC-Datentypen können Sie den Detailvergleich auch direkt aus der Projektnavigation starten.

Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 7917)

Vergleich von PLC-Variablen (Seite 7920)

Detailvergleich von Bausteinen durchführen (Seite 7925)

20.1.3 Vergleich von PLC-Variablen

Einführung

Wenn Sie einen Offline/Offline-Vergleich durchführen, werden auch die PLC-Variablentabellen der Geräte im Vergleichseditor angezeigt. Dabei werden die PLC-Variablentabellen anhand ihres Namens einander zugeordnet und Sie können auf einen Blick erkennen, ob die PLC-Variablentabellen in beiden Geräten vorhanden sind. Bei Bedarf können Sie auch einen Detailvergleich für eine PLC-Variablentabelle durchführen. Für Bausteine, PLC-Variablentabellen und PLC-Datentypen können Sie den Detailvergleich auch direkt aus der Projektnavigation starten.

Bei einem Offline/Online-Vergleich erhalten Sie für CPUs der Baureihe S7-1200 und S7-1500 mit einem Firmware-Stand kleiner V2.5 den Gesamtstatus über alle PLC-Variablen angezeigt. Für S7-1500-CPUs ab Firmware 2.5 haben Sie die gleichen Möglichkeiten wie bei einem Offline/Offline-Vergleich.

Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 7917)

Vergleich von Bausteinen und PLC-Datentypen (Seite 7919)

Detailvergleich von Bausteinen durchführen (Seite 7925)

20.1.4 Vergleich von PLC-Programmen anhand von Prüfsummen

Einführung

PLC-Programme werden beim Übersetzen automatisch mit eindeutigen Prüfsummen gekennzeichnet. Anhand der Prüfsumme können Sie Ihr Programm identifizieren und feststellen, ob zwei PLC-Programme identisch sind.

Da die Prüfsumme zusammen mit dem PLC-Programm in die CPU geladen wird, kann sie auch im Servicefall als wichtige Information dienen. Sie können z. B. leicht feststellen, ob das Programm, das aktuell auf der CPU läuft, dasselbe Programm ist, das Sie vor längerer Zeit geladen haben, oder ob es zwischenzeitlich geändert wurde. Das Originalprogramm muss dabei nicht auf Ihrem Programmiergerät (PG) vorhanden sein.

Erzeugung der Prüfsumme

PLC-Programme erhalten beim Übersetzen automatisch eine eindeutige Prüfsumme. Wenn beim nächsten Übersetzungslauf festgestellt wird, dass das PLC-Programm geändert wurde, erhält das Programm eine neue Prüfsumme. Hat sich das PLC-Programm nicht geändert und wird dennoch neu übersetzt, bleibt die Prüfsumme gleich.

Auch wenn Änderungen durchgeführt und anschließend wieder rückgängig gemacht werden, bleibt die Prüfsumme unverändert.

Die Prüfung findet immer statt, wenn das PLC-Programm übersetzt wird, z.B. wenn Sie eine CPU oder einen Bausteinordner selektieren und den Menübefehl "Übersetzen > Software" wählen.

Es werden zwei Prüfsummen generiert:

- Prüfsumme für Software: Es wird eine Prüfsumme über alle Bausteine im Bausteinordner gebildet.
  - Auch Bausteine, die nicht im Programm aufgerufen werden, aber im Bausteinordner vorhanden sind, fließen in die Berechnung ein.
  - Safety-Bausteine und Safety-PLC-Datentypen (UDT) werden dabei nicht berücksichtigt.
  - Änderungen an Beobachtungs- oder Forcetabellen bewirken keine Änderung der Prüfsumme.
  - Datenbausteine, die zur Laufzeit z. B. durch die Anweisungen "WRIT_DBL", "CREAT_DB" und "DELETE_DB" erzeugt oder geändert werden, haben keinen Einfluss auf die Bildung der Prüfsumme.
- Prüfsumme für Textlisten: Es wird eine Prüfsumme für Textlisten im Projekt gebildet.

Laden der Prüfsumme

Die Prüfsumme wird zusammen mit dem PLC-Programm in die CPU geladen und steht im Online-Programm zur Verfügung. Beim Laden aus der CPU wird die Prüfsumme nicht in das Offline-Projekt übernommen, da sie automatisch beim nächsten Übersetzen wieder erzeugt wird.

Auswerten der Prüfsumme

Die Prüfsumme wird in den CPU-Eigenschaften angezeigt (Eigenschaften > Allgemein > Prüfsummen). Von dort können Sie sie manuell in Ihre Unterlagen übernehmen. Um die Prüfsumme zur Laufzeit im Programm auszulesen, steht die Erweiterte Anweisung "GetChecksum" zur Verfügung.

Siehe auch

GetChecksum: Prüfsumme auslesen (Seite 2713)

20.2 Bausteine vergleichen

20.2.1 Bausteine im Vergleichseditor vergleichen

Sie haben folgende Möglichkeiten Bausteine im Vergleichseditor zu vergleichen:

- Offline/Online-Vergleich
  Die Bausteine im Projekt werden mit den Bausteinen des gewählten Geräts verglichen.
- Automatischer Offline/Offline-Vergleich
  Alle Bausteine der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich
  Die ausgewählten Bausteine der Geräte werden offline verglichen.

Offline/Online-Vergleich von Bausteinen durchführen

Um einen Offline/Online-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Online-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".
3. Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".
   Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.
4. Öffnen Sie den Ordner "Programmbausteine".
   Den Status können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Beachten Sie dabei jedoch, dass Sie in einer Synchronisationsaktion nur Aktionen in eine Richtung ausführen können.

Automatischen Offline/Offline-Vergleich von Bausteinen durchführen

Um einen automatischen Offline/Offline-Vergleich von Bausteinen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.

3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "Programmbausteine".
   Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Manueller Offline/Offline-Vergleich von Bausteinen durchführen

Um einen manuellen Offline/Offline-Vergleich von Bausteinen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die Objekte, die Sie miteinander vergleichen möchten.
   Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich von Bausteinen durchführen (Seite 7925)

PLC-Variablen vergleichen (Seite 7951)

PLC-Datentypen vergleichen (Seite 7955)

20.2.2 Detailvergleich von Bausteinen durchführen

20.2.2.1 Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten

Sie können für Bausteine einen Detailvergleich starten. Dabei werden die verglichenen Versionen eines Bausteins nebeneinander geöffnet und die Unterschiede werden hervorgehoben. Sie haben die folgenden Möglichkeiten, einen Detailvergleich für Bausteine zu starten:

- Detailvergleich über den Vergleichseditor starten
  Der Detailvergleich für Bausteine ist über den Vergleichseditor sowohl für Offline/Online als auch für Offline/Offline möglich.
- Detailvergleich in der Projektnavigation starten
  Der Detailvergleich für Bausteine ist über die Projektnavigation sowohl für Offline/Online als auch für Offline/Offline möglich. Für den Offline/Online-Vergleich ist eine bestehende Online-Verbindung notwendig.
- Detailvergleich innerhalb des Programmiereditors starten
  Hier erfolgt ein Vergleich des geöffneten Bausteins mit seiner Online-Version.

Hinweis

Beachten Sie folgende Hinweise:

- Für Bausteine, die in der Programmiersprache SCL erstellt sind, steht Ihnen der Detailvergleich für CPUs der S7-1200-Baureihe mit einer Version kleiner als 2.0 nicht zur Verfügung.
- Sie können einen Detailvergleich für zwei SCL-Bausteine nur dann starten, wenn beide Bausteine die gleiche Schnittstellenart verwenden, also entweder beide tabellarisch oder beide textuell.
- S7-1500: Durch das gemeinsame parallele Arbeiten auf einer CPU ist es möglich, dass ein weiterer Anwender einen Ladevorgang auf der CPU ausführt. Falls durch diesen Ladevorgang der gerade verglichene Baustein geändert oder gelöscht wird, wird der Detailvergleich beendet und Sie erhalten eine Meldung. Starten Sie den Detailvergleich in diesem Fall bei Bedarf erneut.

Detailvergleich über den Vergleichseditor starten

Um für einen Baustein einen Detailvergleich über den Vergleichseditor zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Offline/Online- oder einen Offline/Offline-Vergleich durch.
   Der Vergleichseditor wird geöffnet.
2. Selektieren Sie im Vergleichseditor den Baustein, für den Sie einen Detailvergleich durchführen möchten.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten" oder wählen Sie im Kontextmenü den Befehl "Detailvergleich starten".

Detailvergleich in der Projektnavigation starten

Um für einen Baustein einen Offline/Offline-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie vergleichen möchten. Dies kann auch ein Baustein aus einem Referenzprojekt sein.
2. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Als linkes Objekt auswählen".
3. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie mit dem Baustein vergleichen möchten, den Sie zuvor als linkes Objekt ausgewählt haben.
4. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit <ausgewähltes Objekt> vergleichen". "<ausgewähltes Objekt>" steht für das linke Vergleichsobjekt.

Um für einen Baustein einen Offline/Online-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, auf dem der Baustein vorhanden ist.
2. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie mit seinem Online-Objekt vergleichen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit dem Online-Objekt vergleichen".

Detailvergleich innerhalb des Programmiereditors starten

Für die Vergleichsart Offline/Online können Sie den Detailvergleich direkt im Programmiereditor starten. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie einen Detailvergleich durchführen möchten.
2. Stellen Sie eine Online-Verbindung her.
   Siehe auch: Online verbinden und Online-Verbindung trennen

   Hinweis

   Beachten Sie, dass der Baustein online vorhanden sein muss, damit Sie den Detailvergleich für den Baustein innerhalb des Programmiereditors starten können.

3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich".
4. Bestätigen Sie den Dialog zum Schließen des Bausteins mit "Ja".

Ergebnis

Für die miteinander verglichenen Versionen des Bausteins wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Darstellung des Vergleichsergebnisses für KOP/FUP (Seite 7930)

Darstellung des Vergleichsergebnisses für AWL (Seite 7934)

Darstellung des Vergleichsergebnisses für SCL (Seite 7937)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

20.2.2.2 Detailvergleich für GRAPH-Bausteine starten

Sie können für GRAPH-Bausteine einen Detailvergleich starten. Dabei werden die verglichenen Versionen eines Bausteins nebeneinander geöffnet und die Unterschiede werden hervorgehoben. Sie haben die folgenden Möglichkeiten, einen Detailvergleich für GRAPH-Bausteine zu starten:

- Detailvergleich über den Vergleichseditor starten
  Der Detailvergleich für Bausteine ist über den Vergleichseditor sowohl für Online/Offline als auch für Offline/Offline möglich.
- Detailvergleich in der Projektnavigation starten
  Der Detailvergleich für Bausteine ist über die Projektnavigation sowohl für Online/Offline als auch für Offline/Offline möglich. Für den Online/Offline-Vergleich ist eine bestehende Online-Verbindung notwendig.
- Detailvergleich innerhalb des Programmiereditors starten
  Hier erfolgt ein Vergleich des geöffneten Bausteins mit seiner Online-Version.

Für GRAPH-Bausteine stehen Ihnen die folgenden Vergleichsmodi zur Verfügung:

- Kette vergleichen
  In diesem Modus werden die kompletten Ablaufketten miteinander verglichen. Ein Detailvergleich für einen GRAPH-Baustein beginnt immer im Vergleichsmodus "Kette vergleichen". Das bedeutet, der Vergleich beginnt am Anfang der Ablaufkette und die Unterschiede zwischen den Ablaufketten werden angezeigt. Wenn strukturelle Unterschiede zwischen den Ablaufketten bestehen, werden nur die Vergleichsergebnisse bis zum ersten Strukturunterschied angezeigt.
- Selektion vergleichen
  Der Vergleichsmodus "Selektion vergleichen" ermöglicht Ihnen, beliebige Bereiche miteinander zu vergleichen. Dadurch können Sie auch Teile der Ablaufkette miteinander vergleichen, die im Ablauf hinter einem strukturellen Unterschied liegen.

Sie können beliebig zwischen den beiden Vergleichsmodi wechseln.

Hinweis

S7-1500: Durch das gemeinsame parallele Arbeiten auf einer CPU ist es möglich, dass ein weiterer Anwender einen Ladevorgang auf der CPU ausführt. Falls durch diesen Ladevorgang der gerade verglichene Baustein geändert oder gelöscht wird, wird der Detailvergleich beendet und Sie erhalten eine Meldung. Starten Sie den Detailvergleich in diesem Fall bei Bedarf erneut.

Detailvergleich über den Vergleichseditor starten

Um einen Detailvergleich zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Online/Offline- oder einen Offline/Offline-Vergleich zwischen zwei Geräten durch.
   Der Vergleichseditor wird geöffnet.
2. Selektieren Sie im Vergleichseditor den GRAPH-Baustein, für den Sie einen Detailvergleich durchführen möchten.

   Hinweis

   Bei einem Offline/Offline-Vergleich steht Ihnen zusätzlich der manuelle Vergleich zur Verfügung. Dadurch können Sie im Vergleichseditor beliebige GRAPH-Bausteine für den Vergleich selektieren.

3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".
   Für die miteinander verglichenen Versionen der GRAPH-Bausteine wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet. Falls Unterschiede bestehen, werden diese gekennzeichnet. Der Vergleich erfolgt im Modus "Kette vergleichen".

Detailvergleich in der Projektnavigation starten

Um für einen GRAPH-Baustein einen Offline/Offline-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie vergleichen möchten. Dies kann auch ein Baustein aus einem Referenzprojekt sein.
2. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Als linkes Objekt auswählen".
3. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie mit dem Baustein vergleichen möchten, den Sie zuvor als linkes Objekt ausgewählt haben.
4. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit <ausgewähltes Objekt> vergleichen". "<ausgewähltes Objekt>" steht für das linke Vergleichsobjekt.
   Für die miteinander verglichenen Versionen der GRAPH-Bausteine wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet. Falls Unterschiede bestehen, werden diese gekennzeichnet. Der Vergleich erfolgt im Modus "Kette vergleichen".

Um für einen GRAPH-Baustein einen Online/Offline-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, auf dem der Baustein vorhanden ist.
2. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie mit seinem Online-Objekt vergleichen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit dem Online-Objekt vergleichen".
   Für die miteinander verglichenen Versionen der GRAPH-Bausteine wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet. Falls Unterschiede bestehen, werden diese gekennzeichnet. Der Vergleich erfolgt im Modus "Kette vergleichen".

Detailvergleich innerhalb des Programmiereditors starten

Für die Vergleichsart Online/Offline können Sie den Detailvergleich direkt im Programmiereditor starten. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie den GRAPH-Baustein, für den Sie einen Detailvergleich durchführen möchten.
2. Stellen Sie eine Online-Verbindung her.
   Siehe auch: Online verbinden und Online-Verbindung trennen

   Hinweis

   Beachten Sie, dass der Baustein online vorhanden sein muss, damit Sie den Detailvergleich für den Baustein innerhalb des Programmiereditors starten können.

3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich".
4. Bestätigen Sie den Dialog zum Schließen des Bausteins mit "Ja".
   Für die miteinander verglichenen Versionen der GRAPH-Bausteine wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet. Falls Unterschiede bestehen, werden diese gekennzeichnet. Der Vergleich erfolgt im Modus "Kette vergleichen".

Bereiche vergleichen

Um bestimmte Bereiche innerhalb der angezeigten Ablaufketten miteinander zu vergleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in jeder Ablaufkette den Schritt, ab dem der Vergleich starten soll.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Vergleichsmodus".
   Der Vergleichsmodus ändert sich in "Selektion vergleichen" und das Vergleichsergebnis wird aktualisiert. Die Ablaufketten werden ab den selektierten Schritten miteinander verglichen. Die verglichenen Teilketten werden hervorgehoben.
3. Um andere Bereiche miteinander zu vergleichen, selektieren Sie die Schritte, die als Startpunkt für den Vergleich dienen sollen.
4. Klicken Sie in der Funktionsleiste auf "Vergleichsergebnisse aktualisieren".

   Hinweis

   Wenn Sie den Vergleichsmodus wechseln, werden dabei auch gleichzeitig die Vergleichsergebnisse aktualisiert. Bleiben Sie innerhalb des Modus "Selektion vergleichen", müssen Sie nach jedem Ändern des Startpunkts die Vergleichsergebnisse manuell aktualisieren.

5. Um wieder die kompletten Ablaufketten miteinander zu vergleichen, klicken Sie in der Funktionsleiste auf "Vergleichsmodus".
   Mit jedem Klick auf die Schaltfläche "Vergleichsmodus" ändern Sie den Vergleichsmodus. Sie können auch über den Pfeil die Klappliste öffnen und den gewünschten Modus aus der Liste auswählen.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Darstellung des Vergleichsergebnisses für GRAPH (Seite 7940)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

20.2.2.3 Darstellung des Vergleichsergebnisses

Darstellung des Vergleichsergebnisses für KOP/FUP

Einführung

Über den Detailvergleich können Sie genau die Stellen identifizieren, die sich in den verglichenen Versionen eines Bausteins unterscheiden. Damit Sie diese Stellen möglichst schnell finden, gelten folgende Kennzeichnungen:

- Die Zeilen, in denen es Unterschiede gibt, werden grau hinterlegt.
- Die unterschiedlichen Operanden und Anweisungen werden grün hinterlegt.
- Bei einer unterschiedlichen Anzahl von Netzwerken werden Pseudonetzwerke eingefügt, damit eine synchronisierte Darstellung identischer Netzwerke möglich ist. Diese Pseudonetzwerke sind grau hinterlegt und enthalten in der Titelleiste des Netzwerks den Text "Ein entsprechendes Netzwerk wurde nicht gefunden.". Pseudonetzwerke können nicht bearbeitet werden.

Hinweis

Wenn zwischen zwei Netzwerken ein Unterschied von mehr als 50 Prozent besteht, dann werden diese Netzwerke als nicht zusammengehörend betrachtet und es erfolgt kein Vergleich zwischen diesen Netzwerken. In diesem Fall werden ebenfalls Pseudonetzwerke eingefügt, bis ein passendes Netzwerk gefunden wird.

Um die Übersichtlichkeit zu bewahren, werden nicht alle Unterschiede hervorgehoben, sondern jeweils der erste Unterschied einer Operation. Wenn z. B. bei einer Anweisung mit mehreren Eingängen alle Eingänge in der Offline- und Online-Version des Bausteins unterschiedlich sind, wird nur der erste Eingang als Unterschied hervorgehoben. Sie können diesen Unterschied beheben und die Vergleichsliste aktualisieren. Anschließend wird der nächste Eingang als Unterschied hervorgehoben.

Die Anzahl der hervorgehobenen Unterschiede innerhalb eines Netzwerks ist daher abhängig von der Anzahl der vorhandenen Anweisungen.

Aufbau des Detailvergleichs

Das folgende Bild zeigt ein Beispiel für den Offline/Online-Detailvergleich für die Programmiersprache KOP:

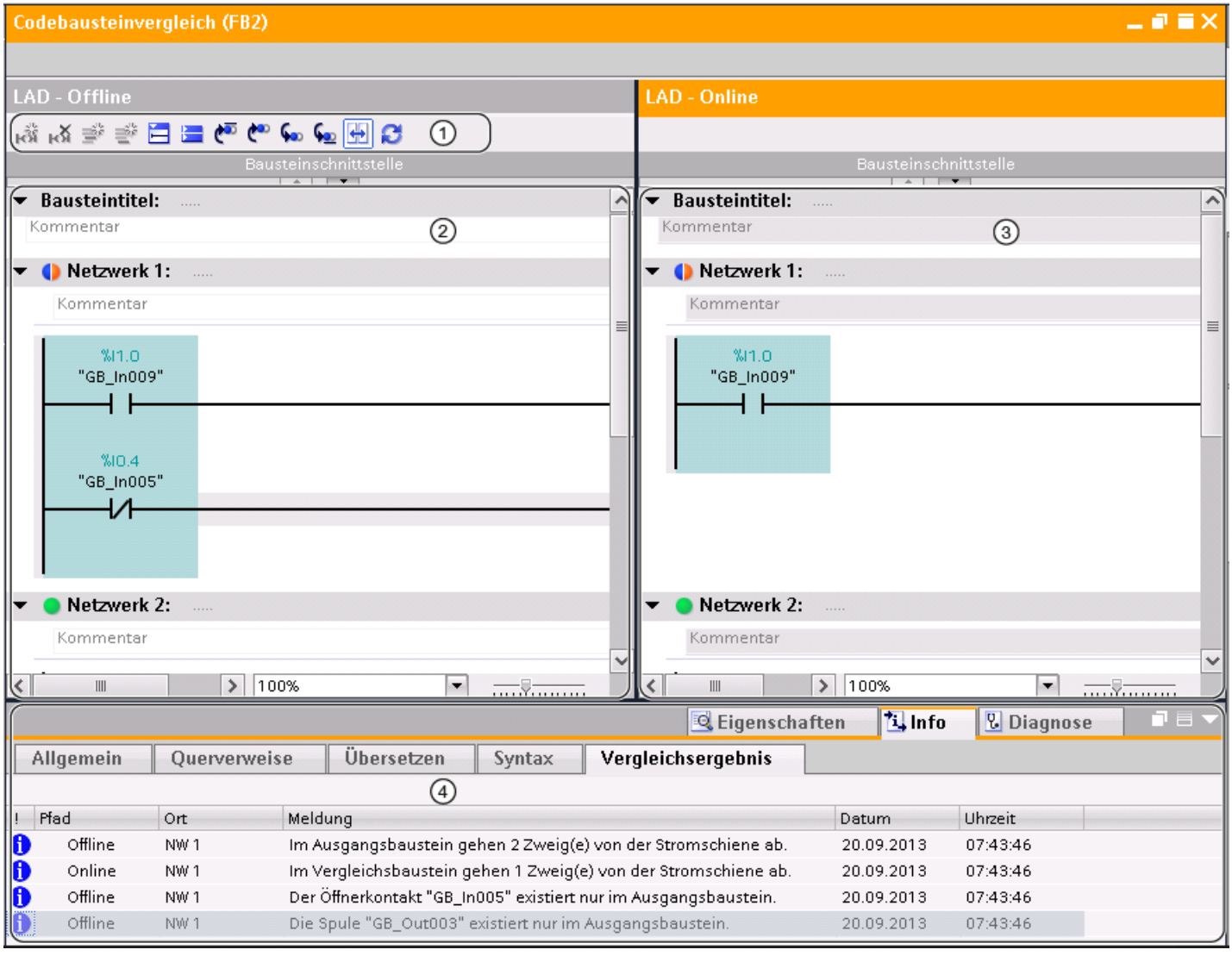

① Funktionsleiste des Detailvergleichs für KOP

② Ausgangsbaustein

③ Vergleichsbaustein

④ Vergleichsergebnis im Inspektorfenster

Das folgende Bild zeigt ein Beispiel für den Offline/Online-Detailvergleich für die Programmiersprache FUP:

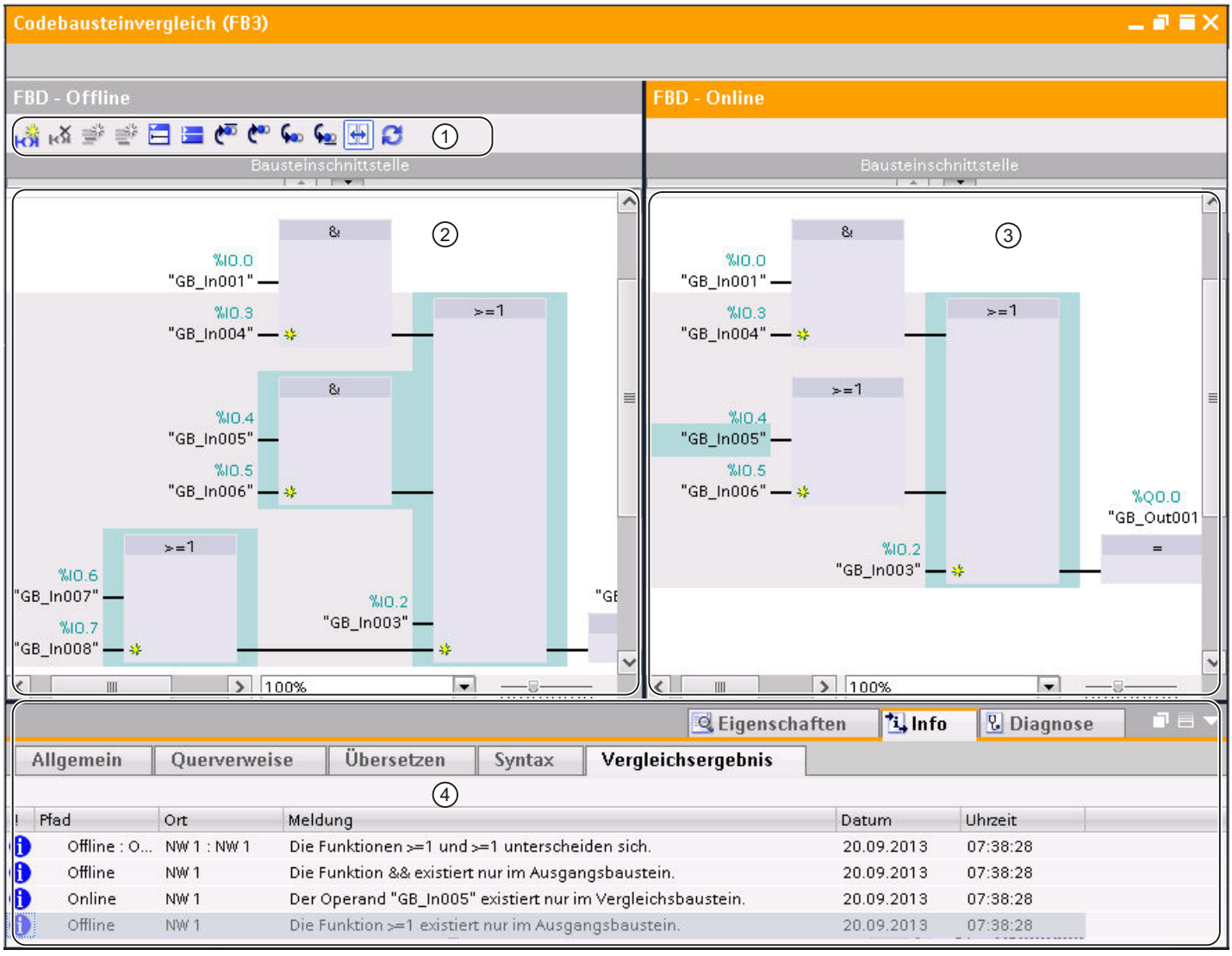

① Funktionsleiste des Detailvergleichs für FUP

② Ausgangsbaustein

③ Vergleichsbaustein

④ Vergleichsergebnis im Inspektorfenster

Hinweis

Die Anzeige der symbolischen Namen der Online-Version eines Bausteins ist nur für S7-1200 und S7-1500 möglich.

Funktionsleiste des Detailvergleichs

Über die Funktionsleiste haben Sie Zugriff auf folgende Funktionen:

- Allgemeine Funktionen
  - Netzwerk einfügen
  - Netzwerk löschen
  - Zeile einfügen
  - Zeile hinzufügen
  - Alle Netzwerke öffnen
  - Alle Netzwerke schließen
- Vergleichsspezifische Funktionen
  - Auf erstem Unterschied positionieren
  - Auf vorherigem Unterschied positionieren
  - Auf nächstem Unterschied positionieren
  - Auf letztem Unterschied positionieren
  - Bildlauf der Editoren synchronisieren
  - Vergleichsergebnisse aktualisieren

Ausgangsbaustein

Der Ausgangsbaustein wird im linken Fenster angezeigt. Bei einem Offline/Online-Vergleich ist der Ausgangsbaustein die Offline-Version des Bausteins.

Vergleichsbaustein

Der Vergleichsbaustein wird im rechten Fenster angezeigt. Bei einem Offline/Online-Vergleich ist der Vergleichsbaustein die Online-Version des Bausteins.

Vergleichsergebnis im Inspektorfenster

Im Register "Info > Vergleichsergebnis" des Inspektorfensters wird eine tabellarische Übersicht der Unterschiede angezeigt. Klicken Sie doppelt auf eine Zeile, um zum entsprechenden Unterschied im Baustein zu navigieren.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

Darstellung des Vergleichsergebnisses für AWL

Einführung

Über den Detailvergleich können Sie genau die Stellen identifizieren, die sich in den verglichenen Versionen eines Bausteins unterscheiden. Damit Sie diese Stellen möglichst schnell finden, gelten folgende Kennzeichnungen:

- Die Zeilen, in denen es Unterschiede gibt, werden grau hinterlegt.
- Die unterschiedlichen Operanden und Anweisungen werden grün hinterlegt.
- Bei einer unterschiedlichen Anzahl von Netzwerken werden Pseudonetzwerke eingefügt, damit eine synchronisierte Darstellung identischer Netzwerke möglich ist. Diese Pseudonetzwerke sind grau hinterlegt und enthalten in der Titelleiste des Netzwerks den Text "Ein entsprechendes Netzwerk wurde nicht gefunden.". Pseudonetzwerke können nicht bearbeitet werden.

Aufbau des Detailvergleichs

Das folgende Bild zeigt ein Beispiel für den Online/Offline-Detailvergleich für die Programmiersprache AWL:

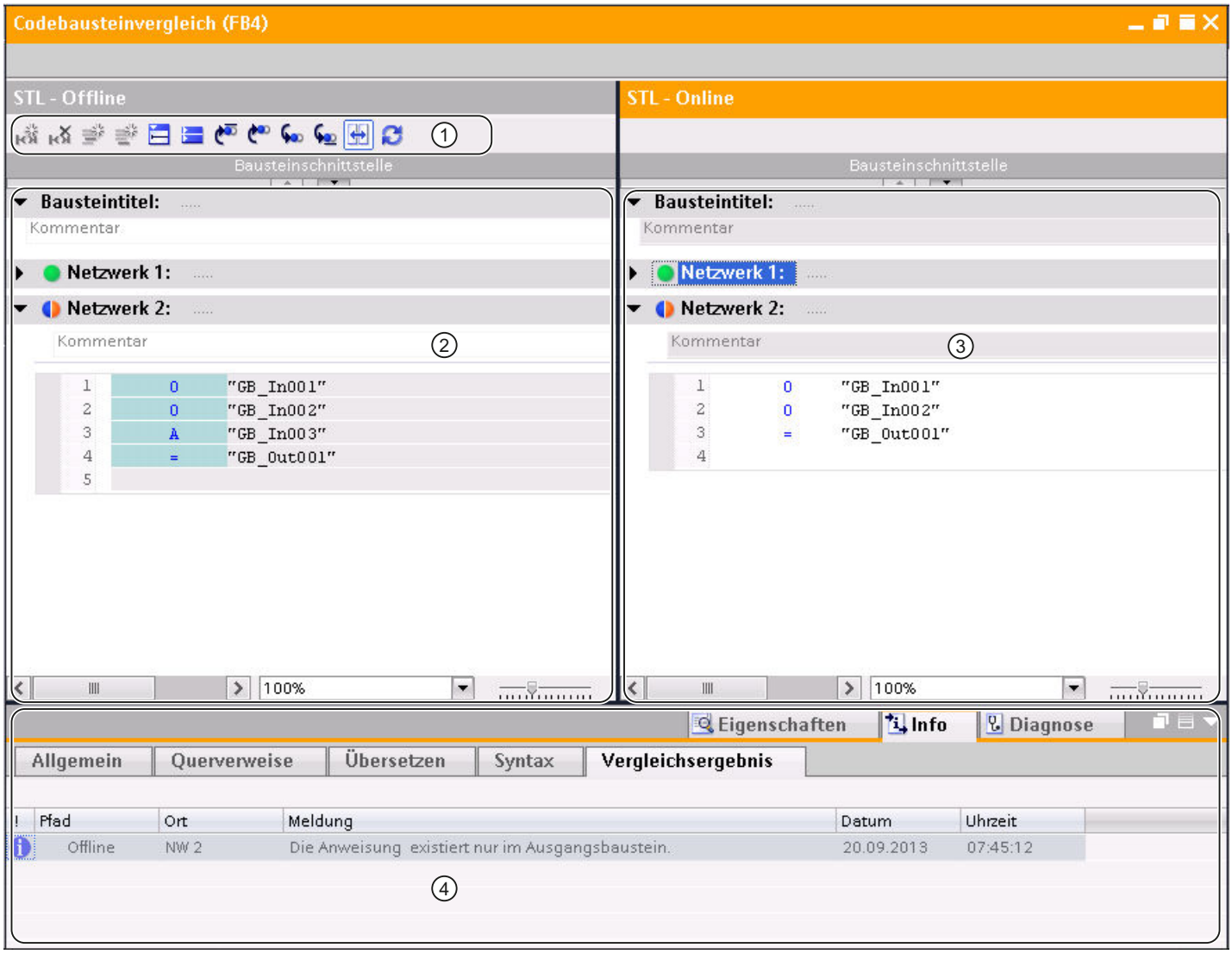

① Funktionsleiste des Detailvergleichs für AWL

② Ausgangsbaustein

③ Vergleichsbaustein

④ Vergleichsergebnis im Inspektorfenster

Hinweis

Die Anzeige der symbolischen Namen der Online-Version des Bausteins ist nur für S7-1500 möglich.

Funktionsleiste des Detailvergleichs

Über die Funktionsleiste haben Sie Zugriff auf folgende Funktionen:

- Allgemeine Funktionen
  - Netzwerk einfügen
  - Netzwerk löschen
  - Zeile einfügen
  - Zeile hinzufügen
  - Alle Netzwerke öffnen
  - Alle Netzwerke schließen
- Vergleichsspezifische Funktionen
  - Auf erstem Unterschied positionieren
  - Auf vorherigem Unterschied positionieren
  - Auf nächstem Unterschied positionieren
  - Auf letztem Unterschied positionieren
  - Bildlauf der Editoren synchronisieren
  - Vergleichsergebnisse aktualisieren

Ausgangsbaustein

Der Ausgangsbaustein wird im linken Fenster angezeigt. Bei einem Online/Offline-Vergleich ist der Ausgangsbaustein die Offline-Version des Bausteins.

Vergleichsbaustein

Der Vergleichsbaustein wird im rechten Fenster angezeigt. Bei einem Online/Offline-Vergleich ist der Vergleichsbaustein die Online-Version des Bausteins.

Vergleichsergebnis im Inspektorfenster

Im Register "Info > Vergleichsergebnis" des Inspektorfensters wird eine tabellarische Übersicht der Unterschiede angezeigt. Klicken Sie doppelt auf eine Zeile, um zum entsprechenden Unterschied im Baustein zu navigieren.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

Darstellung des Vergleichsergebnisses für SCL

Einführung

Über den Detailvergleich können Sie genau die Stellen identifizieren, die sich in den verglichenen Versionen eines Bausteins unterscheiden. Damit Sie diese Stellen möglichst schnell finden, gelten folgende Kennzeichnungen:

- Die Zeilen, in denen es Unterschiede gibt, werden grau hinterlegt.
- Die unterschiedlichen Operanden und Anweisungen werden grün hinterlegt.

Hinweis

Der Offline/Online-Detailvergleich ist für SCL für die CPU-Familien S7-300/400 und für S7-1200 mit einer Version kleiner 2.0 nicht verfügbar.

Aufbau des Detailvergleichs

Das folgende Bild zeigt ein Beispiel für den Offline/Online-Detailvergleich für die Programmiersprache SCL:

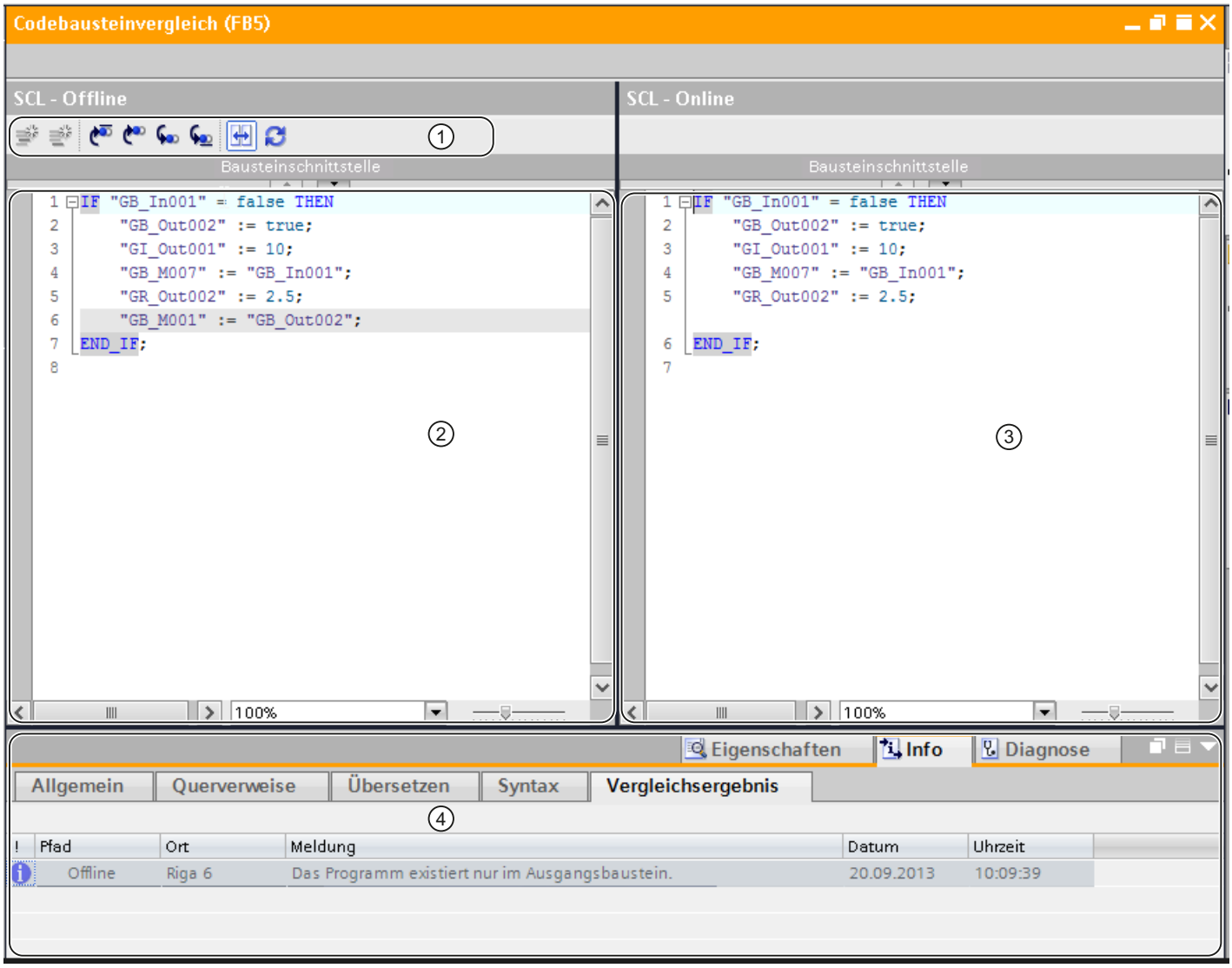

① Funktionsleiste des Detailvergleichs für SCL

② Ausgangsbaustein

③ Vergleichsbaustein

④ Vergleichsergebnis im Inspektorfenster

Hinweis

Die Anzeige der symbolischen Namen der Online-Version des Bausteins ist nur für S7-1200 und S7-1500 möglich.

Funktionsleiste des Detailvergleichs

Über die Funktionsleiste haben Sie Zugriff auf folgende Funktionen:

- Allgemeine Funktionen
  - Zeile einfügen
  - Zeile hinzufügen
- Vergleichsspezifische Funktionen
  - Auf erstem Unterschied positionieren
  - Auf vorherigem Unterschied positionieren
  - Auf nächstem Unterschied positionieren
  - Auf letztem Unterschied positionieren
  - Bildlauf der Editoren synchronisieren
  - Vergleichsergebnisse aktualisieren

Ausgangsbaustein

Der Ausgangsbaustein wird im linken Fenster angezeigt. Bei einem Offline/Online-Vergleich ist der Ausgangsbaustein die Offline-Version des Bausteins.

Vergleichsbaustein

Der Vergleichsbaustein wird im rechten Fenster angezeigt. Bei einem Offline/Online-Vergleich ist der Vergleichsbaustein die Online-Version des Bausteins.

Vergleichsergebnis im Inspektorfenster

Im Register "Info > Vergleichsergebnis" des Inspektorfensters wird eine tabellarische Übersicht der Unterschiede angezeigt. Klicken Sie doppelt auf eine Zeile, um zum entsprechenden Unterschied im Baustein zu navigieren.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

Darstellung des Vergleichsergebnisses für GRAPH

Einführung

Über den Detailvergleich können Sie genau die Stellen identifizieren, die sich in den verglichenen Versionen eines Bausteins unterscheiden. Wenn Sie für einen GRAPH-Baustein einen Detailvergleich starten, wird zunächst die Navigation geöffnet. Um zwischen der Navigation und der aktuell eingestellten Ansicht zu wechseln, können Sie die Fensterteiler verwenden. Über die Funktionsleiste des Detailvergleichs können Sie andere Ansichten wählen.

Das Ergebnis des Vergleichs wird mit den Vergleichssymbolen gekennzeichnet.

Siehe auch: Übersicht über den Vergleichseditor

Aufbau des Detailvergleichs

Das folgende Bild zeigt ein Beispiel für die Ansicht der Navigation bei einem Online/Offline-Detailvergleich für die Programmiersprache GRAPH:

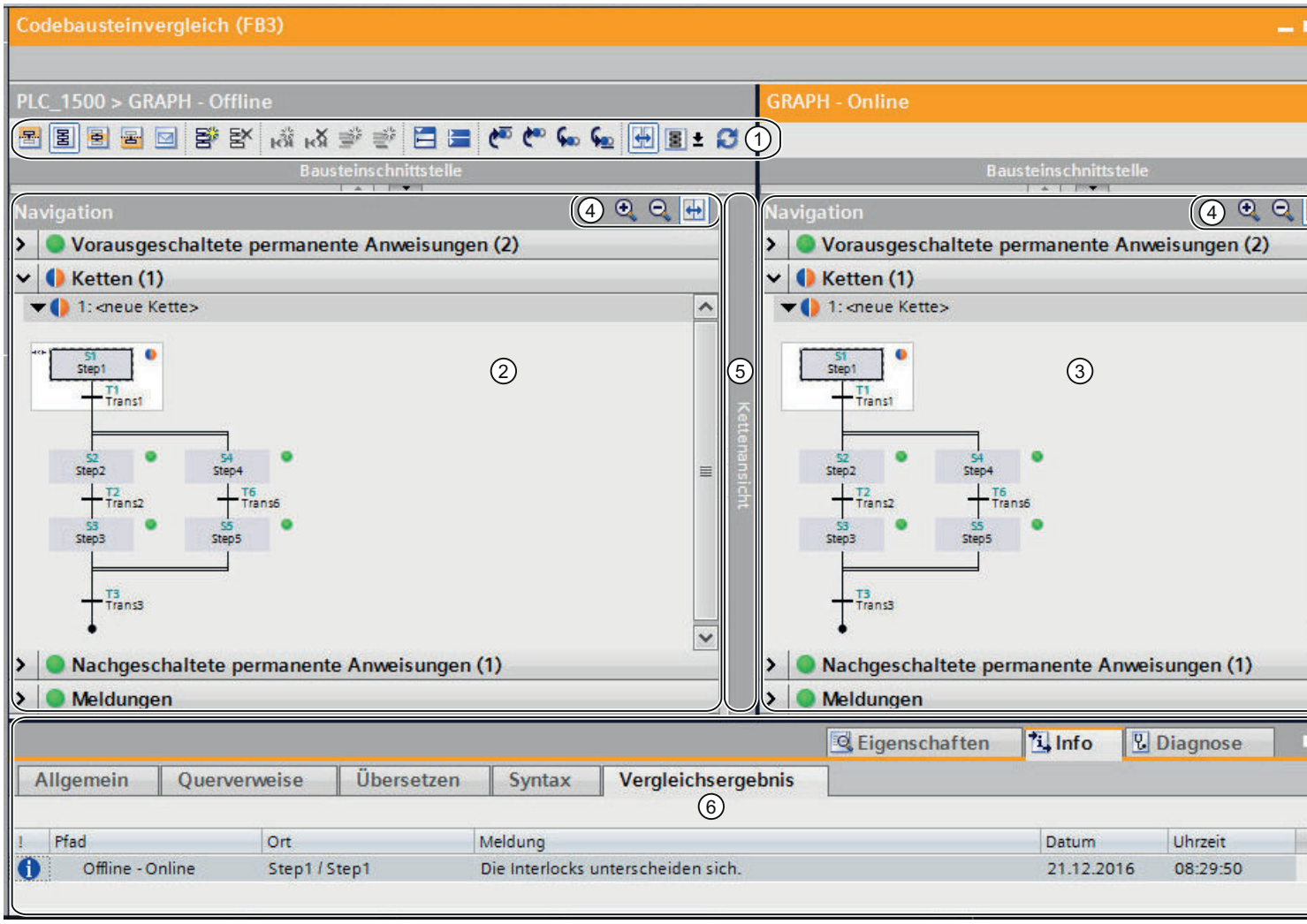

① Funktionsleiste des Detailvergleichs für GRAPH

② Ausgangsbaustein

③ Vergleichsbaustein

④ Funktionsleiste der Navigation

⑤ Fensterteiler

⑥ Vergleichsergebnis im Inspektorfenster

Das folgende Bild zeigt ein Beispiel für die Kettenansicht bei einem Online/Offline-Detailvergleich für die Programmiersprache GRAPH:

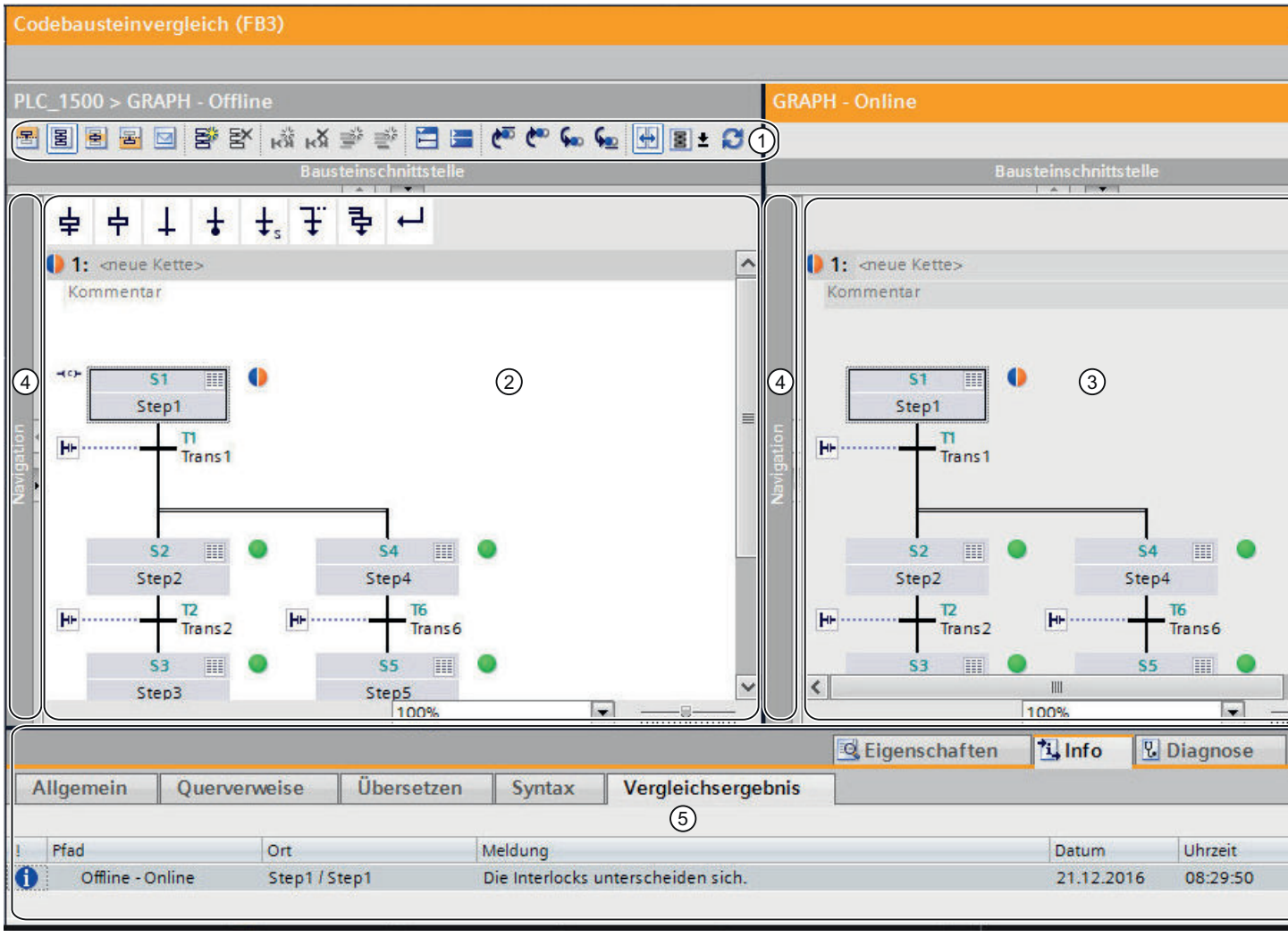

① Funktionsleiste des Detailvergleichs für GRAPH

② Ausgangsbaustein

③ Vergleichsbaustein

④ Fensterteiler

⑤ Vergleichsergebnis im Inspektorfenster

Hinweis

Falls strukturelle Unterschiede zwischen den Bausteinen vorliegen, werden in der Kettenansicht nur die Vergleichsergebnisse bis zum ersten Strukturunterschied angezeigt.

Das folgende Bild zeigt ein Beispiel für die Einzelschrittansicht bei einem Online/Offline-Detailvergleich für die Programmiersprache GRAPH:

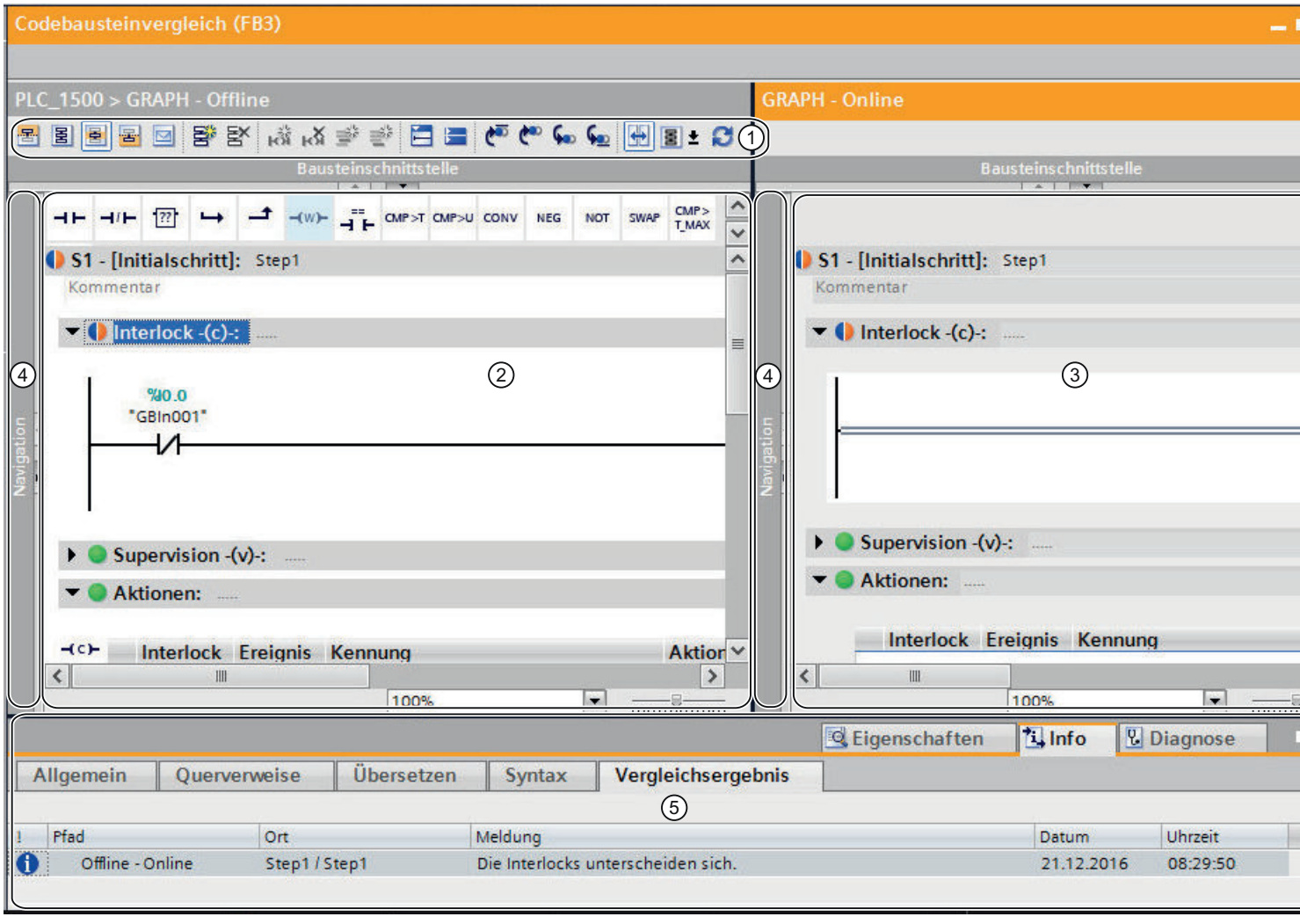

① Funktionsleiste des Detailvergleichs für GRAPH

② Ausgangsbaustein

③ Vergleichsbaustein

④ Fensterteiler

⑤ Vergleichsergebnis im Inspektorfenster

Hinweis

Das Ergebnis des Vergleichs bezieht sich auf das komplette Netzwerk. Unterschiede innerhalb der Netzwerke werden nicht gekennzeichnet.

Das folgende Bild zeigt ein Beispiel für die Ansicht von permanenten Anweisungen bei einem Online/Offline-Detailvergleich für die Programmiersprache GRAPH:

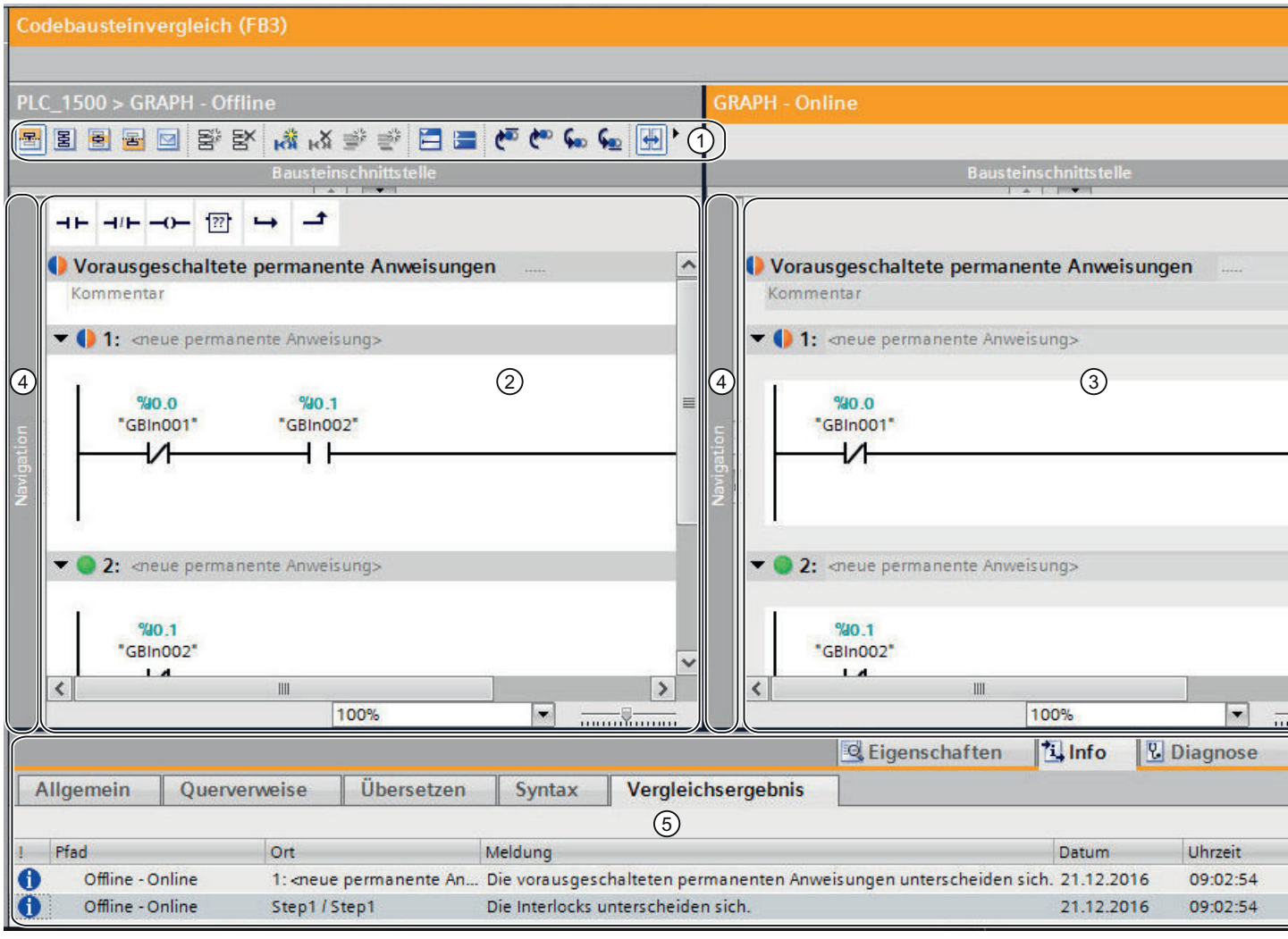

① Funktionsleiste des Detailvergleichs für GRAPH

② Ausgangsbaustein

③ Vergleichsbaustein

④ Fensterteiler

⑤ Vergleichsergebnis im Inspektorfenster

Hinweis

Das Ergebnis des Vergleichs bezieht sich auf das komplette Netzwerk. Unterschiede innerhalb der Netzwerke werden nicht gekennzeichnet.

Das folgende Bild zeigt ein Beispiel für die Meldungsansicht bei einem Offline/Offline-Detailvergleich für die Programmiersprache GRAPH:

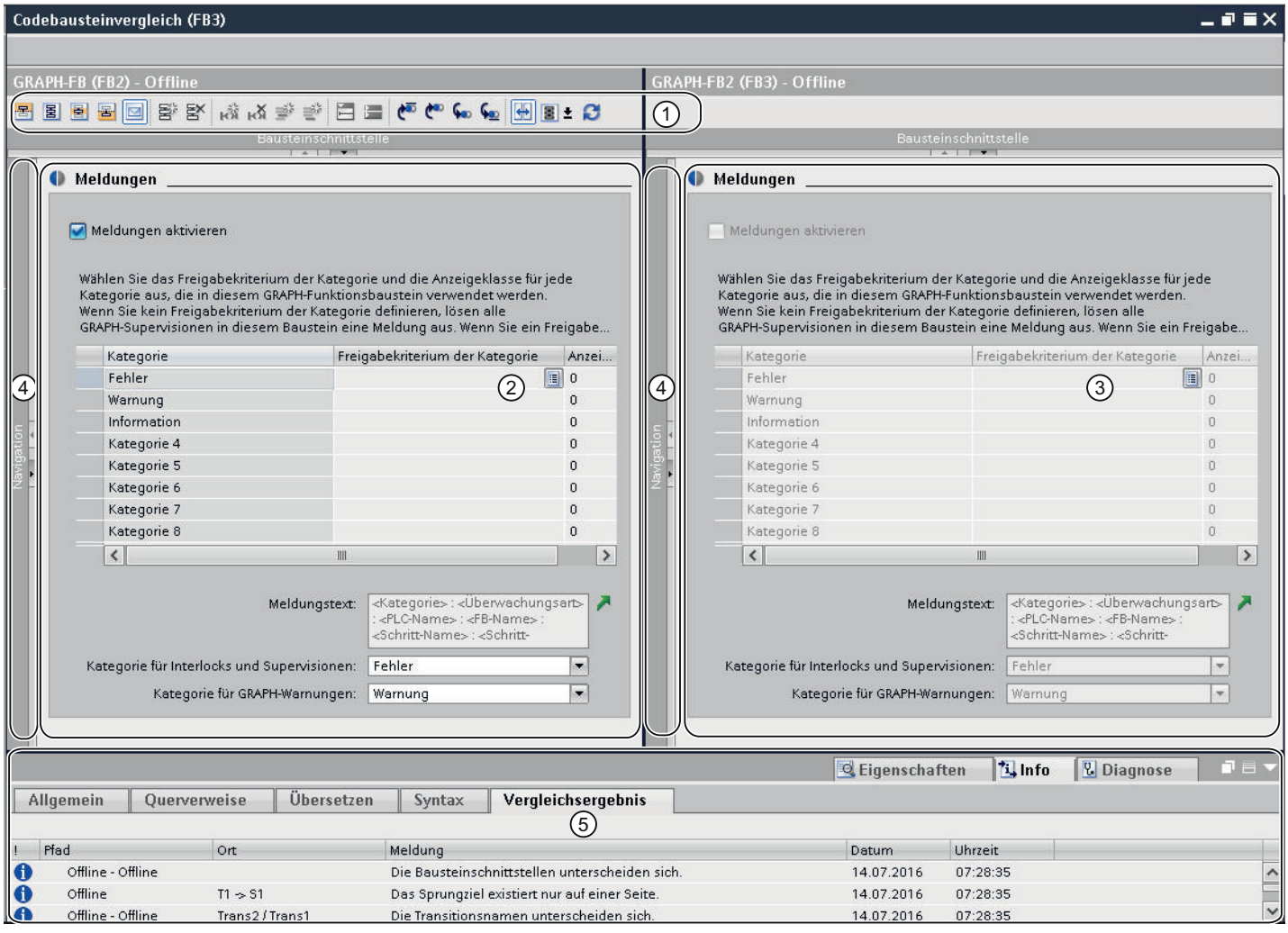

① Funktionsleiste des Detailvergleichs für GRAPH

② Ausgangsbaustein

③ Vergleichsbaustein

④ Fensterteiler

⑤ Vergleichsergebnis im Inspektorfenster

Funktionsleisten

Über die Funktionsleiste des Detailvergleichs haben Sie Zugriff auf folgende Funktionen:

- Allgemeine Funktionen
  - Zu den vorausgeschalteten permanenten Anweisungen wechseln
  - Zur Kettenansicht wechseln
  - Zur Einzelschrittansicht wechseln
  - Zu den nachgeschalteten permanenten Anweisungen wechseln
  - Zur Meldungsansicht wechseln
  - Kette einfügen
  - Kette löschen
  - Netzwerk einfügen
  - Netzwerk löschen
  - Zeile einfügen
  - Zeile hinzufügen
  - Alle Netzwerke öffnen
  - Alle Netzwerke schließen
- Vergleichsspezifische Funktionen
  - Auf erstem Unterschied positionieren
  - Auf vorherigem Unterschied positionieren
  - Auf nächstem Unterschied positionieren
  - Auf letztem Unterschied positionieren
  - Bildlauf der Editoren synchronisieren
  - Vergleichsergebnisse aktualisieren
  - Vergleichsmodus auswählen

Die Navigation verfügt über eine eigene Funktionsleiste mit den folgenden Funktionen:

- Elemente innerhalb der Navigation vergrößern oder verkleinern
- Navigation synchronisieren

Ausgangsbaustein

Der Ausgangsbaustein wird im linken Fenster angezeigt. Bei einem Online/Offline-Vergleich ist der Ausgangsbaustein die Offline-Version des Bausteins.

Vergleichsbaustein

Der Vergleichsbaustein wird im rechten Fenster angezeigt. Bei einem Online/Offline-Vergleich ist der Vergleichsbaustein die Online-Version des Bausteins.

Fensterteiler

Sie können auf die Fensterteiler klicken, um schnell zwischen der Navigation und der aktuellen Ansicht umzuschalten.

Vergleichsergebnis im Inspektorfenster

Im Register "Info > Vergleichsergebnis" des Inspektorfensters wird eine tabellarische Übersicht der Unterschiede angezeigt. Klicken Sie doppelt auf eine Zeile, um zum entsprechenden Unterschied im Baustein zu navigieren.

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Detailvergleich für GRAPH-Bausteine starten (Seite 7927)

Im Detailvergleich navigieren (Seite 7946)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

20.2.2.4 Im Detailvergleich navigieren

Voraussetzung

Ein Detailvergleich wurde durchgeführt.

Zu den Unterschieden navigieren

Um zu einem Unterschied zwischen den beiden Bausteinen zu navigieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Ergebnisliste des Detailvergleichs im Inspektorfenster unter "Info > Vergleichsergebnis".
2. Doppelklicken Sie auf einen Unterschied.
   Der entsprechende Unterschied wird in beiden Editoren selektiert.

Oder:

1. Klicken Sie in der Funktionsleiste auf eine der folgenden Navigationsschaltflächen:
   - Auf erstem Unterschied positionieren
     Navigiert zum ersten Unterschied im Baustein und zeigt den Unterschied in beiden Editoren an.
   - Auf vorherigem Unterschied positionieren
     Navigiert ausgehend von der aktuellen Position zum vorherigen Unterschied und zeigt den Unterschied in beiden Editoren an.
   - Auf nächstem Unterschied positionieren
     Navigiert ausgehend von der aktuellen Position zum nächsten Unterschied und zeigt den Unterschied in beiden Editoren an.
   - Auf letztem Unterschied positionieren
     Navigiert zum letzten Unterschied im Baustein und zeigt den Unterschied in beiden Editoren an.

Synchronisation des vertikalen Bildlaufs zwischen den Editoren aus- und einschalten

Damit bei einem vertikalen Bildlauf die sich entsprechenden Netzwerke parallel sichtbar bleiben, erfolgt ein synchroner Bildlauf für beide Editoren. Sie können dieses Verhalten beliebig aus- und einschalten. Gehen Sie dazu folgendermaßen vor:

1. Um den synchronen Bildlauf auszuschalten, klicken Sie in der Funktionsleiste auf die Schaltfläche "Bildlauf der Editoren synchronisieren".
2. Um den synchronen Bildlauf wieder einzuschalten, klicken Sie erneut in der Funktionsleiste auf die Schaltfläche "Bildlauf der Editoren synchronisieren".

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Detailvergleich für GRAPH-Bausteine starten (Seite 7927)

Darstellung des Vergleichsergebnisses für KOP/FUP (Seite 7930)

Darstellung des Vergleichsergebnisses für AWL (Seite 7934)

Darstellung des Vergleichsergebnisses für SCL (Seite 7937)

Darstellung des Vergleichsergebnisses für GRAPH (Seite 7940)

Bausteine während des Detailvergleichs ändern (Seite 7948)

Vergleichsergebnisse aktualisieren (Seite 7949)

20.2.2.5 Bausteine während des Detailvergleichs ändern

Während Sie einen Detailvergleich durchführen, können Sie Änderungen an den Bausteinen durchführen, die verglichen werden. Beachten Sie dabei Folgendes:

- Offline/Online-Vergleich: Sie können nur den Offline-Baustein ändern.
- Offline/Offline-Vergleich: Sie können nur den Offline-Baustein im linken Bereich ändern.

Nach einer Bausteinänderung ist es evtl. notwendig, dass Sie im Vergleichseditor das Vergleichsergebnis manuell aktualisieren, damit der Vergleichsstatus korrekt angezeigt wird. Anschließend können Sie Aktionen festlegen, um die Objekte zu synchronisieren.

Hinweis

SCL-Bausteine können Sie nicht manuell ändern. Sie können aber Änderungen aus einem Baustein in den anderen Baustein übernehmen. Beachten Sie dabei die folgenden Besonderheiten:

- Sie können keine Änderungen in einen Online-Baustein übernehmen.
- In einen Offline-Baustein können Sie die Änderungen nur übernehmen, wenn dieser nicht schreibgeschützt ist. Dies ist z. B. der Fall, wenn die Bausteine des Detailvergleichs aus unterschiedlichen CPUs stammen. Dann ist auch die Übernahme der Änderungen in den Baustein im rechten Bereich möglich.

KOP-, FUP- oder AWL-Bausteine ändern

Um KOP-, FUP- oder AWL-Bausteine zu ändern, gehen Sie folgendermaßen vor:

1. Ändern Sie den Baustein im linken Bereich nach Ihren Bedürfnissen.
2. Klicken Sie in der Funktionsleiste ggf. auf "Vergleichsergebnisse aktualisieren".

GRAPH-Bausteine ändern

Um GRAPH-Baustein zu ändern, gehen Sie folgendermaßen vor:

1. Wechseln Sie in die Kettenansicht, indem Sie zwischen den beiden Bausteinen auf "Kettenansicht" klicken.
2. Ändern Sie den Baustein im linken Bereich nach Ihren Bedürfnissen.
3. Klicken Sie in der Funktionsleiste ggf. auf "Vergleichsergebnisse aktualisieren".

SCL-Bausteine ändern

Um eine Änderung von einem Baustein in den anderen zu übernehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Seitenleiste des Bausteins, von dem Sie eine Änderung in den anderen Baustein übernehmen möchten, auf den Pfeil in der entsprechenden Zeile.
   Die Zeile wird in den anderen Baustein eingefügt und die Pfeilschaltflächen werden entfernt.

   Hinweis

   Die Farben der Pfeile haben folgende Bedeutungen:
   - Grau: Die Änderungen können nicht in den anderen Baustein übernommen werden, da der andere Baustein entweder ein Online-Baustein oder schreibgeschützt ist.
   - Blau: Die Änderungen werden aus einem Offline-Baustein in den anderen Baustein übernommen.
   - Orange: Die Änderungen werden aus einem Online-Baustein in den anderen Baustein übernommen.

2. Klicken Sie in der Funktionsleiste ggf. auf "Vergleichsergebnisse aktualisieren".

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)

Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)

Detailvergleich für GRAPH-Bausteine starten (Seite 7927)

Darstellung des Vergleichsergebnisses für KOP/FUP (Seite 7930)

Darstellung des Vergleichsergebnisses für AWL (Seite 7934)

Darstellung des Vergleichsergebnisses für SCL (Seite 7937)

Darstellung des Vergleichsergebnisses für GRAPH (Seite 7940)

Im Detailvergleich navigieren (Seite 7946)

Vergleichsergebnisse aktualisieren (Seite 7949)

20.2.2.6 Vergleichsergebnisse aktualisieren

Sobald Sie ein Objekt ändern, sind die Vergleichsergebnisse nicht mehr gültig und müssen aktualisiert werden.

Voraussetzung

Ein Detailvergleich wurde durchgeführt.

Vorgehen

Um die Vergleichsergebnisse zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf "Vergleichsergebnisse aktualisieren".

Siehe auch

Grundlagen zum Vergleichen von PLC-Programmen (Seite 7917)
Detailvergleich für KOP/FUP/AWL/SCL-Bausteine starten (Seite 7925)
Detailvergleich für GRAPH-Bausteine starten (Seite 7927)
Darstellung des Vergleichsergebnisses für KOP/FUP (Seite 7930)
Darstellung des Vergleichsergebnisses für AWL (Seite 7934)
Darstellung des Vergleichsergebnisses für SCL (Seite 7937)
Darstellung des Vergleichsergebnisses für GRAPH (Seite 7940)
Im Detailvergleich navigieren (Seite 7946)
Bausteine während des Detailvergleichs ändern (Seite 7948)

20.3 PLC-Variablen vergleichen

Sie haben folgende Möglichkeiten, PLC-Variablen zu vergleichen:

- Automatischer Offline/Offline-Vergleich im Vergleichseditor
  Die PLC-Variablentabellen der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich im Vergleichseditor
  Die ausgewählten PLC-Variablentabellen der Geräte werden offline verglichen.
- Offline/Online-Vergleich (nur S7-1500 ab Version 2.5)
  Die PLC-Variablen im Projekt werden mit den PLC-Variablen des gewählten Geräts verglichen.
- Detailvergleich
  Über den Detailvergleich können Sie die Unterschiede innerhalb der PLC-Variablentabellen ermitteln. Sie können den Detailvergleich entweder im Vergleichseditor, in der Projektnavigation oder in der PLC-Variablentabelle "PLC-Variablen" (nur Offline/Online) starten. Der Offline/Offline-Detailvergleich steht Ihnen für alle Geräte zur Verfügung, der Offline/Online-Detailvergleich für S7-1500-CPUs ab Version 2.5.

Automatischen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen automatischen Offline/Offline-Vergleich von PLC-Variablentabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "PLC-Variablen".
   Den Status der PLC-Variablentabellen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status können Sie bestimmte Aktionen festlegen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Manuellen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen manuellen Offline/Offline-Vergleich von PLC-Variablentabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.

3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die PLC-Variablentabellen, die Sie miteinander vergleichen möchten. Der Eigenschaftsvergleich wird angezeigt. Den Status können Sie an den Symbolen ablesen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Offline/Online-Vergleich durchführen (nur S7-1500 ab Version 2.5)

Um einen Offline/Online-Vergleich von PLC-Variablentabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Online-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".
3. Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".
   Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.
4. Öffnen Sie den Ordner "PLC-Variablen".
   Den Status der PLC-Variablentabellen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Beachten Sie dabei jedoch, dass Sie in einer Synchronisationsaktion nur Aktionen in eine Richtung ausführen können.

Detailvergleich über den Vergleichseditor starten

Um einen Detailvergleich für eine PLC-Variablentabelle über den Vergleichseditor zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Offline/Offline- oder Offline/Online-Vergleich durch.
   Der Vergleichseditor wird geöffnet.
2. Selektieren Sie bei einem automatischen Offline/Offline-Vergleich oder Offline/Online-Vergleich im Vergleichseditor die PLC-Variablentabelle, für die Sie einen Detailvergleich durchführen möchten. Beachten Sie bei einem manuellen Offline/Offline-Vergleich, dass zwei PLC-Variablentabellen für einen Vergleich ausgewählt sein müssen.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".
   Es wird ein eigener Vergleichseditor geöffnet. Alle vorhandenen PLC-Variablen und Anwenderkonstanten der ausgewählten PLC-Variablentabellen werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Systemkonstanten werden dagegen nicht angezeigt. Den Status der PLC-Variablen können Sie an den Symbolen ablesen. Abhängig vom Status der PLC-Variablen können Sie bei einem Offline/Offline-Vergleich bestimmte Aktionen festlegen.

Detailvergleich in der Projektnavigation starten

Um für eine PLC-Variablentabelle einen Offline/Offline-Detailvergleich in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die PLC-Variablentabelle, die Sie vergleichen möchten. Dies kann auch eine PLC-Variablentabelle aus einem Referenzprojekt sein.
2. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Als linkes Objekt auswählen".
3. Klicken Sie mit der rechten Maustaste auf die PLC-Variablentabelle, die Sie mit der PLC-Variablentabelle vergleichen möchten, die Sie zuvor als linkes Objekt ausgewählt haben.
4. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit <ausgewähltes Objekt> vergleichen". "<ausgewähltes Objekt>" steht für das linke Vergleichsobjekt.
   Es wird ein eigener Vergleichseditor geöffnet. Alle vorhandenen PLC-Variablen und Anwenderkonstanten der ausgewählten PLC-Variablentabellen werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Systemkonstanten werden dagegen nicht angezeigt. Den Status der PLC-Variablen können Sie an den Symbolen ablesen. Abhängig vom Status der PLC-Variablen können Sie bestimmte Aktionen festlegen.

Um für eine PLC-Variablentabelle einen Offline/Online-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, auf dem die PLC-Variablentabelle vorhanden ist.
2. Klicken Sie mit der rechten Maustaste auf die PLC-Variablentabelle, die Sie mit ihrem Online-Objekt vergleichen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit dem Online-Objekt vergleichen".
   Es wird ein eigener Vergleichseditor geöffnet. Alle vorhandenen PLC-Variablen und Anwenderkonstanten der ausgewählten PLC-Variablentabellen werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Systemkonstanten werden dagegen nicht angezeigt. Den Status der PLC-Variablen können Sie an den Symbolen ablesen.

Detailvergleich in PLC-Variablentabelle "PLC-Variablen" starten

Um für eine PLC-Variablentabelle einen Offline/Online-Detailvergleich in der PLC-Variablentabelle "Alle Variablen anzeigen" zu starten, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, auf dem die PLC-Variablentabelle vorhanden ist.
2. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
3. Doppelklicken Sie auf den Eintrag "Alle Variablen anzeigen".
   Die PLC-Variablentabelle "PLC-Variablen" wird geöffnet.
4. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".
   Es wird ein eigener Vergleichseditor geöffnet. Alle vorhandenen PLC-Variablen und Anwenderkonstanten der ausgewählten PLC-Variablentabellen werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Systemkonstanten werden dagegen nicht angezeigt. Den Status der PLC-Variablen können Sie an den Symbolen ablesen.

Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 7917)
PLC-Datentypen vergleichen (Seite 7955)
Bausteine vergleichen (Seite 7923)

20.4 PLC-Datentypen vergleichen

Sie haben folgende Möglichkeiten, PLC-Datentypen zu vergleichen:

- Offline/Online-Vergleich (nur S7-1200/1500)
  Die PLC-Datentypen im Projekt werden mit den PLC-Datentypen des gewählten Geräts verglichen.
- Automatischer Offline/Offline-Vergleich im Vergleichseditor
  Die PLC-Datentypen der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich im Vergleichseditor
  Die ausgewählten PLC-Datentypen der Geräte werden offline verglichen.
- Detailvergleich
  Über den Detailvergleich können Sie die Unterschiede zwischen PLC-Datentypen ermitteln. Sie können den Detailvergleich entweder über den Vergleichseditor oder in der Projektnavigation starten.

Offline/Online-Vergleich von PLC-Datentypen im Vergleichseditor durchführen

Um einen Offline/Online-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Online-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".
   Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".
   Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.
3. Öffnen Sie den Ordner "PLC-Datentypen".
   Den Status der PLC-Datentypen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des PLC-Datentyps und des entsprechenden PLC-Datentyps des zugeordneten Geräts im Eigenschaftsvergleich angezeigt.

Automatischen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen automatischen Offline/Offline-Vergleich von PLC-Datentypen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.

3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "PLC-Datentypen".
   Den Status der PLC-Datentypen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status können Sie bestimmte Aktionen festlegen. Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Manuellen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen manuellen Offline/Offline-Vergleich von PLC-Datentypen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".
   Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die PLC-Datentypen, die Sie miteinander vergleichen möchten.
   Der Eigenschaftsvergleich wird angezeigt. Den Status können Sie an den Symbolen ablesen. Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

Detailvergleich über den Vergleichseditor starten

Um einen Detailvergleich für einen PLC-Datentyp über den Vergleichseditor zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Offline/Offline-Vergleich durch. Für CPUs der Baureihe S7-1200/1500 können Sie auch einen Offline/Online-Vergleich verwenden.
2. Selektieren Sie bei einem automatischen Offline/Offline-Vergleich im Vergleichseditor den PLC-Datentyp, für den Sie einen Detailvergleich durchführen möchten. Beachten Sie bei einem manuellen Offline/Offline-Vergleich, dass zwei PLC-Datentypen für einen Vergleich ausgewählt sein müssen.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".
   Das Fenster "PLC-Datentypvergleich" wird geöffnet und alle Variablen der verglichenen PLC-Datentypen werden angezeigt. Im Inspektorfenster können Sie sich unter "Info > Vergleichsergebnis" die Unterschiede ansehen.

Detailvergleich in der Projektnavigation starten

Um für einen PLC-Datentyp einen Offline/Offline-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den PLC-Datentyp, den Sie vergleichen möchten. Dies kann auch ein PLC-Datentyp aus einem Referenzprojekt sein.
2. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Als linkes Objekt auswählen".
3. Klicken Sie mit der rechten Maustaste auf den PLC-Datentyp, den Sie mit dem PLC-Datentyp vergleichen möchten, den Sie zuvor als linkes Objekt ausgewählt haben.
4. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit <ausgewähltes Objekt> vergleichen". "<ausgewähltes Objekt>" steht für das linke Vergleichsobjekt.
   Das Fenster "PLC-Datentypvergleich" wird geöffnet und alle Variablen der verglichenen PLC-Datentypen werden angezeigt. Im Inspektorfenster können Sie sich unter "Info > Vergleichsergebnis" die Unterschiede ansehen.

Um für eine CPU der Baureihe S7-1200/1500 für einen PLC-Datentyp einen Offline/Online-Detailvergleich direkt in der Projektnavigation zu starten, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, auf dem der Baustein vorhanden ist.
2. Klicken Sie mit der rechten Maustaste auf den PLC-Datentyp, den Sie mit seinem Online-Objekt vergleichen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Schnellvergleich > Mit dem Online-Objekt vergleichen".
   Das Fenster "PLC-Datentypvergleich" wird geöffnet und alle Variablen der verglichenen PLC-Datentypen werden angezeigt. Im Inspektorfenster können Sie sich unter "Info > Vergleichsergebnis" die Unterschiede ansehen.

Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 7917)

Bausteine vergleichen (Seite 7923)

PLC-Variablen vergleichen (Seite 7951)

Programminformationen anzeigen 21

21.1 Übersicht der möglichen Programminformationen

Programminformationen

Die Programminformationen eines Anwenderprogramms enthalten die in der nachfolgenden Tabelle angegebenen Ansichten.

Ansicht	Anwendung
Belegungsplan (Seite 7960)	Gibt einen Überblick, welche Bits der Operanden der Speicherbereiche E, A und M innerhalb des Anwenderprogramms bereits belegt sind. Zeigt zudem an, ob eine Adresse durch einen Zugriff aus einem S7-Programm heraus belegt ist oder ob die Adresse einer SIMATIC-S7-Baugruppe zugeordnet ist
Aufrufstruktur (Seite 7969)	Zeigt die Aufrufhierarchie der Bausteine innerhalb eines Anwenderprogramms und gibt einen Überblick über die verwendeten Bausteine und deren Abhängigkeiten.
Abhängigkeitsstruktur (Seite 7977)	Zeigt die Liste der im Anwenderprogramm verwendeten Bausteine. In der ersten Ebene steht der jeweilige Baustein und darunter eingerückt die Bausteine, die diesen Baustein aufrufen oder verwenden. Instanz- Datenbausteine werden im Gegensatz zur Aufrufstruktur separat aufgelistet.
Speicherauslastung (Seite 7983)	Zeigt die Hardware-Speicherauslastung der CPU für Objekte (OB, FC, FB, DB, anwenderdefinierte Datentypen und PLC-Variablen), für Speicherbereiche der CPU und für die vorhandenen Ein-/Ausgabebaugruppen an.

Gleichzeitige Anzeige mehrerer Ansichten

Sie können mehrere Ansichten zu einem oder mehreren Anwenderprogrammen erzeugen und anzeigen lassen, um sich das Testen und Ändern Ihres Anwenderprogramms zu erleichtern.

Das Anzeigen von mehreren Ansichten ermöglicht Ihnen z. B.:

- Alle Programminformationen zu einem Anwenderprogramm nebeneinander anzuzeigen
- Eine Gegenüberstellung unterschiedlicher Anwenderprogramme vorzunehmen

21.2 Belegungsplan anzeigen

21.2.1 Einführung zum Belegungsplan

Programminformationen im Belegungsplan

Der Belegungsplan zeigt an, ob eine Adresse durch einen Zugriff aus einem S7-Programm heraus belegt ist oder ob die Adresse einer SIMATIC-S7-Baugruppe zugeordnet ist. Er ist somit eine wichtige Grundlage für die Fehlersuche oder für Änderungen im Anwenderprogramm.

Sie erhalten im Belegungsplan eine CPU-spezifische Übersicht, welches Bit in welchem Byte der nachfolgend aufgelisteten Speicherbereiche verwendet wird:

- Eingang (E)
- Ausgang (A)
- Merker (M)
- Zeiten (T)
- Zähler (Z)
- Peripherie (P)

Anzeige des Belegungsplans

Der Belegungsplan wird in mehreren Arbeitsfenstern angezeigt, jeweils getrennt für Ein- und Ausgänge, sowie für Merker.

Filtern

Die Anzeige innerhalb des Belegungsplans können Sie filtern. Sie können dazu vordefinierte Filter verwenden oder eigene Filter erstellen.

Anzeige der Querverweisinformationen

Im Belegungsplan haben Sie die Möglichkeit die Querverweisinformationen zu ausgewählten Adressen anzuzeigen.

Die Querverweise zu einer selektierten Adresse zeigen Sie im Inspektorfenster mit dem Kontextmenübefehl "Querverweis-Informationen" an. Über den Befehl "Werkzeuge > Querverweise" können Sie zusätzlich auch die Querverweisliste zum selektierten Objekt öffnen.

Anzeige der PLC-Variablentabelle

Aus dem Belegungsplan heraus können Sie die PLC-Variablentabelle öffnen und dort die Eigenschaften der verwendeten Variablen bearbeiten.

Dazu selektieren Sie eine Adresse des Belegungsplans und wählen Sie im Kontextmenü den Befehl "Editor öffnen"

Anzeige der Remanenz aktivieren

Im Belegungsplan haben Sie die Möglichkeit die Anzeige der Remanenz für Merker über das Symbol "Remanenz ein/ausblenden" der Funktionsleiste zu aktivieren und deaktivieren.

Siehe auch

Symbole im Belegungsplan (Seite 7962)

Aufbau des Belegungsplans (Seite 7961)

21.2.2 Aufbau des Belegungsplans

Aufbau des Belegungsplans

Der Belegungsplan wird CPU-abhängig in mehreren Arbeitsfenstern mit folgenden Operanden angezeigt.

Für S7-300/400er und S7-1500er CPUs:

- Eingänge
- Ausgänge
- Merker
- Zeiten
- Zähler

Für S7-1200er CPUs:

- Eingänge
- Ausgänge
- Merker

Anzeige von Eingängen, Ausgängen , Merkern, Zeiten und Zählern

Es werden alle verwendeten Operanden sowie deren Belegung im S7-Programm angezeigt.

Jede Zeile des Belegungsplans beinhaltet für alle angezeigten Operanden jeweils ein Byte des Speicherbereichs, in dem die zugehörigen acht Bits von 7 bis 0 je nach Zugriff gekennzeichnet werden. Im Anschluss wird durch einen "Balken" angezeigt, ob der Zugriff durch einen Byte- (B), Wort- (W) oder Doppelwortzugriff (D) erfolgt.

Eine Erklärung zu den Symbolen im Belegungsplan finden Sie hier: (Seite 7962)

Siehe auch

Einführung zum Belegungsplan (Seite 7960)

21.2.3 Symbole im Belegungsplan

Bedeutung der Symbole im Belegungsplan

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole im Belegungsplan:

Symbol	Bedeutung
	Zeigt die Adressbelegung im selektierten Zustand an.
	Zeigt die Adressbelegung im nicht selektierten Zustand an.
	Zeigt an, dass eine Pointerstartadresse und eine Variablenadresse auf denselben Adressbereich zugreifen und dass diese selektiert sind.
	Zeigt an, dass eine Pointerstartadresse und eine Variablenadresse auf denselben Adressbereich zugreifen und dass diese nicht selektiert sind.
	Zeigt die Pointerbelegung im selektierten Zustand an.
	Zeigt die Pointerbelegung im nicht selektierten Zustand an.
	Zeigt an, dass das Byte durch einen Bytezugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablentabelle angezeigt werden.
	Zeigt an, dass das Byte durch einen Bytezugriff belegt ist und die entsprechende Variable nicht selektiert ist.
	Zeigt an, dass das Byte durch einen Wortzugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablentabelle angezeigt werden.
	Zeigt an, dass das Byte durch einen Wortzugriff belegt ist und die entsprechende Variable nicht selektiert ist.
	Zeigt an, dass das Byte durch einen Doppelwortzugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablentabelle angezeigt werden.
	Zeigt an, dass das Byte durch einen Doppelwortzugriff belegt ist und die entsprechende Variable nicht selektiert ist.
Hintergrundfarbe: grau	Zeigt an, dass ein Byte-, Wort- oder Doppelwortzugriff vorhanden ist und dass die Adresse auch von der Hardware belegt ist. Die graue Hintergrundfarbe ist ein Hinweis auf einen überlappenden Speicherzugriff.
Hintergrundfarbe: gelb	Zeigt an, dass die Adresse von der Hardware nicht belegt ist.

Symbol	Bedeutung
	Zeigt an, dass der Speicherbereich als Systemmerker definiert ist.
	Zeigt an, dass der Speicherbereich als Taktmerker definiert ist.

Siehe auch

Aufbau des Belegungsplans (Seite 7961)

Einführung zum Belegungsplan (Seite 7960)

21.2.4 Belegungsplan anzeigen

Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

Vorgehen

Um den Belegungsplan anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Belegungsplan".

Ergebnis

Der Belegungsplan zum selektierten Programm wird angezeigt.

Ansichtsoptionen im Belegungsplan

Beachten Sie die jeweils eingestellten Ansichtsoptionen, um die gewünschten Informationen im Belegungsplan angezeigt zu bekommen.

Siehe auch

Ansichtsoptionen für Belegungsplan einstellen (Seite 7964)

Aufbau des Belegungsplans (Seite 7961)

21.2.5 Ansichtsoptionen für Belegungsplan einstellen

Einführung

Für den Belegungsplan stehen folgende Ansichtsoptionen zur Verfügung:

- Verwendete Adressen:
  Wenn dieses Optionskästchen aktiviert ist, werden die im Programm verwendeten Adressen, Peripherien und Pointer angezeigt.
- Freie Hardware-Adressen:
  Wenn dieses Optionskästchen aktiviert ist, werden nur die freien Hardware-Adressen angezeigt.

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

Vorgehen

Um die Ansichtsoptionen für den Belegungsplan einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols ("Ansichtsoptionen"). Die Ansichtsoptionen zum Belegungsplan werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden im Belegungsplan angezeigt.

21.2.6 Filtermöglichkeiten im Belegungsplan

Filtereinstellungen

Sie haben die Möglichkeit, selbst definierte Filtereinstellungen für den Belegungsplan festzulegen. Folgende Möglichkeiten für das Definieren von Filtern stehen Ihnen zur Verfügung:

- Anzeige aller verwendeten Adressen aus den nachfolgend genannten Adressbereichen.
- Anzeige einzelner, definierter Adressen, aus dem selektierten Adressbereich, z. B. "0" und "200".
- Anzeige kompletter Bereiche aus dem selektierten Adressbereich, z. B. "0 - 256".

Die folgende Tabelle zeigt alle Auswahlmöglichkeiten im Überblick:

Selektion im	Auswahl	Symbol	Bedeutung
Adressbereich	Alle CPU-abhängig angezeigten Adressen (E, A, M, T ,Z) wie in der Voreinstellung oder nur einzelne Adressbereiche können aktiviert werden.	Optionskästchen ist aktiviert	Nur die aktivierten Adressbereiche (E, A, M, T, Z) werden im Belegungsplan angezeigt.
Filterbereich	Belegung für alle Adressen anzeigen	*	Belegung aller verwendeten Adressen für die aktivierten Adressbereiche (E, A, M) wird angezeigt.
	Belegung für ausgewählte Adressen anzeigen, z. B. für die Eingänge "EB 0" und "EB 256"	0;256 Einzelne Adressen und Bereiche sind durch Semikolon zu trennen.	Belegung der genannten Adressen für die aktivierten Adressbereiche (E) wird angezeigt.
	Belegung für ausgewählte Bereiche anzeigen, z. B. für die Eingänge von "EB 0 bis EB 100" und von "EB 200 bis EB 256".	0-100;200-256 Zusammenhängende Bereiche sind durch Bindestrich zu verbinden.	Belegung der genannten Bereiche für die aktivierten Adressbereiche (E) wird angezeigt.

21.2.7 Filter für Belegungsplan definieren

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

Filter definieren

Um einen Filter für den Belegungsplan zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol ("Filter").
   Der Dialog "Assignmentlist Filter" wird geöffnet.
2. Klicken Sie auf das Symbol ("Neuen Filter erstellen").
   Ein neuer Filter wird unter dem Namen "Filter_1" erstellt. Als Voreinstellung für den Filter werden CPU-abhängig die Optionskästchen aller Adressen (z. B. Eingänge, Ausgänge, Merker, Zeiten und Zähler) aktiviert.
3. Wenn Sie den Namen des Filters ändern wollen, klicken Sie in der Funktionsleiste auf die Klappliste und geben einen neuen Filternamen ein.
4. Deaktivieren Sie die Optionskästchen der Adressen, die das Filtern nicht beeinflussen sollen.

5. Tragen Sie im Filterbereich der aktivierten Adressen eine der folgenden Optionen ein:
   - Anzeige aller verwendeten Adressen = "*"
   - Anzeige einzelner, definierter Adressen, z. B. EB 0 und EB 25 = "0,25". Einzelne Adressen und Adressbereiche werden durch Kommata oder Semikolon getrennt.
   - Anzeige komplette Adressbereiche, z. B. von EB 0 bis EB 256 = "0-256". Komplette Adressbereiche sind durch Bindestrich zu verbinden.
6. Bestätigen Sie die Eingaben mit "OK".
   Der neu definierte Filter wird unter dem vorgegebenen Namen in der Funktionsleiste des Belegungsplans angezeigt.

Filter löschen

Um einen Filter zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol ("Filter").
   Der Filter-Dialog für den Belegungsplan wird geöffnet.
2. Wählen Sie aus Klappliste in der Funktionsleiste den Filter aus, den Sie löschen wollen.
3. Klicken Sie auf das Symbol ("Ausgewählten Filter löschen").
   Der ausgewählte Filter wird gelöscht.

Siehe auch

Filtermöglichkeiten im Belegungsplan (Seite 7964)

Belegungsplan anzeigen (Seite 7963)

Einführung zum Belegungsplan (Seite 7960)

21.2.8 Belegungsplan filtern

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

Vorgehen

1. Klicken Sie in der Funktionsleiste auf den Pfeil der Klappliste.
   Die vorhandenen Filter werden angezeigt.
2. Wählen Sie den gewünschten Filter aus.

Ergebnis

Der Belegungsplan wird nach den Einstellungen des ausgewählten Filters gefiltert.

Hinweis

Die Filtereinstellungen werden beim Schließen des Projekts gespeichert.

21.2.9 Remanente Speicherbereiche für Merker definieren

Einführung

Im Belegungsplan können Sie die Breite des remanenten Speicherbereichs für Merker definieren. Der Inhalt der Variablen, die im remanenten Speicherbereich adressiert sind, bleibt nach Ausschalten der Versorgungsspannung und nach Einschalten bei einem Übergang von STOP nach RUN erhalten.

Die Anzeige der remanenten Merker kann im Belegungsplan aktiviert und deaktiviert werden. Remanente Merker erkennen Sie bei aktivierter Anzeige am Icon in der Spalte "Adresse".

Voraussetzung

Der Belegungsplan ist geöffnet.

Vorgehen

Um die Breite des remanenten Speicherbereichs für Merker zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz".
   Der Dialog "Remanenter Speicher" wird geöffnet.
2. Legen Sie die Breite des remanenten Speicherbereichs fest, indem Sie von 0 aus gezählt das letzte Byte des Bereichs in das Eingabefeld eingeben. Achten Sie dabei auf die Adressen vorhandener Variablen, die als remanent definiert sind.
3. Laden Sie die geänderten Bausteine anschließend ins Zielsystem. Selektieren Sie dazu den Ordner "Programmbausteine" in der Projektnavigation und wählen Sie im Kontextmenü das Untermenü "Laden in Gerät".

Ergebnis

Die Breite des remanenten Speicherbereichs wird definiert. Falls die Anzeige der Remanenz im Belegungsplan aktiviert ist, wird in der Spalte "Adresse" für alle remanenten Variablen das entsprechende Icon angezeigt.

21.2.10 Anzeige der Remanenz für Merker aktivieren

Einführung

Im Belegungsplan können Sie die Anzeige der remanenten Merker aktivieren und deaktivieren. Wenn Sie die Anzeige der Remanenz aktivieren, werden die als remanent definierte Merker mit einem Icon in der Spalte "Adresse" gekennzeichnet.

Voraussetzung

Der Belegungsplan ist geöffnet.

Vorgehen

Um die Anzeige der remanenten Merker zu aktivieren und deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz ein/ausblenden".

Ergebnis

Wenn die Anzeige der Remanenz aktiviert ist, werden die als remanent definierten Variablen mit einem Icon in der Spalte "Adresse" des Merker-Bereichs gekennzeichnet. Wird die Anzeige der Remanenz deaktiviert, werden die Icons in der Spalte "Adresse" ausgeblendet.

21.3 Aufrufstruktur anzeigen

21.3.1 Einführung zur Aufrufstruktur

Aufrufstruktur

Die Aufrufstruktur beschreibt die Aufrufhierarchie der Bausteine innerhalb eines S7-Programms.

Sie gibt einen Überblick über:

- Verwendete Bausteine
- Sprünge zu den Verwendungsstellen der Bausteine
- Abhängigkeiten zwischen den Bausteinen
- Lokaldatenbedarf der Bausteine
- Status der Bausteine

Informationen in der Aufrufstruktur

Mit der Anzeige der Aufrufstruktur erhalten Sie eine Liste der im Anwenderprogramm verwendeten Bausteine. Die erste Ebene der Aufrufstruktur ist farblich hervorgehoben und zeigt die Bausteine an, die von keinem anderen Baustein im Programm aufgerufen werden. Organisationsbausteine werden immer in der ersten Ebene der Aufrufstruktur angezeigt. Funktionen, Funktionsbausteine und Datenbausteine werden nur dann in der ersten Ebene angezeigt, wenn sie von keinem Organisationsbaustein aufgerufen werden. Wenn ein Baustein andere Bausteine oder Funktionen aufruft, werden diese unter dem Baustein eingerückt aufgelistet. Anweisungen und Bausteine werden in der Aufrufstruktur nur dann dargestellt, wenn sie von einem Baustein aufgerufen werden.

Hinweis

Bitte beachten Sie, dass für Know-how-geschütze Bausteine keine Aufrufstruktur angezeigt wird.

Ansichtsoptionen

Für die Aufrufstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:
  Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Aufrufstruktur angezeigt.
- Mehrere Aufrufe zusammenfassen:
  Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Anzahl der Aufrufe" angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

Anzeige der Bausteinaufrufe

Die in einem Baustein enthaltenen Bausteinaufrufe können Sie anzeigen, indem Sie auf den Pfeil vor dem Bausteintitel klicken. Um die Aufrufinformationen aller Bausteine einzublenden, klicken Sie in der Funktionsleiste auf das Symbol "Liste erweitern".

Das Zuklappen der Gesamtübersicht erfolgt über das Symbol "Liste reduzieren".

Anzeige der Querverweisinformationen

Die Querverweisinformationen zu einem Baustein können Sie im Inspektorfenster einblenden, indem Sie auf den entsprechenden Baustein mit der rechten Maustaste klicken und im Kontextmenü den Befehl "Querverweis-Informationen" wählen.

Um die Ansicht "Querverweise" zu öffnen, klicken Sie im Kontextmenü auf den Befehl "Querverweise".

Anzeige der Bausteine im Programmeditor

Aus der Aufrufstruktur heraus können Sie den Programmeditor öffnen und dort die Bausteine bearbeiten.

Dazu selektieren Sie den gewünschten Baustein in der Aufrufstruktur und wählen Sie im Kontextmenü den Befehl "Editor öffnen".

Anzeige gelöschter Bausteine

Die Zeilen zu gelöschten Bausteinen werden mit einem Symbol gekennzeichnet.

Hinweis

Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.

Siehe auch

Symbole in der Aufrufstruktur (Seite 7971)

21.3.2 Symbole in der Aufrufstruktur

Bedeutung der Symbole in der Aufrufstruktur

Die nachfolgende Tabelle zeigt beispielhaft die Bedeutung der Symbole in der Aufrufstruktur:

Symbol	Bedeutung
	Zeigt einen Organisationsbaustein (OB) an.
	Zeigt einen Funktionsbaustein (FB) an.
	Zeigt eine Funktion (FC) an.
	Zeigt einen Datenbaustein (DB) an.
	Zeigt an, dass der Baustein als Multiinstanz oder als Parameterinstanz deklariert ist.
	Zeigt an, dass das Objekt eine Schnittstellen-Abhängigkeit zum links verbundenen Objekt besitzt.
	Zeigt an, dass der Baustein neu übersetzt werden muss.
	Zeigt an, dass der Datenbaustein neu übersetzt werden muss.
	Zeigt an, dass das Objekt nicht vorhanden ist.
	Zeigt an, dass die Schnittstelle einen Zeitstempelkonflikt verursacht.
	Zeigt an, dass eine Variable einen Zeitstempelkonflikt verursacht.
	Zeigt an, dass der Baustein von einem OB nicht direkt oder indirekt aufgerufen wird.
	Zeigt an, dass ein Objekt mit Know-how-Schutz versehen ist.
	Zeigt an, dass eine Variablendeklaration in der Schnittstelle eine rekursive Abhängigkeit besitzt: • Fall 1: FB1 ruft FB2 auf und dieser wieder den FB1. Die Instanzdatenbausteine dieser FBs haben in der Schnittstelle eine Rekursion. • Fall 2: Ein Multiinstanz-FB verwendet den Instanz-DB seines Vater-FBs als Globalen DB.
	Zeigt an, dass der Baustein regulär rekursiv aufgerufen wird.
	Zeigt an, dass ein Baustein rekursiv mit Bedingung aufgerufen wird.
	Zeigt an, dass ein Baustein rekursiv ohne Bedingung aufgerufen wird.

21.3.3 Aufbau der Aufrufstruktur

Aufbau der Aufrufstruktur

Die Ansicht der Aufrufstruktur besteht aus folgenden Spalten:

Spalte	Inhalt/Bedeutung
Aufrufstruktur	Zeigt die Übersicht der aufgerufenen Bausteine an Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen werden mehrere Bausteinaufrufe zusammengefasst und die Spalte "Anzahl der Aufrufe" wird eingeblendet.
Aufrufart (!)	Zeigt die Art des Aufrufs an, z. B. rekursiver Bausteinaufruf.
Adresse	Zeigt die absolute Adresse des Bausteins an. Bei einem Funktionsbaustein wird die absolute Adresse des zugehörigen Instanzdatenbausteins mit angezeigt.
Anzahl der Aufrufe	Zeigt die Anzahl der Mehrfachaufrufe von Bausteinen an.
Details	Zeigt das Netzwerk bzw. die Schnittstelle des aufrufenden Bausteins an. In dieser Spalte sind alle Informationen als Link angeboten. Über diesen Link erreichen Sie im Programmeditor die Stelle, die den Bausteinaufruf enthält. Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen" werden die Aufrufe zusammengefasst und die Links in einer Klappliste angeboten.
Lokaldaten (im Pfad)	Zeigt den Lokaldatenbedarf des gesamten Pfades an. Bausteine mit optimiertem Zugriff haben einen höheren Lokaldatenbedarf, da die Information für die symbolische Adressierung mit hinterlegt wird. Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.
Lokaldaten (für Bausteine)	Zeigt den Lokaldatenbedarf des Bausteins an. Bausteine mit optimiertem Zugriff haben einen höheren Lokaldatenbedarf, da die Information für die symbolische Adressierung mit hinterlegt wird. Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.

Siehe auch

Symbole in der Aufrufstruktur (Seite 7971)

Einführung zur Konsistenzprüfung in der Aufrufstruktur (Seite 7975)

21.3.4 Aufrufstruktur anzeigen

Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

Vorgehen

Um die Aufrufstruktur anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Aufrufstruktur".

Ergebnis

Die Aufrufstruktur zum selektierten Programm wird angezeigt.

Hinweis

Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.

Siehe auch

Ansichtsoptionen für die Aufrufstruktur einstellen (Seite 7974)

21.3.5 Ansichtsoptionen für die Aufrufstruktur einstellen

Einführung

Für die Aufrufstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:
  Wenn dieses Optionskästchen aktiviert ist, werden nur konfliktbehaftete Bausteine innerhalb der Aufrufstruktur angezeigt.
  Folgende Bausteine verursachen einen Konflikt:
  - Bausteine, die bezüglich eines ihrer Aufrufe einen älteren oder jüngeren Code-Zeitstempel haben.
  - Bausteine, die einen Baustein aufrufen, dessen Schnittstelle sich geändert hat.
  - Bausteine, die eine Variable verwenden, deren Adresse und/oder Datentyp sich geändert hat.
  - Bausteine, die weder direkt noch indirekt von einem OB aufgerufen werden.
  - Bausteine, die einen Baustein aufrufen, der nicht mehr existiert.
- Mehrere Aufrufe zusammenfassen:
  Wenn diese Ansichtsoption aktiviert ist, werden mehrere Bausteinaufrufe und Datenbausteinzugriffe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Anzahl der Aufrufe" angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Aufrufstruktur ist geöffnet.

Vorgehen

Um die Ansichtsoptionen für die Aufrufstruktur einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols ("Ansichtsoptionen"). Die Ansichtsoptionen zur Aufrufstruktur werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden in der Aufrufstruktur angezeigt.

21.3.6 Einführung zur Konsistenzprüfung in der Aufrufstruktur

Konsistenzprüfung

Änderungen des Zeitstempels eines Bausteins während oder nach der Programmerstellung können zu Zeitstempel-Konflikten führen, die wiederum Inkonsistenzen zwischen den aufrufenden und aufgerufenen Bausteinen verursachen.

Verwendung der Konsistenzprüfung

Die Funktion "Konsistenzprüfung" dient zur Visualisierung von Inkonsistenzen bei Zeitstempel-Konflikten. Wenn die Konsistenzprüfung durchgeführt wird, werden die inkonsistenten Bausteine in der Aufrufstruktur angezeigt und mit den entsprechenden Symbolen gekennzeichnet.

- Ein großer Teil der Zeitstempel- und Schnittstellenkonflikte können entfernt werden, indem die Bausteine neu übersetzt werden.
- Wenn sich Inkonsistenzen durch die Übersetzung nicht bereinigen lassen, können Sie über den Link in der Spalte "Details" zu der genauen Problemstelle im Programmeditor wechseln und die Inkonsistenzen manuell bereinigen.
- Die rot gekennzeichneten Bausteine müssen neu übersetzt werden.

Siehe auch

Symbole in der Aufrufstruktur (Seite 7971)

21.3.7 Bausteinkonsistenz in der Aufrufstruktur prüfen

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Aufrufstruktur ist geöffnet.

Vorgehen

Um die Bausteinkonsistenz zu prüfen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol ("Konsistenzprüfung").
   Die Bausteinkonsistenz wird geprüft. Die Bausteine, die Inkonsistenzen aufweisen, werden durch das entsprechende Symbol gekennzeichnet.
2. Falls ein Baustein Inkonsistenzen aufweist, klicken Sie in der Aufrufstruktur auf den Pfeil vor dem entsprechenden Bausteintitel.
   Die inkonsistenten Bausteine werden angezeigt. In der Spalte "Details" werden die genauen Problemstellen als Links aufgelistet.

3. Klicken Sie in der Spalte "Details" auf den entsprechenden Link, um an die zu bereinigende Stelle im jeweiligen Baustein zu springen.
4. Überprüfen und bereinigen Sie die Inkonsistenzen in den Bausteinen.
5. Übersetzen Sie die Bausteine neu indem Sie die gewünschten Bausteine selektieren und im Kontextmenü auf den Befehl "Übersetzen" klicken.
6. Laden Sie die korrigierten Bausteine ins Zielsystem, indem Sie im Kontextmenü auf den Befehl "Laden in Gerät" klicken.

Ergebnis

Die Bausteinkonsistenz wird geprüft. Die Inkonsistenzen der Bausteine werden bereinigt. Die korrigierten Bausteine sind ins Zielsystem geladen.

Siehe auch

Symbole in der Aufrufstruktur (Seite 7971)

21.4 Abhängigkeitsstruktur anzeigen

21.4.1 Einführung zur Abhängigkeitsstruktur

Einführung

Die Abhängigkeitsstruktur zeigt für jeden Baustein im Programm die Abhängigkeit zu anderen Bausteinen an.

Informationen in der Abhängigkeitsstruktur

Mit der Anzeige der Abhängigkeitsstruktur erhalten Sie eine Liste der im Anwenderprogramm verwendeten Bausteine. Ganz links steht der jeweilige Baustein und darunter eingerückt die Bausteine, die diesen Baustein aufrufen oder verwenden.

Die Abhängigkeitsstruktur zeigt zusätzlich den Status der einzelnen Bausteine durch Symbole an.

Objekte, die einen Zeitstempelkonflikt verursachen und zu einer Inkonsistenz im Programm führen können, werden mit verschiedenen Symbolen gekennzeichnet.

Die Abhängigkeitsstruktur stellt eine Erweiterung der Querverweisliste für Objekte dar.

Ansichtsoptionen

Für die Abhängigkeitsstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:
  Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Abhängigkeitsstruktur angezeigt.
- Mehrere Aufrufe zusammenfassen:
  Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Abhängigkeitsstruktur" numerisch angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

Anzeige der Abhängigkeiten

Wenn Sie auf den Pfeil vor dem Bausteintitel klicken, zeigen sie die Bausteine an, in denen dieser Baustein aufgerufen bzw. verwendet wird. Um die Abhängigkeiten aller Bausteine einzublenden,

klicken Sie in der Funktionsleiste auf das Symbol "Liste erweitern".

Das Zuklappen der Gesamtübersicht erfolgt über das Symbol "Liste reduzieren".

Anzeige der Querverweisinformationen

Die Querverweisinformationen zu einem Baustein können Sie im Inspektorfenster einblenden, indem Sie auf den entsprechenden Baustein mit der rechten Maustaste klicken und im Kontextmenü den Befehl "Querverweis-Informationen" wählen.

Um die Ansicht "Querverweise" zu öffnen, klicken Sie im Kontextmenü auf den Befehl "Querverweise".

Anzeige der Bausteine im Programmeditor

Aus der Abhängigkeitsstruktur heraus können Sie den Programmeditor öffnen und dort die Bausteine bearbeiten. Dazu selektieren Sie den gewünschten Baustein in der Abhängigkeitsstruktur und wählen Sie im Kontextmenü den Befehl "Öffnen".

21.4.2 Aufbau der Abhängigkeitsstruktur

Aufbau der Abhängigkeitsstruktur

Die Ansicht der Abhängigkeitsstruktur besteht aus folgenden Spalten:

Spalte	Inhalt/Bedeutung
Abhängigkeit	Zeigt in einer Übersicht für jeden Baustein die Abhängigkeiten zu anderen Bausteinen an.
Aufrufart (!)	Zeigt die Art des Aufrufs an, z. B. rekursiver Bausteinaufruf.
Adresse	Zeigt die absolute Adresse des Bausteins an.
Anzahl der Aufrufe	Zeigt die Anzahl der Mehrfachaufrufe von Bausteinen an.
Details	Zeigt das Netzwerk bzw. die Schnittstelle des aufgerufenen Bausteins an. In dieser Spalte sind alle Informationen als Link angeboten. Über diesen Link erreichen Sie im Programmeditor die Stelle, die den Bausteinaufruf enthält. Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen" werden die Aufrufe zusammengefasst und die Links in einer Klappliste angeboten.

Siehe auch

Symbole in der Abhängigkeitsstruktur (Seite 7979)

21.4.3 Symbole in der Abhängigkeitsstruktur

Bedeutung der Symbole in der Abhängigkeitsstruktur

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole in der Abhängigkeitsstruktur:

Symbol	Bedeutung
	Zeigt einen Organisationsbaustein (OB) an.
	Zeigt einen Funktionsbaustein (FB) an.
	Zeigt eine Funktion (FC) an.
	Zeigt einen Datenbaustein (DB) an.
	Zeigt an, dass der Baustein als Multiinstanz oder als Parameterinstanz deklariert ist.
	Zeigt an, dass das Objekt eine Schnittstellen-Abhängigkeit zum links verbundenen Objekt besitzt.
	Zeigt an, dass der Baustein neu übersetzt werden muss.
	Zeigt an, dass der Datenbaustein neu übersetzt werden muss.
	Zeigt an, dass für dieses Objekt eine Inkonsistenz vorliegt.
	Zeigt an, dass ein Objekt mit Know-how-Schutz versehen ist.
	Zeigt an, dass eine Variablendeklaration in der Schnittstelle eine rekursive Abhängigkeit besitzt: • Fall 1: FB1 ruft FB2 auf und dieser wieder den FB1. Die Instanzdatenbausteine dieser FBs haben in der Schnittstelle eine Rekursion. • Fall 2: Ein Multiinstanz-FB verwendet den Instanz-DB seines Vater-FBs als Globalen DB.
	Zeigt an, dass der Baustein regulär rekursiv aufgerufen wird.
	Zeigt an, dass ein Baustein rekursiv mit Bedingung aufgerufen wird.
	Zeigt an, dass ein Baustein rekursiv ohne Bedingung aufgerufen wird.

21.4.4 Abhängigkeitsstruktur anzeigen

Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

Vorgehen

Um die Abhängigkeitsstruktur anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Abhängigkeitsstruktur".

Ergebnis

Die Abhängigkeitsstruktur zum selektierten Programm wird angezeigt.

Siehe auch

Ansichtsoptionen für die Abhängigkeitsstruktur einstellen (Seite 7980)

21.4.5 Ansichtsoptionen für die Abhängigkeitsstruktur einstellen

Einführung

Für die Abhängigkeitsstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:
  Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Abhängigkeitsstruktur angezeigt.
  Folgende Bausteine verursachen einen Konflikt:
  - Bausteine, die bezüglich eines ihrer Aufrufe einen älteren oder jüngeren Code-Zeitstempel haben.
  - Bausteine, die von einem Baustein aufgerufen werden, dessen Schnittstelle sich geändert hat.
  - Bausteine, die eine Variable verwenden, deren Adresse und/oder Datentyp sich geändert hat.
  - Bausteine, die weder direkt noch indirekt von einem OB aufgerufen werden.
- Mehrere Aufrufe zusammenfassen:
  Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinaufrufe wird in der entsprechenden Spalte angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Abhängigkeitsstruktur ist geöffnet.

Vorgehen

Um die Ansichtsoptionen für die Abhängigkeitsstruktur einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols ("Ansichtsoptionen"). Die Ansichtsoptionen zur Abhängigkeitsstruktur werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden in der Abhängigkeitsstruktur angezeigt.

21.4.6 Einführung zur Konsistenzprüfung in der Abhängigkeitsstruktur

Konsistenzprüfung

Änderungen des Zeitstempels eines Bausteins während oder nach der Programmerstellung können zu Zeitstempel-Konflikten führen, die wiederum Inkonsistenzen zwischen den aufrufenden und aufgerufenen Bausteinen verursachen.

Verwendung der Konsistenzprüfung

Die Funktion "Konsistenzprüfung" dient zur Visualisierung von Inkonsistenzen. Wenn die Konsistenzprüfung durchgeführt wird, werden die inkonsistenten Bausteine in der Abhängigkeitsstruktur angezeigt und mit den entsprechenden Symbolen gekennzeichnet.

- Ein großer Teil der Zeitstempel- und Schnittstellenkonflikte können entfernt werden, indem die Bausteine neu übersetzt werden.
- Wenn sich Inkonsistenzen durch die Übersetzung nicht bereinigen lassen, können Sie über den Link in der Spalte "Details" zu der genauen Problemstelle im Programmeditor wechseln und die Inkonsistenzen manuell bereinigen.
- Die rot gekennzeichneten Bausteine müssen neu übersetzt werden.

Siehe auch

Aufbau der Abhängigkeitsstruktur (Seite 7978)

Symbole in der Abhängigkeitsstruktur (Seite 7979)

21.4.7 Bausteinkonsistenz in der Abhängigkeitsstruktur prüfen

Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Abhängigkeitsstruktur ist geöffnet.

Vorgehen

Um die Bausteinkonsistenz zu prüfen, gehen Sie folgendermaßen vor:

1. Klicken in der Funktionsleiste auf das Symbol ("Konsistenzprüfung").
   Die Bausteinkonsistenz wird geprüft. Die Bausteine, die Inkonsistenzen aufweisen, werden durch das entsprechende Symbol gekennzeichnet.
2. Falls ein Baustein Inkonsistenzen aufweist, klicken Sie in der Abhängigkeitsstruktur auf den Pfeil vor dem entsprechenden Bausteintitel.
   Die inkonsistenten Bausteine werden angezeigt. In der Spalte "Details" werden die genauen Problemstellen als Links aufgelistet.
3. Überprüfen und bereinigen Sie die Inkonsistenzen in den Bausteinen.
4. Übersetzen Sie die Bausteine neu indem Sie die gewünschten Bausteine selektieren und im Kontextmenü auf den Befehl "Übersetzen" klicken.
5. Laden Sie die korrigierten Bausteine ins Zielsystem, indem Sie im Kontextmenü auf den Befehl "Laden in Gerät" klicken.

Ergebnis

Die Bausteinkonsistenz wird geprüft. Die Inkonsistenzen der Bausteine werden bereinigt. Die korrigierten Bausteine sind ins Zielsystem geladen.

Siehe auch

Symbole in der Abhängigkeitsstruktur (Seite 7979)

21.5 Speicherauslastung der CPU anzeigen

21.5.1 Einführung zur Speicherauslastung

Einführung

Das Register "Speicherauslastung" zeigt die jeweilige Hardware-Speicherauslastung der projektierten CPU für:

- die verwendeten Programmierobjekte,
- die Belegung der unterschiedlichen Speicherbereiche innerhalb der CPU und
- die belegten Ein- und Ausgänge auf den vorhandenen Ein-/Ausgabebaugruppen an.

Informationen im Register "Speicherauslastung"

Mit der Anzeige der Speicherauslastung erhalten Sie einen Überblick über die Hardware-Speicherauslastung. Die Anzeige in diesem Register ist abhängig von der von Ihnen verwendeten CPU. Folgende Informationen werden angezeigt:

- die in der CPU vorhandenen Programmier-Objekte (z. B. OB, FC, FB, DB, Datentypen und PLC-Variablen),
- die in der CPU vorhandenen Speicherbereiche (Ladespeicher, Arbeitsspeicher, je nach CPU unterteilt in Code-Arbeitsspeicher und Daten-Arbeitsspeicher und Remanenzspeicher), deren maximale Größe und deren Belegung durch die o. g. verwendeten Programmier-Objekte,
- die für die CPU konfigurierbaren Ein- und Ausgänge auf den zur Verfügung stehenden Baugruppen (Ein-/Ausgabebaugruppen, Digitale Eingabebaugruppen, Digitale Ausgabebaugruppen, Analoge Eingabebaugruppen und Analoge Ausgabebaugruppen) und die davon bereits belegten Ein- und Ausgänge.

Anzeige des maximal verfügbaren Ladespeichers

Die Größe des maximal verfügbaren Ladespeichers können Sie in der Spalte "Ladespeicher" in der Zeile "Gesamt" über eine Auswahlliste auswählen.

Hinweis

Anzeige des belegten Ladespeichers in der CPU

Beachten Sie bitte, dass die Summe des belegten Ladespeichers nicht genau bestimmt werden kann, falls nicht alle Bausteine übersetzt worden sind.

In diesem Fall zeigt ein vorangestelltes „>“ vor der Summe an, dass der Wert für den belegten Speicherplatz größer sein kann als angezeigt, da nicht übersetzte Bausteine für die Summenbildung nicht mit berücksichtigt werden.

Anzeige des maximal verfügbaren Arbeitsspeichers

Die Größe des maximal verfügbaren Arbeitsspeichers wird in der Spalte "Arbeitsspeicher" bzw. in den Spalten "Code-Arbeitsspeicher" und "Daten-Arbeitsspeicher" in der Zeile "Gesamt" angezeigt.

Anzeige des maximal verfügbaren Remanenzspeichers

Die Größe des maximal verfügbaren Remanenzspeichers wird in der Spalte "Remanenzspeicher" in der Zeile "Gesamt" angezeigt.

Hinweis

Remanenzdaten

Als remanent definierte Merker und Datenbausteine werden in die Berechnung der Remanenzdaten mit einbezogen.

Aktualisieren der Anzeige im Register "Speicherauslastung"

Um die Anzeige der gezeigten Objekte zu aktualisieren, klicken Sie in der Funktionsleiste auf das Symbol "Ansicht aktualisieren".

Nutzen der Anzeige im Register "Speicherauslastung"

Im Register "Speicherauslastung" der Programminformationen erhalten Sie eine detaillierte Auflistung für jedes verwendete Objekt und den dazugehörigen belegten Speicherbereich.

Eine Überbelegung der CPU wird ebenfalls angezeigt und kann somit vermieden werden.

Nicht kompilierte Bausteine sind identifizierbar, da deren Größe durch Fragezeichen angezeigt wird.

Siehe auch

Aufbau des Registers "Speicherauslastung" (Seite 7985)

Speicherauslastung anzeigen (Seite 7986)

Maximal verfügbaren Ladespeicher auswählen (Seite 7987)

21.5.2 Aufbau des Registers "Speicherauslastung"

Aufbau des Registers "Speicherauslastung" in den Programminformationen

Die Ansicht des Registers "Speicherauslastung" besteht aus folgenden Spalten:

Spalte	Inhalt/Bedeutung
Objekte	Zeigt unter "Details" eine Übersicht der in der CPU vorhandenen Programmier-Objekte und deren jeweilige Speicherbelegung an.
Ladespeicher	Zeigt die maximale Speicherauslastung für den Ladespeicher der CPU in Prozentwerten und absolut an. Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Ladespeicher maximal zur Verfügung stehenden Speicherplatz. Die unter "Belegt" angezeigten Werte geben Auskunft über den im Ladespeicher tatsächlich belegten Speicherplatz. Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.
Arbeitsspeicher bzw. Code- und Daten-Arbeitsspeicher	Zeigt die maximale Speicherauslastung für den Arbeitsspeicher der CPU in Prozentwerten und absolut an. Der Arbeitsspeicher ist CPU-abhängig und ist z. B. für eine CPU aus der Familie S7-400 und S7-1500 unterteilt in "Code-Arbeitsspeicher" und "Daten-Arbeitsspeicher". Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Arbeitsspeicher maximal zur Verfügung stehenden Speicherplatz. Die unter "Belegt" angezeigten Werte geben Auskunft über den im Arbeitsspeicher tatsächlich belegten Speicherplatz. Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.
Remanenzspeicher	Zeigt die maximale Speicherauslastung für den remanenten Speicherbereich der CPU in Prozentwerten und absolut an. Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Remanenzspeicher maximal zur Verfügung stehenden Speicherplatz. Die unter "Belegt" angezeigten Werte geben Auskunft über den im Remanenzspeicher tatsächlich belegten Speicherplatz. Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.

Spalte	Inhalt/Bedeutung
E/A	Zeigt die auf der CPU zur Verfügung stehenden Ein- und Ausgänge an, die in den nachfolgenden Spalten typenspezifisch angezeigt werden. Die unter "Konfiguriert" angezeigten Werte geben Auskunft über die maximal zur Verfügung stehenden Ein- und Ausgänge. Die unter "Belegt" angezeigten Werte geben Auskunft über die tatsächlich belegten Ein- und Ausgänge.
DE / DA / AE / AA	Zeigt die Anzahl der projektieren und genutzten Ein-/Ausgänge an: DE = Digitaleingänge DA = Digitalausgänge AE = Analogeingänge AA = Analogausgänge Die unter "Konfiguriert" angezeigten Werte geben Auskunft über die maximal zur Verfügung stehenden Ein- und Ausgänge. Die unter "Belegt" angezeigten Werte geben Auskunft über die tatsächlich belegten Ein- und Ausgänge.

Siehe auch

Speicherauslastung anzeigen (Seite 7986)

Maximal verfügbaren Ladespeicher auswählen (Seite 7987)

Einführung zur Speicherauslastung (Seite 7983)

21.5.3 Speicherauslastung anzeigen

Einführung

Die Speicherauslastung zeigt Ihnen an, welcher Speicherplatz von welchen Objekten in der CPU belegt wird.

Hinweis

Anzeige des belegten Ladespeichers in der CPU

Beachten Sie bitte, dass die Summe des belegten Ladespeichers nicht genau bestimmt werden kann, falls nicht alle Bausteine übersetzt worden sind.

In diesem Fall zeigt ein vorangestelltes „>“ vor der Summe an, dass der Wert für den belegten Speicherplatz größer sein kann als angezeigt, da nicht übersetzte Bausteine für die Summenbildung nicht mit berücksichtigt werden.

Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

Vorgehen

Um die Speicherauslastung für die jeweiligen Speicherbereiche der CPU anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner unterhalb der gewünschten CPU bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Speicherauslastung".

Ergebnis

Die Speicherauslastung der zugeordneten CPU wird angezeigt.

21.5.4 Maximal verfügbaren Ladespeicher auswählen

Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

Vorgehen

Um in der Speicherauslastung den maximal verfügbaren Ladespeicher auszuwählen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner unterhalb der gewünschten CPU, bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Speicherauslastung".
3. Öffnen Sie im angezeigten Dialog in der Spalte "Ladespeicher" im Feld "Gesamt" die Klappliste, indem Sie auf das Symbol klicken.
4. Wählen Sie aus der Klappliste den entsprechenden Wert für die verwendete CPU aus, indem Sie darauf klicken.

Ergebnis

Im Feld "Gesamt" wird die ausgewählte maximale Speicherauslastung angezeigt.

Hinweis

Anzeige der maximalen Speicherauslastung

Wird bei der Speicherauslastung ein Wert in rot dargestellt, so ist die verfügbare Speicherkapazität überschritten.

Passen Sie in diesem Fall die Speicherkapazität wie oben beschrieben an.

Querverweise anzeigen 22

22.1 Allgemeine Informationen zu Querverweisen

Einführung

Die Querverweisliste bietet einen Überblick über die Verwendung von Objekten und Geräten innerhalb des Anwenderprogramms.

Sie können ablesen, welche Objekte in gegenseitiger Abhängigkeit stehen, welche Beziehungen sie zueinander haben und wo sich die einzelnen Objekte befinden.

Zusätzlich können Sie direkt zur Verwendungsstelle eines Objekts springen.

Nutzen der Querverweise

Die Querverweisliste bietet Ihnen folgende Vorteile:

- Bei der Programmerstellung und bei Änderungen behalten Sie die Übersicht über die verwendeten Geräte und Objekte, wie z. B. Bausteine, Operanden, Variablen usw.
- Sie sehen, ob das jeweilige Objekt andere Objekte verwendet oder selbst verwendet wird.
- Sie können die angezeigten Querverweisinformationen nach bestimmten Kriterien filtern. Dafür stehen Ihnen vordefinierte Filter für Quell- und Referenzobjekte zur Verfügung.
- Darüber hinaus können Sie auch anwenderdefinierte Filter erstellen, um relevante Referenzen einfacher zu finden.
- Von den Querverweisen aus können Sie über den blau markierten Link direkt an die jeweilige Verwendungsstelle des ausgewählten Objekts springen.
- Bei einem Programmtest oder bei der Fehlersuche erfahren Sie z. B.:
  - In welchem Baustein und mit welchem Befehl ein Operand bearbeitet wird.
  - Welche Variable wie und wo verwendet wird.
  - Welcher Baustein von welchem anderen Baustein aufgerufen wird.
  - Querverweis-Informationen für unter- und übergeordnete Strukturen.
- Als Teil der Projektdokumentation liefern die Querverweise einen umfassenden Überblick über alle verwendeten Operanden, Speicherbereiche, Bausteine, Variablen, Bilder usw.

Hinweis

Angezeigte Objekte in der Querverweisliste

- Welche Objekte in der Querverweisliste angezeigt werden können, ist abhängig von den installierten Produkten.
- Ab dem TIA Portal V15 können auch Anweisungen mit Versionskennung bei den Querverweisen angezeigt werden. Anweisungen ohne Versionskennung werden **nicht** angezeigt.

Siehe auch

Aufbau der Querverweisliste (Seite 7991)

Querverweisliste anzeigen (Seite 7997)

Querverweise im Inspektorfenster anzeigen und bearbeiten (Seite 8022)

22.2 Aufbau der Querverweisliste

Einführung

Die Querverweisliste ermöglicht die Anzeige von Referenzobjekten und von unterlagerten Objekten zu einem in der Projektnavigation selektierten Quellobjekt.

Das Quellobjekt ist das jeweils zuerst angezeigte Objekt in der Querverweisliste.

Die Anzeige der Querverweise zu einem Objekt ist projekt- und gerätespezifisch.

Sie können auch mehrere Querverweislisten gleichzeitig geöffnet haben.

Aufbau der Querverweisliste

Wenn Sie ein Quellobjekt selektiert haben und die Querverweisliste öffnen, werden die Objekte folgendermaßen dargestellt:

- Das Quellobjekt und alle unterlagerten Objekte dieses Quellobjekts sowie die zugehörigen Referenzen werden in einer Standardansicht strukturiert angezeigt.
- Das Quellobjekt ist das Objekt, zu dem Sie die Querverweise geöffnet haben, und wird als erstes Objekt in einer grau hinterlegten Zeile angezeigt.
- Unterlagerte Objekte sind mit einem vorangestellten blauen Viereck gekennzeichnet.
- Wenn unterlagerte Objekte ebenfalls über Referenzobjekte verfügen, werden diese in der Objektspalte grau hinterlegt dargestellt und in einer hierarchischen Struktur angezeigt.
- Wenn ein Objekt mehrfach verwendet wird, so wird es auch mehrfach angezeigt.
- Am Ende einer befüllten Querverweisliste wird die Schaltfläche "Neues Quellobjekt hinzufügen" angezeigt. Wenn Sie auf diese Schaltfläche klicken, können Sie ein neues Objekt in der Querverweisliste hinzufügen.
- Im Inspektorfenster werden das Quellobjekt und die unterlagerten Objekte zugeklappt dargestellt. Unterlagerte Objekte mit Referenzobjekten werden in der Objektspalte grau hinterlegt dargestellt.

Beispiel

Die nachfolgende Grafik zeigt beispielhaft die Struktur der Querverweisliste zu einem Quellobjekt:

Anzeige der neu hinzugefügten Quellobjekte in der Querverweisliste

Objekte, die über die Schaltfläche "Neues Quellobjekt hinzufügen" eingefügt wurden, werden wie folgt dargestellt:

- Sie stehen immer am Ende der Querverweisliste und sind automatisch aufgeklappt.
- Sie werden zunächst unabhängig von der ausgewählten Filtereinstellung angezeigt.
- Nach einem Aktualisieren oder, wenn ein anderer Filter eingestellt wird, werden neu hinzugefügte Objekte richtig einsortiert und den Filtereinstellungen entsprechend dargestellt.

Spalten in der Querverweisliste

Die Querverweisliste hat folgenden Aufbau:

Spalte	Inhalt/Bedeutung
Objekt	Zeigt den Namen des Quellobjektes an, das beim Öffnen der Querverweisliste selektiert war, mit allen unterlagerten Objekten und den zugehörigen Referenzobjekten.
Verwendungsstelle	Zeigt die jeweilige Verwendungsstelle der Objekte an, z. B. ein Netzwerk.

Spalte	Inhalt/Bedeutung
Verwendungstyp	Zeigt die Beziehung zwischen dem Quellobjekt und den referenzierten Objekten an: • "Verwendet": Das Quellobjekt verwendet dieses Objekt. • "Verwendet von": Das Quellobjekt wird von diesem Objekt verwendet. • "Typ->Instanz": Das Quellobjekt ist ein Typ des referenzierten Objektes. • "Instanz->Typ": Das Quellobjekt ist eine Instanz des referenzierten Objektes. • "Gruppe->Element": Das Quellobjekt ist eine Gruppe des referenzierten Objektes. • "Element->Gruppe": Das Quellobjekt ist ein Element des referenzierten Objektes. • "Definiert": Das Quellobjekt definiert das referenzierte Objekt. • "Definiert durch": Das Quellobjekt wird durch das referenzierte Objekt definiert.
Als	Zeigt zusätzliche Informationen zu den referenzierten Objekten an.
Zugriff	Zeigt die Zugriffsart an, z. B. ob es sich um einen lesenden (R) und/ oder schreibenden (W) Zugriff auf den Operanden handelt.
Adresse	Zeigt die Adresse des jeweiligen Objektes an.
Typ	Zeigt den Typ und die Erstellsprache des Objektes an.
Gerät	Zeigt den zugehörigen Gerätenamen an, z. B. "CPU_1".
Pfad	Zeigt den Pfad des Objektes in der Projektnavigation mit Angaben über Ordner und Gruppen.
Kommentar	Zeigt den Kommentar zu einzelnen Objekten an, sofern vorhanden.

Siehe auch

Allgemeine Informationen zu Querverweisen (Seite 7989)

Querverweisliste anzeigen (Seite 7997)

22.3 Einstellungen in der Querverweisliste

Einstellungen in der Querverweisliste

In der Querverweisliste können über die Schaltflächen in der Funktionsleiste und über die Auswahl vordefinierter Filter Einstellungen vorgenommen werden.

Einstellungen über die Schaltflächen in der Querverweisliste

Folgende Einstellungen über die Schaltflächen sind möglich:

- **Schaltfläche "Aktualisieren"**
  Aktualisiert die angezeigten Inhalte in der geöffneten Querverweisliste.
  Wenn Sie diese Schaltfläche drücken, werden die angezeigten Inhalte der Querverweisliste mit den aktuellen Inhalten aus dem zugehörigen Projekt aktualisiert. Eine automatische Aktualisierung innerhalb der Querverweisliste erfolgt nicht, wenn sich die Datenhaltung im Projekt geändert hat.
- **Schaltfläche "Alles aufklappen"**
  Erweitert die Einträge in der aktuellen Querverweisliste, indem alle unterlagerten Objekte aufgeklappt werden.
- **Schaltfläche "Alles zuklappen"**
  Reduziert die Einträge in der aktuellen Querverweisliste, indem die unterlagerten Objekte zugeklappt werden.
- **Schatlfäche "Anordnung merken"**
  Speichert die aktuell vorgenommene Anordnung in der Querverweisliste.
  Dazu gehören z. B. die angezeigten Spalten, die Breite der Spalten und die Anordnung der Spalten.
  Die jeweiligen Filtereinstellungen in der Querverweisliste werden **nicht** gespeichert.

Filtereinstellungen in der Querverweisliste

Sie haben die Möglichkeit, vordefinierte Filter für die Anzeige von Objekten auszuwählen.

Sie haben folgende Möglichkeiten bei der Filterauswahl, die sich auf Quellobjekte beziehen:

- **Zeige Objekte mit Referenzen:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte mit vorhandenen Referenzen an. Sie ist die voreingestellte Einstellung bei jedem Öffnen der Querverweisliste.
- **Zeige Objekte ohne Referenzen:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte ohne Referenzen an. Diese nicht referenzierten Objekte erhalten in der Spalte "Verwendungsstelle" den Eintrag "keine Referenzen vorhanden". Es werden auch unterlagerte Elemente ohne Referenzen angezeigt, wenn deren Eltern-Objekte Referenzen enthalten.

- **Zeige nicht verwendete Objekte:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten, aber nicht verwendeten Objekte an. Diese nicht verwendeten Objekte erhalten in der Spalte "Verwendungsstelle" den Eintrag "Objekt ist nicht verwendet".
- **Zeige alle Objekte:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte mit oder ohne Referenzen an.

Sie haben folgende Möglichkeiten bei der Filterauswahl, die sich auf Referenzobjekte beziehen:

- **Zeige nur "verwendet":**
  Diese Einstellung zeigt nur die verwendeten Referenzobjekte mit dem Verwendungstyp "verwendet", "Element->Gruppe", "Instanz->Typ" und "definiert".
- **Zeige nur "verwendet von":**
  Diese Einstellung zeigt alle Referenzobjekte mit dem Verwendungstyp "verwendet von", "Gruppe->Element","Typ->Instanz" und "definiert durch".
- **Zeige nur lokale Referenzen:**
  Diese Einstellung zeigt nur lokale Referenzen an, die zu einem bestimmten Gerät gehören (z. B. zur CPU_1).

Sortieren in der Querverweisliste

Die Einträge werden abhängig vom zugehörigen Objekt, dem zugehörigen Gerät und vom Objekt-Typ in alphabetischer Reihenfolge dargestellt. Zusammengehörige Objekte werden nacheinander dargestellt.

Folgende Sortierkriterien werden angewandt:

- für Quellobjekte: Sortierung nach Objekttyp, Objektname und Nummer
- für referenzierte Objekte: Sortierung nach Zuordnung zum Gerät, Objektname und Nummer

---

Hinweis

Keine Sortierung nach Namen bei Anzeige von mehr als 10.000 Objekten

Bitte beachten Sie, dass keine Sortierung nach Namen vorgenommen wird, wenn in der Querverweisliste mehr als 10.000 Objekte angezeigt werden.

In diesem Fall informiert Sie ein Bannertext darüber, dass momentan keine Sortierung für die angezeigten Objekte vorhanden ist.

Wenn Sie im Bannertext auf den Link klicken, können Sie die Sortierung nach Namen einschalten. Danach wird ein Dialog angezeigt, der Sie über den Fortschritt für die Sortierung informiert.

Anzeige der Spalten in der Querverweisliste

Die in der Querverweisliste angezeigten Spalten können über das Kontextmenü benutzerspezifisch angepasst werden.

Das Kontextmenü bietet Ihnen folgende Optionen:

- Selektierte Spalten anzeigen oder verbergen
- Alle vorhandenen Spalten anzeigen
- Spaltenbreite optimieren für die ausgewählte Spalte oder für alle Spalten.

Hinweis

Selektierte Spalten anzeigen oder verbergen

Die Spalten "Objekt" und "Verwendungsstelle" können in der Querverweisliste nicht ausgeblendet werden.

Sie können die von Ihnen vorgenommenen Einstellungen über die Schaltfläche "Anordnung merken" speichern.

Hinzufügen neuer Objekte in der Querverweisliste

Sie haben folgende Möglichkeiten, neue Quellobjekte in eine bestehende Querverweisliste einzufügen:

- Durch Klicken auf die Schaltfläche "Neues Quellobjekt hinzufügen", die am Ende der Querverweisliste angezeigt wird.

Hinweis

Mehrfaches Einfügen von identischen Objekten

Falls Sie ein identisches Objekt mehrfach in die Querverweisliste einfügen, werden die Duplikate nach dem nächsten Aktualisieren automatisch entfernt.

22.4 Querverweisliste anzeigen

Einführung

Abhängig von dem in der Projektnavigation oder in der Taskcard "Anweisungen" selektierten Objekt, werden alle zu diesem Quellobjekt gehörenden Querverweise angezeigt.

Sie haben alternativ folgende Möglichkeiten, die Querverweisliste zu öffnen:

- Wählen Sie im Menü "Werkzeuge" den Befehl "Querverweise".
- Wählen Sie im Kontextmenü den Befehl "Querverweise".
- Klicken Sie auf das Symbol für die Querverweise in der Funktionsleiste.
- Selektieren Sie ein Objekt in der Projektnavigation und drücken Sie die Taste "F11".
- Selektieren Sie eine Anweisung mit Versionskennung in der Taskcard "Anweisungen" und drücken Sie die Taste "F11".

Es werden immer die Querverweise für das jeweils selektierte Objekt und für alle ihm unterlagerten Objekte angezeigt.

Die Anzeige der Querverweise zu einem Objekt ist projekt- und gerätespezifisch. Sie können auch mehrere Querverweislisten gleichzeitig geöffnet haben.

Hinweis

Anzeige von Querverweisen

- Welche Objekte in der Querverweisliste angezeigt werden können, ist jeweils abhängig von den installierten Produkten.
- Wenn mehr als 50 Objekte selektiert wurden, werden diese in der Querverweisliste zugeklappt dargestellt.
- Wenn weniger als 50 Objekte selektiert wurden, werden diese in der Querverweisliste aufgeklappt dargestellt.

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Querverweise anzeigen

Um Querverweise in der Querverweisliste anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das gewünschte Objekt und öffnen Sie die Querverweise z. B. über das Kontextmenü.
2. Die Querverweisliste zum selektierten Objekt wird geöffnet.
   - Das jeweils selektierte Objekt ist das "Quellobjekt" und steht an oberster Stelle in der Spalte "Objekt".
   - In der Spalte "Verwendungsstelle" sehen Sie, an welcher Stelle die in der Querverweisliste angezeigten Objekte verwendet werden.
   - Die Beziehungen zwischen den Objekten werden in der Spalte "Verwendungstyp" z. B. durch "Verwendet von" und "Verwendet" angezeigt.
3. Um zur Verwendungsstelle des jeweiligen Objektes zu springen, klicken Sie auf den entsprechenden Link.
4. Über die Symbole in der Funktionsleiste können Sie folgende Aktionen durchführen:
   - Querverweisliste aktualisieren
   - Filter für die Querverweisliste festlegen
   - Einträge reduzieren
   - Einträge erweitern
   - Einstellungen für die Querverweisliste speichern
   - Überprüfen, ob überlappende Zugriffe in der aktuell geöffneten Querverweisliste vorhanden sind.
   - Anzeige der überlappenden Zugriffe in einem abgetrennten Bereich unterhalb des Arbeitsfensters
   - Neue Quellobjekte hinzufügen
5. Um die Einträge z. B. in der Spalte "Objekt" auf- bzw. absteigend zu sortieren, klicken Sie auf die jeweilige Spaltenüberschrift.

Siehe auch

Allgemeine Informationen zu Querverweisen (Seite 7989)

Aufbau der Querverweisliste (Seite 7991)

22.5 Querverweisliste sortieren

Einführung

Für die Querverweisliste wird per Voreinstellung eine Sortierung für die angezeigten Quell- und Referenzobjekte durchgeführt.

Diese Sortierung wird auch im Inspektorfenster entsprechend angezeigt.

Hinweis

Keine Sortierung nach Namen bei Anzeige von mehr als 10.000 Objekten

Bitte beachten Sie, dass keine Sortierung nach Namen möglich ist, wenn in der Querverweisliste mehr als 10.000 Objekte angezeigt werden.

In diesem Fall informiert Sie ein Bannertext darüber, dass momentan keine Sortierung für die angezeigten Objekte vorhanden ist.

Wenn Sie im Bannertext auf den Link klicken, können Sie die Sortierung einschalten. Danach wird ein Dialog angezeigt, der Sie über den Fortschritt für die Sortierung informiert.

Sortierkriterien für die Anzeige der Querverweise

Die Sortierung erfolgt nach folgenden Kriterien:

Für Quellobjekte und zugeordnete Objekte:

- nach Objekttyp: die Quellobjekte werden nach ihrem Objekttyp geordnet und zusammen angezeigt.
- nach Name: in alphanumerischer Reihenfolge

Für Referenzobjekte:

- nach Gerät: die Referenzobjekte für ein Gerät werden immer zusammen angezeigt.
- nach Objekttyp: die Referenzobjekte werden nach ihrem Objekttyp geordnet und zusammen angezeigt.
- nach Name: in alphanumerischer Reihenfolge

Für die Verwendungsstelle:

- nach Name: in alphanumerischer Reihenfolge

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Querverweisliste sortieren

Um die Querverweisliste zu sortieren, falls mehr als 10.000 Objekte angezeigt werden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Link im angezeigten Bannertext, um die Querverweisliste nach Namen zu sortieren.
   Ergebnis: Ein Dialog informiert Sie über den Fortschritt beim Sortieren der Querverweise.
2. Nach Abschluss des Sortiervorganges werden die angezeigten Objekte auch nach Namen sortiert dargestellt.

22.6 Filtermöglichlichkeiten für Querverweise

22.6.1 Einführung zum Filtern von Querverweisen

Einführung

Um die Suche nach bestimmten Querverweisen zu vereinfachen und übersichtlicher zu gestalten, haben Sie die Möglichkeit, die Querverweisliste zu filtern.

Die jeweils gültige Filtereinstellung wird in der Klappliste gelb hinterlegt dargestellt.

Es stehen Ihnen systemseitig vordefinierte Filtereinstellungen (nachfolgend Systemfilter genannt) über die Klappliste in der Funktionsleiste der Querverweise zur Verfügung.

Zusätzlich zu den systemseitig angebotenen Filtermöglichkeiten können Sie auch anwenderdefinierte Filter nach Ihren eigenen Anforderungen erstellen.

Die Systemfilter sind in der Klappliste zur Filterauswahl immer vorhanden und können nicht gelöscht werden.

Anwenderdefinierte Filter werden in der Klappliste zur Filterauswahl nach deren Erstellung unterhalb der Systemfilter angezeigt und können nach Bedarf geändert, umbenannt und gelöscht werden.

Im nachfolgenden Bild sehen Sie die immer vorhandenen Systemfilter sowie zwei anwenderdefinierte Filter "Filter_1" und "Filter_2".

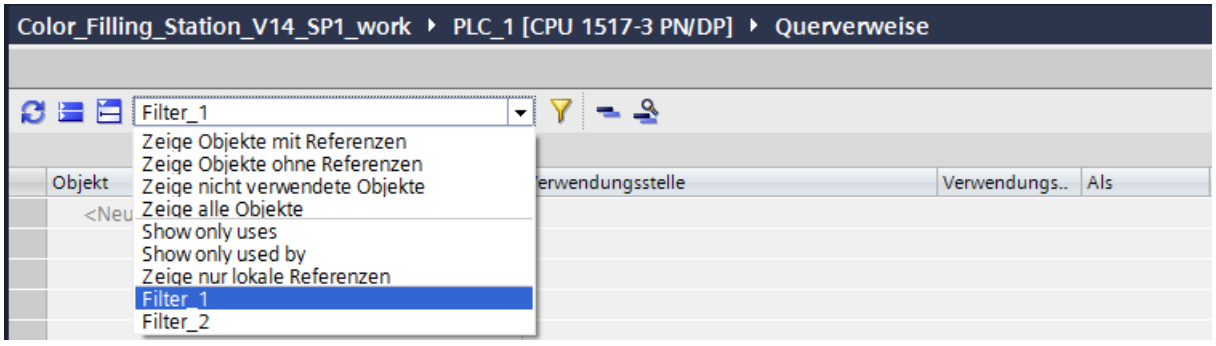

Systemseitig definierte Filtereinstellungen in der Querverweisliste

Die im oberen Teilbereich der Klappliste angezeigten Systemfilter gelten für die angezeigten Quellobjekte, die im unteren Teilbereich angezeigten Filtermöglichkeiten werden für die Referenzobjekte angewendet.

Folgende Systemfilter stehen Ihnen zur Verfügung:

- **Zeige Objekte mit Referenzen:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte mit vorhandenen Referenzen an. Sie ist die voreingestellte Einstellung bei jedem Öffnen der Querverweisliste.
- **Zeige Objekte ohne Referenzen:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte ohne Referenzen an. Diese nicht referenzierten Objekte erhalten in der Spalte "Verwendungsstelle" den Eintrag "keine Referenzen vorhanden". Es werden auch unterlagerte Elemente ohne Referenzen angezeigt, wenn deren Eltern-Objekte Referenzen enthalten.
- **Zeige nicht verwendete Objekte:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten, aber nicht verwendeten Objekte an. Diese nicht verwendeten Objekte erhalten in der Spalte "Verwendungsstelle" den Eintrag "Objekt ist nicht verwendet".
- **Zeige alle Objekte:**
  Diese Einstellung zeigt alle dem Quellobjekt unterlagerten Objekte mit oder ohne Referenzen an.
- **Zeige nur "verwendet":**
  Diese Einstellung zeigt nur die verwendeten Referenzobjekte mit dem Verwendungstyp "verwendet", "Element->Gruppe", "Instanz->Typ" und "definiert".
- **Zeige nur "verwendet von":**
  Diese Einstellung zeigt alle Referenzobjekte mit dem Verwendungstyp "verwendet von", "Gruppe->Element","Typ->Instanz" und "definiert durch".
- **Zeige nur lokale Referenzen:**
  Diese Einstellung zeigt nur lokale Referenzen an, die zu einem bestimmten Gerät gehören (z. B. zur CPU_1).

Sobald Sie einen dieser Systemfilter über die Klappliste ausgewählt haben, werden die zugehörigen Querverweise nach den ausgewählten Kriterien gefiltert und angezeigt.

Siehe hierzu auch: Querverweisliste filtern (Seite 8014)

Anwenderdefinierte Filter in der Querverweisliste

Sie können zusätzlich zu den immer vorhandenen Systemfiltern auch anwenderdefinierte Filter für die Querverweise erstellen.

Dies ist sowohl im Querverweis-Editor als auch im Inspektorfenster im Register "Info > Querverweise" möglich.

Öffnen Sie die Querverweise und klicken Sie auf das Filtersymbol in der Funktionsleiste des Querverweis-Editors.

Klicken Sie im nachfolgenden Dialog auf "Neuen Filter erstellen" und tragen Sie die gewünschten Filterkriterien ein. Klicken Sie auf "Anwenden", um die zugehörigen Querverweise nach den ausgewählten Filterkriterien anzuzeigen.

Anwenderdefinierte Filter werden in der Klappliste unterhalb der Systemfilter mit voreingestellten Namen z. B. als "Filter_1" angezeigt.

Im Gegensatz zu den Systemfiltern können anwenderdefinierten Filter beliebig umbenannt, geändert und gelöscht werden.

Ungültige Filter werden nach dem Öffnen mit rot unterlegtem Hintergrund angezeigt. Diese Filter können nicht angewendet werden.

Siehe hierzu auch: Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

22.6.2 Aufbau des Filterdialogs

Einführung

Für die Anzeige der Querverweise können Sie auch anwenderdefinierte Filter erstellen.

Nachfolgend erhalten Sie Informationen zum Aufbau des Filterdialogs und zu den verwendeten Symbolen im Filterdialog.

Aufbau des Filterdialogs

Der Filterdialog hat folgenden Aufbau:

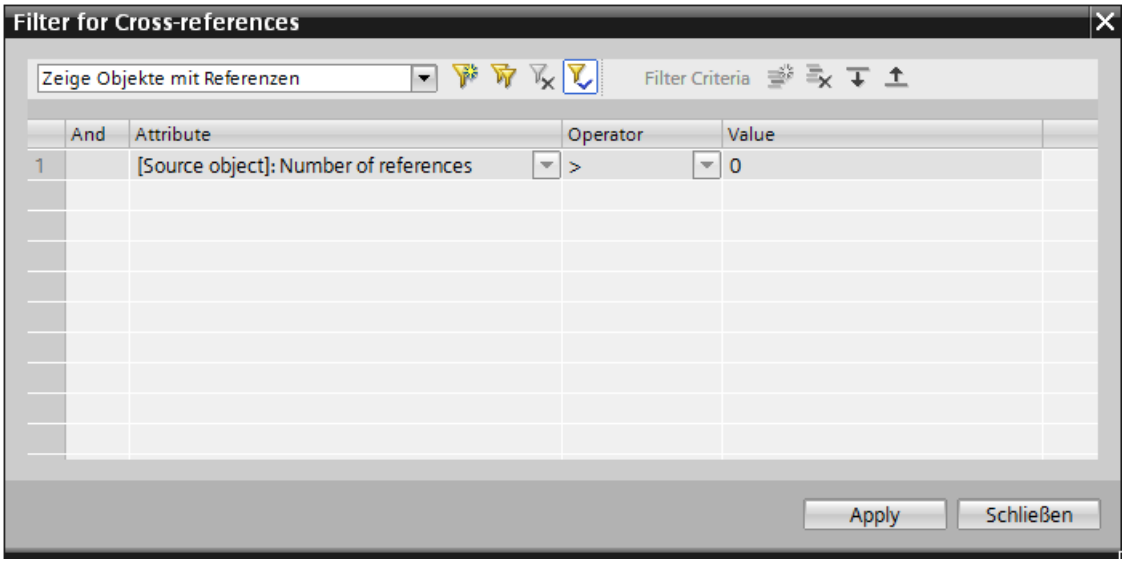

Im Filterdialog finden Sie:

- Die Klappliste zum Auswählen von vorhandenen Filtern.
  Der oben in der Klappliste angezeigte Filter ist der aus der geöffneten Querverweisliste übernommene Systemfilter.
- Die Symbole für das Erstellen, Duplizieren und Löschen von Filtern.
- Das Symbol, für das Festlegen eines Filters als Voreinstellung.
- Die Symbole für das Hinzufügen, Löschen und Verschieben von Filterkriterien.
- Die Spalten "Seq." (Sequenz), "Attribute", "Operator " und "Wert ".
  Die Spalte "Seq." zeigt die Nummer der jeweiligen Filterkriterien an.
  In den restlichen Spalten definieren Sie die benötigten Kriterien für neue Filter.

- Schaltfläche "Anwenden": Wenn Sie den Filterdialog mit "Anwenden" verlassen, werden die Änderungen im aktuellen Nutzerprofil gespeichert und im unterlagerten Editor in der Querverweisliste angewendet.
- Schaltfläche "Schließen": Wenn Sie den Filterdialog mit "Schließen" verlassen, werden die Änderungen zwar im aktuellen Nutzerprofil gespeichert, aber nicht im Editor angewendet.

Symbole im Filterdialog

Der Filterdialog zeigt folgende Symbole in der Funktionsleiste:

Symbol	Bedeutung	Aktion
	Neuen Filter erstellen	Erstellt einen neuen anwenderdefinierten Filter.
	Selektierten Filter duplizieren	Dupliziert den ausgewählten Filter, damit vorhandene Filterkriterien übernommen und weiter bearbeitet werden können.
	Filter löschen	Löscht den ausgewählten anwenderdefinierten Filter. Der Filter kann nach dem Löschen nicht wieder hergestellt werden! Hinweis: Systemfilter können nicht gelöscht werden!
	Filter als Voreinstellung übernehmen	Übernimmt den ausgewählten Filter als Voreinstellung. Dieser Filter wird beim erneuten Öffnen der Querverweise angewendet.
	Neues Kriterium einfügen	Fügt ein neues Filterkriterium ein.
	Kriterium löschten	Löscht das ausgewählte Filterkriterium.
	Kriterium nach oben verschieben	Verschiebt das ausgewählte Filterkriterium eine Zeile nach oben.
	Kriterium nach unten verschieben	Verschiebt das ausgewählte Filterkriterium um eine Zeile nach unten.

22.6.3 Attribute, Operatoren und Werte im Filterdialog

Kernaussage

Für die Anzeige der Querverweise können Sie auch anwenderdefinierte Filter erstellen.

Nachfolgend erhalten Sie Informationen zu Attributen, Operatoren und Werten im Filterdialog, sowie zu den anwendbaren Kriterien.

Attribute, Operatoren und Werte im Filterdialog

Um einen gültigen Filter zu erstellen, müssen Sie die zugehörigen Attribute, Operatoren und Werte definieren.

Wenn Sie in die Spalte "Attribut" oder "Operator" klicken, wird eine Klappliste angezeigt, aus der Sie die gewünschten Einträge auswählen können.

Die Spalte "Wert" wird rot unterlegt angezeigt, wenn Sie ein Attribut und einen Operator ausgewählt haben, aber noch keinen gültigen Wert eingetragen haben.

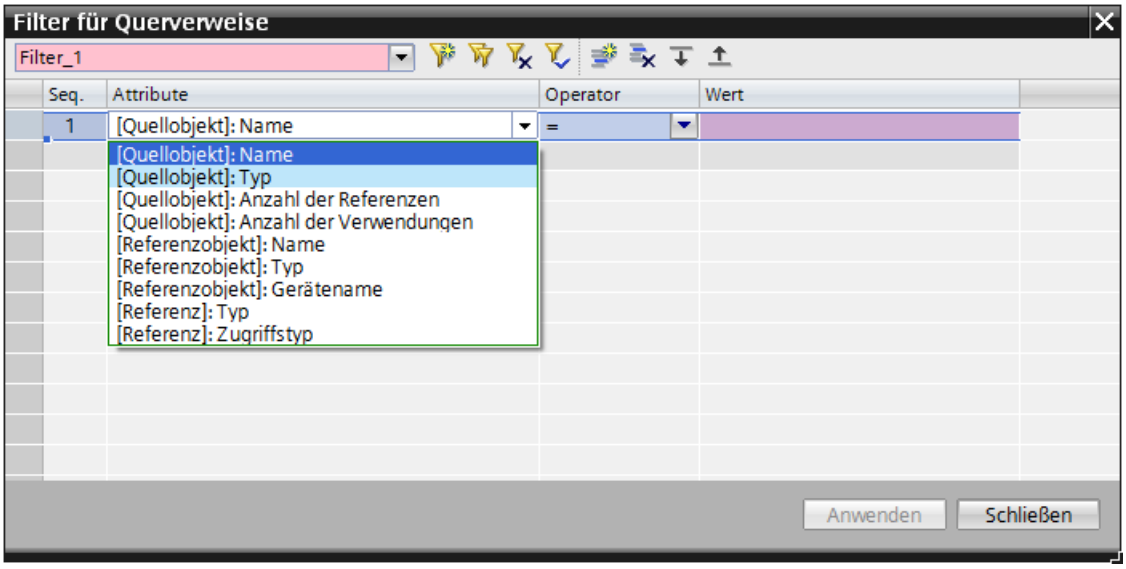

Regeln für Kriterien im Filterdialog

Die nachfolgend genannten Regeln sind beim Erstellen neuer anwenderdefinierter Filter zu beachten:

- Filterkriterien können für Quellobjekte, für Referenzobjekte und für Referenzen definiert werden.
- Quellobjekte sind nicht betroffen, wenn ein Kriterium nur ein Referenzobjekt oder eine Referenz betrifft.
- Die Filterkriterien werden in der vorgegebenen Reihenfolge angewendet.

  Hinweis

  Beachten Sie, dass das Ergebnis des ersten Kriteriums als Vorgabe für das zweite Kriterium usw. verwendet wird. Von daher ist auch die Reihenfolge der Kriterien entscheidend.

  Mit den entsprechenden Symbolen in der Funktionsleiste können Sie vorhandene Kriterien "nach oben" oder "nach unten" verschieben. Selektieren Sie dazu das Kriterium und klicken Sie dann auf das gewünschte Symbol.

- Einzelne Kriterien werden nacheinander hinzugefügt, in Abhängigkeit zur jeweiligen Sequenz-Nummer.
  Beispiel: Das erste Kriterium hat die Seq.-Nr. "1", das zweite Kriterium die "2" usw.
- Zum jeweiligen Attribut können die zulässigen Operatoren aus der Klappliste in Spalte "Operator" ausgewählt werden. Die Klappliste wird angezeigt, sobald Sie in die entsprechende Zeile im Filterdialog klicken.
- Falls ein Filter ungültig ist, weil z. B. noch kein gültiger Wert für ein Kriterium eingetragen wurde, wird die Spalte "Wert" rot hinterlegt dargestellt. Ein solcher ungültiger Filter kann zwar gespeichert, aber nicht angewendet werden. In diesem Fall können Sie den Filterdialog nur über "Schließen" verlassen.
- Falls ein Kriterium auf ein strukturiertes Quellobjekt zutrifft, so wird dieses Quellobjekt mit allen Eltern und Kindobjekten und deren Referenzen angezeigt.

- Die Elternobjekte des strukturierten Quellobjekts werden, falls sie nicht betroffen sind, nur wegen der Hierarchie und somit ohne Referenzen und in grauer Farbe dargestellt.
- Falls ein Kriterium auf ein strukturiertes Referenzobjekt zutrifft, so wird das Referenzobjekt mit allen unterlagerten Verwendungsstellen angezeigt.
- Falls ein Kriterium nur auf Elemente eines strukturierten Referenzobjekts zutrifft, so wird nur die betroffene unterlagerte Verwendungsstelle angezeigt.

22.6.4 Anwenderdefinierte Filter für Querverweise erstellen

Einführung

Für die Anzeige der Querverweise können Sie auch anwenderdefinierte Filter erstellen.

Nach dem Erstellen und Speichern sind anwenderdefinierte Filter ebenfalls über die Klappliste in der Funktionsleiste auswählbar.

Neu erstellte anwenderdefinierte Filter sind in allen aktuell geöffneten Querverweis-Editoren sofort verfügbar.

Dies ermöglicht Ihnen, die Anzahl der angezeigten Objekte in der Querverweisliste zu reduzieren und somit gezielt nach bestimmten Objekten zu suchen.

Hinweis

Anwenderdefinierte Filter erstellen

Anwenderdefinierte Filter können Sie in allen Querverweis-Editoren erstellen, auch im Inspektorfenster unter "Info > Querverweise".

Die Vorgehensweise ist in allen Querverweis-Editoren identisch.

Voraussetzung für das Erstellen eines anwenderdefinierten Filters

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Anwenderdefinierten Filter erstellen

Um anwenderdefinierte Filter für die Querverweisliste zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner Programmbausteine und öffnen Sie die Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Klicken Sie in der geöffneten Querverweisliste auf das Filtersymbol in der Funktionsleiste. Daraufhin öffnet sich der Filterdialog und zeigt die Kriterien an, die dem ausgewählten Systemfilter entsprechen.
   Beispiel: Sie hatten den Systemfilter "Zeige Objekte mit Referenzen" vor dem Öffnen des Filterdialogs in der Klappliste ausgewählt. Sie sehen nun die zu diesem Systemfilter definierten Attribute, Operatoren und Werte im Filterdialog.

3. Klicken Sie auf das Symbol "Neuen Filter erstellen".
   Daraufhin wird in der Klappliste der neue vordefinierte Filtername "Filter_1" angezeigt. Die Symbole für das Einfügen neuer Filterkriterien werden aktiviert.
4. Definieren Sie die zugehörigen Attribute, Operatoren und Werte.
   Zulässige Attribute und Operatoren können aus Klapplisten ausgewählt werden. Die Klapplisten werden angezeigt, sobald Sie in die entsprechende Zeile im Filterdialog klicken.
5. Falls Sie mehrere Kriterien benötigen, klicken Sie auf das Symbol "Neues Kriterium hinzufügen".

   Hinweis

   Beachten Sie, dass das Ergebnis des ersten Kriteriums als Vorgabe für das zweite Kriterium usw. verwendet wird. Von daher ist auch die Reihenfolge der Kriterien entscheidend.

   Mit den entsprechenden Symbolen in der Funktionsleiste können Sie vorhandene Kriterien "nach oben" oder "nach unten" verschieben. Selektieren Sie dazu das Kriterium und klicken Sie dann auf das gewünschte Symbol.

6. Tragen Sie alle benötigten Kriterien für den neuen Filter ein.
7. Vergeben Sie einen Namen für den neuen Filter, wenn Sie den voreingestellten Namen "Filter_1" nicht übernehmen möchten. Sie können den gewünschten Namen direkt in die Klappliste eintragen.

   Hinweis

   Falls Sie einen Namen in die Klappliste eintragen, der als Filtername bereits existiert, wird dieser Filter im Querverweis-Editor angewendet.

   Der neue Filter wird unter dem vom System vorgeschlagenen Namen (z. B. "Filter_10") gespeichert und kann über die Klappliste im Editor angewählt werden.

8. Klicken Sie auf "Schließen", falls Sie die neuen Filtereinstellungen zwar speichern, aber nicht sofort im unterlagerten Querverweis-Editor anzeigen möchten.
9. Klicken Sie auf "Anwenden", um die neuen Filtereinstellungen in allen unterlagerten Querverweis-Editoren sofort anzuzeigen.

Ergebnis

Beim Verlassen des Filterdialogs mit "Schließen" wird der neue Filter zwar gespeichert, aber nicht angewendet. Der neue Filter wird erst dann angewendet, wenn Sie diesen aus der Klappliste im Querverweis-Editor auswählen.

Beim Verlassen des Filterdialogs mit "Anwenden" wird der neue Filter gespeichert und in den geöffneten Querverweis-Editoren angewendet.

22.6.5 Anwenderdefinierte Filter bearbeiten

Einführung

Anwenderdefinierte Filter für die Anzeige von Querverweisen können nach dem Erstellen über die Klappliste in der Funktionsleiste des Querverweis-Editors ausgewählt werden.

Im Gegensatz zu den Systemfiltern können anwenderdefinierte Filter bearbeitet und umbenannt werden.

In der Funktionsleiste finden Sie die Symbole für das Erstellen und Bearbeiten von Filtern und von Filterkriterien.

Hinweis

Umbenennen von Filtern in den Querverweisen

Es können nur anwenderdefinierte Filter umbenannt werden, Systemfilter können nicht umbenannt werden und sind auch nicht löschbar.

Voraussetzung für das Bearbeiten eines anwenderdefinierten Filters

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Es wurde mindestens ein anwenderdefinierter Filter für die Querverweise erstellt.

Siehe hierzu: Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

Anwenderdefinierten Filter bearbeiten und umbenennen

Um anwenderdefinierte Filter für die Querverweisliste zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner Programmbausteine und öffnen Sie die Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Wählen Sie den anwenderdefinierten Filter aus der Klappliste aus, den Sie bearbeiten möchten.
3. Klicken Sie in der geöffneten Querverweisliste auf das Filtersymbol in der Funktionsleiste. Daraufhin öffnet sich der Filterdialog und zeigt die Kriterien an, die dem ausgewählten Filter entsprechen.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Neues Kriterium einfügen".
5. Definieren Sie die zugehörigen Attribute, Operatoren und Werte.
   Zulässige Attribute und Operatoren können aus den Klapplisten ausgewählt werden. Die Klapplisten werden angezeigt, sobald Sie in die entsprechende Zeile im Filterdialog klicken.

6. Tragen Sie alle gewünschten Kriterien für den neuen Filter ein.

Hinweis

Beachten Sie, dass das Ergebnis des ersten Kriteriums als Vorgabe für das zweite Kriterium usw. verwendet wird. Von daher ist auch die Reihenfolge der Kriterien entscheidend.

Mit den entsprechenden Symbolen in der Funktionsleiste können Sie vorhandene Kriterien "nach oben" oder "nach unten" verschieben. Selektieren Sie dazu das Kriterium und klicken Sie dann auf das gewünschte Symbol.

7. Klicken Sie auf "Kriterium löschen", wenn Sie ein Filterkriterium löschen möchten.
8. Vergeben Sie, falls gewünscht, einen Namen für den neuen Filter. Sie können den gewünschten Namen direkt in die Klappliste eintragen.

Hinweis

Falls Sie einen Namen in die Klappliste eintragen, der als Filtername bereits existiert, wird dieser Filter im Querverweis-Editor angewendet.

Der neue Filter wird unter dem vom System vorgeschlagenen Namen (z. B. "Filter_10") gespeichert und kann über die Klappliste im Editor angewählt werden.

9. Klicken Sie auf "Schließen", falls Sie die neuen Filtereinstellungen zwar speichern, aber nicht im unterlagerten Querverweis-Editor anwenden möchten.
10. Klicken Sie auf "Anwenden", um die neuen Filtereinstellungen in allen unterlagerten Querverweis-Editoren sofort anzuwenden.

Ergebnis

Beim Verlassen des Filterdialogs mit "Schließen" wird der geänderte Filter zwar gespeichert, aber nicht angewendet. Der Filter wird erst dann angewendet, wenn Sie im Querverweis-Editor auf die Schaltfläche "Aktualisieren" klicken.

Beim Verlassen des Filterdialogs mit "Anwenden" wird der neue Filter gespeichert und in den geöffneten Querverweis-Editoren angewendet.

Ungültige anwenderdefinierte Filter

Wenn ein anwenderdefinierter Filter durch das Bearbeiten ungültig wird, weil z. B. ein Wert noch nicht eingetragen wurde, so wird die entsprechende Zeile im Filterdialog rot hinterlegt dargestellt. Dies hat folgende Auswirkungen:

- Der Filter kann nicht angewendet werden, da die Schaltfläche "Anwenden" nicht aktiv geschaltet ist.
- Nach dem Schließen des Filterdialogs werden keine Änderungen im Editor angezeigt.
- Wenn Sie im Querverweis-Editor auf "Aktualisieren" klicken, wird der als Voreinstellung definierte Filter für die Anzeige der Querverweise angewendet.
- Ist kein anwenderdefinierter Filter als Voreinstellung definiert, so wird der Systemfilter "Zeige Objekte mit Referenzen" angewendet.

Siehe auch

Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

22.6.6 Anwenderdefinierte Filter löschen

Einführung

Anwenderdefinierte Filter für die Anzeige von Querverweisen können nach dem Erstellen über die Klappliste in der Funktionsleiste des Querverweis-Editors ausgewählt werden.

Im Gegensatz zu den Systemfiltern können anwenderdefinierte Filter auch wieder gelöscht werden.

In der Funktionsleiste finden Sie die Symbole für das Löschen von Filtern und von Filterkriterien.

Hinweis

Umbenennen und Löschen von Filtern in den Querverweisen

Es können nur anwenderdefinierte Filter gelöscht werden, Systemfilter können nicht umbenannt werden und sind auch nicht löschbar.

Voraussetzung für das Löschen eines anwenderdefinierten Filters

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Es wurde mindestens ein anwenderdefinierter Filter für die Querverweise erstellt.

Siehe hierzu: Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

Anwenderdefinierten Filter löschen

Um anwenderdefinierte Filter für die Querverweisliste zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner Programmbausteine und öffnen Sie die Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Wählen Sie den anwenderdefinierten Filter aus der Klappliste aus, den Sie löschen möchten.
3. Klicken Sie in der geöffneten Querverweisliste auf das Filtersymbol in der Funktionsleiste. Daraufhin öffnet sich der Filterdialog und zeigt die Kriterien an, die dem ausgewählten Filter entsprechen.

   Hinweis

   Beachten Sie, dass der Filter ohne weitere Rückfrage unwiederbringlich gelöscht wird, sobald Sie auf das Symbol "Filter löschen" klicken.

4. Klicken Sie in der Funktionsleiste auf das Symbol "Filter löschen", wenn Sie den ausgewählten Filter unwiederbringlich löschen möchten.
5. Verlassen Sie den Dialog mit "Schließen" oder "Anwenden".

Ergebnis

Der ausgewählte Filter wurde gelöscht und wird in der Klappliste nicht mehr angezeigt.

22.6.7 Anwenderdefinierte Filter exportieren und importieren

Einführung

Anwenderdefinierte Filter für die Anzeige von Querverweisen können nach dem Erstellen exportiert und danach wieder importiert werden.

Dies hat den Vorteil, dass Sie einen einmal erstellten anwenderdefinierten Filter auch an andere Benutzer weitergeben können.

Für den Export und Import von anwenderdefinierten Filtern nutzen Sie die Funktionalität des TIA Portals.

Voraussetzung für das Exportieren eines anwenderdefinierten Filters

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Es wurde mindestens ein anwenderdefinierter Filter für die Querverweise erstellt.

Siehe hierzu: Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

Anwenderdefinierte Filter exportieren

Um anwenderdefinierte Filter zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen exportieren...".
   Der Dialog "Exporteinstellungen" wird geöffnet.
4. Klicken Sie auf "Alles abwählen", um nur die gewünschten Filter-Einstellungen für den Export auszuwählen.
5. Wählen Sie unter "Querverweise" > "Anwenderdefinierte Filter" den gewünschten Filter für den Export aus.
   Wenn Sie die Auswahl der Einstellungen von früheren Exportvorgängen übernehmen wollen, aktivieren Sie das Optionskästchen bei "Auswahl aus einer vorherigen Exportdatei verwenden". Tragen Sie die Datei in das Feld ein.

6. Tragen Sie einen Dateinamen für die Einstellungsdatei in das Feld "Dateiname" ein.
7. Klicken Sie auf "Export", um die ausgewählten Filter zu exportieren.

Ergebnis

Die Einstellungen werden in eine Datei mit der Endung ".tps14" exportiert.

Anwenderdefinierte Filter importieren

Um anwenderdefinierte Filter zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden angezeigt.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
3. Klicken Sie im Bereich "Einstellungen importieren/exportieren" auf die Schaltfläche "Einstellungen importieren...".
   Der Dialog "Exporteinstellungen" wird geöffnet.
4. Wählen Sie die gewünschte Datei mit der Endung "-tps14" aus.

   Hinweis

   Falls bereits ein Filter mit identischem Namen existiert, wird dieser durch den Import überschrieben und steht nicht mehr zur Verfügung.

5. Klicken Sie auf "Öffnen", um die Datei zu importieren.

Ergebnis

Die ausgewählte Datei wird importiert und nach einem Neustart des TIA Portals werden die anwenderdefinierten Filtereinstellungen in den Querverweisen übernommen.

22.6.8 Filter als Voreinstellung setzen

Einführung

Für die Anzeige der Querverweise können Sie verschiedene systemseitig vordefinierte Filtereinstellungen (nachfolgend Systemfilter genannt) über die Klappliste in der Funktionsleiste auswählen. Zusätzlich können Sie anwenderdefinierte Filter nach Ihren Anforderungen erstellen.

Sie haben die Möglichkeit, einen beliebigen Filter für die Anzeige von Querverweisen als Voreistellung zu setzen.

Ist ein Filter als Voreinstellung gesetzt, wird dieser Filter beim Öffnen der Querverweise immer angewendet.

Regeln für das Voreinstellen von Filtern

Folgende Regeln gelten für das Voreinstellen von Filtern:

- Es kann immer nur ein Filter als Voreinstellung ausgewählt werden.
- Der voreingestellte Filter wird mit dem entsprechenden Symbol gekennzeichnet.
- Es können auch ungültige Filter als Voreinstellung gesetzt, aber nicht angewendet werden. Statt des ungültigen Filters wird jedoch automatisch der Systemfilter "Zeige Objekte mit Referenzen" beim Öffnen der Qerverweislsite angewendet.

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Filter als Voreinstellung setzen

Um einen Filter als Voreinstellung für die Anzeige der Querverweise zu setzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner Programmbausteine und öffnen Sie die Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Wählen Sie den Filter aus der Klappliste aus, den Sie als Voreinstellung setzen möchten.
3. Klicken Sie in der geöffneten Querverweisliste auf das Filtersymbol in der Funktionsleiste. Daraufhin öffnet sich der Filterdialog und zeigt die Kriterien an, die dem ausgewählten Filter entsprechen.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Filter als Voreinstellung übernehmen".
5. Klicken Sie auf "Schließen", falls Sie die neue Voreinstellung zwar speichern, aber nicht sofort anwenden möchten.
6. Klicken Sie auf "Anwenden", um die neuen Filtereinstellungen in allen unterlagerten Querverweis-Editoren sofort anzuwenden.

Ergebnis

Beim Verlassen des Filterdialogs mit "Schließen" wird die neue Voreinstellung zwar gespeichert, aber nicht angewendet. Die neue Voreinstellung wird erst angewendet, wenn Sie im Querverweis-Editor auf die Schaltfläche "Aktualisieren" klicken.

Beim Verlassen des Filterdialogs mit "Anwenden" wird die neue Voreinstellung gespeichert und in den geöffneten Querverweis-Editoren angewendet.

Siehe auch

Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

22.6.9 Filter duplizieren

Einführung

Für die Anzeige der Querverweise können Sie verschiedene systemseitig vordefinierte Filtereinstellungen (nachfolgend Systemfilter genannt) über die Klappliste in der Funktionsleiste auswählen. Zusätzlich können Sie anwenderdefinierte Filter nach Ihren Anforderungen erstellen.

Sie haben die Möglichkeit, einen vorhandenen Filter zu duplizieren.

Dies erleichtert Ihnen das Hinzufügen von weiteren Filterkriterien.

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Filter duplizieren

Um einen Filter zu duplizieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner Programmbausteine und öffnen Sie die Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Wählen Sie den Filter aus der Klappliste aus, den Sie duplizieren möchten.
3. Klicken Sie in der geöffneten Querverweisliste auf das Filtersymbol in der Funktionsleiste. Daraufhin öffnet sich der Filterdialog und zeigt die Kriterien an, die dem ausgewählten Filter entsprechen.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Selektierten Filter duplizieren". Daraufhin wird der ausgewählte Filter dupliziert und in der Klappliste unter neuem Namen "Filter_x" angezeigt.
5. Tragen Sie weitere Filterkriterien nach Bedarf ein oder verändern Sie bestehende Attribute, Operatoren oder Werte.

Ergebnis

Der duplizierte Filter wird beim Verlassen des Filterdialogs gespeichert und in den geöffneten Querverweis-Editoren angewendet.

22.6.10 Querverweisliste filtern

Einführung

Für die Anzeige der Querverweise können Sie verschiedene systemseitig vordefinierte Filtereinstellungen (nachfolgend Systemfilter genannt) über die Klappliste in der Funktionsleiste auswählen.

Zusätzlich zu den bereits angebotenen Filtermöglichkeiten können Sie anwenderdefinierte Filter nach Ihren Anforderungen erstellen.

Die Systemfilter sind in der Klappliste zur Filterauswahl immer vorhanden und können nicht gelöscht werden.

Anwenderdefinierte Filter werden in der Klappliste zur Filterauswahl nach deren Erstellung unterhalb der Systemfilter angezeigt und können gelöscht werden.

Siehe auch: Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Querverweisliste filtern

Um die Querverweisliste zu filtern, gehen Sie folgendermaßen vor:

1. Selektieren Sie das gewünschte Quellobjekt und öffnen Sie die dazugehörige Querverweisliste über den Kontextmenübefehl "Querverweise".
2. Alternativ können Sie die Querverweise öffnen, indem Sie Objekte in der Projektnavigation selektieren, für die Querverweise existieren, wie z. B.:
   - ein Gerät
   - den Ordner "Programmbausteine"
   - einen Baustein
3. Klicken Sie danach auf das Symbol "Querverweise" in der Funktionsleiste des TIA Portals.
4. Wählen Sie in der geöffneten Querverweisliste über die Klappliste die gewünschte Filtereinstellung aus.

Ergebnis

Die ausgewählte Filtereinstellung wird für die angezeigten Objekte in den unterlagerten Querverweis-Editoren übernommen.

22.7 Querverweisliste drucken

Einführung

Sie können die in der Querverweisliste angezeigten Inhalte für Ihre Projektdokumentation in der Druckvorschau anschauen und ausdrucken.

Dazu stehen Ihnen im TIA Portal die Befehle "Projekt" > "Druckvorschau" und "Projekt" > "Drucken" zur Verfügung.

Zum Ausdrucken gelten folgende Regeln:

- Die Querverweisliste wird immer aufgeklappt und mit allen vorhandenen Spalten (auch den ausgeblendeten Spalten) gedruckt.
- Die eingestellten Filterkriterien werden mit übernommen.
- Die Spaltenbreite wird automatisch optimiert.
- Eventuell angezeigte überlappende Zugriffe werden nicht mit ausgedruckt.
- Unabhängig davon, welches Objekt in der Querverweisliste selektiert ist, startet der Ausdruck immer mit dem angezeigten Quellobjekt.

Hinweis

Drucken einer leeren Querverweisliste

Falls die zum Drucken ausgewählte Querverweisliste keine Einträge enthält, wird eine leere Querverweisliste ausgedruckt.

Eine Querverweisliste, die ohne Inhalte ausgedruckt wird, enthält den Hinweis "Die Querverweisliste ist leer."

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Querverweisliste drucken

Um die Querverweisliste zu drucken, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein Objekt in der Querverweisliste und starten Sie die Druckvorschau über den Befehl "Projekt" > "Druckvorschau".
2. Vergewissern Sie sich, dass Sie die gewünschte Querverweisliste ausdrucken und wählen Sie die gewünschten Optionen aus.
3. Starten Sie den Ausdruck über den Befehl "Projekt" > "Drucken".
4. Wählen Sie im nachfolgenden Dialog die gewünschten Optionen für das Drucken aus und klicken Sie auf "OK".

Ergebnis

Die ausgewählte Querverweisliste wird mit den eingestellten Druckoptionen ausgedruckt.

22.8 Neues Objekt in der Querverweisliste hinzufügen

Einführung

Abhängig vom selektierten Objekt werden alle zu diesem Quellobjekt gehörenden Querverweise beim Öffnen der Querverweisliste angezeigt.

Sie haben folgende Möglichkeit, neue Quellobjekte in eine bestehende Querverweisliste einzufügen:

- Durch Klicken auf die Schaltfläche "Neues Quellobjekt hinzufügen", die am Ende der Querverweisliste angezeigt wird.

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit Referenzen enthält.

Neues Objekt in der Querverweisliste hinzufügen

Um neue Objekte in der Querverweisliste hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine bestehende Querverweisliste.
2. Klicken Sie auf die Schaltfläche "Neues Quellobjekt hinzufügen" und tragen Sie das gewünschte Objekt ein. Beim Einfügen eines bereits bestehenden Objektes werden Sie durch die intelligente Vorauswahl unterstützt.

Hinweis

Mehrfaches Einfügen von identischen Objekten

Falls Sie ein identisches Objekt mehrfach in die Querverweisliste einfügen, werden die Duplikate nach dem nächsten Aktualisieren automatisch entfernt.

Hinweis

Funktionalität für neu hinzugefügte Quellobjekte

Beachten Sie, dass die Funktionalität "Aktionen rückgängig machen und wiederholen" für neu hinzugefügte Quellobjekte nicht zur Verfügung steht.

Ergebnis

Das neu hinzugefügte Objekt wird in der Querverweisliste als letztes Objekt angezeigt.

Die aktuelle Filterauswahl wird auf das neu hinzugefügte Objekt erst nach dem nächsten Aktualisieren angewandt.

22.9 Überlappende Zugriffe in der Querverweisliste anzeigen

Einführung

In einem zweiten Fenster im unteren Teil der Querverweisliste können die überlappenden Zugriffe zu einem selektierten Objekt angezeigt werden.

Die Anzeige der überlappenden Zugriffe kann nicht gespeichert werden und steht nach einem erneuten Öffnen der Querverweisliste nicht mehr zur Verfügung.

Definition

Überlappende Zugriffe für Variablen und strukturierte Variablen werden auf Grund der absoluten Adressen ermittelt.

Man spricht dann von einem "überlappenden Zugriff", wenn die Adresse für eine Variable oder strukturierte Variable teilweise oder komplett im Adressraum des zugehörigen Quellobjektes liegt.

Hinweis

"Überlappende Zugriffe" anzeigen

Die überlappenden Zugriffe werden jeweils nur zu einem einzelnen selektierten Quellobjekt oder unterlagerten Quellobjekt in der Querverweisliste angezeigt.

Sind mehrere Objekte in den angezeigten Querverweisen selektiert, werden keine überlappenden Zugriffe angezeigt.

Beispiel

Die nachfolgende Grafik zeigt die Querverweisliste mit der Anzeige der überlappenden Zugriffe für "Tag_1":

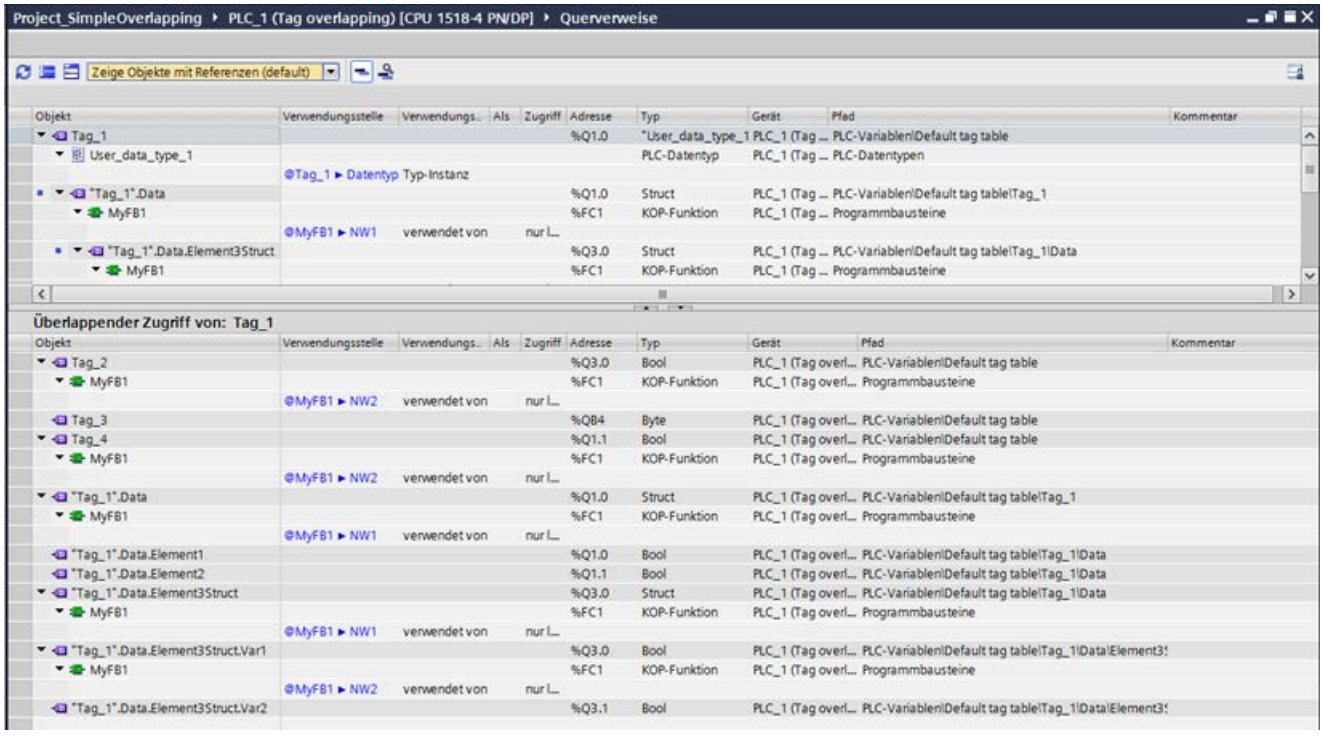

Voraussetzung

Ein Projekt ist angelegt, das Objekte mit überlappenden Zugriffen enthält.

Überlappende Zugriffe in der Querverweisliste anzeigen

Um überlappende Zugriffe in der Querverweisliste anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Querverweisliste und klicken Sie in der Funktionsleiste auf die Schaltfläche "Überlappende Zugriffe überprüfen".
   Ergebnis: Alle Objekte, die überlappende Zugriffe haben, erhalten in der Spalte "Objekt" das entsprechende Symbol zur Kennzeichnung der überlappenden Zugriffe.
2. Selektieren Sie im oberen Teil der Querverweisliste ein gekennzeichnetes Objekt und klicken Sie auf die Schaltfläche "Überlappende Zugriffe anzeigen".
   Ergebnis: Die überlappenden Zugriffe zum selektierten Objekt werden in einem zweiten Fenster mit der Überschrift "Überlappende Zugriffe von: <Objektname> angezeigt.

Hinweis

Anzeige der überlappenden Zugriffe

Die Anzeige der überlappenden Zugriffe kann nicht gespeichert werden und wird nach einem erneuten Öffnen der Querverweisliste nicht mehr angezeigt.

Um die überlappenden Zugriffe erneut anzuzeigen, führen Sie die oben beschriebene Vorgehensweise aus.

22.10 Querverweise nach Projektupgrade wieder herstellen

Einführung

Nach einem Projektupgrade auf eine höhere TIA Portal-Version haben Sie die Möglichkeit, die zugehörigen Querverweisinformationen für das aktualisierte Projekt neu zu erstellen.

Dies geschieht automatisiert am Ende des Upgrade-Vorgangs. Danach wird das aktualisierte Projekt geöffnet und die Querverweise stehen Ihnen wieder zur Verfügung.

Falls die Querverweisinformationen inkonsistent oder nicht vollständig sind, erhalten Sie einen Hinweis. Beim Öffnen der Querverweisliste wird Ihnen ein Bannertext mit einem weiterführenden Link angezeigt.

In diesem Fall können Sie selbst eine Reorganisation der Querverweisinformationen durchführen.

Voraussetzung

Sie haben ein Projektupgrade durchgeführt und die Meldung erhalten, dass die Querverweisinformationen inkonsistent oder unvollständig sind.

Querverweisinformationen neu erstellen

Um die Querverweise neu zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der angezeigten Meldung auf den Link zur Wiederherstellung der Querverweise.
   Alternativ können Sie auch den Befehl über "Extras" > "Einstellungen" > "Allgemein" > "Querverweise" aufrufen und ausführen.
   Daraufhin öffnen sich die Einstellungen zum TIA Portal unter "Querverweise".
2. Klicken Sie auf die angezeigte Schaltfläche "Querverweisinformationen neu erstellen", um den Vorgang zu starten.
3. Überprüfen Sie nach Abschluss der Wiederherstellung die Meldung im Inspektorfenster, ob der Vorgang erfolgreich war.
   Falls der Vorgang erfolgreich war, erhalten Sie eine Meldung, die Sie mit "OK" bestätigen müssen.
   Falls der Vorgang nicht erfolgreich war, erhalten Sie eine Fehlermeldung.
4. Klicken Sie in der Querverweisliste auf die Schaltfläche "Aktualisieren", damit der angezeigte Banner wieder geschlossen wird.

Ergebnis

Die Querverweisinformationen für das ausgewählte Projekt wurden neu erstellt.

22.11 Querverweise im Inspektorfenster anzeigen und bearbeiten

Einführung

Im Inspektorfenster werden die Querverweisinformationen zu einem selektierten Objekt im Register "Info > Querverweise" angezeigt.

Wenn Sie in einem beliebigen Editor im TIA Portal oder in der Taskcard "Anweisungen" bzw. "Erweiterte Anweisungen" ein Objekt selektieren, werden in diesem Register des Inspektorfensters die zum selektierten Objekt zugehörigen Referenzen angezeigt.

Hinweis

Anzeige von Anweisungen

- Ab dem TIA Portal V15 können auch Anweisungen mit Versionskennung bei den Querverweisen und im Inspektorfenster angezeigt werden.
- Anweisungen ohne Versionskennung werden **nicht** angezeigt.

Aufbau

Die Querverweisinformationen werden im Inspektorfenster tabellarisch dargestellt.

Die Querverweisliste hat folgenden Aufbau:

Spalte	Inhalt/Bedeutung
Objekt	Zeigt den Namen des Quellobjektes an, das beim Öffnen der Querverweisliste selektiert war, mit allen unterlagerten Objekten und den zugehörigen Referenzobjekten.
Verwendungsstelle	Zeigt die jeweilige Verwendungsstelle der Objekte an, z. B. ein Netzwerk.
Verwendungstyp	Zeigt die Beziehung zwischen dem Quellobjekt und den referenzierten Objekten an: • "Verwendet": Das Quellobjekt verwendet dieses Objekt. • "Verwendet von": Das Quellobjekt wird von diesem Objekt verwendet. • "Typ->Instanz": Das Quellobjekt ist ein Typ des referenzierten Objektes. • "Instanz->Typ": Das Quellobjekt ist eine Instanz des referenzierten Objektes. • "Gruppe->Element": Das Quellobjekt ist eine Gruppe des referenzierten Objektes. • "Element->Gruppe": Das Quellobjekt ist ein Element des referenzierten Objektes. • "Definiert": Das Quellobjekt definiert das referenzierte Objekt. • "Definiert durch": Das Quellobjekt wird durch das referenzierte Objekt definiert.

Spalte	Inhalt/Bedeutung
Als	Zeigt zusätzliche Informationen zu Objekten an, z. B. dass eine Variable von mehreren Geräten verwendet wird.
Zugriff	Zeigt die Zugriffsart an, z. B. ob es sich um einen lesenden (R) und/oder schreibenden (W) Zugriff auf den Operanden handelt.
Adresse	Zeigt die Adresse des jeweiligen Objektes an.
Typ	Zeigt den Typ und die Erstellsprache des Objektes an.
Gerät	Zeigt den zugehörigen Gerätenamen an, z. B. "CPU_1".
Pfad	Zeigt den Pfad des Objektes in der Projektnavigation mit Angaben über Ordner und Gruppen.
Kommentar	Zeigt den Kommentar zu einzelnen Objekten an, sofern vorhanden.

Querverweise im Inspektorfenster anzeigen

Um Querverweisinformationen im Inspektorfenster anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das gewünschte Objekt (z. B. in der Projektnavigation oder im Programmiereditor) und klicken Sie im Kontextmenü auf "Querverweisinformationen anzeigen".
   Ergebnis: Die Referenzen zum selektierten Objekt werden im Inspektorfenster im Register "Info" > "Querverweise" angezeigt.

Hinweis

Beachten Sie bitte, dass im Inspektorfenster nur die Querverweisinformationen für unterstützte Objekte angezeigt werden können.

Wenn Sie z. B. einen Ordner selektiert haben, erhalten Sie dazu keine Querverweisinformation.

Querverweise im Inspektorfenster bearbeiten

Sie haben auch die Möglichkeit, Querverweisinformationen im Inspektorfenster zu bearbeiten und zu filtern.

Die Vorgehensweise zum Filtern von Querverweisinformationen ist identisch zu der Vorgehensweise im Querverweis-Editor.

Siehe hierzu auch:

Anwenderdefinierte Filter für Querverweise erstellen (Seite 8006)

Anwenderdefinierte Filter löschen (Seite 8010)

Filter als Voreinstellung setzen (Seite 8012)

Querverweisliste filtern (Seite 8014)

Anwenderprogramm testen 23

23.1 Grundlagen zum Testen des Anwenderprogramms

Einführung

Sie haben die Möglichkeit, den Ablauf Ihres Anwenderprogramms auf dem Gerät zu testen. Sie können Signalzustände und Werte von Variablen beobachten und Variablen mit Werten vorbelegen, um damit bestimmte Situationen für den Programmablauf zu simulieren.

Voraussetzung

Ein ablauffähiges Programm muss auf das Gerät geladen sein.

Testmöglichkeiten

Folgende Testmöglichkeiten stehen Ihnen zur Verfügung:

- Testen mit dem Programmstatus
  Der Programmstatus ermöglicht es Ihnen, den Programmablauf zu beobachten. Dabei können Sie sich die Werte der Operanden und die Verknüpfungsergebnisse (VKE) anzeigen lassen und dadurch logische Fehler in Ihrem Programm finden und beheben. Innerhalb einer S7-300/400 CPU können Sie entweder mit Programmstatus oder im Einzelschrittmodus testen. Sie können jedoch nicht gleichzeitig beide Testmöglichkeiten verwenden.
- Testen mit Haltepunkten
  Bausteine, die Sie in AWL oder SCL erstellt haben, können Sie mit Haltepunkten testen. Dazu setzen Sie Haltepunkte in den Programmcode, an denen die Programmausführung gestoppt wird. Anschließend können Sie das Programm schrittweise weiter ausführen. Innerhalb einer S7-300/400 CPU können Sie entweder mit Programmstatus oder im Einzelschrittmodus testen. Sie können jedoch nicht gleichzeitig beide Testmöglichkeiten verwenden.
  Beachten Sie vor dem Testen mit Haltepunkten auf jeden Fall die folgenden Hinweise:
  Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)
  Security-Hinweise zum Testen mit Haltepunkten (Seite 8060)

- Testen mit der Beobachtungstabelle
  Die Beobachtungstabelle ermöglicht es Ihnen, die aktuellen Werte einzelner Variablen des Anwenderprogramms bzw. einer CPU zu beobachten und zu steuern. Sie können einzelne Variablen zum Testen mit Werten vorbelegen und somit verschiedene Ablaufsituationen durchspielen. Außerdem haben Sie die Möglichkeit, den Peripherieausgängen einer CPU im Betriebszustand STOP feste Werte zuzuweisen, um damit z. B. die Verdrahtung zu überprüfen.
- Testen mit der Forcetabelle
  Die Forcetabelle ermöglicht es Ihnen, die aktuellen Werte einzelner Variablen des Anwenderprogramms bzw. einer CPU zu beobachten und zu forcen. Beim Forcen überschreiben Sie einzelne Variablen mit vorgegebenen Werten. Damit können Sie Ihr Anwenderprogramm testen und verschiedene Ablaufsituationen durchspielen.
  Beachten Sie vor dem Forcen auf jeden Fall die notwendigen Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134)!

Siehe auch

Einführung zum Testen mit Programmstatus (Seite 8027)

Einführung zum Testen mit Haltepunkten (Seite 8058)

Security-Hinweise zum Testen mit Haltepunkten (Seite 8060)

Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)

Einführung zum Testen mit der Beobachtungstabelle (Seite 8084)

Einführung zum Testen mit der Forcetabelle (Seite 8115)

23.2 Testen mit Programmstatus

23.2.1 Einführung zum Testen mit Programmstatus

Funktion Programmstatus

Durch die Anzeige des Programmstatus können Sie den Programmablauf beobachten. Sie sehen die Werte der einzelnen Operanden und deren Verknüpfungsergebnisse. Dadurch können Sie feststellen, ob die Komponenten des Automatisierungssystems korrekt gesteuert werden.

Die Anzeige des Programmablaufs im Programmstatus kann sich geringfügig unterscheiden, abhängig von der eingesetzten CPU-Familie.

Testen mit Programmstatus für S7-300/400

Beim Testen mit Programmstatus kann sich die CPU-Zykluszeit verlängern, da die Erfassung der Daten die Dauer der programmierten Anweisungen beeinflusst.

Beim Ausführen folgender Testfunktionen wird für jede Online-Sitzung einmalig eine Meldung angezeigt, die Sie darauf hinweist, dass die Gefahr einer Zykluszeitüberschreitung besteht:

- Beim Testen mit Aufrufbedingungen
- Beim Testen mit Haltepunkten

Sie können diese Testfunktionen erst nach Bestätigen der Meldung ausführen.

Hinweis

Bei älteren CPUs aus den CPU-Familien S7-300/400 müssen Sie das Betriebsverhalten über die Hardware-Konfiguration ändern und anschließend die Hardware-Konfiguration in das Gerät laden. Sie haben die Möglichkeit, für das Betriebsverhalten "Prozessbetrieb" oder "Testbetrieb" einzustellen.

Testen mit Programmstatus für S7-1200/1500

Wenn Sie die Funktion „Testen mit Programmstatus“ mit einer CPU aus der Familie S7-1500 ausführen, kann sich kann sich aufgrund der schnellen Programmabarbeitung beim Beobachten von Schleifen die Zykluszeit deutlich erhöhen. Speziell bei Überschreitung des eingestellten Zeittaktes für Interrupt-OB's oder der eingestellten maximalen Zykluszeit kann die CPU eventuell in STOP gehen.

Hinweis

Um einen möglichen STOP der CPU zu vermeiden, sollten Sie darauf achten, dass beim "Testen mit Programmstatus" im aktiven Beobachtungsfenster keine programmierten Schleifen eingeblendet sind. Alternativ dazu können Sie auch die maximal zulässige Zykluszeit der CPU erhöhen.

Einschränkungen bei der Funktion "Programmstatus"

Das Beobachten von Schleifen kann die Zykluszeit deutlich erhöhen, jeweils abhängig von der Anzahl der zu beobachtenden Variablen und abhängig von der tatsächlichen Schleifendurchlaufzahl.

Um eine möglichst geringe Beeinflussung der Zykluszeit zu gewährleisten, wird die Funktion "Programmstatus" wie nachfolgend beschrieben eingeschränkt:

Für CPUs aus der Familie S7-300/400:

- Die Statusanzeige einer programmierten Schleife wird an der Rücksprungstelle abgebrochen.

Für CPUs aus der Familie S7-1200/1500:

- Das Beobachten von Schleifen ist defaultmäßig deaktiviert. Falls gewünscht, kann das Beobachten über das Kontextmenü aktiviert werden.
- Ist das Beobachten von Schleifen bei einer CPU S7-1200 aktiviert, so wird der Programmstatus für den Bereich nach einer mehrfach durchlaufenen Schleife nicht angezeigt. Verschieben Sie in diesem Fall das sichtbare Editorfenster auf den Bereich nach der Schleife, damit die Beobachtungswerte wieder angezeigt werden.
- Bei einer CPU S7-1200 ab V4.2 und bei einer CPU S7-1500 wird immer die erste Ausführung einer Anweisung beobachtet.

⚠ WARNUNG

Testen mit Programmstatus

Ein Test mit der Funktion "Programmstatus" kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen.

Stellen Sie sicher, dass keine gefährlichen Zustände eintreten können, bevor Sie einen Test mit der Funktion "Programmstatus" durchführen!

Empfohlene Vorgehensweise bei der Gefahr einer Zykluszeitüberschreitung

Wenn beim Beobachten mit Programmstatus die Gefahr einer Zykluszeitüberschreitung besteht, erhalten Sie vom System eine Meldung, die Sie darauf hinweist.

Um das Beobachten weiter durchführen zu können, müssen Sie diese Meldung mit "Ja" bestätigen.

In diesem Fall haben Sie zusätzlich folgende Möglichkeiten, ein Überschreiten der maximal zulässigen Zykluszeit beim Testen mit Programmstatus zu vermeiden:

- Reduzieren Sie die Größe des Editorfensters, damit weniger Variablen angezeigt und dadurch auch weniger Variablen beobachtet werden.
- Beobachten Sie die gewünschten Variablen innerhalb der Beobachtungstabelle.

23.2.2 Aufrufumgebung einstellen

23.2.2.1 Grundlagen zur Aufrufumgebung

Funktion

Abhängig von der jeweils verwendeten CPU können Sie die Aufrufumgebung für das Beobachten von Bausteinen und für Haltepunkte (nicht bei S7-1500) definieren. Damit legen Sie fest, unter welchen Bedingungen der Programmstatus eines Bausteins aufgezeichnet oder die Programmausführung an einem Haltepunkt unterbrochen wird.

Gültig für S7-1500er CPUs ab Firmwarestand V2.5:

Aufrufumgebung für das Beobachten von Bausteinen:

- Sobald ein Haltepunkt in einem Baustein aktiviert ist, ist es **nicht** möglich, eine Aufrufumgebung für das Beobachten einzustellen.
- Umgekehrt kann ein Haltepunkt in einem Baustein **nicht** aktiviert werden, falls bereits eine Aufrufumgebung eingestellt ist.
- Falls Sie versuchen, beide Funktionalitäten anzuwenden, erhalten Sie jeweils eine Fehlermeldung mit Hinweisen für die weitere Vorgehensweise.

Festlegen der Aufrufumgebung für das Beobachten von Bausteinen

Für das Festlegen der Aurfrufumgebung können Sie alternativ eine der folgenden Optionen aktivieren:

- Keine Bedingung definiert
  Diese Option ist voreingestellt, so lange keine andere Option ausgewählt wurde.
- Instanzdatenbaustein
  Der Programmstatus eines Funktionsbausteins wird nur dann aufgezeichnet, wenn der Funktionsbaustein mit dem ausgewählten Instanzdatenbaustein aufgerufen wird.

- Aufrufpfad
  Der Programmstatus eines Bausteins wird nur dann aufgezeichnet, wenn der Baustein von einem bestimmten Baustein oder aus einem bestimmten Pfad heraus aufgerufen wird.
- Manuell angepasster Aufrufpfad
  In dieses Feld können Sie die gewünschte Aufrufumgebung manuell eintragen. Über die Schaltfläche "Übertrage nach "manuell angepasst" wird der unter "Aufrufumgebung" ausgewählte Inhalt übernommen, den Sie bei Bedarf bearbeiten können.
  Der Programmstatus eines Bausteins wird nur dann aufgezeichnet, wenn der Baustein von einem bestimmten Baustein oder aus einem bestimmten Pfad heraus aufgerufen wird.

Festlegen der Aufrufumgebung für Haltepunkte

Beim Arbeiten mit einer CPU S7-300/400 können Sie für jeden Haltepunkt eine eigene Aufrufbedingung festlegen.

Beim Arbeiten mit einer CPU S7-1500 ab V2.5 können Sie keine Aufrufumgebung für Haltepunkte festlegen.

Wenn Sie keine Aufrufumgebung festlegen, wird der Programmstatus eines beliebigen Bausteinaufrufs innerhalb der Aufrufstruktur aufgezeichnet und die Programmausführung wird am jeweiligen Haltepunkt immer unterbrochen. Legen Sie daher, falls möglich, eine Aufrufumgebung fest, wenn Sie den Programmstatus für einen bestimmten Aufruf anzeigen möchten.

Siehe auch

Aufrufumgebung zum Beobachten von Bausteinen einstellen (Seite 8030)

23.2.2.2 Aufrufumgebung zum Beobachten von Bausteinen einstellen

Durch das Einstellen der Aufrufumgebung können Sie festlegen, wann der Programmstatus eines Bausteins aufgezeichnet wird.

Das Einstellen der Aufrufumgebung für Haltepunkte finden Sie im Kapitel "Aufrufumgebung für Haltepunkte einstellen (Seite 8065)".

Voraussetzung

- Der Baustein ist geöffnet.

Aufrufumgebung festlegen

Um die Aufrufumgebung festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Klicken Sie in der Palette "Aufrufumgebung" auf die Schaltfläche "Ändern".
   Der Dialog "Aufrufumgebung des Bausteins" wird geöffnet.
3. Wählen Sie die gewünschte Bedingung aus.
   Siehe auch: Grundlagen zur Aufrufumgebung (Seite 8029)
4. Bestätigen Sie Ihre Auswahl mit "OK".

Ergebnis

Die ausgewählte Aufrufumgebung wird in der Task Card "Testen" innerhalb der Palette "Aufrufumgebung" angezeigt. Der Programmstatus wird nun abhängig von der eingestellten Aufrufumgebung durchgeführt.

Aufrufumgebung ändern

Um die Aufrufumgebung zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   Falls bereits eine Aufrufumgebung eingestellt ist, wird diese in der Palette "Aufrufumgebung" angezeigt.
2. Klicken Sie in der Palette "Aufrufumgebung" auf die Schaltfläche "Ändern".
   Der Dialog "Aufrufumgebung des Bausteins" wird geöffnet.
3. Wählen Sie die gewünschte Bedingung aus.
   Siehe auch: Grundlagen zur Aufrufumgebung (Seite 8029)
4. Bestätigen Sie Ihre Auswahl mit "OK".

Ergebnis

Die ausgewählte Aufrufumgebung wird in der Task Card "Testen" innerhalb der Palette "Aufrufumgebung" angezeigt. Der Programmstatus wird nun abhängig von der eingestellten Aufrufumgebung durchgeführt.

Siehe auch

Grundlagen zur Aufrufumgebung (Seite 8029)

Einführung zum Testen mit Programmstatus (Seite 8027)

23.2.3 Test mit Programmstatus ein-/ausschalten

Sie können alle Bausteine beobachten, in dem Sie den Programmstatus des Bausteins einschalten. Diese Funktion steht Ihnen unabhängig von der verwendeten Programmiersprache für alle Codebausteine zur Verfügung. Für Bausteine, die mit KOP, FUP oder SCL programmiert wurden, können Sie den Programmstatus zusätzlich ab einer bestimmten Position oder für eine bestimmte Selektion einschalten. Sie können den Programmstatus für einen geöffneten Baustein direkt einschalten oder einen Baustein vom aufrufenden Baustein aus öffnen und den Programmstatus anzeigen lassen.

Ein Tooltipp zeigt Ihnen an, ob aktuell Daten zum Programmstatus empfangen werden oder nicht. Platzieren Sie dazu die Maus in der oberen rechten Ecke des Arbeitsfensters über dem blau gestrichelten "Indikator".

Hinweis

Beachten Sie die folgenden Hinweise:

- Die Ressourcen für das Testen mit dem Programmstatus sind begrenzt. Falls die Ressourcen für einen weiteren Test nicht ausreichen, werden bereits laufende Tests beendet.
- Durch das gemeinsame parallele Arbeiten auf einer CPU ist es möglich, dass ein weiterer Anwender einen Ladevorgang auf die gewählte CPU ausführt. Dadurch kann es in den folgenden Fällen vorkommen, dass Sie entweder den Test mit Programmstatus nicht starten können oder dass ein laufender Test beendet wird:
  - Durch den Ladevorgang wird der Baustein neu geladen, für den Sie den Test mit Programmstatus starten möchten oder bereits durchführen.
  - Sie verwenden für den Test mit Programmstatus einen Instanz-Datenbaustein als Aufrufumgebung und dieser ändert sich durch den Ladevorgang strukturell, z. B. durch Umnummerierung.
  - Sie verwenden für den Test mit Programmstatus einen Aufrufpfad als Testbedingung und ein Baustein innerhalb des Aufrufpfads ändert sich durch den Ladevorgang.

  Falls ein laufender Test beendet wird, erhalten Sie eine entsprechende Meldung im Inspektorfenster.

Voraussetzung

- Codebaustein: Der Code des Offline-Bausteins ist identisch mit dem Code des Online-Bausteins. In diesem Fall sind die "Code-Zeitstempel" der Bausteine identisch.
- Datenbaustein: Die Struktur des Offline-Bausteins ist identisch mit der Struktur des Online-Bausteins. In diesem Fall sind die "Schnittstellen-Zeitstempel" der Bausteine identisch.

Programmstatus direkt im Baustein ein- oder ausschalten

Um den Programmstatus direkt in einem Baustein ein- oder auszuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie den Programmstatus einschalten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
   Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen. Siehe auch: Online verbinden und Online-Verbindung trennen
3. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

Programmstatus ab einer bestimmten Stelle in einem Netzwerk ein- oder ausschalten

Um den Programmstatus für KOP und FUP ab einer bestimmten Stelle einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie den Programmstatus einschalten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
3. Klicken Sie mit der rechten Maustaste auf die Variable, ab der Sie den Programmstatus einschalten möchten.
4. Wählen Sie im Kontextmenü "Steuern > Beobachten ab hier".
5. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

Um den Programmstatus in SCL ab einer bestimmten Stelle einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie den Programmstatus einschalten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
3. Selektieren Sie die Variable, ab der Sie den Programmstatus einschalten möchten.
4. Klicken Sie danach in der Funktionsleiste auf die Schaltfläche "Programmstatus ab der selektierten Zeile anzeigen".
5. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".
6. Um die Übersichtlichkeit zu erhöhen können über die Kontextmenübefehle "Aufklappen" und "Zuklappen", bzw. "Alle Aufklappen" und "Alle zuklappen" einzelne oder alle geschlossenen Codeabschnitte innerhalb des Bausteins aufgeklappt bzw. wieder zugeklappt werden.

Programmstatus für selektierte Variablen ein- oder ausschalten

Um den Programmstatus für KOP und FUP für selektierte Variablen einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie den Programmstatus einschalten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
3. Selektieren Sie die Variablen, für die Sie den Programmstatus einschalten möchten.
4. Wählen Sie im Kontextmenü "Steuern > Selektion beobachten".
5. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

Programmstatus vom aufrufenden Baustein aus einschalten

Um den Programmstatus für einen Baustein vom aufrufenden Baustein (z. B. OB1) aus einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den aufrufenden Baustein.
2. Klicken Sie mit der rechten Maustaste auf den Bausteinaufruf.
3. Wählen Sie im Kontextmenü den Befehl "Öffnen und beobachten".
   Der Baustein wird im Programmiereditor geöffnet. Anschließend wird eine Online-Verbindung hergestellt und der Programmstatus angezeigt. Dabei ist der aufrufende Baustein im Aufrufpfad eingestellt.

Ergebnis

Wenn Sie die Anzeige des Programmstatus einschalten, wird eine Online-Verbindung hergestellt und der Programmstatus angezeigt. Wenn Sie die Anzeige des Programmstatus ausschalten, können Sie gleichzeitig die Online-Verbindung trennen.

Unterhalb der Bausteinschnittstelle wird der Aufrufpfad des Bausteins angezeigt. Bei Bedarf können Sie am rechten Rand bei „Testen“ und „Optionen“ im Abschnitt „Aufrufumgebung“ die Aufrufumgebung ändern. Für CPUs der Baureihe S7-1200/1500 wird zusätzlich am rechten Rand die „Aufrufhierarchie“ angezeigt. Sie können den aufrufenden Baustein öffnen, indem Sie auf den Link klicken.

Siehe auch

Aufrufumgebung einstellen (Seite 8029)

23.2.4 Schleifen beobachten

Einführung zum Beobachten von Schleifen

Durch die Anzeige des Programmstatus können Sie den Programmablauf innerhalb eines Bausteins beobachten. Dadurch erhalten Sie eine Übersicht über die Werte der einzelnen Operanden und der Verknüpfungsergebnisse und können feststellen, ob die Komponenten des Automatisierungssystems korrekt gesteuert werden.

Sind die Werte in einer helleren Schrift mit geringerer Sättigung dargestellt, so stammen diese Werte nicht aus dem aktuellen Zyklus.

Hinweis

Das Beobachten von Schleifen kann die Zykluszeit der CPU deutlich erhöhen, jeweils abhängig von der Anzahl der zu beobachtenden Variablen und abhängig von der tatsächlichen Schleifendurchlaufzahl.

Für das Beobachten von Schleifen gelten besondere Regeln, abhängig von der jeweils eingesetzten CPU und der verwendeten Programmiersprache.

Regeln für das Beobachten von Schleifen (S7-300/400)

Um eine möglichst geringe Beeinflussung der Zykluszeit zu gewährleisten, wird die Funktion "Programmstatus" beim Beobachten von Schleifen mit einer CPU aus der Familie S7-300/400 bei allen Programmiersprachen wie folgt eingeschränkt:

- Die Statusanzeige einer programmierten Schleife wird an der Rücksprungstelle abgebrochen.

Regeln für das Beobachten von Schleifen in AWL (S7-1500)

Beim Beobachten von Schleifen in AWL ist die Anzeige des Programmstatus innerhalb programmierter Schleifen defaultmäßig ausgeschaltet. Es werden keine Werte für die Variablen innerhalb von Schleifen zwischen dem "Label" und der zugehörigen Anweisung "Loop" angezeigt, um die Zykluszeit nicht zu belasten.

Falls Sie einen AWL-Baustein mit programmierten Schleifen beobachten, informiert Sie ein Tooltipp darüber, dass das Beobachten von Schleifen momentan ausgeschaltet ist und bei Bedarf über das Kontextmenü aktiviert werden kann.

Hinweis

Beachten Sie bitte, dass nur solche Schleifen erkannt werden, die innerhalb eines Netzwerks programmiert sind.

Beobachten von Schleifen in AWL ein-/ausschalten (S7-1500)

Um das Beobachten von Schleifen in AWL einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den AWL-Baustein, den Sie beobachten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
   Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen. Siehe auch: Online verbinden und Online-Verbindung trennen
3. Aktivieren Sie das Beobachten innerhalb des Programmiereditors mit dem Kontextmenübefehl "Beobachten" > "Schleifen beobachten".
4. Bestätigen Sie die nachfolgende Abfrage mit "Ja".
   Ergebnis: Das Beobachten von Schleifen wird für den gesamten Baustein aktiviert und gilt so lange, bis das Beobachten von Schleifen wieder deaktiviert oder die Online-Verbindung zur CPU beendet wird.
5. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

Regeln für das Beobachten von Schleifen in SCL (S7-1200/1500)

Beim Beobachten von Schleifen in SCL ist die Anzeige des Programmstatus innerhalb programmierter Schleifen per Voreinstellung ausgeschaltet.

Dies betrifft die Anweisungen "FOR", "WHILE" und REPEAT-UNTIL": Innerhalb dieser Anweisungen werden keine Werte für die Variablen angezeigt, um die Zykluszeit nicht zu belasten.

Falls Sie einen SCL-Baustein mit programmierten Schleifen beobachten, informiert Sie ein Tooltipp darüber, dass das Beobachten von Schleifen momentan ausgeschaltet ist und bei Bedarf über das Kontextmenü aktiviert werden kann.

Beobachten von Schleifen in SCL ein-/ausschalten (S7-1200/1500)

Um das Beobachten von Schleifen in SCL einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den SCL-Baustein, den Sie beobachten möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
   Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen. Siehe auch: Online verbinden und Online-Verbindung trennen
3. Klicken Sie mit der rechten Maustaste innerhalb der Schleife z. B. in die Spalte "Wert" und aktivieren Sie das Beobachten mit dem Kontextmenübefehl "Schleifen beobachten".
4. Bestätigen Sie die nachfolgende Abfrage mit "Ja".
   Ergebnis: Das Beobachten von Schleifen wird für den gesamten Baustein aktiviert und gilt so lange, bis das Beobachten von Schleifen wieder deaktiviert oder die Online-Verbindung zur CPU beendet wird.
5. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

23.2.5 Strukturen beobachten

Einführung zum Beobachten von Strukturen

Sie können den Programmablauf innerhalb eines Bausteins beobachten. Dadurch erhalten Sie eine Übersicht über die Werte der einzelnen Operanden und der Verknüpfungsergebnisse und können feststellen, ob die Komponenten des Automatisierungssystems korrekt gesteuert werden.

Ab dem TIA Portal V14 ist auch das Beobachten von strukturierten PLC-Variablen möglich.

Für das Beobachten von Strukturen gelten besondere Regeln, abhängig von der jeweils eingesetzten CPU, der verwendeten Programmiersprache und den Eigenschaften der jeweiligen Strukturen.

Regeln für das Beobachten von Strukturen (S7-1200/1500)

Beim Beobachten von Strukturen werden die Werte einer strukturierten PLC-Variablen im Programmiereditor und im Inspektorfenster angezeigt, mit folgender Ausnahme:

- Strukturen, deren Elemente einstellbare Remanenz-Eigenschaften haben, können nicht beobachtet werden.

Um die Beobachtungswerte für eine Struktur anzuzeigen, müssen Sie zunächst das Beobachten über das Kontextmenü aktivieren.

Anzeige der Beobachtungswerte für Strukturen im Inspektorfenster

Wenn das Beobachten von Strukturen aktiviert und im Programmiereditor eine Strukturvariable selektiert ist, werden die zugehörigen Beobachtungswerte im Inspektorfenster im Register "Diagnose" > "Beobachtungswerte" angezeigt.

Das Inspektorfenster hat folgenden Aufbau:

- Spalte "Name" zeigt die ausgewählte Strukturvariable mit den auf der ersten Stufe unterlagerten Elementen aufgeklappt an. Wird der Name der Strukturvariablen angeklickt, so wird im Programmiereditor zur Verwendungsstelle dieser Variablen gesprungen.
- Spalte "Datentyp" zeigt die zugehörigen Datentypen an.
- Spalte "Wert" zeigt die zugehörigen Beobachtungswerte an. Werden hier keine Werte sondern Striche angezeigt, ist das Beobachten von Schleifen deaktiviert. Schalten Sie in diesem Fall das Beobachten von Schleifen über das Kontextmenü im Programmiereditor wieder ein.
- Spalten "Wert - IN" und "Wert - OUT": Diese Spalten mit den entsprechenden Werten werden nur angezeigt, wenn die ausgewählte Struktur als IN/Out-Parameter verwendet wird.
- Werden die Beobachtungswerte in einer Schrift mit nur 50 % Sättigung dargestellt, so bedeutet dies, dass diese Werte nicht aus dem aktuellen Zyklus stammen.
- Die im Inspektorfenster angezeigten Beobachtungswerte verwenden denselben Aufrufpfad, der für den Baustein voreingestellt ist.
- Die im Inspektorfenster angezeigte Tabelle mit den Beobachtungswerten kann in eine Excel-Datei kopiert werden.
- Falls im Inspektorfenster keine Beobachtungswerte angezeigt werden können, erhalten Sie eine Info, dass diese nicht verfügbar sind.

Vorgehensweise für das Beobachten von Strukturen

Um das Beobachten von Strukturen einzuschalten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
   Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen.
   Siehe auch: Online verbinden und Online-Verbindung trennen
3. Falls die zu beobachtende Struktur in einer Schleife liegt, aktivieren Sie zunächst das Beobachten von Schleifen.
   Siehe auch: Schleifen beobachten (Seite 8034)
4. Klicken Sie mit der rechten Maustaste auf eine Struktur und aktivieren Sie das Beobachten von Strukturen mit dem Kontextmenübefehl "Beobachten" > "Werte beobachten".
   Ergebnis: Die selektierte Strukturvariable wird im Inspektorfenster im Register "Diagnose" > "Beobachtungswerte" bis zur ersten Ebene aufgeklappt mit den aktuellen Werten angezeigt.

5. Klappen Sie ggf. noch geschlossene Elemente der Struktur auf, um alle unterlagerten Elemente und deren Werte anzuzeigen.
6. Um den Programmstatus auszuschalten, klicken Sie erneut auf die Schaltfläche "Beobachten ein/aus".

Hinweis

Beobachten einer Struktur innerhalb eines Arrays

Klicken Sie direkt hinter der schließenden eckigen Klammer, um die betreffende Struktur für das Beobachten auszuwählen.

Siehe auch

Schleifen beobachten (Seite 8034)

23.2.6 Bausteine während des Programmtests bearbeiten

Wenn Sie Bausteine bearbeiten, während diese beobachtet werden, wird das Beobachten unterbrochen und Sie können den Baustein offline bearbeiten. Falls der Baustein offline im Projekt nicht vorhanden ist, müssen Sie ihn zunächst vom Gerät ins Projekt laden. Nach der Bearbeitung ist ein erneutes Übersetzen und Laden des Bausteins nötig.

Vorgehen

Um Bausteine zu bearbeiten, während diese beobachtet werden, gehen Sie folgendermaßen vor:

1. Bearbeiten Sie den Baustein gemäß den Erfordernissen.
   Das Beobachten wird unterbrochen und der Baustein offline geschaltet, falls er offline bereits vorhanden ist.
2. Falls der Baustein offline nicht vorhanden ist, laden Sie ihn vom Gerät ins Projekt.
3. Übersetzen Sie den Baustein.
   Siehe auch: Bausteine übersetzen (Seite 7865)
4. Laden Sie den Baustein in das Gerät.
   Siehe auch: Bausteine laden für S7-1200/1500 (Seite 7871)

Ergebnis

Der Baustein enthält nun online und offline Ihre Änderungen. Die Online-Verbindung kann wieder hergestellt und das Beobachten fortgesetzt werden.

23.2.7 Variablen während des Beobachtens steuern

Sie haben die Möglichkeit, Variablen während des Beobachtens im Programmstatus sofort und einmalig zu steuern.

Bei booleschen Variablen können Sie zwischen den Werten "0" und "1" umschalten.

Bei Variablen, die nicht vom Datentyp "BOOL" sind, können Sie einen Steuerwert eingeben.

Beachten Sie jedoch, dass Sie keine Peripherieeingänge, z. B. über TagName:P, steuern können.

Das Steuern von Variablen beim Beobachten können Sie entweder durch einen Doppelklick auf den Onlinewert (nicht beim Datentyp BOOL) oder nach Anklicken der Variablen über das Kontextmenü durchführen.

Vorgehen

Um Variablen während des Beobachtens zu steuern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den gewünschten Baustein und starten Sie das Beobachten von Variablen, indem Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus" klicken.
2. Doppelklicken Sie auf den Onlinewert (nicht beim Datentyp BOOL), den Sie steuern möchten.
   Alternativ können Sie die gewünschte Variable anklicken und das Steuern über das Kontextmenü starten.
3. Tragen Sie im folgenden Dialog in das Feld "Steuerwert" den gewünschten Wert ein und bestätigen Sie Ihre Eingabe mit "OK".
   Oder bestätigen Sie im folgenden Dialog die Abfrage, ob der boolesche Online-Wert umgeschaltet werden soll.
   Beispiel:
   - Hat die boolesche Variable den Wert "0", so wird sie auf den Wert "1" umgeschaltet, wenn Sie die Abfrage mit "Ja" bestätigen.
   - Hat die boolesche Variable den Wert "1", so wird sie auf den Wert "0" umgeschaltet, wenn Sie die Abfrage mit "Ja" bestätigen.
   - Wenn Sie auf "Nein" klicken oder den Dialog über "Schließen" verlassen, bleibt der bestehende Wert erhalten.
4. Alternativ wählen Sie im Kontextmenü einen der folgenden Befehle:
   - "Steuern > Steuern auf 1"
     Steuert Variablen vom Datentyp "BOOL" auf den Wert "TRUE".
   - "Steuern > Steuern auf 0"
     Steuert Variablen vom Datentyp "BOOL" auf den Wert "FALSE"
   - "Steuern > Operand steuern"
5. Wenn Sie "Operand steuern" wählen, wird der Dialog "Operand steuern" geöffnet.
6. Tragen Sie im folgenden Dialog in das Feld "Steuerwert" den gewünschten Wert ein und bestätigen Sie Ihre Eingabe mit "OK".

Siehe auch

Test mit Programmstatus ein-/ausschalten (Seite 8031)

23.2.8 Anzeigeformate im Programmstatus umstellen

Einführung

Die Anzeigeformate für Variablen sind abhängig vom jeweiligen Variablentyp. Beim Beobachten haben Sie die Möglichkeit, das aktuelle Anzeigeformat über das Kontextmenü umzustellen. Die für eine Variable möglichen Anzeigeformate werden im Kontextmenü angeboten.

Sie haben die Option das Anzeigeformat für die jeweils selektierte Variable, für alle Variablen im Netzwerk oder für alle Variablen im Baustein umzustellen.

Dies ist hilfreich, wenn Sie z. B. eine hexadezimale Darstellung benötigen, um nach einem hexadezimalen Fehlercode zu suchen.

Vorgehen

Um das Anzeigeformat umzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den gewünschten Baustein im Programmiereditor.
2. Schalten Sie den Programmstatus ein, indem Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus" klicken.
   Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen.
3. Selektieren Sie die Variable, die Sie Beobachten möchten, und wählen Sie im Kontextmenü unter "Anzeigeformat" das gewünschte Anzeigeformat aus.

Ergebnis

Das Anzeigeformat wird entsprechend umgestellt.

Hinweis

Umstellen des Anzeigeformats beim Beobachten

Beachten Sie bitte, dass für nicht beschaltete Ausgänge eine Umstellung des Anzeigeformats nicht möglich ist, da hier kein Beobachtungswert ausgegeben wird.

23.2.9 Besonderheiten beim Testen von GRAPH-Programmen

23.2.9.1 Grundlagen zum Testen von GRAPH-Programmen

Einführung

Durch die Anzeige des Programmstatus eines GRAPH-Programms können Sie Logik, Konsistenz und Funktion einer Ablaufsteuerung überprüfen.

Hierbei können Sie z. B. die folgenden Fehler ermitteln:

- Programmfehler, z. B. Verschiebungen zwischen den programmierten Schritten und Bedingungen und dem tatsächlichen Prozessablauf
- Programmierfehler in der Ablaufsteuerung, z. B. falsch festgelegte Überwachungszeiten in Supervisionen

Um ein Anwenderprogramm testen zu können, muss eine Online-Verbindung zur CPU bestehen.

⚠ WARNUNG

Gefahr beim Test bei laufendem Anlagenbetrieb

Ein Test bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen.

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie einen Test durchführen.

Hinweis

Durch das gemeinsame parallele Arbeiten auf einer CPU ist es möglich, dass ein weiterer Anwender einen Ladevorgang auf die gewählte CPU ausführt. Falls durch diesen Ladevorgang Änderungen am Instanz-Datenbaustein des GRAPH-Bausteins erfolgen, wird der Test mit Programmstatus für den GRAPH-Baustein beendet und Sie erhalten eine Meldung im Inspektorfenster.

Testfunktionen von GRAPH

Für das Testen von Ablaufsteuerungen, die in GRAPH erstellt wurden, stehen die folgenden Testfunktionen zur Verfügung:

- Programmstatus für Ketten
- Programmstatus für Bedingungen und Aktionen
- Programmstatus für Interlocks und Supervision
- Programmstatus für voraus- und nachgeschaltete permanente Anweisungen

Während eines Tests haben Sie die Möglichkeit die Ablaufkette zu steuern oder auch mit dem aktuellen Prozesszustand zu synchronisieren.

Betriebsarten zum Testen von GRAPH-Programmen

Für das Testen von GRAPH-Programmen stehen Ihnen die folgenden drei Betriebsarten zur Verfügung:

- Automatischer Betrieb
  In dieser Betriebsart schaltet die Kette automatisch in den nächsten Schritt, sobald die Transition erfüllt ist.
- Halbautomatischer Betrieb
  In dieser Betriebsart schaltet die Kette in den nächsten Schritt, wenn eine der folgenden Bedingungen erfüllt ist:
  - Transition ist erfüllt.
  - Am Parameter "T_PUSH" liegt eine steigende Signalflanke an.
  - Sie schalten manuell mit der Schaltfläche "Ignoriere Transition" weiter.
- Handbetrieb
  In dieser Betriebsart können Sie entweder manuell von einem Schritt zum nächsten schalten oder gezielt einen Schritt auswählen.

Systemsynchronisation

Wenn ein Prozess im Handbetrieb in einen anderen Zustand gebracht wurde, ist er möglicherweise nicht mehr synchron zur Ablaufkette. Um den Prozess und das Programm wieder zu synchronisieren, können Sie nach Synchronisationspunkten suchen lassen und anschließend die Synchronisation durchführen.

Für das Auffinden der Synchronisationspunkte können Sie zwischen den folgenden beiden Methoden wählen:

- Vorhergehende Transition erfüllt
  Es werden alle Schritte markiert, deren vorhergehende Transition erfüllt ist.
- Interlock erfüllt
  Es werden alle Schritte markiert, deren Interlocks erfüllt sind.

Bei beiden Möglichkeiten werden nur Schritte markiert, für die die nachfolgende Transition nicht erfüllt ist. Anschließend können Sie einen oder mehrere der vorgeschlagenen Schritte aktivieren.

Lernmodus (nur S7-1500)

Über das Testen mit Programmstatus haben Sie die Möglichkeit, die maximale Schrittaktivierungszeit und Warnungszeit vom System ermitteln zu lassen. Diese Zeiten können Sie für Ihre Supervisionsbedingungen verwenden.

Hinweis

Für GRAPH-Bausteine mit einer Version kleiner 2.0 steht Ihnen der Lernmodus für das Ermitteln von "T_MAX" und "T_WARN" nicht zur Verfügung. Sie können die Version in den Eigenschaften des GRAPH-Bausteins unter "Allgemein > Baustein" ändern.

Siehe auch

Grundlagen zum Testen des Anwenderprogramms (Seite 8025)
Einführung zum Testen mit Programmstatus (Seite 8027)
Übersicht über die Testeinstellungen von GRAPH (Seite 8043)
Ablaufkette steuern (Seite 8045)
Supervisionsfehler quittieren (Seite 8045)
Testen in verschiedenen Betriebsarten (Seite 8046)
Systemsynchronisation durchführen (Seite 8047)

23.2.9.2 Testeinstellungen festlegen

Übersicht über die Testeinstellungen von GRAPH

Übersicht

Die folgende Tabelle zeigt eine Übersicht der Einstellungen für das Testen von GRAPH-Programmen:

Einstellung	Beschreibung
Bildausschnitt nachführen	Der jeweils aktuell aktive Schritt der Ablaufkette wird in der Einzelschrittansicht oder in der Kettenansicht angezeigt.
Schritte überspringen	Ein Schritt, dessen vorangehende und nachfolgende Transition gültig ist, wird nicht aktiviert. Die Standardeinstellung dieser Option wird von den entsprechenden Bausteineinstellungen übernommen. Diese Option steht Ihnen nicht zur Verfügung, wenn Sie für den GRAPH-Funktionsbaustein die Option "Speicherplatzminimierter DB erzeugen" verwenden.
Quittierpflicht bei Supervisionsfehlern	Legt fest, dass Supervisionsfehler quittiert werden müssen, bevor das Programm weiterläuft. Diese Option überschreibt die Ketteneigenschaft "Quittierpflicht bei Supervisionsfehlern" in den Einstellungen des GRAPH-Funktionsbausteins und gilt bis zum nächsten Laden des GRAPH-Instanzdatenbausteins.
Kette anhalten	Die Ablaufkette wird angehalten, auch wenn die folgende Transition erfüllt ist.
Zeiten deaktivieren	Bei Aktivierung dieser Option werden alle Schrittaktivierungszeiten gestoppt. Wenn Sie die Option deaktivieren, laufen die Zeiten weiter.
Alle Interlocks bearbeiten	Alle Interlocks, auch die der nicht aktiven Schritte, werden bearbeitet. In der Kettenansicht können Sie erkennen, welche Interlocks erfüllt sind.
Alle Transitionen bearbeiten	Es werden immer alle Transitionen bearbeitet. In der Kettenansicht können Sie erkennen, welche Transitionen erfüllt sind.
Aktionen aktivieren	Aktiviert die Aktionen in aktiven Schritten. Diese Option ist standardmäßig aktiviert. Wenn Sie diese Option deaktivieren, werden keine weiteren Aktionen in aktiven Schritten ausgeführt.

Einstellung	Beschreibung
Supervisionen aktivieren	Aktiviert die Supervisionen in aktiven Schritten. Diese Option ist standardmäßig aktiviert. Wenn Sie die Option deaktivieren, werden die Bedingungen der Supervision in aktiven Schritten ignoriert.
Interlocks aktivieren	Aktiviert die Interlocks in aktiven Schritten. Diese Option ist standardmäßig aktiviert. Wenn Sie die Option deaktivieren, werden die Bedingungen des Interlocks in aktiven Schritten ignoriert.

Siehe auch

Grundlagen zum Testen von GRAPH-Programmen (Seite 8041)

Testeinstellungen ändern (Seite 8044)

Testeinstellungen ändern

⚠ WARNUNG

Änderung der Testeinstellungen

Beachten Sie, dass Änderungen an den Testeinstellungen ungewollte Auswirkungen auf den Prozess haben können. Stellen Sie sicher, dass keine gefährlichen Zustände eintreten können, bevor Sie die Testeinstellungen ändern.

Voraussetzung

Ein GRAPH-Funktionsbaustein ist geöffnet.

Vorgehen

Um die Testeinstellungen von GRAPH zu ändern, gehen Sie folgendermaßen vor:

1. Starten Sie den Test mit Programmstatus für den GRAPH-Funktionsbaustein.
   Siehe auch: Test mit Programmstatus ein-/ausschalten (Seite 8031)
2. Öffnen Sie die Task Card "Testen".
3. Ändern Sie die Einstellungen in der Palette "Testeinstellungen".

Siehe auch

Übersicht über die Testeinstellungen von GRAPH (Seite 8043)

23.2.9.3 Ablaufkette beim Testen steuern

Ablaufkette steuern

Während des Testens mit dem Programmstatus können Sie die Ablaufkette neu initialisieren. Dadurch beginnt die Ablaufkette wieder mit den Initialschritten. Zusätzlich haben Sie die Möglichkeit, alle Schritte beim Testen zu deaktivieren.

Voraussetzung

Der Programmstatus wird angezeigt.

Ablaufkette initialisieren

Um die Ablaufkette zu initialisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Klicken Sie in der Palette "Kettensteuerung" auf "Initialisieren".
   Die Ablaufkette wird ab den Initialschritten bearbeitet.

Alle Schritte deaktivieren

Um alle Schritte beim Testen einer Ablaufkette zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Klicken Sie in der Palette "Kettensteuerung" auf "Alle deaktivieren".
   Alle Schritte der Ablaufkette werden deaktiviert.

Siehe auch

Supervisionsfehler quittieren (Seite 8045)

Testen in verschiedenen Betriebsarten (Seite 8046)

Systemsynchronisation durchführen (Seite 8047)

Grundlagen zum Testen von GRAPH-Programmen (Seite 8041)

Einführung zum Testen mit Programmstatus (Seite 8027)

Test mit Programmstatus ein-/ausschalten (Seite 8031)

Supervisionsfehler quittieren

Sie können in den Testeinstellungen festlegen, dass Supervisionsfehler beim Testen quittiert werden müssen, damit das Programm fortgesetzt wird.

Voraussetzung

Der Programmstatus wird angezeigt.

Vorgehen

Um Supervisionsfehler zu quittieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Klicken Sie in der Palette "Kettensteuerung" auf "-(V)- Quittieren.

Siehe auch

Ablaufkette steuern (Seite 8045)
Testen in verschiedenen Betriebsarten (Seite 8046)
Systemsynchronisation durchführen (Seite 8047)
Grundlagen zum Testen von GRAPH-Programmen (Seite 8041)
Übersicht über die Testeinstellungen von GRAPH (Seite 8043)
Testeinstellungen ändern (Seite 8044)
Einführung zum Testen mit Programmstatus (Seite 8027)
Test mit Programmstatus ein-/ausschalten (Seite 8031)

Testen in verschiedenen Betriebsarten

Voraussetzung

Der Programmstatus wird angezeigt.

Testen im automatischen Betrieb

Um eine Ablaufkette im automatischen Betrieb zu testen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Wählen Sie in der Palette "Kettensteuerung" die Betriebsart "Automatischer Betrieb".

Testen im halbautomatischen Betrieb

Um eine Ablaufkette im halbautomatischen Betrieb zu testen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Wählen Sie in der Palette "Kettensteuerung" die Betriebsart "Halbautomatischer Betrieb".
3. Wenn die Weiterschaltung nicht automatisch durch eine erfüllte Transition oder über den Parameter "T_PUSH" erfolgt, klicken Sie auf die Schaltfläche "Ignoriere Transition".

Testen im Handbetrieb

Um eine Ablaufkette im Handbetrieb zu testen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Wählen Sie in der Palette "Kettensteuerung" die Betriebsart "Handbetrieb".
3. Schalten Sie in einen anderen Schritt:
   - Wenn Sie den nächsten Schritt aktivieren möchten, klicken Sie auf die Schaltfläche "Nächster".
   - Wenn Sie einen beliebigen Schritt aktivieren oder deaktivieren möchten, geben Sie in das Feld "Schritt manuell auswählen" die Schrittnummer ein oder selektieren Sie einen Schritt in der Ablaufkette und klicken Sie auf die Schaltfläche "Aktivieren" oder "Deaktivieren".

Siehe auch

Ablaufkette steuern (Seite 8045)

Supervisionsfehler quittieren (Seite 8045)

Systemsynchronisation durchführen (Seite 8047)

Test mit Programmstatus ein-/ausschalten (Seite 8031)

Grundlagen zum Testen von GRAPH-Programmen (Seite 8041)

Einführung zum Testen mit Programmstatus (Seite 8027)

Systemsynchronisation durchführen

Voraussetzung

- Der Programmstatus wird angezeigt.
- Der Test erfolgt in der Betriebsart "Handbetrieb".

Vorgehen

Um eine Systemsynchronisation durchzuführen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Aktivieren Sie in der Palette "Kettensteuerung" das Optionskästchen "Synchronisation aktivieren".

3. Wählen Sie eine der beiden folgenden Methoden für das Auffinden von Synchronisationspunkten:
   - "Vorhergehende Transition erfüllt"
     Es werden alle Schritte markiert, deren vorhergehenden Transitionen erfüllt und nachfolgenden Transitionen nicht erfüllt sind.
   - "Interlock erfüllt"
     Es werden alle Schritte markiert, deren Interlocks erfüllt und nachfolgenden Transitionen nicht erfüllt sind.
4. Wählen Sie einen Schritt, mit dem die Systemsynchronisation beginnen soll. Sie können auch mehrere Schritte auswählen.
5. Klicken Sie auf die Schaltfläche "Aktivieren".

Ergebnis

Der Prozess und die Ablaufketten werden miteinander synchronisiert.

Siehe auch

Grundlagen zum Testen von GRAPH-Programmen (Seite 8041)

Ablaufkette steuern (Seite 8045)

Supervisionsfehler quittieren (Seite 8045)

Testen in verschiedenen Betriebsarten (Seite 8046)

Einführung zum Testen mit Programmstatus (Seite 8027)

Test mit Programmstatus ein-/ausschalten (Seite 8031)

Zeiten für T_MAX und T_WARN beim Testen ermitteln

Mithilfe des Lernmodus können Sie die Werte für die maximale Schrittaktivierungszeit und die Warnungszeit für die Gesamtdauer eines Schritts vom System ermitteln lassen. Dazu muss jeder Schritt mindestens einmal ausgeführt werden. Je länger der Lernmodus aktiv ist, desto häufiger wird ein Schritt ausgeführt und umso genauer sind die gelernten Werte. Die maximale Schrittaktivierungszeit und die Warnungszeit für einen Schritt setzen sich aus der gelernten Zeit und einem additiven Schwellenwert zusammen. Die additiven Schwellenwerte für die Supervision und der Warnung legen Sie selbst fest.

Nach dem Beenden des Lernmodus werden die Werte im Instanzdatenbaustein im Gerät gespeichert. Zusätzlich können Sie die Werte auch im Offline-Instanzdatenbaustein speichern. Dabei werden die Werte als Startwerte für die statischen Parameter "T_MAX" und "T_WARN" übernommen. Anschließend werden die gelernten Werte in den Vergleichsanweisungen "CMP>T_MAX" und "CMP>T_WARN" verwendet.

Sie können die Zeiten für "T_MAX" und "T_WARN", die über den Lernmodus ermittelt wurden, jederzeit auf ihre Defaultwerte zurücksetzen. Dadurch werden sowohl der Online- als auch der Offline-Datenbaustein zurückgesetzt.

Hinweis

Für GRAPH-Bausteine mit einer Version kleiner 2.0 steht Ihnen der Lernmodus für das Ermitteln von "T_MAX" und "T_WARN" nicht zur Verfügung. Sie können die Version in den Eigenschaften des GRAPH-Bausteins unter "Allgemein > Baustein" ändern.

Voraussetzung

- Der GRAPH-Funktionsbaustein ist identisch im Gerät vorhanden.
- Für mindestens einen Schritt ist eine Schrittzeit-Überwachung programmiert.

Zeiten ermitteln

Um die maximale Schrittaktivierungszeit und die Warnungszeit mithilfe des Lernmodus zu ermitteln, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein.
2. Schalten Sie den Test mit Programmstatus für den GRAPH-Funktionsbaustein ein.
   Siehe auch: Test mit Programmstatus ein-/ausschalten (Seite 8031)
3. Öffnen Sie die Task Card "Testen".
4. Geben Sie den additiven Schwellenwert für die Supervision in Prozent an.
5. Geben Sie den additiven Schwellenwert für Warnungen in Prozent an.
6. Aktivieren Sie das Optionskästchen "Lernmodus aktivieren".
7. Warten Sie, bis jeder Schritt mindestens einmal aktiv war.
8. Deaktivieren Sie das Optionskästchen "Lernmodus aktivieren".
   Der Dialog "Gelernte Zeiten speichern" wird geöffnet.
9. Wenn Sie die gelernten Zeiten im Offline-Instanzdatenbaustein speichern möchten, klicken Sie auf "OK".

   Hinweis

   Unabhängig von Ihrer Auswahl werden die gelernten Zeiten in den Online-Instanzdatenbaustein übernommen.

   Die gelernten Zeiten werden in die statischen Parameter "T_MAX" und "T_WARN" der Schritte eingetragen, die während des Lernmodus aktiv waren.

Zeiten zurücksetzen

Um die Zeiten wieder zurückzusetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den GRAPH-Funktionsbaustein.
2. Schalten Sie den Test mit Programmstatus für den GRAPH-Funktionsbaustein ein.
   Siehe auch: Test mit Programmstatus ein-/ausschalten (Seite 8031)
3. Öffnen Sie die Task Card "Testen".
4. Klicken Sie auf die Schaltfläche "Gelernte Zeiten zurücksetzen".
   Die Werte für "T_MAX" und "T_WARN" im Instanzdatenbaustein werden auf die Defaultwerte zurückgesetzt.

Siehe auch

Schrittzeit-Überwachung verwenden (Seite 7784)

23.2.10 Beispiele für die Programmstatusanzeige

23.2.10.1 Programmstatusanzeige für KOP-Programme

Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter KOP:

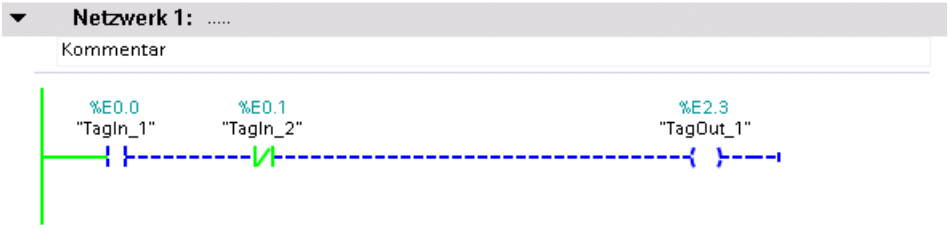

Darstellung des Programmstatus

Den Status der einzelnen Anweisungen und Linien eines Netzwerks können Sie schnell an Farbe und Art der Linien und Symbole erkennen. Die folgende Tabelle zeigt die Zuordnung von Darstellung und Status:

Darstellung	Status
Grün durchgezogen	Erfüllt
Blau gestrichelt	Nicht erfüllt
Grau durchgezogen	Unbekannt oder nicht durchlaufen
Schwarz	Nicht beschaltet

Darstellung	Status
Linien oder Parameter in einem Rahmen mit einer Sättigung von 100 %	Wert ist aktuell
Linien oder Parameter in einem Rahmen mit einer Sättigung von 50 %	Wert stammt aus einem vorhergehenden Zyklus. Die Programmstelle wurde im aktuellen Zyklus nicht durchlaufen.

23.2.10.2 Programmstatusanzeige für FUP-Programme

Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter FUP:

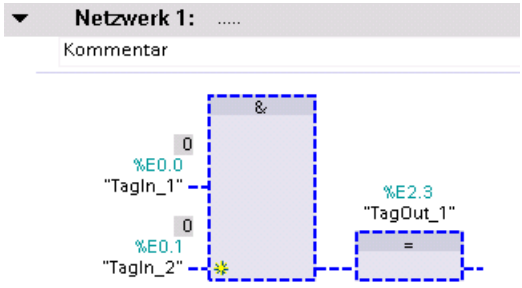

Darstellung des Programmstatus

Den Status der einzelnen Anweisungen und Linien eines Netzwerks können Sie schnell an Farbe und Art der Linien und Symbole erkennen. Die folgende Tabelle zeigt die Zuordnung von Darstellung und Status:

Darstellung	Status
Grün durchgezogen	Erfüllt
Blau gestrichelt	Nicht erfüllt
Grau durchgezogen	Unbekannt oder nicht durchlaufen
Schwarz	Nicht beschaltet
Linien oder Parameter in einem Rahmen mit einer Sättigung von 100 %	Wert ist aktuell
Linien oder Parameter in einem Rahmen mit einer Sättigung von 50 %	Wert stammt aus einem vorhergehenden Zyklus. Die Programmstelle wurde im aktuellen Zyklus nicht durchlaufen.

Die Werte der Operanden werden über dem jeweiligen Operandennamen in einem grauen Kästchen angezeigt.

Hinweis

Programmstatusanzeige bei nicht beschalteten Ausgängen

Beachten Sie bitte, dass für nicht beschaltete Ausgänge auch kein Beobachtungswert ausgegeben werden kann.

23.2.10.3 Programmstatusanzeige für AWL-Programme

Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert und in Tabellen dargestellt. Die Tabellen werden direkt neben dem AWL-Programm eingeblendet. Sie können für jede Programmzeile den Programmstatus ablesen. Die Anzeige ist abhängig von der jeweils verwendeten CPU (S7-300, S7-400 oder S7-1500).

Die Tabellen zur Anzeige im Programmstatus enthalten immer folgende Informationen:

- VKE
  In der Spalte "VKE" wird für jede Programmzeile das Verknüpfungsergebnis angezeigt. Den Wert des jeweiligen VKE können Sie an der Hintergrundfarbe der Tabellenzelle ablesen. Dabei bedeutet grün ein VKE von 1 und lila ein VKE von 0.
- Wert
  In der Spalte "Wert" wird der aktuelle Wert des Operanden angezeigt.
- Extra
  In der Spalte "Extra" werden zusätzliche Informationen in Abhängigkeit der jeweiligen Operation angezeigt, z. B. relevante Statusbits bei mathematischen Anweisungen, Zeit- bzw. Zählwerte bei Zeiten und Zählern oder die Speicheradressen bei der indirekten Adressierung.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige einer CPU S7-300 unter AWL:

Netzwerk 1: .....

Kommentar

			VKE	Wert	Extra
1	AUF "DB1"	%DB1	0	DB1	
2	L %DBW2	%DBW2	0	16#4D2	DB1
3	L %DBW4	%DBW4	0	16#3333	DB1
4	+I		0	14341	OS=0,OV=0,A0=0,A1=1
5	T "Tag_15"	%MW4	0	14341	
6					
7	L 3.0	3.0	0	3.0	
8	T "Tag_24"	%MD10	0	3.0	
9	L 5.0	5.0	0	5.0	
10	T "Tag_52"	%MD20	0	5.0	
11	L MD ["Tag_29"]	%MD26	0	16#40400000	P#10.0
12	LAR1 P#M16.0	P#M16.0	0	P#M16.0	
13	L MD [AR1 , P#4.0]	P#4.0	0	16#40A00000	P#M20.0
14	+R		0	8.0	OS=0,OV=0,A0=0,A1=1
15	T "Tag_17"	%MD30	0	8.0	
16					
17	AUF DB ["Tag_48"]	%MW14	0	DB1	
18	L DBW ["Tag_38"]	%MD60	0	16#3333	P#4.0
19	L W#16#7777	W#16#7777	0	16#7777	
20	XOW		0	16#4444	
21	T "Tag_47"	%MW70	0	16#4444	
22					
23	U "Tag_1"	%E0.0	1	1	
24	L S5T#55S	S5T#55S	1	S5T#55S	
25	SV "Tag_49"	%T5	1	S5T#47S600MS	
26	U "Tag_49"	%T5	1	1	S5T#47S600MS
27	= "Tag_3"	%A0.0	1	1	
28					
29	U "Tag_55"	%E0.2	1	1	
30	ZV "Tag_23"	%Z1	1	C#556	
31	U "Tag_23"	%Z1	1	1	C#556
32	= "Tag_3"	%A0.0	1	1	
33					
34	U "Tag_31"	%M10.0	0	0	
35	UN M ["Tag_51"]	%MD64	0	0	P#5.1
36	O DBX ["Tag_38"]	%MD60	1	1	P#4.0
37	LAR2 P#M10.0	P#M10.0	1	P#M10.0	
38	X [AR2 , P#3.4]	P#3.4	1	0	P#M13.4
39	= A ["Tag_51"]	%MD64	1	1	P#5.1
40					

23.2.10.4 Programmstatusanzeige für SCL-Programme

Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert und in einer Tabelle dargestellt. Die Tabelle wird direkt neben dem SCL-Programm eingeblendet und Sie können für jede Programmzeile den Programmstatus ablesen. Die Tabelle enthält folgende Informationen:

- Variablennamen
- Wert

Sie können die Tabelle jederzeit nach rechts oder links verschieben.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter SCL:

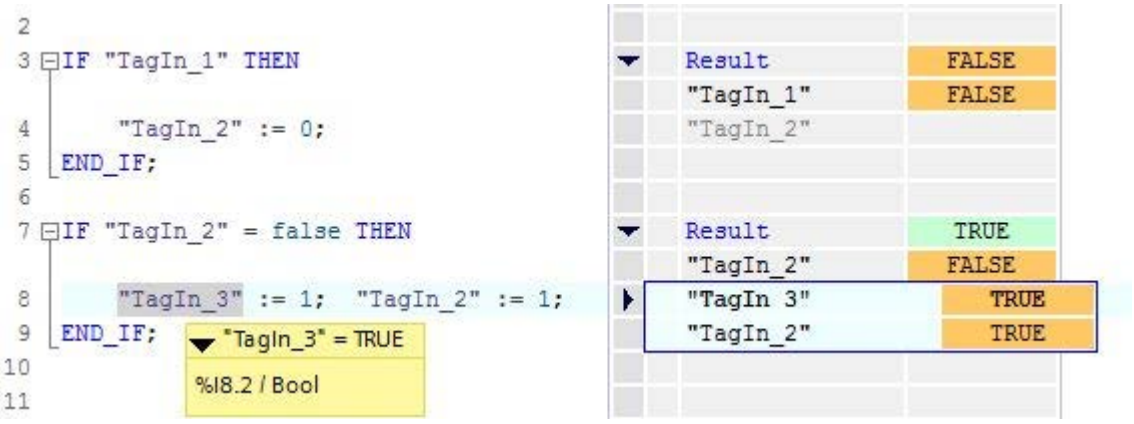

Erklärung zum Beobachten in SCL

- Wird der Code der Zeile nicht ausgeführt, wird der Variablennamen in der Wertetabelle in grauer Schriftfarbe angezeigt.
- In der letzten Spalte werden die aktuellen Werte der Variablen angezeigt. Wenn für eine Variable keine Werte angezeigt werden können, erhält die entsprechende Zelle einen gelben Hintergrund und es werden drei Fragezeichen angezeigt. Aktivieren Sie in diesem Fall das Optionskästchen "Erweiterte Statusinformationen erstellen" in den Eigenschaften des Bausteins und laden Sie den Baustein erneut in das Gerät. Anschließend werden alle Werte angezeigt.
- In der ersten Spalte finden Sie den Variablennamen, für den der aktuelle Wert angezeigt wird.
- Falls die Zeile die Anweisung "IF", "WHILE" oder "REPEAT" enthält, wird in der Zeile das Ergebnis der Anweisung mit "True" oder "False" angezeigt.
- Wenn die Zeile mehrere Variablen enthält, wird der Wert der ersten Variablen angezeigt. In beiden Fällen werden alle Variablen dieser Zeilen in einer eigenen Liste mit ihren Werten angezeigt, sobald Sie die entsprechende Zeile selektieren.
- Wenn Sie im Programmcode den Cursor in eine Variable setzen, wird diese in der Liste fett markiert. Sie können die weiteren Variablen einer Zeile auch explizit anzeigen, wenn Sie auf den Pfeil-nach-rechts klicken, der sich vor den Variablennamen der Zeilen befindet, in der es mehrere Variable gibt.

23.2.10.5 Programmstatusanzeige für GRAPH-Programme

Anzeigen im Programmstatus

Für GRAPH-Programme können Sie sich den Programmstatus in der Ketten- und in der Einzelschrittansicht und für die permanenten Anweisungen anzeigen lassen. Die Programmstatusanzeige der permanenten Anweisungen entspricht dabei der Programmstatusanzeige für KOP-/FUP-Programme. Die Anzeige des Programmstatus wird zyklisch aktualisiert.

Die folgende Tabelle zeigt die Zuordnung von Darstellung und Status:

Darstellung	Bereich	Status
Grün	Schritt, Ablaufkette	Es liegt keine Störung vor.
Hellgrün	Schritt, Ablaufkette	Es liegt keine Störung vor. Die Werte stammen aus vergangenen Zyklen.
	Bedingung	Die Transition ist erfüllt.
Rot	Schritt, Ablaufkette	Es liegt ein Supervisionsfehler vor.
Gelb	Schritt, Ablaufkette	Es liegt ein Interlockfehler vor.
Schwarz	Bedingung	Die Transition ist nicht erfüllt.

Die folgenden Bilder zeigen Beispiele für die Anzeige des Programmstatus in der Kettenansicht:

In Schritt 2 liegt ein Supervisionsfehler an. Die Transition zum Weiterschalten von Schritt 2 nach Schritt 3 ist erfüllt.

In Schritt 2 liegt ein Interlockfehler an. Die Transition zum Weiterschalten nach Schritt 3 ist erfüllt.

Das folgende Bild zeigt ein Beispiel für die Anzeige des Programmstatus in der Einzelschrittansicht für eine S7-300:

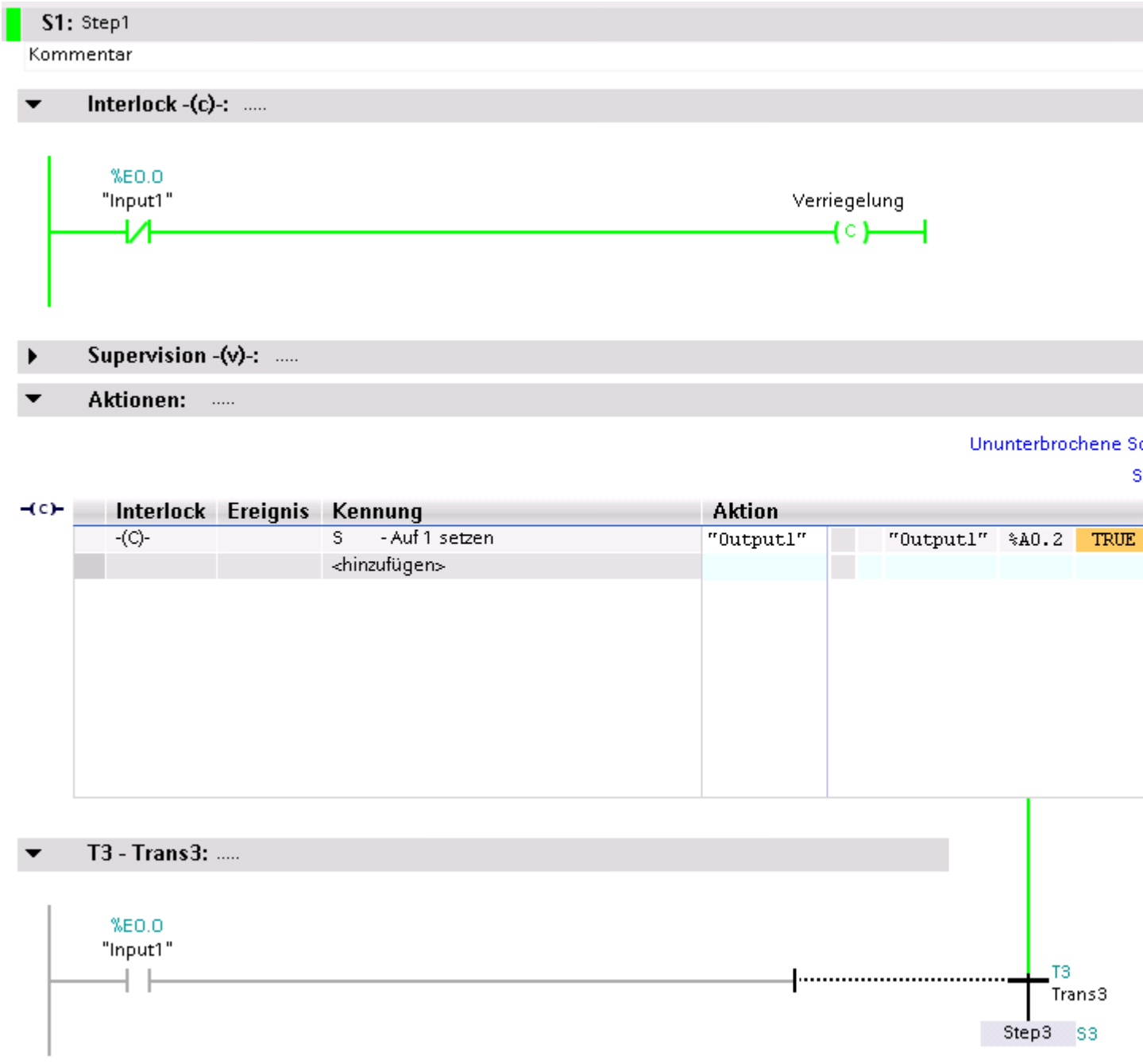

Im Schritt liegt kein Interlockfehler vor. Die nachfolgende Transition ist nicht erfüllt.

23.3 Testen mit Haltepunkten

23.3.1 Einführung zum Testen mit Haltepunkten

Einführung

Bausteine, die Sie in einer der textuellen Programmiersprachen AWL oder SCL erstellt haben, können Sie testen, indem Sie jede Anweisung einzeln, Schnitt für Schritt, ausführen lassen. Dazu setzen Sie Haltepunkte in den Programmcode, erstellen eine Online-Verbindung und aktivieren die Haltepunkte in der CPU.

Der Programmcode in AWL und SCL kann auch dann mit Haltepunkten getestet werden, wenn dieser Code in einen KOP/FUP-Baustein eingefügt ist.

Ab dem TIA Portal V15 unterstützen folgende CPU-Familien das Setzen von Haltepunkten:

- S7-300 und S7-400
- S7-1500, ab dem Firmware-Stand V2.5

Hinweis

Sicherheitshinweise beim Testen mit Haltepunkten

Da Sie beim Testen mit Haltepunkten in den laufenden Prozess eingreifen, sollten Sie vorher die Sicherheitshinweise zum Testen mit Haltepunkten kennen und beachten.

Siehe hierzu:

Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)

Security-Hinweise zum Testen mit Haltepunkten (Seite 8060)

Hinweis

Kompatibilitätsregeln für Haltepunkte

- S7-1500 CPUs mit älteren Firmware-Ständen (< 2.5) unterstützen keine Haltepunkte.
- Haltepunkte sind nicht verfügbar für Projekte mit älteren Firmware-Ständen, die auf das TIA Portal V15 hochgerüstet wurden.
- Haltepunkte werden ebenfalls nicht unterstützt, wenn eine CPU S7-1500 mit einem Firmware-Stand V2.5 auf einen älteren Firmware-Stand zurückgesetzt wird. In diesem Fall werden eventuell im Code vorhandene Haltepunkte nicht mehr angezeigt.
- Bausteine, die in älteren TIA Portal-Versionen erstellt wurden, müssen auf einer CPU S7-1500 ab dem Firmware-Stand V2.5 erneut übersetzt werden, um Haltepunkte nutzen zu können.

Testen mit Haltepunkten

Wenn Sie gültige Haltepunkte gesetzt und diese aktiviert haben, können Sie das Programm beim Testen im Einzelschrittmodus bis zum ersten Haltepunkt ausführen lassen und von dort aus mit den folgenden Funktionen schrittweise durchlaufen:

- Ausführen
  Fortsetzen bis zum nächsten aktiven Haltepunkt.
- Ausführen bis Markierung
  Fortsetzen bis zur Cursor-Position.
- Überspringen (nur für S7-300/400)
  Die aktuell markierte Anweisung wird ausgeführt.
- Hineinspringen (nur für S7-300/400)
  Sprung in einen unterlagerten Baustein bzw. Aufruf eines unterlagerten Bausteins.
- Zurückspringen (nur für S7-300/400)
  Wenn ein Bausteinaufruf bearbeitet wird und die Programmausführung im aufgerufenen Baustein an einem Haltepunkt unterbrochen wurde, können Sie mit "Zurückspringen" in den aufrufenden Baustein zurückspringen.

Sie können beliebig viele Haltepunkte im Programmcode setzen. Die Anzahl der Haltepunkte, die Sie in der CPU aktivieren können, ist jeweils CPU-abhängig. Für die einzelnen Haltepunkte können Sie bei einer CPU S7-400 über die Aufrufumgebung festlegen, wann die Programmausführung an den Haltepunkten unterbrochen werden soll.

Hinweis

- Wenn Sie die Online-Verbindung zur CPU trennen, werden die Haltepunkte deaktiviert.
- Die Haltepunkte werden nicht zusammen mit dem Projekt gespeichert.
- Wenn Sie das Projekt schließen, gehen alle Haltepunkte verloren.

Siehe auch

Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)

Regeln zum Testen mit Haltepunkten (Seite 8063)

23.3.2 Security-Hinweise zum Testen mit Haltepunkten

Haftungsausschluss für die Verwendung von Haltepunkten bei SIMATIC S7-1500

Die SIMATIC S7-1500 unterstützt die Verwendung von Haltepunkten zum gezielten Anhalten des Programmablaufs bei der Fehlersuche im Anwenderprogramm. Das Erreichen eines gesetzten Haltepunkts durch das Anwenderprogramm führt dazu, dass der Verarbeitungszyklus der SIMATIC S7-1500 an diesen Haltepunkten angehalten wird. Dabei werden die Ausgangswerte für die angeschlossen Ausgangsbaugruppen in ihrem jeweiligen Zustand gehalten und (im Gegensatz zur SIMATIC S7-300/S7-400) weder zurückgesetzt noch auf die projektierten Ersatzwerte gesetzt werden. Darüber hinaus kann die SIMATIC S7-1500 in diesem Zustand nicht mehr auf Eingangswerte von Eingangsbaugruppen reagieren.

Beachten Sie diese Information für den Betrieb Ihrer Anlage und verwenden Sie Haltepunkte nur dann, wenn das oben beschriebene Verhalten bei Ihrer Anwendung nicht zu Schäden an Umwelt, Gegenständen oder zu Verletzungen von Personen führen kann.

Jeder Kunde ist für den sachgemäßen und sicheren Betrieb der Produkte und seiner Anlage innerhalb der geltenden Vorschriften selbst verantwortlich.

Unsere Haftung, gleich aus welchem Rechtsgrund, für Ihre Verwendung von Haltepunkten ist ausgeschlossen, soweit nicht z. B. nach dem Produkthaftungsgesetz, in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird.

Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird.

Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden.

23.3.3 Sicherheitshinweise beim Testen mit Haltepunkten

Sicherheitshinweise beim Testen mit Haltepunkten

Ab dem TIA Portal V15 können Sie auf allen CPU-Familien mit Haltepunkten testen. Allerdings ist das Verhalten beim Testen mit Haltepunkten innerhalb der CPU-Familien unterschiedlich.

Da Sie beim Testen mit Haltepunkten in den laufenden Prozess eingreifen, sollten Sie folgende Sicherheitshinweise unbedingt beachten.

Testen mit Haltepunkten auf einer S7-300, S7-400 und S7-1500 CPU

Der nachfolgende Hinweis ist gültig für das Testen mit Haltepunkten auf einer S7-300, S7-400 und S7-1500 CPU!

⚠ GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Testen mit Haltepunkten" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Das Testen mit Haltepunkten sollte **nicht** im laufenden Anlagenbetrieb angewendet werden!

Das Setzen von Haltepunkten im Standard-Anwenderprogramm wird zu Fehlern im Sicherheitsprogramm führen.

Wenn Sie zum Testen dennoch Haltepunkte verwenden wollen, müssen Sie vorher den Sicherheitsbetrieb deaktivieren.

Folgende Fehler werden trotz deaktiviertem Sicherheitsbetrieb beim Einsatz mit Haltepunkten auftreten:

- Fehler bei der Kommunikation mit der F-Peripherie
- Fehler bei der sicherheitsgerichteten CPU-CPU-Kommunikation

Testen mit Haltepunkten auf einer S7-1500 CPU

Der nachfolgende Hinweis ist nur gültig für das Testen mit Haltepunkten auf einer S7-1500 CPU, ab Firmware-Stand V2.5!

⚠ GEFAHR

Personen- und Sachschaden verhindern!

- Wenn Sie die Funktion "Testen mit Haltepunkten" aktivieren, bleiben die Ausgänge freigeschaltet. Die Ausgänge behalten ihren letzten Status und werden nicht mehr aktualisiert, wenn die CPU in den Betriebszustand "HALT" geht!
- Beim Testen mit Haltepunkten werden aktive Ausgänge **nicht** deaktiviert und **nicht** auf sichere Ersatzwerte zurückgesetzt.
  Dies kann z. B. dazu führen, dass Motoren weiter laufen und Ventile offen bleiben.
- Das Testen mit Haltepunkten sollte **nicht** im laufenden Anlagenbetrieb angewendet werden!
- Insbesondere wenn ein Motion Control TO-Objekt in der CPU vorhanden ist, darf **nicht** mit Haltepunkten getestet werden. Wenn in diesem Fall ein Haltepunkt aktiviert ist und erreicht wird, geht die CPU nach dem "HALT" unmittelbar in "STOP".
  Dies kann z. B. dazu führen, dass angeschlossene Motoren und Antriebe angehalten werden, Achsreferenzierungen verloren gehen und Gleichläufe unkontrolliert abgebrochen werden.
- Das Testen mit Haltepunkten kann die Zykluszeit der CPU erhöhen.
- Bevor Sie die Funktion "Testen mit Haltepunkten" starten, sollten Sie sicherstellen, dass niemand gleichzeitig auf derselben CPU diese Funktion ebenfalls ausführt.
- Das Setzen von Haltepunkten im Standard-Anwenderprogramm wird zu Fehlern im Sicherheitsprogramm führen. Beim Testen mit Haltepunkten werden aktive Ausgänge auf der S7-1500 **nicht** deaktiviert. Damit schalten die F-Module erst nach Ablauf ihrer eingestellten Überwachungszeit automatisch in den sicheren Zustand und würden damit verzögert abschalten!

Testen mit Haltepunkten auf einer S7-1500 CPU

Der nachfolgende Hinweis ist nur gültig für das Testen mit Haltepunkten auf einer S7-1500 CPU, ab Firmware-Stand V2.5!

ACHTUNG
Fehlerhafte Anzeige von Variablenwerten beim Beobachten • Bedingt durch Optimierung des Codes können auch aktuell noch nicht erreichte Werte als bereits zugewiesen angezeigt werden. Da diese Werte bereits errechnet werden konnten, werden sie ebenfalls in schwarz dargestellt. • Beim Beobachten von Arrays innerhalb von Schleifen kann es möglicherweise vorkommen, dass falsche Werte angezeigt werden. In diesen Fällen werden nur die Werte richtig angezeigt, die innerhalb der Schleife und vor einem Breakpoint stehen. Angezeigte Werte, die nach dem Breakpoint innerhalb der Schleife stehen und der angezeigte Wert am Breakpoint selbst können möglicherweise nicht korrekt sein. Unabhängig von der inkorrekten Anzeige wird der Code korrekt ausgeführt. • Durch bestimmte Anweisungen (z. B. „Label“ und „GOTO“ in SCL und „LOOP“ in AWL) können ebenfalls Schleifen erzeugt werden. Auch in diesen Fällen werden nur die Werte richtig angezeigt, die innerhalb der Schleife und vor einem Breakpoint stehen. Angezeigte Werte, die nach dem Breakpoint innerhalb der Schleife stehen und der angezeigte Wert am Breakpoint selbst können möglicherweise nicht korrekt sein. Unabhängig von der inkorrekten Anzeige wird der Code korrekt ausgeführt.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Regeln zum Testen mit Haltepunkten (Seite 8063)

23.3.4 Regeln zum Testen mit Haltepunkten

Einführung

Abhängig von der eingesetzten CPU und der verwendeten Programmiersprache, gibt es die nachfolgend aufgelisteten Regeln für das Setzen von Haltepunkten.

Hinweis

Haltepunkte müssen nach dem Setzen aktiviert werden, bevor Sie ausgeführt werden können.

Haltepunkte werden **nicht** im Projekt gespeichert. Wenn Sie das Projekt schließen, gehen alle Haltepunkte verloren.

Regeln für das Setzen von Haltepunkten bei S7-1500 CPUs in AWL und SCL

Haltepunkte können zwar am Anfang jeder Codezeile gesetzt werden, sind jedoch eventuell nicht gültig, falls dieser Code beim Kompilieren optimiert wird.

Wird ein Haltepunkt an einer beim Kompilieren optimierten Codezeile gesetzt, so wird dieser Haltepunkt als "ungültig" gekennzeichnet und kann nicht aktiviert werden.

Deshalb ist es sinnvoll, Haltepunkte am Anfang von logisch zusammengehörigen Codeteilen zu setzen.

Gültige Haltepunkte in AWL

Gültige Haltepunkte können in AWL z. B. an folgenden Stellen gesetzt werden:

- An die erste Anweisung am Anfang eines AWL-Netzwerks.
- Vor einer Anweisung mit lesendem Operanden-Zugriff, die direkt nach einer Anweisungen mit schreibenden Operanden-Zugriff folgt.
- An der ersten Anweisung nach einem bedingten Sprung.
- Vor und nach Bausteinaufrufen.

Gültige Haltepunkte in SCL

Gültige Haltepunkte können in SCL grundsätzlich an allen Statements gesetzt werden.

Ausnahmen sind:

- Werden mehrere Anweisungen durch den Compiler zusammengefasst, so bleibt nur der erste Haltepunkt vor Beginn der ersten Anweisung gültig
- Es sind keine Haltepunkte am Anfang und am Ende von Kontrollstrukturen möglich (REPEAT, END_IF, END_FOR, END_WHILE, END_CASE).

23.3.5 Statusanzeige für Haltepunkte

Einführung

Die Anzeige für Haltepunkte wird sowohl im geöffneten Programmiereditor in der linken Spalte als auch in der Taskcard "Testen" unter "Haltepunkte" identisch angezeigt.

Von Ihnen vorgenommene Änderungen werden sowohl im Programmiereditor als auch in der Taskcard "Testen" unmittelbar aktualisiert.

Haltepunkte werden abhängig von ihrem jeweiligen Status mit unterschiedlichen Symbolen visualisiert.

Statusanzeigen für Haltepunkte

Die nachfolgende Tabelle zeigt, wie Haltepunkte abhängig von ihrem Status im Programmcode im Programmiereditor dargestellt werden:

Statusanzeige für Haltepunkte in einer S7-300/400 und S7-1500 CPU:

Symbol	Bedeutung
	Haltepunkt ist inaktiv
	Haltepunkt ist inaktiv mit Bedingung, (z.B. Aufrufumgebung für den Haltepunkt ist definiert)
	Haltepunkt ist inaktiv, Programmausführung steht an dieser Stelle
	Haltepunkt ist inaktiv, hat eine Bedingung (z.B. Aufrufumgebung für den Haltepunkt ist definiert) und die Programmausführung steht an dieser Stelle.
	Haltpunkt ist aktiv
	Haltepunkt ist aktiv mit Bedingung, (z.B. Aufrufumgebung für den Haltepunkt ist definiert)
	Haltepunkt ist aktiv und die Programmausführung steht an dieser Stelle.
	Haltepunkt ist aktiv, hat eine Bedingung (z.B. Aufrufumgebung für den Haltepunkt ist definiert) und die Programmausführung steht an dieser Stelle.
	Programmausführung steht an dieser Stelle

In der CPU-Familie S7-1500 ab Firmwarestand V2.5 gibt es zusätzlich noch folgende Symbole:

Symbol	Bedeutung
	Der Haltepunkt ist inaktiv und kann nicht aktiviert werden. Mögliche Ursachen: • Dieser Haltepunkt kann nach dem Kompilieren aufgrund von Code-Optimierungen nicht mehr erreicht werden. • Der Haltepunkt kann nicht aktiviert werden, weil eine Aufrufumgebung für das Beobachten des Bausteins definiert wurde. Siehe hierzu: Regeln zum Testen mit Haltepunkten (Seite 8063)
	Der Haltepunkt ist inaktiv und kann nicht aktiviert werden und die Programmausführung steht an dieser Stelle. Mögliche Ursachen: • Der Haltepunkt kann nicht aktiviert werden, weil eine Aufrufumgebung für das Beobachten des Bausteins definiert wurde. Siehe hierzu: Regeln zum Testen mit Haltepunkten (Seite 8063)

Siehe auch

Regeln zum Testen mit Haltepunkten (Seite 8063)

23.3.6 Aufrufumgebung für Haltepunkte einstellen

Einführung

Durch das Einstellen der Aufrufumgebung können Sie festlegen, wann die Programmausführung an den Haltepunkten unterbrochen wird.

Eine Aufrufumgebung können Sie jedoch nur für S7-400 CPUs einstellen. Wenn Sie eine andere CPU verwenden, ist es **nicht** möglich, eine Aufrufumgebung für einen Haltepunkt einzustellen.

Für das Einstellen einer Aufrufumgebung für das Beobachten von Bausteinen gelten folgende Regeln:

Gültig für S7-1500er CPUs ab Firmwarestand V2.5:

- Sobald ein Haltepunkt in einem Baustein aktiviert ist, ist es **nicht** möglich, eine Aufrufumgebung für das Beobachten einzustellen.
- Umgekehrt kann ein Haltepunkt in einem Baustein **nicht** aktiviert werden, falls bereits eine Aufrufumgebung eingestellt ist.
- Falls Sie versuchen, beide Funktionalitäten anzuwenden, erhalten Sie jeweils eine Fehlermeldung mit Hinweisen für die weitere Vorgehensweise.

Voraussetzung

- Im Programm sind gültige Haltepunkte vorhanden.
- Die CPU unterstützt die Verwendung von Haltepunkten und das Einstellen einer Aufrufumgebung.

Vorgehen

Um die Aufrufumgebung für einen Haltepunkt einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
2. Klicken Sie in der Palette "Haltepunkte" hinter dem Haltepunkt, für den Sie die Aufrufumgebung einstellen möchten, auf die Schaltfläche "...".
   Der Dialog "Aufrufumgebung eines Bausteins" wird geöffnet.
3. Wählen Sie die gewünschte Bedingung aus.
   Siehe auch: Grundlagen zur Aufrufumgebung (Seite 8029)
4. Bestätigen Sie Ihre Auswahl mit "OK".

Ergebnis

Für den Haltepunkt wird die Aufrufumgebung festgelegt und das Symbol für den Haltepunkt ändert sich von einem einfachen grauen Kreis in einen grauen Kreis mit senkrechten Strichen vor und nach dem Kreis.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Haltepunkte setzen (Seite 8067)

Haltepunkte aktivieren (Seite 8068)

Testen mit Haltepunkten im Einzelschrittmodus durchführen (Seite 8076)

Haltepunkte löschen (Seite 8074)

23.3.7 Haltepunkte setzen

Einführung

Das Setzen von Haltepunkten im Programmcode ist ab dem TIA Portal V15 für AWL und SCL mit einer CPU S7-300, S7-400 und S7-1500 ab Firmware-Stand V2.5 möglich.

Um die Funktionalität von Haltepunkten zu nutzen, müssen ältere Projekte mit der neuen Version des TIA Portals erneut übersetzt werden.

Sie haben verschiedene Möglichkeiten, Haltepunkte zu setzen:

- Über das Menü "Online" > "Haltepunkte" > "Haltepunkt setzen".
  Dieser Menüeintrag ist nur dann sichtbar, wenn der Cursor in einem geöffneten Baustein nicht in der Bausteinschnittstelle platziert ist.
- Über das Kontextmenü "Haltepunkt setzen".
- Über die Symbole unter der Schaltfläche "Haltepunkte" in der Taskcard "Testen".
- Über den Tastaturbefehl "STRG+Shift+F9"
- Durch Klicken in die graue Spalte vor der Zeilennummer im geöffneten Baustein.

Neu gesetzte Haltepunkte sind per Voreinstellung zunächst noch deaktiviert und müssen vor der Ausführung zuerst aktiviert werden.

Siehe hierzu auch:

Security-Hinweise zum Testen mit Haltepunkten (Seite 8060)

Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)

Regeln zum Testen mit Haltepunkten (Seite 8063)

Voraussetzung

Der Baustein wurde in AWL oder SCL erstellt.

Sie arbeiten auf eine CPU S7-300, S7-400 oder S7-1500 ab Firmware-Stand V2.5, die Haltepunkte unterstützt.

Vorgehen

Um einen Haltepunkt in den Programmcode zu setzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, in den Sie einen Haltepunkt setzen möchten.
2. Klicken Sie mit der rechten Maustaste in der Zeile, in die Sie den Haltepunkt setzen möchten, in die graue Spalte vor der Zeilennummer.
3. Wählen Sie im Kontextmenü den Befehl "Haltepunkt setzen".

Oder:

1. Öffnen Sie den Baustein, in den Sie einen Haltepunkt setzen möchten.
2. Klicken Sie in der Zeile, in die Sie den Haltepunkt setzen möchten, in die graue Spalte vor der Zeilennummer.

Ergebnis

In den Programmcode wird ein Haltepunkt gesetzt. Der Haltepunkt wird in der Task Card "Testen" in der Palette "Haltepunkte" angezeigt, zusätzlich wird der Haltepunkt vor der entsprechenden Zeile im Programmcode als grauer Kreis dargestellt. Damit die Programmausführung an dem neu gesetzten Haltepunkt unterbrochen wird, muss der Haltepunkt zuerst aktiviert werden.

Hinweis

Haltepunkte während des Beobachtens setzen und aktiivieren

Ab dem TIA Portal V15 können Sie Haltepunkte auch setzen, während das Beobachten von Bausteinen aktiv ist.

Mit einer CPU S7-1500 ab dem Firmwarestand V2.5 können Sie Haltepunkte auch aktivieren, während das Beobachten von Bausteinen aktiv ist.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Aufrufumgebung für Haltepunkte einstellen (Seite 8065)

Haltepunkte aktivieren (Seite 8068)

Testen mit Haltepunkten im Einzelschrittmodus durchführen (Seite 8076)

Haltepunkte löschen (Seite 8074)

Security-Hinweise zum Testen mit Haltepunkten (Seite 8060)

Sicherheitshinweise beim Testen mit Haltepunkten (Seite 8060)

Regeln zum Testen mit Haltepunkten (Seite 8063)

23.3.8 Haltepunkte aktivieren

Einführung

Um Haltepunkte beim Testen zu verwenden, müssen Sie die gesetzten Haltepunkte aktivieren.

Durch die Aktivierung werden die Haltepunkte in der CPU installiert und aktiv geschaltet.

Sie können die Haltepunkte entweder einzeln oder alle zusammen aktivieren.

Sie haben die Möglichkeit, Haltepunkte über die Taskcard "Testen" oder alternativ über das Menü Online und über das Kontextmenü zu bearbeiten.

Gültig für S7-1500er CPUs ab Firmwarestand V2.5:

- Sobald ein Haltepunkt in einem Baustein aktiviert ist, ist es **nicht** möglich, eine Aufrufumgebung für das Beobachten einzustellen.
- Umgekehrt kann ein Haltepunkt in einem Baustein **nicht** aktiviert werden, falls bereits eine Aufrufumgebung eingestellt ist.
- Falls Sie versuchen, beide Funktionalitäten anzuwenden, erhalten Sie jeweils eine Fehlermeldung mit Hinweisen für die weitere Vorgehensweise.

Hinweis

Die Anzahl der Haltepunkte, die in einer CPU aktiviert werden können, ist CPU-abhängig.

Wenn Sie die maximale Anzahl erreicht haben, können Sie keine weiteren Haltepunkte aktivieren.

Da die Einzelschrittfunktionen "Überspringen", "Hineinspringen" und "Ausführen bis Markierung" ebenfalls Haltepunkte benutzen, können diese beim Erreichen der maximalen Anzahl von Haltpunkten ebenfalls nicht ausgeführt werden. Um weitere Haltepunkte zu aktivieren oder um diese Einzelschrittfunktionen verwenden zu können, müssen Sie zuerst bereits aktivierte Haltepunkte deaktivieren.

Hinweis

Haltepunkte während des Beobachtens aktivieren

Ab dem TIA Portal V15 können Haltepunkte mit einer CPU S7-1500 ab Firmwarestand V2.5 auch aktiviert werden, während das Beobachten von Bausteinen aktiv ist.

Voraussetzung

- Es wurden gültige Haltepunkte gesetzt.
- Es besteht eine Online-Verbindung zur CPU und diese CPU unterstützt die Verwendung von Haltepunkten.
- Die maximal aktivierbare Anzahl von Haltepunkten für die CPU ist nicht erreicht.
- Der Offline- und der Online-Baustein auf der CPU sind identisch.

Einzelne Haltepunkte aktivieren

Um einzelne Haltepunkte zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, den Sie aktivieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Haltepunkt aktivieren".

ACHTUNG
Haltepunkte aktivieren auf einer CPU S7-1500 ab Firmwarestand V2.5 Sobald Sie auf einer CPU S7-1500 ab V2.5 einen Haltepunkt aktivieren, erhalten Sie die Information, dass die Ausgangswerte für die angeschlossen Ausgangsbaugruppen in ihrem jeweiligen Zustand gehalten und (im Gegensatz zur SIMATIC S7-300/S7-400) weder zurückgesetzt noch auf die projektierten Ersatzwerte gesetzt werden. Darüber hinaus kann die SIMATIC S7-1500 in diesem Zustand nicht mehr auf Eingangswerte von Eingangsbaugruppen reagieren. Diese Information wird Ihnen in jeder Online-Sitzung nur einmal angezeigt.

Ergebnis

Der selektierte Haltepunkt wird in der CPU installiert und aktiviert und die Programmausführung wird am Haltepunkt unterbrochen, wenn die entsprechende Programmstelle erreicht wird. Die Darstellung des Haltepunktes ändert sich von einem grauen Kreis zu einem roten Kreis.

Alle Haltepunkte aktivieren

Um alle Haltepunkte zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Achten Sie darauf, dass kein Haltepunkt selektiert ist.
3. Klicken Sie in der Funktionsleiste der Palette "Haltepunkte" auf "Alle Haltepunkte aktivieren".

ACHTUNG
Haltepunkte aktivieren auf einer CPU S7-1500 ab Firmwarestand V2.5 Sobald Sie auf einer CPU S7-1500 ab Firmwarestand V2.5 einen Haltepunkt aktivieren, erhalten Sie die Information, dass die Ausgangswerte für die angeschlossen Ausgangsbaugruppen in ihrem jeweiligen Zustand gehalten und (im Gegensatz zur SIMATIC S7-300/S7-400) weder zurückgesetzt noch auf die projektierten Ersatzwerte gesetzt werden. Darüber hinaus kann die SIMATIC S7-1500 in diesem Zustand nicht mehr auf Eingangswerte von Eingangsbaugruppen reagieren. Diese Information wird Ihnen in jeder Online-Sitzung nur einmal angezeigt.

Ergebnis

Abhängig von der Kapazität der CPU werden alle Haltepunkte in der CPU installiert und aktiviert und die Programmausführung wird an allen Haltepunkten unterbrochen, die während der Programmausführung erreicht werden.

- Falls Sie mehr Haltepunkte gesetzt haben als die CPU zulässt, wird nur die zulässige Anzahl der Haltepunkte aktiviert.
- Die Auswahl, welche Haltepunkte aktiviert werden, kann sich zur Laufzeit verändern.
- Ein wiederholtes Ausführen der Funktion kann zu unterschiedlichen Ergebnissen führen.

Die Darstellung der aktivierten Haltepunkte ändert sich von grauen Kreisen zu roten Kreisen.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Haltepunkte setzen (Seite 8067)

Aufrufumgebung für Haltepunkte einstellen (Seite 8065)

Testen mit Haltepunkten im Einzelschrittmodus durchführen (Seite 8076)

Haltepunkte löschen (Seite 8074)

23.3.9 Haltepunkte deaktivieren

Einführung

Sie können aktivierte Haltepunkte entweder einzeln oder alle zusammen deaktivieren. Nach dem Deaktivieren von Haltepunkten werden diese bei der Programmausführung nicht mehr beachtet.

Sie haben die Möglichkeit, Haltepunkte über die Taskcard "Testen" oder alternativ über das Menü Online und über das Kontextmenü zu bearbeiten.

Hinweis

Wissensertes zu Haltepunkten

- Wenn Sie die Online-Verbindung zur CPU trennen, werden die Haltepunkte deaktiviert.
- Die Haltepunkte werden nicht zusammen mit dem Projekt gespeichert.
- Wenn Sie das Projekt schließen, gehen alle Haltepunkte verloren.

Hinweis

Haltepunkte während des Beobachtens deaktivieren

Ab dem TIA Portal V15 können Haltepunkte auch deaktiviert werden, während das Beobachten von Bausteinen aktiv ist.

Voraussetzung

- Es wurden gültige Haltepunkte gesetzt und aktiviert.
- Es besteht eine Online-Verbindung zur CPU und diese CPU unterstützt die Verwendung von Haltepunkten.

Einzelne Haltepunkte deaktivieren

Um einzelne Haltepunkte zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, den Sie deaktivieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Haltepunkt deaktivieren".

Ergebnis

Der selektierte Haltepunkt wird in der CPU deinstalliert und deaktiviert und die Programmausführung wird am Haltepunkt nicht mehr unterbrochen. Die Darstellung des Haltepunktes ändert sich von einem roten Kreis zu einem grauen Kreis.

Alle Haltepunkte deaktivieren

Um alle Haltepunkte zu deaktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Achten Sie darauf, dass kein Haltepunkt selektiert ist.
3. Klicken Sie in der Funktionsleiste der Palette "Haltepunkte" auf "Alle Haltepunkte deaktivieren".

Ergebnis

Alle Haltepunkte werden in der CPU deinstalliert und deaktiviert und die Programmausführung wird an keinem Haltepunkt mehr unterbrochen. Die Darstellung der Haltepunkte ändert sich von roten Kreisen zu grauen Kreisen.

23.3.10 Zu den Haltepunkten navigieren

Einführung

Um zu den Haltepunkten in Ihrem Programm zu navigieren, können Sie entweder von Haltepunkt zu Haltepunkt oder gezielt zu der Programmstelle mit einem bestimmten Haltepunkt springen.

Von Haltepunkt zu Haltepunkt navigieren

Um in Ihrem Programm von Haltepunkt zu Haltepunkt zu navigieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen Baustein und selektieren Sie den Programmiereditor.
2. Wählen Sie im Menü "Online > Haltepunkte" den Befehl "Gehe zum nächsten Haltepunkt", um zum nächsten Haltepunkt nach der aktuellen Cursorposition zu springen. Nach dem letzten Haltepunkt wird zum ersten Haltepunkt gesprungen.
3. Wählen Sie im Menü "Online > Haltepunkte" den Befehl "Gehe zum vorangehenden Haltepunkt ", um zum vorhergehenden Haltepunkt vor der aktuellen Cursorposition zu springen. Nach dem ersten Haltepunkt wird zum letzten Haltepunkt gesprungen.

Zu einem bestimmten Haltepunkt navigieren

Um in Ihrem Programm zu einem bestimmten Haltepunkt zu navigieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie doppelt auf den Haltepunkt, zu dem Sie navigieren möchten.
   Der Baustein mit dem Haltepunkt wird im Programmiereditor angezeigt und die Zeile mit dem Haltepunkt wird hervorgehoben.

Oder:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, zu dem Sie navigieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Gehe zu Haltepunkt".
   Der Baustein mit dem Haltepunkt wird im Programmiereditor angezeigt und die Zeile mit dem Haltepunkt wird hervorgehoben.

23.3.11 Haltepunkte löschen

Einführung

Sie können Haltepunkte entweder einzeln oder alle zusammen löschen.

Sie haben alternativ folgende Möglichkeiten, um Haltepunkte zu löschen:

- Über das Menü "Online > Haltepunkte > Haltepunkte setzen/löschen", oder "Online > Haltepunkte > Alle löschen".
- Über die entsprechenden Kontextmenü-Einträge.
- Über die entsprechenden Tastaturbefehle (STRG+Shift+F9).
- Über die Befehle in der Taskcard "Testen".

Hinweis

- Wenn Sie die Online-Verbindung zur CPU trennen, werden die Haltepunkte deaktiviert.
- Die Haltepunkte werden nicht zusammen mit dem Projekt gespeichert.
- Wenn Sie das Projekt schließen, gehen alle Haltepunkte verloren.

Hinweis

Haltepunkte während des Beobachtens löschen

Ab dem TIA Portal V15 können Haltepunkte mit einer CPU S7-1500 ab dem Firmwarestand V2.5 auch gelöscht werden, während das Beobachten von Bausteinen aktiv ist.

Einzelne Haltepunkte löschen

Um einen einzelnen Haltepunkt aus dem Programm zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Taskcard "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Selektieren Sie den Haltepunkt, den Sie löschen möchten.
3. Klicken Sie mit der rechten Maustaste auf den selektierten Haltepunkt.
4. Wählen Sie im Kontextmenü den Befehl "Haltepunkt löschen".

Oder:

1. Öffnen Sie den Baustein, aus dem Sie einen Haltepunkt löschen möchten.
2. Klicken Sie in der Zeile mit dem Haltepunkt, den Sie löschen möchten, in die graue Spalte vor der Zeilennummer.

Ergebnis

Der ausgewählte Haltepunkt wird gelöscht.

Alle Haltepunkte löschen

Um alle Haltepunkte zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie in der Funktionsleiste der Palette "Haltepunkte" auf "Haltepunkte löschen".

Ergebnis

Alle vorhandenen Haltepunkte werden gelöscht.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Haltepunkte setzen (Seite 8067)

Aufrufumgebung für Haltepunkte einstellen (Seite 8065)

Haltepunkte aktivieren (Seite 8068)

Testen mit Haltepunkten im Einzelschrittmodus durchführen (Seite 8076)

23.3.12 Testen mit Haltepunkten im Einzelschrittmodus durchführen

Einführung

Wenn die Programmausführung an einem Haltepunkt unterbrochen wurde, können Sie das Programm schrittweise weiterlaufen lassen.

Voraussetzung

- Im Programmcode sind Haltepunkte gesetzt und aktiviert.
- Die maximal aktivierbare Anzahl von Haltepunkten für die CPU ist nicht erreicht.
- Ein Baustein ist geöffnet.

Vorgehen

Um die Programmausführung im Einzelschrittmodus auszuführen, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zur CPU her.
2. Aktivieren Sie Haltepunkte.
   Der Programmcode wird bis zum aktivierten Haltepunkt ausgeführt und dann unterbrochen, d. h. die CPU geht in den Betriebszustand "HALT". Der rote Kreis für den Haltepunkt wird mit einem gelben Pfeil überlagert.
3. Um die Programmausführung schrittweise weiter auszuführen, wählen Sie in der Funktionsleiste einen der folgenden Befehle:
   - Ausführen
     Die Programmausführung wird fortgesetzt. Falls weitere Haltepunkte vorhanden sind, wird die Programmausführung am nächsten Haltepunkt erneut unterbrochen.
   - Ausführung bis Markierung
     Das Programm wird bis zu der Position fortgesetzt, an der sich die Einfügemarke befindet. Falls sich ein weiterer Haltepunkt vor der Einfügemarke befindet, wird die Programmausführung zunächst an diesem Haltepunkt unterbrochen.
   - Überspringen (nur für S7-300/400)
     Die nächste Anweisung nach dem Haltepunkt wird ausgeführt. Wenn es sich bei der Anweisung um einen Bausteinaufruf handelt, wird der Aufruf bearbeitet und es wird zur nächsten Anweisung nach dem Bausteinaufruf gesprungen.
   - Hineinspringen (nur für S7-300/400)
     Die nächste Anweisung nach dem Haltepunkt wird ausgeführt. Wenn es sich bei der Anweisung um einen Bausteinaufruf handelt, wird in den aufgerufenen Baustein gesprungen. Dort können Sie im Einzelschritt weitertesten oder weitere Haltepunkte setzen. Am Ende des Bausteins erfolgt ein Rücksprung zur nächsten Anweisung nach dem Bausteinaufruf.
   - Zurückspringen (nur für S7-300/400)
     Wenn ein Bausteinaufruf bearbeitet wird und die Programmausführung im aufgerufenen Baustein an einem Haltepunkt unterbrochen wurde, können Sie mit "Zurückspringen" in den aufrufenden Baustein zurückspringen. Dabei werden die restlichen Anweisungen des aufgerufenen Bausteins ausgeführt und die Programmausführung wird im aufrufenden Baustein nach dem Bausteinaufruf erneut angehalten.

Siehe auch

Einführung zum Testen mit Haltepunkten (Seite 8058)

Haltepunkte setzen (Seite 8067)

Aufrufumgebung für Haltepunkte einstellen (Seite 8065)

Haltepunkte aktivieren (Seite 8068)

Haltepunkte löschen (Seite 8074)

Einführung zum Testen mit Programmstatus (Seite 8027)

23.3.13 Haltepunkte "Ausführen bis Markierung"

Einführung

Die Funktion "Ausführen bis Markierung" setzt an der Einfügemarke einen temporären Haltepunkt und löscht diesen wieder, nachdem er erreicht wurde.

Sie können diese Funktion sowohl über das entsprechende Symbol in der Taskcard "Testen" als auch über das Kontextmenü und auch über die Tastaturkombination "CTL + Shift + 8" ausführen.

Hinweis

Die Anzahl der Haltepunkte, die in einer CPU aktiviert werden können, ist CPU-abhängig.

Wenn Sie die maximale Anzahl erreicht haben, können Sie keine weiteren Haltepunkte aktivieren.

Da die Einzelschrittfunktionen "Überspringen", "Hineinspringen" und "Ausführen bis Markierung" ebenfalls Haltepunkte benutzen, können diese beim Erreichen der maximalen Anzahl von Haltpunkten ebenfalls nicht ausgeführt werden. Um weitere Haltepunkte zu aktivieren oder um diese Einzelschrittfunktionen verwenden zu können, müssen Sie zuerst bereits aktivierte Haltepunkte deaktivieren.

Voraussetzung

- Es besteht eine Online-Verbindung zur CPU und diese CPU unterstützt die Verwendung von Haltepunkten.
- Es wurden gültige Haltepunkte gesetzt und aktiviert.
- Die maximal aktivierbare Anzahl von Haltepunkten für die CPU ist nicht erreicht.
- Der Offline- und der Online-Baustein auf der CPU sind identisch.

Ausführen bis Markierung

Um die Funktion "Ausführen bis Markierung" zu aktivieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Testen".
   In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
2. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, den Sie aktivieren möchten.
3. Wählen Sie im Kontextmenü den Befehl "Haltepunkt aktivieren".
4. Warten Sie bis der Haltepunkt erreicht wird und die CPU am definierten Haltepunkt in den Status "HALT" geht.

5. Setzen Sie die Einfügemarke im Programcode an eine andere Stelle, die für das Einfügen eines weiteren Haltepunktes gültig ist.
6. Starten Sie die Funktion "Ausführen bis Markierung" über das Symbol in der Taskcard "Testen" oder über das Kontextmenü.

ACHTUNG
Haltepunkte aktivieren auf einer CPU S7-1500 ab Firmwarestand V2.5 Sobald Sie auf einer CPU S7-1500 ab V2.5 einen Haltepunkt aktivieren, erhalten Sie die Information, dass die Ausgangswerte für die angeschlossen Ausgangsbaugruppen in ihrem jeweiligen Zustand gehalten und (im Gegensatz zur SIMATIC S7-300/S7-400) weder zurückgesetzt noch auf die projektierten Ersatzwerte gesetzt werden. Darüber hinaus kann die SIMATIC S7-1500 in diesem Zustand nicht mehr auf Eingangswerte von Eingangsbaugruppen reagieren. Diese Information wird Ihnen in jeder Online-Sitzung nur einmal angezeigt.

Ergebnis

Die Funktion wird ausgeführt und ein temporärer Haltepunkt an der Einfügemarke gesetzt. Die CPU geht an der durch die Einfügemarke gekennzeichneten Stelle in "HALT".

Die Stelle, an der der Haltepunkt gesetzt wurde, wird durch das kleine gelbe Dreieck als "erreicht" angezeigt, der temporäre Haltepunkt wurde gelöscht.

23.3.14 Variablen bei erreichtem Haltepunkt beobachten

Einführung

Für eine CPU S7-1500 ab dem Firmwarestand V2.5 können Sie Variablen auch im Zusammenspiel mit Haltepunkten beobachten, z. B., wenn die CPU beim Erreichen eines Haltepunktes im Betriebszustand "HALT" ist.

Diese Funktionalität steht nur für globale Operanden (Eingänge, Ausgänge, Merker, SIMATIC-Zeiten, SIMATIC-Zähler und DBs) in elementaren Datentypen zur Verfügung.

Wenn Sie den Mauszeiger über der Variablen positionieren, wird der Wert des Operanden in einem Tooltipp angezeigt.

Voraussetzung

- Es besteht eine Online-Verbindung zur CPU.
- Es wurden gültige Haltepunkte gesetzt und aktiviert.
- Ein Haltepunkt wird sowohl in der Taskcard "Testen" als auch im Programmiereditor als "erreicht" gekennzeichnet.
- Die CPU ist im Betriebszustand "HALT", der in der Taskcard "Testen" sowohl beim Bedienpanel als auch bei den Haltepunkten angezeigt wird.

Variablen bei erreichtem Haltepunkt beobachten (S7-1500)

Um Variablen im "HALT" zu beobachten, gehen Sie vor wie folgt:

1. Öffnen Sie im Programmiereditor einen AWL- oder SCL-Baustein, der gültige Haltepunkte enthält.
2. Öffnen Sie die Taskcard "Testen". In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
3. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, den Sie aktivieren möchten.
4. Wählen Sie im Kontextmenü den Befehl "Haltepunkt aktivieren".
5. Warten Sie, bis der Haltepunkt bei der Programmausführung erreicht wird und die CPU in "HALT" geht.
6. Positionieren Sie den Mauszeiger im Programmcode über einer globalen Variable.

Ergebnis

Der Beobachtungswert zu dieser globalen Variable wird in einem gelb hinterlegten Tooltipp angezeigt.

Regeln zum Beobachten von Variablen bei aktiven Haltepunkten

Um Variablen beobachten zu können, muss diese Funktionalität über die Schaltfläche "Beobachten ein/aus" in der Funktionsleiste des Programmiereditors eingeschaltet sein.

Abhängig vom Zeitpunkt, wann Sie das Beobachten von Variablen während der Programmausführung einschalten, erhalten Sie unterschiedliche Ergebnisse in der Anzeige des Programmstatus:

- Wird ein Haltepunkt erreicht, bevor Sie das Beobachten eingeschaltet haben, werden beim Programmstatus keine Werte angezeigt.
- Wird ein Haltepunkt während des ersten Durchlaufs erreicht, nachdem das Beobachten eingeschaltet wurde, werden beim Programmstatus nur die Werte bis zum erreichten Haltepunkt angezeigt.
- Wird ein Haltepunkt erst erreicht, nachdem der Baustein schon mehrere Male aufgerufen wurde, werden die Beobachtungswerte beim Programmstatus wie folgt angezeigt:
  - Werte aus vorherigen Zyklen werden in grauer Farbe dargestellt,
  - Werte aus dem aktuellen Zyklus, in dem die CPU den Haltepunkt erreicht hat, werden in schwarzer Farbe dargestellt.

Hinweis

- Die Haltepunkte werden nicht zusammen mit dem Projekt gespeichert. Wenn Sie das Projekt schließen, gehen alle Haltepunkte verloren.
- Wenn Sie die Online-Verbindung zur CPU trennen, werden die Haltepunkte deaktiviert.

Siehe auch

Variablen während des Beobachtens steuern (Seite 8039)

23.3.15 Variablen bei erreichtem Haltepunkt steuern

Einführung

Für eine CPU S7-1500 ab dem Firmwarestand V2.5 können Sie Variablen auch im Zusammenspiel mit Haltepunkten steuern, z. B., wenn die CPU beim Erreichen eines Haltepunktes im Betriebszustand "HALT" ist.

Diese Funktionalität steht nur für globale Operanden (Eingänge, Ausgänge, Merker, SIMATIC-Zeiten, SIMATIC-Zähler und DBs) in elementaren Datentypen zur Verfügung.

Folgende Optionen für das Steuern stehen Ihnen im Kontextmenü zur Verfügung:

- Steuern auf 0
- Steuern auf 1
- Operand steuern: Tragen Sie im nachfolgenden Dialog den gewünschten Steuerwert ein.

Das Steuern wird für den selektierten Operanden sofort und einmalig durchgeführt.

Mit einer CPU S7-1500 ab dem Firmwarestand V2.5 können Sie globale Operanden während des Beobachtens auch direkt im Code steuern.

Mit einer CPU S7-300/400 können Sie globale Operanden in einem SCL-Baustein ebenfalls während des Beobachtens direkt im Code steuern.

Wenn die CPU einen Haltepunkt erreicht hat und im Betriebszustand "HALT" ist, haben Sie auch die Möglichkeit, globale Operanden über die Beobachtungstabelle zu steuern.

Voraussetzung

- Es besteht eine Online-Verbindung zur CPU.
- Es wurden gültige Haltepunkte gesetzt und aktiviert.
- Ein Haltepunkt wird sowohl in der Taskcard "Testen" als auch im Programmiereditor als "erreicht" gekennzeichnet.
- Die CPU ist im Betriebszustand "HALT", der in der Taskcard "Testen" sowohl beim Bedienpanel als auch bei den Haltepunkten angezeigt wird.

Variablen bei erreichtem Haltepunkt steuern (S7-1500)

Um Variablen im "HALT" zu steuern, gehen Sie vor wie folgt:

1. Öffnen Sie im Programmiereditor einen AWL- oder SCL-Baustein, der gültige Haltepunkte enthält.
2. Starten Sie das Beobachten über die Schaltfläche in der Funktionsleiste "Beobachten ein/aus".

3. Öffnen Sie die Taskcard "Testen". In der Palette "Haltepunkte" sind alle Haltepunkte aufgelistet, die für die aktuelle CPU gesetzt sind.
4. Klicken Sie mit der rechten Maustaste auf den Haltepunkt, den Sie aktivieren möchten.
5. Wählen Sie im Kontextmenü den Befehl "Haltepunkt aktivieren".
6. Warten Sie, bis der Haltepunkt bei der Programmausführung erreicht wird und die CPU in "HALT" geht.
7. Steuern Sie den Operanden auf den gewünschten Wert, indem Sie im Kontextmenü auf den entsprechenden Befehl klicken.
   - "Steuern > Steuern auf 1"
     Steuert Variablen vom Datentyp "BOOL" auf den Wert "TRUE".
   - "Steuern > Steuern auf 0"
     Steuert Variablen vom Datentyp "BOOL" auf den Wert "FALSE"
   - "Steuern > Operand steuern"
     Wenn Sie "Operand steuern" wählen, wird der Dialog "Operand steuern" geöffnet. Tragen Sie im folgenden Dialog in das Feld "Steuerwert" den gewünschten Wert ein und bestätigen Sie Ihre Eingabe mit "OK".
8. Alternativ können Sie Sie auf den Onlinewert doppelklicken (nicht beim Datentyp BOOL), den Sie steuern möchten.
9. Tragen Sie im folgenden Dialog in das Feld "Steuerwert" den gewünschten Wert ein und bestätigen Sie Ihre Eingabe mit "OK".
   Oder bestätigen Sie im folgenden Dialog die Abfrage, ob der boolesche Online-Wert umgeschaltet werden soll.

- **Beispiel:**
  - Hat die boolesche Variable den Wert "0", so wird sie auf den Wert "1" umgeschaltet, wenn Sie die Abfrage mit "Ja" bestätigen.
  - Hat die boolesche Variable den Wert "1", so wird sie auf den Wert "0" umgeschaltet, wenn Sie die Abfrage mit "Ja" bestätigen.
  - Wenn Sie auf "Nein" klicken oder den Dialog über "Schließen" verlassen, bleibt der bestehende Wert erhalten.

  Hinweis

  Steuern von Variablen im Betriebszustand "HALT"

  Beachten Sie, dass die angezeigten Variablen nicht mehr aktualisiert werden, wenn sich die CPU im Betriebszustand "HALT" befindet.

  Es werden immer die im "HALT" angezeigten Variablen gesteuert.

Ergebnis

Der selektierte Operand wird auf den gewünschten Wert gesteuert.

Siehe auch

Variablen während des Beobachtens steuern (Seite 8039)

23.3.16 Programmausführung fortsetzen

Einführung

Wenn Sie gültige Haltepunkte gesetzt und diese aktiviert haben, können Sie das Programm beim Testen bis zu einem beliebigen Haltepunkt ausführen lassen. Sobald ein gültiger Haltepunkt erreicht wurde, befindet sich die CPU im Zustand "HALT".

Die Programmbearbeitung wird bei einer CPU S7-1500 erst dann fortgesetzt, wenn Sie die CPU wieder in den Betriebszustand "RUN" setzen.

Sie können die Programmausführung mit oder ohne Haltepunkte fortsetzen.

Voraussetzung

- Es besteht eine Online-Verbindung zur CPU.
- Es wurden gültige Haltepunkte gesetzt und aktiviert.
- Ein Haltepunkt wird sowohl in der Taskcard "Testen" als auch im Programmiereditor als "erreicht" gekennzeichnet.
- Die CPU ist im Betriebszustand "HALT", der in der Taskcard "Testen" sowohl beim Bedienpanel als auch bei den Haltepunkten angezeigt wird.

Programmausführung fortsetzen

Sie haben folgende Möglichkeiten, die Programmausführung nach dem Erreichen eines Haltepunktes wieder fortzusetzen:

- Klicken Sie in der Taskcard "Testen" unter "Haltepunkte" auf die Schaltfläche "RUN".
- Wählen Sie die Option "RUN" über das Kontextmenü im geöffneten Programmiereditor.
- Verwenden Sie im geöffneten Programmiereditor die Tastaturbefehle "Ctrl+Shift+F7".

Ergebnis

Die CPU wechselt von "HALT" in "RUN" und die normale Programmbearbeitung wird bis zum nächste Haltepunkt fortgesetzt.

Falls es nicht möglich ist, die Programmausführung fortzusetzen, erhalten Sie eine entsprechende Meldung.

Sind keine weiteren Haltepunkte aktiviert, wird die Programmausführung ohne Halteunkte fortgesetzt.

23.4 Testen mit der Beobachtungstabelle

23.4.1 Einführung zum Testen mit der Beobachtungstabelle

Übersicht

Innerhalb der Beobachtungstabelle stehen Ihnen folgende Funktionen zur Verfügung:

- **Beobachten von Variablen**
  Damit lassen sich die aktuellen Werte einzelner Variablen eines Anwenderprogramms bzw. einer CPU am PG/PC anzeigen.
- **Steuern von Variablen**
  Mit dieser Funktion können Sie einzelnen Variablen eines Anwenderprogramms bzw. einer CPU feste Werte zuweisen. Das Steuern ist auch beim Testen mit Programmstatus (Seite 8027) möglich.
- **"PA freischalten" und "Steuern sofort"**
  Diese beiden Funktionen geben Ihnen die Möglichkeit, einzelnen Peripherieausgängen einer CPU im Betriebszustand STOP feste Werte zuzuweisen. Sie können damit auch Ihre Verdrahtung überprüfen.

Beobachten und Steuern von Variablen

Folgende Variablen können beobachtet bzw. gesteuert werden:

- Eingänge, Ausgänge und Merker
- Inhalte von Datenbausteinen
- Inhalte von UDTs
- Peripherie

Team Engineering in der Beobachtungstabelle

Ab dem TIA Portal V13 SP1 können mit einer CPU S7-1500 mit einem Firmwarestand >= 1.7 im Rahmen von Team Engineering mehrere Engineering Systeme gleichzeitig online auf die CPU zugreifen und z. B. gleichzeitig Variablen beobachten und steuern und Bausteine auf die CPU laden. Beachten Sie hierzu unbedingt die für das Team Engineering gültigen Voraussetzungen und Regeln, die Sie im Informationssystem unter "Team Engineering einsetzen" im Kapitel "Projekte gemeinsam in Betrieb nehmen" finden.

Einsatzmöglichkeit

Die Beobachtungstabelle bietet den Vorteil, unterschiedliche Testumgebungen abspeichern zu können. So können Tests während einer Inbetriebsetzung oder zu Service- und Wartungszwecken reproduziert werden.

Siehe auch

Aufbau der Beobachtungstabelle (Seite 8085)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 8086)

Symbole in der Beobachtungstabelle (Seite 8087)

Beobachtungstabellen erstellen und bearbeiten (Seite 8089)

23.4.2 Aufbau der Beobachtungstabelle

Einführung

Eine Beobachtungstabelle enthält die von Ihnen definierten und ausgewählten CPU-weit gültigen Variablen. Für jede im Projekt angelegte CPU wird automatisch ein Ordner "Beobachtungs- und Forcetabellen" angelegt. Über den Befehl "Neue Beobachtungstabelle hinzufügen" legen Sie eine neue Beobachtungstabelle in diesem Ordner an.

Aufbau der Beobachtungstabelle

Die in der Beobachtungstabelle angezeigten Spalten sind abhängig davon, ob Sie im Basismodus oder im Erweiterten Modus arbeiten.

Im Erweiterten Modus werden folgende Spalten zusätzlich angezeigt:

- Beobachten mit Trigger
- Steuern mit Trigger

Außerdem können die Namen der Spalten je nach Aktion dynamisch wechseln.

Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten im Basismodus und im Erweiterten Modus:

Mode	Spalte	Erläuterung
Basismodus	i	Kennzeichnungsspalte
	Name	Name der eingefügten Variablen
	Adresse	Adresse der eingefügten Variablen
	Anzeigeformat	Ausgewähltes Anzeigeformat
	Beobachtungswert	Werte der Variablen, abhängig vom ausgewählten Anzeigeformat.
	Steuerwert	Wert, mit dem die Variable gesteuert wird.
		Auswahl der zu steuernden Variablen durch Aktivieren des entsprechenden Optionskästchens.
	Kommentar	Möglichkeit, einen Kommentar zur Dokumentation der Variablen in der Beobachtungstabelle einzutragen.
	Variablen-Kommentar	Anzeige des Kommentars zur ausgewählen Variable, der in anderen Editoren eingetragen wurde. Diese Spalte ist nicht editierbar.
Im Erweiterten Modus werden zusätzlich folgende Spalten angezeigt:	Beobachten mit Trigger	Anzeige des ausgewählten Beobachtungsmodus
	Steuern mit Trigger	Anzeige des ausgewählten Steuermodus

Siehe auch

Symbole in der Beobachtungstabelle (Seite 8087)

23.4.3 Basismodus und Erweiterter Modus in der Beobachtungstabelle

Unterschied zwischen Basismodus und Erweitertem Modus in der Beobachtungstabelle

Abhängig vom eingestellten Modus, werden in der Beobachtungstabelle unterschiedliche Spalten und Spaltenüberschriften angezeigt, mit denen unterschiedliche Aktionen durchgeführt werden können.

Eine detaillierte Auflistung der jeweiligen Spalten finden Sie unter Aufbau der Beobachtungstabelle (Seite 8085).

Umschalten zwischen Basismodus und Erweitertem Modus

Um zwischen Basismodus und Erweitertem Modus umzuschalten, haben Sie folgende Möglichkeiten:

- Klicken Sie auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden". Ein erneutes Klicken auf dieses Symbol schaltet wieder in den Basismodus um.
  Oder:
- Aktivieren Sie im Menü "Online" das Optionskästchen vor dem Befehl "Erweiterter Modus". Ein Deaktivieren des Kontrollkästchens schaltet wieder in den Basismodus um.

Funktionalität im Erweiterten Modus

Folgende Funktionalität ist nur im Erweiterten Modus möglich:

- Beobachten mit Trigger
- Steuern mit Trigger
- Peripherieausgänge freischalten
- Peripherieeingänge beobachten
- Peripherieausgänge steuern

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen und Steuern von Peripherieausgängen Beachten Sie, dass das Beobachten von Peripherieeingängen und das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Siehe auch

Festlegen des Beobachtungs- und Steuermodus (Seite 8101)

23.4.4 Symbole in der Beobachtungstabelle

Bedeutung der Symbole

Die folgende Tabelle zeigt die Bedeutung der Symbole in der Beobachtungstabelle:

Symbol	Bedeutung
	Kennzeichnet eine Tabelle innerhalb der Projektnavigation als Beobachtungstabelle.
i	Zeigt Informationen in der Kennzeichnungsspalte an.

Symbol	Bedeutung
	Fügt eine Zeile vor der markierten Zeile ein.
	Fügt eine Zeile nach der markierten Zeile ein.
	Fügt eine Kommentarzeile oberhalb der markierten Zeile ein.
	Steuert die Adressen aller ausgewählten Variablen sofort und einmalig. Dieser Befehl wird einmalig und schnellstmöglich ausgeführt, ohne Bezug auf einem bestimmten Triggerpunkt im Anwenderprogramm.
	Steuert die Adressen aller ausgewählten Variablen mit Bezug auf einen bestimmten Triggerpunkt im Anwenderprogramm.
	Schaltet die Befehlsausgabesperre der Peripherieausgänge (BASP) ab. Dies ermöglicht das Steuern der Peripherieausgänge im STOP-Zustand der CPU.
	Blendet alle Spalten des Erweiterten Modus ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Spalten des Erweiterten Modus ausgeblendet.
	Blendet alle Steuerspalten ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Steuerspalten ausgeblendet.
	Startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle. Im Basismodus ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie für das Beobachten der Variablen bestimmte Triggerpunkte einstellen.
	Startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle. Dieser Befehl wird sofort ausgeführt und die Variablen werden einmalig beobachtet.
	Zeigt das Optionskästchen für die Auswahl der zu steuernden Variablen an.
	Zeigt an, dass die selektierte Variable auf den Wert "1" gesteuert wurde.
	Zeigt an, dass die selektierte Variable auf den Wert "0" gesteuert wurde.
	Zeigt an, dass die Adresse mehrfach verwendet wird.
	Zeigt an, dass ein Ersatzwert verwendet wird. Ersatzwerte sind Werte, die bei fehlerhaften Signalausgabebaugruppen an den Prozess ausgegeben werden, bzw. bei fehlerhaften Signaleingabebaugruppen im Anwenderprogramm anstelle eines Prozesswertes verwendet werden. Die Ersatzwerte sind vom Anwender vorgebbar (z. B. alten Wert beibehalten).
	Zeigt an, dass die Adresse gesperrt ist, da bereits gesteuert wird.
	Zeigt an, dass die Adresse nicht gesteuert werden kann.
	Zeigt an, dass die Adresse nicht beobachtet werden kann.
	Zeigt an, dass eine Adresse geforct wird.
	Zeigt an, dass eine Adresse teilweise geforct wird.
	Zeigt an, dass eine zugehörige Peripherie-Adresse vollständig oder teilweise geforct wird.

Symbol	Bedeutung
	Zeigt an, dass eine Adresse evtl. nicht vollständig geforct werden kann. Beispiel: Es ist zwar möglich, die Adresse AW0:P zu forcen, es ist jedoch nicht möglich, die Adresse AD0:P zu forcen, da dieser Adressbereich auf der CPU evtl. nicht vorhanden ist.
	Zeigt an, dass ein Syntaxfehler aufgetreten ist.
	Zeigt an, dass die Adresse zwar ausgewählt ist, im Moment aber z. B. noch nicht gesteuert wird.

Siehe auch

Aufbau der Beobachtungstabelle (Seite 8085)

23.4.5 Beobachtungstabellen erstellen und bearbeiten

23.4.5.1 Beobachtungstabelle erstellen

Einführung

Mit Hilfe der Beobachtungstabelle können Sie Variablen im Anwenderprogramm beobachten und steuern. Eine einmal erstellte Beobachtungstabelle können Sie speichern, duplizieren, ausdrucken und immer wieder zum Beobachten und Steuern von Variablen verwenden.

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um eine Beobachtungstabelle zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Statusleiste auf "Projektansicht".
   Die Projektansicht wird angezeigt.
2. Doppelklicken Sie in der Projektnavigation auf die entsprechende CPU, für die Sie eine Beobachtungstabelle erstellen wollen.
3. Doppelklicken Sie zunächst auf den Ordner "Beobachtungs- und Forcetabellen" und dann auf den Befehl "Neue Beobachtungstabelle hinzufügen".
   Eine neue Beobachtungstabelle wird hinzugefügt.
4. Tragen Sie in der Spalte "Name" oder in der Spalte "Adresse" den Namen oder die absolute Adresse für die Variablen ein, die Sie beobachten oder steuern wollen.
5. Falls Sie das voreingestellte Anzeigeformat ändern wollen, wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus.
6. Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder steuern wollen und tragen Sie ggf. die gewünschten Werte für das Steuern der Variablen ein.

23.4.5.2 Beobachtungstabelle öffnen

Voraussetzung

Eine Beobachtungstabelle ist angelegt.

Vorgehen

Um eine Beobachtungstabelle zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Beobachtungs- und Forcetabellen" unterhalb der gewünschten CPU.
2. Doppelklicken Sie auf die gewünschte Beobachtungstabelle im Ordner.

Ergebnis

Die ausgewählte Beobachtungstabelle wird geöffnet.

23.4.5.3 Beobachtungstabelle kopieren und einfügen

Voraussetzung

Eine Beobachtungstabelle ist angelegt.

Vorgehen

Um eine Beobachtungstabelle zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Beobachtungstabelle, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Öffnen Sie in der Projektnavigation die Ordnerstruktur für die CPU, in die Sie die kopierte Beobachtungstabelle einfügen möchten.
4. Klicken Sie mit der rechten Maustaste auf den Ordner "Beobachtungs- und Forcetabellen".
5. Wählen Sie im Kontextmenü den Befehl "Einfügen".
6. Alternativ können Sie auch den gesamten Inhalt der Beobachtungstabelle markieren und per Drag & Drop in eine andere Beobachtungstabelle ziehen.

Ergebnis

Eine Kopie der selektierten Beobachtungstabelle wird in den Ordner "Beobachtungs- und Forcetabellen" der entsprechenden CPU eingefügt.

23.4.5.4 Beobachtungstabelle speichern

Voraussetzung

Eine Beobachtungstabelle ist angelegt.

Vorgehen

Um eine Beobachtungstabelle zu speichern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Beobachtungstabelle, die Sie speichern wollen.
2. Falls Sie den voreingestellten Namen der Tabelle ändern wollen, wählen Sie im Kontextmenü den Befehl "Umbenennen" und geben Sie einen neuen Namen für die Tabelle ein.
3. Wählen Sie im Menü "Projekt" den Befehl "Speichern". Beachten Sie, dass mit diesem Speichervorgang das gesamte Projekt gespeichert wird.

Ergebnis

Die Inhalte der Beobachtungstabelle und das Projekt werden gespeichert.

Hinweis

Gespeicherte Beobachtungstabellen können Sie beim erneuten Testen Ihres Programms wiederholt zum Beobachten und Steuern von Variablen verwenden.

23.4.6 Variablen in die Beobachtungstabelle eingeben

23.4.6.1 Grundlagen für die Eingabe von Variablen in die Beobachtungstabelle

Empfohlene Vorgehensweise

- Bestimmen Sie die Variablen, deren Werte Sie beobachten oder steuern wollen und tragen Sie diese in die Beobachtungstabelle ein.
- Beachten Sie bei der Eingabe von Variablen in die Beobachtungstabelle, dass es sinnvoll ist, diese Variablen zuvor in der PLC-Variablentabelle zu definieren.
- Bei der Eingabe von Variablen gehen Sie von "außen" nach "innen" vor. Dies bedeutet, dass Sie zuerst die Variablen für die Eingänge in die Beobachtungstabelle eintragen. Danach geben Sie die Variablen ein, die von den Eingängen beeinflusst werden, bzw. die die Ausgänge beeinflussen. Anschließend geben Sie die Variablen für die Ausgänge ein.

Beispiel für das Ausfüllen einer Beobachtungstabelle

- Tragen Sie in der Spalte "Adresse" die absolute Adresse ein, die beobachtet oder gesteuert werden soll.
- Tragen Sie in der Spalte "Name" den symbolischen Namen für die Variable ein.
- Wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus, falls Sie die Voreinstellung nicht verwenden wollen.
- Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder steuern wollen. Tragen Sie für das Steuern die gewünschten Werte sowie einen Kommentar in die entsprechenden Spalten der Beobachtungstabelle ein.

Kommentarzeile erzeugen

Durch die Eingabe der Zeichenfolge "//" in der Spalte "Name" können Sie, falls gewünscht, eine Kommentarzeile erzeugen.

Syntaxprüfung

Wenn Sie Variablen in die Beobachtungstabelle eintragen, wird vor dem Verlassen der Zelle eine Syntaxprüfung durchgeführt. Fehlerhafte Einträge werden rot gekennzeichnet.

Hinweis

Wenn Sie den Cursor in eine rot markierte Zelle stellen, so erhalten Sie über eine kurze Information weitere Hinweise zum Fehler.

Siehe auch

Erlaubte Operanden für die Beobachtungstabelle (Seite 8092)

Erlaubte Steuerwerte für die Beobachtungstabelle (Seite 8094)

23.4.6.2 Erlaubte Operanden für die Beobachtungstabelle

Zulässige Operanden für die Beobachtungstabelle

Die folgende Tabelle zeigt die Operanden, die für die Beobachtungstabelle zugelassen sind:

Erlaubter Operand	Beispiel für Datentyp	Beispiel (Mnemonik deutsch)
Eingang/Ausgang/Merker	BOOL	E1.0; A1.7; M10.1 E0.0:P; A0.0:P
Eingang/Ausgang/Merker	BYTE	EB1; AB10; MB100 EB1:P; AB1:P
Eingang/Ausgang/Merker	WORD	EW1; AW10; MW100 EW2:P; AW3:P

Erlaubter Operand	Beispiel für Datentyp	Beispiel (Mnemonik deutsch)
Eingang/Ausgang/Merker	DWORD	ED4; AD10; MD100 ED2:P; AD1:P
Zeiten	TIMER	T1
Zähler	COUNTER	Z1
Datenbaustein	BOOL	DB1.DBX1.0
Datenbaustein	BYTE	DB1.DBB1
Datenbaustein	WORD	DB1.DBW1
Datenbaustein	DWORD	DB1.DBD1

Hinweis

Beachten Sie bitte die folgenden Hinweise für das Arbeiten mit der Beobachtungstabelle

- Die Eingabe von "DB0..." ist aufgrund interner Benutzung nicht erlaubt!
- Peripherieausgänge können zwar gesteuert, aber nicht beobachtet werden!
- Peripherieeingänge können zwar beobachtet, aber nicht gesteuert werden!

Hinweis

Kein Zugriff auf Peripherie-Bits für S7-300/400 CPUs

Die Beobachtungstabelle erlaubt für die CPUs der Familie S7-300/400 keinen Zugriff auf Peripherie-Bits, wie z.B. %I0.0:P; %Q0.0:P.

Diese können weder beobachtet noch gesteuert werden.

Bitte verwenden Sie stattdessen die Bits im Prozessabbild, wie z.B. %I0.0; %Q0.0.

ACHTUNG

Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen und Steuern von Peripherieausgängen

Beachten Sie, dass das Beobachten von Peripherieeingängen und das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

Siehe auch

Grundlagen für die Eingabe von Variablen in die Beobachtungstabelle (Seite 8091)

Übersicht über die gültigen Datentypen (Seite 243)

23.4.6.3 Erlaubte Steuerwerte für die Beobachtungstabelle

Eingabe von Steuerwerten in die Beobachtungstabelle

Die folgende Tabelle zeigt die Operanden, die für die Eingabe von Steuerwerten in die Beobachtungstabelle zugelassen sind:

Tabelle 23-1 Bitoperanden

Mögliche Bitoperanden	Beispiel für erlaubte Steuerwerte
E1.0	true
M1.7	false
A1.0	0
A1.1:P	1
DB1.DBX1.1	2#0
M1.6	2#1

Tabelle 23-2 Byteoperanden

Mögliche Byteoperanden	Beispiel für erlaubte Steuerwerte
EB1	2#0011_0011
MB12	B#16#1F
AB10	1F
AB11:P	'a'
DB1.DBB1	10

Tabelle 23-3 Wortoperanden

Mögliche Wortoperanden	Beispiel für erlaubte Steuerwerte
EW1	2#0011_0011_0011_0011
MW12	W#16#ABCD
MW14	ABCD
AW10	B#(12, 34)
AW12:P	12345
DB1.DBW1	'ab'
MW16	S5T#9s_340ms
MW18	C#123
MW9	D#2006-12-31

Tabelle 23-4 Doppelwortoperanden

Mögliche Doppelwortoperanden	Beispiel für erlaubte Steuerwerte
ED1	2#0011_0011_0011_0011_0011_0011_0011_0011
AD10	Dw#16#abcdef10
AD12:P	ABCDEF10
DB1.DBD2	b#(12,34,56,78)
MD8	L#-12
MD12	L#12
MD16	123456789
MD20	123456789
MD24	T#12s_345ms
MD28	Tod#1:2:34.567
MD32	P#e0.0

Tabelle 23-5 Zeiten

Mögliche Operanden vom Typ "Zeit"	Erlaubte Steuerwerte	Erläuterung
T1	0 ms	Zeitwert in Millisekunden (ms)
T12	20 ms	Zeitwert in Millisekunden (ms)
T14	12400 ms	Zeitwert in Millisekunden (ms)
T16	S5t#12s300ms	Zeitwert 12s 300 ms

Tabelle 23-6 Zähler

Mögliche Operanden vom Typ "Zähler"	Erlaubte Steuerwerte
Z1	0
Z14	20
Z16	C#123

Anmerkungen zu Zeiten und Zählern

- Zeiten

  Hinweis

  Das Steuern einer Zeit beeinflusst nur den Wert, nicht den Zustand. Die Zeit T1 kann auf den Wert "0" gesteuert werden, das Verknüpfungsergebnis bei U T1 wird nicht verändert.

  Die Zeitfolgen "s5t" und "s5time" können sowohl klein als auch groß geschrieben werden.

- Zähler

  Hinweis

  Das Steuern eines Zählers beeinflusst nur den Wert, nicht den Zustand. Der Zähler Z1 kann auf den Wert "0" gesteuert werden, das Verknüpfungsergebnis bei U Z1 wird nicht verändert.

23.4.6.4 Übersicht über die Anzeigeformate

Anzeigeformate in der Beobachtungstabelle

Mit der Auswahl eines Anzeigeformats legen Sie fest, wie der Wert einer Variablen dargestellt wird.

Beim Eintragen der Adresse wird automatisch ein Anzeigeformat vorbelegt. Um dieses zu ändern, können Sie das gewünschte Anzeigeformat aus der Klappliste in der Spalte "Anzeigeformate" auswählen. In der Klappliste werden nur die Anzeigeformate angeboten, die für diesen Datentyp zulässig sind. Das zuerst genannte Anzeigeformat ist das voreingestellte Format.

Beispiel

Die folgende Tabelle zeigt die für alle CPU-Familien in der Beobachtungstabelle zulässigen 32-Bit-Datentypen und ihre möglichen Anzeigeformate:

Datentyp	Mögliche Anzeigeformate
BOOL	Bool, Hex, BCD, Oktal, Bin, Dez, Dez+/-
BYTE	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Zeichen
WORD	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Unicode_Zeichen, SIMATIC_Zeit, Datum, Zähler
DWORD	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Unicode_Zeichen, Gleitpunktzahl, Tageszeit, Zeit
SINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen
INT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum
DINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit
USINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen
UINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum
UDINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit
REAL	Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit
DATE	Datum, Dez, Hex, Bin
TIME_OF_DAY	Tageszeit, Dez, Hex, Bin
TIME	Zeit, Hex, Bin, Dez, Dez+/-
DATE_AND_TIME	Datum und Uhrzeit,
TIMER	SIMATIC_Zeit, Hex, BCD, Bin
CHAR	Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-,
WCHAR	Unicode_Zeichen, Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-
STRING	Zeichenkette
WSTRING	Unicode_Zeichenkette
POINTER	Zeiger, Hex

Datentyp	Mögliche Anzeigeformate
COUNTER	Zähler, Hex, BCD, Bin
S5TIME	SIMATIC_Zeit, Hex, BCD, Bin

Für die CPU-Familie S7-1200 sind alle 32-Bit-Datentypen zulässig, siehe obige Tabelle, sowie zusätzlich der 64-Bit-Datentyp LREAL mit folgenden möglichen Anzeigeformaten:

Datentyp	Mögliche Anzeigeformate
LREAL	in einem mit dem TIA Portal < V12 erstellten Projekt: Gleitpunktzahl Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.
LREAL	in einem mit dem TIA Portal >= V12 erstellten Projekt: Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.

Für die CPU-Familie S7-1500 sind zusätzlich zu 32-Bit-Datentypen auch die in der Tabelle aufgelisteten 64-Bit-Datentypen zulässig mit folgenden möglichen Anzeigeformaten:

Datentyp	Mögliche Anzeigeformate
LWORD	Hex, Oktal, BCD, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Datum und Uhrzeit
LINT	Dez+/-, Dez, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
ULINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
LREAL	Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
LTIME	Zeit, Dez+/-, Dez, Hex, Bin
LTOD	Tageszeit, Dez, Hex, Bin
LDT	Datum und Uhrzeit, Dez, Hex, Bin

Weitere Informationen hierzu finden Sie bei der Beschreibung der gültigen Datentypen (Seite 243).

Hinweis

Rundungen bei Gleitpunktzahlen

In der Beobachtungstabelle werden Gleitpunktzahlen binär im IEEE-Format gespeichert. Da sich nicht jede an der Oberfläche darstellbare Gleitpunkt-Zahl (real, longreal) eindeutig auf das IEEE-Format abbilden lässt, kann es zu Rundungen bei Gleitpunktzahlen kommen.Wird eine solche gerundete Gleitpunktzahl in der Beobachtungstabelle kopiert und in ein anderes Eingabefeld wieder eingefügt, erhält man durch die Rundung unter Umständen eine geringe Abweichung.

Hinweis

Nur symbolische Adressierung möglich

In der Beobachtungstabelle können LongDataTypes wie z. B. LWORD oder LREAL nur symbolisch adressiert werden.

23.4.6.5 Anzeigeformat für Variablen auswählen

Vorgehen

Um das Anzeigeformat der Variablen auszuwählen, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Klicken Sie in der Spalte "Anzeigeformat" auf die gewünschte Zelle und öffnen Sie die Klappliste.
   Die zulässigen Anzeigeformate werden in der Klappliste angezeigt.
3. Wählen Sie aus der Klappliste das gewünschte Anzeigeformat aus.

   Hinweis

   Falls das ausgewählte Anzeigeformat nicht übernommen werden kann, wird automatisch das zuletzt ausgewählte Anzeigeformat angezeigt.

Siehe auch

Übersicht über die gültigen Datentypen (Seite 243)

23.4.6.6 Kommentarzeilen erstellen und bearbeiten

Grundlagen zu Kommentarzeilen

Zur besseren Strukturierung der Beobachtungstabelle können Sie zusätzlich zu den zeilenbezogenen Kommentaren in der Kommentarspalte nun auch komplette Kommentarzeilen erzeugen.

Die Inhalte der Kommentarzeile werden im Ordner "Sprachen & Ressourcen" im Register "Projekttexte" gespeichert und können in andere Projektsprachen übersetzt werden.

Zum Einfügen von Kommentarzeilen haben Sie alternativ folgende Möglichkeiten:

- Über das entsprechende Symbol in der Funktionsleiste der Beobachtungstabelle.
- Über den Menübefehl "Einfügen > Kommentarzeile einfügen".
- Über den Kontextmenübefehl "Kommentarzeile einfügen".
- Durch Eingabe der Zeichenfolge "//" in der Spalte "Name" der Beobachtungstabelle.

Kommentarzeilen einfügen

Zum Einfügen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Beobachtungstabelle und tragen Sie die gewünschten Adressen ein.
2. Fügen Sie eine neue Kommentarzeile ein:
   - Klicken Sie dazu auf das Symbol "Kommentarzeile einfügen" in der Funktionsleiste der Beobachtungstabelle.
   - Oder klicken Sie auf den Menübefehl "Einfügen > Kommentarzeile einfügen".
   - Oder klicken Sie auf den Kontextmenübefehl "Kommentarzeile einfügen".
     Ergebnis: Eine neue Kommentarzeile wird oberhalb der selektierten Zeile in die Beobachtungstabelle eingefügt.
3. Tragen Sie den gewünschten Kommentar in die Kommentarzeile ein. Der eingegebene Kommentar wird in Grün dargestellt.
4. Um alle von Ihnen eingegebenen Kommentare anzuzeigen, doppelklicken Sie in der Projektnavigation unter "Sprachen & Ressourcen" auf "Projekttexte".
5. Falls Sie in mehrsprachigen Projekten arbeiten und Ihren Kommentar in andere Sprachen übersetzen wollen, können Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" die zusätzlich zur Editiersprache gewünschten Projektsprachen einstellen.

Kommentarzeilen erzeugen

Zum Erzeugen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Beobachtungstabelle und tragen Sie die gewünschten Adressen ein.
2. Geben Sie in der Spalte "Name" die Zeichenfolge "//" in einer freien Zeile ein. Dadurch wird diese Zeile zur Kommentarzeile.
   In den restlichen Spalten können keine Kommentarzeilen erzeugt werden.

3. Tragen Sie den gewünschten Kommentar in die Kommentarzeile ein. Der eingegebene Kommentar wird in Grün dargestellt.
4. Um alle von Ihnen eingegebenen Kommentare anzuzeigen, doppelklicken Sie in der Projektnavigation unter "Sprachen & Ressourcen" auf "Projekttexte".
5. Falls Sie in mehrsprachigen Projekten arbeiten und Ihren Kommentar in andere Sprachen übersetzen wollen, können Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" die zusätzlich zur Editiersprache gewünschten Projektsprachen einstellen.

Kommentarzeilen löschen

Zum Löschen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Beobachtungstabelle mit vorhandenen Kommentarzeilen.
2. Löschen Sie den gesamten Kommentar mit der einleitenden Zeichenfolge "//", falls Sie diesen nicht mehr benötigen.
3. Löschen Sie alternativ nur die einleitende Zeichenfolge "//", so bleibt der vorhandene Kommentar erhalten und wird in der Spalte "Kommentar" in der Beobachtungstabelle angezeigt.

Hinweis

Löschen von Kommentareilen

Beim Löschen von Kommentarzeilen werden die für diese Kommentare eingestellten Projektsprachen und evtl. vorhandene Übersetzungen ebenfalls gelöscht.

23.4.7 Variablen in der Beobachtungstabelle beobachten

23.4.7.1 Einführung zum Beobachten von Variablen in der Beobachtungstabelle

Einführung

In der Beobachtungstabelle können Sie die Variablen der projektierten Ein- und Ausgabebaugruppen in der CPU beobachten, abhängig vom jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 8101). Um Variablen zu beobachten, muss eine Online-Verbindung zur CPU bestehen.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Optionen zum Beobachten von Variablen

Folgende Optionen für das Beobachten von Variablen stehen zur Verfügung:

- Sofort beobachten
  Dieser Befehl startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle sofort und einmalig.
- Alle beobachten
  Dieser Befehl startet das Beobachten aller sichtbaren Variablen in der aktiven Beobachtungstabelle, abhängig vom ausgewählten Beobachtungsmodus:
  - Im Basismodus ist der Beobachtungsmodus "permanent" voreingestellt.
  - Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen.

  Hinweis

  Wird der Beobachtungsmodus im Erweiterten Modus umgestellt und wieder in den Basismodus umgeschaltet, wird der zuvor eingestellte Beobachtungsmodus auch im Basismodus übernommen.

CPU-spezifische Einschränkungen beim Beobachten von Variablen

Es bestehen folgende CPU-spezifische Unterschiede:

- CPU S7-300/400:
  CPUs aus dieser Familie können lediglich die ersten 30 Zeichen eines Strings beobachten.
- CPU S7-1200/1500:
  CPUs aus dieser Familie können einen String bis zur Gesamtgröße von 254 Zeichen beobachten.

23.4.7.2 Festlegen des Beobachtungs- und Steuermodus

Einführung

Mit der Auswahl des Beobachtungs- und Steuermodus legen Sie den Triggerpunkt und die Dauer des Beobachtens von Variablen in der Beobachtungstabelle und der Forcetabelle fest.

Möglicher Beobachtungs- und Steuermodus (Dauer des Beobachtens bzw. Steuerns)

Folgende Beobachtungs- und Steuermodi stehen zur Verfügung:

Trigger	Ausführung	CPU-Status	Dauer
Permanent	Permanent Beim Beobachten: die Eingänge werden am Ende und die Ausgänge werden am Anfang des Zyklus beobachtet. Beim Steuern: die Eingänge werden am Anfang und die Ausgänge werden am Ende des Zyklus gesteuert.	RUN	Wird so lange ausgeführt, bis der Anwender die Aktion beendet oder die Online-Verbindung zur CPU unterbrochen wird.
Zyklusbeginn permanet	Permanent am Zyklusbeginn	RUN	Wird so lange ausgeführt, bis der Anwender die Aktion beendet oder die Online-Verbindung zur CPU unterbrochen wird.
Zyklusende permanent	Permanent am Zyklusende	RUN	Wird so lange ausgeführt, bis der Anwender die Aktion beendet oder die Online-Verbindung zur CPU unterbrochen wird.
Übergang von RUN nach STOP permanent	Permanent beim Übergang von RUN nach STOP	RUN > STOP	Wird so lange ausgeführt, bis der Anwender die Aktion beendet oder die Online-Verbindung zur CPU unterbrochen wird.
Zyklusbeginn einmalig	Einmalig am Zyklusbeginn	RUN	Wird nach der einmaligen Ausführung automatisch beendet.
Zyklusende einmalig	Einmalig am Zyklusende	RUN	Wird nach der einmaligen Ausführung automatisch beendet.
Übergang von RUN nach STOP einmalig	Einmalig beim Übergang von RUN nach STOP	STOP > RUN	Wird nach der einmaligen Ausführung automatisch beendet.

Besonderheiten beim Modus "Permanent"

Der Modus "Permanent" wird für das Beobachten und das Steuern von Variablen unterschiedlich ausgeführt.

- Beobachten: die Eingänge werden am Ende und die Ausgänge werden am Anfang des Zyklus beobachtet.
- Steuern: die Eingänge werden am Anfang und die Ausgänge werden am Ende des Zyklus gesteuert.

Triggerpunkt festlegen

Die Triggerpunkte "Zyklusbeginn", "Zyklusende" und "Übergang von RUN nach STOP" legen fest, zu welchem Zeitpunkt die Variablen aus der CPU gelesen bzw. in der CPU aktualisiert werden sollen.

Das folgende Bild zeigt die Lage der genannten Triggerpunkte:

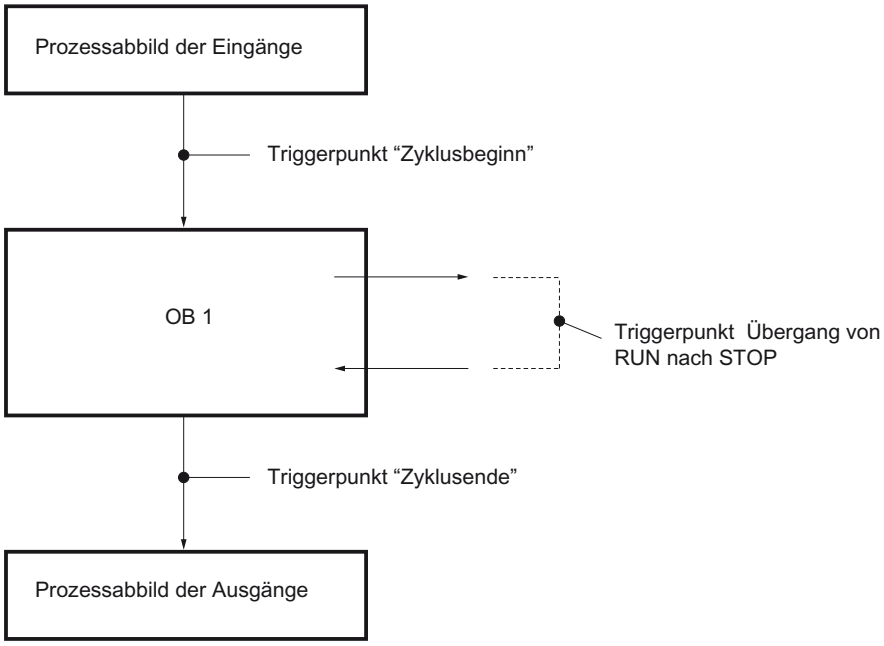

Lage der Triggerpunkte

Aus der Lage der Triggerpunkte geht hervor:

- Das Steuern von Eingängen ist nur sinnvoll am Zyklusbeginn (entspricht dem Beginn des Anwenderprogramms OB 1), da sonst das Prozessabbild der Eingänge nach dem Steuern wieder aktualisiert und damit überschrieben wird.
- Das Steuern von Ausgängen ist nur sinnvoll am Zyklusende (entspricht dem Ende des Anwenderprogramms OB 1), da sonst das Prozessabbild der Ausgänge vom Anwenderprogramm überschrieben werden kann.
- Der gesteuerte Wert wird in der Spalte "Beobachtungswert" angezeigt, sofern aktiv beobachtet wird und der gesteuerte Wert nicht vom Anwenderprogramm überschrieben wird.

Steuern von Variablen

Beim Steuern von Variablen gilt bezüglich der Triggerpunkte:

- Wenn Sie als Steuermodus "einmalig" eingestellt haben, so erhalten Sie eine Meldung, falls die ausgewählten Variablen nicht zu steuern sind.
- Beim Steuermodus "permanent" erhalten Sie keine Rückmeldung.

Anmerkung zum Befehl "Sofort steuern"

Sie können die Werte ausgewählter Variablen über den Befehl "Online > Steuern >Sofort steuern" sofort steuern. Dieser Auftrag wird einmalig und schnellstmöglich durchgeführt, ohne Bezug zu einer bestimmten Stelle (Triggerpunkt) im Anwenderprogramm. Diese Funktion wird hauptsächlich im STOP-Zustand der CPU zum Steuern angewendet.

23.4.7.3 Variablen "Alle beobachten"

Einführung

Mit dem Befehl "Alle beobachten" können Sie das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle starten. Im Basismodus der Beobachtungstabelle ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen. In diesem Fall werden die Variablen bezüglich der festgelegten Triggerpunkte beobachtet.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehensweise

Um den Befehl "Alle beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Beobachtungstabelle ein.
2. Schalten Sie in den Erweiterten Modus um, indem Sie in der Funktionsleiste auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" klicken.
3. Wenn Sie den voreingestellten Beobachtungsmodus einer Variablen ändern wollen, klicken Sie in der Spalte "Beobachten mit Trigger" auf die entsprechende Zelle und wählen Sie aus der Klappliste den gewünschten Beobachtungsmodus aus.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Alle beobachten".

Ergebnis

Die Variablen der aktiven Beobachtungstabelle werden mit dem jeweils eingestellten Beobachtungsmodus beobachtet.

Siehe auch

Symbole in der Beobachtungstabelle (Seite 8087)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 8086)

Variablen in die Beobachtungstabelle eingeben (Seite 8091)

23.4.7.4 Variablen "Sofort beobachten"

Einführung

Mit dem Befehl "Sofort beobachten" wird das Beobachten der Variablen sofort, ohne Bezug auf bestimmte Triggerpunkte gestartet. Die Variablenwerte werden einmalig ausgelesen und in der Beobachtungstabelle angezeigt.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehensweise

Um den Befehl "Sofort beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Beobachtungstabelle ein.
2. Klicken Sie in der Funktionsleiste auf das Symbol "Sofort beobachten".

Ergebnis

Die Variablen der aktiven Beobachtungstabelle werden sofort und einmalig beobachtet.

Siehe auch

Symbole in der Beobachtungstabelle (Seite 8087)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 8086)

Variablen in die Beobachtungstabelle eingeben (Seite 8091)

23.4.8 Variablen in der Beobachtungstabelle steuern

23.4.8.1 Einführung zum Steuern von Variablen

Einführung

Mithilfe der Beobachtungstabelle können Sie die Variablen der projektierten Ein- und Ausgabebaugruppen in der CPU steuern, abhängig vom jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 8101).

Um die Variablen zu beobachten, muss eine Online-Verbindung zur CPU bestehen.

⚠ GEFAHR
Gefahr beim Steuern: Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

ACHTUNG
Gefahr durch paralleles Steuern eines identischen Operanden mit unterschiedlichen Steuerwerten in mehreren Beobachtungstabellen Beim Arbeiten mit mehreren Beobachtungstabellen ist darauf zu achten, identische Operanden nicht mehrfach mit unterschiedlichen Steuerwerten permanent zu steuern! Wenn ein identischer Operand in mehreren Beobachtungstabellen gleichzeitig mit unterschiedlichen Steuerwerten permanent gesteuert wird, so wird in allen Beobachtungstabellen als Beobachtungswert der zuletzt gesteuerte Wert angezeigt, da in diesem Fall immer der zuletzt zugewiesene Steuerwert gesteuert wird.

Optionen zum Steuern von Variablen

Folgende Optionen für das Steuern von Variablen stehen zur Verfügung:

- Steuern auf "0"
  Dieser Befehl steuert die selektierte Adresse auf den Steuerwert "0".
- Steuern auf "1"
  Dieser Befehl steuert die selektierte Adresse auf den Steuerwert "1".
- Einmalig und sofort steuern
  Dieser Befehl steuert alle aktivierten Adressen in der aktiven Beobachtungstabelle "einmalig und sofort".
- Steuern mit Trigger
  Dieser Befehl steuert alle aktivierten Adressen in der aktiven Beobachtungstabelle mit dem jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 8101).
  Die Funktion "Steuern mit Trigger" ist nur im Erweiterten Modus verfügbar. Sie erhalten keine Rückmeldung, ob die aktivierten Adressen tatsächlich mit dem vorgegebenen Wert gesteuert wurden oder nicht. Falls Sie eine solche Rückmeldung benötigen, sollten Sie die Funktion "Einmalig und sofort steuern" verwenden.
- Peripherieausgänge freischalten
  Dieser Befehl schaltet die Befehlsausgabesperre (BASP) ab.
  Diese Funktion ist nur durchführbar im Erweiterten Modus, wenn die CPU im STOP ist und wenn das Forcen (Seite 8132) von Variablen nicht aktiviert ist. Deaktivieren Sie diese Funktion, falls gewünscht, in der Forcetabelle.

Hinweis

Beachten Sie beim Steuern:

Das Steuern von Variablen kann **nicht** rückgängig gemacht werden!

23.4.8.2 Variablen steuern auf "0"

Einführung

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR

Gefahr beim Steuern:

Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!
Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehen

Um Variablen auf "0" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Wählen Sie den Befehl "Online > Steuern > Steuern auf 0", um die ausgewählte Adresse mit dem vorgegebenen Wert zu steuern.

Ergebnis

Die ausgewählte Adresse wird auf "0" gesteuert.

Hinweis

Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

23.4.8.3 Variablen steuern auf "1"

Einführung

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR
Gefahr beim Steuern: Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann! Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehen

Um Variablen auf "1" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Wählen Sie den Befehl "Online > Steuern > Steuern auf 1", um die ausgewählte Adresse mit dem vorgegebenen Wert zu steuern.

Ergebnis

Die ausgewählte Adresse wird auf "1" gesteuert.

Hinweis

Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

23.4.8.4 Variablen "Sofort steuern"

Einführung

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese sofort steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ **GEFAHR**

Gefahr beim Steuern:

Ein Verändern der der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!
Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

ACHTUNG

Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen

Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehen

Um Variablen sofort zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.
   Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Wählen Sie den Befehl "Online > Steuern > Steuern sofort", um die ausgewählten Adressen einmalig und sofort mit den vorgegebenen Werten zu steuern.

Ergebnis

Die ausgewählten Adressen werden einmalig und sofort gesteuert.

Hinweis

Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

23.4.8.5 Variablen "Steuern mit Trigger"

Einführung

Sie können Adressen in Abhängigkeit vom definierten Beobachtungs- und Steuerungsmodus Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag durchgeführt, wie im Beobachtungs- und Steuermodus festgelegt, mit Bezug zu der als Trigger bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR

Gefahr beim Steuern:

Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!
Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

ACHTUNG
Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen
Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!
Die CPU geht dann in den Betriebszustand "STOP".

Voraussetzungen

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die Beobachtungstabelle muss im Erweiterten Modus sein.

Vorgehen

Um Variablen "mit Trigger" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.
   Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Schalten Sie über das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" in der Funktionsleiste oder über den Befehl "Online > Erweiterter Modus" in den erweiterten Modus um.
   Die Spalten "Beobachten mit Trigger" und "Steuern mit Trigger" werden angezeigt.
4. Wählen Sie in der Spalte "Steuern mit Trigger" den gewünschten Steuermodus aus der Auswahlliste aus. Folgende Optionen stehen zur Verfügung:
   - Permanent
   - Zyklusbeginn, permanent
   - Zyklusbeginn, einmalig
   - Zyklusende, permanent
   - Zyklusende, einmalig
   - Übergang nach STOP, permanent
   - Übergang nach STOP, einmalig
5. Starten Sie das Steuern über den Befehl "Online > Steuern > Steuern mit Trigger".
6. Bestätigen Sie die Anfrage mit "Ja", falls Sie das Steuern mit Trigger starten wollen.

Ergebnis

Die aktivierten Variablen werden mit dem ausgewählten Beobachtungs- und Steuermodus gesteuert. Das gelbe Dreieck wird nicht mehr angezeigt.

Hinweis

Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

23.4.8.6 Peripherieausgänge freischalten

Einführung

Die Funktion "Peripherieausgänge freischalten" schaltet die Befehlsausgabesperre der Peripherieausgänge (BASP) ab. Dies ermöglicht das Steuern der Peripherieausgänge, wenn sich die CPU im Betriebszustand STOP befindet.

Diese Funktion steht in der Beobachtungstabelle nur im "Erweiterten Modus" und nur für CPUs aus der Familie S7-300/400 und S7-1200 zur Verfügung.

⚠ GEFAHR

Gefahr beim Freischalten der Peripherieausgänge:

Beachten Sie, dass das Freischalten der Peripherieausgänge schwere Sach- und Personenschäden verursachen kann!
Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Peripherieausgänge freischalten" ausführen.

Voraussetzungen

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die CPU muss für das Freischalten der Peripherieausgänge im Betriebszustand STOP sein.
- Die Beobachtungstabelle muss im Erweiterten Modus sein.
- Das Forcen (Seite 8132) von Variablen darf nicht aktiv sein.

Hinweis

Funktion "Peripherieausgänge freischalten"

- Diese Funktion ist nur im Betriebszustand STOP möglich. Sie wird beendet durch einen Betriebszustandswechsel der CPU und beim Schließen der Online-Verbindung.
- Solange die Funktion aktiv ist, kann nicht geforct werden.

Vorgehen

Um die Peripherieausgänge im Betriebszustand STOP freizuschalten, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.
   Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Schalten Sie über das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" in der Funktionsleiste oder über den Befehl "Online > Erweiterter Modus" in den erweiterten Modus um.
   Die Spalten "Beobachten mit Trigger" und "Steuern mit Trigger" werden angezeigt.
4. Schalten Sie die zugeordnete CPU über das Bedienpanel in den Betriebszustand STOP.
5. Rufen Sie mit der rechten Maustaste das Kontextmenü auf und aktivieren Sie den Befehl "Peripherieausgänge freischalten".
6. Bestätigen Sie die Anfrage mit "Ja", falls Sie die Befehlsausgabesperre für die Peripherieausgänge freischalten wollen.
7. Steuern Sie die Peripherieausgänge über den Befehl "Online > Steuern > Steuern sofort".

Ergebnis

Die Peripherieausgänge werden mit den ausgewählten Steuerwerten gesteuert. Das gelbe Dreieck wird nicht mehr angezeigt.

Freischaltung der Peripherieausgänge

Die Funktion "Peripherieausgänge freischalten" bleibt so lange aktiviert bis:

- Der Befehl "Peripherieausgänge freischalten" über das Kontextmenü oder über den Befehl "Online > Steuern > Peripherieausgänge freischalten" wieder deaktiviert wird.
- Die CPU nicht mehr im Betriebszustand STOP ist.
- Die Online-Verbindung beendet wird.

Hinweis

Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

23.5 Testen mit der Forcetabelle

23.5.1 Einführung zum Testen mit der Forcetabelle

Übersicht

Mit Hilfe der Forcetabelle können Sie einzelne Variablen des Anwenderprogramms mit festen Werten vorbelegen. Diesen Vorgang nennt man "Forcen".

Innerhalb der Forcetabelle stehen Ihnen folgende Funktionen zur Verfügung:

- Beobachten von Variablen
  Damit lassen sich die aktuellen Werte einzelner Variablen eines Anwenderprogramms bzw. einer CPU am PG/PC anzeigen. Das Beobachten kann mit oder ohne Triggerbedingung ausgeführt werden.
- Forcen von Variablen
  Damit können Sie einzelnen Peripherie-Variablen eines Anwenderprogramms einen festen Wert zuweisen.

Beobachten und Forcen von Variablen

Das Beobachten und Forcen von Variablen ist immer abhängig vom Operandenumfang der eingesetzten CPU.

Folgende Variablen können beobachtet werden:

- Eingänge, Ausgänge und Merker
- Inhalte von Datenbausteinen
- Peripherieeingänge

Folgende Variablen können geforct werden:

- Peripherieeingänge
- Peripherieausgänge

Beispiel

Unabhängig von der jeweils eingesetzten CPU kann ausschließlich Peripherie geforct werden, also z. B.: "Tag_1":P oder "AW0:P" oder "EW0:P". Beachten Sie dabei, dass "Tag_1":P nicht die symbolische Bezeichnung für einen Merker sein darf.

Einsatzmöglichkeit

Die Forcetabelle bietet den Vorteil, sowohl unterschiedliche Testumgebungen simulieren zu können als auch Variablen in der CPU mit einem festen Wert überschreiben zu können. Damit haben Sie die Möglichkeit, regulierend in den laufenden Prozess einzugreifen.

Siehe auch

Aufbau der Forcetabelle (Seite 8117)

Basismodus und Erweiterter Modus in der Forcetabelle (Seite 8118)

Symbole in der Forcetabelle (Seite 8118)

Forcetabelle öffnen und bearbeiten (Seite 8120)

23.5.2 Sicherheitsvorkehrungen beim Forcen von Variablen

Sicherheitsvorkehrungen zum Forcen von Variablen

Da Sie mit der Funktionalität des Forcens permanent in den Prozess eingreifen, sind die folgenden Hinweise unbedingt zu beachten:

⚠ GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

⚠ VORSICHT

Personen- und Sachschäden verhindern!

- Bevor Sie die Funktion "Forcen" starten, sollten Sie sicherstellen, dass niemand gleichzeitig auf derselben CPU diese Funktion ebenfalls ausführt.
- Das Forcen kann nur durch Klicken auf das Symbol "Forcen beenden" oder über den Befehl "Online > Forcen > Forcen beenden" beendet werden. Das Schließen der aktiven Forcetabelle beendet das Forcen **nicht**!
- "Forcen" kann **nicht** rückgängig gemacht werden!
- Informieren Sie sich über die Unterschiede zwischen "Steuern von Variablen" (Seite 8106) und "Forcen von Variablen" (Seite 8132)".
- Falls eine CPU die Funktionalität des "Forcens" nicht unterstützt, so sind alle das Forcen betreffenden Symbole nicht anwählbar.
- Wenn die Funktion "Peripherieausgänge freischalten" auf einer CPU aktiv ist, ist das Forcen auf dieser CPU **nicht** möglich. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

23.5.3 Aufbau der Forcetabelle

Einführung

In die Forcetabelle tragen Sie die von Ihnen definierten und ausgewählten CPU-weit gültigen Variablen ein, die in der zugeordneten CPU geforct werden sollen. Es können nur Peripherieeingänge und Peripherieausgänge geforct werden.

Für jede im Projekt angelegte CPU wird automatisch eine Forcetabelle im Ordner "Beobachtungs- und Forcetabellen" angelegt. Einer CPU kann nur eine einzige Forcetabelle zugeordnet werden. Diese Forcetabelle zeigt alle in der zugeordneten CPU geforcten Adressen an.

Aufbau der Forcetabelle

Die in der Forcetabelle angezeigten Spalten sind abhängig davon, ob Sie im Basismodus oder im Erweiterten Modus arbeiten.

Im Erweiterten Modus wird die Spalte "Beobachten mit Trigger" zusätzlich angezeigt.

Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten im Basismodus und im Erweiterten Modus:

Mode	Spalte	Erläuterung
Basismodus	i	Kennzeichnungsspalte
	Name	Name der eingefügten Variablen
	Adresse	Adresse der eingefügten Variablen
	Anzeigeformat	Ausgewähltes Anzeigeformat
	Beobachtungswert	Werte der Variablen, abhängig vom ausgewählten Anzeigeformat.
	Forcewert	Wert, mit dem die Variable geforct wird.
	F ("Forcen")	Auswahl der zu forcenden Variablen durch Aktivieren des entsprechenden Optionskästchens.
	Kommentar	Möglichkeit, einen Kommentar zur Dokumentation der Variablen in der Focetabelle einzutragen.
	Variablen-Kommentar	Anzeige des Kommentars zur ausgewählten Variable, der in anderen Editoren eingetragen wurde. Diese Spalte ist nicht editierbar.
Im Erweiterten Modus wird zusätzlich folgende Spalte angezeigt:	Beobachten mit Trigger	Anzeige des ausgewählten Beobachtungsmodus

Siehe auch

Symbole in der Forcetabelle (Seite 8118)

Basismodus und Erweiterter Modus in der Forcetabelle (Seite 8118)

23.5.4 Basismodus und Erweiterter Modus in der Forcetabelle

Unterschied zwischen Basismodus und Erweitertem Modus in der Forcetabelle

Im Erweiterten Modus wird in der Forcetabelle zusätzlich die Spalte "Beobachten mit Trigger" angezeigt.

Eine detaillierte Auflistung der jeweiligen Spalten finden Sie unter Aufbau der Forcetabelle (Seite 8117).

Umschalten zwischen Basismodus und Erweitertem Modus

Um zwischen Basismodus und Erweitertem Modus umzuschalten, haben Sie folgende Möglichkeiten:

- Klicken Sie auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden". Ein erneutes Klicken auf dieses Symbol schaltet wieder in den Basismodus um.
  Oder:
- Aktivieren Sie im Menü "Online" das Optionskästchen vor dem Befehl "Erweiterter Modus". Ein Deaktivieren des Kontrollkästchens schaltet wieder in den Basismodus um.

Funktionalität im Erweiterten Modus

Folgende Funktionalität ist nur im Erweiterten Modus möglich:

- Beobachten mit Trigger
- Peripherieeingänge beobachten

23.5.5 Symbole in der Forcetabelle

Bedeutung der Symbole

Die folgende Tabelle zeigt die Bedeutung der Symbole in der Forcetabelle:

Symbol	Bedeutung
	Kennzeichnet eine Tabelle innerhalb der Projektnavigation als Forcetabelle.
	Kennzeichnungsspalte
	Fügt eine Zeile vor der markierten Zeile ein.
	Fügt eine Zeile nach der markierten Zeile ein.
	Fügt eine Kommentarzeile oberhalb der markierten Zeile ein.
	Blendet alle Spalten des Erweiterten Modus ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Spalten des Erweiterten Modus wieder ausgeblendet.

Symbol	Bedeutung
	Aktualisiert alle momentan auf der CPU geforcten Operanden und Werte in der geöffneten Forcetabelle.
	Startet das Forcen für alle Adressen der ausgewählten Variablen. Falls das Forcen bereits läuft, wird der vorherige Vorgang unterbrechungsfrei ersetzt.
	Beendet das Forcen von Adressen in der Forcetabelle.
	Startet das Beobachten der sichtbaren Variablen in der Forcetabelle. Im Basismodus ist für das Beobachten "permanent" voreingestellt. Im Erweiterten Modus wird eine zusätzliche Spalte eingeblendet und Sie können für das Beobachten der Variablen bestimmte Triggerpunkte einstellen.
	Startet das Beobachten der sichtbaren Variablen in der Forcetabelle. Dieser Befehl wird sofort ausgeführt und die Variablen werden einmalig beobachtet.
	Zeigt das Optionskästchen für die Auswahl der zu forcenden Variablen an.
	Zeigt an, dass eine Adresse evtl. nicht vollständig geforct werden kann. Beispiel: Es ist zwar möglich, die Adresse AW0:P zu forcen, es ist jedoch nicht möglich, die Adresse AD0:P zu forcen, da dieser Adressbereich auf der CPU evtl. nicht vorhanden ist.
	Zeigt an, dass eine Adresse nicht beobachtet werden kann.
	Zeigt an, dass eine Adresse geforct wird.
	Zeigt an, dass eine Adresse teilweise geforct wird.
	Zeigt an, dass die zugehörige Peripherie-Adresse geforct wird.
	Zeigt an, dass ein Syntaxfehler aufgetreten ist.
	Zeigt an, dass die Adresse zwar ausgewählt ist, im Moment aber z. B. noch nicht geforct wird.
	Zeigt an, dass die selektierte Variable auf den Wert "1" beobachtet wurde.
	Zeigt an, dass die selektierte Variable auf den Wert "0" beobachtet wurde.
	Zeigt an, dass die Adresse mehrfach verwendet wird.
	Zeigt an, dass ein Ersatzwert verwendet wird. Ersatzwerte sind Werte, die bei fehlerhaften Signalausgabebaugruppen an den Prozess ausgegeben werden, bzw. bei fehlerhaften Signaleingabebaugruppen im Anwenderprogramm anstelle eines Prozesswertes verwendet werden. Die Ersatzwerte sind vom Anwender vorgebbar (z. B. alten Wert beibehalten).

Siehe auch

Aufbau der Forcetabelle (Seite 8117)

23.5.6 Forcetabelle öffnen und bearbeiten

23.5.6.1 Forcetabelle anzeigen

Einführung

Eine Forcetabelle kann nicht neu erstellt werden, sie ist für jede CPU bereits einmal vorhanden. Sie ist dieser CPU fest zugeordnet und kann nicht kopiert oder dupliziert werden.

Voraussetzung

Ein Projekt mit einer zugeordneten CPU muss geöffnet sein.

Anzeige einer Forcetabelle

Die Forcetabelle wird immer unterhalb einer CPU im Ordner "Beobachtungs- und Forcetabellen" angezeigt.

23.5.6.2 Forcetabelle öffnen

Voraussetzung

Ein Projekt mit einer zugeordneten CPU muss angelegt sein.

Vorgehen

Um eine Forcetabelle zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Beobachtungs- und Forcetabellen" unterhalb der gewünschten CPU.
2. Doppelklicken Sie auf die in diesem Ordner enthaltene "Forcetabelle".

Ergebnis

Die ausgewählte Forcetabelle wird geöffnet.

23.5.6.3 Forcetabelle speichern

Voraussetzung

Ein Projekt mit einer zugeordneten CPU ist angelegt.

Vorgehen

Um eine Forcetabelle zu speichern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Änderungen in die Forcetabelle ein.
2. Wählen Sie im Menü "Projekt" den Befehl "Speichern" oder klicken auf das Symbol "Projekt speichern" in der Funktionsleiste. Beachten Sie, dass mit diesem Speichervorgang das gesamte Projekt gespeichert wird.

Ergebnis

Die Inhalte der Forcetabelle und das zugehörige Projekt werden gespeichert.

Hinweis

Eine Forcetabelle kann nicht umbenannt werden!

23.5.7 Variablen in die Forcetabelle eingeben

23.5.7.1 Grundlagen für die Eingabe von Variablen in die Forcetabelle

Empfohlene Vorgehensweise

Bestimmen Sie die Variablen, deren Werte Sie beobachten oder forcen wollen und tragen Sie diese in die Forcetabelle ein.

Beachten Sie bei der Eingabe von Variablen in die Forcetabelle, dass es sinnvoll ist, diese Variablen zuvor in der PLC-Variablentabelle zu definieren.

Beispiel für das Ausfüllen einer Forcetabelle

- Sie können in der Spalte "Adresse" die absolute Adresse eintragen, die geforct oder beobachtet werden soll, oder Sie tragen in der Spalte "Name" den symbolischen Namen für die Variable ein.
- Wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus, falls Sie die Voreinstellung nicht verwenden wollen.
- Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder forcen wollen. Tragen Sie dazu den gewünschten Forcewert sowie einen Kommentar in die entsprechenden Spalten der Forcetabelle ein.
- Beachten Sie bitte, dass nur Peripherieeingänge und Peripherieausgänge geforct werden können und informieren Sie sich über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134).

Kommentarzeile erzeugen

Durch Eingabe der Zeichenfolge "//" in der Spalte "Name" können Sie, falls gewünscht, eine Kommentarzeile erzeugen.

Syntaxprüfung

Wenn Sie Variablen in die Forcetabelle eintragen, so wird vor dem Verlassen der Zelle eine Syntaxprüfung durchgeführt. Fehlerhafte Einträge werden rot gekennzeichnet.

Hinweis

Wenn Sie den Cursor in eine rot markierte Zelle stellen, so erhalten Sie über eine kurze Information weitere Hinweise zum Fehler.

23.5.7.2 Erlaubte Operanden für die Forcetabelle

Zulässige Operanden für die Forcetabelle

Die folgende Tabelle zeigt die Operanden, die für das Forcen in der Forcetabelle zugelassen sind:

Erlaubter Operand	Beispiel für Datentyp	Beispiel (Mnemonik deutsch)
Peripherieeingang/Peripherieausgang	BOOL	E0.0:P; A0.0:P
Peripherieeingang/Peripherieausgang	BYTE	EB1:P; AB1:P
Peripherieeingang/Peripherieausgang	WORD	EW2:P; AW3:P
Peripherieeingang/Peripherieausgang	DWORD	ED2:P; AD1:P

Die nachfolgende Tabelle zeigt die Operanden, die für das Beobachten in der Forcetabelle zugelassen sind:

Erlaubter Operand	Beispiel für Datentyp	Beispiel (Mnemonik deutsch)
Eingang/Ausgang/Merker	BOOL	E1.0; A1.7; M10.1 E0.0:P
Eingang/Ausgang/Merker	BYTE	EB1; AB10; MB100 EB1:P
Eingang/Ausgang/Merker	WORD	EW1; AW10; MW100 EW2:P
Eingang/Ausgang/Merker	DWORD	ED4; AD10; MD100 ED2:P
Zeiten	TIMER	T1
Zähler	COUNTER	Z1

Erlaubter Operand	Beispiel für Datentyp	Beispiel (Mnemonik deutsch)
Datenbaustein	BOOL	DB1.DBX1.0
Datenbaustein	BYTE	DB1.DBB1
Datenbaustein	WORD	DB1.DBW1
Datenbaustein	DWORD	DB1.DBD1

Hinweis

Die Eingabe von "DB0..." ist aufgrund interner Benutzung nicht erlaubt!

Hinweis

Kein Zugriff auf Peripherie-Bits für S7-400 CPUs

Die Forcetabelle erlaubt für die CPUs der Familie S7-400 keinen Zugriff auf Peripherie-Bits, wie z.B. %I0.0:P; %Q0.0:P.

Diese können nicht geforct werden!

23.5.7.3 Erlaubte Forcewerte für die Forcetabelle

Eingabe von Forcewerten in die Forcetabelle

Die folgende Tabelle zeigt die Operanden, die für die Eingabe von Forcewerten in die Forcetabelle zugelassen sind:

Tabelle 23-7 Bitoperanden

Mögliche Bitoperanden	Beispiel für erlaubte Forcewerte
E1.0:P	true
E1.1:P	false
A1.0:P	0
A1.1:P	1
E2.0:P	2#0
E2.1:P	2#1

Tabelle 23-8 Byteoperanden

Mögliche Byteoperanden	Beispiel für erlaubte Forcewerte
EB1:P	2#00110011
EB2:P	B#16#1F
AB14:P	1F

Mögliche Byteoperanden	Beispiel für erlaubte Forcewerte
AB10:P	'a'
EB3:P	10

Tabelle 23-9 Wortoperanden

Mögliche Wortoperanden	Beispiel für erlaubte Forcewerte
EW0:P	2#0011001100110011
EW2:P	W#16#ABCD
AW10:P	ABCD
AW12:P	B#(12, 34)
EW4:P	'ab'
EW6:P	12300
EW8:P	S5T#9S_300ms
EW10:P	C#123
EW12:P	D#2006-12-31

Tabelle 23-10 Doppelwortoperanden

Mögliche Doppelwortoperanden	Beispiel für erlaubte Forcewerte
ED0:P	2#00110011001100110011001100110011
ED4:P	1.2
AD10:P	1.234.e4
AD14:P	Dw#16#abcdef10
ED8:P	16#ABCDEF10
ED12:P	b#(12,34,56,78)
ED16:P	L#-12
ED20:P	L#12
ED24:P	123456789
ED28:P	123456789
ED32:P	T#12s300ms
ED36:P	Tod#14:20:40.645
ED40:P	P#e0.0

23.5.7.4 Übersicht über die Anzeigeformate

Anzeigeformate in der Forcetabelle

Mit der Auswahl eines Anzeigeformats legen Sie fest, wie der Wert einer Variablen dargestellt wird.

Beim Eintragen der Adresse wird automatisch ein Anzeigeformat vorbelegt. Um dieses zu ändern, können Sie das gewünschte Anzeigeformat aus der Klappliste in der Spalte "Anzeigeformate" auswählen. In der Klappliste werden nur die Anzeigeformate angeboten, die

für diesen Datentyp zulässig sind. Das zuerst genannte Anzeigeformat ist das voreingestellte Format.

Beispiel

Die folgende Tabelle zeigt die für alle CPU-Familien in der Forcetabelle zulässigen 32-Bit-Datentypen und ihre möglichen Anzeigeformate:

Datentyp	Mögliche Anzeigeformate
BOOL	Bool, Hex, BCD, Oktal, Bin, Dez, Dez+/-
BYTE	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Zeichen
WORD	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, SIMATIC_Zeit, Datum, Unicode_Zeichen, Zähler
DWORD	Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Gleitpunktzahl, Tageszeit, Zeit, Unicode_Zeichen
SINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen
INT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum
DINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit
USINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen
UINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum
UDINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit
REAL	Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit
DATE	Datum, Dez, Hex, Bin
TIME_OF_DAY	Tageszeit, Dez, Hex, Bin
TIME	Zeit, Hex, Bin, Dez, Dez+/-
DATE_AND_TIME	Datum und Uhrzeit,
TIMER	SIMATIC_Zeit, Hex, BCD, Bin
CHAR	Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-,
WCHAR	Unicode_Zeichen, Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-
STRING	Zeichenkette
WSTRING	Unicode_Zeichenkette
POINTER	Zeiger, Hex
COUNTER	Zähler, Hex, BCD, Bin
S5TIME	SIMATIC_Zeit, Hex, BCD, Bin

Für die CPU-Familie S7-1200 sind alle 32-Bit-Datentypen zulässig, siehe obige Tabelle, sowie zusätzlich der 64-Bit-Datentyp LREAL mit folgenden möglichen Anzeigeformaten:

Datentyp	Mögliche Anzeigeformate
LREAL	in einem mit dem TIA Portal < V12 erstellten Projekt: Gleitpunktzahl Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.
LREAL	in einem mit dem TIA Portal >= V12 erstellten Projekt: Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.

Für die CPU-Familie S7-1500 sind zusätzlich zu 32-Bit-Datentypen auch die in der Tabelle aufgelisteten 64-Bit-Datentypen zulässig mit folgenden möglichen Anzeigeformaten:

Datentyp	Mögliche Anzeigeformate
LWORD	Hex, Oktal, BCD, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Datum und Uhrzeit
LINT	Dez+/-, Dez, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
ULINT	Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
LREAL	Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit
LTIME	Zeit, Dez+/-, Dez, Hex, Bin
LTOD	Dez, Hex, Bin, Tageszeit
LDT	Dez, Hex, Datum und Uhrzeit, Bin

Weitere Informationen hierzu finden Sie bei der Beschreibung der gültigen Datentypen (Seite 243) .

Hinweis

Rundungen bei Gleitpunktzahlen

In der Forcetabelle werden Gleitpunktzahlen binär im IEEE-Format gespeichert. Da sich nicht jede an der Oberfläche darstellbare Gleitpunkt-Zahl (real, longreal) eindeutig auf das IEEE-Format abbilden lässt, kann es zu Rundungen bei Gleitpunktzahlen kommen.Wird eine solche gerundete Gleitpunktzahl in der Forcetabelle kopiert und in ein anderes Eingabefeld wieder eingefügt, erhält man durch die Rundung unter Umständen eine geringe Abweichung.

Hinweis

Nur symbolische Adressierung möglich

In der Forcetabelle können LongDataTypes wie z. B. LWORD oder LREAL nur symbolisch adressiert werden.

23.5.7.5 Anzeigeformat für Variablen auswählen

Vorgehen

Um das Anzeigeformat der Variablen auszuwählen, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
2. Klicken Sie in der Spalte "Anzeigeformat" auf die gewünschte Zelle und öffnen Sie die Klappliste.
   Die zulässigen Anzeigeformate werden in der Klappliste angezeigt.
3. Wählen Sie aus der Klappliste das gewünschte Anzeigeformat aus.

   Hinweis

   Falls das ausgewählte Anzeigeformat nicht übernommen werden kann, wird automatisch das zuletzt ausgewählte Anzeigeformat angezeigt.

23.5.7.6 Kommentarzeilen erstellen und bearbeiten

Grundlagen zu Kommentarzeilen

Zur besseren Strukturierung der Forcetabelle können Sie zusätzlich zu den zeilenbezogenen Kommentaren in der Kommentarspalte nun auch komplette Kommentarzeilen erzeugen.

Die Inhalte der Kommentarzeile werden im Ordner "Sprachen & Ressourcen" im Register "Projekttexte" gespeichert und können in andere Projektsprachen übersetzt werden.

- Über das entsprechende Symbol in der Funktionsleiste der Forcetabelle.
- Über den Menübefehl "Einfügen > Kommentarzeile einfügen".
- Über den Kontextmenübefehl "Kommentarzeile einfügen".
- Durch Eingabe der Zeichenfolge "//" in der Spalte "Name" der Forcetabelle.

Kommentarzeilen einfügen

Zum Einfügen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Forcetabelle und tragen Sie die gewünschten Adressen ein.
2. Fügen Sie eine neue Kommentarzeile ein:
   - Klicken Sie dazu auf das Symbol "Kommentarzeile einfügen" in der Funktionsleiste der Forcetabelle.
   - Oder klicken Sie auf den Menübefehl "Einfügen > Kommentarzeile einfügen".
   - Oder klicken Sie auf den Kontextmenübefehl "Kommentarzeile einfügen".
     Ergebnis: Eine neue Kommentarzeile wird oberhalb der selektierten Zeile in die Forcetabelle eingefügt.
3. Tragen Sie den gewünschten Kommentar in die Kommentarzeile ein. Der eingegebene Kommentar wird in Grün dargestellt.
4. Um alle von Ihnen eingegebenen Kommentare anzuzeigen, doppelklicken Sie in der Projektnavigation unter "Sprachen & Ressourcen" auf "Projekttexte".
5. Falls Sie in mehrsprachigen Projekten arbeiten und Ihren Kommentar in andere Sprachen übersetzen wollen, können Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" die zusätzlich zur Editiersprache gewünschten Projektsprachen einstellen.

Kommentarzeilen erzeugen

Zum Erzeugen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Forcetabelle und tragen Sie die gewünschten Adressen ein.
2. Geben Sie in der Spalte "Name" die Zeichenfolge "//" in einer freien Zeile ein. Dadurch wird diese Zeile zur Kommentarzeile.
   In den restlichen Spalten können keine Kommentarzeilen erzeugt werden.
3. Tragen Sie den gewünschten Kommentar in die Kommentarzeile ein. Der eingegebene Kommentar wird in Grün dargestellt.
4. Um alle von Ihnen eingegebenen Kommentare anzuzeigen, doppelklicken Sie in der Projektnavigation unter "Sprachen & Ressourcen" auf "Projekttexte".
5. Falls Sie in mehrsprachigen Projekten arbeiten und Ihren Kommentar in andere Sprachen übersetzen wollen, können Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" die zusätzlich zur Editiersprache gewünschten Projektsprachen einstellen.

Kommentarzeilen löschen

Zum Löschen von Kommentarzeilen gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle mit vorhandenen Kommentarzeilen.
2. Löschen Sie den gesamten Kommentar mit der einleitenden Zeichenfolge "//", falls Sie diesen nicht mehr benötigen.
3. Löschen Sie alternativ nur die einleitende Zeichenfolge "//", so bleibt der vorhandene Kommentar erhalten und wird in der Spalte "Kommentar" in der Forcetabelle angezeigt.

Hinweis

Löschen von Kommentareilen

Beim Löschen von Kommentarzeilen werden die für diese Kommentare eingestellten Projektsprachen und evtl. vorhandene Übersetzungen ebenfalls gelöscht.

23.5.8 Variablen in der Forcetabelle beobachten

23.5.8.1 Einführung zum Beobachten von Variablen in der Forcetabelle

Einführung

In der Forcetabelle können Sie die Variablen der projektierten Ein- und Ausgabebaugruppen in der CPU beobachten, abhängig vom jeweils ausgewählten Beobachtungsmodus (Seite 8130). Um Variablen zu beobachten zu können, muss eine Online-Verbindung zur CPU bestehen.

Optionen zum Beobachten von Variablen

Folgende Optionen für das Beobachten von Variablen stehen zur Verfügung:

- Alle beobachten
  Dieser Befehl startet das Beobachten aller sichtbaren Variablen in der aktiven Forcetabelle, abhängig vom ausgewählten Beobachtungsmodus:
  - Im Basismodus ist der Beobachtungsmodus "permanent" voreingestellt.
  - Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen.

  Hinweis

  Wird der Beobachtungsmodus im Erweiterten Modus umgestellt und wieder in den Basismodus umgeschaltet, wird der zuvor eingestellte Beobachtungsmodus auch im Basismodus übernommen.

- Sofort beobachten
  Dieser Befehl startet das Beobachten der sichtbaren Variablen in der aktiven Forcetabelle sofort und einmalig.

CPU-spezifische Einschränkungen beim Beobachten von Variablen

Es bestehen folgende CPU-spezifische Unterschiede:

- CPU S7-300/400:
  CPUs aus dieser Familie können lediglich die ersten 30 Zeichen eines Strings beobachten.
- CPU S7-1200:
  CPUs aus dieser Familie können einen String bis zur Gesamtgröße von 254 Zeichen beobachten.

23.5.8.2 Festlegen des Beobachtungsmodus in der Forcetabelle

Einführung

Mit der Auswahl des Beobachtungsmodus legen Sie den Triggerpunkt und die Dauer des Beobachtens von Variablen in der Forcetabelle fest.

Möglicher Beobachtungsmodus (Dauer des Beobachtens)

Folgende Auswahlmöglichkeiten stehen zur Verfügung:

- Permanent
- Zyklusbeginn einmalig
- Zyklusende einmalig
- Zyklusbeginn permanent
- Zyklusende permanent
- Übergang von RUN nach STOP einmalig
- Übergang von RUN nach STOP permanent

Besonderheiten beim Modus "Permanent"

Der Modus "Permanent" wird für das Beobachten von Variablen wie folgt ausgeführt: die Eingänge werden am Ende und die Ausgänge werden am Anfang des Zyklus beobachtet.

Triggerpunkt festlegen

Die Triggerpunkte "Zyklusbeginn", "Zyklusende" und "Übergang von RUN nach STOP" legen fest, zu welchem Zeitpunkt die Variablen aus der CPU gelesen bzw. in der CPU aktualisiert werden sollen.

Das folgende Bild zeigt die Lage der genannten Triggerpunkte:

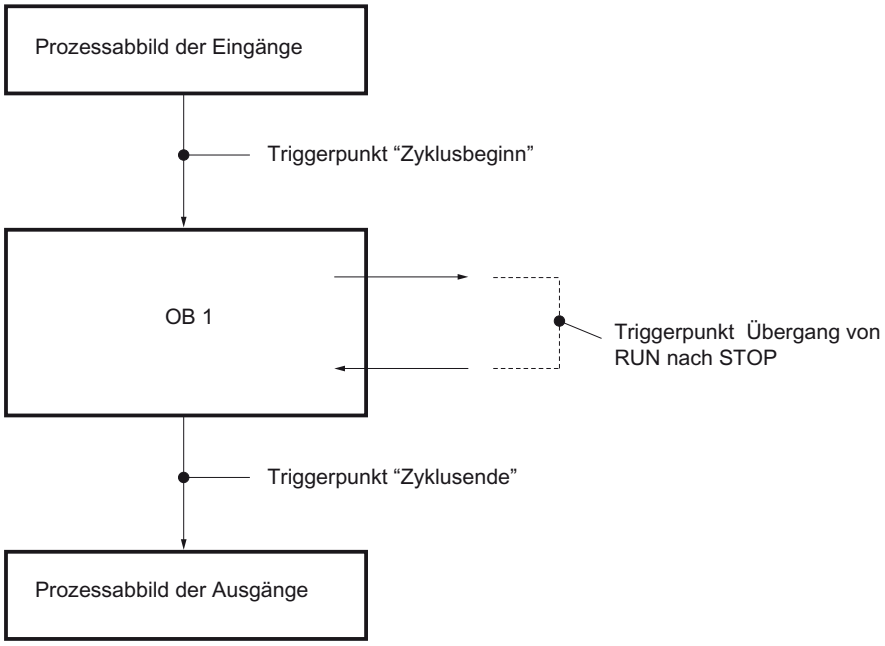

23.5.8.3 Variablen "Alle beobachten"

Einführung

Mit dem Befehl "Alle beobachten" können Sie das Beobachten der sichtbaren Variablen in der aktiven Forcetabelle starten. Im Basismodus der Forcetabelle ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen. In diesem Fall werden die Variablen bezüglich der festgelegten Triggerpunkte beobachtet.

Voraussetzung

Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehensweise

Um den Befehl "Alle beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Forcetabelle ein.
2. Schalten Sie in den Erweiterten Modus um, indem Sie in der Funktionsleiste auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" klicken.
3. Wenn Sie den voreingestellten Beobachtungsmodus einer Variablen ändern wollen, klicken Sie in der Spalte "Beobachten mit Trigger" auf die entsprechende Zelle und wählen Sie aus der Klappliste den gewünschten Beobachtungsmodus aus.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Alle beobachten".

Ergebnis

Die Variablen der aktiven Forcetabelle werden mit dem jeweils eingestellten Beobachtungsmodus beobachtet.

23.5.8.4 Variablen "Sofort beobachten"

Einführung

Mit dem Befehl "Sofort beobachten" wird das Beobachten der Variablen sofort, ohne Bezug auf bestimmte Triggerpunkte gestartet. Die Variablenwerte werden einmalig ausgelesen und in der Forcetabelle angezeigt.

Voraussetzung

Eine Online-Verbindung zur CPU ist vorhanden.

Vorgehensweise

Um den Befehl "Sofort beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Forcetabelle ein.
2. Klicken Sie in der Funktionsleiste auf das Symbol "Sofort beobachten".

Ergebnis

Die Variablen der aktiven Forcetabelle werden sofort und einmalig beobachtet.

23.5.9 Variablen in der Forcetabelle forcen

23.5.9.1 Einführung zum Forcen von Variablen

Einführung

Mit Hilfe der Forcetabelle können Sie einzelne Variablen des Anwenderprogramms mit festen Werten vorbelegen. Diesen Vorgang nennt man Forcen. Es können nur Peripherieeingänge und Peripherieausgänge geforct werden.

Voraussetzung für das Forcen ist, dass eine Online-Verbindung zur CPU besteht und dass die verwendete CPU diese Funktionalität unterstützt.

Wenn Sie eine Forcetabelle im Ordner "Beobachtungs- und Forcetabellen" unterhalb einer CPU öffnen, auf der bereits ein Forceauftrag läuft, werden Sie aufgefordert, zunächst die

Anzeige der geforcten Operanden zu aktualisieren. Ein Forcen und Forcen beenden in der geöffneten Forcetabelle ist erst nach dem Ausführen dieses Befehls möglich.

Einsatzmöglichkeiten

Durch das feste Vorbelegen von Variablen mit definierten Werten, können Sie für Ihr Anwenderprogramm bestimmte Vorgaben setzen und damit die programmierten Funktionen testen. Das Forcen ist im Basismodus und im Erweiterten Modus (Seite 8118) möglich.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen (Seite 8134) bei diesem Vorgang informieren.

> ⚠ **GEFAHR**
>
> **Personen- und Sachschaden verhindern!**
>
> Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:
> - Das Leben oder die Gesundheit von Personen gefährden kann.
> - Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Optionen zum Forcen von Variablen

Folgende Optionen für das Forcen von Variablen stehen zur Verfügung:

- Forcen auf "0"
  Dieser Befehl forct die selektierte Adresse in der CPU auf den Forcewert "0"
- Forcen auf "1"
  Dieser Befehl forct die selektierte Adresse in der CPU auf den Forcewert "1".
- Alle forcen
  Dieser Befehl startet das Forcen der aktivierten Adressen in der aktiven Forcetabelle oder ersetzt unterbrechungsfrei einen bestehenden Forceauftrag.
- Forcen beenden
  Dieser Befehl beendet das Forcen aller Adressen in der aktiven Forcetabelle.

Einschränkungen beim Forcen von Variablen

Beachten Sie bitte die nachfolgend genannten Einschränkungen beim Forcen:

- Das Forcen ist immer abhängig vom Operandenumfang der eingesetzten CPU.
- Es können grundsätzlich nur Peripherieeingänge und Peripherieausgänge geforct werden.
- Wenn die Funktion "Peripherieausgänge freischalten" auf Ihrer CPU aktiv ist, ist ein Forcen nicht möglich! Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Besonderheiten beim Forcen von Variablen

Beachten Sie, dass das Forcen von Variablen Werte in der CPU überschreibt und auch nach Beenden der Online-Verbindung zur CPU weiterbesteht.

- **Forcen beenden**
  Das Beenden der Online-Verbindung alleine stoppt den Vorgang des Forcens nicht! Um das Forcen zu beenden, müssen Sie auf den Befehl "Online > Forcen > Forcen beenden" klicken. Erst dann werden die in der aktiven Forcetabelle sichtbaren Variablen nicht mehr geforct.
- **Beenden des Forcens einzelner Variablen**
  Der Befehl "Online > Forcen > Forcen beenden" gilt immer für alle in der Forcetabelle angezeigten Variablen. Um das Forcen für einzelne Variablen zu beenden, müssen Sie das Optionskästchen für das Forcen dieser Variablen in der Forcetabelle deaktivieren und das Forcen über den Befehl "Online > Forcen > Alle forcen" erneut starten.

23.5.9.2 Sicherheitsvorkehrungen beim Forcen von Variablen

Sicherheitsvorkehrungen zum Forcen von Variablen

Da Sie mit der Funktionalität des Forcens permanent in den Prozess eingreifen, sind die folgenden Hinweise unbedingt zu beachten:

GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

⚠ VORSICHT

Personen- und Sachschäden verhindern!

- Bevor Sie die Funktion "Forcen" starten, sollten Sie sicherstellen, dass niemand gleichzeitig auf derselben CPU diese Funktion ebenfalls ausführt.
- Das Forcen kann nur durch Klicken auf das Symbol "Forcen beenden" oder über den Befehl "Online > Forcen > Forcen beenden" beendet werden. Das Schließen der aktiven Forcetabelle beendet das Forcen **nicht**!
- "Forcen" kann **nicht** rückgängig gemacht werden!
- Informieren Sie sich über die Unterschiede zwischen "Steuern von Variablen" (Seite 8106) und "Forcen von Variablen" (Seite 8132)".
- Falls eine CPU die Funktionalität des "Forcens" nicht unterstützt, so sind alle das Forcen betreffenden Symbole nicht anwählbar.
- Wenn die Funktion "Peripherieausgänge freischalten" auf einer CPU aktiv ist, ist das Forcen auf dieser CPU **nicht** möglich. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

23.5.9.3 Geforcte Operanden aktualisieren

Einführung

Wenn auf einer CPU bereits ein Forceauftrag läuft, müssen Sie nach dem Öffnen der Forcetabelle zunächst sicherstellen, dass die aktuell auf der CPU geforcten Operanden und Werte in der Forcetabelle angezeigt werden.

Mit dem Befehl "Online" > "Forcen" > "Geforcte Operanden aktualisieren" werden alle momentan auf der CPU geforcten Operanden und Werte in der geöffneten Forcetabelle aktualisiert.

Ein "Forcen" und "Forcen beenden" in der geöffneten Forcetabelle ist erst nach dem Ausführen dieses Befehls möglich.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134) informieren.

> ⚠ GEFAHR
>
> **Personen- und Sachschaden verhindern!**
>
> Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:
> - Das Leben oder die Gesundheit von Personen gefährden kann.
> - Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- Eine Online-Verbindung zur CPU ist möglich.
- Auf der verwendeten CPU läuft aktuell ein Forceauftrag.

Vorgehen

Um die geforcten Operanden und Werte zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine Forcetabelle.
2. Stellen Sie eine Online-Verbindung zur CPU her.
3. Bestätigen Sie den nachfolgenden Dialog "Geforcte Operanden aktualisieren" mit "Ja".

Ergebnis

In der geöffneten Forcetabelle werden alle geforcten Operanden mit den entsprechenden Werten aktualisiert. In der ersten Spalte wird ein rotes "F" eingeblendet, das anzeigt, welche Operanden geforct werden.

Damit werden die Schaltflächen "Alle Forcen" und "Forcen beenden" freigeschaltet und Sie können diese Funktionen ausführen.

Hinweis

Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

23.5.9.4 Variablen forcen auf "0"

Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134) informieren.

GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Vorgehen

Um Variablen auf "0" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
3. Wählen Sie den Befehl "Online > Forcen > Forcen auf 0", um die ausgewählte Adresse mit dem vorgegebenen Wert zu forcen.
4. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Die ausgewählte Adresse wird auf "0" geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Das Forcen der ausgewählten Werte wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

Hinweis

Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

23.5.9.5 Variablen forcen auf "1"

Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134) informieren.

⚠ GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden sowie
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Vorgehen

Um Variablen auf "1" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
3. Wählen Sie den Befehl "Online > Forcen > Forcen auf 1", um die ausgewählte Adresse mit dem vorgegebenen Wert zu forcen.
4. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Die ausgewählte Adresse wird auf "1" geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Das Forcen der ausgewählten Werte wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

Hinweis

Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

23.5.9.6 Variablen forcen "Alle forcen"

Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.

Falls bereits ein Forcen läuft, so wird dieses durch den Befehl "Online > Forcen > Alle forcen" unterbrechungsfrei ersetzt. Dabei werden die nicht aktivierten geforcten Adressen nicht mehr geforct.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134) informieren.

⚠ GEFAHR

Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Vorgehen

Um Variablen mit dem Befehl "Online > Forcen > Alle forcen" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschten Adressen und Forcewerte in die Forcetabelle ein.
3. Wählen Sie die zu forcenden Adressen aus, indem Sie die Optionskästchen für das Forcen in der Spalte nach dem "Forcewert" aktivieren.
   Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.
4. Wählen Sie den Befehl "Online > Forcen > Alle forcen", um die ausgewählten Adressen mit den vorgegebenen Werten zu forcen.
5. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Die ausgewählten Adressen werden auf die vorgegebenen Werte geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Das Forcen der ausgewählten Adressen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

Hinweis

Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

23.5.10 Forcen von Variablen beenden

23.5.10.1 Forcen aller Variablen beenden

Einführung

Beachten Sie die nachfolgenden Hinweise, bevor Sie das Forcen von Variablen beenden:

- Das Beenden des Forcens kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8134) informieren.

> ⚠ GEFAHR
>
> **Personen- und Sachschaden verhindern!**
>
> Beachten Sie, dass beim Beenden der Funktion "Forcen" eine falsche Handlung:
> - Das Leben oder die Gesundheit von Personen gefährden kann.
> - Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- In einer Forcetabelle werden Variablen geforct.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist nicht aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Vorgehen

Um das **Forcen aller Variablen** zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Wählen Sie den Befehl "Online > Forcen > Forcen beenden", um das Forcen der angezeigten Adressen zu beenden.
3. Bestätigen Sie den Dialog "Forcen beenden" mit "Ja".

Ergebnis

Das Forcen aller Variablen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen vorgemerkt ist, momentan aber nicht geforct wird.

23.5.10.2 Forcen einzelner Variablen beenden

Einführung

Beachten Sie die nachfolgenden Hinweise, bevor Sie das Forcen von Variablen beenden:

- Das Beenden des Forcens kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 8116) informieren.

> ⚠ **GEFAHR**
>
> **Personen- und Sachschaden verhindern!**
>
> Beachten Sie, dass beim Beenden der Funktion "Forcen" eine falsche Handlung:
> - Das Leben oder die Gesundheit von Personen gefährden kann.
> - Schäden an der Maschine oder der gesamten Anlage verursachen kann.

Voraussetzung

- In einer Forcetabelle werden Variablen geforct.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist nicht aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

Vorgehen

Um das **Forcen einzelner Variablen** zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Deaktivieren Sie das Optionskästchen für die Adressen, die nicht mehr geforct werden sollen.
3. Wählen Sie erneut den Befehl "Online > Forcen".

Ergebnis

Das Forcen der deaktivierten Adressen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen vorgemerkt ist, momentan aber nicht geforct wird.

Maschinen und Anlagen mit ProDiag überwachen 24

24.1 Systemvoraussetzungen für ProDiag

24.1.1 Hinweise zu Lizenzen

Beschreibung

Die Projektierung von Überwachungen ist komplett in STEP 7 Professional integriert. Es kann von jedem STEP 7 Engineering-Platz und unabhängig von der Applikation genutzt werden. Lizenziert wird ausschließlich die Anzahl der projektierten Überwachungen, wobei die ersten 25 Überwachungen kostenlos genutzt werden können.

Die folgende Tabelle zeigt Ihnen, wie viele Lizenzen Sie benötigen:

Anzahl der verwendeten Überwachungen	<= 25	<= 250	<= 500	<= 750	<= 1000	> 1000
Anzahl der benötigten Lizenzen	Keine	1	2	3	4	5

Weitere Informationen zur allgemeinen Bereitstellung von Lizenzen finden Sie unter: "Installation > Systemvoraussetzungen für die Installation > Hinweise zu Lizenzen".

24.1.2 ProDiag-Lizenzen einstellen

Voraussetzung

Sie haben eine S7-1500-CPU angelegt.

Vorgehen

Um die Lizenzen für ProDiag einzustellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation die Gerätekonfiguration der S7-1500-CPU. Die Gerätekonfiguration wird in der Gerätesicht geöffnet.
2. Klicken Sie im Inspektorfenster "Eigenschaften" das untergeordnete Register "Allgemein" an.
3. Klicken Sie auf den Pfeil neben dem Abschnitt "Runtime-Lizenzen". Es werden die verfügbaren Runtime-Lizenzen angezeigt.

4. Klicken Sie den Eintrag "ProDiag" an.
   Im Abschnitt "Überwachungen" werden Ihnen alle verwendeten Überwachungen innerhalb der S7-1500-CPU angezeigt.
5. Wählen Sie, entsprechend der Anzahl der verwendeten Überwachungen, im Abschnitt "Runtime-Lizenzen" die Anzahl der benötigten Lizenzen aus.

Ergebnis

Die benötigten Lizenzen sind eingestellt.

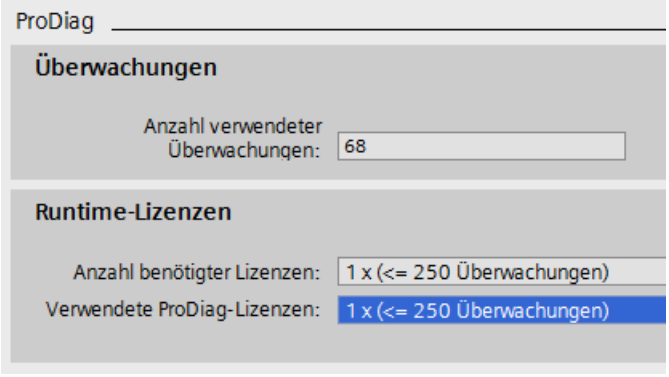

24.1.3 Software- und Hardwarevoraussetzungen

Voraussetzungen für ProDiag

Die folgende Tabelle zeigt die minimalen Software- und Hardwarevoraussetzungen, die für die Verwendung von ProDiag mindestens erfüllt sein müssen:

Software/Hardware	Voraussetzung
TIA Portal	STEP 7 Professional
CPU der Baureihe S7-1500	>= Firmware V2.0

24.2 Technische Daten für ProDiag

24.2.1 Technische Daten für ProDiag

Beschreibung

Die folgende Tabelle zeigt Ihnen die möglichen max. Werte:

Objekt	Anzahl der Objekte
ProDiag-Funktionsbausteine	Es sind max. 100 ProDiag-FBs in einem Projekt verwendbar.
ProDiag-Überwachungen	ProDiag-FB V1.0: Einem ProDiag-FB können max. 250 Überwachungen zugewiesen werden. ProDiag-FB V2.0: Einem ProDiag-FB können max. 1000 Überwachungen zugewiesen werden.

24.3 Grundlagen der Überwachung mit ProDiag

24.3.1 Einführung in ProDiag

Beschreibung

Im TIA Portal haben Sie ab Version V14 die Möglichkeit, Ihre Maschine oder Anlage mithilfe der Funktionalität ProDiag zu überwachen und im Fall einer Störung einzugreifen. Die Überwachungsmeldungen, die Sie für unterschiedliche Störungen anlegen können, liefern Ihnen gezielte Informationen zur Überwachungsart, zum Ort und der Ursache der Störung. Zu den ermittelten Störungen können Sie zusätzlich Hinweise zur Behebung ausgeben. Dadurch können Sie nicht nur Störungen erkennen, sondern auch bereits im Vorfeld eine mögliche Gefahr für eine Störung identifizieren und entsprechend gegensteuern.

Bereits mit wenigen Projektierungsschritten können Sie einfache Überwachungen in Ihr Programm integrieren ohne dabei den Programmcode zu verändern. Dazu können Sie z. B. eine Überwachung erstellen, indem Sie einen booleschen Operanden auf seinen Signalzustand hin überwachen. Sobald der Operand den eingestellten Signalzustand liefert, wird eine ProDiag-Überwachungsmeldung anhand der projektierten ProDiag-Überwachungseinstellungen ausgegeben. Die Projektierung der Überwachungen ist unabhängig von den Programmiersprachen des TIA Portals, da nur einzelne Operanden überwacht werden und Sie keine weiteren Programmierabschnitte benötigen.

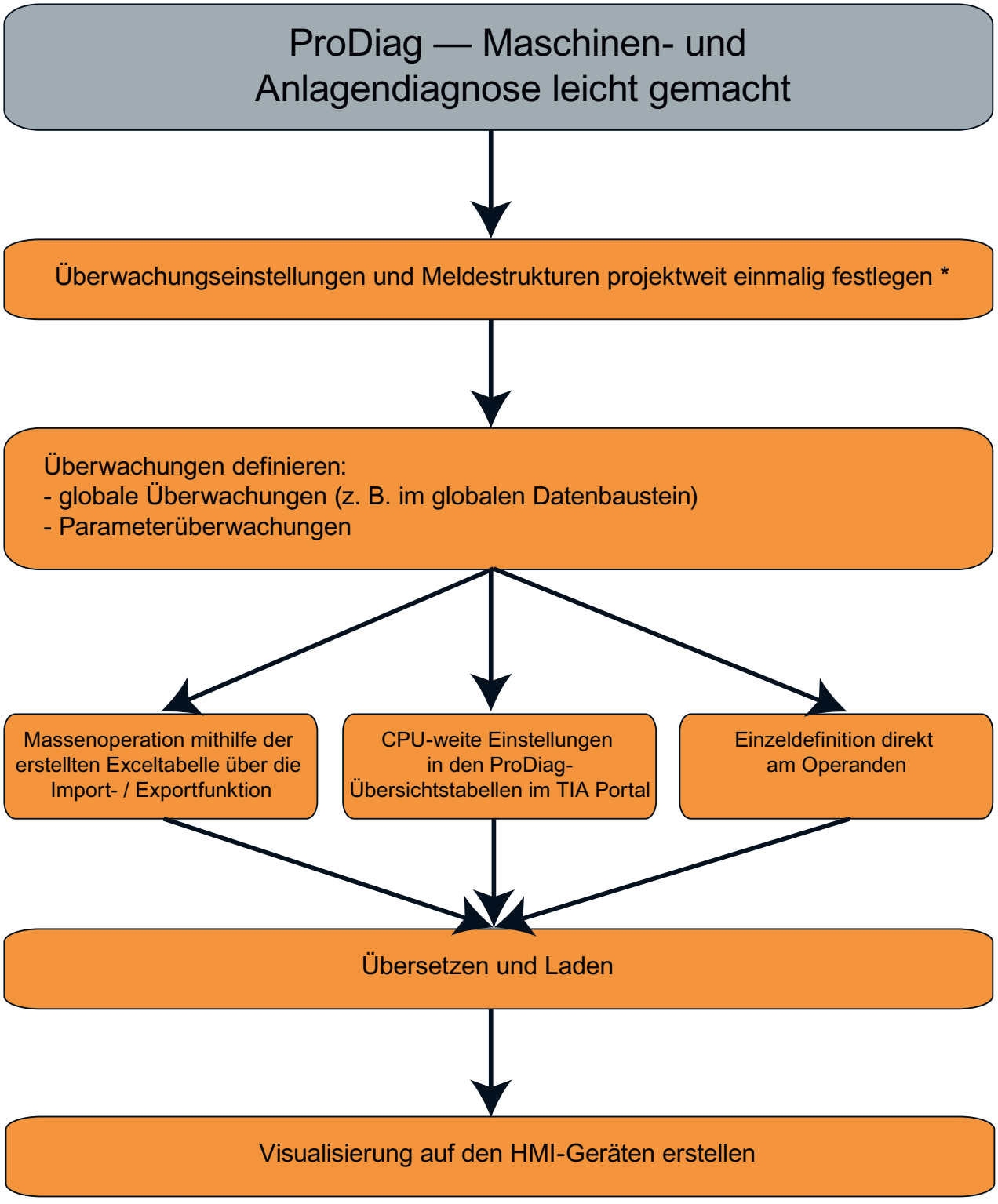

* Die Überwachungseinstellungen sind erst verfügbar, wenn Sie einen GRAPH-FB oder eine Überwachung angelegt haben.

Ein Einführungsbeispiel, wie die Erstellung von Überwachungen mithilfe von ProDiag aussehen kann, finden Sie hier: Beispiel für die Erstellung einer Überwachung anhand einer Operandenüberwachung (Seite 8151)

Vorteile

Das Überwachen Ihrer Maschine oder Anlage mit ProDiag bietet Ihnen viele Vorteile:

- Ihr Programmcode wird durch das Anlegen der Überwachungen nicht verändert und die ProDiag-Funktionsbausteine automatisch im ProDiag-Organisationsbaustein aufgerufen, solange Sie diese nicht selbst in Ihrem Programmcode aufrufen möchten.
- Mithilfe der ProDiag-Funktionsbausteine können Sie Einheiten erstellen und darin alle Überwachungen einer Einheit, z. B. eines Motors, nach technologischen Gesichtspunkten gruppieren.
- Sie können Stillstandszeiten und Produktionsausfälle nach dem Auftreten einer Störung, bzw. bei vorhersehbaren Ereignissen, durch frühe Warnungen oder Hinweise, reduzieren.
- Mit gezielten Hinweisen zur Ursache einer Störung für den Maschinenbediener vor Ort werden die Behebung der Störung und mögliche Wartungsarbeiten erheblich erleichtert.
- Unabhängig davon, welche S7-1500-CPU Sie verwenden, ist die Projektierung der Überwachungen einfach und leicht verständlich.
- Sowohl F-Bausteine als auch know-how-geschützte Funktionsbausteine können Sie ebenfalls problemlos mit ProDiag überwachen, da sie nicht verändert und damit nicht neu übersetzt werden müssen.
- Meldetexte werden automatisch von im Projekt bestehenden Texten abgeleitet, z. B.:
  - Kategorie
  - CPU-Name
  - FB-Name
  - Instanz-Name, etc.
- Änderungen in der CPU-Projektierung werden vom HMI-Gerät automatisch übernommen. Die Meldetexte werden direkt in der CPU verwaltet.
- Verhältnismäßig geringe Zykluszeitbeeinflussung.
- Das Änderungsmanagement der Überwachungen wird komplett im TIA Portal verwaltet, d. h., der Anwender hat keinen Aufwand bzgl. Konsistenzsicherung. Damit wird die Fehlerwahrscheinlichkeit von falschen Meldetexten weitestgehend ausgeschlossen.

24.3.2 Beispiel für die Erstellung einer Überwachung anhand einer Operandenüberwachung

Beschreibung

Das nachfolgende Beispiel gibt Ihnen einen kurzen Überblick über die Erstellung einer Überwachung anhand einer Operandenüberwachung. Detaillierte Informationen zu den einzelnen Schritten finden Sie in den nachfolgenden Kapiteln.

1. Legen Sie eine ProDiag-Überwachungsgruppe an. Fügen Sie dafür einen neuen Funktionsbaustein vom Typ PRODIAG mit dem Namen "Station_1" ein:

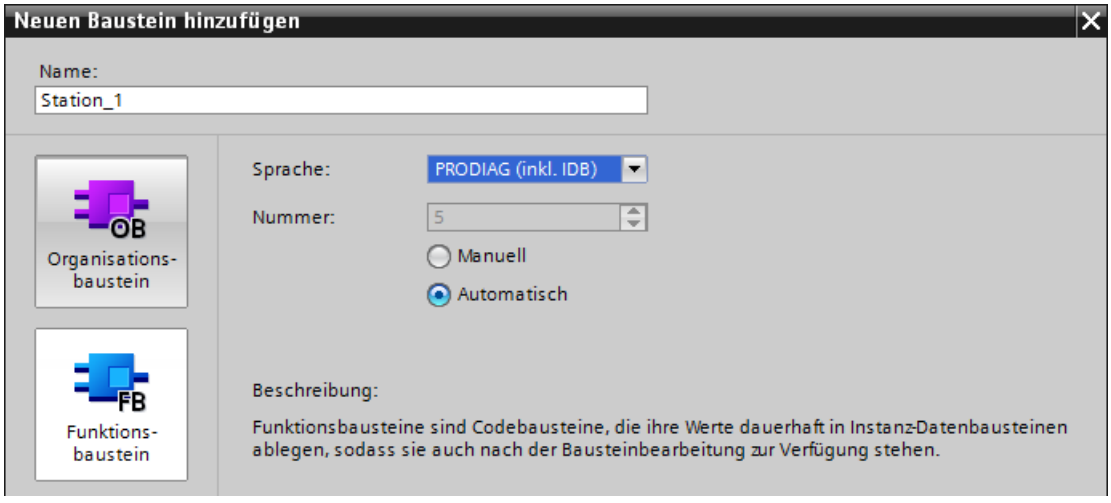

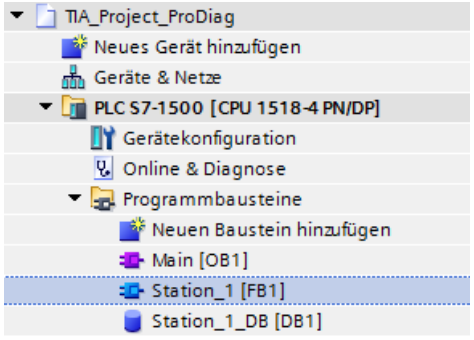

2. Rufen Sie die ProDiag-Überwachungseinstellungen in der Projektnavigation im Ordner "Gemeinsame Daten" auf:

3. Legen Sie die ProDiag-Überwachungseinstellungen fest:

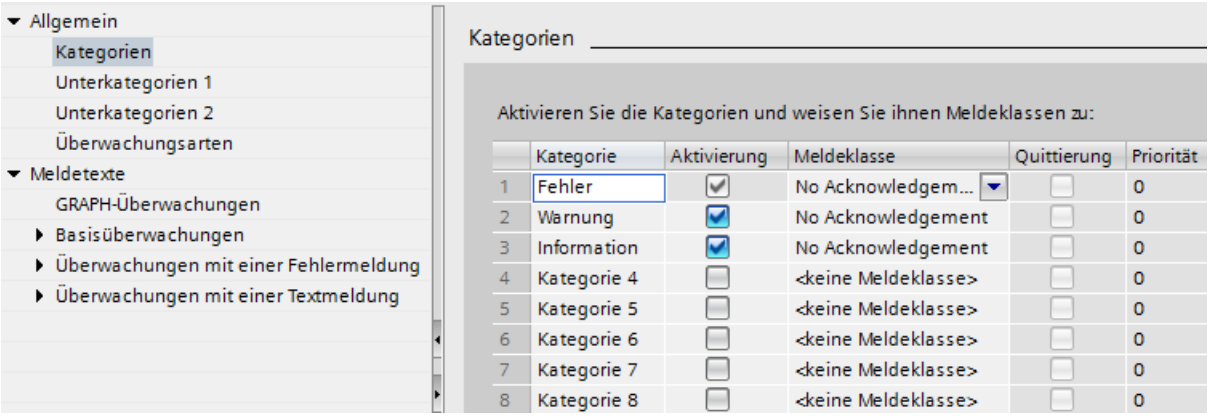

4. Deklarieren Sie in der Standard-Variablentabelle eine boolesche Variable und fügen Sie eine Überwachung hinzu:

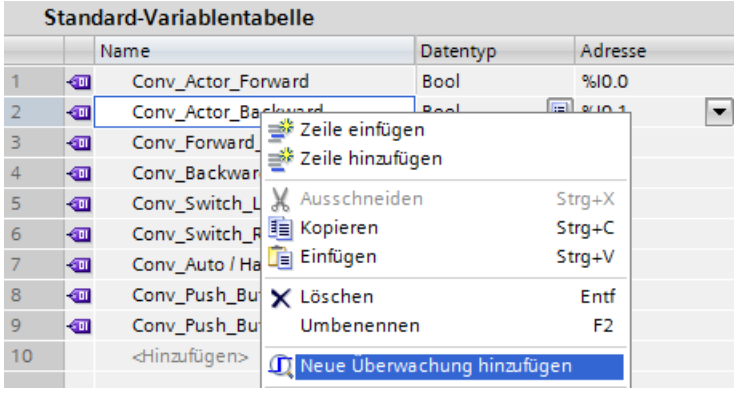

5. Definieren Sie im Inspektorfenster "Eigenschaften > Überwachungen" die Eigenschaften der Operandenüberwachung:

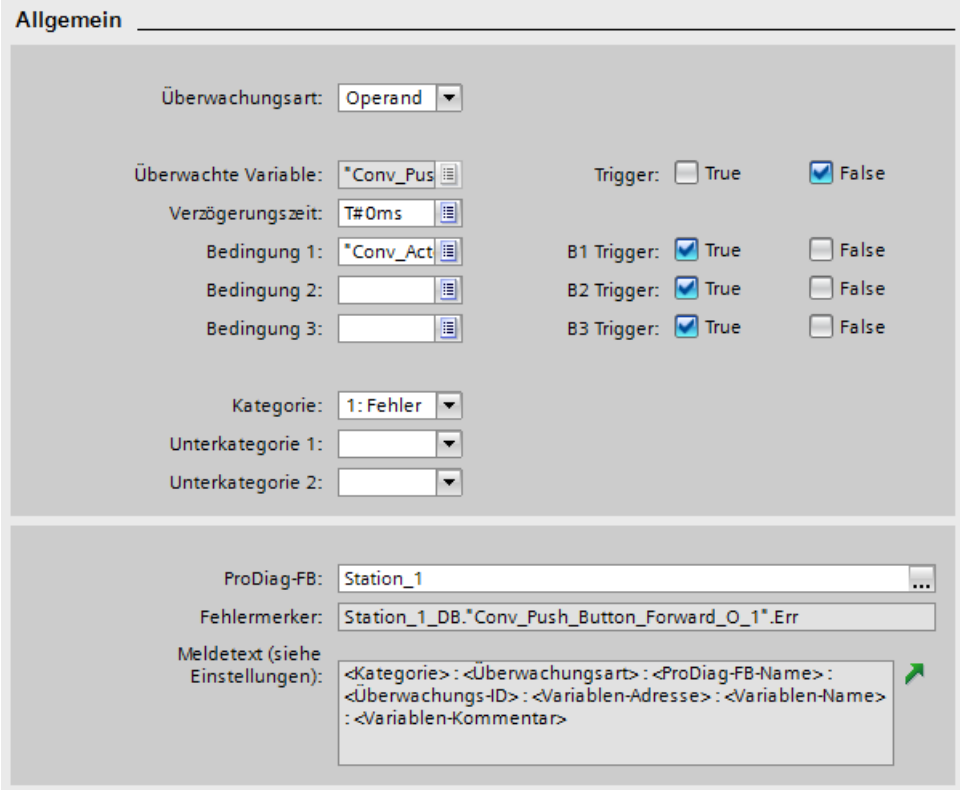

Sie können entweder eine globale Variablenüberwachung oder eine lokale Parameterüberwachung am Funktionsbaustein anlegen. Sie können dazu entweder einen Operanden verwenden, der bereits im Programmcode enthalten ist oder Sie legen einen neuen Operanden an und rufen diesen anschließend im Programmcode auf. Wenn Sie eine Parameterüberwachung anlegen, dann müssen Sie den Funktionsbaustein im Programmcode aufrufen.

6. Übersetzen und laden Sie das Projekt.
   Der ProDiag-Funktionsbaustein "Station_1" wird automatisch im ProDiag-Organisationsbaustein aufgerufen.

Siehe auch

ProDiag-Funktionsbausteine anlegen (Seite 8190)

ProDiag-Überwachungseinstellungen festlegen (Seite 8222)

ProDiag-Überwachungen anlegen (Seite 8245)

24.3.3 Überwachungsarten

24.3.3.1 Übersicht über die Überwachungsarten

Beschreibung

Mit der Funktionalität ProDiag können Sie für die folgenden Überwachungsarten unterschiedliche Meldetexte definieren:

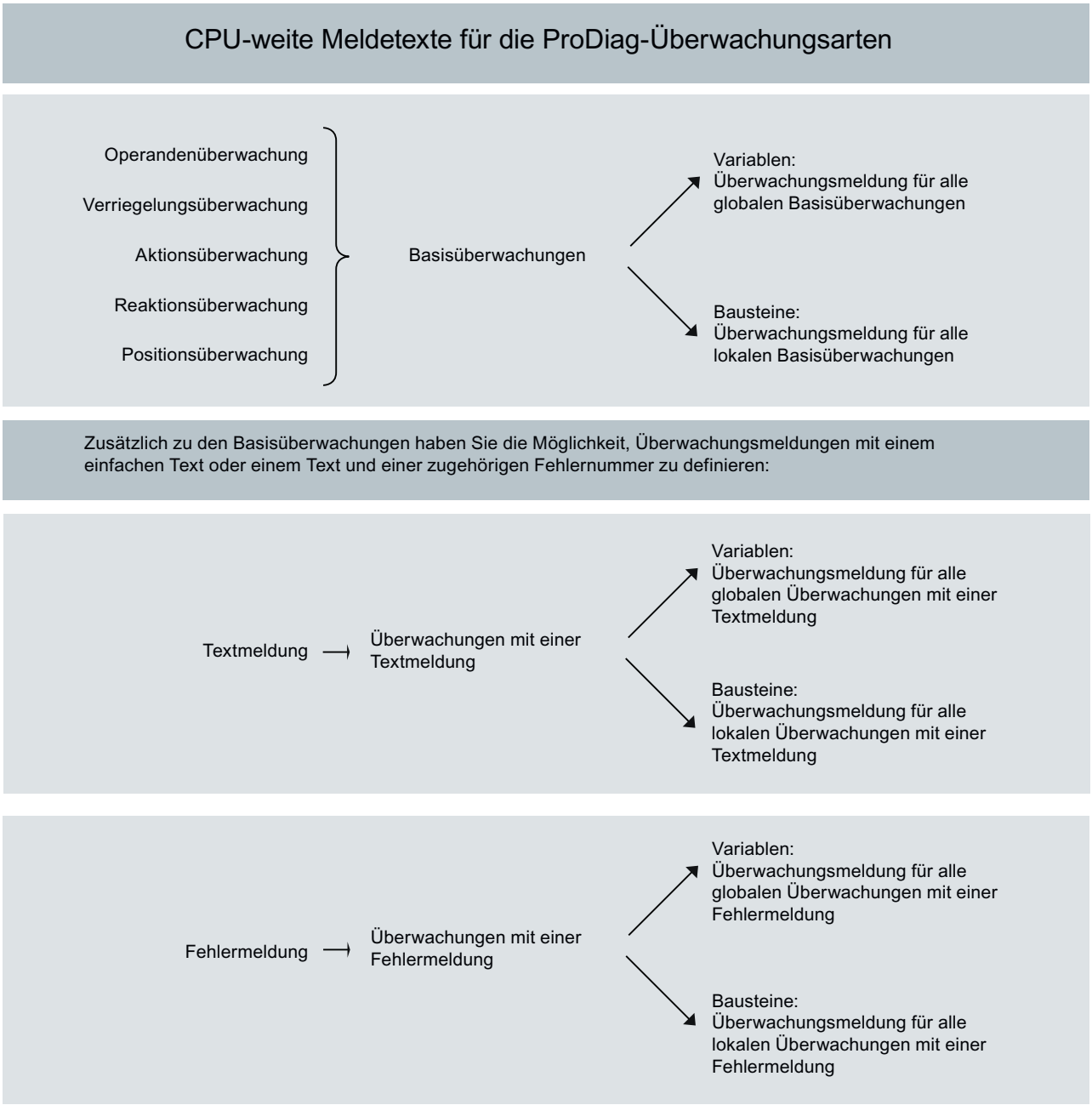

Weitere Informationen zur Definition der Meldetexte finden Sie hier: Zentrale ProDiag-Meldetexte festlegen (Seite 8233)

Hinweis

Detaillierte Informationen zu den einzelnen Überwachungsarten finden Sie hier:

Operandenüberwachung (Seite 8155)

Verriegelungsüberwachung (Seite 8157)

Aktionsüberwachung (Seite 8159)

Reaktionsüberwachung (Seite 8161)

Positionsüberwachung (Seite 8163)

Textmeldung (Seite 8165)

Fehlermeldung (Seite 8167)

Hinweis

Beispiele zu den einzelnen Überwachungsarten finden Sie hier:

Beispiel für eine Operandenüberwachung (Seite 8170)

Beispiel für eine Verriegelungsüberwachung (Seite 8172)

Beispiel für eine Aktionsüberwachung (Seite 8176)

Beispiel für eine Reaktionsüberwachung (Seite 8179)

Beispiel für eine Positionsüberwachung (Seite 8182)

Beispiel für eine Textmeldung (Seite 8185)

Beispiel für eine Fehlermeldung (Seite 8187)

24.3.3.2 Operandenüberwachung

Beschreibung

Die Operandenüberwachung überwacht den Signalzustand (TRUE oder FALSE) eines booleschen Operanden und wird hauptsächlich zur Überwachung sogenannter "Statischer Signale", wie z. B. Druck, Temperatur, Signal von der Anlage, etc., genutzt.

Mithilfe der Operandenüberwachung können Sie auch zwei boolesche Operanden in Abhängigkeit zueinander überwachen.

Das Auslösen einer Operandenüberwachung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe

des Objekts "ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Operandenüberwachung:

Eigenschaften	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Verzögerungszeit	optional	Zeitwert, z. B. T#2s oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, während der überwachte Signalzustand des Operanden ansteht. Wird die Zeit, z. B. eine Prellzeit, überschritten, dann liegt eine Störung vor und eine Überwachungsmeldung wird ausgegeben.
Bedingung 1	optional	TRUE oder FALSE	Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	

Eigenschaften	Eintrag	Zustand des Operanden	Beschreibung
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Operandenüberwachung (Seite 8170)

24.3.3.3 Verriegelungsüberwachung

Beschreibung

Die Verriegelungsüberwachung überwacht, ob nach der Ansteuerung der Aktion (Auslöser) und nach dem Ablauf der Verzögerungszeit (optional), die Aktion ausgeführt wird und, ob die Verriegelungsbedingungen erfüllt sind. Wenn die Verriegelungsbedingungen nicht erfüllt sind, dann liegt eine Störung vor und die Aktion wird nicht ausgeführt.

Die Verriegelungsüberwachung wird häufig zur Überwachungen von Ausgängen, die einen Motor ansteuern, verwendet.

Das Auslösen einer Verriegelungsüberwachung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe

des Objekts "ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Verriegelungsüberwachung:

Eigenschaften	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Verzögerungszeit	optional	Zeitwert, z. B. T#2s oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, während der überwachte Signalzustand des Operanden ansteht. Wird die Zeit, z. B. eine Prellzeit, überschritten, dann liegt eine Störung vor und eine Überwachungsmeldung wird ausgegeben.
Auslöser (Bedingung 1)	zwingend	TRUE oder FALSE	Die Bedingung 1 muss vorhanden sein. Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	

Eigenschaften	Eintrag	Zustand des Operanden	Beschreibung
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Verriegelungsüberwachung (Seite 8172)

Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

24.3.3.4 Aktionsüberwachung

Beschreibung

Die Aktionsüberwachung überwacht, ob das Produktionsstück, das sich z. B. auf einem Fließband befindet, nach der Maschinenbedienung die Startlage innerhalb der Anlaufzeit verlassen hat. Das ist z. B. dann nicht der Fall, wenn das Produktionsstück in seiner Bewegung blockiert ist. Die Verwendung der Aktionsüberwachung macht vor allem bei langsamen Abläufen, wie z. B. dem Befüllen eines Silos, Sinn.

Das Auslösen einer Aktionsüberwachung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe des Objekts

"ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Aktionsüberwachung:

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Anlaufzeit	zwingend	Zeitwert, min. T#150ms oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, bis die Startlage verlassen sein muss. Die Anlaufzeit muss größer als die Zykluszeit sein.
Aktion (Bedingung 1)	zwingend	TRUE oder FALSE	Die Bedingung 1 muss vorhanden sein. Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet sein. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Aktionsüberwachung (Seite 8176)

Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

24.3.3.5 Reaktionsüberwachung

Beschreibung

Die Reaktionsüberwachung überwacht, ob das Produktionsstück, das sich z. B. auf einem Fließband befindet, nach einer Maschinenbedienung die gewünschte Endlage innerhalb der Reaktionszeit erreicht hat. Das kann z. B. nicht der Fall sein, wenn das Produktionsstück herunter gefallen ist oder sich verklemmt hat. Die Verwendung der Reaktionsüberwachung macht vor allem bei schnellen Abläufen Sinn.

Das Auslösen einer Reaktionsüberwachung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe des Objekts "ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Reaktionsüberwachung:

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Reaktionszeit	zwingend	Zeitwert, z. B. T#5000ms oder eine Variable vom Datentyp TIME	Maximale Zeit, die vergehen darf, bis die Endlage erreicht sein muss. Wird die Zeit überschritten, dann liegt eine Störung vor und eine Überwachungsmeldung wird ausgegeben.
Aktion (Bedingung 1)	zwingend	TRUE oder FALSE	Die Bedingung 1 muss vorhanden sein. Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Reaktionsüberwachung (Seite 8179)

Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

24.3.3.6 Positionsüberwachung

Beschreibung

Die Positionsüberwachung überwacht, ob die aktuell erreichte Endlage (Start oder Ziel) des Produktionsstücks nach dem Ablauf der Verzögerungszeit stabil eingehalten wird. Die Endlage darf ohne Ansteuerung nicht wieder verlassen werden. Das ist z. B. dann der Fall, wenn das Produktionsstück von einem Gabelstabler vom Förderband gestoßen wird.

Das Auslösen einer Positionsüberwachung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe des Objekts "ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Positionsüberwachung:

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Verzögerungszeit	optional	Zeitwert, z. B. T#2s oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, während der überwachte Signalzustand des Operanden ansteht. Wird die Zeit, z. B. eine Prellzeit überschritten, dann liegt eine Störung vor und eine Überwachungsmeldung wird ausgegeben.
Aktion (Bedingung 1)	zwingend	TRUE oder FALSE	Die Bedingung 1 muss vorhanden sein. Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit angepasst werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Positionsüberwachung (Seite 8182)

Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

24.3.3.7 Textmeldung

Beschreibung

Mithilfe dieser zusätzlichen Überwachungsart haben Sie die Möglichkeit, nur einen Text, den Sie im spezifischen Textfeld definieren, auszugeben. Das Auslösen einer Textmeldung setzt nicht das Sammelfehlerbit "All" der Zustandsvariable "State" im ProDiag-Instanz-Datenbaustein.

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Textmeldung:

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Verzögerungszeit	optional	Zeitwert, z. B. T#2s oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, während der überwachte Signalzustand des Operanden ansteht. Wird die Zeit, z. B. eine Prellzeit, überschritten, dann liegt eine Störung vor und eine Textmeldung wird ausgegeben.
Bedingung 1	optional	TRUE oder FALSE	Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Textmeldung wurde ausgelöst • FALSE = Keine Textmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

Siehe auch

Beispiel für eine Textmeldung (Seite 8185)

24.3.3.8 Fehlermeldung

Beschreibung

Die Überwachungsart "Fehlermeldung" gibt eine einfache Textmeldung plus einer Fehlernummer aus. Mithilfe dieser zusätzlichen Überwachungsart haben Sie die Möglichkeit, nur einen Text, den Sie im spezifischen Textfeld definieren, und die Fehlernummer auszugeben.

Das Auslösen einer Fehlermeldung setzt den Fehlermerker im zugewiesenen ProDiag-Funktionsbaustein. Dadurch können Sie die Störung auf einem HMI-Gerät mithilfe des Objekts "ProDiag-Übersicht" visualisieren und in Ihrem Programm abfragen, um z. B. die Maschine bei bestimmten Störungen abzuschalten.

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Projektierung

Die folgende Tabelle zeigt die Projektierung einer Fehlermeldung:

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Überwachte(r) Variable / Parameter *	zwingend	TRUE oder FALSE	Operand, der überwacht werden soll
Verzögerungszeit	optional	Zeitwert, z. B. T#2s oder eine Variable vom Datentyp TIME	Zeit, die vergehen darf, während der überwachte Signalzustand des Operanden ansteht. Wird die Zeit, z. B. eine Prellzeit, überschritten, dann liegt eine Störung vor und eine Fehlermeldung wird ausgegeben.
Bedingung 1	optional	TRUE oder FALSE	Wenn Sie mehrere Bedingungen verwenden, dann werden sie automatisch mit einer UND-Verknüpfung verknüpft.
Bedingung 2	optional		
Bedingung 3	optional		

Einstellungen	Eintrag	Zustand des Operanden	Beschreibung
Kategorie	zwingend	-	Sie können bis zu 8 Kategorien definieren. Die ersten 3 Kategorien sind per Voreinstellung vorgegeben. Sie können sie aber jederzeit verändert werden: 1. Fehler 2. Warnung 3. Information Welche Kategorien zur Verfügung stehen sollen, können Sie in den ProDiag-Überwachungseinstellungen festlegen. Weitere Informationen finden Sie hier: Überblick über die Kategorien (Seite 8224)
Unterkategorie 1	optional	-	Sie können pro Unterkategorie bis zu 16 Einträge definieren. Weitere Informationen finden Sie hier: Überblick über die Unterkategorien (Seite 8227)
Unterkategorie 2	optional	-	
ProDiag-FB (Nur bei globalen Überwachungen)	zwingend	-	Jede Überwachung muss eindeutig einem ProDiag-Funktionsbaustein zugeordnet werden. Weitere Informationen finden Sie hier: Überblick über die ProDiag-Funktionsbausteine (Seite 8190)
Fehlermerker (Nur bei globalen Überwachungen)	Wird vom System eingetragen.	-	Es zeigt an, ob eine Störung aufgetreten ist: • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Der Fehlermerker ist erreichbar aus HMI und sichtbar in HMI.
Meldetext (siehe Einstellungen)	Wird vom System eingetragen.	-	Den Inhalt des Meldetextes legen Sie in den ProDiag-Überwachungseinstellungen fest. Weitere Informationen finden Sie hier: Aufbau des Meldetexteditors (Seite 8233)
* Je nachdem, ob Sie eine lokale oder eine globale Überwachung anlegen, wird im Register "Eigenschaften" entweder eine überwachte Variable (global) oder ein überwachter Parameter (lokal) angezeigt.			

24.3.4 Beispiele für die einzelnen Überwachungsarten

24.3.4.1 Einführung

Beschreibung

Anhand des folgenden Beispiels werden Ihnen alle Überwachungsarten erklärt, indem das Produktionsstück schrittweise auf dem Fließband von links nach rechts bewegt wird. Während des Ablaufs können verschieden Störungen auftreten, wie z. B. das Produktionsstück verlässt seine Startlage nicht oder das Produktionsstück erreicht nicht rechtzeitig seine Endlage. Um den gesamten Ablauf zu überwachen und alle Störungen zu identifizieren, können Sie verschiedene Überwachungen mithilfe der Funktionalität ProDiag realisieren.

Das folgende Bild dient als Vorlage für alle nachfolgenden Beispiele:

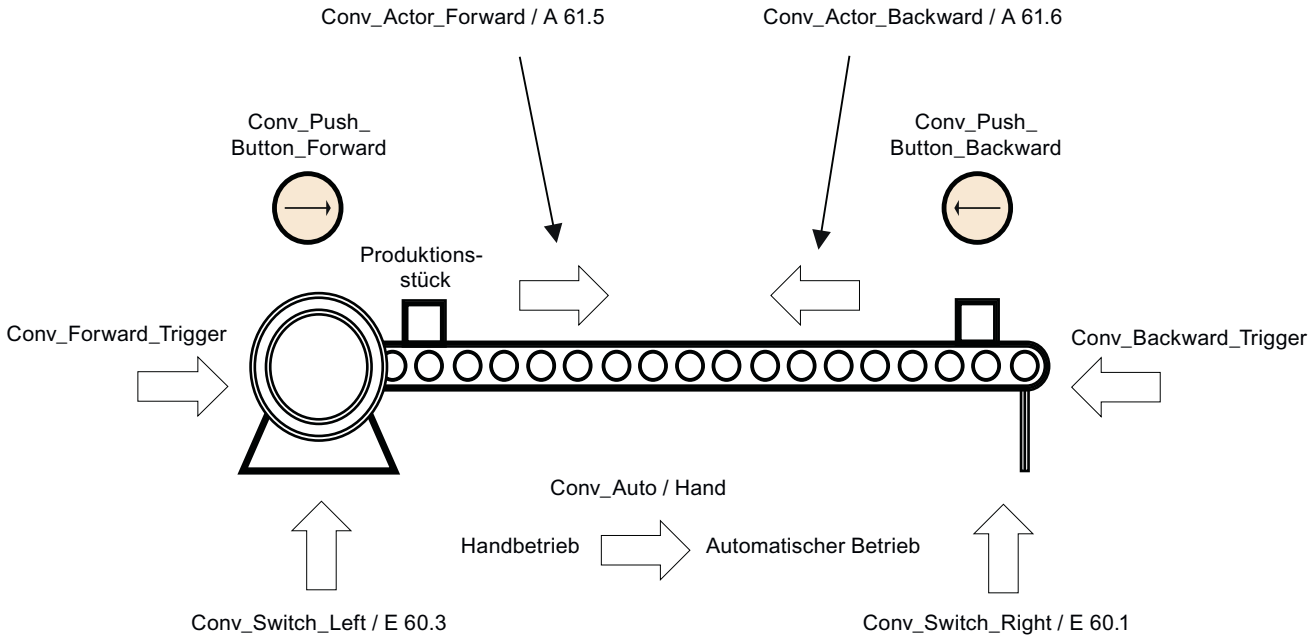

Die Operanden, die in den jeweiligen Beispielen benötigt werden, sind grau hinterlegt.

In den nachfolgenden Kapiteln finden Sie Beispiele für alle möglichen Überwachungsarten:

- Beispiel für eine Operandenüberwachung (Seite 8170)
- Beispiel für eine Verriegelungsüberwachung (Seite 8172)
- Beispiel für eine Aktionsüberwachung (Seite 8176)
- Beispiel für eine Reaktionsüberwachung (Seite 8179)
- Beispiel für eine Positionsüberwachung (Seite 8182)

24.3.4.2 Beispiel für eine Operandenüberwachung

Voraussetzung

Sie haben eine Überwachung für den Operanden "Conv_Switch_Left" angelegt und als Überwachungsart "Operand" ausgewählt.

Beschreibung

Das folgende Bild zeigt Ihnen ein Fließband im Ruhezustand, auf dem sich das Produktionsstück befindet. Das Produktionsstück befindet sich links in der Startlage und wird aktuell nicht angesteuert. Die Pfeile visualisieren die Bewegung des Produktionsstücks von links nach rechts und von rechts nach links, sobald die Produktion anläuft:

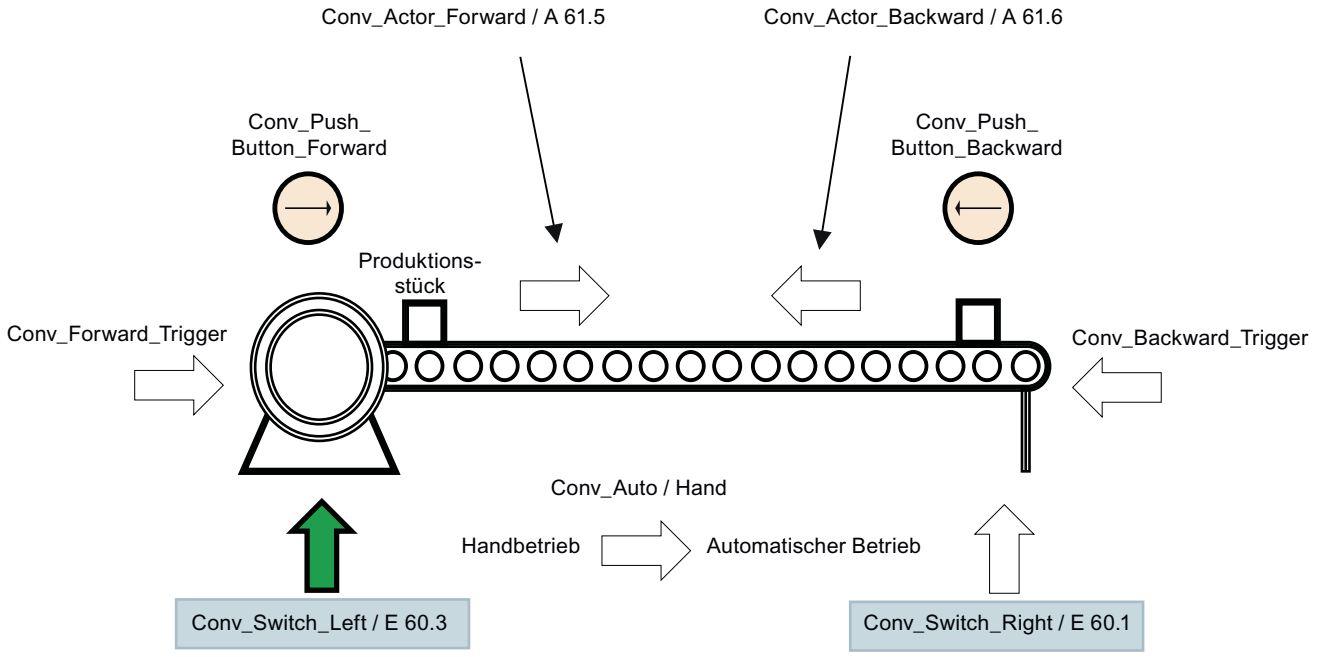

Die für diese Überwachung benötigten Operanden sind grau hinterlegt.

Operand	Erklärung
Conv_Actor_Forward Conv_Actor_Backward	Das sind die Steuersignale (Ausgang des Motors) aus dem Anwenderprogramm für die Bewegungen nach links und rechts.
Conv_Forward_Trigger Conv_Backward_Trigger	Das sind die Steuersignale aus dem Anwenderprogramm (z. B. aus einem GRAPH-Funktionsbaustein) zum Anstoßen der Bewegungen nach links und rechts.
Conv_Switch_Left Conv_Switch_Right	Das ist die Start- und Ziellage des Produktionsstücks.
Conv_Auto / Hand	Das ist der Schalter zur Anwahl der Betriebsart.
Conv_Push_Button_Forward Conv_Push_Button_Backward	Das sind die jeweiligen Handtaster für die Vorwärts- und die Rückwärtsbewegung.

Vorgehen

Geben Sie die folgenden Werte in die Eigenschaften der Operandenüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Operand	-
Überwachte Variable	Conv_Switch_Left	TRUE Sobald der Operand in den Signalzustand TRUE wechselt und alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Störung muss sofort erkannt und gemeldet werden. Daher ist als Verzögerungszeit 0ms angegeben.
Bedingung 1	Conv_Switch_Right	TRUE Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	-	Die Bedingungen werden nicht benötigt.
Bedingung 3	-	
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Conv_Switch_Left_O_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	Beide Operanden haben den Signalzustand TRUE	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Ergebnis

Der überwachte Operand "Conv_Switch_Left" und die Bedingung 1 "Conv_Switch_Right" werden beide auf den Signalzustand TRUE überwacht. Das bedeutet, solange nur einer der beiden vom Signalzustand FALSE auf TRUE wechselt, liegt keine Störung vor. Sobald aber beide auf den Signalzustand TRUE wechseln, liegt eine Störung vor und eine Überwachungsmeldung wird ausgegeben.

Je nachdem, wie Sie die Einstellungen für eine Basisüberwachung in den ProDiag-Überwachungseinstellungen festgelegt haben, kann die Überwachungsmeldung folgendermaßen aussehen:

Überwachungsmeldung für einen Operandenfehler

```
Fehler : Operand : CPU_50 : Default_SupervisionFB : E 60.3 : Conv_Switch_Left : Beide
Operanden haben den Signalzustand TRUE
```

Siehe auch

Operandenüberwachung (Seite 8155)

24.3.4.3 Beispiel für eine Verriegelungsüberwachung

Voraussetzung

Sie haben zwei Überwachungen für den Operanden "Conv_Actor_Forward" angelegt und als Überwachungsart "Verriegelung" ausgewählt:

- Überwachung_ID_1 (Default_SupervisionFB)
- Überwachung_ID_2 (Default_SupervisionFB)

Auf diese Weise können Sie ODER-Verknüpfungen realisieren. Sie können z. B. bei <Überwachung_ID_1> als Bedingung "Automatikbertrieb" und bei <Überwachung_ID2> als Bedingung "Handbetrieb" definieren.

Hinweis

Verwendungsmöglichkeiten

Achten Sie darauf, dass Sie die globale Variable "Conv_Actor_Forward" nicht als Eingang deklarieren.

Weitere Informationen zu den Verwendungsmöglichkeiten finden Sie hier: Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

Beschreibung

Das folgende Bild zeigt Ihnen ein Fließband im Ruhezustand, auf dem sich das Produktionsstück befindet. Das Produktionsstück befindet sich links in der Startlage und wird aktuell nicht angesteuert. Die Pfeile visualisieren die Bewegung des Produktionsstücks von links nach rechts und von rechts nach links, sobald die Produktion anläuft:

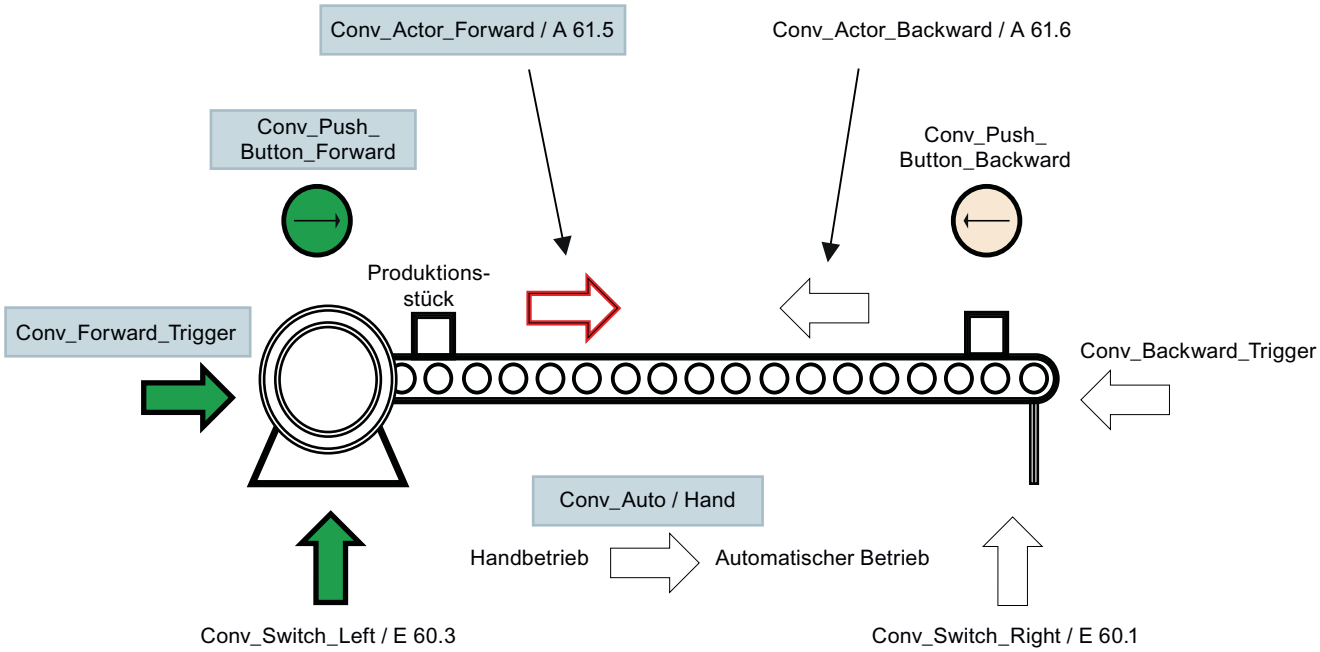

Die für diese Überwachung benötigten Operanden sind grau hinterlegt.

Operand	Erklärung
Conv_Actor_Forward Conv_Actor_Backward	Das sind die Steuersignale (Ausgang des Motors) aus dem Anwenderprogramm für die Bewegungen nach links und rechts.
Conv_Forward_Trigger Conv_Backward_Trigger	Das sind die Steuersignale aus dem Anwenderprogramm (z. B. aus einem GRAPH-Funktionsbaustein) zum Anstoßen der Bewegungen nach links und rechts.
Conv_Switch_Left Conv_Switch_Right	Das ist die Start- und Ziellage des Produktionsstücks.
Conv_Auto / Hand	Das ist der Schalter zur Anwahl der Betriebsart.
Conv_Push_Button_Forward Conv_Push_Button_Backward	Das sind die jeweiligen Handtaster für die Vorwärts- und die Rückwärtsbewegung.

Vorgehen für den Automatischen Betrieb

Geben Sie die folgenden Werte in die Eigenschaften der Verriegelungsüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Verriegelung	-
Überwachte Variable	Conv_Actor_Forward	FALSE Sobald der Operand in den Signalzustand FALSE wechselt oder auch den Signalzustand TRUE beibehält und alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Störung muss sofort erkannt und gemeldet werden. Daher ist als Verzögerungszeit 0ms angegeben.
Auslöser (Bedingung 1)	Conv_Forward_Trigger	TRUE Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	Conv_Auto / Manu	TRUE = Automatischer Betrieb
Bedingung 3	-	Die Bedingung wird nicht benötigt.
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Conv_Actor_Forward_I_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	Automatischer Betrieb	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Vorgehen für den Handbetrieb

Geben Sie die folgenden Werte in die Eigenschaften der Operandenüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Verriegelung	-
Überwachter Operand	Conv_Actor_Forward	FALSE Sobald der Operand in den Signalzustand FALSE wechselt und alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Störung muss sofort erkannt und gemeldet werden. Daher ist als Verzögerungszeit 0ms angegeben.
Auslöser (Bedingung 1)	Conv_Push_Button_Forward	TRUE Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	Conv_Auto / Manu	FALSE = Handbetrieb
Bedingung 3	-	Die Bedingung wird nicht benötigt.
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlerbit	Default_SupervisionDB.Conv_Actor_Forward_I_1.Err	Sobald eine Störung vorliegt, wird dieses Fehlerbit gesetzt.
Spezifisches Textfeld	Handbetrieb	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Ergebnis

Die Bewegung des Produktionsstücks von links nach rechts wurde entweder vom Programm mithilfe des Operanden "Conv_Forward_Trigger" oder per Handtaster mithilfe des Operanden "Conv_Push_Button_Forward" angesteuert. Das Produktionsstück hat die Startlage links jedoch nicht verlassen und der Operand "Conv_Actor_Forward" wechselt dadurch nicht in den Signalzustand TRUE. Das kann z. B. auch mitten in der Bewegung vorkommen. Damit sind alle Voraussetzungen für das Auslösen der Überwachungsmeldung gegeben.

Je nachdem, wie Sie die Einstellungen für eine Basisüberwachung in den ProDiag-Überwachungseinstellungen festgelegt haben, können die Überwachungsmeldungen folgendermaßen aussehen:

Überwachungsmeldung für einen Verriegelungsfehler im Automatischen Betrieb

```
Fehler : Interlock : CPU_50 : Default_SupervisionFB : E 61.5 :
Conv_Actor_Forward : Automatischer Betrieb
```

Überwachungsmeldung für einen Verriegelungsfehler im Handbetrieb

```
Fehler : Interlock : CPU_50 : Default_SupervisionFB : E 61.5 : Conv_Actor_Forward :
Handbetrieb
```

Siehe auch

Verriegelungsüberwachung (Seite 8157)

24.3.4.4 Beispiel für eine Aktionsüberwachung

Voraussetzung

Sie haben eine Überwachung für den Operanden "Conv_Switch_Left" angelegt und als Überwachungsart "Aktion" ausgewählt.

Hinweis

Verwendungsmöglichkeiten

Achten Sie darauf, dass Sie die globale Variable "Conv_Switch_Left" nicht als Ausgang deklarieren.

Weitere Informationen zu den Verwendungsmöglichkeiten finden Sie hier: Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

Beschreibung

Das folgende Bild zeigt Ihnen ein Fließband im Ruhezustand, auf dem sich das Produktionsstück befindet. Das Produktionsstück befindet sich links in der Startlage. Die Pfeile visualisieren die Bewegung des Produktionsstücks von links nach rechts und von rechts nach links, sobald die Produktion anläuft. Die Bewegung des Produktionsstücks von links nach rechts wird aktuell vom Anwenderprogramm angesteuert, doch das Produktionsstück verlässt die Startlage nicht rechtzeitig:

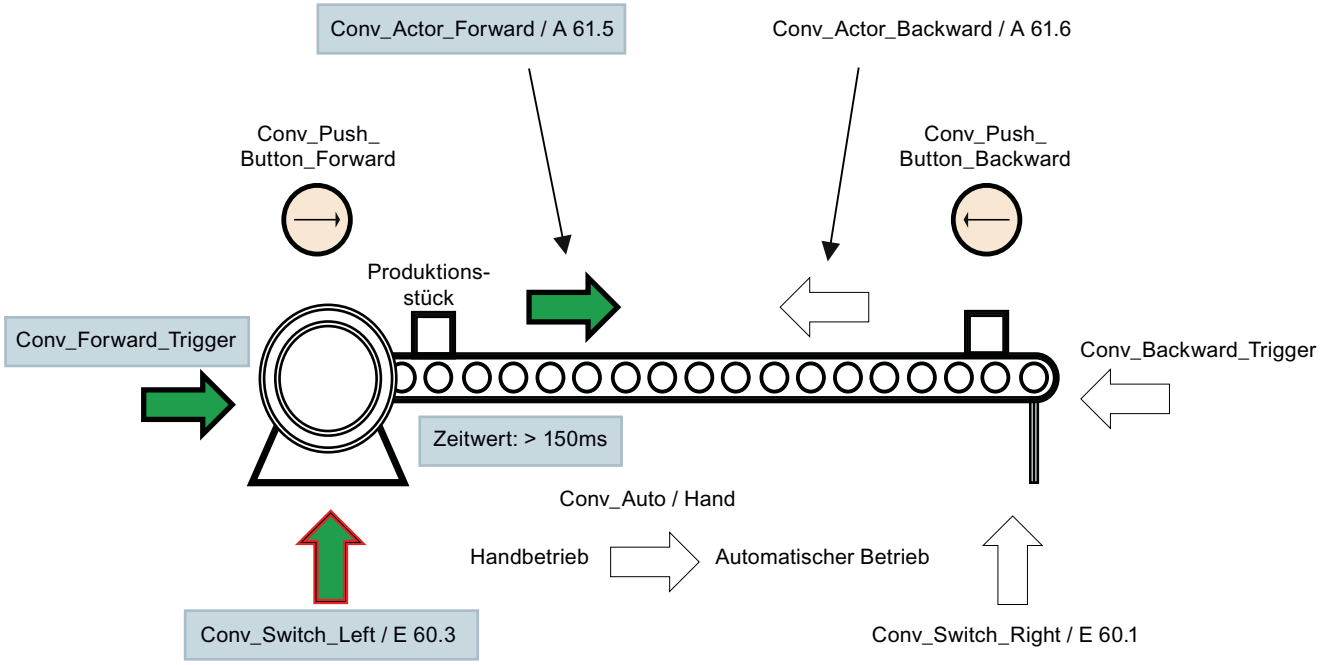

Die für diese Überwachung benötigten Operanden sind grau hinterlegt.

Operand	Erklärung
Conv_Actor_Forward Conv_Actor_Backward	Das sind die Steuersignale (Ausgang des Motors) aus dem Anwenderprogramm für die Bewegungen nach links und rechts.
Conv_Forward_Trigger Conv_Backward_Trigger	Das sind die Steuersignale aus dem Anwenderprogramm (z. B. aus einem GRAPH-Funktionsbaustein) zum Anstoßen der Bewegungen nach links und rechts.
Conv_Switch_Left Conv_Switch_Right	Das ist die Start- und Ziellage des Produktionsstücks.
Conv_Auto / Hand	Das ist der Schalter zur Anwahl der Betriebsart.
Conv_Push_Button_Forward Conv_Push_Button_Backward	Das sind die jeweiligen Handtaster für die Vorwärts- und die Rückwärtsbewegung.

Vorgehen

Geben Sie die folgenden Werte in die Eigenschaften der Aktionsüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Aktion	-
Überwachte Variable	Conv_Switch_Left	TRUE Wenn das Produktionsstück seine Startlage links auf dem Förderband nicht verlässt, dann bleibt der Operand "Conv_Switch_Left" im Signalzustand TRUE. Wenn zusätzlich alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Anlaufzeit	T#200ms	Das Produktionsstück muss seine Startlage innerhalb von 200ms verlassen.
Aktion (Bedingung 1)	Conv_Actor_Forward	TRUE Es liegt das Steuerungssignal für die Bewegung an. Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	-	Die Bedingungen werden nicht benötigt.
Bedingung 3	-	
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Conv_Switch_Left_A_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	-	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Ergebnis

Die Bewegung des Produktionsstücks von links nach rechts wurde vom Programm mithilfe des Operanden "Conv_Forward_Trigger" angesteuert. Der Operand "Conv_Actor_Forward" wechselt in den Signalzustand TRUE und die Vorwärtsbewegung des Produktionsstücks soll von links nach rechts starten. Das Produktionsstück verlässt die Startlage links jedoch nicht rechtzeitig und der Operand "Conv_Switch_Left" bleibt im Signalzustand TRUE. Damit sind alle Voraussetzungen für das Auslösen der Überwachungsmeldung gegeben.

Je nachdem, wie Sie die Einstellungen für eine Basisüberwachung in den ProDiag-Überwachungseinstellungen festgelegt haben, können die Überwachungsmeldungen folgendermaßen aussehen:

Überwachungsmeldung für einen Verriegelungsfehler im Automatischen Betrieb

```
Fehler : Action : CPU_50 : Default_SupervisionFB : E 60.3 : Conv_Switch_Left
```

Siehe auch

Aktionsüberwachung (Seite 8159)

24.3.4.5 Beispiel für eine Reaktionsüberwachung

Voraussetzung

Sie haben eine Überwachung für den Operanden "Conv_Switch_Right" angelegt und als Überwachungsart "Reaktion" ausgewählt.

Hinweis

Verwendungsmöglichkeiten

Achten Sie darauf, dass Sie die globale Variable "Conv_Switch_Right" nicht als Ausgang deklarieren.

Weitere Informationen zu den Verwendungsmöglichkeiten finden Sie hier: Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

Beschreibung

Das folgende Bild zeigt Ihnen ein Fließband im Ruhezustand, auf dem sich das Produktionsstück befindet. Das Produktionsstück befindet sich auf dem Weg von links (Startlage) nach rechts (Endlage). Die Pfeile visualisieren die Bewegung des Produktionsstücks von links nach rechts und von rechts nach links, sobald die Produktion anläuft. Die Bewegung des Produktionsstücks von links nach rechts wurde vom Anwenderprogramm angesteuert, doch das Produktionsstück erreicht seine Endlage nicht rechtzeitig:

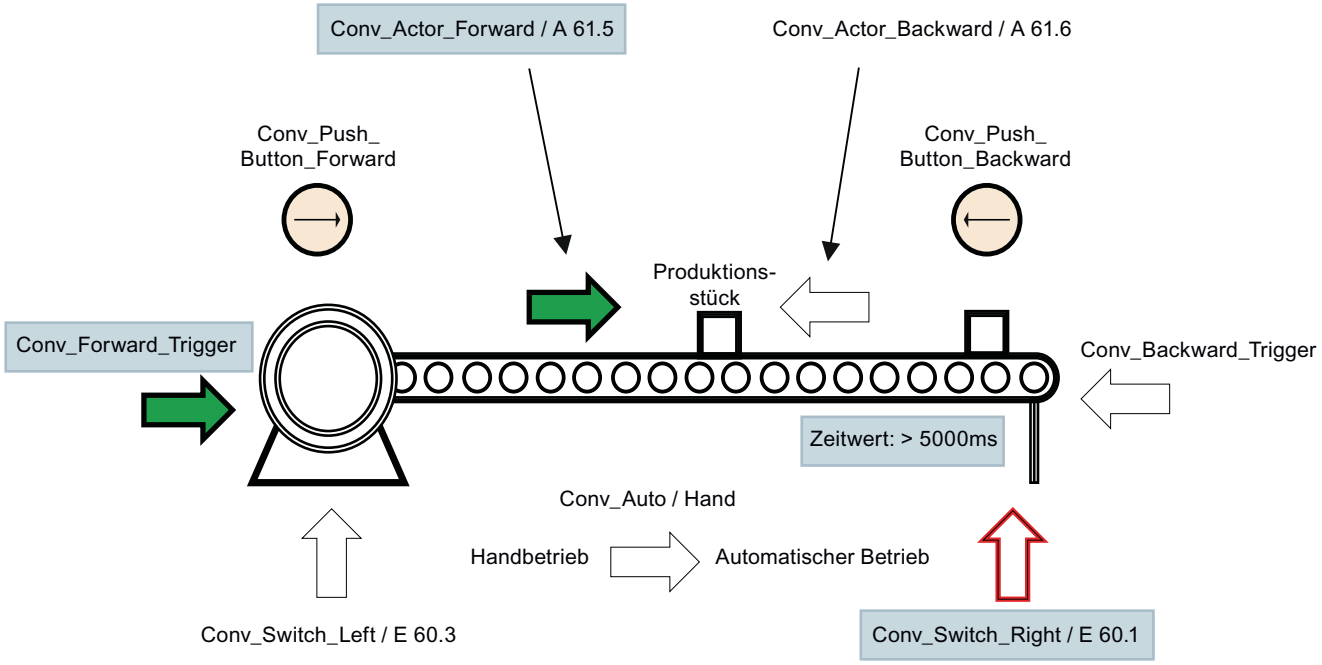

Die für diese Überwachung benötigten Operanden sind grau hinterlegt.

Operand	Erklärung
Conv_Actor_Forward Conv_Actor_Backward	Das sind die Steuersignale (Ausgang des Motors) aus dem Anwenderprogramm für die Bewegungen nach links und rechts.
Conv_Forward_Trigger Conv_Backward_Trigger	Das sind die Steuersignale aus dem Anwenderprogramm (z. B. aus einem GRAPH-Funktionsbaustein) zum Anstoßen der Bewegungen nach links und rechts.
Conv_Switch_Left Conv_Switch_Right	Das ist die Start- und Ziellage des Produktionsstücks.
Conv_Auto / Hand	Das ist der Schalter zur Anwahl der Betriebsart.
Conv_Push_Button_Forward Conv_Push_Button_Backward	Das sind die jeweiligen Handtaster für die Vorwärts- und die Rückwärtsbewegung.

Vorgehen

Geben Sie die folgenden Werte in die Eigenschaften der Reaktionsüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Reaktion	-
Überwachte Variable	Conv_Switch_Right	FALSE Sobald das Produktionsstück seine Endlage erreicht, wechselt der Operand in den Signalzustand TRUE. Wenn das nicht innerhalb der Reaktionszeit geschieht und alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Reaktionszeit	T#5000ms	Die Bewegung des Produktionsstücks von links (Startlage) nach rechts (Endlage) darf nicht länger als 5s dauern.
Aktion (Bedingung 1)	Conv_Actor_Forward	TRUE Es liegt das Steuerungssignal für die Bewegung an. Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	-	Die Bedingungen werden nicht benötigt.
Bedingung 3	-	
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Conv_Switch_Right_R_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	-	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Ergebnis

Die Bewegung des Produktionsstücks von links nach rechts wurde vom Programm mithilfe des Operanden "Conv_Forward_Trigger" angesteuert. Der Operand "Conv_Actor_Forward" wechselt in den Signalzustand TRUE und die Vortwärtsbewegung des Produktionsstücks startet von links nach rechts. Das Produktionsstück erreicht die Endlage rechts jedoch nicht innerhalb von 5s und der Operand "Conv_Switch_Right" bleibt im Signalzustand FALSE. Damit sind alle Voraussetzungen für das Auslösen der Überwachungsmeldung gegeben.

Je nachdem, wie Sie die Einstellungen für eine Basisüberwachung in den ProDiag-Überwachungseinstellungen festgelegt haben, können die Überwachungsmeldungen folgendermaßen aussehen:

Überwachungsmeldung für einen Reaktionsfehler

```
Fehler : Reaction : CPU_50 : Default_SupervisionFB : E 60.1 :
Conv_Switch_Right
```

Siehe auch

Reaktionsüberwachung (Seite 8161)

24.3.4.6 Beispiel für eine Positionsüberwachung

Voraussetzung

Sie haben eine Überwachung für den Operanden "Conv_Switch_Left" angelegt und als Überwachungsart "Position" ausgewählt.

Hinweis

Verwendungsmöglichkeiten

Achten Sie darauf, dass Sie die globale Variable "Conv_Switch_Left" nicht als Ausgang deklarieren.

Weitere Informationen zu den Verwendungsmöglichkeiten finden Sie hier: Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

Beschreibung

Das folgende Bild zeigt Ihnen ein Fließband im Ruhezustand, auf dem sich das Produktionsstück befindet. Das Produktionsstück befindet sich auf dem Weg von links (Startlage) nach rechts (Endlage). Die Pfeile visualisieren die Bewegung des Produktionsstücks von links nach rechts und von rechts nach links, sobald die Produktion anläuft. Die Bewegung des Produktionsstücks von links nach rechts wurde allerdings vom Anwenderprogramm nicht angesteuert und das Produktionsstück hat seine Startlage unerlaubt verlassen:

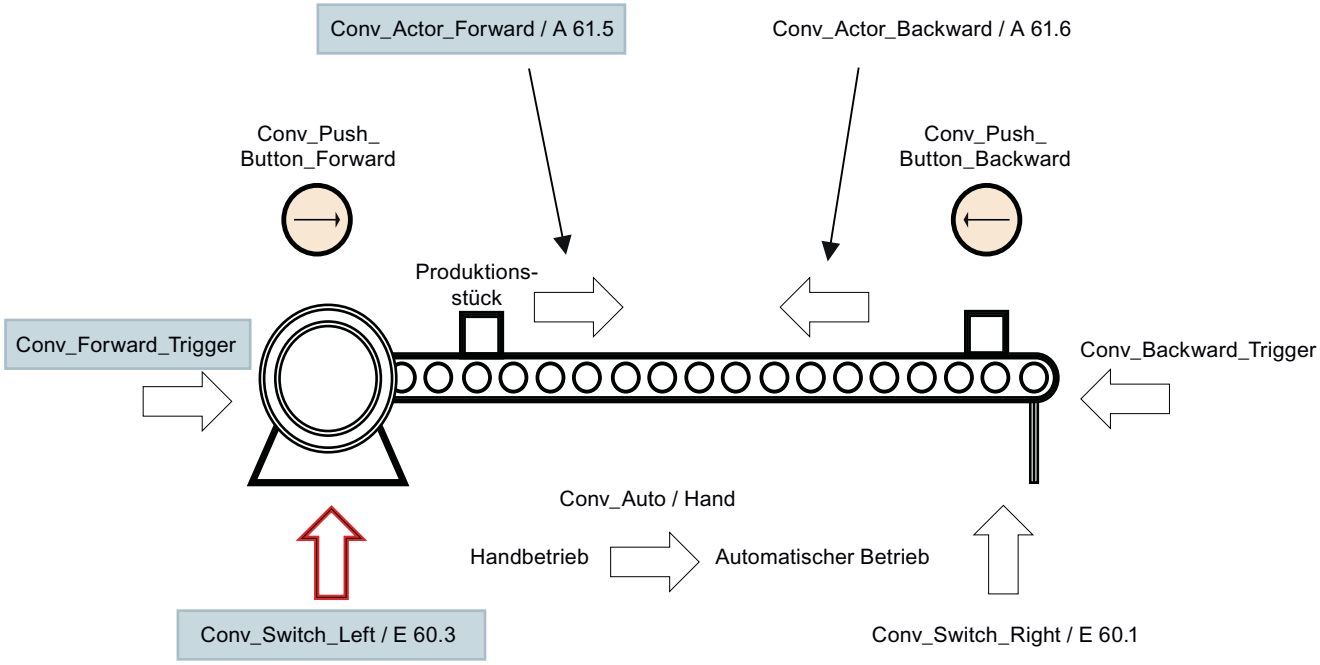

Die für diese Überwachung benötigten Operanden sind grau hinterlegt.

Operand	Erklärung
Conv_Actor_Forward Conv_Actor_Backward	Das sind die Steuersignale (Ausgang des Motors) aus dem Anwenderprogramm für die Bewegungen nach links und rechts.
Conv_Forward_Trigger Conv_Backward_Trigger	Das sind die Steuersignale aus dem Anwenderprogramm (z. B. aus einem GRAPH-Funktionsbaustein) zum Anstoßen der Bewegungen nach links und rechts.
Conv_Switch_Left Conv_Switch_Right	Das ist die Start- und Ziellage des Produktionsstücks.
Conv_Auto / Hand	Das ist der Schalter zur Anwahl der Betriebsart.
Conv_Push_Button_Forward Conv_Push_Button_Backward	Das sind die jeweiligen Handtaster für die Vorwärts- und die Rückwärtsbewegung.

Vorgehen

Geben Sie die folgenden Werte in die Eigenschaften der Reaktionsüberwachung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Position	-
Überwachte Variable	Conv_Switch_Left	FALSE Sobald der Operand unerlaubt in den Signalzustand FALSE wechselt und alle Bedingungen erfüllt sind, wird die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Störung muss sofort erkannt und gemeldet werden. Daher ist als Verzögerungszeit 0ms angegeben.
Aktion (Bedingung 1)	Conv_Actor_Forward	FALSE Es liegt kein Steuerungssignal für die Bewegung an. Mithilfe der Bedingung 1 können Sie die beiden booleschen Operanden in Abhängigkeit zueinander überwachen.
Bedingung 2	-	Die Bedingungen werden nicht benötigt.
Bedingung 3	-	
Kategorie	2: Warnung	Das Auslösen der Überwachungsmeldung wird als Warnung eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Conv_Switch_Left_P_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	Startlage unerlaubt verlassen	Erklärender Text, der in der Überwachungsmeldung mit ausgegeben werden kann.

Ergebnis

Die Bewegung des Produktionsstücks von links nach rechts wurde nicht vom Programm mithilfe des Operanden "Conv_Forward_Trigger" angesteuert. Der Operand "Conv_Actor_Forward" bleibt im Signalzustand FALSE. Die Vortwärtsbewegung des Produktionsstücks von links nach rechts soll nicht starten. Das Produktionsstück verlässt jedoch unerlaubt seine Startlage. Damit sind alle Voraussetzungen für das Auslösen der Überwachungsmeldung gegeben.

Je nachdem, wie Sie die Einstellungen für eine Basisüberwachung in den ProDiag-Überwachungseinstellungen festgelegt haben, können die Überwachungsmeldungen folgendermaßen aussehen:

Überwachungsmeldung für einen Verriegelungsfehler im Automatischen Betrieb

```
Warnung : Position : CPU_50 : Default_SupervisionFB : E 60.3 : Conv_Switch_Left : Startlage
unerlaubt verlassen
```

Siehe auch

Positionsüberwachung (Seite 8163)

24.3.4.7 Beispiel für eine Textmeldung

Voraussetzung

Sie haben eine Überwachung für den Operanden "Trigger_Message" angelegt und als Überwachungsart "Textmeldung" ausgewählt.

Beschreibung

Sie möchten automatisiert vom Zentralrechner eine Textmeldung an alle HMI-Geräte ausgeben, dass z. B. die Anlage in einer Stunde runtergefahren wird um gewartet werden zu können. Damit Sie in der Überwachungsmeldung wirklich nur diese kurze Information ausgeben können und keine zusätzlichen Informationen, wie z. B. CPU-Name, etc., enthalten sind, können Sie die Überwachungsart "Textmeldung" verwenden.

Der gewünschte Text, der in der Überwachungsmeldung enthalten sein soll, wird in das spezifische Textfeld eingetragen.

Die für diese Überwachung benötigte Variable lautet folgendermaßen:

Operand	Erklärung
Trigger_Message	Eine boolesche Variable als Auslöser für die Überwachungsmeldung.

Vorgehen

Geben Sie die folgenden Werte in die Eigenschaften derTextmeldung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Textmeldung	-
Überwachte Variable	Trigger_Message	TRUE Sobald die Variable in den Signalzustand TRUE wechselt, wird die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Überwachungsmeldung soll unmittelbar ausgegeben werden.
Bedingung 1	-	Die Bedingungen werden nicht benötigt.
Bedingung 2	-	
Bedingung 3	-	
Kategorie	3: Information	Das Auslösen der Überwachungsmeldung wird als Information eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Trigger_Message_O_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	Die Anlage wird in 1 Stunde runtergefahren.	Erklärender Text, der in der Überwachungsmeldung ausgegeben wird.

Ergebnis

Sobald die Variable "Trigger_Message" den Signalzustand TRUE einnimmt, wird die Überwachungsmeldung mit dem Text aus dem spezifischen Textfeld ausgegeben.

Die Überwachungsmeldung sieht folgendermaßen aus:

Textmeldung

```
Die Anlage wird in 1 Stunde runtergefahren.
```

Siehe auch

Textmeldung (Seite 8165)

24.3.4.8 Beispiel für eine Fehlermeldung

Voraussetzung

Sie haben eine Überwachung für die Variable "+01R01_Error" angelegt und als Überwachungsart "Fehlermeldung" ausgewählt.

Beschreibung

Sie haben von Ihrem Lieferanten einen Roboter geliefert bekommen. Zusätzlich zur Hardware haben Sie auch eine Textliste mit Fehlertexten, speziell für diesen Robotor, und einen Funktionsbaustein zur Robotorsteuerung bekommen. Der Funktionsbaustein kommuniziert über Profinet mit Ihrem Programm. Dadurch kann der Roboter selbst über seine Steuerung Fehler melden. Sie können sich diese Fehlertexte im Störfall in einer Überwachungsmeldung ausgeben lassen.

Der Funktionsbaustein sieht z. B. folgendermaßen aus:

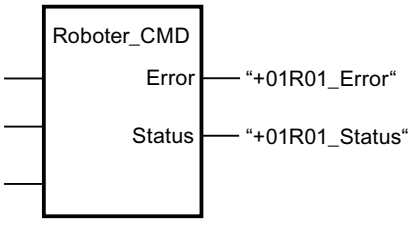

Die Textliste mit den Fehlertexten des Robotors sieht z. B. folgendermaßen aus:

Textlisten-nummer	Eintrag
1	Meldung / Problem 1111
2	Meldung / Problem 2222
3	Meldung / Problem 3333
4	Meldung / Problem 4444

Zur Verwendung dieses Funktionsbausteins in Ihrem Programmcode werden die folgenden Variablen benötigt:

Variablen	Erklärung
+01R01_Error	Eine boolesche Variable als Auslöser für die Auslösung der Fehlermeldung.
+01R01_Status	Entspricht der Fehlernummer

Vorgehen

Um die Fehlertexte des Robotors ausgeben zu können, gehen Sie folgendermaßen vor:

1. Legen Sie die zwei benötigten Variablen in der Standard-Variablentabelle an.

		Name	Datentyp	Adresse
1		+01R01_Error	Bool	%E0.0
2		+01R01_Status	UInt	%EW0

2. Rufen Sie den mitgelieferten Funktionsbaustein in Ihrem Programmcode auf.
3. Geben Sie die folgenden Werte in die Eigenschaften der Meldung ein:

Eigenschaften	Werte	Zustand des Operanden / Erläuterung
Überwachungsart	Fehlermeldung	-
Überwachte Variable	+01R01_Error	TRUE Sobald der Funktionsbaustein des Robotors einen Fehler ausgibt und den Fehlermerker setzt, wird die Variable auf den Signalzustand TRUE gesetzt und die Überwachungsmeldung ausgelöst.
Verzögerungszeit	T#0ms	Die Überwachungsmeldung soll unmittelbar ausgegeben werden.
Bedingung 1	-	Die Bedingungen werden nicht benötigt.
Bedingung 2	-	
Bedingung 3	-	
Kategorie	1: Fehler	Das Auslösen der Überwachungsmeldung wird als Fehler eingestuft.
Unterkategorie 1	-	Die Unterkategorien werden nicht benötigt.
Unterkategorie 2	-	
ProDiag-FB	Default_SupervisionFB	Das ist der zugehörige ProDiag-Funktionsbaustein. Solange Sie keinen eigenen Funktionsbaustein verwenden, wird per Voreinstellung der Default_SupervisionFB verwendet.
Fehlermerker	Default_SupervisionDB.Trigger_Message_O_1.Err	Sobald eine Störung vorliegt, wird dieser Fehlermerker gesetzt.
Spezifisches Textfeld	Folgende Begleitwerte: • @4%4d@ (Fehlernummer) • @4%t#T_Robotor@ (Text aus Textliste)	Fehlertext, der in der Überwachungsmeldung ausgegeben wird.

Weitere Informationen zu den Begleitwerten finden Sie hier:Verwendung der Begleitwerte (Seite 8255)

Ergebnis

Sobald die Variable "+01R01_Error" den Signalzustand TRUE einnimmt, wird die Überwachungsmeldung mit dem Text aus dem spezifischen Textfeld ausgegeben. Die Fehlernummer wird dreistellig dezimal und der Fehlertext aus der Textliste angezeigt.

Je nachdem, wie Sie die Einstellungen für die Fehlermeldung in den ProDiag-Überwachungseinstellungen festgelegt haben, kann eine Fehlermeldung folgendermaßen aussehen:

Fehlermeldung

```
Error : Err_Txt : CPU_50 : Default_SupervisionFB : +01R01_Error : 3 : Meldung / Problem 3333
```

24.3.5 Anwendungsbeispiel für die Überwachung mit ProDiag

Beschreibung

Ein ausführliches Anwendungsbeispiel für die Überwachung mit ProDiag finden Sie im Siemens Industry Online Support hier: Anwendungsbeispiel für ProDiag (https://support.industry.siemens.com/cs/ww/de/view/109740151)

24.4 ProDiag-Funktionsbausteine anlegen

24.4.1 Überblick über die ProDiag-Funktionsbausteine

Beschreibung

Um den Programmcode entsprechend der Maschinen- und Anlageneinheiten zu strukturieren, z. B. Station_1, Station_2, etc., können Sie sich pro Einheit einen ProDiag-Funktionsbaustein anlegen und entsprechend benennen. Ein ProDiag-Funktionsbaustein wird in der Programmiersprache PRODIAG erstellt und kann in der Version 1.0 maximal 250 Überwachungen und in der Version 2.0 maximal 1000 Überwachungen beinhalten.

Sobald Sie einen neuen ProDiag-Funktionsbaustein anlegen, wird ein entsprechender ProDiag-Instanzdatenbaustein angelegt.

Eine detaillierte Ansicht, welche ProDiag-Bausteine in Ihrem Programmcode verwendet werden und wie sie miteinander in Beziehung stehen, finden Sie hier: Überwachte anwenderdefinierte Bausteine anzeigen (Seite 8286)

Vorteile der anwenderdefinierten ProDiag-Funktionsbausteine

Die anwenderdefinierten Einheiten bieten Ihnen viele Vorteile:

- Auf diese Weise können Sie Ihr Programm besser strukturieren und Performance-Vorteile nutzen.
- Jede Einheit bietet Sammelfehlerbits, mit deren Hilfe Sie Reaktionen im Anwenderprogramm definieren können. Zusätzlich können Sie am HMI-Gerät eine einfache Anzeige realisieren.
- Mithilfe des Objekts "ProDiag-Übersicht" können Sie sich auf einem HMI-Gerät eine Gesamtübersicht aller Sammelfehlerbits dieser Einheit erstellen.

Siehe auch

ProDiag-FBs beim Erstellen globaler Überwachungen anlegen (Seite 8192)

Aufbau eines ProDiag-Funktionsbausteins (Seite 8191)

Überblick über den Aufruf der ProDiag-Funktionsbausteine (Seite 8274)

ProDiag-Funktionsbausteine zuweisen (Seite 8263)

ProDiag-Funktionsbausteine anlegen (Seite 8192)

ProDiag-Funktionsbausteine bearbeiten (Seite 8271)

24.4.2 Aufbau eines ProDiag-Funktionsbausteins

Aufbau eines ProDiag-Funktionsbausteins

Das folgende Bild zeigt exemplarisch die Komponenten eines ProDiag-Funktionsbausteins:

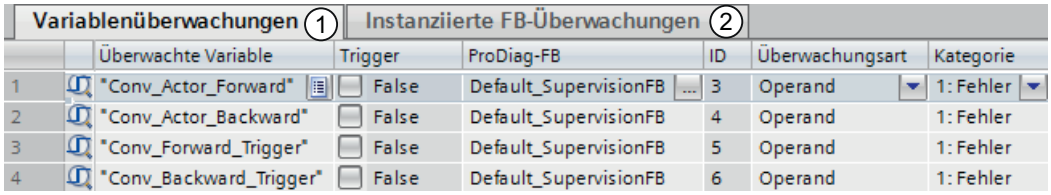

Nummerierung	Erklärung	Weiterführende Informationen
1	Übersicht aller globalen Überwachungen	Alle Überwachungen, die diesem ProDiag-Funktionsbaustein zugeordnet sind.
2	Übersicht aller instanziierten FB-Überwachungen	

24.4.3 Anordnung der Spalten im ProDiag-FB sichern

Sie haben die Möglichkeit, die Darstellung der Spalten im ProDiag-Funktionsbaustein an Ihre Bedürfnisse anzupassen. So können Sie z. B. die Spalten ausblenden, die Sie nicht benötigen. Die Reihenfolge der Spalten können Sie nicht verändern. Die von Ihnen angepasste Ansicht können Sie anschließend sichern.

Vorgehen

Um die Anordnung der Spalten zu sichern, gehen Sie folgendermaßen vor:

1. Passen Sie die Spalten nach Ihren Bedürfnissen an.
2. Klicken Sie innerhalb des ProDiag-Funktionsbausteins auf die Schaltfläche "Anordnung merken".

Ergebnis

Die Anordnung wird gesichert. Wenn Sie den ProDiag-Funktionsbaustein erneut öffnen, wird diese Anordnung verwendet.

24.4.4 ProDiag-Funktionsbausteine anlegen

24.4.4.1 ProDiag-FBs beim Erstellen globaler Überwachungen anlegen

Voraussetzung

Sie haben mindestens eine globale Variable angelegt.

Vorgehen

Um einen ProDiag-Funktionsbaustein direkt beim Anlegen einer globalen Überwachung anzulegen, gehen Sie folgendermaßen vor:

1. Legen Sie eine Überwachung für eine globale Variable an.
   Im Inspektorfenster "Eigenschaften" wird das Register "Überwachungen" geöffnet und eine Überwachung für die Variable angelegt. Beim Anlegen der ersten Überwachung wird auch automatisch der ProDiag-Funktionsbaustein "Default_SupervisionFB" angelegt.
2. Geben Sie die gewünschten Eigenschaften ein.
3. Klicken Sie bei ProDiag-FB auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Um einen neuen ProDiag-Funktionsbaustein anzulegen, klicken Sie in der rechten Spalte auf "Hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
6. Klicken Sie auf die Schaltfläche "Funktionsbaustein".
7. Geben Sie einen Namen für den neuen ProDiag-Funktionsbaustein ein.
8. Wählen Sie im Abschnitt "Sprache > PRODIAG (inkl. IDB)" aus.
9. Um weitere Eigenschaften des neuen ProDiag-Funktionsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
10. Geben Sie die gewünschten Eigenschaften ein.
11. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der ProDiag-Funktionsbaustein nach dem Erstellen gleich geöffnet werden soll.
12. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Neben dem neuen ProDiag-Funktionsbaustein wird auch ein entsprechender ProDiag-Instanzdatenbaustein erzeugt. Sie finden die Bausteine in der Projektnavigation im Ordner "Programmbausteine".

24.4.4.2 ProDiag-FBs beim Bausteinaufruf anlegen

Voraussetzung

Sie haben einen Funktionsbaustein und mindestens eine lokale Überwachung angelegt.

Vorgehen

Um einen ProDiag-Funktionsbaustein direkt beim Aufruf eines Funktionsbausteins anzulegen, gehen Sie folgendermaßen vor:

1. Rufen Sie den Funktionsbaustein, der die Überwachungen enthält, im Programmcode auf. Der Dialog "Aufrufoptionen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Einzel-Instanz".
3. Klicken Sie bei ProDiag-FB auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet. Soweit vorhanden, wird automatisch der ProDiag-Funktionsbaustein "Default_SupervisionFB" angezeigt. Wenn es diesen nicht gibt, dann wird der erste gefundene anwenderdefinierte ProDiag-Funktionsbaustein angezeigt.
4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Um einen neuen ProDiag-Funktionsbaustein anzulegen, klicken Sie in der rechten Spalte auf "Hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
6. Klicken Sie auf die Schaltfläche "Funktionsbaustein".
7. Geben Sie einen Namen für den neuen ProDiag-Funktionsbaustein ein.
8. Wählen Sie im Abschnitt "Sprache > PRODIAG (inkl. IDB)" aus.
9. Um weitere Eigenschaften des neuen ProDiag-Funktionsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
10. Geben Sie die gewünschten Eigenschaften ein.
11. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der ProDiag-Funktionsbaustein nach dem Erstellen gleich geöffnet werden soll.
12. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Neben dem neuen ProDiag-Funktionsbaustein wird auch ein entsprechender ProDiag-Instanzdatenbaustein erzeugt. Sie finden die Bausteine in der Projektnavigation im Ordner "Programmbausteine". Die Überwachungen des Funktionsbausteins sind instanziiert.

24.4.4.3 ProDiag-FBs am Instanzdatenbaustein eines FBs anlegen

Voraussetzung

Sie haben einen Funktionsbaustein mit mindestens einer lokalen Überwachung angelegt und übersetzt.

Vorgehen

Um einen ProDiag-Funktionsbaustein in den Eigenschaften des Instanzdatenbausteins anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie den Instanzdatenbaustein eines Funktionsbausteins, der lokale Überwachungen enthält, mit einem rechten Mausklick an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaften-Dialog des Instanzdatenbausteins wird geöffnet.
3. Klicken Sie den Abschnitt "Attribute" an.
4. Klicken Sie bei "ProDiag-FB zuweisen" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
5. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
6. Um einen neuen ProDiag-Funktionsbaustein anzulegen, klicken Sie in der rechten Spalte auf "Hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
7. Klicken Sie auf die Schaltfläche "Funktionsbaustein".
8. Geben Sie einen Namen für den neuen ProDiag-Funktionsbaustein ein.
9. Wählen Sie im Abschnitt "Sprache > PRODIAG (inkl. IDB)" aus.
10. Um weitere Eigenschaften des neuen ProDiag-Funktionsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
    Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
11. Geben Sie die gewünschten Eigenschaften ein.
12. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der ProDiag-Funktionsbaustein nach dem Erstellen gleich geöffnet werden soll.
13. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Neben dem neuen ProDiag-Funktionsbaustein wird auch ein entsprechender ProDiag-Instanzdatenbaustein erzeugt. Sie finden die Bausteine in der Projektnavigation im Ordner "Programmbausteine". Die Überwachungen des Funktionsbausteins sind instanziiert.

24.4.4.4 ProDiag-FBs mithilfe des Dialogs Neuen Baustein hinzufügen anlegen

Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

Vorgehen

Um einen ProDiag-Funktionsbaustein mithilfe des Dialogs "Neuen Baustein hinzufügen" anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktionsbaustein".
3. Geben Sie einen Namen für den neuen ProDiag-Funktionsbaustein ein.
4. Wählen Sie im Abschnitt "Sprache > PRODIAG (inkl. IDB)" aus.
5. Um weitere Eigenschaften des neuen ProDiag-Funktionsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
6. Geben Sie die gewünschten Eigenschaften ein.
7. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der ProDiag-Funktionsbaustein nach dem Erstellen gleich geöffnet werden soll.
8. Bestätigen Sie Ihre Eingaben mit "OK".

Oder:

Um einem bereits angelegten anwenderdefinierten Funktionsbaustein, der lokale Überwachungen enthält, einen ProDiag-Funktionsbaustein mithilfe des Dialogs "Neuen Baustein hinzufügen" zuzuweisen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein".
3. Wählen Sie bei Typ aus der Klappliste den bereits angelegten Funktionsbaustein, der lokale Überwachungen enthält, aus.
4. Klicken Sie bei ProDiag-FB auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
5. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
6. Um einen neuen ProDiag-Funktionsbaustein anzulegen, klicken Sie in der rechten Spalte auf "Hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
7. Ab hier gilt die gleiche Vorgehensweise wie ab Punkt 2 oben beschrieben.

Ergebnis

Neben dem neuen ProDiag-Funktionsbaustein wird auch ein entsprechender ProDiag-Instanzdatenbaustein erzeugt. Sie finden die Bausteine in der Projektnavigation im Ordner "Programmbausteine".

24.4.4.5 ProDiag-FBs in den ProDiag-Übersichtstabellen anlegen

Voraussetzung

Sie haben lokale und globale Überwachungen angelegt.

Vorgehen

Um einen ProDiag-Funktionsbaustein in den ProDiag-Übersichtstabellen "Variablenüberwachungen" und "Instanziierte FB-Überwachungen" anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
2. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
3. Um einen neuen ProDiag-Funktionsbaustein anzulegen, klicken Sie in der rechten Spalte auf "Hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "Funktionsbaustein".
5. Geben Sie einen Namen für den neuen ProDiag-Funktionsbaustein ein.
6. Wählen Sie im Abschnitt "Sprache > PRODIAG (inkl. IDB)" aus.
7. Um weitere Eigenschaften des neuen ProDiag-Funktionsbausteins einzugeben, klicken Sie auf "Weitere Informationen".
   Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
8. Geben Sie die gewünschten Eigenschaften ein.
9. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der ProDiag-Funktionsbaustein nach dem Erstellen gleich geöffnet werden soll.
10. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Neben dem neuen ProDiag-Funktionsbaustein wird auch ein entsprechender ProDiag-Instanzdatenbaustein erzeugt. Sie finden die Bausteine in der Projektnavigation im Ordner "Programmbausteine". Die lokalen Überwachungen sind instanziiert.

24.5 Zentralen Zeitstempel verwenden

24.5.1 Zentralen Zeitstempel verwenden

Beschreibung

Ab der ProDiag-Funktionsbaustein Version 2.0 haben Sie die Möglichkeit, eine zentrale Zeitstempel-Variable in den ProDiag-Überwachungseinstellungen vorzugeben und diese anschließend im gesamten Projekt in jeder CPU zu verwenden.

Die zentrale Zeitstempel-Variable wird dafür als globale Variable oder als globales Datenbausteinelement in der jeweiligen CPU definiert.

Allen ProDiag-Überwachungsmeldungen innerhalb eines Programmzyklus, die von ProDiag-Funktionsbausteinen mit der aktivierten Option "Zentralen Zeitstempel verwenden" verschickt werden, wird die gleiche Systemzeit zugewiesen.

Voraussetzung

Der ProDiag-Funktionsbaustein hat die Version 2.0.

Hinweis

ProDiag-Funktionsbaustein Version 1.0

Allen ProDiag-Überwachungsmeldungen, die von einem ProDiag-Funktionsbaustein Version 1.0 verschickt werden, wird weiterhin der lokale Baustein-Zeitstempel zugewiesen.

Vorgehen

Um einen zentralen Zeitstempel zu verwenden, gehen Sie folgendermaßen vor:

1. Stellen Sie in den Bausteineigenschaften des ProDiag-Funktionsbausteins sicher, dass der Baustein die Version 2.0 hat.
2. Öffnen Sie in der Projektnavigation unter "Gemeinsame Daten" > "Überwachungseinstellungen" > "Allgemein" den Bereich "Zentraler Zeitstempel".
3. Definieren Sie eine Zeitstempel-Variable, indem Sie in das Textfeld einen freien Text eintragen.

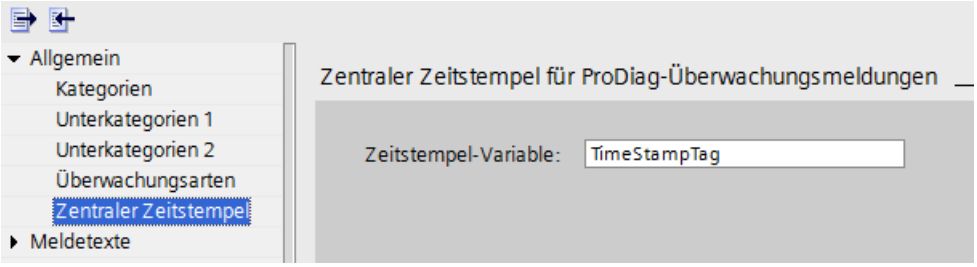

4. Definieren Sie in jeder CPU, in der Sie diesen Zeitstempel verwenden möchten, eine globale Variable oder ein globales Datenbausteinelement mit der Bezeichnung aus dem freien Textfeld vom Datentyp LDT.

TimeStampTag	LDT	%M1.0

5. Erweitern Sie Ihren Programmcode (z. B. im Main[OB1]) in den entsprechenden CPUs um eine Uhrzeitfunktion, z. B. "RD_SYS_T", um den aktuellen Zeitstempel in die globale Zeitstempel-Variable "TimeStampTag" zu kopieren:

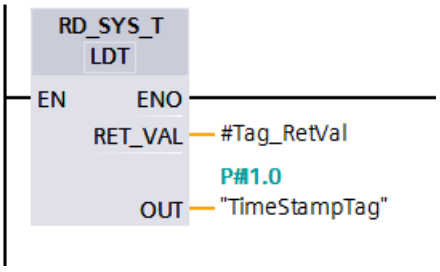

6. Wenn Sie den zentralen Zeitstempel in bereits vorhandenen ProDiag-Funktionsbausteinen verwenden möchten, dann müssen Sie die Option "Zentralen Zeitstempel verwenden" in den Bausteineigenschaften des jeweiligen ProDiag-Funktionsbausteins aktivieren:

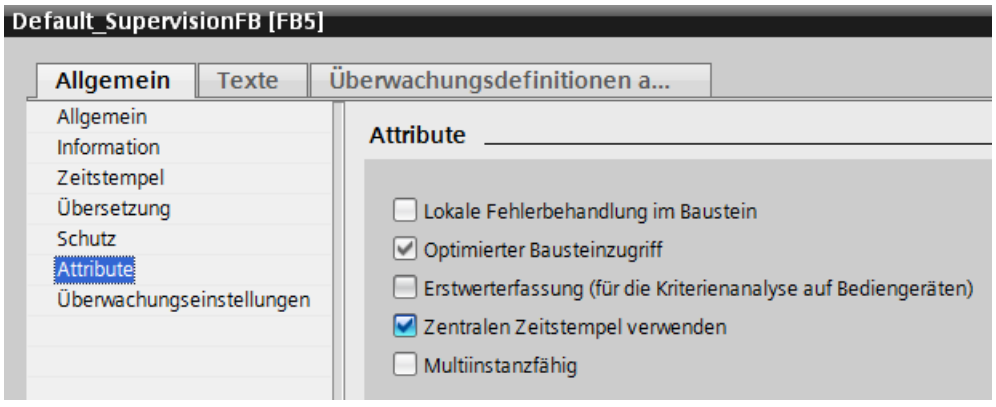

Wenn Sie die ProDiag-Funktionsbausteine erst nach der Definition der zentralen Zeitstempel-Variable anlegen, dann wird diese Option automatisch aktiviert.

Ergebnis

Innerhalb eines Programmzyklus erhalten alle ProDiag-Überwachungsmeldungen, bei deren ProDiag-Funktionsbaustein die Option "Zentralen Zeitstempel verwenden" aktiviert wurde, einen einheitlichen Zeitstempel.

24.5.2 Mehrere Überwachungsmeldungen eines ProDiag-FBs innerhalb einer Zykluszeit

Beschreibung

Pro Zykluszeit einer CPU kann jeweils nur eine Überwachungsmeldung pro ProDiag-Funktionsbaustein ausgegeben werden, auch wenn mehrere Störungen gleichzeitig aufgetreten sind. Alle gleichzeitig ausgelösten ProDiag-Überwachungsmeldungen eines ProDiag-Funktionsbausteins erhalten den gleichen Zeitstempel, der die Uhrzeit der auslösenden Störung enthält. Die restlichen ProDiag-Überwachungsmeldungen können der Reihe nach abgearbeitet werden, d. h. jeweils eine pro Zykluszeit der CPU. Dadurch werden Sprünge in der Zykluszeit durch Meldeschwalle verhindert.

Die Zeitstempel der Überwachungsmeldungen können Sie sich z. B. in der Spalte "Zeit" in der Meldeanzeige anzeigen lassen.

Je nachdem, ob Sie die Option "Zentralen Zeitstempel verwenden" im ProDiag-Funktionsbaustein aktiviert haben, wird entweder die aktuelle Systemzeit oder die lokale Bausteinzeit als Zeitstempel verwendet.

Siehe auch

Zentralen Zeitstempel verwenden (Seite 8197)

24.6 ProDiag-Instanzdatenbausteine verwenden

24.6.1 Aufbau eines ProDiag-Instanzdatenbausteins der Version 1.0

Aufbau eines ProDiag-Instanzdatenbausteins der Version 1.0

Das folgende Bild zeigt exemplarisch die Komponenten eines ProDiag-Instanzdatenbausteins:

Default_SupervisionDB (1)

	Name	Datentyp
1	Input	
2	Output	
3	InOut	
4	Static	
5	State (2)	SV_FB_State
6	Conv_Actor_Forward_O_2	SV_State
7	Conv_Actor_Backward_O_2	SV_State
8	Conv_Forward_Trigger_O_2	SV_State
9	Conv_Actor_Forward_O_1	SV_State
10	Conv_Actor_Backward_O_1	SV_State
11	Conv_Forward_Trigger_O_1 (3)	SV_State
12	Conv_Backward_Trigger_O_1	SV_State
13	Conv_Switch_Left_O_1	SV_State
14	Conv_Switch_Right_O_1	SV_State
15	Conv_Auto_Hand_O_1	SV_State
16	Conv_Push_Button_Forward_O_1	SV_State
17	Conv_Push_Button_Backward_O_1	SV_State

Nummerierung	Erklärung	Weiterführende Informationen
1	Die Bausteinschnittstelle des ProDiag-Instanzdatenbausteins	Die Bausteinschnittstelle ist nicht editierbar.
2	Die Zustandsvariable "State" im Abschnitt "Static"	Die Zustandsvariable "State" vom PLC-Datentyp "SV_FB_State" beinhaltet die Sammelfehlerbits aller Überwachungen, die diesem ProDiag-Instanzdatenbaustein zugeordnet sind. Es gibt zwei verschiedene Arten von Sammelfehlerbits: • Sammelfehlerbit für die Überwachungsarten (All, O, I, R, A, P, Merr und Mtxt) • Sammelfehlerbit für die Kategorien (C1, C2, C3, C4, C5, C6, C7 und C8) Diese Sammelfehlerbits können Sie entweder auf Ihrem HMI-Gerät mithilfe des Objekts "ProDiag-Übersicht" visualisieren oder Sie können sie in Ihrem Programmcode als Bedingung zur weiteren Ausführung des Programmcodes programmieren. Auf die Sammelfehlerbits können Sie im Programmcode nur lesend zugreifen. Sobald eine Überwachungsmeldung ausgelöst wird, werden die Sammelfehlerbits des zugewiesenen ProDiag-Instanzdatenbausteins gesetzt.
3	Zustandsvariablen	Diese Variablen werden angezeigt, wenn Überwachungen angelegt und diesem ProDiag-Instanzdatenbaustein zugeordnet wurden.

24.6.2 Inhalt eines ProDiag-Instanzdatenbausteins der Version 1.0

Bausteinschnittstelle

Sobald Sie einen neuen ProDiag-Funktionsbaustein anlegen, wird ein entsprechender ProDiag-Instanzdatenbaustein angelegt.

Im Abschnitt "Static" stehen Ihnen die folgenden Elemente zur Verfügung:

State

Name	Datentyp	Erläuterung
State	SV_FB_State	Systeminterner, Know-how-geschützter PLC-Datentyp

Name	Datentyp	Erläuterung
Total_Number	UINT	Anzahl aller Überwachungen, denen dieser ProDiag-Funktionsbaustein zugewiesen wurde.
All	BOOL	Sammelfehlerbit für alle enthaltenen Überwachungsarten, außer der Textmeldung: • TRUE => Es wurde mindestens eine Überwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Überwachung ausgelöst. Erreichbar aus HMI und sichtbar in HMI.
O (Operand)	BOOL	Sammelfehlerbit für alle Operandenüberwachungen: • TRUE => Es wurde mindestens eine Operandenüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Operandenüberwachung ausgelöst.
I (Interlock)	BOOL	Sammelfehlerbit für alle Verriegelungsüberwachungen: • TRUE => Es wurde mindestens eine Verriegelungsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Verriegelungsüberwachung ausgelöst.
R (Reaction)	BOOL	Sammelfehlerbit für alle Reaktionsüberwachungen: • TRUE => Es wurde mindestens eine Reaktionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Reaktionsüberwachung ausgelöst.
A (Action)	BOOL	Sammelfehlerbit für alle Aktionsüberwachungen: • TRUE => Es wurde mindestens eine Aktionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Aktionsüberwachung ausgelöst.
P (Position)	BOOL	Sammelfehlerbit für alle Positionsüberwachungen: • TRUE => Es wurde mindestens eine Positionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Positionsüberwachung ausgelöst.
Merr (Message error)	BOOL	Sammelfehlerbit für alle Fehlermeldungen: • TRUE => Es wurde mindestens eine Fehlermeldung im aktuellen Programmzyklus ausgelöst.

Name		Datentyp	Erläuterung
			• FALSE => Es wurde keine Fehlermeldung ausgelöst.
	Mtxt (Message text)	BOOL	Sammelfehlerbit für alle Textmeldungen: • TRUE => Es wurde mindestens eine Textmeldung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Textmeldung ausgelöst.
	SV_Types (Supervision types)	BYTE	Byte für alle Sammelfehlerbits der Überwachungsarten Jedem einzelnen Bit ist eine Überwachungsart zugeordnet. Dadurch können Sie dieses Byte z. B. im Objekt "ProDiag-Übersicht" auf dem HMI-Gerät verwenden, um sich eine Anzeige zu erstellen, die Ihnen auf den ersten Blick anzeigt, wenn eine Überwachung einer bestimmten Überwachungsart ausgelöst wurde. • Bit 1 => Operand • Bit 2 => Interlock • etc. Erreichbar aus HMI und sichtbar in HMI.
	C1 (Category)	BOOL	Erste Kategorie (Per Voreinstellung lautet die erste Kategorie = Fehler und das können Sie auch nicht verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C1 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt kein Fehler vor.
	C2 (Category)	BOOL	Zweite Kategorie (Per Voreinstellung lautet die zweite Kategorie = Warnung. Diese Einstellung können Sie jederzeit verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C2 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C3 (Category)	BOOL	Dritte Kategorie (Per Voreinstellung lautet die dritte Kategorie = Info. Diese Einstellung können Sie jederzeit verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C3 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C4 (Category)	BOOL	Vierte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C4 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C5 (Category)	BOOL	Fünfte Kategorie

Name		Datentyp	Erläuterung
			• TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C5 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C6 (Category)	BOOL	Sechste Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C6 im aktuellen Programmzyklus ausgelöst. • FALSE =>Es liegt keine Störung vor.
	C7 (Category)	BOOL	Siebte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C7 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C8 (Category)	BOOL	Achte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C8 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	Categories	BYTE	Sammelfehler-Byte für alle Kategorien Jedem einzelnen Bit ist eine Kategorie zugeordnet. Dadurch können Sie dieses Byte z. B. im Objekt "ProDiag-Übersicht" auf dem HMI-Gerät verwenden, um sich eine Anzeige zu erstellen, die Ihnen auf den ersten Blick anzeigt, wenn eine Überwachung einer bestimmten Kategorie ausgelöst wurde. • Bit 1 => Kategorie 1 (C1) • Bit 2 => Kategorie 2 (C2) • etc. Erreichbar aus HMI und sichtbar in HMI.
	Sub_Cat_1 (Subcategory 1)	DWORD	Sammelfehlerbits für die Unterkategorien 1 (Bit 1 bis 16)
	Sub_Cat_2 (Subcategory 2)	DWORD	Sammelfehlerbits für die Unterkategorien 2 (Bit 1 bis 16)
	Reserved	BYTE	Reserve-Byte

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Optionale Zustandsvariable, sobald eine Überwachung angelegt wurde

Sobald Sie eine Überwachung anlegen, wird in der Bausteinschnittstelle zusätzlich dieser Abschnitt angezeigt:

Name	Datentyp	Erläuterung
z. B. Conv_Push_Button_Forward_O_1 • Conv_Push_Button_Forward = Name des überwachten Operanden • O = Überwachungsart "Operand" • 1 = Anzahl der Überwachungen von der Überwachungsart "Operand"	SV_State	Systeminterner, know-how-geschützter PLC-Datentyp
Err (Error)	BOOL	Fehler, der auch in den Eigenschaften der Überwachung angezeigt wird. Es zeigt an, ob eine Überwachungsmeldung ausgelöst wurde. • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Erreichbar aus HMI und sichtbar in HMI
Err_Neg_Flag (Error_Negativ_Flag)	BOOL	Für die negative Flankenauswertung des Fehlermerkers
T_D_E_Going (Time_Date_Error_Going)	LDT	Uhrzeit und Tag im Programmzyklus, wann die Störung gegangen ist.
E_Flag (Error_Flag)	BOOL	Fehlermerker: Es wird nur eine Überwachungsmeldung pro ProDiag-Funktionsbaustein und pro Programmzyklus versendet.
E_Pos_Flag (Error_Positive_Flag)	BOOL	Für die positive Flankenauswertung des Fehlermerkers
T_D_E_Coming (Time_Date_Error_Coming)	LDT	Uhrzeit und Tag im Programmzyklus, wann die Störung aufgetreten ist.
S_Coming (Signal_Coming)	BOOL	Merker • TRUE = Die Überwachungsmeldung wurde störungsfrei versandt • FALSE = Es liegt eine Störung vor.
Pos_Flag (Positive_Flag)	BOOL	Dieser Merker wird nur für die Überwachungsart "Position" benötigt.
Ti (Timer)	TON_TIME	Diese Zeit wird für die Bildung der Verzögerungszeit benötigt.
PT (Preset Time)	TIME	Voreingestellter Zeitwert
ET (Elapsed Time)	TIME	Aktueller Zeitwert
IN (Input)	TIME	Starteingang
Q (Output)	TIME	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird

24.6.3 Aufbau eines ProDiag-Instanzdatenbausteins der Version 2.0

Aufbau eines ProDiag-Instanzdatenbausteins der Version 2.0

Das folgende Bild zeigt exemplarisch die Komponenten eines ProDiag-Instanzdatenbausteins:

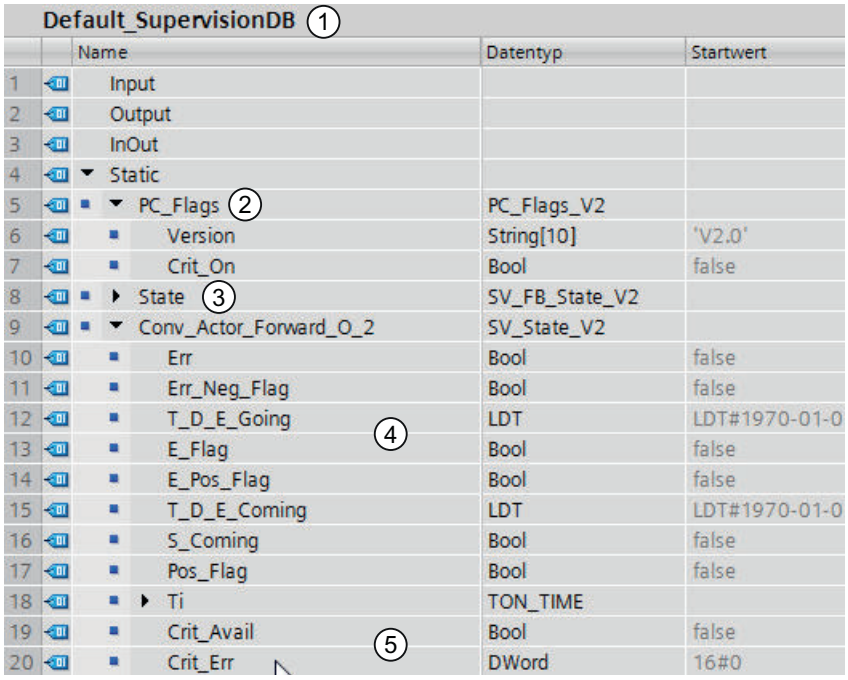

	Default_SupervisionDB (1)		
	Name	Datentyp	Startwert
1	Input		
2	Output		
3	InOut		
4	Static		
5	PC_Flags (2)	PC_Flags_V2	
6	Version	String[10]	'V2.0'
7	Crit_On	Bool	false
8	State (3)	SV_FB_State_V2	
9	Conv_Actor_Forward_O_2	SV_State_V2	
10	Err	Bool	false
11	Err_Neg_Flag	Bool	false
12	T_D_E_Going (4)	LDT	LDT#1970-01-0
13	E_Flag	Bool	false
14	E_Pos_Flag	Bool	false
15	T_D_E_Coming	LDT	LDT#1970-01-0
16	S_Coming	Bool	false
17	Pos_Flag	Bool	false
18	Ti	TON_TIME	
19	Crit_Avail (5)	Bool	false
20	Crit_Err	DWord	16#0

Nummerierung	Erklärung	Weiterführende Informationen
1	Die Bausteinschnittstelle des ProDiag-Instanzdatenbausteins	Die Bausteinschnittstelle ist nicht editierbar.
2	Die Zustandsvariable "PC_Flags"	Die Zustandsvariable "PC_Flags" vom PLC-Datentyp "PC_Flags_V2" beinhaltet die Version des ProDiag-Funktionsbausteins (Parameter Version) und den Parameter CRIT_ON, der anzeigt, ob die Erstwerterfassung aktiviert ist.

Nummerierung	Erklärung	Weiterführende Informationen
3	Die Zustandsvariable "State"	Die Zustandsvariable "State" vom PLC-Datentyp "SV_FB_State_V2" beinhaltet die Sammelfehlerbits aller Überwachungen, die diesem ProDiag-Instanzdatenbaustein zugeordnet sind. Es gibt zwei verschiedene Arten von Sammelfehlerbits: • Sammelfehlerbit für die Überwachungsarten (All, O, I, R, A, P, Merr und Mtxt) • Sammelfehlerbit für die Kategorien (C1, C2, C3, C4, C5, C6, C7 und C8) Diese Sammelfehlerbits können Sie entweder auf Ihrem HMI-Gerät mithilfe des Objekts "ProDiag-Übersicht" visualisieren oder Sie können sie in Ihrem Programmcode als Bedingung zur weiteren Ausführung des Programmcodes programmieren. Auf die Sammelfehlerbits können Sie im Programmcode lesend und schreibend zugreifen. Sobald eine Überwachungsmeldung ausgelöst wird, werden die Sammelfehlerbits des zugewiesenen ProDiag-Instanzdatenbausteins gesetzt.
4	Zustandsvariablen	Diese Variablen vom PLC-Datentyp "SV_State_V2" werden angezeigt, wenn Überwachungen angelegt und diesem ProDiag-Instanzdatenbaustein zugeordnet wurden.
5	Die Parameter Crit_Avail und Crit_Err	• Crit_Avail: Der Signalzustand = TRUE: Die Erstwerte wurden aufgezeichnet. • Crit_Err: Ablage der 32 Signalzustände pro Netzwerk eines KOP- oder FUP-Bausteins.

24.6.4 Inhalt eines ProDiag-Instanzdatenbausteins der Version 2.0

Bausteinschnittstelle

Sobald Sie einen neuen ProDiag-Funktionsbaustein anlegen, wird ein entsprechender ProDiag-Instanzdatenbaustein angelegt.

Im Abschnitt "Static" stehen Ihnen die folgenden Elemente zur Verfügung:

State

Name		Datentyp	Erläuterung
PC_Flags		PC_Flags_V2	Systeminterner, Know-how-geschützter PLC-Datentyp
	Version	String[10]	Version des ProDiag-Funktionsbausteins
	Crit_On	BOOL	Zeigt an, ob die Erstwerterfassung aktiviert ist.
State		SV_FB_State_V2	Systeminterner, Know-how-geschützter PLC-Datentyp

Name		Datentyp	Erläuterung
	Total_Number	UINT	Anzahl aller Überwachungen, denen dieser ProDiag-Funktionsbaustein zugewiesen wurde.
	All	BOOL	Sammelfehlerbit für alle enthaltenen Überwachungsarten, außer der Textmeldung: • TRUE => Es wurde mindestens eine Überwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Überwachung ausgelöst. Erreichbar aus HMI und sichtbar in HMI.
	O (Operand)	BOOL	Sammelfehlerbit für alle Operandenüberwachungen: • TRUE => Es wurde mindestens eine Operandenüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Operandenüberwachung ausgelöst.
	I (Interlock)	BOOL	Sammelfehlerbit für alle Verriegelungsüberwachungen: • TRUE => Es wurde mindestens eine Verriegelungsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Verriegelungsüberwachung ausgelöst.
	R (Reaction)	BOOL	Sammelfehlerbit für alle Reaktionsüberwachungen: • TRUE => Es wurde mindestens eine Reaktionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Reaktionsüberwachung ausgelöst.
	A (Action)	BOOL	Sammelfehlerbit für alle Aktionsüberwachungen: • TRUE => Es wurde mindestens eine Aktionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Aktionsüberwachung ausgelöst.
	P (Position)	BOOL	Sammelfehlerbit für alle Positionsüberwachungen: • TRUE => Es wurde mindestens eine Positionsüberwachung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Positionsüberwachung ausgelöst.
	Merr (Message error)	BOOL	Sammelfehlerbit für alle Fehlermeldungen: • TRUE => Es wurde mindestens eine Fehlermeldung im aktuellen Programmzyklus ausgelöst.

Name		Datentyp	Erläuterung
			• FALSE => Es wurde keine Fehlermeldung ausgelöst.
	Mtxt (Message text)	BOOL	Sammelfehlerbit für alle Textmeldungen: • TRUE => Es wurde mindestens eine Textmeldung im aktuellen Programmzyklus ausgelöst. • FALSE => Es wurde keine Textmeldung ausgelöst.
	SV_Types (Supervision types)	BYTE	Byte für alle Sammelfehlerbits der Überwachungsarten Jedem einzelnen Bit ist eine Überwachungsart zugeordnet. Dadurch können Sie dieses Byte z. B. im Objekt "ProDiag-Übersicht" auf dem HMI-Gerät verwenden, um sich eine Anzeige zu erstellen, die Ihnen auf den ersten Blick anzeigt, wenn eine Überwachung einer bestimmten Überwachungsart ausgelöst wurde. • Bit 1 => Operand • Bit 2 => Interlock • etc. Erreichbar aus HMI und sichtbar in HMI.
	C1 (Category)	BOOL	Erste Kategorie (Per Voreinstellung lautet die erste Kategorie = Fehler und das können Sie auch nicht verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C1 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt kein Fehler vor.
	C2 (Category)	BOOL	Zweite Kategorie (Per Voreinstellung lautet die zweite Kategorie = Warnung. Diese Einstellung können Sie jederzeit verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C2 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C3 (Category)	BOOL	Dritte Kategorie (Per Voreinstellung lautet die dritte Kategorie = Info. Diese Einstellung können Sie jederzeit verändern.) • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C3 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C4 (Category)	BOOL	Vierte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C4 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C5 (Category)	BOOL	Fünfte Kategorie

Name		Datentyp	Erläuterung
			• TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C5 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C6 (Category)	BOOL	Sechste Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C6 im aktuellen Programmzyklus ausgelöst. • FALSE =>Es liegt keine Störung vor.
	C7 (Category)	BOOL	Siebte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C7 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	C8 (Category)	BOOL	Achte Kategorie • TRUE => Es wurde mindestens eine Überwachungsmeldung der Kategorie C8 im aktuellen Programmzyklus ausgelöst. • FALSE => Es liegt keine Störung vor.
	Categories	BYTE	Sammelfehler-Byte für alle Kategorien Jedem einzelnen Bit ist eine Kategorie zugeordnet. Dadurch können Sie dieses Byte z. B. im Objekt "Pro-Diag-Übersicht" auf dem HMI-Gerät verwenden, um sich eine Anzeige zu erstellen, die Ihnen auf den ersten Blick anzeigt, wenn eine Überwachung einer bestimmten Kategorie ausgelöst wurde. • Bit 1 => Kategorie 1 (C1) • Bit 2 => Kategorie 2 (C2) • etc. Erreichbar aus HMI und sichtbar in HMI.
	Sub_Cat_1 (Sub-category 1)	DWORD	Sammelfehlerbits für die Unterkategorien 1 (Bit 1 bis 16)
	Sub_Cat_2 (Sub-category 2)	DWORD	Sammelfehlerbits für die Unterkategorien 2 (Bit 1 bis 16)
	Reserved	BYTE	Reserve-Byte

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Optionale Zustandsvariable, sobald eine Überwachung angelegt wurde

Sobald Sie eine Überwachung anlegen, wird in der Bausteinschnittstelle zusätzlich dieser Abschnitt angezeigt:

Name	Datentyp	Erläuterung
z. B. Conv_Push_Button_Forward_O_1 • Conv_Push_Button_Forward = Name des überwachten Operanden • O = Überwachungsart "Operand" • 1 = Anzahl der Überwachungen von der Überwachungsart "Operand"	SV_State_V2	Systeminterner, know-how-geschützter PLC-Datentyp

Name	Datentyp	Erläuterung
Err (Error)	BOOL	Fehler, der auch in den Eigenschaften der Überwachung angezeigt wird. Es zeigt an, ob eine Überwachungsmeldung ausgelöst wurde. • TRUE = Überwachungsmeldung wurde ausgelöst • FALSE = Keine Überwachungsmeldung, d. h. die Störung ist nicht eingetreten Erreichbar aus HMI und sichtbar in HMI
Err_Neg_Flag (Error_Negativ_Flag)	BOOL	Für die negative Flankenauswertung des Fehlermerkers
T_D_E_Going (Time_Date_Error_Going)	LDT	Uhrzeit und Tag im Programmzyklus, wann die Störung gegangen ist.
E_Flag (Error_Flag)	BOOL	Fehlermerker: Es wird nur eine Überwachungsmeldung pro ProDiag-Funktionsbaustein und pro Programmzyklus versendet.
E_Pos_Flag (Error_Positive_Flag)	BOOL	Für die positive Flankenauswertung des Fehlermerkers
T_D_E_Coming (Time_Date_Error_Coming)	LDT	Uhrzeit und Tag im Programmzyklus, wann die Störung aufgetreten ist.
S_Coming (Signal_Coming)	BOOL	Merker • TRUE = Die Überwachungsmeldung wurde störungsfrei versandt • FALSE = Es liegt eine Störung vor.
Pos_Flag (Positive_Flag)	BOOL	Dieser Merker wird nur für die Überwachungsart "Position" benötigt.
Ti (Timer)	TON_TIME	Diese Zeit wird für die Bildung der Verzögerungszeit benötigt.
PT (Preset Time)	TIME	Voreingestellter Zeitwert
ET (Elapsed Time)	TIME	Aktueller Zeitwert
IN (Input)	TIME	Starteingang
Q (Output)	TIME	Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird
Crit_Avail	BOOL	Der Signalzustand = TRUE: Die Erstwerte wurden aufgezeichnet.
Crit_Err	DWORD	Ablage der 32 Signalzustände pro Netzwerk eines KOP- oder FUP-Bausteins.

24.7 Eigenschaften eines ProDiag-Funktionsbausteins definieren

24.7.1 Aufbau der Überwachungseinstellungen am ProDiag-FB

Aufbau der Überwachungseinstellungen

Das folgende Bild zeigt die Komponenten der Überwachungseinstellungen:

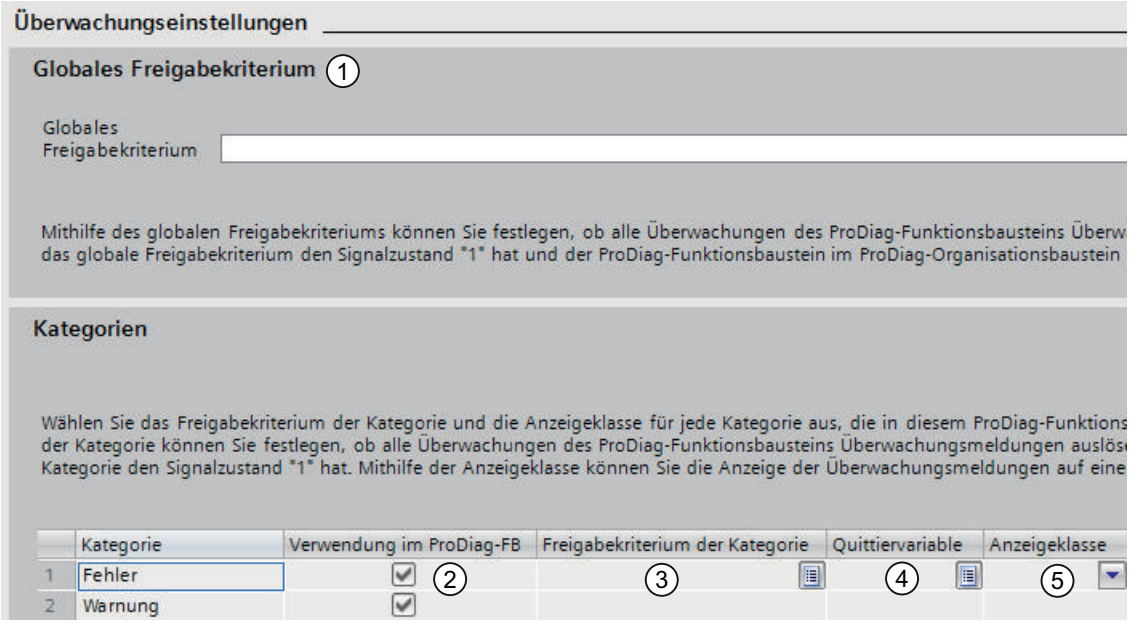

Nummerierung	Erklärung	Weiterführende Informationen
1	Globales Freigabekriterium eines ProDiag-Funktionsbausteins	Mithilfe des globalen Freigabekriteriums können Sie den Aufruf eines ProDiag- Funktionsbausteins im ProDiag-Organisationsbaustein steuern. Weitere Informationen finden Sie hier: Globales Freigabekriterium verwenden (Seite 8218)
2	Die Spalte "Verwendung im ProDiag-FB" ist nicht editierbar.	Wenn eine Überwachung dieser Kategorie dem ProDiag-Funktionsbaustein zugewiesen wurde, dann ist die Kategorie aktiviert.

Nummerierung	Erklärung	Weiterführende Informationen
3	Freigabekriterium der Kategorie	Mithilfe des Freigabekriteriums der Kategorie können Sie für den ProDiag-Funktionsbaustein und für jede einzelne darin enthaltene Kategorie ein Freigabekriterium (globale boolesche Variable) definieren und damit die Überwachungen dieser Kategorie aktivieren bzw. deaktivieren. Weitere Informationen finden Sie hier: Freigabekriterium der Kategorie verwenden (Seite 8218)
4	Quittiervariable	Mithilfe der Quittiervariable können Sie die Überwachungsmeldung einer Kategorie innerhalb des STEP 7-Projekts bestätigen. Damit stellen Sie sicher, dass die Überwachungsmeldung erst dann wieder geht, wenn die Störung behoben wurde. Mit dem entsprechenden Fehlermerker können Sie den Programmablauf steuern. Weitere Informationen finden Sie hier: Quittiervariable verwenden (Seite 8220)
5	Anzeigeklasse	Mithilfe der Anzeigeklasse können Sie definieren, welche Überwachungen an einem HMI-Gerät angezeigt werden sollen. Weitere Informationen finden Sie hier: Anzeigeklasse einstellen (Seite 8220)

24.7.2 Freigabekriterien für Überwachungsmeldungen anlegen

24.7.2.1 Überblick über die Freigabekriterien

Beschreibung

Sie haben die Möglichkeit, in den Einstellungen eines ProDiag-Funktionsbausteins Freigabekriterien (Boolesche Variablen) anzulegen. Dabei stehen Ihnen zwei verschiedene Arten von Freigabekriterien zur Verfügung:

- Globales Freigabekriterium
- Freigabekriterium der Kategorie

Globales Freigabekriterium

Mithilfe des globalen Freigabekriteriums können Sie den Aufruf der ProDiag-Funktionsbausteine im ProDiag-Organisationsbaustein steuern. Der ProDiag-OB wird automatisch angelegt, sobald Sie Ihr Projekt übersetzen. Sie können z. B. alle Überwachungen eines ProDiag-Funktionsbaustein abschalten, indem Sie ein globales Freigabekriterium definieren, das vom System automatisch als Aufrufbedingung verwendet wird, sobald der ProDiag-Funktionsbaustein im ProDiag-Organisationsbaustein aufgerufen wird. Ist die Aufrufbedingung nicht erfüllt (Signalzustand = False), dann wird der ProDiag-Funktionsbaustein nicht aufgerufen und es werden keine Überwachungsmeldungen ausgegeben. Das globale Freigabekriterium muss eine globale boolesche Variable sein.

Wenn Sie den ProDiag-Funktionsbaustein allerdings nicht im ProDiag-OB, sondern in einem anderen Baustein innerhalb Ihres Programms aufrufen, dann müssen Sie das globale Freigabekriterium selbst im Programmcode integrieren, wenn Sie es verwenden möchten.

Das globale Freigabekriterium bietet Ihnen die folgenden Vorteile:

- Sie können steuern, welcher ProDiag-Funktionsbaustein in welchem Programmzyklus bearbeitet werden soll. Alle nicht bearbeiteten ProDiag-Funktionsbausteine geben keine Überwachungsmeldungen aus, auch wenn eine Störung aufgetreten ist.
- Wenn Sie allen ProDiag-Funktionsbausteinen die gleiche Variable als Freigabekriterium zuweisen, dann können Sie z. B. ganz leicht programmieren, dass während der Inbetriebnahme keine unerwünschten Überwachungen ausgelöst werden sollen.
- Bei sehr kritischen Zykluszeiten können Sie Performance-Verbesserungen erzielen, indem Sie den ProDiag-Funktionsbausteinen verschiedene globale boolesche Variablen als Freigabekriterien zuordnen und damit die Bausteine im Programmcode gezielt aufrufen.
- Wenn bereits eine Überwachungsmeldung ausgelöst wurde, dann bleibt diese so lange bestehen, bis der entsprechende ProDiag-Funktionsbaustein wieder aufgerufen wird. Sobald der ProDiag-Funktionsbaustein wieder bearbeitet wird, wird die Überwachungsmeldung ausgegeben.

Freigabekriterium der Kategorie

Mithilfe des Freigabekriteriums der Kategorie können Sie für jeden ProDiag-Funktionsbaustein und für jede einzelne darin enthaltene Kategorie ein Freigabekriterium (globale boolesche Variable) definieren und damit die Überwachungen dieser Kategorie aktivieren bzw. deaktivieren. So können Sie z. B. alle Überwachungen der Kategorie "Fehler" eines ProDiag-Funktionsbausteins beeinflussen. Wenn Sie das Freigabekriterium der Kategorie als Aufrufbedingung innerhalb Ihres Programmcodes verwenden und die Aufrufbedingung ist zur Laufzeit nicht erfüllt (Signalzustand = False), dann werden alle Überwachungen dieser Kategorie nicht ausgeführt. Das bedeutet, dass die Sammelfehlerbits in der Bausteinschnittstelle des ProDiag-Funktionsbausteins nicht verändert und keine Überwachungsmeldungen ausgegeben werden.

Das Freigabekriterium der Kategorie bietet Ihnen die folgenden Vorteile:

- Sie können gezielt die Überwachungsmeldungen einer einzelnen Kategorie innerhalb eines ProDiag-Funktionsbausteins deaktivieren.
- Wenn bereits eine Überwachungsmeldung ausgelöst wurde, dann wird sie als "gegangen" zurück genommen. Solange die globale boolesche Variable den Signalzustand = False hat, werden, z. B. während Wartungsarbeiten, keine unnötigen Überwachungsmeldungen ausgegeben.

Siehe auch

Globales Freigabekriterium verwenden (Seite 8218)

Freigabekriterium der Kategorie verwenden (Seite 8218)

24.7.2.2 Globales Freigabekriterium verwenden

Voraussetzung

Sie haben einen ProDiag-Funktionsbaustein und eine Überwachung angelegt.

Vorgehen

Um ein globales Freigabekriterium zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie im Register "Allgemein" den Eintrag "Überwachungseinstellungen" an.
3. Geben Sie eine globale boolesche Variable als globales Freigabekriterium an.
   Damit das globale Freigabekriterium Einfluss auf den Ablauf des Programmcodes nehmen kann, dürfen Sie den ProDiag-Funktionsbaustein nicht in einem anwenderdefinierten Programmbaustein aufrufen. Dadurch wird der ProDiag-Funktionsbaustein automatisch im ProDiag-Organisationsbaustein aufgerufen.

Ergebnis

Es werden nur die Überwachungsmeldungen der ProDiag-Funktionsbausteine ausgegeben, deren globales Freigabekriterium den Signalzustand "1" hat.

24.7.2.3 Freigabekriterium der Kategorie verwenden

Voraussetzung

Sie haben einen ProDiag-Funktionsbaustein und eine Überwachung angelegt.

Vorgehen

Um ein Freigabekriterium für eine Kategorie zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbausteins wird geöffnet.
2. Klicken Sie im Register "Allgemein" den Eintrag "Überwachungseinstellungen" an.
3. Verwenden Sie im Bereich "Kategorien" für die einzelnen Kategorien ein Freigabekriterium. Damit das Freigabekriterium zusätzlich auch noch Einfluss auf den Ablauf des Programmcodes nehmen kann, müssen Sie es als Weiterschaltbedingung im Programmcode verwenden.

Ergebnis

Es werden nur die Überwachungsmeldungen der Kategorien des ProDiag-Funktionsbausteins ausgegeben, deren Freigabekriterium den Signalzustand "1" hat.

24.7.3 Quittiervariable für einzelne Kategorien definieren

24.7.3.1 Überblick über die Quittiervariablen

Beschreibung

Sie haben die Möglichkeit, in jedem ProDiag-Funktionsbaustein für jede einzelne Kategorie eine globale boolesche Variable als Quittiervariable zu verwenden.

Mithilfe der Quittiervariable bewirken Sie zunächst, dass der Fehlermerker und die Sammelfehlerbits den Signalzustand TRUE behalten und die Überwachungsmeldung nicht wieder geht. Damit auch z. B. das Förderband sofort angehalten wird, müssen Sie diese Fehlerbits explizit in Ihrem Programmcode verwenden. Dadurch haben Sie jetzt die Möglichkeit, dass das Förderband erst dann wieder anläuft, wenn der Überwachungsfehler explizit vom Maschinenbediener quittiert wurde, auch wenn die Ursache der Störung, z. B. eine durchbrochene Lichtschranke, bereits vorher behoben wurde. Im Vergleich dazu gibt es noch die Quittierung der Überwachungsmeldung auf einem HMI-Gerät, die aber keinen Einfluss auf die Bearbeitung des Programmcodes hat.

Sie können für jede der 8 Kategorien eine eigene Quittiervariable verwenden und diese in Ihren Programmcode als Weiterschaltbedingung für den weiteren Programmablauf mit einbauen.

Die Fehlerbits bleiben im ProDiag-Funktionsbaustein erhalten, nachdem die Störung behoben wurde, bis sich der Signalzustand der Quittiervariable von "0" auf "1" geändert hat. Erst, wenn Sie die Quittiervariable bestätigen, d. h. sich der Signalzustand von "0" auf "1" verändert, kann die Bearbeitung des Programmcodes fortgesetzt werden:

- Der Fehlermerker wird wieder auf den Signalzustand "0" zurückgesetzt.
- Die Sammelfehlerbits für HMI werden ebenfalls auf "0" zurückgesetzt.
- Die Überwachungsmeldung am HMI-Gerät verschwindet.

24.7.3.2 Quittiervariable verwenden

Voraussetzung

Sie haben einen ProDiag-Funktionsbaustein und eine Überwachung angelegt.

Vorgehen

Um eine Quittiervariable zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie im Register "Allgemein" den Eintrag "Überwachungseinstellungen" an.
3. Verwenden Sie im Bereich "Kategorien" für die einzelnen Kategorien Quittiervariablen.
   Damit die Quittiervariablen Einfluss auf den Ablauf des Programmcodes nehmen können, müssen Sie den Fehlermerker als Weiterschaltbedingung im Programmcode verwenden.

Ergebnis

Eine Störung ist aufgetreten und die Überwachungsmeldung wurde ausgelöst. Der Programmcode wird erst dann weiter bearbeitet, wenn die Störung behoben und die Quittiervariable bestätigt wurde.

24.7.4 Anzeigeklassen einstellen

24.7.4.1 Überblick über die Anzeigeklassen

Beschreibung

Sie haben die Möglichkeit, in jedem ProDiag-Funktionsbaustein für jede einzelne Kategorie eine Anzeigeklasse zu definieren.

Mithilfe der Anzeigeklassen können Sie definieren, welche Überwachungen an einem HMI-Gerät angezeigt werden sollen. Das ist vor allem dann sinnvoll, wenn mehrere HMI-Geräte an einer CPU angebunden sind. Damit die Überwachungen korrekt angezeigt werden können, müssen Sie bei jedem HMI-Gerät die Anzeigeklassen aktivieren, die angezeigt werden sollen.

Es sind maximal 17 Anzeigeklassen möglich.

24.7.4.2 Anzeigeklasse einstellen

Voraussetzung

Sie haben einen ProDiag-Funktionsbaustein und eine Überwachung angelegt.

Vorgehen

Um eine Anzeigeklasse einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie im Register "Allgemein" den Eintrag "Überwachungseinstellungen" an.
3. Stellen Sie im Bereich "Kategorien" für die einzelnen Kategorien die Anzeigeklassen ein.

Ergebnis

Wenn Sie z. B. der Kategorie "Fehler" die Anzeigeklasse "1" zugewiesen haben und diese Anzeigeklasse bei einem HMI-Gerät aktivieren, dann werden Ihnen auf diesem HMI-Gerät nur die Überwachungsmeldungen angezeigt, die der Kategorie "Fehler" zugeordnet sind.

24.7.5 Anzahl der Überwachungen eines ProDiag-FBs anzeigen

Beschreibung

Sie haben die Möglichkeit in den Eigenschaften eines ProDiag-Funktionsbausteins auszulesen wieviele ProDiag-Überwachungen diesem ProDiag-FB zugewiesen sind.

Vorgehen

Um die Anzahl der zugewiesenen ProDiag-Überwachungen auszulesen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften des ProDiag-Funktionsbausteins, indem Sie entweder mit der rechten Maustaste auf den ProDiag-FB klicken und im Kontextmenü "Eigenschaften..." auswählen oder klicken Sie doppelt auf den ProDiag-FB.
   Der Eigenschaftendialog des ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie das Register "Information" an.

Ergebnis

Die tatsächlich enthaltene Anzahl der möglichen ProDiag-Überwachungen wird angezeigt.

24.8 ProDiag-Überwachungseinstellungen festlegen

24.8.1 Überblick über die ProDiag-Überwachungseinstellungen

Beschreibung

Damit Sie die ProDiag-Überwachungseinstellungen editieren können, müssen Sie vorher in Ihrem Programm entweder einen GRAPH-Funktionsbaustein oder eine Überwachung angelegt haben.

In den ProDiag-Überwachungseinstellungen legen Sie die Einstellungen fest, die für alle Überwachungen im gesamten TIA Portal-Projekt gelten. Sie können Einstellungen hinzufügen und verändern. Die ProDiag-Überwachungseinstellungen finden Sie in der Projektnavigation unter "Gemeinsame Daten".

Siehe auch

Allgemeine ProDiag-Überwachungseinstellungen festlegen (Seite 8223)

Zentrale ProDiag-Meldetexte festlegen (Seite 8233)

24.8.2 Allgemeine ProDiag-Überwachungseinstellungen festlegen

24.8.2.1 Aufbau der ProDiag-Überwachungseinstellungen

Aufbau der ProDiag-Überwachungseinstellungen

Das folgende Bild zeigt exemplarisch die Komponenten der ProDiag-Überwachungseinstellungen:

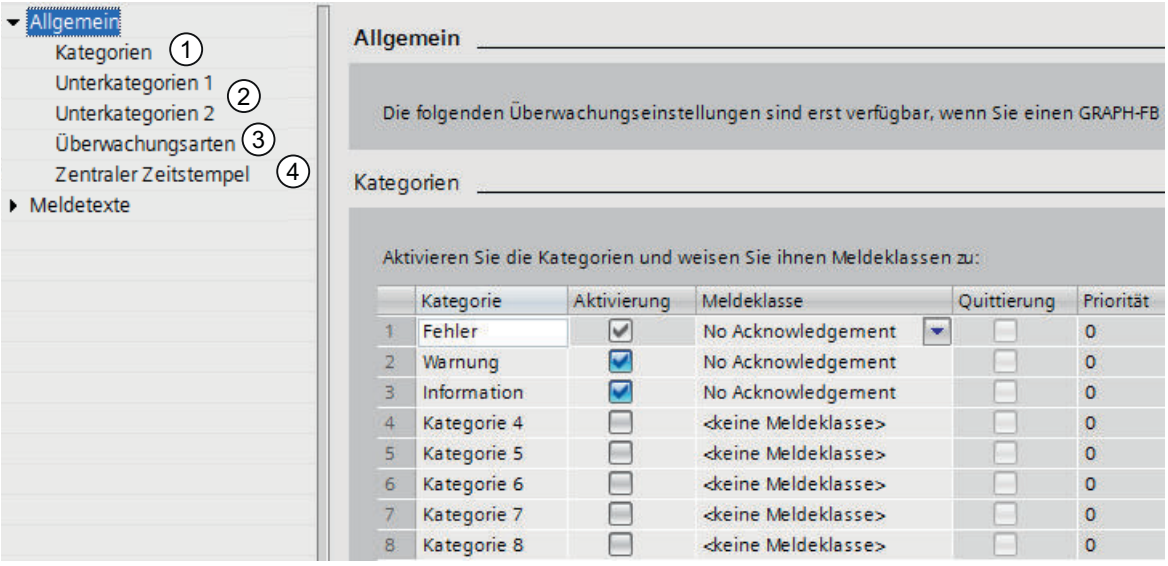

Nummerierung	Erklärung	Weiterführende Informationen
1	Es können 8 Kategorien definiert werden.	Mithilfe der Kategorien können Sie die auftretenden Störungen in unterschiedliche Dringlichkeitsstufen einteilen, wie z. B. Fehler, Warnung, etc. Weitere Informationen zu den Kategorien finden Sie hier: Kategorien (Seite 8224)
2	Es können 32 Unterkategorien definiert werden.	Mithilfe der Unterkategorien können Sie die Informationen innerhalb einer Überwachungsmeldung noch genauer spezifizieren. Weitere Informationen zu den Unterkategorien finden Sie hier: Unterkategorien (Seite 8227)

Nummerierung	Erklärung	Weiterführende Informationen
3	Es gibt 10 Überwachungsarten.	Sie können für jede Überwachungsart entweder den vorgegebenen Namen verwenden oder eigene definieren. Weitere Informationen zu den Überwachungsarten finden Sie hier: Überwachungsarten (Seite 8230)
4	Ein zentraler Zeitstempel für alle ProDiag-Funktionsbausteine im Projekt.	Die Zeitstempel-Variable wird in den Überwachungseinstellungen zentral für das gesamte Projekt vorgegeben und kann in jeder im Projekt verfügbaren CPU verwendet werden. Weitere Informationen zum zentralen Zeitstempel finden Sie hier: Zentralen Zeitstempel verwenden (Seite 8197)

24.8.2.2 Kategorien

Überblick über die Kategorien

Beschreibung

Mithilfe der Kategorien können Sie auftretende Störungen in unterschiedliche Dringlichkeitsstufen einteilen. Jeder Kategorie können Sie eine Meldeklasse zuweisen.

Die Kategorien können entweder als Bestandteil einer Überwachungsmeldung ausgegeben werden oder sie können als Sammelfehlerbit im Objekt "ProDiag-Übersicht" auf einem HMI-Gerät angezeigt werden.

Das folgende Bild zeigt den Abschnitt "Kategorien" in den ProDiag-Überwachungseinstellungen mit voreingestellten Werten:

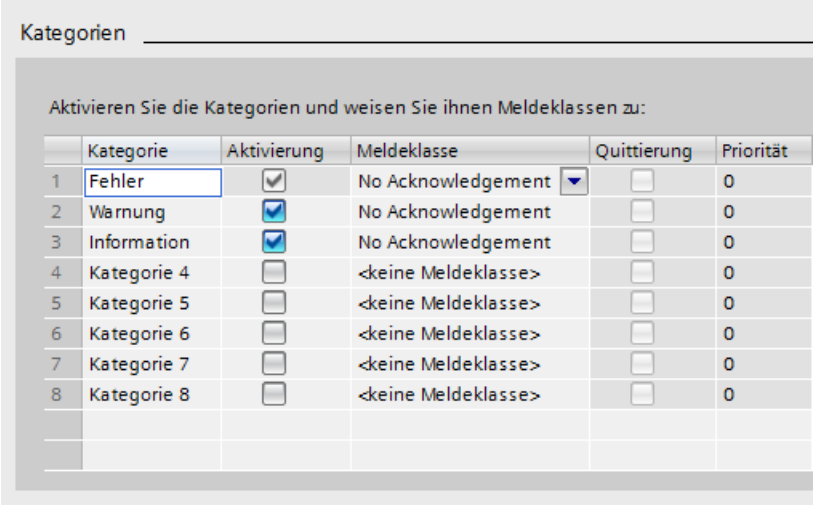

	Kategorie	Aktivierung	Meldeklasse	Quittierung	Priorität
1	Fehler	☑	No Acknowledgement	☐	0
2	Warnung	☑	No Acknowledgement	☐	0
3	Information	☑	No Acknowledgement	☐	0
4	Kategorie 4	☐	<keine Meldeklasse>	☐	0
5	Kategorie 5	☐	<keine Meldeklasse>	☐	0
6	Kategorie 6	☐	<keine Meldeklasse>	☐	0
7	Kategorie 7	☐	<keine Meldeklasse>	☐	0
8	Kategorie 8	☐	<keine Meldeklasse>	☐	0

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Meldeklasse

Wenn Sie den Kategorien unterschiedliche Meldeklassen zuweisen, dann können Sie sich, entsprechend der definierten Meldeklassen, in der Meldeanzeige auf Ihrem HMI-Gerät unterschiedliche Bereiche anlegen.

Wenn Sie den Meldeklassen zusätzlich noch unterschiedliche Prioritäten zuordnen, dann können Sie damit die Reihenfolge der Überwachungsmeldungen im Objekt "Meldeanzeige" steuern. Für jede Meldeklasse können Sie noch festlegen, ob die Überwachungsmeldung mit oder ohne Quittierungspflicht ausgegeben werden soll.

Siehe auch

Meldeklassen anlegen (Seite 7853)

Kategorien definieren

Sie können jederzeit die Einstellungen ("Kategorie", "Aktivierung" oder "Meldeklasse") einer Kategorie ändern, auch wenn diese Kategorie bereits einer Überwachung zugeordnet wurde. Die Änderungen werden automatisch in der Überwachung übernommen.

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um eine Kategorie anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Allgemein" den Abschnitt "Kategorien" aus.
   Per Voreinstellung gibt es bereits die drei Kategorien "Fehler", "Warnung" und "Information". Zusätzlich gibt es noch fünf weitere Platzhalter-Kategorien. Die Namen der Kategorien können Sie verändern und übersetzen. Weitere Informationen zur Übersetzung der Namen finden Sie hier: Übersetzungen der Kategorienamen anzeigen (Seite 8226)

3. Aktivieren bzw. deaktivieren Sie die Kategorie in der Spalte "Aktivierung".

Hinweis

Kategorie "Fehler"

Die voreingestellte Kategorie "Fehler" kann nicht deaktiviert werden. Sie bleibt immer aktiviert.

4. Per Voreinstellung wird in der Spalte "Meldeklasse" die systemeigene Meldeklasse "No Acknowledgement" eingestellt. Neben der systemeigenen Meldeklasse "No Acknowledgement" (= Nicht quittierpflichtig) stehen Ihnen auch die Meldeklasse "Acknowledgement" (= quittierpflichtig) oder selbstdefinierte Meldeklassen zur Verfügung. Passen Sie ggf. die Voreinstellungen an.

Hinweis

Meldeklasse "Acknowledgement"

Wenn Sie die Meldeklasse "Acknowledgement" auswählen, dann erscheint auf Ihrem HMI-Gerät eine quittierpflichtige Überwachungsmeldung. Diese Quittierung hat keinen Einfluss auf den Fortgang Ihres Programms. Wenn Sie den Fortgang Ihres Programms nach einer aufgetretenen Störung beeinflussen möchten, dann müssen Sie die Quittierung an einer Kategorie mithilfe der Quittierungsvariable verwenden. Diese Quittierung können Sie in den Bausteineigenschaften des ProDiag-Funktionsbausteins einstellen.

Die Spalten "Quittierung" und "Priorität" sind nicht editierbar, da diese Einstellungen der Meldeklasse nur im Meldeklasseneditor geändert werden können. Weitere Information zum Erstellen von Meldeklassen finden Sie hier: Meldeklassen anlegen (Seite 7853)

5. Speichern Sie das Projekt.

Übersetzungen der Kategorienamen anzeigen

Sie können sich die Namen aller Kategorien in allen Projektsprachen anzeigen lassen, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben und deren Übersetzungen bereits vorliegen. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um sich den Namen einer Kategorie in allen Projektsprachen anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Allgemein" den Abschnitt "Kategorien" aus.

3. Markieren Sie die Zeile mit der Kategorie, deren Übersetzungen Sie sich anzeigen lassen wollen.
   Wenn Sie sich die Übersetzungen mehrerer Kategorien gleichzeitig anzeigen lassen wollen, dann müssen Sie die entsprechenden Zeilen markieren.
4. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
   Wenn Ihnen die Namen der Kategorien nur in der aktuellen Oberflächensprache angezeigt werden, dann haben Sie noch keine Projektsprachen ausgewählt bzw. es sind noch keine Übersetzungen vorhanden. Sie können die Übersetzungen entweder direkt in die Tabelle im Register "Texte" eintragen oder Sie können die Übersetzungen für die einzelnen Projektsprachen in die globale Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

Ergebnis

Im untergeordneten Register "Texte" werden Ihnen die markierten Zeilen in den gewünschten Projektsprachen und, soweit vorhanden, die Übersetzungen der Kategorienamen angezeigt.

Texte

Deutsch (Deu...	Chinesisch (V...	Englisch (USA)	Französisch (Frankreich)	Italienisch (Italien)	Spanisch (Spanien)
Fehler	错误	Error	Erreur	Errore	Error
Warnung	警告	Warning	Avertissement	Avviso	Advertencia
Information	Info	Info	Information	Informazione	Información
Kategorie 4	类别 4	Category 4	Catégorie 4	Categoria 4	Categoría 4
Kategorie 5	类别 5	Category 5	Catégorie 5	Categoria 5	Categoría 5
Kategorie 6	类别 6	Category 6	Catégorie 6	Categoria 6	Categoría 6
Kategorie 7	类别 7	Category 7	Catégorie 7	Categoria 7	Categoría 7
Kategorie 8	类别 8	Category 8	Catégorie 8	Categoria 8	Categoría 8

24.8.2.3 Unterkategorien

Überblick über die Unterkategorien

Beschreibung

Zusätzlich zu den 16 Unterkategorien der "Unterkategorie 1" können Sie noch 16 Unterkategorien der "Unterkategorie 2" aktivieren. Die Kategorien und Unterkategorien sind unabhängig voneinander und können innerhalb einer Überwachung frei miteinander kombiniert werden.

Mithilfe der Unterkategorien haben Sie die Möglichkeit, die Informationen innerhalb einer Überwachungsmeldung noch genauer zu spezifizieren. Wenn Sie z. B. bereits die Kategorien "Fehler", "Warnung" und "Information" angelegt haben, können Sie diese mit den Unterkategorien "technisch" oder "organisatorisch" erweitern. So erhalten Sie detailliertere Informationen zu einer aufgetretenen Störung.

Die Unterkategorien können entweder als Bestandteil einer Überwachungsmeldung ausgegeben werden oder sie können als Sammelfehlerbit im Objekt "ProDiag-Übersicht" auf einem HMI-Gerät angezeigt werden.

Das folgende Bild zeigt den Abschnitt "Unterkategorien 1" in den ProDiag-Überwachungseinstellungen:

Definieren und aktivieren Sie die Unterkategorien und deren Beschreibungen:

	Unterkategorie 1	Aktivierung	Beschreibung
1	Unterkategorie 1	☑	
2	Unterkategorie 2	☑	
3	Unterkategorie 3	☐	

Hinweis

Weitere Informationen zum HMI-Objekt "ProDiag-Übersicht"

Weitere Informationen finden Sie im Informationssystem des TIA-Portals hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen > Überwachungen visualisieren > ProDiag-Übersicht projektieren"

Unterkategorien definieren

Sie können jederzeit die Einstellungen ("Unterkategorie x", "Aktivierung" oder "Beschreibung") einer Unterkategorie ändern, auch wenn diese Unterkategorie bereits einer Überwachung zugeordnet ist. Die Änderungen werden automatisch in der Überwachung übernommen.

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um eine Unterkategorie zu aktivieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Allgemein" entweder den Abschnitt "Unterkategorien 1" oder "Unterkategorien 2" aus.
3. Geben Sie den gewünschten Unterkategorien Namen.
   Die Namen der Unterkategorien sind übersetzbar. Weitere Informationen zur Übersetzung der Namen finden Sie hier: Übersetzungen der Unterkategorienamen und Beschreibungstexte anzeigen und bearbeiten (Seite 8229)
4. Aktivieren bzw. deaktivieren Sie die gewünschten Unterkategorien in der Spalte "Aktivierung".
5. Geben Sie ggf. einen Text in der Spalte "Beschreibung" ein.
   Mithilfe der Spalte "Beschreibung" können Sie Informationen an den Programmierer weitergeben, der das STEP 7-Projekt erstellt. Diese Informationen stehen nur innerhalb des TIA Portals zur Verfügung und werden nicht an Runtime Professional weitergegeben. Zusätzlich zu den Namen der Unterkategorien werden Ihnen auch die Beschreibungstexte im Inspektorfenster "Eigenschaften" im untergeordneten Register "Texte" angezeigt.
6. Speichern Sie das Projekt.

Ergebnis

Es wird eine neue Unterkategorie 1 angelegt, die Sie bei der Erstellung einer Überwachung mit einer Kategorie verknüpfen können.

Definieren und aktivieren Sie die Unterkategorien und deren Beschreibungen:

	Unterkategorie 1	Aktivierung	Beschreibung
1	Technisch	☑	Technischer Fehler
2	Organisatorisch	☑	Organisatorischer Fehler

Übersetzungen der Unterkategorienamen und Beschreibungstexte anzeigen und bearbeiten

Sie können sich die Namen aller Kategorien in allen Projektsprachen anzeigen lassen, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben und deren Übersetzungen bereits vorliegen. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um sich den Namen einer Unterkategorie in allen Projektsprachen anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Allgemein" den Abschnitt "Unterkategorien x" aus.
3. Markieren Sie die Zeile mit der Unterkategorie, deren Übersetzungen Sie sich anzeigen lassen wollen.
   Wenn Sie sich die Übersetzungen mehrerer Unterkategorien gleichzeitig anzeigen lassen wollen, dann müssen Sie die entsprechenden Zeilen markieren.
4. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
   Wenn Ihnen die Namen der Unterkategorien nur in der aktuellen Oberflächensprache angezeigt werden, dann haben Sie noch keine Projektsprachen ausgewählt bzw. es sind noch keine Übersetzungen vorhanden. Sie können die Übersetzungen entweder direkt in die Tabelle im Register "Texte" eintragen oder Sie können die Übersetzungen für die einzelnen Projektsprachen in die globale Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

Ergebnis

Im untergeordneten Register "Texte" werden Ihnen die markierten Zeilen in den gewünschten Projektsprachen und, soweit vorhanden, die Übersetzungen der Unterkategorienamen und deren Beschreibungstexte angezeigt.

Texte

Deutsch (Deutschland)	Englisch (USA)	Französisch (Frankreich)	Italienisch (Italien)
Technisch	Technical	Technique	Tecnico
Technischer Fehler	Technical error	Erreur technique	Errore tecnico
Organisatorisch	Organizational	Organisation	Organizzativo
Organisatorischer Fehler	Organizational error	Erreur d'organisation	Errore organizzativo

24.8.2.4 Überwachungsarten

Überblick über die Überwachungsarten

Beschreibung

Es gibt zehn Überwachungsarten mit voreingestellten Namen. Die Namen sind schreibgeschützt und die Übersetzungen werden in den ausgewählten Oberflächensprachen vom System zur Verfügung gestellt. Sie können zusätzlich für jede Überwachungsart einen eigenen Anzeigenamen der Überwachungsart im Meldetext definieren und diesen in alle Oberflächensprachen übersetzen. Sie können für jede Überwachungsart innerhalb Ihres Projekts Voreinstellungen festlegen und damit definieren, auf welchen Signalzustand ein Operand oder eine Bedingung hin überwacht werden soll. Die Voreinstellungen gelten jeweils für neu erstellte Überwachungen oder, wenn Sie die Überwachungsart ändern. Zusätzlich können Sie auch eine Verzögerungszeit für jede Überwachungsart voreinstellen.

Das folgende Bild zeigt den Abschnitt "Überwachungsarten" in den ProDiag-Überwachungseinstellungen mit voreingestellten Werten:

Überwachungsarten

Legen Sie die Voreinstellungen für jede Überwachungsart innerhalb Ihres Projekts fest. Diese Voreinstellunge
Wenn Sie das Optionskästchen aktivieren, dann wird auf den Signalzustand TRUE überwacht.
Sie können die Voreinstellungen jeder neuen Überwachung anpassen. Ihre Änderungen beeinflussen die Mel

	Überwachungsart	Trigger	B1 Trigger	B2 Trigger	B3 Trigger	Verzögerungszeit
1	Operand	☐	☑	☑	☑	T#0ms
2	Verriegelung	☐	☑	☑	☑	T#0ms
3	Reaktion	☐	☑	☑	☑	T#5s
4	Aktion	☑	☑	☑	☑	T#200ms
5	Position	☐	☐	☑	☑	T#0ms
6	Fehlermeldung	☑	☑	☑	☑	T#0ms
7	Textmeldung	☑	☑	☑	☑	T#0ms
8	GRAPH-Interlock	☐	☐	☐	☐	T#0ms
9	GRAPH-Supervision	☐	☐	☐	☐	T#0ms
10	GRAPH-Warnung	☐	☐	☐	☐	T#0ms

Aktivierung

Beim Anlegen eines GRAPH-Funktionsbausteins oder einer Überwachung werden, abgesehen von den GRAPH-Einstellungen, automatisch alle Optionen per Voreinstellung vom System aktiviert. Sie können die Voreinstellungen für die Trigger und die Verzögerungszeit jederzeit verändern, indem Sie die jeweiligen Spalten aktivieren oder deaktivieren. Wenn Sie den Haken setzen, dann wird der Zustand des Operanden auf den Signalzustand TRUE überwacht. Wenn Sie den Haken entfernen, wird der Zustand des Operanden auf den Signalzustand FALSE überwacht.

Übersetzungen der Namen der Überwachungsarten anzeigen

Sie können sich die Namen aller Überwachungsarten in allen Projektsprachen anzeigen lassen, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben und deren Übersetzungen bereits vorliegen. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter "Projektdaten bearbeiten > Mit mehrsprachigen Projekten arbeiten > Projektsprachen auswählen".

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um sich den Namen einer Überwachungsart in allen Projektsprachen anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Allgemein" den Abschnitt "Überwachungsarten" aus.
3. Markieren Sie die Zeile mit der Überwachungsart, deren Übersetzungen Sie sich anzeigen lassen wollen.
   Wenn Sie sich die Übersetzungen mehrerer Überwachungsarten gleichzeitig anzeigen lassen wollen, dann müssen Sie die entsprechenden Zeilen markieren.
4. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
   Solange Sie die Namen der Überwachungsarten nicht in allen ausgewählten Projektsprachen übersetzt haben, werden Ihnen die Texte per Voreinstellung in der aktuellen Oberflächensprache angezeigt. Wenn die Übersetzungen noch nicht vorhanden sind, dann können Sie diese entweder direkt in die Tabelle eintragen oder Sie können die Übersetzungen für die einzelnen Projektsprachen in die globale Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Weitere Informationen zum Übersetzen von Texten finden Sie unter "Projektdaten bearbeiten > Mit mehrsprachigen Projekten arbeiten > Grundlagen zu Projekttexten".

Ergebnis

Im untergeordneten Register "Texte" werden Ihnen die gewünschten Projektsprachen und, soweit vorhanden, die Übersetzungen der Überwachungsartennamen angezeigt.

Texte

Deutsch (Deutschland)	Englisch (USA)	Französisch (Frankreich)	Italienisch (Italien)	Spanisch (Spanien)
Operand	Operand	Opérande	Operando	Operando
Verriegelung	Interlock	Verrouillage	Interblocco	Enclavamiento
Reaktion	Reaction	Réaction	Reazione	Reacción
Aktion	Action	Action	Azione	Acción
Position	Position	Position	Posizione	Posición
Fehlermeldung	Error message	Message d'erreur	Messaggio di errore	Mensaje de error
Textmeldung	Text message	Message texte	Messaggio di testo	Texto de aviso
GRAPH-Interlock	GRAPH-Interlock	GRAPH-Interlock	GRAPH-Interlock	GRAPH-Interlock
GRAPH-Supervision	GRAPH-Supervision	GRAPH-Supervision	GRAPH-Supervision	GRAPH-Supervision
GRAPH-Warnung	GRAPH-Warning	GRAPH-Avertissement	GRAPH-Avviso	GRAPH-Advertencia

24.8.3 Zentrale ProDiag-Meldetexte festlegen

24.8.3.1 Aufbau des Meldetexteditors

Aufbau des Meldetexteditors

Das folgende Bild zeigt exemplarisch die Komponenten des Meldetexteditors:

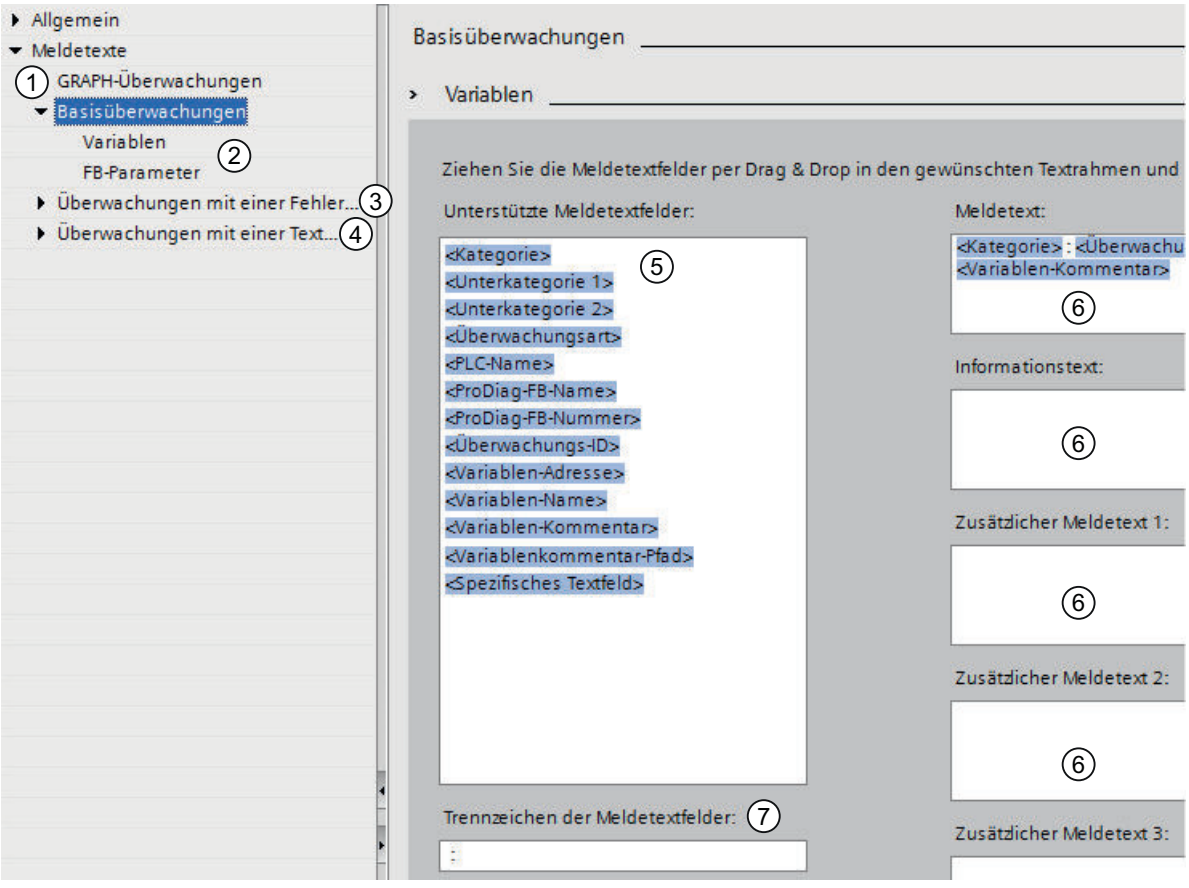

Nummerierung	Erklärung	Weiterführende Informationen
1	Anlegen von Meldetextstrukturen für GRAPH-Überwachungen (Nur in Verbindung mit der Programmiersprache GRAPH)	Weitere Informationen zu den GRAPH-Überwachungsmeldungen finden Sie hier: Aufbau einer GRAPH-Überwachungsmeldung (Seite 8236)
2	Anlegen von Meldetextstrukturen für die Basisüberwachungen: • Operand • Verriegelung • Aktion • Reaktion • Position Für jede Basisüberwachung können Sie eine eigene Meldetextstruktur für globale Überwachungen (Variablen) und lokale Überwachungen (FB-Parameter) anlegen.	Weitere Informationen zu den globalen Überwachungsmeldungen finden Sie hier: Aufbau einer globalen Überwachungsmeldung (Variable) (Seite 8237) Weitere Informationen zu den lokalen Überwachungsmeldungen finden Sie hier: Aufbau einer lokalen Überwachungsmeldung (Parameter) (Seite 8238)
3	Anlegen von Meldetextstrukturen für Überwachungen mit einer Fehlermeldung. Für jede Überwachung mit einer Fehlermeldung können Sie jeweils eine Meldetextstruktur für alle globalen Überwachungen (Variablen) und eine für alle lokalen Überwachungen (FB-Parameter) anlegen.	
4	Anlegen von Meldetextstrukturen für Überwachungen mit einer Textmeldung. Für jede Überwachung mit einer Textmeldung können Sie jeweils eine Meldetextstruktur für alle globalen Überwachungen (Variablen) und eine für alle lokalen Überwachungen (FB-Parameter) anlegen.	
5	Auflistung aller unterstützter Meldetextfelder, die Sie per Drag & Drop in die einzelnen Textrahmen ziehen können.	

Nummerierung	Erklärung	Weiterführende Informationen
6	Verschiedene Textrahmen, die Sie mit Meldetextfeldern befüllen können um die gewünschten Überwachungsmeldungen zu konfigurieren.	Sie können bis zu 11 Textrahmen definieren (Meldetext, Informationstext und neun zusätzliche Meldetexte). Die neun zusätzlichen Meldetexte können nur in WinCC Runtime Professional angezeigt werden.
7	Trennzeichen der Meldetextfelder	Per Voreinstellung wird zur Trennung der Meldetextfelder innerhalb eines Textrahmens ein Doppelpunkt verwendet. Sie können das Trennzeichen jederzeit verändern bzw. löschen, wenn Sie kein Trennzeichen verwenden wollen. Das zuvor verwendete Trennzeichen, z. B. der Doppelpunkt, wird dann automatisch in allen Textrahmen durch das neue Trennzeichen, z. B. ein Semikolon, ersetzt.

Beschreibung

Für den Textrahmen "Meldetext" wird Ihnen eine Textfeldstruktur vorgegeben, die Sie jederzeit verändern können.

Aus dem Bereich "Unterstützte Meldetextfelder" können Sie die jeweils zur Verfügung stehenden Meldetextfelder in einen der Textrahmen ziehen oder kopieren. Jedes Meldetextfeld kann innerhalb eines Textrahmens nur einmal verwendet werden. Welche Meldetextfelder Ihnen zur Auswahl stehen, hängt davon ab, welchen Bereich Sie in der Navigation unterhalb von "Meldetexte" angeklickt haben. Die Reihenfolge der Meldetextfelder können Sie innerhalb eines Textrahmens per Drag & Drop verändern.

Wenn Sie das Meldetextfeld "<Spezifisches Textfeld>" verwenden, dann können Sie sich einen spezifischen Text zu dieser Überwachung in der Überwachungsmeldung anzeigen lassen. Damit das Meldetextfeld die gewünschten Informationen anzeigen kann, müssen Sie in den Eigenschaften der einzelnen Überwachungen im untergeordneten Register "Spezifisches Textfeld" Informationen für diese Überwachung parametrieren.

Siehe auch

Spezifisches Textfeld anlegen (Seite 8255)

24.8.3.2 Aufbau einer GRAPH-Überwachungsmeldung

Beschreibung

Diese Meldetextstruktur für eine GRAPH-Überwachung gilt für alle drei GRAPH-Überwachungsarten:

- GRAPH-Interlock
- GRAPH-Supervision
- GRAPH-Warnung (Schrittzeit-Überwachung)

Die nachfolgende Tabelle zeigt Ihnen alle Bestandteile, die eine GRAPH-Überwachung enthalten kann. Sie können die Anzahl der Bestandteile jederzeit verringern:

Bestandteile einer GRAPH-Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Kategorie>	Name der Kategorie	x	Ja
<Unterkategorie 1>	Name der Unterkategorie 1	-	
<Unterkategorie 2>	Name der Unterkategorie 2	-	
<Überwachungsart>	Name der Überwachungsart	x	
<PLC-Name>	Name der PLC	x	Nein
<FB-Name>	Name des Funktionsbausteins	x	
<FB-Nummer>	Nummer des Funktionsbausteins	-	
<Schritt-Name>	Schrittname	x	
<Schritt-Nummer>	Schrittnummer	x	
<Instanz-DB-Name>	Name des Instanz-Datenbausteins	-	
<Instanz-DB-Nummer>	Nummer des Instanz-Datenbausteins	-	
<Schrittspezifisches Textfeld>	Spezifische Interlock- oder Supervision-Information, die Sie am jeweiligen Schritt definiert haben.	x	

Sie können die Anzahl der Bestandteile einer Überwachungsmeldung jederzeit verändern, indem Sie neue Bestandteile hinzufügen oder bereits enthaltene wieder entfernen. Alle Bestandteile können optional verwendet werden.

Siehe auch

Meldetexte anlegen (Seite 8240)

Meldungen bearbeiten (Seite 7806)

Meldungsanzeige aktivieren und deaktivieren (Seite 7806)

24.8.3.3 Aufbau einer globalen Überwachungsmeldung (Variable)

Beschreibung

Globale Überwachungsmeldungen können Sie für Überwachungen einer Variablen anlegen, die in einer Variablentabelle oder einem Datenbaustein definiert wurde.

Die nachfolgende Tabelle zeigt Ihnen alle Bestandteile, die eine globale Überwachungsmeldung enthalten kann. Sie können die Anzahl der Bestandteile jederzeit verringern:

Bestandteile einer globalen Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Kategorie>	Name der Kategorie	x	Ja
<Unterkategorie 1>	Name der Unterkategorie 1	-	
<Unterkategorie 2>	Name der Unterkategorie 2	-	Nein
<Überwachungsart>	Name der Überwachungsart	x	Ja
<PLC-Name>	Name der PLC	-	Nein
<ProDiag-FB-Name>	Name des ProDiag-Funktionsbausteins	x	
<ProDiag-FB-Nummer>	Nummer des ProDiag-Funktionsbausteins	-	
<Überwachungs-ID>	ID der Überwachung	x	
<Variablen-Adresse>	• Variablenadresse aus der Variablentabelle • Bei booleschen Elementen aus einem Datenbaustein wird keine Adresse ausgegeben.	x	
<Variablen-Name>	Name der überwachten Variable Wenn es sich um ein Element einer Struktur (PLC-Datentyp, ARRAY oder STRUCT) handelt, dann werden alle übergeordneten Strukturebenen angezeigt, z. B. "Motoren".Motor_1.Element_BOOL	x	

Bestandteile einer globalen Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Variablen-Kommentar>	• Kommentar der Variable aus der Variablentabelle • Kommentar des Elements aus dem Datenbaustein	x	Ja
<Variablenkommentar-Pfad>	Kommentare der Variablen aller Ebenen einer strukturierten Variable, getrennt durch Punkte	-	
<Spezifisches Textfeld>	Spezifische Information für eine Überwachung (Text mit 3 Variablen): • <Begleitwert SD_5> • <Begleitwert SD_6> • <Begleitwert SD_7>	-	

Sie können die Anzahl der Bestandteile einer Überwachungsmeldung jederzeit verändern, indem Sie neue Bestandteile hinzufügen oder bereits enthaltene wieder entfernen. Alle Bestandteile können optional verwendet werden.

Siehe auch

Meldetexte anlegen (Seite 8240)

24.8.3.4 Aufbau einer lokalen Überwachungsmeldung (Parameter)

Beschreibung

Lokale Überwachungsmeldungen können Sie für Überwachungen eines Parameters anlegen, der in der Bausteinschnittstelle eines Funktionsbausteins definiert ist.

Hinweis

Strukturierte Variablen

In der Bausteinschnittstelle können keine booleschen Elemente einer strukturierten Variablen (PLC-Datentyp, ARRAY oder STRUCT) überwacht werden.

Die nachfolgende Tabelle zeigt Ihnen alle Bestandteile, die eine lokale Überwachungsmeldung enthalten kann. Sie können die Anzahl der Bestandteile jederzeit verringern:

Bestandteile einer lokalen Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Kategorie>	Name der Kategorie	x	Ja
<Unterkategorie 1>	Name der Unterkategorie 1	-	

Bestandteile einer lokalen Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Unterkategorie 2>	Name der Unterkategorie 2	-	Nein
<Überwachungsart>	Name der Überwachungsart	x	Ja
<PLC-Name>	Name der CPU	-	Nein
<ProDiag-FB-Name>	Name des ProDiag-Funktionsbausteins	x	
<ProDiag-FB-Nummer>	Nummer des ProDiag-Funktionsbausteins	-	
<Überwachungs-ID>	ID der Überwachung	x	
<FB-Name>	Name des Funktionsbausteins	-	
<Instanz-Pfad>	Name und Pfad des Instanz-Datenbausteins	-	
<Instanz-Name>	Name des Instanz-Datenbausteins (z. B. Block_9_DB)	x	
<Instanz-Nummer>	Nummer des Instanz-Datenbausteins	-	
<Instanz-Kommentar>	• Bei einer Einzelinstanz: Es wird der Titel des Instanz-DBs ausgegeben. • Bei einer Multiinstanz: Es wird der Kommentar der Multiinstanz aus der Bausteinschnittstelle ausgegeben.	-	Ja
<Parameter-Name>	Name des überwachten Parameters	x	Nein
<Parameter-Kommentar>	Parameterkommentar	-	Ja

Bestandteile einer lokalen Überwachungsmeldung	Beschreibung	Per Voreinstellung angegeben	Übersetzungsrelevant
<Variablen-Adresse>	• Variablenadresse aus der Variablentabelle • Bei booleschen Elementen aus einem Datenbaustein wird keine Adresse ausgegeben.	x	Nein
<Variablen-Name>	In diesem Fall wird der Parameter (<Parameter-Name>) überwacht und die Variable (<Variablen-Name>) ist die vorverschaltete Variable, falls es eine gibt.	x	
<Variablen-Kommentar>	• Kommentar der Variable aus der Variablentabelle • Kommentar des Elements aus dem Datenbaustein	x	Ja
<Variablenkommentar-Pfad>	Kommentare der Variablen aller Ebenen einer strukturierten Variable, getrennt durch Punkte	-	
<Spezifisches Textfeld>	Spezifische Information für eine Überwachung (Text mit 3 Variablen): • <Begleitwert SD_5> • <Begleitwert SD_6> • <Begleitwert SD_7>	-	

Sie können die Anzahl der Bestandteile einer Überwachungsmeldung jederzeit verändern, indem Sie neue Bestandteile hinzufügen oder bereits enthaltene wieder entfernen. Alle Bestandteile können optional verwendet werden.

Siehe auch

Meldetexte anlegen (Seite 8240)

24.8.3.5 Meldetexte anlegen

Voraussetzung

Sie haben in der Projektnavigation den Ordner "Gemeinsame Daten" geöffnet.

Vorgehen

Um einen Meldetext anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "Überwachungseinstellungen". Die ProDiag-Überwachungseinstellungen werden geöffnet.
2. Wählen Sie im Register "Meldetexte" den gewünschten Bereich aus, für den Sie einen Meldetext anlegen möchten, z. B. Basisüberwachungen > Variablen.
3. Ziehen Sie per Drag & Drop die gewünschten Meldetextfelder aus dem Feld "Unterstützte Meldetextfelder" in den Textrahmen "Meldetext".
   Die Reihenfolge der einzelnen Textfelder können Sie jederzeit verändern.
4. Wenn Sie ein Textfeld wieder löschen möchten, dann klicken Sie auf das entsprechende Textfeld mit einem rechten Mausklick und wählen Sie aus dem Kontextmenü "Löschen" aus.
5. Wählen Sie ein gewünschtes Trennzeichen für die Textfelder aus.

Ergebnis

Die Textfelder dienen als Platzhalter und sobald eine Überwachungsmeldung ausgelöst wird, werden die Textfelder mit aktuellen Prozesswerten befüllt.

24.8.3.6 Export von ProDiag- und GRAPH-Texten

Sie können Projekttexte von ProDiag und GRAPH filtern und zur Übersetzung exportieren. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann in Microsoft Excel oder in vielen anderen Tabellenkalkulationsprogrammen bearbeitet werden.

Den Export von ProDiag- und GRAPH-Texten definieren Sie durch folgende Filteroptionen:

Option	Inhalt des Exports
ProDiag / GRAPH: Meldetexte	• Anwenderdefinierte Textlisten für Überwachungen • Anzeigenamen für Schritte und Transitionen in GRAPH • Kommentare und Kommentarpfade der überwachten Variablen • Kommentare der überwachten Elemente in der Bausteinschnittstelle • Titel der überwachten Instanz-Datenbausteine und Instanzkommentare im Fall einer Multiinstanz • Kommentare und Kommentarpfade der Variablen, die mit überwachten Elementen der Bausteinschnittstelle vorverschaltet sind (falls nur ein Operand vorverschaltet) • Schrittspezifische Interlock- und Überwachungs-Meldetexte für GRAPH
ProDiag / GRAPH: Texte für HMI-Anzeige	• Variablenkommentare in Transitions- und Interlock-Netzwerken • Variablenkommentare in mit ProDiag überwachten permanenten Netzwerken • Variablenkommentare in mit ProDiag überwachten Netzwerken • Titel der mit ProDiag überwachten Netzwerken (inkl. permanenter Netzwerke)
ProDiag / GRAPH: Alle Texte	• ProDiag / GRAPH Meldetexte • ProDiag / GRAPH Texte für HMI-Anzeige

Hinweis

Die genannten Filteroptionen sind nicht für CPU S7-300/400 verfügbar.

Weitere Informationen finden Sie im Kapitel "Projekttexte exportieren".

24.8.4 ProDiag-Überwachungseinstellungen exportieren

Sie können die gesamten ProDiag-Überwachungseinstellungen in eine .dat-Datei exportieren, um sie in ein neues Projekt zu importieren.

Voraussetzung

Sie haben die ProDiag-Überwachungseinstellungen für ein Projekt definiert.

Vorgehen

Um die ProDiag-Überwachungseinstellungen zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten".
2. Doppelklicken Sie den Eintrag "Überwachungseinstellungen".

3. Klicken Sie auf "Exportiert Überwachungseinstellungen". Wenn die Option deaktiviert ist, dann haben Sie noch keine ProDiag-Überwachung oder einen GRAPH-Funktionsbaustein angelegt.
   Der Dialog "Speichern unter" wird geöffnet.
4. Wählen Sie einen Speicherort aus und vergeben Sie einen Dateinamen.
5. Klicken Sie auf "Speichern".

Ergebnis

Am ausgewählten Speicherort wird eine .dat-Datei angelegt, die nicht bearbeitet werden kann.

24.8.5 ProDiag-Überwachungseinstellungen importieren

Voraussetzung

Sie haben die ProDiag-Überwachungseinstellungen eines Projekts in eine .dat-Datei exportiert.

Sie können die ProDiag-Überwachungseinstellungen auch dann in Ihr Projekt importieren, wenn Sie noch keine ProDiag-Überwachung oder einen GRAPH-Funktionsbaustein angelegt haben.

Hinweis

Projekt hochrüsten

Wenn Sie Ihr Projekt auf die TIA Portal Version 15 hochgerüstet haben und in den ProDiag-Überwachungseinstellungen im Bereich der Unterkategorien die Schaltfläche "Reparieren" sichtbar ist, dann müssen Sie zuerst diese Schaltfläche betätigen, bevor Sie die ProDiag-Überwachungseinstellungen aus einem anderen Projekt importieren können.

Vorgehen

Um die ProDiag-Überwachungseinstellungen in ein neues Projekt zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation im Ordner "Gemeinsame Daten" die "Überwachungseinstellungen".
2. Klicken Sie auf "Importiert Überwachungseinstellungen".
   Wenn "Importiert Überwachungseinstellungen" ausgegraut ist und im Bereich der Unterkategorien die Schaltfläche "Reparieren" sichtbar ist, dann müssen Sie zuerst dieseSchaltfläche betätigen. Dadurch werden die Unterkategorien repariert. Anschließendkönnen Sie die restlichen ProDiag-Überwachungseinstellungen importieren.
3. Der Dialog "Öffnen" wird geöffnet.
   Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist.

4. Wählen Sie die gewünschte .dat-Datei aus.
5. Bestätigen Sie den Dialog mit "Öffnen".

Ergebnis

Alle in der .dat-Datei eingetragenen ProDiag-Überwachungseinstellungen werden in das neue Projekt übernommen.

Hinweis

Import nicht existierender ProDiag-Überwachungseinstellungen

Wenn Sie z. B. Meldeklassen importieren, die im neuen Projekt noch nicht existieren, dann erscheint eine Fehlermeldung und den Kategorien werden die Meldeklassen zugeordnet, die per Voreinstellungen festgelegt wurden.

Wenn Sie die Meldeklassen nach dem Import anlegen, dann müssen Sie diese anschließend manuell den gewünschten Kategorien zuordnen.

24.9 ProDiag-Überwachungen anlegen

24.9.1 Überblick über die Überwachungen

Beschreibung

Sie haben die Möglichkeit, globale und lokale Überwachungen anzulegen. Darüberhinaus können Sie für jeden Operanden mehr als nur eine Überwachung anlegen.

Für jede Überwachung, die Sie anlegen, wird im Inspektorfenster "Eigenschaften" im Register "Überwachungen" ein neuer Eintrag erstellt. Im Register "Überwachungen" können Sie die Einstellungen jeder einzelnen Überwachung festlegen. Wenn Sie für einen Operanden mehrere Überwachungen anlegen, dann werden diese aufsteigend mit dem Zusatz "ID_<fortlaufende Überwachungs-ID> (zugewiesener ProDiag-FB)" nummeriert.

Hinweis

Überwachungen in F-Instanz-Datenbausteinen anlegen

Sie können in der Bausteinschnittstelle eines F-Instanz-Datenbausteins für die booleschen Parameter globale ProDiag-Überwachungen anlegen.

Sobald Sie die erste globale Überwachung innerhalb Ihres Projekts anlegen oder eine lokale Überwachung instanziieren, werden automatisch die ProDiag-Bausteine "Default_SupervisionFB" und "Default_SupervisionDB" in der Projektnavigation im Ordner "Programmbausteine" angelegt.

Sie haben die Möglichkeit auf einen Blick zu sehen, wieviele ProDiag-Überwachungen einem ProDiag-Funktionsbaustein zugeordnet sind. Weitere Informationen dazu finden Sie hier: Anzahl der Überwachungen eines ProDiag-FBs anzeigen (Seite 8221)

Globale Überwachungen

Überwachung einer booleschen Variable innerhalb einer Variablentabelle, eines globalen Datenbausteins oder eines ARRAY-Datenbausteins.

Das folgende Bild zeigt Ihnen, welche Datentypen Sie überwachen können:

GlobalDB

	Name	Datentyp	Startwert	Rem...	Errei...	Nur l...	Sich...	Eins..	Überwachung
1	Static								
2	Start_Motor	Bool	false		☑		☑		
3	Flash_Light	Array[0..1] of Bool			☑		☑		
4	Flash_Light[0]	Bool	false		☑		☑		
5	Flash_Light[1]	Bool	false		☑		☑		
6	Stop_Motor	Struct			☑		☑		
7	Input_1	Bool	false		☑		☑		

Lokale Überwachungen

Überwachung eines booleschen Parameters eines Funktionsbausteins.

Das folgende Bild zeigt Ihnen, welche Parameter Sie überwachen können:

LAD_FB_1500

	Name	Datentyp	Defaultwert	Re...	Errei...	Nur le...	Sich..	Ei...	Überwachung
1	Input								
2	Start_Motor	Bool	false	Nic...	☑	☐	☑	☐	
3	Stop_Motor	Bool	false	Nic...	☑	☐	☑	☐	
4	Output								
5	Flash_Light	Bool	false	Nic...	☑	☐	☑	☐	
6	InOut								
7	Static								
8	Signal_Storage	Bool	false	Nic...	☑	☐	☑	☐	
9	Temp								
10	Constant								

Register "Überwachungen" im Inspektorfenster "Eigenschaften"

Hier können Sie pro angelegter Überwachung eines Operanden die Einstellungen dieser Überwachung festlegen:

Gruppe	Eigenschaft	Beschreibung
Allgemein	Überwachungsart	Es stehen Ihnen die folgenden Überwachungsarten zur Verfügung: • Operand, z. B. zur Überwachung statischer Signale • Verriegelung, z. B. zur Überwachung von Steuersignalen • Aktion, z. B. zur Überwachung, ob die Startlage verlassen wurde • Reaktion, z. B. zur Überwachung, ob die Ziellage rechtzeitig erreicht wurde • Position, z. B. zur Überwachung, ob die Ziellage unerlaubt verlassen wurde • Textmeldung (Gibt eine einfache Textmeldung aus.) • Fehlermeldung (Gibt eine Fehlernummer und den dazugehörigen Text aus.) Je nachdem, welche Überwachungsart Sie auswählen, müssen manche der Felder in den Eigenschaften zwingend ausgefüllt werden. Welche Felder das sind, finden Sie hier: Überwachungsarten (Seite 8154)
	Überwachte(r) Variable / Parameter	Zeigt an, welcher Operand überwacht wird. Die Signalzustände TRUE und FALSE des überwachten Operanden geben an, auf welchen Signalzustand hin der Operand überwacht werden soll.
	Verzögerungzeit / Reaktionszeit / Anlaufzeit	Je nachdem, welche Überwachungsart Sie auswählen, wird eine andere Zeit angezeigt: • Verzögerungszeit: Operand, Verriegelung, Position, Textmeldung, Fehlermeldung • Reaktionszeit: Reaktion • Anlaufzeit: Aktion
	Bedingungen	Als Bedingungen können Sie nur boolesche Variablen und Slice-Zugriffe vom Datentyp BIT verwenden. Die Signalzustände TRUE und FALSE der einzelnen Bedingungen geben an, auf welchen Signalzustand hin der Operand überwacht werden soll.
	Fehlermerker (nur bei globalen Variablen)	Der Fehlermerker wird in den gleichnamigen Datenbaustein des ProDiag-Funktionsbausteins geschrieben.
	ProDiag-FB (nur bei globalen Variablen)	Wenn Sie einer Überwachung keinen anwenderdefinierten ProDiag-Funktionsbaustein zuweisen, dann wird sie automatisch dem ProDiag-Funktionsbaustein "Default_SupervisionFB" zugewiesen. In diesem ProDiag-Funktionsbaustein werden alle globalen und die bereits instanziierten lokalen Überwachungen zusammengefasst, denen dieser Baustein zugewiesen wurde. Sie können sich aber auch eigene ProDiag-Funktionsbausteine erstellen, um die Überwachungen in technologische Einheiten zusammenzufassen. Eine Anleitung zur Erstellung von ProDiag-Funktionsbausteinen finden Sie hier: ProDiag-Funktionsbausteine anlegen (Seite 8190)
	Kategorie	In der Klappliste "Kategorie" werden Ihnen alle 8 Kategorien angezeigt.
	Unterkategorie 1 und 2	Damit Sie in den Klapplisten der Felder "Unterkategorie" Auswahlmöglichkeiten haben, müssen Sie diese in den ProDiag-Überwachungseinstellungen festlegen. Ein Beispiel finden Sie hier: ProDiag-Überwachungseinstellungen festlegen (Seite 8222)
	Meldetext	Der Meldetext ist nicht editierbar, da dieser ebenfalls in den ProDiag-Überwachungseinstellungen definiert wird.

Gruppe	Eigenschaft	Beschreibung
Spezifisches Textfeld	-	Hier können Sie für jede Überwachung ein spezifisches Textfeld mithilfe von Begleitwerten anlegen. Ein Beispiel finden Sie hier: Spezifisches Textfeld anlegen (Seite 8257)

Siehe auch

Überwachungen aller aufgerufenen Funktionsbausteine anzeigen (Seite 8279)

Überwachungen aller globalen Variablen anzeigen (Seite 8277)

Globale Überwachung anlegen (Seite 8250)

Verwendungsmöglichkeiten der Überwachungen (Seite 8249)

Mehrere Überwachungsmeldungen eines ProDiag-FBs innerhalb einer Zykluszeit (Seite 8199)

Lokale Überwachung anlegen (Seite 8252)

Spezifisches Textfeld anlegen (Seite 8255)

24.9.2 Verwendungsmöglichkeiten der Überwachungen

Beschreibung

Sie können Überwachungen für die folgenden Speicherbereiche anlegen:

Speicherbereich/Abschnitt der Bausteinschnittstelle	Speicherort	Überwachung
Eingänge	Variablentabelle	Globale Überwachungen (Variable)
Ausgänge		
Merker		
Elemente eines Datenbausteins, die vom Datentyp BOOL sind	Globaler Datenbaustein oder ein ARRAY-Datenbaustein vom Datentyp BOOL	
Input	Bausteinschnittstelle	Lokale Überwachungen (Parameter)
Output		
Static		

Hinweis

Strukturierte Variablen

In der Bausteinschnittstelle eines Anwenderbausteins können boolesche Elemente einer strukturierten Variablen (PLC-Datentyp, ARRAY oder STRUCT) überwacht werden, wenn diese im Abschnitt "Static" deklariert wurde.

Für jede Überwachungsart stehen Ihnen verschiedene Verwendungsmöglichkeiten der Überwachungen zur Verfügung:

	Globale Überwachungen			Lokale Überwachungen		
Überwachungsart	**Eingänge**	**Ausgänge**	**Merker/Elemente eines Datenbausteins vom Datentyp BOOL**	**Input**	**Output**	**Static**
Operand	x	x	x	x	x	x
Verriegelung	-	x	x	x	-	-
Aktion	x	-	x	x	-	-
Reaktion	x	-	x	x	-	-
Position	x	-	x	x	-	-
Fehlermeldung	x	x	x	x	x	x
Textmeldung	x	x	x	x	x	x
x: Die Verwendung der Überwachung ist möglich. -: Die Verwendung der Überwachung ist nicht möglich.						

24.9.3 Globale Überwachung anlegen

Voraussetzung

Sie haben entweder in einer Variablentabelle oder in einem Datenbaustein mindestens eine boolesche Variable angelegt.

Standard-Variablentabelle

	Name	Datentyp	Adresse	Rema...	Sichtb..	Erreic...	Nur le...	Überwachung
1	Conv_Actor_Forward	Bool	%E0.0	☐	☑	☑	☐	
2	Conv_Actor_Backward	Bool	%E0.1	☐	☑	☑	☐	
3	Conv_Forward_Trigger	Bool	%E0.2	☐	☑	☑	☐	
4	Conv_Backward_Trigger	Bool	%E0.3	☐	☑	☑	☐	
5	Conv_Switch_Left	Bool	%E0.4	☐	☑	☑	☐	
6	Conv_Switch_Right	Bool	%E0.5	☐	☑	☑	☐	
7	Conv_Auto / Hand	Bool	%E0.6	☐	☑	☑	☐	
8	Conv_Push_Button_Forward	Bool	%E0.7	☐	☑	☑	☐	
9	Conv_Push_Button_Backward	Bool	%E1.0	☐	☑	☑	☐	

Vorgehen

Um eine globale Überwachung anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie mit der rechten Maustaste eine oder mehrere Variablen und wählen Sie aus dem Kontextmenü "Neue Überwachung hinzufügen". Der Kontextmenüeintrag wird Ihnen nur angeboten, wenn alle selektierten Variablen überwachbar (boolesche Variablen) sind.
   Im Inspektorfenster "Eigenschaften" wird das Register "Überwachungen" geöffnet und eine Überwachung für die Variable angelegt. Die Überwachungen werden aufsteigend nummeriert (Überwachung_ID_<fortlaufende Überwachungs-ID>(zugewiesener ProDiag-FB)).
2. Per Voreinstellung ist die Überwachungsart "Operand" ausgewählt. Wählen Sie gegebenenfalls eine andere Überwachungsart aus.
   Welche Einstellungen zwingend ausgefüllt werden müssen, richtet sich nach der Überwachungsart, die Sie ausgewählt haben.
   Weitere Informationen zu den Überwachungsarten finden Sie hier: Überwachungsarten (Seite 8154)

   Hinweis

   Einstellungen werden überschrieben

   Wenn Sie die Überwachungsart einer bereits erstellten Überwachung ändern, dann werden Ihre Einstellungen durch die voreingestellten Werte der neuen Überwachungsart überschrieben.

3. Stellen Sie gegebenenfalls eine Verzögerungszeit ein.
4. Wählen Sie gegebenenfalls Bedingungen aus, unter denen eine Überwachungsmeldung ausgelöst werden soll.
5. Wenn Sie nicht den voreingestellten ProDiag-FB "Default_SupervisionFB" verwenden möchten, dann können Sie entweder einen anderen bereits erstellten ProDiag-FB auswählen oder einen neuen ProDiag-FB anlegen.
6. Wählen Sie eine Kategorie und gegebenenfalls Unterkategorien aus.
7. Falls Sie die überwachte Variable noch nicht in Ihrem Programmcode verwenden, dann fügen Sie sie jetzt ein.

Oder:

1. Klicken Sie die gewünschte Variable an.
2. Doppelklicken Sie im Inspektorfenster "Eigenschaften" im Register "Überwachungen" den Eintrag "Neue Überwachung hinzufügen".
   Es wird eine neue Überwachung angelegt. Die Überwachungen werden aufsteigend nummeriert (Überwachung_ID_<fortlaufende Überwachungs-ID>(zugewiesener ProDiag-FB)).
3. Ab hier gilt die gleiche Vorgehensweise wie ab Punkt 2 oben beschrieben.

Ergebnis

Die Überwachung einer globalen Variable wurde erstellt und die Variable wird in der Variablentabelle mit dem ProDiag-Symbol gekennzeichnet. Die Variablen können Sie in allen Programmbausteinen innerhalb Ihres Programms verwenden.

Standard-Variablentabelle

	Name	Datentyp	Adresse	Rema...	Sichtb..	Erreic...	Nur le...	Überwachung
1	Conv_Actor_Forward	Bool	%E0.0	☐	☑	☑	☐	
2	Conv_Actor_Backward	Bool	%E0.1	☐	☑	☑	☐	
3	Conv_Forward_Trigger	Bool	%E0.2	☐	☑	☑	☐	
4	Conv_Backward_Trigger	Bool	%E0.3	☐	☑	☑	☐	
5	Conv_Switch_Left	Bool	%E0.4	☐	☑	☑	☐	
6	Conv_Switch_Right	Bool	%E0.5	☐	☑	☑	☐	
7	Conv_Auto / Hand	Bool	%E0.6	☐	☑	☑	☐	
8	Conv_Push_Button_Forward	Bool	%E0.7	☐	☑	☑	☐	
9	Conv_Push_Button_Backward	Bool	%E1.0	☐	☑	☑	☐	

Zeigt an, dass die Variable überwacht wird.

Hinweis

Variablen kopieren

Wenn Sie eine globale Variable, für die Sie bereits Überwachungen angelegt haben, von einer CPU in eine andere CPU oder von einem Projekt in ein anderes Projekt kopieren, dann werden die Überwachungen nicht mitkopiert.

24.9.4 Lokale Überwachung anlegen

Voraussetzung

Sie haben einen booleschen Parameter in der Bausteinschnittstelle eines Funktionsbausteins in einem der drei Abschnitte angelegt:

LAD_FB_1500

	Name	Datentyp	Defaultwert	Re...	Errei...	Nur le...	Sich..	Ei...	Überwachung
1	Input				☐	☐	☐	☐	
2	Start_Motor	Bool	false	Nic...	☑	☐	☑	☐	
3	Stop_Motor	Bool	false	Nic...	☑	☐	☑	☐	
4	Output				☐	☐	☐	☐	
5	Flash_Light	Bool	false	Nic...	☑	☐	☑	☐	
6	InOut				☐	☐	☐	☐	
7	Static				☐	☐	☐	☐	
8	Signal_Storage	Bool	false	Nic...	☑	☐	☑	☐	
9	Temp				☐	☐	☐	☐	
10	Constant				☐	☐	☐	☐	

Hinweis

Strukturierte Variablen

In der Bausteinschnittstelle können keine booleschen Elemente einer strukturierten Variablen (PLC-Datentyp, ARRAY oder STRUCT) überwacht werden.

Hinweis

Eigenschaften einer lokalen Überwachung

Alle Eigenschaften einer lokalen Überwachung (Variablen für die Bedingungen, Verzögerungszeit oder das spezifische Textfeld) müssen ebenfalls lokal in der Bausteinschnittstelle deklariert sein.

Vorgehen

Um eine lokale Überwachung anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Parameter und wählen Sie aus dem Kontextmenü "Neue Überwachung hinzufügen".
   Im Inspektorfenster "Eigenschaften" wird das Register "Überwachungen" geöffnet und eine Überwachung für den Parameter angelegt. Die Überwachungen werden aufsteigend nummeriert (Überwachung_ID_<fortlaufende Überwachungs-ID>(zugewiesener ProDiag-FB)).
2. Per Voreinstellung ist die Überwachungsart "Operand" ausgewählt. Wählen Sie gegebenenfalls eine andere Überwachungsart aus.
   Welche Einstellungen zwingend ausgefüllt werden müssen, richtet sich nach der Überwachungsart, die Sie ausgewählt haben.
   Weitere Informationen zu den Überwachungsarten finden Sie hier: Überwachungsarten (Seite 8154)

   Hinweis

   Einstellungen werden überschrieben

   Wenn Sie die Überwachungsart einer bereits erstellten Überwachung ändern, dann werden Ihre Einstellungen durch die voreingestellten Werte der neuen Überwachungsart überschrieben.
3. Stellen Sie gegebenenfalls eine Verzögerungszeit ein.
4. Wählen Sie gegebenenfalls Bedingungen aus, unter denen eine Überwachungsmeldung ausgelöst werden soll.
5. Wählen Sie eine Kategorie und gegebenenfalls Unterkategorien aus.
6. Verwenden Sie den überwachten Parameter im Programmcode.
7. Rufen Sie den Funktionsbaustein im Programm auf und wählen Sie im Dialog "Aufrufoptionen" im Bereich "Einzelinstanz" einen Instanz-Datenbaustein aus.
   Erst wenn Sie den Funktionbaustein aufgerufen und einen Instanz-Datenbaustein für den Funktionsbaustein angelegt haben, werden die Überwachungen in Ihrem Projekt aktiviert.

Oder:

1. Klicken Sie den gewünschten Parameter ab.
2. Doppelklicken Sie im Inspektorfenster "Eigenschaften" im Register "Überwachungen" den Eintrag "Neue Überwachung hinzufügen".
   Es wird eine neue Überwachung angelegt. Die Überwachungen werden aufsteigend nummeriert (Überwachung_ID_<fortlaufende Überwachungs-ID>(zugewiesener ProDiag-FB)).
3. Ab hier gilt die gleiche Vorgehensweise wie ab Punkt 2 oben beschrieben.

Ergebnis

Die Überwachungen der Parameter wurden erstellt und in der Bausteinschnittstelle mit dem ProDiag-Symbol gekennzeichnet.

LAD_FB_1500

		Name	Datentyp	Defaultwert	Re...	Errei...	Nur le...	Sich..	Ei...	Überwachung
1		Input								
2		Start_Motor	Bool	false	Nic...	☑	☐	☑	☐	
3		Stop_Motor	Bool	false	Nic...	☑	☐	☑	☐	
4		Output								
5		Flash_Light	Bool	false	Nic...	☑	☐	☑	☐	
6		InOut								
7		Static								
8		Signal_Storage	Bool	false	Nic...	☑	☐	☑	☐	
9		Temp								
10		Constant								

Zeigt an, dass der Parameter überwacht wird.

Wenn Sie den Funktionsbaustein in der Projektnavigation mit der rechten Maustaste anklicken, dann wird Ihnen der Eigenschaftendialog des Funktionsbausteins angezeigt. Im Register "Überwachungsdefinitionen am FB" werden Ihnen alle Überwachungen angezeigt, die Sie für diesen Funktionsbaustein angelegt haben.

Weitere Informationen zur Übersicht der Überwachungen in einem Funktionsbaustein finden Sie hier: Überwachungen eines Funktionsbausteins anzeigen (Seite 8283)

Hinweis

Code Viewer

Bei Überwachungen von Input-Parametern eines Funktionsbausteins können Sie sich im Code Viewer die Vorverschaltung der Input-Parameter anzeigen lassen.

Bei Überwachungen von Output- und/oder Static-Parametern eines Funktionsbausteins können Sie sich im Code Viewer den kompletten Aufruf des Funktionsbausteins anzeigen lassen.

24.9.5 Spezifisches Textfeld anlegen

24.9.5.1 Verwendung der Begleitwerte

Die Begleitwerte SD_4, SD_5 und SD_6 können wie folgt referenziert werden:

@<Nummer des Begleitwerts><Formatangabe>@

Das Zeichen "@" leitet die Verwendung eines Begleitwerts ein und schließt auch damit ab.

Nummer des Begleitwerts

Die Nummer gibt an, welcher Begleitwert bzw. welche Variable ausgelesen werden soll:

Nummer	Erklärung
4	Begleitwert SD_4 (Variable 1)
5	Begleitwert SD_5 (Variable 2)
6	Begleitwert SD_6 (Variable 3)

Maximal erlaubte Größe der Begleitwerte

Wenn die insgesamte Größe der Begleitwerte die maximal erlaubte Größe überschritten hat, dann liegt das in der Regel daran, dass einer oder mehrere der Begleitwerte den Datentyp STRING oder WSTRING haben und keine tatsächliche Länge angegeben wurde. Wenn Sie z. B. für den WSTRING keine tatsächliche Länge angeben, dann wird die maximal mögliche Länge verwendet und die maximal erlaubte Größe ist bereits mit einem Begleitwert überschritten.

Verwenden Sie bei den Datentypen STRING und WSTRING immer die eckigen Klammern [] um die tatsächliche Länge zu definieren.

Formatangabe

Bestimmen Sie das Ausgabeformat für den Begleitwert auf dem Anzeigegerät. Die Formatangabe wird durch das Zeichen "%" eingeleitet. Für die Begleitwerte gibt es folgende feste Formatangaben:

Formatangabe	Beschreibung
%[i]X	Hexadezimalzal mit i Stellen
%[i]u	Dezimalzahl ohne Vorzeichen mit i Stellen
%[i]d	Dezimalzahl mit Vorzeichen mit i Stellen
%[i]b	Binärzahl mit i Stellen
%[i][.y]f	Gleitpunktzahl mit Vorzeichen mit y Stellen nach dem Dezimalpunkt und i Gesamtstellen
%[i]s	Zeichenkette (ANSI String) mit i Stellen Zeichen werden gedruckt bis zum ersten 0 Byte (00Hex).
%t#<Name der Textbibliothek>	Zugriff auf Textbibliothek

Ist die Stellenanzahl [i] zu klein, so wird der Wert trotzdem in voller Länge ausgegeben.

Ist die Stellenanzahl [i] zu groß, so wird vor dem Wert eine passende Anzahl Füllzeichen ausgegeben.

Ist die Stellenanzahl [i] zu groß und wird (bei Dezimal-/Gleitpunktzahlen) ein Vorzeichen ausgegeben, so verringert sich die Anzahl der Füllzeichen um eine Stelle. Gleiches gilt für das Punkt- bzw. Kommazeichen innerhalb von Gleitpunktzahlen.

Hinweis

Bitte beachten Sie, dass Sie "[i]" optional angeben können, wobei Sie die eckigen Klammern bei der Eingabe weglassen müssen.

Beispiele

In der folgenden Tabelle sehen Sie einige Beispiele:

Wert der Variable	Begleitwert	Erklärung	Ausgabe
123*	@4%6d@	Der Wert des Begleitwerts SD_4 wird als Dezimalzahl mit Vorzeichen und 6 Gesamtstellen dargestellt.	000123
1234.567	@4%8.3f@	Der Wert des Begleitwerts SD_4 wird als Gleitpunktzahl mit Vorzeichen mit 3 Stellen nach dem Dezimalpunkt und 8 Gesamtstellen dargestellt. Es wird kein Füllzeichen eingefügt, da der Dezimalpunkt das achte Zeichen darstellt.	1234.567
5.4*	@5%6f@	Der Wert des Begleitwerts SD_5 wird als Ganzzahl mit 6 Gesamtstellen dargestellt. Es wurde keine Anzahl der Stellen nach dem Dezimalpunkt definiert.	0005.4
5.4	@5%2f@	Der Wert des Begleitwerts SD_5 wird als Ganzzahl mit 3 Gesamtstellen dargestellt. Der Wert wird nicht gekürzt.	5.4
Inhalt der Textbibliothek <TextLib1>	@6%t#TextLib1@	Der Begleitwert SD_6 vom Datentyp WORD ist der Index, der auf die verwendete Textbibliothek "TextLib1" referenziert.	90°

Die Art des Füllzeichens (z. B. Nullen oder Leerzeichen) hängt vom Ausgabesystem ab.

24.9.5.2 Spezifisches Textfeld anlegen

Über das spezifische Textfeld können Sie zusätzliche überwachungsgranulare Informationen mithilfe von Begleitwerten in einer Überwachungsmeldung mit ausgeben. Diese Informationen können im spezifischen Textfeld frei miteinander kombiniert werden und bestehen aus den folgenden Elementen:

- Freier Text (z. B. Temperatur, Handbetrieb, Automatikbetrieb)
- Wert der drei verschiedenen Variablen (Begleitwerte SD_4 - SD_6), die zur Laufzeit ausgelesen werden
- Textlisteneintrag

Hinweis

Verwendung des Zeichens @ im spezifischen Textfeld

Bitte verwenden Sie das Zeichen @ nur für die Deklaration der Begleitwerte, z. B. @4%6d@. Wenn Sie das Zeichen @ darüber hinaus im Freitext des spezifischen Textfelds verwenden, dann kann das unter Umständen dazu führen, dass die im Freitext enthaltenen Begleitwerte in der Überwachungsmeldung nicht aufgelöst werden können.

Voraussetzung

Sie haben entweder eine globale oder eine lokale Überwachung angelegt und im Inspektorfenster "Eigenschaften" das Register "Überwachungen" geöffnet.

Vorgehen

Um das spezifische Textfeld für eine Überwachung anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Gruppe "Spezifisches Textfeld" an.
2. Wählen Sie aus der Autovervollständigung der drei Begleitwerte SD_4 - SD_6 die gewünschten Variablen aus.
   Die Variablen müssen vom Datentyp BOOL, BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, REAL, LREAL, CHAR, WCHAR, STRING oder WSTRING sein.
3. Geben Sie im Textfeld den gewünschten Text ein. Dieser kann z. B. aus einem freien Text und einer Referenz auf die Variablen bestehen, die Sie als Begleitwerte definiert haben. (@4 referenziert z. B. auf die Variable des Begleitswerts SD_4)
   Wie die Syntax der Referenz auf eine Variable aussehen muss, lesen Sie hier: Verwendung der Begleitwerte (Seite 8255)

Hinweis

Erscheinen des spezifischen Textfelds

Diese Information wird nur dann in der Überwachungsmeldung mit ausgegeben, wenn Sie in den ProDiag-Überwachungseinstellungen beim Projektieren der Meldetexte das Textfeld "<Spezifisches Textfeld>" verwendet haben. Weitere Informationen zu den Meldetexten in den ProDiag-Überwachungseinstellungen finden Sie hier: Zentrale ProDiag-Meldetexte festlegen (Seite 8233)

Ergebnis

Sobald eine Überwachungsmeldung durch eine Störung ausgelöst wird, wird die Information aus dem spezifischen Textfeld mit ausgegeben, wenn Sie in der Meldetextstruktur das Meldetextfeld "<Spezifisches Textfeld>" verwendet haben.

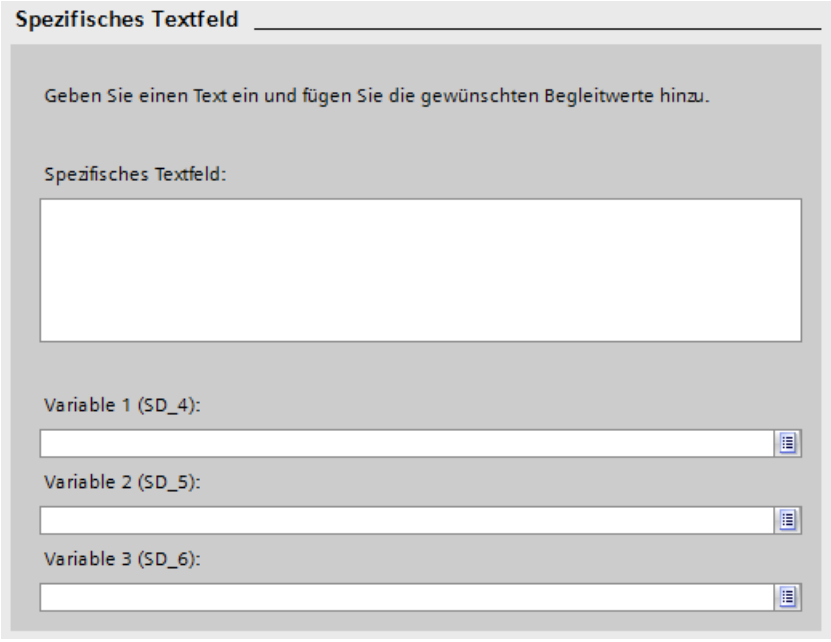

24.9.5.3 Übersetzungen des spezifischen Textfelds anzeigen und editieren

Sie können sich das spezifische Textfeld in allen Projektsprachen anzeigen lassen, die Sie in den Einstellmöglichkeiten zu den Projektsprachen aktiviert haben und deren Übersetzungen bereits vorliegen. Weitere Informationen zur Aktivierung von Projektsprachen finden Sie unter AUTOHOTSPOT.

Voraussetzung

Sie haben mindestens für eine ProDiag-Überwachung ein spezifisches Textfeld angelegt.

Vorgehen

Sie haben mehrere Möglichkeiten, um sich die Übersetzungen des spezifischen Textfelds anzeigen zu lassen:

- Im ProDiag-Übersichtseditor
- In den Bausteineigenschaften des ProDiag-Funktionsbausteins
- Im ProDiag-Funktionsbaustein

Um sich die Übersetzungen im ProDiag-Übersichtseditor anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unterhalb der CPU auf den Eintrag "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie entweder das Register "Variablenüberwachungen" oder "Überwachungsdefinitionen am FB" an.
3. Wählen Sie entweder die gesamte Zeile der gewünschten Überwachung oder die Zelle, in der der Text des spezifischen Textfelds steht, aus. Wenn Sie mehrere Texte gleichzeitig im Register "Texte" anzeigen wollen, dann wählen Sie mehrere Zellen mit spezifischen Texten an.
   Der Text der spezifischen Textfelder wird im Inspektorfenster im Register "Texte" angezeigt.

Um sich die Übersetzungen in den Bausteineigenschaften des ProDiag-Funktionsbausteins anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie das Register "Texte" an.
   Die Texte der spezifischen Textfelder werden angezeigt.

Um sich die Übersetzungen im ProDiag-Funktionsbaustein anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie den ProDiag-Funktionsbaustein.
   Der ProDiag-Funkionsbaustein wird geöffnet.
2. Klicken Sie im Inspektorfenster das Register "Texte" an.
   Die Texte der spezifischen Textfelder werden angezeigt.

Wenn Ihnen die Namen der Kategorien nur in der aktuellen Oberflächensprache angezeigt werden, dann haben Sie noch keine Projektsprachen ausgewählt bzw. es sind noch keine Übersetzungen vorhanden. Sie können die Übersetzungen entweder direkt in die Tabelle im Register "Texte" eintragen oder Sie können die Übersetzungen für die einzelnen Projektsprachen in die globale Tabelle "Projekttexte" eingeben. Die Tabelle finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

Weitere Informationen zum Übersetzen von Texten finden Sie unter AUTOHOTSPOT.

24.9.6 Überwachung löschen

Voraussetzung

Sie haben mindestens eine Überwachung angelegt.

Vorgehen

Um eine Überwachung zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie die Variable oder den Parameter mit der zu löschenden Überwachung an. Im Inspektorfenster "Eigenschaften" wird das Register "Überwachungen" geöffnet.
2. Klicken Sie die Überwachung mit einem rechten Mausklick an und wählen Sie aus dem Kontextmenü "Überwachung löschen".

Ergebnis

Die Überwachung wurde gelöscht.

Hinweis

Überwachung wird nicht gelöscht

Wenn Sie den Datentyp eines überwachten Operanden ändern oder den überwachten Operanden löschen, dann wir die Überwachung nicht automatisch mit gelöscht. Wenn Sie tatsächlich den Operanden inklusive Überwachung löschen wollen, dann müssen Sie die Überwachung separat löschen.

24.10 ProDiag-Funktionsbausteine kopieren

24.10.1 Überblick über das Kopieren von ProDiag-Funktionsbausteinen

Beschreibung

Anstatt die ProDiag-Funkionsbausteine neu anzulegen, können Sie sie auch kopieren. Dabei stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Sie können den ProDiag-FB innerhalb eines Projekts von einer S7-1500-CPU in eine andere S7-1500-CPU kopieren.
- Sie können den ProDiag-FB von einem Projekt in ein anderes Projekt kopieren.
- Sie können den ProDiag-FB von einem Projekt in eine Projektbibliothek kopieren.
- Sie können den ProDiag-FB von einem Projekt in eine globale Bibliothek kopieren.
- Sie können den ProDiag-FB von einer Projektbibliothek in eine globale Bibliothek und umgekehrt kopieren.

Alle im ProDiag-Funktionsbaustein enthaltenen globalen Überwachungen bleiben beim Kopieren erhalten. Daher ist es empfehlenswert, auch die überwachten globalen Variablen mit zu kopieren, da es sonst zu einem Übersetzungsfehler kommt.

Die enthaltenen lokalen Überwachungen werden nicht mitkopiert.

Hinweis

Konsistenz der ProDiag-Konfiguration gewährleisten

Ein ProDiag-Funktionsbaustein muss nach dem Kopieren immer übersetzt werden, um die Konsistenz der ProDiag-Konfiguration zu gewährleisten. Bei einem Offline-Vergleich nach dem Kopieren wird immer ein Unterschied angezeigt.

Siehe auch

ProDiag-Funktionsbaustein kopieren (Seite 8261)

ProDiag-Funktionsbaustein einfügen (Seite 8262)

24.10.2 ProDiag-Funktionsbaustein kopieren

Voraussetzung

Sie haben den ProDiag-Funktionsbaustein markiert.

Vorgehen

Um den ProDiag-Funktionsbaustein zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

Ergebnis

Eine Kopie des ProDiag-Funktionsbausteins befindet sich nun in der Zwischenablage und kann an der gewünschten Stelle wieder eingefügt werden.

24.10.3 ProDiag-Funktionsbaustein einfügen

Voraussetzung

Sie haben einen ProDiag-Funktionsbaustein kopiert.

Vorgehen

Um einen kopierten ProDiag-Funktionsbaustein und seine globalen Überwachungen einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die gewünschte Stelle, in die Sie den kopierten Baustein einfügen möchten.
2. Klicken Sie mit der rechten Maustaste und wählen Sie im Kontextmenü den Befehl "Einfügen".
   - Falls Sie den ProDiag-Funktionsbaustein in die S7-1500-CPU des Originalbausteins einfügen, dann wird die Kopie mit der Namenserweiterung "_<fortlaufende Nummer>" eingefügt.
   - Falls Sie den ProDiag-Funktionsbaustein in eine andere S7-1500-CPU einfügen und es einen ProDiag-Funktionsbaustein mit diesem Namen schon gibt, dann wird der Dialog "Einfügen" geöffnet. Wählen Sie die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

Ergebnis

Der kopierte ProDiag-Funktionsbaustein wurde eingefügt.

24.11 ProDiag-Funktionsbausteine zuweisen

24.11.1 Überblick über die Zuweisung zu einem ProDiag-FB

Beschreibung

Jede ProDiag-Überwachung muss einem ProDiag-Funktionsbaustein zugewiesen werden, damit sie im Anwenderprogramm aktiviert ist.

Beim Anlegen einer globalen Überwachung wird diese automatisch einem ProDiag-Funktionsbaustein zugewiesen. Diese Zuweisung können Sie entweder im Inspektorfenster "Eigenschaften" oder in der ProDiag-Übersichtstabelle "Variablenüberwachungen" verändern. Weitere Informationen finden Sie hier: Globale Überwachungen einem ProDiag-FB zuweisen (Seite 8263)

Bei lokalen Überwachungen können Sie die Überwachungen entweder beim Aufruf des anwenderdefinierten Funktionsbausteins im Programmcode, in der ProDiag-Übersichtstabelle "Instanziierte FB-Überwachungen" oder in den Eigenschaften des anwenderdefinierten Instanz-Datenbausteins einem ProDiag-Funktionsbaustein zuweisen. Weitere Informationen finden Sie hier: Lokale Überwachungen einem ProDiag-FB zuweisen (Seite 8264)

Wenn Sie keinen anwenderdefinierten ProDiag-Funktionsbaustein auswählen, dann werden die Überwachungen per Voreinstellung entweder dem ProDiag-Funktionsbaustein "Default_SupervisionFB" oder dem ersten gefundenen anwenderdefinierten ProDiag-Funktionsbaustein zugeordnet.

Sie können die Zuordnung einer Überwachung zu einem ProDiag-Funktionsbaustein jederzeit auch im Nachhinein noch verändern.

Siehe auch

Instanz-Datenbaustein einem ProDiag-FB zuweisen (lokale Überwachungen) (Seite 8266)

Bestehende Zuweisung eines ProDiag-FBs ändern (Seite 8267)

Bestehende Zuweisung eines ProDiag-FBs entfernen (Seite 8269)

24.11.2 Globale Überwachungen einem ProDiag-FB zuweisen

Voraussetzung

Sie haben eine globale Überwachung angelegt.

Vorgehen

Um globale Überwachungen einem ProDiag-Funktionsbaustein zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Inspektorfenster "Eigenschaften" das Register "Überwachungen" an.
2. Klicken Sie die gewünschte Überwachung an.
3. Klicken Sie bei ProDiag-FB auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Oder:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Variablenüberwachungen" an.
   Es werden alle globalen Überwachungen angezeigt.
3. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
4. Ab hier gilt die gleiche Vorgehensweise wie ab Punkt 4 oben beschrieben.

Ergebnis

Die Überwachung wurde einem ProDiag-Funktionsbaustein zugewiesen.

Siehe auch

Globale Überwachung anlegen (Seite 8250)

24.11.3 Lokale Überwachungen einem ProDiag-FB zuweisen

Voraussetzung

Sie haben mindestens eine lokale Überwachung an einem Funktionsbaustein angelegt.

Vorgehen

Um lokale Überwachungen eines Funktionsbausteins einem ProDiag-Funktionsbaustein zuzuweisen, gehen Sie folgendermaßen vor:

1. Rufen Sie den Funktionsbaustein im Programmcode auf.
   Der Dialog "Aufrufoptionen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Einzel-Instanz".
3. Wählen Sie den gewünschten ProDiag-Funktionsbaustein aus dem Auswahlfeld "..." aus.
4. Bestätigen Sie Ihre Eingaben mit "OK".

Oder:

1. Rufen Sie den Funktionsbaustein im Programmcode auf.
   Der Dialog "Aufrufoptionen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Multi-Instanz".
3. Wählen Sie die gewünschte Multiinstanz aus.
4. Bestätigen Sie Ihre Eingaben mit "OK".
5. Rufen Sie den Funktionsbaustein, der die Multiinstanz beinhaltet, ebenfalls im Programmcode auf.
   Der Dialog "Aufrufoptionen" wird geöffnet.
6. Klicken Sie auf die Schaltfläche "Einzel-Instanz".
7. Wählen Sie den gewünschten ProDiag-Funktionsbaustein aus dem Auswahlfeld "..." aus.
8. Bestätigen Sie Ihre Eingaben mit "OK".

Oder:

Um lokale Überwachungen einem ProDiag-Funktionsbaustein mithilfe des Dialogs "Neuen Baustein hinzufügen" zuzuweisen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".
   Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein".
3. Wählen Sie bei Typ aus der Klappliste den bereits angelegten Funktionsbaustein, der lokale Überwachungen enthält, aus.
4. Wählen Sie den gewünschten ProDiag-Funktionsbaustein aus dem Auswahlfeld "..." aus.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Oder:

Um lokale Überwachungen einem ProDiag-Funktionsbaustein im ProDiag-Übersichtseditor "PLC-Überwachungen & -Meldungen" zuzuweisen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Instanziierte FB-Überwachungen" an.
   Es werden alle lokalen Überwachungen angezeigt.

3. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Ergebnis

Die ProDiag-Überwachungen des anwenderdefinierten Funktionsbausteins sind instanziiert.

24.11.4 Instanz-Datenbaustein einem ProDiag-FB zuweisen (lokale Überwachungen)

Beschreibung

Sie können einen Instanz-Datenbaustein eines anwenderdefinierten Funktionsbausteins direkt einem ProDiag-Funktionsbaustein zuweisen.

Diese Zuweisung können Sie auch dann machen, wenn der anwenderdefinierte Funktionsbaustein noch keine Überwachungen enthält. Darüber hinaus bleibt die Zuweisung zum ProDiag-Funktionsbaustein bestehen, wenn der anwenderdefinierte Funktionsbaustein gelöscht wird.

Vorgehen

Um einen Instanz-Datenbaustein eines anwenderdefinierten Funktionsbausteins einem ProDiag-Funktionsbaustein zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie den Instanz-Datenbaustein eines Funktionsbausteins mit einem rechten Mausklick an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaften-Dialog des Instanz-Datenbausteins wird geöffnet.
3. Klicken Sie den Abschnitt "Attribute" an.
4. Klicken Sie bei "ProDiag-FB zuweisen" auf das Auswahlfeld "...", um einen ProDiag-Funktionsbaustein auszuwählen.
   Ein Auswahl-Dialog wird geöffnet.
5. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
6. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Oder:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Instanziierte FB-Überwachungen" an.
   Es werden alle lokalen Überwachungen angezeigt.
3. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...", um einen ProDiag-Funktionsbaustein auszuwählen.
   Ein Auswahl-Dialog wird geöffnet.
4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Ergebnis

Die ProDiag-Überwachungen des anwenderdefinierten Funktionsbausteins sind instanziiert.

24.11.5 Bestehende Zuweisung eines ProDiag-FBs ändern

Voraussetzung

Sie haben bereits globale oder lokale Überwachungen angelegt und ihnen einen ProDiag-Funktionsbaustein zugewiesen.

Vorgehen bei globalen Überwachungen

Um die bereits bestehenden Zuweisungen globaler Überwachungen zu ProDiag-Funktionsbausteinen zu verändern, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Variablenüberwachungen" an.
   Es werden alle globalen Überwachungen angezeigt.
3. Weisen Sie den entsprechenden Überwachungen in der Spalte "ProDiag-FB" einen anderen ProDiag-Funktionsbaustein zu.

oder:

1. Klicken Sie die überwachte Variable z. B. in der Standard-Variablentabelle an.
   Im Inspektorfenster "Eigenschaften" wird das Register "Überwachungen" geöffnet und die Eigenschaften der Überwachung angezeigt.
2. Klicken Sie bei ProDiag-FB auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.

3. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
4. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

oder:

1. Doppelklicken Sie in der Projektnavigation den entsprechenden ProDiag-Funktionsbaustein.
   Der ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
3. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
4. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Vorgehen bei lokalen Überwachungen

Um einen Instanz-Datenbaustein eines anwenderdefinierten Funktionsbausteins einem ProDiag-Funktionsbaustein zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie den Instanz-Datenbaustein eines Funktionsbausteins mit einem rechten Mausklick an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaften-Dialog des Instanz-Datenbausteins wird geöffnet.
3. Klicken Sie den Abschnitt "Attribute" an.
4. Klicken Sie bei "ProDiag-FB zuweisen" auf das Auswahlfeld "...", um einen ProDiag-Funktionsbaustein auszuwählen.
   Ein Auswahl-Dialog wird geöffnet.
5. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
6. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Oder:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Instanziierte FB-Überwachungen" an.
   Es werden alle lokalen Überwachungen angezeigt.
3. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...", um einen ProDiag-Funktionsbaustein auszuwählen.
   Ein Auswahl-Dialog wird geöffnet.

4. Klicken Sie in der linken Spalte auf den Ordner "Programmbausteine".
   Es werden Ihnen in der rechten Spalte alle bereits vorhandenen ProDiag-Funktionsbausteine angezeigt.
5. Wählen Sie den entsprechenden ProDiag-Funktionsbaustein aus.

Ergebnis

Die Zuweisungen des ProDiag-Funktionsbausteins wurde geändert.

Siehe auch

Überwachungen aller aufgerufenen Funktionsbausteine anzeigen (Seite 8279)

Überwachungen aller globalen Variablen anzeigen (Seite 8277)

24.11.6 Bestehende Zuweisung eines ProDiag-FBs entfernen

Voraussetzung

Sie haben bereits lokale Überwachungen angelegt und ihnen einen ProDiag-Funktionsbaustein zugewiesen.

Vorgehen

Um die bereits bestehende Zuweisung eines Instanz-Datenbausteins zu einem ProDiag-Funktionsbaustein zu verändern, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Instanziierte FB-Überwachungen" an.
   Es werden alle lokalen Überwachungen angezeigt.
3. Klicken Sie in der Spalte "ProDiag-FB" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
4. Klicken Sie auf "Keiner".

oder:

1. Klicken Sie den Instanzdatenbaustein eines Funktionsbausteins, der lokale Überwachungen enthält, mit einem rechten Mausklick an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".
   Der Eigenschaften-Dialog des Instanzdatenbausteins wird geöffnet.
3. Klicken Sie den Abschnitt "Attribute" an.
4. Klicken Sie bei "ProDiag-FB zuweisen" auf das Auswahlfeld "...".
   Ein Auswahl-Dialog wird geöffnet.
5. Klicken Sie auf "Keiner".

Ergebnis

Die bestehende Zuweisung des ProDiag-Funktionsbausteins wurde entfernt.

24.12 ProDiag-Funktionsbausteine bearbeiten

24.12.1 Neue Überwachungen im ProDiag-FB einfügen

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch neue Überwachungen hinzufügen.

Voraussetzung

Sie haben einem ProDiag-Funktionsbaustein globale Überwachungen zugewiesen und ihn geöffnet.

Vorgehen

Um eine neue Überwachung hinzuzufügen, haben Sie zwei Möglichkeiten:

1. Klicken Sie mit der rechten Maustaste auf die letzte Zeile der Tabelle und wählen Sie aus dem Kontextmenü "Neue Überwachung hinzufügen".
   Es wird eine neue Überwachung mit den voreingestellten Werten (Überwachungsart: "Operand", Kategorie: "1: Fehler", Verzögerungszeit: T#0ms) angelegt.
2. Wählen Sie eine Variable aus, die überwacht werden soll.
3. Überschreiben Sie die Voreinstellungen, indem Sie die Einstellungen entsprechend ändern.

oder:

1. Geben Sie in der letzten Zeile in der Spalte "Überwachte Variable" eine Variable ein, die überwacht werden soll.
   Es wird eine neue Überwachung mit den voreingestellten Werten (Überwachungsart: "Operand", Kategorie: "1: Fehler", Verzögerungszeit: T#0ms) angelegt.
2. Wenn Sie eine Variable angeben, die Sie noch nicht definiert haben, dann wird das Feld "Überwachte Variable" rot hinterlegt.
3. Definieren Sie die Variable entweder in einer Variablentabelle oder in einem Datenbaustein.

Ergebnis

Die neuen Überwachungen wurden hinzugefügt.

24.12.2 Überwachungen im ProDiag-FB kopieren

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch einzelne oder mehrere Überwachungen innerhalb des Registers oder in ein anderes Register "Variablenüberwachungen" kopieren.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die gesamte Zeile einer Überwachung zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile, die Sie kopieren möchten.
   Sie können auch mehrere Zeilen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke am Ende der Tabelle.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

- Die Überwachung wird an das Ziel kopiert.
- Der kopierten Überwachung wird eine neue ID zugeordnet.
- Alle weiteren Eigenschaften der Überwachung bleiben gleich.

24.12.3 Überwachungen im ProDiag-FB löschen

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch einzelne oder mehrere Überwachungen löschen.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die gesamte Zeile einer Überwachung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Überwachung.
   Sie können auch mehrere Zeilen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die Überwachung ist gelöscht.

24.12.4 Überwachungen im ProDiag-FB sortieren

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch die Überwachungen in den Spalten "Überwachte Variable" und "ProDiag-FB" sortieren.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die Überwachungen zu sortieren, gehen Sie folgendermaßen vor:

1. Wählen Sie eine der beiden Spalten aus, nach der Sie sortieren möchten.
2. Klicken Sie in den Spaltenkopf.
   Die Spalte wird aufsteigend sortiert.
   Ein Pfeil-nach-oben zeigt die Sortierreihenfolge an.
3. Um die Sortierreihenfolge zu ändern, klicken Sie auf den Pfeil.
   Die Spalte wird absteigend sortiert. Ein Pfeil-nach-unten zeigt die Sortierreihenfolge an.
4. Um die ursprüngliche Anordnung wiederherzustellen, klicken Sie ein drittes Mal auf den Spaltenkopf.

Ergebnis

Die Überwachung sind wie gewünscht sortiert.

24.13 ProDiag-Funktionsbausteine aufrufen

24.13.1 Überblick über den Aufruf der ProDiag-Funktionsbausteine

Beschreibung

Sobald Sie nach dem Anlegen und Konfigurieren der Überwachungen Ihr Projekt übersetzen, werden die angelegten ProDiag-Funktionsbausteine automatisch im ProDiag-Organisationsbaustein aufgerufen.

Damit sind die Überwachungen in Ihrem Projekt aktiviert. Auf Ihrem HMI-Gerät haben Sie nun verschiedene Möglichkeiten die Überwachungen zu visualisieren.

Wenn Sie die ProDiag-Funktionsbausteine nicht im ProDiag-OB aufrufen lassen möchten, dann können Sie sie auch in anderen Bausteinen (FCs, FBs oder OBs) aufrufen. Es ist empfehlenswert, die Aufrufe entweder direkt in einem Zyklus-OB oder in einem Baustein, der wiederum in einem Zyklus-OB aufgerufen wird, zu programmieren. In diesem Fall wird der jeweilige ProDiag-Funktionsbaustein nicht mehr im ProDiag-OB aufgerufen.

Siehe auch

Überwachte anwenderdefinierte Bausteine anzeigen (Seite 8286)

24.14 ProDiag-Bausteine laden

24.14.1 ProDiag-Bausteine laden

Beschreibung

Beim Ladevorgang werden alle Informationen, die zur Rekonstruktion des Programms notwendig sind, auch symbolische Informationen, wie z. B. Namen und Kommentare für Code- und Datenbausteine, in der aktuellen Projektsprache mitgeladen.

Die ProDiag-Bausteine haben keine Speicherreserve und Sie können sie nur mit Reinitialisierung laden.

Weitere Informationen zum Laden von Bausteinen finden Sie hier: Bausteine laden für S7-1200/1500 (Seite 7871)

24.15 ProDiag-Überwachungen anzeigen

24.15.1 Überblick über die Anzeigemöglichkeiten

Beschreibung

Sie haben die Möglichkeit, sich alle angelegten Überwachungen in unterschiedlichen Übersichten anzeigen zu lassen. Abhängig davon, ob es sich um eine lokale oder eine globale Überwachung handelt, stehen Ihnen die folgenden Anzeigemöglichkeiten zur Verfügung:

Anzeigemöglichkeit	Beschreibung
ProDiag-Übersichtseditor	Im ProDiag-Übersichtseditor können Sie sich sowohl die globalen als auch die lokalen Überwachungen einer CPU anzeigen lassen.
Eigenschaften eines Funktionsbausteins	In den Eigenschaften einen Funktionsbausteins können Sie sich alle lokalen Überwachungen anzeigen lassen, die Sie in der Bausteinschnittstelle des Funktionsbausteins angelegt haben.
ProDiag-Funktionsbaustein	Im geöffneten ProDiag-Funktionsbaustein können Sie sich alle globalen und instanziierten lokalen Überwachungen anzeigen lassen, denen Sie diesen ProDiag-Funktionsbaustein zugeordnet haben.

Siehe auch

Überwachungen eines Funktionsbausteins anzeigen (Seite 8283)

Überwachungen aller Funktionsbausteine anzeigen (Seite 8278)

Überwachungen aller aufgerufenen Funktionsbausteine anzeigen (Seite 8279)

Überwachungen aller globalen Variablen anzeigen (Seite 8277)

Neue Variablenüberwachungen im ProDiag-Übersichtseditor einfügen (Seite 8280)

Überwachungen eines ProDiag-Funktionsbausteins anzeigen (Seite 8284)

Überwachungen im ProDiag-Übersichtseditor anzeigen (Seite 8277)

24.15.2 Überwachungen im ProDiag-Übersichtseditor anzeigen

24.15.2.1 Aufbau des ProDiag-Übersichtseditors

Aufbau des ProDiag-Übersichtseditors

Das folgende Bild zeigt exemplarisch die Komponenten des ProDiag-Übersichtseditors:

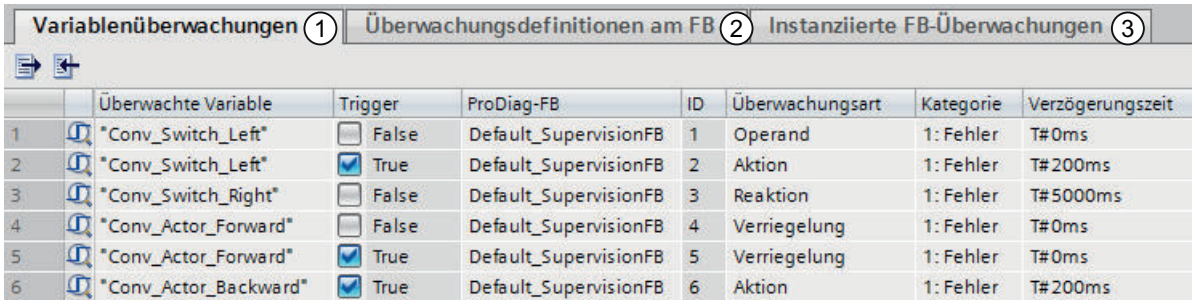

		Überwachte Variable	Trigger	ProDiag-FB	ID	Überwachungsart	Kategorie	Verzögerungszeit
1		"Conv_Switch_Left"	False	Default_SupervisionFB	1	Operand	1: Fehler	T#0ms
2		"Conv_Switch_Left"	True	Default_SupervisionFB	2	Aktion	1: Fehler	T#200ms
3		"Conv_Switch_Right"	False	Default_SupervisionFB	3	Reaktion	1: Fehler	T#5000ms
4		"Conv_Actor_Forward"	False	Default_SupervisionFB	4	Verriegelung	1: Fehler	T#0ms
5		"Conv_Actor_Forward"	True	Default_SupervisionFB	5	Verriegelung	1: Fehler	T#0ms
6		"Conv_Actor_Backward"	True	Default_SupervisionFB	6	Aktion	1: Fehler	T#200ms

Nummerierung	Erklärung	Weiterführende Informationen
1	Das Register "Variablenüberwachungen" zeigt alle globalen Überwachungen an.	Weitere Informationen finden Sie hier: Überwachungen aller globalen Variablen anzeigen (Seite 8277)
2	Das Register "Überwachungsdefinitionen am FB" zeigt alle in der Bausteinschnittstelle von anwenderdefinierten Funktionsbausteinen definierten Überwachungen an, auch wenn dieser anwenderdefinierte Funktionsbaustein noch nicht im Programmcode aufgerufen wurde.	Weitere Informationen finden Sie hier: Überwachungen aller Funktionsbausteine anzeigen (Seite 8278)
3	Das Register "Instanziierte FB-Überwachungen" zeigt alle Überwachungen der Funktionsbausteine an, die bereits im Programmcode aufgerufen wurden.	Weitere Informationen finden Sie hier: Überwachungen aller aufgerufenen Funktionsbausteine anzeigen (Seite 8279)
	Zeigt an, dass für die Variable mindestens eine Überwachung angelegt wurde.	

24.15.2.2 Überwachungen aller globalen Variablen anzeigen

Voraussetzung

Es wird mindestens eine Variable in einem globalen Datenbaustein oder einer Variablentabelle überwacht.

Vorgehen

Um sich alle globalen Überwachungen einer CPU anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Variablenüberwachungen" an.
   Es werden alle globalen Überwachungen angezeigt.
3. Bei Bedarf können Sie weitere Spalten in der Tabelle ein- und ausblenden. Klicken Sie dazu in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
4. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
5. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
6. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.
   Die Reihenfolge der Spalten kann nicht verändert werden.

Ergebnis

Im Register "Variablenüberwachungen" werden alle globalen Überwachungen und die gewünschten Spalten angezeigt.

24.15.2.3 Überwachungen aller Funktionsbausteine anzeigen

Voraussetzung

In der Bausteinschnittstelle mehrerer Funktionsbausteine wurden Überwachungen angelegt

Vorgehen

Um sich die Überwachungen aller Funktionsbausteine anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Überwachungsdefinitionen am FB" an.
   Es werden Ihnen alle Überwachungen aller Funktionsbaustein der CPU angezeigt, auch diejenigen, die Sie noch nicht instanziiert haben.
3. Bei Bedarf können Sie weitere Spalten in der Tabelle ein- und ausblenden. Klicken Sie dazu in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.

4. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
5. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
6. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.
   Die Reihenfolge der Spalten kann nicht verändert werden.

Ergebnis

Im Register "Überwachungsdefinitionen am FB" werden alle Überwachungen aller Funktionsbausteine und die gewünschten Spalten angezeigt.

24.15.2.4 Überwachungen aller aufgerufenen Funktionsbausteine anzeigen

Voraussetzung

Es wurde mindestens ein Funktionsbaustein, der Überwachungen enthält, im Programmcode aufgerufen.

Vorgehen

Um sich die Überwachungen aller aufgerufenen Funktionsbausteine anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
   Der Meldungs- und Überwachungseditor wird geöffnet.
2. Klicken Sie das Register "Überwachungen > Instanziierte FB-Überwachungen" an.
   Es werden alle Instanz-Datenbausteine, die Überwachungen von Funktionsbausteinen enthalten, angezeigt.
3. Bei Bedarf können Sie weitere Spalten in der Tabelle ein- und ausblenden. Klicken Sie dazu in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
4. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
5. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
6. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.
   Die Reihenfolge der Spalten kann nicht verändert werden.

Ergebnis

Im Register "Instanziierte Überwachungen am FB" werden alle Instanz-Datenbausteine, die Überwachungen von Funktionsbausteinen enthalten, und die gewünschten Spalten angezeigt.

24.15.3 ProDiag-Übersichtseditor bearbeiten

24.15.3.1 Neue Variablenüberwachungen im ProDiag-Übersichtseditor einfügen

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch neue Überwachungen hinzufügen.

Vorgehen

Um eine neue Überwachung hinzuzufügen, haben Sie drei Möglichkeiten:

1. Klicken Sie mit der rechten Maustaste auf die letzte Zeile der Tabelle und wählen Sie aus dem Kontextmenü "Neue Überwachung hinzufügen".
   Es wird eine neue Überwachung mit den voreingestellten Werten (Überwachungsart: "Operand", Kategorie: "1: Fehler", Verzögerungszeit: T#0ms) angelegt.
2. Wählen Sie eine Variable aus, die überwacht werden soll.
3. Überschreiben Sie die Voreinstellungen, indem Sie die Einstellungen entsprechend ändern.

oder:

1. Geben Sie in der letzten Zeile in der Spalte "Überwachte Variable" eine Variable ein, die überwacht werden soll.
   Es wird eine neue Überwachung mit den voreingestellten Werten (Überwachungsart: "Operand", Kategorie: "1: Fehler", Verzögerungszeit: T#0ms) angelegt.
2. Wenn Sie eine Variable angeben, die Sie noch nicht definiert haben, dann wird das Feld "Überwachte Variable" rot hinterlegt.
3. Definieren Sie die Variable entweder in einer Variablentabelle oder in einem Datenbaustein.

Ergebnis

Die neuen Überwachungen wurden hinzugefügt.

24.15.3.2 Variablenüberwachungen im ProDiag-Übersichtseditor kopieren

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch einzelne oder mehrere Überwachungen innerhalb des Registers oder in ein anderes Register "Variablenüberwachungen" kopieren.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die gesamte Zeile einer Überwachung zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile, die Sie kopieren möchten.
   Sie können auch mehrere Zeilen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke am Ende der Tabelle.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

- Die Überwachung wird an das Ziel kopiert.
- Der kopierten Überwachung wird eine neue ID zugeordnet.
- Alle weiteren Eigenschaften der Überwachung bleiben gleich.

24.15.3.3 Variablenüberwachungen im ProDiag-Übersichtseditor löschen

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können auch einzelne oder mehrere Überwachungen löschen.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die gesamte Zeile einer Überwachung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Überwachung.
   Sie können auch mehrere Zeilen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die Überwachung ist gelöscht.

24.15.3.4 Variablenüberwachungen im ProDiag-Übersichtseditor sortieren

Beschreibung

Im Register "Variablenüberwachungen" können Sie sich nicht nur die Informationen zu den erstellten globalen Überwachungen anzeigen lassen, sondern Sie können sie auch sortieren. Die Text-Spalten werden alphabetisch sortiert.

Voraussetzung

Sie haben globale Überwachungen angelegt.

Vorgehen

Um die Überwachungen spaltenweise zu sortieren, gehen Sie folgendermaßen vor:

1. Wählen Sie eine der Spalten aus, nach der Sie sortieren möchten.
2. Klicken Sie auf den Spaltenkopf.

Ergebnis

Die Tabelle ist nach der markierten Spalte sortiert. Im Spaltenkopf erscheint ein Dreieck mit Spitze nach oben.

Durch wiederholtes Klicken auf den Spaltenkopf wird die Sortierung wie folgt geändert:

- Symbol "▲": Die Tabelle wird aufsteigend sortiert.
- Symbol "▼": Die Tabelle wird absteigend sortiert.
- Kein Symbol: Die Sortierung wird wieder aufgehoben. Die Tabelle nimmt die defaultmäßige Anzeige an.

24.15.3.5 Überwachungsdefinitionen am Funktionsbaustein im ProDiag-Übersichtseditor editieren

Beschreibung

Im Register "Überwachungsdefinitionen am FB" können Sie sich nicht nur die in der Bausteinschnittstelle von anwenderdefinierten Funktionsbausteinen definierten Überwachungen anzeigen lassen, sondern Sie können sie auch editieren.

Voraussetzung

Sie haben Überwachungen für Funktionsbausteine definiert.

Vorgehen

Um eine Überwachung zu editieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie das zu editierende Feld der Überwachung.
2. Ändern Sie die zu editierende Eigenschaft der Überwachung.

Ergebnis

Die Überwachung ist editiert.

24.15.3.6 Überwachungsdefinitionen am Funktionsbaustein im ProDiag-Übersichtseditor löschen

Beschreibung

Im Register "Überwachungsdefinitionen am FB" können Sie sich nicht nur die in der Bausteinschnittstelle von anwenderdefinierten Funktionsbausteinen definierten Überwachungen anzeigen lassen, sondern Sie können auch einzelne oder mehrere Überwachungen löschen.

Voraussetzung

Sie haben Überwachungen für Funktionsbausteine definiert.

Vorgehen

Um eine komplette Überwachung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Überwachung.
   Sie können auch mehrere Zeilen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   oder
   Drücken Sie auf Ihrer Tastatur die Taste "Entf".

Ergebnis

Die Überwachung ist gelöscht.

24.15.4 Überwachungen eines Funktionsbausteins anzeigen

Voraussetzung

In der Bausteinschnittstelle eines anwenderdefinierten Funktionsbausteins wurden Überwachungen angelegt.

Vorgehen

Um sich die Überwachungen des Funktionsbausteins anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Klicken Sie den Funktionsbaustein in der Projektnavigation mit der rechten Maustaste an und wählen Sie aus dem Kontextmenü "Eigenschaften" aus.
   Der Eigenschaftendialog des Funktionsbausteins wird geöffnet.
2. Klicken Sie das Register "Überwachungsdefinitionen am FB" an.
3. Bei Bedarf können Sie weitere Spalten in der Tabelle ein- und ausblenden. Klicken Sie dazu in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/ Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
4. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
5. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
6. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.
   Die Reihenfolge der Spalten kann nicht verändert werden.

Oder:

1. Doppelklicken Sie den Funktionsbaustein in der Projektnavigation.
   Solange Sie in der Bausteinschnittstelle nichts selektieren, werden Ihnen die Überwachungen im Inspektorfenster "Eigenschaften" im Register "Überwachungsdefinitionen am FB" angezeigt.
2. Ab hier gilt die gleiche Vorgehensweise wie ab Punkt 3 oben beschrieben.

Ergebnis

Im Register "Überwachungsdefinitionen am FB" werden alle Überwachungen des anwenderdefinierten Funktionsbausteins und die gewünschten Spalten angezeigt.

24.15.5 Überwachungen eines ProDiag-Funktionsbausteins anzeigen

Voraussetzung

Es wurde dem ProDiag-Funktionsbaustein mindestens eine Überwachung zugewiesen.

Vorgehen

Um sich alle globalen und lokalen Überwachungen eines ProDiag-Funktionsbaustein anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den gewünschten ProDiag-Funktionsbaustein. Der ProDiag-Funktionsbaustein wird geöffnet.
2. Klicken Sie entweder das Register "Variablenüberwachungen" oder "Instanziierte FB-Überwachungen" an.
3. Bei Bedarf können Sie weitere Spalten in der Tabelle ein- und ausblenden. Klicken Sie dazu in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/ Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
4. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
5. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.
6. Um mehrere Spalten ein- oder auszublenden, klicken Sie auf "Mehr" und aktivieren oder deaktivieren Sie im Dialog "Anzeigen/Verbergen" die Optionskästchen der entsprechenden Spalten.
   Die Reihenfolge der Spalten kann nicht verändert werden.

Ergebnis

Im Register "Variablenüberwachungen" werden alle globalen Überwachungen und im Register "Instanziierte FB-Überwachungen" werden alle lokalen Überwachungen angezeigt, die diesem ProDiag-Funktionsbaustein zugeordnet sind.

24.16 Überwachte anwenderdefinierte Bausteine anzeigen

24.16.1 Überwachte anwenderdefinierte Bausteine anzeigen

Sie haben die Möglichkeit, sich alle Bausteine einer CPU, in denen Sie Überwachungen angelegt haben, in der Übersicht der Projektnavigation anzeigen zu lassen.

Voraussetzung

Sie haben ProDiag-Überwachungen angelegt und Ihr Programm übersetzt.

Vorgehen

Um die Übersicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Klicken Sie auf die Schaltfläche "Maximiert/minimiert die Übersicht".
3. Klicken Sie in den Spaltenkopf und wählen Sie im Kontextmenü den Befehl "Anzeigen/ Verbergen".
4. Die Auswahl der verfügbaren Spalten wird angezeigt.
5. Aktivieren Sie das Optionskästchen der Spalte "Überwachungen", um die gewünschte Spalte anzuzeigen.

Ergebnis

Bei allen Bausteinen, die Überwachungen enthalten, wird in der Spalte "Überwachungen" das ProDiag-Symbol angezeigt.

24.17 ProDiag-Überwachungen und Eigenschaften eines ProDiag-FB exportieren und importieren

24.17.1 Überblick über das Exportieren und Importieren von Überwachungen und Eigenschaften eines ProDiag-FBs

Beschreibung

Es ist möglich, neben den angelegten Überwachungen aus den ProDiag-Übersichtstabellen "Variablenüberwachungen" und "Überwachungsdefinitionen am FB" auch die Eigenschaften eines oder mehrerer ProDiag-Funktionsbausteine zu exportieren.

Diese Überwachungen und Eigenschaften können Sie in ein standardisiertes XLSX-Format (Excel Spreadsheet) exportieren und mit externen Tabelleneditoren bearbeiten (z. B. übersetzen). Ebenso können Sie neue Überwachungen, Eigenschaften und neue ProDiag-Funktionsbausteine, die mit externen Tabelleneditoren hinzugefügt wurden, in das TIA Portal importieren. Die Überwachungen, Eigenschaften und ProDiag-Funktionsbausteine werden automatisch vom System angelegt.

Aus der ProDiag-Übersichtstabelle "Instanziierte FB-Überwachungen" können Sie die Zuordnungen von ProDiag-Funktionsbausteinen zu lokalen Überwachungen exportieren und importieren.

Format der Exportdatei

Die Benennungen der Spaltentitel sind vom System fest vorgegeben und dürfen nicht verändert werden. Es werden alle Spalten exportiert, auch wenn Sie diese in der ProDiag-Übersichtstabelle ausgeblendet haben. Jede Überwachung erhält eine separate Zeile in der Datei.

Die folgende Tabelle gibt an, welche Inhalte im Registerblatt "ProDiag Supervisions" in den einzelnen Spalten erwartet werden:

Spalte	Erläuterung
ProDiag FB	Name des zugehörigen ProDiag-Funktionsbaust-eins
ID	Nummer der Überwachung
Supervised tag / parameter	Überwachte(r) Variable / Parameter
Trigger	Signalzustand, auf den der Operand überwacht wird: • TRUE • FALSE

Spalte	Erläuterung
Type of supervision	Überwachungsart: • Operand • Verriegelung • Reaktion • Aktion • Position • Textmeldung • Fehlermeldung
Category	Kategorie: • 1 (Fehler) • 2 (Warnung) • 3 (Information)
Subcategory 1 und 2	Unterkategorien, die Sie in den ProDiag-Überwachungseinstellungen angelegt haben
Delay time	Je nach Überwachungsart, entweder: • Verzögerungszeit • Reaktionszeit • Anlaufzeit
C1, C2 oder C3 Trigger	Signalzustand, auf den die Bedingungen überwacht werden: • TRUE • FALSE
Condition 1, 2 oder 3	Operand, der als Bedingung verwendet wird
Specific text field	Spezifische Textfeld
Tag 1 (SD_4), Tag 2 (SD_5) oder Tag 3 (SD_6)	Begleitwerte des spezifischen Textfeldes, die zur Laufzeit ausgelesen werden.

Die folgende Tabelle gibt an, welche Inhalte im Registerblatt "ProDiag Settings" in den einzelnen Spalten erwartet werden:

Spalte	Erläuterung
ProDiag FB	Name des oder der zugehörigen ProDiag-Funktionsbausteins
Global enabler	Globales Freigabekriterium
Category	Kategorie: • 1 (Fehler) • 2 (Warnung) • 3 (Information)
Category enabler	Freigabekriterien der Kategorien
Acknowledge tag	Quittiervariablen einzelner Kategorien
Display class	Anzeigeklasse

Beispiel

Das folgende Bild zeigt beispielhaft einen Auszug aus der Exportdatei:

	A	B	C	D	E	F
1	ProDiag FB	ID	Supervised tag	Trigger	Type of supervision	Category
2	"Default_SupervisionFB"	3	"Conv_Actor_Forward"	False	O	1
3	"Default_SupervisionFB"	4	"Conv_Actor_Backward"	False	O	1
4	"Default_SupervisionFB"	5	"Conv_Forward_Trigger"	False	O	1
5	"Default_SupervisionFB"	6	"Conv_Backward_Trigger"	False	O	1
6	"Default_SupervisionFB"	7	"Conv_Switch_Left"	False	O	1
7	"Default_SupervisionFB"	8	"Conv_Switch_Right"	False	O	1
8	"Default_SupervisionFB"	9	"Conv_Push_Button_Forward"	False	O	1
9	"Default_SupervisionFB"	10	"Conv_Push_Button_Backward"	False	O	1

Siehe auch

Überwachungen und Eigenschaften eines ProDiag-FB exportieren (Seite 8289)

Überwachungen und Eigenschaften eines ProDiag-FB importieren (Seite 8291)

24.17.2 Überwachungen und Eigenschaften eines ProDiag-FB exportieren

Voraussetzung

Sie haben globale Überwachungen angelegt und Eigenschaften für einen oder mehrere ProDiag-Funktionsbausteine definiert.

Aus den Bausteineigenschaften werden die Überwachungseinstellungen und die folgenden Einstellungen exportiert:

- Bereich "Allgemein": Die Baustein-Version
- Bereich "Attribute": Die Optionen "Erstwerterfassung" und "Zentralen Zeitstempel verwenden"
- Bereich "Überwachungseinstellungen": Alle Einstellungen

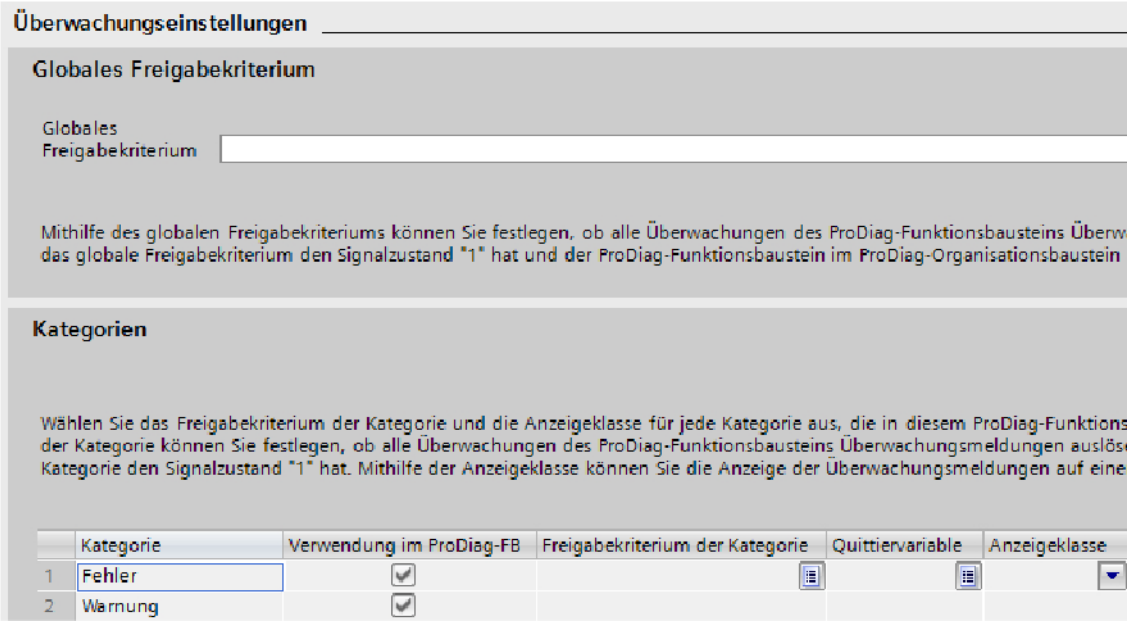

Hinweis

Export der Eigenschaften eines ProDiag-Funktionsbausteins

Die Eigenschaften eines ProDiag-Funktionsbausteins können nur aus dem Register "Variablenüberwachungen" mit exportiert werden.

Vorgehen

Um die Überwachungen und Eigenschaften eines oder mehrerer ProDiag-Funktionsbausteine zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
2. Klicken Sie das Register "Überwachungen" an.
3. Klicken Sie im Register "Variablenüberwachungen" auf "Exportiert Variablenüberwachungen" in der Funktionsleiste.
   Der Dialog "Speichern unter" wird geöffnet.
4. Wählen Sie einen Speicherort aus und vergeben Sie einen Dateinamen.

5. Klicken Sie auf "Speichern".
   Wenn das Speichern erfolgreich war, dann wird der Dialog "Export abgeschlossen" geöffnet.
6. Bestätigen Sie mit "OK".

oder:

1. Öffnen Sie den ProDiag-Funktionsbaustein mit einem Doppelklick.
2. Klicken Sie auf "Exportiert Variablenüberwachungen" in der Funktionsleiste. Aus dem ProDiag-Funktionsbaustein können Sie nur globale Überwachungen exportieren.
   Der Dialog "Speichern unter" wird geöffnet.
3. Wählen Sie einen Speicherort aus und vergeben Sie einen Dateinamen.
4. Klicken Sie auf "Speichern".
   Sobald das Speichern erfolgreich war, wird der Dialog "Export abgeschlossen" geöffnet.
5. Bestätigen Sie mit "OK".

Ergebnis

Am ausgewählten Speicherort wird eine .xlsx-Datei angelegt:

- Im Registerblatt "ProDiag Supervisions" finden Sie alle exportierten Überwachungen.

	A	B	C	D	E	F
1	ProDiag FB	ID	Supervised tag	Trigger	Type of supervision	Category
2	"Default_SupervisionFB"	1	"Monday"	False	O	1
3	"Default_SupervisionFB"	2	"Tuesday"	False	R	1
4	"Station_1"	1	"Wednesday"	False	P	1

- Im Registerblatt "ProDiag Settings" finden Sie alle ProDiag-spezifischen Eigenschaften, die Sie für die ProDiag-Funktionsbausteine definiert haben.

	A	B	C	D	E
1	ProDiag FB	Version	Initial value acquisition	Use central time stamp	Global enabler
2	"Default_SupervisionFB"	V2.0	False	False	
3	"Station_1"	V2.0	False	True	"Wednesday"
4	ProDiag FB	Category	Category enabler	Acknowledge tag	Display class
5	"Station_1"	1	"Tuesday"		0

24.17.3 Überwachungen und Eigenschaften eines ProDiag-FB importieren

Voraussetzung

Sie haben Überwachungen und Eigenschaften eines ProDiag-Funktionsbausteins in eine .xlsx-Datei exportiert.

Vorgehen

Um die Überwachungen und Eigenschaften eines oder mehrerer ProDiag-Funktionsbausteine zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
2. Klicken Sie das Register "Überwachungen" an.
3. Klicken Sie im Register "Variablenüberwachungen" auf "Importiert Variablenüberwachungen" in der Funktionsleiste.
   Der Dialog "Importiert Variablenüberwachungen" wird geöffnet.
4. Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist.
   Der Dialog "Öffnen" wird geöffnet.
5. Wählen Sie die gewünschte .xlsx-Datei aus.
6. Bestätigen Sie den Dialog mit "Öffnen".
7. Im Dialog "Importiert Variablenüberwachungen" haben Sie verschiedene Möglichkeiten:
   - "Überwachungen importieren"
     Wenn Sie die Optionen "Überwachungen importieren" und "Existierende Überwachungen löschen" aktivieren, dann werden die bereits bestehenden Überwachungen von ProDiag-Funktionsbausteinen, zu denen es in der .xlsx-Importdatei auch Überwachungen gibt, in der Übersichtstabelle gelöscht und die neu erstellten Überwachungen aus der .xlsx-Datei hinzugefügt. Wenn es zu einem ProDiag-Funktionsbaustein keine Überwachungen in der .xlsx-Importdatei gibt, dann wird dieser auch vom Löschen ausgenommen.
     Wenn Sie die Option "Existierende Überwachungen löschen" deaktivieren, dann bleiben die bereits vorhandenen Überwachungen bestehen und die neu erstellten Überwachungen werden hinzugefügt.
   - "Überwachungseinstellungen importieren und überschreiben"
     Wenn Sie die Option "Überwachungseinstellungen importieren und überschreiben" aktivieren und der ProDiag-Funktionsbaustein bereits Überwachungseinstellungen enthält, dann werden die alten Einstellungen mit den neuen Einstellungen überschrieben.
8. Klicken Sie auf die Schaltfläche "Importieren".
   Der Dialog "Variablenüberwachungen importieren" wird geöffnet. Der Dialog zeigt den aktuellen Status des Imports an.
9. Wenn das Importieren erfolgreich war, dann wird der Dialog "Import abgeschlossen" geöffnet.
10. Wenn es während des Imports zu Fehlern kommt, dann wird der Dialog "Import abgeschlossen mit Warnungen" geöffnet.
    Es können folgende Fehler auftreten:
    - Es gibt fehlerhafte Einträge in der .xlsx-Datei.
    - Die maximale Anzahl an Überwachungen für einen ProDiag-Funktionsbaustein wurde überschritten.
    - Der in der .xlsx-Datei angegebene Funktionsbaustein ist in Ihrem Programm nicht enthalten. (Kann nur beim Import von Überwachungen innerhalb der Übersichtstabelle "Überwachungsdefinitionen am FB" auftreten.)

11. Klicken Sie in diesem Fall auf "Hier klicken, um die Protokoll-Datei einzusehen".
12. Bestätigen Sie mit "OK".

Oder:

1. Öffnen Sie den ProDiag-Funktionsbaustein mit einem Doppelklick.
2. Klicken Sie auf "Importiert Variablenüberwachungen" in der Funktionsleiste.
   Der Dialog "Importiert Variablenüberwachungen" wird geöffnet.
3. Alle weiteren Schritte wie oben ab Schritt 4 beschrieben befolgen.

Ergebnis

In der ProDiag-Übersichtstabelle werden die importierten Überwachungen und in den ProDiag-Funktionsbausteinen die Bausteineigenschaften entsprechend der vorher exportierten Eigenschaften angezeigt.

Für globale Überwachungen gilt folgendes:

Wenn Sie in der .xlsx-Datei neue Überwachungen hinzugefügt haben, dann werden diese automatisch nach dem Import dem entsprechenden ProDiag-Funktionsbaustein zugewiesen.

Für lokale Überwachungen gilt folgendes:

Bei lokalen Überwachungen werden zunächst nur die Überwachungsdefinitionen importiert und erst durch den Import der Zuweisungen werden die Überwachungen dann auch konkreten ProDiag-Funktionsbausteinen zugeordnet.

24.18 Zuweisungen von ProDiag-FBs zu Instanz-DBs exportieren und importieren

24.18.1 Überblick über das Exportieren und Importieren von Zuweisungen

Beschreibung

Es ist möglich, die angelegten Zuweisungen von ProDiag-Funktionsbausteinen zu den Instanz-Datenbausteinen anwenderdefinierter Funktionsbausteine aus der ProDiag-Übersichtstabelle "Instanziierte FB-Überwachungen" zu exportieren.

Diese Zuweisungen können Sie in ein standardisiertes XLSX-Format (Excel Spreadsheet) exportieren und mit externen Tabelleneditoren bearbeiten (z. B. übersetzen). Ebenso können Sie neue Zuweisungen, die mit externen Tabelleneditoren hinzugefügt wurden, in das TIA Portal importieren.

Format der Exportdatei

Die Benennungen der Spaltentitel sind vom System fest vorgegeben und dürfen nicht verändert werden. Jede Zuweisung erhält eine separate Zeile in der Datei.

Die folgende Tabelle gibt an, welche Inhalte in den einzelnen Spalten erwartet werden:

Spalte	Erläuterung
Instance	Name des zugehörigen Instanz-Datenbausteins eines anwenderdefinierten Funktionsbausteins
ProDiag FB	Name des ProDiag-Funktionsbausteins

Beispiel

Das folgende Bild zeigt beispielhaft einen Auszug aus der Exportdatei:

1	Instance	ProDiag FB
2	"LAD_FB_1500_DB"	"Default_SupervisionFB"
3	"FBD_FB_1500_DB"	

Siehe auch

Zuweisungen exportieren (Seite 8295)

Zuweisungen importieren (Seite 8295)

24.18.2 Zuweisungen exportieren

Voraussetzung

Sie haben lokale Überwachungen angelegt und den Funktionsbaustein im Programmcode aufgerufen.

Vorgehen

Um die Zuweisungen zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
2. Klicken Sie im Register "Überwachungen" das Register "Instanziierte FB-Überwachungen" an.
3. Klicken Sie auf "Exportiert Zuweisungen von ProDiag-FBs zu Instanz-DBs" in der Funktionsleiste.
   Der Dialog "Speichern unter" wird geöffnet.
4. Wählen Sie einen Speicherort aus und vergeben Sie einen Dateinamen.
5. Klicken Sie auf "Speichern".
   Wenn das Speichern erfolgreich war, dann wird der Dialog "Export abgeschlossen" geöffnet.
6. Bestätigen Sie mit "OK".

Ergebnis

Am ausgewählten Speicherort wird eine .xlsx-Datei angelegt.

24.18.3 Zuweisungen importieren

Voraussetzung

Sie haben Zuweisungen in eine .xlsx-Datei exportiert.

Vorgehen

Um die Zuweisungen zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Überwachungen & -Meldungen".
2. Klicken Sie im Register "Überwachungen" das Register "Instanziierte FB-Überwachungen" an.
3. Klicken Sie auf "Importiert Zuweisungen von ProDiag-FBs zu Instanz-DBs" in der Funktionsleiste.
   Der Dialog "Importiert Zuweisungen von ProDiag-FBs zu Instanz-DBs" wird geöffnet.

4. Wählen Sie den Pfad aus, in dem die zu importierende Datei abgelegt ist.
   Der Dialog "Öffnen" wird geöffnet.
5. Wählen Sie die gewünschte .xlsx-Datei aus.
6. Bestätigen Sie den Dialog mit "Öffnen".
7. Klicken Sie auf die Schaltfläche "Importieren".
   Der Dialog "Importiert Zuweisungen von ProDiag-FBs zu Instanz-DBs" wird geöffnet. Der Dialog zeigt den aktuellen Status des Imports an.
8. Wenn das Importieren erfolgreich war, dann wird der Dialog "Import abgeschlossen" geöffnet.
9. Wenn es während des Imports zu Fehlern kommt, dann wird der Dialog "Import abgeschlossen mit Warnungen" geöffnet.
   Es können folgende Fehler auftreten:
   – Es gibt fehlerhafte Einträge in der .xlsx-Datei.
   – Es gibt fehlende Zuweisungen in der .xlsx-Datei.
10. Klicken Sie in diesem Fall auf "Hier klicken, um die Protokoll-Datei einzusehen".
11. Bestätigen Sie mit "OK".

Ergebnis

In der ProDiag-Übersichtstabelle werden die importierten Zuweisungen angezeigt.

24.19 ProDiag-Querverweise anzeigen

24.19.1 Übersicht zu Querverweisen

Beschreibung

Die Querverweisliste bietet Ihnen innerhalb Ihres Anwenderprogramms einen Überblick über die Verwendungsstellen von überwachten Variablen und Variablen, die z. B. als Bedingung innerhalb einer ProDiag-Überwachung verwendet werden.

Sie können aus der Querverweisliste direkt zur Verwendungsstelle einer überwachten Variablen springen.

Weitere Informationen zur Anzeige von Querverweisen finden Sie hier: Allgemeine Informationen zu Querverweisen (Seite 7989)

24.19.2 Querverweise anzeigen

Beschreibung

Die Querverweisinformationen zu einer selektierten überwachten Variable (globale Überwachungen) werden im Inspektorfenster im Register "Info > Querverweisliste" und in der Querverweisliste im Programmierfenster angezeigt. Für jede selektierte überwachte Variable wird gezeigt, welchem ProDiag-Funktionsbaustein sie zugeordnet ist und an welchen Stellen sie verwendet wird. Wenn die selektierte Variable nicht nur überwacht wird, sondern z. B. auch als Bedingung innerhalb einer anderen ProDiag-Überwachung verwendet wird, dann wird diese Verwendung ebenfalls aufgelistet. Eine überwachte Variable erkennen Sie daran, dass in der Spalte "Zugriff" die Kennung "PRODIAG Supervision" steht. Wenn die Variable als Bedingung verwendet wird, dann wird diese in der Spalte "Zugriff" mit der Kennung "Read-only" gekennzeichnet.

An den folgenden Stellen im TIA Portal können Sie sich die Querverweise anzeigen lassen:

- Standard-Variablentabelle
- ProDiag-Übersichtseditor
  - Register "Variablenüberwachungen"
- ProDiag-Funktionsbaustein
  - Register "Variablenüberwachungen"

Vorgehen

Um sich die Querverweise im Inspektorfenster anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Klicken Sie die überwachte Variable, z. B. in der Standard-Variablentabelle, an.
   Im Inspektorfenster im Register "Info > Querverweisliste" wird Ihnen die überwachte Variable und der zugewiesene ProDiag-Funktionsbaustein angezeigt.
2. Klicken Sie auf den Pfeil links neben dem ProDiag-Funktionsbaustein.
   Es werden Ihnen alle Verwendungsstellen (Überwachungs-IDs) dieser Variable angezeigt.
3. Klicken Sie auf eine der angezeigten Verwendungsstellen, um direkt dorthin zu springen.
   Der zugehörigen ProDiag-Funktionsbaustein wird geöffnet und die entsprechende Zelle markiert.

Hinweis

Markierung einer Zelle

Wenn keine Zelle markiert wird, dann ist die entsprechende Spalte, die die Variable enthält, nicht eingeblendet.

24.20 Erstwerte erfassen

24.20.1 Übersicht über die Erstwerterfassung

Voraussetzung

Damit Sie die Erstwerterfassung aktivieren können, müssen Sie im ProDiag-Funktionsbaustein statt der Version 1.0 die Version 2.0 einstellen.

Die Erstwerterfassung für ProDiag steht Ihnen in den Programmiersprachen KOP und FUP zur Verfügung.

Folgende Voraussetzungen müssen im Anwenderprogramm erfüllt sein, damit die Erstwerte erfasst werden können:

- Es können max. 32 Signalzustände aufgezeichnet werden. Wenn ein KOP-/FUP-Netzwerk mehr als 32 Elemente hat, dann findet keine Aufzeichnung statt.
- Bei globalen Überwachungen: Das Anwenderprogramm darf maximal einen Schreibzugriff auf die zu überwachende Variable haben und dieser Schreibzugriff muss an der rechten Stromschiene platziert sein.
- Bei lokalen Überwachungen: Für jede neue Instanz wird eine eigene Instanz-Überwachung angelegt. Wenn jedoch im aufrufenden Baustein lokale Variablen als Vorverschaltung verwendet werden, funktioniert die Erstwertaufzeichnung nur dann, wenn die zugehörige Instanz des aufrufenden Bausteins den Namen "FBName_DB" hat und ausschließlich die Abschnitte Input, Output und Static in der Bausteinschnittstelle verwendet wurden.

Weitere Informationen finden Sie hier: Erstwerterfassung aktivieren (Seite 8302)

Aufzeichnung der Erstwerte

Die Erstwerte können für folgende Überwachungsarten aufgezeichnet werden:

Überwa-chungsarten	Globale Überwachungen			Lokale Überwachungen		
	Input	Output	Merker	Input	Output	Static
Operand	-	-	-	x	-	-
Verriegelung	-	x	x	x	-	-
Aktion	-	-	-	x	-	-
Reaktion	-	-	-	x	-	-
Position	-	-	-	x	-	-
Textmeldung	-	-	-	-	-	-
Fehlermeldung	-	-	-	-	-	-

Beschreibung

Mithilfe der Erstwerterfassung haben Sie die Möglichkeit zu analysieren, warum eine Überwachungsmeldung ausgelöst wurde. Dafür werden die Erstwerte der Operanden aufgezeichnet, die im Anwenderprogramm vor dem überwachten Operanden programmiert sind.

Sie haben die Möglichkeit, den Ablauf des Anwenderprogramms auf einem HMI-Gerät zu verfolgen. Dabei werden die Daten und Werte auf dem Bediengerät stetig mit der CPU abgeglichen und aktualisiert. Für jeden KOP- oder FUP-Baustein sehen Sie immer den aktuellen Programmstatus auf dem Bediengerät.

Bei globalen Überwachungen werden die Erstwerte aller booleschen Operanden aufgezeichnet, die sich im Netzwerk befinden. Wenn das Netzwerk mehrere einzelne Stromschienen enthält, die nicht miteinander verbunden sind, dann werden nur die Erstwerte der jeweils betroffenen Stromschiene aufgezeichnet.

Bei lokalen Überwachungen werden nur die Erstwerte aufgezeichnet, die beim Bausteinaufruf als Bedingungen für den überwachten Parameter angegeben sind.

Sobald ein Ereignis auftritt, das eine Überwachungsmeldung auslöst, werden die Erstwerte aufgezeichnet. Die Erstwerte werden im ProDiag-Instanzdatenbaustein im statischen Parameter CRIT_ERR in einem DWORD gespeichert und jeder einzelne Signalzustand belegt ein Bit. Sie werden in dem Zyklus aufgezeichnet, in dem das Ereignis aufgetreten ist. Die Überwachungsmeldung wird dann im nächsten Zyklus verschickt. Es können max. 32 Signalzustände pro Netzwerk innerhalb eines KOP- oder FUP-Bausteins aufgezeichnet werden. Die Erstwerte werden bis zum Auftreten eines neuen Ereignisses im aktiven Netzwerk nicht mehr verändert.

Mithilfe der PLC-Code-Anzeige können Sie sich die Aktualwerte anzeigen lassen. Sobald ein Ereignis auftritt, können Sie sich zusätzlich auch die Erstwerte anzeigen lassen und die fehlerhaften Operanden sofort erkennen (Kriterienanalyse). Sie können zwischen der Anzeige der Aktualwerte und der Anzeige der Erstwerte hin und her schalten.

Für die Kriterienanalyse auf dem HMI-Gerät werden die Operanden immer von rechts an der Spule beginnend entgegen des Stromlaufs nach links ausgelesen.

Weitere Informationen zum Visualisieren auf dem HMI-Gerät finden Sie hier: "Prozesse visualisieren > Diagnosefunktionen nutzen > Maschinen und Anlagen mit ProDiag überwachen"

Unterstützte Anweisungen

Für die Erstwerterfassung werden folgende Anweisungen in KOP und FUP unterstützt:

Anweisungen	Anzeige auf dem Bediengerät (HMI)
Bitverknüpfungen	
Schließerkontakt	Erstwerte und Kriterienanalyse
Öffnerkontakt	
VKE invertieren	Die Anweisung wird unterstützt, ist aber nicht relevant für die Erstwerte bzw. die Kriterienanalyse.
Zuweisung	
Zuweisung negieren	
Ausgang rücksetzen	Erstwerte
Ausgang setzen	
Flipflop setzen/rücksetzen	Erstwerte und Kriterienanalyse bis einschließlich der Anweisungsbox
Flipflop rücksetzen/setzen	
AND (FUP)	Erstwerte und Kriterienanalyse
OR (FUP)	
Vergleicher	

Anweisungen	Anzeige auf dem Bediengerät (HMI)
Gleich	Erstwerte und Kriterienanalyse
Ungleich	
Größer gleich	
Kleiner gleich	
Größer	
Kleiner	
Zeiten	
TP	Erstwerte und Kriterienanalyse bis einschließlich der Anweisungsbox
TON	Erstwerte und Kriterienanalyse
TOF	Erstwerte und Kriterienanalyse bis einschließlich der Anweisungsbox
TONR	
Zähler	
CTU	Erstwerte und Kriterienanalyse bis einschließlich der Anweisungsbox
CTD	
CTUD	

Bei den Bitverknüpfungen wird der Zustand des Operanden und bei den Vergleichern das Ergebnis des Vergleichs aufgezeichnet.

Bei den Flipflops werden beide Eingänge (R und S) aufgezeichnet, wenn sie verschaltet sind.

Bei den Zeiten und Zählern wird der Zustand des Operanden am Ausgang und zusätzlich die Eingänge, wenn sie verschaltet sind, aufgezeichnet. (Zum Beispiel bei CTUD: CU, CD, R, LD)

Auswertungsreihenfolge auf dem HMI-Gerät im der PLC-Code-Anzeige

Die Aufzeichnung der Signalzustände im jeweiligen DWORD bzw. die Auflistung der Operanden in der Symboltabelle der PLC-Code-Anzeige findet in einer fest definierten Reihenfolge statt. Dabei wird z. B. in einem KOP-Netzwerk immer an der Stromschiene links oben begonnen und endet rechts unten.

Hinweis

Auswertungsreihenfolge bei der Programmierung beachten

Auch bei der Kriterienanalyse auf dem HMI-Gerät wird diese feste Reihenfolge bei der Auflistung der fehlerhaften Operanden in der Symboltabelle eingehalten. Beachten Sie diese Auswertungsreihenfolge bei der Programmierung der Netzwerke.

Weitere Informationen zur Reihenfolge der Auflistung finden Sie hier: Beispiel für die Reihenfolge der Auflistung (Seite 8302)

Siehe auch

Aufbau eines ProDiag-Instanzdatenbausteins der Version 2.0 (Seite 8207)

Inhalt eines ProDiag-Instanzdatenbausteins der Version 2.0 (Seite 8208)

24.20.2 Erstwerterfassung aktivieren

Vorgehen

Um die Erstwerterfassung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den ProDiag-Funktionsbaustein und wählen Sie aus dem Kontextmenü "Eigenschaften...".
   Der Eigenschaftendialog des ProDiag-Funktionsbausteins wird geöffnet.
2. Wählen Sie im Register "Allgemein > Baustein" die Version V2.0 aus.
3. Klicken Sie das Register "Attribute" an.
4. Aktivieren Sie die Option "Erstwerterfassung (für die Kriterienanalyse auf Bediengeräten)".
5. Klicken Sie auf "OK".
6. Übersetzen und laden Sie das Anwenderprogramm.

Ergebnis

Sobald das Anwenderprogramm übersetzt wurde, ist die Erstwerterfassung aktiviert. Das bedeutet, dass im Fehlerfall die entsprechenden Signalzustände aufgezeichnet werden.

24.20.3 Beispiel für die Reihenfolge der Auflistung

Die Aufzeichnung der Aktual- bzw. Erstwerte und die Auflistung der Operanden in der Symboltabelle der PLC-Code-Anzeige unterliegen der gleichen fest definierten Reihenfolge.

Anhand der folgenden Beispiele sehen Sie, wie die Aufzeichnung der Signalzustände innerhalb eines KOP-Netzwerks und die Auflistung der Operanden auf dem HMI-Gerät funktionieren.

Aufzeichnung der Signalzustände im Fehlerfall

Die Signalzustände werden in der folgenden Reihenfolge in den statischen Parameter CRIT_ERR geschrieben:

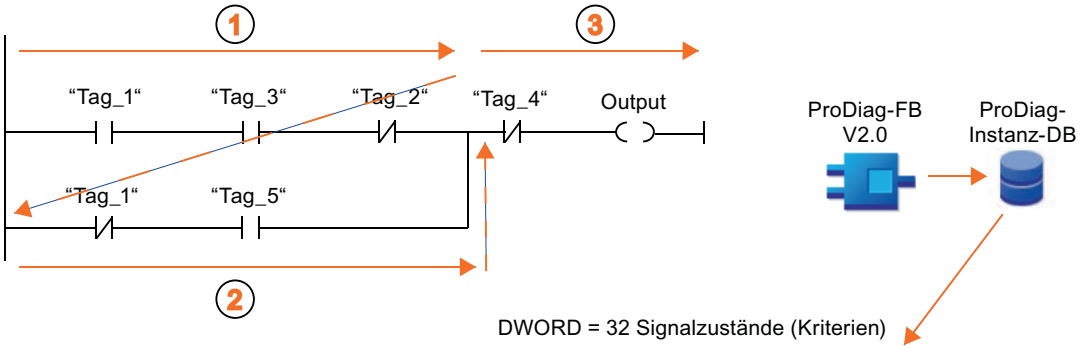

Bit 31	Bit ...	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	Tag_4	Tag_5	Tag_1	Tag_2	Tag_3	Tag_1

Auf dem HMI-Gerät werden alle Operanden genau nach dieser Reihenfolge in der Symboltabelle der PLC-Code-Anzeige aufgelistet:

Symbolname	Operand	Kommentar
Tag_1	z. B. %M2xy	Automatikbetrieb / Handbetrieb
Tag_3		Motor einschalten
Tag_2		Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster
Tag_4		Lichtschranke

Auflistung der Operanden im Fehlerfall

Wenn nun im Netzwerk Fehler auftreten, dann werden die fehlerhaften Operanden farblich markiert und in der Symboltabelle werden nur noch die fehlerhaften Operanden aufgelistet:

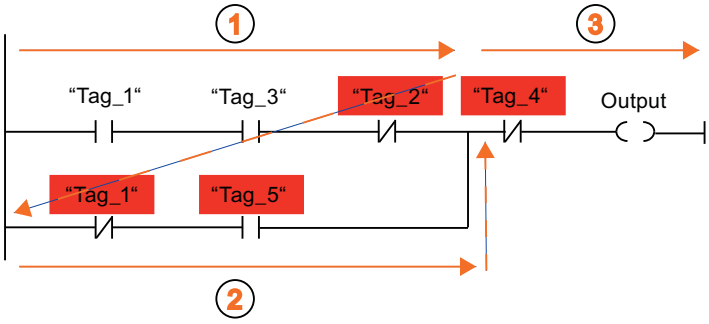

Auflistung der fehlerhaften Operanden in der Symboltabelle der PLC-Code-Anzeige:

Symbolname	Operand	Kommentar
Tag_2	z. B. %M2xy	Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster
Tag_4		Lichtschranke

Optimierte Programmierung des Netzwerks im Fehlerfall

Damit Sie schnell auf den ersten Blick erkennen können, dass die Lichtschranke für den Fehler verantwortlich ist, ist es in diesem Beispiel von Vorteil, den Operanden "Tag_4: Lichtschranke" an erster Stelle zu programmieren:

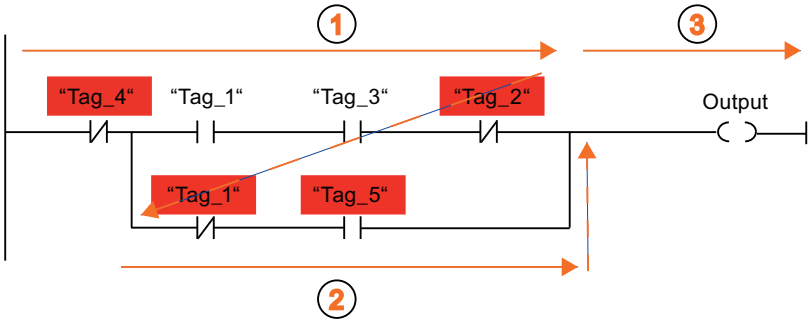

Damit wird die Information zur offenen Lichtschranke in der ersten Zeile der Symboltabelle dargestellt:

Symbolname	Operand	Kommentar
Tag_4	z. B. %M2xy	Lichtschranke
Tag_2		Verriegelung_1
Tag_1		Automatikbetrieb / Handbetrieb
Tag_5		Handtaster

Index

-

-, 7609
-D, 1598, 4978
-I, 1595, 4974
-R, 1606, 4987

#

#, 7275

&

&, 7617, 7618

(

(), 7618

*

*, 7609
**, 7609
*D, 1599, 4979
*I, 1596, 4975
*R, 1607, 4988

/

/, 7609
/D, 1600, 4980
/ER, 183
/I, 1596, 4975
/R, 1607, 4988

:

:=, 7618, 7620

+

+, 7609
+AR1, 1721, 5104
+AR2, 1722, 5105
+D, 1598, 4977
+I, 1594, 4973
+R, 1605, 4986

<

<, 7612
<=, 7612
<>, 7612, 7618

=

=, 7612
==, 7618

>

>, 7612
>=, 7612

7

7-Segment-Anzeige, 972, 1332, 1979, 4586, 4810, 4914, 5263, 5389

A

A, 1542, 4917
A(, 1550, 4926
A_DEAD_B
    Anweisung, 5959
    Ausgangsparameter, 5963
    Eingangsparameter, 5962
A0, 183, 188, 189, 190, 192, 193, 196, 201
A1, 183, 188, 189, 190, 192, 193, 196, 201
AB, 7376
Abfragearchitektur, 4098, 6502
Abfragearchitektur Master, 4098, 6502
Abfragearchitektur Slave, 4099, 6502
Abhängigkeitsstruktur, 7977
    Ansichtsoptionen einstellen, 7980
    anzeigen, 7979
    Aufbau, 7978
    Bedeutung der Symbole, 7979
    Einführung, 7977
Ablaufkette
    beenden, 7755

einfügen, 7752
Elemente, 7688
Grundlagen, 7687
initialisieren, 8045
kommentieren, 7753
löschen, 7756
neue Elemente einfügen, 7754
umbenennen, 7752
ABS, 766, 1125, 1608, 1781, 2006, 4448, 4669, 4989, 5148, 5286
ABS_CTRL, 6143
Arbeitsweise, 6143
Aufruf, 6143
Aufträge, 6160
Auftragsstatus, 6163
Beschreibung, 6143
Fehlerverhalten, 6145
Funktionsschalter, 6161
Rückgabewerte, 6144
Steuersignale, 6144
Verwendete Datenbausteine, 6144
ABS_CTRL_451, 6147
Arbeitsweise, 6147
Aufruf, 6147
Aufträge, 6163
Beschreibung, 6147
Fehlerverhalten, 6149
Funktionsschalter, 6163
Parameter, 6148
Rückgabewerte, 6148
Steuersignale, 6148
Verwendete Datenbausteine, 6147
ABS_DIAG, 6145
Anlauf, 6146
Arbeitsweise, 6145
Aufruf, 6145
Aufträge, 6146
Beschreibung, 6145
Diagnosedaten, 6146
Fehlerverhalten, 6146
Rückgabewerte, 6146
Verwendete Datenbausteine, 6145
ABS_DIAG_451, 6149
Anlauf, 6150
Arbeitsweise, 6149
Aufruf, 6149
Aufträge, 6150
Beschreibung, 6149
Diagnosedaten, 6150
Fehlerverhalten, 6150
Parameter, 6150
Rückgabewerte, 6150
Verwendete Datenbausteine, 6149
ABS_INIT, 6142
Aufruf, 6142
Beschreibung, 6142
Parameter, 6143
Rückgabewerte, 6143
Verwendete Datenbausteine, 6142
Absolute Operanden ein- und ausblenden, 7635
Absolutwert, 766, 1125, 1608, 1781, 2006, 4448, 4669, 4989, 5148, 5286
Achs-DB
Aufgabe, 6109
Inhalt, 6109
ACOS, 784, 1144, 1615, 1796, 2019, 4465, 4687, 4997, 5160, 5299
ACT_TINT, 2558
AD, 1697, 5078, 7376
ADD, 755, 1115, 4440, 4661
Addieren, 755, 1115, 1594, 1598, 1601, 1605, 4661, 4973, 4977, 4981, 4986
Zeiten addieren mit T_ADD, 2182, 5405
ADM_ZONE
Anweisung, 6324
Ausgangsparameter, 6326
Durchgangsparameter, 6326
Statische Variablen, 6327
Adressen
einer Baugruppe ermitteln mit GEO_LOG, 2862, 5789
MAC-Adresse auslesen mit GetStationInfo, 2707
Stationsadresse auslesen mit GetStationInfo, 2707
Adressieren von Operanden, 108, 109, 111, 113, 114, 115, 116, 119, 120, 126, 127, 131, 132, 133, 135, 136, 137, 138, 139, 140, 283, 290, 294, 303
Adressregister, 132, 133, 139, 140, 1721, 1722, 5104, 5105
Adressumwandlung
Anweisungen zur, 2849
AG_CNTEX, 6792
AG_CNTRL, 6807
AG_LOCK, 6788
AG_LRECV, 6929
AG_LSEND, 6922
AG_RECV / AG_LRECV / AG_SRECV, 6782
AG_SEND / AG_LSEND / AG_SSEND, 6777
AG_UNLOCK, 6790
AG-RECV, 6929
AG-SEND, 6922
Akkumulator, 1718, 1719, 1720, 5099, 5100, 5101, 5103, 7557

Aktion
ändern, 7778
ausschneiden, 7779
Bausteinaufruf, 7701
einfügen, 7773, 7779
Einführung, 7693
Ereignisabhängige Aktionen, 7698
Grundlagen, 7695
kopieren, 7779
löschen, 7781
Standardaktionen, 7696
verschieben, 7780
Zähler, 7704
Zeiten, 7702
Aktualisierung
DP-Verschaltung, 7072
PROFINET CBA-Komponenten, 7072
Alarme
Alarmstatus lesen mit Systemzustandsliste, 5706
eines I-Slave senden mit SALRM, 5507
empfangen von Peripheriebaugruppe mit RALRM, 2367, 5466
Prozessalarm auslösen mit DP_PRAL, 5514
Uhrzeitalarm abfragen mit QRY_TINT, 2560, 5610
Uhrzeitalarm aktivieren mit ACT_TINT, 2558, 5609
Uhrzeitalarm stellen mit SET_TINT, 2552, 5607
Uhrzeitalarm stellen mit SET_TINTL, 2555
Uhrzeitalarm stornieren mit CAN_TINT, 2557, 5608
Verzögerungsalarm abfragen mit QRY_DINT, 2573, 5614
Verzögerungsalarm starten mit SRT_DINT, 2570, 5613
Verzögerungsalarm stornieren mit CAN_DINT, 2572, 5614
Alarmereignis
freigeben mit EN_AIRT, 2612, 5630
freigeben mit EN_IRT, 2605, 5628
sperren mit DIS_IRT, 2603, 5626
verzögern mit DIS_AIRT, 2611, 5629
Alle remanenten Daten zurücksetzen, 919, 1280, 1502, 1937, 2107
Alternativzweig
Einführung, 7689
erstellen, 7767
AN, 1543, 4919
AN(, 1551, 4927
ANALOG, 6346
Analogwert, 884, 887, 1244, 1246, 1473, 1475, 1891, 1893, 2085, 2087
Analogwertverarbeitung, 884, 887, 1244, 1246, 1473, 1475, 1891, 1893, 2085, 2087
AND, 926, 978, 1286, 4506, 4592, 4728, 7617, 7618
Anlauf
ABS_DIAG, 6146
ABS_DIAG_451, 6150
Anweisungen zur Erzeugung von PLC-Meldungen,
CAM_DIAG, 6178
CAM_DIAG_452, 6182
CAM_MSRM_452, 6184
CPU, 6144, 6148, 6177, 6180
FM 351, 6144
FM 352, 6177, 6180
FM 451, 6148
FM 452, 6177, 6180
Anlaufverhalten
CNT_CTL1, 6116
CNT_CTRL, 6113
Anordnung der Bytereihenfolge, 841, 1201, 1839, 2058, 5301
Ansichtsoptionen
für den Belegungsplan, 7964
für die Abhängigkeitsstruktur einstellen, 7980
für die Aufrufstruktur einstellen, 7974
Anweisung, 6917
AG_SEND / AG_LSEND / AG_SSEND, 6777
ausschneiden, 7675
Datentyp festlegen, 7444, 7445, 7511, 7512, 7578
einfügen, 7675
kopieren, 7674
löschen, 7675
Regeln, 7642
suchen, 7133
Versionen, 7137
Anweisung PNIO_ALARM
Formalparameter, 6840
Anweisung PNIO_RECV
Formalparameter, 6827
Anweisung PNIO_RW_REC
Formalparameter, 6836
Anweisung PNIO_SEND
Formalparameter, 6820
Anweisungen
AG_LOCK / AG_UNLOCK, 6786
AG_RECV / AG_LRECV / AG_SRECV, 6782
Allgemeine Hinweise, 6917
Automatisierungssystem S7-400, 6693
für AG-AG-Verbindung, 6920
für DP-Betriebsart, 6934
für projektiert Verbindungen, 6774

P3964_Config (Protokollkonfiguration), 4112, 6515
Port_Config (Portkonfiguration), 4101, 6505
Receive_Config (Empfangskonfiguration), 4106, 6509
Receive_P2P (Punkt-zu-Punkt-Daten empfangen), 4117, 6520
Receive_Reset (Empfänger zurücksetzen), 4119, 6522
Send_Config (Sendekonfiguration), 4105, 6508
Send_P2P (Punkt-zu-Punkt-Daten senden), 4113, 6516
Signal_Get (RS232-Signale abrufen), 4120, 6523
Signal_Set (RS232-Signale setzen), 4122, 6525
Symbolischer Name, 6917
USS_Drive_Control / USS_Drive_Control_31, 4149, 6552
USS_Port_Scan, 4145, 6548
USS_Port_Scan_31, 4145, 6548
USS_Read_Param / USS_Read_Param_31, 4153, 6556
USS_Write_Param / USS_Write_Param_31, 4155, 6558
Anweisungen zur Zugriffskoordinierung bei FETCH/WRITE, 6786
Anweisungsprofil
aktivieren und deaktivieren, 7145
bearbeiten, 7144
erzeugen, 7142
Grundlagen, 7141
löschen, 7145
öffnen, 7143
Anwenderdiagnosemeldung
anlegen, 7829
bearbeiten, 7830
löschen, 7830
Anwenderprogramm
Fehler suchen, 7154
Funktion, 36
testen, 8025
Anwenderseiten synchronisieren, 4002
ANY, 359
Anzeige der Beobachtungswerte für Strukturen im Inspektorfenster, 8037
Anzeige einer Forcetabelle, 8120
Anzeigen
Abhängigkeitsstruktur, 7979
Aufrufstruktur, 7973
Belegungsplan, 7963
der Programminformationen, 7959
Kommunikations-Anweisungen für nichtprojektierte S7-Verbindungen, 6377
Ladespeicher, 7987
maximal verfügbaren Ladespeicher, 7987
Speicherauslastung der CPU, 7987
von Querverweisen, 7998
Applikationszyklus, 3589
Arbeitsspeicher, 7983
Arbeitsweise
ABS_CTRL, 6143
ABS_CTRL_451, 6147
ABS_DIAG, 6145
ABS_DIAG_451, 6149
CAM_CTRL, 6176
CAM_CTRL_452, 6179
CAM_DIAG, 6178
CAM_DIAG_452, 6181
CNT_CTL1, 6115
CNT_CTL2, 6117
CNT_CTRL, 6112
CNT2_CTR, 6129
CNT2RDPN, 6129
CNT2WRPN, 6128
DIAG_INF, 6114
DIAG_RD, 6130
High_Speed_Counter, 2883
MC_Control, 6066
MC_GearIn, 6068
MC_Home, 6059
MC_Init, 6050
MC_MoveAbsolute, 6051
MC_MoveJog, 6056
MC_MoveRelative, 6054
MC_Simulation, 6067
MC_StopMotion, 6064
SSI_Absolute_Encoder, 2894
Archiv
Daten senden mit AR_SEND, 5671
Arcuscosinus, 784, 1144, 1615, 1796, 2019, 4465, 4687, 4997, 5160, 5299
Arcussinus, 782, 1142, 1613, 1795, 2018, 4463, 4685, 4995, 5159, 5298
Arcustangens, 1617, 1797, 4999, 5161
Arcustangenswert, 785, 1145, 2020, 4466, 4688, 5300
ARRAY, 127, 138, 211
Adressieren, 114, 303
Beispiel, 310
Deklaration in der Bausteinschnittstelle, 7277
Deklaration in globalen Datenbausteinen, 7331
Deklaration in PLC-Datentypen, 7415
Format, 296
textuelle Bausteinschnittstelle, 7297

Array-Datenbaustein, 843, 845, 848, 851, 1203, 1205, 1208, 1211, 1428, 1430, 1433, 1436, 1840, 1842, 1844, 1847, 2061, 2063, 2065, 2068
ARRAY-DB, 44
ARRAY-Grenzen, 7631
ASI_CTRL, 2456, 5532
ASIN, 782, 1142, 1613, 1795, 2018, 4463, 4685, 4995, 5159, 5298
Asynchronfehlerereignis
    Anweisungen, 5625
    freigeben mit EN_AIRT, 2612, 5630
    freigeben mit EN_IRT, 2605, 5628
    sperren mit DIS_IRT, 2603, 5626
    verzögern mit DIS_AIRT, 2611, 5629
AT, 116
ATAN, 785, 1145, 1617, 1797, 2020, 4466, 4688, 4999, 5161, 5300
ATH, 2251, 5440
ATTACH, 2534
ATTR_DB, 2835
AUF, 1678, 5060
Aufbau
    Aufrufstruktur, 7972
    der Abhängigkeitsstruktur, 7978
    der Querverweisliste, 7991
    Register Speicherauslastung, 7985
Aufbau der Forcetabelle, 8117
AUFDI, 1679, 5061
Aufruf
    ABS_CTRL, 6143
    ABS_CTRL_451, 6147
    ABS_DIAG, 6145
    ABS_DIAG_451, 6149
    ABS_INIT, 6142
    CAM_CTRL, 6176
    CAM_CTRL_452, 6179
    CAM_DIAG, 6178
    CAM_DIAG_452, 6181
    CAM_INIT, 6175
    CAM_MSRM_452, 6183
    CNT_CTL1, 6115
    CNT_CTL2, 6117
    CNT_CTRL, 6112
    CNT2_CTR, 6129
    CNT2RDPN, 6129
    CNT2WRPN, 6128
    DIAG_INF, 6114
    DIAG_RD, 6130
    EncoderAbsSensorDP, 6082
    EncoderCPU314C, 6089
    EncoderET200S1Count, 6085
    EncoderET200S1SSI, 6080
    EncoderFM350, 6086
    EncoderFM450, 6077
    EncoderIM174, 6091
    EncoderIM178, 6075
    EncoderSINAMICS, 6094
    EncoderSM338, 6083
    EncoderUniversal, 6096
    High_Speed_Counter, 2883
    MC_Control, 6066
    MC_GearIn, 6072
    MC_Home, 6062
    MC_Init, 6050
    MC_MoveAbsolute, 6053
    MC_MoveJog, 6058
    MC_MoveRelative, 6055
    MC_Simulation, 6067
    MC_StopMotion, 6065
    OutputCPU314C, 6100
    OutputET200S2AO, 6100
    OutputIM174, 6105
    OutputIM178, 6098
    OutputMM4_DP, 6103
    OutputSINAMICS, 6108
    OutputSM332, 6101
    OutputSM432, 6099
    OutputUniversal, 6102
    SSI_Absolute_Encoder, 2893
Aufrufdaten, 5959
Aufrufen, 1687, 1689, 1690, 4527, 4749, 5068, 5070, 5071
Aufrufhierarchie, 63
Aufrufstruktur, 7969
    Ansichtsoptionen einstellen, 7974
    anzeigen, 7973
    Aufbau, 7972
    Bedeutung der Symbole, 7971
    Einführung, 7969
Aufrufumgebung
    für Bausteine einstellen, 8030
    für Haltepunkte einstellen, 8066
    Grundlagen, 8029
Aufträge
    ABS_CTRL, 6160
    ABS_CTRL_451, 6163
    ABS_DIAG, 6146
    ABS_DIAG_451, 6150
    CAM_CTRL, 6195
    CAM_CTRL_452, 6197
    CAM_DIAG, 6178
    CAM_DIAG_452, 6182
    CNT_CTL1, 6116
    CNT_CTRL, 6113

Auftragsabarbeitung
Reihenfolge, 6161, 6164
Auftragsheader, 6776, 6920, 6922
Auftragspuffer, 6920
Auftragsstatus, 6165
ABS_CTRL, 6163
CAM_CTRL, 6196
CAM_CTRL_452, 6198
Auftragsverwaltung, 6143, 6147
Ausdruck
arithmetischer Ausdruck, 7609
Grundlagen, 7608
Logischer Ausdruck, 7617
Vergleichsausdruck, 7612
Ausgang
einfügen, 7462, 7529
entfernen, 7464, 7531
Ausgangsparameter
High_Speed_Counter, 2890
SSI_Absolute_Encoder, 2898
Ausgangsparameter GRAPH-Funktionsbaustein, 7714
Ausgangstreiber
Aufgabe, 6097
Aufruf, 6097
Fehlerbehandlung, 6098
Initialisierung, 6097
Simulationsmodus, 6098
Auslesen
Betriebsstundenzähler mit READ_RTM, 5418
dynamisch belegte Systemressourcen mit READ_SI, 5677
LED-Zustand mit LED, 2693
Lokalzeit mit LOC_TIME, 5419
Lokalzeit mit RD_LOC_T, 2195
SZL-Teilliste mit RDSYSST, 5683
Ausschaltverzögerung, 651, 663, 682, 693, 1012, 1043, 1054, 1343, 1580, 1737, 1757, 1992, 4391, 4404, 4415, 4612, 4625, 4636, 4819, 4958, 5121, 5134, 5277
Ausschaltverzögerung erzeugen, 1992
Autovervollständigung
Anweisung einfügen, 7149
Funktion, 7147
Variable einfügen, 7147, 7149
AW, 1691, 5072, 7376
AWL, 7557, 7596
Detailvergleich, 7934
Parameterversorgung, 7561
AWL-Anweisung
aus Zwischenablage einfügen, 7594
ausschneiden, 7593
einfügen, 7577
eingeben, 7575
kopieren, 7592
löschen, 7594
Operand einfügen, 7579
selektieren, 7592
AWL-Programm
Anweisung einfügen, 7577
Anweisung eingeben, 7575
Anweisung selektieren, 7592
Bausteinaufruf einfügen, 7583, 7585
Kommentarabschnitt, 7590
Programmstatusanzeige, 8052
Regeln, 7572
Zeilenkommentar, 7590
AWL-Programme beobachten, 8052
AWL-Quelle
Bausteine generieren, 7236
AX, 7376

B

Basismodus, 8085
Basiszeit
aus Lokalzeit berechnen mit LT_BT, 5421
Baugruppe
Anfangsadresse ermitteln mit GEO_LOG, 2862, 5789
Baugruppenklasse, 5692
Baugruppenzustandsinformation über Systemzustandsliste, 5720
Datensatz asynchron lesen mit RD_DPARA, 2528, 5597
Datensatz lesen mit RD_DPAR, 2522, 5595
Diagnosedaten, 5632
Diagnoseinformationen, Systemzustandsliste, 5740
Identifikation über Systemzustandsliste, 5694
logische Adresse des Steckplatzes ermitteln mit LOG_GEO, 2864, 5791
logische Adresse ermitteln mit GADR_LGC, 2867, 5794
logische Adressen ermitteln mit RD_LGADR, 2865, 5793
parametrieren mit PARM_MOD, 5598
Steckplatz einer logischen Adresse ermitteln mit LGC_GADR, 2869, 5796
Baustein
Ablaufkette von dem Laden eines GRAP'H-DB ausschalten, 7912
Arten, 39
Auf Memory Card schreiben, 7886, 7910

aufrufen, 4527, 4749
bedingt aufrufen, 1689, 5070
Bibliothek verwenden, 7205
Codebausteine vergleichen, 7919
Eigenschaften, 7215
Eigenschaften anzeigen, 7222
Eigenschaften bearbeiten, 7222
einfügen, 7207, 7211
in externe Quelldatei exportieren, 7233
know-how-geschützte Bausteine drucken, 7258
know-how-geschützte Bausteine öffnen, 7256
Know-how-Schutz, 7239
Kommentar eingeben, 7212
Konsistenzprüfung, 7865, 7975, 7981, 7982
Konsistenzprüfung in der Aufrufstruktur, 7975
kopieren, 7207, 7210
Kopierschutz einrichten, 7250
Kopierschutz entfernen, 7250
Laden auf eine Memory Card, 7886, 7910
Laden in Gerät, 7871, 7879, 7897, 7904
Laden in Gerät im Betriebszustand RUN, 7875, 7900
Laden von einer Memory Card, 7888, 7912
Laden von Gerät, 7871, 7897
offline löschen, 7227
öffnen, 7224
online löschen, 7228
Optimierter Zugriff, 55, 58
Passwort für know-how-geschützte Bausteine ändern, 7260
schließen, 7226
Schreibschutz einrichten, 7252
Schreibschutz entfernen, 7252
speichern, 7226
suchen und öffnen, 7225
Titel eingeben, 7212
übersetzen, 7865
Übersetzen im Programmiereditor, 7869
Übersetzen in der Projektnavigation, 7868
umbenennen, 7227
Vergleich, 7917
vergleichen, 7923
verlassen, 1920, 5193
von Gerät laden, 7884, 7909
Zeitstempel, 7219
Bausteinaufruf
"Call by reference" oder "Call by value", 84
aktualisieren, 7451, 7518, 7587, 7669, 7803
ändern, 7453, 7520, 7805
Aufruf als Einzel- oder Multiinstanz, 65
Aufruftyp ändern, 7453, 7520, 7589
einfügen, 7449, 7516, 7583, 7585, 7661, 7663, 7664, 7665, 7667, 7800, 7801
Einzelinstanz, 65
GRAPH, 7701
Grundlagen, 62
Multiinstanz, 67
Parameterübergabe als Kopie oder als Pointer, 84
Schachtelungstiefe, 63
Bausteineigenschaft
anzeigen, 7222
bearbeiten, 7222
Funktion, 7214
Übersicht, 7215
Bausteinende, 1684, 1685, 1686, 5065, 5066, 5067
Bausteinkommentar
ausblenden, 7128
einblenden, 7128
Bausteinkonsistenz
in der Abhängigkeitsstruktur prüfen, 7982
prüfen, 7975
Bausteinordner, 7201
Bausteinparameter, 78, 80, 83, 87, 89, 90, 91, 92
Bausteinschnittstelle, 7272
Bausteinschnittstelle
aktualisieren, 7314
ARRAY deklarieren, 7277
Aufbau, 7272, 7290
ausblenden, 7128
Eigenschaften von Variablen, 7308, 7310
einblenden, 7128
GRAPH, 7707
Gültige Datentypen, 7266, 7268
Multiinstanz, 7282
PLC-Datentyp deklarieren, 7281
Remanenz, 7269
Rückgabewert, 7291
STRING deklarieren, 7280
STRUCT deklarieren, 7279
textuelle Bausteinschnittstelle, 7290, 7291, 7292, 7293, 7301, 7302, 7303, 7304, 7305, 7306
Variable deklarieren, 7274, 7275, 7281
Variablen importieren und exportieren, 7289
Zweck der Variablendeklaration, 7263
Bausteinvergleich
Bildlauf synchronisieren, 7947
Codebausteine vergleichen, 7919
Detailvergleich, 7925, 7928, 7930, 7934, 7937, 7940
Grundlagen, 7917
Navigation, 7946
Vergleichsergebnisse aktualisieren, 7949

Bausteinzugriff
Datenbaustein, 7338
BCD, 1634, 1636, 5016, 5018
BCD in Ganzzahl umwandeln, 1636, 5018
BCDCPL, 974, 1333, 1539, 1981, 2152, 4588, 4811, 4915, 5265, 5391
BE, 1684, 5065
BEA, 1686, 5067
BEB, 1685, 5066
BEC, 1685, 5066
Bedeutung der Spalten in der Forcetabelle, 8117
Bedingung
Einführung, 7706
Interlock programmieren, 7782
Permanente Anweisung einfügen, 7797
Supervision programmieren, 7783
Transition programmieren, 7789
Beenden
Anwenderprogramm, 911, 1271, 1495, 1931, 2101, 4499, 4721, 4850, 5197, 5326
Beenden des Forcens, 8141, 8143
Befüllen
Bereich, 814, 872, 1175, 1232, 1404, 1470, 1816, 1880, 2037, 2077, 4477, 4699, 4840, 5169, 5311
Bereich ununterbrechbar, 818, 1178, 1408, 1819, 2040
Begleitwert
Beispiel, 7842
einfügen, 7838
Einstellungen, 7840
Beispiel
Berechnen einer Gleichung, 7487, 7551, 7600
Erfassen der Richtung eines Förderbandes, 7596, 7678
Erfassen der Richtung eines Förderbands, 7482, 7545
Erfassen des Füllstands eines Lagerbereichs, 7484, 7547, 7598, 7680
für die Anzeige von Ein- und Ausgängen im Belegungsplan, 7961
für die Anzeige von Merkern im Belegungsplan, 7961
für die Eingabe von Forcewerten in die Forcetabelle, 8123
Heizen eines Ofens, 7602
Kontrollieren der Raumtemperatur, 7490, 7553
Steuern eines Förderbands, 7480, 7595, 7676
Steuerwerte in der Beobachtungstabelle, 8094
Beispiele
Steuern eines Förderbands, 7543
Belegungsplan
Ansichtsoptionen einstellen, 7964
Anzeige der remanenten Merker aktivieren, 7968
anzeigen, 7963
Aufbau, 7961
Bedeutung der Symbole, 7962
Beispiel für die Anzeige von Ein- und Ausgängen, 7961
Beispiel für die Anzeige von Merkern, 7961
Einführung, 7960
Filter definieren, 7965
Filter löschen, 7965
Filtermöglichkeiten, 7964
filtern, 7966
Beobachten
Befehl "Alle beobachten", 8104, 8131
Befehl "Sofort beobachten", 8105, 8132
Variablen im DB-Editor beobachten und steuern, 7354, 7355, 7356
Beobachten von AWL-Programmen, 8052
Beobachten von FU-Programmen, 8051
Beobachten von GRAPH-Programmen, 8054
Beobachten von KOP-Programmen, 8050
Beobachten von Schleifen, 8034
Beobachten von SCL-Programmen, 8053
Beobachten von Strukturen, 8036
Beobachten von Strukturen (S7-1200/1500), 8036
Beobachtungs- und Steuermodus, 8101
Beobachtungsmodus, 8101
Beobachtungstabelle
Aufbau, 8085
Basismodus, 8085
Bedeutung der Spalten, 8085
Bedeutung der Symbole, 8087
Beispiel für das Ausfüllen einer Beobachtungstabelle, 8091
Beobachtungs- und Steuermodus, 8101
Einsatzmöglichkeiten, 8084
erstellen, 8089
Erweiterter Modus, 8085
kopieren, 8090
öffnen, 8090
speichern, 8091
Syntaxprüfung, 8091
Testen der Verdrahtung, 8084
Übersicht der Testmöglichkeiten, 8084
Übersicht über die Anzeigeformate, 8096
Umschalten zwischen Basismodus und Erweitertem Modus, 8086
zugelassene Operanden für Steuerwerte, 8094
zugelassenen Operanden, 8092
Beobachtungswerte für Strukturen im Inspektorfenster anzeigen, 8037
Berechnen, 752, 1111, 7454, 7521

Bereich
befüllen, 814, 872, 1175, 1232, 1404, 1470, 1816, 1880, 2037, 2077, 4477, 4699, 4840, 5169, 5311
Bereich, 1226, 4692
bitweise rücksetzen, 4542, 4764, 4866, 5220, 5349
bitweisesetzen, 4537, 4758, 4861, 5216, 5344
byteweise rücksetzen, 4546, 4767, 4869, 5223, 5351
byteweise setzen, 4540, 4762, 4864, 5219, 5347
im Peripherie rücksetzen, 4544, 4765, 4867, 5221, 5350
im Peripherie setzen, 4538, 4863, 5217, 5345
kopieren, 804, 866, 1164, 1396, 1464, 1808, 1874, 2029, 2071, 4470, 4833, 5162, 5304
ununterbrechbar befüllen, 818, 1178, 1408, 1819, 2040
ununterbrechbar kopieren, 811, 870, 1172, 1230, 1402, 1467, 1814, 1877, 2035, 2074, 4474, 4696, 4836, 5166, 5307
Wert außerhalb Bereich, 735, 1095
Wert innerhalb Bereich, 733, 1094
Bereich kopieren, 807, 1167, 1398, 1810, 2031
Bereiche in SCL, 7646, 7649, 7650, 7652, 7653
Beschreibung
ABS_CTRL, 6143
ABS_CTRL_451, 6147
ABS_DIAG, 6145
ABS_DIAG_451, 6149
ABS_INIT, 6142
CAM_CTRL, 6176
CAM_CTRL_452, 6179
CAM_DIAG, 6178
CAM_DIAG_452, 6181
CAM_INIT, 6175
CAM_MSRM_452, 6183
CNT_CTL1, 6114
CNT_CTL2, 6117
CNT_CTRL, 6112
CNT2_CTR, 6129
CNT2RDPN, 6128
CNT2WRPN, 6128
DIAG_INF, 6114
DIAG_RD, 6130
EncoderAbsSensorDP, 6081
EncoderCPU314C, 6088
EncoderET200S1Count, 6084
EncoderET200S1SSI, 6079
EncoderFM350, 6086
EncoderFM450, 6077
EncoderIM174, 6090
EncoderIM178, 6075
EncoderSINAMICS, 6093
EncoderSM338, 6082
EncoderUniversal, 6096
MC_Control, 6065
MC_GearIn, 6067
MC_Home, 6059
MC_Init, 6050
MC_MoveAbsolute, 6051
MC_MoveJog, 6056
MC_MoveRelative, 6054
MC_Simulation, 6067
MC_StopMotion, 6063
OutputCPU314C, 6100
OutputET200S2AO, 6099
OutputIM174, 6104
OutputIM178, 6098
OutputMM4_DP, 6103
OutputSINAMICS, 6105
OutputSM332, 6101
OutputSM432, 6099
OutputUniversal, 6102
Betrag, 766, 1125, 1608, 2006, 4448, 4669, 4989, 5286
Betriebsart
Testen im automatischen Betrieb, 8046
Testen im halbautomatischen Betrieb, 8046
Testen im Handbetrieb, 8046
zum Testen von GRAPH-Programmen, 8042
Betriebsstundenzähler
auslesen mit READ_RTM, 5418
Eigenschaften, 5414
hantieren mit RTM, 2212, 5415
setzen mit SET_RTM, 5416
setzen, starten, stoppen und auslesen mit RTM, 5415
starten und stoppen mit CTRL_RTM, 5417
Wertebereich, 5414
Betriebssystem, 35
BEU, 1686, 5067
Bibliothek
Baustein hinzufügen und verwenden, 7205
Bibliothek für das USS-Protokoll
Allgemeine Informationen zur Antriebseinrichtung, 4157, 6560
Übersicht, 4140, 6543
USS_Drive_Control / USS_Drive_Control_31, 4149, 6552
USS_Port_Scan, 4145, 6548
USS_Port_Scan_31, 4145, 6548
USS_Read_Param / USS_Read_Param_31, 4153, 6556

USS_Write_Param /
USS_Write_Param_31, 4155, 6558
Voraussetzungen für den Einsatz, 4142, 6545
BIE, 183, 196, 1562, 4529, 4750, 4939
Bildaufbau, 1724, 5113
Binärergebnis, 183, 196
Bitfeld
im Peripheriebereich rücksetzen, 4544, 4765, 4867, 5221, 5350
im Peripheriebereich setzen, 4538, 4759, 4863, 5217, 5345
rücksetzen, 630, 991, 4542, 4764, 4866, 5220, 5349
setzen, 629, 989, 4537, 4758, 4861, 5216, 5344
Bitfolge, 249, 250, 251
64 Bit, 252
Bitmaske, 964, 1325, 1526, 1972, 2140, 4570, 4792, 4894, 5248, 5375
Bitmuster für 7-Segment-Anzeige erzeugen, 1538, 2151
Bits
zählen, 975, 1334, 1540, 1982, 2153, 4590, 4813, 4916, 5266, 5392
BITSUM, 975, 1334, 1540, 1982, 2153, 4590, 4813, 4916, 5266, 5392
Bitverknüpfung
Eingang einfügen, 983, 4595
EXKLUSIV ODER, 982, 1547, 1548, 1554, 1555, 4594, 4923, 4924, 4930, 4931
ODER, 979, 981, 1544, 1546, 1549, 1552, 1553, 4592, 4593, 4920, 4921, 4925, 4928, 4929
UND, 976, 978, 1542, 1543, 1550, 1551, 4591, 4592, 4917, 4919, 4926, 4927
BLD, 1724, 5113
BLKMOV, 866, 1226, 1464, 1874, 2071, 4470, 4692, 4833, 5162, 5304
BLOCK, 363
BLOCK_DB_TO_WORD, 614
BOOL, 248, 372, 423, 510, 530, 581, 592
BOOL_TO_, 423, 530, 592
BRCV, 3617
BSEND, 3615
BTD, 1636, 5018
BTI, 1634, 5016
BUSY bei asynchron arbeitenden Anweisungen, 4370
BY, 1912
BYTE, 249, 373, 425, 511, 531, 582, 593
BYTE_TO_, 425, 531, 593
Bytes
Anordnung ändern, 841, 1201, 1839, 2058, 5301

C

CAD, 1646, 5028
CALC, 752, 1111
CALCULATE, 7454, 7521
CALL, 1687, 4527, 4749, 5068
CAM_CTRL, 6176
Arbeitsweise, 6176
Aufruf, 6176
Aufträge, 6195
Auftragsstatus, 6196
Beschreibung, 6176
Fehlerverhalten, 6177
Funktionsschalter, 6196
Rückgabewerte, 6177
Verwendete Datenbausteine, 6177
CAM_CTRL_452, 6179
Arbeitsweise, 6179
Aufruf, 6179
Aufträge, 6197
Auftragsstatus, 6198
Beschreibung, 6179
Fehlerverhalten, 6181
Funktionsschalter, 6197
Parameter, 6180
Rückgabewerte, 6180
Verwendete Datenbausteine, 6180
CAM_DIAG
Anlauf, 6178
Arbeitsweise, 6178
Aufruf, 6178
Aufträge, 6178
Beschreibung, 6178
Fehlerverhalten, 6179
Rückgabewerte, 6178
Verwendete Datenbausteine, 6178
CAM_DIAG_452
Anlauf, 6182
Arbeitsweise, 6181
Aufruf, 6181
Aufträge, 6182
Beschreibung, 6181
Fehlerverhalten, 6182
Parameter, 6182
Rückgabewerte, 6182
Verwendete Datenbausteine, 6181
CAM_INIT
Aufruf, 6175
Beschreibung, 6175
Parameter, 6176
Rückgabewerte, 6176

CAM_MSRM_452, 6183
Anlauf, 6184
Aufruf, 6183
Beschreibung, 6183
Fehlerverhalten, 6184
Parameter, 6183
Rückgabewerte, 6183
Verwendete Datenbausteine, 6183
CAN_DINT, 2572
CAN_TINT, 2557
CAR, 1629, 5011
CASE, 1910, 5183
CAW, 1645, 5027
CD, 1588, 4967
CDB, 1680, 5061
CEIL, 880, 1240, 1886, 2082, 4483, 4705, 5174, 5316
CH_DIAG
Anweisung, 6230
Ausgangsparameter, 6231
Blockschaltbild, 6233
Eingangsparameter, 6231
CH_DIAG_455
Anweisung, 6311
Ausgangsparameter, 6312
Blockschaltbild, 6314
Eingangsparameter, 6312
CHAR, 276, 416, 497, 526, 527, 573, 574, 591, 612
CHAR_TO_, 497, 573, 574, 612
Chars_TO_Strg, 2234
CHECKBACK_SIGNALS
CNT2_CTR, 6138
CiR, 4503, 4726, 4855, 5201, 5330
CiR-Vorgang
steuern, 4726, 4855, 5201
CJ_T_PAR
Anweisung, 6240
Ausgangsparameter, 6241
Eingangsparameter, 6241
CJ_T_PAR_455
Anweisung, 6320
Ausgangsparameter, 6321
Eingangsparameter, 6321
CLR, 1561, 4938
CMP <, 731, 1091, 4439, 4660
CMP <=, 725, 1086, 4437, 4657
CMP <>, 718, 1079, 4434, 4655
CMP ==, 712, 1074, 4433, 4654
CMP >, 728, 1089, 4438, 4659, 5396, 5397, 5401, 5402
CMP >=, 723, 1084, 4435, 4656
CMP>T, 2159, 2171
CMP>T_MAX, 2162, 2174
CMP>T_WARN, 2164, 2176
CMP>U, 2161, 2172
CNT_CTL1, 6114
Anlaufverhalten, 6116
Arbeitsweise, 6115
Aufruf, 6115
Aufträge, 6116
Beschreibung, 6114
Fehlerverhalten, 6116
Parameter, 6115
Wertübergabe, 6123
CNT_CTL2, 6117
Arbeitsweise, 6117
Aufruf, 6117
Beschreibung, 6117
CNT_CTRL, 6112
Anlaufverhalten, 6113
Arbeitsweise, 6112
Aufruf, 6112
Aufträge, 6113
Beschreibung, 6112
Fehlerverhalten, 6114
Parameter, 6113
CNT2_CTR, 6129
Arbeitsweise, 6129
Aufruf, 6129
Beschreibung, 6129
CHECKBACK_SIGNALS, 6138
CONTROL_SIGNALS, 6137
Parameter, 6130
CNT2RDPN, 6128
Arbeitsweise, 6129
Aufruf, 6129
Beschreibung, 6128
Leseauftrag, 6141
Parameter, 6129
Status des Leseauftrags, 6141
CNT2WRPN, 6128
Arbeitsweise, 6128
Aufruf, 6128
Beschreibung, 6128
Parameter, 6128
Schreibauftrag, 6138
Status des Schreibauftrags, 6140
Codeabschnitte auf- und zuklappen, 7637
COMPRESS, 4497, 4719, 4848, 5195, 5324
CONCAT, 2258
CONF_DB See Konfigurations-Datenbaustein, 6881
CONT_C
Arbeitsweise, 5801
Ausgangsparameter, 5805

Blockschaltbild, 5803
Eingangsparameter, 5804
CONT_S
Anweisung, 5806
Arbeitsweise, 5807
Ausgangsparameter, 5810
Blockschaltbild, 5808
Eingangsparameter, 5809
CONTINUE, 1918, 5190
CONTROL_SIGNALS
CNT2_CTR, 6137
CONVERT, 876, 1236, 1884, 2080, 4480, 4702, 5172, 5314
COS, 780, 1139, 1614, 1793, 2016, 4461, 4682, 4996, 5157, 5296
Cosinus, 780, 1139, 1614, 1793, 2016, 4461, 4682, 4996, 5157, 5296
COUNT, 6365
COUNTER, 363
CountOfElements, 858, 1217, 1460, 1870
CP-Adresse, 6917
CPU
Anlauf, 6144, 6148, 6177, 6180
Datum und Uhrzeit lesen mit RD_SYS_T, 2193, 5409
identifizieren mit Systemzustandsliste, 5700
Merkmale lesen mit Systemzustandsliste, 5695
Uhrzeit stellen mit WR_SYS_T, 2190, 5408
Zusammenhang Protokollvariante, Datenlänge, 6453
CPU-Datenbaustein
Definition, 54
löschen, 7229
CREATE_DB, 2821
CRP_IN
Anweisung, 5963
Ausgangsparameter, 5964
Eingangsparameter, 5964
CRP_OUT
Anweisung, 5964
Ausgangsparameter, 5965
Eingangsparameter, 5965
CTD, 697, 1058, 1354, 1766, 1999, 4419, 4640, 4823, 5138, 5281
CTRL_HSC, 2877
CTRL_HSC_EXT, 2880
CTRL_PTO, 2755
CTRL_PWM, 2752
CTU, 695, 1056, 1352, 1764, 1997, 4417, 4638, 4821, 5136, 5279
CTUD, 700, 1061, 1357, 1769, 2002, 4421, 4642, 4825, 5140, 5283
CU, 1587, 4966

D

D_ACT_DP, 2406
Data Log
erstellen mit DataLogCreate, 2779
erstellen mit DataLogNewFile, 2799
erstellen mit DataLogTypedNewFile, 2801
leeren mit DataLogClear, 2791
löschen mit DataLogDelete, 2797
öffnen mit DataLogOpen, 2787
öffnen mit DataLogTypedOpen, 2789
schließen mit DataLogClose, 2795
schreiben mit DataLogWrite, 2793
DataLogClear, 2791
DataLogClose, 2795
DataLogCreate, 2779
DataLogDelete, 2797
DataLogNewFile, 2799
DataLogOpen, 2787
DataLogTypedNewFile, 2801
DataLogTypedOpen, 2789
DataLogWrite, 2793
DATE, 271, 410, 490, 523, 569, 589, 609
DATE_AND_LTIME, 274
DATE_AND_TIME, 272
DATE_TO_, 490, 569, 609
Daten empfangen
und ablegen mit RCV_PTP, 7049
und bereitstellen mit SERVE_RK, 7062
von außerhalb der eigenen S7-Station mit X_RCV, 7078
Daten lesen
aus DP-Normslaves/PROFINET IO-Device mit DPRD_DAT, 2434, 5496
aus remoter CPU mit GET, 3594
außerhalb der eigenen S7-Station mit X_GET, 7083
innerhalb der eigenen S7-Station mit I_GET, 7066
Daten schreiben
außerhalb der eigenen S7-Station mit X_PUT, 7085
in DP-Normslaves/PROFINET IO-Device mit DPWR_DAT, 2436, 5499
in remote CPU mit PUT, 3600, 6395
innerhalb der eigenen S7-Station mit I_PUT, 7068

Daten senden
außerhalb der eigenen S7-Station mit X_SEND, 7077
mit TSEND, 6468
Datenaustausch, 7557
Datenbaustein
aktualisieren, 7326
anlegen, 7204, 7324
anlegen mit CREAT_DB, 5759
ARRAY deklarieren, 7331
ARRAY-Datenbaustein, 42, 55, 111, 114, 7204, 7319, 7324, 7365, 7889
Attribute lesen mit ATTR_DB, 2835
aus Ladespeicher lesen mit READ_DBL, 2825, 5753
basierend auf einem PLC-Datentyp, 7335
Bausteinänderungen ohne Reinitialisierung laden, 7328
CONF_DB, 6484
CPU-Datenbaustein, 54
Datenwerte während der Inbetriebnahme anpassen, 7363, 7365, 7366, 7367, 7368
Defaultwert, 7336
Deklarationstabelle, 7322
Eigenschaften von Variablen, 7341, 7344, 7345
Einstellwerte nutzen, 7363, 7365, 7366, 7367, 7368
erzeugen mit CREA_DB, 5760
erzeugen mit CREA_DBL, 5762
erzeugen mit CREATE_DB, 2821
Globaler Datenbaustein, 42, 7319
Instanz-Datenbaustein, 53, 7319
löschen mit DEL_DB, 5757
löschen mit DELETE_DB, 2839
Momentaufnahme, 7357, 7359, 7361
öffnen, 1678, 1679, 4495, 4496, 4717, 4718, 5061
Optimierter Zugriff, 55, 58
programmieren, 7319, 7330
Regel-DB WS_RULES, 5435
Remanenzverhalten, 7338, 7339
schreiben in Ladespeicher mit WRIT_DBL, 2829, 5755
Startwert, 7336, 7337
STRUCT deklarieren, 7332
testen mit TEST_DB, 5765
Variable importieren und exportieren, 7350
Variablen beobachten und steuern, 7354, 7355, 7356, 7357, 7359, 7361
Datenbaustein-Bereich
holen mit FETCH_RK, 7057
senden mit SEND_PTP, 7047
senden mit SEND_RK, 7053
Datenbausteinregister, 1680, 5061
Datenkonsistenz, 3589, 6383
Datensatz
aus projektierten Systemdaten lesen mit RD_DPARM, 2530, 5601
bereitstellen auf I-Device mit PRVREC, 2440, 5504
Datensätze schreiben und lesen, 2521, 5593
Dynamische Parameter übertragen mit WR_PARM, 5602
einer Baugruppe asynchron lesen mit RD_DPARA, 2528, 5597
einer Baugruppe lesen mit RD_DPAR, 2522, 5595
empfangen auf I-Device mit RCVREC, 2438, 5502
lesen mit RD_REC, 2428, 5490
lesen mit RDREC, 2306, 5456
schreiben mit WR_REC, 2433, 5494
übertragen mit WRREC, 2326, 5458
vordefinierte Parameter schreiben mit WR_DPARM, 2532, 5603
Datentyp
ANY, 359
ARRAY, 296
BOOL, 248, 372, 423, 510, 530, 581, 592
BYTE, 249, 373, 425, 511, 531, 582, 593
CHAR, 276, 416, 497, 526, 527, 573, 574, 591, 612
DATE, 271, 410, 490, 523, 569, 589, 609
DINT, 258, 386, 453, 518, 553, 586, 603
DT, 272, 404, 482, 590, 611
DTL, 274, 408, 488, 525, 571
DWORD, 251, 376, 431, 513, 536, 584, 597
explizit konvertieren, 876, 1236, 1884, 2080, 5172
Explizite Konvertierung, 423, 425, 428, 431, 435, 439, 443, 446, 450, 453, 457, 461, 465, 468, 471, 474, 476, 479, 482, 485, 488, 490, 492, 494, 497, 499, 501, 504, 530, 531, 533, 536, 540, 543, 546, 550, 553, 557, 560, 563, 566, 569, 570, 571, 573, 574, 575, 577, 592, 593, 595, 597, 600, 603, 606, 607, 608, 609, 610, 611, 612, 613
GRAPH-Anweisung, 7776
Gültigkeit, 243

Implizite Konvertierung, 372, 373, 374, 378, 380, 381, 383, 384, 386, 388, 390, 392, 394, 396, 398, 400, 402, 404, 406, 408, 410, 412, 414, 416, 418, 419, 421, 510, 511, 512, 514, 515, 516, 517, 518, 519, 520, 521, 525, 526, 527, 528, 529, 581, 582, 583, 585, 586, 588, 590, 591
INT, 256, 383, 446, 516, 546, 585, 600
Konvertierung, 369, 376, 507, 513, 522, 523, 524, 578, 584, 587, 589, 590
LDT, 274, 406, 485
LINT, 260, 390, 461
LREAL, 265, 396, 471, 521, 563
LTIME, 270, 402, 479
LTIME_OF_DAY, 272
LTOD, 414, 494
LWORD, 252, 378, 435
PLC-Datentyp, 286
POINTER, 356
REAL, 264, 394, 468, 520, 560, 586, 606
S5TIME, 268, 398, 474, 588, 608
SCL-Anweisung, 7654, 7655, 7656
SINT, 254, 380, 439, 514, 540
STRING, 277, 419, 501, 528, 529, 575, 577, 591, 613
STRUCT, 291
TIME, 269, 400, 476, 522, 566, 587, 607
TIME_OF_DAY, 271
TOD, 412, 492, 524, 570, 590, 610
UDINT, 259, 388, 457, 519, 557
UINT, 257, 384, 450, 517, 550
ULINT, 261, 392, 465
USINT, 255, 381, 443, 515, 543
VARIANT, 339
WCHAR, 276, 418, 499
WORD, 250, 374, 428, 512, 533, 583, 595
WSTRING, 280, 421, 504
Datentypen, 207
Datenübertragung, anstoßen, 4113, 6516
Datum, 271, 272, 274
DB, 7204, 7324
DB 101 bis 104
Ausgangsparameter, 6246
Durchgangsparameter, 6248
Eingangsparameter, 6243
DB 101 bis 116
Ausgangsparameter, 6343
Durchgangsparameter, 6345
Eingangsparameter, 6340
DB_ANY_TO_VARIANT, 1480, 1901
DB-Variable
Definition, 100
DCAT, 957, 1318, 1519, 1966, 2133, 4563, 4785, 4888, 5241, 5369
DEAD_T
Anweisung, 5966
Ausgangsparameter, 5967
Eingangsparameter, 5967
Globaler Datenbaustein, 5968
DEADBAND
Anweisung, 5969
Ausgangsparameter, 5970
Eingangsparameter, 5970
DEC, 764, 1124, 1603, 4983
DECO, 932, 1292, 1508, 1943, 2112, 4511, 4734, 4858, 5203, 5334
Decodieren, 932, 1292, 1508, 1943, 2112, 4511, 4734, 4858, 5203, 5334
Definieren
Filter für den Belegungsplan, 7965
Dekrementieren, 764, 1124, 1603, 4983
DELETE, 2265
DELETE_DB, 2839
Demultiplexen, 939, 1300, 1949, 2118
DEMUX, 939, 1300, 1949, 2118
Deserialisieren, 795, 1156, 1386, 1799, 2022
Deserialize, 795, 1156, 1386, 1799, 2022
Deskalieren, 892, 1251, 1485, 1906, 2092, 4489, 4711, 4845, 5179, 5321
DETACH, 2538
Detailvergleich
Darstellung des Vergleichsergebnisses für AWL, 7934
Darstellung des Vergleichsergebnisses für GRAPH, 7940
Darstellung des Vergleichsergebnisses für KOP/FUP, 7930
Darstellung des Vergleichsergebnisses für SCL, 7937
starten, 7925, 7928
DeviceStates, 2726
DIAG_INF, 6114
Arbeitsweise, 6114
Aufruf, 6114
Beschreibung, 6114
Parameter, 6114
DIAG_RD, 6130
Arbeitsweise, 6130
Aufruf, 6130
Beschreibung, 6130
Parameter, 6130
Diagnose
Baugruppe, Systemzustandsliste, 5740
Baugruppendiagnosedaten, 5743

Diagnosedaten, 5632
Diagnosedaten, kanalspezifisch, 5634
Diagnosedatensatz, 5742
Diagnoseereignis in Diagnosepuffer schreiben mit WR_USMSG, 5667
DP-Slave, Diagnosedaten lesen mit DPNRM_DG, 2450, 5522
Verbindungen mit C_DIAG, 5748
Diagnosedaten
ABS_DIAG, 6146
ABS_DIAG_451, 6150
auslesen mit GET_DIAG, 2740
Diagnose-DB
Aufbau, 6194
Aufgabe, 6194
Inhalt, 6194
Diagnose-DB FM x51
Aufbau, 6159
Aufgabe, 6159
Inhalt, 6159
Diagnoseinformation generieren, 2737
Diagnosepuffer
Systemzustandsliste, 5739
DIF
Ausgangsparameter, 5973
Eingangsparameter, 5972
Differenzierer, 5971
DIGITAL, 6356
Digitalausgang
TIO_DQ, 3583
Digitaleingang
TIO_DI, 3575
DINT, 258, 386, 453, 518, 553, 586, 603
DINT_TO_, 453, 553, 603
DIS_AIRT, 2611
DIS_IRT, 2603
DIV, 759, 1119, 4444, 4665
Dividieren, 759, 1119, 1596, 1600, 1607, 4444, 4665, 4975, 4980, 4988
Divisionsrest, 761, 1120, 1604, 4446, 4666, 4984
DMSK_FLT, 2588
DO, 1912, 1915, 5188
Doppelwort, 251
DP_CTRL, 6952
DP_DIAG, 6944
DP_RECV, 6938
DP_SEND, 6935
DP_TOPOL, 2453
DP-Mastersystem
Informationen, Systemzustandsliste, 5719
Topologie ermitteln mit DP_TOPOL, 2453, 5526
DP-Normslaves
alle Ausgänge schreiben mit SETIO, 2347, 5461
alle Eingänge lesen mit GETIO, 2346, 5460
einen Teil der Ausgänge schreiben mit SETIO_PART, 2357, 5464
einen Teil der Eingänge lesen mit GETIO_PART, 2355, 5462
konsistente Daten lesen mit DPRD_DAT, 2434, 5496
konsistente Daten schreiben mit DPWR_DAT, 2436, 5499
DPNRM_DG, 2450
DPRD_DAT, 2434
DP-Slave
deaktivieren und aktivieren mit D_ACT_DP, 2406, 5484
Diagnosedaten lesen mit DPNRM_DG, 2450, 5522
Gruppen synchronisieren mit DP_SYC_FR, 2444, 5516
DPSYC_FR, 2444
DP-Verschaltungen aktualisieren mit PN_DP, 7076
DPWR_DAT, 2436
DRUM, 951, 1311, 1512, 1959, 2127, 4549, 4770, 4872, 5226, 5354
DRUM_X, 4555, 4776, 4879, 5233, 5361
DT, 272, 404, 482, 590, 611
DT_TO_, 482, 611
DTB, 1638, 5020
DTL, 274, 408, 488, 525, 571
DTL_TO_, 488, 571
DTR, 1639, 5021
Durchgängige Symbolik, 205
DWORD, 251, 376, 431, 513, 536, 584, 597
DWORD_TO_, 431, 536, 597
Dynamisch belegte Systemressourcen
freigeben mit DEL_SI, 5679

E

EA-Adressse, 2849
EB, 7376
ED, 7376
Ein- und Ausrücken
Zeilen, 7635
Einerkomplement, 930, 1291, 1640, 1641, 4510, 4733, 5022, 5023
Eingang
einfügen, 983, 4595, 7462
entfernen, 7464, 7531
Eingang einfügen, 7529

Eingangsparameter
High_Speed_Counter, 2888
LP_SCHED_S, 6048
SSI_Absolute_Encoder, 2896
Eingangsparameter GRAPH-Funktionsbaustein, 7710
EingangsparameterPID_ES, 5935
Eingangstreiber
Aufgabe, 6073
Aufruf, 6074
Fehlerbehandlung, 6074
Initialisierung, 6074
Simulationsmodus, 6075
Einschaltverzögerung, 648, 660, 677, 689, 1009, 1038, 1050, 1340, 1576, 1734, 1752, 1990, 4389, 4399, 4411, 4610, 4620, 4632, 4816, 4954, 5118, 5129, 5274
speichernd, 680, 691, 1041, 1052, 1578, 1754, 4402, 4413, 4623, 4634, 4956, 5131
Einschaltverzögerung erzeugen, 1990
Einstellen
Ansichtsoptionen für den Belegungsplan, 7964
Ansichtsoptionen für die Abhängigkeitsstruktur, 7980
Einstellungen
allgemein, 7150
ändern, 7153, 7428, 7496, 7563, 7632, 7731
AWL, 7563
FUP, 7428, 7496
GRAPH, 7729
KOP, 7428, 7496
SCL, 7631
Einstellungen in der Querverweisliste, 7994
Einzelinstanz
Aufruftyp ändern, 7453, 7520, 7589
Beispiel, 65
Definition, 65
Einzelschrittansicht, 7744
Einzelschrittmodus
Aufruf ausführen, 8077
Aufrufumgebung für Haltepunkte einstellen, 8066
Grundlagen, 8058, 8079, 8081, 8083
Haltepunkte aktivieren, 8069, 8072, 8078
Haltepunkte deaktivieren, 8069, 8072, 8078
Haltepunkte löschen, 8075
Haltepunkte setzen, 8067
Nächste Anweisung ausführen, 8077
Programmausführung bis zur Markierung fortsetzen, 8077
Programmausführung fortsetzen, 8077
zu den Haltepunkten navigieren, 8073
ELSE, 1908, 1910, 5181, 5183
ELSIF, 1908, 5181
E-Mail
E-Mail übertragen mit TM_MAIL, 4362
E-Mail übertragen mit TMAIL_C, 3662
E-MAIL-Verbindung
Parameterblock für, 6890
Empfangen von Daten
auf I-Device mit RCVREC, 2438, 5502
Empfangspuffer löschen mit RES_RCVB, 7052
EN-/ENO-Mechanismus
AWL-Beispiel, 162
FUP-Beispiel, 159
Grundlagen, 150
KOP-Beispiel, 154
SCL-Beispiel, 167
EN_AIRT, 2612
EN_IRT, 2605
ENCO, 933, 1293, 1509, 1944, 2113, 4512, 4735, 4859, 5204, 5335
EncoderAbsSensorDP, 6081
Aufruf, 6082
Beschreibung, 6081
Fehlerbehandlung, 6082
Parameter, 6082
Synchronisation, 6082
EncoderCPU314C, 6088
Aufruf, 6089
Beschreibung, 6088
Fehlerbehandlung, 6090
Parameter, 6089
Synchronisation, 6089
EncoderET200S1Count, 6084
Aufruf, 6085
Beschreibung, 6084
Fehlerbehandlung, 6086
Parameter, 6085
Synchronisation, 6085
EncoderET200S1SSI, 6079
Aufruf, 6080
Beschreibung, 6079
Fehlerbehandlung, 6081
Parameter, 6080
Synchronisation, 6080
EncoderFM350, 6086
Aufruf, 6086
Beschreibung, 6086
Fehlerbehandlung, 6088
Parameter, 6087
Synchronisation, 6087
EncoderFM450, 6077
Aufruf, 6077
Beschreibung, 6077

Fehlerbehandlung, 6079
Parameter, 6078
Synchronisation, 6078
EncoderIM174, 6090
Aufruf, 6091
Beschreibung, 6090
Fehlerbehandlung, 6093
Fehlerquittierung, 6093
Parameter, 6092
Parametrierung der IM 174 (Hardware-Konfiguration), 6090
Synchronisation, 6092
EncoderIM178, 6075
Aufruf, 6075
Beschreibung, 6075
Fehlerbehandlung, 6077
Fehlerquittierung, 6077
Parameter, 6076
Parametrierung der IM 178-4 (Hardware-Konfiguration), 6075
Synchronisation, 6076
EncoderSINAMICS, 6093
Aufruf, 6094
Beschreibung, 6093
Fehlerbehandlung, 6096
Fehlerquittierung, 6096
Parameter, 6094
Parametrierung des SINAMICS (Hardware-Konfiguration), 6093
Synchronisation, 6095
EncoderSM338, 6082
Aufruf, 6083
Beschreibung, 6082
Fehlerbehandlung, 6084
Parameter, 6083
Synchronisation, 6083
EncoderUniversal, 6096
Arbeitsweise, 6096
Aufruf, 6096
Beschreibung, 6096
Fehlerbehandlung, 6097
Parameter, 6096
Synchronisation, 6097
Encodieren, 933, 1293, 1509, 1944, 2113, 4512, 4735, 4859, 5204, 5335
END_CASE, 1910, 5183
END_FOR, 1912
END_IF, 1908, 5181
END_REPEAT, 1916, 5189
END_WHILE, 1915, 5188
Endian, 1449, 1451, 1453, 1455, 1859, 1861, 1863, 1865
ENDIS_PW, 904, 1264, 1488, 1922, 2095
ENO automatisch sezten, 7631
ENT, 5101
EQ_ElemType, 742, 1102, 1369
EQ_Type, 738, 1098, 1361, 1365
Ereignis
Organisationsbaustein (OB) zuordnen mit ATTACH, 2534
Zuordnung zu Organisationsbaustein (OB) aufheben mit DETACH, 2538
ERR_MON
Anweisung, 5973
Ausgangsparameter, 5975
Eingangsparameter, 5975
ErrorID, 2893, 2899
Ersatzteilfall, 6772, 6918
Ersatzwert, 4548, 4769, 4871, 5225, 5353
Erstabfrage, 183
Erstellen einer Beobachtungstabelle, 8089
Erweiterte Statusinformation, 7631
Erweiterter Modus, 8085
EW, 7376
EX, 7376
EXIT, 1918, 5191
EXKLUSIV ODER, 981, 982, 4593, 4594
Bitverknüpfung, 981, 4593
EXP, 777, 1137, 1611, 1791, 2014, 4458, 4680, 4992, 5156, 5294
Exponentialwert, 777, 1137, 1611, 2014, 4458, 4680, 4992, 5294
Exponentialwert bilden, 1791, 5156
EXPT, 788, 1148
Externe Quelldatei
Baustein exportieren, 7233
bearbeiten, 7236
Dateitypen mit Editor verknüpfen, 7236
einbinden, 7235
Grundlagen, 7230
öffnen, 7236
Regeln zur Programmierung, 7232

F

F_TRIG, 644, 1005, 1337, 1727
FALSE, 643, 644, 1003, 1005, 1336, 1337, 1726, 1727
Favoriten
ausblenden, 7128
einblenden, 7128
entfernen, 7448, 7515, 7583, 7659, 7751
hinzufügen, 7446, 7513, 7581, 7657, 7749
verwenden, 7447, 7514, 7582, 7658, 7750

FB, 41
FC, 40
Fehlerauswertung mit RET_VAL, 171
Fehlerbehandlung
GetError, 912, 1272, 1495, 1931, 2101
GetErrorID, 916, 1276, 1499, 1935, 2105
Fehlercodes
TIO_DI, 3577
TIO_DQ, 3586
TIO_IOLink_IN, 3573
TIO_IOLink_OUT, 3581
TIO_SYNC, 3569
Fehlerinformation, 171, 912, 916, 1272, 1276, 1495, 1499, 1931, 1935, 2101, 2105
Fehlerverhalten
ABS_CTRL, 6145
ABS_CTRL_451, 6149
ABS_DIAG, 6146
ABS_DIAG_451, 6150
CAM_CTRL, 6177
CAM_CTRL_452, 6181
CAM_DIAG, 6179
CAM_DIAG_452, 6182
CAM_MSRM_452, 6184
CNT_CTL1, 6116
CNT_CTRL, 6114
High_Speed_Counter, 2887
SSI_Absolute_Encoder, 2896
Feld, 296
siehe ARRAY, 7277, 7331
Feld lesen, 862, 1222
Feld schreiben, 864, 1224
FETCH_RK, 7057
FieldRead, 862, 1222
FieldWrite, 864, 1224
FIFO, 351
FIFOQueue, 351
FILL, 872, 1232, 1470, 1880, 2077, 4477, 4699, 4840, 5169, 5311
FILL_BLK, 814, 1175, 1404, 1816, 2037
Filter
auswählen, 7966
für den Belegungsplan definieren, 7965
im Belegungsplan, 7964
löschen, 7965
Filtern
Belegungsplan, 7966
FIND, 2271
Flanke
negativ, 636, 639, 641, 997, 1000, 1002, 1563, 1727, 4383, 4386, 4604, 4607, 4940
positiv, 634, 638, 640, 995, 998, 1001, 1564, 1726, 4382, 4385, 4602, 4605, 4942
Flipflop
rücksetzen/setzen, 633, 994, 4380, 4601
setzen/rücksetzen, 631, 992, 4379, 4600
FLOOR, 881, 1241, 1888, 2083, 4484, 4706, 5175, 5317
FM 351
Anlauf, 6144
FM 352
Anlauf, 6177, 6180
FM 451
Anlauf, 6148
FM 452
Anlauf, 6177, 6180
FMT_CJ_T
Anweisung, 6287
Ausgangsparameter, 6288
Durchgangsparameter, 6288
Eingangsparameter, 6287
FMT_DS1
Anweisung, 6274
Ausgangsparameter, 6274
Durchgangsparameter, 6275
Eingangsparameter, 6274
FMT_PAR
Anweisung, 6269
Ausgangsparameter, 6270
Durchgangsparameter, 6271
Eingangsparameter, 6270
FMT_PID
Anweisung, 6249
Arbeitsweise, 6250
Ausgangsparameter, 6253
Blockschaltbild, 6261
Durchgangsparameter, 6254
Eingangsparameter, 6253
Statische Variablen, 6254, 6256, 6258
FMT_PV
Anweisung, 6282
Ausgangsparameter, 6286
Durchgangsparameter, 6286
Eingangsparameter, 6285
FMT_TUN
Anweisung, 6277
Ausgangsparameter, 6278
Durchgangsparameter, 6279
Eingangsparameter, 6278
Statische Variablen, 6279

FN, 1563, 4940
FOR, 1912, 5185
FORCE355
Anweisung, 6226
Ausgangsparameter, 6228
Eingangsparameter, 6227
FORCE455
Anweisung, 6307
Ausgangsparameter, 6309
Eingangsparameter, 6309
Forcen
Alle forcen, 8140
Forcen beenden, 8141, 8143
Forcen von Variablen
Sicherheitsvorkehrungen, 8116, 8134
Forcetabelle
Anzeige, 8120
Aufbau, 8117
Basismodus, 8118
Bedeutung der Spalten, 8117
Bedeutung der Symbole, 8118
Beobachtungs- und Steuermodus, 8101
Erweiterter Modus, 8118
Funktionalität, 8115
öffnen, 8120
speichern, 8120
Syntaxprüfung, 8121
Testmöglichkeiten, 8115
Übersicht der Testmöglichkeiten, 8115
Übersicht über die Anzeigeformate, 8124
Umschalten zwischen Basismodus und Erweitertem Modus, 8118
Variablen sofort beobachten, 8132
zugelassene Operanden, 8122
zugelassene Operanden für Forcewerte, 8123
Forcewert
zugelassene Operanden, 8123
FP, 1564, 4942
FR, 1566, 1582, 4943, 4960
FRAC, 787, 1147, 1798, 2021
Freier Kommentar
bearbeiten, 7456, 7523
einfügen, 7456, 7523
Einführung, 7455, 7522
löschen, 7458, 7525
Freigeben
Meldungen mit EN_MSG, 5673
FREQUENC, 6369
Frequenzmessung mit FREQUENC, 6369
FTP, 4013, 6964
FTP_CMD, 4013, 6893, 6964
FTPS, 4013, 6964
FTP-Verbindung
Parameterblock für, 6891
Funktion (FC)
anlegen, 7203
Definition, 40
in externe Quelldatei exportieren, 7233
Funktionalität der Forcetabelle, 8115
Funktionsbaustein (FB)
anlegen, 7203
Definition, 41
in externe Quelldatei exportieren, 7233
Instanz-Datenbaustein, 41
Funktionsplan, 7493
Funktionsschalter
ABS_CTRL, 6161
ABS_CTRL_451, 6163
CAM_CTRL, 6196
CAM_CTRL_452, 6197
FUP, 7493, 7543, 7545, 7547
Detailvergleich, 7930
FUP-Element
aus Zwischenablage einfügen, 7528
ausschneiden, 7527
einfügen, 7508, 7509
ersetzen, 7529
kopieren, 7526
löschen, 7532
Operand einfügen, 7533
Regeln beim Einfügen, 7507
selektieren, 7526
FUP-Netzwerk
Bausteinaufruf einfügen, 7449, 7516
Programmstatusanzeige, 8051
Regeln zu Zweigen, 7537
Zweig, 7537
Zweig einfügen, 7538
Zweig löschen, 7538
FUP-Programme beobachten, 8051
FUZ_355
Anweisung, 6224
Ausgangsparameter, 6225
Durchgangsparameter, 6226
Eingangsparameter, 6225
FUZ_455
Anweisung, 6305
Ausgangsparameter, 6306
Durchgangsparameter, 6307
Eingangsparameter, 6306

G

GADR_LGC, 2867

Ganzzahl, 883, 1243, 1648, 1889, 2084, 4485, 4707, 5030, 5176, 5318
16 Bit, 256, 257
32 Bit, 258, 259
64 Bit, 260, 261
8 Bit, 254, 255
Ganzzahlen, 188
Gemeinsame PtP-Parameterfehler, 4100, 6504
GEN_DIAG, 2737
Gen_UsrMsg, 2637
Genauigkeit, 7623
GEO_LOG, 2862
GEO2LOG, 2850
Gerätestatus
eines remoten Partners abfragen mit STATUS, 6435
Wechsel empfangen mit USTATUS, 6437
Gesteuertes Positionieren
analog mit ANALOG, 6346
digital mit DIGITAL, 6356
GET, 3594
Get_AlarmState, 2632
GET_DIAG, 2740
Get_IM_Data, 2696
GET_NAME, 2702
GetBlockName, 2287
GetChecksum, 2713
GetError, 912, 1272, 1495, 1931, 2101
GetErrorID, 916, 1276, 1499, 1935, 2105
GetInstanceName, 2279
GetInstancePath, 2282
GETIO, 2346
GETIO_PART, 2355
GetSMCinfo, 2716
GetStationInfo, 2707
GetSymbolName, 2272
GetSymbolPath, 2276
Gleich, 745, 1105, 1372
GLEICH ARRAY, 748, 1108, 1375, 1779
Gleitpunktzahl, 264, 265
Gültigkeit prüfen, 736, 1096
Ungültigkeit prüfen, 737, 1097
Gleitpunktzahlen, 189, 266
ungültige, 266
Globale Bibliothek
Überblick über das USS-Protokoll, 4140, 6543
Globale Variable, (Siehe PLC-Variable)
Globaler Datenbaustein, 42
Remanenzverhalten, 7339
GOTO, 1919, 5192
GRAPH, 169
Ablaufkette initialisieren, 8045
Ablaufketten, 7687, 7688, 7752, 7753, 7754, 7755, 7756
Aktionen, 7695, 7696, 7698, 7701, 7702, 7704, 7773, 7778, 7779, 7780, 7781
alle Schritte deaktivieren, 8045
Alternativzweig, 7689, 7767
Bausteinaufrufe, 7800, 7801, 7803
Bausteinschnittstelle, 7707, 7710, 7714, 7718, 7720, 7721
Bedingungen, 7706, 7782, 7783, 7789, 7797
Betriebsarten, 8042, 8046
Detailvergleich, 7940
Einstellungen, 7729
Einstellungen ändern, 7731
Einzelschrittansicht, 7744
Grundlagen zu den Datentypen, 7776
Interlock, 7706
interne Parameter als remanent deklarieren, 7737
interne Parameter als sichbar und erreichbar aus HMI deklarieren, 7737
Kettenansicht, 7742
Kettenende, 7691
Lernmodus, 7784, 7786, 7787, 8049
Meldungen bearbeiten, 7807
Meldungsanzeige aktivieren, 7806
Meldungsanzeige deaktivieren, 7806
Nachgeschaltete permanente Anweisungen, 7746
Permanente Anweisungen, 7705
Programmierfenster, 7739
Programmiersprache, 7685
Programmiersprache festlegen, 7733
Programmstatusanzeige, 8054
Registrierungsanforderungen zurücksetzen, 7709
Schnittstellenparametersatz festlegen, 7735
Schritt, 7692, 7693, 7757, 7758, 7759, 7761, 7762, 7763, 7764, 7765
Schrittzeit-Überwachung, 7784, 7785, 7786, 7787, 8049
Simultanzweig, 7688, 7767
Speicherplatzmodell festlegen, 7736
Sprung zu Schritt, 7690, 7770, 7771, 7772
Supervision, 7706
Supervisionsfehler quittieren, 8046
Systemsynchronisation, 8042, 8047
T_MAX, 8049
T_MAX und T_WARN manuell festlegen, 7786
T_WARN, 8049
Testeinstellungen, 8043, 8044
Testen mit Programmstatus, 8041

Transitionen, 7694, 7757, 7759, 7761, 7763, 7765
Verzweigung löschen, 7769
Verzweigung schließen, 7690, 7768
Vorausgeschaltete permanente Anweisungen, 7741
GRAPH-Programme beobachten, 8054
Grenzwert, 771, 1130, 1786, 5151
Größer maximale Schrittaktivierungszeit, 2162, 2174
Größer Schrittaktivierungszeit, 2159, 2171, 5396, 5401
Größer ununterbrochene Schrittaktivierungszeit, 2161, 2172, 5397, 5402
Größer Warnungszeit, 2164, 2176

H

Haltepunkt
aktivieren, 8069, 8072, 8078
Aufruf ausführen, 8077
Aufrufumgebung einstellen, 8066
deaktivieren, 8069, 8072, 8078
löschen, 8075
Nächste Anweisung ausführen, 8077
Programmausführung bis zur Markierung fortsetzen, 8077
Programmausführung fortsetzen, 8077
setzen, 8067
Testen im Einzelschrittmodus, 8058, 8079, 8081, 8083
zu den Haltepunkten navigieren, 8073
Hardwaredatentypen, 366
Hardware-Kennung, 2849
ermitteln mit LOG2MOD, 2855
Herunterfahren (PC-System, 909, 1269, 1493, 1929
High_Speed_Counter, 2883
Arbeitsweise, 2883
Aufruf, 2883
Ausgangsparameter, 2890
Beschreibung, 2883
Eingangsparameter, 2888
Fehlerverhalten, 2887
Statische Variablen, 2891
HTA, 2254, 5441
HW-Kennung, 2849, 4032, 6987, 7392

I

I&M-Daten: auslesen mit Get_IM_Data, 2696
I_ABORT, 7070
I_GET, 7066
I_PUT, 7068
IB, 7376
ID, 7376
IEC-Prüfung, 507
einstellen, 372, 510, 581
IEC-Zeit, 269, 270
IF, 1908, 5181
IF_CONF für Projektierungsdaten (Telecontrol), 4053, 7002
IM0_Data, 2696
IMC, 964, 1325, 1526, 1972, 2140, 4570, 4792, 4894, 5248, 5375
Impuls, 645, 657, 671, 685, 1006, 1032, 1046, 1338, 1572, 1731, 1746, 1988, 4387, 4393, 4407, 4608, 4614, 4628, 4814, 4950, 5115, 5124, 5272
verlängert, 674, 687, 1035, 1048, 1574, 1749, 4396, 4409, 4617, 4630, 4952, 5126
Impuls erzeugen, 1988
Impulsfolge
mit vorgegebener Frequenz ausgeben, 2755
Impulsgenerator
aktivieren / deaktivieren mit CTRL_PWM, 2752
IN_RANGE, 733, 1094
INC, 763, 1123, 1602, 4982
Indirekte Adressierung, 119, 120, 126, 127, 131, 132, 135, 136, 137, 138, 139, 140, 208, 211
INIT_RD, 919, 1280, 1502, 1937, 2107
Initialschritt
abwählen, 7758
festlegen, 7758
Inkrementieren, 763, 1123, 1602, 4982
INSERT, 2267
Instanz-Datenbaustein
Datentyp bei IEC-Zeiten oder -Zählern ändern, 7655
Definition, 53
erzeugen, 53
Remanenzverhalten, 7338
Instanz-DB
anlegen, 7828, 7838
INT, 256, 383, 446, 516, 546, 585, 600
INT_TO_, 446, 546, 600
INTEG
Anweisung, 5976
Ausgangsparameter, 5978
Eingangsparameter, 5978
Interlock
Einführung, 7693, 7706
programmieren, 7782

Interne Parameter
als remanent deklarieren, 7737
als sichtbar und erreichbar aus HMI deklarieren, 7737
INV, 2111, 5333
INVD, 1641, 5023
INVERT, 930, 1291, 4510, 4733
Invertieren, 624, 930, 984, 1291, 1559, 1640, 1641, 4376, 4510, 4596, 4733, 4936, 5022, 5023
INVI, 1640, 5022
IO2MOD, 2856
IO-Link, 3568, 3571, 3579
IO-Systeme umkonfigurieren, 2411
IP_CONFIG, 6875
IP-Konfiguration
Parameter aus dem Anwenderprogramm ändern, 3768, 6481
IS_ARRAY, 748, 1108, 1375, 1779
IS_NULL, 745, 1105, 1372
I-Slave
Alarm senden mit SALRM, 5507
Prozessalarm auslösen mit DP_PRAL, 5514
ISO-on-TCP Verbindung
Parameterblock für, 6889
ITB, 1635, 5017
ITD, 1637, 5019
IW, 7376
IX, 7376

J

JBI, 1658, 5040
JC, 1653, 5035
JCB, 1655, 5037
JCN, 1654, 5036
JL, 1675, 5057
JM, 1668, 5050
JMP, 894, 1254, 4491, 4713
JMP_LIST, 897, 1257
JMPN, 895, 1255, 4492, 4714
JMZ, 1671, 5053
JN, 1665, 5047
JNB, 1657, 5039
JNBI, 1659, 5041
JO, 1660, 5042
JOIN, 2238
JOS, 1661, 5043
JP, 1666, 5048
JPZ, 1670, 5052
JU, 1652, 5034
JUO, 1673, 5055
JZ, 1663, 5045

K

Kanal-Datenbaustein, 6184
Kanal-DB
Aufbau, 6151, 6184
Aufgabe, 6151, 6184
Inhalt, 6152, 6184
Kanaldiagnose
Diagnosedaten, 5634
Kettenansicht, 7742
Kettenende
Ablaufketten beenden, 7755
Einführung, 7691
Know-how-Schutz
Baustein drucken, 7258
Bausteine öffnen, 7256
Einführung, 7239
einrichten, 7255
entfernen, 7261
Passwort ändern, 7260
Kommentarabschnitt, 7590
Kommentarabschnitt einfügen, 1921, 5193
Kommentare
in SCL-Programm einfügen, 7672
Kommunikation
Abfragearchitektur, 4098, 6502
Open User Communication, 6440
Zustandsdaten, 5712
Kommunikationsmodul (CM), 4243
Datenempfang, 4117, 6520
Programmieren, 4096, 6499
Kommunikationsschnittstellen
Programmieren, 4096, 6499
Komprimieren, 4497, 4719, 4848, 5195, 5324
Konfigurations-Datenbaustein, 6881
Konsistenzprüfung, 7975, 7981, 7982
Einführung, 7975, 7981
Konsistezprüfung, 7975
Konstante
Datentypen, 105
Definition, 102
globale Konstante, 103, 105, 106, 7391, 7392, 7396
lokale Konstante, 103, 105, 106, 7272, 7274, 7308
symbolische Kontante, 103
typisierte Konstante, 102
untypisierte Konstante, 102
Konstanten, 1601, 4981, 7627
Kontaktplan, 7425

Konvertieren, 876, 1236, 1634, 1635, 1637, 1638, 1639, 1884, 2080, 4480, 5016, 5017, 5019, 5020, 5021, 5172, 5314
einer Zeichenkette in Hexadezimal mit ATH, 2251, 5440
Hexadezimal in Zeichenkette mit HTA, 2254, 5441
Zahl in Zahlenzeichenkette konvertieren mit VAL_STRG, 2226
Zahlenzeichenkette in Zahl konvertieren mit STRG_VAL, 2222
Zeichen in Zahlenzeichenkette kopieren mit Chars_TO_Strg, 2234
Zeichenkette in Array mit Strg_TO_Chars, 2231
Zeichenkette mit S_CONV, 2219, 5437
Zeiten konvertieren mit T_CONV, 2180, 5404
Konvertierung, 369, 507, 578
explizit, 423, 425, 428, 431, 435, 439, 443, 446, 450, 453, 457, 461, 465, 468, 471, 474, 476, 479, 482, 485, 488, 490, 492, 494, 497, 499, 501, 504, 530, 531, 533, 536, 540, 543, 546, 550, 553, 557, 560, 563, 566, 569, 570, 571, 573, 574, 575, 577, 592, 593, 595, 597, 600, 603, 606, 607, 608, 610, 611, 612, 613, 614
implizit, 372, 373, 376, 378, 380, 381, 383, 384, 386, 388, 390, 392, 394, 396, 398, 400, 402, 404, 406, 408, 410, 412, 414, 416, 418, 419, 421, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 581, 582, 584, 585, 586, 587, 588, 589, 590, 591
KOP, 7425, 7482
Detailvergleich, 7930
KOP-Element
aus Zwischenablage einfügen, 7461
ausschneiden, 7460
einfügen, 7441, 7442
ersetzen, 7462
kopieren, 7460
löschen, 7465
Operand einfügen, 7466
Regeln beim Einfügen, 7439
selektieren, 7459
Kopieren
Bereich, 804, 866, 1164, 1226, 1396, 1464, 1808, 1874, 2029, 2071, 4470, 4692, 4833, 5162, 5304
Bereich ununterbrechbar, 811, 870, 1172, 1230, 1402, 1467, 1814, 1877, 2035, 2074, 4474, 4696, 4836, 5166, 5307
Wert, 789, 1150, 1383, 4468, 4690
Zeichenkette in Array mit Strg_TO_Chars, 2231
Kopierschutz, 7250
KOP-Netzwerk
Bausteinaufruf einfügen, 7449, 7516
Kreuzung, 7474
Kreuzung einfügen, 7474
Kreuzung löschen, 7476
Kreuzung umhängen, 7475
Programmstatusanzeige, 8050
Regeln zur parallelen Verzweigungen, 7471
Unzulässige Verschaltungen, 7441
Zweig, 7471
Zweig einfügen, 7472
Zweig löschen, 7473
Zweig schließen, 7473
KOP-Programme beobachten, 8050
Kreuzung
Definition, 7474
einfügen, 7474
löschen, 7476
umhängen, 7475

L

L, 1567, 1583, 1618, 4945, 4961, 5000
L DBLG, 1680, 5062
L DBNO, 1681, 5063
L DILG, 1682, 5063
L DINO, 1683, 5064
L STW, 1620, 5002
LABEL, 896, 1256, 4493, 4715
Laden, 1618, 1620, 1621, 1622, 1623, 1624, 1625, 1680, 1681, 1682, 1683, 5000, 5002, 5003, 5004, 5005, 5006, 5007, 5062, 5063, 5064
Ablaufkette vor dem Laden eines GRAPH-DB ausschalten, 7912
Bausteinänderungen ohne Reinitialisierung laden, 7889, 7891, 7892, 7893, 7894, 7895
Bausteine auf eine Memory Card laden, 7886, 7910
Bausteine in Gerät, 7875, 7900
Bausteine in Gerät laden, 7880, 7881, 7905, 7906
Bausteine von einer Memory Card laden, 7888, 7912
im Betriebszustand RUN, 7875, 7900
Zählwert, 1583, 1584, 4961, 4962
Zeitwert, 1567, 1569, 4945, 4947
Ladespeicher, 7983
anzeigen, 7987
LAG1ST
Anweisung, 5978
Ausgangsparameter, 5982
Eingangsparameter, 5981

LAG2ND
Ausgangsparameter, 5985
Eingangsparameter, 5985
LAR1, 1621, 5003
LAR1 <D>, 1622, 5004
LAR1 AR2, 1623, 5005
LAR2, 1624, 5006
LAR2 <D>, 1625, 5007
Laufzeit
messen mit RT_INFO, 2687
Laufzeitmessung, 923, 1283, 1506, 1941
LC, 1569, 1584, 4947, 4962
LDT, 274, 406, 485
LDT_TO_, 485
Lead- und Lag Algorithmus, 969, 1329, 1535, 1977, 2148, 4575, 4797, 4903, 5253, 5379
LEAD_LAG, 969, 1329, 1535, 1977, 2148, 4575, 4797, 4903, 5253, 5379
LEAVE, 5103
LED, 2693
LED-Zustand
auslesen mit LED, 2693
lesen über Systemzustandsliste, 5717
Leerbox
FUP-Element einfügen, 7510
KOP-Element einfügen, 7443
LEFT, 2259
LEN, 2256
Lernmodus, 7784, 7785, 7786, 7787, 8049
Leseauftrag
CNT2RDPN, 6141
Leseaufträge, 6162, 6164
Lesen
asynchron - Datensatz einer Baugruppe mit RD_DPARA, 2528, 5597
aus einem Datenbaustein im Ladespeicher mit READ_DBL, 2825, 5753
Datensatz einer Baugruppe mit RD_DPAR, 2522, 5595
eines Datensatzes mit RD_REC, 2428, 5490
Rückmeldesignale, 6143, 6147
Lesezeichen
Funktion, 7638
löschen, 7640
navigieren, 7640
setzen, 7639
LGC_GADR, 2869
LIMALARM
Anweisung, 5985
Ausgangsparameter, 5987
Eingangsparameter, 5987
LIMIT, 771, 1130, 1381, 1786, 2010, 4453, 4674, 4831, 5151, 5290
LIMITER
Anweisung, 5987
Ausgangsparameter, 5989
Eingangsparameter, 5989
Limitieren, 771, 1130, 1381, 1786, 2010, 4453, 4674, 4831, 5151, 5290
Lineare Programmierung, 37
LINT, 260, 390, 461
LINT_TO_, 461
LMNGEN_C
Anweisung, 5989
Ausgangsparameter, 5993
Durchgangsparameter, 5993
Eingangsparameter, 5992
Statische Variablen, 5993
LMNGEN_S
Anweisung, 5993
Ausgangsparameter, 5999
Durchgangsparameter, 6000
Eingangsparameter, 5999
Statische Variablen, 6000
LN, 776, 1136, 1612, 1790, 2013, 4457, 4678, 4993, 5155, 5294
LOG_GEO, 2864
LOG2GEO, 2853
LOG2MOD, 2855
Logarithmus, 776, 1136, 1612, 1790, 2013, 4457, 4678, 4993, 5155, 5294
Logische Adresse
einer Baugruppe ermitteln mit GADR_LGC, 2867, 5794
einer Baugruppe ermitteln mit RD_LGADR, 2865, 5793
zugehörigen Steckplatz ermitteln mit LGC_GADR, 2869, 5796
zugehörigen Steckplatz ermitteln mit LOG_GEO, 2864, 5791
Lokale Variable, 7263
Lokalzeit
aus der Basiszeit berechnen mit BT_LT, 5420
auslesen mit RD_LOC_T, 2195
auslesen und berechnen mit LOC_TIME, 5419
berechnen mit SET_TIMEZONE, 2200
schreiben mit WR_LOC_T, 2196
Uhrzeitalarm stellen mit S_LTINT, 5423
Long word, 252
LOOP, 1676, 5058
Löschen
Filter im Belegungsplan, 7965

LP_SCHED
Eingangsparameter, 5957
LP_ZONE
Anweisung, 6321
Ausgangsparameter, 6323
Durchgangsparameter, 6324
Eingangsparameter, 6323
Statische Variablen, 6324
LREAL, 265, 396, 471, 521, 563, 7623
LREAL_TO_, 471, 563
LTIME, 270, 402, 479
LTIME_TO_, 479
LTOD, 272, 414, 494
LTOD_TO_, 494
LWORD, 252, 378, 435
LWORD_TO_, 435

M

Machine Tailoring, (siehe IO-Systeme umkonfigurieren)
Master Control Relay, 4530, 4532, 4534, 4535, 4751, 4753, 4755, 4757, 5106, 5109, 5110, 5111
Master Control Relay (MCR)
Formalparameterzugriffe, 203
Funktionalität, 201
Hinweise zur Benutzung, 203
Parameterübergabe, 203
Mathematische Funktionen
CALCULATE, 7454, 7487, 7521, 7551
Matrixscanner, 967, 1327, 1530, 1975, 2144, 4572, 4795, 4899, 5251, 5377
MAX, 769, 1129, 1380, 1784, 2008, 4451, 4672, 4830, 5150, 5289
MAX_LEN, 2237
Maximal verfügbaren Ladespeicher anzeigen, 7987
Maximum, 769, 1129, 1380, 1784, 2008, 4451, 4672, 4830, 5150, 5289
MB, 7376
MB_CLIENT, 4271, 4291
MB_COMM_LOAD, 4254
MB_MASTER, 4257
MB_RED_CLIENT, 4320
MB_SERVER, 4281, 4306
MB_SLAVE, 4265
MC_AbortMeasuringInput, 3287, 3444
MC_CamIn, 3312, 3319, 3470, 3476
MC_CamTrack, 3291, 3293, 3449, 3452
MC_ChangeDynamic
Anweisung, 3166, 3202
Parameter, 3166, 3203
MC_CommandTable
Anweisung, 3164, 3200
Parameter, 3165, 3201
MC_Control, 6065
Arbeitsweise, 6066
Aufruf, 6066
Beschreibung, 6065
Handbetrieb, 6066
Nachführbetrieb, 6066
Parameter, 6066
Regelbetrieb, 6065
MC_DefineKinematicsZone, 3380
MC_DefineTool, 3389
MC_DefineWorkspaceZone, 3378
MC_GearIn, 3294, 3298, 3453, 3456, 6067
Arbeitsweise, 6068
Aufruf, 6072
Beschreibung, 6067
Parameter, 6072
MC_GearInPos, 3299, 3303, 3457, 3461
MC_GetCamFollowingValue, 3324, 3481
MC_GetCamLeadingValue, 3326, 3483
MC_GroupContinue, 3347, 3349
MC_GroupInterrupt, 3346
MC_GroupStop, 3351, 3353
MC_Halt, 3252, 3255, 3410, 3413
Anweisung, 3146, 3183
Funktionsdiagramm, 3148, 3185
Parameter, 3147, 3184, 3218
MC_Home, 3247, 3405, 6059
Anweisung, 3141, 3178
Arbeitsweise, 6059
Aufruf, 6062
Beschreibung, 6059
Bezugspunktsetzen, 6060
Parameter, 3143, 3180, 6063
Referenzpunktfahrt, 6059
MC_Init, 6050
Arbeitsweise, 6050
Aufruf, 6050
Beschreibung, 6050
MC_InterpolateCam, 3323, 3480
MC_MeasuringInput, 3280, 3282, 3437, 3439
MC_MeasuringInputCyclic, 3283, 3285, 3439, 3442
MC_MotionInPosition, 3331, 3334
MC_MotionInVelocity, 3328, 3330
MC_MoveAbsolute, 3256, 3259, 3414, 3417, 6051
Anweisung, 3149, 3186
Arbeitsweise, 6051
Aufruf, 6053
Beschreibung, 6051

Funktionsdiagramm, 3152, 3188
Parameter, 3150, 3187, 6053
MC_MoveCircularAbsolute, 3364, 3369
MC_MoveCircularRelative, 3371, 3376
MC_MoveJog, 3269, 3273, 3427, 3430, 6056
Anweisung, 3161, 3197
Arbeitsweise, 6056
Aufruf, 6058
Beschreibung, 6056
Funktionsdiagramm, 3163, 3199
Parameter, 3162, 3198, 6058
MC_MoveLinearAbsolute, 3354, 3358
MC_MoveLinearRelative, 3359, 3363
MC_MoveRelative, 3260, 3263, 3418, 3421, 6054
Anweisung, 3153, 3189
Arbeitsweise, 6054
Aufruf, 6055
Beschreibung, 6054
Funktionsdiagramm, 3155, 3192
Parameter, 3154, 3190, 6055
MC_MoveSuperimposed, 3274, 3277, 3431, 3434
MC_MoveVelocity, 3264, 3268, 3422, 3426
Anweisung, 3156, 3193
Funktionsdiagramm, 3160, 3196
MC_OutputCam, 3288, 3290, 3445, 3448
MC_PhasingAbsolute, 3304, 3307, 3462, 3465
MC_PhasingRelative, 3308, 3311, 3466, 3469
MC_Power, 3238, 3244, 3398, 3402
Anweisung, 3135, 3172
Funktionsdiagramm, 3139, 3176
Parameter, 3135, 3173
MC_ReadParam
Anweisung, 3168, 3204
Parameter, 3169, 3205
MC_Reset, 3140, 3177, 3245, 3403
MC_SetKinematicsZoneActive, 3386
MC_SetKinematicsZoneInactive, 3387
MC_SetOcsFrame, 3392
MC_SetSensor, 3278
MC_SetTool, 3391
MC_SetWorkspaceZoneActive, 3382
MC_SetWorkspaceZoneInactive, 3384
MC_Simulation, 6067
Arbeitsweise, 6067
Aufruf, 6067
Beschreibung, 6067
MC_StopMotion, 6063
Arbeitsweise, 6064
Aufruf, 6065
Beschreibung, 6063
Parameter, 6065
Unterbrechen einer Fahrt, 6064
MC_SynchronizedMotionSimulation, 3321, 3478
MC_TorqueAdditive, 3341, 3342
MC_TorqueLimiting, 3336, 3339, 3487
MC_TorqueRange, 3343, 3345
MC_WriteParam
Anweisung, 3171, 3207
Parameter, 3171, 3207
MCAT, 960, 1321, 1522, 1969, 2136, 4566, 4788, 4891, 5245, 5372
MCR, 4530, 4532, 4534, 4535, 4751, 4753, 4755, 4757, 5106, 5109, 5110, 5111
MCRA, 4534, 4755, 5110
MCRD, 4535, 4757, 5111
MD, 7376
Meldebausteine, 7818
Meldeklasse
anlegen, 7853
bearbeiten, 7854
kopieren, 7854
Meldung
Bestandteile, 7817
Formalparameter, 7822
GRAPH-Meldungen bearbeiten, 7807
GRAPH-Meldungsanzeige aktivieren und deaktivieren, 7806
Meldungsdatentyp, 7822
zeitgestempelt übertragen mit TIMESTMP, 5430
Meldungseditor
Aufbau, 7823
Meldungsnummer
vergeben, 7816
Meldungsprojektierung
Attribut entriegeln, 7849
Attribut verriegeln, 7848
Begleitwert löschen, 7834
Einführung, 7815
Infotext, 7845
Text eingeben, 7845
Text entriegeln, 7848
Text verriegeln, 7847
Zusatztext, 7845
Meldungstyp, 7820
bearbeiten, 7828, 7837
Memory Card
Informationen auslesen mit GetSMCinfo, 2716
Merker
Anzeige der remanenten Merker aktivieren, 7968
MID, 2263
MIN, 767, 1127, 1378, 1782, 2007, 4449, 4670, 4828, 5149, 5287
Minimum, 767, 1127, 1378, 1782, 2007, 4449, 4670, 4828, 5149, 5287

Mnemonik einstellen, 7130
MOD, 761, 1120, 1604, 4446, 4666, 4984, 7609
Modbus
    Modbus_Comm_Load, 4176, 6579
    Modbus_Slave, 4180, 4188, 6583, 6591
Modbus Master, 6710
Modbus_Comm_Load, 4176, 6579
Modbus_Slave, 4180, 4188, 6583, 6591
Modbus-Kommunikation
    als Modbus-Master mit MB_MASTER, 4257
    als Slave mit MB_SLAVE, 4265
    Port konfigurieren mit MB_COMM_LOAD, 4254
ModuleStates, 2732
MOVE, 789, 1150, 1383, 4468, 4690
MOVE_BLK, 804, 1164, 1396, 1808, 2029
MOVE_BLK_VARIANT, 348, 807, 1167, 1398, 1810, 2031
MoveVelocity
    Parameter, 3157, 3194
MSK_FLT, 2587
MUL, 758, 1117, 4443, 4664
Multicomputingalarm
    auslösen mit MP_ALM, 5631
Multiinstanz, 113
    Aufruftyp ändern, 7453, 7520, 7589
    Definition, 67
    deklarieren, 7282
Multiplexen, 936, 1297, 1947, 2116, 5207
Multiplizieren, 758, 1117, 1596, 1599, 1607, 4443, 4664, 4975, 4979, 4988
MUX, 936, 1297, 1947, 2116, 5207
MW, 7376
MX, 7376

N

N, 636, 639, 997, 1000, 4383, 4604
N_TRIG, 641, 1002, 4386, 4607
Nachkommastellen, 787, 1147, 1798, 2021
NE_ElemType, 744, 1104, 1371
NE_Type, 740, 1100, 1367
NEG, 762, 1122, 2005, 4447, 4667, 5285
NEGD, 1643, 5025
NEGI, 1642, 5024
Negieren, 762, 1122, 1642, 1643, 1644, 2005, 4447, 4667, 5024, 5025, 5026, 5285
NEGR, 1644, 5026
Netzwerk
    aufklappen, 7434, 7502, 7567, 7793
    einfügen, 7431, 7433, 7499, 7500, 7564, 7566, 7791, 7792
    in Bausteinen navigieren, 7437, 7505, 7570, 7796
    Kommentar eingeben, 7436, 7504, 7570
    kopieren, 7433, 7500, 7566, 7792
    löschen, 7433, 7501, 7566, 7793
    selektieren, 7432, 7500, 7565, 7791
    Titel einfügen, 7435, 7503, 7569, 7795
    verwenden, 7430, 7498, 7564, 7790
    zuklappen, 7434, 7502, 7567, 7793
Netzwerkkommentar
    ausblenden, 7128
    einblenden, 7128
Neue Datentypen, 207
NONLIN
    Anweisung, 6000
    Ausgangsparameter, 6003
    Eingangsparameter, 6003
NOP 0, 1724, 5113
NOP 1, 1725, 5114
NORM
    Anweisung, 6005
    Ausgangsparameter, 6006
    Eingangsparameter, 6006
NORM_X, 887, 1246, 1475, 1893, 2087
Normieren, 887, 1246, 1475, 1893, 2087
NOT, 624, 1559, 2111, 4376, 4936, 5333, 7617, 7618
NOT_NULL, 747, 1107, 1374
NOT_OK, 737, 1097
NULL, 356
Nullanweisung, 1724, 1725, 5113, 5114

O

O, 1544, 1549, 4920, 4925
O(, 1552, 4928
OD, 1699, 5079
ODER, 979, 981, 4592, 4593
OF, 1910, 5183
Öffnen einer Beobachtungstabelle, 8090
Öffnen einer Forcetabelle, 8120
Öffnerkontakt, 623, 4375
OK, 736, 1096
ON, 1546, 4921
ON(, 1553, 4929
Open User Communication, 6440
    Arbeitsweise der Anweisungen, 6441
    Daten empfangen mit TRCV, 3709, 3714, 6471
    Daten empfangen über UDP mit TURCV, 6478
    Daten senden mit TSEND, 3702, 3705, 6468
    Daten über UDP empfangen mit TURCV, 3738

Daten über UDP senden mit TUSEND, 3733, 6475
FW_IOT, 6491
FW_TCP, 6487
IP-Konfigurationsparameter ändern mit IP_CONF, 6481
IP-Konfigurationsparameter ändern mit T_CONFIG, 3768
Kommunikationsverbindung abbauen mit TDISCON, 3700
Kommunikationsverbindung aufbauen mit TCON, 3689, 3692
lokalen Kommunikationszugangspunkt auflösen mit TDISCON, 6467
Parametrierung der Verbindungen bei TCP und ISO on TCP, 6445
Struktur der Adressinformation des remoten Partners bei UDP, 6452
TCON, 6463
über Industrial Ethernet, 6440
Verbindung aufbauen und Daten lesen mit TRCV_C, 3641, 3646
Verbindung aufbauen und Daten senden mit TSEND_C, 3626, 3631
Open User Communication
Parametrierung des lokalen Zugangspunkts bei UDP, 6450
Operand, 93, 94, 100, 102, 103, 105, 106, 108, 109, 111, 113, 114, 115, 116, 283, 290, 294, 303
einfügen, 7466, 7533, 7579
Operand auf negative Signalflanke abfragen, 636, 997, 4383, 4604
Operand auf positive Signalflanke abfragen, 634, 995, 4382, 4602
Operand bei negativer Signalflanke setzen, 639, 1000
Operand bei positiver Signalflanke setzen, 638, 998
OPN, 4495, 4717
OPNI, 4496, 4718
Optimierter Bausteinzugriff, 206
Optimierter Zugriff, 55, 58
OR, 183, 927, 981, 1287, 4507, 4593, 4730, 7618
Organisationsbaustein (OB)
anlegen, 7202
Ereignis zuordnen mit ATTACH, 2534
Ereigniszuordnung aufheben mit DETACH, 2538
Funktion, 39
in externe Quelldatei exportieren, 7233
Laufzeit messen mit RT_INFO, 2687
Programmlaufzeit ermitteln mit OB_RT, 5744
Startinformation, 39
Startinformationen auslesen mit RD_SINFO, 2676, 5681
Uhrzeitalarm-OBs aktivieren aus Anwenderprogramm, 5609
Weckalarm-OB, Parameter abfragen mit QRY_CINT, 2546
Weckalarm-OB, Parameter setzen mit SET_CINT, 2544
Organisationsbausteine (OB)
Verzögerungsalarm-OB abfragen aus Anwenderprogramm, 5614
Verzögerungsalarm-OB aktivieren aus Anwenderprogramm, 5613
Verzögerungsalarm-OB stornieren aus Anwenderprogramm, 5614
OS, 183, 188, 189, 190, 196
OUT_RANGE, 735, 1095
OutputCPU314C, 6100
Aufruf, 6100
Beschreibung, 6100
Parameter, 6101
OutputET200S2AO, 6099
Aufruf, 6100
Beschreibung, 6099
Parameter, 6100
OutputIM174, 6104
Aufruf, 6105
Beschreibung, 6104
Parameter, 6105
Parametrierung der IM 174 (Hardware-Konfiguration), 6104
OutputIM178, 6098
Aufruf, 6098
Beschreibung, 6098
Digitalausgänge, 6099
Parameter, 6098
OutputMM4_DP, 6103
Aufruf, 6103
Beschreibung, 6103
Parameter, 6103
OutputSINAMICS, 6105
Aufruf, 6108
Beschreibung, 6105
Parameter, 6108
Parametrierung des SINAMICS in HWCN, 6105
OutputSM332, 6101
Aufruf, 6101
Beschreibung, 6101
Parameter, 6101
OutputSM432, 6099
Aufruf, 6099

Beschreibung, 6099
Parameter, 6099
OutputUniversal, 6102
Arbeitsweise, 6102
Aufruf, 6102
Beschreibung, 6102
Parameter, 6102
OV, 183, 188, 189, 190, 192, 193, 196, 201
OVERRIDE
Anweisung, 6006
Ausgangsparameter, 6008
Eingangsparameter, 6008
OW, 1693, 5074

P

P, 634, 638, 995, 998, 4382, 4602
P_RCV_RK, 6710
P_SND_RK, 6710
P_TRIG, 640, 1001, 4385, 4605
P3964_Config (Protokollkonfiguration), 4112, 6515
PARA_CLT
Ausgangsparameter, 6010
Eingangsparameter, 6009
PARA_CTL
Anweisung, 6008
Parameter, 6376
ABS_CTRL_451, 6148
ABS_DIAG_451, 6150
ABS_INIT, 6143
CAM_CTRL_452, 6180
CAM_DIAG_452, 6182
CAM_INIT, 6176
CAM_MSRM_452, 6183
CNT_CTL1, 6115
CNT_CTRL, 6113
CNT2_CTR, 6130
CNT2RDPN, 6129
CNT2WRPN, 6128
DIAG_INF, 6114
DIAG_RD, 6130
dynamische - übertragen mit WR_PARM, 5602
EncoderAbsSensorDP:, 6082
EncoderCPU314C, 6089
EncoderET200S1Count, 6085
EncoderET200S1SSI, 6080
EncoderFM350, 6087
EncoderFM450, 6078
EncoderIM174, 6092
EncoderIM178, 6076
EncoderSINAMICS, 6094
EncoderSM338, 6083
EncoderUniversal, 6096
ErrorID, 2893, 2899
interne - Einstellungen ändern, 7737
MC_Control, 6066
MC_GearIn, 6072
MC_Home, 6063
MC_MoveAbsolute, 6053
MC_MoveJog, 6058
MC_MoveRelative, 6055
MC_StopMotion, 6065
OutputCPU314C, 6101
OutputET200S2AO, 6100
OutputIM174, 6105
OutputIM178, 6098
OutputMM4_DP, 6103
OutputSINAMICS, 6108
OutputSM332, 6101
OutputSM432, 6099
OutputUniversal, 6102
S7-Basiskommunikation, 6376
Versteckte Parameter, 7467, 7534
vordefinierte - schreiben mit WR_DPARM, 2532, 5603
Parameter BUFFER, Send_P2P, 4116, 6519
Parameter LENGTH, Send_P2P, 4116, 6519
Parameter-DB
Aufbau, 6191
Aufgabe, 6191
Inhalt, 6192
Parameter-DB FM x51
Aufbau, 6157
Aufgabe, 6157
Inhalt, 6158
Parameterkonfiguration
LENGH und BUFFER für Send_P2P, 4116, 6519
Parametertypen, 363, 4079, 7029
Parameterübergabe, 78, 80, 83, 87, 89, 90, 91, 92
Übergabe als Kopie oder als Pointer, 84
Parameterversorgung
über Register aktivieren, 7561
Parametrieren
einer Baugruppe mit PARM_MOD, 5598
PARM_MOD, 5598
Passwort-Legitimierung, 904, 1264, 1488, 1922, 2095
Passwort-Legitimierung einschränken und freigeben, 904, 1264, 1488, 1922, 2095
PE_CMD, 2472, 5547
PE_DS3_Write_ET200S, 2477, 5552
PE_End_RSP, 2510, 5582
PE_Error_RSP, 2507, 5580
PE_Get_Mode_RSP, 2512, 5585

PE_I_DEV, 2503, 5576
PE_List_Modes_RSP, 2511, 5584
PE_Measurement_List_RSP, 2518, 5590
PE_Measurement_Value_RSP, 2519, 5592
PE_Mode_ID, 6844, 6868
PE_PEM_Status_RSP, 2514, 5587
PE_Start_End, 2467, 5543
PE_Start_RSP, 2508, 5581
PE_WOL, 2478, 5553
PEEK, 1439, 1850
  Speicheradresse lesen, 1439, 1850
PEEK_BOOL, 1441, 1852
  Speicherbit lesen, 1441, 1852
Performance, 205
Peripherie, 109
Permanente Anweisung
  einfügen, 7797
  Einführung, 7705
PID
  Anweisung, 6010
  Arbeitsweise, 6011
  Ausgangsparameter, 6013
  Eingangsparameter, 6013
PID_3Step
  Anweisung, 2960, 2994
  Ausgangsparameter, 2972, 3004
  Durchgangsparameter, 2974
  Eingangsparameter, 2970, 3002
  Statische Variablen, 3005
PID_Compact
  Anweisung, 2938
  Ausgangsparameter, 2911, 2942
  Durchgangsparameter, 2912
  Eingangsparameter, 2910, 2941
  Statische Variablen, 2943
PID_CP
  Ausgangsparameter, 5908
  Beschreibung, 5891
  Durchgangsparameter, 5909
  Eingangsparameter, 5906
PID_ES
  Ausgangsparameter, 5937
PID_FM
  Anweisung, 6207
  Arbeitsweise, 6208
  Ausgangsparameter, 6211
  Blockschaltbilder, 6218
  Durchgangsparameter, 6214
  Eingangsparameter, 6210
PID_FM_455
  Anweisung, 6288
  Arbeitsweise, 6289
  Blockschaltbilder, 6298
  Durchgangsparameter, 6295
  Eingangsparameter, 6291
PID_PAR
  änderbare Parameter, 6237
  Anweisung, 6234
  Ausgangsparameter, 6237
  Eingangsparameter, 6236
PID_PAR_455
  änderbare Parameter, 6318
  Anweisung, 6315
  Ausgangsparameter, 6317
  Eingangsparameter, 6317
PID_Temp
  Arbeitsweise, 3032
  Ausgangsparameter, 3040
  Durchgangsparameter, 3042
  Eingansparameter, 3038
  Kaskade, 3042
  Mode, 3042
  Parameter ErrorBits, 3086
  Parameter State und Mode PID_Temp, 3077
  PwmPeriode, 3092
  Variable ActivateRecoverMode, 3088
  Variable Warning, 3091
PING-Kommando, 6793
PIP_Mode, 3589
PLC-Datentyp, 231
  Adressieren, 290
  anlegen, 7412
  ARRAY deklarieren, 7415
  Definition, 286
  Deklaration in Datenbausteinen, 7335
  Deklarationstabelle für PLC-Datentypen, 7409
  deklarieren, 7418
  deklarieren in der Bausteinschnittstelle, 7281
  Eigenschaften von Variablen, 7419, 7420
  löschen, 7413
  Nummer ändern, 7414
  STRUCT deklarieren, 7417
  Struktur programmieren, 7415
  vergleichen, 7955
PLC-Datentypen, 207
PLC-Meldungen mit Anweisungen
  Anlaufverhalten der Anweisungen, 5653
  dynamisch belegte Systemressourcen auslesen mit READ_SI, 5677
  dynamisch belegte Systemressourcen freigeben mit DEL_SI, 5679
  Erzeugung, 5637
  freigeben mit EN_MSG, 5673

mit Begleitwerten für acht Signale mit ALARM_8P, 5645
mit Quittierungsanzeige mit ALARM, 5650
ohne Begleitwerte für acht Signale mit ALARM_8, 5643
ohne Quittierungsanzeige mit NOTIFY, 5648
ohne Quittierungsanzeige mit NOTIFY_8P, 5640
quittierbar mit ALARM_DQ, 5664
Quittierzustand ermitteln mit ALARM_SC, 5666
sperren mit DIS_MSG, 5675
stets quittiert mit ALARM_D, 5662
Störverhalten der Anweisungen, 5653
Systemressourcen mit ALARM_SQ und ALARM_S, 5657, 5659
Übersicht, 5654
PLC-Variable
beobachten, 7398
Definition, 100
deklarieren, 7382, 7383, 7387, 7389
Eigenschaften, 7394
importieren- und exportieren, 7402, 7403, 7404, 7405, 7406
kopieren, 7400
Offline/Offline-Vergleich, 7920
PLC-Variablentabelle, 7371, 7373, 7379, 7380
PLC-Variablentabelle vergleichen, 7920
Regeln, 7378
Remanenzverhalten, 7384, 7385, 7386
Vergleich, 7917
vergleichen, 7951
Zeilen sortieren, 7401
Zellen automatisch ausfüllen, 7401
Zulässige Adressen und Datentypen, 7378
PNIO_RECV, 6826
PNIO_SEND, 6819
POINTER, 356
POKE, 1443, 1854
Speicheradresse schreiben, 1443, 1854
POKE_BLK, 1446, 1857
Speicherbereich schreiben, 1446, 1857
POKE_BOOL, 1444, 1855
Speicherbit schreiben, 1444, 1855
POP, 1720, 5100
PORT_CFG, 4219
Port_Config (Portkonfiguration), 4101, 6505
Potenzieren, 788, 1148
Pragmas in SCL, 7671
PROFIenergy
Abgefragte Energiedaten als Antwort generieren, 2512, 5585
Abgefragte Engiesparmodi als Antwort generieren, 2511, 5584
Angeforderte Messwerte als Antwort generieren, 2519, 5592
Antwort auf Kommando zum Pausenende generieren, 2510, 5582
Antwort auf Kommando zum Pausenstart generieren, 2508, 5581
Beschreibung, 2464, 5541
Energiesparmodus starten und beenden, 2467, 2472, 5543, 5547
Liste der unterstützten Messwerte als Antwort generieren, 2518, 5590
Negative Antwort auf Kommando generieren, 2507, 5580
PE_CMD, 2472, 5547
PE_DS3_Write_ET200S, 2477, 5552
PE_End_RSP, 2510, 5582
PE_Error_RSP, 2507, 5580
PE_Get_Mode_RSP, 2512, 5585
PE_I_DEV, 2503, 5576
PE_Identify_RSP, 2516, 5589
PE_List_Modes_RSP, 2511, 5584
PE_Measurement_List_RSP, 2518, 5590
PE_Measurement_Value_RSP, 2519, 5592
PE_PEM_Status_RSP, 2514, 5587
PE_Start_End, 2467, 5543
PE_Start_RSP, 2508, 5581
PEM-Status als Antwort generieren, 2514, 5587
PROFIenergy-Kommandos im I-Device steuern, 2503, 5576
Schaltverhalten der Powermodule einstellen, 2477, 5552
Unterstützte PROFIenergy-Kommandos als Antwort generieren, 2516, 5589
Wake On LAN, 2478, 5553
Zustandsinformationen auslesen, 2472, 5547
PROFIenergy-Controller, 6843
PROFIenergy-Device, 6843
PROFINET IO-Devices
alle Ausgänge schreiben mit SETIO, 2347, 5461
einen Teil der Ausgänge schreiben mit SETIO_PART, 2357, 5464
einen Teil der Eingänge lesen mit GETIO_PART, 2355, 5462
konsistente Daten lesen mit DPRD_DAT, 2434, 5496
konsistente Daten schreiben mit DPWR_DAT, 2436, 5499
PROFINET CBA
Ausgänge der PROFINET-Schnittstelle aktualisieren mit PN_OUT, 7075

Eingänge der Programmschnittstelle aktualisieren mit PN_IN, 7074
Verschaltungen aktualisieren mit PN_DP, 7076
PROFINET CBA-Komponente
aktualisieren mit Anweisungen, 7072
Program_Alarm, 2624
Programm
Laufzeit ermitteln mit OB_RT, 5744
Programmiereditor
Allgemeine Einstellungen, 7150
Aufbau, 7118
Bausteinschnittstelle, 7119
Favoriten, 7119
Fläche des Programmierfensters vergrößern, 7127
Funktion, 7117
Funktionsleiste, 7119
Programmierfenster, 7119
Task Card Anweisungen, 7119
Task Card Testen, 7120
Tastaturbedienung, 7122
Programmierempfehlungen, 208
Programmierfenster
anpassen, 7634
Einzelschrittansicht, 7744
GRAPH, 7739
Kettenansicht, 7742
Nachgeschaltete permanente Anweisungen, 7746
SCL, 7633
Vorausgeschaltete permanente Anweisungen, 7741
Programmiersprache
FUP, 7493
GRAPH, 7685
KOP, 7425
Regeln zum Umschalten, 7156
umschalten, 7156, 7733
Programmierung
Linear, 37
PtP-Anweisungen, 4096, 6499
Strukturiert, 37
Programminformationen
Ansichten, 7959
anzeigen, 7959
im Belegungsplan, 7960
im Register, 7983
in der Abhängigkeitsstruktur, 7977
in der Aufrufstruktur, 7969
Programmmeldung
anlegen, 7826, 7835
bearbeiten, 7826, 7827, 7836
generieren mit Program_Alarm, 2624
löschen, 7827, 7837
Programmressourcen, 7201
Programmschleife, 1676, 5058
Programmstatus
Ablaufkette initialisieren, 8045
alle Schritte deaktivieren, 8045
Aufrufumgebung, 8029, 8030
ausschalten, 8032
AWL, 8052
Bausteine bearbeiten, 8038
Betriebsarten, 8042, 8046
einschalten, 8032
Funktion, 8027
FUP, 8051
GRAPH, 8041, 8045, 8046, 8054
KOP, 8050
SCL, 8053
Systemsynchronisation, 8042, 8047
Variable steuern, 8039
Programmsteuerung, 911, 1271, 1495, 1931, 2101, 4499, 4721, 4850, 5197, 5326
Projektnavigation
ausblenden, 7127
einblenden, 7127
PROTECT, 4501, 4723, 4851, 5198, 5327
Prozessabbild
Ausgänge schreiben mit SETIO, 2347, 5461
der Ausgänge aktualisieren mit UPDAT_PO, 2292, 5451
der Ausgänge synchronisieren mit SYNC_PO, 2297, 5454
der Eingänge synchronisieren mit SYNC_PI, 2295, 5452
Eingänge aktualisieren mit UPDAT_PI, 2289, 5449
Eingänge lesen mit GETIO, 2346, 5460
Prozessabbildbereich lesen mit GETIO_PART, 2355, 5462
Prozessabbildbereich übertragen mit SETIO_PART, 2357, 5464
Prozessalarm
beim DP-Master auslösen mit DP_PRAL, 5514
Prozessdaten
AA_Data, 3579
SA_Data, 3571
Prüfsumme
auslesen mit GetChecksum, 2713
PRVREC, 2440
PT, 1030
PtP-Anweisungen Rückgabewerte, 4100, 6503
PtP-Fehlerklassen, 4101, 6504

PtP-Kommunikation
Programmieren, 4096, 6499
PULSE, 6373
PULSEGEN
Anweisung, 5810
Arbeitsweise, 5812
Ausgangsparameter, 5820
Eingangsparameter, 5819
PULSEGEN_M
Anweisung, 6014
Arbeitsweise, 6015
Ausgangsparameter, 6022
Eingangsparameter, 6021
Pulsweitenmodulation mit PULSE, 6373
Punkt-zu-Punkt-Kopplung
Ausgangssignale setzen mit SGN_SET, 4238
Datenübertragung anstoßen mit SEND_PTP, 4232
Empfang freigeben mit RCV_PTP, 4235
Empfangsparameter projektieren mit RCV_CFG, 4224
Empfangspuffer löschen mit RCV_RST, 4236
Komunikationsparameter projektieren mit PORT_CFG, 4219
serielle Übertragungsparameter projektieren mit SEND_CFG, 4222
Signalzustand abfragen mit SGN_GET, 4237
Punkt-zu-Punkt-Programmierung, 4096, 6499
PUSH, 1719, 5100
PUT, 3600

Q

QB, 7376
QD, 7376
QRY_CINT, 2546
QRY_DINT, 2573
QRY_TINT, 2560
Quadrat, 773, 1133, 1609, 1788, 4454, 4676, 4990, 5153, 5292
Quadrat bilden, 2011
Quadratwurzel, 774, 1134, 1610, 1789, 4456, 4677, 4991, 5154, 5293
Quadratwurzel bilden, 2012
Querverweise
Angezeigte Objekte, 7989
anzeigen, 7998
Neues Objekt hinzufügen, 8017
Querverweisliste, (Siehe Querverweise)
anzeigen, 7998
Aufbau, 7991
Einführung, 7989
Einstellungen, 7994
Hinzufügen von neuen Objekten, 8017
Nutzen, 7989
Quellobjekt, 7991
Spalten sortieren, 7991
Überblick, 7998
Querverweislistliste
Vorteile, 7989
QW, 7376
QX, 7376

R

R, 627, 987, 1558, 1571, 1585, 4377, 4598, 4934, 4949, 4964
R_TRIG, 643, 1003, 1336, 1726
RALRM, 2367
RCV_CFG, 4224
RCV_PTP, 4235, 7049
RCV_RST, 4236
RCVREC, 2438
RD_ADDR, 2859
RD_DPAR, 2522
RD_DPARA, 2528
RD_DPARM, 2530
RD_LGADR, 2865
RD_LOC_T, 2195
RD_REC, 2428
RD_SINFO, 2676
RD_SYS_T, 2193
RDREC, 2306
RE_TRIGR, 910, 1270, 1494, 1930, 2100, 4499, 4721, 4850, 5196, 5325
READ_355
Anweisung, 6228
Ausgangsparameter, 6230
Eingangsparameter, 6229
READ_455
Anweisung, 6310
Ausgangsparameter, 6311
Eingangsparameter, 6310
READ_BIG, 1453, 1863
READ_DBL, 2825
READ_ERR, 2589
READ_LITTLE, 1449, 1859
ReadFromArrayDB, 843, 1203, 1428, 1840, 2061
ReadFromArrayDBL, 848, 1208, 1433, 1844, 2065
REAL, 264, 394, 468, 520, 560, 586, 606, 7623
REAL_TO_, 468, 560, 606
Receive_Config (Empfangskonfiguration), 4106, 6509

Receive_P2P (Punkt-zu-Punkt-Daten empfangen), 4117, 6520
Receive_Reset (Empfänger zurücksetzen), 4119, 6522
RecipeExport, 2758
RecipeImport, 2761
ReconfigIOSystem, 2411
Referenzen, 207
Regelkreisaufrufe sperren, 6046
Regeln
    AWL-Netzwerk, 7572
Regeln für das Beobachten von Schleifen mit S7-1200/1500, 8035
Regeln für das Beobachten von Schleifen mit S7-300/400, 8035
Regeln für das Beobachten von Strukturen (S7-1200/1500), 8036
Register Speicherauslastung
    Aufbau, 7985
Reihenfolge
    Auftragsabarbeitung, 6161, 6164
Relaisspule, 625, 4376
    negiert, 626
Remanente Merker
    Anzeige aktivieren, 7968
Remanenz, 55, 58
    Bausteinschnittstelle, 7269
    Datenbaustein, 7338, 7339
    Merker, Zeiten, Zähler, 7384
    PLC-Variable, 7384, 7385, 7386
Remanenzspeicher, 7983
Remanenzverhalten
    Merker, Zeiten, Zähler, 7385, 7386
REPEAT, 1916, 5189
REPL_VAL, 4548, 4769, 4871, 5225, 5353
REPLACE, 2269
REQ bei asynchron arbeitenden Anweisungen, 4370
RES_RCVB, 7052
RESET, 4542, 4764, 4866, 5220, 5349
RESET_BF, 630, 991
RESETI, 4546, 4767, 4869, 5223, 5351
RESETP, 4544, 4765, 4867, 5221, 5350
RET, 902, 1262, 4495, 4717
RET_VAL bei asynchron arbeitenden Anweisungen, 4370
RETURN, 1920, 5193
Rezept
    in CSV-Datei exportieren, 2758
    in Datenbaustein importieren, 2761
RIGHT, 2261
Ring-Puffer, 351
RLD, 1713, 5093
RLDA, 1716, 5097
RMP_SOAK
    Anweisung, 6022
    Ausgangsparameter, 6028
    Eingangsparameter, 6028
RND, 1647, 5029
RND-, 1650, 5032
RND+, 1649, 5031
ROC_LIM
    Anweisung, 6030
    Ausgangsparameter, 6036
    Eingangsparameter, 6035
ROL, 948, 1309, 1957, 2125, 4521, 4744, 5214, 5342
ROR, 946, 1307, 1955, 2123, 4519, 4742, 5212, 5340
Rotieren
    links, 948, 1309, 1713, 1715, 1716, 1957, 2125, 4521, 4744, 5093, 5095, 5097, 5214, 5342
    rechts, 946, 1307, 1717, 1955, 2123, 4519, 4742, 5098, 5212, 5340
ROUND, 879, 1238, 1885, 2081, 4481, 4703, 5173, 5315
RRD, 1715, 5095
RRDA, 1717, 5098
RS, 633, 994, 4380, 4601
RT, 668, 1029, 1348, 1743
RT_INFO, 2687
RTM, 366, 2212
Rückgabewerte
    ABS_CTRL, 6144
    ABS_CTRL_451, 6148
    ABS_DIAG, 6146
    ABS_DIAG_451, 6150
    ABS_INIT, 6143
    CAM_CTRL, 6177
    CAM_CTRL_452, 6180
    CAM_DIAG, 6178
    CAM_DIAG_452, 6182
    CAM_INIT, 6176
    CAM_MSRM_452, 6183
    PtP-Anweisungen, 4100, 6503
Rückgabewerte Empfangs-Laufzeit, 4117, 6520
Rückmeldesignale
    lesen, 6143, 6147
Rücksetzen
    Bereich bitweise, 4542, 4764, 4866, 5220, 5349
    Bereich byteweise, 4546, 4767, 4869, 5223, 5351
    Bitfeld, 630, 991, 4542, 4764, 4866, 5220, 5349
    IEC-Zeit, 668, 1348, 1743

Operand, 627, 633, 987, 994, 1558, 4377, 4380, 4598, 4601, 4934
Peripheriebereich, 4544, 4765, 4867, 5221, 5350
VKE, 1561, 4938
Zähler, 1585, 4964
Zeit, 1571, 4949
Runden, 879, 880, 881, 883, 1238, 1240, 1241, 1243, 1647, 1648, 1649, 1650, 1885, 1886, 1888, 2081, 2082, 2083, 2084, 4481, 4483, 4484, 4485, 4703, 4705, 4706, 4707, 5029, 5030, 5031, 5032, 5173, 5174, 5175, 5315, 5316, 5317, 5318
RUNTIME, 923, 1283, 1506, 1941
Laufzeitmessung, 923, 1283, 1506, 1941
RW_REC, 6835

S

S, 628, 988, 1558, 1586, 4378, 4599, 4935, 4964
S_AVERZ, 682, 1043, 4404, 4625
S_COMP, 2217
S_CONV, 2219
S_CU, 1772, 5143
S_EVERZ, 677, 1038, 4399, 4620
S_IMPULS, 671, 1032, 4393, 4614
S_MOVE, 2215
S_ODT, 677, 1038, 1752, 4399, 4620, 5129
S_ODTS, 680, 1041, 1754, 4402, 4623, 5131
S_OFFDT, 682, 1043, 1757, 4404, 4625, 5134
S_PEXT, 674, 1035, 1749, 4396, 4617, 5126
S_PULSE, 1746, 5124
S_SEVERZ, 680, 1041, 4402, 4623
S_VIMP, 674, 1035, 4396, 4617
S5TIME, 268, 398, 474, 588, 608
S5TIME_TO_, 474, 608
S7-Basiskommunikation
gemeinsame Parameter, 6376
I_ABORT, 7070
I_PUT, 7068
X_ABORT, 7087
X_GET, 7083
X_PUT, 7085
X_RCV, 7078
X_SEND, 7077
S7-Kommunikation
Anlaufverhalten, 6388
Anweisungsübersicht, 6381
Betriebszustands-Abfrage, 6383
Betriebszustands-Änderung, 6382
Daten an Drucker senden mit PRINT, 6423
Daten an eine remote Partner-Anweisung mit USEND, 3606, 6398
Daten an eine remote Partner-Anweisung senden mit BSEND, 3615, 6403
Daten an eine remote Partner-Anweisung senden mit USEND_S, 6415
Daten aus einer remoten CPU auslesen mit GET_S, 6411
Daten aus remoter CPU lesen mit GET, 6392
Daten in eine remote CPU schreiben mit PUT_S, 6413
Daten schreiben in remote CPU mit PUT, 6395
Daten von einer remoten Partner-Anweisung empfangen mit BRCV, 3617, 6406
Daten von einer remoten Partner-Anweisung empfangen mit URCV, 3608, 6400, 6418
Datenaustausch, 6382
Gerät in den Betriebszustand STOP setzen mit STOP, 6431
Gerätestatus abfragen mit STATUS, 6435
Neu- oder Kaltstart durchführen mit START, 6429
Nutzdatengröße, 3593, 6387
Parameter, 3591, 6384
Sende- und Empfangsparameter, 3592, 6387
Störverhalten, 6390
Verbindungs-Abfrage, 6383
Verbindungszustand abfragen mit C_CNTRL, 6409
Verbindungszustand ermitteln mit CONTROL, 6420
Wechsel des Gerätestatus empfangen mit USTATUS, 6437
Wiederanlauf durchführen mit RESUME, 6433
SA, 693, 1054, 1580, 4415, 4636, 4958
SAVE, 1562, 4529, 4750, 4939
SCALE, 889, 1249, 1482, 1904, 2090, 4487, 4709, 4843, 5177, 5319
SCALE_M
Anweisung, 6037
Ausgangsparameter, 6038
Eingangsparameter, 6038
SCALE_X, 884, 1244, 1473, 1891, 2085
Schachtelungstief, 63
Schachtelungstiefe, 38
Schattenspeicher, 7072
Schieben
links, 944, 1305, 1706, 1710, 1953, 2121, 4517, 4740, 5087, 5090, 5210, 5339
rechts, 942, 1303, 1703, 1705, 1708, 1711, 1952, 2120, 4515, 4738, 5083, 5085, 5088, 5092, 5209, 5337
Schieberegister, 4581, 4584, 4803, 4806, 4909, 4912, 5259, 5261, 5385, 5387

Schleife, 1908, 1912, 1915, 1916, 1918, 5181, 5185, 5188, 5189, 5190, 5191
Schleifen beobachten, 8034
Schleifen beobachten mit S7-1200/1500, 8035
Schleifen beobachten mit S7-300/400, 8035
Schleifensprung, 1676, 5058
Schließerkontakt, 621, 4373
Schlüsselwörter, 94
Schlüsselworthervorhebung, 7631
Schnittstelle, (Siehe Bausteinschnittstelle)
Schnittstellenkonfiguration
    Anweisungen, 4096, 6500
Schnittstellenparametersatz, 7707, 7710, 7714, 7735
Schreibauftrag
    CNT2WRPN, 6138
Schreibaufträge, 6161, 6164
Schreiben
    eines Datensatzes mit WR_REC, 2433, 5494
    Steuersignale, 6143, 6147
Schreibschutz, 7252
Schriftart, -größe und farbe einstellen, 7634
Schritt
    Aktionen, 7693
    alle Schritte deaktivieren, 8045
    einfügen, 7757
    Elemente, 7693
    Grundlagen, 7692
    Interlock, 7693
    kommentieren, 7762
    kopieren und einfügen, 7763
    löschen, 7765
    neu nummerieren, 7759
    Statische Parameter, 7718
    Supervision, 7693
    überspringen, 7764
    umbenennen, 7761
Schrittschaltwerk, 951, 1311, 1512, 1959, 2127, 4549, 4555, 4770, 4776, 4872, 4879, 5226, 5233, 5354, 5361
Schrittzeit-Überwachung
    Grundlagen, 7784
    programmieren, 7785
    T_MAX und T_WARN emitteln, 8049
    T_MAX und T_WARN ermitteln, 7787
    T_MAX und T_WARN manuell festlegen, 7786
Schutzstufe
    ändern, 4501, 4723, 4851, 5198, 5327
SCL, 7607, 7676, 7678
    Bereich einfügen, 7649
    Bereich kopieren und einfügen, 7650
    Bereich löschen, 7653
    Bereiche verwenden, 7646
    Detailvergleich, 7937
    In Bereichen navigieren, 7652
    Kommentar einfügen, 1921, 5193
    Pragmas einfügen, 7671
    Programmierfenster, 7633
SCL-Anweisung
    Datentyp ändern, 7655, 7656
    einfügen, 7643, 7644
    Grundlagen zu den Datentypen, 7654
    Regeln, 7642
    selektieren, 7674
SCL-Programm
    Anweisung einfügen, 7643, 7644
    Anweisung selektieren, 7674
    Bausteinaufruf einfügen, 7660, 7661, 7663, 7664, 7665, 7667
    Kommentare einfügen, 7672
    Parameterliste anzeigen und ausblenden, 7670
    Programmstatusanzeige, 8053
SCL-Programme beobachten, 8053
SD, 1576, 4954
SE, 687, 689, 1050, 1574, 1576, 4409, 4411, 4632, 4952, 4954
SEG, 972, 1332, 1538, 1979, 2151, 4586, 4810, 4914, 5263, 5389
SEL, 934, 1295, 1510, 1945, 2114, 4514, 4737, 4860, 5205, 5336
Selektieren, 934, 1295, 1510, 1945, 2114, 4514, 4737, 4860, 5205, 5336
SEND_CFG, 4222
Send_Config (Sendekonfiguration), 4105, 6508
Send_P2P (Punkt-zu-Punkt-Daten senden), 4113, 6516
    Parameter LENGH und BUFFER, 4116, 6519
SEND_PTP, 4232, 7047
SEND_RK, 7053
Sendetakt, 3589
Serialisieren, 799, 1160, 1391, 1803, 2025
Serialize, 799, 1160, 1391, 1803, 2025
SERVE_RK, 7062
Service_Request_ID, 6853
SET, 1560, 4537, 4758, 4861, 4937, 5216, 5344
SET_BF, 629, 989
SET_CINT, 2544
SET_CLKS, 5411
SET_TIMEZONE, 2200
SET_TINT, 2552
SET_TINTL, 2555
SETI, 4540, 4762, 4864, 5219, 5347
SETIO, 2347
SETIO_PART, 2357, 5464

SETP, 4538, 4759, 4863, 5217, 5345
Setzen
Bereich bitweise, 4537, 4758, 4861, 5216, 5344
Bereich byteweise, 4540, 4762, 4864, 5219, 5347
Bitfeld, 629, 989, 4537, 4758, 4861, 5216, 5344
Operand, 628, 631, 988, 992, 1558, 4378, 4379, 4599, 4600, 4935
Peripheriebereich, 4538, 4863, 5217, 5345
VKE, 1560, 4937
Zähler, 1586, 4964
Zähleranfangswert, 709, 1070, 4429, 4650
SF, 1580, 4958
SGN_GET, 4237
SGN_SET, 4238
SHL, 944, 1305, 1953, 2121, 4517, 4740, 5210, 5339
SHR, 942, 1303, 1952, 2120, 4515, 4738, 5209, 5337
SHRB, 4584, 4806, 4912, 5261, 5387
SHUT_DOWN, 909, 1269, 1493, 1929
SI, 685, 1046, 1572, 4407, 4628, 4950
Sicherheitsvorkehrungen beim Forcen von Variablen, 8116, 8134
Sieben-Segment-Anzeige, 972, 1332, 1979, 4586, 4810, 4914, 5263, 5389
Signal_Get (RS232-Signale abrufen), 4120, 6523
Signal_Set (RS232-Signale setzen), 4122, 6525
Simultanzweig
Einführung, 7688
erstellen, 7767
SIN, 778, 1138, 1612, 1792, 2015, 4459, 4681, 4994, 5157, 5295
SINT, 254, 380, 439, 514, 540
SINT_TO_, 439, 540
Sinus, 778, 1138, 1612, 1792, 2015, 4459, 4681, 4994, 5157, 5295
Skalieren, 884, 889, 1244, 1249, 1473, 1482, 1891, 1904, 2085, 2090, 4487, 4709, 4843, 5177, 5319
SLD, 1710, 5090
Slice-Zugriff, 115
Slicing, 208
SLW, 1706, 5087
SMC, 967, 1327, 1530, 1975, 2144, 4572, 4795, 4899, 5251, 5377
SMS - empfangen (TC_RECV), 4042, 6992
SMS - senden (TC_SEND), 4037, 6988
SMS-Text (TC_RECV), 4044, 6994
SMS-Text (TC_SEND), 4039, 6990
SMS-Text (TCON_PHONE), 4051, 7000
SNC_RTCB, 2209
Sommer-/Winterzeit
Uhrzeitstatus umstellen mit SET_SW_S, 5427
umschalten mit SET_SW, 5425
SP, 1572, 4950
SP_GEN
Anweisung, 6038
Ausgangsparameter, 6040
Eingangsparameter, 6040
SPA, 1652, 5034
Spalten ein- und ausblenden, 7288, 7350, 7402, 7424
SPB, 1653, 5035
SPBB, 1655, 5037
SPBI, 1658, 5040
SPBIN, 1659, 5041
SPBN, 1654, 5036
SPBNB, 1657, 5039
Speicher
komprimieren, 4497, 4719, 4848, 5195, 5324
Speicheradresse lesen, 1439, 1850
Speicheradresse schreiben, 1443, 1854
Speicherauslastung, 7983
Arbeitsspeicher, 7983
Code-Arbeitsspeicher, 7983
Daten-Arbeitsspeicher, 7983
Einführung, 7983
Ladespeicher, 7983
Remanenzdaten, 7983
Remanenzspeicher, 7983
Speicherauslastung der CPU
anzeigen, 7987
Speicherbereich
der Baugruppe lesen mit Systemzustandsliste, 5697
Speicherbereich schreiben, 1446, 1857
Speicherbereiche
Systemspeicher, lesen mit Systemzustandsliste, 5698
Speicherbit lesen, 1441, 1852
Speicherbit schreiben, 1444, 1855
Speichern
VKE im BIE-Bit, 1562, 4529, 4750, 4939
Speichern einer Beobachtungstabelle, 8091
Speichern einer Forcetabelle, 8120
Speicherplatzmodell, 7707, 7736
SPL, 1675, 5057
SPLIT, 2243
SPLT_RAN
Anweisung, 6040
Ausgangsparameter, 6042
Eingangsparameter, 6042
SPM, 1668, 5050

SPN, 1665, 5047
SPO, 1660, 5042
SPP, 1666, 5048
SPPZ, 1670, 1671, 5052, 5053
Springen, 894, 895, 896, 897, 899, 902, 1254, 1255, 1256, 1257, 1259, 1262, 1651, 1675, 4491, 4492, 4493, 4495, 4713, 4714, 4715, 4717, 5033, 5057
    absolut, 1652, 5034
    bei BIE = 0, 1659, 5041
    bei BIE = 1, 1658, 5040
    bei Ergebnis größer Null, 1666, 5048
    bei Ergebnis größer oder gleich Null, 1670, 5052
    bei Ergebnis kleiner Null, 1668, 5050
    bei Ergebnis kleiner oder gleich Null, 1671, 5053
    bei Ergebnis nicht Null, 1665, 5047
    bei Ergebnis ungültig, 1673, 5055
    bei OS = 1, 1661, 5043
    bei OV = 1, 1660, 5042
    bei VKE = 0, 1654, 1657, 5036, 5039
    bei VKE = 1, 1653, 1655, 5035, 5037
    beiErgebnis Null, 1663, 5045
Sprung
    Ablaufketten beenden, 7755
Sprung zu Schritt
    einfügen, 7770
    Einführung, 7690
    löschen, 7772
    Sprungziel ändern, 7771
    Zum Sprungziel navigieren, 7771
Sprungliste, 897, 1257, 1675, 5057
Sprungmarke, 896, 1256, 1651, 1919, 4493, 4715, 5033, 5192
Sprungverteiler, 899, 1259
SPS, 1661, 5043
SPU, 1673, 5055
SPZ, 1663, 5045
SQR, 773, 1133, 1609, 1788, 2011, 4454, 4676, 4990, 5153, 5292
SQRT, 774, 1134, 1610, 1789, 2012, 4456, 4677, 4991, 5154, 5293
SR, 631, 992, 4379, 4600
SRD, 1711, 5092
SRT_DINT, 2570
SRW, 1708, 5088
SS, 691, 1052, 1578, 4413, 4634, 4956
SSD, 1705, 5085
SSI, 1703, 5083
SSI_Absolute_Encoder, 2893
    Arbeitsweise, 2894
    Aufruf, 2893
    Ausgangsparameter, 2898
    Beschreibung, 2893
    Eingangsparameter, 2896
    Fehlerverhalten, 2896
    Statische Variablen, 2899
STA, 183
Startinformation, 39
Station
    Informationen auslesen mit GetStationInfo, 2707
Statische Parameter GRAPH-Funktionsbaustein, 7718, 7720, 7721
Statische Variablen
    High_Speed_Counter, 2891
    SSI_Absolute_Encoder, 2899
Status, 183
Status des Leseauftrags
    CNT2RDPN, 6141
Status des Schreibauftrags
    CNT2WRPN, 6140
Statusbit, 183, 188, 189, 190, 192, 193, 196, 201
    abfragen, 193, 197, 4747
Statusbits
    abfragen, 195, 199
Statuswort, 183, 188, 189, 190, 192, 193, 196, 197, 201
Steckplatz, 2849
    logische Adresse ermitteln mit LGC_GADR, 2869, 5796
    logische Adresse ermitteln mit LOG_GEO, 2864, 5791
Steuermodus, 8101
Steuern
    Schnelle Zähler, 2877, 2880
Steuersignale
    ABS_CTRL, 6144
    ABS_CTRL_451, 6148
    schreiben, 6143, 6147
Steuerungszeitalarm, 957, 960, 1318, 1321, 1519, 1522, 1966, 1969, 2133, 2136, 4563, 4566, 4785, 4788, 4888, 4891, 5241, 5245, 5369, 5372
Steuerwert
    zugelassene Operanden, 8094
STP, 911, 1271, 1495, 1931, 2101, 4499, 4721, 4850, 5197, 5326
Strg_TO_Chars, 2231
STRG_VAL, 2222
STRING, 131, 277, 419, 501, 528, 529, 575, 577, 591, 613
    Adressieren, 283
    Deklarieren in der Bausteinschnittstelle, 7280
    Variablen vergleichen mit S_COMP,
STRING/WSTRING
    textuelle Bausteinschnittstelle, 7300
STRING_TO_, 501, 575, 577, 613

Strompfad
einfügen, 7478
löschen, 7478
Verwendung, 7477
STRUCT
Adressieren, 294
Aufbau, 291
Deklaration in globalen Datenbausteinen, 7332
Deklarieren in der Bausteinschnittstelle, 7279
Deklarieren in einem PLC-Datentyp, 7417
textuelle Bausteinschnittstelle, 7294
Struktur
STRUCT, 291
Strukturen beobachten, 8036
Strukturierte Programmierung, 37
SUB, 756, 1116, 4441, 4662
Subblock-Typen, 6883
Subtrahieren, 756, 1116, 1595, 1598, 1606, 4441, 4662, 4974, 4978, 4987
Zeiten subtrahieren mit T_SUB, 2184, 5406
Supervision
Einführung, 7693, 7706
programmieren, 7783
Supervisionsfehler quittieren, 8046
SV, 687, 1048, 1574, 4409, 4630, 4952
SWAP, 841, 1201, 1839, 2058, 5301
SWITCH, 899, 1259
Anweisung, 6042
Ausgangsparameter, 6043
Eingangsparameter, 6043
Symbole
im Belegungsplan, 7962
in der Abhängigkeitsstruktur, 7979
in der Aufrufstruktur, 7971
Symbole in der Beobachtungstabelle, 8087
Symbole in der Forcetabelle, 8118
Symbolik, 205
Symbolische Adressierung, 209
Symbolische Programmierung
Einblenden absoluter Adressen, 7131
SYNC_PI, 2295
SYNC_PO, 2297
Synchronfehlerereignisse
abfragen aus Ereignisstatusregister mit READ_ERR, 2589, 5624
demaskieren mit DMSK_FLT, 2588, 5623
maskieren mit MSK_FLT, 2587, 5622
Synchronisation
Uhrzeitslaves mit SNC_RTCB, 2209, 5413
von DP-Slaves mit DP_SYC_FR, 2444, 5516
Syntaxfehler
Fehler finden, 7154
Grundlagen, 7154
Systembaustein
Ordner Systembausteine, 7201
Systemdaten
Parameterblock für, 6882
Systemkonstante, 7392
Systemsynchronisation
beim Testen von GRAPH-Programmen, 8042
durchführen, 8047
Systemzeit lesen mit TIME_TCK, 2210, 5410
Systemzustandsliste
Aufbau einer Teilliste, 5691
Definition, 5690
SZL-ID W#16#00B1 - Baugruppendiagnoseinfo, 5740
SZL-ID W#16#00B2 - Diagnosedatensatz1 über physikalische Adresse, 5742
SZL-ID W#16#00B3 - Baugruppendiagnosedaten über logische Basisadresse, 5743
SZL-ID W#16#00B4 - Diagnosedaten eines DP-Slaves, 5743
SZL-ID W#16#0x94 - Baugruppenträger-/Stationszustandsinformation, 5729
SZL-ID W#16#xy11 - Baugruppen-Identifikation, 5694
SZL-ID W#16#xy12 - CPU-Merkmale, 5695
SZL-ID W#16#xy13 - Anwenderspeicherbereiche, 5697
SZL-ID W#16#xy14 - Systembereiche, 5698
SZL-ID W#16#xy15 - Bausteintypen, 5699
SZL-ID W#16#xy1C - Identifikation einer Komponente, 5700
SZL-ID W#16#xy22 - Alarmstatus, 5706
SZL-ID W#16#xy25 - Zuordnung zwischen Teilprozessabbildern und OBs, 5709
SZL-ID W#16#xy32 - Kommunikationszustandsdaten, 5712
SZL-ID W#16#xy74 - Zustand der Baugruppen-LEDs, 5717
SZL-ID W#16#xy90 - DP-Mastersystem-Information, 5719
SZL-ID W#16#xy91 - Baugruppenzustandsinformation, 5720
SZL-ID W#16#xy92 - Baugruppenträger-/Stationszustandsinformation, 5726
SZL-ID W#16#xy95 - Erweiterte DP-Mastersystem-Information, 5731
SZL-ID W#16#xy96 - Baugruppenzustandsinformation PROFINET IO und PROFIBUS DP, 5732

SZL-ID W#16#xy9C - Werkzeugwechslerinformationen (PROFINET IO), 5735
SZL-ID W#16#xyA0 - Diagnosepuffer, 5739
SZL-ID, Aufbau, 5692
SZL-Teillistenauszug auslesen mit RDSYSST, 5683
Teillisten, Übersicht, 5693
Teillistenauszug SZL-ID W#16#0132 Index W#16#0005, 5713
Teillistenauszug SZL-ID W#16#0132 Index W#16#0008, 5713
Teillistenauszug SZL-ID W#16#0132 Index W#16#000B, 5715
Teillistenauszug SZL-ID W#16#0132 Index W#16#000C, 5716
Teillistenauszug SZL-ID W#16#0232 Index W#16#0004, 5716
SZ, 709, 1070, 4429, 4650

T

T, 1626, 5008
T STW, 1628, 5010
T_ADD, 2182
T_COMBINE, 2188
T_COMP, 2178
T_CONFIG, 3768
T_CONV, 2180
T_DIAG, 3758
T_DIFF, 2186
T_RESET, 3753
T_SUB, 2184
Tabelle
Daten verteilen / sammeln mit PACK, 5787
Ersten Wert ausgeben mit FIFO, 5767
Letzten Wert ausgeben mit LIFO, 5771
Operationen ausführen mit TBL, 5773
Standardabweichung berechnen mit DEV, 5780
Tabellenwerte vergleichen mit CDT, 5782
verknüpfen mit TBL_TBL, 5784
Wert eintragen mit ATT, 5766
Wert kopieren mit TBL_WRD, 5775
Wert logisch verknüpfen und speichern mit WRD_TBL, 5777
Wert suchen mit TBL_FIND, 5769
Tabulatorweite einstellen, 7634
TAD, 1646, 5028
TADDR_Param, 3742
TADDR_RCV_IP, 3744
TADDR_SEND_QDN, 3744
TAK, 1718, 5099
Tan, 1794, 5158
Tangens, 781, 1141, 1616, 1794, 2017, 4462, 4684, 4998, 5158, 5297
TAR, 1629, 5011
TAR1, 1629, 5012
TAR1 <D>, 1630, 5013
TAR1 AR2, 1631, 5014
TAR2, 1632, 5014
TAR2 <D>, 1633, 5015
Task Card
nach Anweisung suchen, 7133
Tastaturbedienung
AWL, 7122
Editieren von Texten, 7122
FUP, 7122
GRAPH, 7122
KOP, 7122
Programmiereditor, 7122
SCL, 7122
Tauschen, 1629, 1645, 1646, 1680, 5011, 5027, 5028, 5061
TAW, 1645, 5027
TCON, 3689, 3692
TCON_IP_RFC für Telecontrol-Verbindungen, 4048, 6998
TCON_IP_V4 für Telecontrol-Verbindungen, 4048, 6998
TCON_PHONE für Telecontrol-Verbindungen, 4048, 6998
TCON_WDC für Telecontrol-Verbindungen, 4048, 6998
TCONT_CP
Anweisung, 5820
Arbeitsweise, 5821
Ausgangsparameter, 5835
Durchgangsparameter, 5836
Eingangsparameter, 5834
Statische Variablen, 5836
TCONT_S
Anweisung, 5842
Arbeitsweise, 5844
Ausgangsparameter, 5850
Durchgangsparameter, 5850
Eingangsparameter, 5849
Statische Variablen, 5850
TCP-Verbindung
Parameterblock für, 6887
TDB, 1680, 5061
TDISCON, 3700
Telegrammkonfiguration
Anweisungen, 4096, 6500

TeleService
AS_DIAL, 7093
AS_MAIL, 7105
E-Mail senden, 7105
E-Mail übertragen mit TM_MAIL, 4362
PG_DIAL, 7088
SMS Nachricht senden, 7099
SMS_SEND, 7099
Verbindung zu PG/PC aufbauen, 7088
Verbindung zur AS aufbauen, 7093
Testeinstellungen von GRAPH
ändern, 8044
Übersicht, 8043
Testmöglichkeiten in der Forcetabelle, 8115
Texte aus Textlisten in Meldungen integrieren, 7851
Beispiel, 7852
Textlisten für Meldungen, 7850
Textuelle Bausteinschnittstelle
ARRAY, 7297
Attribute für Variablen, 7302
Aufbau, 7290
Bausteindeklaration, 7291
Bearbeiten, 7304
Beispiel, 7306
Deklaration und Initialisierung von Variablen und Konstanten, 7293
Deklarationsabschnitte, 7292
Instanzen, 7301
Kommentare, 7303
Standard-Schnittstelle eines Organisationsbausteins wiederherstellen, 7305
STRING/WSTRING, 7300
STRUCT, 7294
THEN, 1908, 5181
THIS, 111
TIME, 269, 400, 476, 522, 566, 587, 607
TIME_TCK, 2210
TIME_TO_, 476, 566, 607
TIMER, 363
TIO_DI, 3575
TIO_DQ, 3583
TIO_IOLink_IN, 3571
TIO_IOLink_OUT, 3579
TIO_SYNC, 3568
TIO_Time
TIO_DI, 3575
TIO_DQ, 3583
TIO_IOLink_IN, 3571
TIO_SYNC, 3568
UDT "TIO_SYNC_Data", 3589
TM_MAIL, 4362
TMAIL_C, 3662
TMAIL_C V4.0, 3667
TO, 1912
TOD, 271, 412, 492, 524, 570, 590, 610
TOD_TO_, 492, 570, 610
TOF, 651, 663, 1012, 1023, 1343, 1737, 1992, 4391, 4612, 4819, 5121, 5277
TON, 648, 660, 1009, 1021, 1340, 1734, 1990, 4389, 4610, 4816, 5118, 5274
TONR, 654, 665, 1015, 1026, 1345, 1740, 1995, 5382
TONR_X, 4578, 4800, 4906, 5256
TP, 645, 657, 1006, 1018, 1338, 1731, 1988, 4387, 4608, 4814, 5115, 5272
Transferieren, 1626, 1628, 1629, 1630, 1631, 1632, 1633, 5008, 5010, 5012, 5013, 5014, 5015
Transition
einfügen, 7757
Einführung, 7694
kopieren und einfügen, 7763
löschen, 7765
neu nummerieren, 7759
programmieren, 7789
Statische Parameter, 7720
umbenennen, 7761
TRCV, 3709, 3714
TRCV_C, 3641, 3646
TRUE, 643, 644, 1003, 1005, 1336, 1337, 1726, 1727
TRUNC, 883, 1243, 1648, 1889, 2084, 4485, 4707, 5030, 5176, 5318
TSEND, 3702, 3705
TSEND_C, 3626, 3631
TUN_EC
Anweisung, 5853
Arbeitsweise, 5857
Ausgangsparameter, 5864
Durchgangsparameter, 5866
Eingangsparameter, 5862
TUN_ES
Anweisung, 5872
Arbeitsweise, 5876
Ausgangsparameter, 5881
Durchgangsparameter, 5884
Eingangsparameter, 5880
TURCV, 3738
TUSEND, 3733
TypeOf, 1777
TypeOfElements, 1778
typisiert, 7627

U

U, 1542, 4917
U(, 1550, 4926
Überblick über die Querverweisliste, 7998
Überlagern
    Variable, 116
Überlagern von Variablen, 7281
Überlauf, 183, 196
Übersetzen
    Baustein, 7868
    Bausteine, 7865
    Konsistenzprüfung, 7865
    Übersetzungfehler beheben, 7870
Übersicht
    über die Anzeigeformate in der Beobachtungstabelle, 8096
    über die Anzeigeformate in der Forcetabelle, 8124
UBLKMOV, 870, 1230, 1467, 1877, 2074, 4474, 4696, 4836, 5166, 5307
UC, 1690, 5071
UD, 1697, 5078
UDINT, 259, 388, 457, 519, 557
UDINT_TO_, 457, 557
UDP- Verbindung
    Parameterblock für, 6888
UDT, 207, 231
UDT "TIO_SYNC_Data", 3589
UFILL_BLK, 818, 1178, 1408, 1819, 2040
Uhrzeit, 271, 272, 274
    Basiszeit berechnen mit LT_BT, 5421
    der CPU stellen mit WR_SYS_T, 2190, 5408
    Lokalzeit berechnen mit BT_LT, 5420
    Lokalzeit berechnen mit SET_TIMEZONE, 2200
    stellen und Uhrzeitstatus setzen mit SET_CLKS, 5411
    Systemzeit der CPU lesen mit TIME_TCK, 2210, 5410
    Uhrzeit und Datum der CPU lesen mit RD_SYS_T, 2193, 5409
    Uhrzeitalarm stellen mit S_LTINT, 5423
    zwischen Slaves und Master synchronisieren, 5413
Uhrzeitalarm
    abfragen mit QRY_TINT, 2560, 5610
    aktivieren mit ACT_TINT, 2558, 5609
    Anweisungen, 5604
    Eigenschaften von SET_TINT, CAN_TINT, ACT_TINT, QRY_TINT, 5605
    stellen mit SET_TINT, 2552, 5607
    stellen mit SET_TINTL, 2555
    stornieren mit CAN_TINT, 2557, 5608
Uhrzeitstatus
    verstellen mit SET_SW_S, 5427
Uhrzeitstatus setzen mit SET_CLKS, 5411
UINT, 257, 384, 450, 517, 550
UINT_TO_, 450, 550
ULINT, 261, 392, 465
ULINT_TO_, 465
Umkonfigurieren von IO-Systemen, 2411
UMOVE_BLK, 811, 1172, 1402, 1814, 2035
Umschalten zwischen Basismodus und Erweitertem Modus in der Beobachtungstabelle, 8086
Umschalten zwischen Basismodus und Erweitertem Modus in der Forcetabelle, 8118
Umwandeln
    EA-Adresse in HW-Kennung, 2849
    HW-Kennung in Steckplatz, 2849
    Steckplatz in HW-Kennung, 2849
Umwandler
    Zeiten konvertieren mit T_CONV, 2180, 5404
UN, 1543, 4919
UN(, 1551, 4927
UND, 976, 978, 4591, 4592
Ungleich, 747, 1107, 1374
Universelles-serielles-Schnittstellen-Protokoll, (Siehe USS-Protokoll)
UNSCALE, 892, 1251, 1485, 1906, 2092, 4489, 4711, 4845, 5179, 5321
UNTIL, 1916, 5189
untypisiert, 7627
UPDAT_PI, 2289
UPDAT_PO, 2292
URCV, 3608
USEND, 3606
USINT, 255, 381, 443, 515, 543
USINT_TO_, 443, 543
USS_DRIVE, 4246
USS_Drive_Control, 4141, 6544
USS_Drive_Control / USS_Drive_Control_31, 4149, 6552
USS_PORT, 4245
USS_Port_Scan, 4141, 4145, 6544, 6548
USS_Port_Scan_31, 4145, 6548
USS_Read_Param, 4141, 6544
USS_Read_Param / USS_Read_Param_31, 4153, 6556
USS_RPM, 4249
USS_WPM, 4251
USS_Write_Param, 4141, 6544
USS_Write_Param / USS_Write_Param_31, 4155, 6558

USS-Kommunikation
Datenaustsch mit Antrieben über USS_DRIVE, 4246
Parameter aus Antrieb lesen mit USS_RPM, 4249
Parameter im Antrieb ändern mit USS_WPM, 4251
Übertragung zu Antrieb steuern mit USS_PORT, 4245
USS-Protokoll
Anweisungen, 4241
USS-Statuscodes, 4252
UW, 1691, 5072

V

VAL_STRG, 2226
Variable
negativ, 644, 1005, 1337
PLC-Variable, 7371
PLC-Variablen und DB-Variablen, 100
positiv, 643, 1003, 1336
überlagern, 116
Übersicht, 100
Variableninformationen ein- oder ausblenden, 7468, 7535, 7580, 7656, 7798
Variablen
Alle beobachten, 8104, 8131
Sofort beobachten, 8105, 8132
Variablendeklaration
ARRAY deklarieren, 7277
basierend auf einem PLC-Datentyp, 7335
Bausteinschnittstelle, 7272
Bausteinschnittstelle aktualisieren, 7314
Eigenschaften von Variablen, 7308, 7310, 7341, 7344, 7345
Gültige Datentypen, 7266, 7268
Multiinstanz, 7282
PLC-Datentyp deklarieren, 7281
Remanenz, 7269
Reservierte Schlüsselwörter, 94
Spalten ein- und ausblenden, 7288, 7350, 7402, 7424
STRING deklarieren, 7280
STRUCT deklarieren, 7279
Tabellenzeile am Ende einfügen, 7287, 7348, 7422
Tabellenzeile einfügen, 7287, 7348, 7400, 7422
Variable deklarieren, 7274, 7275, 7281
Variable importieren und exportieren, 7350
Variable löschen, 7287, 7348, 7401, 7422
Variablen importieren und exportieren, 7289
Variablen überlagern, 7281
Zeilen sortieren, 7401
Zellen automatisch ausfüllen, 7288, 7349, 7401, 7423
Zweck der Variablendeklaration, 7263
Variableninformationen ein- oder ausblenden, 7468, 7535, 7580, 7656, 7798
variabler Index, 211
VARIANT, 339, 342, 343, 348, 351, 738, 740, 742, 744, 854, 856, 858, 1098, 1100, 1102, 1104, 1214, 1215, 1217, 1361, 1365, 1367, 1369, 1371, 1457, 1458, 1460, 1477, 1480, 1777, 1778, 1867, 1869, 1870, 1899, 1901
VARIANT_TO_DB_ANY, 1477, 1899
VariantGet, 854, 1214, 1457, 1867
VariantPut, 856, 1215, 1458, 1869
Verbindung
abbrechen zu Partner außerhalb der eigenen S7-Station mit X_ABORT, 7087
abbrechen zu Partner innerhalb der eigenen S7-Station mit I_ABORT, 7070
aufbauen mit TCON, 6463
diagnostizieren mit C_DIAG, 5748
Zustand abfragen mit C_CNTRL, 6409
Verbindung überprüfen, 3758
Verbindung zurücksetzen, 3753
Verbindungs-ID, 6886
Wertebereich, 6886
Verbindungstypen
Parameterblöcke für, 6886
Verdrahtung testen, 8084
Vergleichen
Bitmaske, 964, 1325, 1526, 1972, 2140, 4570, 4792, 4894, 5248, 5375
Ganzzahlen (16-Bit), 1589, 4968
Ganzzahlen (32-Bit), 1591, 4969
Gleich, 712, 1074, 4433, 4654
Gleitpunktzahlen, 1592, 4971
Größer, 728, 1089, 2159, 2161, 2162, 2164, 2171, 2172, 2174, 2176, 4438, 4659, 5396, 5397, 5401, 5402
Größer gleich, 723, 1084, 4435, 4656
Kleiner, 731, 1091, 4439, 4660
Kleiner gleich, 725, 1086, 4437, 4657
Ungleich, 718, 1079, 4434, 4655
Zeichenketten mit S_COMP, 2217, 5436
Vergleicher
Zeitvariablen, 2178, 5403
Verknüpfen
EXKLUSIV ODER, 929, 1289, 4508, 4731
OD, 1699, 5079
ODER, 927, 1287, 4507, 4730

OW, 1693, 5074
UD, 1697, 5078
UND, 926, 1286, 4506, 4728
UW, 1691, 5072
XOD, 1701, 5081
XOW, 1695, 5076
Verknüpfungsergebnis, 183, 196
Verknüpfungspfad
einfügen, 7541
löschen, 7542
Verwendung, 7540
Versteckte Parameter, 7467, 7534
Verwendete Datenbausteine
ABS_CTRL, 6144
ABS_CTRL_451, 6147
ABS_DIAG, 6145
ABS_DIAG_451, 6149
ABS_INIT, 6142
CAM_CTRL, 6177
CAM_CTRL_452, 6180
CAM_DIAG, 6178
CAM_DIAG_452, 6181
CAM_MSRM_452, 6183
Verwendung von UDTs, 231
Verzögerungsalarm
abfragen mit QRY_DINT, 2573, 5614
Anweisungen, 2569, 5611
starten mit SRT_DINT, 2570, 5613
stornieren mit CAN_DINT, 2572, 5614
Verzweigung, 1556, 1908, 1910, 4932, 5181, 5183
löschen, 7769
schließen, 7768
Verzweigung schließen, 7690
VKE, 183, 196
auf 0 rücksetzen, 1561, 4938
auf 1 setzen, 1560, 4937
invertieren, 624, 984, 1559, 4376, 4596, 4936
VKE auf negative Signalflanke abfragen, 641, 1002, 4386, 4607
VKE auf positive Signalflanke abfragen, 640, 1001, 4385, 4605
VOID, 363
Vorgehensweise für das Beobachten von Strukturen, 8037
Vorzeichen, 766, 1125, 1608, 2006, 4448, 4669, 4989, 5286

W

WAIT, 921, 1281, 1504, 1939, 2108, 4500, 4722, 4851, 5197, 5326
Wake On LAN, 2478, 5553
WCHAR, 276, 418, 499
WCHAR_TO_, 499
Weckalarm-OB
Parameter abfragen mit QRY_CINT, 2546
Parameter setzen mit SET_CINT, 2544
Wert konvertieren, 4702
Wertübergabe
CNT_CTL1, 6123
DB-Parameter für Messbetriebsarten, 6127
DB-Parameter für Zählbetriebsarten, 6125
Wertzuweisung, 7620
WHILE, 1915, 5188
Wiederanlauf
durchführen mit RESUME, 6433
WORD, 250, 374, 428, 512, 533, 583, 595
WORD_TO_, 428, 533, 595
WORD_TO_BLOCK_DB, 614
Wortverknüpfungen, 201
WR_DPARM, 2532
WR_LOC_T, 2196
WR_REC, 2433
WR_SYS_T, 2190
WRIT_DBL, 2829
WRITE_BIG, 1455, 1865
WRITE_LITTLE, 1451, 1861
WriteToArrayDB, 845, 1205, 1430, 1842, 2063
WriteToArrayDBL, 851, 1211, 1436, 1847, 2068
WRREC, 2326
WSR, 4581, 4803, 4909, 5259, 5385
WSTRING, 131, 280, 421, 504
Adressieren, 283
WSTRING_TO_, 504
Wurzel Quadrat, 1789, 5154
WWW Anwenderseiten synchronisieren, 4002

X

X, 1547, 4923
X(, 1554, 4930
XN, 1548, 4924
XN(, 1555, 4931
XOD, 1701, 5081
XOR, 929, 981, 982, 1289, 4508, 4593, 4594, 4731, 7617, 7618
XOW, 1695, 5076

Z

Z_RUECK, 705, 1066, 4426, 4647
Z_VORW, 703, 1065, 4424, 4645
ZAEHLER, 707, 1068, 1775, 4427, 4648, 5146

Zählen
rückwärts, 705, 711, 1066, 1073, 1354, 1588, 1766, 1773, 1999, 4426, 4431, 4647, 4653, 4823, 4967, 5138, 5144
vorwärts, 703, 710, 1065, 1071, 1352, 1587, 1764, 1772, 1997, 4424, 4430, 4645, 4651, 4821, 4966, 5136, 5143
vorwärts und rückwärts, 707, 1068, 1357, 1769, 1775, 2002, 4427, 4648, 4825, 5140, 5146
Zähler
freigeben, 1582, 4960
rücksetzen, 1585, 4964
Schnelle Zähler, 2877, 2880
Schnelle Zähler steuern, 2877
Schnelle Zähler steuern (erweitert), 2880
setzen, 1586, 4964
Zähler aus Anwenderprogramm steuern mit COUNT, 6365
Zähler-DB
Aufbau, 6118
Aufgabe, 6118, 6130
Inhalt, 6118, 6131
Zehnerkomplement, 974, 1333, 1539, 1981, 2152, 4588, 4811, 4915, 5265, 5391
Zeichen, 276
Zeichenkette, 277, 280
aus Hexadezimal konvertieren mit HTA, 2254, 5441
in Hexadezimal konvertieren mit ATH, 2251, 5440
konvertieren mit S_CONV, 2219, 5437
kopieren, 5437
Länge ermitteln mit LEN, 2256, 5443
linke Zeichen auslesen mit LEFT, 2259, 5444
Maximale Länge ermitteln mit MAX_LEN, 2237
mittlere Zeichen auslesen mit MID, 2263, 5446
rechte Zeichen auslesen mit RIGHT, 2261, 5445
String-Variablen vergleichen mit S_COMP, 2217, 5436
verschieben mit S_MOVE, 2215
Zahl in Zeichenkette konvertieren mit VAL_STRG, 2226
Zahlenzeichenkette konvertieren mit STRG_VAL, 2222
Zeichen einfügen mit INSERT, 2267, 5447
Zeichen ersetzen mit REPLACE, 2269, 5448
Zeichen finden mit FIND, 2271, 5449
Zeichen in Zeichenkette kopieren mit Chars_TO_Strg, 2234
Zeichen löschen mit DELETE, 2265, 5446
Zeichenketten zusammenfassen mit CONCAT, 2258, 5444
Zeiger
ANY, 359
POINTER, 356
VARIANT, 339
Zeilenkommentar, 7590
Zeilennummern anzeigen, 7634
Zeit, 268, 269, 270, 671, 1032, 4393, 4614
erfassen, 665
freigeben, 1566, 4943
rücksetzen, 1571, 4949
Sommer-/Winterzeit umschalten mit SET_SW, 5425
Zeit akkumulieren, 1026, 1995, 4578, 4800, 4906, 5256
Zeit als Ausschaltverzögerung starten, 1023
Zeit als Einschaltverzögerung starten, 1021
Zeit als Impuls starten, 1018
Zeit rücksetzen, 1029
Zeitakkumulator, 654, 1015, 1345, 1740, 1995, 4578, 4800, 4906, 5256, 5382
Zeitdauer, 268, 670, 1030, 1350, 1744
Zeiten
addieren mit der Anweisung T_ADD, 2182, 5405
Datum und Uhrzeit zusammenfassen mit T_COMBINE, 2188
Differenz ermitteln mit T_DIFF, 2186, 5407
konvertieren mit T_CONV, 2180, 5404
subtrahieren mit T_SUB, 2184, 5406
Zeitvariablen vergleichen mit T_COMP, 2178, 5403
zusammenfassen mit T_COMBINE, 5407
Zeitstempel
bei Meldungsübertragung mit TIMESTMP, 5430
Zeitverzögerung, 921, 1281, 1504, 1939, 2108, 4500, 4722, 4851, 5197, 5326
Zielsystem herunterfahren, 909, 1269, 1493, 1929
ZONE
Anweisung, 6327
Durchgangsparameter, 6330
Eingangsparameter, 6329
Statische Variablen, 6330, 6334
ZR, 711, 1073, 4431, 4653
Zugelassene Operanden für die Forcetabelle, 8122
Zugelassenen Operanden für die Beobachtungstabelle, 8092
Zugriff auf Operanden, 108, 109, 111, 113, 114, 115, 116, 283, 290, 294, 303
Zugriffskoordinierung, 6786
Zuweisung, 625, 985, 1557, 4376, 4597, 4933, 7620
negiert, 626, 986
ZV, 710, 1071, 1587, 1588, 4430, 4651, 4966, 4967

Zweierkomplement, 762, 1122, 1642, 1643, 1644, 2005, 4447, 4667, 5024, 5025, 5026, 5285
Zweig
    Definition, 7471, 7537
    einfügen, 7472, 7538
    löschen, 7473, 7538
    Regeln, 7471, 7537
    schließen, 7473
Zyklusüberwachungszeit, 910, 1270, 1494, 1930, 2100, 4499, 4721, 4850, 5196, 5325

SIEMENS

WinCC
Prozesse visualisieren

Systemhandbuch

Bilder erstellen 1

Mit Variablen arbeiten 2

Mit Meldungen arbeiten 3

Mit Archiven arbeiten 4

Zugriff auf Archivdaten über WinCC OLE DB-Provider 5

Mit Rezepturen arbeiten 6

Mit Protokollen arbeiten 7

Eine Benutzerverwaltung projektieren 8

Mit Systemfunktionen und Runtime Scripting arbeiten 9

Mobile Panels 10

Aufgaben planen 11

Diagnosefunktionen nutzen 12

Mit Steuerungen kommunizieren 13

Übergreifende Funktionen nutzen 14

Übersetzen und Laden 15

Leistungsmerkmale 16

Optionen 17

Umstieg auf WinCC im TIA Portal 18

Ausdruck der Onlinehilfe

10/2018
Ausdruck der Onlinehilfe

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

Dokumentbestellnummer: Ausdruck der Onlinehilfe
℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Bilder erstellen....33

1.1 Grundlagen....33
1.1.1 Grundlagen zu Bildern....33
1.1.2 Grundlagen zu Bildern....35
1.1.3 Geräteabhängiger Funktionsumfang von Bildern....36
1.1.4 Elemente und Grundeinstellungen....38
1.1.4.1 Task Cards....38
1.1.4.2 Ansicht zoomen....40
1.1.4.3 Ansicht verschieben....40
1.1.5 Mit Bildern arbeiten....41
1.1.5.1 Arbeitsschritte....41
1.1.5.2 Arbeitsschritte....42
1.1.5.3 Neues Bild anlegen....42
1.1.5.4 Neues Bild anlegen....43
1.1.5.5 Bilder verwalten....44
1.1.5.6 Startbild des Projekts definieren....46
1.1.5.7 Startbild des Projekts definieren....47
1.1.5.8 Bildschirmauflösung ändern....48
1.1.6 Arbeiten mit Vorlagen....49
1.1.6.1 Grundlagen zum Arbeiten mit Vorlagen....49
1.1.6.2 Globales Bild....50
1.1.6.3 Neue Vorlage anlegen....52
1.1.6.4 Permanentbereiche in der Vorlage projektieren....53
1.1.6.5 Vorlage verwalten....54
1.1.6.6 Vorlage im Bild verwenden....55
1.1.7 Arbeiten mit Pop-up-Bildern....56
1.1.7.1 Grundlagen zum Arbeiten mit Pop-up-Bildern....56
1.1.7.2 Neues Pop-up-Bild anlegen....57
1.1.7.3 Pop-up-Bilder verwalten....58
1.1.7.4 Pop-up-Bild verwenden....59
1.1.8 Arbeiten mit Slide-in-Bildern....60
1.1.8.1 Grundlagen zum Arbeiten mit Slide-in-Bildern....60
1.1.8.2 Slide-in-Bilder konfigurieren....62
1.1.8.3 Slide-in-Bilder für Geräte ohne Multitouch-Funktion konfigurieren....64
1.1.8.4 Slide-in-Bilder verwenden....65
1.1.9 Arbeiten mit Designs....66
1.1.9.1 Grundlagen zum Arbeiten mit Designs....66
1.1.9.2 Editor Designs....66
1.1.9.3 Design erstellen....68
1.1.9.4 Design ändern....69
1.1.9.5 Design löschen....69
1.1.9.6 Design anwenden....70
1.1.9.7 Basisdesign der Bedienoberfläche eines Bediengeräts festlegen....71
1.1.9.8 Designvorlagen auf Bilder anwenden....71
1.1.9.9 Designvorlagen auf Basisobjekte und Elementen anwenden....72
1.1.9.10 Designvorlagen auf Controls anwenden....72

1.1.9.11 Standarddesign verwenden....73
1.1.10 Arbeiten mit Stilen und Formatvorlagen....74
1.1.10.1 Grundlagen zum Arbeiten mit Stilen....74
1.1.10.2 Vordefinierte Stile anzeigen lassen....76
1.1.10.3 Stil definieren....77
1.1.10.4 Stile verwalten....79
1.1.10.5 Schriftgröße an Gerätegröße anpassen....80
1.1.10.6 Stilelemente verwenden....82
1.1.10.7 Grundlagen zum Arbeiten mit Formatvorlagen....84
1.1.10.8 Neue Formatvorlage erstellen....85
1.1.10.9 Formatvorlagen verwalten....87
1.1.10.10 Formatvorlagen verwenden....88

1.2 Arbeiten mit Objekten und Objektgruppen....91
1.2.1 Arbeiten mit Objekten....91
1.2.1.1 Übersicht über die Objekte....91
1.2.1.2 Übersicht über die Objekte....95
1.2.1.3 Übersicht über die Objekte....100
1.2.1.4 Bearbeitungsmöglichkeiten von Objekten....103
1.2.1.5 Objekt einfügen....105
1.2.1.6 Objekt löschen....106
1.2.1.7 Objekt positionieren....107
1.2.1.8 Größe eines Objekts ändern....109
1.2.1.9 Mehrere Objekte auswählen....111
1.2.1.10 Objekte ausrichten....112
1.2.1.11 Objekte durch Tastenkombinationen verschieben....114
1.2.1.12 Objekt nach vorn oder hinten schieben....115
1.2.1.13 Objekte außerhalb des Bildbereichs einblenden....116
1.2.1.14 Objekt drehen....116
1.2.1.15 Objekt spiegeln....118
1.2.1.16 Objekt gestalten....119
1.2.1.17 Objekt gestalten....119
1.2.1.18 Füllmuster gestalten....120
1.2.1.19 Bildlaufleisten gestalten....122
1.2.1.20 Vordefinierte Stile anwenden....125
1.2.1.21 Texte in einem Objekt formatieren....126
1.2.1.22 Variablen und Textlisten im Text anbinden....127
1.2.1.23 Grafiken in einem Objekt formatieren....129
1.2.1.24 Objekt in Runtime drehen....130
1.2.1.25 Tabellenbasierte Objekte gestalten....132
1.2.1.26 Rahmen gestalten....133
1.2.1.27 Farbverlauf definieren....134
1.2.1.28 Bereiche konfigurieren....136
1.2.1.29 Blinken....138
1.2.1.30 Blinken....139
1.2.1.31 Mehrere Objekte eines Typs einfügen (stempeln)....139
1.2.1.32 Position und Größe mehrerer Objekte verändern....141
1.2.1.33 Objekte verbergen oder anzeigen....142
1.2.1.34 Anwenderdatentypen mit Objekten verknüpfen....142
1.2.1.35 Eigene Controls verwalten....145
1.2.1.36 Externe Grafiken....149
1.2.1.37 Externe Grafiken verwalten....150
1.2.1.38 Externe Grafik in der Grafiksammlung hinterlegen....152

1.2.1.39 Grafiken in Controls individuell anpassen....154
1.2.2 Arbeiten mit Objektgruppen....160
1.2.2.1 Grundlagen zu Gruppen....160
1.2.2.2 Objekte gruppieren....161
1.2.2.3 Gruppierung aufheben....163
1.2.2.4 Objekte zur Gruppe hinzufügen....164
1.2.2.5 Objekte aus der Gruppe entfernen....165
1.2.2.6 Objekt in einer Gruppe bearbeiten....166
1.2.3 Konfigurieren des Tastenzugriffs....168
1.2.3.1 Übersicht zum Tastaturzugriff....168
1.2.3.2 Übersicht zum Tastaturzugriff....169
1.2.3.3 Cursormodus festlegen....170
1.2.3.4 Bedienberechtigung und Bedienfreigabe für ein Objekt festlegen....171
1.2.3.5 Objekttypen für den Schaltcursormodus festlegen....172
1.2.3.6 Tab-Reihenfolge festlegen....173
1.2.3.7 Tab-Reihenfolge festlegen....175
1.2.4 Beispiele....176
1.2.4.1 Beispiel: Rechteck einfügen und konfigurieren....176
1.2.4.2 Beispiel: Rechteck einfügen....177
1.2.4.3 Beispiel: Rechteck konfigurieren....178

1.3 Arbeiten mit Textlisten und Grafiklisten....180
1.3.1 Arbeiten mit Textlisten....180
1.3.1.1 Grundlagen zu Textlisten....180
1.3.1.2 Textliste erstellen....182
1.3.1.3 Texte und Werte einer Bereich-Textliste zuordnen....183
1.3.1.4 Texte und Werte einer Bereich-Textliste zuordnen....185
1.3.1.5 Texte und Werte einer Bit-Textliste zuordnen....187
1.3.1.6 Texte und Werte einer Bitnummer-Textliste zuordnen....188
1.3.1.7 Hinweise zur Bitnummer-Textliste....190
1.3.1.8 Objekt mit Textliste projektieren....191
1.3.1.9 Objekt mit Textliste projektieren....193
1.3.2 Arbeiten mit Grafiklisten....194
1.3.2.1 Grundlagen zu Grafiklisten....194
1.3.2.2 Grafikliste erstellen....196
1.3.2.3 Grafikliste erstellen....197
1.3.2.4 Grafik und Werte einer Bereich-Grafikliste zuordnen....198
1.3.2.5 Grafiken und Werte einer Bereich-Grafikliste zuordnen....200
1.3.2.6 Grafiken und Werte einer Bit-Grafikliste zuordnen....202
1.3.2.7 Grafiken und Werte einer Bitnummer-Grafikliste zuordnen....203
1.3.2.8 Hinweise zur Bitnummer-Grafikliste....205
1.3.2.9 Objekt mit Grafikliste projektieren....206
1.3.3 Arbeiten mit C-Textlisten....208
1.3.3.1 Grundlagen zur C-Textliste....208
1.3.3.2 Einträge in die C-Textliste erstellen....208

1.4 Dynamisieren von Bildern....210
1.4.1 Grundlagen zur Dynamisierung von Bildern....210
1.4.1.1 Grundlagen zur Dynamisierung von Bildern....210
1.4.1.2 Grundlagen zur Dynamisierung von Bildern....210
1.4.1.3 Dynamisieren im Inspektorfenster....211
1.4.1.4 Dynamisieren im Inspektorfenster....213
1.4.1.5 Übersicht der Dynamisierungen einblenden....216

1.4.2 Dynamisieren mit Animationen....217
1.4.2.1 Neue Animation projektieren....217
1.4.2.2 Gestaltung eines Objekts dynamisieren....219
1.4.2.3 Bewegung projektieren....221
1.4.2.4 Direkte Bewegung projektieren....223
1.4.2.5 Bedienbarkeit eines Bedienobjekts dynamisieren....224
1.4.2.6 Sichtbarkeit eines Objekts dynamisieren....226
1.4.2.7 Animation mit Variablenanbindung projektieren....227
1.4.2.8 Animationen von Objektgruppen....229
1.4.2.9 Animationen von Objektgruppen....230
1.4.2.10 Animationen bei Mehrfachauswahl....231
1.4.3 Dynamisierung mit Eigenschaftsanimation....232
1.4.3.1 Dynamisieren mit Eigenschaftsanimationen....232
1.4.3.2 Eigenschaftsanimation vom Typ Bereich projektieren....234
1.4.3.3 Eigenschaftsanimation vom Typ Direkt projektieren....236
1.4.3.4 Eigenschaftsanimation vom Typ Bool projektieren....237
1.4.4 Dynamisieren in der Eigenschaftsliste....239
1.4.4.1 Grundlagen zum Dynamisieren in der Eigenschaftsliste....239
1.4.4.2 Objekteigenschaft mit Variable dynamisieren....242
1.4.4.3 Dynamisierung an eine Variablenänderung projektieren....244
1.4.4.4 Objekteigenschaft mit lokalen Skripten dynamisieren....245
1.4.5 Dynamisieren mit Systemfunktionen....246
1.4.5.1 Grundlagen zu Ereignissen....246
1.4.5.2 Systemfunktion an das Ereignis "Klicken" projektieren....246
1.4.5.3 Beispiel: Schaltfläche zur Sprachumschaltung projektieren....247

1.5 Arbeiten mit Funktionstasten....248
1.5.1 Arbeiten mit Funktionstasten....248
1.5.2 Funktionstasten global belegen....250
1.5.3 Funktionstasten lokal belegen....252
1.5.4 Einer Funktionstaste eine Funktion zuweisen....253
1.5.5 Bedienberechtigung für eine Funktionstaste vergeben....255
1.5.6 Einer Funktionstaste eine Grafik zuweisen....257
1.5.7 LED-Variablen projektieren....259
1.5.8 Beispiel: Funktionstasten zur Bildnavigation verwenden....261

1.6 Arbeiten mit Ebenen....263
1.6.1 Grundlagen zum Arbeiten mit Ebenen....263
1.6.2 Objekte auf Ebenen verschieben....264
1.6.3 Aktive Ebene einstellen....265
1.6.4 Ebenen ein- und ausblenden....267
1.6.5 Ebenen ein- und ausblenden....269
1.6.6 Ebenen und Objekte in Runtime verbergen....270
1.6.7 Ebenen umbenennen....271

1.7 Arbeiten mit Bildbausteinen....273
1.7.1 Grundlagen zu Bildbausteinen....273
1.7.1.1 Grundlage zu Bildbausteinen....273
1.7.1.2 Geräteabhängigkeit von Bildbausteinen....274
1.7.1.3 Editor Bildbausteine....275
1.7.2 Bildbaustein-Typen erzeugen und verwalten....278
1.7.2.1 Bildbaustein-Typ erzeugen....278
1.7.2.2 Bildbaustein-Typ konfigurieren....280
1.7.2.3 Kategorie und Eigenschaft eines Bildbaustein-Typs bearbeiten....286

1.7.2.4 Variable im Bildbaustein-Typ projektieren....287
1.7.2.5 Skripte im Bildbaustein-Typ projektieren....288
1.7.2.6 Ereignis im Bildbaustein-Typ konfigurieren....289
1.7.2.7 Ereignis im Bildbaustein-Typ konfigurieren....291
1.7.2.8 Textliste im Bildbaustein-Typ projektieren....292
1.7.2.9 Grafikliste im Bildbaustein-Typ projektieren....293
1.7.2.10 Texte im Bildbaustein-Typ direkt übersetzen....295
1.7.2.11 Texte im Bildbaustein-Typ exportieren und importieren....296
1.7.2.12 Bildbaustein-Typ bearbeiten....298
1.7.2.13 Größe eines Bildbausteins anpassen....299
1.7.2.14 Größe eines Bildbausteins anpassen....300
1.7.2.15 Instanz des Bildbaustein-Typs erstellen....301
1.7.2.16 Bildbaustein von Bildbaustein-Typ lösen....302
1.7.3 Bildbausteine dynamisieren....303
1.7.3.1 Grundlagen zur Dynamisierung von Bildbausteinen....303
1.7.3.2 Bildbaustein dynamisieren....304
1.7.4 Beispiel: Bildbaustein erstellen und dynamisieren....305
1.7.4.1 Beispiel: Bildbaustein projektieren....305
1.7.4.2 Beispiel: Bildbaustein-Typ erzeugen....305
1.7.4.3 Beispiel: Bildbaustein-Typ konfigurieren....306
1.7.4.4 Beispiel: Skript im Bildbaustein-Typ erstellen....307
1.7.4.5 Beispiel: Instanz des Bildbaustein-Typs erstellen....309
1.7.4.6 Beispiel: Enthaltene Objekte projektieren....310

1.8 Bildnavigation projektieren....312
1.8.1 Grundlagen zur Bildnavigation....312
1.8.1.1 Grundlagen zur Bildnavigation....312
1.8.1.2 Grundlagen zur Bildnavigation....312
1.8.1.3 Schaltfläche mit Bildwechsel belegen....313
1.8.1.4 Funktionstaste mit einem Bildwechsel belegen....315
1.8.2 Menüs und Symbolleisten für Runtime Professional....316
1.8.2.1 Grundlagen....316
1.8.2.2 Arbeiten mit Menüs und Symbolleisten....320
1.8.2.3 Beispiel: Bildwechsel mit Symbolleisten projektieren....331

1.9 Anzeige- und Bedienobjekte....337
1.9.1 Geräteabhängigkeit der Objekte....337
1.9.1.1 Objekte für Basic Panels....337
1.9.1.2 Objekte für Comfort Panels....338
1.9.1.3 Objekte für Mobile Panels....340
1.9.1.4 Objekte für WinCC Runtime Advanced....343
1.9.1.5 Objekte für WinCC Runtime Professional....344
1.9.2 Objekte....347
1.9.2.1 Balken....347
1.9.2.2 Balken....349
1.9.2.3 Benutzeranzeige (ab V13)....350
1.9.2.4 Benutzeranzeige, einfach....352
1.9.2.5 Benutzeranzeige, erweitert....353
1.9.2.6 Beobachtungstabelle....355
1.9.2.7 Bildfenster....357
1.9.2.8 Bildlaufleiste....361
1.9.2.9 Datum/Uhrzeit-Feld....362
1.9.2.10 Doppel-T-Stück....364

1.9.2.11 Druckauftrag/Skriptdiagnose....365
1.9.2.12 E/A-Feld....368
1.9.2.13 E/A-Feld: Verhalten bei Bedienung....370
1.9.2.14 E/A-Feld: Darstellungsformat....371
1.9.2.15 E/A-Feld: Darstellungsformat....373
1.9.2.16 Editierbares Textfeld....376
1.9.2.17 Ellipse....377
1.9.2.18 Ellipsenbogen....378
1.9.2.19 Ellipsensegment....379
1.9.2.20 f(t)-Kurvenanzeige....380
1.9.2.21 f(x)-Kurvenanzeige....384
1.9.2.22 f(x)-Kurvenanzeige....387
1.9.2.23 Grafikanzeige....391
1.9.2.24 Grafisches E/A-Feld....393
1.9.2.25 Grafisches E/A-Feld....395
1.9.2.26 GRAPH-Übersicht....396
1.9.2.27 Handrad....400
1.9.2.28 Hilfeindikator....401
1.9.2.29 HTML-Browser....402
1.9.2.30 HTML-Browser....407
1.9.2.31 Kamera-Anzeige....409
1.9.2.32 Kanaldiagnose-Anzeige....413
1.9.2.33 Kombinationsfeld....414
1.9.2.34 Kontrollkästchen....415
1.9.2.35 Kreis....416
1.9.2.36 Kreisbogen....417
1.9.2.37 Kreissegment....418
1.9.2.38 Kriterienanalyse-Anzeige....420
1.9.2.39 Kurvenanzeige....422
1.9.2.40 Ladezustand....424
1.9.2.41 Leuchtdrucktaster....425
1.9.2.42 Linie....429
1.9.2.43 Listenfeld....430
1.9.2.44 Media Player....432
1.9.2.45 Meldeanzeige....435
1.9.2.46 Meldeanzeige (ab V13)....440
1.9.2.47 Meldeanzeige, einfach....445
1.9.2.48 Meldeanzeige, erweitert....447
1.9.2.49 Meldefenster....452
1.9.2.50 Meldeindikator....455
1.9.2.51 Optionsschaltflächen....456
1.9.2.52 PDF-Anzeige....457
1.9.2.53 PLC-Code-Anzeige....463
1.9.2.54 PLC-Code-Anzeige....467
1.9.2.55 Polygon....470
1.9.2.56 Polygonzug....471
1.9.2.57 ProDiag-Übersicht....473
1.9.2.58 Rechteck....476
1.9.2.59 Rezepturanzeige....477
1.9.2.60 Rezepturanzeige (ab V13)....480
1.9.2.61 Rezepturanzeige, einfach....482
1.9.2.62 Rezepturanzeige, erweitert....483

1.9.2.63 Rundschaltfläche....486
1.9.2.64 Rohr....487
1.9.2.65 Rohrbogen....487
1.9.2.66 Schalter....488
1.9.2.67 Schaltfläche....489
1.9.2.68 Schieberegler....491
1.9.2.69 Schlüsselschalter....493
1.9.2.70 Sm@rtClient-Anzeige....497
1.9.2.71 Speicherplatzanzeige....500
1.9.2.72 Symbolbibliothek....501
1.9.2.73 Symbolisches E/A-Feld....503
1.9.2.74 Symbolisches E/A-Feld....505
1.9.2.75 System-Diagnoseanzeige....506
1.9.2.76 System-Diagnoseanzeige....508
1.9.2.77 System-Diagnoseanzeige....510
1.9.2.78 System-Diagnosefenster....513
1.9.2.79 Tabellenanzeige....515
1.9.2.80 Textfeld....519
1.9.2.81 T-Stück....520
1.9.2.82 Uhr....521
1.9.2.83 Verbinder....522
1.9.2.84 Wertetabelle....523
1.9.2.85 Wirkbereichs-Name....525
1.9.2.86 Wirkbereichs-Name (RFID)....528
1.9.2.87 Wirkbereichs-Signal....529
1.9.2.88 WLAN-Empfang....531
1.9.2.89 Zeigerinstrument....532
1.9.2.90 Zonen-Name....534
1.9.2.91 Zonen-Signal....535

2 Mit Variablen arbeiten....537

2.1 Grundlagen....537
2.1.1 Grundlagen zu Variablen....537
2.1.2 Übersicht über HMI-Variablentabellen....538
2.1.3 Externe Variablen....540
2.1.4 Externe Variablen....542
2.1.5 Adressierung externer Variablen....545
2.1.6 Interne Variablen....547
2.1.7 Interne Variablen....548

2.2 Arbeiten mit Variablen....550
2.2.1 Variablen anlegen....550
2.2.1.1 Externe Variable anlegen....550
2.2.1.2 Interne Variable anlegen....552
2.2.1.3 Mehrere Variablen anlegen....553
2.2.1.4 Datentyp einer Variable anpassen....556
2.2.2 Bearbeiten von Variablen....558
2.2.2.1 Variablen verwalten....558
2.2.2.2 Variablen bearbeiten....560
2.2.2.3 Mehrere Variablen gleichzeitig konfigurieren....561
2.2.2.4 Mehrere Variablen gleichzeitig in einem Bild verwenden....562
2.2.2.5 Einstellungen zum Synchronisieren....564
2.2.2.6 Variablen synchronisieren....566

2.2.3 Variablen konfigurieren....567
2.2.3.1 Grundlegende Variablenkonfiguration....567
2.2.3.2 Spezielle Konfigurationsmöglichkeiten....576
2.2.3.3 Spezielle Konfigurationsmöglichkeiten....591

2.3 Arbeiten mit Arrays....597
2.3.1 Grundlagen zu Arrays....597
2.3.2 Arrayvariable anlegen....600
2.3.3 Beispiele zu Arrays....602

2.4 Arbeiten mit Arrays....603
2.4.1 Grundlagen zu Arrays....603
2.4.2 Arrayvariable anlegen....605

2.5 Arbeiten mit Anwenderdatentypen....607
2.5.1 Grundlagen zu Anwenderdatentypen....607
2.5.2 Anwenderdatentyp erzeugen....609
2.5.3 Anwenderdatentyp-Elemente anlegen....610
2.5.4 Versionen von Anwenderdatentypen verwalten....611
2.5.5 Variable mit Datentyp Anwenderdatentyp anlegen....613

2.6 Arbeiten mit Zyklen....616
2.6.1 Grundlagen zu Zyklen....616
2.6.2 Zyklus definieren....617

2.7 Arbeiten mit Zyklen....619
2.7.1 Grundlagen zu Zyklen....619
2.7.2 Zyklus definieren....621
2.7.3 Startpunkt festlegen....621

2.8 Referenz....623
2.8.1 Abbildung und Codierung der Datentypen....623
2.8.2 Codierung und Wertebereich der Datentypen....625
2.8.3 Datentypen....631
2.8.3.1 Datentypen für SIMATIC S7-300/400....631
2.8.3.2 Datentypen für SIMATIC S7 1200 und S7 1500....632
2.8.3.3 Datentypen für OPC-DA....635
2.8.3.4 Datentypen für OPC-XML-DA....636
2.8.3.5 Datentypen für Allen Bradley Ethernet IP....637
2.8.3.6 Datentypen für Modicon Modbus TCP/IP....637
2.8.3.7 Datentypen für Mitsubishi MC....638

2.9 Variablen archivieren....639
2.9.1 Grundlagen zur Variablenarchivierung....639
2.9.2 Variablenarchivierung....639
2.9.2.1 Grundlagen....639
2.9.2.2 Arbeiten mit Variablenarchiven....641
2.9.3 Variablenarchivierung....658
2.9.3.1 Variablenarchivierung....658
2.9.3.2 Grundlagen....659
2.9.3.3 Arbeiten mit Variablenarchiven....677

2.10 Variablen darstellen....704
2.10.1 Variablen mit Basic Panels darstellen....704
2.10.1.1 Ausgeben von Variablenwerten in Bildern (Basic)....704
2.10.1.2 Kurvenanzeige für Werte aus der Steuerung projektieren (Basic)....704

2.10.2 Variablen mit der Runtime Advanced und Panels darstellen....706
2.10.2.1 Kurven....706
2.10.2.2 Ausgeben von Variablenwerten in Bildern....709
2.10.2.3 Kurvenanzeige für Werte aus der Steuerung projektieren....709
2.10.2.4 Kurvenanzeige für ein Variablenarchiv projektieren....712
2.10.2.5 Aufbau einer *.CSV-Datei mit Variablen....713
2.10.2.6 Direktzugriff auf die ODBC-Archivdatenbank....714
2.10.2.7 Darstellen von Bahnkurven in der Kurvenanzeige....715
2.10.3 Variablen mit Runtime Professional darstellen....718
2.10.3.1 Grundlagen....718
2.10.3.2 Variablenwerte in Kurvenform ausgeben....728
2.10.3.3 Variablenwerte in Tabellenform ausgeben....741
2.10.3.4 Wertetabelle projektieren....754
2.10.3.5 Controls in Runtime bedienen....761

3 Mit Meldungen arbeiten....785

3.1 Grundlagen....785
3.1.1 Das Meldesystem in WinCC....785
3.1.2 Das Meldesystem in WinCC....786
3.1.3 Das Meldesystem in WinCC....788
3.1.4 Meldeverfahren....790
3.1.4.1 Übersicht Meldeverfahren....790
3.1.4.2 Übersicht Meldeverfahren....791
3.1.4.3 Übersicht Meldeverfahren....792
3.1.4.4 Benutzerdefinierte Meldeverfahren....793
3.1.4.5 Systemdefinierte Meldungen....796
3.1.5 Meldezustände....798
3.1.6 Meldeklassen....799
3.1.6.1 Grundlagen zu Meldeklassen....799
3.1.6.2 Grundlagen zu Meldeklassen....801
3.1.6.3 Vordefinierte Meldeklassen....802
3.1.7 Quittierung....804
3.1.7.1 Quittieren von Meldungen....804
3.1.7.2 Quittierkonzept....806
3.1.7.3 Quittierkonzept....806
3.1.8 Meldegruppen....807
3.1.9 Meldegruppen....808
3.1.9.1 Grundlagen zu Meldegruppen....808
3.1.9.2 Hierarchische Meldegruppen....810
3.1.9.3 Variablen einer Meldegruppe....811
3.1.10 Meldenummer....812
3.1.11 Meldenummer....813

3.2 Arbeiten mit Meldungen....814
3.2.1 Bestandteile und Eigenschaften von Meldungen....814
3.2.2 Projektieren von Meldungen....816
3.2.2.1 Übersicht über das Projektieren von Meldungen....816
3.2.2.2 Übersicht über das Projektieren von Meldungen....817
3.2.2.3 Projektieren von Meldeklassen....818
3.2.2.4 Projektieren von Meldegruppen....824
3.2.2.5 Meldungen projektieren....827
3.2.2.6 Meldungen projektieren....839
3.2.2.7 Loop-In-Alarm projektieren....864

3.2.2.8 Meldungen im Editor "HMI-Variablen"....866
3.2.3 Projektieren der Ausgabe von Meldungen....869
3.2.3.1 Übersicht zum Projektieren der Ausgabe von Meldungen....869
3.2.3.2 Projektieren der Meldeanzeige....870
3.2.3.3 Projektieren der Meldeanzeige....886
3.2.3.4 Protokollieren von Meldungen....909
3.2.4 Quittierung projektieren....912
3.2.4.1 Quittierpflicht einer Meldung über Meldeklasse projektieren....912
3.2.4.2 Trigger zum Quittieren von Meldungen projektieren....913
3.2.4.3 Meldungsquittierung an die Steuerung senden....915
3.2.5 Quittierung projektieren....915
3.2.5.1 Quittieren einer Meldung projektieren....915
3.2.5.2 Quittieren einer Meldegruppe projektieren....917
3.2.5.3 Quittieren des zentralen Melders projektieren....918
3.2.5.4 Sammelquittieren von Meldungen projektieren....919
3.2.5.5 Rücksetzen des Quittierbits der Quittiervariablen projektieren....919

3.3 Archivieren von Meldungen....921
3.3.1 Projektieren der Meldearchivierung....921
3.3.1.1 Grundlagen zum Archivieren von Meldungen....921
3.3.1.2 Meldearchiv anlegen....923
3.3.1.3 Meldungen archivieren....925
3.3.1.4 Meldeanzeige für archivierte Meldungen projektieren....926
3.3.1.5 Meldepufferüberlauf projektieren....927
3.3.1.6 Direktzugriff auf die ODBC-Archivdatenbank....928
3.3.1.7 ODBC-Datenquelle einrichten....929
3.3.1.8 Prüfsumme für ein Archiv projektieren....929
3.3.1.9 Prüfsumme von Archivdaten auswerten....931
3.3.1.10 Prüfsummen für Archive in regulierten Projekten....932
3.3.1.11 Verhalten des Archivs bei Sprachumschaltung in Runtime steuern....933
3.3.1.12 Verhalten des Archivs bei Runtime-Start steuern....934
3.3.1.13 Archivierung Füllstandsabhängig steuern....935
3.3.2 Projektieren der Meldearchivierung....937
3.3.2.1 Grundlagen zum Archivieren von Meldungen....937
3.3.2.2 Meldungen archivieren....938
3.3.2.3 Meldearchiv-Backup projektieren....942
3.3.2.4 Meldearchiv-Backup verbinden und trennen....946
3.3.2.5 Direktzugriff mit C-API/ODK auf die Archivdatenbank....947
3.3.2.6 Direktzugriff mit ADO/OLE DB auf die Archivdatenbank....948
3.3.2.7 Rückladen von Meldungen nach Netzausfall projektieren....948

3.4 Bedienen von Meldungen in Runtime....950
3.4.1 Meldungen in Runtime....950
3.4.2 Meldungen in Runtime....952
3.4.3 Bedienung Meldefenster, Meldeanzeige....954
3.4.3.1 Meldefenster, Meldeanzeige in Runtime....954
3.4.3.2 Meldefenster, Meldeanzeige bedienen....956
3.4.4 Bedienung einfaches Meldefenster, Meldeanzeige....958
3.4.4.1 Einfache Meldeanzeige, Meldefenster in Runtime....958
3.4.4.2 Einfache Meldeanzeige, Meldefenster bedienen....960
3.4.5 Bedienung Meldeindikator....961
3.4.5.1 Meldeindikator in Runtime....961
3.4.5.2 Meldeindikator mit der Maus bedienen....962

3.4.6 Meldungen quittieren....962
3.4.7 Meldungen filtern....963

3.5 Bedienen von Meldungen in Runtime....965
3.5.1 Meldungen in Runtime....965
3.5.2 Meldeanzeige....968
3.5.3 Schaltflächen der Meldeanzeige....970
3.5.4 Meldungen in Runtime filtern....971
3.5.5 Meldungen in Runtime sortieren....973
3.5.6 Meldeanzeige in Runtime umschalten....974
3.5.7 Archivierte Meldungen in Runtime anzeigen....975
3.5.8 Meldungen in Runtime drucken....976
3.5.9 Sperren von Meldungen in Runtime....977
3.5.9.1 Einzelne Meldungen sperren....977
3.5.9.2 Meldungen über die Meldenummer sperren....978
3.5.9.3 Meldungen über die Meldeklasse oder Meldegruppe sperren....979
3.5.10 Quittieren von Meldungen....981
3.5.10.1 Meldungen quittieren....981
3.5.10.2 Meldungen sammelquittieren....982
3.5.10.3 Meldungen not-quittieren....982
3.5.10.4 Zentralen Melder quittieren....983

3.6 Security Ereignisse anzeigen....985
3.6.1 Security Ereignisse am Bediengerät anzeigen....985
3.6.2 Anzeige von Security Ereignissen projektieren....986

3.7 Vollständige Meldung von der Steuerung an das Bediengerät senden....987
3.7.1 Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren....987
3.7.2 Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren....989
3.7.3 Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren....991

3.8 Referenz....993
3.8.1 Systemfunktionen für Meldungen....993
3.8.2 Ereignisse für Meldungen....994
3.8.3 Ereignisse für Meldungen....995
3.8.4 Systemfunktionen für Archive....995
3.8.5 Aufbau einer *.CSV-Datei mit Meldungen....996
3.8.6 Beschreibung der Systemblöcke....997
3.8.7 SQL-Statements zum Filtern der Meldeanzeige....1000
3.8.8 Systemmeldungen....1002
3.8.8.1 Grundlagen zu Systemmeldungen....1002
3.8.8.2 9999 - Meldung Global....1003
3.8.8.3 10000 - Meldungen Drucker....1003
3.8.8.4 20000 - Meldungen Globale Skripte....1004
3.8.8.5 30000 - Meldungen Fehler bei Verwendung von Systemfunktionen....1005
3.8.8.6 40000 - Meldungen Lineare Skalierung....1005
3.8.8.7 50000 - Meldungen Datenserver....1006
3.8.8.8 60000 - Meldungen Windows-Funktionen....1006
3.8.8.9 70000 - Meldungen Windows-Funktionen....1007
3.8.8.10 80000 - Meldungen Archive....1010
3.8.8.11 90000 - Meldungen FDA....1013
3.8.8.12 110000 - Meldungen Offline Funktionen....1014

3.8.8.13 120000 - Meldungen Kurvendarstellung....1015
3.8.8.14 130000 - Meldungen Systeminformationen....1016
3.8.8.15 140000 - Meldungen Verbindung: Verbindung + Gerät....1017
3.8.8.16 160000 - Meldungen Verbindung: OPC: Verbindung....1020
3.8.8.17 170000 - Meldungen S7-Dialog....1021
3.8.8.18 180000 - Meldungen Allgemeines....1022
3.8.8.19 190000 - Meldungen Variablen....1022
3.8.8.20 190100 - Meldungen Bereichszeiger....1024
3.8.8.21 200000 - Meldungen PLC-Koordination....1024
3.8.8.22 200100 - Meldungen Bereichszeiger Projektkennung....1025
3.8.8.23 210000 - Meldungen PLC-Job....1026
3.8.8.24 220000 - Meldungen WinCC-Kommunikationstreiber....1026
3.8.8.25 230000 - Meldungen Bildobjekte....1027
3.8.8.26 240000 - Meldungen Autorisierung....1030
3.8.8.27 250000 - Meldungen S7-Status/Steuern....1030
3.8.8.28 260000 - Meldungen Passwortsystem....1031
3.8.8.29 270000 - Meldungen Meldungen System....1034
3.8.8.30 280000 - Meldungen SIMATIC 505 DP Communication Driver (CSP)....1035
3.8.8.31 290000 - Meldungen Rezeptursystem....1035
3.8.8.32 300000 - Meldungen Alarm_S....1041
3.8.8.33 310000 - Meldungen Protokollierung....1041
3.8.8.34 330000 - Meldungen Interaktion....1042
3.8.8.35 340000 - Meldungen Infotext....1042
3.8.8.36 350000 - Meldungen PROFIsafe....1042
3.8.8.37 380000 - Meldungen Sm@rtServer....1043
3.8.8.38 390000-Meldungen Kennwortverschlüsselung und Zertifikate....1044
3.8.9 Systemmeldungen....1044
3.8.9.1 Systemmeldungen....1044
3.8.9.2 1000000 - Meldungen Runtime....1045
3.8.9.3 1000800 - Meldungen WinCC-Client....1046
3.8.9.4 1000900 - Meldungen Performance Monitor....1046
3.8.9.5 1001000 - Meldungen Bilder in Runtime....1047
3.8.9.6 1002000 - Meldungen Variablenarchivierung....1047
3.8.9.7 1003018 - Meldungen Meldesystem....1048
3.8.9.8 1004000 - Meldungen Druckaufträge....1051
3.8.9.9 1005000 - Meldungen Textbibliothek....1052
3.8.9.10 1006000 - Meldungen Globale Skripte....1053
3.8.9.11 1008000 - Meldungen Benutzerverwaltung....1054
3.8.9.12 1009000 - Meldungen Lifebeat Monitoring....1054
3.8.9.13 1010000 - Meldungen STRRT....1055
3.8.9.14 1011000 - Meldungen Sammelanzeige....1055
3.8.9.15 1011101 - Meldungen Picture Tree Manager....1055
3.8.9.16 1011201 - Meldungen Split Screen Manager....1056
3.8.9.17 1012001 - Meldungen Uhrzeitsynchronisation....1056
3.8.9.18 1012200 - Meldungen Redundancy....1057
3.8.9.19 1012400 - Meldungen WebNavigatorClient....1060
3.8.9.20 1012500 - Meldungen Process Historian....1060
3.8.9.21 1012700 - Meldungen Redundancy Selbstdiagnose....1062
3.8.9.22 1016000 - Meldungen IndustrialDataBridge....1062
3.8.9.23 12508141 - Meldungen 1063
3.8.10 Bitbelegungen Variablen....1063
3.8.10.1 Variablen für Meldegruppen und Meldungen....1063

3.8.10.2 Variablen für Meldegruppen....1068

4 Mit Archiven arbeiten....1071

4.1 Arbeiten mit Archiven....1071
4.1.1 Grundlagen zu Archiven....1071
4.1.2 Eigenschaften von Archiven....1072
4.1.3 Ablageorte von Archiven....1073

4.2 Arbeiten mit Archiven....1077
4.2.1 Grundlagen zu Archiven....1077
4.2.2 Eigenschaften von Archiven....1077

5 Zugriff auf Archivdaten über WinCC OLE DB-Provider....1079

5.1 WinCC Archive Connector....1079
5.1.1 WinCC Archive Connector....1079
5.1.2 Registerkarte Konfiguration....1081
5.1.3 Registerkarte Archive verbinden/trennen....1082

5.2 Zugriff auf Archivdaten über OLE DB-Provider....1084
5.2.1 Möglichkeiten des Zugriffs auf Archivdaten....1084
5.2.2 Projektierung des Zugriffs auf Archivdaten....1084
5.2.3 Verbindung zur Archivdatenbank herstellen....1085

5.3 Anwendungsfälle....1088
5.3.1 Anwendungsfall 1: Lokaler Zugriff auf WinCC RT-Datenbank....1088
5.3.2 Anwendungsfall 2: Remote-Zugriff auf WinCC RT-Datenbank....1088
5.3.3 Anwendungsfall 3: Lokaler Zugriff auf WinCC-Archivdatenbanken....1089
5.3.4 Anwendungsfall 4: Remote-Zugriff auf WinCC-Archivdatenbanken....1090
5.3.5 Anwendungsfall 5: Lokaler Zugriff auf WinCC-Anwenderarchive....1091
5.3.6 Anwendungsfall 6: Remote-Zugriff auf WinCC-Anwenderarchive....1092

5.4 Abfragen der Archivdaten....1093
5.4.1 Darstellung der Prozesswertarchive....1093
5.4.2 Abfrage für Prozesswertarchive....1093
5.4.3 Abfrage für Meldearchive....1099
5.4.4 Darstellung der Meldearchive....1101
5.4.5 Abfrage für Anwenderarchive....1103
5.4.6 Zugriff über den Wizard "SQL Server Import/Export" projektieren....1105
5.4.7 Voraussetzungen für die Nutzung der Reporting Services schaffen....1106
5.4.8 Abfragen der Archivdaten....1108
5.4.9 Darstellung der Anwenderarchive....1109

5.5 Analysefunktionen für Meldungen und Prozesswerte....1110
5.5.1 Analysefunktionen für Meldungen und Prozesswerte....1110
5.5.2 Darstellung von Meldearchiven bei Analyseabfragen....1113

5.6 Grundlagen zu OLE DB....1115

5.7 WinCC OLE DB-Provider....1117

5.8 Installation....1119

6 Mit Rezepturen arbeiten....1121

6.1 Mit Rezepturen arbeiten....1121
6.1.1 Grundlagen....1121
6.1.1.1 Definition und Einsatzgebiete....1121

6.1.1.2 Beispiele zum Einsatz von Rezepturen....1123
6.1.1.3 Aufbau von Rezepturen....1123
6.1.1.4 Anzeige der Rezepturen....1125
6.1.1.5 Übertragen von Rezepturdatensätzen....1127
6.1.1.6 Konfiguration von Rezepturen....1130
6.1.1.7 Besonderheiten bei einigen Geräten....1132
6.1.1.8 Synchronisation von Rezepturdatensätzen mit der Steuerung....1134
6.1.1.9 Editor "Rezepturen"....1135
6.1.2 Rezepturen in Runtime....1138
6.1.2.1 Rezepturbild und Rezepturanzeige....1138
6.1.2.2 Einfache Rezepturanzeige....1142
6.1.2.3 Erweiterte Rezepturanzeige (vor V13)....1154
6.1.2.4 Erweiterte Rezepturanzeige (ab V13)....1165
6.1.2.5 Rezepturdaten exportieren und importieren....1180
6.1.3 Rezepturen projektieren....1184
6.1.3.1 Rezepturen anlegen und bearbeiten....1184
6.1.3.2 Anzeige der Rezeptur projektieren....1195
6.1.4 Beispiele....1201
6.1.4.1 Beispiel für das Erstellen einer Rezeptur....1201
6.1.4.2 Beispiel für das Projektieren eines Rezepturbildes....1203
6.1.4.3 Szenario für das Eingeben von Rezepturdatensätzen in Runtime....1205
6.1.4.4 Szenatio für einen manuellen Produktionsablauf....1206
6.1.4.5 Szenario für einen automatischen Produktionsablauf....1208

6.2 Mit Rezepturen arbeiten....1210
6.2.1 Grundlagen....1210
6.2.1.1 Definition und Einsatzgebiete....1210
6.2.1.2 Aufbau von Rezepturen....1211
6.2.1.3 Darstellung von Rezepturen in Runtime....1214
6.2.1.4 Kommunikation mit der Steuerung....1215
6.2.1.5 Editor "Rezepturen"....1215
6.2.1.6 SQL-Sprache....1218
6.2.1.7 SQL-Schlüsselwörter....1219
6.2.2 Rezepturen projektieren....1221
6.2.2.1 Allgemeine Vorgehensweise beim Projektieren....1221
6.2.2.2 Neue Rezeptur anlegen....1221
6.2.2.3 Rezepturelemente und Datensätze anlegen....1224
6.2.2.4 Rezepturen in die Projektierung importieren und wieder exportieren....1226
6.2.2.5 Vorgehen bei Änderung der Rezepturstruktur....1228
6.2.2.6 Rezepturabfrage anlegen....1229
6.2.3 Anzeige von Rezepturen projektieren....1232
6.2.3.1 Anzeige von Rezepturen....1232
6.2.3.2 Eigenschaften der Rezepturanzeige....1233
6.2.3.3 Rezepturanzeige projektieren....1235
6.2.3.4 Rezepturdaten exportieren in Runtime....1239
6.2.3.5 Persistenz projektieren....1240
6.2.4 Datenübertragung zur Steuerung....1242
6.2.4.1 Grundlagen zur Datenübertragung....1242
6.2.4.2 Rohdatenvariable....1247
6.2.5 Bedienung in Runtime....1256
6.2.5.1 Rezepturanzeige....1256
6.2.5.2 Rezepturbild bedienen....1257
6.2.5.3 Maus- und Tastatur-Bedienung....1259

6.2.5.4 Rezepturdaten bearbeiten....1262
6.2.5.5 Daten auswählen und konfigurieren....1263
6.2.5.6 Daten selektieren....1266
6.2.5.7 Daten sortieren....1268
6.2.6 Beispiele....1270
6.2.6.1 Beispiel: Rezepturen erstellen....1270
6.2.6.2 Beispiel: Rezepturabfrage erstellen....1273
6.2.6.3 Beispiel: Rezepturanzeige projektieren....1274
6.2.6.4 Verwendung von Steuervariablen....1276

7 Mit Protokollen arbeiten....1279

7.1 Grundlagen....1279
7.1.1 Protokolle....1279
7.1.2 Aufbau von Protokollen....1281
7.1.3 Aufbau von Protokollen....1283

7.2 Arbeiten mit Protokollen....1287
7.2.1 Erstellen von Protokollen....1287
7.2.1.1 Grundlagen zur Protokollerstellung....1287
7.2.1.2 Grundlagen zur Protokollerstellung....1290
7.2.1.3 Protokoll erstellen....1294
7.2.1.4 Werkzeuge zur Protokollgestaltung....1296
7.2.1.5 Fortlaufende Protokollierung von Meldungen....1300
7.2.1.6 CSV-Datei zur Ausgabe von Kurven....1301
7.2.1.7 CSV-Datei zur Ausgabe einer Tabelle....1303
7.2.1.8 Detailseiten verwalten....1306
7.2.2 Ausgeben von Protokollen....1307
7.2.2.1 Protokollausgabe in Runtime Advanced....1307
7.2.2.2 Protokollausgabe in Runtime Professional....1308
7.2.2.3 Systemprotokolle und Systemdruckaufträge....1316
7.2.3 Arbeiten mit Objekten....1320
7.2.3.1 Objekt einfügen....1320
7.2.3.2 Objekt löschen....1321
7.2.3.3 Mehrere Objekte eines Typs einfügen....1322
7.2.3.4 Objekt positionieren....1323
7.2.3.5 Objekt nach vorn oder hinten schieben (Basic, Advanced, Professional)....1325
7.2.3.6 Größe eines Objekts ändern....1326
7.2.3.7 Mehrere Objekte auswählen....1327
7.2.3.8 Position und Größe mehrerer Objekte verändern....1329
7.2.3.9 Objekte ausrichten....1330
7.2.3.10 Objekt drehen....1331
7.2.3.11 Objekt spiegeln....1333
7.2.4 Bedienung in Runtime....1334
7.2.4.1 Ausdrucken eines Protokolls....1334
7.2.4.2 Ausdrucken eines Protokolls....1334

7.3 Objekte in Protokollen....1336
7.3.1 Übersicht....1336
7.3.2 Audit-Protokoll....1336
7.3.3 Control Daten....1337
7.3.4 Control Hardcopy....1338
7.3.5 Control Informationen....1339
7.3.6 CSV-Provider Kurven....1340

7.3.7 CSV-Provider Tabellen....1341
7.3.8 Datum/Uhrzeit-Feld....1342
7.3.9 Datum/Uhrzeit-Feld....1343
7.3.10 E/A-Feld....1345
7.3.11 f(t)-Kurvenanzeige in Protokollen....1346
7.3.12 f(x)-Kurvenanzeige in Protokollen....1347
7.3.13 Grafikanzeige....1348
7.3.14 Grafisches E/A-Feld....1349
7.3.15 Hardcopy....1350
7.3.16 Meldeanzeige in Protokollen....1351
7.3.17 Meldefolgeprotokoll....1352
7.3.18 Meldeprotokoll....1353
7.3.19 ODBC-Datenbankfeld....1354
7.3.20 ODBC-Datenbanktabelle....1356
7.3.21 Projektname....1358
7.3.22 Protokollname....1359
7.3.23 Rezepturanzeige in Protokollen....1360
7.3.24 Rezepturprotokoll....1361
7.3.25 Seitenzahl....1362
7.3.26 Seitenzahl....1363
7.3.27 Symbolisches E/A-Feld....1364
7.3.28 Tabellenanzeige in Protokollen....1365
7.3.29 Textfeld....1365

8 Eine Benutzerverwaltung projektieren....1367

8.1 Grundlagen....1367
8.1.1 Einsatzgebiet der Benutzerverwaltung....1367
8.1.2 Aufbau der Benutzerverwaltung....1368
8.1.3 Benutzer....1369
8.1.4 Arbeitsbereich Benutzer....1370
8.1.5 Benutzergruppen....1371
8.1.6 Arbeitsbereich Benutzergruppen....1372
8.1.7 Einstellungen für die Benutzerverwaltung....1374
8.1.8 Einstellungen für die Benutzerverwaltung....1377
8.1.9 Einstellungen für die Benutzerverwaltung....1381
8.1.10 Zielgruppen in der Benutzerverwaltung....1382

8.2 Benutzerverwaltung aufbauen....1383
8.2.1 Benutzer für Runtime administrieren....1383
8.2.1.1 Berechtigung erstellen....1383
8.2.1.2 Benutzergruppe anlegen....1383
8.2.1.3 Benutzergruppe anlegen....1384
8.2.1.4 Berechtigung zuweisen....1385
8.2.1.5 Benutzer anlegen....1386
8.2.1.6 Benutzer einer Benutzergruppe zuordnen....1387
8.2.1.7 Benutzer verwalten....1387
8.2.1.8 Benutzergruppen verwalten....1389
8.2.1.9 Nummer der Benutzergruppe ändern....1391
8.2.1.10 Benutzer dynamisch anmelden....1391
8.2.2 Benutzer auf dem Server verwalten....1393
8.2.2.1 Zentrale Benutzerverwaltung mit SIMATIC Logon....1393
8.2.2.2 Authentifizierung und Umgang mit Zertifikaten in SIMATIC Logon....1395
8.2.2.3 Einstellungen auf dem Server vornehmen....1395

8.2.2.4 Anmelden über SIMATIC Logon....1396
8.2.2.5 Anmelden über SIMATIC Logon....1398
8.2.3 Benutzer für Web Client administrieren....1399
8.2.3.1 Benutzergruppe für Web Client administrieren....1399
8.2.3.2 Benutzer für Web Client administrieren....1400
8.2.4 Benutzer in Runtime administrieren....1402
8.2.4.1 Benutzer in Runtime....1402
8.2.4.2 Benutzeranzeige....1402
8.2.4.3 Benutzeranzeige projektieren....1404
8.2.4.4 Benutzer anlegen....1404
8.2.4.5 Benutzer mit der komplexen Benutzeranzeige verwalten....1406
8.2.4.6 Gesperrte Benutzer entsperren....1408
8.2.4.7 Sich als Benutzer anmelden....1408
8.2.4.8 Sich als Benutzer anmelden....1410
8.2.4.9 Benutzerverwaltung exportieren....1412
8.2.4.10 Benutzerverwaltung importieren....1413
8.2.5 Zugriffsschutz projektieren....1414
8.2.5.1 Zugriffsschutz....1414
8.2.5.2 Bedienberechtigung projektieren....1414

8.3 Referenz....1416
8.3.1 Objekte mit Zugriffsschutz....1416
8.3.2 Objekte mit Zugriffsschutz....1416
8.3.3 Objekte mit Zugriffsschutz....1417
8.3.4 Vordefinierte Benutzergruppen und Berechtigungen....1419
8.3.5 Vordefinierte Benutzergruppen und Berechtigungen....1419

8.4 Beispiele....1420
8.4.1 Beispiel: Schaltfläche mit Anmeldedialog projektieren....1420
8.4.2 Beispiel: Anmelden und Abmelden protokollieren....1421
8.4.3 Beispiel: Aufbau einer Benutzerverwaltung....1422
8.4.4 Beispiel: Berechtigung erstellen und projektieren....1423
8.4.5 Beispiel: Schaltfläche mit Zugriffsschutz projektieren....1424
8.4.6 Beispiel: Benutzergruppen anlegen und Berechtigungen zuweisen....1426
8.4.7 Beispiel: Benutzer anlegen und einer Benutzergruppe zuordnen....1427

9 Mit Systemfunktionen und Runtime Scripting arbeiten....1431

9.1 Grundlagen....1431
9.1.1 Runtime Scripting....1431
9.1.2 Systemfunktionen....1433
9.1.3 Benutzerdefinierte Funktionen....1435
9.1.4 Lokale Skripte....1437

9.2 Arbeiten mit Funktionslisten....1439
9.2.1 Grundlagen zur Funktionsliste....1439
9.2.2 Eigenschaften einer Funktionsliste....1440
9.2.3 Funktionsliste projektieren....1440
9.2.4 Funktionsliste bearbeiten....1442

9.3 Arbeiten mit benutzerdefinierten Funktionen....1444
9.3.1 Editor "Skripte"....1444
9.3.2 Aufruf von Systemfunktionen....1447
9.3.3 Aufruf von benutzerdefinierten Funktionen....1449
9.3.4 Zugriff auf HMI-Variablen....1450

9.3.5 Syntax von benutzerdefinierten Funktionen testen....1453

9.4 Erstellen von benutzerdefinierten VB-Funktionen....1455
9.4.1 Zugriff auf Objekte mit VBS....1455
9.4.2 Verwendung von VB-Variablen....1455
9.4.3 Benutzerdefinierte VB-Funktion erstellen....1458
9.4.4 Benutzerdefinierte VB-Funktion umbenennen....1460
9.4.5 Benutzerdefinierte VB-Funktion ausführen....1460
9.4.6 Übergabe und Rückgabe von Werten in VBS....1462

9.5 Erstellen von benutzerdefinierten C-Funktionen....1464
9.5.1 Zugriff auf Objekte mit C....1464
9.5.2 C-Header erstellen....1464
9.5.3 Verwendung globaler C-Variablen....1466
9.5.4 Funktionen aus DLLs in benutzerdefinierter C-Funktion aufrufen....1467
9.5.5 Codepage für C-Funktion einrichten....1468
9.5.6 Benutzerdefinierte C-Funktion erstellen....1469
9.5.7 Benutzerdefinierte C-Funktion umbenennen....1473
9.5.8 Benutzerdefinierte C-Funktion ausführen....1473
9.5.9 Übergabe und Rückgabe von Werten in C....1475

9.6 Schützen von benutzerdefinierten Funktionen....1477
9.6.1 Know-how-Schutz einrichten....1477
9.6.2 Know-how-geschützte benutzerdefinierte Funktionen öffnen....1477
9.6.3 Know-how-Schutz entfernen....1478
9.6.4 Passwort ändern....1479

9.7 Erstellen von lokalen Skripten....1480
9.7.1 Lokale Skripte....1480
9.7.2 Editor für lokale Skripte....1482
9.7.3 Funktionsliste in lokales Skript umwandeln....1483
9.7.4 Objekteigenschaft mit lokalem Skript dynamisieren....1484

9.8 Debuggen....1485
9.8.1 Debuggen von benutzerdefinierten VB-Funktionen....1485
9.8.2 Grundlagen des Debuggens....1486
9.8.3 Debugger aktivieren....1489
9.8.4 Aufbau von VBScript-Dateien....1490
9.8.5 Aufbau von VBScript-Dateien....1490
9.8.6 Namen im Debugger....1491
9.8.7 Haltepunkte....1492
9.8.8 Schrittweise arbeiten....1493
9.8.9 Befehle ausführen....1494
9.8.10 Variablen- oder Eigenschaftswerte ermitteln....1495
9.8.11 Diagnose....1496
9.8.11.1 Diagnose....1496
9.8.11.2 GSC-Diagnose....1497
9.8.11.3 GSC-Diagnosefenster in ein Bild einfügen....1498

9.9 Laufzeitverhalten in Runtime....1499
9.9.1 Abarbeitung von benutzerdefinierten Funktionen in Runtime....1499
9.9.2 Abarbeitung der Funktionsliste in Runtime....1500
9.9.3 Abarbeitung von Funktionen und lokalen Skripten in Runtime....1501
9.9.4 Reihenfolge bei Abarbeitung von benutzerdefinierten Funktionen und Systemfunktionen..1502
9.9.5 Dynamisieren von Objekteigenschaften in Runtime....1504

9.10 Beispiele....1505
9.10.1 Beispiele zu VBS....1505
9.10.1.1 Beispiel: Umrechnen von Fahrenheit in Grad Celsius....1505
9.10.1.2 Beispiel: Umrechnen von Inch in Meter....1507
9.10.1.3 Beispiel: Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige....1509
9.10.2 Beispiele zu VBS....1513
9.10.2.1 Übersicht....1513
9.10.2.2 Beispiel für das Zugreifen auf Objekte im Editor Bilder....1513
9.10.2.3 Beispiel für das Bestimmen von Farben von Objekten....1514
9.10.2.4 Beispiel für das Projektieren einer Sprachumschaltung....1514
9.10.2.5 Beispiel für das Deaktivieren der Runtime....1515
9.10.2.6 Beispiel für das Projektieren eines Bildwechsels über Property....1515
9.10.2.7 Beispiel für das Projektieren einer Diagnoseausgabe über Trace....1515
9.10.2.8 Beispiel für das Schreiben von Variablenwerten....1517
9.10.2.9 Beispiel für das Lesen von Variablenwerten....1519
9.10.2.10 Beispiel für das Schreiben von Objekteigenschaften....1522
9.10.2.11 Beispiel für das Projektieren der Rezepturanzeige....1524
9.10.2.12 Beispiel für das Projektieren der Meldeanzeige....1525
9.10.2.13 Beispiel für das statistische Auswerten von Variablenwerten....1527
9.10.2.14 Beispiel für das Starten eines Skripts am Server (Logging-Objekt)....1528
9.10.2.15 Beispiel für das Öffnen der PLC-Code-Anzeige aus der GRAPH-Übersicht....1529
9.10.3 Beispiele zu C....1530
9.10.3.1 Übersicht....1530
9.10.3.2 Beispiel für das Zugreifen auf Objekte im Editor Bilder....1531
9.10.3.3 Beispiel für das Bestimmen von Farben von Objekten....1531
9.10.3.4 Beispiel für das Projektieren einer Sprachumschaltung....1532
9.10.3.5 Beispiel für das Deaktivieren der Runtime....1532
9.10.3.6 Beispiel für das Projektieren eines Bildwechsels über Property....1533
9.10.3.7 Beispiel für das Projektieren einer Diagnoseausgabe über Trace....1533
9.10.3.8 Beispiel für das Schreiben von Variablenwerten....1534
9.10.3.9 Beispiel für das Lesen von Variablenwerten....1534
9.10.3.10 Beispiel für das Schreiben von Objekteigenschaften....1535

9.11 Referenz....1536
9.11.1 Funktionsliste....1536
9.11.1.1 Geräteabhängigkeit....1536
9.11.1.2 Systemfunktionen....1569
9.11.1.3 Systemfunktionen....1726
9.11.2 Ereignisse....1775
9.11.2.1 Übersicht....1775
9.11.2.2 Übersicht....1780
9.11.2.3 Übersicht....1787
9.11.2.4 Ereignisse....1799
9.11.3 WinCC Engineering V15.1 - Programmierreferenz....1832

10 Mobile Panels....1833

10.1 Grundlagen....1833
10.1.1 Einsatzgebiet Mobile Panel Wireless....1833
10.1.2 Funktionsweise des Transpondersystems....1834
10.1.3 Funktionsweise des RFID-Systems....1837
10.1.4 Zonen....1839
10.1.5 Arbeitsbereich Zonen an Transpondern....1840

10.1.6 Arbeitsbereich Zonen an Anschluss-Boxen....1841
10.1.7 Wirkbereiche....1842
10.1.8 Arbeitsbereich Wirkbereiche....1843
10.1.9 Wirkbereiche (RFID)....1845
10.1.10 Arbeitsbereich Wirkbereiche (RFID)....1846

10.2 Projektierung der Mobile Panels V2 für den fehlersicheren Betrieb (ab V12 SP1)....1848
10.2.1 Projektierungsübersicht....1848
10.2.2 PLC für den fehlersicheren Betrieb anlegen....1848
10.2.3 Mobile Panel anlegen....1849
10.2.4 Verbindung zwischen Mobile Panel und PLC herstellen....1850

10.3 Projektierung der KTP Mobile Panels für den fehlersicheren Betrieb....1852
10.3.1 KTP Mobile Panel für den fehlersicheren Betrieb anlegen....1852
10.3.2 Abbau der PROFIsafe Verbindung projektieren....1853

10.4 Arbeiten mit Zonen an Transpondern....1854
10.4.1 Zone im Transpondersystem projektieren....1854
10.4.2 Bild bei Zoneneintritt anzeigen....1855
10.4.3 Objekt abhängig von der Zone anzeigen....1856

10.5 Arbeiten mit Zonen an Anschluss-Boxen....1858
10.5.1 Zone einer Anschluss-Box projektieren....1858
10.5.2 Systemfunktionen bei der Verbindung zur Anschluss-Box aufrufen....1859

10.6 Arbeiten mit Wirkbereichen (Transponder)....1861
10.6.1 Übersicht....1861
10.6.2 Wirkbereich (Transponder) projektieren....1862
10.6.3 Wirkbereichs-Name projektieren....1863
10.6.4 Weitere Objekte der Mobile Wireless projektieren....1864

10.7 Arbeiten mit Wirkbereichen (RFID)....1865
10.7.1 Übersicht....1865
10.7.2 Wirkbereich (RFID) projektieren....1866
10.7.3 Wirkbereichs-Name projektieren (RFID)....1867

10.8 Referenz....1868
10.8.1 PROFIsafe-Adresse....1868
10.8.2 Power Management....1868
10.8.3 Zonen-ID/Anschlusspunkt-ID....1869

11 Aufgaben planen....1871

11.1 Grundlagen....1871
11.1.1 Einsatzgebiet des Aufgabenplaners....1871
11.1.2 Arbeiten mit Aufgaben und Triggern....1872
11.1.3 Arbeitsbereich des Editors Aufgabenplaner....1872
11.1.4 Trigger....1874
11.1.5 Trigger....1874
11.1.6 Timer bei zyklischen Triggern....1876
11.1.7 Variablentrigger....1876
11.1.8 Funktionsliste....1877
11.1.9 Funktionsliste....1878
11.1.10 Druckauftrag....1878
11.1.11 Anbindung von lokalen VB-Skripten....1880
11.1.12 Anbindung von lokalen C-Skripten....1881

11.2 Aufgaben planen....1883
11.2.1 Aufgabe planen mit azyklischem Trigger....1883
11.2.2 Aufgabe planen mit zyklischem Trigger....1884
11.2.3 Aufgabe planen mit Ereignis-Trigger....1886
11.2.4 Aufgabe verwalten....1888

11.3 Beispiele für Runtime Professional....1889
11.3.1 Beispiel: Täglich Startbild aktivieren....1889
11.3.2 Beispiel: Einmaliger Ausdruck eines Protokolls....1890
11.3.3 Beispiel: Lesen des aktuellen Benutzernamens....1892
11.3.4 Beispiel: Eine Variable hochzählen....1893

11.4 Beispiele für Runtime Advanced und Panels....1894
11.4.1 Beispiel: Benutzer bei Benutzerwechsel aktualisieren....1894
11.4.2 Beispiel: Wöchentlicher Ausdruck eines Protokolls....1895
11.4.3 Beispiel: Startzeitpunkt einer Aufgabe in Runtime ändern....1896
11.4.4 Beispiel: 1. Variable für Runtime projektieren....1897
11.4.5 Beispiel: 2. Datum/Uhrzeit-Feld projektieren....1898
11.4.6 Beispiel: 3. Archivierung mit Timervariable projektieren....1898

12 Diagnosefunktionen nutzen....1901

12.1 Systemdiagnose projektieren....1901
12.1.1 Systemdiagnose....1901
12.1.1.1 Grundlagen für Basic Panels....1901
12.1.1.2 Grundlagen für Panels und Runtime Advanced....1904
12.1.1.3 Objekte der Systemdiagnose projektieren....1908
12.1.2 Systemdiagnose....1917
12.1.2.1 Grundlagen für Runtime Professional....1917
12.1.2.2 Objekte der Systemdiagnose projektieren....1920
12.1.2.3 Beispiel: Systemdiagnose mit allen Objekten....1924

12.2 Überwachungsfunktionen für SIMATIC IPCs....1931

12.3 Maschinen und Anlagen mit ProDiag überwachen....1933
12.3.1 Grundlagen der Überwachung mit ProDiag....1933
12.3.2 Grundlagen der Überwachung von Operanden....1933
12.3.3 Voraussetzungen und Lizenzierung....1935
12.3.4 Überwachungen visualisieren....1936
12.3.4.1 Objekte zur Überwachung und Diagnose von Anlagen....1936
12.3.4.2 Zusammenwirkung der Objekte zur Überwachung und Diagnose....1937
12.3.4.3 Zusammenwirkung der Objekte zur Überwachung und Diagnose....1940
12.3.4.4 ProDiag-Übersicht projektieren....1943
12.3.4.5 GRAPH-Übersicht projektieren....1948
12.3.4.6 PLC-Code-Anzeige projektieren....1954
12.3.4.7 PLC-Code-Anzeige projektieren....1968
12.3.4.8 Darstellungsprobleme in der PLC-Code-Anzeige beheben....1983
12.3.5 Beispiel: Visualisierung der ProDiag-Funktionalität....1984
12.3.5.1 Handlungsübersicht....1984
12.3.5.2 Voraussetzungen....1986
12.3.5.3 Bilder erstellen....1987
12.3.5.4 Diagnoseobjekte einfügen....1987
12.3.5.5 Diagnoseobjekte konfigurieren....1988
12.3.5.6 Meldeanzeige für ProDiag konfigurieren....1990
12.3.5.7 Einsprung von der GRAPH-Übersicht projektieren....1992

12.3.5.8 Einsprung von der ProDiag-Übersicht projektieren....1995
12.3.6 Erstwerterfassung und Kriterienanalyse....1996
12.3.6.1 Überblick über die Erstwerterfassung und Kriterienanalyse....1996
12.3.6.2 Unterstützte Anweisungen....1999
12.3.6.3 Beispiel für die Reihenfolge der Kriterienanalyse....2001
12.3.6.4 Kriterienanalyse-Anzeige projektieren....2003
12.3.6.5 Kriterienanalyse-Anzeige projektieren....2004
12.3.6.6 Meldungen mit Kriterien ausgeben....2005

13 Mit Steuerungen kommunizieren....2009

13.1 WinCC Engineering V15.1 - Kommunikation....2009

14 Übergreifende Funktionen nutzen....2011

14.1 Mit dem Bediengeräte-Assistent arbeiten....2011
14.1.1 Grundlagen zum Bediengeräte-Assistenten....2011
14.1.2 Grundlagen zum Bediengeräte-Assistenten....2014

14.2 Arbeiten mit Bibliotheken....2016
14.2.1 Grundlagen zu Bibliotheken....2016
14.2.2 Übersicht über die Bibliotheksansicht....2018
14.2.3 Kopiervorlagen und Typen....2020
14.2.4 Bibliotheken in WinCC....2021
14.2.5 Bibliotheken verwalten....2023
14.2.5.1 Übersicht über die Bibliotheksverwaltung....2023
14.2.5.2 Bibliotheksverwaltung öffnen....2024
14.2.5.3 Typen in der Bibliotheksverwaltung filtern....2025
14.2.5.4 Globale Bibliothek erstellen....2026
14.2.5.5 Globale Bibliothek speichern....2027
14.2.5.6 Globale Bibliothek öffnen....2028
14.2.5.7 Protokolle zu globalen Bibliotheken anzeigen....2029
14.2.5.8 Projekt mit den Inhalten einer Bibliothek aktualisieren....2029
14.2.5.9 Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren....2030
14.2.5.10 Typen einer Bibliothek hochrüsten....2031
14.2.5.11 Bibliothekstexte exportieren und importieren....2033
14.2.6 Objekte in einer Bibliothek verwalten....2035
14.2.6.1 Bibliotheksobjekte anzeigen....2035
14.2.6.2 Objekt in einer Bibliothek hinterlegen....2038
14.2.6.3 Bibliotheksobjekt einfügen....2039
14.2.7 Typen und ihre Versionen verwenden....2040
14.2.7.1 Zustand von Typ-Versionen....2040
14.2.7.2 Skript als Typ erzeugen....2041
14.2.7.3 Bild als Typ erzeugen....2042
14.2.7.4 Stil als Typ erzeugen....2042
14.2.7.5 Formatvorlage als Typ erzeugen....2043
14.2.7.6 Neue Version eines Typen erstellen....2043

14.3 Projektdaten importieren und exportieren....2045
14.3.1 Datenaustausch zwischen Projekten und externen Anwendungen....2045
14.3.2 Rezepturdatensätze importieren und exportieren....2046
14.3.2.1 Rezepturdatensätze exportieren....2046
14.3.2.2 Rezepturdatensätze importieren....2048
14.3.2.3 Rezepturdatensätze exportieren....2050
14.3.2.4 Rezepturdatensätze importieren....2051

14.3.2.5 Bearbeitung von exportierten Rezepturdatensätzen....2052
14.3.2.6 Format der Daten von Rezepturdatensätze....2053
14.3.3 Meldungen importieren und exportieren....2054
14.3.3.1 Meldungen exportieren....2054
14.3.3.2 Meldungen importieren....2055
14.3.3.3 Meldungen importieren....2057
14.3.3.4 Format der Daten von Analogmeldungen....2058
14.3.3.5 Format der Daten von Bitmeldungen....2061
14.3.3.6 Format der Daten von Analogmeldungen....2063
14.3.3.7 Format der Daten von Bitmeldungen....2067
14.3.4 Variablen importieren und exportieren....2069
14.3.4.1 Variablen exportieren....2069
14.3.4.2 Variablen importieren (ab WinCC V13 SP1)....2070
14.3.4.3 Variablen importieren (vor WinCC V13 SP1)....2072
14.3.4.4 Format der Variablendaten....2073
14.3.4.5 Format der Variablendaten....2076
14.3.5 Textlisten importieren und exportieren....2078
14.3.5.1 Textlisten exportieren....2078
14.3.5.2 Textlisten importieren....2079
14.3.5.3 Format der Daten für Textlisten....2080
14.3.6 Projekttexte importieren und exportieren....2083
14.3.6.1 Projekttexte importieren....2083
14.3.6.2 Projekttexte exportieren....2083

14.4 Querverweise nutzen....2086
14.4.1 Allgemeine Informationen zu Querverweisen....2086
14.4.2 Besonderheiten beim Arbeiten mit Querverweisliste in WinCC....2086
14.4.3 Textuelle Querverweise....2087
14.4.4 Ungültige Querverweise....2089
14.4.5 Variablen in den Bildern umverdrahten....2091

14.5 Sprachen verwalten....2095
14.5.1 Sprachbegriffe in WinCC....2095
14.5.2 Einstellungen für Sprachen im Betriebssystem....2097
14.5.3 Einstellungen im Betriebssystem für asiatische Sprachen....2097
14.5.4 Projektsprachen einstellen....2098
14.5.4.1 Oberflächensprache wählen....2098
14.5.4.2 Projektsprachen aktivieren....2099
14.5.4.3 Referenzsprache und Editiersprache wählen....2100
14.5.5 Ein Projekt in mehreren Sprachen erstellen....2102
14.5.5.1 Arbeiten mit mehreren Sprachen....2102
14.5.5.2 Grundlagen zu Projekttexten....2103
14.5.5.3 Texte direkt übersetzen....2104
14.5.5.4 Texte mithilfe von Referenztexten übersetzen....2106
14.5.5.5 Projekttexte exportieren....2106
14.5.5.6 Projekttexte importieren....2108
14.5.6 Sprachabhängige Grafiken einsetzen....2109
14.5.6.1 Editor "Grafiksammlung"....2109
14.5.6.2 Grafik in der Grafiksammlung hinterlegen....2110
14.5.6.3 Externe Grafik in der Grafiksammlung hinterlegen....2112
14.5.7 Sprachen in Runtime....2114
14.5.7.1 Sprachen und Schriftarten in Runtime....2114
14.5.7.2 Methoden der Sprachumschaltung....2115

14.5.7.3 Runtime-Sprache aktivieren....2116
14.5.7.4 Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen....2117
14.5.7.5 Standardschrift für eine Runtime-Sprache einstellen....2119
14.5.7.6 Schriftart für alle Sprachen vereinheitlichen....2120
14.5.7.7 Archivsprache wählen....2120
14.5.7.8 Besonderheiten asiatischer und östlicher Sprachen in Runtime....2121
14.5.8 Beispiel zur mehrsprachigen Projektierung....2122
14.5.8.1 Beispiel: Schaltfläche zur Sprachumschaltung projektieren....2122
14.5.8.2 Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren....2123
14.5.8.3 Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren....2124

14.6 Gerät tauschen....2126
14.6.1 Grundlagen....2126
14.6.2 Projektierung der Verbindungen anpassen....2127
14.6.3 Geräteabhängige Funktionen....2128
14.6.3.1 Geräteabhängige Funktionen....2128
14.6.3.2 Tastenbelegung beim Tausch eines Geräts....2129
14.6.4 Bilder an das neue Gerät anpassen....2131
14.6.4.1 Möglichkeiten der Bildanpassung....2131
14.6.4.2 Objekte an Inhalt anpassen....2133
14.6.4.3 Position der Bildobjekte festlegen....2135
14.6.5 Beispiel: Gerät tauschen....2136
14.6.5.1 Beispiel: Handlungsübersicht....2136
14.6.5.2 Beispiel: Bilder anpassen....2136
14.6.5.3 Beispiel: Gerät tauschen....2137

14.7 Zwischen Bediengeräten und Editoren kopieren....2140
14.7.1 Grundlagen....2140
14.7.1.1 Grundlagen....2140
14.7.1.2 Möglichkeiten für das Kopieren und Einfügen....2141
14.7.1.3 Nicht unterstützte Objekte und Funktionalitäten (Professional)....2142
14.7.1.4 Nicht unterstützte Objekte und Funktionalitäten....2142
14.7.2 Kopieren und Einfügen....2144
14.7.2.1 Kopieren von Objekten mit verknüpften Objekten....2144
14.7.2.2 Automatisch kopierte verknüpfte Objekte....2144
14.7.2.3 Automatisch kopierte verknüpfte Objekte....2145
14.7.2.4 Kopieren von Bediengeräten mit HMI-Verbindungen....2145
14.7.2.5 Mit Anwenderdatentypen verknüpfte Bildobjekte kopieren....2146
14.7.2.6 Drag&Drop aus der Detailansicht....2146
14.7.2.7 Kopieren von Bildern....2147
14.7.2.8 Kopieren von Rezepturen innerhalb eines Bediengeräts....2148
14.7.2.9 Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional....2149
14.7.3 Kopieren und Einfügen zwischen Runtime Advanced und Panels und Runtime Professional....2150
14.7.3.1 Grundlagen....2150
14.7.3.2 Systemfunktionen....2151
14.7.3.3 Einschränkungen beim Kopieren....2152
14.7.3.4 Objekte in Bildern....2152
14.7.3.5 Bildobjekte....2153
14.7.3.6 Verbindungen....2155
14.7.3.7 Kopieren von Variablen....2156
14.7.3.8 Datentypen und Eigenschaften für Variablen....2157
14.7.3.9 Strukturen....2158

14.7.3.10 Meldungen....2161
14.7.3.11 Archive....2166
14.7.3.12 Kopieren von VB-Skripten....2169
14.7.3.13 Kopieren von Rezepturen zwischen Runtime Advanced und Panels und Runtime Professional....2170
14.7.3.14 Kopieren von Zyklen....2171
14.7.3.15 Protokolle....2172
14.7.3.16 Text- und Grafiklisten....2174
14.7.3.17 Benutzerverwaltung....2175
14.7.3.18 Aufgabenplaner....2176
14.7.4 Kopieren zwischen unterschiedlichen RT- und ES-Versionen....2177

14.8 WinCC-Versionskompatibilität nutzen....2180
14.8.1 Grundlagen zur Versionskompatibilität....2180
14.8.2 Grundlagen zur Versionskompatibilität....2182
14.8.3 Projekt hochrüsten....2183
14.8.4 Globale Bibliothek hochrüsten....2184
14.8.5 Projekt einer WinCC-Vorgängerversion bearbeiten....2186
14.8.6 Wechsel zwischen Geräteversionen....2187
14.8.7 Geräteversion ändern....2188

14.9 Farben zentral verwalten....2189
14.9.1 Grundlagen zur Farbverwaltung....2189
14.9.2 Farben suchen und ersetzen....2189

14.10 Dateien über den Editor "HMI Dateien" laden....2193

14.11 Austausch von Steuerungsdaten aus anderen Projekten....2196

14.12 Projektstatus am SIMATIC WinCC RT Symbol erkennen....2197

15 Übersetzen und Laden....2199

15.1 Herstellen der Verbindung zum Bediengerät....2199

15.2 Basic Panels....2201
15.2.1 Einstellungen für Runtime....2201
15.2.1.1 Einstellungen für Runtime....2201
15.2.1.2 Darstellung in Runtime projektieren....2203
15.2.1.3 Bedienung in Runtime projektieren....2204
15.2.1.4 Kompatibilitätsprüfung mit der Steuerung einstellen....2206
15.2.1.5 Kennwortschutz in Runtime....2206
15.2.2 Übersicht zum Übersetzen und Laden von Projekten....2208
15.2.3 Projekt übersetzen....2210
15.2.4 Simulieren von Projekten....2211
15.2.4.1 Grundlagen zur Simulation....2211
15.2.4.2 Projekt simulieren....2212
15.2.4.3 Arbeiten mit dem Variablen-Simulator....2214
15.2.4.4 Einschränkungen bei der Simulation....2216
15.2.5 Laden von Projekten....2217
15.2.5.1 Übersicht zum Laden von Projekten....2217
15.2.5.2 Projekt laden....2218
15.2.5.3 Projekt von externem Speichermedium laden....2221
15.2.6 Übersetzen und Laden mit Mulltiuser Engineering....2223
15.2.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)....2223
15.2.6.2 Übersetzen in der Serverprojekt-Ansicht....2224

15.2.6.3 Übersetzen in der lokalen Session....2225
15.2.7 Runtime starten....2226
15.2.7.1 Runtime auf dem Bediengerät starten....2226
15.2.8 Fehlermeldungen beim Laden von Projekten (Basic)....2227
15.2.9 Anpassen des Projekts für ein anderes Bediengerät....2229
15.2.10 Projektgröße reduzieren....2231
15.2.11 Grundlagen zur Bedienung in Runtime....2232
15.2.11.1 Übersicht....2232
15.2.11.2 Bedienung über Touchscreen....2233
15.2.11.3 Bedienung über Tasten....2237
15.2.11.4 Navigieren im Display (BS)....2241
15.2.11.5 Aktion auslösen....2242
15.2.11.6 Wert eingeben....2243
15.2.11.7 Bedienelemente bewegen....2243
15.2.11.8 Hilfetext anzeigen....2244
15.2.11.9 Runtime-Sprache umstellen....2245
15.2.12 Barcodes mit optischen Handlesegeräten eingeben....2246
15.2.13 Sicherheit auf dem Bediengerät....2247
15.2.13.1 Grundlagen zu Signaturen....2247
15.2.13.2 Überprüfung der Signatur ausschalten....2248

15.3 Runtime Advanced und Panels....2249
15.3.1 Einstellungen für Runtime....2249
15.3.1.1 Einstellungen für Runtime....2249
15.3.1.2 Darstellung in Runtime projektieren....2251
15.3.1.3 Bedienung in Runtime projektieren....2253
15.3.1.4 Kompatibilitätsprüfung mit der Steuerung einstellen....2255
15.3.1.5 Uhrzeitbasis einstellen....2255
15.3.1.6 Drucken in Runtime....2256
15.3.1.7 Kennwortschutz in Runtime....2256
15.3.2 Übersicht zum Übersetzen und Laden von Projekten....2258
15.3.3 Projekt übersetzen....2260
15.3.4 Simulieren von Projekten....2261
15.3.4.1 Grundlagen zur Simulation....2261
15.3.4.2 Simulation von Runtime Advanced....2262
15.3.4.3 Projekt simulieren....2263
15.3.4.4 Bild simulieren....2265
15.3.4.5 Arbeiten mit dem Variablen-Simulator....2265
15.3.4.6 Einschränkungen bei der Simulation....2267
15.3.5 Laden von Projekten....2268
15.3.5.1 Übersicht zum Laden von Projekten....2268
15.3.5.2 Übersicht zum Laden von Projekten....2271
15.3.5.3 Projekt laden....2273
15.3.5.4 "Pack&Go"-Datei erzeugen....2277
15.3.5.5 Projekt aus "Pack&Go"-Datei auf Bediengerät laden....2278
15.3.5.6 Laden über die USB-Schnittstelle....2279
15.3.5.7 Projekt von externem Speichermedium laden....2280
15.3.5.8 USB-Treiber unter Windows 7 installieren....2284
15.3.6 Übersetzen und Laden mit Multiuser Engineering....2285
15.3.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)....2285
15.3.6.2 Übersetzen in der Serverprojekt-Ansicht....2287
15.3.6.3 Übersetzen in der lokalen Session....2288
15.3.7 Runtime starten....2289

15.3.7.1 Runtime auf dem Projektierungs-PC starten....2289
15.3.7.2 Runtime Advanced und Panel Runtime starten....2290
15.3.8 Fehlermeldungen beim Laden von Projekten....2291
15.3.9 Anpassen des Projekts für ein anderes Bediengerät....2293
15.3.10 Projektgröße reduzieren....2295
15.3.11 Unterstützte Netzwerkkameras und Einstellungen....2296
15.3.12 Daten einer Speicherkarte anzeigen....2300
15.3.12.1 Grundlagen....2300
15.3.12.2 Arbeiten mit Backups....2300
15.3.13 Arbeiten mit Bediengeräte-Images....2303
15.3.13.1 Bediengeräte-Image anzeigen....2303
15.3.13.2 Bediengeräte-Image löschen....2304
15.3.13.3 Bediengeräte-Image auf der Speicherkarte erstellen....2305
15.3.13.4 Bediengeräte-Images ab V15....2307
15.3.14 Grundlagen zur Bedienung in Runtime....2309
15.3.14.1 Übersicht....2309
15.3.14.2 Bedienung über Touchscreen....2309
15.3.14.3 Bedienung über Tasten....2319
15.3.14.4 Navigieren im Display (BS)....2323
15.3.14.5 Aktion auslösen....2324
15.3.14.6 Wert eingeben....2325
15.3.14.7 Bedienelemente bewegen....2325
15.3.14.8 Hilfetext anzeigen....2326
15.3.14.9 Runtime-Sprache umstellen....2327
15.3.14.10 Webbrowser der WebKit-Engine: Übersicht....2328
15.3.15 Barcodes mit optischen Handlesegeräten eingeben....2333
15.3.16 Sicherheit auf dem Bediengerät....2335
15.3.16.1 Grundlagen zu Signaturen....2335
15.3.16.2 Herunterladen von Optionen der Drittanbieter....2335
15.3.16.3 Überprüfung der Signatur ausschalten....2336

15.4 Runtime Professional....2337
15.4.1 Einstellungen für Runtime....2337
15.4.1.1 Einstellungen der Runtime-Software....2337
15.4.1.2 Darstellung in Runtime projektieren (Professional)....2339
15.4.1.3 Bedienung in Runtime projektieren....2341
15.4.1.4 Startsequenz einrichten....2349
15.4.1.5 Uhrzeitsynchronisation aktivieren....2350
15.4.1.6 Uhrzeitbasis einstellen....2355
15.4.1.7 Einstellung der Persistenz....2355
15.4.1.8 Mit Service-Mode arbeiten....2357
15.4.1.9 Kennwortschutz in Runtime....2361
15.4.2 Übersicht zum Übersetzen und Laden von Projekten....2363
15.4.3 Projekt übersetzen....2365
15.4.4 Simulieren von Projekten....2366
15.4.4.1 Grundlagen zur Simulation....2366
15.4.4.2 Simulation von Runtime Professional....2367
15.4.4.3 Projekt simulieren....2368
15.4.4.4 Bild simulieren....2369
15.4.4.5 Arbeiten mit dem Variablen-Simulator....2370
15.4.4.6 Einschränkungen bei der Simulation....2371
15.4.5 Laden von Projekten....2371
15.4.5.1 Übersicht zum Laden von Projekten....2371

15.4.5.2 Übersicht zum Laden von Projekten....2374
15.4.5.3 Übersicht zum Laden von Projekten....2376
15.4.5.4 Projekt auf ein Bediengerät laden....2378
15.4.5.5 Projekt in ein Dateisystem laden....2381
15.4.5.6 Online-Deltaladen für Runtime Professional....2382
15.4.5.7 Runtime-Projekt auf einem Bediengerät auf V13 SP1 aktualisieren....2386
15.4.6 Übersetzen und Laden mit Multiuser Engineering....2386
15.4.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)....2386
15.4.6.2 Übersetzen in der Serverprojekt-Ansicht....2388
15.4.6.3 Übersetzen in der lokalen Session....2389
15.4.6.4 Regeln zum Online-Deltaladen....2390
15.4.6.5 Online-Deltaladen aus der Serverprojekt-Ansicht....2392
15.4.6.6 Online-Deltaladen aus der lokalen Session....2393
15.4.7 Runtime starten....2395
15.4.7.1 Runtime auf dem Projektierungs-PC starten....2395
15.4.7.2 Runtime Professional starten....2397
15.4.8 Grundlagen zur Bedienung in Runtime....2399
15.4.8.1 Übersicht zur Bedienung in Runtime (Professional)....2399
15.4.8.2 Bedienung über Touchscreen....2400
15.4.8.3 Bedienung über Tasten....2408
15.4.8.4 Navigieren im Display (BS)....2412
15.4.8.5 Aktion auslösen....2413
15.4.8.6 Wert eingeben....2413
15.4.8.7 Bedienelemente bewegen....2414
15.4.8.8 Hilfetext anzeigen (Professional)....2415
15.4.8.9 Runtime-Sprache umstellen....2415
15.4.8.10 Runtime-Daten exportieren....2416

15.5 In die Projektierung springen....2417
15.5.1 In die Projektierung springen....2417
15.5.2 Lesemodus im TIA Portal....2418

15.6 In die Projektierung springen....2420
15.6.1 In die Projektierung springen....2420
15.6.2 Lesemodus im TIA Portal....2421
15.6.3 Lesemodus projektieren....2423
15.6.4 Beispiel: Lesemodus mithilfe der API-Funktion projektieren....2424

15.7 Wartung des Bediengeräts....2426
15.7.1 Übersicht zur Wartung des Bediengeräts (Basic Panels)....2426
15.7.2 Übersicht zur Wartung des Bediengeräts (Panels)....2427
15.7.3 Übersicht zur Wartung des Bediengeräts (RT Adv, RT Prof)....2428
15.7.4 ProSave....2429
15.7.5 Datensicherung des Bediengeräts....2429
15.7.6 Daten des Bediengeräts sichern und wiederherstellen....2431
15.7.7 Aktualisierung des Betriebssystems....2432
15.7.8 Betriebssystem des Bediengeräts aktualisieren....2434
15.7.9 Betriebssystem des Bediengeräts von einem Datenträger aktualisieren....2435
15.7.10 Übertragung von License Keys....2437
15.7.11 Lizenzen verwalten....2438
15.7.12 Option installieren und deinstallieren....2440
15.7.13 Asiatische Schriftarten auf das Bediengerät laden....2441

15.8 Druckertreiber....2442

15.8.1 Gültigkeit....2442
15.8.2 Unterstützte Bediengeräte....2442
15.8.3 Druckertreiber einrichten....2443
15.8.3.1 Postscript....2443
15.8.3.2 HTML....2444
15.8.3.3 PDF....2445
15.8.4 Installation....2446
15.8.5 Transfer der Optionen....2446

16 Leistungsmerkmale....2449

16.1 Allgemeine technische Daten....2449
16.1.1 Verwendbare Zeichen....2449
16.1.2 Empfohlene Drucker....2450
16.1.3 Drucken über Druckserver....2451
16.1.4 Speicherplatzbedarf von Rezepturen....2452
16.1.5 Speicherplatzbedarf von Rezepturen für Basic Panels....2453

16.2 Leistungsmerkmale der Geräte....2455
16.2.1 Basic Panel....2455
16.2.2 Basic Panel 2nd Generation....2458
16.2.3 Mobile Panel....2462
16.2.4 Comfort Panel....2467
16.2.5 WinCC Runtime Advanced....2473
16.2.6 WinCC Runtime Professional....2478
16.2.7 Orientierungshilfe Performance in Runtime (WinCC Runtime Professional)....2482
16.2.8 Orientierungshilfe Kommunikation in Runtime (WinCC Runtime Professional)....2484

17 Optionen....2489

17.1 WinCC Engineering V15.1 - Optionen....2489

18 Umstieg auf WinCC im TIA Portal....2491

18.1 Übersicht....2491

18.2 WinCC V7....2492
18.2.1 Aktionen....2492
18.2.2 Anwender-Objekte....2493
18.2.3 Benutzerverwaltung....2495
18.2.4 Direktverbindung....2498
18.2.5 Dynamic Wizard....2499
18.2.6 Dynamik Dialog....2501
18.2.7 Effizientes Arbeiten....2503
18.2.8 Strukturen und Strukturvariablen....2505

18.3 WinCC flexible....2508
18.3.1 Bibliotheken....2508
18.3.2 Bilder und Vorlagen....2510
18.3.3 Skripte in Bildbausteinen....2512
18.3.4 Synchronisierung von Rezepturen....2513
18.3.5 Besonderheiten beim Umstieg....2514

Index....2517

Bilder erstellen

1

1.1 Grundlagen

1.1.1 Grundlagen zu Bildern

Einleitung

Mit WinCC erstellen Sie Bilder zum Bedienen und Beobachten von Maschinen und Anlagen. Um Bilder zu erstellen, stehen Ihnen vordefinierte Objekte zur Verfügung, mit denen Sie Ihre Anlage nachbilden, Prozessabläufe anzeigen und Prozesswerte vorgeben.

Anwendungsbeispiel

Diese Abbildung zeigt ein Bild, das mit WinCC erstellt wurde. Die Anlagenbediener beobachten und bedienen mit diesem Bild eine Mischstation zur Herstellung verschiedener Fruchtsäfte. Aus verschiedenen Tanks werden Fruchtsaft-Anteile in einen Mischer gefüllt und gemischt. Angezeigt wird die Füllhöhe in den Tanks.

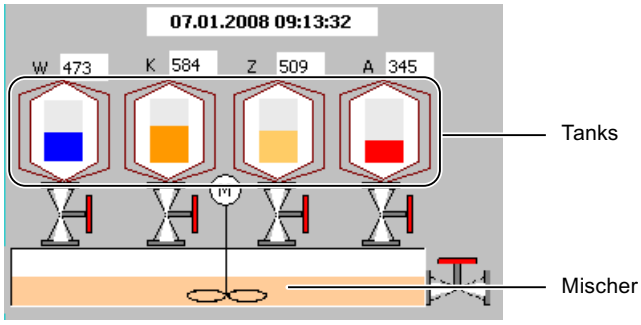

Aufbau eines Bildes

Sie fügen die Objekte in Ihr Bild ein, die Sie für die Darstellung des Prozesses benötigen. Sie konfigurieren die Objekte entsprechend den Anforderungen Ihres Prozesses.

Ein Bild kann aus statischen und dynamischen Elementen bestehen.

- Statische Elemente, z. B. Text und Grafik, ändern sich in Runtime nicht. Im Beispiel der Mischanlage sind z. B. die Beschriftungen der Tanks (W, K, Z, A) statisch.
- Dynamische Elemente ändern sich abhängig vom Prozess. Sie visualisieren aktuelle Prozesswerte wie folgt:
  - Aus dem Speicher der Steuerung
  - Aus dem Speicher des Bediengeräts in Form alphanumerischer Anzeigen, Kurven und Balken.

  Zu den dynamischen Objekten gehören auch Eingabefelder am Bediengerät. Im Beispiel der Mischanlage sind die Füllstände der Tanks dynamische Objekte.

Prozesswerte und Eingaben werden zwischen Steuerung und Bediengerät über Variablen ausgetauscht.

Bildeigenschaften

Die Darstellung des Bildes ist abhängig vom Bediengerät, für das Sie projektieren. Die Darstellung entspricht dem Aussehen der Bedienoberfläche des Geräts. Wenn das eingestellte Bediengerät Funktionstasten besitzt, zeigt das Bild diese Funktionstasten. Andere Eigenschaften, wie die Bildauflösung, die verfügbaren Schriftarten und Farben hängen ebenfalls vom eingestellten Bediengerät ab.

Funktionstasten

Eine Funktionstaste ist eine Taste am Bediengerät, die Sie in WinCC mit einer oder mehreren Funktionen belegen können. Die Funktionen werden getriggert, sobald der Bediener die Taste am Bediengerät drückt.

Eine Funktionstaste belegen Sie global oder lokal:

Funktionstasten mit globaler Belegung lösen stets dieselbe Aktion aus, unabhängig vom aktuell angezeigten Bild.

Funktionstasten mit lokaler Belegung lösen abhängig vom angezeigten Bild am Bediengerät unterschiedliche Aktionen aus. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.

Navigation

Damit der Bediener ein Bild in Runtime am Bediengerät aufrufen kann, projektieren Sie eine Bildnavigation.

- Sie nutzen den Editor "Bilder", um in den Bildern Schaltflächen zu projektieren, mit denen andere Bilder aufgerufen werden.
- Sie nutzen den Editor "Globales Bild", um global belegte Funktionstasten zu projektieren.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)
Grundlagen zu Bildern (Seite 35)
Geräteabhängiger Funktionsumfang von Bildern (Seite 36)
Task Cards (Seite 38)
Arbeitsschritte (Seite 41)
Grundlagen zum Arbeiten mit Vorlagen (Seite 49)
Grundlagen zum Arbeiten mit Ebenen (Seite 263)
Übersicht über die Objekte (Seite 91)
Grundlagen zu Textlisten (Seite 180)
Grundlagen zur Dynamisierung von Bildern (Seite 210)
Grundlagen zu Bibliotheken (Seite 2016)
Grundlage zu Bildbausteinen (Seite 273)
Druckauftrag/Skriptdiagnose (Seite 365)
Grundlagen zur Bildnavigation (Seite 312)

1.1.2 Grundlagen zu Bildern

Einleitung

Mit WinCC erstellen Sie Bilder zum Bedienen und Beobachten von Maschinen und Anlagen. Um Bilder zu erstellen, stehen Ihnen vordefinierte Objekte zur Verfügung, mit denen Sie Ihre Anlage nachbilden, Prozessabläufe anzeigen und Prozesswerte vorgeben.

Anwendungsbeispiel

Diese Abbildung zeigt ein Bild, das mit WinCC erstellt wurde. Die Anlagenbediener beobachten und bedienen mit diesem Bild eine Mischstation zur Herstellung verschiedener Fruchtsäfte. Aus verschiedenen Tanks werden Fruchtsaft-Anteile in einen Mischer gefüllt und gemischt. Angezeigt wird hier die Füllhöhe in den Tanks.

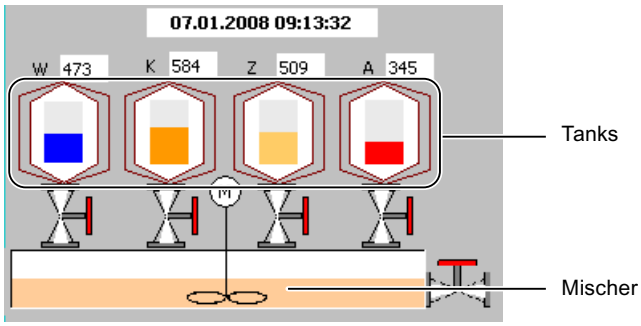

Aufbau eines Bildes

Sie fügen die Objekte in Ihr Bild ein, die Sie für die Darstellung des Prozesses benötigen. Sie konfigurieren die Objekte entsprechend den Anforderungen Ihres Prozesses.

Ein Bild kann aus statischen und dynamischen Elementen bestehen:

- Statische Elemente, z. B. im Bild oben Text und Grafik, ändern sich in Runtime nicht. Im Beispiel der Mischanlage sind z. B. die Beschriftungen der Tanks (W, K, Z, A) statisch.
- Dynamische Elemente ändern sich abhängig vom Prozess. Sie visualisieren aktuelle Prozesswerte wie folgt:
  - Aus dem Speicher der Steuerung
  - Aus dem Speicher des Bediengeräts in Form alphanumerischer Anzeigen, Kurven und Balken

  Zu den dynamischen Objekten gehören auch Eingabefelder am Bediengerät. Im Beispiel der Mischanlage sind die Füllstände der Tanks dynamische Objekte.

Prozesswerte und Eingaben werden zwischen Steuerung und Bediengerät über Variablen ausgetauscht.

Navigation

Damit der Bediener ein Bild in Runtime am Bediengerät aufrufen kann, projektieren Sie eine Bildnavigation.

- Sie nutzen den Editor "Menüs & Symbolleisten", um die Navigation zwischen den Bildern zu projektieren.
- Sie nutzen den Editor "Bilder", um in den Bildern Schaltflächen zu projektieren, mit denen andere Bilder aufgerufen werden.

Siehe auch

Grundlagen zu Bildern (Seite 33)

Geräteabhängiger Funktionsumfang von Bildern (Seite 36)

1.1.3 Geräteabhängiger Funktionsumfang von Bildern

Einleitung

Die Funktionen eines Bediengeräts bestimmen die Darstellung des Geräts in WinCC und den Funktionsumfang der Editoren.

Folgende Eigenschaften von Bildern sind abhängig vom eingestellten Bediengerät:

- Gerätelayout
- Bildauflösung
- Anzahl der Farben

- Schriftarten
- Verwendbare Objekte

Gerätelayout

Das Gerätelayout eines Bildes bildet das Bediengerät ab, für das Sie projektieren. Wenn das Bediengerät z. B. Funktionstasten besitzt, erscheinen diese im Gerätelayout des Bildes.

Bildauflösung

Da unterschiedliche Bediengeräte unterschiedliche Displaygrößen besitzen, hängt die Auflösung eines Bildes vom eingestellten Bediengerät ab. Die Bildauflösung können Sie nur verändern, wenn Sie das Bediengerät "WinCC Runtime Advanced" bzw. "WinCC Runtime Professional" projektieren.

Anzahl der Farben

Den Objekten eines Bildes können Sie Farben zuordnen. Die Anzahl der möglichen Farben hängt davon ab, wie viele und welche Farben das eingestellte Bediengerät unterstützt.

Schriftarten

In allen Bildobjekten, die statischen oder dynamischen Text enthalten, können Sie das Erscheinungsbild der Texte anpassen. So heben Sie einzelner Texte in einem Bild hervor. Sie wählen z. B. Schriftart, Schriftschnitt und Schriftgrad aus und stellen zusätzliche Effekte ein, z. B. Unterstreichung.

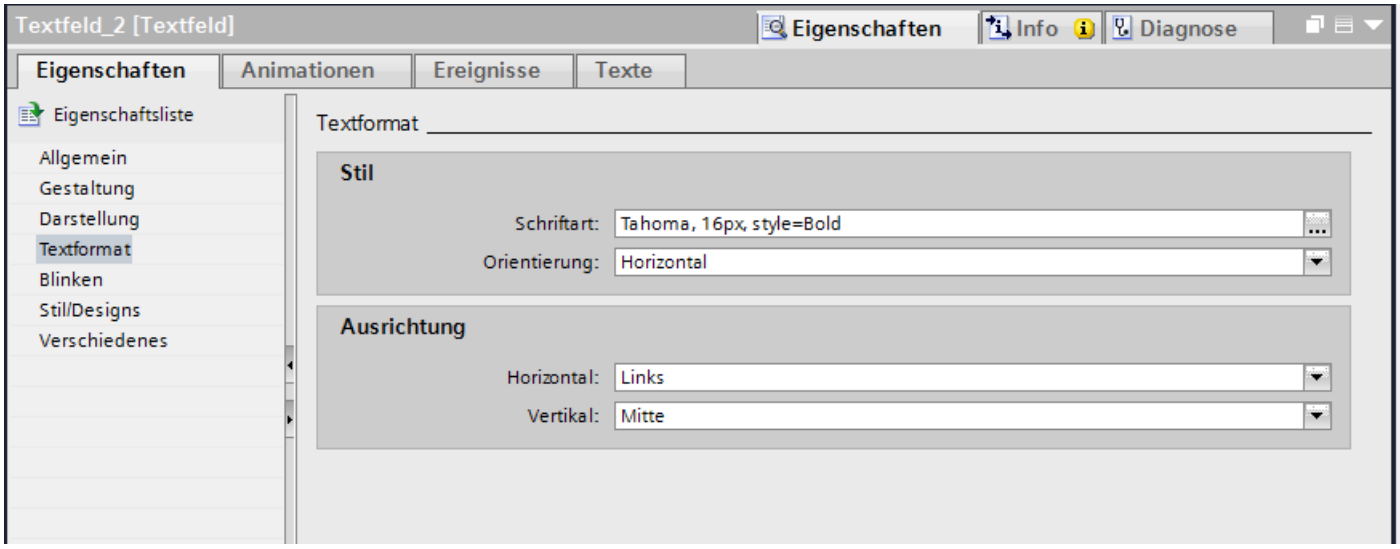

Die Einstellungen für Textauszeichnungen wie Schriftschnitt und Effekte beziehen sich immer auf den gesamten Text eines Bildobjekts. So können Sie z. B. eine Überschrift insgesamt fett darstellen, nicht aber einzelne Buchstaben oder Wörter dieser Überschrift.

Verwendbare Objekte

Einige Bildobjekte können nicht für alle Bediengeräte projektiert werden. In der Task Card "Werkzeuge" werden diese Bildobjekte nicht angezeigt. So können Sie z. B. für das Bediengerät KTP1000 keinen Schieberegler projektieren.

Siehe auch

Grundlagen zu Bildern (Seite 35)

1.1.4 Elemente und Grundeinstellungen

1.1.4.1 Task Cards

Einleitung

Im Editor "Bilder" stehen folgende Task Cards zur Verfügung:

- Werkzeuge: Anzeige- und Bedienobjekte
- Animationen: Vorlagen zur dynamischen Projektierung
- Layout: Hilfsmittel zur Anpassung der Darstellung
- Bibliotheken: Verwaltung der Projektbibliothek und der Globalen Bibliotheken

Hinweis
WinCC Basic

Die Task Card "Animationen" steht in WinCC Basic nicht zur Verfügung.

Werkzeuge

Die Task Card "Werkzeuge" enthält Objekte in unterschiedlichen Paletten:

- Basisobjekte
- Elemente
- Controls
- Eigene Controls (optional)
- Grafiken

Aus den Paletten fügen Sie die Objekte per Drag&Drop oder Doppelklick in Ihre Bilder ein. Die Auswahl an Objekten ist abhängig vom Bediengerät, das Sie gerade projektieren. Mit den folgenden Symbolen wechseln Sie den Anzeigemodus:

Symbol	Bedeutung
	Stellt die Objekte als Liste dar.
	Stellt die Objekte als Grafik dar.

Animationen

Die Task Card "Animationen" enthält in den Paletten die möglichen Dynamisierungen eines Bildobjekts. Aus den Paletten "Bewegungen", "Anzeige" und "Variablenanbindung" fügen Sie per Drag&Drop oder Doppelklick die Animation einem Bildobjekt hinzu.

Layout

Die Task Card "Layout" enthält folgende Paletten zum Anzeigen von Objekten und Elementen:

- Ebenen: Dient der Verwaltung der Ebenen von Bildobjekten. Die Ebenen werden in einer Baumansicht gezeigt und enthalten Informationen zur aktiven Ebene und zur Sichtbarkeit aller Ebene.
- Raster: Sie geben an, ob Sie die Objekte an einem Raster oder an anderen Objekten ausrichten wollen und stellen für ein Raster die Rastergröße ein.
- Objekte außerhalb des Bereichs: Objekte mit Namen, Position und Typ werden angezeigt, die sich außerhalb des sichtbaren Bereichs befinden

Bibliotheken

Die Task Card "Bibliotheken" zeigt folgende Bibliotheken in getrennten Paletten:

- Projektbibliothek: Die Projektbibliothek wird zusammen mit dem Projekt gespeichert.
- Globale Bibliothek: Die Globale Bibliothek wird als separate Datei im angegebenen Pfad auf dem Projektierungsrechner gespeichert.

Siehe auch

Ansicht zoomen (Seite 40)

Ansicht verschieben (Seite 40)

Grundlagen zu Bildern (Seite 33)

Arbeitsschritte (Seite 41)

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.4.2 Ansicht zoomen

Einleitung

Wenn Sie einen kleinen Ausschnitt des Bildes detaillierter sehen wollen, vergrößern Sie die Ansicht des Bildes im Arbeitsbereich. Maximal zoomen Sie auf 800 %.

Sie zoomen über die Symbolleiste im Arbeitsbereich oder über die Task Card "Layout > Zoom".

Sie haben verschieden Methoden für die Vergrößerung des Bildes, z. B. über den Vergrößerungsfaktor oder Sie passen den Arbeitsbereich an Höhe des Bildes an.

Voraussetzung

Das Bild ist geöffnet.

Vorgehen

Um eine Ansicht mit Auswahlrahmen zu zoomen, gehen Sie wie folgt vor:

1. Klicken Sie in der Symbolleiste auf das Symbol .
2. Ziehen Sie mit der Maus im Bild einen Auswahlrahmen auf.

Nachdem Sie die Maustaste losgelassen haben, wird der Ausschnitt auf die Größe des gesamten Arbeitsbereichs vergrößert, der sich innerhalb des Auswahlrahmens befindet.

Alternativ verwenden Sie in der unteren rechten Ecke des Bildes den Schieberegler.

Ergebnis

Der gewählte Bildausschnitt wird vergrößert angezeigt.

Siehe auch

Task Cards (Seite 38)

1.1.4.3 Ansicht verschieben

Einleitung

Um im Arbeitsbereich nur ein Ausschnitt des gesamten Bildes darzustellen, verwenden Sie das Symbol des Editors "Bilder".

Voraussetzung

- Ein Bild ist geöffnet.
- Im Arbeitsbereich wird nur ein Ausschnitt des gesamten Bildes dargestellt.

Vorgehen

Um eine Ansicht zu verschieben:

1. Klicken Sie in der unteren rechten Ecke des Arbeitsbereichs auf das Symbol und drücken Sie die linke Maustaste.
   Eine Miniaturansicht des gesamten Bildes wird aufgeblendet. Ein orangefarbener Rahmen zeigt den aktuellen Ausschnitt.
2. Halten Sie die Maustaste gedrückt und verschieben Sie den Rahmen in den gewünschten Ausschnitt.

Hinweis

Wenn Sie ein Bildobjekt aus dem sichtbaren Ausschnitt zu einem in der aktuellen Ansicht unsichtbaren Ausschnitt ziehen, scrollt das Bild mit.

Siehe auch

Task Cards (Seite 38)

1.1.5 Mit Bildern arbeiten

1.1.5.1 Arbeitsschritte

Arbeitsschritte

Zum Erstellen von Bildern sind folgende grundlegende Schritte nötig:

- Planen Sie die Struktur der Prozessdarstellung: Anzahl der Bilder und deren Anordnung. Beispiel: Teilprozesse werden in einzelnen Bildern dargestellt und in einem Hauptbild zusammengeführt.
- Planen Sie die Navigation innerhalb eines Bildes und zwischen den einzelnen Bildern.
- Passen Sie die Vorlagen und das Globale Bild an.
  Sie definieren Objekte zentral und belegen z. B. Funktionstasten.
- Erstellen Sie die Bilder. Nutzen Sie dabei folgende Möglichkeiten zum effizienten Erstellen:
  - Arbeiten mit Bibliotheken
  - Arbeiten mit Ebenen
  - Arbeiten mit Bildbausteinen

Siehe auch

Arbeitsschritte (Seite 42)

Neues Bild anlegen (Seite 43)

Bilder verwalten (Seite 44)

Startbild des Projekts definieren (Seite 47)

Grundlagen zu Bildern (Seite 33)

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

Task Cards (Seite 38)

1.1.5.2 Arbeitsschritte

Arbeitsschritte

Zum Erstellen von Bildern sind folgende grundlegende Schritte nötig:

- Planen Sie die Struktur der Prozessdarstellung: Anzahl der Bilder und deren Anordnung. Beispiel: Teilprozesse werden in einzelnen Bildern dargestellt und in einem Hauptbild zusammengeführt.
- Planen Sie die Navigation innerhalb eines Bildes und zwischen den einzelnen Bildern. Legen Sie Menüs und Symbolleisten an, um zwischen den Bildern zu navigieren.
- Erstellen Sie die Bilder. Nutzen Sie dabei folgende Möglichkeiten für eine effiziente Projektierung:
  - Arbeiten mit Bibliotheken
  - Arbeiten mit Bildbausteinen
  - Arbeiten mit Ebenen
  - Arbeiten mit Designs

Siehe auch

Arbeitsschritte (Seite 41)

Grundlagen zum Arbeiten mit Designs (Seite 66)

1.1.5.3 Neues Bild anlegen

Einleitung

Um Prozesse in Ihrer Anlage darzustellen, legen Sie Bilder an.

Voraussetzung

- Das Projekt ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Bilder > Neues Bild hinzufügen".
   Das Bild wird im Projekt angelegt und im Arbeitsbereich angezeigt. Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Geben Sie einen aussagekräftigen Namen für das Bild ein.
3. Konfigurieren Sie die Bildeigenschaften im Inspektorfenster:
   - Legen Sie fest, ob und auf welcher Vorlage das Bild beruhen soll.
   - Legen Sie "Hintergrundfarbe" und "Bildnummer" fest.
   - Hinterlegen Sie unter "Tooltip" einen dokumentierenden Text.
   - Legen Sie unter "Ebenen" fest, welche Ebenen im Engineering System angezeigt werden sollen.
   - Dynamisieren Sie die Anzeige des Bildes unter "Animationen".
   - Legen Sie unter "Ereignisse" fest, welche Funktionen beim Aufrufen und Verlassen des Bildes oder bei anderen Ereignissen in Runtime abgearbeitet werden sollen.

   Hinweis

   Nicht alle Bediengeräte unterstützen die Animation "Sichtbarkeit".

Ergebnis

Das Bild ist in Ihrem Projekt angelegt. Sie können in weiteren Arbeitsschritten Objekte und Bedienelemente aus der Task Card "Werkzeuge" einfügen und Funktionstasten belegen.

Siehe auch

Arbeitsschritte (Seite 41)

1.1.5.4 Neues Bild anlegen

Einleitung

Um Prozesse in Ihrer Anlage darzustellen, legen Sie Bilder an.

Voraussetzung

- Das Projekt ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein neues Bild anzulegen:

1. Doppelklicken Sie in der Projektnavigation auf "Bilder > Neues Bild hinzufügen".
   Das Bild wird im Projekt angelegt und im Arbeitsbereich angezeigt. Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Geben Sie einen aussagekräftigen Namen für das Bild ein, z. B. "Übersichtsplan".
   Verwenden Sie nicht die Sonderzeichen ?, . , ", /, \, *, <, >
   Dieser Name ist sprachneutral.
3. Geben Sie bei Bedarf einen aussagekräftigen Anzeigenamen ein. Dieser Name ist sprachabhängig und kann für die gewünschten Sprachen übersetzt werden. Sie können den Anzeigenamen z. B. mit der Systemfunktion SetPropertyByProperty in einem Textfeld ausgeben.
4. Konfigurieren Sie das Bild im Inspektorfenster:
   - Legen Sie unter "Allgemein" die Hintergrundfarbe und die Rasterfarbe fest.
   - Legen Sie einen dokumentierenden Tooltip fest.
   - Geben Sie unter "Darstellung" die Größe des Bildes an.
   - Geben Sie unter "Gestaltung" das Muster und die Farbe eines Füllmusters an.
   - Legen Sie unter "Ebenen" fest, welche Ebenen im Engineering System angezeigt werden sollen.
   - Legen Sie unter "Verschiedenes" die Tab-Reihenfolge für die Anwahl der Bildobjekte und die Auswahl der anzuwählenden Objekte fest.
   - Legen Sie unter "Verbergen/Anzeigen" fest, ob für Ebenen und/oder Objekte eine zu definierende Größe die Anzeige bestimmen soll.
   - Dynamisieren Sie die Anzeige- und Bedienobjekte unter "Animationen".
   - Legen Sie unter "Ereignisse" fest, welche Funktionen beim Aufrufen und Verlassen des Bildes oder bei anderen Ereignissen in Runtime abgearbeitet werden sollen.

Ergebnis

Das Bild ist in Ihrem Projekt angelegt. Sie können in weiteren Arbeitsschritten Objekte und Bedienelemente aus der Task Card "Werkzeuge" einfügen.

Siehe auch

Arbeitsschritte (Seite 41)

1.1.5.5 Bilder verwalten

Einleitung

In der Projektnavigation können Sie Bilder innerhalb eines Projekts in andere Gruppen verschieben, kopieren, umbenennen oder löschen.

Bild in Gruppe verschieben

1. Selektieren Sie in der Projektnavigation den Ordner "Bilder".
2. Wählen Sie im Kontextmenü den Befehl "Gruppe hinzufügen".
   Ein Ordner mit dem Namen "Gruppe_x" wird eingefügt.
3. Selektieren Sie in der Projektnavigation das Bild.
4. Ziehen Sie das Bild mit Drag&Drop auf die gewünschte Gruppe.
   Das Bild wird in diese Gruppe verschoben.

Bild kopieren

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren", um das Bild in die Zwischenablage zu kopieren.
3. Markieren Sie in der Projektnavigation die Stelle, an der Sie das Bild einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen", um das Bild einzufügen.
   Eine Kopie des Bildes wird eingefügt. Der Name des Originals wird bei der Kopie um eine laufende Nummer erweitert.

Alternativ halten Sie die Taste <Strg> gedrückt und ziehen Sie das Bild an die gewünschte Stelle.

Hinweis

Wenn Sie ein Bild mit verknüpfter Vorlage geräteübergreifend oder projektübergreifend kopieren, wird die Vorlage mitkopiert. Eine eventuell vorhandene, passende Vorlage wird nicht genutzt. Dies gilt insbesondere auch, wenn Sie Bilder über Drag&Drop kopieren.

Bild umbenennen

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen neuen Namen ein.
4. Drücken Sie die Taste <Enter>.

Alternativ verwenden Sie zum Umbenennen des Bildes die Taste <F2>.

Bild löschen

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Das Bild mit allen enthaltenen Objekten wird aus dem aktuellen Projekt gelöscht.

Siehe auch

Arbeitsschritte (Seite 41)

1.1.5.6 Startbild des Projekts definieren

Einleitung

Das Startbild ist das erste Bild, das beim Start des Projekts in Runtime angezeigt wird. Sie können für jedes Bediengerät ein eigenes Startbild definieren. Vom Startbild aus ruft der Bediener die weiteren Bilder auf.

Voraussetzung

Das Bild, das Startbild werden soll, ist im Projekt angelegt.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen > Allgemein".

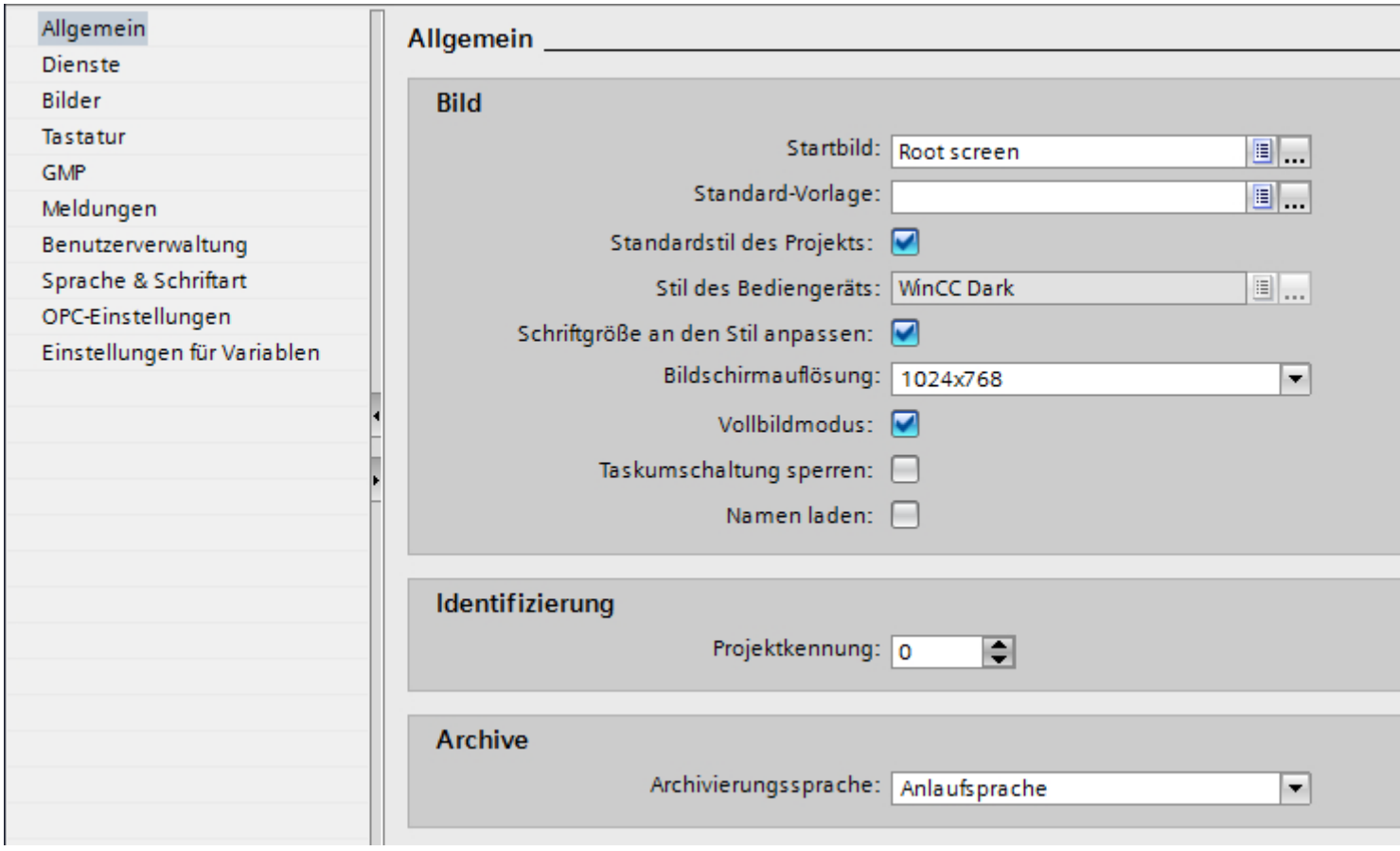

2. Wählen Sie als "Startbild" das gewünschte Bild aus.

Alternativ selektieren Sie ein Bild in der Projektnavigation und wählen im Kontextmenü "Als Startbild verwenden".

Ergebnis

Wenn Sie Runtime starten, wird das Startbild am Bediengerät angezeigt.

Siehe auch

Arbeitsschritte (Seite 41)

1.1.5.7 Startbild des Projekts definieren

Einleitung

Das Startbild ist das erste Bild, das beim Start des Projekts in Runtime angezeigt wird. Sie können für jedes Bediengerät ein eigenes Startbild definieren. Vom Startbild aus ruft der Bediener die weiteren Bilder auf.

Voraussetzung

Das Bild, das Startbild werden soll, ist im Projekt angelegt.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen > Allgemein".

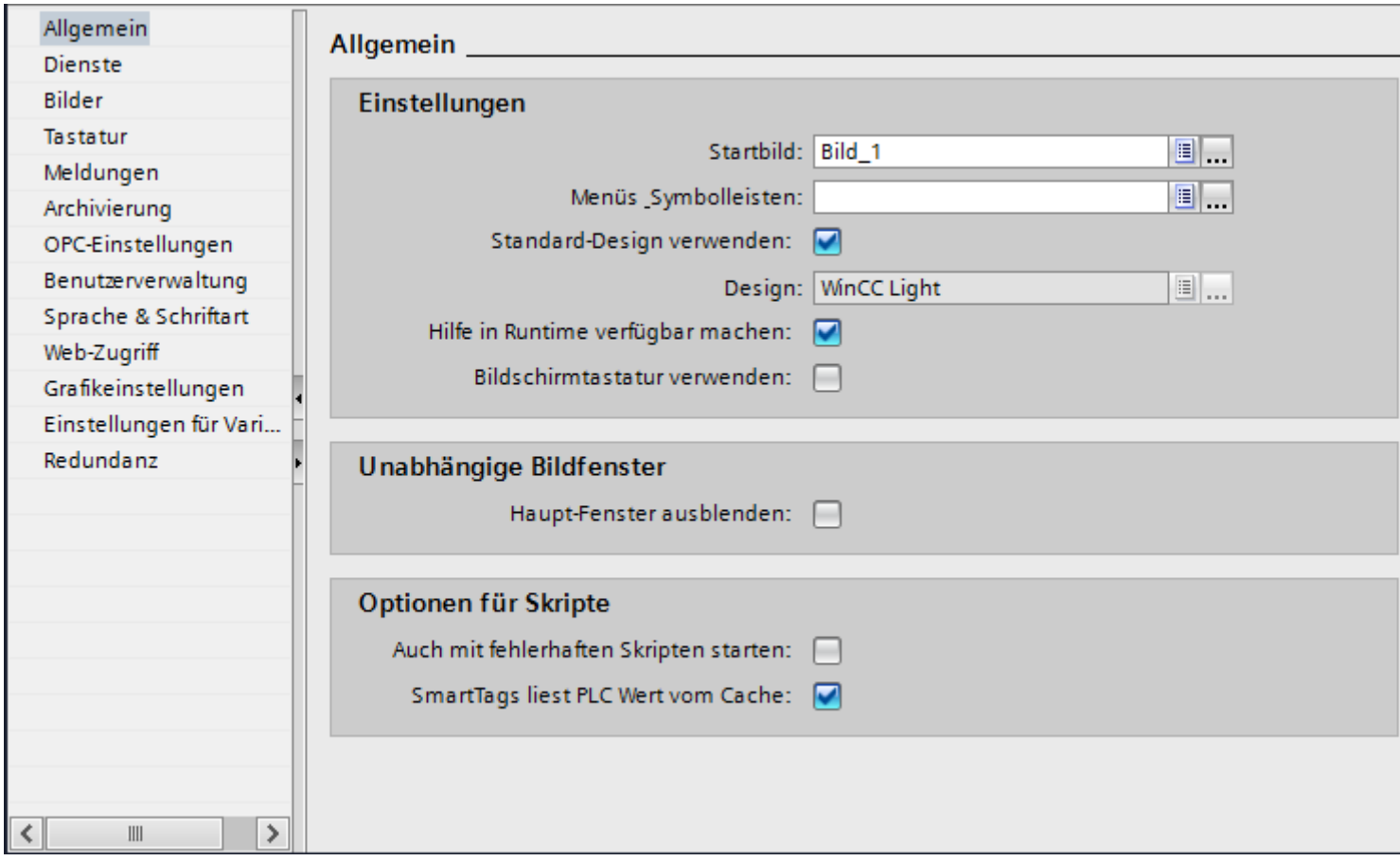

2. Wählen Sie als "Startbild" das gewünschte Bild aus.

   Alternativ selektieren Sie ein Bild in der Projektnavigation und wählen im Kontextmenü "Als Startbild verwenden".

Ergebnis

Wenn Sie Runtime starten, wird das Startbild am Bediengerät angezeigt.

1.1.5.8 Bildschirmauflösung ändern

Die Bildschirmauflösung ändern Sie in den Runtime-Einstellungen des Bediengeräts.

Die Bildschirmauflösung ist bei folgenden Bediengeräten projektierbar:

- RT Advanced

Feste Bildschirmauflösung projektieren

1. Öffnen Sie die Runtime-Einstellungen des Bediengeräts.
2. Wählen Sie unter "Allgemein > Bild > Bildschirmauflösung" die Bildschirmauflösung aus..

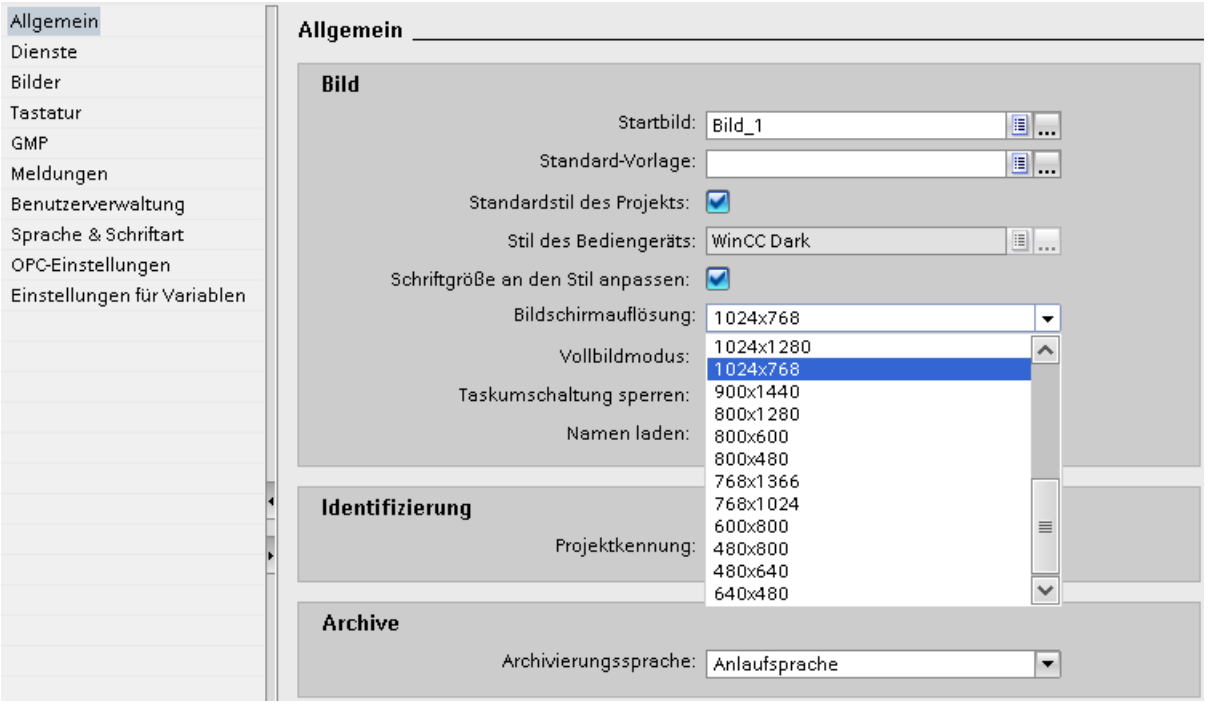

1.1.6 Arbeiten mit Vorlagen

1.1.6.1 Grundlagen zum Arbeiten mit Vorlagen

Einleitung

In einer Vorlage projektieren Sie Objekte, die in allen Bildern dargestellt werden, die auf dieser Vorlage basieren.

Dabei gelten folgende Regeln:

- Ein Bild muss nicht auf einer Vorlage basieren.
- Ein Bild beruht nur auf einer Vorlage.
- Für ein Gerät können Sie mehrere Vorlagen erstellen.
- Eine Vorlage kann nicht auf einer anderen Vorlage basieren.

Objekte für eine Vorlage

Innerhalb der Vorlage legen Sie Funktionen und Objekte fest, die für alle Bilder auf der Grundlage dieser Vorlage gelten sollen:

- Belegung von Funktionstasten: Bei Bediengeräten mit Funktionstasten belegen Sie auch Funktionstasten in der Vorlage. Diese Belegung überschreibt eine mögliche globale Belegung.
- Permanentbereiche: Einige Geräte, unterstützen im oberen Bereich des Bildes einen Permanentbereich für alle Bilder. Im Unterschied zur Vorlage belegt der Permanentbereich einen Bereich des Bildes nur für sich.
- Bedienobjekte: Sie können alle Bildobjekte in eine Vorlage einfügen, die Sie auch für ein Bild nutzen.

Anwendungsbeispiele

- In der Vorlage belegen Sie z. B. eine Funktionstaste mit der Funktion "AktiviereBild". Mit dieser Taste wechselt der Bediener in Runtime zu einem anderen Bild. Diese Projektierung gilt für alle Bilder, die auf dieser Vorlage basieren.
- In der Vorlage fügen Sie eine Grafik mit Ihrem Firmenlogo ein. Das Firmenlogo wird in allen Bildern angezeigt, die auf der Vorlage basieren.

Hinweis

Wenn sich ein Objekt aus der Vorlage an derselben Position befindet wie ein Objekt im Bild, wird das Vorlagenobjekt verdeckt.

Siehe auch

Grundlagen zu Bildern (Seite 33)

Arbeitsschritte (Seite 41)

Task Cards (Seite 38)

Neue Vorlage anlegen (Seite 52)

Vorlage verwalten (Seite 54)

Vorlage im Bild verwenden (Seite 55)

Globales Bild (Seite 50)

Permanentbereiche in der Vorlage projektieren (Seite 53)

1.1.6.2 Globales Bild

Einleitung

Für alle Bilder eines Bediengeräts legen Sie globale Elemente fest, unabhängig von der verwendeten Vorlage.

Funktionstasten

Bei Bediengeräten mit Funktionstasten belegen Sie die Funktionstasten global innerhalb des Editors "Globales Bild". Diese globale Belegung gilt für alle Bilder des Bediengeräts.

Um Funktionstasten lokal in Bildern oder Vorlagen zu belegen, gehen Sie wie folgt vor:

1. Klicken Sie in Ihren Bildern bzw. Vorlagen auf die Funktionstaste.
2. Deaktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Globale Zuordnung verwenden".

Anzeige- und Bedienobjekte für Meldungen

Die als globale Objekte zur Verfügung stehenden Objekte "Meldefenster" und "Meldeindikator" projektieren Sie innerhalb des Editors "Globales Bild".

Die Objekte "Meldefenster" und "Meldeindikator" werden immer im Vordergrund eingeblendet.

Für Comfort Panels haben Sie auch die Möglichkeit eine "System-Diagnoseanzeige" im Globalen Bild zu projektieren.

Hinweis

Wenn Sie in einer Vorlage einen Permanentbereich projektiert haben, positionieren Sie Meldefenster und Meldeindikator nicht im Bereich des Permanentbereichs. Das Meldefenster und der Meldeindikator werden sonst in Runtime nicht angezeigt.

Den Permanentbereich sehen Sie im Editor "Globales Bild" jedoch nicht.

Reihenfolge bei der Projektierung von Bildern

Bei der Projektierung gilt folgende Reihenfolge:

- das Globale Bild liegt vor Bildern und Vorlagen
- Bilder liegen vor Vorlagen

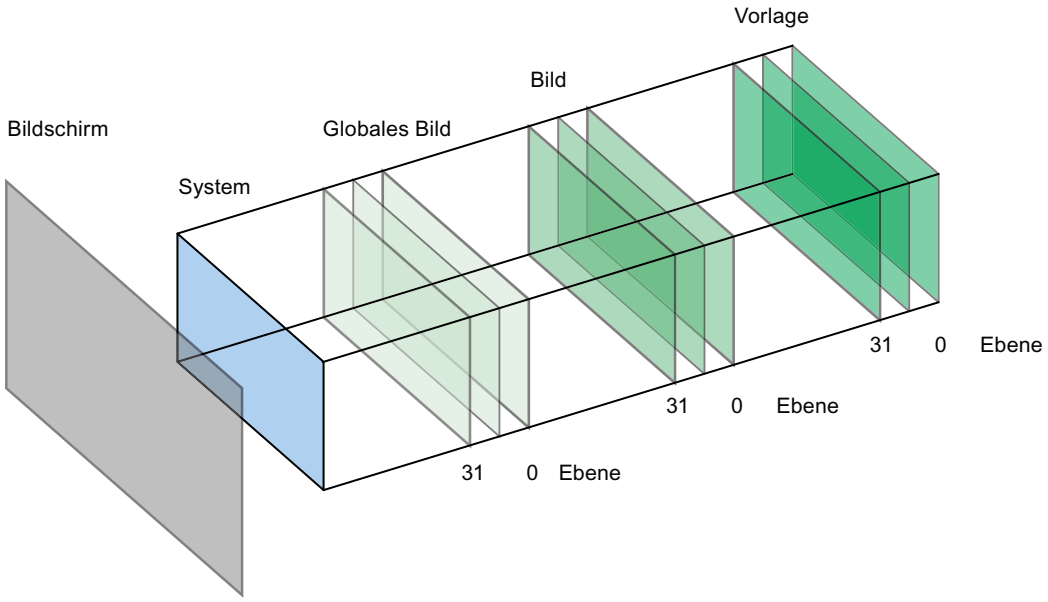

Die Systemebene ist nicht konfigurierbar. In ihr liegen

- Eingabedialoge
- Meldungen vom Betriebssystem
- bei Touch Panels die Direkttasten

Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.6.3 Neue Vorlage anlegen

Einleitung

In einer Vorlage bearbeiten Sie zentral Objekte und Funktionstasten. Wenn Sie in der Vorlage ein Objekt oder die Belegung einer Funktionstaste ändern, ändert sich das Objekt in allen Bildern, die auf der Vorlage basieren.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Voraussetzung

- Das Projekt ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation unter "Bilderverwaltung > Vorlagen > Neue Vorlage hinzufügen".
   Die Vorlage wird im Projekt angelegt und im Arbeitsbereich angezeigt.
   Die Eigenschaften der Vorlage werden im Inspektorfenster angezeigt.
2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen der Vorlage fest
3. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Ebenen" fest, welche Ebenen im Engineering System angezeigt werden.
4. Fügen sie die benötigten Objekte aus der Task Card "Werkzeuge" hinzu.
5. Projektieren Sie die Funktionstasten.

Ergebnis

Die Vorlage ist in Ihrem Projekt angelegt.

Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.6.4 Permanentbereiche in der Vorlage projektieren

Einleitung

Im Permanentbereich projektieren Sie Objekte, die in allen Bildern sichtbar sind. Im Unterschied zur Vorlage belegt der Permanentbereich einen Bereich des Bildes nur für sich.

Permanentbereiche können Sie in einer Vorlage oder in einem Bild projektieren. In der Projektnavigation finden Sie den Permanentbereich unter "Bildverwaltung > Permanentbereich".

Permanentbereiche projektieren

Um einen Permanentbereich zu projektieren, gehen Sie wie folgt vor:

1. Öffnen Sie eine Vorlage für ein Bediengerät mit Permanentbereich.
2. Ziehen Sie mit der Maus (Cursorform:) die Begrenzung am oberen Rand des editierbaren Bildbereichs nach unten.

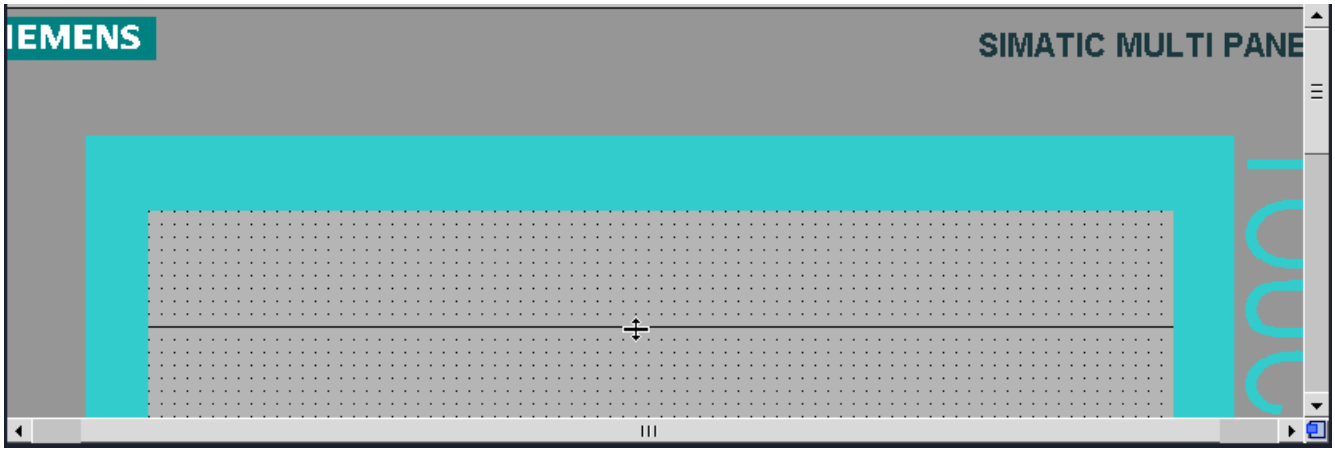

3. Projektieren Sie die gewünschten Objekte im Bereich des Permanentbereichs.

Die voreingestellte Höhe des Permanentbereichs ist "0". Die maximale Höhe ist die maximale Höhe des Bildes minus 10 Pixel.

Ergebnis

Der Inhalt des Permanentbereichs wird in allen Bildern und in der Vorlage angezeigt. Der Bereich oberhalb dieser Linie wird jetzt in allen Bildern des Bediengeräts als Permanentbereich genutzt. Bereits projektierte Objekte in den Bildern werden um die Höhe des Permanentbereichs nach unten geschoben.

Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.6.5 Vorlage verwalten

Einleitung

Im Projektfenster verschieben, kopieren, umbenennen oder löschen Sie Vorlagen innerhalb eines Projekts.

Vorlage in Gruppe verschieben

1. Selektieren Sie in der Projektnavigaton "Bildverwaltung > Vorlagen".
2. Wählen Sie im Kontextmenü "Gruppe hinzufügen".
   Ein Ordner mit dem Namen "Gruppe_x" wird eingefügt.

3. Selektieren Sie in der Projektnavigation die Vorlage.
4. Ziehen Sie die Vorlage mit Drag&Drop auf die gewünschte Gruppe.
   Die Vorlage wird in diese Gruppe verschoben.

Vorlage kopieren

1. Selektieren Sie in der Projektnavigation die Vorlage
2. Wählen Sie im Kontextmenü "Kopieren".
3. Markieren Sie in der Projektnavigation die Stelle, an der Sie die Vorlage einfügen wollen.
4. Um die Vorlage einzufügen, wählen Sie im Kontextmenü "Einfügen".
   Die Kopie erhält automatisch einen eindeutigen Namen.

Alternativ ziehen Sie die Vorlage mit gedrückter Taste <Strg> an die gewünschte Stelle.

Vorlage löschen

1. Selektieren Sie in der Projektnavigation die zu löschende Vorlage.
2. Wählen Sie im Kontextmenü "Löschen".
   Die Vorlage mit allen enthaltenen Objekten wird aus dem aktuellen Projekt gelöscht.

Vorlage einem Bild zuweisen

1. Selektieren Sie in der Projektnavigation das Bild, dem Sie die Vorlage zuweisen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Wählen Sie unter "Vorlage" die gewünschte Vorlage aus.
   Die ausgewählte Vorlage mit allen enthaltenen Objekten wird dem Bild zugeordnet.

Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.6.6 Vorlage im Bild verwenden

Einleitung

Sie verwenden eine Vorlage im Bild. Alle in der Vorlage projektierten Objekte sind auch im Bild verfügbar.

Voraussetzung

Eine Vorlage ist angelegt.

Ein Bild ist angelegt.

Vorgehensweise

Um eine Vorlage in einem Bild zu verwenden, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Projektnavigation auf ein Bild. Das Bild wird im Arbeitsbereich geöffnet.
2. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Wählen Sie unter "Vorlage" eine Vorlage, die auf das Bild angewendet wird.

Vorlage im Bild einblenden

Wenn Sie ein Bild editieren, können Sie eine vorhandene Vorlage im Bild einblenden.

Um eine Vorlage im Bild einzublenden, gehen Sie wie folgt vor:

1. Aktivieren Sie im Menü "Extras > Einstellungen > Visualisierung > Vorlage in Bildern anzeigen".

Ergebnis

Das Bild basiert auf der gewählten Vorlage. Alle Objekte, die Sie in der Vorlage projektiert haben, sind im Bild vorhanden. Die Vorlage wird im Bild angezeigt.

Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 49)

1.1.7 Arbeiten mit Pop-up-Bildern

1.1.7.1 Grundlagen zum Arbeiten mit Pop-up-Bildern

Einleitung

Mit einem Pop-up-Bild projektieren Sie ein Bild für zusätzliche Inhalte, z. B. Einstellungen eines Objekts. Das Pop-up-Bild wird über dem aktuellen Bild eingeblendet, sobald eine Systemfunktion aufgerufen wird.

In einem Bild wird nur ein Pop-up-Bild gleichzeitig angezeigt.

Hinweis

Geräteabhängigkeit des Objekts "Pop-up-Bild"

Das Objekt "Pop-up-Bild" ist nur für KTP Mobile Panels, Comfort Panels und RT Advanced verfügbar.

Verwendung

Sie projektieren im Bild eine Schaltfläche und belegen diese mit der Funktion "ZeigePopupBild" oder "ZeigePopupBildGrößeVeränderbar" um ein Pop-up-Bild mit einer definierten Größe anzuzeigen.

Beim Betätigen der Schaltfläche wird ein Pop-up-Bild geöffnet oder geschlossen. Im Pop-up-Bild nimmt der Bediener Einstellungen vor.

Größe

Sie haben die Möglichkeit, die Pop-up-Bilder zu projektieren, die größer als die Bildschirmgröße sind:

- Horizontal, maximal bis zur doppelten Breite
- Vertikal, maximal bis zur sechsfachen Höhe

Objekte im Pop-up-Bild

Innerhalb des Pop-up-Bilds projektieren Sie folgende Objekte:

- Basisobjekte, wie z. B. Linien und Polygon-basierte Objekte
- Elemente, wie z. B. Schaltflächen
- Controls, wie z. B. Kurvenanzeige oder Meldeanzeige
- Bildbausteine

Hinweis

Im Pop-up-Bild können Meldefenster, System-Diagnosefenster und Meldeindikator nicht projektiert werden.

Softkeys und Hotkeys werden im Objekt "Pop-up-Bild" nicht unterstützt.

Hinweis

Objekte im Pop-up-Bild sowie das Pop-up-Bild unterstützten keinen Zugriff mit VB-Scripting.

1.1.7.2 Neues Pop-up-Bild anlegen

Einleitung

Ein neues Pop-up-Bild legen Sie in der Projektnavigation > Bildverwaltung an. Die Größe eines Pop-up-Bildes legen Sie in den Eigenschaften fest.

Voraussetzungen

- Das Projekt ist angelegt
- Das Inspektorfenster ist geöffnet

Neues Pop-up-Bild anlegen

1. Doppelklicken Sie in der Projektnavigation unter "Bilderverwaltung > Pop-up-Bilder > Neues Pop-up-Bild hinzufügen".
   Das Pop-up-Bild wird im Projekt angelegt und im Arbeitsbereich angezeigt.
   Die Eigenschaften des Pop-up-Bilds werden im Inspektorfenster angezeigt.
2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen des Pop-up-Bilds fest.
3. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen" fest, welche Ebenen im Engineering System angezeigt werden.
4. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung" die Höhe und die Breite des Pop-up-Bilds fest.
5. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Bildlaufleiste" die Farben für die Bildlaufpfeile und den Ziehpunkt in der Bildlaufleiste fest.
6. Fügen Sie die benötigten Objekte aus der Task Card "Werkzeuge" hinzu.
7. Projektieren Sie die Funktionstasten.

Hinweis

Das Objekt "Pop-up-Bild" und darin projektierte Objekte können nicht mithilfe von VB-Scripting adressiert werden.

Hinweis

Objekte und Controls im Pop-up-Bild können dynamisiert werden. Das Pop-up-Bild an sich unterstützt keine Dynamisierung.

Ergebnis

Sie haben ein neues Pop-up-Bild angelegt, in welchem Sie Objekte und Funktionen projektiert haben.

1.1.7.3 Pop-up-Bilder verwalten

Einleitung

Im Projektfenster verschieben, kopieren, umbenennen oder löschen Sie Pop-up-Bilder innerhlab eines Projekts.

Pop-up-Bild in Gruppe verschieben

1. Selektieren Sie in der Projektnavigaton "Bildverwaltung > Pop-up-Bilder".
2. Wählen Sie im Kontextmenü "Gruppe hinzufügen".
   Ein Ordner mit dem Namen "Gruppe_x" wird eingefügt.
3. Selektieren Sie in der Projektnavigation das Pop-up-Bild.
4. Ziehen Sie das Pop-up-Bild mit Drag&Drop auf die gewünschte Gruppe.
   Das Pop-up-Bild wird in diese Gruppe verschoben.

Pop-up-Bild kopieren

1. Selektieren Sie in der Projektnavigation das Pop-up-Bild.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Markieren Sie in der Projektnavigation die Stelle, an der sie das Pop-up-Bild einfügen wollen.
4. Wählen Sie im Kontextmenü "Einfügen".
   Die Kopie erhält automatisch einen eindeutigen Namen.

Pop-up-Bild löschen

1. Selektieren Sie in der Projektnavigation das zu löschende Pop-up-Bild.
2. Wählen Sie im Kontextmenü "Löschen".
   Das Pop-up-Bild mit allen enthaltenen Objekten wird as dem aktuellen Projekt gelöscht.

1.1.7.4 Pop-up-Bild verwenden

Einleitung

Sie projektieren die Systemfunktion zum Öffnen eines Pop-up-Bildes an ein Ereignis eines Bedienobjekts. Wenn das Ereignis in Runtime eintritt, wird die verknüpfte Systemfunktion ausgeführt.

Hinweis

Sie projektieren eine Schaltfläche zum Öffnen eines Pop-up-Bilds in einem Slide-in-Bild.

Falls die projektierte Position des Pop-up-Bilds außerhalb des Slide-in-Bildes liegt, wird beim Schließen dieses Pop-up-Bildes auch das Slide-in-Bild geschlossen.

Voraussetzungen

- Ein Pop-up-Bild ist projektiert.
- Ein Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Projektieren Sie eine Schaltfläche.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Wählen Sie ein Ereignis, z. B. "Klicken".
4. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
5. Wählen Sie die Systemfunktion "ZeigePopupBild".

Ergebnis

Wenn der Bediener in Runtime auf die Schaltfläche klickt, öffnet sich das Pop-up-Bild.

1.1.8 Arbeiten mit Slide-in-Bildern

1.1.8.1 Grundlagen zum Arbeiten mit Slide-in-Bildern

Einleitung

Das Objekt „Slide-in-Bild“ ermöglicht eine schnelle Navigation zwischen dem Startbild und den außerhalb vom Startbild hinterlegten Slide-in-Bildern, auf denen zusätzliche Inhalte projektiert sind. Die Navigation wird durch die Multitouch-Funktionen unterstützt. Die konfigurierbaren Handles, die an den Rändern des Startbildes erscheinen, ermöglichen einen schnellen Zugriff auf die hinterlegten Slide-in-Bilder.

Die Größe der Slide-in-Bilder hängt vom jeweiligen Bediengerät ab.

Hinweis

Geräteabhängigkeit des Objekts "Slide-in-Bild"

Das Objekt "Slide-in-Bild" ist nur für KTP Mobile Panels, Comfort Panels und RT Advanced verfügbar.

Verwendung

Mit dem Objekt "Slide-in-Bild" konfigurieren Sie bis zu vier Slide-in-Bilder für zusätzliche Inhalte und Bedienelemente, z. B. eine virtuelle Tastatur oder einen Systemdialog.

Standardmäßig sind für jedes Gerät folgende Slide-in-Bilder angelegt:

- "Slide-in-Bild oben": erscheint in Runtime im Startbild oben
- "Slide-in-Bild unten": erscheint in Runtime im Startbild unten
- "Slide-in-Bild links": erscheint in Runtime im Startbild links
- "Slide-in-Bild rechts": erscheint in Runtime im Startbild rechts

In jedem dieser Slide-in-Bilder können Sie Objekte projektieren.

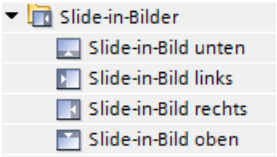

Hinweis

Es ist nicht möglich, die angelegten Slide-in-Bilder zu löschen oder eigene Slide-in-Bilder anzulegen.

Objekte im Slide-in-Bild

Innerhalb des Slide-in-Bildes projektieren Sie folgende Objekte:

- Basisobjekte, wie z. B. Linien und Polygon-basierte Objekte
- Elemente, wie z. B. Schaltflächen
- Controls, wie z. B. Kurvenanzeige oder Meldeanzeige
- Bildbausteine

Hinweis

Objekte im Slide-in-Bild sowie das Slide-in-Bild unterstützen keinen Zugriff mit VB-Scripting.

Hinweis

Im Slide-in-Bild können Freigabeschaltfächen und gesperrte Bedienelemente nicht projektiert werden.

Softkeys und Hotkeys werden im Slide-in-Bild nicht unterstützt.

Siehe auch

Slide-in-Bilder konfigurieren (Seite 62)

Slide-in-Bilder für Geräte ohne Multitouch-Funktion konfigurieren (Seite 64)

Slide-in-Bilder verwenden (Seite 65)

1.1.8.2 Slide-in-Bilder konfigurieren

Einleitung

Standardmäßig sind für jedes Gerät vier Slide-in-Bilder in der Projektnavigation unter "Bildverwaltung > Slide-in-Bilder" angelegt. Für jedes dieser Slide-in-Bilder konfigurieren Sie über die Eigenschaften die Größe, die Darstellung und den Handle für die Bedienung in Runtime.

Voraussetzung

- Das Projekt ist angelegt.
- Ein Bediengerät ist angelegt.
- Das Inspektorfenster ist geöffnet.

Eigenschaften für Slide-in-Bilder konfigurieren

1. Selektieren Sie in der Projektnavigation unter "Bilderverwaltung > Slide-in-Bilder" das Slide-in-Bild, das Sie konfigurieren möchten.
   Die Eigenschaften des Slide-in-Bildes werden im Inspektorfenster angezeigt.

   Hinweis

   Die Namen aller Slide-in-Bilder sind schreibgeschützt und können nicht geändert werden.

2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" in der Auswahlliste "Hintergrundfarbe" die Hintergrundfarbe des Slide-in-Bildes fest.
3. Legen Sie in der Auswahlliste "Rasterfarbe" die Rasterfarbe des Slide-in-Bildes fest.
4. Aktivieren Sie das Kontrollkästchen "Aktivieren".
5. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen" fest, welche Ebenen im Engineering System angezeigt werden.
6. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung" die Höhe und die Breite des Slide-in-Bildes fest:
   - Für "Slide-in-Bild oben" und "Slide-in-Bild unten" legen Sie nur die Höhe fest.
   - Für "Slide-in-Bild links" und "Slide-in-Bild rechts" legen Sie nur die Breite fest.

Hinweis

Wenn Sie nicht alle vier Slide-in-Bilder benötigen, aktivieren Sie nur die für Sie relevanten Slide-in-Bilder.

Wenn Sie keine Slide-in-Bilder benötigen, lassen Sie das Kontrollkästchen "Aktivieren" bei allen vier Slide-in-Bildern leer.

Hinweis

Es ist nicht möglich die Slide-in-Bilder zu löschen oder auf ein anderes Bediengerät zu kopieren.

Jedoch können Sie die in einem Slide-in-Bild projektierten Objekte in ein anderes Slide-in-Bild kopieren.

Handles für Slide-in-Bilder konfigurieren

1. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Handle" im Bereich "Linien" in der Auswahlliste "Farbe" die Farbe der ersten Linie des Handles fest.
2. Legen Sie in der Auswahlliste "Alternativfarbe" die Farbe der zweiten Linie des Handles fest.
3. Legen Sie unter "Bedienbarer Bereich" in der Auswahlliste "Farbe" die Farbe für den bedienbaren Bereich des Handles fest.
4. Aktivieren Sie im Bereich "Sichtbarkeit" eine der folgenden Optionen:
   - "Handle automatisch ausblenden" (standardmäßig aktiviert)
   - "Handle immer anzeigen"
   - "Handle nie anzeigen"

Hinweis

Wenn Sie die Option "Handle automatisch ausblenden" auswählen, wird der Handle in Runtime erst dann sichtbar, wenn der Bediener den bedienbaren Bereich antippt.

Die Option "Handle nie anzeigen" wählen Sie, wenn das Slide-in-Bild über Systemfunktion "ZeigeSlideInBild" konfiguriert wird.

Hinweis

Wenn ein als sichtbar konfigurierter Handle ein projektiertes Objekt in einem Bild, z. B. eine Schaltfläche, komplett oder teilweise verdeckt, informiert eine Warnung Sie darüber.

Hinweis

Für den Handle sind folgende Maße festgelegt:

- Die Breite eines Handles der Objekte "Slide-in-Bild oben" und oder "Slide-in-Bild unten" beträgt 1/3 der Bildlänge.
- Die Höhe eines Handles der Objekte "Slide-in-Bild links" und "Slide-in-Bild rechts" beträgt 1/3 der Bildbreite.

Ergebnis

Sie haben die Eigenschaften der Slide-in-Bilder und den Handle für die Bedienung in Runtime konfiguriert.

Siehe auch

Grundlagen zum Arbeiten mit Slide-in-Bildern (Seite 60)

1.1.8.3 Slide-in-Bilder für Geräte ohne Multitouch-Funktion konfigurieren

Einleitung

Um die Slide-in-Bilder auf Geräten ohne Multitouch-Funktionen, z. B. Key Panels, verfügbar zu machen, verwenden Sie die Systemfunktion "ZeigeSlideInBild". Diese Systemfunktion verknüpfen Sie mit Schaltflächen. Mithilfe von Schaltflächen kann der Bediener zwischen dem Startbild und den Slide-in-Bildern navigieren.

Voraussetzung

- Das Projekt ist angelegt.
- Ein Bediengerät ist angelegt.
- Ein Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie aus dem Werkzeugfenster das Objekt "Schaltfläche" in das Bild.

   Hinweis

   Wenn Sie direkt im Startbild die Schaltflächen für die Bedienung der Slide-in-Bilder anlegen, ist die Navigation für den Bediener in Runtime einfacher.

2. Geben Sie einen geeigneten, beliebig langen Text für die Schaltfläche ein.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
   Der Dialog "Funktionsliste" wird geöffnet.
4. Wählen Sie aus der Gruppe "Tastaturbedienung für Bildobjekte" die Systemfunktion "ZeigeSlideInBild".
5. Weisen Sie der Funktion "ZeigeSlideInBild" die ID des Slide-in-Bildes zu, das in Runtime beim Betätigen der Schaltfläche erscheint:
   - "Slide-in-Bild links"
   - "Slide-in-Bild oben"
   - "Slide-in-Bild rechts"
   - "Slide-in-Bild unten"
6. Weisen Sie der Funktion "ZeigeSlideInBild" den passenden Modus zu:
   - "Umschalten"
   - "Aus"
   - "Ein"

Ergebnis

Sie haben die Slide-in-Bilder für die Bedienung auf den Geräten ohne Multitouch-Funktion konfiguriert. Wenn der Bediener in Runtime auf die betreffende Schaltfläche im Bild klickt, verhält sich das Slide-in-Bild entsprechend der Projektierung.

Siehe auch

Grundlagen zum Arbeiten mit Slide-in-Bildern (Seite 60)

1.1.8.4 Slide-in-Bilder verwenden

Einleitung

In Slide-in-Bildern projektieren Sie zusätzliche Inhalte und Bedienelemente, z. B. Navigationsschaltflächen, eine Meldeanzeige oder eine Trendanzeige.

Voraussetzung

- Das Projekt ist angelegt.
- Ein Bediengerät ist angelegt.
- Ein Slide-in-Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Fügen Sie in das Slide-in-Bild die benötigten Objekte aus der Task Card "Werkzeuge" hinzu.
2. Projektieren Sie die Funktionstasten.

Hinweis

Das Objekt "Slide-in-Bild" und darin projektierte Objekte können nicht mithilfe von VB-Scripting adressiert werden.

Hinweis

Objekte und Controls in den Slide-in-Bildern können dynamisiert werden. Die Slide-in-Bilder an sich unterstützen keine Dynamisierung.

Ergebnis

Sie haben im Slide-in-Bild Objekte und Funktionen projektiert.

Siehe auch

Grundlagen zum Arbeiten mit Slide-in-Bildern (Seite 60)

1.1.9 Arbeiten mit Designs

1.1.9.1 Grundlagen zum Arbeiten mit Designs

Was sind Designs?

Designs sind Vorlagen für die Darstellung von Bildern und der darin projektierten Objekte. Ein Design enthält z. B. Farbvorgaben für Vordergrund, Rahmenlinie, Text, usw..

Das Design wird auf folgende Objekte angewendet:

- Bilder
- Menüs und Symbolleisten
- Objekte in Bildern

Verwendung

Um die Benutzeroberflächen mehrerer Bediengeräte in einem einheitlichen Layout zu gestalten, weisen Sie jedem Bediengerät das gleiche Design zu.

Wenn Sie das Layout der Bilder und Objekte ändern wollen, passen Sie das aktuelle Design an Ihre Bedürfnisse an. Das neue Design wird automatisch von allen Bildern und Objekten des Bediengeräts übernommen.

Designs verwalten

Designs verwalten Sie im Editor "Designs".

WinCC enthält einige vordefinierte Designs, wie "WinCC Classic" für Benutzeroberflächen im klassischen WinCC-Layout und "WinCC Glass" für Benutzeroberflächen im Stil von Windows Vista. Vordefinierte Designs können weder gelöscht noch verändert werden.

Siehe auch

Standarddesign verwenden (Seite 73)

1.1.9.2 Editor Designs

Einleitung

Designs sind projektspezifisch. Sie finden den Editor "Designs" in der Projektnavigation unter "Gemeinsame Daten".

Aufbau des Editors "Designs"

Der Bereich "Designs" listet alle Designs auf. Der Bereich "Vorschau" zeigt, wie Bilder und Objekte mit dem ausgewählten Design dargestellt werden.

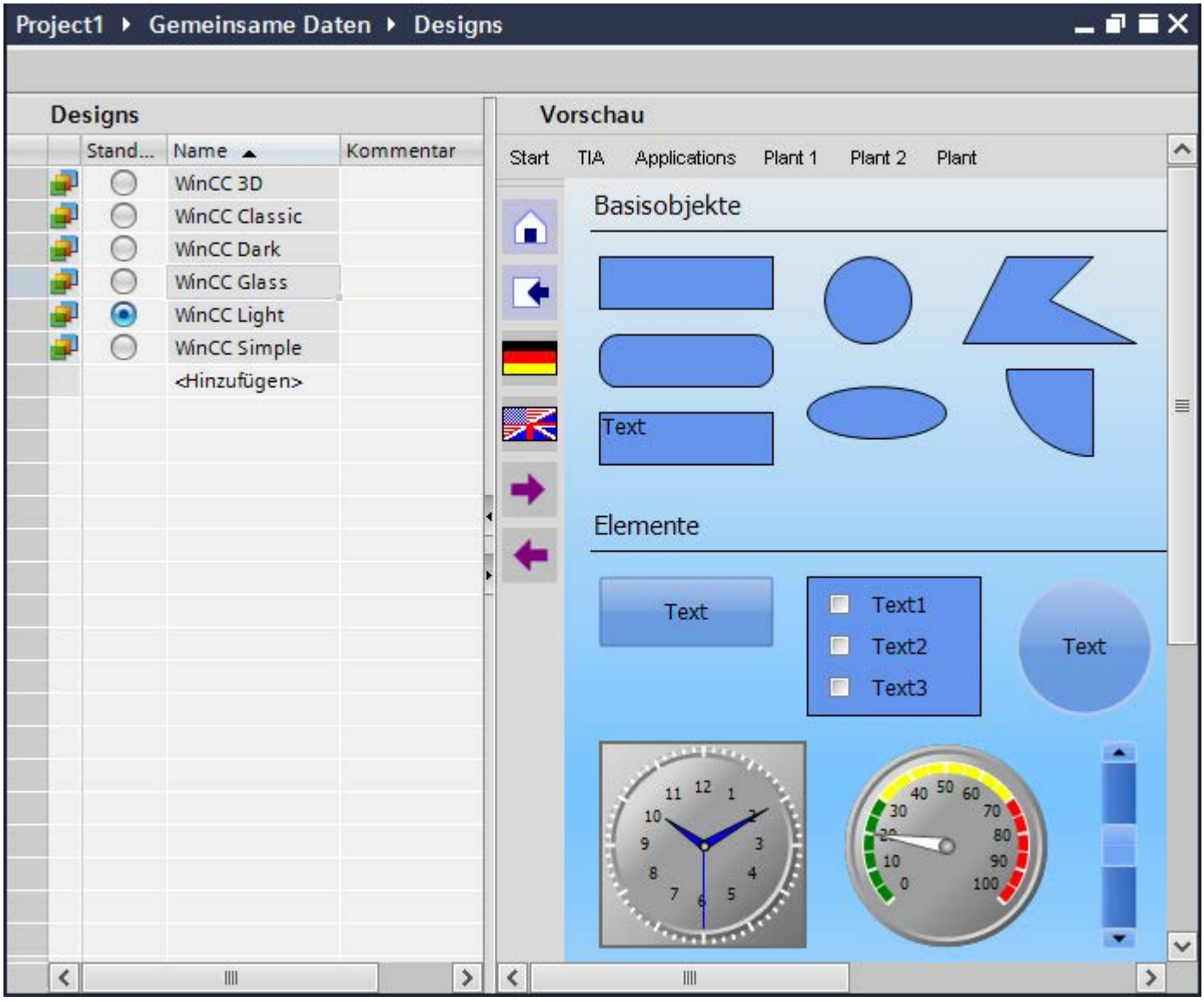

Vordefinierte Designs sind grau hinterlegt, können weder gelöscht noch verändert werden. Das Standarddesign ist in der Spalte "Standard" selektiert.

Für eine Vorschau auf ein anderes Design, klicken Sie in der Liste "Designs" auf das gewünschte Design. Die "Vorschau" im rechten Bereich wird aktualisiert:

- Die erste Zeile bietet eine Vorschau auf die Darstellung des Menüs.
- Die Symbole am linken Rand bieten eine Vorschau auf die Darstellung der Symbolleiste.
- Die Objekte der Task Card "Werkzeuge" sind nach Basisobjekten, Elementen und Controls gruppiert.

Im Inspektorfenster informieren Sie sich im Detail darüber, welche Darstellungseigenschaften das Design festlegt.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Standarddesign verwenden (Seite 73)

Design erstellen (Seite 68)

Design ändern (Seite 69)

Design löschen (Seite 69)

1.1.9.3 Design erstellen

Voraussetzung

Der Editor "Designs" ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Liste der Designs auf den Eintrag "Hinzufügen".
   Ein neues Design wird angelegt. Für Bilder, Menüs, Symbolleisten und Objekte sind die WinCC-Standardeigenschaften voreingestellt.
2. Überschreiben Sie den vorgeschlagenen Namen mit einem aussagekräftigen Namen.
3. Geben Sie in der Spalte "Kommentar" eine Beschreibung des Designs ein, z. B. den Verwendungszweck.
4. Konfigurieren Sie das Design wie gewünscht.

Ergebnis

Sie haben eine neues Design angelegt und konfiguriert.

Siehe auch

Editor Designs (Seite 66)

Design ändern (Seite 69)

Design löschen (Seite 69)

Grundlagen zum Arbeiten mit Designs (Seite 66)

1.1.9.4 Design ändern

Voraussetzung

Der Editor "Designs" ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Liste der Designs das Design, das Sie ändern wollen.

   Hinweis

   Die vordefinierten Designs von WinCC können Sie nicht ändern.

2. Nehmen Sie im Inspektorfenster die gewünschten Anpassungen vor.
3. Kontrollieren Sie das Ergebnis in der Vorschau.

Die Änderungen werden sofort wirksam.

Siehe auch

Editor Designs (Seite 66)
Design erstellen (Seite 68)
Design löschen (Seite 69)
Grundlagen zum Arbeiten mit Designs (Seite 66)

1.1.9.5 Design löschen

Voraussetzung

Der Editor "Designs" ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

- Wählen Sie im Kontextmenü des gewünschten Designs "Löschen".

  Hinweis

  Die vordefinierten Designs von WinCC können Sie nicht ändern.

Ergebnis

Das Design wird gelöscht. WinCC weist Bediengeräten, denen das gelöschte Design in den Runtime-Einstellungen als Standarddesign zugewiesen wurde, das vordefinierte Design "WinCC Classic" als neues Standarddesign zu.

Siehe auch

Editor Designs (Seite 66)

Design erstellen (Seite 68)

Design ändern (Seite 69)

Grundlagen zum Arbeiten mit Designs (Seite 66)

1.1.9.6 Design anwenden

Kernaussage

In WinCC ist jedem Bediengerät ein Design als Basisdesign zugewiesen. Folgende Vorgaben des Basisdesigns werden auf alle Bilder des Bediengeräts angewendet:

- Einstellungen bezüglich der Darstellung des Schattens eines Objekts (falls "Schatten" für das Objekt aktiviert ist)
- Einstellungen bezüglich der Darstellung von Elementen während der Mauszeiger über sie fährt (Hover-Effekt)
- Einstellungen bezüglich der Darstellung der Menüs und Symbolleisten

Für die einzelnen Bildern, Elemente und Controls legen Sie darüber hinaus einzeln fest, ob die Vorgaben des Basisdesigns oder die individuellen Einstellungen im Inspektorfenster angewendet werden.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Editor Designs (Seite 66)

Standarddesign verwenden (Seite 73)

Basisdesign der Bedienoberfläche eines Bediengeräts festlegen (Seite 71)

Designvorlagen auf Bilder anwenden (Seite 71)

Designvorlagen auf Basisobjekte und Elementen anwenden (Seite 72)

Designvorlagen auf Controls anwenden (Seite 72)

1.1.9.7 Basisdesign der Bedienoberfläche eines Bediengeräts festlegen

Vorgehen

Um das Basisdesign der Bedienoberfläche eines Bediengeräts festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die "Runtime-Einstellungen" des Bediengeräts.
2. Legen Sie unter "Allgemein > Einstellungen" fest, ob das jeweilige Standarddesign oder ein anderes Design als Basisdesign verwendet wird.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Design anwenden (Seite 70)

Designvorlagen auf Bilder anwenden (Seite 71)

Designvorlagen auf Basisobjekte und Elementen anwenden (Seite 72)

Designvorlagen auf Controls anwenden (Seite 72)

1.1.9.8 Designvorlagen auf Bilder anwenden

Vorgehen

Um die Designvorlagen für die Darstellung von Bildern auf ein Bild anzuwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie das gewünschte Bild im Editor "Bilder".
2. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Design > Einstellungen > Verwende Farbschema".

Ergebnis

Die Einstellungen des Designs werden auf das Bild angewendet.

Alle Einstellungen, die nicht durch das Design vorgegeben sind, sind aktiviert und können geändert werden.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Basisdesign der Bedienoberfläche eines Bediengeräts festlegen (Seite 71)

Design anwenden (Seite 70)

Designvorlagen auf Basisobjekte und Elementen anwenden (Seite 72)

Designvorlagen auf Controls anwenden (Seite 72)

1.1.9.9 Designvorlagen auf Basisobjekte und Elementen anwenden

Vorgehen

Um die Designvorgaben für die Darstellung von Basiselementen und Elementen anzuwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Bild im Editor "Bilder".
2. Klicken Sie im Arbeitsbereich auf das gewünschte Objekt.
3. Wählen Sie im Inspektorfenster die Einstellungen der Gruppe "Eigenschaften > Eigenschaften > Designs" an.
4. Um das Objekt mit einem Schatten darzustellen, aktivieren Sie "Schatten".
5. Um die Farbvorgaben aus dem Design anzuwenden, aktivieren Sie "Verwende Farbeschema".

Ergebnis

Die Einstellungen des Designs werden auf das Objekt angewendet. Im Inspektorfenster sind die Felder zum Einstellen von Hintergrundfarbe, Rahmen etc. deaktiviert. Alle Einstellungen, die nicht durch das Design vorgegeben sind, sind aktiviert und können geändert werden.

Wenn im Design die Option "Schatten" deaktiviert ist, wird das Objekt im Bild ohne Schatten angezeigt - auch wenn in den Eigenschaften des Objekts ein "Schatten" aktiviert ist.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Design anwenden (Seite 70)

Basisdesign der Bedienoberfläche eines Bediengeräts festlegen (Seite 71)

Designvorlagen auf Bilder anwenden (Seite 71)

Designvorlagen auf Controls anwenden (Seite 72)

1.1.9.10 Designvorlagen auf Controls anwenden

Vorgehen

Um die Designvorlagen für die Darstellung von Controls anzuwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Bild im Editor "Bilder".
2. Klicken Sie im Arbeitsbereich auf das gewünschte Objekt.
3. Deaktivieren Sie "Eigenschaften > Eigenschaften > Gestaltung > Stil > Projekteinstellungen für Design verwenden".
4. Wählen Sie in der Auswahlliste das gewünschte Design.

Das Control wird in dem im Design festgelegten Stil dargestellt.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Design anwenden (Seite 70)

Basisdesign der Bedienoberfläche eines Bediengeräts festlegen (Seite 71)

Designvorlagen auf Bilder anwenden (Seite 71)

Designvorlagen auf Basisobjekte und Elementen anwenden (Seite 72)

1.1.9.11 Standarddesign verwenden

Einleitung

Ein Design ist als Standarddesign festgelegt. Wenn Sie die Benutzeroberflächen mehrerer Bediengeräte im einheitlichen Layout gestalten wollen, legen Sie das gewünschte Design als Standarddesign fest und weisen den Bediengeräten in den Runtime-Einstellungen "Standarddesign" als Basisdesign zu.

Wenn Sie das Layout der Bilder und Objekte ändern wollen, passen Sie das Standarddesign an Ihre Anforderungen an oder konfigurieren ein anderes Design als Standarddesign. Letzteres ist der eigentliche Vorteil des Arbeitens mit einem Standarddesign: Sie können quasi per Mausklick das Layout für die Bilder und Objekte aller Bediengeräte wechseln.

Standarddesign konfigurieren

Gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "Designs".
2. Selektieren Sie im Arbeitsbereich in der Liste der Designs in der Spalte "Standard" das gewünschte Design.
   Oder:
   Zeigen Sie im Arbeitsbereich eine Vorschau des gewünschte Designs an und selektieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Design > Standard".
3. Passen Sie das Design gegebenenfalls an Ihre Anforderungen an.

Standarddesign verwenden

Die Einstellungen des Standarddesigns sind für die Objekte wirksam, die folgende Bedingungen erfüllen:

- Für das Bediengerät ist "Runtime-Einstellungen > Allgemein > Einstellungen > Standard-Design verwenden" aktiviert.
- Für Bilder, Basiselemente und Elemente: Im Inspektorfenster ist "Eigenschaften > Eigenschaften> Design > Einstellungen > Verwende Farbeschema" aktiviert.
- Für Controls: Im Inspektorfenster ist "Eigenschaften > Eigenschaften> Gestaltung > Stil > Design verwenden" aktiviert.

Siehe auch

Grundlagen zum Arbeiten mit Designs (Seite 66)

Editor Designs (Seite 66)

Design anwenden (Seite 70)

1.1.10 Arbeiten mit Stilen und Formatvorlagen

1.1.10.1 Grundlagen zum Arbeiten mit Stilen

Einleitung

Der Stileditor ist ein globaler Editor. Im Stileditor definieren Sie ein einheitliches Aussehen für Anzeige- und Bedienelemente. Damit harmonisieren Sie die Darstellung der Objekte in Runtime. Der Stileditor bietet bereits vorgefertigte Stile. Weitere vorgefertigte Stile zum Herunterladen finden Sie in Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/de/91174767).

Sie wählen aus einem der vordefinierten Stile oder erstellen einen eigenen Stil. Sie verwenden den Stileditor unabhängig von Geräten und Projekten.

Ansichten im Stileditor

Im Stileditor werden die Objekte nach ihren wichtigsten visuellen Merkmalen gruppiert.

Folgende Gruppen sind vorhanden:

- Schaltflächen, Linien und Polygon basierte Objekte,
- Text basierte Objekte,
- Diagramm basierte Objekte, z. B. Kurvenanzeige,
- Grafik oder Wert basierte Objekte, z. B. Wertetabelle.

Stilelemente

In einem Stilelement definieren Sie das Aussehen eines Bildobjekts innerhalb von einem Stil. Sie können in einem Stil mehrere Objekte des gleichen Typs unterschiedlich gestalten. Sie definieren z. B. unterschiedliche Stilelemente für einfache Schaltflächen und Navigationsschaltflächen.

Sie erstellen und verwalten Stilelemente im Stileditor.

Formatvorlagen

Mit Formatvorlagen konfigurieren Sie gleiche Eigenschaften für eine Gruppe von Objekten. In einer Formatvorlage können Sie, z. B. Breite und Hintergrundfarbe von Tabellenüberschriften aller tabellenbasierten Objekte definieren. Formatvorlagen sind nach enthaltenen Eigenschaften in vordefinierte Typen unterteilt, z. B. Schaltflächen, Grenzen, Tabellenüberschriften.

Mit Formatvorlagen definieren Sie die Gestaltung gemeinsamer Eigenschaften von Objekten. Einige besondere Eigenschaften von Objekten sind in Formatvorlagen nicht enthalten. Um alle Objekte eines Projekts konsistent visuell zu gestalten, nutzen Sie Stile.

Stile und Formatvorlagen verwalten

Sie erstellen und bearbeiten Stile und Formatvorlagen in der Projektbibliothek. Um in umfangreichen Bibliotheken einen besseren Überblick über die Typen zu bekommen, legen Sie Stile und Formatvorlagen in Ordnern mit aussagekräftigen Namen ab. Vergeben Sie auch den von Ihnen erstellten Stilen und Formatvorlagen sinnvolle Namen.

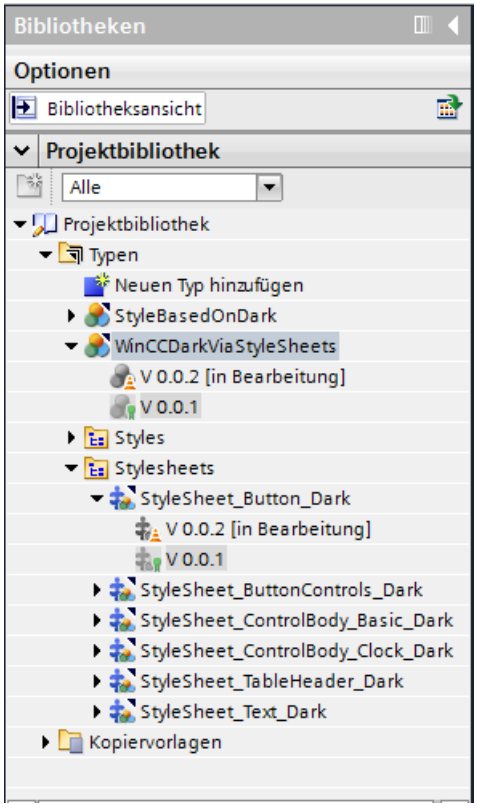

Unterstützung von Stilen in Bildbausteinen

Bildbaustein-Instanzen können Stile und Stilelemente zugewiesen werden.

Es besteht aber keine direkte Verbindung zwischen den Bildbaustein-Typen und Stilen, deshalb ist die Auswahl und die Vorschau eines Stilelements im Bildbaustein-Editor nicht

möglich. Um einem Bildbaustein ein Stilelement zuzuweisen, geben Sie den Namen des Stilelements unter „Gestaltung Stilelement“ ein.

In der Instanz eines Bildbausteins wird das konfigurierte Stilelement verwendet, solange der Name des Stilelements im verwendeten Stil existiert. Achten Sie beim Eingeben des Namens auf die korrekte Schreibweise.

Siehe auch

Vordefinierte Stile anzeigen lassen (Seite 76)

Stil definieren (Seite 77)

Stile verwalten (Seite 79)

Stilelemente verwenden (Seite 82)

1.1.10.2 Vordefinierte Stile anzeigen lassen

Einleitung

Im Editor "Stile" sehen Sie vordefinierte Stile und erstellen eigene Stile für die Darstellung der Anzeige- und Bedienobjekte in Runtime.

Editor "Stile" öffnen

1. Doppelklicken Sie in der Projektnavigation "Gemeinsame Daten".
2. Klicken Sie auf den Editor "Stile". Der Editor wird geöffnet.

Hinweis
Stileditor in der Projektnavigation

Der Stileditor ist nur in der Projektnavigation sichtbar, wenn im Projekt ein Gerät angelegt ist, das Stile unterstützt, z. B. Comfort Panels.

3. Im Arbeitsbereich des Editors "Stile" werden die Standardstile angezeigt:

- WinCC Light
- WinCC Dark
- WinCC Fresh
- WinCC Wireframe.

Ergebnis

Im Arbeitsbereich werden die Einstellungen für die Gruppen und einzelnen Objekte angezeigt.

In den nächsten Schritten weisen Sie vordefinierte Stile zu oder definieren eigene Stile.

Vordefinierte Stile sind schreibgeschützt und können nicht geändert werden.

Siehe auch

Grundlagen zum Arbeiten mit Stilen (Seite 74)

1.1.10.3 Stil definieren

Einleitung

Im Editor "Stile" sehen Sie vordefinierte Stile. Vordefinierte Stile sind schreibgeschützt und können nicht verändert werden.

Des Weiteren haben Sie die Möglichkeit eigene Stile zu definieren und auf einzelne Objekte, Gruppen aber auch Projekte anzuwenden.

Stil definieren

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten".
2. Öffnen Sie den Editor "Stile".
3. Klicken Sie im Bereich "Stile" auf "Hinzufügen".
   Ein neuer Stil wird angelegt.
   Die Bibliotheksansicht wird geöffnet. Im Ordner "Typen" wird die Version des Typs angezeigt. Die Version 0.0.1 ist im Zustand "in Bearbeitung".

4. Im Inspektorfenster wählen Sie die Farben, Rahmen und weitere Einstellungen für einzelne Objekte.

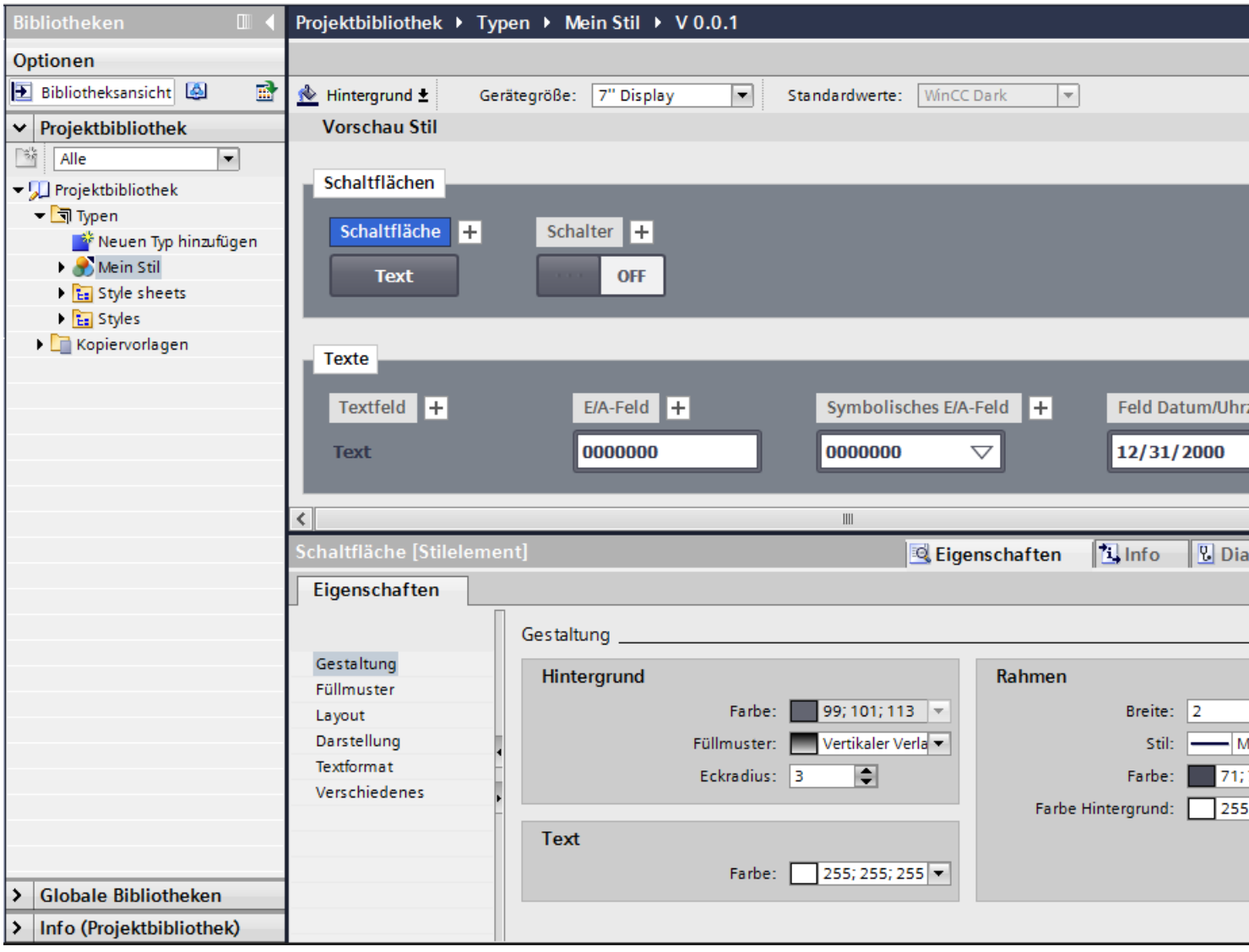

5. Um neuen Stil wirksam zu machen selektieren Sie die aktuelle Version "in Bearbeitung" und wählen Sie im Kontextmenü "Version freigeben".
   Der neue Stil wird im Editor "Stile" angezeigt.

Ergebnis

Sie haben einen neuen Stil hinzugefügt, den Sie auf Objekte oder Projekte anwenden können.

Stil definieren basierend auf einem vordefinierten Stil

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten".
2. Öffnen Sie den Editor "Stile".

3. Wählen Sie im Arbeitsbereich einen vordefinierten Stil.
4. Wählen Sie im Kontextmenü des ausgewählten Stils "Typ duplizieren".
   Der Dialog "Typ duplizieren" wird geöffnet.
5. Überschreiben Sie den vorgeschlagenen Namen mit einem aussagekräftigen Namen.
6. Geben Sie in der Spalte "Kommentar" eine Beschreibung des Stils ein, z. B. den Verwendungszweck.
7. Bestätigen Sie ihre Eingaben mit "OK".
8. Wählen Sie den erstellten Typ in der Projektbibliothek.
9. Wählen Sie im Kontextmenü "neuen Typ editieren".
   Die Bibliotheksansicht wird geöffnet. Im Ordner "Typen" wird die Version des Typs angezeigt. Die Version 0.0.2 ist im Zustand "in Bearbeitung".
10. Konfigurieren Sie den Typ wie gewünscht.
11. Wählen Sie im Kontextmenü der bearbeiteten Version "Version freigeben".

Ergebnis

Sie haben einen neuen Stil konfiguriert, der auf dem vordefinierten Stil von WinCC basiert.

Siehe auch

Grundlagen zum Arbeiten mit Stilen (Seite 74)

Zustand von Typ-Versionen (Seite 2040)

1.1.10.4 Stile verwalten

Stil löschen

Stile, die Sie selbst definiert haben, können Sie löschen.

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten".
2. Öffnen Sie den Editor "Stil".
3. Selektieren Sie im Arbeitsbereich den Stil, den Sie löschen möchten.

   Hinweis

   Die vordefinierte Stile von WinCC können Sie nicht löschen.

4. Selektieren Sie im Kontextmenü den Eintrag "Löschen"

Der Stil wird gelöscht.

Stil bearbeiten

Sie können die Stile, die Sie selbst definiert haben, bearbeiten.

1. Öffnen Sie die Projektbibliothek.
2. Wählen Sie die freigegebene Version des Stils, den sie ändern möchten.
3. Wählen Sie im Kontextmenü "Typ bearbeiten".
   Eine neue Version "in Bearbeitung" wird erstellt.
4. Nehmen Sie im Inspektorfenster die gewünschten Anpassungen vor.
5. Um die Änderungen wirksam zu machen, wählen sie im Kontextmenü der bearbeiteten Version "Version freigeben".
   Die vorgenommenen Änderungen werden wirksam.

Standardstil für ein Projekt festlegen

Um in einem Projekt eine visuelle Einheitlichkeit aller Anzeigeobjekte zu erreichen, weisen Sie einem Projekt einen Standardstil zu.

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten".
2. Öffnen Sie den Editor "Stil".
3. Aktivieren Sie im Arbeitsbereich einen Stil.

Der aktivierte Stil wird als Standardstil im Projekt verwendet. Wenn Sie ein Gerät im Projekt hinzufügen, verwendet das Gerät den Standardstil.

Stil für ein bestimmtes Gerät ändern

Innerhalb eines Projekts ändern Sie den Stil für ein bestimmtes Gerät.

1. Öffnen Sie den Editor "Runtime-Einstellungen" des Bediengeräts.
2. Selektieren Sie unter "Allgemein" einen Gerätestil.

Der selektierte Stil wird auf das Gerät angewendet.

Wenn der Gerätestil nicht zugewiesen ist, verwendet das Gerät den Standardstil des Projekts.

Siehe auch

Grundlagen zum Arbeiten mit Stilen (Seite 74)

1.1.10.5 Schriftgröße an Gerätegröße anpassen

Einleitung

In Stileditor definieren Sie die Größe des Geräts, auf dessen Objekte Sie einen Stil anwenden. Somit legen Sie geräteabhängige Schriftgrößen in einem Stil fest. Mit der Auswahl einer bestimmten Referenzgröße werden alle Schriftgrößen an die Gerätegröße automatisch angepasst.

Folgende Tabelle zeigt die Gerätegrößen und die Referenzschriftgrößen in Pixel:

Gerätegröße	Schriftgröße in Pixel
4" Display	15 px
7" Display	17 px
9" Display	15 px
12" Display	19 px
15" Display	19 px
19" Display	17 px
22" Display	21 px
WinCC RT Advanced	17 px

Automatische Anpassung von Schriftgrößen aktivieren

1. Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen".
2. Klicken Sie auf "Allgemein".
3. Aktivieren Sie die Option "Schriftgröße im Stil anpassen" unter "Allgemein > Bild".

Automatische Anpassung von Schriftgrößen an die Gerätegröße in einem Stil wurde aktiviert.

Schriftgröße in einem Gerätestil anpassen

1. Öffnen Sie den entsprechenden Stil zur Bearbeitung in der Projektbibliothek.
2. Wählen Sie die Referenzgröße des Bediengeräts im Arbeitsbereich.

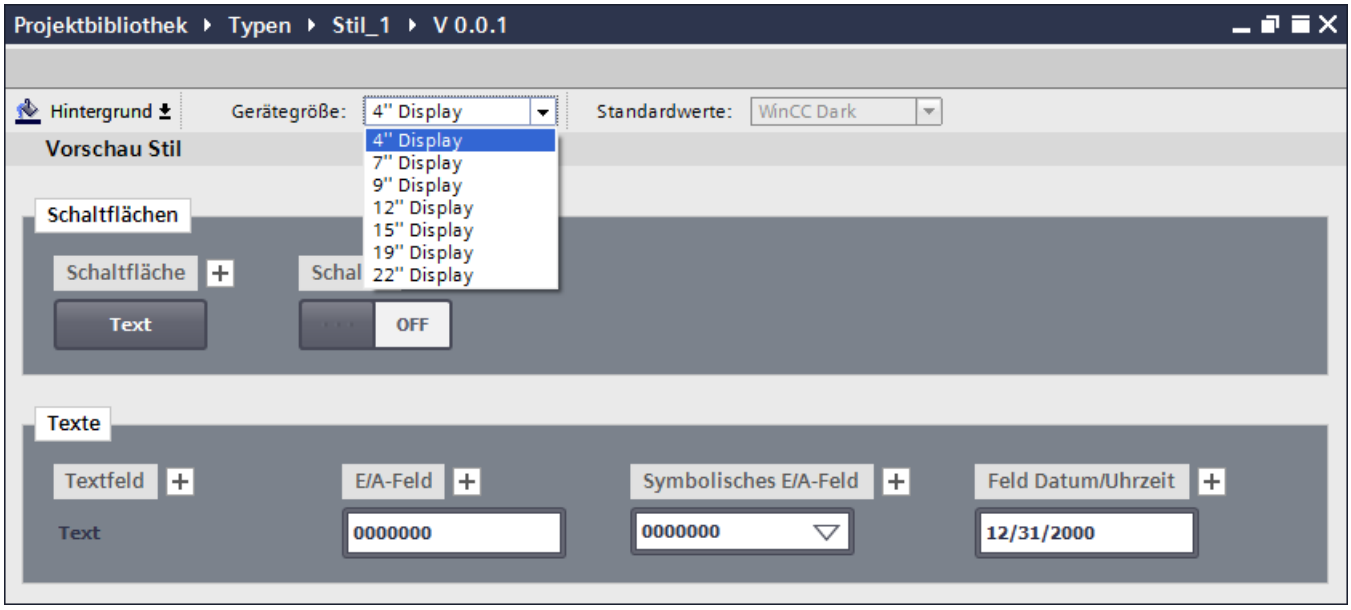

Alle Schriftgrößen im Stil wurden automatisch an die gewählte Gerätegröße angepasst.

1.1.10.6 Stilelemente verwenden

Einleitung

Mithilfe von Stilelementen ändern Sie den Stil eines Bildobjekts ohne den gesamten Gerätestil zu verändern. Sie können z. B. Typen von Schaltflächen mithilfe von verschiedenen Stilelementen definieren, die in einem oder mehreren Geräten einheitliches Aussehen erhalten.

Stilelemente definieren Sie im Stileditor.

Neues Stilelement hinzufügen

1. Öffnen Sie die Projektbibliothek.
2. Wählen Sie die freigegebene Version des Stils, den sie ändern möchten.
3. Wählen Sie im Kontextmenü "Typ bearbeiten".
   Eine neue Version "in Bearbeitung" wird erstellt.
4. Wählen Sie das Objekt, welches Sie in einem anderen Stil verwenden möchten.

5. Klicken Sie auf das Plus-Zeichen neben dem Objektnamen.

Ein neues Stilelement wurde angelegt.
Alternativ wählen Sie "Stilelement duplizieren" im Kontextmenü des Objekts.

6. Nehmen Sie im Inspektorfenster die gewünschten Anpassungen vor.
7. Unter "Eigenschaften > Verschiedenes > Objekt > Name " vergeben Sie dem neuen Stilelement einen aussagekräftigen Namen.
8. Geben Sie die Version des Stils frei.

Stilelement löschen

Stilelemente, die Sie selbst definiert haben, können Sie löschen.

1. Öffnen Sie die Projektbibliothek.
2. Wählen Sie den Stil, welches das Stilelement enthält.
3. Öffnen Sie die Version "in Bearbeitung".
4. Selektieren Sie im Arbeitsbereich den Stilelement, den Sie löschen möchten.
5. Selektieren Sie im Kontextmenü den Eintrag "Löschen".

Hinweis

Vordefinierte Stilelemente von WinCC können nicht umbenannt oder gelöscht werden.

Der Stilelement wird gelöscht.

Stilelement für ein Objekt festlegen

Wenn Sie für ein Objekt zusätzliche Stilelemente im Gerätestil konfiguriert haben, können Sie für ein Bildobjekt das vordefinierte Stilelement verwenden.

1. Öffnen Sie "Eigenschaften > Stile/Designs > Einstellungen".
2. Aktivieren Sie die Option "Stil/Design verwenden".
3. Wählen Sie unter "Gestaltung Stilelement" das vordefinierte Stilement.

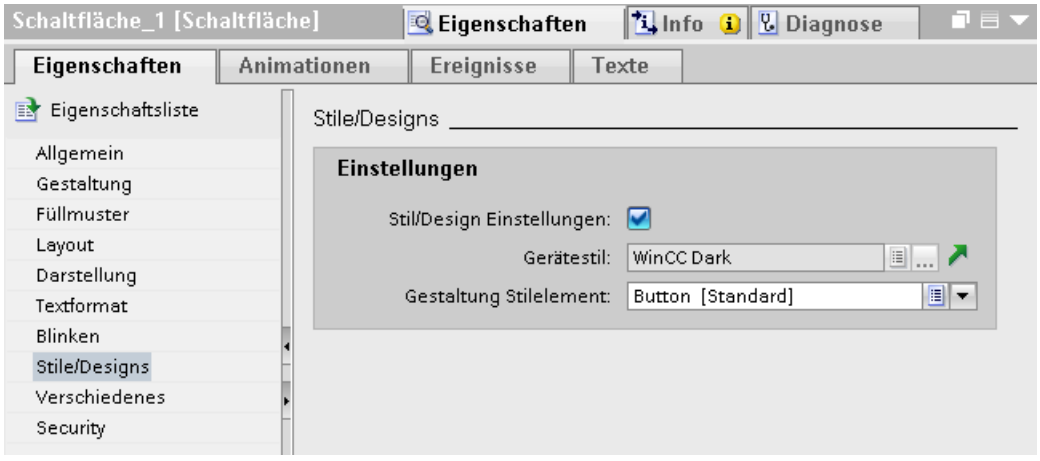

Das Objekt wird als ausgewähltes Stilelement dargestellt.

1.1.10.7 Grundlagen zum Arbeiten mit Formatvorlagen

Einleitung

Einige Bildobjekte besitzen gleiche Eigenschaften, die auf ähnliche Weise konfiguriert werden. Diese allgemeine Eigenschaften sind in Formatvorlagen gruppiert. Mithilfe von Formatvorlagen können Sie mehrere Eigenschaften gleichzeitig und konsistent für mehrere Objekte konfigurieren.

Typen von Formatvorlagen

Formatvorlagen sind in mehrere vorgegebene Kategorien unterteilt. Jede Kategorie beinhaltet eine bestimmte Menge von gemeinsamen Eigenschaften, welche verschiedene Objekte besitzen. Bei der Erstellung einer Formatvorlage wählen Sie die gewünschte Kategorie aus. Eine Formatvorlage kann nur auf Objekte angewendet werden, für welche die jeweilige Kategorie definiert ist.

Folgende Tabelle zeigt die verfügbare Kategorien und damit konfigurierbare Objekte.

Kategorie/Objekte	Polygone/ Linien	Text-basierte Objekte	Schaltflächen	Wertebasierte Objekte	Tabellenba-sierte Objekte	Diagramme
Schaltfläche	-	-	X	-	X	X
Objekt Rumpf	X	-	-	X	X	X
Diagramm Rumpf	-	-	-	-	-	X

Kategorie/Objekte	Polygone/ Linien	Text-basierte Objekte	Schaltflächen	Wertebasierte Objekte	Tabellenba-sierte Objekte	Diagramme
Fokus und Aus-wahl	-	X	X	-	X	X
Grenzen	-	X	-	X	-	-
Beschriftungen	-	-	-	X	-	-
Skala	-	-	-	X	-	-
Rumpf Tabelle	-	-	-	-	X	X
Überschrift Tabelle	-	-	-	-	X	X
Textfeld	-	X	-	-	-	-

Verwendung von Formatvorlagen

Während der Bearbeitung eines Stils wenden Sie eine Formatvorlage auf ein Stilelement an. Die Eigenschaften, die in der Formatvorlage konfiguriert sind, werden auf das Stilelement übertragen.

Um eine Formatvorlage auf ein Stilelement anzuwenden, ziehen Sie die gewünschte Formatvorlage aus dem Task Card per Drag&Drop auf das Stilelement im Arbeitsbereich.

Verwaltung von Formatvorlagen

Formatvorlagen sind Bibliothekstypen. Sie verwalten Formatvorlagen in der Projektbibliothek. Kategorien von Formatvorlagen finden Sie während der Bearbeitung eines Stils in der Projektbibliothek in der Task Card "Formatvorlagen > Kategorien".

In der Task Card "Formatvorlagen > Visuelle Attribute" stehen die für einen Formatvorlagentyp verfügbaren Formatvorlagen zur Verfügung. In der Task Card finden Sie sowohl die vordefinierte als auch von Ihnen erstellte und freigegebene Formatvorlagen.

1.1.10.8 Neue Formatvorlage erstellen

Einleitung

In einer Formatvorlage konfigurieren Sie die Eigenschaften für mehrere Objekte eines ähnlichen Typs.

Neue Formatvorlage erstellen

1. Öffnen Sie die Projektbibliothek.
2. Klicken Sie auf "Neuen typ hinzufügen".
   Dialog "Neuen Typ hinzufügen" öffnet sich.
3. Wählen Sie im Dialog die Option "HMI Formatvorlage".

4. Wählen Sie den Typ der Formatvorlage.

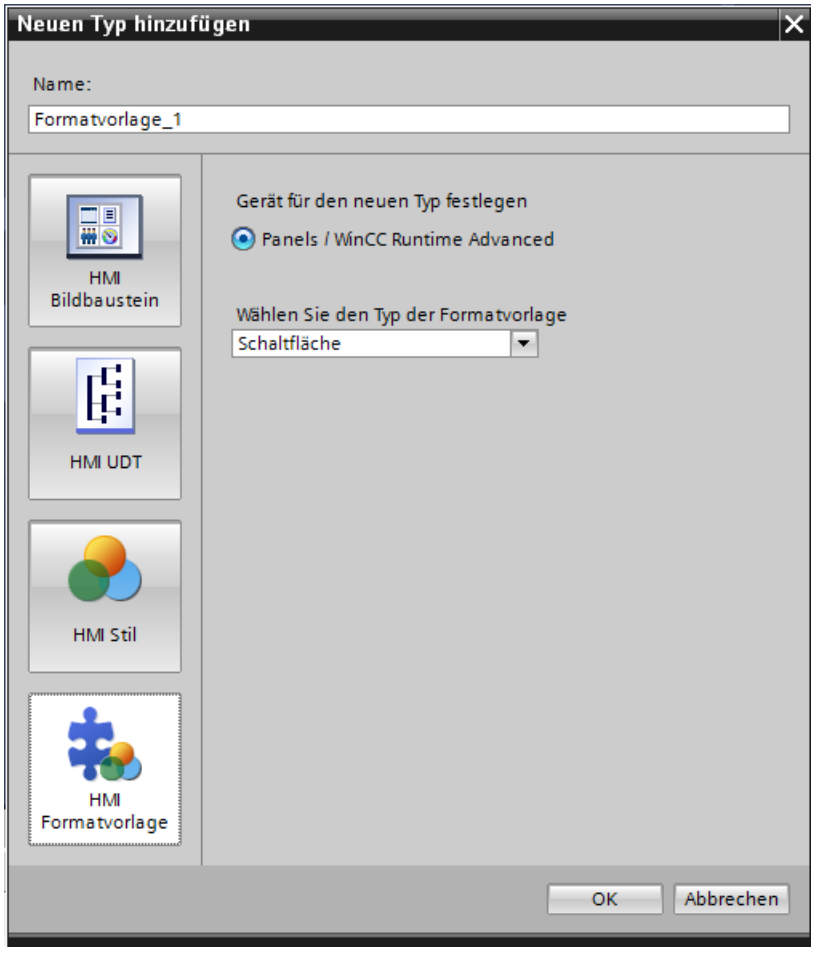

5. Vergeben Sie für die Formatvorlage einen aussagekräftigen Namen.
6. Bestätigen Sie Ihre Eingaben mit "OK".
   Neue Formatvorlage wurde angelegt.

Neu Formatvorlage basiert auf einem Stil erstellen

Um während der Bearbeitung eines Stils die Eigenschaften für ähnliche Bildobjekte automatisch zu konfigurieren, erstellen Sie eine neue Formatvorlage basierend auf bereits konfigurierten Eigenschaften eines Stilelements.

1. Markieren Sie das Stilelement, anhand dessen Sie eine Formatvorlage erstellen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Neue Formatvorlage erstellen".
3. Wählen Sie im Untermenü die Kategorie für die Formatvorlage.
   Dialog "Typ hinzufügen" öffnet sich.

4. Vergeben Sie für die Formatvorlage einen aussagekräftigen Namen.
5. Bestätigen Sie mit "OK".
   Die neue Formatvorlage, die auf einem Stil basiert, wurde erstellt und in die Task Card "Formatvorlagen > Visuelle Attribute" hinzugefügt.
   Die Eigenschaften in der Formatvorlage wurden aus dem markierten Stilelement übernommen. Die Eigenschaften der Formatvorlage, die der markierte Stil nicht beinhaltet, wurden mit Standardwerten belegt.

1.1.10.9 Formatvorlagen verwalten

Einleitung

Sie bearbeiten, kopieren und löschen die Formatvorlagen in der Projektbibliothek.

Außerdem können Sie Formatvorlagen über die Befehle im Kontextmenü in der Task Card "Formatvorlagen > Kategorien" öffnen, duplizieren, neu erstellen oder löschen.

Formatvorlage bearbeiten

Sie bearbeiten die Formatvorlagen, die Sie selbst erstellt haben.

1. Öffnen Sie die Projektbibliothek.
2. Wählen Sie die freigegebene Version der Formatvorlage, die Sie ändern möchten.
3. Wählen Sie im Kontextmenü "Typ bearbeiten".
   Eine neue Version "in Bearbeitung" wird erstellt.
4. Nehmen Sie im Inspektorfenster die gewünschten Anpassungen vor.
5. Um die Änderungen wirksam zu machen, wählen sie im Kontextmenü der bearbeiteten Version "Version freigeben".
   Die vorgenommenen Änderungen werden wirksam.

Formatvorlage löschen

Formatvorlagen, die Sie selbst definiert haben, können Sie löschen.

1. Öffnen Sie die Projektbibliothek.
2. Selektieren Sie in der Projektbibliothek die Formatvorlage, die Sie löschen möchten.

   Hinweis

   Die vordefinierte Formatvorlagen von WinCC können Sie nicht löschen.

   Hinweis

   Die Formatvorlage, die einen Typ "in Bearbeitung" enthält, kann nicht gelöscht werden. Um die Formatvorlage zu löschen, geben Sie zuerst alle Versionen der Formatvorlage frei.

3. Selektieren Sie im Kontextmenü den Eintrag "Löschen".

Die Formatvorlage wird gelöscht.

Formatvorlage kopieren

1. Öffnen Sie die Projektbibliothek.
2. Wählen Sie Formatvorlage, die Sie duplizieren möchten.
3. Wählen Sie im Kontextmenü "Typ duplizieren".
4. Überschreiben Sie den vorgeschlagenen Namen mit einem aussagekräftigen Namen.
5. Geben Sie in der Spalte "Kommentar" eine Beschreibung für die Formatvorlage, z. B. den Verwendungszweck.
6. Bestätigen Sie ihre Eingaben mit "OK".
   Die gewünschte Formatvorlage wurde kopiert und im gleichen Ordner abgelegt.

1.1.10.10 Formatvorlagen verwenden

Einleitung

Sie verwenden eine Formatvorlage während der Bearbeitung eines Stils im Stileditor. Formatvorlagen sind objektabhängig und werden nur auf Stilelemente angewendet, mit denen sie kompatibel sind.

Nur eine freigegebene Version einer Formatvorlage kann auf ein Stilelement angewendet werden.

Sie haben folgende Möglichkeiten um eine Formatvorlage anzuwenden:

- Um das gewählte Stilelement zu gestalten, ziehen Sie die Formatvorlage auf ein Stilelement im Stileditor.
- Um mehrere Stilelemente einer Gruppe zu gestalten, ziehen Sie die Formatvorlage auf eine Gruppe von Stilelementen.
- Um alle Stilelemente des Stils zu gestalten, die mit der Formatvorlage kompatibel sind, ziehen Sie die Formatvorlage auf den leeren Bereich im Stileditor.

Voraussetzung

- Ein Stil ist angelegt.
- Eine Version des Stils ist "in Bearbeitung" geöffnet.

Formatvorlage auf ein Stilelement anwenden

1. Markieren Sie die Kategorie der Formatvorlage in der Task Card "Formatvorlagen > Kategorien".
2. In der Task Card "Formatvorlagen > Visuelle Attribute" markieren Sie die gewünschte Formatvorlage.
3. Ziehen Sie die markierte Formatvorlage per Drag&Drop auf ein Stilelement.

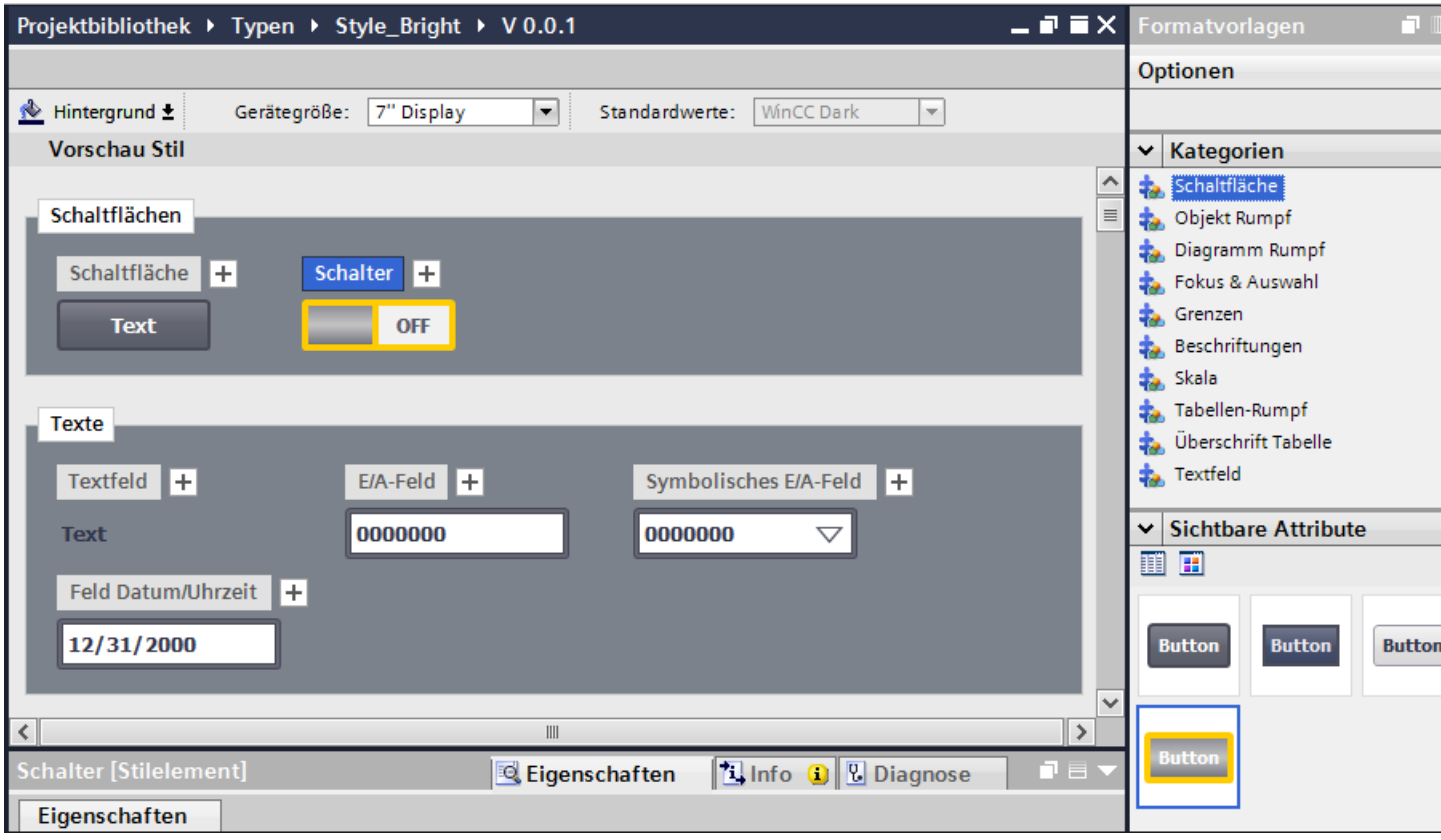

Wenn Sie die Formatvorlage auf das Stilelement fallen lassen, übernimmt das Stilelement die Einstellungen aus der Formatvorlage.

Hinweis

Es besteht keine Verbindung zwischen dem Stilelement und der verwendeten Formatvorlage. Wenn Sie die Formatvorlage verändern, werden die Änderungen im Stilelement nicht automatisch übernommen. Um Änderungen an der Formatvorlage im Stilelement wirksam zu machen, wenden Sie die Formatvorlage erneut an.

Hinweis

Formatvorlagen sind mit Bildobjekten nicht verlinkt. Durch das Aufräumen oder Versionieren der Bibliothek werden nicht verwendete oder alte Versionen von Formatvorlagen gelöscht.

Ergebnis

Sie haben eine Formatvorlage auf ein Stilelement in einem Stil angewendet.

1.2 Arbeiten mit Objekten und Objektgruppen

1.2.1 Arbeiten mit Objekten

1.2.1.1 Übersicht über die Objekte

Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Die Task Card "Werkzeuge" enthält alle Objekte, die für das Bediengerät verfügbar sind.

Das Werkzeugfenster blenden Sie im Menü "Ansicht" ein, indem Sie die Option "Task Card" aktivieren.

Abhängig vom aktuell geöffneten Editor enthält das Werkzeugfenster unterschiedliche Paletten. Wenn der Editor "Bilder" geöffnet ist, werden im Werkzeugfenster folgende Paletten bereitgestellt:

- "Basisobjekte"
  Zu den Basisobjekten gehören grundlegende grafische Objekte, z. B. "Linie", "Kreis", "Textfeld" oder "Grafikanzeige.
- "Elemente"
  Zu den Elementen gehören grundlegende Bedienelemente, z. B. "E/A-Feld", "Schaltfläche" oder "Zeigerinstrument".
- "Controls"
  Die Controls haben einen erweiterten Funktionsumfang. Sie stellen auch Prozessabläufe dynamisch dar, z. B. "Kurvenanzeige" und "Rezepturanzeige".
- "Eigene Controls"
  In diese Palette fügen Sie ActiveX-Controls und einfache .Net-Controls zum Werkzeugfenster hinzu. Von dieser Palette binden Sie die ActiveX-Controls in Ihr Projekt ein. Es gelten folgende Einschränkungen:
  - Nur die ActiveX-Controls sind verfügbar, die im Betriebssystem des Projektierungsrechners registriert sind.
  - Nur die einfachen .Net-Controls werden vom TIA Portal unterstützt. XAML Controls, z. B. WPF (Windows Presentation Foundation) Controls werden nicht unterstützt.

- "Grafiken"
  Grafiken sind thematisch in Form eines Verzeichnisbaums gegliedert. Die verschiedenen Ordner enthalten z. B. folgende grafische Darstellungen:
  - Maschinen- und Anlagenbereiche
  - Messgeräte
  - Bedienelemente

  Sie können Verknüpfungen zu Ihren eigenen Grafikordnern erstellen. Die externen Grafiken liegen in diesen Ordnern und Unterordnern. Sie werden im Werkzeugfenster angezeigt und über die Verknüpfung in das Projekt eingebunden.
- Task Card "Bibliotheken"
  Zusätzlich zu den Anzeige- und Bedienobjekten stehen Ihnen die Bibliotheksobjekte zur Verfügung. Sie befinden sich innerhalb der Paletten der Task Card "Bibliotheken". Eine Bibliothek enthält fertig konfigurierte Objekte, z. B. Grafiken von Rohren, Pumpen oder vorkonfigurierten Schaltflächen. Sie können Bibliotheksobjekte auch mehrfach in Ihr Projekt einbinden, ohne sie jeweils neu zu konfigurieren.
  Mit WinCC werden Bibliotheken mitgeliefert z. B. "HMI Buttons & Switches".
  Sie können auch benutzerdefinierte Objekte und Bildbausteine in eigenen Bibliotheken hinterlegen. Bildbausteine sind Objekte, die Sie aus vorhandenen Bildobjekten zusammenstellen und für die Sie projektierbare Eigenschaften festlegen.

Basisobjekte

Symbol	Objekt	Hinweise
	"Linie"	-
	"Polygonzug"	Der Polygonzug ist ein offenes Objekt. Selbst wenn Startpunkt und Endpunkt die gleichen Koordinaten haben, können Sie die umschlossene Fläche nicht füllen. Wenn Sie ein Polygon füllen wollen, wählen Sie das Objekt "Polygon".
	"Polygon"	-
	"Ellipse"	-
	"Kreis"	-
	"Rechteck"	-
	"Kreisbogen"	Der Kreisbogen ist ein offenes Objekt. Selbst wenn Startpunkt und Endpunkt die gleichen Koordinaten haben, können Sie die umschlossene Fläche nicht füllen. Wenn Sie einen Kreis füllen wollen, wählen Sie das Objekt "Kreis" oder "Kreissegment".
	"Ellipsenbogen"	Ein Ellipsenbogen ist ein offenes Objekt. Selbst wenn Startpunkt und Endpunkt die gleichen Koordinaten haben, können Sie die umschlossene Fläche nicht füllen. Wenn Sie eine Ellipse füllen wollen, wählen Sie das Objekt "Ellipse" oder "Ellipsensegment".
	"Kreissegment"	-
	"Ellipsensegment"	-
	"Verbinder"	Verbindet Objekte mit Linien.

Symbol	Objekt	Hinweise
	"Textfeld"	Stellt einzeiligen oder mehrzeiligen Text dar. Schrift und Darstellung sind einstellbar.
	"Grafikanzeige"	Zeigt Grafiken aus externen Grafikprogrammen an und fügt OLE-Objekte ein. Folgende Grafikformate können verwendet werden: "*.emf", "*.wmf", "*.dib", "*.bmp", "*.jpg", "*.jpeg", "*.gif" , "*.tif" und "*.svg".
	"Rohr"	Stellt ein mehrfach geknicktes durchgehendes Rohrs in einer Rohranlage dar.
	"Doppel-T-Stück"	Stellt eine Rohrkreuzung in einer Rohranlage dar.
	"T-Stück"	Stellt eine T-förmige Rohrverbindung dar.
	"Rohrbogen"	Stellt ein gebogenes Rohrstück dar.

Elemente

Symbol	Objekt	Hinweise
	"E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Sie legen Grenzwerte für die im E/A-Feld dargestellten Variablenwerte fest. Für eine unsichtbare Eingabe des Bedieners in Runtime projektieren Sie "Verdeckte Eingabe".
	"Editierbares Textfeld"	Stellt im Bild mehrere Zeilen Text in einem Rechteck dar.
	"Listenfeld"	Stellt eine Auswahl von mehreren Listeneinträgen dar.
	"Kombinationsfeld"	Stellt eine Auswahl von mehreren Einträgen in einer Aufklappliste dar.
	"Schaltfläche"	Führt abhängig von der Projektierung eine Funktionsliste oder ein Skript aus.
	"Rundschaltfläche"	Bedient Prozesse, z. B. Meldungen bestätigen.
	"Symbolisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird ein Text aus einer Textliste angezeigt.
	"Grafisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird eine Grafik aus einer Grafikliste angezeigt.
	"Balken"	Stellt einen Wert aus der Steuerung in Form einer Säule dar, die mit einer Skala versehen ist.
	"Symbolbibliothek"	Fügt Bildobjekte ein, die auf gleichnamigen Controls basieren.
	"Schieberegler"	Zeigt einen aktuellen Wert an oder gibt einen numerischen Wert ein.
	"Bildlaufleiste"	Gibt einen numerischen Wert in die Steuerung ein. Der Wert ist stufenlos einstellbar.
	"Kontrollkästchen"	Wählt Optionen aus.

Symbol	Objekt	Hinweise
	"Optionsschaltfläche"	Wählt eine Option aus.
	"Zeigerinstrument"	Zeigt numerische Werte an. Die Darstellung des Zeigerinstruments ist einstellbar.
	"Uhr"	Zeigt die Systemzeit analog oder digital an.
	"Speicherplatzanzeige"	Zeigt den vorhandenen Speicherplatz an.

Controls

Symbol	Objekt	Beschreibung
	"Bildfenster"	Zeigt andere Bilder des Objekts an.
	"f(t)-Kurvenanzeige"	Stellt mehrere Kurven mit Werteverläufen aus der Steuerung oder aus einem Archiv dar.
	"f(x)-Kurvenanzeige "	Stellt die Werte einer Variablen als Funktion einer anderen Variablen dar.
	"Tabellenanzeige"	Stellt aktuelle oder archivierte Prozessdaten in einer Tabelle dar.
	"Wertetabelle"	Stellt ausgewertete Daten und Statistiken in einer Tabelle dar.
	"System-Diagnoseanzeige"	Ermöglicht eine Übersicht über alle diagnosefähigen Geräte. Zeigt Fehler in der Anlage an.
	"Benutzeranzeige"	Ermöglicht einem Administrator die Benutzerverwaltung am Bediengerät. Ermöglicht einem Bediener ohne Administratorrechte, sein Kennwort zu ändern.
	"HTML-Browser"	Zeigt HTML-Seiten an.
	"Druckauftrag/Skriptdiagnose"	Zeigt Daten der Applikation Global Script und des Protokollsystems zur Bearbeitung an.
	"Rezepturanzeige"	Dient der Anzeige und Änderung von Rezepturen.
	Media Player	Ermöglicht das Abspielen von Video- oder Audiodateien.
	"Kanaldiagnose-Anzeige"	Ermöglicht dem Bediener die Kontrolle der aktiven Verbindungen.
	"Meldeanzeige"	Zeigt aktuell anstehende Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv an.
	"PLC-Code-Anzeige"[1)]	Zeigt die PLC-Programme, die in den Programmiersprachen KOP oder FUP programmiert worden sind.
	"ProDiag-Übersicht"[2)]	Zeigt den aktuellen Zustand von Operandenüberwachungen.
	"GRAPH-Übersicht"[2)]	Zeigt den aktuellen Programmstatus in einer Ketten- und Einzelschrittansicht an.
	"Kriterienanalyse-Anzeige"[3)]	Zeigt die fehlerhafte Operanden im Programm, die durch die Kriterienanalyse ermittelt wurden.

[1] Verfügbar ab Geräteversion V14.

[2] Verfügbar ab Geräteversion V14.

[3] Verfügbar ab Geräteversion V15.

Hinweis

Abhängig vom Bediengerät, das Sie projektieren, sind einige Objekte aus dem Werkzeugfenster nicht oder nur mit eingeschränkter Funktionsweise verfügbar. Nicht verfügbare Eigenschaften eines Objekts werden deaktiviert dargestellt und sind nicht auswählbar.

Hinweis

Unterstützung der SVG-Grafiken ab TIA Portal Version V15

Die SVG-Grafiken können Sie in folgenden Bildobjekten verwenden:

Bilder (als Hintergrundbild)

runde Schaltflächen

Grafikanzeige

Grafisches E/A-Feld

Graiklisteeintrag

Siehe auch

Übersicht über die Objekte (Seite 95)

Objekt gestalten (Seite 119)

Objekt gestalten (Seite 119)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.1.2 Übersicht über die Objekte

Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Die Task Card "Werkzeuge" enthält alle Objekte, die für das Bediengerät verfügbar sind. Sie blenden die Task Card mit dem Menübefehl "Ansicht" ein, indem Sie die Option "Task Card" aktivieren.

Abhängig vom aktuell geöffneten Editor enthält das Werkzeugfenster unterschiedliche Paletten. Wenn der Editor "Bilder" geöffnet ist, werden im Werkzeugfenster folgende Paletten bereitgestellt:

- "Basisobjekte"
  Zu den Basisobjekten gehören grundlegende grafische Objekte, z. B. "Linie", Kreis", "Textfeld" oder "Grafikanzeige".
- "Elemente"
  Zu den Elementen gehören grundlegende Bedienelemente, z. B. "E/A-Feld", "Schaltfläche" oder "Zeigerinstrument".
- "Controls"
  Die Controls haben einen erweiterten Funktionsumfang. Sie stellen auch Prozessabläufe dynamisch dar, z. B. Kurvenanzeige und Rezepturanzeige.
- "Grafiken"
  Grafiken sind thematisch in Form eines Verzeichnisbaums gegliedert. Die verschiedenen Ordner enthalten z. B. folgende grafische Darstellungen:
  - Maschinen- und Anlagenbereiche
  - Messgeräte
  - Bedienelemente
  - Flaggen
  - Gebäude

  Sie erstellen Verknüpfungen zu Ihren eigenen Grafikordnern. Die externen Grafiken liegen in diesen Ordnern und Unterordnern. Sie werden im Werkzeugfenster angezeigt und über die Verknüpfung in das Projekt eingebunden.
- Task Card "Bibliotheken"
  Zusätzlich zu den Anzeige- und Bedienobjekten stehen Ihnen die Bibliotheksobjekte zur Verfügung. Sie befinden sich innerhalb der Paletten der Task Card "Bibliotheken". Eine Bibliothek enthält fertig konfigurierte Objekte, z. B. Grafiken von Rohren, Pumpen, vorkonfigurierten Schaltflächen. Sie binden Bibliotheksobjekte auch mehrfach in Ihr Projekt ein, ohne sie jeweils neu zu konfigurieren.
  Mit WinCC werden Bibliotheken mitgeliefert z. B. "HMI Buttons & Switches".
  Sie hinterlegen auch benutzerdefinierte Objekte und Bildbausteine in eigenen Bibliotheken. Bildbausteine sind Objekte, die Sie aus vorhandenen Bildobjekten zusammenstellen und für die Sie projektierbare Eigenschaften festlegen.

Hinweis

Bediengeräteabhängigkeit

Abhängig vom Bediengerät, das Sie projektieren, sind einige Objekte aus dem Werkzeugfenster nicht oder nur mit eingeschränkter Funktionsweise verfügbar. Nicht verfügbare Eigenschaften eines Objekts werden deaktiviert dargestellt und sind nicht auswählbar.

Basisobjekte

Symbol	Objekt	Hinweise
	"Linie"	-
	"Polygonzug"	Der Polygonzug ist ein offenes Objekt. Selbst wenn Startpunkt und Endpunkt die gleichen Koordinaten haben, können Sie die umschlossene Fläche nicht füllen. Wenn Sie ein Polygon füllen wollen, wählen Sie das Objekt "Polygon".
	"Polygon"	-
	"Ellipse"	-
	"Kreis"	-
	"Rechteck"	-
A	"Textfeld"	Stellt einzeiligen oder mehrzeiligen Text dar. Schrift und Darstellung sind einstellbar.
	"Grafikanzeige"	Zeigt Grafiken aus externen Grafikprogrammen an und fügt OLE-Objekte ein. Folgende Grafikformate können verwendet werden: "*.emf", "*.wmf", "*.dib", "*.bmp", "*.jpg", "*.jpeg", "*.gif" und "*.tif".

Elemente

Symbol	Objekt	Hinweise
	"E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Sie legen Grenzwerte für die im E/A-Feld dargestellten Variablenwerte fest. Für eine unsichtbare Eingabe des Bedieners in Runtime projektieren Sie "Verdeckte Eingabe".
	"Schaltfläche"	Führt abhängig von der Projektierung eine Funktionsliste oder ein Skript aus.
	"Symbolisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird ein Text aus einer Textliste angezeigt.
	"Grafisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird eine Grafik aus einer Grafikliste angezeigt.
	"Datum/Uhrzeit-Feld"	Gibt Datum und Uhrzeit der Systemzeit oder aus einer Variablen aus. Ermöglicht dem Bediener die Eingabe neuer Werte. Die Anzeigeform ist einstellbar.

Symbol	Objekt	Hinweise
	"Balken"	Stellt einen Wert aus der Steuerung in Form einer Säule dar, die mit einer Skala versehen ist.
	"Ladezustand"	Zeigt den Ladezustand der Batterie für Mobile Wireless
	"Schalter"	Schaltet zwischen zwei definierten Zuständen um. Sie können einen Schalter mit Text oder mit einer Grafik beschriften.
	"Schieberegler"	Zeigt einen aktuellen Wert aus der Steuerung an oder gibt einen numerischen Wert in die Steuerung ein.
	"Symbolbibliothek"	Fügt Bildobjekte ein, die auf gleichnamigen Controls basieren.
	"Wirkbereichs-Name"	Zeigt den Namen des Wirkbereichs an
	"Wirkbereichs-Name (RFID)"	Zeigt den Namen des Wirkbereichs (RFID) an
	"Wirkbereichs-Signal"	Zeigt an, wie gut sich das Mobile Panel Wireless in einem Wirkbereich befindet.
	"WLAN-Empfang"	Zeigt an, wie gut die WLAN-Funkverbindung ist.
	"Zeigerinstrument"	Zeigt numerische Werte an. Die Darstellung des Zeigerinstruments ist einstellbar.
	"Zonen-Name"	Zeigt den Namen der Zone an.
	"Zonen-Signal"	Zeigt an, wie gut sich das Mobile Panel Wireless noch in einer Zone befindet.
	"Uhr"	Zeigt die Systemzeit analog oder digital an.

Controls

Symbol	Objekt	Beschreibung
	"Meldeanzeige"	Zeigt aktuell anstehende Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv an.
	"Kurvenanzeige"	Stellt mehrere Kurven mit Werteverläufen aus der Steuerung oder aus einem Archiv dar.
	"Benutzeranzeige"	Ermöglicht einem Administrator die Benutzerverwaltung am Bediengerät. Ermöglicht einem Bediener ohne Administratorrechte, sein Kennwort zu ändern.

Symbol	Objekt	Beschreibung
	"HTML Browser"	Zeigt HTML-Seiten an.
	"Kamera-Anzeige"[1]	Zeigt Bilder von einer angeschlossenen Netzwerkkamera an.
	"PDF-Anzeige"	Zeigt PDF-Dokumente an.
	"Beobachtungstabelle"	Ermöglicht dem Bediener, vom Bediengerät direkt lesend und schreibend auf einzelne Adressbereiche in der angeschlossenen SIMATIC S7 zuzugreifen.
	"Sm@rtClient-Anzeige"	Ermöglicht dem Bediener die Fernüberwachung und Fernwartung eines angeschlossenen Geräts.
	"Rezepturanzeige"	Zeigt Datensätze an und ermöglicht die Bearbeitung der Datensätze.
	"f(x)-Kurvenanzeige"	Stellt die Werte einer Variablen als Funktion einer anderen Variablen dar.
	"System-Diagnoseanzeige"	Ermöglicht eine Übersicht über alle diagnosefähigen Geräte. Zeigt Fehler in der Anlage an.
	"PLC-Code-Anzeige"[2]	Zeigt die PLC-Programme, die in den Programmiersprachen KOP oder FUP programmiert worden sind.
	"GRAPH-Übersicht"[3]	Zeigt den aktuellen Programmstatus für ausgeführte Schritte der PLC-Ablaufkette.
	"ProDiag-Übersicht"[3]	Zeigt den aktuellen Zustand von Operandenüberwachungen.
	"Kriterienanalyse-Anzeige"[4]	Zeigt die fehlerhafte Operanden im Programm, die durch die Kriterienanalyse ermittelt wurden.
	"Media Player"	Im Mediaplayer zeigen Sie Mediadateien an.

[1] Verfügbar für Comfort Panels und KTP Mobile Panels (ab Geräteversion V13) und RT Advanced (ab Geräteversion V14 SP1)

[2] Verfügbar für WinCC Runtime Advanced (ab Geräteversion V14) und Comfort Panels und KTP Mobile Panels (ab Geräteversion V14 mit einer Displaygröße > 4")

[3] Verfügbar für WinCC Runtime Advanced, Comfort Panels und KTP Mobile Panels (ab Geräteversion V14)

[4] Verfügbar ab Geräteversion V15.

Siehe auch

Übersicht über die Objekte (Seite 91)

Bearbeitungsmöglichkeiten von Objekten (Seite 103)

Objekt einfügen (Seite 105)

Objekt löschen (Seite 106)

Objekt positionieren (Seite 107)

Größe eines Objekts ändern (Seite 109)
Mehrere Objekte auswählen (Seite 111)
Objekte ausrichten (Seite 112)
Objekt nach vorn oder hinten schieben (Seite 115)
Objekte außerhalb des Bildbereichs einblenden (Seite 116)
Objekt drehen (Seite 116)
Objekt spiegeln (Seite 118)
Objekt gestalten (Seite 119)
Objekt gestalten (Seite 119)
Objekte verbergen oder anzeigen (Seite 142)
Eigene Controls verwalten (Seite 145)
Externe Grafiken (Seite 149)
Externe Grafiken verwalten (Seite 150)
Externe Grafik in der Grafiksammlung hinterlegen (Seite 152)
Grundlagen zu Gruppen (Seite 160)
Übersicht zum Tastaturzugriff (Seite 168)
Beispiel: Rechteck einfügen und konfigurieren (Seite 176)
Blinken (Seite 139)
Mehrere Objekte eines Typs einfügen (stempeln) (Seite 139)
Grundlagen zu Bildern (Seite 33)
Blinken (Seite 138)
Übersicht über die Objekte (Seite 100)
Objekt in Runtime drehen (Seite 130)
Position und Größe mehrerer Objekte verändern (Seite 141)

1.2.1.3 Übersicht über die Objekte

Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Die Task Card "Werkzeuge" enthält alle Objekte, die für das Bediengerät verfügbar sind. Sie blenden die Task Card mit dem Menübefehl "Ansicht" ein, indem Sie die Option "Task Card" aktivieren.

Abhängig vom aktuell geöffneten Editor enthält das Werkzeugfenster unterschiedliche Paletten. Wenn der Editor "Bilder" geöffnet ist, werden im Werkzeugfenster folgende Paletten bereitgestellt:

- "Basisobjekte"
  Zu den Basisobjekten gehören grundlegende grafische Objekte, z. B. "Linie", Kreis", "Textfeld" oder "Grafikanzeige".
- "Elemente"
  Zu den Elementen gehören grundlegende Bedienelemente, z. B. "E/A-Feld", "Schaltfläche" oder "Zeigerinstrument".
- "Controls"
  Die Controls haben einen erweiterten Funktionsumfang. Sie stellen auch Prozessabläufe dynamisch dar, z. B. Kurvenanzeige und Rezepturanzeige.
- "Grafiken"
  Grafiken sind thematisch in Form eines Verzeichnisbaums gegliedert. Die verschiedenen Ordner enthalten z. B. folgende grafische Darstellungen:
  - Maschinen- und Anlagenbereiche
  - Messgeräte
  - Bedienelemente
  - Flaggen
  - Gebäude

  Sie erstellen Verknüpfungen zu Ihren eigenen Grafikordnern. Die externen Grafiken liegen in diesen Ordnern und Unterordnern. Sie werden im Werkzeugfenster angezeigt und über die Verknüpfung in das Projekt eingebunden.
- Task Card "Bibliotheken"
  Zusätzlich zu den Anzeige- und Bedienobjekten stehen Ihnen die Bibliotheksobjekte zur Verfügung. Sie befinden sich innerhalb der Paletten der Task Card "Bibliotheken". Eine Bibliothek enthält fertig konfigurierte Objekte, z. B. Grafiken von Rohren, Pumpen, vorkonfigurierten Schaltflächen. Sie binden Bibliotheksobjekte auch mehrfach in Ihr Projekt ein, ohne sie jeweils neu zu konfigurieren.
  Mit WinCC werden Bibliotheken mitgeliefert z. B. "HMI Buttons & Switches".
  Sie hinterlegen auch benutzerdefinierte Objekte und Bildbausteine in eigenen Bibliotheken. Bildbausteine sind Objekte, die Sie aus vorhandenen Bildobjekten zusammenstellen und für die Sie projektierbare Eigenschaften festlegen.

Hinweis

Bediengeräteabhängigkeit

Abhängig vom Bediengerät, das Sie projektieren, sind einige Objekte aus dem Werkzeugfenster nicht oder nur mit eingeschränkter Funktionsweise verfügbar. Nicht verfügbare Eigenschaften eines Objekts werden deaktiviert dargestellt und sind nicht auswählbar.

Basisobjekte

Symbol	Objekt	Hinweise
	"Linie"	-
	"Ellipse"	-
	"Kreis"	-
	"Rechteck"	-
	"Textfeld"	Stellt einzeiligen oder mehrzeiligen Text dar. Schrift und Darstellung sind einstellbar.
	"Grafikanzeige"	Zeigt Grafiken aus externen Grafikprogrammen an und fügt OLE-Objekte ein. Folgende Grafikformate können verwendet werden: "*.emf", "*.wmf", "*.dib", "*.bmp", "*.jpg", "*.jpeg", "*.gif" und "*.tif".

Elemente

Symbol	Objekt	Hinweise
	"E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Sie legen Grenzwerte für die im E/A-Feld dargestellten Variablenwerte fest. Für eine unsichtbare Eingabe des Bedieners in Runtime projektieren Sie "Verdeckte Eingabe".
	"Schaltfläche"	Führt abhängig von der Projektierung eine Funktionsliste oder ein Skript aus.
	"Symbolisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird ein Text aus einer Textliste angezeigt.
	"Grafisches E/A-Feld"	Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird eine Grafik aus einer Grafikliste angezeigt.
	"Datum/Uhrzeit-Feld"	Gibt Datum und Uhrzeit der Systemzeit oder aus einer Variablen aus. Ermöglicht dem Bediener die Eingabe neuer Werte. Die Anzeigeform ist einstellbar.
	"Balken"	Stellt einen Wert aus der Steuerung in Form einer Säule dar, die mit einer Skala versehen ist.
	"Schalter"	Schaltet zwischen zwei definierten Zuständen um. Sie können einen Schalter mit Text oder mit einer Grafik beschriften.

Controls

Symbol	Objekt	Beschreibung
	"Meldeanzeige"	Zeigt aktuell anstehende Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv an.
	"Kurvenanzeige"	Stellt mehrere Kurven mit Werteverläufen aus der Steuerung oder aus einem Archiv dar.

Symbol	Objekt	Beschreibung
	"Benutzeranzeige"	Ermöglicht einem Administrator die Benutzerverwaltung am Bediengerät. Ermöglicht einem Bediener ohne Administratorrechte, sein Kennwort zu ändern.
	"HTML Browser"[1)]	Zeigt HTML-Seiten an.
	"Rezepturanzeige"	Zeigt Datensätze an und ermöglicht die Bearbeitung der Datensätze.
	"System-Diagnoseanzeige"	Ermöglicht eine Übersicht über alle diagnosefähigen Geräte. Zeigt Fehler in der Anlage an.

[1)]Verfügbar für Basic Panels 2nd Generation.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.4 Bearbeitungsmöglichkeiten von Objekten

Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Sie haben folgende Möglichkeiten zum Bearbeiten von Objekten:

- Objekte über das Kontextmenü kopieren, einfügen oder löschen. Wenn Sie ein Objekt in ein Bild kopieren und das Bild bereits ein gleichnamiges Objekt enthält, wird der Name des Objekts geändert.
- Objekte in der Standardgröße einfügen oder beim Einfügen die Größe der Objekte bestimmen
- Objekt vor oder hinter andere Objekte schieben
- Objekt drehen
- Objekte spiegeln
- Tab-Reihenfolge für Objekte festlegen
- Stempeln: Mehrere Objekte gleichen Typs einfügen
- Mehrere Objekte gleichzeitig markieren

- Position und Größe mehrerer Objekte verändern
- Objekten weisen Sie externe Grafiken zu, z. B. in die Grafikanzeige.
  Nur die Grafiken werden angezeigt, die vorher in der Grafiksammlung des WinCC Projekts hinterlegt wurden.
  In der Grafiksammlung hinterlegen Sie Grafiken:
  - Per Drag&Drop aus der Objektgruppe "Grafiken" in den Arbeitsbereich
  - Als Grafik-Datei folgender Formate: *.bmp, *.dib, *.ico, *.emf, *.wmf, *.gif, *.tif, *.svg, *.jpeg oder *.jpg
  - Als OLE-Objekt
    Das OLE-Objekt erstellen Sie entweder neu oder Sie hinterlegen eine vorhandene Grafik-Datei als OLE-Objekt. Um OLE-Objekte zu hinterlegen, muss auf dem Projektierungsrechner ein OLE-fähiges Grafikprogramm installiert sein.

Siehe auch

Objekt einfügen (Seite 105)

Objekt löschen (Seite 106)

Objekt positionieren (Seite 107)

Größe eines Objekts ändern (Seite 109)

Mehrere Objekte auswählen (Seite 111)

Objekte ausrichten (Seite 112)

Objekt nach vorn oder hinten schieben (Seite 115)

Objekte außerhalb des Bildbereichs einblenden (Seite 116)

Objekt drehen (Seite 116)

Objekt spiegeln (Seite 118)

Mehrere Objekte eines Typs einfügen (stempeln) (Seite 139)

Objekte verbergen oder anzeigen (Seite 142)

Eigene Controls verwalten (Seite 145)

Externe Grafiken (Seite 149)

Externe Grafiken verwalten (Seite 150)

Externe Grafik in der Grafiksammlung hinterlegen (Seite 152)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

Übersicht über die Objekte (Seite 95)

Position und Größe mehrerer Objekte verändern (Seite 141)

1.2.1.5 Objekt einfügen

Einleitung

Im Editor "Bilder" oder "Protokolle" fügen Sie die Objekte der Task Card "Werkzeuge" ein. Sie ziehen die Objekte mit der Maus in den Arbeitsbereich. Sie fügen die Objekte in der Standardgröße ein oder bestimmen die Größe beim Einfügen.

Darüber hinaus können Sie Objekte über die Zwischenablage von einem Editor in einen anderen Editor kopieren oder verschieben, z. B. um ein Bildobjekt in ein Protokoll zu übernehmen. Alternativ können Sie zum Kopieren und Verschieben statt der Zwischenablage auch die Maus verwenden:

- Kopieren: <Strg + Drag&Drop>
- Verschieben: Drag&Drop

Hinweis

Basic Panels

Der Editor "Protokolle" steht für Basic Panels nicht zur Verfügung.

Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

Objekt in Standardgröße einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt oder die gewünschte Grafik im WinCC Grafikordner.
   Wenn Sie den Mauszeiger im Arbeitsbereich bewegen, wird er zu einem Fadenkreuz mit angehängtem Objektsymbol.
2. Klicken Sie auf die Stelle im Arbeitsbereich, an der Sie das Objekt oder die Grafik einfügen wollen.
   Das Objekt wird in der Standardgröße an der gewünschten Position im Arbeitsbereich eingefügt.

Alternativ doppelklicken Sie in der Task Card "Werkzeuge" auf ein Objekt.

Objekt kopieren

1. Markieren Sie das gewünschte Objekt.
2. Wählen Sie im Kontextmenü "Kopieren".
3. Klicken Sie an die gewünschte Stelle und wählen Sie im Kontextmenü "Einfügen".

WinCC fügt eine Kopie des Objekts an der gewünschten Stelle ein. Sie können nur die Eigenschaften ändern, die in dem jeweiligen Kontext angemessen sind.

Beispiel: Im Editor "Bilder" legen Sie für E/A-Felder den Modus für Ein- und Ausgabe fest. Im Editor "Protokolle" ist der Modus auf "Ausgabe" festgelegt.

Original und Kopie sind nicht miteinander verbunden und werden unabhängig voneinander konfiguriert.

Linien einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt.
2. Klicken Sie an einer Stelle im Arbeitsbereich. Eine Linie wird in der Standardgröße eingefügt.

Polygon oder Polygonzug einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt "Polygonzug" oder "Polygon".
2. Klicken Sie an einer Stelle im Arbeitsbereich. Der Anfangspunkt des Objekts ist festgelegt.
3. Klicken Sie an eine andere Stelle im Arbeitsbereich. Ein Eckpunkt ist festgelegt.
4. Für jeden weiteren Eckpunkt klicken Sie im Arbeitsbereich an die entsprechende Stelle.
5. Doppelklicken Sie an einer Stelle im Arbeitsbereich. Der letzte Eckpunkt ist festgelegt. Alle Punkte des Polygons oder Polygonzugs sind festgelegt.

Hinweis

Basic Panels

Die Objekte "Polygonzug" und "Polygon" stehen für Basic Panels nicht zur Verfügung.

Hinweis

Um nacheinander mehrere Objekte gleichen Typs einzufügen, nutzen Sie die Funktion "Stempel". Sie vermeiden dadurch, das Objekt vor jedem Einfügen erneut in der Task Card "Werkzeuge" zu markieren. Wählen Sie dazu in der Symbolleiste der Task Card "Werkzeuge" das Symbol .

Siehe auch

Objekt nach vorn oder hinten schieben (Seite 115)

Übersicht über die Objekte (Seite 95)

1.2.1.6 Objekt löschen

Einleitung

Sie löschen Objekte einzeln oder über eine Mehrfachauswahl.

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie löschen wollen.
   Um mehrere Objekte zu löschen, halten Sie die Taste <Shift> gedrückt und markieren Sie nacheinander die zu löschenden Objekte. Alternativ ziehen Sie mit der Maus einen Bereich um die gewünschten Objekte auf.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die selektierten Objekte werden gelöscht.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.7 Objekt positionieren

Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem Rechteck mit Anfassern umrahmt. Dieses Rechteck ist das Objektumfassende Rechteck. Die Position eines Objekts ist definiert durch die Koordinaten der linken oberen Ecke des Objektumfassenden Rechtecks.

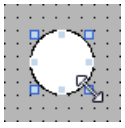

Hinweis

Wenn die Position außerhalb des Arbeitsbereichs liegt, wird das Objekt in Runtime nicht dargestellt.

Positionieren und Ausrichten

Sie haben die Möglichkeit im Arbeitsbereich ein Raster anzeigen zu lassen. Um Objekte einfacher zu positionieren, stehen Ihnen drei Optionen zur Verfügung:

- "Am Raster ausrichten": Wenn Sie Objekte bewegen, werden sie automatisch am Raster ausgerichtet und eingefügt. Wenn Sie gleichzeitig die Taste <Alt> gedrückt halten, wird das Objekt nicht am Raster ausgerichtet.
- "An Objekten ausrichten": Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.
- "Kein": Sie positionieren die Objekte an beliebiger Position.

Sie aktivieren und deaktivieren das Raster und die Optionen wie folgt:

- Im Menü "Extras > Einstellungen > Visualisierung > Bilder"
- In der Task Card "Layout > Raster"

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie verschieben wollen.
   Das markierte Objekt wird von einem Rechteck mit Anfassern umrahmt.

   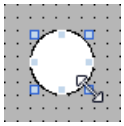

2. Klicken Sie auf das Objekt und halten Sie die linke Maustaste gedrückt.
3. Bewegen Sie den Mauszeiger auf die neue Position.
   Der Umriss des Objekts bewegt sich mit der Maus und zeigt die neue Position für das Objekt an.

   Das Objekt selbst behält zunächst seine ursprüngliche Position.
4. Lassen Sie die Maustaste los.
   Das Objekt wird auf die Position verschoben, die vorher durch den Objektumriss angezeigt wurde.

Alternatives Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die X-Werte und Y-Werte der Position ein.

Ergebnis

Das Objekt wird an der neuen Position angezeigt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.8 Größe eines Objekts ändern

Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem objektumfassenden Rechteck mit Anfassern umrahmt. Sie haben folgende Möglichkeiten, die Größe eines Objekts zu ändern:

- Die Anfasser mit der Maus ziehen
- Eigenschaft "Größe" im Inspektorfenster ändern

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, dessen Größe Sie ändern wollen.
   Das objektumfassende Rechteck wird angezeigt. Das folgende Bild zeigt ein markiertes Objekt:

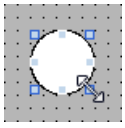

2. Ziehen Sie einen Anfasser des Rechtecks an eine neue Position.
   Die Größe des Objekts ändert sich.
   - Wenn die Funktion "Am Raster ausrichten" aktiviert ist, ändert sich die Objektgröße entsprechend den Rasterpunkten.
   - Wenn Sie beim Ziehen des Anfassers die Taste <Alt> gedrückt halten, wird diese Funktion nicht verwendet.
     Um das Objekt proportional zu skalieren, halten Sie die Taste <Shift> gedrückt, während Sie mit der Maus die Größe ändern.

Alternatives Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die Größe des Objekts ein.

Objektgröße vereinheitlichen

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der folgenden Schaltflächen: oder oder
   Die Größe der ausgewählten Objekte wird aneinander angepasst.

Das folgende Bild zeigt, wie Sie die ausgewählten Objekte an die Höhe des Referenzobjekts anpassen:

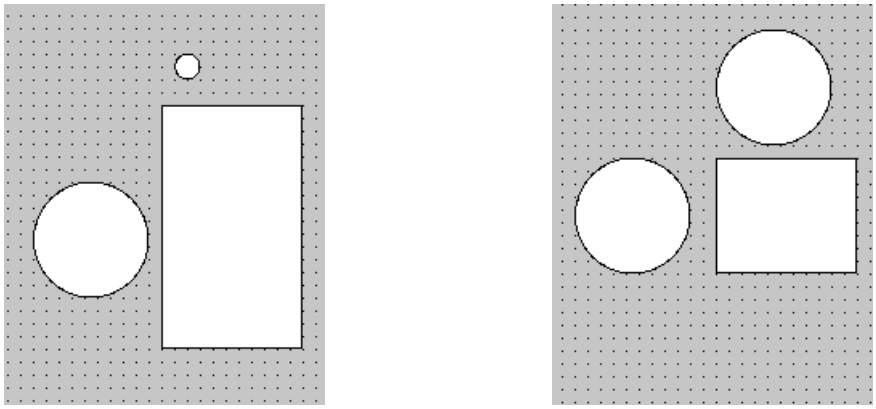

Symbol	Beschreibung
	Passt die ausgewählten Objekte an die Breite des Referenzobjekts an.
	Passt die ausgewählten Objekte an die Höhe des Referenzobjekts an.
	Passt die ausgewählten Objekte an die Breite und die Höhe des Referenzobjekts an.

Ergebnis

Das Objekt wird in der neuen Größe angezeigt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.9 Mehrere Objekte auswählen

Einleitung

Um die Eigenschaften von mehreren Objekten zugleich zu ändern oder die Objekte aneinander auszurichten, wählen Sie alle betreffenden Objekte aus. Dieser Vorgang heißt "Mehrfachauswahl".

Im Inspektorfenster erscheinen alle Eigenschaften der markierten Objekte.

Sie haben folgende Möglichkeiten, mehrere Objekte zu markieren:

- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte.
- Sie klicken mit gedrückter Taste <Shift> auf die gewünschten Objekte.

Auswahlrahmen einer Mehrfachauswahl

Der Auswahlrahmen umschließt alle Objekte einer Mehrfachauswahl. Der Auswahlrahmen ist vergleichbar mit dem objektumfassenden Rechteck eines einzelnen Objekts.

Der Auswahlrahmen ist nicht sichtbar. Nach Abschluss der Mehrfachauswahl werden folgende Rahmen angezeigt:

- Das Referenzobjekt wird mit dem objektumfassenden Rechteck angezeigt.
- Die anderen ausgewählten Objekte werden mit einem gestrichelten Rahmen angezeigt.

Referenzobjekt festlegen

Das Referenzobjekt ist das Objekt, an dem die anderen Objekte ausgerichtet werden. Das Referenzobjekt ist von einem Rechteck mit Anfassern umrahmt. Das folgende Bild zeigt ein Referenzobjekt mit zwei weiteren ausgewählten Objekten:

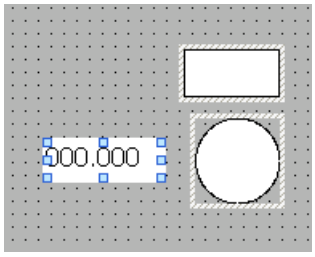

Um das Referenzobjekt festzulegen, haben Sie folgende Möglichkeiten:

- Sie wählen die Objekte über Mehrfachauswahl aus. Das zuerst ausgewählte Objekt ist dann das Referenzobjekt.
- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte. Das Referenzobjekt wird automatisch festgelegt. Wenn Sie ein anderes Objekt innerhalb der Auswahl als Referenzobjekt festlegen wollen, klicken Sie auf das gewünschte Objekt. Die Mehrfachauswahl wird dadurch nicht aufgehoben.

Voraussetzung

Der Arbeitsbereich mit mindestens zwei Objekten ist geöffnet.

Mehrere Objekte mit einem Auswahlrahmen auswählen

1. Positionieren Sie den Mauszeiger im Arbeitsbereich in der Nähe eines der auszuwählenden Objekte.
2. Ziehen Sie bei gedrückter Maustaste einen Auswahlrahmen um die auszuwählenden Objekte.

Oder:

1. Halten Sie die Taste <Shift> gedrückt.
2. Klicken Sie mit der Maus nacheinander auf die gewünschten Objekte.
   Alle ausgewählten Objekte werden durch Rahmen markiert.
   Das zuerst markierte Objekt wird als Referenzobjekt gekennzeichnet.

Hinweis

Sie entfernen ein Objekt aus der Mehrfachauswahl, indem Sie die Taste <Shift> gedrückt halten und das Objekt erneut anklicken.

Ergebnis

Mehrere Objekte sind markiert. Eines davon ist als Referenzobjekt gekennzeichnet. Sie können jetzt folgende Schritte ausführen:

- Die Objekteigenschaften aller Objekte verändern
- Die Größe aller Objekte um einen gleichen Faktor ändern, indem Sie mit der Maus den Auswahlrahmen entsprechend größer oder kleiner ziehen
- Alle Objekte gemeinsam verschieben
- Die Objekte am Referenzobjekt ausrichten

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.10 Objekte ausrichten

Vorgehen

1. Wählen Sie die gewünschten Objekte über Mehrfachauswahl aus.
2. Legen Sie ein Objekt als Referenzobjekt fest.
3. Wählen Sie in der Symbolleiste oder im Kontextmenü den gewünschten Befehl - siehe Tabelle unten.
   Die ausgewählten Objekte werden ausgerichtet.

Objekte bündig ausrichten

Die ausgewählten Objekte werden bündig am Referenzobjekt ausgerichtet.

Symbol	Beschreibung
	Richtet die ausgewählten Objekte an der linken Kante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der vertikalen Mittelachse des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der rechten Kante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der Oberkante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der horizontalen Mittelachse des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der Unterkante des Referenzobjekts aus.
	Zentriert die ausgewählten Objekte auf dem Mittelpunkt des Referenzobjekts.
	Zentriert die ausgewählten Objekte vertikal im Bild.

An einem Objekt ausrichten

Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.

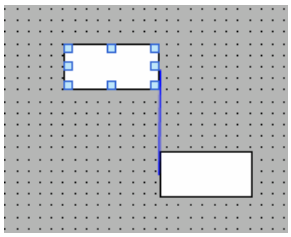

Wenn Sie mit der Tastatur arbeiten, drücken Sie die Alt-Taste. Wenn Sie das gewählte Objekt mit den Pfeiltasten bewegen, wird Ihnen der nächste Ankerpunkt angezeigt.

Objekte gleichmäßig verteilen

Sie benötigen mindestens drei ausgewählte Objekte. Ein Referenzobjekt wird nicht benötigt.

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der Schaltflächen "Horizontal gleichmäßig verteilen" oder "Vertikal gleichmäßig verteilen".
   Die ausgewählten Objekte werden in gleichmäßigen Abständen verteilt.

Das folgende Bild zeigt, wie Sie den vertikalen Abstand zwischen den ausgewählten Objekten vereinheitlichen:

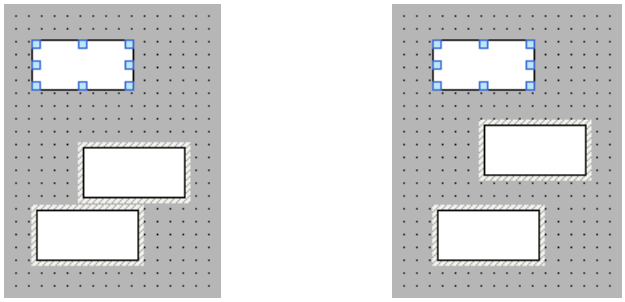

Symbol	Beschreibung
	Vereinheitlicht den horizontalen Abstand zwischen den Objekten. Die Lage der äußersten Objekte (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt.
	Vereinheitlicht den vertikalen Abstand zwischen den Objekten. Die Lage des obersten und untersten Objekts (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.11 Objekte durch Tastenkombinationen verschieben

Einleitung

Sie können die Objekte im Bild mithilfe von Tastenkombinationen verschieben oder nach Typ selektieren.

Objekte mit der Tastatur bedienen

Funktion	Tastenkombination
Verschiebt das ausgewählte Bildobjekt nach links zur ersten verfügbaren vertikalen Ausrichteposition.	<Alt+Shift+Links>
Verschiebt das ausgewählte Bildobjekt nach rechts zur ersten verfügbaren vertikalen Ausrichteposition.	<Alt+Shift+Rechts>
Verschiebt das ausgewählte Bildobjekt nach oben zur ersten verfügbaren horizontalen Ausrichteposition.	<Alt+Shift+Oben>
Verschiebt das ausgewählte Bildobjekt nach unten zur ersten verfügbaren horizontalen Ausrichteposition.	<Alt+Shift+Unten>
Selektiert alle Bildobjekte von diesem Typ.	<Ctrl+Shift+A>
Selektiert alle Objekte, die Verwendung von Stilen unterstützen.	<Ctrl+Shift+D>

1.2.1.12 Objekt nach vorn oder hinten schieben

Einleitung

Mit den Funktionen "Reihenfolge" im Kontextmenü zu einem ausgewählten Objekt oder in der Symbolleiste schieben Sie das Objekt innerhalb einer Ebene vor oder hinter andere Objekte.

Hinweis

ActiveX-Controls liegen immer vor anderen Objekten einer Ebene (.NET-Eigenschaft).

Voraussetzung

Ein Bild mit mehreren Objekten auf einer Ebene ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, das Sie nach vorn oder nach hinten schieben wollen.
2. Wählen Sie im Kontextmenü den Befehl "Reihenfolge" und einen der folgenden Befehle:

Symbol	Beschreibung
	Schiebt das markierte Objekt vor alle anderen Objekte derselben Ebene
	Schiebt das markierte Objekt hinter alle anderen Objekte derselben Ebene
	Schiebt das markierte Objekt um eine Position nach vorn
	Schiebt das markierte Objekt um eine Position nach hinten

Alternatives Vorgehen

1. Öffnen Sie die Palette "Ebenen" der Task Card "Layout".
2. Navigieren Sie zum gewünschten Objekt.
3. Ziehen Sie das Objekt in der Baumstruktur mit gedrückter Maustaste an die gewünschte Position innerhalb der Ebene.
4. Lassen Sie die Maustaste los.

Ergebnis

Das Objekt wird nach vorn oder nach hinten geschoben.

Siehe auch

Objekt einfügen (Seite 105)

Übersicht über die Objekte (Seite 95)

1.2.1.13 Objekte außerhalb des Bildbereichs einblenden

Einleitung

Wenn Sie Objekten Positionen zuweisen, die sich außerhalb des projektierbaren Bereichs befinden, werden diese Objekte ausgeblendet. Mit den Funktionen der Palette "Objekte außerhalb des sichtbaren Bereichs" in der Task Card "Layout" schieben Sie diese Objekte zurück in das Bild.

Voraussetzung

- Ein Bild mit Objekten außerhalb des projektierbaren Bereichs ist geöffnet.
- Die Task Card "Layout" ist geöffnet.

Vorgehen

1. Öffnen Sie die Task Card "Layout > Objekte außerhalb des Bereichs".
   Die Liste der Objekte, die sich außerhalb des projektierbaren Bereichs befinden, wird angezeigt.
2. Selektieren Sie in der Liste das Objekt, das Sie zurück in das Bild schieben wollen.
3. Wählen Sie im Kontextmenü des Objekts "Ins Bild ziehen".

Alternativ öffnen Sie die Task Card "Layout > Ebene". Objekte außerhalb des Bereichs werden mit dem Symbol dargestellt. Wenn Sie auf dieses Symbol klicken, wird das Objekt wieder in das Bild verschoben.

Ergebnis

Die Objekte werden in den projektierbaren Bereich gezogen.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.14 Objekt drehen

Einleitung

Sie können ein geeignetes Objekt in Schritten von 90° im oder gegen den Uhrzeigersinn um sich selbst drehen.

Hinweis

Nicht alle Objekte sind drehbar. Einige Objekte, die in Bildern drehbar sind, können in Protokollen nicht gedreht werden.

Mithilfe der Mehrfachauswahl drehen Sie auch mehrere Objekte zugleich. Bestimmte WinCC Objekte, z. B. Schaltflächen, lassen sich nicht drehen.

In einem gedrehten Objekt ändert sich die Ausrichtung der im Objekt enthaltenen Elemente. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Drehen eines Objekts verhalten:

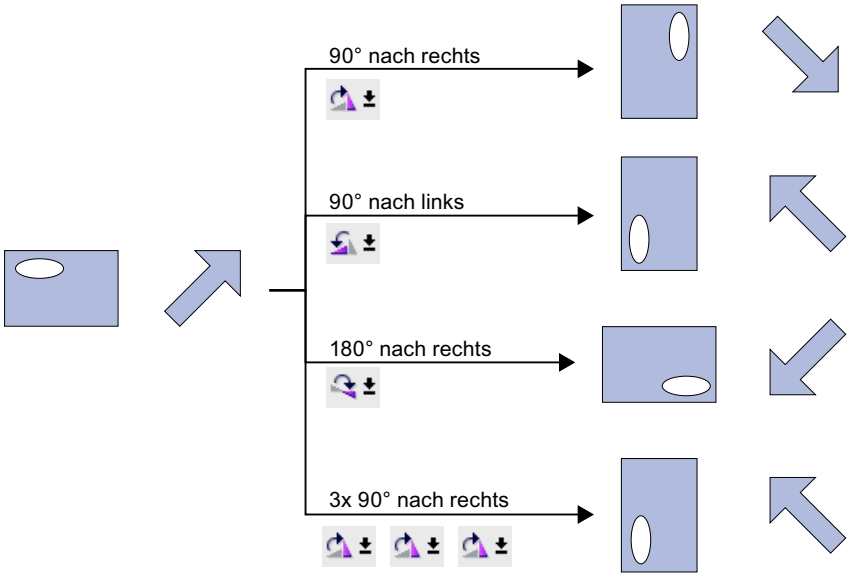

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie drehen wollen.
2. Klicken Sie in der Symbolleiste auf eines der folgenden Symbole:
   , um das Objekt im Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
   , um das Objekt gegen den Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
   , um das Objekt im Uhrzeigersinn um 180° zu drehen.

Ergebnis

Das Objekt wird gedreht angezeigt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.15 Objekt spiegeln

Einleitung

Sie können ein Objekt an seiner vertikalen oder horizontalen Mittelachse spiegeln. Wenn Sie ein Objekt spiegeln, ändert sich auch die Ausrichtung von im Objekt enthaltenen Elementen. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Spiegeln eines Objekts verhalten.

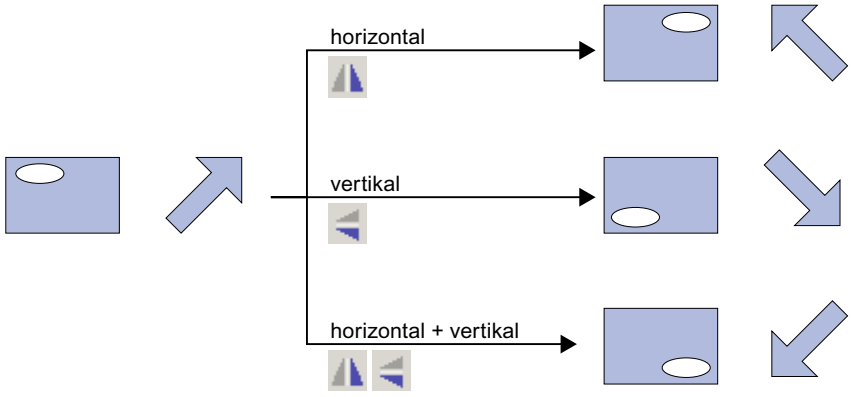

Voraussetzung

Ein Bild mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, das Sie spiegeln wollen.
2. Klicken Sie im Kontextmenü auf den Befehl "Spiegeln" und wählen Sie eine der angebotenen Möglichkeiten:
   – , um das markierte Objekt an seiner vertikalen Mittelachse zu spiegeln.
   – , um das markierte Objekt an seiner horizontalen Mittelachse zu spiegeln.

Ergebnis

Das Objekt wird gespiegelt angezeigt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.16 Objekt gestalten

Einleitung

Sie gestalten Rahmen und Hintergrund von einem Objekt.

Voraussetzung

Eine Linie ist in einem Bild angelegt.

Vorgehen

1. Markieren Sie die Linie im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
3. Wählen Sie als Stil "Strich".
4. Um die gestrichelte Linie zweifarbig darzustellen, wählen Sie für die Linienbreite "1".
5. Wählen Sie im Bereich "Linienenden" die Einstellung "Pfeil".

Ergebnis

Die Linie wird zweifarbig und gestrichelt dargestellt. Das Linienende ist ein Pfeil.

Siehe auch

Übersicht über die Objekte (Seite 91)

1.2.1.17 Objekt gestalten

Einleitung

Sie gestalten Rahmen und Hintergrund von einem Objekt.

Voraussetzung

Ein Kreis ist in einem Bild angelegt.

Vorgehen

1. Markiere Sie den Kreis im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
3. Wählen Sie als Hintergrund das Füllmuster "Massiv".
4. Aktivieren Sie "Füllzustand".

5. Geben Sie für die Füllhöhe "50" ein.
6. Wählen Sie für Stil "Massiv".

Ergebnis

Das Objekt wird als Halbkreis ohne Rahmen dargestellt.

Siehe auch

Übersicht über die Objekte (Seite 91)

1.2.1.18 Füllmuster gestalten

Einleitung

WinCC bietet die Möglichkeit, die Hintergrundfarbe und das Füllmuster eines Objekts zu gestalten. Je nachdem in welchem Objekt Sie das Füllmuster gestalten, ändern sich die Gestaltungsoptionen im Inspektorfenster.

Für bestimmte Objekte können Sie nicht nur farbigen, sondern auch einen transparenten Hintergrund oder einen Hintergrund mit Farbverlauf definieren.

Hinweis

Geräteabhängigkeit der Eigenschaften

Welche Füllart zur Verfügung steht, ist abhängig vom Objekt und vom verwendeten Bediengerät.

Voraussetzung

Objekt ist angelegt und ausgewählt.

Hintergrundfarbe eines Objekts gestalten

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Wählen Sie die Farbe für den Hintergrund des Objekts, z. B. gelb.
   Das Objekt wird mit der gewählten Farbe gefüllt.

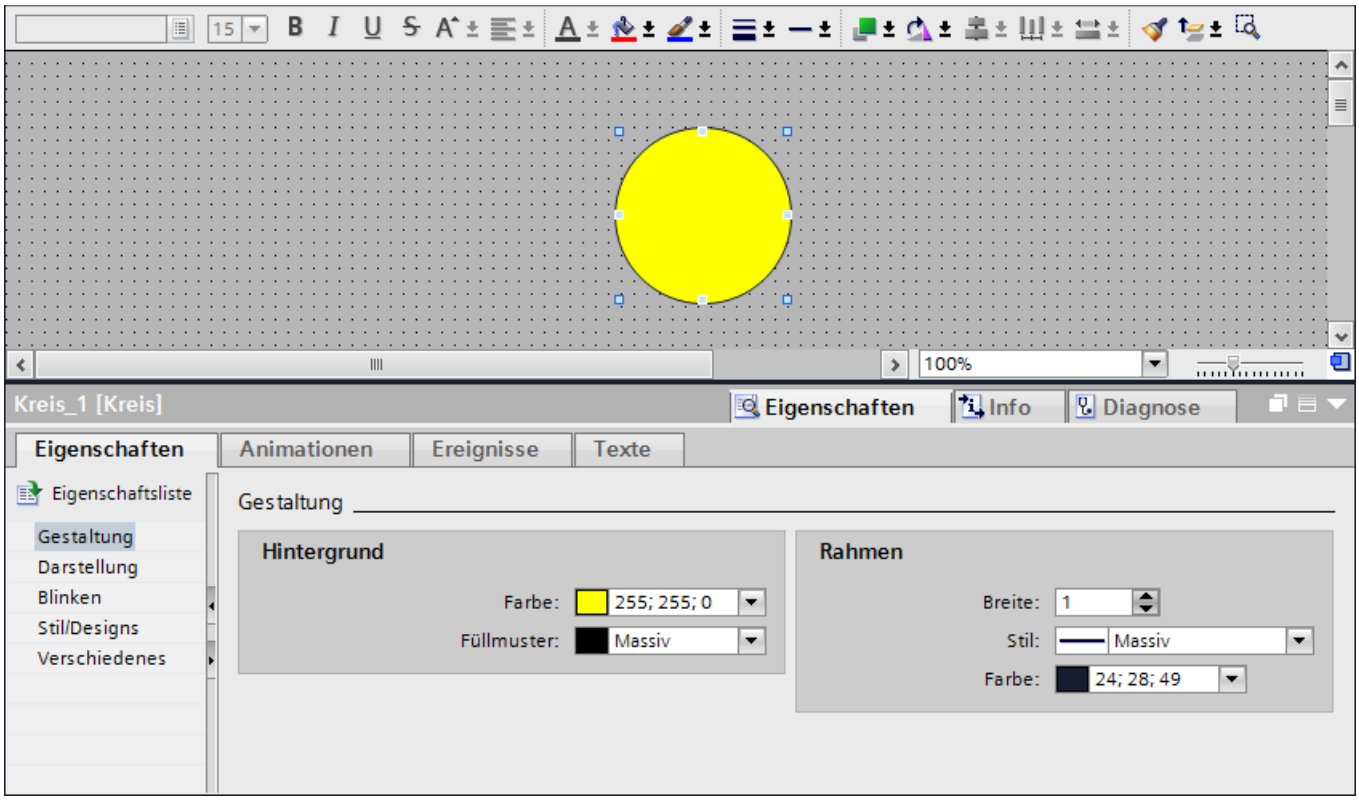

Füllmuster eines Objekts gestalten

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Um einen transparenten Hintergrund für das Objekt zu definieren, wählen Sie transparent. Informationen über die Gestaltung eines Füllmusters mit einem Farbverlauf finden Sie im Kapitel Farbverlauf definieren (Seite 134)
   Das Objekt wird transparent dargestellt.

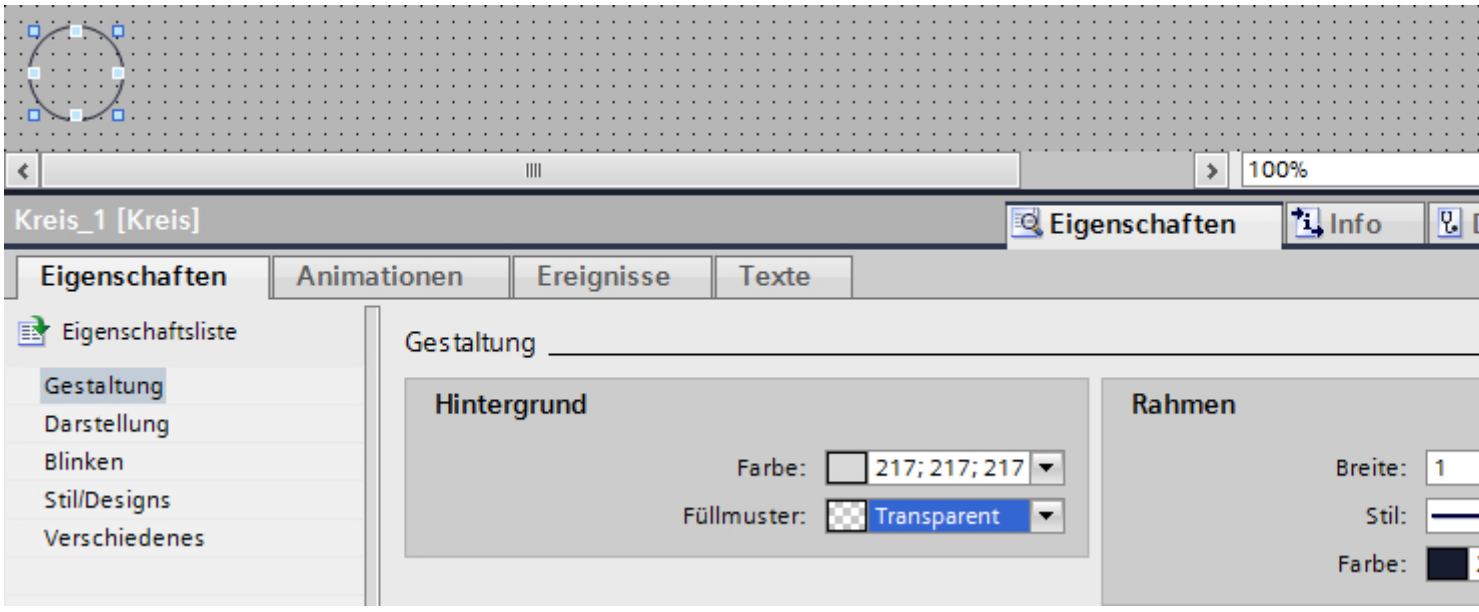

Alternativ gestalten Sie den Füllmuster Im Inspektorfenster über "Eigenschaften > Eigenschaften > Füllmuster".

Siehe auch

Farbverlauf definieren (Seite 134)

1.2.1.19 Bildlaufleisten gestalten

In den tabellenbasierten Objekten können Sie für die Navigation die Bildlaufleisten nutzen:

- Mithilfe der Bildlaufpfeile scrollen Sie zeilenweise nach oben und unten oder innerhalb der Ansicht nach links und rechts
- Mithilfe des Ziehpunkts in der Bildlaufleiste scrollen Sie im Objekt langsam zu einer gewünschten Stelle
- Durch das Klicken im Bildlauffeld navigieren Sie im Objekt schnell durch größere Datenmengen

Hinweis

Auf den Bediengeräten mit Tasten bewegen Sie die Bildlaufleiste mithilfe von Pfeiltasten.

In Pop-up-Bildern scrollen Sie mithilfe von Tastenkombination <ALT>+<Pfeiltaste>.

In WinCC ab Version V14 können Sie für die Bildlaufleisten verschiedene Farben konfigurieren.

Bildlauf in Controls

Unter "Runtime-Einstellungen > Bilder > Scrollen in Controls" stehen Ihnen zwei Darstellungsmöglichkeiten für Bildlauf zur Verfügung:

- Bildlaufleiste mit Scrollpfeilen für Geräte ab Geräteversion V14
  Im Bildobjekt wird die Bildlaufleiste mit Scrollpfeilen und Ziehpunkt angezeigt.
- Bildlaufindikator
  Im Bildobjekt wird ein Bildlaufindikator dargestellt

Im Engineering System sehen Sie die Vorschau des Bildlaufs in allen Controls mit Ausnahme von HTML-Browser, PDF-Anzeige und NC-Teilprogrammanzeige.

Hinweis

Die Anzeige von Bildlaufindikatoren wird in der PLC-Code-Anzeige und in den Pop-up-Bildern nicht unterstützt.

Daten vollständig anzeigen

Die Bildlaufleisten dienen zur Anzeige von Inhalten auf begrenztem Platz auf dem Bildschirm. Beachten Sie bei der Projektierung, dass die Bildlaufleisten an sich auch Platz im Control beanspruchen. Die Breite und Höhe der Bildlaufleisten unterscheiden sich je nach Bildschirmauflösung.

Wenn Sie ein Control mit der Darstellung von nur wenigen Zeilen projektieren, kann es in Runtime vorkommen, dass die Bildlaufleisten die Inhalte verdecken. Um das Verbergen der Daten zu vermeiden, berücksichtigen Sie bei der Projektierung die Darstellung von Bildlaufleisten.

Hinweis

Vermeiden Sie bei der Projektierung der Benutzeranzeige die Einstellung "Objekt an Inhalt anpassen". Mit dieser Einstellung wird die Objektgröße in Runtime automatisch berechnet und kann in Runtime nicht geändert werden.

Farben der Bildlaufleiste konfigurieren

Sie haben die Möglichkeit für einige Objekte die Farben der Bildlaufleiste in Verbindung mit anderen Farben zu konfigurieren.

- Scrollpfeile und Ziehpunkt werden in der Textfarbe der Tabelle dargestellt
- Hintergrundfarbe der Bildlaufleiste werden in der Alternativfarbe der Tabelle dargestellt

Hinweis

In den Objekten Graphisches E/A-Feld, HTML-Browser, PDF-Anzeige und File-Browser können die Farben der Bildlaufleiste nicht verändert werden.

Beispiel

Dieses Beispiel zeigt die Konfiguration der Farben einer Bildlaufleiste für eine Meldeanzeige.

1. Öffnen Sie "Eigenschaften > Eigenschaften > Gestaltung".
2. Wählen Sie die Farbe für den Tabellenvordergrund, z. B. "blau" für die Bildlaufpfeile und den Ziehpunkt der Bildlaufleiste
3. Wählen Sie die Alternativfarbe, z. B. "gelb" für die gerade Zeilen der Tabelle und Hintergrund der Bildlaufleiste.
   Die Bildlaufleiste wird in den konfigurierten Farben dargestellt.

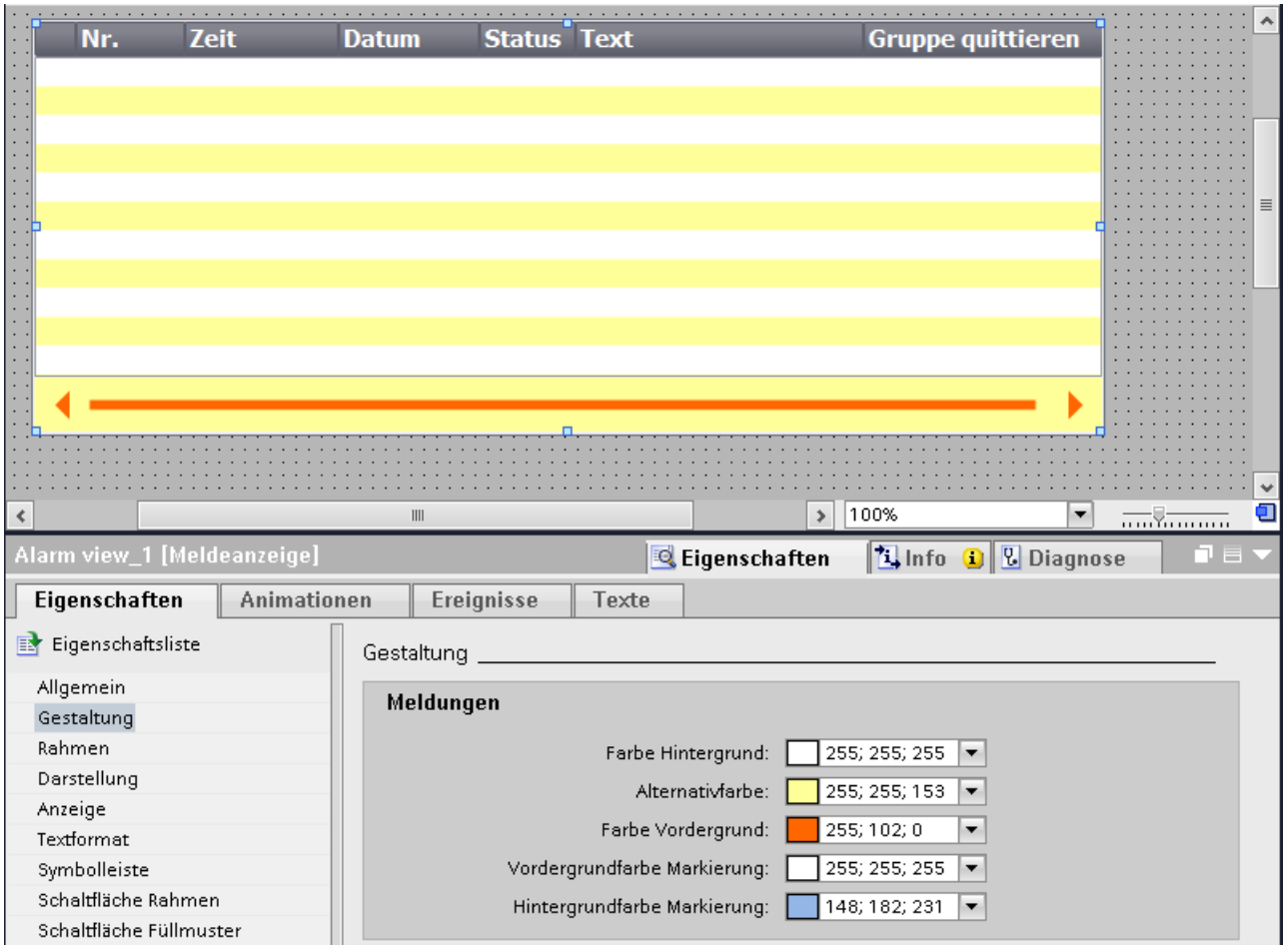

1.2.1.20 Vordefinierte Stile anwenden

Einleitung

In Runtime können Sie den Objekten und Bildbausteinen vordefinierte Stile zuweisen. Mit Hilfe von vordefinierten Stilen ändern Sie die Hintergrundfarbe der Anzeige- und Bedienobjekte. Damit harmonisieren Sie die Darstellung in Runtime.

Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

Vordefinierte Stile setzen

1. Wählen Sie in der Task Card "Werkzeuge" das Objekt, das Sie einfügen wollen.
2. In der Symbolleiste der Task Card "Werkzeuge" wählen Sie eine der folgenden Optionen:
   – "Standard", um den Standard-Stil zu verwenden
   – "Benutze Gerätedesign", um die Einstellungen für Farben aus dem aktuellen Gerätedesign zu verwenden
3. Fügen Sie das gewünschte Objekt in Arbeitsbereich ein.
   Das Objekt wird im gewählten Stil dargestellt.
4. Die Objekte werden in dem gewählten Stil erstellt, solange der Stil in der Symbolleiste aktiviert bleibt.
   Um wieder den Stadard-Stil zu setzten, wählen Sie die Option "Standard" in der Symbolleiste.

Siehe auch

Arbeiten mit Stilen und Formatvorlagen (Seite 74)

1.2.1.21 Texte in einem Objekt formatieren

Einleitung

Einige Objekte z. B. das E/A-Feld unterstützen Texte innerhalb des Objekts.

Um Text auszurichten haben Sie verschiedene Möglichkeiten.

Abstand zum Objektrand projektieren

1. Wählen Sie im Inpektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie z. B. für den linken Rand den Wert "5" ein.

Der Text wird fünf Pixel vom linken Objektrand ausgerichtet.

Textposition ausrichten

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Textformat".
2. Wählen Sie die horizontale Ausrichtung, z.B. zentriert.
3. Wählen Sie die vertikale Ausrichtung, z.B. oben.
   Der Text wird im Objekt oben mittig angezeigt.

Textorientierung definieren

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Textformat".
2. Wählen Sie die Orientierung des Textes, z.B. vertikal, rechts.
   Der Textfluss wird vertikal von unten nach oben dargestellt.

1.2.1.22 Variablen und Textlisten im Text anbinden

Einleitung

Comfort Panels und WinCC Runtime Advanced bieten die Möglichkeit bei folgenden Objekten Ausgabefelder für Variablen oder Einträge einer Textliste einzufügen:

- Symbolisches E/A-Feld
- Textfeld

Hinweis

Im symbolischen E/A-Feld steht die Anbindung von Variablen und Textlisteneinträgen im Modus "Zwei Zustände" zur Verfügung.

Voraussetzung

- Das Objekt ist angelegt und ausgewählt

Variablen in einem Textfeld anbinden

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie das Kontextmenü des Eingabefelds.

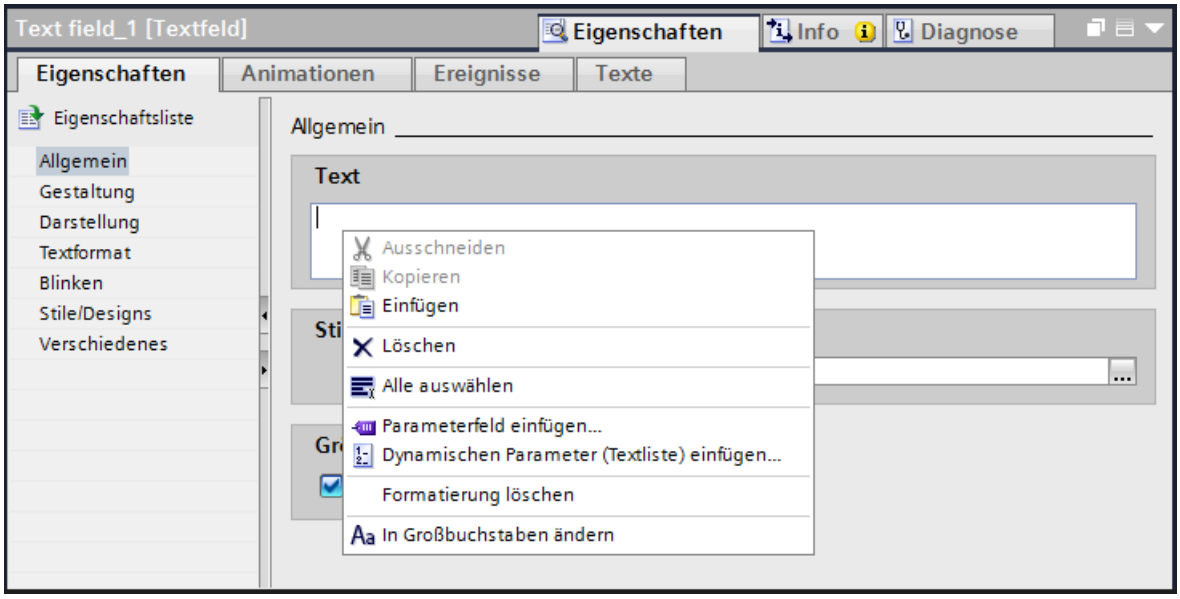

3. Wählen Sie den Befehl "Parameterfeld einfügen ...".
   Ein Dialog wird geöffnet.
4. Wählen Sie die Variable und das Format, in welchem die Variable angezeigt wird.
5. Bestätigen Sie Ihre Eingaben.

Im Textfeld des Objekts wird der Name der angebundenen Variable angezeigt.

Textliste anbinden

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie das Kontextmenü des Eingabefelds.
3. Wählen Sie z. B. den Befehl "Dynamischen Parameter (Textliste) einfügen ...".
   Ein Dialog wird geöffnet.
4. Wählen Sie die Textliste, aus der ein Texteintrag angezeigt wird.
5. Bestätigen Sie Ihre Eingaben.

Im Textfeld des Objekts wird der Verweis auf Textlisteneintrag angezeigt.

Hinweis

Die Anzahl der Verweise auf Textlisteneinträge, die wiederum Verweise auf Textlisteneinträge oder Variablen enthalten, ist auf drei Ebenen begrenzt.

1.2.1.23 Grafiken in einem Objekt formatieren

Einleitung

WinCC bietet die Möglichkeit, in manche Objekte Grafiken einzufügen und sie zu formatieren. Sie können die Größe, die Ausrichtung sowie die Abstände einer Grafik zum Objektrand verändern. Die Eigenschaften einer Grafik ändern Sie im Inspektorfenster des betreffenden Objekts.

Voraussetzung

- Das ausgewählte Objekt ist im Grafik-Modus.
- Das ausgewählte Objekt enthält mindestens eine Grafik.

Größe einer Grafik anpassen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Bild nicht strecken" oder "Bild strecken".
   Die enthaltene Grafik wird in Originalgröße dargestellt oder auf die Größe des Objekts gestreckt.

Hinweis

Um die Objektgröße an die enthaltene Grafik anzupassen aktivieren Sie "Objekt an Inhalt anpassen".

Grafik horizontal und vertikal ausrichten

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Um die horizontale Position der Grafik zu bestimmen, wählen Sie z.B. "rechts".
3. Um die vertikale Position der Grafik zu bestimmen, wählen Sie z.B. "oben".
   Die Grafik ist somit rechts oben im Objekt dargestellt.

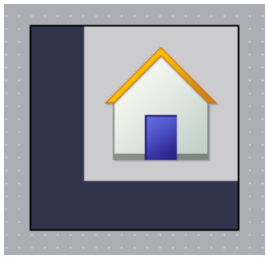

Abstand zum Objektrand definieren

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie den Wert für den Abstand zum Objektrand ein, z.B. "20" für den oberen Rand. Die Grafik wird mit dem Abstand von 20 pixel zum oberen Objektrand dargestellt.

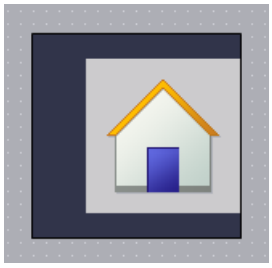

Symbol	Beschreibung
	Legt den Abstand zum linken Objektrand fest.
	Legt den Abstand zum rechten Objektrand fest.
	Legt den Abstand zum oberen Objektrand fest.
	Legt den Abstand zum unteren Objektrand fest.

1.2.1.24 Objekt in Runtime drehen

Einleitung

Sie definieren die Drehung eines Objekts um einen Referenzpunkt. Den Referenzpunkt legen Sie mit "X", "Y" im Inspektorfenster in der Gruppe "Eigenschaften > Eigenschaften > Darstellung" fest.

Die Drehung des Objekts ist nur in Runtime sichtbar.

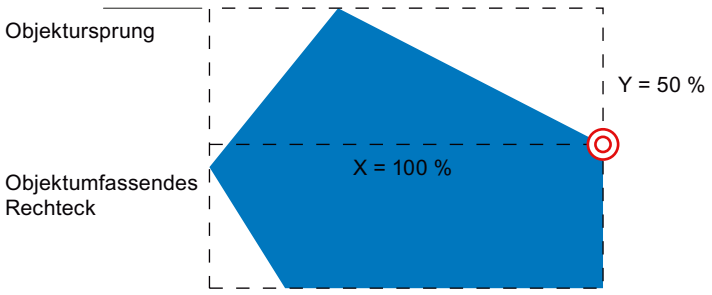

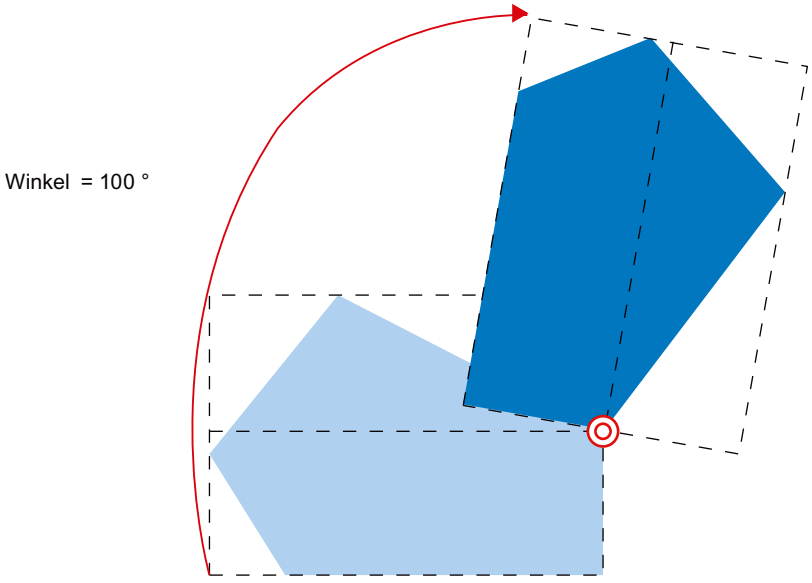

X, Y

Die Attribute "X" und "Y" definieren den horizontalen und den vertikalen Abstand des Referenzpunkts vom Objektursprung.

Die Werte werden in Prozent angegeben. Die Objektbreite bzw. Objekthöhe entspricht einem Wert von 100 Prozent. Der Referenzpunkt kann außerhalb des objektumfassenden Rechtecks liegen. Daher sind sowohl negative als auch positive Werte zulässig, deren Betrag größer als 100 Prozent ist.

Winkel

Der Winkel definiert die Drehung eines Objekts um den Referenzpunkt. Der Wert des Winkels wird in Grad angegeben. Die projektierte Anfangsstellung entspricht einem Wert von 0°. Die Lage eines Objekts weicht um den Wert des Winkels von seiner projektierten Anfangsstellung ab.

Siehe auch

Polygon (Seite 470)

Polygonzug (Seite 471)

Übersicht über die Objekte (Seite 95)

1.2.1.25 Tabellenbasierte Objekte gestalten

Einleitung

WinCC bietet verschiedene Eigenschaften, um die Anzeige- und Bedienobjekte zu gestalten.

Die Eigenschaften von tabellenbasierten Objekten ändern Sie im Inspektorfenster des betreffenden Objekts.

Farben gestalten

Unter "Eigenschaften > Eigenschaften > Gestaltung" wählen Sie die Farben für Schriftart, Markierungen und Flächen in einer Tabelle.

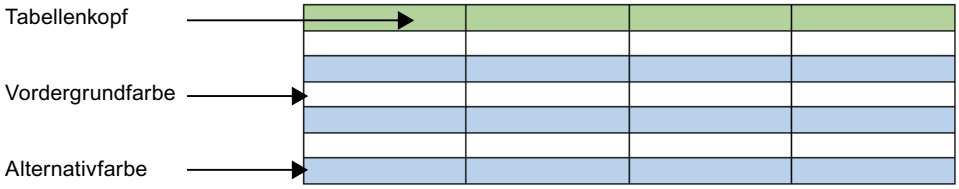

Bild 1-1 Auswahl von Tabelleneigenschaften

Rahmen des Tabellenkopfs gestalten

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tabellenkopf Rahmen".
2. Wählen Sie für "Stil" z. B. massiv.
3. Wählen Sie für "Breite" z. B. 5.
4. Wählen Sie eine Vordergrundfarbe z. B. rot.

Der Rahmen des Tabellenkopfs wird mit einem roten Rahmen und 5 Pixel breit dargestellt. Beim Rahmenstil "massiv" ist nur die Vordergrundfarbe sichtbar.

Farben und Farbverlauf im Tabellenkopf

1. Selektieren Sie das tabellenbasierte Objekt im Bild.
2. Wählen Sie im Inspektorfester "Eigenschaften > Eigenschfaten > Tabellenkopf Füllmuster".
3. Wählen Sie Füllmuster z.B. "Horizontaler Verlauf".
4. Wählen Sie unter "Verlauf" eine Hintergrundfarbe, z. B. blau.

5. Aktivieren Sie "Verlauf 1".
6. Wählen Sie eine Farbe für "Verlauf 1", z. B. weiss.
7. Wählen Sie eine "Breite" für den Farbverlauf, z. B. 12.
8. Aktivieren Sie Verlauf 2.
9. Wählen Sie eine Farbe für Verlauf 2, z. B. gelb.
10. Wählen Sie eine "Breite" für den Farbverlauf, z. B. 10

Der Tabellenkopf wird mit einem Farbverlauf dargestellt.

1.2.1.26 Rahmen gestalten

Einleitung

WinCC bietet verschiedene Eigenschaften, um die Anzeige- und Bedienobjekte zu gestalten.

Am Beispiel eines E/A-Feldes sehen Sie die möglichen Einstellungen für das Gestalten von Rahmen.

Die Gestaltung ändern Sie im Inspektorfenster des betreffenden Objekts.

Rahmen gestalten

1. Öffnen Sie "Eigenschaften > Eigenschaften > Gestaltung".
2. Wählen Sie die Breite des Rahmens, z. B. "5".
3. Wähllen Sie den Stil des Rahmens, z. B. Doppelte Linie.
4. Wählen Sie die Vordergrundfarbe.
5. Wählen Sie die Hintergrundfarbe.

Bild 1-2 Rahmenstil "Doppelte Linie"

Rahmenstil

Das nachfolgende Bild zeigt einen Rahmen im "3D Stil".

Das nachfolgende Bild zeigt einen Rahmen vom Stil "Massiv".

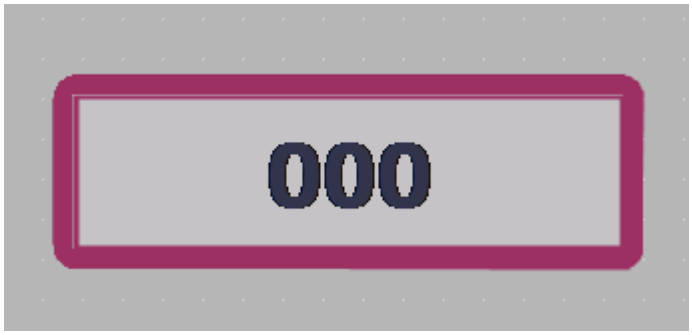

Hinweis

Geräteabhängigkeit der Eigenschaften

Welcher Rahmenstil zur Verfügung steht, ist abhängig vom Objekt und vom verwendeten Bediengerät.

1.2.1.27 Farbverlauf definieren

Einleitung

Für die Objekte in WinCC lassen sich verschieden Flächen mit Farbverläufen hinterlegen.

Je nachdem welche Fläche Sie mit einem Farbverlauf füllen, ändern Sie die Kategorie im Inspektorfenster. Das Vorgehen bleibt gleich.

Im Folgenden wird der Farbverlauf von Schaltflächen projektiert.

Horizontaler Farbverlauf mit zwei Farben

1. Selektieren Sie ein Objekt mit Schaltflächen, z. B. Schaltfläche.
2. Wählen Sie im Inspektorfester "Eigenschaften > Eigenschfaten > Füllmuster".
3. Wählen Sie "Füllmuster > Horizontaler Verlauf".
4. Wählen Sie eine Hintergrundfarbe für den vertikalen Farbverlauf z. B. orange.
5. Aktivieren Sie "Verlauf 1".
6. Wählen Sie eine "Farbe" für Verlauf 1, z. B. rot.
7. Wählen Sie eine "Breite" für Verlauf 1, z. B. "10".

Der Hintergrund der Schaltläche wird orange dargestellt.

Verlauf 1 wird bei horizontalem Farbverlauf vom linken Rand eingeblendet. Der Verlauf 1 ist 10 Pixel breit.

Vertikaler Farbverlauf mit drei Farben

1. Selektieren Sie eine Schaltfläche im Bild.
2. Wählen Sie im Inspektorfester "Eigenschaften > Eigenschaften > Füllmuster".
3. Wählen Sie Hintergrund "Vertikaler Verlauf".
4. Wählen Sie unter "Verlauf" eine Hintergrundfarbe, z. B. orange.
5. Aktivieren Sie "Verlauf 1".
6. Wählen Sie eine Farbe für "Verlauf 1", z. B. rot.
7. Wählen Sie eine "Breite" für den Farbverlauf, z. B. 8.
8. Aktivieren Sie Verlauf 2.
9. Wählen Sie eine Farbe für Verlauf 2, z. B. gelb.
10. Wählen Sie eine "Breite" für den Farbverlauf, z. B. 10.

	← Farbe Verlauf 1
	← Hintergrundfarbe
	← Farbe Verlauf 2

Verlauf 1 wird bei vertikalm Farbverlauf von oben nach unten eingeblendet. Der Verlauf 1 ist 8 Pixel breit.

Der Hintergrund der Schaltläche wird orange dargestellt.

Verlauf 2 wird am unteren Rand eingeblendet. Der Verlauf 2 ist 10 Pixel breit.

1.2.1.28 Bereiche konfigurieren

Einleitung

Ab WinCC V14 haben Sie die Möglichkeit, für die Objekte "Zeigerinstrument", "Balken" und "Schieberegler" bis zu fünf Wertebereiche zu definieren und diese zentral an einer Prozessvariable zu konfigurieren. Damit können Sie, z.B. zwischen verschiedenen Betriebszuständen farblich unterscheiden. Um die an der Variablen definierten Bereiche im Bildobjekt anzuzeigen, aktivieren Sie im Inspektorfenster unter "Eigenschaften > Grenzen/ Bereiche" die Option "Bereiche von Variablen anzeigen".

Diese Einstellung steht für die Darstellung mit inneren Skala nicht zur Verfügung, z. B. beim Balken muss die Option "Layout mit innerem Balken" deaktiviert sein.

Sie definieren folgende fünf Bereiche:

Grenze/Bereich	Verwendung
Oben 2	Zeigt den oberen Gefahrenbereich an.
Oben 1	Zeigt den oberen Warnbereich an.
Normal	Zeigt den Normalbereich an.
Unten 1	Zeigt den unteren Warnbereich an.
Unten 2	Zeigt den unteren Gefahrenbereich an.

Grenzen/Bereiche

Für numerische Variablen legen Sie Grenzwerte fest, um einen Wertebereich einzuschränken. Der Bereich wird durch einen oberen Grenzwert und einen unteren Grenzwert definiert. Verwendung und Farben von Bereichen sowie die Werte für Maximum und Minimum legen Sie am Bildobjekt fest.

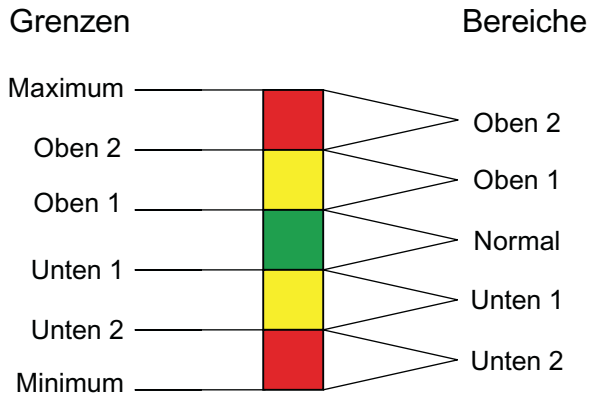

Weitere Informationen zu Grenzwerten finden Sie unter "Grenzwerte einer Variable festlegen (Seite 568)".

Voraussetzung

- Mindestens eine Variable wurde angelegt
- An der Variablen sind vier Bereiche definiert
- Das Bildobjekt wurde angelegt

Bereiche konfigurieren

1. Legen Sie unter "Eigenschaften > Allgemein > Prozessvariable" die Variable fest, die Bereiche für das Bildobjekt vorgibt.
2. Legen Sie ggf. die Werte für das Maximum und Minimum fest.
3. Aktivieren Sie unter "Eigenschaften > Grenzen/Bereiche" die Option "Bereiche von Variablen anzeigen".

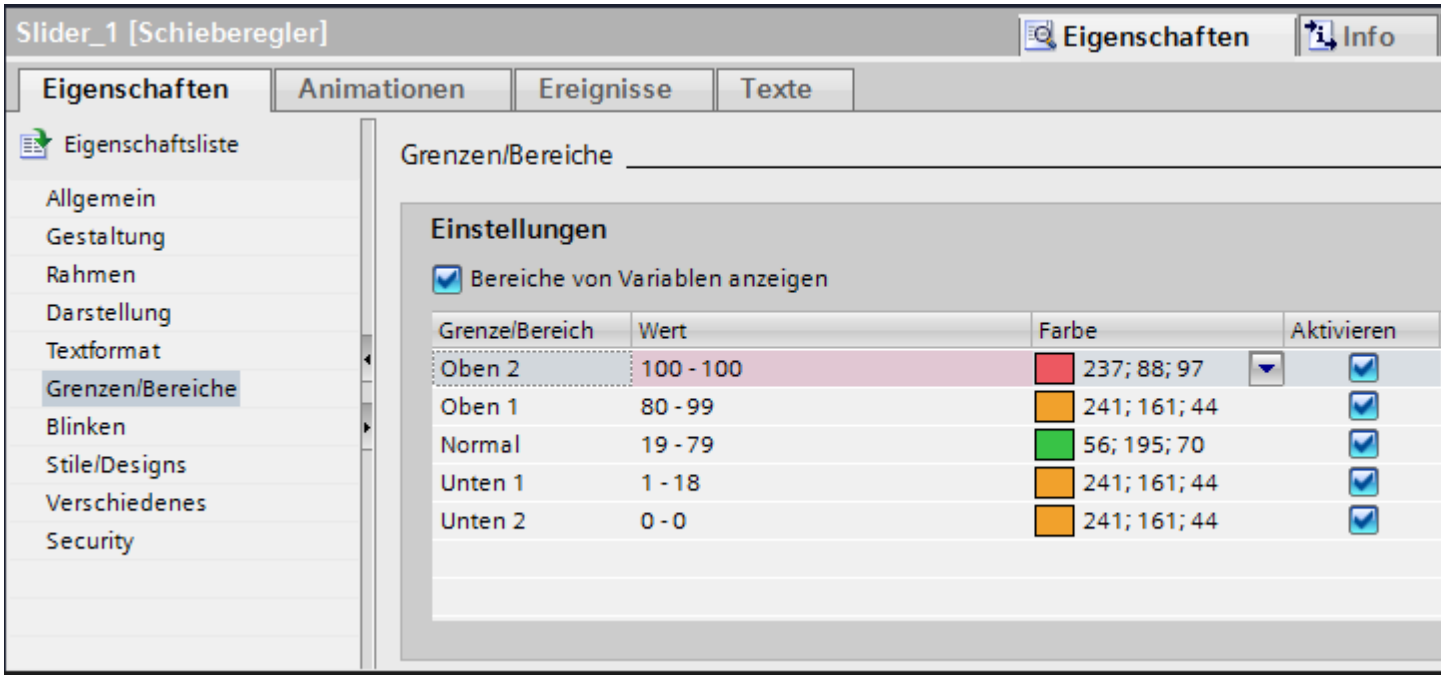

4. Legen Sie fest, welche Bereiche im Objekt dargestellt werden.
5. Ändern Sie bei Bedarf die voreingestellten Farben für die Bereiche.

Ergebnis

Sie haben in einem Bildobjekt Bereiche konfiguriert. Die Bereiche werden in Runtime in verschiedenen konfigurierten Farben dargestellt.

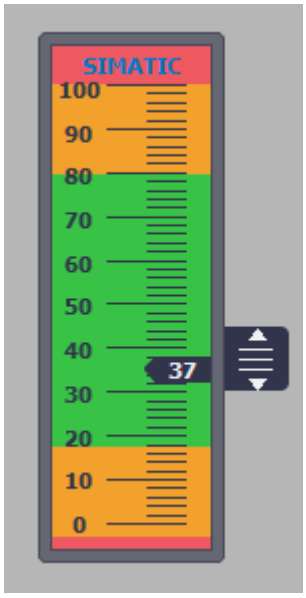

Siehe auch

Grenzwerte einer Variable festlegen (Seite 568)

1.2.1.29 Blinken

Einleitung

Sie wollen, dass ein Objekt in Runtime blinkt.

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Markieren Sie das Objekt, das blinken soll.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Blinken" den Typ "Standard aktiviert".

Ergebnis

Das Objekt blinkt in Runtime.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.30 Blinken

Einleitung

Sie stellen Objekte in Runtime blinkend dar. Sie wählen die Blinkfrequenz und die Farben für das Objekt.

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, das blinken soll.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Blinken" den Typ "Erweitert".
3. Damit in Runtime die Linie blinkt, aktivieren Sie "Linie".
4. Damit in Runtime der Text blinkt, aktivieren Sie "Text".
5. Damit in Runtime der Hintergrund blinkt, aktivieren Sie "Hintergrund".
6. Wählen Sie die Frequenz des Blinkens.
7. Wählen Sie eine Farbe für den Zustand "AUS".
8. Wahlen Sie eine Farbe für den Zustand "EIN".

Hinweis

Das Blinken ist in Runtime nur sichtbar, wenn sich die beiden Farben unterscheiden.

Ergebnis

In Runtime blinkt das Objekt in den gewählten Farben und der Blinkfrequenz.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.31 Mehrere Objekte eines Typs einfügen (stempeln)

Einleitung

WinCC bietet die Möglichkeit, mehrere Objekte des gleichen Typs direkt hintereinander zu "stempeln" d. h. einzufügen, ohne dabei jedesmal das Objekt neu auszuwählen. Außerdem haben Sie die Möglichkeit, ein bereits eingefügtes Objekt beliebig zu vervielfachen.

Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

Mehrere Objekte eines Typs einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das Objekt, das Sie einfügen wollen.
2. Klicken Sie in der Symbolleiste der Task Card "Werkzeuge" auf das Symbol .
   Die Funktion "Stempel" ist aktiviert.
3. Wenn Sie das Objekt in Standardgröße einfügen wollen, klicken Sie auf die gewünschte Position im Arbeitsbereich.
   Wenn Sie das Objekt in einer anderen Größe einfügen wollen, positionieren Sie den Mauszeiger auf die gewünschte Position im Arbeitsbereich. Ziehen Sie das Objekt bei gedrückter linker Maustaste auf die gewünschte Größe.
   Sobald Sie die Maustaste loslassen, ist das Objekt in den Arbeitsbereich eingefügt.
4. Um weitere Objekte desselben Typs einzufügen, wiederholen Sie Schritt 3.
5. Klicken Sie erneut auf das Symbol .
   Die Funktion "Stempel" ist deaktiviert.

Hinweis

Vorhandene Objekte kopieren Sie, indem Sie Drag&Drop nutzen und gleichzeitig die Taste <Strg> drücken. Das vorhandene Objekt wird dann nicht verschoben. Stattdessen wird eine Kopie davon an der neuen Position eingefügt.

Objekt einfügen und vervielfachen

1. Fügen Sie das gewünschte Objekt aus der Task Card "Werkzeuge" ein.
2. Drücken Sie die Taste <Strg> und positionieren Sie den Mauszeiger auf einen der im folgenden Bild dargestellten Anfasser.

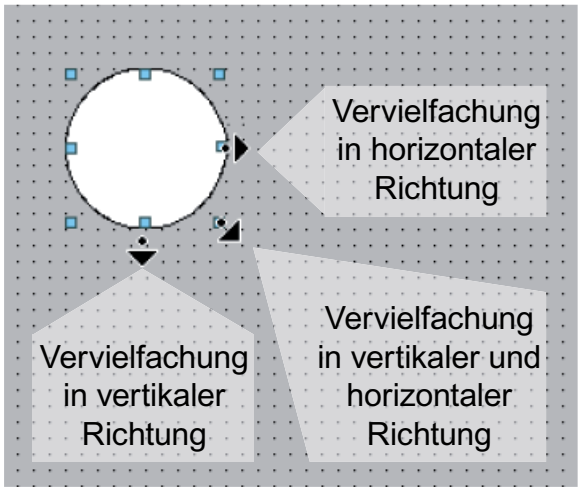

3. Ziehen Sie die Anfasser bei gedrückter linker Maustaste nach rechts und/oder unten.
4. Das Objekt wird dem verfügbaren Platz entsprechend vervielfacht, je weiter Sie den Mauszeiger bewegen.

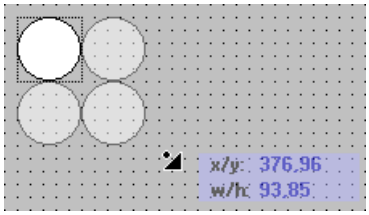

Ergebnis

Sie haben ein Objekt in einem Bild eingefügt und vervielfältigt.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.32 Position und Größe mehrerer Objekte verändern

Mögliche Änderungen

Nachdem Sie mehrere Objekte ausgewählt haben, bearbeiten Sie diese:

- Mit der Maus verschieben
  - Um die absolute Lage der markierten Objekte zu verändern, positionieren Sie den Mauszeiger über ein Objekt und verschieben bei gedrückter Maustaste die Mehrfachauswahl.
  - Um die Größe aller Objekte um den gleichen Faktor zu ändern, ziehen Sie mit der Maus an den Anfassern des Referenzobjekts.
- Mit den Symbolen der Symbolleiste über dem Arbeitsbereich verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen
- Mit den Befehlen im Kontextmenü des Arbeitsbereichs verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.33 Objekte verbergen oder anzeigen

Einleitung

Sie können Objekte ab oder bis zu einer definierten Größe verbergen und wieder anzeigen. So lassen Sie sich z.B. Objekte nur anzeigen, wenn eine ausreichende Größe das erforderliche Detail auch zeigt.

Voraussetzung

Ein Bild ist geöffnet.

Vorgehen

Um spezifische Objekte in einem Bild zu verbergen, gehen Sie wie folgt vor:

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist.
2. Aktivieren Sie im Inspektorfenster unter "Verbergen/Anzeigen" das Kontrollkästchen "Für Objekte aktivieren".
3. Geben die minimale und die maximale Pixelanzahl an, innerhalb derer die jeweiligen Objekte angezeigt werden sollen. Die angegebenen Grenzwerte werden ebenfalls noch angezeigt.

Ergebnis

Es werden die Objekte angezeigt, deren Größe sich innerhalb des angegebenen Bereichs befindet, wenn die Objektebene sichtbar ist. Das Objekt wird solange angezeigt, wie sich entweder die Höhe oder die Breite innerhalb des angegebenen Bereichs befindet.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.34 Anwenderdatentypen mit Objekten verknüpfen

Einleitung

Wenn Sie mit Anwenderdatentypen arbeiten, legen Sie nur einmal einen Anwendertyp an, den Sie mehrfach für Objekte, Animationen, Funktionslisten und Skripte verwenden können. Wenn z. B. ein Bauteil in Ihrer Anlage mehrfach verwendet wird, verknüpfen Sie das Bauteil mit einem passenden Anwendertyp, der strukturierte Unterelemente für jedes untergeordnete Bauteil enthält.

Sie weisen einem Bild einen Anwenderdatentyp zu, der in der Projektbibliothek definiert ist, und verknüpfen die Elemente dieses Anwenderdatentyps mit Objekten. Die Elemente eines Anwenderdatentyps lassen sich für Bilder und Bildobjekte anstelle von HMI-Variablen bzw. PLC-Variablen verwenden. Sie verwenden Sie zum Beispiel auch für Animationen, Funktionslisten und Skripte.

Vorgehen

1. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Anwenderdatentyp" das Optionskästchen "Anwenderdatentyp anwenden".
2. Wählen Sie in der Auswahlliste "Anwenderdatentyp" einen HMI-Anwenderdatentyp oder PLC-Anwenderdatentyp aus der Projektbibliothek, den Sie dem Bild zuweisen möchten.

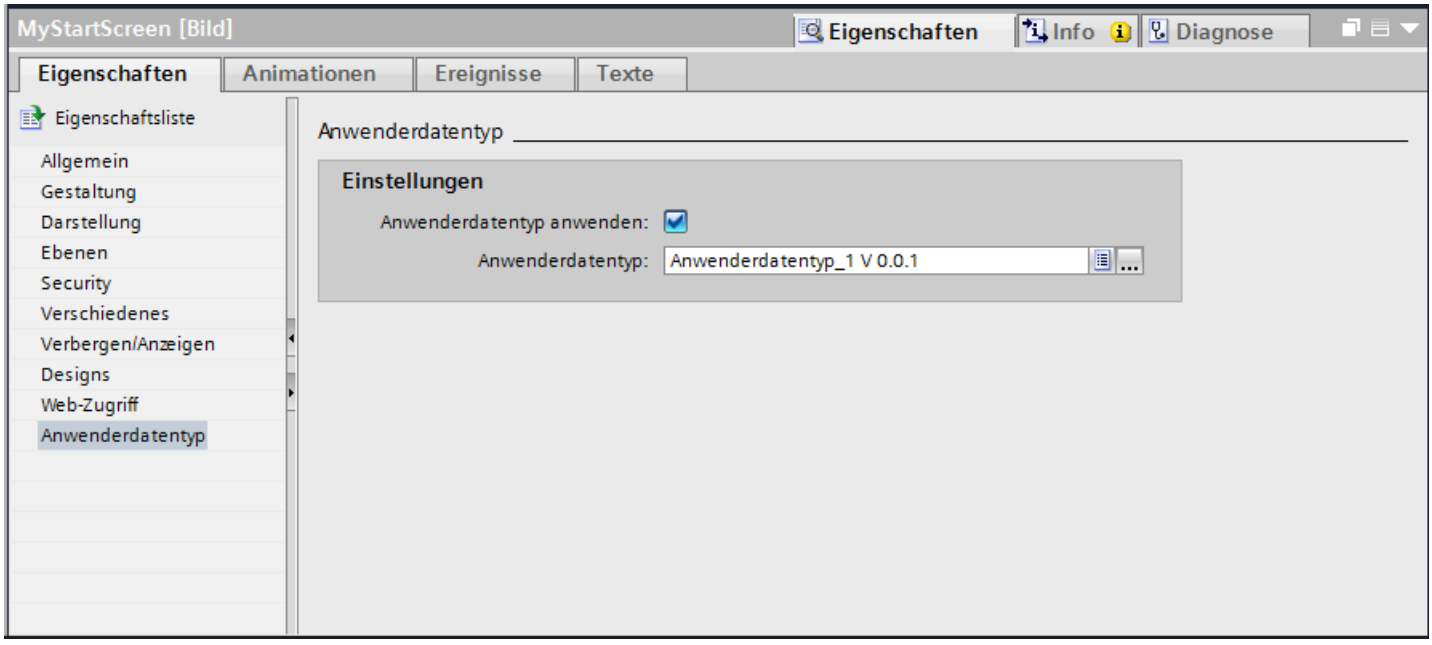

Der Anwenderdatentyp ist dem Bild zugewiesen.

Elemente eines Anwenderdatentyps mit Objekten verknüpfen

1. Fügen Sie in das Bild ein Objekt, das Sie mit einem Element eines Anwenderdatentyps verknüpfen möchten.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" in der Auswahlliste "Variable" das Element, das Sie mit dem Objekt verknüpfen möchten.

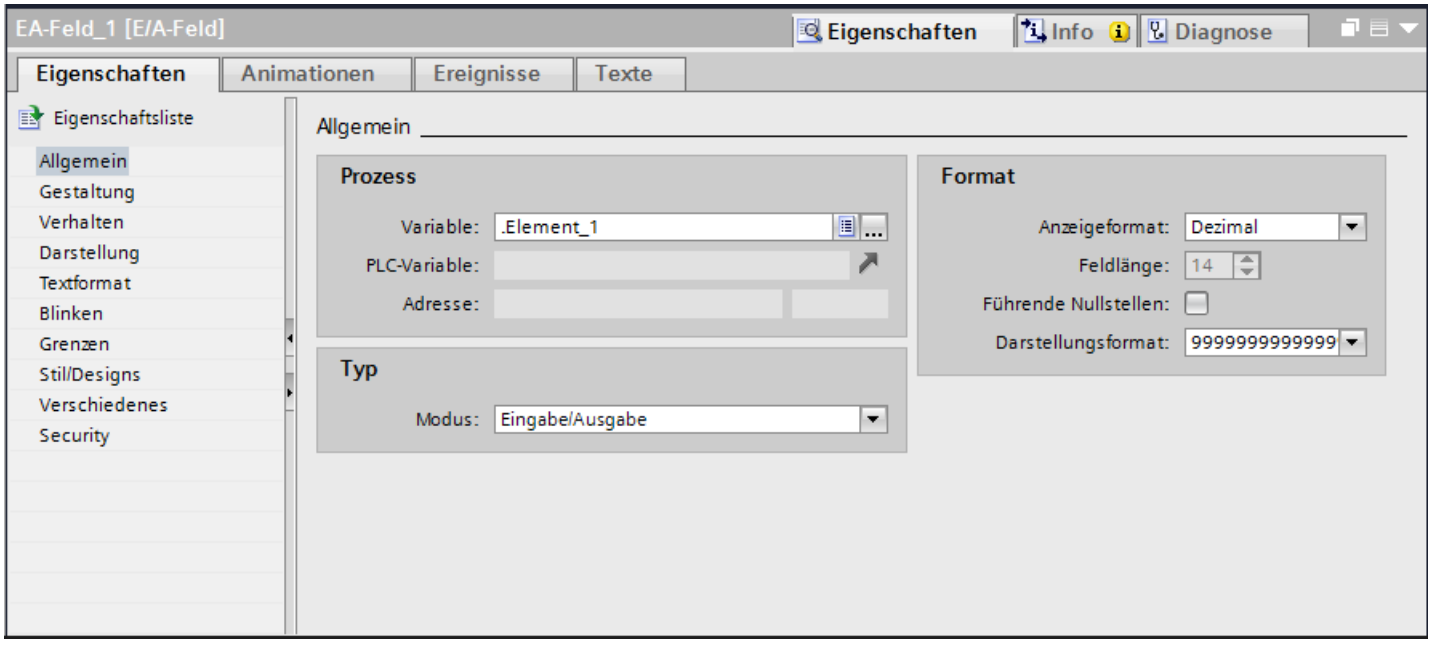

Alternativ können Sie das Element mithilfe von Autovervollständigung verknüpfen.

Hinweis

An dem der Elementbezeichnung vorangestellten Punkt erkennen Sie, dass in der Auswahlliste "Variable" ein Element eines Anwenderdatentyps verwendet wurde.

Am Präfix "@NOP" erkennen Sie, dass in der Auswahlliste "Variable" eine HMI-Variable oder eine PLC-Variable verwendet wurde.

Bemerkungen

- Wenn Sie das Optionskästchen "Anwenderdatentyp anwenden" unter "Eigenschaften > Eigenschaften > Anwenderdatentyp" deaktivieren, wird die Verknüpfung zwischen dem Element und dem Objekt aufgehoben.
- Wenn Sie dem Bild einen anderen Anwenderdatentyp zuweisen, wird die Verknüpfung zwischen dem Element und dem Objekt aufgehoben. Jedoch wenn auf derselben Strukturebene ein gleichnamiges Element im neuen Anwenderdatentyp enthalten ist, bleibt die Verknüpfung zwischen dem Element und dem Objekt erhalten.

- Wenn Sie das Element umbenennen, wird die Verknüpfung zwischen dem Element und dem Objekt aufgehoben.
- Wenn Sie den Datentyp des Elements ändern, bleibt die Verknüpfung zwischen dem Element und dem Objekt bestehen.

Siehe auch

Bildbaustein-Typ konfigurieren (Seite 280)

Arbeiten mit Anwenderdatentypen (Seite 607)

1.2.1.35 Eigene Controls verwalten

Einleitung

In den WinCC-Editionen "WinCC Advanded" und "WinCC Professional" können Sie eigene Controls verwenden. In der Task Card "Werkzeuge > Eigene Controls" fügen Sie standardisierte oder benutzerdefinierte Controls hinzu. Für die Verwendung in WinCC stehen alle im Betriebssystem registrierten ActiveX-Controls sowie die auf Ihrem System vorhandenen .Net-Controls zur Verfügung.

- ActiveX-Controls
  ActiveX-Controls sind Steuerelemente beliebiger Anbieter, die über eine definierte Schnittstelle, basierend auf OLE, von anderen Programmen benutzt werden können.
- .Net-Controls
  .Net-Controls sind Steuerelemente beliebiger Anbieter auf Basis des .Net-Frameworks von Microsoft.
- Spezifische .Net- Controls
  Benutzerdefinierte .Net-Controls sind Steuerelemente beliebiger Anbieter auf Basis des .Net-Frameworks von Microsoft.

Hinweis

Es werden nur die einfachen .Net controls unterstützt. XAML Controls, z. B. WPF (Windows Presentation Foundation) Controls, werden im TIA Portal nicht unterstützt.

Wenn der Name eines Ereignisses eines Controls die Zeichenkette "Changed" enthält, wird dieses Ereignis in WinCC nicht angezeigt.

Wenn Sie in Ihrem Projekt beliebige Controls verwenden, die nicht zum Lieferumfang gehören, müssen Sie

- überprüfen, ob diese auf dem Bediengerät ablauffähig sind
- diese zuerst in die Palette "Eigene Controls" hinzufügen, bevor Sie das Projekt auf andere PCs kopieren.

Hinweis

Wenn Sie beliebige Controls verwenden, die nicht zum Lieferumfang gehören, z. B. von Drittanbietern, kann es zu Fehlern z. B. Performanceeinbußen oder Systemblockaden kommen.

Für Probleme, die durch den Einsatz solcher Controls verursacht werden, haftet der Anwender der Software. Wir empfehlen daher eine genaue Prüfung vor dem Einsatz.

Voraussetzung

- Das gewünschte ActiveX-Control ist im Betriebssystem des PCs installiert.
- Das gewünschte .Net-Control ist auf dem PC in einer Assembly verfügbar.
- Ein Bild ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

Neues Control zur Gruppe "Eigene Controls" hinzufügen

1. Öffnen Sie in die Task Card "Werkzeuge > Eigene Controls".
2. Wählen Sie im Kontextmenü den Befehl "Objekte wählen".
   Ein Dialogfenster wird geöffnet, in dem alle auf dem System vorhandenen Controls angezeigt werden. Bereits in der Task Card "Werkzeuge" enthaltene Controls sind mit einem Häkchen gekennzeichnet.

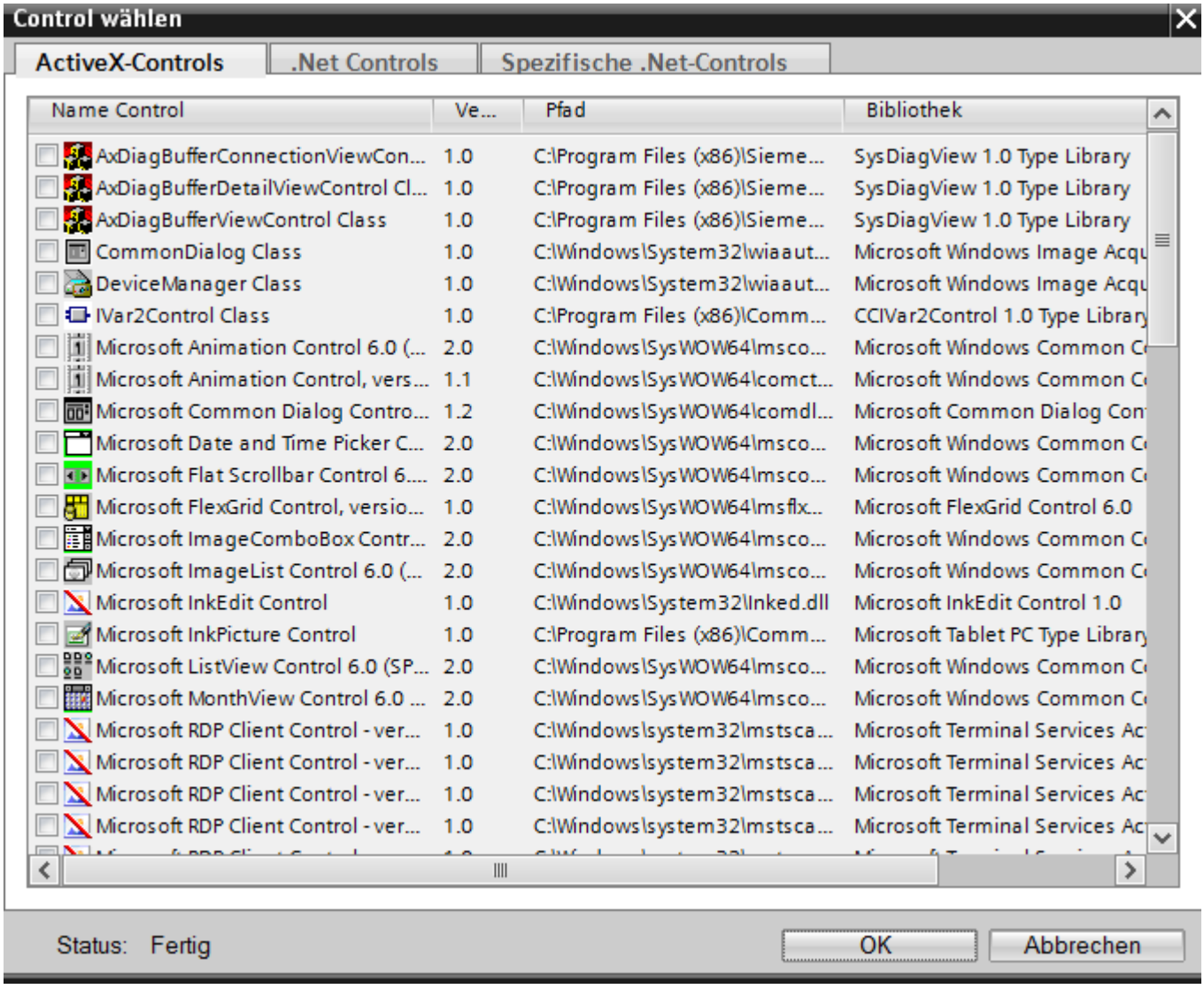

3. Wählen Sie die gewünschte Registerkarte z. B. ActiveX-Controls.
4. Aktivieren Sie das gewünschte Control.
5. Bestätigen Sie ihre Auswahl mit "OK".

Ein spezifisches .Net- Control zur Gruppe "Eigene Controls" einfugen

1. Wählen Sie die Registerkarte "Spezifische .Net-Controls". Beim ersten Aufruf ist die Liste leer.
2. Klicken Sie auf die Schaltfläche "…".
3. Wählen Sie den Ordner, in dem die Assembly enthalten ist.
4. Wählen Sie eine Datei.
5. Klicken Sie auf "Öffnen". In der Registerkarte "Spezifische .Net-Controls" werden alle verfügbaren Controls angezeigt.
6. Aktivieren Sie das gewünschte .Net-Control.
7. Bestätigen Sie ihre Auswahl mit "OK".

Hinweis

.Net-Controls in Runtime Advanced

Wenn Sie in Ihrem Projekt ein .Net-Control als "Spezifisches .Net-Control" eingebunden haben, müssen Sie die zu diesen Controls gehörenden Dateien in das Installations-Verzeichnis der WinCC Runtime, z. B. "C:\ProgramFiles\Siemens\Automation\WinCC RT Advanced", kopieren. Ansonsten kann das Control in Runtime nicht geladen werden.

Ergebnis

Das Control wird in der Task Card "Werkzeuge > Eigene Controls" angezeigt und kann in einem Bild projektiert werden.

Ein Control der Gruppe "Eigene Controls" entfernen

1. Wählen Sie im Kontextmenü des betreffenden Controls den Befehl "Löschen".
   Das Control wird aus der Gruppe "Eigene Controls" entfernt.

Synchronisirerung der ActiveX-Controls

Änderungen von Eigenschaften bei ActiveX-Controls werden nur dann gespeichert, wenn das Control die Eigenschaft OCXState bei Änderungen richtig synchronisiert.

Bei Controls wie z.B. „Microsoft Date and Time Picker Control“, die die Eigenschaft OCXState nicht richtig synchronisieren, werden die Änderungen nach dem Schließen des Bildes, beim Kompilieren oder nach der Migration verworfen.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.36 Externe Grafiken

Einleitung

In WinCC können Sie Grafiken verwenden, die mit einem externen Grafikprogramm erstellt wurden. Um diese Grafiken zu verwenden, hinterlegen Sie diese in der Grafiksammlung des WinCC Projekts.

In der Grafiksammlung hinterlegen Sie Grafiken:

- Wenn Sie grafische Objekte per Drag&Drop aus der Palette "Grafiken" in den Arbeitsbereich ziehen, werden diese automatisch in der Grafiksammlung hinterlegt. Die Grafiknamen werden in der Reihenfolge ihrer Erstellung nummeriert, z. B. "Grafik_1". Um die Grafik umzubenennen, verwenden Sie die Taste <F2>.
- Als Grafikdatei mit folgenden Formaten:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tif, *.png, *.svg, *.jpeg oder *.jpg
- Als OLE-Objekt, das mit einem externen Grafikprogramm verknüpft ist und in WinCC eingebettet wird. Bei einer OLE-Verknüpfung öffnen Sie von WinCC aus das externe Grafikprogramm. Mit dem Grafikprogramm bearbeiten Sie das verknüpfte Objekt. Nur wenn das externe Grafikprogramm OLE unterstützt und wenn das Grafikprogramm auf dem Projektierungsrechner installiert ist, funktioniert eine OLE-Verknüpfung.

Verwendung von Grafiken aus der Grafiksammlung

Grafiken aus der Grafiksammlung verwenden Sie in Ihren Bildern:

- In einer Grafikanzeige
- In einer Grafikliste
- Als Beschriftung für eine Schaltfläche/Funktionstaste

Transparente Grafiken

In WinCC verwenden Sie auch Grafiken mit transparentem Hintergrund. Wenn Sie eine Grafik mit transparentem Hintergrund in einem Grafikobjekt von WinCC einfügen, wird die Transparenz durch die im Grafikobjekt festgelegte Hintergrundfarbe ersetzt. Die gewählte Hintergrundfarbe wird fest mit der Grafik verbunden. Wenn Sie die Grafik in einem weiteren Grafikobjekt von WinCC verwenden, wird diese mit der gleichen Hintergrundfarbe dargestellt wie an dem zuerst projektierten Grafikobjekt. Wenn Sie die Grafik mit unterschiedlichen Hintergrundfarben verwenden wollen, nehmen Sie diese Grafik unter einem anderen Namen erneut in die Grafiksammlung auf. Die zusätzliche Hintergrundfarbe projektieren Sie bei der Verwendung der Grafik am entsprechenden Grafikobjekt von WinCC.

Hinweis

Transparente Grafiken in WinCC Professional

Transparente Grafiken und Grafiken mit dem Format *.svg werden im WinCC Classic Design nicht unterstützt.

Grafiken verwalten

Mit WinCC wird eine umfangreiche Sammlung von Grafiken und Symbolen installiert, z. B.:

Im Werkzeugfenster in der Palette "Grafik" sind die grafischen Objekte im Ordner "WinCC Grafikordner" thematisch gegliedert. Die Verknüpfung zum Grafikordner von WinCC lässt sich nicht entfernen, bearbeiten oder umbenennen.

Über die Palette "Grafiken" verwalten Sie auch ihre externen Grafiken. Folgende Möglichkeiten stehen zur Verfügung:

- Verknüpfung zu Grafikordnern anlegen
  Die externen Grafiken, die in diesem Ordner und in den Unterordner liegen, werden im Werkzeugfenster angezeigt und so in das Projekt eingebunden.
- Ordnerverknüpfungen bearbeiten
- Das Programm für die Bearbeitung der externen Grafik öffnen Sie von WinCC aus.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.37 Externe Grafiken verwalten

Einleitung

Externe Grafiken, die Sie in WinCC verwenden wollen, verwalten Sie im Editor "Bilder" über die Task Card "Werkzeuge" in der Palette "Grafiken".

Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.
- Die Grafiken liegen vor.
- Die Grafiken besitzen folgende Formate:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tif, *.svg, *.jpeg oder *.jpg

Ordnerverknüpfung anlegen

1. Klicken Sie auf "Eigene Grafikordner".
2. Wählen Sie im Kontextmenü "Verknüpfung…".
   Der Dialog "Verknüpfung zu Ordner erstellen" wird geöffnet. Für die neue Ordnerverknüpfung wird ein Name vorgeschlagen.

3. Ändern Sie bei Bedarf den Namen. Wählen Sie den Pfad aus, in dem die Grafiken abgelegt sind.
4. Bestätigen Sie die Eingabe mit "OK".
   Die neue Ordnerverknüpfung wird in der Objektgruppe "Grafiken" hinzugefügt. Im Werkzeugfenster werden die externen Grafiken angezeigt, die im Zielordner und in Unterordner liegen.

Ordnerverknüpfungen bearbeiten

1. Markieren Sie die Ordnerverknüpfung, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Verknüpfung bearbeiten…".
   Der Dialog "Verknüpfung zu Ordner bearbeiten" wird geöffnet.
3. Ändern Sie bei Bedarf den Namen der Ordnerverknüpfung und den Pfad.
4. Bestätigen Sie die Eingabe mit "OK".

Ordnerverknüpfungen umbenennen

1. Markieren Sie die Ordnerverknüpfung, die Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie der neuen Ordnerverknüpfung einen Namen.

Ordnerverknüpfungen entfernen

1. Markieren Sie die Ordnerverknüpfung, die Sie löschen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Entfernen".

Externe Grafiken bearbeiten

1. Markieren Sie die Grafik, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Grafik bearbeiten".
   Das mit der Grafik-Datei verknüpfte Bildbearbeitungs-Programm wird geöffnet.

Grafikordner von WinCC aus bearbeiten

1. Markieren Sie die Grafik, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Übergeordneten Ordner öffnen".
   Der Windows-Explorer wird geöffnet.

Siehe auch

Übersicht über die Objekte (Seite 95)

1.2.1.38 Externe Grafik in der Grafiksammlung hinterlegen

Einleitung

Um in einem externen Grafikprogramm erstellte Grafiken in Ihren Bildern anzuzeigen, hinterlegen Sie diese vorher in der Grafiksammlung des WinCC Projekts.

Voraussetzung

- Ein Bild ist angelegt.
- In dem Bild ist eine Grafikanzeige eingefügt.
- Das Inspektorfenster der Grafikanzeige ist geöffnet.

Um eine externe Grafik in der Grafiksammlung zu hinterlegen:

- Eine Grafik liegt vor.

Um ein OLE-Objekt in der Sammlung zu hinterlegen:

- Ein OLE-fähiges Grafikprogramm ist auf dem Projektierungsrechner installiert.

Grafikdatei hinterlegen

1. Öffnen Sie den Windows-Explorer.
2. Selektieren Sie die Grafik, die Sie hinterlegen wollen.
3. Ziehen Sie per Drag&Drop die Grafik in die Grafiksammlung.

Neue Grafik als OLE-Objekt erstellen und hinterlegen

1. Markieren Sie die Grafikanzeige im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie die Grafikauswahlliste.
4. Klicken Sie auf das Symbol .
5. Der Dialog "Objekt einfügen" wird geöffnet.

   Hinweis

   Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

6. Wählen Sie im Dialog "Objekt einfügen" die Option "Neu erstellen" und einen Objekttyp. Welche Objekttypen angezeigt werden, ist abhängig von den Einstellungen unter "Einstellungen > OLE-Einstellungen".
7. Klicken Sie auf "OK". Das dazugehörende Grafikprogramm wird geöffnet.
   Wenn Sie die Grafik erstellt haben, dann beenden Sie das Grafikprogramm über "Datei > Beenden" oder "Datei > Beenden & zurück zu WinCC".
   Die Grafik wird im Standardformat des Grafikprogramms abgespeichert und in der Grafiksammlung angezeigt.

Erstellte Grafik in WinCC einfügen

Hinweis

Wenn Sie eine neue Grafik als OLE-Objekt erstellen, wird die Grafik nach dem Speichern in dem externen Grafikprogramm eventuell nicht direkt in WinCC übernommen.

1. Öffnen Sie den Dialog für das Einfügen einer Grafik erneut.
2. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
3. Klicken Sie auf die Schaltfläche "Durchsuchen".
4. Navigieren Sie zu der erstellten Grafik und wählen Sie diese aus.

Vorhandene Grafik als OLE-Objekt hinterlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie die Grafikauswahlliste.
3. Klicken Sie auf das Symbol .
4. Der Dialog "Objekt einfügen" wird geöffnet.

   Hinweis

   Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

5. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
6. Klicken Sie auf die Schaltfläche "Durchsuchen".
7. Navigieren Sie mithilfe des Dialogs zu dem Ordner, in dem die Grafikdatei gespeichert ist.

   Hinweis

   Um Grafikdateien zu importieren, beachten Sie folgende Größenbeschränkungen:

   *.bmp, *.tif, *.emf, *.wmf ≤4 MByte

   *.jpg, *.jpeg, *.ico, *.gif "*≤1 MByte

Ergebnis

Die Grafik ist in der Grafiksammlung hinterlegt. In einem Bild wird sie mithilfe einer Grafikanzeige dargestellt oder als Listenelement in einer Grafikliste aufgenommen.

Die in der Grafiksammlung hinterlegten OLE-Objekte öffnen Sie durch Doppelklicken mit dem dazugehörigen Grafikprogramm, um sie zu bearbeiten. Wenn Sie die Grafik bearbeitet haben, dann beenden Sie das Grafikprogramm über "Datei > Beenden" oder "Datei > Beenden & zurück zu WinCC". Die Änderungen werden in WinCC übernommen.

Siehe auch

Übersicht über die Objekte (Seite 91)

1.2.1.39 Grafiken in Controls individuell anpassen

Einleitung

In der Standardkonfiguration sind die Grafiken der Schaltflächen von Controls und ihre Größe nicht veränderbar. Jedoch besteht in WinCC Professional die Möglichkeit, die Darstellung von Controls an eigene Anforderungen anzupassen. Es ist beispielsweise möglich, die Schaltflächen von Controls für die Touch-Bedienung zu vergrößern.

Sie individualisieren die Controls durch Erstellung und Bearbeitung der Grafiken, die von der Software genutzt werden. Um die vorgenommene Änderung zur Laufzeit sichtbar zu machen, wählen Sie den neuen Stil über den Konfigurationsdialog des Controls.

Für die Individualisierung der Controls haben Sie die folgenden Möglichkeiten:

- Größe und Design der Schaltflächen verändern
- Symbole in den Tabellenelementen anpassen
- Design der Bildlaufleiste verändern

Weiterführende Informationen, Beispielprojekt und Beispielgrafiken finden Sie im Internet unter Individualisierung von WinCC Runtime Professional Controls. (http://support.automation.siemens.com/WW/view/de/76327375)

Grundlagen

Für die folgenden Controls steht Ihnen die Möglichkeit der individuellen Anpassung zur Verfügung:

- Rezepturanzeige
- Meldeanzeige
- f(x)-Kurvenanzeige
- f(t)-Kurvenanzeige
- Tabellenanzeige
- Wertetabelle

Nach der Installation von TIA Portal ist für das Design der Controls der Ordner "CCAxControlSkins" unter "C:\Program Files(x86)\Common Files\Siemens\bin\" angelegt.

Um modifizierte Designs zu verwenden, müssen Sie innerhalb des Ordners "CCAxControlSkins" Unterordner anlegen. Die Zahl und die Namen der Unterordner richten sich danach, welche Elemente in welchen Controls Sie anpassen wollen.

Das Design eines Controls ist dann als "Stil"-Eigenschaft im Konfigurationsdialog des Controls auf der Seite "Allgemein" in der Auswahlliste "Stil" auswählbar.

Damit die erstellten Symbole für Tabellenelemente eines Controls sichtbar sind, muss für die entsprechenden Spalten im Bereich "Darstellung" die Option "Inhalt als Symbol" aktiviert sein.

Hinweis

Beim Erstellen eines neuen Designs müssen Sie nicht alle Dateien anlegen. Für alle Dateien, die nicht vorhanden sind, werden die Standard-Einstellungen der Controls verwendet.

Einstellungen im Engineering System

Damit Sie Controls individualisieren können, sind einige Anpassungen im Engineering System erforderlich.

1. Markieren Sie im Bild das Control, das Sie individualisieren wollen.
   Die Eigenschaften des Controls werden im Inspektorfenster angezeigt.
2. Wählen Sie unter "Eigenschaften > Eigenschaften > Spalten" in der Spalte "Titel als" für die Anzeige des jeweiligen Spaltentitels einen von den folgenden Einträgen aus:
   - "Beide" (Text und Symbol)
   - "Keine"
   - "Symbol"
   - "Text"

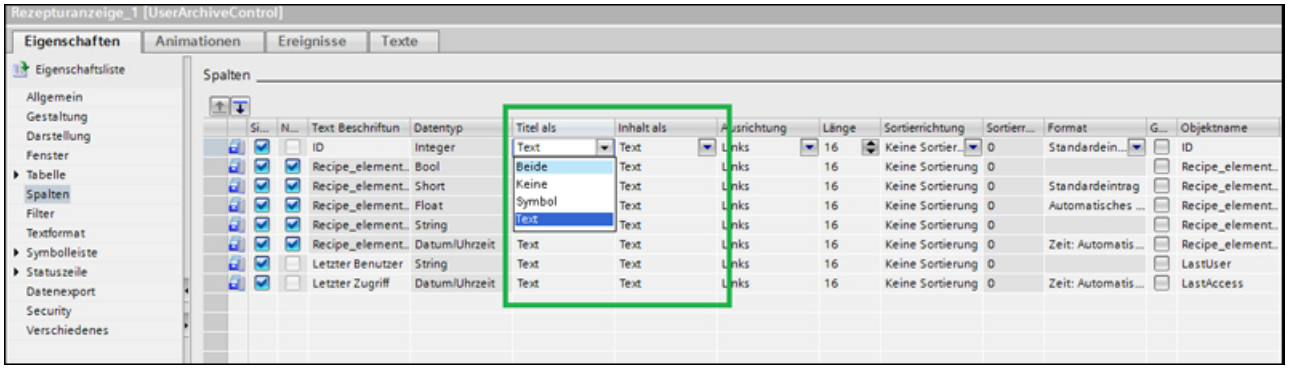

Den gewünschten Stil müssen Sie in Runtime konfigurieren.
Um die Individualisierung direkt im Engineering System vorzunehmen, verwenden Sie Dynamisierungen mit Variablen oder lokalen C- und VB-Skripten.

Ordnerstruktur für eigene Designs anlegen

Um die Verwendung von neuen Grafiken zu ermöglichen, muss lokal eine Ordnerstruktur nach bestimmten Regeln erstellt werden.

1. Legen Sie unter "C:\Program Files(x86)\Common Files\Siemens\bin\CCAxControlsSkins" einen neuen Ordner an, aus dem Grafiken im Control geladen werden, z.B. "NewControlStyle".
2. Legen Sie im neuen Ordner einen Unterordner an mit dem Namen des Controls, dessen Aussehen Sie verändern möchten, z. B. "AlarmControl" für die Meldeanzeige.

3. In dem neuen Ordner legen Sie weitere Unterordner an:
   - Ordner "Toolbar" für die Grafiken der Schaltflächen der Symbolleiste
   - Ordner "GridIcons" für die Grafiken in den Tabellenzellen

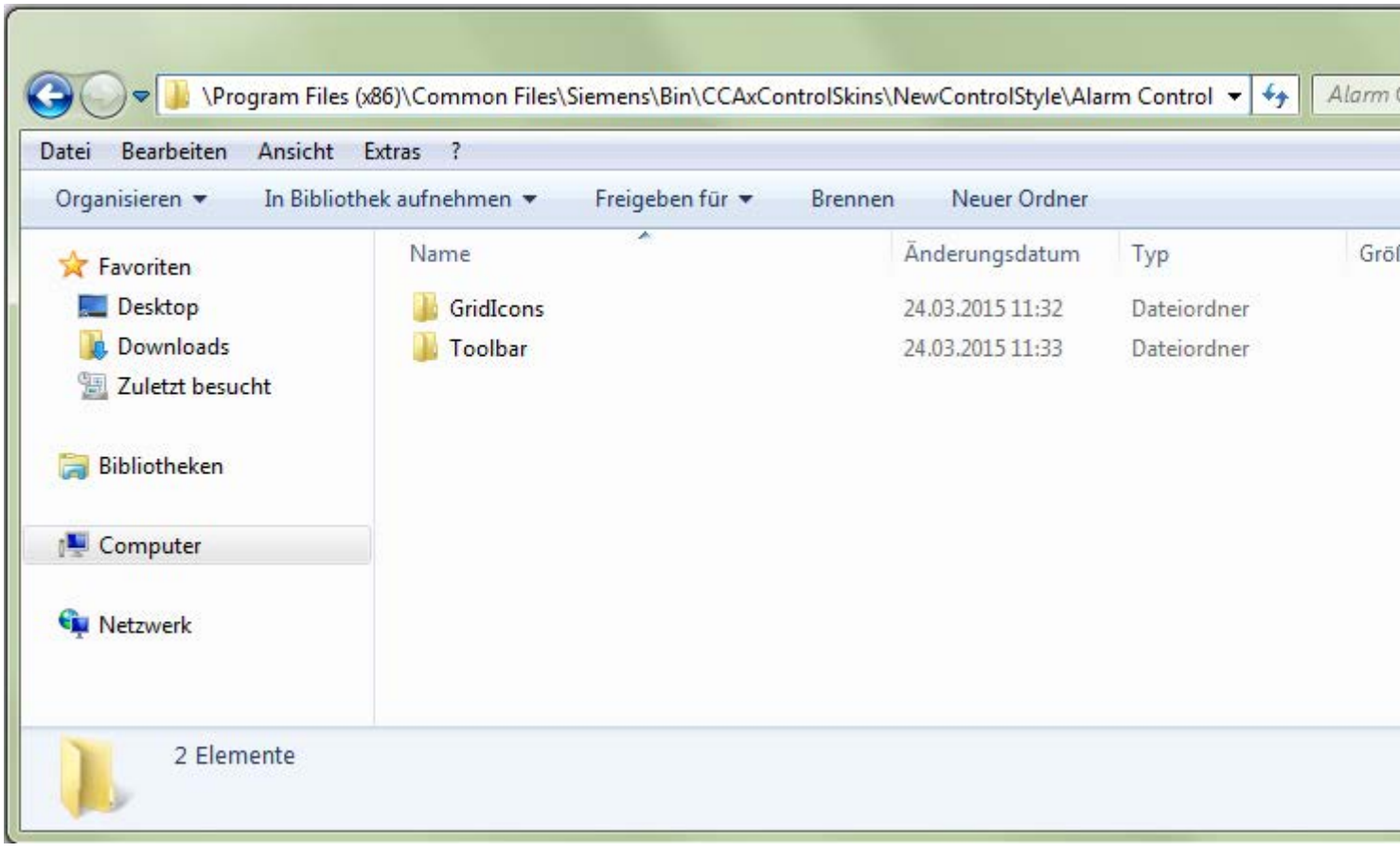

4. Wenn Sie das Aussehen der Bildlaufleiste verändern möchten, erstellen Sie unter "C:\Program Files(x86)\Common Files\Siemens\bin\CCAxControlsSkins" einen Ordner "Scrollbar" für die Grafiken der Bildlaufleiste.

Hinweis

Für die Anpassungen werden nur die Grafiken verwendet, die in den Ordnern abgelegt sind. Für die Dateien, die in den Ordnern nicht vorhanden sind, werden Standardeinstellungen der Controls verwendet.

Hinweis

Beachten Sie die korrekte Schreibweise der Datei- und Ordnernamen, damit die Software die Grafiken beziehen kann.
Zulässige Grafikformate sind PNG, GIF, JPG und ICO.

Hinweis

Beachten Sie insbesondere bei der Verwendung von Mehrplatzsystemen, bzw. Option "WebNavigator", dass der Ordner "CCAxControlSkins" lokal und somit auf dem WinCC Client vorhanden sein muss.

Schaltflächen der Symbolleiste anpassen

1. Erstellen Sie im Ordner "Toolbar" des jeweiligen Controls die Datei "IconsNormal.png".
2. Fügen Sie in einer Bildbearbeitungssoftware einzelne Grafiken der Schaltflächen in die Datei aneinandergereiht ein.
3. Um neue Grafiken zu verwenden, passen Sie die Dateien "IconsNormal.png" und "IconsDisabled.png" an.
4. Um die Größe der Symbolleisten-Grafiken zu verändern, vergrößern bzw. verkleinern Sie die Grafik "IconsBackground.png".
   Diese Datei legt die Größe für alle Symbolleisten-Schaltflächen einheitlich fest.
5. Um den Abstand um den Trennstrich anzupassen, legen Sie die Datei "Space.png" im Ordner "Toolbar" ab.
6. Erstellen Sie die Datei "Left.png", mit der Sie die Breite des Randes links vor der Schaltfläche festlegen.
7. Um die Änderung zur Laufzeit sichtbar zu machen, wählen Sie den neu angelegten Stil im Konfigurationsdialog des Controls.

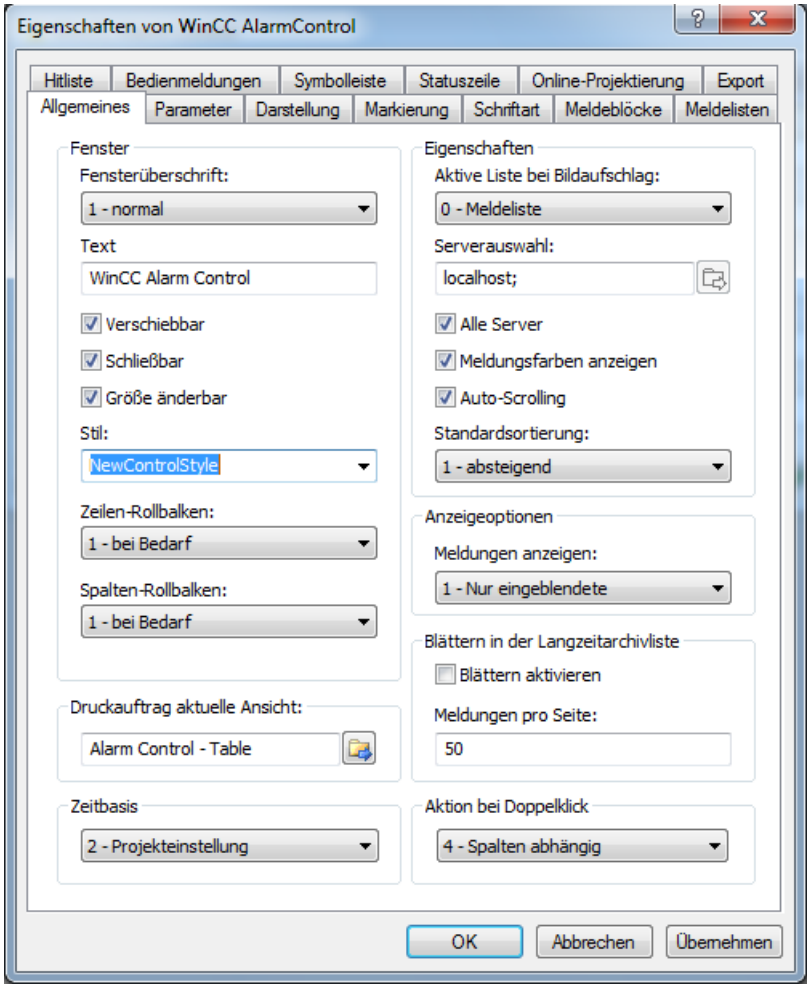

Hinweis

Der Parameter "Skin" ist nur über die Dynamisierung der Eigenschaft "Design" mit VB-Scripting einstellbar, z. B. HMIRuntime.Screens("Alarm").ScreenItems("Alarm view_1").SkinName = "MyOwnSkin".

Tabellenelemente anpassen

Um verschiedene Symbole in Abhängigkeit von Werten in den Tabellenzellen anzuzeigen, gehen Sie wie folgt vor:

1. Erstellen Sie im Ordner "..\[ControlName]\GridIcons" die Unterordner für jede Spalte der Tabelle, in der die Grafiken angezeigt werden.
   Benennen Sie die Unterordner nach dem Namen der Objekteigenschaft und verwenden Sie bei der Benennung englische Namen, z. B. "State" für die Spalte "Status".

   Hinweis

   Für die Spalten "Datum" und "Uhrzeit" können keine Symbole angezeigt werden.

2. In dem Ordner "State" müssen Sie die Grafiken mit den jeweiligen Zustandsnamen in Englisch ablegen, z. B. "ComeQuit".
   Für den Zustand, für den Sie eine Grafik abgelegt haben, wird bei Eintreten des Zustands das neue Symbol in der Tabellen-Zelle angezeigt.
3. Um Symbole für Meldenummer zu vergeben, vergeben Sie für jeden Zahlenwert eine Grafik. Wenn Sie Symbole für bestimmte Intervalle festlegen möchten, definieren Sie den Bereich im Dateinamen mithilfe eines Unterstrichs, z. B. "50_100.png" für das Intervall zwischen 50 und 100. Die Grenzen sind im Intervall enthalten.
4. Um für einen Meldeblock/Spalte nur Symbole statt des Textes anzuzeigen, müssen Sie für jeden auftretenden Text eine Grafikdatei festlegen.
5. Konfigurieren Sie die Anzeigeoptionen für den Inhalt zur Laufzeit im Konfigurationsdialog unter der Registerkarte "Meldeblöcke".

Bildlaufleiste anpassen

Anzeige der Bildlaufleiste besteht aus der Vielzahl von Grafiken und wird zur Laufzeit aus den einzelnen Grafikdateien zusammengesetzt. Um die Bildlaufleiste anzupassen, gehen Sie wie folgt vor:

1. Erstellen Sie unter "C:\Program Files(x86)\Common Files\Siemens\bin \CCAxControlsSkins" den Ordner "Scrollbar" für die Grafiken der Bildlaufleiste.
2. Legen Sie im Ordner "Scrollbar" die Unterordner "horizontal" und "vertikal" an.
3. In diesem Ordner legen Sie die einzelnen Grafiken der Bildlaufleiste ab, aus denen die Bildlaufleiste während der Laufzeit zusammengesetzt wird.
4. Wählen Sie den von Ihnen angelegten Stil zur Laufzeit im Konfigurationsdialog des Controls aus.
   Die Bildlaufleisten werden im neuen Stil angezeigt.

Ergebnis

Wenn Sie in Runtime den neuen Stil für das Control wählen, werden im Control die angepassten Grafiken angezeigt.

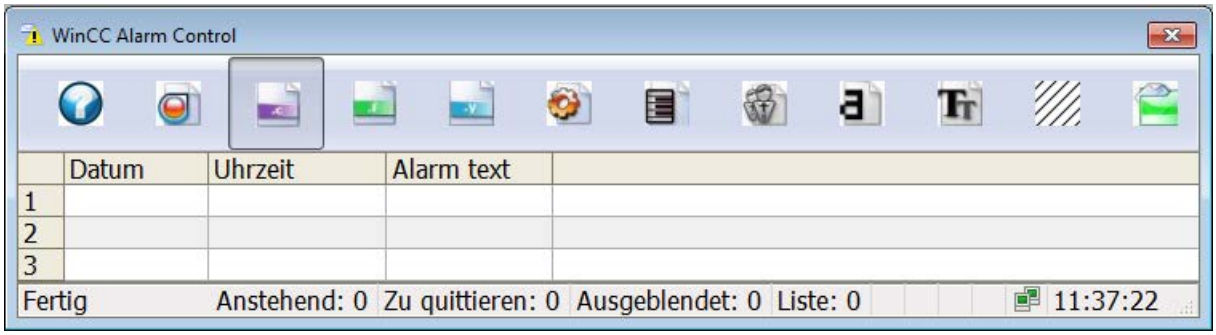

Siehe auch

Dynamisieren im Inspektorfenster (Seite 213)

1.2.2 Arbeiten mit Objektgruppen

1.2.2.1 Grundlagen zu Gruppen

Einleitung

Gruppen sind mehrere Objekte, die mit der Funktion "Gruppieren" zusammengefügt sind. Sie bearbeiten eine Gruppe wie ein einzelnes Objekt.

Übersicht

WinCC bietet Folgendes, um mehrere Objekte gemeinsam zu bearbeiten.

- Mehrfachauswahl (Seite 111)
- Objekte gruppieren (Seite 161)

Bearbeitungsmodus

Um ein Objekt einer Gruppe einzeln zu bearbeiten, selektieren Sie innerhalb der Task Card "Layout > Ebenen" das Objekt.

Alternativ wählen Sie im Kontextmenü der Objektgruppe "Gruppieren > Gruppe bearbeiten".

Hierarchische Gruppen

Um eine Gruppe zu erweitern, fügen Sie weitere Objekte oder Gruppen hinzu. Die Gruppe vergrößert sich um die neuen Objekte und baut sich hierarchisch in Haupt- und Untergruppen oder -objekte auf. Sie lösen eine solche hierarchische Gruppe nur stufenweise wieder auf. In der Reihenfolge, wie Sie die Objekte oder Gruppen gruppiert haben, lösen Sie die Gruppe auch wieder auf. Um eine solche hierarchische Gruppe aufzulösen, sind genauso viele Schritte erforderlich wie zum Gruppieren.

Objektumfassendes Rechteck

Bei einer Gruppe wird nur noch ein Objektumfassendes Rechteck für die gesamte Gruppe angezeigt. Bei der Mehrfachauswahl dagegen werden die Objektumfassenden Rechtecke aller Objekte angezeigt.

Ebenen

Alle Objekte einer Gruppe befinden sich in der gleichen Ebene.

Eigenschaften einer Gruppe

Eine Gruppe besitzt eigene Eigenschaften, z. B. Berechtigung. Wenn Sie an einer Gruppe eine Eigenschaft setzten, erben alle Objekte der Gruppe diese Einstellung. Wenn an einem Objekt der Gruppe die gleiche Eigenschaft anders konfiguriert ist, gilt in Runtime der Wert der Eigenschaft an dem Objekt und nicht der Wert in der Gruppe.

Siehe auch

Gruppierung aufheben (Seite 163)

Objekte zur Gruppe hinzufügen (Seite 164)

Objekte aus der Gruppe entfernen (Seite 165)

Objekt in einer Gruppe bearbeiten (Seite 166)

Übersicht über die Objekte (Seite 95)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.2.2 Objekte gruppieren

Einleitung

Mit dem Menübefehl "Gruppieren" fassen Sie mehrere Objekte zu einer Gruppe zusammen.

Sie können die Größe und Position der Gruppe ändern. Dafür gelten folgende Regeln:

- Wenn Sie die Position einer Gruppe ändern, wird die Position der gruppierten Objekte an die neuen Koordinaten angepasst. Relativ zur Gruppe ändert sich die Position der gruppierten Objekte nicht.
- Wenn Sie die Größe einer Gruppe ändern, werden Höhe und Breite der gruppierten Objekte entsprechend skaliert.
- Um die Größe der Gruppe proportional zu ändern, ziehen Sie mit der Maus das objektumfassende Rechteck mit gedrückter Taste <Shift> in die gewünschte Größe.

Hinweis

Um eine hierarchische Gruppe aufzubauen, gruppieren Sie einzelne Gruppen wie Objekte.

Voraussetzung

- Ein Bild mit mindestens zwei Objekten ist geöffnet.

Objekte gruppieren

1. Selektieren Sie alle Objekte, die Sie gruppieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppieren".

Die Objekte der Gruppe werden mit einem objektumfassenden Rechteck dargestellt.

Objekte innerhalb einer Gruppe gruppieren

1. Selektieren Sie die Gruppe, die Sie bearbeiten möchten.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".
   Die Gruppe, die Sie bearbeiten, wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie alle Objekte, in der Gruppe, die Sie zu einer Untergruppe zusammenfassen wollen.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppieren".

Eine Untergruppe mit den Objekten wird angelegt.

Objekte in eine bestehende Gruppe aufnehmen

1. Selektieren Sie die Gruppe, zu der Sie Objekte hinzufügen möchten.
2. Drücken Sie die <Shift> Taste und selektieren Sie das Objekt, das Sie in die Gruppe aufnehmen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Zu Gruppe hinzufügen.".

Das Objekt wird in die Gruppe eingefügt.

Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

Ergebnis

Die markierten Objekte sind in einer Gruppe enthalten. Das objektumfassende Rechteck der Mehrfachauswahl wird zum objektumfassenden Rechteck der Gruppe. Die Anfasser sind nur noch für die Gruppe eingeblendet. Die Gruppe befindet sich in der aktiven Ebene.

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Gruppierung aufheben (Seite 163)

Objekte zur Gruppe hinzufügen (Seite 164)

Objekte aus der Gruppe entfernen (Seite 165)

Objekt in einer Gruppe bearbeiten (Seite 166)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.2.3 Gruppierung aufheben

Einleitung

Mit dem Befehl "Gruppierung aufheben" lösen Sie eine Gruppe wieder in einzelne Objekte auf.

Voraussetzung

- Ein Bild mit einer Gruppe ist geöffnet.

Gruppe aufheben

1. Markieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppierung aufheben".

Gruppierung innerhalb einer Gruppe auflösen

1. Selektieren Sie die übergeordnete Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".
   Die Gruppe, die Sie bearbeiten, wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie die untergeordnete Gruppe.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppierung aufheben".

Ergebnis

Die untergeordnete Gruppe wird aufgelöst. Die Objekte werden der nächsthöheren Gruppe zugeordnet.

Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Objekte gruppieren (Seite 161)

Objekte zur Gruppe hinzufügen (Seite 164)

Objekte aus der Gruppe entfernen (Seite 165)

Objekt in einer Gruppe bearbeiten (Seite 166)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.2.4 Objekte zur Gruppe hinzufügen

Einleitung

Mit dem Befehl "Objekte zur Gruppe hinzufügen" fügen Sie Objekte zu einer Gruppe hinzu, ohne diese vorher aufzulösen.

Voraussetzungen

Ein Bild mit einer Gruppe und mindestens einem weiteren Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie die Gruppe.
2. Drücken Sie die <Shift> Taste und selektieren Sie das Objekt, das Sie in die Gruppe aufnehmen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Zu Gruppe hinzufügen".

Ergebnis

Die Gruppe besteht aus den ursprünglichen Objekten und den neu hinzugefügten Objekten. Die hinzugefügten Objekte sind in der Tiefenstaffelung der Gruppe vorne angeordnet.

Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.2.5 Objekte aus der Gruppe entfernen

Einleitung

Mit dem Befehl "Objekte aus Gruppe entfernen" entfernen Sie einzelne Objekte aus einer Gruppe, ohne diese vorher aufzulösen.

Zur Bearbeitung eines Objekts in einer Gruppe müssen Sie das Objekt nicht aus der Gruppe entfernen. Objekte einer Gruppen können Sie einzeln bearbeiten.

Voraussetzung

- Ein Bild mit einer Gruppe ist geöffnet.

Objekte aus einer Gruppe entfernen

Um ein Objekt aus einer Gruppe herauszulösen:

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".
   Die Gruppe, die Sie bearbeiten wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie in der Gruppe alle Objekte, die Sie aus der Gruppe entfernen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Aus Gruppe entfernen".

Die Objekte werden aus der Gruppe herausgelöst.

Hinweis

Wenn nur noch zwei Objekte in der Gruppe sind, dann steht der Menübefehl "Aus Gruppe entfernen" nicht zur Verfügung.

Objekte aus einer Gruppe löschen

Um ein Objekt aus der Gruppe und aus dem Bild zu entfernen:

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".
   Die Gruppe, die Sie bearbeiten wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie in der Gruppe alle Objekte, die Sie löschen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Wenn nur noch zwei Objekte in der Gruppe sind, dann steht der Menübefehl "Löschen" nicht zur Verfügung.

Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.2.6 Objekt in einer Gruppe bearbeiten

Einleitung

Objekte einer Gruppe können einzeln bearbeitet werden.

Voraussetzung

Ein Bild mit einer Gruppe ist geöffnet.

Gruppierte Objekte bearbeiten

1. Selektieren Sie die Gruppe.

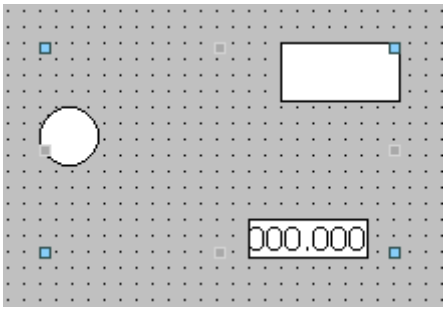

   Die Eigenschaften der Gruppe werden im Inspektorfenster angezeigt.

2. Ändern Sie die Position und Größe der gruppierten Objekte unter "Eigenschaften > Eigenschaften > Darstellung".
3. Ändern Sie den Namen der Gruppe unter "Eigenschaften > Eigenschaften > Verschiedenes".

Eigenschaften eines Objekts in einer Gruppe ändern

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Inspektorfenster das Objekt, dessen Eigenschaften Sie ändern wollen.

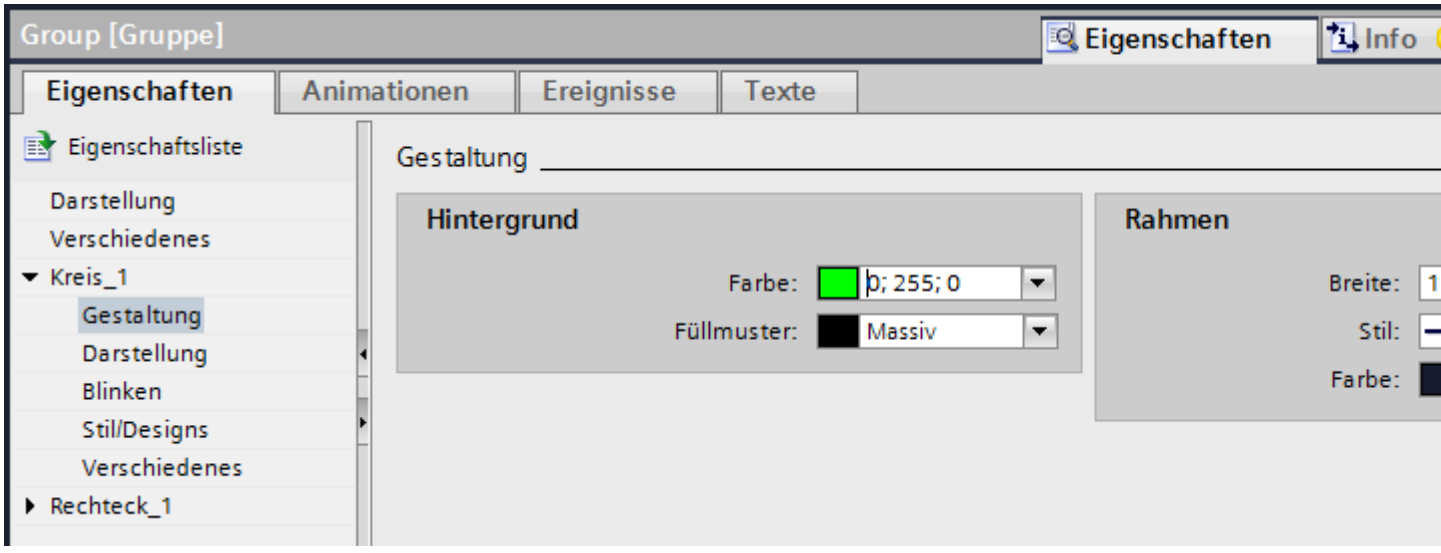

   Die Eigenschaften des Objektes werden angezeigt.

3. Ändern Sie die Objekteigenschaften.

Hinweis

In einer Gruppe ist die Dynamisierung von Eigenschaften für alle Objekte der Gruppe, die diese Eigenschaft besitzen, nicht möglich. Sie können die Eigenschaften der zu einer Gruppe gehörenen Objekte nur für jedes Objekt selbst dynamisieren.

Ergebnis

Das Objekt wurde geändert, bleibt aber weiterhin Bestandteil der Gruppe. Die anderen Objekte der Gruppe bleiben unverändert.

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3 Konfigurieren des Tastenzugriffs

1.2.3.1 Übersicht zum Tastaturzugriff

Einleitung

An Tastengeräten ohne Maus aktiviert der Bediener Bedienobjekte mit der Taste <Tab>. Um komfortabel zu bedienen und um sicherzustellen, dass der Bediener alle nötigen Werte eingibt, richten Sie für ihn die Eingabe über die Tastatur ein. Wenn Sie die Tastatur bedienen, aktivieren Sie mit der Taste <Tab> die Objekte in einer bestimmten Reihenfolge und geben die erforderlichen Werte ein.

Bei Bediengeräten ohne Tasten simulieren Sie die Taste <Tab>, indem Sie die Systemfunktion "SimuliereSystemtaste" an eine Funktionstaste projektieren.

Bedienberechtigung und Bedienfreigabe

Wenn Sie ein Objekt für die Bedienung mit der Taste <Tab> projektieren, muss das Objekt eine Bedienberechtigung und eine Bedienfreigabe haben.

Tab-Reihenfolge bearbeiten

Die Tab-Reihenfolge wird beim Anlegen der Bedienobjekte automatisch bestimmt. Die Nummern der Tab-Reihenfolge werden in der Reihenfolge vergeben, in der die Bildobjekte angelegt werden.

In folgenden Fällen ist die Änderung der Tab-Reihenfolge sinnvoll:

- Der Bediener wechselt direkt auf ein bestimmtes Bedienobjekt
- Das Bild erfordert eine bestimmte Reihenfolge

Um die Tab-Reihenfolge zu ändern, wechseln Sie in den Tab-Reihenfolgemodus. In diesem Modus wird an den Bedienobjekten links oben die Tab-Reihenfolge-Nummer eingeblendet. Sie sehen auch die Tab-Reihenfolge-Nummern von verdeckten Objekten. Die Verteilung dieser Nummern bearbeiten Sie mit der Maus.

Hinweis

Im Tab-Reihenfolge-Modus sind keine anderen Funktionen verfügbar.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 169)

Cursormodus festlegen (Seite 170)

Bedienberechtigung und Bedienfreigabe für ein Objekt festlegen (Seite 171)

Objekttypen für den Schaltcursormodus festlegen (Seite 172)

Tab-Reihenfolge festlegen (Seite 173)

Tab-Reihenfolge festlegen (Seite 175)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

Übersicht über die Objekte (Seite 95)

1.2.3.2 Übersicht zum Tastaturzugriff

Einleitung

Um den Prozess zu beeinflussen, bearbeitet der Bediener in Runtime bedienbare Objekte, z.B. E/A-Felder oder Schaltflächen. Um komfortabel zu bedienen und um sicher zu stellen, dass der Bediener alle notwendigen Werte eingibt, richten Sie für ihn die Eingabe über Tastatur ein. Wenn der Bediener Eingaben macht, dann aktiviert er mit der Taste <Tab> die Objekte in einer festgelegten Reihenfolge und gibt die erforderlichen Werte ein.

Tastaturzugriff

Es gibt zwei Arten der mauslosen Bedienung:

- Alphacursor: Der Bediener aktiviert in Runtime mit der Taste <Tab> nur Objekte vom Typ "E/A-Feld" und "Textliste".
- Schaltcursor: Je nach Einstellung aktiviert der Bediener in Runtime mit der Taste <Tab> alle Objekte, die eine Eingabe zulassen.

Sie legen den Cursormodus für den Start von Runtime fest. Den Cursormodus stellen Sie im Inspektorfenster des Bildes ein.

Alphacursor und Schaltcursor kombinieren

Wenn Sie einen Hotkey festgelegt haben, schaltet der Bediener den Cursormodus in Runtime bei Bedarf zwischen den beiden Modi um. Sie legen die Tab-Reihenfolge in diesem Fall für beide Cursorarten fest.

Bedienung in Runtime

Sie legen über die Tab-Reihenfolge fest, in welcher Reihenfolge der Bediener Objekte aktiviert. Die Standardeinstellung ist die Bedienung über die Taste <Tab>. Bei Bedarf legen Sie die Bedienung über einen Hotkey oder über die Maus fest.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3.3 Cursormodus festlegen

Einleitung

Sie legen den Cursormodus für den Start von Runtime fest. Der Bediener kann den Cursormodus in Runtime bei Bedarf umschalten. Sie stellen den Cursormodus in den Objekteigenschaften des Bildes ein.

Voraussetzung

Ein Bild mit mehreren Objekten ist geöffnet.

Vorgehen

1. Klicken Sie auf eine freie Fläche im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes".
3. Wählen Sie unter "Tab-Reihenfolge" den gewünschten Cursormodus aus.

Ergebnis

Beim Start von Runtime ist der gewünschte Cursormodus aktiv.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3.4 Bedienberechtigung und Bedienfreigabe für ein Objekt festlegen

Einleitung

Wenn Sie ein Objekt für die Bedienung mit der Taste <Tab> projektieren, muss das Objekt eine Bedienberechtigung und eine Bedienfreigabe haben.

Voraussetzung

Ein Bild mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt.
2. Wählen Sie in im Inspektorfenster "Eigenschaften > Eigenschaften > Sicherheit".
3. Wählen Sie unter "Berechtigung" die Bedienberechtigung aus.
4. Aktivieren Sie die Berechtigung zur Bedienung.

Ergebnis

In Runtime kann der Bediener mit der Taste <Tab> das Objekt anwählen.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3.5 Objekttypen für den Schaltcursormodus festlegen

Einleitung

Sie legen die Objekttypen, die der Bediener mit der Taste <Tab> im Modus "Schaltcursor" aktiviert, im Inspektorfenster fest.

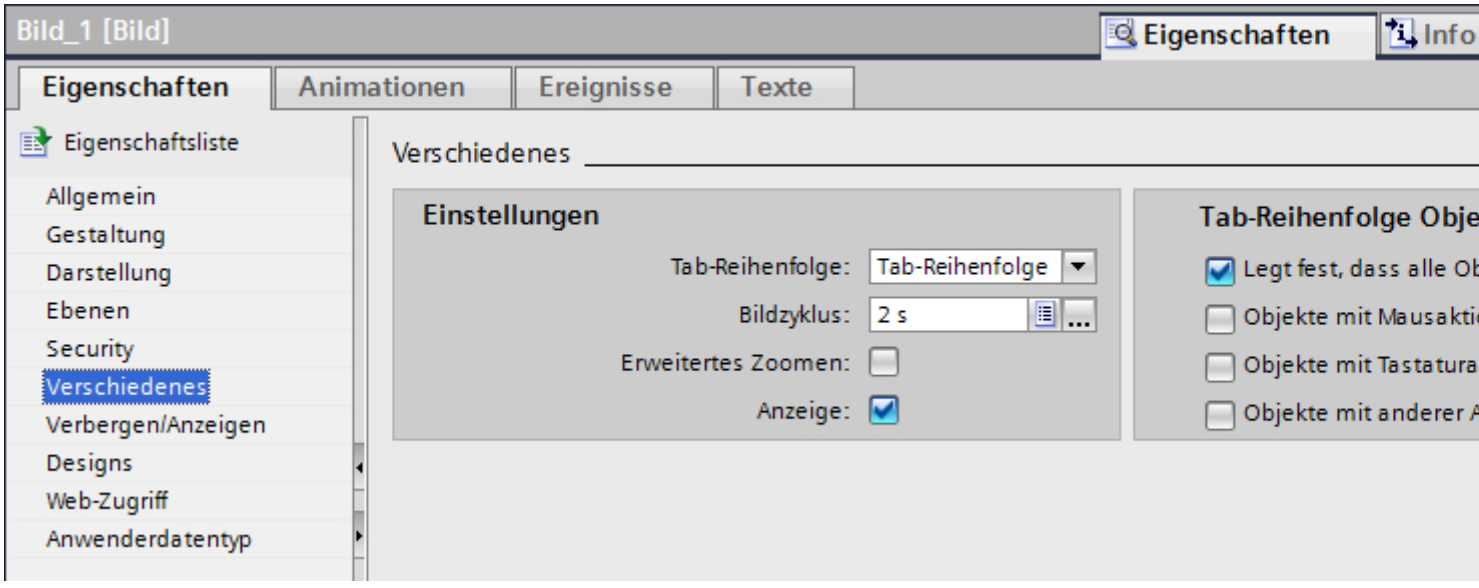

Voraussetzung

- Ein Bild mit mindestens einem Objekt ist geöffnet.
- Die Objekte sind zur Bedienung in Runtime freigegeben und haben eine Bedienberechtigung.
- Der Modus "Schaltcursor" ist festgelegt.

Vorgehen

1. Klicken Sie auf eine freie Fläche im Bild.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes > Tab-Reihenfolge Objekttypen" die Objekttypen aus, die mit der Taste <Tab> in Runtime bedient werden.

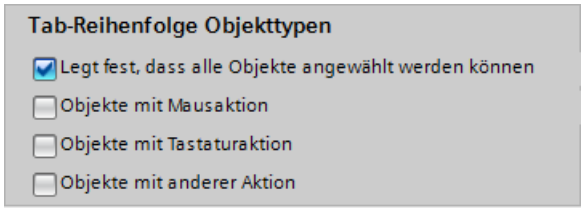

Ergebnis

In Runtime aktiviert der Benutzer im Cursormodus "Schaltcursor" mit der Taste <Tab> alle festgelegten Objekttypen.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3.6 Tab-Reihenfolge festlegen

Einleitung

In Runtime sind alle bedienbaren Objekte per Taste <Tab> erreichbar. Sie legen mit dem Befehl "Tab-Reihenfolge" die Reihenfolge fest, in der die Objekte in Runtime aktiviert werden.

Hinweis

Objekte mit dem Modus "Ausgabe" oder "Zwei Zustände" sind in Runtime nicht per Taste <Tab> erreichbar.

In Runtime bedienen Sie das Bild:

- Per Taste <Tab>
- Per Maus
- Per vorher eingestellten Hotkey

Voraussetzung

- Das aktive Bild enthält bedienbare Objekte.
- Kein Objekt ist markiert.
- Die Objekte sind zur Bedienung in Runtime freigegeben und haben eine Bedienberechtigung.

Vorgehen

1. Wählen Sie im Menü "Bearbeiten > Tab-Reihenfolge bearbeiten".
   Der Tab-Reihenfolgemodus wird aktiviert. Die Tab-Reihenfolge-Nummer wird bei allen bedienbaren Objekten angezeigt. Die Tab-Reihenfolge-Nummer wird auch bei verdeckten Objekten angezeigt.
2. Um den Tab-Reihenfolgemodus zu bearbeiten, klicken Sie die bedienbaren Objekte in der Reihenfolge an, in der die Objekte in Runtime mit <Tab> aktiviert werden sollen.
   Das folgende Bild zeigt, wie im Bild die Tab-Reihenfolge festgelegt wird. In Runtime aktiviert die Taste <Tab> zuerst die Meldeanzeige (Nummer 1), dann das E/A-Feld (Nummer 2) und dann die Schaltfläche (Nummer 3):

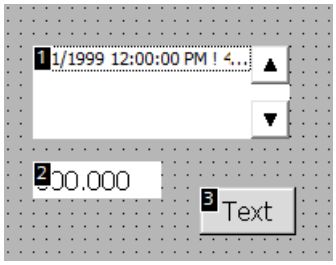

3. Um ein Bildobjekt aus der Tab-Reihenfolge auszuschließen, halten Sie die Tastenkombination <Shift+Strg> gedrückt und klicken Sie auf das gewünschte Objekt.
   Die Tab-Reihenfolge-Nummer wird nicht mehr im Bildobjekt angezeigt. Das Bildobjekt ist jetzt aus der Tab-Reihenfolge ausgeschlossen. Die verbleibenden Tab-Reihenfolge-Nummern werden automatisch um 1 erniedrigt.
4. Um ein ausgeschlossenes Bildobjekt wieder in die Tab-Reihenfolge aufzunehmen, wiederholen Sie Schritt 3.
   Das Bildobjekt wird als erstes Objekt in die Tab-Reihenfolge aufgenommen.

Ergebnis

In Runtime selektiert der Bediener mit der Taste <Tab> in der festgelegten Reihenfolge die Objekte.

Hinweis

Änderungen nach dem Hochrüsten auf eine neue Software-Version

Wenn Sie in einer älteren WinCC-Version in Bildern mit Bildbausteinen eine Tab-Reihenfolge projektiert hatten, überprüfen Sie die Tab-Reihenfolge dieser Bilder nach dem Hochrüsten auf eine neuere WinCC-Version. Die Tab-Reihenfolge kann sowohl im Bild als auch im Bildbaustein geändert sein.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.3.7 Tab-Reihenfolge festlegen

Einleitung

Im Modus "Schaltcursor" sind in Runtime beliebige Objekte bedienbar. Sie legen mit dem Befehl "Tab-Reihenfolge" die Reihenfolge fest, in der die Objekte in Runtime aktiviert werden. In Runtime bedienen Sie das Prozessbild:

- Per Taste <Tab>
- Per Maus
- Per vorher eingestellten Hotkey

Voraussetzung

- Ein Bild mit mindestens einem Objekt ist geöffnet.
- Die Objekte sind zur Bedienung in Runtime freigegeben und haben eine Bedienberechtigung.
- Der Modus "Schaltcursor" oder "Alphacursor" ist aktiviert.

Vorgehen

1. Wählen Sie im Menü "Bearbeiten > Tab-Reihenfolge > Schaltcursor bearbeiten" oder "Alphacursor bearbeiten".
   Der Tab-Reihenfolgemodus wird aktiviert. Die Tab-Reihenfolge-Nummer wird bei allen bedienbaren Objekten angezeigt. Die Tab-Reihenfolge-Nummer wird auch bei verdeckten Objekten angezeigt.
2. Um den Tab-Reihenfolgemodus zu bearbeiten, klicken Sie die bedienbaren Objekte in der Reihenfolge an, in der die Objekte in Runtime mit <Tab> aktiviert werden sollen.
   Das folgende Bild zeigt, wie im Bild die Tab-Reihenfolge festgelegt wird. In Runtime aktiviert die Taste <Tab> zuerst das E/A-Feld (Nummer 2), dann die Schaltfläche (Number 3):

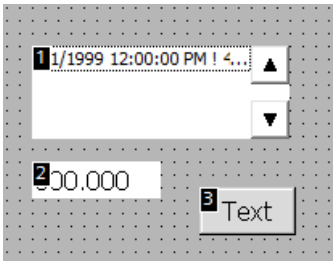

Ergebnis

In Runtime selektiert der Bediener mit der Taste <Tab> in der festgelegten Reihenfolge die Objekte.

Hinweis

Änderungen nach dem Hochrüsten auf eine neue Software-Version

Wenn Sie in einer älteren WinCC-Version in Bildern mit Bildbausteinen eine Tab-Reihenfolge projektiert hatten, überprüfen Sie die Tab-Reihenfolge dieser Bilder nach dem Hochrüsten auf eine neuere WinCC-Version. Die Tab-Reihenfolge kann sowohl im Bild als auch im Bildbaustein geändert sein.

Siehe auch

Übersicht zum Tastaturzugriff (Seite 168)

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.4 Beispiele

1.2.4.1 Beispiel: Rechteck einfügen und konfigurieren

Aufgabe

In diesem Beispiel fügen Sie ein Rechteck in ein Bild ein. Dabei konfigurieren Sie folgende Eigenschaften:

- Name = "MyRectangle"
- Position = (20, 20)
- Größe = (100,100)
- Farbe = rot
- 2 Pixel breiter schwarzer Rahmen

Prinzip

Das Rechteck ist ein geschlossenes Objekt, das Sie mit einer Farbe oder einem Muster füllen können. Die Höhe und Breite eines Rechtecks lassen sich beliebig verändern, sodass eine horizontale oder vertikale Ausrichtung möglich ist.

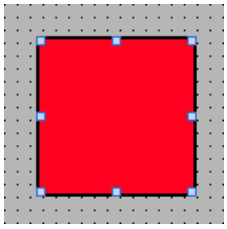

Übersicht

Um ein Rechteck zu erstellen, sind folgende Schritte notwendig:

- Rechteck einfügen
- Rechteck konfigurieren

Siehe auch

Grundlagen zu Gruppen (Seite 160)

Beispiel: Rechteck einfügen (Seite 177)

Beispiel: Rechteck konfigurieren (Seite 178)

Übersicht über die Objekte (Seite 95)

1.2.4.2 Beispiel: Rechteck einfügen

Aufgabe

In diesem Beispiel fügen Sie ein Rechteck ein und benennen es um. Verwenden Sie für den Namen nicht die Sonderzeichen ?, ", /, \, *, <, >.

Voraussetzung

- Ein Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

Vorgehen

1. Klicken Sie in der Task Card "Werkzeuge" auf die Palette "Basisobjekte".
2. Ziehen Sie das Objekt "Rechteck" in das Bild.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes".
4. Geben Sie als neuen Namen "MyRectangle" ein.

Ergebnis

Das Rechteck ist eingefügt und "MyRectangle" benannt. Das Rechteck besitzt die Standard-Eigenschaften des Objekts "Rechteck".

Siehe auch

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.2.4.3 Beispiel: Rechteck konfigurieren

Aufgabe

In diesem Beispiel konfigurieren Sie das Rechteck:

- Farbe = rot
- 2 Pixel breiter schwarzer Rahmen
- Position = (20, 20)
- Größe = (100,100)

Farbe des Rechtecks ändern

Um die Farbe des Rechtecks zu ändern:

1. Selektieren Sie das Rechteck.
2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Hintergrund > Farbe " die Hintergrundfarbe fest.
3. Wählen Sie "Massiv" als Füllmuster.
4. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Rahmen > Farbe " die Farbe für den Rahmen fest.
5. Geben Sie für "Breite" den Wert "2" ein.
6. Wählen Sie "Massiv" als "Stil".

Zwischenergebnis

Das Rechteck ist rot und hat einen 2 Pixel breiten schwarzen Rahmen.

Position und Größe des Rechtecks ändern

Um die Position und Größe des Rechtecks zu ändern:

1. Selektieren Sie das Rechteck.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Darstellung".

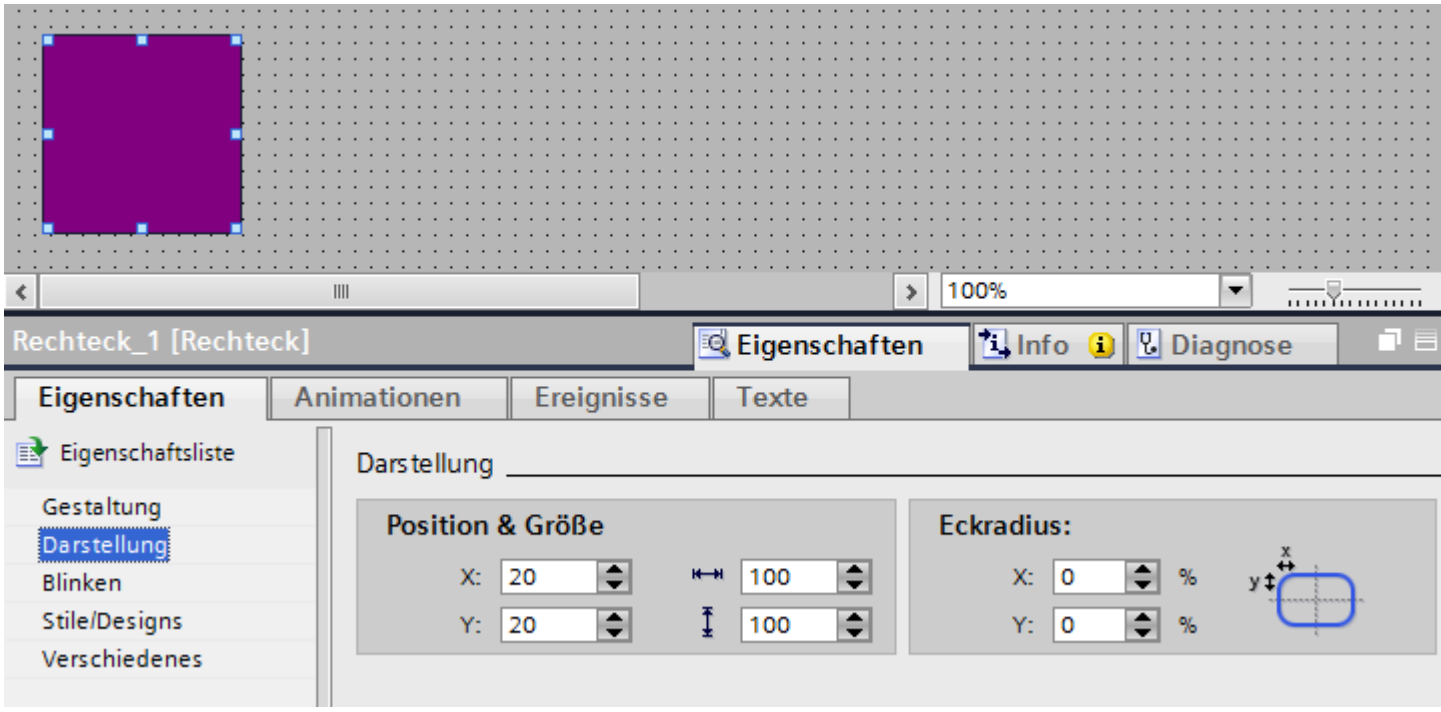

3. Geben Sie unter "Position & Größe" für X und Y jeweils den Wert "20" ein.
4. Geben Sie als Breite und Höhe jeweils den Wert "100" ein.

Ergebnis

Das Rechteck ist an den Koordinaten (20, 20) positioniert und ist je 100 Pixel breit und hoch.

Siehe auch

Beispiel: Rechteck einfügen und konfigurieren (Seite 176)

1.3 Arbeiten mit Textlisten und Grafiklisten

1.3.1 Arbeiten mit Textlisten

1.3.1.1 Grundlagen zu Textlisten

Einleitung

In einer Textliste sind Texte den Werten einer Variablen zugeordnet. Bei der Projektierung ordnen Sie die Textliste z. B. einem Symbolischen E/A-Feld zu. Damit liefern Sie dem Objekt die anzuzeigenden Texte. Im Editor "Textlisten" erstellen Sie die Textlisten. Die Anbindung der Textliste an eine Variable projektieren Sie an dem Objekt, das die Textliste verwendet.

Die Auswahl der Objekte, die mit einer Textliste versehen werden können, hängt von der Runtime ab.

Verwendung

Sie verwenden die Textliste z. B. für die Anzeige einer Auswahlliste in einem Symbolischen E/A-Feld.

Wenn das Symbolische E/A-Feld ein Anzeigefeld ist, werden die zugehörigen Texte abhängig vom Wert der projektierten Variablen angezeigt. Wenn das Symbolische E/A-Feld ein Eingabefeld ist, nimmt die projektierte Variable den zugehörigen Wert an, sobald der Bediener in Runtime den entsprechenden Text auswählt.

Hinweis

Anzeige von Variablenwerten ohne Text

Die Darstellung von Variablenwerten, denen kein Text zugeordnet wurde, ist von der Runtime abhängig:

- Anzeige- und Bedienobjekt bleibt leer.
- Drei Sternchen *** werden angezeigt.

Bereiche für die Textliste

Für die Textlisten stehen Ihnen drei Typen zur Auswahl:

- Wert/Bereich
  Mit dieser Einstellung werden ganzzahligen Werten bzw. Wertebereichen einer Variablen die Texteinträge aus der Textliste zugeordnet. Die Anzahl der Texteinträge können Sie frei wählen. Die maximale Anzahl der Einträge ist abhängig vom verwendeten Bediengerät. Sie legen einen Standardwert fest, der dann angezeigt wird, wenn der Wert der Variablen außerhalb des definierten Bereichs liegt.
- Bit (0, 1)
  Mit dieser Einstellung werden zwei Zuständen einer Binär-Variablen die Texteinträge aus der Textliste zugeordnet. Für jeden Zustand der Binär-Variable können Sie einen Texteintrag erstellen.
- Bitnummer (0 - 31)
  Mit dieser Einstellung wird jedem Bit einer Variablen ein Texteintrag aus der Textliste zugeordnet. Die maximale Anzahl der Texteinträge beträgt 32. Diese Form der Textliste kann z. B. in einer Ablaufsteuerung beim Abarbeiten einer Schrittkette verwendet werden, wo immer nur ein Bit der verwendeten Variable gesetzt sein darf. Über das gesetzte Bit mit der kleinsten Wertigkeit und einen Standardwert beeinflussen Sie das Verhalten von Bitnummer (0 - 31).

Mehrsprachige Texte

Sie können die Texte in einer Textliste mehrsprachig projektieren. Die Texte werden in Runtime in der eingestellten Runtime-Sprache angezeigt. Dazu stellen Sie die Sprachen im Projektfenster unter "Sprachunterstützung > Projektsprachen" ein.

Projektierungsschritte

Zur Anzeige von Texten z. B. in einem Symbolischen E/A-Feld sind folgende Schritte erforderlich:

1. Erstellen der Textliste.
2. Zuordnung der Texte zu Werten bzw. Wertebereichen einer Textliste.
3. Zuordnung einer Textliste im Anzeigeobjekt, z. B. dem Symbolischen E/A-Feld.

Siehe auch

Textliste erstellen (Seite 182)

Texte und Werte einer Bereich-Textliste zuordnen (Seite 185)

Texte und Werte einer Bit-Textliste zuordnen (Seite 187)

Texte und Werte einer Bitnummer-Textliste zuordnen (Seite 188)

Hinweise zur Bitnummer-Textliste (Seite 190)

Objekt mit Textliste projektieren (Seite 193)

Grundlagen zu Bildern (Seite 33)

Texte aus Textliste in Meldung ausgeben (Seite 835)

Textliste im Bildbaustein-Typ projektieren (Seite 292)

1.3.1.2 Textliste erstellen

Einleitung

Die Textliste bietet die Möglichkeit, Werten spezifische Texte zuzuordnen und diese in Runtime, z.B. in einem Symbolischen E/A-Feld, auszugeben. Dabei kann der Typ des Symbolischen E/A-Feldes angegeben werden, z.B. als reines Eingabefeld.

Folgende Listenarten stehen zur Verfügung:

- Wert/Bereich
- Bit
- Bitnummer

Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf "Text- und Grafiklisten".
2. Öffnen Sie die Registerkarte "Textlisten".

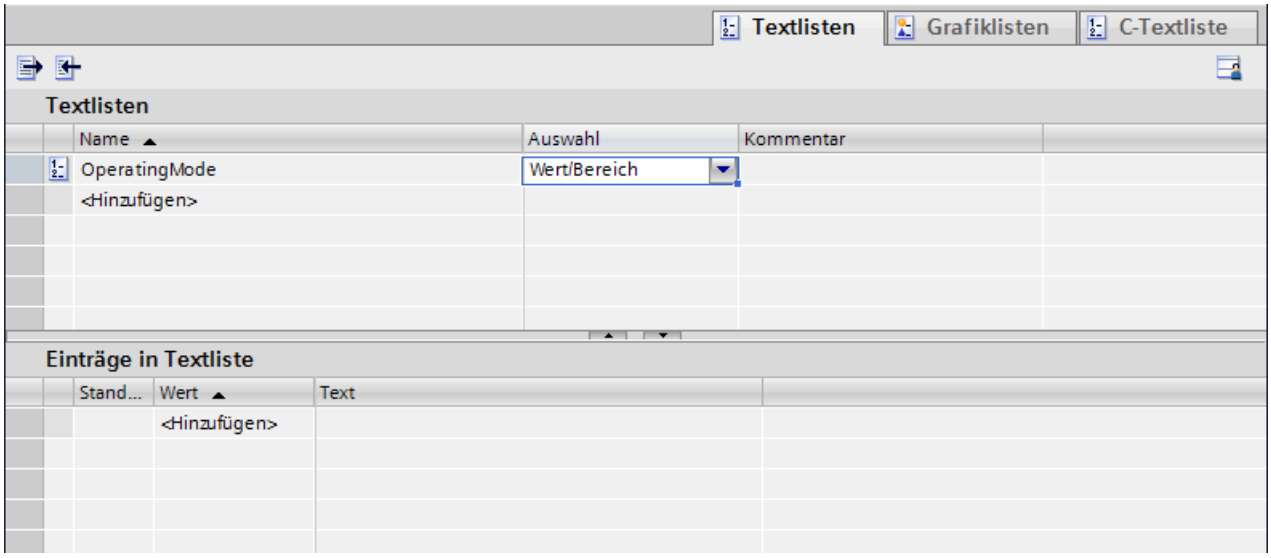

3. Klicken Sie in der Tabelle "Textlisten" auf "Hinzufügen".
   Das Inspektorfenster zur Textliste wird geöffnet.
4. Geben Sie der Textliste einen aussagekräftigen Namen.

5. Wählen Sie unter "Auswahl" den Textlistentyp:
   - Wert/Bereich: Text aus Textliste wird angezeigt, wenn die Variable einen Wert im angegebenen Bereich angenommen hat.
   - Bit (0,1): Ein Text aus der Textliste wird angezeigt, wenn die Variable den Wert 0 angenommen hat, ein anderer Text aus der Textliste wird angezeigt, wenn die Variable den Wert 1 angenommen hat.
   - Bitnummer (0-31): Text aus der Textliste wird angezeigt, wenn die Variable die zugeordnete Bitnummer angenommen hat.
6. Geben Sie einen Kommentar für die Textliste ein.

Hinweis

In den Texten einer Textliste in WinCC Runtime Professional können Sie kein Semikolon verwenden. Das Semikolon ist ein Steuerzeichen und wird in einem Text automatisch gelöscht.

Ergebnis

Eine Textliste ist erstellt.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

Textliste im Bildbaustein-Typ projektieren (Seite 292)

1.3.1.3 Texte und Werte einer Bereich-Textliste zuordnen

Einleitung

Zu jeder Bereich-Textliste geben Sie an, bei welchem Wertebereich welche Texte angezeigt werden.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bereich-Textliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Bereich".

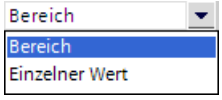

   – Geben Sie für "Min" z. B. den Wert "1" ein.
   – Geben Sie für "Max" z. B. den Wert "20" ein.
   – Geben Sie für "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.

   Hinweis

   Verwenden Sie maximal 320 Zeichen für den Text.

3. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen". Ein zweiter Listeneintrag wird erstellt.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Bereich".
   – Geben Sie für "Min" z. B. den Wert "21" ein.
   – Geben Sie für "Max" z. B. den Wert "40" ein.
   – Geben Sie für "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet
5. Bei Bedarf aktivieren Sie "Standardeintrag".
   Der eingegebene Text erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.

Ergebnis

Eine Bereich-Textliste ist erstellt. Den möglichen Wertebereichen sind Texte zugeordnet.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.1.4 Texte und Werte einer Bereich-Textliste zuordnen

Einleitung

Zu jeder Bereich-Textliste geben Sie an, bei welchem Wertebereich welche Texte angezeigt werden.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bereich-Textliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" eine der folgenden Einstellungen aus.

   – "Bereich": minimaler bis maximaler Variablenwert, z. B. 1 ≦ Wert ≦ 21
   – "Bis": maximaler Variablenwert, z. B. Wert ≦ 13
   – "Einzelner Wert": genau ein Variablenwert, z. B. Wert = 21
   – "Von": minimaler Variablenwert, z. B. Wert ≧ 2
3. Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable den angegebenen Wert angenommen hat oder sich innerhalb des angegebenen Wertebereichs befindet.

   Hinweis

   Verwenden Sie kein Semikolon und maximal 255 Zeichen für den Text.

4. Bei Bedarf aktivieren Sie "Standardeintrag".
   Der eingegebene Text erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.
5. Legen Sie für weitere Wertebereiche derselben Textliste zusätzliche entsprechende Listeneinträge an.

Ergebnis

Eine Bereich-Textliste ist erstellt. Den möglichen Werten sind Texte zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.1.5 Texte und Werte einer Bit-Textliste zuordnen

Einleitung

Zu jeder Textliste geben Sie an, bei welchem Bit-Wert welcher Text angezeigt wird.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bit-Textliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
   - Geben Sie für "Wert" "0" ein.
   - Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Bitvariable auf "0" gesetzt wird.

3. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen". Ein zweiter Listeneintrag wird erstellt.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
   – Geben Sie unter "Wert" "1" ein.
   – Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Bitvariable auf "1" gesetzt wird.

Hinweis

Verwenden Sie maximal 320 Zeichen für den Text.

Verwenden Sie bei WinCC Runtime Professional kein Semikolon und maximal 255 Zeichen für den Text.

Ergebnis

Eine Bit-Textliste ist erstellt. Den möglichen Werten "0" und "1" sind Texte zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.1.6 Texte und Werte einer Bitnummer-Textliste zuordnen

Einleitung

Zu jeder Bitnummer-Textliste geben Sie an, bei welcher Bitnummer welche Texte angezeigt werden.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bitnummer-Textliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
   - Geben Sie für "Wert" z. B. "10" ein.
   - Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable den Wert "10" angenommen hat.
3. Bei Bedarf aktivieren Sie "Standardeintrag".
   Der eingegebene Text erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.
4. Legen Sie für weitere Bitnummern derselben Textliste weitere Listeneinträge an.

Hinweis

Verwenden Sie maximal 320 Zeichen für den Text.

Verwenden Sie bei WinCC Runtime Professional kein Semikolon und maximal 255 Zeichen für den Text.

Hinweis

Bitselektion für Textlisten

Die Ausgabe der Textliste hängt von der Option „Bitselektion für Text- und Grafiklisten“ in Runtime-Einstellungen ab. Mit dieser Option legen Sie fest, dass die Bitauswahl für Textlisten auf Ihrem Bediengerät verwendet wird. Wenn Sie die Option aktivieren, wird jeweils der Text angezeigt, der beim gesetzten Bit mit der kleinsten Wertigkeit projektiert ist. Wenn Sie diese Option deaktivieren, wird der Text angezeigt, der nur beim gesetzten Bit projektiert ist.

Ergebnis

Eine Bitnummer-Textliste ist erstellt. Den angegebenen Bitnummern sind Texte zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.1.7 Hinweise zur Bitnummer-Textliste

Einleitung

Mit dem Bereich Bitnummer (0-31) wird jedem Bit einer Variablen ein Texteintrag aus der Liste zugeordnet. Wenn in Runtime-Einstellungen die Option "Bitselektion für Text- und Grafiklisten" deaktiviert ist und kein Standardwert gesetzt ist, gilt folgendes Standardverhalten:

- Wenn von allen gesetzten Bits nur 1 Bit projektiert ist, wird beim projektierten Bit das hinterlegte Text angezeigt.
  Im folgenden Beispiel ist nur das gesetzte Bit mit der Wertigkeit "4" projektiert. Text 2 wird angezeigt.

Wertigkeit	7	6	5	4	3	2	1	0
Gesetzte Bits	0	0	1	1	0	1	0	0
Projektiert	-	Text 3	-	Text 2	Text 1	-	-	-

- Wenn kein Bit gesetzt ist oder mehrere projektierte Bits gesetzt sind, wird kein Text angezeigt.

Standardwert

Um eine leere Anzeige zu vermeiden, legen Sie einen Standardwert fest. Ein projektierter Standardwert wird in folgenden Fällen angezeigt:

- Die Option "Bitselektion für Text- und Grafiklisten" ist deaktiviert. In der Variablen ist kein Bit oder mehrere projektierte Bits gesetzt sind.
- Die Option "Bitselektion für Text- und Grafiklisten" ist aktiviert und kein Bit gesetzt ist oder beim gesetzten Bit mit der kleinsten Wertigkeit ist kein Text projektiert.

Standardwert anzeigen

1. Aktivieren Sie in der Spalte "Standard" der Tabelle "Einträge in Textliste" den Text für den Standardeintrag.
   In der Spalte "Wert" des Texteintrags erscheint der Wert "Standardeintrag".
2. Alternativ aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Option "Standard".

Gesetztes Bit mit der kleinsten Wertigkeit

Wenn beim gesetzten Bit mit der kleinsten Wertigkeit kein Text projektiert ist und auch kein

Standardwert, wird nichts angezeigt. Wenn ein Standardwert projektiert ist, wird der Standardwert angezeigt.

Wenn beim gesetzten Bit mit der kleinsten Wertigkeit kein Text projektiert ist und auch kein Standardwert, wird nichts angezeigt. Wenn ein Standardwert projektiert ist, wird der Standardwert angezeigt.

Um jeweils nur den Text beim gesetzten Bit mit der kleinsten Wertigkeit anzuzeigen, aktivieren Sie im Editor "Runtime-Einstellungen" die Option "Bitselektion in Text- und Grafiklisten".

Wegen der Abwärtskompatibilität ist die Einstellung standardmäßig deaktiviert. Die Einstellung gilt für alle Textlisten des Bediengeräts.

Mehrzeilige Textlisteneinträge

Mit der Tastenkombination <SHIFT>+<RETURN> geben Sie einen Zeilenumbruch in den Texteintrag ein. Die Zeilenumbrüche werden mit dem Absatzzeichen "¶" dargestellt.

Mehrzeilige Textlisteneinträge werden nur in symbolischen Ausgabefeldern sowie auf Schaltflächen mehrzeilig ausgegeben. In allen anderen Fällen werden die mehrzeiligen Texte mit dem Absatzzeichen dargestellt.

1.3.1.8 Objekt mit Textliste projektieren

Einleitung

Ausgabewert und Wertübernahme für Textlisten werden in dem Anzeige- und Bedienobjekt festgelegt, das die Texte der Textliste in Runtime anzeigt. Die Eigenschaften dieser Objekte werden nach Bedarf konfiguriert.

Voraussetzung

- Eine Textliste ist erstellt.
- Eine Variable ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Ein Bild mit einem Symbolischen E/A-Feld ist geöffnet. Das Objekt ist selektiert.

Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Textliste" die Textliste aus, die Sie in Runtime anzeigen lassen wollen.
2. Wählen Sie als "Modus" die Einstellung "Ausgabe".

   Hinweis

   Runtime-Abhängigkeit

   Je nach Runtime stehen unterschiedliche Feldtypen für ein Symbolisches E/A-Feld zur Verfügung.

3. Wählen Sie als "Variable" die Variable, deren Werte die Anzeige im Symbolischen E/A-Feld bestimmt.

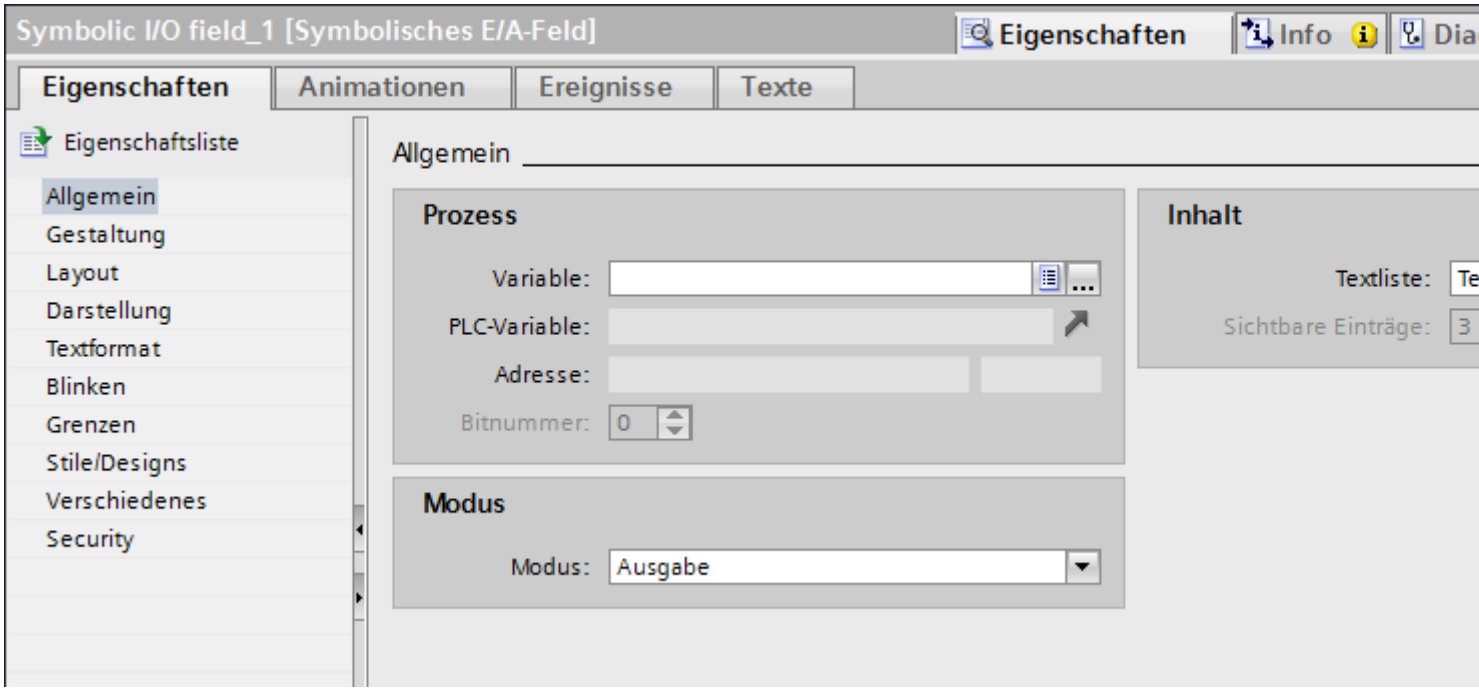

Ergebnis

Wenn die Variable den angegebenen Wert annimmt, werden in Runtime die definierten Texte der Textliste im Symbolischen E/A-Feld angezeigt.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.1.9 Objekt mit Textliste projektieren

Einleitung

Ausgabewert und Wertübernahme für Textlisten werden in dem Anzeige- und Bedienobjekt festgelegt, das die Texte der Textliste in Runtime anzeigt. Die Eigenschaften dieser Objekte werden nach Bedarf konfiguriert.

Voraussetzung

- Eine Textliste ist erstellt.
- Eine Variable ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Ein Bild mit einem Symbolischen E/A-Feld ist geöffnet. Das Objekt ist selektiert.

Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Textliste" die Textliste aus, die Sie in Runtime anzeigen lassen wollen.
2. Wählen Sie als "Modus" die Einstellung "Ausgabe".

   Hinweis

   Runtime-Abhängigkeit

   Je nach Runtime stehen unterschiedliche Feldtypen für ein Symbolisches E/A-Feld zur Verfügung.

3. Wählen Sie als "Variable" die Variable, deren Werte die Anzeige im Symbolischen E/A-Feld bestimmt.

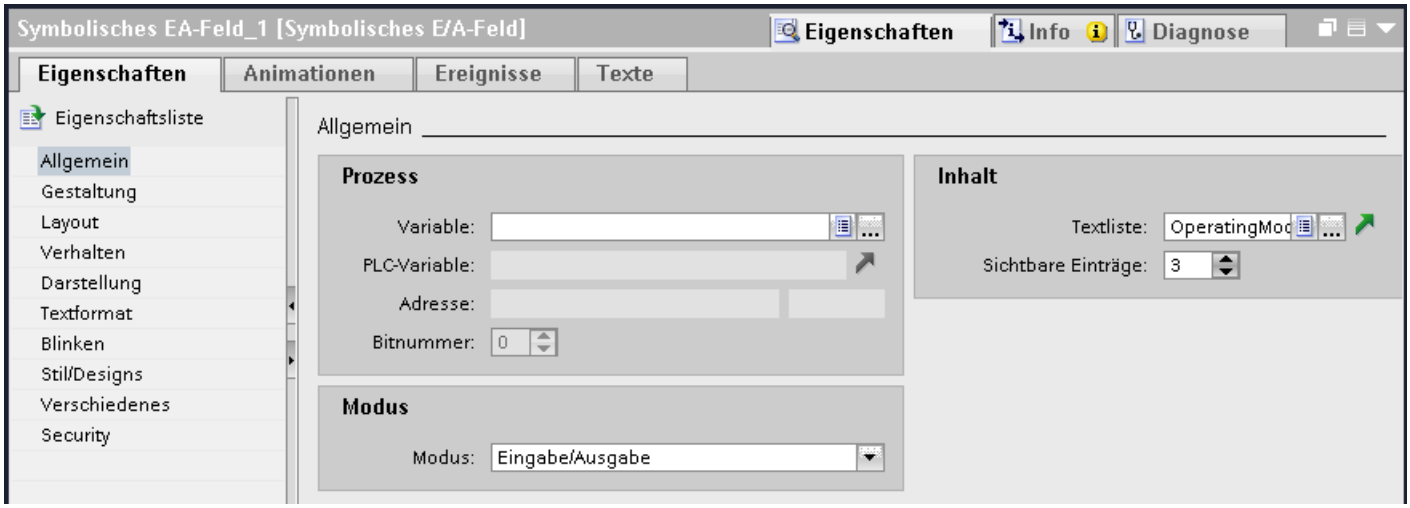

Ergebnis

Wenn die Variable den angegebenen Wert annimmt, werden in Runtime die definierten Texte der Textliste im Symbolischen E/A-Feld angezeigt.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

1.3.2 Arbeiten mit Grafiklisten

1.3.2.1 Grundlagen zu Grafiklisten

Einleitung

In einer Grafikliste sind bestimmte Grafiken den möglichen Werten einer Variablen zugeordnet. Bei der Projektierung ordnen Sie die Grafikliste einer Schaltfläche oder einem Grafischen E/A-Feld zu. Damit liefern Sie dem Objekt die anzuzeigenden Grafiken.

Sie erstellen die Grafiklisten mit dem Editor "Text- und Grafiklisten". Die Anbindung der Grafikliste an eine Variable projektieren Sie an dem Objekt, das die Grafikliste verwendet. Die Verfügbarkeit der Grafikliste hängt vom verwendeten Bediengerät ab.

Verwendung

Sie projektieren die Grafikliste für folgende Anwendungsfälle:

- Auswahlliste mit einem Grafischen E/A-Feld
- Zustandsabhängige Grafik auf einer Schaltfläche

Sie können die Grafiken in einer Grafikliste mehrsprachig projektieren. In Runtime werden die Grafiken der eingestellten Runtime-Sprache angezeigt.

Grafikquellen

Die Grafiken können aus folgenden Quellen in die Grafikliste eingefügt werden:

- Auswahl aus der Grafiksammlung
- Auswahl einer vorhandenen Datei
  Sie können folgende Dateitypen verwenden:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tiff, *.png, *.svg, *.jpeg und *.jpg.
- Erstellen einer neuen Datei

Funktion

Wenn das Grafische E/A-Feld ein Anzeigefeld ist, werden abhängig vom Wert der projektierten Variablen die zugehörigen Grafiken angezeigt. Wenn das Grafische E/A-Feld ein Eingabefeld ist, nimmt die projektierte Variable den zugehörigen Wert an, sobald der Bediener in Runtime eine Grafik auswählt.

Bereiche für die Grafikliste

Für die Grafiklisten stehen Ihnen drei Typen zur Auswahl:

- Wert/Bereich
  Mit dieser Einstellung werden ganzzahligen Werten bzw. Wertebereichen einer Variablen die Grafikeinträge aus der Grafikliste zugeordnet. Die Anzahl der Grafikeinträge können Sie frei wählen. Die maximale Anzahl der Einträge ist abhängig vom verwendeten Bediengerät.
  Sie legen einen Standardwert fest, der dann angezeigt wird, wenn der Wert der Variablen außerhalb des definierten Bereichs liegt.
- Bit (0, 1)
  Mit dieser Einstellung werden zwei Zuständen einer Binär-Variablen die Grafikeinträge aus der Grafikliste zugeordnet. Für jeden Zustand der Binär-Variable können Sie einen Grafikeintrag erstellen.
- Bitnummer (0 - 31)
  Mit dieser Einstellung wird jedem Bit einer Variablen ein Grafikeintrag aus der Grafikliste zugeordnet. Die maximale Anzahl der Grafikeinträge beträgt 32. Diese Form der Grafikliste kann z. B. in einer Ablaufsteuerung beim Abarbeiten einer Schrittkette verwendet werden, wo immer nur ein Bit der verwendeten Variable gesetzt sein darf. Über das gesetzte Bit mit der kleinsten Wertigkeit und einen Standardwert beeinflussen Sie das Verhalten von Bitnummer (0 - 31).

Projektierungsschritte

Zur Anzeige von Grafiken z.B. in einem Grafischen E/A-Feld sind folgende Schritte erforderlich:

1. Erstellen der Grafikliste.
2. Zuordnung der Grafiken zu Werten bzw. Wertebereichen einer Grafikliste.
3. Zuordnung einer Grafikliste im Anzeigeobjekt, z.B. dem Grafischen E/A-Feld.

Siehe auch

Grafikliste erstellen (Seite 197)

Grafik und Werte einer Bereich-Grafikliste zuordnen (Seite 198)

Grafiken und Werte einer Bit-Grafikliste zuordnen (Seite 202)

Grafiken und Werte einer Bitnummer-Grafikliste zuordnen (Seite 203)

Hinweise zur Bitnummer-Grafikliste (Seite 205)

Objekt mit Grafikliste projektieren (Seite 206)

Grafikliste im Bildbaustein-Typ projektieren (Seite 293)

Grundlagen zu Bildern (Seite 33)

1.3.2.2 Grafikliste erstellen

Einleitung

Die Grafikliste bietet die Möglichkeit, variablen Werten spezifische Grafiken zuzuordnen und diese in Runtime in einem Grafische E/A-Feld auszugeben. Dabei kann der Typ des Grafischen E/A-Feldes angegeben werden, z.B. als reines Ausgabefeld.

Vorgehen

1. Doppelklicken Sie Sie im Projektfenster auf "Text- und Grafiklisten"
2. Öffnen Sie die Registerkarte "Grafiklisten".

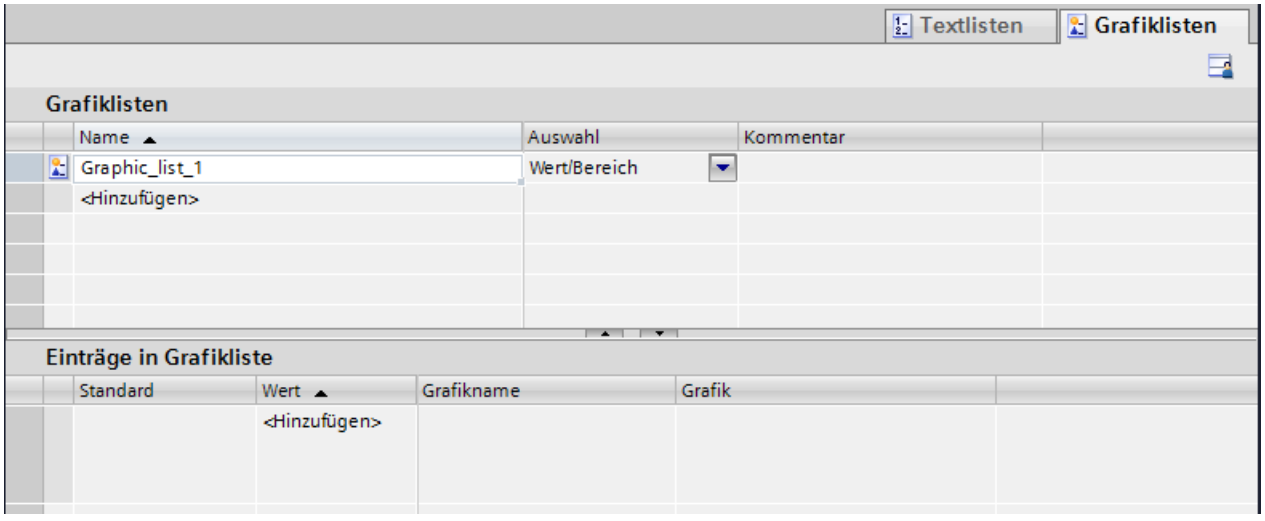

3. Klicken Sie in der Tabelle "Grafiklisten" auf "Hinzufügen". Das Inspektorfenster zur Grafikliste wird geöffnet.

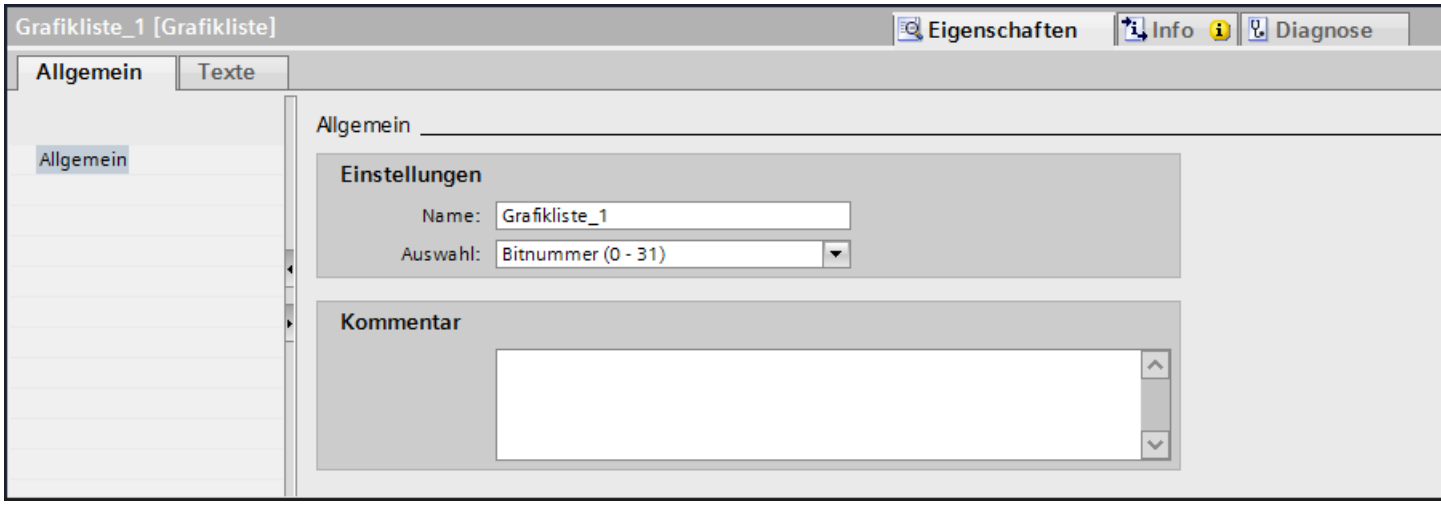

4. Geben Sie der Grafikliste einen aussagekräftigen Namen.
5. Wählen Sie unter "Auswahl" z. B. den Grafiklistentyp "Bitnummer (0 - 31)"
6. Geben Sie einen Kommentar für die Grafikliste ein.

Ergebnis

Eine Grafikliste vom Typ "Bereich (0 - 31)" ist erstellt.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

1.3.2.3 Grafikliste erstellen

Einleitung

Die Grafikliste bietet die Möglichkeit, variablen Werten spezifische Grafiken zuzuordnen und diese in Runtime in einem Grafische E/A-Feld auszugeben. Dabei kann der Typ des Grafischen E/A-Feldes angegeben werden, z. B. als reines Ausgabefeld.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Text- und Grafiklisten"
2. Öffnen Sie die Registerkarte "Grafiklisten".

3. Klicken Sie in der Tabelle "Grafiklisten" auf "Hinzufügen".
   Das Inspektorfenster zur Grafikliste wird geöffnet.

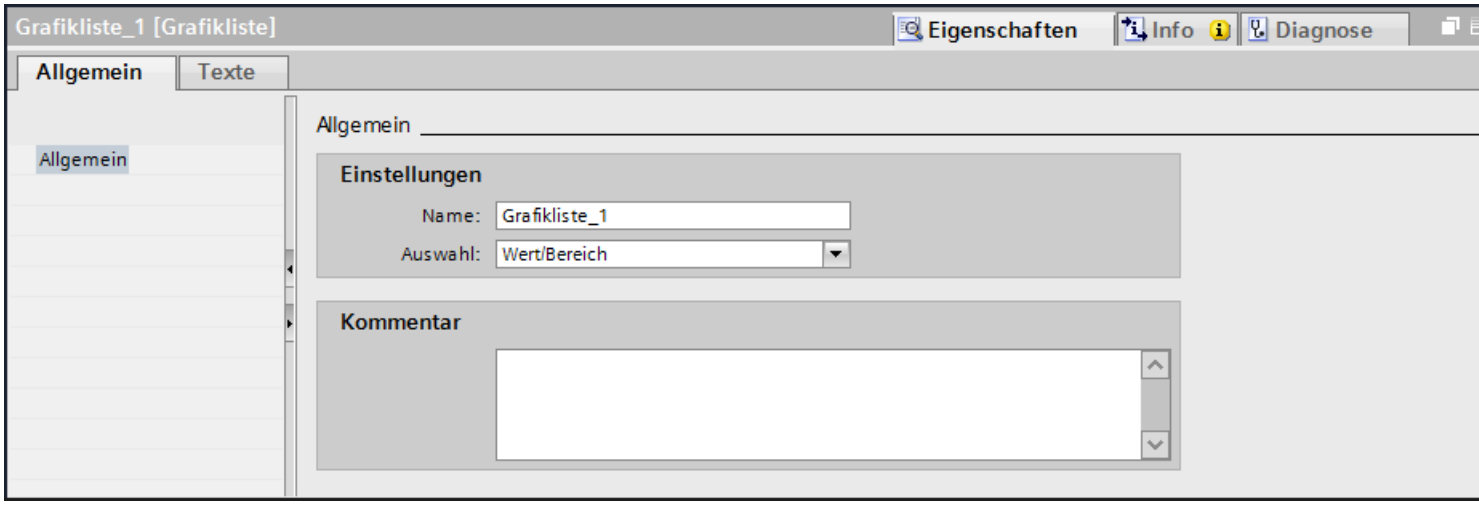

4. Geben Sie der Grafikliste einen aussagekräftigen Namen.
5. Wählen Sie unter "Auswahl" den Grafiklistentyp "Wert/Bereich".
6. Geben Sie einen Kommentar für die Grafikliste ein.

Ergebnis

Eine Bereich-Grafikliste ist erstellt.

Siehe auch

Grafikliste erstellen (Seite 196)

Grafikliste im Bildbaustein-Typ projektieren (Seite 293)

1.3.2.4 Grafik und Werte einer Bereich-Grafikliste zuordnen

Einleitung

Zu jeder Bereich-Grafikliste geben Sie an, bei welchem Wertebereich welche Grafiken angezeigt werden.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafiklisten" ist geöffnet.
- Eine Bereich-Grafikliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

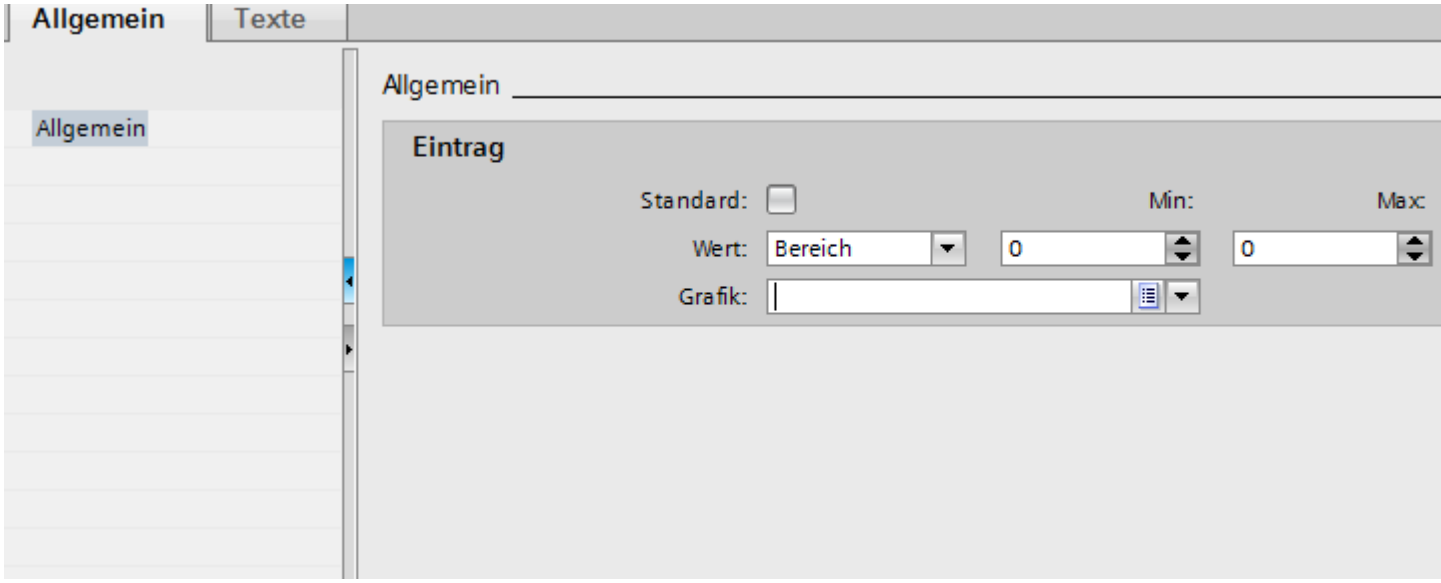

2. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Bereich":
   - Geben Sie für "Min" z. B. den Wert "1" ein.
   - Geben Sie für "Max" z. B. den Wert "20" ein.
   - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.

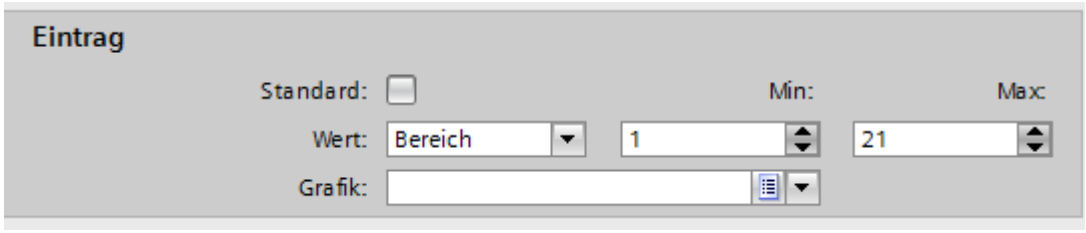

Hinweis

Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen". Ein weiterer Listeneintrag wird erstellt.

4. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
   – Geben Sie z. B. als Wert "21" ein.
   – Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "21" gesetzt wird.

5. Bei Bedarf aktivieren Sie "Standardeintrag".
   Die Grafik erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.

Ergebnis

Eine Bereich-Grafikliste ist erstellt. Den möglichen Werten sind Grafiken zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

1.3.2.5 Grafiken und Werte einer Bereich-Grafikliste zuordnen

Einleitung

Zu jeder Bereich-Grafikliste geben Sie an, bei welchem Wertebereich welche Grafiken angezeigt werden.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafiklisten" ist geöffnet.
- Eine Bereich-Grafikliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

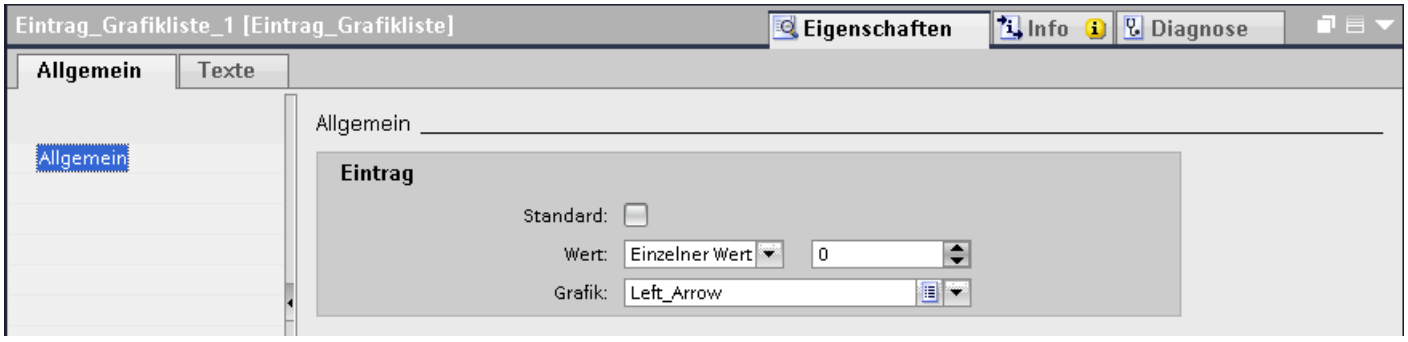

2. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
   – Geben Sie als Wert z. B. "0" ein.
   – Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn das Bit 0 gesetzt wird

Hinweis

Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Bei Bedarf aktivieren Sie "Standardeintrag".
   Die Grafik erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.
4. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Transparente Farbe verwenden" und wählen Sie eine Farbe aus.
5. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Blinken > Blinken verwenden".
6. Wählen Sie z. B. "Blinkfrequenz > Langsam".
7. Legen Sie für weitere Bitnummern derselben Grafikliste weitere Listeneinträge an.

Ergebnis

Eine Bereich-Grafikliste ist erstellt. Den angegebenen Bitnummern sind Grafiken zugeordnet, die in Runtime erscheinen. Die Grafik blinkt, wenn die Variable einen bestimmten Wert annimmt.

1.3.2.6 Grafiken und Werte einer Bit-Grafikliste zuordnen

Einleitung

Zu jeder Grafikliste geben Sie an, bei welchem Bit-Wert welche Grafik angezeigt wird.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafikliste" ist geöffnet.
- Eine Bit-Grafikliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

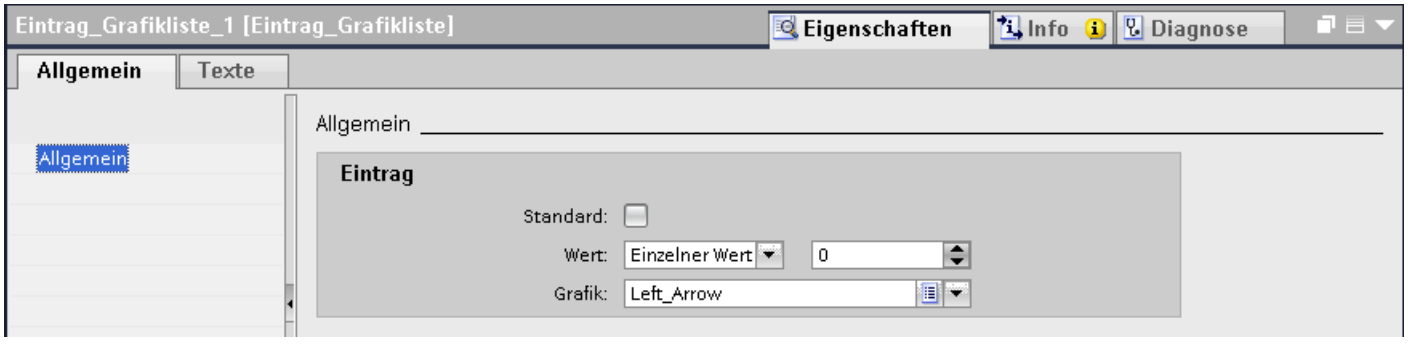

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
   - Geben Sie als Wert "0" ein.
   - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "0" gesetzt wird.

   Hinweis

   Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:
   1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
   2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen". Ein neuer Listeneintrag wird erstellt.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert > Einzelner Wert":
   – Geben Sie als Wert "1" ein.
   – Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "1" gesetzt wird.

Ergebnis

Eine Bit-Grafikliste ist erstellt. Den Werten "0" und "1" sind Grafiken zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

1.3.2.7 Grafiken und Werte einer Bitnummer-Grafikliste zuordnen

Einleitung

Zu jeder Bitnummer-Grafikliste geben Sie an, bei welcher Bitnummer welche Grafiken angezeigt werden sollen.

Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafiklisten" ist geöffnet.
- Eine Bitnummer-Grafikliste ist angelegt und selektiert.

Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".
   Das Inspektorfenster für diesen Listeneintrag wird geöffnet.

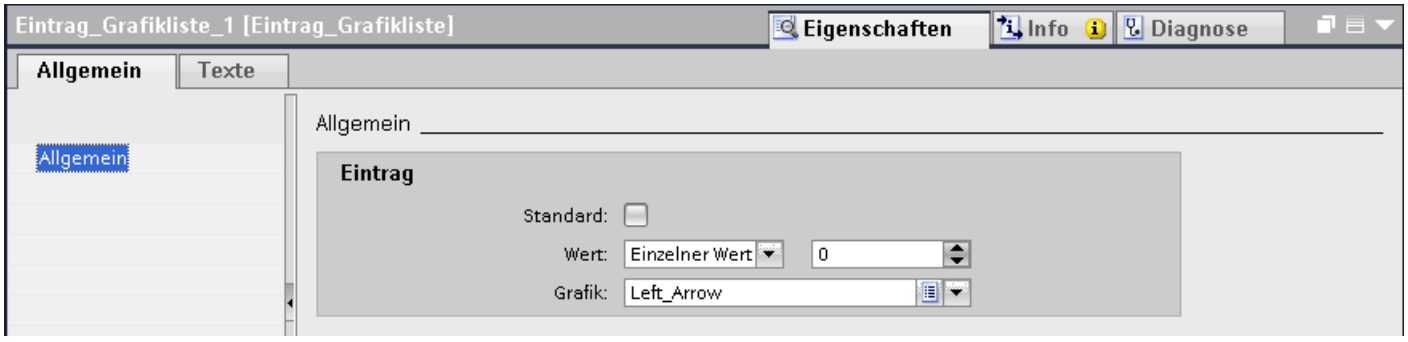

2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
   - Geben Sie als Wert z. B. "1" ein.
   - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "0" gesetzt wird

Hinweis

Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Bei Bedarf aktivieren Sie "Standardeintrag".
   Die Grafik erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.
4. Legen Sie für weitere Bitnummern derselben Grafikliste weitere Listeneinträge an.

Hinweis

Bitselektion für Grafiklisten

Die Ausgabe der Grafikliste hängt von der Option „Bitselektion für Text- und Grafiklisten“ in Runtime-Einstellungen ab. Mit dieser Option legen Sie fest, dass die Bitauswahl für Grafiklisten auf Ihrem Bediengerät verwendet wird. Wenn Sie die Option aktivieren, wird jeweils die Grafik angezeigt, die beim gesetzten Bit mit der kleinsten Wertigkeit projektiert ist. Wenn Sie diese Option deaktivieren, wird die Grafik angezeigt, die nur beim gesetzten Bit projektiert ist.

Ergebnis

Eine Bitnummer-Grafikliste ist erstellt. Den angegebenen Bitnummern sind Grafiken zugeordnet, die in Runtime erscheinen.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

1.3.2.8 Hinweise zur Bitnummer-Grafikliste

Einleitung

Mit dem Bereich Bitnummer (0-31) wird jedem Bit einer Variablen ein Grafikeintrag aus der Liste zugeordnet. Wenn in Runtime-Einstellungen die Option "Bitselektion für Text- und Grafiklisten" deaktiviert ist und kein Standardwert gesetzt ist, gilt folgendes Standardverhalten:

- Wenn von allen gesetzten Bits nur 1 Bit projektiert ist, wird beim projektierten Bit die hinterlegte Grafik angezeigt.
  Im folgenden Beispiel ist nur das gesetzte Bit mit der Wertigkeit "4" projektiert. Grafik 2 wird angezeigt.

Wertigkeit	7	6	5	4	3	2	1	0
Gesetzte Bits	0	0	1	1	0	1	0	0
Projektiert	-	Grafik 3	-	Grafik 2	Grafik 1	-	-	-

- Wenn kein Bit gesetzt ist oder mehrere Bits gesetzt sind, die auch projektiert sind, wird nur die Kakteengrafik angezeigt.

Standardwert

Um eine leere Anzeige zu vermeiden, legen Sie einen Standardwert fest. Ein projektierter Standardwert wird in folgenden Fällen angezeigt:

- Die Option "Bitselektion für Text- und Grafiklisten" ist deaktiviert. In der Variablen ist kein Bit oder mehrere projektierte Bits gesetzt sind.
- Die Option "Bitselektion für Text- und Grafiklisten" ist aktiviert und kein Bit gesetzt ist oder beim gesetzten Bit mit der kleinsten Wertigkeit ist keine Grafik projektiert.

Standardwert anzeigen

1. Aktivieren Sie in der Spalte "Standard" der Tabelle "Einträge in Grafikliste" die Grafik für den Standardeintrag.
   In der Spalte "Wert" des Eintrags erscheint der Wert "Standardeintrag".
2. Alternativ aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Option "Standard".

Gesetztes Bit mit der kleinsten Wertigkeit

Wenn "Bitauswahl in Text- und Grafiklisten" aktiviert ist, wird die Grafik angezeigt, die beim gesetzten Bit mit der kleinsten Wertigkeit projektiert ist.

Wenn beim gesetzten Bit mit der kleinsten Wertigkeit keine Grafik projektiert ist und auch kein Standardwert, wird die Kakteengrafik angezeigt. Wenn ein Standardwert projektiert ist, wird die beim Standardwert projektierte Grafik angezeigt.

Um jeweils nur die Grafik beim gesetzten Bit mit der kleinsten Wertigkeit anzuzeigen, aktivieren Sie im Editor "Runtime-Einstellungen" die Option "Bitselektion in Text- und Grafiklisten".

Wegen der Abwärtskompatibilität ist die Einstellung standardmäßig deaktiviert. Die Einstellung gilt für alle Grafiklisten des Bediengeräts.

1.3.2.9 Objekt mit Grafikliste projektieren

Einleitung

Ausgabewert und Wertübernahme für Grafiklisten werden in dem Anzeige- und Bedienobjekt festgelegt, das die Grafiken der Grafikliste in Runtime anzeigt. Die Eigenschaften dieser Objekte werden nach Bedarf konfiguriert.

Voraussetzung

- Eine Grafikliste ist erstellt. Die Werte sind definiert. Den Werten sind Grafiken zugeordnet.
- Eine Variable ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Ein Bild mit einem Grafischen E/A-Feld ist geöffnet. Das Objekt ist selektiert.

Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Grafikliste" die Grafikliste aus, deren Grafiken Sie in Runtime anzeigen lassen wollen.
2. Wählen Sie für "Modus" die Einstellung "Eingabe/Ausgabe".

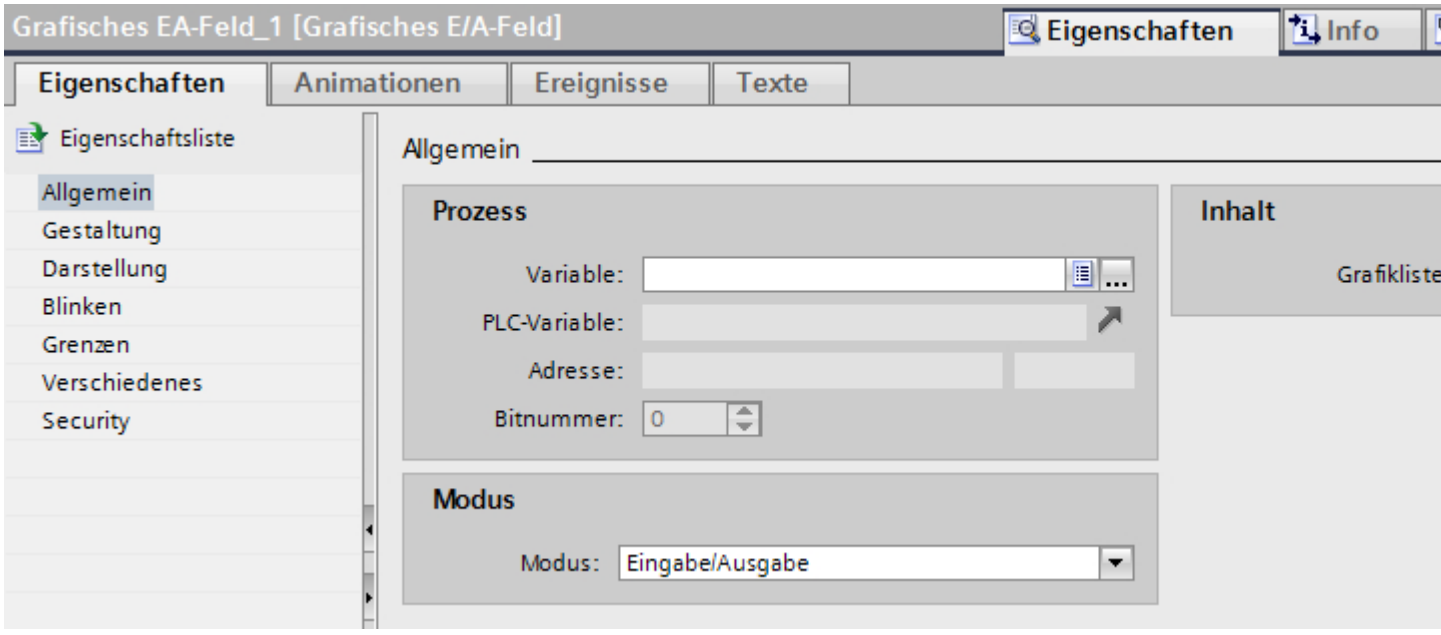

> **Hinweis**
> **Runtime-Abhängigkeit**
>
> Je nach Runtime stehen unterschiedliche Feldtypen für ein Grafisches E/A-Feld zur Verfügung.

3. Wählen Sie als "Variable" die Variable, deren Werte die Anzeige im Grafischen E/A-Feld bestimmt.

Ergebnis

In Runtime werden die definierten Grafiken der Grafikliste im Grafischen E/A-Feld angezeigt, wenn die Variable den angegebenen Wert annimmt.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

1.3.3 Arbeiten mit C-Textlisten

1.3.3.1 Grundlagen zur C-Textliste

Einleitung

Jedem Bediengerät ist eine C-Textliste zugeordnet. In dieser Liste erstellen Sie mehrsprachige Texte, die Sie über die Text-ID in einem C-Skript referenzieren. Sie nutzen die C-Textliste ausschließlich mit Funktionen der API-Schnittstelle.

Verwendung

Sie verwenden die C-Textliste z. B. für die Anzeige von Texten in einem Bildobjekt.

Mehrsprachige Texte

Sie können die Texte in einer C-Textliste mehrsprachig projektieren. Die Texte werden in Runtime in der eingestellten Runtime-Sprache angezeigt. Dazu stellen Sie die Sprachen im Projektfenster unter "Sprachunterstützung > Projektsprachen" ein.

Projektierungsschritte

Zur Anzeige von Texten, z. B. in einem Bildobjekt, sind folgende Schritte erforderlich:

1. Einträge in der C-Textliste erstellen
2. Bildobjekt erstellen
3. C-Skript anbinden, z. B. an ein Ereignis des Bildobjekts
4. Einträge im C-Skript referenzieren über die entsprechenden Funktionen

Siehe auch

Einträge in die C-Textliste erstellen (Seite 208)

1.3.3.2 Einträge in die C-Textliste erstellen

Einleitung

In der C-Textliste erstellen Sie mehrsprachige Texte und referenzieren diese über die Text-ID in einem C-Skript. Die Texte der C-Textliste nutzen Sie ausschließlich mit Funktionen der API-Schnittstelle.

Vorgehen

Um einen Eintrag in der C-Textliste zu erstellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Text- und Grafiklisten".
2. Öffnen Sie die Registerkarte "C-Textliste".

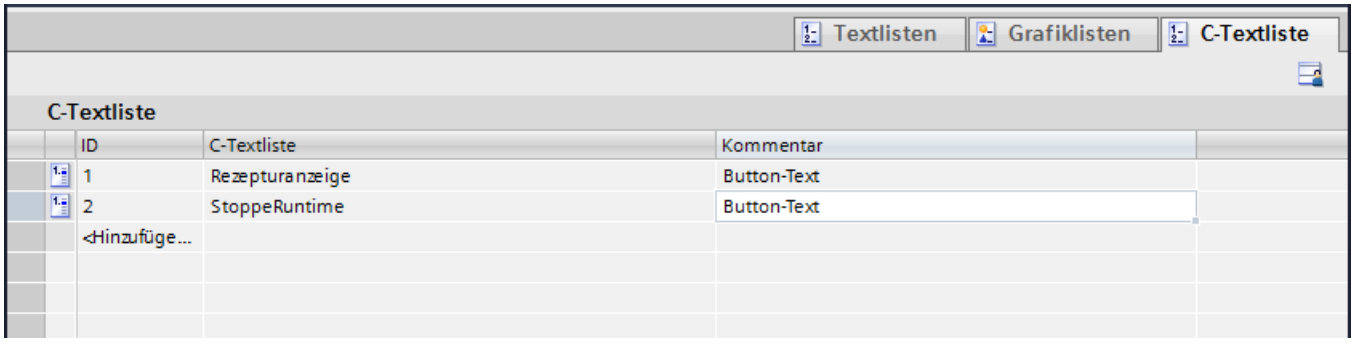

3. Klicken Sie in der Tabelle auf "Hinzufügen".
   Das Inspektorfenster zur C-Textliste wird geöffnet.
4. Tragen Sie den Text ein und bei Bedarf Ihren Kommentar dazu.
   Die Text-ID wird automatisch erstellt.
5. Um zusammengehörige Texte zu kennzeichnen, ändern Sie bei Bedarf die ID.
   Die Text-ID muss innerhalb des Nummernbereichs 1 bis 2147483647 liegen und eindeutig sein. Die Eindeutigkeit wird vom System geprüft.

   Hinweis

   Wenn Sie die Text-ID eines bereits referenzierten Eintrags ändern, wird diese Änderung von der Syntaxprüfung im Skript-Editor ignoriert.

Ergebnis

Ein Eintrag ist in der C-Textliste erstellt.

Siehe auch

Grundlagen zur C-Textliste (Seite 208)

1.4 Dynamisieren von Bildern

1.4.1 Grundlagen zur Dynamisierung von Bildern

1.4.1.1 Grundlagen zur Dynamisierung von Bildern

Dynamisierung von Objekten

In WinCC dynamisieren Sie Objekte, um Ihre Anlage auf Bediengeräten abzubilden und Prozessabläufe anzuzeigen.

Sie realisieren Dynamisierungen durch

- Animationen
- Variablen
- Systemfunktionen

Ein Beispiel ist die Abbildung eines Tanks, dessen Flüssigkeitspegel abhängig von einem Prozesswert steigt oder fällt.

Welche Möglichkeiten der Dynamisierung verfügbar sind, hängt vom Objekt ab. Wenn Sie ein Objekt kopieren, werden auch die Dynamisierungen mitkopiert.

Siehe auch

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Dynamisieren im Inspektorfenster (Seite 213)

Neue Animation projektieren (Seite 217)

Übersicht der Dynamisierungen einblenden (Seite 216)

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Grundlagen zu Bildern (Seite 33)

Grundlagen zu Ereignissen (Seite 246)

1.4.1.2 Grundlagen zur Dynamisierung von Bildern

Dynamisierung von Objekten

In WinCC dynamisieren Sie Objekte, um Ihre Anlage auf Bediengeräten abzubilden und Prozessabläufe anzuzeigen.

Sie realisieren Dynamisierungen durch

- Animationen
- Systemfunktionen

- Variablen
- Lokale Skripte

Ein Beispiel ist die Abbildung eines Tanks, dessen Flüssigkeitspegel abhängig von einem Prozesswert steigt oder fällt.

Welche Möglichkeiten der Dynamisierung verfügbar sind, hängt vom Objekt ab. Wenn Sie ein Objekt kopieren, werden auch die Dynamisierungen mitkopiert.

Benutzerdefinierte Menüs und Symbolleisten

Im Editor "Menüs und Symbolleisten" projektieren Sie benutzerdefinierte Menüs und Symbolleisten. In allen Bildern ihres Projekts oder in Bildfenstern werden diese Menüs und Symbolleisten zusätzlich angezeigt. Die Menüeinträge und Symbole verbinden Sie mit Variablen oder VB-Skripten.

Mit den benutzerspezifischen Menüs und Symbolleisten wechseln Sie z. B. von einem Bild ihres Projekts in jedes beliebige andere Bild.

Siehe auch

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Dynamisieren im Inspektorfenster (Seite 213)

Grundlagen zu Zyklen (Seite 619)

1.4.1.3 Dynamisieren im Inspektorfenster

Einleitung

Grundsätzlich können Sie alle Bildobjekte, die Sie in einem Bild projektiert haben, dynamisieren. Vom Gerät und dem selektierten Objekt hängt ab, welche Möglichkeiten der Dynamisierung und welche Ereignisse verfügbar sind.

Animationen

WinCC unterstützt Sie bei der Dynamisierung durch bereits vordefinierte Animationen. Wenn Sie ein Objekt animieren wollen, projektieren Sie zunächst die gewünschte Animation im Inspektorfenster des Objekts. Anschließend passen Sie die Animation an die Erfordernisse ihres Projekts an.

Die Auswahl der unterstützten Animationen ist abhängig vom Bediengerät und vom selektierten Objekt. Sie wählen zwischen folgenden Animationsarten:

- Darstellung: Gestaltung, Sichtbarkeit
- Bewegungen: direkte, diagonale, horizontale und vertikale Bewegung
- Variablenanbindung

Die Animationenart "Variablenanbindung" können Sie für ein Objekt mehrfach projektieren.

Animationen projektieren Sie im Inspektorfenster "Eigenschaften > Animationen".

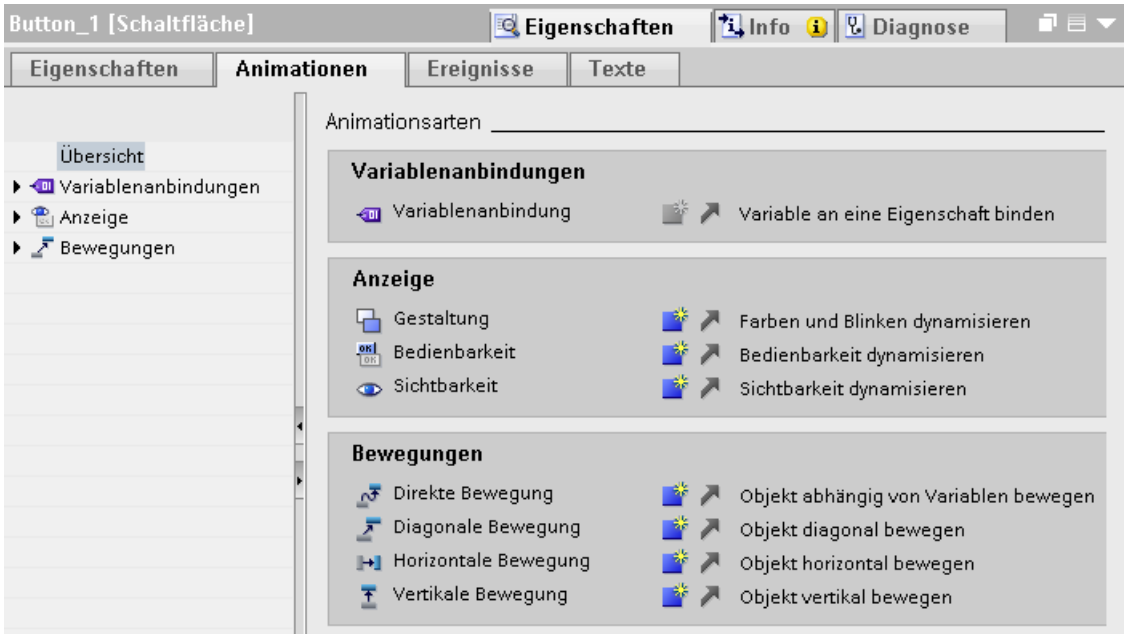

Ereignisse

Bedienbare Objekte reagieren zusätzlich auf Ereignisse, z. B. einen Mausklick.

An ein Ereignis projektieren Sie eine Funktionsliste mit Systemfunktionen. Als Reaktion auf das ausgelöste Ereignis werden die Systemfunktionen abgearbeitet.

Ereignisse projektieren Sie im Inspektorfenster "Eigenschaften > Ereignisse".

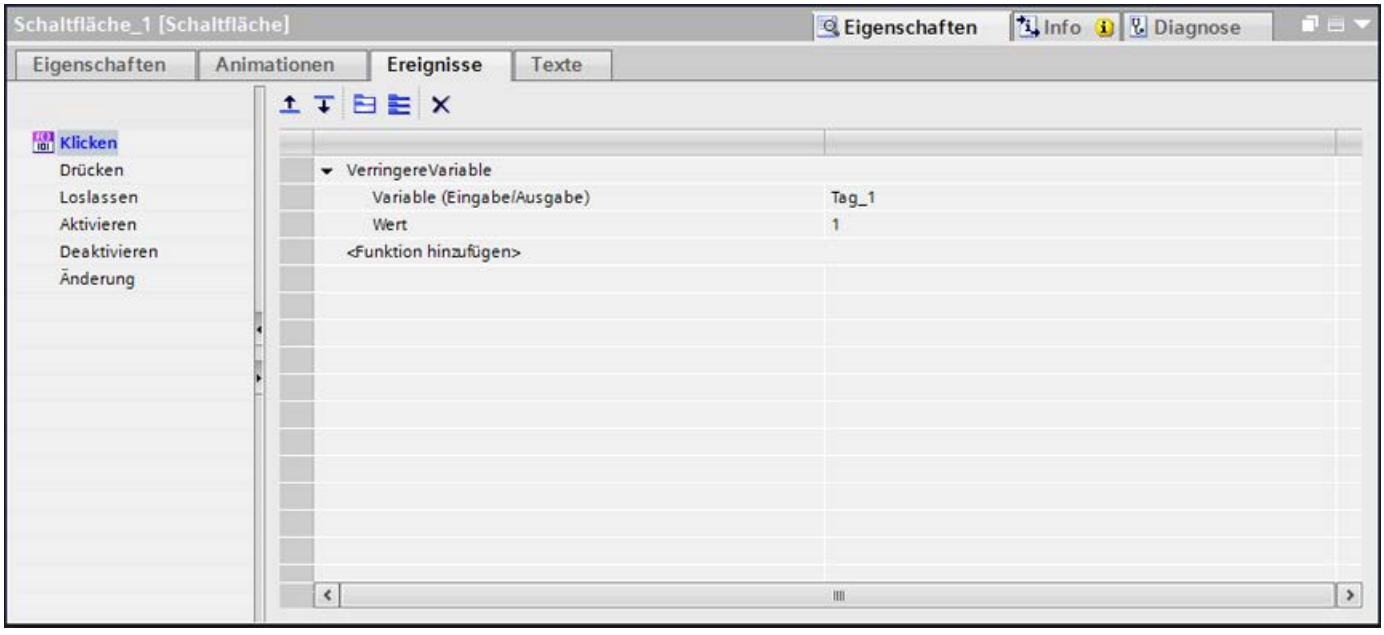

Nähere Hinweise hierzu finden Sie unter "Arbeiten mit Funktionslisten (Seite 1439)".

Siehe auch

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Arbeiten mit Funktionslisten (Seite 1439)

1.4.1.4 Dynamisieren im Inspektorfenster

Einleitung

Grundsätzlich können Sie alle Bildobjekte, die Sie in einem Bild projektiert haben, dynamisieren. Vom Gerät und dem selektierten Objekt hängt ab, welche Möglichkeiten der Dynamisierung und welche Ereignisse verfügbar sind.

Das Inspektorfenster bietet Ihnen in drei Registerkarten die Möglichkeiten zur Dynamisierung von Bildobjekten.

Eigenschaftsliste

Mit der Schaltfläche schalten Sie im Inspektorfenster "Eigenschaften > Eigenschaften" zur Eigenschaftsliste.

In der Eigenschaftsliste erstellen Sie für die einzelnen Objekteigenschaften beliebige Dynamisierungen mit Variablen oder lokalen C- und VB-Skripten. Sie verändern z. B. die Hintergrundfarbe eines Objekts dynamisch.

In der Spalte "Dynamischer Wert" können Sie die Objekteigenschaft von einer Variablen, einem C-Skript oder einem VB-Skript abhängig machen.

Ob eine Objekteigenschaft dynamisierbar ist, sehen Sie in der Eigenschaftsliste anhand der Hintergrundfarbe:

- Hellgrau: Dynamisierbar
- Dunkelgrau: Nicht dynamisierbar

Bereits projektierte Animationen werden ebenfalls in der Eigenschaftsliste des Objekts angezeigt, wo Sie die Animationen bearbeiten können.

Die folgende Tabelle zeigt, wie Sie die Eigenschaftsliste sortieren können:

	Alphabetische Sortierung
	Sortierung nach Kategorien

Mit der Schaltfläche schalten Sie zurück zu den Eigenschaftsseiten im Inspektorfenster.

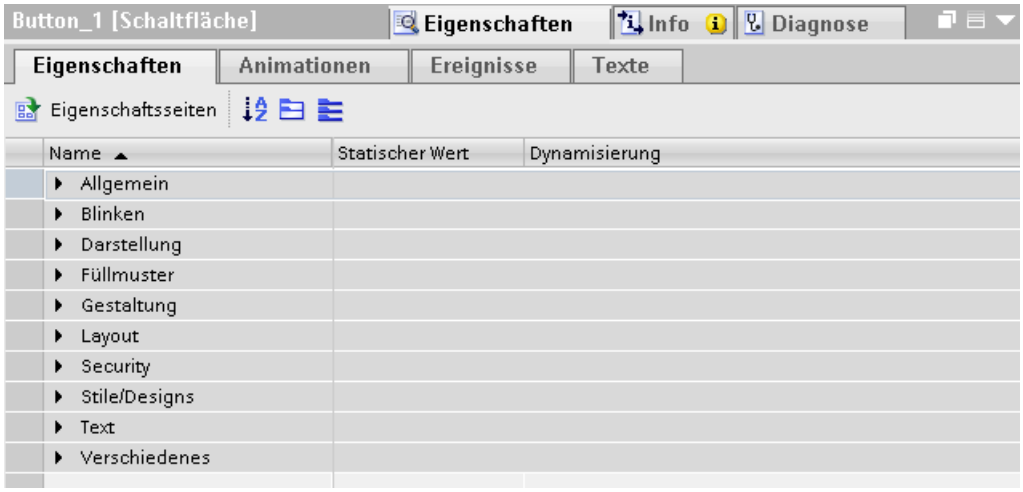

Animationen

Um Objekte zu animieren, stehen Ihnen zwei verschiedene Möglichkeiten zur Verfügung.

Sie projektieren Animationen

- im Inspektorfenster unter "Eigenschaften > Animationen",
- in der Task Card "Animationen".

WinCC unterstützt Sie bei der Dynamisierung durch bereits vordefinierte Animationen. Wenn Sie ein Objekt animieren wollen, projektieren Sie zunächst die gewünschte Animation mit den Werkzeugen des Werkzeugfensters oder im Inspektorfenster des Objekts. Anschließend passen Sie die Animation im Inspektorfenster an die Erfordernisse ihres Projekts an.

Die Auswahl der unterstützten Animationen ist abhängig vom Gerät und vom selektierten Objekt. Sie wählen zwischen folgenden Animationsarten:

- Variablenanbindung
- Darstellung: Gestaltung, Bedienbarkeit, Sichtbarkeit
- Bewegungen: direkte, diagonale, horizontale und vertikale Bewegung
- Eigenschaftsanimation

Hinweis

Bediengeräteabhängigkeit

Die Animation "Bedienbarkeit" steht nicht für alle Bediengeräte zur Verfügung.

Die Animationenarten "Variablenanbindung" und "Eigenschaftsanimation" können Sie für ein Objekt mehrfach projektieren.

Animationen projektieren Sie im Inspektorfenster "Eigenschaften > Animationen".

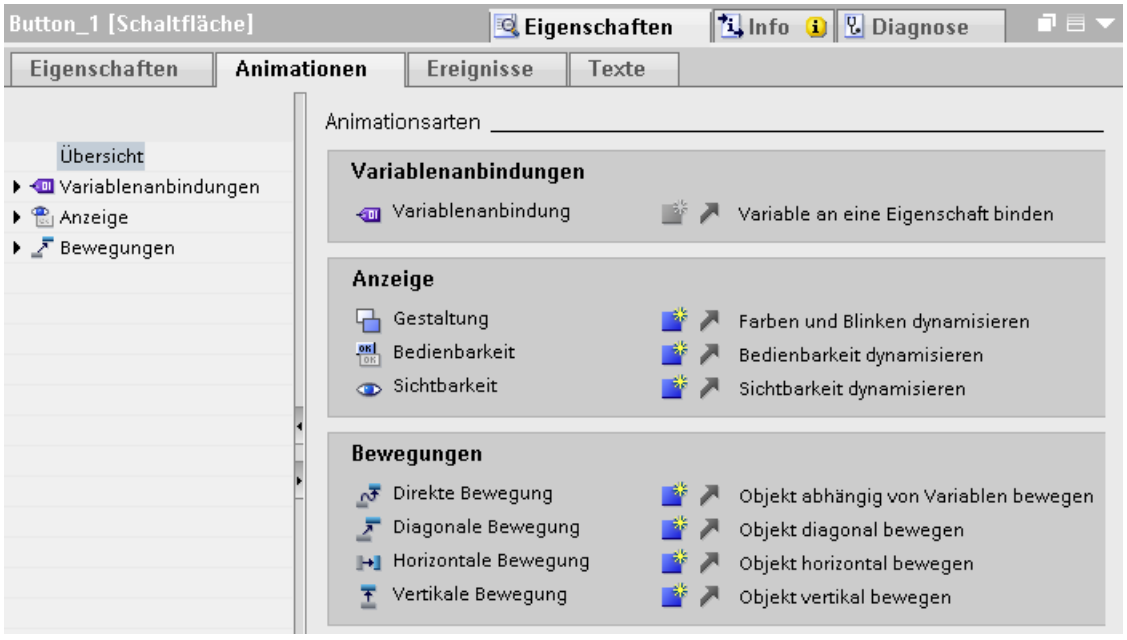

Ereignisse

Bedienbare Objekte reagieren zusätzlich auf Ereignisse, z. B. einen Mausklick.

An ein Ereignis projektieren Sie eine Funktionsliste mit Systemfunktionen oder lokale Skripte. Als Reaktion auf das ausgelöste Ereignis werden die Systemfunktionen bzw. das lokale Skript abgearbeitet.

Ereignisse projektieren Sie im Inspektorfenster "Eigenschaften > Ereignisse".

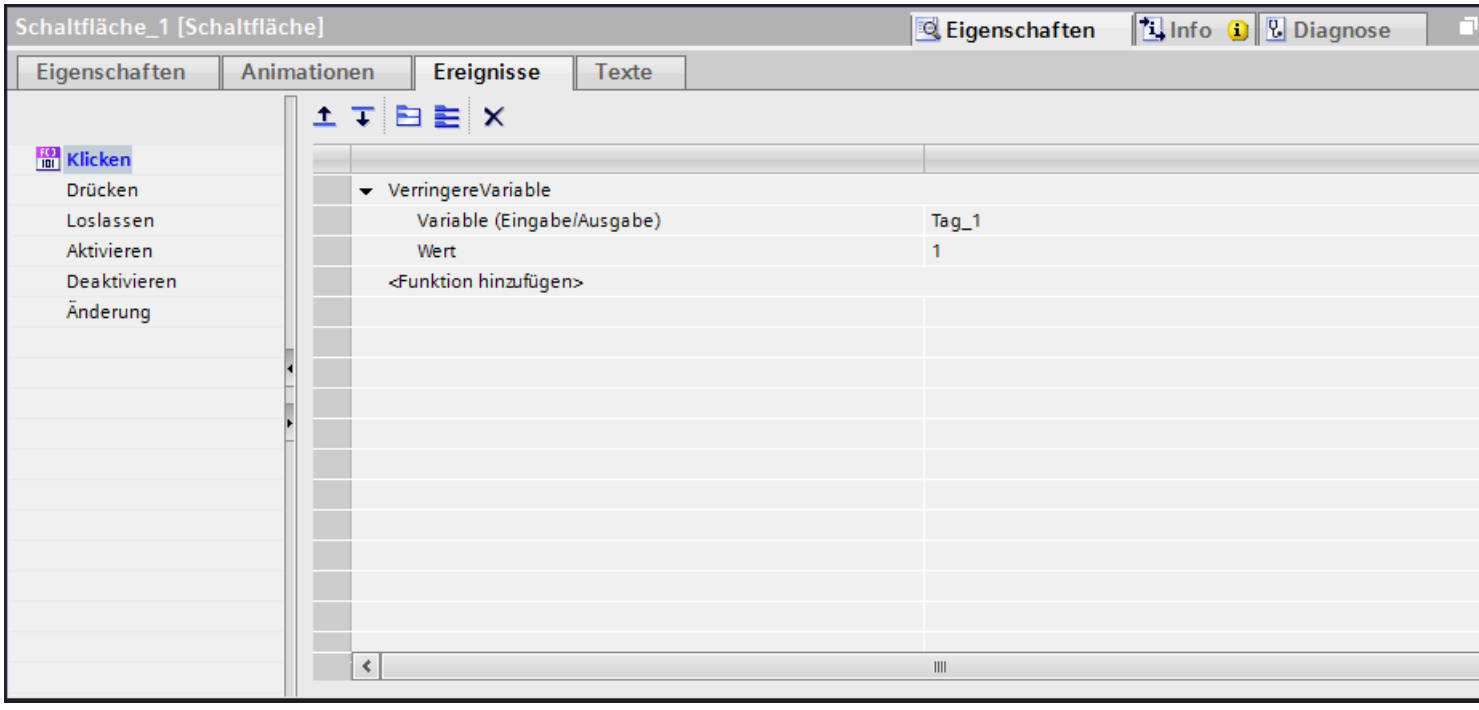

Nähere Hinweise hierzu finden Sie unter "Arbeiten mit Funktionslisten (Seite 1439)".

Siehe auch

Lokale Skripte (Seite 1437)

Arbeiten mit Funktionslisten (Seite 1439)

1.4.1.5 Übersicht der Dynamisierungen einblenden

Einleitung

Sie können für jede einzelne Dynamisierung den Aktualisierungszyklus getrennt einstellen. Die Anzahl unterschiedlicher Aktualisierungszyklen ist pro Projekt auf fünf beschränkt.

Übersicht über vorhanden Dynamisierungen einblenden

1. Selektieren Sie in der Projektnavigation ein Bild.
2. Wählen Sie im Kontextmenü "Übersicht Dynamisierungen".

Ergebnis

Im Arbeitsbereich werden alle Objekte und deren Dynamisierungen im Bild anzeigt einschließlich. Unter "Zyklus" wählen Sie bei Bedarf einen anderen Aktualisierungszyklus.

Siehe auch

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

1.4.2 Dynamisieren mit Animationen

1.4.2.1 Neue Animation projektieren

Einleitung

Um Bildobjekte zu dynamisieren, verwenden Sie vordefinierte Animationen.

Voraussetzung

- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

Vorgehen im Inspektorfenster

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
2. Wählen Sie die gewünschte Animation aus.
3. Klicken Sie auf die Schaltfläche .

Vorgehen in der Task Card "Animationen"

1. Öffnen Sie in der Task Card "Animationen" die Objektgruppe, welche die gewünschte Animation enthält.
2. Ziehen Sie die gewünschte Animation auf das Objekt, das Sie dynamisieren wollen.

Alternativ selektieren Sie das Objekt im Bild und doppelklicken in der Task Card "Animation" auf die gewünschte Animation.

Ergebnis

Die Animation wird im Inspektorfenster des Objekts eingetragen. In den folgenden Schritten konfigurieren Sie die Animation.

In der Übersicht der Animationen symbolisiert Ihnen der grünen Pfeil, welche Animation bereits projektiert ist. Wenn Sie auf den grünen Pfeil klicken, dann öffnet sich die projektierte Animation im Inspektorfenster.

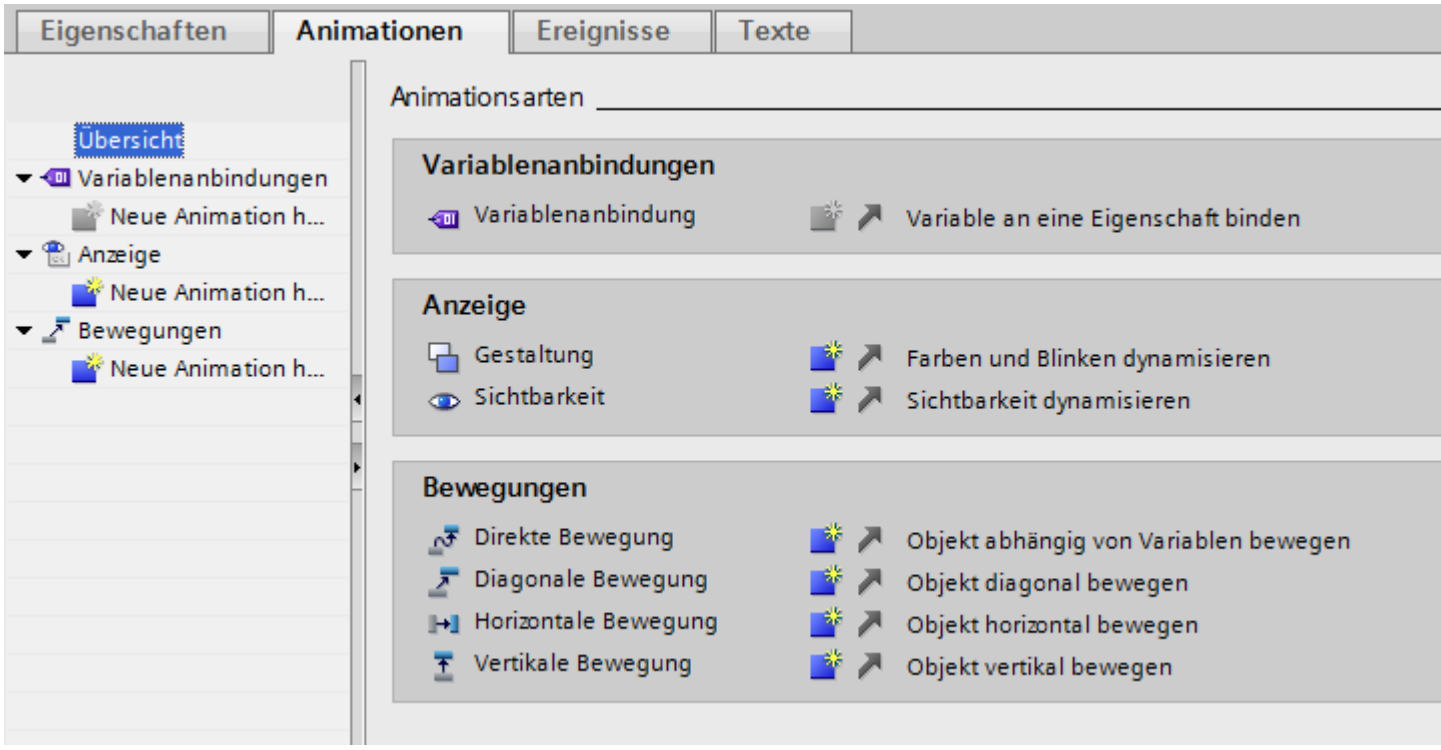

Siehe auch

Gestaltung eines Objekts dynamisieren (Seite 219)
Bewegung projektieren (Seite 221)
Direkte Bewegung projektieren (Seite 223)
Bedienbarkeit eines Bedienobjekts dynamisieren (Seite 224)
Sichtbarkeit eines Objekts dynamisieren (Seite 226)
Animation mit Variablenanbindung projektieren (Seite 227)
Animationen bei Mehrfachauswahl (Seite 231)
Grundlagen zur Dynamisierung von Bildern (Seite 210)
Animationen von Objektgruppen (Seite 230)
Dynamisieren mit Eigenschaftsanimationen (Seite 232)

1.4.2.2 Gestaltung eines Objekts dynamisieren

Einleitung

In Runtime wird das Erscheinungsbild eines Bildobjekts durch die Wertänderung einer Variablen gesteuert. Wenn die Variable einen bestimmten Wert annimmt, ändert das Bildobjekt entsprechend der Projektierung seine Farbe oder sein Blinkverhalten.

Typ

Je nach Auswahl werden Bereiche oder Einzelwerte der Variablen in Runtime betrachtet. Das Erscheinungsbild des Objekts ändert sich entsprechend der Projektierung.

Hinweis

Auswahl "Mehrere Bits"

Wenn Sie die Option "Mehrere Bits" verwenden, aktivieren Sie in den Runtime-Einstellungen unter "Bilder > Allgemein" zusätzlich die Option "Bitselektion für die Gestaltung".

Voraussetzung

- Ein Bild ist geöffnet.
- Im Bild ist ein dynamisierbares Objekt enthalten und markiert.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
2. Wählen Sie die Animation "Gestaltung" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
3. Wählen Sie "Variable > Name" eine Variable aus.
4. Wählen Sie z. B. "Typ > Bereich".
5. Klicken Sie in der Tabelle auf "Hinzufügen".
6. Geben Sie in die Spalte "Bereich" z. B. das Variablenintervall "0 - 20" ein.
7. Wählen Sie für "Vordergrundfarbe" und "Hintergrundfarbe" die Farbe aus, die das Objekt annimmt, sobald in Runtime die Variable das Intervall erreicht.

8. Wählen Sie aus der Liste "Blinken" ein Blinkverhalten für das Objekt.
9. Um ein weiteres Variablenintervall z. B."21 - 60" anzulegen, wiederholen Sie die Schritte 5-8.

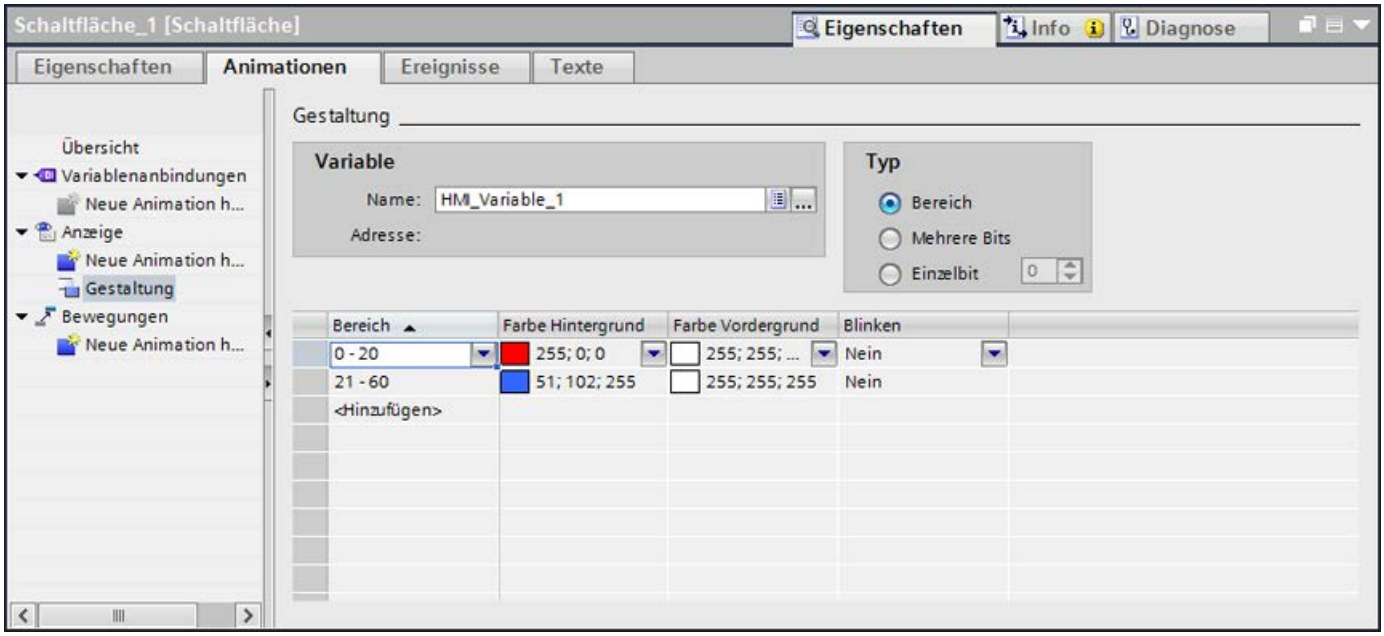

Ergebnis

Abhängig vom Prozesswert der Variablen ändern sich in Runtime das Blinkverhalten und die Farbe des Objekts.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.3 Bewegung projektieren

Einleitung

Dynamische Objekte können Sie so projektieren, dass sie sich auf einer bestimmten Bahn bewegen. Die Bewegung wird über Variablen gesteuert. Bei jeder Aktualisierung der Variablen wird das Objekt bewegt.

Für jedes Objekt können Sie nur eine einzige Bewegungsart projektieren.

Voraussetzung

- Eine Variable ist angelegt.
- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.

- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

Vorgehen

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Horizontale Bewegung" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
   Im Arbeitsbereich wird eine transparente Kopie des Objekts eingeblendet, die durch einen Pfeil mit dem Originalobjekt verbunden ist.
4. Wählen Sie eine Variable zur Bewegungssteuerung aus.
5. Schieben Sie die Objektkopie an die gewünschte Zielposition. Im Inspektorfenster werden die Pixelwerte der Endposition automatisch eingetragen.
6. Passen Sie bei Bedarf den Wertebereich der Variablen an.

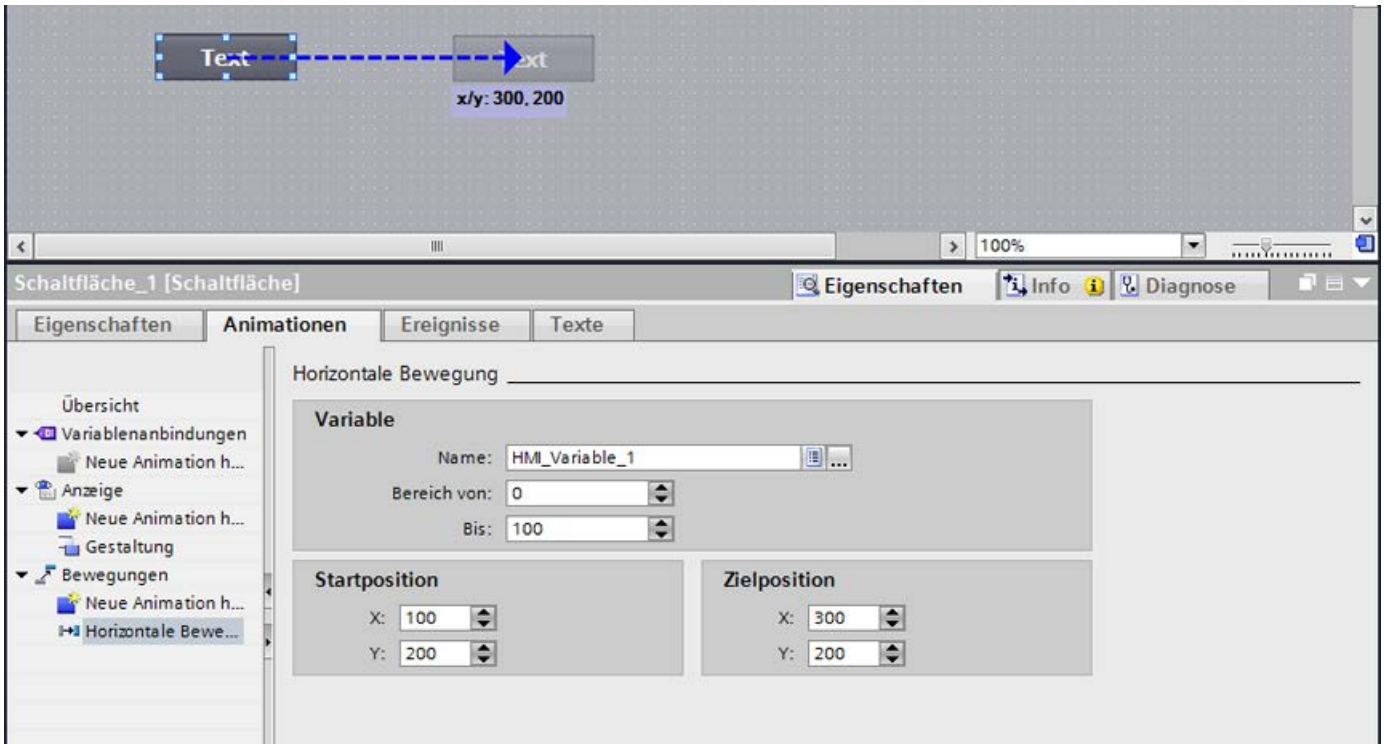

Ergebnis

In Runtime bewegt sich das Objekt bei jeder Wertänderung der Variablen, die zur Steuerung der Bewegung dient. Die Bewegungsrichtung entspricht der projektierten Bewegungsart "horizontal".

Hinweis

Vertikale und diagonale Bewegung projektieren Sie auf die gleiche Art wie die horizontale Bewegung.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.4 Direkte Bewegung projektieren

Einleitung

Bei der "Direkten Bewegung" wird das Objekt jeweils in x-Richtung und in y-Richtung verschoben. Zwei Variablen legen die Anzahl der Pixel fest, um die sich das Objekt von seiner ursprünglichen statischen Anfangsposition bewegt.

Voraussetzung

- Zwei Variable sind angelegt.
- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

"Direkte Bewegung" projektieren

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
3. Wählen Sie "Direkte Bewegung" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.

4. Wählen Sie für die X-Position eine Variable, mit welcher die Bewegung in x-Richtung gesteuert wird.
5. Wählen Sie für die Y-Position eine Variable, mit welcher die Bewegung in y-Richtung gesteuert wird.

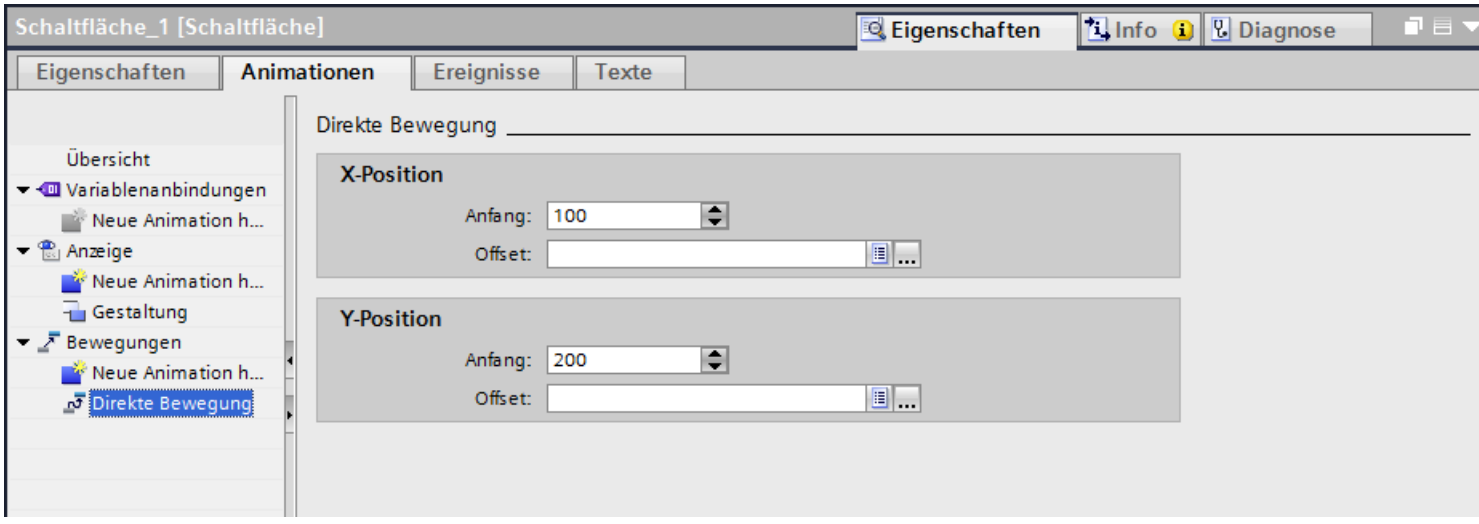

Ergebnis

In Runtime bewegt sich das Objekt bei jeder Wertänderung der Variablen, die zur Steuerung der Bewegung dient.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.5 Bedienbarkeit eines Bedienobjekts dynamisieren

Einleitung

Die Bedienbarkeit eines Bedienobjekts können Sie dynamisieren. Abhängig vom Wert einer Variablen kann der Bediener das Objekt in Runtime bedienen oder nicht.

Diese Dynamisierung verwenden Sie, damit der Bediener ein Bedienobjekt nur in bestimmten Situationen, z. B. während der Wartung, bedienen kann.

Voraussetzung

- Eine Variable ist angelegt.
- Ein Bild mit einem Bedienobjekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

Vorgehen

1. Markieren Sie im Bild das Bedienobjekt, das Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Bedienbarkeit" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.

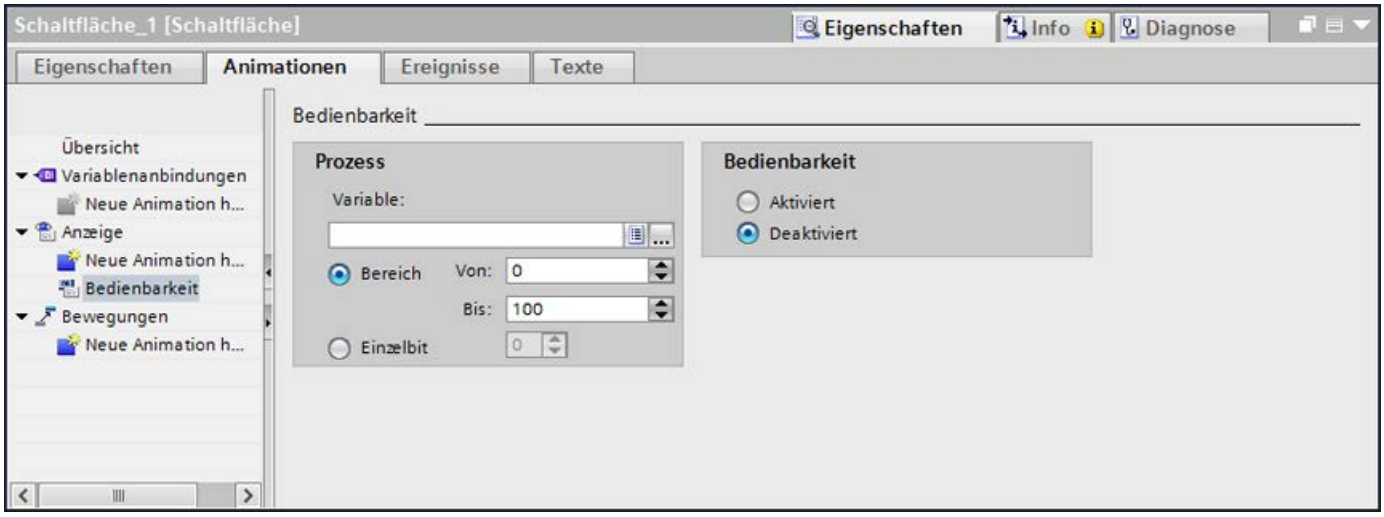

4. Wählen Sie eine Variable.
5. Aktivieren Sie "Bereich".
   - Geben Sie für "Von:" z. B. den Wert "0"
   - Geben Sie für "Bis:" z. B. den Wert "100".
6. Aktivieren Sie "Bedienbarkeit > Aktiviert".

Ergebnis

Das Bildobjekt ist in Runtime abhängig vom Variablenwert bedienbar:

- Wenn der Variablenwert zwischen 0 und 100 liegt, wird das Objekt bedienbar.
- Wenn der Variablenwert außerhalb des Bereichs liegt, ist das Objekt nicht bedienbar.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.6 Sichtbarkeit eines Objekts dynamisieren

Einleitung

Mit dem Dynamisieren der Eigenschaft "Sichtbarkeit" können Sie z. B. eine Warnung im Bild einblenden, wenn eine Variable einen kritischen Wert überschreitet. Wenn der Wert der Variablen wieder im unkritischen Bereich liegt, wird die Warnung ausgeblendet.

Die Objekte "Einfache Rezepturanzeige" und "Einfache Meldeanzeige" sind immer sichtbar.

Voraussetzung

- Eine Variable ist angelegt.
- Ein Bild mit einem Objekt, das Sie in Runtime einblenden oder ausblenden wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Sichtbarkeit" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
4. Wählen Sie eine Variable.
5. Aktivieren Sie "Bereich".
6. Wählen Sie z.B. "Von 20 und "Bis 40".
7. Aktivieren Sie "Sichtbar".

Ergebnis

Das Bildobjekt wird in Runtime abhängig vom Variablenwert eingeblendet oder ausgeblendet:

- Wenn der Variablenwert mit dem projektierten Bereich zwischen 20 und 40 übereinstimmt, wird das Bildobjekt eingeblendet.
- Wenn der Variablenwert ausserhalb des projektierten Bereichs liegt, wird das Bildobjekt ausgeblendet.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.7 Animation mit Variablenanbindung projektieren

Einleitung

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, wird in Runtime die Objekteigenschaft abhängig vom Variablenwert geändert.

Das folgende Beispiel zeigt die Variablenanbindung im Inspektorfenster "Eigenschaften > Animationen". Wenn Sie die Variablenbindung projektiert haben, ist sie im Inspektorfenster auch in der Eigenschaftsliste bzw. in den Eigenschaftsseiten sichtbar.

Voraussetzung

- Ein Objekt mit der zu dynamisierenden Eigenschaft ist selektiert.
- Eine Variable ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Markieren Sie im Bild das Objekt, dessen Eigenschaften Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animation".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Variablenanbindung" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.

4. Wählen Sie "Eigenschaften > Animation > Variablenanbindung > Eigenschaft > Name" die Objekteigenschaft aus, die Sie dynamisieren wollen, z.B. "Farbe Hintergrund".
5. Wählen Sie eine Variable.

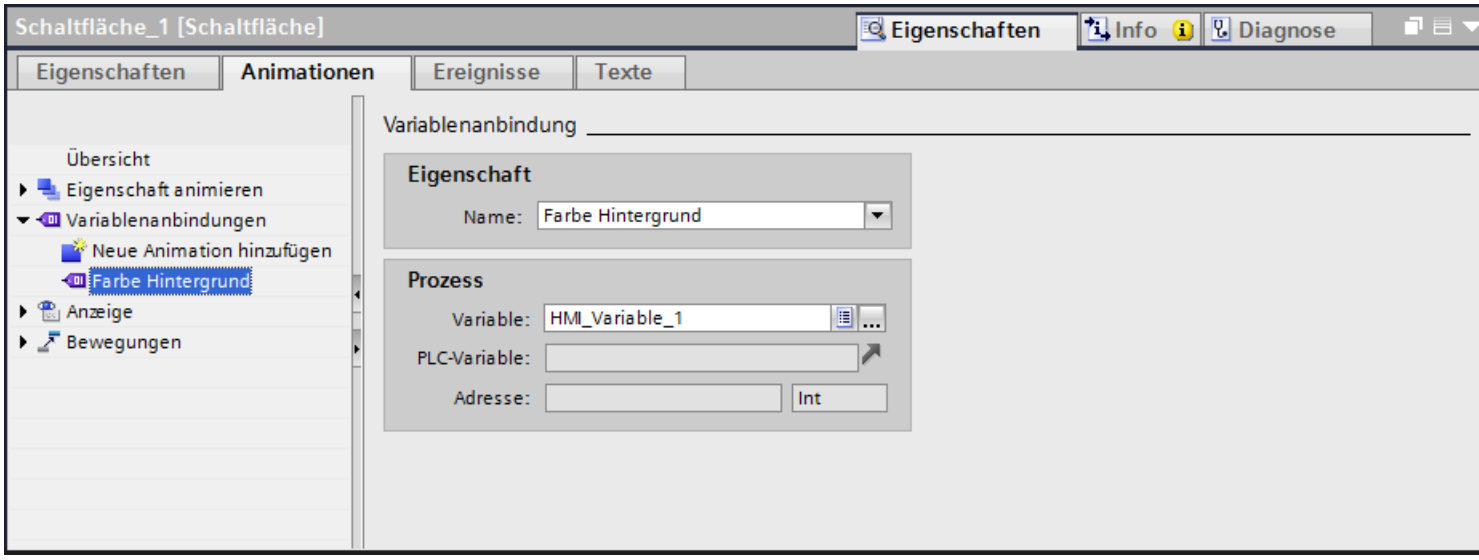

Ergebnis

Sie haben die Eigenschaft "Farbe Hintergrund" mit einer Variablen dynamisiert. Je nachdem welchen Wert die Variable annimmt, ändert sich die Hintergrundfarbe des Bildobjekts in Runtime erforderlich.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.8 Animationen von Objektgruppen

Animationen auf Objektgruppen anwenden

Das Inspektorfenster zeigt alle Objekte einer Gruppe und ihre möglichen Animationen. Zusätzlich werden die Animationsarten separat aufgeführt, die von allen Objekten der Gruppe unterstützt werden.

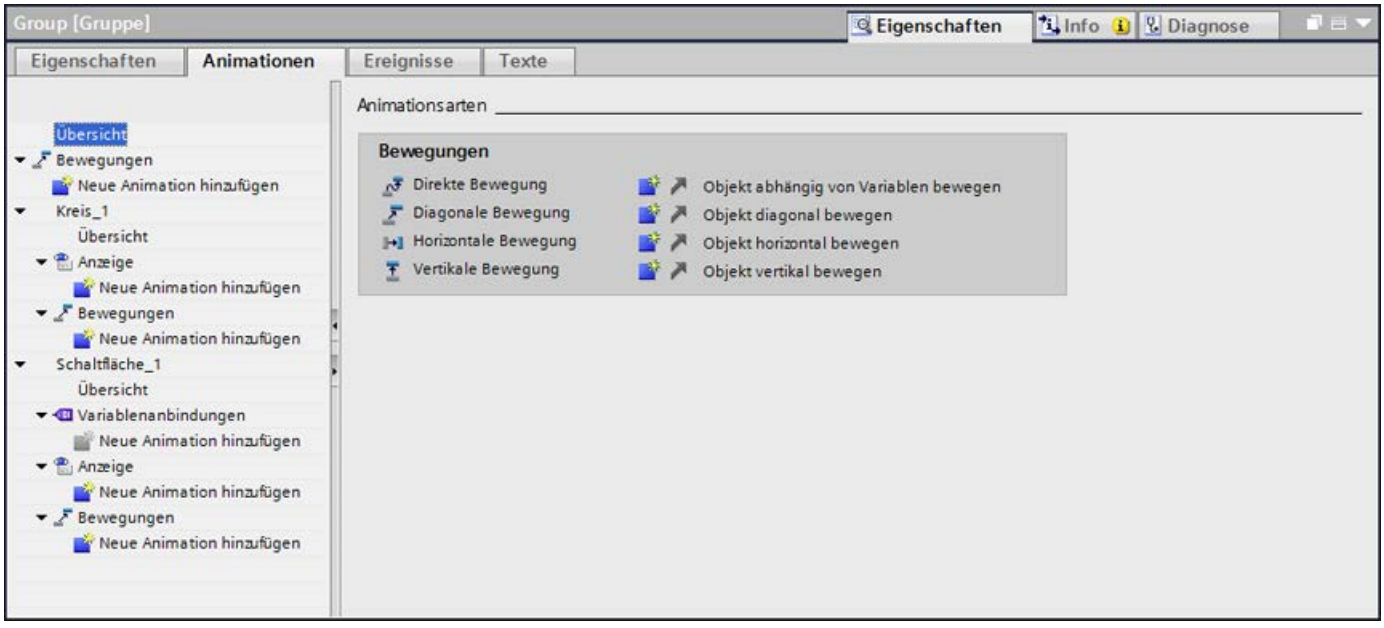

Wenn Sie eine Animation für eine Objektgruppe projektieren, gilt diese Animation für alle Einzelobjekte, die diese Animation unterstützen.

Die Animation von hierarchischen Gruppen wird nicht unterstützt.

Anwendungsbeispiel

Für das Objekt einer Objektgruppe ist die Animation "horizontale Bewegung" projektiert. Für die gesamte Objektgruppe ist die Animation "direkte Bewegung" projektiert. In Runtime wird nur die Animation der Objektgruppe d. h. "direkte Bewegung" ausgeführt. Dies gilt auch für Objektgruppen innerhalb von Objektgruppen. Es wird nur die Animation der Gruppe der obersten Ebene aufgeführt.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.9 Animationen von Objektgruppen

Animationen von Objektgruppen ändern

Das Inspektorfenster zeigt alle Objekte einer Gruppe und ihre möglichen Animationen. Zusätzlich werden die Animationsarten separat aufgeführt, die von allen Objekten der Gruppe unterstützt werden.

Wenn Sie eine Animation für eine Objektgruppe projektieren, gilt diese Animation für alle Einzelobjekte, die diese Animation unterstützen.

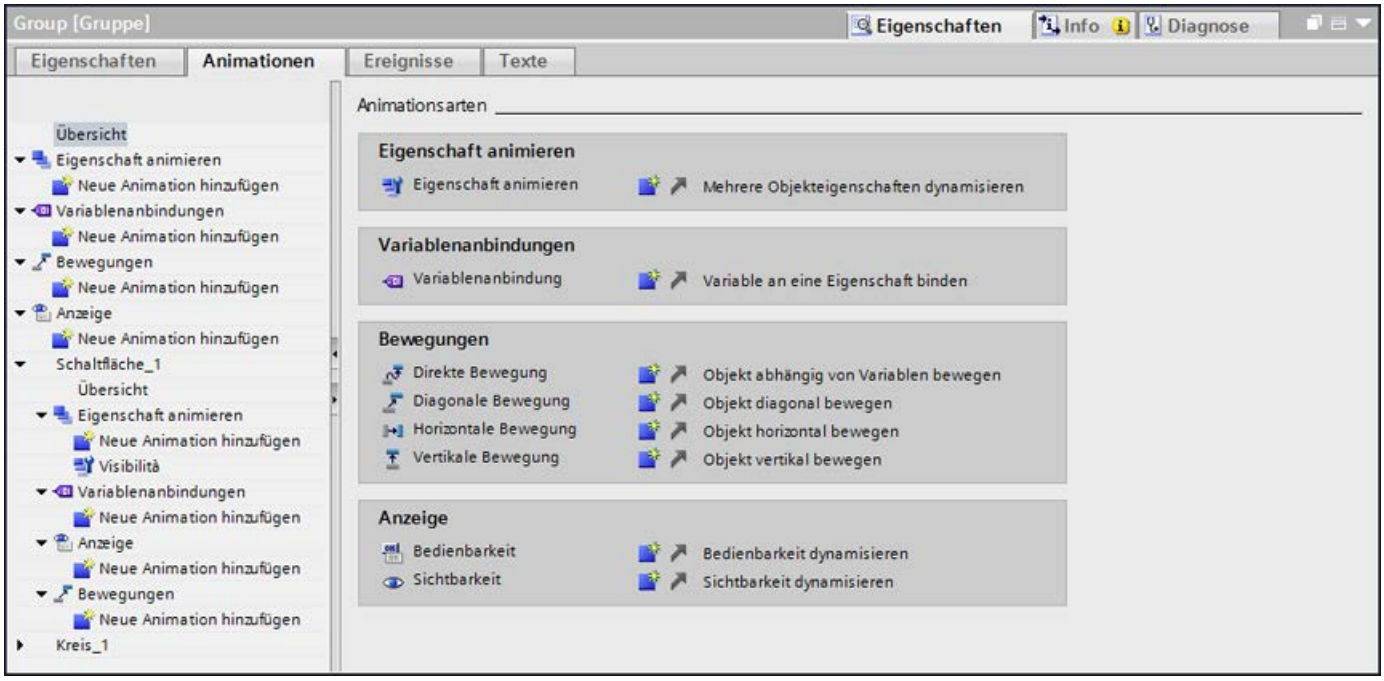

Anwendungsbeispiel

Für das Objekt einer Objektgruppe ist die Animation "horizontale Bewegung" projektiert. Für die gesamte Objektgruppe ist die Animation "direkte Bewegung" projektiert. In Runtime werden beide Animationsarten ausgeführt. Dies gilt auch für Objektgruppen innerhalb von Objektgruppen.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.2.10 Animationen bei Mehrfachauswahl

Animationen von mehreren Objekten ändern

Bei einer Mehrfachauswahl werden im Inspektorfenster die Animationen angezeigt, die für das Referenzobjekt projektiert sind. Die Animationen können Sie wie gewohnt ändern. Die Änderungen gelten für alle Objekte der Mehrfachauswahl, welche die projektierte Animation unterstützen. Somit erhalten auch Objekte die Animation des Referenzobjekts, für die Sie bisher keine Animation projektiert haben.

Anwendungsbeispiel

Sie markieren gleichzeitig eine Schaltfläche und einen Kreis. Die Schaltfläche ist das Referenzobjekt. Für die Schaltfläche ist bereits die Animation "Gestaltung" projektiert und wird daher im Inspektorfenster der Mehrfachauswahl angezeigt. Wenn Sie in im Inspektorfenster "Eigenschaften > Animationen >Gestaltung > Blinken" aktivieren, gelten danach die Einstellungen der Animation "Gestaltung" für die Schaltfläche und für den Kreis.

Neue Animationen für mehrere Objekte projektieren

Wenn Sie für die Objekte einer Mehrfachauswahl eine neue Animation projektieren, gilt diese Animation für alle markierten Objekte, welche die projektierte Animation unterstützen. Wenn die neue Animation eine bereits bestehende Animation ersetzen würde, wird eine Sicherheitsabfrage angezeigt.

Anwendungsbeispiel

Sie markieren einen Kreis und ein Rechteck. Für den Kreis ist bereits die Animation "diagonale Bewegung" projektiert. Sie projektieren für die Mehrfachauswahl die Animation "horizontale Bewegung". Diese Animation gilt für das Rechteck, da dafür noch keine Animation des Typs Bewegung projektiert ist. Für den Kreis werden Sie gefragt, ob Sie die bestehende Animation "diagonale Bewegung" durch die neue Animation "horizontale Bewegung" ersetzen wollen.

Siehe auch

Neue Animation projektieren (Seite 217)

1.4.3 Dynamisierung mit Eigenschaftsanimation

1.4.3.1 Dynamisieren mit Eigenschaftsanimationen

Einleitung

Mit der Eigenschaftsanimation dynamisieren Sie Eigenschaften von Bildern und Bildobjekten in Abhängigkeit von bestimmten Wertebereichen oder Variablen. Sie wählen Wertebereiche und bestimmen, wie sich die verknüpfte Objekteigenschaft ändert, wenn die Variable einen bestimmten Wert in Runtime annimmt. Sie haben die Möglichkeit mehrere Objekteigenschaften gleichzeitig zu dynamisieren, z. B. die Position und die Farbe eines Bildobjekts. Sie können auch mehrere Eigenschaftsanimationen an einem Objekt projektieren.

Hinweis

Mehrere Objekteigenschaften gleichzeitig dynamisieren

Sie dynamisieren mehrere Objekteigenschaften mit einer Animation. Die Abarbeitungsreihenfolge in Runtime entspricht nicht zwingend der Reihenfolge bei der Projektierung der Eigenschaften im Inspektorfenster unter "Eigenschaften > Animation".

Hinweis

Wenn Sie im Ausdruck die Operanden "=", ">", "<", "!=", verwenden, achten Sie auf Folgendes:
Der Wert für "wahr" = -1
Der Wert für "falsch" = 0

Bitte beachten Sie, dass der Runtime Wert einer Boolschen Variablen "wahr" = 1 und "falsch" = 0 ist.

Typ

Für die Festlegung des Wertebereichs stehen folgende Typen zur Verfügung:

- Bereich (Integer)
  Betrachtet werden ein oder mehrere Wertebereiche einer Variablen. Sie projektieren verschiedene Wertebereiche. Für jeden Wertebereich legen Sie die entsprechenden Werte der Objekteigenschaft fest. Wenn der Wert in Runtime innerhalb eines projektierten Wertebereichs liegt, wird die verknüpfte Objekteigenschaft dynamisiert.
  Legen Sie Wertebereiche nur ganzzahlig fest. Ein nicht ganzzahliger Variablenwert oder Ausdruck wird entsprechend auf- oder abgerundet.
- Bereich (Gleitkomma)
  Sie projektieren verschiedene Wertebereiche mit Gleitkommazahlen. Jeden Wertebereich geben Sie in eckigen Klammern ein und legen fest, ob die Bereichgrenze noch zu diesem Wertebereich gehört. Wenn der Wert in Runtime innerhalb eines projektierten Wertebereichs liegt, wird die verknüpfte Objekteigenschaft dynamisiert.

- Multiple Bits
  Betrachtet werden eine oder mehrere Bitnummern einer Variablen. Sie projektieren verschiedene Bitnummern. Nur wenn der Variablenwert in Runtime genau mit einer der projektierten Bitnummern übereinstimmt, wird die verknüpfte Objekteigenschaft dynamisiert. Wenn der Variablenwert nicht übereinstimmt, wird der statische Wert der Objekteigenschaft angezeigt.
- Single Bit
  Sie wählen das Bit der Variablen, das überwacht werden soll. Wenn das Bit gesetzt sein soll, projektieren Sie den Wert "0". Wenn das Bit nicht gesetzt sein soll, projektieren Sie den Wert "1".
- Bool (wahr, falsch)
  Betrachtet wird eine Boolsche Variable oder ein Ausdruck, der einen logischen Vergleich beinhaltet. Wenn der Variablen- oder Rückgabewert des Ausdrucks einen der beiden Werte - wahr oder falsch - zurückgibt, wird der Wert der dynamisierten Eigenschaft zugewiesen,
- Direkt
  Im Unterschied zu den restlichen Typen wird beim Typ "Direkt" der Variablenwert oder Ausdruck direkt in die Objekteigenschaft geschrieben.

Ausdruck

Die Option "Ausdruck" steht nur für die Typen "Bereich", "Boolen" und "Direkt" zur Verfügung.

Sie können einen Ausdruck formulieren, indem Sie Variablen, VB-Skripte und arithmetische Operatoren verwenden. Der Wert des Ausdrucks wird in Runtime ermittelt und mit den projektierten Wertebereichen verglichen. Innerhalb eines Ausdrucks steht Ihnen folgende Auswahl an VBS-Schlüsselworten und Konstanten zur Verfügung:

```
"mod", "not", "and" "or","xor","eqv", "imp", "vbTrue", "vbFalse"
```

Auswertung "Quality Code"

Um den Status und die Qualität einer Variablen zu prüfen, verwenden Sie den "Quality Code". Im angezeigten "Quality Code" ist die Qualität der gesamten Wertübertragung und Wertverarbeitung der jeweiligen Variablen zusammengefasst. Die Überwachung des "Quality Code" ermöglicht z. B. Rückschlüsse auf die Qualität der Variablen im Prozess.

Die Auswertung des "Quality Code" hat Vorrang vor den definierten Wertebereichen.

Wenn Sie mehrere Variablen verwenden, dann wird der "Quality Code" der Variablen in der Reihenfolge ausgewertet, wie sie im Ausdruck von links nach rechts vorkommen. Nur wenn der "Quality Code" der Variablen korrekt ist, werden die projektierten Werte betrachtet.

Eine Auswahl der angebotenen "Quality Codes" zeigt das folgende Bild. Die "Quality Codes" sind in dieser Aufstellung nach fallenden Prioritäten geordnet.

Value ▲
Auswertung
bad (0x1C) out of service
bad (0x18) no communication (no usable value)
bad (ox14) no communication (last usable value)
bad (0x0C) device failure
bad (0x08) not connected
bad (0x04) configuration error, value not accepted
bad (0x00) non-specific
bad (0x28) process related, substitute value
bad (0x2B) process related, no maintenance
bad miscellaneous states
uncertain(0x68) maintenance demanded
uncertain (0x60) simulated value

Siehe auch

Neue Animation projektieren (Seite 217)

Eigenschaftsanimation vom Typ Bereich projektieren (Seite 234)

Eigenschaftsanimation vom Typ Direkt projektieren (Seite 236)

Eigenschaftsanimation vom Typ Bool projektieren (Seite 237)

1.4.3.2 Eigenschaftsanimation vom Typ Bereich projektieren

Einleitung

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, wird in Runtime die Objekteigenschaft abhängig vom Variablenwert geändert. Das folgende Beispiel zeigt die Dynamisierung eines Kreises. Der Kreis ändern in Abhängigkeit von einer Variablen seine Farbe und seine Position.

Voraussetzung

- Eine Variable "Tag1" ist angelegt.
- Ein Bild ist angelegt.
- Ein Kreis ist im Bild eingefügt und selektiert.
- Das Inspektorfenster ist geöffnet.

Animationsart, Variable und Typ wählen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animation".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
2. Wählen Sie "Eigenschaftsanimation" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
3. Geben Sie unter "Einstellungen > Name" den Namen "Ampel" ein.

4. Wählen Sie als Typ "Bereich".
5. Aktivieren Sie "Prozess > Variable".
6. Wählen Sie "Prozess > Variable" die Variable "Tag1".

Objekteigenschaften und Bereiche wählen

1. Klicken Sie auf "Eigenschaft hinzufügen". Eine Auswahlliste wird geöffnet.
2. Wählen Sie die Eigenschaft "Position Y". Die Tabelle wird um eine Spalte "Position Y" erweitert.
3. Klicken Sie auf "Eigenschaft hinzufügen". Eine Auswahlliste wird geöffnet.
4. Wählen Sie die Eigenschaft "Farbe Hintergrund". Die Tabelle wird um eine Spalte "Farbe Hintergrund" erweitert.
5. Klicken Sie in der Tabelle auf "Hinzufügen".
6. Legen Sie den ersten Bereich von "0 - 49" fest.
7. Klicken Sie in der Tabelle auf "Hinzufügen".
8. Legen Sie den zweiten Bereich von "50 - 99" fest.
9. Klicken Sie in der Tabelle auf "Hinzufügen".
10. Legen Sie den zweiten Bereich von "100 - 149" fest.

Hintergrundfarbe und Position einem Bereich zuordnen

Legen Sie folgende Einstellungen für die Bereiche fest:

- Bereich 0 - 49: Position Y "200"; Farbe "Grün"
- Bereich 50 - 99: Position Y "220"; Farbe "Gelb"
- Bereich 100 - 149: Position Y "240"; Farbe "Rot"

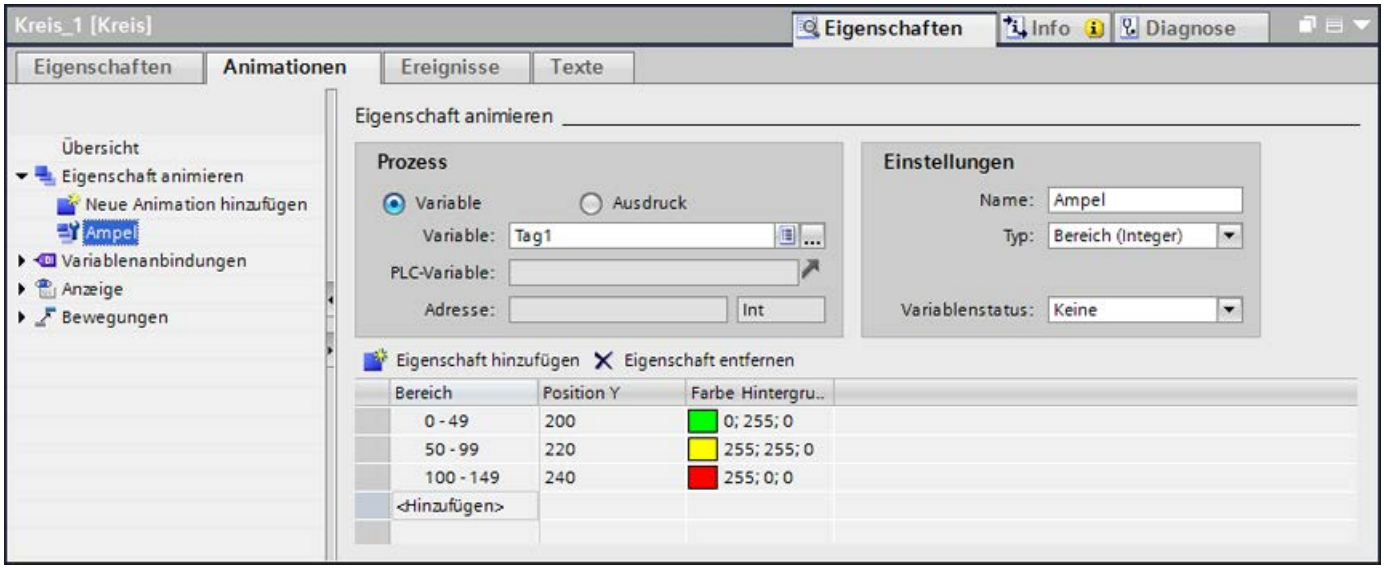

Ergebnis

Abhängig davon, in welchem Wertebereich sich die Variable in Runtime befindet, ändert der Kreis seine Position und seine Hintergrundfarbe.

Siehe auch

Dynamisieren mit Eigenschaftsanimationen (Seite 232)

1.4.3.3 Eigenschaftsanimation vom Typ Direkt projektieren

Einleitung

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, wird in Runtime die Objekteigenschaft abhängig vom Variablenwert geändert. Das folgende Beispiel zeigt die Dynamisierung eines E/A-Feldes. Das E/A-Feld ändern in Abhängigkeit von einer Variablen seinen Ausgabewert.

Voraussetzung

- Eine Variable "Tag1" ist angelegt.
- Ein Bild ist angelegt.
- Ein E/A-Feld mit dem Modus "Ausgabe" ist im Bild eingefügt und selektiert.
- Das Inspektorfenster ist geöffnet.

Animationsart, Variable und Typ wählen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animation".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
2. Wählen Sie "Eigenschaftsanimation" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
3. Geben Sie unter "Einstellungen > Name" den Namen "Füllstand" ein.
4. Wählen Sie als Typ "Direkt".
5. Aktivieren Sie "Prozess > Variable".
6. Wählen Sie "Prozess > Variable" die Variable "Tag1".

Ergebnis

Der Variablenwert in Runtime wird direkt in das E/A-Feld geschrieben.

Siehe auch

Dynamisieren mit Eigenschaftsanimationen (Seite 232)

1.4.3.4 Eigenschaftsanimation vom Typ Bool projektieren

Einleitung

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, wird in Runtime die Objekteigenschaft abhängig vom Variablenwert geändert. Im folgenden Beispiel wird eine Schaltfläche abhängig vom Wert eines Ausdrucks ein- und ausgeblendet.

Hinweis

Boolsche Variablen bei dem Typ "Bool" werden implizit mittels VB-Skriptfunktion CBool auf die VB-Skriptwerte "wahr" = -1 und "falsch" = 0 konvertiert.

Verwenden Sie für einen eindeutigen Abgleich im Ausdruck die Konstanten "`vbTrue`" und "`vbFalse`".

Voraussetzung

- Eine Variable "TCelsius" vom Typ "Uint" ist angelegt.
- Ein Bild ist angelegt.
- Eine Schalfläche ist im Bild eingefügt und selektiert.
- Das Inspektorfenster ist geöffnet.

Animationsart, Ausdruck und Typ wählen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animation".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
2. Wählen Sie "Eigenschaftsanimation" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
3. Geben Sie unter "Einstellungen > Name" den Namen "Sichtbarkeit" ein.
4. Wählen Sie als Typ "Boolean".
5. Aktivieren Sie "Prozess > Ausdruck".
6. Geben Sie im Feld den folgenden Ausdruck ein: "(('TCelsius' × 9) / 5) + 32 < 100".

   Hinweis

   Beachten Sie bei der direkten Eingabe darauf Variablennamen in einfachen Anführungszeichen zu schreiben.

   Hinweis

   Boolsche Variablen können vereinfacht ausgewertet werden:
   "'Booltag1' AND NOT 'Booltag2'".

Alternativ geben Sie den Ausdruck mit den Schaltflächen über dem Feld ein. Sie klicken z. B. auf die Schaltfläche und wählen eine Variable über die Auswahlliste.

Objekteigenschaft wählen

1. Klicken Sie auf "Eigenschaft hinzufügen". Eine Auswahlliste wird geöffnet.
2. Wählen Sie die Eigenschaft "Sichtbarkeit". Die Tabelle wird um eine Spalte "Sichtbarkeit" erweitert.
3. Aktivieren Sie in der Tabelle für den Wert "Wahr" die Eigenschaft "Sichtbar".

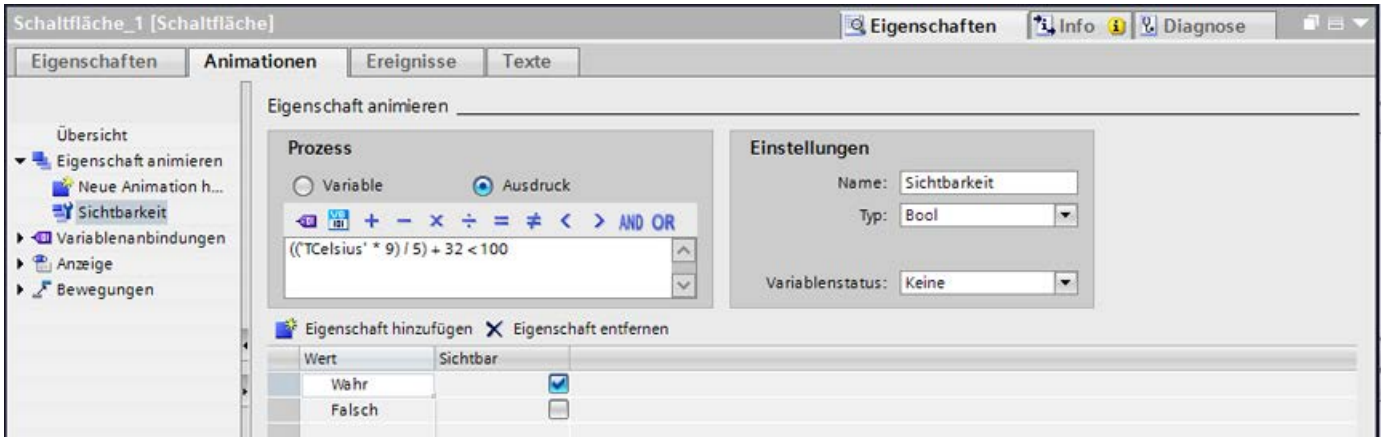

Ergebnis

Der Wert des Ausdrucks wird in Runtime ermittelt. Wenn der Wert mit dem projektierten Wert "True" übereinstimmt, wird die Schaltfläche eingeblendet.

Siehe auch

Dynamisieren mit Eigenschaftsanimationen (Seite 232)

1.4.4 Dynamisieren in der Eigenschaftsliste

1.4.4.1 Grundlagen zum Dynamisieren in der Eigenschaftsliste

Grundprinzip der Dynamisierung

Bei einer dynamisierbaren Objekteigenschaft legen Sie zunächst den Dynamisierungstyp fest. Das folgende Bild zeigt die verfügbaren Dynamisierungstypen:

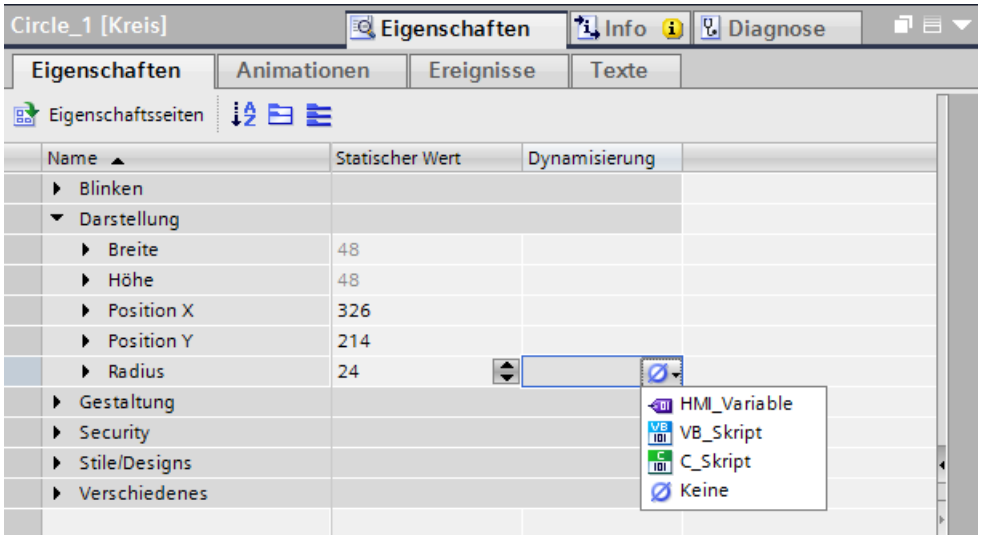

Abhängig vom Dynamisierungstyp wählen Sie entweder eine Variable über die Objektliste aus oder Sie erstellen ein lokales Skript.

Dynamisieren mit Variablen

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, können Sie zusätzlich eine Funktionsliste an folgende Ereignisse projektieren:

- Geändert
- Variablenstatus geändert
- Quality Code geändert

Die Variablen werden in dem für das Bild eingestellten Aktualisierungszyklus aktualisiert.

Das folgende Bild zeigt die mit einer Variablen dynamisierte Objekteigenschaft "Radius". Bei Wertänderung der Variablen wird zusätzlich eine Funktionsliste abgearbeitet:

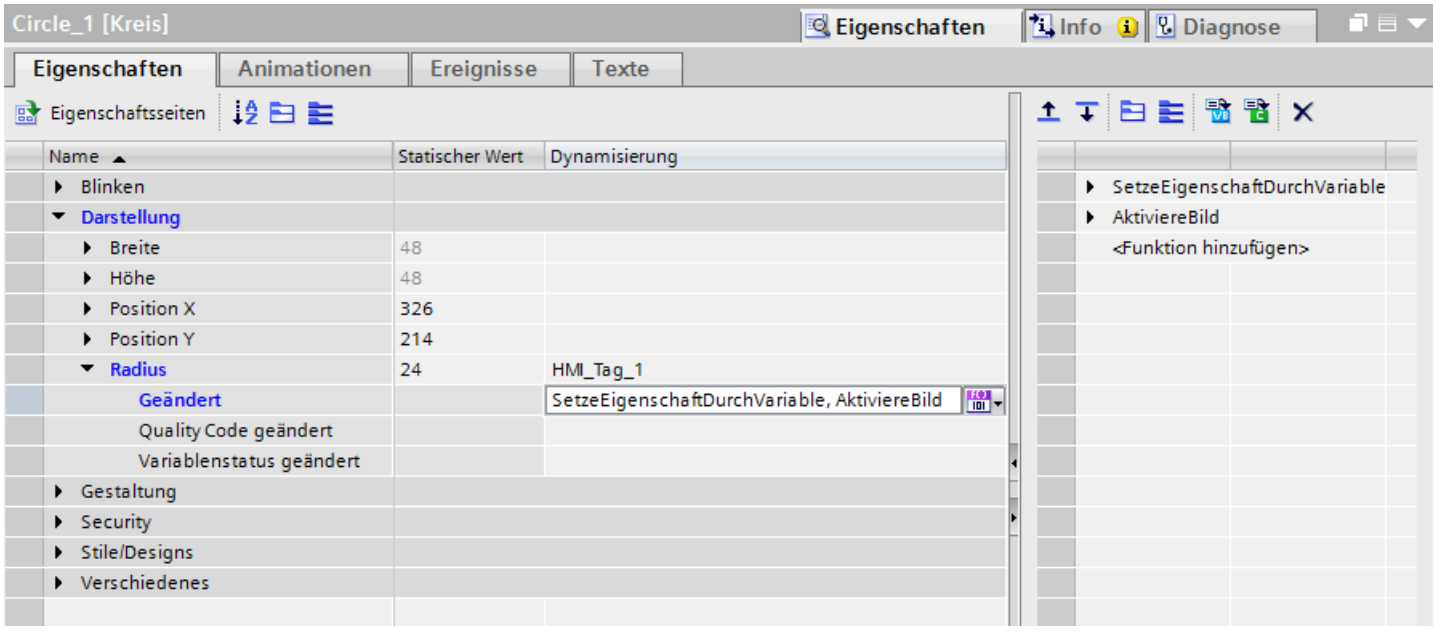

Sie können die Funktionsliste auch in ein lokales Skript umwandeln.

Quality Code

Um den Status und die Qualität einer Variablen zu prüfen, verwenden Sie den "Quality Code". Im angezeigten "Quality Code" ist die Qualität der gesamten Wertübertragung und Wertverarbeitung der jeweiligen Variablen zusammengefasst. Die Überwachung des "Quality Code" ermöglicht z. B. Rückschlüsse auf die Qualität der Variablen im Prozess.

Dynamisieren mit lokalen Skripten

Mit einem lokalen Skript dynamisieren Sie in Runtime eine Objekteigenschaft über den Rückgabewert des lokalen Skripts. Das Skript erstellen Sie direkt an der Verwendungsstelle. Das folgende Bild die Realisierung einer Statusanzeige über die Dynamisierung der Hintergrundfarbe mit einem VB-Skript:

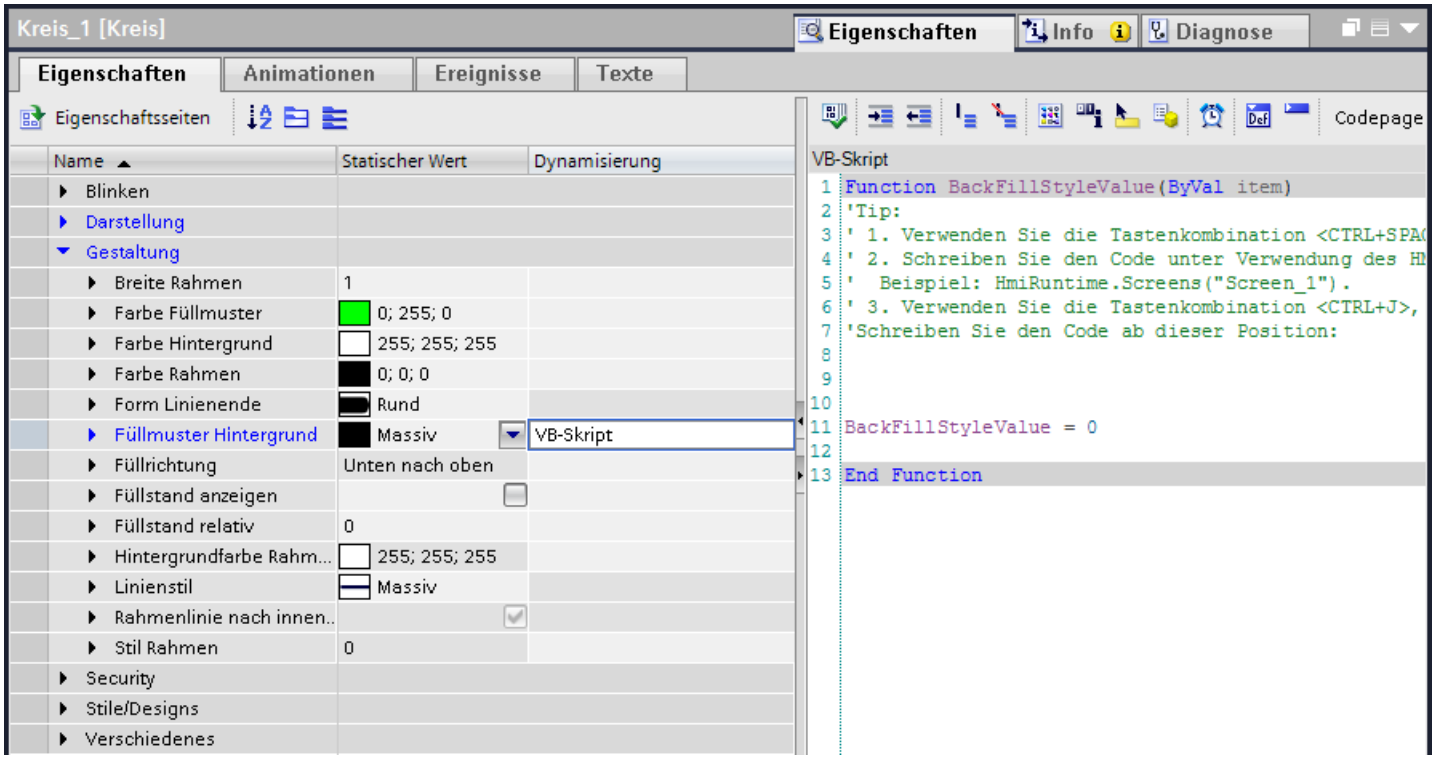

Siehe auch

Objekteigenschaft mit Variable dynamisieren (Seite 242)

Dynamisierung an eine Variablenänderung projektieren (Seite 244)

Objekteigenschaft mit lokalen Skripten dynamisieren (Seite 245)

Übersicht der Dynamisierungen einblenden (Seite 216)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Lokale Skripte (Seite 1480)

1.4.4.2 Objekteigenschaft mit Variable dynamisieren

Einleitung

Wenn Sie eine Objekteigenschaft mit einer Variablen dynamisieren, wird in Runtime die Objekteigenschaft abhängig vom Variablenwert geändert. Wenn Sie die Objekteigenschaft indirekt dynamisieren, verbinden Sie die Eigenschaft mit einer Variable, die in Runtime den Eigenschaftswert festlegt.

Voraussetzung

- Ein Objekt mit der zu dynamisierenden Eigenschaft ist selektiert.
- Eine Variable ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Markieren Sie im Bild das Objekt, dessen Eigenschaften Sie dynamisieren wollen.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf die Schaltfläche .
   Die Eigenschaftsliste mit den Eigenschaften des markierten Objekts wird angezeigt.
3. Wählen Sie die zu dynamisierende Eigenschaft.
4. Wählen Sie in der Spalte "Dynamisierung" den Eintrag "HMI_Variable" aus.
   Ein Dialog wird geöffnet

5. Wählen Sie eine Variable.
6. Um die Objekteigenschaft indirekt zu dynamisieren, aktivieren Sie "Indirekte Adressierung verwenden".

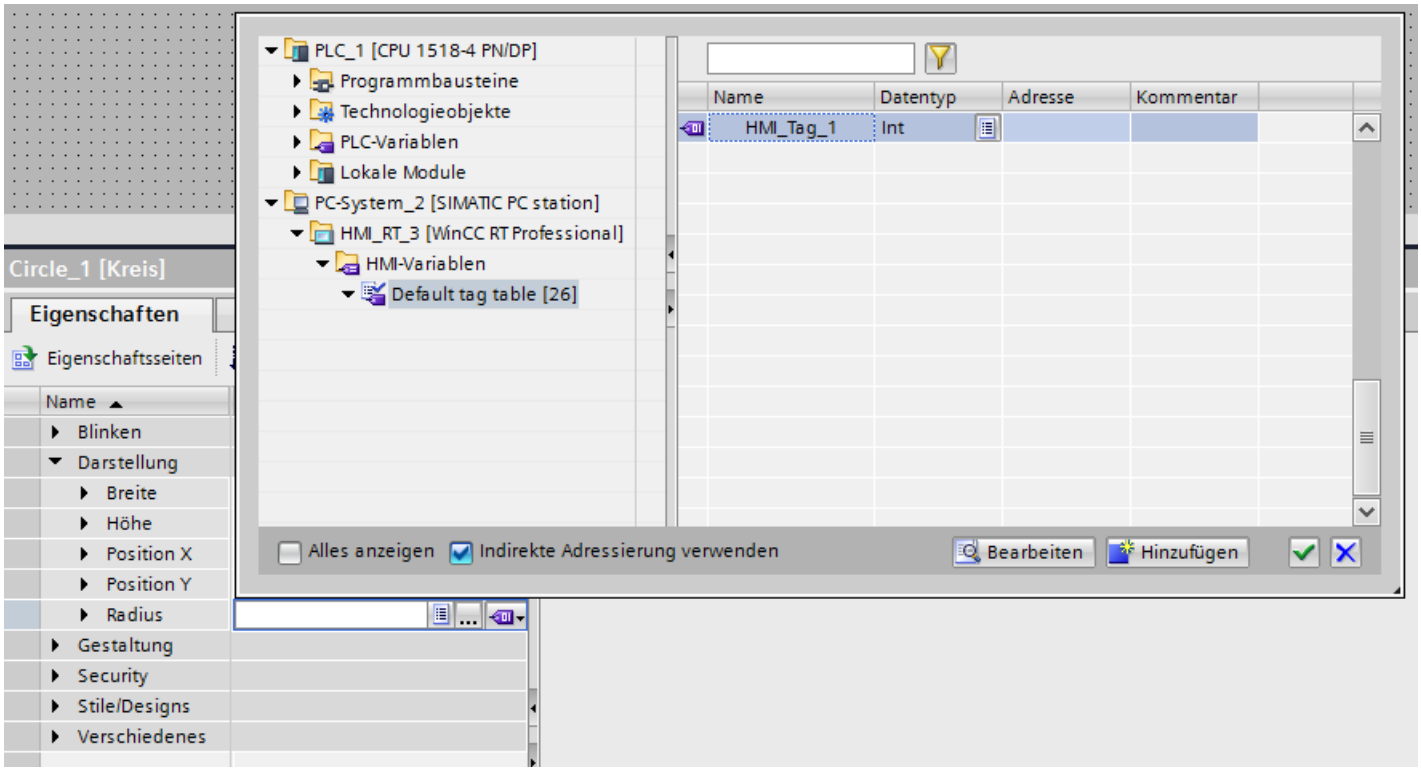

Ergebnis

In Runtime ändert sich die Objekteigenschaft entsprechend der Projektierung in der Eigenschaftsliste.

Darstellung einer Mehrfachselektion in der Eigenschaftsliste

Wenn Sie mehrere Objekte gleichzeitig markiert haben, wird die Eigenschaftsliste um die Spalte "Objekttyp" erweitert. In dieser Spalte werden die Objekte angezeigt, welche die in der Zeile angezeigte Eigenschaft unterstützen. Änderungen oder Dynamisierungen einer Eigenschaft wirken auf alle in der Spalte "Objekttyp" angezeigten Objekte.

Siehe auch

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

1.4.4.3 Dynamisierung an eine Variablenänderung projektieren

Einleitung

Wenn Sie in der Spalte "Dynamisierung" eine Variable eingetragen haben, können Sie zusätzlich an verschiedene Ereignisse eine Funktionsliste projektieren.

Voraussetzung

- Ein Objekt mit der zu dynamisierenden Eigenschaft ist selektiert.
- Eine Variable ist angelegt.
- Im Inspektorfenster ist die Eigenschaftsliste des Objektes geöffnet.
- Eine Objekteigenschaft ist mit der Variablen verbunden.

Vorgehen

1. Erweitern Sie die Eigenschaftsliste, indem Sie neben die Eigenschaft klicken, die Sie mit einer Variablen dynamisiert haben.
2. Unter der Eigenschaft werden drei weitere Zeilen mit folgenden Ereignissen angezeigt:
   - Geändert
   - Variablenstatus geändert
   - Quality Code geändert
3. Projektieren Sie eine Funktionsliste an das gewünschte Ereignis.
   Alternativ können Sie die Funktionsliste in ein lokales Skript umwandeln.

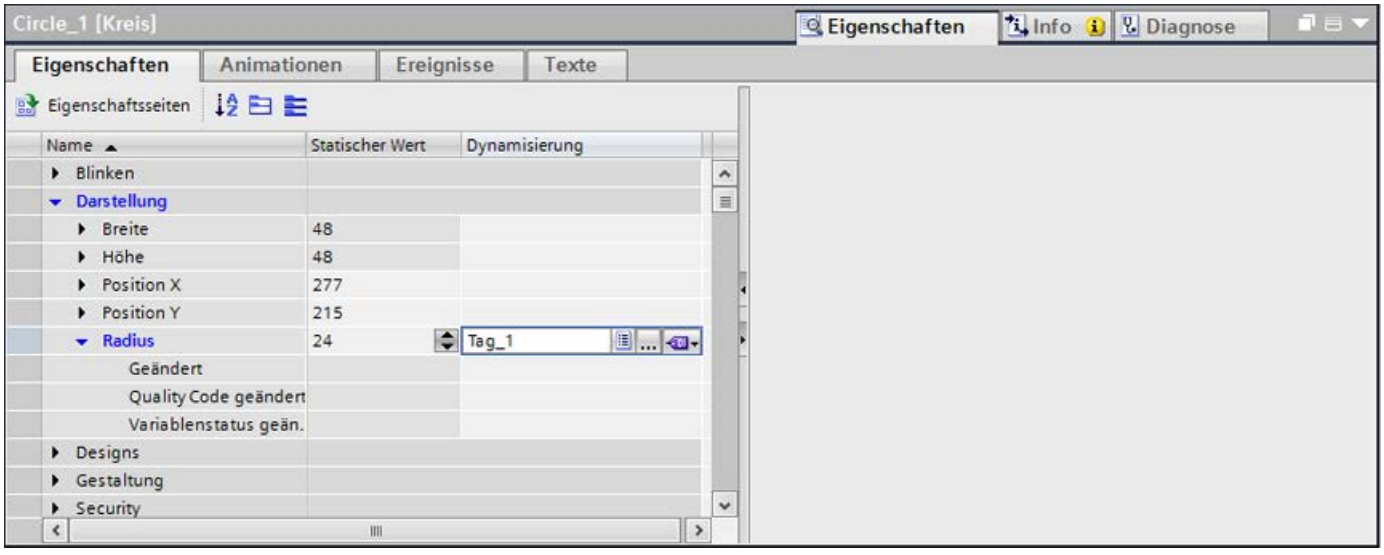

Ergebnis

In Runtime ändert sich die Objekteigenschaft entsprechend der Projektierung in der Eigenschaftsliste.

Siehe auch

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

1.4.4.4 Objekteigenschaft mit lokalen Skripten dynamisieren

Einleitung

Mit einem lokalen Skript dynamisieren Sie in Runtime eine Objekteigenschaft über den Rückgabewert des Skripts.

Voraussetzung

- Das Objekt mit der zu dynamisierenden Eigenschaft ist selektiert.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Klicken Sie im Inspektorfenster auf .
2. Selektieren Sie die zu dynamisierende Objekteigenschaft.
3. Wählen Sie in der Spalte "Dynamisierung" den Eintrag "VB-Skript" oder "C-Skript". Ein Dialog wird geöffnet.
4. Schreiben Sie den Programmcode.
5. Um den Trigger zu ändern, klicken Sie in der Symbolleiste auf " ".

Ergebnis

In Runtime ändert sich die Objekteigenschaft entsprechend der Projektierung in der Eigenschaftsliste.

Dynamisierung löschen

1. Selektieren Sie in der Eigenschaftsliste die dynamisierte Objekteigenschaft.
2. Wählen Sie in der Spalte "Dynamisierung" den Eintrag "Keine".

Siehe auch

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Erstellen von lokalen Skripten (Seite 1480)

1.4.5 Dynamisieren mit Systemfunktionen

1.4.5.1 Grundlagen zu Ereignissen

Einleitung

Bildobjekte reagieren auf Ereignisse. An die Ereignisse eines Objekts projektieren Sie eine Funktionsliste mit Systemfunktionen.

Ereignisse

Welche Ereignisse und Systemfunktionen verfügbar sind, hängt vom verwendeten Objekt ab.

Wenn der Bediener z. B. ein Bildobjekt aktiviert, wird die projektierte Systemfunktion ausgeführt.

Weitere Informationen finden Sie unter "Grundlagen zur Funktionsliste (Seite 1439)".

Siehe auch

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Systemfunktion an das Ereignis "Klicken" projektieren (Seite 246)

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 247)

Arbeiten mit Funktionslisten (Seite 1439)

1.4.5.2 Systemfunktion an das Ereignis "Klicken" projektieren

Einleitung

Sie projektieren eine Funktionsliste an ein Ereignis eines Objekts. Wenn das Ereignis in Runtime eintritt, wird die verknüpfte Systemfunktion ausgeführt.

Voraussetzungen

Ein Bild ist geöffnet.

Eine Schaltfläche ist im Bild angelegt.

Das Inspektorfenster ist geöffnet.

Vorgehen

1. Selektieren Sie die Schaltfläche.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Wählen Sie das Ereignis "Klicken".

4. Klicken Sie in der Tabelle auf "Funktion hinzufügen"
5. Wählen Sie die Systemfunktion "ZeigeMeldefenster".

Ergebnis

Wenn der Bediener in Runtime auf die Schaltfläche klickt, öffnet sich das Meldefenster im Bild.

Siehe auch

Grundlagen zu Ereignissen (Seite 246)
Grundlagen zur Funktionsliste (Seite 1439)

1.4.5.3 Beispiel: Schaltfläche zur Sprachumschaltung projektieren

Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, mit der Sie in Runtime zwischen mehreren Runtime-Sprachen umschalten.

Voraussetzungen

- Sie haben das Beispiel "Mehrsprachige Beschriftung einer Schaltfläche projektieren" ausgeführt.
- Das Bild "Screen_1" ist geöffnet.
- Die Schaltfläche ist im Bild selektiert.

Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
2. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
3. Wählen Sie die Systemfunktion "SetzeSprache" und die Einstellung "Umschalten".

Ergebnis

Sie haben der Schaltfläche die Funktion "SetzeSprache" zugewiesen. Wenn Sie in Runtime die Schaltfläche drücken, dann schaltet die Runtime-Sprache um. Die Reihenfolge der Runtime-Sprachen wird so umgesetzt, wie sie im Editor "Sprache & Schriftart" durch die Nummerierung der Sprachen vorgegeben ist.

Siehe auch

Grundlagen zu Ereignissen (Seite 246)

1.5 Arbeiten mit Funktionstasten

1.5.1 Arbeiten mit Funktionstasten

Einleitung

Eine Funktionstaste ist eine Taste am Bediengerät mit projektierbarer Funktionsbelegung. An die Ereignisse "Taste gedrückt" und "Taste loslassen" ist eine Funktionsliste projektierbar.

Sie können eine Funktionstaste global oder lokal belegen.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Globale Funktionstasten

Funktionstasten mit globaler Belegung lösen immer die gleiche Aktion aus, unabhängig vom angezeigten Bild.

Sie projektieren globale Funktionstasten im Editor "Globales Bild". Die globale Belegung gilt für alle Bilder des eingestellten Bediengeräts.

Global belegte Funktionstasten reduzieren den Projektierungsaufwand deutlich, weil Sie den globalen Tasten nicht in jedem einzelnen Bild Funktionen zuweisen.

Lokale Funktionstasten in Bildern

Lokale Funktionstasten in Bildern können in jedem Bild eine andere Aktion triggern. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.

Innerhalb eines Bildes hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

Hinweis

Wenn ein Bild mit lokalen Funktionstasten von einer Meldeanzeige oder einem Meldefenster überdeckt wird, dann sind die Funktionstasten in Runtime trotzdem aktiv. Dieser Fall kann besonders bei Bediengeräten mit kleinem Display eintreten.

Lokale Funktionstasten in Vorlagen

Lokale Funktionstasten, die in Vorlagen belegt sind, gelten für alle Bilder, die auf dieser Vorlage basieren. Sie können in jedem Bild eine andere Aktion triggern. Funktionstasten in Vorlagen belegen Sie in der Vorlage im Editor "Bilder". Innerhalb einer Vorlage hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

Belegung mit Hotkeys

Sie können Bedienobjekte mit Hotkeys belegen, z. B. Schaltflächen. Welche Hotkeys zur Verfügung stehen, ist vom Bediengerät abhängig.

Hinweis

Die Funktionstaste ist mit einer lokalen oder globalen Aktion belegt. Wenn Sie die Funktionstaste zusätzlich mit einem Hotkey belegen, wird in Runtime die Funktion des Hotkeys ausgeführt.

Grafiken

Wenn eine Funktionstaste unmittelbar neben dem Display angeordnet ist, können Sie der Funktionstaste eine Grafik zuordnen, welche die Funktion der Funktionstaste verdeutlicht.

Anzeige der Belegung

Tabelle 1-1 Die folgende Tabelle zeigt, welche Symbole die Belegung der Funktionstasten anzeigen:

Funktionstaste	Beschreibung
	Nicht belegt
	Global belegt
	Lokal belegt in der Vorlage
	Lokal belegt
	Lokal belegt (lokale Belegung der Vorlage überschreibt die globale Belegung)
	Lokal belegt (lokale Belegung überschreibt die globale Belegung)
	Lokal belegt (lokale Belegung überschreibt lokale Belegung der Vorlage)

Funktionstaste	Beschreibung
	Lokal belegt (lokale Belegung überschreibt lokale Belegung der Vorlage, die bereits die globale Belegung überschreibt)
	Schaltfläche mit Bildnavigation belegt

Hinweis

Basic Panels

Der Editor "Bildnavigation" steht für Basic Panels nicht zur Verfügung.

Siehe auch

Funktionstasten global belegen (Seite 250)

Funktionstasten lokal belegen (Seite 252)

Einer Funktionstaste eine Funktion zuweisen (Seite 253)

Bedienberechtigung für eine Funktionstaste vergeben (Seite 255)

Einer Funktionstaste eine Grafik zuweisen (Seite 257)

Beispiel: Funktionstasten zur Bildnavigation verwenden (Seite 261)

1.5.2 Funktionstasten global belegen

Einleitung

Sie definieren die globale Belegung einer Funktionstaste im Editor "Globales Bild". Die globale Belegung gilt in allen Bildern des eingestellten Bediengeräts.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Voraussetzung

- Das Projekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um einer Funktionstaste eine bildunabhängige Funktion zuzuweisen, gehen Sie folgendermaßen vor:

1. Um den Editor "Globales Bild" zu öffnen, doppelklicken Sie im Projektfenster in der Gruppe "Bildverwaltung" auf "Globales Bild".
2. Selektieren Sie die gewünschte Funktionstaste.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.

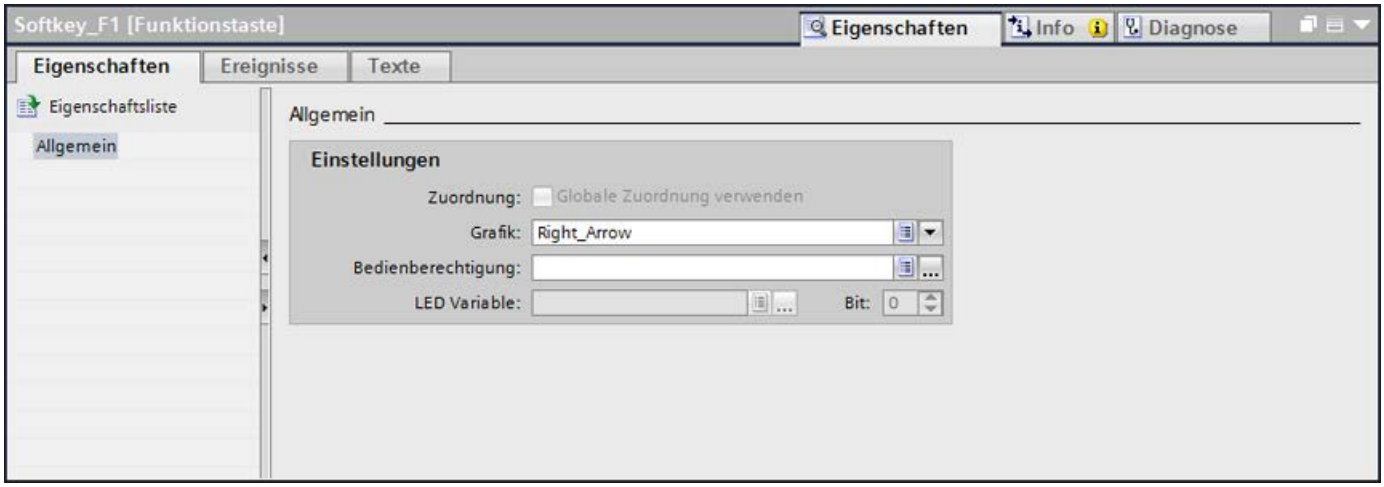

3. Konfigurieren Sie unter "Allgemein" eine Grafik und eine Bedienberechtigung für die Funktionstaste.
4. Projektieren Sie unter "Ereignisse" eine Funktionsliste an das gewünschte Ereignis.

Ergebnis

Wenn nicht eine lokale Belegung die globale Belegung überschreibt, dann ändert sich in allen Bildern des eingestellten Bediengeräts die Belegung der Funktionstaste entsprechend Ihrer Eingabe.

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.5.3 Funktionstasten lokal belegen

Einleitung

Funktionstasten werden global und lokal belegt. Eine lokale Belegung der Funktionstasten gilt nur für das Bild oder die Vorlage, in der sie definiert wurde. Folgende lokale Funktionstasten gibt es:

- Lokale Funktionstasten eines Bildes
  Für jedes Bild belegen Sie die Funktionstaste unterschiedlich mit Funktionen. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.
- Lokale Funktionstasten einer Vorlage
  Sie belegen die Funktionstasten in einer Vorlage. Die Belegung gilt für alle Bilder, die auf dieser Vorlage basieren und nicht durch eine lokale Belegung in einem Bild überschrieben sind.

Eine lokale Belegung überschreibt die globale Belegung einer Funktionstaste.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Verwendung vorhandener Belegungen

Die Option zur Verwendung von vorhandenen Belegungen ist im Inspektorfenster folgendermaßen benannt:

- In einer Vorlage: "Globale Zuordnung verwenden"
- In einem Bild:
  - Bild basiert auf einer Vorlage: "Lokale Vorlage verwenden"
  - Bild basiert nicht auf einer Vorlage: "Globale Zuordnung verwenden"

Die Option "Lokale Vorlage verwenden" beinhaltet die Verwendung der lokalen Belegung in der Vorlage und die globale Belegung.

Voraussetzung

- Das Bild oder die Vorlage, in der Sie eine Funktionstaste lokal belegen wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bild oder in der Vorlage die gewünschte Funktionstaste.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".

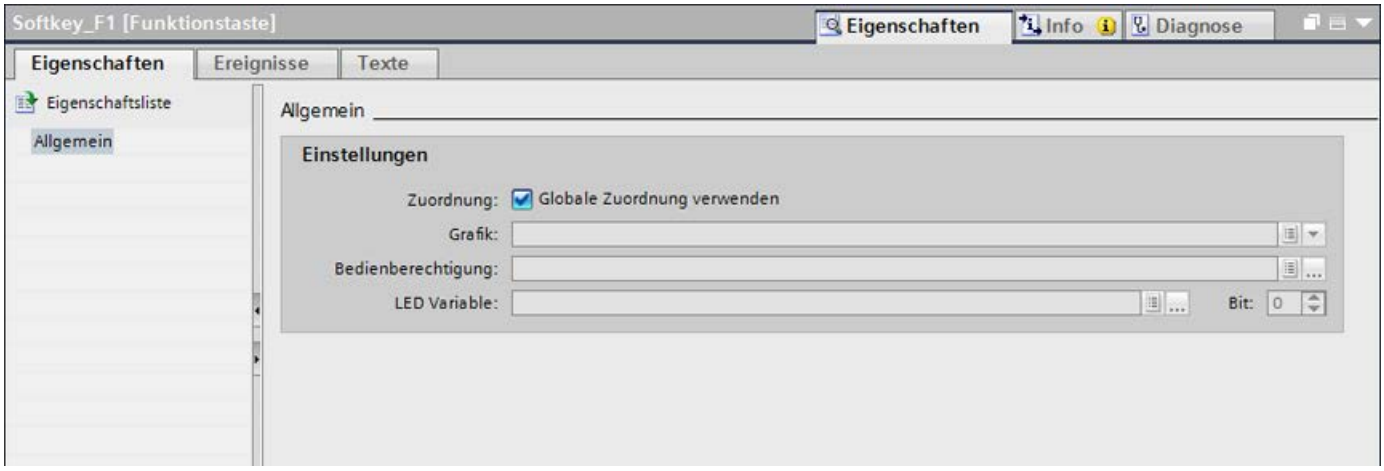

3. Deaktivieren Sie die Option "Lokale Vorlage verwenden" oder "Globale Zuordnung verwenden".
4. Konfigurieren Sie unter "Allgemein" eine Grafik und eine Bedienberechtigung für die Funktionstaste.
5. Projektieren Sie unter "Ereignisse" eine Funktionsliste an das gewünschte Ereignis.

Ergebnis

Im Bild oder in der Vorlage ist die Funktionstaste mit den projektierten Funktionen belegt.

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.5.4 Einer Funktionstaste eine Funktion zuweisen

Einleitung

Eine Funktionstaste kann zwei Zustände haben:

- Gedrückt: Definiert durch das Ereignis "Taste gedrückt"
- Losgelassen: Definiert durch das Ereignis "Taste loslassen"

Beide Ereignisse projektieren Sie im Inspektorfenster der Funktionstaste. Sie können jedem Ereignis eine Funktionsliste zuweisen, die Systemfunktionen oder Skripte enthält. In Runtime wird die Funktionsliste beim Auftreten des Ereignisses abgearbeitet.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Hinweis

Basic Panels

Skripte stehen für Basic Panels nicht zur Verfügung.

Voraussetzung

Wenn Sie eine Funktionstaste global belegen wollen:

- Der Editor "Globales Bild" ist geöffnet.

Wenn Sie eine Funktionstaste lokal belegen wollen:

- Das Bild, in dem Sie eine Funktionstaste belegen wollen, ist geöffnet.

Wenn Sie eine Funktionstaste in einer Vorlage lokal belegen wollen:

- Die Vorlage, in der Sie eine Funktionstaste belegen wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Markieren Sie die Funktionstaste, die Sie definieren wollen.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Projektieren Sie im Inspektorfenster unter "Eigenschaften" in der Gruppe "Allgemein" die Funktionsliste an das gewünschte Ereignis.

Ergebnis

Wenn der Bediener die Funktionstaste drückt oder loslässt, dann wird in Runtime die Funktionsliste abgearbeitet.

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.5.5 Bedienberechtigung für eine Funktionstaste vergeben

Einleitung

In WinCC können Sie in Runtime für eine Funktionstaste eine Bedienberechtigung vergeben. Damit schränken Sie bereits beim Erstellen Ihres Projekts den Benutzerkreis für Funktionstasten auf spezielle Personen oder Bedienergruppen ein. Nur autorisiertes Personal verändert dann in Runtime wichtige Parameter und Einstellungen.

Wenn Sie einen Zugriffsschutz projektieren, verhindern Sie Fehlbedienungen und erhöhen die Sicherheit der Anlage oder Maschine.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Voraussetzung

- Die Benutzergruppen sind definiert.

Wenn Sie eine globale Funktionstaste schützen wollen:

- Der Editor "Globales Bild" ist geöffnet.

Wenn Sie eine lokale Funktionstaste eines Bildes oder einer Vorlage schützen wollen:

- Das Bild oder die Vorlage mit der Funktionstaste ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Markieren Sie die gewünschte Funktionstaste.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".

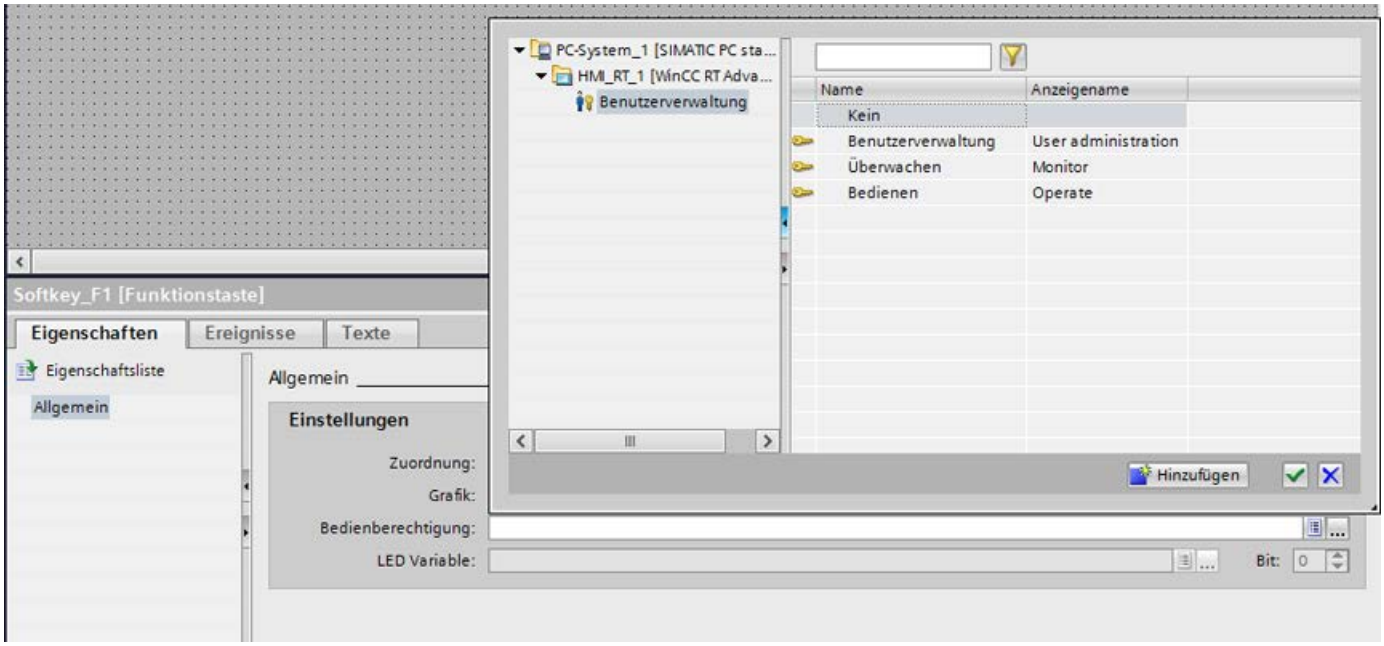

3. Wählen Sie aus der Liste "Berechtigung" die Benutzergruppe, welche die Funktionstaste in Runtime bedienen darf.

Ergebnis

Die Bedienberechtigung ist projektiert.

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.5.6 Einer Funktionstaste eine Grafik zuweisen

Einleitung

Um die Funktion einer Taste zu verdeutlichen, können Sie in einem Bild eine Grafik neben der Funktionstaste einfügen. Grafiken weisen Sie nur solchen Funktionstasten zu, die an den Bildschirm des Bediengeräts angrenzen.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Hinweis

Zuweisen von Grafiken zu einer Funktionstaste

Einer Funktionstaste kann nur dann eine Grafik zugewiesen werden, wenn der untere Rand des Permanentfensters nicht den Bereich der Grafik der Funktionstaste überdeckt.

Voraussetzung

Wenn Sie einer globalen Funktionstaste eine Grafik zuweisen wollen:

- Der Editor "Globales Bild" ist geöffnet.

Wenn Sie einer lokalen Funktionstaste in einem Bild oder einer Vorlage eine Grafik zuweisen wollen:

- Das Bild oder die Vorlage mit der betreffenden Funktionstaste ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die Grafik für die Funktionstaste ist erstellt.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Markieren Sie die gewünschte Funktionstaste.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".

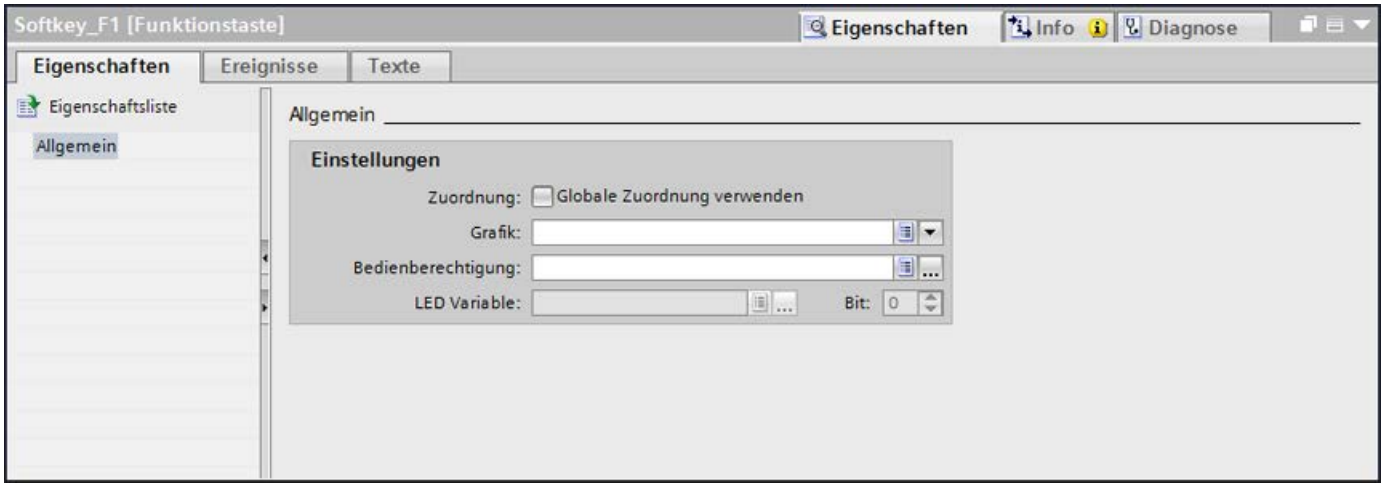

3. Klicken Sie unter "Grafik" in die Liste.
   Die Grafiksammlung des WinCC-Projekts wird angezeigt. Links werden Ihnen die externen Grafiken angezeigt, die bereits in der Sammlung hinterlegt sind. Im rechten Anzeigebereich sehen Sie eine Vorschau der Grafik, die in der Sammlung markiert ist.

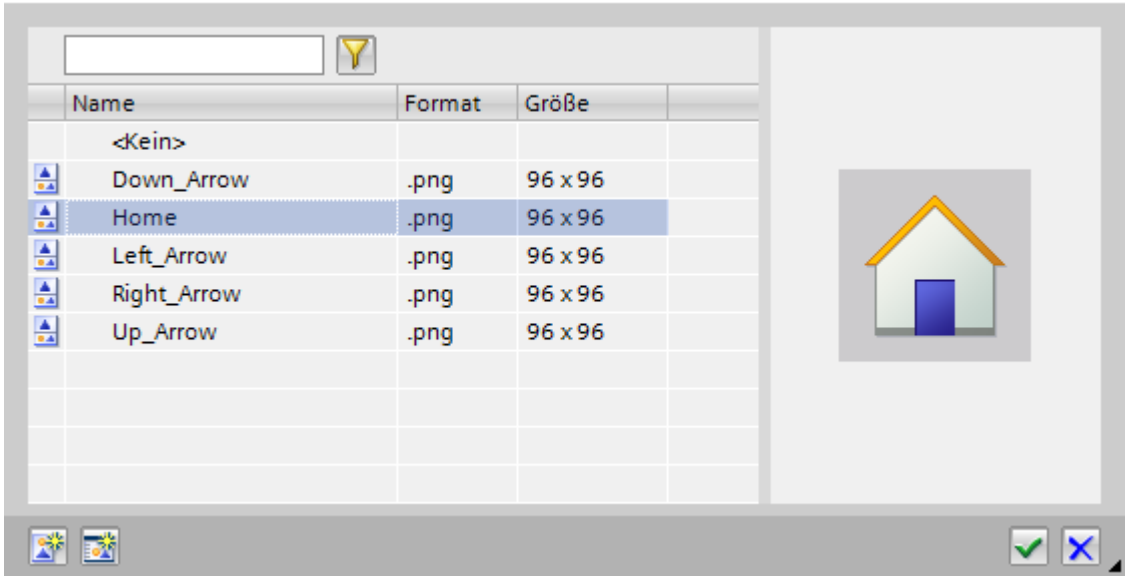

Mit den Symbolen und stellen Sie die Sammlung als Thumbnails oder als Liste dar.
Um OLE-Objekte mit dem dazugehörigen Grafikprogramm zu öffnen und zu bearbeiten, doppelklicken Sie auf das Objekt.

4. Klicken Sie in der Sammlung auf die gewünschte Grafik oder hinterlegen Sie die Grafik in der Grafiksammlung.
   Die Grafik wird rechts in der Vorschau angezeigt.
5. Um die Grafik in das Bild einzufügen, klicken Sie auf "Auswählen".
   Mit "Abwählen" entfernen Sie eine Grafik aus dem Bild.

Ergebnis

Die Grafik wird neben der Funktionstaste angezeigt.

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.5.7 LED-Variablen projektieren

Voraussetzungen

- Bediengerät mit Tastenbedienung ist angelegt.
- LED Variable ist angelegt.
- Das Globale Bild ist geöffnet.

Vorgehensweise

1. Klicken Sie auf eine F-Taste des Bediengeräts.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften" auf "Allgemein".

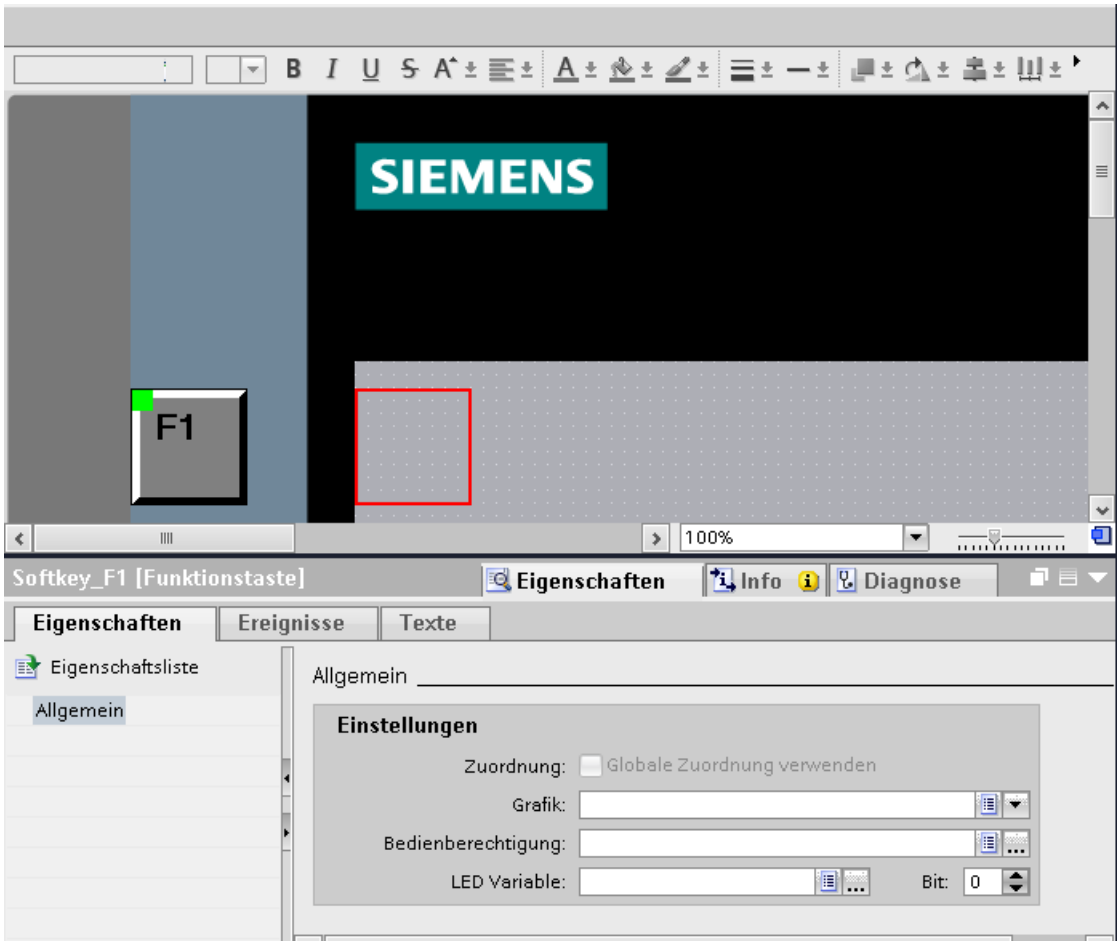

3. Wählen Sie im Bereich "Allgemein > Einstellungen" unter "LED Variable" eine Variable aus.
4. Geben Sie unter "Bit" die passende Bit-Nummer ein.
   Die passende Bitnummer ist abhängig vom Bediengerät und der Belege der Ein- und Ausgänge am Bediengerät.

1.5.8 Beispiel: Funktionstasten zur Bildnavigation verwenden

Aufgabe

In diesem Beispiel belegen Sie eine Funktionstaste lokal in einem Bild. Wenn der Bediener diese Funktionstaste drückt, wird ein Bildwechsel zu einem vordefinierten Bild z. B. mit dem Namen "Kessel2" getriggert.

Hinweis

Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

Voraussetzung

- Das Bild, in dem Sie die Funktionstaste belegen wollen, ist geöffnet.
- Das Bild "Kessel2" ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um die Funktion "AktiviereBild" zu verwenden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte Funktionstaste.
   Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie auf "Allgemein".
3. Um eine globale Belegung zu überschreiben, deaktivieren Sie die Option "Lokale Vorlage verwenden".
4. Klicken Sie unter "Ereignisse" auf "Taste gedrückt".

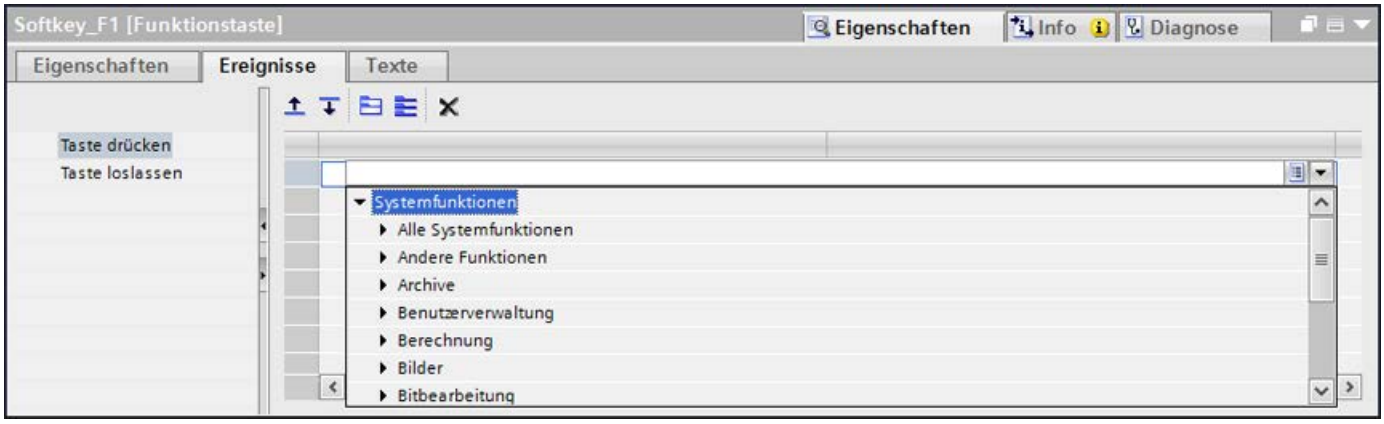

5. Wählen Sie aus der Liste die Systemfunktion "AktiviereBild" aus.
   Die Funktion "AktiviereBild" mit den Parametern "Bildname" und "Objektnummer" wird unter der Gruppe "Funktionsliste" angezeigt.

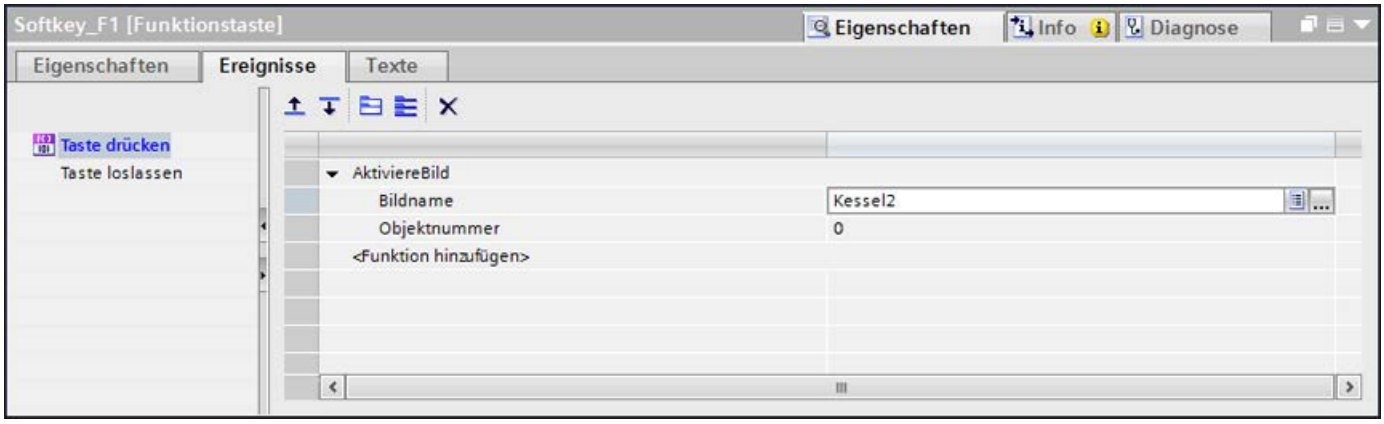

6. Wählen Sie aus der Liste "Bildname" das Bild "Kessel 2" aus.

Ergebnis

Wenn der Bediener in Runtime die ausgewählte Funktionstaste drückt, wechselt er zum Bild "Kessel 2".

Siehe auch

Arbeiten mit Funktionstasten (Seite 248)

1.6 Arbeiten mit Ebenen

1.6.1 Grundlagen zum Arbeiten mit Ebenen

Ebenen

Um Objekte eines Bildes differenziert zu bearbeiten, verwenden Sie Ebenen. Ein Bild besitzt 32 Ebenen, die Sie beliebig benennen können. Wenn Sie den Ebenen Objekte zuordnen, dann legen Sie damit die Tiefenstaffelung des Bildes fest. Objekte der Ebene 0 liegen im Hintergrund des Bildes, Objekte der Ebene 31 im Vordergrund.

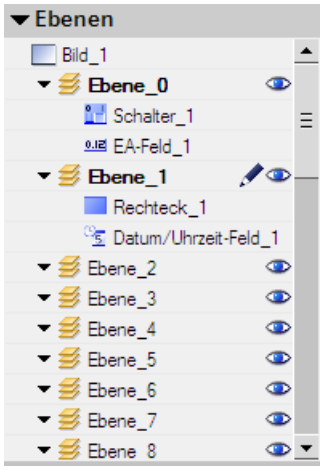

Die Objekte einer einzelnen Ebene sind auch hierarchisch angeordnet. Wenn Sie ein Bild erstellen, wird das zuerst eingefügte Objekt innerhalb der Ebene hinten angeordnet. Jedes weitere Objekt wird jeweils eine Position weiter vorn eingefügt. Innerhalb einer Ebene können Sie die Objekte nach vorne und hinten verschieben.

Prinzip der Ebenentechnik

Von den 32 Ebenen ist immer eine Ebene aktiv. Wenn Sie Objekte in ein Bild einfügen, werden diese standardmäßig der aktiven Ebene zugeordnet. Die Nummer der aktiven Ebene wird im Inspektorfenster des Bildes angezeigt und in der Task Card "Layout > Ebenen".

Wenn Sie ein Bild öffnen, werden immer alle 32 Ebenen des Bildes angezeigt. Im Inspektorfenster des Bildes und in der Task Card "Layout > Ebenen" können Sie alle Ebenen außer der aktiven Ebene ausblenden. Sie bearbeiten dann gezielt die Objekte der aktiven Ebene.

In der Baumansicht der Palette "Ebenen" in der Task Card "Layout" verwalten Sie Ebenen und Objekte per Drag&Drop und über das Kontextmenü.

Anwendungsbeispiele

Ebenen verwenden Sie z. B. in folgenden Fällen:

- Um die Beschriftung von Objekten auszublenden, während Sie editieren
- Um Objekte, z. B. Meldefenster, auszublenden, während Sie weitere Objekte projektieren

Einschränkungen für die Objekte in RT Professional

Einige Controls in WinCC RT Professional erscheinen in Runtime in einem separaten Fenster.

Folgende Controls werden in Runtime unabhängig von den Ebenen angezeigt:

- Tabellenanzeige
- F(x)-Kurvenanzeige
- F(t)-Kurvenanzeige
- Wertetabelle
- Rezepturanzeige
- Meldeanzeige
- .Net-Control
- HTML-Browser
- Bildfenster

Die Zuordnung dieser Controls zu bestimmten Ebenen ist in Runtime nicht wirksam.

Siehe auch

Grundlagen zu Bildern (Seite 33)

Objekte auf Ebenen verschieben (Seite 264)

Aktive Ebene einstellen (Seite 265)

Ebenen ein- und ausblenden (Seite 267)

Ebenen ein- und ausblenden (Seite 269)

Ebenen und Objekte in Runtime verbergen (Seite 270)

Ebenen umbenennen (Seite 271)

1.6.2 Objekte auf Ebenen verschieben

Einleitung

Standardmäßig liegen neu eingefügte Objekte auf der aktiven Ebene. Sie können ein Objekt nachträglich einer anderen Ebene zuordnen.

Voraussetzung

- Ein Bild mit einem Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt im Bild.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes > Ebene" die Ebene ein, auf die Sie das Objekt schieben wollen.

Alternativ wählen Sie in der Task Card "Layout" das Objekt und ziehen Sie es per Drag&Drop auf die gewünschte Ebene.

Reihenfolge von Objekten ändern

1. Selektieren Sie das Objekt im Bild.
   Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Um das Objekt in der Reihenfolge nach vorne oder nach hinten zu verschieben, wählen Sie im Kontextmenü den Befehl "Reihenfolge > Eins nach hinten" oder "Eins nach vorne". Alternativ verwenden Sie die Schaltfläche oder in der Symbolleiste.

Ergebnis

Das Objekt wird der ausgewählten Ebene zugewiesen und ganz im Vordergrund eingeordnet.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

1.6.3 Aktive Ebene einstellen

Einleitung

In einem Bild sind die Bildobjekte immer einer von 32 Ebenen zugeordnet. Im Bild ist immer eine Ebene aktiv. Wenn Sie Objekte in ein Bild einfügen, werden diese standardmäßig der aktiven Ebene zugeordnet.

Die Nummer der aktiven Ebene wird in der Symbolleiste "Ebene" angezeigt. Die aktive Ebene wird in der Task Card "Layout > Ebenen" mit dem Symbol gekennzeichnet.

Zu Beginn der Projektierung ist die Ebene 0 die aktive Ebene. Während des Projektierens aktivieren Sie bei Bedarf eine andere Ebene.

Voraussetzung

- Ein Bild mit mindestens einem Objekt ist geöffnet.
- Das Inspektorfenster des aktiven Bildes ist geöffnet.

Vorgehen

1. Klicken Sie im Inspektorfenster des aktuellen Bildes "Eigenschaften > Eigenschaften > Ebenen".
2. Geben Sie unter " Einstellungen > Aktive Ebene" die Nummer der Ebene ein.

Alternatives Vorgehen

1. Wählen Sie in der Task Card "Layout > Ebenen".
2. Wählen Sie im Kontextmenü einer Ebene den Befehl "Auf aktiviert setzen".

Ergebnis

Die Ebene mit der angegebenen Nummer wird zur aktiven Ebene.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

1.6.4 Ebenen ein- und ausblenden

Einleitung

Sie blenden die Ebenen eines Bildes bei Bedarf ein oder aus. Sie legen fest, welche Ebenen im Engineering System eingeblendet werden. Wenn Sie ein Bild öffnen, sind immer alle Ebenen eingeblendet.

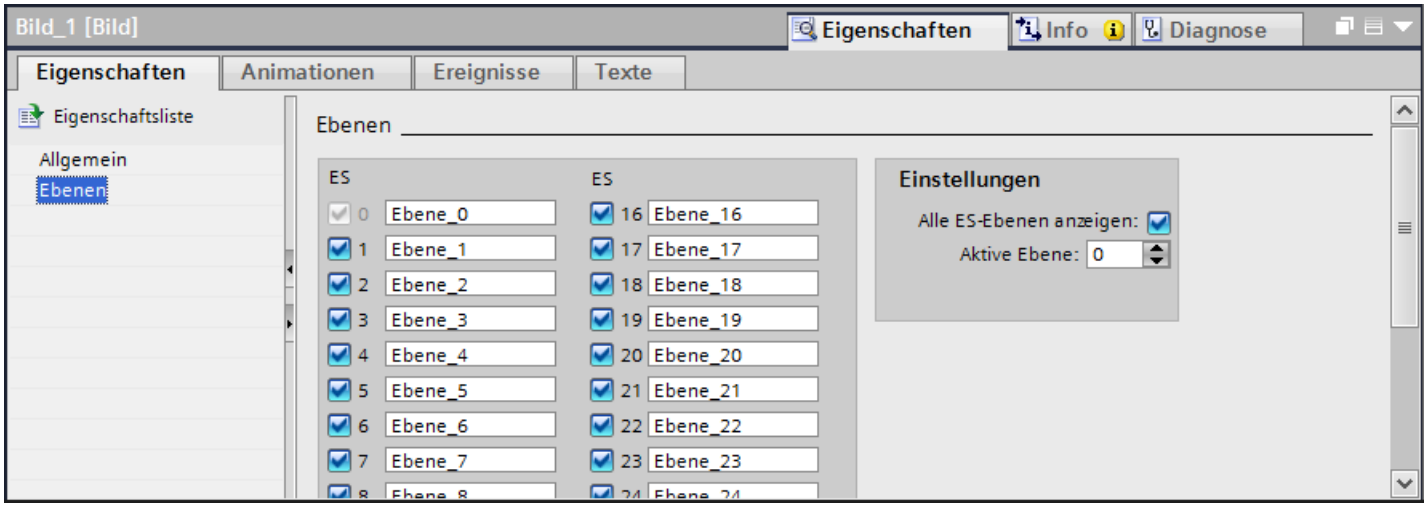

Voraussetzung

- Das Bild ist geöffnet.
- Die Task Card "Layout" ist geöffnet.

Vorgehen

1. Wählen Sie in der Task Card "Layout > Ebenen" die Ebene, die Sie aus- oder einblenden wollen.
2. Klicken Sie auf eines der Symbole neben der entsprechenden Ebene:
   - eingeblendete Ebene wird ausgeblendet
   - ausgeblendete Ebene wird eingeblendet

   Hinweis

   Die aktive Ebene können Sie nicht ausblenden.

Alternatives Vorgehen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist.
   Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen".

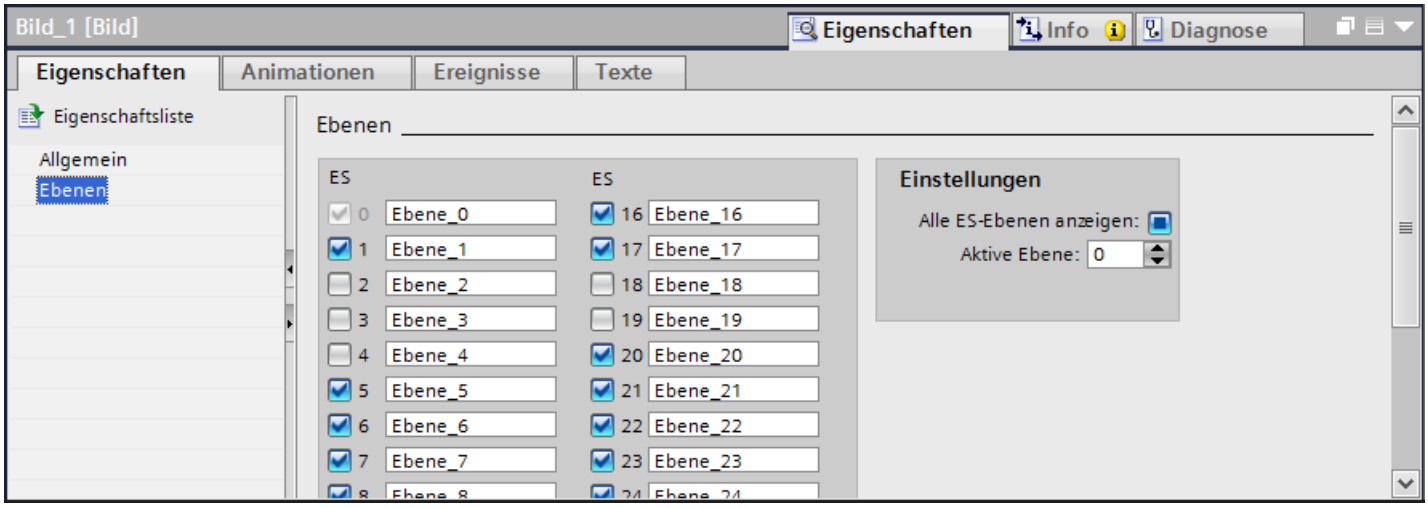

3. Deaktivieren Sie in der Liste die Ebenen, die Sie ausblenden wollen.
   Wenn Sie für eine Ebene "Alle ES-Ebenen" aktivieren, werden die Objekte dieser Ebene im Engineering System angezeigt.

Ergebnis

Die Ebenen werden entsprechend Ihren Einstellungen angezeigt.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

1.6.5 Ebenen ein- und ausblenden

Einleitung

Bei Bedarf blenden Sie die Ebenen eines Bildes ein oder aus. Sie legen fest, ob die Ebenen im Engineering System und/oder in Runtime eingeblendet werden. Wenn Sie ein Bild öffnen, sind immer alle Ebenen eingeblendet.

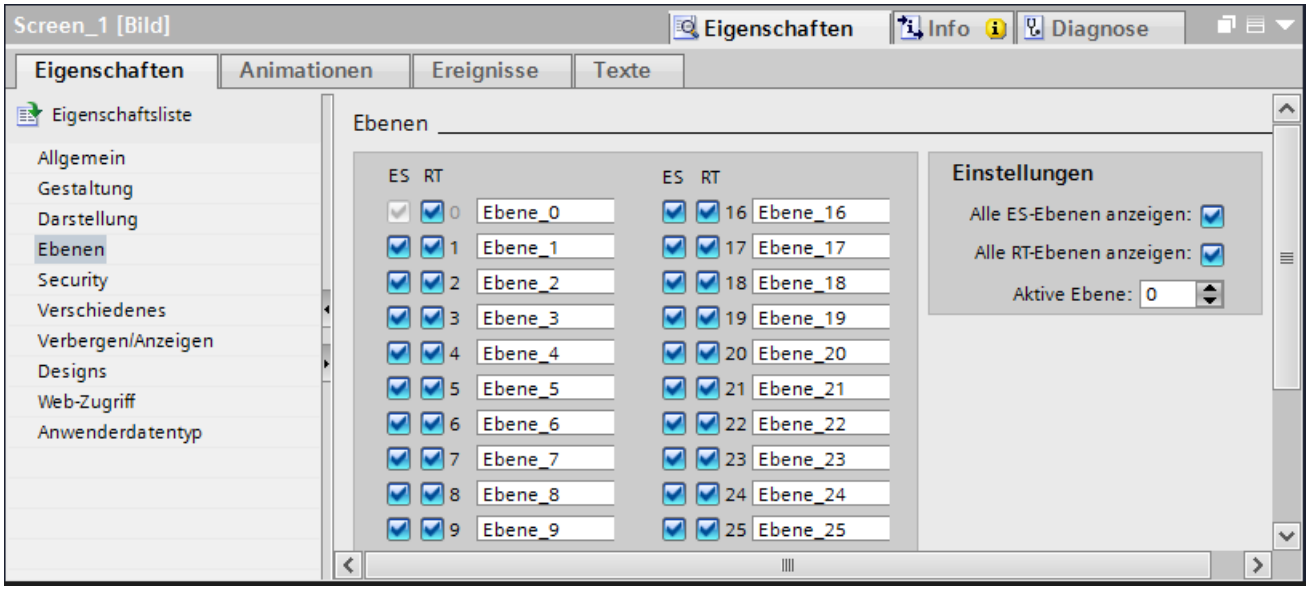

Voraussetzung

- Das Bild ist geöffnet.

Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen".
2. Deaktivieren Sie die Ebenen, die im Engineering System nicht angezeigt werden sollen.
3. Deaktivieren Sie die Ebenen, die in Runtime nicht angezeigt werden sollen.

Alternativ können Sie die Ebenen auch in der Task Card "Layout > Ebenen" aktivieren und deaktivieren.

Ergebnis

In ES und in Runtime werden die aktivierten Ebenen angezeigt.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

1.6.6 Ebenen und Objekte in Runtime verbergen

Einleitung

Abhängig vom Zoomfaktor lassen sich Ebenen und die darin abgelegten Objekte systemabhängig in Runtime ein- und ausblenden. So lassen Sie sich z. B. eine Ebene nur anzeigen, wenn ein ausreichender Zoomfaktor das erforderliche Detail auch zeigt.

Sie können Objekte abhängig von der definierten Pixelgröße verbergen oder anzeigen.

Voraussetzung

Ein Bild ist geöffnet.

Ebenen verbergen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verbergen/Anzeigen > Für Ebenen aktivieren".

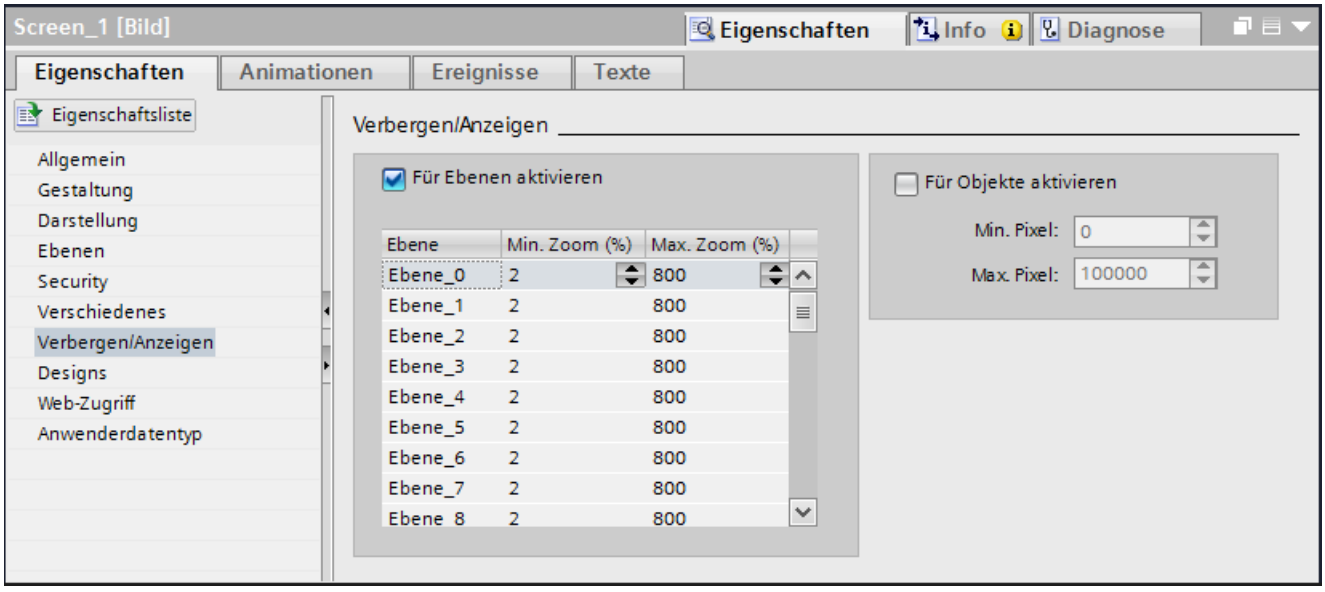

3. Geben Sie in der Tabelle das Minimum und das Maximum für den Zoomfaktor an, innerhalb dessen die jeweilige Ebene angezeigt wird.

Objekte verbergen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist.
2. Aktivieren Sie im Inspektorfenster unter ""Eigenschaften > Eigenschaften > Verbergen/Anzeigen > Für Objekte aktivieren".

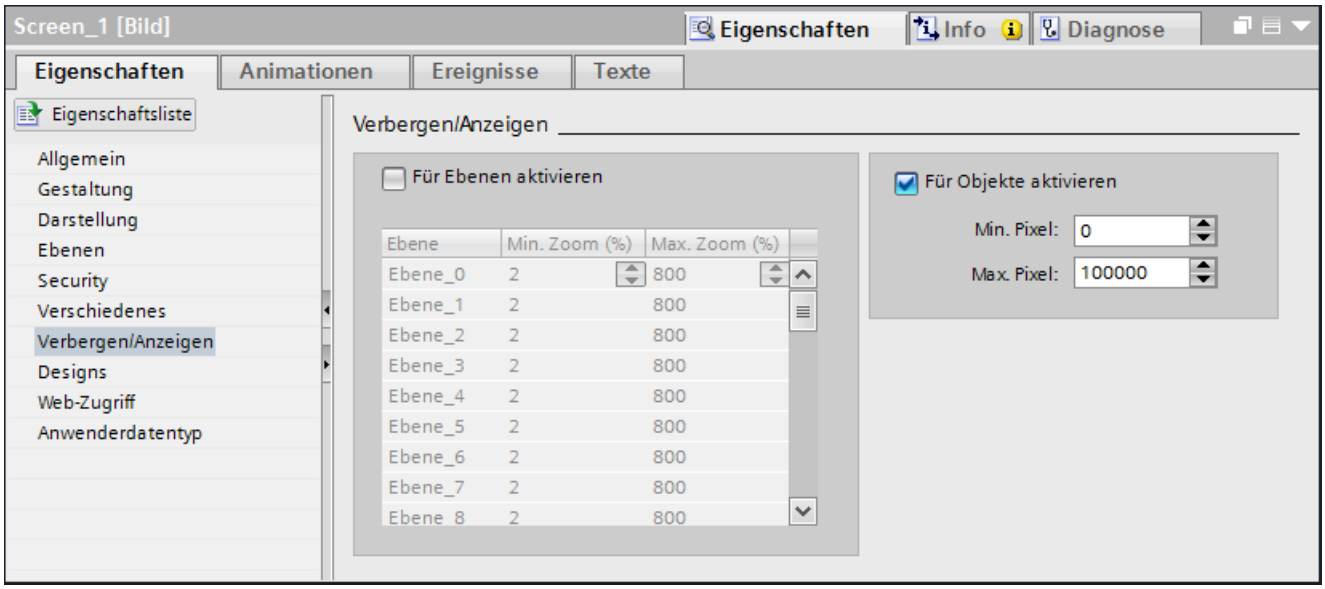

3. Geben Sie unter "Min. Pixels" das Minimum der Größe ein, in der die Objekte angezeigt werden.
4. Geben Sie unter "Max. Pixels" das Maximum der Größe ein, in der die Objekte angezeigt werden.

Ergebnis

In Runtime werden Ebenen, die sich außerhalb des Zoomfaktors befinden, nicht angezeigt. Objekte, die sich außerhalb der angegebenen Größe befinden, werden ebenfalls nicht angezeigt.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

Unterstützte Gesten (Seite 2403)

1.6.7 Ebenen umbenennen

Einleitung

Wenn Sie ein Bild anlegen, sind die 32 Ebenen standardmäßig durchnummeriert. Zur besseren Übersicht benennen Sie die Ebenen nach Ihren Anforderungen um.

Voraussetzung

- Das Bild ist geöffnet.

Vorgehen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist.
   Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Ebenen".

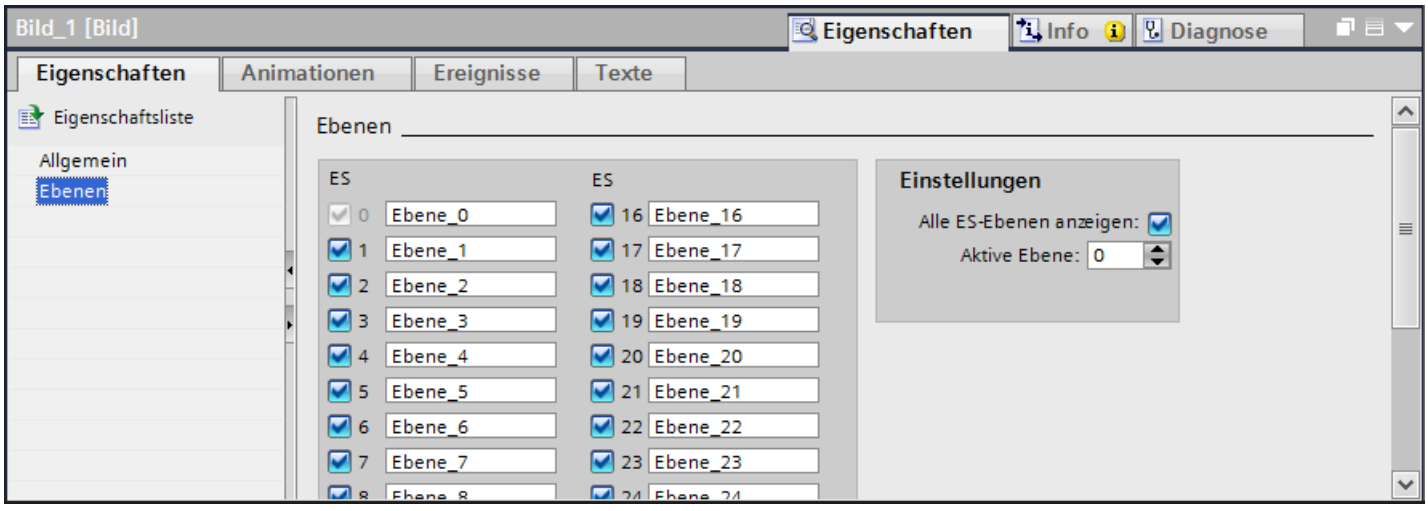

3. Geben Sie den neuen Ebenennamen ein.

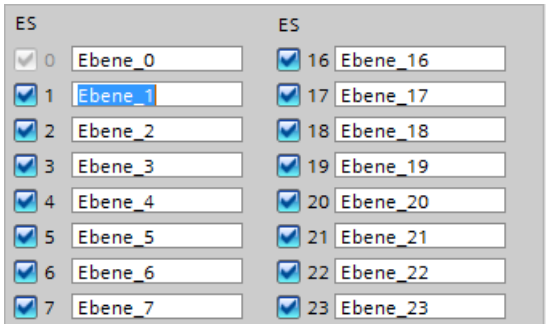

Ergebnis

Die Ebene wird mit neuem Namen angezeigt.

Siehe auch

Grundlagen zum Arbeiten mit Ebenen (Seite 263)

1.7 Arbeiten mit Bildbausteinen

1.7.1 Grundlagen zu Bildbausteinen

1.7.1.1 Grundlage zu Bildbausteinen

Einleitung

Bildbausteine sind eine konfigurierte Gruppe von Anzeige- und Bedienobjekten, die Sie zentral in einer Bibliothek verwalten und ändern. Einen Bildbaustein verwenden Sie bei Bedarf in mehreren Projekten. Die Bildbausteine sind in der Projektbibliothek abgelegt.

Verwendung

Bildbausteine verwenden Sie, um individuell projektierte Anzeige- und Bedienobjekte zu erstellen. Sie verwenden Bildbausteine mehrfach im Projekt oder projektübergreifend. Alle Instanzen eines Bildbausteins im Projekt ändern Sie zentral. Dadurch reduziert sich der Projektierungsaufwand.

Typen und Instanzen

Um die zentrale Änderbarkeit zu unterstützen, beruhen Bildbausteine auf einem Typ-Instanz-Modell. In Typen legen Sie zentral Eigenschaften für ein Objekt an. Die Instanzen stellen lokale Verwendungsstellen der Typen dar.

- Bildbaustein-Typ
  Sie erstellen ein Anzeige- und Bedienobjekt nach Ihren Bedürfnissen und legen es in der Projektbibliothek ab. Die Instanzen sind an den jeweiligen Bildbaustein-Typ gebunden. Wenn Sie eine Eigenschaft eines Bildbaustein-Typs ändern, wird die Eigenschaft zentral gespeichert und auch in allen Instanzen geändert. Im Bildbaustein-Typ legen Sie fest, welche Eigenschaften am Bildbaustein geändert werden können. Sie bearbeiten einen Bildbaustein-Typ im Editor "Bildbausteine".
- Bildbaustein
  Der Bildbaustein ist eine Instanz des Bildbaustein-Typs. Sie verwenden einen Bildbaustein in Bildern wie ein Anzeige- und Bedienobjekt. Die variablen Eigenschaften des Bildbaustein-Typs projektieren Sie an der Instanz. Sie weisen z. B. die Variablen Ihres Projekts dem Bildbaustein zu. Wenn Sie Eigenschaften am Bildbaustein projektieren, überschreiben Sie Eigenschaften des Bildbaustein-Typs. Die Änderungen am Bildbaustein werden an der Verwendungsstelle gespeichert und haben keine Auswirkungen auf den Bildbaustein-Typ. Um die variablen Eigenschaften am Bildbaustein wieder auf die Eigenschaften des Bildbaustein-Typs zurückzusetzen, wählen Sie im Kontextmenü "Instanz zurücksetzen".

Das folgende Bild stellt die Zusammenhänge zwischen Bildbaustein-Typ und Bildbaustein dar. Die Hintergrundfarbe des E/A-Felds ist im Bildbaustein-Typ so projektiert, dass jede Instanz die Eigenschaft ändern kann. Im Bildbaustein-Typ wird die Hintergrundfarbe blau verwendet. Im Bildbaustein verwenden Sie die Hintergrundfarbe gelb.

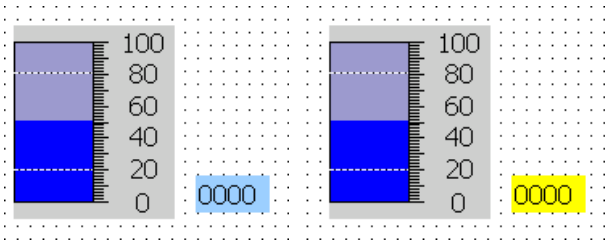

Wenn Sie die Hintergrundfarbe im Bildbaustein-Typ ändern, hat dies keine Auswirkungen auf den Bildbaustein, da die Eigenschaft "Hintergrundfarbe" instanzspezifisch belegt ist.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Editor Bildbausteine (Seite 275)

Grundlagen zur Dynamisierung von Bildbausteinen (Seite 303)

Beispiel: Bildbaustein projektieren (Seite 305)

Grundlagen zu Bildern (Seite 33)

1.7.1.2 Geräteabhängigkeit von Bildbausteinen

Geräteabhängigkeit von Bildbausteinen

Nicht alle Bediengeräte unterstützen jedes Anzeige- und Bedienobjekt. Die Bildobjekte, die beim jeweiligen Bediengerät nicht verfügbar sind, werden beim Verwenden des Bildbausteins nicht angezeigt.

Beim Erzeugen von Bildbausteinen werden geräteunabhängig alle Eigenschaften zur Projektierung angeboten. Bei der Verwendung eines Bildbausteins in einem Bild stehen jeweils nur die von dem projektierten Gerät unterstützen Eigenschaften zur Verfügung.

Folgende Geräte unterstützen Bildbausteine:

Runtimes

- WinCC Runtime Advanced
- WinCC Runtime Professional

Comfort Panels

- KTP 400
- KP 400
- TP 700
- KP 700
- TP 900

- KP 900
- TP 1200
- KP 1200
- TP 1500
- KP 1500
- TP 1900
- TP 2200

Mobile Panels

- Mobile Panel 277
- Mobile Panel 277 IWLAN V2
- Mobile Panel 277F IWLAN V2
- Mobile Panel 277F IWLAN (RFID Tag)
- Mobile Panel KTP 400F
- Mobile Panel KTP 700
- Mobile Panels KTP 700F
- Mobile Panel KTP 900
- Mobile Panel KTP 900F

1.7.1.3 Editor Bildbausteine

Öffnen

Bildbaustein-Typen erstellen und bearbeiten Sie in der Bibliotheksansicht.

Aufbau

Die Bibliotheksansicht für Bildbaustein-Typen besteht aus mehreren Editoren z. B. "Bilder" und "Variablen", die im Arbeitsbereich und im Konfigurationenbereich angeordnet sind.

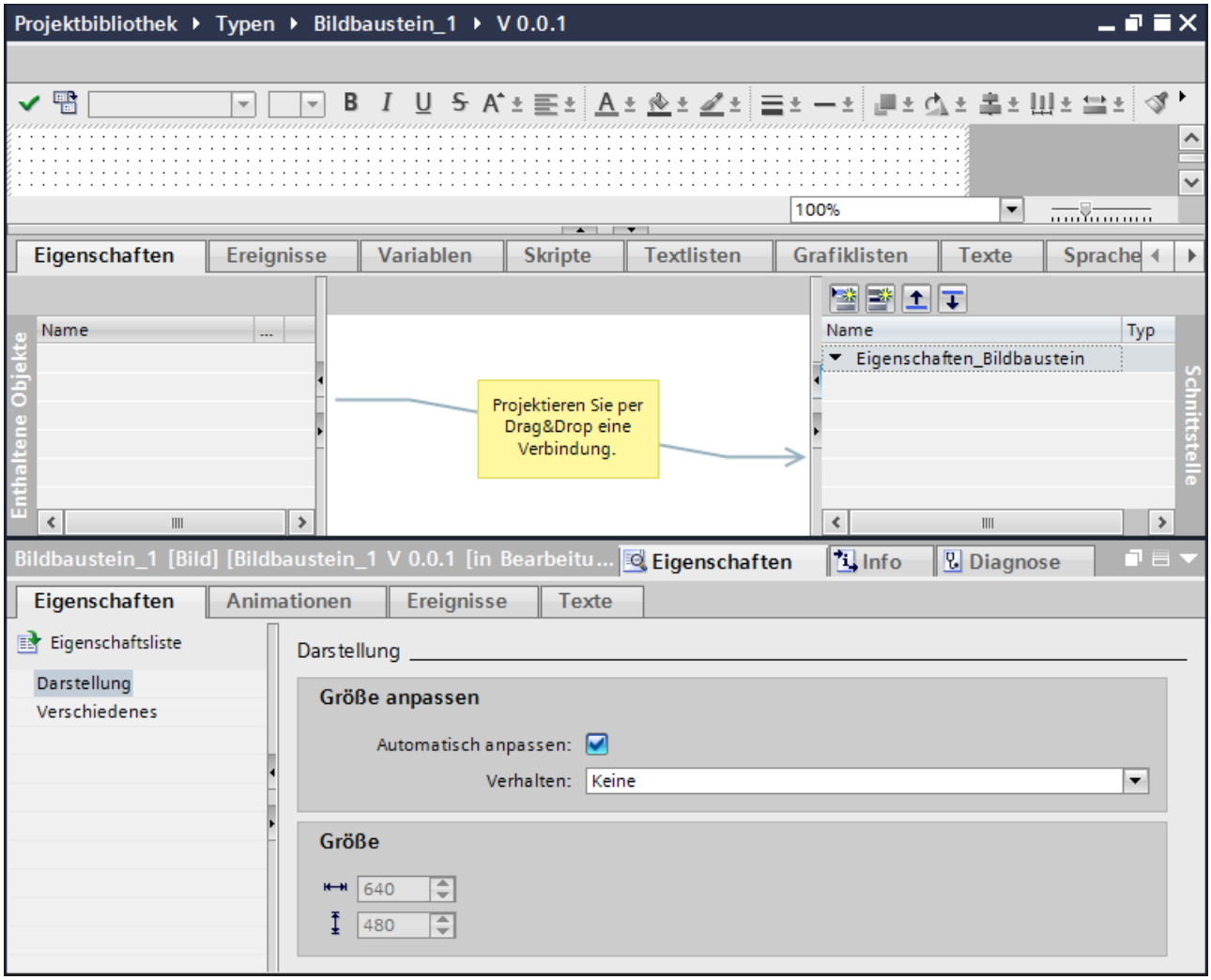

Arbeitsbereich

Im Arbeitsbereich platzieren Sie die im Bildbaustein-Typ enthaltenen Objekte, wie Sie es vom Editor "Bilder" gewohnt sind. Sie entfernen Objekte oder fügen aus der Task Card "Werkzeuge" neue Objekte ein.

Hinweis

Bildbausteine können nicht gedreht oder gespiegelt werden.

Konfigurationsbereich

- Eigenschaften
  Die statischen und dynamischen Eigenschaften sind wie folgt gekennzeichnet:
  - Das Symbol kennzeichnet eine dynamische Eigenschaft. Sie dynamisieren diese Eigenschaft mit Variablen oder Text- und Grafiklisten.
  - Das Symbol kennzeichnet eine statische Eigenschaft. Sie ändern nur die Werte einer statischen Eigenschaft.

  Hinweis

  Geräteabhängigkeit von statischen Eigenschaften

  Statische Eigenschaften sind nur für Panels und RT Advanced verfügbar.

  In der Registerkarte "Eigenschaften" legen Sie die Eigenschaften des Bildbaustein-Typs fest.
  Sie verwenden dazu die beiden folgenden Listen:
  - Die Liste "Enthaltene Objekte"
    Diese Liste enthält die Eigenschaften der im Bildbaustein-Typ enthaltenen Objekte. Diese Eigenschaften sind nur im Bildbaustein-Typ im Editor "Bildbausteine" projektierbar.
  - Die Liste "Schnittstelle" enthält die Eigenschaften und Variablen des Bildbausteins. Die Liste "Schnittstelle" besteht aus der vordefinierten Kategorie "Eigenschaften_Bildbaustein" und benutzerdefinierten Kategorien.
- Ereignisse
  In der Registerkarte "Ereignisse" legen Sie die Ereignisse des Bildbaustein-Typs fest. Sie verwenden dazu die beiden folgenden Listen:
  - Die Liste "Enthaltene Objekte" enthält die Ereignisse der im Bildbaustein-Typ enthaltenen Objekte.
  - Die Liste "Schnittstelle" enthält die Ereignisse des Bildbausteins.
    Sie erstellen per Drag&Drop Verbindungen zwischen den beiden Listen.
- Variablen
  In der Registerkarte "Variablen" legen Sie Bildbaustein-Variablen an. Die Variablen sind nur innerhalb des Bildbaustein-Typs verfügbar. Die Variablen des Bildbaustein-Typs verschalten Sie direkt an ein im Bildbaustein-Typ enthaltenes Objekt.

- Skripte
  In der Registerkarte "Skripte" projektieren Sie Skripte für den Bildbaustein-Typ. Im Skript rufen Sie Systemfunktionen auf oder programmieren neue Funktionen, z. B. zum Umrechnen von Werten. Die Skripte sind nur innerhalb des Bildbaustein-Typs verfügbar. Sie arbeiten mit dem VB-Script-Code.

  Hinweis

  Die Registerkarte "Skripte" steht für Runtime Professional nicht zur Verfügung.

- Textlisten
  In der Registerkarte "Textlisten" legen Sie Textlisten für den Bildbaustein-Typ an und bearbeiten sie. Diese Textlisten sind nur innerhalb des Bildbaustein-Typs verfügbar. Die Textlisten des Bildbaustein-Typs verschalten Sie direkt an ein im Bildbaustein-Typ enthaltenes Objekt, z. B. an ein symbolisches E/A-Feld.
- Grafiklisten
  In der Registerkarte "Grafiklisten" legen Sie bei Bedarf Grafiklisten für den Bildbaustein-Typ an. Diese Grafiklisten sind nur innerhalb des Bildbaustein-Typs verfügbar. Die Grafiklisten des Bildbaustein-Typs verschalten Sie direkt an ein im Bildbaustein-Typ enthaltenes Objekt, z. B. an ein Grafisches E/A-Feld.
- Anwendertexte
  Die Registerkarte "Anwendertexte" zeigt die enthaltenen Anwendertexte des geöffneten Bildbaustein-Typs z. B. Einträge aus Textlisten. Sie können die Übersetzungen für die einzelnen Projektsprachen direkt in der Registerkarte "Anwendertexte" eingeben.

Siehe auch

Grundlage zu Bildbausteinen (Seite 273)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2 Bildbaustein-Typen erzeugen und verwalten

1.7.2.1 Bildbaustein-Typ erzeugen

Einleitung

Bildbaustein-Typen sind Anzeige- und Bedienobjekte, die aus mehreren Objekten zusammengesetzt sind, wie z. B. Reglerbausteine.

Voraussetzung

Ein Bild mit mehreren Objekten ist geöffnet.

Vorgehen in der Projektansicht

1. Markieren Sie alle Objekte, die Sie für den Bildbaustein-Typ benötigen.
2. Wählen Sie im Kontextmenü der Mehrfachauswahl den Befehl "Erzeuge Bildbaustein-Typ". Ein Dialog wird geöffnet.
3. Geben Sie einen Namen für den Bildbaustein ein.
4. Bei Bedarf fügen Sie einen Kommentar hinzu oder ändern die Versionsnummer.

Ergebnis

Die Bibliotheksansicht wird geöffnet. Im Ordner "Typen" werden zwei Versionen des Typs angezeigt. Die Version 0.0.1 ist freigegeben und enthält die aktuell selektierten Objekte.

Die Version 0.0.2 ist im Zustand "in Bearbeitung". Sie bearbeiten den Bildbaustein-Typ in Version 0.0.2.

Vorgehen in der Bibliotheksansicht

1. Die Bibliotheksansicht ist geöffnet.
2. Wählen Sie im Kontextmenü der Projektbibliothek den Befehl "Neuen Typ hinzufügen". Ein Dialog wird geöffnet.
3. Wählen Sie die Runtime, für die der Bildbaustein-Typ zur Verfügung stehen soll. Ein leerer Bildbaustein-Typ wird angelegt.
4. Ziehen Sie per Drag & Drop Objekte aus der Task Card "Werkzeuge" in den Arbeitsbereich des Bildbaustein-Typs.

Ergebnis

Der neue Bildbaustein-Typ ist erzeugt und wird unter dem gewählten Namen in der Projektbibliothek angezeigt.

Der Bildbaustein-Typ erhält den Status "In Bearbeitung" und die Version 0.0.1.

Siehe auch

Grundlage zu Bildbausteinen (Seite 273)

Bildbaustein-Typ konfigurieren (Seite 280)

Kategorie und Eigenschaft eines Bildbaustein-Typs bearbeiten (Seite 286)

Variable im Bildbaustein-Typ projektieren (Seite 287)

Skripte im Bildbaustein-Typ projektieren (Seite 288)

Ereignis im Bildbaustein-Typ konfigurieren (Seite 291)

Textliste im Bildbaustein-Typ projektieren (Seite 292)

Grafikliste im Bildbaustein-Typ projektieren (Seite 293)

Texte im Bildbaustein-Typ direkt übersetzen (Seite 295)

Bildbaustein-Typ bearbeiten (Seite 298)

Größe eines Bildbausteins anpassen (Seite 299)

Instanz des Bildbaustein-Typs erstellen (Seite 301)

Bildbaustein von Bildbaustein-Typ lösen (Seite 302)

Kopiervorlagen und Typen (Seite 2020)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.2 Bildbaustein-Typ konfigurieren

Einleitung

Im Konfigurationsbereich des Editors "Bildbausteine" legen Sie in der Registerkarte "Eigenschaften" fest, welche Eigenschaften und Prozesswerte der enthaltenen Objekte im Bildbaustein projektierbar sind.

Hinweis

Damit projektierte Tooltiptexte von im Bildbaustein enthaltenen Objekten in Runtime angezeigt werden, fügen Sie die Eigenschaft "Tooltiptext" als Bildbausteineigenschaft hinzu.

Hinweis

Beim Erzeugen von Bildbausteinen werden geräteunabhängig alle Eigenschaften zur Projektierung angeboten. Bei der Verwendung eines Bilbausteins stehen jeweils nur die von dem projektierten Gerät unterstützen Eigenschaften zur Verfügung.

Hinweis

Die Verwendung von benutzerdefinierten PLC-Datentypen (UDTs) der PLCs S7-300 und S7-400 wird nicht unterstützt. Diese PLC-Datentypen können als Bildbaustein-Eigenschaft nicht ausgewählt werden.

Voraussetzung

- Der Bildbaustein ist erstellt und wird im Editor "Bildbausteine" angezeigt.
- Im Konfigurationsbereich ist die Registerkarte "Eigenschaften" geöffnet.

Vorgehen

1. Um eine neue Eigenschaft anzulegen, klicken Sie in der Liste "Schnittstelle" auf die Schaltfläche "Eigenschaft hinzufügen".
   Eine neue Eigenschaft wird in der Liste "Schnittstelle" angezeigt.
2. Um eine neue Kategorie anzulegen, klicken Sie in der Liste "Schnittstelle" auf die Schaltfläche "Kategorie hinzufügen".
   Eine neue Kategorie einschließlich einer neuen Eigenschaft wird in der Liste "Schnittstelle" angezeigt.

3. Klicken Sie auf den Namen der Eigenschaft und vergeben Sie einen Namen z. B. "Farbe".
4. Wählen Sie den Datentyp aus.

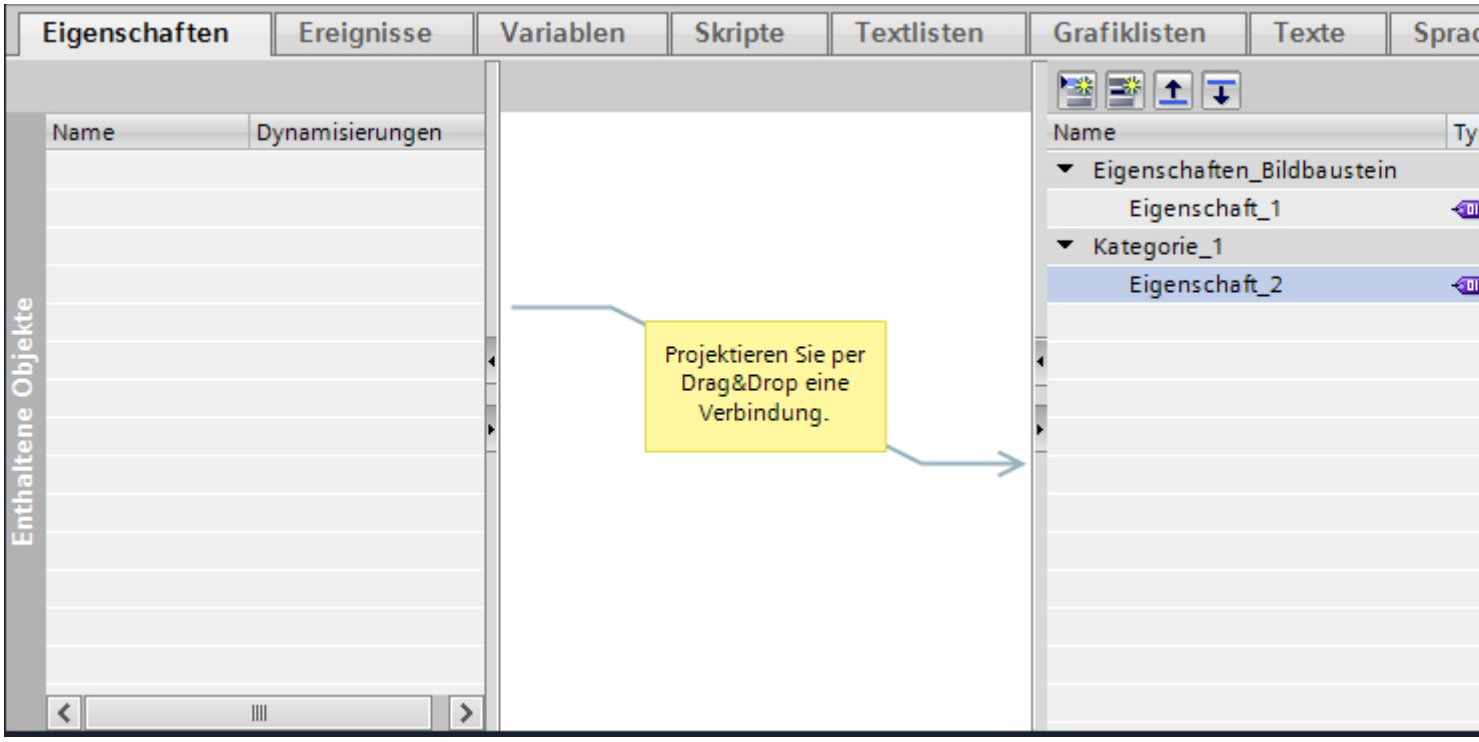

Hinweis

Eigenschaftsnamen der Bildbausteine bei RT Professional

Achten Sie bei der Eingabe von Eigenschaftsnamen der Bildbausteine auf folgende Regeln:

- Der Name muss mit einem Buchstaben beginnen.
- Der Name darf alphanumerische Zeichen und Unterstrich enthalten.
- Der Name darf maximal 255 Zeichen enthalten.
- Der Name darf keine UNICODE Zeichen (z. B. chinesische Schriftzeichen) enthalten.

Hinweis

Verwendung von Anwenderdatentypen (UDTs) in Bildbausteinen in RT Professional

Ist einem Element eines Bildbausteins als Datentyp der Anwenderdatentyp (UDT) zugewiesen, gilt die folgende Regel:

Die Länge des Namens einer Eigenschaft des Bildbausteins darf zusammen mit der Länge des Namens des untergeordneten Elements vom Datentyp "Anwenderdatentyp" (UDT) darf 126 Zeichen nicht überschreiten.

Zwischenergebnis

Sie haben eine neue Eigenschaft erstellt.

1. Klicken Sie in der Liste "Enthaltene Objekte" auf ein Objekt.
2. Wählen Sie eine Eigenschaft z. B. "Gestaltung > Farbe Hintergrund".
3. Ziehen Sie per Drag&Drop die Eigenschaft der Liste "Enthaltene Objekte" auf die gewünschte Eigenschaft in der Liste "Schnittstelle".
   Die Verbindung wird als farbige Linie angezeigt.
   Das System weist der Eigenschaft in der Liste "Schnittstelle" den gleichen Datentyp zu, den die Eigenschaft in der Liste "Enthaltene Objekte" besitzt.
   Um den Datentyp zu ändern, wählen Sie den Datentyp aus den Datentypen der verbunden Eigenschaft aus.
4. Wiederholen Sie Schritt 3, um weitere Eigenschaften der enthaltenen Objekte mit der Schnittstelle zu verbinden.

Hinweis

Die Eigenschaft aus der Liste "Enthaltene Objekte" muss mit dem Datentyp der bereits verbundenen Eigenschaft der Liste "Schnittstelle" übereinstimmen.

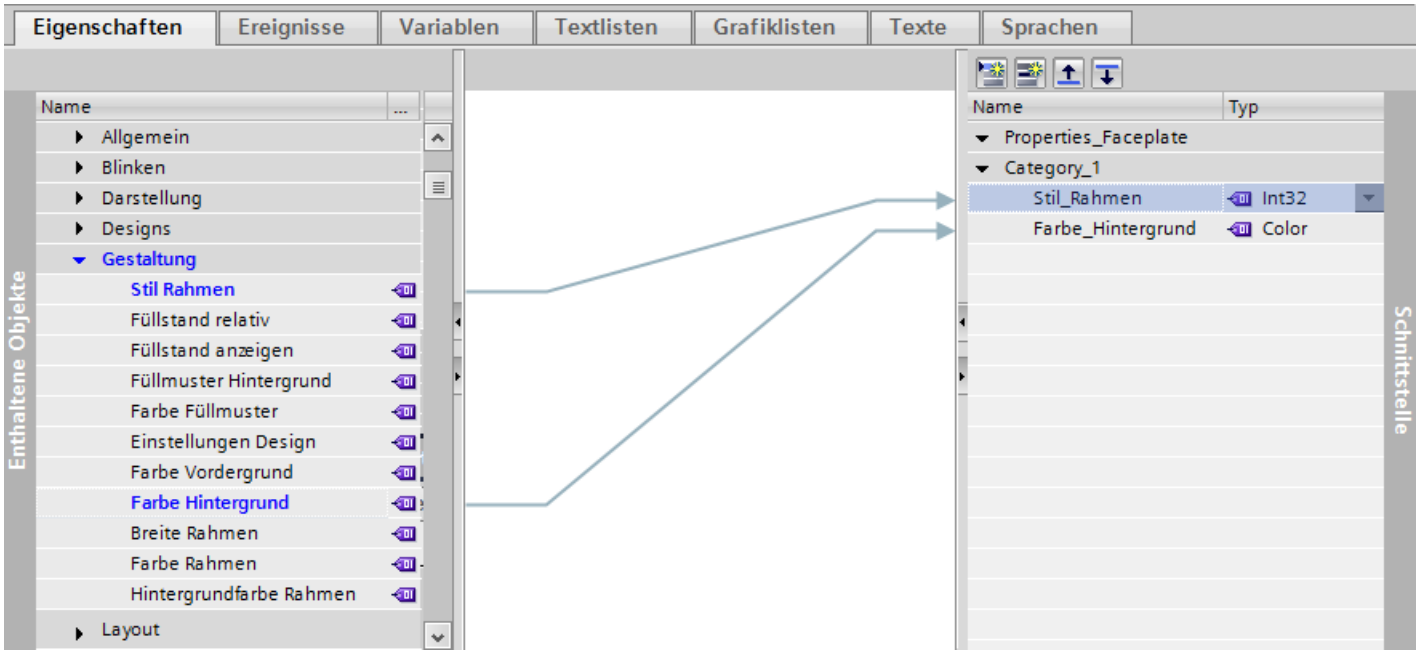

Verbindung löschen

1. Klicken Sie auf die Verbindung, die Sie löschen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Löschen" oder verwenden Sie die Taste <Entf>.
   Die Verbindung wird gelöscht.

Anwenderdatentypen mit Bildbausteinen verschalten

Ab V13 SP1 stehen Ihnen die HMI-Anwenderdatentypen und die PLC-Anwenderdatentypen als Eigenschaften der Bildbausteine zur Verfügung.

Die HMI-Anwenderdatentypen und die PLC-Anwenderdatentypen verschalten Sie mit den Bildbausteinen mithilfe von Schnittstelleneigenschaften, indem Sie die Bildbausteine mit den Strukturelementen der Schnittstelleneigenschaften verbinden.

Um die HMI-Anwenderdatentypen und die PLC-Anwenderdatentypen als Eigenschaften der Bildbausteine festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Anwenderdatentyp in der Auswahlliste unter "Typ".

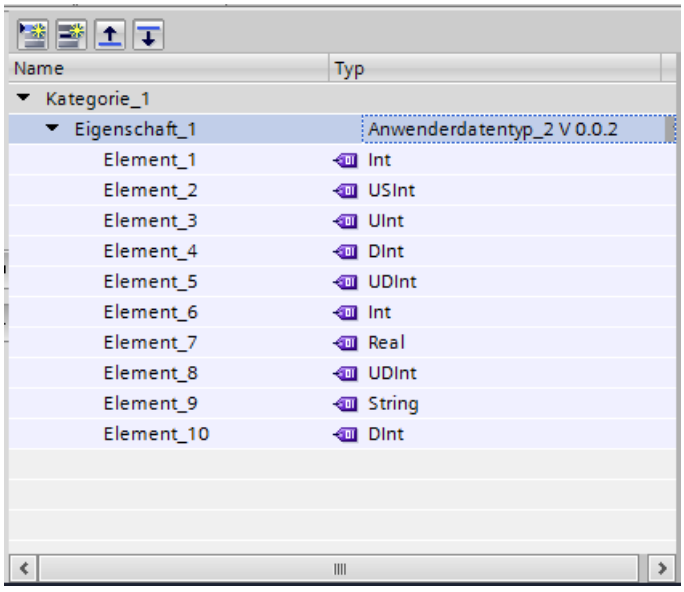

2. Im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" in der Auswahlliste "Variable" wählen Sie die Schnittstelleneigenschaft des Bildbausteins.

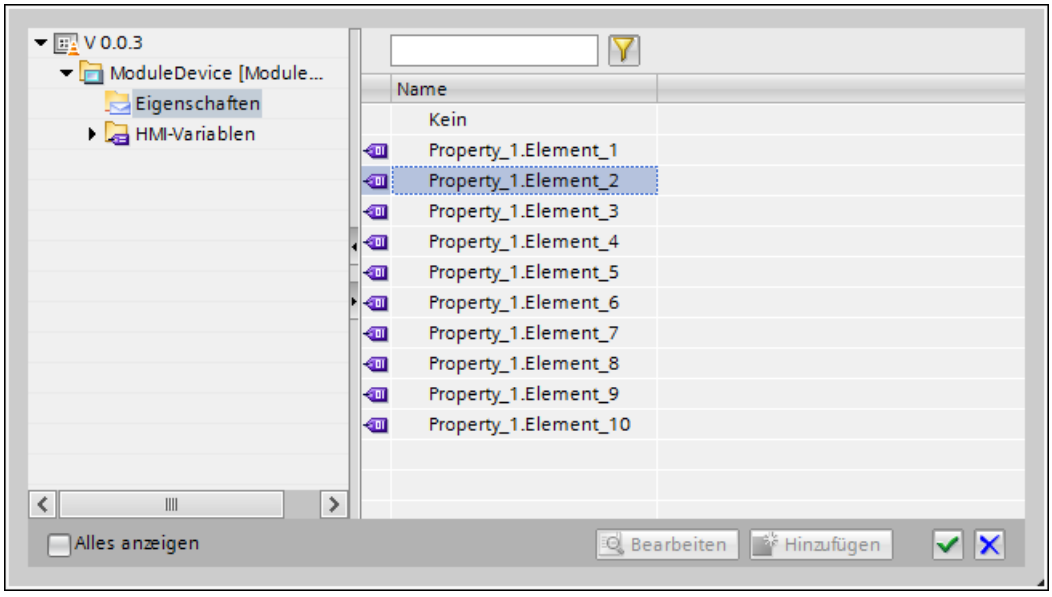

Der Bildbaustein wird mit dem Anwenderdatentyp verschaltet.

Ergebnis

Sie haben eine neue Eigenschaft bzw. eine neue Kategorie mit einer Eigenschaft angelegt. Sie haben die Eigenschaft der Schnittstelle mit einer Eigenschaft der enthaltenen Objekte verbunden. Wenn Sie eine Instanz des Bildbaustein-Typs in einem Bild verwenden, projektieren Sie die Eigenschaften des Bildbausteins, die Sie in der Liste "Schnittstelle" angelegt haben.

Hinweis

Geräteabhängigkeit von statischen Eigenschaften

Statische Eigenschaften stehen nur für Panels und RT Advanced zur Verfügung. Auf RT Professional arbeiten Sie mit dynamischen Eigenschaften.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

Anwenderdatentypen mit Objekten verknüpfen (Seite 142)

1.7.2.3 Kategorie und Eigenschaft eines Bildbaustein-Typs bearbeiten

Eigenschaft in der Liste "Schnittstelle" löschen

1. Wählen Sie im Kontextmenü "Typ bearbeiten". Der Editor "Bildbausteine" wird geöffnet.
2. Selektieren Sie in der Liste "Schnittstelle" die Eigenschaft, die Sie löschen wollen.
3. Wählen Sie im Kontextmenü "Löschen".

Die Eigenschaft wird aus der Liste "Schnittstelle" gelöscht. Bestehende Verbindungen zu den enthaltenen Objekten werden gelöscht. Alle verknüpften Bildbausteine verlieren ihre instanzspezifische Eigenschaft.

Kategorie in der Liste "Schnittstelle" löschen

1. Wählen Sie im Kontextmenü "Typ bearbeiten". Der Editor "Bildbausteine" wird geöffnet.
2. Selektieren Sie in der Liste "Schnittstelle" die Kategorie, die Sie löschen wollen.
3. Wählen Sie im Kontextmenü "Löschen".

Die Kategorie einschließlich der Eigenschaften wird aus der Liste "Schnittstelle" gelöscht. Bestehende Verbindungen zu den enthaltenen Objekten werden gelöscht. Alle verknüpften Bildbausteine verlieren ihre instanzspezifischen Eigenschaften.

Eigenschaft in eine andere Kategorie verschieben

1. Selektieren Sie in der Liste "Schnittstelle" eine Eigenschaft.
2. Ziehen Sie die Eigenschaft per "Drag&Drop" in eine neue Kategorie.

Sie haben die Eigenschaft in eine neue Kategorie verschoben.

Datentyp einer Eigenschaft ändern

1. Selektieren Sie in der Liste "Schnittstelle" eine Eigenschaft.
2. Klicken Sie in der zweiten Spalte auf die Auswahlliste.
3. Wählen Sie einen Datentyp.

Sie haben den Datentyp der Eigenschaft in der Liste "Schnittstelle" geändert.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.4 Variable im Bildbaustein-Typ projektieren

Einleitung

Im Bildbaustein-Typ verbinden Sie eine Variable direkt mit den Eigenschaften der im Bildbaustein-Typ enthaltenen Objekte. Mit Variablen im Bildbaustein-Typ definieren Sie z. B. zentral die Dynamisierung im Bildbaustein-Typ.

Die Variablen eines Bildbaustein-Typs haben eingeschränkte Funktionalitäten. Die Registerkarte "Variablen" des Konfigurationsbereichs ist aufgebaut wie der Editor "Variablen".

Hinweis

Die Verwendung von Zeichen "." und "@" in Namen von Variablen in Bildbaustein-Typen ist unzulässig. Verwenden Sie die Sonderzeichen nicht in den Variablennamen in Bildbausteinen.

Arrayelemente bei Bildbausteinen

In folgenden Fällen dürfen Sie kein Arrayelement bzw. keine Multiplex-Variable, die mit einem Array verschaltet ist, an eine Eigenschaft eines Bildobjekts projektieren:

- Die Eigenschaft ist als Parameter einer Systemfunktion in einem Bildbaustein projektiert.
- An Ereignisse der Eigenschaft in einem Bildbaustein sind Systemfunktionen oder Skripte projektiert.

Voraussetzungen

- Die Bibliotheksansicht ist geöffnet.
- Der Bildbaustein-Typ ist erstellt und im Zustand "in Bearbeitung".
- Die Registerkarte "Variablen" ist im Konfigurationsbereich geöffnet.

Vorgehen

1. Klicken Sie in der Tabelle "HMI-Variablen" auf "Hinzufügen". Eine neue Variable wird im Bildbaustein-Typ angelegt.
2. Bei Bedarf klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Sie projektieren z. B. das Ereignis "Wertänderung" an die Variable.
3. Markieren Sie im Arbeitsbereich das Objekt, dem Sie diese Variable zuordnen wollen, z. B. ein E/A-Feld.
4. Wählen Sie im Inspektorfenster des E/A-Feldes "Eigenschaften > Eigenschaften > Allgemein > Prozess > Variable" die Variable aus.

   Hinweis

   Innerhalb des Editors "Bildbausteine" werden in der Objektliste nur Variablen des Bildbaustein-Typs angezeigt.

Ergebnis

Sie haben eine Variable im Bildbaustein-Typ erstellt. Sie verwenden die Variable z. B. für Skripte innerhalb des Bildbausteins.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.5 Skripte im Bildbaustein-Typ projektieren

Einleitung

Im Konfigurationsbereich des Editors "Bildbausteine" erstellen Sie Skripte, die Sie nur innerhalb eines Bildbaustein-Typs verwenden. Innerhalb des Skripts können Sie nur auf Variablen des Bildbaustein-Typs oder Eigenschaften der enthaltenen Objekte verweisen. In der Instanz des Bildbaustein-Typs wird das Skript als Kopie verwendet.

Voraussetzung

- Die Bibliotheksansicht ist geöffnet.
- Der Bildbaustein-Typ ist erstellt und im Zustand "in Bearbeitung".
- Bildbaustein-Variablen oder dynamische Eigenschaften sind angelegt.

Vorgehen

1. Öffnen Sie im Konfigurationsbereich des Bildbausteins die Registerkarte "Skripte".
2. Doppelklicken Sie auf "Bildbaustein-Skripte > VB-Skript hinzufügen".
3. Schreiben Sie den Programmcode.
4. Um Variablen des Bildbaustein-Typs im Skript zu verwenden, drücken Sie die Tastenkombination <STRG+J>. Die Objektliste wird geöffnet.
5. Klicken Sie auf "HMI-Variablen".
   In der Objektliste werden alle Variablen des Bildbaustein-Typs angezeigt.
6. Wählen Sie eine Variable und bestätigen Sie die Auswahl.
7. Um im Skript Eigenschaften des Bildbaustein-Typs zu verwenden, drücken Sie die Tastenkombination <STRG+J>. Die Objektliste wird geöffnet.
8. Klicken Sie auf "Eigenschaften".
   In der Objektliste werden alle dynamischen Eigenschaften der Liste "Schnittstelle" des Bildbaustein-Typs angezeigt.

Ergebnis

Sie haben ein Skript im Bildbaustein-Typ erstellt. Die Instanz des Bildbaustein-Typs verwendet eine Kopie des Skripts. Je nach Projektierung wird das Skript in Runtime ausgeführt.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.6 Ereignis im Bildbaustein-Typ konfigurieren

Einleitung

Im Konfigurationsbereich des Editors "Bildbausteine" legen Sie in der Registerkarte "Ereignisse" fest, welche Ereignisse der enthaltenen Objekte am Bildbaustein projektierbar sind. An die Ereignisse des Bildbausteins projektieren Sie im Editor "Bilder" eine Funktionsliste.

Voraussetzung

- Die Bibliotheksansicht ist geöffnet.
- Der Bildbaustein-Typ ist erstellt und im Zustand "in Bearbeitung".
- Die Registerkarte "Ereignisse" des Konfigurationsbereichs ist geöffnet.

Vorgehen

1. Klicken Sie in der Liste "Enthaltene Objekte" auf ein Objekt.
2. Wählen Sie ein Ereignis z.B. "Aktivieren".
3. Ziehen Sie das Ereignis aus der Liste "Enthaltene Objekte" in eine Kategorie in der Liste "Schnittstelle".
   Das neue Ereignis wird in der Liste "Schnittstelle" angezeigt.
   Die Verbindung wird als farbige Linie dargestellt.

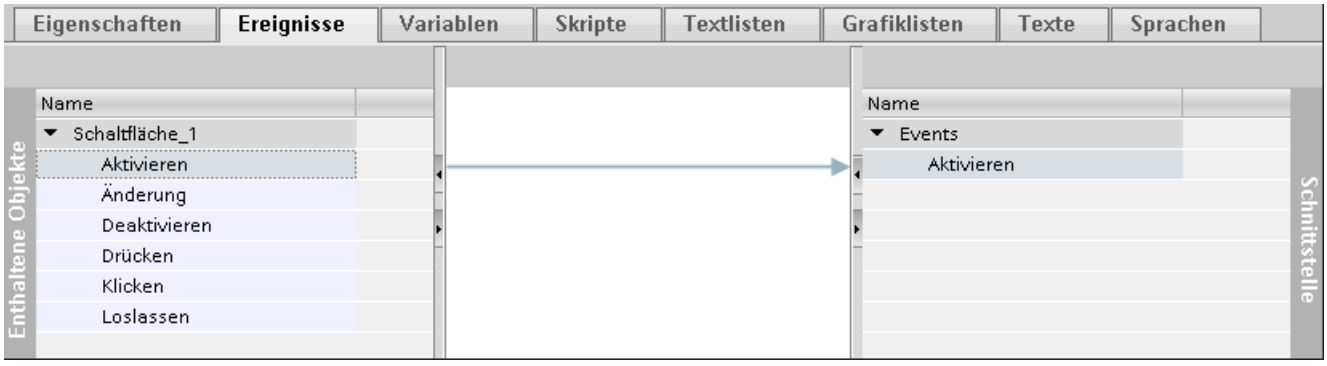

4. Doppelklicken Sie auf das Ereignis und ändern Sie bei Bedarf den Namen des Ereignisses.

Ergebnis

Sie haben ein Ereignis im Bildbaustein-Typ angelegt. Wenn Sie eine Instanz des Bildbaustein-Typs verwenden, werden Ihnen im Inspektorfenster nur die projektierten Ereignisse angeboten. Am Bildbaustein projektieren Sie an das Ereignis eine Funktionsliste.

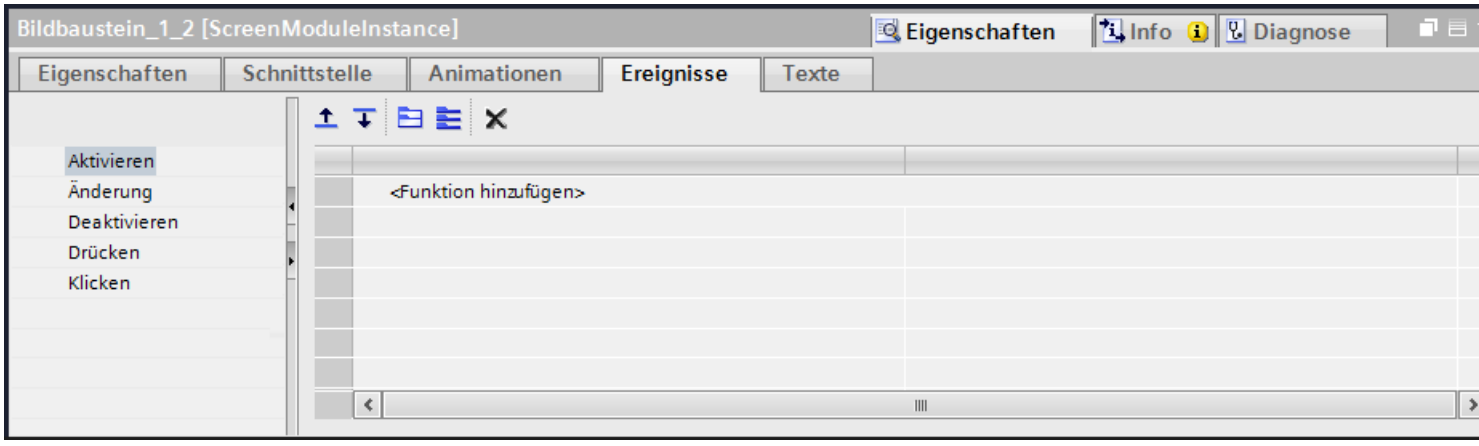

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.7 Ereignis im Bildbaustein-Typ konfigurieren

Einleitung

Im Konfigurationsbereich des Editors "Bildbausteine" legen Sie in der Registerkarte "Ereignisse" fest, welche Ereignisse der enthaltenen Objekte am Bildbaustein projektierbar sind. An die Ereignisse des Bildbausteins projektieren Sie im Editor "Bilder" eine Funktionsliste.

Voraussetzung

- Der Bildbaustein-Typ ist erstellt und wird im Editor "Bildbausteine" angezeigt.
- Die Registerkarte "Ereignisse" des Konfigurationsbereichs ist geöffnet.

Vorgehen

1. Klicken Sie in der Liste "Enthalten Objekte" auf ein Objekt.
2. Wählen Sie ein Ereignis z.B. "Aktivieren".
3. Ziehen Sie das Ereignis aus der Liste "Enthaltene Objekte" in eine Kategorie in der Liste "Schnittstelle".
   Das neue Ereignis wird in der Liste "Schnittstelle" angezeigt.
   Die Verbindung wird als farbige Linie dargestellt.

4. Doppelklicken Sie auf das Ereignis und ändern Sie bei Bedarf den Namen des Ereignisses.

Ergebnis

Sie haben ein Ereignis im Bildbaustein-Typ angelegt. Wenn Sie eine Instanz des Bildbaustein-Typs verwenden, werden Ihnen im Inspektorfenster alle Standardeinstellungen für Ereignisse angeboten. Am Bildbaustein projektieren Sie an das Ereignis eine Funktionsliste oder ein VB-Skript bzw. C-Skript.

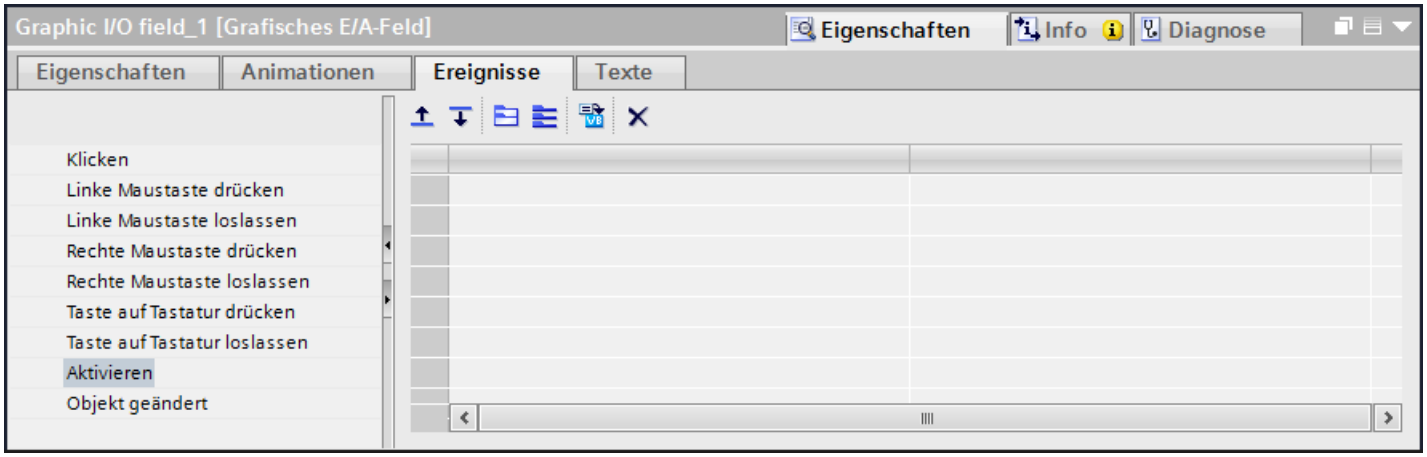

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.8 Textliste im Bildbaustein-Typ projektieren

Einleitung

Im Bildbaustein-Typ verbinden Sie eine Textliste mit einem im Bildbaustein-Typ enthaltenen Objekt. In einer Textliste sind Texte den Werten einer Variablen zugeordnet. Sie verwenden die Textliste z. B. für die Anzeige einer Auswahlliste in einem Symbolischen E/A-Feld, das im Bildbaustein-Typ enthalten ist. Die Anbindung der Textliste an eine Variable des Bildbaustein-Typs projektieren Sie an dem Objekt, das die Textliste verwendet.

Voraussetzungen

- Die Bibliotheksansicht ist geöffnet.
- Der Bildbaustein-Typ ist erstellt und ist im Zustand "in Bearbeitung".
- Die Registerkarte "Textlisten" ist im Konfigurationsbereich geöffnet.
- Eine Variable ist im Bildbaustein-Typ angelegt.

Vorgehen

1. Klicken Sie in der Tabelle "Textlisten" auf "Hinzufügen". Eine neue Textliste wird im Bildbaustein-Typ angelegt.
2. Geben Sie der Textliste einen aussagekräftigen Namen.
3. Wählen Sie unter "Auswahl" den Textlistentyp z. B. "Wert/Bereich".
4. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".
5. Legen Sie die Werte bzw. Wertebereiche der Textliste fest.
6. Geben Sie für jeden Wertebereich einen Text ein, der in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.
7. Selektieren Sie im Arbeitsbereich des Bildbaustein-Typs ein Objekt, z. B. ein Symbolisches E/A-Feld.
8. Aktivieren Sie im Inspektorfenster des Objekts "Eigenschaften > Eigenschaften > Allgemein > Beschriftung > Textliste".
9. Wählen Sie aus der Auswahlliste die Textliste.
10. Wählen Sie unter "Prozess > Variable" eine Variable des Bildbaustein-Typs.

Ergebnis

Sie haben eine Textliste im Bildbaustein-Typ erstellt. Die Textliste ist mit einem im Bildbaustein-Typ enthaltenen Objekt verbunden. Sie haben eine Variable des Bildbaustein-Typs an das Objekt projektiert.

Siehe auch

Grundlagen zu Textlisten (Seite 180)

Textliste erstellen (Seite 182)

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.9 Grafikliste im Bildbaustein-Typ projektieren

Einleitung

In einer Grafikliste sind bestimmte Grafiken den möglichen Werten einer Variablen zugeordnet. Im Bildbaustein-Typ verbinden Sie eine Grafikliste mit einem im Bildbaustein-Typ enthaltenen Objekt. In einer Grafikliste sind Grafiken den Werten einer Variablen zugeordnet. Sie verwenden die Grafikliste z. B. für die Anzeige einer Auswahlliste in einem Grafischen E/A-Feld, das im Bildbaustein-Typ enthalten ist. Die Anbindung der Grafikliste an eine Variable des Bildbaustein-Typs projektieren Sie an dem Objekt, das die Grafikliste verwendet.

Voraussetzungen

- Die Bibliotheksansicht ist geöffnet.
- Der Bildbaustein-Typ ist erstellt und ist im Zustand "in Bearbeitung".
- Die Registerkarte "Grafiklisten" ist im Konfigurationsbereich geöffnet.
- Eine Variable ist im Bildbaustein-Typ angelegt.

Vorgehen

1. Klicken Sie in der Tabelle "Grafiklisten" auf "Hinzufügen". Eine neue Grafikliste wird im Bildbaustein-Typ angelegt.
2. Geben Sie der Grafikliste einen aussagekräftigen Namen.
3. Wählen Sie unter "Auswahl" einen Grafiklistentyp, z. B. " Bereich (... - ...)".
4. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen". Ein neuer Listeneintrag wird erstellt.
5. Legen Sie die Werte bzw. Wertebereiche der Grafikliste fest.
6. Wählen Sie für jeden Wert bzw. Wertebereich eine Grafik, die in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.
7. Selektieren Sie im Arbeitsbereich des Bildbaustein-Typs ein Objekt, z. B. ein Grafisches E/A-Feld.
8. Aktivieren Sie im Inspektorfenster des Objekts "Eigenschaften > Eigenschaften > Allgemein > Inhalt > Grafikliste".
9. Wählen Sie aus der Auswahlliste die Grafikliste.
10. Wählen Sie "Prozess > Variable" eine Variable des Bildbaustein-Typs.

Ergebnis

Sie haben eine Grafikliste im Bildbaustein-Typ erstellt. Die Grafikliste ist mit einem im Bildbaustein-Typ enthaltenen Objekt verbunden. Sie haben eine Variable des Bildbaustein-Typs an das Objekt projektiert.

Siehe auch

Grundlagen zu Grafiklisten (Seite 194)

Grafikliste erstellen (Seite 197)

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.10 Texte im Bildbaustein-Typ direkt übersetzen

Texte übersetzen

Wenn Sie in Ihrem Projekt mehrere Sprachen verwenden, können Sie einzelne Texte direkt übersetzen. Sobald Sie die Sprache der Softwareoberfläche ändern, stehen die übersetzten Texte in der jeweiligen Sprache zur Verfügung.

Hinweis

Beim Erstellen eines Bildbausteins werden in der Registerkarte "Texte" Meldeklassentexte und Meldetexte automatisch angelegt. Diese Texte werden für das Meldesystem von WinCC benötigt.

Voraussetzungen

- Ein Projekt ist geöffnet.
- Der Editor "Bildbausteine" ist geöffnet.
- Sie haben einen Bildbaustein-Typen erstellt.
- Der Bildbaustein-Typ enthält mindestens ein Objekt mit Text, z. B. ein Textfeld.

Vorgehen

Um einzelne Texte zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Aufgaben".
2. Klicken Sie unter "Sprachen & Ressourcen > Editiersprache" auf die Schaltfläche . Der Editor "Projektsprachen" wird geöffnet.
3. Aktivieren Sie mindestens zwei Projektsprachen.
4. Öffnen Sie im Editor "Bildbausteine" die Registerkarte "Texte".
   Im Arbeitsbereich erscheint eine Liste mit den Texten im Projekt. Für jede Projektsprache existiert eine eigene Spalte.

5. Um identische Texte zu gruppieren klicken Sie in der Funktionsleiste auf die Schaltfläche .
6. Um Texte mit fehlenden Übersetzungen auszublenden, klicken Sie in der Funktionsleiste auf die Schaltfläche .
7. Klicken Sie in eine leere Spalte und geben Sie die Übersetzung ein.

Ergebnis

Sie haben einzelne Texte in der Registerkarte "Texte" des Bildbaustein-Typs übersetzt. In Runtime werden die Texte in der jeweiligen Runtime-Sprache angezeigt.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.11 Texte im Bildbaustein-Typ exportieren und importieren

Texte exportieren und importieren

Sie haben die Möglichkeit, die Texte der Bildbausteine zur Übersetzung zu exportieren und anschließend wieder zu importieren. Beim Export wird eine Open Office XML-Datei erstellt, die mit Microsoft Excel und anderen Tabellenkalkulationsprogrammen bearbeitet werden kann. Die verfügbaren Sprachen für den Export und Import legen Sie in der Registerkarte "Sprachen" fest. Sie exportieren und importieren die Texte eines Bildbaustein-Typs mithilfe der Schaltflächen "Projekttexte exportieren" und "Projekttexte importieren" der Funktionsleiste.

Alternativ können Sie die Texte der Bildbausteine mithilfe der Befehle "Texte exportieren" und "Texte importieren" in der Task Card "Bibliothek" exportieren und importieren. In diesem Fall werden für den Export und Import die für dieses Projekt definierten Projektsprachen verwendet.

Voraussetzungen

- Ein Projekt ist geöffnet.
- Der Editor "Bildbausteine" ist geöffnet.
- Sie haben einen Bildbaustein-Typen erstellt.
- Der Bildbaustein-Typ enthält mindestens ein Objekt mit Text, z. B. ein Textfeld.

Texte exportieren

Um die Texte zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Bildbaustein-Typ in der Projektbibliothek.
2. Öffnen Sie im Editor "Bildbausteine" die Registerkarte "Texte".
   Im Arbeitsbereich erscheint eine Liste mit den Texten im Projekt. Für jede Projektsprache existiert eine eigene Spalte.

3. Um die Texte in eine .xslx Datei zu exportieren, klicken Sie auf die Schaltfläche "Projekttexte exportieren" .
   Der Dialog "Exportieren" wird geöffnet.
4. Wählen Sie im Dialog die Quellsprache und die Zielsprache.
5. Wählen Sie für den Export entweder den im aktuellen Fenster ausgewählten Inhalt oder wählen Sie die Anwendertexte über Kategorien aus.
6. Geben Sie den Namen und den Pfad für die Exportdatei ein.
7. Klicken Sie auf "Exportieren".
   Nachdem der Export erfolgreich abgeschlossen wurde, wird unter dem angegebenen Pfad die Exportdatei abgelegt.

Texte importieren

Um die Texte nach der Bearbeitung oder Übersetzung wieder in das TIA Portal zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Bildbaustein-Typ in der Projektbibliothek.
2. Öffnen Sie im Editor "Bildbausteine" die Registerkarte "Texte".
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Projekttexte importieren" .
   Der Dialog "Importieren" wird geöffnet.

4. Wählen Sie im Feld "Zu importierende Datei auswählen" den Pfad und den Namen der Importdatei aus.
   Wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten, aktivieren Sie das Optionskästchen "Quellsprache importieren".
5. Klicken Sie auf "Importieren".

Ergebnis

Sie haben die Texte eines Bildbausteins für die Übersetzung exportiert und nach der Übersetzung wieder in das TIA Portal importiert.

1.7.2.12 Bildbaustein-Typ bearbeiten

Einleitung

Alle Bildbaustein-Typen in der Projektbibliothek haben einen Status. Sie erstellen einen Bildbaustein-Typ im Status "in Bearbeitung". In diesem Status bearbeiten Sie den Bildbaustein-Typen beliebig. Wenn Sie die Bearbeitung abgeschlossen haben, geben Sie den Bildbaustein-Typ frei.

Voraussetzung

- Ein Bildbaustein-Typ ist angelegt.
- Der Bildbaustein-Typ hat die Version 0.0.1. und ist im Zustand "in Bearbeitung".
- Die Task Card "Bibliothek" oder die Bibliotheksansicht ist geöffnet.

Bildbaustein-Typ freigeben

1. Selektieren Sie in der Projektbibliothek die Version 0.01 des Bildbaustein-Typs.
2. Wählen Sie im Kontextmenü "Version freigeben".

Sie haben die Version 0.0.1 des Bildbaustein-Typs freigegeben.

Bildbaustein-Typ bearbeiten

1. Selektieren Sie in der Projektbibliothek z.B. die freigegebene Version 0.0.1 eines Bildbaustein-Typs.
2. Wählen Sie im Kontextmenü "Bildbaustein-Typ bearbeiten".

Die Bibliotheksansicht wird geöffnet. Die neue Version 0.0.2 des Bildbaustein-Typen wurde erzeugt.

Der Bildbaustein-Typ hat den Status "in Bearbeitung".

Letzte Version des Bildbaustein-Typs wiederherstellen

Die letzte freigegebene Version des Bildbaustein-Typs ist Version 0.0.2.

Sie bearbeiten den Bildbaustein-Typ. Eine neue Version wird erzeugt 0.0.3 und erhält den Status "in Bearbeitung".

1. Selektieren Sie in der Projektbibliothek den Bildbaustein-Typ.
2. Wählen Sie im Kontextmenü "Änderungen verwerfen und Typ wiederherstellen".

Alle Änderungen des Bildbaustein-Typs seit der letzten Freigabe werden verworfen. Der Bildbaustein-Typ ist wieder freigegeben und hat die Version 0.0.2.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

1.7.2.13 Größe eines Bildbausteins anpassen

Einleitung

Im Bildbaustein-Typ legen Sie fest, wie sich der Bildbaustein verhält, wenn seine Größe verändert wird.

Voraussetzungen

- Ein Bildbaustein-Typ ist angelegt.
- Der Editor "Bildbausteine" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Verhalten bei der Größenanpassung festlegen

1. Aktivieren Sie "Eigenschaften > Eigenschaften > Darstellung > Größe anpassen > Automatisch anpassen".
2. Wählen Sie z. B. die Einstellung "Festes Seitenverhältnis".
   Damit wird bei der Größenanpassung das Seitenverhältnis beibehalten.
3. Klicken Sie in der Task Card "Bibliotheken" auf den Bildbaustein.
4. Wählen Sie im Kontextmenü "Version freigeben".

Ergebnis

Sie haben festgelegt, wie sich der Bildbaustein verhält, wenn seine Größe in einem Bild angepasst wird. Sie können das Verhalten für einen bestimmten Bildbaustein ändern.

Selektieren Sie einen Bildbaustein im Bild. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Verhalten" die gewünschte Einstellung.

1.7.2.14 Größe eines Bildbausteins anpassen

Einleitung

Im Bildbaustein-Typ legen Sie fest, wie sich der Bildbaustein verhält, wenn seine Größe verändert wird.

Voraussetzungen

- Ein Bildbaustein-Typ ist angelegt.
- Der Editor "Bildbausteine" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Verhalten bei der Größenanpassung festlegen

1. Aktivieren Sie "Eigenschaften > Eigenschaften > Darstellung > Größe anpassen > Automatisch anpassen".
2. Wählen Sie für "Verhalten beim Anpassen" eine der folgenden Einstellungen:
   - Feste Größe
     Legt fest, dass die Größe des Bildbausteins unveränderlich ist.
   - Proportional
     Der Bildbaustein wird proportional skaliert. Das objektumfassende Rechteck wird nicht skaliert.
   - Keine
     Keine Einschränkung bei der Größenanpassung des Bildbausteins.
   - Originalgröße
     Legt fest, dass die Größe des Bildbausteins unveränderlich ist, jedoch die Größe des objektumfassenden Rechtecks angepasst werden kann.
   - Seitenverhältnis beibehalten
     Die Größe des Bildbausteins ist veränderbar. Bei der Größenanpassung wird das Seitenverhältnis beibehalten.
3. Klicken Sie in der Task Card "Bibliotheken" auf den Bildbaustein.
4. Wählen Sie im Kontextmenü "Bildbaustein-Typ freigeben".

Ergebnis

Sie haben festgelegt, wie sich der Bildbaustein verhält, wenn seine Größe in einem Bild angepasst wird. Sie können das Verhalten für einen bestimmten Bildbaustein ändern.

Selektieren Sie einen Bildbaustein im Bild. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Verhalten" die gewünschte Einstellung.

1.7.2.15 Instanz des Bildbaustein-Typs erstellen

Einleitung

Der Bildbaustein-Typ ist in der Projektbibliothek abgelegt. Wenn Sie den Bildbaustein-Typ in einem Bild verwenden, erstellen Sie eine Instanz des Bildbaustein-Typen.

Hinweis

Beachten Sie, dass ein Bildbaustein immer für eine bestimmte Klasse von Bediengeräten konfiguriert ist. So können Sie z. B. einen Bildbaustein-Typ der für "RT Professional" konfiguriert ist nicht in einem Bild eines Bediengeräts "RT Advanced" verwenden.

Hinweis

Die Anzahl der Instanzen von Bildbausteinen in einem Bild ist nicht beschränkt. Bitte beachten Sie dabei, dass die Performance beim Öffnen oder Aktualisieren eines Bilds durch die Anzahl der Bildbaustein-Instanzen oder die Verwendung von Skripten in den Bildbaustein-Instanzen beeinflusst wird.

Hinweis

Wenn Sie eine Bildbaustein-Instanz aus einem Projekt in ein anderes Projekt kopieren möchten, kopieren Sie zuerst den Bildbaustein-Typ in das Ziel-Projekt und danach die Bildbaustein-Instanz.

Voraussetzung

- Ein Bild ist geöffnet.
- Die Task Card "Bibliotheken" ist geöffnet.
- Die Projektbibliothek ist geöffnet und enthält mindestens einen Bildbaustein-Typ.

  Hinweis

  Bildbaustein-Typen hinterlegen Sie auch in globalen Bibliotheken. Wenn Sie einen Bildbaustein-Typ aus einer globalen Bibliothek in ein Bild einfügen, wird eine Kopie dieses Bildbaustein-Typs in der Projektbibliothek hinterlegt.

- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie den gewünschten Bildbaustein-Typ per Drag&Drop aus einer Bibliothek in das Bild.

Ergebnis

Sie haben eine Instanz des Bilbaustein-Typen erstellt. Die Instanz erhält die Versionsnummer des zuletzt freigegebenen Typen.

Konfigurieren Sie die Eigenschaften des Bildbaustein im Inspektorfenster nach Bedarf. Um den Bildbaustein zu dynamisieren, öffnen Sie die Schnittstelle im Inspektorfenster.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.2.16 Bildbaustein von Bildbaustein-Typ lösen

Einleitung

Um bestimmte Instanzen von der Aktualisierung des Bildbaustein-Typs auszunehmen, heben Sie die Verwendung auf.

Voraussetzung

- Ein Bildbaustein-Typ ist angelegt.
- Der Bildbaustein ist mindestens in einem Bild verwendet.

Bildbaustein vom Bildbaustein-Typ lösen

1. Selektieren Sie einen Bildbaustein im Bild.
2. Wählen Sie im Kontextmenü "Bildbaustein von Bildbaustein-Typ lösen".

Ergebnis

Der Bildbaustein wird in eine vom Bildbaustein-Typ unabhängige Gruppe umgewandelt. Änderungen des Bildbaustein-Typs haben keine Auswirkung auf diese Gruppe.

Siehe auch

Bildbaustein-Typ erzeugen (Seite 278)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.3 Bildbausteine dynamisieren

1.7.3.1 Grundlagen zur Dynamisierung von Bildbausteinen

Verwendung

Ereignisse und Eigenschaften von Bildbausteinen dynamisieren Sie auf zwei Arten.

- Bildbaustein dynamisieren
  Sie projektieren am Bildbaustein die Ereignisse oder dynamischen Eigenschaften individuell für die Verwendungsstelle. Dazu legen Sie im Editor "Bildbausteine" fest, dass diese Eigenschaften und Ereignisse am Bildbaustein projektierbar sind. Um die Bildbausteine im Editor "Bilder" zu dynamisieren, verwenden Sie Skripte und Variablen, die im Projekt angelegt sind.
- Bildbaustein-Typ dynamisieren
  Sie dynamisieren die Objekte, die im Bildbaustein-Typ enthalten sind, im Editor "Bildbausteine". Sie projektieren die einzelnen Objekte wie im Editor "Bilder". Um Eigenschaften und Ereignisse zu dynamisieren, stehen im Editor "Bildbausteine" Variablen und Skripte zur Verfügung. Auf die Variablen und Skripte des Projekts haben Sie innerhalb des Bildbausteins keinen Zugriff.
  Jeder mit dem Bildbaustein-Typ erzeugte Bildbaustein besitzt die gleiche, vorkonfigurierte Dynamisierung. Sie bearbeiten diese Dynamisierung nur im Editor "Bildbaustein".

Hinweis

Dynamisierung von Instanzen eines Bildbaustein-Typs in einer Gruppe

Sie verwenden die Instanz eines Bildbaustein-Typs in einer Objektgruppe. Die Eigenschaften der Instanz werden auch als Eigenschaften der Gruppe angezeigt. Jede Dynamisierung mit Variablen, Skripten oder Animationen der Gruppe wird in Runtime nicht angezeigt.

Animation

WinCC stellt in der Task Card "Animationen" vorkonfigurierte Dynamisierungen zur Verfügung. Sie verwenden Animationen, um Bildbausteine und Bildbaustein-Typen zu dynamisieren.

Siehe auch

Bildbaustein dynamisieren (Seite 304)

Beispiel: Bildbaustein projektieren (Seite 305)

Grundlage zu Bildbausteinen (Seite 273)

1.7.3.2 Bildbaustein dynamisieren

Einleitung

Sie dynamisieren einen Bildbaustein genauso, wie Sie ein Objekt aus der Task Card "Werkzeuge" dynamisieren.

Die dynamischen Eigenschaften des Bildbausteins verbinden Sie im Editor "Bilder" mit einer Variablen oder einem Skript, welches die Eigenschaft in Runtime mit Werten versorgt.

Für den Bildbaustein verwenden Sie die Variablen und Skripte, die Sie im Projekt angelegt haben.

Voraussetzung

- Ein Bildbaustein ist im Bild eingefügt.

Vorgehen

1. Markieren Sie den Bildbaustein.
2. Öffnen Sie im Inspektorfenster "Eigenschaften > Animation".
   Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Horizontale Bewegung" und klicken Sie auf die Schaltfläche .
   Die Parameter der Animation werden angezeigt.
   Im Arbeitsbereich wird eine transparente Kopie des Objekts eingeblendet, die durch einen Pfeil mit dem Originalobjekt verbunden ist.
4. Wählen Sie eine Variable zur Bewegungssteuerung aus.
5. Schieben Sie die Objektkopie an die gewünschte Zielposition. Im Inspektorfenster werden die Pixelwerte der Endposition automatisch eingetragen.
6. Passen Sie bei Bedarf den Wertebereich der Variablen an.

Ergebnis

In Runtime bewegt sich das Objekt bei jeder Wertänderung der Variablen, die zur Steuerung der Bewegung dient. Die Bewegungsrichtung entspricht der projektierten Bewegungsart "horizontal".

Siehe auch

Grundlagen zur Dynamisierung von Bildbausteinen (Seite 303)

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.4 Beispiel: Bildbaustein erstellen und dynamisieren

1.7.4.1 Beispiel: Bildbaustein projektieren

Einleitung

Im folgenden Beispiel erstellen Sie einen Bildbaustein, in dem Sie den Wert eines Längenmessgeräts von Kilometer in Meter umrechnen. Die Länge in Meter wird in einem Ausgabefeld angezeigt.

Handlungsübersicht

Das Beispiel gliedert sich in folgende Arbeitsschritte:

1. Bildbaustein-Typ erzeugen
2. Bildbaustein-Typ konfigurieren
3. Skript im Bildbaustein-Typ erstellen
4. Skript mit den enthaltenen Objekten des Bildbaustein-Typs verbinden
5. Bildbaustein erstellen und mit einer Variablen verbinden

Siehe auch

Grundlage zu Bildbausteinen (Seite 273)
Beispiel: Bildbaustein-Typ erzeugen (Seite 305)
Beispiel: Bildbaustein-Typ konfigurieren (Seite 306)
Beispiel: Skript im Bildbaustein-Typ erstellen (Seite 307)
Beispiel: Instanz des Bildbaustein-Typs erstellen (Seite 309)
Beispiel: Enthaltene Objekte projektieren (Seite 310)

1.7.4.2 Beispiel: Bildbaustein-Typ erzeugen

Aufgabe

Sie erzeugen einen Bildbaustein-Typ.

Voraussetzung

- Ein Bild ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

Einstellungen

Für das Beispiel benötigen Sie Objekte mit folgenden Einstellungen:

Objekt	Objektname	Eigenschaften
Textfeld	Label_Meter	Text: Meter
E/A-Feld	Output_Meter	Modus: Ausgabe
Schaltfläche	KM_to_Meter	Text AUS: Umrechnen

Vorgehensweise

1. Klicken Sie ins Werkzeugfenster auf die einzelnen Objekte und ziehen Sie die Objekte per Drag&Drop ins Bild.
2. Stellen Sie die Eigenschaften wie oben aufgeführt ein.
3. Selektieren Sie alle Objekte.
4. Wählen Sie im Kontextmenü der Mehrfachauswahl den Befehl "Erzeuge Bildbaustein-Typ". Das Fenster "Typ hinzufügen" wird geöffnet.
5. Geben Sie als Typnamen "KMtoMeter" ein.

Ergebnis

Der Bildbaustein-Typ erscheint unter dem Namen "KMtoMeter" in der Projektbibliothek. Der Bildbaustein-Typ erhält den Status "In Bearbeitung".

Siehe auch

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.4.3 Beispiel: Bildbaustein-Typ konfigurieren

Aufgabe

Sie erstellen eine dynamische Eigenschaft und Variable des Bildbaustein-Typs.

Neue Eigenschaft erstellen

1. Klicken Sie Im Konfigurationsbereich des Bildbausteins auf "Eigenschaften".
2. Klicken Sie in der Liste "Schnittstelle" auf das Symbol "Eigenschaft hinzufügen". Eine neue Eigenschaft wird in der Liste in "Schnittstelle" hinzugefügt.

3. Doppelklicken Sie auf den Namen der Eigenschaft und geben Sie "KMToMeter" ein.
4. Klicken sie auf die Auswahlliste und wählen Sie den Datentyp "Int".

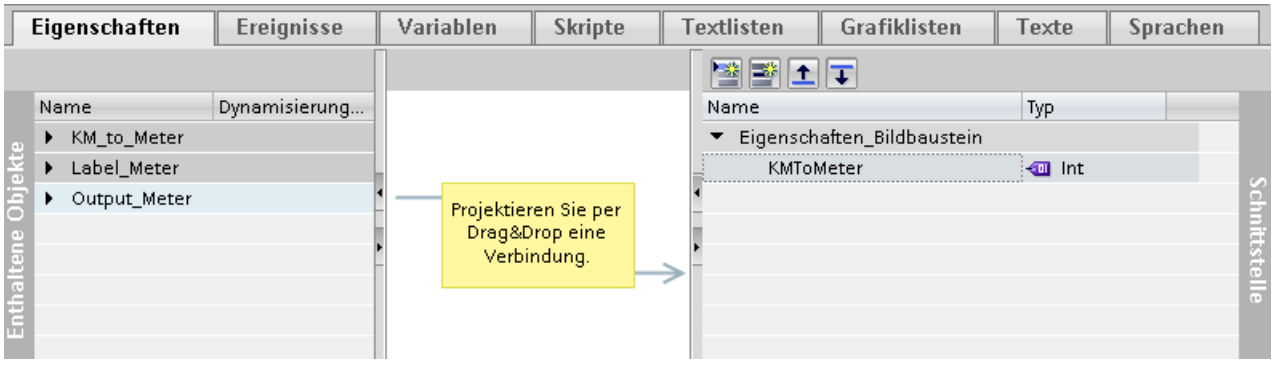

Zwischenergebnis

In der Liste "Schnittstelle" wird die Eigenschaft "KMToMeter" angezeigt. Das Symbol kennzeichnet die dynamische Eigenschaft. Die Eigenschaft verbinden Sie im nächsten Schritt mit einer Variablen.

Variable im Bildbaustein-Typ anlegen

1. Klicken Sie im Konfigurationsbereich auf "Variablen".
2. Klicken Sie in der Tabelle "HMI-Variablen" auf "Hinzufügen". Eine neue Variable wird der Tabelle hinzugefügt. Das Inspektorfenster der Variable wird geöffnet.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Geben Sie als Namen "BB_Tag" ein. Wählen Sie als Datentyp "Int".

Ergebnis

Die Eigenschaft "KMToMeter" und die Variable "BB_Tag" des Bildbaustein-Typs sind im Konfigurationsbereich des Editors "Bildbausteine" angelegt.

Siehe auch

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.4.4 Beispiel: Skript im Bildbaustein-Typ erstellen

Aufgabe

Sie erstellen ein Skript, das den Wert eines Längenmessgeräts von Kilometer in Meter umrechnet.

Vorgehen

1. Klicken Sie im Konfigurationsbereich auf "Skripte > Bildbaustein-Skripte".
2. Doppelklicken Sie auf "VB-Skript hinzufügen".
3. Drücken Sie die Tastenkombination <STRG+J>.
   Die Objektliste wird geöffnet.
4. Klicken Sie auf "HMI-Variablen".
   In der Objektliste wird die Bildbaustein-Variable "BB_Tag" angezeigt.

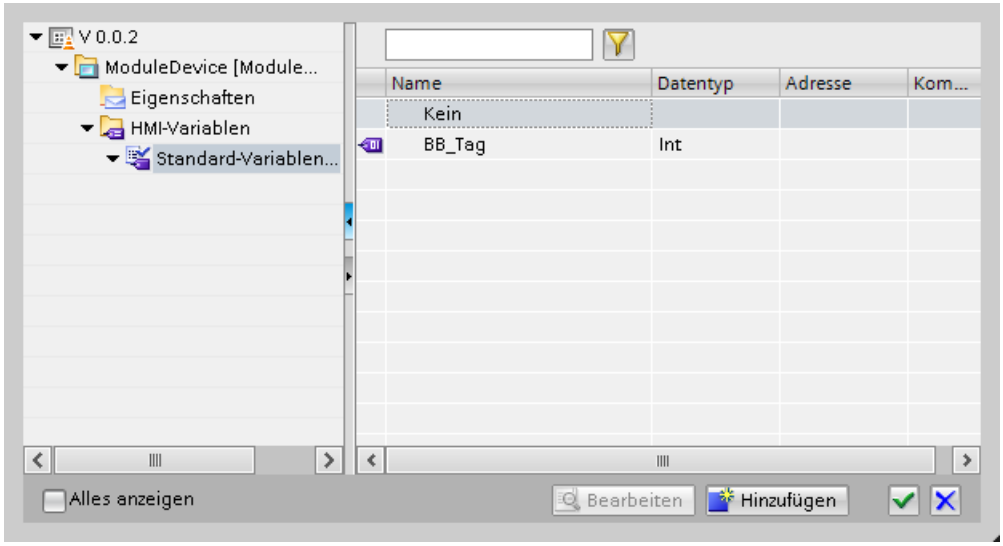

5. Wählen Sie die Variable des Bildbaustein-Typs "BB_Tag" aus. Bestätigen Sie die Auswahl.
6. Fügen Sie ein "="-Zeichen ein.
7. Drücken Sie die Tastenkombination <STRG+J>.Die Objektliste wird geöffnet.
8. Klicken Sie "Eigenschaften".
   In der Objektliste wird die dynamische Eigenschaft "KMtoMeter" angezeigt.

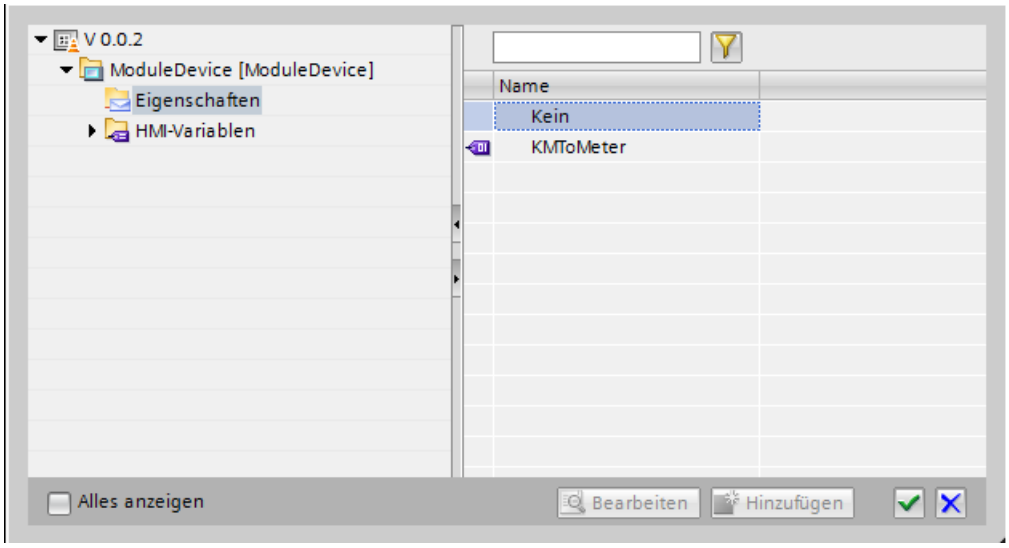

9. Wählen Sie die dynamische Eigenschaft "KMToMeter" aus. Bestätigen Sie die Auswahl.

10. Geben Sie am Ende des Codes "*1000" ein. Das entspricht dem Umrechnungsfaktor von Kilometer in Meter ein.
11. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Einstellung > Name" den Namen "Script_1" ein.

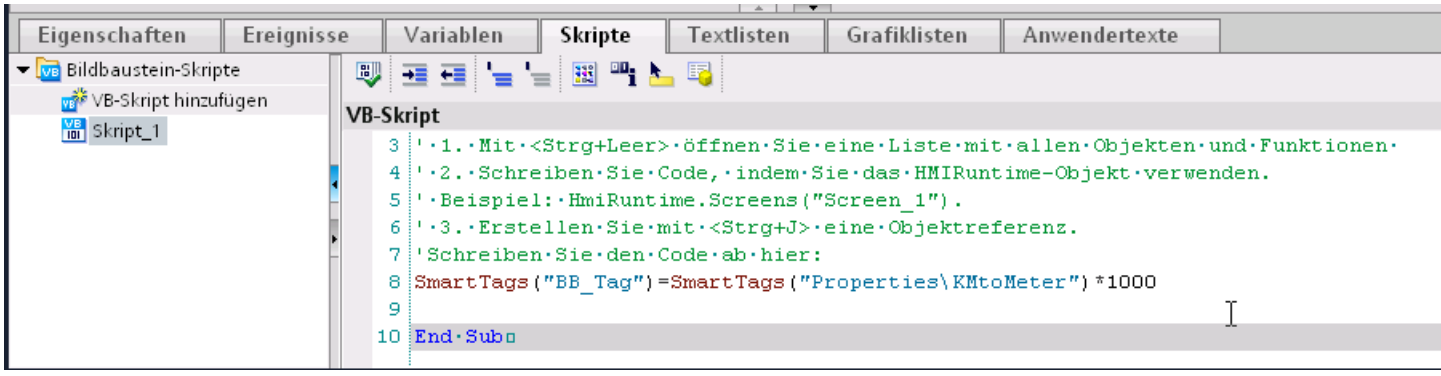

Ergebnis

Das Skript im Bildbaustein-Typ ist angelegt. Wenn Sie den Bildbaustein-Typ "KMtoMeter" in einem Bild verwenden, dann weisen Sie der Eigenschaft "KMtoMeter" eine Variable zu. Der Wert dieser Variablen wird mit dem Faktor 1000 multipliziert und der Variablen des Bildbaustein-Typs "BB_Tag" als neuer Wert zugewiesen. Auf diese Weise versorgen Sie Variablen des Bildbaustein-Typs mit Werten.

Siehe auch

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.4.5 Beispiel: Instanz des Bildbaustein-Typs erstellen

Aufgabe

Sie fügen den Bildbaustein-Typ "KMtoMeter" in ein Bild ein und weisen der dynamischen Eigenschaft "KMtoMeter" eine Variable zu.

Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Ein neues Bild ist angelegt.

Einstellungen

Für das Beispiel benötigen Sie eine Variable mit folgenden Einstellungen:

Name	Steuerungsanbindung	Typ
Kilometer	Nein	ULong

Vorgehensweise

1. Erstellen Sie die HMI-Variable "Kilometer" mit den oben genannten Einstellungen.
2. Öffnen Sie in der Task Card "Bibliotheken" die "Projektbibliothek".
3. Ziehen Sie mit Drag&Drop den Bildbaustein-Typ "KMtoMeter" in das Bild.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Schnittstelle". Die Tabelle zeigt alle Eigenschaften, die Sie im Bildbaustein dynamisieren können

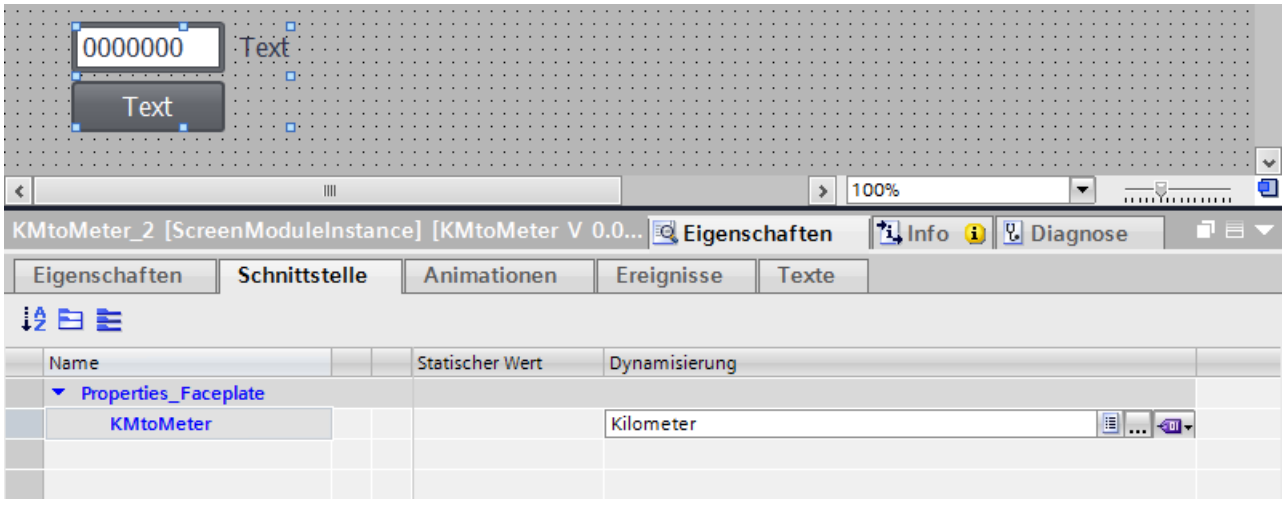

5. Klicken Sie in der Zeile "KMtoMeter" auf die Spalte "Dynamisierung".
6. Wählen Sie "HMI-Variable".
7. Klicken sie auf die Schaltfläche "..."
8. Wählen Sie in der Objektliste die Variable "Kilometer" aus.
9. Bestätigen Sie die Auswahl.

Ergebnis

Die dynamische Eigenschaft "KMtoMeter" ist mit der Variablen "Kilometer" verbunden. In Runtime wird die dynamische Eigenschaft mit Werten der Variable "'Kilometer" versorgt. Die Werte werden im Skript des Bildbaustein-Typs mit dem Faktor 1000 multipliziert und der Variablen im Bildbaustein als neuer Wert zugewiesen.

Siehe auch

Beispiel: Bildbaustein projektieren (Seite 305)

1.7.4.6 Beispiel: Enthaltene Objekte projektieren

Aufgabe

Sie verbinden die im Bildbaustein "KMtoMeter" enthaltenen Objekte mit einer Bildbaustein-Variablen und einem Bildbaustein-Skript.

Das E/A-Feld verbinden Sie mit der Variablen "BB_Var" des Bildbaustein-Typen. Sie verbinden das Ereignis "Klicken" einer Schaltfläche mit dem Skript "Script_1".

Voraussetzung

Der Bildbaustein wird im Editor "Bildbausteine" angezeigt.

E/A-Feld mit einer Variablen verschalten

1. Markieren Sie im Bildbaustein das E/A-Feld "Output_Meter".
2. Verbinden Sie im Inspektorfenster "Allgemein > Prozess > Variable" das E/A-Feld mit der Bildbaustein-Variable "BB_Tag".

Ein Ereignis mit einem Bildbaustein-Skript verbinden

1. Markieren Sie im Bildbaustein die Schaltfläche "KM_to_Meter".
2. Wählen Sie im Inspektorfenster unter "Ereignisse > Klicken" das Skript "Script_1" aus.

Ergebnis

Das E/A-Feld ist mit der Variablen des Bildbaustein-Typs verbunden. Die Schaltfläche ist mit dem Skript des Bildbaustein-Typs verbunden.

Wenn Sie in Runtime auf die Schaltfläche des Bildbausteins klicken, wird das Skript ausgeführt. Im E/A-Feld wird der Wert der Variablen des Bildbaustein ausgegeben.

Siehe auch

Beispiel: Bildbaustein projektieren (Seite 305)

1.8 Bildnavigation projektieren

1.8.1 Grundlagen zur Bildnavigation

1.8.1.1 Grundlagen zur Bildnavigation

Navigationsarten für den Bildwechsel

Für einen Fertigungsprozess aus mehreren Teilprozessen projektieren Sie mehrere Bilder. Damit der Bediener in Runtime von einem Bild zum Nächsten schalten kann, haben Sie folgende Möglichkeiten:

- Bildwechsel an Schaltflächen projektieren
- Bildwechsel an lokale Funktionstaste projektieren

Vorgehen

Bevor Sie einen Bildwechsel erstellen, legen Sie die Anlagenstruktur fest und leiten Sie daraus die Bildwechsel ab, die Sie projektieren wollen.

Das Startbild legen Sie unter "Runtime-Einstellungen > Allgemein > Startbild" an.

Siehe auch

Grundlagen zur Bildnavigation (Seite 312)

Schaltfläche mit Bildwechsel belegen (Seite 313)

Funktionstaste mit einem Bildwechsel belegen (Seite 315)

Übersicht zum Arbeiten mit Menüs und Symbolleisten (Seite 316)

Ein Menü erstellen (Seite 320)

Grundlagen zu Bildern (Seite 33)

1.8.1.2 Grundlagen zur Bildnavigation

Navigationsarten für den Bildwechsel

Für einen Fertigungsprozess aus mehreren Teilprozessen projektieren Sie mehrere Bilder. Damit der Bediener in Runtime von einem Bild zum Nächsten schalten kann, haben Sie folgende Möglichkeiten:

- Bildwechsel an Schaltflächen projektieren

- Bildwechsel an Menü und Symbolleisten projektieren

Hinweis

Für einige Bediengeräte können Sie alternativ Schaltflächen für den Bildwechsel im Permanentbereich hinterlegen. Die Schaltfläche ist dann in allen Bildern eingebunden.

Vorgehen

Bevor Sie einen Bildwechsel erstellen, legen Sie die Bildhierarchie basierend auf der Anlagenstruktur fest und leiten die Bildwechsel ab, die Sie projektieren wollen.

Das Startbild legen Sie unter "Runtime-Einstellungen > Allgemein > Startbild" an.

Siehe auch

Grundlagen zur Bildnavigation (Seite 312)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

1.8.1.3 Schaltfläche mit Bildwechsel belegen

Einleitung

Um im laufenden Betrieb am Bediengerät zwischen den Bildern umzuschalten, projektieren Sie im Bild eine Schaltfläche.

Hinweis

Wenn Sie im Inspektorfenster eines Bildes die Animation "Sichtbarkeit" auf "ausgeblendet" eingestellt haben, kann dieses Bild in Runtime nicht aufgerufen werden.

Voraussetzungen

- Ein Projekt ist angelegt.
- Das Bild "Screen_2" ist angelegt.
- Das Bild "Screen_1" ist angelegt.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Screen_1". Das Bild wird im Arbeitsbereich angezeigt.
2. Ziehen Sie per Drag&Drop aus dem Projektnavigation "Screen_2" in das geöffnete Bild. Eine Schaltfläche mit dem Namen "Screen_1" wird eingefügt.

3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
   In der "Funktionsliste" wird die Systemfunktion "AktiviereBild" angezeigt.

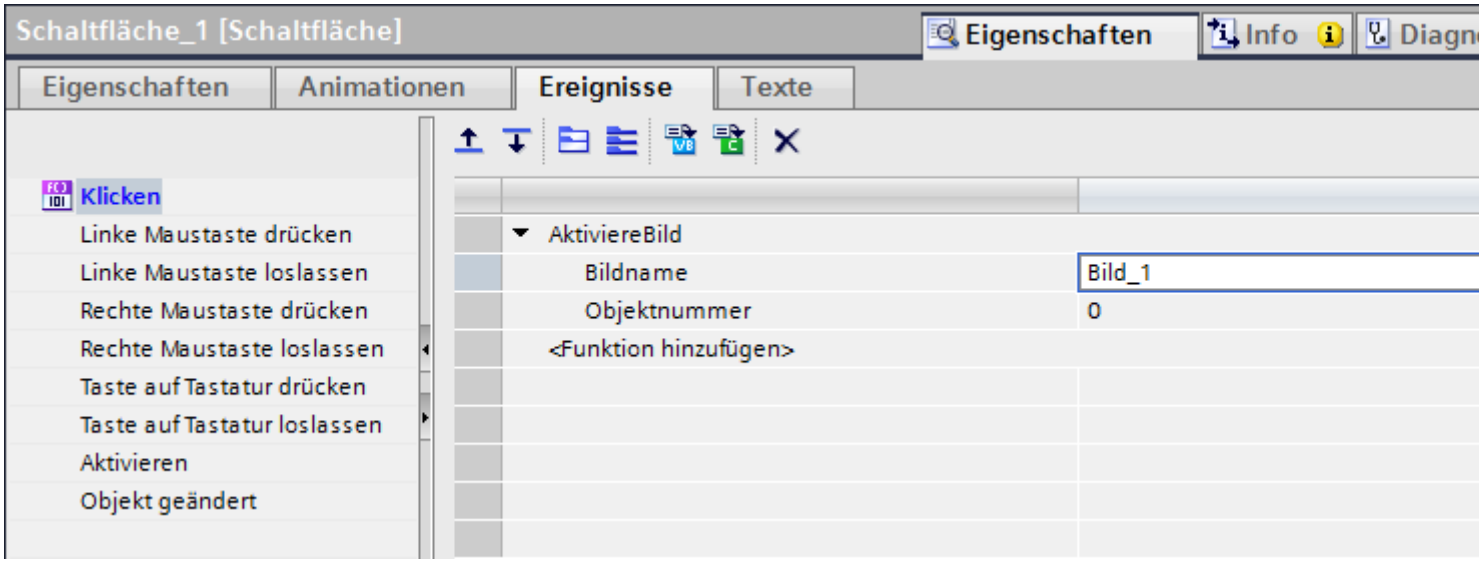

4. Geben Sie bei Bedarf als "Objektnummer" die Tab-Reihenfolge-Nummer des Objekts ein, das nach dem Bildwechsel den Fokus erhalten soll. Sie können auch eine Variable angeben, welche die Objektnummer enthält.

Alternatives Vorgehen

1. Ziehen Sie per Drag&Drop eine Schaltfläche aus der Task Card "Werkzeuge" in das Bild "Screen2".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
3. Wählen Sie die Systemfunktion "AktiviereBild".
4. Wählen Sie für "Bildnummer" das Bild "Screen_2".

Ergebnis

In Runtime wechselt der Bediener mit der Schaltfläche zum Bild "Screen_1". Wenn Sie eine Objektnummer angegeben haben, erhält das Objekt mit dieser Objektnummer nach dem Bildwechsel den Fokus.

Siehe auch

Grundlagen zur Bildnavigation (Seite 312)

1.8.1.4 Funktionstaste mit einem Bildwechsel belegen

Einleitung

Um im laufenden Betrieb am Bediengerät zwischen den Bildern umzuschalten, projektieren Sie im Bild eine Funktionstaste zum Bildwechsel.

Hinweis

Wenn Sie im Inspektorfenster eines Bildes die Animation "Sichtbarkeit" auf "ausgeblendet" eingestellt haben, kann dieses Bild in Runtime nicht aufgerufen werden.

Voraussetzungen

- Ein Projekt ist angelegt.
- Das Bild "Screen_2" ist angelegt.
- Das Bild "Screen_1" ist angelegt.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Screen_1". Das Bild wird im Arbeitsbereich angezeigt.
2. Ziehen Sie aus dem Projektnavigation "Screen_2" auf eine Funktionstaste z. B. "F2". Die projektierte Funktionstaste zeigt ein gelbes Dreieck an.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Taste drücken". Die Systemfunktion wird "AktiviereBild" angezeigt.

Ergebnis

In Runtime wechselt der Bediener mit der Funktionstaste "F2" zum angegebenen Bild "Screen_2".

Siehe auch

Grundlagen zur Bildnavigation (Seite 312)

1.8.2 Menüs und Symbolleisten für Runtime Professional

1.8.2.1 Grundlagen

Übersicht zum Arbeiten mit Menüs und Symbolleisten

Einleitung

Im Editor "Menüs & Symbolleisten" projektieren Sie benutzerdefinierte Menüs und Symbolleisten. Die benutzerdefinierten Menüs und Symbolleisten werden in allen Bildern eines Projekts und zusätzlich in Bildfenstern angezeigt. Die Menüeinträge und Symbole verbinden Sie mit den lokalen Skripten.

Verwendung

Sie verwenden benutzerdefinierte Menüs und Symbolleisten, um z. B. eine Bildnavigation zu realisieren. Die benutzerdefinierten Menüs und Symbolleisten werden in jedem Bild des Projekts oder in einem Bildfenster angezeigt. Auf diese Weise können Sie von jedem Bild in alle Bilder wechseln, zu denen Sie Bildwechsel an Menüeinträge oder Symbole projektiert haben.

Siehe auch

Grundlagen zu Menüs (Seite 316)

Konfiguration (Seite 318)

Editor "Menüs und Symbolleisten" (Seite 318)

Grundlagen zur Bildnavigation (Seite 312)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Grundlagen zu Menüs

Einleitung

Ein Menü wird im Bild links beginnend am oberen Rand platziert. In einer Konfigurationsdatei legen Sie fest, welches Menü und welche Symbolleisten mit dieser Datei in Bildern oder Bildfenstern angelegt werden. Pro Konfigurationsdatei können Sie eine Menüzeile projektieren.

Menüaufbau

Die Projektierungsmöglichkeiten für ein Menüelement sind abhängig von der Position, in der sich das Menüelement im Menüaufbau befindet.

Für ein Menü erstellen Sie folgende Bestandteile:

- Hauptmenübefehl
  - Wird im Bild angezeigt.
  - Öffnet ein Untermenü.
- Menübefehl
  - Wird im Bild angezeigt, wenn das Hauptmenü geöffnet ist.
  - Führt beim Klicken eine definierte Prozedur aus.
  - Unter "Eigenschaften > Ereignisse" können Sie einen Parameter (z. B. einen Bildnamen) eintragen, der an die Prozedur übergeben wird.
  - Öffnet wenn vorhanden ein Untermenü.
- Untermenübefehl
  - Wird im Bild angezeigt, wenn ein übergeordneter Menübefehl geöffnet ist.
- Trennlinien zwischen den Menüeinträgen

Das folgende Bild zeigt einen typischen Menüaufbau mit verschiedenen Menüelementen:

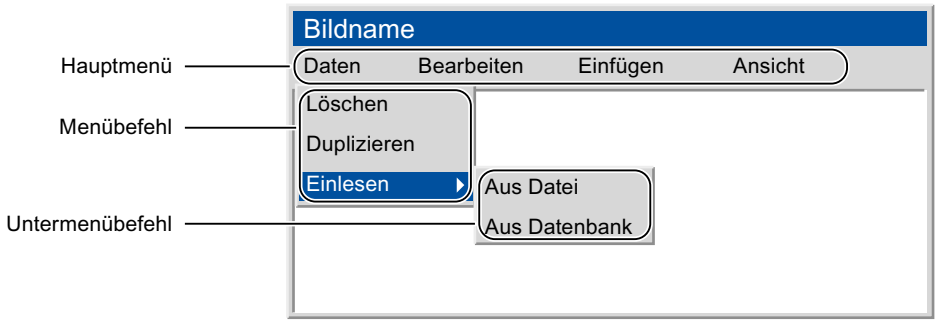

Bearbeitungsmöglichkeiten

Zur Projektierung eines Menüs haben Sie folgende Möglichkeiten:

- Neues Hauptmenü hinzufügen
- Neuen Menübefehl / Untermenübefehl hinzufügen
- Trennlinie hinzufügen
- Selektierten Menübefehl nach links verschieben
- Selektierten Menübefehl nach rechts verschieben
- Einstellungen im Inspektorfenster ändern

Siehe auch

Übersicht zum Arbeiten mit Menüs und Symbolleisten (Seite 316)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Konfiguration

Einleitung

Menüs und Symbolleisten, die Sie projektiert haben, kombinieren Sie in den Konfigurationsdateien. Jede Konfigurationsdatei enthält maximal ein Menü, kann aber mehrere Symbolleisten enthalten. Sie können beliebig viele Konfigurationsdateien anlegen. Eine Konfigurationsdatei wählen Sie aus folgenden Gründen aus:

- Um einem Bildfenster ein Menü und eine Symbolleiste hinzuzufügen
- Um allen Bildern Ihres Projekts ein Menü und eine Symbolleiste hinzuzufügen
- Um dynamisiert die Menüs und Symbolleisten in Runtime auszutauschen

Konfigurationsdatei projektieren

Sie haben folgende Möglichkeit, um die Konfigurationsdatei zu projektieren:

- Berechtigungen zuweisen
  Wenn ein angemeldeter Bediener nicht die benötigte Berechtigung besitzt, werden die Elemente automatisch deaktiviert.
- Menüeinträge und Symbole ausblenden oder deaktivieren
  Wenn Sie den veränderten Funktionsumfang in einer neuen Konfigurationsdatei speichern, können Sie die Konfigurationsdatei z. B. bei einem Benutzerwechsel in Runtime austauschen.

Siehe auch

Übersicht zum Arbeiten mit Menüs und Symbolleisten (Seite 316)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Editor "Menüs und Symbolleisten"

Einleitung

Im Editor "Menüs & Symbolleisten" erstellen, konfigurieren und bearbeiten Sie benutzerdefinierte Menüs und Symbolleisten. Die benutzerdefinierten Menüs und Symbolleisten werden in den Konfigurationsdateien kombiniert und sind dann in allen Bildern eines Projekts und zusätzlich in Bildfenstern anzuzeigen. Die Menüeinträge und Symbole verbinden Sie mit den lokalen Skripten.

Aufbau

Der Editor "Menüs & Symbolleisten" hat drei Registerkarten:

- Symbolleisten
- Menüs
- Konfigurationen

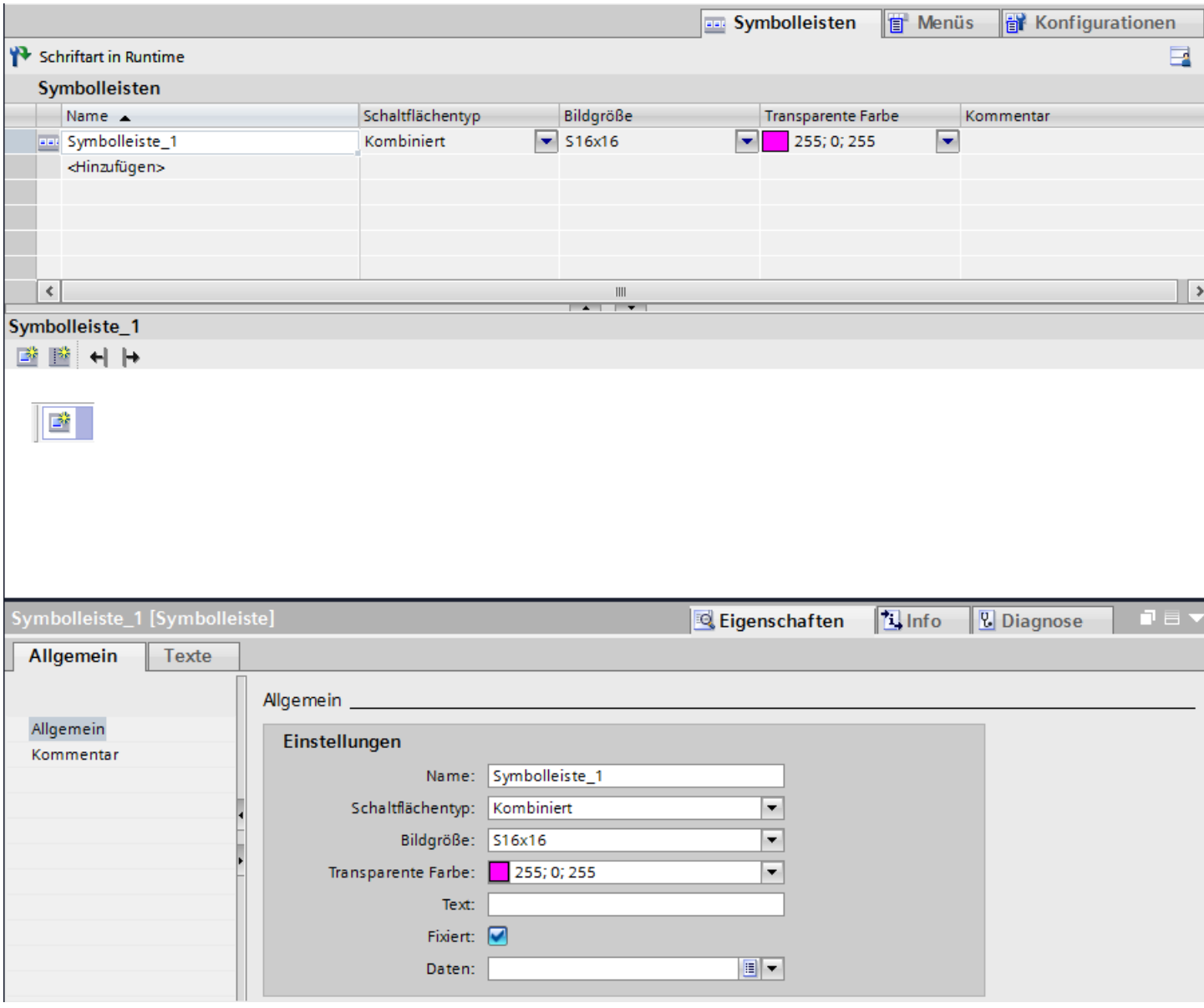

Sie öffnen den Editor in der Projektnavigation mit einem Doppelklick auf "Bildverwaltung > Menüs & Symbolleisten".

Arbeitsbereich

In den Registerkarten "Symbolleisten" und "Menüs" ist der Arbeitsbereich in zwei Bereiche geteilt:

- In der Tabelle definieren Sie die gesamte Symbolleiste und/oder das gesamte Menü.
- Der untere Bereich bietet Ihnen eine Vorschau auf Ihre Symbolleisten bzw. Menüs. Sie erstellen und konfigurieren die einzelnen Symbole und/oder Menübefehle und legen Aufbau, beschriftende Texte und symbolisierende Grafiken fest.

Die Registerkarten "Konfigurationen" hat einen ungeteilten Arbeitsbereich, der den tabellarischen Editor zur Auswahl der zugehörigen Symbolleisten und des zugehörigen Menüs enthält.

Inspektorfenster

Im Inspektorfenster unter "Eigenschaften > Eigenschaften" konfigurieren Sie die Menüs und Symbolleisten.

Siehe auch

Übersicht zum Arbeiten mit Menüs und Symbolleisten (Seite 316)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

1.8.2.2 Arbeiten mit Menüs und Symbolleisten

Ein Menü erstellen

Einleitung

Im Editor "Menüs & Symbolleisten" erstellen Sie ein Menü mit Hauptmenübefehlen und den dazugehörigen Menübefehlen mit bis zu sechs Hierarchieebenen. Das Menü wird dabei so angezeigt, wie es später in den Bildern oder einem Bildfenster angezeigt wird.

Voraussetzung

- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Menüs" ist geöffnet.

Menü anlegen

1. Doppelklicken Sie in der Tabelle "Menüs" auf "Hinzufügen".
   In der Vorschau der Menübefehle wird der erste Hauptmenübefehl angelegt.
2. Ändern Sie Namen des Menüs und geben Sie evtl. einen Kommentar ein.

Hauptmenübefehle anlegen

1. Klicken Sie in der Vorschau des Menüs auf den bereits erstellten Hauptmenübefehl. Um das Objektumfassende Rechteck werden vier gelbe Sterne angezeigt.

2. Klicken Sie auf den gelben Stern rechts vom Standardtext. Ein weiterer Hauptmenübefehl wird angelegt.

3. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Name" einen Namen ein.
4. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Text" die Beschriftung des Hauptmenübefehls ein.

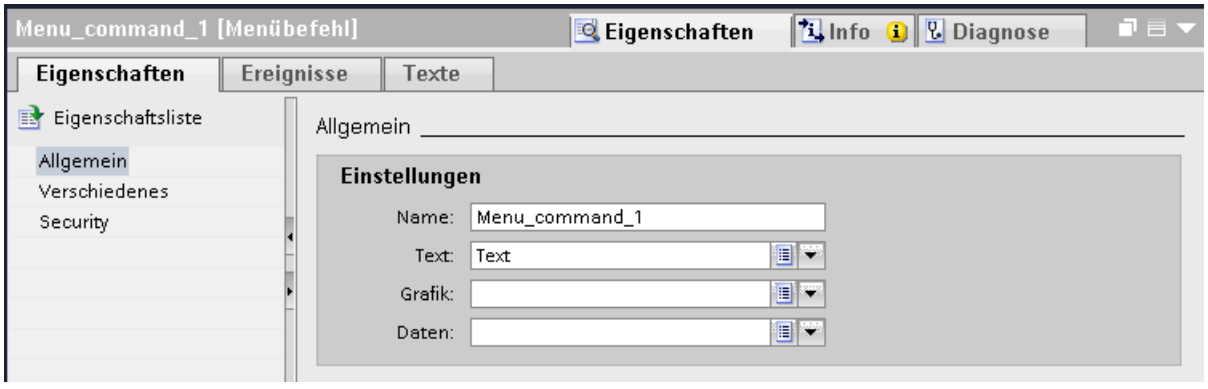

1. Um den Hauptmenübefehl in Runtime sichtbar darzustellen, aktivieren Sie "Eigenschaften > Eigenschaften > Verschiedenes > Sichtbarkeit".
2. Um eine Bedienberechtigung festzulegen, wählen Sie "Eigenschaften > Eigenschaften > Sicherheit" eine Berechtigung aus.

Menübefehle anlegen

1. Klicken Sie in der Vorschau auf einen Hauptmenübefehl.
   Um das Objektumfassende Rechteck werden gelbe Sterne angezeigt.

2. Klicken Sie auf den gelben Stern unterhalb des Objektumfassenden Rechtecks.
   Ein Menübefehl unterhalb des Hauptmenübefehls wird angelegt.

3. Konfigurieren Sie den Menübefehl im Inspektorfenster wie einen Hauptmenübefehl.

Hinweis

Die Berechtigung, Sichtbarkeit und Aktivierung für Runtime sind sowohl im Menü als auch in jedem einzelnen Menübefehl projektierbar. Bei unterschiedlicher Projektierung gilt: Menü überschreibt Menübefehl.

Untermenübefehle anlegen

1. Klicken Sie in der Vorschau auf einen Menübefehl.
   Um das Objektumfassende Rechteck werden gelbe Sterne angezeigt.

2. Klicken Sie auf den gelben Stern rechts vom Menübefehl.
   Ein Untermenübefehl wird angelegt.

3. Konfigurieren Sie den Untermenübefehl im Inspektorfenster wie einen Hauptmenübefehl.

Ergebnis

Das Menü ist erstellt und kann im Register "Konfigurationen" als Menü für eine spezifische Konfigurationsdatei ausgewählt werden.

Siehe auch

Eine Symbolleiste erstellen (Seite 323)
Eine Konfigurationsdatei anlegen (Seite 326)
Menüs und Symbolleisten bearbeiten (Seite 328)
Menüs und Symbolleisten für alle Bilder projektieren (Seite 329)
Menüs und Symbolleisten in einem Bildfenster projektieren (Seite 330)
Grundlagen zur Bildnavigation (Seite 312)
Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)
Menüs und Symbolleisten in Runtime anpassen (Seite 330)
Menübefehl oder Symbol mit einer Prozedur verbinden (Seite 325)

Eine Symbolleiste erstellen

Einleitung

Im Editor "Menüs & Symbolleisten" erstellen Sie eine Symbolleiste mit Schaltflächen. In der Vorschau wird die Symbolleiste so angezeigt wie später in den Bildern oder einem Bildfenster.

Pro Konfiguration können Sie beliebig viele Symbolleisten konfigurieren. Sie legen für jede Symbolleiste fest, an welcher Position die Symbolleiste im Bild platziert wird. Darüber hinaus können Sie festlegen, ob eine Symbolleiste vom Bediener frei platziert werden kann.

Hinweis

Symbolleisten sind aufgrund der verbundenen Skripte gerätespezifisch. Wenn sie für ein anderes Gerät verwendet werden, gehen die Funktionalitäten verloren.

Voraussetzung

- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Symbolleisten" wird angezeigt.

Symbolleiste anlegen

1. Doppelklicken Sie in der Tabelle "Symbolleisten" auf "Hinzufügen".
   Eine Symbolleiste wird angelegt und mit einem vorbelegten Namen versehen.
   In der Vorschau der Symbolleiste wird eine Schaltfläche angelegt.
2. Geben Sie in der Spalte "Name" einen Namen für die Symbolleiste ein.

3. Wählen Sie als "Schaltflächentyp" eine der folgenden Einstellungen:
   - Text und Grafik kombiniert
   - Nur Grafik
   - Nur Text
4. Geben Sie in der Spalte "Bildgröße" die Größe der Schaltflächen an.

Schaltflächen anpassen

1. Selektieren Sie die Symbolleiste, zu der Sie Schaltflächen anlegen wollen.
2. Klicken Sie in der Vorschau der Symbolleiste auf die bereits vorhandene Schaltfläche. Die Schaltfläche enthält je nach Typ einen Standardtext und/oder eine Standardgrafik.

Im Inspektorfenster werden die Eigenschaften angezeigt.

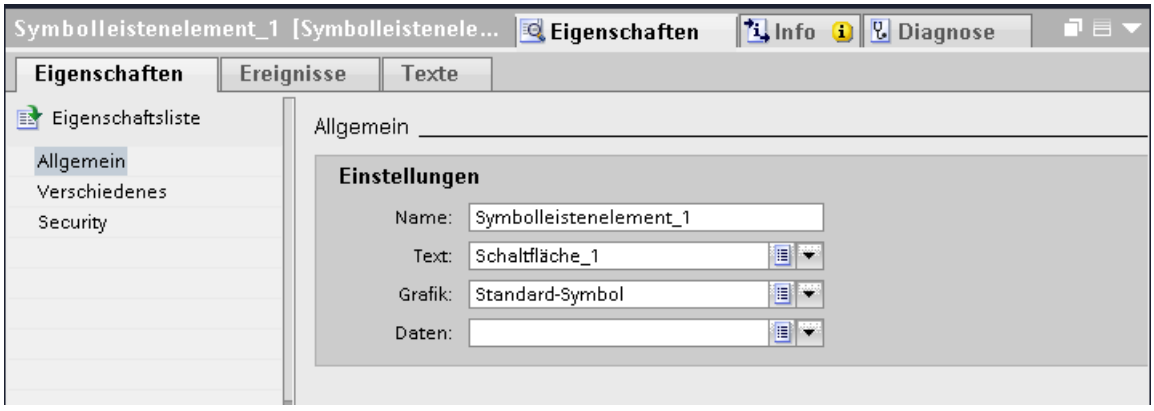

3. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" einen Namen ein.
4. Wählen Sie abhängig vom Schaltflächentyp einen Text und/oder eine Grafik für die Gestaltung der Schaltfläche.
5. Um die Schaltfläche in Runtime sichtbar darzustellen, aktivieren Sie "Eigenschaften > Eigenschaften > Verschiedenes > Sichtbarkeit".
6. Um in Runtime einen Tooltip anzuzeigen, geben Sie unter "Eigenschaften > Eigenschaften > Verschiedenes > "Tooltip" einen Text ein.

7. Um eine Bedienberechtigung festzulegen, wählen Sie "Eigenschaften > Eigenschaften > Sicherheit" eine Berechtigung aus.
8. Um weitere Schaltflächen einzufügen, klicken Sie in der Vorschau der Symbolleiste auf das Symbol .

Hinweis

Die Berechtigung, Sichtbarkeit und Aktivierung für Runtime ist sowohl in der Symbolleiste als auch in jeder einzelnen Schaltfläche projektierbar. Bei unterschiedlicher Projektierung gilt folgende Regel: Symbolleiste überschreibt Schaltfläche.

Ergebnis

Eine Symbolleiste mit den definierten Schaltflächen ist angelegt und kann im Register "Konfigurationen" als Menü für eine spezifische Konfigurationsdatei ausgewählt werden.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Menübefehl oder Symbol mit einer Prozedur verbinden

Einleitung

Im Editor "Menüs & Symbolleisten" projektieren Sie benutzerdefinierte Menüs und Symbolleisten. Die benutzerdefinierten Menüs und Symbolleisten werden in allen Bildern eines Projekts und/oder in einem Bildfenster angezeigt. Die einzelnen Menübefehle und Schaltflächen verbinden Sie mit einer Funktionsliste. Als Ereignis steht nur "Klicken" zur Verfügung.

Voraussetzung

- Ein Bild "Screen_1" ist angelegt.
- Ein Bild "Screen_2" ist angelegt.
- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Eine Schaltfläche oder ein Menübefehl ist angelegt.

Ereignis mit Systemfunktion an eine Schaltfläche oder ein Menübefehl projektieren

1. Selektieren Sie die Schaltfläche oder den Menübefehl.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".

3. Wählen Sie aus der Funktionsliste die Systemfunktion "AktiviereBild".
4. Wählen Sie "Screen_2" als "Bildname".

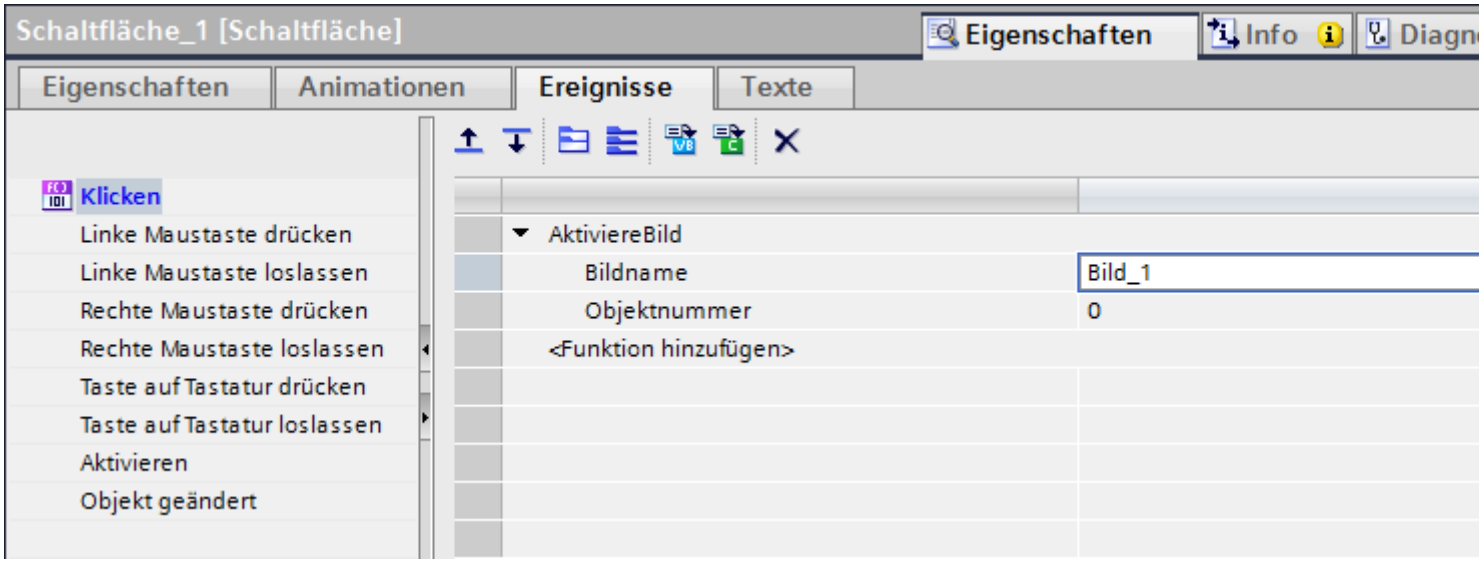

Ein Ereignis über Skripte an eine Schaltfläche oder einen Menübefehl projektieren

1. Selektieren Sie die Schaltfläche oder den Menübefehl.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Ereignisse > Klicken".
3. Klicken Sie in der Symbolleiste des Inspektorfensters auf das Symbol "VB-Skript".
4. Schreiben Sie das erforderliche Skript.
   "Item" ist dabei das Objekt, das das Ereignis auslöst, in diesem Fall die Schaltfläche oder der Menübefehl. Es wird mit der Eigenschaft "Name" im Skript angegeben.

Ergebnis

Wenn der Bediener auf die Schaltfläche oder auf den Menübefehl klickt, wird die projektierte Funktion in Runtime ausgeführt.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Eine Konfigurationsdatei anlegen

Einleitung

Eine Konfigurationsdatei verwenden Sie, um in allen Bildern Ihres Projekts oder in einem Bildfenster Menüs und/oder Symbolleisten anzeigen zu lassen. Eine Konfigurationsdatei besteht dabei aus einem/keinem Menü und/oder beliebig vielen Symbolleisten.

Voraussetzung

- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Konfigurationen" wird angezeigt.
- Menüs und Symbolleisten sind angelegt.

Konfigurationsdatei anlegen

1. Doppelklicken Sie in der Tabelle "Konfigurationen" auf "Hinzufügen". Eine Konfigurationsdatei wird angelegt.
2. Ändern Sie den Namen der Konfigurationsdatei und geben Sie evtl. einen Kommentar ein.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Menü > Name" das Menü aus.

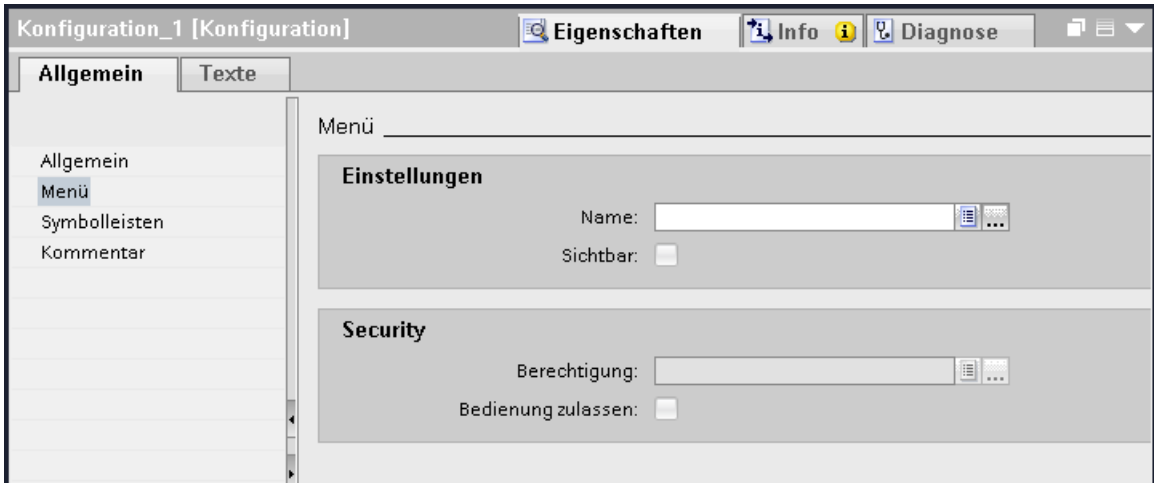

4. Legen Sie fest, ob das Menü in Runtime aktiviert und sichtbar sein soll und welche Berechtigung für die Bedienung erforderlich ist.
5. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleisten" die Symbolleisten aus, die der Konfigurationsdatei hinzugefügt werden sollen.

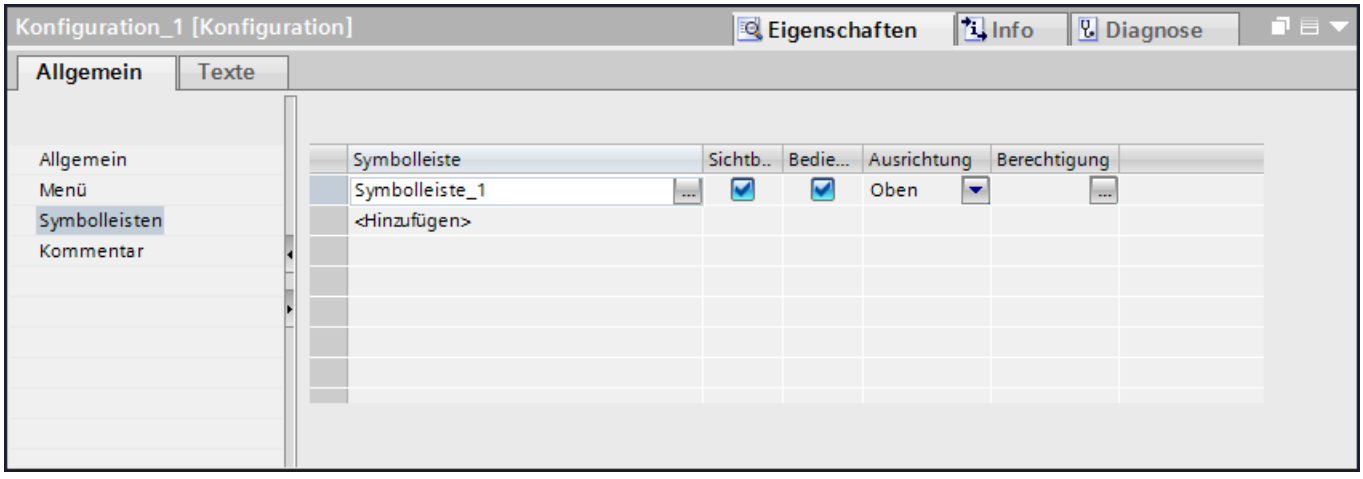

6. Legen Sie für jede gewählte Symbolleiste fest, ob die Symbolleiste in Runtime aktiviert und sichtbar sein soll und welche Berechtigung für die Bedienung erforderlich ist.
7. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleisten" die Ausrichtung für jede Symbolleiste fest.

Ergebnis

Die Konfigurationsdatei kann im Projekt für alle Bilder oder in einem Bildfenster ausgewählt werden.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Menüs und Symbolleisten bearbeiten

Einleitung

Einmal erstellte Menüs und Symbolleisten bearbeiten und verändern Sie ebenfalls im Arbeitsbereich des Editors "Menüs & Symbolleisten".

Trennstrich einfügen

1. Öffnen Sie die Registerkarte "Menüs" oder die Registerkarte "Symbolleisten".
2. Selektieren Sie den Menübefehl bzw. die Schaltfläche, vor dem/der Sie einen Trennstrich einfügen wollen.
3. Klicken Sie auf das Symbol .
   Vor dem selektierten Menübefehl wird ein Trennstrich angezeigt.

Alternativ fügen Sie Trennstriche über das Kontextmenü mit dem Befehl "Trennzeichen hinzufügen" ein.

Menübefehl/Schaltfläche verschieben

1. Öffnen Sie die Registerkarte "Menüs" oder die Registerkarte "Symbolleisten".
2. Selektieren Sie den Menübefehl oder die Schaltfläche, den/die Sie verschieben wollen.
3. Klicken Sie auf eines der Symbole oder .
   Der Menübefehl oder die Schaltfläche wird um eine Stelle nach links oder rechts verschoben.

Alternativ verschieben Sie ein selektiertes Element per Drag&Drop an die gewünschte Position. Eine Markierung zeigt Ihnen die gewählte Zielposition an.

Menüs anzeigen und ausblenden.

1. Öffnen Sie die Registerkarte "Menüs"
2. Um alle Menübefehle anzeigen zu lassen, klicken Sie auf das Symbol .
3. Um nur die Hauptmenübefehle anzeigen zu lassen, klicken Sie auf das Symbol .
   Nur das Hauptmenü wird angezeigt.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Menüs und Symbolleisten für alle Bilder projektieren

Einleitung

In den Runtime-Einstellungen können Sie eine Konfigurationsdatei mit einem benutzerdefinierten Menü und Symbolleisten als Standard festlegen. Das benutzerdefinierte Menü und die Symbolleisten werden in jedem Bild an der projektierten Position angezeigt.

Mit VB-Skript können Sie in Runtime andere Konfigurationsdateien laden. Wenn sich z. B. ein anderer Benutzer anmeldet, können Sie eine Konfigurationsdatei mit verändertem Funktionsumfang bei den benutzerdefinierten Menüs und Symbolleisten laden.

Menüs und Symbolleisten für alle Bilder zu projektieren

1. Öffnen Sie den Editor "Runtime-Einstellungen".
2. Wählen Sie unter "Allgemein > Menüs & Symbolleisten" die Konfigurationsdatei aus, deren Menü und Symbolleisten in allen Bildern Ihres Projekts angezeigt werden sollen.

Ergebnis

Wenn Sie das Projekt aktivieren, werden in allen Bildern des Projekts die benutzerdefinierten Menüs und Symbolleisten angezeigt.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Menüs und Symbolleisten in einem Bildfenster projektieren

Einleitung

In einem Bildfenster können Sie unabhängig von der Startkonfiguration eine zusätzliche Konfigurationsdatei mit benutzerdefinierten Menüs und Symbolleisten laden.

Im Bildfenster können damit die benutzerdefinierten Menüs und Symbolleisten des Bildfensters und der Startkonfiguration angezeigt werden.

Voraussetzung

- Eine Konfugurationsdatei ist angelegt.
- Ein Bild ist geöffnet.

Menüs und Symbolleisten in einem Bildfenster projektieren

1. Ziehen Sie per Drag&Drop das Objekt "Bildfenster" in das Bild.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" die Konfigurationsdatei aus.

Ergebnis

Im Bildfenster werden in Runtime die benutzerdefinierten Menüs und Symbolleisten angezeigt.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Menüs und Symbolleisten in Runtime anpassen

Einleitung

Im Editor "Menüs & Symbolleisten" projektieren Sie benutzerdefinierte Menüs und Symbolleisten. Die benutzerdefinierten Menüs und Symbolleisten werden in allen Bildern eines Projekts und zusätzlich in Bildfenstern angezeigt. Die Menüeinträge und Symbole verbinden Sie mit Prozeduren aus den lokalen Skripten.

Beispiel

In einem Bildfenster können Sie die Eigenschaft "MT Konfiguration" dynamisieren. Außerdem besteht die Möglichkeit, die Konfiguration des Grundbildes und der Bildfenster über VB-Skript zu ändern.

Das folgende Beispiel zeigt eine Prozedur, der die zu ladende Konfigurationsdatei als Parameter übergeben wird.

```
Sub ChangeMenuToolbarConfigFile (ByVal strMTConfigFile)
HMIRuntime.MenuToolBarConfig = strMTConfigFile
End Sub
```

Hinweis

In jeder Konfiguration können Sie ein Menü und mehrere Symbolleisten projektieren.

Sie haben die Möglichkeit, einzelne Elemente von Menüs oder Schaltflächen inaktiv oder unsichtbar zu schalten. Sie können damit in verschiedenen Bildern einen angepassten Funktionsumfang darstellen, indem Sie die Konfiguration bearbeiten und unter neuem Namen speichern.

Siehe auch

Ein Menü erstellen (Seite 320)

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

1.8.2.3 Beispiel: Bildwechsel mit Symbolleisten projektieren

Beispiel: Symbolleisten und Schaltflächen projektieren

Einleitung

Sie haben Ihren Prozess in sechs Teilprozessbilder dargestellt. Ein Bild ist das Startbild, das kein übergeordnetes Bild hat.

Voraussetzung

- Die Bilder des Projekts sind angelegt.
- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Symbolleisten" ist geöffnet.

Symbolleiste für Startbild anlegen

1. Klicken Sie in der Tabelle "Symbolleisten" auf "Hinzufügen". Eine Symbolleiste wird angelegt.
2. Geben Sie in der Spalte "Name" den Namen "Bildwechsel_Startbild".
3. Wählen Sie als "Schaltflächentyp" die Einstellung "Kombiniert".

4. Wählen Sie die Bildgröße "S48x48".

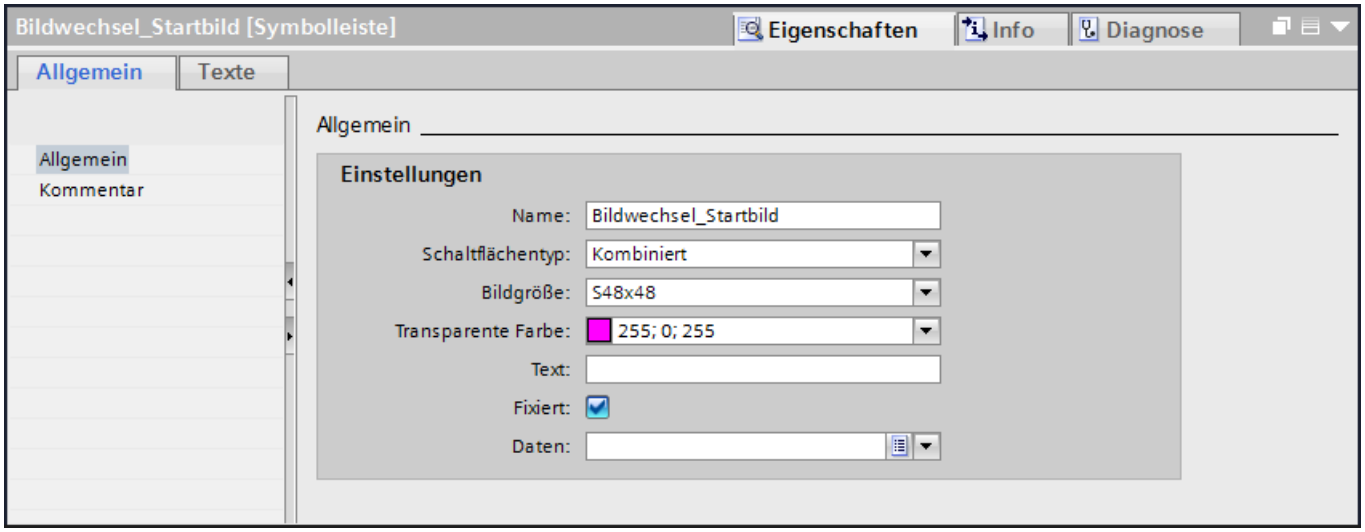

5. Wiederholen Sie das Vorgehen. Legen Sie für jedes Bild eine Symbolleiste mit einem entsprechenden Namen an.

Schaltflächen für Startbild anlegen

1. Selektieren Sie die erste Schaltfläche und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".

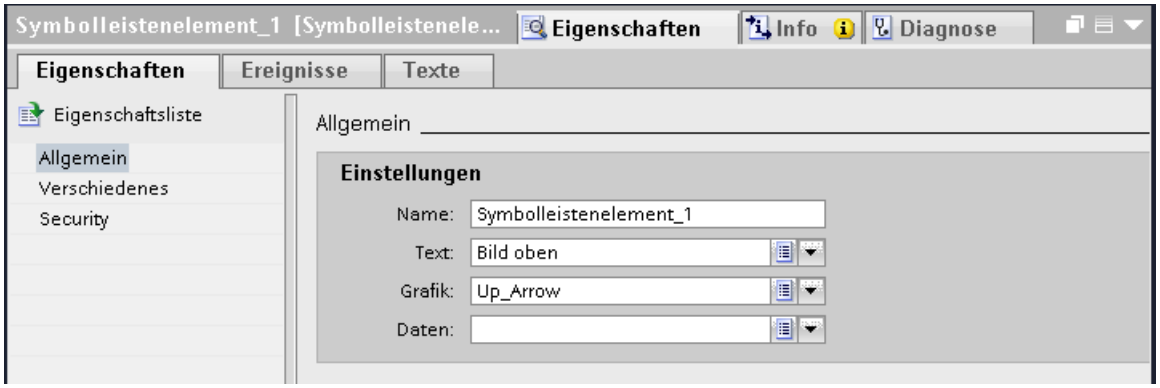

2. Geben Sie unter "Text" "Bild oben" ein.
3. Wählen Sie unter "Grafik" die Grafik "Up_Arrow".
4. Um die Schaltfläche in Runtime sichtbar darzustellen, aktivieren Sie "Eigenschaften > Eigenschaften > Verschiedenes > Sichtbarkeit".
5. Geben Sie unter "Eigenschaften > Eigenschaften > Verschiedenes > Tooltip" den Text "Wechselt in übergeordnetes Bild" ein.

6. Klicken Sie in der Vorschau der Symbolleiste auf das Symbol .

Eine Schaltfläche wird angelegt.

7. Wiederholen Sie das Vorgehen für alle Schaltflächen mit folgenden Eigenschaften:
   - Schaltfläche_2: Text: "Bild unten"; Grafik: "Down_Arrow".
   - Schaltfläche_3: Text: "Bild links"; Grafik: "Left_Arrow".
   - Schaltfläche_4: Text: "Bild rechts"; Grafik: "Right_Arrow".
   - Schaltfläche_5: Text: "Startbild"; Grafik: "Home".

Ergebnis

Ihnen steht für jedes Bild eine spezifische Symbolleiste zur Verfügung.

Weitere Schritte

Folgende Schritte sind noch notwendig, um die Symbolleisten für den Bildwechsel zu nutzen:

- Schaltflächen mit Systemfunktionen belegen
- Symbolleiste in eine Konfiguration einfügen
- Konfiguration im Bildfenster projektieren

Siehe auch

Beispiel: Funktionen an Schaltflächen projektieren (Seite 333)

Beispiel: Konfigurationsdatei anlegen (Seite 335)

Beispiel: Konfigurationsdatei im Bildfenster projektieren (Seite 336)

Grundlagen zur Bildnavigation (Seite 312)

Beispiel: Funktionen an Schaltflächen projektieren

Einleitung

Damit jede Schaltfläche die gewünschte Funktion zum Bildwechsel ausführt, projektieren Sie die Systemfunktionen mit Hilfe der Funktionsliste.

Voraussetzung

- Die Symbolleisten für alle Bilder sind angelegt.
- Für jede Symbolleiste sind Schalflächen für die entsprechenden Bilder angelegt.
- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Symbolleisten" ist geöffnet.

Systemfunktion an eine Schaltfläche der Symbolleiste projektieren

1. Öffnen Sie die Symbolleiste für das Startbild.
2. Selektieren Sie die erste Schaltfläche.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
4. Wählen Sie aus der Funktionsliste die Systemfunktion "AktiviereBild".
5. Wählen Sie als "Bildname" den Namen des Bildes ein, zu dem gewechselt werden soll..
6. Geben Sie unter "Objektnummer" als Bildobjekt im Bild, zu dem gewechselt werden soll, "2" an. Das Bildobjekt mit der Tab-Reihenfolge-Nummer "2" erhält den Fokus.

   Hinweis

   Die Objektnummer wird bei der Projektierung der Tab-Reihenfolge im Bild definiert. Die Reihenfolge lassen Sie sich im Bild mit dem Menübefehl "Bearbeiten > Tab-Reihenfolge > Tab-Reihenfolge bearbeiten" anzeigen.

7. Wiederholen Sie das Vorgehen für alle Schaltflächen dieser Symbolleiste.
   Wählen Sie als "Bildname" immer das Bild aus, das mit der jeweiligen Schaltfläche aktiviert werden soll.
8. Wiederholen Sie das Vorgehen für die Symbolleisten der anderen Bilder.

Ergebnis

Die Schaltflächen aller Symbolleisten sind mit den gewünschten Systemfunktionen zum Bildwechsel projektiert.

Weitere Schritte

Folgende Schritte sind noch notwendig, um die Symbolleisten für den Bildwechsel zu nutzen:

- Konfigurationsdateien mit Symbolleisten anlegen
- Konfigurationsdateien im Bildfenster projektieren

Siehe auch

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Beispiel: Konfigurationsdatei anlegen

Einleitung

In einer Konfigurations legen Sie fest, welches Menü und welche Symbolleisten mit dieser Datei in Bildern oder Bildfenstern angelegt werden.

Voraussetzung

- Die Symbolleisten für alle Bilder sind angelegt.
- Für jede Symbolleiste sind Schalflächen angelegt.
- Der Editor "Menüs & Symbolleisten" ist geöffnet.
- Die Registerkarte "Konfigurationen" ist geöffnet

Konfiguration festlegen

1. Klicken Sie in der Tabelle "Konfigurationen" auf "Hinzufügen".
   Eine neue Konfiguration wird angelegt.
2. Geben Sie in der Spalte "Name" den Namen "Konfiguration_Startbild" ein.
3. Wählen Sie in der Tabelle "Symbolleisten" die Symbolleiste "Symbolleiste_Startbild".
4. Wählen Sie unter "Ausrichtung" die Einstellung "Unten".
   Die Symbolleiste wird im Bildfenster am unteren Rand des Anzeigebereichs dargestellt.
5. Um die Symbolleiste mit einer Bedienberechtigung in Runtime zu schützen, wählen Sie eine Berechtigung.
6. Wiederholen Sie das Vorgehen für jedes Bildfenster, dass Sie anlegen wollen.

Ergebnis

Die Konfigurationsdateien zur Anzeige der Symbolleisten für alle Bildfenster sind angelegt.

Weitere Schritte

Folgende Schritte sind noch notwendig, um die Symbolleisten für den Bildwechsel zu nutzen:

- Konfigurationsdateien im Bildfenster projektieren

Siehe auch

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

Beispiel: Konfigurationsdatei im Bildfenster projektieren

Einleitung

Um von jedem Bild aus zu den weiteren Bildern zu wechseln, legen Sie in einem Übersichtsbild so viele Bildfenster an, wie Sie zur Darstellung Ihres kompletten Prozesses benötigen. In jedes Bildfenster wird sowohl das entsprechende Bild eingefügt als auch die zugehörige Konfiguration projektiert.

Voraussetzung

- Bilder sind angelegt.
- Eine Konfigurationsdatei für jedes Bild ist angelegt.

Konfigurationsdatei in einem Bildfenster projektieren

1. Doppelklicken Sie im Projektfenster auf "Bilder > Bild hinzufügen". Ein neues Bild wird angelegt.
2. Geben Sie diesem Bild den Namen "Übersichtsbild_Projekt".
3. Ziehen Sie per Drag&Drop das Objekt "Bildfenster" in das Bild ein.
4. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Titel" einen Namen für die Titelleiste ein.
5. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Name" ein Bild aus, das im Bildfenster angezeigt wird.
6. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Konfigurationen" die Konfiguration fest.
7. Wiederholen Sie das Vorgehen für alle Bilder und Konfigurationen.

Ergebnis

Ein Übersichtbild mit Bildfenstern ist angelegt. Für jedes Bildfenster sind ein Bild und eine Konfiguration festgelegt.

Siehe auch

Beispiel: Symbolleisten und Schaltflächen projektieren (Seite 331)

1.9 Anzeige- und Bedienobjekte

1.9.1 Geräteabhängigkeit der Objekte

1.9.1.1 Objekte für Basic Panels

Verfügbarkeit der Anzeige- und Bedienobjekte für Basic Panels

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren, verwendet werden können. Die nachfolgende Tabelle zeigt die Verfügbarkeit der Anzeige- und Bedienobjekte auf Basic Panels.

Auf den Geräten mit der Geräteversion älter als V13 projektieren Sie einfache Anzeigeobjekte. Auf den Geräten mit der Geräteversion V13 oder neuer projektieren Sie tabellenbasierte Anzeigeobjekte.

Übersicht

	KP300 Basic KP400 Basic	KTP400 Basic KTP600 Basic KTP1000 Basic TP1500 Basic	KTP700 Basic PN KTP900 Basic	KTP700 Basic DP KTP1200 Basic DP	KTP1200 Basic PN
Balken	ja	ja	ja	ja	ja
Benutzeranzeige	ja	ja	ja [1)]	ja [1)]	ja [1)]
Datum/Uhrzeit-Feld	ja	ja	ja	ja	ja
E/A-Feld	ja	ja	ja	ja	ja
Ellipse	ja	ja	ja	ja	ja
Grafikanzeige	ja	ja	ja	ja	ja
Grafisches E/A-Feld	ja	ja	ja	ja	ja
Hilfeindikator	ja	nein	nein	nein	nein
HTML-Browser	nein	nein	ja	nein	ja
Kreis	ja	ja	ja	ja	ja
Kurvenanzeige	ja	ja	ja [1)]	ja [1)]	ja [1)]
Linie	ja	ja	ja	ja	ja
Meldeanzeige Meldefenster	ja	ja	ja [1)]	ja [1)]	ja [1)]
Meldeindikator	ja	ja	ja	ja	ja
Rechteck	ja	ja	ja	ja	ja
Rezepturanzeige	ja	ja	ja [1)]	ja [1)]	ja [1)]
Schaltfläche	ja	ja	ja	ja	ja
Schalter	ja	ja	ja	ja	ja
Symbolisches E/A-Feld	ja	ja	ja	ja	ja

	KP300 Basic KP400 Basic	KTP400 Basic KTP600 Basic KTP1000 Basic TP1500 Basic	KTP700 Basic PN KTP900 Basic	KTP700 Basic DP KTP1200 Basic DP	KTP1200 Basic PN
System-Diagnoseanzeige	ja	ja	ja [1]	ja [1]	ja [1]
Textfeld	ja	ja	ja	ja	ja

1) Die Objekte werden auf den Panels als tabellenbasierte Objekte angezeigt.

Siehe auch

Objekte für Comfort Panels (Seite 338)

Objekte für Mobile Panels (Seite 340)

Objekte für WinCC Runtime Advanced (Seite 343)

Objekte für WinCC Runtime Professional (Seite 344)

1.9.1.2 Objekte für Comfort Panels

Verfügbarkeit der Anzeige- und Bedienobjekte für Comfort Panels

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren, verwendet werden können. Die nachfolgende Tabelle zeigt die Verfügbarkeit der Anzeige- und Bedienobjekte auf Comfort Panels.

Übersicht

	KP400 Comfort KTP400 Comfort	KP700 Comfort TP700 Comfort	KP900 Comfort TP900 Comfort	KP1200 Comfort TP1200 Comfort	KP1500 Comfort TP1500 Comfort	TP1900 Comfort TP2200 Comfort
Balken	ja	ja	ja	ja	ja	ja
Benutzeranzeige	ja	ja	ja	ja	ja	ja
Beobachtungstabelle	ja	ja	ja	ja	ja	ja
Datum/Uhrzeit-Feld	ja	ja	ja	ja	ja	ja
E/A-Feld	ja	ja	ja	ja	ja	ja
Ellipse	ja	ja	ja	ja	ja	ja
f(x)-Kurvenanzeige	ja	ja	ja	ja	ja	ja
Funktionstaste	nur auf Tastengeräten					
Grafikanzeige	ja	ja	ja	ja	ja	ja
Grafisches E/A-Feld	ja	ja	ja	ja	ja	ja

	KP400 Comfort KTP400 Comfort	KP700 Comfort TP700 Comfort	KP900 Comfort TP900 Comfort	KP1200 Comfort TP1200 Comfort	KP1500 Comfort TP1500 Comfort	TP1900 Comfort TP2200 Comfort
GRAPH-Übersicht	ja	ja	ja	ja	ja	ja
Handrad	nein	nein	nein	nein	nein	nein
Hilfeindikator	nein	nein	nein	nein	nein	nein
HTML-Browser	nein	ja	ja	ja	ja	ja
Kamera-Anzeige	ja	ja	ja	ja	ja	ja
Kreis	ja	ja	ja	ja	ja	ja
Kriterienanalyse-Anzeige	ja	ja	ja	ja	ja	ja
Kurvenanzeige	ja	ja	ja	ja	ja	ja
Ladezustand	nein	nein	nein	nein	nein	nein
Leuchtdrucktaster	nein	nein	nein	nein	nein	nein
Linie	ja	ja	ja	ja	ja	ja
Media Player	ja	ja	ja	ja	ja	ja
Meldeanzeige Meldefenster	ja	ja	ja	ja	ja	ja
Meldeindikator	ja	ja	ja	ja	ja	ja
PDF-Anzeige	ja	ja	ja	ja	ja	ja
PLC-Code-Anzeige	nein	ja	ja	ja	ja	ja
Polygon	ja	ja	ja	ja	ja	ja
Polygonzug	ja	ja	ja	ja	ja	ja
ProDiag-Übersicht	ja	ja	ja	ja	ja	ja
Rechteck	ja	ja	ja	ja	ja	ja
Rezepturanzeige	ja	ja	ja	ja	ja	ja
Schaltfläche	ja	ja	ja	ja	ja	ja
Schalter	ja	ja	ja	ja	ja	ja
Schieberegler	ja	ja	ja	ja	ja	ja
Schlüsselschalter	nein	nein	nein	nein	nein	nein
Sm@rtClient-Anzeige	ja	ja	ja	ja	ja	ja
Symbolbibliothek	ja	ja	ja	ja	ja	ja
Symbolisches E/A-Feld	ja	ja	ja	ja	ja	ja
System-Diagnoseanzeige	ja	ja	ja	ja	ja	ja

	KP400 Comfort KTP400 Comfort	KP700 Comfort TP700 Comfort	KP900 Comfort TP900 Comfort	KP1200 Comfort TP1200 Comfort	KP1500 Comfort TP1500 Comfort	TP1900 Comfort TP2200 Comfort
System-Diagnosefenster	ja	ja	ja	ja	ja	ja
Textfeld	ja	ja	ja	ja	ja	ja
Uhr	ja	ja	ja	ja	ja	ja
Wirkbereichs-Signal	nein	nein	nein	nein	nein	nein
Wirkbereichs-Name	nein	nein	nein	nein	nein	nein
Wirkbereichs-Name (RFID)	nein	nein	nein	nein	nein	nein
WLAN-Empfang	nein	nein	nein	nein	nein	nein
Zeigerinstrument	ja	ja	ja	ja	ja	ja
Zonen-Name	nein	nein	nein	nein	nein	nein
Zonen-Signal	nein	nein	nein	nein	nein	nein

Siehe auch

Objekte für Basic Panels (Seite 337)

1.9.1.3 Objekte für Mobile Panels

Verfügbarkeit der Anzeige- und Bedienobjekte für Mobile Panels

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren, verwendet werden können. Die nachfolgende Tabelle zeigt die Verfügbarkeit der Anzeige- und Bedienobjekte auf Mobile Panels.

Übersicht

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP700 Mobile KTP700F Mobile KTP900 Mobile KTP900F Mobile	KTP400F Mobile
Balken	ja	ja	ja	ja	ja	ja	ja	ja
Benutzeranzeige	ja	ja	ja	ja	ja	ja	ja	ja
Beobachtungstabelle	ja	ja	ja	ja	ja	ja	ja	ja
Datum/Uhrzeit-Feld	ja	ja	ja	ja	ja	ja	ja	ja
E/A-Feld	ja	ja	ja	ja	ja	ja	ja	ja
Ellipse	ja	ja	ja	ja	ja	ja	ja	ja
f(x)-Kurvenanzeige	nein	nein	nein	nein	nein	nein	ja	ja
Funktionstaste	ja	ja	ja	ja	ja	ja	ja	ja
Grafikanzeige	ja	ja	ja	ja	ja	ja	ja	ja
Grafisches E/A-Feld	ja	ja	ja	ja	ja	ja	ja	ja
GRAPH-Übersicht	nein	nein	nein	nein	nein	nein	ja	ja
Handrad	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	nein	nein
Hilfeindikator	nein	nein	nein	nein	nein	nein	nein	nein
HTML-Browser	nein	nein	nein	nein	nein	nein	ja	ja
Kamera-Anzeige	nein	nein	nein	nein	nein	nein	ja	ja
Kreis	ja	ja	ja	ja	ja	ja	ja	ja
Kriterienanalyse-Anzeige	nein	nein	nein	nein	nein	nein	ja	ja
Kurvenanzeige	ja	ja	ja	ja	ja	ja	ja	ja
Ladezustand	nein	nein	nein	ja	ja	ja	nein	nein
Leuchtdrucktaster	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]
Linie	ja	ja	ja	ja	ja	ja	ja	ja
Media Player	nein	nein	nein	nein	nein	nein	nein	nein
Meldeanzeige Meldefenster	ja	ja	ja	ja	ja	ja	ja	ja
Meldeindikator	ja	ja	ja	ja	ja	ja	ja	ja
PDF-Anzeige	nein	nein	nein	nein	nein	nein	ja	ja
PLC-Code-Anzeige	nein	nein	nein	nein	nein	nein	ja	nein
Polygon	ja	ja	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP700 Mobile KTP700F Mobile KTP900 Mobile KTP900F Mobile	KTP400F Mobile
Polygonzug	ja	ja	ja	ja	ja	ja	ja	ja
ProDiag-Übersicht	nein	nein	nein	nein	nein	nein	ja	ja
Rechteck	ja	ja	ja	ja	ja	ja	ja	ja
Rezepturanzeige	ja	ja	ja	ja	ja	ja	ja	ja
Schaltfläche	ja	ja	ja	ja	ja	ja	ja	ja
Schalter	ja	ja	ja	ja	ja	ja	ja	ja
Schieberegler	ja	ja	ja	ja	ja	ja	ja	ja
Schlüsselschalter	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]	ja [1)]
Sm@rtClient-Anzeige	nein	ja	ja	ja	ja	ja	ja	ja
Symbolbibliothek	ja	ja	ja	ja	ja	ja	ja	ja
Symbolisches E/A-Feld	ja	ja	ja	ja	ja	ja	ja	ja
System-Diagnoseanzeige	nein	nein	nein	nein	nein	nein	ja	ja
System-Diagnosefenster	nein	nein	nein	nein	nein	nein	ja	ja
Textfeld	ja	ja	ja	ja	ja	ja	ja	ja
Uhr	nein	nein	ja	ja	ja	ja	ja	ja
Wirkbereichs-Signal	nein	nein	nein	nein	ja	nein	nein	nein
Wirkbereichs-Name	nein	nein	nein	nein	ja	nein	nein	nein
Wirkbereichs-Name (RFID)	nein	nein	nein	nein	nein	ja	nein	nein
WLAN-Empfang	nein	nein	nein	ja	ja	ja	nein	nein
Zeigerinstrument	ja	ja	ja	ja	ja	ja	ja	ja
Zonen-Name	nein	nein	nein	ja	ja	nein	nein	nein
Zonen-Signal	nein	nein	nein	ja	ja	nein	nein	nein
1) optionales Bedienelement								

Siehe auch

Objekte für Basic Panels (Seite 337)

1.9.1.4 Objekte für WinCC Runtime Advanced

Verfügbarkeit der Anzeige- und Bedienobjekte für WinCC Runtime Advanced

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren, verwendet werden können. Die nachfolgende Tabelle zeigt die Verfügbarkeit der Anzeige- und Bedienobjekte für WinCC Runtime Advanced.

Übersicht

	WinCC Runtime Advanced
Balken	ja
Benutzeranzeige	ja
Beobachtungstabelle	ja
Datum/Uhrzeit-Feld	ja
E/A-Feld	ja
Ellipse	ja
f(x)-Kurvenanzeige	ja
Funktionstaste	nein
Grafikanzeige	ja
Grafisches E/A-Feld	ja
GRAPH-Übersicht	ja
Handrad	nein
Hilfeindikator	nein
HTML-Browser	ja
Kamera-Anzeige	ja
Kreis	ja
Kriterienanalyse-Anzeige	ja
Kurvenanzeige	ja
Ladezustand	nein
Leuchtdrucktaster	nein
Linie	ja
Media Player	nein
Meldeanzeige Meldefenster	ja
Meldeindikator	ja
PDF-Anzeige	ja
PLC-Code-Anzeige	ja
Polygon	ja
Polygonzug	ja
ProDiag-Übersicht	ja
Rechteck	ja
Rezepturanzeige	ja

	WinCC Runtime Advanced
Schalter	ja
Schaltfläche	ja
Schieberegler	ja
Schlüsselschalter	nein
Sm@rtClient-Anzeige	ja
Symbolbibliothek	ja
Symbolisches E/A-Feld	ja
System-Diagnoseanzeige	ja
System-Diagnosefenster	ja
Textfeld	ja
Uhr	ja
Wirkbereichs-Signal	nein
Wirkbereichs-Name	nein
Wirkbereichs-Name (RFID)	nein
WLAN-Empfang	nein
Zeigerinstrument	ja
Zonen-Name	nein
Zonen-Signal	nein

Siehe auch

Objekte für Basic Panels (Seite 337)

Kamera-Anzeige (Seite 409)

PDF-Anzeige (Seite 457)

1.9.1.5 Objekte für WinCC Runtime Professional

Verfügbarkeit der Anzeige- und Bedienobjekte

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren, verwendet werden können. Die nachfolgenden Tabellen zeigen die Verfügbarkeit der Anzeige- und Bedienobjekte für WinCC Runtime Professional.

Übersicht

Tabelle 1-2 WinCC Runtime Professional

	WinCC Runtime Professional
Balken	ja
Benutzeranzeige	ja
Beobachtungstabelle	nein

	WinCC Runtime Professional
Bildfenster	ja
Bildlaufleiste	ja
Datum/Uhrzeit-Feld	nein
Doppel-T-Stück	ja
Druckauftrag/Skriptdiagnose	ja
E/A-Feld	ja
Editierbares Textfeld	ja
Ellipse	ja
Ellipsenbogen	ja
Ellipsensegment	ja
Funktionstaste	nein
f(x)-Kurvenanzeige	ja
f(t)-Kurvenanzeige	ja
Grafikanzeige	ja
Grafisches E/A-Feld	ja
GRAPH-Übersicht	ja
HTML-Browser	ja
Kamera-Anzeige	nein
Kanaldiagnose-Anzeige	ja
Kombinationsfeld	ja
Kontrollkästchen	ja
Kreis	ja
Kreisbogen	ja
Kreissegment	ja
Kriterienanalyse-Anzeige	ja
Kurvenanzeige	nein
Ladezustand	nein
Leuchtdrucktaster	nein
Linie	ja
Listenfeld	ja
Media Player	ja
Meldeanzeige	ja
Meldeindikator	nein
Optionsschaltfläche	ja
PLC-Code-Anzeige	ja
Polygon	ja
Polygonzug	ja
ProDiag-Übersicht	ja
Rechteck	ja
Rezepturanzeige	ja
Rundschaltfläche	ja

	WinCC Runtime Professional
Rohr	ja
Rohrbogen	ja
Schalter	nein
Schaltfläche	ja
Schieberegler	ja
Sm@rtClient-Anzeige	nein
Speicherplatzanzeige	ja
Symbolbibliothek	ja
Symbolisches E/A-Feld	ja
System-Diagnoseanzeige	ja
System-Diagnosefenster	nein
Tabellenanzeige	ja
Textfeld	ja
T-Stück	ja
Uhr	ja
Verbinder	ja
Wertetabelle	ja
Wirkbereichs-Name	nein
Wirkbereichs-Name (RFID)	nein
Wirkbereichs-Signal	nein
WLAN-Empfang	nein
Zeigerinstrument	ja
Zonen-Name	nein
Zonen-Signal	nein

Siehe auch

Objekte für Basic Panels (Seite 337)

1.9.2 Objekte

1.9.2.1 Balken

Verwendung

Mit dem Objekt "Balken" stellen Sie Variablen grafisch dar. Die Balkenanzeige kann mit einer Werteskala beschriftet werden.

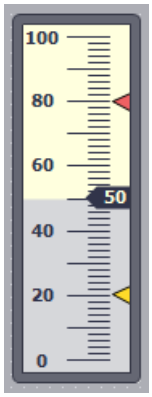

Darstellung

Sie ändern im Inspektorfenster die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Farbverlauf: Legt die Darstellung des Farbumschlags beim Überschreiten von Grenzwerten fest.
- Grenzwertlinie, Grenzwertmarkierung anzeigen: Zeigt den projektierten Grenzwert als Linie oder Markierung an.
- Balkensegmente definieren: Legt die Einteilung der Balkenskala fest.
- Skaleneinteilung festlegen: Legt die Unterteilungen, Teilstriche und Intervalle einer Balkenskala fest.

Farbverlauf

Sie legen im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" die Darstellung des Farbumschlags fest.

Farbverlauf	Beschreibung
"Segmentweise"	Wenn eine bestimmte Grenze erreicht wurde, färbt sich der Balken segmentweise. Mit segmentweiser Darstellung visualisieren Sie z.B. welche Grenzen von dem angezeigten Wert überschritten werden.
"Ganzer Balken"	Wenn eine bestimmte Grenze erreicht wurde, färbt sich der gesamte Balken.

Grenzwertlinien und Grenzwertmarkierung anzeigen

Sie zeigen mit der Eigenschaft "Linien" und "Markierungen" den projektierten Grenzwert im Balken in Runtime als Linie oder als Markierung an:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Aktivieren Sie "Linien" und "Markierungen".

Balkensegmente definieren

Sie legen mit der Eigenschaft "Unterteilungen" die Anzahl der Segmente fest, in die der Balken durch die Hauptteilstriche der Skala unterteilt wird.

Sie definieren mit der Eigenschaft "Intervall" den Abstand zwischen den Hauptteilstrichen. Der Wert wird als Wertdifferenz zwischen zwei benachbarten Hauptteilstrichen angegeben:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Skalen".
2. Aktivieren Sie "Skala anzeigen".
3. Wählen Sie für "Einstellungen > Unterteilungen" den entsprechenden Wert.
4. Wählen Sie für "Einstellungen > Beschriftung Teilstriche" den entsprechenden Wert.
5. Wählen Sie für "Großes Intervall > Intervall" den entsprechenden Wert.

Grenzen/Bereiche konfigurieren

Sie können Grenzen und Bereiche farblich unterschiedlich darstellen. Die Farben legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Grenzen/Bereiche".
2. Aktivieren Sie die Grenzen/Bereiche, die in Runtime angezeigt werden sollen
3. Ändern Sie bei Bedarf die voreingestellten Farben für die Grenzen/Bereiche.

Hinweis

Wenn Sie die Option "Bereiche von Variablen anzeigen" aktivieren, können Sie im Schieberegler bis zu fünf Bereiche darstellen, deren Werte über eine Prozessvariable vorgegeben werden. Die Werte für die Bereiche definieren Sie an einer Prozessvariablen, die Sie mit dem Bildobjekt verschalten.

Die Option "Bereiche von Variablen anzeigen" steht für Comfort Panels, KTP Mobile Panels und RT Advanced zur Verfügung.

Hinweis

Wenn die Option "Bereiche von Variablen anzeigen" aktiviert ist, hat die Einstellung "Farbverlauf Segmentweise" unter "Eigenschaften > Gestaltung > Balken" in Runtime keine Wirkung.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.2 Balken

Verwendung

Mit dem Objekt "Balken" stellen Sie Variablen grafisch dar. Die Balkenanzeige kann mit einer Werteskala beschriftet werden.

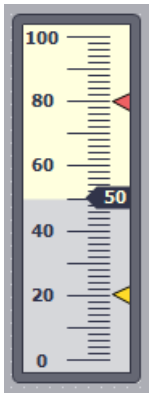

Darstellung

Sie ändern im Inspektorfenster die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Farbverlauf: Legt die Darstellung des Farbumschlags beim Überschreiten von Grenzwerten fest.
- Grenzwertmarkierung anzeigen: Zeigt den projektierten Grenzwert als Pfeil an.
- Balkensegmente definieren: Legt die Einteilung der Balkenskala fest.
- Skaleneinteilung festlegen: Legt die Position des Nullpunkts einer Balkenskala fest.

Farbverlauf

Sie legen im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" die Darstellung des Farbumschlags fest.

Farbverlauf	Beschreibung
"Segmentweise"	Wenn eine bestimmte Grenze erreicht wurde, färbt sich der Balken segmentweise. Mit segmentweiser Darstellung visualisieren Sie z.B. welche Grenzen von dem angezeigten Wert überschritten werden.
"Ganzer Balken"	Wenn eine bestimmte Grenze erreicht wurde, färbt sich der gesamte Balken.

Grenzwertmarkierung anzeigen

Sie zeigen mit der Eigenschaft "Markierungen" den projektierten Grenzwert im Balken in Runtime als Pfeil an:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Aktivieren Sie "Markierungen".

Balkensegmente definieren

Sie legen mit der Eigenschaft "Skalen" die Anzahl der Unterteilungen fest, in die der Balken durch die Hauptteilstriche der Skala unterteilt wird.

Sie definieren mit der Eigenschaft "Intervall" den Abstand zwischen den Hauptteilstrichen. Der Wert wird als Wertdifferenz zwischen zwei benachbarten Hauptteilstrichen angegeben:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Skalen".
2. Aktivieren Sie "Skala anzeigen".
3. Geben Sie für "Großes Intervall > Intervall" den entsprechenden Wert ein.

Hinweis

Ein Intervall verändern Sie nur, wenn "Automatisch anpassen" deaktiviert ist.

Skalenteilung festlegen

Sie legen mit der Eigenschaft "Nullpunkt" die Position des Nullpunkts der Balkenskala fest. Sie geben den Wert in Prozent an. Der Prozentwert bezieht sich auf den Abstand der Skalen-Endwerte. Für einen Wert von zum Beispiel 0 % wird der Nullpunkt auf Höhe des geringsten Skalenwerts dargestellt. Sie können den Nullpunkt auch außerhalb des dargestellten Bereichs legen.

Die Eigenschaft "Nullpunkt" wird nur ausgewertet, wenn die Eigenschaft "Balkenskalierung" den Wert "Automatisch anpassen" besitzt:

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Skalen
2. Aktivieren Sie "Automatisch anpassen".
3. Tragen Sie für "Startwerte > Nullpunkt" den gewünschten Wert ein.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.3 Benutzeranzeige (ab V13)

Verwendung

Das Objekt "Benutzeranzeige" dient der Einrichtung und Verwaltung von Benutzern und Berechtigungen. Mit der Benutzeranzeige legen Sie in Runtime z. B. neue Benutzer an und ordnen Sie einer Benutzergruppe zu.

Auf den Bediengeräten mit der Geräteversion V13 oder höher steht Ihnen zur Verwaltung von Benutzern und Berechtigungen die tabellenbasierte Benutzeranzeige zur Verfügung.

Benutzer	Kennwort	Gruppe	Abmeldezeit

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl Zeilen: Legt die maximale Anzahl der sichtbaren Einträge fest.
- Spalten verschiebbar: Legt fest, ob der Bediener die Reihenfolge der Spalten in Runtime ändern kann.

Außerdem gestalten Sie im Inspektorfenster den Rahmen, das Füllmuster und die Farben des Tabellenkopfs.

Anzahl Zeilen

Die Anzahl der Zeilen in der Benutzeranzeige, die in Runtime dargestellt werden, wird im Inspektorfenster festgelegt. Die Einstellung der Anzahl der Zeilen wirkt nur, wenn "Objekt an Inhalt anpassen" aktiviert ist.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Anzeige".
2. Geben Sie unter "Anzahl Zeilen" einen ganzzahligen Zahlenwert ein.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
4. Aktivieren Sie "Objekt an Inhalt anpassen".

Spalten verschiebbar

Mit der Eigenschaft "Spalten verschiebbar" ändern Sie in Runtime die Reihenfolge der Spalten.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Spalten verschiebbar".

Spaltenbreite

Im Inspektorfenster ändern Sie die Breite der Spalten, die in Runtime angezeigt werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Spalten" ganzzahlige Zahlenwerte für die Breite der Spalten ein.

Hinweis

Die Spalte "Kennwort" ist eine dynamische Spalte, die sich der äußeren Größe des Objekts anpasst, sodass der Inhalt des Objekts die komplette Breite ausfüllt. Die von Ihnen angegebene Spaltenbreite wird als Minimum-Größe übernommen. Dabei wird bei einer dynamischen Spalte die Minimum-Größe geändert, allerdings wird die Breite wieder dynamisch angezeigt, um die Anzeige komplett auszufüllen.

1.9.2.4 Benutzeranzeige, einfach

Verwendung

Das Objekt "Benutzeranzeige" dient der Einrichtung und Verwaltung von Benutzern und Berechtigungen.

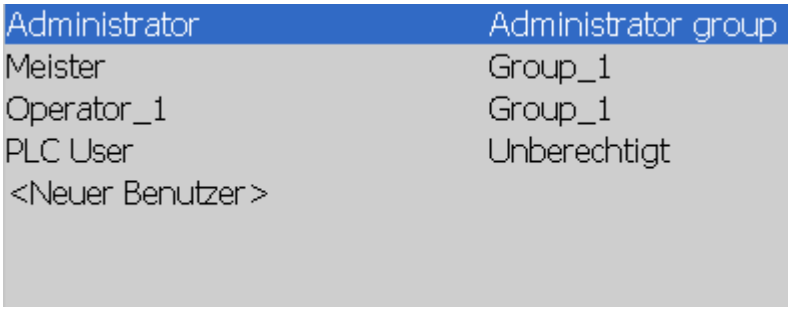

Hinweis

Verwenden Sie die Benutzeranzeige nicht in einer Gruppe.

Hinweis

Das Objekt "Einfache Benutzeranzeige" kann nicht mit einem Skript dynamisiert werden.

Hinweis

Das Objekt "Einfache Benutzeranzeige" projektieren Sie auf den Geräten mit der Geräteversion 12.0.0.0 oder älter.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl Zeilen: Legt die maximale Anzahl der sichtbaren Einträge fest.

Anzahl Zeilen

Die Anzahl der Zeilen in der Benutzeranzeige, die in Runtime dargestellt werden, wird im Inspektorfenster festgelegt. Die Einstellung der Anzahl der Zeilen wirkt nur, wenn die Eigenschaft "Objekt an Inhalt anpassen" aktiviert ist.

1. Klicken Sie im Inspektorfenster auf die Gruppe "Eigenschaften > Eigenschaften > Anzeige".
2. Geben Sie unter "Anzahl Zeilen" einen ganzzahligen Zahlenwert ein.
3. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
4. Aktivieren Sie "Objekt an Inhalt anpassen".

Darstellung in Runtime

Die Darstellung ist abhängig von den Berechtigungen:

- Beim Administrator oder einem Benutzer mit der Berechtigung zur Benutzerverwaltung werden in der Benutzeranzeige alle auf dem Bediengerät vorhandenen Benutzer angezeigt.
- Bei einem Benutzer ohne die Berechtigung zur Benutzerverwaltung wird nur der eigene Benutzereintrag angezeigt.

Bedienung

Abhängig von der Projektierung können Sie:

- Benutzer verwalten, z. B. neu anlegen, löschen.
- Vorhandene Benutzerdaten ändern.
- Benutzerdaten exportieren oder importieren.

Hinweis

An einem Bediengerät ist die Anzahl auf 100 Benutzer und einem PLC-User begrenzt. Diese Grenze gilt nicht für PCs. An einem PC wird die maximale Anzahl der Benutzer durch den physikalischen Speicher begrenzt.

1.9.2.5 Benutzeranzeige, erweitert

Verwendung

Das Objekt "Benutzeranzeige" dient der Einrichtung und Verwaltung von Benutzern und Berechtigungen. Mit der Benutzeranzeige legen Sie in Runtime z. B. neue Benutzer an und ordnen sie einer Benutzergruppe zu.

Erweiterte Benutzeranzeige

Je nach Bediengeräten steht Ihnen zur Verwaltung von Benutzern und Berechtigungen die erweiterte bzw. die einfache Benutzeranzeige zur Verfügung.

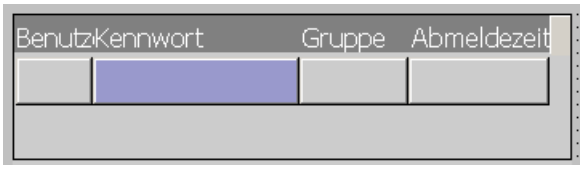

Anzeigetyp ändern

Auf Bediengeräten mit einer Displaygröße größer als 6" können Sie sowohl die erweiterte Benutzeranzeige als auch eine vereinfachte Benutzeranzeige projektieren. Anzeigetyp steht zum Konfigurieren nur auf Geräten bis Geräteversion V13 zur Verfügung.

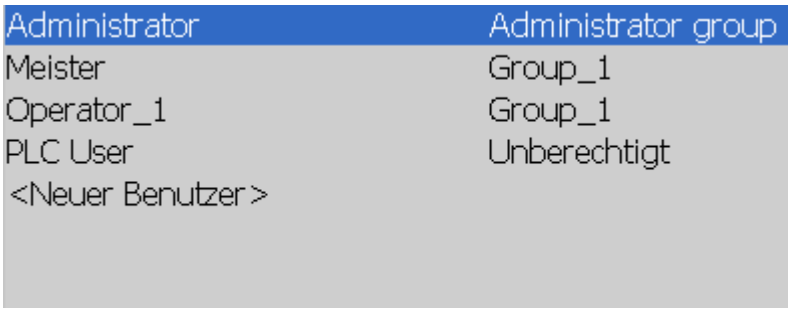

Um die entsprechende Benutzeranzeige zu projektieren, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Modus".
2. Wählen Sie "Vereinfacht" oder "Erweitert".

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl Zeilen: Legt die maximale Anzahl der sichtbaren Einträge fest.
- Spalten verschiebbar: Legt fest, ob der Bediener die Reihenfolge der Spalten in Runtime ändern kann.

Anzahl Zeilen

Die Anzahl der Zeilen in der Benutzeranzeige, die in Runtime dargestellt werden, wird im Inspektorfenster festgelegt. Die Einstellung der Anzahl der Zeilen wirkt nur, wenn "Objekt an Inhalt anpassen" aktiviert ist.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Anzeige".
2. Geben Sie unter "Anzahl Zeilen" einen ganzzahligen Zahlenwert ein.

3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
4. Aktivieren Sie "Objekt an Inhalt anpassen".

Spalten verschiebbar

Mit der Eigenschaft "Spalten verschiebbar" kann der Bediener in Runtime die Reihenfolge der Spalten ändern.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Spalten verschiebbar".

Diese Option steht Ihnen nur bei der komplexen Benutzeranzeige zur Verfügung.

Siehe auch

Benutzeranzeige (Seite 1402)

Benutzer anlegen (Seite 1404)

Benutzer mit der komplexen Benutzeranzeige verwalten (Seite 1406)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.6 Beobachtungstabelle

Verwendung

Mit dem Objekt "Beobachtungstabelle" projektieren Sie einen Editor, über den Sie in Runtime einzelne Adressbereiche einer SIMATIC S7 direkt bearbeiten können.

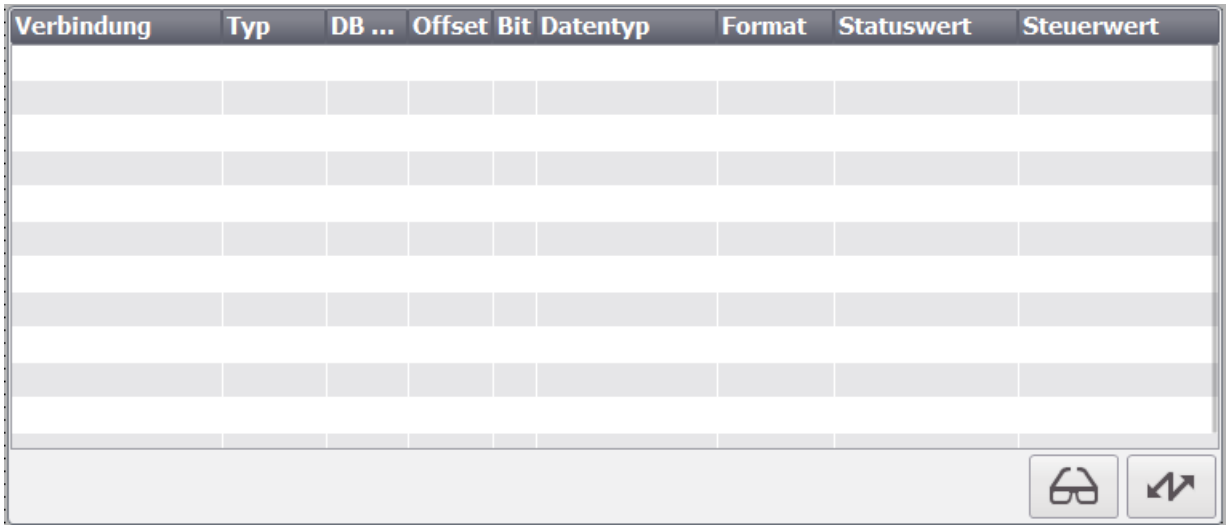

Hinweis

Das Objekt "Beobachtungstabelle" ist für folgende Steuerungen freigegeben:

- SIMATIC S7-300
- SIMATIC S7-400

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Steuerungstyp: Legt fest, für welchen Steuerungstyp im Objekt der Adressbereich in Runtime ausgegeben wird.
- Bedienelemente: Legt die Bedienelemente des Objekts fest.
- Sichtbare Spalten: Legt die in Runtime darzustellenden Spalten fest.
- Spalten verschiebbar: Legt fest, ob der Bediener die Reihenfolge der Spalten in Runtime ändern kann.

Bedienelemente

Die Bedienelemente, mit denen das Objekt "Beobachtungstabelle" in Runtime bedient werden kann, legen Sie im Inspektorfenster "Eigenschaften > Eigeschaften > Anzeige" im Bereich "Einstellungen" fest.

	Funktion
	Mit dieser Schaltfläche aktualisieren Sie die Anzeige in der Spalte Statuswert.
	Mit dieser Schaltfläche übernehmen Sie den neuen Wert in der Spalte Steuerwert. Der Steuerwert wird daraufhin in die Steuerung geschrieben.

Sichtbare Spalten

Die Spalten, die in Runtime angezeigt werden, werden im Inspektorfenster in der Gruppe " Allgemein " im Bereich "Sichtbare Spalten" festgelegt.

Spalte	Beschreibung
"Verbindung"	Die Steuerung, deren Adressbereiche angezeigt werden.
"Typ" "DB-Nummer" "Offset" "Bit"	Der Adressbereich des Operanden.
"Datentyp" "Format"	Der Datentyp des Operanden.

Spalte	Beschreibung
"Statuswert"	Der Wert, der aus der angegebenen Adresse des Operanden gelesen wurde.
"Steuerwert"	Der Wert, der in die angegebene Adresse des Operanden geschrieben wird.

Reihenfolge der Spalten

Mit der Eigenschaft "Reihenfolge der Spalten" kann der Bediener in Runtime die Reihenfolge der Spalten verändern.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Spalten".
2. Aktivieren Sie im Bereich "Eigenschaften der Spalten > Reihenfolge der Spalten".

Systemfunktionen

Für die Aktualisierung der Werte in der Beobachtungstabelle und die Übertragung der Werte in die Steuerung stehen Ihnen zwei Systemfunktionen zur Verfügung:

- StatusSteuernLeseWerte
- StatusSteuernSchreibeWerte

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.7 Bildfenster

Verwendung

Mit dem Objekt "Bildfenster" stellen Sie andere Bilder des Projekts im aktuellen Bild dar. Um zum Beispiel den Inhalt eines Bildfensters stetig zu aktualisieren, dynamisieren Sie das Objekt. Mit benutzerdefinierten Menüs und Symbolleisten fügen Sie dem Bildfenster spezifische Schaltflächen hinzu.

Sie können auch unabhängige Bildfenster losgelöst vom jeweiligen Bild verwenden. Bei entsprechender Hardwareausstattung und Unterstützung durch das Betriebssystem können Sie so auch mehrere Monitore ansteuern und Prozesse umfassender und differenzierter abbilden.

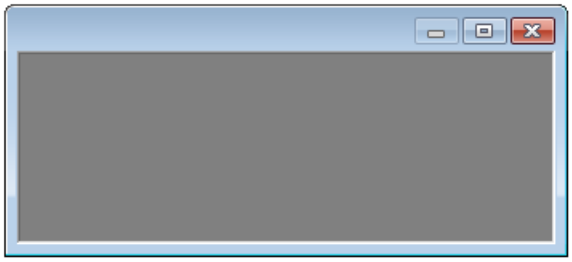

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Zoomfaktor: Legt fest, in welcher Größe das eingebettete Bild dargestellt wird.
- Bildausschnitt: Legt den Ausschnitt des eingebetteten Bildes fest, der im Bildfenster angezeigt wird. Wenn das eingebettete Bild größer ist als das Bildfenster, projektieren Sie Bildlaufleisten für das Bildfenster.
- Unabhängige Bildfenster: Legt fest, dass die Bildfenster unabhängig vom Bild dargestellt werden, in dem sie projektiert wurden.

Hinweis

Kaskadierung von Bildfenstern

Bildfenster können auch Bilder anzeigen, in denen wiederum Bildfenster enthalten sind. Die Darstellbarkeit von kaskadierten Bildfenstern in Runtime ist abhängig vom Betriebssystem. In der 32-Bit-Version können 22 kaskadierte Bildfenster angezeigt werden. In der 64-Bit-Version können 14 kaskadierte Bildfenster angezeigt werden.

Größe von eingebettetem Bild und Bildfenster abstimmen

Sie haben folgende Möglichkeiten, die Größe des eingebetteten Bildes mit der Größe des Bildfensters abzustimmen:

- Sie wollen das eingebettete Bild verkleinert darstellen.
  Stellen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den gewünschten Zoomfaktor ein.
- Sie wollen zu einem Ausschnitt des eingebetteten Bildes scrollen.
  Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Bildlaufleisten".
  Der Benutzer kann in Runtime zu Details des eingebetteten Bildes scrollen.
  Im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Position der Bildlaufleiste" verschieben Sie die Position der Bildlaufleiste horizontal oder vertikal.
- Sie passen das eingebettete Bild an die Größe des Bildfensters an oder umgekehrt.
  Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" eine der beiden Einstellungen "Fenster an Bild anpassen" oder "Bild an Fenster anpassen".
  Wenn Sie die Einstellung "Fenster an Bild anpassen" aktivieren, wird die die Größe des Bildfensters in Runtime an das eingebettete Bild angepasst und ist im Engineering System nicht mehr veränderbar.
  Für die Einstellung "Bild an Fenster anpassen" wählen Sie beim Skalierungsmodus zwischen zwei Optionen:
  - Einheitlich: Das eingebettete Bild wird im Bildfenster vollständig dargestellt. Bei abweichendem Seitenverhältnis füllt das Bild das Bildfenster nicht aus.
  - Füllen: Das Bild füllt das Bildfenster aus. Bei abweichendem Seitenverhältnis werden die Teile des Bildes abgeschnitten.

  Bei beiden Optionen wird das Seitenverhältnis beibehalten.

Bildausschnitt festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Offset" Werte zur horizontalen und vertikalen Verschiebung des Nullpunkts der Bildanzeige ein.
   Der gewünschte Ausschnitt wird im Bildfenster angezeigt.
3. Wenn Sie von dem definierten Ausschnitt zu anderen Bereichen des Bildes scrollen wollen, aktivieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung > Bildlaufleisten".

Unabhängige Bildfenster festlegen

1. Fügen Sie im Startbild ein Bildfenster ein.
2. Wählen Sie "Eigenschaften > Eigenschaften > Allgemein > Name" das Bild, das im Bildfenster angezeigt wird.
3. Aktivieren Sie "Eigenschaften > Eigenschaften > Allgemein > Unabhängige Bildfenster"
4. Wählen Sie "Position und Modus":
   - Wie projektiert: Das Bildfenster wird in der projektierten Größe und an der projektierten Position angezeigt.
   - Zentriert: Das Bildfenster wird in der projektierten Größe in zentraler Position angezeigt.
   - Maximiert: Das Bildfenster wird an die Monitorgröße angepasst.
5. Wenn Sie mehrere Monitore verwenden, wählen Sie "Monitornummer" den gewünschten Monitor für das Bildfenster.

   Hinweis

   Die Monitornummer bezieht sich auf die in Microsoft Windows eingestellte Anzahl der Monitore: "Systemsteuerung > Anzeige > Einstellungen > Anzeige"

6. Wiederholen Sie die Schritte 1 bis 5 für alle weiteren unabhängigen Bildfenster.

Variablen-Präfix festlegen

Um in Runtime auf Strukturinstanzen zuzugreifen, legen Sie ein Variablen-Präfix fest. Das Variablen-Präfix ist frei definierbar, muss aber mit dem Namen der Strukturinstanz übereinstimmen.

Hinweis

Dynamische Verschaltung von Variablen zur Laufzeit

Um Variablen dynamisch zur Laufzeit zu verschalten, verwenden Sie den Variablen-Präfix oder indirekt adressierten Variablen.

Vermeiden Sie die gleichzeitige Verwendung beider Methoden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Inhalt".
2. Geben Sie unter "Variablen-Präfix" eine der erstellten Strukturinstanzen mit einem Punkt hinter dem Namen ein z. B. "Motor1.".

In dem Bildfenster wird an einem Objekt die Variable "Temperatur" angefordert. Wenn dem Bildfenster ein Variablen-Präfix "Motor1." zugeordnet ist, dann wird die Variable "Motor1.Temperatur" angefordert.

Wenn das Variablen-Präfix für ein Bildfenster gesetzt ist, wird das Variablen-Präfix allen Variablen vorangestellt, die im anzuzeigenden Bild enthalten sind. Das gilt auch, wenn die Anforderung in einer Funktion erfolgt. Wenn eine Variable ohne Variablen-Präfix gelesen werden soll, müssen Sie dem Variablennamen den Zusatz "@NOTP::" voranstellen.

Hinweis

Verfügbarkeit des Variablen-Präfixes in Anzeige- und Bedienobjekten

Die Eigenschaft Variablen-Präfix steht nur für die Objekte "Schieberegler" und "Zeigerinstrument" zur Verfügung.

Das Variablen-Präfix wird von den Objekten der Palette "Controls" nicht unterstützt.

Hinweis

Wenn dem Bild ein Anwenderdatentyp zugewiesen ist, ist der „Variablen-Präfix“ nicht editierbar.

Hinweis

Im untergeordneten Bildfenster keinen zusätzlichen Variablen-Präfix festlegen

Wenn in einem referenzierten Bild eines Bildfenster ein Bildfenster projektiert ist, wird in dem untergeordneten Bildfenster der Variablen-Präfix des übergeordneten Bildfensters übernommen. Sie sollten im untergeordneten Bildfenster keinen zusätzlichen Variablen-Präfix festlegen, da in WinCC gebündelte Strukturvariablen nicht unterstützt werden. Die Interpretation "Variablenpräfix1.Variablenpräfix2.Variablenname" kann nicht für Dynamisierungen verwendet werden.

Strukturierte Prozessvariable festlegen

Alternativ zum Variablen-Präfix weisen Sie dem Objekt "Bildfenster" eine strukturierte Prozessvariable zu. Damit eine strukturierte Prozessvariable auswählbar ist, muss dem Bild im Bildfenster ein Anwenderdatentyp zugewiesen sein. Dieser Anwenderdatentyp muss in der Projektbibliothek angelegt sein.

1. Wenn dem Bild der HMI-Anwenderdatentyp zugewiesen ist, wählen Sie aus der Auswahlliste "Prozessvariable" eine Prozessvariable vom HMI-Anwenderdatentyp.
   Oder:
   Wenn dem Bild der PLC-Anwenderdatentyp zugewiesen ist, wählen Sie eine Prozessvariable, die mit einer PLC-Variable oder mit einem PLC-Datenblock verbunden ist.

Die ausgewählte Prozessvariable erscheint unter "Variablen-Präfix" und unter "Prozessvariable".

Hinweis

Die Eigenschaft "Prozessvariable" kann nicht dynamisiert werden. Verwenden Sie für die Dynamisierung die Eigenschaft "Variablen-Präfix".

Die Eigenschaft "Prozessvariable" kann nicht mithilfe von VB-Scripting oder C-Scripting adressiert werden. Verwenden Sie für VB-Scripting oder C-Scripting die Eigenschaft "Variablen-Präfix".

Hinweis

Kaskadierung von Bildfenstern

Bei kaskadierenden Bildfenstern muss dem Bild, in dem Sie das Bildfenster projektieren, ein Anwendertyp zugewiesen sein. In diesem Fall ist unter "Prozessvariable" sowohl eine Prozessvariable als auch ein Strukturelement des zugewiesenen Anwenderdatentyps auswählbar. Um den Anwenderdatentyp auswählen zu können, muss dieser in der Projektbibliothek angelegt sein.

Besonderheit bei der Sichtbarkeitsanimation

Wenn sie die Sichtbarkeit eines Bildfensters animieren wollen und das Bildfenster zuerst nicht sichtbar sein soll, deaktivieren Sie die Einstellung "Sichtbarkeit" unter "Eigenschaften > Verschiedenes".

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.8 Bildlaufleiste

Verwendung

Mit dem Objekt "Bildlaufleiste" steuern Sie Prozesse. Sie verändern mit der Bildlaufleiste zum Beispiel stufenlos Werte. Um eine Bildlaufleiste in den Prozess einzubinden, dynamisieren Sie die entsprechenden Attribute.

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Stil, und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften nach Bedarf an:

- Maximaler Wert und Minimaler Wert: Legt den oberen und den unteren Endwert der Skala fest.
- Ausrichtung: Legt die horizontale oder vertikale Ausrichtung fest.

Maximaler Wert und minimaler Wert

Sie legen im Inspektorfenster den oberen und unteren Endwert der Skala fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie unter "Maximum Statisch" und "Minimum Statisch" eine Zahl ein. Wenn Sie als Endwert der Skala eine Variable auswählen, dann ist die Zahl nicht mehr verfügbar.

Ausrichtung festlegen

Sie legen im Inspektorfenster fest, ob die Bildlaufleiste vertikal oder horizontal ausgerichtet ist.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie unter "Stil" die gewünschte Ausrichtung.

Siehe auch

Bedienmeldungen projektieren (Seite 862)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.9 Datum/Uhrzeit-Feld

Verwendung

Das Objekt "Datum/Uhrzeit-Feld" zeigt die Systemzeit und das Systemdatum an. Die Darstellung im "Datum/Uhrzeit-Feld" ist abhängig von der am Bediengerät eingestellten Sprache.

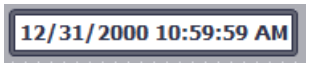

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Systemzeit anzeigen: Legt fest, dass die Systemzeit angezeigt wird.
- Variable verwenden: Legt fest, dass die Uhrzeit der verbundenen Variable angezeigt wird.
- Langes Datum/Uhrzeit-Format: Legt fest, in welchem Format Datum und Uhrzeit angezeigt werden.

Systemzeit anzeigen

Die Uhrzeit, die im "Datum/Uhrzeit-Feld" auf dem Bediengerät angezeigt wird, wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Format > Systemzeit".

Variable verwenden

Im Datum/Uhrzeit-Feld wird die Uhrzeit der verbundenen Variable angezeigt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Bereich "Format" eine Variable vom Datentyp "DateTime", z. B. eine interne Variable.

Langes Datum/Uhrzeit-Format

Die Darstellung von Datum und Uhrzeit wird im Inspektorfenster unter " Eigenschaften > Eigenschaften > Allgemein > Format" festgelegt.

Option	Beschreibung
"Aktiviert"	Datum und Uhrzeit werden vollständig angezeigt, z. B. "Sunday, December 31, 2000 10:59:59 AM"
"Deaktiviert"	Datum und Uhrzeit werden verkürzt angezeigt, z. B. "12/31/2000 10:59:59 AM"

Verhalten bei der Bedienung

Wenn der Bediener in Runtime bei der Eingabe von Werten die Syntax nicht beachtet oder unzulässige Werte eingibt, werden diese Eingaben nicht übernommen. Stattdessen wird im Datum-Uhrzeit-Feld wieder der ursprüngliche Wert (plus die inzwischen weitergelaufene Zeitspanne) angezeigt und eine Systemmeldung am Bediengerät ausgegeben.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.10 Doppel-T-Stück

Verwendung

Mit dem Objekt "Doppel-T-Stück" bilden Sie eine Rohrkreuzung nach. Um eine Rohranlage nachzubilden, verbinden Sie das Objekt "Doppel-T-Stück" mit weiteren Objekten z. B. Rohrbogen oder T-Stück.

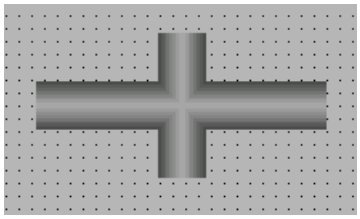

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Farbe und Breite des Objekts.

Siehe auch

Rohr (Seite 487)

Rohrbogen (Seite 487)

T-Stück (Seite 520)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.11 Druckauftrag/Skriptdiagnose

Verwendung

Sie verwenden das Objekt "Druckauftrag/Skriptdiagnose", um Inhalte aus den Editoren "Druckaufträge" und "Skripte" darzustellen. In Runtime werden Informationen an das Objekt "Druckauftrag/Skriptdiagnose" übergeben und damit eine Bedienung ermöglicht. Je nachdem welchen Fensterinhalt und welche Vorlage Sie wählen, können Sie sich z.B. eine Vorschau des Druckauftrags anzeigen lassen.

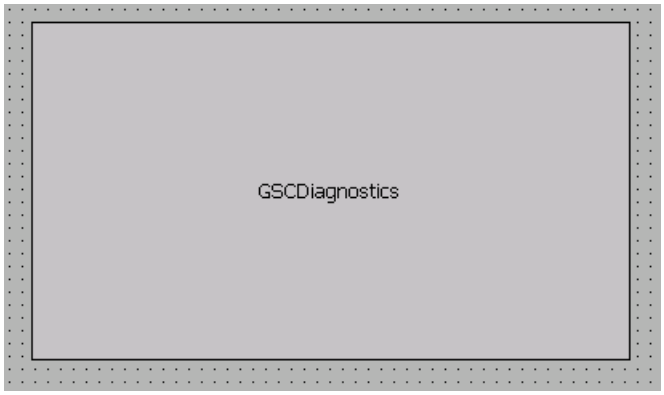

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Fensterinhalt: Legt die Applikation fest, die im Objekt "Druckauftrag/Skriptdiagnose" angezeigt wird.
- Vorlage: Legt die Vorlage fest, in welcher der Fensterinhalt im Objekt "Druckauftrag/Skriptdiagnose" angezeigt wird.

Vorlagen für die Verwendung von Funktionen

Für Funktionen sind die folgenden Vorlagen verfügbar:

- GSC – Diagnose
  Das Objekt "Druckauftrag/Skriptdiagnose" wird von Applikationen der Funktionen versorgt. Dargestellt werden die Ergebnisse des Diagnosesystems.

Vorlagen für die Verwendung von Druckaufträgen

Für den Fensterinhalt "Druckaufträge" sind die folgenden Vorlagen verfügbar:

- Alle Druckaufträge
  Das Objekt "Druckauftrag/Skriptdiagnose" wird von den Druckaufträgen versorgt. Die verfügbaren Druckaufträge werden als Liste dargestellt.
- Kontextmenü für alle Aufträge
  Das Objekt "Druckauftrag/Skriptdiagnose" wird von den Druckaufträgen versorgt. Die verfügbaren Druckaufträge werden als Liste dargestellt. Ein Kontextmenü ermöglicht folgende Aktionen:
  - Druckvorschau darstellen
  - Protokoll ausdrucken
- Aufträge Detailansicht
  Das Objekt "Druckauftrag/Skriptdiagnose" wird von den Druckaufträgen versorgt. Für die verfügbaren Druckaufträge gibt es ein Auswahlmenü. Für den gewählten Druckauftrag werden Detailinformationen angezeigt.
- Kontextmenü für ausgewählte Aufträge
  Das Objekt "Druckauftrag/Skriptdiagnose" wird von den Druckaufträgen versorgt. Die verfügbaren Druckaufträge werden als Liste dargestellt.
  Um Drückaufträge in der Liste anzuzeigen, aktivieren Sie im Editor Druckauftrag "Eigenschaften > Eigenschaften > Allgemein > Im Control Druckauftrag/Skriptdiagnose anzeigen". Ein Kontextmenü ermöglicht folgende Aktionen:
  - Druckvorschau darstellen
  - Protokoll ausdrucken

Fensterinhalt konfigurieren

Sie legen im Inspektorfenster den Fensterinhalt des Objekts fest:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Allgemein".
2. Wählen Sie den Fensterinhalt aus, z. B. "Druckaufträge".

Vorlage konfigurieren

Sie legen im Inspektorfenster die Vorlage für den Fensterinhalt des Objekts fest:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Allgemein".
2. Wählen Sie Vorlage aus.

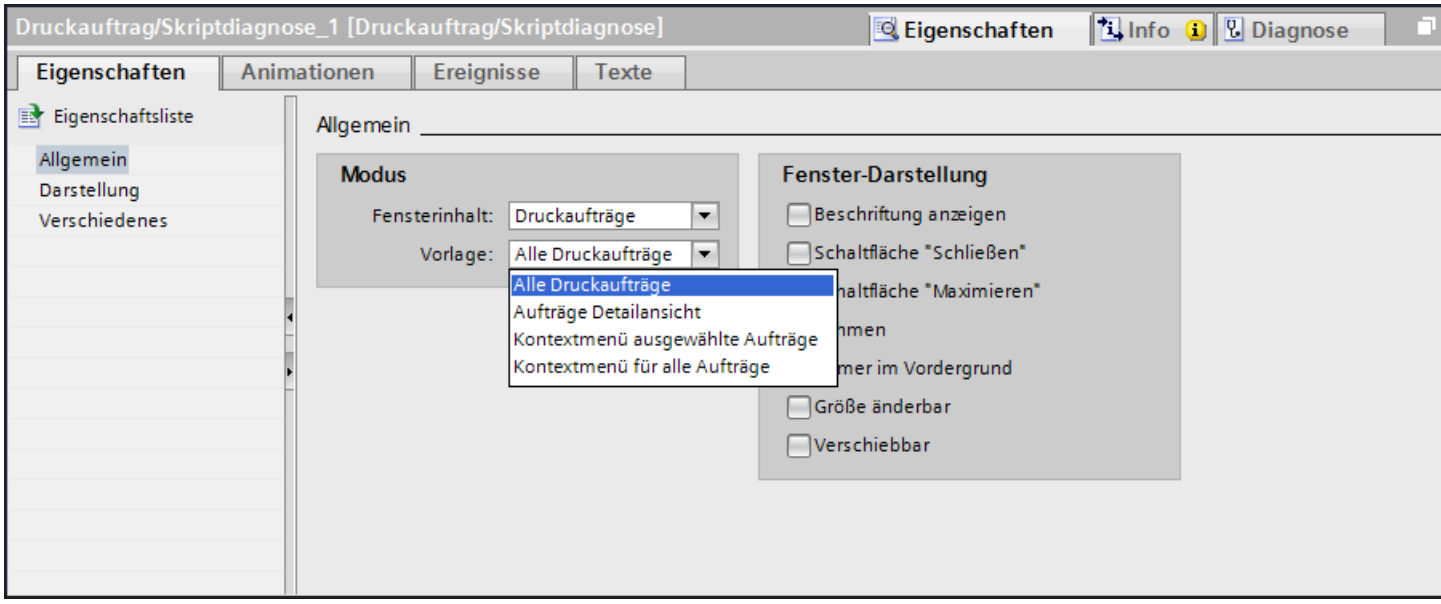

Fensterdarstellung in Runtime

Das Objekt "Druckauftrag/Skriptdiagnose" wird in Runtime als eigenständiges Fenster innerhalb eines Prozessbildes dargestellt. Sie passen die Merkmale der Fensterdarstellung mit Attributen unter "Fenster-Darstellung" an.

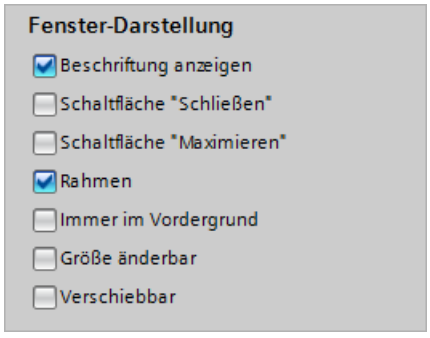

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.12 E/A-Feld

Verwendung

Das Objekt "E/A-Feld" dient der Eingabe und Anzeige von Prozesswerten.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Anzeigeformat: Legt das Anzeigeformat im E/A-Feld für die Eingabe und Ausgabe von Werten fest.
- Verdeckte Eingabe: Legt fest, ob der Eingabewert während der Eingabe normal oder verschlüsselt angezeigt wird.

Hinweis

Protokolle

In Protokollen geben E/A-Felder ausschließlich Daten aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Eingabe stehen nicht zur Verfügung, z. B. "Verdeckte Eingabe".

Modus

Das Verhalten des E/A-Feldes legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" fest.

Modus	Beschreibung
"Eingabe"	In das E/A-Feld können in Runtime nur Werte eingegeben werden.
"Eingabe/Ausgabe"	In das E/A-Feld können in Runtime Werte eingegeben und ausgegeben werden.
"Ausgabe"	Das E/A-Feld wird nur zur Ausgabe von Werten eingesetzt.

Anzeigeformat

Das Anzeigeformat für die Eingabe und Ausgabe von Werten legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Format" fest.

Anzeigeformat	
"Binär"	Eingabe und Ausgabe von Werten in binärer Schreibweise.
"Datum"	Eingabe und Ausgabe von Datumsangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.
"Datum/Uhrzeit"	Eingabe und Ausgabe von Datumsangaben und Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.

Anzeigeformat	
"Dezimal"	Eingabe und Ausgabe von Werten in dezimaler Schreibweise.
"Hexadezimal"	Eingabe und Ausgabe von Werten in hexadezimaler Schreibweise.
"Uhrzeit"	Eingabe und Ausgabe von Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.
"Zeichenkette"	Eingabe und Ausgaben von Zeichenketten.

Hinweis

Datenformate

Für Runtime Professional stehen nicht alle Datenformate zur Auswahl.

Verdeckte Eingabe

In Runtime kann die Eingabe normal oder verschlüsselt angezeigt werden, z. B. zur verdeckten Eingabe eines Kennworts. Bei der verdeckten Eingabe wird für jedes Zeichen ein "*" angezeigt. Das Datenformat des Eingabewerts ist nicht erkennbar.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verhalten".
2. Aktivieren Sie "Verdeckte Eingabe".

Überschneidungen vermeiden bei Ausgabefeldern

Wenn in einem Bild mehrere E/A-Felder als Ausgabefelder mit transparentem Hintergrund projektiert sind, überschneiden sich eventuell diese E/A-Felder. Der transparente Teil des einen Feldes überdeckt die Ziffern des anderen Feldes. Dabei kommt es möglicherweise zu Darstellungsproblemen in Runtime. Um solche Überschneidungen zu vermeiden, setzen Sie in den Objekteigenschaften unter "Eigenschaften > Eigenschaften > Gestaltung" die Rahmen der E/A-Felder auf null. Aktivieren Sie "Eigenschaften > Eigenschaften > Darstellung > Objekt an Inhalt anpassen".

Grenzen

Im Inspektorfenster unter "Eigenschaften > Eigenschaften > Grenzen" legen Sie die Farben für die Werte fest, die die Grenzwerte überschreiten oder unterschreiten.
Die Grenzen definieren Sie über Eigenschaften einer Variable. In Runtime Professional können Sie festlegen, ob die Farben verwendet werden sollen.

Bei einer Grenzüberschreitung in Runtime wechselt dann die Hintergrundfarbe im E/A-Feld ensprechend Ihrer Projektierung, auch wenn sich das E/A-Feld im Modus "Eingabe" befindet.

In Runtime Professional können Sie außerdem einen Grenzwertebereich für die Eingabe ins E/A-Feld über "Eigenschaften > Eigenschaften > Grenzen" definieren.

Wenn Sie einen numerischen Wert eingeben, der außerhalb der Grenzen liegt, wird dieser nicht übernommen, z. B. 80 beim Grenzwert 78. In diesem Fall wird am Bediengerät eine Systemmeldung ausgegeben, wenn ein Meldefenster projektiert ist. Der ursprüngliche Wert wird wieder angezeigt.

Nachkommastellen bei numerischen Werten

Der Projekteur kann für ein numerisches Eingabefeld die Anzahl der Nachkommastellen festlegen. Wenn Sie einen Wert in ein solches EA-Feld eingeben, wird die Anzahl der Nachkommastellen überprüft. Zu viel eingegebene Nachkommastellen werden ignoriert. Zu wenig eingegebene Nachkommastellen werden mit "0" aufgefüllt.

Verhalten beim Wechsel zwischen Eingabefeldern

Wenn durch Bedienung innerhalb eines Bildes von einem Eingabefeld zu einem anderen gewechselt wird und dadurch die Bildschirmtastatur angezeigt wird, so wird für das vorherige Feld das Ereignis "Feld verlassen" nicht sofort ausgelöst, sondern erst nach Schließen der Bildschirmtastatur.

Siehe auch

Bedienmeldungen projektieren (Seite 862)

E/A-Feld: Darstellungsformat (Seite 371)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.13 E/A-Feld: Verhalten bei Bedienung

Einleitung

In WinCC Runtime Professional können Sie die Fenster-Darstellung des Objekts "E/A-Feld" über Eigenschaften im Inspektorfenster konfigurieren.

Verhalten bei Bedienung

Wertübernahme bei vollständiger Eingabe

Die Eigenschaft "Wertübernahme bei vollständiger Eingabe" legt fest, wann ein Eingabewert übernommen wird. Wenn die Eingabe mit <ENTER> bestätigt wurde, wird der eingegebene Wert übernommen, falls das Attribut den Wert "Nein" besitzt. Sonst wird der Eingabewert automatisch übernommen, sobald die voreingestellte Anzahl an Zeichen eingegeben wurde.

Wertübernahme beim Verlassen

Mit der Eigenschaft "Wertübernahme beim Verlassen" kann eine Wertübernahme auch für den Fall aktiviert werden, dass das E/A-Feld ohne vorherige Bestätigung oder Erreichen der erforderlichen Zeichenzahl verlassen wird.

Löschen bei fehlerhafter Eingabe

Mit der Eigenschaft "Löschen bei fehlerhafter Eingabe" kann die Übernahme eines fehlerhaften Eingabewerts beim Verlassen des Feldes verhindert werden. Fehlerhaft ist beispielsweise ein Eingabewert, der dem vordefinierten Datenformat des Eingabefeldes nicht entspricht.

Löschen beim Aktivieren

Die Eigenschaft "Löschen beim Aktivieren" legt fest, ob der Feldinhalt beim Anwählen des Eingabefeldes gelöscht wird.

Cursorsteuerung

Legt einen Wert fest, bei dem der Alpha-Cursor automatisch in das nächste Feld in der TAB-Reihenfolge springt, nachdem eine Eingabe erfolgt.

Bearbeiten beim Aktivieren

Das Attribut "Bearbeiten beim Aktivieren" legt für Eingabefelder fest, ob beim Springen auf das Objekt direkt in den Eingabemodus gewechselt wird oder nicht.

Verdeckte Eingabe

Das Attribut "Verdeckte Eingabe" legt fest, ob der Eingabewert während der Eingabe normal oder verschlüsselt angezeigt wird. Besitzt dieses Attribut den Wert "Ja", so wird jedes eingegebene Zeichen durch das Zeichen "*" ersetzt. Der Eingabewert und das Datenformat des Eingabewerts sind nicht erkennbar.

1.9.2.14 E/A-Feld: Darstellungsformat

Einleitung

Für die Ein- und Ausgabe von Werten in einem EA-Feld stehen vier verschiedene Anzeigeformate zur Verfügung. Zahlenwerte können in binärer, dezimaler oder hexadezimaler Schreibweise verarbeitet werden. Für die Anzeige von Text muss dem EA-Feld das Anzeigeformat "Zeichenkette" zugewiesen werden.

Basierend auf dem Anzeigeformat können unterschiedliche Darstellungsformate gewählt oder frei definiert werden.
Die Definition für ein Darstellungsformat kann als eine Aneinanderreihung von Formatierungscodes umschrieben werden. Diese Formatierungscodes stehen als Platzhalter für eine bestimmte Gruppe von Zeichen. Ist für eine bestimmte Stelle in der Anzeige des EA-Feldes beispielsweise ein Formatierungscode vorgegeben, der nur die Darstellung der Ziffern 0-9 erlaubt, so ist an dieser Stelle eine Eingabe von Buchstaben nicht möglich.

In einem E/A-Feld können Sie neben den numerischen Werten auch Zeichenketten und Datum/Uhrzeit ausgeben. Für die Ausgabe einer Zeichenkette verwenden Sie eine Variable vom Datentyp "String" mit der maximalen Feldlänge von 320 Zeichen. Für die Anzeige von Datum/Uhrzeit-Formaten nutzen Sie die Variablen vom Datentyp "DateTime". Die Anzeige der Datum/Uhrzeit-Formate hängt in Runtime von der am Bediengerät eingestellten Sprache ab.

Formatierungscodes - Datenformat "Binär"

1 Platzhalter für die Binärwerte 0 und 1. Die Anzahl des Formatierungscodes "1" legt die erlaubte Stellenanzahl für die Anzeige eines Binärwerts fest.

Tabelle 1-3 Beispiel: Der achtstellige Binärwert 10011101 kann wie folgt angezeigt werden:

Darstellungsformat	Erlaubte Stellenanzahl	Anzeige
1111111111	8	10011101

Weitere Formatierungsmöglichkeiten sind abhängig von verwendeten Datentypen der Variablen, z. B. im Datentyp Int ist die Formatierung "11111111 11111111".

Formatierungscodes - Datenformat "Dezimal"

9 Platzhalter für die Ziffern 0 bis 9. Die Anzahl des Formatierungscodes "9" legt die erlaubte Anzahl der Stellen für die Anzeige eines Dezimalwerts fest und kann auch für das negative Vorzeichen "-" stehen. Übersteigt die tatsächliche Anzahl der Nachkommastellen die im Anzeigeformat vorgegebene Anzahl, so wird der angezeigte Wert gerundet.

, Ein Komma definiert die Stelle für das Dezimalzeichen. Der Formatierungscode "," kann an einer beliebigen Stelle des Darstellungsformats stehen, darf aber nur einmal verwendet werden. Ein Komma ist nur bei Gleitpunkt-Datentypen unterstützt. Wenn Sie Variablenwerte von Festpunkt-Datentypen anzeigen möchten, aktiveren Sie die Option "Dezimalstellen" für die implizite Umrechnung der Werte.

s Positive Dezimalzahlen werden mit Vorzeichen angezeigt. Der Formatierungscode "s" muss an der ersten Stelle des Ausgabeformats stehen und darf nur einmal verwendet werden.

e Die Dezimalzahl wird in Exponentenschreibweise angezeigt. Der Formatierungscode "e" muss an der letzten Stelle des Ausgabeformats stehen und darf nur einmal verwendet werden.

Tabelle 1-4 Beispiel: Die sechsstellige Gleitpunktzahl 123,456 kann wie folgt angezeigt werden:

Darstellungsformat	Erlaubte Stellenanzahl	Anzeige
999	3	123
999,9	4	123,5
s999,9	4 + Vorzeichen	+123,5
999,999	6	123,456
9999,9999	8[1)]	0123,4560
s9999,9999	8 + Vorzeichen[1)]	+0123,4560
9,99999e	6	1,23456e+002[2)]

[1)] Aktivieren Sie dafür die Option "Führende Nullstellen"

[2)] Diese Darstellung ist nur in einem E/A-Feld im Ausgabemodus möglich

E/A-Feld mit Anzeigeformat "Dezimal" und Darstellungsformat ohne Vorzeichen "s"

Sie haben eine Prozessvariable mit einem E/A-Feld verbunden. Das Anzeigeformat des E/A-Feldes ist "Dezimal". Als Darstellungsformat können Sie ein Format ohne Vorzeichen wählen.

Formatierungscodes - Datenformat "Hexadezimal"

Die sechstellige Festpunktzahl 123456 kann wie folgt angezeigt werden:

Darstellungsformat	Stellenanzahl	Anzeige
FFFFFFFF	8	0001E240

Weitere Formatierungsmöglichkeiten sind abhängig von verwendeten Datentypen der Variablen, z. B. im Datentyp Int ist die Formatierung "FFFF".

Siehe auch

E/A-Feld (Seite 368)

1.9.2.15 E/A-Feld: Darstellungsformat

Einleitung

Für die Ein- und Ausgabe von Werten in einem EA-Feld stehen in WinCC Runtime Professional vier verschiedene Anzeigeformate zur Verfügung. Zahlenwerte können in binärer, dezimaler oder hexadezimaler Schreibweise verarbeitet werden. Für die Anzeige von Text muss dem EA-Feld das Anzeigeformat "Zeichenkette" zugewiesen werden.

Basierend auf dem Anzeigeformat können unterschiedliche Darstellungsformate gewählt oder frei definiert werden.
Die Definition für ein Darstellungsformat kann als eine Aneinanderreihung von Formatierungscodes umschrieben werden. Diese Formatierungscodes stehen als Platzhalter für eine bestimmte Gruppe von Zeichen. Ist für eine bestimmte Stelle in der Anzeige des EA-Feldes beispielsweise ein Formatierungscode vorgegeben, der nur die Darstellung der Ziffern 0-9 erlaubt, so ist an dieser Stelle eine Eingabe von Buchstaben nicht möglich.

Hinweis

Wenn der anzuzeigende Wert nicht exakt der Definition des Darstellungsformats entspricht, werden nur drei Sterne angezeigt. Dies gilt sowohl für die Länge der gesamten Zeichenfolge als auch für Art und Position der einzelnen Zeichen.

Formatierungscodes - Datenformat "Binär"

1 Platzhalter für die Binärwerte 0 und 1. Die Anzahl des Formatierungscodes "1" legt die erlaubte Stellenanzahl für die Anzeige eines Binärwerts fest.

0 Bei Bedarf wird dem Binärwert eine führende Null vorangestellt. Das Darstellungsformat kann also mit dem Formatierungscode "0" beginnen, darf diesen aber auch nur einmal enthalten.

Tabelle 1-5 Beispiel: Der achtstellige Binärwert 10011101 kann wie folgt angezeigt werden:

Darstellungsformat	Erlaubte Stellenanzahl	Anzeige
11	2	01
011	2 + führende Null	001
1111	4	1101
01111	4 + führende Null	01101
1111111	7	0011101
01111111	7 + führende Null	00011101
1111111111	10	10011101
01111111111	10 + führende Null	010011101

Formatierungscodes - Datenformat "Dezimal"

9 Platzhalter für die Ziffern 0 bis 9. Die Anzahl des Formatierungscodes "9" legt die erlaubte Anzahl der Stellen für die Anzeige eines Dezimalwerts fest. Übersteigt die tatsächliche Anzahl der Nachkommastellen die im Anzeigeformat vorgegebene Anzahl, so wird der angezeigte Wert gerundet.

, Ein Komma definiert die Stelle für das Dezimalzeichen. Der Formatierungscode "," kann an einer beliebigen Stelle des Darstellungsformats stehen, darf aber nur einmal verwendet werden.

s Positive Dezimalzahlen werden mit Vorzeichen angezeigt. Der Formatierungscode "s" muss an der ersten Stelle des Ausgabeformats stehen und darf nur einmal verwendet werden.

0 Führende und nachfolgende Nullen werden angezeigt, wenn die tatsächliche Anzahl der Vor- oder Nachkommastellen geringer ist als die im Anzeigeformat vorgegebene Anzahl. Der Formatierungscode "0" muss vor der ersten "9" stehen und darf nur einmal verwendet werden.

e Die Dezimalzahl wird in Exponentenschreibweise angezeigt. Der Formatierungscode "e" muss an der letzten Stelle des Ausgabeformats stehen und darf nur einmal verwendet werden.

Tabelle 1-6 Beispiel: Die sechsstellige Dezimalzahl 123,456 kann wie folgt angezeigt werden:

Darstellungsformat	Erlaubte Stellenanzahl	Anzeige
999	3	124
999,9	4	123,5
s999,9	4 + Vorzeichen	+123,5
999,999	6	123,456
09999,9999	8 + Nullen	0123,4560
s09999,9999	8 + Vorzeichen + Nullen	+0123,4560
1111111111	10	10011101
9,99999e	6	1,23456e+002

Hinweis

Bei Ablage einer Gleitkommazahl im IEEE-Format der S5 sollte für die Darstellung in einem EA-Feld ein Darstellungsformat verwendet werden, das Vorzeichen und Exponent vorsieht (bspw. s0999,999e).

E/A-Feld mit Anzeigeformat "Dezimal" und Darstellungsformat ohne Vorzeichen "s"

Sie haben eine Prozessvariable mit einem E/A-Feld verbunden. Das Anzeigeformat des E/A-Feldes ist "Dezimal". Als Darstellungsformat können Sie ein Format mit und ohne Vorzeichen wählen,

Wenn Sie als "Darstellungsformat" ein Format ohne "s" wählen z. B. "999", dann hat das folgende Auswirkungen:

1. In Runtime können Sie keine negativen Werte über das E/A-Feld setzen.
2. Wenn die Variable einen negativen Wert annimmt, dann erstellt das E/A-Feld ein Zweierkomplement und ein verfälschter positiver Wert wird ausgegeben.

Formatierungscodes - Datenformat "Hexadezimal"

f Platzhalter für die Buchstaben A-F bzw. a-f und die Ziffern 0-9, die zur Darstellung von Hexadezimalzahlen verwendet werden. Die erlaubte Anzahl der Zeichen wird durch die Anzahl des Formatierungscodes "f" im Ausgabeformat definiert.

0 Führende Nullen des Hexadezimalwerts werden angezeigt, wenn das Ausgabeformat mit dem Formatierungscode "0" beginnt. Der Formatierungscode "0" darf nur einmal enthalten sein.

Tabelle 1-7 Beispiel: Der achtstellige Binärwert ABCDEF01 kann wie folgt angezeigt werden:

Darstellungsformat	Erlaubte Stellenanzahl	Anzeige
ff	2	01
0ff	2 + führende Null	001
ffff	4	EF01
0ffff	4 + führende Null	0EF01
ffffffffff	10	ABCDEF01
0ffffffffff	7 + führende Null	0ABCDEF01

Formatierungscodes - Datenformat "Zeichenkette"

Die erlaubte Länge einer Zeichenfolge wird durch die Anzahl der verwendeten Formatierungscodes definiert (Ausnahme "*").

*	Eingabe einer beliebigen Zeichenfolge in beliebiger Länge
?	Eingabe einer beliebigen Zeichenfolge
a	Erlaubt sind Kleinbuchstaben, Großbuchstaben und Ziffern keine Trennzeichen o. Ä.
A	Erlaubt sind Großbuchstaben und Ziffern keine Kleinbuchstaben, Trennzeichen, o. Ä.
b	Erlaubt sind Kleinbuchstaben und Großbuchstaben keine Ziffern, Trennzeichen o. Ä.
B	Erlaubt sind nur Großbuchstaben keine Kleinbuchstaben, Ziffern, Trennzeichen o. Ä.
1-9	Die Formatierungscodes "1", "2", ..., "9" werden als Platzhalter für Ziffern verwendet. Der gewählte Formatierungscode definiert zugleich die tatsächlich erlaubten Ziffern: Wird beispielsweise die "2" angegeben, so können nur die Ziffern 0, 1 oder 2 angezeigt werden. Der Formatierungscode "8" erlaubt alle Ziffern außer der 9.
h	Erlaubt sind nur die Ziffern 0-9 sowie die Buchstaben A-F bzw. a-f Der Formatierungscode "h" erlaubt also nur Zeichen, die zur Darstellung von Hexadezimalzahlen benötigt werden.
t	Der Formatierungscode "t" erzwingt die Eingabe eines Trennzeichens an der vorgegebenen Stelle. Als Trennzeichen gelten: Schrägstrich, Doppelpunkt, Komma, Punkt und Leerzeichen.

1.9.2.16 Editierbares Textfeld

Verwendung

Das Objekt "Editierbares Textfeld" bietet die Möglichkeit, im Bild mehrere Zeilen Text in einem Rechteck darzustellen. Wenn der Text größer als das Rechteck ist, fügt WinCC automatisch eine Laufleiste am rechten Rand hinzu.

Wenn Sie die Bedienung freigeben, kann der Bediener den Text in Runtime bearbeiten. Durch Anbindung von Variablen können Sie den mehrzeiligen Text zur Eingabe oder Ausgabe von Text nutzen.

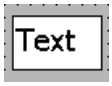

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Text: Legt den Text für das "Editierbare Textfeld" fest.

Text

Der Text für das "Editierbaren Textfelds" wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie einen beliebig langen Text ein.
   Für mehrzeiligen Text können Sie einen Zeilenumbruch durch Drücken der Taste <Enter>, mit der Tastenkombination <Umschalttaste + Enter> oder mit der Tastenkombination <Strg + Enter> setzen.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.17 Ellipse

Verwendung

Das Objekt "Ellipse" ist ein geschlossenes Objekt, das mit einer Farbe oder einem Muster gefüllt werden kann.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius horizontal: Legt den horizontalen Radius des elliptischen Objekts fest.
- Radius vertikal: Legt den vertikalen Radius des elliptischen Objekts fest.

Radius horizontal

Den horizontalen Radius des Objekts "Ellipse" legen Sie im Inspektorfenster fest. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Horizontal" einen Wert zwischen 0 und 2500 ein.

Radius vertikal

Den vertikalen Radius des Objekts "Ellipse" legen Sie im Inspektorfenster fest. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie bei "Vertikal" einen Wert zwischen 0 und 2500 ein.

Siehe auch

Ellipsenbogen (Seite 378)
Ellipsensegment (Seite 379)
Objekt drehen (Seite 116)
Objekt spiegeln (Seite 118)
Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.18 Ellipsenbogen

Verwendung

Das Objekt "Ellipsenbogen" ist ein offenes Objekt. Wenn Sie das Objekt mit Farbe füllen wollen, verwenden Sie das Objekt "Ellipsensegment". Ein Ellipsenbogen besteht standardmäßig aus einer Viertel-Ellipse, die Sie beliebig anpassen.

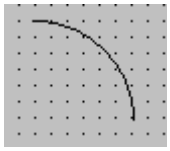

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radien horizontal und vertikal: Legen den horizontalen und vertikalen Radius des elliptischen Objekts fest.
- Anfangs und Endwinkel: Legen fest, wo sich Anfangs- und Endwinkel auf einem virtuellen Kreis mit 360° befinden.

Radien festlegen

Im Inspektorfenster legen Sie den horizontalen und den vertikalen Radius des Objekts "Ellipsenbogen" fest. Die Werte geben Sie in der Einheit Pixel ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie für "Radius horizontal" und "Radius vertikal" jeweils einen Wert ein.

Anfangs- und Endwinkel festlegen

Die Länge des Ellipsenbogens bestimmen Sie durch die Attribute "Anfangswinkel" und "Endwinkel". Die Werte geben Sie in der Einheit Grad ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie für "Anfang Winkel" und "Ende Winkel" jeweils einen Wert ein.

Alternatives Vorgehen

Die Anfangs- und Endpunkte des Ellipsenbogens können Sie auch mit der Maus ändern.

1. Positionieren Sie den Mauszeiger auf einem Anfangs- oder Endpunkt des Ellipsenbogens. Anfangs- und Endpunkt sind durch kleine blaue Quadrate symbolisiert.
   Der Mauszeiger verwandelt sich in ein Kreuz.
2. Ziehen Sie den Anfangs- oder Endpunkt bei gedrückter Maustaste auf die gewünschte Position.

Siehe auch

Ellipse (Seite 377)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.19 Ellipsensegment

Verwendung

Das Objekt "Ellipsensegment" ist ein geschlossenes Objekt, das Sie mit einer Farbe oder einem Muster füllen können. Ein Ellipsensegment besteht standardmäßig aus einer Viertel-Ellipse, die Sie beliebig anpassen.

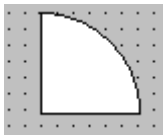

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radien horizontal und vertikal: Legen den horizontalen und vertikalen Radius des elliptischen Objekts fest.
- Anfangs- und Endwinkel: Legen fest, wo sich Anfangs- und Endpunkt auf einem virtuellen Kreis mit 360 Grad befinden.

Radien festlegen

Im Inspektorfenster legen Sie den horizontalen und den vertikalen Radius des Objekts "Ellipsensegment" fest. Die Werte geben Sie in der Einheit Pixel ein:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Bogen" bei "Radius horizontal" und "Radius vertikal" jeweils einen Wert ein.

Anfangs- und Endwinkel festlegen

Die Größe des Ellipsensegments bestimmen Sie durch die Attribute "Anfangswinkel" und "Endwinkel". Die Werte geben Sie in der Einheit Grad ein:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Bogen" bei "Anfangswinkel" und "Endwinkel" jeweils einen Wert ein.

Alternatives Vorgehen

Die Anfangs- und Endpunkte des Ellipsensegments können Sie auch mit der Maus ändern.

1. Positionieren Sie den Mauszeiger auf einem Anfangs- oder Endpunkt des Ellipsensegments. Anfangs- oder Endpunkt sind durch kleine blaue Quadrate symbolisiert. Der Mauszeiger verwandelt sich in ein Kreuz.
2. Ziehen Sie den Anfangs- oder Endpunkt bei gedrückter Maustaste auf die gewünschte Position.

Siehe auch

Ellipse (Seite 377)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.20 f(t)-Kurvenanzeige

Verwendung

Mit dem Objekt "f(t)-Kurvenanzeige" stellen Sie Variablenwerte aus dem aktuellen Prozess oder dem Archiv in Form von Kurven als Funktionen der Zeit dar.

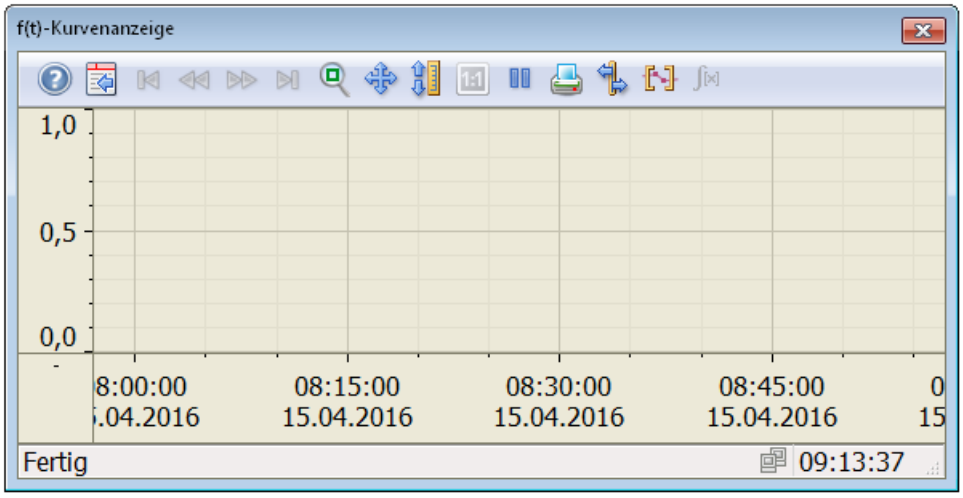

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Kurven hinzufügen und konfigurieren
- Diagramm konfigurieren
- Einstellungen für das Fenster
- Persistenz
- Symbolleiste

Kurven konfigurieren

1. Wählen Sie "Eigenschaften > Eigenschaften > Kurven > Datenquelle" die Datenversorgung für die Kurve.
   - "Archivvariablen": Die Kurvenanzeige wird mit Werten aus einem Variablenarchiv versorgt.
   - "Benutzerdefiniert": Die Kurvenanzeige wird in Runtime über Skripte benutzerdefiniert versorgt.
   - "Variablen": Die Kurvenanzeige wird mit den Werten einer Variablen versorgt.
2. Legen Sie unter "Datenbereich" fest, ob die Kurve die Werte statisch darstellt oder die Werte aktualisiert werden.
   - "Automatisch": Der angezeigte Wertebereich wird automatisch an die aktuellen Werte angepasst.
   - "Anfangswert" / "Endwert": Sie legen den minimalen Wert und den maximalen Wert für den Wertebereich fest.
3. Konfigurieren Sie unter "Grenzen" die farbige Kennzeichnung spezieller Werte, z. B. Oberer und Unterer Grenzwert.
4. Wählen Sie "Eigenschaften > Eigenschaften > Zeitachse".
5. Konfigurieren Sie den "Zeitbereich" der Kurvendarstellung.
   - "Zeitspanne": Sie legen den Zeitbereich über einen Startzeitpunkt und eine nachfolgende Zeitspanne fest.
   - "Endzeit": Sie legen den Zeitbereich über einen Startzeitpunkt und einen Endzeitpunkt fest.
   - "Messpunkte": Sie legen den Zeitbereich über einen Startzeitpunkt und eine Anzahl von Messpunkten fest.

Diagramm konfigurieren

Die Darstellung mehrerer Kurven konfigurieren Sie unter "Diagramm":

- Gemeinsames oder einzelne Diagramme
- Gemeinsame oder unterschiedliche Achsen
- Schreibrichtung aller Kurven

Einstellungen für das Fenster

Unter "Darstellung > Fenster" legen Sie fest, ob der Benutzer in Runtime die Größe des Fensters ändern und das Fenster schließen kann.

Persistenz

Unter "Eigenschaften > Eigenschaften > Security > Persistenz" legen Sie fest, ob ein Benutzer in Runtime die Projektierung der Kurvenanzeige ändern kann und wie lange Änderungen erhalten bleiben.

- "Keine Persistenz"
  Die Änderungen an der Konfiguration werden nicht gespeichert. Die Änderungen werden beim nächsten Bildwechsel und beim Beenden von Runtime verworfen.
- "Persistenz"
  Die Änderungen an der Konfiguration werden gespeichert. Die Änderungen bleiben auch nach dem Beenden von Runtime erhalten. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.
- "Persistenz in Runtime"
  Die Änderungen an der Konfiguration werden nur während Runtime gespeichert. Wenn Sie das Runtime beenden, werden die Änderungen verworfen. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.

Symbolleiste

Die Bedienelemente der f(t)-Kurvenanzeige in Runtime legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste" fest. Folgende Bedienelemente stehen für die f(t)-Kurvenanzeige zur Verfügung:

Schaltfläche	Name	Funktion
	Online-Hilfe	Öffnet die Online-Hilfe.
	Konfiguration öffnen	Öffnet den Konfigurationsdialog.
	Erster Datensatz	Im Archiv blättern: Zeigt den Kurvenverlauf beginnend mit dem ersten archivierten Wert.
	Vorhergehender Datensatz	Im Archiv blättern: Zeigt den Kurvenverlauf des vorhergehenden Zeitintervalls.
	Nächster Datensatz	Im Archiv blättern: Zeigt den Kurvenverlauf des nachfolgenden Zeitintervalls.
	Letzter Datensatz	Im Archiv blättern: Zeigt den Kurvenverlauf bis zum letzten archivierten Wert.
	Lineal anzeigen/verbergen	Ermittelt die Koordinaten eines Punkts der Kurve.
	Zoombereich	Vergrößert einen beliebigen Ausschnitt des Kurvenfensters.

Schaltfläche	Name	Funktion
	Originalansicht	Schaltet von der vergrößerten Kurvenanzeige zurück zur Normalansicht.
	Variablenauswahl	Öffnet den Dialog zur Auswahl der Archive und Variablen.
	Kurvenauswahl	Öffnet den Dialog zum Einstellen der Sichtbarkeit von Kurven.
	Auswahl des Zeitbereichs	Öffnet den Dialog zum Einstellen des Zeitbereichs, der im Kurvenfenster dargestellt wird.
	Vorherige Kurve	Zeigt die vorherige Kurve im Vordergrund.
	Nächste Kurve	Zeigt die nächste Kurve im Vordergrund.
	Starte/Beende Update	Stoppt und startet die Kurvenaktualisierung. Die Werte werden zwischengespeichert und nachgetragen, sobald Sie die Kurvenaktualisierung wieder starten.
	Druckarchiv	Startet den Druck der im Kurvenfenster angezeigten Kurven. Den Druckauftrag legen Sie unter "Allgemein > Drucken > Druckauftrag" fest.
	Auswahl des Statistikbereichs	Ermöglicht die Festlegung eines Zeitbereichs, für den statistische Werte ermittelt werden. Im Kurvenfenster werden vertikale Linien angezeigt, mit denen Sie den Zeitbereich einstellen.
	Statistikfenster	Öffnet das Statistikfenster zur Anzeige von Minimum, Maximum, Durchschnitt und Standardabweichung für den gewählten Zeitbereich und die gewählte Kurve.
	Prozessdaten in Runtime als CSV-Datei exportieren	Öffnet den Dialog zum Speichern der Kurvendaten im CSV-Format.

Die Reihenfolge der Schaltflächen ist festgelegt. Unter "Hotkey" können Sie für alle Schaltflächen der Symbolleiste individuelle Tastenbelegungen und Tastenkombinationen einrichten.

Siehe auch

f(x)-Kurvenanzeige (Seite 387)

Tabellenanzeige (Seite 515)

Kurvenanzeige projektieren (Seite 728)

Wertetabelle (Seite 523)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.21 f(x)-Kurvenanzeige

Verwendung

Mit dem Objekt "f(x)-Kurvenanzeige" stellen Sie die Werte einer Variable als Funktion einer anderen Variable dar. Auf diese Weise stellen Sie z. B. Temperaturverläufe als Funktion des Drucks dar.

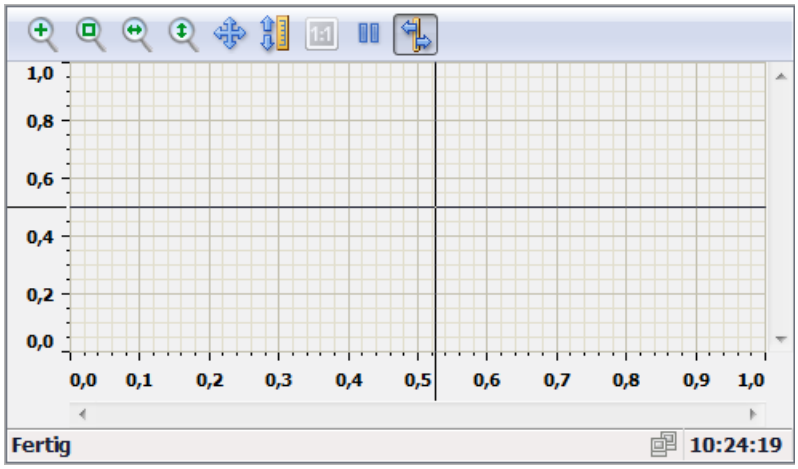

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Kurven hinzufügen und konfigurieren
- Diagramm konfigurieren
- Einstellungen für das Fenster
- Persistenz
- Symbolleiste

Kurven konfigurieren

Im Inspektorfenster "Eigenschaften > Eigenschaften > Kurve" erstellen und konfigurieren Sie Kurven.

1. Wählen Sie in der Spalte "Datenquelle" die Datenversorgung für die Kurve.
   - "Archivvariable": Die Kurvenanzeige wird mit Werten aus einem Variablenarchiv versorgt.
   - "Variablen": Die Kurvenanzeige wird mit aktuellen Werten aus der Steuerung versorgt. Die Werte werden ständig aktualisiert.
2. Konfigurieren Sie unter "Datenbereich" den Bereich der Kurvendarstellung.
   - "Zeitspanne": Sie legen den Zeitbereich über einen Startzeitpunkt und eine nachfolgende Zeitspanne fest.
   - "Endzeit": Sie legen den Zeitbereich über einen Startzeitpunkt und einen Endzeitpunkt fest.
   - "Messpunkte": Sie legen den Zeitbereich über einen Startzeitpunkt und eine Anzahl von Messpunkten fest.
3. Konfigurieren Sie unter "X-Achse " und unter "Y-Achse" jeweils den Wertebereich der Kurvendarstellung.
4. Konfigurieren Sie unter "Grenzen" die farbige Kennzeichnung spezieller Werte, z. B. Oberer und Unterer Grenzwert, Unsicherer Status.

Diagramm konfigurieren

Die Darstellung mehrerer Kurven konfigurieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung":

- Gemeinsames Diagramm oder einzelne Diagramme
- Gemeinsame oder unterschiedliche Achsen
- Schreibrichtung aller Kurven

Symbolleiste

Die Bedienelemente der f(x)-Kurvenanzeige in Runtime legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste" fest. Folgende Bedienelemente stehen für die f(x)-Kurvenanzeige zur Verfügung:

Schaltfläche	Name	Funktion
	Zoom +/-	Vergrößert bzw. verkleinert die Kurven im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die Kurven. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die Kurven.
	Zoombereich	Vergrößert einen beliebigen Ausschnitt des Kurvenfensters. Mit der Maus legen Sie im Kurvenfenster einen Bereich fest. Dieser Bereich des Kurvenfensters wird vergrößert.
	Zoom X-Achse	Vergrößert bzw. verkleinert die X-Achse im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die X-Achse. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die X-Achse.

Schaltfläche	Name	Funktion
	Zoom Y-Achse	Vergrößert bzw. verkleinert die Y-Achse im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die Y-Achse. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die Y-Achse.
	Kurvenbereich	Mit der Schaltfläche verschieben Sie die Kurven im Kurvenfenster entlang der X-Achse und Y-Achse.
	Achsenbereich	Mit der Schaltfläche verschieben Sie die Kurven im Kurvenfenster entlang der Werteachse.
	Originalansicht	Schaltet von der vergrößerten Kurvenanzeige zurück zur Normalansicht.
	Start/Stopp	Stoppt und startet die Kurvenaktualisierung. Die Werte werden zwischengespeichert und nachgetragen, sobald Sie die Kurvenaktualisierung wieder starten.
	Lineal	Ermittelt die Koordinaten eines Punkts der Kurve.

Die Reihenfolge der Schaltflächen ist festgelegt. Der Benutzer kann für alle Schaltflächen der Symbolleiste individuelle Tastenbelegungen und Tastenkombinationen einrichten.

Hinweise zur Performance

Die Darstellung von Variablen mit kurzen Erfassungszyklen kann dazu führen, dass die in der f(x)-Kurvenanzeigeangezeigte Daten langsam aktualisiert werden.

Um die verzögerte Darstellung im Control zu vermeiden, konfigurieren Sie den Erfassungszyklus der Variablen von größer oder gleich 1 s.

Siehe auch

Kurvenanzeige projektieren (Seite 728)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.22 f(x)-Kurvenanzeige

Verwendung

Mit dem Objekt "f(x)-Kurvenanzeige" stellen Sie die Werte einer Variable als Funktion einer anderen Variable dar. Auf diese Weise stellen Sie z. B. Temperaturverläufe als Funktion des Drucks dar. Zusätzlich können Sie die Kurve mit einer Sollwertkurve vergleichen.

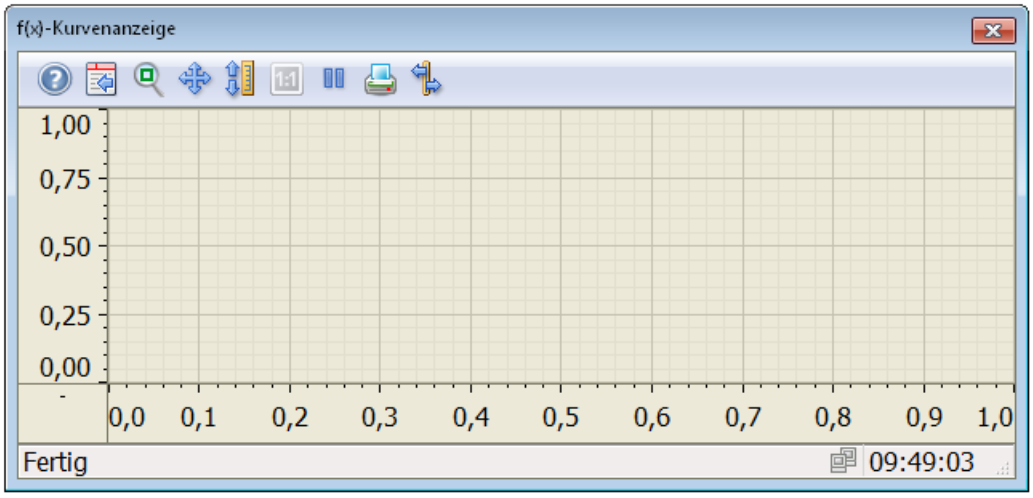

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Kurven hinzufügen und konfigurieren
- Diagramm konfigurieren
- Einstellungen für das Fenster
- Persistenz
- Symbolleiste

Kurven konfigurieren

1. Wählen Sie "Eigenschaften > Eigenschaften > Kurven > Datenquelle" die Datenversorgung für die Kurve.
   - "Archivvariablen": Die Kurvenanzeige wird mit Werten aus einem Variablenarchiv versorgt.
   - "Benutzerdefiniert": Die Kurvenanzeige wird in Runtime über Skripte benutzerdefiniert versorgt.
   - "Rezeptur": Die Kurvenanzeige wird mit Werten aus einer Rezeptur versorgt.
   - "Variablen": Die Kurvenanzeige wird mit den Werten einer Variablen versorgt.
2. Konfigurieren Sie unter "Datenbereich" den Bereich der Kurvendarstellung.
   - "Zeitspanne": Sie legen den Zeitbereich über einen Startzeitpunkt und eine nachfolgende Zeitspanne fest.
   - "Endzeit": Sie legen den Zeitbereich über einen Startzeitpunkt und einen Endzeitpunkt fest.
   - "Messpunkte": Sie legen den Zeitbereich über einen Startzeitpunkt und eine Anzahl von Messpunkten fest.
   - "Start ID" und "Anzahl Datensätze": Bei der Datenversorgung über Rezepturen legen Sie den Bereich durch die ID des ersten Datensatzes und die Anzahl darauf folgender Datensätze fest.
3. Konfigurieren Sie unter "X-Achse " und unter "Y-Achse" jeweils den Wertebereich der Kurvendarstellung.
   - "Automatisch": Der angezeigte Wertebereich wird automatisch an die aktuellen Werte angepasst.
   - "Anfangswert" / "Endwert": Sie legen den minimalen Wert und den maximalen Wert für den Wertebereich fest.
4. Konfigurieren Sie unter "Grenzen" die farbige Kennzeichnung spezieller Werte, z. B. Oberer und unterer Grenzwert, Unsicherer Status.

Diagramm konfigurieren

Die Darstellung mehrerer Kurven konfigurieren Sie unter "Gestaltung":

- Gemeinsames Diagramm oder einzelne Diagramme
- Gemeinsame oder unterschiedliche Achsen
- Schreibrichtung aller Kurven

Einstellungen für das Fenster

Unter "Darstellung > Fenster" legen Sie fest, ob der Benutzer in Runtime die Größe des Fensters ändern und das Fenster schließen kann.

Persistenz

Unter "Security > Persistenz" legen Sie fest, wie lange die Änderungen eines Benutzers an der Kurvenanzeige in Runtime erhalten bleiben.

- "Persistenz erlauben"
  Der Benutzer kann Einstellungen ändern. Die Änderungen bleiben bis zum Bildwechsel erhalten.
- "Persistenz bis Abmelden"
  Änderungen der Benutzergruppe bleiben bis zum Abmelden erhalten.
- "Persistenz bis zum nächsten Laden"
  Änderungen der Benutzergruppe bleiben erhalten, bis das Projekt auf dem Bediengerät aktualisiert wird.

Symbolleiste

Die Bedienelemente der f(x)-Kurvenanzeige in Runtime legen Sie im Inspektorfenster unter "Eigenschaften > Symbolleiste" fest. Folgende Bedienelemente stehen für die f(x)-Kurvenanzeige zur Verfügung:

Schaltfläche	Name	Funktion
	Online-Hilfe	Öffnet die Online-Hilfe.
	Konfigurationsdialog	Öffnet den Konfigurationsdialog.
	Zoom +/-	Vergrößert bzw. verkleinert die Kurven im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die Kurven. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die Kurven.
	Zoombereich	Vergrößert einen beliebigen Ausschnitt des Kurvenfensters. Mit der Maus legen Sie im Kurvenfenster einen Bereich fest. Dieser Bereich des Kurvenfensters wird vergrößert.
	Zoom X-Achse	Vergrößert bzw. verkleinert die X-Achse im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die X-Achse. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die X-Achse.
	Zoom Y-Achse	Vergrößert bzw. verkleinert die Y-Achse im Kurvenfenster. Mit der linken Maustaste vergrößern Sie die Y-Achse. Wenn Sie die <Shift> gedrückt halten, verkleinern Sie mit der linken Maustaste die Y-Achse.
	Kurvenbereich verschieben	Mit der Schaltfläche verschieben Sie die Kurven im Kurvenfenster entlang der X-Achse und Y-Achse.
	Achsenbereich verschieben	Mit der Schaltfläche verschieben Sie die Kurven im Kurvenfenster entlang der Werteachse.
	Originalansicht	Schaltet von der vergrößerten Kurvenanzeige zurück zur Normalansicht.
	Datenverbindung wählen	Öffnet einen Dialog zur Auswahl von Archiven und Variablen.
	Kurven wählen	Öffnet einen Dialog zum Einstellen der Sichtbarkeit von Kurven.

Schaltfläche	Name	Funktion
	Zeitbereich wählen	Öffnet einen Dialog zur Einstellung des im Kurvenfensters dargestellten Zeitbereich.
	Vorherige Kurve	Zeigt die vorherige Kurve im Vordergrund.
	Nächste Kurve	Zeigt die nächste Kurve im Vordergrund.
	Start/Stop	Stoppt und startet die Kurvenaktualisierung. Die Werte werden zwischengespeichert und nachgetragen, sobald Sie die Kurvenaktualisierung wieder starten.
	Drucken	Startet den Ausdruck der im Kurvenfenster angezeigten Kurve. Den beim Druck verwendeten Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest.
	Daten exportieren	Mit der Schaltfläche exportieren Sie alle oder ausgewählte Runtime-Daten in eine csv-Datei. Wenn die Option "Dialog anzeigen" aktiviert ist, öffnet sich ein Dialog, in dem Sie die Einstellungen zum Export sehen und den Export starten können. Mit den entsprechenden Berechtigungen dürfen Sie auch die Datei und das Verzeichnis für den Export auswählen. Wenn kein Dialog angezeigt wird, startet der Export der Daten in die voreingestellte Datei.
	Lineal	Ermittelt die Koordinaten eines Punkts der Kurve.
	Backup verbinden	Öffnet einen Dialog, in dem ausgewählte Archive mit WinCC Runtime verbunden werden.
	Backup trennen	Öffnet einen Dialog, in dem ausgewählte Archive von WinCC Runtime getrennt werden.

Die Reihenfolge der Schaltflächen ist festgelegt. Der Benutzer kann für alle Schaltflächen der Symbolleiste individuelle Tastenbelegungen und Tastenkombinationen einrichten.

Siehe auch

f(t)-Kurvenanzeige (Seite 380)

Tabellenanzeige (Seite 515)

Kurvenanzeige projektieren (Seite 728)

Wertetabelle (Seite 523)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.23 Grafikanzeige

Verwendung

Das Objekt "Grafikanzeige" dient der Anzeige von Grafiken.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Grafik: Legt die Grafikdatei fest, die in dem Objekt angezeigt wird.
- Grafik anpassen: Legt die automatische Größenanpassung für Objekte mit Grafiken fest.
- Transparente Farbe: Legt fest, ob die transparente Farbe für die Grafik verwendet wird.

Grafik einfügen

Im Objekt "Grafikanzeige" verwenden Sie die folgenden Grafikformate: *.bmp, *.tif, *.png, *.ico, *.emf, *.wmf, *.gif, *.svg, *.jpg oder *.jpeg. Sie können Grafiken auch als OLE-Objekte in der Grafikanzeige verwenden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie die Grafik, die Sie einfügen wollen.
   Die Grafik wird rechts in der Vorschau angezeigt.
   Sie haben die Möglichkeit, über die Schaltfläche die Grafik aus einer Datei zu erzeugen oder über die Schaltfläche eine neue Grafik aus einem OLE Objekt zu erzeugen.
3. Um die Grafik in die Grafikanzeige einzufügen, klicken Sie auf "Übernehmen".

Grafik anpassen

Im Inspektorfenster wird festgelegt, ob die in einer Grafikanzeige dargestellte Grafik sich in Runtime an die Größe der Grafikanzeige anpasst oder nicht.

Option	Beschreibung
"Nicht automatisch anpassen"	Die Größe wird in Runtime nicht automatisch angepasst.
"Grafik an Objektgröße anpassen"	Die Grafik wird in Runtime an die Objektgröße angepasst.
"Objektgröße an Grafik anpassen"	Grafikanzeige wird an die größte verwendete Grafik angepasst und andere Grafiken werden gestreckt.
"Objektgröße an Grafik angleichen"	Grafikanzeige wird an die größte verwendete Grafik angepasst, aber die Grafiken werden in der Originalgröße angezeigt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie die gewünschte Größenanpassung für die Grafik.

Transparente Farbe

Mit dieser Eigenschaft legen Sie fest, ob die Transparentfarbe für die anzuzeigende Grafik verwendet wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Aktivieren Sie "Hintergrund > Transparent".
3. Wählen Sie eine transparente Farbe.

Hinweis

Bei der Verwendung von Bitmaps in WinCC Bildern erfordert die Einstellung "Transparente Farbe" eine hohe Zeichenperformance bei der Darstellung auf Panels. Um eine bessere Darstellungsperformance zu erreichen, deaktivieren Sie die Einstellung "Transparent" in den Eigenschaften des jeweiligen Anzeigeobjekts. Diese Einschränkung gilt insbesondere bei der Verwendung von Bitmaps als Hintergrundbilder.

Hinweis

Basic Panels

Die Eigenschaft "Transparent" steht für Basic Panels nicht zur Verfügung.

Hinweis

Falls die ausgewählte transparente Farbe der Grafik trotz aktivierter Eigenschaft "Transparent" nicht durchsichtig dargestellt wird, stimmt die Farbe in der Grafik nicht mit der ausgewählten Farbe komplett überein. Zum Beispiel, sieht die ausgewählte transparente Farbe [R=255;G=255;B=255] fast genauso aus, wie die Farbe in der Grafik, die transparent erwartet wird [R=255;G=251;B=255]. In diesem Fall muss die im Objekt verwendete Grafik nachbearbeitet werden. Wählen Sie unter „Projektnavigation > Sprachen & Ressourcen > Grafiksammlung“ die Grafik, die Sie transparent darstellen möchten. Wählen Sie im Kontextmenü der Standardgrafik den Befehl „Bearbeiten“ und füllen Sie den Hintergrund der Grafik mit der gewünschten Farbe, die in der Grafik durchsichtig angezeigt wird. Speichern Sie die Grafik.

Siehe auch

Externe Grafik in der Grafiksammlung hinterlegen (Seite 152)

Bearbeitungsmöglichkeiten von Objekten (Seite 103)

Übersicht über die Objekte (Seite 91)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.24 Grafisches E/A-Feld

Verwendung

Mit dem Objekt "Grafisches E/A-Feld" projektieren Sie eine Liste, die der Anzeige und der Auswahl von Grafikdateien dient.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Typ Bildlaufleiste: Legt die grafische Darstellung der Bildlaufleiste fest.
- Modus: Legt das Verhalten des Objekts in Runtime fest.

Hinweis

Bildlaufleiste

Die Eigenschaft "Bildlaufleiste > Typ" steht für Panels und Runtime Advanced bis Geräteversion V12 zur Verfügung.

Hinweis

Protokolle

In Protokollen geben Grafische E/A-Felder ausschließlich Grafiken aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Auswahl von Grafiken stehen nicht zur Verfügung, z. B. "Bildlaufleiste".

Hinweis

Rahmen

Die Breite und den Stil des Rahmens von einem grafischen E/A-Feld konfigurieren Sie im Modus "Zwei Zustände".

Die Dynamisierung der Rahmenfarben im Modus "Zwei Zustände" hat zur Laufzeit keine Auswirkung.

Modus

Das Verhalten des Objekts "Grafisches E/A-Feld" wird im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Typ > Modus" festgelegt.

Modus	Beschreibung
"Eingabe"	Das Objekt "Grafisches E/A-Feld" wird nur zur Auswahl von Grafiken eingesetzt.
"Eingabe/Ausgabe"	Das Objekt "Grafisches E/A-Feld" wird zur Auswahl und zur Anzeige von Grafiken eingesetzt.
"Ausgabe"	Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt.
"Zwei Zustände"	Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt und kann maximal zwei Zustände besitzen. Sie verwenden keine Grafikliste, sondern fügen je eine Grafik für den Zustand "EIN" und "AUS" ein.

Grafik anpassen

Im Inspektorfenster wird festgelegt, ob die in einem Grafischen E/A-Feld dargestellte Grafik sich in Runtime an die Größe der Anzeige anpasst oder nicht.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie die gewünschte Größenanpassung für die Grafik.

Typ Bildlaufleiste

Das Verhalten für die grafische Darstellung der Bildlaufleiste legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Gestaltung > Bildlaufleiste > Typ" fest.

Typ	Beschreibung
"Dauerhaft"	Die Bildlaufleiste ist immer sichtbar.
"Kein Rollbalken"	Die Bildlaufleiste ist nicht sichtbar.
"Sichtbar nach dem Klicken"	Die Bildlaufleiste wird durch Mausklick sichtbar.

Verhalten bei der Bedienung

Wenn im grafischen EA-Feld eine Kakteengrafik angezeigt wird, wurde im Projekt für einen bestimmten Wert keine Grafik zur Ausgabe definiert. Die Aktivierung wird Ihnen am Bediengerät dadurch angezeigt, dass der Inhalt des grafischen EA-Feldes seine Farbe ändert. Der Rahmen im 3D-Stil wird nur in einem Ausgabefeld grafisch dargestellt.

Siehe auch

Symbolisches E/A-Feld (Seite 503)

Druckauftrag/Skriptdiagnose (Seite 365)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.25 Grafisches E/A-Feld

Verwendung

Mit dem Objekt "Grafisches E/A-Feld" projektieren Sie eine Grafikliste, die der Anzeige und der Auswahl von Grafikdateien dient.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Größenanpassung: Legt das Verhältnis von Anzeigefenster und eingebetteter Grafik fest.

Modus

Das Verhalten des Objekts "Grafisches E/A-Feld" legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Modus" fest.

Modus	Beschreibung
"Ausgabe"	Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt.
"Zwei Zustände"	Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt und kann maximal zwei Zustände besitzen. Sie verwenden keine Grafikliste, sondern fügen je eine Grafik für den Zustand "EIN" und "AUS" ein.

Größenanpassung

Das Verhältnis für die grafische Darstellung im Anzeigefenster legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Größenanpassung" fest.

Typ Größenanpassung	Beschreibung
"Objektgröße an Inhalt anpassen"	Das Anzeigefenster wird an die Größe der Grafik angepasst.
"Inhalt an Objektgröße anpassen"	Größe der Grafik wird an das Anzeigefenster angepasst.

Siehe auch

Symbolisches E/A-Feld (Seite 505)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.26 GRAPH-Übersicht

Verwendung

Das Objekt "GRAPH-Übersicht" dient zur Anzeige des aktuellen Programmstatus für ausgeführte Schritte der GRAPH-Ablaufkette. Fehler beim Ablauf eines Programms werden direkt am entsprechenden Schritt angezeigt.

Im Objekt "GRAPH-Übersicht" werden folgende Informationen angezeigt:

- Name und Zustand des Funktionsbausteins
- Status von Initial- und Simultanschritten
- Nummer und Name des ersten aktuell ausgeführten Schritts
- Betriebsmodus für die Ausführung der GRAPH-Ablaufkette

WinCC unterstützt die Anzeige von mehrsprachigen Schrittnamen für die GRAPH-Bausteine ab der Version 5.0. Beim Sprachenwechsel in Runtime werden die Schrittnamen in der ausgewählten Runtime-Sprache angezeigt. Falls die ausgewählte Sprache im GRAPH-Baustein nicht verfügbar ist, werden die Namen in der Standardsprache Englisch angezeigt.

Hinweis

Das Objekt "GRAPH-Übersicht" kann nicht in Bildbausteinen und im Globalen Bild verwendet werden.

Hinweis

Geräteabhängigkeit des Objekts "GRAPH-Übersicht"

Das Objekt "GRAPH-Übersicht" ist für RT Advanced, RT Professional, Comfort Panels und KTP Mobile Panels verfügbar.

Hinweis

Voraussetzung für die Anzeige in GRAPH-Übersicht

Um den Programmstatus eines S7-GRAPH-Instanzdatenbausteins im Objekt "GRAPH-Übersicht" anzeigen zu können, müssen die instanzspezifische Eigenschaften des Blocks als "Sichtbar in HMI" und "Erreichbar aus HMI" gesetzt sein.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Zugewiesene GRAPH DB-Variable
- Schaltflächen der Symbolleiste

Betriebsmodus

Für die Ausführung der GRAPH-Kette stehen Ihnen vier Betriebsmodi zur Verfügung:

- Auto (Standardeinstellung) - Schaltet automatisch in den nächsten Schritt, wenn die Transition erfüllt ist.
- TAP - Schaltet in den nächsten Schritt, wenn die Transition erfüllt ist und Flankenwechsel von "0" auf "1" am Parameter T_PUSH vorliegt.
- TOP - Schaltet in den nächsten Schritt, wenn die Transition erfüllt ist oder Flankenwechsel von "0" auf "1" am Parameter T_PUSH vorliegt.
- Manuell - Beim Erfüllen der Transition wird der nächste Schritt nicht automatisch aktiviert. Die Schritte können manuell ausgewählt und abgewählt werden.

Hinweis

Betriebsmodus setzten Sie durch Modifizieren der Schnittstellenparameter des GRAPH-Blocks in Ihrem Steuerungsprogramm.

In WinCC Runtime Professional haben Sie die Möglichkeit, den Namen für den Betriebsmodus, der in der GRAPH-Übersicht angezeigt wird, anzupassen.

Dreizeiligen Modus mit Kriterienanalyse projektieren

Im dreizeiligen Modus sehen Sie in der GRAPH-Übersicht die zusätzlichen Zeilen mit dem vorherigen und nächsten Schritt sowie den ersten fehlerhaften Operanden der Kriterienanalyse.

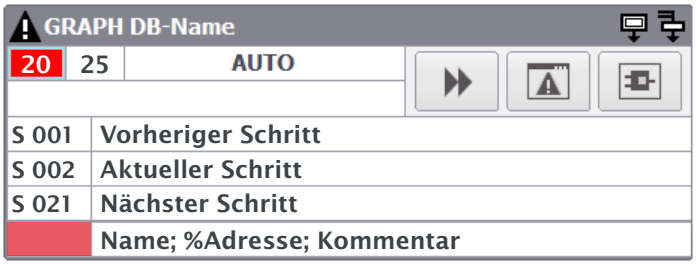

- Um den vorherigen und den nachfolgenden Schritt anzuzeigen, aktivieren Sie unter "Eigenschaften > Gestaltung > Optionen" die Option "Vorheriger und nächster Schritt".
- Um den ersten fehlerhaften Operanden der Kriterienanalyse anzuzeigen, aktivieren Sie unter "Eigenschaften > Gestaltung > Optionen" die Option "Kriterienanalyse".

Hinweis

Die Kriterienanalyse wird in folgenden Fällen keine Informationen in der GRAPH-Übersicht ausgeben:

- wenn die Version des GRAPH-Bausteins älter als 4.0 ist
- wenn die Erstwerterfassung nicht aktiviert wurde oder wenn die Erstwerte nicht erfasst wurden
- wenn im aktuellen Schritt kein Fehler ansteht

Kompakte Ansicht projektieren

Die GRAPH-Übersicht können Sie auch kompakt ohne Schaltflächen der Symbolleiste und Betriebsmodusanzeige projektieren, wie es bereits in WinCC Professional vor Version V14 möglich war.

Um die GRAPH-Übersicht kompakt im einzeiligen Kompatibilitätsmodus anzuzeigen, aktivieren Sie unter "Eigenschaften > Gestaltung > Optionen" die Option "Einzeilieger Modus".

Symbole

Die Symbole, die in der GRAPH-Übersicht angezeigt werden, legen Sie im Inspektorfenster unter "Eigenschaften > Gestaltung > Optionen" fest. Folgende Symbole stehen für GRAPH-Übersicht zur Verfügung:

Symbol	Name	Funktion
	Fehler	Zeigt an, dass während der Ausführung eines Schritts ein Fehler aufgetreten ist.
	Initialschritt	Zeigt an, dass der aktuell ausgeführte Schritt der erste Schritt im GRAPH-Baustein ist.
	Simultanschritt	Zeigt an, dass es im GRAPH-Baustein neben dem aktuellen Schritt noch weitere Simultanschritte gibt.

Schaltflächen

Schaltflächen, die in der GRAPH-Übersicht angezeigt werden, legen Sie unter "Eigenschaften > Symbolleiste > Allgemein" fest.

Schaltfläche	Name	Funktion
	Nächster Schritt	Springt zum nächsten Schritt, wenn der GRAPH-Baustein mehr als vier Schritte beinhaltet.
	Letzter Schritt	Nach Erreichen des letzten Schritts springt zum ersten Schritt der Ablaufkette.
	Meldeanzeige	Öffnet bei Panels und RT Advanced das konfigurierte Meldefenster mit der Fehlermeldung. Öffnet bei RT Professional die konfigurierte Meldeanzeige mit der Fehlermeldung.
	PLC-Code-Anzeige	Öffnet die konfigurierte PLC-Code-Anzeige.

Aktuellen Programmstatus anzeigen

Eine PLC mit einem GRAPH-Instanzdatenbaustein ist angelegt und enthält mindestens eine PLC-Variable, die in HMI sichtbar und aus HMI erreichbar ist.

1. Ziehen Sie per Drag&Drop die GRAPH-Übersicht aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie die Auswahlschaltfläche unter "Prozess > Variable".
   Ein Dialog "Neues Objekt hinzufügen" öffnet sich.
4. Wählen Sie im Ordner "Programmbausteine" die entsprechende PLC.
5. Selektieren Sie die entsprechende PLC-Variable des GRAPH-Instanzdatenbausteins.
6. Sie haben die Möglichkeit die Eigenschaften im Objekt "GRAPH-Übersicht" mit Variablen zu dynamisieren und Zustände farbig zu kennzeichnen.
   Ausführliche Beschreibung zum Dynamisieren einer Eigenschaft finden Sie im Kapitel Dynamisieren von Bildern (Seite 210)

Hinweis

Wenn zwischen dem HMI-Gerät und der gewählten PLC keine Verbindung existiert, wird automatisch eine Verbindung erstellt.

Außerdem wird eine HMI-Variable erstellt, die mit der PLC-Variablen verbunden ist.

Siehe auch

Erstwerterfassung und Kriterienanalyse (Seite 1996)

Maschinen und Anlagen mit ProDiag überwachen (Seite 1933)

GRAPH-Übersicht projektieren (Seite 1951)

Status der GRAPH-Ablaufkette anzeigen (Seite 1948)

Eigenschaftsanimation vom Typ Bereich projektieren (Seite 234)

Objekte für WinCC Runtime Professional (Seite 344)

Dynamisieren von Bildern (Seite 210)

1.9.2.27 Handrad

Verwendung

Das Objekt "Handrad" ist ein Bedienelement für einige SIMATIC Mobile Panel. In Runtime geben Sie mit dem Objekt "Handrad" Inkrementalwerte ein.

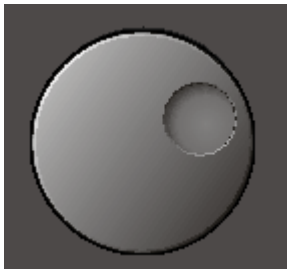

Darstellung

Im Inspektorfenster passen Sie die folgende Eigenschaft an:

- Variable: Legt fest, mit welcher Variablen das Objekt "Handrad" verbunden ist.

Globale Belegung

Mit der Eigenschaft "Variable" wird im Globalen Bild die globale Belegung des Objekts "Handrad" festgelegt. Pro Bediengerät gibt es nur ein Globales Bild.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie unter "Einstellungen > Variable" eine Variable aus der Auswahlliste.

Lokale Belegung

Sie belegen das Objekt "Handrad" mit einer anderen Variablen. Dazu projektieren Sie eine Funktion, die in Runtime dem Handrad die Variable zuweist.

Beispiel für die Projektierung an einer Schaltfläche:

1. Öffnen Sie das Bild, in dem das Objekt "Handrad" mit einer anderen Variablen belegt wird.
2. Ziehen Sie das Objekt "Schaltfläche" mit Drag&Drop aus dem Werkzeugfenster in das Bild. Markieren Sie die Schaltfläche im Bild.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse" auf "Klicken".

4. Der Dialog "Funktionsliste" erscheint. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen und Skripte erscheint.
5. Wählen Sie aus der Gruppe "Sonstige Funktionen" die Systemfunktion "VerbindeVariableMitHandrad". Wählen Sie bei "Wert" eine Variable aus der Auswahlliste.

Hinweis

Wenn während der Bedienung des Handrades der Wert in der Steuerung von einer anderen Stelle aus geändert wird, ignoriert das Mobile Panel möglicherweise die Änderung. Der Wert in der Steuerung wird wieder überschrieben.

Siehe auch

Leuchtdrucktaster (Seite 425)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.28 Hilfeindikator

Verwendung

Das Objekt "Hilfeindikator" ist für das Bediengerät KP300 Basic verfügbar. Wenn zum angewählten Objekt ein Tooltip existiert, dann wird der Hilfeindikator in Runtime eingeblendet. Wenn für das geöffnete Bild ein Tooltip projektiert wurde, ist der Hilfeindikator immer sichtbar.

Das Objekt "Hilfeindikator" projektieren Sie ausschließlich im Globalen Bild.

Darstellung

Im Inspektorfenster passen Sie folgende Eigenschaft an:

- Position: Legt die Position des Objekts "Hilfeindikator" fest.

Position

Mit dieser Eigenschaft legen Sie die Position des Objekts "Hilfeindikator" fest.

1. Markieren Sie in der Vorlage das Objekt "Hilfeindikator".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
3. Geben Sie bei X und Y die Werte ein. Alternativ können Sie das markierte Objekt mithilfe der Pfeiltasten positionieren.

Wenn Sie in einem Bild an diese Position ein Bildobjekt projektiert haben, überdeckt der eingeblendete Hilfeindikator das Bildobjekt. Der Hilfeindikator wird nur von auftretenden Systemmeldungen und Dialogen überdeckt.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.29 HTML-Browser

Verwendung

Das Objekt "HTML-Browser" ist für die Visualisierung von einfachen HTML Seiten konzipiert. Damit erstellen Sie zentral hinterlegte maschinenspezifische Beschreibungen, die von verschiedenen Bediengeräten aus angezeigt werden.

Hinweis

Das Umschalten der Funktionalität des HTML-Browsers als Datei-Explorer, z. B. auf folgenden Wegen, ist im Rahmen von WinCC nicht freigegeben:

- Eingabe eines Ordners oder Laufwerks, z. B. "\" oder "C:" oder
- Verbindung mit einem FTP-Server, z. B. "ftp://"

Grund ist unter anderem, dass es damit nicht möglich ist, Dateien unbeabsichtigt zu verändern, auszuführen oder zu löschen.

Achten Sie bei der Projektierung darauf, dass der Bediener nur gültige Internetadressen eingeben kann, z. B. über symbolische E/A-Felder. Für Servicezwecke projektieren Sie eine kennwortgeschützte Eingabe.

Hinweis

Aus Speicherplatzgründen wird bei Panels nur eine Instanz des HTML-Browsers pro Bild unterstützt.

Wenn Sie von einem Slide-in-Bild zu einem Pop-up-Bild schnell wechseln und in beiden Bildern jeweils ein HTML-Browser projektiert ist, wird der zweite HTML-Browser nicht geöffnet. Warten Sie nach dem Schließen des Slide-in-Bildes zwei Sekunden bis der Handle nicht mehr sichtbar ist und öffnen Sie dann das Pop-up-Bild.

Hinweis

Geräteabhängigkeit des Objekts "HTML-Browser"

Das Objekt "HTML-Browser" ist für Basic Panels, Comfort Panels, KTP Mobile Panels ab Displaygröße 7" und RT Advanced verfügbar.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position und Größe des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

- URL: Legt fest, welche Internetadresse im HTML-Browser geöffnet wird.
- Variable für URL: Legt die Variable fest, welche die Internetadresse vorgibt.

Adresse

Die Internetadresse wird im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > URL" festgelegt.

Hinweis

Für Basic Panels 2nd Generation:

Wenn Sie auf Ihr Bediengerät eine Webseite von einem USB-Stick laden wollen, muss die Adresse im Feld „URL" mit dem Präfix „\media\usb\" beginnen.

Beispiel: Sie wollen die Webseite html5.htm vom USB-Gerät USB_XY laden. Geben Sie im Inspektorfenster „\media\usb\USB_XY\html5.htm" ein.

Hinweis

Der DNS-Dienst für den Internet-Zugang mit dem Objekt "HTML-Browser" wird bei den Bediengeräten Basic Panels 2nd Generation nicht unterstützt.
Damit wird nur der Zugang über die IP-Adresse unterstützt.

Browsertyp (RT Advanced ab V13 SP1)

In Runtime Advanced ab Version V13 SP1 können Sie den Browsertyp auswählen.

Den Typ des Webbrowsers legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Typ HTML-Browser" fest.

Es stehen Ihnen zwei Optionen zur Verfügung:

- Webbrowser, basiert auf WebKit Engine, ohne Active X Unterstützung

  Hinweis

  Dialoge aus Webseiten werden im WebKit Browser nicht unterstützt. Stattdessen wird immer "Ja" oder "OK" als Antwort gesetzt.

- Webbrowser, basiert auf Internet Explorer mit Active X Unterstützung

  Hinweis

  Wenn der Internet Explorer die Webseite nicht lädt, benutzen Sie den WebKit Browser ohne Active X Unterstützung.

Prinzipielles Vorgehen

1. Legen Sie im Editor "Variable" eine interne Variable vom Datentyp "String" an, z. B. InternetAddresse.
2. Fügen Sie im Editor "Bilder" das Objekt "HTML-Browser" in das Bild ein.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Wählen Sie bei "Variable für die URL" die Variable "InternetAddresse" aus der Auswahlliste.
5. Fügen Sie im Editor "Bilder" ein E/A-Feld in das Bild ein.
6. Wählen Sie im Inspektorfenster des E/A-Felds "Eigenschaften > Eigenschaften > Allgemein > Variable für URL" die Variable "InternetAddresse" aus der Auswahlliste.

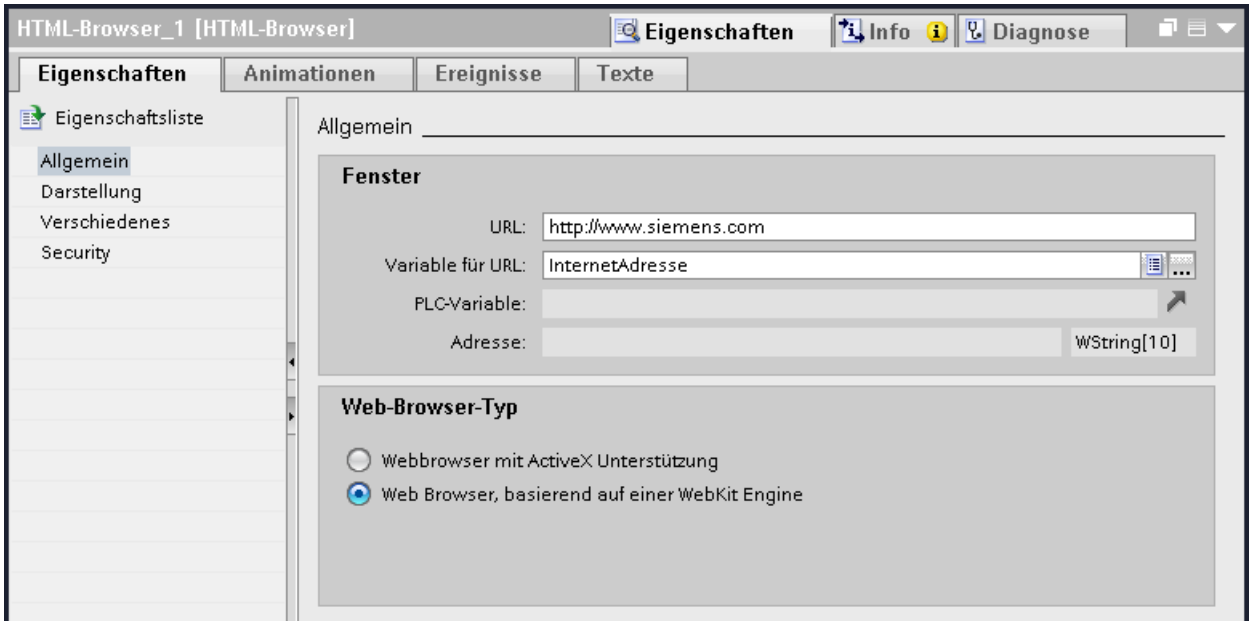

Wenn der Bediener im E/A-Feld eine Adresse eingibt, öffnet der HTML-Browser die betreffende Seite.

Bedienelemente

Bedienelemente sind projektierte Schaltflächen im Prozessbild oder Softkeys am Bediengerät.

Das Objekt "HTML-Browser" besitzt keine eigenen Bedienelemente. Projektieren Sie an die selbst projektierten Bedienelemente, z. B. Schaltflächen, die Systemfunktionen zum Bedienen des Objekts "HTML-Browser".

1. Ziehen Sie aus dem Werkzeugfenster z. B. das Objekt "Schaltfläche" in das Bild.
2. Geben Sie einen beliebig langen Text ein oder wählen Sie eine Grafik, z. B. Aktualisieren.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
   Der Dialog "Funktionsliste" wird geöffnet.

4. Die Systemfunktionen zum Bedienen des Objekts "HTML-Browser" finden Sie in der Liste der Systemfunktionen in der Gruppe "Tastaturbedienung für Bildobjekte".
   Wählen Sie aus der Liste eine Funktion, z. B. HTMLBrowserAktualisieren.
5. Wählen Sie aus der Liste "Bildobjekt" den Objektnamen des HTML-Browsers, in dem der Befehl ausgeführt wird.

Auswahl der Webseiten vorgeben

Um die Auswahl der Webseiten, die im HTML-Browser aufgerufen werden, vorzugeben, definieren Sie eine Anzahl von erlaubten Web-Adressen.

Sie definieren eine Variable vom Typ String oder WString für die Eigenschaft "Variable für URL" des HTML-Browsers, die über folgende Wege mit den vordefinierten Web-Adressen gefüllt werden kann:

- über die Systemfunktion "SetzeVariable", die Sie z.B. an einer Schaltfläche projektieren
- über ein E/A-Feld

  Hinweis

  Durch die Eingabe der Adresse in einem E/A-Feld kann auch jede beliebige Webseite aufgerufen werden.

  Um die Auswahl der erlaubten Webseiten einzuschränken, nutzen Sie die Übergabe der vordefinierten Webadressen über eine Systemfunktion oder ein Skript.

- über ein Skript (nur Panels und RT Advanced)
- von der PLC aus

SIMOTION PLC Webseiten im HTML-Browser anzeigen

Falls die Webseiten von SIMOTION PLC im HTML-Browser nicht richtig angezeigt werden, fügen Sie nach der Webseiten-URL "/basic" ein, um die Webseiten im Basismodus anzuzeigen.

Bemerkungen

Gegenüber dem Internet Explorer hat der HTML-Browser einen eingeschränkten Funktionsumfang:

- Der HTML-Browser unterstützt nur die Anzeige von reinen HTML-Seiten. VBScript, Java, JavaScript , Flash und ActiveX-Controls werden nicht unterstützt. Erstellen Sie die HTML-Seiten zur Anzeige im HTML-Browser mit einem Texteditor oder mit einem einfachen HTML-Editor.
- Links auf eingebettete Dateien, z. B. *.pdf oder *.xls, werden nicht unterstützt.
- Abfragen und Dialoge, die beim Zugriff auf z. B. geschützte Seiten angezeigt werden, werden nicht unterstützt. Geben Sie beim Zugriff auf geschützte Seiten in dem URL Benutzername und Kennwort mit ein: <http://Benutzername:Kennwort@Servername> (z. B. "http://Administrator:Admin123@192.168.0.199").

Hinweis

Das Objekt "HTML-Browser" unterstützt Dynamisierung.

Wenn Sie den HTML-Browser-Typ "Webbrowser, basierend auf einer WebKit Engine" auswählen, ist für die Dynamisierung unter "Adresse" nur die Variablenanbindung verfügbar.

Hinweis

Für Basic Panels 2nd Generation und Comfort Panels:

Das Objekt "HTML-Browser" unterstützt keine Proxy-Einstellungen. Damit kann das Objekt "HTML-Browser" auf keine URLs zugreifen, die Änderung der Proxy-Einstellungen benötigen.

Hinweis

HTML-Browser im Modus "Bildlaufanzeiger"

Wenn Sie unter "Runtime-Einstellungen > Bilder > Scrolling in Controls" den Bildlaufmodus "Bildlaufanzeiger" festgelegt haben, kann es vorkommen, dass manche Webseiten nicht richtig angezeigt werden.

Hinweis

HTML-Browser im Modus "Web-Browser, basiert auf WebKit Engine" mit Bildlaufleisten

Sie projektieren das Objekt "HTML-Browser" im Modus "Web-Browser, basiert auf WebKit Engine" und aktivieren den Bildlaufmodus für die Anzeige. In Runtime sind die rechte und die untere Leisten für den Bildlauf aktiviert.

Wenn in Runtime die Bildlaufleisten der angezeigten HTML-Seite und des Objekts "HTML-Browser" überlappen, bedienen Sie die Bildlaufleisten des HTML-Browsers mit einem Mausklick und die Bildlaufleisten der Webseite mit Doppelklick. Versuchen Sie bereits bei der Projektierung des Objekts im HMI-Bild die doppelten Bildlaufleisten zu vermeiden, indem Sie das Objekt der Größe nach an die anzuzeigenden Webseiten anpassen.

Siehe auch

Webbrowser der WebKit-Engine: Übersicht (Seite 2328)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.30 HTML-Browser

Verwendung

Das Objekt "HTML-Browser" ist für die Visualisierung von einfachen HTML Seiten konzipiert. Damit erstellen Sie zentral hinterlegte maschinenspezifische Beschreibungen, die von verschiedenen Bediengeräten aus angezeigt werden.

Hinweis

Das Umschalten der Funktionalität des HTML-Browsers als Datei-Explorer, z. B. auf folgenden Wegen, ist im Rahmen von WinCC nicht freigegeben:

- Eingabe eines Ordners oder Laufwerks, z. B. "\" oder "C:" oder
- Verbindung mit einem FTP-Server, z. B. "ftp://"

Grund ist unter anderem, dass es damit möglich ist, Dateien unbeabsichtigt zu verändern, auszuführen oder zu löschen.

Achten Sie bei der Projektierung darauf, dass der Bediener nur gültige Internetadressen eingeben kann, z. B. über symbolische E/A-Felder. Für Servicezwecke projektieren Sie eine kennwortgeschützte Eingabe.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position und Größe des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

- Adresse: Legt fest, welche Internetadresse im HTML-Browser geöffnet wird.
- Symbolleiste: Legt fest, ob eine Symbolleiste für die Navigation angezeigt wird.

Adresse

Die Internetadresse wird im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Fenster > URL" festgelegt.

Prinzipielles Vorgehen

1. Legen Sie im Editor "Variable" eine interne Variable vom Datentyp "String" an, z. B. InternetAddress.
2. Fügen Sie im Editor "Bilder" das Objekt "HTML-Browser" in das Bild ein.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Wählen Sie bei "Variable für die URL" die Variable "InternetAddress" aus der Auswahlliste.
5. Fügen Sie im Editor "Bilder" ein E/A-Feld in das Bild ein.
6. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Variable für URL" die Variable "InternetAddress" aus der Auswahlliste.

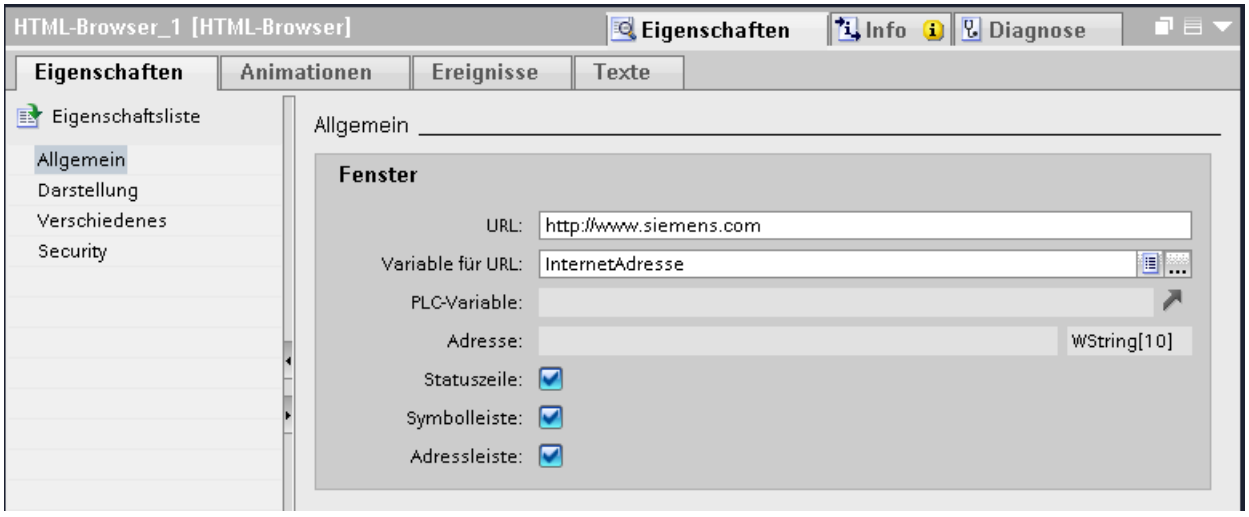

Wenn der Bediener eine Adresse eingibt oder aktiviert, öffnet der HTML-Browser die betreffende Seite.

Bemerkungen

Gegenüber dem Internet Explorer hat der HTML-Browser einen eingeschränkten Funktionsumfang:

- Der HTML-Browser unterstützt nur die Anzeige von reinen HTML-Seiten. VBScript, Java, JavaScript, Flash und ActiveX-Controls werden nicht unterstützt. Erstellen Sie die HTML-Seiten zur Anzeige im HTML-Browser mit einem Texteditor oder mit einem einfachen HTML-Editor.
- Links auf eingebettete Dateien, z. B. *.pdf oder *.xls, werden nicht unterstützt.
- Abfragen und Dialoge, die beim Zugriff auf z. B. geschützte Seiten angezeigt werden, werden nicht unterstützt. Geben Sie beim Zugriff auf geschützte Seiten in dem URL Benutzername und Kennwort mit ein: <http://Benutzername:Kennwort@Servername> (z. B. "http://Administrator:Admin123@192.168.0.199").

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.31 Kamera-Anzeige

Verwendung

Mit dem Objekt "Kamera-Anzeige" projektieren Sie die Kamera-Anzeige, auf der dem Bediener in Runtime Bilder von einer angeschlossenen Netzwerkkamera ausgegeben werden.

Die Überwachung mithilfe der Netzwerkkamera bietet sich an, wenn der Bediener eingeschränkte Sicht auf die Anlage hat oder ein Aufenthalt in Maschinennähe nicht möglich ist. Beim Auftreten von Störungen reagiert der Bediener dann auf eine mögliche Störungsursache schneller.

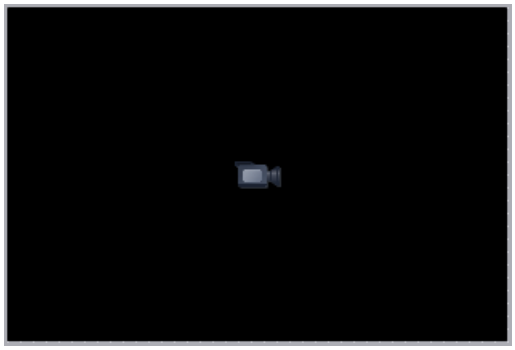

Das Projektieren der Kamera-Anzeige ist in folgenden Bildern möglich:

- In einem Bild
- In einer Vorlage
- In einem Slide-in-Bild
- In einem Pop-up-Bild

Hinweis

Geräteabhängigkeit des Objekts "Kamera-Anzeige"

Das Objekt "Kamera-Anzeige" ist für Comfort Panels, KTP Mobile Panels (ab Geräteversion V13) und RT Advanced verfügbar.

Hinweis

In Runtime darf nur eine Kamera-Anzeige zur selben Zeit in Betrieb sein.

Hinweis

Verwenden Sie das Objekt "Kamera-Anzeige" nur mit Netzwerkkameras.
Achten Sie darauf, dass die angeschlossene Netzwerkkamera sich in einem gesicherten Netzwerk mit bekannten Teilnehmern und IP-Adressen befindet.

Weitere Informationen zu Security finden Sie unter:

http://www.siemens.com/industrialsecurity (http://www.industry.siemens.com/topics/global/de/industrial-security/Seiten/Default.aspx)

Darstellung

Im Inspektorfenster machen Sie Angaben zu der Quelle des Video-Streams sowie zu der Position und Größe der Kamera-Anzeige. Insbesondere passen Sie die folgenden Eigenschaften an:

- URL für die Quelle des Video-Streams:
  - Kamera URL: Legt eine statische URL fest, die nur beim Projektieren eingegeben wird.
  - Kamera URL Variable: Legt eine dynamische URL fest, die der Bediener nachträglich in Runtime ändern kann.
- Übertragungsprotokoll: Legt das Übertragungsprotokoll fest, über das der Datenaustausch zwischen dem Panel und der Netzwerkkamera erfolgt.
- Position und Größe der Kamera-Anzeige: Legt Position, Größe der Kamera-Anzeige sowie Seitenverhältnis und Größe des angezeigten Video-Streams fest.

Unterstützte Netzwerkkameras und empfohlene Einstellungen

Verwenden Sie das Objekt "Kamera-Anzeige" mit den Netzwerkkameras, die das Streaming-Protokoll RTP/RTSP und Codecs MJPEG, H.264 und MPEG4 unterstützen.

Informationen zu den unterstützten Netzwerkkameras und empfohlenen Einstellungen finden Sie unter Unterstützte Netzwerkkameras und Einstellungen. (Seite 2296)

URL für die Quelle des Video-Streams festlegen

1. Fügen Sie im Editor "Bilder" die Kamera-Anzeige in das Bild ein. Die Kamera-Anzeige finden Sie in der Task Card "Werkzeuge > Controls".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Medien".

3. Geben Sie in das Feld "Kamera URL" die statische Kamera URL ein, z. B. "rtsp://user:password@192.168.56.208" für eine Siemens-Kamera.
   Die Adressierung der Netzwerkkamera entnehmen Sie dem Handbuch der Netzwerkkamera.

   Hinweis

   Die Kamera URL fängt immer mit "rtsp://" an und darf nur erlaubte Zeichen enthalten. Wenn nicht erlaubte Zeichen eingegeben werden oder die URL nicht mit "rtsp://" anfängt, ist in Runtime kein Kamera-Betrieb möglich. Der Bediener wird durch eine Fehlermeldung in der "Kamera-Anzeige" darüber informiert.

   Folgende Zeichen des US-ASCII-Zeichensatzes sind erlaubt:

   0123456789

   A...Z a...z

   !#$%&()*+,'-_./:;<=>?@ [\]_{|}~^(")`

4. Weisen Sie im Inspektorfenster unter "Kamera URL Variable" der Kamera-Anzeige eine Variable zu. Die Variable muss bei externen Variablen vom Datentyp "String" und bei internen Variablen vom Datentyp "WString" sein.
   Wenn Sie der Kamera-Anzeige eine Variable zuweisen, kann der Benutzer in Runtime die Quelle des Video-Streams ändern.

Hinweis

Wenn Sie das Feld "Kamera URL Variable" leer lassen, wird die statische URL in Runtime verwendet.

Wenn Sie unter "Kamera URL" und "Kamera URL Variable" Angaben machen, wird die dynamische URL in Runtime verwendet.

Übertragungsprotokoll festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Medien".
2. Geben Sie über das Optionskästchen "UDP anstatt TCP verwenden" das Übertragungsprotokoll vor.
   Die Auswahl des Protokolls hängt von den Protokollarten ab, die Ihre Netzwerkkamera unterstützt. Die Information zu den unterstützten Protokollen entnehmen Sie dem Handbuch der Netzwerkkamera.
   TCP-Verbindung hat in den meisten Fällen eine zuverlässige Synchronisierung mit der Kamera-Anzeige.

Position und Größe der Kamera-Anzeige festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Darstellung".
2. Stellen Sie im Bereich "Position und Größe" die Größe des Objekts "Kamera-Anzeige" ein. Um die bestmögliche Bildqualität des Video-Streams zu erzielen, geben Sie für die Größe der Kamera-Anzeige die physikalische Auflösung der Netzwerkkamera ein.
   Im Bereich "Größenanpassung" ist die empfohlene Einstellung "Videogröße beibehalten" standardmäßig aktiviert. Das Optionskästchen "Seitenverhältnis beibehalten" ist deaktiviert und ist nicht auswählbar.
3. Um das Seitenverhältnis des Video-Streams beizubehalten, deaktivieren Sie "Videogröße beibehalten".
   Das Optionskästchen "Seitenverhältnis beibehalten" ist automatisch aktiviert.

Hinweis

Durch die Beibehaltung der Größe des Video-Streams erfolgt keine Skalierung des Bildes. Die Bildgröße der Netzwerkkamera wird 1:1 über die Kamera-Anzeige ausgegeben. Wenn die Kamera-Anzeige kleiner als die gewählte Auflösung der Netzwerkkamera ist, wird nur ein entsprechender Ausschnitt des Kamerabildes über die Kamera-Anzeige ausgegeben. Wenn die Kamera-Anzeige größer als die gewählte Auflösung der Netzwerkkamera ist, werden frei bleibende Flächen als horizontale bzw. vertikale schwarze Ränder ausgegeben. Für die optimale Leistung auf Panels empfehlen wir die Verwendung der Option "Videogröße beibehalten" und Konfiguration der Kamera-Anzeige passend zur Auflösung der Kamera, um schwarze Ränder in der Kamera-Anzeige zu vermeiden.

Durch die Beibehaltung des Seitenverhältnisses des Video-Streams werden mögliche frei bleibende Flächen als horizontale bzw. vertikale schwarze Ränder über die Kamera-Anzeige ausgegeben.

Hinweis

Wenn Sie die Optionskästchen "Videogröße beibehalten" und "Seitenverhältnis beibehalten" deaktivieren, kann es zu Verzerrungen des Kamerabildes führen.

Bemerkungen

Hinweis

Für die Kamera-Anzeige sind keine Systemfunktionen projektierbar.

Hinweis

Wenn Sie ein Projekt mit dem Simulator testen, wird ein schwarzes Bild mit einem Kamera-Symbol ausgegeben.

Siehe auch

Objekte für WinCC Runtime Advanced (Seite 343)

Unterstützte Netzwerkkameras und Einstellungen (Seite 2296)

1.9.2.32 Kanaldiagnose-Anzeige

Verwendung

Sie verwenden das Objekt "Kanaldiagnose-Anzeige" zur Überwachung der aktiven Verbindungen in Runtime. Mit dem Objekt "Kanaldiagnose-Anzeige" geben Sie folgende Status- und Diagnoseinformationen der Kanal-Units aus:

- Ausgabe von Status-/Statistik-Informationen der Verbindungen in Runtime
- Textausgabe in Logbuch-Datei zur Störungsanalyse und Fehlerbehebung
- Textausgabe in Trace-Datei zur Unterstützung der Hotline bei der Eingrenzung von Verbindungsproblemen

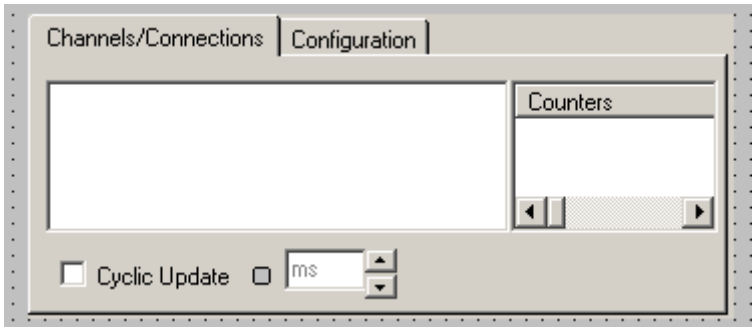

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts.

Kanal und Verbindung prüfen

Aktivieren Sie dazu Ihr Projekt. Im Objekt "Kanaldiagnose-Anzeige" werden auf der Registerkarte Channels/Connections Status-Informationen zu allen projektierten Verbindungen angezeigt.

1. Öffnen Sie ein Bild.
2. Fügen Sie aus der Objektpalette "Controls" die "Kanaldiagnose-Anzeige" in das Bild ein.
3. Speichern Sie das Bild.
4. Aktivieren Sie Runtime.
5. Wählen Sie das Bild an, in dem Sie die Kanaldiagnose-Anzeige projektiert haben.
6. Auf der Registerkarte "Channels/Connection" werden die projektierten Verbindungen angezeigt.
7. Prüfen Sie die Symbole vor dem Namen der Verbindung. Wenn vor dem jeweiligen Eintrag ein grünes Häkchen angezeigt wird, dann ist der Status des Kanals oder der Verbindung in Ordnung.

8. Wenn vor dem Namen von Kanal und Verbindung kein grünes Häkchen erscheint, klicken Sie auf eine Verbindung. Im rechten Bereich werden die Statusinformationen dieser Verbindung angezeigt.
9. Wechseln Sie auf die Registerkarte "Configuration". Wählen Sie einen der angezeigten Kanäle aus und konfigurieren Sie, welche Fehleranzeigen in die zugehörige Log-Datei eingetragen werden sollen.
10. Prüfen Sie die kanalspezifische Log-Datei. Öffnen Sie dazu die Log-Datei im Ordner "Siemens\Automation\SCADA_2007\WinCC\Diagnose" mit einem Texteditor.
11. Prüfen Sie die neuesten Einträge mit dem Flag "ERROR".
12. Aktivieren Sie die Trace-Funktion.
    Wenn Sie den Fehler auch nach der Prüfung der Log-Datei nicht lokalisieren können, setzen Sie sich mit dem Customer Support in Verbindung.

Übersicht der Statusanzeigen

Folgende Statusanzeigen werden im Objekt "Kanaldiagnose-Anzeige" angezeigt:

- Kanal / Verbindung ohne Einschränkung betriebsbereit
- Kanal / Verbindung mit Einschränkung betriebsbereit
- Keine Aussage über Verbindungsstatus möglich
- Kanal / Verbindung gestört

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.33 Kombinationsfeld

Verwendung

Sie verwenden das Objekt "Kombinationsfeld" zur Darstellung und Auswahl von Einträgen in einer Aufklappliste. Sie aktivieren einen Eintrag standardmäßig, damit der Bediener den voreingestellten Wert nur bei Bedarf ändert. Um Kombinationsfelder in den Prozess einzubinden, dynamisieren Sie die entsprechenden Eigenschaften.

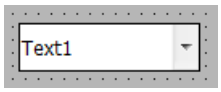

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl der Einträge: Legt die Anzahl des Kombinationsfelds fest.
- Auswahl der Einträge: Legt fest, welcher Eintrag voreingestellt aktiviert dargestellt wird.

Anzahl der Einträge festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Legen Sie unter "Beschriftung > Einträge" die gewünschte Anzahl an Listenfeldern fest.

Der Wert des Attributs "Einträge" gibt gleichzeitig den oberen Grenzwert für die Eigenschaft "Index" unter "Beschriftung" an. Die Änderung des Werts kann folgende Auswirkungen haben:

Verändern der Anzahl

Wenn Sie die Anzahl der Einträge erhöhen, werden neue Felder unter dem Feld mit dem höchsten Wert der Eigenschaft "Index" eingefügt.

Wenn Sie die Anzahl der Einträge verringern, werden alle Felder entfernt, bei denen der Wert der Eigenschaft "Index" höher ist als die neue Anzahl.

Siehe auch

Listenfeld (Seite 430)

Bedienmeldungen projektieren (Seite 862)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.34 Kontrollkästchen

Verwendung

Sie verwenden das Objekt "Kontrollkästchen" zur Darstellung und Auswahl von mehreren Optionen. Sie aktivieren Kontrollkästchen standardmäßig, damit der Bediener den voreingestellten Wert nur bei Bedarf ändert. In Runtime kann der Bediener mehrere Optionen auswählen. Um Kontrollkästchen in den Prozess einzubinden, dynamisieren Sie die entsprechenden Eigenschaften.

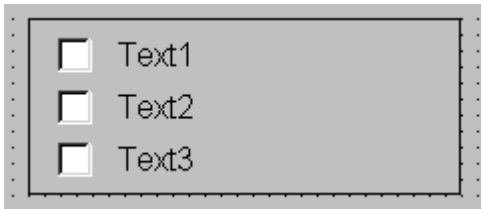

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl der Kontrollkästchen: Legt die Anzahl der Optionen fest.
- Auswahl der Kontrollkästchen: Legt fest, welche Optionen voreingestellt aktiviert dargestellt werden.

Anzahl der Kontrollkästchen festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Legen Sie unter "Beschriftung > Einträge" die gewünschte Anzahl an Kontrollkästchen fest.

Der Wert des Attributs gibt gleichzeitig den oberen Grenzwert für die Eigenschaft "Index" unter "Beschriftung" an. Die Änderung des Werts kann folgende Auswirkungen haben:

- Erhöhung der Anzahl
  Neue Felder werden unter dem Feld mit dem höchsten Wert der Eigenschaft "Index" eingefügt. Sie ändern die Standardbeschriftung des neuen Feldes mit der Eigenschaft "Text" unter "Beschriftung".
- Verringerung der Anzahl
  Alle Felder werden entfernt, bei denen der Wert der Eigenschaft "Index" höher ist als die neue Anzahl.

Voreinstellung der Kontrollkästchen festlegen

In der Tabelle unter "Eigenschaften > Eigenschaften > Allgemein > Beschriftung > Voreingestellt aktiviert " legen Sie fest, welche Optionen in einer Kontrollkästchenliste aktiviert dargestellt werden. Sie können mehrere Optionen aktivieren.

Jede Option wird durch ein Bit in einem 32-Bit-Wort repräsentiert. Um eine Option zu aktivieren, muss das zugehörige Bit den Wert "1" besitzen. Das 32-Bit-Wort enthält die Informationen für alle Optionen der Kontrollkästchenliste. Der Wert der Eigenschaft "Voreingestellt aktiviert" wird in hexadezimaler Schreibweise angegeben.

Siehe auch

Optionsschaltflächen (Seite 456)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.35 Kreis

Verwendung

Das Objekt "Kreis" ist ein geschlossenes Objekt, das mit einer Farbe oder einem Muster gefüllt werden kann.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius: Legt die Größe des Kreises fest.

Radius

Den Radius des Objekts "Kreis" wird im Inspektorfenster festgelegt. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radius" einen Wert zwischen 0 und 2500 ein.

Siehe auch

Kreisbogen (Seite 417)

Kreissegment (Seite 418)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.36 Kreisbogen

Verwendung

Das Objekt "Kreisbogen" ist ein offenes Objekt. Wenn Sie das Objekt mit Farbe füllen wollen, verwenden Sie das Objekt "Kreissegment". Ein Kreisbogen besteht standardmäßig aus einem Viertel-Kreis, den Sie beliebig anpassen.

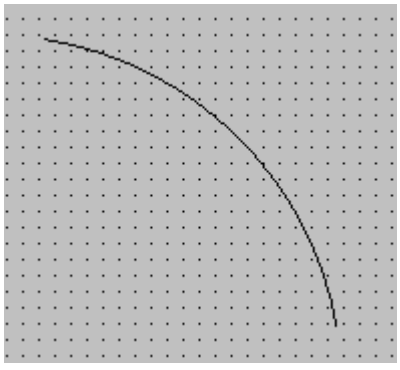

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius: Legt die Größe des Kreisbogens fest.
- Anfangs- und Endwinkel: Legen fest, wo Anfangs- und Endpunkt auf einem virtuellen Kreis von 360° liegen.

Radius festlegen

Sie legen im Inspektorfenster den horizontalen und den vertikalen Radius des Objekts "Kreisbogen" fest. Sie geben die Werte in der Einheit Pixel ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Bogen" bei "Radius" einen Wert ein.

Anfangs- und Endwinkel festlegen

Sie bestimmen die Länge des Kreisbogens durch die Attribute "Anfangswinkel" und "Endwinkel". Sie geben die Werte in der Einheit Grad ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radien" bei "Anfangswinkel" und "Endwinkel" jeweils einen Wert ein.

Alternatives Vorgehen

Die Anfangs- und Endpunkte des Kreisbogens können Sie auch mit der Maus ändern.

1. Positionieren Sie den Mauszeiger auf einen Anfangs- oder Endpunkt des Kreisbogens. Anfangs- und Endpunkt sind durch kleine blaue Quadrate symbolisiert.
   Der Mauszeiger verwandelt sich in ein Kreuz.
2. Ziehen Sie den Anfangs- oder Endpunkt bei gedrückter Maustaste auf die gewünschte Position.

Siehe auch

Kreis (Seite 416)

Kreissegment (Seite 418)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.37 Kreissegment

Verwendung

Das Objekt "Kreissegment" ist ein geschlossenes Objekt, das Sie mit einer Farbe oder einem Muster füllen können. Ein Kreissegment besteht standardmäßig aus einem Viertel-Kreis, den Sie beliebig anpassen.

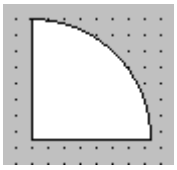

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius: Legt die Größe des Kreissegments fest.
- Anfangs- und Endwinkel: Legen fest, wo der Anfangs- und Endpunkt auf einem virtuellen Kreis mit 360° liegt.

Radius

Sie legen im Inspektorfenster den Radius des Objekts "Kreissegment" fest. Den Wert geben Sie in der Einheit Pixel ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Darstellung".
2. Geben Sie im Bereich "Bogen" bei "Radius" einen Wert ein.

Anfangs- und Endwinkel festlegen

Die Größe des Kreissegments bestimmen Sie durch die Attribute "Anfangswinkel" und "Endwinkel". Die Werte geben Sie in der Einheit Grad ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radien" bei "Anfangswinkel" und "Endwinkel" jeweils einen Wert ein.

Alternatives Vorgehen

Die Anfangs- und Endpunkte des Kreissegments können Sie auch mit der Maus ändern.

1. Positionieren Sie den Mauszeiger auf einem Anfangs- oder Endpunkt des Kreissegments. Anfangs- und Endpunkt sind durch kleine blaue Quadrate symbolisiert.
   Der Mauszeiger verwandelt sich in ein Kreuz.
2. Ziehen Sie den Anfangs- oder Endpunkt bei gedrückter Maustaste auf die gewünschte Position.

Siehe auch

Kreis (Seite 416)

Kreisbogen (Seite 417)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.38 Kriterienanalyse-Anzeige

Verwendung

Das Objekt "Kriterienanalyse-Anzeige" zeigt Ihnen die fehlerhaften Operanden im Anwenderprogramm, die eine ausgewählte ProDiag- oder GRAPH-Meldung ausgelöst haben. Somit haben sie die Möglichkeit, zusätzlich zur Meldung die Liste der fehlerhaften Operanden im gleichen Bild zu sehen.

Um die Auswertung der Kriterienanalyse im Objekt "Kriterienanalyse-Anzeige" in Runtime zu sehen, aktivieren Sie die Erstwerterfassung in den Einstellungen der Funktionsbausteine im Anwenderprogramm. Erstwerterfassung steht für GRAPH-Funktionsbausteine ab Version 4.0 und ProDiag-Funktionsbausteine ab Version 2.0 zur Verfügung.

Um die Verbindung zur entsprechenden Fehlermeldung zu ermöglichen, projektieren Sie eine Referenz zu einer bereits konfigurierten Meldeanzeige. Wenn Sie in Runtime eine GRAPH-Meldung oder eine ProDiag-Meldung in der Meldeanzeige auswählen, wird in der Kriterienanalyse-Anzeige der Name, die Adresse, der Kommentar und der Wert des Operanden angezeigt, der diesen Fehler verursacht hat.

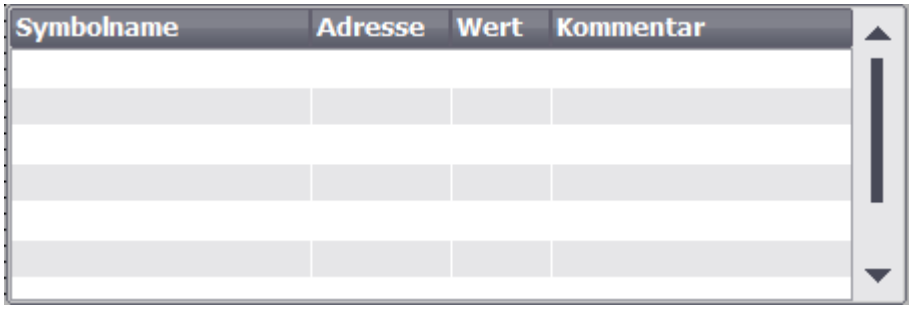

Die kommenden Meldungen und die fehlerhaften Operanden sehen Sie in Runtime auf einen Blick, wenn Sie die Kriterienanalyse-Anzeige und die damit verbundene Meldeanzeige im gleichen Bild projektieren.

Hinweis

Kriterienanalyse steht nur für die Anwenderprogramme zur Verfügung, bei welchen Erstwerterfassung aktiviert wurde.

Erstwerterfassung aktivieren Sie in den Eigenschaften folgender Bausteine:

- ProDiag-Funktionsbausteine mit der Version größer oder gleich 2.0
- GRAPH-Funktionsbausteine mit der Version größer oder gleich 4.0

Hinweis

Das Objekt "Kriterienanalyse-Anzeige" steht im Globalen Bild nicht zur Verfügung.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Stil, Farben und Schriftarten des Objekts.

Spalten

Folgende Spalten werden in der Kriterienanalyse-Anzeige in Runtime angezeigt:

Spalte	Beschreibung
Symbolname	Der symbolische Name des Operanden im Anwenderprogramm.
Adresse	Die absolute Adresse des Operanden.
Kommentar	Die zusätzlichen Kommentare aus dem Anwenderprogramm in der Sprache, die in die Steuerung geladen ist.
Wert	Der Wert des Operanden zum Zeitpunkt des Fehlers.

Kriterienanalyse-Anzeige projektieren

1. Ziehen Sie per Drag&Drop die Kriterienanalyse-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie unter der Eigenschaft "Variable" die Auswahlschaltfläche.
4. Selektieren Sie die Zustandsvariable der entsprechenden Meldeanzeige.
   Um die Verbindung zu Meldungen zu ermöglichen, geben Sie bei der projektierten Meldeanzeige unter "Eigenschaften > Eigenschaften > Anzeige" ebenfalls die Variable für die Kriterienanalyse ein.
5. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
6. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen.
7. Passen Sie bei Bedarf die Überschriften und die Breite der Spalten an.

Kriterienanalyse-Anzeige projektieren (RT Professional)

1. Ziehen Sie per Drag&Drop die Kriterienanalyse-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Geben Sie unter "Datenquelle" den Namen der konfigurierten Meldeanzeige ein.

Kriterienanalyse-Anzeige mit der Tastatur bedienen

Um die PLC-Code-Anzeige mit der Tastatur zu bedienen, gehen Sie folgendermaßen vor:

1. Drücken Sie so oft die Taste <Tab>, bis sich der Fokus auf der gewünschten Schaltfläche befindet.
   Mit der Tastenkombination <Umschalttaste + Tab> navigieren Sie rückwärts zur vorherigen Schaltfläche.
2. Drücken Sie die Taste <Enter>.
   Der Befehl wird ausgeführt.

Siehe auch

Überblick über die Erstwerterfassung und Kriterienanalyse (Seite 1996)

Beispiel für die Reihenfolge der Kriterienanalyse (Seite 2001)

Erstwerterfassung und Kriterienanalyse (Seite 1996)

1.9.2.39 Kurvenanzeige

Verwendung

Die Kurvenanzeige dient zur grafischen Darstellung von Variablenwerten aus dem aktuellen Prozess oder aus einem Archiv in Form von Kurven.

Darstellung

Im Inspektionsfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Wertetabelle, Lineal, Raster anzeigen: Legt fest, ob zusätzlich zum Koordinatensystem eine Wertetabelle, ein Lineal oder Raster zur leichteren Ablesbarkeit eingeblendet wird.
- Symbolleisten: Legt die Anzeige der Bedienelemente fest.

Wertetabelle, Lineal und Raster anzeigen

Zur leichteren Ablesbarkeit zeigen Sie in Runtime eine Wertetabelle, ein Lineal und ein Raster an.

1. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung > Lineal anzeigen".
2. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle anzeigen".
3. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Tabelle > Raster anzeigen".

Symbolleisten

Die Darstellung der Bedienelemente legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste" fest.

Hinweis

Basic Panels

Da die Archivierung bei Basic Panels erst ab Geräten der 2nd Generation möglich ist, stehen die Bedienelemente nicht auf allen Basic Panels zur Verfügung.

Schaltfläche Symbolleiste	Kurzbeschreibung	Beschreibung
	"Gehe zu Start"	Blättert zum aktuellen Wert der Kurve.
	"Vergrößern"	Verkleinert den dargestellten Zeitabschnitt.
	"Verkleinern"	Vergrößert den dargestellten Zeitabschnitt.
	"Lineal rückwärts"	Bewegt das Lineal zurück.
	"Lineal vorwärts"	Bewegt das Lineal vorwärts.
	"Rückwärts"	Blättert um eine Anzeigenbreite zurück.
	"Vorwärts"	Blättert um eine Anzeigenbreite vorwärts.
	"Lineal"	Blendet das Lineal ein oder aus. Das Lineal zeigt den zu einem X-Wert gehörenden Y-Wert an.
	"Start/Stopp "	Hält die Kurvenaufzeichnung an oder setzt die Kurvenaufzeichnung fort.

Bedienung auf Basic Panels und Geräten mit kleiner Displaygröße

Auf Basic Panels und Bediengeräten mit einer Displaygröße kleiner als 6" werden die Schaltflächen der Kurvenanzeige nicht angezeigt. Sie können die Kurvenanzeige über Funktionstasten bedienen, indem Sie dafür entsprechende Systemfunktionen an die Funktionstasten des Bediengeräts projektieren.

Anzeige von Spaltenüberschriften

Die Darstellung der Tabelle in der Kurvenanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften in der Tabelle abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Konsistenzprüfung

Wenn bei einer Konsistenzprüfung an Kurvenanzeigen Warnungen oder Fehler im Ausgabefenster angezeigt werden, wird nicht in allen Fällen mit dem Befehl "Gehe zu Fehler/ Variable" an die exakte Fehlerstelle gesprungen. Eventuell wird nur die Kurvenanzeige als Fehlerursache angezeigt.

Kurven hinzufügen, konfigurieren und entfernen

Die Kurven der Kurvenanzeige verwalten Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Kurve". Sie können Kurven zwischen verschiedenen Kurvenanzeigen kopieren.

Siehe auch

Kurven (Seite 706)

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 709)

Kurvenanzeige projektieren (Seite 728)

Kurven für das Kurvenfenster erstellen (Seite 732)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.40 Ladezustand

Verwendung

Das Objekt "Ladezustand" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "Ladezustand" zeigt den Ladezustand der Batterie an. Laden Sie die Batterie rechtzeitig. Alternativ können Sie die Batterie durch eine Ersatzbatterie austauschen.

Darstellung

Symbol	Bedeutung	Ladezustand
	Batterie ist aufgeladen	>20 %
	Batterie muss aufgeladen werden	<20 % und > 6 %
	Batterie ist schwach	< 6 %

Bedienung

Das Objekt dient zur reinen Anzeige und kann nicht bedient werden.

Siehe auch

Wirkbereichs-Name (Seite 525)

Wirkbereichs-Name (RFID) (Seite 528)

Wirkbereichs-Signal (Seite 529)

WLAN-Empfang (Seite 531)

Zonen-Name (Seite 534)

Zonen-Signal (Seite 535)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.41 Leuchtdrucktaster

Verwendung

Das Objekt "Leuchtdrucktaster" ist ein Bedienelement für einige SIMATIC Mobile Panel. Die Leuchtdiode kann von der Steuerung aus angesteuert werden. Durch eine leuchtende LED signalisieren Sie dem Bediener in Runtime, dass er den Leuchtdrucktaster drücken muss. Den Leuchtdrucktaster projektieren Sie im Globalen Bild.

Darstellung

Im Inspektorfenster passen Sie die folgenden Eigenschaften an:

- Variable: Legt fest, in welche Variable der Zustand des Leuchtdrucktasters geschrieben wird.
- LED-Zuordnung: Legt die Zuordnung der LED zu dem Bit in der LED-Variablen fest.

Variable

Mit der Eigenschaft "Variable" wird im Globalen Bild die globale Belegung des Objekts "Leuchtdrucktaster" festgelegt.

Hinweis

Hinweise zur Projektierung

Beim Start von Runtime und bei jedem Drücken des Tasters wird der aktuelle Zustand des Leuchtdrucktasters in die Variable geschrieben. Dabei kann es zu Inkonsistenzen zwischen dem Zustand des Leuchtdrucktasters und der Leuchtdrucktaster-Variablen kommen.

1. Wenn die Variable eine Verbindung zur Steuerung hat, wird nach Kommunikationsaufbau der aktuelle Wert aus der Steuerung in die Variable übertragen. Die Variable, die den aktuellen Zustand des Tasters enthält, wird dabei überschrieben. Deshalb spiegelt die Leuchtdrucktaster-Variable eventuell nicht mehr den aktuellen Wert des Leuchtdrucktasters wider, z. B. wenn das Mobile Panel ausgeschaltet wird, während der Leuchtdrucktaster gedrückt war.
2. Wenn der Leuchtdrucktaster gedrückt wird, bevor die Kommunikation zur Steuerung aufgebaut werden konnte (z. B. nach einem Geräte-Anlauf), kann der Wert der Leuchtdrucktaster-Variablen eventuell nicht zur Steuerung geschickt werden. In diesem Fall stimmt dann der Wert der Variablen in der Steuerung nicht mit dem aktuellen Zustand des Tasters überein.

Damit der Leuchtdrucktaster immer den tatsächlichen Zustand wiedergibt, führen Sie die folgenden Projektierungsschritte aus.

Prinzipielles Vorgehen

1. Legen Sie im Editor "Verbindungen" die Verbindung zur Steuerung an. Aktivieren Sie den Bereichszeiger "Koordinierung", damit das Lebensbit auf der Steuerungsseite verfügbar ist.

Parameter | Bereichszeiger

Aktiv	Anzeigename	PLC-Variable	Zugriffsart	Adresse	Länge	Erfassungsart
☑	Koordinierung	<Undefiniert>	<Absoluter Zugriff>	%DB1.DBW0	1	Zyklisch fortl
☐	Datum/Uhrzeit	<Undefiniert>	<Symbolischer Zugriff>		6	Zyklisch fortl
☐	Steuerungsauftr..	<Undefiniert>	<Symbolischer Zugriff>		4	Zyklisch fortl
☐	Datensatz	<Undefiniert>	<Symbolischer Zugriff>		5	Zyklisch fortl

Globale Bereichszeiger des Bediengeräts

Verbindung	Anzeigename	PLC-Variable	Zugriffsart	Adresse	Länge	Erfassungs
<Undefiniert>	Projektkennung	<Undefiniert>	<Symbolischer Zugriff>		1	Zyklisch fo
<Undefiniert>	Bildnummer	<Undefiniert>	<Symbolischer Zugriff>		5	Zyklisch fo
<Undefiniert>	Datum/Uhrzeit PLC	<Undefiniert>	<Symbolischer Zugriff>		6	Zyklisch fo

2. Öffnen Sie den Editor "HMI-Variablen" und legen Sie drei Variablen an.
   - Interne Variable: Pushbutton_State
   - Externe Variable: BufferTag
   - Externe Variable: Pushbutton_PLC
3. Öffnen Sie die das Inspektorfenster der Variablen "Pushbutton_State"
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Wertänderung". Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen erscheint.

5. Wählen Sie die Systemfunktion "SetzeVariable".
   – Wählen Sie bei "Variable (Ausgabe)" die Variable "Pushbutton_PLC".
   – Wählen Sie bei "Wert" die Variable "Pushbutton_State".

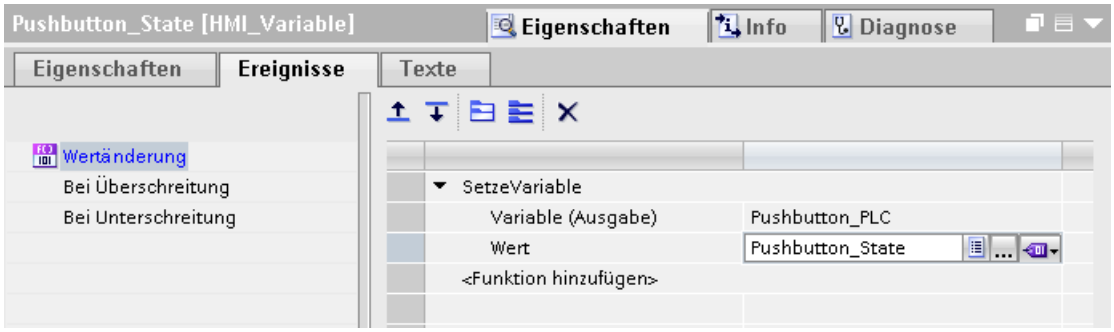

Mit der Variablen "Pushbutton_PLC" wird der Wert der Variablen "Pushbutton_State" in die Steuerung geschrieben. In der Steuerung wertet ein Programm das Lebensbit aus. Wenn die Kommunikation aufgebaut ist, wird der aktuelle Wert aus der Steuerung in die Variable "Pushbutton_PLC" geschrieben. Damit der aktuelle Zustand des Tasters an die Steuerung übergeben wird, wird die Variable "BufferTag" benötigt.

6. Öffnen Sie das Inspektorfenster der Variablen "BufferTag".
7. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Wertänderung".
8. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen erscheint.
9. Wählen Sie die Systemfunktion "SetzeVariable".
   – Wählen Sie bei "Variable (Ausgabe)" die Variable "Pushbutton_PLC".
   – Wählen Sie bei "Wert" die Variable "Pushbutton_State".

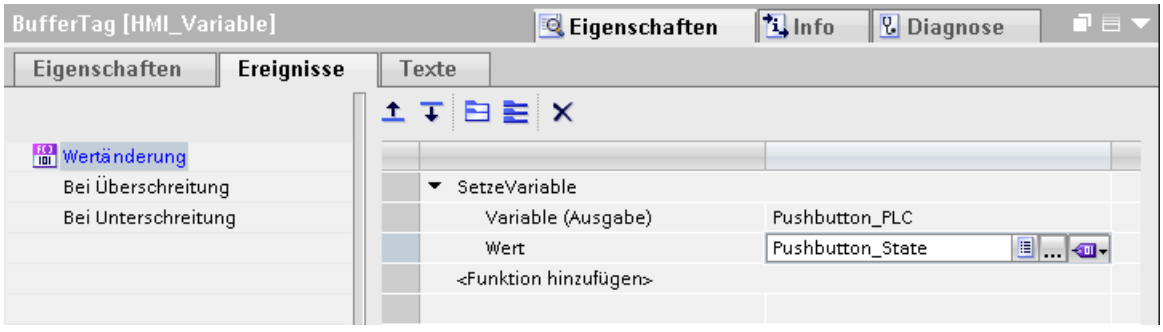

Nach Aufbau der Kommunikation wird der aktuelle Wert aus der Steuerung in die Variable "BufferTag" geschrieben. Durch die Wertänderung in der Hilfsvariablen wird die Systemfunktion "SetzeVariable" ausgeführt. Die Systemfunktion weist der Variablen "Pushbutton_PLC" wieder den Wert der Variablen "Pushbutton_State" zu.

10. Öffnen Sie im Editor "Bildnavigation" das Globale Bild.

11. Markieren Sie im Globalen Bild die Leuchttaste.
12. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein >Einstellungen > Variable" die Variable "Pushbutton_State".
    Wenn Sie den Leuchtdrucktaster bedienen, wird der Wert in die Variable "Pushbutton_PLC" geschrieben.

LED-Zuordnung

Um die Leuchtdiode anzusteuern, muss in der Steuerung eine LED-Variable oder eine Arrayvariable eingerichtet und bei der Projektierung als LED-Variable angegeben werden. Die LED-Variable wird in der Gruppe "Allgemein" im Bereich "Einstellungen" bei "LED-Variable" festgelegt. Bei "LED-Bit" wird das zugeordnete Bit angegeben. Durch das Setzen des Bits können in Runtime folgende Zustände der LED realisiert werden.

Bit n+1	Bit n	LED-Funktion
0	0	Aus
0	1	Blinken schnell
1	0	Blinken langsam
1	1	Dauerlicht

Siehe auch

Handrad (Seite 400)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.42 Linie

Verwendung

Das Objekt "Linie" ist ein offenes Objekt. Länge und Neigung der Linie bestimmen Sie durch Höhe und Breite des objektumfassenden Rechtecks.

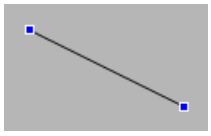

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Linienstil
- Linienanfang und Linienende

Linienstil

Die Darstellung der Linie legen Sie Im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" fest. Wenn Sie z. B. "Massiv" wählen, wird die Linie durchgezogen dargestellt.

Hinweis

Die verfügbaren Linienstile sind abhängig vom gewählten Bediengerät.

Linienanfang und Linienende

Den Anfangspunkt und den Endpunkt der Linie legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Linienenden" fest.

Als Anfangs- bzw. Endpunkte verwenden Sie z. B. Pfeilspitzen. Welche Anfangs- und Endpunkte Ihnen zur Verfügung stehen, ist geräteabhängig.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.43 Listenfeld

Verwendung

Sie verwenden das Objekt "Listenfeld" zur Darstellung und Auswahl von mehreren Listeneinträgen. Sie aktivieren Listeneinträge standardmäßig, damit der Bediener den voreingestellten Eintrag nur bei Bedarf ändert. Wenn das Listenfeld größer ist als das objektumfassende Rechteck, fügt WinCC automatisch eine Laufleiste am rechten Rand hinzu.

Um Listenfelder in den Prozess einzubinden, dynamisieren Sie die entsprechenden Eigenschaften.

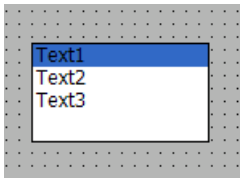

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl der Einträge: Legt die Anzahl der Listeneinträge fest.
- Auswahl der Einträge: Legt fest, welcher Eintrag voreingestellt aktiviert dargestellt wird.

Anzahl der Einträge festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Unter "Einträge" legen Sie die gewünschte Anzahl der Listeneinträge fest.

Der Wert des Attributs "Einträge" gibt gleichzeitig den oberen Grenzwert für die Eigenschaft "Index" unter "Beschriftung" an. Die Änderung des Werts kann folgende Auswirkungen haben:

Verändern der Anzahl

Wenn Sie die Anzahl der Einträge erhöhen, werden neue Felder unter dem Feld mit dem höchsten Wert der Eigenschaft "Index" eingefügt. Sie ändern die Standardbeschriftung des neuen Feldes mit der Eigenschaft "Text" unter "Beschriftung".

Wenn Sie die Anzahl der Einträge verringern, werden alle Felder entfernt, bei denen der Wert der Eigenschaft "Index" höher ist als die neue Anzahl.

Voreinstellung der Listenfelder festlegen

Mit "Eigenschaften > Eigenschaften > Allgemein > Voreingestellt aktiviert" legen Sie fest, welcher Listeneintrag aktiviert dargestellt wird.

Jede Option wird durch ein Bit in einem 32-Bit-Wort repräsentiert. Um eine Option zu aktivieren, muss das zugehörige Bit den Wert "1" besitzen. Das 32-Bit-Wort enthält die Informationen für alle Texte der Listenfelderliste. Der Wert der Eigenschaft "Selektierte Felder" wird in hexadezimaler Schreibweise angegeben.

Siehe auch

Kombinationsfeld (Seite 414)

Bedienmeldungen projektieren (Seite 862)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.44 Media Player

Verwendung

In Runtime dient der Media Player zum Abspielen von Multimedia-Dateien.

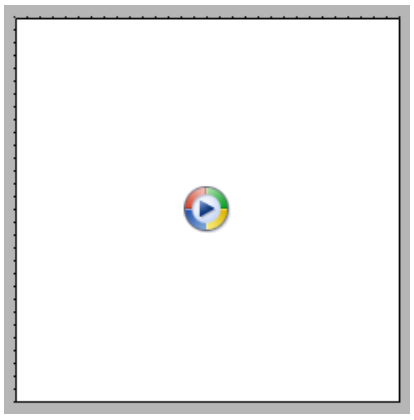

Der Media Player ist nur für folgende Geräte verfügbar:

- Comfort Panels
- RT Professional

Hinweis

Videos auf Comfort Panels abspielen

Auf Comfort Panels können Sie im Bildobjekt "Media Player" Videosequenzen abspielen. Nähere Hinweise zum Abspielen von Videos finden Sie im Internet unter: http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/de/62101921) (Beitrags-ID 62101921)

Darstellung

Im Inspektorfenster können Sie die folgenden Eigenschaften anpassen:

- Statuszeile anzeigen: Legt fest, ob die Statuszeile angezeigt wird.
- Steuerelemente anzeigen: Legt die Bedienelemente in Runtime fest.
- Zeitleiste anzeigen: Legt fest, ob ein Schieberegler für die Bedienung zur Verfügung steht.

Bedienelemente

Die Bedienelemente, mit denen der Media Player in Runtime bedient werden kann, werden im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Elemente" festgelegt.

Anzahl Wiedergaben (Comfort Panels)

Um die Datei mehrere Male abzuspielen, legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Anzahl Wiedergaben" die Anzahl der Wiederholungen der Wiedergabe fest. Wählen Sie einen Wert zwischen 0 und 2.147.483.647.

Unterstützte Dateiformate (Comfort Panels)

Folgende Audio-Formate werden vom Media Player unterstützt:

- Moving Picture Experts Group standard 1, Layer 1,2, 3 (.mpa, .mp2, .mp3)
- Waveform Audio (.wav)

Folgende Video-Formate werden vom Media Player unterstützt:

- Moving Picture Experts Group standard 1 (.mpg, .mpeg, .mpv, .mpe)
- Advanced Streaming Format (.asf)
- Windows Media Video Format (.wmv)
- Audio-Video Interleaved (.avi)

Hinweis

Der Media Player unterstützt nicht das Anlegen und Verwalten von Favoriten. Wenn Sie Favoriten anlegen und den Media Player schließen, werden die Favoriten beim erneuten Öffnen nicht mehr angezeigt.

Hinweis

Der Media Player ist nicht mit der Tastatur bedienbar.

Eine Simulation des Media Players ist nicht möglich. Im Simulationsfenster wird anstelle des Media Players ein statisches Bild angezeigt.

Unterstützte Dateiformate (RT Professional)

Folgende Formate werden vom Media Player unterstützt:

- Grafikformate (.gif, .bmp, .jpg, .jpeg, .png)
- Moving Picture Experts Group standard 1 (.mpg, .mpeg, .mp4)
- QuickTime Movie (.mov, .qt)
- Advanced Streaming Format (.asf)
- Windows Media Video Format (.wmv)
- Audio Video Interleave (.avi)

Hinweis

Voraussetzung bei Videodateien

Um Videodateien in Windows Server 2008 R2 SP1 und 2012 R2 abzuspielen, installieren Sie das Microsoft-Feature "Desktop Experience". Weiterführende Informationen zu diesem Thema finden Sie im Internet in der Dokumentation von Microsoft.

Hinweis

Die Wiedergabe der Multimedia-Dateien im Control ist neben dem Dateiformat auch von den auf dem Rechner installierten Video- und Audio-Codecs abhängig.

Hinweis

Datenverlust beim Kopieren des Projekts

Wenn Sie das Projekt auf einen anderen Rechner kopieren, beachten Sie Folgendes:

Im WinCC Media Control angegebene Dateien werden nicht mitkopiert, wenn sie dynamisch verlinkt und nicht mit UNC-Pfad angegeben sind. Sie müssen die Dateien erneut in das Projekt aufnehmen.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.45 Meldeanzeige

Verwendung

Das Objekt "Meldeanzeige" zeigt Meldungen an, die während des Prozesses in einer Anlage auftreten. Die Meldeanzeige nutzen Sie auch, um Meldungen in Listen darzustellen. WinCC bietet verschiedene Anzeigen an, z. B. Anzeige "Aktuelle Meldungen" oder "Kurzzeitarchiv".

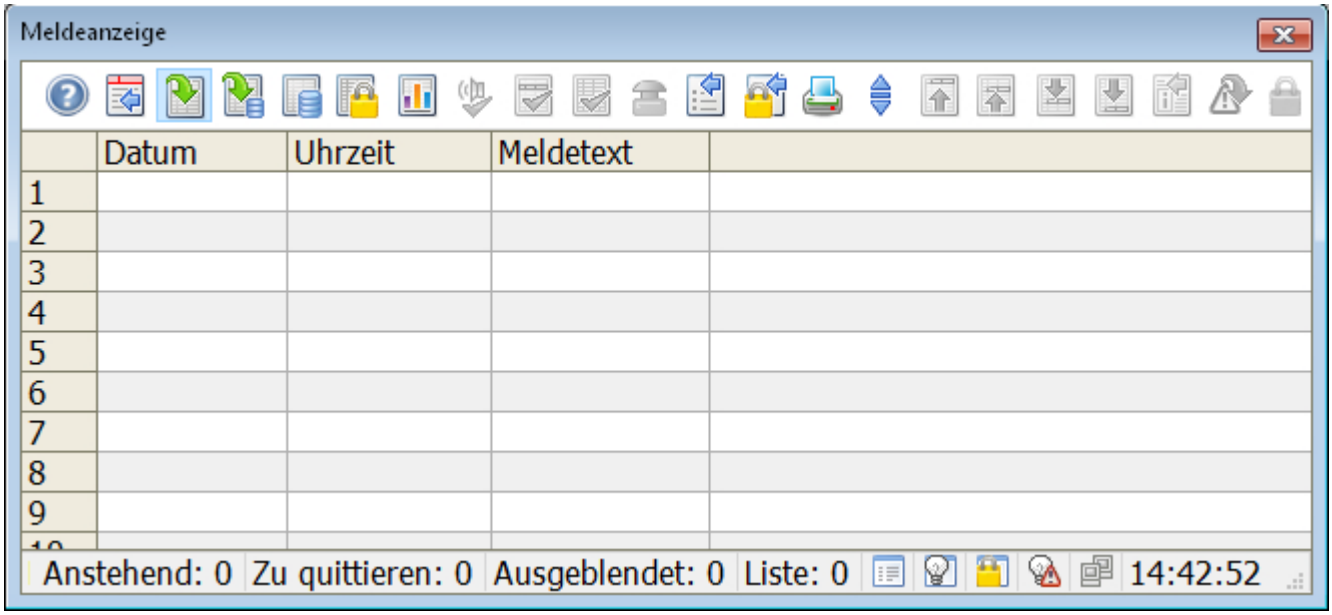

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Bedienelemente: Legt die Bedienelemente der Meldeanzeige fest.
- Statuszeile: Legt die Elemente der Statuszeile fest.
- Zeitbasis: Legt fest, auf welcher Zeitbasis die Meldungen ausgegeben werden.
- Autoscroll: Legt fest, ob immer die aktuellste Meldung in der Meldeanzeige markiert wird.
- Textformat: Legt Schriftart und -größe fest.
- Sortierung: Legt fest, ob und wie in Runtime die Meldungen sortiert werden können.
- Tabelle: Legt die Eigenschaften der Tabelle in der Meldeanzeige fest.
- Loop-In-Alarm: Legt fest, dass mit Doppelklicken auf eine Meldung ein projektierter Bildwechsel ausgelöst wird.

Hinweis

Das Objekt "Meldeanzeige" lässt sich nicht gruppieren.

Sie projektieren nur an das Ereignis "Kommen" ein Skript.

Ausgabe der Meldungen projektieren

- Auswahl der Anzeigen und Filter: Legt fest, welche Meldungen in dieser Meldeanzeige angezeigt werden.
- Meldestatistik: Aktiviert die Anzeige einer statistischen Übersicht zu Anzeigedauer und Häufigkeit der Meldungen.
- Drucken: Legt fest, welcher Druckauftrag für die Meldeanzeige verwendet wird.

Bedienelemente

Die Bedienelemente der Meldeanzeige in Runtime und deren Bedienberechtigungen legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste" fest. Folgende Bedienelemente stehen für die Meldeanzeige zur Verfügung:

Schaltfläche		Funktion
	Anzeige "Aktuelle Meldungen"	Zeigt die aktuell anstehenden Meldungen an.
	Anzeige "Kurzzeitarchiv"	Zeigt die archivierten und anstehenden Meldungen an.
	Anzeige "Langzeitarchiv"	Zeigt die archivierten Meldungen an.
	Anzeige "Gesperrte Meldungen"	Zeigt alle im System gesperrten Meldungen an.
	Anzeige "Meldestatistik"	Zeigt eine statistische Übersicht zu Anzeigedauer und Häufigkeit der Meldungen.
	"Quittierung zentraler Melder"	Quittiert einen optischen oder akustischen Signalgeber.
	"Einzelquittierung"	Quittiert eine einzelne Meldung.
	"Sammelquittierung"	Quittiert alle anstehenden, sichtbaren und quittierpflichtigen Meldungen in der Meldeanzeige, wenn sie nicht einzelquittierpflichtig sind.
	"Autoscroll"	Legt fest, dass immer die aktuellste Meldung in der Meldeanzeige markiert wird. Der sichtbare Bereich der Meldeanzeige wird bei Bedarf verschoben. Nur wenn "Autoscroll" nicht aktiviert ist, markieren Sie Meldungen individuell.
	"Filter"	Öffnet einen Dialog zum Filtern von Meldungen in der Meldeanzeige. Auch die Meldungen, die in der gefilterten Anzeige nicht angezeigt werden, werden archiviert.

Schaltfläche		Funktion
	"Dialog Anzeigeoptionen"	Öffnet einen Dialog zum Festlegen, welche Meldungen in der Meldeanzeige angezeigt werden. Wenn die Option "Alle Meldungen" aktiviert ist, werden in der Meldeanzeige sowohl ausgeblendete als auch eingeblendete Meldungen angezeigt. Wenn die Option "Eingeblendete Meldungen" aktiviert ist, werden in der Meldeanzeige ausschließlich eingeblendete Meldungen angezeigt. Wenn die Option "Ausgeblendete Meldungen" aktiviert ist, werden in der Meldeanzeige ausschließlich ausgeblendete Meldungen angezeigt.
	"Sperr-Dialog"	Öffnet einen Dialog zum Sperren von Meldungen. Alle Meldungen, die diese Kriterien erfüllen, werden nicht angezeigt und nicht archiviert
	"Druck Meldeprotokoll"	Druckt das Meldeprotokoll, das für die Meldeanzeige projektiert ist.
	"Druck aktuelle Ansicht"	Startet den Ausdruck der in der Meldeanzeige angezeigten Meldungen.
	"Not-Quittierung"	Not-quittiert eine quittierpflichtige Meldung. Auch wenn die Meldung nicht ansteht, wird das Quittierbit der Meldung direkt an das AS geschickt.
	"Erste Meldung"	Markiert die erste der anstehenden Meldungen. Der sichtbare Bereich der Meldeanzeige wird bei Bedarf verschoben. Nur wenn die Funktion "Autoscroll" deaktiviert ist, ist diese Schaltfläche bedienbar.
	"Letzte Meldung"	Markiert die letzte der anstehenden Meldungen. Der sichtbare Bereich der Meldeanzeige wird bei Bedarf verschoben. Nur wenn die Funktion "Autoscroll" deaktiviert ist, ist diese Schaltfläche bedienbar.
	"Nächste Meldung"	Markiert die nächste Meldung, ausgehend von der aktuell markierten Meldung. Der sichtbare Bereich der Meldeanzeige wird bei Bedarf verschoben. Nur wenn die Funktion "Autoscroll" deaktiviert ist, ist diese Schaltfläche bedienbar.
	"Vorherige Meldung"	Markiert die vorherige Meldung, ausgehend von der aktuell markierten Meldung. Der sichtbare Bereich der Meldeanzeige wird bei Bedarf verschoben. Nur wenn die Funktion "Autoscroll" deaktiviert ist, ist diese Schaltfläche bedienbar.
	"Hilfetext-Dialog"	Öffnet einen Dialog zur Anzeige eines Hilfetexts.
	"Kommentar-Dialog"	Öffnet einen Texteditor zum Eintragen von Kommentaren. Nur wenn die Anzeige "Langzeitarchiv" aktiviert ist, steht diese Schaltfläche in der Meldeanzeige zur Verfügung.
	"Loop-In-Alarm"	Wenn ein Bildwechsel projektiert ist, wechselt die Anzeige zu einem Bild, das Informationen zur Meldung enthält.
	"Meldung freigeben"	Entsperrt in der Anzeige "Gesperrte Meldungen" eine Meldung.
	"Meldung sperren"	Sperrt eine Meldung in der aktuellen Meldeliste und in den Meldearchivlisten. Die Meldung wird in die Anzeige "Gesperrte Meldungen" eingetragen.

Schaltfläche		Funktion
	"Sortier-Dialog"	Öffnet einen Dialog zum Einstellen einer benutzerdefinierten Sortierung der angezeigten Meldungen.
	"Zeitbasis-Dialog"	Öffnet einen Dialog zum Wählen der Zeitbasis für die in den Meldungen angezeigten Zeitangaben.
	"Liste der ausgeblende-ten Meldungen"	Öffnet die Anzeige "Anzeigeunterdrückte Meldungen".
	"Meldung ausblenden"	Unterdrückt die Anzeige einer Meldung in den Anzeigen "Aktu-elle Meldungen", "Kurzzeitarchiv" oder "Langzeitarchiv". Die Meldung wird in die Anzeige "Anzeigeunterdrückte Meldungen" eingetragen.
	"Meldung einblenden"	Blendet in der Anzeige "Anzeigeunterdrückte Meldungen" eine Meldung wieder ein in die Anzeigen "Aktuelle Meldungen", "Kurzzeitarchiv" oder "Langzeitarchiv". Die Meldung wird aus der Anzeige "Anzeigeunterdrückte Meldungen" entfernt.

Statuszeile

Die Anzeige der Elemente der Statuszeile legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Statuszeile" fest. Folgende Anzeigeelemente stehen für die Meldeanzeige zur Verfügung:

Schaltfläche	Funktion
Datum	Systemdatum
Uhrzeit	Systemzeit
	Sperre ist gesetzt.
	Ein Filter ist aktiviert
Liste:	Anzahl der aktuellen Meldungen
Fenster:	Anzahl der Meldungen im Fenster
Quit:	Anzahl der anstehenden, quittierpflichtigen Meldungen

Zugriffsschutz in Runtime

Im Inspektorfenster "Eigenschaften > Eigenschaften > Sicherheit" projektieren Sie den Zugriffsschutz. Wenn ein angemeldeter Benutzer die erforderliche Berechtigung besitzt, kann er mithilfe der Bedienelemente in der Meldeanzeige Meldungen quittieren und bearbeiten.

In folgenden Fällen wird der Anmeldedialog eingeblendet:

- Der angemeldete Benutzer versucht, die Meldung zu bearbeiten und hat nicht die erforderliche Berechtigung.
- Ein Benutzer versucht, die Meldung zu bearbeiten, während kein Benutzer angemeldet ist.

Zeitbasis einstellen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie unter "Eigenschaften > Eigenschaften > Verschiedenes > Zeitbasis" die gewünschte Zeitbasis.
3. Wählen Sie unter "zeitliche Standardsortierung", ob die Meldungen in Runtime aufsteigend oder absteigend sortiert werden.

Autoscroll aktivieren

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Fenster > Automatischer Bildlauf".

Textformat festlegen

Um den Meldetext zu formatieren:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Textformat".
2. Klicken Sie im Bereich "Schriftart" auf die Schaltfläche "...".
   Ein Dialog wird geöffnet.
3. Wählen Sie im Dialog Schriftart und -größe.

Sortierung einrichten

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Spalten".

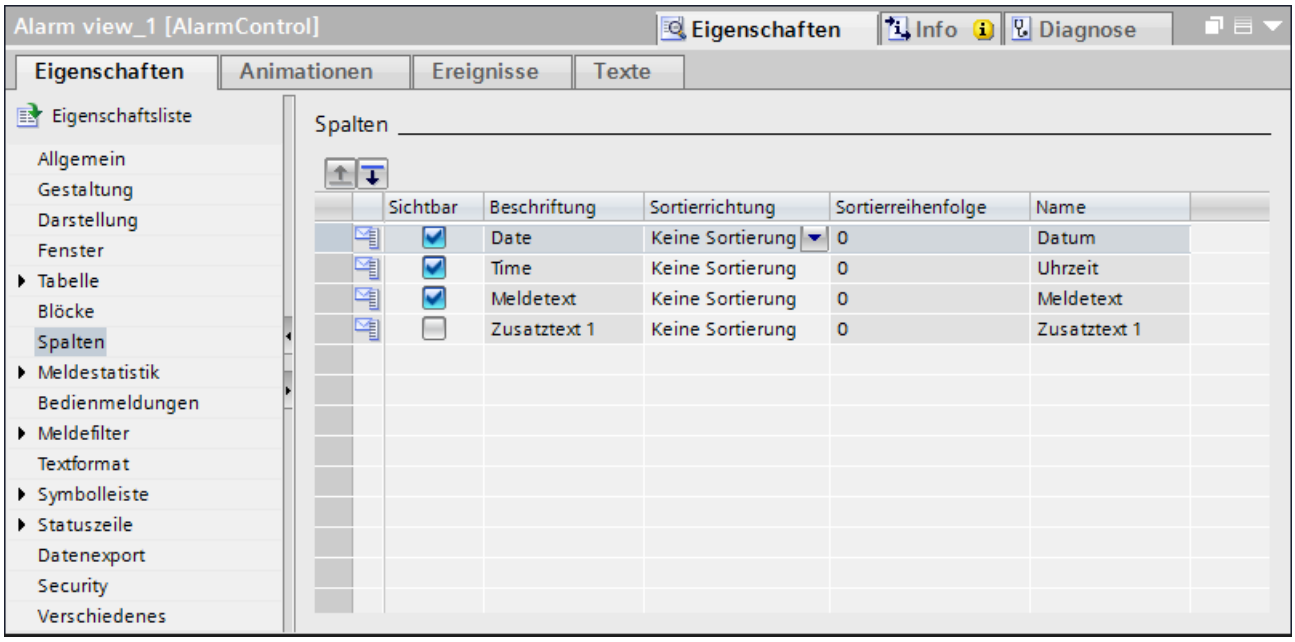

2. Wählen Sie Sortierkriterien und Sortierreihenfolge aus.

Tabelle in der Meldeanzeige einrichten

1. Klicken Sie im Inspektorfenster unter " Eigenschaften > Eigenschaften > Schriftart in Tabelle".
2. Legen Sie im Bereich "Rumpf Tabelle" Einstellungen für die Zeilen und Zellen fest.
   – "Optimale Zeilenhöhe" aktiviert: Die Schriftgröße passt sich an die Höhe der Zeile an.
   – "Zelleninhalte abschneiden": Wenn der Text länger ist als die Zelle, wird der Text abgeschnitten dargestellt.
3. Wählen Sie unter "Zeilennummerierung", ob jede Zeile eine Überschrift hat.
4. Wählen Sie unter "Überschrift Spalten" die Darstellung der Spaltenüberschrift aus.

Loop-In-Alarm einstellen

1. Klicken Sie im Inspektorfenster " Eigenschaften > Eigenschaften >Verschiedenes".
2. Aktivieren Sie die Option "Loop-In-Alarm bei Doppelklick".

Anzeige von Spaltenüberschriften

Die Darstellung der Meldeanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Registerkarte "Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Siehe auch

Meldeanzeige für aktuelle Meldungen projektieren (Seite 872)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.46 Meldeanzeige (ab V13)

Verwendung

Erweiterte Meldeanzeige dient zur Darstellung von Meldungen am Bediengerät.

Auf den Bediengeräten mit der Geräteversion V13 oder höher steht Ihnen zur Verwaltung von Meldungen die erweiterte tabellenbasierte Meldeanzeige zur Verfügung.

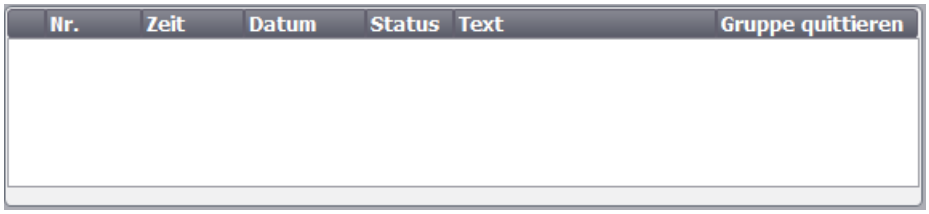

Meldezeile

Die Meldezeile ermöglicht in Runtime die Darstellung der aktuellsten anstehenden Meldung. Das folgende Bild zeigt eine Meldezeile:

So projektieren Sie die Meldezeile:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Darstellung".
2. Wählen Sie unter "Modus > Meldezeile".

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts.

Hinweis

Die Auswahl an Schriftarten ist abhängig davon, welche Schriftarten Sie in den Runtime-Einstellungen unter "Sprache & Schriftart" projektiert haben und welche Schriftarten Ihr Bediengerät unterstützt.

Insbesondere passen Sie die folgenden Eigenschaften an:

- Symbolleiste: Legt die Bedienelemente der Meldeanzeige fest.
- Meldeklassen: Legt fest, welche Meldeklassen in der Meldeanzeige angezeigt werden.
- Spalten: Legt die in Runtime darzustellenden Spalten fest.
- Spalten verschiebbar: Legt fest, ob die Reihenfolge der Spalten in Runtime geändert werden kann.
- Kennzeichnung der Meldeklassen: Um verschiedene Meldeklassen unterscheiden zu können, kennzeichnen Sie diese in der ersten Spalte der Meldeanzeige.
- Filter: Legt fest, dass nur Meldungen angezeigt werden, die im Meldetext eine bestimmte Zeichenkette enthalten.

- Anzeigebereich festlegen: Legt fest, ab welchem Datum Meldungen in der Meldeanzeige angezeigt werden.
- Sortieren nach Datum/Uhrzeit möglich: Legt fest, ob in Runtime die Sortierung der Meldungen nach Datum und Uhrzeit möglich ist.

Hinweis

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

Bedienelemente

Die Bedienelemente, mit welchen die Meldeanzeige in Runtime bedient werden kann, werden im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste" festgelegt. Die folgende Tabelle zeigt die Bedienelemente der Meldeanzeige und ihre Funktionen:

Schaltfläche	Bezeichnung	Funktion
	"Infotext"	Zeigt Tooltip für eine Meldung an.
	"Quittieren"	Quittiert eine Meldung.
	"Loop-in-alarm"	Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1)]

[1)] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Meldeklassen auswählen

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie unter "Meldeklassen" die Meldeklassen, die in der Meldeanzeige in Runtime dargestellt werden sollen.

Zugriffsschutz in Runtime

Den Zugriffsschutz projektieren Sie in den Eigenschaften der Meldeanzeige in der Gruppe "Eigenschaften > Security". Wenn ein angemeldeter Benutzer die erforderliche Berechtigung besitzt, kann er mithilfe von Bedienelementen in der Meldeanzeige Meldungen quittieren und editieren. Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, wird beim Drücken der Schaltflächen "Quittieren", "Editieren" oder durch Doppelklick auf eine Meldezeile der Anmeldedialog eingeblendet.

Spalten festlegen

Die Spalten, die in Runtime in der Meldeanzeige angezeigt werden, legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Sichtbare Spalten" die Spalten, die in Runtime dargestellt werden sollen.

Reihenfolge der Spalten

Wenn diese Eigenschaft aktiviert ist, dann kann in Runtime die Reihenfolge der Spalten in der Meldeanzeige verändert werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Eigenschaften Spalten > Spalten verschiebbar".

Sortierung

Wenn diese Eigenschaft aktiviert ist, dann können in Runtime die in der Meldeanzeige dargestellten Meldungen nach Datum und Uhrzeit sortiert werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Eigenschaften der Spalten > Sortieren nach Datum/Uhrzeit möglich".

Meldungen filtern

Mit dieser Eigenschaft legen Sie für die erweiterte Meldeanzeige fest, dass in Runtime nur Meldungen angezeigt werden, die im Meldetext eine projektierte Zeichenkette enthalten.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Filter".
2. Geben Sie in das Feld "Filterzeichenfolge" den gewünschten Begriff zur Filterung ein. Alternativ projektieren Sie über das Feld "Filtervariable" eine Filtervariable. Der Inhalt der Filtervariable dient in Runtime als Filterkriterium.

Meldeklassen kennzeichnen

In der ersten Spalte der Meldeanzeige wird ein Symbol angezeigt. Dieses Symbol ermöglicht die Zuordnung der Meldung zu einer Meldeklasse.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie "Sichtbare Spalten > Name der Meldeklasse".
3. Öffnen Sie den Editor "HMI-Meldungen" und klicken Sie auf die Registerkarte "Meldeklassen".
4. Legen Sie in der Spalte "Anzeigename" für eine Meldeklasse ein Symbol fest, mit dem die Meldungen dieser Meldeklasse gekennzeichnet werden sollen.

Hinweis

Das Objekt "Meldeanzeige" lässt sich nicht gruppieren.

Farben für Meldeklassen festlegen

Sie haben die Möglichkeit, Meldungen der ausgewählten Meldeklassen in Runtime in unterschiedlichen Farben darstellen zu lassen. Die Hintergrundfarben der Meldungen legen Sie im Editor "HMI-Meldungen" fest.

1. Öffnen Sie den Editor "HMI-Meldungen" und klicken Sie auf die Registerkarte "Meldeklassen".
2. Legen Sie unter "Eigenschaften > Eigenschaften > Farben" die Farben für Meldungen fest.
3. Abhängig vom Bediengerät ändern Sie bei Bedarf auch das Blinkverhalten.
4. Aktivieren Sie unter "Runtime-Einstellungen > Meldungen > Allgemein" die Option "Farben Meldeklassen", um Meldungen in der Meldeanzeige in den definierten Farben darzustellen.

Anzeigebereich festlegen

Sie wählen eine Variable, die den Zeitpunkt festlegt, ab dem Meldungen angezeigt werden. Folgende Datentypen sind zulässig:

- Externe Variablen: Date, Date_and_Time, Time_of_Day
- Interne Variablen: DateTime

Um den Anzeigebereich festzulegen, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Anzeige".
2. Legen Sie unter "Steuervariable für Anzeigebereich" die Variable fest, in der der Zeitpunkt festgelegt ist.

Zur PLC-Code-Anzeige wechseln

Sie projektieren die Systemfunktion "AktivierePLCCodeAnzeige" an eine Schaltfläche, um den Einsprung von einer Überwachungsmeldung in die Anzeige des betroffenen Programmcodes in der PLC-Code-Anzeige zu ermöglichen. In der Meldeanzeige legen Sie eine Steuervariable fest, die Sie für die Dynamisierung der Schaltfläche nutzen. Steuervariable dient dazu es auszuwerten, ob der Einsprung von der selektierten Überwachungsmeldung in die PLC-Code-Anzeige möglich ist. Um die Steuervariable festzulegen, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Anzeige".
2. Legen Sie unter "Eigenschaften > Anzeige > Steuervariable für PLC-Code-Anzeige" eine boolesche Variable fest, die den Einsprung von der letzten aktiven ProDiag-Meldung oder GRAPH-Meldung in die PLC-Code-Anzeige steuern wird.

Sie haben die Möglichkeit, die Schaltfläche mithilfe von Animationen zu dynamisieren, sodass z. B. beim Kommen einer Meldung die Schaltfläche ihre Farbe verändert.

Anzeige von Spaltenüberschriften

Die Darstellung der Meldeanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Registerkarte Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

1.9.2.47 Meldeanzeige, einfach

Verwendung

Meldungen werden am Bediengerät in der Meldeanzeige oder im Meldefenster dargestellt.

Das folgende Bild zeigt eine einfache Meldeanzeige:

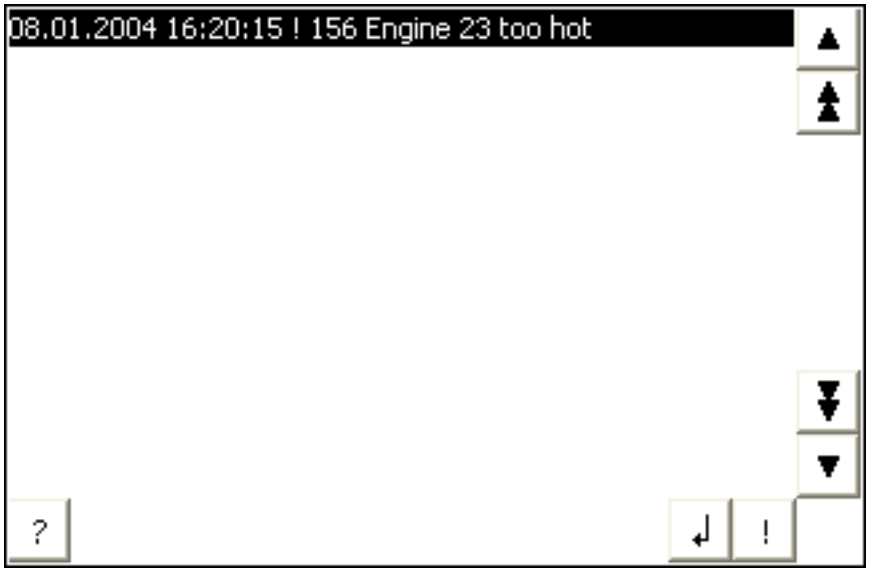

Hinweis

Das Objekt "Einfache Meldeanzeige" kann nicht mit einem Skript dynamisiert werden.

Hinweis

Das Objekt "Einfache Meldeanzeige" projektieren Sie auf den Geräten mit der Geräteversion 12.0.0.0 oder älter.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts.

Hinweis

Die Auswahl an Schriftarten ist abhängig davon, welche Schriftarten Sie in den Runtime-Einstellungen unter "Sprache & Schriftart" projektiert haben und welche Schriftarten Ihr Bediengerät unterstützt.

Insbesondere passen Sie die folgenden Eigenschaften an:

- Bedienelemente: Legt die Bedienelemente der Meldeanzeige fest.
- Meldeklassen: Legt fest, welche Meldeklassen in der Meldeanzeige angezeigt werden.
- Spalten: Legt die in Runtime darzustellenden Spalten fest.

Hinweis

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

Bedienelemente

Die Bedienelemente, mit denen die Meldeanzeige in Runtime bedient werden kann, werden im Inspektorfenster unter "Anzeige > Einstellungen" festgelegt. Die folgende Tabelle zeigt die Bedienelemente der Meldeanzeige und ihre Funktion:

Schaltfläche		Funktion
"Hilfetext"	?	Zeigt Hilfetext für eine Meldung an.
"Quittieren"	!	Quittiert eine Meldung.
"Loop-In-Alarm"	↲	Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1]

[1] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Meldeklassen auswählen

1. Klicken Sie im Inspektorfenster auf "Eigenschaften".
2. Aktivieren Sie unter "Meldeklassen" die Meldeklassen, die in der Meldeanzeige in Runtime dargestellt werden sollen.

Spalten festlegen

Die Spalten, die in Runtime in der Meldeanzeige angezeigt werden, legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Spalten".
2. Aktivieren Sie unter "Spalten" die Spalten, die in Runtime dargestellt werden sollen.

Anzeige von Spaltenüberschriften

Die Darstellung der Meldeanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Registerkarte Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Hinweis

Im Engineering System können Sie im Inspektorfenster in der Gruppe "Animationen" z. B. die Sichtbarkeit eines Objekts dynamisieren. Das Objekt "Einfache Meldeanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projektiert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, dann wird im Ausgabe-Fenster eine Fehlermeldung ausgegeben.

1.9.2.48 Meldeanzeige, erweitert

Meldeanzeige

Meldungen werden am Bediengerät in der Meldeanzeige oder im Meldefenster dargestellt. Das folgende Bild zeigt eine Meldeanzeige ohne Inhalt:

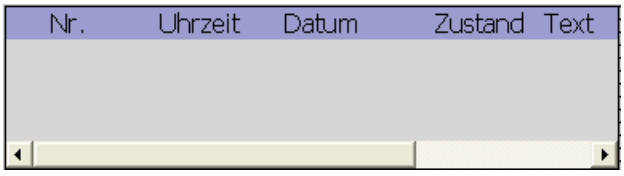

Meldezeile

Die Meldezeile ermöglicht in Runtime die Darstellung der aktuellsten anstehenden Meldung. Das folgende Bild zeigt eine Meldezeile:

So projektieren Sie die Meldezeile:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Darstellung".
2. Wählen Sie unter "Modus > Meldezeile".

Anzeigetyp ändern

Auf Bediengeräten mit einer Displaygröße größer als 6" können Sie sowohl die erweiterte Meldeanzeige als auch eine vereinfachte Meldeanzeige projektieren. Anzeigetyp steht zum Konfigurieren nur auf Geräten bis Geräteversion V13 zur Verfügung.

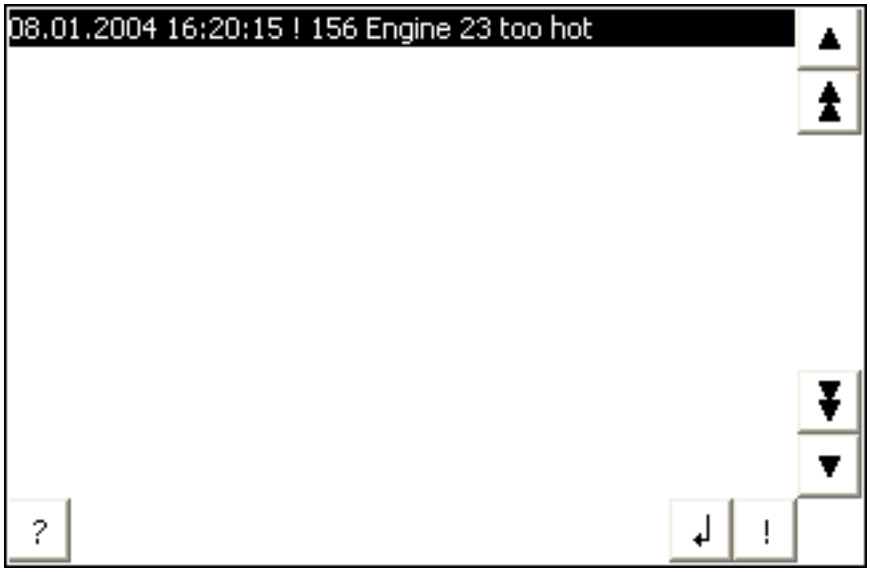

Um die vereinfachte Meldeanzeige zu projektieren, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Modus".
2. Wählen Sie "Vereinfacht".

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts.

Hinweis

Die Auswahl an Schriftarten ist abhängig davon, welche Schriftarten Sie in den Geräteeinstellungen unter "Sprache & Schriftart" projektiert haben und welche Schriftarten Ihr Bediengerät unterstützt.

Insbesondere passen Sie die folgenden Eigenschaften an:

- Bedienelemente: Legt die Bedienelemente der Meldeanzeige fest.
- Meldeklassen: Legt fest, welche Meldeklassen in der Meldeanzeige angezeigt werden.
- Spalten: Legt die in Runtime darzustellenden Spalten fest.
- Reihenfolge der Spalten: Legt fest, ob die Reihenfolge der Spalten in Runtime geändert werden kann.
- Kennzeichnung der Meldeklassen: Um verschiedene Meldeklassen unterscheiden zu können, kennzeichnen Sie diese in der ersten Spalte der Meldeanzeige.

- Filter: Legt fest, dass nur Meldungen angezeigt werden, die im Meldetext eine bestimmte Zeichenkette enthalten.
- Sortierung freigeben: Legt fest, ob in Runtime die Sortierung der Meldungen nach Datum und Uhrzeit möglich ist.
- Anzeigebereich festlegen: Legt fest, ab welchem Datum Meldungen in der Meldeanzeige angezeigt werden.

Hinweis

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

Bedienelemente

Die Bedienelemente, mit denen die Meldeanzeige in Runtime bedient werden kann, werden im Inspektorfenster unter "Eigenschaften > Eigenschaften > Anzeige > Einstellungen" festgelegt. Die folgende Tabelle zeigt die Bedienelemente der Meldeanzeige und ihre Funktion:

Schaltfläche		Funktion
"Tooltip"	?	Zeigt Tooltip für eine Meldung an.
"Quittieren"	!	Quittiert eine Meldung.
"Loop-In-Alarm"	↵	Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1]

[1] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Meldeklassen auswählen

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie unter "Meldeklassen" die Meldeklassen, die in der Meldeanzeige in Runtime dargestellt werden sollen.

Zugriffsschutz in Runtime

In den Eigenschaften der Meldeanzeige projektieren Sie in der Gruppe "Eigenschaften > Security" den Zugriffsschutz. Wenn ein angemeldeter Benutzer die erforderliche Berechtigung besitzt, kann er mit Hilfe der Bedienelemente in der Meldeanzeige Meldungen quittieren und editieren. Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, wird beim Drücken der Schaltflächen "Quittieren", "Editieren" oder durch Doppelklick auf eine Meldezeile der Anmeldedialog eingeblendet.

Spalten festlegen

Die Spalten, die in Runtime in der Meldeanzeige angezeigt werden, legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Spalten" die Spalten, die in Runtime dargestellt werden sollen.

Reihenfolge der Spalten

Wenn diese Eigenschaft aktiviert ist, dann kann in Runtime die Reihenfolge der Spalten in der Meldeanzeige verändert werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Eigenschaften Spalten > Reihenfolge der Spalten".

Sortierung freigeben

Wenn diese Eigenschaft aktiviert ist, dann können in Runtime die in der Meldeanzeige dargestellten Meldungen nach Datum und Uhrzeit sortiert werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie unter "Eigenschaften der Spalten > Sortieren nach Datum/Uhrzeit möglich".

Meldungen filtern

Mit dieser Eigenschaft legen Sie für die erweiterte Meldeanzeige fest, dass in Runtime nur Meldungen angezeigt werden, die im Meldetext eine projektierte Zeichenkette enthalten.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Filter".
2. Geben Sie in das Feld "Filterzeichenfolge" den gewünschten Begriff zur Filterung ein. Alternativ projektieren Sie über das Feld "Filtervariable" eine Filtervariable. Der Inhalt der Filtervariable dient in Runtime als Filterkriterium.

Meldeklassen kennzeichnen

In der ersten Spalte der Meldeanzeige wird ein Symbol angezeigt. Dieses Symbol ermöglicht die Zuordnung der Meldung zu einer Meldeklasse.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
2. Aktivieren Sie "Spalten > Meldeklasse".
3. Öffnen Sie den Editor "Meldungen" und klicken Sie auf die Registerkarte "Meldeklassen".
4. Legen Sie in der Spalte "Anzeigename" für eine Meldeklasse ein Symbol fest, mit dem die Meldungen dieser Meldeklasse gekennzeichnet werden sollen.

Hinweis

Das Objekt "Meldeanzeige" lässt sich nicht gruppieren.

Farben für Meldeklassen festlegen

Sie haben die Möglichkeit, Meldungen der ausgewählten Meldeklassen in Runtime in unterschiedlichen Farben darstellen zu lassen. Die Hintergrundfarben der Meldungen legen Sie im Editor "HMI-Meldungen" fest.

1. Öffnen Sie den Editor "HMI-Meldungen" und klicken Sie auf die Registerkarte "Meldeklassen".
2. Legen Sie unter "Eigenschaften > Eigenschaften > Farben" die Farben für Meldungen fest.
3. Aktivieren Sie unter "Runtime-Einstellungen > Meldungen > Allgemein" die Option "Farben Meldeklassen", um Meldungen in der Meldeanzeige in den definierten Farben darzustellen.

Anzeigebereich festlegen

Sie wählen eine Variable, die den Zeitpunkt festlegt, ab dem Meldungen angezeigt werden. Folgende Datentypen sind zulässig:

- Externe Variablen: Date, Date_and_Time, Time_of_Day
- Interne Variablen: DateTime

Um den Anzeigebereich festzulegen, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Anzeige".
2. Legen Sie unter "Steuervariable für Anzeigenbereich" die Variable fest, in der der Zeitpunkt festgelegt ist.

Verhalten bei Projektierung

Anzeige von Spaltenüberschriften

Die Darstellung der Meldeanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Registerkarte Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Darstellung der Schaltflächen zur Bedienung

Die Darstellung der Schaltflächen zur Bedienung der Meldeanzeige ist abhängig von der projektierten Größe. Prüfen Sie auf dem Bediengerät, ob alle benötigten Schaltflächen zur Verfügung stehen.

Siehe auch

Meldefenster (Seite 452)

Meldeindikator (Seite 455)

Meldeanzeige für aktuelle Meldungen projektieren (Seite 872)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.49 Meldefenster

Verwendung

Meldungen werden am Bediengerät in der Meldeanzeige oder im Meldefenster dargestellt. Aussehen und Bedienung des Meldefensters sind ähnlich der Meldeanzeige. Das Meldefenster hat die folgenden gleichen Ausprägungen wie die Meldeanzeige:

- Meldefenster
- Meldezeile: Die Meldezeile steht auf Basic Panels nicht zur Verfügung.

Das Meldefenster projektieren Sie im Editor "Globales Bild".

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.

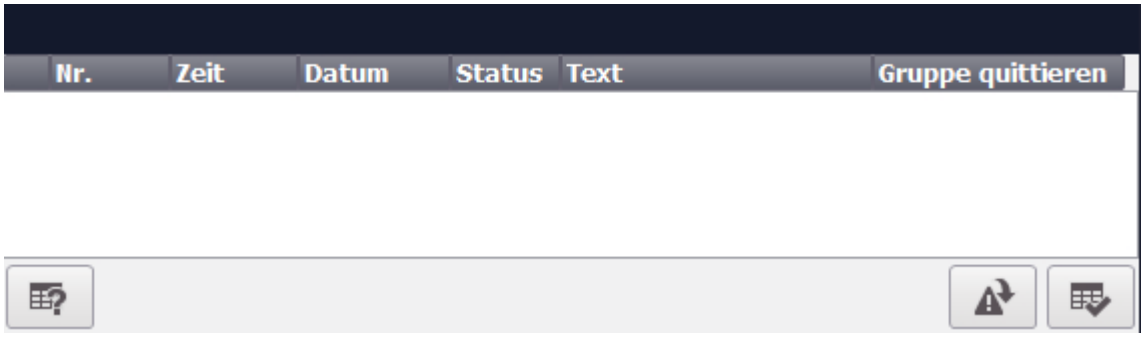

Hinweis

Im Engineering System dynamisieren Sie im Inspektorfenster "Eigenschaften > Animationen", z. B. die Sichtbarkeit eines Objekts. Das Objekt "Einfaches Meldefenster" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projektiert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, dann wird im Ausgabe-Fenster eine Fehlermeldung ausgegeben.

Einfaches Meldefenster

Auf Basic Panels mit der Geräteversion früher als V13 steht Ihnen zur Darstellung der Meldungen einfaches Meldefenster zur Verfügung.

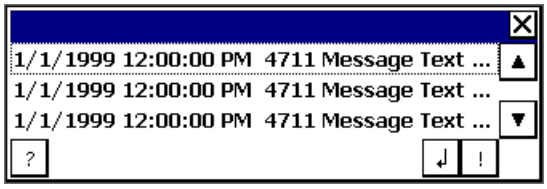

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Sie projektieren das Meldefenster wie die Meldeanzeige. Abweichend davon passen Sie die folgenden Eigenschaften an:

- Gebundene Meldefenster: Legt fest, dass das Meldefenster beim Bildwechsel den Fokus behält.
- Fenster: Sie legen Bedienung und Verhalten des Meldefensters in Runtime fest.

Hinweis

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

Bedienelemente

Die Bedienelemente, mit denen die Meldeanzeige in Runtime bedient werden kann, legen Sie im Inspektorfenster "Eigenschaften > Anzeige > Einstellungen" fest. Die folgende Tabelle zeigt die Bedienelemente des Meldefensters und ihre Funktion:

Schaltfläche		Funktion
"Tooltip"	?	Zeigt Tooltip für eine Meldung an.
"Quittieren"	!	Quittiert eine Meldung.
"Loop-In-Alarm"	↲	Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1]

[1] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Zugriffsschutz in Runtime

Im Inspektorfenster der Meldeanzeige projektieren Sie "Eigenschaften > Eigenschaften > Security" den Zugriffsschutz. Wenn ein angemeldeter Benutzer die erforderliche Berechtigung besitzt, kann er mit Hilfe der Bedienelemente in der Meldeanzeige Meldungen quittieren und editieren. Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, wird beim Betätigen der Schaltflächen "Quittieren", "Editieren" oder durch Doppelklick auf eine Meldezeile der Anmeldedialog eingeblendet.

Hinweis

Basic Panels

Der Zugriffsschutz steht für Basic Panels nicht zur Verfügung.

Fokus für das Meldefenster aktivieren

Damit auch beim Bildwechsel das Meldefenster nicht den Fokus verliert, wählen Sie folgende Option:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Modus".
2. Aktivieren Sie "Beschriftung".

Fenster

Das Verhalten des Meldefensters wird im Inspektorfenster "Eigenschaften > Eigenschaften > Modus > Fenster" festgelegt. Die folgende Tabelle zeigt die möglichen Eigenschaften:

Option	Funktion
Automatisch anzeigen	Meldefenster wird automatisch eingeblendet, z. B. beim Auftreten einer Systemmeldung.
Schließbar	Nach Ablauf einer bestimmten Zeit wird das Fenster wieder geschlossen. Die Anzeigedauer legen Sie in den Meldeeinstellungen fest.
Gebunden	Meldefenster ist an eine Bestätigung gebunden, z. B. Meldung muss quittiert werden. Wenn das gebundene Meldefenster den Fokus besitzt, dann lassen sich die Schaltflächen im dahinterliegenden Bild nicht bedienen. Die an eine Funktionstaste projektierten Funktionen werden ausgeführt.
Größe veränderbar	In Runtime kann die Größe des Meldefensters geändert werden.

Siehe auch

Meldeanzeige, erweitert (Seite 447)

Meldeindikator (Seite 455)

Meldefenster projektieren (Seite 879)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.50 Meldeindikator

Verwendung

Der Meldeindikator ist ein grafisches Symbol, das je nach Projektierung auf anstehende oder zu quittierende Meldungen hinweist. Den Meldeindikator projektieren Sie im Editor "Globales Bild". Das folgende Bild zeigt einen Meldeindikator:

Meldeklassen

Welche Meldeklassen mit Meldeindikator eingeblendet werden, legen Sie im Inspektorfenster "Allgemein > Meldeklassen" fest. Meldeklassen sind z. B. "Warnings" oder "Errors".

Bedienung in Runtime festlegen

1. Markieren Sie den Meldeindikator im Bild.
2. Klicken Sie im Inspektorfenster "Ereignisse > Klicken" oder "Klicken bei Blinken".
3. Die "Funktionsliste" wird geöffnet. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen und Skripte wird geöffnet.
4. Wählen Sie unter "Meldungen" die Systemfunktion "ZeigeMeldefenster".
5. Wählen Sie bei "Objektname" den Namen des Meldefensters aus der Auswahlliste. Legen Sie unter "Darstellung" fest, ob das Meldefenster eingeblendet, ausgeblendet oder zwischen beiden Zuständen hin und hergeschaltet wird.

Siehe auch

Meldeanzeige, erweitert (Seite 447)

Meldefenster (Seite 452)

Meldeindikator projektieren (Seite 880)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.51 Optionsschaltflächen

Verwendung

Sie verwenden das Objekt "Optionsschaltfläche" zur Darstellung und Auswahl verschiedener Optionen, von denen der Bediener nur eine auswählen kann. Damit der Bediener den voreingestellten Wert nur bei Bedarf ändert, aktivieren Sie eine der Optionen standardmäßig. Um eine Optionsschaltfläche in den Prozess einzubinden, dynamisieren Sie die entsprechenden Attribute.

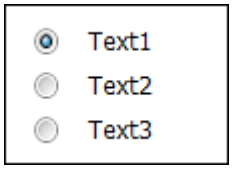

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl der Felder
- Auswahl der Felder: Legt fest, welche Felder aktiviert dargestellt werden.

Anzahl der Felder festlegen

Sie legen im Inspektorfenster die Anzahl der Felder fest:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Legen Sie im Bereich "Beschriftung > Anzahl Felder" die gewünschte Anzahl an Feldern fest.

Der Wert der Eigenschaft "Anzahl Felder" gibt gleichzeitig den oberen Grenzwert für die Eigenschaft "Index" an. Die Änderung des Werts kann folgende Auswirkungen haben:

- Erhöhung der Anzahl der Felder
  Neue Felder werden unter dem Feld mit dem höchsten Wert der Eigenschaft "Index" eingefügt. Die Standardbeschriftung des neuen Feldes ändern Sie mit "Schrift > Text".
- Verringerung der Anzahl der Felder
  Alle Felder, bei denen der Wert der Eigenschaft "Index" höher ist als die neue Anzahl, werden entfernt.

Voreinstellung der Optionsschaltflächen festlegen

Mit der Eigenschaft "Default" legen Sie fest, welche der Felder der Optionsschaltflächen aktiviert dargestellt wird. Aktivieren Sie unter "Allgemein" in der Tabellenzeile des Feldes "Default", wenn das Feld voreingestellt aktiviert sein soll. Nur ein Feld darf gleichzeitig aktiviert sein.

Siehe auch

Kontrollkästchen (Seite 415)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.52 PDF-Anzeige

Verwendung

Mit dem Objekt "PDF-Anzeige" projektieren Sie eine PDF-Anzeige, auf der der Bediener in Runtime Dokumente im PDF-Format aufrufen und anzeigen lassen kann. Die Dokumente im PDF-Format befinden sich dabei im internen oder externen Speicher des jeweiligen Bediengeräts.

Die PDF-Anzeige unterstützt die PDF-Versionen von 1.0 bis 1.7. Die Suche wird für die PDF-Versionen 1.4 bis 1.7 unterstützt.

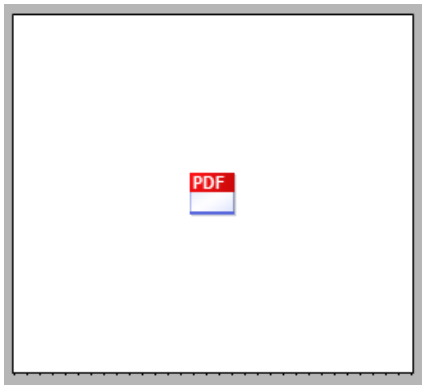

Das Projektieren der PDF-Anzeige ist im Folgenden möglich:

- In einem Bild
- In einer Vorlage
- In einem Slide-in-Bild
- In einem Pop-up-Bild

Hinweis

Das Objekt "PDF-Anzeige" darf in einem Projekt mehrfach projektiert sein. Jedoch darf maximal eine PDF-Anzeige zur selben Zeit aktiv sein.

Hinweis

Geräteabhängigkeit des Objekts "PDF-Anzeige"

Das Objekt "PDF-Anzeige" ist nur für KTP Mobile Panels, Comfort Panels und RT Advanced verfügbar.

Hinweis

Pro Bild darf nur ein Objekt "PDF-Anzeige" eingefügt werden.

Darstellung

Im Inspektorfenster machen Sie Angaben zu der Position und Größe der PDF-Anzeige. Insbesondere passen Sie die folgenden Eigenschaften an:

- Dateiname: Legt die PDF-Datei fest, die standardmäßig in der PDF-Anzeige geöffnet wird.
- Dateiname Variable: Legt die Variable fest, die eine beliebige PDF-Datei öffnet.
- Symbolleiste: Legt die Schaltflächen der Symbolleiste fest. Wenn alle Schaltflächen deaktiviert sind, wird die Symbolleiste in Runtime nicht angezeigt.

Prinzipielles Vorgehen

1. Legen Sie im Editor "Variable" eine interne Variable vom Datentyp "String" an, z. B. "PDFFile".
2. Fügen Sie im Editor "Bilder" die PDF-Anzeige in das Bild ein. Die PDF-Anzeige finden Sie in der Task Card "Werkzeuge > Controls".
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Geben Sie bei "Dateiname" den Dateinamen der PDF-Datei ein, die in Runtime standardmäßig geöffnet wird.
   Alternativ können Sie das Feld "Dateiname" leer lassen.
5. Wählen Sie bei "Dateiname Variable" die Variable "PDFFile" aus der Auswahlliste.
6. Fügen Sie im Editor "Bilder" ein E/A-Feld in das Bild ein.
7. Wählen Sie im Inspektorfenster des E/A-Felds "Eigenschaften > Eigenschaften > Allgemein > Variable" die Variable "PDFFile" aus der Auswahlliste.
8. Weisen Sie im Inspektorfenster unter "Dateiname Variable" der PDF-Anzeige die Variable "PDFFile" zu.
9. Legen Sie zu dem projektierten E/A-Feld ein Textfeld für die Beschriftung an, z. B. "PDF-Pfad".

Wenn der Bediener im E/A-Feld den Dateinamen zusammen mit dem Pfad eingibt, öffnet die PDF-Anzeige die betreffende Datei.

Symbolleiste (RT Advanced)

In Runtime Advanced haben Sie die Möglichkeit, für die PDF-Anzeige eine Symbolleiste zu projektieren, mit welcher die PDF-Anzeige in Runtime bedient wird.

Die Bedienelemente, die in der PDF-Anzeige nur in Runtime sichtbar sind, projektieren Sie im Engineering System im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste":

- Durchsuchen: Legt fest, ob die Suchleiste und Schaltflächen "Vorwärts" und "Rückwärts" eingeblendet werden.
  Zum Durchsuchen des angezeigten Dokuments geben Sie den Suchbegriff in das Suchfeld ein. Der gesuchte Text darf maximal 255 Zeichen enthalten. Der gefundene Text wird in der PDF-Anzeige visuell hervorgehoben. Mithilfe der Schaltflächen "Vorwärts" und "Rückwärts" navigieren Sie im Text zwischen den Suchergebnissen. Wenn der gefundene Text sich außerhalb des sichtbaren Bereichs befindet, scrollt die PDF-Anzeige automatisch zum Suchergebnis.
  Falls der gesuchte Begriff im Dokument nicht gefunden werden kann, wird das Suchfeld gelb hervorgehoben und die Navigationsschaltflächen werden ausgegraut. Der gesuchte Text darf maximal 255 Zeichen enthalten. Zeilenumbrüche und andere Formatierungszeichen werden bei der Suche nicht berücksichtigt.

  Hinweis

  Die Suche nach einem Text, der Sonderzeichen oder Umlaute enthält, wird in PDF-Dokumenten der Version 1.3 oder älter nicht unterstützt.

- Kopieren: Legt fest, dass die Schaltflächen "Markieren"/"Verschieben" und "Kopieren" eingeblendet werden.
  Um den gewünschten Textabschnitt zu markieren, schalten Sie mithilfe des Symbols "Markieren" den Auswahlmodus in der PDF-Anzeige an. Klicken Sie vor dem ersten Zeichen und ziehen Sie den Cursor über den Text, um den gewünschten Textabschnitt zu markieren. Der gewünschte Abschnitt wird in blau hervorgehoben. Nachdem Sie den Text markiert haben, klicken Sie auf "Kopieren", um den ausgewählten Text zu kopieren. Den Text, der in die Zwischenablage gespeichert wird, können Sie in ein anderes Objekt einfügen.
  Auch wenn der ausgewählte Abschnitt Bilder oder andere Objekte enthält, wird aus dem markierten Bereich nur der Text in die Zwischenablage kopiert.
  In die Zwischenablage wird nur der Text kopiert, der in der PDF-Anzeige aktuell angezeigt wird. Stellen Sie beim Markieren sicher, dass der gewünschte Textabschnitt vollständig sichtbar ist.

- Zoom: Legt fest, dass die Schaltflächen "Vergrößern", "Verkleinern", "An Seitenbreite anpassen" und "An Seitenhöhe anpassen" eingeblendet werden.
  Mithilfe der Schaltflächen der Zoom-Leiste passen Sie die Ansicht in der PDF-Anzeige an. Standardmäßig wird das Dokument mit der Einstellung "An Seitenbreite anpassen" geöffnet. Wenn Sie die Anzeige an die Seitenhöhe anpassen, wird die gesamte Seite in der PDF-Anzeige dargestellt.

Bedienelement	Bezeichnung	Funktion
	"Durchsuchen"	Sucht nach einer Zeichenfolge innerhalb des Dokuments. Mithilfe der Pfeile navigieren Sie vor und zurück zwischen den Suchergebnissen.
	"Markieren"	Markiert den Text.

Bedienelement	Bezeichnung	Funktion
	"Kopieren"	Kopiert den ausgewählten Text.
	"Verschieben"	Verschiebt die Ansicht horizontal und vertikal.
	"An Seitenbreite anpassen"	Passt die Anzeige an die Seitenbreite an.
	"An Seitenhöhe anpassen"	Passt die Anzeige an die Seitenhöhe an.
	"Vergrößern"	Vergrößert die Anzeige.
	"Verkleinern"	Verkleinert die Anzeige.

Bedienelemente projektieren

Bedienelemente sind projektierte Schaltflächen im Prozessbild oder Softkeys am Bediengerät zum Bedienen des Objekts.

Projektieren Sie selbst Bedienelemente, z. B. Schaltflächen, und verknüpfen Sie die Bedienelemente mit Systemfunktionen.

1. Ziehen Sie aus dem Werkzeugfenster z. B. das Objekt "Schaltfläche" in das Bild.
2. Geben Sie einen beliebig langen Text ein.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
   Der Dialog "Funktionsliste" wird geöffnet.
4. Die Systemfunktionen zum Bedienen der PDF-Anzeige finden Sie in der Liste der Systemfunktionen in der Gruppe "Tastaturbedienung für Bildobjekte".
   Wählen Sie aus der Liste eine Funktion, z. B. "PDFZoomIn".
5. Wählen Sie aus der Liste "Bildobjekt" den Objektnamen der PDF-Anzeige, in der der Befehl ausgeführt wird.

Systemfunktionen

Für das Objekt "PDF-Anzeige" sind folgende Systemfunktionen verfügbar:

Systemfunktion	Beschreibung
"PDFHeranzoomen"	Vergrößert die Darstellung um eine Zoomstufe
"PDFHerauszoomen"	Verkleinert die Darstellung um eine Zoomstufe
"PDFGeheZuNächster-Seite"	Wechselt zur nächsten Seite
"PDFGeheZuVorheri-gerSeite"	Wechselt zur vorherigen Seite

Systemfunktion	Beschreibung
"PDFGeheZuErsterSeite"	Wechselt zur ersten Seite
"PDFGeheZuLetzterSeite"	Wechselt zur letzten Seite
"PDFAnpassenAufBreite"	Passt die Darstellung an die Breite des Fensters der PDF-Anzeige an
"PDFAnpassenAufHöhe"	Passt die Darstellung an die Höhe des Fensters der PDF-Anzeige an
"PDFZoomOriginal"	Ändert die Darstellung auf die Originalgröße
"PDFAufwärtsScrollen"	Blättert aufwärts
"PDFAbwärtsScrollen"	Blättert abwärts
"PDFLinksScrollen"	Blättert nach links
"PDFRechtsScrollen"	Blättert nach rechts
"PDFGeheZuSeite"	Wechselt zur eingegebenen Seite

Hinweis

Das Blättern funktioniert auf allen Bediengeräten auch mit Hilfe von Multitouch-Bedienung.

Das folgende Bild zeigt eine PDF-Anzeige mit den projektierten Bedienelementen:

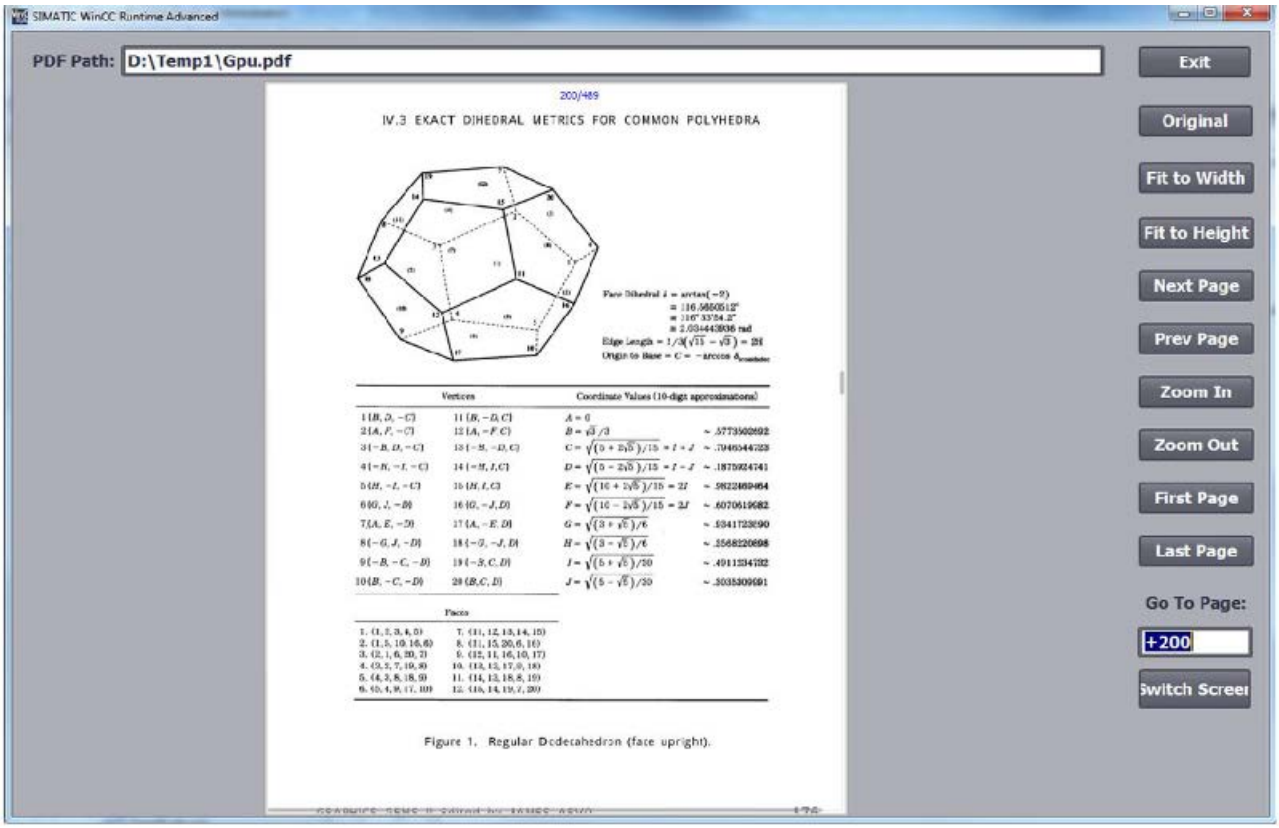

Bemerkungen

Gegenüber dem Adobe Acrobat hat die PDF-Anzeige einen eingeschränkten Funktionsumfang:

- Die PDF-Anzeige unterstützt keine Verlinkungen.
- Die PDF-Anzeige unterstützt folgende Zoomstufen:
  - die Zoomstufen 25 % bis 200 % für KTP Mobile Panels, Comfort Panels
  - die Zoomstufen 25% bis 400% für Runtime Advanced

Hinweis

In Runtime Advanced können Sie mit Hilfe von Multitouch-Gesten die Anzeige stufenlos skalieren. Sie vergrößern und verkleinern die Anzeige per Zwei-Finger-Zoom.

Hinweis

Das Objekt "PDF-Anzeige" kann nicht mithilfe von VB-Scripting adressiert werden.

Hinweis

Bei der Anzeige und beim Blättern von großen PDF-Dateien kann es zu hohem Arbeitsspeicherverbrauch kommen. In einem solchen Fall wird die Systemmeldung 130000 "Nicht genügend Arbeitsspeicher vorhanden“ angezeigt.

Siehe auch

Objekte für WinCC Runtime Advanced (Seite 343)

1.9.2.53 PLC-Code-Anzeige

Verwendung

Das Objekt "PLC-Code-Anzeige" dient zur Anzeige des aktuellen Programmstatus der Anwenderprogramme und Sicherheitsprogramme (F-Bausteine), die in den Programmiersprachen KOP, FUP oder GRAPH programmiert worden sind. Fehler beim Ablauf einer GRAPH-Schrittkette werden direkt am entsprechenden Schritt angezeigt.

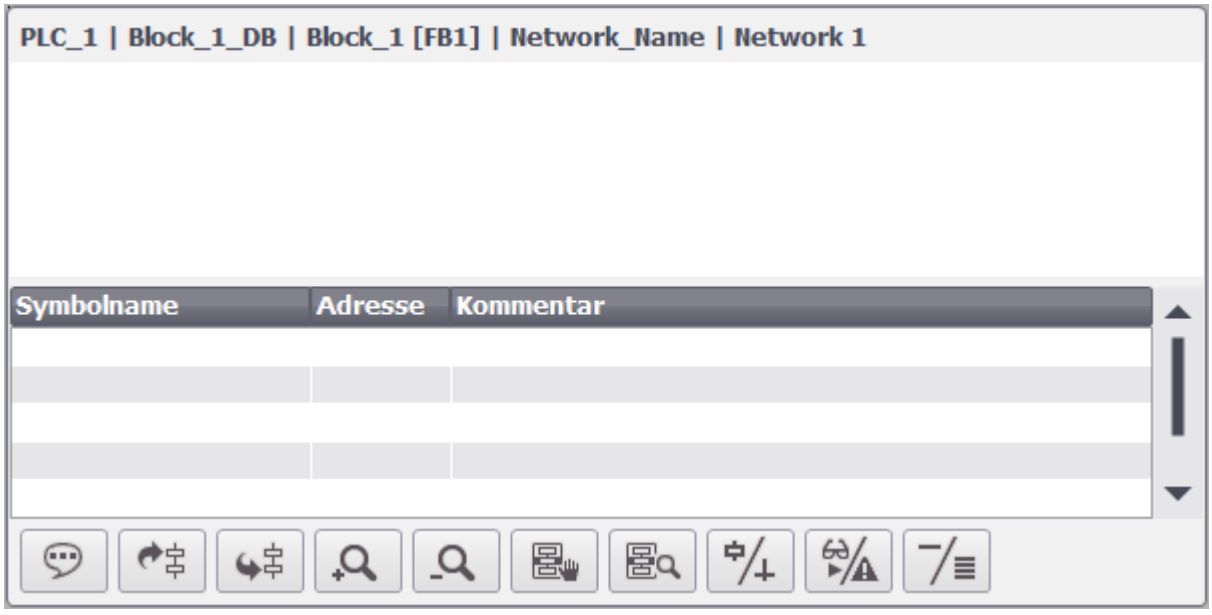

Die PLC-Code-Anzeige ist in folgende Bereiche unterteilt:

- Überschrift
  Überschrift zeigt den PLC-Namen, FB-Namen sowie die Nummer und den Namen des Netzwerks in Runtime.
- Informationsbereich
  Im Informationsbereich wird das PLC-Netzwerk in Runtime grafisch dargestellt.
- Symboltabelle
  Die Symboltabelle zeigt die symbolischen Namen der Operanden, absolute Adressen der Operanden und entsprechende Kommentare.
- Symbolleiste
  In der Symbolleiste werden die Schaltflächen angezeigt, die in der PLC-Code-Anzeige in Runtime verfügbar sind.

Hinweis

Das Objekt "PLC-Code-Anzeige" kann nicht in Bildbausteinen, im Globalen Bild, in Pop-up-Bildern und Slide-in-Bildern verwendet werden.

Hinweis

Geräteabhängigkeit des Objekts "PLC-Code-Anzeige"

Das Objekt "PLC-Code-Anzeige" ist für RT Advanced, Comfort Panels und KTP Mobile Panels ab Displaygröße 7" verfügbar.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zur Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die Schaltflächen der Symbolleiste an.

- Die Anzeige der Pfadinformationen in der Überschrift der PLC-Code-Anzeige legen Sie unter "Anzeige > Überschrift (Pfad) anzeigen" fest.
- Die Anzahl der Zeilen, die in Runtime in der Symboltabelle dargestellt werden, legen Sie unter "Symboltabelle Linien > Anzahl sichtbare Zeilen" fest.
- Die Breite der Detailansicht im Verhältnis zum Informationsbereich legen Sie unter "GRAPH Detailansicht > GRAPH Detailansicht Breite (%)" fest.
- Die zweizeilige Darstellung von GRAPH-Schrittnamen in Runtime legen Sie unter "GRAPH Anzeige > Zweizeilige Anzeige" fest.
  Um die zweizeilige Anzeige zu ermöglichen, muss die Option "Zwei Zeilen anzeigen" unter "Extras > Einstellungen > PLC-Programmierung > GRAPH > Länge von Schritt- und Transitionsnamen" aktiviert sein.

Symbolleiste

Die Bedienelemente, mit denen die PLC-Code-Anzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste > Schaltflächen".

Bedienelement	Bezeichnung	Funktion
	"Symbolbereich"	Zeigt den Symbolbereich an.
	"Vorheriges Netzwerk"	Navigiert zum vorherigen Netzwerk.
	"Nächstes Netzwerk"	Navigiert zum nächsten Netzwerk.
	"Vergrößern"	Vergrößert den Informationsbereich.
	"Verkleinern"	Verkleinert den Informationsbereich.
	"Detail"	Zeigt die Detailansicht an.
	"Schrittmodus"	Wechselt zwischen der manuellen Schrittauswahl und automatischer Auswahl des aktiven Schrittes.

Bedienelement	Bezeichnung	Funktion
	"Transition oder Interlock"	Wechselt zwischen den Darstellungen von Transitions- und Interlocknetzwerken.
	"Aktualwerte oder Erstwerte"	Wechselt zwischen den Ansichten von Aktualwerten und Erstwerten.
	"Einzelnes Netzwerk oder gesamter Funktionsbaustein"	Wechselt zwischen der Ansicht von einem einzelnen Netzwerk mit der Inputlogik und der Ansicht des gesamten Funktionsbausteins.

PLC-Code-Anzeige projektieren

1. Ziehen Sie per Drag&Drop die PLC-Code-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste".
3. Aktivieren Sie die Schaltflächen, die Sie in Runtime benötigen, z. B.: Nächstes Netzwerk, vorheriges Netzwerk, Schrittmodus.
4. Konfigurieren Sie bei Bedarf die Darstellung der Schaltflächen, z. B. Hintergrundfarbe oder Füllmuster.
5. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
6. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen.
7. Passen Sie bei Bedarf die Überschriften oder die Reihenfolge der Spalten an.
8. Wählen Sie "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

PLC-Code-Anzeige mit der Tastatur bedienen

Um die PLC-Code-Anzeige mit der Tastatur zu bedienen, gehen Sie folgendermaßen vor:

1. Drücken Sie so oft die Taste <Tab>, bis sich der Fokus auf der gewünschten Schaltfläche befindet.
   Mit der Tastenkombination <Umschalttaste + Tab> navigieren Sie rückwärts zur vorherigen Schaltfläche.
2. Drücken Sie die Taste <Enter>.
   Der Befehl wird ausgeführt.

Bedienung über Touchscreen

Scrollen durch Ziehen in der Symboltabelle der PLC-Code-Anzeige wird nicht unterstützt. Um in der Symboltabelle horizontal und vertikal zu scrollen, nutzen Sie die Bildlaufleisten und die Bildlaufpfeile.

Mehrsprachige GRAPH-Namen

WinCC unterstützt die Anzeige von mehrsprachigen Schrittnamen und Transitionsnamen für die GRAPH-Bausteine ab der Version 5.0. Beim Sprachenwechsel in Runtime werden die Schrittnamen und Transitionsnamen in der ausgewählten Runtime-Sprache angezeigt. Falls die ausgewählte Sprache im GRAPH-Baustein nicht verfügbar ist, werden die Namen in der Standardsprache Englisch angezeigt.

Mnemonik in der PLC-Code-Anzeige

Die PLC-Code-Anzeige zeigt die Daten in der Mnemonik, die Sie im TIA Portal eingestellt haben. Voreingestellt ist die internationale Mnemonik. Mit der internationalen Mnemonik enthalten z. B. die angezeigten Adressen in allen projektierten Sprachen in Runtime die Kennung "IO", unabhängig von konfigurierten Runtime-Sprachen.

Eigenschaften einer Steuerung in Runtime

Wenn in WinCC eine Steuerung mit der PLC-Code-Anzeige verbunden ist und Runtime aktiv ist, führt die Änderung des Namens der Steuerung zu Fehlern in Runtime. Ändern Sie daher nicht den Namen der Steuerung, die IP-Adresse oder andere Eigenschaften der HMI-Verbindung.

Adressierung von Variablen im Objekt „PLC-Code-Anzeige“

Das Objekt "PLC-Code-Anzeige" unterstützt nur symbolische Adressierung von Variablen. Wenn der Operand nicht symbolisch adressiert wird, kann das Netzwerk mit dem Operanden nicht angezeigt werden und es wird eine Fehlermeldung ausgegeben.

Darstellung von Zeitwert-Konstanten

Bei der Darstellung von Zeitwert-Konstanten wird in der PLC-Code-Anzeige in Runtime die Zeit im Format "Stunden/Minuten/Sekunden/" angezeigt, unabhängig von der im Engineering System projektierten Eingabe. Zum Beispiel, eine im Engineering System festgelegte Konstante "150 s" erscheint in Runtime als "2 min 30 s".

Siehe auch

Maschinen und Anlagen mit ProDiag überwachen (Seite 1933)

Unterstützte Datentypen (Seite 1965)

Unterstützte Anweisungen (Seite 1960)

PLC-Code-Anzeige projektieren (Seite 1954)

PLC-Code-Anzeige projektieren (Seite 1954)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.54 PLC-Code-Anzeige

Verwendung

Das Objekt "PLC-Code-Anzeige" ist für das Bediengerät WinCC Runtime Professional verfügbar. Dieses Objekt dient zur Anzeige des aktuellen Programmstatus von PLC-Programmen, die in den Programmiersprachen KOP, FUP oder GRAPH programmiert worden sind. Fehler beim Ablauf einer GRAPH Schrittkette werden direkt am entsprechenden Schritt angezeigt.

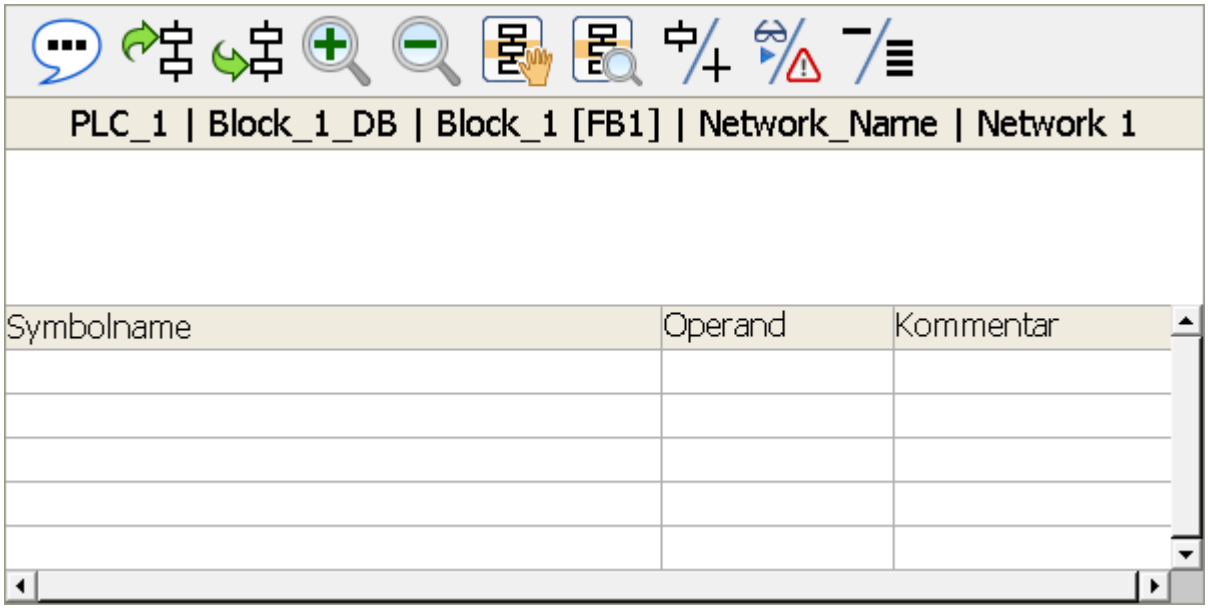

Die PLC-Code-Anzeige ist in folgende Bereiche unterteilt:

- Symbolleiste
  In der Symbolleiste werden die Schaltflächen angezeigt, die in der PLC-Code-Anzeige in Runtime verfügbar sind.
- Überschrift
  Überschrift zeigt den PLC-Namen, FB-Namen sowie die Nummer und den Namen des Netzwerks in Runtime.
- Netzwerkbereich
  Im Netzwerkbereich wird das PLC-Netzwerk in Runtime grafisch dargestellt.
- Symboltabelle
  Die Symboltabelle zeigt die symbolischen Namen der Operanden, absolute Adressen der Operanden und entsprechende Kommentare.

Hinweis

Das Objekt "PLC-Code-Anzeige" kann nicht in Gruppen oder Bildbausteinen verwendet werden.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Schaltflächen der Symbolleiste unter "Symbolleiste > Symbolleiste - Schaltflächen"
- Anzahl der Zeilen, die in Runtime in der Symboltabelle dargestellt werden unter "Symboltabelle Linien > Anzahl sichtbare Zeilen"
- Zweizeilige Darstellung von GRAPH-Schrittnamen in Runtime unter "GRAPH Anzeige > Zweizeilige Anzeige"
  Um die zweizeilige Anzeige zu ermöglichen, muss die Option "Zwei Zeilen anzeigen" unter "Extras > Einstellungen > PLC-Programmierung > GRAPH > Länge von Schritt- und Transitionsnamen" aktiviert sein.

Netzwerk anzeigen

Das in der PLC-Code-Anzeige darzustellende Netzwerk wird über benutzerdefinierte Funktionen festgelegt. Abhängig von der Programmiersprache des PLC-Programms wird innerhalb der benutzerdefinierten Funktion eine der folgenden API-Funktionen aufgerufen:

- OpenViewerIECPLByCall
- OpenViewerIECPLByFCCall
- OpenViewerIECPLByAssignment
- OpenViewerS7GraphByBlock

Detaillierte Informationen und Beispiele zum Erstellen einer benutzerdefinierten Funktion finden Sie in der Beschreibung der API-Funktionen.

Symbolleiste

Die Bedienelemente, mit denen die PLC-Code-Anzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste > Schaltflächen".

Bedienelement	Bezeichnung	Funktion
	"Symbolbereich"	Zeigt den Symbolbereich an.
	"Vorheriges Netzwerk"	Navigiert zum vorherigen Netzwerk.
	"Nächstes Netzwerk"	Navigiert zum nächsten Netzwerk.
	"Vergrößern"	Vergrößert den Informationsbereich.
	"Verkleinern"	Verkleinert den Informationsbereich.

Bedienelement	Bezeichnung	Funktion
	"Detail"	Zeigt die Detailansicht an.
	"Schrittmodus"	Wechselt zwischen der manuellen Schrittauswahl und automatischer Auswahl des aktiven Schrittes.
	"Transtition oder Interlock"	Wechselt zwischen Transition-Ansicht und Interlock-Ansicht.
	"Aktualwerte oder Erstwerte"	Wechselt zwischen den Ansichten von Aktualwerten und Erstwerten.
	"Einzelnes Netzwerk oder gesamter Funktionsbaustein"	Wechselt zwischen der Ansicht von einem einzelnen Netzwerk mit der Inputlogik und der Ansicht des gesamten Funktionsbausteins.

Mehrsprachige GRAPH-Namen

WinCC unterstützt die Anzeige von mehrsprachigen Schrittnamen und Transitionsnamen für die GRAPH-Bausteine ab der Version 5.0. Beim Sprachenwechsel in Runtime werden die Schrittnamen und Transitionsnamen in der ausgewählten Runtime-Sprache angezeigt. Falls die ausgewählte Sprache im GRAPH-Baustein nicht verfügbar ist, werden die Namen in der Standardsprache Englisch angezeigt.

Eigenschaften einer Steuerung in Runtime

Wenn in WinCC eine Steuerung mit der PLC-Code-Anzeige verbunden ist und Runtime aktiv ist, führt die Änderung des Namens der Steuerung zu Fehlern in Runtime. Ändern Sie daher nicht den Namen der Steuerung, die IP-Adresse oder andere Eigenschaften der HMI-Verbindung.

Fehlermeldung beim Laden der PLC mit einer neueren Version des TIA Portals

Wenn PLC-Code-Anzeige aus einer älteren Version von WinCC auf ein Programm der Steuerung zugreift, die mit einer neueren Version geladen ist, wird eine Fehlermeldung ausgegeben. Achten Sie darauf, dass sowohl die Steuerung als auch HMI mit der gleichen Version des TIA Portals geladen sind.

Adressierung von Variablen im Objekt „PLC-Code-Anzeige“

Das Objekt "PLC-Code-Anzeige" unterstützt nur symbolische Adressierung von Variablen. Wenn das Operand nicht symbolisch adressiert wird, kann das Netzwerk mit dem Operanden nicht angezeigt werden und es wird eine Fehlermeldung ausgegeben.

Siehe auch

PLC-Code-Anzeige projektieren (Seite 1954)

PLC-Code-Anzeige projektieren (Seite 1954)

Maschinen und Anlagen mit ProDiag überwachen (Seite 1933)

Unterstützte Anweisungen (Seite 1974)

Unterstützte Datentypen (Seite 1965)

1.9.2.55 Polygon

Verwendung

Das Objekt "Polygon" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.

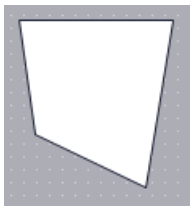

Darstellung

Sie ändern im Inspektionsfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

Geometrie: Ändert, löscht oder fügt Eckpunkte hinzu.

Geometrie

Die Eckpunkte werden in der Reihenfolge ihrer Erstellung nummeriert. Sie können die Eckpunkte verändern, löschen oder weitere Eckpunkte hinzufügen:

1. Klicken Sie im Inspektionsfenster auf "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie im Bereich "Geometrie" den gewünschten Eckpunkt aus. Geben Sie bei "Position X" und "Position Y" einen Wert ein.
3. Um einen Eckpunkt hinzuzufügen, klicken Sie auf die Schaltfläche .
4. Um einen Eckpunkt zu löschen, klicken Sie auf die Schaltfläche .

Alternatives Vorgehen

Die Eckpunkte können Sie auch mit der Maus ändern, löschen oder hinzufügen:

1. Markieren Sie das Objekt. Positionieren Sie den Mauszeiger auf dem gewünschten Eckpunkt. Der Mauszeiger verwandelt sich in ein Fadenkreuz.
2. Ziehen Sie den Eckpunkt bei gedrückter Maustaste auf die gewünschte Position.
3. Klicken Sie mit der rechten Maustaste auf die Stelle, an der Sie den Eckpunkt einfügen wollen. Wählen Sie im Kontextmenü den Befehl "Punkt hinzufügen".

Rotation in Runtime konfigurieren

Sie konfigurieren das Objekt "Polygon" so, dass es in Runtime um einen Referenzpunkt rotiert. Die Rotation in Runtime ist nur für Geräte mit Runtime Professional möglich.

Sie tragen die Werte für den Rotationswinkel in der Einheit Grad ein.

1. Klicken Sie im Inspektionsfenster auf "Eigenschaften > Eigenschaften > Darstellung".
2. Tragen Sie im Bereich "Drehung" die gewünschten Werte für folgende Attribute ein:
   - X
   - Y
   - Winkel

Siehe auch

Polygonzug (Seite 471)

Objekt in Runtime drehen (Seite 130)

1.9.2.56 Polygonzug

Verwendung

Das Objekt "Polygonzug" ist ein offenes Objekt. Wenn Sie das Objekt mit Farbe füllen wollen, dann verwenden Sie das Objekt "Polygon".

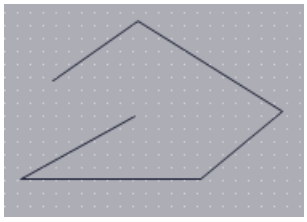

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Linienanfang und Linienende: Legt die Art des Linienanfangs und des Linienendes fest.
- Geometrie: Ändert, löscht oder fügt Eckpunkte hinzu.

Linienanfang und Linienende

Sie legen im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" den Anfangspunkt und den Endpunkt der Linie fest. Als Anfangs- bzw. Endpunkte verwenden Sie z. B. Pfeilspitzen. Welche Anfangs- und Endpunkte Ihnen zur Verfügung stehen, ist geräteabhängig.

Geometrie

Die Eckpunkte werden in der Reihenfolge ihrer Erstellung nummeriert. Sie können die Eckpunkte verändern, löschen oder weitere Eckpunkte hinzufügen:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie im Bereich "Geometrie" den gewünschten Eckpunkt aus. Geben Sie bei "Position X" und "Position Y" einen Wert ein.
3. Um einen Eckpunkt hinzuzufügen, klicken Sie auf die Schaltfläche .
4. Um einen Eckpunkt zu löschen, klicken Sie auf die Schaltfläche .

Alternatives Vorgehen

Die Eckpunkte können Sie auch mit der Maus ändern, löschen oder hinzufügen:

1. Markieren Sie das Objekt. Positionieren Sie den Mauszeiger auf dem gewünschten Eckpunkt. Der Mauszeiger verwandelt sich in ein Fadenkreuz.
2. Ziehen Sie den Eckpunkt bei gedrückter Maustaste auf die gewünschte Position.

Rotation in Runtime konfigurieren

Sie konfigurieren das Objekt "Polygonzug" so, dass es in Runtime um einen Referenzpunkt rotiert. Die Rotation in Runtime ist nur für Geräte mit Runtime Professional möglich.

Sie tragen die Werte für den Rotationswinkel in der Einheit Grad ein:

1. Klicken Sie im Inspektorfenster auf "Darstellung".
2. Tragen Sie im Bereich "Drehung" die gewünschten Werte für folgende Attribute ein:
   - X
   - Y
   - Winkel

Siehe auch

Polygon (Seite 470)

1.9.2.57 ProDiag-Übersicht

Verwendung

Das Objekt "ProDiag-Übersicht" bietet Ihnen einen Überblick über den aktuellen Zustand der projektierten Überwachungen in Runtime. Beim Auftreten eines Fehlers werden in der ProDiag-Übersicht die Fehlerart und die Fehlerkategorie ermittelt. Sie können direkt zur Fehlermeldung in einer Meldeanzeige navigieren, sowie den betroffenen Programmcode in der PLC-Code-Anzeige darstellen lassen.

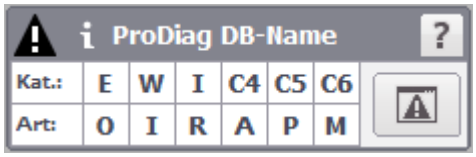

Hinweis

Das Objekt "ProDiag-Übersicht" kann nicht in Bildbausteinen und im Globalen Bild verwendet werden.

Hinweis

Geräteabhängigkeit des Objekts "ProDiag-Übersicht"

Das Objekt "ProDiag-Übersicht" ist für RT Advanced, RT Professional, Comfort Panels und KTP Mobile Panels verfügbar.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Angezeigte Schaltflächen
- Bezeichnungen und Farben für die Kategorien
- Bezeichnungen und Farben für die Überwachungsarten

ProDiag-Übersicht projektieren

Eine CPU und ein ProDiag-FB sind angelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie unter der Eigenschaft "Variable" die Auswahlschaltfläche.
3. Selektieren Sie die Zustandsvariable des ProDiag-FB.
   Alternativ ziehen Sie die Zustandsvariable aus der Detailansicht per Drag&Drop an die gewünschte Verwendungsstelle.
4. Legen Sie unter "Ausgabe > Kategorien" die Bezeichnungen, die Farben und das Blinkverhalten für die Kategorien fest.

5. Legen Sie unter "Ausgabe > Überwachungsarten" die Bezeichnungen, die Farben und das Blinkverhalten für die Überwachungsarten fest.
6. Wählen Sie unter "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

Kompakte Ansicht projektieren

Sie können die ProDiag-Übersicht in einer kompakten Ansicht projektieren.

Sie deaktivieren die Anzeige von Kategorien und Überwachungsarten unter "Eigenschaften > Gestaltung > Optionen". Anzeige von Schaltflächen deaktivieren Sie unter "Eigenschaften > Symbolleiste > Allgemein".

Überwachungsarten und Kategorien

Im Objekt "ProDiag-Übersicht" können Sie maximal acht Kategorien und sechs Überwachungsarten anzeigen. Folgende vordefinierte Kategorien und Überwachungstypen stehen Ihnen zur Verfügung:

Bezeichnung	Kategorien
E	Fehler
W	Warnung
I	Info

Benennen Sie die Kategorien C4 bis C8, die standardmäßig angelegt werden, nach Ihren Anforderungen um.

Bezeichnung	Überwachungsarten
O	Operandenfehler
I	Interlockfehler
R	Reaktionsfehler
A	Aktionsfehler
P	Positionsfehler
M	Meldung

Anzeigesymbole der Überwachungsarten und Kategorien können Sie jederzeit im Inspektorfenster unter "Ausgabe" ändern.

Symbole

Die Symbole, die in der ProDiag-Übersicht angezeigt werden, legen Sie im Inspektorfenster unter "Eigenschaften > Gestaltung > Optionen" fest. Folgende Symbole stehen für ProDiag-Übersicht zur Verfügung:

Symbol	Name	Funktion
?	Hilfe	Öffnet den benutzerdefinierten Tooltip.
i	Info	Zeigt die Information zur Überwachung an.
!	Fehler	Zeigt an, dass ein Fehler aufgetreten ist.

Schaltfläche "Meldeanzeige"

Um die Schaltfläche "Meldeanzeige" in der ProDiag-Übersicht anzuzeigen, aktivieren Sie die Eigenschaft "Schaltfläche "Meldeanzeige" anzeigen" unter "Eigenschaften > Symbolleiste > Allgemein".

Schaltfläche	Name	Funktion
	Meldeanzeige	Öffnet die konfigurierte Meldeanzeige mit der Fehlermeldung.

Deaktivierte Anzeige

Falls es in Runtime zu einem Fehler in der Verbindung mit der Steuerung kommt, wird das Objekt "ProDiag-Übersicht" ausgegraut dargestellt. Diese deaktivierte Anzeige kann folgende Gründe haben:

- Die Steuerung ist deaktiviert
- Der projektierte ProDiag-Programmbaustein wurde aus dem Steuerungsprogramm entfernt
- Die Steuerung ist im Stoppmodus

Sobald die Fehlerursache behoben wurde und die Verbindung wieder aufgebaut wird, zeigt die ProDiag-Übersicht den aktuellen Online-Status der Überwachungen in Runtime.

Siehe auch

Grundlagen der Überwachung mit ProDiag (Seite 1933)

Grundlagen der Überwachung von Operanden (Seite 1933)

ProDiag-Übersicht projektieren (Seite 1943)

Maschinen und Anlagen mit ProDiag überwachen (Seite 1933)

1.9.2.58 Rechteck

Verwendung

Das Objekt "Rechteck" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius Ecken: Legt den horizontalen bzw. vertikalen Abstand zwischen der Ecke des objektumfassenden Rechtecks und dem Startpunkt einer Eckrundung fest.

Radius Ecken

Die Ecken des Objekts "Rechteck" können beliebig abgerundet werden. Wenn die Eigenschaften "X" und "Y" auf den Wert 100 % gesetzt werden, wird das Rechteck als Ellipse dargestellt. Sobald eine der beiden Eigenschaften den Wert 0 % hat, wird ein normales Rechteck ohne Eckrundung dargestellt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radius Ecken" einen Wert für "X" ein.
   Der Werteingabe ist der prozentuale Anteil an der halben Breite des Rechtecks.
3. Geben Sie im Bereich "Radius Ecken" einen Wert für "Y" ein.
   Der Werteingabe ist der prozentuale Anteil an der halben Höhe des Rechtecks.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.59 Rezepturanzeige

Verwendung

Das Objekt "Rezepturanzeige" dient der Anzeige und Änderung von Rezepturen.

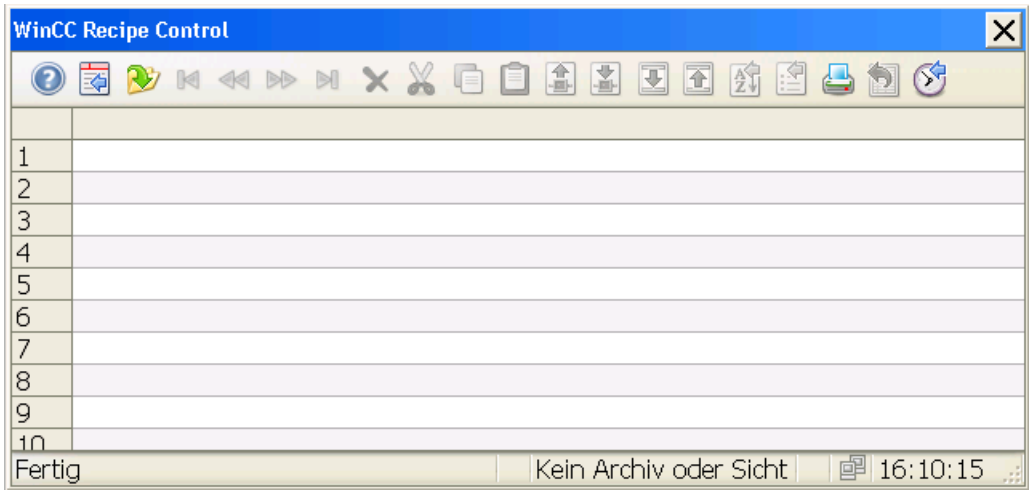

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Archiv/Anzeige: Rezeptur oder Rezepturabfrage festlegen
- Symbolleiste: Legt die Bedienelemente der Rezepturanzeige fest.
- Spalten: Legt die Spalten fest, die im Rezepturdatensatz angezeigt werden.

Archiv/Anzeige

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Archiv/Anzeige" fest, welche Rezeptur oder Rezepturabfrage in der Rezepturanzeige dargestellt wird.

Symbolleiste

Die Bedienelemente, mit denen die Rezepturanzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Schaltflächen". In der einfachen Rezepturanzeige werden die Bedienelemente auf die Funktionen des Menüs abgebildet.

Bedienelement		Beschreibung
	"Tooltip"	Ruft den Hilfetext zur Rezepturanzeige auf.
	"Konfigurationsdialog"	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften der Rezepturanzeige verändern.

Bedienelement		Beschreibung
	"Datenverbindung wählen"	Öffnet einen Dialog zur Auswahl einer Rezeptur oder Rezepturabfrage. Die Rezepturdaten werden dann in der Tabelle der Rezepturanzeige dargestellt.
	"Erste Zeile"	Über die Schaltfläche wird in der Tabelle der erste Datensatz dargestellt.
	"Zeile vorher"	Über die Schaltfläche wird in der Tabelle der vorangehende Datensatz dargestellt.
	"Nächste Zeile"	Über die Schaltfläche wird in der Tabelle der nachfolgende Datensatz dargestellt.
	"Letzte Zeile"	Über die Schaltfläche wird in der Tabelle der letzte Datensatz dargestellt.
	"Zeile löschen"	Der Inhalt der markierten Zeilen wird gelöscht.
	"Zeilen ausschneiden"	Der Inhalt der markierten Zeilen wird ausgeschnitten.
	"Zeilen kopieren"	Der Inhalt der markierten Zeilen wird kopiert.
	"Zeilen einfügen"	Der Inhalt der kopierten bzw. ausgeschnittenen Zeilen wird ab der markierten Zeile eingefügt.
	"Variablen lesen"	Über die Schaltfläche werden die Inhalte der verbundenen WinCC-Variablen gelesen und in die Rezepturelemente geschrieben. Um die Schaltfläche nutzen zu können, muss bei der angezeigten Rezeptur die Kommunikationsart "Variablen" aktiviert sein. Die Rezepturelemente müssen mit Variablen verbunden sein.
	"Variablen schreiben"	Über die Schaltfläche werden die Inhalte der Rezepturelemente in die verbundenen WinCC-Variablen geschrieben. Um die Schaltfläche nutzen zu können, muss bei der angezeigten Rezeptur die Kommunikationsart "Variablen" aktiviert sein. Die Rezepturelemente müssen mit Variablen verbunden sein.
	"Archiv importieren"	Über die Schaltfläche wird eine CSV-Datei aus dem Verzeichnis "ua" des Projektordners in die Tabelle der Rezepturanzeige importiert.
	"Archiv exportieren"	Der beim Laden ursprüngliche Inhalt der Tabelle wird mit Tabellenüberschriften über die Schaltfläche exportiert. Die Rezeptur wird als CSV-Datei im Verzeichnis "ua" des Projektordners abgelegt.
	"Sortier-Dialog"	Öffnet einen Dialog zur Einstellung einer benutzerdefinierten Sortierung der angezeigten Spalten.
	"Drucken"	Startet den Ausdruck der angezeigten Werte. Der beim Druck verwendete Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest.

Bedienelement		Beschreibung
	"Zeitbasis einstellen"	Öffnet einen Dialog zur Einstellung der Zeitbasis für die verwendeten Zeiten.
	"Daten exportieren"	Über die Schaltfläche exportieren Sie die Rezeptur oder Rezepturabfrage wie projektiert, jedoch mit den aktuellen Runtime-Daten in eine CSV-Datei. Wenn die Option "Dialog anzeigen" aktiviert ist, öffnet sich ein Dialog, in dem Sie die Einstellungen zum Export sehen und den Export starten können. Mit den entsprechenden Bedienberechtigungen dürfen Sie auch die Datei und das Verzeichnis für den Export auswählen. Wenn kein Dialog angezeigt wird, startet sofort der Export der Daten in die voreingestellte Datei.
	"Benutzerdefiniert"	Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert.

Spalten

Im Inspektorfenster unter "Eigenschaften > Eigenschaften > Spalten > Spalteneigenschaften" legen Sie fest, welche Spalten in der Rezepturanzeige dargestellt werden.

Verhalten bei Projektierung

Anzeige der Spaltenüberschriften

Die Darstellung der Rezepturanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Siehe auch

Anzeige von Rezepturen (Seite 1232)

Eigenschaften der Rezepturanzeige (Seite 1233)

Rezepturdaten bearbeiten (Seite 1262)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.60 Rezepturanzeige (ab V13)

Verwendung

Das Objekt "Rezepturanzeige" dient zur Anzeige von Rezepturen am Bediengerät.

Hinweis

Geräteabhängigkeit des Objekts "erweiterte Rezepturanzeige"

Das Objekt "erweiterte Rezepturanzeige" ist auf den Bediengeräten Basic Panels 2nd Generation, Comfort Panels, Mobile Panels mit der Geräteversion V13 verfügbar.

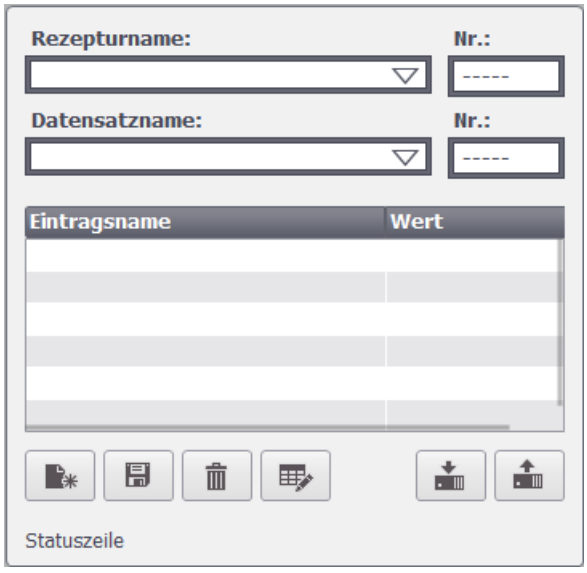

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Symbolleiste: Legt die Bedienelemente der Rezepturanzeige fest.
- Nummer anzeigen: Legt fest, ob die Nummer der Rezeptur und die Nummer des Rezepturdatensatzes angezeigt werden.
- Beschriftung: Legt die Beschriftungen für den Namen der Rezeptur und des Rezepturdatensatzes fest.

Bedienelemente

Die Bedienelemente, mit denen die Rezepturanzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Schaltflächen". In der einfachen Rezepturanzeige werden die Bedienelemente auf die Funktionen des Menüs abgebildet.

Bedienelement		Beschreibung
	"Infotext"	Ruft den projektierten Tooltip zur ausgewählten Rezeptur auf.
	"Datensatz hinzufügen"	Legt einen neuen Rezepturdatensatz in der Rezeptur an.
	"Datensatz löschen"	Löscht den markierten Datensatz.
	"Umbenennen"	Ändert den Namen des markierten Datensatzes.
	"Speichern"	Speichert den geänderten Datensatz unter dem aktuellen Namen.
	"Speichern unter..."	Speichert den geänderten Datensatz unter einem neuen Namen.
	"In PLC schreiben"	Überträgt den aktuellen Wert in die Steuerung.
	"Von PLC lesen"	Liest den aktuellen Wert aus der Steuerung.
	"Rezepturvariablen abgleichen"	Vergleicht die Werte des markierten Datensatzes mit den Werten in der Steuerung.

Nummer anzeigen

In Runtime wird die Nummer der Rezeptur und die Nummer des Rezepturdatensatzes angezeigt. Die Nummer der Rezeptur identifiziert die Rezeptur innerhalb des Projekts eindeutig.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Anzeige".
2. Aktivieren Sie "Anzeigen > Nummern anzeigen".

Beschriftung anzeigen

Mit dieser Eigenschaft legen Sie die Namen fest, die für Rezepturen und Datensätze in der Rezepturanzeige dargestellt werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Beschriftung".
2. Aktivieren Sie "Beschriftungen anzeigen".
3. Geben Sie die Beschriftungen für die Rezeptur und den Rezepturdatensatz ein.

Verhalten bei der Projektierung

Anzeige der Spaltenüberschriften

Die Darstellung der Rezepturanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Scrollen in der Rezepturanzeige

Eine Rezepturanzeige enthält ein Element, das in der Rezepturtabelle als Textliste dargestellt wird. Sie können in der Tabelle der Rezepturanzeige erst dann zwischen den Elementen scrollen, wenn Sie vorher ein Element ausgewählt haben, das keine Textliste enthält.

1.9.2.61 Rezepturanzeige, einfach

Verwendung

Das Objekt "Rezepturanzeige" dient der Anzeige und Änderung von Rezepturen.

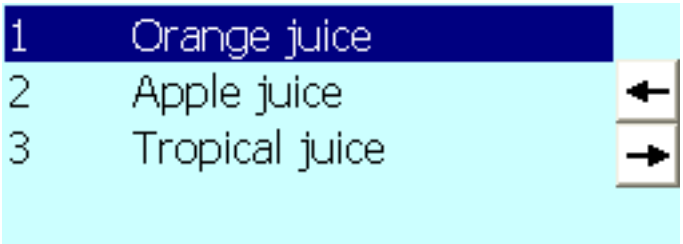

Hinweis

Das Objekt "Einfache Rezepturanzeige" projektieren Sie auf den Geräten mit der Geräteversion 12.0.0.0 oder älter.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Symbolleiste: Legt die Menüeinträge der Rezepturanzeige fest.

Symbolleiste

Die Menüeinträge, mit denen die Rezepturanzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste".

Menüeintrag	Beschreibung
"Tooltip"	Ruft den projektierten Tooltip zur ausgewählten Rezeptur auf.
"Datensatz hinzufügen"	Legt einen neuen Rezepturdatensatz in der Rezeptur an.
"Datensatz löschen"	Löscht den markierten Datensatz.
"Speichern"	Speichert den geänderten Datensatz unter dem aktuellen Namen.
"Speichern unter"	Speichert den geänderten Datensatz unter einem neuen Namen.
"In PLC schreiben"	Überträgt den aktuellen Wert in die Steuerung.
"Von PLC lesen"	Liest den aktuellen Wert aus der Steuerung.
"Umbenennen"	Legt fest, dass die Schaltfläche "Umbennen" angezeigt wird.
"Vorwärts"	Legt fest, dass die Schaltflächen des Menüs sichtbar sind.
"Zurück"	Legt fest, dass die Schaltfläche "Zurück" angezeigt wird.

1.9.2.62 Rezepturanzeige, erweitert

Erweiterte Rezepturanzeige

Das Objekt "Rezepturanzeige" dient der Anzeige und Änderung von Rezepturen. Je nach Bediengerät steht Ihnen die erweiterte bzw. die einfache Benutzeranzeige zur Verfügung

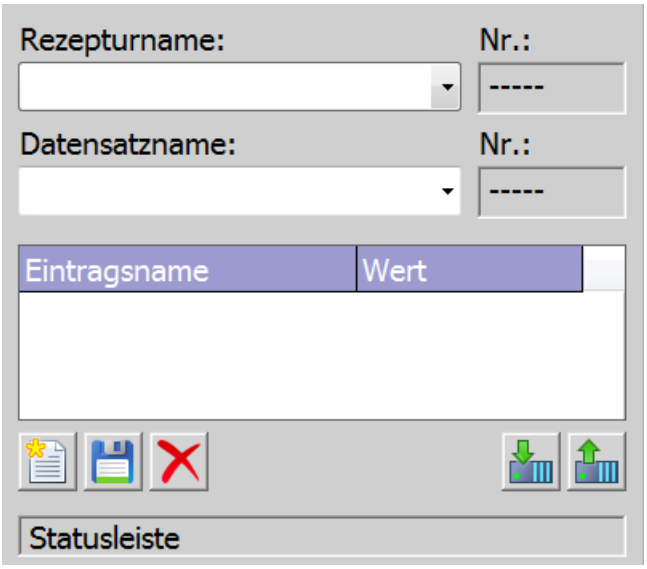

Anzeigetyp ändern

Auf Bediengeräten mit einer Displaygröße größer als 6" können Sie zur Anzeige und Bearbeitung von Rezepturen sowohl die erweiterte Rezepturanzeige als auch die vereinfachte Rezepturanzeige verwenden. Anzeigetyp steht zum Konfigurieren nur auf Geräten bis Geräteversion V13 zur Verfügung.

Hinweis

Verwenden Sie die einfache Rezepturanzeige nicht in einer Gruppe.

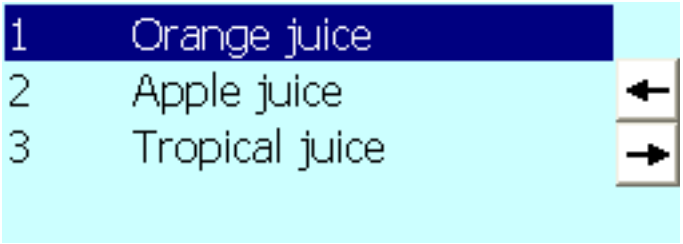

So projektieren Sie die einfache Rezepturanzeige:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Modus > Anzeigetyp > Vereinfacht".

Hinweis

Verhalten für Runtime Professional, Runtime Advanced und Panels

Im Engineering System können Sie im Inspektorfenster im Register "Animationen", z. B. die Sichtbarkeit eines Objekts dynamisieren. Das Objekt "Einfache Rezepturanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projektiert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, wird im Ausgabefenster eine Fehlermeldung ausgegeben.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Bedienelemente: Legt die Bedienelemente der Rezepturanzeige fest.
- Nummer anzeigen: Legt fest, ob die Nummer der Rezeptur und die Nummer des Rezepturdatensatzes angezeigt werden.

Bedienelemente

Die Bedienelemente, mit denen die Rezepturanzeige in Runtime bedient wird, projektieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Schaltflächen". In der einfachen Rezepturanzeige werden die Bedienelemente auf die Funktionen des Menüs abgebildet.

Bedienelement		Beschreibung
	"Tooltip"	Ruft den projektierten Tooltip zur ausgewählten Rezeptur auf.
	"Datensatz hinzufügen"	Legt einen neuen Rezepturdatensatz in der Rezeptur an.
	"Datensatz löschen"	Löscht den markierten Datensatz.
	"Speichern"	Speichert den geänderten Datensatz unter dem aktuellen Namen.
	"Speichern unter"	Speichert den geänderten Datensatz unter einem neuen Namen.
	"Rezepturvariablen abgleichen"	Vergleicht die Werte des markierten Datensatzes mit den Werten in der Steuerung.
	"In PLC schreiben"	Überträgt den aktuellen Wert in die Steuerung.
	"Von PLC lesen"	Liest den aktuellen Wert aus der Steuerung.

Nummer anzeigen

In Runtime wird die Nummer der Rezeptur und die Nummer des Rezepturdatensatzes angezeigt. Die Nummer der Rezeptur identifiziert die Rezeptur innerhalb des Projekts eindeutig.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Anzeige".
2. Aktivieren Sie "Anzeigen > Nummern anzeigen".

Verhalten bei Projektierung

Anzeige der Spaltenüberschriften

Die Darstellung der Rezepturanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

Siehe auch

Rezepturanzeige, einfach (Seite 482)

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.63 Rundschaltfläche

Verwendung

Mit dem Objekt "Rundschaltfläche" bedienen Sie Prozesse. Eine Rundschaltfläche verwenden Sie zum Beispiel, um Meldungen zu bestätigen. Um eine Rundschaltfläche in den Prozess einzubinden, dynamisieren Sie die entsprechenden Attribute.

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

Modus: Legt fest, ob die Rundschaltfläche gedrückt oder als Wechselschalter dargestellt wird.

Typ: Die Rundschaltfläche kann z. B. mit Text und Grafik dargestellt werden.

Rundschaltfläche gedrückt darstellen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Modus > Gedrückt".
3. Um die Rundschaltfläche als Wechselschalter darzustellen, aktivieren Sie "Modus > Umschalten".

Typ der Rundschaltfläche festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie als "Typ" z. B. "Grafik und Text".

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.64 Rohr

Verwendung

Mit dem Objekt "Rohr" stellen Sie ein gegebenenfalls mehrfach geknicktes durchgehendes Rohr nach. Um eine Rohranlage nachzubilden, verbinden Sie das Objekt "Rohr" mit weiteren Objekten z. B. Rohrbogen oder T-Stück.

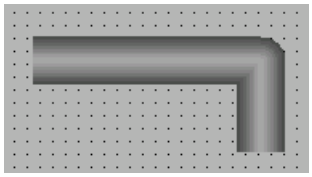

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Geometrie: Sie fügen zusätzliche Punkte hinzu und erweitern das Objekt.

Ein Rohr kann beliebig viele Eckpunkte haben. Die Eckpunkte sind in der Reihenfolge ihrer Erstellung nummeriert. Sie können sie einzeln verschieben.

Siehe auch

Rohrbogen (Seite 487)

T-Stück (Seite 520)

Doppel-T-Stück (Seite 364)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.65 Rohrbogen

Verwendung

Mit dem Objekt "Rohrbogen" stellen Sie ein gebogenes Rohrstück dar. Um eine Rohranlage nachzubilden, verbinden Sie das Objekt "Rohrbogen" mit weiteren Objekten z. B. Rohr oder Doppel-T-Stück.

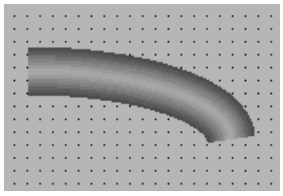

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Winkel: Sie bestimmen den Anfangs- und Endwinkel des Rohrbogens.
- Radius: Sie bestimmen den horizontalen und vertikalen Radius des Rohrbogens.

Die Attribute sind abhängig voneinander.

- Wenn Sie "Anfang Winkel" und "Ende Winkel" ändern, ändern sich Breite und Höhe automatisch mit.
  "Radius horizontal" und "Radius vertikal" bleiben erhalten.
- Wenn Sie "Radius horizontal" und "Radius vertikal" ändern, ändern sich Breite und Höhe automatisch mit.
  "Anfang Winkel" und "Ende Winkel" bleiben erhalten.
- Wenn Sie Breite und Höhe ändern, ändern sich "Radius horizontal" und "Radius vertikal" automatisch mit.
  "Anfang Winkel" und "Ende Winkel" erhalten.

Siehe auch

Rohr (Seite 487)

T-Stück (Seite 520)

Doppel-T-Stück (Seite 364)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.66 Schalter

Verwendung

Mit dem Objekt "Schalter" projektieren Sie einen Schalter, mit dem Sie in Runtime zwischen zwei vordefinierten Zuständen umschalten. Den aktuellen Zustand des Objekts "Schalter" visualisieren Sie durch eine Beschriftung oder durch eine Grafik.

Das folgende Bild zeigt einen Schalter vom Typ "Schalter".

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

- Typ: Legt die grafische Darstellung des Objekts fest.

Typ

Die Darstellung des Schalters wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" festgelegt.

Typ	Beschreibung
"Schalter"	Die beiden Zustände des Schalters werden in Form eines Schalters dargestellt. Die Stellung des Schalters zeigt den aktuellen Zustand an. Durch Verschieben des Schalters wird in Runtime umgeschaltet. Bei diesem Typ legen Sie unter "Schaltrichtung" die Bewegungsrichtung des Schalters fest.
"Schalter mit Text"	Der Schalter wird als Schaltfläche dargestellt. Der aktuelle Zustand wird durch eine Beschriftung visualisiert. In Runtime wird durch Klicken auf die Schaltfläche umgeschaltet.
"Schalter mit Grafik"	Der Schalter wird als Schaltfläche dargestellt. Der aktuelle Zustand wird durch eine Grafik visualisiert. In Runtime wird durch Klicken auf die Schaltfläche umgeschaltet.

Hinweis

Basic Panels

Der Typ "Schalter" steht bei Basic Panels nicht zur Verfügung.

Siehe auch

Übersicht über die Objekte (Seite 95)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.67 Schaltfläche

Verwendung

Mit dem Objekt "Schaltfläche" projektieren Sie ein Objekt, mit dem der Bediener in Runtime eine beliebig projektierbare Funktion ausführt.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt die grafische Darstellung des Objekts fest.
- Text / Grafik: Legt fest, ob die grafische Darstellung statisch oder dynamisch ist.
- Hotkey festlegen: Legt eine Taste oder Tastenkombination fest, mit der der Bediener die Schaltfläche bedient.

Hinweis

Die Festlegung eines Hotkey ist nur bei Bediengeräten mit Tasten möglich.

Modus

Die Darstellung der Schaltfläche wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein >Modus" festgelegt.

Modus	Beschreibung
"Unsichtbar"	Die Schaltfläche ist in Runtime nicht sichtbar.
"Text"	Die Schaltfläche wird mit Text angezeigt. Der Text dient dazu, die Funktion der Schaltfläche ablesbar zu machen.
"Grafik"	Die Schaltfläche wird mit einer Grafik angezeigt. Die Grafik dient dazu, die Funktion der Schaltfläche darzustellen.
"Grafik oder Text"	Die Schaltfläche wird mit Text oder Grafik angezeigt. Wenn die Grafik nicht angezeigt werden kann, wird der entsprechende Text angezeigt.
"Grafik und Text"	Die Schaltfläche wird mit Text und Grafik angezeigt.

Abhängig vom Gerät stehen Ihnen unterschiedliche Optionen zur Verfügung.

Text / Grafik

Abhängig von der Eigenschaft "Modus" können Sie die Anzeige statisch oder dynamisch festlegen. Die Anzeige wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Text" oder "Grafik" festgelegt.

Vom Typ "Grafik" oder "Text" stehen Ihnen z. B. folgende Optionen zur Auswahl.

Typ	Option	Beschreibung
"Grafik"	"Grafik"	Bei "Grafik wenn Schaltfläche "nicht gedrückt" legen Sie eine Grafik fest, die in der Schaltfläche für den Zustand "AUS" angezeigt wird. Wenn Sie "Grafik wenn Schaltfläche "gedrückt" aktivieren, können Sie eine Grafik für den Zustand "EIN" eingeben.
	"Grafikliste"	Die Grafik in der Schaltfläche ist abhängig vom Zustand. Je nach Zustand wird der entsprechende Eintrag aus der Grafikliste angezeigt.

Typ	Option	Beschreibung
"Text"	"Text"	Bei "Text wenn Schaltfläche "nicht gedrückt" legen Sie den Text fest, der in der Schaltfläche für den Zustand "AUS" angezeigt wird. Wenn Sie "Text wenn Schaltfläche "gedrückt" aktivieren, können Sie einen Text für den Zustand "EIN" eingeben.
	"Textliste"	Der Text in der Schaltfläche ist abhängig vom Zustand. Je nach Zustand wird der entsprechende Eintrag aus der Textliste angezeigt.

Hinweis

Die Verwendung von Textlisten und Grafiklisten in Schaltflächen steht auf WinCC Runtime Professional nicht zur Verfügung.

Hotkey festlegen

Im Inspektorfenster wird eine Taste oder Tastenkombination festgelegt, mit der der Bediener die Schaltfläche in Runtime bedienen kann.

1. Klicken Sie im Inspektorfenster " Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Bereich "Hotkey" in der Auswahlliste eine Taste oder eine Tastenkombination.

Siehe auch

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 247)

Beispiel: Schaltfläche mit Zugriffsschutz projektieren (Seite 1424)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.68 Schieberegler

Verwendung

Mit dem Objekt "Schieberegler" überwachen Sie Prozesswerte innerhalb eines definierten Bereichs und passen sie an. Der überwachte Bereich wird in Form eines Schiebereglers visualisiert. Durch Verschieben des Reglers greifen Sie in den Prozess ein und korrigieren den angezeigten Prozesswert.

Hinweis

Projektieren Sie bei einem regulierten Projekt keine Systemfunktion an das Ereignis "Ändern", wenn damit eine GMP relevante Aktion ausgeführt wird.

Projektieren Sie stattdessen die Systemfunktionen an das Ereignis "Wertänderung" der Variablen, um Anzahl der Benutzeraktionen zu reduzieren.

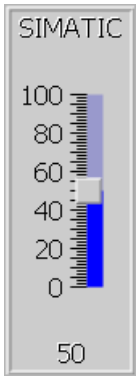

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften nach Bedarf an:

- Maximaler Wert und Minimaler Wert: Legt den oberen und den unteren Endwert der Skala fest.
- Aktuellen Wert anzeigen: Legt fest, ob unterhalb des Schiebereglers die aktuelle Position des Reglers erscheint.
- Anzeige Balken: Die Anzeige der Schiebeflächen oberhalb und unterhalb des Reglers kann ausgeblendet werden

Maximaler Wert und Minimaler Wert

Den oberen und den unteren Endwert der Skala legen Sie im Inspektorfenster fest.

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie bei "Maximaler Skalenwert" und bei "Minimaler Skalenwert" eine Zahl ein. Wenn Sie als Endwert der Skala eine Variable auswählen, dann ist die Zahl nicht mehr verfügbar.

Grenzen/Bereiche konfigurieren

Sie können Grenzen und Bereiche farblich unterschiedlich darstellen. Die Farben legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Grenzen/Bereiche".
2. Aktivieren Sie Grenzen/Bereiche, die in Runtime angezeigt werden sollen
3. Ändern Sie bei Bedarf die voreingestellten Farben für die Bereiche.

Hinweis

Wenn Sie die Option "Bereiche von Variablen anzeigen" aktivieren, können Sie im Schieberegler bis zu fünf Bereiche darstellen, deren Werte über eine Prozessvariable vorgegeben werden. Die Werte für die Bereiche definieren Sie an einer Prozessvariable, die Sie mit dem Bildobjekt verschalten.

Die Option "Bereiche von Variablen anzeigen" steht für Comfort Panels, KTP Mobile Panels und RT Advanced zur Verfügung.

Aktuellen Wert anzeigen

Dass der Wert der aktuellen Position des Reglers unterhalb des Schiebereglers angezeigt wird, wird im Inspektorfenster festgelegt.

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Aktuellen Wert anzeigen".

Anzeige Balken

Die Anzeige des Balken kann ausgeblendet werden.

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Deaktivieren Sie "Balken anzeigen".

Verhalten bei Bedienung

In den folgenden Fällen kann der am Schieberegler angezeigte Wert vom tatsächlichen Wert der zugehörigen Variablen abweichen:

- Der für den Schieberegler projektierte Wertebereich (Minimal- und Maximalwert) korrespondiert nicht mit den projektierten Grenzwerten für die Schieberegler-Variable.
- Für einen kennwortgeschützten Schieberegler wird ein ungültiges Kennwort eingegeben.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.69 Schlüsselschalter

Verwendung

Das Objekt "Schlüsselschalter" ist ein optionales Bedienelement der SIMATIC Mobile Panel. In Runtime dient der Schlüsselschalter der Verriegelungen von Funktionen, die über das Mobile Panel ausgelöst werden.

Das Objekt "Schlüsselschalter" projektieren Sie im "Globalen Bild".

Variable

Mit der Eigenschaft "Variable" wird im Globalen Bild die Belegung des Objekts "Schlüsselschalter" festgelegt.

Hinweis

Hinweise zur Projektierung

Beim Start von Runtime und bei jeder Betätigung des Schlüssels wird die aktuelle Stellung des Schlüsselschalters in die Variable geschrieben. Dabei kann es zu Inkonsistenzen zwischen der Stellung des Schlüsselschalters und der Schlüsselschalter-Variablen kommen.

1. Wenn die Variable eine Prozessverbindung hat, wird nach Kommunikationsaufbau der aktuelle Wert aus der Steuerung in die Variable übertragen. Die Variable, die die aktuelle Stellung des Schlüssels enthält, wird dabei überschrieben. So kann es passieren, dass die Schlüsselschalter-Variable nicht mehr den aktuellen Wert des Schlüsselschalters widerspiegelt, z.B. wenn das Mobile Panel ausgeschaltet wird, während der Schlüssel gedreht war.
2. Wird der Schlüsselschalter gedrückt, bevor die Kommunikation zur Steuerung aufgebaut werden konnte z.B. nach einem Geräte-Anlauf, so kann es vorkommen, dass der Wert der Schlüsselschalter-Variablen nicht zur Steuerung geschickt werden kann. In diesem Fall stimmt dann der Wert der Variablen in der Steuerung nicht mit der aktuellen Stellung des Schlüssels überein.

Um immer sicherstellen zu können, dass die Schlüsselschalter-Variable die tatsächliche Stellung des Schlüsselschalters widerspiegelt, müssen Sie bestimmte Projektierungen vornehmen. In der folgenden Anleitung finden Sie dazu das prinzipielle Vorgehen.

Prinzipielles Vorgehen

1. Legen Sie im Editor "Verbindungen" die Verbindung zur Steuerung an. Damit das Lebensbit auf der Steuerungsseite verfügbar ist, aktivieren Sie den Bereichszeiger "Koordinierung".

2. Legen Sie im Editor "Variable" drei Variablen an.
   - Interne Variable: Stellung_Schluessel
   - Externe Variable: Hilfsvariable
   - Externe Variable: Schluessel_Steuerung
3. Öffnen Sie das Inspektorfenster der Variablen "Stellung_Schluessel".
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Wertänderung".
5. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen erscheint.

6. Wählen Sie die Systemfunktion "SetzeVariable".
   – Wählen Sie bei "Variable (Ausgabe)" die Variable "Schluessel_Steuerung".
   – Wählen Sie bei "Wert" die Variable "Stellung_Schluessel".

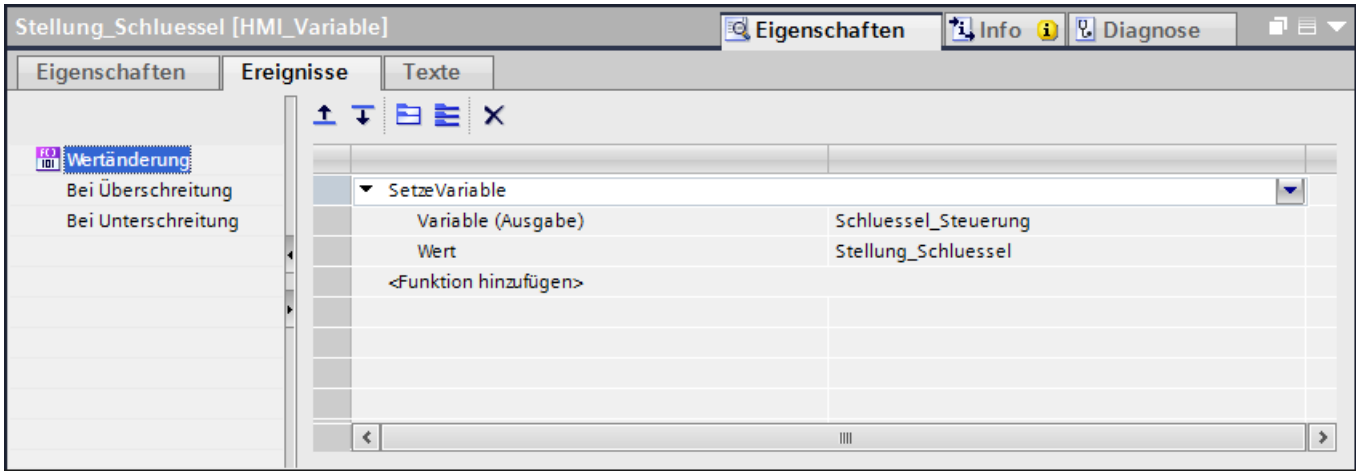

   Mit der Variablen "Schluessel_Steuerung" wird der Wert der Variablen "Stellung_Schluessel" in die Steuerung geschrieben. In der Steuerung wertet ein Programm das Lebensbit aus. Wenn die Kommunikation aufgebaut ist, wird der aktuelle Wert aus der Steuerung in die Variable "Schluessel_Steuerung" geschrieben.

7. Öffnen Sie das Inspektorfenster der Variablen "Hilfsvariable".
8. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Wertänderung".
9. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen erscheint.

10. Wählen Sie die Systemfunktion "SetzeVariable".
    - Wählen Sie bei "Variable (Ausgabe)" die Variable "Schluessel_Steuerung".
    - Wählen Sie bei "Wert" die Variable "Stellung_Schluessel".

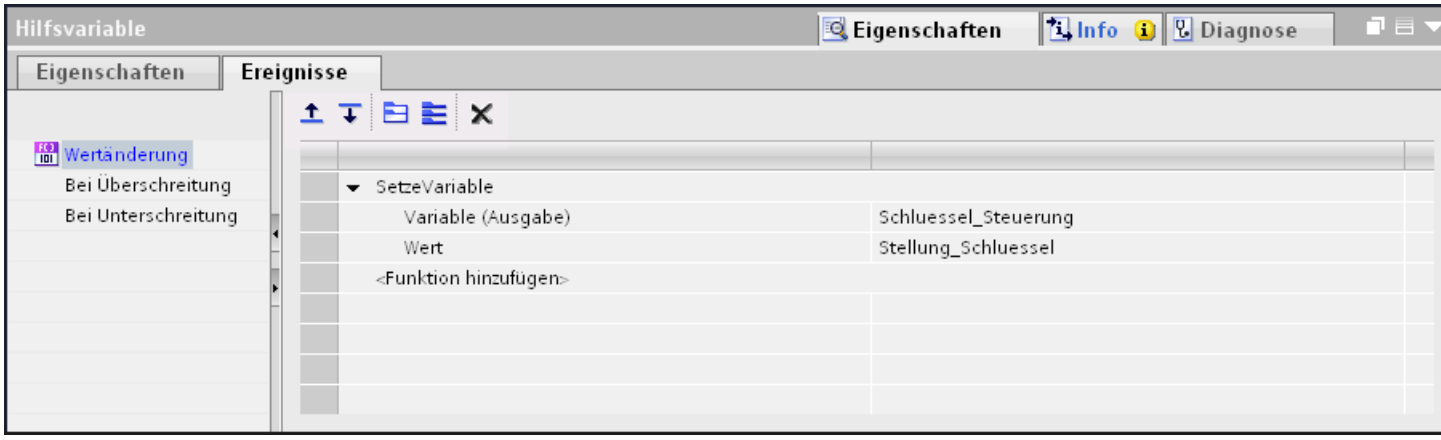

Nach Aufbau der Kommunikation wird der aktuelle Wert aus der Steuerung in die Variable "Hilfsvariable" geschrieben. Durch die Wertänderung in der Hilfsvariablen wird die Systemfunktion "SetzeVariable" ausgeführt. Die Systemfunktion weist der Variablen "Schluessel_Steuerung" wieder den Wert der Variablen "Stellunng_Schluessel" zu.

11. Öffnen Sie im Editor "Bildverwaltung" das Globale Bild.
12. Markieren Sie im Bild den Schlüsselschalter.
13. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein >Einstellungen > Variable" die Variable "Stellung_Schluessel".
    Wenn Sie den Schlüsselschalter bedienen, wird der Wert in die Variable "Stellung_Schlüssel" geschrieben.

1.9.2.70 Sm@rtClient-Anzeige

Verwendung

Mit dem Objekt "Sm@rtClient-Anzeige" projektieren Sie eine Netzwerkverbindung zur Fernüberwachung und Fernwartung eines angeschlossenen Geräts.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Beim Start verbinden: Legt fest, ob beim Öffnen der Sm@rtClient-Anzeige automatisch die Verbindung zum Bediengerät aufgebaut wird, das fernbedient werden soll.
- Gemeinsam genutzt: Legt fest, dass ein Bediengerät von mehreren Sm@rtClient-Anzeigen gemeinsam genutzt wird.
- Lokaler Cursor: Legt fest, ob die Cursordaten separat übertragen werden, um die Performance zu steigern.
- Bildlauf mit Cursortasten: Legt für Tastengerät die Steuerung des horizontalen und vertikalen Bildlaufs auf die Cursortasten.
- Erweiterte Objekte anzeigen: Legt fest, ob es zusätzliche Felder für die Eingabe von Adresse und Kennwort gibt.
- Menü zulassen: Legt fest, ob das Kontextmenü zur Steuerung der Sm@rtClient-Anzeige aktiviert wird.
- Verbindungstyp: Legt die zu erwartende Übertragungsgeschwindigkeit für die Fernüberwachung fest.
- Skalierung: Legt fest, ob sich die Sm@rtClient-Anzeige vergrößern oder verkleinern lässt.

Beim Start verbinden

Wenn diese Eigenschaft gesetzt ist, dann baut die Sm@rtClient-Anzeige automatisch die Verbindung zum Bediengerät auf, das fernbedient wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften >Allgemein".
2. Aktivieren Sie "Anzeige > Beim Start verbinden".

Gemeinsam genutzt

Unter "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Gemeinsam genutzt" legen Sie fest, ob das Bediengerät von mehreren Sm@rtClient-Anzeigen genutzt werden kann.

Gemeinsam genutzt	
Aktiviert	Die Bediengeräte, die die Fernbedienung aktiviert haben, können auf das entfernte Bediengerät zugreifen und den Prozess steuern. Hierbei kann immer nur ein Bediengerät aktiv sein. Ein anderes Bediengerät kann die Steuerung erst dann übernehmen, wenn an dem gerade aktiven Bediengerät in einer festgelegten Zeitspanne keine Aktivität erfolgt.
Deaktiviert	Immer nur ein Bediengerät kann die Fernbedienung nutzen. Auf den anderen Bediengeräten können die Aktivitäten verfolgt werden. Je nach Einstellung am Sm@rtServer gibt es bei Anmeldung eines weiteren Bediengeräts folgende Möglichkeiten: • Das Bediengerät wird abgewiesen. • Die Verbindung zum aktiven Bediengerät wird getrennt und mit dem neuen Bediengerät aufgebaut.

Lokaler Cursor

Wenn diese Eigenschaft gesetzt ist, dann werden die Cursordaten separat übertragen.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Anzeige > Lokaler Cursor".

Erweiterte Objekte anzeigen

Wenn diese Eigenschaft gesetzt ist, dann gibt es zusätzliche Felder für die Eingabe von Adresse und Kennwort. Damit die Felder angezeigt werden, darf die Option "Beim Start verbinden" nicht aktiviert sein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Anzeige > Erweiterte Objekte anzeigen".

Menü zulassen

Wenn diese Eigenschaft gesetzt ist, dann wird in Runtime ein Kontextmenü zur Steuerung der Sm@rtClient-Anzeige geöffnet.

Bei der Bedienung mit der Maus muss die linke Maustaste einige Sekunden gedrückt bleiben, um das Kontextmenü zu öffnen.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Anzeige > Menü zulassen".

Verbindungstyp

Mit der Eigenschaft "Verbindungstyp" legen Sie die zu erwartende Übertragungsgeschwindigkeit für die Fernüberwachung fest.

Den Verbindungstyp legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Verbindungstyp" fest.

Verbindungstyp	
LAN	Verbindung über Netzwerk
Langsam	Verbindung mit niedriger Übertragungsgeschwindigkeit
Mittelschnell	Verbindung mit mittlerer Übertragungsgeschwindigkeit
Modem	Verbindung über ein analoges Modem
Sehr langsam	Verbindung mit sehr niedriger Übertragungsgeschwindigkeit

Skalierung

Wenn diese Eigenschaft gesetzt ist, dann können Sie über "Skalierungsfaktor Client" und "Skalierung Server" die Sm@rtClient-Anzeige am Bediengerät vergrößern oder verkleinern.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Skalierung".
2. Aktivieren Sie "Skalierung > Skalierung".
3. Geben Sie bei "Skalierungsfaktor Client" und bei "Skalierung Server" die Werte für die Skalierung ein.

Verhalten bei Bedienung

Wenn die dynamische Adresse einer Sm@rtClient-Anzeige nicht gesetzt ist, so wird nach einer Bildanwahl die Verbindung zu dem in der statischen Adresse projektierten Server aufgebaut. Wenn die Variable mit der dynamischen Adresse gesetzt wird, könnte eine zweite Verbindung zu diesem Server aufgebaut werden. Zur Abhilfe belegen Sie die Variablen der dynamischen Adresse, z. B. über ein Skript, vor.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.71 Speicherplatzanzeige

Verwendung

Sie verwenden das Objekt "Speicherplatzanzeige" zur Darstellung der Speicherplatzbelegung. Sie erstellen pro Laufwerk eine Speicherplatzanzeige.

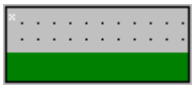

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Laufwerk: Legt das Laufwerk fest, dessen Speicherbelegung in der Speicherplatzanzeige angezeigt wird.
- Überwachungsgrenzen: Legt die Werte für die Überwachungsgrenzen fest.

Laufwerk festlegen

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Tragen Sie unter "Laufwerksbuchstabe" den entsprechenden Buchstaben ein.

Überwachungsgrenzen ändern

Im Inspektorfenster geben Sie die Überwachungsgrenzen für Alarm, Warnung und Toleranz ein. Die Werte werden in Prozent angegeben. Überwacht wird der prozentuale Anteil des belegten Speicherplatzes an der Gesamtkapazität des angegebenen Laufwerks:

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Tragen Sie im Bereich "Grenzen" die entsprechenden Werte für die jeweiligen Überwachungsgrenzen in Prozent ein.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.72 Symbolbibliothek

Verwendung

Das Objekt "Symbolbibliothek" beinhaltet eine umfangreiche Sammlung von vorgefertigten Symbolen. Mit diesen Symbolen stellen Sie in Bildern Anlagen und Anlagenbereiche dar.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Symbol wählen: Legt fest, welches Bibliotheksobjekt verwendet wird.
- Füllart: Legt die grafische Gestaltung des Symbols fest.
- Spiegeln: Legt die Art der Spiegelung des Bibliotheksobjekts fest.
- Drehen: Legt den Winkel fest, um den das Bibliotheksobjekt gedreht wird.
- Festes Seitenverhältnis: Legt fest, ob bei Änderung der Größe das Seitenverhältnis beibehalten wird.

Symbol wählen

Sie wählen das Symbol im Inspektorfester unter "Eigenschaften > Eigenschaften > Allgemein".

Füllart

Die Gestaltung des Bibliotheksobjekts legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Gestaltung > Stil" fest.

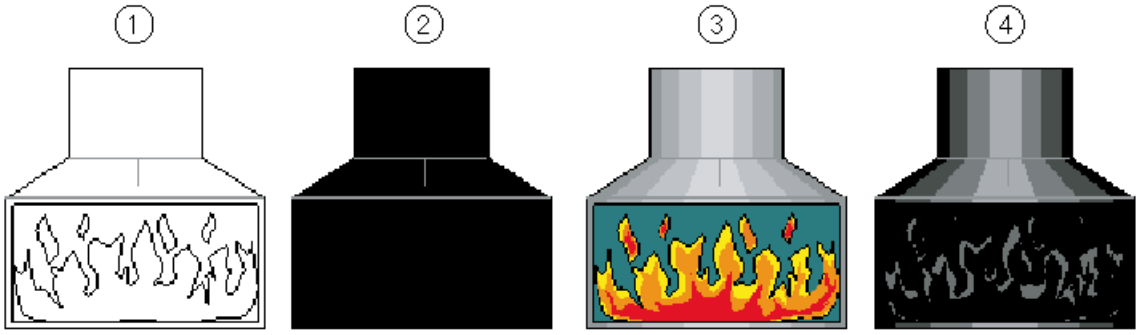

1	"Hohl"	Das Bibliotheksobjekt wird als Umriss dargestellt.
2	"Massiv"	Linien der Farbe "Schwarz" bleiben als Umrisslinien erhalten. Andersfarbige Elemente des Symbols wird eine Grundfarbe zugewiesen.
3	"Original"	Das Bildobjekt wird nicht verändert.
4	"Schattiert"	Linien der Farbe "Schwarz" bleiben als Umrisslinien erhalten. Andersfarbige Elemente des Symbols werden als Helligkeitsstufen einer Grundfarbe dargestellt.

Wenn Sie die Einstellung von "Hohl" nach "Original" oder "Schattiert" ändern, dann kann die Darstellung am Projektierungs-PC von der Darstellung am Bediengerät farblich abweichen. Die Abweichung wird durch die Unterschiede in der Farbtiefe verursacht. Alternativ können Sie auch ein grafisches Objekt aus der Objektgruppe "Grafik" verwenden. Im Ordner "WinCC Grafikordner" sind die grafischen Objekte thematisch und nach Farbtiefe gegliedert.

Spiegeln und Drehen

Die Spiegelung des Bildinhalts wird jeweils an der horizontalen oder vertikalen Mittelachse des Symbols ausgeführt. Das Symbol kann horizontal, vertikal oder gleichzeitig horizontal und vertikal gespiegelt werden.

Die Drehung des Bildinhalts wird um den Mittelpunkt des Symbols ausgeführt. Das Symbol wird im Uhrzeigersinn in Schritten von 90, 180 oder 270 Grad gedreht.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie unter "Orientierung > Spiegeln" die gewünschte Spiegelungsart.
3. Wählen Sie unter "Orientierung > Drehen" den gewünschten Winkel für die Drehung.

Festes Seitenverhältnis

Die kleinere Seite des Bibliotheksobjekts bestimmt die maximale Größe des Symbols. Wenn Sie die Größe des Bibliotheksobjekts unproportional ändern, wird das Symbol weiterhin proportional skaliert. Um ein festes Seitenverhältnis beizubehalten, gehen Sie wie folgt vor:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Form > Festes Seitenverhältnis".

Verhalten bei Bedienung

Abhängig von der Projektierung wird in Runtime die Möglichkeit der Mausbedienung durch eine Änderung des Cursorsymbols signalisiert. Eine Bedienungsrückmeldung, z.B. durch Farbänderung, erfolgt nicht.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.73 Symbolisches E/A-Feld

Verwendung

Mit dem Objekt "Symbolisches E/A-Feld" projektieren Sie eine Auswahlliste, die der Eingabe und Ausgabe von Texten in Runtime dient.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Textliste: Legt die Textliste fest, die mit dem Objekt verknüpft wird.
- Schaltfläche für Auswahlliste: Legt fest, dass das Objekt eine Schaltfläche zum Öffnen der Auswahlliste besitzt.

Hinweis

Protokolle

In Protokollen geben symbolische E/A-Felder ausschließlich Daten aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Auswahl von Grafiken stehen nicht zur Verfügung, z. B. "Schaltfläche für Auswahlliste".

Modus

Das Verhalten des Symbolischen E/A-Feldes legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" fest.

Modus	Beschreibung
"Ausgabe"	Das Symbolische E/A-Feld dient zur Ausgabe von Werten.
"Eingabe"	Das Symbolische E/A-Feld dient zur Eingabe von Werten.

Modus	Beschreibung
"Eingabe/Ausgabe"	Das Symbolische E/A-Feld dient zur Eingabe und Ausgabe von Werten.
"Zwei Zustände"	Das Symbolische E/A-Feld wird nur zur Ausgabe von Werten eingesetzt und besitzt maximal zwei Zustände. Das Feld schaltet zwischen zwei vordefinierten Texten um. Damit werden z. B. die beiden Zustände eines Ventils, geschlossen oder offen, visualisiert.

Hinweis

Welches Verhalten für das Symbolische E/A-Feld möglich ist, hängt von der Runtime ab.

Textliste

Im Inspektorfenster legen Sie fest, welche Textliste mit dem Symbolischen E/A-Feld verknüpft wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie unter "Inhalt" die Auswahlliste für "Textliste".
3. Wählen Sie eine Textliste aus.

Schaltfläche für Auswahlliste

Mit der Eigenschaft "Schaltfläche für Auswahlliste" wird eine Schaltfläche zum Öffnen der Auswahlliste angezeigt.

1. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Verhalten > Schaltfläche für Auswahlliste".

   Hinweis
   Basic Panels

   Die Option "Schaltfläche für Auswahlliste" steht für Basic Panels nicht zur Verfügung.

Verhalten bei der Bedienung

Wenn das symbolische EA-Feld eine leere Textzeile in der Auswahlliste enthält, ist im Projekt kein Eintrag definiert worden. Die Aktivierung wird Ihnen am Bediengerät dadurch angezeigt, dass der Inhalt des symbolischen EA-Feldes seine Farbe ändert.

Besonderheiten

- Bei Comfort Panels und RT Advanced steht die Eigenschaft "Eckradius" für Symbolisches E/A-Feld nicht zur Verfügung. Der Eckradius, den Sie für das Objekt "Symbolisches E/A-Feld" im Stileditor festlegen, hat im Bildeditor für diese Zielsysteme keine Wirkung.

Siehe auch

Variablen und Textlisten im Text anbinden (Seite 127)

Grafisches E/A-Feld (Seite 393)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.74 Symbolisches E/A-Feld

Verwendung

Mit dem Objekt "Symbolisches E/A-Feld" projektieren Sie eine Auswahlliste, die der Eingabe und Ausgabe von Texten in Runtime dient.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere können Sie die folgenden Eigenschaften anpassen:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Textliste: Legt die Textliste fest, die mit dem Objekt verknüpft wird.

Modus

Das Verhalten des Symbolischen E/A-Feldes wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" festgelegt.

Modus	Beschreibung
"Ausgabe"	Das Symbolische E/A-Feld dient zur Ausgabe von Werten.
"Eingabe"	Das Symbolische E/A-Feld dient zur Eingabe von Werten.
"Eingabe/Ausgabe"	Das Symbolische E/A-Feld dient zur Eingabe und Ausgabe von Werten.
"Zwei Zustände"	Das Symbolische E/A-Feld wird nur zur Ausgabe von Werten eingesetzt und besitzt maximal zwei Zustände. Das Feld schaltet zwischen zwei vordefinierten Texten um. Damit werden z. B. die beiden Zustände eines Ventils, geschlossen oder offen, visualisiert.

Textliste

Im Inspektorfenster wird festgelegt, welche Textliste mit dem Symbolischen E/A-Feld verknüpft wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie die Auswahlliste für "Textliste".
3. Wählen Sie eine Textliste aus.

Siehe auch

Grafisches E/A-Feld (Seite 395)

Bedienmeldungen projektieren (Seite 862)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.75 System-Diagnoseanzeige

Einleitung

Die System-Diagnoseanzeige bietet Ihnen einen Überblick über alle verfügbaren Geräte in Ihrer Anlage. Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

Verwendung

Durch die System-Diagnoseanzeige erreichen Sie den höchstmöglichen Detaillierungsgrad der Diagnosedaten. Eine präzise Diagnose ist möglich, da alle verfügbaren Daten angezeigt werden. Sie haben den Systemzustand der gesamten Anlage auf einen Blick.

Ansichten in der System-Diagnoseanzeige

Auf Basic Panels mit der Geräteversion kleiner als V13 steht Ihnen eine einfache System-Diagnoseanzeige zur Verfügung.

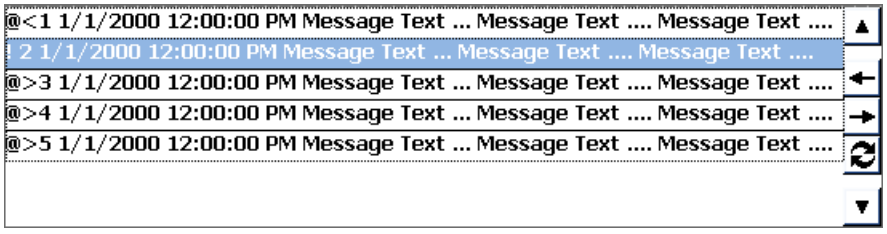

Auf Basic Panels mit der Geräteversion V13 und höher steht Ihnen die erweiterte System-Diagnoseanzeige zur Verfügung.

In der System-Diagnoseanzeige stehen Ihnen drei verschiedene Ansichten zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht

Geräteansicht

Die Geräteansicht der System-Diagnoseanzeige zeigt in tabellarischer Sicht alle verfügbaren Verbindungen. Durch einen Doppelklick auf eine Verbindung öffnet die Detailansicht. Die Geräteansicht wird nur angezeigt, wenn mehr als eine Verbindung im Editor "Geräte & Netze" angelegt wurde.

Diagnosepuffer-Ansicht

In der Diagnosepuffer-Ansicht werden die aktuellen Daten aus dem Diagnosepuffer angezeigt.

Detailansicht

Die Detailansicht zeigt ausführliche Informationen über die selektierte Verbindung. Fehlertexte können Sie in der Detailansicht nicht sortieren. Die Detailansicht ist nur verfügbar, wenn eine integrierte Verbindung zu einer S7-1200 oder S7-1500 existiert.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Zeilen pro Eintrag: Legt die Anzahl der Zeilen fest, die für einen Eintrag angezeigt werden.

System-Diagnoseanzeige projektieren

1. Ziehen Sie per Drag&Drop die System-Diagnoseanzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Darstellung".
3. Wählen Sie für "Zeilen pro Eintrag" z. B. 5.
4. Wählen Sie "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

Siehe auch

Grundlagen für Basic Panels (Seite 1901)

Objekte der Systemdiagnose projektieren (Seite 1908)

1.9.2.76 System-Diagnoseanzeige

Einleitung

Die System-Diagnoseanzeige bietet Ihnen einen Überblick über alle verfügbaren Geräte in Ihrer Anlage. Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

Verwendung

Durch die System-Diagnoseanzeige erreichen Sie den höchstmöglichen Detaillierungsgrad der Diagnosedaten. Eine präzise Diagnose ist möglich, da alle verfügbaren Daten angezeigt werden. Sie haben den Systemzustand der gesamten Anlage auf einen Blick.

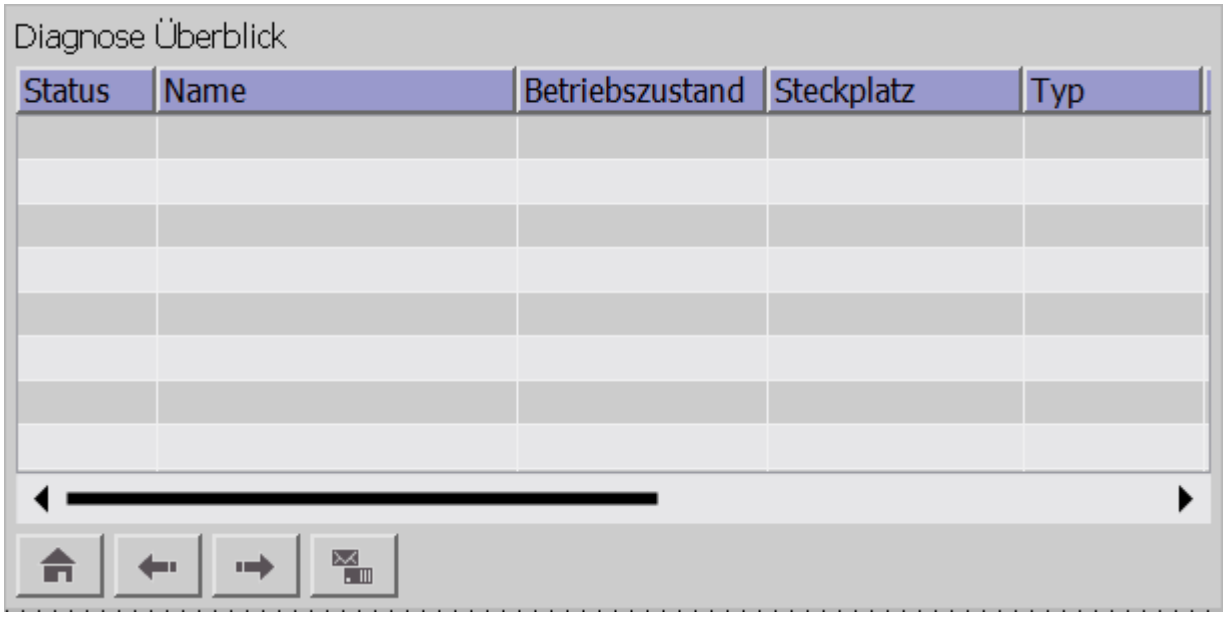

Vier verschiedene Ansichten stehen Ihnen in der System-Diagnoseanzeige zur Verfügung:

- Geräteansicht
- Detailansicht
- Diagnosepuffer-Ansicht
- Dezentrale I/O Ansicht

Hinweis

Das Objekt "System-Diagnoseanzeige" kann nicht in Bildbausteinen verwendet werden.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Spalten: Legt die Elemente der Geräteansicht und Detailansicht fest.
- Anzeige teilen: Legt fest, dass die Geräteansicht und die Detailansicht gleichzeitig angezeigt werden.

System-Diagnoseanzeige projektieren

1. Ziehen Sie per Drag&Drop die System-Diagnoseanzeige aus dem Werkzeugfenster.
2. Legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Modus" die Startansicht für die System-Diagnoseanzeige fest.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
4. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. : Betriebzustand, Name, Steckplatz.
5. Aktivieren Sie die Spalten, die Sie in der Detailansicht für Runtime benötigen, z. B. : Betriebszustand, Name, Adresse.
6. Aktivieren Sie die Spalten, die Sie in der Diagnosepuffer-Ansicht benötigen z. B. Zustand, Name, Baugruppenträger.
7. Passen Sie bei Bedarf die Überschriften für die Spalten an.
8. Geben Sie z. B. für "Adresse" "IP" ein.
   In Runtime wird in der Gerätesicht die Spaltenüberschrift "IP" angezeigt.
9. Benennen Sie bei Bedarf weitere Elemente um.
10. Wählen Sie "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

Geräteansicht und Detailansicht in einer Anzeige darstellen

Sie können die System-Diagnosenanzeige aufteilen und die Geräte- und Detailansicht im gleichen Fenster anzeigen lassen.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Layout".
2. Aktivieren Sie "Anzeige teilen".

Symbole in der System-Diagnoseaneige

Symbol	Funktion
	Gerät in Betrieb
	Gerät nicht erreichbar
	Fehler in Gerät
	Ein-/Ausgangsdaten sind nicht verfügbar
	Gerät deaktiviert
	Wartung erforderlich
	Wartung empfohlen
	Überlagerndes Symbol Zeigt den untergeordneten Status
	Laufende Konfiguration
	Stop z.B. Update, Urladen, Eigeninitialisierung
	Halt

Siehe auch

System-Diagnosefenster (Seite 513)

Geräteabhängigkeit der Objekte (Seite 337)

Grundlagen für Panels und Runtime Advanced (Seite 1904)

Objekte der Systemdiagnose projektieren (Seite 1908)

1.9.2.77 System-Diagnoseanzeige

Einleitung

Die System-Diagnoseanzeige bietet Ihnen einen Überblick über alle verfügbaren Geräte in Ihrer Anlage. Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

Verwendung

Durch die System-Diagnoseanzeige erreichen Sie den höchstmöglichen Detaillierungsgrad der Diagnosedaten. Eine präzise Diagnose ist möglich, da alle verfügbaren Daten angezeigt werden. Sie haben den Systemzustand der gesamten Anlage auf einen Blick.

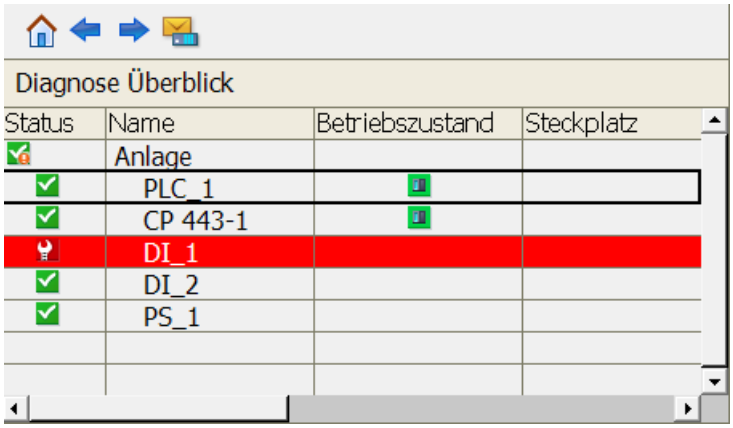

Drei verschiedene Ansichten stehen Ihnen in der System-Diagnoseanzeige zur Verfügung.

- Geräteansicht
- Detailansicht
- Diagnose-Pufferansicht

Hinweis

Das Objekt "System-Diagnoseanzeige" kann nicht in Bildbausteinen verwendet werden.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Spalten: Legt die Elemente der Geräteansicht und Detailansicht fest.
- Anzeige teilen: Legt fest, dass die Geräteansicht und die Detailansicht gleichzeitig angezeigt werden.

System-Diagnoseanzeige projektieren

1. Ziehen Sie per Drag&Drop die System-Diagnoseanzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
3. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. : Zustand, Name, Steckplatz.
4. Aktivieren Sie die Spalten, die Sie in der Detailansicht für Runtime benötigen, z. B. : Zustand, Name, Adresse.
5. Passen Sie bei Bedarf die Überschriften für die Spalten an.

6. Geben Sie z. B. für "Adresse" "IP" ein.
   In Runtime wird in der Gerätesicht die Spaltenüberschrift "IP" angezeigt.
7. Benennen Sie bei Bedarf weitere Elemente um.
8. Wählen Sie "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

Geräteansicht und Detailansicht in einer Anzeige darstellen

Sie können die System-Diagnosenanzeige aufteilen und die Geräte- und Detailansicht im gleichen Fenster anzeigen lassen.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Layout".
2. Aktivieren Sie "Anzeige teilen".

Einfache Anzeige

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie "Modus > Anzeigentyp > Einfach".

Symbole in der System-Diagnoseaneige

Symbol	Funktion
	Gerät in Betrieb
	Gerät nicht erreichbar
	Fehler in Gerät
	Ein-/Ausgangsdaten sind nicht verfügbar
	Gerät deaktiviert
	Wartung erforderlich
	Wartung empfohlen
	Überlagerndes Symbol Zeigt den untergeordneten Status
	Laufende Konfiguration
	Stop z.B. Update, Urladen, Eigeninitialisierung
	Halt

Siehe auch

Grundlagen für Runtime Professional (Seite 1917)

Objekte der Systemdiagnose projektieren (Seite 1920)

Beispiel: Systemdiagnose mit allen Objekten (Seite 1924)

1.9.2.78 System-Diagnosefenster

Einleitung

Das System-Diagnosefenster bietet Ihnen einen Überblick über alle verfügbaren Geräte in Ihrer Anlage. Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

Das System-Diagnosefenster ist nur im Globalen Bild verfügbar.

Verwendung

Durch das System-Diagnosefenster erreichen Sie den höchstmöglichen Detaillierungsgrad der Diagnosedaten. Eine präzise Diagnose ist möglich, da alle verfügbaren Daten angezeigt werden. Sie haben den Systemzustand der gesamten Anlage auf einen Blick.

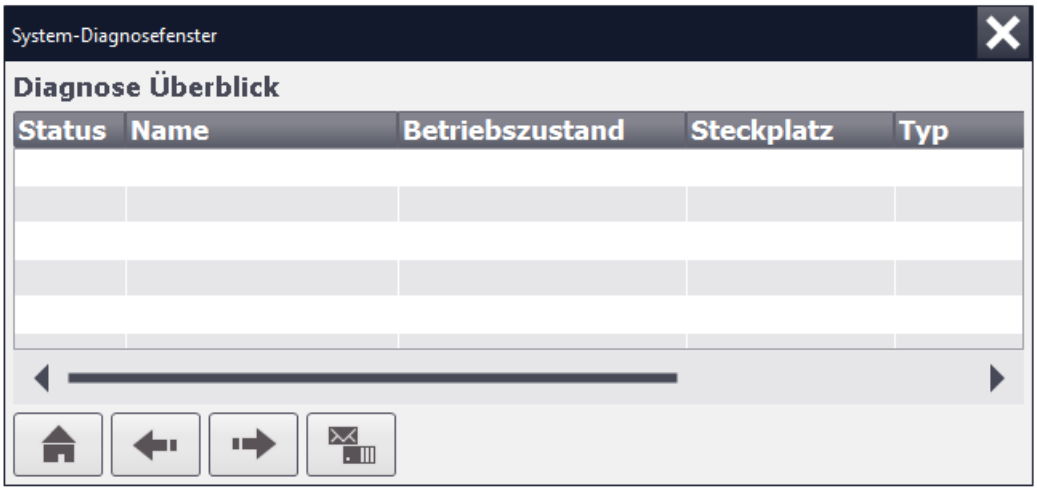

Vier verschiedene Ansichten stehen Ihnen im System-Diagnosefenster zur Verfügung:

- Geräteansicht
- Detailansicht
- Diagnosepuffer-Ansicht
- Dezentrale I/O Ansicht

Hinweis

Das Objekt "System-Diagnosefenster" kann nicht in Bildbausteinen verwendet werden.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Spalten: Legt die Elemente der Geräteansicht und Detailansicht fest.
- Spaltenköpfe: Legt die Spaltenköpfe der Anzeige fest.
- Fenster: Legt fest, ob das System-Diagnosefenster z. B. schließbar ist.

System-Diagnosefenster projektieren

1. Ziehen Sie per Drag&Drop das System-Diagnoseanzeige aus dem Werkzeugfenster in das Globale Bild.
2. Legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Modus" die Startansicht für das System-Diagnosefenster fest.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
4. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. :Betriebszustand, Name, Steckplatz.
5. Aktivieren Sie die Spalten, die Sie in der Detailansicht für Runtime benötigen, z. B. Betriebszustand, Name, Anlagenkennzeichen
6. Um die Spaltenüberschrift zu ändern, klicken Sie in das Feld und geben Sie z. B. für "Adresse" "IP" ein.
   In Runtime wird in der Gerätesicht die Spaltenüberschrift "IP" angezeigt.
7. Benennen Sie bei Bedarf weitere Elemente um.
8. Um das System-Diagnosefenster in Runtime schließen zu können, aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Fenster > Schließbar".

Symbole im System-Diagnosefenster

Schaltfläche	Funktion
	Gerät in Betrieb
	Gerät nicht erreichbar
	Fehler in Gerät
	Gerät deaktiviert
	Ein-/Ausgangsdaten sind nicht verfügbar
	Wartung erforderlich
	Wartung empfohlen
	Überlagerndes Symbol Zeigt den untergeordneten Status
	Laufende Konfiguration

Schaltfläche	Funktion
	Stop z.B. Update, Urladen, Eigeninitialisierung
	Halt

Systemdiagnose-Fenster: Drahtbruchfehler anzeigen

Sie haben für eine Baugruppe Drahtbruch-Diagnose aktiviert. Beim Auftreten eines Drahtbruchfehlers wird im System-Diagnosefenster der Fehlertext angezeigt. In der Detailansicht der Baugruppe sehen Sie den Fehlertext und mögliche Fehlerursachen. Die Ausgabe der Kanalnummer des betroffenen Kanals wird nicht unterstützt.

Siehe auch

System-Diagnoseanzeige (Seite 508)

Geräteabhängigkeit der Objekte (Seite 337)

Grundlagen für Panels und Runtime Advanced (Seite 1904)

Objekte der Systemdiagnose projektieren (Seite 1908)

1.9.2.79 Tabellenanzeige

Verwendung

Mit dem Objekt "Tabellenanzeige" stellen Sie die Werte von Variablen in einer Tabelle dar. Sie können in der Tabelle aktuell anfallende oder archivierte Werte darstellen.

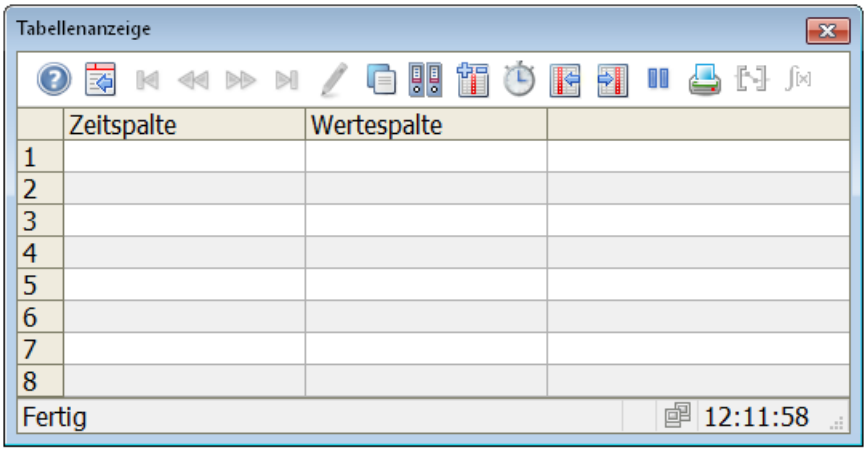

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Datenquelle und Spalten konfigurieren
- Tabelle konfigurieren
- Einstellungen für das Fenster
- Persistenz
- Symbolleiste

Datenquelle und Spalten konfigurieren

1. Öffnen Sie "Eigenschaften > Eigenschaften > Wertspalte".
2. Wählen Sie unter "Datenquelle" die Archivvariable oder Verdichtungsarchivvariable, welche die Spalte mit Werten versorgt.
3. Wählen Sie unter "Zeitspalte", in welcher Spalte der Tabellenanzeige die Datenquelle angezeigt wird.
4. Wählen Sie unter "Eigenschaften > Eigenschaften >Zeitspalte > Zeitbereich" den Zeitbereich der Tabelle:
   - "Zeitspanne": Sie legen den Zeitbereich über einen Startzeitpunkt und eine nachfolgende Zeitspanne fest.
   - "Endzeit": Sie legen den Zeitbereich über einen Startzeitpunkt und einen Endzeitpunkt fest.
   - "Messpunkte": Sie legen den Zeitbereich über einen Startzeitpunkt und eine Anzahl von Messpunkten fest.
5. Legen Sie mit "Format > Aktualisierung" fest, ob die Tabelle die Werte statisch darstellt oder die Werte aktualisiert werden.
6. Legen Sie unter "Stil" das Format für die Anzeige der Uhrzeit und die Ausrichtung innerhalb der Spalte fest.

Tabelle konfigurieren

Unter "Eigenschaften > Eigenschaften > Tabelle" legen Sie fest, ob die Spalten der Tabelle eine Überschrift haben und ob die Rasterlinien der Tabelle angezeigt werden.

Einstellungen für das Fenster

Unter "Eigenschaften > Eigenschaften > Darstellung > Fenster" legen Sie fest, ob der Benutzer in Runtime die Größe des Fensters ändern und das Fenster schließen kann.

Persistenz

Unter "Eigenschaften > Eigenschaften > Security > Persistenz" legen Sie fest, wie lange die Änderungen eines Benutzers an der Kurvenanzeige in Runtime erhalten bleiben.

- "Persistenz erlauben"
  Der Benutzer kann Einstellungen ändern. Die Änderungen bleiben bis zum Bildwechsel erhalten.
- "Persistenz bis Abmelden"
  Änderungen der Benutzergruppe bleiben bis zum Abmelden erhalten.
- "Persistenz bis zum nächsten Laden"
  Änderungen der Benutzergruppe bleiben erhalten, bis das Projekt auf dem Bediengerät aktualisiert wird.

Symbolleiste

Die Bedienelemente der Tabellenanzeige in Runtime legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste" fest. Folgende Bedienelemente stehen für die Tabellenanzeige zur Verfügung:

Schaltfläche	Name	Funktion
	Online-Hilfe	Öffnet die Online-Hilfe.
	Konfiguration öffnen	Öffnet den Konfigurationsdialog.
	Erster Datensatz	Im Archiv blättern: Zeigt die Werte der Variable beginnend mit dem ersten archivierten Wert.
	Vorhergehender Datensatz	Im Archiv blättern: Zeigt die Werte der Variable im vorhergehenden Zeitintervall.
	Nächster Datensatz	Im Archiv blättern: Zeigt die Werte der Variable im nachfolgenden Zeitintervall.
	Letzter Datensatz	Im Archiv blättern: Zeigt die Werte der Variable bis zum letzten archivierten Wert.
	Daten bearbeiten	Ermöglicht die Bearbeitung von Daten in einem beliebigen Tabellenfeld, dass der Benutzer mit einem Doppelklick öffnet.
	Variablenauswahl	Öffnet den Dialog zur Auswahl der Archive und Variablen.
	Spalte auswählen	Öffnet den Dialog zum Einstellen der Sichtbarkeit von Spalten.
	Auswahl des Zeitbereichs	Öffnet den Dialog zum Einstellen des Zeitbereichs, der in der Tabellenanzeige dargestellt wird.
	Vorhergehende Spalte	Zeigt die vorhergehende Spalte im Vordergrund.

Schaltfläche	Name	Funktion
	Nächste Spalte	Zeigt die nächste Spalte im Vordergrund.
	Starte/Beende Update	Stoppt und startet die Spaltenaktualisierung. Die Werte werden zwischengespeichert und nachgetragen, sobald Sie die Spaltenaktualisierung wieder starten.
	Druckarchiv	Startet den Druck der in der Tabellenanzeige angezeigten Spalten. Den Druckauftrag legen Sie unter "Allgemein > Drucken > Druckauftrag" fest.
	Auswahl des Statistikbereichs	Ermöglicht die Festlegung eines Zeitbereichs, für den statistische Werte ermittelt werden.
	Statistikfenster	Öffnet das Statistikfenster zur Anzeige von Minimum, Maximum, Durchschnitt und Standardabweichung für den gewählten Zeitbereich und die gewählte Spalte.

Wertespalte in der Tabellenanzeige dynamisieren (RT Professional)

Beachten Sie folgende Informationen bei der Dynamisierung der Eigenschaften in der Wertespalte einer Tabellenanzeige:

- Die Eigenschaft "ValueColumnName" legt den Namen der ausgewählten Wertspalte fest. Das Attribut "ValueColumnName" ist über das Attribut "ValueColumnRename" dynamisierbar. Der Datentyp ist STRING.
- Die Eigenschaft "ValueColumnRename" ändert den Namen der Wertspalte, die über das Attribut "ValueColumnIndex" referenziert wird. Das Attribut ist mit dem Namen "ValueColumnRename" dynamisierbar. Mit "ValueColumnRename" dynamisieren Sie auch das Attribut "ValueColumnName". Der Datentyp ist STRING.
- Die Eigenschaft "TimeColumnName" legt den Namen der ausgewählten Zeitspalte fest. Das Attribut "TimeColumnName" ist über das Attribut "TimeColumnRename" dynamisierbar. Der Datentyp ist STRING.
- Die Eigenschaft "TimeColumnRename" ändert den Namen der Zeitspalte, die über das Attribut "TimeColumnIndex" referenziert wird. Das Attribut ist mit dem Namen "TimeColumnRename" dynamisierbar. Mit "TimeColumnRename" dynamisieren Sie auch das Attribut "TimeColumnName". Der Datentyp ist STRING.

Siehe auch

f(t)-Kurvenanzeige (Seite 380)

f(x)-Kurvenanzeige (Seite 387)

Tabellenanzeige projektieren (Seite 741)

Wertetabelle (Seite 523)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

Tabellenanzeige in Runtime bedienen (Seite 775)

1.9.2.80 Textfeld

Verwendung

Das Objekt "Textfeld" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Text: Legt den Text für das Textfeld fest.
- Größe des Textfelds: Legt fest, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

Text

Der Text für das Textfeld wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie einen Text ein.
   Für mehrzeiligen Text können Sie einen Zeilenumbruch durch Drücken der Tastenkombination <Umschalttaste + Enter> setzen.

Größe des Textfelds

Im Inspektorfenster wird festgelegt, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Größenanpassung > Größe an Inhalt anpassen".

Siehe auch

Variablen und Textlisten im Text anbinden (Seite 127)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.81 T-Stück

Verwendung

Mit dem Objekt "T-Stück" bilden Sie eine T-förmige Rohrverbindung nach. Um eine Rohranlage nachzubilden, verbinden Sie das Objekt "T-Stück" mit weiteren Objekten z. B. Rohr oder Rohrbogen.

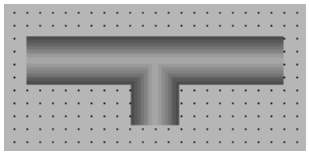

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

Drehung: Legt die Drehung des T-Stücks in Grad fest.

Drehung

Im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung > Drehung" legen Sie fest, um wie viel Grad das "T-Stück" gedreht wird. Der eingegebene Wert wird automatisch auf ein Vielfaches von 90 gerundet.

- 0: das "Bein" des T-Stücks zeigt nach unten
- 90: das "Bein" des T-Stücks zeigt nach links
- 180: das "Bein" des T-Stücks zeigt nach oben
- 270: das "Bein" des T-Stücks zeigt nach rechts

Siehe auch

Rohr (Seite 487)

Rohrbogen (Seite 487)

Doppel-T-Stück (Seite 364)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.82 Uhr

Verwendung

Das Objekt "Uhr" dient zur Anzeige von Datum und Uhrzeit.

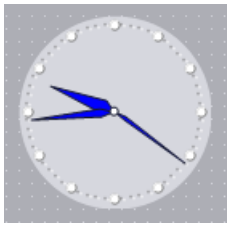

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Analoge Anzeige: Legt fest, ob die Uhr als Analoguhr oder als Digitaluhr dargestellt wird.
- Ziffernblatt anzeigen: Legt fest, ob die Stundenmarken der Analoguhr angezeigt werden.
- Breite und Länge der Zeiger: Legt die Breite und Länge der Zeiger fest.

Analoge Anzeige

Im Inspektorfenster wird festgelegt, ob die Uhr als Analoguhr oder als Digitaluhr angezeigt wird. Die Digitaluhr enthält zusätzlich zur Uhrzeit das aktuelle Datum. Das Anzeigeformat ist von der eingestellten Sprache am Bediengerät abhängig.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Analog".

Ziffernblatt anzeigen

Im Inspektorfenster wird festgelegt, ob die Stundenmarken angezeigt werden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Analog".
3. Aktivieren Sie "Ziffernblatt anzeigen".

Breite und Länge der Zeiger

Für die analoge Anzeige stellen Sie die Breite von Sekundenzeiger, Minutenzeiger und Stundenzeiger ein.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie für "Breite" und "Länge" Werte ein.
   Die Werte für die Länge werden als prozentualer Anteil an der Größe der Uhr interpretiert. Der Wert für die Länge des Sekundenzeigers beeinflusst gleichzeitig den Radius des Ziffernblatts.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.83 Verbinder

Verwendung

Mit dem Objekt "Verbinder" verbinden Sie mehrere Objekte durch eine Linie miteinander. Sie können auch mehrere Verbinder aneinanderhängen.

Es gibt die folgenden beiden Verbindertypen:

- Automatisch: Der Verbinder setzt sich aus horizontalen und vertikalen Teilstücken zusammen.

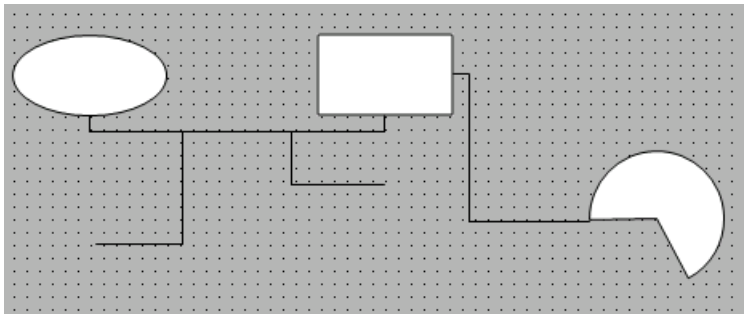

- Einfach: Die Verbindungspunkte werden durch eine gerade Linie verbunden.

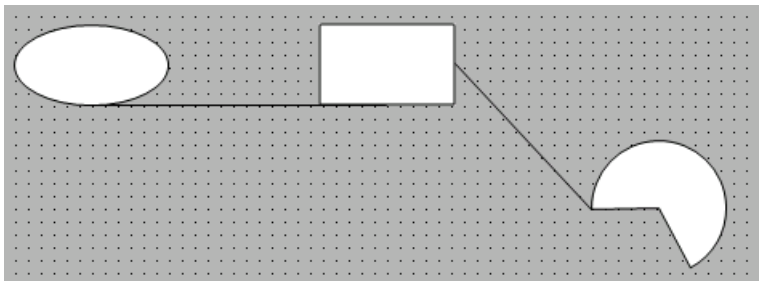

Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Linienstil
- Linienanfang und Linienende

Objekte verbinden

Um Objekte zu verbinden, führen Sie die Enden der Verbinder mit den Verbindungspunkten der Objekte zusammen.

1. Klicken Sie in der Task Card "Werkzeuge" auf das Objekt "Verbinder".
2. Klicken Sie im Bild auf ein Objekt. Der Verbinder ist mit dem ersten Objekt verbunden.
3. Klicken Sie auf ein anderes Objekt im Bild.
   Die beiden Objekte sind miteinander verbunden.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.84 Wertetabelle

Verwendung

Sie verwenden das Objekt "Wertetabelle", um ausgewertete Daten und Statistiken in einer Tabelle anzuzeigen.

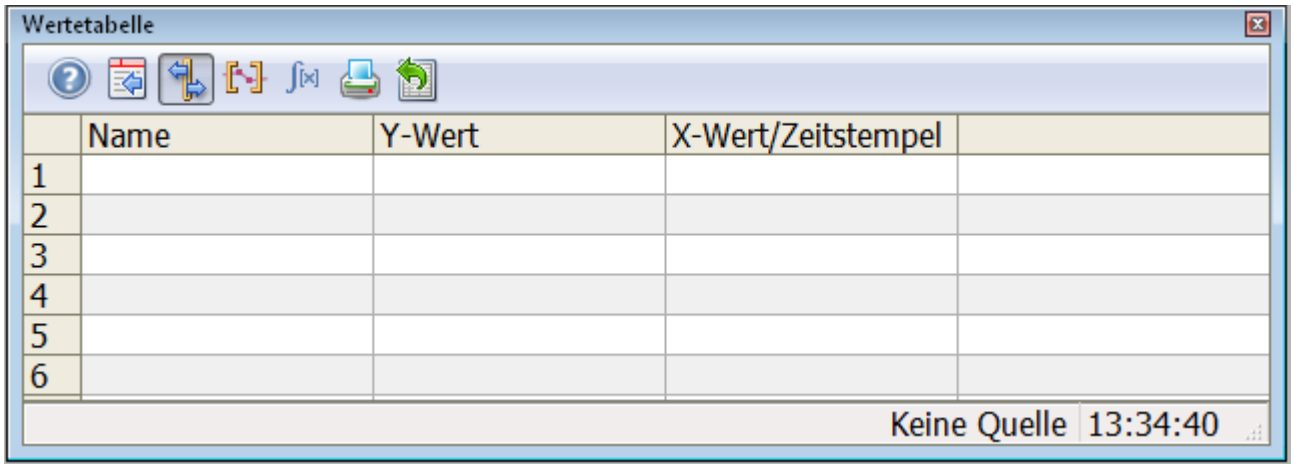

- f(t)-Kurvenanzeige
- f(x)-Kurvenanzeige
- Tabellenanzeige

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Datenquelle zur Darstellung der Werte
- Modus
- Symbolleiste

Datenquelle zur Darstellung der Werte

Welche Werte in der Wertetabelle dargestellt werden, legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Datenquelle" fest.

Wählen Sie zwischen folgenden Objekten:

- f(t)-Kurvenanzeige
- f(x)-Kurvenanzeige
- Tabellenanzeige

Standardmäßig wird bei der Projektierung für das Anzeigeformat das Format des verbundenen Controls übernommen. Dabei werden Größe, Wertebereich und Zoomfaktor des Controls berücksichtigt, um die optimale Anzahl von Nachkommastellen darzustellen. Sie können die Anzeigeformate für einzelne Werte im Inspektorfenster der Wertetabelle selbst konfigurieren, z. B. die Anzeige einer genauen Anzahl von Nachkommastellen.

Modus

Den Modus legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Modus" fest. Je nach Datenquelle haben Sie drei verschiedene Typen zur Auswahl.

- Das Linealfenster zeigt die Koordinatenwerte der Kurven am Lineal oder Werte einer selektierten Zeile in der Tabelle.
- Das Statistikbereich-Fenster zeigt die Werte der Untergrenze und Obergrenze der Kurven zwischen zwei Linealen oder des selektierten Bereichs in der Tabelle. Das Statistikbereich-Fenster ist nicht für das Objekt f(x)-Kurvenanzeige vorgesehen.
- Das Statistikfenster zeigt die statistische Auswertung der Kurven zwischen zwei Linealen oder der selektierten Werte in der Tabelle. Das Statistikfenster ist nicht für das Objekt "f(x)-Kurvenanzeige" vorgesehen.

Symbolleiste

Die Bedienelemente der Wertetabelle in Runtime legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen" fest. Folgende Bedienelemente stehen für die Wertetabelle zur Verfügung:

Schaltfläche	Kurzbeschreibung	Beschreibung
	Tooltip	Ruft die Hilfe zum WinCC Wertetabelle auf.
	Konfigurationsdialog	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften der Wertetabelle verändern.

	Linealfenster	Über die Schaltfläche fragen Sie die Koordinatenpunkte einer Kurve ab. Die Kurvendaten werden im Linealfenster angezeigt.
	Statistikbereich festlegen	Ermöglicht die Festlegung eines Zeitbereichs, für den statistische Werte ermittelt werden.
	Statistik berechnen	Über die Schaltfläche werden im Statistikfenster die statistischen Werte angezeigt. Die angezeigten Werte beziehen sich auf eine ausgewählte Kurve mit dem projektierten Berechnungszeitraum. Die Taste ist nur bedienbar, wenn ein Statistikfenster einer f(t)-Kurvenanzeige verbunden ist.
	Drucken	Startet den Ausdruck der in der Tabelle angezeigten Werte. Den zu verwendeten Druckauftrag legen im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Druckauftrag" fest.
	Daten exportieren	Über die Schaltfläche exportieren Sie alle oder ausgewählte Runtime-Daten in eine "CSV"-Datei. Wenn im Inspektorfenster "Eigenschafte > Eigenschaften > Datenexport > Dialog zeigen" aktiviert ist, öffnet sich in Runtime ein Dialog. Im Dialog sehen Sie die Einstellungen zum Export und starten den Export. Mit den entsprechenden Berechtigungen wählen Sie auch die Datei und das Verzeichnis für den Export aus. Wenn kein Dialog angezeigt wird, startet sofort der Export der Daten in die voreingestellte Datei.
	Benutzerdefiniert 1	Zeigt die erste, vom Benutzer erstellte Tastenfunktion an. Die Funktion der Taste ist benutzerdefiniert.

Siehe auch

Tabellenanzeige (Seite 515)

f(t)-Kurvenanzeige (Seite 380)

f(x)-Kurvenanzeige (Seite 387)

Grundlagen zur Wertetabelle (Seite 754)

Wertetabelle projektieren (Seite 755)

Einstellung der Persistenz (Seite 2355)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.85 Wirkbereichs-Name

Verwendung

Das Objekt "Wirkbereichs-Name" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "Wirkbereichs-Name" zeigt den Namen des Wirkbereichs an, an dem sich das Bediengerät befindet und den Anmeldestatus.

Darstellung

Symbol	Bedeutung	Beschreibung	Anmeldezustand
MixingAxisControl	Zustimmtaster ist ohne Wirkung	Innerhalb des Wirkbereichs "MixingAxisControl"	Anmeldung am Wirkbereich ist möglich
MixingAxisControl	Zustimmtaster funktioniert	Innerhalb des Wirkbereichs "MixingAxisControl"	Am Wirkbereich angemeldet
MixingAxisControl	Zustimmtaster ist ohne Wirkung	Innerhalb des Wirkbereichs "MixingAxisControl"	Eine Anmeldung am Wirkbereich ist nicht möglich, weil ein anderes Bediengerät bereits an diesem Wirkbereich angemeldet ist. Hinweis: Bei Verwendung des Modus "Override": Eine Anmeldung am Wirkbereich ist nicht möglich, weil der Override-Schalter noch eingeschaltet ist, obwohl kein Bediengerät mehr im Wirkbereich angemeldet ist.
	Zustimmtaster ist ohne Wirkung	Außerhalb jedes Wirkbereichs	Hinweis: Die Anmeldung ist nur innerhalb des Wirkbereichs möglich

Im Inspektorfenster "Eigenschaften > Eigenschaften > Security" vergeben Sie eine Berechtigung, um sich in Runtime an einer Maschine anzumelden.

Voraussetzungen für das Anmelden

- Das Bediengerät befindet sich innerhalb des Wirkbereichs.
- Kein Bediengerät ist an diesem Wirkbereich angemeldet.

Vorgehen beim Anmelden

1. Klicken Sie auf das Objekt "Wirkbereichs-Name".
   Der Dialog "Wirkbereich-Anmeldung" wird geöffnet.
2. Lesen Sie die ID von der zu bedienenden Maschine oder Anlage ab.
3. Geben Sie die abgelesene ID ein.
4. Bestätigen Sie Ihre Angaben mit "OK".

Ergebnis

Wenn ID und Wirkbereich zusammenpassen, ist das Bediengerät am Wirkbereich angemeldet. Eine erfolgreiche Anmeldung wird durch ein grün hinterlegtes Symbol signalisiert. Kein weiteres Bediengerät kann sich an diesem Wirkbereich anmelden.

Voraussetzung für Abmelden

Das Bediengerät ist an diesem Wirkbereich angemeldet.

Vorgehen beim Abmelden

Bevor Sie den Wirkbereich verlassen oder Runtime beenden, melden Sie sich zuerst vom Wirkbereich ab. Wenn Sie den Wirkbereich ohne Abmelden verlassen, erscheint eine Warnung.

1. Klicken Sie auf das Objekt "Wirkbereichs-Name". Der Dialog "Wirkbereich-Abmeldung" wird geöffnet.
2. Bestätigen Sie das Abmelden mit "Ja".

Ergebnis

Das Bediengerät ist nicht mehr am Wirkbereich angemeldet. Andere Bediengeräte können sich an diesem Wirkbereich anmelden.

Hinweis

Wenn der Modus "Override" aktiviert war, deaktivieren Sie den Modus "Override" erst wieder, bevor Sie ein anderes Bediengerät anmelden.

Siehe auch

Wirkbereichs-Name projektieren (Seite 1863)

Wirkbereichs-Signal (Seite 529)

WLAN-Empfang (Seite 531)

Zonen-Name (Seite 534)

Zonen-Signal (Seite 535)

Ladezustand (Seite 424)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.86 Wirkbereichs-Name (RFID)

Verwendung

Das Objekt "Wirkbereichs-Name (RFID)" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "Wirkbereichs-Name (RFID)" zeigt den Namen des Wirkbereichs (RFID) an, in dem sich das Bediengerät befindet und den Anmeldestatus.

Hinweis

RFID Tags im Anlagenbereich

RFID Tags dürfen Sie nur in abgegrenzten Anlagenbereichen, die durch zusätzliche Schutzmaßnahmen z. B. Schutzzäune gesichert sind, anbringen.

Darstellung

Symbol	Bedeutung	Beschreibung	Anmeldezustand
Scan	Zustimmtaster ist ohne Wirkung	Innerhalb des Wirkbereichs (RFID) "MixingAxisControl"	Anmeldung am Wirkbereich (RFID) ist möglich
Scanning...	Zustimmtaster funktioniert	Innerhalb des Wirkbereichs (RFID) "MixingAxisControl"	Anmeldung am Wirkbereich (RFID) läuft
MixingAxisControl	Zustimmtaster funktioniert	Innerhalb des Wirkbereichs (RFID) "MixingAxisControl"	Am Wirkbereich (RFID) angemeldet

Im Inspektorfenster "Eigenschaften > Eigenschaften > Security" vergeben Sie eine Berechtigung für die Benutzergruppe, die sich in Runtime am Wirkbereich (RFID) anmeldet.

Voraussetzungen für das Anmelden

- Das Bediengerät befindet sich innerhalb des Wirkbereichs (RFID).
- Die Maschine ist durch zusätzliche Schutzmaßnahmen abgegrenzt.
- Kein Bediengerät ist an diesem Wirkbereich (RFID) angemeldet.

Vorgehen beim Anmelden

1. Klicken Sie auf das Objekt "Wirkbereichs-Name (RFID)".
   Der Dialog "Wirkbereich-Anmeldung" wird geöffnet.
2. Lesen Sie die ID von der zu bedienenden Maschine oder Anlage ab.
3. Geben Sie die abgelesene ID ein.
4. Bestätigen Sie Ihre Angaben mit "OK".

Ergebnis

Wenn ID und Wirkbereich (RFID) zusammenpassen, ist das Bediengerät an der Maschine angemeldet. Eine erfolgreiche Anmeldung wird durch ein grün hinterlegtes Symbol signalisiert. Kein weiteres Bediengerät kann sich an diesem Wirkbereich (RFID) bzw. der Maschine anmelden.

Voraussetzung für das Abmelden

Das Bediengerät ist an diesem Wirkbereich (RFID) angemeldet.

Vorgehen beim Abmelden

Bevor Sie den Wirkbereich (RFID) verlassen oder Runtime beenden, melden Sie sich zuerst vom Wirkbereich (RFID) ab. Wenn Sie den Wirkbereich (RFID) ohne Abmelden verlassen, wird ein lokaler Rampdown eingeleitet.

1. Klicken Sie auf das Objekt "Wirkbereichs-Name (RFID) ". Der Dialog "Wirkbereich (RFID)-Abmeldung" wird geöffnet.
2. Bestätigen Sie das Abmelden mit "Ja".

Ergebnis

Das Bediengerät ist nicht mehr am Wirkbereich (RFID) bzw. Maschine angemeldet. Andere Bediengeräte können sich an diesem Wirkbereich (RFID) anmelden.

Siehe auch

WLAN-Empfang (Seite 531)

Ladezustand (Seite 424)

Wirkbereichs-Name projektieren (RFID) (Seite 1867)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.87 Wirkbereichs-Signal

Verwendung

Das Objekt "Wirkbereichs-Signal" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "Wirkbereichs-Signal" zeigt an, wie gut sich das Mobile Panel Wireless in einem Wirkbereich befindet.

Darstellung

Symbol	Bedeutung
	Innerhalb eines Wirkbereichs
	Beim Austritt aus einem Wirkbereich Beim Eintritt in einen Wirkbereich
	Außerhalb jedes Wirkbereichs
	Overridefunktion ist aktiviert: Innerhalb eines gesonderten Schutzbereichs des Wirkbereichs

Um welchen Wirkbereich es sich handelt, zeigt das Objekt "Wirkbereichs-Name".

Overridefunktion

Ein Wirkbereich z. B. "Roboter" wird nach außen durch Transponder geformt. Jedoch innerhalb eines gesonderten Schutzbereichs z. B. "Roboterzelle" wird der Wirkbereich nicht durch Transponder, sondern durch alternative Begrenzungen geformt z. B. durch Gitter und eine Tür.

Der Bediener meldet sich zunächst am Wirkbereich an. Wenn der Bediener den abgegrenzten und gesicherten Schutzbereich "Roboterzelle" betritt, muss er einen zusätzlichen Schalter betätigen, der die Overridefunktion aktiviert:

- Der Wirkbereich ist fest zugeordnet und nicht mehr durch die Transponder festgelegt.
- Daher spielt die Qualität keine Rolle und wird nicht mehr angezeigt.
- Das Bediengerät bleibt weiter am Wirkbereich angemeldet.
- Es kann sich kein weiteres Bediengerät mehr an diesem Wirkbereich anmelden.

Berechnen der Signalqualität

Die Qualität innerhalb eines Wirkbereichs hängt wie folgt von der gemessenen Entfernung ab:

- In der Mitte des Wirkbereichs ist die Qualität 100%.
- Am Transponder und an der Grenze des Wirkbereichs ist die Qualität 0%.

Bedienung

Das Objekt dient zur reinen Anzeige und kann nicht bedient werden.

Siehe auch

Wirkbereichs-Name (Seite 525)

WLAN-Empfang (Seite 531)

Zonen-Name (Seite 534)

Zonen-Signal (Seite 535)

Ladezustand (Seite 424)

Weitere Objekte der Mobile Wireless projektieren (Seite 1864)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.88 WLAN-Empfang

Verwendung

Das Objekt "WLAN-Empfang" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "WLAN-Empfang" zeigt an, wie gut die WLAN-Funkverbindung ist. Dazu wird die Signalstärke gemessen und anhand von 5 Balken angezeigt.

Darstellung

Symbol	Bedeutung	Signalstärke
	Keine Funkverbindung	Kein Signal
	Sehr schlechte Funkverbindung	=<20 %
	Schlechte Funkverbindung	>20 % =<40 %
	Funkverbindung in Ordnung	>40 % =<60 %
	Gute Funkverbindung	>60 % =<80 %
	Sehr gute Funkverbindung	>80 %

Bedienung

Das Objekt dient zur reinen Anzeige und kann nicht bedient werden.

Siehe auch

Wirkbereichs-Name (Seite 525)

Wirkbereichs-Name (RFID) (Seite 528)

Wirkbereichs-Signal (Seite 529)

Zonen-Name (Seite 534)

Zonen-Signal (Seite 535)

Ladezustand (Seite 424)

Weitere Objekte der Mobile Wireless projektieren (Seite 1864)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.89 Zeigerinstrument

Verwendung

Das Objekt "Zeigerinstrument" stellt numerische Werte in Form eines analogen Zeigerinstruments dar. In Runtime ist auf einen Blick erkennbar, ob z. B. der Kesseldruck im Normalbereich liegt.

Hinweis

Das Zeigerinstrument dient zur reinen Anzeige und kann nicht bedient werden.

Projektieren Sie Systemfunktionen, die bei Wertänderung ausgeführt werden, in den Eigenschaften der Variablen an das Ereignis "Wertänderung" und nicht an das Ereignis "Ändern" des Zeigerinstruments. Wenn ihr Projekt den Richtlinien nach GMP (Good Manufacturing Practice) entsprechen soll und die Systemfunktion eine GMP-relevante Aktion ausführt, z. B. Erhöhen einer GMP-relevanten Variable, wird jede Wertänderung der am Zeigerinstrument verwendeten Variablen als Benutzeraktion interpretiert.

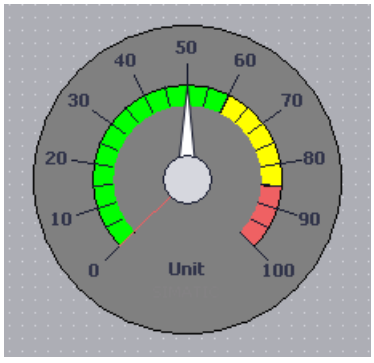

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Spitzenwert anzeigen: Legt fest, ob der reale Messbereich mit einem Schleppzeiger gekennzeichnet wird.
- Maximaler Wert und Minimaler Wert: Legt den oberen und den unteren Endwert der Skala fest.
- Anfangswert Gefahrenbereich und Anfangswert Warnbereich: Legt fest, ab welchem Skalenwert der Gefahrenbereich und der Warnbereich beginnen.

- Normalbereich anzeigen: Legt fest, ob der Normalbereich farbig auf der Skala eingeblendet wird.
- Farbe der einzelnen Bereiche: Um verschiedene Betriebszustände zu unterscheiden, werden diese farblich unterschiedlich dargestellt, z. B. Normalbereich, Warnbereich und Gefahrenbereich.

Hinweis

Die Verwendung von vielen unterschiedlich großen Objekten „Zeigerinstrument“ kann zur Verringerung der Performance in Runtime führen. Vermeiden Sie beim „Zeigerinstrument“ die Verwendung von minimal unterschiendlichen Höhen und Breiten, z. B. 48 Pixel, 49 Pixel, 51 Pixel, usw. Verwenden Sie stattdessen gleiche Größen.

Spitzenwert anzeigen

Mit der Eigenschaft "Spitzenwert anzeigen" können Sie für den maximalen und für den minimalen Zeigerausschlag in Runtime eine Merkfunktion aktivieren. Der reale Messbereich wird mit einem Schleppzeiger gekennzeichnet.

1. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Spitzenwert anzeigen".

Maximaler Wert und Minimaler Wert

Sie legen den oberen und den unteren Endwert der Skala im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie unter "Prozess > Maximaler Skalenwert" und bei "Minimaler Skalenwert" eine Zahl ein.
   Wenn Sie als Endwert der Skala eine Variable auswählen, dann ist die Zahl nicht mehr verfügbar.

Hinweis

Beachten Sie, insbesondere wenn Sie Minmalwert oder Maximalwert über eine Variable dynamisieren, dass der Minimalwert immer kleiner ist als der Maximalwert. Wenn der Minimalwert größer ist als der Maximalwert, erfolgt keine Anpassung der Skalenendwerte.

Anfangswert Gefahrenbereich und Anfangswert Warnbereich

Ab welchem Skalenwert der Gefahrenbereich und der Warnbereich beginnen, wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Grenzen/Bereiche".
2. Geben Sie bei "Oben 2" den Anfangswert für den Gefahrenbereich und bei "Oben 1" den Anfangswert für den Warnbereich ein.
3. Aktivieren Sie "Anfangswert Gefahrenbereich" und "Anfangswert Warnbereich", damit die Bereiche in der Skala angezeigt werden.

Normalbereich anzeigen

Im Inspektorfenster legen Sie fest, ob der Normalbereich farbig auf der Skala eingeblendet wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Grenzen/Bereiche".
2. Aktivieren Sie "Normal".

Farbe der einzelnen Bereiche

Sie können der Normalbereich, den Gefahrenbereich und den Warnbereich farblich unterschiedlich darstellen. Die Farbe legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster in der Gruppe "Eigenschaften > Eigenschaften > Grenzen/Bereiche".
2. Wählen Sie für die drei Bereiche eine Farbe der Farbpalette aus.

Hinweis

Bei Bediengeräten mit Windows CE bis Version 5.0 gibt es folgende Einschränkungen:

- Die drei Bereiche (Normalbereich, Warnbereich, Gefahrenbereich) werden in Runtime nicht farblich unterschiedlich angezeigt.

Hinweis

Wenn Sie die Option "Bereiche von Variablen anzeigen" aktivieren, können Sie im Zeigerinstrument bis zu fünf Bereiche darstellen, deren Werte über eine Prozessvariable vorgegeben werden. Die Werte für die Bereiche definieren Sie an einer Prozessvariable, die Sie mit dem Bildobjekt verschalten.

Die Option "Bereiche von Variablen anzeigen" steht für Comfort Panels, KTP Mobile Panels und RT Advanced zur Verfügung.

Siehe auch

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.90 Zonen-Name

Verwendung

Das Objekt "Zonen-Name" ist ein Bedienelementfür für einige SIMATIC Mobile Panel. Das Objekt "Zonen-Name" zeigt den Namen der Zone an, in dem sich das Bediengerät befindet.

Darstellung

Symbol	Bedeutung	Beschreibung
MixingPlant	Die Ereignisse "Beim Eintritt" und "Beim Austritt" der Zone und die Zonen-ID können Sie in der Projektierung verwenden.	Innerhalb der Zone "MixingPlant"
	–	Außerhalb jeder Zone

Vorgehen mit Ereignis "Beim Eintritt"

1. Projektieren Sie im Editor "Zonen" bzw. "Zonen und Wirkbereiche" an das Ereignis "Beim Eintritt" die Funktion "AktiviereBild".

Vorgehen mit Zonen-ID

1. Wählen Sie im Editor "Runtime-Êinstellungen > Allgemein" als "Zonen-ID / Anschlusspunkt-ID" eine interne Variable.
2. Animieren Sie die Sichtbarkeit eines Objekts z. B. "E/A-Feld" über die interne Variable.

Bedienung

Das Objekt dient zur reinen Anzeige und wird nicht bedient.

Siehe auch

Wirkbereichs-Name (Seite 525)

Wirkbereichs-Signal (Seite 529)

WLAN-Empfang (Seite 531)

Zonen-Signal (Seite 535)

Ladezustand (Seite 424)

Geräteabhängigkeit der Objekte (Seite 337)

1.9.2.91 Zonen-Signal

Verwendung

Das Objekt "Zonen-Signal" ist ein Bedienelement für einige SIMATIC Mobile Panel. Das Objekt "Zonen-Signal" zeigt an, wie gut sich das Mobile Panel noch in einer Zone befindet. Im Gegensatz zum Objekt "WLAN-Empfang" wird nicht die Signalstärke gemessen, sondern aus der Entfernung berechnet.

Darstellung

Symbol	Bedeutung	Qualität
	Innerhalb einer Zone	>=15 %
	Beim Austritt aus einer Zone Beim Eintritt in eine Zone	<15 % und >0 %
	Außerhalb jeder Zone	=0 %

Um welche Zone es sich handelt, zeigt das Objekt "Zonen-Name".

Berechnung der Signalqualität

Die Qualität des Signals innerhalb einer Zone hängt wie folgt von der gemessenen Entfernung ab:

- In der Mitte der Zone ist die Qualität 100%.
- Nach hinten zum Transponder und nach vorn weg vom Transponder nimmt die Qualität jeweils linear ab.
- Am Transponder und an der Grenze der Zone ist die Qualität 0%.

Bedienung

Das Objekt dient zur reinen Anzeige und wird nicht bedient.

Siehe auch

Wirkbereichs-Name (Seite 525)

Wirkbereichs-Signal (Seite 529)

WLAN-Empfang (Seite 531)

Zonen-Name (Seite 534)

Ladezustand (Seite 424)

Geräteabhängigkeit der Objekte (Seite 337)

Mit Variablen arbeiten 2

2.1 Grundlagen

2.1.1 Grundlagen zu Variablen

Einleitung

Prozesswerte werden in Runtime durch Variablen weitergegeben. Prozesswerte sind Daten, die im Speicher eines der angeschlossenen Automatisierungssysteme abgelegt werden. Sie stellen den Zustand einer Anlage dar, z. B. als Temperaturen, Füllstände oder Schaltzustände. Für die Verarbeitung der Prozesswerte in WinCC definieren Sie externe Variablen.

WinCC arbeitet mit zwei Arten von Variablen:

- Externe Variablen
- Interne Variablen

Die externen Variablen sind das Bindeglied zwischen WinCC und den Automatisierungssystemen. Die Werte von externen Variablen entsprechen den Prozesswerten aus dem Speicher eines Automatisierungssystems. Der Wert einer externen Variable wird ermittelt, indem der Prozesswert aus dem Speicher des Automatisierungssystems ausgelesen wird. Umgekehrt können Sie einen Prozesswert auch in den Speicher des Automatisierungssystems zurückschreiben.

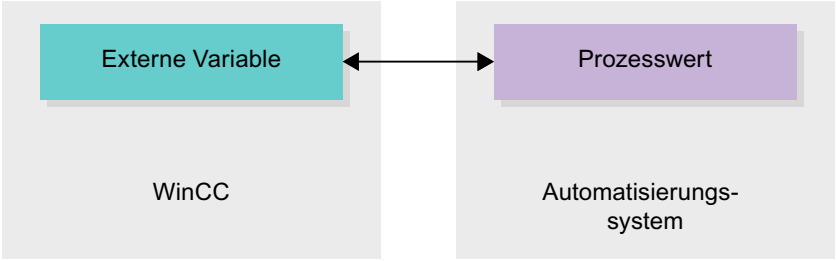

Interne Variablen besitzen keine Prozessanbindung und transportieren nur Werte innerhalb von WinCC. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

Variablen in WinCC

Bei externen Variablen wird in den Eigenschaften der jeweiligen Variable festgelegt, über welche Verbindung WinCC mit dem Automatisierungssystem kommuniziert und wie der Datenaustausch erfolgt.

Variablen ohne Werteversorgung durch den Prozess, die so genannten internen Variablen, werden nicht mit dem Automatisierungssystem verbunden. In der Eigenschaft "Verbindung" der Variable erkennen Sie dies am Eintrag "Interne Variable".

Zur besseren Übersicht legen Sie die Variablen in verschiedenen Variablentabellen an. In der Projektnavigation greifen Sie dann im Knoten "HMI-Variablen" direkt auf die einzelnen Variablentabellen zu. Mithilfe der Tabelle "Zeige alle Variablen" lassen Sie sich die Variablen aus allen Variablentabellen anzeigen.

Mit Strukturen fassen Sie eine Anzahl verschiedener Variablen zusammen, die eine logische Einheit bilden. Strukturen sind projektzugehörige Daten und sind für alle HMI-Geräte des Projekts verfügbar. Für die Erstellung und Bearbeitung einer Struktur verwenden Sie den Editor "Typen" in der Projektbibliothek.

Siehe auch

Interne Variablen (Seite 547)

Externe Variablen (Seite 542)

Grundlagen zu Zyklen (Seite 616)

Grundlagen zu Anwenderdatentypen (Seite 607)

Grundlagen zu Arrays (Seite 597)

Übersicht über HMI-Variablentabellen (Seite 538)

Adressierung externer Variablen (Seite 545)

Externe Variable anlegen (Seite 550)

Grundlagen zu Zyklen (Seite 619)

Grundlagen zur Variablenarchivierung (Seite 639)

Externe Variablen (Seite 540)

Interne Variablen (Seite 548)

Ausgeben von Variablenwerten in Bildern (Basic) (Seite 704)

2.1.2 Übersicht über HMI-Variablentabellen

Einleitung

HMI-Variablentabellen enthalten die Definitionen der geräteweit gültigen HMI-Variablen. Für jedes im Projekt angelegte HMI-Gerät wird automatisch eine Variablentabelle angelegt.

In der Projektnavigation gibt es für jedes HMI-Gerät einen Ordner "HMI-Variablen". Darin können folgende Tabellen enthalten sein:

- Standard-Variablentabelle
- Anwenderdefinierte Variablentabellen
- Tabelle aller Variablen

In der Projektnavigation können Sie im Ordner HMI-Variablen weitere Variablentabellen anlegen, um Variablen und Konstanten darin zu sortieren und zu gruppieren. Die Variablen können Sie per Drag&Drop oder mithilfe des Felds "Variablentabelle" in eine andere

Variablentabelle verschieben. Das Feld "Variablentabelle" aktivieren Sie über das Kontextmenü der Spaltenüberschriften.

Standard-Variablentabelle

Für jedes HMI-Gerät des Projekts gibt es eine Standard-Variablentabelle. Sie kann nicht gelöscht oder verschoben werden. Die Standard-Variablentabelle enthält HMI-Variablen und abhängig vom HMI-Gerät auch Systemvariablen. Sie können alle HMI-Variablen in der Standard-Variablentabelle deklarieren, oder nach Bedarf in weiteren, anwenderdefinierten Variablentabellen.

Anwenderdefinierte Variablentabellen

Sie können mehrere anwenderdefinierte Variablentabellen für jedes HMI-Gerät anlegen, um Variablen nach Ihren Bedürfnissen zu gruppieren. Anwenderdefinierte Variablentabellen können Sie umbenennen, in Gruppen zusammenfassen oder löschen. Zur Gruppierung von Variablentabellen legen Sie im Ordner HMI-Variablen weitere Unterordner an.

Alle Variablen

Die Tabelle "Alle Variablen" zeigt eine Übersicht über alle HMI-Variablen und Systemvariablen des jeweiligen HMI-Geräts. Diese Tabelle kann nicht gelöscht, umbenannt oder verschoben werden. Diese Tabelle enthält zusätzlich die Spalte "Variablentabelle, die anzeigt, in welcher Variablentabelle eine Variable enthalten ist. Über das Feld "Variablentabelle" können Sie die Zuordnung einer Variablen zu einer Variablentabelle ändern.

Bei Geräten für Runtime Professional enthält die Tabelle "Alle Variablen" eine weitere Registerkarte "Systemvariablen". Die Systemvariablen werden vom System angelegt und für die interne Verwaltung des Projekts benutzt. Die Namen der Systemvariablen beginnen mit dem Zeichen "@". Systemvariablen dürfen Sie weder löschen noch umbenennen. Den Wert einer Systemvariablen können Sie auswerten, jedoch nicht ändern.

Zusätzliche Tabellen

In einer HMI-Variablentabelle stehen Ihnen zusätzlich folgende Tabellen zur Verfügung:

- Bitmeldungen
- Analogmeldungen
- Archivvariablen

Mithilfe dieser Tabellen projektieren Sie Meldungen und Archivvariablen für die aktuell selektierte HMI-Variable.

Tabelle Bitmeldungen

In der Tabelle "Bitmeldungen" projektieren Sie Bitmeldungen an die in der HMI-Variablentabelle selektierte HMI-Variable. Wenn Sie eine Bitmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablentabelle nicht möglich. Projektieren Sie die Bitmeldungen für jede HMI-Variable separat.

Tabelle Analogmeldungen

In der Tabelle "Analogmeldungen" projektieren Sie Analogmeldungen an die in der HMI-Variablentabelle selektierte HMI-Variable. Wenn Sie eine Analogmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablentabelle nicht möglich. Projektieren Sie die Analogmeldungen für jede HMI-Variable separat.

Tabelle Archivvariablen

In der Tabelle " Archivvariablen" projektieren Sie Archivvariablen an die in der HMI-Variablentabelle selektierte HMI-Variable. Wenn Sie eine Archivvariable projektieren, ist eine Mehrfachselektion in der HMI-Variablentabelle nicht möglich. Projektieren Sie die Archivvariablen für jede HMI-Variable separat. Die Tabelle "Archivvariablen" ist nur verfügbar, wenn das verwendete HMI-Gerät die Archivierung unterstützt.
Bei Verwendung von WinCC Runtime Professional können Sie einer Variablen auch mehrere Archivvariablen zuweisen. Bei den übrigen HMI-Geräten können Sie einer Variablen lediglich eine Archivvariable zuweisen.

Siehe auch

Grundlagen zu Variablen (Seite 537)

2.1.3 Externe Variablen

Einleitung

Externe Variablen ermöglichen die Kommunikation, also den Datenaustausch, zwischen den Komponenten eines Automatisierungssystems, z. B. zwischen Bediengerät und Steuerung.

Prinzip

Eine externe Variable ist das Abbild eines definierten Speicherplatzes in der Steuerung. Auf diesen Speicherplatz können Sie sowohl vom Bediengerät als auch von der Steuerung lesend und schreibend zugreifen.

Da externe Variablen das Abbild eines Speicherplatzes in der Steuerung sind, hängen die anwendbaren Datentypen von der Steuerung ab, mit der das Bediengerät verbunden ist.

Wenn Sie in STEP 7 ein PLC-Steuerungsprogramm schreiben, werden die im Steuerungsprogramm erstellten PLC-Variablen in die PLC-Variablentabelle eingetragen. Wenn Sie eine externe Variable mit einer PLC-Variable verbinden wollen, greifen Sie über die PLC-Variablentabelle direkt auf die PLC-Variablen zu und verbinden sie mit der externen Variable.

Datentypen

An einer externen Variable stehen Ihnen in WinCC alle Datentypen zur Verfügung, die an der verbundenen Steuerung verfügbar sind. Informationen über Datentypen, die bei einer Verbindung zu anderen Steuerungen zur Verfügung stehen, finden Sie in der Dokumentation zu den entsprechenden Kommunikationstreibern.

Nähere Hinweise hierzu finden Sie unter "AUTOHOTSPOT".

Hinweis

Für die Kommunikation zwischen Bediengerät und Steuerung stehen Ihnen neben den externen Variablen auch Bereichszeiger zur Verfügung. Die Bereichszeiger können Sie im Editor "Verbindungen" einrichten und aktivieren.

Zentrale Variablenverwaltung in STEP 7

Sie können mit den HMI-Variablen auch DB-Instanzen von benutzerdefinierten PLC-Datentypen (UDT) verbinden.

Der PLC-Datentyp und die zugehörigen DB-Instanzen werden zentral in STEP 7 angelegt und aktualisiert. In WinCC können Sie als PLC-Variable (DB-Instanzen) folgende Quellen verwenden:

- Datenbaustein-Elemente, die einen UDT als Datentyp verwenden
- Datenbausteininstanzen von einem UDT

Der Datentyp wird aus STEP 7 übernommen und nicht auf einen HMI-Datentyp umgewandelt. Die Zugriffsart ist immer "Symbolischer Zugriff". Abhängig von der Freigabe für WinCC in STEP 7 werden auch Elemente und strukturierte Elemente des PLC-Datentyps nach WinCC übernommen. Elemente eines strukturierten UDT werden übernommen und in der PLC-Variablentabelle angezeigt, wenn die instanzspezifischen Eigenschaften "Sichtbar in HMI" und "Erreichbar aus HMI" gesetzt wurden.

Hinweis

Zugriff auf PLC-Datentypen

Der Zugriff auf PLC-Datentypen steht Ihnen nur in Verbindung mit SIMATIC S7-1200 oder S7-1500 zur Verfügung.

Synchronisierung mit PLC-Variablen

In den Runtime-Einstellungen stehen Ihnen unter "Einstellungen für Variablen" verschiedene Optionen zum Synchronisieren von externen Variablen mit den PLC-Variablen zur Verfügung.

Beim Synchronisieren haben Sie die Möglichkeit, die Variablennamen der PLC in externen Variablen automatisch zu übernehmen und bestehende Variablen neu zu verbinden.

Der generierte Variablenname leitet sich aus der Position des Datenwerts in der hierarchischen Struktur des Datenbausteins ab.

Aktualisierung von Variablenwerten

Bei externen Variablen werden in Runtime die aktuellen Variablenwerte über die Kommunikationsverbindung zwischen WinCC und den verbundenen Automatisierungssystemen übertragen und im Speicher der Runtime abgelegt. Danach wird der Variablenwert mit der eingestellten Zykluszeit aktualisiert. Für die Verwendung im Runtime-Projekt greift WinCC auf Variablenwerte im Speicher der Runtime zu, die zum vorherigen Zykluszeitpunkt aus der Steuerung gelesen wurden. Dadurch kann sich der Wert in der Steuerung bereits ändern, während der Wert aus dem Speicher der Runtime verarbeitet wird.

Hinweis

PLC-Arrayelemente in Verbindung mit S7-1200 oder S7-1500

Der Index von PLC-Arrayelementen kann mit einer beliebigen Zahl beginnen. In WinCC beginnt die Indexierung immer mit 0.

Eine PLC-Variable "Array[1..3] of Int" wird in WinCC z. B. mit "Array[0..2] of Int" abgebildet.

Wenn Sie in einem Skript auf ein Array zugreifen, achten Sie auf die korrekte Indexierung.

Siehe auch

Grundlagen zu Variablen (Seite 537)

2.1.4 Externe Variablen

Einleitung

Externe Variablen ermöglichen die Kommunikation, also den Datenaustausch, zwischen den Komponenten eines Automatisierungssystems, z. B. zwischen Bediengerät und Steuerung.

Prinzip

Eine externe Variable ist das Abbild eines definierten Speicherplatzes in der Steuerung. Auf diesen Speicherplatz können Sie sowohl vom Bediengerät als auch von der Steuerung lesend und schreibend zugreifen.

Da externe Variablen das Abbild eines Speicherplatzes in der Steuerung sind, hängen die anwendbaren Datentypen von der Steuerung ab, mit der das Bediengerät verbunden ist.

Wenn Sie in STEP 7 ein PLC-Steuerungsprogramm schreiben, werden die im Steuerungsprogramm erstellten PLC-Variablen in die PLC-Variablentabelle eingetragen. Wenn Sie eine externe Variable mit einer PLC-Variable verbinden wollen, greifen Sie über die PLC-Variablentabelle direkt auf die PLC-Variablen zu und verbinden sie mit der externen Variable.

Datentypen

An einer externen Variable stehen Ihnen in WinCC alle Datentypen zur Verfügung, die an der verbundenen Steuerung verfügbar sind. Wenn der Datentyp der PLC-Variable in WinCC nicht verfügbar ist, wird an der HMI-Variable automatisch ein kompatibler Datentyp verwendet. Wenn Sie PLC-Datentypen verwenden, wird der Datentyp nach WinCC übernommen. Bei Bedarf können Sie den Datentyp an der HMI-Variable ändern.

Zentrale Variablenverwaltung in STEP 7

Sie können mit den HMI-Variablen auch DB-Instanzen von benutzerdefinierten PLC-Datentypen (UDT) verbinden.

Der PLC-Datentyp und die zugehörigen DB-Instanzen werden zentral in STEP 7 angelegt und aktualisiert. In WinCC können Sie als PLC-Variable (DB-Instanzen) folgende Quellen verwenden::

- Datenbaustein-Elemente, die einen UDT als Datentyp verwenden
- Datenbausteininstanzen von einem UDT

Der Datentyp wird aus STEP 7 übernommen und nicht auf einen HMI-Datentyp umgewandelt. Die Zugriffsart ist immer "Symbolischer Zugriff". Abhängig von der Freigabe für WinCC in STEP 7 werden auch Elemente und strukturierte

Elemente des PLC-Datentyps nach WinCC übernommen.

Hinweis

Zugriff auf PLC-Datentypen

Der Zugriff auf PLC-Datentypen steht Ihnen nur in Verbindung mit SIMATIC S7-1200 oder S7-1500 zur Verfügung.

Synchronisierung mit PLC-Variablen

In den Runtime-Einstellungen stehen Ihnen unter "Einstellungen für Variablen" verschiedeneOptionen zum Synchronisieren von externen Variablen mit den PLC-Variablen zur Verfügung.

Beim Synchronisieren haben Sie die Möglichkeit, die Variablennamen der PLC in externenVariablen automatisch zu übernehmen und bestehende Variablen neu zu verbinden.

Der generierte Variablenname leitet sich aus der Position des Datenwerts in der hierarchischenStruktur des Datenbausteins ab.

Aktualisierung von Variablenwerten

Bei externen Variablen werden in Runtime die aktuellen Variablenwerte über die Kommunikationsverbindung zwischen WinCC und den verbundenen Automatisierungssystemen übertragen und im Speicher der Runtime abgelegt. Danach wird der Variablenwert mit der eingestellten Zykluszeit aktualisiert. Für die Verwendung im Runtime-Projekt greift WinCC auf Variablenwerte im Speicher der Runtime zu, die zum vorherigen Zykluszeitpunkt aus der Steuerung gelesen wurden. Dadurch kann sich der Wert in der Steuerung bereits ändern, während der Wert aus dem Speicher der Runtime verarbeitet wird.

Bei Bedarf können Sie Prozesswerte auch direkt aus dem Speicher der Steuerung auslesen. Dafür stehen Ihnen die GetTag-Funktionen zur Verfügung. Nähere Hinweise finden Sie in der Dokumentation zu C-Scripting unter "AUTOHOTSPOT".

Hinweis

PLC-Arrayelemente in Verbindung mit S7-1200 oder S7-1500

Der Index von PLC-Arrayelementen kann mit einer beliebigen Zahl beginnen. In WinCC beginnt die Indexierung immer mit 0.

Eine PLC-Variable "Array[1..3] of Int" wird in WinCC z. B. mit "Array[0..2] of Int" abgebildet.

Wenn Sie in einem Skript auf ein Array zugreifen, achten Sie auf die korrekte Indexierung.

Anzeigen des Kommentars der PLC-Variable

Wenn Sie den Kommentar der verbundenen PLC-Variable sehen wollen, aktivieren Sie in der HMI-Variablentabelle die Spalte "Quellkommentar". Der Quellkommentar ist in Runtime nicht verfügbar.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Datentyp einer Variable anpassen (Seite 556)

Variablen synchronisieren (Seite 566)

Einstellungen zum Synchronisieren (Seite 564)

Benutzerdefinierte PLC-Datentypen (UDT) (Seite 634)

2.1.5 Adressierung externer Variablen

Einleitung

Die Möglichkeiten zur Adressierung externer Variablen sind abhängig von der Verbindungsart zwischen WinCC und der jeweiligen Steuerung. Folgende Verbindungsarten müssen Sie unterscheiden:

- Integrierte Verbindung
  Verbindungen von Geräten, die sich innerhalb eines Projekts befinden und mit dem Editor "Geräte & Netze" erstellt wurden, werden als integrierte Verbindungen bezeichnet.
- Nicht integrierte Verbindung
  Verbindungen von Geräten, die mit dem Editor "Verbindungen" erstellt wurden, werden als nicht integrierte Verbindungen bezeichnet. Die Geräte müssen sich nicht alle innerhalb eines Projekts befinden.

Die Art der Verbindung erkennen Sie auch an ihrem Symbol.

	Integrierte Verbindung
	Nicht integrierte Verbindung

Nähere Hinweise hierzu finden Sie im Kapitel AUTOHOTSPOT.

Adressierung bei integrierter Verbindung

Eine integrierte Verbindung bietet den Vorteil, dass Sie eine Variable sowohl symbolisch als auch absolut adressieren können.

Bei der symbolischen Adressierung wählen Sie die PLC-Variable über ihren Namen aus und verbinden Sie mit der HMI-Variable. Der gültige Datentyp für die HMI-Variable wird vom System automatisch gewählt. Bei der Adressierung von Elementen in Datenbausteinen müssen Sie folgende Fälle unterscheiden:

Symbolische Adressierung von Datenbausteinen mit optimiertem Zugriff und mit Standardzugriff:

Bei der symbolischen Adressierung eines Datenbausteins mit optimiertem Zugriff und mit Standardzugriff wird die Adresse eines Elements im Datenbaustein dynamisch vergeben und bei einer Änderung automatisch in die HMI-Variable übernommen. Ein Übersetzen des verbundenen Datenbausteins oder des WinCC-Projekts ist dafür nicht notwendig.
Bei Datenbausteinen mit optimiertem Zugriff steht Ihnen nur die symbolische Adressierung zur Verfügung.

Bei der symbolischen Adressierung von Elementen eines Datenbausteins muss das WinCC-Projekt lediglich bei folgenden Änderungen neu übersetzt und geladen werden:

- Wenn sich der Name oder der Datentyp des verbundenen Datenbaustein-Elements oder der globalen PLC-Variable geändert hat.
- Wenn sich der Name oder der Datentyp eines im Datenbaustein-Element oder der globalen PLC-Variable übergeordneten Strukturknotens des verbundenen Elements geändert hat.
- Wenn sich die Nummer des verbundenen Datenbausteins geändert hat.

Die symbolische Adressierung steht Ihnen derzeit mit folgenden Steuerungen zur Verfügung:

- SIMATIC S7-1200
- SIMATIC S7-1500
- SIMATIC ET 200 CPU
- SIMATIC S7-1500 Software Controller

Die symbolische Adressierung steht Ihnen außerdem bei einer integrierten Verbindung zur Verfügung.

Die absolute Adressierung können Sie auch bei einer integrierten Verbindung verwenden. PLC-Variablen aus einer SIMATIC S7-300/400 Steuerung können Sie nur absolut adressieren. Wenn Sie eine HMI-Variable mit einer PLC-Variable verbunden haben, und sich die Adresse der PLC-Variablen ändert, müssen Sie das Steuerungsprogramm neu übersetzen, damit die neue Adresse in WinCC aktualisiert wird. Übersetzen Sie anschließend auch das WinCC-Projekt neu und laden Sie es auf das Bediengerät.

In WinCC ist die symbolische Adressierung als Standard voreingestellt. Um die Standardeinstellung zu ändern, wählen Sie den Menübefehl "Extras > Einstellungen". Im Dialog "Einstellungen" wählen Sie "Visualisierung > Variablen". Deaktivieren Sie bei Bedarf die Option "Symbolischer Zugriff".

Die Verfügbarkeit einer integrierten Verbindung ist abhängig von der verwendeten Steuerung. Die folgende Tabelle zeigt die Verfügbarkeit:

Steuerung	Integrierte Verbindung	Bemerkungen
S7-300/400	Ja	Die Verknüpfung der Variablen wird in Runtime nicht geprüft. Wenn sich in der Steuerung die Variablenadresse ändert, und das HMI-Gerät nicht neu übersetzt und geladen wurde, wird die Änderung in Runtime nicht registriert.
S7-1200	Ja	Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7-300/400 beschriebene Verhalten.
S7-1500	Ja	Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7-300/400 beschriebene Verhalten.

Steuerung	Integrierte Verbindung	Bemerkungen
SIMATIC ET 200 CPU	Ja	Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7-300/400 beschriebene Verhalten.
SIMATIC S7-1500 Software Controller	Ja	Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7-300/400 beschriebene Verhalten.

Eine integrierte Verbindung erstellen Sie im Editor "Geräte & Netze". Wenn die Steuerung im Projekt enthalten ist und integrierte Verbindungen unterstützt, dann können Sie sich die Verbindung auch automatisch erstellen lassen. Wählen Sie dazu bei der Projektierung der HMI-Variable einfach eine bestehende PLC-Variable aus, mit der Sie die HMI-Variable verbinden wollen. Die integrierte Verbindung wird dann vom System automatisch erstellt.

Adressierung bei nicht integrierter Verbindung

Bei einem Projekt mit nicht integrierter Verbindung projektieren Sie eine Variablenverbindung immer mit absoluter Adressierung. Den gültigen Datentyp müssen Sie selbst wählen. Wenn sich bei einem Projekt mit nicht integrierter Verbindung im Projektverlauf die Adresse einer PLC-Variable ändert, müssen Sie die Änderung in WinCC ebenfalls ändern. Die Variablenverbindung kann in Runtime nicht auf Gültigkeit geprüft werden, es erfolgt keine Fehlermeldung.

Eine nicht integrierte Verbindung ist für alle unterstützten Steuerungen verfügbar.

Die symbolische Adressierung ist bei nicht integrierter Verbindung nicht verfügbar.

Bei einer nicht integrierten Verbindung muss das Steuerungsprogramm nicht Bestandteil des WinCC Projekts sein. Die Projektierung der Steuerung und des WinCC-Projekts können Sie unabhängig voneinander durchführen. Für die Projektierung in WinCC müssen lediglich die verwendeten Adressen in der Steuerung und deren Funktion bekannt sein.

Siehe auch

Grundlagen zu Variablen (Seite 537)

2.1.6 Interne Variablen

Einleitung

Interne Variablen haben keine Verbindung zur Steuerung. Interne Variablen transportieren Werte innerhalb des Bediengeräts. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

Prinzip

Interne Variablen werden im Speicher des Bediengeräts abgelegt. Daher kann nur dieses Bediengerät lesend und schreibend auf die internen Variablen zugreifen. Interne Variablen legen Sie z. B. an, um lokale Berechnungen durchzuführen.

Für interne Variablen stehen Ihnen die HMI-Datentypen zur Verfügung. Die Verfügbarkeit ist abhängig vom verwendeten Bediengerät.

Folgende HMI-Datentypen sind verfügbar:

HMI-Datentyp	Datenformat
Array	eindimensionales Array
Bool	Binäre Variable
DateTime	Datum/ Uhrzeit Format
DInt	Vorzeichenbehafteter 32-Bit Wert
Int	Vorzeichenbehafteter 16-Bit Wert
LReal	Gleitkommazahl 64-Bit IEEE 754
Real	Gleitkommazahl 32-Bit IEEE 754
SInt	Vorzeichenbehafteter 8-Bit Wert
UDnt	Vorzeichenloser 32-Bit Wert
UInt	Vorzeichenloser 16-Bit Wert
USInt	Vorzeichenloser 8-Bit Wert
WString	Textvariable 16-Bit Zeichensatz

Siehe auch

Grundlagen zu Variablen (Seite 537)

2.1.7 Interne Variablen

Einleitung

Interne Variablen haben keine Verbindung zur Steuerung.

Prinzip

Interne Variablen werden im Speicher des Bediengeräts abgelegt. Daher kann nur dieses Bediengerät lesend und schreibend auf die internen Variablen zugreifen. Interne Variablen legen Sie z. B. an, um lokale Berechnungen durchzuführen.

Für interne Variablen stehen Ihnen die HMI-Datentypen zur Verfügung. Die Verfügbarkeit ist abhängig vom verwendeten Bediengerät.

Folgende HMI-Datentypen sind verfügbar:

HMI-Datentyp	Datenformat
Array	Eindimensionales Array
Bool	Binäre Variable
DateTime	Datum/ Uhrzeit Format
DInt	Vorzeichenbehafteter 32-Bit Wert
Int	Vorzeichenbehafteter 16-Bit Wert
LReal	Gleitkommazahl 64-Bit IEEE 754
Raw	Rohdaten
Real	Gleitkommazahl 32-Bit IEEE 754
SInt	Vorzeichenbehafteter 8-Bit Wert
String	Textvariable 8-Bit Zeichensatz
TextRef	Textreferenz
UDInt	Vorzeichenloser 32-Bit Wert
UInt	Vorzeichenloser 16-Bit Wert
USInt	Vorzeichenloser 8-Bit Wert
WString	Textvariable 16-Bit Zeichensatz

Systemvariablen

Systemvariablen werden zur internen Verwaltung des Projekts benötigt. Die Namen dieser Variablen beginnen jeweils mit dem Zeichen "@". Diese Variablen dürfen Sie weder löschen noch umbenennen. Den Wert der Variable können Sie nicht ändern.

Ausnahmen sind die Variablen, die von der Option "Redundancy" angelegt werden. Diese können z. B. über Skripte gesetzt werden:

- @RM_MASTER
- @RM_MASTER_NAME

Hinweis

Sie dürfen keine Variablen anlegen, deren Name mit einem @ beginnt.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Datentyp einer Variable anpassen (Seite 556)

2.2 Arbeiten mit Variablen

2.2.1 Variablen anlegen

2.2.1.1 Externe Variable anlegen

Einleitung

Mit einer externen Variable greifen Sie über eine PLC-Variable auf eine Adresse in der Steuerung zu. Für die Adressierung stehen folgende Möglichkeiten zur Verfügung:

- Symbolische Adressierung
- Absolute Adressierung

Nähere Hinweise zur symbolischen Adressierung finden Sie im Kapitel "Adressierung externer Variablen (Seite 545)". Verwenden Sie möglichst die symbolische Adressierung bei der Projektierung einer Variablen. Variablen legen Sie entweder in der Standard-Variablentabelle oder in einer selbst definierten Variablentabelle an.

Voraussetzung

- Das Projekt ist geöffnet.
- Eine Verbindung zur Steuerung ist projektiert.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

Um eine externe Variable anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "HMI-Variablen" und doppelklicken Sie die Standard-Variablentabelle. Die Variablentabelle wird geöffnet.
   Alternativ legen Sie eine neue Variablentabelle an und öffnen diese anschließend.
2. Doppelklicken Sie in der Variablentabelle in der Spalte "Name" auf "Hinzufügen". Eine neue Variable wird angelegt.
3. Wählen Sie im Inspektorfenster die Kategorie "Eigenschaften > Eigenschaften > Allgemein" und geben Sie bei Bedarf im Feld "Name" einen eigenen Variablennamen ein. Der Variablenname muss geräteweit eindeutig sein.
4. Geben Sie bei Bedarf im Feld "Anzeigename" einen Namen ein, der in Runtime angezeigt werden soll. Der Anzeigename ist sprachabhängig und kann für die gewünschten Runtime-Sprachen übersetzt werden. Der Anzeigename ist für Basic Panels, Panels und Runtime Advanced verfügbar.

5. Wählen Sie im Feld "Verbindung" die Verbindung zur gewünschten Steuerung aus. Falls Ihnen die gewünschte Verbindung nicht angezeigt wird, müssen Sie zunächst die Verbindung zur Steuerung erstellen. Die Verbindung zu einer SIMATIC S7-Steuerung erstellen Sie im Editor "Geräte & Netze". Die Verbindung zu Fremdsteuerungen erstellen Sie im Editor "Verbindungen".
   Wenn die Steuerung im Projekt enthalten ist und integrierte Verbindungen unterstützt, dann können Sie sich die Verbindung auch automatisch erstellen lassen. Wählen Sie dazu bei der Projektierung der HMI-Variable einfach eine bestehende PLC-Variable aus, mit der Sie die HMI-Variable verbinden wollen. Die integrierte Verbindung wird dann vom System automatisch erstellt.
6. Wenn Sie mit einer integrierten Verbindung arbeiten, klicken Sie im Feld "PLC-Variable" auf die Schaltfläche ... und wählen Sie in der Objektliste eine bereits angelegte PLC-Variable aus. Bestätigen Sie die Auswahl über die Schaltfläche ✓.

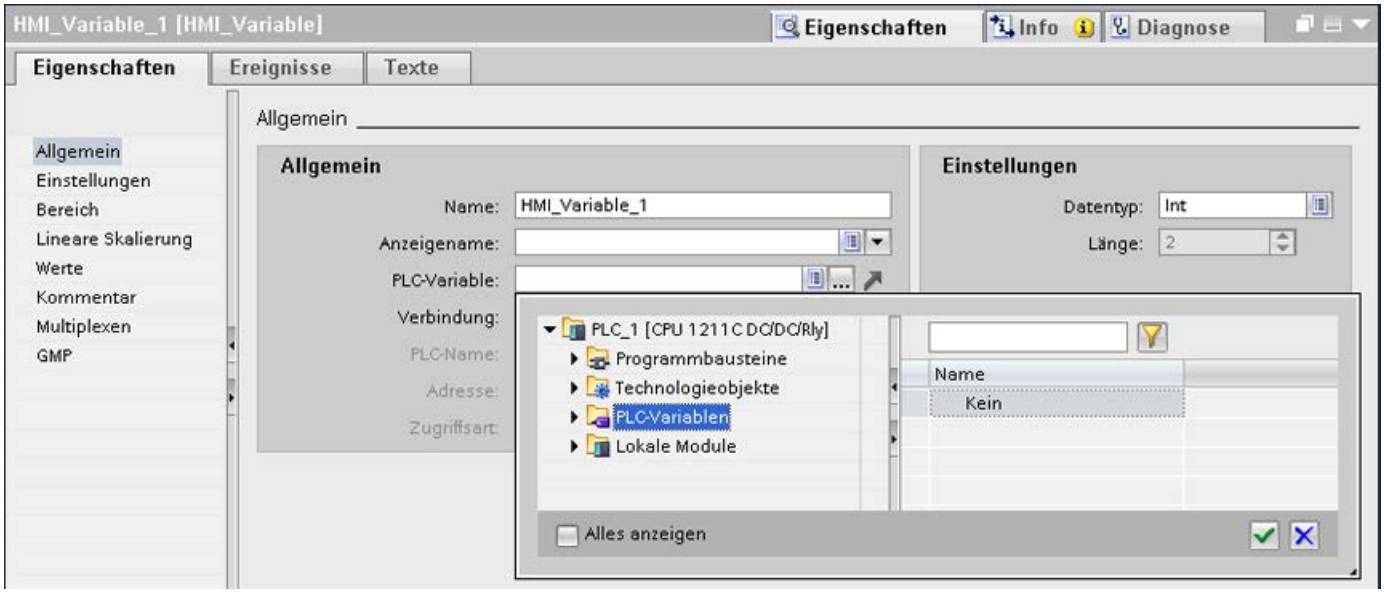

   Alternativ nutzen Sie die Autovervollständigung, um eine PLC-Variable für eine integrierte Verbindung auszuwählen.
   Wenn Sie eine PLC-Variable aus der Autovervollständigungsliste auswählen, wird sie in den Eingabepfad übernommen. Die Elemente der PLC-Variablen werden in der Autovervollständigungsliste angezeigt. Falls Sie eine PLC-Variable ausgewählt haben, die mit der HMI-Variablen verbunden werden kann, wird die PLC-Variable mit der HMI-Variablen verbunden.
7. Wenn Sie mit einer nicht integrierten Verbindung arbeiten, geben Sie im Feld "Adresse" die Adresse aus der Steuerung ein. Das Feld "PLC-Variable bleibt leer.
8. Projektieren Sie die weiteren Eigenschaften der Variable im Inspektorfenster.

Alternativ konfigurieren Sie die Variableneigenschaften direkt in der Variablentabelle". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Neue Variablen erstellen Sie alternativ auch direkt an der Verwendungsstelle, z. B. an einem EA-Feld. Hierzu klicken Sie in der Objektliste auf die Schaltfläche Neu hinzufügen. Sie konfigurieren dann die neue Variable im Inspektorfenster.

Ergebnis

Eine externe Variable ist angelegt und mit einer PLC-Variable oder einer Adresse in der Steuerung verbunden.

Alternative Vorgehensweise

Externe HMI-Variablen können Sie auch über ein Drag&Drop von Datenbaustein-Elementen oder globalen PLC-Variablen in eine HMI-Variablen-Tabelle erzeugen.

Siehe auch

Interne Variable anlegen (Seite 552)

Mehrere Variablen anlegen (Seite 553)

Adressierung externer Variablen (Seite 545)

Datentyp einer Variable anpassen (Seite 556)

Variablen verwalten (Seite 558)

Grenzwerte einer Variable (Seite 567)

Grundlagen zu Variablen (Seite 537)

2.2.1.2 Interne Variable anlegen

Einleitung

Bei internen Variablen müssen mindestens der Name und der Datentyp definiert sein. Statt einer Verbindung zu einer Steuerung wird der Eintrag "interne Variable" gewählt.

Zu Dokumentationszwecken ist es sinnvoll, zu jeder Variable einen Kommentar einzugeben.

Voraussetzung

Das Projekt ist geöffnet.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "HMI Variablen" und doppelklicken Sie den Eintrag "Standard-Variablentabelle". Die Variablentabelle wird geöffnet.
   Alternativ legen Sie eine neue Variablentabelle an und öffnen diese anschließend.
2. Doppelklicken Sie in der Variablentabelle in der Spalte "Name" auf "Hinzufügen". Eine neue Variable wird angelegt.
3. Wenn das Inspektorfenster nicht geöffnet ist, wählen Sie im Menü "Ansicht" die Option "Inspektorfenster".

4. Wählen Sie im Inspektorfenster die Kategorie "Eigenschaften > Eigenschaften > Allgemein" und geben Sie bei Bedarf im Feld "Name" einen eigenen Variablennamen ein. Der Variablenname muss projektweit eindeutig sein.
5. Geben Sie bei Bedarf im Feld "Anzeigename" einen Namen ein, der in Runtime angezeigt werden soll. Der Anzeigename ist sprachabhängig und kann für die gewünschten Runtime-Sprachen übersetzt werden. Der Anzeigename ist für Basic Panels, Panels und Runtime Advanced verfügbar.
6. Wählen Sie im Feld "Verbindung" als Verbindung den Eintrag "interne Variable".
7. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp aus.
8. Abhängig vom gewählten Datentyp müssen Sie im Feld "Länge" die Anzahl der Zeichen festlegen, die maximal in der Variable abgelegt werden. Bei numerischen Variablen ist die Länge automatisch durch den Datentyp definiert.
9. Optional geben Sie einen Kommentar zur Verwendung der Variable ein. Klicken Sie dazu im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kommentar" und geben Sie einen Text ein.

Alternativ konfigurieren Sie die Variableneigenschaften direkt in der Variablentabelle". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Neue Variablen erstellen Sie alternativ auch direkt an der Verwendungsstelle, z. B. an einem EA-Feld. Hierzu klicken Sie in der Objektliste auf die Schaltfläche 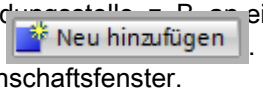

. Sie konfigurieren dann die neue Variable in dem sich öffnenden Eigenschaftsfenster.

Ergebnis

Eine interne Variable wird angelegt, die Sie in Ihrem Projekt einsetzen können.

In weiteren Schritten konfigurieren Sie die Variable, indem Sie z. B. Startwert und Grenzwerte festlegen.

Siehe auch

Externe Variable anlegen (Seite 550)

2.2.1.3 Mehrere Variablen anlegen

Einleitung

Im einer Variablentabelle legen Sie eine größere Anzahl gleicher Variablen an, indem Sie die Tabellenzeilen unterhalb einer Variable automatisch ausfüllen lassen.

Beim automatischen Ausfüllen werden die Variablennamen automatisch hoch gezählt.

Durch automatisches Ausfüllen können Sie auch einzelne Eigenschaften einer Variable auf die darunter liegenden Tabellenzellen und damit in die entsprechenden Variablen übertragen.

Wenn Sie das automatische Ausfüllen auf bereits gefüllte Zellen einer Variablentabelle anwenden, werden Sie gefragt, ob Sie die Zellen überschreiben oder neue Variablen einfügen wollen.

Wenn Sie bereits konfigurierte Variablen nicht überschreiben wollen, aktivieren Sie den Einfügemodus. Den Einfügemodus aktivieren Sie, indem Sie während des Einfügens die Taste <Strg> gedrückt halten. Bereits bestehende Einträge in der Variablentabelle werden bei aktiviertem Einfügemodus nach unten verschoben.

Voraussetzung

- Das Projekt ist geöffnet.
- Eine Variablentabelle ist geöffnet.
- Die Variable, die als Vorlage für weitere Variablen dienen soll, ist konfiguriert.

Vorgehen

1. Wenn Sie neue Variablen erzeugen wollen, markieren Sie in der Spalte "Name" die Variable, die als Vorlage für die neuen Variablen dienen soll.
   Wenn Sie eine Eigenschaft einer Variable auf darunter liegende Variablen übertragen wollen, markieren Sie die Zelle, die diese Eigenschaft enthält.
   Die markierte Zelle wird farbig hinterlegt und an der unteren rechten Ecke erscheint ein kleines, blaues Quadrat. Positionieren Sie den Mauszeiger über dem Quadrat, der Cursor ändert sich in ein schwarzes Kreuz.

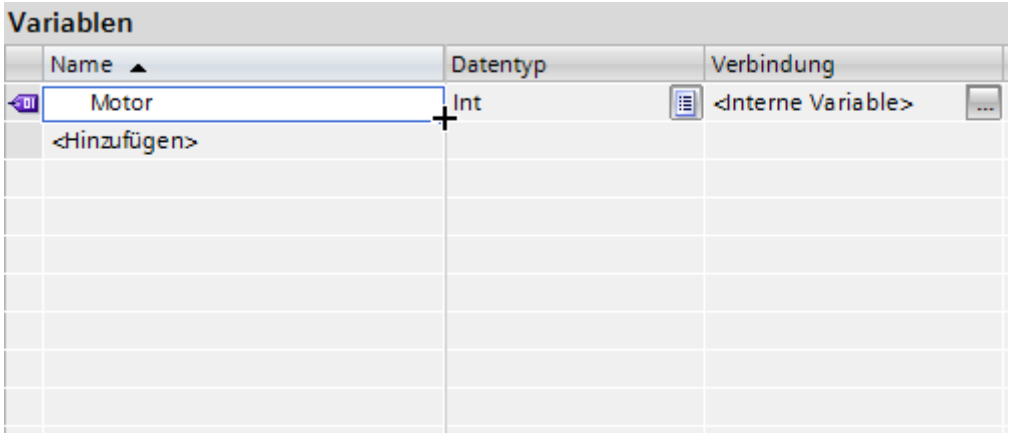

2. Ziehen Sie den Cursor mit gedrückter Maustaste über die Zellen nach unten, die Sie automatisch ausfüllen möchten.
   Die Markierung wird auf diesen Bereich erweitert.

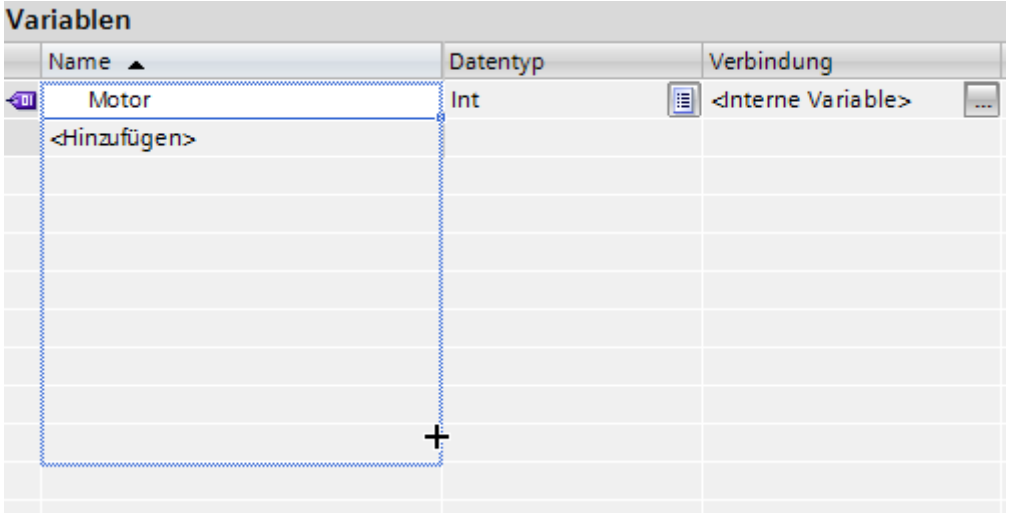

3. Lassen Sie die Maustaste los. Alle markierten Zellen werden automatisch ausgefüllt.
   In allen leeren Zeilen des markierten Bereichs werden neue Variablen angelegt.

Variablen

Export

	Datentyp	Verbindung
Motor	Int	<Interne Variable>
Motor_1	Int	<Interne Variable>
Motor_2	Int	<Interne Variable>
Motor_3	Int	<Interne Variable>
Motor_4	Int	<Interne Variable>
Motor_5	Int	<Interne Variable>
Motor_6	Int	<Interne Variable>

Ergebnis

Abhängig davon, welche Zelle markiert war, werden einzelne Eigenschaften automatisch ausgefüllt oder neue Variablen angelegt.

Siehe auch

Externe Variable anlegen (Seite 550)

2.2.1.4 Datentyp einer Variable anpassen

Einleitung

Wenn Sie eine Variable anlegen, dann weisen Sie der Variable einen der möglichen Datentypen zu. Dieser Datentyp hängt von der Art der Daten ab, für die Sie die Variable verwenden möchten.

Die verfügbaren Datentypen sind abhängig von dem verbundenen Kommunikationspartner, z. B. einer Steuerung.

Hinweis

Wenn Sie bei einer bereits bestehenden, externen Variablen den Datentyp ändern, dann wird die bereits definierte Variablenadresse als ungültig gekennzeichnet. Dies liegt daran, dass sich bei einer Datentypänderung auch die PLC-Adresse ändert.

Formatanpassung

Der Datentyp einer externen Variablen wird von WinCC entsprechend dem Datentyp der verbundenen PLC-Variable eingestellt. Wenn der Datentyp der PLC-Variable in WinCC nicht verfügbar ist, wird an der HMI-Variable automatisch ein kompatibler Datentyp verwendet. Bei Bedarf können Sie festlegen, dass WinCC einen anderen Datentyp verwendet und eine Formatanpassung zwischen dem Datentyp der PLC-Variable und dem Datentyp der HMI-Variable durchführt.

In WinCC stehen folgende Datentypen zur Verfügung:

HMI-Datentyp	Beschreibung	Wertebereich
Array	Variable mit enthaltenen Elementen[2)3)]	
DateTime	Datum/ Uhrzeit Format[2)]	
SInt	Vorzeichenbehafteter 8-Bit Wert	-128 ... +127
USInt	Vorzeichenloser 8-Bit Wert	0 ... 255
Int	Vorzeichenbehafteter 16-Bit Wert	-32768 ... +32767
UInt	Vorzeichenloser 16-Bit Wert	0 ... 65535
DInt	Vorzeichenbehafteter 32-Bit Wert	-2147483648 ... +2147483647
UDInt	Vorzeichenloser 32-Bit Wert	0 ... 4294967295
Real	Gleitkommazahl 32-Bit IEEE 754	+-3.402823e+38
LReal	Gleitkommazahl 64-Bit IEEE 754	+-1.79769313486231e+308
Bool	Binäre Variable	0 ... 1
String	Textvariable 8-Bit Zeichensatz	
WString	Textvariable 16-Bit Zeichensatz	
TextRef	Textreferenz[1]	
Raw	Rohdatentyp	

[1]: Der Datentyp "TextRef" ist nur bei internen Variablen verfügbar. Mit Variablen vom Datentyp "TextRef" verweisen Sie auf einen Eintrag in den Projekttexten. Wenn Sie an einem Objekt einen Text anzeigen möchten, der bei einer Sprachumschaltung zum Text der eingestellten Sprache wechselt, verwenden Sie den Datentyp "TextRef". Der Variable weisen Sie den Projekttext zu.

[2]: Nur in Verbindung mit S7-1200 oder S7-1500 verfügbar

[3]: Um Strukturkonflikte in Runtime zu vermeiden, synchronisieren Sie die Variablennamen und ersetzen Sie dabei die Trennzeichen "[" und "]".

Für die Formatanpassung wählen Sie an der jeweiligen externen Variable den gewünschten PLC-Datentyp aus. Im Feld "HMI-Datentyp" wird dann automatisch der passende Standarddatentyp für die Verwendung in WinCC ausgewählt. Bei Bedarf ändern Sie die Formatanpassung. Nähere Hinweise hierzu finden Sie im Kapitel

Abbildung und Codierung der Datentypen (Seite 623)

Datentypen ohne Formatanpassung

Die Datentypen ohne Formatanpassung werden 1:1 abgebildet.

SIMATIC S7-300/400-Datentypen ohne Formatanpassung

PLC-Datentyp	Beschreibung
Bool	Keine Formatanpassung
String	Keine Formatanpassung

OPC-DA-Datentypen ohne Formatanpassung

OPC-DA-Datentyp	Beschreibung
VT_BSTR	Keine Formatanpassung
VT_BOOL	Keine Formatanpassung

OPC-DA-XML-Datentypen ohne Formatanpassung

OPC-DA-Datentyp	Beschreibung
String	Keine Formatanpassung
boolean	Keine Formatanpassung
Base64Binary	Keine Formatanpassung

Siehe auch

Externe Variable anlegen (Seite 550)

Abbildung und Codierung der Datentypen (Seite 623)

Einstellungen zum Synchronisieren (Seite 564)

Datentypen (Seite 631)

2.2.2 Bearbeiten von Variablen

2.2.2.1 Variablen verwalten

Einleitung

Eine Variable können Sie jederzeit umbenennen, kopieren oder löschen.

Wenn Sie eine Variable umbenennen, muss der neue Name geräteweit eindeutig sein.

Wenn Sie eine Variable mit dem Befehl "Kopieren" in die Zwischenablage kopieren, werden die mit der Variablen Referenzen ebenfalls kopiert.

Wenn Sie eine Variable mit dem Befehl "Einfügen" in ein anderes Gerät einfügen, dann wird die Variable ohne die verbundenen Referenzen eingefügt. Von einer Referenz wird lediglich der Objektname eingefügt. Wenn im Zielsystem eine gleichnamige Referenz mit gültigen Eigenschaften existiert, dann wird die existierende Referenz mit der kopierten Variable verbunden.

Wenn Sie eine Variable kopieren, werden einige mit der Variablen verknüpfte Objekte ebenfalls kopiert. Folgende Objekte werden kopiert:

- Archivvariablen
- Zyklen
- Meldungen

Wenn Sie die kopierte Variable in ein anderes Gerät einfügen, dann wird die Variable zusammen mit den verknüpften Objekten eingefügt.

Voraussetzung

- Die Variable, die Sie umbenennen, kopieren oder löschen wollen, ist angelegt.
- Die Variablentabelle, in der die Variable enthalten ist, ist geöffnet.

Variable umbenennen

Hinweis

Unter folgenden Voraussetzungen kann bei einer Umbenennung die Verbindung zur Variablen in Runtime unterbrochen werden:

- HMI-Variable wird in einem Typ verwendet, z. B. zur Dynamisierung einer Objekteigenschaft über ein Skript.
- Eine oder mehrere Instanzen des Typs werden im Projekt verwendet.
- Projekt ist in Runtime.
- Nach der Umbenennung führen Sie den Befehl "Übersetzen > Software (nur Änderungen)" aus.

Abhilfe: Beenden Sie Runtime und benennen Sie die Variable um. Führen Sie den Befehl "Übersetzen > Software (komplett Übersetzen)" aus.

1. Markieren Sie das Feld "Name" der Variable in der Variablentabelle.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen neuen Namen ein.
   Die Variable wird unter dem neuen Namen angezeigt.

Variable kopieren

1. Markieren Sie eine oder mehrere Variablen in der Variablentabelle.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Klicken Sie auf die Stelle, an der Sie die Variable einfügen wollen. Klicken Sie z. B. auf eine andere Variablentabelle desselben Geräts oder auf eine Variablentabelle in einem zweiten Gerät.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen". Die Variable wird wie oben beschrieben eingefügt.

Variable löschen

1. Markieren Sie eine oder mehrere Variablen in der Variablentabelle.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Querverweise". Überprüfen Sie im Editor "Querverweise", wo die Variablen verwendet werden. Sie sehen so, welchen Einfluss das Löschen der Variablen auf Ihr Projekt hat.
3. Wählen Sie im Kontextmenü der Variablen den Befehl "Löschen".
   Alle markierten Variablen werden gelöscht.

Hinweis

Wenn Sie möchten, dass bei gelöschten HMI-Variablen in Bildern eine Warnung beim Übersetzen ausgegeben wird, deaktivieren Sie unter "Runtime-Einstellungen > Allgemein > Compiler-Optionen" das Optionskästchen "Gelöschte HMI-Variablen in Bildern nicht prüfen". Wenn diese Option nicht aktiviert ist, wird das betroffene Feld mit der gelöschten Variablen als ungültig markiert.

Export und Import von Variablen

WinCC bietet Ihnen die Möglichkeit, Variablen zu exportieren und zu importieren. Mit dem Export und Import haben Sie die Möglichkeit, Variablen aus einem Projekt zu exportieren und in ein anderes Projekt zu importieren. Außerdem haben Sie die Möglichkeit, größere Mengen von Variablen außerhalb von WinCC anzulegen, zu bearbeiten und anschließend in beliebige WinCC-Projekte zu importieren.

Siehe auch

Variablen bearbeiten (Seite 560)

Mehrere Variablen gleichzeitig konfigurieren (Seite 561)

Mehrere Variablen gleichzeitig in einem Bild verwenden (Seite 562)

Variablen synchronisieren (Seite 566)

Externe Variable anlegen (Seite 550)

Variablen importieren und exportieren (Seite 2069)

2.2.2.2 Variablen bearbeiten

Einleitung

Variablen können Sie jederzeit an geänderte Anforderungen im Projekt anpassen.

Variablen bearbeiten

Um die Konfiguration einer Variablen zu ändern, haben Sie folgende Möglichkeiten:

- Sie öffnen die Variablentabelle, in der die Variable enthalten ist.
- Sie öffnen die Variablentabelle "Alle Variablen anzeigen".
- Sie öffnen das Inspektorfenster einer Variable über die Schaltfläche "Bearbeiten" in der Objektauswahl am Anzeige- und Bedienobjekt.

Sie können in den Variablentabellen z. B. die Eigenschaften mehrerer Variablen vergleichen und aufeinander abstimmen oder die Variablen nach ihren Eigenschaften sortieren.

Ändern Sie die Eigenschaften entweder direkt in der Tabelle oder im Inspektorfenster.

Wenn Sie eine Variableneigenschaft ändern und durch die Änderung Widersprüche zu anderen Eigenschaften auftreten, werden Sie durch eine farbige Hinterlegung darauf aufmerksam gemacht. Wenn Sie z. B. die Variable mit einer anderen Steuerung verbinden, die den eingestellten Datentyp nicht unterstützt, wird diese Eigenschaft farbig hinterlegt.

Siehe auch

Variablen verwalten (Seite 558)

2.2.2.3 Mehrere Variablen gleichzeitig konfigurieren

Einleitung

In WinCC weisen Sie mehreren Variablen in einem Arbeitsgang übereinstimmende Eigenschaften zu. Dies ermöglicht Ihnen, effizient zu projektieren.

Voraussetzung

- Die Variablen, die Sie konfigurieren wollen, sind angelegt.
- Die Variablentabelle ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Wählen Sie in der Variablentabelle alle Variablen aus, die Sie gemeinsam konfigurieren wollen.
   Wenn die Variablen in der ausgewählten Eigenschaft übereinstimmen, wird die Einstellung für diese Eigenschaft im Inspektorfenster angezeigt. Wenn die Eigenschaften nicht übereinstimmen, bleibt das zugehörige Feld leer.
2. Legen Sie die gemeinsame Eigenschaft im Inspektorfenster oder direkt in der Variablentabelle fest.

Wenn Sie an mehreren Variablen gemeinsam eine Eigenschaft ändern, dann wird nur diese eine Eigenschaft geändert. Die weiteren Eigenschaften der Variablen bleiben unverändert.

Ergebnis

Alle markierten Variablen werden neu konfiguriert.

Wenn Sie anschließend Variableneigenschaften konfigurieren wollen, in denen sich die Variablen unterscheiden, heben Sie dazu die Mehrfachauswahl auf.

Siehe auch

Variablen verwalten (Seite 558)

2.2.2.4 Mehrere Variablen gleichzeitig in einem Bild verwenden

Einleitung

In WinCC können Sie in einem Arbeitsgang mehrere E/A-Felder in einem Bild erstellen, die mit Variablen verbunden sind. Dies ermöglicht Ihnen, effizient zu projektieren.

Voraussetzung

- Mehrere Variablen sind angelegt.
- Ein Bild ist geöffnet.

Vorgehen

1. Markieren Sie in der Projektnavigation unter "HMI-Variablen" die gewünschte Variablentabelle.

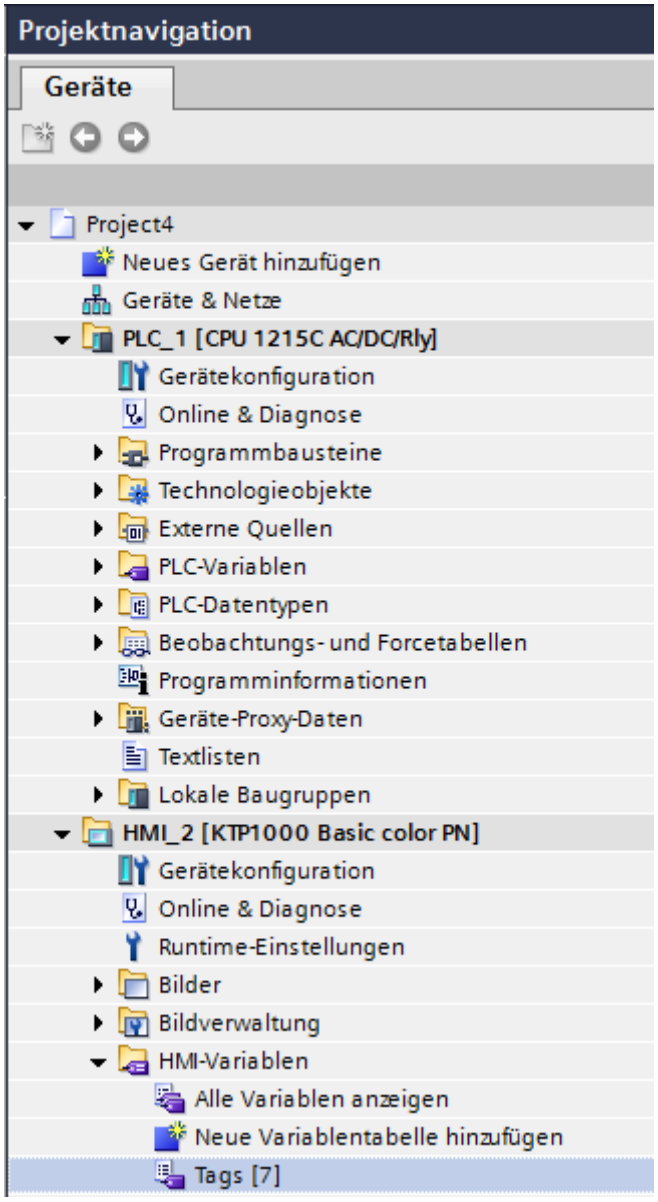

2. Wählen Sie das Detailfenster im unteren Bereich der Projektnavigation. Im Detailfenster werden die in der gewählten Variablengruppe vorhandenen Variablen angezeigt.

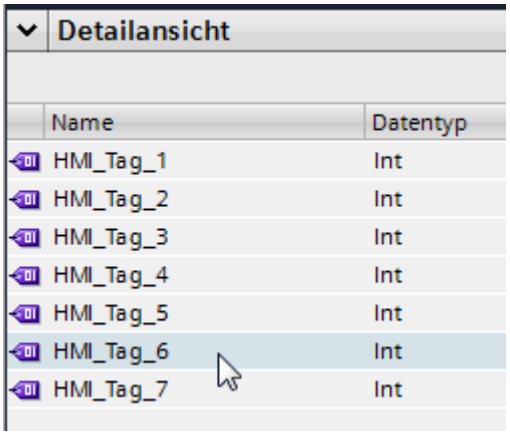

3. Markieren Sie im Detailfenster die Variablen.
4. Ziehen Sie die Variablen per Drag&Drop in das Bild. Für jede Variable wird ein E/A-Feld erstellt, das mit der jeweiligen Variable verbunden ist.

Hinweis

Wenn Sie eine PLC-Variable mit Drag&Drop aus dem Detailfenster in den Arbeitsbereich ziehen, wird zusätzlich im Editor "Geräte & Netze" ein Netzwerk und eine Verbindung angelegt.

Siehe auch

Variablen verwalten (Seite 558)

2.2.2.5 Einstellungen zum Synchronisieren

Einleitung

In WinCC haben Sie die Möglichkeit, die Position eines Datenwerts in der Struktur eines Datenbausteins im WinCC-Variablennamen abzubilden. Dazu synchronisieren Sie die WinCC-Variablen mit den verbundenen PLC-Variablen. Bei Bedarf wird dabei der PLC-Name als Präfix gesetzt.

Den Namensabgleich projektieren Sie abhängig von den Anforderungen Ihres Projekts unter "Runtime-Einstellungen > Einstellungen für Variablen". WinCC bietet Ihnen dafür mehrere Optionen.

Ob die Namen abgeglichen werden und unter welchen Bedingungen legen Sie während des Synchronisierens fest.

Einstellungen für Variablen

Um Konflikte innerhalb komplexer Variablentypen zu vermeiden, projektieren Sie unter "Einstellung zu Variablen" wie beim Namensabgleich die Trennzeichen der Pfadangabe aus STEP 7 in WinCC ersetzt werden:

- Kompatibilitätsmodus
  Im Kompatibilitätsmodus wird ein Unterstrich zwischen den Namen des Datenbausteins und dem Elementnamen der ersten Hierarchiestufe gesetzt. Die Trennzeichen der unteren Hierarchiestufen bleiben erhalten. Wenn Sie diese Option aktivieren, sind alle anderen Optionen deaktiviert.
- Trennzeichen ersetzen
  Entsprechend Ihrer Auswahl werden beim Synchronisieren die Trennzeichen aller Hierarchiestufen ersetzt.
- PLC-Präfix
  Der PLC-Name wird als Präfix vor den WinCC-Variablennamen gesetzt. Die Option setzen Sie jeweils für eine HMI-Verbindung.

Hinweis

Doppelte Variablennamen

Wenn der neu generierte Variablenname bereits verwendet ist, wird in runden Klammern eine Nummer hinzugefügt, z. B. Datablock_1_Static_2{1}(1).

Beispiel

Die Steuerung "PLC1" enthält den strukturierten Datenbaustein "DB1". Das Datenbausteinelement "Db1.a[1].b.c[3]" wird in einem Bild verwendet. Abhängig von Ihren Einstellungen wird der WinCC-Variablenname wie folgt generiert:

WinCC-Variablenname	Gewählte Option	Anmerkung
Db1_a[1].b.c[3]	Kompatibilitätsmodus	
Plc1.Db1.a[1].b.c[3]	PLC-Präfix	
Db1;a(1);b;c(3)	Punkt und Klammer ersetzen mit ; ( )	
Plc1_Db1_a{10}_b_c{3}	Punkt und Klammer ersetzen mit _ { } PLC-Präfix	
Db1.a[1].b.c[3]	Trennzeichen ersetzen ohne Zeichenauswahl	An der Verwendungsstelle im Bild wird der Variablenname in Anführungszeichen gesetzt: "Db1.a[1].b.c[3]"

Siehe auch

Variablen synchronisieren (Seite 566)

2.2.2.6 Variablen synchronisieren

Einleitung

Um PLC- und HMI-Variablen zu synchronisieren bietet Ihnen WinCC folgende Möglichkeiten:

- Variablen mit oder ohne Namensabgleich zwischen PLC und WinCC synchronisieren
  Dabei stehen Ihnen verschiedene Optionen zur Verfügung.
- Variablen mit Adressen in der Steuerung verbinden
  Diese Vorgehensweise bietet sich an, wenn z. B. die Verbindung zwischen Bediengerät und Steuerung geändert wurde und dabei die Variablenanbindungen verloren gingen. Die Funktion ist auch verwendbar, wenn Sie Steuerungsprogramm und HMI-Projekt separat projektiert haben.

Voraussetzung

- Externe HMI-Variablen sind angelegt.
- PLC-Variablen sind angelegt.
- Eine HMI-Verbindung zu einer Steuerung im Projekt ist angelegt.

Vorgehen

Um HMI-Variablen mit PLC-Variablen zu synchronisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Verzeichnis, welches die betreffenden Variablen enthält.
2. Wählen Sie im Kontextmenü "Synchronisieren zum Abgleich mit der PLC-Variable".
   Ein Dialog wird geöffnet.

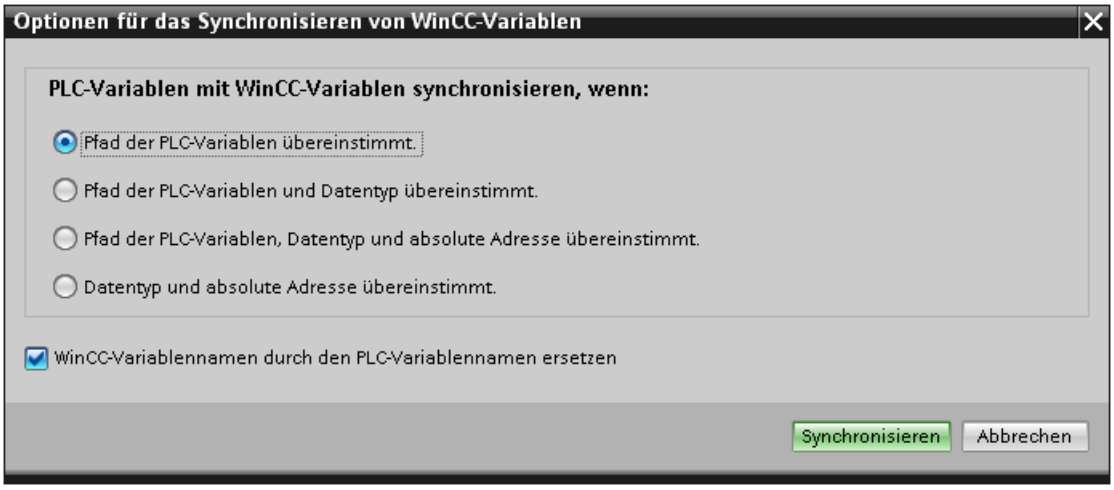

3. Wählen Sie die gewünschte Option.
   Wenn Sie die Variablen ohne Namensabgleich synchronisieren wollen, deaktivieren Sie "WinCC-Variablennamen durch den PLC-Variablennamen ersetzten".
   Wenn Sie HMI-Variablen mit absolutem Zugriff neu verbinden wollen, wählen Sie "Datentyp und absolute Adresse übereinstimmt".
4. Bestätigen Sie mit "Synchronisieren".
   Das System sucht entsprechend der gewählten Option nach einer passenden PLC-Variable.

Ergebnis

Die externen HMI-Variablen sind mit den PLC-Variablen synchronisiert.

Wenn Sie die Option "Datentyp und absolute Adresse übereinstimmt" gewählt haben, ist die Variablenverbindung etabliert, sobald eine passende PLC-Variable gefunden wurde.

Wenn Sie eine andere Option gewählt haben, sind die WinCC-Variablen entsprechend aktualisiert und die PLC-Variablennamen nach WinCC übernommen.

Siehe auch

Variablen verwalten (Seite 558)

Einstellungen zum Synchronisieren (Seite 564)

2.2.3 Variablen konfigurieren

2.2.3.1 Grundlegende Variablenkonfiguration

Grenzwerte einer Variable

Einleitung

Für numerische Variablen können Sie Wertebereiche durch Grenzwerte einschränken.

Prinzip

Für numerische Variablen legen Sie einen Wertebereich fest, der durch einen oberen Grenzwert und einen unteren Grenzwert definiert wird.

Sie projektieren vier Grenzwerte, die den Wertebereich einschränken. Mithilfe von Grenzen Oben 2 und Unten 2 legen Sie den maximalen und minimalen Wert für den Wertebereich fest.

Die Grenzen Oben 1 und Unten 1 legen die Schwellenwerte fest, bei welchen der Normalbereich überschritten, bzw. unterschritten wird.

Grenze	Verwendung
Oben 2	Legt den maximalen Wert fest.
Oben 1	Legt den Wert fest, bei dem der Normalbereich überschritten wird.
Unten 1	Legt den Wert fest, bei dem der Normalbereich unterschritten wird.
Unten 2	Legt den minimalen Wert fest.

Wenn der Bediener einen Wert für die Variable eingibt, der außerhalb des projektierten Wertebereichs liegt, wird die Eingabe abgewiesen. Der Wert wird nicht übernommen. Bei der Projektierung für Runtime Advanced und Panels projektieren Sie eine Funktionsliste. Wenn der Variablenwert den Wertebereich verlässt, wird die Funktionsliste abgearbeitet.

Hinweis

Wenn Sie bei Verletzung eines Grenzwerts eine Analogmeldung ausgeben wollen, projektieren Sie dies auf der Registerkarte "Analogmeldungen" der jeweiligen Variable. Alternativ projektieren Sie die Analogmeldung im Editor "HMI-Meldungen". Die Werte für die Ausgabe einer Analogmeldung sind unabhängig von den projektierten Grenzwerten einer Variable.

Anwendungsbeispiel

Verwenden Sie Grenzwerte, z. B. um den Bediener zu warnen, wenn der Wert einer Variable in einen kritischen Bereich eintritt.

Siehe auch

Grenzwerte einer Variable festlegen (Seite 568)

Startwert einer Variable (Seite 570)

Startwert einer Variable festlegen (Seite 570)

Variable mit einer anderen Steuerung verbinden (Seite 571)

Lineares Skalieren einer Variable (Seite 572)

Variable linear skalieren (Seite 573)

Grenzwerte einer Variable festlegen

Einleitung

Für numerische Variablen definieren Sie einen Wertebereich, indem Sie einen unteren und einen oberen Grenzwert sowie die Schwellwerte definieren.

Für Runtime Advanced und Panels können Sie an das Unterschreiten oder Überschreiten des Wertebereichs zusätzlich das Abarbeiten einer Funktionsliste projektieren.

Voraussetzung

- Die Variable, für die Sie Grenzwerte festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

Vorgehensweise

Um Grenzwerte einer Variable festzulegen gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Bereich". Wenn Sie für einen der Grenzwerte einen konstanten Wert festlegen wollen, wählen Sie über die Schaltfläche den Eintrag "Konstante" aus. Geben Sie im entsprechenden Feld eine Zahl ein.
   Wenn Sie für einen der Grenzwerte den Wert einer Variable verwenden wollen, wählen Sie über die Schaltfläche den Eintrag "HMI-Variable" aus. Mithilfe der Objektliste legen Sie die Variable für den Grenzwert fest.
2. Wiederholen Sie Schritt 1 sinngemäß, um weitere Grenzwerte für die Variable festzulegen.

Alternative Vorgehensweise

Alternativ konfigurieren Sie die Grenzwerte direkt in der Variablentabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Funktionsliste projektieren für Runtime Advanced und Panels

Wenn Sie für Runtime Advanced und Panels projektieren, können Sie bei überschreiten oder unterschreiten des Wertebereichs eine Funktionsliste starten.

1. Wenn Sie bei Unterschreiten des Wertebereichs eine Funktionsliste starten wollen, klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Bei Unterschreitung". Legen Sie dort eine Funktionsliste an.
2. Wenn Sie bei Überschreiten des Wertebereichs eine Funktionsliste starten wollen, klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Bei Überschreitung". Legen Sie dort eine Funktionsliste an.

Ergebnis

Sie haben für die ausgewählte Variable einen Wertebereich festgelegt, der durch die Grenzwerte definiert ist. Bei Unterschreiten oder Überschreiten des Wertebereichs wird gegebenenfalls eine Funktionsliste abgearbeitet.

Siehe auch

Grenzwerte einer Variable (Seite 567)

Startwert einer Variable

Wert einer Variablen zu Runtime-Start

Für numerische Variablen und Variablen für Datum-/Uhrzeit-Werte können Sie einen Startwert projektieren. Beim Start von Runtime wird die Variable mit diesem Wert vorbelegt. Auf diese Weise schaffen Sie einen definierten Zustand der Variablen zu Runtime-Start.

Bei externen Variablen wird der Startwert so lange am Bediengerät angezeigt, bis er von der Steuerung oder durch eine Eingabe überschrieben wird.

Wenn kein Startwert projektiert wurde, enthält die Variable beim Start von Runtime den Wert "0".

In WinCC Runtime Professional können Sie an einer Variablen mit dem Datentyp "String" an Stelle des Startwerts einen Variablenwert eingeben. Der Variablenwert wird im Editor "Projekttexte" abgelegt und ist mehrsprachig. Nach der Übersetzung des Textes wird er in Runtime als sprachabhängiger Startwert angezeigt.

Anwendungsbeispiel

Ein E/A-Feld können Sie mit einem Defaultwert vorbelegen. Legen Sie den gewünschten Defaultwert als Startwert für die Variable fest, die mit dem E/A-Feld verknüpft ist.

Siehe auch

Grenzwerte einer Variable (Seite 567)

Startwert einer Variable festlegen (Seite 570)

Startwert einer Variable festlegen

Einleitung

In WinCC können Sie für eine numerische Variable und eine Variable für Datum-/Uhrzeit-Werte einen Startwert projektieren, den diese bei Runtime-Start annimmt.

Voraussetzung

- Die Variable, für die Sie einen Startwert festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften der Variable ist geöffnet.

Vorgehensweise

Um einen Startwert zu projektieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Werte".
2. Geben Sie den gewünschten "Startwert" ein.

Alternative Vorgehensweise

Alternativ konfigurieren Sie den Startwert direkt in der Variablentabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Der gewählte Startwert für die Variable wird in das Projekt übernommen.

Siehe auch

Grenzwerte einer Variable (Seite 567)

Startwert einer Variable (Seite 570)

Variable mit einer anderen Steuerung verbinden

Einleitung

Sie können in WinCC jederzeit die Steuerungsanbindung einer Variable ändern. Dies ist z. B. nötig, wenn Sie die Konfiguration Ihrer Anlage ändern.

Abhängig von der gewählten Steuerung müssen Sie anschließend die Konfiguration der Variable anpassen. Anhand der farbigen Hinterlegung sehen Sie, welche Variableneigenschaften Sie ändern müssen.

Voraussetzung

- Die externe Variable, deren Verbindung Sie ändern wollen, ist angelegt.
- Die Verbindungen zu den Steuerungen sind angelegt.
- Das Eigenschaftsfenster für die Variable ist geöffnet.

Vorgehensweise

Um die Steuerungsanbindung einer Variable zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Feld "Verbindung" die neue Verbindung aus.
   Die Variableneigenschaften, die Sie anpassen müssen, werden in der Variablentabelle und im Inspektorfenster farbig hinterlegt.
3. Passen Sie alle farbig hinterlegten Eigenschaften der Variable an die Erfordernisse der neuen Steuerung an.

Ergebnis

Die externe Variable wird mit der neuen Steuerung verbunden.

Siehe auch

Grenzwerte einer Variable (Seite 567)

Lineares Skalieren einer Variable

Einleitung

Numerische Datentypen können Sie linear skalieren. Die in der Steuerung vorhandenen Prozesswerte einer externen Variable lassen sich auf einen bestimmten Wertebereich im Projekt abbilden.

Prinzip

Für die lineare Skalierung einer Variable werden ein Wertebereich am Bediengerät und ein Wertebereich an der Steuerung angegeben. Die Wertebereiche werden linear aufeinander abgebildet.

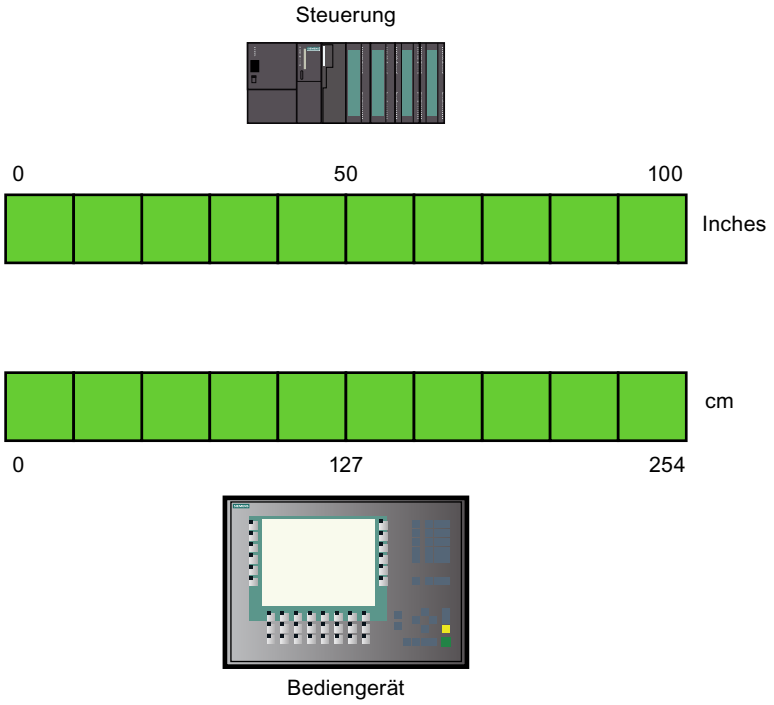

Sobald ein Prozesswert vom Bediengerät in eine externe Variable geschrieben wird, wird er automatisch auf den Wertebereich der Steuerung abgebildet. Sobald ein Prozesswert vom Bediengerät aus der externen Variablen gelesen wird, wird eine entsprechende Umkehrtransformation vorgenommen.

Hinweis

Für das automatische Umrechnen von Prozesswerten können Sie auch die Systemfunktionen "LineareSkalierung" und "InvertiereLineareSkalierung" einsetzen.

Anwendungsbeispiel

Der Benutzer gibt Längenmaße in cm ein, die Steuerung erwartet jedoch Inch. Die eingegebenen Werte werden automatisch umgerechnet, bevor sie an die Steuerung weitergeleitet werden. Mit der linearen Skalierung passen Sie den Wertebereich der Steuerung [0 ... 100] auf den Wertebereich [0 ... 254] im Bediengerät an.

Siehe auch

Variable linear skalieren (Seite 573)

Grenzwerte einer Variable (Seite 567)

Variable linear skalieren

Einleitung

Für die lineare Skalierung einer Variable werden ein Wertebereich am Bediengerät und ein Wertebereich an der Steuerung angegeben. Die Wertebereiche werden linear aufeinander abgebildet.

Voraussetzung

- Die externe Variable, die Sie linear skalieren wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

Vorgehensweise

Um eine Variable linear zu skalieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Lineare Skalierung".
2. Klicken Sie auf "Aktivieren", damit die lineare Skalierung eingeschaltet wird.
   Mit dieser Option schalten Sie die lineare Skalierung zeitweilig aus, z. B. zu Testzwecken. Bereits vorgenommene Einstellungen für die lineare Skalierung bleiben dabei erhalten.
3. Geben Sie im Bereich "Steuerung" den Endwert und den Anfangswert für den Wertebereich ein, der für die Prozesswerte in der Steuerung gilt.
4. Geben Sie im Bereich "Bediengerät" den Endwert und den Anfangswert für den Wertebereich ein, der für die Prozesswerte am Bediengerät verwendet wird.

Ergebnis

In Runtime werden die Prozesswerte automatisch von einem Wertebereich auf den anderen abgebildet.

Hinweis

Für das automatische Umrechnen von Prozesswerten stehen Ihnen auch die Systemfunktionen "LineareSkalierung" und "InvertiereLineareSkalierung" zur Verfügung.

Siehe auch

Lineares Skalieren einer Variable (Seite 572)

Grenzwerte einer Variable (Seite 567)

Zeichen für HMI-Variablen vom Typ WString/WChar bereitstellen

Einleitung

In Variablen vom Typ WString oder WChar legen Sie die Zeichenketten fest, die mit bestimmten Schriftarten dargestellt werden können.

Über die Projektierung werden nur die Zeichen auf die Basic Panels und Comfort Panels übertragen, die in Texten in der Projektierung verwendet sind. Daher ist es möglich, dass Zeichen, die über Variablen vom Typ WString oder WChar angezeigt werden, nicht dargestellt werden.

Auf den Comfort Panels sind die Schriftarten Tahoma und Courier New vollständig vorinstalliert. Auf den Basic Panels ist die Schriftart Tahoma vollständig vorinstalliert. Mit der Schriftart Tahoma können die Zeichen der meisten Sprachen dargestellt werden. Die Zeichen der Sprachen Chinesisch, Taiwanesisch, Japanisch und Koreanisch können mit Tahoma nicht dargestellt werden.

WinCC Runtime Advanced unterstützt alle Schriftarten, die auf dem Projektierungs-PC und dem Runtime-PC installiert sind. Wenn Sie eine andere Schriftart verwenden möchten, installieren Sie diese auf Ihrem Projektierungs-PC und Runtime-PC.

Für Comfort Panels ist es möglich, die komplette Schriftart über ProSave auf das Bediengerät zu laden.

Für Basic Panels ist die Installation der Schriftarten über ProSave nicht möglich. Daher müssen die Zeichen immer in den projektierten Texten enthalten sein.

Grundlagen

Für Comfort Panels können Sie über die zusätzlichen ProSave-Optionen folgende Schriftarten auf das Bediengerät laden:

- SimSun für die Eingabe von Zeichenketten in Chinesisch (Volksrepublik China), Chinesisch (Singapur), Englisch, Deutsch, Italienisch, Französisch und Spanisch
- Gulim für die Eingabe von Zeichenketten in Koreanisch
- MS PGothic für die Eingabe von Zeichenketten in Japanisch

Weitere Informationen zum Thema "Asiatische Schriftarten über ProSave laden" finden Sie unter Asiatische Schriftarten auf das Bediengerät laden (Seite 2441).

Hinweis

"Chinesisch traditionell" (Taiwan, Hong Kong, Macao)

Für die Darstellung der traditionellen chinesischen Zeichen für Taiwan, Hong Kong und Macao benötigen Sie die Schriftart "MingLiU". Es ist nicht möglich, diese Schriftart über die ProSave-Optionen auf die Comfort Panels zu laden. Die Schriftarten SimSun und Tahoma können für die Darstellung der traditionellen chinesischen Zeichen nicht verwendet werden.

Daher müssen die traditionellen chinesischen Zeichen, die in WStrings/WChars vorkommen, auf Comfort Panels in den projektierten Texten enthalten sein.

Zeichen als konfigurierte Texte in Bildern verwenden

Um auf einem Bediengerät in Runtime nicht installierte Schriftarten zu verwenden, stellen Sie die Zeichen als konfigurierte Texte in einem Bild bereit:

1. Erstellen Sie ein neues Bild.
2. Fügen Sie ein Textfeld ein.
3. Stellen Sie in den Eigenschaften des Textfelds eine Schriftart ein, die die Zeichen enthält, die Sie später verwenden möchten.
4. Fügen Sie die Zeichen der entsprechenden Schriftart, die Sie später verwenden möchten, in das Textfeld ein.

Nach dem Übersetzten des Projekts können Sie die im Text konfigurierten Zeichen in HMI-Variablen vom Typ WString/WChar verwenden.

Siehe auch

Asiatische Schriftarten auf das Bediengerät laden (Seite 2441)

2.2.3.2 Spezielle Konfigurationsmöglichkeiten

Adress-Multiplexen

Einleitung

Beim Adress-Multiplexen können Sie mit einer einzigen Variablen eine Vielzahl an Speicherplätzen im Adressbereich der Steuerung ansprechen. Sie können auf die Adressen schreibend und lesend zugreifen, ohne für jede einzelne Adresse eine Variable zu definieren.

Die eckigen Klammern zeigen an, dass die Adresse gemultiplext wird.

Multiplexen mit absoluter Adressierung

Beim Multiplexen mit absoluten Adressen projektieren Sie Variablen als Platzhalter für die anzusprechende Adresse in der Steuerung.

Wenn Sie z. B. auf eine Adresse des Formats "%DBx.DBWy" zugreifen möchten, sieht der Ausdruck für das Multiplexen folgendermaßen aus:

"%DB[HMITag1].DBW[HMITag2]"

Die Variable "HMITag1" versorgen Sie in Runtime mit dem gewünschten Wert für den Datenbaustein, den Sie adressieren möchten.

Die Variable "HMITag2" versorgen Sie in Runtime mit der gewünschten Adresse aus dem Datenbaustein.

Die Versorgung der Variablen geschieht z. B. mithilfe von Werten aus der Steuerung oder über ein Skript.

Das Multiplexen mit absoluten Adressen wird für folgende Steuerungen und Kommunikationstreiber unterstützt.

- SIMATIC S7-300/400
- SIMATIC S7-1200
- SIMATIC S7-1500

Das Multiplexen mit absoluten Adressen steht für Datenbausteine mit optimiertem Zugriff nicht zur Verfügung.

Multiplexen mit symbolischer Adressierung

Beim Multiplexen mit symbolischer Adressierung greifen Sie mithilfe einer Multiplexvariablen und einer Indexvariablen auf unterschiedliche Array-Elemente einer Array-Variablen in einem Datenbaustein der verbundenen Steuerung zu.

Zuerst verbinden Sie ein Element der gewünschten Array-Variablen in einem Datenbaustein mit der Multiplexvariablen im HMI-Gerät. Danach ersetzen Sie den Array-Index, der in der symbolischen Adresse am weitesten links steht, durch die Indexvariable.

Wenn Sie z. B. auf unterschiedliche Elemente der Array-Variablen "Arraytag_1" im Datenbaustein "Datablock_1" zugreifen möchten, sieht der Ausdruck für die symbolische Adressierung folgendermaßen aus:

"Datablock_1.Arraytag_1["HMITag_1"]

Mit der HMI-Variablen "HMITag_1" steuern Sie den Zugriff auf den Index der Array-Elemente. Die Variable versorgen Sie in Runtime mit dem Index des Array-Elements, auf das Sie jeweils zugreifen möchten.

Das Multiplexen mit symbolischer Adressierung ist nur verfügbar, wenn folgende Komponenten die symbolische Adressierung unterstützen:

- das Bediengerät
- die Steuerung
- der Kommunikationstreiber

Die symbolische Adressierung wird von folgenden Kommunikationstreibern unterstützt:

- SIMATIC S7-1200
- SIMATIC S7-1500

Für das Multiplexen mit symbolischer Adressierung können auch strukturierte HMI-Variablen mit benutzerdefinierten PLC-Datentypen (UDT) sowie Arrays verwendet werden.

Das Multiplexen ist unter folgenden Bedingungen möglich:

- Der Pfad der verbundenen PLC-Variablen enthält mindestens einen Array-Index.
- Das Array, das im Pfad der PLC-Variablen ganz links steht, ist eindimensional.
- Wenn mehrere Arrays im Pfad der PLC-Variablen enthalten sind, ist nur das Array ganz links für das Multiplexen verfügbar.
- Die Indexvariable muss vom Datentyp "USInt", "UInt", "UDInt", "SInt", "Int" oder "DInt" sein.

Das folgende Bild zeigt das Multiplexen mit symbolischer Adressierung mithilfe von einer strukturierten HMI-Variablen mit benutzerdefinierten PLC-Datentypen (UDT). Das Datenbausteinelement "db1.udt3Arr[0]" ist mit der HMI-Variablen "db1_udt3Arr{0}" verbunden. Die symbolische Adressierung der HMI-Variablen "db1_udt3Arr{0}" wird durch die HMI-Variable "index" gemultiplext:

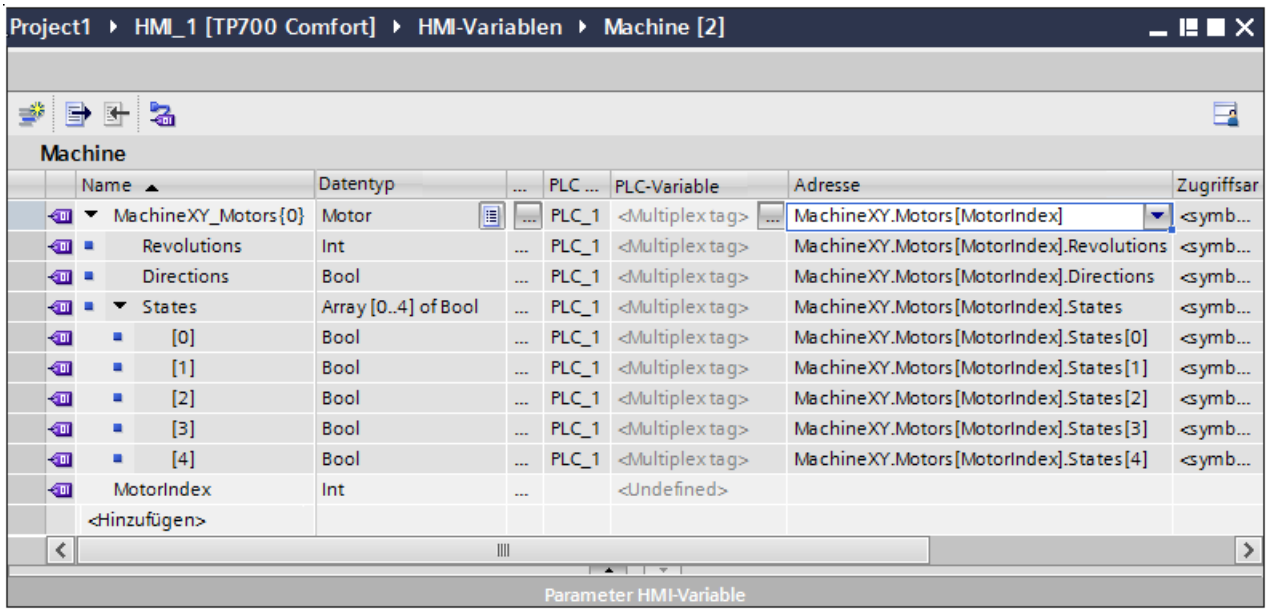

Folgende Variablen können nicht als Indexvariablen für das Multiplexen mit symbolischer Adressierung verwendet werden:

- HMI-Variablen, die nicht vom Datentyp "USInt", "UInt", "UDInt", "SInt", "Int" oder "DInt" sind
- HMI-Variablen, die bereits gemultiplext sind
- Array-Elemente
- HMI-Variablen, die Elemente von strukturierten HMI-Variablen sind

Hinweis

In Runtime wird das erste Array-Element immer mit dem Indexwert "0" adressiert. Lassen Sie Ihr Array-Element auch in den Datenbausteinen mit dem Index "0" beginnen.

Hinweis

Es ist möglich, die HMI-Variablen, die für das Multiplexen mit symbolischer Adressierung verwendet werden, nach MS Excel zu exportieren. Sie können solche HMI-Variablen auch aus MS Excel in eine HMI-Variablentabelle importieren, die alle HMI-Variablen des HMI-Bediengeräts enthält.

Siehe auch

Adress-Multiplexen projektieren mit absoluter Adressierung (Seite 579)

Adress-Multiplexen projektieren mit symbolischer Adressierung (Seite 582)

Aktualisieren des Variablenwerts in Runtime (Seite 588)

Indirekte Adressierung von Variablen (Seite 585)

Variable indirekt adressieren (Seite 586)

Variable zum Auslösen von Funktionen nutzen (Seite 587)

Erfassungszyklus einer Variable festlegen (Seite 589)

Adress-Multiplexen projektieren mit absoluter Adressierung

Einleitung

Beim Adress-Multiplexen greifen Sie mithilfe von wenigen Variablen effizient auf unterschiedliche Adressen in der Steuerung zu. An Stelle der absoluten Adresse in der Steuerung verwenden Sie Variablen, um die Adresse in Runtime verändern zu können.

Voraussetzung

- Die Variable für das Adress-Multiplexen ist angelegt und mit der Steuerung verbunden.
- Das Eigenschaftsfenster für diese Variable ist geöffnet.

Vorgehensweise

1. Selektieren Sie in der Variablentabelle die Variable für das Adress-Multiplexen und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein". Die allgemeinen Eigenschaften der Variable werden angezeigt.

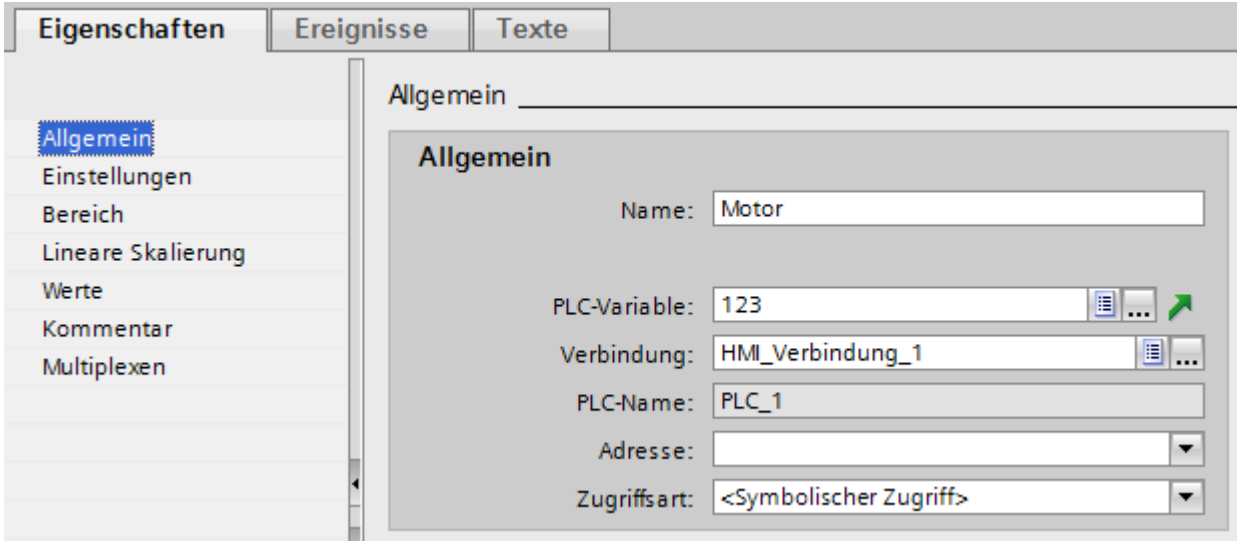

2. Wählen Sie für dieses Beispiel den Datentyp "Int".
3. Wählen Sie die Zugriffsart "Absolute Adressierung".

4. Klicken Sie auf die Auswahlschaltfläche im Feld "Adresse". Der Adressdialog wird geöffnet.

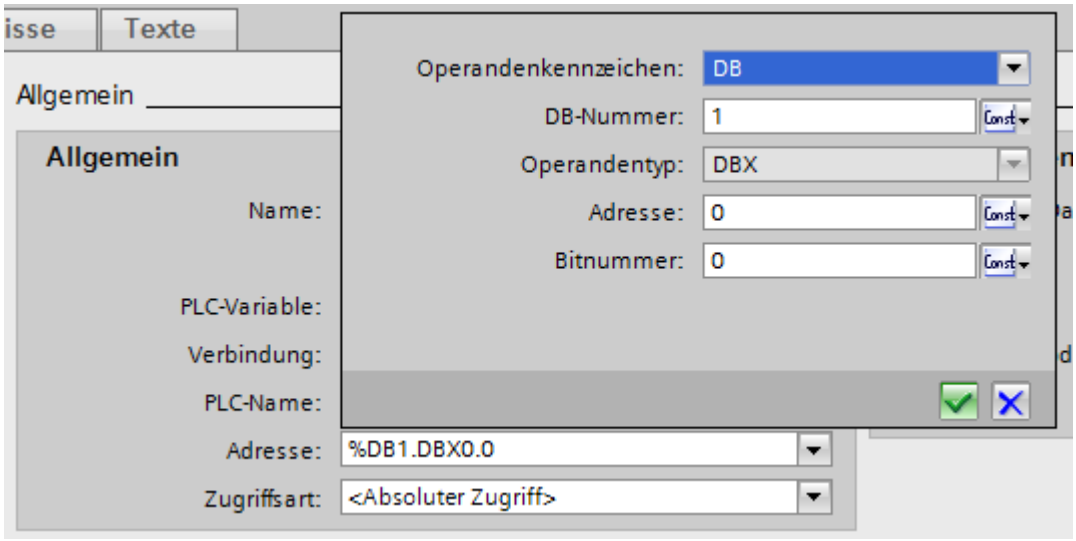

5. Klicken Sie auf die Auswahlschaltfläche im Feld "DB-Nummer" und wählen Sie den Eintrag "HMI-Variable".

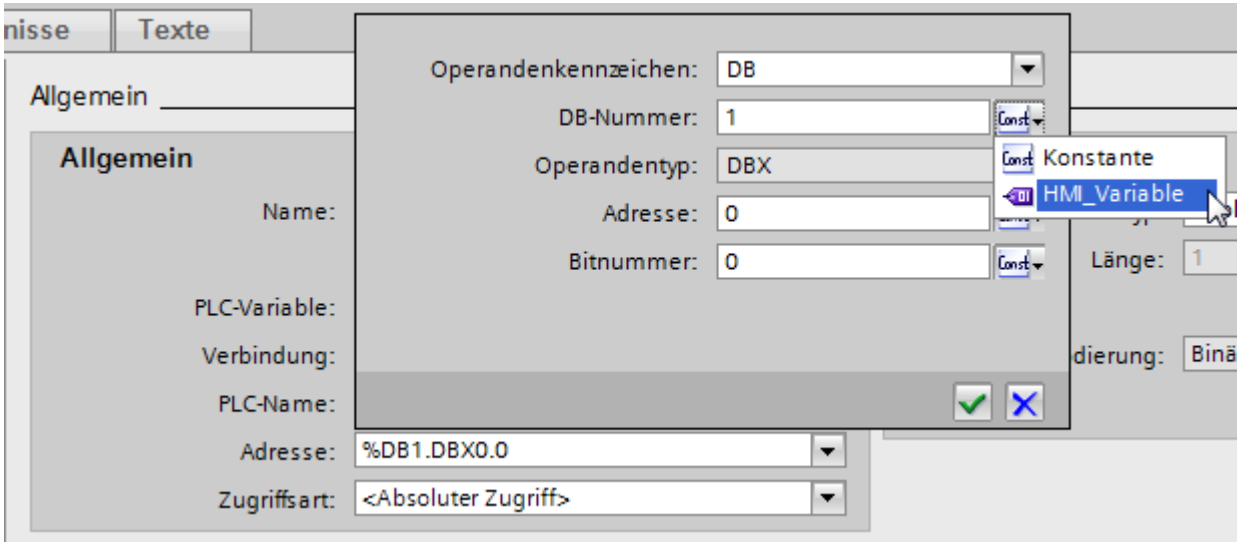

6. Klicken Sie im Feld "DB-Nummer" auf die Schaltfläche ... und wählen Sie in der Objektliste eine Variable für die DB-Nummer aus. Alternativ legen Sie mithilfe der Objektliste eine neue Variable an. Übernehmen Sie die Variable durch Klicken auf die Schaltfläche .

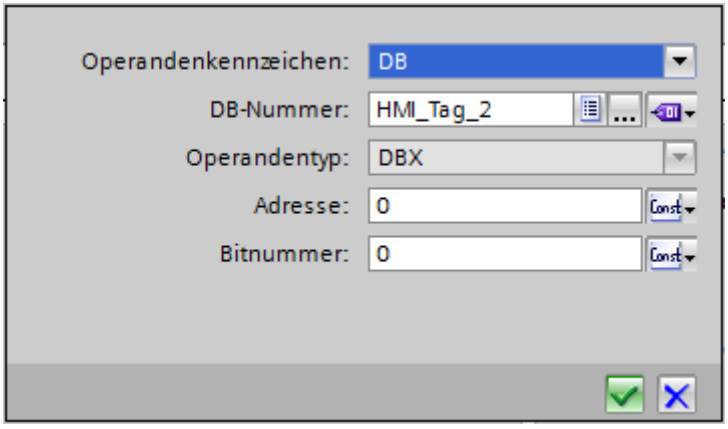

7. Wiederholen Sie die Schritte 3 und 4 für das Feld "Adresse" und projektieren Sie eine weitere Variable für den Aufruf des Adressbereichs im Datenbaustein.

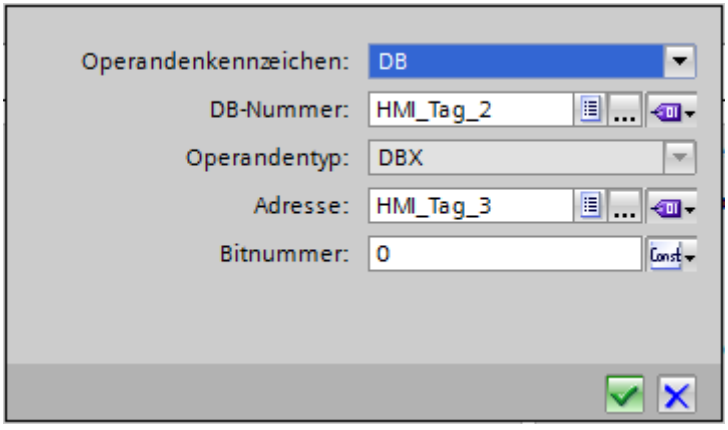

Die Auswahlmöglichkeiten im Adress-Dialog sind abhängig vom gewählten Datentyp der Multiplex-Variable. Der Adressdialog bietet nur Adresseinstellungen an, die mit dem gewählten Datentyp projektierbar sind.

Ergebnis

In Runtime wird mit der Multiplex-Variablen auf den Speicherbereich zugegriffen, dessen Adresse die Variable aktuell enthält. Mit der Variable im Feld DB-Nummer steuern Sie den Zugriff auf den Datenbaustein. Mit der Variable im Feld "Adresse" steuern Sie den Zugriff auf die Adresse im gewählten Datenbaustein.

Hinweis

Der Wert im Speicherbereich wird erst mit dem nächsten Aktualisierungszyklus der angesprochenen Variablen gelesen.

Wenn Sie z. B. in einem Skript eine Multiplex-Variable verwenden, greifen Sie nicht unmittelbar nach Änderung des Speicherbereichs auf dessen Inhalt zu.

Siehe auch

Adress-Multiplexen (Seite 576)

Adress-Multiplexen projektieren mit symbolischer Adressierung (Seite 582)

Adress-Multiplexen projektieren mit symbolischer Adressierung

Einleitung

Beim Adress-Multiplexen greifen Sie mithilfe von wenigen Variablen effizient auf unterschiedliche Adressen in der Steuerung zu. An Stelle der symbolischen Adresse in der Steuerung verwenden Sie Variablen, um die Adresse in Runtime verändern zu können.

Voraussetzung

- Die Variable für das Adress-Multiplexen ist angelegt.
- Das Eigenschaftsfenster für diese Variable ist geöffnet.
- In der verbundenen Steuerung ist ein Datenbaustein mit einer Array-Variable angelegt.
- Der Datenbaustein wurde übersetzt.

Vorgehensweise

1. Selektieren Sie in der Variablentabelle die Variable für das Adress-Multiplexen und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein". Die allgemeinen Eigenschaften der Variable werden angezeigt.

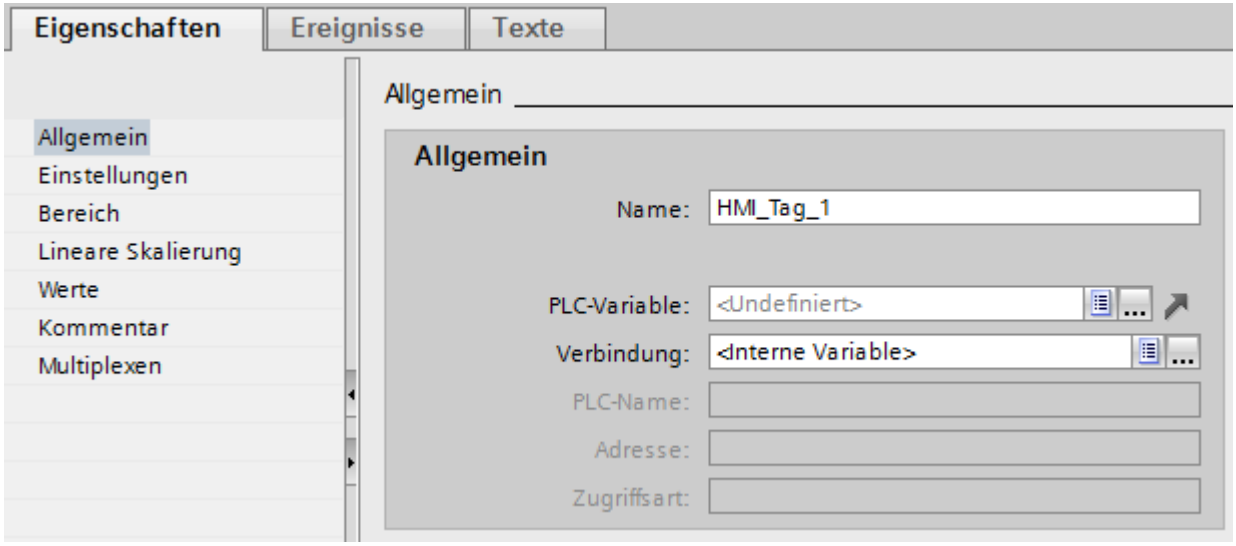

2. Wählen Sie über das Feld "Verbindung" die Verbindung zur Steuerung aus.

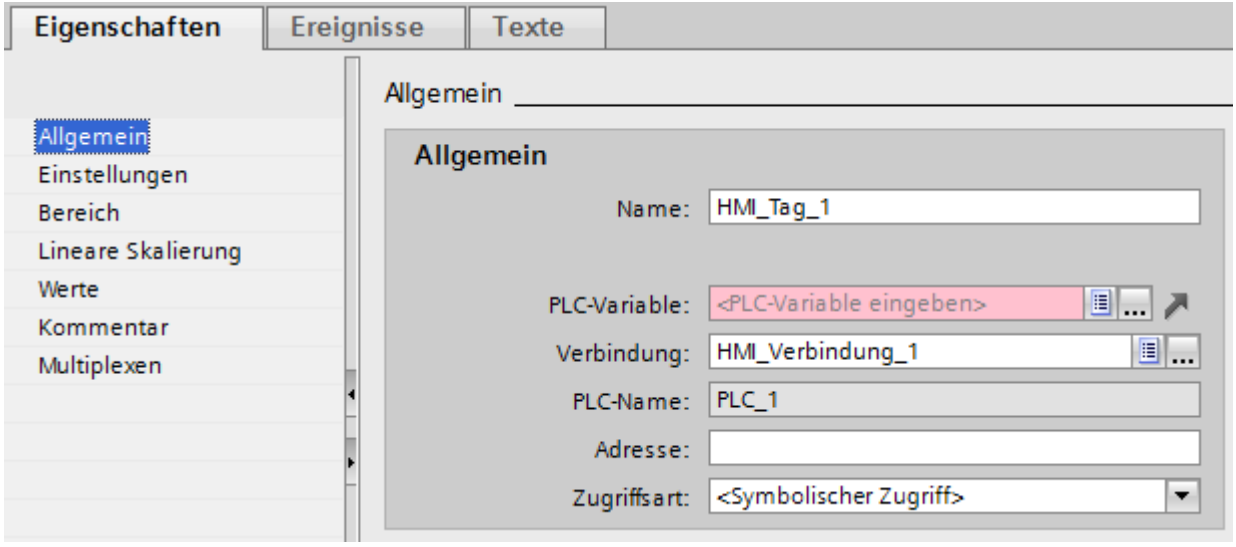

3. Wählen Sie die Zugriffsart "Symbolischer Zugriff".

4. Navigieren Sie über das Feld "PLC-Variable" zu dem Datenbaustein der Steuerung und wählen Sie ein Array-Element der Arrayvariable aus.

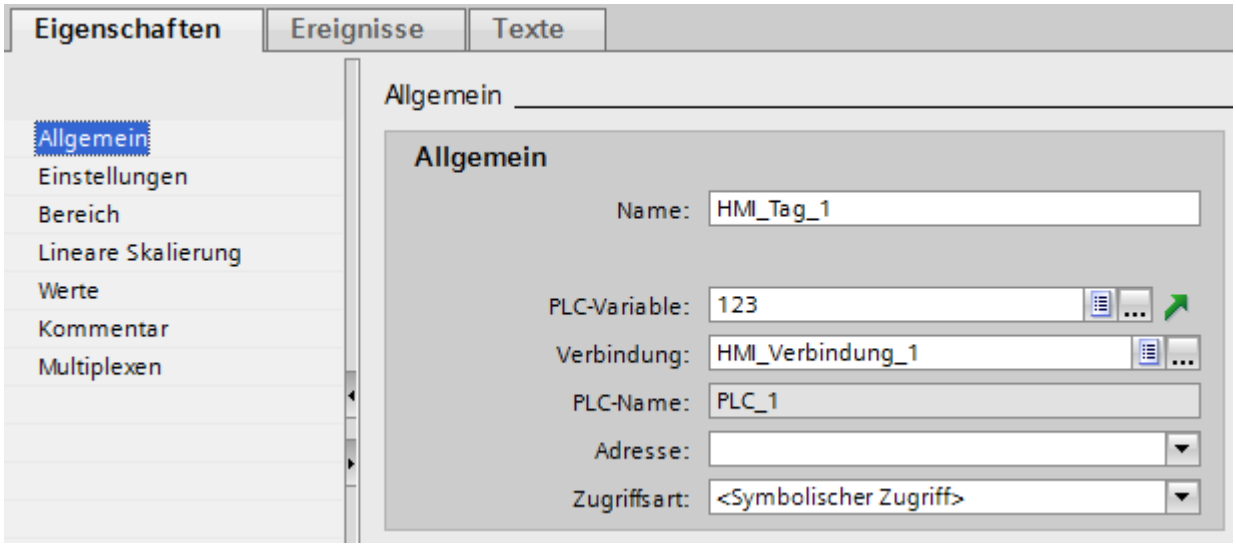

5. Klicken Sie im Feld "Adresse" auf die Auswahlschaltfläche. Der Adressdialog wird geöffnet.
6. Klicken Sie im Feld "Indexvariable" auf die Auswahlschaltfläche und wählen Sie den Eintrag "HMI-Variable".

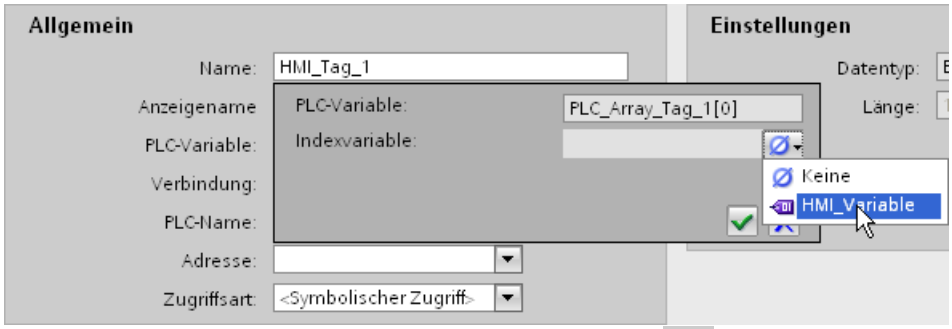

7. Klicken Sie im Feld "Indexvariable" auf die Schaltfläche ... und wählen Sie in der Objektliste eine Variable für den Array-Index aus. Alternativ legen Sie mithilfe der Objektliste eine neue Variable an. Übernehmen Sie die Variable durch Klicken auf die Schaltfläche ✓.

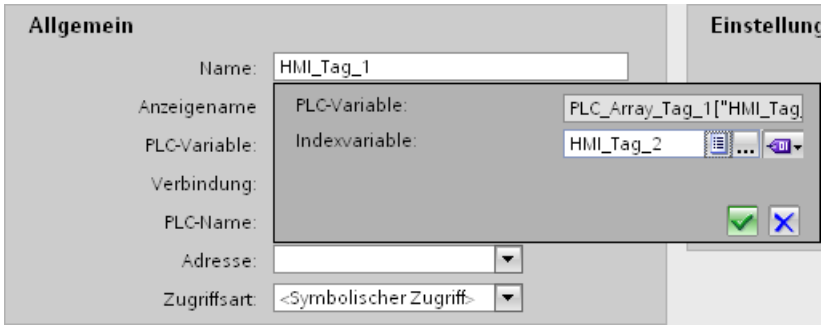

Ergebnis

In Runtime wird auf das Array-Element zugegriffen, dessen Indexwert in der Indexvariable enthalten ist.

Siehe auch

Adress-Multiplexen (Seite 576)

Adress-Multiplexen projektieren mit absoluter Adressierung (Seite 579)

Indirekte Adressierung von Variablen

Prinzip

Beim Multiplexen, einer Art indirekte Adressierung, wird die genutzte Variable erst zur Laufzeit ermittelt. Für die Multiplexvariable wird eine Liste von Variablen festgelegt. Aus der Liste von Variablen wird die relevante Variable in Runtime ausgewählt. Die Auswahl der Variable hängt vom Wert der Indexvariablen ab.

In Runtime wird zunächst der Wert der Indexvariable ausgelesen. Dann wird auf die Variable zugegriffen, die an der entsprechenden Stelle der Variablenliste steht.

Anwendungsbeispiel

Mit der indirekten Adressierung können Sie folgendes Szenario projektieren:

Der Bediener wählt aus einer Auswahlliste eine von mehreren Maschinen aus. Abhängig von der Auswahl des Bedieners wird ein Prozesswert der Maschine in einem Ausgabefeld angezeigt.

Um ein solches Szenario zu projektieren, projektieren Sie an ein symbolisches E/A-Feld die Indexvariable. An ein E/A-Feld projektieren Sie die Multiplexvariable. Die Variablenliste der Multiplexvariable projektieren Sie parallel zum Aufbau der Auswahlliste.

Wenn der Bediener eine andere Maschine auswählt, ändert sich dadurch der Wert der Indexvariable. Das Auswahlfeld zeigt dann den Inhalt der Variablen an, die in der Variablenliste der Multiplexvariable zum neuen Indexwert gehört.

Siehe auch

Variable indirekt adressieren (Seite 586)

Adress-Multiplexen (Seite 576)

Variable indirekt adressieren

Einleitung

Bei der indirekten Adressierung wird die genutzte Variable erst zur Laufzeit ermittelt. Anstelle einer einzelnen Variable wird eine Liste von Variablen festgelegt. Die Listeneinträge bestehen aus einem Indexwert und dem Namen der Variablen, die genutzt werden soll. Über eine Indexvariable steuern Sie, auf welchen Eintrag der Variablenliste zugegriffen wird.

Hinweis

Datentyp der Indexvariablen

Wenn Sie in der Variablenliste Variablen mit unterschiedlichen Datentypen verwenden, werden die Eigenschaften des größten Datentyps verwendet. Wenn Sie z. B. Variablen vom Datentyp INT und REAL verwenden, werden die Werte in Gleitkommazahlen angezeigt.

Voraussetzung

- Die Variable, die Sie für die indirekte Adressierung verwenden wollen, ist angelegt.
- Die Indexvariable ist angelegt.
- Die Variablen, die in der Variablenliste enthalten sein sollen, sind angelegt.
- Das Inspektorfenster mit den Eigenschaften für Variablen ist geöffnet.

Vorgehensweise

Um Variablen indirekt zu adressieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Multiplexen".
2. Aktivieren Sie die Option "Multiplexen", damit die indirekte Adressierung eingeschaltet wird. Mit dieser Option schalten Sie die indirekte Adressierung zeitweilig aus, z. B. zu Testzwecken. Bereits vorgenommene Einstellungen für die indirekte Adressierung bleiben erhalten.
3. Wählen Sie im Feld "Indexvariable" die vorgesehene Indexvariable aus oder definieren Sie eine neue Variable mithilfe der Objektliste.
4. Klicken Sie in der Variablenliste auf den ersten Eintrag der Spalte "Variablen".
5. Wählen Sie eine Variable als Listeneintrag aus oder erstellen Sie eine neue Variable mithilfe der Objektliste.
   Der Eintrag in der Spalte "Index" wird automatisch erzeugt.
6. Wiederholen Sie Schritt 5 für alle Variablen, die Sie in die Variablenliste aufnehmen wollen.
7. Bei Bedarf ändern Sie die Reihenfolge der Listeneinträge mit Drag&Drop.

Ergebnis

In Runtime wird dynamisch auf die Variable aus der Variablenliste zugegriffen, deren Index gleich dem Wert der Indexvariablen ist.

Siehe auch

Indirekte Adressierung von Variablen (Seite 585)

Adress-Multiplexen (Seite 576)

Variable zum Auslösen von Funktionen nutzen

Einleitung

Variablenwerte können Sie als auslösendes Ereignis für eine Aktion in Runtime nutzen. Um eine Aktion in Runtime zu starten, projektieren Sie eine Funktionsliste an eine Variable. In die Funktionsliste nehmen Sie eine oder mehrere Systemfunktionen oder VB-Skripte auf. Die Funktionsliste wird beim Eintritt des projektierten Ereignisses abgearbeitet.

Folgende Ereignisse sind an einer Variablen verfügbar:

- Wertänderung der Variablen
  Die Funktionsliste wird bei jeder Änderung des Variablenwerts abgearbeitet.
  Wenn die Variable als Arrayvariable definiert ist, dann wird die Funktionsliste bei jeder Änderung eines Arrayelements abgearbeitet.
- Überschreiten des oberen Grenzwerts der Variablen
  Die Funktionsliste wird bei Überschreiten des oberen Grenzwerts abgearbeitet.
- Unterschreiten des unteren Grenzwerts der Variablen
  Die Funktionsliste wird bei Unterschreiten des unteren Grenzwerts abgearbeitet.

Voraussetzung

- Die Variable, deren Wert Sie als Ereignis nutzen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

Vorgehensweise

Um eine Funktionsliste an eine Variable zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" das Ereignis, für das Sie eine Funktionsliste erstellen wollen.
   Die zum selektierten Ereignis gehörende Funktionsliste wird angezeigt.
2. Klicken Sie auf "<Funktion hinzufügen>". In der zweiten Tabellenspalte wird eine Auswahlschaltfläche angezeigt.
3. Klicken Sie auf die Auswahlschaltfläche und wählen Sie eine Systemfunktion aus.
4. Legen Sie die Parameterwerte fest.

Alternativ wählen Sie in der Funktionsliste anstelle einer Systemfunktion ein Skript zur Abarbeitung aus. Der Eintrag "VB-Skripte" wird nur angezeigt, wenn im Projekt ein Skript verfügbar ist.

Ergebnis

Wenn das projektierte Ereignis in Runtime eintritt, wird die Funktionsliste oder das Skript abgearbeitet.

Siehe auch

Adress-Multiplexen (Seite 576)

Aktualisieren des Variablenwerts in Runtime

Einleitung

Variablen enthalten Prozesswerte, die sich im Verlauf von Runtime ändern. Bei internen und externen Variablen werden Wertänderungen auf unterschiedliche Weise ausgelöst.

Prinzip

Beim Start von Runtime ist der Wert einer Variable gleich ihrem Startwert. In Runtime kommt es zu Wertänderungen der Variablen.

In Runtime haben Sie folgende Möglichkeiten, den Wert einer Variable zu ändern:

- Durch Wertänderung einer externen Variablen in der Steuerung.
- Durch Eingabe, z. B. in einem E/A-Feld.
- Durch Ausführen einer Systemfunktion zur Wertänderung, z. B. der Systemfunktion "SetzeWert".
- Durch Wertzuweisung in einem Skript.

Aktualisierung des Werts von externen Variablen

Der Wert einer externen Variable wird folgendermaßen aktualisiert:

- Zyklisch im Betrieb
  Wenn Sie die Erfassungsart "Zyklisch im Betrieb" wählen, wird die Variable in Runtime aktualisiert, so lange sie in einem Bild angezeigt oder so lange sie archiviert wird. Der Erfassungszyklus bestimmt, in welchem Zeitintervall der Wert der Variable am Bediengerät aktualisiert wird. Die zyklische Erfassung kann auf einem vorgegebenen oder auf einem von Ihnen selbst definierten Zyklus beruhen.
- Zyklisch fortlaufend
  Wenn Sie die Erfassungsart "Zyklisch fortlaufend" wählen, wird die Variable in Runtime ständig aktualisiert, auch wenn sie sich nicht im aufgeschlagenen Bild befindet. Die Einstellung wird z. B. für Variablen aktiviert, an deren Wertänderung eine Funktionsliste projektiert ist.
  Verwenden Sie "Zyklisch fortlaufend" nur für Variablen, die tatsächlich immer aktualisiert werden müssen. Häufige Lesevorgänge führen zu einer höheren Kommunikationslast.
- Auf Anforderung
  Wenn Sie die Erfassungsart "Auf Anforderung" wählen, wird die Variable nicht zyklisch aktualisiert. Die Aktualisierung erfolgt nur auf Anforderung, z. B. durch die Systemfunktion "Aktualisiere Variable" oder durch ein Skript.

Siehe auch

Adress-Multiplexen (Seite 576)

Erfassungszyklus einer Variable festlegen

Einleitung

Der Wert einer externen Variable kann in Runtime durch die Steuerung geändert werden, mit der die Variable verknüpft ist. Damit das Bediengerät von einer Wertänderung durch die Steuerung erfährt, muss der Variablenwert am Bediengerät aktualisiert werden. Die Aktualisierung erfolgt in regelmäßigen Abständen, solange die Variable im Prozessbild angezeigt wird oder archiviert wird. Eine regelmäßige Aktualisierung wird durch einen Erfassungszyklus gesteuert. Die Aktualisierung kann aber auch fortlaufend erfolgen.

Voraussetzung

- Die Variable, für die Sie einen Erfassungszyklus festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für die Variable ist geöffnet.

Vorgehen

Um einen Erfassungszyklus für eine Variable zu projektieren, gehen sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wenn Sie die Variable regelmäßig aktualisieren wollen, solange sie im Bild angezeigt oder archiviert wird, wählen Sie die Erfassungsart "Zyklisch im Betrieb".
   Oder:
   Wenn Sie die Variable regelmäßig aktualisieren wollen, auch wenn sie nicht im Bild angezeigt oder archiviert wird, wählen Sie die Erfassungsart "Zyklisch fortlaufend".
   Die Einstellung "Zyklisch fortlaufend" wird z. B. für eine Variable aktiviert, an deren Wertänderung eine Funktionsliste projektiert ist und die in keinem Bild direkt sichtbar ist.
3. Wählen Sie im Feld "Erfassungszyklus" die gewünschte Zykluszeit aus oder definieren Sie einen eigenen Erfassungszyklus mithilfe der Objektliste.

Alternativ konfigurieren Sie den Erfassungszyklus direkt im Arbeitsbereich der Variablentabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Hinweis

Verwenden Sie die Erfassungsart "Zyklisch fortlaufend" nur für Variablen, die tatsächlich fortlaufend aktualisiert werden müssen. Häufige Lesevorgänge erzeugen eine höhere Kommunikationslast.

Hinweis

Bei strukturierten HMI-Variablen kann die Erfassungsart sowohl für die jeweilige strukturierte HMI-Variable als auch für ihre einzelnen untergeordneten Elemente gewählt werden. Die Erfassungsart der Elemente der Array-Variablen kann nicht geändert werden. Den Elementen der Array-Variablen ist automatisch die Erfassungsart der übergeordneten Array-Variablen zugewiesen.

Wenn die Erfassungsart der strukturierten HMI-Variablen geändert wurde, wird sie für alle untergeordneten Elemente übernommen. Somit werden mögliche Änderungen der Erfassungsart an den untergeordneten Elementen überschrieben.

Ergebnis

Die projektierte Variable wird in Runtime mit dem gewählten Erfassungszyklus aktualisiert.

Siehe auch

Adress-Multiplexen (Seite 576)

2.2.3.3 Spezielle Konfigurationsmöglichkeiten

Ersatzwert festlegen

Einleitung

Sie können für eine Variable einen bestimmten Wert als Ersatzwert festlegen.

Im Bereich "Ersatzwert verwenden" wird festgelegt, wann WinCC diesen Ersatzwert verwendet. Der aktuelle Prozesswert aus dem Automatisierungssystem wird dann nicht übernommen.

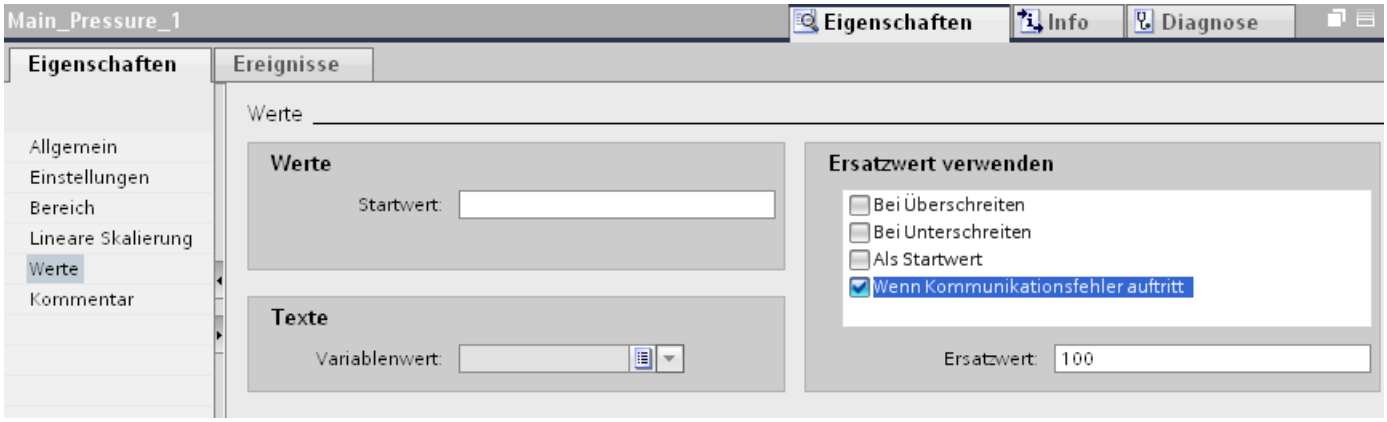

Sie können einen Ersatzwert für folgende Fälle festlegen:

- Als Startwert
  WinCC setzt den Ersatzwert, wenn das Projekt aktiviert wird und kein aktueller Prozesswert vorliegt.
- Bei Kommunikationsfehlern
  WinCC setzt den Ersatzwert, wenn die Verbindung zum Automatisierungssystem gestört ist und kein gültiger Prozesswert vorliegt.
- Maximum überschritten
  Wenn Sie eine Obergrenze für die Variable festgelegt haben, dann setzt WinCC den Ersatzwert, sobald der Prozesswert diese Obergrenze überschreitet.
- Minimum unterschritten
  Wenn Sie eine Untergrenze für die Variable festgelegt haben, dann setzt WinCC den Ersatzwert, sobald der Prozesswert diese Untergrenze unterschreitet.

Voraussetzung

- Die Variablentabelle ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für Variablen ist geöffnet.

Vorgehen

Um einen Ersatzwert zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Variablentabelle die gewünschte Variable und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Werte".
2. Wählen Sie im Bereich "Ersatzwert verwenden" aus, wann WinCC diesen Ersatzwert in der Variablen setzen soll.
3. Geben Sie im Feld "Ersatzwert" den gewünschten Ersatzwert ein.

Ergebnis

In Runtime wird die projektierte Variable mit dem Ersatzwert befüllt, sobald die gewählte Bedingung erfüllt ist.

Hinweis

Wenn Sie in einem E/A-Feld eine Obergrenze bzw. Untergrenze festgelegt haben, dann können Sie keinen Wert außerhalb dieser Grenzen eingeben.

WinCC ignoriert die fehlerhafte Eingabe und setzt darum auch keinen Ersatzwert. Den Ersatzwert setzt WinCC nur dann, wenn ein fehlerhafter Prozesswert gelesen wird.

Siehe auch

Erfassungszyklus einer Variable festlegen (Seite 592)

Grenzwerte einer Variable (Seite 567)

Indirekte Adressierung von Variablen (Seite 593)

Variable indirekt adressieren (Seite 594)

Erfassungszyklus einer Variable festlegen

Einleitung

Bei der Verwendung von Variablen legen Sie den Erfassungszyklus in Runtime Professional an dem Objekt fest, an dem die Variable verwendet wird.

Wenn Sie eine Variable archivieren, legen Sie den Erfassungszyklus an der verbundenen Archivvariable fest. Die verbundene Variable wird mit diesem Erfassungszyklus aktualisiert.

Wenn Sie eine Variable an einem Grafikobjekt verwenden, wird die Variable mit dem Erfassungszyklus des Grafikobjekts aktualisiert. Die Grafikobjekte, die in einem Bild enthalten sind, werden standardmäßig mit dem Aktalisierungszyklus des Bildes aktualisiert. Die Aktualisierungszyklen des Bildes und der enthaltenen Objekte projektieren Sie im Editor "Übersicht Dynamisierung". Im Editor "Übersicht Dynamisierung" werden alle in einem Bild verwendeten Objekte dargestellt, die einen Trigger benötigen. Für jedes Objekt wählen Sie den gewünschten Erfassungszyklus aus. Um eine performante Projektierung zu erreichen, sollten Sie den Erfassungszyklus einzelner Objekte nur bei Bedarf abweichend vom Bildzyklus projektieren.

Voraussetzung

- Sie haben ein Bild projektiert.
- Das Bild enthält ein Objekt, an das eine Variable projektiert ist.

Vorgehen

Um den Erfassungszyklus festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Bild und wählen Sie den Kontextmenübefehl "Übersicht Dynamisierungen".
   Der Editor "Übersicht Dynamisierung" wird geöffnet.

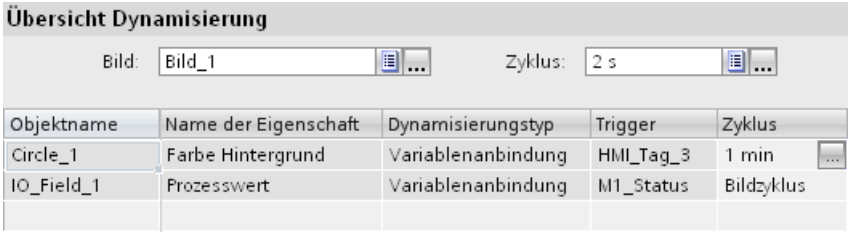

2. Den Aktualisierungszyklus für das Bild projektieren Sie im oberen Bereich des Dialogs "Übersicht Dynamisierung". Wählen Sie dort im Feld "Zyklus" den gewünschten Aktualisierungszyklus für das Bild aus oder definieren Sie einen eigenen Zyklus mithilfe der Objektliste. Die eingestellte Zykluszeit wirkt sich auf alle Objekte im Bild aus, deren Zyklus auf den Wert "Bildzyklus" eingestellt ist.
3. Wenn Sie für ein einzelnes Objekt eine andere Zykluszeit projektieren wollen, selektieren Sie in der Tabelle des Editors das gewünschte Objekt und wählen Sie im Feld "Zyklus" den Erfassungszyklus aus oder definieren Sie einen eigenen Zyklus mithilfe der Objektliste.

Wenn Sie den Erfassungszyklus lediglich für ein einzelnes Objekt ändern wollen, ist es ausreichend, das Objekt im Bild zu selektieren und den Kontextmenübefehl "Übersicht Dynamisierungen" zu wählen.

Ergebnis

Das projektierte Objekt wird in Runtime mit dem gewählten Zyklus aktualisiert.

Siehe auch

Ersatzwert festlegen (Seite 591)

Indirekte Adressierung von Variablen

Einleitung

Bei der indirekten Adressierung wird die genutzte Variable erst zur Laufzeit ermittelt. In WinCC Runtime Professional benutzen Sie für die indirekte Adressierung eine String-Variable. Weiterhin benötigen Sie eine Reihe von Variablen, deren Werte in Runtime dargestellt werden sollen. In Runtime wird in die String-Variable der Name der Variablen eingetragen, deren Wert angezeigt oder verarbeitet werden soll.

Anwendungsbeispiel

Der Bediener gibt den Namen einer von mehreren Maschinen in ein Eingabefeld ein. Abhängig von der Eingabe des Bedieners wird ein Prozesswert der Maschine in einem Ausgabefeld angezeigt.

Um ein solches Szenario zu projektieren, projektieren Sie an ein E/A-Feld die String-Variable mit Aktivierung der indirekten Adressierung. Dieses EA-Feld benutzen Sie für die Ausgabe der Werte. An ein weiteres E/A-Feld projektieren Sie dieselbe Variable ohne Aktivierung der indirekten Adressierung. Dieses EA-Feld benutzen Sie für die Eingabe der Maschinennamen. Zusätzlich projektieren Sie die Reihe der Variablen, deren Werte ausgegeben werden sollen.

Wenn der Bediener den Namen einer anderen Maschine eingibt, ändert sich dadurch der Wert der String-Variable für die indirekte Adressierung. Das Ausgabefeld zeigt dann den Inhalt der Variablen an, deren Name in der String-Variable enthalten ist.

Die Versorgung der String-Variable für die indirekte Adressierung können Sie auch über andere Mechanismen in WinCC projektieren, z. B. über ein Skript.

Siehe auch

Variable indirekt adressieren (Seite 594)

Ersatzwert festlegen (Seite 591)

Variable indirekt adressieren

Einleitung

Bei der indirekten Adressierung wird die genutzte Variable erst zur Laufzeit ermittelt. Anstelle einer einzelnen Variablen wird eine Reihe von Variablen festgelegt. In Runtime wird über den Variablennamen auf die einzelnen Variablen zugegriffen.

Hinweis

Dynamische Verschaltung von Variablen am Bildfenster

Um Variablen dynamisch zur Laufzeit am Bildfenster zu verschalten, verwenden Sie den Variablen-Präfix am Bildfenster oder indirekt adressierte Variablen. Vermeiden Sie die gleichzeitige Verwendung beider Methoden.

Voraussetzung

- Die Variable "Temperature_Motor", die Sie für die indirekte Adressierung verwenden wollen, ist angelegt.
- Eine Reihe von Variablen ist angelegt, z. B. die Variablen "Motor1", "Motor2" und "Motor3".
- Das Inspektorfenster mit den Eigenschaften für Variablen ist geöffnet.
- Ein Bild ist angelegt und im Editor "Bilder" geöffnet.

Vorgehensweise

Um Variablen indirekt zu adressieren, gehen Sie folgendermaßen vor:

1. Fügen Sie ein EA-Feld in das Bild ein und wählen Sie für dieses Beispiel im Infofenster unter "Eigenschaften > Eigenschaften > Allgemein" folgende Einstellungen:
   - Modus: Ausgabe
   - Anzeigeformat: Dezimal
2. Öffnen Sie im Feld "Variable" mithilfe der Auswahlschaltfläche die Objektliste, selektieren Sie die Variable "Temperature_Motor" und aktivieren Sie die Option "Indirekte Adressierung verwenden".

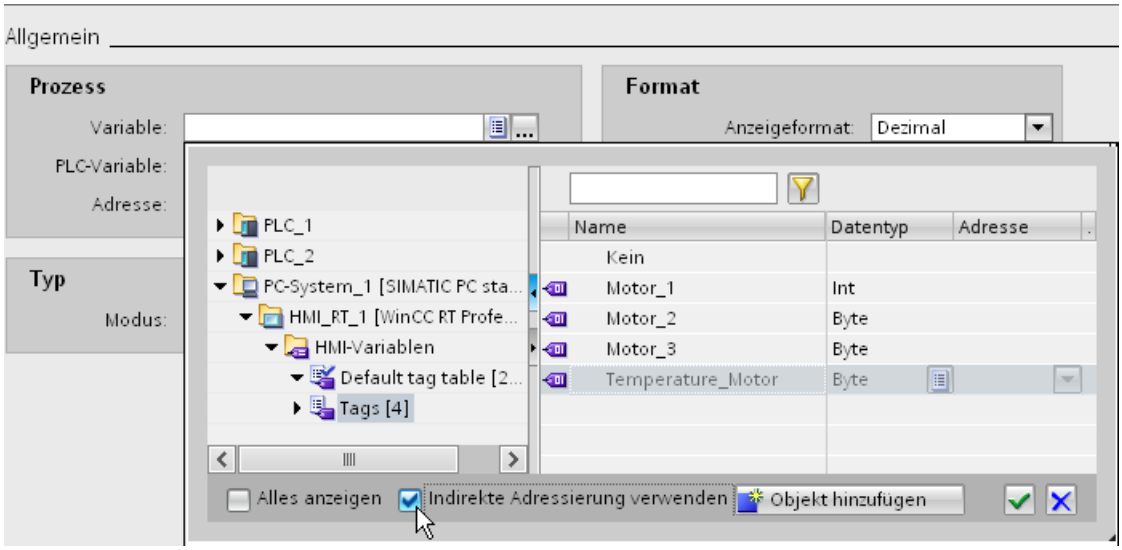

3. Die Variable "Temperature_Motor" ist für die indirekte Adressierung projektiert. Im Feld "Variable" wird vor dem Variablennamen der Zusatz "(indirekt)" angezeigt.

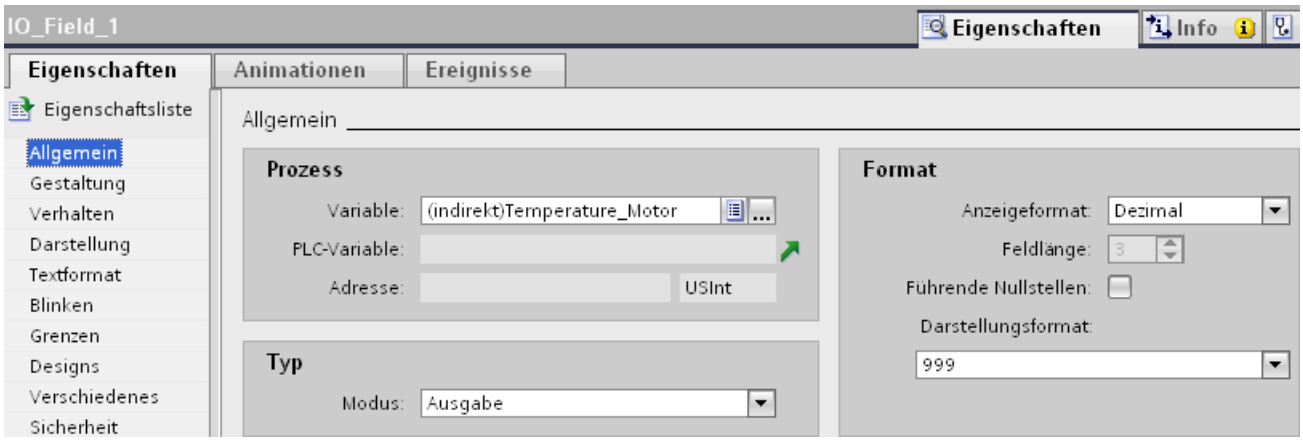

4. Fügen Sie ein weiteres EA-Feld in das Bild ein und wählen Sie für dieses Beispiel im Infofenster unter "Eigenschaften > Eigenschaften > Allgemein" folgende Einstellungen:
   - Modus: Eingabe
   - Anzeigeformat: Zeichenkette

5. Öffnen Sie im Feld "Variable" mithilfe der Auswahlschaltfläche die Objektliste, selektieren Sie die Variable "Temperature_Motor". Aktivieren Sie hier nicht die Option "Indirekte Adressierung verwenden".
6. Speichern Sie das Projekt.

Ergebnis

Wenn Sie in Runtime in das Eingabefeld den Namen einer der Variablen "Motor1", "Motor2" oder "Motor3" eingeben, wird im Ausgabefeld der Wert der jeweiligen Variable angezeigt.

Siehe auch

Indirekte Adressierung von Variablen (Seite 593)

Ersatzwert festlegen (Seite 591)

2.3 Arbeiten mit Arrays

2.3.1 Grundlagen zu Arrays

Definition

Als Array wird eine Datenliste mit Datenelementen eines einheitlichen Datentyps bezeichnet. Diese Datenliste wird als Arrayvariable oder kurz als Array bezeichnet. Die Datenelemente werden als Arrayelemente bezeichnet.

Die einzelnen Arrayelemente können über einen oder mehrere ganzzahlige Indizes adressiert werden. Die Eigenschaften eines jeden Arrayelements sind überwiegend gleich und können zentral an der Arrayvariablen projektiert werden.

HMI-Variablen

Name	Variablentabelle	Datentyp	Verbindung
HMI_Variable_1	Standard-Variablentabelle	Array [0..3] of Int	<Interne Variable>
[0]	Standard-Variablentabelle	Int	<Interne Variable>
[1]	Standard-Variablentabelle	Int	<Interne Variable>
[2]	Standard-Variablentabelle	Int	<Interne Variable>
[3]	Standard-Variablentabelle	Int	<Interne Variable>
[4]	Standard-Variablentabelle	Int	<Interne Variable>
<Hinzufügen>			

Vorteile

Mit nur einer Arrayvariablen fassen Sie viele gleichartige Arrayelemente strukturiert zusammen. Die gemeinsamen Eigenschaften der gleichartigen Arrayelemente können Sie zentral an der Arrayvariablen festlegen. Jedes einzelne Arrayelement verwenden Sie dann in der Projektierung grundsätzlich wie einzelne Variable mit demselben Datentyp.

Einschränkungen

Für die Verwendung von Arrays auf Bediengeräten, RT Advanced und RT Professional gibt es die folgenden Einschränkungen:

- Ein Array kann lediglich eine Dimension enthalten. Mehrdimensionale Arrays der Steuerungen werden auf Einzelvariable auf den HMI-Systemen abgebildet.
- Als Datentyp eines Arrays können nur elementare Datentypen verwendet werden, die nicht weiter unterstrukturiert sind. Es werden also nur skalare Arrays unterstützt.
- Der untere Index eines Arrays muss bei "0" beginnen. Bei anderen Untergrenzen auf den Steuerungen werden die Indizes entsprechend verschoben.
- Skalare Arrays unterstützen im Gegensatz zu Einzelvariablen keine Bereiche, keine lineare Skalierung und kein Variablen-Multiplexen.

Anwendungsbeispiele

Arrayvariablen verwenden Sie in folgenden Fällen:

- Um Prozesswerte in Profilkurven zusammenzufassen: Z. B. bilden Sie Prozesswerte in Kurven ab, die zu verschiedenen Zeitpunkten erfasst wurden.
- Um gezielt auf die in Kurven zusammengefassten Prozesswerte zuzugreifen: Z. B. geben Sie alle erfassten Werte der Profilkurve aus, indem Sie die Indexvariable schrittweise erhöhen.
- Um Bitmeldungen mit einer fortlaufenden Bitnummer zu projektieren.
- Um Maschinendatensätze als Ganzes in einer Rezeptur abzulegen.

Variablenzählung bei Arrays

- Skalare Arrays auf Bediengeräten und RT Advanced werden als eine Variable gezählt.
- Bei RT Professional wird jedes enthaltene Arrayelement separat gezählt.
- Multidimensionale Arrays der Steuerungen werden auf Einzelvariable desselben Datentyps abgebildet und wie diese gezählt.
- Strukturierte Arrays der Steuerung werden auf Einzelvariablen mit dem entsprechenden Datentyp abgebildet und wie diese gezählt.

Besonderheiten

Auf Bediengeräten sowie in RT Advanced werden bei Zugriff auf ein Element eines skalaren Arrays (mit Ausnahme von Arrays von String/WString) immer alle Arrayelemente auf einmal von der Steuerung gelesen oder geschrieben.

Bei RT Professional werden die Arrayelemente immer einzeln von und zur Steuerung gelesen und geschrieben.

Beispiel zu den Bediengeräten und RT Advanced:

- Eine Arrayvariable mit 100 Arrayelementen vom Datentyp "Real" ist projektiert.
- Wenn sich ein 4 Bytes großes Arrayelement ändert, werden 100 x 4 Bytes in die Steuerung geschrieben.

⚠ WARNUNG

Erhöhte Systemauslastung und Performanceeinbußen

Aufgrund der gleichzeitigen Übertragung aller Arrayelemente eines skalaren Arrays dauert speziell das Schreiben zur Steuerung bei Zugriff auf ein Element eines größeren Arrays länger als das Schreiben einer Variablen mit gleichem Datentyp. Berücksichtigen Sie dies bei der Auslegung der Kommunikation.

Beachten Sie auch, dass bei skalaren Arrays das Lesen von ganzen Arrayvariablen von der Steuerung in der Regel schneller ist als die Summe der Lesezugriffe auf die gleiche Anzahl von elementaren Variablen des gleichen Datentyps.

⚠ VORSICHT

Dateninkonsistenz bei Arrayvariablen

Wenn in einer skalaren Arrayvariablen der Wert eines einzelnen Elements geändert werden soll, wird das gesamte Array gelesen, geändert und als vollständiges Array wieder zurückgeschrieben. Änderungen an anderen Arrayelementen, die zwischenzeitlich in der Steuerung vorgenommen werden, werden beim Zurückschreiben überschrieben.

Achten Sie darauf, dass unterschiedliche Stellen, z. B. Bediengerät und Steuerung, nicht zur selben Zeit Werte in dieselbe Arrayvariable schreiben. Um die Arrayvariable mit der Steuerung zu synchronisieren, benutzen Sie z. B. die synchrone Übertragung von Rezepturdatensätzen.

Hinweis

Arrayvariablen als Listeneintrag von Multiplexvariablen

Arrayvariablen des Datentyps Char oder WChar können wie Variablen des Datentyps String verwendet werden.

Die Verwendung einer Arrayvariablen des Datentyps Char oder WChar als Listeneintrag einer Multiplexvariablen im Editor "HMI-Variablen" wird aber nicht unterstützt.

Verwendung in Skripten

Um die Performance zu erhalten, verwenden Sie in Skripten zum Ändern von skalaren Arrays immer interne Arrays.

1. Am Anfang des Skripts kopieren Sie das PLC-Array in das interne Array.
2. Während das interne Array durch das Skript bearbeitet wird, belasten Sie dadurch nicht die Kommunikation zur Steuerung.

⚠ VORSICHT

Dateninkonsistenz bei Arrayvariablen

Wenn in einer Arrayvariablen der Wert eines einzelnen Elements geändert werden soll, wird das ganze Array gelesen, geändert und als vollständiges Array wieder zurückgeschrieben. Änderungen an anderen Arrayelementen, die zwischenzeitlich in der Steuerung vorgenommen werden, werden beim Zurückschreiben überschrieben.

Achten Sie darauf, dass nicht zur selben Zeit unterschiedliche Stellen, z. B. Bediengerät und Steuerung, Werte in dieselbe Arrayvariable schreiben. Um die Arrayvariable mit der Steuerung zu synchronisieren, benutzen Sie z. B. die synchrone Übertragung von Rezepturdatensätzen.

Siehe auch

Arrayvariable anlegen (Seite 600)

Beispiele zu Arrays (Seite 602)

Grundlagen zu Variablen (Seite 537)

WinCC Runtime Advanced (Seite 2473)

Basic Panel (Seite 2455)

Mobile Panel (Seite 2462)

Comfort Panel (Seite 2467)

Basic Panel 2nd Generation (Seite 2458)

2.3.2 Arrayvariable anlegen

Einleitung

Um eine große Zahl von Variablen gleichen Datentyps zu projektieren, legen Sie eine Arrayvariable an. Die einzelnen Arrayelemente werden in einem fortlaufenden Adressbereich gespeichert.

Eine Arrayvariable können Sie als interne Variable oder als externe Variable anlegen.

Hinweis

HMI-Variablen mit PLC-Arrayvariablen verbinden

Wenn Sie eine Arrayvariabel als externe Variable anlegen wollen, dann projektieren Sie zuerst eine Arrayvariable in einem Datenbaustein der verbundenen Steuerung. Anschließend verbinden Sie die Arrayvariable mit einer HMI-Variable.

Hinweis

HMI-Arrayvariablen der folgenden Datentypen werden nur mit PLC-Arrayvariablen unterstützt:

- Char
- WChar
- String
- WString

HMI-Arrayvariablen anlegen

Um eine Arrayvariable anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die HMI-Variablentabelle.
2. Doppelklicken Sie in der HMI-Variablentabelle in der Spalte "Name" auf <Hinzufügen>. Eine neue HMI-Variable wird angelegt.

Hinweis

Sie können einzelne Elemente der Arrayvariablen über das Kontextmenü löschen.

HMI-Arrayvariablen konfigurieren

1. Klicken Sie in der HMI-Variablentabelle in der Spalte Datentyp auf und wählen Sie den Datentyp "Array".
2. Klicken Sie in der Spalte Datentyp auf .
   Der Dialog zur Projektierung des Arrays wird geöffnet.

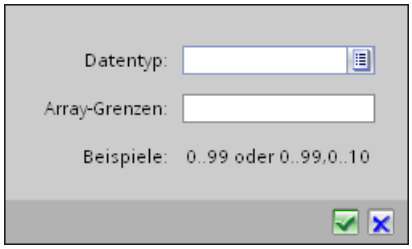

3. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp für die Arrayvariable.

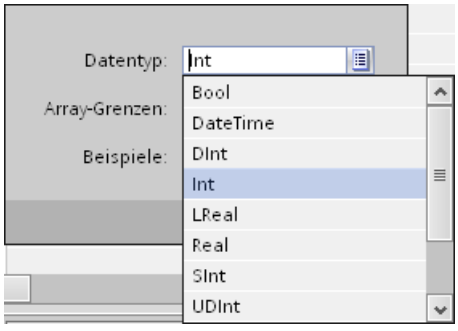

4. Legen Sie im Feld "Array-Grenzen" die Anzahl der Arrayelemente fest. Die untere Grenze muss mit "0" beginnen.
5. Klicken Sie auf . Die Einstellungen für das Array werden übernommen.
6. Wie bei Variablen mit elementaren Datentypen müssen Sie noch die Adresse der Arrayvariable in der PLC vergeben.
7. Speichern Sie das Projekt.

Ergebnis

Eine Arrayvariable ist angelegt. Die Eigenschaften der Arrayelemente werden von der übergeordneten Arrayvariablen übernommen.

Siehe auch

Grundlagen zu Arrays (Seite 597)

2.3.3 Beispiele zu Arrays

Einleitung

Arrayvariablen fassen viele Variablen, z. B. 100 Arrayelemente zusammen.

Die gemeinsamen Eigenschaften der gleichartigen Arrayelemente können Sie zentral an der Arrayvariablen festlegen. Jedes einzelne Arrayelement verwenden sie dann in der Projektierung grundsätzlich wie eine einzelne Variable mit demselben Datentyp. Arrayelemente können jedoch nicht als Parameter in Funktionslisten verwendet werden..

Arrayvariablen verwenden Sie als vollständige Arrays an folgenden Stellen:

- Im Editor "Meldungen"
- Im Editor "Rezepturen"
- Beim Adress-Multiplexen
- In Funktionslisten

Beispiele

Sie können eine PLC-Arrayvariable direkt aus der PLC-Variablentabelle in die HMI-Variablentabelle ziehen.

Alternativ projektieren Sie eine Arrayvariable separat mit der entsprechenden Anzahl Arrayelemente wenn Sie viele Variablen vom selben Datentyp haben.

- Greifen Sie auf die einzelnen Arrayelemente indirekt zu, z. B. über eine Indexvariable.
- Über die Indexvariable bedienen und beobachten Sie die Arrayelemente.

Siehe auch

Grundlagen zu Arrays (Seite 597)

2.4 Arbeiten mit Arrays

2.4.1 Grundlagen zu Arrays

Definition

Die einzelnen Arrayelemente können über einen oder mehrere ganzzahlige Indizes adressiert werden. Die Eigenschaften eines jeden Arrayelements sind überwiegend gleich und können zentral an der Arrayvariablen projektiert werden.

Als Array wird eine Datenliste mit Datenelementen eines einheitlichen Datentyps bezeichnet. Diese Datenliste wird als Arrayvariable oder kurz als Array bezeichnet. Die Datenelemente werden als Arrayelemente bezeichnet.

Default tag table

Name	Variablentabelle	Datentyp	Verbindung
HMI_Tag_1	Default tag table	Array [0..4] of Int	<Interne Variable>
[0]	Default tag table	Int	<Interne Variable>
[1]	Default tag table	Int	<Interne Variable>
[2]	Default tag table	Int	<Interne Variable>
[3]	Default tag table	Int	<Interne Variable>
[4]	Default tag table	Int	<Interne Variable>

Vorteile

Mit nur einer Arrayvariablen fassen Sie viele gleichartige Arrayelemente strukturiert zusammen. Die gemeinsamen Eigenschaften der gleichartigen Arrayelemente können Sie zentral an der Arrayvariablen festlegen. Jedes einzelne Arrayelement verwenden Sie dann in der Projektierung grundsätzlich wie einzelne Variable mit demselben Datentyp.

Einschränkungen

Für die Verwendung von Arrays gibt es folgende Einschränkungen:

- Ein Array kann lediglich eine Dimension enthalten. Mehrdimensionale Arrays der Steuerungen werden auf Einzelvariable auf den HMI-Systemen abgebildet.
- Der untere Index eines Arrays muss bei "0" beginnen. Bei anderen Untergrenzen auf den Steuerungen werden die Indizes entsprechend verschoben.
- Als Datentyp eines Arrays können nur elementare Datentypen verwendet werden, die nicht weiter unterstrukturiert sind. Es werden also nur skalare Arrays unterstützt.

- Skalare Arrays unterstützen im Gegensatz zu Einzelvariablen keine Bereiche, keine lineare Skalierung und kein Variablen-Multiplexen.
- Externe Arrayvariablen sind nur möglich mit den Steuerungen S7-1200 und S7-1500.

Hinweis

Synchronisation

Um Strukturkonflikte in Runtime zu vermeiden, synchronisieren Sie bei externen Arrayvariablen die Variablennamen und ersetzen Sie dabei die Trennzeichen "[" und "]".

Projektieren Sie dafür in den Runtime-Einstellungen die "Einstellungen für Variablen".

Variablenzählung bei Arrays

- Skalare Arrays auf Bediengeräten und RT Advanced werden als eine Variable gezählt.
- Bei RT Professional wird jedes enthaltene Arrayelement separat gezählt.
- Multidimensionale Arrays der Steuerungen werden auf Einzelvariable desselben Datentyps abgebildet und wie diese gezählt.
- Strukturierte Arrays der Steuerung werden auf Einzelvariablen mit dem entsprechenden Datentyp abgebildet und wie diese gezählt.

Besonderheiten

Auf Bediengeräten sowie in RT Advanced werden bei Zugriff auf ein Element eines skalaren Arrays (mit Ausnahme von Arrays von String/WString) immer alle Arrayelemente auf einmal von der Steuerung gelesen oder geschrieben.

Bei RT Professional werden die Arrayelemente immer einzeln von und zur Steuerung gelesen und geschrieben.

In Verbindung mit PLC-Datentypen (UDT) der Steuerungen S7-1200 und S7-1500 sind auch multidimensionale Arrays möglich.

Hinweis

Arrayvariablen als Listeneintrag von Multiplexvariablen

Arrayvariablen des Datentyps Char oder WChar können wie Variablen des Datentyps String verwendet werden.

Die Verwendung einer Arrayvariablen des Datentyps Char oder WChar als Listeneintrag einer Multiplexvariablen im Editor "HMI-Variablen" wird aber nicht unterstützt.

Beispiel zum Schreiben eines Arrayelements in die Steuerung

- Eine Arrayvariable mit 100 Arrayelementen vom Datentyp "Real" ist projektiert.
- Wenn sich ein 4 Bytes großes Arrayelement ändert, werden 4 Bytes in die Steuerung geschrieben.

Siehe auch

Einstellungen zum Synchronisieren (Seite 564)

2.4.2 Arrayvariable anlegen

Einleitung

Um eine große Zahl von Variablen gleichen Datentyps zu projektieren, legen Sie eine Arrayvariable an. Die einzelnen Arrayelemente werden in einem fortlaufenden Adressbereich gespeichert.

Eine Arrayvariable können Sie als interne Variable oder als externe Variable anlegen.

Hinweis

HMI-Variablen mit PLC-Arrayvariablen verbinden

Wenn Sie eine Arrayvariabel als externe Variable anlegen wollen, dann projektieren Sie zuerst eine Arrayvariable in einem Datenbaustein der verbundenen Steuerung. Anschließend verbinden Sie die Arrayvariable mit einer HMI-Variable.

Hinweis

HMI-Arrayvariablen der folgenden Datentypen werden nur mit PLC-Arrayvariablen unterstützt:

- Char
- WChar
- String
- WString

HMI-Arrayvariablen anlegen

Um eine Arrayvariable anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die HMI-Variablentabelle.
2. Doppelklicken Sie in der HMI-Variablentabelle in der Spalte "Name" auf <Hinzufügen>. Eine neue HMI-Variable wird angelegt.

Hinweis

Sie können einzelne Elemente der Arrayvariablen über das Kontextmenü löschen.

HMI-Arrayvariablen konfigurieren

1. Klicken Sie in der HMI-Variablentabelle in der Spalte Datentyp auf und wählen Sie den Datentyp "Array".
2. Klicken Sie in der Spalte Datentyp auf .
   Der Dialog zur Projektierung des Arrays wird geöffnet.

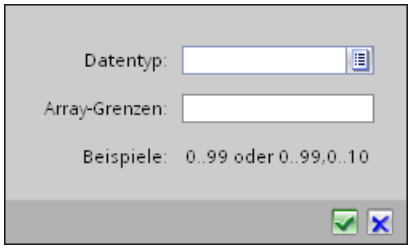

3. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp für die Arrayvariable.

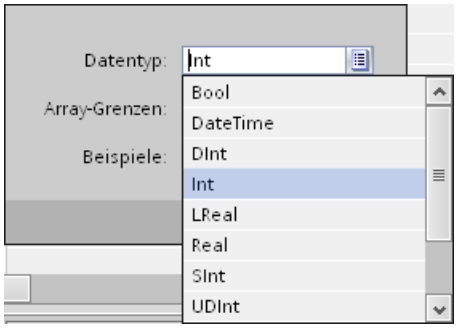

4. Legen Sie im Feld "Array-Grenzen" die Anzahl der Arrayelemente fest. Die untere Grenze muss mit "0" beginnen.
5. Klicken Sie auf . Die Einstellungen für das Array werden übernommen.
6. Wie bei Variablen mit elementaren Datentypen müssen Sie noch die Adresse der Arrayvariable in der PLC vergeben.
7. Speichern Sie das Projekt.

Ergebnis

Eine Arrayvariable ist angelegt. Die Eigenschaften der Arrayelemente werden von der übergeordneten Arrayvariablen übernommen.

2.5 Arbeiten mit Anwenderdatentypen

2.5.1 Grundlagen zu Anwenderdatentypen

Einleitung

Mit Anwenderdatentypen fassen Sie eine Anzahl verschiedener Variablen zusammen, die eine logische Einheit bilden. Ein Anwenderdatentyp legen Sie als Typ an und verwenden Instanzen dieses Typs im Projekt. Anwenderdatentypen sind projektzugehörige Daten und sind für alle HMI-Geräte des Projekts verfügbar.

Anwenderdatentypen unterscheiden sich in ihrer Zugehörigkeit zu einer Gerätefamilie. Anwenderdatentypen sind für folgende Gerätefamilien verfügbar:

- WinCC Runtime Advanced und Panels
- WinCC Runtime Professional

WinCC unterstützt auch die Anbindung von PLC-Datentypen (UDTs) als Anwenderdatentypen an HMITags.

Anwenderdatentypen unterscheiden sich zusätzlich durch ihre Verwendbarkeit mit einem bestimmten Kommunikationstreiber. Anwenderdatentypen sind für folgende Kommunikationstreiber verfügbar:

- SIMATIC S7-300/400
- SIMATIC S7-1200
- SIMATIC S7-1500

Anwenderdatentypen und Anwenderdatentyp-Elemente legen Sie in der Projektbibliothek an.

Prinzip

Die unterschiedlichen Zustände eines Motors lassen sich z. B. mit 6 einzelnen, boolschen Variablen beschreiben.

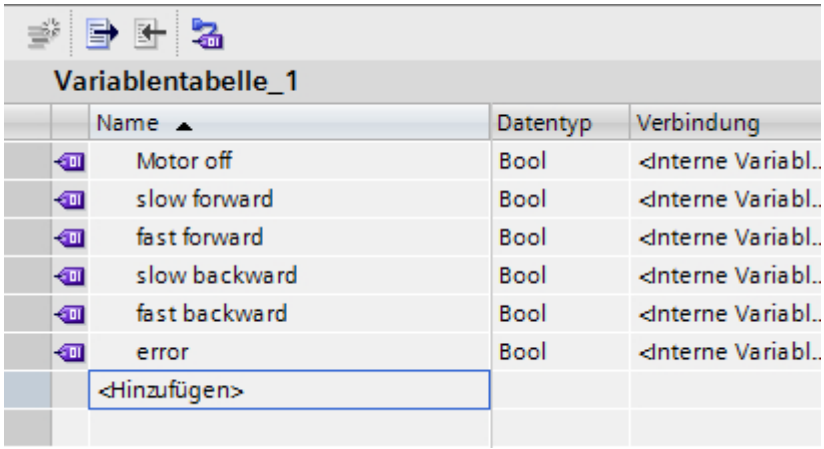

Soll der Gesamtzustand mit einer Variablen beschrieben werden, kann diese Variable auf Basis eines Anwenderdatentyps erstellt werden. Für jede der einzelnen boolschen Variablen legen Sie im Anwenderdatentyp ein Anwenderdatentyp-Element an.

Dieser Anwenderdatentyp kann dann einem Bildbaustein für den Motor als Ganzes zugewiesen werden. Die erzeugte und freigegebene Version des Anwenderdatentyps wird an der Variablen im Auswahlfeld "Datentyp" angezeigt.

Die projektierten Eigenschaften eines Anwenderdatentyps werden in den Instanzen des Anwenderdatentyps verwendet. Bei Bedarf ändern Sie einzelne Eigenschaften direkt an der Verwendungsstelle, z. B. an einer Variable. Die Änderung einer Eigenschaft an der Variablen wirkt sich nicht auf den erstellten Anwenderdatentyp aus.

ACHTUNG
Verfügbarkeit von Anwenderdatentypen Eine Anbindung von Anwenderdatentypen an Bildbausteine ist nur für WinCC Runtime Advanced und Panels möglich.

Lizenzregelung in Runtime

Wenn Sie externe Variablen vom Datentyp "Anwenderdatentyp" in der Instanz eines Bildbausteins verwenden, wird in Runtime jedes Anwenderdatentyp-Element als eine Variable gezählt.

Beispiel

Im Editor "Bilder" haben Sie zwei Bilder angelegt: Bild_1 und Bild_2.

Im Bild_1 sind 3 Instanzen eines Bildbaustein-Typs eingefügt. Im Bild_2 sind 4 Instanzen eines Bildbausteintyps eingefügt.

Jede Instanz eines Bildbausteins ist mit einer externen Variable vom Datentyp "Anwenderdatentyp" verbunden. Der Anwenderdatentyp enthält 10 Anwenderdatentyp-Elemente.

Bild 1: 3 Instanzen eines Bildbausteins * 10 Anwenderdatentyp-Elemente entspricht 30 externen Variablen = 30 PowerTags.

Bild 2: 4 Instanzen eines Bildbausteins * 10 Anwenderdatentyp-Elemente entspricht 40 externen Variablen = 40 PowerTags.

In Runtime werden für beide Bilder zusammen 70 PowerTags gezählt. Das gilt auch für die Anwenderdatentyp-Elemente, die nicht benötigt werden.

Siehe auch

Variable mit Datentyp Anwenderdatentyp anlegen (Seite 613)

Grundlagen zu Variablen (Seite 537)

Anwenderdatentyp-Elemente anlegen (Seite 610)

2.5.2 Anwenderdatentyp erzeugen

Einleitung

Einen Anwenderdatentyp legen Sie in der Projektbibliothek an.

Voraussetzung

- Ein Projekt ist geöffnet.
- Die Ansicht der Task Card "Bibliotheken" oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um einen Anwenderdatentyp zu erzeugen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Task Card "Bibliotheken".
2. Doppelklicken Sie auf den Eintrag "Projektbibliothek". Die Ordnerstruktur der Projektbibliothek wird geöffnet.
3. Klicken Sie auf den Eintrag "Neuen Typ hinzufügen".
   Ein Dialog wird geöffnet.

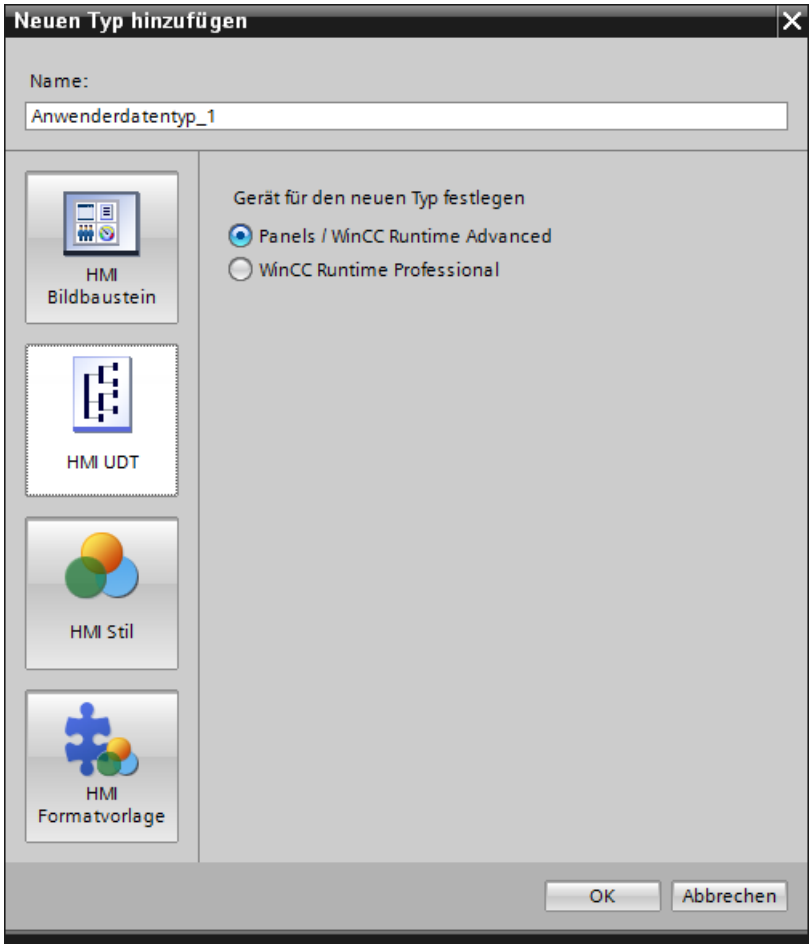

4. Klicken Sie im Dialog auf die Schaltfläche "HMI UDT"
5. Wählen Sie im Bereich "Gerät für neuen Typ festlegen" das Bediengerät aus, auf dem der Anwenderdatentyp verwendet wird.
6. Geben Sie im Feld "Name" einen aussagekräftigen Namen ein.
7. Übernehmen Sie die Einstellungen mit "OK". Der Anwenderdatentyp wird erstellt. Die Bibliotheksansicht wird geöffnet.

Ergebnis

Ein Anwenderdatentyp mit den projektierten Eigenschaften wird erzeugt. Die Version 0.0.1 des Anwenderdatentypen wird erstellt und erhält den Status "In Bearbeitung".

Legen Sie im nächsten Schritt die benötigten Anwenderdatentyp-Elemente an.

Bevor Sie die Version eines Anwenderdatentypen z. B. an einer Variablen verwenden, muss die Version freigegeben werden.

2.5.3 Anwenderdatentyp-Elemente anlegen

Einleitung

Anwenderdatentyp-Elemente legen Sie in der Bibliotheksansicht fest. Im Arbeitsbereich fügen Sie in der Tabelle "HMI-Anwenderdatentypen" Elemente hinzu oder löschen sie.

Voraussetzung

- Die Bibliotheksansicht ist geöffnet.
- Ein Anwenderdatentyp ist angelegt und im Editor geöffnet.

Vorgehen

1. Wählen Sie einen Kommunikationstreiber für den Anwenderdatentyp.

- Wenn Sie den Eintrag <Interne Kommunikation> wählen, können Sie den Anwenderdatentyp nur internen Variablen als Datentyp zuweisen.
- Wenn Sie als Kommunikationstreiber eine Verbindung zu einer Steuerung wählen, können Sie den Anwenderdatentyp nur Variablen mit Verbindung zu dieser Steuerung als Datentyp zuweisen.
- Der eingestellte Kommunikationstyp gilt für alle Anwenderdatentyp-Elemente eines Anwenderdatentyps. In einem Anwenderdatentyp für die Gerätefamilie "WinCC Runtime Professional" können Sie für jedes Anwenderdatentyp-Element festlegen, ob der projektierte Treiber zur Steuerung oder die interne Kommunikation verwendet wird.

1. Doppelklicken Sie in der Tabelle in der Spalte "Name" auf "Hinzufügen". Dem Anwenderdatentyp wird ein neues Anwenderdatentyp-Element hinzugefügt.
2. Vergeben Sie einen Namen.
3. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp aus.
4. Legen Sie so viele Anwenderdatentyp-Elemente an, wie Sie benötigen.
5. Konfigurieren Sie die Anwenderdatentyp-Elemente im Inspektorfenster in der Gruppe "Eigenschaften".

Alternativ konfigurieren Sie die Eigenschaften der Anwenderdatentyp-Elemente direkt in der Tabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Sie haben in der Version 0.0.1 des Anwenderdatentyps Elemente hinzugefügt. Die Version 0.0.1 ist im Zustand "in Bearbeitung".

Um die Version im Projekt zu verwenden geben Sie die Version im nächsten Schritt frei.

Siehe auch

Grundlagen zu Anwenderdatentypen (Seite 607)

2.5.4 Versionen von Anwenderdatentypen verwalten

Einleitung

Alle Anwenderdatentypen haben mindestens eine Version. Beim Erstellen des Anwenderdatentyps wird gleichzeitig eine Version erstellt, die im Zustand "in Bearbeitung" ist. In diesem Zustand bearbeiten Sie den Anwenderdatentypen beliebig. Wenn Sie die Bearbeitung abgeschlossen haben, geben Sie die Version des Anwenderdatentyps frei.

Voraussetzung

- Ein Anwenderdatentyp ist angelegt.
- Der Anwenderdatentyp hat die Version 0.0.1. und ist im Zustand "in Bearbeitung".
- Die Task Card "Bibliothek" oder die Bibliotheksansicht ist geöffnet.

Version freigeben

1. Selektieren Sie in der Projektbibliothek die Version 0.0.1 des Anwenderdatentyps.
2. Wählen Sie im Kontextmenü "Version freigeben".
   Ein Dialog wird geöffnet.
3. Ändern Sie bei Bedarf die Eigenschaften der Version:
   – Geben Sie in das Feld "Name" einen Namen für den Typ ein.
   – Legen Sie im Feld "Version" eine Haupt- und eine Zwischenversionsnummer für die frei zu gebende Version fest.
   – Um die Versionsverwaltung des Typs zu bereinigen, aktivieren Sie "Nicht verwendete Versionen des Typs aus der Bibliothek löschen".

Sie haben die Version 0.0.1 des Anwenderdatentyps freigegeben.

Version bearbeiten

1. Selektieren Sie in der Projektbibliothek z. B. die freigegebene Version 0.0.1 eines Anwenderdatentyps.
2. Wählen Sie im Kontextmenü "Typ editieren".

Die Bibliotheksansicht wird geöffnet. Die neue Version 0.0.2 des Anwenderdatentyps wurde erzeugt.

Die Version hat den Status "in Bearbeitung".

Letzte Version des Anwenderdatentyps wiederherstellen

Die letzte freigegebene Version des Anwenderdatentyps ist die Version 0.0.2.

Sie bearbeiten den Anwenderdatentyp. Eine neue Version des Anwenderdatentyps wird erzeugt 0.0.3 und erhält den Status "in Bearbeitung".

1. Selektieren Sie in der Projektbibliothek die Version des Anwenderdatentyps.
2. Wählen Sie im Kontextmenü "Änderungen verwerfen und Version löschen".

Alternativ

1. Wenn Sie eine Version zur Bearbeitung geöffnet haben, klicken Sie in der Funktionsleiste auf die Schaltfläche "Änderungen verwerfen und Version löschen".
   Die Version wird gelöscht.

Alle Änderungen des Anwenderdatentyps seit der letzten Freigabe werden verworfen. Der Anwenderdatentyp ist wieder freigegeben und hat die Version 0.0.2.

Anwenderdatentyp löschen

Wenn Sie einen Anwenderdatentyp löschen, werden alle Instanzen und Kopiervorlagen des Anwenderdatentyps mitgelöscht. Dies gilt auch für HMI-Variablen, die einen HMI-Anwenderdatentyp verwenden.

Um einen Anwenderdatentyp zu löschen, gehen Sie wie folgt vor:

1. Selektieren Sie in der Projektbibliothek unter "Typen" den zu löschenden Anwenderdatentyp.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

2.5.5 Variable mit Datentyp Anwenderdatentyp anlegen

Einleitung

Beim Anlegen einer Variable weisen Sie die Version eines Anwenderdatentyps als Datentyp zu. Sie können im Editor "Variablen" interne Variablen oder Variablen mit Anbindung zu einer Steuerung anlegen. An einer Variable stehen alle Anwenderdatentyp-Versionen zur Verfügung, die den gleichen Kommunikationstreiber verwenden wie die Variable selbst.
In Verbindung mit einer Steuerung können Sie einen Anwenderdatentyp nur verwenden, wenn Sie die absolute Adressierung gewählt haben.

Voraussetzung

- Ein Anwenderdatentyp mit Anwenderdatentyp-Elementen ist angelegt.
- Der Anwenderdatentyp ist freigegeben.
- Die Variablentabelle ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften von Variablen ist geöffnet.

Vorgehen

Um eine Variable vom Datentyp "Anwenderdatentyp" anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Variablentabelle in der Spalte "Name" auf "Hinzufügen". Eine neue Variable wird angelegt.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Geben Sie im Feld "Name" einen eindeutigen Variablennamen ein.
4. Wählen Sie im Feld "Verbindung" die Verbindung zur gewünschten Steuerung aus.

5. Wählen Sie im Feld "Datentyp" die gewünschte Version eines Anwenderdatentyps aus. Von der gewählten Verbindung hängt ab, welche Anwenderdatentypen Ihnen angezeigt werden.
   Bei internen Variablen stehen Ihnen nur Anwenderdatentyp-Vesionen vom Typ <Interner Anwenderdatentyp> zur Verfügung.

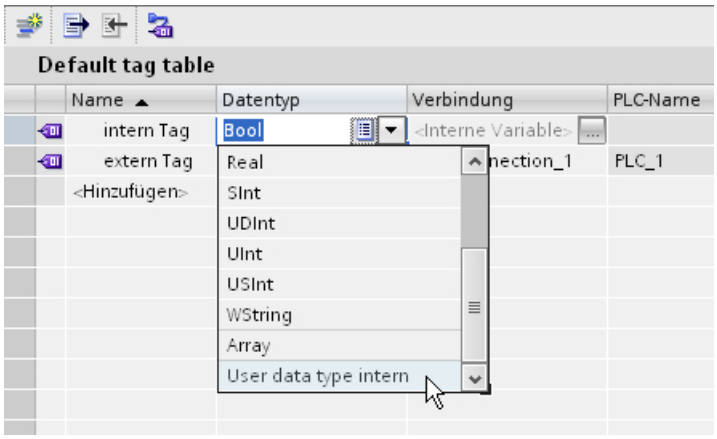

   Bei Variablen mit Verbindung zu einer Steuerung stehen Ihnen nur die Anwenderdatentypen zur Verfügung, die eine Anbindung zu einer Steuerung enthalten.

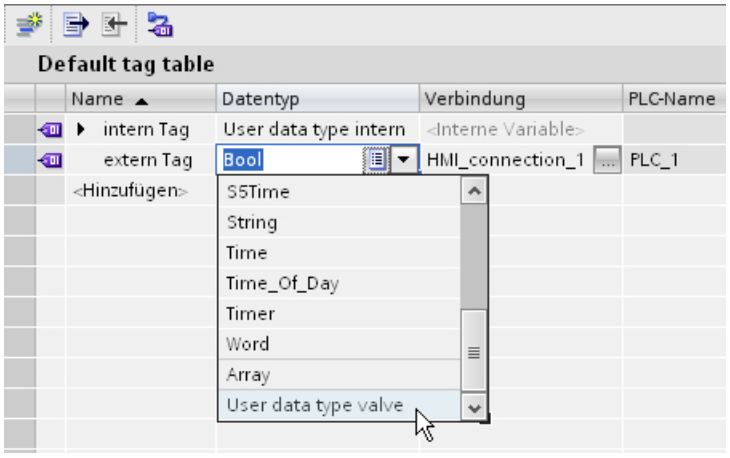

6. Tragen Sie im Bereich "Einstellungen" im Feld "Adresse" die Adresse aus der Steuerung ein, auf die Sie mit der externen Variablen zugreifen wollen. Die eingegebene Adresse muss immer auf das Start-Datenbit verweisen, z. B. <DB1.DBX0.0>.

Ergebnis

Sie haben eine Variable vom Datentyp "Anwenderdatentyp" angelegt. In weiteren Schritten konfigurieren Sie die Variable, indem Sie z. B. Startwert und Grenzwerte festlegen.

Mit der Variable vom Datentyp "Anwenderdatentyp" können Sie die dynamischen Eigenschaften einer Instanz eines Bildbausteins verbinden, welche diese Eigenschaften in Runtime mit Werten versorgt.

Wenn Sie die Eigenschaften einer Anwenderdatentyp-Variable ändern wollen, müssen Sie die Eigenschaften des Anwenderdatentyp-Elements ändern.

Eigenschaften wie z. B. "Startwert", "Ersatzwert verwenden" oder "Lineare Skalierung" können Sie auch in den verwendeten Instanzen des Anwenderdatentyps ändern.

Siehe auch

Grundlagen zu Anwenderdatentypen (Seite 607)

2.6 Arbeiten mit Zyklen

2.6.1 Grundlagen zu Zyklen

Einleitung

Zyklen werden verwendet, um regelmäßig wiederkehrende Aktionen in Runtime zu steuern. Klassische Anwendungen sind der Erfassungszyklus, der Archivierungszyklus und der Aktualisierungszyklus. Neben den bereits in WinCC vorgegebenen Zyklen können Sie auch eigene Zyklen definieren.

Prinzip

In Runtime werden regelmäßig wiederkehrende Aktionen durch Zyklen gesteuert. Typische Anwendungen für Zyklen:

- Erfassung von externen Variablen
  Der Erfassungszyklus bestimmt, wann der Prozesswert einer externen Variablen vom Bediengerät aus der Steuerung ausgelesen wird. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs.
  Wählen Sie den Erfassungszyklus nicht zu klein, da sonst die Kommunikationslast des Prozesses stark ansteigt.
- Archivierung von Prozesswerten
  Der Archivierungszyklus bestimmt, wann ein Prozesswert in der Archivdatenbank gespeichert wird. Der Archivierungszyklus ist immer ein ganzzahliges Vielfaches des Erfassungszyklus.

Der kleinste Wert für einen Zyklus hängt vom Bediengerät ab, für das Sie projektieren. Für die meisten Bediengeräte beträgt dieser Wert 100 ms. Die Werte aller weiteren Zyklen sind immer ein ganzzahliges Vielfaches des kleinsten Werts.

Wenn die in WinCC vorgegebenen Zyklen nicht den Erfordernissen Ihres Projekts genügen, definieren Sie eigene Zyklen mithilfe des Editors "Zyklen". Selbst definierte Zyklen dürfen auch nur ein ganzzahliges Vielfaches des kleinsten Werts sein.

Hinweis

Geräteabhängigkeit

Für Basic Panels können keine selbst definierten Zykluszeiten projektiert werden.

Anwendungsbeispiel

Zyklen verwenden Sie z. B. für folgende Aufgaben:

- Um einen Prozess regelmäßig zu protokollieren.
- Um auf Wartungsintervalle aufmerksam zu machen.

Aktualisierung der externen Variablen in Runtime

Wenn Sie die Erfassungsart "Zyklisch im Betrieb" für die Aktualisierung der externen Variablen in Runtime für Comfort Panels wählen und die externe Variable nicht im aktuellen Bild verwendet wird, enthält die Variable nicht in allen Fällen den aktuellen Wert. In folgenden Fällen ist es daher erforderlich, die Erfassungsart "Zyklisch fortlaufend" zu wählen:

- In einem Tooltip, der Grenzen beim Editieren des Rezepturelements anzeigt. D.h. wenn die Grenzen von Rezepturelementen (Rezepturvariablen) dynamisch ("Oben 2" und "Unten 2" mit Variablen belegt) sind, müssen die Variablen dieser Grenzen auf "Zyklisch fortlaufend" eingestellt werden.
- Die Triggervariablen von HMI-Meldungen (Bitmeldungen und Analogmeldungen) werden automatisch auf "Zyklisch fortlaufend" umgestellt. Das darf nicht zurückgestellt werden.
- Bei der Verwendung von verketteten Textlisten für dynamische Parameter von HMI-Meldungen (Bitmeldungen und Analogmeldungen) müssen die Variablen der zweiten und aller weiteren untergeordneten Ebenen auf "Zyklisch fortlaufend" umgestellt werden.
- Bei "bitgetriggerten" Kurven müssen die Steuervariablen Kurvenanforderung, Kurventransfer 1 und Kurventransfer 2 auf "Zyklisch fortlaufend" gestellt werden.
- In VB-Skripten, wenn der Variablenname sich dynamisch zusammensetzt, indem Strings verknüpft werden.

Siehe auch

Zyklus definieren (Seite 617)

Grundlagen zu Variablen (Seite 537)

2.6.2 Zyklus definieren

Einleitung

Zyklen setzen Sie ein, um regelmäßig wiederkehrende Aktionen in Runtime zu steuern. Neben den bereits in WinCC vorgegebenen Zyklen können Sie auch eigene Zyklen definieren.

Voraussetzung

Das Projekt ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Zyklen".
   Der Editor "Zyklen" wird geöffnet.
2. Doppelklicken Sie im Editor "Zyklen" in der Spalte "Name" auf den Eintrag "Hinzufügen".
   Eine neue Zykluszeit wird angelegt.
3. Geben Sie im Feld "Name" einen eindeutigen Namen ein.
4. Wählen Sie die gewünschte Zykluseinheit aus.

5. Wählen Sie den gewünschten Wert für die Zykluszeit.
   Die zur Auswahl stehenden Werte hängen von der eingestellten Zykluseinheit ab. Der kleinste Wert für einen Zyklus hängt vom Bediengerät ab, für das Sie projektieren. Für die meisten Bediengeräte beträgt dieser Wert 100 ms. Alle einstellbaren Werte sind immer ein ganzzahliges Vielfaches dieses Werts.
6. Optional geben Sie einen Kommentar zur Verwendung des Zyklus ein.

Ergebnis

Der von Ihnen projektierte Zyklus wird angelegt und steht neben den in WinCC vorgegebenen Zyklen beim Projektieren zur Verfügung.

Siehe auch

Grundlagen zu Zyklen (Seite 616)

2.7 Arbeiten mit Zyklen

2.7.1 Grundlagen zu Zyklen

Einleitung

Zyklen werden verwendet, um regelmäßig wiederkehrende Aktionen in Runtime zu steuern. Klassische Anwendungen sind der Erfassungszyklus, der Archivierungszyklus und der Bildzyklus. Neben den bereits in WinCC vorgegebenen Zyklen können Sie auch eigene Zyklen definieren.

Prinzip

In Runtime werden regelmäßig wiederkehrende Aktionen durch Zyklen gesteuert. Typische Anwendungen für Zyklen:

- Aktualisierung von Bildern
  Der Bildzyklus bestimmt, wie häufig ein Bild neu aufgebaut wird. Mit dem Bildzyklus haben Sie die Möglichkeit, alle Dynamisierungen innerhalb des Bildes gleichzeitig zu triggern. Als Zykluszeit können Sie die in WinCC vorgegebenen Zyklen oder selbst definierte Zyklen verwenden. Die Standardeinstellung für den Bildzyklus beträgt 2 Sekunden.
- Aktualisierung von Objekten
  In der Standardeinstellung verwenden alle Dynamisierungen, die in einem Bild enthalten sind, den Bildzyklus. Bei Bedarf können Sie aber für einzelne Objekte andere Zyklen für die Aktualisierung verwenden. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Dynamisierung von Bildern (Seite 210).
- Triggern von Aufgaben
  Im Aufgabenplaner haben Sie die Möglichkeit, eine Aufgabe mit zyklischem Trigger zu projektieren. Mithilfe der Zykluszeit bestimmen Sie, wann die Aufgabe ausgeführt wird. Nähere Hinweise hierzu finden Sie unter Trigger (Seite 1874)
- Archivierung von Prozesswerten
  Der Archivierungszyklus bestimmt, wann ein Prozesswert in der Archivdatenbank gespeichert wird. Der Archivierungszyklus ist immer ein ganzzahliges Vielfaches des Erfassungszyklus. Nähere Hinweise hierzu finden Sie unter Zyklen und Ereignisse (Seite 665).

Die kleinsten Werte für einen Zyklus in Runtime Professional betragen 100 und 125 ms. Die nächstmöglichen Werte betragen 250 und 300 ms. Alle weiteren Werte können Sie mit einer Schrittweite von 100 ms projektieren.

Hinweis

Zykluswert 500 ms

Der Zykluswert von 500 ms steht ausschließlich für die Archivierung von Prozesswerten zur Verfügung.

Hinweis

Zykluswert 125 ms

Der Zykluswert von 125 ms wird ausschließlich im Runtime Scripting von der Methode "ActivateDynamic" unterstützt. Alle anderen Zyklen definieren Sie im Editor "Zyklen".

Hinweis

Eigene Zyklen definieren

Wenn die in WinCC vordefinierten Zyklen nicht den Erfordernissen Ihres Projekts genügen, definieren Sie eigene Zyklen mithilfe des Editors "Zyklen".

Sie können eine unbegrenzte Anzahl an Zyklen anlegen und bei der Projektierung der Archivierung verwenden. Beachten Sie, dass in Runtime maximal 15 unterschiedliche Aktualisierungszyklen überwacht werden können.

Hinweis

Verwendung von Zyklen in Bildern

Für die Dynamisierung der Objekt- und Bildeigenschaften und als Bildzyklus nutzen Sie die 15 systemdefinierte Zyklen.

Startpunkt

Für jeden Zyklus in WinCC können Sie einen Startpunkt festlegen, dies gilt auch für die vom System fest vorgegebenen Zyklen. Folgende Startpunkte sind verfügbar:

- Zyklus anstoßen bei Start von Runtime
- Zyklus anstoßen beim Herunterfahren des Systems
- Zyklus anstoßen nach selbst definiertem Startzeitpunkt

Startpunkte für Zyklen verwenden Sie für folgende Aufgaben:

- Synchronisation unterschiedlicher Aufgaben im System
- Entlastung der Kommunikationskanäle durch zeitliche Verschiebung von kommunikationslastigen Aufgaben.

Anwendungsbeispiel

Zyklen verwenden Sie z. B. für folgende Aufgaben:

- Um einen Prozess regelmäßig zu protokollieren.
- Um auf Wartungsintervalle aufmerksam zu machen.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Startpunkt festlegen (Seite 621)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Trigger (Seite 1874)

Zyklen und Ereignisse (Seite 665)

2.7.2 Zyklus definieren

Einleitung

Zyklen setzen Sie ein, um regelmäßig wiederkehrende Aktionen in Runtime zu steuern. Neben den bereits in WinCC vorgegebenen Zyklen können Sie auch eigene Zyklen definieren.

Voraussetzung

Das Projekt ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Zyklen".
   Der Editor "Zyklen" wird geöffnet.
2. Doppelklicken Sie im Editor "Zyklen" in der Spalte "Name" auf den Eintrag "Hinzufügen".
   Eine neue Zykluszeit wird angelegt.
3. Geben Sie im Feld "Name" einen eindeutigen Namen ein.
4. Wählen Sie die gewünschte Zykluseinheit aus.
5. Wählen Sie den gewünschten Wert für die Zykluszeit.
   Die zur Auswahl stehenden Werte hängen von der eingestellten Zykluseinheit ab.
6. Optional geben Sie einen Kommentar zur Verwendung des Zyklus ein.

Ergebnis

Der von Ihnen projektierte Zyklus wird angelegt und steht neben den in WinCC vorgegebenen Zyklen beim Projektieren zur Verfügung.

Siehe auch

Grundlagen zu Zyklen (Seite 619)

2.7.3 Startpunkt festlegen

Einleitung

Für Zyklen legen Sie Startpunkte fest, um verschiedene Zyklen zu unterschiedlichen Zeiten zu starten. Durch den verteilten Start von Aktualisierungsvorgängen verteilen Sie z. B. die Kommunikationslast auf der Verbindung zur Steuerung.

Voraussetzung

- Der Editor "Zyklen" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften der Zyklen ist geöffnet.

Vorgehen

Um den Startpunkt für einen Zyklus festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gewünschten Zyklus in der Tabelle des Editors "Zyklen".
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Startpunkt".
   Die Eigenschaften für den Startpunkt des Zyklus werden angezeigt.
3. Wählen Sie die gewünschten Einstellungen für den Startpunkt.
4. Speichern Sie das Projekt.

Ergebnis

Der gewählte Zyklus wird in Runtime zum projektierten Startzeitpunkt gestartet.

Siehe auch

Grundlagen zu Zyklen (Seite 619)

2.8 Referenz

2.8.1 Abbildung und Codierung der Datentypen

Einleitung

Der Datentyp einer externen Variablen wird von WinCC entsprechend dem Datentyp der verbundenen PLC-Variable eingestellt. Wenn der Datentyp der PLC-Variable in WinCC nicht verfügbar ist, wird an der HMI-Variable automatisch ein kompatibler Datentyp verwendet. Bei Bedarf können Sie festlegen, dass WinCC einen anderen Datentyp verwendet und eine Formatanpassung zwischen dem Datentyp der PLC-Variable und dem Datentyp der HMI-Variable durchführt. In den nachfolgenden Kapiteln finden Sie die Datentypen, die standardmäßig für die Abbildung verwendet werden. Weiterhin finden Sie die möglichen Abbildungsvarianten, die Sie selbst einstellen können.

Codierung der Datentypen

Zusätzlich zur Abbildung der Datentypen können Sie an vielen Abbildungsvarianten zusätzlich die Codierung ändern. Die Codierung wird verwendet, um die binären Informationen von PLC-Variablen den Anforderungen entsprechend abbilden und verarbeiten zu können. Die Werte der PLC-Variablen können mit folgenden Codierungen abgebildet werden:

BCD

Der BCD-Code ist ein binärer Dezimalcode, mit dem jede einzelne Ziffer einer Zahl binär dargestellt wird. Die Wertigkeit der Stellen ist gemäß der Ziffernfolge 8-4-2-1. Damit ergibt sich folgende Wertetabelle:

Dezimalziffer	BCD-Code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Aiken

Der Aiken-Code ist ein komplementärer BCD-Code. Die Wertigkeit der Stellen ist gemäß der Ziffernfolge 2-4-2-1. Damit ergibt sich folgende Wertetabelle:

Dezimalziffer	Aiken-Code
0	0000
1	0001

Dezimalziffer	Aiken-Code
2	0010
3	0011
4	0100
5	1011
6	1100
7	1101
8	1110
9	1111

Excess

Für die Excess-Codierung wird der Excess-3 Code verwendet. Der Excess-3 Code ist ein komplementärer BCD-Code. Die Wertigkeit der Stellen ist gemäß der Ziffernfolge 8-4-(-2)-(-1). Damit ergibt sich folgende Wertetabelle:

Dezimalziffer	Excess-Code
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

ExtBCD

Die Codierung "ExtBCD" verwendet für die Codierung den oben beschriebenen BCD-Code. Allerdings wird bei "ExtBCD" das Most Significant Bit als Vorzeichen interpretiert.

MSB

Die Codierung "MSB" verwendet für die Codierung den Binär-Code Standard. Allerdings wird bei "MSB" das Most Significant Bit als Vorzeichen interpretiert.

NanoToMilliseconds

Die PLC-Variablen mit den Datentypen "LTime" und "LTime_Of_Day" enthalten eine Anzahl von Nanosekunden. Da die HMI-Variablen aber keine Nanosekunden verarbeiten, werden sie durch das Coding „NanoToMilliseconds“ in den HMI-Runtimes in Millisekunden umgerechnet:

Wert der PLC-Variable mit PLC-Datentyp "LTime" oder "LTime_Of_Day"	Umgerechneter Wert in der HMI-Variablen mit dem HMI-Datentyp "DInt"oder "UDInt"
1000000	1
2000000	2
10000000	10
100000000	100
...	...

Siehe auch

Codierung und Wertebereich der Datentypen (Seite 625)
Datentyp einer Variable anpassen (Seite 556)

2.8.2 Codierung und Wertebereich der Datentypen

Codierung und Wertebereich

Abhängig von der gewählten Codierung ergeben sich für die verschiedenen Datentypen die in den Tabellen genannten Wertebereiche.

PLC-Datentyp Byte

Konvertierung	Wertebereich
Byte to UnsignedByte	0...255
Byte to UnsignedWord	0...255
Byte to UnsignedDword	0...255
Byte to SignedByte	0...127
Byte to SignedWord	0...255
Byte to SignedDword	0...255
Byte to BCDByte	0...99
Byte to BCDWord	0...255
Byte to BCDDword	0...255
Byte to AikenByte	0...99
Byte to AikenWord	0...255
Byte to AikenDword	0...255
Byte to ExcessByte	0...99
Byte to ExcessWord	0...255
Byte to ExcessDword	0...255

PLC-Datentyp Char

Konvertierung	Wertebereich
Char to UnsignedByte	0...127
Char to UnsignedWord	0...127
Char to UnsignedDword	0...127
Char to SignedByte	-128...+127
Char to SignedWord	-128...+127
Char to SignedDword	-128...+127
Char to MSBByte	-128...+127
Char to MSBWord	-128...+127
Char to MSBDword	-128...+127
Char to BCDByte	0...99
Char to BCDWord	0...127

Konvertierung	Wertebereich
Char to BCDDword	0...127
Char to SignedBCDByte	-9...+9
Char to SignedBCDWord	-128...+127
Char to SignedBCDDword	-128...+127
Char to ExtSignedBCDByte	-79...+79
Char to ExtSignedBCDWord	-128...+127
Char to ExtSignedBCDDword	-128...+127
Char to AikenByte	0...99
Char to AikenWord	0...127
Char to AikenDword	0...127
Char to SignedAikenByte	-9...+9
Char to SignedAikenWord	-128...+127
Char to SignedAikenDword	-128...+127
Char to ExcessByte	0...99
Char to ExcessWord	0...127
Char to ExcessDword	0...127
Char to SignedExcessByte	-9...+9
Char to SignedExcessWord	-128...+127
Char to SignedExcessDword	-128...+127
Char to UnsignedByte	0...127
Char to UnsignedWord	0...127
Char to UnsignedDword	0...127

PLC-Datentyp Real

Konvertierung	Wertebereich
Real to Real	+-1.79769313486231e+308
Real to UnsignedByte	0...255
Real to UnsignedWord	0...65535
Real to UnsignedDword	0...4294967295
Real to SignedByte	-128...+127
Real to SignedWord	-32768...+32767
Real to SignedDword	-2147483647...+2147483647
Real to Float	+-3.402823e+38
Real to MSBByte	-127...+127
Real to MSBWord	-32767...+32767
Real to MSBDword	-2147483647...+2147483647
Real to BCDByte	0...99
Real to BCDWord	0...9999
Real to BCDDword	0...99999999
Real to SignedBCDByte	-9...+9
Real to SignedBCDWord	-999...+999
Real to SignedBCDDword	-9999999...+9999999

Konvertierung	Wertebereich
Real to ExtSignedBCDByte	-79...+79
Real to ExtSignedBCDWord	-7999...+7999
Real to ExtSignedBCDDword	-79999999...+79999999
Real to AikenByte	0...99
Real to AikenWord	0...9999
Real to AikenDword	0...99999999
Real to SignedAikenByte	-9...+9
Real to SignedAikenWord	-999...+999
Real to SignedAikenDword	-9999999...+9999999
Real to ExcessByte	0...99
Real to ExcessWord	0...9999
Real to ExcessDword	0...99999999
Real to SignedExcessByte	-9...+9
Real to SignedExcessWord	-999...+999
Real to SignedExcessDword	-9999999...+9999999
Real to S5Timer	10...9990000
Real to S5Float	+-1.701411e+38

PLC-Datentyp DWord

Konvertierung	Wertebereich
DWord to UnsignedDword	0...4294967295
DWord to UnsignedByte	0...255
DWord to UnsignedWord	0...65535
DWord to SignedByte	0...127
DWord to SignedWord	0...32767
DWord to SignedDword	0...2147483647
DWord to BCDByte	0...99
DWord to BCDWord	0...9999
DWord to BCDDword	0...99999999
DWord to AikenByte	0...99
DWord to AikenWord	0...9999
DWord to AikenDword	0...99999999
DWord to ExcessByte	0...99
DWord to ExcessWord	0...9999
DWord to ExcessDword	0...99999999
DWord to SimaticTimer	10...9990000
DWord to SimaticBCDTimer	10...9990000
DWord to UnsignedDword	0...4294967295
DWord to UnsignedByte	0...255
DWord to UnsignedWord	0...65535

PLC-Datentyp Float

Konvertierung	Wertebereich
Float to Float	+-3.402823e+38 (no Conversion)
Float to UnsignedByte	0...255
Float to UnsignedWord	0...65535
Float to UnsignedDword	0...4.294967e+09
Float to SignedByte	-128...+127
Float to SignedWord	-32768...+32767
Float to SignedDword	-2.147483e+09...+2.147483e+09
Float to Real	+-3.402823e+38
Float to MSBByte	-127...+127
Float to MSBWord	-32767...+32767
Float to MSBDword	-2.147483e+09...+2.147483e+09
Float to BCDByte	0...99
Float to BCDWord	0...9999
Float to BCDDword	0...9.999999e+07
Float to SignedBCDByte	-9...+9
Float to SignedBCDWord	-999...+999
Float to SignedBCDDword	-9999999...+9999999
Float to ExtSignedBCDByte	-79...+79
Float to ExtSignedBCDWord	-7999...+7999
Float to ExtSignedBCDDword	-7.999999e+07...+7.999999e+07
Float to AikenByte	0...99
Float to AikenWord	0...9999
Float to AikenDword	0...9,999999e+07
Float to SignedAikenByte	-9...+9
Float to SignedAikenWord	-999...+999
Float to SignedAikenDword	-9999999...+9999999
Float to ExcessByte	0...99
Float to ExcessWord	0...9999
Float to ExcessDword	0...9.999999e+07
Float to SignedExcessByte	-9...+9
Float to SignedExcessWord	-999...+999
Float to SignedExcessDword	-9999999...+9999999
Float to S5Timer	10...9990000
Float to S5Float	+-1.701411e+38

PLC-Datentyp Long

Konvertierung	Wertebereich
Long to SignedDword	-2147483647...+2147483647
Long to UnsignedByte	0...255
Long to UnsignedWord	0...65535
Long to UnsignedDword	0...2147483647

Konvertierung	Wertebereich
Long to SignedByte	-128...+127
Long to SignedWord	-32768...+32767
Long to MSBByte	-127...+127
Long to MSBWord	-32767...+32767
Long to MSBDword	-2147483647...+2147483647
Long to BCDByte	0...99
Long to BCDWord	0...9999
Long to BCDDword	0...99999999
Long to SignedBCDByte	-9...+9
Long to SignedBCDWord	-999...+999
Long to SignedBCDDword	-9999999...+9999999
Long to ExtSignedBCDByte	-79..+79
Long to ExtSignedBCDWord	-7999...+7999
Long to ExtSignedBCDDword	-79999999...+79999999
Long to AikenByte	0...99
Long to AikenWord	0...9999
Long to AikenDword	0...99999999
Long to SignedAikenByte	-9...+9
Long to SignedAikenWord	-999...+999
Long to SignedAikenDword	-9999999...+9999999
Long to ExcessByte	0...99
Long to ExcessWord	0...9999
Long to ExcessDword	0...99999999
Long to SignedExcessByte	-9...+9
Long to SignedExcessWord	-999...+999
Long to SignedExcessDword	-9999999...+9999999
Long to SimaticTimer	10...9990000
Long to SimaticBCDTimer	10...9990000

PLC-Datentyp Short

Konvertierung	Wertebereich
Short to UnsignedByte	0...255
Short to UnsignedWord	0...32767
Short to UnsignedDword	0...32767
Short to SignedByte	-128...+127
Short to SignedWord	-32768...+32767
Short to SignedDword	-32768...+32767
Short to MSBByte	-127...+127
Short to MSBWord	-32767...+32767
Short to MSBDword	-32768...+32767
Short to BCDByte	0...99
Short to BCDWord	0...9999

Konvertierung	Wertebereich
Short to BCDDword	0...32767
Short to SignedBCDByte	-9...+9
Short to SignedBCDWord	-999...+999
Short to SignedBCDDword	-32768...+32767
Short to ExtSignedBCDByte	-79...+79
Short to ExtSignedBCDWord	-7999...+7999
Short to ExtSignedBCDDword	-32768...+32767
Short to AikenByte	0...99
Short to AikenWord	0...9999
Short to AikenDword	0...32767
Short to SignedAikenByte	-9...+9
Short to SignedAikenWord	-999...+999
Short to SignedAikenDword	-32768...+32767
Short to ExcessByte	0...99
Short to ExcessWord	0...9999
Short to ExcessDword	0...32767
Short to SignedExcessByte	-9...+9
Short to SignedExcessWord	-999...+999
Short to SignedExcessDword	-32768...+32767
Short to UnsignedByte	0...255
Short to UnsignedWord	0...32767
Short to UnsignedDword	0...32767

PLC-Datentyp Word

Konvertierung	Wertebereich
Word to UnsignedWord	0...65535
Word to UnsignedByte	0...255
Word to UnsignedDword	0...65535
Word to SignedByte	0...127
Word to SignedWord	0...32767
Word to SignedDword	0...65535
Word to BCDByte	0...99
Word to BCDWord	0...9999
Word to BCDDword	0...65535
Word to AikenByte	0...99
Word to AikenWord	0...9999
Word to AikenDword	0...65535
Word to ExcessByte	0...99
Word to ExcessWord	0...9999
Word to ExcessDword	0...65535
Word to SimaticCounter	0...999
Word to SimaticBCDCounter	0...999

Konvertierung	Wertebereich
Word to UnsignedWord	0...65535
Word to UnsignedByte	0...255
Word to UnsignedDword	0...65535

Siehe auch

Abbildung und Codierung der Datentypen (Seite 623)

2.8.3 Datentypen

2.8.3.1 Datentypen für SIMATIC S7-300/400

Übersicht

Die folgende Tabelle zeigt die Datentypen für SIMATIC S7-300/400 mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bool	Bool	0 (FALSE), 1 (TRUE)
Byte	USInt	0 … 255
Word	UInt	0 … 65 535
DWord	UDInt	0 … 4294967295
Char	USInt	0 … 255
Int	Int	−32768 … 32767
DInt	DInt	−2147483648 … +2147483647
Real	Real	±1.17549E-38 bis ±3.40282E +38 und 0.0
Time	DInt	−2147483648 … +2147483647
Date	DateTime/UInt	0 … 65535
Time_of_Day, TOD	DateTime/UDInt	0 … 86399999
S5Time	UDInt	0 … 9990000
Counter	UInt	0 … 999
Timer	UDInt	0 … 9990000
String	String	-

2.8.3.2 Datentypen für SIMATIC S7 1200 und S7 1500

Datentypen für SIMATIC S7-1200 - V2

Übersicht

Die folgende Tabelle zeigt die Datentypen für SIMATIC S7-1200 - V2 mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bool	Bool	0 (FALSE), 1 (TRUE)
SInt	SInt	-128 … +127
Int	Int	-32768 … +32767
DInt	DInt	−2147483648 … +2147483647
USInt	USInt	0 … 255
UInt	UInt	0 … 65535
UDInt	UDInt	0 … 4294967295
Real	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
LReal	LReal	±1.79769313486231E+308 bis ±2.22507385850720E-308 und 0.0
Time	DInt	−2147483648 … +2147483647
Date	DateTime/UInt	0 … 65535
DTL	DateTime	-
Time_of_Day, TOD	DateTime/UDInt	0 … 86399999
String	String	-
Char	USInt	0 … 255
Array of Char	String	-
Byte	USInt	0 … 255
Word	UInt	0 … 65535
DWord	UDInt	0 … 4294967295
PlcUDT	PlcUDT	

Datentypen für SIMATIC S7-1500

Übersicht

Die folgende Tabelle zeigt die Datentypen für SIMATIC S7-1500 mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bool	Bool	0 (FALSE), 1 (TRUE)
SInt	SInt	-128 … +127

Datentyp PLC	HMI-Datentyp	Wertebereich
Int	Int	-32768 … +32767
DInt	DInt	−2147483648 … +2147483647
LInt	LReal	-4,503,599,627,370,495 … +4,503,599,627,370,495
USInt	USInt	0 … 255
UInt	UInt	0 … 65535
UDInt	UDInt	0 … 4294967295
ULInt	LReal	0 … 4,503,599,627,370,495
Real	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
LReal	LReal	±1.79769313486231E+308 bis ±2.22507385850720E-308 und 0.0
S5Time	UDInt	0 … 9990000
Time	DInt	−2147483648 … +2147483647
LTime	DInt	−2147483648 … +2147483647
Date	DateTime/UInt	0 … 65535
Time_of_Day, TOD	DateTime/UDInt	0 … 86399999
LTime_of_Day, LTOD	DateTime/UDInt	0 … 86399999
Date_And_Time, DT	DateTime	-
LDT	DateTime	-
DTL	DateTime	-
String	String	-
WString	WString	-
Char	USInt	0 … 255
WChar	WString	-
Byte	USInt	0 … 255
Word	UInt	0 … 65535
DWord	UDInt	0 … 4294967295
Timer	UDInt	0 … 9990000
Counter	UInt	0 … 999
PLCUDT	PLCUDT	

Hinweis

Die 64-Bit PLC-Datentypen LINT, UINT und LWORD werden im HMI-Kanal auf den HMI-Datentyp LREAL abgebildet. Dabei ist mit einem Genauigkeitsverlust zu rechnen, wenn der Wert die Größe 2^50 überschreitet.

Benutzerdefinierte PLC-Datentypen (UDT)

Übersicht

Sie können mit den HMI-Variablen auch DB-Instanzen von benutzerdefinierten PLC-Datentypen (UDT) verbinden.

Der PLC-Datentyp und die zugehörigen DB-Instanzen werden zentral in STEP 7 angelegt und aktualisiert. In WinCC können Sie als PLC-Variable (DB-Instanzen) folgende Quellen verwenden:

- Datenbaustein-Elemente, die einen UDT als Datentyp verwenden
- Instanz-Datenbausteine von einem UDT

Der Datentyp wird aus STEP 7 übernommen und nicht zu einen HMI-Datentyp umgewandelt. Die Zugriffsart ist immer "Symbolischer Zugriff".

Elemente eines PLC-Datentyps

Bei einem strukturierten PLC-Datentyp haben Sie in WinCC auf folgende Elemente Zugriff:

- Elemente, die in STEP 7 für WinCC freigegeben worden sind.
- Elemente, deren Datentypen in WinCC unterstützt werden.

Hinweis

In WinCC ungültige Elemente eines PLC-Datentyps

Ungültige Elemente erzeugen in WinCC einen Fehler.

Wenn Sie für die entsprechenden Elemente des dazugehörigen PLC-Datentyps in STEP 7 die Option "Erreichbar aus HMI" deaktivieren, sind diese Elemente in WinCC ausgeschlossen.

Namenskonventionen

Folgende Zeichen sind im Namen des PLC-Datentyps ungültig und erzeugen in WinCC einen Fehler:

- Punkt: "."
- Eckige Klammern: "(" und ")"

Eigenschaften

Die Eigenschaften des PLC-Datentyps und seiner Elemente werden in WinCC übernommen. Abhängig vom verwendeten Datentyp sind die Eigenschaften in WinCC nur lesbar oder auch schreibbar.

Wenn Sie in WinCC die Verbindung des PLC-Datentyps ändern, werden alle Elemente des PLC-Datentyps gelöscht und die Eigenschaften der neu verbunden PLC-Variable verwendet.

In WinCC haben Sie Zugriff auf STEP 7-Kommentare zu Elementen des PLC-Datentyps.

Bei folgenden Elementen von PLC-Datentypen haben Sie in WinCC beschränkten Zugriff auf Eigenschaften:

- Elemente vom Datentyp "Struct"
- PLC-Datentyp
- Multidimensionale Arrays
- Arrays von komplexen Datentypen, ausgenommen "DTL"

Abbildung des Datentyps "DTL"

Wenn ein PLC-Datentyp Elemente vom Datentyp "DTL" enthält, werden diese Elemente in WinCC ohne untergeordnete Elemente abgebildet. Der Datentyp "DTL" wird in WinCC zu "DateTime".

Variablen mit Elementen des Datentyps "DTL"

Variablen, die elementweise den Datentyp "DTL" verwenden, können bei symbolischer Adressierung nur lesend verwendet werden, z. B. mit SIMATIC S7 1200 und SIMATIC S7 1500. Bei absoluter Adressierung ist auch schreibender Zugriff möglich.

2.8.3.3 Datentypen für OPC-DA

Übersicht

Die folgende Tabelle zeigt die Datentypen für OPC-DA mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
VT_EMPTY	DInt	4
VT_BOOL	Bool	1
VT_I1	SInt	1
VT_UI1	USInt	1
VT_I2	Int	2
VT_UI2	UInt	2
VT_I4	DInt	4
VT_UI4	UDInt	4
VT_R4	Real	4
VT_R8	LReal	8
VT_BSTR	WString	Anzahl Zeichen
VT_DATE | VT_ARRAY (Array [lo..hi] of VT_DATE)	Raw / Array [lo..hi] of DateTime	-
VT_I1 | VT_ARRAY (Array [lo..hi] of VT_I1)	Raw / Array [lo..hi] of SInt	-
VT_UI1 | VT_ARRAY (Array [lo..hi] of VT_UI1)	Raw / Array [lo..hi] of USInt	-

Datentyp PLC	HMI-Datentyp	Wertebereich
VT_I2 | VT_ARRAY (Array [lo..hi] of VT_I2)	Raw / Array [lo..hi] of Int	-
VT_UI2 | VT_ARRAY (Array [lo..hi] of VT_UI2)	Raw / Array [lo..hi] of UInt	-
VT_I4 | VT_ARRAY (Array [lo..hi] of VT_I4)	Raw / Array [lo..hi] of DInt	-
VT_UI4 | VT_ARRAY (Array [lo..hi] of VT_UI4)	Raw / Array [lo..hi] of UDInt	-
VT_R4 | VT_ARRAY (Array [lo..hi] of VT_R4)	Raw / Array [lo..hi] of Real	-
VT_R8 | VT_ARRAY (Array [lo..hi] of VT_R8)	Raw / Array [lo..hi] of LReal	-

2.8.3.4 Datentypen für OPC-XML-DA

Übersicht

Die folgende Tabelle zeigt die Datentypen für OPC-XML-DA mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Boolean	Bool	0 (FALSE), 1 (TRUE)
Byte	SInt	-128 … +127
UnsignedByte	USInt	0 … 255
Short	Int	-32768 … +32767
UnsignedShort	UInt	0 … 65535
Int	DInt	−2147483648…+2147483647
UnsignedInt	UDInt	0 … 4294967295
Float	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
Double	LReal	±1.79769313486231E+308 bis ±2.22507385850720E-308 und 0.0
String	WString	-
Base64Binary	Raw	-
ArrayOfByte (Array [lo … hi] of byte)	Raw	-
ArrayOfShort (Array [lo … hi] of short)	Raw	-
ArrayOfUnsignedShort (Array [lo … hi] of unsignedShort)	Raw	-
ArrayOfInt (Array [lo … hi] of int)	Raw	-
ArrayOfUnsignedInt (Array [lo … hi] of unsignedInt)	Raw	-
ArrayOfLong (Array [lo … hi] of long)	Raw	-

Datentyp PLC	HMI-Datentyp	Wertebereich
ArrayOfUnsignedLong (Array [lo … hi] of unsignedLong)	Raw	-
ArrayOfFloat (Array [lo … hi] of float)	Raw	-
ArrayOfDecimal (Array [lo … hi] of decimal)	Raw	-
ArrayOfDouble (Array [lo … hi] of double)	Raw	-
ArrayOfBoolean (Array [lo … hi] of boolean)	Raw	-
ArrayOfString (Array [lo … hi] of string)	Raw	-
ArrayOfDateTime (Array [lo … hi] of dateTime)	Raw	-

2.8.3.5 Datentypen für Allen Bradley Ethernet IP

Übersicht

Die folgende Tabelle zeigt die Datentypen für Allen Bradley Ethernet IP mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bool	Bool	0 (FALSE), 1 (TRUE)
SInt	SInt	-128 … +127
USInt	USInt	0 … 255
Int	Int	-32768..+32767
UInt	UInt	0 … 65535
DInt	DInt	−2147483648 … +2147483647
UDInt	UDInt	0 … 4294967295
Real	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
String	String	nicht anwendbar

2.8.3.6 Datentypen für Modicon Modbus TCP/IP

Übersicht

Die folgende Tabelle zeigt die Datentypen für Modicon Modbus TCP/IP mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bit	Bool	0 (FALSE), 1 (TRUE)
+/- Int	Int	-32768 … +32767

Datentyp PLC	HMI-Datentyp	Wertebereich
Int	UInt	0 … 65535
16 bit group	UInt	0 … 65535
+/- Double	DInt	−2147483648 … +2147483647
Double	UDInt	0 … 4294967295
Float	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
ASCII	String	nicht anwendbar

2.8.3.7 Datentypen für Mitsubishi MC

Übersicht

Die folgende Tabelle zeigt die Datentypen für Mitsubishi MC mit den entsprechenden HMI-Datentypen und Wertebereichen in WinCC.

Datentyp PLC	HMI-Datentyp	Wertebereich
Bool	Bool	0 (FALSE), 1 (TRUE)
4-bit block	USInt	0 … 255
8-bit block	USInt	0 … 255
12-bit block	UInt	0 … 65535
16-bit block	UInt	0 … 65535
Int	Int	-32768 … +32767
Word	UInt	0 … 65535
20-bit block	UDInt	0 … 4294967295
24-bit block	UDInt	0 … 4294967295
28-bit block	UDInt	0 … 4294967295
32-bit block	UDInt	0 … 4294967295
DInt	DInt	−2147483648 … +2147483647
DWord	UDInt	0 … 4294967295
Real	Real	±1.17549E-38 bis ±-3.40282E +38 und 0.0
String	String	nicht anwendbar

2.9 Variablen archivieren

2.9.1 Grundlagen zur Variablenarchivierung

Einleitung

Die Variablenarchivierung dient zur Erfassung, Verarbeitung und Archivierung von Prozessdaten aus einer industriellen Anlage.

Aus den archivierten Prozessdaten gewinnen Sie in einer anschließenden Auswertung wesentliche betriebswirtschaftliche und technische Erkenntnisse über den Betriebszustand einer Anlage.

Anwendung der Variablenarchivierung

Die Variablenarchivierung verwenden Sie zur Analyse von Fehlerzuständen und zur Dokumentation des Prozesses. Durch die Auswertung von Variablenarchiven können Sie Wartungszyklen optimieren, die Produktqualität erhöhen und Qualitätsstandards sicherstellen.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Archivieren von Variablen (Seite 641)

Variablenarchivierung (Seite 658)

2.9.2 Variablenarchivierung

2.9.2.1 Grundlagen

Variablenarchivierung

Einleitung

Prozesswerte sind Daten, die im Prozess anfallen und im Speicher eines der angeschlossenen Automatisierungssysteme abgelegt werden. Sie stellen den Zustand einer Anlage dar, z. B. als Temperaturen, Füllstände oder Schaltzustände. Für die Erfassung und Bearbeitung der Prozesswerte definieren Sie in WinCC Variablen.

In WinCC dienen externe Variablen zur Erfassung von Prozesswerten und greifen auf eine Speicheradresse im angeschlossenen Automatisierungssystem zu. Interne Variablen haben keine Prozessanbindung und stehen nur am jeweiligen Bediengerät zur Verfügung.

Prinzip

Die Werte von externen und internen Variablen archivieren Sie in Variablenarchiven. Für jede Variable legen Sie eine Archivvariable an, und legen fest, in welchem Archiv sie gespeichert wird.

Die Variablenarchivierung wird über Zyklen und Ereignisse gesteuert. Die Archivierungszyklen dienen zur kontinuierlichen Erfassung und Speicherung der Variablenwerte. Darüber hinaus können Sie die Variablenarchivierung durch Ereignisse triggern, z. B. durch die Wertänderung einer Variable. Diese Einstellungen legen Sie für jede Archivvariable getrennt fest.

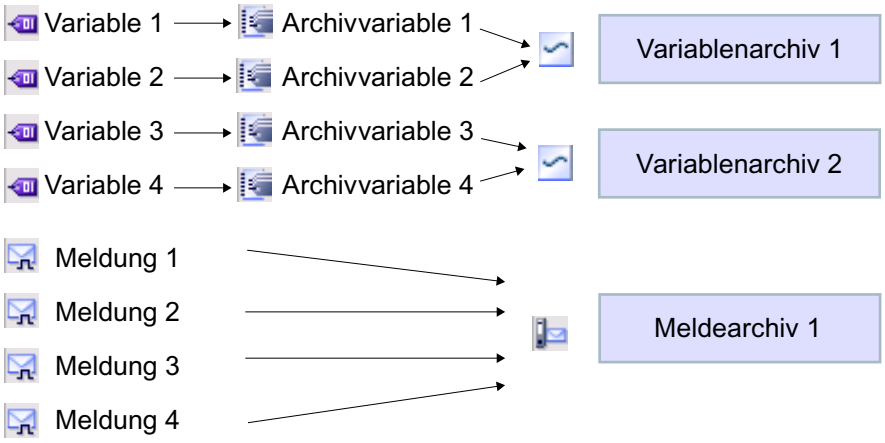

In Runtime werden die zu archivierenden Variablenwerte erfasst, verarbeitet und im Variablenarchiv gespeichert. Wo das Variablenarchiv gespeichert wird, ist abhängig vom verwendeten Bediengerät. Die gespeicherten Daten können Sie in anderen Programmen weiter bearbeiten, z. B. zu Analysezwecken.

Archivierungsmethoden

In WinCC stehen Ihnen folgende Archivierungsmethoden zur Verfügung:

- Umlaufarchiv
- Segmentiertes Umlaufarchiv
- Umlaufarchiv mit füllstandsabhängiger Systemmeldung
- Umlaufarchiv mit Ausführen von Systemfunktionen bei vollem Archiv

Ausgeben von archivierten Daten

In Runtime können Sie die archivierten Variablenwerte in den Prozessbildern als Kurven ausgeben.

Siehe auch

Archivieren von Variablen (Seite 641)

2.9.2.2 Arbeiten mit Variablenarchiven

Archivieren von Variablen

Einleitung

In Runtime legen Sie Variablenwerte in Archiven ab. Die archivierten Daten werten Sie später für Ihre Zwecke aus. Für die Archivierung einer Variable legen Sie folgende Bedingungen fest:

- Die Archivvariable, über die die Werte der verbundenen Variable archiviert werden.
- Das Variablenarchiv, in dem die Variable gespeichert wird.
- Den Zyklus oder das Ereignis, mit dem die Variable gespeichert wird.
- Den Wertebereich, innerhalb dessen die Variable gespeichert wird.

Hinweis

Sie verwenden die Variablenarchivierung vorrangig zur Archivierung von Werten der externen Variablen. Sie können aber auch die Werte von internen Variablen archivieren.

Prinzip

Bei der Variablenarchivierung wirken mehrere Schritte zusammen:

- Variablenarchiv anlegen und konfigurieren
  Beim Anlegen eines Variablenarchivs legen Sie folgende Einstellungen fest:
  - Allgemeinen Einstellungen, z. B. Name, Größe, Speicherort
  - Verhalten bei Runtime-Start
  - Verhalten bei vollem Archiv
- Archivierung der Variablen konfigurieren
  Für jede Archivvariable geben Sie ein Variablenarchiv an, in dem in Runtime die Werte der verbundenen Variable und weitere Informationen, z. B. der Archivierungszeitpunkt, archiviert werden.
  Weiterhin legen Sie fest, wann und wie oft die Werte einer Archivvariablen archiviert werden. Hierfür gibt es folgende Möglichkeiten:
  - "Auf Anforderung":
    Die Variablenwerte werden durch Aufruf der Systemfunktion "ArchiviereVariable" archiviert.
  - "Bei Änderung":
    Die Variablenwerte werden archiviert, sobald das Bediengerät eine Wertänderung der Variablen feststellt.
  - "Zyklisch":
    Die Variablenwerte werden in regelmäßigen Abständen archiviert. Die in WinCC vorgegebenen Zyklen können Sie um eigene Zyklen ergänzen, die auf den vorgegebenen Zyklen basieren. Der kleinste einzustellende Wert beträgt 1 s. Alle anderen Werte sind immer ein ganzzahliges Vielfaches dieses Werts.
    Darüber hinaus können Sie die Archivierung auf Werte innerhalb oder außerhalb eines Toleranzbandes beschränken. Auf diese Weise beschränken Sie die Archivierung auf Werte innerhalb eines relevanten Wertebereichs.

  Wenn Sie eine Variable auf Anforderung archivieren, beachten Sie folgendes:
  - Archivieren Sie solche Variablen nicht in einem segmentierten Umlaufarchiv, in dem Variablen zyklisch fortlaufend oder bei Änderung archiviert werden.

  Hintergrund:
  - Wenn die Archivierung auf Anforderung nur selten eintritt, wird das Archivsegment z. B. durch zyklisch archivierte Werte gefüllt und das nächste Archivsegment wird angelegt. Wenn anschließend ein Zugriff auf die Variable erfolgt, die auf Anforderung archiviert wurde, kann die Variable nicht dargestellt werden, da in Runtime auf das aktuelle Archivsegment zugegriffen wird. Zur Abhilfe legen Sie für selten archivierte Variablen ein eigenes Variablenarchiv an.
- Archivierte Variablenwerte weiterverarbeiten
  Die archivierten Variablenwerte können Sie direkt in Ihrem Projekt, z. B. in einer Kurvenanzeige oder mit anderen Anwenderprogrammen, z. B. Excel, auswerten.

Siehe auch

Grundlagen zur Variablenarchivierung (Seite 639)

Variablenarchivierung (Seite 639)

Prüfsumme für ein Archiv projektieren (Seite 651)

Prüfsumme von Archivdaten auswerten (Seite 652)

Verhalten des Archivs bei Sprachumschaltung in Runtime steuern (Seite 655)

Variablenarchiv anlegen

Einleitung

Variablenarchive dienen der Speicherung von Werten externer und interner Variablen in Runtime. Beim Anlegen eines Archivs legen Sie den Namen des Archivs, seine Größe sowie seinen Speicherort fest. Zusätzlich können Sie zu jedem Archiv einen Kommentar eingeben.

Voraussetzung

- Ein Projekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein Variablenarchiv anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Archive".
   Der Editor für Variablenarchive und Meldearchive wird geöffnet.
2. Klicken Sie auf die Registerkarte "Variablenarchive" und doppelklicken Sie im Editor "Variablenarchive" in der Spalte "Name" auf "Hinzufügen".
   Ein neues Variablenarchiv wird angelegt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Geben Sie im Feld "Name" einen eindeutigen Namen für das Archiv ein.
5. Legen Sie im Feld "Datensätze pro Archiv" die Anzahl der zu archivierenden Datensätze pro Archiv fest.
   Die Größe des Archivs errechnet sich folgendermaßen: Anzahl der Einträge * Länge eines einzelnen zu archivierenden Variablenwerts.
   Im Inspektorfenster wird unter dem Eingabefeld für die Anzahl der Datensätze angezeigt, welche maximale Größe das Archiv annimmt, wenn die gerade gewählte Anzahl Datensätze beibehalten wird.
6. Wählen Sie im Feld "Ablageort" aus, wie die Archiveinträge gespeichert werden.
7. Wählen Sie abängig vom gewählten "Ablageort" entweder den "Pfad" oder den "Namen der Datenquelle" aus.
8. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften eines Archivs direkt in der Tabelle des Editors "Variablenarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Das Variablenarchiv wird angelegt.

In weiteren Schritten konfigurieren Sie Variablen so, dass ihre Werte in diesem Archiv abgelegt werden.

Für die weitere Konfiguration des Archivs führen Sie folgende Schritte durch:

- Festlegen des Startverhaltens des Archivs beim Start von Runtime.
- Festlegen, wie sich das Archiv verhält, wenn es voll ist.
- Projektieren einer Funktionsliste an das Ereignis "Überlauf".

Verhalten des Archivs bei Runtime-Start steuern

Einleitung

Bei der Projektierung eines Archivs legen Sie das Startverhalten des Archivs für den Start von Runtime fest. In den Eigenschaften des Archivs legen Sie fest, ob die Archivierung beim Runtime-Start beginnt. Weiterhin legen Sie fest, ob ein bestehendes Archiv fortgeführt oder überschrieben wird.

Das Startverhalten legen Sie für jedes Archiv gesondert fest.

Voraussetzung

- Ein Archiv ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften des Archivs ist geöffnet.

Vorgehen

Um das Startverhalten eines Variablenarchivs zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor Archive" das Archiv, für das Sie das Startverhalten festlegen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Startverhalten".

3. Wenn die Archivierung bei Runtime-Start beginnen soll, aktivieren Sie im Bereich "Archivierung" die Option "Archivierung bei Runtime-Start aktivieren".
   Alternativ starten Sie die Archivierung z. B. mit der Systemfunktion "StarteArchivierung".
4. Wählen Sie im Bereich "Verhalten bei Neustart" das Startverhalten des Archivs.
   - Mit der Option "Archiv zurücksetzen" werden die archivierten Werte gelöscht und die Archivierung wird neu begonnen.
   - Mit der Option "Archiv fortführen" werden die neu zu archivierenden Werte an das bestehende Archiv angefügt.

Alternativ konfigurieren Sie das Startverhalten eines Archivs direkt in der Tabelle des Editors "Archive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

In Runtime wird die Archivierung entsprechend Ihren Einstellungen gestartet.

Archivierung Füllstandsabhängig steuern

Einleitung

Die Größe eines Archivs wird durch die Anzahl der Einträge bestimmt. Mit der Archivierungsmethode bestimmen Sie, wie sich das Archiv verhält, wenn es vollständig gefüllt ist.

Archivierungsmethoden

Folgende Archivierungsmethoden stehen zur Verfügung:

- Umlaufarchiv
  Wenn die projektierte Größe des Archivs erreicht ist, werden die ältesten Einträge gelöscht. Bei Erreichen der projektierten Größe des Archivs werden ca. 20 % der ältesten Einträge gelöscht. Daher ist es nicht möglich, die vollständige Anzahl der projektierten Einträge anzuzeigen. Wählen Sie bei der Projektierung eine entsprechende Größe des Umlaufarchivs. Alternativ projektieren Sie ein segmentiertes Umlaufarchiv.
- Segmentiertes Umlaufarchiv
  In einem segmentierten Umlaufarchiv werden mehrere gleich große Archivsegmente erstellt, die nacheinander gefüllt werden. Wenn alle Archive vollständig gefüllt sind, wird das älteste Archiv neu beschrieben.

- Archiv mit füllstandsabhängiger Systemmeldung
  Bei Erreichen eines definierbaren Füllstands, z. B. von 90 %, wird eine Systemmeldung ausgelöst. Wenn das Archiv zu 100 % befüllt ist, werden neue Variablenwerte nicht archiviert.
- Archiv mit füllstandsabhängigem Auslösen eines Ereignisses.
  Bei vollständig gefülltem Archiv wird das Ereignis "Überlauf" ausgelöst. An das Ereignis projektieren Sie eine Funktionsliste, die ausgeführt wird, wenn das Ereignis "Überlauf" eintritt. Wenn die projektierte Größe des Archivs erreicht ist, werden neue Variablenwerte nicht archiviert.
  Für die weitere Bearbeitung befüllter Archive stehen Ihnen Systemfunktionen zur Verfügung.

Voraussetzung

- Ein Archiv ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften des Archivs ist geöffnet.

Vorgehensweise

1. Selektieren Sie im Editor "Archive" das Archiv, für das Sie die Archivierungsmethode festlegen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Archivierungsmethode" und wählen Sie die gewünschte Archivierungsmethode.
3. Wenn Sie ein Archiv mit der Einstellung "Segmentiertes Umlaufarchiv" gewählt haben, geben Sie die Anzahl der Archivsegmente ein. Dabei wird vom System zusätzlich ein weiteres Archivsegment für den Hauptarchiv angelegt. Somit ergibt sich die Gesamtanzahl der erstellten Archivdateien aus der Anzahl der konfigurierten Segmente sowie dem automatisch erstellten Archiv.
   Wenn Sie ein Archiv mit der Einstellung "Systemmeldung anzeigen bei" gewählt haben, geben Sie den Füllstand in Prozent an, bei dessen Erreichen eine Systemmeldung ausgegeben wird.
   Wenn Sie die Einstellung "Ereignis auslösen" gewählt haben, projektieren Sie in der Gruppe "Ereignisse" die Funktionsliste.

Alternativ konfigurieren Sie die Archivierungsmethode eines Archivs direkt in der Tabelle des Editors "Archive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Das Ereignis "Überlauf" steht in der Tabelle des Editors nicht zur Verfügung. Die Funktionsliste müssen Sie daher im Inspektorfenster projektieren.

Ergebnis

Das gewählte Archiv verhält sich in Runtime entsprechend den Einstellungen.

Siehe auch

Systemfunktionen für Archive (Seite 650)

Prozesswerte archivieren

Einleitung

In Runtime können Sie die Prozesswerte einer Variable in einem Variablenarchiv speichern. Für die Archivierung einer Variable legen Sie folgende Bedingungen fest:

- Die Archivvariable, über die die Werte der verbundenen Variable archiviert werden.
- In welchem Archiv werden die Werte gespeichert
- Unter welchen Randbedingungen werden die Werte gespeichert
- Werden nur Prozesswerte eines bestimmten Wertebereichs gespeichert

Zur Archivierung der Variablenwerte weisen Sie einer HMI-Variable eine Archivvariable zu. Die Archivvariable wird im Variablenarchiv abgelegt und archiviert die Werte der verbundenen HMI-Variable. Archivvariablen können Sie direkt im Editor "HMI-Variablen" projektieren. Der Editor "HMI-Variablen" enthält dafür die Editiertabelle "Archivvariablen".

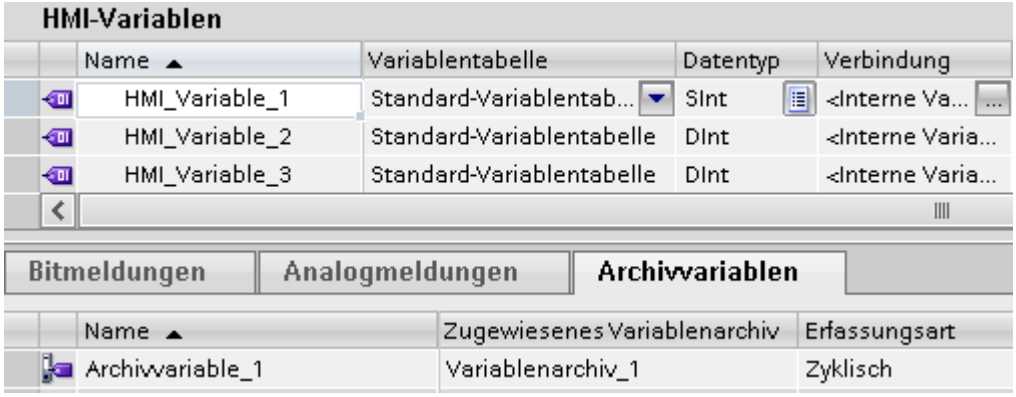

Wenn die Ansicht der Tabelle "Archivvariablen" minimiert ist, klicken Sie auf die Pfeilschaltfläche unterhalb der Variablentabelle.

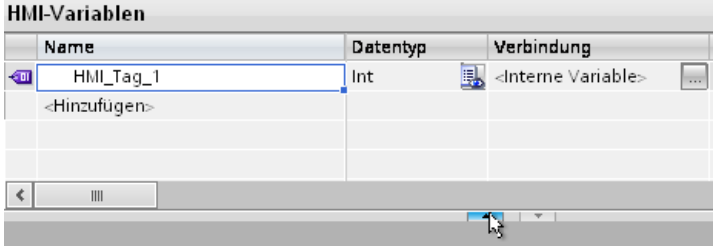

Bild 2-1 Die Tabelle "Archivvariablen" wird eingeblendet.

Voraussetzung

- Das Variablenarchiv ist angelegt.
- Die Variable, für die Sie die Archivierung festlegen wollen, ist angelegt.
- Der Editor "Variablen" ist geöffnet.

- Die Tabelle "Archivvariablen" ist eingeblendet.
- Das Inspektorfenster mit den Eigenschaften für die Variable ist geöffnet.

Vorgehen

Um Prozesswerte in einer Variable zu archivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine Variable in der Variablentabelle.
2. Doppelklicken Sie in der Tabelle "Archivvariablen" im Feld "Name" auf "Hinzufügen".
   Eine neue Archivvariable wird angelegt, sie erhält den gleichen Namen wie die zugehörige HMI-Variable.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" das Variablenarchiv aus, in dem die Werte der Variable archiviert werden.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Archivierungsart" den Archivtyp für die Archivierung aus:
   - "Zyklisch": Die Variablenwerte werden entsprechend dem eingestellten Archivierungszyklus archiviert.
   - "Bei Änderung": Die Variablenwerte werden archiviert, sobald das Bediengerät eine Wertänderung feststellt.
   - "Auf Anforderung": Die Variablenwerte werden durch Aufruf der Systemfunktion "ArchiviereVariable" archiviert.
5. Wenn Sie Variablenwerte zyklisch archivieren wollen, wählen Sie im Feld "Archivierungszyklus" eine Zykluszeit aus. Alternativ definieren Sie einen eigenen Zyklus mithilfe der Objektliste. Der kleinste einzustellende Wert beträgt 1 s. Alle anderen Werte sind immer ein ganzzahliges Vielfaches dieses Werts.
6. Wenn Sie nur Variablenwerte außerhalb oder innerhalb eines definierten Wertebereichs archivieren wollen, wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Totzone für Archivierung". Legen Sie Werte für die Obergrenze und die Untergrenze fest.
   Wenn Sie einen dynamischen Grenzwert projektieren wollen, wählen Sie über die Auswahlschaltfläche den Eintrag "HMI_Variable". Wählen Sie anschließend im zweiten Auswahlfeld die Variable aus, die den Grenzwert enthält.
   Wenn Sie einen festen Grenzwert projektieren wollen, wählen Sie den Eintrag "Konstante". Geben Sie im zweiten Feld den Grenzwert ein.
   Wenn Sie einen Grenzwert offen lassen wollen, wählen Sie den Eintrag "Keine".
7. Legen Sie unter "Umfang" fest, ob nur Variablenwerte archiviert werden, die innerhalb der definierten Grenzen liegen, oder nur Variablenwerte außerhalb der definierten Grenzen.

Alternativ konfigurieren Sie die Archivierung einer Variablen direkt in der Tabelle des Editors "Archivvariablen". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Die vollständige Projektierung einer Archivvariable können Sie auch im Editor "Archive" durchführen.

Ergebnis

Die Prozesswerte der projektierten Variable werden in Runtime entsprechend der gewählten Einstellungen archiviert.

Hinweis

Damit die Variablenwerte in Runtime tatsächlich archiviert werden, müssen Sie dafür sorgen, dass das Variablenarchiv gestartet wird. Der Start des Archivs kann automatisch mit Runtime-Start ausgeführt werden oder durch Systemfunktionen gesteuert werden. Für den automatischen Start des Archivs nehmen Sie die Einstellung in den Eigenschaften des Archivs vor.

Bei gefülltem Archiv Systemfunktion auslösen

Einleitung

Sie können eine Archivierungsmethode wählen, bei der eine Funktionsliste abgearbeitet wird, sobald das Archiv gefüllt ist.

Anwendungsbeispiel

Systemfunktionen nutzen Sie z. B., um die Daten einer gefüllten Archivdatei in ein anderes Archiv auszulagern, bevor diese überschrieben werden. Die ausgelagerten Daten können Sie in einem anderen Programm weiterbearbeiten. Hierzu müssen Sie die Systemfunktion "KopiereArchiv" an das Ereignis "Überlauf" projektieren.

Voraussetzung

- Ein Archiv mit der Archivierungsmethode "Ereignis auslösen" ist erstellt.
- Der Editor "Variablenarchive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für Variablenarchive ist geöffnet.

Vorgehen

Um eine Systemfunktion an das Ereignis "Überlauf" zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle des Editors "Variablenarchive" das gewünschte Archiv.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Überlauf".
   Die Funktionsliste öffnet sich.
3. Doppelklicken Sie auf "Funktion hinzufügen" und wählen Sie die gewünschte Systemfunktion aus.
4. Projektieren Sie die erforderlichen Parameter der gewählten Systemfunktion.

Ergebnis

In Runtime wird die Funktionsliste abgearbeitet, sobald das Archiv gefüllt ist.

Systemfunktionen für Archive

Systemfunktionen

Für die Archivierung stehen Ihnen folgende Systemfunktionen zur Verfügung:

Name der Funktion	Funktionsweise
ArchiviereProtokolldatei	Diese Funktion verschiebt oder kopiert ein Archiv zur Langzeitarchivierung an einen anderen Speicherort. Wenn Sie z. B. das Audit Trail von einem lokalen Speichermedium auf den Server verschieben wollen, verwenden Sie die Systemfunktion. Verwenden Sie bei Audit Trails immer den Modus "Archiv verschieben (hmiMove)", da sonst durch doppelte Datenhaltung gegen die FDA-Richtlinie verstoßen wird.
ArchiviereVariable	Speichert den Wert der angegebenen Variablen im angegebenen Variablenarchiv. Wenn Sie einen Prozesswert zu einem bestimmten Zeitpunkt archivieren wollen, verwenden Sie die Systemfunktion.
StarteArchivierung	Startet die Archivierung in dem angegebenen Archiv. Sie unterbrechen die Archivierung in Runtime mit der Systemfunktion "StoppeArchivierung".
StoppeArchivierung	Stoppt die Archivierung in dem angegebenen Archiv. Sie setzen die Archivierung in Runtime wieder fort mit der Systemfunktion "StarteArchivierung".
LöscheArchiv	Löscht alle Einträge des angegebenen Archivs.
StarteNächstesArchiv	Stoppt die Archivierung in dem angegebenen Archiv. Die Archivierung wird in dem Archivsegment fortgesetzt, das Sie für das angegebene Archiv projektiert haben.
SchliesseAlleArchive	Schließt alle Archive. Die Verbindung zwischen WinCC und den Archivdateien oder der Archivdatenbank wird getrennt. Wenn Sie z. B. das Speichermedium des Bediengeräts wechseln wollen, ohne die Runtime-Software zu beenden, verwenden Sie diese Systemfunktion.
ÖffneAlleArchive	Öffnet alle Archive, um die Archivierung fortzusetzen. Die Verbindung zwischen WinCC und den Archivdateien oder der Archivdatenbank wird wiederhergestellt.
KopiereArchiv	Kopiert die Inhalte aus dem angegebenen Archiv in ein anderes Archiv.

Prüfsumme für ein Archiv projektieren

Einleitung

In einem regulierten Projekt haben Sie die Möglichkeit, die Archivdaten eines Variablenarchivs oder eines Meldearchivs mit einer Prüfsumme zu versehen. Anhand dieser Prüfsumme kann im Betrieb der Anlage festgestellt werden, ob die Daten eines Archivs nachträglich verändert wurden.

Hinweis

Geräteabhängigkeit

Die Option "Prüfsumme" steht nur für Anzeige- und Bediengeräte zur Verfügung, welche die "GMP-konforme Projektierung" unterstützen.

Voraussetzung

- Die GMP-konforme Projektierung ist aktiviert.
- Ein Variablenarchiv bzw. ein Meldearchiv ist angelegt.

Vorgehensweise

Um ein Variablenarchiv oder ein Meldearchiv mit Verwendung der Prüfsumme zu projektieren, gehen Sie wie folgt vor:

1. Öffnen Sie das Variablenarchiv oder das Meldearchiv im entsprechenden Archiveditor.
2. Wählen Sie im Feld "Ablageort" den Eintrag "Datei - CSV (ASCII)" oder "Datei -TXT (Unicode)".

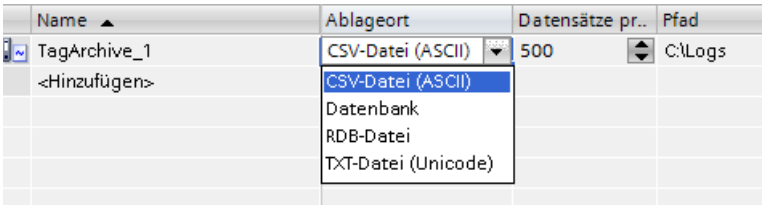

3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Archivierungsmethode" die Option "Systemmeldung anzeigen bei" oder "Ereignis auslösen".

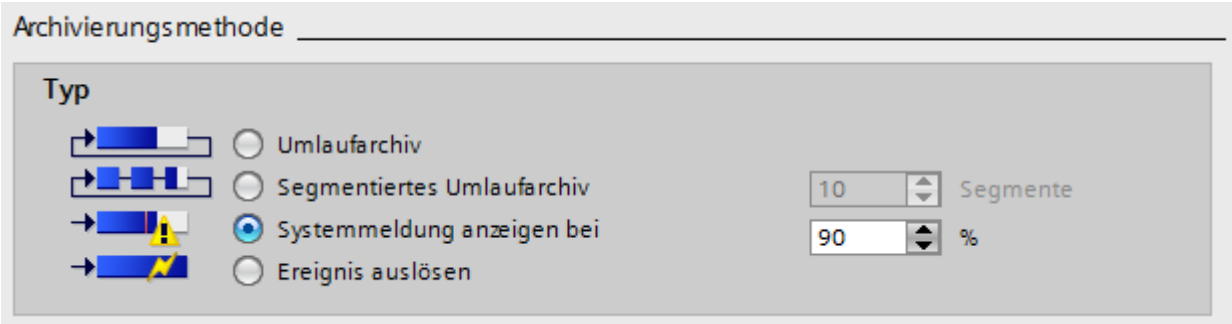

4. Aktivieren Sie in der Tabelle des Editors die Option "Prüfsumme aktivieren".
5. Aktivieren Sie in der Tabelle des Editors die Option "Archivierung bei Runtime-Start aktivieren".
   Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

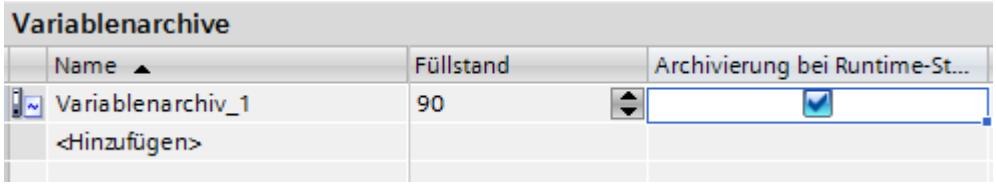

6. Speichern Sie das Projekt.

Ergebnis

Die Archivdaten des Archivs werden in Runtime mit einer Prüfsumme versehen.

Siehe auch

Archivieren von Variablen (Seite 641)

Prüfsumme von Archivdaten auswerten (Seite 652)

Prüfsumme von Archivdaten auswerten

Einleitung

Wenn Sie ein Variablenarchiv oder ein Meldearchiv mit der Bildung einer Prüfsumme projektiert haben, können Sie prüfen, ob die Archivdaten nachträglich verändert wurden.

Hinweis

Geräteabhängigkeit

Die Option "Prüfsumme" steht nur für Anzeige- und Bediengeräte zur Verfügung, welche die "GMP-konforme Projektierung" unterstützen.

Zur Prüfung der Integrität der Archivdaten steht Ihnen das DOS-Programm "HmiCheckLogIntegrity" zur Verfügung.

Das Programm "HmiCheckLogIntegrity" unterstützt die Prüfung folgender Dateien:

- Archivdateien von Meldearchiven, Variablenarchiven und Audit im CSV-Format
- Archivdateien von Meldearchiven, Variablenarchiven und Audit im TXT-Format
- Rezepturdatensätze im CSV-Format
- Rezepturdatensätze im TXT-Format

Das Programm "HmiCheckLogIntegrity.exe" finden Sie im Installationsverzeichnis von WinCC im Ordner "WinCC Runtime Advanced", z. B. <C:\Program Files\Siemens\Automation WinCC Runtime Advanced>

Hinweis

Audit Trail und Archive mit Prüfsumme

Bevor Sie WinCC durch einen Service Pack oder eine neue Version aktualisieren, beenden und sichern Sie den Audit Trail bzw. die Archive mit Prüfsumme. Nach der Aktualisierung von WinCC wird der Audit Trail bzw. die Archive mit Prüfsumme mit neuen Dateien fortgeführt.

Damit stellen Sie sicher, dass mit der neuen Version die Archive auf einem definierten Stand begonnen werden.

Vorgehensweise

1. Kopieren Sie die zu prüfende Datei vom Bediengerät auf Ihren Projektierungsrechner.
2. Öffnen Sie die Eingabeaufforderung über "Start > Programme > Zubehör > Eingabeaufforderung".
3. Geben Sie in der Eingabeaufforderung den Pfad zur "HmiCheckLogIntegrity.exe" und anschließend ein Leerzeichen ein. Nach dem Leerzeichen geben Sie den Ablageort der zu prüfenden Datei in Anführungszeichen ein.

4. Drücken Sie die Taste <Enter>.
5. Die Prüfung wird durchgeführt.
   Wenn die geprüften Daten konsistent sind, wird die Meldung "Consistency check succeeded" ausgegeben.

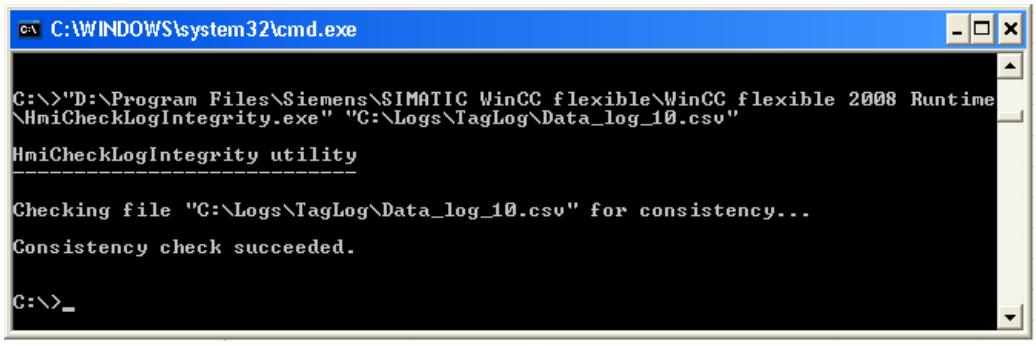

   Wenn die geprüften Daten nicht konsistent sind, wird die Meldung "Consistency check failed" ausgegeben. Zusätzlich wird ein Hinweis auf die erste inkonsistente Zeile in der Datei ausgegeben.

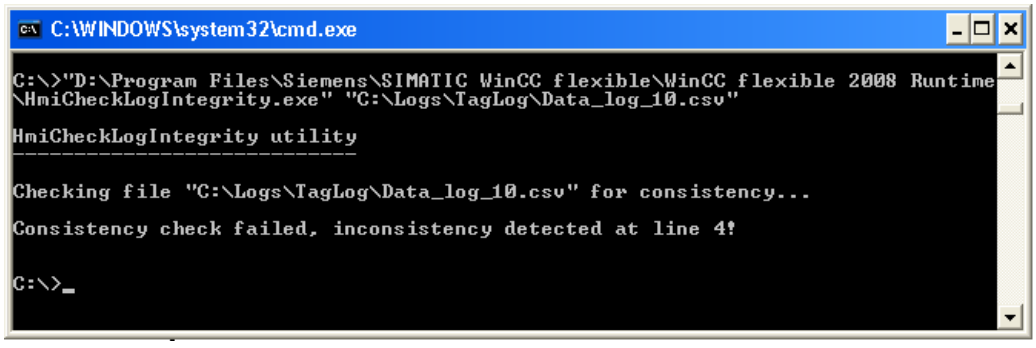

Hinweis

Wenn in der Pfadangabe zu dem Programm "HmiCheckLogIntegrity.exe" Leerzeichen vorkommen, dann müssen Sie die Pfadangabe in Anführungszeichen setzen.

Alternativ prüfen Sie die Integrität der Archivdaten mit dem AuditViewer.

Siehe auch

Archivieren von Variablen (Seite 641)

Prüfsumme für ein Archiv projektieren (Seite 651)

Verhalten des Archivs bei Sprachumschaltung in Runtime steuern

Einleitung

In den Runtime-Einstellungen Ihres Bediengeräts wählen Sie die Sprache, die verwendet wird, um Archive in Runtime zu schreiben.

Voraussetzung

- Im Editor "Projektsprachen" sind die Sprachen aktiviert, die Sie in Ihrem Projekt verwenden z. B. "Deutsch (Deutschland)" und "Englisch (USA)".

Vorgehen

Um die Anlaufsprache festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Sprache und Schriftart".
2. Aktivieren Sie die Runtime-Sprachen z. B. "Deutsch (Deutschland)" und "Englisch (USA)".
3. Legen Sie die "Reihenfolge für Sprachumschaltung" fest. Mit der 0 legen Sie die Anlaufsprache fest, z. B. folgendermaßen:
   - Deutsch 0.
   - Englisch 1
     Deutsch ist mit "0" als "Anlaufsprache" festgelegt.
4. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Allgemein".
5. Wählen Sie "Archive > Archivierungssprache > Anlaufsprache".

Ergebnis

Nach dem Transfer startet das Projekt. Durch "Reihenfolge für Sprachumschaltung" ist Deutsch als "Anlaufsprache" festgelegt. Die Archive werden in Deutsch geschrieben. Während Runtime schaltet der Bediener die Runtime-Sprache auf Englisch. Die Archive werden jedoch weiterhin in Deutsch geschrieben.

Der Bediener beendet Runtime. Durch die vorhergehende Sprachumschaltung in Runtime ist beim erneuten Start die "Anlaufsprache" Englisch. Da Englisch die Anlaufsprache ist, werden nun auch die Archive in Englisch geschrieben.

Bei einer weiteren Sprachumschaltung in Runtime bleibt die Archivierungssprache Englisch, bis zu einem erneuten Ende von Runtime.

Wenn Sie anstatt "Anlaufsprache" eine andere Option bzw. Sprache wählen, werden die Archive immer in der gewählten Sprache geschrieben. Unabhängig davon, ob der Bediener in Runtime die Sprache umschaltet.

Siehe auch

Archivieren von Variablen (Seite 641)

Archivvariablen im Editor "Archive"

Archivvariable projektieren

Einleitung

In WinCC können Sie auch im Editor "Archive" Archivvariablen anlegen und bearbeiten. Die Eigenschaften der Archivvariablen projektieren Sie ebenfalls direkt im Editor "Archive".

Hinweis

Wenn Sie im Editor "Archive" löschen, verschieben oder kopieren, wirken sich diese Änderungen auch in der Variablentabelle aus.

Voraussetzung

- Der Editor "Archive" ist geöffnet, die Registerkarte "Variablenarchive" ist aktiviert.
- Ein Variablenarchiv ist angelegt.

Vorgehen

Um im Editor "Archive" eine Archivvariable zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle des Editors "Variablenarchive" ein bestehendes Variablenarchiv.
   Alternativ doppelklicken Sie in der Spalte "Name" auf "Hinzufügen ...", um ein neues Variablenarchiv anzulegen.

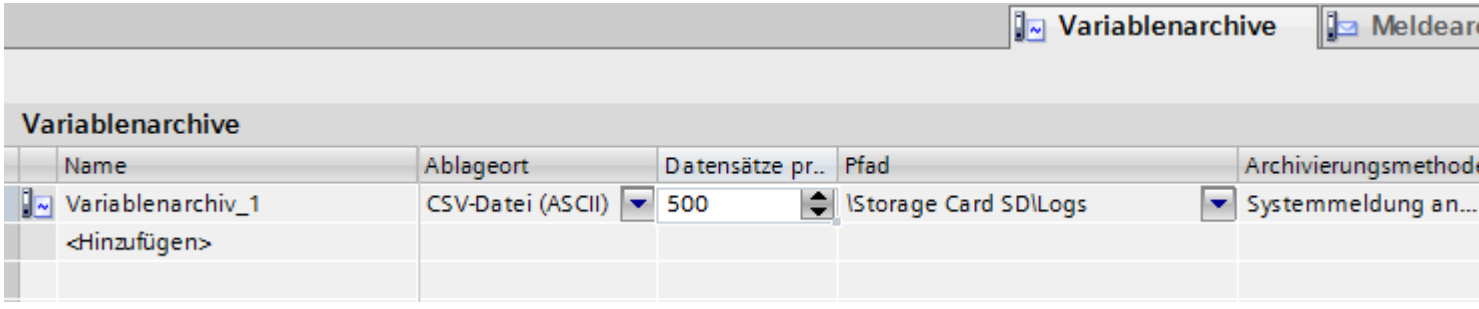

2. Doppelklicken Sie in der Tabelle des Editors "Archivvariablen" in der Spalte "Name" auf "Hinzufügen ...".

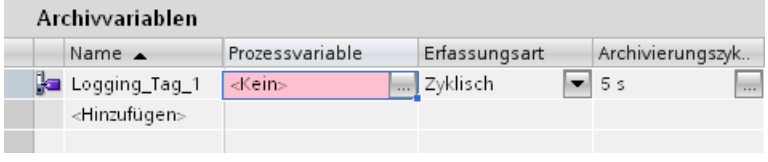

3. Geben Sie im Feld "Name" einen eindeutigen Namen für die Archivvariable ein.

4. Klicken Sie im Feld "Prozessvariable" auf die Auswahlschaltfläche und wählen Sie in der Objektliste die Prozessvariable aus, die archiviert werden soll.

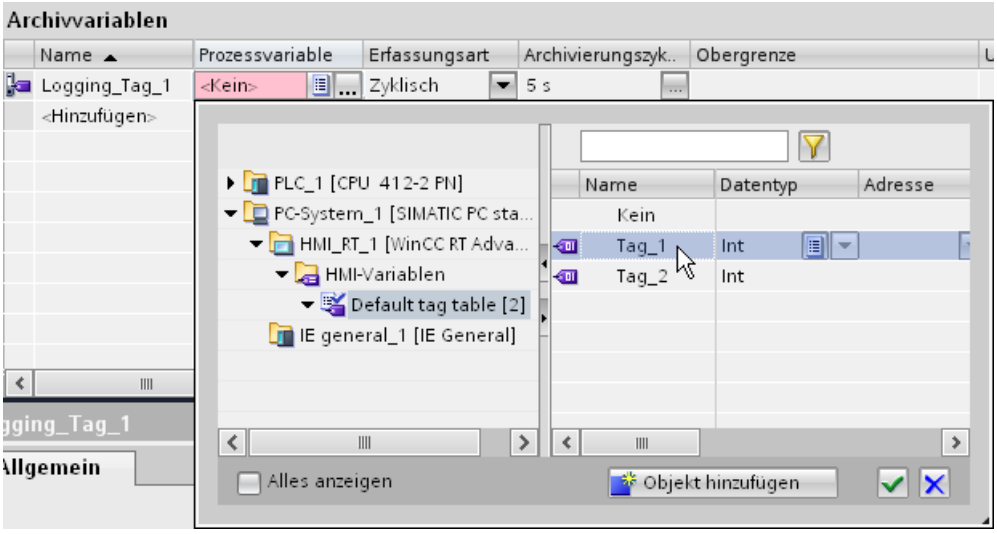

5. Wählen Sie im Feld "Archivtyp" den gewünschten Triggermodus aus:
   - "Zyklisch": Die Variablenwerte werden entsprechend dem eingestellten Archivierungszyklus archiviert.
   - "Bei Änderung": Die Variablenwerte werden archiviert, sobald das Bediengerät eine Wertänderung feststellt.
   - "Auf Anforderung": Die Variablenwerte werden durch Aufruf der Systemfunktion "ArchiviereVariable" archiviert.
6. Wenn Sie Variablenwerte zyklisch archivieren wollen, wählen Sie im Feld "Archivierungszyklus" die gewünschte Zykluszeit aus. Alternativ definieren Sie einen eigenen Zyklus mithilfe der Objektliste. Der kleinste einzustellende Wert beträgt 1 s. Alle anderen Werte sind immer ein ganzzahliges Vielfaches dieses Werts.
7. Projektieren Sie die weiteren Parameter für die Archivierung in der Tabelle des Editors oder im Inspektorfenster.

Ergebnis

Die projektierte Archivvariable ist im Editor "Archive" angelegt und wird auch in der Variablentabelle angezeigt.

2.9.3 Variablenarchivierung

2.9.3.1 Variablenarchivierung

Einleitung

Das Archivsystem ist in Runtime für die Variablenarchivierung zuständig. Das Archivsystem verarbeitet die in der Runtime-Datenbank zwischengespeicherten Prozesswerte und schreibt sie in die Archivdatenbank.

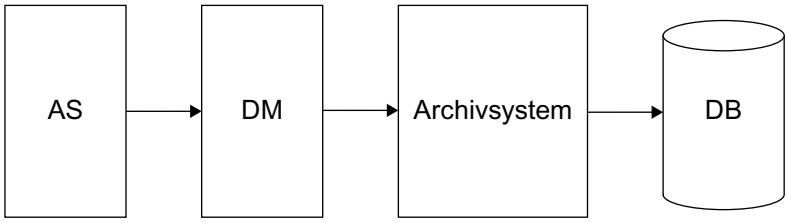

Folgende WinCC-Teilsysteme sind an der Variablenarchivierung beteiligt:

- Automatisierungsystem (AS): Speichert die Prozesswerte, die über Kommunikationstreiber an WinCC übermittelt werden.
- Datenmanager (DM): Verarbeitet die Prozesswerte und liefert Sie über Prozessvariablen an das Archivsystem.
- Archivsystem: Verarbeitet die erfassten Prozesswerte, indem es z. B. den Mittelwert bildet. Die Verarbeitungsart ist abhängig von der Projektierung des Archivs.
- Runtime-Datenbank (DB): Speichert die zu archivierenden Prozesswerte.

Begriffsdefinitionen

Ob und wann Prozesswerte erfasst und archiviert werden, ist von verschiedenen Parametern abhängig. Welche der folgenden Parameter Sie projektieren, hängt von der verwendeten Archivierungsart ab:

- Erfassungszyklus: Bestimmt, wann der Wert einer Prozessvariable im Automatisierungssystem ausgelesen wird. Einen Erfassungszyklus projektieren Sie z. B. bei der zyklischen Variablenarchivierung.
- Archivierungszyklus: Bestimmt, wann der verarbeitete Prozesswert in der Archivdatenbank gespeichert wird. Einen Archivierungszyklus projektieren Sie z. B. bei der zyklischen Variablenarchivierung.
- Start-/Stopvariable: Startet die Variablenarchivierung, wenn die zugeordnete Binärvariable den Wert "1" annimmt. Die Archivierung wird gestoppt, sobald die Startbedingung nicht mehr gegeben ist. Die Start-/Stopvariable projektieren Sie bei der zyklisch-selektiven Variablenarchivierung.

- Archivierung auf Anforderung: Prozesswerte werden durch eine Anforderung archiviert. Die Anforderung wird durch eine Binärvariable gesteuert. Für die Projektierung verwenden Sie den Triggermodus "Auf Anforderung".
- Variablenarchivierung bei Änderung: Prozesswerte werden nur bei Änderung archiviert. Für die Projektierung verwenden Sie den Triggermodus "Bei Änderung".

Siehe auch

Grundlagen zur Variablenarchivierung (Seite 639)

2.9.3.2 Grundlagen

Grundlagen zur Variablenarchivierung

Einleitung

Die Variablenarchivierung dient zur Erfassung, Verarbeitung und Archivierung von Prozessdaten aus einer industriellen Anlage. Aus den archivierten Prozessdaten gewinnen Sie in einer anschließenden Auswertung wesentliche betriebswirtschaftliche und technische Erkenntnisse über den Betriebszustand einer Anlage.

Funktionsweise

In Runtime werden die zu archivierenden Prozesswerte erfasst, verarbeitet und in der Archivdatenbank gespeichert. Sie können aktuelle oder bereits archivierte Prozesswerte in Runtime als Tabelle oder Kurve ausgeben. Darüber hinaus besteht die Möglichkeit, archivierte Prozesswerte als Protokoll auszudrucken.

Projektierung

Die Archivierung von Prozesswerten projektieren Sie im Editor "Archive". Sie erstellen Variablenarchive und Verdichtungsarchive. Sie definieren die Erfassungszyklen, Archivierungszyklen und wählen die zu archivierenden Prozesswerte aus.

Im Editor "Bilder" projektieren Sie Kurvenanzeigen und Tabellenanzeigen zur Darstellung von Prozessdaten in Runtime. Mit den Anzeigen geben Sie die Prozessdaten in Kurvenform und in Tabellenform aus.

Im Editor "Protokolle" projektieren Sie die Ausgabe von archivierten Prozessdaten als Protokoll. Die Prozesswerte geben Sie im Protokoll in Form einer Tabelle oder als Kurve aus.

Verwendung

Die Variablenarchivierung verwenden Sie z. B. für folgende Aufgaben:

- Frühzeitige Erkennung von Gefahrzuständen und Störzuständen
- Steigerung der Produktivität
- Erhöhung der Produktqualität

- Optimierung von Wartungszyklen
- Dokumentation des Verlaufs von Prozesswerten

Siehe auch

Telegrammvariablen (Seite 660)

Aufbau eines Telegramms mit Rohdatenvariablen (Seite 661)

Archivvariablen

Einleitung

Die Werte von externen und internen Variablen werden in einem Variablenarchiv in Archivvariablen gespeichert.

Archivvariablen

Zu jeder Variable, deren Werte archiviert werden sollen, muss im Variablenarchiv eine Archivvariable angelegt sein. Für die verschiedenen Datentypen der Variablen sind entsprechende Archivvariablentypen verfügbar. Folgende Archivvariablentypen werden unterschieden:

- Eine analoge Archivvariable speichert numerische Prozesswerte, z. B. den Füllstand eines Tanks.
- Eine binäre Archivvariable speichert binäre Prozesswerte, z. B. ob ein Motor ausgeschaltet oder eingeschaltet wurde.

Darüber hinaus können archivierte Prozesswerte verdichtet werden. Die Verdichtung wird durch die Anwendung von mathematischen Funktionen, z. B. durch Mittelwertbildung, erreicht. Solche verdichteten Prozesswerte werden in einem Verdichtungsarchiv in Verdichtungsarchivvariablen gespeichert.

Telegrammvariablen

Einleitung

Telegrammvariablen werden benötigt, um sich schnell ändernde Prozesswerte zu erfassen oder um mehrere Mess-Stellen einer Anlage zusammenzufassen.

Hinweis

Telegrammvariablen sind in WinCC vom Typ "Rohdatentyp" und werden deswegen auch als "Rohdatenvariable" bezeichnet.

Prinzip

Im Automatisierungssystem werden die Prozesswerte in eine Binärdatei geschrieben, als Telegramm an WinCC übertragen und dort in einer Rohdatenvariable gespeichert.

Archivierung von Telegrammen

Wenn die erfassten Prozesswerte einer Rohdatenvariable archiviert werden sollen, projektieren Sie im Variablenarchiv eine prozessgesteuerte Variable. Damit das Archivsystem das Telegramm in der prozessgesteuerten Variable verarbeiten kann, wählen Sie eine Normierungs-DLL aus. Die Normierungs-DLL wird mit dem verwendeten Automatisierungssystem mitgeliefert und zerlegt das Telegramm, um z. B. die Prozesswerte zu ermitteln. Die Prozesswerte werden danach in die Archivdatenbank geschrieben.

Eine Normierungs-DLL für die SIMATIC S7 ist standardmäßig im Lieferumfang von WinCC enthalten.

Wenn Sie eine Rohdatenvariable archivieren wollen, müssen Sie in den Eigenschaften der Rohdatenvariable die Option "Archivdatenverweis" aktivieren. Die Option aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Rohdaten". Wenn die Option "Archivdatenverweis" aktiviert ist, können Sie an der Archivvariable unter "Eigenschaften > Allgemein" im Feld "Prozessvariable" die Rohdatenvariable auswählen und verbinden.

Siehe auch

Grundlagen zur Variablenarchivierung (Seite 659)

Aufbau eines Telegramms mit Rohdatenvariablen (Seite 661)

Aufbau eines Telegramms mit Rohdatenvariablen

Einleitung

Ein Telegramm zur Übermittlung von Rohdatenvariablen besteht aus zwei Teilen, dem Telegrammkopf und dem Telegrammrumpf.

Telegrammkopf	Telegrammrumpf mit Blockstatuswort, Zeitinformationen, Messbereich, Messwerten, ...

Telegrammkopf

Im Telegrammkopf werden allgemeine Daten des Telegramms, wie zum Beispiel die Länge des Telegramms übermittelt. Das Highbyte des 0. Datenwortes wird vom System nicht benutzt und kann daher von Ihnen frei belegt werden.

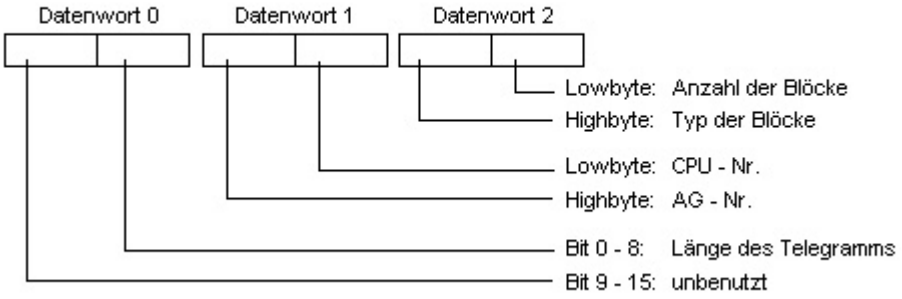

Telegrammrumpf

Im Blockstatuswort wird unter anderem das Format von Messwerten und Messbereichen festgelegt. Bit 10 ist reserviert und wird in späteren Versionen für die Umschaltung von Sommer- und Winterzeit verwendet werden (Sommerzeit = 1).

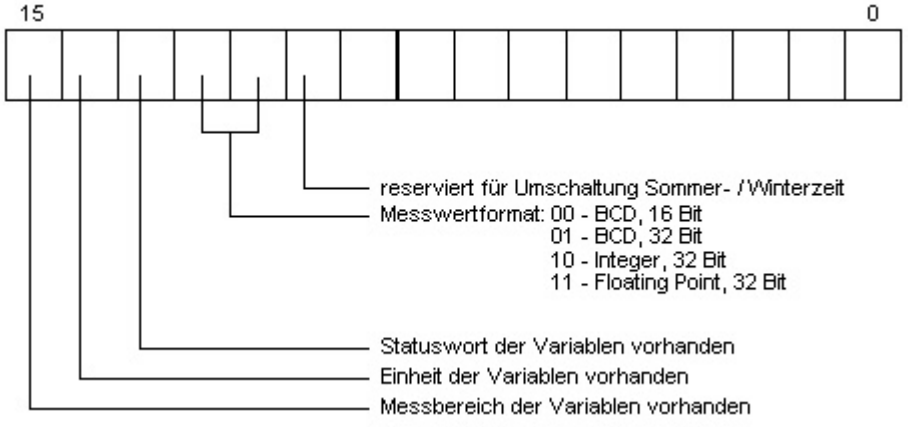

Uhrzeit und Datum werden gemäß den Definitionen des zeitfolgerichtigen Meldens aufgebaut.

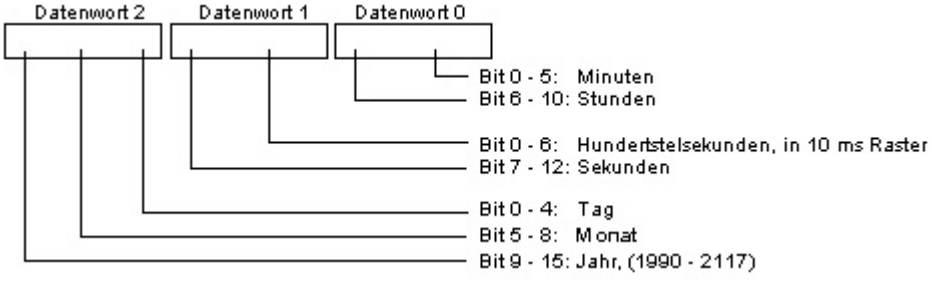

Soll ein Messbereich mit übertragen werden, so sind dafür 8 Datenworte notwendig, in denen die unteren und oberen Grenzen der Variable und des Archives angegeben werden.

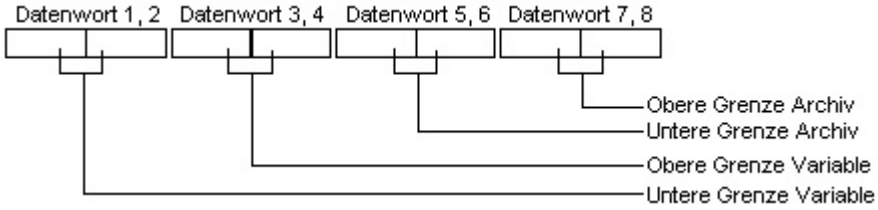

Im Folgenden werden verschiedene beispielhafte Telegrammtypen aufgezeigt.

Typ 1

Messwert einer Prozessgröße + Datum und Uhrzeit

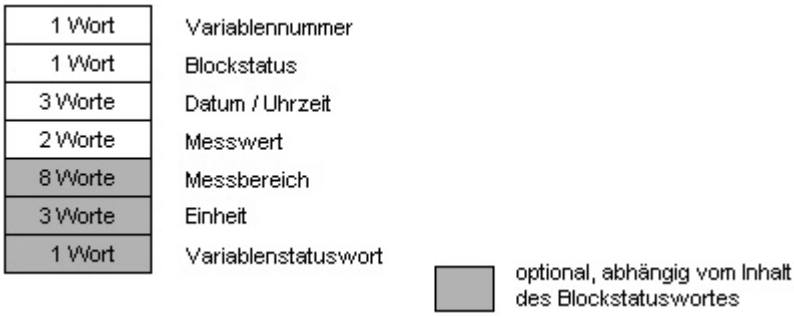

Typ 2

n Messwerte einer Prozessgröße + Datum und Uhrzeit jedes Messwertes

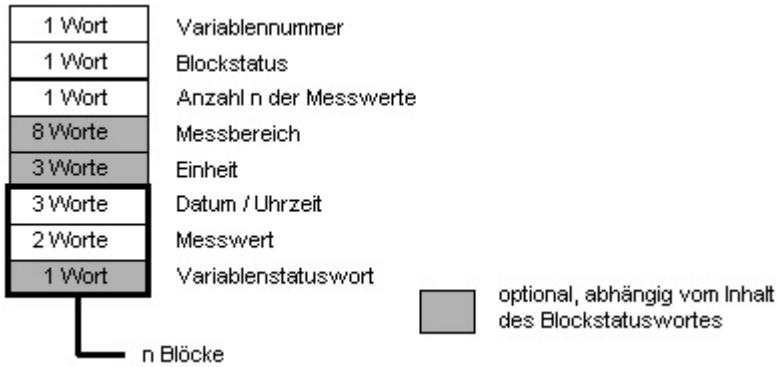

Typ 3

n Messwerte einer Prozessgröße mit Datum / Uhrzeit und Abtastzyklus

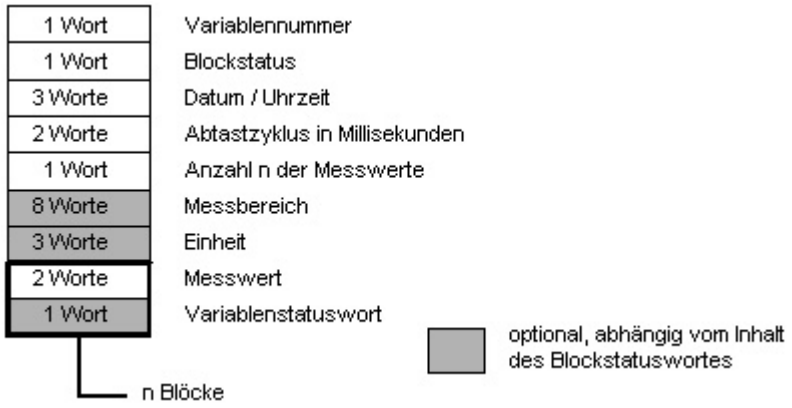

Typ 4

n Messwerte verschiedener Prozessgrößen mit Datum / Uhrzeit

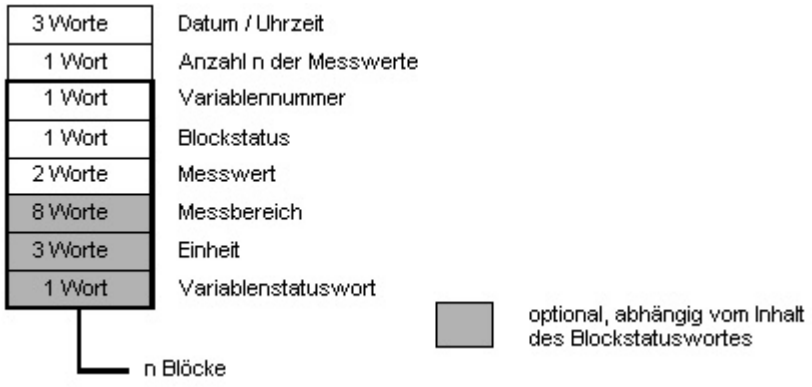

Typ 5

n Messwerte verschiedener Prozessgrößen ohne Datum und Uhrzeit

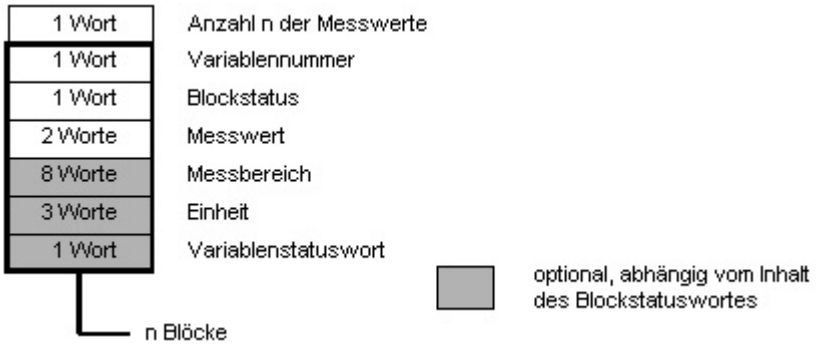

Siehe auch

Grundlagen zur Variablenarchivierung (Seite 659)

Telegrammvariablen (Seite 660)

Archivierungsarten

Triggermodus

Einleitung

Sie können verschiedene Archivierungsarten zum Archivieren von Prozesswerten anwenden. So überwachen Sie z. B. einen einzelnen Prozesswert zu bestimmten Zeitpunkten. Die Überwachung machen Sie von bestimmten Ereignissen abhängig. Sie archivieren sich schnell ändernde Prozesswerte, ohne die Systembelastung dadurch zu erhöhen. Um die Datenmenge zu reduzieren, verdichten Sie bereits archivierte Prozesswerte.

Archivierungsarten

Die Archivierungsart wird festgelegt durch den gewählten Triggermodus. Folgende Archivierungsarten stehen in Runtime zur Verfügung:

- Zyklische Variablenarchivierung: Kontinuierliche Variablenarchivierung, z. B. zur Überwachung eines Prozesswerts.
- Zyklisch-selektive Variablenarchivierung: Aktionsgesteuerte, kontinuierliche Variablenarchivierung, z. B. zur Überwachung eines Prozesswerts in einem bestimmten Zeitraum.
- Variablenarchivierung auf Anforderung: Prozesswerte werden nur auf Anforderung durch eine Triggervariable archiviert.
- Variablenarchivierung bei Änderung: Prozesswerte werden nur bei Änderung archiviert.
- Verdichtungsarchiv: Verdichtung einzelner Archivvariablen oder ganzer Variablenarchive, z. B. die stündliche Mittelwertbildung minütlich archivierter Prozesswerte.

Zyklen und Ereignisse

Einleitung

Die Variablenarchivierung wird über Zyklen und Ereignisse gesteuert. Die Erfassungszyklen und Archivierungszyklen benutzen Sie für eine kontinuierliche Erfassung und Speicherung der Prozesswerte. Darüber hinaus lösen Sie die Variablenarchivierung auch durch Ereignisse und auf Anforderung aus oder beenden sie. Zyklen und Ereignisse können Sie kombinieren. Ein Prozesswert kann z. B. regelmäßig erfasst werden; die Archivierung wird aber erst durch ein binäres Ereignis ausgelöst.

Erfassungszyklus

Der Erfassungszyklus bestimmt, in welchem Intervall der Prozesswert einer Prozessvariablen ausgelesen wird. Der kleinste einzustellende Wert beträgt 500 ms. Alle anderen Werte sind immer ein ganzzahliges Vielfaches dieses Werts. Der Startzeitpunkt beim Erfassungszyklus wird vom Startzeitpunkt von WinCC-Runtime bestimmt.

Hinweis

Ein kurzer Erfassungszyklus kann zu einer hohen Systembelastung führen. Wählen Sie einen Erfassungszyklus nur so klein, wie Sie ihn tatsächlich brauchen.

Archivierungszyklus

Der Archivierungszyklus bestimmt, wann ein Prozesswert in der Archivdatenbank gespeichert wird. Der Archivierungszyklus ist immer ein ganzzahliges Vielfaches des Erfassungszyklus. Der Startzeitpunkt beim Archivierungszyklus ist entweder vom Startzeitpunkt von WinCC-Runtime oder vom gewählten Startpunkt abhängig.

Durch die Angabe des Startpunkts wird erreicht, dass die Werte verzögert archiviert werden und die Archivierungslast verteilt wird. Zur Verdeutlichung zwei Beispiele:

- Prozesswerte werden in drei Zyklen archiviert: jede Minute, alle zwei Minuten und alle drei Minuten. Dadurch tritt z. B. alle sechs Minuten eine hohe Archivierungslast auf. Weisen Sie den drei Zyklen jeweils einen unterschiedlichen Startpunkt zu.

Archivierungszyklus	Gewählter Startpunkt
1 min	Zur 15. Sekunde
2 min	Zur 30. Sekunde
3 min	Zur 45. Sekunde

- Viele Prozesswerte müssen alle zehn Sekunden archiviert werden. Um die Archivierungslast zu verteilen, projektieren Sie zwei Zyklen von 10 Sekunden mit unterschiedlichem Startpunkt. Zu jedem der beiden gewählten Startpunkte wird eine Archivierung durchgeführt.
  Den Startpunkt für einen Zyklus projektieren Sie im Editor "Zyklen" in den Eigenschaften des Zyklus.

Durch die unterschiedlichen Startpunkte verteilt sich die Archivierungslast.

Hinweis

Wenn für Erfassung und Archivierung der gleiche Zyklus verwendet wird, dann bedeutet dies nicht zwangsläufig, dass Erfassung und Archivierung gleichzeitig gestartet werden.

Zwischen dem Zeitpunkt der Erfassung und der Archivierung kann systembedingt eine Differenz bis zur Länge des Erfassungszyklus liegen.

Bearbeitungsmethode

Alle Prozesswerte, die im Zeitraum zwischen Erfassung und Archivierung aus der Prozessvariablen ausgelesen worden sind, werden von der Archivierungsfunktion verarbeitet. Für binäre und analoge Werte stehen Ihnen unterschiedliche Bearbeitungsmethoden zur Verfügung.

In einem Variablenarchiv verwenden Sie für analoge Werte eine der folgenden Bearbeitungsmethoden:

- Aktuell: Speichert den zuletzt erfassten Prozesswert.
- Summe: Speichert die Summe aller erfassten Prozesswerte.
- Maximum: Speichert den größten aller erfassten Prozesswerte.
- Minimum: Speichert den kleinsten aller erfassten Prozesswerte.
- Mittelwert: Speichert den Mittelwert aller erfassten Prozesswerte.
- Berechnung: Der zu archivierende Wert wird aus dem zuletzt erfassten Prozesswert berechnet. Die Berechnung erfolgt durch eine im Editor "Skripte" erstellte Funktion.

In einem Variablenarchiv verwenden Sie für binäre Werte eine der folgenden Bearbeitungsmethoden:

- 0 bis 1: Die Archivierung erfolgt bei jedem Wechsel des Werts von 0 nach 1.
- 1 bis 0: Die Archivierung erfolgt bei jedem Wechsel von 1 nach 0.
- Bei jeder Änderung: Die Archivierung erfolgt sowohl bei einem Wechsel von 0 nach 1 als auch bei einem Wechsel von 1 nach 0.
- Immer: Die Archivierung erfolgt abhängig vom gewählten Zyklus und unabhängig von einer Wertänderung.

Ereignisse

Ereignisse starten und beenden die Variablenarchivierung. Die Bedingungen, die ein Ereignis triggern, können Sie an Variablen koppeln. In WinCC wird zwischen folgenden Ereignissen unterschieden:

- Binäre Aktion: Reagiert auf die Änderung einer booleschen Variable.
  Beispiel: Das Einschalten eines Motors startet die Variablenarchivierung.
- Totzonen-Kriterium: Legt einen absoluten oder prozentualen Wert fest, um den der neu erfasste Variablenwert vom vorherigen Variablenwert abweichen muss, damit der neu erfasste Variablenwert archiviert wird. Die Wertänderung kann absolut oder relativ erfolgen.
  Beispiel: Bei Temperaturschwankungen von mehr als 2 % wird die Archivierung ausgelöst.
- Zeitgesteuertes Ereignis: Reagiert auf einen festen Zeitpunkt oder eine verstrichene Zeitspanne nach Start der Variablenarchivierung.
  Beispiel: Bei jedem Schichtwechsel wird ein Archiv ausgelagert.

Zyklische Variablenarchivierung

Einleitung

Die zyklische Variablenarchivierung beginnt mit dem Start von Runtime. Die Prozesswerte werden in konstanten Zeitzyklen erfasst und in der Archivdatenbank gespeichert. Zusätzlich können Grenzwerte festgelegt werden. Der Prozesswert wird nur dann archiviert, wenn er innerhalb der festgelegten Grenzen liegt. Der Vergleich des Prozesswerts mit den Grenzwerten erfolgt nach der Erfassung des Prozesswerts.

Die zyklische Variablenarchivierung endet mit dem Beenden von Runtime.

Funktionsweise

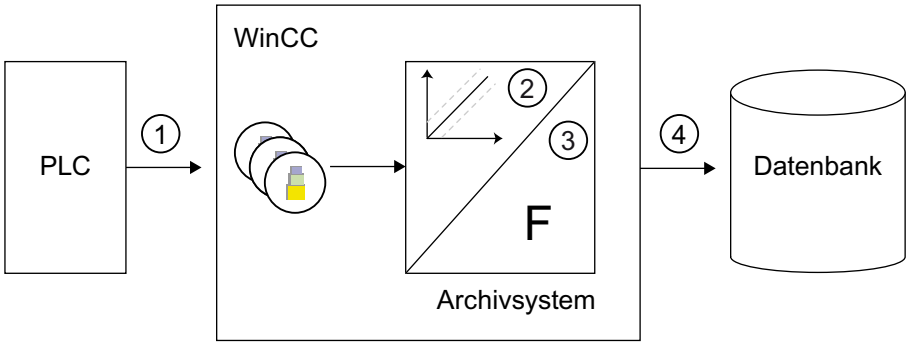

Die externen Variablen in WinCC entsprechen einem bestimmten Prozesswert im Speicher einer angeschlossenen PLC-Steuerung. Der Erfassungszyklus (1) regelt, wann der Prozesswert aus dem Speicher der angeschlossenen PLC-Steuerung ausgelesen wird.

Die Runtime-Komponente des Archivsystems verarbeitet den Prozesswert:

- Ob der Prozesswert überhaupt archiviert wird, hängt von Ihrer Projektierung ab. Der Prozesswert muss sich z. B. um einen bestimmten Betrag oder Prozentsatz ändern (2).
- Die Archivierungsfunktion (3) legt fest, wie die erfassten Prozesswerte verarbeitet werden, z. B. durch Mittelwertbildung.

Der Archivierungszyklus (4) bestimmt, wann der verarbeitete Prozesswert in die Archivdatenbank geschrieben wird.

Zyklisch-selektive Variablenarchivierung

Einleitung

Der Start der zyklisch-selektiven Variablenarchivierung wird gesteuert durch eine Variable vom Typ "Bool". Sobald die Variable in Runtime den Wert "1" annimmt und der Archivierungszyklus getriggert wurde (standardmäßig wenn Sie die Runtime starten), wird die Archivierung gestartet. Die Prozesswerte werden nach dem Start in konstanten Zeitzyklen erfasst und in der Archivdatenbank gespeichert. Zusätzlich können Grenzwerte festgelegt werden. Der Prozesswert wird nur dann archiviert, wenn er innerhalb der festgelegten Grenzen liegt. Der Vergleich des Prozesswerts mit den Grenzwerten erfolgt nach der Erfassung des Prozesswerts.

Die zyklische Variablenarchivierung endet durch folgende Ereignisse:

- Wenn die Variable für den Start der Archivierung den Wert "0" annimmt.
- Beim Beenden von Runtime.

Beim Beenden der Runtime wird zusätzlich der zuletzt erfasste Prozesswert archiviert.

Funktionsweise

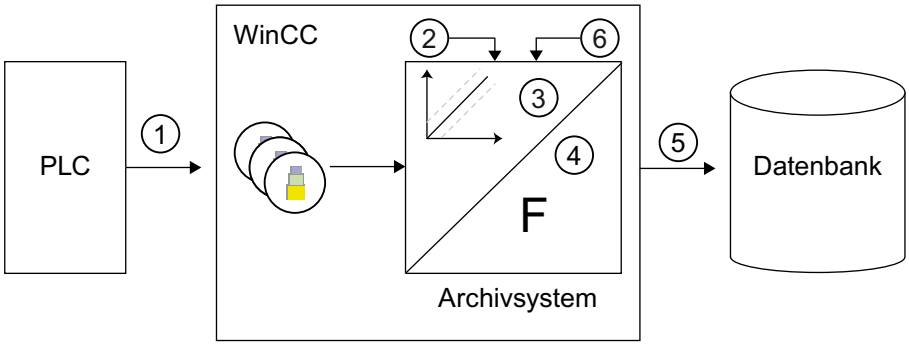

Die externen Variablen in WinCC entsprechen einem bestimmten Prozesswert im Speicher einer angeschlossenen PLC-Steuerung. Sobald die Variable für den Start der Archivierung in Runtime den Wert "1" annimmt (2), beginnt die Variablenarchivierung. Der Erfassungszyklus (1) regelt, wann der Prozesswert aus dem Speicher der angeschlossenen PLC-Steuerung ausgelesen wird.

Die Runtime-Komponente des Archivsystems verarbeitet den Prozesswert:

- Ob der Prozesswert überhaupt archiviert wird, hängt von Ihrer Projektierung ab. Der Prozesswert muss sich z. B. um einen bestimmten Betrag oder Prozentsatz ändern (3).
- Die Archivierungsfunktion (4) legt fest, wie die erfassten Prozesswerte verarbeitet werden, z. B. durch Mittelwertbildung.

Bis zum Beenden der Archivierung (6) bestimmt der Archivierungszyklus (5), wann der verarbeitete Prozesswert in die Archivdatenbank geschrieben wird.

Variablenarchivierung auf Anforderung

Einleitung

Die Variablenarchivierung auf Anforderung speichert in Runtime den aktuellen Prozesswert nur auf Anforderung in der Archivdatenbank. Die Anforderung wird gesteuert durch eine Triggervariable. Die Triggervariable ist vom Datentyp "Bool". Die Archivierung erfolgt bei jedem Zustandswechsel der Triggervariablen. Zusätzlich können Grenzwerte festgelegt werden. Der Prozesswert wird nur dann archiviert, wenn er innerhalb der festgelegten Grenzen liegt. Der Vergleich des Prozesswerts mit den Grenzwerten erfolgt nach der Erfassung des Prozesswerts.

Die Variablenarchivierung auf Anforderung endet mit dem Beenden von Runtime.

Funktionsweise

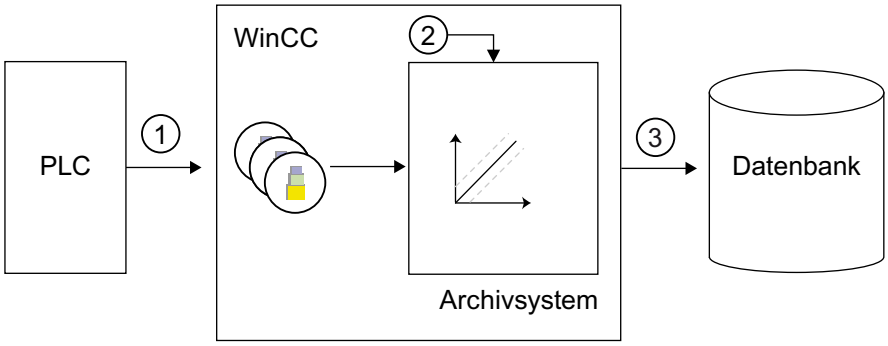

Die externen Variablen in WinCC entsprechen einem bestimmten Prozesswert im Speicher einer angeschlossenen PLC-Steuerung. Beim Eintritt der Anforderung durch die Triggervariable (2) wird der Prozesswert aus der PLC-Steuerung ausgelesen (1). Die Runtime-Komponente des Archivsystems verarbeitet den Prozesswert. Der Momentanwert des Prozesswerts wird dann in die Archivdatenbank geschrieben (3).

Bei jeder neuen Anforderung durch die Triggervariable (2) wird der Prozesswert neu aus dem Speicher der angeschlossenen PLC-Steuerung ausgelesen (1), verarbeitet und archiviert (3).

Variablenarchivierung bei Änderung

Einleitung

Die Variablenarchivierung bei Änderung speichert in Runtime den aktuellen Prozesswert in der Archivdatenbank nur, wenn sich der Prozesswert ändert. Zusätzlich können Grenzwerte festgelegt werden. Der Prozesswert wird nur dann archiviert, wenn er innerhalb der festgelegten Grenzen liegt. Der Vergleich des Prozesswerts mit den Grenzwerten erfolgt nach der Erfassung des Prozesswerts.

Die Variablenarchivierung bei Änderung endet mit dem Beenden von Runtime.

Funktionsweise

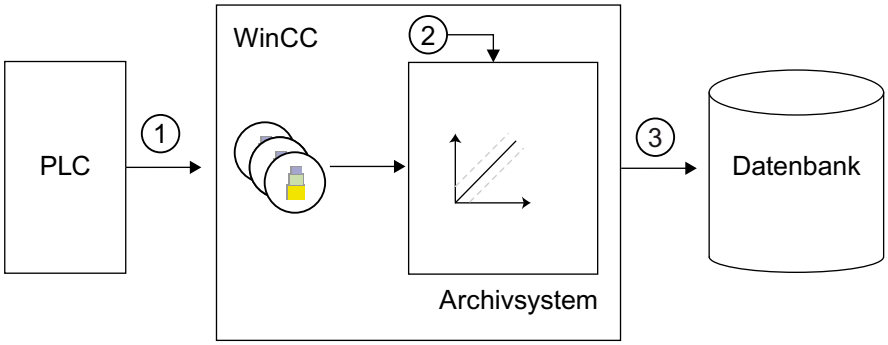

Die externen Variablen in WinCC entsprechen einem bestimmten Prozesswert im Speicher einer angeschlossenen PLC-Steuerung. Der Prozesswert wird zyklisch aus dem Speicher der angeschlossenen PLC-Steuerung ausgelesen (1) und auf Änderung überwacht. Bei einer Änderung (2) der externen Variablen wird der Prozesswert archiviert.

Die Runtime-Komponente des Archivsystems verarbeitet den Prozesswert.

Der Momentanwert des Prozesswerts wird dann in die Archivdatenbank geschrieben (3).

Verdichtungsarchiv

Einleitung

Zur Reduzierung der Datenmenge in der Archivdatenbank verdichten Sie die Archivvariablen eines bestimmten Zeitraums. Dazu legen Sie ein Verdichtungsarchiv an, das jede Archivvariable in einer Verdichtungsarchivvariable speichert. Die Archivvariablen bleiben erhalten, können aber auch kopiert, verschoben oder gelöscht werden. Das Verdichtungsarchiv wird wie das Variablenarchiv in der Archivdatenbank gespeichert.

Funktionsweise

Die Verdichtung wird durch die Anwendung von mathematischen Funktionen erreicht. Dazu wird auf die archivierten Prozesswerte eines angegebenen Zeitraums eine der folgenden Funktionen angewendet:

- Maximalwert: Speichert den größten Prozesswert in der Verdichtungsarchivvariable.
- Minimalwert: Speichert den kleinsten Prozesswert in der Verdichtungsarchivvariable.
- Summe: Speichert die Summe der Prozesswerte in der Verdichtungsarchivvariable.
- Mittelwert: Speichert den Mittelwert der Prozesswerte in der Verdichtungsarchivvariable.
- Gewichteter Mittelwert: Speichert den gewichteten Mittelwert der Prozesswerte in der Verdichtungsarchivvariable. Dabei wird die Zeitdauer, in der sich der Wert einer Variablen nicht geändert hat, in die Berechnung einbezogen. Für Variablen, die "Bei Änderung" erfasst werden, liefert diese Funktion aussagekräftigere Ergebnisse als die Funktion "Mittelwert".

Wie nach der Verdichtung mit den alten, archivierten Prozesswerten verfahren wird, hängt von der verwendeten Verdichtungsmethode ab:

- Berechnen: Die Prozesswerte der Archivvariablen aus dem angegebenen Zeitraum werden ausgelesen und verdichtet. Die Prozesswerte der Archivvariablen bleiben erhalten.
- Berechnen und kopieren: Die Prozesswerte der Archivvariablen aus dem angegebenen Zeitraum werden ausgelesen, verdichtet und zusätzlich in das Verdichtungsarchiv kopiert.
- Berechnen und löschen: Die Prozesswerte der Archivvariablen aus dem angegebenen Zeitraum werden ausgelesen, verdichtet und anschließend gelöscht.
- Berechnen, kopieren und löschen: Die Prozesswerte der Archivvariablen aus dem angegebenen Zeitraum werden ausgelesen, verdichtet, in das Verdichtungsarchiv kopiert und anschließend gelöscht.

Beispiel

Folgendes Beispiel verdeutlicht die Funktionsweise des Verdichtungsarchivs:

Ein Prozesswert wird 1-mal pro Minute archiviert und liefert in einer Stunde 60 Werte. Die Verdichtung erfolgt über den Zeitraum von einer Stunde, z. B. durch Mittelwertbildung. Demnach wird jede Stunde der Mittelwert aus den 60 Werten gebildet und in der Verdichtungsarchivvariable gespeichert. Wie mit den 60 Werten verfahren wird, hängt von der oben genannten Verdichtungsmethode ab.

Siehe auch

Verdichtungsarchiv anlegen (Seite 679)

Verdichtungsarchivvariable erstellen (Seite 685)

Speicherung von Prozesswerten

Einleitung

Prozesswerte speichern Sie entweder auf der Festplatte in der Archivdatenbank oder im Hauptspeicher der Runtime-Archivierung.

Speicherung in der Archivdatenbank

Die zu archivierenden Prozesswerte werden in zwei getrennten Umlaufarchiven in der Archivdatenbank gespeichert.

- Variablenarchiv Fast (A)
- Variablenarchiv Slow (B)

Jedes Umlaufarchiv besteht aus einer projektierbaren Anzahl von Segmenten. Für die Segmente legen Sie eine Größe in MB und ein Zeitraum fest, z. B. ein Tag.

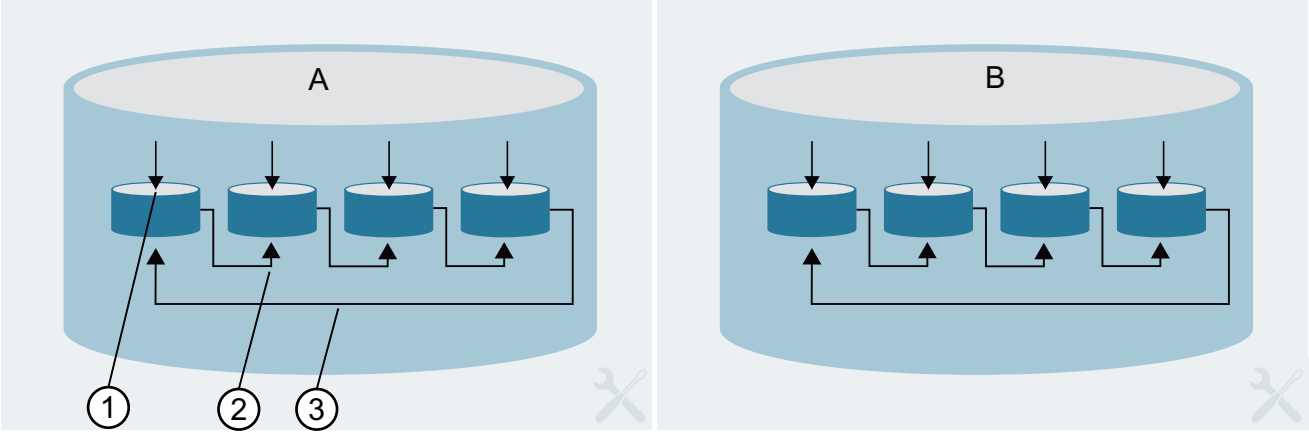

Die Prozesswerte werden fortlaufend in das erste Segment geschrieben (1). Ist die projektierte Größe des Segments erreicht oder die Zeitspanne überschritten, wird zum nächsten Segment gewechselt (2). Wenn alle Segmente voll sind, werden die Prozessdaten des ersten Segments überschrieben (3). Damit Prozessdaten durch Überschreiben nicht verloren gehen, können sie ausgelagert werden.

Das "Variablenarchiv Fast" speichert Prozesswerte, deren Erfassungszyklus kleiner oder gleich einer Minute ist. Diese Prozesswerte werden zunächst in einer Binärdatei gespeichert und komprimiert. Erst wenn die Binärdatei eine bestimmte Größe erreicht hat, wird Sie im Umlaufarchiv gespeichert.

Das "Variablenarchiv Slow" speichert Prozesswerte, deren Erfassungszyklus größer als eine Minute ist. Das "Variablenarchiv Slow" speichert auch Verdichtungsarchive. Diese Daten werden sofort in das Umlaufarchiv geschrieben und nicht komprimiert.

Hinweis

Beim Start von Runtime prüft das System, ob die projektierte Größe des Datenpuffers ausreichend bemessen ist. Bei zu gering projektierter Größe passt das System die Mindestgröße automatisch an.

Speicherung im Hauptspeicher

Im Gegensatz zur Speicherung in der Archivdatenbank sind die im Hauptspeicher archivierten Prozesswerte nur so lange verfügbar, wie Runtime aktiv ist. Die Speicherung im Hauptspeicher hat allerdings den Vorteil, dass die Werte sehr schnell geschrieben und ausgelesen werden. Die im Hauptspeicher gespeicherten Prozesswerte können nicht ausgelagert werden.

Hinweis

Verdichtungsarchive können nicht im Hauptspeicher gespeichert werden.

Optimierte Projektierung

Mithilfe der verfügbaren Archivarten und den unterschiedlichen Speicherverfahren können Sie die Archivierung an die Dynamik der sich ändernden Prozesswerte anpassen.

Wählen Sie die Archivierungsart entsprechend der Häufigkeit der Wertänderungen einer Variable und der Notwendigkeit zur Archivierung.

Kombinieren Sie die Archivierungsart mit der Auswahl des passenden Umlaufarchivs. Wählen Sie z. B. kurze Archivierungszyklen nur für Variablen, deren Werte sich schnell ändern.

Auslagerung von Prozesswerten

Einleitung

Sie können Prozesswerte aus der Archivdatenbank als Backup auslagern. Es werden alle Prozesswerte auslagert, die in einem Archivsegment enthalten sind. Ein Archivsegment wird immer dann ausgelagert, wenn es voll ist und ein neues Segment begonnen wird. Ein Archivsegment wird auch dann ausgelagert, wenn der projektierte Zeitpunkt für einen Segmentwechsel erreicht ist und ein neues Segment begonnen wird.

Prinzip

Die Auslagerung von Prozesswerten projektieren Sie in den Einstellungen der Archivierung. In den Einstellungen konfigurieren Sie die Größe des Archivs und der Archivsegmente sowie den Zeitpunkt für den Segmentwechsel. In den Einstellungen legen Sie auch fest, ob ein Backup gemacht wird.

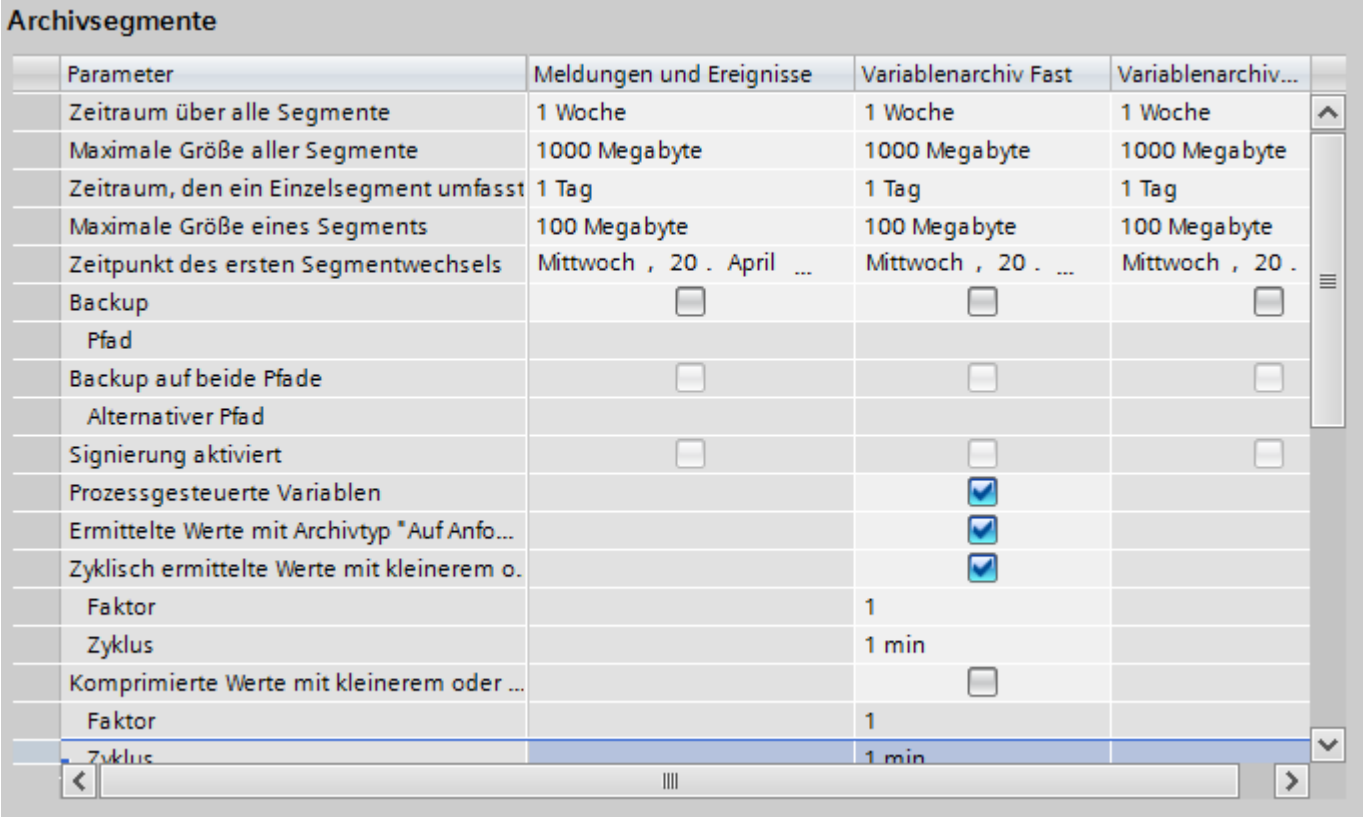

Archivsegmente

Parameter	Meldungen und Ereignisse	Variablenarchiv Fast	Variablenarchiv...
Zeitraum über alle Segmente	1 Woche	1 Woche	1 Woche
Maximale Größe aller Segmente	1000 Megabyte	1000 Megabyte	1000 Megabyte
Zeitraum, den ein Einzelsegment umfasst	1 Tag	1 Tag	1 Tag
Maximale Größe eines Segments	100 Megabyte	100 Megabyte	100 Megabyte
Zeitpunkt des ersten Segmentwechsels	Mittwoch , 20 . April ...	Mittwoch , 20	Mittwoch , 20 .
Backup	☐	☐	☐
Pfad			
Backup auf beide Pfade	☐	☐	☐
Alternativer Pfad			
Signierung aktiviert	☐	☐	☐
Prozessgesteuerte Variablen		☑	
Ermittelte Werte mit Archivtyp "Auf Anfo...		☑	
Zyklisch ermittelte Werte mit kleinerem o.		☑	
Faktor		1	
Zyklus		1 min	
Komprimierte Werte mit kleinerem oder ...		☐	
Faktor		1	
Zyklus		1 min	

In den Zeilen für die Backup-Konfiguration legen Sie fest, wo das Backup erstellt wird und ob die Backup-Datei signiert wird.

Hinweis

In Runtime können Sie über die Tabellenanzeige einen angezeigten Prozesswert ändern.

Wenn der Prozesswert aus dem Variablenarchiv bereits ausgelagert ist, wird der geänderte Wert nicht in das ausgelagerte Archiv übernommen. Die Änderung liegt nur im lokalen Archivsegment.

Wenn das Archivsegment noch nicht ausgelagert ist, wird der geänderte Wert dauerhaft übernommen.

Siehe auch

Einstellungen zur Archivierung (Seite 687)

Bedeutung der Flags von Archivwerten

Von der Variablenarchivierung werden Flags gesetzt zu jedem Wert, der ins Archiv geschrieben wird. Die Flags geben Aufschluss über den Zustand der Variablen.

Die Flags werden in einem 2 - Wort - Wert dargestellt. Der Wert ist dezimal codiert und wird in der 3.Spalte des Archivs der Datenbank geführt wird. Zur Analyse der Flags muss der Wert in seine hexadezimale Darstellung umgewandelt werden.

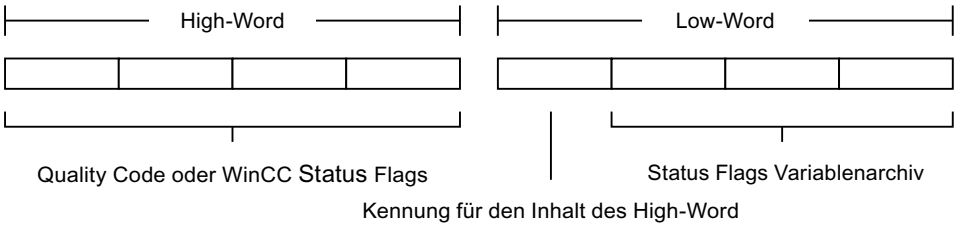

Aufbau des 2 - Wort - Werts:

Das High-Word enthält die WinCC Status Flags oder den Quality Code. Das Low-Word enthält die Status Flags der Variablenarchivierung und eine Kennung für den Inhalt des High-Word.

Kennung für den Inhalt des High-Word:

	Bedeutung
0x0	High-Word enthält WinCC Status Flags
0x1	High-Word enthält Quality Code

Status Flags der Variablenarchivierung:

Name des Flags	Wert	Bedeutung
PDE_RT_DAYLIGHT	0x001	Sommerzeit
PDE_RT_SUBSTITUTION	0x002	Ersatzwert
PDE_RT_TIME_BEVOR_JUMP	0x004	Wert vor einem Zeitsprung
PDE_RT_TIME_BEHIND_JUMP	0x008	Wert nach einem Zeitsprung
PDE_RT_TIME_OVERLAPPED	0x010	Werte während einer Zeitüberlappung
PDE_RT_LOAD_SYSTEM	0x020	Erstmalig nach dem Anlegen des Archivs archivierter Wert
PDE_RT_RELOAD_SYSTEM	0x040	Erster Wert nach Aktivierung von RT
PDE_RT_CMPCOPY	0x080	Verdichteter Wert
PDE_RT_TIME_CHANGED	0x100	Zeitänderung hat stattgefunden
PDE_RT_HAND	0x200	Variablenversorgung von Hand

Beispiele

Wert in der Datenbank	16842753
Hexadezimale Darstellung	0101 0001
Kennung des High-Word	0: High-Word enthält WinCC Status Flag
Status Flag der Variablenarchivierung	001: Sommerzeit
WinCC Status Flag	0101: Verbindung zum Partner nicht aufgebaut; Initialisierungswert der Variablen

Wert in der Datenbank	266242
Hexadezimale Darstellung	0004 1002
Kennung des High-Word	1: High-Word enthält Quality Code
Status Flag der Variablenarchivierung	002: Ersatzwert
Quality Code	0004: Configuration Error, value not accepted

2.9.3.3 Arbeiten mit Variablenarchiven

Variablenarchiv anlegen

Archivieren von Variablen

Prinzip

Bei der Projektierung von Archiven wird zwischen den folgenden Archivtypen unterschieden:

- Das Variablenarchiv speichert Prozesswerte in Archivvariablen. Bei der Projektierung des Variablenarchivs wählen Sie die zu archivierenden Prozessvariablen sowie den Speicherort aus.
- Das Verdichtungsarchiv verdichtet Archivvariablen aus Variablenarchiven. Das Verdichtungsarchiv kann auch bestehende Verdichtungsarchive weiter verdichten. Bei der Projektierung des Verdichtungsarchivs wählen Sie eine Berechnungsmethode sowie den Verdichtungszeitraum.

Datenpuffer

Für ein Variablenarchiv legen Sie fest, ob der Datenpuffer auf der Festplatte oder im Hauptspeicher abgelegt wird.

Im Gegensatz zur Speicherung in der Archivdatenbank sind die im Hauptspeicher archivierten Prozesswerte nur so lange verfügbar, wie Runtime aktiv ist. Die Speicherung im Hauptspeicher hat allerdings den Vorteil, dass die Werte sehr schnell geschrieben und ausgelesen werden. Die im Hauptspeicher gespeicherten Prozesswerte können nicht ausgelagert werden.

Hinweis

Verdichtungsarchive können nur auf der Festplatte gespeichert werden.

Siehe auch

Einstellungen zur Archivierung (Seite 687)

Archivvariable projektieren (Seite 699)

Variablenarchiv anlegen

Einleitung

Die Projektierung eines Variablenarchivs gliedert sich in folgende Schritte:

- Neues Variablenarchiv erstellen
- Variablenarchiv konfigurieren, z. B. den Speicherort auswählen
- Auswahl der Variablen für die Archivierung

Voraussetzung

- Ein Projekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein Variablenarchiv anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Archive".
   Der Editor für Variablenarchive und Verdichtungsarchive wird geöffnet.
2. Klicken Sie auf die Registerkarte "Variablenarchive" und doppelklicken Sie im Editor "Variablenarchive" in der Spalte "Name" auf "Hinzufügen".
   Ein neues Variablenarchiv wird angelegt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Geben Sie im Feld "Name" einen eindeutigen Namen für das Archiv ein.
5. Wählen Sie im Feld "Speicherort" aus, ob die Prozesswerte im Hauptspeicher oder in einer Datenbank gespeichert werden.
6. Wenn Sie als Ablageort den Eintrag "Speicher" gewählt haben, legen Sie im Feld "Anzahl Datensätze" die Anzahl der zu archivierenden Datensätze pro Archivvariable fest.
7. Legen Sie im Bereich "Status" über die Option "Gesperrt" fest, ob der Archivierungsprozess beim Start von Runtime gesperrt ist.
   Wenn Sie die Archivierung in Runtime starten wollen, müssen Sie die Option wieder deaktivieren.
8. Wenn sich der Status der Archivsperre ändert, können Sie sich über eine C-Funktion informieren lassen oder weiterführende Schritte projektieren. Wählen Sie dazu im Inspektorfenster unter "Eigenschaften > Ereignisse" eine vorhandene C-Funktion aus. Die C-Funktion wird in Runtime vom Archivserver bei Änderung des Status "Gesperrt" aufgerufen. Die C-Funktion müssen Sie vorher im Editor "Skripte" erzeugen. Nähere Hinweise hierzu finden Sie unter "Funktion zur Statuskontrolle der Archivsperre (Seite 701)". Über das Ereignis "Sperre geändert" kann nur eine C-Funktion abgearbeitet werden.
9. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften eines Archivs direkt in der Tabelle des Editors "Variablenarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Das Variablenarchiv wird angelegt.

In weiteren Schritten konfigurieren Sie übergeordnete Parameter für die Archivierung. Nähere Hinweise hierzu finden Sie im Kapitel "Archiveinstellungen festlegen (Seite 687)".

Hinweis

Der Archivname darf eine maximale Länge von 32 Zeichen haben.

Folgende Zeichen sind nicht erlaubt: '\\','\'', '.', ',', ';', ':', '!', '?', '"', '´', '`', '^', '~', '-', '+', '=', '/', '*', '#', '%', '&', '§', '°', '(', ')', '[', ']','{', '}', '<', '>', '|', ' ', 'ä', 'ö', 'ü', 'Ä', 'Ö', 'Ü', 'ß'

Großschreibung und Kleinschreibung werden nicht unterschieden.

Die Zeichen _ @ # $ dürfen nicht als erstes Zeichen des Namens verwendet werden.

Folgende Begriffe dürfen für die Namensvergabe nicht verwendet werden: "times", "userarchives", "project", "archives", "defaultarchives", "defaultvariables", "templates", "tabletemplateitems", "curvetemplateitems".

In einem Projekt müssen die Namen von Archiven eindeutig sein. Die Eindeutigkeit der Archivnamen muss auch gewährleistet sein, wenn für verschiedene Archive unterschiedliche Speicherorte gewählt werden. Für Meldearchive und Variablenarchive müssen Sie ebenfalls unterschiedliche Namen vergeben.

Siehe auch

Funktion zur Statuskontrolle der Archivsperre (Seite 701)

Archiveinstellungen festlegen (Seite 687)

Verdichtungsarchiv anlegen

Einleitung

Die Projektierung eines Verdichtungsarchivs gliedert sich in folgende Schritte:

- Erstellen Sie das neue Verdichtungsarchiv.
- Konfigurieren Sie das Verdichtungsarchiv, indem Sie z. B. die Bearbeitungsmethode und den Zeitraum für die Verdichtung auswählen.
- Wählen Sie die Archivvariablen aus, die Sie in das Verdichtungsarchiv aufnehmen wollen.

Voraussetzung

- Ein Projekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein Verdichtungsarchiv anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Archive".
   Der Editor für Variablenarchive und Verdichtungsarchive wird geöffnet.
2. Klicken Sie auf die Registerkarte "Verdichtungsarchive" und doppelklicken Sie im Editor "Verdichtungsarchive" in der Spalte "Name" auf "Hinzufügen".
   Ein neues Verdichtungsarchiv wird angelegt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
4. Geben Sie im Feld "Name" einen eindeutigen Namen für das Archiv ein.
5. Wählen Sie im Bereich "Verdichtung" die Bearbeitungsmethode und den Zeitraum für die Verdichtung aus.
   Als Verdichtungszeitraum werden Ihnen alle Zeiten angeboten, die im Editor "Zyklen" angelegt sind und >= 1 Minute sind. Wenn Ihnen der gewünschte Verdichtungszyklus nicht angeboten wird, projektieren Sie eine neue Zykluszeit. Anschließend wählen Sie den neuen Zyklus als Verdichtungszeitraum.
6. Legen Sie über die Option "Gesperrt" fest, ob der Archivierungsprozess beim Start von Runtime gesperrt ist.
   Wenn Sie die Archivierung in Runtime starten wollen, müssen Sie die Option wieder deaktivieren.
7. Wenn sich der Status der Archivsperre ändert, können Sie sich über eine C-Funktion informieren lassen oder weiterführende Schritte projektieren. Wählen Sie dazu im Inspektorfenster unter "Eigenschaften > Ereignisse" eine vorhandene C-Funktion aus. Die C-Funktion wird in Runtime vom Archivserver bei Änderung des Status "Gesperrt" aufgerufen. Die C-Funktion müssen Sie vorher im Editor "Skripte" erzeugen. Nähere Hinweise hierzu finden Sie unter "Funktion zur Statuskontrolle der Archivsperre (Seite 701)".
8. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften eines Archivs direkt in der Tabelle des Editors "Verdichtungsarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Das Verdichtungsarchiv wird angelegt.

Hinweis

Der Archivname darf eine maximale Länge von 32 Zeichen haben.

Folgende Zeichen sind nicht erlaubt: '\\','\", '.', ',', ';', ':', '!', '?', "", '´', '`', '^', '~', '-', '+', '=', '/', '*', '#', '%', '&', '§', '°', '(', ')', '[', ']','{', '}', '<', '>', '|', ' ', 'ä', 'ö', 'ü', 'Ä', 'Ö', 'Ü', 'ß'

Großschreibung und Kleinschreibung werden nicht unterschieden.

Die Zeichen _ @ # $ dürfen nicht als erstes Zeichen des Namens verwendet werden.

Folgende Begriffe dürfen für die Namensvergabe nicht verwendet werden: "times", "userarchives", "project", "archives", "defaultarchives", "defaultvariables", "templates", "tabletemplateitems", "curvetemplateitems".

In einem Projekt müssen die Namen von Archiven eindeutig sein. Für Verdichtungsarchive und Variablenarchive müssen Sie ebenfalls unterschiedliche Namen vergeben.

Siehe auch

Funktion zur Statuskontrolle der Archivsperre (Seite 701)

Verdichtungsarchiv (Seite 671)

Archivvariable anlegen

Verwendung von Archivvariablen

Prinzip

Archivvariablen speichern die zu archivierenden Prozesswerte. Über die Prozessvariable legen Sie fest, welche Prozessdaten archiviert werden. Der Datentyp wird aus der gewählten Prozessvariable automatisch übernommen. Mit dem Triggermodus projektieren Sie, ob eine Variable zeitabhängig oder ereignisabhängig archiviert wird.

In einem Verdichtungsarchiv wird jede verdichtete Archivvariable in einer separaten Verdichtungsarchivvariable gespeichert.

Prinzipielle Vorgehensweise

Bei binären und analogen Archivvariablen projektieren Sie den Triggermodus, z. B. zyklisch, sowie Erfassungszyklus und Archivierungszyklus. Abhängig vom Triggermodus wählen Sie die Parameter, welche die Archivierung starten oder beenden. Abhängig vom Archivvariablentyp projektieren Sie Darstellungsbegrenzungen und Parameter zur Bearbeitung des Prozesswerts.

Bei der Projektierung einer Verdichtungsarchivvariable wählen Sie die Bearbeitungsmethode aus, die aus der zu verdichtenden Archivvariable z. B. den Mittelwert bildet.

Hinweis

Wenn Sie eine Archivvariable löschen, das Projekt speichern und anschließend eine Variable mit dem Namen der gelöschten Variable neu anlegen, können Sie für die Anzeige und Archivierung nicht mehr auf die zuvor gespeicherten Werte der gelöschten Variable zugreifen. Grund: Die neu angelegte Archivvariable erhält eine neue ID. Auf die ID der gelöschten Archivvariable kann nicht mehr zugegriffen werden.

Binäre Archivvariable erstellen

Einleitung

Um eine binäre Archivvariable zu projektieren, legen Sie zuerst eine Archivvariable an und verbinden sie mit einer binären Prozessvariable. Der Datentyp der Archivvariable wird dann automatisch eingestellt.

Anschließend legen Sie die Eigenschaften für die Archivvariable fest.

Voraussetzung

- Ein Variablenarchiv ist angelegt.
- Die binäre Variable, für die Sie die Archivierung festlegen wollen, ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Archivvariable ist geöffnet.

Vorgehen

Um eine binäre Archivvariable anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle des Editors "Variablenarchive" das Variablenarchiv, in dem Sie die Archivvariable erstellen wollen.
2. Doppelklicken Sie in der Tabelle "Archivvariablen" in der Spalte "Name" auf "Hinzufügen". Eine Archivvariable wird angelegt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" und geben Sie im Feld "Name" einen eindeutigen Namen für die Archivvariable ein.
4. Wählen Sie im Feld "Prozessvariable" die binäre Variable aus, mit der die Archivvariable verbunden wird.
   Über die Option "Gesperrt" legen Sie fest, ob die Archivierung beim Start von Runtime frei gegeben ist oder nicht. Wenn Sie die Archivierung in Runtime starten wollen, müssen Sie die Option wieder deaktivieren.

5. Um die archivierten Werte für andere Zwecke weiter zu verwenden, können Sie den Wert der Archivvariable zusätzlich in eine interne Variable schreiben. Die Aktualisierung richtet sich nach der Zykluseinstellung der Archivvariablen. Wählen Sie im Feld "Archivierten Wert in Variable kopieren" die gewünschte Variable aus.
6. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Archivierungsart" und wählen Sie den gewünschten Archivtyp.
   Nähere Hinweise hierzu finden Sie im Kapitel "Triggermodus (Seite 665)".
7. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Bearbeitungsmethode" und wählen Sie die gewünschte Bearbeitungsmethode aus.
   Nähere Hinweise zur Auswahl finden Sie im Kapitel "Zyklen und Ereignisse (Seite 665)".
8. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften einer Archivvariable direkt in der Tabelle des Editors "Variablenarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Archivvariablen können Sie auch in der Variablentabelle auf der Registerkarte "Archivvariablen" projektieren.

Ergebnis

Die Prozesswerte der projektierten Variable werden in Runtime entsprechend der gewählten Einstellungen archiviert.

Siehe auch

Triggermodus (Seite 665)

Zyklen und Ereignisse (Seite 665)

Analoge Archivvariable erstellen

Einleitung

Um eine analoge Archivvariable zu projektieren, legen Sie zuerst eine Archivvariable an und verbinden sie mit einer analogen Prozessvariable. Der Datentyp der Archivvariable wird dann automatisch eingestellt.

Anschließend legen Sie die Eigenschaften für die Archivvariable fest.

Voraussetzung

- Das Variablenarchiv ist angelegt.
- Die analoge Variable, für die Sie die Archivierung festlegen wollen, ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Archivvariable ist geöffnet.

Vorgehen

Um eine analoge Archivvariable anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle des Editors "Variablenarchive" das Variablenarchiv, in dem Sie die Archivvariable erstellen wollen.
2. Doppelklicken Sie in der Tabelle "Archivvariablen" in der Spalte "Name" auf "Hinzufügen". Eine Archivvariable wird angelegt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" und geben Sie im Feld "Name" einen eindeutigen Namen für die Archivvariable ein.
4. Wählen im Feld "Prozessvariable" die analoge Variable aus, mit der die Archivvariable verbunden wird.
   Über die Option "Gesperrt" legen Sie fest, ob die Archivierung beim Start von Runtime frei gegeben ist oder nicht. Wenn Sie die Archivierung in Runtime starten wollen, müssen Sie die Option wieder deaktivieren.
5. Um die archivierten Werte für andere Zwecke weiter zu verwenden, können Sie den Wert der Archivvariable zusätzlich in eine interne Variable schreiben. Die Aktualisierung richtet sich nach der Zykluseinstellung der Archivvariablen. Wählen Sie im Feld "Archivierten Wert in Variable kopieren" die gewünschte Variable aus.
6. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Archivierungsart" und wählen Sie den gewünschten Archivtyp.
   Nähere Hinweise hierzu finden Sie im Kapitel "Triggermodus (Seite 665)".
7. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Totzone für Archivierung" und geben Sie die gewünschten Grenzwerte ein. Um die Eingabe zu aktivieren, öffnen Sie das Auswahlfeld und wählen den Eintrag "Konstante". Geben Sie anschließend den gewünschten Grenzwert ein. Prozesswerte, die außerhalb der Grenzwerte liegen, werden nicht archiviert.
8. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Bearbeitungsmethode" und wählen Sie die gewünschte Bearbeitungsmethode aus.
   Nähere Hinweise hierzu finden Sie im Kapitel "Zyklen und Ereignisse (Seite 665)".
   Bei der Bearbeitungsmethode "Berechnung" können Sie eine C-Funktion zum Umrechnen von Variablenwerten angeben. Die C-Funktion müssen Sie vorher im Editor "Skripte" erzeugen. Nähere Hinweise hierzu finden Sie unter "Umrechnung von Werten aus Archivvariablen (Seite 701)".
9. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften eines Archivs direkt in der Tabelle des Editors "Variablenarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Die Prozesswerte der projektierten Variable werden in Runtime entsprechend der gewählten Einstellungen archiviert.

Siehe auch

Triggermodus (Seite 665)

Zyklen und Ereignisse (Seite 665)

Umrechnung von Werten aus Archivvariablen (Seite 701)

Verdichtungsarchivvariable erstellen

Einleitung

In einer Verdichtungsarchivvariable speichern Sie die Prozesswerte einer Archivvariable oder einer Verdichtungsarchivvariable aus einem anderen Verdichtungsarchiv in verdichteter Form. Zur Verdichtung der enthaltenen Prozesswerte wählen Sie eine mathematische Funktion, mit der die Daten verdichtet werden.

Folgende Funktionen stehen für die Verdichtung zur Verfügung:

- Gewichteter Mittelwert
- Maximum
- Minimum
- Mittelwert
- Summe

Eine Verdichtungsarchivvariable legen Sie im Editor "Verdichtungsarchive" an.

Voraussetzung

- Ein Verdichtungsarchiv ist angelegt.
- Eine Archivvariable, die Sie verdichten wollen, ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Verdichtungsarchivvariable ist geöffnet.

Vorgehen

Um eine Verdichtungsarchivvariable anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "Archive" auf die Registerkarte "Verdichtungsarchive".
   Der Editor "Verdichtungsarchive wird geöffnet.
2. Selektieren Sie in der Tabelle des Editors "Verdichtungsarchive" das Verdichtungsarchiv, in dem Sie die Verdichtungsarchivvariable erstellen wollen.

3. Klicken Sie am rechten Fensterrand auf vertikale Registerkarte "Archivvariablen". Die Taskcard "Archivvariablen" wird geöffnet. In der Taskcard werden alle Objekte angezeigt, die Sie in das selektierte Verdichtungsarchiv aufnehmen können.

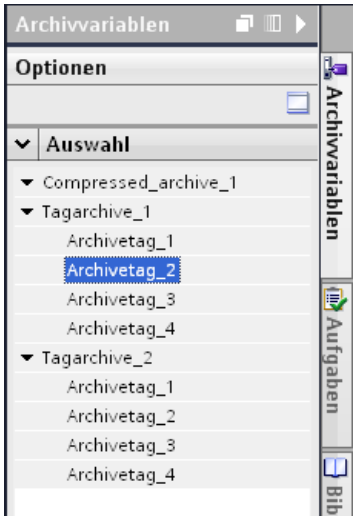

4. Selektieren Sie in der Taskcard die Objekte, die Sie in das Verdichtungsarchiv aufnehmen wollen. Folgende Objekte sind selektierbar:
   - Einzelne oder mehrere Archivvariablen
   - Ein komplettes Variablenarchiv
   - Einzelne oder mehrere Verdichtungsarchivvariablen
   - Ein komplettes Verdichtungsarchiv
5. Öffnen Sie das Kontextmenü der selektierten Objekte und wählen Sie den Menübefehl "Hinzufügen". Für jedes selektierte Objekt wird eine Verdichtungsarchivvariable angelegt und mit dem jeweiligen Objekt verbunden. Wenn Sie ein komplettes Variablenarchiv oder ein komplettes Verdichtungsarchiv selektiert haben, wird für jede in diesem Archiv enthaltene Variable eine Verdichtungsarchivvariable angelegt.
   Alternativ ziehen Sie die selektierten Objekte per Drag&Drop auf das in Schritt 2 selektierte Verdichtungsarchiv.
6. Selektieren Sie in der Tabelle Verdichtungsarchivvariablen eine oder mehrere Verdichtungsarchivvariablen. Legen Sie im Inspektorfenster weitere Eigenschaften für die selektierten Variablen fest.
   Über die Option "Gesperrt" legen Sie fest, ob die Archivierung beim Start von Runtime frei gegeben ist oder nicht. Wenn Sie die Archivierung in Runtime starten wollen, müssen Sie die Option wieder deaktivieren.
   Im Feld "Verdichtung" wählen Sie die Bearbeitungsmethode für die Verdichtung aus.
7. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Alternativ konfigurieren Sie die Eigenschaften einer Verdichtungsarchivvariable direkt in der Tabelle des Editors "Verdichtungsarchive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

Die Verdichtungsarchivvariablen werden mit den gewählten Archivvariablen verbunden. In Runtime werden die Werte aus der Archivvariable mit den eingestellten Parametern verdichtet.

Wenn Sie Verdichtungsarchivvariablen aus einem anderen Verdichtungsarchiv gewählt haben, werden diese noch weiter verdichtet.

Siehe auch

Verdichtungsarchiv (Seite 671)

Archiveinstellungen festlegen

Einstellungen zur Archivierung

Einleitung

Die logische Strukturierung von Variablenarchiven und Verdichtungsarchiven projektieren Sie im Editor "Archive". Die grundlegenden Eigenschaften von Archiven der Runtime Professional projektieren Sie im Dialog "Runtime-Einstellungen > Archivierung", z. B. die Segmentgröße.

Die Einstellungen für Variablenarchive nehmen sie in dem Dialog getrennt vor für Archive der Art "Variablenarchiv Fast" und "Variablenarchiv Slow".

Bei Verwendung der Standardeinstellungen gelten folgende Bedingungen:

Im Variablenarchiv Fast werden alle Werte gespeichert, deren Archivierungszyklus <= 1 Minute ist.

Im Variablenarchiv Slow werden alle Werte gespeichert, deren Archivierungszyklus > 1 Minute ist.

Die Zyklusgrenze für die Zuordnung zur Archivart ändern Sie im Dialog "Runtime-Einstellungen > Archivierung" im Feld "Zyklus".

Handlungsübersicht

Die übergeordnete Konfiguration der Archive projektieren Sie im Dialog "Runtime-Einstellungen > Archivierung". Die Beschreibung zur Projektierung entnehmen Sie den nachfolgend aufgeführten Anleitungen:

Schritt	Beschreibung
1	Archivgröße und Segmentierung festlegen (Seite 688)
2	Archiv-Backup konfigurieren (Seite 693)
3	Archivvariablen den Archivarten zuordnen (Seite 695)

Siehe auch

Archivgröße und Segmentierung festlegen (Seite 688)

Archiv-Backup konfigurieren (Seite 693)

Archiv-Backup verbinden (Seite 697)

Archiv-Backup trennen (Seite 698)

Archivvariablen den Archivarten zuordnen (Seite 695)

Archivieren von Variablen (Seite 677)

Auslagerung von Prozesswerten (Seite 674)

Archivgröße und Segmentierung festlegen

Einleitung

Im Dialog "Runtime-Einstellungen > Archivierung" legen Sie die Größe für die beiden Archivarten "Variablenarchiv Fast" und "Variablenarchiv Slow" in der Datenbank fest. Beide Archivarten können Sie zusätzlich in Segmente unterteilen. Mit der Größe der Einzelsegmente definieren Sie damit auch die Anzahl der Einzelsegmente.

In Runtime werden die einzelnen Segmente nacheinander befüllt. Wenn ein Segment vollständig befüllt ist, wird zum nächsten Segment gewechselt. Sie können den Segmentwechsel aber auch zeitabhängig projektieren. Wenn Sie einen Zeitpunkt für den Segmentwechsel festlegen, wird nach dem Erreichen des Zeitpunkts das nächste Archivsegment befüllt.

Hinweis

Achten Sie darauf, dass die Archivgröße nicht den freien Speicherplatz überschreitet. Das System prüft die gewählten Einstellungen nicht auf Plausibilität. Eine hohe Anzahl von verbundenen Archivsegmenten kann beim Starten und Beenden von Runtime zu längeren Wartezeiten im System führen.

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um die übergeordneten Einstellungen zur Archivierung zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner mit den Inhalten der Runtime Professional und doppelklicken Sie den Eintrag "Runtime-Einstellungen".
   Der Dialog für die Runtime-Einstellungen wird geöffnet.

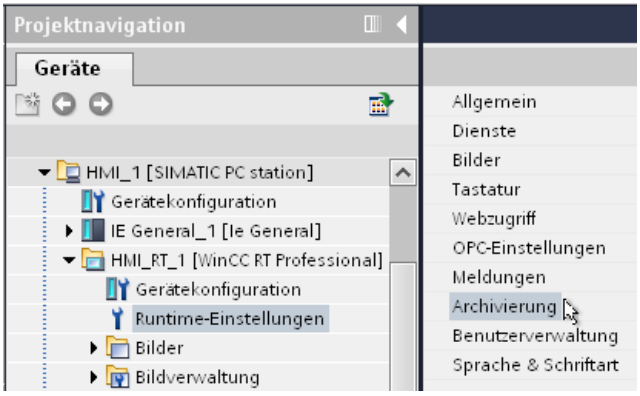

2. Klicken Sie im Dialog auf den Eintrag "Archivierung".
   Der Dialog "Archivierung" wird geöffnet.

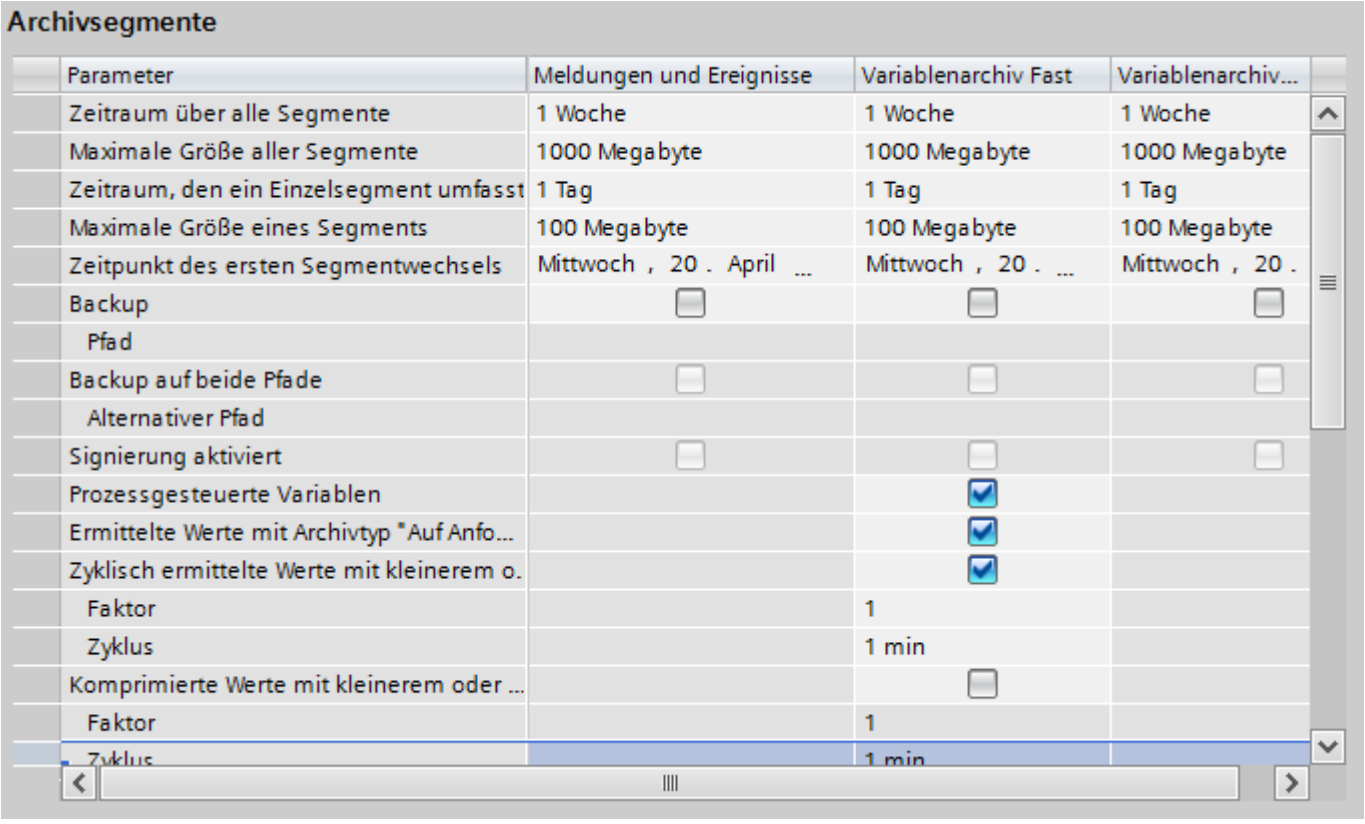

Archivsegmente

Parameter	Meldungen und Ereignisse	Variablenarchiv Fast	Variablenarchiv...
Zeitraum über alle Segmente	1 Woche	1 Woche	1 Woche
Maximale Größe aller Segmente	1000 Megabyte	1000 Megabyte	1000 Megabyte
Zeitraum, den ein Einzelsegment umfasst	1 Tag	1 Tag	1 Tag
Maximale Größe eines Segments	100 Megabyte	100 Megabyte	100 Megabyte
Zeitpunkt des ersten Segmentwechsels	Mittwoch , 20 . April ...	Mittwoch , 20	Mittwoch , 20 .
Backup	☐	☐	☐
Pfad			
Backup auf beide Pfade	☐	☐	☐
Alternativer Pfad			
Signierung aktiviert	☐	☐	☐
Prozessgesteuerte Variablen		☑	
Ermittelte Werte mit Archivtyp "Auf Anfo...		☑	
Zyklisch ermittelte Werte mit kleinerem o.		☑	
Faktor		1	
Zyklus		1 min	
Komprimierte Werte mit kleinerem oder ...		☐	
Faktor		1	
Zyklus		1 min	

3. Geben Sie in den Feldern "Zeitraum über alle Segmente" und "Maximale Größe aller Segmente" die gewünschten Werte ein.
   – Mit den Eingaben für den Zeitraum über alle Segmente sowie dessen maximale Größe legen Sie die Größe der Archivdatenbank fest. Wenn eines der beiden Kriterien überschritten wird, wird ein neues Segment begonnen. Wenn alle Segmente befüllt sind, wird das älteste Segment überschrieben.
4. Geben Sie in den Feldern "Zeitraum, den ein Einzelsegment umfasst" und "Maximale Größe eines Segments" die gewünschten Werte ein.
   – Mit den Eingaben zum Zeitraum für die Einzelsegmente sowie dessen maximale Größe legen Sie die Größe und Anzahl der Einzelsegmente fest. Wenn eines der Kriterien überschritten wird, wird ein neues Einzelsegment begonnen. Wenn das Kriterium für "Zeitraum über alle Segmente" überschritten wird, wird zusätzlich das älteste Einzelsegment gelöscht.
5. Im Feld "Zeitpunkt des ersten Segmentwechsels" geben Sie das Startdatum und die Startzeit für den ersten Segmentwechsel ein.
6. Fahren Sie in der Projektierung mit Schritt 2 der Handlungsübersicht fort und projektieren Sie das Archiv-Backup. Nähere Hinweise hierzu finden Sie im Kapitel "AUTOHOTSPOT".

Hinweis

Wenn Sie in Runtime die Archivgröße oder den Zeitraum ändern, wird die Änderung erst beim nächsten Segmentwechsel übernommen.

Beispiel

Folgende Eingaben sind projektiert:

Eigenschaft	Wert
Zeitraum über alle Segmente	1 Woche
Maximale Größe aller Segmente	700 Megabyte
Zeitraum, den ein Einzelsegment umfasst	1 Tag
Maximale Größe eines Segments	100 Megabyte
Zeitpunkt des ersten Segmentwechsels	Freitag, 23.November 2010 18:30

Mit der Projektierung wie in der Tabelle, wird das begonnene Segment am 23. November 2010 um 18:30 Uhr zum ersten Mal gewechselt. Die nächsten, zeitlich bedingten Segmentwechsel, erfolgen ab dem projektierten Zeitpunkt zyklisch nach jeweils 1 Tag. Das Segment wird auch dann gewechselt, wenn die projektierte Größe von 100 MB innerhalb eines Tages überschritten wird. Wenn die maximale Größe des Archivs von 700 MB überschritten wird, wird das älteste Einzelsegment gelöscht.

Archivart ändern

Wenn der Anwender nicht die entsprechenden Einstellungen ändert, dann werden Archivvariablen mit einer Zykluszeit <= 1 min im Archiv "Variablenarchiv Fast" archiviert. Archivvariablen mit einer Zykluszeit > 1 min werden im Archiv "Variablenarchiv Slow" archiviert. Die Zykluszeit einer Archivvariable darf über die o. g. Grenzen hinweg in Runtime nicht geändert werden. Der Wechsel einer Archivvariable von der Archivart "Fast" nach "Slow" und umgekehrt ist in Runtime nicht möglich.

Wenn Variablen nach einer Zyklusänderung oder Umprojektierung in einer anderen Archivart gespeichert werden, dann werden die Variablen aus dem gültigen Archiv gelesen. Ältere Archivwerte dieser Variablen sind in Runtime nicht mehr zugänglich.

Wenn Sie Archivvariablen so umprojektieren, dass deren Werte in einer anderen als der vorherigen Archivart erfasst werden, dann steigt damit auch der Speicherplatzbedarf im betreffenden Archiv. Passen Sie nach umfangreichen Änderungen die Archivgröße entsprechend an.

Hinweis

Die Runtime-Daten in den Archiven werden bei einem Reset in der Archivkonfiguration der Variablenarchivierung gelöscht. Nur bereits ausgelagerte Datenbanken bleiben erhalten.

Reset der Archivdatenbank

Die Einstellungen für ein Reset der Archivdatenbank erreichen Sie über den Dialog "Erweitertes Laden". Selektieren Sie dazu das HMI-Gerät in der Projektnavigation und wählen Sie den Menübefehl "Online > Erweitertes Laden in Gerät". Wählen Sie die Schnittstelle zu Ihrem HMI-Gerät aus. Klicken Sie auf "Laden". Der Dialog "Vorschau laden" wird geöffnet. Im Dialog "Vorschau Laden" treffen Sie die Einstellungen für den Reset.

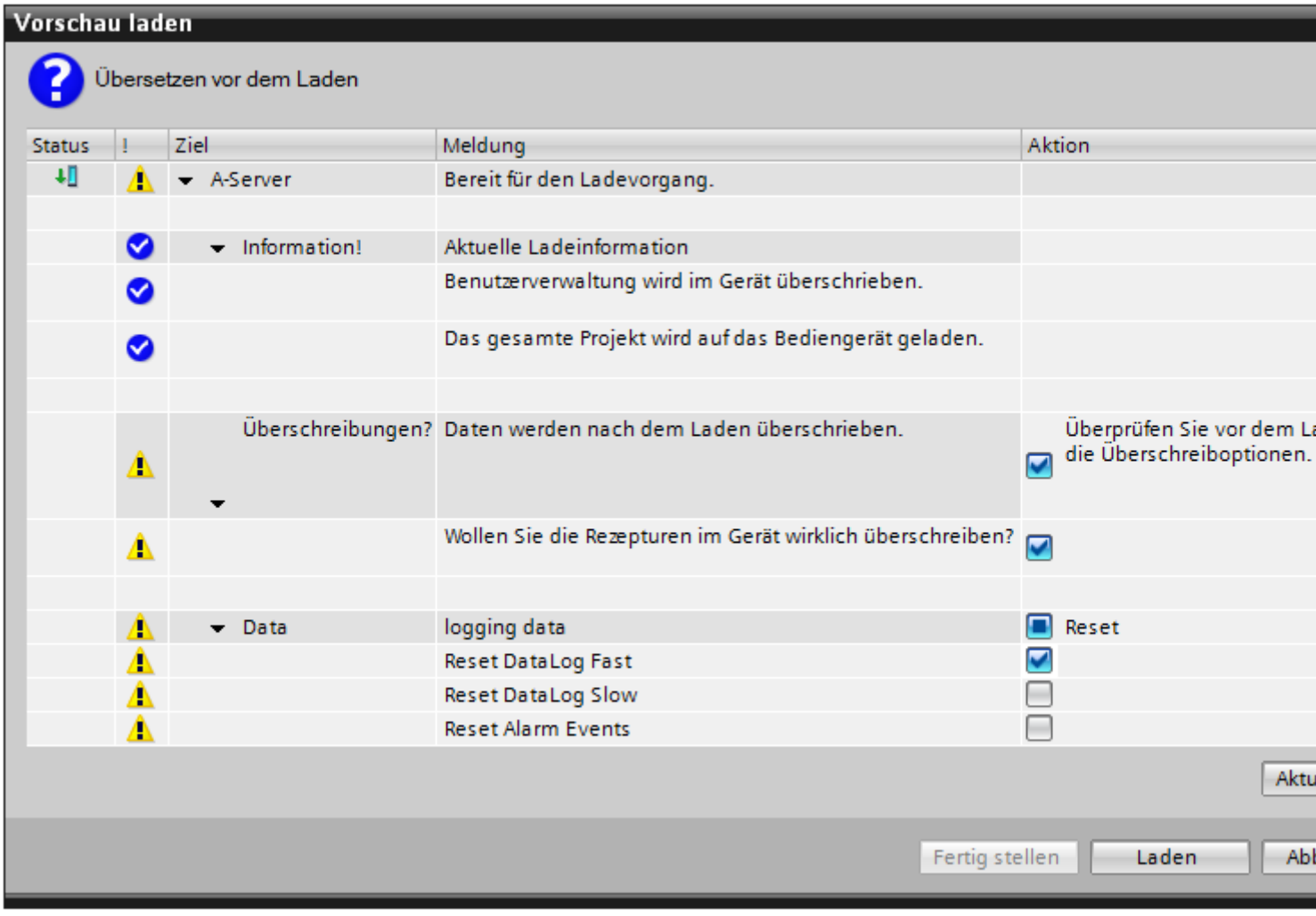

Wenn Sie anschließend den Ladevorgang starten, wird der Reset durchgeführt.

Archiv-Backup konfigurieren

Einleitung

Um eine lückenlose Dokumentation Ihres Prozesses sicherzustellen, erstellen Sie regelmäßige Backups Ihrer Archivdaten.

Hinweis

Das Backup wird standardmäßig eine Viertelstunde nach dem ersten zeitlich bedingten Segmentwechsel begonnen. Wenn sowohl der Start des Backups als auch der Segmentstart synchron mit dem Start von Runtime beginnen sollen, dann legen Sie den Zeitpunkt für den Segmentwechsel vor den Start von Runtime.

In Runtime können Sie über die Tabellenanzeige einen angezeigten Prozesswert ändern. Wenn das Archivsegment, in dem der Prozesswert gespeichert ist, bereits ausgelagert ist, wird der geänderte Wert nicht in das ausgelagerte Archiv übernommen. Die Änderung liegt nur im lokalen Archivsegment.

Wenn das Archivsegment noch nicht ausgelagert ist, wird der geänderte Wert dauerhaft übernommen.

Voraussetzung

- Schritt 1 der Handlungsübersicht ist abgeschlossen - Archivgröße und Segmentierung festlegen (Seite 688)
- Der Dialog "Runtime-Einstellungen > Archivierung" ist geöffnet.

Vorgehen

Um das Archiv-Backup zu konfigurieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie die Option "Backup" in der Spalte der Archivart, für die Sie das Backup konfigurieren wollen, z. B. in der Spalte "Variablenarchiv Fast".
2. Geben Sie in der gleichen Spalte, im Feld "Pfad", den Ablageort für das Backup ein oder navigieren Sie mithilfe des Dialogs zum gewünschten Ablageort im Dateisystem. Als Ablageort ist auch ein Netzlaufwerk möglich.
3. Wenn Sie das Backup redundant speichern wollen, aktivieren Sie die Option "Backup auf beide Pfade".
4. Geben Sie im Feld "Alternativer Pfad" einen alternativen Ablageort für das Backup ein oder navigieren Sie mithilfe des Dialogs zum gewünschten Ablageort im Dateisystem. Als Ablageort ist auch ein Netzlaufwerk möglich. Der alternative Ablageort wird in folgenden Fällen verwendet:
   - Wenn der Speicherplatz eines Backup-Mediums belegt ist.
   - Wenn der ursprüngliche Ablageort nicht verfügbar ist, z. B. wegen Netzwerkausfall.

   Wenn Sie entsprechende Systemmeldungen projektiert haben, werden Meldungen ausgegeben, wenn ein Ablageort nicht verfügbar ist.

5. Wenn die Archiv-Backup-Dateien eine Signatur erhalten sollen, aktivieren Sie die Option "Signierung aktiviert". Durch die Signatur erkennt das System beim erneuten Verbinden mit WinCC, ob an einer Archiv-Backup-Datei nach ihrer Auslagerung Änderungen vorgenommen wurden.
6. Fahren Sie in der Projektierung mit Schritt 3 der Handlungsübersicht fort und ordnen Sie die Archivvariablen den Archivarten zu. Nähere Hinweise hierzu finden Sie im Kapitel "Archivvariablen den Archivarten zuordnen (Seite 695)".

Aufbau der Archiv-Backup-Datei

Ein Archiv-Backup besteht aus zwei Dateien mit den Endungen "*.LDF" und "*.MDF". Um ein Archiv-Backup z. B. auf einen anderen Rechner zu übertragen, kopieren Sie die entsprechende LDF-Datei und MDF-Datei. Der Dateiname setzt sich wie folgt zusammen: "<Computername>_<Projektname>_<Typ>_<Zeitraum_von>_<Zeitraum_bis>". Der Typ bezeichnet den Archivtyp:

- TLG_F: Variablenarchiv mit einem Archivierungszyklus von einer Minute oder weniger.
- TLG_S: Variablenarchiv mit einem Archivierungszyklus von größer als einer Minute.

Der Zeitraum wird im folgenden Format angegeben: yyyymmddhhmm, z. B. 201012021118, was dem 2. Dezember 2010 um 11:18 Uhr entspricht. Unterstriche "_" im Projektnamen werden als "#" dargestellt.

Beispiel:

FLEX-LOG2_WINCC#10#11#29#14#37#40_TLG_F_201011291414_201012030502.ldf

FLEX-LOG2_WINCC#10#11#29#14#37#40_TLG_F_201011291414_201012030502.mdf

Signierung der Archiv-Backup-Dateien

Bei aktivierter Signierung und aktiviertem Backup wird jede Archiv-Backup-Datei bei ihrer Auslagerung signiert. Mit der Signatur wird beim erneuten Verbinden der Datei mit WinCC festgestellt, ob die Datei nach der Auslagerung verändert wurde.

Wenn Sie die Signierung von Archiven einsetzen, darf die maximale Größe eines Einzelsegments 200 MByte nicht überschreiten.

Zur Verifizierung der Daten muss die Option "Signierung aktiviert" aktiviert sein. Das Verbinden von Archiven der Art "Slow" dauert durch die Signierung länger.

Für die Archivierung von signierten Daten gelten folgende Werte:

Archivierung von signierten Daten in Runtime	Werte/Sekunde
Variablenarchiv Fast	1.000
Variablenarchiv Slow	500 [1)]

[1)] Bei Variablenarchiven vom Typ "Slow" müssen Sie bei gleichen Mengengerüsten mit längeren Bildanwahlzeiten rechnen als bei Variablenarchiven vom Typ "Fast".

Hinweis

Wenn Sie die Signierung deaktivieren, um z. B. eine schnelle Verbindung zu den Backup-Dateien aufzubauen, ist ein Segmentwechsel während der Deaktivierung zu vermeiden. Nach der Verbindung müssen Sie die Signierung wieder aktivieren, damit die zu archivierenden Daten eine Signatur erhalten.

Signierung aktivieren und deaktivieren

Die Signierung aktivieren und deaktivieren Sie in den Runtime-Einstellungen des HMI-Geräts.

Aktivieren oder deaktivieren Sie die Signierung. Wenn Runtime auf der Engineering-Station läuft, dann übersetzen Sie das Projekt neu. Die Änderung wird nach dem Übersetzen wirksam. Wenn Runtime auf einem anderen PC läuft als WinCC ES, übernehmen Sie die Änderung mit Online-Deltaladen. Nähere Hinweise hierzu finden Sie unter Übersicht zum Online-Deltaladen (Seite 2382).

Siehe auch

Archivgröße und Segmentierung festlegen (Seite 688)

Archivvariablen den Archivarten zuordnen (Seite 695)

Übersicht zum Online-Deltaladen (Seite 2382)

Archiv-Backup verbinden (Seite 697)

Archiv-Backup trennen (Seite 698)

Archivvariablen den Archivarten zuordnen

Einleitung

Die Variablenarchivierung verwendet für die Speicherung der Daten zwei unterschiedliche Archivarten:

- Variablenarchiv Fast
- Variablenarchiv Slow

Die Archivvariablen werden der jeweiligen Archivart automatisch zugeordnet. Diese Zuordnung ändern Sie in den Einstellungen für die Archivierung.

In Runtime wird eine geänderte Zuordnung erst übernommen, wenn das Projekt deaktiviert und Runtime neu gestartet wird.

Voraussetzung

- Schritt 2 der Handlungsübersicht ist abgeschlossen - Archiv-Backup konfigurieren (Seite 693)
- Der Dialog "Runtime-Einstellungen > Archivierung" ist geöffnet.

Vorgehen

Um die Archivvariablen den Archivarten zuzuordnen, gehen Sie folgendermaßen vor:

1. Wenn Sie die auf Anforderung ermittelten Werte im Variablenarchiv Fast archivieren wollen, dann aktivieren Sie in der Spalte "Variablenarchiv Fast" die Option "Ermittelte Werte mit Archivtyp "Auf Anforderung"".

Archivsegmente

Parameter	Meldungen und Ereignisse	Variablenarchiv Fast	Variablenarchiv...
Zeitraum über alle Segmente	1 Woche	1 Woche	1 Woche
Maximale Größe aller Segmente	1000 Megabyte	1000 Megabyte	1000 Megabyte
Zeitraum, den ein Einzelsegment umfasst	1 Tag	1 Tag	1 Tag
Maximale Größe eines Segments	100 Megabyte	100 Megabyte	100 Megabyte
Zeitpunkt des ersten Segmentwechsels	Mittwoch , 20 . April ...	Mittwoch , 20	Mittwoch , 20 .
Backup	☐	☐	☐
Pfad			
Backup auf beide Pfade	☐	☐	☐
Alternativer Pfad			
Signierung aktiviert	☐	☐	☐
Prozessgesteuerte Variablen		☑	
Ermittelte Werte mit Archivtyp "Auf Anfo...		☑	
Zyklisch ermittelte Werte mit kleinerem o.		☑	
Faktor		1	
Zyklus		1 min	
Komprimierte Werte mit kleinerem oder ...		☐	
Faktor		1	
Zyklus		1 min	

2. Wenn Sie Archivvariablen mit dem Triggermodus "Zyklisch" im Variablenarchiv Fast archivieren wollen, dann aktivieren Sie die Option "Zyklisch ermittelte Werte mit kleinerem oder gleichem Zyklus".
3. Wählen Sie im Feld "Zyklus" eine Zykluszeit als obere Grenze für den Archivierungszyklus ein. Bei Bedarf wählen Sie im Feld "Faktor" einen Faktor für die Multiplikation der gewählten Zykluszeit.
4. Wenn Sie alle verdichteten Werte mit einem Archivierungszyklus kleiner oder gleich der Zuordnungsgrenze für das Variablenarchiv Fast archivieren wollen, dann aktivieren Sie die Option "Komprimierte Werte mit kleinerem oder gleichem Zyklus".

5. Wählen Sie im Feld "Zyklus" eine Zykluszeit als obere Grenze für den Archivierungszyklus. Bei Bedarf wählen Sie im Feld "Faktor" einen Faktor für die Multiplikation der gewählten Zykluszeit.
6. Speichern Sie das Projekt.

Ergebnis

Alle Archivvariablen, für die diese Einstellungen zutreffen, werden im Variablenarchiv Fast archiviert. Die Archivvariablen, für die diese Einstellungen nicht zutreffen, werden im Variablenarchiv Slow archiviert.

Archiv-Backup verbinden

Einleitung

Wenn Sie auf die Daten eines Archiv-Backups in Runtime zugreifen wollen, verbinden Sie das Archiv-Backup wieder mit dem Projekt. Sie können die Verbindung über eine VB-Funktion herstellen oder automatisch eine Verbindung aufbauen lassen. Sie können alle Backup-Segmente verbinden, die mit dem jeweiligen Projekt erstellt wurden. Voraussetzung ist, dass keine Änderungen an dem Archiv oder den Archivvariablen durchgeführt wurden.

Voraussetzung

- Die entsprechende LDF-Datei und MDF-Datei des Archiv-Backups befinden sich in einem lokalen Ordner auf dem Projektierungsrechner, z. B. auf der Festplatte, MOD oder CD.

Vorgehen

Um ein Archiv-Backup mithilfe einer VB-Funktion zu verbinden, gehen Sie folgendermaßen vor:

1. Projektieren Sie eine Schaltfläche in einem Prozessbild und beschriften Sie die Schaltfläche z. B. mit "Archiv-Backup verbinden".
2. Schreiben Sie eine VB-Funktion unter Verwendung der Methode "Restore" nach folgendem Beispiel:
   HMIRuntime.Trace "Ret: " & HMIRuntime.Logging.Restore("D:\Backup","","2010-11-14",-1) & vbNewLine
   Nähere Hinweise hierzu finden Sie unter "AUTOHOTSPOT".
3. Projektieren Sie eine Funktionsliste an das Ereignis "Klicken" der projektierten Schaltfläche.
4. Wählen Sie in der Funktionsliste die erstellte VB-Funktion zur Ausführung aus.
5. Speichern Sie das Projekt.

Ergebnis

Die VB-Funktion wird in Runtime nach Klicken auf die Schaltfläche ausgeführt. Das Archiv-Backup wird mit den festgelegten Parametern aus der Funktion verbunden.

Die archivierten Prozesswerte aus dem Archiv-Backup werden in der projektierten Anzeige entsprechend dem Zeitstempel einsortiert.

Das Verbinden von "Tag Logging Slow"-Archiven dauert bei aktivierter Signatur länger.

Archiv automatisch verbinden lassen

1. Fügen Sie die Archiv-Backup-Dateien in den Ordner "CommonArchiving" ein.
   Den Ordner "CommonArchiving" finden Sie im Projektordner Ihres WinCC Projekts.
2. In Runtime wird das Variablenarchiv automatisch mit dem Projekt verbunden.

Wenn die Signierung aktiviert ist, werden veränderte, signierte Archiv-Backup-Dateien nicht automatisch verbunden. Eine WinCC Systemmeldung wird erzeugt und in der Windows Ereignisanzeige unter "Anwendung" wird ein Eintrag hinzugefügt.

Siehe auch

Archiv-Backup konfigurieren (Seite 693)

Archiv-Backup trennen (Seite 698)

Archiv-Backup trennen

Einleitung

Wenn Sie ein Archiv-Backup in Runtime verbunden haben, können Sie die Verbindung auch wieder trennen. Sie können die Verbindung über eine VB-Funktion trennen oder die Backup-Archiv-Dateien manuell aus dem Ordner "CommonArchiving" entfernen.

Voraussetzung

- Die entsprechende LDF-Datei und MDF-Datei des Archiv-Backups befinden sich in dem Ordner "CommonArchiving" im Projektordner des WinCC-Projekts.

Vorgehen

Um ein Archiv-Backup mithilfe einer VB-Funktion zu trennen, gehen Sie folgendermaßen vor:

1. Projektieren Sie eine Schaltfläche in einem Prozessbild und beschriften Sie die Schaltfläche z. B. mit "Archiv-Backup trennen".
2. Schreiben Sie eine VB-Funktion unter Verwendung der VBS-Methode "Remove" nach folgendem Beispiel:
   HMIRuntime.Trace "Ret: " & HMIRuntime.Logging.Remove("","2010-11-14",-1) & vbNewLine
   Nähere Hinweise hierzu finden Sie unter "AUTOHOTSPOT".
3. Projektieren Sie eine Funktionsliste an das Ereignis "Klicken" der projektierten Schaltfläche.
4. Wählen Sie in der Funktionsliste die erstellte VB-Funktion zur Ausführung aus.
5. Speichern Sie das Projekt.

Ergebnis

Die VB-Funktion wird in Runtime nach Klicken auf die Schaltfläche ausgeführt. Das Archiv-Backup wird von der Archivdatenbank getrennt. In Runtime haben Sie keinen Zugriff mehr auf die Daten aus der Backup-Datei.

Alternativ entfernen Sie die Backup-Dateien manuell aus dem Ordner "CommonArchiving". Den Ordner "CommonArchiving" finden Sie im Projektordner Ihres WinCC Projekts.

Siehe auch

Archiv-Backup konfigurieren (Seite 693)

Archiv-Backup verbinden (Seite 697)

Archivvariablen im Editor "Variablen"

Archivvariable projektieren

Einleitung

In WinCC können Sie im Editor "HMI-Variablen" auch Archivvariablen anlegen und bearbeiten. Die Eigenschaften der Archivvariablen projektieren Sie ebenfalls direkt im Editor "Variablen".

Hinweis

Wenn Sie im Editor "HMI-Variablen" löschen, verschieben oder kopieren, wirken sich diese Änderungen auch im Editor "Archive" aus.

Voraussetzung

- Im Editor "HMI-Variablen" ist die Registerkarte "Archivvariablen" geöffnet.
- Eine Variable ist angelegt.

Vorgehen

Um im Editor "HMI-Variablen" eine Archivvariable zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Variablentabelle eine bestehende Variable.
   Alternativ doppelklicken Sie in der Spalte "Name" auf "Hinzufügen ...", um eine neue Variable anzulegen.
2. Klicken Sie im unteren Editor auf "Archivvariablen". Der Editor für Archivvariablen wird in den Vordergrund gebracht.

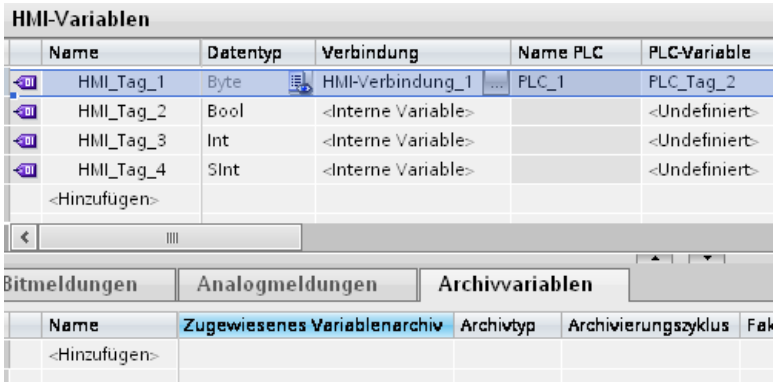

3. Doppelklicken Sie in Tabelle des Editors "Archivvariablen" in der Spalte "Name" auf "Hinzufügen ...". Eine neue Archivvariable wird angelegt. Die Archivvariable ist mit der im ersten Schritt selektierten Variable verbunden. Der Datentyp der Archivvariable wird automatisch aus der verbundenen Variable übernommen.

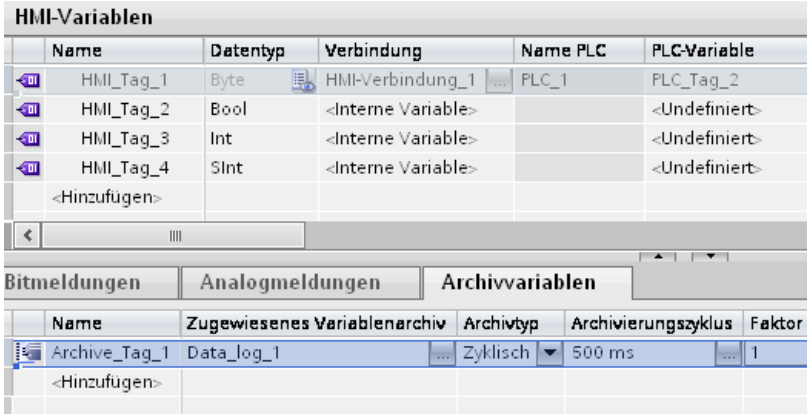

4. Wählen Sie im Feld "Variablenarchiv" das gewünschte Variablenarchiv aus. Alternativ erstellen Sie ein neues Archiv mithilfe der Objektliste.
5. Projektieren Sie die weiteren Parameter für die Archivierung in der Tabelle des Editors oder im Inspektorfenster.

Ergebnis

Die projektierte Archivvariable ist im Editor "Variablen" angelegt und wird in unterschiedlichen Editoren angezeigt.

Siehe auch

Archivieren von Variablen (Seite 677)

Funktionen bei der Variablenarchivierung

Funktion zur Statuskontrolle der Archivsperre

Einleitung

Den Zustand "Gesperrt" eines Variablenarchivs können Sie mithilfe einer C-Funktion überwachen. Die C-Funktion müssen Sie im Editor "Skripte" mit der korrekten Syntax erstellen.

Beschreibung

Diese Funktion wird vom Archivserver bei Änderung des Status "Gesperrt" ausgelöst. Sie kann im Editor "Variablenarchive" oder "Verdichtungsarchive" in der Gruppe "Ereignisse" dem Ereignis "Sperre geändert" zugeordnet werden. Die C-Funktion wird unter dem Ereignis "Sperre geändert" nur dann angezeigt, wenn die unten beschriebene Syntax in der Funktion korrekt eingegeben ist.

Syntax

void Funktionsname (BOOL fFlag);

Parameter

fFlag

TRUE - Archiv wurde gesperrt

FALSE - Archiv wurde freigeben

Verwendungsbeispiel

Über die Funktion können Sie sich z. B. über den Sperr-Status eines Archivs informieren lassen.

Umrechnung von Werten aus Archivvariablen

Beschreibung

Mit dieser C-Funktion können Variablenwerte vor der Archivierung umgerechnet werden. Diese C-Funktion wird im Editor "Archivvariablen" in den Eigenschaften der Archivvariable unter "Eigenschaften > Bearbeitungsmethode" zugeordnet. Die C-Funktion wird Ihnen im Feld "Analoge Verarbeitungsmethode" nur dann zur Auswahl angeboten, wenn die unten beschriebene Syntax in der Funktion korrekt eingegeben ist.

Syntax

double Funktionsname (double dLastValue,

double dCurrentValue,

DWORD dwCount,

BOOL fArchivingCycle);

Parameter

dLastValue: Rückgabewert der Funktion, die im letzten Erfassungszyklus getriggert wurde

dCurrentValue: aktuell erfasster Wert

dwCount: aktueller Erfassungszyklus seit dem letzten Archivierungszyklus (z. B. Archivierungszyklus ist 4 x 1 Sekunde Erfassungszyklus; dann gibt es einen Zählerstand von 1 bis 4, bis der Zählerstand erneut beginnt).

fArchivingCycle: TRUE = Zyklus ist Archivierungszyklus (default); FALSE = Zyklus ist Erfassungszyklus

Verwendungsbeispiel

Über diese Funktion können Sie zum Beispiel eine eigene Mittelwertsberechnung realisieren. Es wird dann nicht mehr der tatsächliche Variablenwert, sondern der berechnete Mittelwert realisiert.

Die folgende Funktion ermittelt das geometrische Mittel der erfassten Prozesswerte:

```
double CalculateGeometricMean (double dLastValue,double
dCurrentValue,DWORD dwCount,BOOL fArchivingCycle)
{

static double dCurrentProduct = 1.0;

    double dReturnValue = 0.0;

if (!fArchivingCycle)

{

// 1. continiously calculate the product of all values with every
scan cycle

dCurrentProduct *= dCurrentValue;

printf ("current value=%f current product=%f archiving cycle flag=%d
\r\n",dCurrentValue,dCurrentProduct ,fArchivingCycle);

dReturnValue = dCurrentProduct ; // optinonally - the return value
is actually not of interest in a scan cycle

}
```

```
else

{

// 2. if this is the archiving cycle, use the product of all scan
values to calculate

// the geometric mean value

dReturnValue = pow (dCurrentProduct ,(double) 1 / dwCount);

      printf ("archiving cycle: geometric mean value=%f\r
\n",dReturnValue);

         printf ("\r\n");

         dCurrentProduct = 1.0; // reset current product

// the finally calculated value will be returned - this value will
be archived

// since this is the archiving cycle

}

return dReturnValue;

}
```

2.10 Variablen darstellen

2.10.1 Variablen mit Basic Panels darstellen

2.10.1.1 Ausgeben von Variablenwerten in Bildern (Basic)

Einleitung

In Runtime können Sie Variablenwerte in den Bildern am Bediengerät in Form einer Kurve ausgeben. Eine Kurve ist die grafische Darstellung der Werte, die eine Variable in Runtime annimmt. Für die Darstellung verwenden Sie das Grafikobjekt "Kurvenanzeige". Prozesswerte für die Darstellung als Kurve werden aus dem laufenden Prozess von der Steuerung geladen.

Die darzustellenden Werte werden einzeln in einem festen, einstellbaren Zyklus ermittelt. Zyklisch getriggerte Kurven eignen sich für die Darstellung kontinuierlicher Verläufe, z. B. die Darstellung der Betriebstemperatur eines Motors.

Dargestellte Werte

Damit Variablenwerte am Bediengerät angezeigt werden, projektieren Sie in einem Bild eine Kurvenanzeige. Bei der Projektierung der Kurvenanzeige geben Sie an, welche Variablenwerte dargestellt werden.

Die Aktualisierung der Kurvenanzeige steuern Sie durch die Definition der Zykluszeit.

Siehe auch

Kurvenanzeige für Werte aus der Steuerung projektieren (Basic) (Seite 704)

Grundlagen zu Variablen (Seite 537)

2.10.1.2 Kurvenanzeige für Werte aus der Steuerung projektieren (Basic)

Einleitung

Mit einer Kurvenanzeige stellen Sie Werte grafisch dar, die eine Variable im Prozess annimmt.

Voraussetzung

- Ein Bild ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Kurvenanzeige ist geöffnet.

Vorgehen

Um eine Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus dem Werkzeugfenster aus der Gruppe "Control" das Objekt "Kurvenanzeige" in das Bild ein.

2. Selektieren Sie im Inspektorfenster in der Gruppe "Eigenschaften" die Kategorie "Kurve" und doppelklicken Sie in der Spalte "Name" den Eintrag "<Hinzufügen>".

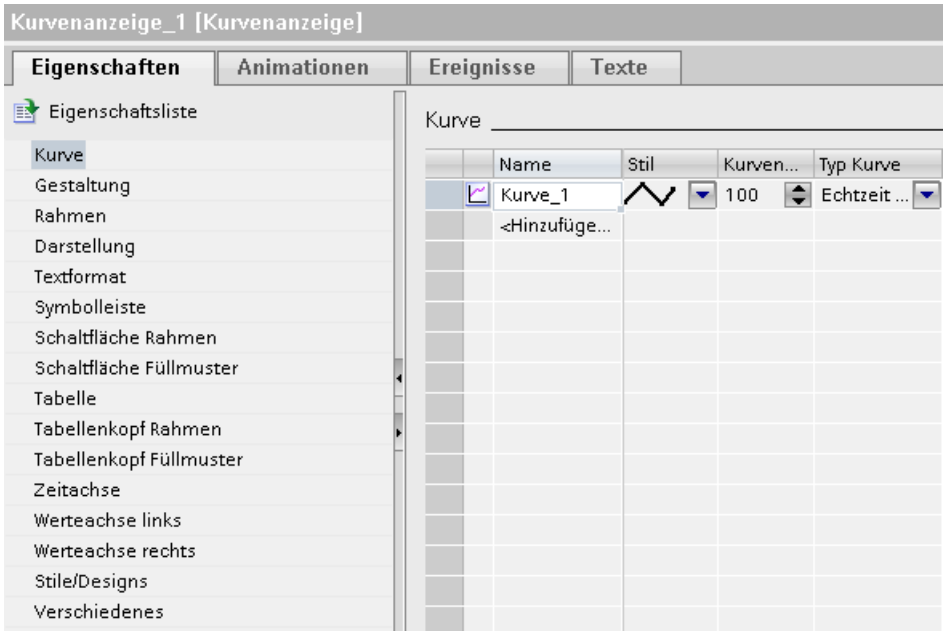

3. Vergeben Sie in der Spalte "Name" einen Namen für die Kurve.
4. Öffnen Sie in der Spalte "Stil" über die Auswahlschaltfläche den Dialog "Stil" und wählen Sie die Darstellung der Linie aus.
5. Wählen Sie in der Spalte "Kurvenwerte" die Anzahl der Kurvenwerte.

6. Öffnen Sie in der Spalte "Einstellungen" über die Auswahlschaltfläche den Dialog "Datenquelle" und wählen Sie die Variable für die Werteversorgung der Kurve aus. Geben Sie einen Zyklus für das Lesen der Variablen aus der Steuerung an.

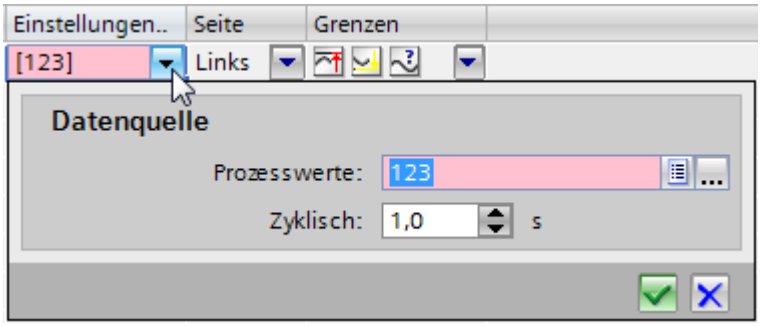

7. Nehmen Sie weitere Einstellungen in den Dialogen des Inspektorfensters vor. Aktivieren Sie z. B. in der Kategorie "Tabelle" die Option "Tabelle anzeigen", um unter der Kurvenanzeige eine Wertetabelle anzuzeigen.

Hinweis

Wenn Sie die Taste <STRG> gedrückt halten und einen Doppelklick auf die Kurvenanzeige machen, wird die Kurvenanzeige aktiv geschaltet. Im aktiven Modus können Sie im Tabellenkopf die Spaltenbreite und die Position der Spalten der Wertetabelle einstellen. Zur Aktivierung der Kurvenanzeige muss der Zoomfaktor auf 100 % eingestellt sein.

Ergebnis

In Runtime werden in der projektierten Kurvenanzeige die Werte der gewählten Variablen angezeigt.

Siehe auch

Ausgeben von Variablenwerten in Bildern (Basic) (Seite 704)

2.10.2 Variablen mit der Runtime Advanced und Panels darstellen

2.10.2.1 Kurven

Einleitung

Eine Kurve ist die grafische Darstellung der Werte, die eine Variable in Runtime annimmt. Um Kurven darzustellen, projektieren Sie in einem Bild Ihres Projekts eine Kurvenanzeige.

Bei der Projektierung der Kurvenanzeige legen Sie den Kurventyp der darzustellenden Werte fest:

- Archiv: zur Darstellung der archivierten Werte einer Variablen
- Echtzeit zyklisch: zur zeitgesteuerten Darstellung von Werten

- Echtzeit bitgetriggert: zur ereignisgesteuerten Darstellung von Werten
- Puffer bitgetriggert: zur ereignisgesteuerten Darstellung mit gepufferter Datenerfassung

Archivierte Werte darstellen

Die Kurvenanzeige zeigt in Runtime die Werte einer Variable aus einem Variablenarchiv an. Die Kurve gibt die archivierten Werte eines bestimmten Zeitfensters wieder. Um die gewünschte Information aus dem Archiv zu erhalten, kann der Bediener das Zeitfenster in Runtime verschieben.

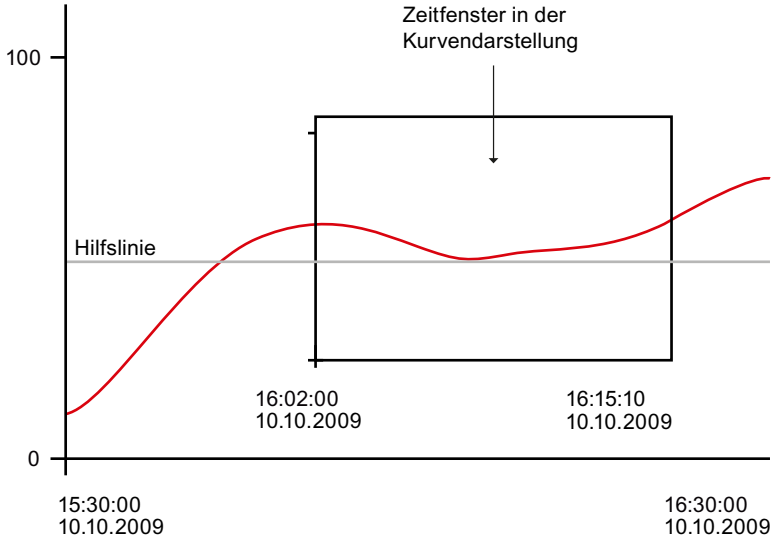

Zyklisch getriggerte Kurven

Die darzustellenden Werte werden einzeln in einem festen, einstellbaren Zeitraster ermittelt. Zyklisch getriggerte Kurven eignen sich für die Darstellung kontinuierlicher Verläufe, z. B. die Darstellung der Betriebstemperatur eines Motors.

Bitgetriggerte Kurven

Die darzustellenden Werte werden ereignisgesteuert, durch das Setzen eines definierten Bits in der Variablen "Kurvenübertragung", ermittelt. Nach dem Lesen wird das Bit wieder zurückgesetzt. Bitgetriggerte Kurven eignen sich zur Darstellung schnell veränderlicher Werte, z. B. zur Darstellung des Einspritzdrucks bei der Fertigung von Kunststoffteilen.

Bitgetriggerte Kurven mit gepufferter Datenerfassung

Bei der gepufferten Datenerfassung werden die darzustellenden Werte in der Steuerung zwischengespeichert und bitgetriggert als Block gelesen. Diese Kurven eignen sich zur Darstellung schneller Änderungen, wenn der Kurvenverlauf in seiner Gesamtheit interessant ist und weniger die einzelnen Werte.

Damit die Steuerung, während der Kurvenpuffer gelesen wird, die neu anfallenden Werte weiterhin schreiben kann, projektieren Sie in der Steuerung einen Wechselpuffer. Der

Wechselpuffer verhindert, dass die Steuerung Werte überschreibt, während das Bediengerät die Werte für die Kurve liest.

Der Wechsel zwischen Kurvenpuffer und Wechselpuffer funktioniert folgendermaßen:

Jedes Mal, wenn das der Kurve zugeordnete Bit in der Variable "Kurvenübertragung 1" gesetzt wird, werden alle Werte aus dem Kurvenpuffer gleichzeitig gelesen und als Kurve am Bediengerät dargestellt. Nach dem Lesen wird das Bit in "Kurvenübertragung 1" wieder zurückgesetzt.

Während das Bediengerät die Variablenwerte aus dem Kurvenpuffer liest, schreibt die Steuerung die neuen Variablenwerte in den Wechselpuffer. Wenn das der Kurve zugeordnete Bit in der Variablen "Kurvenübertragung 2" gesetzt wird, werden alle Kurvenwerte aus dem Wechselpuffer gelesen und am Bediengerät dargestellt. Während das Bediengerät den Wechselpuffer liest, schreibt die Steuerung wieder in den Kurvenpuffer.

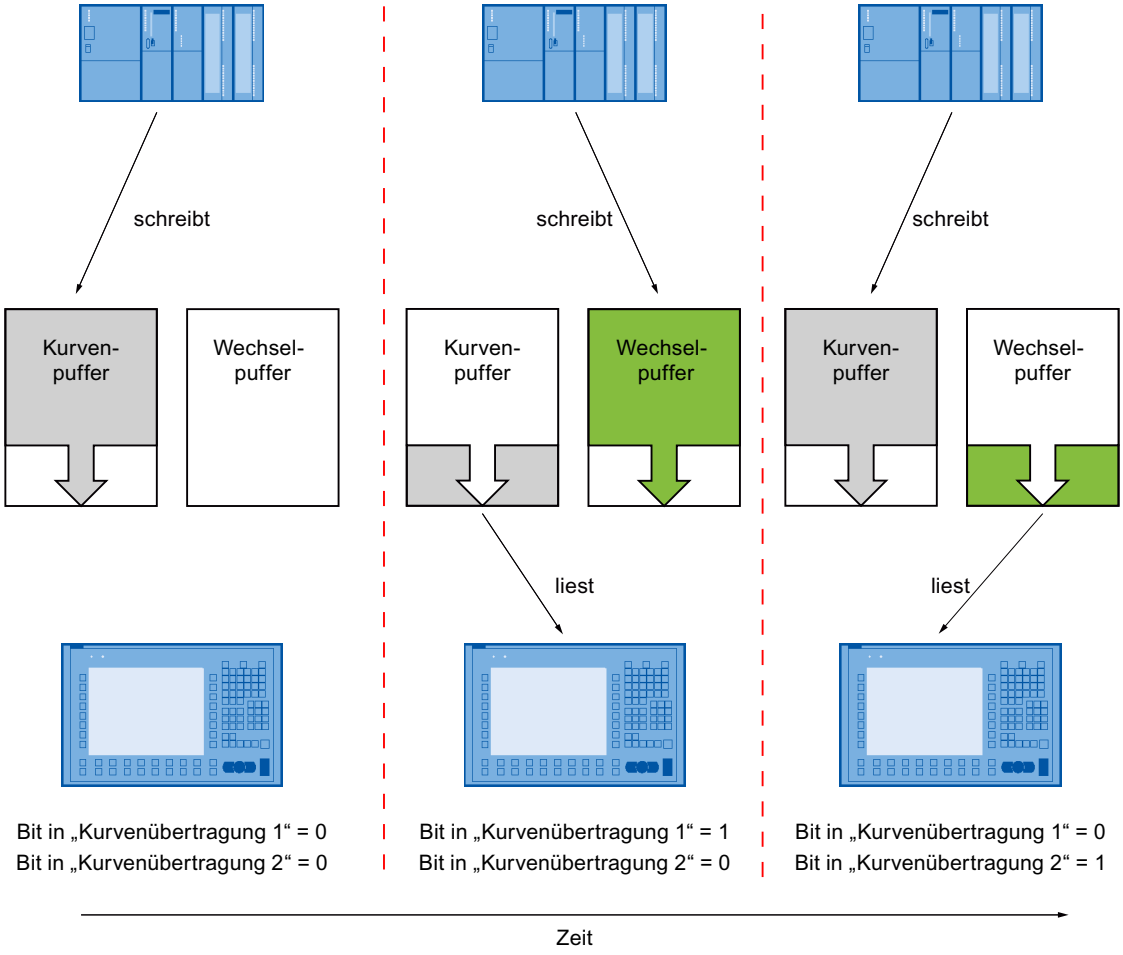

2.10.2.2 Ausgeben von Variablenwerten in Bildern

Einleitung

In Runtime können Sie Variablenwerte in den Bildern am Bediengerät in Form einer Kurve ausgeben. Die Prozesswerte werden aus dem laufenden Prozess von der Steuerung oder aus einer Archivdatenbank geladen.

Dargestellte Werte

Damit Variablenwerte am Bediengerät angezeigt werden, projektieren Sie in einem Bild eine Kurvenanzeige. Bei der Projektierung der Kurvenanzeige geben Sie an, welche Variablenwerte dargestellt werden.

Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- Aktuelle Werte aus der Steuerung
  Die Kurve wird um einzelne Werte aus der Steuerung fortgeschrieben. Für die Anzeige in Echtzeit stehen Ihnen folgende Trigger-Typen zur Verfügung:
  - Echtzeit bitgetriggert: Den Zeitpunkt der Aktualisierung steuern Sie durch das Setzen eines Bits. Bitgetriggerte Kurven werden in der Regel zur Darstellung sich schnell ändernder Werte verwendet. Ein Beispiel hierfür ist der Einspritzdruck bei der Fertigung von Kunststoffteilen.
  - Echtzeit zyklisch: Den Zeitpunkt der Aktualisierung steuern Sie über einen Zyklus. Zyklisch getriggerte Kurven eignen sich für die Darstellung kontinuierlicher Verläufe, z. B. die Änderung der Betriebstemperatur eines Motors.
  - Puffer bitgetriggert: Die Kurve zeigt Werte, die in der Zeit zwischen zwei Lesevorgängen von der Steuerung in einem Puffer abgelegt wurden. Wählen Sie diese Darstellungsform, wenn der Kurvenverlauf in seiner Gesamtheit interessant ist und weniger die einzelnen Werte.
- Archivierte Variablenwerte
  Die Kurvenanzeige zeigt in Runtime die Werte einer Variable aus einem Variablenarchiv an. Für die Anzeige von archivierten Variablenwerten steht Ihnen folgender Triggertyp zur Verfügung:
  - Variablenarchiv: Die Kurve gibt die archivierten Werte eines bestimmten Zeitfensters wieder. Um die gewünschte Information aus dem Archiv zu erhalten, verwendet der Bediener die entsprechenden Bedienelemente im Bild oder in der Kurvenanzeige.

2.10.2.3 Kurvenanzeige für Werte aus der Steuerung projektieren

Einleitung

Mit einer Kurvenanzeige stellen Sie Werte grafisch dar, die eine Variable im Prozess annimmt.

Voraussetzung

- Ein Bild ist geöffnet.

Vorgehen

Um eine Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus dem Werkzeugfenster aus der Gruppe "Controls" das Objekt "Kurvenanzeige" in das Bild ein.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Kurve" und doppelklicken Sie im Feld "Name" auf "Hinzufügen". Eine neue Kurve wird angelegt.
3. Geben Sie bei Bedarf im Feld "Name" einen eigenen Namen für die Kurve ein.
4. Öffnen Sie in der Spalte "Stil" über die Auswahlschaltfläche den Dialog "Stil" und wählen Sie die Darstellung der Linie aus.
5. Wählen Sie in der Spalte "Kurvenwerte" die Anzahl der Kurvenwerte.
6. Wählen Sie in der Spalte "Typ Kurve" den gewünschten Kurventyp aus. Folgende Einträge stehen zur Verfügung:
   - "Echtzeit bitgetriggert"
   - "Echtzeit zyklisch"
   - "Puffer bitgetriggert"
   - "Variablenarchiv"

7. Öffnen Sie in der Spalte "Einstellungen Quelle" über die Auswahlschaltfläche den Dialog "Datenquelle" und wählen Sie die Variable für die Werteversorgung der Kurve aus.
   Wenn Sie den Kurventyp "Echtzeit zyklisch" gewählt haben, geben Sie einen Zyklus für das Lesen der Variablen aus der Steuerung an.
   Nur für bitgetriggerte Kurven:
   - Geben Sie unter "Bit" an, das wievielte Bit der Variablen "Kurvenanforderung" und "Kurventransfer" der Kurve zugewiesen wird. Wählen Sie für jede Kurve einen anderen Wert.
   - Geben Sie unter "Kurvenanforderung" eine Variable an, die mindestens so viele Bits enthält, wie Kurven in der Kurvenanzeige dargestellt werden sollen. Der Status der Bits dieser Variablen gibt an, welche Kurve gerade am Bediengerät dargestellt wird.

   Nur für Kurventyp "Echtzeit bitgetriggert":
   - Geben Sie unter "Kurventransfer" eine Variable an, die mindestens so viele Bits enthält, wie Kurven in der Kurvenanzeige dargestellt werden sollen. Der Status der Bits dieser Variablen gibt an, für welche Kurve Werte aus der Steuerung gelesen werden.

   Nur für Kurventyp "Puffer bitgetriggert":
   - Geben Sie eine Puffervariable für den Wechselpuffer an. Die Variable muss vom gleichen Typ und von der gleichen Länge sein wie die Kurvenvariable.
   - Geben Sie unter "Kurventransfer 1" und "Kurventransfer 2" jeweils eine Variable an, die mindestens so viele Bits enthält, wie Kurven in der Kurvenanzeige dargestellt werden sollen. Die Status der Bits dieser Variablen geben an, für welche Kurve Werte der Steuerung gelesen werden. "Kurventransfer 1" steuert das Lesen aus dem Kurvenpuffer, "Kurventransfer 2" steuert das Lesen aus dem Wechselpuffer.
8. Nehmen Sie weitere Einstellungen in den Dialogen des Inspektorfensters vor. Aktivieren Sie z. B. in der Kategorie "Tabelle" die Option "Tabelle anzeigen", um unter der Kurvenanzeige eine Wertetabelle anzuzeigen.

Hinweis

Wenn Sie die Taste <STRG> gedrückt halten und einen Doppelklick auf die Kurvenanzeige machen, wird die Kurvenanzeige aktiv geschaltet. Im aktiven Modus können Sie die Spaltenbreite und die Position der Spalten in der Wertetabelle einstellen. Zur Aktivierung der Kurvenanzeige muss der Zoomfaktor auf 100 % eingestellt sein.

Ergebnis

In Runtime werden in der projektierten Kurvenanzeige die Werte der gewählten Variablen angezeigt.

2.10.2.4 Kurvenanzeige für ein Variablenarchiv projektieren

Einleitung

Mithilfe einer Kurvenanzeige können Sie in Runtime archivierte Variablenwerte anzeigen. Die Kurvenanzeige gibt die archivierten Variablenwerte eines bestimmten Zeitfensters wieder. Um die gewünschte Information aus dem Archiv zu erhalten, verwendet der Bediener die entsprechenden Bedienelemente im Bild oder in der Kurvenanzeige.

Anwendungsbeispiel

Ein Bediener will sich am Ende einer Schicht über den abgelaufenen Prozess informieren.

Voraussetzung

- Ein Variablenarchiv ist erstellt.
- Ein Bild ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Kurvenanzeige ist geöffnet.

Vorgehen

Um eine Kurvenanzeige für die Darstellung von Werten aus einem Variablenarchiv zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus dem Werkzeugfenster aus der Gruppe "Controls" das Objekt "Kurvenanzeige" in das Bild ein.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Kurve" und doppelklicken Sie im Feld "Name" auf "Hinzufügen". Eine neue Kurve wird angelegt.
3. Geben Sie bei Bedarf im Feld "Name" einen eigenen Namen für die Kurve ein.
4. Öffnen Sie in der Spalte "Stil" über die Auswahlschaltfläche den Dialog "Stil" und wählen Sie die Darstellung der Linie aus.
5. Wählen Sie in der Spalte "Typ Kurve" den Kurventyp "Variablenarchiv" aus.
6. Öffnen Sie in der Spalte "Einstellungen" über die Auswahlschaltfläche den Dialog "Datenquelle" und wählen Sie das Variablenarchiv und die Variable für die Werteversorgung der Kurve aus.
7. Nehmen Sie weitere Einstellungen in den Dialogen des Inspektorfensters vor. Aktivieren Sie z. B. in der Kategorie "Tabelle" die Option "Tabelle anzeigen", um unter der Kurvenanzeige eine Wertetabelle anzuzeigen.

Hinweis

Wenn Sie die Taste <STRG> gedrückt halten und einen Doppelklick auf die Kurvenanzeige machen, wird die Kurvenanzeige aktiv geschaltet. Im aktiven Modus können Sie die Spaltenbreite und die Position der Spalten in der Wertetabelle einstellen. Zur Aktivierung der Kurvenanzeige muss der Zoomfaktor auf 100 % eingestellt sein.

Ergebnis

In Runtime werden in der projektierten Kurvenanzeige die archivierten Werte der gewählten Variablen angezeigt.

Siehe auch

Darstellen von Bahnkurven in der Kurvenanzeige (Seite 715)

2.10.2.5 Aufbau einer *.CSV-Datei mit Variablen

Einleitung

Im Dateiformat "*.csv" (Comma Separated Value) werden die Tabellenspalten "Name" und "Wert" des Eintrags durch Semikola getrennt. Jede Tabellenzeile wird durch einen Zeilenumbruch abgeschlossen.

Beispiel einer "*.CSV"-Datei

Das Beispiel zeigt eine Datei mit archivierten Variablenwerten:

"VarName";"TimeString";"VarValue";"Validity";"Time_ms"

"Var_107";"01.04.98 11:02:52";66,00;1;35886460322,81

"Var_108";"01.04.98 11:02:55";60,00;1;35886460358,73

"Var_109";"01.04.98 11:02:57";59,00;1;35886460381,22

Aufbau einer Archivdatei im Format "*.csv"

In den einzelnen Spalten einer Archivdatei sind folgende Werte eingetragen:

Parameter	Beschreibung
VarName	Name der Variablen aus WinCC
TimeString	Zeitstempel als STRING, d.h. als lesbares Datenformat
VarValue	Wert der Variablen
Validity	Gültigkeit: 1 = Wert ist gültig 0 = Fehler ist aufgetreten (z. B. unterbrochene Verbindung)
Time_ms	Zeitstempel als Dezimalwert angegeben (Umrechnung siehe unten). Wird nur benötigt zum Anzeigen der Variablenwerte als Kurve.

Umrechnung des Dezimalwerts des Zeitstempels

Falls Sie den Wert mit einem anderen Programm weiterverarbeiten wollen, gehen Sie folgendermaßen vor:

1. Dividieren Sie Time_ms durch 1.000.000.
   Beispiel: : 36343476928:1 000 000 = 36343,476928
2. Der ganzzahlige Anteil (36344) ist das Datum ab dem 31.12.1899 berechnet.
   Beispiel: 36343 ergibt den 02.07.1999
   In Excel können Sie jetzt den Zeitstempel in Tage umrechnen, indem Sie der Zelle, die den Zeitstempel enthält, ein entsprechendes Format aus der Gruppe "Datum" zuweisen.
   Ergebnis: 37986 ergibt den 31.12.2003
3. Der Wert nach dem Komma (0,476928) ergibt die Uhrzeit:
   - Multiplizieren des Werts (0,476928) mit 24 ergibt die Stunden (11,446272).
   - Multiplizieren des Rests (0,446272) mit 60 ergibt die Minuten (26,77632)
   - Multiplizieren des Rests (0,77632) mit 60 ergibt die Sekunden (46,5792)

   Ergebnis 11:26:46,579
   Diese Umrechnung wird z. B. von Microsoft Excel unterstützt.

Siehe auch

Darstellen von Bahnkurven in der Kurvenanzeige (Seite 715)

2.10.2.6 Direktzugriff auf die ODBC-Archivdatenbank

Einleitung

Der Ablageort eines Archivs kann eine Datenbank oder eine Datei sein.

Die Datenbank wird über ihren "Data Source Name" (DSN) angesprochen. Im Windows-Startmenü unter "Einstellungen > Systemsteuerung > ODBC Data Sources" wählen Sie, welche Datenbank Sie in WinCC verwenden.

Zur Ablage von Archivdaten geben Sie beim Projektieren anstatt eines Verzeichnisnamens den DSN an. Mit dem DSN referenzieren Sie die Datenbank und den Ablageort.

Anwendung

Für die Weiterbearbeitung und Auswertung der Archivdaten steht Ihnen dann der gesamte Funktionsumfang der Datenbank zur Verfügung.

Prinzip

Die Data Source stellt die Verbindung zur Datenbank her. Sie legen die Data Source auf dem gleichen Rechner an, auf dem die Runtime-Software abgelegt wird. Den dort konfigurierten DSN geben Sie dann in WinCC beim Anlegen eines Archivs an.

Über die ODBC-Schnittstelle greifen Sie mit anderen Programmen direkt auf die Datenbank zu, z. B. mit MS Access oder mit MS SQL-Server.

Zusätzlich ist es möglich, den Aufruf eines Programms auf dem Bediengerät mit der Systemfunktion "StarteProgramm" zu projektieren, z. B. für den Aufruf von MS Access. Dabei wird die Runtime nicht unterbrochen.

2.10.2.7 Darstellen von Bahnkurven in der Kurvenanzeige

Einleitung

Um z. B. zweidimensionale Verfahrwege einer Maschine vor Ausführung zu visualisieren, können Sie die f(x)-Kurvenanzeige verwenden.

Umsetzung

Die Grundlage für die Umsetzung ist ein Variablenarchiv, in dem die Koordinaten mit einem identischen Zeitstempel gespeichert sind.

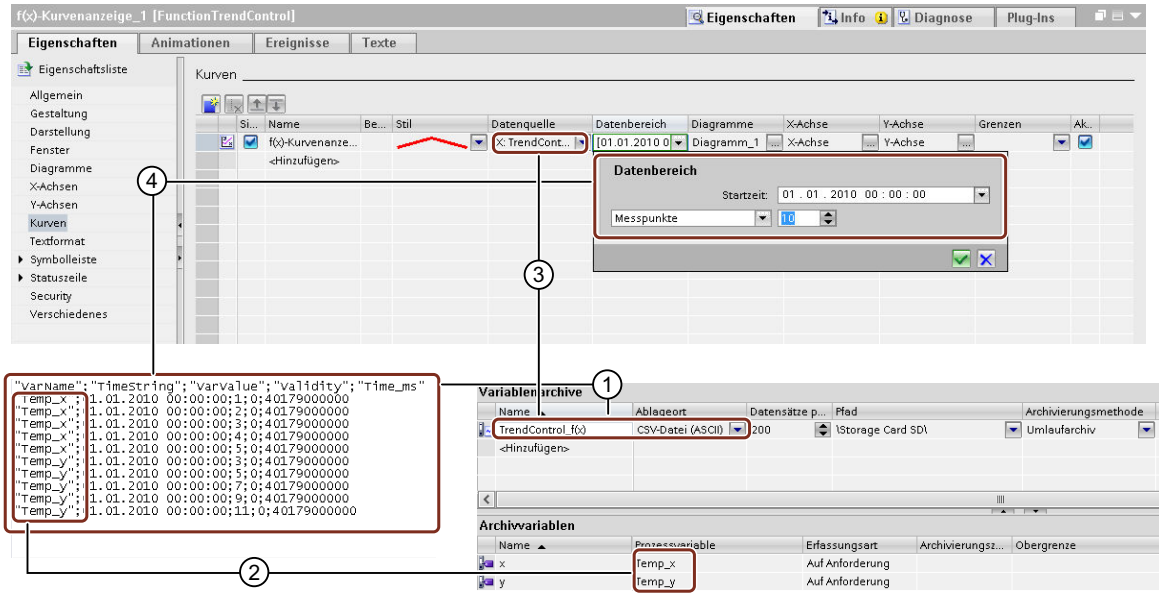

Die Grundlage für die Darstellung von Wertepaaren in der f(x)-Kurvenanzeige ist ein Variablenarchiv, das die berechneten Wertepaare aus einer CSV-Datei liest. Die Zeitstempel jedes Wertepaares sind identisch, z. B. "01.01.2010 00:00:00".

Die Datensätze in der CSV-Datei enhalten die Archivvariablen mit den berechneten Werten. Die Werte können Sie z. B. in der PLC berechnen und in zwei Datenbausteinen speichern. Mit einem VB-Skript können Sie dann z. B. die CSV-Datei erzeugen.

Die Datenquelle für die Kurve ist das Variablenarchiv.

Für die Konfiguration des Datenbereichs sind folgende Einstellungen relevant:

- Der Zeitbereich muss exakt den Zeitstempeln aus der CSV-Datei entsprechen
- Die Anzahl der "Messpunkte" muss exakt der Anzahl der Datensätze in der CSV-Datei entsprechen.

VB-Skript zum Erzeugen einer CSV-Datei

Das folgende VB-Skript erzeugt aus den Werten der HMI-Variablen "Input_X" und "Input_Y". Die HMI-Variablen sind vom Datentyp "Array (0..10) of Real".

```
'##########################################################################
' Location of date log and name of data log which are used in f(x) control
Dim ObjFileName, ObjFilePath

' Name of tag log
ObjFileName = "TrendControl_f(x)0.csv"

' File location for tag log
ObjFilePath = "\Storage Card SD\"

'##########################################################################
' create objects to edit files and use filesystem
Dim ObjFile, ObjFileSystem
Set ObjFile = CreateObject("FileCtl.File")
Set ObjFileSystem = CreateObject("FileCtl.Filesystem")

'##########################################################################
' Write header into the new data log
ObjFile.LinePrint(Chr(34)&"VarName"&Chr(34)&";"&Chr(34)&"TimeString"&Chr(34)&";"&Chr(34)&"
VarValue"&Chr(34)&";"&Chr(34)&"Validity"&Chr(34)&";"&Chr(34)&"Time_ms"&Chr(34))
If Err.Number <> 0 Then
    ShowSystemAlarm "Error: Can not write to data log. " & Err.Number & " " & Err.Description
     Err.Clear
     OpenAllLogs
     TrendControl = 3
     Exit Function
End If
'##########################################################################
' Write entries into data log
' Adapt loop counter to length of archive and array of data tag
Dim ObjLoopCount
ObjLoopCount = 10 ' last element of array

Dim ObjIndex, ObjTime, ObjTimeStamp

For ObjIndex = 0 To ObjLoopCount
     ' Calculate time stamp
     ' 40179000000 = 01.01.2010 00:00:00
     ObjTime = ObjIndex/(60*60*24)*1000
     ObjTimeStamp = 40179000000 + ObjTime
     ' Write x element to log
     Dim Objx
     Objx = SmartTags("Input_X")(ObjIndex)
     ObjFile.LinePrint(Chr(34)&"Temp_x"&Chr(34)&";"&Chr(34)&"01.01.2010
00:00:00"&Chr(34)&";"&Objx&";1;"&ObjTimeStamp)
    ' Write y element to log
     Dim Objy
     Objy = SmartTags("Input_Y")(ObjIndex)
     ObjFile.LinePrint(Chr(34)&"Temp_y"&Chr(34)&";"&Chr(34)&"01.01.2010
00:00:00"&Chr(34)&";"&Objy&";1;"&ObjTimeStamp)
Next
' Write line counter to log
Dim ObjLineCount
ObjLineCount = 41
ObjFile.LinePrint(Chr(34)&"$RT_COUNT$"&Chr(34)&";"&ObjLineCount&";;;;")
```

```
'##########################################################################
' Close new log
ObjFile.close
Set ObjFile = Nothing
Set ObjFileSystem = Nothing
```

Siehe auch

Aufbau einer *.CSV-Datei mit Variablen (Seite 713)

Kurvenanzeige für ein Variablenarchiv projektieren (Seite 712)

2.10.3 Variablen mit Runtime Professional darstellen

2.10.3.1 Grundlagen

Ausgeben von Variablenwerten

Überblick

In Runtime können Sie Variablenwerte als Tabelle oder als Kurve ausgeben. Als Quelle für die Variablenwerte verwenden Sie entweder Prozesswerte oder archivierte Werte.

Um Variablenwerte zu vergleichen, verwenden Sie z. B. eine Tabelle. In der Tabelle können Sie z. B. Füllstände von Vorratstanks vergleichen.

Um Variablenwerte grafisch darzustellen, verwenden Sie z. B. eine Kurve. Mit der Kurve können Sie z. B. den Verlauf einer Motortemperatur darstellen.

Controls zur Darstellung von Variablenwerten

Um Variablenwerte in einer Tabelle darzustellen, verwenden Sie die "Tabellenanzeige".

Um Variablenwerte als Kurve darzustellen, verwenden Sie die Kurvenanzeige. Die Kurvenanzeige steht in zwei Varianten zur Verfügung:

- Mit der "f(t)-Kurvenanzeige" stellen Sie einen Variablenwert in Abhängigkeit von der Zeit dar, z. B. den Temperaturverlauf.
- Mit der "f(x)-Kurvenanzeige" stellen Sie einen Variablenwert in Abhängigkeit von einem zweiten Variablenwert dar, z. B. die Motordrehzahl in Abhängigkeit von der abgegebenen Wärme.

Zusätzlich steht Ihnen mit der "Wertetabelle" ein Control zur Verfügung, mit dem Sie z. B. Statistiken aus den dargestellten Werten erstellen können. Darüber hinaus können Sie die "Wertetabelle" auch als Ablesehilfe für die Kurvenanzeige verwenden.

Projektierung

Im Wesentlichen projektieren Sie bei der Tabellen- und Kurvenanzeige folgende Eigenschaften:

- Datenanbindung
- Darstellung
- Auswertung von Daten
- Projektierung in Runtime
- Persistenz von geänderten Eigenschaften in Runtime

Siehe auch

Tabellenanzeige (Seite 719)

Eigenschaften der Kurvenanzeige (Seite 720)

f(t)-Kurvenanzeige (Seite 721)

f(x)-Kurvenanzeige (Seite 721)

Tabellenanzeige

Überblick

Mit der Tabellenanzeige zeigen Sie in Runtime aktuell anstehende Prozesswerte oder archivierte Werte in einer Tabelle an. Die Darstellung der Tabelle gestalten Sie nach ihren Wünschen.

In Runtime erstellen Sie aus selektierten Daten Statistiken. Darüber hinaus exportieren Sie die Daten zur Weiterverwendung.

Das folgende Bild zeigt eine Tabellenanzeige, in der die Füllstände von drei Vorratstanks angezeigt werden:

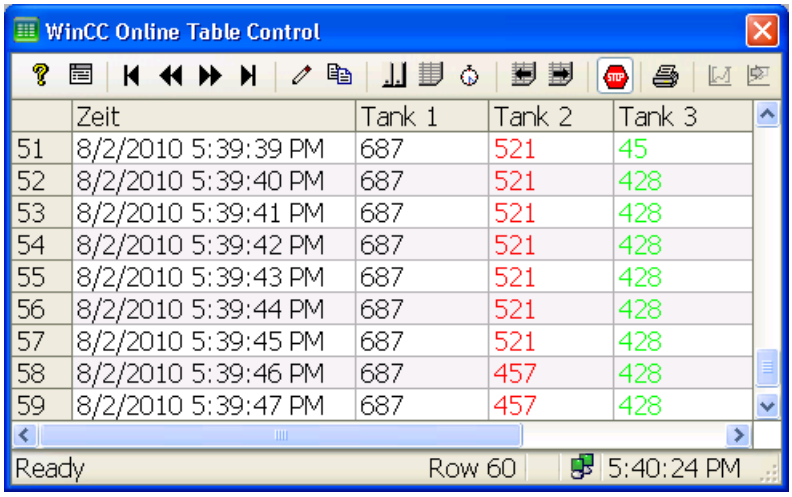

Siehe auch

Ausgeben von Variablenwerten (Seite 718)

Eigenschaften der Kurvenanzeige

Anzahl Kurven

Sowohl in der "f(x)-Kurvenanzeige" als auch der "f(t)-Kurvenanzeige" stellen Sie beliebig viele Kurven dar. Projektieren Sie aus Gründen der Übersichtlichkeit nicht mehr als 8 Kurven gleichzeitig.

Auflösung einer Kurvendarstellung

Die Anzahl der auf dem Bildschirm darstellbaren Werte einer Kurve ist durch die Bildschirmauflösung und die projektierte Größe der Kurvenanzeige begrenzt. Deshalb können in der Kurvenanzeige weniger Werte dargestellt werden als im darzustellenden Zeitraum tatsächlich existieren.

Wenn zum Beispiel in einem Bereich mit 100 Bildpunkten 200 Messwerte archiviert werden, repräsentiert jeder Bildpunkt ein Wertepaar mit 2 Messwerten. Auf dem Bildschirm wird für den darzustellenden Wert der Messwert mit dem jüngsten Zeitstempel verwendet.

Das folgende Bild zeigt einen Kurvenverlauf, wo jeder Bildpunkt ein Wertepaar mit 2 Messwerten repräsentiert:

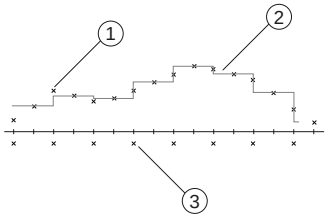

① Reale Messwerte
② Ermittelte Kurve
③ Bildpunkte

Siehe auch

f(t)-Kurvenanzeige (Seite 721)

f(x)-Kurvenanzeige (Seite 721)

Ausgeben von Variablenwerten (Seite 718)

f(t)-Kurvenanzeige

Überblick

Mit der f(t)-Kurvenanzeige stellen Sie in Runtime aktuell anstehende Prozesswerte oder archivierte Werte als Kurve in Abhängigkeit von der Zeit an. Die Kurvendarstellung gestalten Sie nach ihren Wünschen.

Das folgende Bild zeigt eine f(t)-Kurvenanzeige, in der zwei Kurven gleichzeitig dargestellt werden:

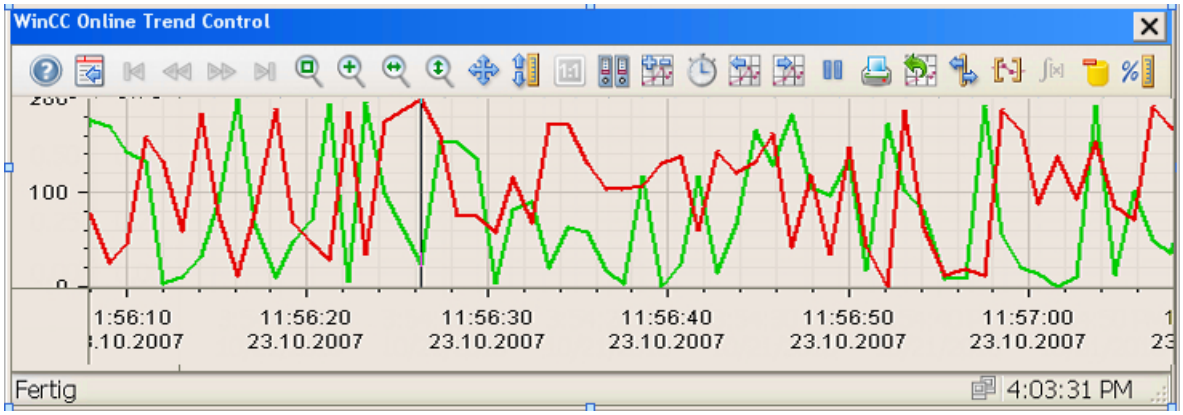

Siehe auch

Eigenschaften der Kurvenanzeige (Seite 720)

Ausgeben von Variablenwerten (Seite 718)

f(x)-Kurvenanzeige

Überblick

Mit der f(x)-Kurvenanzeige stellen Sie in Runtime aktuell anstehende Prozesswerte oder archivierte Werte als Kurve in Abhängigkeit einer anderen Variablen dar. Die Kurvendarstellung gestalten Sie nach ihren Wünschen.

Das folgende Bild zeigt eine f(x)-Kurvenanzeige, in der zwei Kurven gleichzeitig dargestellt werden:

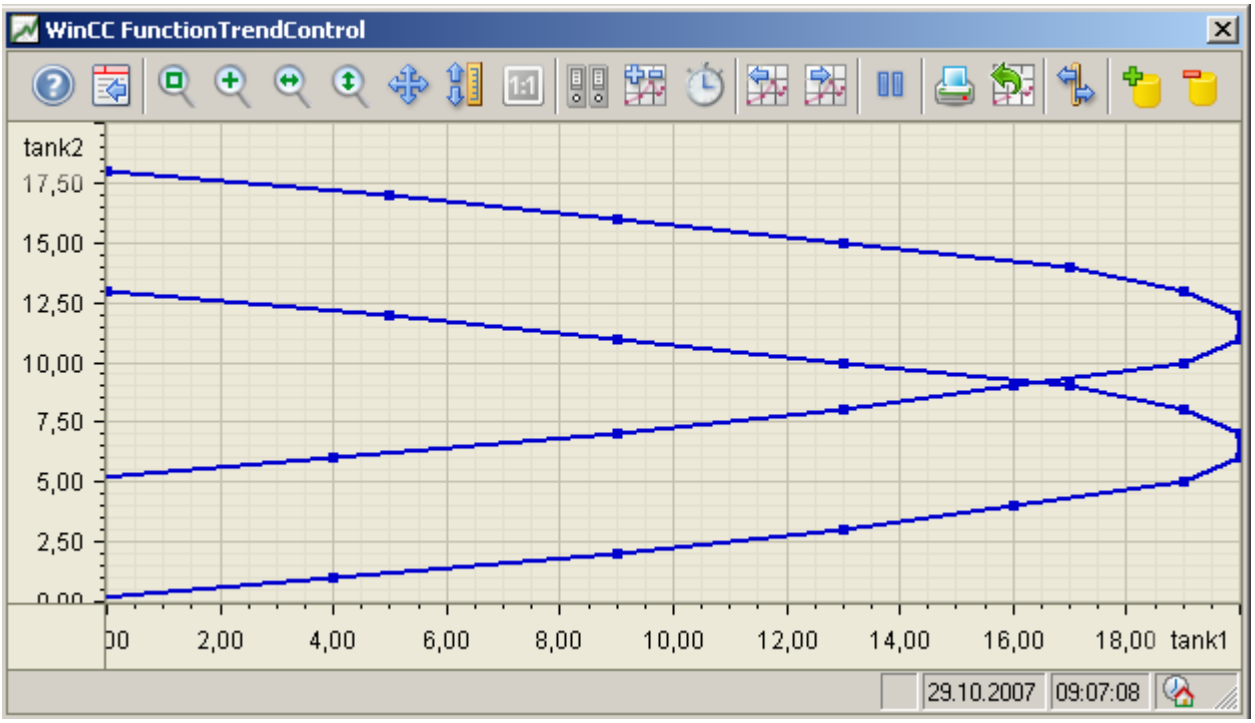

Regeln zur Datenversorgung

Für die Darstellung von Kurven in der f(x)-Kurvenanzeige gelten folgende Regeln:

- In jeder Kurve werden maximal 10000 Wertepaare dargestellt
- Folgende Quellen sind zur Datenversorgung einer Kurve zulässig:
  - Onlinevariablen
  - Archivvariablen
  - Rezepturelemente
  - Benutzerdefiniert: Datenversorgung der Wertepaare über ein Skript
- In der f(x)-Kurvenanzeige werden Wertepaare dargestellt. Damit die Wertepaare gleichzeitig dargestellt werden, sollten die Onlinevariablen oder Archivvariablen einer Kurve in Runtime zur gleichen Zeit aktualisiert werden. Verwenden Sie z. B. für jede Variable den gleichen Aktualisierungszyklus.
- Die Archivvariablen einer Kurve müssen den gleichen Aktualisierungszyklus haben und zyklisch-kontinuierlich erfasst sein

Siehe auch

Eigenschaften der Kurvenanzeige (Seite 720)

Ausgeben von Variablenwerten (Seite 718)

Darstellung von Kurvenverläufen

Einleitung

In einer Kurvenanzeige stellen Sie einen Kurvenverlauf mit einer der folgenden Darstellungsformen dar:

- Punkte
- Interpoliert
- Abgestuft

Die Kurvendarstellung projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven > Stil".

Punkte

Die Werte werden als Punkte dargestellt. Die Darstellung der Punkte ist projektierbar.

Das folgende Bild zeigt den Kurvenverlauf mit der Darstellungsform "Punkte":

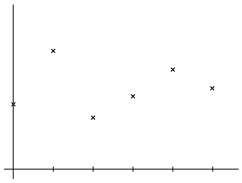

Interpoliert

Der Verlauf der Kurve wird aus den Werten linear interpoliert. Die Darstellung der Linien und der Punkte ist projektierbar.

Das folgende Bild zeigt den Kurvenverlauf mit der Darstellungsform "Interpoliert":

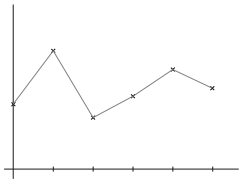

Abgestuft

Der Verlauf der Kurve wird aus den Werten als Treppenkurve ermittelt. Die Darstellung der Linien und der Punkte ist projektierbar.

Das folgende Bild zeigt den Kurvenverlauf mit der Darstellungsform "Abgestuft":

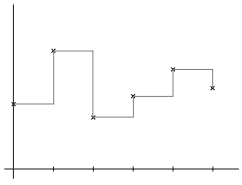

Kurvenverlauf

Über den "Kurvenverlauf" legen Sie fest, wo die aktuellen Werte für alle Kurvenfenster eingetragen werden. Standardmäßig werden die aktuellen Werte von rechts in das Kurvenfenster geschrieben.

Den "Kurvenverlauf" projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Anzeige".

Siehe auch

Kurven für das Kurvenfenster erstellen (Seite 732)

Grundlagen zum Zeitbereich

Zeitbereich

Der Zeitbereich ist der Bereich, aus dem Werte am Bediengerät angezeigt werden. Der Zeitbereich wird bestimmt durch den Startzeitpunkt und den Endzeitpunkt. Der Zeitbereich liegt immer in der Vergangenheit. Wenn der Endzeitpunkt jenseits der aktuellen Systemzeit liegt, wird die aktuelle Systemzeit als temporärer Endzeitpunkt verwendet.

Unterschieden wird zwischen folgenden Zeitbereichen:

- Statischer Zeitbereich
- Dynamischer Zeitbereich

Statischer Zeitbereich

Der statische Zeitbereich wird durch einen festen Start- und Endzeitpunkt bestimmt. Innerhalb dieses Zeitbereichs werden die Werte angezeigt.

Dynamischer Zeitbereich

Der dynamische Zeitbereich wird ausgehend von einem festen Startzeitpunkt durch eine Zeitspanne bestimmt. Der Endzeitpunkt entspricht damit dem Ablauf der Zeitspanne.

Die Zeitspanne legen Sie wie folgt fest:

- Zeitdauer, z. B. 30 Minuten
- Anzahl Messpunkte; multipliziert mit dem Aktualisierungszyklus ergibt sich ebenfalls eine Zeitdauer.

Zeitbereich projektieren

Den Zeitbereich projektieren Sie für alle Controls. Für die Tabellenanzeige und die f(t)-Kurvenanzeige projektieren Sie den Zeitbereich in der Zeitspalte oder in der Zeitachse. Für die f(x)-Tabellenanzeige projektieren Sie den Zeitbereich direkt an der Kurve.

Siehe auch

Zeitachsen der Kurvenanzeige projektieren (Seite 729)

Archivierte Werte darstellen (Seite 781)

Darstellung mit gemeinsamen Achsen

Einleitung

Wenn Sie in der Kurvenanzeige mehrere Kurven gleichzeitig darstellen, ordnen Sie jeder Kurve eigene Wert- und Zeitachsen zu. Alternativ können Sie für mehrere Kurven eine gemeinsame Zeitachse und / oder Werteachse zuordnen.

Die Achsen einer Kurvenanzeige projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zeitachse / Wertachse".

Die Achsen ordnen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" den projektierten Kurven zu.

Darstellung mit unterschiedlichen Achsen

Wenn die in einer Kurvenanzeige darzustellenden Werte sehr unterschiedlich sind, ist eine gemeinsame Wertachse nicht sinnvoll. Wenn Sie jeder Kurve eine eigene Wertachse zuordnen, stellen Sie auch unterschiedliche Skalierungen übersichtlich dar. Einzelne Achsen können Sie bei Bedarf ausblenden.

Das folgende Bild zeigt am Beispiel der f(t)-Kurvenanzeige zwei Kurven mit unterschiedlichen Wertachsen:

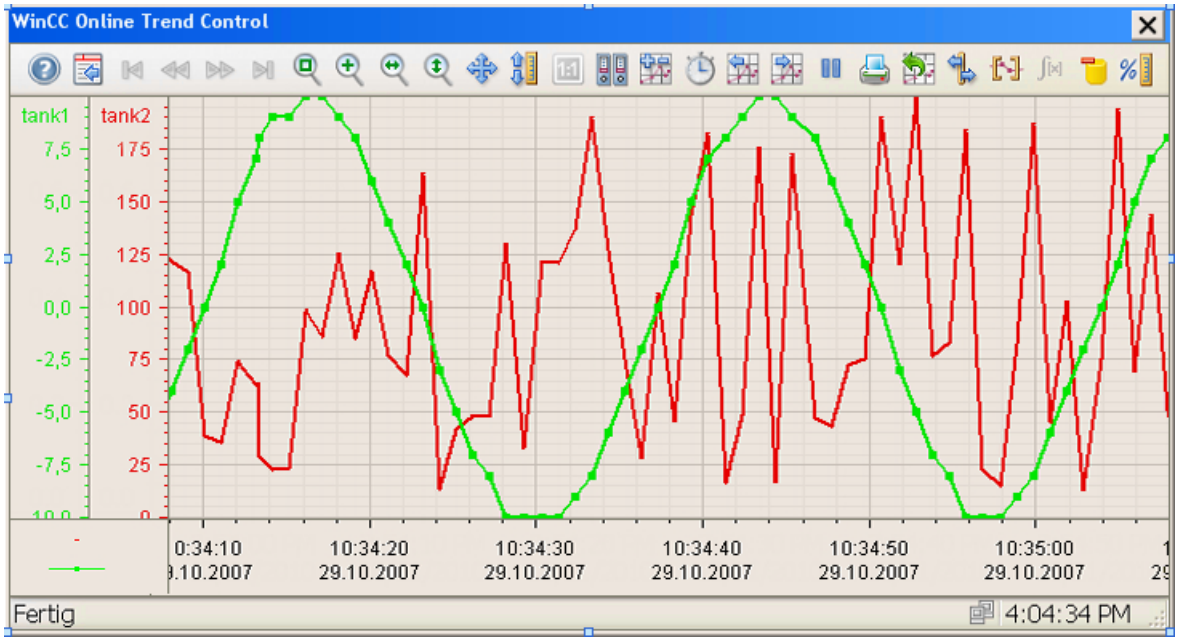

Darstellung mit gemeinsamen Achsen

Wenn die Vergleichbarkeit der Kurvenverläufe wichtig ist, sind gemeinsame Achsen in einer Kurvenanzeige sinnvoll. Verbundene Kurvenanzeigen können eine gemeinsame Zeitachse haben.

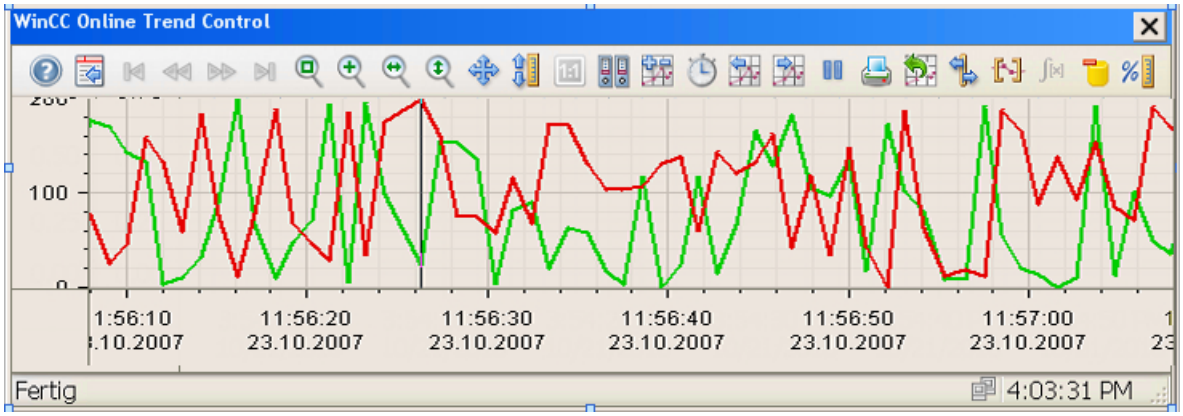

Wenn Sie Kurven mit einer gemeinsamen Zeitachse projektieren, verwenden Sie zur Datenversorgung Variablen mit gleichem Aktualisierungszyklus.

Bei unterschiedlichen Aktualisierungszyklen ist die Länge der Zeitachse nicht für alle Variablen identisch. Da die Kurven wegen unterschiedlicher Aktualisierungszyklen zu verschiedenen Zeiten aktualisiert werden, entsteht bei jedem Wechsel ein geringfügig unterschiedlicher Endzeitpunkt für die Zeitachse. Dadurch springen die dargestellten Kurven bei jedem Wechsel leicht hin und her.

Schaltflächen der Symbolleiste dynamisieren

Einleitung

Wenn Sie das Control nicht über die Symbolleiste bedienen möchten, projektieren Sie an ein Ereignis eines beliebigen Bedienelements z. B. ein Skript. Jede Schaltfläche eines Controls besitzt eine eindeutige ID.

Wenn Sie eine Schaltfläche eines Controls über ein Skript auslösen möchten, weisen Sie der Eigenschaft "ToolbarButtonClick" die Idee der Schaltfläche zu.

Wenn Sie auswerten wollen, welche Schaltfläche eines Controls gedrückt wurde, projektieren Sie an das Ereignis "Schaltfläche Symbolleiste klicken" ein VB-Skript oder ein C-Skript. Die ID der Schaltfläche wird als Parameter "toolbarbuttonID" an das Skript übergeben.

Beispiel: Konfigurationsdialog des Controls mit VBS öffnen

Um den Konfigurationsdialog des Controls "Control_1" mit VBS zu öffnen, haben Sie folgende Möglichkeiten:

- Aufruf im Skript an einem Ereignis eines Bedienelements, z. B. einer Schaltfläche:
  ```
  ScreenItems("Control_1").ToolbarButtonClick = 2
  ```
- Alternativ zu der Eigenschaft "ToolbarButtonClick" stehen in VBS auch Methoden zur Bedienung der Symbolleiste zur Verfügung:
  ```
  Dim obj
  Set obj = ScreenItems("Control_1")
  obj.ShowPropertyDialog
  ```

Beispiel: Konfigurationsdialog des Controls mit C-Skript öffnen

Um den Konfigurationsdialog des Controls "Control_1" in einem C-Skript zu öffnen, haben Sie folgende Möglichkeiten:

- ```
  SetPropertyByConstant ("Screen_1", "Control_1",
  "ToolbarButtonClick", "2");
  ```

Siehe auch

Bedienung der Kurven- und Tabellenanzeige in Runtime (Seite 761)
  ```

2.10.3.2 Variablenwerte in Kurvenform ausgeben

Kurvenanzeige projektieren

Einleitung

Um Variablenwerte in Runtime grafisch darzustellen, fügen Sie eine Kurvenanzeige in ein Bild ein. In der Taskcard "Werkzeuge" sind unter "Controls" zwei Kurvenanzeigen enthalten:

- f(x)-Kurvenanzeige
- f(t)-Kurvenanzeige

Die Achsenbezeichnungen sind bei den beiden Kurvenanzeigen unterschiedlich:

- Die f(x)-Kurvenanzeige hat eine "X-Achse" und eine "Y-Achse"
- Die f(t)-Kurvenanzeige hat eine "Zeitachse" und eine "Wertachse"

Voraussetzung

- Bild ist geöffnet
- Inspektorfenster ist geöffnet

Vorgehen

Um eine Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Taskcard "Werkzeuge" die gewünschte Kurvenanzeige in das Bild ein.
2. Projektieren Sie im Inspektorfenster das Aussehen der Kurvenanzeige:
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Allgemein" die Grundeigenschaften der Kurvenanzeige, z. B. die "Zeitbasis" oder den "Kurvenverlauf".
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung" das Aussehen der Kurvenanzeige.
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Fenster" die Eigenschaften der Kurvenanzeige in Runtime.
3. Projektieren Sie bei Bedarf zusätzliche Diagramme.
4. Projektieren Sie im Inspektorfenster die Achsen der Kurvenanzeige.
   - Konfigurieren Sie die Eigenschaften der Achsen.
   - Ordnen Sie jede Achse einem Diagramm der Kurvenanzeige zu.
5. Fügen Sie im Inspektorfenster die Kurven ein, die in der Kurvenanzeige angezeigt werden:
   - Legen Sie für jede Kurve die "Datenquelle" fest.
   - Ordnen Sie jeder Kurve das "Diagramm" zu, in dem sie dargestellt wird.
   - Ordnen Sie jeder Kurve die Achsen zu.
   - Konfigurieren Sie das Aussehen jeder Kurve.

6. Konfigurieren Sie die Symbolleiste und Statuszeile der Kurvenanzeige.
7. Konfigurieren Sie bei Bedarf den Datenexport aus der Kurvenanzeige.
8. Konfigurieren Sie bei Bedarf die Sicherheitseinstellungen der Kurvenanzeige.

Ergebnis

Die Kurvenanzeige ist projektiert. Wenn Sie die Daten der Kurvenanzeige in Runtime auswerten wollen, projektieren Sie zusätzlich eine Werteanzeige. Die Werteanzeige können Sie auch als "Lineal" konfigurieren.

Siehe auch

Zeitachsen der Kurvenanzeige projektieren (Seite 729)

Wertachse der Kurvenanzeige projektieren (Seite 730)

Diagramme in Kurvenanzeige erstellen (Seite 731)

Kurven für das Kurvenfenster erstellen (Seite 732)

Datenversorgung festlegen (Seite 734)

Symbolleiste und Statuszeile konfigurieren (Seite 736)

Export der Runtime-Daten projektieren (Seite 737)

Auswirkung der Online-Projektierung bestimmen (Seite 738)

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Zeitachsen der Kurvenanzeige projektieren

Einleitung

Den Zeitbereich für die Kurvendarstellung projektieren Sie über Zeitachsen. In einer Kurvenanzeige können Sie mehrere Zeitachsen anlegen, die Sie einem oder mehreren Diagrammen zuordnen.

Wenn Sie einer Kurvenanzeige mehrere Zeitachsen zuordnen, entspricht die Reihenfolge der Zeitachse im Inspektorfenster der Reihenfolge in der Kurvenanzeige. Wenn mehrere Zeitachsen an der gleichen Seite einer Kurvenanzeige ausgerichtet sind, nimmt die 1. Zeitachse in der Liste die Position unten links ein. Die letzte Zeitachse in der Liste nimmt dann die Position oben rechts ein.

Die Zeitachse ist nur bei der f(t)-Kurvenanzeige verfügbar.

Voraussetzung

- Kurvenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Zeitachse einer Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine neue Zeitachse hinzuzufügen, doppelklicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zeitachse" auf "Hinzufügen".
2. Geben Sie den "Namen" und den "Anzeigenamen" der Zeitachse ein.
3. Konfigurieren Sie den "Achsenstil".
4. Wählen Sie das "Diagramm", zu dem die Zeitachse zugeordnet wird.
5. Konfigurieren Sie den "Zeitbereich" und das "Format" der Zeitdarstellung der Zeitachse.
6. Um die der Zeitachse zugeordneten Kurven in der Kurvenanzeige immer zu aktualisieren, aktivieren Sie die "Online".

Hinweis

Drucken eines festen Zeitbereichs

Wenn Sie einen definierten Zeitbereich drucken wollen, deaktivieren Sie "Online".

Wenn Sie z. B. eine aktuelle Kurvendarstellung mit einer früheren Kurvendarstellung vergleichen wollen, versorgen Sie die Vergleichskurve aus einer Archivvariablen.

Ergebnis

Die Zeitachse ist projektiert. Ordnen Sie die Zeitachse einer oder mehreren Kurven zu.

Siehe auch

Grundlagen zum Zeitbereich (Seite 724)

Aufbau der Zeitformatvorgabe (Seite 739)

Aufbau der Datumsformatvorgabe (Seite 740)

Wertachse der Kurvenanzeige projektieren

Einleitung

In einer Kurvenanzeige können Sie mehrere Wertachsen anlegen, die Sie einem oder mehreren Diagrammen zuordnen.

Standardmäßig sind bei Wertachse folgende Eigenschaften voreingestellt:

- Der Wertbereich richtet sich nach den aktuellen Werten der zugeordneten Kurve
- Die Skalierung der Wertachse richtet sich linear nach dem Wertebereich

Alternativ können Sie auch eine logarithmische Skalierung konfigurieren:

- Bei der logarithmischen Skalierung werden nur positive Werte dargestellt
- Bei der negativ logarithmischen Skalierung nur negative Werte dargestellt

Wenn Sie eine Wertachse für die f(t)-Kurvenanzeige projektieren, können Sie zusätzlich Achsenabschnitte einrichten. Einem Achsenabschnitt weisen Sie einen Wertebereich und einen Anzeigenamen zu.

Bei der f(x)-Kurvenanzeige entspricht die Wertachse sowohl der "X-Achse" als auch der "Y-Achse".

Voraussetzung

- Kurvenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Wertachse einer Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine neue Wertachse hinzuzufügen, doppelklicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Wertachse" auf "Hinzufügen".
2. Geben Sie den "Namen" und den "Anzeigenamen" der Wertachse ein.
3. Konfigurieren Sie den "Achsenstil".
4. Wählen Sie das "Diagramm", zu dem die Wertachse zugeordnet wird.
5. Konfigurieren Sie bei Bedarf den "Wertebereich" der Wertachse.
6. Konfigurieren Sie bei Bedarf das "Format" der Beschriftung der Wertachse.
7. Konfigurieren Sie bei Bedarf die "Skalierung" der Wertachse.
8. Wenn Sie eine f(t)-Kurvenanzeige projektieren, legen Sie bei Bedarf "Achsenabschnitte" fest:
   - Aktivieren Sie "benutzerdefinierte Achsenabschnitte".
   - Um einen Achsenabschnitt hinzuzufügen, doppelklicken Sie auf "Hinzufügen".
   - Geben Sie den "Wertebereich" des Achsenabschnitts ein.
   - Konfigurieren Sie den "Anzeigebereich" des Achsenabschnitts.
   - Geben Sie den "Anzeigenamen" des Achsenabschnitts ein.

Ergebnis

Die Wertachse ist projektiert. Ordnen Sie die Wertachse einer oder mehreren Kurven zu.

Diagramme in Kurvenanzeige erstellen

Einleitung

Eine Kurvenanzeige enthält standardmäßig einen Anzeigebereich, in dem Sie eine oder mehrere Kurven projektieren.

Alternativ können Sie den Anzeigebereich in mehrere "Diagramme" unterteilen. Jedes "Diagramm" funktioniert wie eine eigenständige Kurvenanzeige. Auf diese Weise können Sie z. B. Temperaturverläufe anschaulich darstellen, ohne dass sich die Kurven überlagern. Die Höhe eines Diagramms legen Sie über den "Bereichsanteil" fest. Der "Bereichsanteil" legt fest, wie viel Platz einem Diagramm in der Kurvenanzeige zur Verfügung gestellt wird.

Berechnung der Bereichsanteile

Die Höhe jedes Bereichsanteils errechnet sich aus der Gesamtanzahl der Bereichsteile. Wenn Sie z. B. insgesamt drei Diagramme projektiert haben, erhalten Sie mit einem Bereichsanteil von jeweils "1" drei gleichgroße Diagramme. Um die Bereichsanteile relativ zueinander zu vergrößern, erhöhen Sie den Bereichsanteil eines oder mehrerer Diagramme. Die Änderung der Bereichsanteile wirkt sich sofort in der Kurvenanzeige aus.

Voraussetzung

- Kurvenanzeige ist selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um eine Kurvenanzeige in mehrere Diagramme zu unterteilen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Diagramme" auf "Hinzufügen".
   Das Diagramm wird hinzugefügt.
2. Geben Sie einen sprechenden "Namen" für das Diagramm ein.
3. Wählen Sie den "Bereichsanteil", den das Diagramm in der Kurvenanzeige belegt.
4. Konfigurieren Sie das "Raster" des Diagramms.
5. Konfigurieren Sie das "Lineal" des Diagramms.

Ergebnis

Das Diagramm ist angelegt. Sie können jedem Diagramm eine oder mehrere Kurven zuordnen.

Kurven für das Kurvenfenster erstellen

Einleitung

Mit einer Kurve stellen Sie Werte in der Kurvenanzeige dar. Sie ordnen jeder Kurve eine Zeit- und eine Wertachse zu. Die der Kurve zugeordnete Wertachse bestimmt die Kurvenanzeige, in der die Kurve angezeigt wird.

Standardmäßig richtet sich der Datenbereich nach den aktuellen Werten der zugeordneten Kurve.

Sie können für eine Kurve zusätzlich die Visualisierung von Grenzwerten und Werten mit "unsicherem Status" konfigurieren. Wenn eine Kurve einen projektierten Grenzwert über- oder unterschreitet, wird die Kurve eingefärbt.

Wenn Sie in einer Kurvenanzeige mehrere Kurven projektieren, entspricht die Reihenfolge der Kurven im Inspektorfenster der Reihenfolge in der Kurvenanzeige.

Voraussetzung

- Kurvenanzeige ist im Bild selektiert
- Zeitachsen und Wertachsen sind konfiguriert
- Variablen für die Datenversorgung sind projektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um eine Kurve hinzuzufügen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" auf "Hinzufügen".
2. Geben Sie eine "Beschriftung" der Kurve ein.
3. Konfigurieren Sie bei Bedarf den "Stil" der Kurve.
   Die Änderungen werden in einer Vorschau angezeigt.
4. Konfigurieren Sie die Datenquelle.
5. Konfigurieren Sie bei Bedarf den "Datenbereich" der Kurve.
6. Wählen Sie das "Diagramm", in dem die Kurve dargestellt wird.
7. Ordnen Sie der Kurve je eine "Wertachse" und "Zeitachse" zu.
8. Konfigurieren Sie bei Bedarf die "Grenzen" der Kurve.

Ergebnis

Die Kurve ist projektiert.

Siehe auch

Darstellung von Kurvenverläufen (Seite 723)

Datenversorgung festlegen (Seite 734)

Datenversorgung festlegen

Einleitung

Mit der Datenversorgung legen Sie fest, aus welcher Quelle die Werte in Runtime am Bediengerät angezeigt werden. Folgende Quellen stehen zur Verfügung:

- Aktuelle Prozesswerte aus Variablen
- Archivierte Werte aus Archivvariablen
- Rezepturelemente (nur f(x)-Kurvenanzeige)
- Datenversorgung in Runtime per Skript

Voraussetzung

- Eine Online-Variable oder Archivvariable ist projektiert
- Wertspalte oder Kurve ist angelegt
- Inspektorfenster ist geöffnet

Aktuelle Prozesswerte darstellen

Um aktuelle Prozesswerte darzustellen, gehen Sie folgendermaßen vor:

1. Wenn Sie die Datenquelle für eine Tabellenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Wertspalten".
2. Wenn Sie die Datenquelle für eine Kurvenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kurven".
3. Konfigurieren Sie die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Variablen".
   - Wenn Sie die Kurve einer f(x)-Kurvenanzeige konfigurieren, wählen Sie unter "X-Werte" und "Y-Werte" jeweils eine Variable.
   - Wenn Sie die Kurve einer f(t)-Kurvenanzeige oder eine Wertspalte konfigurieren, wählen Sie eine "Variable".
   - Wählen Sie den "Aktualisierungszyklus.

Werte aus einem Archiv darstellen

Um Werte aus einem Archiv darzustellen, gehen Sie folgendermaßen vor:

1. Wenn Sie die Datenquelle für eine Tabellenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Wertspalten".
2. Wenn Sie die Datenquelle für eine Kurvenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kurven".
3. Konfigurieren Sie die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Archivvariablen".
   - Wenn Sie die Kurve einer f(x)-Kurvenanzeige konfigurieren, wählen Sie unter "X-Werte" und "Y-Werte" jeweils eine Variable.
   - Wenn Sie die Kurve einer f(t)-Kurvenanzeige oder eine Wertspalte konfigurieren, wählen Sie eine "Variable".

Werte aus Rezepturen darstellen

Werte aus Rezepturen geben Sie in einer f(x)-Kurvenanzeige aus.

Um Werte aus Rezepturen darzustellen, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Rezeptur".
   - Wählen Sie unter "X-Werte" und "Y-Werte" jeweils ein Rezepturelement.

Benutzerdefinierte Datenversorgung in Runtime einrichten

Wenn Sie erst in Runtime per Skript die Kurvenwerte mit Daten versorgen wollen, benötigen Sie ein Skript über die API-Schnittstelle.

Um eine Anbindung in Runtime per Skript herzustellen, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Benutzerdefiniert".
   - Binden Sie das Skript über die API-Schnittstelle an die entsprechende Kurve an.

Ergebnis

Die Datenversorgung ist projektiert.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Grundlagen zur Variablenarchivierung (Seite 639)

Kurvenanzeige projektieren (Seite 728)

Kurven für das Kurvenfenster erstellen (Seite 732)

Symbolleiste und Statuszeile konfigurieren

Einleitung

Sie bedienen die Controls in Runtime über die Schaltflächen in der Symbolleiste. Die Statuszeile zeigt Statusmitteilungen des Controls an. Während der Projektierung konfigurieren Sie den Inhalt der Symbolleiste und Statuszeile.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Symbolleiste projektieren

Um die Symbolleiste zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Allgemein" die allgemeinen Eigenschaften der Symbolleiste, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen" die Schaltflächen, die Sie in Runtime benötigen. Informationen zu der Funktion der einzelnen Schaltflächen erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Schaltflächen festzulegen, markieren Sie in der Liste die Schaltfläche. Verschieben Sie dann die Schaltfläche an die gewünschte Position.
4. Legen Sie bei Bedarf für die Schaltflächen der Symbolleiste einen "Hotkey" fest.
5. Wählen Sie bei Bedarf die Berechtigung, die in Runtime zum Bedienen der Schaltfläche notwendig ist.
6. Wenn eine Schaltfläche in Runtime nicht bedient werden soll, deaktivieren Sie "Aktivieren". Sie können eine deaktivierte Schaltfläche z. B. mit einem Skript in Runtime wieder aktivieren.

Statuszeile projektieren

Um die Statuszeile zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Allgemein" die allgemeinen Eigenschaften der Statuszeile, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Elemente" die Elemente, die Sie in Runtime benötigen.
   Informationen zu den Elementen der Statuszeile erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Elemente festzulegen, markieren Sie in der Liste das Element. Verschieben Sie dann das Element an die gewünschte Position.
4. Um die Breite eines Elements in der Statuszeile anzupassen, deaktivieren Sie "Automatisch anpassen". Geben Sie dann einen Pixel-Wert für die Breite an.

Siehe auch

Bedienung der Kurven- und Tabellenanzeige in Runtime (Seite 761)

Kurvenanzeige projektieren (Seite 728)

Elemente der Statuszeile (Seite 782)

Export der Runtime-Daten projektieren

Einleitung

In Runtime können Sie die angezeigten Daten eines Controls im CSV-Format exportieren. Abhängig vom verwendeten Control und der Projektierung exportiert der Bediener alle Daten oder nur die markierten Daten.

Den Dateinamen der CSV-Datei sowie den Speicherort legt der Bediener beim Export fest. Alternativ können Sie den Dateinamen auch automatisch aus folgenden Platzhaltern generieren:

- @OBJECTNAME: Objektname des Controls
- @CURRENTDATE: aktuelles Datum
- @CURRENTTIME: aktuelle Zeit

Voraussetzung

- Control ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um den Export von Runtime-Daten zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Datenexport". Die Datenexport-Einstellungen sind bereits vorkonfiguriert.
2. Geben Sie bei Bedarf einen eigenen Dateinamen und einen Pfad für die Exportdatei ein.
3. Wählen Sie den "Umfang" des Datenexports.
4. Konfigurieren Sie die "Bedienung in Runtime".
5. Damit der Bediener den Datenexport starten kann, aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen > Export Data".

Ergebnis

In Runtime kann der Bediener die Daten des Controls über die Symbolleiste exportieren.

Siehe auch

Kurvenanzeige projektieren (Seite 728)

Auswirkung der Online-Projektierung bestimmen

Einleitung

In WinCC projektieren Sie das Aussehen und die Funktion eines Controls in Runtime. Zusätzlich können Sie festlegen, dass der Bediener in Runtime ein Control konfigurieren kann. Ob Konfigurationsänderungen in Runtime z. B. auch nach einem Bildwechsel erhalten bleiben, legen Sie Control unter "Eigenschaften > Eigenschaften > Security > Persistenz" fest:

- Keine Persistenz
  Die Änderungen an der Konfiguration werden nicht gespeichert. Die Änderungen werden beim nächsten Bildwechsel und beim Deaktivieren des Projekts verworfen.
- Persistenz
  Die Änderungen an der Konfiguration werden gespeichert. Die Änderungen bleiben auch nach dem Deaktivieren / Aktivieren des Projekts erhalten. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.
- Persistenz in Runtime
  Die Änderungen an der Konfiguration werden nur während Runtime gespeichert. Wenn Sie das Projekt deaktivieren, werden die Änderungen verworfen. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.

Hinweis

Wenn Sie ein projektiertes Control ändern und auf das Bediengerät laden, werden die Änderungen in Runtime unabhängig von den Einstellungen überschrieben.

Voraussetzung

- Ein Control projektiert
- Inspektorfenster ist geöffnet

Vorgehensweise

Um die Persistenz von Daten in Runtime zu konfigurieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Inspektorfenster "Eigenschaften > Eigenschaften > Security".
2. Wählen Sie die "Online-Projektierung".
3. Wählen Sie bei Bedarf die Berechtigung aus.
4. Wählen Sie das "Verhalten bei Bildwechsel".

Ergebnis

Das Verhalten bei der Online-Projektierung ist projektiert. In Runtime kann der Bediener abhängig von den Einstellungen die Konfiguration des Controls anpassen.

Wenn mehrere Bediener am Bediengerät arbeiten, werden bei einem Bedienerwechsel die individuellen Projektierungen erst nach einem Bildwechsel übernommen.

Siehe auch

Kurvenanzeige projektieren (Seite 728)

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Aufbau der Zeitformatvorgabe

Zeitformat

Die vordefinierten Zeitformate der Auswahlliste sind überschreibbar.

Wenn Sie eigene Zeitformate festlegen, beachten Sie Folgendes:

- Leerzeichen, die in der Formatvorgabe Werte trennen, werden in der Zeitanzeige an die gleiche Stelle gesetzt.
- Beachten Sie die Groß- und Kleinschreibung.
- Zeichen, die in einfache Anführungszeichen gesetzt sind, werden in der Zeitanzeige unverändert an die gleiche Stelle gesetzt.

Um eine Formatvorgabe aufzubauen, verwenden Sie folgende Werte:

Wert	Beschreibung
h	Stunden, 12-Stunden-Uhr, für einstellige Stunden ohne führende Null
hh	Stunden, 12-Stunden-Uhr, für einstellige Stunden mit führender Null
H	Stunden, 24-Stunden-Uhr, für einstellige Stunden ohne führende Null
HH	Stunden, 24-Stunden-Uhr, für einstellige Stunden mit führender Null

Wert	Beschreibung
m	Minuten, ohne führende Null für einstellige Minuten
mm	Minuten, mit führender Null für einstellige Minuten
s	Sekunden, ohne führende Null für einstellige Sekunden
ss	Sekunden, mit führender Null für einstellige Sekunden
t	Zeitbezeichner AM/PM einstellig: A oder P
tt	Zeitbezeichner AM/PM zweistellig: AM oder PM

Beispiel

Um die Zeitanzeige "11:29:40 PM" aufzubauen, verwenden Sie folgende Formatvorgabe:

"hh:mm:ss tt"

Aufbau der Datumsformatvorgabe

Zeitformat

Die vordefinierten Datumsformate der Auswahlliste sind überschreibbar.

Wenn Sie eigene Datumsformate festlegen, beachten Sie Folgendes:

- Leerzeichen, die in der Formatvorgabe Werte trennen, werden in der Datumsanzeige an die gleiche Stelle gesetzt.
- Beachten Sie die Groß- und Kleinschreibung.
- Zeichen, die in einfache Anführungszeichen gesetzt sind, werden in der Datumsanzeige unverändert an die gleiche Stelle gesetzt.

Um eine Formatvorgabe aufzubauen, verwenden Sie folgende Werte:

Wert	Beschreibung
D	Tag des Monats als Zahl, für einstellige Tage ohne führende Null
Dd	Tag des Monats als Zahl, für einstellige Tage mit führender Null
Ddd	Wochentag mit drei Buchstaben abgekürzt
Dddd	Wochentag ausgeschrieben
M	Monat als Zahl, für einstellige Monate ohne führende Null
MM	Monat als Zahl, für einstellige Monate mit führender Null
MMM	Monat mit drei Buchstaben abgekürzt
MMMM	Monat ausgeschrieben
y	Jahr, nur die letzten beiden Zahlen, für Jahre unter 10 mit führender Null, gleich dem Format "yy"
yy	Jahr, nur die letzten beiden Zahlen, für Jahre unter 10 mit führender Null
yyyy	Vollständige, vierstellige Jahrsangabe

Beispiel

Um die Datumsanzeige "Mi, Aug 31 13" aufzubauen, verwenden Sie folgende Format-Vorgabe:

"Ddd, MMM dd yy"

2.10.3.3 Variablenwerte in Tabellenform ausgeben

Tabellenanzeige projektieren

Einleitung

Um Variablenwerte in Runtime tabellarisch darzustellen, fügen Sie eine Tabellenanzeige in ein Bild ein. Zu jedem Wert wird auch ein Zeitstempel angezeigt. Die Werte werden in Wertespalten angezeigt, die Zeitstempel in Zeitspalten. Eine Zeitspalte ordnen Sie einer oder mehreren Wertespalten zu.

Vorgehen

Um eine Tabellenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Taskcard "Werkzeuge" die Tabellenanzeige in das Bild ein.
2. Projektieren Sie im Inspektorfenster das Aussehen der Tabellenanzeige:
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Allgemein" die Grundeigenschaften der Tabellenanzeige, z. B. die "Zeitbasis".
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung" das Aussehen der Tabellenanzeige.
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Fenster" die Eigenschaften der Tabellenanzeige in Runtime.
3. Projektieren Sie im Inspektorfenster die Spalten der Tabellenanzeige.
   - Konfigurieren Sie die Eigenschaften der Spalten.
   - Projektieren Sie eine oder mehrere Zeitspalten mit den Zeitbereichen für die Tabelle.
   - Projektieren Sie eine oder mehrere Wertespalten.
   - Legen Sie für jede Wertespalte die Datenanbindung fest.
   - Ordnen Sie den Wertespalten Zeitspalten zu.
4. Konfigurieren Sie das Aussehen der Tabelle:
   - Konfigurieren Sie die "Überschriften", den "Rumpf" und das "Raster".
   - Konfigurieren Sie bei Bedarf die Einstellungen zum "Markieren" von Tabelleneinträgen.
   - Konfigurieren Sie bei Bedarf die Einstellungen zum "Sortieren" von Tabelleneinträgen.
5. Konfigurieren Sie die Symbolleiste und Statuszeile der Tabellenanzeige.
6. Konfigurieren Sie bei Bedarf den Datenexport aus der Tabellenanzeige.
7. Konfigurieren Sie bei Bedarf die Sicherheitseinstellungen der Tabellenanzeige.

Ergebnis

Die Tabellenanzeige ist projektiert. Wenn Sie die Daten der Tabellenanzeige in Runtime auswerten wollen, projektieren Sie zusätzlich eine Wertetabelle. Die Wertetabelle können Sie auch als "Lineal" konfigurieren.

Siehe auch

Zeitspalten der Tabellenanzeige projektieren (Seite 742)

Wertspalten der Tabellenanzeige projektieren (Seite 743)

Aussehen der Tabellenelemente konfigurieren (Seite 744)

Markierung der ausgewählten Zellen und Zeilen projektieren (Seite 745)

Sortierung von Tabellenspalten projektieren (Seite 746)

Schaltflächen der Symbolleiste dynamisieren (Seite 727)

Datenversorgung festlegen (Seite 747)

Symbolleiste und Statuszeile konfigurieren (Seite 749)

Export der Runtime-Daten projektieren (Seite 750)

Auswirkung der Online-Projektierung bestimmen (Seite 751)

Online-Projektierung der Tabellenanzeige (Seite 775)

Zeitspalten der Tabellenanzeige projektieren

Einleitung

Den Zeitbereich für die tabellarische Darstellung projektieren Sie über Zeitspalten. Eine Tabelle kann für mehrere Wertespalten getrennte Zeitspalten oder eine gemeinsame Zeitspalte haben.

Wenn Sie einer Tabellenanzeige mehrere Zeitspalten zuordnen, entspricht die Reihenfolge der Zeitspalten im Inspektorfenster der Reihenfolge in der Tabellenanzeige.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Zeitspalte einer Tabellenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Um Zeitspalten einzufügen, klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zeitspalten" auf "Hinzufügen".
2. Geben Sie den "Namen" und die "Beschriftung" der Zeitspalte ein.
3. Konfigurieren Sie den "Stil" der Zeitspalte.

4. Konfigurieren Sie den "Zeitbereich" und das "Format" der Zeitdarstellung der Zeitspalte.
5. Konfigurieren Sie die Darstellung von "Kopf" und "Zellen".
6. Wenn die Werte der Zeitspalte automatisch aktualisiert werden sollen, aktiveren Sie "Aktualisieren".

Ergebnis

Die Zeitspalte ist projektiert. Ordnen Sie die Zeitspalte einer oder mehreren Wertspalten zu.

Siehe auch

Wertspalten der Tabellenanzeige projektieren (Seite 743)

Aufbau der Zeitformatvorgabe (Seite 752)

Aufbau der Datumsformatvorgabe (Seite 753)

Wertspalten der Tabellenanzeige projektieren

Einleitung

Die Werte der Tabellenanzeige projektieren Sie über Wertespalten. Sie können mehrere Wertespalten in einer Tabelle darstellen, z. B. um die Füllstände von mehreren Behältern miteinander zu vergleichen. Jede Wertespalte ist mit einer Zeitspalte verbunden. Die Wertespalten können eine gemeinsame Zeitspalte haben.

Wenn Sie einer Tabellenanzeige mehrere Wertspalten zuordnen, entspricht die Reihenfolge der Wertspalten im Inspektorfenster der Reihenfolge in der Tabellenanzeige. Wenn Sie mehreren Wertspalten dieselbe Zeitspalte zuordnen, werden die Wertspalten in der Liste automatisch nach zugeordneter Zeitspalte gruppiert.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet
- Zeitspalte ist angelegt

Vorgehen

Um eine Wertspalte zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine neue Wertspalte hinzuzufügen, doppelklicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Wertspalten" auf "Hinzufügen".
2. Geben Sie den "Namen" und die "Beschriftung" der Wertspalte ein.
3. Konfigurieren Sie die Datenquelle.
4. Ordnen Sie der Wertspalte eine Zeitspalte zu.

5. Konfigurieren Sie bei Bedarf die Darstellung von "Kopf" und "Zellen".
6. Konfigurieren Sie bei Bedarf die "Sortierung" der Wertspalte.

Ergebnis

Die Wertspalte ist projektiert.

Siehe auch

Datenversorgung festlegen (Seite 734)
Zeitspalten der Tabellenanzeige projektieren (Seite 742)
Datenversorgung festlegen (Seite 747)

Aussehen der Tabellenelemente konfigurieren

Einleitung

Das Aussehen der Tabellenelemente passen Sie Ihren Bedürfnissen an.

Voraussetzung

- Tabellenanzeige oder Wertetabelle ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Farben der Tabellenelemente zu konfigurieren, gehen Sie folgendermaßen vor:

1. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Überschrift" die folgenden Eigenschaften fest:
   - Sichtbarkeit der Spaltenüberschriften und Zeilenköpfe
   - Ausrichtung der Spaltenüberschriften und Zeilenköpfe
   - Farbgestaltung der Spaltenüberschriften
2. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Rumpf" die folgenden Eigenschaften fest:
   - Farbgestaltung des Tabelleninhalts, z. B. unterschiedliche Farben für Zeilen zur besseren Unterscheidung
   - Sichtbarkeit von leeren Spalten und Zeilen
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Raster" folgende Eigenschaften fest:
   - Gestaltung der Rasterlinien
   - Zeichenabstand in den Tabellenzellen

Ergebnis

Das Aussehen der Tabellenelemente ist projektiert. Zusätzlich können Sie noch weitere Eigenschaften festlegen:

- Das Verhalten und die Farbgestaltung beim Markieren eines Tabelleneintrags
- Die Sortierung von Tabelleneinträgen

Siehe auch

Markierung der ausgewählten Zellen und Zeilen projektieren (Seite 745)

Sortierung von Tabellenspalten projektieren (Seite 746)

Markierung der ausgewählten Zellen und Zeilen projektieren

Einleitung

Wenn ein Bediener z. B. Werte aus einer Zeile exportieren will, muss der Bediener die Zeile markieren. Während der Projektierung legen Sie Folgendes fest:

- Markierungsbereich
  Für den Markierungsbereich konfigurieren Sie folgende Eigenschaften:
  - Die Anzahl der Tabellenelemente, die der Bediener markieren kann
  - Die Gestaltung des Markierungsbereichs
- Farbgestaltung des Markierungsbereichs
  Die Farben wählen Sie entweder individuell aus oder Sie verwenden eine vom System festgelegte Farbe.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um den Markierungsbereich zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Markierung > Markierungsbereich" den "Typ" des Markierungsbereichs.
2. Wählen Sie bei Bedarf den "Rahmen" aus, der um den Markierungsbereich dargestellt wird.
3. Wählen Sie bei Bedarf die Farben für die "Zellen", "Zeilen" und den "Rahmen" aus.

Ergebnis

Der Markierungsbereich der Tabellenanzeige ist projektiert. Um die Daten aus einem markierten Bereich zu exportieren, konfigurieren Sie den "Datenexport". Um die Daten aus einem markierten Bereich z. B. auszuwerten, projektieren Sie eine "Werteanzeige".

Siehe auch

Datenversorgung festlegen (Seite 747)

Sortierung von Tabellenspalten projektieren

Einleitung

Wenn der Bediener in Runtime einzelne Tabellenspalten sortieren können soll, konfigurieren Sie die Sortierung innerhalb der Tabelle.

Abhängig von der Projektierung kann der Bediener eine Tabellenspalte wie folgt sortieren:

- Klicken oder Doppelklicken auf die Spaltenüberschrift
- Klicken auf eine Sortierschaltfläche
  Über die Sortierschaltfläche wird die ausgewählte Spalte in der projektierten Reihenfolge sortiert. Damit die Sortierschaltfläche angezeigt wird, müssen Sie eine vertikale Bildlaufleiste projektieren.

Voraussetzung

- Tabellenanzeige oder Wertetabelle ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Sortierung innerhalb einer Tabelle zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Sortierung > Nach Spaltenüberschrift sortieren" entweder "Sortieren mit Doppelklick" oder "Sortieren mit Klick".
2. Wählen Sie die "Sortierreihenfolge".
3. Aktiveren Sie bei Bedarf "Sortiersymbol anzeigen" und "Sortierindex anzeigen".
4. Aktiveren Sie bei Bedarf "Sortierschaltfläche anzeigen".
5. Wenn Sie die Sortierschaltfläche verwenden, wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Fenster > Bildlauf > Vertikale Leiste > Immer".

Ergebnis

Die Sortierung der Tabellenspalten ist projektiert.

Datenversorgung festlegen

Einleitung

Mit der Datenversorgung legen Sie fest, aus welcher Quelle die Werte in Runtime am Bediengerät angezeigt werden. Folgende Quellen stehen zur Verfügung:

- Aktuelle Prozesswerte aus Variablen
- Archivierte Werte aus Archivvariablen
- Rezepturelemente (nur f(x)-Kurvenanzeige)
- Datenversorgung in Runtime per Skript

Voraussetzung

- Eine Online-Variable oder Archivvariable ist projektiert
- Wertspalte oder Kurve ist angelegt
- Inspektorfenster ist geöffnet

Aktuelle Prozesswerte darstellen

Um aktuelle Prozesswerte darzustellen, gehen Sie folgendermaßen vor:

1. Wenn Sie die Datenquelle für eine Tabellenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Wertspalten".
2. Wenn Sie die Datenquelle für eine Kurvenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kurven".
3. Konfigurieren Sie die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Variablen".
   - Wenn Sie die Kurve einer f(x)-Kurvenanzeige konfigurieren, wählen Sie unter "X-Werte" und "Y-Werte" jeweils eine Variable.
   - Wenn Sie die Kurve einer f(t)-Kurvenanzeige oder eine Wertspalte konfigurieren, wählen Sie eine "Variable".
   - Wählen Sie den "Aktualisierungszyklus.

Werte aus einem Archiv darstellen

Um Werte aus einem Archiv darzustellen, gehen Sie folgendermaßen vor:

1. Wenn Sie die Datenquelle für eine Tabellenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Wertspalten".
2. Wenn Sie die Datenquelle für eine Kurvenanzeige festlegen, klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kurven".
3. Konfigurieren Sie die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Archivvariablen".
   - Wenn Sie die Kurve einer f(x)-Kurvenanzeige konfigurieren, wählen Sie unter "X-Werte" und "Y-Werte" jeweils eine Variable.
   - Wenn Sie die Kurve einer f(t)-Kurvenanzeige oder eine Wertspalte konfigurieren, wählen Sie eine "Variable".

Werte aus Rezepturen darstellen

Werte aus Rezepturen geben Sie in einer f(x)-Kurvenanzeige aus.

Um Werte aus Rezepturen darzustellen, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Rezeptur".
   - Wählen Sie unter "X-Werte" und "Y-Werte" jeweils ein Rezepturelement.

Benutzerdefinierte Datenversorgung in Runtime einrichten

Wenn Sie erst in Runtime per Skript die Kurvenwerte mit Daten versorgen wollen, benötigen Sie ein Skript über die API-Schnittstelle.

Um eine Anbindung in Runtime per Skript herzustellen, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Kurven" die "Datenquelle":
   - Wählen Sie als "Quelle" den Eintrag "Benutzerdefiniert".
   - Binden Sie das Skript über die API-Schnittstelle an die entsprechende Kurve an.

Ergebnis

Die Datenversorgung ist projektiert.

Siehe auch

Grundlagen zu Variablen (Seite 537)

Grundlagen zur Variablenarchivierung (Seite 639)

Tabellenanzeige projektieren (Seite 741)

Wertspalten der Tabellenanzeige projektieren (Seite 743)

Symbolleiste und Statuszeile konfigurieren

Einleitung

Sie bedienen die Controls in Runtime über die Schaltflächen in der Symbolleiste. Die Statuszeile zeigt Statusmitteilungen des Controls an. Während der Projektierung konfigurieren Sie den Inhalt der Symbolleiste und Statuszeile.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Symbolleiste projektieren

Um die Symbolleiste zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Allgemein" die allgemeinen Eigenschaften der Symbolleiste, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen" die Schaltflächen, die Sie in Runtime benötigen. Informationen zu der Funktion der einzelnen Schaltflächen erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Schaltflächen festzulegen, markieren Sie in der Liste die Schaltfläche. Verschieben Sie dann die Schaltfläche an die gewünschte Position.
4. Legen Sie bei Bedarf für die Schaltflächen der Symbolleiste einen "Hotkey" fest.
5. Wählen Sie bei Bedarf die Berechtigung, die in Runtime zum Bedienen der Schaltfläche notwendig ist.
6. Wenn eine Schaltfläche in Runtime nicht bedient werden soll, deaktivieren Sie "Aktivieren". Sie können eine deaktivierte Schaltfläche z. B. mit einem Skript in Runtime wieder aktivieren.

Statuszeile projektieren

Um die Statuszeile zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Allgemein" die allgemeinen Eigenschaften der Statuszeile, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Elemente" die Elemente, die Sie in Runtime benötigen.
   Informationen zu den Elementen der Statuszeile erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Elemente festzulegen, markieren Sie in der Liste das Element. Verschieben Sie dann das Element an die gewünschte Position.
4. Um die Breite eines Elements in der Statuszeile anzupassen, deaktivieren Sie "Automatisch anpassen". Geben Sie dann einen Pixel-Wert für die Breite an.

Siehe auch

Bedienung der Kurven- und Tabellenanzeige in Runtime (Seite 761)

Tabellenanzeige projektieren (Seite 741)

Elemente der Statuszeile (Seite 782)

Export der Runtime-Daten projektieren

Einleitung

In Runtime können Sie die angezeigten Daten eines Controls im CSV-Format exportieren. Abhängig vom verwendeten Control und der Projektierung exportiert der Bediener alle Daten oder nur die markierten Daten.

Den Dateinamen der CSV-Datei sowie den Speicherort legt der Bediener beim Export fest. Alternativ können Sie den Dateinamen auch automatisch aus folgenden Platzhaltern generieren:

- @OBJECTNAME: Objektname des Controls
- @CURRENTDATE: aktuelles Datum
- @CURRENTTIME: aktuelle Zeit

Voraussetzung

- Control ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um den Export von Runtime-Daten zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Datenexport". Die Datenexport-Einstellungen sind bereits vorkonfiguriert.
2. Geben Sie bei Bedarf einen eigenen Dateinamen und einen Pfad für die Exportdatei ein.
3. Wählen Sie den "Umfang" des Datenexports.
4. Konfigurieren Sie die "Bedienung in Runtime".
5. Damit der Bediener den Datenexport starten kann, aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen > Export Data".

Ergebnis

In Runtime kann der Bediener die Daten des Controls über die Symbolleiste exportieren.

Siehe auch

Tabellenanzeige projektieren (Seite 741)

Auswirkung der Online-Projektierung bestimmen

Einleitung

In WinCC projektieren Sie das Aussehen und die Funktion eines Controls in Runtime. Zusätzlich können Sie festlegen, dass der Bediener in Runtime ein Control konfigurieren kann. Ob Konfigurationsänderungen in Runtime z. B. auch nach einem Bildwechsel erhalten bleiben, legen Sie Control unter "Eigenschaften > Eigenschaften > Security > Persistenz" fest:

- Keine Persistenz
  Die Änderungen an der Konfiguration werden nicht gespeichert. Die Änderungen werden beim nächsten Bildwechsel und beim Deaktivieren des Projekts verworfen.
- Persistenz
  Die Änderungen an der Konfiguration werden gespeichert. Die Änderungen bleiben auch nach dem Deaktivieren / Aktivieren des Projekts erhalten. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.
- Persistenz in Runtime
  Die Änderungen an der Konfiguration werden nur während Runtime gespeichert. Wenn Sie das Projekt deaktivieren, werden die Änderungen verworfen. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.

Hinweis

Wenn Sie ein projektiertes Control ändern und auf das Bediengerät laden, werden die Änderungen in Runtime unabhängig von den Einstellungen überschrieben.

Voraussetzung

- Ein Control projektiert
- Inspektorfenster ist geöffnet

Vorgehensweise

Um die Persistenz von Daten in Runtime zu konfigurieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Inspektorfenster "Eigenschaften > Eigenschaften > Security".
2. Wählen Sie die "Online-Projektierung".
3. Wählen Sie bei Bedarf die Berechtigung aus.
4. Wählen Sie das "Verhalten bei Bildwechsel".

Ergebnis

Das Verhalten bei der Online-Projektierung ist projektiert. In Runtime kann der Bediener abhängig von den Einstellungen die Konfiguration des Controls anpassen.

Wenn mehrere Bediener am Bediengerät arbeiten, werden bei einem Bedienerwechsel die individuellen Projektierungen erst nach einem Bildwechsel übernommen.

Siehe auch

Tabellenanzeige projektieren (Seite 741)

Online-Projektierung der Tabellenanzeige (Seite 775)

Aufbau der Zeitformatvorgabe

Zeitformat

Die vordefinierten Zeitformate der Auswahlliste sind überschreibbar.

Wenn Sie eigene Zeitformate festlegen, beachten Sie Folgendes:

- Leerzeichen, die in der Formatvorgabe Werte trennen, werden in der Zeitanzeige an die gleiche Stelle gesetzt.
- Beachten Sie die Groß- und Kleinschreibung.
- Zeichen, die in einfache Anführungszeichen gesetzt sind, werden in der Zeitanzeige unverändert an die gleiche Stelle gesetzt.

Um eine Formatvorgabe aufzubauen, verwenden Sie folgende Werte:

Wert	Beschreibung
h	Stunden, 12-Stunden-Uhr, für einstellige Stunden ohne führende Null
hh	Stunden, 12-Stunden-Uhr, für einstellige Stunden mit führender Null
H	Stunden, 24-Stunden-Uhr, für einstellige Stunden ohne führende Null
HH	Stunden, 24-Stunden-Uhr, für einstellige Stunden mit führender Null

Wert	Beschreibung
m	Minuten, ohne führende Null für einstellige Minuten
mm	Minuten, mit führender Null für einstellige Minuten
s	Sekunden, ohne führende Null für einstellige Sekunden
ss	Sekunden, mit führender Null für einstellige Sekunden
t	Zeitbezeichner AM/PM einstellig: A oder P
tt	Zeitbezeichner AM/PM zweistellig: AM oder PM

Beispiel

Um die Zeitanzeige "11:29:40 PM" aufzubauen, verwenden Sie folgende Formatvorgabe:

"hh:mm:ss tt"

Aufbau der Datumsformatvorgabe

Zeitformat

Die vordefinierten Datumsformate der Auswahlliste sind überschreibbar.

Wenn Sie eigene Datumsformate festlegen, beachten Sie Folgendes:

- Leerzeichen, die in der Formatvorgabe Werte trennen, werden in der Datumsanzeige an die gleiche Stelle gesetzt.
- Beachten Sie die Groß- und Kleinschreibung.
- Zeichen, die in einfache Anführungszeichen gesetzt sind, werden in der Datumsanzeige unverändert an die gleiche Stelle gesetzt.

Um eine Formatvorgabe aufzubauen, verwenden Sie folgende Werte:

Wert	Beschreibung
D	Tag des Monats als Zahl, für einstellige Tage ohne führende Null
Dd	Tag des Monats als Zahl, für einstellige Tage mit führender Null
Ddd	Wochentag mit drei Buchstaben abgekürzt
Dddd	Wochentag ausgeschrieben
M	Monat als Zahl, für einstellige Monate ohne führende Null
MM	Monat als Zahl, für einstellige Monate mit führender Null
MMM	Monat mit drei Buchstaben abgekürzt
MMMM	Monat ausgeschrieben
y	Jahr, nur die letzten beiden Zahlen, für Jahre unter 10 mit führender Null, gleich dem Format "yy"
yy	Jahr, nur die letzten beiden Zahlen, für Jahre unter 10 mit führender Null
yyyy	Vollständige, vierstellige Jahrsangabe

Beispiel

Um die Datumsanzeige "Mi, Aug 31 13" aufzubauen, verwenden Sie folgende Format-Vorgabe:

"Ddd, MMM dd yy"

2.10.3.4 Wertetabelle projektieren

Grundlagen zur Wertetabelle

Funktion

Die Wertetabelle zeigt Werte oder Statistiken aus einem Control an. Den Inhalt der Wertetabelle legen Sie bei dessen Projektierung fest.

Überblick über die Wertetabelle

Die Wertetabelle verbinden Sie mit einem der folgenden Controls:

- Tabellenanzeige
- f(x)-Kurvenanzeige
- f(t)-Kurvenanzeige

In der Wertetabelle legen Sie einen "Darstellungsmodus" fest. Der Darstellungsmodus bestimmt, welche Daten in der Wertetabelle angezeigt werden.

Darstellungsmodi

In der Wertetabelle stehen drei verschiedene Darstellungsmodi zur Verfügung:

- Linealfenster
  Das Linealfenster zeigt die Koordinatenwerte der Kurven am Lineal oder Werte einer selektierten Zeile in der Tabelle.
- Statistikbereich-Fenster
  Das Statistikbereich-Fenster zeigt die Werte der Untergrenze und Obergrenze der Kurven zwischen zwei Linealen oder des selektierten Bereichs in der Tabelle. Das Statistikbereich-Fenster können Sie nur mit der f(t)-Kurvenanzeige oder der Tabellenanzeige verbinden.
- Statistik-Fenster
  Das Statistik-Fenster zeigt die statistische Auswertung der Kurven. Die Statistik umfasst unter anderen:
  - Minimum
  - Maximum
  - Durchschnitt
  - Standardabweichung
  - Integral

Alle Fenster können auch Begleitdaten wie Zeitstempel zu den verbundenen Kurven oder Spalten anzeigen.

Siehe auch

Wertetabelle projektieren (Seite 755)

Statistik von Runtimedaten erstellen (Seite 779)

Koordinaten eines Punktes ermitteln (Seite 773)

Wertetabelle projektieren

Einleitung

Die Inhalte der Wertetabelle werden in Spalten dargestellt. Welche Spalten zur Verfügung stehen, ist abhängig vom verbundenen Control. Jeder Spalte ist ein Block zugeordnet. Über die Blöcke definieren Sie Ausrichtung und Aussehen der Spaltenüberschriften. Standardmäßig wird für das Anzeigeformat das Format des verbundenen Controls übernommen, z. B. die Zeitdarstellung.

Voraussetzung

- Tabellenanzeige oder Kurvenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um eine Wertetabelle zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Taskcard "Werkzeuge" die Tabellenanzeige in das Bild ein.
2. Um die Wertetabelle mit dem selektierten Control zu verbinden, wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Datenquelle".
3. Wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein" den "Modus" der Wertetabelle.
4. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Blöcke" die Eigenschaften der Spalten der Wertetabelle.
5. Konfigurieren Sie die "Eigenschaften > Eigenschaften > Spalten":
   - Ändern Sie bei Bedarf die "Sichtbarkeit"
   - Ändern Sie bei Bedarf die "Beschriftung"
   - Ändern Sie bei Bedarf über "Auf" und "Ab" die Reihenfolge der Spalten
6. Projektieren Sie im Inspektorfenster das Aussehen der Wertetabelle:
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung" das Aussehen der Wertetabelle.
   - Projektieren Sie unter "Eigenschaften > Eigenschaften > Fenster" die Eigenschaften der Wertetabelle in Runtime.

Ergebnis

Die Wertetabelle ist projektiert.

Siehe auch

Grundlagen zur Wertetabelle (Seite 754)

Symbolleiste und Statuszeile konfigurieren (Seite 756)

Aussehen der Tabellenelemente konfigurieren (Seite 757)

Sortierung von Tabellenspalten projektieren (Seite 758)

Export der Runtime-Daten projektieren (Seite 759)

Auswirkung der Online-Projektierung bestimmen (Seite 760)

Symbolleiste und Statuszeile konfigurieren

Einleitung

Sie bedienen die Controls in Runtime über die Schaltflächen in der Symbolleiste. Die Statuszeile zeigt Statusmitteilungen des Controls an. Während der Projektierung konfigurieren Sie den Inhalt der Symbolleiste und Statuszeile.

Voraussetzung

- Tabellenanzeige ist im Bild selektiert
- Inspektorfenster ist geöffnet

Symbolleiste projektieren

Um die Symbolleiste zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Allgemein" die allgemeinen Eigenschaften der Symbolleiste, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen" die Schaltflächen, die Sie in Runtime benötigen. Informationen zu der Funktion der einzelnen Schaltflächen erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Schaltflächen festzulegen, markieren Sie in der Liste die Schaltfläche. Verschieben Sie dann die Schaltfläche an die gewünschte Position.
4. Legen Sie bei Bedarf für die Schaltflächen der Symbolleiste einen "Hotkey" fest.
5. Wählen Sie bei Bedarf die Berechtigung, die in Runtime zum Bedienen der Schaltfläche notwendig ist.
6. Wenn eine Schaltfläche in Runtime nicht bedient werden soll, deaktivieren Sie "Aktivieren". Sie können eine deaktivierte Schaltfläche z. B. mit einem Skript in Runtime wieder aktivieren.

Statuszeile projektieren

Um die Statuszeile zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Allgemein" die allgemeinen Eigenschaften der Statuszeile, z. B. Ausrichtung oder Hintergrundfarbe.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Statuszeile > Statuszeile - Elemente" die Elemente, die Sie in Runtime benötigen.
   Informationen zu den Elementen der Statuszeile erhalten Sie bei der Beschreibung des jeweiligen WinCC Controls unter "Bedienung in Runtime".
3. Um die Reihenfolge der Elemente festzulegen, markieren Sie in der Liste das Element. Verschieben Sie dann das Element an die gewünschte Position.
4. Um die Breite eines Elements in der Statuszeile anzupassen, deaktivieren Sie "Automatisch anpassen". Geben Sie dann einen Pixel-Wert für die Breite an.

Siehe auch

Wertetabelle projektieren (Seite 755)

Elemente der Statuszeile (Seite 782)

Aussehen der Tabellenelemente konfigurieren

Einleitung

Das Aussehen der Tabellenelemente passen Sie Ihren Bedürfnissen an.

Voraussetzung

- Tabellenanzeige oder Wertetabelle ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Farben der Tabellenelemente zu konfigurieren, gehen Sie folgendermaßen vor:

1. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Überschrift" die folgenden Eigenschaften fest:
   - Sichtbarkeit der Spaltenüberschriften und Zeilenköpfe
   - Ausrichtung der Spaltenüberschriften und Zeilenköpfe
   - Farbgestaltung der Spaltenüberschriften
2. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Rumpf" die folgenden Eigenschaften fest:
   - Farbgestaltung des Tabelleninhalts, z. B. unterschiedliche Farben für Zeilen zur besseren Unterscheidung
   - Sichtbarkeit von leeren Spalten und Zeilen
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Raster" folgende Eigenschaften fest:
   - Gestaltung der Rasterlinien
   - Zeichenabstand in den Tabellenzellen

Ergebnis

Das Aussehen der Tabellenelemente ist projektiert. Zusätzlich können Sie noch weitere Eigenschaften festlegen:

- Das Verhalten und die Farbgestaltung beim Markieren eines Tabelleneintrags
- Die Sortierung von Tabelleneinträgen

Siehe auch

Wertetabelle projektieren (Seite 755)

Sortierung von Tabellenspalten projektieren (Seite 758)

Sortierung von Tabellenspalten projektieren

Einleitung

Wenn der Bediener in Runtime einzelne Tabellenspalten sortieren können soll, konfigurieren Sie die Sortierung innerhalb der Tabelle.

Abhängig von der Projektierung kann der Bediener eine Tabellenspalte wie folgt sortieren:

- Klicken oder Doppelklicken auf die Spaltenüberschrift
- Klicken auf eine Sortierschaltfläche
  Über die Sortierschaltfläche wird die ausgewählte Spalte in der projektierten Reihenfolge sortiert. Damit die Sortierschaltfläche angezeigt wird, müssen Sie eine vertikale Bildlaufleiste projektieren.

Voraussetzung

- Tabellenanzeige oder Wertetabelle ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um die Sortierung innerhalb einer Tabelle zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Sortierung > Nach Spaltenüberschrift sortieren" entweder "Sortieren mit Doppelklick" oder "Sortieren mit Klick".
2. Wählen Sie die "Sortierreihenfolge".
3. Aktiveren Sie bei Bedarf "Sortiersymbol anzeigen" und "Sortierindex anzeigen".
4. Aktiveren Sie bei Bedarf "Sortierschaltfläche anzeigen".
5. Wenn Sie die Sortierschaltfläche verwenden, wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Fenster > Bildlauf > Vertikale Leiste > Immer".

Ergebnis

Die Sortierung der Tabellenspalten ist projektiert.

Siehe auch

Wertetabelle projektieren (Seite 755)

Export der Runtime-Daten projektieren

Einleitung

In Runtime können Sie die angezeigten Daten eines Controls im CSV-Format exportieren. Abhängig vom verwendeten Control und der Projektierung exportiert der Bediener alle Daten oder nur die markierten Daten.

Den Dateinamen der CSV-Datei sowie den Speicherort legt der Bediener beim Export fest. Alternativ können Sie den Dateinamen auch automatisch aus folgenden Platzhaltern generieren:

- @OBJECTNAME: Objektname des Controls
- @CURRENTDATE: aktuelles Datum
- @CURRENTTIME: aktuelle Zeit

Voraussetzung

- Control ist im Bild selektiert
- Inspektorfenster ist geöffnet

Vorgehen

Um den Export von Runtime-Daten zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Datenexport". Die Datenexport-Einstellungen sind bereits vorkonfiguriert.
2. Geben Sie bei Bedarf einen eigenen Dateinamen und einen Pfad für die Exportdatei ein.
3. Wählen Sie den "Umfang" des Datenexports.
4. Konfigurieren Sie die "Bedienung in Runtime".
5. Damit der Bediener den Datenexport starten kann, aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen > Export Data".

Ergebnis

In Runtime kann der Bediener die Daten des Controls über die Symbolleiste exportieren.

Siehe auch

Wertetabelle projektieren (Seite 755)

Auswirkung der Online-Projektierung bestimmen

Einleitung

In WinCC projektieren Sie das Aussehen und die Funktion eines Controls in Runtime. Zusätzlich können Sie festlegen, dass der Bediener in Runtime ein Control konfigurieren kann. Ob Konfigurationsänderungen in Runtime z. B. auch nach einem Bildwechsel erhalten bleiben, legen Sie Control unter "Eigenschaften > Eigenschaften > Security > Persistenz" fest:

- Keine Persistenz
  Die Änderungen an der Konfiguration werden nicht gespeichert. Die Änderungen werden beim nächsten Bildwechsel und beim Deaktivieren des Projekts verworfen.
- Persistenz
  Die Änderungen an der Konfiguration werden gespeichert. Die Änderungen bleiben auch nach dem Deaktivieren / Aktivieren des Projekts erhalten. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.
- Persistenz in Runtime
  Die Änderungen an der Konfiguration werden nur während Runtime gespeichert. Wenn Sie das Projekt deaktivieren, werden die Änderungen verworfen. Unter "Verhalten bei Bildwechsel" können Sie zusätzlich festlegen, dass Änderungen bei einem Bildwechsel verworfen werden.

Hinweis

Wenn Sie ein projektiertes Control ändern und auf das Bediengerät laden, werden die Änderungen in Runtime unabhängig von den Einstellungen überschrieben.

Voraussetzung

- Ein Control projektiert
- Inspektorfenster ist geöffnet

Vorgehensweise

Um die Persistenz von Daten in Runtime zu konfigurieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Inspektorfenster "Eigenschaften > Eigenschaften > Security".
2. Wählen Sie die "Online-Projektierung".
3. Wählen Sie bei Bedarf die Berechtigung aus.
4. Wählen Sie das "Verhalten bei Bildwechsel".

Ergebnis

Das Verhalten bei der Online-Projektierung ist projektiert. In Runtime kann der Bediener abhängig von den Einstellungen die Konfiguration des Controls anpassen.

Wenn mehrere Bediener am Bediengerät arbeiten, werden bei einem Bedienerwechsel die individuellen Projektierungen erst nach einem Bildwechsel übernommen.

Siehe auch

Wertetabelle projektieren (Seite 755)

2.10.3.5 Controls in Runtime bedienen

Bedienung der Kurven- und Tabellenanzeige in Runtime

Einleitung

Sie bedienen die Kurven- und Tabellenanzeige über die Schaltflächen der Symbolleiste.

Um eine Schaltfläche über eine Skript zu bedienen, übergeben Sie die "ID" einer Schaltfläche an das Skript. Um die Schaltfläche zu beschriften, können Sie die Symbole aus der Symbolbibliothek verwenden. Sie finden die Symbole in der Taskcard "Werkzeuge > Grafiken" unter "WinCC Grafikordner > Runtime control icons > Trend View".

Schaltfläche der f(x)-Kurvenanzeige

Die folgende Tabelle zeigt die Schaltflächen, die bei der f(x)-Kurvenanzeige verfügbar sind:

Symbol	Name	Funktion	ID
	"Tooltip"	Ruft die Hilfe zum Control auf.	1
	"Konfigurationsdialog"	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften des Controls verändern	2
	"Lineal"	Blendet ein Lineal ein, das die Koordinaten des Schnittpunkts mit einer Kurve in der Wertetabelle anzeigt. Voraussetzung: Wertetabelle mit Darstellungsmodus "Linealfenster" ist projektiert.	3
	"Zoombereich"	Vergrößert den Ausschnitt der Kurvenanzeige. Den Ausschnitt definieren Sie durch Ziehen mit der Maus. Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	4
	"Zoom +/-"	Vergrößert oder verkleinert die Kurven in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	5
	"Zoom X-Achse +/-"	Vergrößert oder verkleinert die Zeitachse in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	6
	"Zoom Y-Achse +/-"	Vergrößert oder verkleinert die Wertachse in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	7
	"Kurvenbereich verschieben"	Verschiebt die Kurven entlang der Wertachse und der Zeitachse	8
	"Achsenbereich verschieben"	Verschiebt die Kurven entlang der Wertachse	9
	"Originalansicht"	Kehrt aus der vergrößerten Darstellung wieder zur Originalansicht zurück	10
	"Datenanbindung wählen"	Öffnet einen Dialog, in dem Sie Datenquelle wählen: • Prozesswertarchiv • Variable • Rezeptur (nur f(x)-Kurvenanzeige)	11
	"Kurven wählen"	Öffnet einen Dialog, in dem Sie Kurven aus- oder einblenden. Zusätzlich können Sie wählen, welche Kurve im Vordergrund dargestellt wird.	12

Symbol	Name	Funktion	ID
	"Zeitbereich wählen"	Öffnet einen Dialog, in dem Sie den Zeitbereich konfigurieren	13
	"Vorherige Kurve"	Stellt die vorhergehende Kurve im Vordergrund dar.	14
	"Nächste Kurve"	Stellt die nächste Kurve im Vordergrund dar.	15
	"Stopp"	Stoppt die aktualisierte Darstellung. Die Daten werden zwischengespeichert.	16
	"Start"	Setzt die aktualisierte Darstellung fort. Die zwischengespeicherten Werte werden nachgetragen.	16
	"Drucken"	Druckt die selektierten Werte. Den Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest	17
	"Backup verbinden"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime verbinden	18
	"Backup trennen"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime trennen	19
	"Daten exportieren"	Exportiert alle oder ausgewählte Daten in eine *.CSV-Datei. Abhängig von der Projektierung und den Berechtigungen können folgende Optionen verfügbar sein: • Exporteinstellungen anzeigen und Export starten • Dateiname und Verzeichnis wählen	20
	"Benutzerdefiniert 1"	Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert	1001

Schaltflächen der f(t)-Kurvenanzeige

Die folgende Tabelle zeigt die Schaltflächen, die bei der f(t)-Kurvenanzeige verfügbar sind:

Symbol	Name	Funktion	ID
	"Tooltip"	Ruft die Hilfe zum Control auf.	1
	"Konfigurationsdialog"	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften des Controls verändern	2
	"Erster Datensatz"	Stellt beginnend mit dem ersten archivierten Wert den Verlauf einer Variablen innerhalb eines festgelegten Zeitbereichs dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	3
	"Vorheriger Datensatz"	Stellt ausgehend vom aktuell dargestellten Zeitintervall den Verlauf einer Variablen innerhalb des vorhergehenden Zeitintervalls dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	4

Symbol	Name	Funktion	ID
	"Nächster Datensatz"	Stellt ausgehend vom aktuell dargestellten Zeitintervall den Verlauf einer Variablen innerhalb des nachfolgenden Zeitintervalls dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	5
	"Letzter Datensatz"	Stellt endend mit dem letzten archivierten Wert den Verlauf einer Variablen innerhalb eines festgelegten Zeitbereichs dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	6
	"Lineal"	Blendet ein Lineal ein, das die Koordinaten des Schnittpunkts mit einer Kurve in der Wertetabelle anzeigt. Voraussetzung: Wertetabelle mit Darstellungsmodus "Linealfenster" ist projektiert.	7
	"Zoombereich"	Vergrößert den Ausschnitt der Kurvenanzeige. Den Ausschnitt definieren Sie durch Ziehen mit der Maus. Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	8
	"Zoomen +/-"	Vergrößert oder verkleinert die Kurven in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	9
	"Zoom Zeitachse +/-"	Vergrößert oder verkleinert die Zeitachse in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	10
	"Zoom Wertachse +/-"	Vergrößert oder verkleinert die Wertachse in der Kurvenanzeige. Linke Maustaste: Vergrößern <Umschalt + Linke Maustaste>: Verkleinern Über die Schaltfläche "Originalansicht" kehren Sie zur Originalansicht zurück.	11
	"Kurvenbereich verschieben"	Verschiebt die Kurven entlang der Wertachse und der Zeitachse	12
	"Achsenbereich verschieben"	Verschiebt die Kurven entlang der Wertachse	13
	"Originalansicht"	Kehrt aus der vergrößerten Darstellung wieder zur Originalansicht zurück	14
	"Datenanbindung wählen"	Öffnet einen Dialog, in dem Sie Datenquelle wählen: • Prozesswertarchiv • Variable • Rezeptur (nur f(x)-Kurvenanzeige)	15
	"Kurven wählen"	Öffnet einen Dialog, in dem Sie Kurven aus- oder einblenden. Zusätzlich können Sie wählen, welche Kurve im Vordergrund dargestellt wird.	16
	"Zeitbereich wählen"	Öffnet einen Dialog, in dem Sie den Zeitbereich konfigurieren	17

Symbol	Name	Funktion	ID
	"Vorhergehende Kurve"	Stellt die vorhergehende Kurve im Vordergrund dar.	18
	"Nächste Kurve"	Stellt die nächste Kurve im Vordergrund dar.	19
	"Stopp"	Stoppt die aktualisierte Darstellung. Die Daten werden zwischengespeichert.	20
	"Start"	Setzt die aktualisierte Darstellung fort. Die zwischengespeicherten Werte werden nachgetragen.	20
	"Drucken"	Druckt die selektierten Werte. Den Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest	21
	"Statistischer Bereich"	Legt den Zeitraum zur Berechnung der Statistik fest.	22
	"Statistische Analyse"	Zeigt im Statistikfenster der Wertetabelle die statistischen Werte aus einem festgelegten "Statistikbereich" der Kurven- oder Tabellenanzeige an. Nur bei projektierter Wertetabelle verfügbar	23
	"Backup verbinden"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime verbinden	24
	"Backup trennen"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime trennen	25
	"Daten exportieren"	Exportiert alle oder ausgewählte Daten in eine *.CSV-Datei. Abhängig von der Projektierung und den Berechtigungen können folgende Optionen verfügbar sein: • Exporteinstellungen anzeigen und Export starten • Dateiname und Verzeichnis wählen	26
	"Relativachse"	Schaltet von der Darstellung der absoluten Werte auf die Prozentwert-Darstellung der Werteachse um. Die Untergrenze und Obergrenze der Kurven entsprechen dem Bereich 0 % bis 100 %.	27
	"Benutzerdefiniert 1"	Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert	1001

Schaltflächen der Tabellenanzeige

Die folgende Tabelle zeigt die Schaltflächen, die bei der Tabellenanzeige verfügbar sind:

Symbol	Name	Funktion	ID
	"Tooltip"	Ruft die Hilfe zum Control auf.	1
	"Konfigurationsdialog"	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften des Controls verändern	2
	"Erster Datensatz"	Stellt beginnend mit dem ersten archivierten Wert den Verlauf einer Variablen innerhalb eines festgelegten Zeitbereichs dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	3

Symbol	Name	Funktion	ID
	"Vorheriger Datensatz"	Stellt ausgehend vom aktuell dargestellten Zeitintervall den Verlauf einer Variablen innerhalb des vorhergehenden Zeitintervalls dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	4
	"Nächster Datensatz"	Stellt ausgehend vom aktuell dargestellten Zeitintervall den Verlauf einer Variablen innerhalb des nachfolgenden Zeitintervalls dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	5
	"Letzter Datensatz"	Stellt endend mit dem letzten archivierten Wert den Verlauf einer Variablen innerhalb eines festgelegten Zeitbereichs dar. Voraussetzung: Werte stammen aus einem Prozesswertarchiv.	6
	"Bearbeiten"	Aktiviert die Bearbeitung von Tabelleneinträgen. Um einen Wert zu bearbeiten, doppelklicken Sie in die gewünschte Tabellenzelle. Voraussetzung: Aktualisierte Darstellung ist angehalten.	7
	"Zeilen kopieren"	Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage. Voraussetzung: Aktualisierte Darstellung ist angehalten.	8
	"Datenanbindung wählen"	Öffnet einen Dialog, in dem Sie Datenquelle wählen: • Prozesswertarchiv • Variable • Rezeptur (nur f(x)-Kurvenanzeige)	9
	"Spalten wählen"	Öffnet einen Dialog, in dem Sie Spalten aus- oder einblenden und die Reihenfolge der Spalten ändern.	10
	"Zeitbereich wählen"	Öffnet einen Dialog, in dem Sie den Zeitbereich konfigurieren	11
	"Vorherige Spalte"	Verschiebt die Wertespalte vor die vorhergehende Wertespalte. Die Funktion bezieht sich auf die Wertespalten, die einer Zeitachse zugeordnet sind.	12
	"Nächste Spalte"	Verschiebt die Wertespalte hinter die nächste Wertespalte. Die Funktion bezieht sich auf die Wertespalten, die einer Zeitachse zugeordnet sind.	13
	"Stopp"	Stoppt die aktualisierte Darstellung. Die Daten werden zwischengespeichert.	14
	"Start"	Setzt die aktualisierte Darstellung fort. Die zwischengespeicherten Werte werden nachgetragen.	14
	"Drucken"	Druckt die selektierten Werte. Den Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest	15
	"Statistischer Bereich"	Legt den Zeitraum zur Berechnung der Statistik fest.	16
	"Statistische Analyse"	Zeigt im Statistikfenster der Wertetabelle die statistischen Werte aus einem festgelegten "Statistikbereich" der Kurven- oder Tabellenanzeige an. Nur bei projektierter Wertetabelle verfügbar	17

Symbol	Name	Funktion	ID
	"Backup verbinden"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime verbinden	18
	"Backup trennen"	Öffnet einen Dialog, in dem Sie ausgewählte Archive von WinCC Runtime trennen	19
	"Daten exportieren"	Exportiert alle oder ausgewählte Daten in eine *.CSV-Datei. Abhängig von der Projektierung und den Berechtigungen können folgende Optionen verfügbar sein: • Exporteinstellungen anzeigen und Export starten • Dateiname und Verzeichnis wählen	20
	"Benutzerdefiniert 1"	Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert	1001

Schaltflächen der Wertetabelle

Die folgende Tabelle zeigt die Schaltflächen, die bei der Wertetabelle verfügbar sind:

Symbol	Name	Funktion	ID
	"Tooltip"	Ruft die Hilfe zum Control auf.	1
	"Konfigurationsdialog"	Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften des Controls verändern	2
	"Lineal"	Blendet ein Lineal ein, das die Koordinaten des Schnittpunkts mit einer Kurve in der Wertetabelle anzeigt. Voraussetzung: Wertetabelle mit Darstellungsmodus "Linealfenster" ist projektiert.	3
	"Statistischer Bereich"	Legt den Zeitraum zur Berechnung der Statistik fest.	4
	"Statistische Analyse"	Zeigt im Statistikfenster der Wertetabelle die statistischen Werte aus einem festgelegten "Statistikbereich" der Kurven- oder Tabellenanzeige an. Nur bei projektierter Wertetabelle verfügbar	5
	"Drucken"	Druckt die selektierten Werte. Den Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest	6
	"Daten exportieren"	Exportiert alle oder ausgewählte Daten in eine *.CSV-Datei. Abhängig von der Projektierung und den Berechtigungen können folgende Optionen verfügbar sein: • Exporteinstellungen anzeigen und Export starten • Dateiname und Verzeichnis wählen	7
	"Benutzerdefiniert 1"	Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert	1001

Kurvenanzeige in Runtime bedienen

Online-Projektierung der f(t)-Kurvenanzeige

Einleitung

In Runtime projektieren Sie online und verändern somit die Darstellung der Kurvenanzeige. Bei der Projektierung der Tabellenanzeige wird festgelegt, ob Online-Projektierungen bei einem Bildwechsel oder nach dem Beenden von Runtime behalten oder verworfen werden.

Überblick

Mit den folgenden Schaltflächen ist eine Online-Projektierung in der Kurvenanzeige möglich:

- Schaltfläche "Konfigurationsdialog"
- Schaltfläche "Datenanbindung wählen"
- Schaltfläche "Kurven wählen"
- Schaltfläche "Zeitbereich wählen"

Konfigurationsdialog aufrufen

Um den Konfigurationsdialog aufzurufen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste auf .

Der Konfigurationsdialog wird geöffnet. Im Konfigurationsdialog können Sie z. B. die Tabellendarstellung ändern.

Hinweis

Drucken eines festen Zeitbereichs

Wenn Sie einen definierten Zeitbereich drucken wollen, stellen Sie sicher, dass folgende Optionen im Konfigurationsdialog deaktiviert sind:

- "Zeitachsen > Aktualisieren"
- "Allgemeines > Bildaufschlag > Aktualisierung starten" .

Datenanbindung ändern

Die folgende Tabelle zeigt Konfigurationsmöglichkeiten für die Datenanbindung:

Feld	Beschreibung
Kurve	Wählen Sie eine der projektierten Kurven aus.
Datenversorgung	Legen Sie fest, ob die ausgewählte Kurve durch eine Archivvariable oder Online-variable versorgt wird.
Variablenname	Wählen Sie den Variablennamen der Datenanbindung.

Um die Datenanbindung zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste auf .
   Der Dialog "Archiv-/Variablenselektion" wird geöffnet.
2. Wählen Sie die "Wertspalte" aus, für die Sie die Datenanbindung ändern wollen.
3. Wählen Sie die "Datenversorgung" und den "Variablennamen".

Zeitbereich ändern

Die folgende Tabelle zeigt die Konfigurationsmöglichkeiten für den Zeitbereich:

Feld	Beschreibung
Zeitachse	Wählen Sie die projektierte Zeitachse aus, für die Sie einen Zeitbereich festlegen wollen.
Zeitbereich	Legen Sie den Zeitbereich fest: • Wenn Sie ein festes Zeitintervall vorgeben wollen, wählen Sie die Einstellung "Beginn- bis Endzeitpunkt". Geben Sie jeweils das Datum und die Uhrzeit an. • Wenn Sie einen Zeitraum festlegen wollen, wählen Sie die Einstellung "Zeitbereich". Legen Sie für den Beginnzeitpunkt das Datum und die Uhrzeit fest. Die Länge des darzustellenden Zeitintervalls errechnet sich aus der Multiplikation von "Faktor" und "Zeiteinheit". • Wenn Sie eine bestimmte Anzahl von Werten anzeigen wollen, wählen Sie die Einstellung "Anzahl der Messpunkte". Legen Sie für den Beginnzeitpunkt das Datum und die Uhrzeit fest. Geben Sie die gewünschte Anzahl der Messpunkte in das Eingabefeld ein.

Um den Zeitbereich zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste der Kurvenanzeige auf .
   Der Dialog "Zeit - Auswahl" wird geöffnet.
2. Wählen Sie die "Zeitachse" aus, deren Zeitbereich Sie anpassen wollen.
   Wenn die Kurven einer Kurvenanzeige mit einer gemeinsamen Zeitachse dargestellt werden, gilt der angegebene Zeitbereich für alle Kurven.
3. Konfigurieren Sie den Zeitbereich.
   Das Eingabeformat von Datum und Zeit ist abhängig von der verwendeten Runtimesprache.

Siehe auch

Kurve im Vordergrund darstellen (Seite 772)

Zoomfunktionen in den Kurvenfenstern verwenden (Seite 770)

Starten und Stoppen der Aktualisierung (Seite 779)

Statistik von Runtimedaten erstellen (Seite 779)

Archivierte Werte darstellen (Seite 781)

Elemente der Statuszeile (Seite 782)

Kurvenanzeige projektieren (Seite 728)

Zoomfunktionen in den Kurvenfenstern verwenden

Einleitung

Über Tastenfunktionen können Sie Kurven, Achsen und beliebige Ausschnitte des Kurvenfensters vergrößern, verkleinern oder zur Originalansicht zurückkehren.

Überblick

Folgende Zoomfunktionen stehen im Kurvenfenster zur Verfügung:

- "Zoomen Ausschnitt"
- "Originalansicht"
- "Zoomen +/-"
- "Zeitachse zoomen +/-"
- "Wertachse zoomen +/-"
- "Kurvenbereich verschieben"

Voraussetzung

- Kurvenanzeige ist geöffnet
- Schaltflächen mit Zoomfunktionen sind projektiert
- Runtime ist aktiviert

Ausschnitt der Kurvenanzeige vergrößern

Um den Ausschnitt der Kurvenanzeige zu vergrößern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf .
   Die aktualisierte Darstellung wird gestoppt.
2. Ziehen Sie mit Drag&Drop den Bereich auf, der vergrößert dargestellt werden soll.
   Wenn sich innerhalb des Bereichs mindestens zwei Messwerte befinden, wird der Bereich der Kurve vergrößert dargestellt.
3. Um die Kurvenanzeige wieder in der Originalansicht darzustellen, klicken Sie auf .
4. Um die Aktualisierung wieder zu starten, klicken Sie auf .

Für die Achse werden die voreingestellten Werte übernommen.

Kurven vergrößern oder verkleinern

Wenn Sie eine Kurve vergrößern oder verkleinern, ist der 50%-Wert der Kurven immer in der Mitte der Werteachsen.

Um eine Kurve zu vergrößern oder zu verkleinern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf .
   Die aktualisierte Darstellung wird gestoppt.
2. Um eine Kurve zu vergrößern, klicken Sie mit der linken Maustaste auf die Kurve.
3. Um eine Kurve zu verkleinern, halten Sie <Umschalt> gedrückt und klicken Sie mit der linken Maustaste auf die Kurve.
4. Um die Kurvenanzeige wieder in der Originalansicht darzustellen, klicken Sie auf .
5. Um die Aktualisierung wieder zu starten, klicken Sie auf .

Für die Achse werden die voreingestellten Werte übernommen.

Hinweis

Wenn Sie während des Zoomens im Konfigurationsdialog auf der Registerkarte "Werteachsen" den Wertebereich einer Wertachse ändern, wird der sichtbare Zoombereich auf den neuen Wertebereich eingestellt.

Zeitachse oder Wertachse zoomen

Wenn Sie die Zeit- oder Wertachse zoomen, ist der 50%-Wert der Kurven immer in der Mitte der Achsen.

Um die Zeitachse oder Wertachse zu zoomen, gehen Sie folgendermaßen vor:

1. Um die Zeitachse zu zoomen, klicken Sie auf .
   Die aktualisierte Darstellung wird gestoppt.
2. Um die Wertachse zu zoomen, klicken Sie auf .
   Die aktualisierte Darstellung wird gestoppt.
3. Um eine Achse zu vergrößern, klicken Sie mit der linken Maustaste auf die Kurvenanzeige.
4. Um eine Achse zu verkleinern, halten Sie <Umschalt> gedrückt und klicken Sie mit der linken Maustaste auf die Kurvenanzeige.
5. Um die Kurvenanzeige wieder in der Originalansicht darzustellen, klicken Sie auf .
6. Um die Aktualisierung wieder zu starten, klicken Sie auf .

Für die Achse werden die voreingestellten Werte übernommen.

Kurvenbereich verschieben

Um den Kurvenbereich zu verschieben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf .
   Die aktualisierte Darstellung wird gestoppt.
2. Halten Sie die linke Maustaste gedrückt und bewegen Sie das Kreuz in der Kurvenanzeige in die gewünschte Richtung.
   Der dargestellte Bereich im Kurvenfenster auf der Zeitachse und auf der Wertachse wird angepasst.
3. Um Kurvenfenster wieder in der ursprünglichen Ansicht darzustellen, klicken Sie erneut auf .

Siehe auch

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Kurve im Vordergrund darstellen

Einleitung

Wenn mehrere Kurven in einem Kurvenfenster angezeigt werden, können Sie über Tastenfunktionen bestimmen, welche Kurve im Vordergrund dargestellt wird.

Voraussetzung

- Folgende Schaltflächen sind projektiert:
  - "Kurven wählen
  - "Vorhergehende Kurve"
  - "Nächste Kurve"
- Projekt ist aktiviert

Vorgehensweise

Um eine Kurve im Vordergrund darzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf .
   Der Dialog zum Einblenden und Ausblenden von Kurven wird geöffnet. In diesem Dialog können Sie auch festlegen, welche Kurve im Vordergrund ist.

   Hinweis

   Die erste Kurve eines Kurvenfensters kann nicht ausgeblendet werden.

2. Um die nächste Kurve im Vordergrund darzustellen, klicken Sie auf .
3. Um die vorhergehende Kurve im Vordergrund darzustellen, klicken Sie auf .

Siehe auch

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Koordinaten eines Punktes ermitteln

Einleitung

Mit der Schaltfläche "Lineal" ermitteln Sie mit einem Lineal die Koordinaten eines Punkts der Kurve. Um die Ermittlung der Koordinaten zu erleichtern, können Sie einen Ausschnitt der Kurven vergrößern. Wenn Sie das Lineal in der Kurvenanzeige eingeblendet haben, können Sie das Lineal jederzeit verschieben.

Wenn Sie mit der Maus auf der Kurve klicken, werden einige Kurvenparameter im Tooltip deder Kurvenanzeige angezeigt.

Voraussetzung

- Eine Kurvenanzeige ist projektiert
- Eine Wertetabelle ist projektiert und mit der Kurvenanzeige verbunden
- In der Wertetabelle ist der Darstellungsmodus "Linealfenster" aktiviert
- Runtime ist aktiviert

Vorgehen

Um die Koordinaten eines Punkts zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Kurvenanzeige auf .
   Das Lineal wird eingeblendet.
2. Verschieben Sie das Lineal mit der Maus an die gewünschte Stelle.
3. Wenn Sie einen Ausschnitt vergrößern wollen, klicken Sie auf .
   – Verschieben Sie das Lineal mit der Maus an die gewünschte Stelle.
   – Um zur Originalansicht zurückzukehren, klicken Sie auf .

Ergebnis

Im Linealfenster der Wertetabelle werden neben dem X-Wert / Zeitstempel und dem Y-Wert die Daten in den Spalten angezeigt, die Sie in der Wertetabelle projektiert haben.

In der Wertetabelle können bei den Werten zusätzlich die Indizes "i" und "u" angezeigt werden:

- "i.": Der angezeigte Wert ist ein interpolierter Wert.
- "u.": Der angezeigte Wert hat einen unsicheren Status:
  - Der Anfangswert nach dem Aktivieren von Runtime nicht bekannt
  - Ein Ersatzwert wird verwendet

Hinweis

Den Status "unsicher" eines Werts können Sie auch im dargestellten Kurvenverlauf anzeigen. Aktivieren Sie dazu auf der Registerkarte "Kurven" unter "Grenzwerte" die Option "Wert mit unsicherem Status".

Das folgende Bild zeigt die Koordinaten eines Punkts im Linealfenster der Wertetabelle:

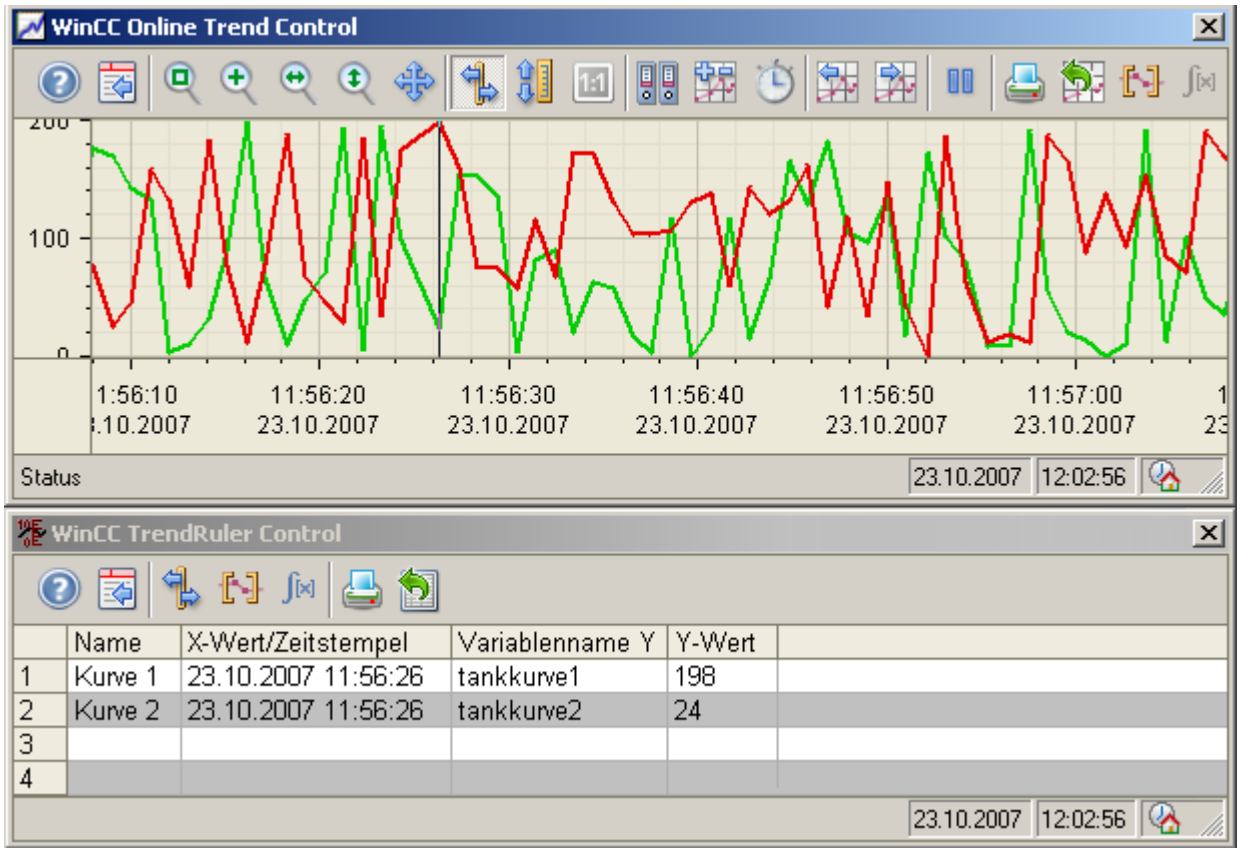

Alternatives Vorgehen

Alternativ können Sie die Wertetabelle auch mit der Tabellenanzeige verbinden. Im Darstellungsmodus "Linealfenster" werden in der Wertetabelle die Werte der selektierten Zeile angezeigt.

Siehe auch

Grundlagen zur Wertetabelle (Seite 754)

Tabellenanzeige in Runtime bedienen

Online-Projektierung der Tabellenanzeige

Einleitung

In Runtime projektieren Sie online und verändern somit die Darstellung der Tabellenanzeige. Bei der Projektierung der Tabellenanzeige wird festgelegt, ob Online-Projektierungen bei einem Bildwechsel oder nach dem Beenden von Runtime behalten oder verworfen werden.

Überblick

Mit den folgenden Schaltflächen ist eine Online-Projektierung in der Tabellenanzeige möglich:

- Schaltfläche "Konfigurationsdialog"
- Schaltfläche "Datenanbindung wählen"
- Schaltfläche "Spalten wählen"
- Schaltfläche "Zeitbereich wählen"

Konfigurationsdialog aufrufen

Um den Konfigurationsdialog aufzurufen, gehen Sie folgendermaßen vor:

Klicken Sie in der Symbolleiste auf .

Der Konfigurationsdialog wird geöffnet. Im Konfigurationsdialog können Sie z. B. die Tabellendarstellung ändern.

Datenanbindung ändern

Die folgende Tabelle zeigt Konfigurationsmöglichkeiten für die Datenanbindung:

Feld	Beschreibung
Wertspalte	Wählen Sie die projektierte Wertespalte aus, für die Sie die Datenanbindung ändern wollen.
Datenversorgung	Legen Sie fest, ob die ausgewählte Wertespalte durch eine Archivvariable oder Onlinevariable versorgt wird.
Variablenname	Wählen Sie den Variablenname der Datenanbindung.

Um die Datenanbindung zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste auf .
   Der Dialog "Archiv-/Variablenselektion" wird geöffnet.
2. Wählen Sie die "Wertspalte" aus, für die Sie die Datenanbindung ändern wollen.
3. Wählen Sie die "Datenversorgung" und den "Variablennamen".

Spalten ein- oder ausblenden

Um Spalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste auf .
   Der Dialog zum Ein- und Ausblenden der Spalten wird geöffnet.
2. Verändern Sie bei Bedarf die Reihenfolge der Wertespalten, die einer Zeitspalte zugeordnet sind.

Die Wertespalten sind nur bezüglich der fest verankerten Zeitspalte verschiebbar.

Hinweis

Die erste Spalte einer Tabellenanzeige ist nicht ausblendbar.

Zeitbereich ändern

Die folgende Tabelle zeigt die Konfigurationsmöglichkeiten für den Zeitbereich:

Feld	Beschreibung
Zeitspalte	Wählen Sie die projektierte Zeitspalte aus, für die Sie einen Zeitbereich festlegen wollen.
Zeitbereich	Legen Sie den Zeitbereich fest: • Wenn Sie ein festes Zeitintervall vorgeben wollen, wählen Sie die Einstellung "Beginn- bis Endzeitpunkt". Geben Sie jeweils das Datum und die Uhrzeit an. • Wenn Sie einen Zeitraum festlegen wollen, wählen Sie die Einstellung "Zeitbereich". Legen Sie für den Beginnzeitpunkt das Datum und die Uhrzeit fest. Die Länge des darzustellenden Zeitintervalls errechnet sich aus der Multiplikation von "Faktor" und "Zeiteinheit". • Wenn Sie eine bestimmte Anzahl von Werten anzeigen wollen, wählen Sie die Einstellung "Anzahl der Messpunkte". Legen Sie für den Beginnzeitpunkt das Datum und die Uhrzeit fest. Geben Sie die gewünschte Anzahl der Messpunkte in das Eingabefeld ein.

Um den Zeitbereich zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Symbolleiste der Tabellenanzeige auf
   Der Dialog "Zeit - Auswahl" wird geöffnet.
2. Wählen Sie die "Zeitspalte" aus, deren Zeitbereich Sie anpassen wollen.
   Wenn die Spalten einer Tabellenanzeige mit einer gemeinsamen Zeitachse dargestellt werden, gilt der angegebene Zeitbereich für alle Spalten.
3. Konfigurieren Sie den Zeitbereich.
   Das Eingabeformat von Datum und Zeit ist abhängig von der verwendeten Runtimesprache.

Siehe auch

Tabellenfeld in Runtime bearbeiten (Seite 777)

Spalten in der Tabelle verschieben (Seite 778)

Starten und Stoppen der Aktualisierung (Seite 779)

Statistik von Runtimedaten erstellen (Seite 779)

Archivierte Werte darstellen (Seite 781)

Elemente der Statuszeile (Seite 782)

Tabellenanzeige projektieren (Seite 741)

Tabellenfeld in Runtime bearbeiten

Einleitung

Sie ändern die in der Tabellenanzeige dargestellten Werte manuell über die Schaltfläche "Bearbeiten".

Voraussetzung

- Tabellenanzeige ist projektiert
- Schaltfläche "Bearbeiten" ist projektiert
- Runtime ist aktiviert

Vorgehen

Um ein Tabellenfeld in Runtime zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie in einer Tabellenanzeige auf .
   Die aktualisierte Darstellung wird gestoppt, die Prozessdaten werden weiter archiviert.
2. Klicken Sie auf .
3. Doppelklicken Sie auf das gewünschte Tabellenfeld einer Wertespalte.

4. Geben Sie den neuen Wert ein.
   Der geänderte Wert wird archiviert.
5. Um er Darstellung der Runtimedaten in der Tabellenanzeige fortzufahren, klicken Sie auf .

Siehe auch

Online-Projektierung der Tabellenanzeige (Seite 775)

Spalten in der Tabelle verschieben

Einleitung

Die Zeitspalte wird immer in der ersten Spalte der Tabelle angezeigt. Dann folgen die Wertespalten, die dieser Zeitspalte zugeordnet sind. Wenn mehrere Zeitspalten projektiert sind, folgt danach die zweite Zeitspalte mit den zugeordneten Wertespalten.

In Runtime können Sie die Reihenfolge der Wertespalten verändern, die einer Zeitspalte zugeordnet sind. Die Wertespalten sind nur bezüglich der fest verankerten Zeitspalte verschiebbar. Die Reihenfolge der Zeitspalten mit den zugeordneten Wertespalten legen Sie auf der Registerkarte "Zeitachsen" fest.

Voraussetzung

Folgende Schaltflächen sind projektiert:

- "Spalten wählen
- "Vorhergehende Spalte"
- "Nächste Spalte".

Vorgehen

Um die Reihenfolge der Spalten zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf .
   Der Dialog zum Ändern der Spaltenreihenfolge wird geöffnet.
2. Um eine Spalte auszublenden, deaktivieren Sie die Option vor dem Spaltennamen.
3. Um eine Wertespalte zu verschieben, selektieren Sie die Wertespalte:
   Die Funktion bezieht sich auf die Wertspalten, die einer Zeitspalte zugeordnet sind.
   – Um die Wertspalte hinter die nächste Wertspalte zu verschieben, klicken Sie auf .
   – lie Wertspalte vor die vorhergehende Wertspalte zu verschieben, klicken Sie auf .

Siehe auch

Online-Projektierung der Tabellenanzeige (Seite 775)

Starten und Stoppen der Aktualisierung

Einleitung

Sie setzen die Aktualisierung der im Control enthaltenen Daten mit den Schaltflächen "Start/ Stopp" fort.

Einige Schaltflächen stoppen die Aktualisierung automatisch, z. B. "Statistikbereich festlegen"

Ob die Aktualisierung gestoppt ist, erkennen Sie am Aussehen der Schaltfläche:

- : Die Aktualisierung ist gestoppt. Um die Aktualisierung fortzusetzen, klicken Sie auf die Schaltfläche.
- : Die Aktualisierung ist gestartet. Um die Aktualisierung zu stoppen, klicken Sie auf die Schaltfläche

Siehe auch

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Online-Projektierung der Tabellenanzeige (Seite 775)

Statistik von Runtimedaten erstellen

Einleitung

Für die Runtimedaten in der Kurven- oder Tabellenanzeige können Sie eine Auswertung der Prozessdaten erstellen. Die ausgewerteten Daten stellen Sie in der Wertetabelle dar.

Voraussetzung

- Eine Kurvenanzeige oder Tabellenanzeige ist projektiert
- Eine Wertetabelle ist projektiert und mit der Kurven- oder Tabellenanzeige verbunden
- Runtime ist aktiviert

Daten in einem Statistikbereich-Fenster anzeigen

Voraussetzung: In der Wertetabelle ist der Darstellungsmodus "Statistikbereich-Fenster" aktiviert.

Um Daten im Statistikbereich-Fenster der Wertetabelle anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Kurven- oder Tabellenanzeige auf .
   Die aktualisierte Darstellung wird gestoppt, die Prozessdaten werden weiter archiviert.
2. Wenn Sie Daten außerhalb des angezeigten Zeitbereichs auswerten wollen:
   – Klicken Sie auf
     Der Dialog "Zeit-Auswahl" wird geöffnet.
   – Geben Sie den gewünschten Zeitraum ein.
     Die Daten werden für den angegebenen Zeitraum dargestellt.
3. Wenn Sie eine Kurvenanzeige verwenden:
   – Klicken Sie auf
     In der Kurvenanzeige werden am rechten und linken Rand zwei senkrechte Linien eingeblendet.
   – Um den Statistikbereich zu definieren, verschieben Sie die beiden Linien an die gewünschte Position.
4. Wenn Sie eine Tabellenanzeige verwenden:
   – Selektieren Sie mit der Maus die Zeilen mit dem gewünschten Zeitraum in der Tabelle. Bei mehreren Spalten mit unterschiedlichen Zeitspalten können Sie verschiedene Zeiträume für den Statistikbereich wählen.
   – Klicken Sie in der Symbolleiste auf

Im Statistikbereich-Fenster werden die ausgewerteten Daten in den Spalten angezeigt, die Sie projektiert haben. Das folgende Bild zeigt die Darstellung im Statistikbereich-Fenster am Beispiel der Kurvenanzeige:

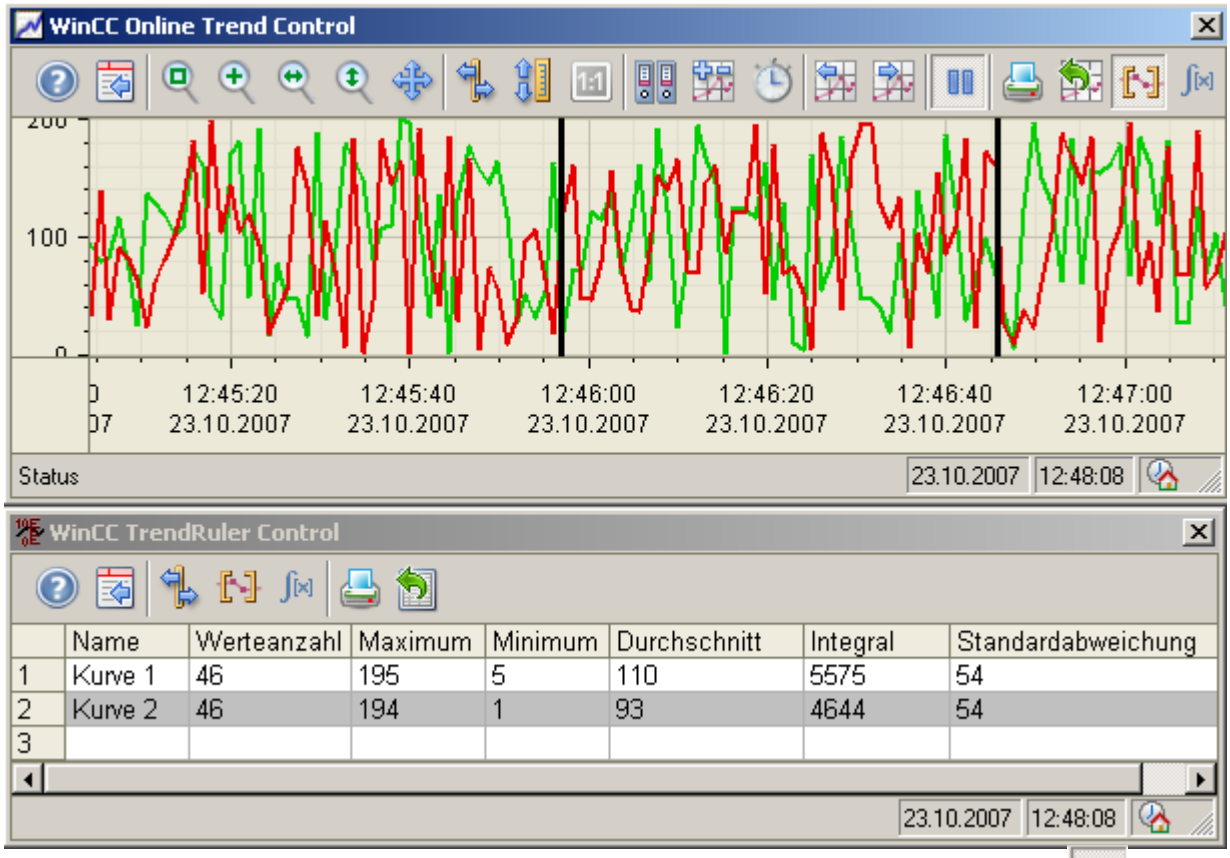

Um mit der Darstellung der Runtimedaten fortzufahren, klicken Sie auf .

Hinweis

Für eine weitergehende statistische Auswertung der Prozessdaten und eine Archivierung der Ergebnisse können Sie Skripte erstellen.

Siehe auch

Grundlagen zur Wertetabelle (Seite 754)

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Online-Projektierung der Tabellenanzeige (Seite 775)

Archivierte Werte darstellen

Einleitung

Mit den Schaltflächen in der Symbolleiste einer Kurven- oder Tabellenanzeige blättern Sie in den dargestellten Daten eines Archivs. Wenn Tastenkombinationen projektiert sind, können Sie alternativ auch diese zum Blättern verwenden.

Die Schaltflächen zum Blättern in Archiven stehen nur bei einer Datenversorgung mit Archivvariablen zur Verfügung.

In der Kurven- oder Tabellenanzeige werden die archivierten Werte einer Variablen innerhalb eines Zeitbereichs dargestellt.

Voraussetzung

- Zeitbereich ist projektiert

Schaltflächen für archivierte Werte

Um in archivierten Werten zu blättern, gehen Sie folgendermaßen vor:

1. Um den ersten Datensatz des Zeitbereichs anzuzeigen, klicken Sie auf .
2. Um den vorherigen Datensatz des Zeitbereichs anzuzeigen, klicken Sie auf .
3. Um den nächsten Datensatz des Zeitbereichs anzuzeigen, klicken Sie auf .
4. Um den letzten Datensatz des Zeitbereichs anzuzeigen, klicken Sie auf .

Siehe auch

Grundlagen zum Zeitbereich (Seite 724)

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Online-Projektierung der Tabellenanzeige (Seite 775)

Elemente der Statuszeile

Elemente der Statuszeile

Die Statuszeile der Kurven- oder Tabellenanzeige kann folgende Elemente enthalten:

Symbol	Name	Beschreibung
	Verbindungsstatus[1]	Keine fehlerhaften Datenverbindungen
		Fehlerhafte Datenverbindungen vorhanden
		Alle Datenverbindungen sind fehlerhaft
"Zeile 1"[2]	Selektierte Zeile	Zeigt die Nummer der selektierten Zeile an.
"Spalte 2"[2]	Selektierte Spalte	Zeigt die Nummer der selektierten Spalte an.
"23.02.2010"	Datum	Zeigt das Systemdatum.
"23:59:59"	Zeit	Zeigt die Systemzeit.
	Zeitbasis	Zeigt die Zeitbasis, die bei der Anzeige von Zeiten verwendet wird.

Symbol	Name	Beschreibung
[1]: Wenn Sie auf das Symbol "Verbindungsstatus" doppelklicken, wird das Fenster "Zustand der Datenanbindungen" geöffnet. In dem Fenster sind folgende Eigenschaften jeder Datenanbindung aufgelistet: • Name • Zustand • Variablenname		
[2]: Nur in der Tabellenanzeige		

Siehe auch

Online-Projektierung der f(t)-Kurvenanzeige (Seite 768)

Online-Projektierung der Tabellenanzeige (Seite 775)

Symbolleiste und Statuszeile konfigurieren (Seite 756)

Mit Meldungen arbeiten 3

3.1 Grundlagen

3.1.1 Das Meldesystem in WinCC

Einleitung

Das Meldesystem ermöglicht am Bediengerät die Anzeige von Betriebszuständen und Störungen, die in einer Anlage anstehen oder auftreten.

Eine Meldung hat z. B. folgende Inhalte:

Nr.	Uhrzeit	Datum	Meldetext	Zustand	Meldeklasse
5	12:50:24:590	24.02.2007	Kesseldruck liegt über der Obergrenze.	Gekommen Gegangen	Warnung: Farbe Rot

Das Meldesystem in WinCC

Das Meldesystem verarbeitet verschiedene Meldeverfahren. Die Meldeverfahren teilen sich auf in systemdefinierte Meldungen und benutzerdefinierte Meldungen:

- Benutzerdefinierte Meldungen dienen zur Überwachung des Anlagenprozesses.
- Systemdefinierte Meldungen dienen zur Überwachung des Bediengeräts.

Die erfassten Meldeereignisse werden am Bediengerät angezeigt. Gezielte Zugriffe auf die Meldungen und ergänzende Informationen zu einzelnen Meldungen gewährleisten eine zügige Störungslokalisierung und Störungsbehebung. Stillstandszeiten werden verringert oder ganz vermieden.

Das folgende Bild zeigt den Aufbau des Meldesystems:

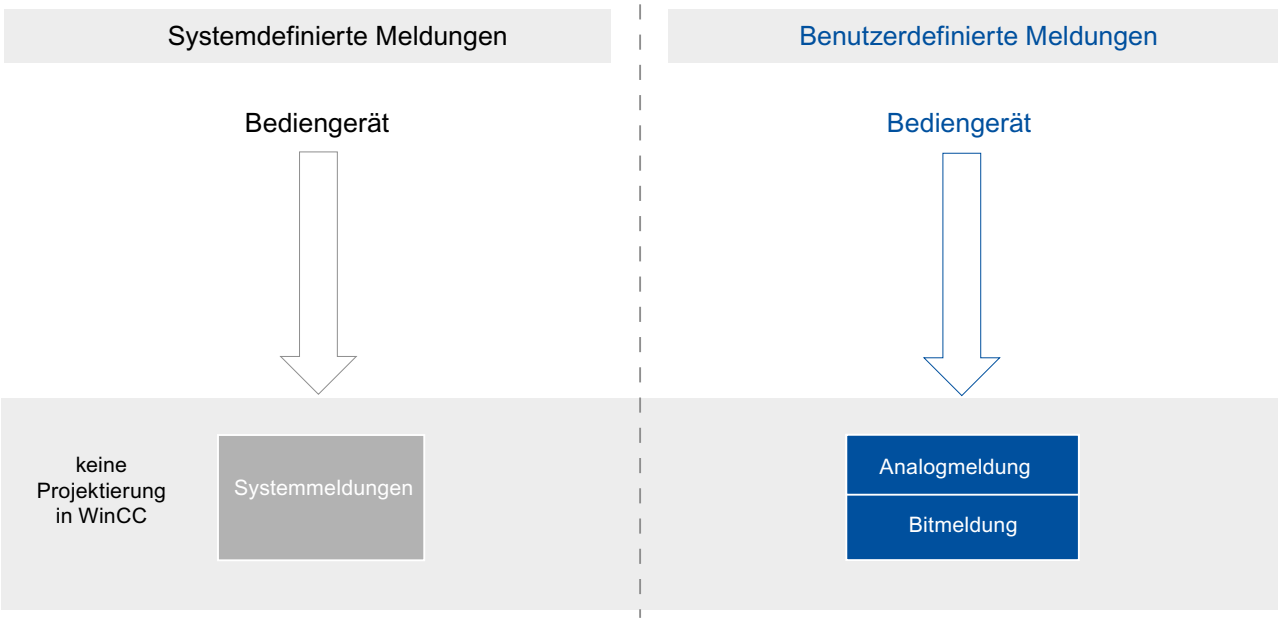

Siehe auch

Grundlagen zu Meldeklassen (Seite 799)

3.1.2 Das Meldesystem in WinCC

Einleitung

Das Meldesystem ermöglicht am Bediengerät die Anzeige und Aufzeichnung von Betriebszuständen und Störungen, die in einer Anlage anstehen oder auftreten.

Eine Meldung hat z. B. folgende Inhalte:

Nr.	Uhrzeit	Datum	Meldetext	Zustand	Meldeklasse
5	12:50:24:590	24.02.2007	Kesseldruck liegt über der Obergrenze.	Gekommen Gegangen	Warnung: Farbe Rot

Das Meldesystem in WinCC

Das Meldesystem verarbeitet verschiedene Meldeverfahren der Steuerung und des Bediengeräts. Die Meldeverfahren teilen sich auf in systemdefinierte Meldungen und benutzerdefinierte Meldungen:

- Benutzerdefinierte Meldungen dienen zur Überwachung der Anlage.
- Systemdefinierte Meldungen dienen zur Überwachung des Bediengeräts oder der Steuerung.

Die erfassten Meldeereignisse werden am Bediengerät angezeigt. Über das Meldesystem archivieren Sie Meldungen des laufenden Prozesses. Gezielte Zugriffe auf die Meldungen und ergänzende Informationen zu einzelnen Meldungen gewährleisten eine zügige Störungslokalisierung und Störungsbehebung. Stillstandszeiten werden verringert oder ganz vermieden.

Die Arten der systemdefinierten Steuerungsmeldungen sind abhängig von der verwendeten Steuerung.

Das folgende Bild zeigt den Aufbau des Meldesystems bei Kommunikation mit den SIMATIC Steuerungen S7-300/400:

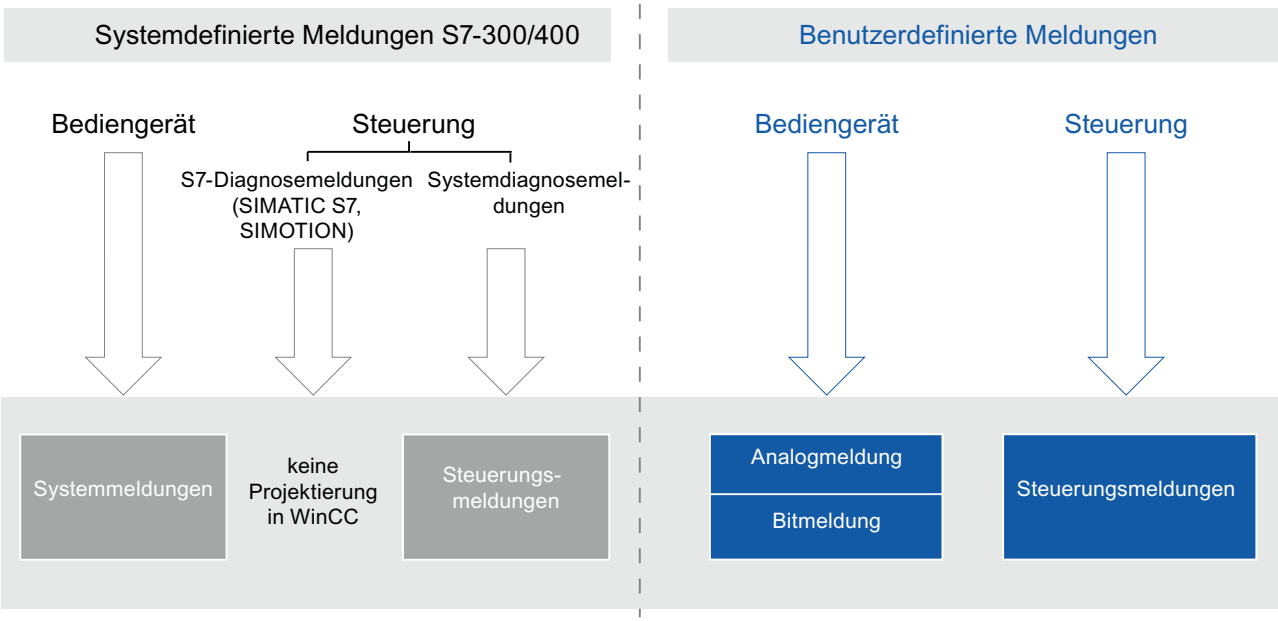

Das folgende Bild zeigt den Aufbau des Meldesystems bei Kommunikation mit den SIMATIC Steuerungen S7-1500:

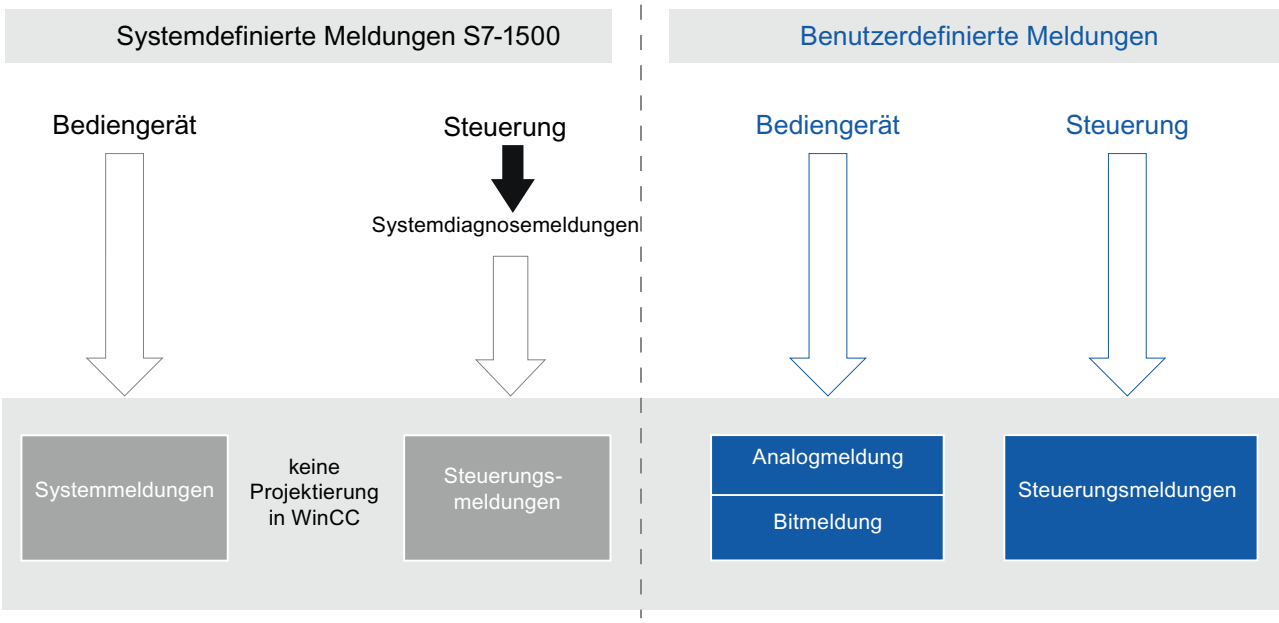

3.1.3 Das Meldesystem in WinCC

Einleitung

Das Meldesystem ermöglicht am Bediengerät die Anzeige und Aufzeichnung von Betriebszuständen und Störungen, die in einer Anlage anstehen oder auftreten.

Eine Meldung hat z. B. folgende Inhalte:

Nr.	Uhrzeit	Datum	Meldetext	Zustand	Meldeklasse
5	12:50:24:590	24.02.2007	Kesseldruck liegt über der Obergrenze.	Gekommen Gegangen	Warnung: Farbe Rot

Das Meldesystem in WinCC

Das Meldesystem verarbeitet verschiedene Meldeverfahren der Steuerung und des Bediengeräts. Die Meldeverfahren teilen sich auf in systemdefinierte Meldungen und benutzerdefinierte Meldungen:

- Benutzerdefinierte Meldungen dienen zur Überwachung der Anlage.
- Systemdefinierte Meldungen dienen zur Überwachung des Bediengeräts oder der Steuerung.

Die erfassten Meldeereignisse werden am Bediengerät angezeigt. Über das Meldesystem archivieren Sie Meldungen des laufenden Prozesses. Gezielte Zugriffe auf die Meldungen und ergänzende Informationen zu einzelnen Meldungen gewährleisten eine zügige

Störungslokalisierung und Störungsbehebung. Stillstandszeiten werden verringert oder ganz vermieden.

Die Arten der systemdefinierten Steuerungsmeldungen sind abhängig von der verwendeten Steuerung.

Das folgende Bild zeigt den Aufbau des Meldesystems bei Kommunikation mit den SIMATIC Steuerungen S7-300/400:

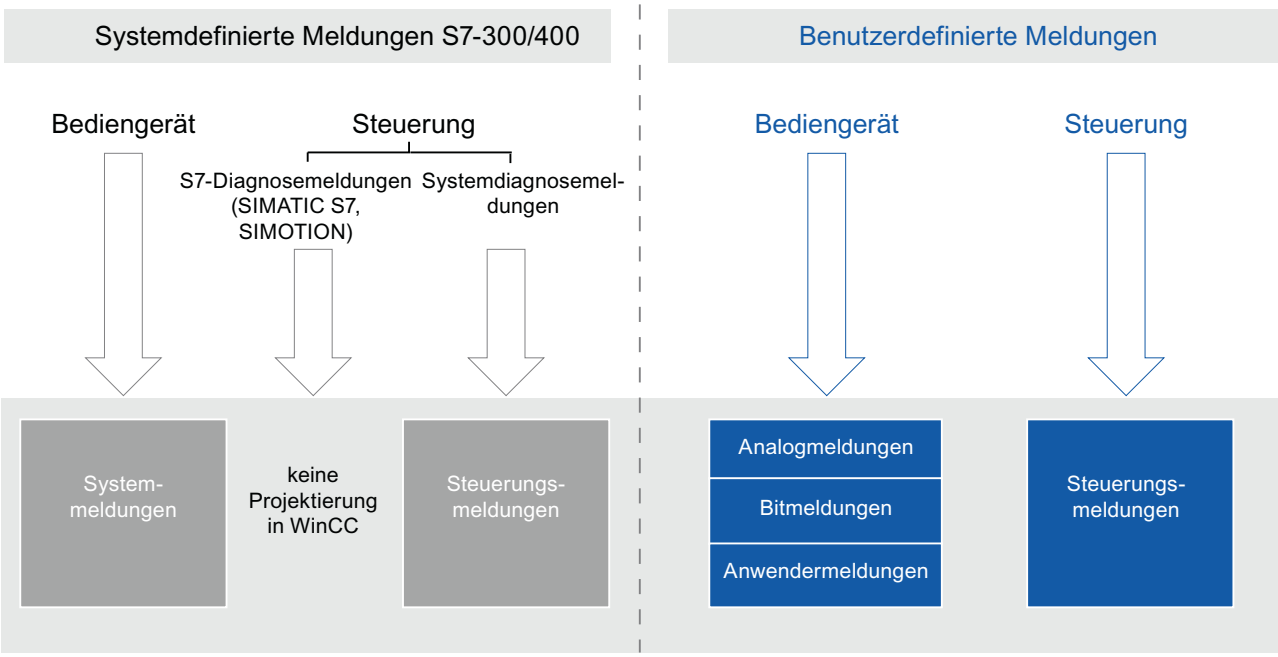

Das folgende Bild zeigt den Aufbau des Meldesystems bei Kommunikation mit den SIMATIC Steuerungen S7-1500:

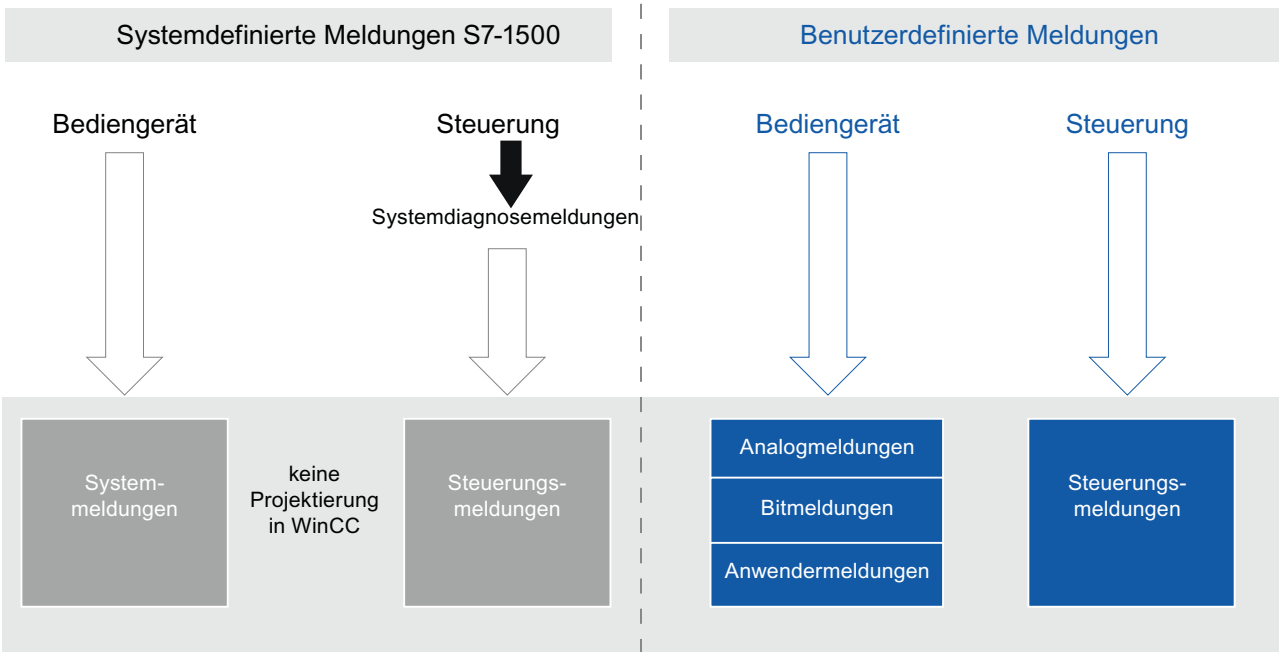

3.1.4 Meldeverfahren

3.1.4.1 Übersicht Meldeverfahren

Einleitung

Die Meldeverfahren dienen verschiedenen Zwecken zur Überwachung der Anlage. Die Meldungen der einzelnen Meldeverfahren werden auf unterschiedliche Arten projektiert und ausgelöst.

Sie projektieren Meldungen nach den einzelnen Meldeverfahren in der jeweiligen Registerkarte im Editor "HMI-Meldungen".

Meldeverfahren in WinCC

WinCC unterstützt folgende Meldeverfahren:

Benutzerdefinierte Meldungen

- **Analogmeldungen**
  - Analogmeldungen dienen zur Grenzwertüberwachung.
- **Bitmeldungen**
  - Bitmeldungen dienen zur Zustandsüberwachung.

Systemdefinierte Meldungen

- **Systemmeldungen**
  - Systemmeldungen gehören zum jeweiligen Bediengerät und werden in das Projekt importiert.
  - Systemmeldungen dienen zur Überwachung des Bediengeräts.

Siehe auch

Analogmeldungen (Seite 793)

3.1.4.2 Übersicht Meldeverfahren

Einleitung

Die Meldeverfahren dienen verschiedenen Zwecken zur Überwachung der Anlage. Die Meldungen der einzelnen Meldeverfahren werden auf unterschiedliche Arten projektiert und ausgelöst.

Sie projektieren Meldungen nach den einzelnen Meldeverfahren in der jeweiligen Registerkarte im Editor "HMI-Meldungen".

Meldeverfahren in WinCC

WinCC unterstützt folgende Meldeverfahren:

Benutzerdefinierte Meldungen

- **Analogmeldungen**
  - Analogmeldungen dienen zur Grenzwertüberwachung.
- **Bitmeldungen**
  - Bitmeldungen dienen zur Zustandsüberwachung.
- **Steuerungsmeldungen**
  - Steuerungsmeldungen projektieren Sie in STEP 7.
  - Steuerungsmeldungen bearbeiten Sie in WinCC weiter.

  Hinweis
  Geräteabhängigkeit

  Steuerungsmeldungen stehen nicht für alle Bediengeräte zur Verfügung.

Systemdefinierte Meldungen

- **Systemdefinierte Steuerungsmeldungen**
  - Systemdefinierte Steuerungsmeldungen bestehen aus Diagnosemeldungen (SIMATIC S7) und Systemfehlern (SFM).
  - Systemdefinierte Steuerungsmeldungen dienen zur Überwachung der Steuerung.

  Hinweis
  Geräteabhängigkeit

  Systemdefinierte Steuerungsmeldungen stehen nicht für alle Bediengeräte zur Verfügung.

- **Systemmeldungen**
  - Systemmeldungen gehören zum jeweiligen Bediengerät und werden in das Projekt importiert.
  - Systemmeldungen dienen zur Überwachung des Bediengeräts.

Siehe auch

Analogmeldungen (Seite 793)

Steuerungsmeldungen (Seite 794)

Systemdefinierte Steuerungsmeldungen (Seite 797)

3.1.4.3 Übersicht Meldeverfahren

Einleitung

Die Meldeverfahren dienen verschiedenen Zwecken zur Überwachung der Anlage. Die Meldungen der einzelnen Meldeverfahren werden auf unterschiedliche Arten projektiert und ausgelöst.

Sie projektieren Meldungen nach den einzelnen Meldeverfahren in der jeweiligen Registerkarte im Editor "HMI-Meldungen".

Meldeverfahren in WinCC

WinCC unterstützt folgende Meldeverfahren:

Benutzerdefinierte Meldungen

- **Analogmeldungen**
  - Analogmeldungen dienen zur Grenzwertüberwachung.
- **Bitmeldungen**
  - Bitmeldungen dienen zur Zustandsüberwachung.

- Anwendermeldungen
  - Anwendermeldungen dienen zur Bedienüberwachung.
  - Anwendermeldungen werden über die Meldenummer getriggert und werden in Skripten in Runtime nach Bedarf verwendet.
- Steuerungsmeldungen
  - Steuerungsmeldungen projektieren Sie in STEP 7.
  - Steuerungsmeldungen bearbeiten Sie in WinCC weiter.

  Hinweis
  Geräteabhängigkeit

  Steuerungsmeldungen und Anwendermeldungen stehen nicht für alle Bediengeräte zur Verfügung.

Systemdefinierte Meldungen

- Systemdefinierte Steuerungsmeldungen
  - Systemdefinierte Steuerungsmeldungen dienen zur Überwachung der Steuerung.
  - Zu den systemdefinierten Steuerungsmeldungen gehören auch Diagnosemeldungen (SIMATIC S7) und Systemfehler (SFM).

  Hinweis
  Geräteabhängigkeit

  Systemdefinierte Steuerungsmeldungen stehen nicht für alle Bediengeräte zur Verfügung.

- Systemmeldungen
  - Systemmeldungen gehören zum jeweiligen Bediengerät und werden in das Projekt importiert.
  - Systemmeldungen dienen zur Überwachung des Bediengeräts.

3.1.4.4 Benutzerdefinierte Meldeverfahren

Analogmeldungen

Beschreibung

Analogmeldungen zeigen Grenzwertverletzungen im laufenden Prozess an.

Beispiel

Die Drehzahl des Mixers in einer Fruchtsaftmischanlage darf nicht zu hoch und nicht zu niedrig sein. Um die Drehzahl des Mixers zu überwachen, projektieren Sie die entsprechenden Analogmeldungen. Wenn der Grenzwert für die Drehzahl des Mixers unter- oder überschritten wird, dann wird auf dem Bediengerät eine Meldung ausgegeben, die z. B. folgenden Meldetext enthält: "Drehzahl des Mixers zu niedrig".

Siehe auch

Analogmeldungen projektieren (Seite 827)

Bitmeldungen

Beschreibung

Bitmeldungen zeigen einen Zustand im laufenden Prozess an.

Beispiel

Eine Fruchtsaftmischanlage hat mehrere Tanks mit den Zutaten. Für das gewünschte Mischungsverhältnis von Wasser, Fruchtkonzentrat, Zucker und Aroma müssen sich die Ventile in den Zuleitungen im richtigen Moment schließen und öffnen. Dieser Vorgang soll überwacht werden.

Dazu projektieren Sie für alle Zustände der Ventile eine entsprechende Bitmeldung. Wenn sich ein Ventil der vier Tanks öffnet oder schließt, wird dann eine Meldung angezeigt, z. B. "Ventil Wasser geschlossen".

Der Benutzer überwacht so, ob die Anlage korrekt produziert.

Siehe auch

Bitmeldungen projektieren (Seite 830)

Steuerungsmeldungen

Beispiel einer Meldung

CPU Betriebsartenschalter auf Stop

Beschreibung

Eine benutzerdefinierte Steuerungsmeldung erstellt der Steuerungsprojekteur in STEP 7. In der Steuerungsmeldung sind die Zustandswerte der Steuerung abgebildet, z. B. Zeitstempel und Prozesswerte. Wenn Steuerungsmeldungen in STEP 7 projektiert sind, übernehmen Sie diese in den integrierten Betrieb von WinCC, sobald eine Verbindung zur Steuerung hergestellt ist.

Hinweis

Automatische Aktualisierung neuer oder geänderter Steuerungsmeldungen auf dem Bediengerät

Wenn Steuerungsmeldungen in STEP 7 projektiert sind und eine Verbindung zu einer S7-1500 Steuerung mit der Firmwareversion 2.0 (oder höher) hergestellt ist, werden die Steuerungsmeldungen an die Bediengeräte der Geräteversion V14 gesendet. Nach Änderungen der Steuerungsmeldungen muss die Bediengeräteprojektierung nicht mehr übertragen werden.

In STEP 7 ist die Steuerungsmeldung einer Meldeklasse zugeordnet. Diese Meldeklasse importieren Sie mit der Steuerungsmeldung als projektweite Meldeklasse.

Typen von Steuerungsmeldungen

SIMATIC S7 unterstützen verschiedene Typen von Steuerungsmeldungen. Abhängig von der verwendeten Runtime werden in WinCC unterschiedliche Typen von Steuerungsmeldungen unterstützt.

Steuerungsmeldungen für mehrere Bediengeräte

Wenn eine Steuerung mit mehreren Bediengeräten verbunden ist, weist der Projekteur in STEP 7 den Steuerungsmeldungen Anzeigeklassen zu. Die Anzeigeklassen regeln die Zuordnung zum Bediengerät. Sie aktivieren für Ihr jeweiliges Bediengerät die Anzeigeklassen, die darauf angezeigt werden sollen. Auf dem Bediengerät werden dann nur die Steuerungsmeldungen dieser Anzeigeklasse angezeigt. Maximal 17 Anzeigeklassen sind möglich.

Siehe auch

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 989)

Steuerungsmeldungen bearbeiten (Seite 837)

Anwendermeldungen

Beschreibung

Anwendermeldungen zeigen Bedienaktionen in Runtime an. Bei Bedarf projektieren Sie Anwendermeldungen auch innerhalb von Skripten in Runtime für benutzerdefinierte Anwendungen.

Beispiel

Auf dem Bediengerät einer Anlage wird eine Meldung der Meldeklasse "Errors" angezeigt. Der Bediener beseitigt den Fehler in der Anlage und quittiert dann die Meldung in der Meldeanzeige des Bediengeräts. Um zu überwachen, von wem und wann der Fehler behoben wurde, projektieren Sie eine Anwendermeldung an die entsprechende Schaltfläche der Meldeanzeige. Die Meldung enthält z. B. folgende Informationen:

- Art und Inhalt der quittierten Meldung
- Zeitpunkt der Quittierung
- Bediener
- Datum

3.1.4.5 Systemdefinierte Meldungen

Systemmeldungen

Beispiele für Meldungen

- "Online-Verbindung zur Steuerung wurde erfolgreich aufgebaut."

Beschreibung

Eine Systemmeldung zeigt den Status des Systems an und Kommunikationsfehler zwischen Bediengerät und System.

Unter "Runtime-Einstellungen > Meldungen" legen Sie fest, wie lange eine Systemmeldung am Bediengerät angezeigt wird.

Support

Sie finden in der Referenz eine Liste der möglichen Systemmeldungen mit Angabe der Ursache und Möglichkeiten zur Abhilfe. Wenn Sie sich wegen einer Systemmeldung des Bediengeräts an den Online Support wenden, benötigen Sie die Meldenummer und die vorhandenen Variablen der Systemmeldung.

Systemmeldungen

Beispiele für Meldungen

- "Der Zwischenpuffer für das Ausdrucken von Zeilen im Textmodus ist voll!"
- "Es ist kein Drucker installiert oder kein Standarddrucker eingerichtet."

Beschreibung

Eine Systemmeldung zeigt den Status des Systems und Kommunikationsfehler zwischen Bediengerät und System an.

Systemmeldungen sind abhängig vom Bediengerät.

Systemmeldungen werden bei Bedarf über eine Datei im Format *.xml in den Editor "HMI-Meldungen" eingelesen.

Support

Sie finden in der Referenz eine Liste der möglichen Systemmeldungen mit Angabe der Ursache und Möglichkeiten zur Abhilfe. Wenn Sie sich wegen einer Systemmeldung des Bediengeräts an den Online Support wenden, benötigen Sie die Meldenummer und die vorhandenen Variablen der Systemmeldung.

Siehe auch

Systemmeldungen (Seite 1044)

Systemmeldungen bearbeiten (Seite 836)

Systemdefinierte Steuerungsmeldungen

Beispiel einer Meldung

Ventil auf wegen Überdruck

Beschreibung

Systemdefinierte Steuerungsmeldungen werden mit STEP 7 installiert und stehen nur zur Verfügung, wenn WinCC im STEP 7-Umfeld betrieben wird.

WinCC verarbeitet systemdefinierte Steuerungsmeldungen vom Typ "SIMATIC S7-Diagnosemeldung".

S7-Diagnosemeldungen zeigen Zustände und Ereignisse der Steuerungen SIMATIC S7 an. Auf S7-Diagnosemeldungen haben Sie nur lesenden Zugriff:

- Das System ordnet die Meldungen der Meldeklasse "Diagnose" zu.
- Die bestehenden Eigenschaften der Meldung werden unveränderlich übernommen, z. B. die Nummer der Meldegruppe.
- S7-Diagnosemeldungen werden nicht quittiert oder protokolliert. Sie haben nur signalisierende Wirkungen.

Siehe auch

Systemdefinierte Steuerungsmeldungen aktivieren (Seite 839)

3.1.5 Meldezustände

Einleitung

Eine Meldung nimmt in Runtime verschiedene Meldezustände an. Anhand der Meldezustände analysiert und protokolliert der Benutzer den Verlauf des Prozesses.

Hinweis

Geräteabhängigkeit

Die Protokollierung und die Archivierung stehen nicht für alle Bediengeräte zur Verfügung.

Beschreibung

Jede Meldung befindet sich in einem Meldezustand. Die Meldezustände setzen sich aus folgenden Ereignissen zusammen:

- **Gekommen**
  Die Bedingung zum Auslösen einer Meldung trifft zu. Die Meldung wird angezeigt, z. B. "Kesseldruck zu hoch".
- **Gegangen**
  Die Bedingung zum Auslösen einer Meldung trifft nicht mehr zu. Die Meldung wird nicht mehr angezeigt, weil der Kessel entlüftet wurde.
- **Quittieren**
  Der Benutzer quittiert die Meldung.

Meldungen ohne Quittierpflicht

Die folgende Tabelle zeigt die Meldezustände für Meldungen ohne Quittierpflicht:

Zustand	Beschreibung
Gekommen	Die Bedingung einer Meldung trifft zu.
Gegangen	Die Bedingung einer Meldung trifft nicht mehr zu.

Meldungen mit Quittierpflicht

Die folgende Tabelle zeigt die Meldezustände für Meldungen mit Quittierpflicht:

Zustand	Beschreibung
Gekommen	Die Bedingung einer Meldung trifft zu
Gegangen, nicht quittiert	Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung nicht quittiert.
Gegangen, dann quittiert	Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung nach diesem Zeitpunkt quittiert.
Gekommen, quittiert	Die Bedingung einer Meldung trifft zu. Der Benutzer hat die Meldung quittiert.
Gegangen, zuvor quittiert	Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung quittiert, als die Bedingung noch zutraf.

Jedes Eintreten dieser Zustände wird bei Bedarf am Bediengerät angezeigt, archiviert und auf einem Drucker protokolliert.

Hinweis

Der Anzeigetext für die Zustände einer Meldung ist sprach- und projektierungsabhängig.

3.1.6 Meldeklassen

3.1.6.1 Grundlagen zu Meldeklassen

Einleitung

In einer Anlage treten viele Meldungen von unterschiedlicher Wichtigkeit auf. Um dem Benutzer klar ersichtlich zu machen, welche Meldungen am wichtigsten sind, ordnen Sie die Meldungen Ihres Projekts in Meldeklassen ein.

Beschreibung

Die Meldeklasse legt die Darstellung einer Meldung fest. Die Meldeklasse legt fest, ob und wie der Benutzer die Meldungen dieser Meldeklasse quittieren muss.

In WinCC wird eine neue Meldeklasse als quittierpflichtige Meldeklasse angelegt.

Hinweis

Die Auswahl der Darstellungsarten für Meldeklassen ist abhängig von den Möglichkeiten Ihres Bediengeräts.

Beispiele zur Anwendung von Meldeklassen

- Die Meldung "Drehzahl Lüfter 1 im oberen Toleranzbereich" hat die Meldeklasse "Warnings". Die Meldung wird mit weißem Hintergrund dargestellt. Die Meldung ist nicht quittierpflichtig.
- Die Meldung "Drehzahl Lüfter 2 hat oberen Warnbereich überschritten" ist der Meldeklasse "Errors" zugeordnet. Die Meldung wird mit rotem Hintergrund dargestellt und blinkt in Runtime mit einer hohen Frequenz. Die Meldung wird so lange angezeigt, bis der Benutzer die Meldung quittiert hat.

Meldeklassen verwenden

Um den Zustandsautomaten und die Darstellung der Meldungen für Ihr Projekt festzulegen, nutzen Sie folgende Meldeklassen:

- Vordefinierte Meldeklassen
  Vordefinierte Meldeklassen können Sie nicht löschen und nur eingeschränkt bearbeiten. Für jedes Bediengerät sind vordefinierte Meldeklassen unter "HMI-Meldungen > Meldeklassen" angelegt.
- Benutzerdefinierte Meldeklassen
  Sie legen Meldeklassen unter "HMI-Meldungen > Meldeklassen" neu an und projektieren die gewünschte Darstellung von Meldungen und ein Quittierkonzept für die Meldungen dieser Meldeklasse. Die mögliche Anzahl benutzerdefinierter Meldeklassen ist abhängig von der in Ihrem Projekt verwendeten Runtime

Hinweis

Meldeklassen aus STEP 7

Wenn ein Bediengerät mit der Steuerung verbunden ist, werden in WinCC auch Meldeklassen aus STEP 7 angezeigt.

Die Anzeige von systemdefinierten Steuerungsmeldungen (projektiert in STEP 7) wird von Basic Panels nicht unterstützt. Weitere Informationen zum Thema „Steuerungsmeldungen“ und „Meldesystem“ finden Sie unter Das Meldesystem in WinCC (Seite 785).

Siehe auch

Meldeklassen anlegen (Seite 818)

Meldeklassen anlegen (Seite 819)

Vordefinierte Meldeklassen (Seite 802)

Projektweite Meldeklassen verwenden (Seite 821)

Das Meldesystem in WinCC (Seite 785)

3.1.6.2 Grundlagen zu Meldeklassen

Einleitung

In einer Anlage treten viele Meldungen von unterschiedlicher Wichtigkeit auf. Um dem Benutzer klar ersichtlich zu machen, welche Meldungen am wichtigsten sind, ordnen Sie die Meldungen Ihres Projekts in Meldeklassen ein.

Beschreibung

Die Meldeklasse legt die Darstellung einer Meldung fest. Die Meldeklasse legt fest, ob und wie der Benutzer die Meldungen dieser Meldeklasse quittieren muss.

In WinCC wird eine neue Meldeklasse als quittierpflichtige Meldeklasse angelegt.

Hinweis

Die Auswahl der Darstellungsarten für Meldeklassen ist abhängig von den Möglichkeiten Ihres Bediengeräts.

Beispiele zur Anwendung von Meldeklassen

- Die Meldung "Drehzahl Lüfter 1 im oberen Toleranzbereich" hat die Meldeklasse "Warnings". Die Meldung wird mit weißem Hintergrund dargestellt. Die Meldung ist nicht quittierpflichtig.
- Die Meldung "Drehzahl Lüfter 2 hat oberen Warnbereich überschritten" ist der Meldeklasse "Errors" zugeordnet. Die Meldung wird mit rotem Hintergrund dargestellt und blinkt in Runtime mit einer hohen Frequenz. Die Meldung wird so lange angezeigt, bis der Benutzer die Meldung quittiert hat.

Meldeklassen verwenden

Um den Zustandsautomaten und die Darstellung der Meldungen für Ihr Projekt festzulegen, nutzen Sie folgende Meldeklassen:

- Vordefinierte Meldeklassen
  Vordefinierte Meldeklassen können Sie nicht löschen und nur eingeschränkt bearbeiten. Für jedes Bediengerät sind vordefinierte Meldeklassen unter "HMI-Meldungen > Meldeklassen" angelegt.
- Benutzerdefinierte Meldeklassen
  Sie legen Meldeklassen unter "HMI-Meldungen > Meldeklassen" neu an und projektieren die gewünschte Darstellung von Meldungen und ein Quittierkonzept für die Meldungen dieser Meldeklasse. Die mögliche Anzahl benutzerdefinierter Meldeklassen ist abhängig von der in Ihrem Projekt verwendeten Runtime.
- Projektweite Meldeklassen
  Projektweite Meldeklassen werden in der Projektnavigation unter "Gemeinsame Daten > Meldeklassen" angezeigt und können für die Meldungen eines Bediengeräts verwendet werden. Projektweite Meldeklassen stammen aus der Meldeprojektierung von STEP 7. Bei Bedarf legen Sie weitere projektweite Meldeklassen in WinCC an.

Siehe auch

Meldeklassen anlegen (Seite 818)

Meldeklassen anlegen (Seite 819)

Vordefinierte Meldeklassen (Seite 802)

Projektweite Meldeklassen verwenden (Seite 821)

3.1.6.3 Vordefinierte Meldeklassen

Vordefinierte Meldeklassen

Die folgenden Meldeklassen sind in WinCC für jedes Bediengerät bereits angelegt:

Meldeklassen für benutzerdefinierte Meldungen

- "Warnings"
  Die Meldeklasse "Warnings" ist dazu vorgesehen, irreguläre Zustände und Verläufe im Prozess anzuzeigen. Meldungen dieser Meldeklasse quittiert der Benutzer nicht.
- "Errors"
  Die Meldeklasse "Errors" ist dazu vorgesehen, kritische oder gefährliche Zustände oder Grenzwertverletzungen im Prozess anzuzeigen. Meldungen dieser Meldeklasse muss der Benutzer quittieren.

Meldeklassen für systemdefinierte Meldungen

- "System"
  Die Meldeklasse "System" enthält Meldungen, die Zustände des Bediengeräts und der Steuerungen anzeigen. Meldungen der Meldeklasse "System" gehören zu den Systemmeldungen.
- "Diagnosis Events"
  Die Meldeklasse "Diagnosis Events" enthält Meldungen, die Zustände und Ereignisse der Steuerungen SIMATIC S7 anzeigen. Meldungen dieser Meldeklasse quittiert der Benutzer nicht.

  Hinweis

  Geräteabhängigkeit

  Die Meldeklasse "Diagnosis Events" steht nicht für alle Bediengeräte zur Verfügung.

- "Safety Warnings"
  Die Meldeklasse "Safety Warnings" enthält Meldungen zum fehlersicheren Betrieb. Meldungen dieser Meldeklasse quittiert der Benutzer nicht. Meldungen der Meldeklasse "Safety Warnings" gehören zu den Systemmeldungen.

  Hinweis

  Geräteabhängigkeit

  Die Meldeklasse "Safety Warnings" steht nicht für alle Bediengeräte zur Verfügung.

Priorität der Meldeklassen auf Comfort Panels und in RT Advanced

Folgende Tabelle zeigt die Priorität der Meldeklassen:

Meldeklasse	Priorität
Errors	12
Warnings	4
System	16
Diagnosis Events	12
Meldung mit Einzelquittierung	12
Meldung ohne Quittierung	4

Wenn Sie in der Meldeanzeige nach Priorität filtern, erscheint die Meldung mit Priorität "0" an erster Stelle.

Hinweis

Wenn Sie für die benutzerdefinierte Meldeklassen die Priorität projektieren, wird die Priorität nur für Steuerungsmeldungen übernommen. Für die anderen Meldungen wird die Priorität vom Engineering System überschrieben.

Siehe auch

Grundlagen zu Meldeklassen (Seite 801)

Meldeklassen anlegen (Seite 819)

Meldeklassen anlegen (Seite 818)

Projektweite Meldeklassen verwenden (Seite 821)

3.1.7 Quittierung

3.1.7.1 Quittieren von Meldungen

Einleitung

Um sicherzustellen, dass eine Meldung vom Benutzer einer Anlage registriert wurde, projektieren Sie diese Meldung so, dass sie so lange angezeigt wird, bis sie der Benutzer quittiert hat. Meldungen, die kritische oder gefährliche Zustände im Prozess anzeigen, müssen quittierpflichtig sein.

Beschreibung

Die Quittierung einer Meldung ist ein Ereignis, das bei Bedarf archiviert und protokolliert wird. Durch die Quittierung einer Meldung ändert sich der Meldezustand einer Meldung von "Gekommen" auf "Quittiert". Mit der Quittierung einer Meldung bestätigt der Benutzer die Bearbeitung des Zustandes, der die Meldung ausgelöst hat.

Hinweis

Bediengeräteabhängigkeit

Die Protokollierung und die Archivierung stehen nicht für alle Bediengeräte zur Verfügung.

Quittierung einer Meldung triggern

Die Quittierung einer Meldung triggert der Bediener in Runtime auf verschiedene Arten:

- Quittieren durch den berechtigten Benutzer am Bediengerät
- Automatische Quittierung durch das System ohne Bedienaktion, z. B. über
  - Variablen
  - Steuerung
  - Systemfunktionen in Funktionslisten oder Skripten

    Hinweis

    Bediengeräteabhängigkeit

    Skripte stehen nicht für alle Bediengeräte zur Verfügung.

Zusammengehörige Meldungen gemeinsam quittieren

Um das Meldesystem in Runtime übersichtlich zu gestalten und einfach bedienbar zu machen, projektieren Sie bei Bedarf eine Meldegruppe. Der Benutzer quittiert mit einer Bedienaktion alle Meldungen, die der Meldegruppe angehören.

Quittieren durch die Steuerung

Bitmeldungen werden bei Bedarf automatisch von der Steuerung quittiert. Die Quittierung wird von einem Bit der "Quittiervariable Steuerung" ausgelöst. Bit und Variable legen Sie bei der Projektierung fest.

Quittieren einer Meldung am Bediengerät

Der Benutzer quittiert in Runtime eine Meldung abhängig von der Projektierung auf folgende Arten:

- Über die Quittiertaste <ACK> auf dem Bediengerät
- Über die Schaltfläche der Meldeanzeige
- Über projektierte Funktionstasten oder Schaltflächen in Bildern

Hinweis

Quittiertaste <ACK> am Bediengerät

Um kritische Meldungen nur von berechtigten Benutzern bearbeitet zu lassen, schützen Sie die Taste "ACK" der Bediengeräte und die Bedien- und Anzeigeobjekte der Meldungen. Verwenden Sie dazu einer entsprechende Bedienberechtigung.

Hinweis

Bediengeräteabhängigkeit

Die Quittiertaste <ACK> steht nicht auf allen Bediengeräten zur Verfügung.

Verhalten beim Quittieren von Meldungen einer Meldegruppe

- Quittieren der Meldung durch den Bediener
  Sie quittieren die Meldung einer Meldegruppe z. B. mit der Quittiertaste <ACK> oder einer Funktionstaste. Alle Meldungen der Meldegruppe werden quittiert.
- Quittieren der Meldung durch die Steuerung
  Die Meldung einer Meldegruppe wird durch das Setzen eines Bits in der Variablen quittiert. Nur die zu quittierende Meldung wird quittiert.

3.1.7.2 Quittierkonzept

Übersicht

Den Zustandsautomaten legen Sie für eine Meldeklasse fest. Meldungen, die dieser Meldeklasse zugeordnet sind, werden nach diesem Zustandsautomaten quittiert. In WinCC gibt es folgende Zustandsautomaten:

- Meldung ohne Quittierung
  Diese Meldung kommt und geht, ohne dass sie eine Quittierung erfordert. Das System zeigt keine Reaktion.
- Meldung mit Einfachquittierung
  Diese Meldung erfordert eine Quittierung, sobald das Ereignis, das die Meldung auslöst, eingetreten ist. Die Meldung steht an bis sie quittiert ist.

3.1.7.3 Quittierkonzept

Übersicht

Den Zustandsautomaten legen Sie für eine Meldeklasse fest. Meldungen, die dieser Meldeklasse zugeordnet sind, werden nach diesem Zustandsautomaten quittiert. In WinCC gibt es folgende Zustandsautomaten:

- Meldung ohne Quittierung
  Diese Meldung kommt und geht, ohne dass sie eine Quittierung erfordert. Das System zeigt keine Reaktion.
- Meldung mit Einfachquittierung
  Diese Meldung erfordert eine Quittierung, sobald das Ereignis, das die Meldung auslöst, eingetreten ist. Die Meldung steht an, bis sie quittiert ist.
- Meldung mit Zweifachquittierung
  Die Meldung erfordert eine Quittierung, sobald das Ereignis, das die Meldung auslöst, eingetreten ist und eine zweite Quittierung, wenn das Ereignis, das die Meldung auslöst, nicht mehr ansteht. Die Meldung steht jeweils an, bis sie quittiert ist.
- Meldung ohne Zustand "Gegangen" mit Quittierung
  Diese Meldung wird in der Meldeanzeige angezeigt, bis sie quittiert ist. Die Meldung erlischt dann aus der Meldeanzeige. Die Meldung wird nur archiviert.
- Meldung ohne Zustand "Gegangen" ohne Quittierung
  Diese Meldung wird angezeigt und erlischt, wenn das Ereignis, das die Meldung ausgelöst hat, nicht mehr ansteht. Die Meldung wird in die Meldeanzeige nicht eingetragen. Die Meldung wird nur archiviert.
- Erstwertmeldung mit Blinken und Einfachquittierung
  Erstwertmeldung ist eine Form der Meldeverarbeitung, bei der aus einer Anzahl von Meldungen diejenige hervorgehoben wird, deren Zustand sich seit der letzten Quittierung als erste geändert hat. Nur die erste Meldung dieser Meldeklasse blinkt bei der Anzeige im Meldefenster.

- Neuwertmeldung mit Blinken und Einfachquittierung
  Neuwertmeldung ist eine Form der Meldeverarbeitung, bei der aus einer Anzahl von Meldungen diejenigen hervorgehoben werden, deren Zustände sich seit der letzten Quittierung geändert haben.
- Neuwertmeldung mit Blinken und Zweifachquittierung
  Die Meldung blinkt, wenn sich der Wert geändert hat, der überwacht wird. Sie quittieren diese Meldung beim Kommen und beim Gehen.

Übersicht Zustandsautomaten für mehrere Meldungen

- Sammelquittierung/Einzelquittierung von Meldungen in der Meldean
  Die Meldeanzeige verfügt über die Schaltfläche "Sammelquittieren" . Diese Schaltfläche löst die Quittierung aller, in der Meldeanzeige anstehenden, sichtbaren und quittierpflichtigen Meldungen aus. Wenn für eine Meldung die Option "Einzelquittierung" aktiviert ist, kann der Benutzer nicht sammelquittieren, sondern muss sie einzeln über die Schaltfläche quittieren.
- Quittieren von Meldungen in Meldegruppen
  Wenn eine Meldegruppe eine Quittiervariable hat, quittiert der Benutzer in Runtime alle Meldungen dieser Meldegruppe gemeinsam über diese Variable. Zur Bedienung dieser Variable projektieren Sie die entsprechenden Schaltflächen oder Funktionstasten.

Zustandsautomat zentraler Melder

Sie haben folgende Möglichkeiten die Quittierung des zentralen Melders zu projektieren:

- Der Benutzer quittiert den zentralen Melder mit der Meldung, die den zentralen Melder ausgelöst hat.
- Der Benutzer quittiert den zentralen Melder über eine eigene Schaltfläche in der Meldeanzeige.
- Über eine festgelegte Variable wird der zentrale Melder angesteuert.

Not-Quittierung

In Ausnahmefällen haben Sie die Möglichkeit eine Meldung anhand ihrer Meldungsnummer zu quittieren. Dabei wird das Quittierbit auch dann an das AS gesandt, wenn die Meldung aktuell nicht ansteht.

Die Funktion sollte daher nur im äußersten Notfall verwendet werden.

3.1.8 Meldegruppen

Einleitung

In einer Anlage treten viele Meldungen aus unterschiedlichen Bereichen und Prozessen auf. In Meldegruppen fassen Sie zusammengehörige Meldungen zusammen.

Meldegruppen

Mithilfe der Meldegruppen beobachten Sie Teile der Anlage und quittieren die dazugehörigen Meldungen bei Bedarf gemeinsam.

Meldegruppen enthalten bei Bedarf Meldungen verschiedener Meldeklassen. Sie ordnen nur quittierpflichtige Meldungen einer Meldegruppe zu.

Meldegruppen verwenden

Die Zusammenstellung von Meldegruppen ist z. B. für folgende Meldungen sinnvoll:

- Meldungen, welche die gleiche Störung als Ursache haben.
- Gleichartige Meldungen
- Meldungen aus einer Maschineneinheit, z. B. "Störung Antrieb XY"
- Meldungen aus einem zusammengehörigen Teil des Prozesses, z. B. "Störung Kühlwasserversorgung"

Darstellung in Runtime

In Runtime wird in der Spalte "Meldegruppe" die Nummer der Meldegruppe angezeigt, zu der die Meldung gehört.

Siehe auch

Meldegruppen projektieren (Seite 824)

3.1.9 Meldegruppen

3.1.9.1 Grundlagen zu Meldegruppen

Einleitung

In einer Anlage treten viele Meldungen aus unterschiedlichen Bereichen und Prozessen auf. Um die Zustände der Anlagenbereiche am Bediengerät übersichtlich darzustellen und um das Meldesystem Ihres Projekts einfach bedienbar zu machen, verwenden Sie Meldegruppen.

Beschreibung

Eine Meldegruppe ist die Zusammenfassung von einzelnen Meldungen. Eine Meldegruppe verfügt über mehrere Variablen, die jeweils eine Eigenschaft der enthaltenen Meldungen ansprechen. Die Variablen und deren Bits legen Sie für eine Gruppe fest. Um eine Meldegruppe am Bediengerät anzuzeigen, projektieren Sie deren Visualisierung individuell über die Zustandsvariable der Meldegruppe.

Verwendung

Sie verwenden Meldegruppen z. B. zu folgenden Zwecken:

- Um Meldungen mit der gleichen Störung als Ursache zusammenzufassen
  z. B. "Stromausfall"
- Um gleichartige Meldungen zusammenzufassen
  z. B. "Sicherungsfall"
- Um Meldungen aus einer Maschineneinheit zu sortieren und zu beobachten
  z. B. "Störung Antrieb XY"
- Meldungen aus einem zusammengehörigen Teil des Prozesses
  z. B. "Störung Kühlwasserversorgung"
- Um alle Meldungen eines Anlagenbereichs gemeinsam zu bearbeiten, z. B.:
  - Quittieren von Meldungen
  - Sperren von Meldungen
  - Anzeige von Meldungen unterdrücken
- Um den Zustand von Anlagenbereichen zu visualisieren
- Um mit der Quittierung von Meldungen Prozesse der Anlage zu verbinden

Meldegruppen in WinCC

In WinCC arbeiten Sie mit zwei Arten von Meldegruppen:

- Benutzerdefinierte Meldegruppen
  Benutzerdefinierte Meldegruppen legen Sie nach Bedarf an. Sie enthalten quittierpflichtige Meldungen und bei Bedarf untergeordnete Meldegruppen. Benutzerdefinierte Meldegruppen bauen Sie bei Bedarf hierarchisch mit bis zu fünf Untergruppen auf.
- Meldegruppen aus Meldeklassen
  Für jede vordefinierte Meldeklasse ist bereits eine Meldegruppe angelegt. Wenn Sie eine neue Meldeklasse anlegen, wird eine Meldegruppe mit dem gleichen Namen erzeugt. Alle Meldungen dieser Meldeklasse sind auch in der Meldegruppe aus der Meldeklasse enthalten. Die Änderungen der Konfiguration der Meldeklassen wird in die Registerkarte "Meldegruppen" übertragen. Die Konfiguration der entsprechenden Meldegruppe wird aktualisiert. Sie bearbeiten für Meldegruppen aus Meldeklassen nur die Eigenschaften der Meldegruppe.

  Hinweis

  Eine Meldung ist immer in einer Meldegruppe aus Meldeklassen enthalten. Eine quittierpflichtige Meldung kann zusätzlich noch einer benutzerdefinierten Meldegruppe angehören.

Siehe auch

Meldegruppen projektieren (Seite 825)

3.1.9.2 Hierarchische Meldegruppen

Einleitung

WinCC bietet die Möglichkeit, Meldungen und benutzerdefinierte Meldegruppen hierarchisch zusammenzufassen. Sie bauen so eine Hierarchie von Meldungen auf, die es ermöglicht, die Zustände komplexer Anlagen übersichtlich und schnell zu erfassen und zu bearbeiten.

Sie erstellen diese Hierarchie aus Meldungen und Meldegruppen nur mit benutzerdefinierte Meldegruppen.

Aufbau hierarchischer Meldegruppen

Sie stellen nach Bedarf benutzerdefinierte Meldegruppen und einzelne Meldungen hierarchisch zusammen. Untergeordnete Meldegruppen fügen Sie auch nachträglich ein.

Das folgende Bild zeigt eine hierarchische Meldegruppe in WinCC:

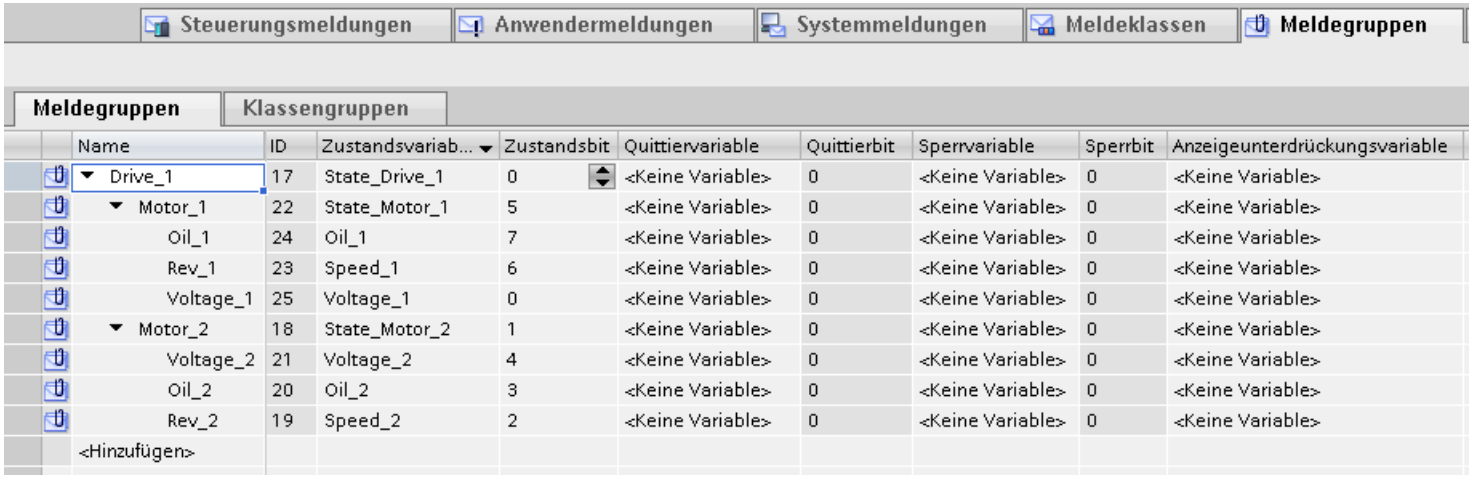

Name	ID	Zustandsvariab...	Zustandsbit	Quittiervariable	Quittierbit	Sperrvariable	Sperrbit	Anzeigeunterdrückungsvariable
Drive_1	17	State_Drive_1	0	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Motor_1	22	State_Motor_1	5	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Oil_1	24	Oil_1	7	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Rev_1	23	Speed_1	6	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Voltage_1	25	Voltage_1	0	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Motor_2	18	State_Motor_2	1	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Voltage_2	21	Voltage_2	4	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Oil_2	20	Oil_2	3	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
Rev_2	19	Speed_2	2	<Keine Variable>	0	<Keine Variable>	0	<Keine Variable>
<Hinzufügen>								

Hinweis

Wenn Sie eine Meldegruppe nachträglich einfügen, ist die Nummerierung nicht mehr fortlaufend.

ACHTUNG

Achten Sie beim Anlegen einer Hierarchie von Meldegruppen darauf, diese nicht rekursiv zu verwenden.

Hinweis

Wenn Sie eine Meldegruppe löschen, die weitere Meldegruppen enthält, werden alle untergeordneten Meldegruppen ebenfalls gelöscht.

Bestandteile übergeordneter Meldegruppen

Eine übergeordnete Meldegruppe enthält nach Bedarf folgende Elemente:

- Einzelne Meldungen
- Untergeordnete Meldegruppen

Anzeigeunterdrückungsvariable

Die Anzeigeunterdrückungsvariable vergeben Sie nur an Meldegruppen, die keine untergeordneten Meldegruppen enthalten und an benutzerdefinierten Meldegruppen.

Siehe auch

Meldegruppen projektieren (Seite 825)

3.1.9.3 Variablen einer Meldegruppe

Variablen einer Meldegruppe

Einer Meldegruppe ordnen Sie bei Bedarf folgender Variablen zu:

Zustandsvariable

Die Zustandsvariable speichert die Meldezustände einer Meldegruppe.

Die Zustandsvariable speichert die Quittieranforderung. Die Quittieranforderung steht in einer Meldegruppe an, wenn mindestens eine der enthaltenen Meldungen ansteht und quittierpflichtig ist.

Die Zustandsvariable dient zur Visualisierung der Meldezustände und der Quittieranforderung einer Meldegruppe.

Die Zustandsvariable kann von anderen WinCC-Komponenten abgefragt werden.

Sperrvariable

Die Sperrvariable speichert den gesperrten Zustand einer Meldegruppe.

Die Sperrvariable dient zur Auswertung des gesperrten Zustands einer Meldegruppe.

Quittiervariable

Die Quittiervariable speichert den Zustand "Quittiert" einer Meldegruppe.

Die Quittiervariable dient zum Quittieren einer Meldegruppe.

Die Quittiervariable kann von anderen WinCC -Komponenten abgefragt werden.

Hinweis

Das Quittierbit einer Quittiervariablen wird nicht automatisch rückgesetzt. Sie projektieren dafür eine Schaltfläche oder Funktionstaste. Der Benutzer setzt damit das Quittierbit der Quittiervariablen zurück.

Die Quittiervariable einer Meldegruppe zeigt die Quittierung über die Meldeanzeige nicht an.

Anzeigeunterdrückungsvariable

Die Anzeigeunterdrückungsvariable dient zum automatischen Unterdrücken der Anzeige aller Meldungen einer Meldegruppe bei einem bestimmten Anlagenzustand

Die Anzeigeunterdrückungsvariable liefert den aktuellen Wert des Anlagenzustandes als Wert zwischen 1 und 32. Alle Meldungen der Meldegruppe, bei denen in der Anzeigeunterdrückungsmaske das entsprechende Bit der Nummer 1 bis 32 gesetzt ist, werden unterdrückt.

Siehe auch

Meldegruppen projektieren (Seite 825)

Projektieren der Anzeigeunterdrückung von Meldungen (Seite 896)

Bitbelegung Sperrvariable (Seite 1068)

Bitbelegung Anzeigeunterdrückungsvariable (Seite 1069)

Bitbelegung Zustandsvariable (Seite 1063)

Bitbelegung Quittiervariable (Seite 1067)

3.1.10 Meldenummer

Meldenummernvergabe

Das System vergibt innerhalb eines Meldeverfahrens eindeutige Meldenummern.

Hinweis

Wenn Sie Meldenummern anpassen, achten Sie auf die Eindeutigkeit der Meldenummer innerhalb eines Meldeverfahrens.

3.1.11 Meldenummer

Meldenummernvergabe

Das System vergibt innerhalb eines Projekts eindeutige Meldenummern.

Hinweis

Wenn Sie Meldenummern anpassen, achten Sie auf die projektweite Eindeutigkeit der Meldenummer.

Die Meldenummer einer Systemmeldung ist vorrangig zu der einer benutzerdefinierten Meldung. Wenn Sie die Meldenummer einer Systemmeldung für eine benutzerdefinierte Meldung verwenden, ändern Sie die Meldenummer der benutzerdefinierten Meldung.

3.2 Arbeiten mit Meldungen

3.2.1 Bestandteile und Eigenschaften von Meldungen

Übersicht

In WinCC projektieren Sie die Bestandteile von Meldungen. Die grundlegenden Bestandteile von Meldungen zeigt die folgende Tabelle:

Meldeklasse	Meldenummer	Uhrzeit	Datum	Meldezustand	Meldetext	Meldegruppe	Tooltip	Triggervariable	Grenzwert
Warning	1	11:09:14	06.08.2007	KG	Maximale Drehzahl erreicht.	2	Diese Meldung ist....	speed_1	27
System	110001	11:25:58	06.08.2007	K	Wechsel in die Betriebsart "Online"	0	Diese Meldung ist....	PLC-Variable_1	–

Meldeklasse

Meldeklassen sind z. B. "Warnings" oder "Errors". Die Meldeklasse legt für eine Meldung Folgendes fest:

- Zustandsautomat
- Darstellung in Runtime (z. B. Farbe)
- Archivierung

Hinweis

Geräteabhängigkeit

Die Archivierung steht nicht für alle Bediengeräte zur Verfügung.

Meldenummer

Eine Meldung wird über eine eindeutige Meldenummer identifiziert. Die Meldenummer wird vom System vergeben. Die Meldenummer ändern Sie bei Bedarf, um z. B. Meldungen, die in Ihrem Projekt zusammengehören, mit einer fortlaufenden Meldenummer zu kennzeichnen.

Uhrzeit und Datum

Jede Meldung zeigt in einem Zeitstempel die Uhrzeit und Datum, zu der die Meldung ausgelöst wurde.

Meldezustand

Eine Meldung hat die Ereignisse "Gekommen", "Gegangen", "Quittieren". Für jedes Ereignis wird eine neue Meldung ausgegeben mit dem aktuellen Zustand der Meldung.

Meldetext

Der Meldetext beschreibt die Ursache der Meldung.

Der Meldetext kann Ausgabefelder für aktuelle Werte enthalten. Welche Werte Sie einfügen können, ist abhängig von der verwendeten Runtime. Der Wert wird zu dem Zeitpunkt festgehalten, zu dem sich der Meldezustand ändert.

Meldegruppe

Die Meldegruppe fasst einzelne Meldungen zusammen.

Tooltip

Sie projektieren bei Bedarf zu jeder Meldung einen eigenen Tooltip, den der Benutzer in Runtime anzeigen lassen kann.

Triggervariable

Jeder Meldung ist eine Variable als Auslöser zugewiesen. Die Meldung wird ausgegeben, wenn diese Triggervariable die festgelegte Bedingung erfüllt, z. B. wenn sie den Zustand ändert oder einen Grenzwert überschreitet.

Grenzwert

Analogmeldungen zeigen Grenzwertverletzungen an. Je nach Konfiguration gibt WinCC die Analogmeldung aus sobald die Triggervariable den konfigurierten Grenzwert z. B. überschreitet oder unterschreitet.

Siehe auch

Grundlagen zu Meldeklassen (Seite 801)

Meldezustände (Seite 798)

Meldegruppen (Seite 808)

Übersicht Meldeverfahren (Seite 791)

Übersicht Meldeverfahren (Seite 792)

3.2.2 Projektieren von Meldungen

3.2.2.1 Übersicht über das Projektieren von Meldungen

Arbeitsschritte zur Projektierung von Meldungen

Sie projektieren Meldungen in WinCC in folgenden Arbeitsschritten:

1. Meldeklassen bearbeiten und erstellen
   Sie legen über die Meldeklasse die Darstellung einer Meldung in Runtime und den Zustandsautomaten einer Meldung fest.
2. Im Editor "HMI-Variablen" Variablen anlegen
   - Sie projektieren die Variablen für Ihr Projekt.
   - Sie legen für die Variablen Bereichswerte an.
3. Im Editor "HMI-Meldungen" Meldungen anlegen
   - Sie legen benutzerdefinierte Meldungen an und ordnen Ihnen die zu überwachende Variable, Meldeklassen, Meldegruppen und weitere Eigenschaften zu.
   - Bei Bedarf ordnen Sie den Ereignissen von Meldungen Systemfunktionen oder Skripte zu.
4. Ausgabe von Meldungen projektieren
   Um die projektieren Meldungen auszugeben, projektieren Sie im Editor "Bilder" eine Meldeanzeige oder ein Meldefenster.

Weitergehende Projektierungsschritte

Abhängig von den Anforderungen Ihres Projekts sind weitere Arbeitsschritte zum Projektieren von Meldungen erforderlich:

1. Meldegruppen anlegen
   In Meldegruppen ordnen Sie die Meldungen Ihres Projekts nach Zusammengehörigkeit, z. B. nach Fehlerursache (z. B. Stromausfall) oder Fehlerquelle (z. B. Motor 1).
2. Loop-In-Alarm projektieren
   Um nach dem Eingang einer Meldung zu dem Bild zu wechseln, das Informationen zu dem aufgetretenen Fehler enthält, wird ein Loop-In-Alarm projektiert.

3.2.2.2 Übersicht über das Projektieren von Meldungen

Arbeitsschritte zur Projektierung von Meldungen

Sie projektieren Meldungen in WinCC in folgenden Arbeitsschritten:

1. Meldeklassen bearbeiten und erstellen
   Sie legen über die Meldeklasse die Darstellung einer Meldung in Runtime und den Zustandsautomaten einer Meldung fest.
2. Im Editor "HMI-Variablen" Variablen anlegen
   - Sie projektieren die Variablen für Ihr Projekt.
   - Sie legen für die Variablen Bereichswerte an.
1. Im Editor "HMI-Meldungen" Meldungen anlegen
   - Sie legen benutzerdefinierte Meldungen an und ordnen Ihnen die zu überwachende Variable, Meldeklassen, Meldegruppen und weitere Eigenschaften zu.
   - Bei Bedarf ordnen Sie den Ereignissen von Meldungen Systemfunktionen oder Skripte zu.
2. Ausgabe von Meldungen projektieren
   Um die projektieren Meldungen auszugeben, projektieren Sie im Editor "Bilder" eine Meldeanzeige oder ein Meldefenster.

Weitergehende Projektierungsschritte

Abhängig von den Anforderungen Ihres Projekts sind weitere Arbeitsschritte zum Projektieren von Meldungen erforderlich:

- Systemmeldungen aktivieren und bearbeiten
  Systemmeldungen importieren Sie bei Bedarf beim 1. Öffnen der Registerkarte "Systemmeldungen" im Editor "HMI-Meldungen". Nach dem Importieren können Sie Systemmeldungen bearbeiten.
- Steuerungsmeldungen aktivieren und bearbeiten
  Im integrierten Betrieb eines Projekts in STEP 7 legen Sie in den Einstellungen zu Meldungen fest, welche Steuerungsmeldungen auf Ihrem Bediengerät angezeigt werden.
- Meldegruppen anlegen
  In Meldegruppen ordnen Sie die Meldungen Ihres Projekts nach Zusammengehörigkeit, z. B. nach Fehlerursache (z. B. "Stromausfall") oder Fehlerquelle (z. B. "Motor 1").
- Loop-In-Alarm projektieren
  Um nach dem Eingang einer Meldung zum Bild zu wechseln, das Ihre Informationen zur markierten Meldung enthält, projektieren Sie einen Loop-In-Alarm.

3.2.2.3 Projektieren von Meldeklassen

Meldeklassen anlegen

Einleitung

Meldeklassen legen Sie im Editor "HMI-Meldungen" auf der Registerkarte "Meldeklassen" an. Standardmäßig sind für jedes Projekt einige Meldeklassen bereits angelegt. Nach Bedarf legen Sie weitere benutzerdefinierte Meldeklassen an. Insgesamt legen Sie maximal 32 Meldeklassen an.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Meldeklasse anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Meldeklassen".
   Die vordefinierten und bestehenden benutzerdefinierten Meldeklassen werden angezeigt.
   Das folgende Bild zeigt die tabellarische Anzeige vordefinierter Meldeklassen:

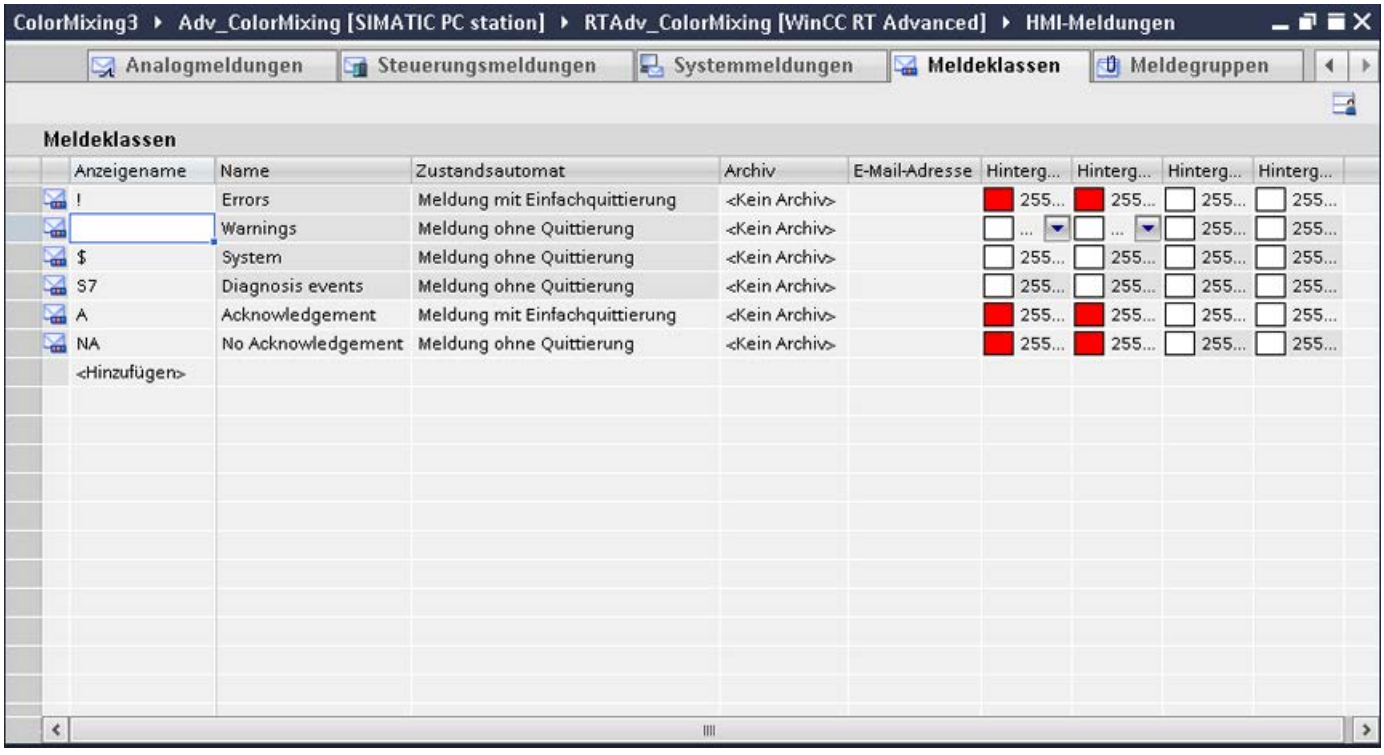

1. Doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Meldeklasse wird angelegt. Jeder neuen Meldeklasse wird automatisch eine feste ID zugewiesen.
   Die Eigenschaften der neuen Meldeklasse werden im Inspektorfenster angezeigt.
2. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse
   – Geben Sie einen "Namen" und den "Anzeigenamen" ein.
   – Abhängig vom Bediengerät aktivieren Sie bei Bedarf auch die Archivierung oder den automatischen Versand von E-Mails.
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung" den Zustandsautomaten der Meldeklasse fest.
4. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustandstexte" den voreingestellten Text.
   Mit diesem Text wird in Runtime der Zustand einer Meldung angezeigt.
5. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Farben" die voreingestellten Farben. Abhängig vom Bediengerät ändern Sie bei Bedarf auch das Blinkverhalten.

Mit diesen Einstellungen legen Sie fest, wie die Meldungen dieser Meldeklasse in Runtime dargestellt werden.

Hinweis

Um Meldeklassen in Runtime farbig darzustellen, muss die Option "Farben Meldeklassen verwenden" aktiviert sein. Aktivieren Sie dazu in der Projektnavigation "Runtime-Einstellungen > Meldungen > Allgemein > Farben Meldeklassen verwenden".

Siehe auch

Grundlagen zu Meldeklassen (Seite 801)

Vordefinierte Meldeklassen (Seite 802)

Meldeklassen anlegen

Einleitung

Meldeklassen legen Sie im Editor "HMI-Meldungen" auf der Registerkarte "Meldeklassen" an. Standardmäßig sind für jedes Projekt einige Meldeklassen bereits angelegt. Nach Bedarf legen Sie weitere benutzerdefinierte Meldeklassen an. Insgesamt sind maximal 256 Meldeklassen in einem Projekt möglich.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Meldeklasse anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Meldeklassen".
   Das folgende Bild zeigt die tabellarische Anzeige der vordefinierten Meldeklassen:

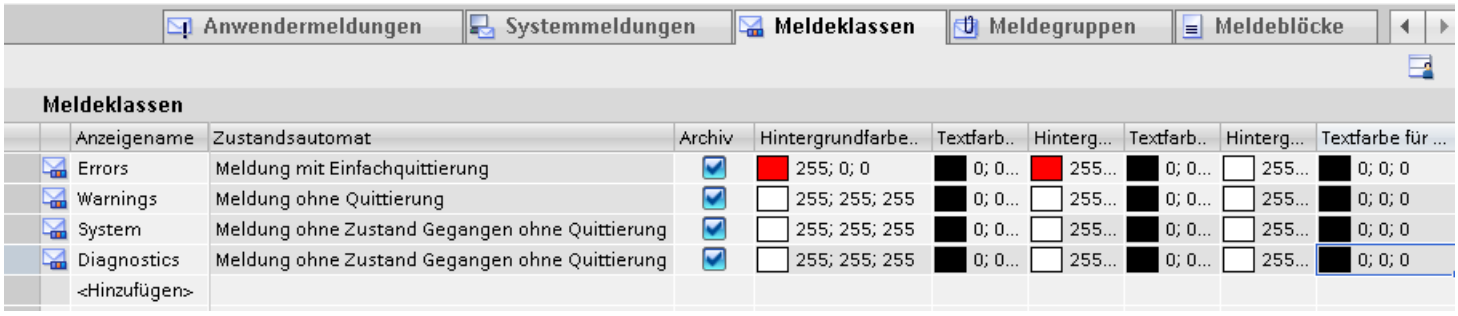

1. Doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Meldeklasse wird angelegt. Jeder neuen Meldeklasse wird automatisch eine feste ID zugewiesen.
   Die Eigenschaften der neuen Meldeklasse werden im Inspektorfenster angezeigt.
2. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse:
   – Geben Sie einen "Namen" und den "Anzeigenamen" ein.
   – Um die Meldungen dieser Meldeklasse zu archivieren, aktivieren Sie "Archiv".
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung" den Zustandsautomaten der Meldeklasse und des zentralen Melders fest.
4. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustand" bei Bedarf den voreingestellten Text.
   Mit diesem Text wird in Runtime der Zustand einer Meldung angezeigt.
5. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Farben" bei Bedarf die voreingestellten Hinter- und Vordergrundfarben.
   Mit diesen Einstellungen legen Sie fest, wie die Meldungen dieser Meldeklasse in Runtime dargestellt werden.

Weitere Einstellungen für Meldeklassen

Kommentare in Runtime

Der Benutzer gibt zu einer kommenden Meldung bei Bedarf in Runtime einen Kommentar ein. Über die Schaltflächen der Meldeanzeige blendet er diese nach Bedarf in der Anzeige "Langzeitarchiv" ein. In der Meldeklasse konfigurieren Sie diese Funktion. Der Benutzer ist im Systemblock "Benutzername" eingetragen.

Kommentare dem Benutzer eindeutig zuordnen

Um die Kommentare dem angemeldeten Benutzer zuzuordnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Meldeklasse.
2. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > Einstellungen > Kommentare Benutzer eindeutig zuordnen".
   Wenn noch kein Kommentar eingetragen ist, kann jeder Benutzer den ersten Kommentar eintragen.
   Wenn der erste Kommentar eingetragen ist, greifen alle anderen Benutzer auf diesen Kommentar nur noch lesend zu.

Kommentar der kommenden Meldung verwenden

Wenn sich der Zustand einer Meldung ändert, z. B. von "Gekommen" auf "Quittiert", wird die Meldung nochmals mit einem anderen Zustand ausgegeben. Der Benutzer ordnet aber nur einer kommenden Meldung Kommentare zu. Um Kommentare auch anzuzeigen, wenn die Meldung mit neuem Zustand ausgegeben wird, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Meldeklasse.
2. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > Einstellungen > Kommentare der kommenden Meldung verwenden".

Siehe auch

Quittieren einer Meldung projektieren (Seite 915)

Quittieren des zentralen Melders projektieren (Seite 918)

Grundlagen zu Meldeklassen (Seite 801)

Vordefinierte Meldeklassen (Seite 802)

Projektweite Meldeklassen verwenden

Einleitung

Projektweite Meldeklassen werden in der Projektnavigation unter "Gemeinsame Daten > Meldeklassen" angezeigt. Bei Bedarf legen Sie weitere projektweite Meldeklassen in WinCC an.

Steuerungsmeldungen sind projektweiten Meldeklassen zugeordnet. Wenn Sie mit Steuerungsmeldungen arbeiten, werden projektweite Meldeklassen automatisch in WinCC geladen.

Voraussetzungen

- Ein Projekt ist angelegt.

Projektweite Meldeklasse anlegen

Um eine projektweite Meldeklasse anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation "Gemeinsame Daten > Meldeklassen". Im Arbeitsbereich wird der Editor " Projektweite Meldeklassen" geöffnet.
2. Um eine projektweite Meldeklasse anzulegen, doppelklicken Sie in die erste leere Zeile des Tabelleneditors.
3. Legen Sie Name und Anzeigename der Meldeklasse fest.
4. Wenn Sie eine Meldeklasse mit Quittierpflicht erstellen wollen, aktivieren Sie "Mit Quittierung".

Meldungen einer projektweiten Meldeklasse zuordnen

Um eine Analog- oder Bitmeldung einer projektweiten Meldeklasse zuzuordnen, gehen Sie folgendermaßen vor:

1. Markieren Sie im Editor "HMI-Meldungen" die Meldung, die Sie der projektweiten Meldeklasse zuordnen wollen.
2. Klicken Sie im Inspektorfenster "Allgemein".
3. Klicken Sie in der Projektnavigation "Gemeinsame Daten > Meldeklassen".
4. Selektieren Sie in der Detailansicht die projektweite Meldeklasse.
5. Ziehen Sie die Meldeklasse per Drag&Drop in das Inspektorfenster der Meldung und dort in das Auswahlfeld "Meldeklasse" oder in die Spalte "Meldeklasse" im Arbeitsbereich.

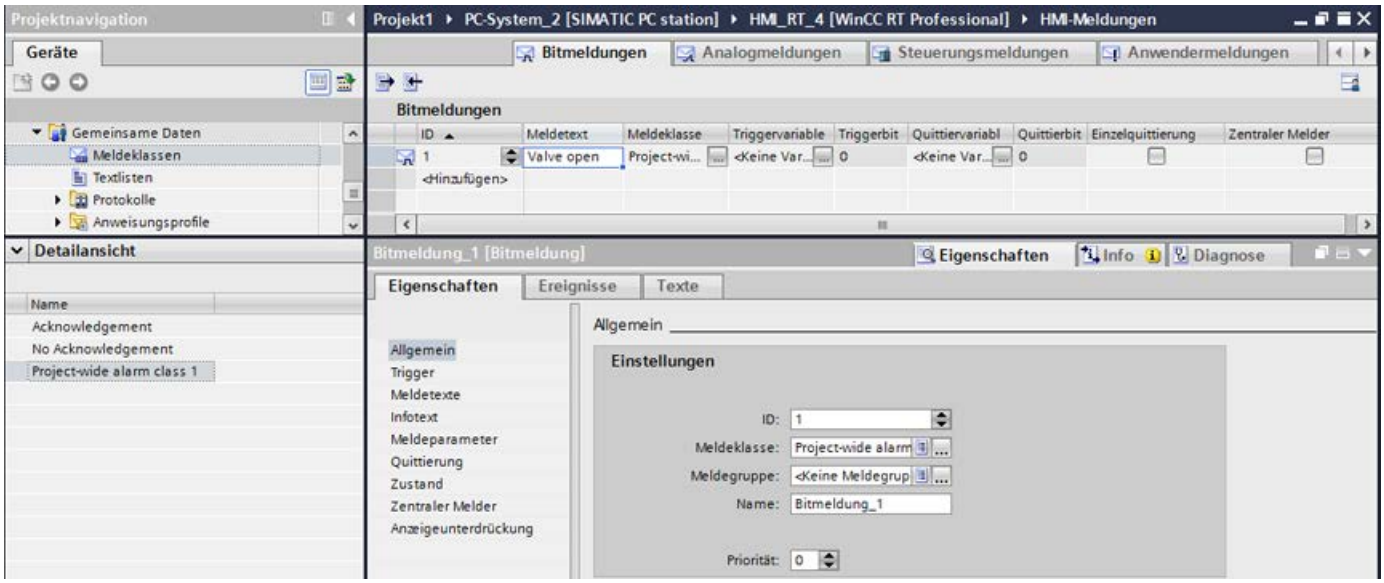

Das System legt eine neue Meldeklasse an, die mit der projektweiten Meldeklasse verknüpft ist. Im Inspektorfenster der Meldung wird unter "Meldeklasse" der neu generierte Name der projektweiten Meldeklasse angezeigt. Der Anzeigename der projektweiten Meldeklasse wird für die Meldeklasse in Ihrem Projekt übernommen.

6. Klicken Sie auf die Registerkarte "Meldeklasse".

7. Prüfen Sie, ob der Anzeigename der projektweiten Meldeklasse in WinCC angezeigt wird.
8. Ändern Sie bei Bedarf den Objektnamen der Meldeklasse.

Alternatives Vorgehen

Um eine Meldung einer projektweiten Meldeklasse zuzuordnen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Meldeklassen".
   Die Meldeklassen werden angezeigt.
2. Klicken Sie in der Projektnavigation "Gemeinsame Daten > Meldeklassen". Das folgende Bild zeigt die tabellarische Anzeige projektweiter Meldeklassen:

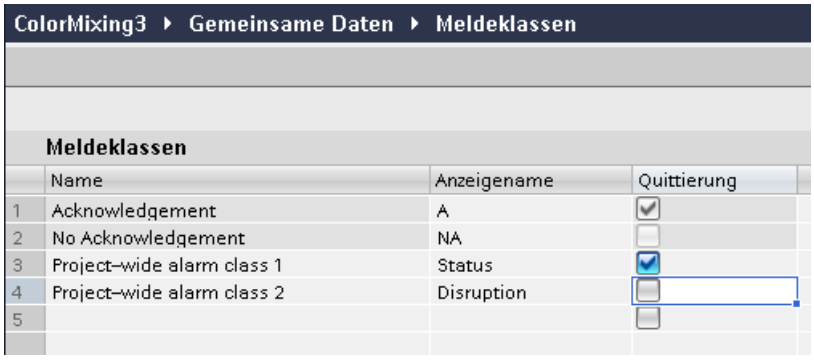

3. Markieren Sie in der Detailansicht die projektweite Meldeklasse.
4. Ziehen Sie die Meldeklasse per Drag&Drop in die Registerkarte "Meldeklassen" des Editors "HMI-Meldungen".
   Das System legt eine neue Meldeklasse an, die mit der projektweiten Meldeklasse verknüpft ist. Unter "Meldeklasse" wird für die Meldung der neu generierte Name der neuen Meldeklasse in Ihrem Projekt angezeigt.
5. Öffnen Sie die Registerkarte des Meldeverfahrens der gewünschten Meldung.
6. Markieren Sie die Meldung und wählen Sie im Inspektorfenster unter "Allgemein" die Meldeklasse, die mit der projektweiten Meldeklasse verknüpft ist.

Hinweis

Löschen einer projektweiten Meldeklasse

Wenn Sie eine projektweite Meldeklasse löschen, bleibt die verknüpfte Meldeklasse erhalten. Bei Bedarf verknüpfen Sie die Meldeklasse mit einer anderen projektweiten Meldeklasse oder verwenden sie als WinCC-Meldeklasse.

Siehe auch

Grundlagen zu Meldeklassen (Seite 801)

3.2.2.4 Projektieren von Meldegruppen

Meldegruppen projektieren

Einleitung

Im Editor "HMI-Meldungen" in der Registerkarte "Meldegruppen" legen Sie Meldegruppen an. Die Meldegruppe ist eine Zusammenfassung einzelner Meldungen. Sie ordnen Meldungen in einer Meldegruppe nach Zusammengehörigkeit, z. B. Fehlerursache oder Fehlerquelle. Wenn Sie in Runtime eine Meldung dieser Meldegruppe quittieren, werden alle Meldungen der Meldegruppe automatisch auch quittiert.

Voraussetzung

- Ein Projekt ist angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Anlegen einer neuen Meldegruppe

1. Klicken Sie auf die Registerkarte "Meldegruppen".
   Die angelegten Meldegruppen werden angezeigt.

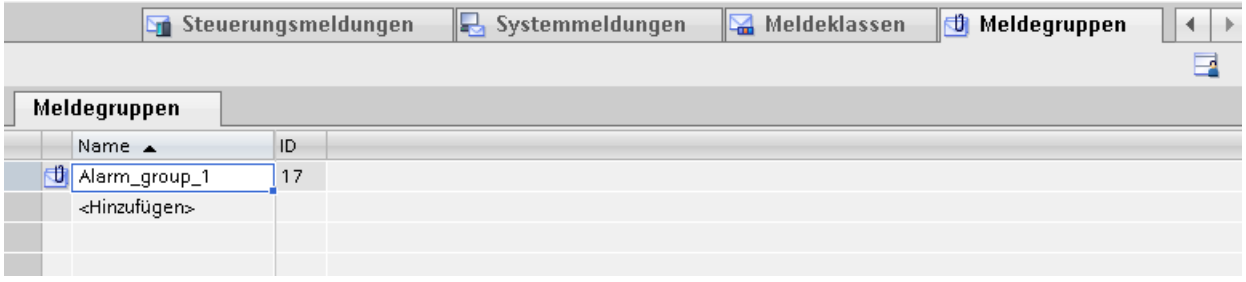

2. Doppelklicken Sie in der Tabelle im Arbeitsbereich in der ersten freie Zeile auf "<Hinzufügen>".
   Eine neue Meldegruppe wird angelegt.
3. Überschreiben Sie bei Bedarf den vorgeschlagenen "Namen".

Ergebnis

Eine Meldegruppe ist angelegt. Um Meldungen in Runtime gemeinsam zu quittieren, ordnen Sie zusammengehörige, quittierpflichtige Meldungen einer Meldegruppe zu.

Siehe auch

Meldegruppen (Seite 807)

Meldegruppen projektieren

Einleitung

Im Editor "HMI-Meldungen" in der Registerkarte "Meldegruppen" legen Sie benutzerdefinierte Meldegruppen an. Meldegruppen aus Meldeklassen sind bereits in einer eigenen Registerkarte angelegt. Sie quittieren in Runtime Meldungen einer Meldegruppe gemeinsam. Dazu projektieren Sie eine Schaltfläche zur Quittierung der Meldegruppe und eine weitere Schaltfläche zum Rücksetzen des Quittierbits der Quittiervariablen.

Voraussetzung

- Ein Projekt ist angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Anlegen einer neuen Meldegruppe

Um eine Meldegruppe anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Meldegruppen".
   Die angelegten Meldegruppen werden angezeigt.

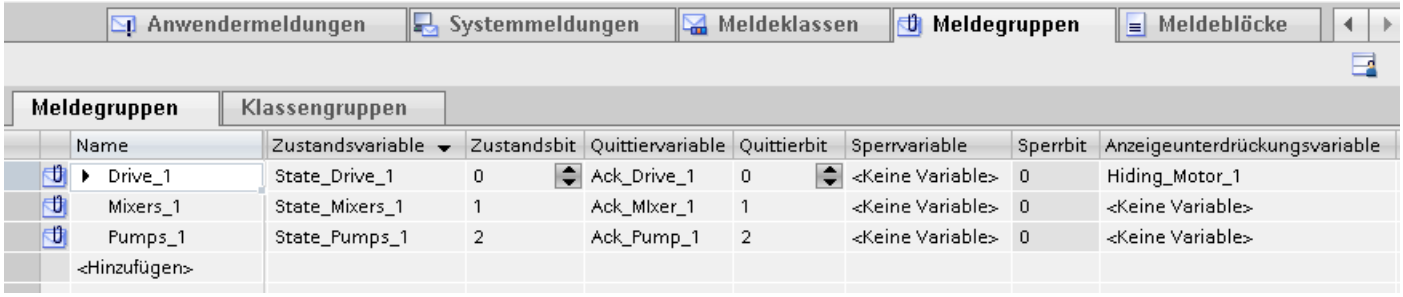

2. Doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Meldegruppe wird angelegt.
   Die Eigenschaften der neuen Meldegruppe werden im Inspektorfenster angezeigt.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" und legen Sie einen Namen für die Meldegruppe fest.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Variablen" und legen Sie bei Bedarf folgende Variablen und Bits fest:
   - Zustandsvariable, zur Visualisierung der Meldezustände der Meldegruppe
   - Quittiervariable, zur Quittierung einer Meldegruppe
   - Sperrvariable, zur Visualisierung des gesperrten Zustands einer Meldegruppe
   - Anzeigeunterdrückungsvariable, zum Filtern von Meldungen in Runtime nach Anlagenzustand

Anlegen einer untergeordneten Meldegruppe

Um eine untergeordnete Meldegruppe anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Meldegruppen".
2. Markieren Sie die Meldegruppen im Arbeitsbereich oder in der Detailansicht, die Sie als Untergruppe einfügen wollen.
3. Ziehen Sie die Meldegruppen per Drag&Drop auf eine andere Meldegruppe.
   Die selektierten Meldegruppen werden hierarchisch unterhalb dieser Meldegruppe eingeordnet.

Wenn Sie für eine Hierarchie aus Meldegruppen eine übergeordnete Meldegruppe anlegen wollen, verschieben Sie die ganze Hierarchie per Drag&Drop auf diese Meldegruppe.

Alternatives Vorgehen

1. Klicken Sie auf die Registerkarte "Meldegruppen".
2. Markieren Sie die Meldegruppen, zu der Sie eine Untergruppe anlegen wollen.
3. Öffnen Sie in der linken Tabellenzelle vor der markierten Zeile das Kontextmenü und wählen Sie "Untergeordnetes Element einfügen".
   Eine untergeordnete Meldegruppe wird angelegt.

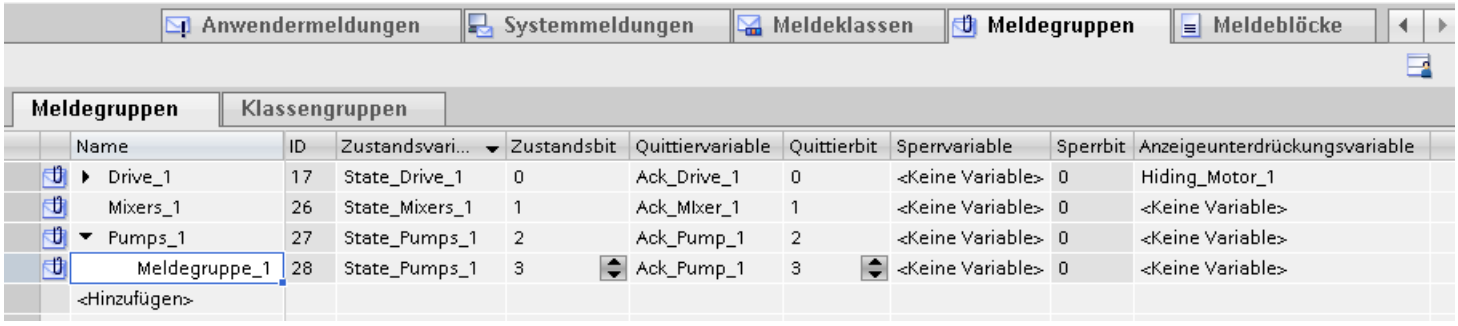

Siehe auch

Grundlagen zu Meldegruppen (Seite 808)

Hierarchische Meldegruppen (Seite 810)

Variablen einer Meldegruppe (Seite 811)

Projektieren der Anzeigeunterdrückung von Meldungen (Seite 896)

Bitbelegungen Variablen (Seite 1063)

3.2.2.5 Meldungen projektieren

Analogmeldungen projektieren

Einleitung

Analogmeldungen zeigen Grenzwertverletzungen an. Wenn z. B. die Drehzahl eines Motors unter einen bestimmten Wert sinkt, wird eine Analogmeldung getriggert.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die benötigten Meldeklassen und Meldegruppen sind angelegt.

Vorgehen

Um eine Analogmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Analogmeldungen".
2. Um eine neue Analogmeldung anzulegen, doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Analogmeldung wird angelegt.

3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
   - Geben Sie als Ereignistext den gewünschten Meldetext ein.
     Über das Kontextmenü formatieren Sie bei Bedarf den Text zeichenweise.
     Über das Kontextmenü fügen Sie bei Bedarf Ausgabefelder für HMI-Variablen oder Texte aus Textlisten ein.
   - Ändern Sie bei Bedarf die Meldenummer der Meldung.
   - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.

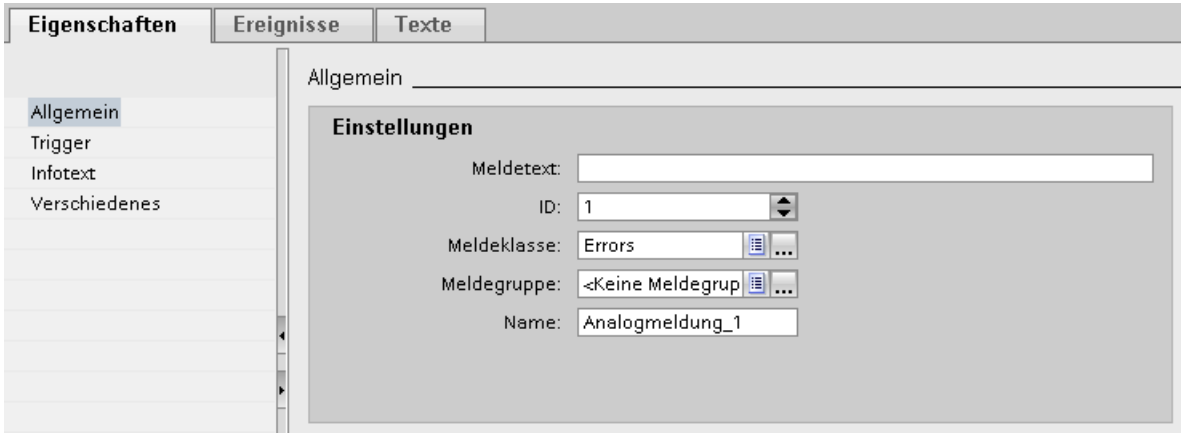

4. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Trigger > Einstellungen" die Variable, welche die Meldung auslöst.
   Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.

Grenzwerte für Analogmeldung projektieren

1. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger > Grenze > Wert" auf die Schaltfläche Const.
   - Um eine Konstante als Grenzwert zu verwenden, wählen Sie "Konstante".
     Geben Sie den gewünschten Grenzwert ein.
   - Um eine Variable als Grenzwert zu verwenden, wählen Sie "HMI_Variable".
     Die Schaltfläche ... wird eingeblendet. Wählen Sie damit die Variable.

   Hinweis

   Wenn in der Auswahlliste die gewünschte Variable noch nicht existiert, legen Sie diese in der Objektliste an und ändern Sie die Eigenschaften später.

2. Wählen Sie den Modus:
   - "Höher": Die Meldung wird bei Überschreitung des Grenzwerts getriggert.
   - "Niedriger": Die Meldung wird bei Unterschreitung des Grenzwerts getriggert.

Optionale Einstellungen für Analogmeldungen

Verzögerungszeit einstellen

Um die Verzögerungszeit einzustellen, gehen Sie folgendermaßen vor:

- Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger > Einstellungen > Verzögerung" eine Zeitspanne ein.
  Die Meldung wird nur dann ausgelöst, wenn die Triggerbedingung nach Ablauf der Verzögerungszeit noch ansteht.

Totzone einstellen

Hinweis

Wenn ein Prozesswert um den Grenzwert pendelt, wird für diese Störung die zugehörige Meldung mehrmals getriggert. Um dies zu verhindern, projektieren Sie eine Totzone oder eine Verzögerungszeit.

Um die Totzone einzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie unter "Eigenschaften > Eigenschaften > Trigger > Totzone > Modus" bei welcher Änderung des Meldezustandes die Totzone berücksichtigt wird.
2. Geben Sie unter "Wert" einen konstanten Wert ein.
3. Um festzulegen, dass der Wert der Totzone als Prozentangabe auf den Grenzwert bezogen wird, aktivieren Sie die Option "in %".

Automatisch protokollieren

Um die Meldungen laufend auf einem Drucker auszugeben:

- Wählen Sie unter "Eigenschaften > Eigenschaften > Verschiedenes > Protokoll" die Option "Aktivieren".

  Hinweis

  Geräteabhängigkeit

  Die Protokollierung steht nicht für alle Bediengeräte zur Verfügung.

Tooltip erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

- Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tooltip" und geben Sie den gewünschten Text ein.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Loop-In-Alarm".

3. Wählen Sie z. B. die Systemfunktion "AktiviereBild" oder eine andere Systemfunktion, die bei Loop-In-Alarm aufgerufen werden soll.
4. Wählen Sie die für die gewählte Systemfunktion notwendigen Parameter.
   Bei "AktiviereBild" ist es das Bild, das aufgerufen werden soll.

Siehe auch

Analogmeldungen (Seite 793)

Wert einer Variablen im Meldetext ausgeben (Seite 833)

Trigger zum Quittieren von Meldungen projektieren (Seite 913)

Protokollieren von Meldungen projektieren (Seite 909)

Meldetext formatieren (Seite 832)

Texte aus Textliste in Meldung ausgeben (Seite 835)

Bitmeldungen projektieren

Einleitung

Bitmeldungen zeigen Zustandsänderungen in einer Anlage an und werden von der Steuerung getriggert. Sie zeigen z. B. an, ob ein Ventil geöffnet oder geschlossen ist.

Im Folgenden ist die Projektierung im Editor "HMI-Meldungen" beschrieben. Alternativ projektieren Sie Bitmeldungen im Editor "HMI-Variablen".

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die benötigten Meldeklassen und Meldegruppen sind angelegt.

Vorgehen

Um eine Bitmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Registerkarte "Bitmeldungen".
2. Um eine neue Bitmeldung anzulegen, doppelklicken Sie im Arbeitsbereich auf "<Hinzufügen>".
   Eine neue Bitmeldung wird angelegt.

3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
   - Geben Sie als Ereignistext den gewünschten Meldetext ein.
     Nutzen Sie die Funktionen des Kontextmenüs, um den Text zeichenweise zu formatieren und um Ausgabefelder für HMI-Variablen oder Texte aus Textlisten einzufügen.
   - Ändern Sie bei Bedarf die Meldenummer der Meldung.
   - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger" die Variable und das Bit, durch das die Meldung getriggert wird. Beachten Sie dabei Folgendes:
   - Wenn Sie eine HMI Variable wählen, verwenden Sie die Datentypen "Int" oder "UInt".
   - Wenn Sie eine PLC Variable wählen, verwenden Sie die Datentypen "Int" oder "Word". Der Ein- und Ausgangsbereich einer PLC-Variablen ist als Trigger ungeeignet.
   - Verwenden Sie ein Bit einer Triggervariablen nur für eine Meldung.
   - Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.
   - Wenn Sie die Meldung über die Steuerung quittieren lassen, verwenden Sie diese Variable auch als PLC Quittiervariable.

Hinweis

Beachten Sie bei der Angabe des Bits die Zählweise der Bits in der verwendeten Steuerung. Weitere Hinweise dazu finden Sie in der Online-Hilfe zu der jeweiligen Steuerung im Abschnitt "Kommunikation".

Hinweis

Wenn in der Auswahl das gewünschte Objekt noch nicht existiert, legen Sie es direkt in der Objektliste an und ändern dessen Eigenschaften später.

Zustandsabhängige Meldetexte

Um abhängig vom Zustand der Meldung einen anderen Text anzuzeigen, binden Sie in den Meldetext eine Textliste ein. Die Textliste steuern Sie über eine Variable.

Weitere Einstellungen für Bitmeldungen

Tooltip erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

- Geben Sie unter "Eigenschaften > Eigenschaften > Tooltip" den gewünschten Text ein.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

Siehe auch

Bitmeldungen (Seite 794)

Trigger zum Quittieren von Meldungen projektieren (Seite 913)

Protokollieren von Meldungen projektieren (Seite 909)

Texte aus Textliste in Meldung ausgeben (Seite 835)

Meldetext formatieren (Seite 832)

Meldetext formatieren

Einleitung

Um bestimmte Ereignisse oder Vorgänge besonders zu betonen, können Sie Meldetexte formatieren oder Sonderzeichen in die Meldetexte einfügen.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Eine Meldung ist angelegt.

Meldetexte formatieren

Um einen Meldetext zu formatieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung, die Sie bearbeiten wollen.
2. Markieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Meldetext" die Zeichen, die Sie formatieren wollen.
3. Wählen Sie im Kontextmenü die gewünschte Formatierung, z. B. "Unterstrichen" oder "Großbuchstaben".
   Die ausgewählten Zeichen werden in Runtime in der gewählten Formatierung dargestellt.

Hinweis

Es ist nicht möglich Variablen und Textlisteneinträge unterstrichen darzustellen.

Hinweis

Darstellung von Sonderzeichen in Meldetexten

Im Engineering System wird für die Projektierung von Meldetexten ein fester Zeichensatz verwendet. Dieser Zeichensatz bietet die Möglichkeit, verschiedenste Sonderzeichen in den Meldetexten zu verwenden.

Für die Darstellung in Runtime werden sprachspezifische Schriftarten verwendet, z. B. MS PGothic, SimSun. Die in Runtime verwendeten Schriftarten unterstützen nicht alle Sonderzeichen. Daher werden einige Sonderzeichen in Runtime nicht angezeigt.

Formatierungen entfernen

Um alle Formatierungen eines Texts zu entfernen, gehen Sie folgendermaßen vor:

1. Markieren Sie im Inspektorfenster im Meldetext die Zeichen, deren Formatierung Sie entfernen wollen.
2. Wählen Sie im Kontextmenü "Formatierungszeichen löschen".
   Die ausgewählten Zeichen werden in Runtime unformatiert dargestellt.

Siehe auch

Übersicht über das Projektieren von Meldungen (Seite 816)

Analogmeldungen projektieren (Seite 827)

Bitmeldungen projektieren (Seite 830)

Wert einer Variablen im Meldetext ausgeben (Seite 833)

Texte aus Textliste in Meldung ausgeben (Seite 835)

Wert einer Variablen im Meldetext ausgeben

Einleitung

In WinCC fügen Sie in den Meldetext Ausgabefelder ein, die den Inhalt von Variablen anzeigen.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Meldung ist markiert.

Wert einer Variablen im Meldetext ausgeben

Um ein Ausgabefeld für den Wert einer Variablen in den Meldetext einzufügen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Ereignistext an der gewünschten Stelle.
2. Wählen Sie im Kontextmenü "Variablenausgabefeld einfügen".
3. Öffnen Sie unter "Variable" die Objektliste und wählen Sie die gewünschte Variable aus. Bei Bedarf können Sie die Variable in der Objektliste anlegen.
4. Legen Sie unter "Format" die Länge des Ausgabefeldes und das Format fest, in dem der Wert der Variable im Meldetext ausgegeben wird.
   Projektieren Sie das Ausgabefeld in ausreichender Größe. Andernfalls wird der Inhalt der Variablen in der Meldung nicht vollständig ausgegeben.
5. Um die Eingaben zu speichern, klicken Sie auf das Symbol .

WinCC fügt in den Meldetext einen Platzhalter für das Ausgabefeld ein:
"<tag: n, [Variablenname]>" wobei n = Textlänge in Zeichen.

Eigenschaften eines Ausgabefelds ändern

Um die Eigenschaften eines Ausgabefelds zu ändern, gehen Sie folgendermaßen vor:

- Doppelklicken Sie im Meldetext das Ausgabefeld und ändern Sie die Einstellungen wie gewünscht.

Ausgabefeld aus dem Meldetext löschen

Um ein Ausgabefeld aus dem Meldetext zu löschen, gehen Sie folgendermaßen vor:

- Markieren Sie im Meldetext das Ausgabefeld und wählen Sie im Kontextmenü "Löschen".

Hinweis

Die Reihenfolge der Ausgabefelder für Variablen im Meldetext ist sprachabhängig. Beim Archivieren von Meldungen in ein CSV-Archiv wird die Reihenfolge der Runtime-Sprache verwendet.

Die Änderung der Variable eines Ausgabefeldes in einer Sprache führt dazu, dass in allen anderen Sprachen das geänderte Ausgabefeld am Meldetextende steht. Dadurch ändert sich die Reihenfolge der Ausgabefelder im Archiv.

Geräteabhängigkeit

Die Archivierung steht nicht für alle Bediengeräte zur Verfügung.

Siehe auch

Analogmeldungen projektieren (Seite 827)

Meldetext formatieren (Seite 832)

Texte aus Textliste in Meldung ausgeben (Seite 835)

Texte aus Textliste in Meldung ausgeben

Einleitung

Mit Textlisten dynamisieren Sie Meldetexte. Sie projektieren z. B. die Ausgabe beider Zustände einer Bitmeldung in einem Meldetext.

Dazu fügen Sie ein entsprechendes Ausgabefeld in den Meldetext ein und legen die Variable fest, die den Suchschlüssel für die Textliste liefert.

Voraussetzung

- Eine Textliste ist angelegt
- Der Editor "HMI-Meldungen" ist geöffnet
- Die Meldung ist markiert

Wert aus einer Textliste im Meldetext ausgeben

Um ein Ausgabefeld für einen Wert aus einer Textliste in den Meldetext einzufügen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Meldetext an der gewünschten Stelle.
2. Wählen Sie im Kontextmenü "Textlistenausgabefeld eingeben".
3. Wählen Sie unter "Textliste" die gewünschte Textliste aus.
4. Öffnen Sie unter "Variable" die Objektliste und wählen Sie die Variable aus, die den Suchschlüssel für die Textliste liefert.
   Bei Bedarf legen Sie die Variable in der Objektliste an.
5. Legen Sie unter "Format" die Länge des Ausgabefeldes fest.
   Projektieren Sie das Ausgabefeld in ausreichender Größe, damit der Text aus der Textliste vollständig in der Meldung dargestellt wird.
6. Um Ihre Eingaben zu speichern, klicken Sie auf das Symbol .
   WinCC fügt in den Meldetext einen Platzhalter für das Ausgabefeld ein:
   "<textlist: n, [Name Textliste]>" wobei n = Textlänge in Zeichen.

Eigenschaften eines Ausgabefeldes ändern

Um die Eigenschaften eines Ausgabefeldes zu ändern, gehen Sie folgendermaßen vor:

- Doppelklicken Sie im Meldetext das Ausgabefeld und ändern Sie die Einstellungen wie gewünscht.

Textliste aus dem Meldetext löschen

Um ein Ausgabefeld aus dem Meldetext zu löschen, gehen Sie folgendermaßen vor:

- Markieren Sie im Meldetext das Ausgabefeld und wählen Sie im Kontextmenü "Löschen".

Siehe auch

Meldetext formatieren (Seite 832)

Analogmeldungen projektieren (Seite 827)

Bitmeldungen projektieren (Seite 830)

Wert einer Variablen im Meldetext ausgeben (Seite 833)

Systemmeldungen bearbeiten

Systemmeldungen bearbeiten

Wenn Sie im Editor "HMI-Meldungen" die Registerkarte "Systemmeldungen" öffnen, fordert WinCC Sie auf, die Systemmeldungen zu importieren oder zu aktualisieren. Sie importieren die Systemmeldungen und legen die Anzeigedauer fest. Weder löschen Sie Systemmeldungen noch legen Sie welche neu an. Sie können bei Systemmeldungen nur den Meldetext bearbeiten.

Voraussetzung

- Die Systemmeldungen sind importiert.

Anzeigedauer festlegen

Um die Anzeigedauer festzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Geben Sie unter "Meldungen > Systemmeldungen > Anzeigedauer" einen Wert ein. Dieser Wert legt fest, wie viele Sekunden Systemmeldungen an diesem Bediengerät angezeigt werden. Wenn die Systemmeldungen dauerhaft angezeigt werden sollen, geben Sie den Wert "0" ein.

Meldetext ändern

Um den Meldetext einer Systemmeldungen zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "HMI-Meldungen" und klicken Sie auf die Registerkarte "Systemmeldungen".
2. Markieren Sie die Systemmeldung, die Sie bearbeiten wollen.
1. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" den Meldetext.

Hinweis

Wenn Sie einen Meldetext einer Systemmeldung ändern, der Platzhalter enthält, lassen Sie die Anzahl der Platzhalter unverändert. Ein Platzhalter ist z. B. eine "%1 ".

Ereignisgesteuerte Aufgaben projektieren

Abhängig von Bediengerät gibt es für bestimmte Systemmeldungen das Ereignis "Gekommen". Um ereignisgesteuerte Aufgaben zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Systemmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Gekommen" und projektieren Sie eine Funktionsliste.

Siehe auch

Übersicht über das Projektieren von Meldungen (Seite 817)

Systemmeldungen (Seite 797)

Steuerungsmeldungen bearbeiten

Einleitung

Steuerungsmeldungen werden in STEP 7 projektiert. Wenn WinCC im STEP 7-Umfeld betrieben wird, stehen in WinCC Steuerungsmeldungen zur Verfügung.

In WinCC bearbeiten Sie nur die bediengerätespezifischen Eigenschaften von Steuerungsmeldungen. Abhängig von Bediengerät und Steuerung sind unterschiedliche Eigenschaften einer Steuerungsmeldung editierbar.

Sie filtern bei Bedarf die angezeigten Steuerungsmeldungen für Ihr Projekt über die Anzeigeklassen, die in der Steuerung für die Steuerungsmeldungen projektiert wurden.

Hinweis

Wenn "Automatische Aktualisierung" für Steuerungsmeldungen für die Verbindung mit der Steuerung S7-1500 eingestellt ist, können Sie die Eigenschaften der jeweiligen Steuerungsmeldungen nicht bearbeiten. In diesem Fall werden die Steuerungsmeldungen im Editor "HMI-Meldungen" nicht angezeigt.

Voraussetzung

- Eine Verbindung zur Steuerung wurde hergestellt.
- In STEP 7 wurden Meldungen projektiert.

Steuerungsmeldungen bearbeiten

Um Steuerungsmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "HMI-Meldungen".
   Der Editor "HMI-Meldungen" wird geöffnet.
2. Öffnen Sie die Registerkarte "Steuerungsmeldungen".

3. Markieren Sie im Arbeitsbereich die Meldung, die Sie ändern wollen.
4. Führen Sie die gewünschten Änderungen aus. Abhängig vom Bediengerät sind nicht alle Eigenschaften der Steuerungsmeldung angezeigt oder aktivierbar.

SIMATIC S7 Meldungen bearbeiten

Diese Steuerungsmeldungen bearbeiten Sie bei Bedarf in der Meldeprojektierung von STEP 7. Gehen Sie dazu folgendermaßen vor:

1. Markieren Sie im Arbeitsbereich die Meldung, die Sie ändern wollen.
2. Wählen Sie im Kontextmenü "Im Meldeeditor der Steuerung öffnen".
   Die Meldungsprojektierung von STEP 7 wird geöffnet.
3. Führen Sie die gewünschten Änderungen aus.

Steuerungsmeldung über Anzeigeklassen filtern

Um Steuerungsmeldungen über die Anzeigeklassen zu filtern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation unter Ihrem Bediengerät "Runtime-Einstellungen > Meldungen".
   Unter "Steuerungsmeldungen" werden eine oder mehrere Verbindungen zu einer Steuerung angezeigt.
2. Wählen Sie für die Verbindung die Anzeigeklassen aus, deren Steuerungsmeldungen Sie anzeigen lassen wollen.

Hinweis

Steuerungsmeldungen nach einem Basic Panel-Gerätetausch

Damit nach einem Gerätetausch ("Gerätesicht > Gerät tauschen") von einem Basic Panel auf ein Comfort Panel, Mobile Panel oder auf WinCC RT Advanced die Steuerungsmeldungen (PLC-Überwachungsmeldungen, Programmmeldungen oder Systemmeldungen) angezeigt werden, müssen Sie unter "Runtime Einstellungen > Meldungen > Steuerungsmeldungen“ für die jeweiligen Verbindungen die Einstellung "Anzeigeklassen“ aktivieren.

Ergebnis

Im Editor "HMI-Meldungen" werden auf der Registerkarte "Steuerungsmeldungen" nur die Steuerungsmeldungen angezeigt, deren Anzeigeklasse Sie für Ihr Bediengerät aktiviert haben.

Siehe auch

Steuerungsmeldungen (Seite 794)

Systemdefinierte Steuerungsmeldungen aktivieren

Systemdefinierte Steuerungsmeldungen aktivieren

Um systemdefinierte Steuerungsmeldungen vom Typ "SIMATIC S7 Diagnosemeldung" für Ihr Bediengerät zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Meldungen".
   Die Einstellungen zu Meldungen werden geöffnet.
2. Aktivieren Sie unter "Systemmeldungen" das Optionskästchen "S7 Diagnosemeldungen (Nummer)".
3. Wenn Sie die Systemmeldungen mit Meldetext anzeigen lassen wollen, aktivieren Sie "Zeige Text".
4. Um diese Meldungen in Runtime anzeigen zu lassen, projektieren Sie eine Meldeanzeige oder ein Meldefenster, welche die Meldungen der Meldeklasse "Diagnosis Events" anzeigt.

Hinweis

Bei Bediengeräten mit kleinem Display legen Sie für systemdefinierte Steuerungsmeldungen fest, dass sie nur mit Meldenummer angezeigt werden.

Hinweis

Die beschriebene Vorgehensweise ist für systemdefinierte Steuerungsmeldungen vom Typ "SIMATIC S7 Diagnosemeldung" nur für die Verbindungen mit Steuerungen SIMATIC S7-300/400 anwendbar.

Siehe auch

Systemdefinierte Steuerungsmeldungen (Seite 797)

3.2.2.6 Meldungen projektieren

Projektieren von Meldeblöcken

Grundlagen zu Meldeblöcken

Einleitung

Die Darstellung von Meldungen setzt sich aus Meldeblöcken zusammen. Jeder Meldeblock entspricht einer Spalte in der tabellarischen Anzeige der Meldeanzeige.

Es gibt drei Gruppen von Meldeblöcken:

- Systemblöcke
  Systemblöcke sind Systemdaten, z. B. Datum, Uhrzeit, Meldenummer und Zustand.
- Anwendertextblöcke
  Anwendertextblöcke beinhalten den Meldetext mit der Beschreibung der Ursache einer Störung und Zusatztexte mit Informationen, z. B. dem Störort zur Lokalisierung der Störung
- Parameterblöcke
  Parameterblöcke dienen der Anbindung der Meldungen an Prozesswerte, z. B. aktuelle Füllstände, Temperaturen oder Drehzahlen. Pro Meldung projektieren Sie bis zu 10 Parameterblöcke.

Meldeblöcke konfigurieren

Sie aktivieren und bearbeiten die Meldeblöcke in der Registerkarte "Meldeblöcke". Um Meldeblöcke in einer Meldeanzeige anzuzeigen, wählen Sie im Editor "Bilder" aus den aktivierten Meldeblöcken die Meldeblöcke aus, die angezeigt und archiviert werden sollen.

Textlänge

Die Textlänge eines Anwendertextblocks beträgt bis zu 255 Zeichen.

Die Textlänge eines Parameterblocks ist wie folgt projektierbar:

- Steuerungsmeldungen: maximal 32 Zeichen
  Um sicherzustellen, dass alle Zeichen angezeigt werden, legen Sie die Länge des Parameterblocks auf 32 Zeichen fest.
- Andere Meldeverfahren: maximal 255 Zeichen

Hinweis

Systemblock "Zustand"

Im Systemblock "Zustand" werden die Zustandstexte der Meldungen angezeigt. Projektieren Sie diesen Meldeblock in ausreichender Länge, damit die Zustandstexte vollständig angezeigt werden.

Blinken

Eine quittierpflichtige Meldung oder einzelne Meldeblöcke stellen Sie in Runtime bei Bedarf blinkend dar. Dazu sind folgende Voraussetzungen notwendig:

- In der Meldeklasse der Meldung muss die Eigenschaft "Blinken ein" aktiviert. Sie projektieren die Eigenschaft in der Registerkarte "Meldeklassen".
- Im Inspektorfenster der Meldeblöcke, die blinken sollen, ist unter "Allgemein" im Bereich "Meldeblock" die Option "Blinken" aktiviert.

Ausrichten

Sie legen auf der Registerkarte "Meldeblöcke" fest, ob der Meldeblock einer Spalte der Meldeanzeige rechts, links oder zentriert angezeigt wird.

Format

Für bestimmte Meldeblöcke wählen Sie bei Bedarf zwischen verschiedenen Darstellungsformaten.

Siehe auch

Blinken von Meldeblöcken einrichten (Seite 842)

Meldeblöcke projektieren

Einleitung

Eine Meldung besteht aus projektierten Meldeblöcken. Sie aktivieren die Meldeblöcke für ein Bediengerät in der Registerkarte "Meldeblöcke". Sie wählen aus den aktivierten Meldeblöcken für jede Meldeanzeige die aktivierten Meldeblöcke aus, die angezeigt und archiviert werden sollen.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Registerkarte "Meldeblöcke" ist geöffnet.

Meldeblöcke aktivieren

1. Wählen Sie in den Registerkarten der Meldeblöcke die Registerkarte "Systemblöcke", "Anwendertextblöcke" oder "Parameterblöcke".
2. Selektieren Sie den Meldeblock, den Sie auf Ihrem Bediengerät anzeigen lassen wollen.
3. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Blöcke > "Aktivieren".
4. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Einstellungen" den Anzeigenamen für den Meldeblock fest.

   Hinweis

   Unter "Eigenschaften > Eigenschaften > Allgemein > Einstellungen" werden in der Spalte "Name im Editor Meldungen" die im System hinterlegten schreibgeschützten Namen der Meldeblöcke angezeigt. Diese Spalte ist für die Registerkarten "Anwendertextblöcke" und "Parameterblöcke" verfügbar.

   Anhand der Spalte "Name im Editor Meldungen" erkennen Sie den Bezug zwischen den Anzeigenamen der Meldeblöcke in Runtime und den Bezeichnungen im Editor "Meldungen", auch wenn die Anzeigenamen der Meldeblöcke geändert wurden.

5. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Blöcke" die Textlänge und die Ausrichtung des Meldeblocks in der Spalte der Meldeanzeige fest.
6. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Einstellungen > Format" das Format des Meldeblocks. Diese Option steht nicht für alle Meldeblöcke zur Verfügung.

Ergebnis

Die Meldeblöcke sind aktiviert und werden im Editor "Bilder" im Inspektorfenster der Meldeanzeige angezeigt.

Meldeblöcke deaktivieren

1. Wählen Sie die Registerkarte der Meldeblöcke, deren Meldeblöcke Sie deaktivieren wollen.
2. Deaktivieren Sie die Meldeblöcke, die Sie für Meldeanzeige und -archivierung nicht mehr verwenden wollen.
3. Verfahren Sie mit allen Meldeblöcken entsprechend, die Sie nicht mehr anzeigen wollen.

Siehe auch

Beschreibung der Systemblöcke (Seite 997)

Blinken von Meldeblöcken einrichten (Seite 842)

Blinken von Meldeblöcken einrichten

Einleitung

Ob ein Meldeblock in Runtime blinkt, ist abhängig davon, welcher Meldeklasse die Meldung angehört. Die Meldeklasse legt fest, dass die enthaltenen Meldungen blinken, wenn Sie zum Quittieren anstehen.

Voraussetzung

- Die Meldeblöcke sind für das Bediengerät aktiviert.
- Die Meldeblöcke sind für die jeweilige Meldung aktiviert.
- Die Meldung ist einer Meldeklasse mit einem Zustandsautomaten zugeordnet, der festlegt, dass enthaltene Meldungen blinken.
- Die Meldeblöcke sind in der Meldeanzeige aktiviert.
- Die Registerkarte "Meldeblöcke" ist geöffnet.

Vorgehen

Um das Blinken von Meldeblöcken in Runtime zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Registerkarte "Meldeblöcke" in den untergeordneten Registerkarten den Meldeblock, der in Runtime blinken soll.
2. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Blöcke > Blinken".
3. Wiederholen Sie diesen Schritt für alle Meldeblöcke, die in Runtime blinken sollen.

Ergebnis

In Runtime blinken diese Meldeblöcke, wenn die Meldung zum Quittieren ansteht.

Siehe auch

Grundlagen zu Meldeblöcken (Seite 839)

Meldeblöcke projektieren (Seite 841)

Projektieren von Analogmeldungen

Analogmeldungen projektieren

Einleitung

Im Editor "HMI-Meldungen" projektieren Sie Analogmeldungen und legen die Eigenschaften der Analogmeldungen fest. Alternativ projektieren Sie Analogmeldungen im Editor "HMI-Variablen".

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Analogmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Analogmeldungen".
2. Um eine neue Analogmeldung anzulegen, doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Analogmeldung wird angelegt.

3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
   - Ändern Sie bei Bedarf den Objektnamen der Meldung.
   - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.
4. Um die CPU-Nummer oder PLC anzeigen zu lassen, geben Sie diese unter "Verbindung zur Steuerung" ein. Diese Option ist nur dann aktiv, wenn auf der Registerkarte "Meldeblöcke > Systemblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
5. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger" die Variable, welche die Meldung triggert. Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.
6. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Meldetexte" den Meldetext.
   - Geben Sie unter "Ereignistext" den Meldetext ein.
   - Legen Sie bei Bedarf zusätzliche Meldetexte unter "Zusatztexte" an.
   - Nutzen Sie die Funktionen des Kontextmenüs, um Text zeichenweise zu formatieren und um Ausgabefelder für Systemparameter oder Meldungsparameter einzufügen.

   Hinweis

   Sie geben nur in die Zeilen Zusatztexte ein, die Sie als Anwendertextblöcke in der Registerkarte "Meldeblöcke" aktiviert haben.

Grenzwerte festlegen

1. Markieren Sie die Analogmeldung, der Sie die Grenzwerte zuordnen wollen.
2. Klicken Sie im unteren Bereich des Arbeitsbereichs unter "Grenzwerte für Analogmeldung" auf "<Hinzufügen>".

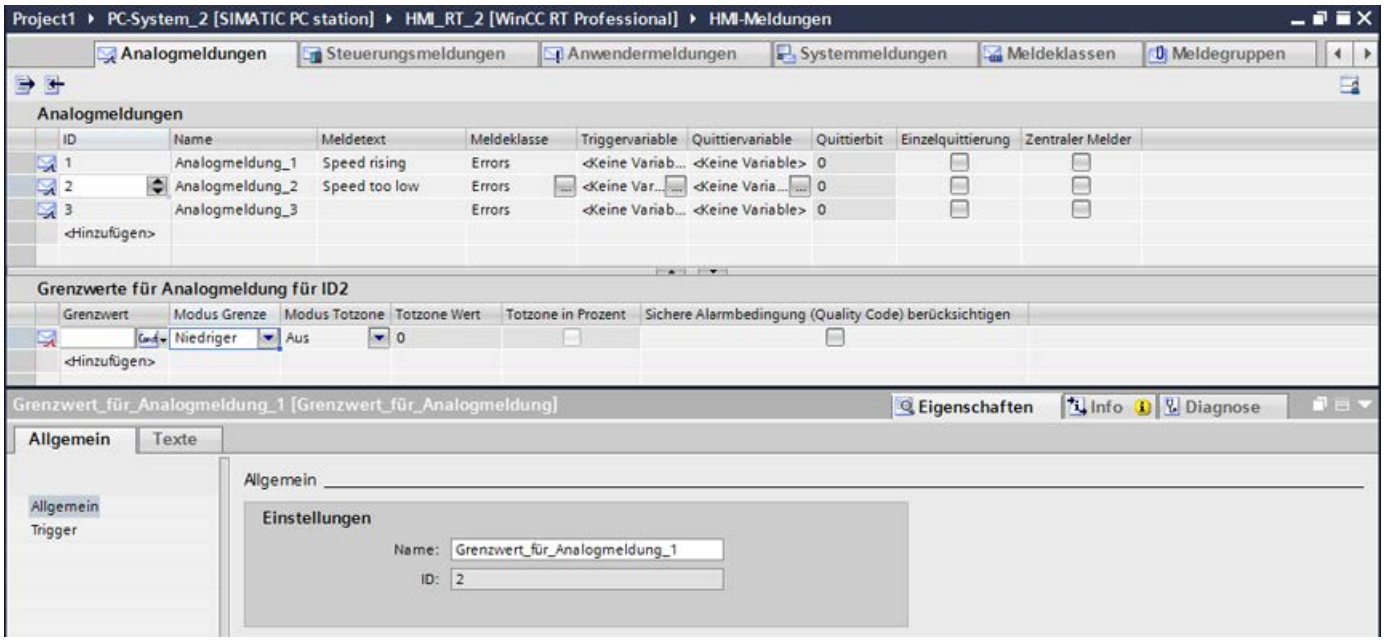

3. Um eine **Konstante** als Grenzwert zu verwenden:
   - Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Trigger > Wert" auf die Schaltfläche und wählen Sie "Konstante".
   - Geben Sie den gewünschten Grenzwert ein.
4. Um eine **Variable** als Grenzwert zu verwenden:
   - Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger > Wert" auf die Schaltfläche und wählen Sie "HMI_Variable".

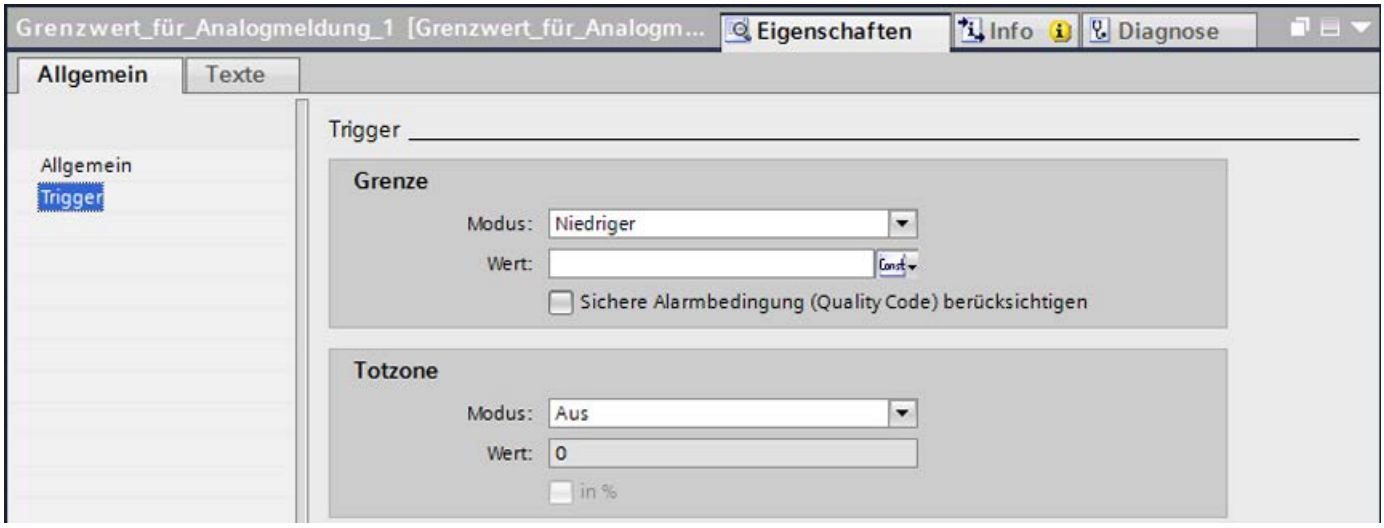

   - Klicken Sie auf die Schaltfläche ... . Die Objektliste wird geöffnet.
   - Wählen Sie die Variable.
5. Wählen Sie im Feld "Modus" den Triggermodus.
   - "Entspricht dem Grenzwert": Wenn der Grenzwert erreicht ist, wird die Meldung getriggert.
   - "Entspricht nicht dem Grenzwert": Wenn der Grenzwert nicht mehr erreicht ist, wird die Meldung getriggert.
   - "Überschreitung": Wenn der Grenzwert überschritten ist, wird die Meldung getriggert.
   - "Unterschreitung": Wenn der Grenzwert unterschritten wird, wird die Meldung getriggert.
6. Legen Sie für die Meldung bei Bedarf weitere Grenzwerte an.

Hinweis

Wenn in der Auswahl das gewünschte Objekt noch nicht existiert, legen Sie es in der Objektliste an und ändern Sie dessen Eigenschaften später.

Hinweis

Wenn ein Prozesswert um den Grenzwert pendelt, wird für diese Störung die zugehörige Meldung mehrmals getriggert. Projektieren Sie dann eine Totzone oder eine Verzögerungszeit.

Optionale Einstellungen für Analogmeldungen

Verzögerungszeit einstellen

Um die Meldung verzögert einblenden zu lassen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Trigger".
3. Geben Sie unter "Verzögerung" eine Zeitspanne ein. Innerhalb dieser Zeit muss sich die Triggerbedingung erfüllen, damit die Meldung getriggert wird.

Totzone einstellen

Um einen Toleranzbereich zum Kommen oder Gehen einer Meldung einzugeben, gehen Sie folgendermaßen vor:

1. Markieren Sie den Grenzwert, dem die Totzone zugeordnet werden soll.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Trigger".
3. Wählen Sie unter "Totzone > Modus", bei welcher Änderung des Meldezustand die Totzone berücksichtigt wird.
4. Geben Sie unter "Totzone > Wert" einen konstanten Wert ein.
5. Um festzulegen, dass der Wert der Totzone als Prozentangabe auf den Grenzwert bezogen wird, aktivieren Sie "%".

Zustand der Meldung speichern

Um die Zustände einer Meldung in einer Variablen zu speichern, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustand " die Variable und das Bit, das die Meldezustände gekommen/gegangen und die Quittieranforderung speichert.

Priorität einstellen

Um die Priorität einzustellen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Priorität" einen Wert zwischen 0 und 16.
   Wenn Sie in der Meldeanzeige nach Priorität filtern, erscheint die Meldung mit Priorität 0 an erster Stelle.

Hilfetext erstellen

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tooltipp" und geben den gewünschten Text ein.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

Siehe auch

Quittieren des zentralen Melders projektieren (Seite 918)

Protokollieren von Meldungen projektieren (Seite 911)

Quittieren einer Meldung projektieren (Seite 915)

Parameter in Analogmeldung ausgeben (Seite 847)

Loop-In-Alarm projektieren (Seite 864)

Meldungen im Editor "HMI-Variablen" (Seite 866)

Projektieren der Anzeigeunterdrückung von Meldungen (Seite 896)

Übersicht über das Projektieren von Meldungen (Seite 817)

Parameter in Analogmeldung ausgeben

Einleitung

In einer Analogmeldung fügen Sie zur Anzeige von System- oder Meldungsparametern ein entsprechendes Ausgabefeld ein.

- Als Meldungsparameter stehen "Grenze", "Totband" oder "Aktueller Wert" zur Auswahl.
- Als Systemparameter stehen folgende Systemblöcke zur Auswahl:
  - Benutzername
  - Computername
  - Kommentar
  - Name der Anwendung

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Analogmeldung ist markiert.

Vorgehen

Um einen Parameter im Meldetext auszugeben, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Meldetext an der gewünschten Stelle.
2. Um einen Meldungsparameter auszugeben, wählen Sie im Kontextmenü "Parameterausgabefeld in Analogmeldung einfügen".
   -ODER-
   Um eine Systemparameter auszugeben, wählen Sie im Kontextmenü "Systemblockausgabefeld einfügen".
3. Legen Sie für Meldungsparameter Länge, Anzahl der Nachkommastellen, Genauigkeit und Ausrichtung des Ausgabefeldes fest.
   Um führende Nullstellen im Ausgabefeld anzuzeigen, aktivieren Sie "Führende Nullen".
4. Um die Eingaben zu speichern, klicken Sie auf das Symbol .

WinCC fügt in den Meldetext einen Platzhalter für das Ausgabefeld ein:

- Für Meldungsparameter: "<tag: n, [###]>" wobei n = Textlänge in Zeichen.
- Für Systemparameter; "<Name des Systemparameters>"

Hinweis

Wenn Sie ein Projekt mit dem Simulator testen, werden die Werte aus Ausgabefeldern innerhalb von Meldetexten nicht ausgegeben.

Parameter aus dem Meldetext löschen

Um ein Ausgabefeld aus dem Meldetext zu löschen, gehen Sie folgendermaßen vor:

- Markieren Sie im Meldetext das Ausgabefeld und wählen Sie im Kontextmenü "Löschen".

Hinweis

Die Reihenfolge der Ausgabefelder im Meldetext ist sprachabhängig. Beim Archivieren von Meldungen in ein CSV-Archiv wird die Reihenfolge der Runtime-Sprache verwendet.

Die Änderung der Variable eines Ausgabefeldes in einer Sprache führt dazu, dass in allen anderen Sprachen das geänderte Ausgabefeld am Meldetextende steht. Dadurch ändert sich die Reihenfolge der Ausgabefelder im Archiv.

Projektieren von Bitmeldungen

Bitmeldungen projektieren

Einleitung

Bitmeldungen zeigen Zustandsänderungen in einer Anlage an und werden von der Steuerung getriggert. Sie zeigen z. B. an, ob ein Ventil geöffnet oder geschlossen ist.

Im Folgenden ist die Projektierung im Editor "HMI-Meldungen" beschrieben. Alternativ projektieren Sie Bitmeldungen im Editor "HMI-Variablen".

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Bitmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Bitmeldungen".
2. Um eine neue Bitmeldung anzulegen, doppelklicken Sie in der Tabelle auf "<Hinzufügen>". Eine neue Bitmeldung wird angelegt.
3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
   - Ändern Sie bei Bedarf den Objektnamen der Meldung.
   - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.
   - Um die CPU-Nummer oder PLC anzeigen zu lassen, geben Sie diese unter "Verbindung" ein. Diese Option ist nur dann aktiv, wenn auf der Registerkarte "Meldeblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger" die Variable und das Bit, welches die Meldung triggert. Beachten Sie dabei Folgendes:
   - Verwenden Sie den Datentyp "Bool" "USInt", "UInt", oder "UDInt".
   - Verwenden Sie ein Bit einer Triggervariablen nur für eine Meldung.
   - Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.
   - Der Ein- und Ausgangsbereich einer PLC-Variablen ist als Trigger ungeeignet.

Hinweis

Wenn in der Auswahl das gewünschte Objekt noch nicht existiert, legen Sie es direkt in der Objektliste an und ändern dessen Eigenschaften später.

Hinweis

Wenn Sie eine Bitmeldung erzeugen, indem Sie den Meldetext kopieren und in eine neue Tabellenzeile einfügen, wird das Triggerbit nicht automatisch hochgezählt. Achten Sie darauf, das Triggerbit nicht mehrfach zu belegen.

Hinweis

Beachten Sie bei der Angabe des Bits die Zählweise der Bits in der verwendeten Steuerung. Weitere Hinweise dazu finden Sie in der Online-Hilfe zu der jeweiligen Steuerung im Abschnitt "Kommunikation".

1. Legen Sie unter "Modus" fest, ob die Meldung bei steigender oder fallender Flanke ausgelöst wird.
2. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Meldetexte" den Meldetext.
   – Geben Sie unter "Ereignistext" den Meldetext ein.
   – Legen Sie bei Bedarf zusätzliche Meldetexte unter "Zusatztexte" an.
   – Nutzen Sie die Funktionen des Kontextmenüs, um Text zeichenweise zu formatieren und um Ausgabefelder für Parameter einzufügen.

   Hinweis

   Sie geben nur in die Zeilen Zusatztexte ein, die Sie als Anwendertextblöcke in der Registerkarte "Meldeblöcke" aktiviert haben.

Ergebnis

Die Bitmeldung ist angelegt.

Weitere Einstellungen für Bitmeldungen

Hilfetext erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Hilfetext" und geben Sie den gewünschten Tip ein.

Parameter für eine Bitmeldung aktivieren

Um Prozesswerte in einem Ausgabefeld im Meldetext auszugeben, ordnen Sie den Parameterblöcken Variablen zu. Gehen Sie dazu folgendermaßen vor:

1. Markieren Sie die Meldung.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Meldeparameter".
3. Geben Sie für den Meldeparameter einen Wert ein oder wählen Sie eine Variable.
4. Geben Sie nach Bedarf mehrere Parameter für die Meldung ein.
   Die aktivierten Prozesswerte fügen Sie als Auswahlfeld in einen Meldetext ein oder blenden Sie als Parameterblock in der Meldeanzeige ein.

   Hinweis

   Um die Prozesswerte als Parameterblöcke einzublenden, müssen die Parameterblöcke in der Meldeanzeige aktiviert sein.

   Sie bearbeiten nur Parameterblöcke, die Sie in der Registerkarte "Meldeblöcke" unter "Parameterblöcke" aktiviert haben.

Meldezustand speichern

Um die Zustände einer Meldung in einer Variablen zu speichern, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bitmeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustand" die Variable und das Bit, das die Meldezustände gekommen/gegangen und die Quittieranforderung speichert.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

Siehe auch

Quittieren einer Meldung projektieren (Seite 915)

Quittieren des zentralen Melders projektieren (Seite 918)

Loop-In-Alarm projektieren (Seite 864)

Meldungen im Editor "HMI-Variablen" (Seite 866)

Projektieren der Anzeigeunterdrückung von Meldungen (Seite 896)

Protokollieren von Meldungen projektieren (Seite 911)

Übersicht über das Projektieren von Meldungen (Seite 817)

Parameter in Bitmeldung ausgeben

Einleitung

In einer Bitmeldung fügen Sie zur Anzeige von System- oder Meldeparametern ein entsprechendes Ausgabefeld ein.

- Als Meldeparameter stehen die Parameter zur Auswahl, die unter "Eigenschaften > Eigenschaften > Meldeparameter" konfiguriert sind.
- Als Systemparameter stehen folgende Systemblöcke zur Auswahl:
  - Benutzername
  - Computername
  - Kommentar
  - Name der Anwendung

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Bitmeldung ist markiert.

Vorgehen

Um einen Parameter im Meldetext auszugeben, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Meldetext an der gewünschten Stelle.
2. Um einen Meldeparameter auszugeben, wählen Sie im Kontextmenü "Parameterausgabefeld einfügen".
   -ODER-
   Um eine Systemparameter auszugeben, wählen Sie im Kontextmenü "Systemblockausgabefeld einfügen".
3. Wählen Sie den gewünschten Parameter aus.
4. Legen Sie für Meldeparameter darüber hinaus bei Bedarf Folgendes fest:
   – Die Variable, die den Parameter mit Werten versorgt.
     Es wird die Variable vorgetragen, die unter "Eigenschaften > Eigenschaften > Meldeparameter" für den Parameter konfiguriert ist. WinCC aktualisiert die Parameterkonfiguration unter "Eigenschaften > Eigenschaften > Meldeparameter", wenn Sie hier eine andere Variable auswählen.
   – Länge, Anzahl der Nachkommastellen, Genauigkeit und Ausrichtung des Ausgabefeldes
   – Um führende Nullstellen im Ausgabefeld anzuzeigen, aktivieren Sie "Führende Nullen".
5. Um die Eingaben zu speichern, klicken Sie auf das Symbol .

   Hinweis

   Wenn Sie ein Projekt mit dem Simulator testen, werden die Werte aus Ausgabefeldern innerhalb von Meldetexten nicht ausgegeben.

Hinweis

Die Reihenfolge der Ausgabefelder für Variablen im Meldetext ist sprachabhängig. Beim Archivieren von Meldungen in ein CSV-Archiv wird die Reihenfolge der Runtime-Sprache verwendet.

Die Änderung der Variable eines Ausgabefeldes in einer Sprache führt dazu, dass in allen anderen Sprachen das geänderte Ausgabefeld am Meldetextende steht. Dadurch ändert sich die Reihenfolge der Ausgabefelder im Archiv.

Siehe auch

Bitmeldungen projektieren (Seite 848)

Bitmeldungen projektieren (Seite 866)

Projektieren von Anwendermeldungen

Anwendermeldung projektieren

Einleitung

Sie verwenden Anwendermeldungen zum Anzeigen von Bedienaktionen in Runtime. Anwendermeldungen werden über die Meldenummer getriggert. Für andere, benutzerdefinierte Anwendungen projektieren Sie Anwendermeldungen, die in Runtime über Skripte getriggert werden.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Anwendermeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Anwendermeldungen".
2. Um eine neue Anwendermeldung anzulegen, doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Eine neue Anwendermeldung wird angelegt.
3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
   - Ändern Sie bei Bedarf den Objektnamen der Meldung.
   - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.
   - Um die CPU-Nummer oder PLC anzeigen zu lassen, geben Sie diese unter "Verbindung" ein. Diese Option ist nur dann aktiv, wenn auf der Registerkarte "Meldeblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
4. Klicken Sie im Inspektorfenster auf "Meldetexte" und konfigurieren Sie den Meldetext.
   - Geben Sie den Meldetext ein.
   - Legen Sie bei Bedarf zusätzliche Meldetexte unter "Zusatztexte" an.
   - Fügen Sie bei Bedarf Ausgabefelder in den Meldetext oder Zusatztext ein.

     Hinweis

     Sie geben nur in die Zeilen Zusatztexte ein, die Sie als Anwendertextblöcke in der Registerkarte "Meldeblöcke" aktiviert haben.

Ergebnis

Die Anwendermeldung ist angelegt.

Weitere Einstellungen für Anwendermeldungen

Tooltip erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anwendermeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tooltip" und geben Sie den gewünschten Text ein.

Meldezustand speichern

Um die Zustände einer Meldung in einer Variablen zu speichern, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anwendermeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustand" die Variable und das Bit, das die Meldezustände Gekommen/Gegangen und die Quittieranforderung speichert.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anwendermeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

Siehe auch

Quittieren einer Meldung projektieren (Seite 915)

Quittieren des zentralen Melders projektieren (Seite 918)

Quittierung projektieren (Seite 915)

Projektieren von Meldeblöcken (Seite 839)

Projektieren der Anzeigeunterdrückung von Meldungen (Seite 896)

Parameter in Anwendermeldungen ausgeben

Einleitung

In einer Anwendermeldung fügen Sie zur Anzeige von System- oder Meldeparametern ein entsprechendes Ausgabefeld ein.

- Als Systemparameter stehen folgende Systemblöcke zur Auswahl
  - Benutzername
  - Computername
  - Kommentar
  - Anwendung
- Meldeparameter konfigurieren Sie im Skript, mit dem Sie die Anwendermeldung in Runtime triggern.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Anwendermeldung ist markiert.

Vorgehen

Um einen Parameter im Meldetext auszugeben, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Meldetext an der gewünschten Stelle.
2. Um einen Meldeparameter auszugeben, wählen Sie im Kontextmenü "Parameterfeld einfügen".
   -ODER-
   Um einen Systemparameter auszugeben, wählen Sie im Kontextmenü "Systemparameter einfügen".
3. Wählen Sie den gewünschten Parameter aus.
4. Um die Eingaben zu speichern, klicken Sie auf .

Hinweis

Wenn Sie ein Projekt mit dem Simulator testen, werden die Werte aus Ausgabefeldern innerhalb von Meldetexten nicht ausgegeben.

Hinweis

Die Reihenfolge der Ausgabefelder für Variablen im Meldetext ist sprachabhängig. Beim Archivieren von Meldungen in ein CSV-Archiv wird die Reihenfolge der Runtime-Sprache verwendet.

Die Änderung der Variable eines Ausgabefeldes in einer Sprache führt dazu, dass in allen anderen Sprachen das geänderte Ausgabefeld am Meldetextende steht. Dadurch ändert sich die Reihenfolge der Ausgabefelder im Archiv.

Siehe auch

Beispiel: Meldeparameter für eine Anwendermeldung definieren (Seite 857)

Beispiel: Trigger für eine Anwendermeldung projektieren

Aufgabe

Im folgenden Beispiel projektieren Sie eine Anwendermeldung, die in Runtime über eine Schaltfläche getriggert wird.

Dabei projektieren Sie an das Ereignis "Klicken" dieser Schaltfläche ein Skript. Dieses Skript ruft eine Meldenummer auf und setzt die dazugehörige Meldung in den Status "Gekommen".

Voraussetzungen

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Ein Bild mit einer Meldeanzeige und einer Schaltfläche ist geöffnet.

Vorgehensweise

1. Projektieren Sie im Editor "HMI-Meldungen" eine Anwendermeldung.

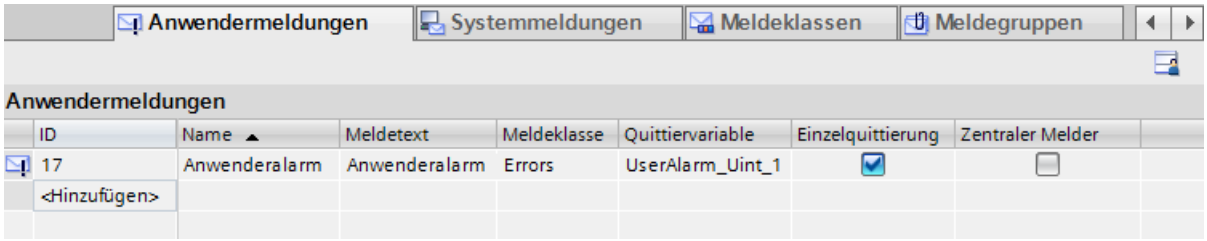

   Die Anwendermeldung hat in diesem Fall die Meldenummer "17".

2. Projektieren Sie im Bild an das Ereignis "Klicken" der Schaltfläche das folgende Skript:

```
Sub onClick(byVal item)
    Dim UserAlarm
    Dim ID_UA
    Dim State_UserAlarm

    ID_UA = "17"
    State_UserAlarm = "5"

    Set UserAlarm = HMIRuntime.Alarms(ID_UA)
    UserAlarm.State = state_UserAlarm
    UserAlarm.Comment = "empty Comment"
    UserAlarm.UserName = "Max Mustermann"
    UserAlarm.Create "MyApplication"
```

```
End Sub
```

Ergebnis

Wenn der Benutzer in Runtime auf diese Schaltfläche klickt, wird in der Meldeanzeige die projektierte Anwendermeldung angezeigt.

Siehe auch

Beispiel: Meldeparameter für eine Anwendermeldung definieren (Seite 857)

Beispiel: Meldeparameter für eine Anwendermeldung definieren

Aufgabe

Im folgenden Beispiel projektieren Sie eine Anwendermeldung, die in Runtime über eine Schaltfläche getriggert wird. Für die Anwendermeldung definieren Sie Meldeparameter, die im Meldetext ausgegeben werden.

Dabei projektieren Sie an das Ereignis "Klicken" dieser Schaltfläche die C-Systemfunktion "TriggerOperatorEvent". Diese Systemfunktion ruft eine Meldenummer auf und übergibt dabei folgende Parameter:

- Variablenname als Parameter 1 im Meldetext
- Startwert der Variable als Parameter 2 im Meldetext
- Endwert der Variable als Parameter 3 im Meldetext

Hinweis

Meldeparameter im Skript

Um Meldeparameter zu definieren, stehen in WinCC verschiedene Funktionen zur Verfügung, z. B. die Funktion "MSRTCreateMsgInstanceWithCommentPlus" der Runtime API-Schnittstelle.

Voraussetzungen

- Ein Bild mit einer Meldeanzeige und einer Schaltfläche ist geöffnet.
- Die Anwendermeldung mit der Meldenummer 0x001 ist projektiert und verwendet im Meldetext die Meldeparameter 1 - 3.
- Die Variable "HMI_Tag_1" vom Typ "Integer" ist projektiert und verschaltet.

Vorgehensweise

1. Projektieren Sie im Bild an das Ereignis "Klicken" der Schaltfläche das folgende Skript:

```
Void SCFunction_1()
(
int a, b;
a = GetTagDWord ("HMI_Tag_1"); // old value
SetTagDWord("HMI_Tag_1", 100);
b = GetTagDWord ("HMI_Tag_1"); // new value
TriggerOperatorEvent(0x001, 1, "HMI_Tag_1", 0, a, b, "myComment");
}
```

Ergebnis

Wenn der Benutzer in Runtime auf diese Schaltfläche klickt, wird in der Meldeanzeige die projektierte Anwendermeldung angezeigt. Der Variablenname wird als Parameter 1 ausgegeben, Start- und Endwert der Variablen als Parameter 2 und 3.

Siehe auch

Beispiel: Trigger für eine Anwendermeldung projektieren (Seite 856)

Parameter in Anwendermeldungen ausgeben (Seite 855)

Meldetext projektieren

Einleitung

Der Meldetext einer Meldung setzt sich aus Meldetext, Zusatztexten und Ausgabefeldern zusammen. Meldetext und Zusatztexte sind Anwendertextblöcke. Für eine Meldung stehen Ihnen bis zu 10 Anwendertextblöcke zur Verfügung.

Jeder Meldetext enthält bis zu 255 Zeichen und beliebig viele Ausgabefelder.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Anwendertextblöcke sind in der Registerkarte "Meldeblöcke" aktiviert.
- Eine Meldung ist angelegt.

Vorgehen

Um dem Meldetext einer Meldung weitere Anwendertextblöcke anzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung auf der entsprechenden Registerkarte, die Sie bearbeiten wollen.
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Meldetexte" den "Meldetext" ein.
3. Fügen Sie bei Bedarf in den Meldetext Ausgabefelder ein.
4. Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Meldetexte > Zusatztext" einen weiteren Meldetext ein.
   Abhängig davon, welche Anwendertextblöcke Sie in der Registerkarte "Meldeblöcke" unter "Anwendertextblöcke" aktiviert haben, geben Sie weitere Zusatztexte ein.

   Hinweis

   Wenn der Text zu lang zur Anzeige im Eingabefeld ist, scrollen Sie mit dem Symbol nach oben oder unten. Die zusätzlichen Textzeilen werden angezeigt.

Ergebnis

Sie haben den Meldetext mit Zusatztexten für diese Meldung projektiert. Um Sie in Runtime anzuzeigen, aktivieren Sie diese Anwendertextblöcke in einer Meldeanzeige.

Siehe auch

Analogmeldungen projektieren (Seite 843)

Bitmeldungen projektieren (Seite 848)

Systemmeldungen bearbeiten

Einleitung

Systemmeldungen sind nur beschränkt änderbar. Sie können keine Systemmeldungen löschen oder neu anlegen.

Voraussetzungen

- Die Systemmeldungen sind importiert.
- Der Editor "HMI-Meldungen" ist geöffnet.

Vorgehen

Um Systemmeldungen zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Systemmeldungen".
2. Markieren Sie die Systemmeldung, die Sie bearbeiten wollen.
3. Ändern Sie im Inspektorfenster bei Bedarf die bearbeitbaren Eigenschaften.
4. Markieren Sie die Systemmeldung, die Sie bearbeiten wollen.
5. Ändern Sie im Inspektorfenster bei Bedarf die bearbeitbaren Eigenschaften.

Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben zu projektieren, gehen Sie folgendermaßen vor:

- Projektieren Sie unter "Eigenschaften > Ereignisse" für das gewünschte Ereignis eine Funktionsliste.

Steuerungsmeldungen bearbeiten

Einleitung

Steuerungsmeldungen werden in STEP 7 projektiert. Wenn WinCC im STEP 7-Umfeld betrieben wird, stehen in WinCC Steuerungsmeldungen zur Verfügung.

In WinCC bearbeiten Sie nur die bediengerätespezifischen Eigenschaften von Steuerungsmeldungen. Abhängig von Bediengerät und Steuerung sind unterschiedliche Eigenschaften einer Steuerungsmeldung editierbar.

Sie filtern bei Bedarf die angezeigten Steuerungsmeldungen für Ihr Projekt über die Anzeigeklassen, die in der Steuerung für die Steuerungsmeldungen projektiert wurden.

Hinweis

Wenn "Automatische Aktualisierung" für Steuerungsmeldungen für die Verbindung mit der Steuerung S7-1500 eingestellt ist, können Sie die Eigenschaften der jeweiligen Steuerungsmeldungen nicht bearbeiten. In diesem Fall werden die Steuerungsmeldungen im Editor "HMI-Meldungen" nicht angezeigt.

Voraussetzung

- Eine Verbindung zur Steuerung wurde hergestellt.
- In STEP 7 wurden Meldungen projektiert.

Steuerungsmeldungen bearbeiten

Um Steuerungsmeldungen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "HMI-Meldungen".
   Der Editor "HMI-Meldungen" wird geöffnet.
2. Öffnen Sie die Registerkarte "Steuerungsmeldungen".
3. Markieren Sie im Arbeitsbereich die Meldung, die Sie ändern wollen.
4. Führen Sie die gewünschten Änderungen aus. Abhängig vom Bediengerät sind nicht alle Eigenschaften der Steuerungsmeldung angezeigt oder aktivierbar.

SIMATIC S7 Meldungen bearbeiten

Diese Steuerungsmeldungen bearbeiten Sie bei Bedarf in der Meldeprojektierung von STEP 7. Gehen Sie dazu folgendermaßen vor:

1. Markieren Sie im Arbeitsbereich die Meldung, die Sie ändern wollen.
2. Wählen Sie im Kontextmenü "Im Meldeeditor der Steuerung öffnen".
   Die Meldungsprojektierung von STEP 7 wird geöffnet.
3. Führen Sie die gewünschten Änderungen aus.

Steuerungsmeldung über Anzeigeklassen filtern

Um Steuerungsmeldungen über die Anzeigeklassen zu filtern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation unter Ihrem Bediengerät "Runtime-Einstellungen > Meldungen".
   Unter "Steuerungsmeldungen" werden eine oder mehrere Verbindungen zu einer Steuerung angezeigt.
2. Wählen Sie für die Verbindung die Anzeigeklassen aus, deren Steuerungsmeldungen Sie anzeigen lassen wollen.

Hinweis

Steuerungsmeldungen nach einem Basic Panel-Gerätetausch

Damit nach einem Gerätetausch ("Gerätesicht > Gerät tauschen") von einem Basic Panel auf ein Comfort Panel, Mobile Panel oder auf WinCC RT Advanced die Steuerungsmeldungen (PLC-Überwachungsmeldungen, Programmmeldungen oder Systemmeldungen) angezeigt werden, müssen Sie unter "Runtime Einstellungen > Meldungen > Steuerungsmeldungen“ für die jeweiligen Verbindungen die Einstellung "Anzeigeklassen“ aktivieren.

Ergebnis

Im Editor "HMI-Meldungen" werden auf der Registerkarte "Steuerungsmeldungen" nur die Steuerungsmeldungen angezeigt, deren Anzeigeklasse Sie für Ihr Bediengerät aktiviert haben.

Siehe auch

Steuerungsmeldungen (Seite 794)

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 989)

Bedienmeldungen projektieren

Einleitung

Sie haben die Möglichkeit, bei der Bedienung im Prozess eine Bedienmeldung auszulösen. Die Projektierung einer Bedienmeldung ist abhängig vom bedienten Objekt sowie von der Art der Bedienung.

Für die Bedienung einiger Objekte, z. B. die Eingabe eines Wertes in ein E/A-Feld oder ein Listenfeld, können Sie die Eigenschaft "Bedienprotokoll" an diesen Objekten aktivieren. Der Aufbau der Meldung im Bedienprotokoll ist nicht editierbar. Das Bedienprotokoll enthält im Prozesswertblock 2 den Wert vor der Bedienung und im Prozesswertblock 3 den geänderten Wert.

Die Eigenschaft "Motiv abfragen" legt fest, ob das Motiv für eine Bedienung vom Bediener protokolliert wird. Durch Aktivieren der Option kann der Benutzer nach einer Bedienung in einem Dialog das Motiv (den Grund) der Bedienung als Kommentar zur Meldung eingeben. Das Bedienprotokoll wird im Meldesystem archiviert. Das Motiv wird als Kommentar zur Bedienmeldung in der Langzeitarchivliste angezeigt. Im Kommentardialog werden ausschließlich für die Bedienmeldung "12508141" der Altwert und der Neuwert angezeigt.

Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Ein E/A-Feld ist angelegt.
- Das Objekt ist mit einer Variablen verbunden.

Vorgehen

1. Markieren Sie das E/A-Feld.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Security".
3. Um Bedienmeldungen für das Objekt zu verwenden, aktivieren Sie die Option "Archivieren > Bedienprotokoll".
4. Wenn Sie das Bedienmotiv protokollieren möchten, aktivieren Sie die Eigenschaft "Motiv abfragen".

Ergebnis

Wenn der Benutzer in Runtime an dem E/A-Feld eine Bedienaktion ausführt, wird die Bedienmeldung ausgelöst. Bei der Bedienung des Objekts öffnet sich in Runtime ein Dialog, in dem der Bediener einen Text eingeben kann.

Siehe auch

Listenfeld (Seite 430)
Symbolisches E/A-Feld (Seite 505)
Kombinationsfeld (Seite 414)
Bildlaufleiste (Seite 361)

Bedienmeldung für eine Meldeanzeige projektieren

Einleitung

Bedienmeldungen für die Meldeanzeige sind vorkonfigurierte Systemmeldungen. Systemmeldungen verwalten Sie im Editor "HMI-Meldungen". Das Auslösen der Bedienmeldung projektieren Sie für die Meldeanzeige im Editor "Bilder".

Voraussetzung

- Die Systemmeldungen des Bediengeräts wurden importiert.
- Der Editor "Bilder" ist geöffnet.
- Ein Objekt "Meldeanzeige" ist angelegt.
- Im Editor "HMI-Meldungen" sind Meldeblöcke aktiviert.

Vorgehen

Um Bedienmeldungen für eine Meldeanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldeanzeige.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Bedienmeldung".
3. Aktivieren Sie die Bedienaktionen an der Meldeanzeige, bei denen eine Bedienmeldung ausgegeben werden soll.
   Im Ausgabefeld "Meldenummer" wird die Meldenummer der vorkonfigurierten Bedienmeldung für diese Bedienaktion angezeigt.
4. Um eine andere Meldung bei dieser Bedienaktion auszugeben, tragen Sie die Meldenummer der gewünschten Meldung ein.
5. Um die in der Bedienmeldung angezeigten Spalten festzulegen, wählen Sie unter "Meldeblöcke" auf die Schaltfläche .
   Ein Dialog wird geöffnet.

6. Aktivieren Sie unter "Meldeblock" die Meldeblöcke, die mit der Bedienmeldung ausgegeben werden sollen.
7. Legen Sie unter "Meldegruppe der Bedienmeldung" fest, welche Meldeblöcke mit der Systemmeldung ausgegeben werden. Unter "Ausgeben Als" legen Sie fest, ob der Inhalt des Meldeblocks als Text oder Wert ausgegeben wird.

Hinweis

Die Systemmeldung und deren Meldenummer finden Sie im Editor "HMI-Meldungen", in der Registerkarte "Systemmeldungen".

Ergebnis

Wenn der Benutzer in Runtime an der Meldeanzeige eine Bedienaktion ausführt, wird die projektierte Bedienmeldung angezeigt.

3.2.2.7 Loop-In-Alarm projektieren

Einleitung

Um nach dem Eingang einer Meldung zu Ihrem Bild zu wechseln, das die Informationen zu dem aufgetretenen Fehler enthält, wird ein Loop-In-Alarm projektiert.

Hinweis

Das Bild, das Informationen zu dem aufgetretenen Fehler enthält, projektieren Sie selbst. Es sind keine Standard-Bilder verfügbar, die automatisch in Abhängigkeit mit der markierten Fehlermeldung aufgerufen werden.

Voraussetzung

- Das Bild, das der Loop-In-Alarm aufruft, ist angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.

Vorgehen

Um für eine Meldung einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte, welche die Meldung enthält, für die Sie den Loop-In-Alarm projektieren wollen.
2. Markieren Sie die Meldung.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Loop-In-Alarm".

4. Wählen Sie die Systemfunktion "AktiviereBild".
5. Wählen Sie als Parameter das Bild, das durch den Loop-In-Alarm aufgerufen werden soll.

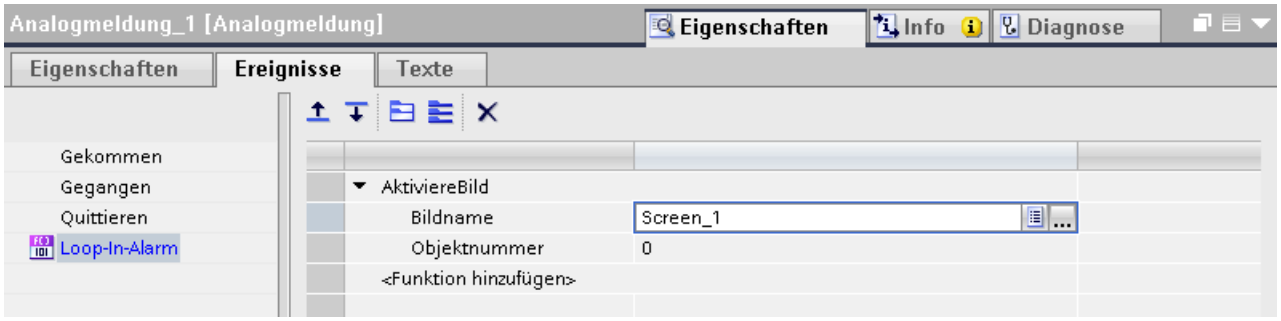

Hinweis

Um den Loop-In-Alarm für eine Meldeanzeige der Ausprägung "Meldezeile" zu projektieren, verwenden Sie eine Schaltfläche mit den folgenden Systemfunktionen:

- "BearbeiteMeldung" für Bediengeräte mit Tasten
- "MeldeanzeigeBearbeiteMeldung" für Bediengeräte ohne Tasten

Diese Systemfunktionen lösen das Ereignis "Loop-In-Alarm" aus. Die Meldezeile hat keine Schaltflächen.

Ergebnis

Wenn Sie in Runtime eine Meldung in der Meldeanzeige markieren und auf die Schaltfläche "Loop-In-Alarm" klicken, wird ein Bild mit Information zur markierten Meldung geöffnet.

Beim ersten Betätigen der Schaltfläche "Loop-In-Alarm" wird genau eine quittierpflichtige Meldung quittiert (falls quittierpflichtig), auch wenn mehrere Meldungen markiert sind. Die für das Meldeereignis "Loop-In-Alarm" hinterlegten Systemfunktionen werden immer ausgeführt, unabhängig davon, ob die Meldung quittierpflichtig ist oder nicht.

Beim erneuten Betätigen der Schaltfläche "Loop-In-Alarm" werden nur die für das Meldeereignis "Loop-In-Alarm" hinterlegten Systemfunktionen ausgeführt.

Hinweis

Wenn kein Loop-In-Alarm für die markierte Meldung projektiert ist, hat das Betätigen der Schaltfläche "Loop-In-Alarm" keine Auswirkung.

Siehe auch

Meldefenster, Meldeanzeige bedienen (Seite 956)

Einfache Meldeanzeige, Meldefenster bedienen (Seite 960)

Meldeanzeige (Seite 968)

Analogmeldungen projektieren (Seite 843)

Bitmeldungen projektieren (Seite 848)

3.2.2.8 Meldungen im Editor "HMI-Variablen"

Bitmeldungen projektieren

Einleitung

In WinCC können Sie auch im Editor "HMI-Variablen" Bit- und Analogmeldungen zusammen mit der Triggervariablen anlegen und bearbeiten.

Hinweis

Wenn Sie im Editor "HMI-Variablen" löschen, verschieben oder kopieren, wirken sich diese Änderungen auch im Editor "HMI-Meldungen" aus.

Voraussetzung

Der Editor "HMI-Variablen" ist geöffnet.

Vorgehen

Um eine Bitmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine Variable anzulegen, klicken Sie im oberen Teil des Arbeitsbereichs in der Tabelle auf "<Hinzufügen>".
   Eine neue Variable wird angelegt.
2. Projektieren Sie nach Bedarf eine interne oder externe Variable.
   - Wenn Sie eine HMI Variable wählen, verwenden Sie die Datentypen "Int" oder "UInt".
   - Wenn Sie eine PLC Variable wählen, verwenden Sie die Datentypen "Int" oder "Word". Der Ein- und Ausgangsbereich einer PLC-Variablen ist als Trigger ungeeignet.
3. Markieren Sie die Variable im oberen Teil des Arbeitsbereichs.
4. Klicken Sie im unteren Teil des Arbeitsbereichs auf der Registerkarte "Bitmeldungen" in der Tabelle auf "<Hinzufügen>".
   Eine neue Bitmeldung wird für die Variable angelegt. Wenn Sie den falschen Datentyp gewählt haben, wird die Variable in der Bitmeldung markiert.

5. Projektieren Sie im Inspektorfenster die Bitmeldung:
   - Geben Sie unter "Eigenschaften >Eigenschaften > Allgemein > Meldetext" den gewünschten Meldetext ein.
     Fügen Sie bei Bedarf Ausgabefelder in den Meldetext ein.
   - Wählen Sie eine Meldeklasse.
   - Wählen Sie unter "Eigenschaften > Eigenschaften > Trigger" das Triggerbit der Variablen, das die Bitmeldung auslöst.
6. Legen Sie bei Bedarf weitere Bitmeldungen zur Überwachung der Variablen an.

Hinweis

Eine Variable wird nur mit einem Meldeverfahren überwacht. Legen Sie also für eine Variable entweder Analogmeldungen **oder** Bitmeldungen an.

Ergebnis

Die projektierten Bitmeldungen sind im Editor "HMI-Variablen" angelegt und werden in den Editoren "HMI-Meldungen" und "HMI-Variablen" angezeigt.

Siehe auch

Bitmeldungen projektieren (Seite 830)

Parameter in Analogmeldung ausgeben (Seite 847)

Arbeiten mit Variablen (Seite 550)

Analogmeldungen projektieren

Einleitung

In WinCC legen Sie im Editor "HMI-Variablen" Bit- und Analogmeldungen zusammen mit der Triggervariablen an. Darüber hinaus bearbeiten Sie die Meldungen wie im Editor "HMI-Meldungen". Sie legen für eine Variable zwei Bereichswerte an, die Sie mit Analogmeldungen überwachen.

Voraussetzung

Der Editor "HMI-Variablen" ist geöffnet.

Vorgehen

Um eine Analogmeldung im Editor "HMI-Variablen" zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine Variable anzulegen, klicken Sie im oberen Teil des Arbeitsbereichs in der Tabelle auf "<Hinzufügen>".
   Eine neue Variable wird angelegt.
2. Projektieren Sie nach Bedarf eine interne oder externe Variable.
3. Projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Bereich" die Bereichswerte der Variablen:
   – Wählen Sie für die gewünschten Bereichswerte, ob eine "Konstante" oder "HMI_Variable" als Grenzwert verwendet wird. Wenn Sie "HMI_Variable" wählen, wird die Objektliste geöffnet. Wählen Sie die gewünschte Variable.

1. Klicken Sie im unteren Teil des Arbeitsbereichs auf die Registerkarte "Analogmeldungen". Legen Sie für beide Bereichswerte eine Analogmeldung an.
2. Markieren Sie eine Analogmeldung und konfigurieren Sie diese im Inspektorfenster:
   – Geben Sie unter "Eigenschaften > Eigenschaften > Allgemein > Meldetext" den gewünschten Meldetext ein.
   – Fügen Sie bei Bedarf Ausgabefelder in den Meldetext ein.
   – Ändern Sie bei Bedarf die voreingestellte Meldeklasse.
3. Projektieren Sie für Analogmeldungen weiter wie im Editor "HMI-Meldungen".
4. Fahren Sie fort, bis alle Analogmeldungen konfiguriert sind.

Hinweis

Eine Variable wird nur mit einem Meldeverfahren überwacht. Legen Sie also für eine Variable entweder Analogmeldungen **oder** Bitmeldungen an.

Ergebnis

Die projektierten Analogmeldungen sind im Editor "HMI-Variablen" angelegt und werden in den Editoren "HMI-Meldungen" und "HMI-Variablen" angezeigt.

Siehe auch

Parameter in Analogmeldung ausgeben (Seite 847)
Analogmeldungen projektieren (Seite 843)
Arbeiten mit Variablen (Seite 550)

3.2.3 Projektieren der Ausgabe von Meldungen

3.2.3.1 Übersicht zum Projektieren der Ausgabe von Meldungen

Arbeitsschritte zum Projektieren der Ausgabe von Meldungen

Sie projektieren die Ausgabe von Meldungen in WinCC in folgenden Arbeitsschritten:

1. Meldeanzeige erstellen
   Um Meldungen in Runtime anzuzeigen, verwenden Sie die Anzeige- und Bedienobjekte im Editor "Bilder". Auch für archivierte Meldungen projektieren Sie eine Meldeanzeige.
2. Quittierung projektieren
   Im Editor "Bilder" richten Sie ein, mit welcher Bedienaktion die Quittierung getriggert wird.
3. Protokollierung projektieren
   Um Meldungen in Runtime zu drucken, legen Sie Protokolle im Editor "Protokolle" an. Sie legen in den Editoren "Aufgabenplaner", "Bilder", "HMI-Meldungen" oder "HMI-Variablen" fest, wann und wie der Ausdruck eines Protokolls getriggert wird.

Hinweis

Geräteabhängigkeit

Die Protokollierung und die Archivierung stehen nicht für alle Bediengeräte zur Verfügung.

Hinweis

Beachten Sie, dass beim direkten Druck der Meldungen in Runtime nur ASCII-Zeichen unterstützt werden.

Weitergehende Projektierungsschritte

Abhängig von den Anforderungen Ihres Projekts sind weitere Arbeitsschritte zum Projektieren der Meldeanzeige erforderlich:

1. Berechtigungen einrichten
   Um sicherzustellen, dass nur berechtigte Benutzer die Meldungen bearbeiten, vergeben Sie Berechtigungen für die Meldeanzeige und die Funktionstasten des Bediengeräts.
2. Filtern der Meldeanzeige projektieren
   Sie projektieren im Editor "Bilder" das Filtern der Meldungen in Runtime. Sie projektieren bei Bedarf auch Meldeanzeigen, die nur ausgewählte Meldungen anzeigen.
3. Bedienmeldungen projektieren
   Projektieren Sie im Editor "Bilder" an die Bedienelemente des Anzeige- und Bediengeräts Bedienmeldungen. Bei einer Bedienaktion wird eine vorkonfigurierte Bedienmeldung ausgegeben. Eine Bedienaktion ist z. B. die Quittierung einer Meldung.

Siehe auch

Projektieren der Meldeanzeige (Seite 870)

Projektieren der Meldeanzeige (Seite 886)

Protokollieren von Meldungen (Seite 909)

3.2.3.2 Projektieren der Meldeanzeige

Anzeige der Meldungen

Möglichkeiten zur Anzeige von Meldungen am Bediengerät

Um Meldungen am Bediengerät anzuzeigen, bietet WinCC folgende Möglichkeiten:

- Meldeanzeige
  Die Meldeanzeige projektieren Sie in einem Bild. Abhängig von ihrer projektierten Größe zeigt sie mehrere Meldungen gleichzeitig an. Sie projektieren mehrere Meldeanzeigen mit unterschiedlichen Inhalten.
- Meldefenster
  Das Meldefenster projektieren Sie im Editor "Globales Bild". Abhängig von seiner projektierten Größe zeigt das Meldefenster mehrere Meldungen gleichzeitig an. Das Meldefenster wird bei Bedarf ereignisgesteuert geschlossen und wieder geöffnet. Um es bei der Projektierung gezielt auszublenden, legen Sie ein Meldefenster in einer eigenen Ebene an.

Zusätzliche Signale

- Meldeindikator
  Der Meldeindikator ist ein projektierbares grafisches Symbol. Wenn eine Meldung kommt, wird der Meldeindikator am Bediengerät angezeigt. Sie projektieren den Meldeindikator im Editor "Globales Bild".
  Der Meldeindikator hat zwei Zustände:
  - Blinkend: Mindestens eine quittierpflichtige Meldung steht an.
  - Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen.
    Abhängig vom Bediengerät zeigt der Meldeindikator auch die Anzahl anstehender Meldungen an.
- Benachrichtigung über E-Mail
  Um beim Kommen von Meldungen einer bestimmten Meldeklasse außer dem Benutzer z. B. einen Techniker zu informieren, ordnen Sie der Meldeklasse eine E-Mail-Adresse zu.

  Hinweis

  Geräteabhängigkeit

  Die Benachrichtigung über E-Mail steht nicht für alle Bediengeräte zur Verfügung.

- Systemfunktionen
  Sie projektieren an ein Ereignis einer Meldung eine Funktionsliste, die beim Eintreten des Ereignisses in Runtime abgearbeitet wird.
  Um die Meldeanzeige oder das Meldefenster nicht über die Symbolleiste zu bedienen, verwenden Sie in WinCC Systemfunktionen für Meldungen.

Anzeige der vordefinierten Meldeklassen in Runtime

Die folgende Tabelle zeigt die Symbole, mit der die vordefinierten Meldeklassen in der Meldeanzeige dargestellt werden:

Meldeklasse	Angezeigtes Symbol
"Diagnosis Events"	S7
"Errors"	!
"System"	$
"Warnings"	<kein Symbol>

Siehe auch

Systemfunktionen für Meldungen (Seite 993)

Meldeanzeige projektieren

Meldeanzeige für aktuelle Meldungen projektieren

Einleitung

Aktuelle Meldungen werden in Runtime in einer Meldeanzeige oder einem Meldefenster angezeigt.

Voraussetzung

- Im Editor "Bilder" ist ein Bild geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

Meldungen für die Meldeanzeige projektieren

Um festzulegen, welche Meldungen in der Meldeanzeige angezeigt werden, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldeanzeige" in das Bild ein.
2. Markieren Sie die Meldeanzeige.
   – Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Aktuelle Meldezustände".
     Legen Sie bei Bedarf fest, ob quittierpflichtige und/oder Meldungen ohne Quittierpflicht angezeigt werden.

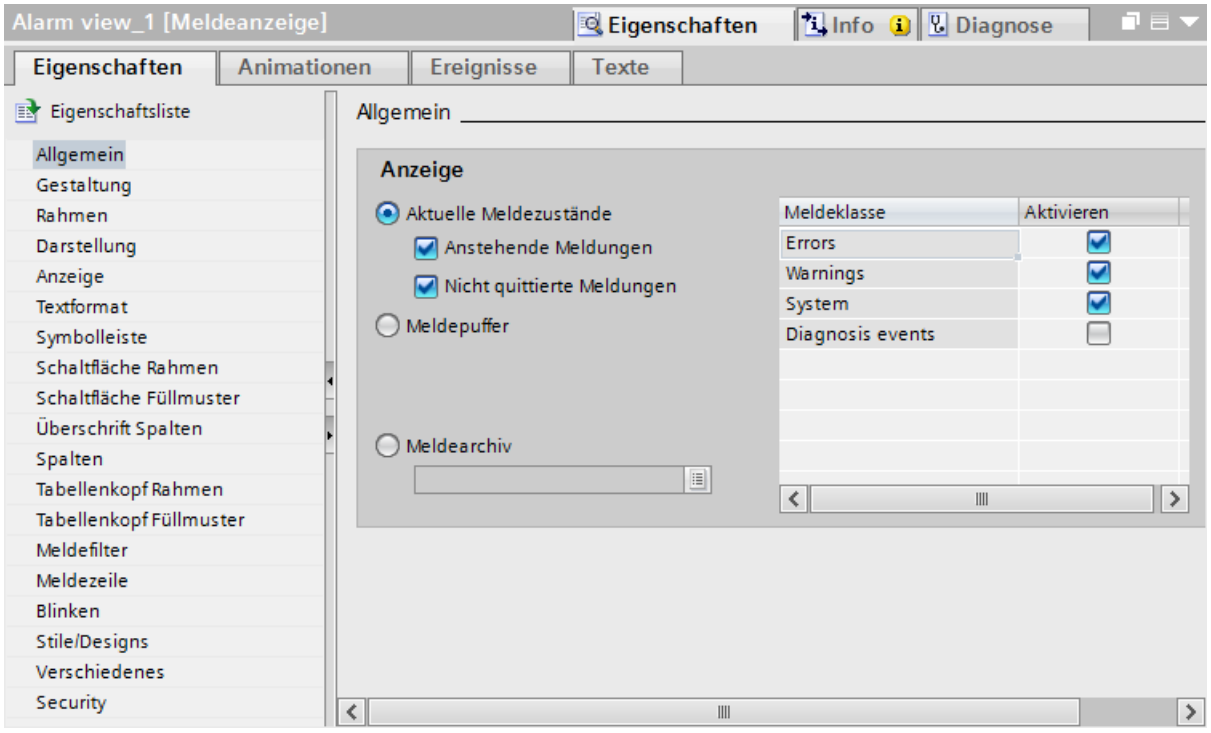

– Um alle Meldungen anzuzeigen, die sich im Meldepuffer befinden, aktivieren Sie "Meldepuffer".

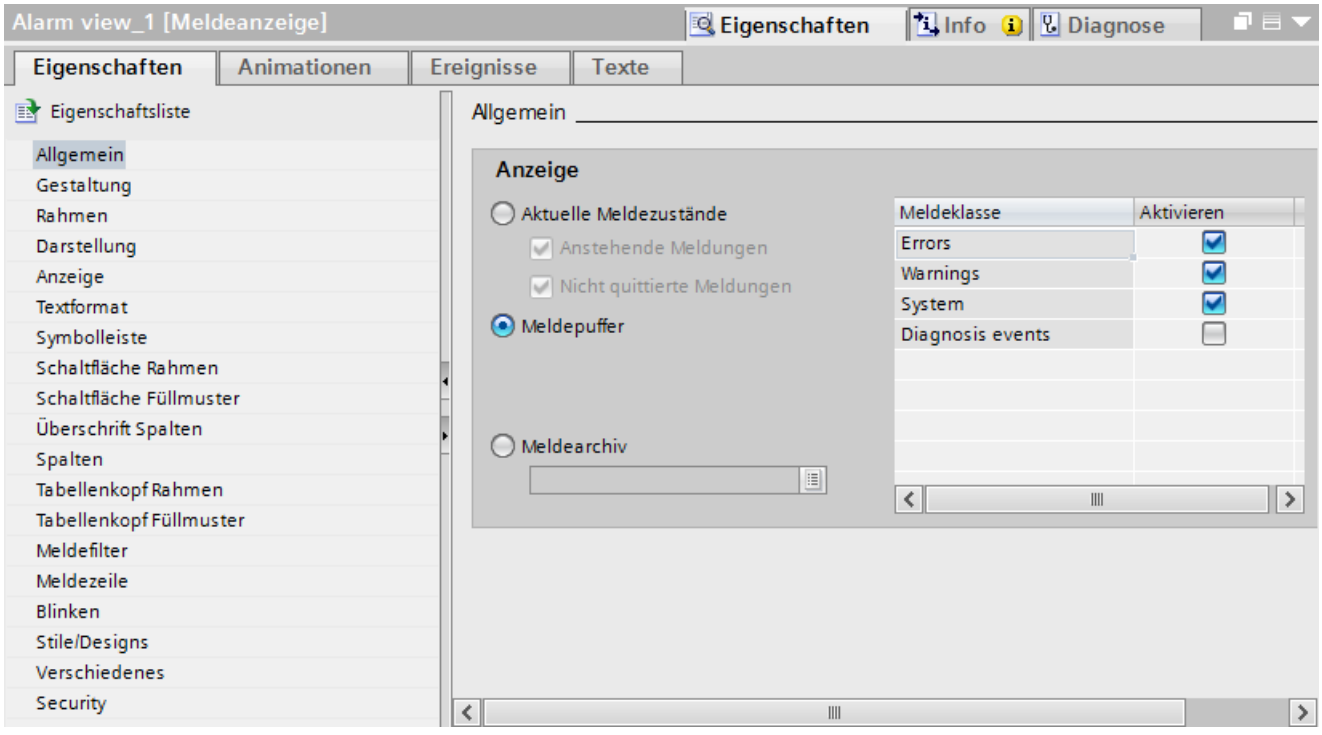

3. Aktivieren Sie in der Tabelle die Meldeklassen, die in der Meldeanzeige angezeigt werden.
4. Legen Sie unter "Eigenschaften > Eigenschaften > Anzeige > Steuervariable für den Anzeigebereich" die Variable fest, die das Datum liefert, ab dem die Meldungen angezeigt werden.

Hinweis

Geräteabhängigkeit

Die Eigenschaft "Steuervariable für Anzeigebereich" steht nicht für alle Bediengeräte zur Verfügung.

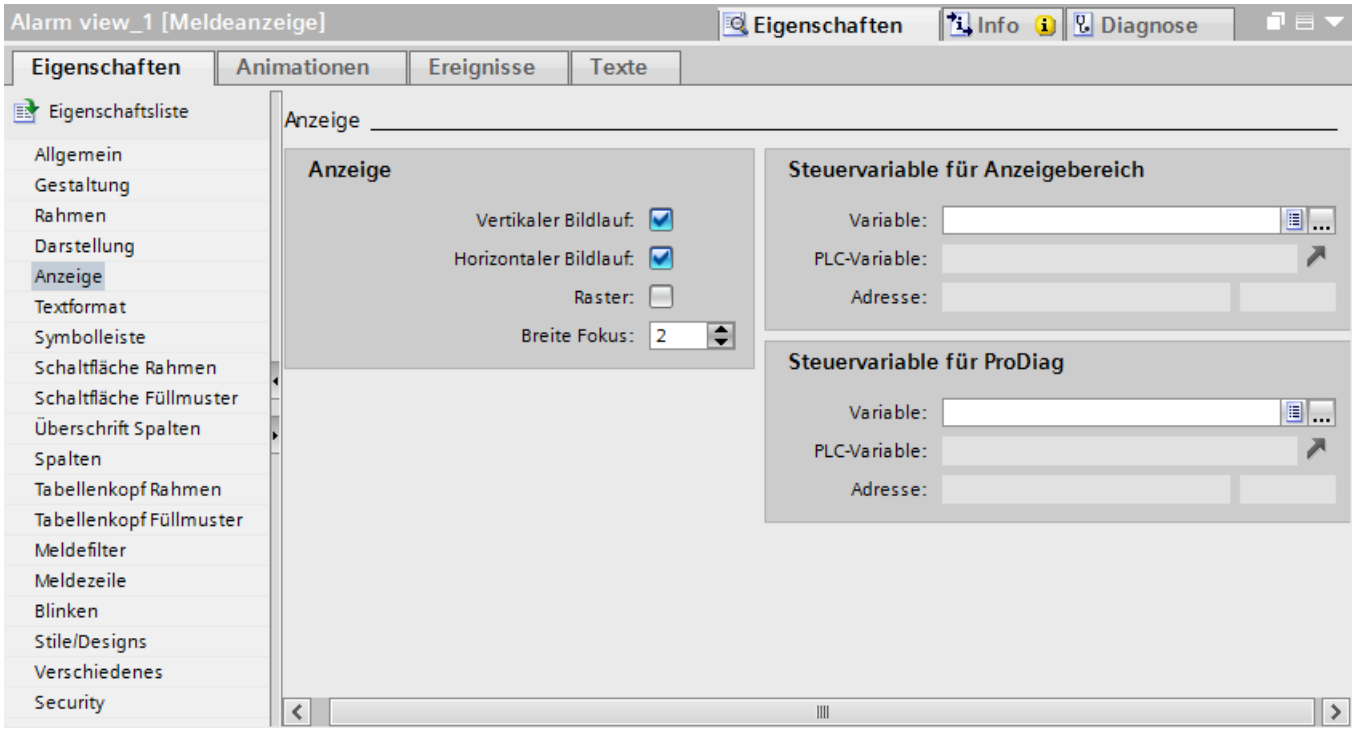

Darstellung der Meldeanzeige projektieren

Um festzulegen, wie die Meldungen in der Meldeanzeige dargestellt werden, gehen Sie folgendermaßen vor.

1. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Darstellung > Einstellungen > Zeilen pro Meldung" die Anzahl der Zeilen fest, mit der jede Meldung angezeigt wird.
2. Wählen Sie unter "Eigenschaften > Eigenschaften > Anzeige" die Bedienelemente, die am Bediengerät zur Verfügung stehen.
3. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Spalten" die Spalten:
   - Wählen Sie unter "Sichtbare Spalten" die Spalten, die in der Meldeanzeige ausgegeben werden.
   - Legen Sie unter "Eigenschaften Spalten" die Eigenschaften der Spalten fest.
   - Wählen Sie unter "Sortieren" die Sortierreihenfolge der Meldungen.

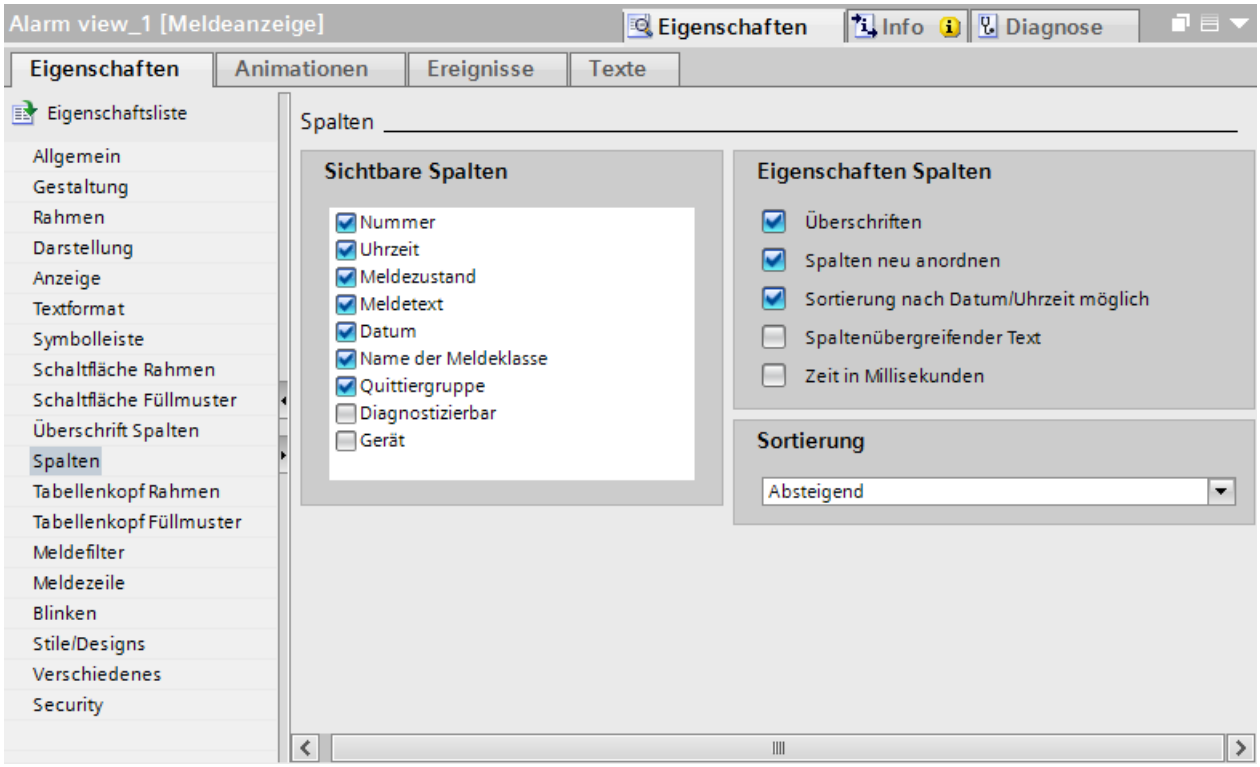

Hinweis

Wenn Sie im Kontextmenü der Meldeanzeige den Befehl "Bearbeiten" wählen, wird die Meldeanzeige aktiv geschaltet. Im aktiven Modus stellen Sie die Spaltenbreite und die Position der Spalten ein. Stellen Sie zur Aktivierung der Meldeanzeige den Zoomfaktor auf 100 %.

Die Meldeanzeige wird deaktiviert, sobald Sie das Objekt im Bild abwählen.

Ergebnis

In Runtime werden in der Meldeanzeige Meldungen verschiedener Meldeklassen ausgegeben.

Siehe auch

Meldeklassen anlegen (Seite 818)

Meldefenster projektieren (Seite 879)

Archivierte Meldungen anzeigen (Seite 885)

Meldeanzeige für S7 Diagnosemeldungen projektieren

Einleitung

Um S7 Diagnosemeldungen anzuzeigen, projektieren Sie eine Meldeanzeige oder ein Meldefenster für die vordefinierte Meldeklasse "Diagnosis Events". Dazu aktivieren Sie die Diagnosemeldungen in den Runtime-Einstellungen Ihres Bediengeräts.

Hinweis

Geräteabhängigkeit

- Die Meldeklasse "Diagnosis Events" steht nicht für alle Bediengeräte zur Verfügung.
- Systemdefinierte Steuerungsmeldungen vom Typ "SIMATIC S7 Diagnosemeldung" können für Ihr Bediengerät nur für die Verbindungen mit einer Steuerung SIMATIC S7-300/400 aktiviert werden.

Hinweis

Verwendung von System-Diagnoseanzeige und/oder System-Diagnosefenster

Sobald eine System-Diagnoseanzeige oder ein System-Diagnosefenster im HMI-Projekt vorhanden sind, werden die Meldetexte für die Bildobjekte zur Systemdiagnose heruntergeladen und dort verwendet. Deshalb hat die Option "Runtime-Einstellungen > Meldungen > Systemmeldungen > Zeige Meldetext" bei Verwendung von System-Diagnoseanzeige oder System-Diagnosefenster keine Auswirkung und gilt immer als aktiviert. Die Größe der Runtime-Datei kann in diesem Fall nicht durch Deaktivieren der Option verringert werden.

Voraussetzung

- Eine Meldeanzeige oder ein Meldefenster ist im Editor "Bilder" projektiert
- Eine Verbindung zu einer Steuerung SIMATIC S7-300/400 ist angelegt.

Anzeige von S7 Diagnosemeldungen projektieren

Um die Anzeige von S7 Diagnosemeldungen zu projektieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Meldungen".
2. Aktivieren Sie unter "Systemmeldungen" das Optionskästchen "S7 Diagnosemeldungen (Nummer)".
3. Um die S7 Diagnosemeldungen mit Text anzuzeigen, aktivieren Sie "Zeige Meldetext".
4. Öffnen Sie das Bild mit der Meldeanzeige und markieren Sie die Meldeanzeige.
5. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Meldeklasse "Diagnosis Events".

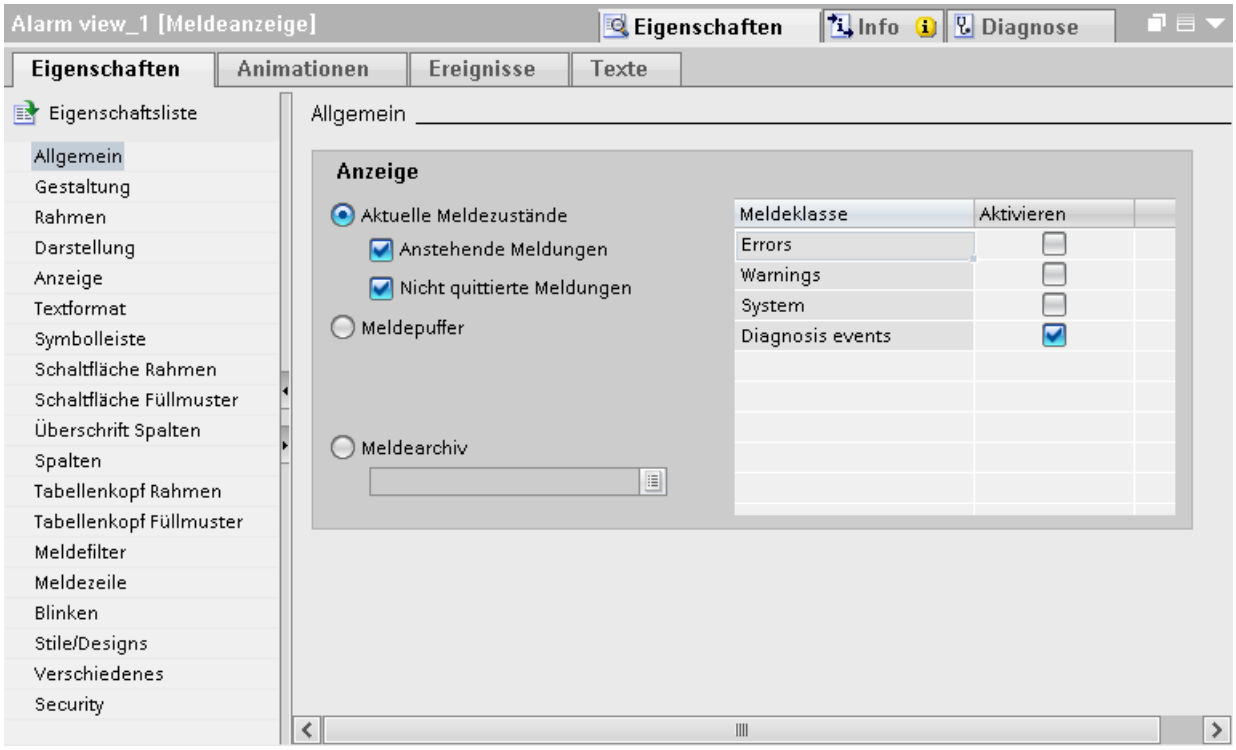

6. Projektieren Sie die Meldeanzeige weiter wie für die Anzeige aktueller Meldungen.

Ergebnis

In Runtime werden in der Meldeanzeige die S7 Diagnosemeldungen der Steuerung angezeigt.

Meldeanzeige für archivierte Meldungen projektieren

Einleitung

Archivierte Meldungen zeigen Sie in Runtime in einer Meldeanzeige oder einem Meldefenster an.

Voraussetzung

- Eine Meldeanzeige oder ein Meldefenster ist im Editor "Bilder" projektiert
- Ein Meldearchiv ist im Editor "Archive" angelegt.
- Meldungen sind im Editor "HMI-Meldungen" als"archivierbar" projektiert.

Meldeanzeige für archivierte Meldungen projektieren

Um die Anzeige archivierter Meldungen in einer Meldeanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Bild mit der Meldeanzeige und markieren Sie die Meldeanzeige.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Meldearchiv".
3. Klicken Sie auf die Schaltfläche "..." und wählen Sie das Meldearchiv.
4. Projektieren Sie die Meldeanzeige weiter wie für die Anzeige aktueller Meldungen.

Ergebnis

In Runtime werden in der Meldeanzeige die archivierten Meldungen ausgegeben.

Meldefenster projektieren

Einleitung

Das Meldefenster zeigt aktuelle Meldungen an. Das Meldefenster wird im Editor "Globales Bild" projektiert Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.. Auch wenn Meldungen anstehen und angezeigt werden, bleibt das Bediengerät bedienbar. Ein Meldefenster wird dargestellt und projektiert wie eine Meldeanzeige.

Um Meldefenster bei der Projektierung gezielt auszublenden, legen Sie diese in einer eigenen Ebene an.

Voraussetzung

- Der Editor "Globales Bild" ist geöffnet.
- Die Task Card "Werkzeuge" wird angezeigt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein Meldefenster zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldefenster" in das Globale Bild ein.
2. Projektieren Sie das Meldefenster wie eine Meldeanzeige.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Modus > Fenster" wie sich das Meldefenster in Runtime verhält und bedient wird.
   – Wenn das Meldefenster in Runtime bei einem Bildwechsel den Fokus behalten soll, aktivieren Sie "Gebunden".
     Diese Option ist wichtig, da das Hin- und Herschalten zwischen Bild und verschiedenen Fenstern mit <Strg+Tab> nicht unterstützt wird.

Ergebnis

In Runtime werden die Meldungen der gewählten Meldeklasse im Meldefenster angezeigt.

Siehe auch

Meldeanzeige für aktuelle Meldungen projektieren (Seite 872)

Meldeklassen anlegen (Seite 818)

Meldeindikator projektieren

Einleitung

Der Meldeindikator zeigt mit einem Warndreieck an, dass Meldungen anstehen oder quittiert werden müssen. Wenn eine Meldung einer mit Meldeindikator projektierten Meldeklasse kommt, wird der Meldeindikator eingeblendet.

Der Meldeindikator hat zwei Zustände:

- Blinkend: Mindestens eine quittierpflichtige Meldung steht an.
- Statisch: Mindestens eine der quittierten Meldungen ist noch nicht gegangen.

Sie legen bei der Projektierung bei Bedarf fest, dass in Runtime das Meldefenster geöffnet wird, wenn der Benutzer den Meldeindikator bedient.

Voraussetzung

- Der Editor "Globales Bild" ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um den Meldeindikator zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldeanzeige" in den Arbeitsbereich ein.
2. Markieren Sie den Meldeindikator.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklassen, die durch die Meldeanzeige angezeigt werden.
   Geben Sie an, ob anstehende und/oder quittierte Meldungen mit Meldenzeige angezeigt werden.

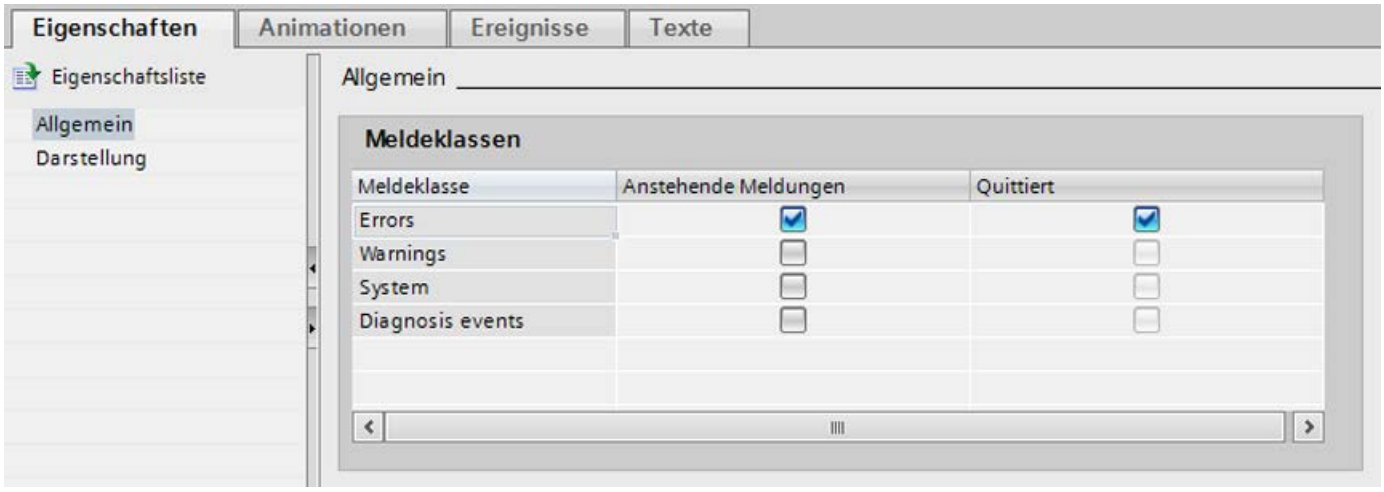

4. Projektieren Sie unter "Eigenschaften > Ereignis" die Systemfunktion "ZeigeMeldefenster" an ein Ereignis der Meldeanzeige.

Hinweis

Wenn Sie im Bild oder der Vorlage einen Permanentbereich projektiert haben, positionieren Sie Meldefenster und Meldeanzeige nicht im Bereich des Permanentbereichs. Das Meldefenster und die Meldeanzeige werden sonst in Runtime nicht angezeigt. Den Permanentbereich sehen Sie im Editor "Globales Bild" jedoch nicht.

Ergebnis

Wenn in Runtime Meldungen der ausgewählten Meldeklasse anstehen oder quittiert werden müssen, wird die Meldeanzeige angezeigt. Wenn der Benutzer die Meldeanzeige bedient, wird das Meldefenster geöffnet.

Filtern von Meldungen projektieren

Einleitung

Sie filtern bei Bedarf die Anzeige von Meldungen in der erweiterten Meldeanzeige. Das Kriterium ist immer eine Zeichenkette. Folgende zwei Möglichkeiten zum Filtern der Meldeanzeige stehen zur Verfügung:

- In WinCC ist ein festes Kriterium projektiert.
  In Runtime werden alle Meldungen angezeigt, welche die vollständige Zeichenkette im Meldetext enthalten.
- Der Benutzer filtert die Meldeanzeige in Runtime.
  Über eine Filtervariable wird die Meldeanzeige in Runtime über ein E/A-Feld mit der gewünschten Zeichenkette versorgt, z. B. wie nachstehend beschrieben.
  Der Filter wirkt auf die Darstellung in der Meldeanzeige. Alle Meldungen im Meldepuffer bleiben erhalten.

Voraussetzung

- Eine Meldeanzeige ist projektiert.
- Das Bild mit der Meldeanzeige ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Filter für eine feststehende Zeichenkette projektieren

1. Markieren Sie die Meldeanzeige.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Meldefilter".
3. Geben Sie unter "Filterzeichenfolge" die gewünschte Zeichenkette für den Filter ein.

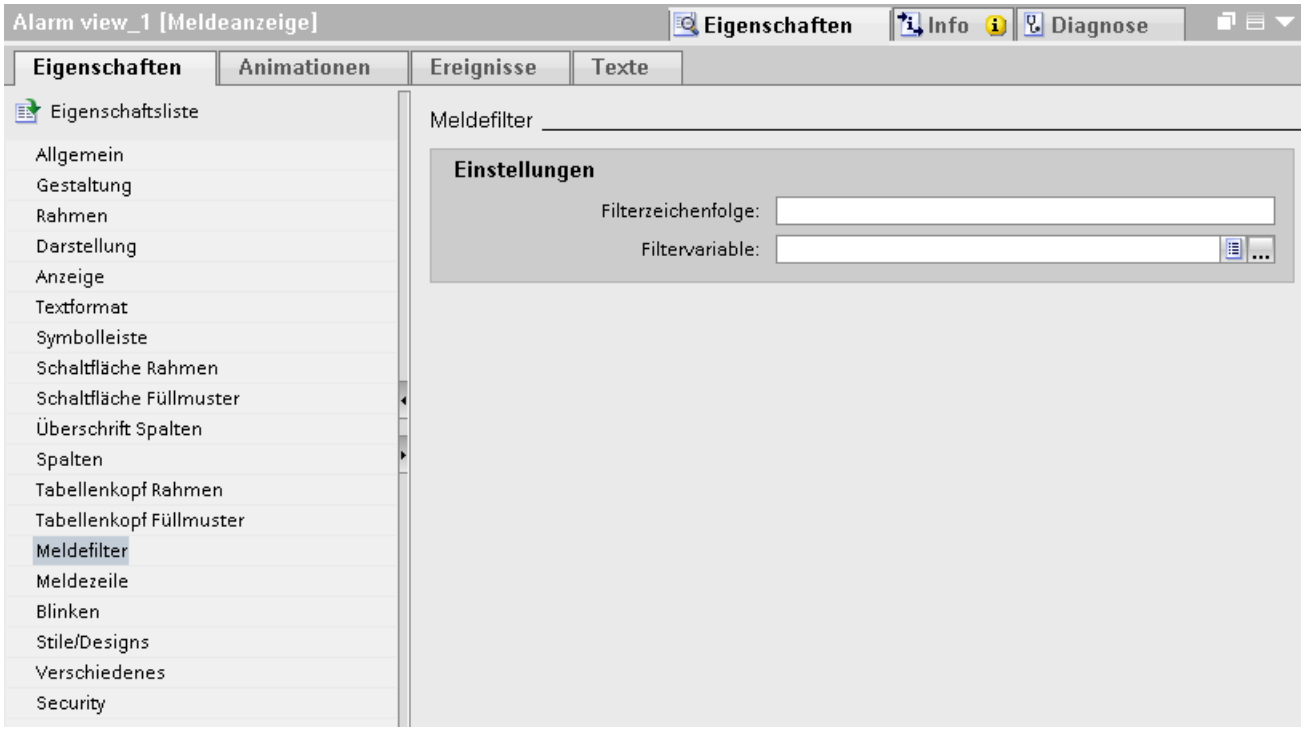

Ergebnis

In Runtime werden nur die Meldungen angezeigt, die im Meldetext die vollständige Zeichenkette aus dem Filter enthalten.

Filter für eine veränderbare Zeichenkette projektieren

1. Markieren Sie die Meldeanzeige.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Filter".

3. Wählen Sie unter "Filtervariable" eine Variable aus. Die Variable muss vom Typ "String" sein.
4. Projektieren Sie im Bild ein E/A-Feld, in das in Runtime eine Zeichenkette als Filter eingegeben wird.
   – Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Format > Anzeigeformat > Zeichenkette".
   – Um das E/A-Feld mit der Meldeanzeige zu verbinden, wählen Sie unter "Prozess" die Filtervariable aus, die Sie für die Meldeanzeige ausgewählt haben.

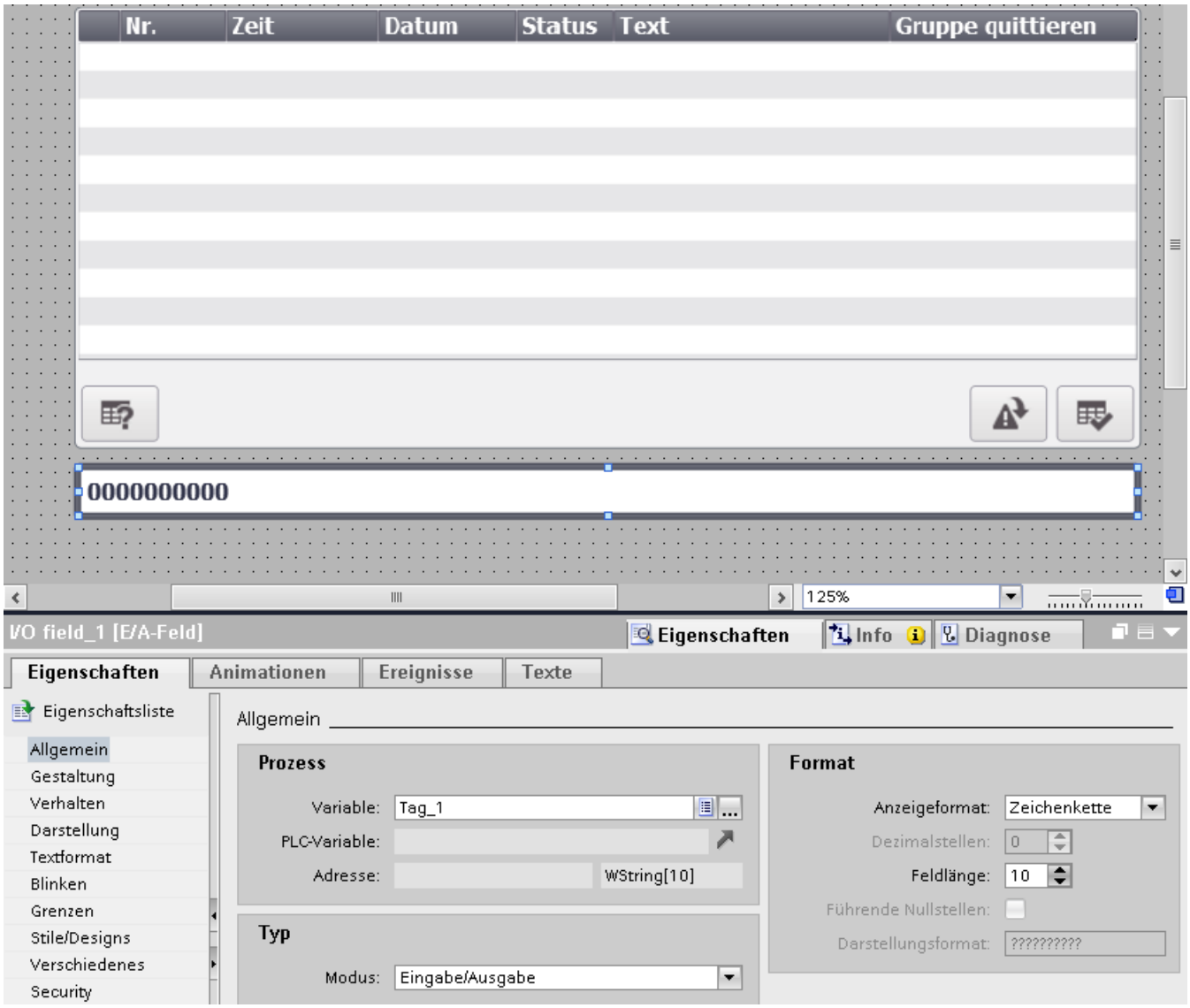

Ergebnis

Wenn Sie in Runtime eine Zeichenkette in das E/A-Feld eingeben, zeigt die Meldeanzeige nur Meldungen an, die im Meldetext die vollständige Zeichenkette enthalten.

Wenn eine fest projektierte Zeichenkette und eine Filtervariable projektiert wurden, werden die Meldungen in Runtime nach dem Inhalt der Filtervariablen gefiltert. Wenn die Filtervariable leer ist, wird nach der fest projektierten Zeichenkette gefiltert.

Hinweis

Für folgende Meldeanzeigen ist eine Filterung nicht möglich:

- Einfache Meldeanzeige
- Meldezeile
- Meldefenster
- Meldeanzeige für archivierte Meldungen

Siehe auch

Arbeiten mit Variablen (Seite 550)

Archivierte Meldungen anzeigen

Einleitung

Archivierte Meldungen zeigen Sie in Runtime in einer Meldeanzeige an. Die Meldeanzeige zeigt den Inhalt eines Meldearchivs an.

Verwendung

Ein Benutzer informiert sich über den abgelaufenen Prozess, z. B. am Ende seiner Schicht.

Voraussetzung

- Ein Meldearchiv ist erstellt.
- Ein Bild geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Meldeanzeige für die Ausgabe eines Meldearchivs zu projektieren, gehen Sie folgendermaßen vor:

1. Projektieren Sie in einem Bild eine Meldeanzeige.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Meldearchiv" das gewünschte Meldearchiv aus.
   Bei Bedarf legen Sie ein neues Meldearchiv an.

Ergebnis

In Runtime werden in der Meldeanzeige die archivierten Meldungen des gewählten Meldearchivs angezeigt.

Hinweis

In Runtime werden nicht die archivierten Meldetexte angezeigt, sondern die im aktuellen Projekt vorhandenen Meldetexte. Die archivierten Meldetexte sind nur zur externen Auswertung der Archivdatei vorgesehen.

Wenn das angezeigte Meldearchiv mit einer anderen Projektierung erstellt wurde, korrespondieren die angezeigten Meldetexte nicht mit den archivierten.

Siehe auch

Meldeanzeige für aktuelle Meldungen projektieren (Seite 872)

3.2.3.3 Projektieren der Meldeanzeige

Anzeige der Meldungen

Möglichkeiten zur Anzeige von Meldungen am Bediengerät

Um Meldungen am Bediengerät anzuzeigen, bietet WinCC folgende Möglichkeiten:

- Meldeanzeige für anstehende und gegangene Meldungen
  Die Meldeanzeige wird für ein Bild projektiert. Abhängig von ihrer projektierten Größe zeigt sie mehrere Meldungen gleichzeitig an. Sie projektieren bei Bedarf mehrere Meldeanzeigen mit unterschiedlichen Inhalten und in verschiedenen Bildern. Um gegangene Meldungen anzuzeigen, verwenden Sie die Anzeigen "Kurzzeitarchiv" und "Langzeitarchiv".
- Meldeanzeige für statistische Werte
  Die Anzeige "Meldestatistik" der Meldeanzeige visualisiert statistische Informationen, z. B. Häufigkeit und Anzeigedauer von Meldungen.
- Zustandsvariable einer Meldegruppe
  Über die Zustandsvariable einer Meldegruppe visualisieren Sie die wechselnden Meldezustände einer Meldegruppe. Sie erstellen deren Darstellung individuell im Editor "Bilder".

Zusätzliche Signale

- Zentraler Melder
  WinCC zeigt eine Meldung zusätzlich zur Meldeanzeige durch eine optische oder akustische Meldeanzeige in der Anlage an. Der zentrale Melder kann eine Hupe oder eine Warnleuchte sein.
- Systemfunktionen und Skripte
  Sie projektieren an ein Ereignis einer Meldung eine Funktionsliste, die beim Eintreten des Ereignisses in Runtime abgearbeitet wird.
  Um die Meldeanzeige nicht über die Symbolleiste zu bedienen, verwenden Sie in WinCC Systemfunktionen für Meldungen.

Siehe auch

Bitbelegung Zustandsvariable (Seite 1063)

Variablen einer Meldegruppe (Seite 811)

Systemfunktionen für Meldungen (Seite 993)

Meldeanzeige projektieren

Einleitung

In der Meldeanzeige werden aktuelle oder archivierte Meldungen angezeigt. In einer Meldeanzeige werden bei Bedarf Meldungen aller Meldeklassen angezeigt. Sie projektieren Kriterien, nach denen die Meldungen gefiltert werden.

Voraussetzung

- Ein Bild ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

Vorgehen

Um eine Meldeanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldeanzeige" in das Bild ein.
2. Markieren Sie die Meldeanzeige und klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
   - Wählen Sie unter "Anzeige > Typ" die Art von Meldungen aus, die als Standard in der Meldeanzeige dargestellt werden.
   - Wählen Sie unter "Anzeige > Anzeige", welche Meldungen angezeigt werden. Dabei blenden Sie die Meldungen, deren Anzeige unterdrückt wurde, ein oder aus.
   - Wenn der Bediener in Runtime auf eine Meldung doppelklickt, wird eine Aktion gestartet. Diese Aktion legen Sie unter "Anzeige > Doppelklick" fest. Wenn Sie den Loop-In-Alarm nicht nur über die Schaltfläche "Loop-In-Alarm" auslösen wollen, wählen Sie "Loop-In-Alarm".

– Wählen Sie unter "Zeitbasis" die gewünschte Zeitbasis.

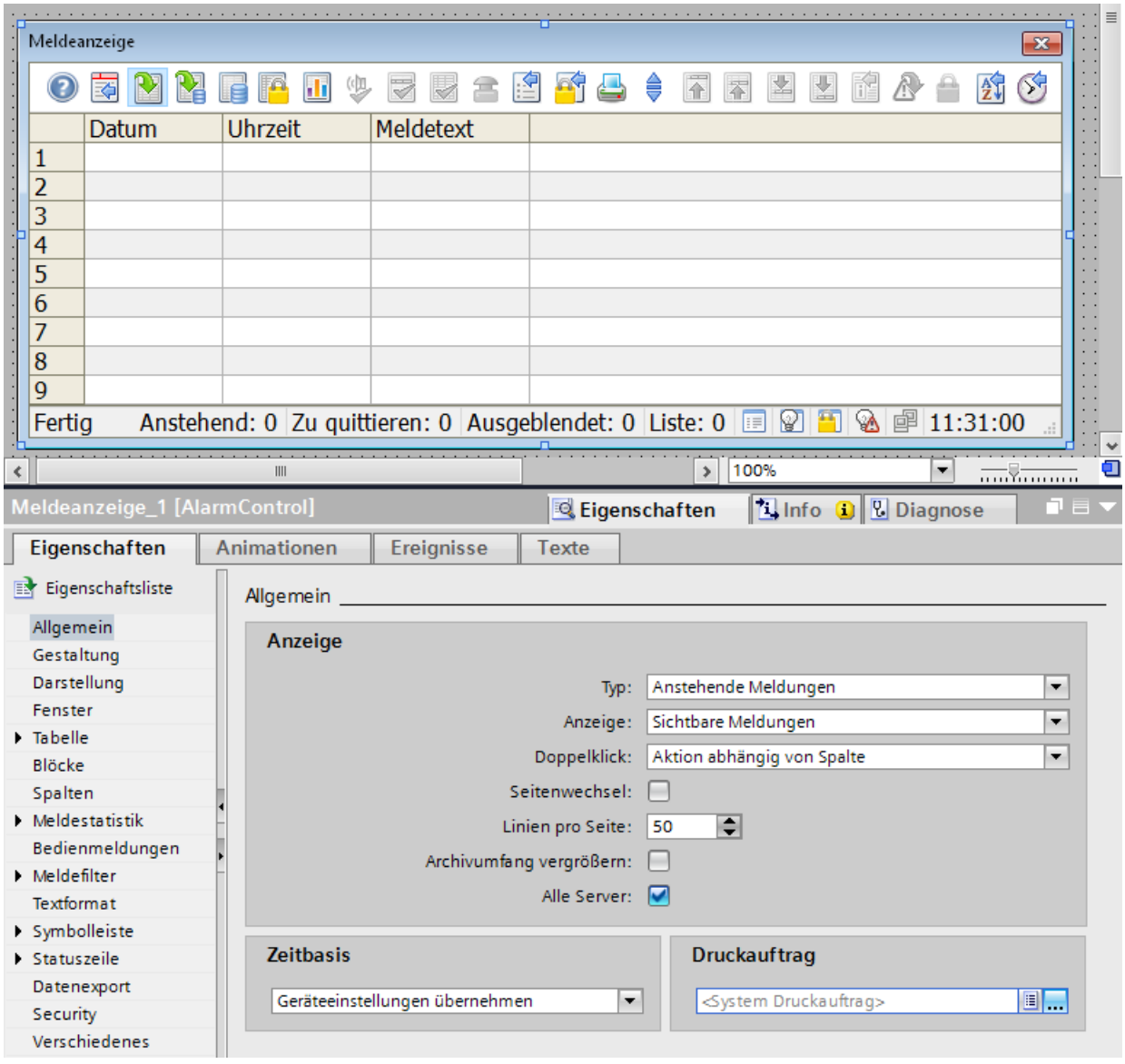

3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Gestaltung" die Gestaltung der Meldeanzeige fest.
4. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Tabelle" die Gestaltung der Kopfzeilen, Inhalte und des Tabellengitters innerhalb der Meldeanzeige fest.
   – Legen Sie unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Markierung" die Darstellung markierter Meldungen in Runtime fest.
   – Legen Sie unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Sortierung" die Methode zum Sortieren der Meldungen in Runtime fest.
5. Legen Sie unter "Eigenschaften > Eigenschaften > Meldeliste" die Sortierreihenfolge der Meldungen in Runtime fest.

Meldeblöcke für die Meldeanzeige aktivieren

1. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Blöcke > Projekteinstellungen für Meldeblöcke übernehmen", ob Sie die Einstellungen aus der Registerkarte "Meldeblöcke" im Editor "HMI-Meldungen" für diese Meldeanzeige übernehmen.
2. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Blöcke" die Meldeblöcke, die in der Meldeanzeige dargestellt werden. Für jeden Meldeblock legen Sie bei Bedarf folgende Eigenschaften fest:
   - Name
   - Darstellung der Kopfzeile
   - Darstellung der Inhalte
   - Ausrichtung
   - Länge
   - Format
   - Blinken

Die Darstellung der Inhalte ist sowohl als Text als auch als Symbol möglich. Die Darstellung als Symbol steht nicht für alle Meldeblöcke zur Verfügung.

Symbol- und Statusleiste der Meldeanzeige projektieren

1. Legen Sie unter "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Allgemein" die Darstellung der Symbolleiste fest.
2. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Bedienelemente" die Bedienelemente, die in der Symbolleiste der Meldeanzeige enthalten sind.
3. Legen Sie unter "Eigenschaften > Eigenschaften > Statusleiste > Statusleiste - Allgemein" die Darstellung der Statusleiste fest.
4. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Statusleiste > Statusleiste - Bedienelemente", die Informationen, die in der Statusleiste der Meldeanzeige angezeigt werden.

Ergebnis

In Runtime werden in der Meldeanzeige Meldungen verschiedener Meldeklassen ausgegeben. Um in Runtime die Anzeige zu wechseln, klicken Sie auf die projektierten Schaltflächen der Symbolleiste in der Meldeanzeige.

Siehe auch

Filter in der Meldeanzeige projektieren (Seite 891)

Meldestatistik projektieren (Seite 892)

Filter in der Meldeanzeige projektieren

Einleitung

WinCC bietet die Möglichkeit, verschiedene Filter für die Meldeanzeige zu projektieren:

- Standardfilter
  Der Bediener hat in Runtime keinen Zugriff auf einen Standardfilter. Einen projektierten Standardfilter kopieren Sie bei Bedarf und verwenden ihn in anderen Meldeanzeigen.
- Benutzerdefinierter Filter
  Der Bediener ändert in Runtime bei Bedarf den benutzerdefinierten Filter. Sie verwenden einen benutzerdefinierten Filter nur in einer einzelnen Meldeanzeige.

Voraussetzung

- Das Bild mit der projektierten Meldeanzeige ist im Editor "Bilder" geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Markieren Sie die Meldeanzeige und klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Filter > Meldefilter".
2. Um einen benutzerdefinierten Filter zu projektieren, klicken Sie auf die Registerkarte "Benutzerdefiniert".
3. Definieren Sie ein oder mehrere Filter. Für jeden Filter legen Sie unter "Ausdruck" ein oder mehrere Filterkriterien mit Bedingung und Wert fest.

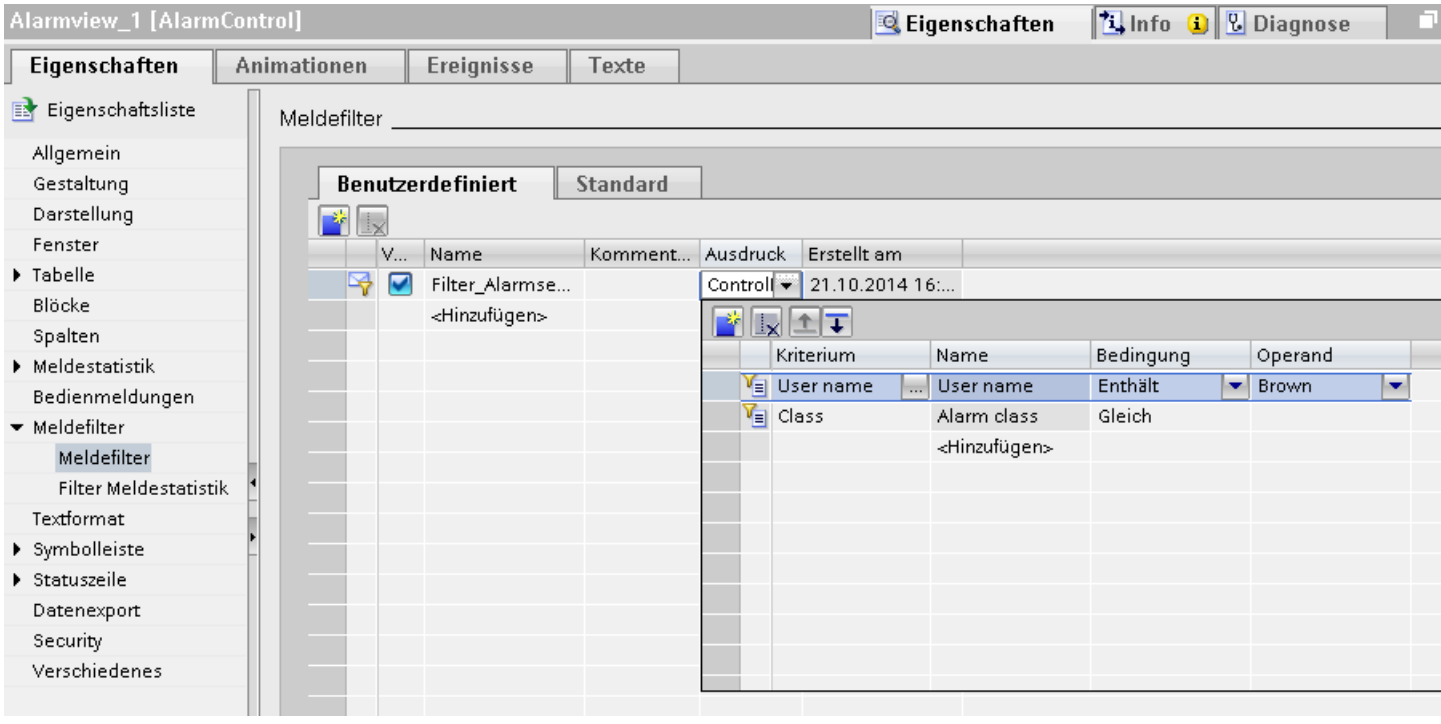

4. Um einen Standardfilter zu projektieren, wählen Sie die Registerkarte "Standard". Sie projektieren mehrere Standardfilter und aktivieren Sie nach Bedarf für die jeweilige Meldeanzeige.
5. Geben Sie dazu einen Namen für den Standardfilter ein und bei Bedarf einen Kommentar.
6. Um den Filter zu konfigurieren, wählen Sie unter "Filter" die Filterkriterien, die Filterbedingungen und die Werte für den Standardfilter.

Filtern nach Prozesswerten

Die Eingabe von Text als Kriterium für einen Prozesswert ist nicht möglich. Wenn Sie nach dem Text eines Parameterblocks oder der dargestellten Prozessvariablen filtern wollen, gehen Sie folgendermaßen vor:

1. Fügen Sie diesen Prozesswert als Meldetext eines Anwendertextblocks in eine Meldung ein.
2. Filtern Sie in Runtime über die Schaltfläche die Meldungen nach dem Text im Meldeblock "Meldetext".

Achten Sie in Mehrplatzsystemen darauf, dass die bei einem Client im Dialog "Selektion festlegen" zur Anzeige kommenden Inhalte auf allen Servern gleich benannt werden.

Filtern nach Uhrzeit

Wenn die Zeitbasis der Meldeanzeige verändert wird, werden beim Filtern nach der Uhrzeit Startwert und Stoppwert nicht automatisch angepasst.

Beispiel

Am Standort des PC mit der Zeitzone "UTC + 1h" hat die Meldeanzeige die Zeitbasis "lokale Zeitzone". Sie filtern nach der Uhrzeit 10:00 bis 11:00 Uhr. Sie stellen die Zeitbasis auf "UTC" um. Um die gleichen Meldungen wie zuvor anzeigen zu lassen, ändern Sie den Startwert oder Stoppwert des Filters auf 9:00 und 10:00 Uhr.

Siehe auch

SQL-Statements zum Filtern der Meldeanzeige (Seite 1000)

Meldestatistik projektieren

Einleitung

Die Meldestatistik stellt statistische Berechnungen zu archivierten Meldungen in der Meldeanzeige dar. Die Meldestatistik zeigt auch eine Auswahl der projektierten Meldeblöcke an. Bei dynamischen Inhalten zeigen die Meldeblöcke die Daten der letzten gekommenen Meldung an. Die Spalten der Meldestatistik stellen Sie individuell zusammen.

Die Meldestatistik zeigt folgende statistische Berechnungen an:

- Häufigkeit einer Meldung
  Das System zählt, wie oft eine Meldung mit dem Status "Gekommen" im Archiv enthalten ist. Wenn die Meldenummer nicht vorkommt, wird diese Meldenummer in die Statistik nicht eingetragen.
- Gesamte Anzeigedauer einer Meldung in Sekunden
  Als Berechnungsgröße stehen Ihnen folgende Zeiträume zwischen den Meldezuständen zur Verfügung.
  - "Gekommen/Gekommen"(Summe +/+),
  - "Gekommen/Gegangen"(Summe +/-)
  - "Gekommen/Erstquittierung"(Summe +/*1)
  - "Gekommen/Zweitquittierung"(Summe +/*2)

  Jede Art der Berechnung wird bei Bedarf in der Meldestatistik in einer Spalte dargestellt.
- Durchschnittliche Anzeigedauer einer Meldung in Sekunden
  Als Berechnungsgröße stehen Ihnen folgende Zeiträume zwischen den Meldezuständen zur Verfügung.
  - "Gekommen/Gekommen"(Durchschnitt +/+),
  - "Gekommen/Gegangen"(Durchschnitt +/-)
  - "Gekommen/Erstquittierung"(Durchschnitt +/*1)
  - "Gekommen/Zweitquittierung"(Durchschnitt +/*2)

  Für jede Art der Berechnung steht Ihnen in der Meldestatistik eine Spalte zur Verfügung.

Die Berechnung des Zeitpunkts der Quittierung berücksichtigt den Meldezustand "Quittiert". Der Meldezustand "Quittiert" schließt eine Not-Quittierung oder die Quittierung durch die Steuerung mit ein.

Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Ein Bild mit einer Meldeanzeige ist angelegt.

Funktionen der Meldestatistik projektieren

Um die Funktionen der Meldestatistik zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldeanzeige und klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Meldestatistik - Allgemein".

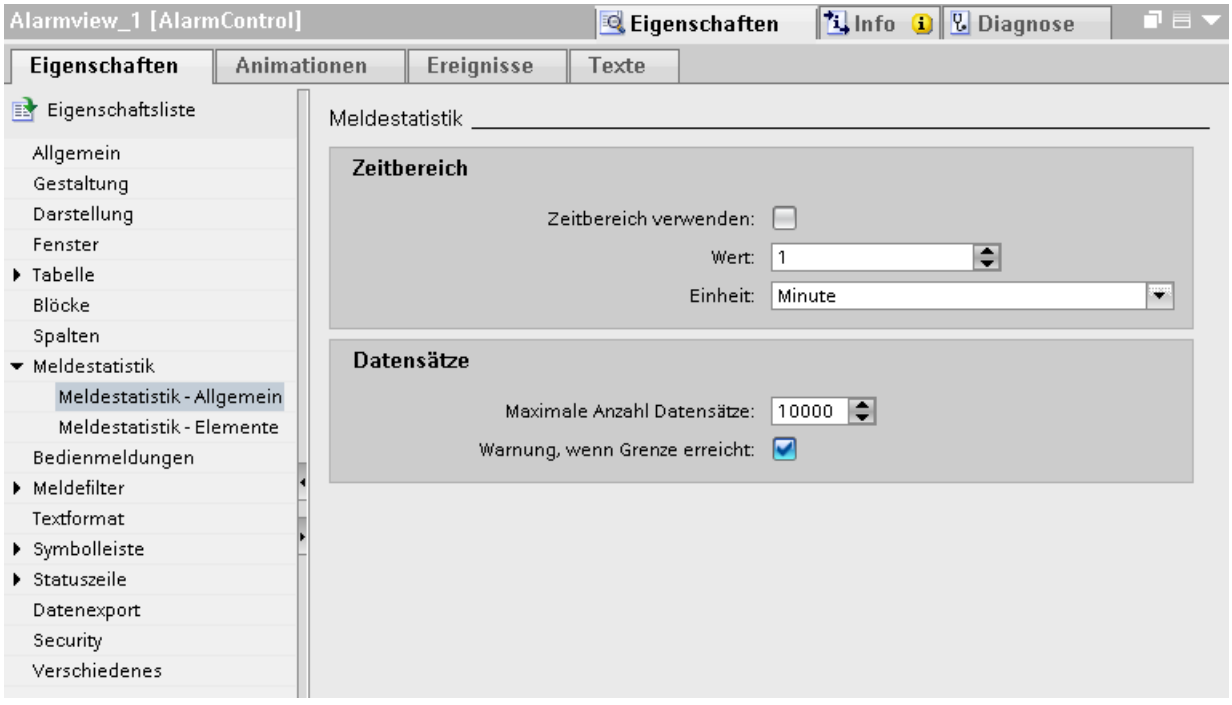

2. Um den Berechnungszeitraum zeitlich zu begrenzen, aktivieren Sie unter "Zeitbereich" die Option "Zeitbereich verwenden". Legen Sie die gewünschte Frequenz fest.
3. Um die maximale Anzahl zu verarbeitender Meldungen in der Meldestatistik festzulegen, geben Sie unter "Datensätze > Maximale Anzahl Datensätze" einen Wert fest.
   Diese Einstellung greift unter folgenden Voraussetzungen:
   – Der Berechnungszeitraum umfasst mehr Meldungen als unter "Maximale Anzahl auslesbarer Datensätze" eingestellt.
   – Der Berechnungszeitraum ist nicht eingestellt.
4. Um eine Warnung beim Überschreiten dieser Anzahl auszugeben, aktivieren Sie die Option "Warnung, wenn Grenze erreicht".

Spalten der Meldestatistik einrichten

Um die Spalten der Meldestatistik einzurichten, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldeanzeige und klicken Sie im Inspektorfenster auf " Eigenschaften > Eigenschaften > Meldestatistik - Elemente".

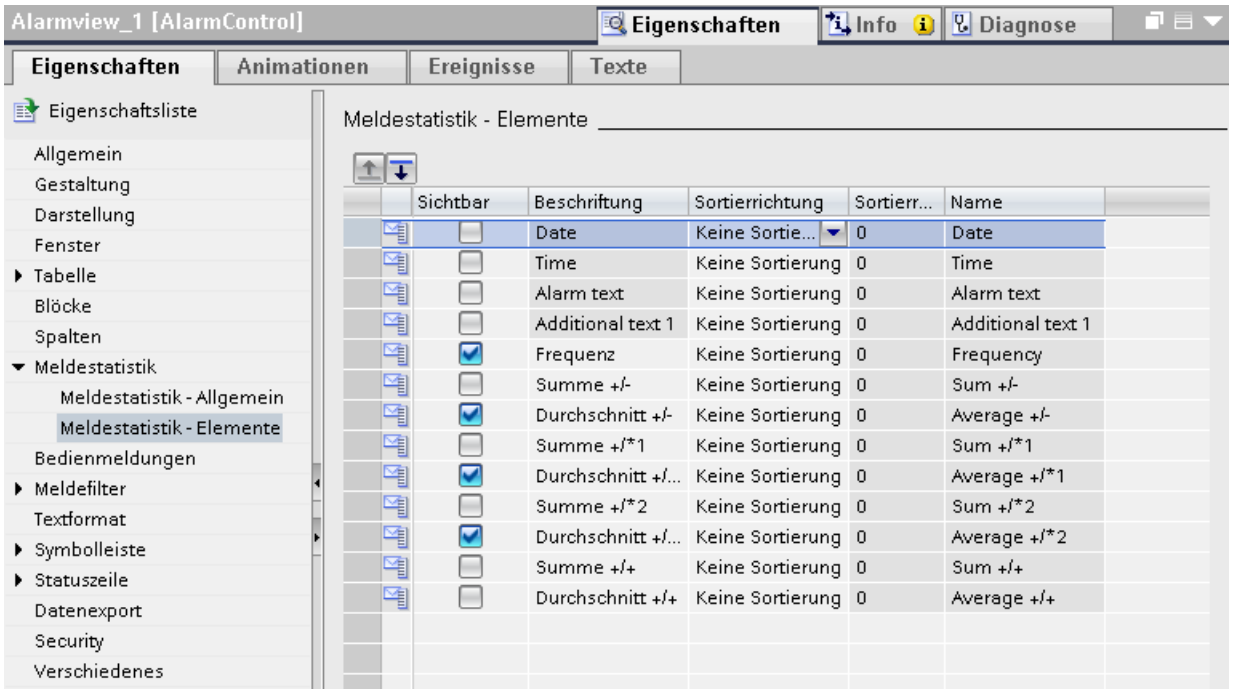

2. Aktivieren Sie in der Tabelle die Spalten der Meldestatistik, die in Runtime angezeigt werden sollen.
3. Um die Anordnung der Spalten der Meldeanzeige in Runtime festzulegen, markieren Sie im Inspektorfenster eine Reihe. Verschieben Sie die Reihe über die Schaltflächen und an die gewünschte Stelle.
   Die Spalten der Meldestatistik werden in Runtime in der gleichen Reihenfolge angezeigt im Inspektorfenster die Reihen angeordnet sind.
4. Legen Sie für jede Zeile unter "Sortieren nach" die Sortierrichtung fest.
5. Legen Sie unter "Sortierreihenfolge" die Reihenfolge fest, indem Sie die Werte 1 - 4 vergeben.

Hinweis

Die Sortierfunktion ist nur aktiv, wenn maximal vier Spalten in der Meldestatistik enthalten sind.

Ergebnis

Der Bediener zeigt in Runtime über die Schaltfläche der Meldeanzeige die Anzeige "Meldestatistik" an.

Hinweis

Für die Berechnung werden Meldungen mit den Zuständen "Quittiert" und "Gegangen" nur dann herangezogen, wenn in der Ergebnismenge zeitlich vorher eine passende Meldung mit dem Zustand "Gekommen" gefunden wird.

Wenn der Benutzer eine Meldung mit Einfachquittierung oder Zweifachquittierung nur einmal quittiert, werden für die Statistikberechnung die Meldezustände "Gekommen/Erstquittierung" und "Gekommen/Zweitquittierung" verwendet.

Wenn eine Meldung der Steuerung ansteht und Runtime mehrfach deaktiviert und aktiviert wird, wird die Meldung mehrfach mit dem Zustand "Gekommen" in das Archiv eingetragen. Die Meldung wird auch mehrfach bei der Berechnung mitgezählt.

Projektieren der Anzeigeunterdrückung von Meldungen

Übersicht zur Anzeigeunterdrückung von Meldungen

Einleitung

Um den Anlagenbediener nicht mit Informationen zu überlasten, unterdrücken Sie die Anzeige von Meldungen. Wenn nur ausgewählte Meldungen angezeigt werden, konzentriert sich der Benutzer auf die wichtigen Meldungen.

Voraussetzung

Die Schaltflächen "Meldung einblenden" und "Meldung ausblenden" sind in der Meldeanzeige projektiert.

Methoden zum Unterdrücken der Anzeige von Meldungen

Um die Anzeige kommender Meldungen zu unterdrücken, haben Sie zwei Möglichkeiten:

- **Automatisches Unterdrücken der Anzeige**
  Die Meldungen werden abhängig von einem bestimmten Anlagenzustand angezeigt. Sie geben diese Anlagenzustände in der jeweiligen Meldung als Anzeigeunterdrückungsmaske an. Die Werte der Anlagenzustände legt der Projekteur fest.
  Um den aktuellen Anlagenzustand an das Meldesystem zu übergeben, legen Sie eine Meldegruppe mit einer Anzeigeunterdrückungsvariablen an. Diese Anzeigeunterdrückungsvariable gibt den Wert der Anlagenzustände zurück.
  Sie ordnen dieser Meldegruppe Meldungen zu, die das System in Runtime bei bestimmten Anlagenzuständen nicht anzeigt.
  Beispiel:
  Sie unterdrücken im Anlagenzustand "Shutdown" die Anzeige der in diesen Anlagenzuständen üblichen Meldungen. Die Meldung "Online-Verbindung zur Steuerung wurde abgebaut" wird nur während des "Shutdown" nicht angezeigt.

  Hinweis

  Wenn Sie für eine Meldung eine Anzeigeunterdrückungsmaske angegeben haben, aber die Meldung keiner Meldegruppe zugeordnet ist, wird die Meldung in Runtime nicht unterdrückt.

  Meldungen mit Anzeigeunterdrückungsmaske ohne Zuordnung zu einer Meldegruppe werden in Runtime nicht unterdrückt. Ordnen Sie deswegen Meldungen, die in Runtime nicht anzeigt werden, immer einer Meldegruppe zu.

  Hinweis

  Sie verwenden Anzeigeunterdrückungsvariablen nur bei benutzerdefinierten Meldegruppen.

- **Manuelles Unterdrücken der Anzeige**
  Der Benutzer unterdrückt in Runtime die Anzeige einer Meldung bei Bedarf über die Schaltfläche "Meldung ausblenden" in der Meldeanzeige.
  Der Benutzer blendet die Meldung bei Bedarf über die Schaltfläche "Meldung einblenden" wieder ein.
  Nach einer einstellbaren Zeit blendet das System die Meldungen wieder ein.
  Um die manuelle Anzeigeunterdrückung zu protokollieren, projektieren Sie eine entsprechende Anwendermeldung, die beim manuellen Unterdrücken der Anzeige getriggert wird. Dazu projektieren Sie in der Meldeanzeige diese Anwendermeldung als Bedienmeldung für die Bedienaktion "Unterdrücken".

Eigenschaften von Meldungen, deren Anzeige unterdrückt ist

Für Meldungen, deren Anzeige unterdrückt ist, gilt Folgendes:

- In den folgenden Anzeigemodi der Meldeanzeige blenden Sie bei Bedarf die Meldungen über Schaltflächen wieder ein:
  - Anzeige "Aktuelle Meldungen"
  - Anzeige "Kurzzeitarchiv"
  - Anzeige "Langzeitarchiv"
- Die Meldungen werden unter der Anzeige "Meldungen mit unterdrückter Anzeige" aufgenommen und angezeigt.
- Die Meldungen werden archiviert.
- Wenn für diese Meldungen ein Zentraler Melder eingerichtet ist, wird er nicht getriggert.

Manuelle Anzeigeunterdrückung

Folgende Eigenschaften gelten nur für Meldungen, deren Anzeige **manuell** unterdrückt ist:

- Die Meldungen werden nach einer festgelegten Zeit wieder eingeblendet.
- Eine quittierpflichtige Meldung wird durch die manuelle Anzeigeunterdrückung quittiert.
- Um die manuelle Anzeigeunterdrückung zu protokollieren, projektieren Sie eine Bedienmeldung, die beim manuellen Unterdrücken der Anzeige getriggert wird.

Automatische Anzeigeunterdrückung

Folgende Eigenschaften gelten nur für Meldungen, deren Anzeige **automatisch** unterdrückt wird:

- Die Anzeigeunterdrückung löst keine Bedienmeldung aus.
- Die Steuerung quittiert quittierpflichtige Meldungen.
  In der Zeit, in der quittierpflichtige Meldungen unterdrückt sind, werden die gehenden Meldungen von der Steuerung quittiert. Wenn eine Meldung den Zustand "Gegangen" nicht besitzt, wird sie sofort von der Steuerung quittiert.

ACHTUNG
Verlust der Quittierpflicht Bei der automatischen Anzeigeunterdrückung verlieren quittierpflichtige Meldungen ihre Quittierpflicht. Verwenden Sie die automatische Anzeigeunterdrückung nur für Meldungen, die während der projektierten Anlagenzustände keine Bedienaktionen erfordern.

Wechselwirkungen

Die beiden Methoden zum Unterdrücken der Anzeige von Meldungen beeinflussen sich gegenseitig folgendermaßen:

- Wenn eine Meldung automatisch unterdrückt ist, blenden Sie die Meldung bei Bedarf über die Schaltflächen "Meldung ausblenden" und "Meldung einblenden" ein und aus.
- Sie haben die Anzeige einer Meldung manuell mit der Schaltfläche "Meldung ausblenden" unterdrückt.
  Wenn die Anlage den Zustand wechselt und die Bedingungen für das automatische Unterdrücken der Anzeige dieser Meldung stehen nicht an, wird die Meldung wieder angezeigt.
  Die als Anzeigeunterdrückungsdauer festgelegte Zeit muss noch nicht abgelaufen sein.
- Sie haben die Meldung manuell mit der Schaltfläche "Meldung ausblenden" unterdrückt. Der Zustand, der über die Anzeigeunterdrückungsvariablen für das automatische Ausblenden definiert wurde, tritt ein. Wenn die als Anzeigeunterdrückungsdauer festgelegte Zeit abgelaufen ist, bleibt die Meldung trotzdem unterdrückt.

Hinweis

Ausgeblendete Systemdiagnosemeldungen

Für ausgeblendete Systemdiagnose-Meldungen werden die Meldetexte in der Liste der ausgeblendeten Meldungen nicht angezeigt.

Die Liste der ausgeblendeten Meldungen stellt lediglich die Konfigurationsdaten der Meldung dar und keine Life-Ansicht.

Um aktive ausgeblendete Systemdiagnose-Meldungen anzuzeigen, müssen Sie den Filter für die Online-Meldeliste anpassen.

Siehe auch

Variablen einer Meldegruppe (Seite 811)

Meldeanzeige (Seite 435)

Dauer der Anzeigeunterdrückung einstellen

Einleitung

Die Zeit, nach der manuell unterdrückte Meldungen wieder automatisch angezeigt werden, legen Sie in den Einstellungen zu Meldungen fest.

Voraussetzung

- Die Projektnavigation ist geöffnet.

Vorgehen

Um die Dauer der Anzeigeunterdrückung für Meldungen einzustellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen > Meldungen". Die Einstellungen zu Meldungen werden geöffnet.
2. Stellen Sie unter "Allgemein > Dauer Anzeigeunterdrückung" die gewünschte Zeitdauer ein, die eine manuell unterdrückte Meldung ausgeblendet bleibt.
   Voreingestellt sind 30 Minuten.

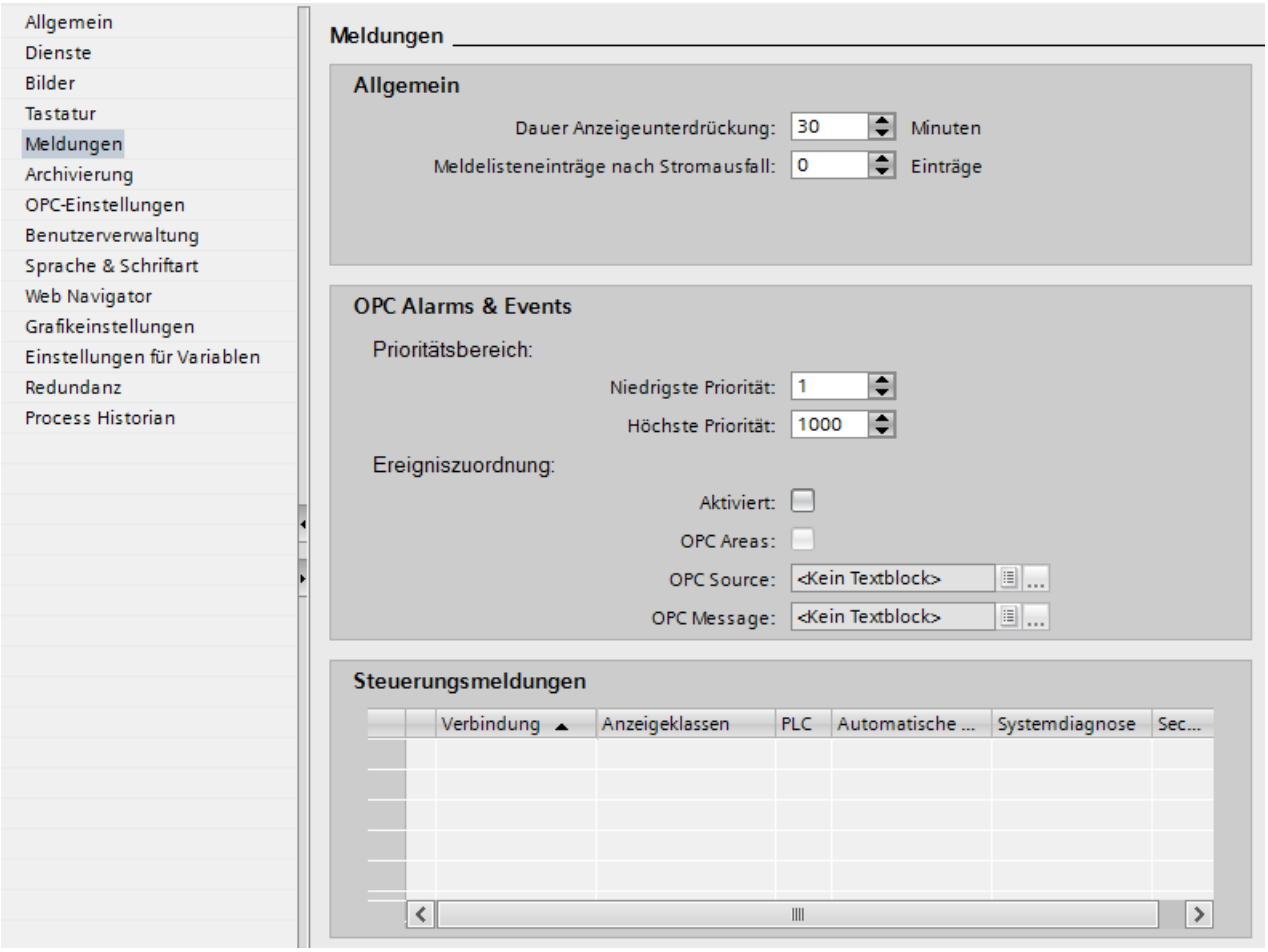

Siehe auch

Automatische Anzeigeunterdrückung projektieren (Seite 901)

Automatische Anzeigeunterdrückung projektieren

Einleitung

Bei der automatischen Anzeigeunterdrückung ist die Anzeige der Meldung abhängig von dem Zustand der Steuerung. Sie legen fest, bei welchem Anlagenzustand eine Meldung ausgeblendet wird. Die automatische Anzeigeunterdrückung projektieren Sie im Editor "HMI-Meldungen". Sie legen eine Meldegruppe mit einer Anzeigeunterdrückungsvariablen an. In der jeweiligen Meldung weisen Sie die Anlagenzustände über eine Anzeigeunterdrückungsmaske zu. Die Anzeigeunterdrückungsvariable muss den Wert der Anzeigeunterdrückungsmaske einer Meldung annehmen, damit diese in Runtime nicht angezeigt wird.

Voraussetzung

Die Schaltfläche "Meldung einblenden" ist in der Meldeanzeige projektiert.

Voraussetzungen

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Registerkarte "Meldegruppen" ist geöffnet.
- Meldungen sind angelegt.
- Die Anzeigeunterdrückungsvariable ist angelegt.

Meldegruppe zur Anzeigeunterdrückung anlegen

1. Doppelklicken Sie in der Tabelle auf "Hinzufügen".
   Eine neue Meldegruppe wird angelegt.
2. Geben Sie der neuen Meldegruppe einen Namen.
3. Wahlen Sie im Inspektorfenster unter "Variablen" die Anzeigeunterdrückungsvariable aus.

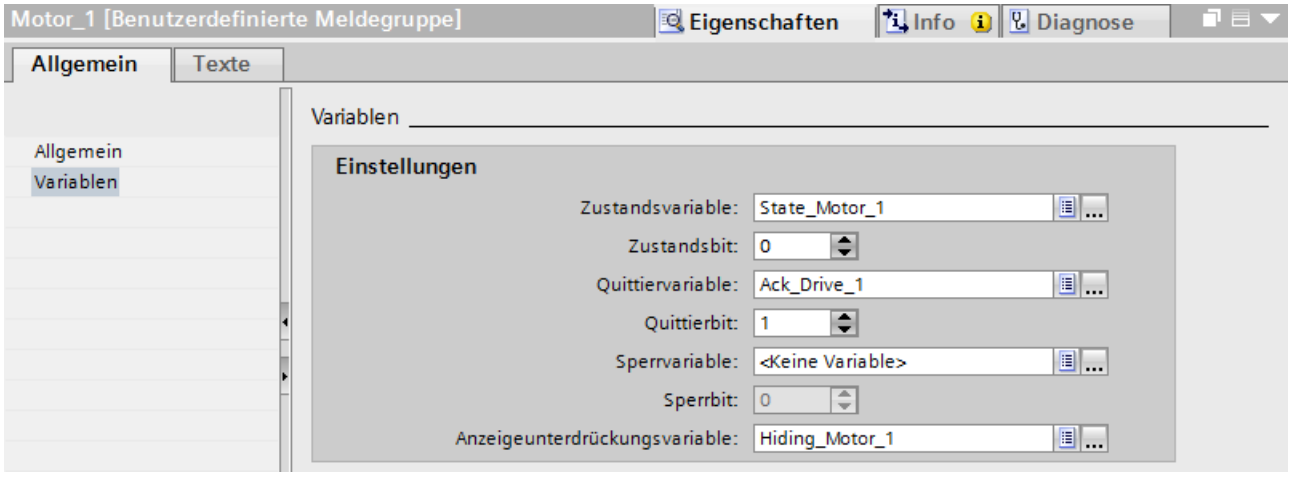

Anzeigeunterdrückungsmaske der Meldung anlegen

1. Markieren Sie die Meldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldegruppe, für die Sie eine Anzeigeunterdrückungsvariable angelegt haben.

3. Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Anzeigeunterdrückung" die Anzeigeunterdrückungsmaske ein.

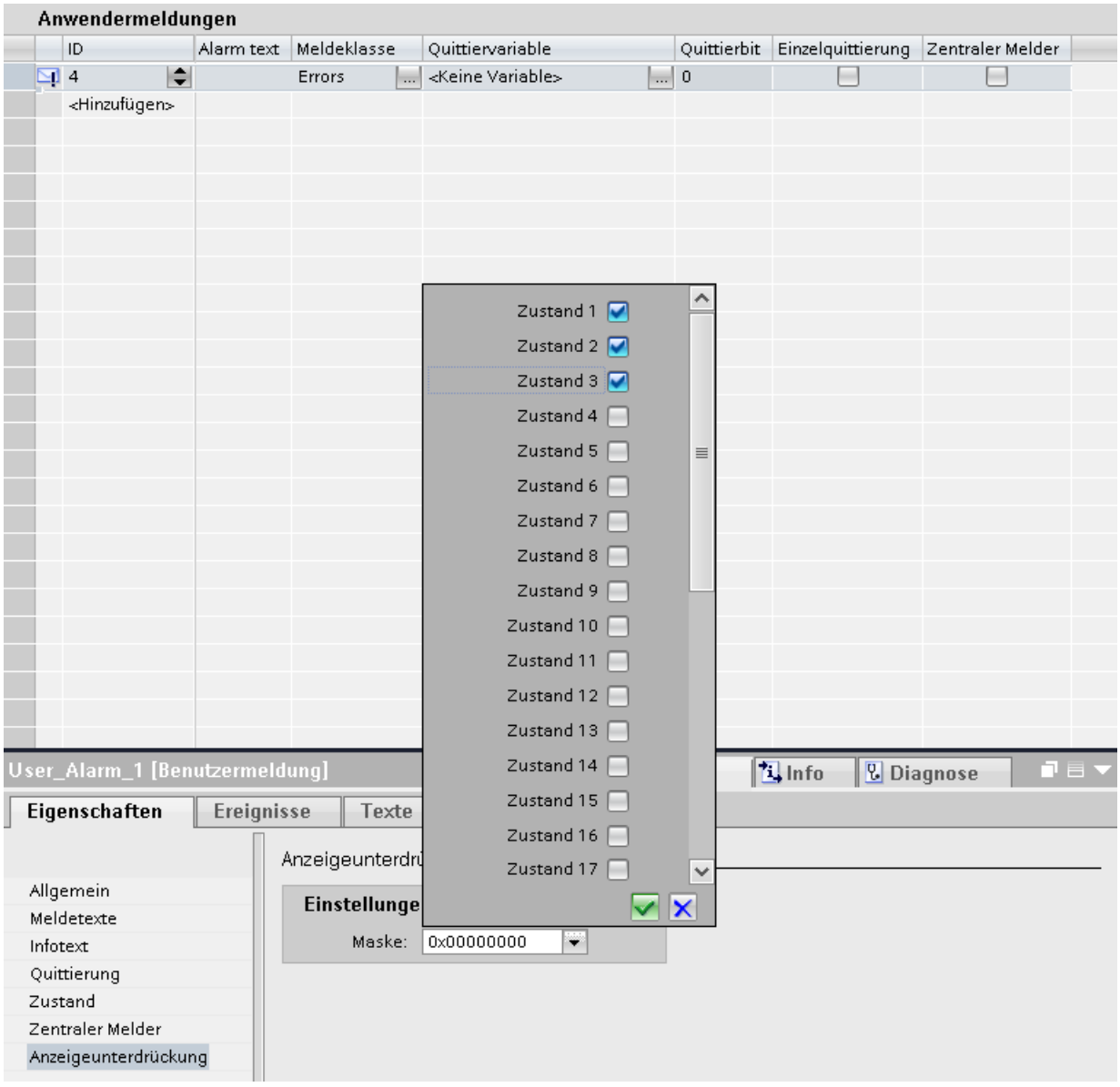
Anwendermeldungen
ID
Alarm text
Meldeklasse
Quittiervariable
Quittierbit
Einzelquittierung
Zentraler Melder
4
Errors
<Keine Variable>
0
<Hinzufügen>
Zustand 1
Zustand 2
Zustand 3
Zustand 4
Zustand 5
Zustand 6
Zustand 7
Zustand 8
Zustand 9
Zustand 10
Zustand 11
Zustand 12
Zustand 13
Zustand 14
Zustand 15
Zustand 16
Zustand 17
User_Alarm_1 [Benutzermeldung]
Info
Diagnose
Eigenschaften
Ereignisse
Texte
Allgemein
Meldetexte
Infotext
Quittierung
Zustand
Zentraler Melder
Anzeigeunterdrückung
Einstellunge
Maske:
0x00000000

Ergebnis

Wenn die Anzeigeunterdrückungsvariable den Wert der Anzeigeunterdrückungsmaske annimmt, wird die Meldung ausgeblendet. Wenn sich die Anlage also im festgelegten Zustand befindet, wird die Meldung ausgeblendet.

Hinweis

Meldungen mit Anzeigeunterdrückungsmaske ohne Zuordnung zu einer Meldegruppe werden in Runtime nicht unterdrückt. Ordnen Sie deswegen Meldungen, deren Anzeige unterdrückt werden soll, immer einer Meldegruppe zu.

Wenn eine Meldung ausgeblendet wird, blenden Sie in Runtime die Meldung über die Schaltfläche "Meldung einblenden" wieder ein.

Siehe auch

Dauer der Anzeigeunterdrückung einstellen (Seite 899)

Meldeanzeige (Seite 435)

Zentralen Melder projektieren

Einleitung

In WinCC zeigen Sie eine Meldung bei Bedarf zusätzlich zur Meldeanzeige durch ein optisches oder akustisches Signal in der Anlage an. Der Zentrale Melder ist z. B. eine Hupe oder eine Warnleuchte. Zusätzlich können Sie den zentralen Melder in einem Bild darstellen und quittieren.

Voraussetzung

- Die Schaltfläche "Quittierung zentraler Melder" ist in der Meldeanzeige projektiert.
- Der Editor "HMI-Meldungen" ist geöffnet.
- Meldungen sind angelegt.
- Das Inspektorfenster ist geöffnet.
- Ihre Anlage verfügt über einen zentralen Melder, der an eine Variable angebunden ist.

Meldungen für den zentralen Melder projektieren

1. Selektieren Sie die Meldung, die Sie mit einem zentralen Melder anzeigen lassen wollen.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Zentraler Melder".
3. Wählen Sie in der Registerkarte "Meldeklassen" die Meldeklasse der Meldungen für den zentralen Melder.
   - Um den zentralen Melder über die Schaltfläche "Quittierung zentraler Melder" der Meldeanzeige zu quittieren, aktivieren Sie im Inspektorfenster "Quittierung > Zentraler Melder > Über die Schaltfläche "Zentralen Melder quittieren".
     Mit der Schaltfläche "Quittierung zentraler Melder" schalten Sie den zentralen Melder aus. Die Meldung steht weiterhin in der Meldeanzeige an.
   - Um die Meldung über den zentralen Melder zu quittieren, wählen Sie die Variable, an die der zentrale Melder angebunden ist.

Zentralen Melder im Bild projektieren

1. Öffnen Sie das Bild, in dem Sie den zentralen Melder darstellen.
2. Projektieren Sie z.B. einen Kreis und dazugehörig eine Schaltfläche mit der Beschriftung "Zentraler Melder".
3. Projektieren Sie an das Ereignis "Klicken" der Schaltfläche ein VBS-Skript mit der Methode "QuitHorn".
4. Dynamisieren Sie die Hintergrundfarbe des Kreises mit der Variablen, an die der zentrale Melder angebunden ist.

Ergebnis

Wenn in Runtime eine Meldung mit dem zentralen Melder angezeigt wird, quittiert der Benutzer die Meldung abhängig von Ihrer Projektierung wie folgt:

- mit einer Schaltfläche "Quittierung zentraler Melder" der Meldeanzeige
- mit der projektierten Schaltfläche "Zentraler Melder"

Den Status des zentralen Melders erkennt der Benutzer an der Hintergrundfarbe des Kreises.

Siehe auch

Quittieren des zentralen Melders projektieren (Seite 918)

Meldeanzeige (Seite 435)

Archivierte Meldungen anzeigen

Einleitung

Eine Meldeanzeige zeigt abhängig von der Projektierung in Runtime auch archivierte Meldungen an. Meldungen aus dem Archiv stellt die Meldeanzeige in den Anzeigen "Kurzzeitarchiv" oder "Langzeitarchiv" dar. Die Anzeige "Kurzzeitarchiv" zeigt zusätzlich aktuelle Meldungen an.

Voraussetzung

- Die Schaltflächen "Kurzzeitarchiv" und "Langzeitarchiv" sind in der Meldeanzeige projektiert.
- Die archivierten Meldungen liegen lokal auf dem Archivserver.
- Das Bild mit der projektierten Meldeanzeige ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um eine Anzeige "Kurzzeitarchiv" oder "Langzeitarchiv" zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldeanzeige.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Aktive Liste > Kurzzeitarchiv" oder "Langzeitarchiv".
3. Projektieren Sie die Meldeanzeige, wie eine Meldeanzeige für aktuelle Meldungen.

Ergebnis

In Runtime zeigt der Benutzer über die Schaltflächen "Kurzzeitarchiv" oder "Langzeitarchiv" archivierte Meldungen an. Um z. B. nur Meldungen der Meldeklasse "Errors" oder nur Meldungen an bestimmten Anlagebereichen darzustellen, filtert oder sortiert der Benutzer die archivierten Meldungen.

Hinweis

In der Anzeige "Kurzzeitarchiv" werden die neu eingehenden Meldungen sofort aktualisiert, in der Anzeige "Langzeitarchiv" hingegen nicht. In der Anzeige "Kurzzeitarchiv" kann der Benutzer keinen Kommentar eingeben oder anzeigen lassen. Die Eingabe eines Kommentars ist nur in der Anzeige "Langzeitarchiv" möglich.

Siehe auch

Meldeanzeige (Seite 435)

Export von Meldungen projektieren

Einleitung

Um in Runtime die Meldungen in eine Datei vom Format "*.csv" zu exportieren, klickt der Bediener auf die Schaltfläche der Meldeanzeige. Sie projektieren diese Funktion in der Meldeanzeige im Editor "Bilder".

Voraussetzung

- Das Bild mit projektierter Meldeanzeige ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um den Export von Meldungen zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldeanzeige und projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften" den "Datenexport":
   - Legen Sie bei Bedarf einen eigenen Dateinamen und ein Verzeichnis fest.
   - Legen Sie unter "Umfang" fest, welche Daten der Meldeanzeige exportiert werden.
   - Legen Sie unter "CSV-Format" das Trennzeichen in der Datei vom Format ".csv" fest.
2. Projektieren Sie unter "Bedienung in Runtime" die Funktion der Schaltfläche in Runtime:
   - Um den Dialog "Datenexport-Voreinstellungen" anzuzeigen, aktivieren Sie "Dialog anzeigen".
   - Um die Daten direkt in die festgelegte Exportdatei zu exportieren, deaktivieren Sie "Dialog anzeigen".
3. Legen Sie außerdem unter "Bedienung in Runtime" fest, ob der Bediener in Runtime den Dateinamen und / oder das Verzeichnis ändern kann.

Auswertung der Meldung in Runtime projektieren

Einleitung

Um die Meldungen in Runtime auszuwerten, projektieren Sie an das Ereignis "Status geändert" einer Meldung eine benutzerdefinierte C-Funktion. Für die Auswertung in Runtime bearbeiten Sie die Funktion nach Ihren Anforderungen.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Eine benutzerdefinierte Funktion zum Auslesen der Daten ist angelegt.
  Diese Funktion muss einen Zeiger auf einen String enthalten, dessen Daten mit scanf auf die Struktur "MSG_RTDATA_STRUCT" abgebildet werden.

Vorgehen

Um für eine Meldung die Auswertung in Runtime zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Status geändert".
3. Wählen Sie die benutzerdefinierte Funktion zum Auslesen der Daten.

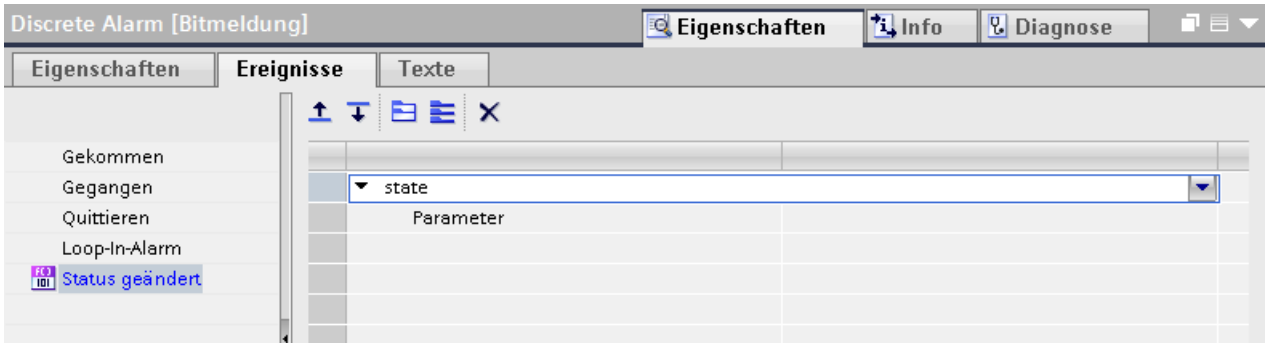

In Runtime werden die Meldungsdaten ausgelesen. Meldungsdaten sind z. B. Meldenummer, Zeitstempel und Meldezustand.

4. Projektieren Sie im Editor "Skripte" innerhalb der Funktion die Auswertung der Meldungsdaten nach Bedarf.

Ergebnis

Wenn sich der Meldezustand der Meldung ändert, werden in Runtime die Meldungsdaten ausgelesen und entsprechend Ihrer Projektierung ausgewertet.

Beispiel

Mit folgender, benutzerdefinierten Funktion "state" lesen Sie z. B. den Status der Meldungen aus:

```
Void state (MSG_RTDATA_STRUCT Parameter)
DWORD state;
state = Parameter.dwMsgState;
printf("Status: %d\r\n", state);
```

3.2.3.4 Protokollieren von Meldungen

Protokollieren von Meldungen projektieren

Übersicht

WinCC kann alle im System auftretenden Meldungen protokollieren. Die folgenden Möglichkeiten stehen Ihnen dazu zur Verfügung:

- Ausgeben eines Meldefolgeprotokolls
  In Runtime wird jede Meldung und deren Zustandsänderungen laufend auf dem Standarddrucker des Bediengeräts ausgegeben.
- Ausgeben eines Meldeprotokolls
  Sie projektieren Editor "Protokolle" ein Meldeprotokoll und legen fest, wann es in Runtime ausgegeben wird:
  - Für eine ereignisgesteuerte Ausgabe projektieren Sie an einem Objekt die Systemfunktion "DruckeProtokoll" an ein Ereignis. Das Objekt ist z. B. eine Schaltfläche oder eine Variable.
  - Für eine zeitgesteuerte Ausgabe legen Sie im Aufgabenplaner eine Aufgabe vom Typ "Druckauftrag" an. Sie weisen dem Auftrag das gewünschte Meldeprotokoll zu.

  Hinweis

  Im Meldeprotokoll legen Sie fest, ob aktuelle Meldungen oder archivierte Meldungen ausgegeben werden.

Voraussetzung

- Am Bediengerät wurde ein Drucker konfiguriert.

Laufende Protokollierung von Meldungen aktivieren

Um die laufende Protokollierung von Meldungen zu aktivieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Aktivieren Sie "Meldungen > Allgemein > Protokoll".

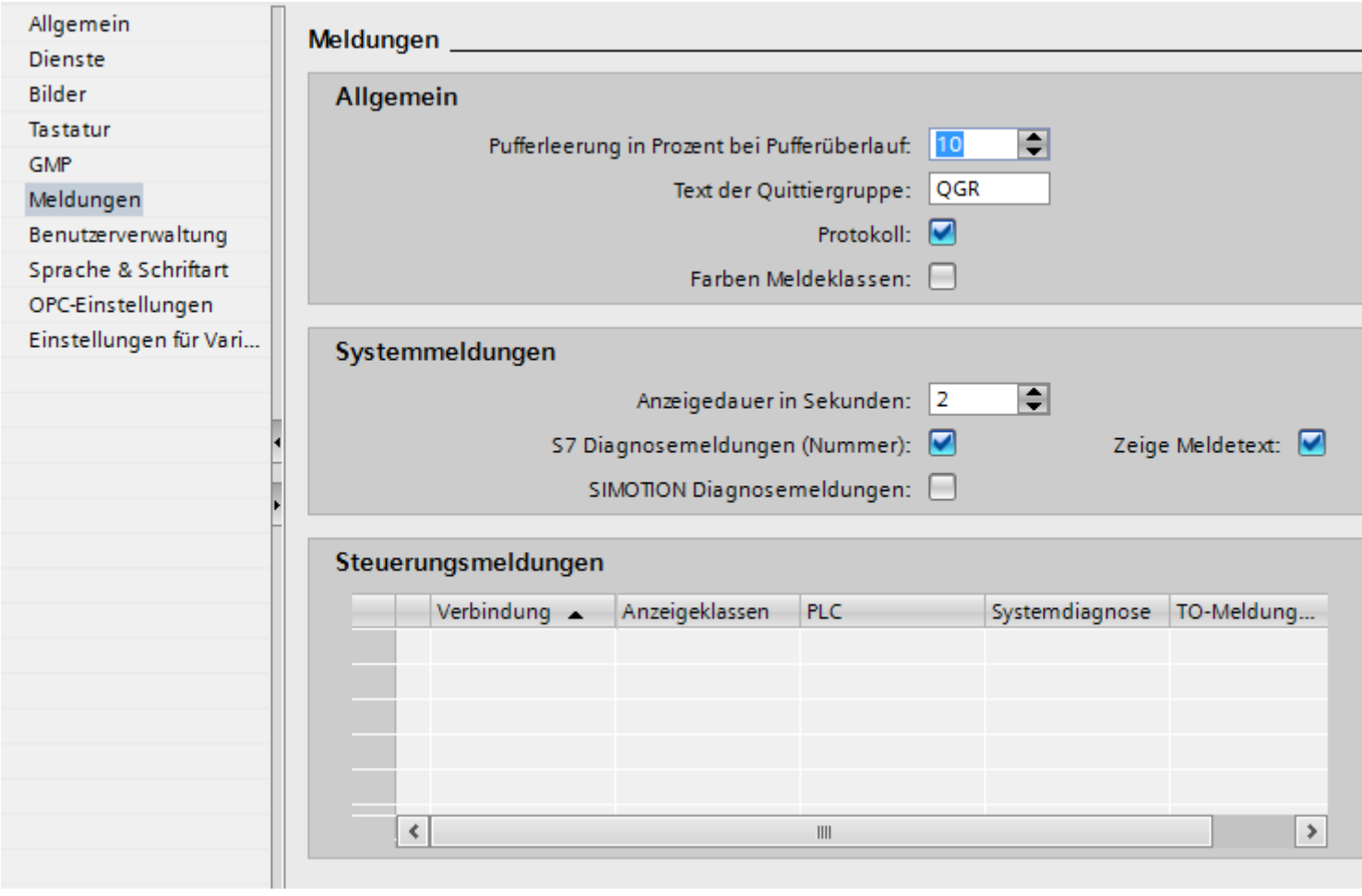

Ergebnis

Die aktuellen Meldungen werden am Drucker des Bediengeräts ausgegeben.

Meldung von der Protokollierung ausschließen

Um bestimmte Meldungen von der Protokollierung auszuschließen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "HMI-Meldungen".
2. Markieren Sie auf der Registerkarte des gewünschten Meldeverfahrens die Meldung, die Sie von der Protokollierung ausnehmen.
3. Deaktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Protokoll".

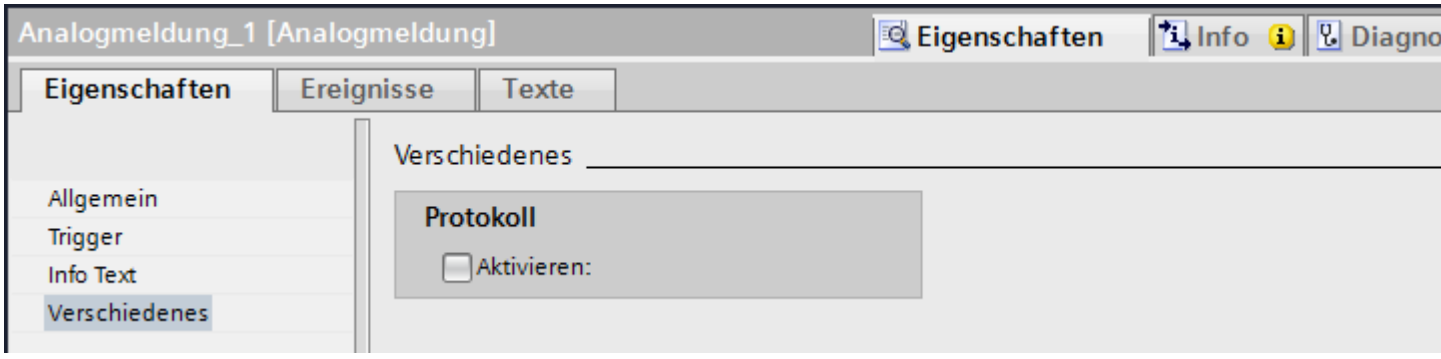

Ergebnis

WinCC gibt diese Meldungen nicht aktuell auf dem angeschlossenen Drucker aus.

Siehe auch

Grundlagen zur Protokollerstellung (Seite 1287)

Protokoll erstellen (Seite 1294)

Protokollausgabe in Runtime Advanced (Seite 1307)

Meldeprotokoll (Seite 1353)

Arbeiten mit Protokollen (Seite 1287)

Protokollieren von Meldungen projektieren

Übersicht

WinCC protokolliert bei Bedarf alle im System auftretenden Meldungen. Die folgenden Möglichkeiten stehen Ihnen dazu zur Verfügung:

- Ausgeben der aktuellen Anzeige einer Meldeanzeige
  Sie haben die Möglichkeit ein Systemprotokoll auszudrucken. Bei Bedarf projektieren Sie ein eigenes Protokoll. Druckumfang, Seitenbereich und Drucker oder Zieldatei legen Sie im Druckauftrag fest.
- Ausgeben eines Meldefolgeprotokolls
  In Runtime wird jede Meldung laufend auf einem Drucker ausgegeben.

Voraussetzungen

- Der Editor "Bilder" ist geöffnet.
- Meldungen sind projektiert.
- Eine Meldeanzeige mit der Schaltfläche "Aktuelle Ansicht drucken" ist projektiert.

Vorgehen

Um die Ausgabe der aktuellen Anzeige einer Meldeanzeige in einem eigenen Protokoll zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die gewünschte Meldeanzeige.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Drucken > Druckauftrag" den gewünschten Druckauftrag.
   Voreingestellt ist der Systemdruckauftrag. Sie wählen bei Bedarf einen eigenen Druckauftrag oder einen geänderten Systemdruckauftrag.

Ergebnis

Wenn der Benutzer in Runtime auf die Schaltfläche klickt, gibt der Drucker das gewählte Protokoll mit den aktuell in der Meldeanzeige angezeigten Meldungen aus.

Hinweis

Drucker einstellen

Den Drucker konfigurieren Sie im Control Panel des jeweiligen Bediengeräts. Die Einstellungen sind abhängig vom Betriebssystem des Bediengeräts.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

Siehe auch

Arbeiten mit Protokollen (Seite 1287)

3.2.4 Quittierung projektieren

3.2.4.1 Quittierpflicht einer Meldung über Meldeklasse projektieren

Einleitung

Um eine Meldung mit Quittierpflicht zu projektieren, ordnen Sie diese einer Meldeklasse mit dem Zustandsautomaten "Meldung mit Einfachquitterung" zu.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.
- Die gewünschte Meldung ist angelegt.

Zustandsautomaten für eine Meldeklasse wählen

Der Zustandsautomat vordefinierter Meldeklassen ist festgelegt. Sie wählen den Zustandsautomaten nur bei benutzerdefinierten Meldeklassen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Meldeklasse" und markieren Sie die gewünschte Meldeklasse.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung" das gewünschte Quittierkonzept.

Meldungen einer quittierpflichtigen Meldeklasse zuordnen

Um einer Meldung eine Meldeklasse mit Quittierpflicht zuzuordnen, gehen Sie folgendermaßen vor.

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte des gewünschten Meldeverfahrens und markieren Sie die gewünschte Meldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse der Meldung aus.

Ergebnis

Die Meldung wird in Runtime erst ausgeblendet, wenn sie der Bediener quittiert hat.

3.2.4.2 Trigger zum Quittieren von Meldungen projektieren

Einleitung

Grundsätzlich legen sie die Quittierpflicht einer Meldung über die Meldeklasse fest. Dann quittiert der Bediener die Meldung über die Funktionstaste "ACK" des Bediengeräts oder die Schaltfläche "Quittieren" der Meldeanzeige.

Außerdem gibt es folgende Möglichkeiten die Quittierung zu triggern:

- Schaltfläche zum Quittieren einer Meldung projektieren
- Bitmeldung durch die Steuerung quittieren

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.

- Die gewünschte Meldung ist angelegt.
- Im Editor "Bilder" sind eine Meldeanzeige und eine Schaltfläche angelegt.

Schaltfläche zum Quittieren einer Meldung projektieren

Um eine Schaltfläche zum Quittieren einer Meldung zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie im Editor "Bilder" die Schaltfläche.
2. Projektieren Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" die Systemfunktion "MeldeanzeigeQuittiereMeldung" an das Ereignis "Klicken".
3. Wählen Sie als Parameter die Meldeanzeige.

Bitmeldung durch die Steuerung quittieren

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Bitmeldung" und markieren Sie die gewünschte Bitmeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > PLC" die Variable und das Bit, durch das die Meldung von der Steuerung quittiert wird.

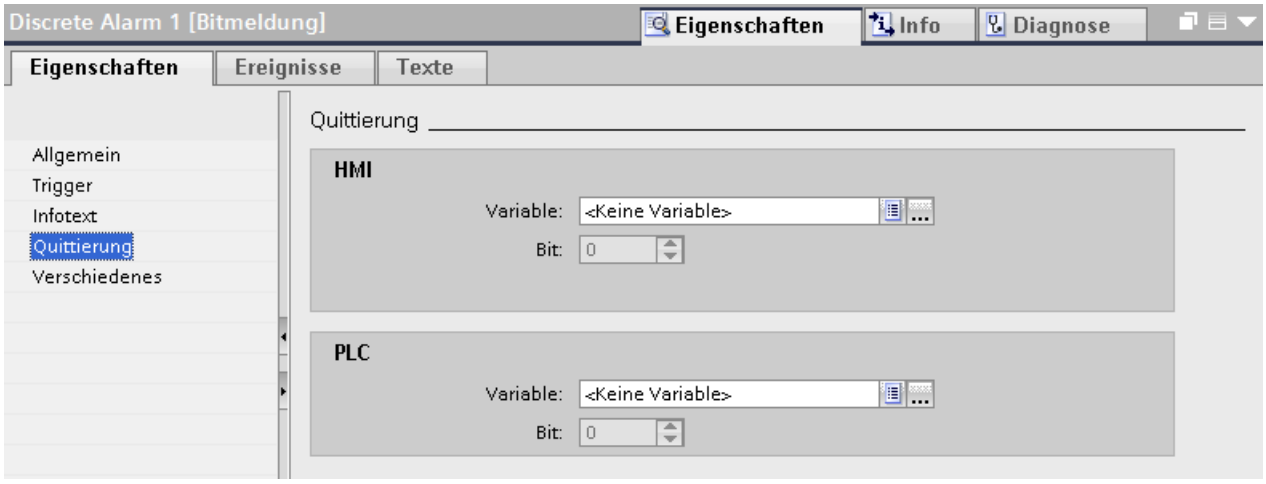

Quittieren von Meldungen einer Meldegruppe

In Meldegruppen wirken die unterschiedlichen Möglichkeiten der Quittierung folgendermaßen:

- Quittieren der Meldung durch den Bediener
  Sie quittieren die Meldung einer Meldegruppe z. B. mit der Quittiertaste <ACK> oder einer Funktionstaste. Alle Meldungen der Meldegruppe werden quittiert.
- Quittieren der Meldung durch die Steuerung
  Die Meldung einer Meldegruppe wird durch das Setzen eines Bits in der Variablen quittiert. Nur die zu quittierende Meldung wird quittiert.

3.2.4.3 Meldungsquittierung an die Steuerung senden

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldung ist angelegt und einer Meldeklasse mit Quittierpflicht zugeordnet.

Hinweis

Die Quittierung von Analogmeldungen kann nicht an die Steuerung gesendet werden.

Meldungsquittierung an die Steuerung senden

Um zu projektieren, dass die Quitterung einer Meldung an die Steuerung gesendet wird, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Bitmeldung" und markieren Sie die gewünschte Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Quittierung".
3. Wählen Sie unter "HMI" die Variable und das Bit, das durch die Meldungsquittierung gesetzt wird.

Hinweis

Auf den Speicherbereich der Quittiervariablen darf von Bediengerät und Steuerung nur lesend zugegriffen werden.

Ergebnis

Wenn der Bediener in Runtime die Meldung quittiert, wird der Bedienvorgang an die Steuerung weitergeleitet.

3.2.5 Quittierung projektieren

3.2.5.1 Quittieren einer Meldung projektieren

Einleitung

In den Meldeklassen ist festgelegt, wie die Meldungen einer Meldeklasse quittiert werden. Wenn Sie eine Meldung einer Meldeklasse zuordnen, legen Sie den Zustandsautomaten der Meldung fest.

Voraussetzungen

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.
- Die gewünschte Meldung ist angelegt.

Vorgehen

Um die Quittierung einer Meldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Meldeklasse" und markieren Sie die gewünschte Meldeklasse.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > Einstellungen" den gewünschten Zustandsautomaten.
   - Wenn Sie "Kommentar Benutzer eindeutig zuordnen" aktivieren, schreibt der Benutzer in Runtime einen Kommentar zur Meldungsquittierung, der mit der Meldung archiviert wird.
   - Wenn Sie "Kommentar einer eingehenden Meldung verwenden" aktivieren, wird jede Statusänderung dieser Meldung mit dem gleichen Kommentar versehen, der beim Meldeereignis "Gekommen" vergeben wurde.

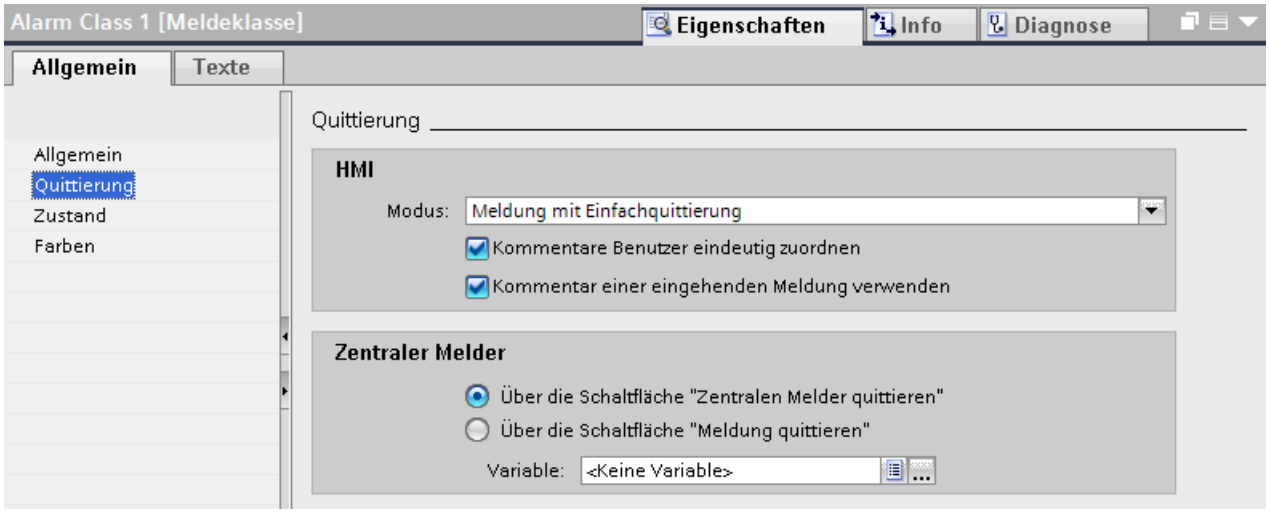

3. Markieren Sie im Editor "HMI-Meldungen" die gewünschte Meldung.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse der Meldung.
5. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > Einstellungen" die Quittiervariable aus, in der die Meldungsquittierung gespeichert wird und das Bit, das dafür gesetzt wird.
6. Um die Meldung von der Sammelquittierung der Meldeanzeige auszunehmen, aktivieren Sie "Einzelquittierung".

Meldung über eine Systemfunktion quittieren

1. Markieren Sie die gewünschte Meldung im Editor "HMI-Meldungen".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Gekommen".
   Die Funktionsliste wird geöffnet.
3. Projektieren Sie im Dialog "Funktionsliste" die Systemfunktion "SetzeBit" an das Ereignis "Gekommen".
4. Geben Sie als Parameter "Variable (In/Out)" die Quittiervariable vom Datentyp "BOOL" an.

3.2.5.2 Quittieren einer Meldegruppe projektieren

Einleitung

Eine Meldegruppe quittieren Sie über die Quittiervariable. Um das Quittierbit zu setzen und rückzusetzen, projektieren Sie ein Anzeige- und Bedienobjekt.

Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Meldegruppen sind angelegt.
- Die gewünschte Meldung ist angelegt.

Vorgehen

Um die Quittierung einer Meldegruppe zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie im Editor "HMI-Meldungen" die gewünschte Meldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldegruppe aus, zu der die Meldung gehört.
3. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Meldegruppen".
4. Markieren Sie die Meldegruppe.
5. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Variablen > Einstellungen" die Quittiervariable und das Quittierbit.
6. Öffnen Sie den Editor "Bilder" und projektieren Sie an die Quittiervariable eine Schaltfläche, die in Runtime das Quittierbit setzt.

Ergebnis

In Runtime beobachtet der Benutzer die Meldungen einer Funktion gemeinsam. Wenn er die Ursache einer Störung erkannt hat, quittiert er alle Meldungen dieser Funktion mit der projektierten Bedienaktion.

3.2.5.3 Quittieren des zentralen Melders projektieren

Einleitung

In WinCC zeigen Sie eine Meldung bei Bedarf zusätzlich zur Meldeanzeige durch ein optisches oder akustisches Signal in der Anlage an. Der Zentrale Melder ist z. B. eine Hupe oder eine Warnleuchte. Sie haben folgende Möglichkeiten den Zentralen Melder zu quittieren:

- Über die Schaltfläche der Meldeanzeige
  Diese Schaltfläche "Zentralen Melder quittieren" muss in der Meldeanzeige projektiert sein. Die Schaltfläche ist immer aktiv, auch wenn eine a e Option zum Quittieren des Zentralen Melders aktiviert ist. Mit der Schaltfläche schalten Sie den zentralen Melder aus. Die Meldung steht weiterhin in der Meldeanzeige an.
- Über Einzelquittierung
  Der Benutzer quittiert den Zentralen Melder mit der Meldung, die den Zentralen Melder triggert. In de ldeanzeige quittiert der Benutzer eine einzelne Meldung mit der Schaltfläche "Einzelquittierung".
- Über eine Variable
  Über die festgelegte Variable wird der zentrale Melder angesteuert.
  Um z. B. den Zentralen Melder über eine Schaltfläche zu quittieren, verwenden Sie diese Option.

Voraussetzungen

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.
- Die gewünschte Meldung ist angelegt und der Meldeklasse zugeordnet.

Vorgehen

Um die Quittierung des Zentralen Melders zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Meldeklassen".
2. Markieren Sie die gewünschte Meldeklasse.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Quittierung > Zentraler Melder" wie der Zentrale Melder quittiert wird.

Ergebnis

Der Benutzer quittiert den Zentralen Melder in Runtime, wie Sie es für die Meldeklasse dieser Meldung festgelegt haben.

3.2.5.4 Sammelquittieren von Meldungen projektieren

Einleitung

Über die Schaltfläche der Meldeanzeige quittieren Sie alle anstehenden, sichtbaren und quittierpflichtigen Meldungen der Meldeanzeige. Bei Bedarf nehmen Sie einzelne Meldungen von der Sammelquittierung aus. Wenn die Sammelquittierung ausgelöst wird, stehen die ausgenommenen Meldungen danach weiterhin an.

Voraussetzung

- Ein Bild ist geöffnet.
- Eine Meldeanzeige ist angelegt.

Vorgehen

Um eine Sammelquittierung einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "Bilder" und markieren Sie die betreffende Meldeanzeige.
2. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste - Schaltflächen > Sammelquittieren".
   Die Schaltfläche wird in der Meldeanzeige eingeblendet. Alle Meldungen einer Meldeanzeige, die nicht von der Sammelquittierung ausgenommen sind, werden über diese Schaltfläche mit einer Bedienaktion quittiert.

Einzelquittierung projektieren

1. Öffnen Sie den Editor "HMI-Meldungen" und wählen Sie die Registerkarte mit dem gewünschten Meldeverfahren.
2. Markieren Sie die Meldung, die Sie von der Sammelquittierung ausnehmen wollen.
3. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > Einzelquittierung".
   Wenn Sie die Meldungen einer Meldeanzeige mit der Schaltfläche sammelquittieren, sind die Meldungen mit der Option "Einzelquittierung" davon ausgenommen. Meldungen mit der Option "Einzelquittierung" quittiert der Benutzer einzeln.

3.2.5.5 Rücksetzen des Quittierbits der Quittiervariablen projektieren

Einleitung

Der Benutzer oder die Steuerung setzt das Quittierbit der Quittiervariablen zurück. Das Quittierbit wird direkt in der Steuerung zurückgesetzt oder in WinCC-Runtime zurückgesetzt, z. B. über ein Skript. Das Quittierbit wird nicht automatisch zurückgesetzt.

Rücksetzen des Quittierbits durch die Steuerung

Um das Quittierbit durch die Steuerung rücksetzen zu lassen, verwenden Sie eine externe Variable als Quittiervariable.

Wenn die Meldung in Runtime in WinCC quittiert wird, wird das festgelegte Quittierbit in der Steuerung gesetzt. In der Steuerung wird das Quittierbit bei Bedarf verarbeitet, bevor es wieder rückgesetzt wird.

Eine Meldung wird in der Steuerung durch einen wieder verwendbaren Funktionsbaustein oder durch Funktionen getriggert. In diesem Baustein kann auch die Meldung quittiert oder das Quittierbit rückgesetzt werden.

Das Auswerten und Rücksetzen des Quittierbits durch die Steuerung hat folgenden Vorteil:

Die Quittierung ist durch externe Signale einfach möglich, z. B. durch Taster im Feld.

Sie quittieren extern und über die Meldeanzeige. Beide Methoden nutzen zur Quittierung das gleiche Bit im Automatisierungssystem.

Hinweis

Damit die Meldeanzeige in WinCC die externe Quittierung zeitlich sicher erfasst, projektieren Sie das Rücksetzen durch das Automatisierungssystem zeitlich verzögert.

Rücksetzen des Quittierbits durch WinCC

Wenn Sie eine Meldung ausschließlich über die Quittiervariable quittieren, setzen Sie das Quittierbit sofort zurück, nachdem das Quittierbit gesetzt wurde.

Wenn Sie eine Meldung über die Meldeanzeige quittieren, projektieren Sie das Rücksetzen der Quittiervariablen separat.

Sie haben folgende Möglichkeiten in WinCC das Quittierbit zurückzusetzen:

- Schaltfläche
  Sie setzen das Quittierbit über eine Schaltfläche zurück.
- Systemfunktion
  Verwenden Sie die Systemfunktion "RücksetzeBitInVariable". Wenn das Quittierbit gesetzt ist, wird es sofort wieder rückgesetzt.

  Hinweis

  Wenn die Quittiervariable vom Datentyp "BOOL" ist, verwenden Sie die Systemfunktion "RuecksetzeBit".

  Hinweis

  Wenn viele Meldungen mit Skripten verarbeitet werden, belastet dies den Arbeitsspeicher. Lassen Sie daher nach Möglichkeit das Quittierbit durch die Steuerung zurücksetzen.

3.3 Archivieren von Meldungen

3.3.1 Projektieren der Meldearchivierung

3.3.1.1 Grundlagen zum Archivieren von Meldungen

Einleitung

In einem Meldearchiv dokumentieren Sie die Meldungen des Projekts.

Hinweis

Die Meldearchivierung steht nicht für alle Bediengeräte zur Verfügung.

Projektierungsschritte

Um eine Meldung zu archivieren, führen Sie folgende Projektierungsschritte aus:

1. Meldearchiv anlegen
   Sie legen für das Meldearchiv folgende Eigenschaften fest:
   - Archivierungsmethode
     Verhalten des Archivs bei Erreichen eines bestimmten Füllstands
   - Ablageort und Dateiformat
   - Archivgröße
   - Verhalten bei Runtime-Start
2. Meldearchiv einer Meldeklasse zuordnen
   Bei Bedarf archivieren Sie in einem Meldearchiv die Meldungen mehrerer Meldeklassen.
3. Meldung der archivierbaren Meldeklasse zuordnen
4. Anzeige der archivierten Meldungen in einer Meldeanzeige projektieren

Inhalt des Meldearchivs

Für projektierte Meldungen werden alle Zustände archiviert. Für eine quittierpflichtige Meldung sind z. B. folgende drei Einträge im Archiv gespeichert:

- 04.08.2007 10:00:25:520, Analogmeldung, ID5, **K**, Error, Füllstand um 10 % überschritten
- 04.08.2007 10:01:20:442, Analogmeldung, ID5, **Q**, Error, Füllstand um 10 % überschritten
- 04.08.2007 10:01:30:112, Analogmeldung, ID5, **G**, Error, Füllstand um 10 % überschritten

Im Beispiel sind die Meldezustände mit folgenden Buchstaben gekennzeichnet:

K = Gekommen

Q = Quittiert

G = Gegangen

Im Meldearchiv werden alle zu einer Meldung gehörenden Daten und die Konfigurationsdaten gespeichert, z. B. Meldeklasse, Zeitstempel und Meldetext.

Die mögliche Anzahl der archivierten Meldungen ist abhängig vom verwendeten Datenträger. Die archivierten Meldungen bearbeiten Sie in anderen Programmen weiter, z. B. zu Analysezwecken.

Hinweis

Meldetext und Störort werden nur archiviert, wenn Sie dies in den Eigenschaften eines Archivs projektiert haben.

Archivierte Meldungen, die Meldetext und Störort enthalten, übersteigen die geschätzte Größe der projektieren Meldungen. Prüfen Sie, ob der angegebene Speicherort trotzdem ausreicht.

Hinweis

Der Zeitstempel einer archivierten Meldung wird immer im Standardformat UTC (Universal Time Coordinated) angegeben.

Archivierungsmethoden

Die Archivierungsmethode bestimmt, wie sich das Meldearchiv bei Erreichen der projektierten Größe verhält. In WinCC stehen Ihnen folgende Archivierungsmethoden zur Verfügung:

- Umlaufarchiv
  Wenn die projektierte Größe des Archivs erreicht ist, werden die ältesten Einträge gelöscht. Bei Erreichen der projektierten Größe des Archivs werden ca. 20 % der ältesten Einträge gelöscht. Es ist daher nicht möglich, die vollständige Anzahl der projektierten Einträge anzuzeigen. Wählen Sie bei der Projektierung eine entsprechende Größe des Umlaufarchivs. Alternativ projektieren Sie ein segmentiertes Umlaufarchiv.
- Segmentiertes Umlaufarchiv
  Hier werden mehrere gleich große Einzelarchive erstellt, die nacheinander gefüllt werden. Wenn alle Archive vollständig gefüllt sind, wird das älteste Archiv neu beschrieben.
- Archiv mit füllstandsabhängigem Systemalarm
  Bei Erreichen eines definierbaren Füllstands, z. B. von 90 %, wird eine Systemmeldung ausgelöst. Wenn die projektierte Größe des Archivs erreicht ist, werden neu auftretende Meldeereignisse nicht archiviert.
- Archiv mit füllstandsabhängigem Ausführen von Systemfunktionen
  Bei vollständig gefülltem Archiv wird das Ereignis "Überlauf" ausgelöst. An das Ereignis projektieren Sie eine Funktionsliste.
  Wenn die projektierte Größe des Archivs erreicht ist, werden neu auftretende Meldeereignisse nicht archiviert.

Anzeigen von archivierten Daten

Die archivierten Daten zeigen Sie am Bediengerät in einer dazu projektierten Meldeanzeige an.

Siehe auch

Arbeiten mit Archiven (Seite 1071)

3.3.1.2 Meldearchiv anlegen

Einleitung

In den Meldearchiven speichern Sie Meldungen in Runtime. Das Meldearchiv geben Sie in der Meldeklasse an. Ein Meldearchiv archiviert die Meldungen mehrerer Meldeklassen. Die Meldearchive legen Sie im Editor "Archive" an. Beim Anlegen eines Meldearchivs legen Sie Folgendes fest:

- Name
- Größe
- Ablageort
- Verhalten bei Runtime-Start
- Archivtyp
- Erweiterter Inhalt
- Prüfsumme

Zusätzlich geben Sie bei Bedarf zu jedem Archiv einen Kommentar ein.

Voraussetzung

- Im Editor "Archive" ist die Registerkarte "Meldearchive" geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

Um ein Meldearchiv anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Tabelle auf "<Hinzufügen>".
   Ein neues Meldearchiv wird angelegt.
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" einen eindeutigen Namen für das Meldearchiv ein.

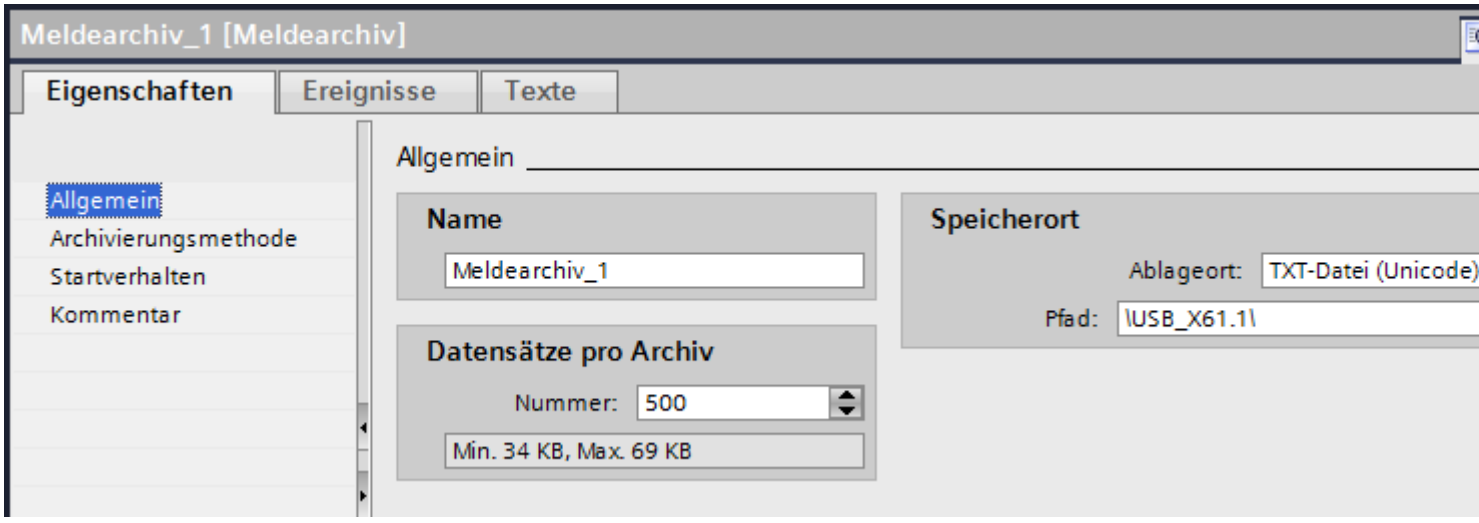

3. Legen Sie unter "Datensätze pro Archiv" die Anzahl der Meldungen fest, die in einem Archiv gespeichert werden.
   Der ungefähre Speicherbedarf auf dem Speichermedium wird angezeigt. Wenn Sie den Meldetext mit Variablenwerten archivieren, erhöht sich der Speicherplatzbedarf entsprechend.
4. Wählen Sie im Feld "Ablageort" aus, wie die Archiveinträge gespeichert werden.
5. Wählen Sie abängig vom gewählten "Ablageort" entweder den "Pfad" oder den "Namen der Datenquelle" aus.
6. Um feststellen zu können, ob Meldedaten nachträglich verändert wurden, aktivieren Sie "Eigenschaften > Eigenschaften > Allgemein > Prüfsumme".

   Hinweis
   Geräteabhängigkeit

   Die Option "Prüfsumme" steht nur für Anzeige- und Bediengeräte zur Verfügung, welche die "GMP-konforme Projektierung" unterstützen.

7. Geben Sie bei Bedarf in der Kategorie "Kommentar" einen erklärenden Text zur Dokumentation Ihrer Projektierung ein.

Ergebnis

Das Meldearchiv ist angelegt. Sie können diesem Meldearchiv eine oder mehrere Meldeklassen zuweisen.

Siehe auch

Meldungen archivieren (Seite 925)
Arbeiten mit Archiven (Seite 1071)

3.3.1.3 Meldungen archivieren

Übersicht

Meldungen archivieren Sie in folgenden Arbeitsschritten:

- Sie legen ein Meldearchiv an.
- Sie ordnen einer Meldeklasse das angelegte Meldearchiv zu.
- Sie ordnen einer Meldung die Meldeklasse mit dem angelegten Meldearchiv zu.
- In den Runtime-Einstellungen legen Sie fest, in welcher Sprache die Archive geschrieben werden.
- Sie werten archivierte Meldungen aus.
  Die archivierten Meldungen werten Sie direkt in Ihrem WinCC Projekt aus, z. B. in einer Meldeanzeige oder mit anderen Anwenderprogrammen, z. B. Microsoft Excel.

Hinweis

Variablenfelder im Meldetext

Die Reihenfolge der Variablenfelder im Meldetext ist sprachabhängig. Beim Archivieren von Meldungen in einer Datei vom Format "*.csv" wird die Reihenfolge der Runtime-Sprache verwendet.

Wenn Sie die Variable eines Ausgabefelds in einer Sprache ändern, wird in allen anderen Sprachen das geänderte Ausgabefeld am Ende des Meldetexts angezeigt. Dadurch kann sich die Reihenfolge der Ausgabefelder in der Archivdatei ändern.

Hinweis

Textlisten im Meldetext

Wenn Sie in den Meldetext eine Textliste eingebunden haben und die Meldung archiviert wurde, werden im Meldearchiv die Textlistenträge angezeigt.

Voraussetzung

- Ein Meldearchiv ist angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.

Meldearchiv einer Meldeklasse zuordnen

Um einer Meldeklasse das Meldearchiv zuzuordnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Editor "HMI-Meldungen" die Registerkarte "Meldeklassen".
2. Markieren Sie die gewünschte Meldeklasse.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Archiv" das gewünschte Meldearchiv aus.

Meldung einer Meldeklasse zuordnen

Um eine Meldung der Meldeklasse zuzuordnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Editor "HMI-Meldungen" die Registerkarte "Analogmeldungen" oder "Bitmeldungen".
2. Markieren Sie die gewünschte Meldung.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Meldeklasse" die Meldeklasse aus, für die das Meldearchiv projektiert wurde.

Ergebnis

Die Meldung wird in das projektierte Meldearchiv gespeichert.

Siehe auch

Meldearchiv anlegen (Seite 923)

Arbeiten mit Archiven (Seite 1071)

3.3.1.4 Meldeanzeige für archivierte Meldungen projektieren

Einleitung

Archivierte Meldungen zeigen Sie in Runtime in einer Meldeanzeige oder einem Meldefenster an.

Voraussetzung

- Eine Meldeanzeige oder ein Meldefenster ist im Editor "Bilder" projektiert
- Ein Meldearchiv ist im Editor "Archive" angelegt.
- Meldungen sind im Editor "HMI-Meldungen" als"archivierbar" projektiert.

Meldeanzeige für archivierte Meldungen projektieren

Um die Anzeige archivierter Meldungen in einer Meldeanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Bild mit der Meldeanzeige und markieren Sie die Meldeanzeige.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Meldearchiv".
3. Klicken Sie auf die Schaltfläche "..." und wählen Sie das Meldearchiv.
4. Projektieren Sie die Meldeanzeige weiter wie für die Anzeige aktueller Meldungen.

Ergebnis

In Runtime werden in der Meldeanzeige die archivierten Meldungen ausgegeben.

3.3.1.5 Meldepufferüberlauf projektieren

Einleitung

Die Größe des Meldepuffers hängt vom Typ des Bediengeräts ab. In WinCC legen Sie den Anteil vom Meldepuffer in Prozent fest, der bei Meldepufferüberlauf gelöscht wird.

Voraussetzung

- Ein Projekt ist angelegt.

Vorgehen

Um das Verhalten bei Meldepufferüberlauf zu projektieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter Ihrem Bediengerät auf "Runtime-Einstellungen".
2. Geben Sie unter "Meldungen > Allgemein > Pufferleerung in Prozent bei Pufferüberlauf" einen Wert zwischen 1 und 100 ein.
   Dieser Wert legt den Prozentsatz des Meldepuffers fest, der beim Meldepufferüberlauf gelöscht wird.

Ergebnis

Wenn die Daten im Meldepuffer die Speichergröße des Meldepuffers überschreiten, werden so viel Prozent der am längsten vorhandenen Meldungen des Meldepuffers gelöscht, wie Sie projektiert haben.

Hinweis

Den Überlauf des Meldepuffers fragen Sie nicht nach Meldeverfahren getrennt ab. Mit der Systemfunktion "LoescheMeldepuffer" löschen Sie gezielt Meldungen bestimmter Meldeklassen aus dem Meldepuffer.

Siehe auch

Arbeiten mit Archiven (Seite 1071)

3.3.1.6 Direktzugriff auf die ODBC-Archivdatenbank

Übersicht

Der Ablageort eines Archivs ist eine Datenbank oder eine Datei.

Die Datenbank wird über ihren "Data Source Name" (DSN) angesprochen. Sie wählen die Datenbank, die Sie in WinCC verwenden, wie folgt:

Wählen Sie im Windows-Startmenü unter "Einstellungen > Systemsteuerung > ODBC Data Sources" die Datenbank aus.

Zur Ablage von Archivdaten geben Sie beim Projektieren anstatt eines Verzeichnisnamens den "Data Source Name" (DSN) an. Mit dem DSN referenzieren Sie die Datenbank und den Ablageort.

Für die Weiterbearbeitung und Auswertung der Archivdaten steht Ihnen dann der gesamte Funktionsumfang der Datenbank zur Verfügung.

Verwendung

Die Data Source stellt die Verbindung zur Datenbank her. Sie legen die Data Source auf dem gleichen PC an, auf dem die Runtime-Software abgelegt wird. Die auf diesem PC konfigurierte DSN geben Sie dann in WinCC beim Anlegen eines Archivs an.

Sie greifen über die ODBC-Schnittstelle mit anderen Programmen direkt auf die Datenbank zu, z. B. MS Access oder MS SQL-Server.

Zusätzlich ist es möglich, den Aufruf eines Programms, z. B. Microsoft Access, auf dem Bediengerät mit der Systemfunktion "StarteProgramm" zu projektieren. Dabei wird die Runtime nicht unterbrochen.

3.3.1.7 ODBC-Datenquelle einrichten

Einleitung

Um in Runtime die Prozesswerte oder Meldungen in der Archivdatenbank zu speichern, richten Sie die ODBC-Datenquelle ein.

Voraussetzung

- Eine Archivdatenbank ist angelegt.
  Die Archivdatenbank erstellen Sie mit SQL Enterprise Manager. Nähere Hinweise hierzu finden Sie bei Microsoft.

Vorgehen

Um eine ODBC-Datenquelle einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Systemsteuerung die "Verwaltung" und wählen Sie "Datenquellen (ODBC)".
   Der Dialog "ODBC Data Source Administrator" wird geöffnet.
2. Klicken Sie "User DSN > Add".
3. Wählen Sie "SQL-Server" und klicken Sie auf "Finish".
4. Geben Sie im folgenden Dialog den einen Namen für die Benutzer DSN und den SQL-Server ein und klicken Sie auf "Next".
5. Legen Sie im nächsten Dialog das Anmeldeverfahren für die SQL-Datenbank fest und klicken Sie auf "Next".
6. Aktivieren Sie "Change the default database to".
7. Wählen Sie die von ihnen erstellte Datenbank und klicken Sie auf "Next".
8. Klicken Sie im folgenden Dialog auf "Fertigstellen".

3.3.1.8 Prüfsumme für ein Archiv projektieren

Einleitung

In einem regulierten Projekt haben Sie die Möglichkeit, die Archivdaten eines Variablenarchivs oder eines Meldearchivs mit einer Prüfsumme zu versehen. Anhand dieser Prüfsumme kann im Betrieb der Anlage festgestellt werden, ob die Daten eines Archivs nachträglich verändert wurden.

Hinweis

Geräteabhängigkeit

Die Option "Prüfsumme" steht nur für Anzeige- und Bediengeräte zur Verfügung, welche die "GMP-konforme Projektierung" unterstützen.

Voraussetzung

- Die GMP-konforme Projektierung ist aktiviert.
- Ein Variablenarchiv bzw. ein Meldearchiv ist angelegt.

Vorgehensweise

Um ein Variablenarchiv oder ein Meldearchiv mit Verwendung der Prüfsumme zu projektieren, gehen Sie wie folgt vor:

1. Öffnen Sie das Variablenarchiv oder das Meldearchiv im entsprechenden Archiveditor.
2. Wählen Sie im Feld "Ablageort" den Eintrag "Datei - CSV (ASCII)" oder "Datei -TXT (Unicode)".

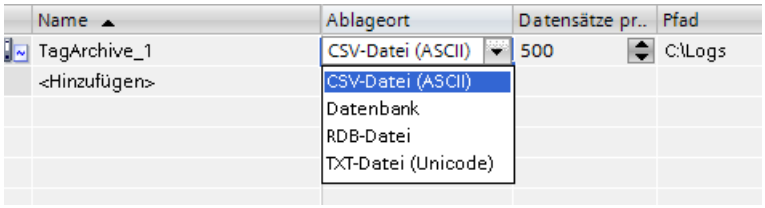

3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Archivierungsmethode" die Option "Systemmeldung anzeigen bei" oder "Ereignis auslösen".

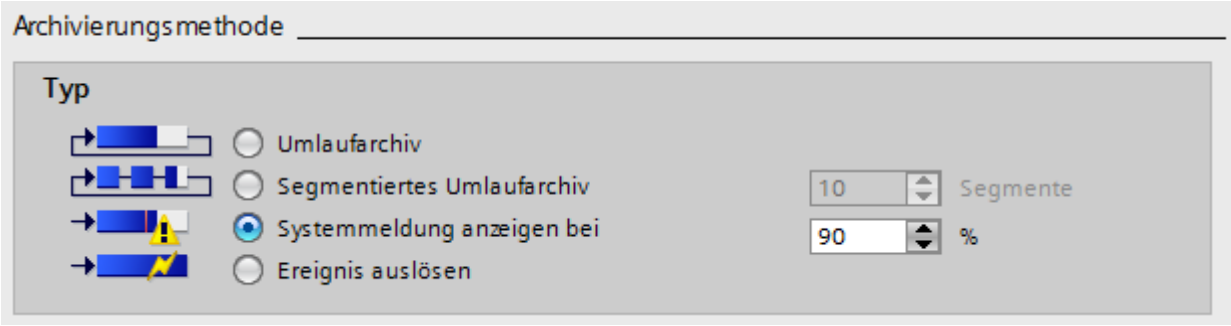

4. Aktivieren Sie in der Tabelle des Editors die Option "Prüfsumme aktivieren".
5. Aktivieren Sie in der Tabelle des Editors die Option "Archivierung bei Runtime-Start aktivieren".
   Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

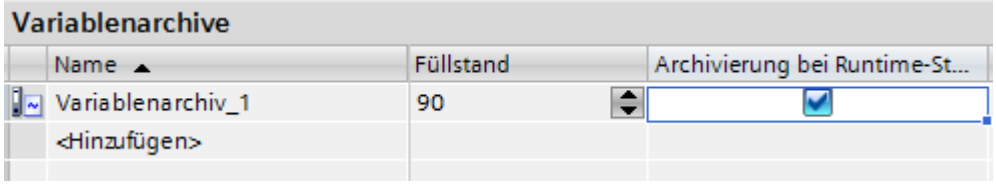

6. Speichern Sie das Projekt.

Ergebnis

Die Archivdaten des Archivs werden in Runtime mit einer Prüfsumme versehen.

3.3.1.9 Prüfsumme von Archivdaten auswerten

Einleitung

Wenn Sie ein Variablenarchiv oder ein Meldearchiv mit der Bildung einer Prüfsumme projektiert haben, können Sie prüfen, ob die Archivdaten nachträglich verändert wurden.

Hinweis

Geräteabhängigkeit

Die Option "Prüfsumme" steht nur für Anzeige- und Bediengeräte zur Verfügung, welche die "GMP-konforme Projektierung" unterstützen.

Zur Prüfung der Integrität der Archivdaten steht Ihnen das DOS-Programm "HmiCheckLogIntegrity" zur Verfügung.

Das Programm "HmiCheckLogIntegrity" unterstützt die Prüfung folgender Dateien:

- Archivdateien von Meldearchiven, Variablenarchiven und Audit im CSV-Format
- Archivdateien von Meldearchiven, Variablenarchiven und Audit im TXT-Format
- Rezepturdatensätze im CSV-Format
- Rezepturdatensätze im TXT-Format

Das Programm "HmiCheckLogIntegrity.exe" finden Sie im Installationsverzeichnis von WinCC im Ordner "WinCC Runtime Advanced", z. B. <C:\Program Files\Siemens\Automation WinCC Runtime Advanced>

Hinweis

Audit Trail und Archive mit Prüfsumme

Bevor Sie WinCC durch einen Service Pack oder eine neue Version aktualisieren, beenden und sichern Sie den Audit Trail bzw. die Archive mit Prüfsumme. Nach der Aktualisierung von WinCC wird der Audit Trail bzw. die Archive mit Prüfsumme mit neuen Dateien fortgeführt.

Damit stellen Sie sicher, dass mit der neuen Version die Archive auf einem definierten Stand begonnen werden.

Vorgehensweise

1. Kopieren Sie die zu prüfende Datei vom Bediengerät auf Ihren Projektierungsrechner.
2. Öffnen Sie die Eingabeaufforderung über "Start > Programme > Zubehör > Eingabeaufforderung".
3. Geben Sie in der Eingabeaufforderung den Pfad zur "HmiCheckLogIntegrity.exe" und anschließend ein Leerzeichen ein. Nach dem Leerzeichen geben Sie den Ablageort der zu prüfenden Datei in Anführungszeichen ein.

4. Drücken Sie die Taste <Enter>.
5. Die Prüfung wird durchgeführt.
   Wenn die geprüften Daten konsistent sind, wird die Meldung "Consistency check succeeded" ausgegeben.

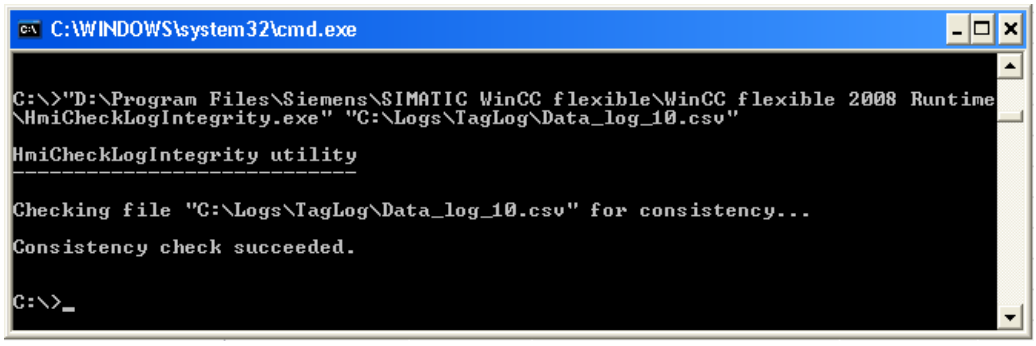

Wenn die geprüften Daten nicht konsistent sind, wird die Meldung "Consistency check failed" ausgegeben. Zusätzlich wird ein Hinweis auf die erste inkonsistente Zeile in der Datei ausgegeben.

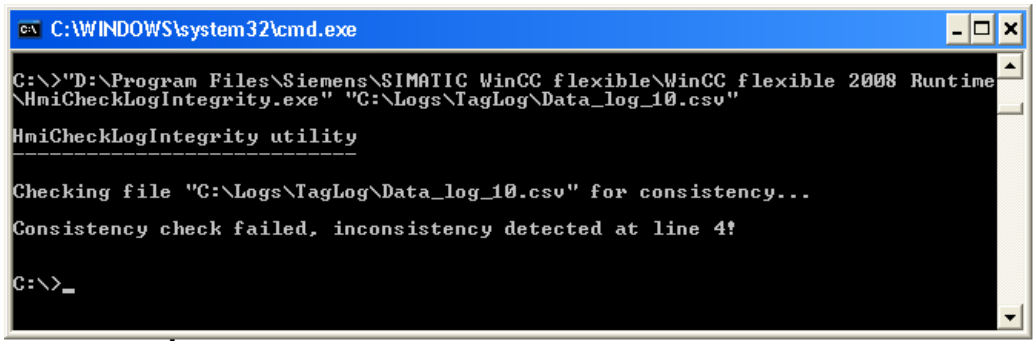

Hinweis

Wenn in der Pfadangabe zu dem Programm "HmiCheckLogIntegrity.exe" Leerzeichen vorkommen, dann müssen Sie die Pfadangabe in Anführungszeichen setzen.

Alternativ prüfen Sie die Integrität der Archivdaten mit dem AuditViewer.

3.3.1.10 Prüfsummen für Archive in regulierten Projekten

Prüfsummen in regulierten Projekten

Mit WinCC projektieren Sie bei Bedarf regulierte Projekte gemäß den Anforderungen der FDA. Für die Erstellung regulierter Projekte steht Ihnen die Option "Audit" zur Verfügung.

In regulierten Projekten versehen Sie die archivierten Daten eines Meldearchivs mit einer Prüfsumme. Mit dieser Prüfsumme stellt der Bediener im Betrieb der Anlage fest, ob die Daten eines Meldearchivs verändert wurden.

Damit die Prüfsumme eindeutig ist, beginnt die Bildung der Prüfsumme mit der ersten Zeile eines Meldearchivs und bezieht bei fortlaufender Prüfsummenbildung die vorherigen Zeilen mit ein. Deshalb steht die Bildung der Prüfsumme nur für folgende Archive zur Verfügung:

- Archiv mit füllstandsabhängiger Systemmeldung
- Archiv mit Ausführen von Systemfunktionen bei vollem Archiv .

Die Bildung der Prüfsumme steht auch nur für Meldearchive zur Verfügung, die als Datei vom Format "*.csv" oder "*.txt"(Unicode) gespeichert werden.

Wenn Sie ein bestehendes Archiv ohne Prüfsumme fortführen wollen mit Prüfsumme, dann wird von dem bestehenden Archiv eine Sicherungskopie im Dateiformat "*.keep" angelegt. Danach wird ein neues Archiv mit Prüfsummen angelegt.

Um die Gültigkeit der Prüfsummen zu erhalten, gelten bei der Verwendung der Systemfunktion "KopiereArchiv" folgende Einschränkungen:

- Ein Archiv ohne Prüfsummen kann nicht kopiert werden in ein Archiv mit Prüfsummen.
- Ein Archiv mit Prüfsummen kann nicht kopiert werden in ein Archiv ohne Prüfsummen.

3.3.1.11 Verhalten des Archivs bei Sprachumschaltung in Runtime steuern

Einleitung

In den Runtime-Einstellungen Ihres Bediengeräts wählen Sie die Sprache, die verwendet wird, um Archive in Runtime zu schreiben.

Voraussetzung

- Im Editor "Projektsprachen" sind die Sprachen aktiviert, die Sie in Ihrem Projekt verwenden z. B. "Deutsch (Deutschland)" und "Englisch (USA)".

Vorgehen

Um die Anlaufsprache festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Sprache und Schriftart".
2. Aktivieren Sie die Runtime-Sprachen z. B. "Deutsch (Deutschland)" und "Englisch (USA)".
3. Legen Sie die "Reihenfolge für Sprachumschaltung" fest. Mit der 0 legen Sie die Anlaufsprache fest, z. B. folgendermaßen:
   - Deutsch 0.
   - Englisch 1
     Deutsch ist mit "0" als "Anlaufsprache" festgelegt.
4. Wählen Sie in der Projektnavigation "Runtime-Einstellungen > Allgemein".
5. Wählen Sie "Archive > Archivierungssprache > Anlaufsprache".

Ergebnis

Nach dem Transfer startet das Projekt. Durch "Reihenfolge für Sprachumschaltung" ist Deutsch als "Anlaufsprache" festgelegt. Die Archive werden in Deutsch geschrieben. Während Runtime schaltet der Bediener die Runtime-Sprache auf Englisch. Die Archive werden jedoch weiterhin in Deutsch geschrieben.

Der Bediener beendet Runtime. Durch die vorhergehende Sprachumschaltung in Runtime ist beim erneuten Start die "Anlaufsprache" Englisch. Da Englisch die Anlaufsprache ist, werden nun auch die Archive in Englisch geschrieben.

Bei einer weiteren Sprachumschaltung in Runtime bleibt die Archivierungssprache Englisch, bis zu einem erneuten Ende von Runtime.

Wenn Sie anstatt "Anlaufsprache" eine andere Option bzw. Sprache wählen, werden die Archive immer in der gewählten Sprache geschrieben. Unabhängig davon, ob der Bediener in Runtime die Sprache umschaltet.

Siehe auch

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 987)

3.3.1.12 Verhalten des Archivs bei Runtime-Start steuern

Einleitung

Bei der Projektierung eines Archivs legen Sie das Startverhalten des Archivs für den Start von Runtime fest. In den Eigenschaften des Archivs legen Sie fest, ob die Archivierung beim Runtime-Start beginnt. Weiterhin legen Sie fest, ob ein bestehendes Archiv fortgeführt oder überschrieben wird.

Das Startverhalten legen Sie für jedes Archiv gesondert fest.

Voraussetzung

- Ein Archiv ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften des Archivs ist geöffnet.

Vorgehen

Um das Startverhalten eines Variablenarchivs zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor Archive" das Archiv, für das Sie das Startverhalten festlegen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Startverhalten".

3. Wenn die Archivierung bei Runtime-Start beginnen soll, aktivieren Sie im Bereich "Archivierung" die Option "Archivierung bei Runtime-Start aktivieren".
   Alternativ starten Sie die Archivierung z. B. mit der Systemfunktion "StarteArchivierung".
4. Wählen Sie im Bereich "Verhalten bei Neustart" das Startverhalten des Archivs.
   - Mit der Option "Archiv zurücksetzen" werden die archivierten Werte gelöscht und die Archivierung wird neu begonnen.
   - Mit der Option "Archiv fortführen" werden die neu zu archivierenden Werte an das bestehende Archiv angefügt.

Alternativ konfigurieren Sie das Startverhalten eines Archivs direkt in der Tabelle des Editors "Archive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Ergebnis

In Runtime wird die Archivierung entsprechend Ihren Einstellungen gestartet.

3.3.1.13 Archivierung Füllstandsabhängig steuern

Einleitung

Die Größe eines Archivs wird durch die Anzahl der Einträge bestimmt. Mit der Archivierungsmethode bestimmen Sie, wie sich das Archiv verhält, wenn es vollständig gefüllt ist.

Archivierungsmethoden

Folgende Archivierungsmethoden stehen zur Verfügung:

- Umlaufarchiv
  Wenn die projektierte Größe des Archivs erreicht ist, werden die ältesten Einträge gelöscht. Bei Erreichen der projektierten Größe des Archivs werden ca. 20 % der ältesten Einträge gelöscht. Daher ist es nicht möglich, die vollständige Anzahl der projektierten Einträge anzuzeigen. Wählen Sie bei der Projektierung eine entsprechende Größe des Umlaufarchivs. Alternativ projektieren Sie ein segmentiertes Umlaufarchiv.
- Segmentiertes Umlaufarchiv
  In einem segmentierten Umlaufarchiv werden mehrere gleich große Archivsegmente erstellt, die nacheinander gefüllt werden. Wenn alle Archive vollständig gefüllt sind, wird das älteste Archiv neu beschrieben.

- Archiv mit füllstandsabhängiger Systemmeldung
  Bei Erreichen eines definierbaren Füllstands, z. B. von 90 %, wird eine Systemmeldung ausgelöst. Wenn das Archiv zu 100 % befüllt ist, werden neue Variablenwerte nicht archiviert.
- Archiv mit füllstandsabhängigem Auslösen eines Ereignisses.
  Bei vollständig gefülltem Archiv wird das Ereignis "Überlauf" ausgelöst. An das Ereignis projektieren Sie eine Funktionsliste, die ausgeführt wird, wenn das Ereignis "Überlauf" eintritt. Wenn die projektierte Größe des Archivs erreicht ist, werden neue Variablenwerte nicht archiviert.
  Für die weitere Bearbeitung befüllter Archive stehen Ihnen Systemfunktionen zur Verfügung.

Voraussetzung

- Ein Archiv ist angelegt.
- Der Editor "Archive" ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften des Archivs ist geöffnet.

Vorgehensweise

1. Selektieren Sie im Editor "Archive" das Archiv, für das Sie die Archivierungsmethode festlegen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Archivierungsmethode" und wählen Sie die gewünschte Archivierungsmethode.
3. Wenn Sie ein Archiv mit der Einstellung "Segmentiertes Umlaufarchiv" gewählt haben, geben Sie die Anzahl der Archivsegmente ein. Dabei wird vom System zusätzlich ein weiteres Archivsegment für den Hauptarchiv angelegt. Somit ergibt sich die Gesamtanzahl der erstellten Archivdateien aus der Anzahl der konfigurierten Segmente sowie dem automatisch erstellten Archiv.
   Wenn Sie ein Archiv mit der Einstellung "Systemmeldung anzeigen bei" gewählt haben, geben Sie den Füllstand in Prozent an, bei dessen Erreichen eine Systemmeldung ausgegeben wird.
   Wenn Sie die Einstellung "Ereignis auslösen" gewählt haben, projektieren Sie in der Gruppe "Ereignisse" die Funktionsliste.

Alternativ konfigurieren Sie die Archivierungsmethode eines Archivs direkt in der Tabelle des Editors "Archive". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

Das Ereignis "Überlauf" steht in der Tabelle des Editors nicht zur Verfügung. Die Funktionsliste müssen Sie daher im Inspektorfenster projektieren.

Ergebnis

Das gewählte Archiv verhält sich in Runtime entsprechend den Einstellungen.

3.3.2 Projektieren der Meldearchivierung

3.3.2.1 Grundlagen zum Archivieren von Meldungen

Einleitung

In einem Meldearchiv dokumentieren Sie die Meldungen des Projekts. Meldearchive werden vom System angelegt. Wenn z. B. in Runtime eine Störung oder Grenzwertverletzung eintritt, wird eine im Editor "HMI-Meldungen" projektierte Meldung in Runtime ausgegeben. Jedes Meldeereignis einer Meldung wird archiviert, z. B. die Zustandsänderung der Meldung von "Meldung gekommen" auf "Meldung quittiert".

Sie speichern die Meldeereignisse in einer Archivdatenbank und/oder archivieren sie als Meldeprotokoll auf Papier. Die in der Datenbank archivierten Meldungen geben Sie bei Bedarf in Runtime aus, z. B. in einer Meldeanzeige.

Die archivierten Meldungen werden in einem Umlaufarchiv abgelegt, das aus mehreren Einzelsegmenten besteht. Die Größe aller Segmente und eines Segments legen Sie in den Einstellungen zu Archiven unter "Meldearchiv" fest.

Für die regelmäßige Sicherung der Daten konfigurieren Sie Backups.

Projektierungsschritte

Um eine Meldung zu archivieren, führen Sie folgende Projektierungsschritte aus:

1. "Einstellungen für Archive" festlegen
   Sie legen die Einstellungen zur Archivierung von Meldungen in den Einstellungen zu Archiven fest.
2. Archivierung für Meldungen aktivieren
   Sie legen fest, dass Meldungen bestimmter Meldeklassen archiviert werden.
3. Meldung der archivierbaren Meldeklasse zuordnen

Inhalt des Meldearchivs

In den Meldearchiven werden alle zu einer Meldung gehörigen Daten inklusive der Konfigurationsdaten gespeichert. Sie lesen aus den Archiven z. B. alle Eigenschaften einer Meldung ab, z. B. Meldeklasse, Zeitstempel und die Meldetexte. Eine nachträgliche Änderung an den Konfigurationsdaten einer Meldung bewirkt, dass ein neues Archivsegment mit den neuen Konfigurationsdaten erstellt wird. Diese Funktion verhindert, dass die Änderung Einfluss auf bereits vor der Änderung archivierte Meldungen hat.

Die mögliche Anzahl der archivierten Meldungen ist abhängig vom verwendeten Server.

Hinweis

Der Zeitstempel einer archivierten Meldung wird immer im Standardformat UTC (Universal Time Coordinated) angegeben.

Da Meldungen sprachabhängig projektiert werden, existiert in den Archiven für jede projektierte Sprache eine Tabelle mit den Konfigurationsdaten.

Speicherort und Speichermedien

Archivdaten werden in einer Datenbank gespeichert. Die gespeicherten Daten bearbeiten Sie in anderen Programmen weiter, z. B. zu Analysezwecken.

Backup für Archivsegmente

Um eine lückenlose Dokumentation Ihres Prozesses sicherzustellen, erstellen Sie Backups der Archivsegmente. Sie legen die Einstellungen dazu unter "Einstellungen zu Archiven" fest.

Anzeigen von archivierten Daten

Die archivierten Daten zeigen Sie am Bediengerät an. Dazu projektieren Sie eine Meldeanzeige, in der die Archivdaten angezeigt werden, oder blenden in Runtime über Schaltflächen die Anzeige "Kurzzeitarchiv" oder "Langzeitarchiv" ein.

Siehe auch

Arbeiten mit Archiven (Seite 1077)

3.3.2.2 Meldungen archivieren

Einleitung

In den Einstellungen zu Archiven legen Sie die Größe und den Zeitbereich der Segmente eines Meldearchivs fest. Mit der Größe der Einzelsegmente definieren Sie auch die Anzahl der Einzelsegmente.

In Runtime werden die einzelnen Segmente nacheinander befüllt. Wenn ein Segment vollständig befüllt ist, wird zum nächsten Segment gewechselt. Sie projektieren den Segmentwechsel bei Bedarf auch zeitabhängig. Wenn Sie einen Zeitpunkt für den Segmentwechsel festlegen, wird nach dem Erreichen des Zeitpunkts das nächste Archivsegment befüllt.

Hinweis

Achten Sie darauf, dass die Archivgröße nicht den freien Speicherplatz überschreitet. Der Archivmanager prüft die gewählten Einstellungen nicht auf Plausibilität. Eine hohe Anzahl von verbundenen Archivsegmenten führt beim Starten und Beenden von Runtime möglicherweise zu längeren Wartezeiten im System.

Übersicht

Meldungen archivieren Sie mit folgenden Projektierungsschritten:

- Sie konfigurieren das Meldearchiv in den Einstellungen zu Archiven.
  Sie legen die Größe und die Segmentierung fest.
- Sie projektieren eine archivierbare Meldeklasse.
- Sie ordnen einer Meldung die archivierbare Meldeklasse zu.

Voraussetzung

- Eine Meldeklasse ist angelegt.

Meldearchiv konfigurieren

Um das Meldearchiv zu konfigurieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Doppelklicken Sie in der Bereichsnavigation auf "Archivierung".
   Die Einstellungen zu Archiven werden geöffnet.

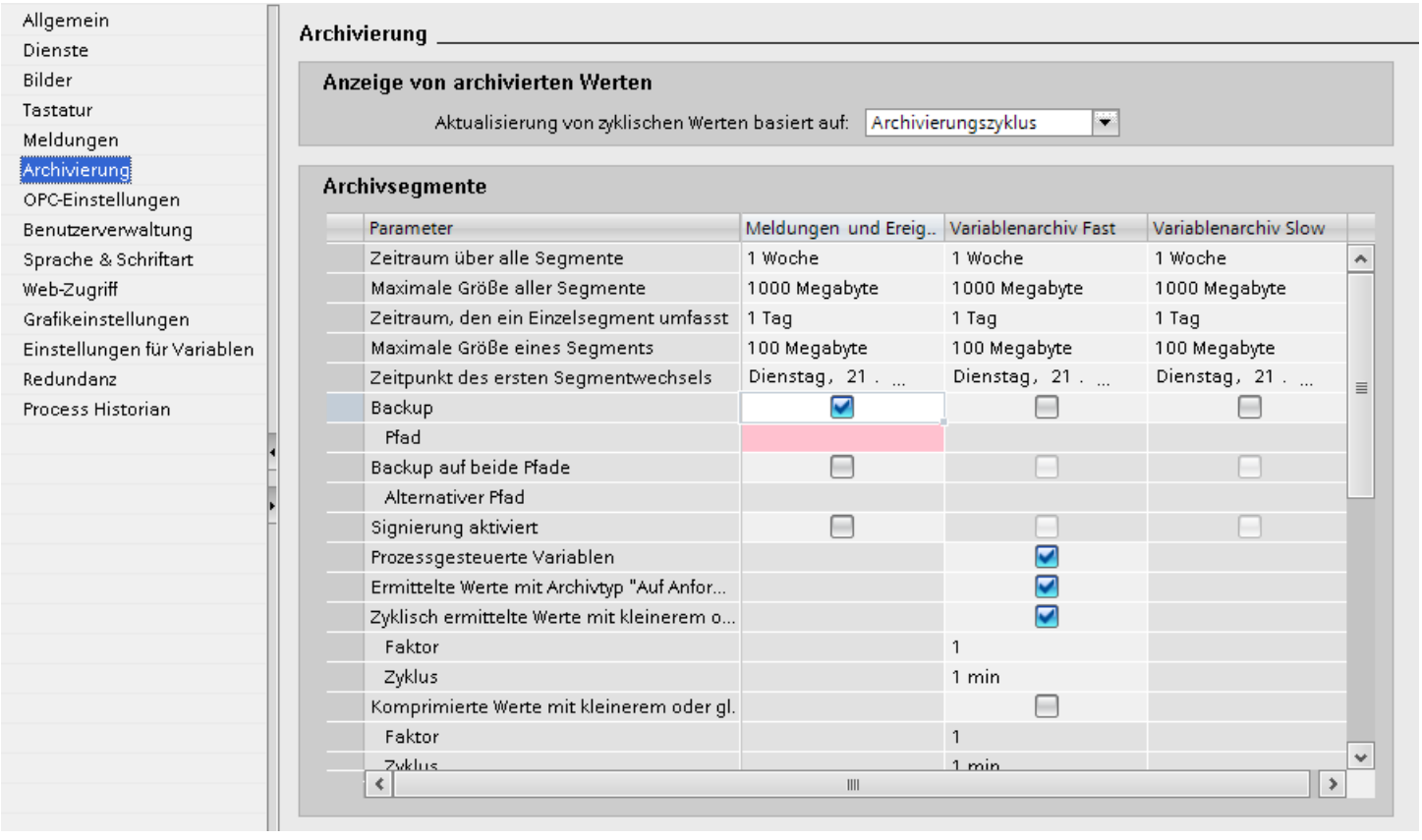

3. Klicken Sie im Inspektorfenster auf "Eigenschaften > Allgemein" und legen Sie folgende Einstellungen fest:
   - Geben Sie unter "Zeitraum über alle Segmente" und "Maximale Größe aller Segmente" die gewünschten Werte ein.
     Mit den Eingaben für den Zeitraum über alle Segmente und dessen maximale Größe legen Sie die Größe der Archivdatenbank fest. Wenn eines der beiden Kriterien überschritten wird, wird ein neues Segment begonnen. Wenn alle Segmente befüllt sind, wird das älteste Segment überschrieben.
   - Geben Sie unter "Zeitraum, den ein Einzelsegment umfasst" und "Maximale Größe eines Segments" die gewünschten Werte ein.
     Mit den Eingaben zum Zeitraum für die Einzelsegmente und dessen maximale Größe legen Sie die Größe und Anzahl der Einzelsegmente fest. Wenn eines der Kriterien überschritten wird, wird ein neues Einzelsegment begonnen. Wenn das Kriterium für "Zeitraum über alle Segmente" überschritten wird, wird zusätzlich das älteste Einzelsegment gelöscht.

– Wählen Sie unter "Zeitpunkt des ersten Segmentwechsels" das Startdatum und die Startzeit des ersten Segmentwechsels.
  Auch wenn die projektierte Größe nach Runtime-Start überschritten wird, wird das Segment gewechselt. In diesem Fall wird auch das älteste Einzelsegment gelöscht.

Meldeklasse mit Archivierung anlegen

Um eine Meldeklasse mit Archivierung anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "HMI-Meldungen".
2. Klicken Sie auf die Registerkarte "Meldeklassen".
3. Markieren Sie die Meldeklasse, deren Meldungen archiviert wird.
4. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Einstellungen > Archiv".
5. Ordnen Sie die Meldung, die archiviert wird, dieser Meldeklasse zu.

Beispiel zum Zeitpunkt des ersten Segmentwechsels

Ein Segment wird zum ersten Mal am 17. Januar um 23:59 Uhr gewechselt. Das nächste Mal wechselt das Segment ab dem projektierten Zeitpunkt mit dem unter "Zeitraum den ein Einzelsegment umfasst" angegebenen Zyklus. Mit der Einstellung "1 Tag" wird jeden Tag um 23:59 Uhr gewechselt. Wenn z. B. der Zyklus auf "1 Monat" festgelegt ist, findet der Segmentwechsel am 17. Tag des nächsten Monats um 23:59 Uhr statt.

Siehe auch

Einstellungen zur Archivierung (Seite 687)

Archivgröße und Segmentierung festlegen (Seite 688)

Arbeiten mit Archiven (Seite 1077)

3.3.2.3 Meldearchiv-Backup projektieren

Prinzip

Um eine lückenlose Dokumentation Ihres Prozesses sicherzustellen, erstellen Sie regelmäßige Backups Ihrer Archivdaten.

Hinweis

Das Backup wird standardmäßig eine Viertelstunde nach dem ersten zeitlich bedingten Segmentwechsel begonnen. Wenn das Backup und das Segment synchron mit Runtime starten sollen, legen Sie den Zeitpunkt für den Segmentwechsel vor den Start von Runtime.

In Runtime können Sie in die Meldeanzeige zu einer angezeigten Meldung einen Kommentar eingeben. Wenn das Archivsegment, in dem die Meldung gespeichert ist, bereits ausgelagert ist, wird der Kommentar nicht in das ausgelagerte Archiv übernommen. Der Kommentar liegt nur im lokalen Archivsegment.

Wenn das Archivsegment noch nicht ausgelagert ist, wird der geänderte Kommentar dauerhaft übernommen.

Voraussetzung

- Die Einstellungen zur Archivierung sind geöffnet.

Vorgehen

Um ein Backup der Archivsegmente anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Bereichsnavigation auf "Archivierung".
2. Aktivieren Sie "Meldearchiv > Backup".

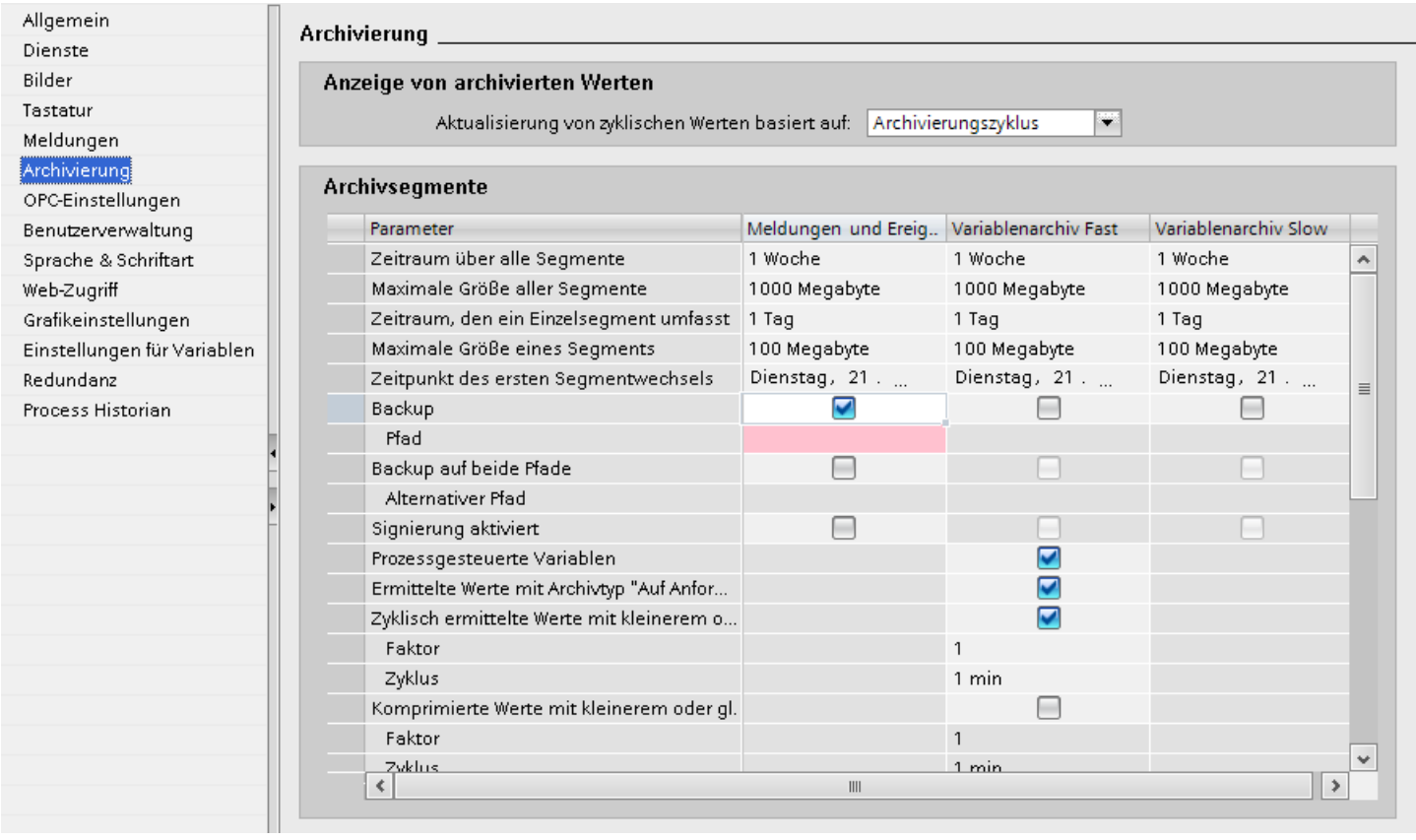

3. Klicken Sie im Inspektorfenster auf "Eigenschaften > Allgemein > Einstellungen Backup > Pfad".

4. Klicken Sie auf ... und suchen Sie nach einem vorhandenen Ordner für die Ablage der Backup-Datei. Legen Sie bei Bedarf mit "Make New Folder" einen neuen Ordner an. Als Ablageort ist auch ein Netzlaufwerk möglich.

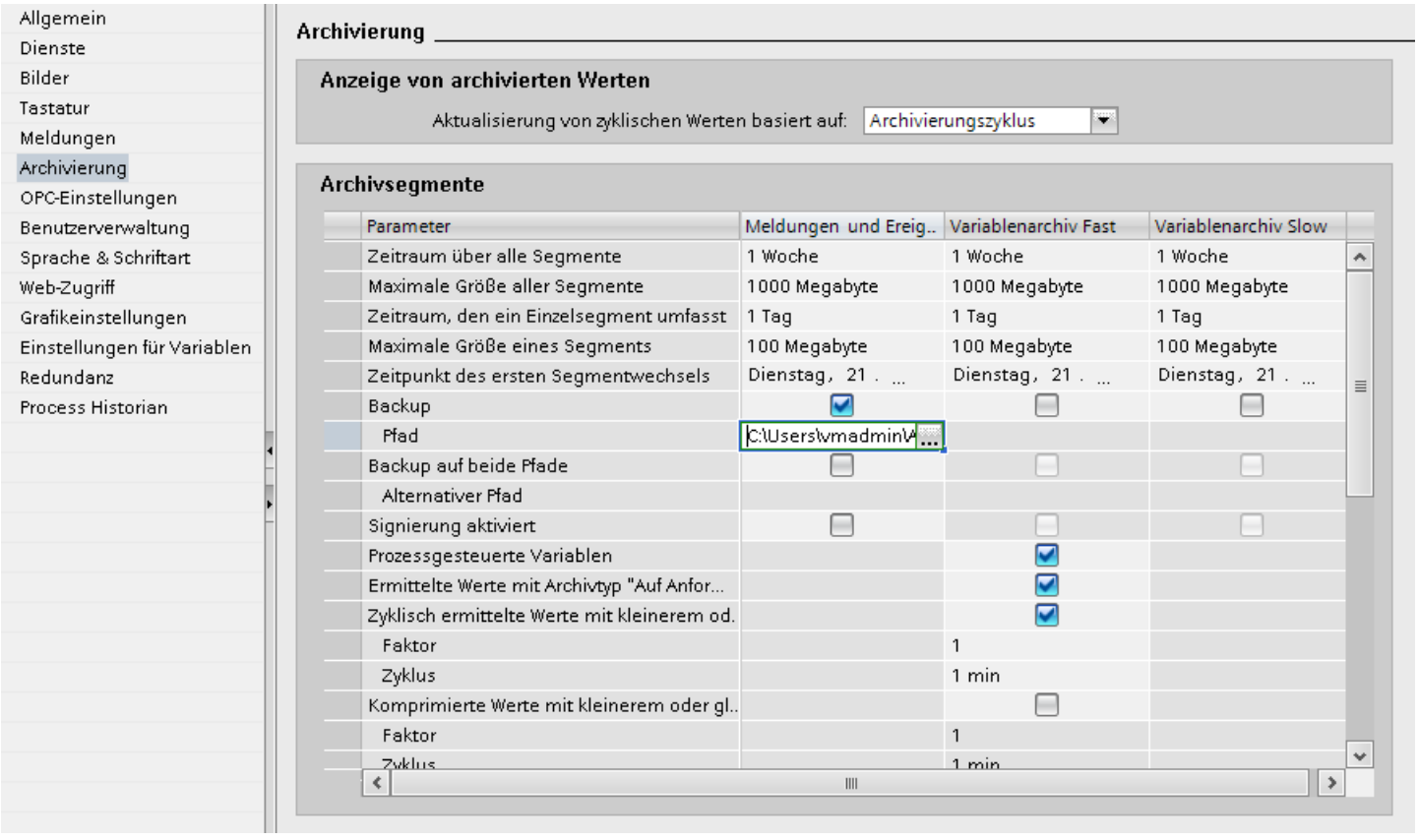

5. Wenn die archivierten Daten gesichert werden sollen, aktivieren Sie "Backup ".
   Wenn die archivierten Daten in den beiden Verzeichnissen "Pfad" und "Alternativer Pfad" gesichert werden sollen, aktivieren Sie zusätzlich "Backup auf beide Pfade".
   Als Ablageort ist auch ein Netzlaufwerk möglich.
   Ohne die Option "Backup auf beide Pfade" wird die zweite Backup-Datei in folgenden Fällen verwendet:
   - Wenn der Speicherplatz eines Backup-Mediums belegt ist.
   - Wenn der ursprüngliche Ablageort nicht verfügbar ist, z. B. wegen Netzwerkausfall.

   Wenn Sie entsprechende Systemmeldungen projektiert haben, werden Meldungen ausgegeben, wenn ein Ablageort nicht verfügbar ist.

6. Wenn die Backup-Datei eine Signatur erhalten soll, aktivieren Sie die Option "Signieren". Durch die Signatur wird beim erneuten Verbinden mit WinCC erkannt, ob in einer Backup-Datei nach ihrer Auslagerung geändert wurde.

Hinweis

Die Option "Signieren" belastet den Arbeitsspeicher. Wenn Sie die Signierung von Archiven einsetzen, darf die maximale Größe eines Einzelsegments 200 MByte nicht überschreiten.

Ergebnis

Das Backup des Archivsegments wird im angegebenen Ablageort gespeichert.

Aufbau der Backup-Datei

Eine Backup-Datei eines Meldearchivs ist folgendermaßen aufgebaut:

- Endung
  Ein Backup-Datei eines Archivsegments besteht aus zwei Dateien von den Formaten "*.ldf" und "*.mdf" Wenn Sie ein Backup auf einen anderen PC übertragen wollen, kopieren Sie beide Dateien.
- Dateiname
  Der Dateiname setzt sich wie folgt zusammen:
  "<Computername>_<Projektkennung>_<Typ>_<Zeitraum_von>_<Zeitraum_bis>"

Bestandteil	Erläuterung	Beispiel
Computername		FLEX-LOG2
Projektkennung	Die Projektkennung wird vom System generiert. Unterstriche werden dabei durch "#" ersetzt.	HMI#4ZQV
Typ	Die Typkennzeichnung des Meldearchivs ist "ALG" (Archive Log).	
Zeitraum_von	Der Zeitraum wird im Format yyyymmddhhmm angegeben.	200812021118 für den 2. Dezember 2008 um 11:18 Uhr
Zeitraum_bis		
Beispiel für den kompletten Dateinamen einer Backup-Datei: FLEX-LOG2_HMI#4ZQV_ALG_200811291414_200812030502.ldf		

Hinweis

Zuordnung zum Projekt

In der generierten Projektkennung im Dateinamen ist der Projektname nicht erkennbar.

Um die Backup-Dateien dem Projekt zuordnen zu können, vergeben Sie für den Ablageort einen Pfadnamen, der den Projektnamen enthält.

Signieren der Backup-Datei

Bei aktivierter Signierung und aktiviertem Backup wird jede Archiv-Backup-Datei bei ihrer Auslagerung signiert. Damit wird beim erneuten Verbinden der Datei mit WinCC festgestellt, ob die Datei nach der Auslagerung verändert wurde.

Wenn Sie die Signierung von Archiven einsetzen, darf die maximale Größe eines Einzelsegments 200 MByte nicht überschreiten.

Zur Verifizierung der Daten muss die Option "Signierung aktiviert" aktiviert sein. Bei aktivierter Signierung dauert es länger, die Archiv-Backup-Datei mit WinCC zu verbinden. Es werden maximal 500 Werte pro Sekunde archiviert.

Hinweis

Wenn Sie die Signierung deaktivieren, um z. B. eine schnelle Verbindung zu den Backup-Dateien aufzubauen, ist ein Segmentwechsel während der Deaktivierung zu vermeiden. Nach der Verbindung müssen Sie die Signierung wieder aktivieren, damit die zu archivierenden Daten eine Signatur erhalten.

Siehe auch

Arbeiten mit Archiven (Seite 1077)

3.3.2.4 Meldearchiv-Backup verbinden und trennen

Einleitung

Wenn Sie auf die Daten eines ausgelagerten Meldearchivs in Runtime zugreifen wollen, verbinden Sie das Archiv-Backup mit dem Projekt. Sie konfigurieren eine automatische Verbindung oder Sie verbinden das Meldearchiv über ein Skript mit dem Projekt. Die archivierten Meldungen werden in Runtime in der Meldeanzeige angezeigt.

Wenn Sie auf das Backup eines Archivsegments in Runtime nicht mehr zugreifen wollen, trennen Sie Archiv-Backup von dem Projekt.

Voraussetzung

- Die entsprechende Dateien der Formate "*. ldf" und "*.mdf" des Backups sind lokal abgelegt.
- Das Projekt befindet sich in Runtime.

Zeitbereich der Anzeige

In Runtime werden die Meldungen nur dann angezeigt, wenn Sie den Zeitbereich in der Meldeanzeige entsprechend konfiguriert haben.

Beispiel

Sie haben den Zeitbereich so konfiguriert, dass nur die Meldungen der letzten 24 Stunden angezeigt werden. Wenn Sie ein Archiv-Backup verbinden, in dem Meldungen gespeichert sind, die älter als 24 Stunden sind, werden sie in der Meldeanzeige nicht angezeigt.

Meldearchiv automatisch verbinden lassen

Um sich mit dem Meldearchiv-Backup automatisch zu verbinden, gehen Sie folgendermaßen vor:

1. Fügen Sie die Archiv-Backup-Dateien in den Ordner "RuntimeProjektPfad\Projektname \CommonArchiving" ein.
2. In Runtime wird das Meldearchiv automatisch mit dem Projekt verbunden.

Wenn die Signierung aktiviert ist, werden veränderte, signierte Archiv-Backup-Dateien nicht automatisch verbunden. Eine WinCC Systemmeldung wird erzeugt und in der Windows Ereignisanzeige unter "Anwendung" ein Eintrag hinzugefügt.

Meldearchiv über ein Skript verbinden

Sie verbinden die Archiv-Backup-Dateien über ein Skript über das VBS-Objekt "AlarmLogs" mit dem Projekt. Die Archivsegmente werden mit der VBS-Methode "Restore" in den Ordner "Common Archiving" des Runtime-Projekts kopiert. Weitere Informationen finden Sie in der Beschreibung des VBS-Objekts "AlarmLogs" und der VBS-Methode "Remove".

Meldearchiv automatisch trennen

Um das Meldearchiv-Backup automatisch vom Projekt zu trennen, gehen Sie folgendermaßen vor:

1. Navigieren Sie zum Ordner "RuntimeProjektPfad\Projektname\CommonArchiving".
2. Entfernen Sie die Archiv-Backup-Dateien aus dem Ordner.

Meldearchiv über ein Skript trennen

Sie trennen die Archiv-Backup-Dateien über ein Skript über das VBS-Objekt "AlarmLogs" vom Projekt. Die Archivsegmente werden mit der VBS-Methode "Remove" aus dem Ordner "Common Archiving" des Runtime-Projekts entfernt. Weitere Informationen finden Sie in der Beschreibung des VBS-Objekts "AlarmLogs" und der VBS-Methode "Remove".

3.3.2.5 Direktzugriff mit C-API/ODK auf die Archivdatenbank

Einleitung

Verschiedene Anbieter stellen Schnittstellen zur Verfügung, die Sie zum Zugriff auf Datenbanken nutzen können. Diese Schnittstellen ermöglichen Ihnen auch den Direktzugriff auf die Archivdatenbanken von WinCC. Im Direktzugriff lesen Sie z. B. Prozesswerte aus, um Sie dann in Tabellenkalkulationsprogrammen weiterzuverarbeiten.

Zugriff auf die Archivdatenbank mit C-API/ODK

Mit der Option "WinCC Open Development Kit" greifen Sie über offen gelegte Programmierschnittstellen auf Daten und Funktionen von WinCC zu.

Weitere Informationen

Weitere Informationen zu diesem Thema finden Sie in der Dokumentation zu WinCC Open Development Kit

3.3.2.6 Direktzugriff mit ADO/OLE DB auf die Archivdatenbank

Einleitung

Verschiedene Anbieter stellen Schnittstellen zur Verfügung, die Sie zum Zugriff auf Datenbanken nutzen können. Diese Schnittstellen ermöglichen Ihnen auch den Direktzugriff auf die Archivdatenbanken von WinCC. Im Direktzugriff lesen Sie z. B. Prozesswerte aus, um Sie dann in Tabellenkalkulationsprogrammen weiterzuverarbeiten.

Zugriff auf die Archivdatenbanken mit ADO/OLE DB

Die Prozesswerte werden in den Archivdatenbanken teilweise in komprimierter Form abgelegt. Um auf diese komprimierten Prozesswerte zuzugreifen, verwenden Sie den WinCC OLE DB Provider. Für nicht komprimierte Daten und Daten der Meldearchivierung nutzen Sie auch z. B. die ADO/OLE DB Schnittstellen von Microsoft. Als Programmiersprachen können Sie z. B. Visual Basic oder Visual C++ verwenden.

Hinweis

Beachten Sie beim Direktzugriff mit ADO/OLE DB auf die Archivdatenbank, dass sich die Struktur des Archivs in einer neuen WinCC-Version ändern kann.

Weitere Informationen

Weiterführende Informationen zu diesem Thema finden Sie im Internet unter:

msdn.microsoft.com (http://msdn.microsoft.com/de-de/library/default.aspx)

3.3.2.7 Rückladen von Meldungen nach Netzausfall projektieren

Einleitung

Um nach einem Netzausfall die letzten Meldungen wieder in Runtime anzuzeigen, projektieren Sie das Rückladen von Meldungen. Bei einem Netzausfall wird aus dem Meldearchiv eine von Ihnen projektierte Anzahl von Meldungen ausgelesen. Daraus wird das zuletzt anstehende Prozessabbild konstruiert.

Vorgehen

Um die Anzahl von Meldungen festzulegen, die nach einem Netzausfall rückgeladen werden, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter Ihrem Bediengerät auf "Runtime-Einstellungen".
2. Klicken Sie in der Bereichsnavigation auf "Meldungen".
3. Geben Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Meldelisteneinträge nach Stromausfall" die Anzahl Meldungen ein, die rückgeladen werden nach einem Stromausfall.

Ergebnis

Nach einem Netzausfall wird die von Ihnen projektierte Anzahl von Meldungen in die Meldeanzeige rückgeladen und in Runtime angezeigt.

3.4 Bedienen von Meldungen in Runtime

3.4.1 Meldungen in Runtime

Meldungen

Meldungen zeigen am Bediengerät Ereignisse und Zustände an, die in der Anlage bzw. im Prozess oder am Bediengerät auftreten. Ein Zustand wird bei dessen Eintritt gemeldet.

Für eine Meldung können folgende Meldeereignisse auftreten:

- Gekommen
- Gegangen
- Quittieren
- Loop-In-Alarm

Der Projekteur legt fest, welche Meldungen vom Benutzer quittiert werden müssen.

Eine Meldung kann folgende Informationen enthalten:

- Datum
- Uhrzeit
- Meldetext
- Störort
- Zustand
- Meldeklasse
- Meldenummer
- Meldegruppe

Meldeklassen

Meldungen sind verschiedenen Meldeklassen zugeordnet.

- "Warnings"
  Meldungen dieser Klasse zeigen normalerweise Zustände in der Anlage an, z. B. "Motor eingeschaltet". Meldungen dieser Meldeklasse müssen nicht quittiert werden.
- "Errors"
  Meldungen dieser Klasse müssen immer quittiert werden. Störmeldungen zeigen normalerweise kritische Störungen in der Anlage an, z. B. "Motortemperatur zu hoch".
- "System"
  Systemmeldungen zeigen Zustände oder Ereignisse des Bediengeräts selbst an. Systemmeldungen informieren z. B. über Fehlbedienungen oder Störungen in der Kommunikation.
- Benutzerdefinierte Meldeklassen
  Die Eigenschaften dieser Meldeklasse werden bei der Projektierung festlegt.

Meldepuffer

Die Meldeereignisse werden in einem internen Puffer gespeichert. Die Größe dieses Meldepuffers hängt vom Typ des Bediengeräts ab.

Mit der Einstellung "Persistenter Meldepuffer" schalten Sie den Meldepuffer in Runtime ein und aus. Aus Speicherplatzgründen kann es erforderlich sein, den Meldepuffer auszuschalten.

Standardmäßig ist die Einstellung "Persistenter Meldepuffer" unter "Runtime-Einstellungen > Meldungen" aktiviert. Bei dieser Einstellung zeigt die Meldeanzeige sowohl aktuelle Meldungen als auch Meldungen aus dem Meldepuffer.

Wenn Sie den Meldepuffer ausschalten, müssen Sie Runtime erneut starten, damit die Meldeanzeige die Meldungen aus dem Meldepuffer nicht mehr darstellt.

Meldeanzeige

In der Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer angezeigt. Über die Projektierung wird festgelegt, ob die auftretenden Meldungen quittiert werden müssen.

Meldefenster

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört.

Die Sortierung der angezeigten Meldungen ist projektierbar. Wahlweise kann die aktuelle oder die älteste Meldung zuerst angezeigt werden. Im Meldefenster kann weiterhin der genaue Störort mit Datum/Uhrzeit des Meldeereignisses angezeigt werden. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das am Bildschirm angezeigt wird, wenn eine Meldung der festgelegten Meldeklasse kommt.

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquittierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

3.4.2 Meldungen in Runtime

Meldungen

Meldungen zeigen am Bediengerät Ereignisse und Zustände an, die in der Anlage bzw. im Prozess oder am Bediengerät auftreten. Ein Zustand wird bei dessen Eintritt gemeldet.

Für eine Meldung können folgende Meldeereignisse auftreten:

- Gekommen
- Gegangen
- Quittieren

Der Projekteur legt fest, welche Meldungen vom Benutzer quittiert werden müssen.

Eine Meldung kann folgende Informationen enthalten:

- Datum
- Uhrzeit
- Meldetext
- Störort
- Zustand
- Meldeklasse
- Meldenummer
- Meldegruppe

Meldeklassen

Meldungen sind verschiedenen Meldeklassen zugeordnet.

- "Warnings"
  Meldungen dieser Klasse zeigen normalerweise Zustände in der Anlage an, z. B. "Motor eingeschaltet". Meldungen dieser Meldeklasse müssen nicht quittiert werden.
- "Errors"
  Meldungen dieser Klasse müssen immer quittiert werden. Störmeldungen zeigen normalerweise kritische Störungen in der Anlage an, z. B. "Motortemperatur zu hoch".
- "System"
  Systemmeldungen zeigen Zustände oder Ereignisse des Bediengeräts selbst an. Systemmeldungen informieren z.B. über Fehlbedienungen oder Störungen in der Kommunikation.
- "Diagnosis Events"
  SIMATIC-Diagnosemeldungen zeigen Zustände und Ereignisse der Steuerung SIMATIC S7 an.
- "Safety Warnings"
  Die Meldeklasse "Safety Warnings" zeigt Meldungen zum fehlersicheren Betrieb an. Meldungen dieser Meldeklasse quittiert der Benutzer nicht.

- STEP 7-Meldeklassen
  Die in STEP 7 projektierten Meldeklassen stehen auch dem Bediengerät zur Verfügung.
- Benutzerdefinierte Meldeklassen
  Die Eigenschaften dieser Meldeklasse werden bei der Projektierung festgelegt.

Meldepuffer

Die Meldeereignisse werden in einem internen Puffer gespeichert. Die Größe dieses Meldepuffers hängt vom Typ des Bediengeräts ab.

Hinweis

Gilt für Panels und RT Advanced

Mit der Einstellung "Persistenter Meldepuffer" schalten Sie den Meldepuffer in Runtime ein und aus. Aus Speicherplatzgründen kann es erforderlich sein, den Meldepuffer auszuschalten.

Standardmäßig ist die Einstellung "Persistenter Meldepuffer" unter "Runtime-Einstellungen > Meldungen" aktiviert. Bei dieser Einstellung zeigt die Meldeanzeige sowohl aktuelle Meldungen als auch Meldungen aus dem Meldepuffer.

Wenn Sie den Meldepuffer ausschalten, müssen Sie Runtime erneut starten, damit die Meldeanzeige die Meldungen aus dem Meldepuffer nicht mehr darstellt.

Meldeprotokoll

Wenn das Protokollieren von Meldungen im Projekt aktiviert wurde, werden Meldungen direkt auf dem angeschlossenen Drucker ausgegeben.

Für jede Meldung kann einzeln projektiert sein, ob sie protokolliert werden soll. Der Druck einer solchen Meldung wird veranlasst, wenn die Meldeereignisse "Gekommen" und "Gegangen" eintreten.

Falls Meldungen der Meldeklasse "System" gedruckt werden sollen, muss dies über den zugehörigen Meldepuffer veranlasst werden. Dann wird der Meldepufferinhalt vollständig ausgedruckt. Zu diesem Zweck muss im Projekt ein Bedienobjekt, um den Druck auszulösen, projektiert worden sein.

Hinweis

Gilt für Panels und RT Advanced

Wenn das Protokollieren von Meldungen im Projekt aktiviert wurde, werden die Meldungen erst gedruckt, wenn die Runtime beendet wurde.

Meldearchiv

Wenn ein Meldearchiv projektiert ist, werden die Meldeereignisse in diesem Meldearchiv gespeichert. Die Kapazität des Archivs ist durch das Speichermedium und die Systemgrenzen begrenzt.

Meldeanzeige

In der Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv angezeigt. Über die Projektierung wird festgelegt, ob die auftretenden Meldungen quittiert werden müssen. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

Meldefenster

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.

Die Sortierung der angezeigten Meldungen ist projektierbar. Entweder die aktuelle oder die älteste Meldung wird zuerst angezeigt. Im Meldefenster kann weiterhin der genaue Störort mit Datum/Uhrzeit des Meldeereignisses angezeigt werden. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das am Bildschirm angezeigt wird, wenn eine Meldung der festgelegten Meldeklasse kommt.

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquittierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

3.4.3 Bedienung Meldefenster, Meldeanzeige

3.4.3.1 Meldefenster, Meldeanzeige in Runtime

Verwendung

Meldungen werden am Bediengerät in der Meldeanzeige oder Meldefenster dargestellt. Aussehen und Bedienung des Meldefensters entsprechen dem der Meldeanzeige.

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.

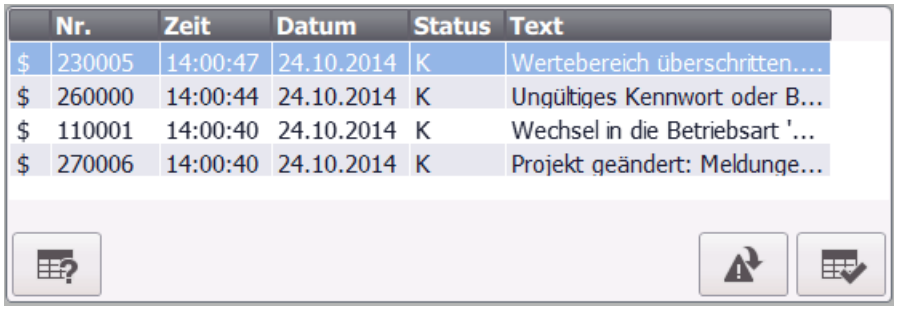

Darstellung

Abhängig von der Projektierung werden in der Meldeanzeige verschiedene Spalten mit Informationen zu einer Meldung oder einem Meldeereignis angezeigt. Wenn ein Filter projektiert ist, so werden nur Meldungen angezeigt, die im Meldetext eine bestimmte Zeichenkette enthalten.

Zur Unterscheidung der verschiedenen Meldeklassen, enthält die erste Spalte in der Meldeanzeige ein Symbol:

Symbol	Meldeklasse
!	"Errors"
Leer	"Warnings"
abhängig von Projektierung	Benutzerdefinierte Meldeklassen
$	"System"
S7	"Diagnosis Event"
!!	"Safety Warnings"

Bedienung

Abhängig von der Projektierung bedienen Sie die Meldeanzeige wie folgt:

- Die Spaltenreihenfolge verändern.
- Die Sortierung der dargestellten Meldungen verändern.
- Meldungen quittieren
- Meldungen bearbeiten

Bedienelemente

Die Schaltflächen haben folgende Funktionen:

Schaltfläche	Funktion
	Tooltip für eine Meldung anzeigen
	Loop-In-Alarm Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1]
	Meldung quittieren

[1] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Verhalten bei Bedienung

Gebundene Meldefenster bei Tastengeräten

Bei der Projektierung des Meldefensters für Tastengeräte aktivieren Sie unter "Eigenschaften>Modus" die Eigenschaft "Gebunden" (modal). Damit wird sichergestellt, dass das Meldefenster bei Bildwechsel den Fokus nicht verliert. Das Hin- und Herschalten zwischen Bild und verschiedenen Fenstern mit <Strg+Tab> wird nicht unterstützt. Wenn das gebundene Meldefenster den Fokus besitzt, dann lassen sich die Schaltflächen im dahinterliegenden Bild nicht betätigen. Die an eine Funktionstaste projektierten Funktionen werden ausgeführt.

Sortierung der angezeigten Meldungen ändern

Durch Klicken auf die Spalte werden zuerst quittierpflichtige Meldungen nach Datum und Uhrzeit sortiert. Danach werden die nicht quittierpflichtigen Meldungen nach Datum und Uhrzeit sortiert.

3.4.3.2 Meldefenster, Meldeanzeige bedienen

Einleitung

Alternativ zur Bedienung mit der Maus bedienen Sie die Meldeanzeige und das Meldefenster mit der Taste <TAB> Ihres Bediengeräts. Damit wählen Sie die Schaltflächen und die zuletzt markierte Meldung der Meldeanzeige aus. Abhängig von der Projektierung bedienen Sie die Meldeanzeige auch über die Funktionstasten.

Bedienung mit der Maus

1. Markieren Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche, deren Funktion Sie ausführen wollen.

Bedienung mit der Tastatur

1. Drücken Sie sooft die Taste <Tab>, bis in der Meldeanzeige die Liste der angezeigten Meldungen markiert ist.
2. Wählen Sie die zu bearbeitende Meldung. Verwenden Sie dazu die Tasten <Auf> und <Ab>.
3. Drücken Sie sooft die Taste <Tab>, bis die Schaltfläche markiert ist, dessen Funktion Sie ausführen wollen.
4. Drücken Sie die Taste <Enter>.

Reihenfolge der Spalten ändern

1. Wählen Sie die Spaltenüberschrift, z. B. die Spaltenüberschrift "Datum".
2. Ziehen Sie die Spaltenüberschrift bei gedrückter Maustaste auf die Spaltenüberschrift "Zeit".
   Die Spalte "Datum" ist vor der Spalte "Zeit" angeordnet.

Sortierung ändern

Sie können die Liste nach Datum oder nach Uhrzeit sortieren.

1. Klicken Sie auf die entsprechende Spaltenüberschrift.
   Die Liste wird in aufsteigender bzw. absteigender Reihenfolge nach diesem Kriterium sortiert.
2. Durch erneutes Klicken auf dieselbe Spaltenüberschrift kehren Sie die Sortierreihenfolge um.

Meldung quittieren

1. Wählen Sie die zu quittierende Meldung.
2. Klicken Sie auf die Schaltfläche .

Loop-In-Alarm triggern

1. Wählen Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche .
   Das Bild, das Informationen zur Meldung enthält, wird angezeigt.

   Hinweis

   Wenn Sie den Loop-In-Alarm bei einer unquittierten Meldung triggern, wird diese automatisch quittiert.

Projektierten Tooltip anzeigen

1. Wählen Sie die zu betreffende Meldung.
2. Klicken Sie auf die Schaltfläche .
   Der projektierte Tooltip zur Meldung wird angezeigt.

3.4.4 Bedienung einfaches Meldefenster, Meldeanzeige

3.4.4.1 Einfache Meldeanzeige, Meldefenster in Runtime

Verwendung

In der einfachen Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer angezeigt. Aussehen und Bedienung des einfachen Meldefensters entsprechen dem der einfachen Meldeanzeige.

Hinweis

Das Objekt "Einfache Meldeanzeige" wird nicht mit einem Skript dynamisiert.

Im Engineering System dynamisieren Sie im Inspektorfenster im Register "Animationen", z. B. die Sichtbarkeit eines Objekts. Das Objekt "Einfache Meldeanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projektiert haben und z. B. eine Konsistenzprüfung des Projekts ausführen, dann wird im Ausgabefenster eine Fehlermeldung ausgegeben.

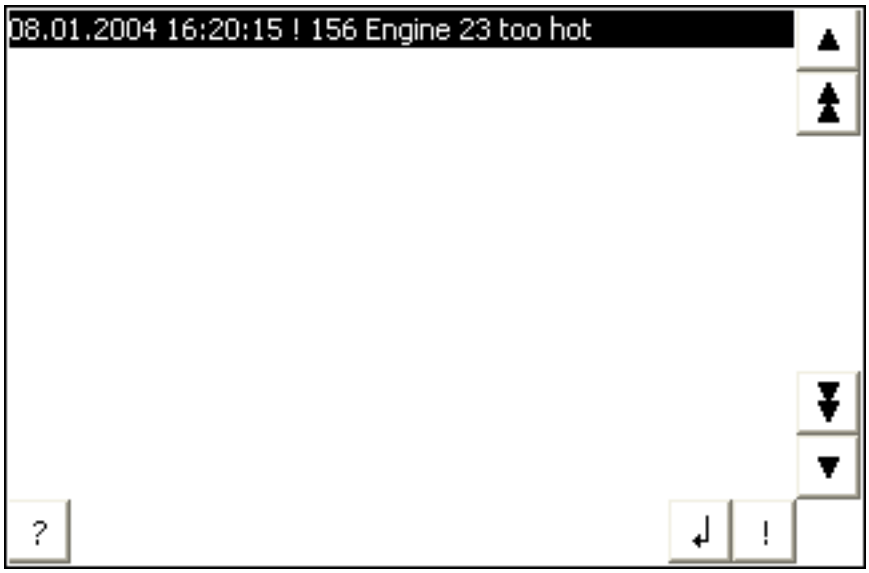

Darstellung

Abhängig von der Projektierung werden in der Meldeanzeige verschiedene Spalten mit Informationen zu einer Meldung oder einem Meldeereignis angezeigt.

Zur Unterscheidung der verschiedenen Meldeklassen enthält die erste Spalte in der Meldeanzeige ein Symbol:

Symbol	Meldeklasse
!	"Errors"
leer	"Warnings"
abhängig von Projektierung	Benutzerdefinierte Meldeklassen
$	"System"

Bedienung

Abhängig von der Projektierung bedienen Sie die Meldeanzeige wie folgt:

- Meldungen quittieren
- Meldungen bearbeiten

Bedienelemente

Die Schaltflächen haben folgende Funktionen:

Schaltfläche	Funktion
!	Meldung quittieren
↲	Loop-In-Alarm Führt bei jedem Tastendruck die projektierte Funktion (z. B.: "AktiviereBild") aus. Dabei ist es möglich, ein Bild aufzurufen, das Ihre Informationen zur markierten Meldung enthält. Quittiert außerdem eine quittierpflichtige Meldung. [1]
?	Tooltip für eine Meldung anzeigen
▶	Anzeige des kompletten Meldetexts zur ausgewählten Meldung in einem separaten Fenster, dem Meldetextfenster Im Meldetextfenster zeigen Sie Meldetexte an, die mehr Platz erfordern, als in der Meldeanzeige vorhanden ist. Schließen Sie das Meldetextfenster mit der Schaltfläche ✕.
▲	Blättert um eine Meldung nach oben.
⏶	Blättert in der Meldeanzeige seitenweise nach oben.
⏷	Blättert in der Meldeanzeige seitenweise nach unten.
▼	Blättert um eine Meldung nach unten.

[1] Nur wenn bei der markierten Meldung beim Ereignis "Loop-In-Alarm" eine Funktion projektiert ist.

Darstellung der Bedienelemente

Die Darstellung der Schaltflächen zur Bedienung der einfachen Meldeanzeige ist abhängig von der projektierten Größe. Überprüfen Sie daher auf dem Bediengerät, ob alle benötigten Schaltflächen zur Verfügung stehen.

3.4.4.2 Einfache Meldeanzeige, Meldefenster bedienen

Einleitung

Alternativ zur Bedienung mit der Maus bedienen Sie die einfache Meldeanzeige mit der Taste <Tab> Ihres Bediengeräts. Damit wählen Sie die Schaltflächen und die zuletzt markierte Meldung der Meldeanzeige aus. Abhängig von der Projektierung bedienen Sie die Meldeanzeige auch über die Funktionstasten.

Bedienung mit der Maus

1. Markieren Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche, deren Funktion Sie ausführen wollen.

Bedienung mit der Tastatur

1. Drücken Sie sooft die Taste <Tab>, bis in der Meldeanzeige die Liste der angezeigten Meldungen selektiert ist.
2. Wählen Sie die zu bearbeitende Meldung. Verwenden Sie dazu die Tasten <Auf> und <Ab>.
3. Drücken Sie sooft die Taste <Tab>, bis die Schaltfläche markiert ist, deren Funktion Sie ausführen wollen.
4. Drücken Sie die Taste <Enter>.

Meldung quittieren

1. Markieren Sie die zu quittierende Meldung.
2. Klicken Sie auf die Schaltfläche !.

Loop-In-Alarm triggern

1. Markieren Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche ↵.
   Das Bild, das Informationen zur Meldung enthält, wird angezeigt.

Hilfetext aufrufen

1. Markieren Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche ?.
3. Schließen Sie das Fenster zur Anzeige des Hilfetexts mit der Schaltfläche ✕ oder mit der Tastenkombination <Alt+F4>.

3.4.5 Bedienung Meldeindikator

3.4.5.1 Meldeindikator in Runtime

Verwendung

Wenn Meldungen der festgelegten Meldeklasse anstehen oder quittiert werden müssen, wird der Meldeindikator angezeigt.

Darstellung

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquittierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

Bedienung

Abhängig von der Projektierung wird bei Bedienung des Meldeindikators ein Meldefenster geöffnet. Der Meldeindikator ist nur per Maus oder Touch-Screen bedienbar.

Hinweis

Geräteabhängigkeit

Die Bedienung mit der Maus ist nicht bei allen Bediengeräten möglich.

3.4.5.2 Meldeindikator mit der Maus bedienen

Bedienung mit der Maus

1. Klicken Sie mit dem Mauszeiger auf den Meldeindikator. Abhängig von der Projektierung wird das Meldefenster geöffnet.
2. Um das Meldefenster zu schließen, klicken Sie auf das Symbol ✕. Durch Klicken auf den Meldeindikator öffnen Sie das Meldefenster wieder.

3.4.6 Meldungen quittieren

Einleitung

Meldungen quittieren Sie in Runtime abhängig von der Projektierung Ihres Projekts. Sie quittieren Meldungen folgendermaßen:

- Mit den Schaltflächen der Anzeige- und Bedienobjekte
- Mit der Taste "ACK" an Ihrem Bediengerät
- Mit individuell projektierte Funktionstasten oder Schaltflächen

Wenn eine Bedienberechtigung für die Bedienelemente projektiert ist, dann sind die Meldungen nur von berechtigten Benutzern quittierbar.

Um Meldungen in Runtime automatisch zu quittieren, verwenden Sie Systemfunktionen und Skripte sowie die Option "Quittieren durch die Steuerung".

Hinweis

Geräteabhängigkeit

Skripte stehen nicht für alle Bediengeräte zur Verfügung.

Varianten der Quittierung

Sie quittieren in Runtime einzelne Meldungen oder mehrere Meldungen gemeinsam. Sie unterscheiden dabei folgendermaßen:

- Einzelquittierung
  Quittieren einer Meldung über eine Schaltfläche oder Funktionstaste.
- Meldegruppen quittieren
  Quittierung aller Meldungen einer Meldegruppe über eine Schaltfläche oder Funktionstaste.

Voraussetzung

- Eine Meldung wird am Bediengerät angezeigt.

Vorgehen

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung.
2. Wenn Sie eine erweiterte Meldeanzeige oder ein Meldefenster verwenden, klicken Sie auf die Schaltfläche .
3. Wenn Sie eine einfache Meldeanzeige oder ein einfaches Meldefenster verwenden, klicken Sie auf die Schaltfläche !.
4. Um eine Meldung in einer Meldezeile zu quittieren, drücken Sie die Taste "ACK" an Ihrem Bediengerät.

Hinweis

Geräteabhängigkeit

Die erweiterte Meldeanzeige oder Meldefenster und die Meldezeile stehen nicht für alle Bediengeräte zur Verfügung.

Ergebnis

Die Meldung erhält den Status "quittiert". Wenn die Bedingung zum Auslösen einer Meldung nicht mehr zutrifft, erhält die Meldung zusätzlich den Zustand "gegangen" und wird am Bediengerät nicht mehr angezeigt.

3.4.7 Meldungen filtern

Einleitung

In der Meldeanzeige ist in Runtime kein Bedienelement für die Eingabe von Kriterien zum Filtern enthalten. Projektieren Sie dazu im Editor "Bilder" ein E/A-Feld.

Voraussetzung

- Ein Filter wurde für die Meldeanzeige projektiert.
- Ein E/A-Feld zur Eingabe des Kriteriums wird angezeigt.
- In Runtime werden Meldungen angezeigt.

Filter aktivieren

1. Geben Sie in das E/A-Feld die Zeichenkette als Kriterium ein.
2. Bestätigen Sie Ihre Eingabe.
   In der Meldeanzeige werden nur die Meldungen angezeigt, welche die eingegebene Zeichenkette enthalten. Dabei wird zwischen Groß- und Kleinschreibung unterschieden.

Filter deaktivieren

1. Löschen Sie die Zeichenkette im E/A-Feld, das zum Filtern der Meldeanzeige vorgesehen ist.
2. Bestätigen Sie Ihre Eingabe.
   In der Meldeanzeige werden wieder alle Meldungen angezeigt.

3.5 Bedienen von Meldungen in Runtime

3.5.1 Meldungen in Runtime

Meldungen

Meldungen zeigen am Bediengerät Ereignisse und Zustände an, die in der Anlage bzw. im Prozess oder am Bediengerät auftreten. Ein Zustand wird bei dessen Eintritt gemeldet.

Für eine Meldung können folgende Meldeereignisse auftreten:

- Gekommen
- Gegangen
- Quittieren

Der Projekteur legt fest, welche Meldungen vom Benutzer quittiert werden müssen.

Eine Meldung kann folgende Informationen enthalten:

- Datum
- Uhrzeit
- Meldetext
- Störort
- Zustand
- Meldeklasse
- Meldenummer
- Meldegruppe

Meldeklassen

Meldungen sind verschiedenen Meldeklassen zugeordnet.

- "Warnings"
  Meldungen dieser Klasse zeigen normalerweise Zustände in der Anlage an, z. B. "Motor eingeschaltet". Meldungen dieser Meldeklasse müssen nicht quittiert werden.
- "Errors"
  Meldungen dieser Klasse müssen immer quittiert werden. Störmeldungen zeigen normalerweise kritische Störungen in der Anlage an, z. B. "Motortemperatur zu hoch".
- "System"
  Systemmeldungen zeigen Zustände oder Ereignisse des Bediengeräts selbst an. Systemmeldungen informieren z.B. über Fehlbedienungen oder Störungen in der Kommunikation.
- "Diagnosis Events"
  SIMATIC-Diagnosemeldungen zeigen Zustände und Ereignisse der Steuerung SIMATIC S7 an.

- "Safety Warnings"
  Die Meldeklasse "Safety Warnings" zeigt Meldungen zum fehlersicheren Betrieb an. Meldungen dieser Meldeklasse quittiert der Benutzer nicht.
- STEP 7-Meldeklassen
  Die in STEP 7 projektierten Meldeklassen stehen auch dem Bediengerät zur Verfügung.
- Benutzerdefinierte Meldeklassen
  Die Eigenschaften dieser Meldeklasse werden bei der Projektierung festgelegt.

Meldepuffer

Die Meldeereignisse werden in einem internen Puffer gespeichert. Die Größe dieses Meldepuffers hängt vom Typ des Bediengeräts ab.

Hinweis

Gilt für Panels und RT Advanced

Mit der Einstellung "Persistenter Meldepuffer" schalten Sie den Meldepuffer in Runtime ein und aus. Aus Speicherplatzgründen kann es erforderlich sein, den Meldepuffer auszuschalten.

Standardmäßig ist die Einstellung "Persistenter Meldepuffer" unter "Runtime-Einstellungen > Meldungen" aktiviert. Bei dieser Einstellung zeigt die Meldeanzeige sowohl aktuelle Meldungen als auch Meldungen aus dem Meldepuffer.

Wenn Sie den Meldepuffer ausschalten, müssen Sie Runtime erneut starten, damit die Meldeanzeige die Meldungen aus dem Meldepuffer nicht mehr darstellt.

Meldeprotokoll

Wenn das Protokollieren von Meldungen im Projekt aktiviert wurde, werden Meldungen direkt auf dem angeschlossenen Drucker ausgegeben.

Für jede Meldung kann einzeln projektiert sein, ob sie protokolliert werden soll. Der Druck einer solchen Meldung wird veranlasst, wenn die Meldeereignisse "Gekommen" und "Gegangen" eintreten.

Falls Meldungen der Meldeklasse "System" gedruckt werden sollen, muss dies über den zugehörigen Meldepuffer veranlasst werden. Dann wird der Meldepufferinhalt vollständig ausgedruckt. Zu diesem Zweck muss im Projekt ein Bedienobjekt, um den Druck auszulösen, projektiert worden sein.

Hinweis

Gilt für Panels und RT Advanced

Wenn das Protokollieren von Meldungen im Projekt aktiviert wurde, werden die Meldungen erst gedruckt, wenn die Runtime beendet wurde.

Meldearchiv

Wenn ein Meldearchiv projektiert ist, werden die Meldeereignisse in diesem Meldearchiv gespeichert. Die Kapazität des Archivs ist durch das Speichermedium und die Systemgrenzen begrenzt.

Meldeanzeige

In der Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv angezeigt. Über die Projektierung wird festgelegt, ob die auftretenden Meldungen quittiert werden müssen. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

Meldefenster

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.

Die Sortierung der angezeigten Meldungen ist projektierbar. Entweder die aktuelle oder die älteste Meldung wird zuerst angezeigt. Im Meldefenster kann weiterhin der genaue Störort mit Datum/Uhrzeit des Meldeereignisses angezeigt werden. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das am Bildschirm angezeigt wird, wenn eine Meldung der festgelegten Meldeklasse kommt.

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquittierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

3.5.2 Meldeanzeige

Einleitung

Das Objekt "Meldeanzeige" zeigt Meldungen an, die während des Prozesses in einer Anlage auftreten. Die Meldeanzeige nutzen Sie auch, um Meldungen in Listen darzustellen. WinCC bietet verschiedene Anzeigen an, z. B. Anzeige "Aktuelle Meldungen" oder "Kurzzeitarchiv".

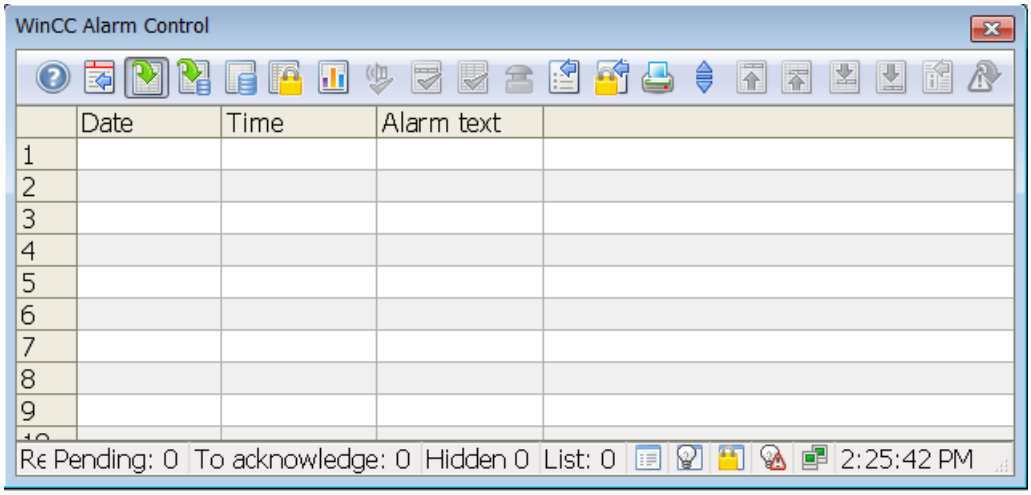

In Runtime können Sie mit der Taste <TAB> nacheinander auf verschiedene Objekte springen, um Werte einzugeben oder Schaltflächen zu klicken. Alternativ bedienen Sie die Meldeanzeige mit der Maus.

Hinweis

Eine Meldung bekommt in der Meldeanzeige einen durchgestrichenen Datum- und Zeitstempel zugeordnet, wenn einer der folgenden Fälle eintritt:

- Eine gesperrte Meldung wird wieder freigeben.
- Eine Meldung wird bei Netzausfall wieder rückgeladen. Dies gilt nur für zeitfolgerichtiges Melden.
- Die AS wird neu gestartet. Dies gilt nur für zeitfolgerichtiges Melden.

Voraussetzung

- Die Schaltfläche "Loop-In-Alarm" ist in der Meldeanzeige projektiert.
- Der Cursormodus ist für das Bild auf "Schaltcursor" eingestellt.
- Die Objekte sind zur Bedienung freigegeben.
- Die Bedienberechtigung ist erteilt.

Bedienung mit der Maus

1. Klicken Sie auf die zu bearbeitende Meldung.
2. Klicken Sie auf das Bedienelement, dessen Funktion Sie ausführen möchten.

Bedienung mit der Tastatur

1. Drücken Sie so oft die Taste <Tab>, bis in der Meldeanzeige die Liste der angezeigten Meldungen selektiert ist.
2. Wählen Sie die zu bearbeitende Meldung. Dazu verwenden Sie die Tasten <Auf> und <Ab>.
3. Drücken Sie so oft die Taste <Tab>, bis das Bedienelement selektiert ist, dessen Funktion Sie ausführen möchten.
4. Drücken Sie die Taste <Enter>.

Alternative Bedienung

Abhängig von der Projektierung bedienen Sie die Meldeanzeige auch über die Funktionstasten.

Beispiel: Reihenfolge der Spalten ändern

Diese Funktion ist projektierbar. Sie steht in Runtime nur zur Verfügung, wenn Sie im Inspektorfenster der Meldeanzeige "Schriftart Tabelle > Überschrift Spalten > Sortieren" aktiviert haben.

1. Wählen Sie die Spaltenüberschrift, z. B. die Spaltenüberschrift "Datum".
2. Ziehen Sie die Spaltenüberschrift bei gedrückter Maustaste auf die Spaltenüberschrift "Zeit".

Beispiel: Sortierung der Spalten ändern

Diese Funktion ist projektierbar. Sie steht in Runtime nur zur Verfügung, wenn Sie im Inspektorfenster der Meldeanzeige "Schriftart Tabelle > Überschrift Spalten > Sortieren nach Spaltenüberschriften" aktiviert haben.

1. Klicken Sie auf die Spaltenüberschrift.
2. Durch erneutes Klicken auf diese Spaltenüberschrift kehren Sie die Sortierreihenfolge um.

Beispiel: Loop-In-Alarm

Loop-In-Alarm löst den projektierten Bildwechsel zu dem Bild aus, das Informationen zur Meldung enthält.

1. Wählen Sie die zu bearbeitende Meldung.
2. Klicken Sie auf die Schaltfläche "Loop-In-Alarm"

Verhalten bei Bedienung

Beim Auslösen von Loop-In-Alarm einer unquittierten Meldung wird diese automatisch quittiert.

Siehe auch

Meldeanzeige (Seite 435)

3.5.3 Schaltflächen der Meldeanzeige

Einleitung

Um dem Benutzer eine bessere Übersicht über die laufenden Meldungen zu geben, zeigt die Meldeanzeige spezifische Meldelisten an. Die Listen filtern und sortieren Meldungen nach bestimmten Eigenschaften. Diese Eigenschaften und die Darstellung der Listen legen Sie beim Projektieren der Meldeanzeige individuell fest.

Listen in der Meldeanzeige

Sie haben die Möglichkeit, sechs verschiedene Listen im Meldefenster anzeigen zu lassen. Um die Meldelisten in der Meldeanzeige anzuzeigen, klicken Sie auf die entsprechende Schaltfläche in der Symbolleiste der Meldeanzeige.

	Liste	Beschreibung
	Meldeliste	Zeigt die anstehenden Meldungen an.
	Kurzzeitarchivliste	Zeigt die archivierten Meldungen an. Bei neu eingehenden Meldungen wird die Anzeige sofort aktualisiert.
	Langzeitarchivliste	Zeigt die archivierten Meldungen an. Bei neu eingehenden Meldungen wird die Anzeige nicht sofort aktualisiert.
	Sperrliste	Zeigt die gesperrten Meldungen an.
	Meldestatistik	Zeigt statistische Informationen an.
	Liste der ausgeblendeten Meldungen	Zeigt die ausgeblendeten Meldungen an. Auch ausgeblendete Meldungen werden archiviert.

Hinweis

Ausgeblendete Systemdiagnosemeldungen

Für ausgeblendete Systemdiagnose-Meldungen werden die Meldetexte in der Liste der ausgeblendeten Meldungen nicht angezeigt.

Die Liste der ausgeblendeten Meldungen stellt lediglich die Konfigurationsdaten der Meldung dar und keine Life-Ansicht.

Um aktive ausgeblendete Systemdiagnose-Meldungen anzuzeigen, müssen Sie den Filter für die Online-Meldeliste anpassen.

Gefilterte Meldeliste

Mit dem Filter der Meldeanzeige projektieren Sie Meldeanzeigen, in denen Sie individuelle Meldelisten anzeigen. Sie können für jede beliebige Meldeliste einen Filter projektieren. Für jede gefilterte Meldeliste projektieren Sie eine eigene Meldeanzeige.

Hinweis

Eine Meldung bekommt in der Meldeliste einen durchgestrichenen Datum- und Zeitstempel zugeordnet, wenn einer der folgenden Fälle eintritt:

- Eine gesperrte Meldung wird wieder freigegeben.
- Eine Meldung wird bei Netzausfall wieder rückgeladen; nur für zeitfolgerichtiges Melden.
- Das AS wird neu gestartet; nur für zeitfolgerichtiges Melden.

Siehe auch

Meldungen in Runtime filtern (Seite 971)

3.5.4 Meldungen in Runtime filtern

Einleitung

Welche Meldungen Sie in der Meldeanzeige darstellen wollen, definieren Sie in Runtime über Kriterien. Im folgenden Beispiel werden nur die Meldungen angezeigt, die den Meldetext "Motor an" beinhalten.

Voraussetzung

Die Schaltfläche "Meldungen filtern" ist in der Meldeanzeige projektiert.

Vorgehen

Um Meldungen in der Meldeanzeige zu filtern, gehen Sie folgendermaßen vor:

1. Klicken Sie in Runtime auf die Schaltfläche "Meldungen filtern".
   Der Dialog "Selektion festlegen" wird geöffnet.
2. Doppelklicken Sie in der Baumansicht auf den Ordner "Textblöcke" und klicken Sie auf "Meldetext". Aktivieren Sie im rechten Fenster die Option "Suchtext" und doppelklicken Sie auf "Suchtext".
3. Der Dialog "Texteingabe" öffnet sich. Geben Sie den Suchtext "Motor an" ein und bestätigen Sie mit "OK".
4. Bestätigen Sie im Dialog "Selektion festlegen" Ihre Eingaben mit "OK".
   Die Meldeanzeige zeigt nur noch die ausgewählten Meldungen an.

Bei einigen Meldeblöcken, z. B. den Systemblöcken "Datum" und "Uhrzeit", legen Sie im Dialog "Selektion festlegen" Beginn- und Endzeitpunkte oder Suchtexte fest. Die Eingabe muss dem im Dialog geforderten Format entsprechen.

Im Dialog "Selektion festlegen" stehen folgende Einstellungen zur Verfügung:

Feld	Beschreibung
Textfilter auf exakte Gleichheit	Wenn die Option "Textfilter auf exakte Gleichheit" nicht aktiviert ist, werden alle Textblöcke ausgewählt, welche die angegebene Zeichenkette enthalten. Wenn die Option "Textfilter auf exakte Gleichheit" aktiviert ist, werden alle Textblöcke ausgewählt, die der angegebenen Zeichenkette entsprechen.
Persistenz in RT	Wenn die Option "Persistent in RT" aktiviert ist, bleiben Änderungen an den Kriterien auch nach einem Bildwechsel erhalten.
Persistenz in CS und RT	Wenn die Option "Persistent in RT und CS" aktiviert ist, werden die veränderten Einstellungen auch ins Engineering System übernommen. Dazu müssen Sie das Bild im Editor "Bilder" öffnen und erneut speichern. Die veränderten Einstellungen werden auch beim erneuten Aktivieren des Projekts verwendet.
Filter löschen	Über diese Schaltfläche löschen Sie alle projektierten Kriterien.

Filtern nach Prozesswerten

Die Eingabe von Text als Kriterium für einen Prozesswert ist nicht möglich.

Um nach dem Text eines Parameterblocks oder der dargestellten Prozessvariablen zu filtern, gehen Sie wie folgendermaßen vor:

1. Fügen Sie diesen Prozesswert als Meldetext eines Anwendertextblocks in eine Meldung ein.
2. Filtern Sie in Runtime die Meldungen nach dem Text im Meldeblock "Meldetext".

Hinweis

Achten Sie in Mehrplatzsystemen darauf, dass die bei einem Client die im Dialog "Selektion festlegen" zur Anzeige kommenden Inhalte auf allen Servern gleich benannt werden.

Wenn Sie die Zeitbasis der Meldeanzeige ändern, werden beim Filtern nach der Uhrzeit Startwert und Stoppwert nicht automatisch angepasst.

Beispiel: Am Standort des Rechners mit der Zeitzone "UTC + 1h" haben Sie in der Meldeanzeige "Lokale Zeitzone" gewählt. Sie filtern nach der Uhrzeit 10:00 Uhr bis 11:00 Uhr. Ändern Sie die Zeitbasis von "Lokale Zeitzone" auf "UTC". Wenn Sie die gleichen Meldungen anzeigen wollen, ändern Sie den Filter auf 9:00 Uhr bis 10:00 Uhr.

Siehe auch

Meldeanzeige (Seite 435)

3.5.5 Meldungen in Runtime sortieren

Einleitung

Die Meldungen in der Meldeanzeige können Sie in Runtime nach Spaltenüberschriften sortieren.

Beispiele für das Sortieren von Meldungen:

- Absteigend nach Datum, Uhrzeit, Meldungsnummer. Die neueste Meldung erscheint oben.
- Nach ihrer Priorität
  Dazu müssen die Prioritäten der Meldungen im Editor "HMI-Meldungen" festgelegt und der Meldeblock "Priorität" in der Meldeanzeige projektiert sein. So erscheint bei einer einzeiligen Meldeanzeige nur die Meldung mit hoher Priorität im Meldefenster. Eine Meldung mit niedrigerer Priorität wird nicht dargestellt, auch wenn sie ein neueres Datum trägt. Die Meldungen werden zeitfolgerichtig angezeigt.
- Der Meldeblock "Meldezustand" wird nach der Art des Zustands sortiert und nicht nach den projektierten Zustandstexten. Bei einer aufsteigenden Sortierung wird in folgender Reihenfolge sortiert:
  - Gekommen
  - Gegangen
  - Quittiert
  - Gesperrt
  - Freigegeben
  - Automatische Quittierung
  - Not-Quittierung
  - Gekommen/Gegangen

  Hinweis

  Sortierkriterien im ES für die Meldeanzeige festlegen

  Sie legen bei Bedarf bei der Projektierung der Meldeanzeige Sortierkriterien fest. Klicken Sie dazu im Inspektorfenster der Meldeanzeige "Eigenschaften > Eigenschaften > Anzeige > Spalten".

Meldungen sortieren

Beim Sortieren der Meldeanzeige nach Spalten legen Sie die Sortierung über maximal vier Spalten fest. Ein Pfeil und eine Zahl werden rechts in der Spaltenüberschrift angezeigt. Der Pfeil zeigt die auf- oder absteigende Sortierung an. Die Zahl neben dem Pfeil zeigt die Sortierung nach den Spaltenüberschriften an.

Voraussetzung

- In der Meldeanzeige ist "Sortieren nach Mausklick" oder "Sortieren nach Mausdoppelklick" gewählt.
  Klicken Sie dazu "Eigenschaften > Eigenschaften > Tabelle > Tabelle - Sortierung > Einstellungen".
- In der Meldeanzeige ist die Schaltfläche "Autoscroll" aktiviert.
  Klicken Sie dazu "Eigenschaften > Eigenschaften > Symbolleiste > Symbolleiste - Schaltflächen".

Vorgehen

Um Meldungen in der Meldeanzeige spaltenweise zu sortieren, gehen Sie folgendermaßen vor:

1. Klicken oder doppelklicken Sie in die Spaltenüberschrift, nach der Sie die Meldungen als Erstes sortieren wollen.
   Die Zahl "1" wird angezeigt mit dem Pfeil nach oben für die aufsteigende Sortierung.
   - Wenn in der Meldeanzeige die Sortierreihenfolge "Oben und unten" aktiviert ist, wechselt die Anzeige bei jedem Klick oder Doppelklick auf die Spaltenüberschrift von aufsteigend zu absteigend.
   - Wenn die Sortierreihenfolge "Oben, unten, keine" aktiviert ist, wird beim 3. Klick die Sortierung aufgehoben.
2. Wenn Sie nach mehreren Spalten sortieren wollen, klicken Sie in der gewünschten Reihenfolge in die jeweilige Spaltenüberschrift.

Hinweis

Die Sortierung reagiert nicht auf Ereignisse oder eine Änderung der Projektierung der Meldeblöcke.

Ein Meldeblock entspricht einem Feld einer Meldung. Wenn Sie einen Meldeblock im Sortierdialog als Sortierkriterium festlegen und den Meldeblock danach unter "Eigenschaften > Eigenschaften > Spalten" aus der Meldezeile entfernen, bleibt die Sortierung in der festgelegten Reihenfolge bestehen. Im Sortierdialog wird anstelle des entfernten Meldeblocks ein leeres Feld angezeigt.

Wenn Sie keine andere Sortierung festlegen und den entfernten Meldeblock in der Projektierung der Meldeblöcke wieder aufnehmen, werden die Meldungen wieder in der ursprünglichen Art sortiert.

3.5.6 Meldeanzeige in Runtime umschalten

Einleitung

In der Meldeanzeige filtern Sie die anzuzeigenden Meldungen über vorkonfigurierte Schaltflächen.

Voraussetzungen

- In der Meldeanzeige sind die entsprechenden sechs Schaltflächen zur Anzeige einer bestimmten Auswahl von Meldungen projektiert.

Vorgehen

Um in der Meldeanzeige eine bestimmte Auswahl von Meldungen anzeigen zu lassen, gehen Sie folgendermaßen vor:

- en Sie auf eine der folgenden Schaltflächen der Meldeanzeige:
  Meldeliste, um die aktuell anstehenden Meldungen anzuzeigen
  Kurzzeitarchivliste, um archivierte Meldungen anzuzeigen
  eu eingehenden Meldungen wird die Anzeige sofort aktualisiert.
  Langzeitarchivliste, um archivierte Meldungen anzuzeigen
  Sperrliste, um die aktuell gesperrten Meldungen anzuzeigen
  Meldestatistik, um statistische Informationen anzuzeigen
  Liste der ausgeblendeten Meldungen, um die Meldungen anzuzeigen, die archiviert werden, aber nicht sichtbar sind

3.5.7 Archivierte Meldungen in Runtime anzeigen

Einleitung

Die Meldeanzeige zeigt in Runtime nicht nur aktuelle Meldungen an, sondern auch Meldungen aus dem Archiv.

Voraussetzungen

- Alle archivierten Daten, die in Runtime angezeigt werden sollen, müssen lokal auf dem Archivserver liegen. Der SQL-Server erlaubt keinen Zugriff auf Backup-Dateien, die z. B. auf einem anderen PC im Netzwerk liegen.
- Die Schaltflächen "Kurzzeitarchivliste" und "Langzeitarchivliste" sind in der Meldeanzeige projektiert.

Vorgehen

Um archivierte Meldungen in Runtime anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Meldeanzeige auf die Schaltflächen "Kurzzeitarchivliste", um archivierte und aktuelle Meldungen anzeigen zu lassen. In der Kurzzeitarchivliste werden die neu eingehenden Meldungen sofort aktualisiert.

- ODER -

1. Klicken Sie in der Meldeanzeige auf die Schaltflächen "Langzeitarchivliste", um nur archivierte Meldungen anzeigen zu lassen. Die Langzeitarchivliste zeigt nur archivierte Meldungen an. In der Langzeitarchivliste kann ein Kommentar eingegeben oder angezeigt werden.

3.5.8 Meldungen in Runtime drucken

Einleitung

Entsprechend Ihren Einstellungen im Editor "HMI-Meldungen" und im Editor "Protokolle" hat der Benutzer folgende Möglichkeiten, Meldungen in Runtime zu drucken:

Meldefolgeprotokoll drucken

Wenn das Meldefolgeprotokoll aktiviert ist, wird in Runtime jede Zustandsänderung einer Meldung laufend auf einem Drucker ausgegeben. In Runtime sind dazu keine Bedienungen notwendig.

Protokoll ausdrucken

Die Bedienung in Runtime ist abhängig von Ihrer Projektierung. Sie haben folgende Möglichkeiten, ein Meldeprotokoll in Runtime auszudrucken:

- Zeitgesteuerte Ausgabe
  - Einmalige, zeitgesteuerte Ausgabe, z. B. bei Schichtbeginn
  - In Intervallen sich wiederholende Ausgabe, z. B. alle zwei Stunden
- Ereignisgesteuerte Ausgabe
  - Wertänderung einer Variable
  - Klicken auf eine projektierte Schaltfläche in einem Bild
  - Klicken auf die Schaltfläche der Meldeanzeige
  - Überlauf eines Archivs
  - Skript

Hinweis

Beachten Sie, dass beim direkten Druck der Meldungen in Runtime nur ASCII-Zeichen unterstützt werden.

Voraussetzung

- Mehrere Meldungen werden auf dem Bediengerät angezeigt.

Vorgehen

Um die aktuelle Ansicht des Meldefensters in Runtime zu drucken, gehen Sie folgendermaßen vor:

1. Filtern Sie die Meldeanzeige nach Bedarf mithilfe der Bedienelemente der Meldeanzeige.
2. Klicken Sie auf die Schaltfläche der Meldeanzeige.

Ergebnis

Auf dem konfigurierten Drucker werden entsprechend Ihren Einstellungen im Editor "Protokolle" die im Meldefenster angezeigten Meldungen ausgegeben.

Drucker einstellen

Den Drucker konfigurieren Sie im Control Panel des jeweiligen Bediengeräts. Die Einstellungen sind abhängig vom Betriebssystem des Bediengeräts. Nähere Hinweise hierzu finden Sie in der Betriebsanleitung Ihres Bediengeräts und im Kapitel "Protokollieren von Meldungen (Seite 909)".

Siehe auch

Arbeiten mit Protokollen (Seite 1287)

3.5.9 Sperren von Meldungen in Runtime

3.5.9.1 Einzelne Meldungen sperren

Einleitung

Wenn Sie eine Meldung sperren, wird diese Meldung nicht angezeigt und nicht archiviert. Bei Bedarf können Sie einzelne Meldungen, Meldeklassen oder Meldegruppen sperren.

Hinweis

Gesperrte Meldung:

Gesperrte Meldungen sind nach einem Neustart der WinCC Runtime nicht mehr gesperrt. Weiterhin gesperrt bleiben nur Meldungen, die über Datenbausteine direkt in der AS gesperrt werden (Sperre über Quelle).

Gesperrte Meldeklassen / Meldegruppen:

Die Sperre von Meldeklassen und Meldegruppen bleibt auch nach einem Neustart der WinCC Runtime erhalten.

Voraussetzung

- Die Meldeanzeige ist projektiert.
- Die Meldeanzeige wird in Runtime angezeigt.

- Die Schaltflächen "Meldung sperren/Meldung entsperren" / und "Gesperrte Meldungen" sind projektiert.
- Der Benutzer besitzt die Berechtigung zum Sperren und Entsperren von Meldungen.

Hinweis

Die Berechtigungen "Meldungen sperren" und "Meldungen freigeben" müssen direkt untereinander projektiert sein. Dies ist erforderlich, weil für die Berechtigung "Meldungen freigeben" automatisch die Berechtigungsstufe verwendet wird, die innerhalb der Berechtigungen direkt unter der Berechtigung "Meldungen sperren" steht.

Vorgehen

Um die Anzeige von Meldungen über die Schaltfläche zu sperren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Meldeanzeige die Meldung, die Sie sperren wollen.
2. Klicken Sie auf die Schaltfläche "Meldung sperren".
   Die Meldung wird aus der Meldeliste entfernt.
3. Um alle gesperrten Meldungen anzuzeigen, klicken Sie auf die Schaltfläche "Gesperrte Meldungen".
   Die Sperrliste wird angezeigt.
4. Wählen Sie die Meldung, die Sie entsperren wollen.
5. Klicken Sie auf die Schaltfläche "Meldung entsperren".
   Die Meldung wird wieder angezeigt und aus der Sperrliste entfernt.

Hinweis

Das Sperren oder Entsperren einer Meldung eines Alarm_8(P) über S7PMC führt immer dazu, dass alle 8 Meldungen dieses Bausteins gesperrt oder entsperrt werden.

Siehe auch

Meldeanzeige (Seite 435)

3.5.9.2 Meldungen über die Meldenummer sperren

Voraussetzungen

- Die Schaltfläche "Sperrdialog" ist in der Meldeanzeige projektiert.

Vorgehen

Um die Anzeige von Meldungen über die Meldenummer zu sperren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche .
   Der Dialog "Sperrliste parametrieren" wird geöffnet.

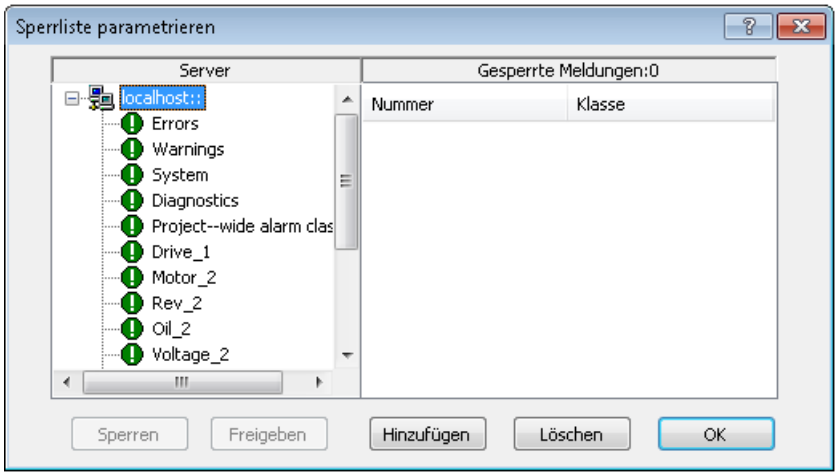

2. Klicken Sie in der Serverliste auf einen Server oder bei Einplatzprojekten auf den lokalen PC.
3. Klicken Sie auf die Schaltfläche "Hinzufügen" und geben Sie im folgenden Dialog die Meldenummer ein, die Sie sperren wollen.
4. Sie sperren bei Bedarf mehrere Meldungen gleichzeitig. Geben Sie die Meldenummer dann mit einem Komma getrennt ein. Für einen Nummernbereich geben Sie den Bereich in der Form "5-10" ein. Nur durchgehende Nummernbereiche werden gesperrt. Wenn sich Lücken in dem angegebenen Bereich befinden, wird ein Hinweis "ungültiger Bereich" ausgegeben.
5. Sie geben eine gesperrte Meldung frei, indem Sie in der Liste der gesperrten Meldungen die Meldung auswählen und auf die Schaltfläche "Löschen" klicken.

Hinweis

Sie können nicht mehr als 50 Meldungen gleichzeitig sperren und freigeben.

3.5.9.3 Meldungen über die Meldeklasse oder Meldegruppe sperren

Einleitung

Sie können die Anzeige von Meldungen auch sperren, indem Sie die Meldeklasse oder Meldegruppe, der die Meldung angehört, insgesamt sperren. Keine der zugehörigen Meldungen dieser gesperrten Meldegruppen und Meldeklassen wird angezeigt.

Voraussetzungen

- Die Schaltfläche "Sperrdialog" ist in der Meldeanzeige projektiert.

Vorgehen

Um Meldungen über die Meldeklasse oder die Meldegruppe zu sperren und entsperren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Meldeanzeige auf die Schaltfläche .
   Der Dialog "Sperrliste parametrieren" wird geöffnet.

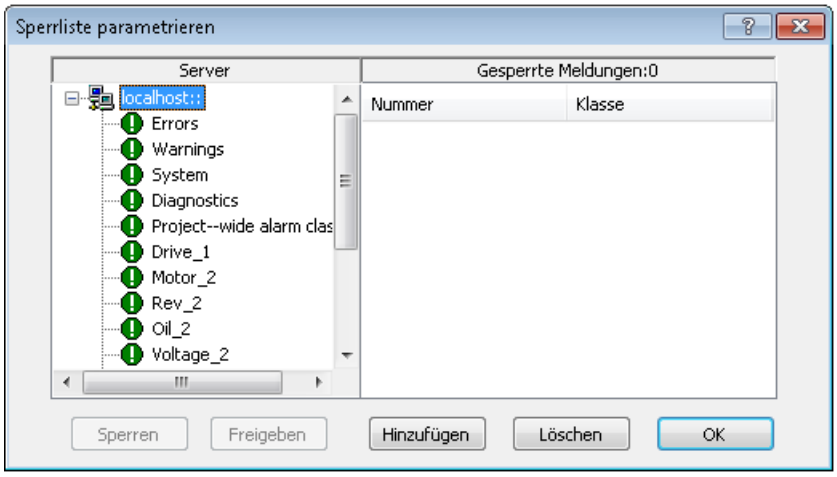

2. Klicken Sie in der Serverliste auf einen Server oder bei Einplatzprojekten auf den lokalen PC.
3. Wählen Sie in der Baumansicht eine Meldeklasse oder eine benutzerdefinierte Meldegruppe aus, die Sie sperren wollen.
4. Klicken Sie auf die Schaltfläche "Sperren".
   Alle Meldungen dieser Meldeklasse oder benutzerdefinierten Meldegruppe werden gesperrt.

Sie geben die gesperrten Meldungen frei, indem Sie die Meldeklasse oder benutzerdefinierte Meldegruppe auswählen und auf die Schaltfläche "Freigeben" klicken.

Bedienmeldung beim Sperren einer Meldung

Sie legen in der Meldeanzeige bei Bedarf fest, ob beim Sperren oder Freigeben einer Meldung eine Bedienmeldung erzeugt wird. Die Bedienmeldung konfigurieren Sie bei Bedarf selbst. Die Bedienmeldung enthält standardmäßig folgende Objekte:

- Einen Zeitstempel
- Den angemeldeten Benutzer

Der Zeitstempel für die Meldung kommt in folgenden Fällen:

- Beim aktiven Sperren von der Meldungsquelle (z. B. AS).

Aktiv gesperrte Meldungen sind auf allen Bediengeräten sichtbar und bedienbar.

3.5.10 Quittieren von Meldungen

3.5.10.1 Meldungen quittieren

Einleitung

Meldungen quittieren Sie in Runtime abhängig von der Projektierung Ihres Projekts. Sie quittieren Meldungen folgendermaßen:

- Mit den Schaltflächen der Anzeige- und Bedienobjekte
- Mit individuell projektierten Schaltflächen

Wenn eine Bedienberechtigung für die Bedienelemente projektiert ist, dann sind die Meldungen nur von berechtigten Benutzern quittierbar.

Um das Quittieren in Runtime zu umgehen, stehen Ihnen in WinCC Systemfunktionen und Skripte zur Verfügung, über die Sie Meldungen automatisch quittieren.

Varianten der Quittierung

Sie quittieren in Runtime einzelne Meldungen oder mehrere Meldungen gemeinsam. Sie unterscheiden dabei zwischen folgenden Möglichkeiten:

- Einzelquittierung
  Quittieren einer Meldung über eine Schaltfläche oder Funktionstaste.
- Meldegruppen quittieren
  Quittierung aller Meldungen einer Meldegruppe über eine Schaltfläche oder Funktionstaste
- Sammelquittierung
  Quittierung aller anstehenden und quittierpflichtigen Meldungen in der Meldeanzeige über die Schaltfläche "Sammelquittierung" der Meldeanzeige.
- Zweifachquittierung
  Wenn eine Meldung zweifach zu quittieren ist, quittieren Sie das Kommen der Meldung und das Gehen.

Voraussetzung

- Eine Meldung wird am Bediengerät angezeigt.
- Die Schaltflächen “Sammelquittierung“ und “Einzelquittierung“ sind in der Meldeanzeige projektiert.

Vorgehen

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

- Markieren Sie die Meldung.
- Klicken Sie auf die Schaltfläche "Einzelquittierung" der Meldeanzeige.

Ergebnis

Die Meldung erhält den Zustand "Quittiert". Wenn die Bedingung zum Auslösen einer Meldung nicht mehr zutrifft, erhält die Meldung zusätzlich den Zustand "Gegangen" und wird am Bediengerät nicht mehr angezeigt.

Siehe auch

Meldeanzeige (Seite 435)

3.5.10.2 Meldungen sammelquittieren

Einleitung

Als Sammelquittierung wird die Quittierung aller anstehenden, sichtbaren und quittierpflichtigen Meldungen im Meldefenster bezeichnet. Sofern für eine Meldung nicht die Option "Einzelquittierung" aktiviert ist, kann sie sammelquittiert werden.

Voraussetzung

Mehrere quittierpflichtige Meldungen stehen in der Meldeanzeige an.

Vorgehen

Um Meldungen gesammelt zu quittieren, gehen Sie folgendermaßen vor:

1. Lesen Sie die Meldetexte der anstehenden Meldungen und veranlassen Sie bei Bedarf Abhilfemaßnahmen.
2. Klicken Sie in der Meldeanzeige auf die Schaltfläche .

Ergebnis

Alle anstehenden Meldungen mit folgenden Eigenschaften wurden quittiert:

- Quittierpflichtig
- Nicht Einzelquittierpflichtig
- Sichtbar

3.5.10.3 Meldungen not-quittieren

Einleitung

In Ausnahmefällen quittiert der Bediener eine Meldung anhand ihrer Meldenummer. Dabei wird das Quittierbit auch dann an das AS gesandt, wenn die Meldung aktuell nicht ansteht. Diese Funktion ist nur für den äußersten Notfall vorgesehen.

Voraussetzung

Die Schaltfläche ist in der Meldeanzeige projektiert.

Vorgehen

Um eine Meldung in Runtime über einer Not-Quittierung zu quittieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche der Meldeanzeige.
   Der Dialog "Not-Quittieren von Meldungen" wird geöffnet.

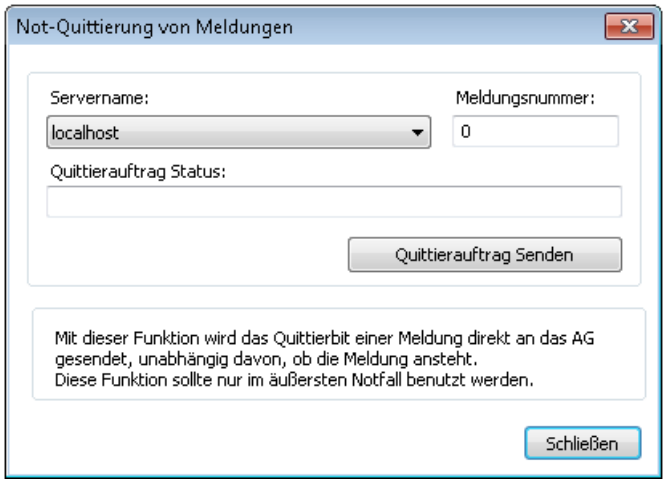

2. Wählen Sie einen Server aus.
3. Geben Sie unter "Meldungsnummer" die Meldenummer ein.
4. Klicken Sie auf die Schaltfläche "Quittierauftrag senden".
   In der Meldeanzeige wird die Meldung nun in der Farbgebung angezeigt, die Sie für eine quittierte Meldung festgelegt haben.

Hinweis

Im Dialog zur Auswahl der Server werden Ihnen nur die Server-Projekte angeboten, deren "Packages" auf den PC geladen wurden.

3.5.10.4 Zentralen Melder quittieren

Einleitung

Wie Sie in Runtime einen zentralen Melder quittieren, ist abhängig davon, wie Sie dessen Quittierung projektiert haben.

Voraussetzung

- Der zentrale Melder und seine Quittierung sind projektiert.
- Der zentrale Melder ist getriggert.
- Die Schaltflächen "Quittierung zentraler Melder" und " Einzelquittierung" sind in der Meldeanzeige projektiert.

Quittieren über einen separaten Schlüssel

- Um den zentralen Melder zu quittieren, klicken Sie in der Meldeanzeige auf die Schaltfläche "Quittierung zentraler Melder".
  Sie quittieren nur den zentralen Melder. Die Meldung, die den zentralen Melder getriggert hat, ist noch im Zustand "Gekommen".

Einzelquittierung

- Um den zentralen Melder zu quittieren, wählen Sie die aktuelle Meldung aus und klicken Sie auf die Schaltfläche "Einzelquittierung" der Meldeanzeige.
  Damit quittieren Sie den zentralen Melder und die Meldung, die den zentralen Melder getriggert hat.

Quittiervariable

Wenn der zentrale Melder über eine Variable quittiert wird, dann quittieren Sie den zentralen Melder z. B. mit einer individuell konfigurierten Schaltfläche oder Funktionstaste.

Ergebnis

Das Quittierbit des zentralen Melders ist gesetzt.

Siehe auch

Meldeanzeige (Seite 435)

3.6 Security Ereignisse anzeigen

3.6.1 Security Ereignisse am Bediengerät anzeigen

Einleitung

Zu den bestehenden Meldungen von WinCC können Sie auch Security Ereignisse am Bediengerät anzeigen.

Security Ereignisse sind sicherheitsrelevante Ereignisse, wie z. B. ein Angriff über das Netzwerk auf ein Gerät oder Änderung der Schutzstufe für die Kommunikation zwischen der Steuerung und dem Bediengerät.

Security Ereignisse werden von der Steuerung erkannt und an Bediengeräte weitergeleitet. Auf den Bediengeräten werden die Security Ereignisse im Meldearchiv angezeigt.

Eine Projektierung oder Aktivierung der Funktionalität Security Ereignisse ist innerhalb der Steuerung nicht notwendig. Security Ereignisse werden von der Steuerung automatisch erkannt.

Einschränkungen

Von folgenden Steuerungen können Sie die Security Ereignisse am Bediengerät anzeigen:

- SIMATIC S7-1200
- SIMATIC S7-1500

Hinweis

Geräteabhängigkeit

Die Security Ereignisse von Steuerungen werden auf Basic Panels und Basic Panels 2nd Generation nicht angezeigt.

Anzeige von Security Ereignissen projektieren

Für die Anzeige von Security Ereignissen am Bediengerät sind folgende Schritte notwendig:

- Aktivierung der Meldungen von Steuerungen
- Meldearchiv für Meldungen von Steuerungen anlegen

Nähere Informationen zur Projektierung finden SIe hier: Anzeige von Security Ereignissen projektieren (Seite 986)

Siehe auch

Anzeige von Security Ereignissen projektieren (Seite 986)

3.6.2 Anzeige von Security Ereignissen projektieren

Voraussetzung

Zwischen Bediengerät und Steuerung besteht eine HMI-Verbindung.

Vorgehensweise

1. Öffnen Sie die "Runtime-Einstellungen" des Bediengeräts.
2. Aktivieren Sie unter "Meldungen" die Funktion "Security Ereignisse".

   Hinweis

   Die Funktion "Security Ereignisse" ist standardmäßig deaktiviert und muss für jede HMI-Verbindung aktiviert werden.

3. Legen Sie im Editor "Archive" unter "Meldearchive" ein neues Meldearchiv an.
4. Öffnen Sie ein HMI-Bild.
5. Legen Sie eine Meldeanzeige an.

Ergebnis

Die Security Ereignisse werden im Meldearchiv angezeigt.

3.7 Vollständige Meldung von der Steuerung an das Bediengerät senden

3.7.1 Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren

Grundlagen

Zusätzlich zu Meldungen in WinCC können Sie in STEP 7 Steuerungsmeldungen projektieren und auf Ihrem Bediengerät anzeigen.

Wenn Steuerungsmeldungen in STEP 7 projektiert sind, eine Verbindung zu einer S7-1500 Steuerung hergestellt ist und aktuell Steuerungsmeldungen anstehen, werden die Steuerungsmeldungen nach Meldungsänderungen (z. B. Änderung des Meldetextes) automatisch an die Bediengeräte gesendet und aktualisiert. Auf diese Weise sparen Sie Zeit, da Sie die Projektierungsänderungen der Meldungen nicht extra auf das Bediengerät laden müssen. Das Bediengerät muss bei Änderungen der Meldungen auch nicht den Runtime-Betrieb verlassen.

Folgende Steuerungsmeldungen lassen sich an das Bediengerät senden:

- Systemdiagnose
- ProDiag-Meldungen
- Programmmeldungen

Die Steuerungsmeldungen können vollständig an die Bediengeräte mit der Geräteversion V14 gesendet werden, wenn entsprechende Einstellungen in der Steuerung und auf dem Bediengerät konfiguriert sind. Auf dem Bediengerät muss die Option "Automatische Aktualisierung" unter "Runtime-Einstellungen Meldungen > Steuerungsmeldungen" für die jeweilige Verbindung aktiviert sein. Weitere Informationen zu den Einstellungen finden Sie unter Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren (Seite 991).

Geräteabhängigkeit

Die Steuerungsmeldungen von folgender Steuerung werden beim Auftreten automatisch und vollständig an das Bediengerät gesendet:

- SIMATIC S7-1500 (ab der Firmwareversion 2.0)

Basic Panels und Basic Panels 2nd Generation unterstützen keine Steuerungsmeldungen. Für Basic Panels und Basic Panels 2nd Generation sowie Bediengeräte mit einer Geräteversion kleiner als V14 müssen die Projektierungsänderungen der Meldungen auf die Steuerung und auf das Bediengerät geladen werden.

Wenn im Bediengerät "Automatische Aktualisierung" deaktiviert ist, müssen die Projektierungsänderungen der Meldungen auf die Steuerung und auf das Bediengerät geladen werden.

Hinweis

Beim Hochrüsten eines Projekts auf die TIA Portal-Version V14 ist die Übertragung der Steuerungsmeldungen an das Bediengerät unter "Runtime-Einstellungen" des Bediengeräts deaktiviert, weil die Verbindung zwischen dem Bediengerät und der jeweiligen Steuerung in einer älteren TIA Portal-Version erstellt wurde.

Spracheinstellungen

Damit die Meldungen am Bediengerät in der richtigen Sprache angezeigt werden, müssen dieselben drei Sprachen (oder weniger) für die Meldungen in der Steuerung und auf dem Bediengerät konfiguriert sein. Eventuell sind Absprachen mit dem Steuerungsprojekteur bezüglich der Sprachauswahl erforderlich.

Wenn auf dem Bediengerät und in der Steuerung unterschiedliche Sprachen projektiert sind, zeigt das Bediengerät im Betrieb statt den Steuerungsmeldungen den Text "###Text missing###"an.

Hinweis

Wenn Sie am Bediengerät mehr als drei Sprachen gleichzeitig verwenden möchten, müssen Sie die Option für die automatische Aktualisierung der Steuerungsmeldungen deaktivieren. Anderenfalls wird bei der vierten und jeder weiteren Sprache statt den Steuerungsmeldungen "###Text missing###" angezeigt.

Besonderheiten der asiatischen Schriftarten

Bei Verwendung von asiatischen Sprachen werden nur die asiatischen Zeichen, die in der Projektierung des Bediengerätes verwendet werden, beim Transfer auf das Bediengerät geladen. Wenn die Texte direkt von der Steuerung an das Bediengerät gesendet werden, kann dies zu Darstellungsproblemen führen. In diesen Fällen können Sie die komplette asiatische Schriftart auf das Bediengerät laden. Weitere Informationen finden Sie unter Asiatische Schriftarten auf das Bediengerät laden (Seite 2441).

Hinweise und Regeln

- Geänderte Texte, die Sie in die Steuerung laden, werden sofort am Bediengerät angezeigt, wenn die Steuerungsmeldung bereits getriggert ist.
- Für Meldungen, die im Meldearchiv angezeigt werden, gilt Folgendes:
  - Das Meldearchiv enthält Meldungen in der eingestellten Sprache. Diese Sprache wird bei der Projektierung des Bediengeräts unter "Runtime-Einstellungen > Allgemein > Archive > Archivierungssprache" eingestellt. Unabhängig davon, ob Sie eine bestimmte Sprache für die Archivierung der Steuerungsmeldungen im Meldearchiv festlegen oder die Anlaufsprache wählen, muss die jeweilige Sprache auch in der Steuerung aktiviert sein. Wenn Sie die Anlaufsprache festgelegen, kann es in Verbindung mit der Option "Automatische Aktualisierung" zu vermischten Sprachen im Meldearchiv führen. Weitere Informationen zur Sprachumschaltung des Archivs in Runtime finden Sie unter Verhalten des Archivs bei Sprachumschaltung in Runtime steuern (Seite 933).
  - Für das jeweilige Meldearchiv muss unter "Eigenschaften > Eigenschaften > Archivierungsmethode" das Optionskästchen "Meldetext und Fehlerstelle archivieren" aktiviert sein, damit die achivierten Steuerungsmeldungen in der Meldeanzeige bzw. dem Meldefenster angezeigt werden können.

Siehe auch

Asiatische Schriftarten auf das Bediengerät laden (Seite 2441)

Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren (Seite 991)

Verhalten des Archivs bei Sprachumschaltung in Runtime steuern (Seite 933)

3.7.2 Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren

Grundlagen

Zusätzlich zu Meldungen in WinCC können Sie in STEP 7 Steuerungsmeldungen projektieren und auf Ihrem Bediengerät anzeigen.

Wenn Steuerungsmeldungen in STEP 7 projektiert sind, eine Verbindung zu einer S7-1500 Steuerung hergestellt ist und aktuell Steuerungsmeldungen anstehen, werden die Steuerungsmeldungen nach Meldungsänderungen (z. B. Änderung des Meldetextes) automatisch an die Bediengeräte gesendet und aktualisiert. Auf diese Weise sparen Sie Zeit, da Sie die Projektierungsänderungen der Meldungen nicht extra auf das Bediengerät laden müssen. Das Bediengerät muss bei Änderungen der Meldungen auch nicht den Runtime-Betrieb verlassen.

Folgende Steuerungsmeldungen lassen sich an das Bediengerät senden:

- Systemdiagnose
- Prozessdiagnose
- Programmmeldungen

Die Steuerungsmeldungen können vollständig an die Bediengeräte mit der Geräteversion V14 gesendet werden, wenn entsprechende Einstellungen in der Steuerung und auf dem Bediengerät konfiguriert sind. Auf dem Bediengerät muss die Option "Automatische Aktualisierung" unter "Runtime-Einstellungen Meldungen > Steuerungsmeldungen" für die jeweilige Verbindung aktiviert sein. Weitere Informationen zu den Einstellungen finden Sie unter Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren (Seite 991).

Geräteabhängigkeit

Die Steuerungsmeldungen von folgender Steuerung werden beim Auftreten automatisch und vollständig an das Bediengerät gesendet:

- SIMATIC S7-1500 (ab der Firmwareversion 2.0)

Basic Panels und Basic Panels 2nd Generation unterstützen keine Steuerungsmeldungen. Für Basic Panels und Basic Panels 2nd Generation sowie Bediengeräte mit einer Geräteversion kleiner als V14 müssen die Projektierungsänderungen der Meldungen auf die Steuerung und auf das Bediengerät geladen werden.

Wenn im Bediengerät "Automatische Aktualisierung" deaktiviert ist, müssen die Projektierungsänderungen der Meldungen auf die Steuerung und auf das Bediengerät geladen werden.

Hinweis

Beim Hochrüsten eines Projekts auf die TIA Portal-Version V14 ist die Übertragung der Steuerungsmeldungen an das Bediengerät unter "Runtime-Einstellungen" des Bediengeräts deaktiviert, weil die Verbindung zwischen dem Bediengerät und der jeweiligen Steuerung in einer älteren TIA Portal-Version erstellt wurde.

Spracheinstellungen

Damit die Meldungen am Bediengerät in der richtigen Sprache angezeigt werden, müssen dieselben drei Sprachen (oder weniger) für die Meldungen in der Steuerung und auf dem Bediengerät konfiguriert sein. Eventuell sind Absprachen mit dem Steuerungsprojekteur bezüglich der Sprachauswahl erforderlich.

Wenn auf dem Bediengerät und in der Steuerung unterschiedliche Sprachen projektiert sind, zeigt das Bediengerät im Betrieb statt den Steuerungsmeldungen den Text "###Text missing###"an.

Hinweis

Wenn Sie am Bediengerät mehr als drei Sprachen gleichzeitig verwenden möchten, müssen Sie die Option für die automatische Aktualisierung der Steuerungsmeldungen deaktivieren. Anderenfalls wird bei der vierten und jeder weiteren Sprache statt den Steuerungsmeldungen "###Text missing###" angezeigt.

Siehe auch

Steuerungsmeldungen bearbeiten (Seite 860)

Steuerungsmeldungen (Seite 794)

Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren (Seite 991)

3.7.3 Automatische Aktualisierung von Steuerungsmeldungen am Bediengerät projektieren

Einleitung

Die Option "Automatische Aktualisierung" ist für alle Verbindungen zwischen den S7-1500-Steuerungen (Firmwareversion 2.0) und den Bediengeräten mit der Geräteversion V14 standardmäßig aktiviert.

Wenn die Verbindung zwischen einem Bediengerät und einer Steuerung S7-1500 in einer älteren TIA Portal-Version als V14 erstellt wurde, ist die Option "Automatische Aktualisierung" auch nach dem Hochrüsten des Projekts auf die Version V14 deaktiviert. Für diese Verbindung können Sie die automatische Aktualisierung der Steuerungsmeldungen nachträglich projektieren.

Voraussetzung

- Zwischen Bediengerät und einer S7-1500 Steuerung (ab Firmwareversion 2.0) oder einem kompatiblem Gerät (z. B. WinAC 1500) besteht eine HMI-Verbindung.
- Steuerung und Bediengerät haben mindestens Version 14 oder höher.
- In der Steuerung ist "Zentrale Meldungsverwaltung" aktiviert (die automatische Aktualisierung wurde steuerungseitig aktiviert).
- In STEP 7 wurden Steuerungsmeldungen projektiert.
- Am Bediengerät wurde eine Meldeanzeige oder ein Meldefenster projektiert.
- Die zu den Steuerungsmeldungen zugehörigen Meldeklassen sind in der Meldeanzeige bzw. im Meldefenster ausgewählt.
- Für Meldungen sind in der Steuerung und auf dem Bediengerät dieselben drei Sprachen (oder weniger) konfiguriert.

Vorgehensweise

1. Öffnen Sie die "Runtime-Einstellungen" des Bediengeräts.
   Unter "Meldungen > Steuerungsmeldungen" werden eine oder mehrere Verbindungen zu Steuerungen angezeigt.
2. Aktivieren Sie die Option "Automatische Aktualisierung" für die jeweilige Verbindung, für die Sie die Steuerungsmeldungen anzeigen lassen möchten.
   Die Funktion "Automatische Aktualisierung" muss für jede Verbindung separat aktiviert werden.

Hinweis

Wenn für die Steuerung andere Sprachen konfiguriert sind als für das Bediengerät, können die Meldungen in Runtime nicht angezeigt werden. Stattdessen wird die Meldung "##Text missing##" angezeigt.

Hinweis

Wenn Sie am Bediengerät mehr als drei Sprachen gleichzeitig verwenden möchten, müssen Sie die Option "Automatische Aktualisierung" deaktivieren. Anderenfalls wird bei der vierten und jeder weiteren Sprache statt den Steuerungsmeldungen die Meldung "##Text missing##" angezeigt.

Ergebnis

In Runtime werden die Steuerungsmeldungen in der Meldeanzeige angezeigt.

Siehe auch

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 989)

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 987)

3.8 Referenz

3.8.1 Systemfunktionen für Meldungen

Systemfunktionen

Systemfunktionen sind vordefinierte Funktionen, mit denen Sie auch ohne Programmierkenntnisse viele Aufgaben in Runtime realisieren. Systemfunktionen verwenden Sie in einer Funktionsliste oder in einem Skript.

Hinweis

Geräteabhängigkeit

Skripte stehen nicht für alle Bediengeräte zur Verfügung.

Die Tabelle zeigt alle Systemfunktionen zum Darstellen und Bearbeiten von Meldungen.

Systemfunktion	Auswirkung
BearbeiteMeldung	Löst das Ereignis "Loop-In-Alarm" für die gewählte Meldung oder bei Mehrfachselektion für die zuletzt gewählte Meldung aus. Wenn die zu bearbeitende Meldung noch nicht quittiert wurde, erfolgt auch die Quittierung der gewählten Meldung.
LoescheMeldepuffer	Löscht Meldungen aus dem Meldepuffer am Bediengerät.
LoescheMeldepufferProTool	Funktion wie "LoescheMeldepuffer". Diese Systemfunktion ist aus Kompatibilitätsgründen enthalten und verwendet die ProTool-Nummerierung.
MeldeanzeigeAktualisieren	Aktualisiert die Meldeanzeige.
MeldeanzeigeLoopInAlarm	Löst das Ereignis "Loop-In-Alarm" für die gewählte Meldung oder bei Mehrfachselektion für die zuletzt gewählte Meldung aus. Wenn die zu bearbeitende Meldung noch nicht quittiert wurde, erfolgt auch die Quittierung der gewählten Meldung.
MeldeanzeigeQuittiereMeldung	Quittiert die Meldungen, die in der angegebenen Meldeanzeige markiert sind.
MeldeanzeigeZeigeHilfetext	Zeigt den projektierten Tooltipp der Meldung an, welcher in der angegebenen Meldeanzeige markiert ist.
QuittiereMeldung	Quittiert alle markierten Meldungen.
SetzeProtokolliermodusMeldung	Schaltet das automatische Protokollieren von Meldungen auf dem Drucker ein oder aus.
ZeigeMeldefenster	Blendet das Meldefenster am Bediengerät ein oder aus.
ZeigeSystemmeldung	Zeigt den Wert des übergebenen Parameters als Systemmeldung am Bediengerät an.

Hinweis
Geräteabhängigkeit

Folgende Systemfunktionen stehen nicht für alle Bediengeräte zur Verfügung:

- SetzeProtokolliermodusMeldung
- MeldeanzeigeAktualisieren
- ZeigeSystemmeldung

3.8.2 Ereignisse für Meldungen

Ereignisse für Meldungen und deren Anzeige- und Bedienobjekte

In Runtime treten bei Meldungen deren Anzeige- und Bedienobjekte nachfolgende Ereignisse ein. Für jedes Ereignis projektieren Sie bei Bedarf eine Funktionsliste.

Objekt	Projektierbare Ereignisse
Meldungen	Gekommen Gegangen Quittieren Loop-In-Alarm
Meldeanzeige	Aktivieren Deaktivieren Auswahl geändert
Meldeindikator	Klicken Klicken bei Blinken

Hinweis
Geräteabhängigkeit

Folgende Ereignisse der Meldeanzeige stehen nicht für alle Bediengeräte zur Verfügung:

- Aktivieren
- Deaktivieren
- Auswahl geändert
- Klicken
- Klicken bei Blinken

3.8.3 Ereignisse für Meldungen

Ereignisse für Meldungen und deren Anzeige- und Bedienobjekte

In Runtime treten bei Meldungen deren Anzeige- und Bedienobjekte nachfolgende Ereignisse ein. Für jedes Ereignis projektieren Sie bei Bedarf eine Funktionsliste.

Objekt	Projektierbare Ereignisse
Meldungen	Gekommen Gegangen Quittieren Loop-In-Alarm Status geändert
Meldeanzeige	Aktivieren Aufgebaut Schaltfläche Symbolleiste Klicken Eigenschaft geändert Eigenschaft Symbolleiste Schaltfläche ändern Eigenschaft Statuszeilenelement ändern Eigenschaft Block geändert Eigenschaft Spalte geändert Eigenschaft Spalte Meldestatistik geändert Eigenschaft Bedienmeldung geändert

3.8.4 Systemfunktionen für Archive

Systemfunktionen

Für die Archivierung stehen Ihnen folgende Systemfunktionen zur Verfügung:

Name der Funktion	Funktionsweise
ArchiviereProtokolldatei	Diese Funktion verschiebt oder kopiert ein Archiv zur Langzeitarchivierung an einen anderen Speicherort. Wenn Sie z. B. das Audit Trail von einem lokalen Speichermedium auf den Server verschieben wollen, verwenden Sie die Systemfunktion. Verwenden Sie bei Audit Trails immer den Modus "Archiv verschieben (hmiMove)", da sonst durch doppelte Datenhaltung gegen die FDA-Richtlinie verstoßen wird.
ArchiviereVariable	Speichert den Wert der angegebenen Variablen im angegebenen Variablenarchiv. Wenn Sie einen Prozesswert zu einem bestimmten Zeitpunkt archivieren wollen, verwenden Sie die Systemfunktion.
StarteArchivierung	Startet die Archivierung in dem angegebenen Archiv. Sie unterbrechen die Archivierung in Runtime mit der Systemfunktion "StoppeArchivierung".
StoppeArchivierung	Stoppt die Archivierung in dem angegebenen Archiv. Sie setzen die Archivierung in Runtime wieder fort mit der Systemfunktion "StarteArchivierung".
LöscheArchiv	Löscht alle Einträge des angegebenen Archivs.

Name der Funktion	Funktionsweise
StarteNächstesArchiv	Stoppt die Archivierung in dem angegebenen Archiv. Die Archivierung wird in dem Archivsegment fortgesetzt, das Sie für das angegebene Archiv projektiert haben.
SchliesseAlleArchive	Schließt alle Archive. Die Verbindung zwischen WinCC und den Archivdateien oder der Archivdatenbank wird getrennt. Wenn Sie z. B. das Speichermedium des Bediengeräts wechseln wollen, ohne die Runtime-Software zu beenden, verwenden Sie diese Systemfunktion.
ÖffneAlleArchive	Öffnet alle Archive, um die Archivierung fortzusetzen. Die Verbindung zwischen WinCC und den Archivdateien oder der Archivdatenbank wird wiederhergestellt.
KopiereArchiv	Kopiert die Inhalte aus dem angegebenen Archiv in ein anderes Archiv.

3.8.5 Aufbau einer *.CSV-Datei mit Meldungen

Einleitung

Sie speichern bei Bedarf ein Meldearchiv als Datei im Format "*.csv". CSV heißt **C**omma **S**eparated **V**alue. In diesem Format werden die Tabellenspalten durch Semikola getrennt, die den Namen und den Wert des Eintrags enthalten. Jede Tabellenzeile wird durch einen Zeilenumbruch abgeschlossen.

Beispiel einer Datei im Format "*.csv"

Das Beispiel zeigt eine Datei mit archivierten Meldungen:

"Time_ms";"MsgProc";"StateAfter";"MsgClass";"MsgNumber";"Var1";...;"Var8";"TimeString";"MsgText";"PLC"37986550590,27;1;1;3;110001;"";...;"";"30.06.99 13:12:51";"Wechsel in die Betriebsart 'online'";37986550682,87;1;1;3;140010;"";...;"";"30.06.99 13:12:59";"Verbindung aufgebaut: Steuerung_1, Station 2, Rack 0, Platz 2";

Aufbau einer Archivdatei im Format "*.csv"

In den einzelnen Spalten einer Archivdatei sind folgende Werte eingetragen:

Parameter	Beschreibung
Time_ms	Zeitstempel als Dezimalwert angegeben (Umrechnung siehe unten)
Msg_Proc	Meldeverfahren: 0 = unbekanntes Meldeverfahren 1 = Systemmeldung 2 = Meldebitverfahren (Betriebsmeldungen) 3 = Meldenummernverfahren ALARM_S 4 = Diagnoseereignis 7 = Analogmeldeverfahren 100 = Meldebitverfahren (Störmeldungen)

Parameter	Beschreibung
StateAfter	Meldeereignis: 0 = Gekommen/Gegangen 1 = Gekommen 2 = Gekommen/Quittiert/Gegangen 3 = Gekommen/Quittiert 4 = Alle Meldungen in der PLC wurden mit SFC106 gelöscht oder nach PLC STOP/RUN steht eine Meldung an 6 = Gekommen/Gegangen/Quittiert
Msg_Class	Meldeklasse: 0 = keine Meldeklasse 1 = "Errors" 2 = "Warnings" 3 = "System" 4 = "Diagnosis Events" 64 ... = vom Benutzer projektierte Meldeklassen
MsgNumber	Meldenummer
Var1 ... Var8	Wert der Triggervariable als STRING
TimeString	Zeitstempel als STRING in einem lesbaren Datenformat
MsgText	Meldung in einem lesbaren STRING
PLC	Lokalisierung der Meldung (betroffene Steuerung)

Umrechnung des Dezimalwerts des Zeitstempels

Wenn Sie den Wert des Zeitstempels mit einem anderen Programm weiterverarbeiten, gehen Sie folgendermaßen vor:

1. Dividieren Sie Time_ms durch 1.000.000.
   Beispiel: 37986476928 : 1.000.000 = 37986,476928
2. Der ganzzahlige Anteil (37986) ist das Datum ab dem 31.12.1899 berechnet.
   In Excel rechnen Sie jetzt den Zeitstempel in Tage um. Weisen Sie dazu der Zelle, die den Zeitstempel enthält, ein entsprechendes Format aus der Gruppe "Datum" zu.
   Ergebnis: 37986 ergibt den 31.12.2003
3. Der Wert nach dem Komma (0,476928) ergibt die Uhrzeit:
   - Multiplizieren des Werts (0,476928) mit 24 ergibt die Stunden (11,446272).
   - Multiplizieren des Rests (0,446272) mit 60 ergibt die Minuten (26,77632)
   - Multiplizieren des Rests (0,77632) mit 60 ergibt die Sekunden (46,5792)

   Ergebnis 11:26:46,579
   Diese Umrechnung wird z. B. von Microsoft Excel unterstützt.

3.8.6 Beschreibung der Systemblöcke

Einleitung

Systemblöcke ermöglichen die Angabe von vordefinierten und nicht frei verwendbaren Informationen, z. B. Datum, Uhrzeit, Dauer.

In der Meldung wird der Wert des Meldeblocks, z. B. die Uhrzeit, angezeigt. Die Beschreibung der Systemblöcke liefert Ihnen Informationen zu den einzelnen Systemblöcken.

Übersicht

Systemblock	Beschreibung	Standardlänge
Datum	Datum der Meldezustände "kommend", "gegangen" und "quittiert".	8
Uhrzeit	Uhrzeit der Meldezustände "kommend", " gegangen " und "quittiert". Genauigkeit eines Zeitstempels, der in WinCC geprägt wird: 1 s. Anzeigegenauigkeit: 10 ms.	9
Dauer	Zeitraum zwischen den Zuständen "kommend", "gegangen " und "quittiert". Es gibt nur eine Spalte für die Dauer: • Bei gekommenen Meldungen ist diese Spalte leer. • Bei gegangenen Meldungen steht die Dauer von "gekommen" bis "gegangen". • Bei quittierten Meldungen steht die Dauer von "gekommen" bis "quittiert".	8
Sommer- / Winterzeit	Zeigt an, ob die Sommerzeit eingestellt ist.	1
Zustand	Zeigt den Meldezustand an, z. B. "gekommen" oder "gegangen". Die in diesem Systemblock angezeigten Buchstaben zur Darstellung des Meldezustands werden in der Meldeklasse vergeben, z. B. "Q".	1
Quittierstatus	Zeigt an, ob eine Meldung quittiert worden ist. Die in diesem Systemblock a angezeigten Buchstaben zur Darstellung des Quittierstatus werden in der Meldeklasse projektiert.	1
ID	Meldenummer	3
Meldeklasse	Meldeklasse Der Text ist frei wählbar.	8
AG/CPU-Nummer	Zeigt die Nummer der CPU und des AS an, in der die Meldung getriggert wurde. In Runtime wird der Wert dieses Systemblocks nicht vom AS übernommen, sondern aus den projektierten Daten der Meldung. Der Wert hat keine Funktion in der Kommunikation mit dem AS.	2
Variable	Variablenname für die Bedienmeldung vom E/A-Feld (und vergleichbare Objekte, die Bedienmeldungen ausgeben können)	1
Archivieren	Kennung, ob eine Meldung archiviert wird.	1
Kommentar	Zeigt an, ob zu dieser Meldung ein Kommentar existiert. Ein Kommentar ist die Eingabe eines Benutzers zu einer Meldung, z. B."Diese Meldung ist heute aufgetreten, weil...".	1

Systemblock	Beschreibung	Standardlänge
Infotext	Zeigt an, ob zu dieser Meldung ein Infotext (Tooltip) vorhanden ist. Tooltips enthalten Informationen zur Meldung mit max. 255 Zeichen, z. B. "Diese Meldung kann auftreten, wenn...". Tooltip sind in Runtime nicht editierbar.	1
Loop-In-Alarm	Zeigt an, ob die Funktion "Loop-In-Alarm" aktiviert ist.	1
Rechnername	Gibt den Namen des PC an: • Wenn der Benutzer in der Anzeige "Langzeitarchiv" einen Kommentar eingegeben und einen Fensterwechsel durchgeführt hat, wird der Name des verwendeten PC in den Anzeigen "Kurzzeitarchiv" und "Langzeitarchiv" angezeigt,. • Wenn eine Meldung auf einem PC quittiert wurde, wird der Name des PC in der Bedienmeldung in den Anzeigen "Kurz- und Langzeitarchiv" angezeigt. Die Bedienmeldung muss in der Meldeanzeige im Inspektorfenster unter "Bedienmeldung" aktiviert sein.	10
Benutzername	Gibt den Namen des Benutzers (Loginname) an: • Wenn der Benutzer in der Anzeige "Langzeitarchiv" einen Kommentar eingegeben hat und die Anzeige verlässt, wird der Benutzername in den Anzeigen "Kurzzeitarchiv" und "Langzeitarchiv" angezeigt. • Wenn eine Meldung von einem in WinCC angemeldeten Benutzer quittiert wurde, wird der Benutzername in der Bedienmeldung in den Anzeigen "Kurzzeitarchiv" und "Langzeitarchiv" angezeigt. Die Bedienmeldung muss in der Meldeanzeige im Inspektorfenster unter "Bedienmeldung" aktiviert sein.	10
Priorität	Legt die Priorität einer Meldung fest. Über die Priorität sortieren Sie die Meldungen in der Meldeanzeige. Durch die Sortierung nach der Priorität erreichen Sie, dass bei einer einzeiligen Meldeanzeige die wichtigste Meldung (hohe Priorität) im Anzeigebereich angezeigt ist. Eine Meldung mit niedrigerer Priorität wird nicht dargestellt, auch wenn sie ein aktuelleres Datum trägt. Wertebereich: 0 - 16 In WinCC ist nicht festgelegt, welcher Wert der höchsten Priorität entspricht.	3

Hinweis

Wenn Sie im Kontextmenü der WinCC RT Professional unter "Eigenschaften > Uhrzeiteinstellungen > Zentrales Datum- und Uhrzeitformat" die Option "ISO8601-Format auf allen Komponenten ändern" aktiviert haben, werden Datum und Zeit auch in den Systemblöcken der Meldungen im ISO8601-Format angezeigt.

3.8.7 SQL-Statements zum Filtern der Meldeanzeige

Einleitung

In der Meldeanzeige sind nur die SQL-Statements zulässig, die auch mit dem Dialog "Selektion festlegen" der Meldeanzeige erzeugt werden können. Für WinCC gelten folgende Bedingungen:

- Der Aufbau: "Feld", "Operand" und "Wert"
  Einzelnen Parameter mit Leerzeichen trennen.
  Beispiel:
  DATETIME >= '2006-12-21 00:00:00.000' AND MSGNR >= 100
  alle Meldungen ab 21.12.2006 mit einer Meldungsnummer größer und gleich 100
- Folgende Angaben werden in einfache Hochkommas gesetzt:
  - Zeichenketten
  - Datum
  - Uhrzeit
- Im Argument "DATETIME" sind Datum und Uhrzeit durch ein Leerzeichen zu trennen. Unabhängig von der Einstellung der Zeitbasis in den Objekteigenschaften basiert die Ausgabe von "DATETIME" auf der Zeitbasis "Local Time". Ausnahme bildet die Einstellung der Zeitbasis auf "UTC", dann basiert die Ausgabe auf der Zeitbasis "UTC"

Zugelassene Argumente

Name	SQL-Name	Typ	Daten	Beispiel
Datum/Uhrzeit	DATETIME	Datum	'YYYY-MM-TT hh:mm:ss.msmsms'	DATETIME >= '2007-05-03 16:00:00.000' Gibt Meldungen ab 03.05.2007, 16 Uhr 00 aus.
Nummer	MSGNR	Integer	Meldenummer	MSGNR >= 10 AND MSGNR <= 12 Gibt die Meldungen mit Meldenummern 10 - 12 aus.
Klasse	CLASS IN	Integer	Meldeklasse ID 1-256 und Meldeklassen "System" 17 + 18	CLASS IN (1) Gibt Meldungen der Meldeklasse 1 aus.
Zustand	STATE	Integer	Wert "ALARM_STATE_xx" Nur die Operanden "=" und "IN(...)" sind zulässig. ALARM_STATE_1 ALARM_STATE_2 ALARM_STATE_3 ALARM_STATE_4 ALARM_STATE_10 ALARM_STATE_11 ALARM_STATE_16 ALARM_STATE_17	STATE IN(1,2,3) Gibt alle gekommenen, gegangenen und quittierten Meldungen aus. Mögliche Werte: 1 = gekommene Meldungen 2 = gegangene Meldungen 3 = quittierte Meldungen 4 = gesperrte Meldungen 10 = ausgeblendete Meldungen 11 = eingeblendete Meldungen 16 = vom System quittierte Meldungen 17 = Notquittierte Meldungen

Name	SQL-Name	Typ	Daten	Beispiel
Priorität	PRIORITY	Integer	Priorität der Meldung 0 - 16	PRIORITY >= 1 AND PRIORITY =< 5 Gibt die Meldungen aus, die eine Priorität zwischen 1 und 5 haben.
AG-Nummer	AG_NR	Integer	AG-Nummer	AG_NR >= 2 AND AG_NR <= 2 Gibt Meldungen mit der AG-Nummer 2 aus.
CPU-Nummer	CPU_NR	Integer	AG-Subnummer	CPU_NR >= 5 AND CPU_NR <= 5 Gibt Meldungen mit der AG-Subnummer 5 aus.
Instanz	INSTANCE	Text	Instanz	-
Block: 1 ... Block: 10	TEXTxx	Text	Suchtext für Text1 - Text10	TEXT2 = 'Störung' Gibt die Meldungen aus, deren Text2 "Störung" entspricht. TEXT2 IN ('Störung','Fehler') Gibt die Meldungen aus, deren Text2 "Störung" oder "Fehler" entspricht. TEXT2 LIKE 'Störung' Gibt die Meldungen aus, deren Text2 "Störung" beinhaltet.
Prozesswert: 1 ... Prozesswert: 10	PVALUExx	Double	Suchwert für PVALUE1 - PVALUE10	PVALUE1 >= 0 AND PVALUE1 <= 50 Gibt den Prozesswert 1 mit Startwert 0 und Stoppwert 50 aus.

Folgende Operanden sind zugelassen

- **>= , <= , = , > , <**
- **IN(...)** : mehrere Werte als Array durch Kommata getrennt, z.B.: CLASS IN(1 ,2 ,3).
- **LIKE** : Text muss String nur beinhalten, z.B.: TEXT1 LIKE 'Störung' gibt die Meldung aus, deren Text1 den Suchtext "Störung" enthält. Der Operand LIKE ist nur für Argumente vom Typ "Text" zugelassen.

Nicht zugelassene Argumente und Operanden

Nur die in der Tabelle genannten Argumente und in der Liste aufgeführten Operanden sind zugelassen.

Nicht zugelassen ist die Gruppierung von Argumenten, z. B. durch Klammerung.

3.8.8 Systemmeldungen

3.8.8.1 Grundlagen zu Systemmeldungen

Systemmeldungen

Systemmeldungen geben am Bediengerät Auskunft über interne Zustände des Bediengeräts und der Steuerung.

Nachfolgend erhalten Sie eine Übersicht, wann eine Systemmeldung auftritt und wie Sie die Fehlerursache beheben.

Hinweis

Bediengeräteabhängigkeit

Für die einzelnen Bediengeräte gilt je nach Funktionsumfang eine Teilmenge der beschriebenen Systemmeldungen.

Hinweis

Systemmeldungen werden in einer Meldeanzeige ausgegeben. Systemmeldungen werden in der Sprache ausgegeben, die aktuell an Ihrem Bediengerät eingestellt ist.

Das Zeitformat (AM/PM oder 24-Stunden-Format) wird von der eingestellten Sprache übernommen. Wenn keine Übersetzung der Meldetexte in dieser Sprache existiert, wird Englisch oder Russisch als Ersatzsprache gewählt und das entsprechende Zeitformat angezeigt.

Parameter der Systemmeldungen

Systemmeldungen können verschlüsselte Parameter enthalten. Die Parameter sind zur Verfolgung eines Fehlers relevant, da sie Hinweise auf den Quellcode der Runtime-Software geben. Die Parameter werden nach dem Text "Fehlercode:" ausgegeben.

3.8.8.2 9999 - Meldung Global

Bedeutung der Systemmeldung

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-1 9999 - Meldung Global

Nummer	Wirkung/Ursache	Abhilfe
9999	Der Systemstart wurde abgebrochen. Überprüfen Sie Folgendes: • Netzspannung • Kabelverbindungen • Systemeinstellungen • Projektierungen • Konvertierungen	Überprüfen Sie die Spannungsversorgung und Kabelverbindungen. Überprüfen Sie die Systemeinstellungen. Überprüfen Sie die Projektierungen und Konvertierungen. Tritt der Fehler wiederholt auf, wenden Sie sich an die Hotline!

3.8.8.3 10000 - Meldungen Drucker

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-2 10000 - Meldungen Drucker

Nummer	Wirkung/Ursache	Abhilfe
10000	Der Druckauftrag konnte aus unbekanntem Grund nicht gestartet werden oder wurde abgebrochen. Der Drucker ist nicht richtig eingerichtet. Oder: es liegt keine Berechtigung für einen Netzwerkdrucker vor. Während der Datenübertragung erfolgte eine Unterbrechung der Netzspannung.	Überprüfen Sie die Druckereinstellungen, Kabelverbindungen und Spannungsversorgung. Richten Sie den Drucker noch einmal ein. Lassen Sie sich eine Berechtigung für den Netzwerkdrucker geben. Tritt der Fehler wiederholt auf, wenden Sie sich an die Hotline!
10001	Es ist kein Drucker installiert oder kein Standarddrucker eingerichtet.	Installieren Sie einen Drucker und/oder markieren Sie ihn als Standarddrucker.
10002	Der Zwischenpuffer für das Ausdrucken von Grafiken ist voll. Es werden bis zu zwei Grafiken gepuffert.	Stoßen Sie das Drucken nicht so schnell hintereinander an.
10003	Grafiken können wieder zwischengespeichert werden.	--
10004	Der Zwischenpuffer für das Ausdrucken von Zeilen im Textmodus (z. B. Meldungen) ist voll. Es werden bis zu 1000 Zeilen gepuffert.	Stoßen Sie das Drucken nicht so schnell hintereinander an.
10005	Textzeilen können wieder zwischengespeichert werden.	--
10006	Das Drucksystem von Windows meldet einen Fehler. Mögliche Ursachen entnehmen Sie bitte dem ausgegebenen Text und gegebenenfalls der Fehlernummer. Es wird nicht oder nicht richtig gedruckt.	Wiederholen Sie gegebenenfalls die Aktion.

3.8.8.4 20000 - Meldungen Globale Skripte

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 3-3 20000 - Meldungen Globale Skripte

Nummer	Wirkung/Ursachen	Abhilfe
20010	In der angegebenen Skript-Zeile ist ein Fehler aufgetreten. Die Ausführung des Skripts wurde daher abgebrochen. Beachten Sie hierzu evtl. auch die vorherige Systemmeldung.	Wählen Sie in der Projektierung die angegebene Zeile im Skript an. Prüfen Sie bei Variablen, ob die verwendeten Typen zulässig sind. Prüfen Sie bei Systemfunktionen, ob die Anzahl und die Typen der Parameter korrekt sind.
20011	Es ist ein Fehler in einem Skript aufgetreten, das von dem angegebenen Skript aufgerufen wurde. Die Ausführung des Skripts wurde daher im aufgerufenen Skript abgebrochen. Beachten Sie hierzu evtl. auch die vorherige Systemmeldung.	Wählen Sie in der Projektierung die Skripte an, die vom angegebenen Skript direkt oder indirekt aufgerufen werden. Prüfen Sie bei Variablen, ob die verwendeten Typen zulässig sind. Prüfen Sie bei Systemfunktionen, ob die Anzahl und die Typen der Parameter korrekt sind.
20012	Es liegen inkonsistente Projektierungsdaten vor. Das Skript konnte daher nicht erzeugt werden.	Übersetzen Sie die Projektierung neu.
20013	Die Skriptkomponente von WinCC Runtime ist nicht richtig installiert. Es können daher keine Skripte ausgeführt werden.	Installieren Sie WinCC Runtime auf Ihrem PC neu. Übersetzen Sie Ihr Projekt mit dem Kontextbefehl "Übersetzen > Software (komplett übersetzen)" erneut und laden Sie das Projekt in das Bediengerät.
20014	Von der Systemfunktion wird ein Wert zurückgegeben, der in keine projektierte Rückgabevariable geschrieben wird.	Wählen Sie in der Projektierung das angegebene Skript an. Prüfen Sie, ob dem Skriptnamen ein Wert zugewiesen wird.
20015	Es wurden zu viele Skripte kurz hintereinander angestoßen. Stehen mehr als 20 Skripte zur Bearbeitung an, werden die nachfolgenden Skripte verworfen. In diesem Fall wird das in der Meldung angegebene Skript nicht ausgeführt.	Überprüfen Sie, wodurch die Skripte ausgelöst werden. Verlängern Sie die Zeiten, z. B. den Erfassungszyklus der Variablen, die das Skript anstößt.

3.8.8.5 30000 - Meldungen Fehler bei Verwendung von Systemfunktionen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 3-4 30000 - Meldungen Fehler bei Verwendung von Systemfunktionen

Nummer	Wirkung/Ursachen	Abhilfe
30010	Die Variable konnte das Ergebnis der Systemfunktion nicht aufnehmen, z. B. bei Wertebereichsüberschreitung.	Überprüfen Sie die Variablentypen der Parameter der Systemfunktion.
30011	Eine Systemfunktion konnte nicht ausgeführt werden, da im Parameter der Systemfunktion ein unzulässiger Wert oder Typ übergeben wurde.	Überprüfen Sie den Parameterwert und Variablentyp des unzulässigen Parameters. Falls als Parameter eine Variable verwendet wird, überprüfen Sie deren Wert.
30012	Eine Systemfunktion konnte nicht ausgeführt werden, da im Parameter der Systemfunktion ein unzulässiger Wert oder Typ übergeben wurde.	Überprüfen Sie den Parameterwert und Variablentyp des unzulässigen Parameters. Falls als Parameter eine Variable verwendet wird, überprüfen Sie deren Wert.

3.8.8.6 40000 - Meldungen Lineare Skalierung

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 3-5 40000 - Meldungen Lineare Skalierung

Nummer	Wirkung/Ursachen	Abhilfe
40010	Die Systemfunktion konnte nicht ausgeführt werden, da die Parameter nicht auf einen gemeinsamen Variablentyp konvertiert werden können.	Überprüfen Sie die Parametertypen in der Projektierung.
40011	Die Systemfunktion konnte nicht ausgeführt werden, da die Parameter nicht auf einen gemeinsamen Variablentyp konvertiert werden können.	Überprüfen Sie die Parametertypen in der Projektierung.

3.8.8.7 50000 - Meldungen Datenserver

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 3-6 50000 - Meldungen Datenserver

Nummer	Wirkung/Ursachen	Abhilfe
50000	Das Bediengerät erhält schneller Daten als es bearbeiten kann. Es werden daher solange keine neuen Daten angenommen, bis die vorhandenen Daten bearbeitet wurden. Danach wird der Datenaustausch wieder aufgenommen.	--
50001	Der Datenaustausch wurde wieder aufgenommen.	--

3.8.8.8 60000 - Meldungen Windows-Funktionen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-7 60000 - Meldungen Win32 Funktionen

Nummer	Wirkung/Ursachen	Abhilfe
60000	Diese Meldung wird durch die Systemfunktion "ZeigeSystemmeldung" erzeugt. Der anzuzeigende Text wird als Parameter an die Systemfunktion übergeben.	--
60010	Die Datei konnte nicht in der angegebenen Richtung kopiert werden, da eine der beiden Dateien zur Zeit geöffnet oder der Quell-/Ziel-Pfad nicht vorhanden ist. Evtl. hat der Windows-Benutzer kein Zugriffsrecht auf eine der beiden Dateien.	Starten Sie die Systemfunktion erneut oder überprüfen Sie den Pfad der Quell-/Ziel-Datei. Unter Windows: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, auf die Dateien zugreifen zu dürfen.
60011	Es wurde versucht eine Datei auf sich selbst zu kopieren. Evtl. hat der Windows-Benutzer kein Zugriffsrecht auf eine der beiden Dateien.	Überprüfen Sie den Pfad der Quell-/Ziel-Datei. Unter Windows mit NTFS: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, auf die Dateien zugreifen zu dürfen.

3.8.8.9 70000 - Meldungen Windows-Funktionen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-8 70000 - Meldungen Win32 Funktionen

Nummer	Wirkung/Ursachen	Abhilfe
70010	Das Programm konnte nicht gestartet werden, da es im angegebenen Pfad nicht gefunden wurde oder weil nicht genügend freier Speicherplatz vorhanden ist.	Überprüfen Sie, ob das Programm im angegebenen Pfad oder Suchpfad existiert oder schließen Sie andere Programme.
70011	Die Systemzeit konnte nicht geändert werden. Die Fehlermeldung erscheint nur im Zusammenhang mit Bereichszeiger "Datum/Uhrzeit Steuerung". Mögliche Ursachen: • Im Steuerungsauftrag wurde eine unzulässige Zeit übergeben. • Der Windows-Benutzer hat kein Benutzerrecht, um die Systemzeit zu ändern. Wird in der Systemmeldung als erster Parameter der Wert 13 angezeigt, so kennzeichnet der zweite Parameter das Byte, das den falschen Wert hat.	Überprüfen Sie die Zeit, die gesetzt werden soll. Unter Windows: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, die Systemzeit des Betriebssystems zu ändern.
70012	Bei der Ausführung der Systemfunktion "Stoppe-Runtime" mit der Option "Runtime und Betriebssystem" trat ein Fehler auf. Windows und WinCC Runtime werden nicht beendet. Eine mögliche Ursache ist, dass sich andere Programme nicht beenden lassen.	Beenden Sie alle zur Zeit laufenden Programme. Beenden Sie dann Windows.
70013	Die Systemzeit konnte nicht geändert werden, da der eingegebene Wert nicht zulässig ist. Evtl. wurden falsche Trennzeichen verwendet.	Überprüfen Sie die Zeit, die gesetzt werden soll.
70014	Die Systemzeit konnte nicht geändert werden. Mögliche Ursachen: • Es wurde eine unzulässige Zeit übergeben. • Der Windows-Benutzer hat kein Benutzerrecht um die Systemzeit zu ändern. Windows lehnt das Setzen ab.	Überprüfen Sie die Zeit, die gesetzt werden soll. Unter Windows: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, die Systemzeit des Betriebssystems zu ändern.
70015	Die Systemzeit konnte nicht gelesen werden, da Windows das Lesen ablehnt.	--
70016	Es wurde versucht, durch eine Systemfunktion oder einen Auftrag ein Bild anzuwählen. Dies ist nicht möglich, da die projektierte Bildnummer nicht existiert. Oder: ein Bild konnte wegen unzureichendem Systemspeicher nicht aufgebaut werden. Oder: Das Bild ist gesperrt. Oder: Bildaufruf wird nicht korrekt ausgeführt.	Vergleichen Sie die Bildnummer in der Systemfunktion oder im Auftrag mit den projektierten Bildnummern. Weisen Sie gegebenenfalls die Nummer einem Bild zu. Kontrollieren Sie die Angaben zum Bildaufruf und ob das Bild für bestimmte Benutzer gesperrt ist.

Nummer	Wirkung/Ursachen	Abhilfe
70017	Datum/Uhrzeit wird nicht aus dem Bereichszeiger gelesen, da die eingestellte Adresse in der Steuerung nicht vorhanden oder nicht eingerichtet ist.	Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.
70018	Rückmeldung für erfolgreichen Import der Kennwortliste.	--
70019	Rückmeldung für erfolgreichen Export der Kennwortliste.	--
70020	Rückmeldung für Aktivierung der Meldeprotokollierung.	--
70021	Rückmeldung für Deaktivierung der Meldeprotokollierung.	--
70022	Rückmeldung für den Start der Aktion Kennwortliste importieren.	--
70023	Rückmeldung für den Start der Aktion Kennwortliste exportieren.	--
70024	Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.	Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.
70025	Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.	Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.
70026	Es sind keine weiteren Bilder in dem internen Bilderspeicher gespeichert. Es ist keine weitere Bildanwahl möglich.	--
70027	Das Sichern des RAM-Dateisystems wurde gestartet.	--
70028	Das Sichern des RAM-Dateisystems wurde erfolgreich beendet. Die Dateien aus dem RAM werden ausfallsicher in den Flash-Speicher kopiert. Bei Neuanlauf werden diese gesicherten Dateien wieder in das RAM-Dateisystem zurückkopiert.	--
70029	Das Sichern des RAM-Dateisystems ist fehlgeschlagen. Das RAM-Dateisystem wurde nicht gesichert.	Überprüfen Sie die Einstellungen im Dialog "Control Panel > OP" und sichern Sie das RAM-Dateisystem über die Schaltfläche "Save Files" in der Registerkarte "Persistent Storage".
70030	Die projektierten Parameter der Systemfunktion sind fehlerhaft. Die Verbindung zur neuen Steuerung wurde nicht aufgebaut.	Vergleichen Sie die projektierten Parameter der Systemfunktion mit den projektierten Parametern der Steuerungen und korrigieren Sie diese gegebenenfalls.
70031	Die projektierte Steuerung in der Systemfunktion ist keine S7-Steuerung. Die Verbindung zur neuen Steuerung wurde nicht aufgebaut.	Vergleichen Sie den projektierten Parameter S7-Steuerungsname der Systemfunktion mit den projektierten Parametern der Steuerungen und korrigieren Sie diesen gegebenenfalls.
70032	In dem angewählten Bild ist das projektierte Objekt mit dieser Nummer in der Tab-Reihenfolge nicht vorhanden. Der Bildwechsel wird ausgeführt, der Fokus wird jedoch auf das erste Objekt gesetzt.	Überprüfen Sie die Nummer in der Tab-Reihenfolge und korrigieren Sie diese gegebenenfalls.

Nummer	Wirkung/Ursachen	Abhilfe
70033	Eine E-Mail konnte nicht versandt werden, weil keine TCP/IP-Verbindung zum SMTP-Server mehr besteht. Die Systemmeldung wird nur beim ersten Fehlversuch generiert. Alle folgenden erfolglosen Versuche, eine E-Mail zu versenden, erzeugen keine Systemmeldung mehr. Die Meldung wird erst wieder erzeugt, wenn inzwischen eine E-Mail erfolgreich versandt werden konnte. Die zentrale E-Mail-Komponente in WinCC Runtime versucht in regelmäßigen Abständen (1 min.), die Verbindung zum SMTP-Server aufzubauen und die verbleibenden E-Mails zu versenden.	Überprüfen Sie die Netzwerkverbindung zum SMTP-Server und stellen Sie diese gegebenenfalls wieder her.
70034	Nach einem Verbindungsabbruch konnte die TCP/IP Verbindung zum SMTP-Server wieder hergestellt werden. Die in der Warteschlange verbliebenen E-Mails werden versandt.	--
70036	Es wurde kein SMTP-Server für den Versand von E-Mails konfiguriert. Eine Verbindung zu einem SMTP-Server kann daher nicht hergestellt werden und es können keine E-Mails gesendet werden. Die Systemmeldung wird durch WinCC Runtime beim ersten Versuch eine E-Mail zu senden, generiert.	Konfigurieren Sie einen SMTP-Server: In WinCC Engineering System über "Geräteeinstellungen > Geräteeinstellungen" Im Betriebssystem Windows CE über "Control Panel > Internet Settings > Email > SMTP Server"
70037	Eine E-Mail konnte aus unbekannten Gründen nicht verschickt werden. Der Inhalt der E-Mail wird verworfen.	Überprüfen Sie die E-Mail-Parameter (Empfänger etc).
70038	Der SMTP-Server hat die Weiterleitung bzw. den Versand der E-Mail abgelehnt, weil die Domäne des Empfängers dem Server nicht bekannt ist oder weil der SMTP-Server eine Authentifizierung benötigt. Inhalt der E-Mail wird verworfen.	Domäne der Empfänger Adresse überprüfen oder Authentifizierung am SMTP-Server deaktivieren falls möglich. Eine SMTP-Authentifizierung wird gegenwärtig von WinCC Runtime nicht behandelt.
70039	Die Syntax der E-Mail Adresse ist falsch oder enthält ungültige Zeichen. E-Mail Inhalt wird verworfen.	E-Mail Adresse des Empfänger Adresse überprüfen.
70040	Die Syntax der E-Mail Adresse ist falsch oder enthält ungültige Zeichen.	--
70041	Der Import der Benutzerverwaltung wurde wegen eines Fehlers abgebrochen. Es wurde kein Import durchgeführt.	Überprüfen Sie ihre Benutzerverwaltung oder laden Sie Ihre Benutzerverwaltung erneut auf das Panel.
70042	Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.	Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.
70043	Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.	Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.
70044	Beim Senden der E-Mails ist ein Fehler aufgetreten. Die E-Mails wurden nicht gesendet.	Überprüfen Sie die SMTP-Einstellungen und beachten Sie die Fehlermeldung in der Systemmeldung.

Nummer	Wirkung/Ursachen	Abhilfe
70045	Eine Datei, die für Verschlüsselung der E-Mail notwendig ist, konnte nicht geladen werden.	Aktualisieren Sie das Betriebssystem und die Runtime.
70046	Der Server unterstützt die Verschlüsselung nicht.	Wählen Sie einen SMTP-Server, der eine Verschlüsselung unterstützt.
70047	Möglicherweise passen die SSL-Versionen von Bediengerät und SMTP-Server nicht zusammen.	Kontaktieren Sie Ihren Netzwerkadministrator oder den Betreiber des SMTP-Servers.

3.8.8.10 80000 - Meldungen Archive

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-9 80000 - Meldungen Archive

Nummer	Wirkung/Ursachen	Abhilfe
80001	Das angegebene Archiv ist bis zur angegebenen Größe (in Prozent) gefüllt und muss ausgelagert werden.	Lagern Sie die Datei oder die Tabelle durch Verschieben oder durch eine Kopierfunktion aus.
80002	Ein Eintrag im angegebenen Archiv fehlt.	--
80003	Der Kopiervorgang bei Archiven ist fehlgeschlagen. Beachten Sie hierzu evtl. auch die nachfolgende Systemmeldung.	--
80006	Da keine Archivierung möglich ist, bewirkt dies einen dauerhaften Verlust der Funktionalität.	Überprüfen Sie im Fall von Datenbanken, ob die entsprechende Datenquelle existiert und fahren Sie das System neu hoch.
80009	Eine Kopieraktion wurde erfolgreich beendet.	--
80010	Da der Ablageort in WinCC fehlerhaft eingegeben wurde, bewirkt dies einen dauerhaften Verlust der Funktionalität.	Projektieren Sie den Ablageort für das jeweilige Archiv neu und fahren Sie das System neu hoch, wenn die volle Funktionalität gefordert ist.
80012	Archiveinträge werden in einem Puffer gespeichert. Wenn die Werte schneller in den Puffer eingetragen werden sollen, als physikalisch geschrieben werden können (z. B. bei Festplatte), kann es zur Überlast kommen und die Aufzeichnung wird gestoppt.	Archivieren Sie weniger Werte. Oder: Erhöhen Sie den Archivierungszyklus.
80013	Der Überlastzustand ist beendet. Die Archivierung zeichnet wieder alle Werte auf.	--
80014	Es wurde zweimal kurz hintereinander dieselbe Aktion ausgelöst. Da das Umkopieren bereits läuft, wird die Aktion nicht noch einmal durchgeführt.	--
80015	Diese Systemmeldung wird dazu verwendet, um DOS- oder Datenbankfehler an den Anwender zu melden.	--
80016	Die Archive sind durch die Systemfunktion "SchliesseAlleArchive" getrennt und die einlaufenden Einträge überschreiten die Größe des Zwischenpuffers. Alle Einträge im Zwischenpuffer werden gelöscht.	Verbinden Sie die Archive erneut.

Nummer	Wirkung/Ursachen	Abhilfe
80017	Die einlaufenden Einträge überschreiten die Größe des Zwischenpuffers. Dies kann z. B. durch mehrere gleichzeitig laufende Kopieraktionen verursacht werden. Alle Kopieraufträge im Zwischenpuffer werden gelöscht.	Beenden Sie den Kopiervorgang.
80019	Die Verbindung zwischen allen Archiven und WinCC wurde getrennt, z. B. nach Ausführung der Systemfunktion "SchliesseAlleArchive". Einträge werden zwischengespeichert und bei erneuter Verbindung in die Archive geschrieben. Es besteht keine Verbindung zum Ablageort und es kann z.B. ein Wechsel des Datenträgers erfolgen.	--
80020	Die max. Anzahl gleichzeitig laufender Kopieraktionen wurde überschritten. Die Kopie wird nicht ausgeführt.	Warten Sie, bis die laufenden Kopieraktionen beendet sind und starten Sie die letzte Kopieraktion erneut.
80021	Es wird versucht, ein Archiv zu löschen, das noch mit einer Kopieraktion beschäftigt ist. Das Löschen wird nicht ausgeführt.	Warten Sie, bis die laufende Kopieraktion beendet ist und starten Sie die letzte Aktion erneut.
80022	Es wurde versucht, durch die Systemfunktion "StarteNaechstesArchiv" ein Folgearchiv bei einem Archiv zu beginnen, das nicht als Folgearchiv projektiert wurde. Es wird kein Folgearchiv angelegt.	Überprüfen Sie in Ihrem Projekt, ob • die Systemfunktion "StarteNaechstesArchiv" korrekt projektiert wurden • die Variablenparameter am Bediengerät korrekt versorgt wurden.
80023	Es wird versucht, ein Archiv auf sich selbst zu kopieren. Das Archiv wird nicht kopiert.	Überprüfen Sie in Ihrem Projekt, ob • die Systemfunktion "KopiereArchiv" korrekt projektiert wurden • die Variablenparameter am Bediengerät korrekt versorgt wurden.
80024	In Ihrer Projektierung ist für die Systemfunktion "KopiereArchiv" vorgegeben, keine Kopie zuzulassen, wenn das Zielarchiv bereits Daten enthält (Parameter "Modus"). Das Archiv wird nicht kopiert.	Ändern Sie gegebenenfalls in Ihrer Projektierung die Systemfunktion "KopiereArchiv". Löschen Sie vor dem Anstoß der Systemfunktion das Zielarchiv.
80025	Sie haben die Kopieraktion abgebrochen. Die bis zu diesem Zeitpunkt geschriebenen Einträge bleiben bestehen. Das Löschen des Zielarchivs (falls projektiert) wird nicht durchgeführt. Der Abbruch wird durch einen Fehlereintrag RT_ERR am Ende des Zielarchivs dokumentiert.	--
80026	Die Meldung wird nach erfolgreicher Initialisierung aller Archive ausgegeben. Ab diesem Zeitpunkt werden Einträge in die Archive geschrieben. Davor werden trotz laufender WinCC Runtime keine Einträge in die Archive geschrieben.	--
80027	Als Ablageort für ein Archiv wurde der interne Flash-Speicher angegeben. Dies ist nicht zulässig. Für dieses Archiv werden keine Einträge archiviert und das Archiv wird nicht angelegt.	Projektieren Sie als Ablageort "Storage Card" oder einen Netzwerkpfad.

Nummer	Wirkung/Ursachen	Abhilfe
80028	Die Meldung dient als Statusrückmeldung, dass momentan die Initialisierung der Archive läuft. Bis zur Ausgabe der Meldung 80026 werden keine Einträge archiviert.	--
80029	Die in der Meldung angegebene Anzahl von Archiven konnte nicht initialisiert werden. Die Initialisierung der Archive wurde beendet. Die fehlerhaften Archive stehen für Archivierungsaufgaben nicht zur Verfügung.	Werten Sie die im Zusammenhang mit dieser Meldung ausgegebenen zusätzlichen Systemmeldungen aus. Überprüfen Sie die Projektierung, die ODBC (Open Database Connectivity) und das angegebene Laufwerk.
80030	Die Struktur des vorhandenen Archivs passt nicht zur erwarteten Archivstruktur. Die Archivierung wird für dieses Archiv gestoppt.	Löschen Sie vorab manuell die vorhandenen Archivdaten.
80031	Das Archiv im csv-Format ist beschädigt. Das Archiv kann nicht mehr verwendet werden.	Löschen Sie die fehlerhafte Datei.
80032	Archive können mit Ereignissen projektiert werden. Diese werden ausgelöst, sobald das Archiv voll ist. Wird WinCC Runtime gestartet und das Archiv ist bereits voll, würde das Ereignis nie ausgelöst werden. Das genannte Archiv archiviert nicht mehr, da es voll ist.	Beenden Sie WinCC Runtime, löschen Sie das Archiv und starten Sie WinCC Runtime erneut. Oder: Projektieren Sie eine Schaltfläche, welche die gleichen Aktionen wie das Ereignis enthält und betätigen Sie diese.
80033	Im Data-Archiv wurde als Data Source Name "System Defined" gewählt. Dies verursachte einen Fehler. Es erfolgt keine Archivierung in die Datenbank-Archive, während die Archivierung in die csv-Archive funktioniert.	SQL Sever 2005 Express neu installieren.
80034	Fehler bei der Initialisierung der Archive. Es wurde versucht die Tabellen als Backup anzulegen. Das hat funktioniert. Es sind von den Tabellen des fehlerhaften Archivs Backups angelegt worden und das Archiv hat neu (leer) aufgesetzt.	Es ist keine Behebung notwendig. Es wird aber empfohlen, die Backups zu sichern oder zu löschen, um den Speicher wieder freizugeben.
80035	Fehler bei der Initialisierung der Archive. Es wurde versucht die Tabellen als Backup anzulegen, was fehlgeschlagen ist. Es hat keine Archivierung und kein Backup stattgefunden.	Es wird empfohlen, die Backups zu sichern oder zu löschen, um den Speicher wieder freizugeben.
80044	Der Export eines Archivs wurde durch Beenden der Runtime oder durch einen Stromausfall unterbrochen. Beim Neuanlauf der Runtime wurde festgestellt, dass der Export fortgesetzt werden muss.	Der Export wird selbsttätig fortgesetzt.
80045	Der Export eines Archivs wurde durch einen Fehler in der Verbindung zum Server oder am Server selbst unterbrochen.	Der Export wird selbsttätig wiederholt. Bitte überprüfen Sie • die Verbindung zum Server • ob der Server läuft • ob auf dem Server genügend freier Speicherplatz vorhanden ist.
80046	Auf dem Server konnte die Zieldatei oder das zugehörige Verzeichnis nicht erstellt werden.	Überprüfen Sie, ob auf dem Server genügen Speicher vorhanden ist und die Berechtigung ausreicht um die Archivdatei abzulegen.
80047	Die Archivdatei konnte nicht gelesen werden.	Überprüfen Sie, ob das Speichermedium korrekt eingesteckt ist.

Nummer	Wirkung/Ursachen	Abhilfe
80049	Beim Vorbereiten des Exports eines Archivs konnte dieses nicht umbenannt werden. Der Auftrag wurde nicht ausgeführt."	Überprüfen Sie, ob das Speichermedium korrekt eingesteckt ist und noch ausreichend Speicherplatz frei ist.
80050	Das Archiv, welches exportiert werden soll, ist nicht geschlossen. Der Auftrag wurde nicht ausgeführt.	Stellen Sie sicher, dass vor Verwendung der Systemfunktion "ExportiereArchiv" die Systemfunktion "SchließeAlleArchive" aufgerufen wird. Ändern Sie bei Bedarf die Projektierung.
80051	Das Archiv, das kopiert werden soll, enthält eine ungültige Checksumme. Das Archiv wurde nicht kopiert.	Wählen Sie ein Archiv mit einer gültigen Checksumme. Das ausgewählte Archiv wurde möglicherweise manipuliert.
80052	Das Lesen des Archivs ist nicht möglich.	Überprüfen Sie das Archiv und den angegebenen Pfad.
80053	Das Lesen des geschlossenen Archivs ist nicht möglich.	Öffnen Sie das Archiv.
80054	Das Speichermedium ist nicht verfügbar.	Überprüfen Sie, ob das Speichermedium korrekt eingesteckt ist.

3.8.8.11 90000 - Meldungen FDA

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-10 90000 - Meldungen FDA

Nummer	Wirkung/Ursachen	Abhilfe
90024	Da auf dem Speichermedium des Archivs kein freier Speicherplatz mehr verfügbar ist, können keine Bedienhandlungen protokolliert werden. Die Bedienhandlung wird somit nicht durchgeführt.	Schaffen Sie mehr freien Speicherplatz auf dem Speichermedium, indem Sie entweder ein leeres Speichermedium einstecken oder die Archivdateien mittels "ExportiereArchiv" auf dem Server sichern.
90025	Wegen eines Fehlers im Archiv können keine Benutzeraktionen archiviert werden. Deshalb wird die Benutzeraktion nicht ausgeführt.	Überprüfen Sie, ob das Speichermedium korrekt eingesteckt ist.
90026	Da das Archiv geschlossen ist, können keine Bedienhandlungen protokolliert werden. Die Bedienhandlung wird nicht durchgeführt.	Vor weiteren Bedienhandlungen müssen die Archive wieder mit Hilfe der Systemfunktion "ÖffneAlleArchive" geöffnet werden. Ändern Sie bei Bedarf die Projektierung.
90028	Das eingegebene Passwort ist falsch.	Geben Sie das korrekte Passwort ein.
90029	Die Runtime wurde während des laufenden Betriebs beendet (evtl. durch einen Stromausfall) oder es wird ein Speichermedium mit einem nicht passenden Audit Trail verwendet. Ein Audit Trail ist dann nicht passend, wenn es zu einem anderen Projekt gehört oder bereits archiviert wurde. Oder die Archivierung wurde nicht gestoppt bevor die automatisch ausgeführte Funktion "SchließeAlleArchive" (z.B. durch den Aufgabenplaner) gestartet wurde.	Stellen Sie sicher, dass das richtige Speichermedium eingesetzt wird.
90030	Die Runtime wurde während des laufenden Betriebs beendet (evtl. durch einen Stromausfall).	--

Nummer	Wirkung/Ursachen	Abhilfe
90031	Die Runtime wurde während des laufenden Betriebs beendet (evtl. durch einen Stromausfall).	--
90032	Auf dem Speichermedium des Archivs ist nur noch wenig Speicherplatz verfügbar.	Schaffen Sie mehr freien Speicherplatz auf dem Speichermedium, indem sie entweder ein leeres Speichermedium einstecken oder die Archivdateien mittels "ExportiereArchiv" auf dem Server sichern.
90033	Auf dem Speichermedium ist kein freier Speicherplatz mehr für das Archiv verfügbar. Es können keine protokollierpflichtigen Bedienhandlungen mehr durchgeführt werden.	Schaffen Sie mehr Platz auf dem Speichermedium, indem Sie entweder ein leeres Speichermedium einstecken, oder die Archivdateien mittels "ExportiereArchiv" auf dem Server sichern.
90039	Sie haben nicht die erforderliche Berechtigung diese Aktion auszuführen.	Berechtigungen anpassen bzw. erweitern.
90040	Audit Trail wurde durch das Erzwingen einer Benutzeraktion abgeschaltet.	Schalten Sie das "Audit Trail" mit Hilfe der Systemfunktion "StarteArchiv" wieder ein.
90041	Eine protokollierpflichtige Benutzeraktion wurde ohne angemeldeten Benutzer ausgeführt.	Eine protokollierpflichtige Aktion sollte nicht ohne Berechtigung möglich sein. Ändern Sie die Projektierung, in dem Sie an dem Eingabeelement eine notwendige Berechtigung projektieren.
90044	Eine bestätigungspflichtige Benutzeraktion wurde blockiert, da eine weitere Benutzeraktion ansteht.	Wiederholen Sie die blockierte Bedienhandlung.
90048	Das Audit Trail kann nicht gedruckt werden, während Daten, die für das Audit relevant sind, archiviert werden.	Beenden Sie die Archivierung mit Hilfe der Systemfunktion "StoppeArchivierung".
90049	Der Zugriff auf eine benötigte Datei ist nicht möglich.	Überprüfen Sie die Netzwerkverbindung oder das Speichermedium.
90056	Die Rezeptur wurde nicht importiert, da die Datei keine Checksumme enthält.	Wählen Sie eine Datei mit Checksumme. Alternativ deaktivieren Sie die Prüfung der Checksumme, indem Sie die Systemfunktion "ImportiereDatensaetze" nutzen.
90057	Die Rezeptur wurde nicht importiert, da die Datei eine ungültige Checksumme enthält. Die ausgewählte Datei wurde möglicherweise manipuliert.	Wählen Sie eine Datei mit einer gültigen Checksumme.

3.8.8.12 110000 - Meldungen Offline Funktionen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-11 110000 - Meldungen Offline Funktionen

Nummer	Wirkung/Ursachen	Abhilfe
110000	Es wurde ein Wechsel der Betriebsart durchgeführt. Die Betriebsart ist jetzt "Offline".	--
110001	Es wurde ein Wechsel der Betriebsart durchgeführt. Die Betriebsart ist jetzt "Online".	--

Nummer	Wirkung/Ursachen	Abhilfe
110002	Die Betriebsart wurde nicht gewechselt.	Überprüfen Sie die Verbindung zu den Steuerungen. Überprüfen Sie, ob der Adressbereich für den Bereichszeiger 88"Koordinierung" in der Steuerung vorhanden ist.
110003	Die Betriebsart der angegebenen Steuerung wurde durch die Systemfunktion "SetzeVerbindungsmodus" gewechselt. Die Betriebsart ist jetzt "Offline".	--
110004	Die Betriebsart der angegebenen Steuerung wurde durch die Systemfunktion "SetzeVerbindungsmodus" gewechselt. Die Betriebsart ist jetzt "Online".	--
110005	Es wurde versucht, über die Systemfunktion "SetzeVerbindungsmodus" die angegebene Steuerung in die Betriebsart "Online" zu schalten, obwohl sich das Gesamtsystem in der Betriebsart "Offline" befindet. Diese Umschaltung ist nicht zulässig. Die Betriebsart der Steuerung bleibt "Offline".	Schalten Sie das Gesamtsystem in die Betriebsart "Online" und führen Sie dann die Systemfunktion erneut aus.
110006	Der Inhalt des Bereichszeigers "Projektkennung" stimmt nicht mit der in WinCC projektierten Projektkennung überein. Deshalb wird WinCC Runtime beendet.	Überprüfen Sie: • die in der Steuerung eingetragene Projektkennung. • die in WinCC eingetragene Projektkennung.

3.8.8.13 120000 - Meldungen Kurvendarstellung

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-12 120000 - Meldungen Kurvendarstellung

Nummer	Wirkung/Ursachen	Abhilfe
120000	Die Kurve wird nicht dargestellt, da eine falsche Achse zur Kurve bzw. eine falsche Kurve projektiert wurde.	Ändern Sie die Projektierung.
120001	Die Kurve wird nicht dargestellt, da eine falsche Achse zur Kurve bzw. eine falsche Kurve projektiert wurde.	Ändern Sie die Projektierung.
120002	Die Kurve wird nicht dargestellt, da die zugeordnete Variable auf eine ungültige Adresse in der Steuerung zugreift.	Überprüfen Sie, ob der Datenbereich für die Variable in der Steuerung existiert, ob die projektierte Adresse richtig ist, oder ob der Wertebereich der Variablen stimmt.

3.8.8.14 130000 - Meldungen Systeminformationen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-13 130000 - Meldungen Systeminformationen

Nummer	Wirkung/Ursachen	Abhilfe
130000	Die Aktion wurde nicht durchgeführt.	Schließen Sie andere Programme. Löschen Sie nicht mehr benötigte Dateien von der Festplatte.
130001	Die Aktion wurde nicht durchgeführt.	Löschen Sie nicht mehr benötigte Dateien von der Festplatte.
130002	Die Aktion wurde nicht durchgeführt.	Schließen Sie andere Programme. Löschen Sie nicht mehr benötigte Dateien von der Festplatte.
130003	Kein Datenträger eingelegt. Der Vorgang wird abgebrochen.	Überprüfen Sie z. B. Folgendes: • Wird auf den richtigen Datenträger zugegriffen? • Ist ein Datenträger eingelegt?
130004	Der Datenträger ist schreibgeschützt. Der Vorgang wird abgebrochen.	Überprüfen Sie, ob der Zugriff auf den richtigen Datenträger erfolgt. Entfernen Sie gegebenenfalls den Schreibschutz.
130005	Die Datei ist schreibgeschützt. Der Vorgang wird abgebrochen.	Überprüfen Sie, ob der Zugriff auf die richtige Datei erfolgt. Ändern Sie gegebenenfalls die Datei-Attribute.
130006	Kein Zugriff auf die Datei möglich. Der Vorgang wird abgebrochen.	Überprüfen Sie z. B. Folgendes: • Wird auf die richtige Datei zugegriffen? • Existiert die Datei? • Verhindert eine andere Aktion den gleichzeitigen Zugriff auf die Datei?
130007	Die Netzwerkverbindung ist unterbrochen. Speichern oder Lesen von Datensätzen über die Netzwerkverbindung nicht möglich.	Überprüfen Sie die Netzwerkverbindung und beseitigen Sie die Störung.
130008	Die Storage Card ist nicht vorhanden. Speichern oder Lesen der genannten Datensätze von Storage Card nicht möglich.	Stecken Sie die Storage Card.
130009	Das angegebene Verzeichnis befindet sich nicht auf der Storage Card. Die Dateien, die in diesem Verzeichnis gespeichert sind, werden nach Ausschalten des Bediengeräts nicht gesichert.	Stecken Sie die Storage Card.

Nummer	Wirkung/Ursachen	Abhilfe
130010	Die maximale Verschachtelungstiefe kann erreicht werden, wenn z.B. in einem Skript über Wertänderung erneut ein anderes Skript aufgerufen wird und in diesem Skript wiederum über Wertänderung ein anderes Skript aufgerufen wird usw. Die projektierte Funktionalität wird nicht angeboten.	Überprüfen Sie die Projektierung.
130013	Die Storage Card ist nicht vorhanden. Speichern oder Lesen der genannten Datensätze von Storage Card nicht möglich.	Stecken Sie die Storage Card.

3.8.8.15 140000 - Meldungen Verbindung: Verbindung + Gerät

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-14 140000 - Meldungen Verbindung: Verbindung + Gerät

Nummer	Wirkung/Ursachen	Abhilfe
140000	Online-Verbindung zur Steuerung wurde erfolgreich aufgebaut.	--
140001	Online-Verbindung zur Steuerung wurde abgebaut.	--
140003	Es erfolgt keine Variablenaktualisierung bzw. Schreiben.	Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist. Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter. Führen Sie einen Neuanlauf durch.
140004	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da der Zugangspunkt oder die Baugruppenparametrierung fehlerhaft ist.	Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist. Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" den Zugangspunkt bzw. die Baugruppenparametrierung (MPI, PPI, PROFIBUS). Führen Sie einen Neuanlauf durch.
140005	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Adresse des Bediengerätes fehlerhaft ist (eventuell zu groß).	Verwenden Sie eine andere Adresse für das Bediengerät. Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist. Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter. Führen Sie einen Neuanlauf durch.
140006	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Baudrate fehlerhaft ist.	Wählen Sie in WinCC eine andere Baudrate (abhängig von Baugruppe, Profil, Kommunikationspartner etc.).

Nummer	Wirkung/Ursachen	Abhilfe
140007	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da das Busprofil fehlerhaft ist (s. %1). Folgende Parameter können nicht in die Registrierungsdatenbank eingetragen werden: 1: Tslot 2: Tqui 3: Tset 4: MinTsdr 5: MaxTsdr 6: Trdy 7: Tid1 8: Tid2 9: Gap Faktor 10: Retry Limit	Überprüfen Sie das benutzerdefinierte Busprofil. Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist. Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter. Führen Sie einen Neuanlauf durch.
140008	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Projektierungsdaten fehlerhaft sind: Folgende Parameter können nicht in die Registrierungsdatenbank eingetragen werden: 0: allgemeiner Fehler 1: falsche Version 2: Profil kann nicht in die Registrierungsdatenbank eingetragen werden. 3: Subnet-Type kann nicht in die Registrierungsdatenbank eingetragen werden. 4: Target Rotation Time kann nicht in die Registrierungsdatenbank eingetragen werden. 5: Höchste Adresse (HSA) fehlerhaft.	Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist. Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter. Führen Sie einen Neuanlauf durch.
140009	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Baugruppe für die S7-Kommunikation nicht gefunden wurde.	Installieren Sie die Baugruppe in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" neu.
140010	Es wurde kein S7-Kommunikationspartner gefunden, da die Steuerung ausgeschaltet ist. DP/T: In der Systemsteuerung ist unter "PG/PC-Schnittstelle einstellen" die Option "PG/PC ist einziger Master am Bus" nicht eingestellt.	Schalten Sie die Steuerung ein. DP/T: Befindet sich nur ein Master am Netz, aktivieren Sie unter "PG/PC-Schnittstelle einstellen" die Option "PG/PC ist einziger Master am Bus". Befinden sich mehr als ein Master am Netz, schalten Sie diese Master ein. Ändern Sie dabei keine Einstellungen, sonst kommt es zu Busstörungen.
140011	Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Kommunikation unterbrochen ist.	Kontrollieren Sie die Verbindung und ob der Kommunikationspartner eingeschaltet ist.
140012	Es liegt ein Initialisierungsproblem vor (z. B. wenn WinCC Runtime im Task-Manager beendet wurde). Oder: ein weiteres Programm (z. B. STEP 7) ist bereits mit anderen Busparametern aktiv und die Treiber können mit den neuen Busparametern (z. B. Baudrate) nicht gestartet werden.	Starten Sie das Bediengerät neu. Oder: Starten Sie zuerst WinCC Runtime und danach weitere Programme.
140013	Das MPI-Kabel steckt nicht und damit fehlt die Stromversorgung.	Überprüfen Sie die Verbindungen.
140014	Projektierte Adresse am Bus bereits belegt.	Ändern Sie in der Projektierung unter Steuerung die Adresse des Bediengerätes.

Nummer	Wirkung/Ursachen	Abhilfe
140015	Falsche Baudrate Oder: Falsche Busparameter (z. B. HSA) Oder: OP-Adresse > HSA oder: falscher Interruptvektor (Interrupt kommt nicht zum Treiber durch)	Korrigieren Sie die falschen Parameter.
140016	Eingestellter Interrupt wird nicht von der Hardware unterstützt.	Ändern Sie die Interruptnummer.
140017	Eingestellter Interrupt wird von einem anderen Treiber genutzt.	Ändern Sie die Interruptnummer.
140018	Der Konsistenzcheck wurde durch SIMOTION Scout deaktiviert. Es erscheint nur ein entsprechender Hinweis.	Aktivieren Sie den Konsistenzcheck mit SIMOTION Scout erneut und laden Sie das Projekt erneut in die Steuerung.
140019	SIMOTION Scout lädt ein neues Projekt in die Steuerung. Die Verbindung zur Steuerung wird abgebrochen.	Warten Sie das Ende der Umkonfiguration ab.
140020	Die Version in der Steuerung und die Version in der Projektierung (FWX-Datei) stimmen nicht überein. Die Verbindung zur Steuerung wird abgebrochen.	Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung: Laden Sie mit SIMOTION Scout die aktuelle Version in die Steuerung. Übersetzen Sie das Projekt mit WinCC ES neu, beenden Sie WinCC Runtime und starten Sie mit neuer Projektierung.
140021	Die Verbindung zur Steuerung wurde nicht aufgebaut. Das "Zugangspasswort" für die Verbindung zur Steuerung wurde falsch projektiert.	Überprüfen Sie im Editor "Verbindungen" im Bereich "Zugangspasswort" das eingegebene Passwort für die Verbindung zur Steuerung. Vergeben Sie das korrekte Passwort. Das "Passwort" für die Verbindung zur Steuerung wird in den Eigenschaften der Steuerung unter "Schutz" vergeben.
140022	Die Verbindung zur Steuerung wurde nicht aufgebaut. Es wurde kein Zugangspasswort für die Verbindung zur Steuerung projektiert.	Geben Sie im Editor "Verbindungen" im Bereich "Zugangspasswort" das Passwort für die Verbindung zur Steuerung ein. Das "Passwort" wird in den Eigenschaften der Steuerung unter "Schutz" vergeben.
140023	Fehler bei der Uhrzeitsynchronisation: Die Systemzeit der PLC %1 konnte nicht gelesen werden.	Prüfen Sie, ob die Steuerung eingeschaltet ist. Überprüfen Sie im Editor "Verbindungen" im Bereich "Zugangspasswort" das eingegebene Passwort für die Verbindung zur Steuerung.
140025	Die Verbindung zur Steuerung wurde nicht aufgebaut. Im Display der Steuerung wurde das Zugangspasswort der Steuerung gesperrt.	Entsperren Sie das Zugangspasswort im Display der Steuerung.

3.8.8.16 160000 - Meldungen Verbindung: OPC: Verbindung

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-15 160000 - Meldungen Verbindung: OPC: Verbindung

Nummer	Wirkung/Ursachen	Abhilfe
160000	Es werden keine Daten mehr geschrieben oder gelesen. Mögliche Ursachen: • Das Kabel ist unterbrochen. • Die Steuerung antwortet nicht, ist defekt, etc. • Der Anschluss findet über die falsche Schnittstelle statt. • Das System ist überlastet.	Überprüfen Sie, ob das Kabel steckt, die Steuerung in Ordnung ist, die richtige Schnittstelle verwendet wird. Starten Sie neu, wenn die Systemmeldung dauerhaft anstehen bleibt.
160001	Die Verbindung besteht wieder, da die Ursache der Unterbrechung beseitigt ist.	--
160010	Es besteht keine Verbindung zum Server, da die Identifikation (CLS-ID) des Servers nicht ermittelt werden kann. Werte können nicht gelesen/geschrieben werden.	Überprüfen Sie die Zugriffsrechte.
160011	Es besteht keine Verbindung zum Server, da die Identifikation (CLS-ID) des Servers nicht ermittelt werden kann. Werte können nicht gelesen/geschrieben werden.	Überprüfen Sie z. B. Folgendes • Stimmt der Servername? • Stimmt der Rechnername? • Ist der Server registriert?
160012	Es besteht keine Verbindung zum Server, da die Identifikation (CLS-ID) des Servers nicht ermittelt werden kann. Werte können nicht gelesen/geschrieben werden.	Überprüfen Sie z. B. Folgendes • Stimmt der Servername? • Stimmt der Rechnername? • Ist der Server registriert? Hinweis für erfahrene Anwender: Interpretieren Sie den Wert von HRESULT.
160013	Der angegebene Server wurde als InProc-Server gestartet. Dies ist nicht freigegeben und kann möglicherweise zu undefiniertem Verhalten führen, da der Server im gleichen Prozessraum läuft wie WinCC Runtime.	Konfigurieren Sie den Server als OutProc-Server oder als Local-Server.
160014	Auf einem PC kann nur ein OPC-Serverprojekt gestartet werden. Beim Versuch, ein zweites Projekt zu starten erscheint eine Fehlermeldung. Das zweite Projekt besitzt keine OPC-Server-Funktionalität und ist von außen nicht als OPC-Server auffindbar.	Starten Sie auf dem Rechner keine zwei Projekte mit OPC-Server-Funktionalität.

3.8.8.17 170000 - Meldungen S7-Dialog

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-16 170000 - Meldungen S7-Dialog

Nummer	Wirkung/Ursachen	Abhilfe
170000	S7-Diagnosemeldungen werden nicht angezeigt, da die Anmeldung an S7-Diagnose an diesem Gerät nicht möglich ist. Der Dienst wird nicht unterstützt.	--
170001	Die Anzeige des S7-Diagnosepuffers ist nicht möglich, da die Kommunikation mit der Steuerung abgeschaltet ist.	Schalten Sie die Steuerung Online.
170002	Die Anzeige des S7-Diagnosepuffers ist nicht möglich, da das Lesen des Diagnosepuffers (SZL) mit Fehler abgebrochen wurde.	--
170003	Die Anzeige einer S7-Diagnosemeldung ist nicht möglich. Es wurde der interne Fehler %2 gemeldet.	--
170004	Die Anzeige einer S7-Diagnosemeldung ist nicht möglich. Es wurde der interne Fehler mit der Fehlerklasse %2 und der Fehlernummer %3 gemeldet.	--
170007	Das Lesen des S7-Diagnosepuffers (SZL) ist nicht möglich, da es mit interner Fehlerklasse %2 und Fehlercode %3 abgebrochen wurde.	--

3.8.8.18 180000 - Meldungen Allgemeines

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-17 180000 - Meldungen Allgemeines

Nummer	Wirkung/Ursachen	Abhilfe
180000	Eine Komponente/OCX erhielt Projektierungsdaten mit einer Versionskennung, die nicht unterstützt wird.	Installieren Sie eine neuere Komponente.
180001	Das System ist überlastet, da zu viele Aktionen gleichzeitig aktiviert wurden. Es können nicht alle Aktionen ausgeführt werden, einige werden verworfen.	Es stehen mehrere verschiedene Abhilfemöglichkeiten zur Verfügung: • Erzeugen Sie Meldungen langsamer (pollen). • Stoßen Sie Skripte und Systemfunktionen in größeren Zeitabständen an. Falls die Meldung häufiger erscheint: Starten Sie das Bediengerät neu.
180002	Die Bildschirmtastatur konnte nicht aktiviert werden. Mögliche Ursache: Die Datei "TouchInputPC.exe" wurde durch fehlerhaft ausgeführtes Setup nicht registriert.	Installieren Sie WinCC Runtime erneut.

Hinweis

Meldungen ohne zusätzliche Information

Meldung 180003 ist nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.19 190000 - Meldungen Variablen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-18 190000 - Meldungen Variablen

Nummer	Wirkung/Ursachen	Abhilfe
190000	Die Variable wird evtl. nicht aktualisiert.	--
190001	Die Variable wird nach einem fehlerhaften Zustand wieder aktualisiert, nachdem der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).	--
190002	Die Variable wird nicht aktualisiert, da die Kommunikation mit der Steuerung unterbrochen ist.	Schalten Sie die Kommunikation über die Systemfunktion "SetOnline" ein.
190004	Die Variable wird nicht aktualisiert, da die projektierte Adresse für diese Variable nicht vorhanden ist.	Überprüfen Sie die Projektierung.

Nummer	Wirkung/Ursachen	Abhilfe
190005	Die Variable wird nicht aktualisiert, da der projektierte Steuerungstyp für diese Variable nicht existiert.	Überprüfen Sie die Projektierung.
190006	Die Variable wird nicht aktualisiert, da eine Abbildung vom Steuerungstyp in den Datentyp der Variablen nicht möglich ist.	Überprüfen Sie die Projektierung.
190007	Der Variablenwert wird nicht geändert, da die Verbindung zur Steuerung unterbrochen oder die Variable offline ist.	Schalten Sie Online bzw. stellen Sie die Verbindung zur Steuerung wieder her.
190008	Die projektierten Grenzwerte der Variablen wurden durch eines der folgenden Ereignisse verletzt: • Werteingabe • Systemfunktion • Skript	Beachten Sie die projektierten oder aktuellen Grenzwerte der Variablen.
190009	Es wurde versucht, der Variablen einen Wert zuzuweisen, der außerhalb des für diesen Datentyp zulässigen Wertebereichs liegt. Z. B. Werteingabe von 260 für eine Byte-Variable oder Werteingabe von -3 für eine vorzeichenlose Wort-Variable.	Beachten Sie den Wertebereich des Datentyps der Variablen.
190010	Die Variable wird zu oft mit Werten beschrieben (z. B. in einer Schleife von einem Skript aus). Es gehen Werte verloren, da maximal 100 Vorgänge zwischengespeichert werden.	Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung: • Erhöhen Sie die Zeit zwischen mehrmaligem Schreiben. • Verwenden Sie bei einer projektierten Quittierung am Bediengerät für "HMI-Quittiervariable" keine Arrayvariable, die länger als 6 Worte ist.
190011	Mögliche Ursache 1: Der eingegebene Wert konnte nicht in die projektierte Steuerungsvariable geschrieben werden, da der Wertebereich über- oder unterschritten wurde. Die Eingabe wurde verworfen und der ursprüngliche Wert wurde wieder hergestellt. Mögliche Ursache 2: Die Verbindung zur Steuerung wurde unterbrochen.	Beachten Sie, dass der eingegebene Wert innerhalb des Wertebereichs der Steuerungsvariablen liegen muss. Kontrollieren Sie die Verbindung zur Steuerung.
190012	Es ist nicht möglich den Wert von einem Quellformat in ein Zielformat zu wandeln, z. B.: Es soll ein Wert außerhalb des gültigen steuerungsabhängigen Wertebereichs für einen Zähler geschrieben werden. Einer Variablen von Typ Integer soll ein Wert vom Typ String zugewiesen werden.	Kontrollieren Sie den Wertebereich oder den Datentyp der Variablen.
190013	Der Benutzer hat einen String eingegeben, dessen Länge größer als die der Variable ist. Der String wird automatisch auf die zulässige Länge gekürzt.	Geben Sie nur Strings ein, welche die Länge der Variable nicht überschreiten.

3.8.8.20 190100 - Meldungen Bereichszeiger

190100 - Meldungen Bereichszeiger

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
190100	Der Bereichszeiger wird nicht aktualisiert, da die projektierte Adresse für diesen Bereichszeiger nicht vorhanden ist. Typ: 1 Warnungen 2 Fehler 3 Quittierung Steuerung 4 Quittierung Bediengerät 5 LED-Abbild 6 Kurvenanforderung 7 Kurvenübertragung 1 8 Kurvenübertragung 2 Nr.: ist die in WinCC ES angezeigte fortlaufende Nummer.	Überprüfen Sie die Projektierung.
190101	Der Bereichszeiger wird nicht aktualisiert, da eine Abbildung vom Steuerungstyp in den Typ des Bereichszeigers nicht möglich ist. Parameter Typ und Nr.: siehe Meldung 190100	--
190102	Der Bereichszeiger wird nach einem fehlerhaften Zustand wieder aktualisiert, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).Parameter Typ und Nr.: Siehe Meldung 190100	--

3.8.8.21 200000 - Meldungen PLC-Koordination

200000 - Meldungen PLC-Koordination

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
200000	Die Koordinierung wird nicht ausgeführt, da die projektierte Adresse in der Steuerung nicht vorhanden/eingerichtet ist.	Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.
200001	Die Koordinierung wird nicht ausgeführt, da an die projektierte Adresse in der Steuerung nicht geschrieben werden kann.	Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung in einem schreibbaren Bereich ein.
200002	Die Koordinierung wird momentan nicht ausgeführt, da das Adressformat des Bereichszeigers nicht zum internen Ablageformat passt.	Interner Fehler

Nummer	Wirkung/Ursachen	Abhilfe
200003	Die Koordinierung wird wieder ausgeführt, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).	--
200004	Die Koordinierung wird evtl. nicht ausgeführt.	--
200005	Es werden keine Daten mehr geschrieben oder gelesen. Mögliche Ursachen: • Das Kabel ist unterbrochen. • Die Steuerung antwortet nicht, ist defekt, etc. • Das System ist überlastet.	Überprüfen Sie, ob das Kabel steckt oder die Steuerung in Ordnung ist. Starten Sie neu, wenn die Systemmeldung weiterhin dauerhaft ansteht.

3.8.8.22 200100 - Meldungen Bereichszeiger Projektkennung

200100 - Meldungen Bereichszeiger Projektkennung

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
200101	Die Projektkennung des Bereichszeigers kann nicht beschrieben werden. Kein Eintrag möglich.	Überprüfen Sie die Projektierung.
200102	Fehler bei der Typkonvertierung.	Überprüfen Sie die Typkonvertierung.
200103	Die Projektkennung des Bereichszeigers wird nach einem fehlerhaften Zustand wieder aktualisiert, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).	--
200104	Unbekannter Fehler	Überprüfen Sie die • Projektierung • HMI-Variablen • HMI-Meldungen • Systemmeldungen • Steuerungsmeldungen Tritt der Fehler wiederholt auf, wenden Sie sich an die Hotline!
200105	Die Online-Verbindung zur Steuerung wurde unterbrochen. Die Verbindung ist gestört.	Überprüfen Sie z. B. Folgendes • Stimmen die Zugriffsrechte? • Stimmt der Servername? • Stimmt der Rechnername? • Ist der Server registriert? • Stimmt die Projektierung?

3.8.8.23 210000 - Meldungen PLC-Job

210000 - Meldungen PLC-Job

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
210000	Aufträge werden nicht bearbeitet, da die projektierte Adresse in der Steuerung nicht vorhanden/eingerichtet ist.	Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.
210001	Aufträge werden nicht bearbeitet, da die projektierte Adresse in der Steuerung nicht lesbar/schreibbar ist.	Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung in einem lesbaren/schreibbaren Bereich ein.
210002	Aufträge werden nicht ausgeführt, da das Adressformat des Bereichszeigers nicht zum internen Ablageformat passt.	Interner Fehler
210003	Das Auftragsfach wird wieder bearbeitet, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).	--
210004	Das Auftragsfach wird evtl. nicht bearbeitet.	--
210005	Es wurde ein Steuerungsauftrag mit einer unzulässigen Nummer angestoßen.	Überprüfen Sie das Steuerungsprogramm.
210006	Während der Ausführung des Steuerungsauftrags trat ein Fehler auf. Der Steuerungsauftrag wird deshalb nicht ausgeführt. Beachten Sie gegebenenfalls auch die nachfolgende / vorhergehende Systemmeldung.	Überprüfen Sie die Parameter des Steuerungsauftrags. Übersetzen Sie die Projektierung neu.

3.8.8.24 220000 - Meldungen WinCC-Kommunikationstreiber

220000 - Meldungen WinCC-Kommunikationstreiber

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
220001	Die Variable wird nicht übertragen, da der unterlagerte Kommunikationstreiber/das Bediengerät beim Schreiben den Datentyp Bool/Bit nicht unterstützt.	Ändern Sie die Projektierung.
220002	Die Variable wird nicht übertragen, da der unterlagerte Kommunikationstreiber/das Bediengerät beim Schreiben den Datentyp Byte nicht unterstützt.	Ändern Sie die Projektierung.
220003	Der Kommunikationstreiber konnte nicht geladen werden. Evtl. ist der Treiber nicht installiert.	Installieren Sie den Treiber indem Sie WinCC Runtime neu installieren.
220004	Die Kommunikation ist unterbrochen, es erfolgt keine Aktualisierung, da das Kabel nicht steckt, oder defekt ist, etc.	Überprüfen Sie die Verbindung.
220005	Die Kommunikation läuft.	--

Nummer	Wirkung/Ursachen	Abhilfe
220006	Die Verbindung zur angegebenen Steuerung an der angegebenen Schnittstelle ist hergestellt.	--
220007	Die Verbindung zur angegebenen Steuerung an der angegebenen Schnittstelle ist unterbrochen.	Überprüfen Sie Folgendes: • Steckt das Kabel? • Ist die Steuerung in Ordnung? • Wird die richtige Schnittstelle verwendet? • Ist ihre Projektierung in Ordnung (Schnittstellenparameter, Protokolleinstellungen, Steuerungsadresse)? Starten Sie neu, wenn die Systemmeldung weiterhin dauerhaft ansteht.
220008	Der Kommunikationstreiber kann nicht auf die angegebene Schnittstelle zugreifen oder diese öffnen. Möglicherweise verwendet bereits ein anderes Programm diese Schnittstelle oder es wird eine nicht am Zielgerät vorhandene Schnittstelle verwendet. Es findet keine Kommunikation mit der Steuerung statt.	Beenden Sie alle Programme, die auf die Schnittstelle zugreifen und starten Sie den Rechner neu. Verwenden Sie eine andere, im System vorhandene, Schnittstelle.
220009	Der Kommunikationspfad ist entweder leer, unvollständig oder in einem ungültigen Format.	Geben Sie das Kommunikationspfad im richtigen Format ein.

3.8.8.25 230000 - Meldungen Bildobjekte

230000 - Meldungen Bildobjekte

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
230000	Der eingegebene Wert konnte nicht übernommen werden. Die Eingabe wird verworfen und der vorherige Wert wird wieder hergestellt. Mögliche Ursache: • Der Wertebereich wurde überschritten • Sie haben unzulässige Zeichen eingegeben • De maximal zulässige Anzahl der Benutzer wurde überschritten.	Geben Sie einen sinnvollen Wert ein oder löschen Sie einen nicht mehr benötigten Benutzer.
230002	Da der angemeldete Benutzer die erforderliche Berechtigung nicht besitzt, wird die Eingabe verworfen und der vorherige Wert wird wiederhergestellt.	Melden Sie sich als Benutzer mit ausreichender Berechtigung an.
230003	Der Wechsel zum angegebenen Bild wird nicht durchgeführt, da das Bild nicht vorhanden/projektiert ist. Das bisherige Bild bleibt angewählt.	Projektieren Sie das Bild und überprüfen Sie die Anwahlfunktion.

Nummer	Wirkung/Ursachen	Abhilfe
230005	Der Wertebereich der Variablen im E/A-Feld wurde überschritten. Der ursprüngliche Wert der Variable bleibt erhalten.	Berücksichtigen Sie bei der Eingabe den Wertebereich der Variablen.
230100	Nach Navigation im Web-Browser wurde eine Meldung zurückgeliefert, die für den Benutzer interessant sein könnte. Der Web-Browser läuft weiter, zeigt die neue Seite aber evtl. nicht (vollständig) an.	Auf eine andere Seite navigieren.
230200	Die Verbindung des HTTP-Kanals wurde unterbrochen, weil ein Fehler aufgetreten ist. Dieser Fehler wird durch eine weitere Systemmeldung näher erläutert. Es werden keine Daten mehr ausgetauscht.	Überprüfen Sie die Netzwerkverbindung. Überprüfen Sie die Konfiguration des Servers.
230201	Die Verbindung des HTTP-Kanals wurde hergestellt. Es werden Daten ausgetauscht.	--
230202	Die WININET.DLL hat einen Fehler festgestellt. Dieser Fehler tritt zumeist auf wenn keine Verbindung zum Server möglich ist, oder der Server eine Verbindung ablehnt, weil der Client sich nicht richtig autorisiert hat. Bei einer Verschlüsselten Verbindung über SSL kann auch ein nicht akzeptiertes Server-Zertifikat die Ursache sein. Nähere Erläuterung liefert der Fehlertext in der Meldung. Dieser Text ist immer in der Sprache der Windows-Installation, da er von Windows geliefert wird. Es werden keine Prozesswerte ausgetauscht. Möglicherweise wird der Teil der Meldung nicht angezeigt, der vom Betriebssystem Windows geliefert wird, z. B. "Ein Fehler ist aufgetreten. Die WININET.DLL liefert folgenden Fehler: Nummer: 12055 Text:HTTP: <no error text available>."	Abhängig von der Ursache: Wenn die Verbindung nicht hergestellt werden kann oder ein Timeout auftritt: • Netzwerkverbindung und Netzwerk überprüfen. • Server-Adresse überprüfen. • Überprüfen, ob der WebServer auf dem Zielrechner tatsächlich läuft. Bei fehlerhafter Autorisierung: • Projektierter Benutzername und/oder Passwort stimmen nicht mit denen des Servers überein. Übereinstimmung herstellen. Bei nicht akzeptierten Server-Zertifikat: Zertifikat von unbekannten CA ( ) signiert: • Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat, das mit einem dem Clientrechner bekannten Root-Zertifikat signiert wurde. Bei ungültigen Datum des Zertifikats: • Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat mit gültigem Datum auf dem Server. Bei ungültigen CN (Common Name oder Computer Name): • Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat mit einem der Server-Adresse entsprechenden Namen.

Nummer	Wirkung/Ursachen	Abhilfe
230203	Obwohl eine Verbindung zum Server möglich ist, hat der HTTP-Server die Verbindung abgelehnt. Mögliche Ursachen: • WinCC Runtime läuft nicht auf dem Server nicht läuft • Der HTTP-Kanal wird nicht unterstützt (503 Service unavailable). Andere Fehler können nur auftreten, wenn der Webserver den HTTP-Kanal nicht unterstützt. Die Sprache des Fehlertext hängt vom Webserver ab. Es werden keine Daten ausgetauscht.	Bei Fehler 503 Service unavailable: Überprüfen Sie, ob WinCC Runtime auf dem Server läuft und ob der HTTP-Kanal unterstützt wird.
230301	Ein interner Fehler ist aufgetreten. Ein englischer Text erläutert den Fehler in der Meldung noch etwas genauer. Eine mögliche Ursache kann z.B. ungenügender Speicher sein. Das OCX funktioniert nicht.	--
230302	Der Name des Remote-Servers kann nicht aufgelöst werden. Es kann keine Verbindung hergestellt werden.	Überprüfen Sie die projektierte Server-Adresse. Überprüfen Sie, ob der DNS-Service des Netzwerkes aktiv ist.
230303	Der Remote-Server ist auf dem adressierten Rechner nicht aktiv. Die Server-Adresse ist falsch. Es kann keine Verbindung hergestellt werden.	Überprüfen Sie die projektierte Server-Adresse. Überprüfen Sie, ob der Remote-Server des Zielrechners läuft.
230304	Der Remote-Server des adressierten Rechners ist inkompatibel zum VNCOCX. Es kann keine Verbindung hergestellt werden.	Verwenden Sie einen kompatiblen Remote-Server.
230305	Die Authentifizierung ist wegen falschem Kennwort gescheitert. Es kann keine Verbindung hergestellt werden.	Projektieren Sie das richtige Kennwort.
230306	Die Verbindung zum Remote-Server ist gestört. Dies kann bei Netzproblemen auftreten. Es kann keine Verbindung hergestellt werden.	Überprüfen Sie, ob das Netzwerkkabel steckt oder Netzprobleme vorliegen.
230307	Die Verbindung wurde vom Remote-Server beendet. Mögliche Ursachen: • Der Remote-Server wurde beendet • Der Benutzer hat den Server angewiesen hat, alle Verbindungen zu beenden. Die Verbindung wird abgebrochen.	--
230308	Diese Meldung informiert Sie über den Verbindungsaufbau. Eine Verbindung wird gerade aufgebaut.	--

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 230400-230409 sowie 230500-230501 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.26 240000 - Meldungen Autorisierung

240000 - Meldungen Autorisierung

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
240000	WinCC Runtime läuft im Demomodus. Sie haben keine oder eine defekte Autorisierung.	Spielen Sie die Autorisierung ein.
240001	WinCC Runtime läuft im Demomodus. Es sind zu viele Variablen für die installierte Version projektiert.	Spielen Sie eine ausreichende Autorisierung/Powerpack ein.
240002	WinCC Runtime läuft mit zeitlich begrenzter Not-Autorisierung.	Stellen Sie die Vollautorisierung wieder her.
240004	Fehler beim Lesen der Not-Autorisierung. WinCC Runtime läuft im Demo-Modus.	Starten Sie WinCC Runtime neu, installieren Sie die Autorisierung oder reparieren Sie die Autorisierung (siehe Inbetriebnahmeanleitung Softwareschutz).
240005	Der Automation License Manager hat eine interne Systemstörung erkannt. Mögliche Ursachen: • Eine zerstörte Datei • defekte Installation • kein freier Speicher für den Automation License Manager o.ä.	Starten sie das Bediengerät/PC neu. Führt dies zu keinem Erfolg, deinstallieren Sie den Automation License Manager und installieren Sie ihn neu.

3.8.8.27 250000 - Meldungen S7-Status/Steuern

250000 Meldungen - S7-Status/Steuern

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
250000	Die in der angegebenen Zeile in "Status/Steuern" eingestellte Variable wird nicht aktualisiert, da die projektierte Adresse für diese Variable nicht vorhanden ist.	Überprüfen Sie die eingestellte Adresse und kontrollieren Sie, ob die Adresse in der Steuerung eingerichtet ist.
250001	Die in der angegebenen Zeile in "Status/Steuern" eingestellte Variable wird nicht aktualisiert, da der projektierte Steuerungstyp für diese Variable nicht existiert.	Überprüfen Sie die eingestellte Adresse.
250002	Die in der angegebenen Zeile in "Status/Steuern" eingestellte Variable wird nicht aktualisiert, da eine Abbildung vom Steuerungstyp in den Variablentyp nicht möglich ist.	Überprüfen Sie die eingestellte Adresse.
250003	Es konnte keine Verbindung zur Steuerung hergestellt werden. Die Variablen werden nicht aktualisiert.	Kontrollieren Sie die Verbindung zur Steuerung. Prüfen Sie, ob die Steuerung eingeschaltet und Online ist.

3.8.8.28 260000 - Meldungen Passwortsystem

260000 - Meldungen Passwortsystem

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
260000	Es wurde ein unbekannter Benutzer oder ein unbekanntes Kennwort am System eingegeben. Der aktuelle Benutzer wird vom System abgemeldet.	Melden Sie sich als Benutzer mit gültigem Kennwort am System an.
260001	Der angemeldete Benutzer hat nicht die ausreichende Berechtigung, um die geschützte Funktionalität auszuführen.	Melden Sie sich als Benutzer am System an, der die notwendige Berechtigung besitzt.
260002	Diese Meldung wird bei Auslösung der Systemfunktion "VerfolgeBenutzeraenderung" ausgegeben.	--
260003	Der Benutzer hat sich am System abgemeldet.	--
260004	Der in der Benutzeranzeige neu eingegebene Benutzername ist bereits in der Benutzerverwaltung vorhanden.	Wählen Sie einen anderen Benutzernamen, da Benutzernamen in der Benutzerverwaltung eindeutig sein müssen.
260005	Eingabe wird verworfen.	Kürzeren Benutzernamen eingeben.
260006	Eingabe wird verworfen.	Kürzeres bzw. längeres Kennwort eingeben.
260007	Die eingegebene Abmeldezeit liegt außerhalb des gültigen Bereichs von 0 bis 60 Minuten. Der eingegebene Wert wird verworfen und der ursprüngliche Wert bleibt bestehen.	Geben Sie einen Wert zwischen 0 und 60 Minuten für die Abmeldezeit ein.
260008	Es wurde versucht, eine mit ProTool V 6.0 erstellte Datei PTProRun.pwl, in WinCC zu lesen. Das Lesen der Datei wurde wegen Inkompatibilität des Formats abgebrochen.	--
260009	Sie haben versucht einen der Benutzer "Administrator" oder "PLC User" zu löschen. Diese beiden Benutzer sind fester Bestandteil der Benutzerverwaltung und dürfen nicht gelöscht werden.	Falls es nötig ist, einen Benutzer zu löschen, z.B. wenn maximale Anzahl der Benutzer erreicht ist, löschen sie einen anderen Benutzer.
260012	Das im Dialog "Kennwort ändern" eingegebenen Kennwort und die Bestätigung stimmen nicht überein. Das Kennwort wird nicht geändert. Der aktuell angemeldete Benutzer wird abgemeldet.	Sie müssen sich erneut am System anmelden. Anschließend geben Sie zwei identische Kennwörter ein, um das Kennwort zu ändern.
260013	Das im Dialog "Kennwort ändern" eingegebene Kennwort wurde bereits verwendet und ist daher nicht gültig. Das Kennwort wird nicht geändert. Der aktuell angemeldete Benutzer wird abgemeldet.	Sie müssen sich erneut am System anmelden. Anschließend geben Sie ein neues Kennwort ein, das bisher noch nicht verwendet wurde.
260014	Sie haben dreimal in Folge versucht sich mit einem falschen Kennwort anzumelden. Sie werden gesperrt und der Gruppe Nr. 0 zugewiesen.	Sie können sich mit Ihrem richtigen Kennwort am System anmelden. Die Gruppenzuordnung kann nur ein Administrator ändern.

Nummer	Wirkung/Ursachen	Abhilfe
260024	Das eingegebene Kennwort entspricht nicht den erforderlichen Sicherheitsrichtlinien.	Geben Sie ein Kennwort ein, das aus mindestens einer Zahl besteht.
260025	Das eingegebene Kennwort entspricht nicht den erforderlichen Sicherheitsrichtlinien.	Geben Sie ein Kennwort ein, das mindestens drei Zeichen beinhaltet.
260026	Das eingegebene Kennwort entspricht nicht den erforderlichen Sicherheitsrichtlinien.	Geben Sie ein Kennwort ein, das mindestens ein Sonderzeichen beinhaltet.
260028	Es wurde versucht beim Systemstart, Anmeldeversuch oder beim Versuch das Passwort eines SIMATIC Logon Benutzers zu ändern, auf den SIMATIC Logon Server zuzugreifen. Beim Anmeldeversuch wird der neue Benutzer nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.	Überprüfen Sie die Verbindung zum SIMATIC Logon Server und dessen Konfiguration, z. B.: 1. Portnummer 2. IP-Adresse 3. Servername 4. Funktionsfähiges Transferkabel. Oder verwenden Sie einen lokalen Benutzer.
260030	Der SIMATIC Logon Benutzer konnte sein Kennwort nicht auf dem SIMATIC Logon Server ändern. Möglicherweise entspricht das neue Kennwort nicht den Kennwortregeln am Server oder der Benutzer hat nicht das Recht, sein Kennwort zu ändern. Das alte Kennwort bleibt und der Benutzer wird abgemeldet.	Melden Sie sich erneut mit einem anderen Kennwort an. Überprüfen Sie die Kennwortregeln am SIMATIC Logon Server.
260033	Die Aktion Kennwort ändern oder Benutzer anmelden konnte nicht durchgeführt werden.	Überprüfen Sie die Verbindung zum SIMATIC Logon Server und dessen Konfiguration, z. B.: 1. Portnummer 2. IP-Adresse 3. Servername 4. Funktionsfähiges Transferkabel Oder verwenden Sie einen lokalen Benutzer.
260034	Der letzte Anmeldevorgang wurde noch nicht abgeschlossen. Eine Benutzeraktion oder ein Anmeldedialog kann daher nicht aufgerufen werden. Der Anmeldedialog wird nicht aufgeblendet. Die Benutzeraktion wird nicht ausgeführt.	Warten Sie, bis der Anmeldevorgang abgeschlossen ist.
260035	Der letzte Versuch, dass Kennwort zu ändern, wurde noch nicht abgeschlossen. Eine Benutzeraktion oder ein Anmeldedialog kann daher nicht aufgerufen werden. Der Anmeldedialog wird nicht aufgeblendet. Die Benutzeraktion wird nicht ausgeführt.	Warten Sie, bis der Vorgang abgeschlossen ist.
260036	Auf dem SIMATIC Logon Server fehlen Lizenzen. Die Anmeldung wird noch zugelassen.	Kontrollieren Sie die Lizensierung auf dem SIMATIC Logon Server.
260037	Auf dem SIMATIC Logon Server ist keine Lizenz verfügbar. Eine Anmeldung ist nicht möglich. Es ist keine Anmeldung über den SIMATIC Logon Server möglich, sondern nur noch über lokale Benutzer.	Kontrollieren Sie die Lizensierung auf dem SIMATIC Logon Server.

Nummer	Wirkung/Ursachen	Abhilfe
260040	Es wurde versucht beim Anmeldeversuch oder beim Versuch das Passwort eines SIMATIC Logon Benutzers zu ändern, auf den SIMATIC Logon Server zuzugreifen. Beim Anmeldeversuch wird der neue Benutzer nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.	Überprüfen Sie die Verbindung zu der Domäne und deren Konfiguration in Editor Runtime-Sicherheitseinstellungen. Oder verwenden Sie einen lokalen Benutzer.
260043	Es war nicht möglich den Benutzer am SIMATIC Logon Server anzumelden. Möglicherweise war der Benutzername oder das Kennwort falsch oder dem Benutzer fehlen die Rechte, sich anzumelden. Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.	Versuchen Sie es erneut. Gegebenenfalls prüfen Sie die Benutzerdaten am SIMATIC Logon Server.
260044	Es war nicht möglich den Benutzer am SIMATIC Logon Server anzumelden, da sein Account gesperrt ist. Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.	Überprüfen Sie die Benutzerdaten am SIMATIC Logon Server.
260045	Der SIMATIC Logon Benutzer gehört keiner oder mehreren Gruppen an. Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.	Überprüfen Sie die Benutzerdaten am SIMATIC Logon Server und die Konfiguration in Ihrem WinCC Projekt. Ein Benutzer darf nur einer Gruppe zugeordnet sein.

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 260010, 260011, 260015-260023, 260029, 260031, 260032, 260038, 260039, 260041, 260042 und 260048 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.29 270000 - Meldungen Meldungen System

270000 - Meldungen Meldungen System

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
270000	In der Meldung wird eine Variable nicht dargestellt, da sie auf eine ungültige Adresse in der Steuerung zugreift.	Überprüfen Sie, ob der Datenbereich für die Variable in der Steuerung existiert, ob die projektierte Adresse richtig ist, ob der Wertebereich der Variablen stimmt.
270001	Es gibt eine geräteabhängige Anzahl von Meldungen, die maximal gleichzeitig anstehen dürfen, um angezeigt werden zu können (siehe Betriebsanleitungen). Diese Anzahl ist überschritten. Die Anzeige enthält nicht mehr alle Meldungen. Im Meldepuffer werden jedoch alle Meldungen eingetragen.	--
270002	Es werden Meldungen aus einem Archiv angezeigt, zu denen es im aktuellen Projekt keine Daten gibt. Für die Meldungen werden Platzhalter ausgegeben.	Löschen Sie gegebenenfalls alte Archivdaten.
270003	Der Dienst kann nicht eingerichtet werden, da zu viele Geräte diesen Dienst einrichten wollen. Maximal können vier Geräte diese Aktion ausführen.	Schließen Sie weniger Bediengeräte an, die den Dienst nutzen sollen.
270004	Zugriff auf persistenten Meldepuffer nicht möglich. Meldungen können nicht wiederhergestellt und auch nicht gesichert werden.	Wenn beim nächsten Wiederanlauf erneut Probleme auftreten, wenden Sie sich bitte an den Customer Support (Flash löschen).
270005	Persistenter Meldepuffer beschädigt: Meldungen können nicht wiederhergestellt werden.	Wenn beim nächsten Wiederanlauf erneut Probleme auftreten, wenden Sie sich bitte an den Customer Support (Flash löschen).
270006	Projekt geändert: Meldungen können aus dem persistenten Meldepuffer nicht wiederhergestellt werden.	Projekt wurde übersetzt und neu auf das Bediengerät geladen; Der Fehler sollte beim nächsten Start des Bediengeräts nicht mehr auftreten.
270007	Ein Konfigurationsproblem verhindert die Wiederherstellung (z.B. DLL gelöscht, Verzeichnis umbenannt o.ä.).	Bitte aktualisieren Sie das Betriebssystem und laden Sie Ihr Projekt neu auf das Bediengerät.

3.8.8.30 280000 - Meldungen SIMATIC 505 DP Communication Driver (CSP)

Meldungen SIMATIC 505 DP Communication Driver (CSP)

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
280000	Die Verbindung zwischen Steuerung und Port ist aktiv.	--
280001	Die Verbindung zur Steuerung ist am Port unterbrochen.	Überprüfen Sie Folgendes: • Ist das Kabel eingesteckt? • Ist die Steuerung aktiv? • Wird der richtige Port benutzt? • Stimmt die Konfiguration?
280002	Die Parameter der Steuerung sind korrekt, die XSUB- und .REC-Dateien sind vorhanden.	--
280003	• Die Parameter der Steuerung sind korrekt, die XSUB- und .REC-Dateien sind nicht vorhanden. • Die Steuerung ist im STOP-Modus.	• Laden Sie die XSUB- und .REC-Dateien in die Steuerung. • Überprüfen Sie, ob die Steuerung sich im STOP-Modus befindet.
280004	• Das Kommunikationskabel zwischen Bediengerät und Steuerung ist getrennt. • Die Steuerung ist im STOP-Modus.	Überprüfen Sie Folgendes: • Das Kommunikationskabel zwischen Bediengerät und Steuerung • Modus der Steuerung (Run/STOP)

3.8.8.31 290000 - Meldungen Rezeptursystem

290000 - Meldungen Rezeptursystem

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
290000	Die Rezepturvariablese konnte nicht gelesen oder geschrieben werden. Sie wird mit dem Startwert belegt. Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Variablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290003 ausgegeben.	Überprüfen Sie in der Projektierung, ob die Adresse in der Steuerung eingerichtet ist.
290001	Es wurde versucht, der Rezepturvariablen einen Wert zuzuweisen, der außerhalb des Wertebereichs liegt, der für diesen Typ zulässig ist. Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Variablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290004 ausgegeben.	Beachten Sie den Wertebereich des Variablentyps.

Nummer	Wirkung/Ursachen	Abhilfe
290002	Es ist nicht möglich, den Wert von einem Quellformat in ein Zielformat zu wandeln. Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Rezepturvariablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290005 ausgegeben.	Kontrollieren Sie den Wertebereich oder den Typ der Variablen.
290003	Diese Meldung werden ausgegeben, wenn die Meldung Nr. 290000 mehr als fünfmal ausgelöst wurde. In diesem Fall wird keine Einzelmeldungen mehr erzeugt.	Überprüfen Sie in der Projektierung, ob die Adressen der Variablen in der Steuerung eingerichtet sind.
290004	Diese Meldung wird ausgegeben, wenn die Meldung Nr. 290001 mehr als fünfmal ausgelöst wurde. In diesem Fall werden keine Einzelmeldungen mehr erzeugt.	Beachten Sie den Wertebereich des Variablentyps.
290005	Diese Meldung wird ausgegeben, wenn die Meldung Nr. 290002 mehr als fünfmal ausgelöst wurde. In diesem Fall werden keine Einzelmeldungen mehr erzeugt.	Kontrollieren Sie den Wertebereich oder den Typ der Variablen.
290006	Die projektierten Grenzwerte der Variablen wurden durch Werteingabe verletzt.	Beachten Sie die projektierten oder aktuellen Grenzwerte der Variablen.
290007	Es besteht ein Unterschied zwischen Quell- und Zielstruktur in der momentan bearbeiteten Rezeptur. Die Quellstruktur enthält eine zusätzliche Rezepturvariable, die in der Zielstruktur nicht vorhanden ist und deshalb nicht zugeordnet werden kann. Der Wert wird verworfen.	Entfernen Sie aus Ihrer Projektierung die angegebene Rezepturvariable in der angegebenen Rezeptur.
290008	Es besteht ein Unterschied zwischen Quell- und Zielstruktur in der momentan bearbeiteten Rezeptur. Die Zielstruktur enthält eine zusätzliche Rezepturvariable, die in der Quellstruktur nicht vorhanden ist. Die angegebene Rezepturvariable wird mit ihrem Startwert belegt.	Fügen Sie die angegebene Rezepturvariable in der Quellstruktur ein.
290010	Der für die Rezeptur projektierte Ablageort ist nicht zulässig. Mögliche Ursachen: Unzulässige Zeichen, Schreibschutz, Datenträger voll oder nicht vorhanden.	Überprüfen Sie den projektierten Ablageort.
290011	Der Datensatz mit der angegebenen Nummer existiert nicht.	Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert).
290012	Die Rezeptur mit der angegebenen Nummer existiert nicht.	Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert).

Nummer	Wirkung/Ursachen	Abhilfe
290013	Es wurde versucht, einen Datensatz unter einer bereits vorhandenen Datensatznummer abzuspeichern. Der Vorgang wird nicht ausgeführt.	Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung: • Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert). • Löschen Sie vorher den Datensatz. • Ändern Sie den Funktionsparameter "Überschreiben".
290014	Die angegebene zu importierende Datei konnte nicht gefunden werden.	Überprüfen Sie Folgendes: • Überprüfen Sie den Dateinamen. • Vergewissern Sie sich, dass die Datei im angegebenen Verzeichnis liegt.
290020	Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung gestartet wurde.	--
290021	Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung fehlerfrei beendet wurde.	--
290022	Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung mit Fehler abgebrochen wurde.	Überprüfen Sie in der Projektierung Folgendes: • Sind die Adressen der Variablen in der Steuerung eingerichtet? • Existiert die Rezepturnummer? • Existiert die Datensatznummer? • Ist der Funktionsparameter "Überschreiben" eingestellt?
290023	Rückmeldung, dass die Übertragung von Datensätzen von der Steuerung zum Bediengerät gestartet wurde.	--
290024	Rückmeldung, dass die Datensätze von der Steuerung zum Bediengerät fehlerfrei übertragen wurden.	---
290025	Rückmeldung, dass die Übertragung von Datensätzen von der Steuerung zum Bediengerät mit Fehler abgebrochen wurde.	Überprüfen Sie in der Projektierung Folgendes: • Sind die Adressen der Variablen in der Steuerung eingerichtet? • Existiert die Rezepturnummer? • Existiert die Datensatznummer? • Ist der Funktionsparameter "Überschreiben" eingestellt?
290026	Es wird versucht, einen Datensatz zu lesen/schreiben, obwohl das Datenfach momentan nicht frei ist. Dieser Fehler kann bei Rezepturen auftreten, für die eine Übertragung mit Synchronisation projektiert wurde.	Setzen Sie im Datenfach den Status auf Null.

Nummer	Wirkung/Ursachen	Abhilfe
290027	Momentan kann keine Verbindung zur Steuerung hergestellt werden. Deshalb kann der Datensatz nicht gelesen oder geschrieben werden. Mögliche Ursachen: Keine physikalische Verbindung zur Steuerung (kein Kabel gesteckt, Kabel defekt) oder Steuerung ausgeschaltet.	Überprüfen Sie die Verbindung zur Steuerung.
290030	Diese Meldung wird nach Wiederanwahl eines Bildes ausgegeben, das eine Rezepturanzeige enthält, in der bereits ein Datensatz ausgewählt ist.	Laden Sie den auf dem Ablageort vorhanden Datensatz erneut oder behalten Sie die aktuellen Werte bei.
290031	Beim Speichern wurde erkannt, dass bereits ein Datensatz mit der angegebenen Nummer existiert.	Überschreiben Sie den Datensatz oder brechen Sie den Vorgang ab.
290032	Beim Exportieren von Datensätzen wurde erkannt, dass bereits eine Datei mit dem angegebenen Namen existiert.	Überschreiben Sie die Datei oder brechen Sie den Vorgang ab.
290033	Sicherheitsabfrage vor dem Löschen von Datensätzen.	--
290040	Ein nicht näher zu spezifizierender Datensatzfehler mit dem Fehlercode %1 ist aufgetreten. Die Aktion wird abgebrochen. Möglicherweise ist in der Steuerung das Datenfach nicht korrekt eingerichtet.	Überprüfen Sie den Ablageort, den Datensatz, den Bereichszeiger "Datensatz" und gegebenenfalls die Verbindung zur Steuerung. Stoßen Sie nach einer kurzen Wartezeit die Aktion erneut an. Tritt der Fehler weiterhin auf, so wenden Sie sich bitte an den Customer Support. Geben Sie dabei den aufgetretenen Fehlercode an.
290041	Das Abspeichern eines Datensatzes oder einer Datei ist nicht möglich, da der Ablageort voll ist.	Löschen Sie nicht mehr benötigte Dateien.
290042	Es wurde versucht, gleichzeitig mehrere Rezepturaktionen auszuführen. Die letzte Aktion wird nicht ausgeführt.	Stoßen Sie nach einer kurzen Wartezeit die Aktion erneut an.
290043	Sicherheitsabfrage vor dem Speichern von Datensätzen.	--
290044	Die Datenablage für die Rezeptur ist zerstört und wird gelöscht.	--
290050	Rückmeldung, dass der Export von Datensätzen gestartet wurde.	--
290051	Rückmeldung, dass der Export von Datensätzen fehlerfrei beendet wurde.	--
290052	Rückmeldung, dass der Export von Datensätzen mit Fehler abgebrochen wurde.	Stellen Sie sicher, dass die Struktur der Datensätze im Ablageort und die aktuelle Rezeptur-Struktur am Bediengerät identisch sind.
290053	Rückmeldung, dass der Import von Datensätzen gestartet wurde.	--
290054	Rückmeldung, dass der Import von Datensätzen fehlerfrei beendet wurde.	--
290055	Rückmeldung, dass der Import von Datensätzen mit Fehler abgebrochen wurde.	Stellen Sie sicher, dass die Struktur der Datensätze im Ablageort und die aktuelle Rezeptur-Struktur am Bediengerät identisch sind.

Nummer	Wirkung/Ursachen	Abhilfe
290056	Der Wert in der angegebenen Zeile/Spalte konnte nicht fehlerfrei gelesen/geschrieben werden. Die Aktion wird abgebrochen.	Überprüfen Sie die angegebene Zeile/Spalte.
290057	Die Rezepturvariablen der angegebenen Rezeptur wurden von der Betriebsart "Offline" nach "Online" umgeschaltet. Jede Änderung einer Variablen dieser Rezeptur wird nun sofort in die Steuerung übertragen.	--
290058	Die Rezepturvariablen der angegebenen Rezeptur wurden von der Betriebsart "Online" nach "Offline" umgeschaltet. Änderungen von Variablen dieser Rezeptur werden nicht mehr sofort in die Steuerung übertragen, sondern müssen gegebenenfalls über eine Datensatzübertragung explizit in die Steuerung übertragen werden.	--
290059	Rückmeldung, dass der angegebene Datensatz erfolgreich gespeichert wurde.	--
290060	Rückmeldung, dass der Datensatzspeicher erfolgreich gelöscht wurde.	--
290061	Rückmeldung, dass das Löschen des Datensatzspeichers mit Fehler abgebrochen wurde.	--
290062	Die max. Datensatznummer liegt über 65536. Dieser Datensatz kann nicht angelegt werden.	Wählen Sie eine andere Nummer.
290063	Tritt auf bei Systemfunktion "ExportiereDatensaetze" mit Parameter "Überschreiben" auf "Nein". Es wurde versucht eine Rezeptur unter einem Dateinamen zu speichern, der bereits vorhanden ist. Exportieren wird abgebrochen.	Überprüfen Sie die Parameter der Systemfunktion "ExportiereDatensaetze".
290064	Rückmeldung, dass das Löschen von Datensätzen gestartet wurde.	--
290065	Rückmeldung, dass das Löschen von Datensätzen fehlerfrei beendet wurde.	--
290066	Sicherheitsabfrage vor dem Löschen von Datensätzen.	--
290068	Sicherheitsabfrage, ob wirklich die ausgewählten Datensätze der Rezeptur gelöscht werden sollen.	--
290069	Sicherheitsabfrage, ob wirklich alle Datensätze der Rezeptur gelöscht werden sollen.	--
290070	Der spezifizierte Datensatz ist in der Import-Datei nicht vorhanden.	Überprüfen sie die Quelle der Datensatznummer oder des Datensatznamens (Konstante oder Variablenwert)
290071	Beim Editieren von Datensatzwerten wurde ein Wert eingegeben, der den unteren Grenzwert der Rezepturvariable unterschreitet. Die Eingabe wird verworfen.	Geben Sie einen Wert innerhalb der Grenzwerte der Rezepturvariable ein.

Nummer	Wirkung/Ursachen	Abhilfe
290072	Beim Editieren von Datensatzwerten wurde ein Wert eingegeben, der den oberen Grenzwert der Rezepturvariable überschreitet. Die Eingabe wird verworfen.	Geben Sie einen Wert innerhalb der Grenzwerte der Rezepturvariable ein.
290073	Eine Aktion (z.B. Speichern eines Datensatzes) konnte aus unbekannten Gründen nicht durchgeführt werden. Der Fehler entspricht der Statusmeldung IDS_OUT_CMD_EXE_ERR der großen Rezeptur-Anzeige.	--
290074	Beim Speichern wurde erkannt, dass bereits ein Datensatz mit der angegebenen Nummer, aber unter einem anderen Namen existiert.	Überschreiben Sie den Datensatz, ändern Sie die Datensatznummer oder brechen Sie den Vorgang ab.
290075	Ein Datensatz mit diesem Namen ist bereits vorhanden. Das Speichern des Datensatzes wird abgebrochen.	Bitte wählen Sie einen anderen Datensatznamen.
290110	Setzen der Standardwerte wegen Fehler abgebrochen.	--
290111	Das Subsystem Rezepturen kann nicht benutzt werden. Rezepturviews haben keinen Inhalt und rezepturbezogene Funktionen werden nicht ausgeführt. Mögliche Ursachen: • Beim Laden der Rezepturen ist ein Fehler aufgetreten. • In ES wurden die Rezepturen strukturell verändert. Beim erneuten Download des Projekts wurden die Rezepturen nicht mit übertragen. Somit passen die neuen Projektierungsdaten nicht mehr zu den alten Rezepturen im Gerät.	Laden Sie erneut das Projekt inklusive Rezepturen (entsprechende Checkbox im Ladedialog muss aktiviert sein) auf das Gerät.

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 290067 und 290076-290109 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.32 300000 - Meldungen Alarm_S

300000 - Meldungen Alarm_S

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
300000	Die Prozessüberwachung (z. B. mit PDiag oder S7-Graph) ist falsch programmiert: Es stehen mehr Meldungen gleichzeitig an, als in den technischen Daten der CPU angegeben ist. Weitere ALARM_S-Meldungen können nicht mehr von der Steuerung verwaltet und an Bediengeräte gemeldet werden.	Ändern Sie die Projektierung der Steuerung.
300001	Die Anmeldung für ALARM_S an dieser Steuerung wird nicht ausgeführt.	Wählen Sie eine Steuerung, die den Dienst ALARM_S unterstützt.

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 300100-300102 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.33 310000 - Meldungen Protokollierung

310000 - Meldungen Protokollierung

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
310000	Es sollen zu viele Protokolle gleichzeitig gedruckt werden. Da nur ein Protokolldruck gleichzeitig zulässig ist, wird der Druckauftrag abgelehnt.	Warten Sie, bis der Druck des letzen aktiven Protokolls beendet wurde. Wiederholen Sie gegebenenfalls den Druckauftrag.
310001	Beim Ansprechen des Druckers ist ein Fehler aufgetreten. Das Protokoll wird nicht oder fehlerhaft gedruckt.	Werten Sie die im Zusammenhang mit dieser Meldung ausgegebenen zusätzlichen Systemmeldungen aus. Wiederholen Sie gegebenenfalls den Druckauftrag

3.8.8.34 330000 - Meldungen Interaktion

330000 - Meldungen Interaktion

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
330022	Auf dem Bediengerät sind zu viele Dialoge geöffnet.	Schließen Sie nicht benötigte Dialoge auf dem Bediengerät.

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 330001-330021, 330023-330070 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.35 340000 - Meldungen Infotext

340000 - Meldungen Infotext

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 340001-340005 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.36 350000 - Meldungen PROFIsafe

350000 - Meldungen PROFIsafe

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Nummer	Wirkung/Ursachen	Abhilfe
350000	PROFIsafe Pakete sind innerhalb der notwendigen Zeit nicht angekommen. Es liegt ein Kommunikationsproblem mit der F-CPU vor. RT wird beendet.	Überprüfen Sie die WLAN-Verbindung.
350001	PROFIsafe Pakete sind innerhalb der notwendigen Zeit nicht angekommen. Es liegt ein Kommunikationsproblem mit der F-CPU vor. PROFIsafe Verbindung wird neu aufgebaut.	Überprüfen Sie die WLAN-Verbindung.
350002	Ein interner Fehler ist aufgetreten. Runtime wird beendet.	Interner Fehler

Nummer	Wirkung/Ursachen	Abhilfe
350003	Rückmeldung für den Aufbau der Verbindung zur F-CPU. Ab sofort sind die Not-Halt-Taster aktiv.	--
350004	Die PROFIsafe-Kommunikation wurde eingestellt und die Verbindung abgebaut. Die Runtime kann beendet werden. Ab sofort sind die Not-Halt-Taster nicht mehr aktiv.	--
350005	Die projektierte Adresse des F-Gerätes ist falsch. Es kann keine PROFIsafe-Verbindung aufgebaut werden.	Überprüfen und ändern Sie die Adresse des F-Gerätes im WinCC ES.
350006	Das Projekt wurde gestartet. Bei Projektstart müssen die Zustimmtaster auf Funktionsfähigkeit überprüft werden.	Betätigen Sie beide Zustimmtaster nacheinander in den Stellungen "Zustimmung" und "Panik".
350008	Es wurde eine falsche Anzahl von Failsafe Tasten projektiert. Es kann keine PROFISafe-Verbindung aufgebaut werden.	Ändern Sie die Anzahl der Failsafe-Tasten im Projekt.
350009	Das Gerät befindet sich im Override-Modus. Eventuell ist keine Transpondererkennung und somit keine Ortsbestimmung mehr möglich.	Verlassen Sie den Override-Modus.
350010	Interner Fehler: Das Gerät hat keine Failsafetasten.	Gerät einschicken. Ansprechpartner weltweit

3.8.8.37 380000 - Meldungen Sm@rtServer

380000 - Meldungen Sm@rtServer

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 380000-380100 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.8.38 390000-Meldungen Kennwortverschlüsselung und Zertifikate

390000-Meldungen Kennwortverschlüsselung und Zertifikate

Nummer	Wirkung/Ursachen	Abhilfe
390000	Das Runtime-Projekt wurde von einem anderen Windows-Benutzer geladen, als den, der das Zertifikat für die Kennwortverschlüsselung erstellt hat.	Ein Runtime-Projekt mit verschlüsselten Kommunikationsverbindungen zu S7-Steuerungen muss für jeden Windows-Benutzer geladen werden, der in Runtime diese Verbindungen nutzen möchte.

Hinweis

Meldungen ohne zusätzliche Information

Meldungen 390001-390003 sind nicht beschrieben, da der Meldetext alle notwendigen Informationen enthält.

3.8.9 Systemmeldungen

3.8.9.1 Systemmeldungen

Einleitung

Im Folgenden werden die Systemmeldungen erläutert, die von den verschiedenen Modulen von WinCC erzeugt werden. Über den Menübefehl "WinCC - Systemmeldungen" können Sie diese Meldungen in Ihr Meldesystem integrieren.

Hinweis

Weitere Informationen zu den Systemmeldungen entnehmen Sie dem Meldeblock "Kommentar" der auftretenden Meldung.

Hinweis

Systemmeldungen werden in einer Meldeanzeige ausgegeben. Systemmeldungen werden in der Sprache ausgegeben, die aktuell an Ihrem Bediengerät eingestellt ist.

Das Zeitformat (AM/PM oder 24-Stunden-Format) wird von der eingestellten Sprache übernommen. Wenn keine Übersetzung der Meldetexte in dieser Sprache existiert, wird Englisch als Ersatzsprache gewählt und das entsprechende Zeitformat angezeigt.

3.8.9.2 1000000 - Meldungen Runtime

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-19 1000000 - Meldungen WinCC Runtime

Nummer	Wirkung/Ursachen	Abhilfe
1000000	Fehler	
1000001	Fehler beim Laden der Object Engine	
1000002	Kanal konnte nicht geladen werden	
1000003	Variable - Verletzung der Untergrenze	
1000004	Variable - Verletzung der Obergrenze	
1000005	Fehler beim Formatieren einer Variablen	
1000006	Fehler beim Skalieren einer Variablen	
1000100	Treiberfehler	
1000200	Status	
1000201	Object Engine wurde geladen	
1000202	Runtime wurde aktiviert	
1000203	Runtime wurde deaktiviert	
1000204	Verbindung nicht aufgebaut	
1000205	Verbindung aufgebaut	
1000206	Client-Verbindung aufgebaut	
1000207	Client-Verbindung abgebaut	
1000208	Client-Verbindung abgebrochen	
1000209	Verbindung gelöscht	
1000210	Verbindung geändert	
1000211	Verbindung neu erstellt	
1000300	Treiberstatus	
1000301	Die Legitimierung der Verbindung @1%s@ ist fehlgeschlagen. Das Kennwort ist falsch!	
1000302	Die PLC ist geschützt. Für die Verbindung @1%s@ muss ein Kennwort konfiguriert werden.	
1000303	Die Legitimierung der Verbindung @1%s@ ist fehlgeschlagen. Das Kennwort ist gesperrt. Bitte lokal (z.B. am Display) entsperren.	

3.8.9.3 1000800 - Meldungen WinCC-Client

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-20 1000800- Meldungen WinCC-Client

Nummer	Wirkung/Ursachen	Abhilfe
1000800	Import	
1000801	Export	
1000802	Löschen	
1000803	Neu	
1000804	Erneut laden	
1000805	DefaultServer	
1000806	Impliziter Update	
1000807	Update	

3.8.9.4 1000900 - Meldungen Performance Monitor

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-21 1000900 - Meldungen Performance Monitor

Nummer	Wirkung/Ursachen	Abhilfe
1000900	@1%s@:@7%s@ ist niedrig	
1000901	@1%s@:@7%s@ ist ok	
1000902	@1%s@:@7%s@ ist hoch	
1000903	@1%s@:@7%s@ ist ok	
1000904	@1%s@:@7%s@ ist niedrig	
1000905	@1%s@:@7%s@ ist ok	
1000906	@1%s@:@7%s@ ist hoch	
1000907	@1%s@:@7%s@ ist ok	
1000908	@1%s@:@7%s@ ist niedrig	
1000909	@1%s@:@7%s@ ist ok	
1000910	@1%s@:@7%s@ ist hoch	
1000911	@1%s@:@7%s@ ist ok	
1000912	@1%s@:Redundanzverlust des Terminaladapters @2%s@	
1000913	@1%s@:Redundanz des Terminaladapters @2%s@ wiederhergestellt	

Nummer	Wirkung/Ursachen	Abhilfe
1000914	@1%s@:Terminaladapter @2%s@ verbunden	
1000915	@1%s@:Terminaladapter @2%s@ getrennt	

3.8.9.5 1001000 - Meldungen Bilder in Runtime

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-22 1001000 - Meldungen Bilder in Runtime

Nummer	Wirkung/Ursachen	Abhilfe
1001000	Allgemeiner Fehler	
1001001	Bedienaktion nicht aktiv	
1001002	Falsches Bildformat	
1001003	Bild wurde nicht gefunden	
1001004	Keine Dynamik im Bild aktiv	
1001005	Variable konnte nicht geschrieben werden	
1001006	Dynamische Aktionen im Bild nicht aktiv	

3.8.9.6 1002000 - Meldungen Variablenarchivierung

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-23 1002000 - Meldungen Variablenarchivierung

Nummer	Wirkung/Ursachen	Abhilfe
1002000	Allgemeiner Fehler	
1002001	Fehler beim Initialisieren	
1002002	Fehler beim Laden der Runtime-Daten	
1002003	Fehler bei der Sprachumschaltung	
1002004	Fehler beim Zugriff auf die Datenbank	
1002005	Fehler beim Erzeugen von Runtime-Objekten	
1002006	Fehler bei der Online-Projektierung	
1002007	Fehler in Client-/Server-Umgebung	
1002008	Fehler im Speichermanagement	
1002009	Fehler bei der Erfassung von Meßwerten	
1002010	Fehler bei der Verarbeitung von Meßwerten	
1002011	Fehler bei der Archivierung von Meßwerten	

Nummer	Wirkung/Ursachen	Abhilfe
1002012	Fehler in Normierungs-DLL	
1002013	Fehler bei Option Anwenderarchiv	
1002014	Fehler bei prozeßgesteuerter Archivierung	
1002015	Fehler im API	
1002016	Fehler beim Applikationsfenster	
1002017	Systemfehler	
1002018	Fehler Überlauf der Datenbank-Queue	
1002019	Fehler Überlauf der Notifizierungs-Queue	
1002020	Fehler Überlauf der Normierungs-DLL-Queue - Datenverlust !	
1002021	Probleme mit der Verbindung zum zentralen Archivserver!	
1002022	Das Problem der Verbindung zum zentralen Archivserver wurde behoben	
1002023	Archivwert bearbeitet - Archivvariable=@10%s@ Zeitstempel=@1%s@ Neuer Wert=@2%s@ Alter Wert=@3%s@	
1002024	Archivwert erzeugt - Archivvariable=@10%s@ Zeitstempel=@1%s@ Wert=@2%s@	

3.8.9.7 1003018 - Meldungen Meldesystem

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-24 1003018 - Meldungen Meldesystem

Nummer	Wirkung/Ursachen	Abhilfe
1003018	Meldung hat keine Quittierpflicht Es wurde versucht eine nicht quittierpflichtige Meldung zu quittieren.	
1003019	Meldung ist bereits quittiert Es wurde versucht eine bereits quittierte Meldung zu quittieren.	
1003020	Meldeklasse zur Meldung nicht gefunden Die zur Meldung gehörende Meldeklasse existiert nicht.	
1003021	Meldungsstatus kann nicht verarbeitet werden Der Zustand der Meldung (Gekommen, Gegangen, Quittiert, ...) konnte nicht interpretiert werden.	
1003022	Meldung ist gesperrt Die Meldung tritt auf, wenn Sie über das API versuchen eine gesperrte Meldung auszulösen.	
1003023	Meldung Datum/Uhrzeit-Stempel ungültig Der vom AS stammende Datum/Uhrzeit-Stempel konnte nicht interpretiert werden.	

Nummer	Wirkung/Ursachen	Abhilfe
1003032	Meldefenster Templatename unbekannt Der Fehler tritt auf, wenn ein im Editor "Bilder" bereits projektiertes Meldefenster nachträglich gelöscht wurde. Bei Client-Server-Projekten könnte auch eine Kommunikationsproblem auf dem Netzwerk Ursache der Meldung sein.	
1003033	Meldefenster konnte nicht erzeugt werden.	
1003034	Meldefenster - Daten ungültig Die Projektdaten sind fehlerhaft.	
1003048	Bit liegt ausserhalb des Variablenbereichs Es wurde z. B. versucht das 18.Bit einer 16 Bit Variablen anzusprechen.	
1003049	Keine Variablenänderung	
1003050	Ereignisvariable(Bit) schon von Meldung belegt Das Bit der Meldevariablen ist bereits mit einer Meldung verschaltet.	
1003051	Quittiervariable(Bit) schon von Meldung belegt Das Bit der Quittiervariablen ist bereits mit einer Meldung verschaltet.	
1003052	Zustandsvariable schon von Meldung belegt Das Bit der Zustandsvariablen ist bereits mit einer Meldung verschaltet.	
1003053	Fehler beim Konvertieren des Variant-Datentyps	
1003054	Ereignisvariable(Bit) schon belegt-->anderer Typ	
1003055	Ereignis - Variable oder VariablenTyp ungültig Der Datentyp der Meldevariablen ist ungültig. Es wurde z. B. eine vorzeichenbehaftete Variable verwendet.	
1003056	Quittier - Variable oder VariablenTyp ungültig	
1003057	Status - Variable oder VariablenTyp ungültig Der Datentyp der Zustandsvariablen ist ungültig. Es wurde z.B. eine vorzeichenbehaftete Variable verwendet.	
1003058	Handle ungültig Die Fehlermeldung kann bei Zugriffen über das API auftreten.	
1003059	Selektionskriterien ungültig Die Fehlermeldung kann bei Zugriffen über das API auftreten.	
1003060	Ressourcen - DLL für Sprache nicht gefunden Es fehlt eine der sprachabhängigen Dateien.	
1003061	Fehler beim Anlegen des Memory Mapped Files Dieser interne Fehler deutet auf ein Speicherproblem hin.	
1003062	Fehler beim Anlegen des Synchronisationsmechanismus. Der Fehler tritt bei einer Überlastung des Betriebssystems auf.	
1003063	Falscher Parameter Die Fehlermeldung kann bei Zugriffen Über das API auftreten.	
1003064	Übergabepuffer zu klein Die Fehlermeldung kann bei Zugriffen Über das API auftreten.	

Nummer	Wirkung/Ursachen	Abhilfe
1003065	Funktion steht zur Zeit noch nicht zur Verfügung Die Fehlermeldung kann bei Zugriffen Über das API auftreten.	
1003066	NormierungsDLL Sendedaten nicht auswertbar	
1003067	NormierungsDLL. Funktion nicht ausführen	
1003068	Keine Meldeblöcke im Protokoll vorhanden	Überprüfen Sie das Meldefolgeprotokoll.
1003069	Ungültige Protokollkennung	
1003070	Protokollausdruck bereits aktiv Es wurde versucht ein aktives Protokoll erneut zu starten.	
1003071	Meldesystem-Runtime <Server> nicht initialisiert Der Server meldet, dass das Projekt ohne die Runtimekomponente der Meldearchivierung aktiviert wurde.	
1003072	Ausdruck des Meldefolgeprotokolls nicht gestartet.	
1003073	Ausdruck des Kurzzeitarchivprotokolls nicht gestartet.	
1003074	Ausdruck des Folgearchivprotokolls nicht gestartet.	
1003075	Max. Anzahl online projektierbarer Meldungen Die Anzahl der online projektierbaren Meldungen (standardmäßig 600) wurde überschritten.	
1003076	Parameterfehler im S7-Telegramm An der Schnittstelle zu S7 ist ein Parameterfehler aufgetreten.	
1003077	Parameterfehler zu ALGGRT An der Schnittstelle zu ALGRT ist ein Parameterfehler aufgetreten.	
1003078	Parameterfehler zu TLGRT An der Schnittstelle zu TLGRT ist ein Parameterfehler aufgetreten.	
1003079	Parameterfehler bei Zusatzdaten Die Zusatzdaten der Meldungen sind fehlerhaft.	
1003080	Parameterfehler bei AR_SEND Aufbau Der Aufbau der AR-SEND-Nutzdaten ist fehlerhaft.	
1003081	Allgemeiner Fehler Es ist ein interner Fehler aufgetreten, dessen Ursache nicht genauer spezifiziert werden kann.	
1003082	Meldungsverlust auf dem Automatisierungssystem	
1003083	Ankoppeln und Aufdaten des Automatisierungssystems aktiv	
1003084	Ankoppeln und Aufdaten des Automatisierungssystems beendet	
1003085	Anlagenänderung im laufenden Betrieb (CiR) aktiv	
1003086	Anlagenänderung im laufenden Betrieb (CiR) beendet	
1003087	Fehler beim Anmelden für Meldungsempfang	
1003098	Überlauf der Meldearchivierung - Meldungen gehen verloren	
1003099	[Rechnername]:Meldung [Meldungsnummer] gesperrt [Meldungstext der gesperrten Meldung] Die Meldung tritt auf, wenn eine Meldung gesperrt wird.	

Nummer	Wirkung/Ursachen	Abhilfe
1003100	[Rechnername]:Meldung [Meldungsnummer] freigegeben [Meldungstext der freigegebenen Meldung] Die Meldung tritt auf, wenn eine Meldung freigegeben wird.	
1003101	Quittieranforderung für Meldung [Meldungsnummer] wurde abgesetzt Die Meldung tritt auf, wenn eine Meldung quittiert wird.	
1003102	[Rechnername]:Meldungsgruppe [Meldungsgruppennummer] gesperrt Die Meldung tritt auf, wenn eine Meldungsgruppe gesperrt wird.	
1003103	[Rechnername]:Meldungsgruppe [Meldungsgruppennummer] freigegeben Die Meldung tritt auf, wenn eine Meldungsgruppe freigegeben wird.	
1003104	[Rechnername]:Überlauf der Meldearchivierung beendet - keine weiteren Meldungen gehen verloren	
1003105	[Rechnername]:Meldungs-Eingangsqueue erreicht kritischen Wert	
1003106	[Rechnername]:Meldungs-Eingangsqueue erreicht normalen Wert	
1003107	[Rechnername]:Meldung [Meldungsnummer] ausgeblendet:@1%s@	
1003108	[Rechnername]:Meldung [Meldungsnummer]eingeblendet:@1%s@	
1003109	Die Verbindung zum Master wurde unterbrochen.	
1003110	Unvollständige Konfigurationsdaten für Meldung [Rechnername]: festgestellt. Status: @1%d@.	
1003300	Geänderte Konfigurationsdaten für Meldung festgestellt.	
1003301	Fehler beim Anmelden für Meldungsempfang	
1003302	Aufgrund eines temporären Ressourcen-Engpasses konnten nicht alle Meldungen quittiert werden. Quittieren Sie erneut.	

3.8.9.8 1004000 - Meldungen Druckaufträge

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-25 100400 - Meldungen Druckaufträge

Nummer	Wirkung/Ursachen	Abhilfe
1004000	Allgemeiner Fehler	
1004001	PRT_OUT-Verzeichnis läuft voll	
1004002	Spool-Verzeichnis läuft voll	
1004003	Protokoll wurde nicht gedruckt. PRT_OUT-Verzeichnis voll	

Nummer	Wirkung/Ursachen	Abhilfe
1004004	Protokoll wurde nicht gedruckt. Spool-Verzeichnis voll	
1004005	Meldefolgeprotokoll wird wieder gedruckt	
1004006	Spool-Verzeichnis läuft voll	
1004007	Hardkopie wurde nicht gedruckt. Spool-Verzeichnis voll	

3.8.9.9 1005000 - Meldungen Textbibliothek

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-26 100500 - Meldungen Textbibliothek

Nummer	Wirkung/Ursachen	Abhilfe
1005000	Allgemeiner Fehler	
1005001	Fehler beim Abmelden der Runtime Applikationen.	
1005002	Fehler beim Anmelden der Runtime Applikationen	
1005003	Fehler beim Initialisieren des MMF. Es liegt ein Speicherfehler vor.	
1005004	Fehler beim Laden des MMF. Fehler beim Zugriff auf die Datenbank.	
1005005	Fehler beim Öffnen des MMF. Es liegt ein Speicherfehler vor.	
1005006	Fehler beim Anlegen des Service Fensters.	
1005007	Keine Sprache gefunden.	
1005008	TextID nicht gefunden. Die angeforderte Text-ID konnte in der Text Library nicht gefunden werden.	
1005009	Lesezugriff auf das MMF verweigert.	
1005010	Sprache nicht gefunden. Die angeforderte Sprache ist in der Text Library nicht projektiert.	
1005011	Sprachen Tabelle konnte nicht geöffnet werden. Entweder sind die Daten fehlerbehaftet oder die Tabelle ist in der Datenbank gesperrt.	
1005012	Text Tabelle konnte nicht geöffnet werden. Entweder sind die Daten fehlerbehaftet oder die Tabelle ist in der Datenbank gesperrt.	
1005013	Ungültige Sprache angegeben. Die angegebene Sprach-ID ist nicht gültig.	
1005014	DBConnect Fehler Es konnte keine Verbindung zur Datenbank eingerichtet werden.	

3.8.9.10 1006000 - Meldungen Globale Skripte

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-27 100600 - Meldungen Globale Skripte

Nummer	Wirkung/Ursachen	Abhilfe
1006000	Fehler	
1007000	Überlauf Überlast: entweder laufen zuviele Aktionen in einem zu kleinen Zyklus oder eine Aktion hängt (Endlosschleife, Ausgabe eines Dialogs). Alle anderen Aktionen befinden sich in der Warteschlange und können nicht bearbeitet werden.	
1007001	Aktionsfehler Einer der folgenden Fehler ist aufgetreten: • Exception in der Aktion (genaue Ursache unbekannt) • Exception beim Zugriff auf das Rückgabeergebnis (char* zugehöriger Speicher nicht gültig) • Stapelüberlauf bei der Ausführung der Aktion • Division durch 0 in der Aktion • Zugriff auf ein nicht vorhandenes Symbol in der Aktion • Access violation in der Aktion	Sie können zur genaueren Analyse des Fehlers die Funktion OnErrorExecute in Ihr Skript einbinden.
1007002	Überlauf Es sind Interne Listen übergelaufen.	
1007003	Verbindungsfehler Die Verbindung zum Server ist abgebrochen.	
1007004	Aktionsfehler 1 Die aufgerufene Funktion ist nicht bekannt.	Überprüfen Sie die Schreibweise des Funktionsaufrufs und die Implementierung der Funktion.
1007005	Aktionsfehler 2 Der Fehler kann mehrere Ursachen haben: • Die Aktion enthält keinen P-Code. Übersetzen Sie die Aktion erneut. • Die Funktion konnte nicht geladen werden, weil z.B. Funktionsname fehlerhaft ist. • Der Typ des Rückgabewerts der Funktion ist ungültig.	Sie können zur genaueren Analyse des Fehlers die Funktion OnErrorExecute in Ihr Skript einbinden.

Nummer	Wirkung/Ursachen	Abhilfe
1007006	Variablenfehler Eine angeforderte Variable wurde innerhalb von 10 sec nicht vom WinCC Explorer geliefert.	Überprüfen Sie die Schreibweise des Variablennamens. Bei externen Variablen könnte auch ein Kommunikationsproblem zwischen dem WinCC Explorer und der Steuerung vorliegen.Sie können zur genaueren Analyse des Fehlers die Funktion OnErrorExecute in Ihr Skript einbinden.
1007007	Info Weitere Hinweise unter "Diagnose von WinCC / Laufzeitüberwachung für Aktionen".	
1007009	Fehler in Thread Weitere Hinweise unter "Diagnose von WinCC / Laufzeitüberwachung für Aktionen".	

3.8.9.11 1008000 - Meldungen Benutzerverwaltung

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-28 100800 - Meldungen Benutzerverwaltung

Nummer	Wirkung/Ursachen	Abhilfe
1008000	Verbindung zum Chipkartenleser unterbrochen	
1008001	Falscher Loginname/Passwort	
1008002	Falscher Loginname/Passwort durch Chipkarte	
1008003	Manueller Login	
1008004	Login durch Chipkarte	
1008005	Manueller Logout	
1008006	Logout durch Chipkarte	
1008007	Automatischer Logout durch Zeitsteuerung	
1008008	Berechtigungen von Service Benutzer/Gruppe '@102%s@' wirksam	

3.8.9.12 1009000 - Meldungen Lifebeat Monitoring

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-29 100900 - Meldungen Lifebeat Monitoring

Nummer	Wirkung/Ursachen	Abhilfe
1009000	Fehler	
1009999	Variable @2%s@ existiert nicht	

3.8.9.13 1010000 - Meldungen STRRT

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-30 101000 - Meldungen STRRT

Nummer	Wirkung/Ursachen	Abhilfe
1010000	Fehler	

3.8.9.14 1011000 - Meldungen Sammelanzeige

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-31 1011000 - Meldungen Sammelanzeige

Nummer	Wirkung/Ursachen	Abhilfe
1011000	Sammelanzeigefehler im Hochlauf	
1011001	Sammelanzeigehierarchie nicht aktualisiert	
1011002	Sammelanzeige: Verbindungsstörung	
1011003	Sammelanzeige: Variable existiert nicht	

3.8.9.15 1011101 - Meldungen Picture Tree Manager

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-32 1011101 - Meldungen Picture Tree Manager

Nummer	Wirkung/Ursachen	Abhilfe
1011101	Fehler beim Hochlauf	

3.8.9.16 1011201 - Meldungen Split Screen Manager

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-33 1011201 - Meldungen Split Screen Manager

Nummer	Wirkung/Ursachen	Abhilfe
1011201	Fehler beim Hochlauf	
1011202	Der Füllstand des Projektlaufwerks beträgt mehr als 80%	
1011203	Das Projekt wurde mit einer falschen WinCC-Version erstellt	

3.8.9.17 1012001 - Meldungen Uhrzeitsynchronisation

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-34 1012001 - Meldungen Uhrzeitsynchronisation

Nummer	Wirkung/Ursachen	Abhilfe
1012001	umgeschaltet auf Masterbetrieb	
1012002	kann Uhrzeittelegramm nicht absetzen	
1012003	Zeitempfangsdienst: Signal schlecht oder ausgefallen	
1012004	kann Uhrzeittelegramm nicht empfangen	
1012005	kann kein Uhrzeittelegramm am redundanten Bus empfangen	
1012006	kein Uhrzeittelegramm. Umschalten auf redundanten Device	
1012007	kann Uhrzeittelegramm ordnungsgemäß absetzen	
1012008	Zeitempfangsdienst arbeitet ordnungsgemäß	
1012009	kann Uhrzeittelegramme ordnungsgemäß empfangen	
1012010	kann Uhrzeittelegramme am redundanten Bus ordnungsgemäß empfangen	
1012011	umgeschaltet auf Slavebetrieb	
1012012	Uhrzeitsynchronisation deaktiviert	
1012013	Uhrzeitsynchronisation aktiviert	
1012014	DCF77-Client-Service ist ausgefallen	
1012015	DCF77-Client-Service arbeitet ordnungsgemäß	
1012016	@2%s@ umgeschaltet auf Masterbetrieb @3%s@	
1012017	@2%s@ umgeschaltet auf Slavebetrieb @3%s@	

Nummer	Wirkung/Ursachen	Abhilfe
1012018	@2%s@ kann Uhrzeittelegramm nicht absetzen	
1012019	@2%s@ kann Uhrzeittelegramm ordnungsgemäß absetzen	
1012020	@2%s@ stellt lokale Zeit	
1012021	LAN- Zeitsynchronisation mit PC @2%s@ gestört	
1012022	LAN- Zeitsynchronisation auf PC @2%s@ eingestellt	
1012023	LAN- Zeitsynchronisation mit PC @2%s@ etabliert	
1012024	Konfigurierte Device Namen der Uhrzeitsynchronisation stimmen nicht mit PC-Installation überein	
1012025	LAN- keine Zeitübernahme vom angebundenen WinCC-Server möglich	
1012026	Uhrzeitsprung bemerkt - umgeschaltet auf dauerhaften Slavebetrieb	
1012027	Uhrzeitsprung bemerkt - Zeitsynchronisation dauerhaft deaktiviert	
1012028	Zeitempfangsdienst nicht gestartet	
1012029	Zeitempfangsdienst läuft	
1012030	Uhrzeitsynchronisation ist deaktiviert	

3.8.9.18 1012200 - Meldungen Redundancy

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-35 1012200 - Meldungen Redundancy

Nummer	Wirkung/Ursachen	Abhilfe
1012200	Partnerstation fällt aus	
1012201	Partnerstation wieder angelaufen	
1012202	Projekte sind nicht funktionsgleich	
1012203	Archivabgleich fehlerhaft	
1012204	Interner Fehler bei Redundancy	
1012205	Verbindung zum Partner gestört	
1012206	Verbindung zum Partner wieder hergestellt	
1012207	Partnerserver - WinCC ist nicht gestartet	
1012208	Archivabgleich beginnt	
1012209	Archivabgleich beendet	
1012210	Tag Logging wird abgeglichen	
1012211	Tag Logging Abgleich beendet	
1012212	Alarm Logging wird abgeglichen	
1012213	Alarm Logging Abgleich beendet	
1012214	User Archive wird abgeglichen	

Nummer	Wirkung/Ursachen	Abhilfe
1012215	User Archive Abgleich beendet	
1012216	Abgleich wurde unterbrochen	
1012217	Partnerserver-Projekt nicht aktiviert	
1012218	Client wurde automatisch umgeschaltet	
1012219	Client wurde manuell umgeschaltet	
1012220	Abgleich nicht für alle User Archive bereit	
1012221	Abgleich für alle User Archive bereit	
1012222	Hauptverbindung gestört	
1012223	Hauptverbindung betriebsbereit	
1012224	Reserveverbindung gestört	
1012225	Reserveverbindung betriebsbereit	
1012226	Partnerserver-Projekt aktiviert	
1012227	Fehler: Partnerrechner ist kein Server	
1012228	Archivabgleich beginnt '@2%s@'	
1012229	Archivabgleich beendet '@2%s@'	
1012240	RedundancyControl Fehler löst Umschaltung aus	
1012241	RedundancyControl: Umschaltung auf Status	
1012242	Deltaladen wurde gestartet	
1012243	Deltaladen wurde beendet	
1012244	Überlast beim Onlineabgleich von Alarm Logging	
1012245	RedundancyControl: Verlust der seriellen Verbindung	
1012246	RedundancyControl: Serielle Verbindung wieder hergestellt	
1012247	OS Server (Standby) Redundanzfehler	
1012248	OS Server (Standby) Redundanz wieder hergestellt	
1012250	Mögliche Inkonsistenz der Archivdatenbanken	
1012251	Interner Fehler	
1012252	Backup:Fehler beim Schreiben auf Backup - Pfad	
1012253	Backup:Unzureichender Speicherplatz auf Ziellaufwerk	
1012254	Keine Verbindung zum WriteArchiveServer	
1012255	Keine Verbindung zu Microsoft Message Queue	
1012256	Keine Verbindung zum WinCC-Projekt	
1012257	Keine Verbindung zur Datenbank	
1012258	Keine Verbindung zur Textbibliothek	
1012259	Fehler beim Erzeugen der TagLogging Daten	
1012260	Fehler beim Erzeugen der AlarmLogging Daten	
1012261	Achtung: Backup wird bis zum Wiederanlauf des Partner-Server verzögert.	
1012265	Datenbankverifizierung fehlgeschlagen	
1012301	Kein Zugriff auf SQL-Server (Datenverlust ist möglich)	

Nummer	Wirkung/Ursachen	Abhilfe
1012348	Nicht genügend freier Speicherplatz auf Projektlaufwerk	
1012349	Verlust der Verbindung über Netzkarte (MAC)-Adresse	
1012350	Verbindung über Netzkarte (MAC)-Adresse wieder hergestellt	
1012351	RedundancyControl: Systemblockade festgestellt. Umschaltung auf Status Fault.	
1012352	RedundancyControl: Systemblockade festgestellt. Starten Sie den Rechner baldmöglichst neu.	
1012354	[Rechnername):RedundancyControl: Status in FAULT geändert. Server-Isolation ist jedoch nicht aktiviert.	
1012355	[Rechnername):RedundancyControl: Status in FAULT geändert. Server-Isolation ist jedoch durch @1%s@ gesperrt. Grund: @2%s@	
1012356	RedundancyControl: Status in FAULT geändert => Server ist isoliert	
1012357	[Rechnername):RedundancyControl: Status in FAULT geändert. Automatischer Neustart ist jedoch nicht aktiviert.	
1012358	[Rechnername):RedundancyControl: Status in FAULT geändert. Automatischer Neustart ist jedoch gesperrt. Der Netzwerkadapter ist getrennt und DHCP frei gegeben.	
1012359	[Rechnername):RedundancyControl: Neustart des Rechners durch @1%s@ gesperrt. Grund: @2%s@	
1012360	[Rechnername):RedundancyControl: Neustart des Rechners abgebrochen. Der letzte Neustart fand vor weniger als @1%s@ s statt.	
1012361	[Rechnername):RedundancyControl: Neustart des Rechners abgebrochen. Nach @1%s@ Neustarts ist kein weiterer Neustart für @2%s@ s zulässig.	
1012362	[Rechnername):RedundancyControl: Rechner wird in @1%s@ s neu gestartet	
1012363	[Rechnername):RedundancyControl: FEHLER-Zustand zurückgestellt bis der Partner einen stabilen Zustand eingenommen hat.	

3.8.9.19 1012400 - Meldungen WebNavigatorClient

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-36 1012400 - Meldungen WebNavigatorClient

Nummer	Wirkung/Ursachen	Abhilfe
1012400	WebClient @1%s@ Verbindung aufgebaut (Benutzer=@2%s@)	
1012401	WebClient @1%s@ Verbindung abgebaut (Benutzer=@2%s@)	

3.8.9.20 1012500 - Meldungen Process Historian

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-37 1012500 - Meldungen Process Historian

Nummer	Wirkung/Ursachen	Abhilfe
1012500	[Rechnername]:Starte Recovery für Process Historian	
1012501	[Rechnername]:Recovery für Process Historian abgeschlossen	
1012502	[Rechnername]:Keine Kommunikation zum Process Historian möglich	
1012503	[Rechnername]:Kommunikation zum Process Historian gestört	
1012504	[Rechnername]:Kommunikation zum Process Historian wiederhergestellt	
1012505	[Rechnername]:Process Historian Server seit @1%s@ offline	
1012506	[Rechnername]:Pufferlimit Kanal @1%s@ überschritten	
1012507	[Rechnername]:Pufferlimit Kanal @1%s@ normal	
1012508	[Rechnername]:Weniger als @1%d@ GB freier Speicherplatz auf Datenträger '@2%s@' für die Kommunikation mit dem Process Historian.	
1012509	[Rechnername]:Kommunikation des Process Historian wurde beendet. Weniger als @1%d@ GB freier Speicherplatz auf Datenträger '@2%s@'.	

Nummer	Wirkung/Ursachen	Abhilfe
1012510	[Rechnername]:Verbindung zum Process Historian konnte nicht aufgebaut werden (Konfiguration überprüfen).	
1012600	[Rechnername]:Datenspeicher zu @1%d@% belegt	
1012601	[Rechnername]:System ausgelastet	
1012602	[Rechnername]:Redundanz ausgefallen	
1012603	[Rechnername]:Redundanz wiederhergestellt	
1012604	[Rechnername]:Lizenzvolumen überschritten. Herunterfahren in @1%d@ Tagen	
1012605	[Rechnername]:PH-Ready @1%s@ ausgefallen	
1012606	[Rechnername]:Es hat eine automatische Redundanzumschaltung der Process Historian-Server stattgefunden.	
1012607	[Rechnername]:Weniger als @1%d@ GB freier Speicherplatz für die Datenbank des Process Historian.	
1012608	[Rechnername]:Weniger als @1%d@ GB freier Speicherplatz für die "tempdb" Datenbank.	
1012609	[Rechnername]:Weniger als @1%d@ GB freier Speicherplatz für die Notfallwiederherstellung.	
1012610	[Rechnername]:Neues Backup für die Notfallwiederherstellung schlug fehl. Zu wenig Speicherplatz.	
1012611	[Rechnername]:Beim Erstellen eines Backups für die Notfallwiederherstellung ist ein unbekannter Fehler aufgetreten.	
1012612	[Rechnername]:Speicherpfad @1%s@ nicht zugänglich für Notfallwiederherstellung.	
1012613	[Rechnername]:Der Not-Schwellenwert für Laufwerk @1%s@ wurde erreicht. Darum ist Process Historian gesperrt.	
1012614	[Rechnername]:Keines der vorbereiteten Segmente existiert. Prüfen Sie bitte warum.	
1012615	[Rechnername]:Nicht alle vorbereiteten Segmente wurden erzeugt.	

3.8.9.21 1012700 - Meldungen Redundancy Selbstdiagnose

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-38 1012700 - Meldungen Redundancy Selbstdiagnose

Nummer	Wirkung/Ursachen	Abhilfe
1012700	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ ist ungültig.	
1012701	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ hat die Fehlergrenze HIGH überschritten.	
1012702	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ hat die Fehlergrenze LOW unterschritten.	
1012703	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ hat die Warngrenze HIGH überschritten.	
1012704	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ hat die Warngrenze LOW unterschritten.	
1012705	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ überschreitet die Fehlergrenze nicht mehr.	
1012706	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ ist OK.	
1012707	[Rechnername]: Teilnehmer @10%s@ verursacht @2%s@.	
1012708	[Rechnername]: Wert @7%s@ von Teilnehmer @10%s@ ist ungültig.	

3.8.9.22 1016000 - Meldungen IndustrialDataBridge

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-39 1016000 - Meldungen IndustrialDataBridge

Nummer	Wirkung/Ursachen	Abhilfe
1016000	@1%s@ Provider-Initialisierung fehlgeschlagen für Verbindung: @2%s@.	
1016001	@1%s@ Consumer-Initialisierung fehlgeschlagen für Verbindung: @2%s@.	
1016002	@1%s@ Provider-Datenübertragung fehlgeschlagen für Verbindung: @2%s@.	
1016003	@1%s@ Consumer-Datenübertragung fehlgeschlagen für Verbindung: @2%s@.	
1016004	Provider-Initialisierung Datenbank fehlgeschlagen für Verbindung: @2%s@.	

Nummer	Wirkung/Ursachen	Abhilfe
1016005	Consumer-Initialisierung Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016006	Provider-Datenübertragung Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016007	Consumer-Datenübertragung Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016008	Provider-Initialisierung dynamische Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016009	Consumer-Initialisierung dynamische Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016010	Provider-Datenübertragung dynamische Datenbank fehlgeschlagen für Verbindung: @2%s@.	
1016011	Consumer-Datenübertragung dynamische Datenbank fehlgeschlagen für Verbindung: @2%s@.	

3.8.9.23 12508141 - Meldungen

Bedeutung der Systemmeldungen

Nachfolgend finden Sie die Liste der wichtigsten Systemmeldungen.

Tabelle 3-40 12508141 - Meldungen

Nummer	Wirkung/Ursachen	Abhilfe
12508141	@10%s@: @7%s@ @102%s@ neu=@3%g@ @8%s@ alt=@2%g@ @8%s@	

3.8.10 Bitbelegungen Variablen

3.8.10.1 Variablen für Meldegruppen und Meldungen

Bitbelegung Zustandsvariable

Einleitung

In WinCC definieren Sie bei Bedarf eine Zustandsvariable für eine Meldung oder eine Meldegruppe. Die Zustandsvariable einer Meldegruppe stellt die Zustände der enthaltenen Meldungen und untergeordnete Meldegruppen dar.

Beschreibung

In einer Zustandsvariablen sind folgende Zustände einer Meldung oder Meldegruppe gespeichert:

- Meldung gekommen / gegangen (Zustandsbit)
- Meldung steht zum Quittieren an / quittiert (Quittierbit)

Datentyp einer Zustandsvariablen

Die Zustandsvariable ist eine interne oder externe WinCC-Variable. Folgende Datentypen verwenden Sie für die Zustandsvariable:

- Vorzeichenloser 8-Bit-Wert
  Der Datentyp stellt die Zustände von maximal vier Meldungen dar.
- Vorzeichenloser 16-Bit-Wert
  Der Datentyp stellt die Zustände von maximal 8 Meldungen dar.
- Vorzeichenloser 32-Bit-Wert
  Der Datentyp stellt die Zustände von maximal 16 Meldungen dar.

Jede Meldung oder Meldegruppe belegt in der Zustandsvariablen zwei Bits.

Hinweis

Ein Zustandsbit einer Zustandsvariablen darf nur genau einer Meldung zugeordnet sein.

Quittierbit der Zustandsvariablen einer Meldung

Wenn eine Meldung gekommen ist und nicht quittiert wurde, hat das Quittierbit den Zustand "1".

Wenn die Meldung quittiert wurde, wechselt das Quittierbit auf den Zustand "0".

Meldeereignis	Quittierbit
Meldung gekommen nicht quittiert	1
Meldung quittiert	0

Quittierbit der Zustandsvariablen einer Meldegruppe

Wenn mindestens eine Meldung gekommen ist und nicht quittiert wurde, wechselt in einer Meldegruppe das Quittierbit in den Zustand "1".

Wenn alle Meldungen der Meldegruppe quittiert wurden, wird das Quittierbit wieder auf "0" gesetzt.

Zustandsbit der Zustandsvariablen einer Meldung oder Meldegruppe

Wenn eine Meldung oder Meldegruppe gekommen ist und noch nicht gegangen, hat das Zustandsbit den Zustand "1". Wenn die Meldung gegangen ist, wechselt das Zustandsbit auf den Zustand "0".

Meldeereignis	Zustandsbit
Meldung oder Meldegruppe gekommen	1
Meldung gegangen	0

Hinweis

Meldungen, deren Meldeklasse die Eigenschaft "Meldungen ohne Zustand" besitzt, sind in einer Zustandsvariablen nicht darstellbar.

Zustandsvariable einer gesperrten Meldung

Wenn Sie eine Meldung in Runtime sperren, werden die Bits in der Zustandsvariablen unabhängig vom Meldezustand rückgesetzt oder nicht gesetzt. In der Zustandsvariablen wird der Zustand "gesperrt" einer Meldung nicht dargestellt.

Hinweis

Um den gesperrten Zustand einer Meldegruppe zu visualisieren, verwenden Sie die Sperrvariable der Meldegruppe.

Zustandsvariable beim zeitfolgerichtigen Melden

Um den Zustand einer zeitfolgerichtigen Meldung darzustellen, verwenden Sie eine Strukturvariable. In der Strukturvariablen repräsentiert z. B. das Element "EventState" den Zustand einer zeitfolgerichtigen Meldung.

Position Zustands- und Quittierbit

Sie legen die Position des Zustandsbits innerhalb einer Zustandsvariablen fest. Das Quittierbit muss zu diesem Bit einen bestimmten Abstand. Für den jeweiligen Datentyp gelten folgende Festlegungen:

Datentyp	Position Zustandsbit	Abstand	Position Quittierbit
Vorzeichenloser 8-Bit-Wert	0 - 3	4	4 - 7
Vorzeichenloser 16-Bit-Wert	0 - 7	8	8 - 15
Vorzeichenloser 32-Bit-Wert	0 - 15	16	16 - 31

Beispiele

Zustandsvariable vom Datentyp "32 Bit unsigned"

Zustandsbit = Position 9

Abstand = 16 Positionen

Quittierbit = Position 25

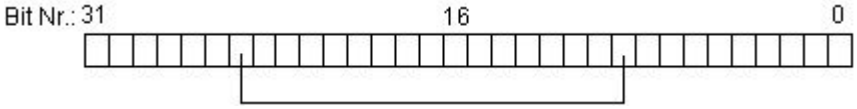

Zustandsvariable vom Datentyp "16 Bit unsigned"

Zustandsbit = 6

Abstand = 8 Positionen

Quittierbit = Position 14

Zustandsvariable vom Datentyp "8 Bit unsigned"

Zustandsbit = 2

Abstand = 4 Positionen

Quittierbit = Position 6

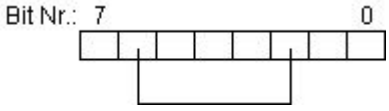

Funktionslisten an eine Wertänderung projektieren

Bei Bedarf projektieren Sie eine Funktionsliste an das Ereignis "Wertänderung" der Variablen. Um die Anzahl der möglichen Signalzustände der Variablen zu reduzieren, blenden Sie nicht relevante Bits der Variable aus. Sie projektieren das Ausblenden der nicht relevanten Bits über C-Skript oder VB-Skript.

Das Ausblenden der nicht relevanten Bits ist unter folgender Voraussetzung nicht notwendig:

- Sie verwenden für jede Meldung eine eigene Zustandsvariable.
- Die nicht relevanten Bits dieser Zustandsvariablen haben immer den Wert "0".

Siehe auch

Variablen einer Meldegruppe (Seite 811)

Bitbelegung Quittiervariable

Einleitung

In WinCC definieren Sie bei Bedarf eine Quittiervariable für eine Meldung oder eine Meldegruppe.

Beschreibung

Die Quittiervariable verwenden Sie wie folgt:

- Eine Meldung quittieren
- Eine Meldegruppe quittieren
  Die Quittiervariable einer Meldegruppe quittiert alle enthaltenen Meldungen und untergeordnete Meldegruppen.
- Den Quittierstatus einer Meldegruppe anzeigen
- Die Quittierung an andere WinCC-Komponenten anbinden
  Über die Quittiervariable verbinden Sie bei Bedarf mit der Quittierung andere WinCC-Komponenten.

Wenn der Benutzer in Runtime eine Meldung oder Meldegruppe quittiert, wird das zugehörige Quittierbit gesetzt.

Verwendung Quittiervariablen

Bei der Projektierung von Quittiervariablen haben Sie folgende Möglichkeiten:

- Sie projektieren für jede Meldung oder Meldegruppe eine eigene Quittiervariable.
- Sie fassen mehrere Meldungen oder Meldegruppen in einer Quittiervariablen zusammen. Über das Quittierbit wird zwischen den Meldungen oder Meldegruppen unterschieden.

Jede Meldung belegt in der Quittiervariablen ein Bit.

Datentyp einer Quittiervariablen

Der Aufbau der Quittiervariablen ist nicht festgelegt. Der Datentyp ist davon abhängig, wie viele Meldungen Sie in der Quittiervariablen darstellen wollen.

Rücksetzen des Quittierbits der Quittiervariablen

Sie projektieren das Setzen oder Rücksetzen des Quittierbits in WinCC, z. B. über eine Schaltfläche oder über die Steuerung.

Hinweis

Wenn der Benutzer einer Meldegruppe in Runtime quittiert, zeigt das Quittierbit die Quittierung in der Meldeanzeige nicht an.

- Siehe auch "Rücksetzen des Quittierbits"
- Siehe auch "Variablen einer Meldegruppe"

Siehe auch

Variablen einer Meldegruppe (Seite 811)

3.8.10.2 Variablen für Meldegruppen

Bitbelegung Sperrvariable

Einleitung

In WinCC definieren Sie bei Bedarf eine Sperrvariable für eine Meldegruppe.

Beschreibung

Die Sperrvariable einer Meldegruppe dient zur Auswertung des gesperrten Zustands der Meldegruppe. Sie legen dafür in der Sperrvariablen ein Sperrbit fest.

Verwendung der Sperrvariablen

Bei der Projektierung von Sperrvariablen haben Sie folgende Möglichkeiten:

- Sie projektieren für jede Meldegruppe eine eigene Sperrvariable.
- Sie fassen mehrere Meldegruppen in einer Sperrvariablen zusammen. Über das Sperrbit wird zwischen den Meldegruppen unterschieden.

Wenn der Benutzer in Runtime eine Meldegruppe sperrt, wird in der projektierten Variablen das zugehörige Sperrbit gesetzt.

Datentyp einer Sperrvariablen

Der Aufbau der Sperrvariablen ist nicht festgelegt. Der Datentyp ist davon abhängig, wie viele Meldungen Sie in der Sperrvariablen darstellen wollen.

Siehe auch

Variablen einer Meldegruppe (Seite 811)

Bitbelegung Anzeigeunterdrückungsvariable

Einleitung

In WinCC definieren Sie bei Bedarf eine Anzeigeunterdrückungsvariable für eine benutzerdefinierte Meldegruppe. Die Anzeigeunterdrückungsvariable speichert die möglichen Anlagenzustände. Nur ein Anlagenzustand kann aktiv sein. Deshalb ist in Anzeigeunterdrückungsvariable immer nur ein Bit gesetzt.

Beschreibung

Über die Anzeigeunterdrückungsmaske legen Sie fest, bei welchen Anlagenzuständen eine Meldung einer Meldegruppe in Runtime nicht angezeigt wird.

Datentyp der Anzeigeunterdrückungsvariablen

Die Anzeigeunterdrückungsvariable ist eine externe WinCC-Variable, die den Anlagenzustand speichert. Folgende Arten von Datentypen verwenden Sie für die Anzeigeunterdrückungsvariable:

- Vorzeichenloser 8-Bit-Wert
- Vorzeichenloser 16-Bit-Wert
- Vorzeichenloser 32-Bit-Wert

Anzeigeunterdrückungsmaske

Im Inspektorfenster einer Meldung legen Sie über die Anzeigeunterdrückungsmaske die Anlagenzustände fest, bei denen die Meldung nicht angezeigt wird. Welche Bits welchen Anlagenzuständen entsprechen, wird vom Projekteur einer Anlage festgelegt. Es kann maximal 32 Zustände einer Anlage geben.

Die ausgewählten Anlagenzustände tragen Sie als Hexadezimalwert im Inspektorfenster der Meldung unter "Anzeigeunterdrückung > Maske" ein.

Die Anzeigeunterdrückungsvariable speichert den Anlagenzustand. Wenn ein Anlagenzustand eintritt, der in der Maske enthalten ist, wird die Meldung nicht mehr angezeigt. Um festzustellen, welche Meldungen im Augenblick aufgrund des Anlagenzustands nicht angezeigt werden, schaltet der Benutzer die Meldeanzeige in Runtime auf die Anzeige "Meldungen mit Anzeigeunterdrückung".

Beispiel

Im Folgenden sehen Sie fünf Beispiele für eine Anzeigeunterdrückungsmaske:

- Anzeigeunterdrückungsmaske 0x0. Die Meldung wird angezeigt.

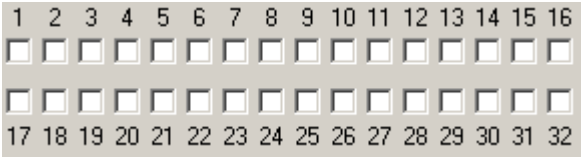

- Anzeigeunterdrückungsmaske 0x1. Wenn in der Anzeigeunterdrückungsvariablen das Bit "1" gesetzt ist, wird die Meldung nicht angezeigt.

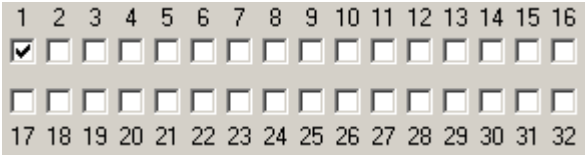

- Anzeigeunterdrückungsmaske 0xD. Wenn in der Anzeigeunterdrückungsvariablen das Bit "1", "3" oder "4" gesetzt ist, wird die Meldung nicht angezeigt.

- Anzeigeunterdrückungsmaske 0x80000008. Wenn in der Anzeigeunterdrückungsvariablen das Bit "4" oder "32" gesetzt ist, wird die Meldung nicht angezeigt.

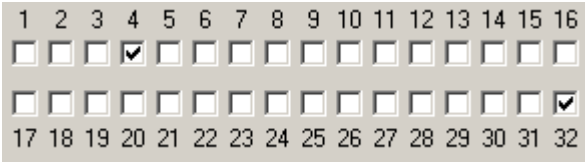

- Anzeigeunterdrückungsmaske 0xFFFFFFFF. Wenn in der Anzeigeunterdrückungsvariablen ein beliebiges Bit gesetzt ist, wird die Meldung nicht angezeigt.

Siehe auch

Variablen einer Meldegruppe (Seite 811)

Mit Archiven arbeiten 4

4.1 Arbeiten mit Archiven

4.1.1 Grundlagen zu Archiven

Einleitung

WinCC stellt Ihnen für die Archivierung von Prozessdaten für die HMI Runtime folgende Archivarten zur Verfügung:

- Variablenarchive
- Meldearchive

Ein Variablenarchiv dient zur Archivierung von Prozessdaten aus einer industriellen Anlage.

Ein Meldearchiv dient zur Archivierung von auftretenden Meldungen in dem beobachteten Prozess.

Prinzip

Der Aufbau und die Funktionsweise der beiden Archivarten ist weitgehend gleich. Die Projektierung wird dadurch sehr einfach und übersichtlich. Bei beiden Archivarten legen Sie die gleichen Eigenschaften für das Archiv fest. Für beide Archivarten stehen Ihnen auch die gleichen Archivierungsmethoden zur Verfügung.

Folgende Archivierungsmethoden sind verfügbar:

- Umlaufarchiv
  Wenn ein Umlaufarchiv vollständig befüllt ist, werden die ältesten Einträge überschrieben.
- Segmentiertes Umlaufarchiv
  Bei einem segmentierten Umlaufarchiv werden mehrere gleich große Archivsegmente erstellt, die nacheinander befüllt werden. Wenn alle Archivsegmente vollständig befüllt sind, wird das älteste Archivsegment überschrieben.
- Archiv mit füllstandsabhängiger Systemmeldung
  Bei Erreichen eines definierten Füllstands wird eine Systemmeldung ausgelöst.
- Archiv mit füllstandsabhängigem Auslösen eines Ereignisses
  Wenn das Archiv vollständig befüllt ist, wird das Ereignis "Überlauf" ausgelöst. Mit dem Ereignis "Überlauf" triggern Sie eine Systemfunktion.

4.1.2 Eigenschaften von Archiven

Einleitung

Die Eigenschaften eines Variablenarchivs legen Sie im Editor "Variablenarchive" fest.

Die Eigenschaften eines Meldearchivs legen Sie im Editor "Meldearchive" fest.

Die Projektierung der Eigenschaften von Variablenarchiv und Meldearchiv ist identisch. Die Eigenschaften projektieren Sie entweder direkt in der Tabelle des jeweiligen Editors oder in den Eigenschaften des Archivs im Inspektorfenster.

Allgemeine Eigenschaften

- Name
  Der Name des Archivs ist frei wählbar und muss mindestens einen Buchstaben oder eine Ziffer enthalten.
- Ablageort
  Der Ablagort legt fest, wo das Archiv gespeichert wird.Welche Ablageorte zur Verfügung stehen, ist abhängig vom Bediengerät.
- Größe
  Die Größe eines Archivs ist abhängig vom Archivtyp und den gewählten Einstellungen.
  - Größe eines Variablenarchivs
    Die Größe eines Variablenarchivs errechnet sich folgendermaßen:
    Anzahl der Einträge * Länge eines einzelnen zu archivierenden Variablenwerts.
    Im Eigenschaftsfenster wird unter dem Eingabefeld "Anzahl Datensätze" angezeigt, welche maximale Größe das Archiv annehmen würde, wenn die gerade gewählte Anzahl an Datensätzen beibehalten wird. Die maximale Archivgröße ist durch das Speichervolumen des Speichermediums begrenzt.
  - Größe eines Meldearchivs
    Die Größe eines Meldearchivs errechnet sich aus der Anzahl der Datensätze, die Sie angeben, und aus der ungefähren Größe deren Einträge. Die Größe eines Eintrags hängt davon ab, ob der Meldetext und die dazu gehörenden Variablenwerte ebenfalls archiviert werden.
- Startverhalten
  Unter Startverhalten können Sie festlegen, dass die Archivierung mit dem Runtime-Start beginnt. Aktivieren Sie dazu das Kontrollkästchen "Archivierung bei Runtime-Start aktivieren".
  Darüber hinaus können Sie das Verhalten bei Runtime-Start steuern. Aktivieren Sie dazu "Archiv zurücksetzen", wenn bereits archivierte Daten mit den neuen Daten überschrieben werden sollen. Wenn Sie bereits archivierte Daten erhalten wollen, wählen Sie die Option "Archiv fortführen". Mit dieser Einstellung werden die zu archivierenden Daten einem bestehenden Archiv hinzugefügt.

Hinweis

Den Neustart eines Archivs in Runtime können Sie über Systemfunktionen steuern.

Automatische Archiveinträge

In Runtime werden folgende Archiveinträge standardmäßig erstellt:

Eintrag	Dateiformat	Archivart	Bedeutung
RT_DIS	Alle	Variablenarchiv	Zeigt an, dass die Verbindung zum Archiv zu diesem Zeitpunkt unterbrochen wurde. (In der Kurvenanzeige wird für diese Zeitspanne eine fett dargestellte Linie angezeigt.)
RT_OFF	Alle	Variablenarchiv	Zeigt an, dass Runtime zu diesem Zeitpunkt heruntergefahren wurde. (In der Kurvenanzeige wird für diese Zeitspanne keine Linie angezeigt.)
RT_ERR	Alle	Variablenarchiv Meldearchiv AuditTrail[1]	Zeigt im Zielarchiv an, dass ein Kopiervorgang nicht erfolgreich war oder unterbrochen wurde. (Die Archivkopie wurde nicht vollständig erstellt.)
$RT_COUNT	*.CSV *. TXT	Variablenarchiv Meldearchiv AuditTrail[1]	Dieser Eintrag wird am Ende des Archivs erstellt und dient zur Steigerung der Systemleistung beim Runtimestart.

[1] Die Archivart "AuditTrail" nicht für alle Bediengeräte zur Verfügung.

Siehe auch

Grundlagen zu Archiven (Seite 1071)

Grundlagen zur Variablenarchivierung (Seite 639)

Grundlagen zum Archivieren von Meldungen (Seite 921)

4.1.3 Ablageorte von Archiven

Ablageort eines Archivs

Wenn Sie in WinCC ein Archiv projektieren, sind die verfügbaren Ablage- und Speicherorte abhängig vom verwendenen Bediengerät.

Bediengeräte	Unterstützte Archive			Unterstützte Ablageorte	Unterstützte Speicherorte
	Meldungen	Variablen	Audit Trail		
Basic Panels[1]	nein	nein	nein	-	-
Basic Panels 2nd Generation[2]	ja	ja	nein	TXT-Datei (Unicode)	USB-Speicher (an USB-Schnittstelle)
Comfort Panels[3]	ja	ja	ja	CSV-Datei (ASCII) RDB-Datei TXT-Datei (Unicode)	Storage Card (SD) Storage Card (USB) Netzlaufwerk

Bediengeräte	Unterstützte Archive			Unterstützte Ablageorte	Unterstützte Speicherorte
	Meldungen	Variablen	Audit Trail		
Mobile Panels[4]	ja	ja	ja	CSV-Datei (ASCII) RDB-Datei TXT-Datei (Unicode)	Storage Card (MMC) Storage Card (USB) Netzlaufwerk
• Mobile Panels 2nd Generation[5]	ja	ja	ja	CSV-Datei (ASCII) RDB-Datei TXT-Datei (Unicode)	Storage Card (SD) USB-Speicher (an USB-Schnittstelle) Netzlaufwerk
PC-Systeme mit Runtime Advanced	ja	ja	ja	CSV-Datei (ASCII) Datenbank RDB-Datei TXT-Datei (Unicode)	Lokales Dateisystem Netzlaufwerk

[1] KP 300, KP 400, KTP 1000, TP 1500

[2] KTP 400, KTP 700, KTP 900, KTP 1200

[3] Alle Bediengeräte aus der Geräteliste

[4] Nur Mobile Panel 277

[5] KTP 400F Mobile, KTP 700 Mobile, KTP 900 Mobile - inklusive fehlersicherer Varianten

Hinweis

Archivieren auf Netzlaufwerken

Archivieren Sie Meldungen, Variablen und AuditTrail nicht direkt auf einem Netzlaufwerk. Unterbrechungen einer Netzleitung sind nicht vorhersehbar. Deshalb ist ein zuverlässiger Betrieb von Archiven und von AuditTrail nicht gewährleistet.

Speichern Sie die Archive auf der lokalen Festplatte oder der lokalen Storage Card. Um die Archive zur Langzeitarchivierung auf einem Netzlaufwerk zu sichern, verwenden Sie die Systemfunktion "ArchiviereProtokolldatei". Dadurch ist ein zuverlässiger Betrieb garantiert.

Syntaxbeispiele für Speicherorte

Speicherort Storage Card:

- <\Storage Card MMC\My_Archives\TagLogs>: Speichert das Archiv auf der Storage Card MMC im Unterverzeichnis "My_Archives\TagLogs".

Speicherort Lokales Dateisystem:

- <C:\My_File_Folder\My_Archives\Machine_1>: Speichert das Archiv auf der lokalen Festplatte C: im Unterverzeichnis "My_File_Folder\My_Archives\Machine_1"

Speicherort Netzlaufwerk:

- <\\ArchiveServer\My_File_Folder\My_Archives\Machine_1>: Speichert das Archiv auf dem Server "ArchiveServer" im Unterverzeichnis "My_File_Folder\My_Archives\Machine_1".

Namenskonventionen

In einem Projekt müssen die Namen von Archiven eindeutig sein. Auch wenn für verschiedene Archive unterschiedliche Speicherorte gewählt werden, muss der Name eines Archivs eindeutig sein.

Hinweis

Die im Namen der Datenquelle verwendbaren Zeichen sind abhängig vom Ablageort.

Die Zeichen \ / * ? : " < > | sind bei folgenden Ablageorten nicht erlaubt:

- Datei - RDB
- Datei - CSV (ASCII)
- Datei - TXT (Unicode)

Beim Ablageort "Datenbank" sind folgende Zeichen erlaubt: a-z A-Z 0-9 _ @ # $

Die Zeichen _ @ # $ dürfen nicht als erstes Zeichen des Namens verwendet werden.

Hinweis

Gilt nur für Basic Panels 2nd Generation

Unicode-Zeichen werden für die Archivnamen und die Archivpfade nicht unterstützt.

Datei - CSV (ASCII)

Daten werden in einer CSV-Datei im ASCII-Standard gespeichert.

Wenn Sie die archivierten Daten ohne WinCC Runtime lesen bzw. auswerten wollen, dann verwenden Sie den Ablageort "CSV-Datei".

Hinweis

Beim Ablageort "CSV-Datei" sind doppelte Anführungszeichen oder mehrere Zeichen als Listentrennzeichen nicht zulässig. Die Einstellung für Listentrennzeichen finden Sie unter "Start > Einstellungen > Systemsteuerung > Regions- und Sprachoptionen".

Hinweis

Archivieren von Variablen vom Datentyp "BOOL"

Boolesche Werte werden als Ziffern archiviert:

- 0 (False) entspricht dem Archiveintrag 0
- 1 (True) entspricht dem Archiveintrag -1

Datei - TXT (Unicode)

Daten werden in Unicode gespeichert.

Dieses Dateiformat unterstützt alle in WinCC und WinCC Runtime verwendbaren Zeichen. Zum Bearbeiten benötigen Sie eine Software, die Unicode speichern kann, z. B. Notepad.

Hinweis

Um asiatische Sprachen zu archivieren, verwenden Sie als Ablageort "Datei - TXT (Unicode)".

Datei - RDB

Daten werden mit Schnellzugriff in einer proprietären Datenbank gespeichert.

Wenn Sie höchste Lese- Performance in Runtime benötigen, dann verwenden Sie den Ablageort "RDB-Datei".

Datenbank

Daten werden in einer Datenbank gespeichert, für die ein ODBC-Zugriff durch den Administrator des PCs eingerichtet wurde.

Archive mit Prüfsumme (Audit Trail)

In besonderen Fällen werden folgende Dateien erzeugt:

***.keep**

1. Wenn ein Archiv ohne Prüfsumme begonnen und mit einer Prüfsumme fortgeführt wird.
2. Wenn Sie WinCC durch einen Service Pack oder eine neue Version aktualisieren und der Audit Trail bzw. das Archiv mit Prüfsumme fortgeschrieben wird.

Der Inhalt der keep-Datei bleibt gegenüber der ursprünglichen csv-Datei bzw. txt-Datei unverändert.

***.bak**

Wenn WinCC Runtime ein gravierendes, irreguläres Problem innerhalb der Datei festgestellt hat.

Siehe auch

Grundlagen zum Archivieren von Meldungen (Seite 921)

Grundlagen zu Archiven (Seite 1071)

4.2 Arbeiten mit Archiven

4.2.1 Grundlagen zu Archiven

Einleitung

WinCC stellt Ihnen für die Archivierung von Prozessdaten in Runtime Professional folgende Archivarten zur Verfügung:

- Variablenarchiv Fast
- Variablenarchiv Slow
- Meldearchive

Variablenarchive dienen zur Archivierung von Prozessdaten aus einer industriellen Anlage.

Meldearchive dienen zur Archivierung von auftretenden Meldungen in dem beobachteten Prozess.

Prinzip

Der Aufbau und die Funktionsweise der Archivarten ist weitgehend gleich. Die Projektierung wird dadurch sehr einfach und übersichtlich. Bei allen drei Archivarten legen Sie die gleichen Eigenschaften für die Archivart fest.

Alle drei Archivarten unterteilen Sie bei Bedarf in einzelne Segmente. Die Segmentierung kann größenbasiert oder zeitbasiert erfolgen.

Für Variablenarchive stehen Ihnen zwei Archivtypen zur Verfügung:

- Variablenarchiv Fast
  Dieser Archivtyp dient zur Archivierung von schnell veränderlichen Prozessdaten mit einer Zykluszeit <= 1 min.
- Variablenarchiv Slow
  Dieser Archivtyp dient zur Archivierung aller Prozessdaten, die nicht im Archivtyp "Variablenarchiv Fast" enthalten sind. Bei der Projektierung können Sie über mehrere Parameter festlegen, welche Variablenwerte in welchem Archivtyp archiviert werden.

Die Einstellungen für Meldearchive sind unabhängig von den Einstellungen der Variablenarchive.

4.2.2 Eigenschaften von Archiven

Einleitung

Die grundlegenden Eigenschaften von Archiven in Runtime Professional projektieren Sie im Dialog "Runtime Einstellungen > Archivierung", z. B. die Segmentgröße. Die in diesem Dialog projektierten Einstellungen gelten für alle Archive im Projekt.

Die speziellen Eigenschaften der einzelnen Archivarten projektieren Sie entweder direkt in der Tabelle des jeweiligen Editors oder in den Eigenschaften des Archivs im Inspektorfenster.

Die logische Strukturierung von Variablenarchiven und Verdichtungsarchiven projektieren Sie im Editor "Archive". Nähere Hinweise hierzu finden Sie unter "Variablenarchivierung (Seite 658)".

Für Meldearchive gibt es keine vergleichbare logische Strukturierung innerhalb des Archivs. Daher existiert in Runtime Professional kein separater Editor "Meldearchive". Sie können lediglich festlegen, ob eine Meldung archiviert wird oder nicht. Die Archivierung von Meldungen projektieren Sie im Editor "HMI-Meldungen" auf der Registerkarte "Meldeklassen". Nähere Hinweise hierzu finden Sie unter "Grundlagen zum Archivieren von Meldungen (Seite 937) ".

Eigenschaften

Für die Segmentierung stehen Ihnen folgende Eigenschaften zur Verfügung:

- Zeitraum über alle Segmente
- Maximale Größe aller Segmente
- Zeitraum, den ein Einzelsegment umfasst
- Maximale Größe eines Einzelsegments
- Zeitpunkt des ersten Segmentwechsels

Zusätzlich können Sie festlegen, wie und wann Backups erstellt werden.
Für den Archivtyp "Variablenarchiv Fast" stehen zusätzliche Eigenschaften zur Verfügung, mit deren Hilfe Sie die Aufteilung der Prozessdaten auf den Archivtyp "Fast" oder "Slow" festlegen. Nähere Hinweise hierzu finden Sie unter "Einstellungen zur Archivierung (Seite 687)".

Siehe auch

Variablenarchivierung (Seite 658)

Grundlagen zum Archivieren von Meldungen (Seite 937)

Einstellungen zur Archivierung (Seite 687)

Grundlagen zu Archiven (Seite 1077)

Variablenarchivierung (Seite 658)

Projektieren der Meldearchivierung (Seite 937)

Zugriff auf Archivdaten über WinCC OLE DB-Provider 5

5.1 WinCC Archive Connector

5.1.1 WinCC Archive Connector

Funkionsweise

Der WinCC "Archive Connector" dient zur Konfiguration des Zugriffs auf die Archivdatenbank.

Das Tool ist Bestandteil von WinCC DataMonitor.

Mit dem Archive Connector können bereits ausgelagerte WinCC-Archive wieder mit einem SQL-Server verbunden werden.

Danach können der DataMonitor-Client oder der WinCC OLE DB-Provider auf die Archive zugreifen.

Funktionen des WinCC Archive Connector:

- Manuelles Verbinden: Lokale Datenbanken können ausgewählt und mit dem lokalen SQL-Server verbunden werden.
- Manuelles Trennen der Verbindung: Verbundene Datenbanken können ausgewählt und die Verbindung mit dem SQL-Server getrennt werden.
- Automatisches Verbinden: Es können lokale Verzeichnisse ausgewählt werden, in denen WinCC-Archive ausgelagert werden. Es werden alle Archive automatisch mit dem SQL-Server verbunden, die in den gewählten Verzeichnissen ab dem Zeitpunkt der Aktivierung der Änderungsüberwachung hinzugefügt werden.

Der Archive Connector kann nur mit einem lokalen SQL-Server und einer Lizenz von WinCC DataMonitor betrieben werden.

Nach dem Abschluss der Konfiguration kann der Archive Connector beendet werden.

Hinweis

Die WinCC RT-Archive im Verzeichnis "<Projektverzeichnis> \ ArchiveManager" und den zugehörigen Unterverzeichnissen dürfen nicht mit dem Archive Connector verbunden oder getrennt werden, da ihre Verbindung mit dem SQL-Server durch das Bediengerät mit WinCC RT Professional verwaltet wird.

Der Pfad der ausgelagerten WinCC-Archive wird in WinCC mit dem Archive Configurator z.B. des Tag Logging projektiert, nicht mit dem WinCC Archive Connector.

Soll auf ausgelagerte Archive zugegriffen werden, die auf Wechselmedien liegen wie z.B. Band- oder MOD-Laufwerke, so ist bei einem Mediumwechsel im Laufwerk zu beachten, dass zuvor die Verbindung zu den Archiven auf diesem Medium mit Hilfe des Archive Connector getrennt wird. Nach dem Mediumwechsel sollte durch den Benutzer im Archive Connector geprüft werden, ob die Archive auf dem neuen Medium verbunden sind.

Die Konfiguration mit dem WinCC Archive Connector sollte nur einem begrenzten Personenkreis möglich sein. Deshalb sollte der Zugriff auf das Tool geschützt werden über das Windows-Benutzerrecht "Administratoren" oder über sonstige Windows-Schutzmechanismen, z. B. Ablage in einem geschützten Verzeichnis.

Hinweis

Verwenden Sie den Archive Connector oder DataMonitor, um auf die verbundenen Archive zuzugreifen.

Sie haben mit folgenden Objekten keinen Zugriff auf die mit dem Archive Connector verbundenen Archive:

- Meldeanzeige
- Tabellenanzeige
- f(t)-Kurvenanzeige
- f(x)-Kurvenanzeige

5.1.2 Registerkarte Konfiguration

Registerkarte Konfiguration

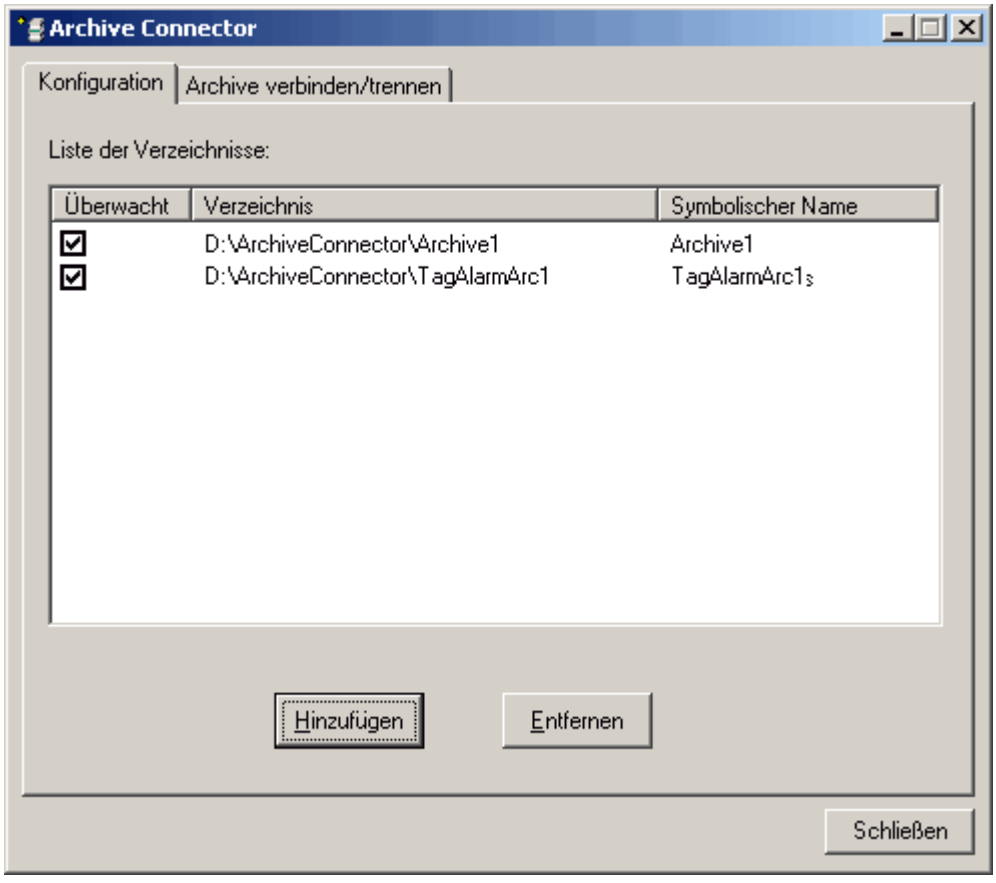

Auf der Registerkarte Konfiguration werden die Archivierungsverzeichnisse angezeigt und verwaltet, auf die der Zugriff über Web oder über den WinCC OLE DB-Provider möglich sein soll.

Über die Schaltflächen können Archivierungsverzeichnisse hinzugefügt oder entfernt werden. Für jedes Archivierungsverzeichnis muss bei der Konfiguration ein symbolischer, eindeutiger Name vergeben werden.

Den symbolischen Namen verwenden der DataMonitor-Client oder der WinCC OLE DB-Provider für den Zugriff auf das Archiv.

Der Name wird auch beim Verwalten und Verbinden ausgelagerter Daten von mehreren Rechnern bzw. Projekten verwendet.

Für den symbolischen Namen dürfen nur in der SQL-Syntax zulässige Zeichen verwendet werden.

Durch Aktivierung des zugehörigen Kontrollkästchens werden alle Archive automatisch mit dem SQL-Server verbunden, die in dem gewählten Verzeichnis ab dem Aktivierungszeitpunkt hinzugefügt werden.

Wenn Sie die Überwachung aktivieren oder deaktivieren, wird die Änderung erst nach dem Schließen des Archive Connector wirksam.

5.1.3 Registerkarte Archive verbinden/trennen

Registerkarte Archive verbinden/trennen

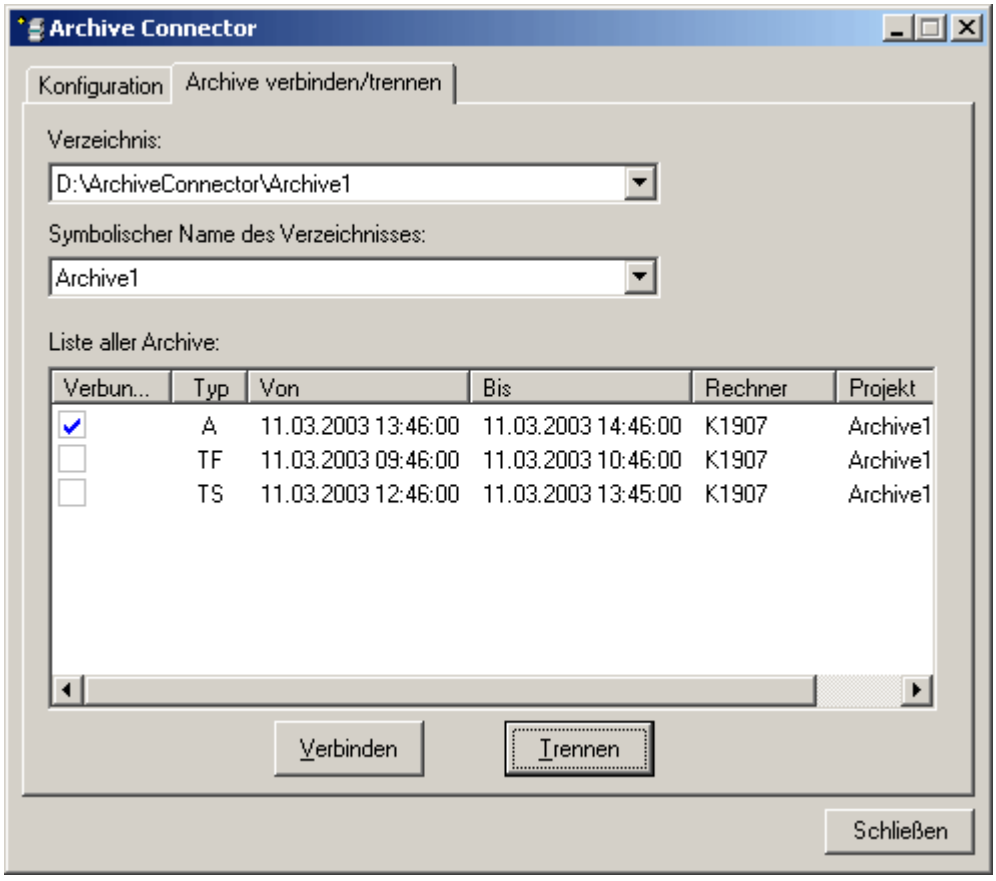

Auf der Registerkarte Archive verbinden/trennen sind alle Archive aufgeführt, die in den Archivierungsverzeichnissen vorhanden sind.

Zu jedem Archiv wird der Verbindungsstatus angezeigt. Die Verbindung zu den Archiven kann über die Schaltflächen aufgebaut oder getrennt werden.

Der Archivtyp wird im Feld "Typ" dargestellt.

- "A" = Alarm Logging;
- "TF" = Tag Logging (Fast);
- "TS"= Tag Logging (Slow).

In den Spalten "Von" und "Bis" erfolgt die Angabe für die lokale Zeitzone.

Hinweis

Das Verbinden mehrerer ausgelagerter Archive mit dem SQL-Server kann einige Sekunden in Anspruch nehmen.

Es ist nicht möglich, eine Datenbankdatei mit dem gleichen Namen zweimal zu verbinden.

Der WinCC Archive Connector verbindet abgeschlossene und über Backup ausgelagerte Archive mit dem SQL-Server. Nicht abgeschlossene Archive werden nicht unterstützt.

Die Oberflächen-Sprache des Archive Connectors ist abhängig von den Einstellungen in den Regions- und Sprachoptionen von Windows.

Hinweis

Mit SQL Server 2000 erzeugte Archive können Sie nur dann verbinden, wenn Sie den Schreibschutz des Archivs entfernen. Archive, die einmal verbunden sind, können Sie nicht mehr mit dem Archive Connector unter SQL Server 2000 verbinden.

Um auf CD oder DVD ausgelagerte Archive zu verbinden, kopieren Sie die Archive auf einen Datenträger, auf dem Sie den Schreibschutz entfernen können.

5.2 Zugriff auf Archivdaten über OLE DB-Provider

5.2.1 Möglichkeiten des Zugriffs auf Archivdaten

Einführung

Mit OLE DB haben Sie folgende Möglichkeiten, auf WinCC-Archivdaten zuzugreifen und diese über eine externe Schnittstelle anzuzeigen.

Zugriff mit WinCC OLE DB-Provider

Mit WinCC OLE DB können Sie auf alle WinCC-Archivdaten zugreifen.

In Abhängigkeit von der Projektierung werden Prozessdaten von WinCC komprimiert abgelegt. Der WinCC OLE DB-Provider erlaubt einen transparenten Zugriff auch auf diese Daten.

Um die Vorteile von Standard-SQL-Abfragen zu nutzen, verwenden Sie den "SQL Server Import / Export Wizard". Die entpackten Daten können Sie mit Hilfe des Wizards in einer Zwischendatenbank ablegen, auf die Sie mit Standard-SQL-Abfragen zugreifen können.

Hinweis

Wenn WinCC ein volles Archiv schließt und ein neues Archiv öffnet, werden über den OLE DB-Provider kurzzeitig keine Daten aus Melde- und Prozesswertarchiven gelesen.

Zugriff mit Microsoft OLE DB

Mit Microsoft OLE DB können Sie auf WinCC User Archives zugreifen.

Hinweis

Mit Microsoft OLE DB ist nur der Zugriff auf WinCC User Archives getestet und freigegeben, nicht jedoch auf Melde- und Prozesswertarchive.
Verwenden Sie für den Zugriff auf Melde- und Prozesswertarchive den WinCC OLE DB-Provider.

5.2.2 Projektierung des Zugriffs auf Archivdaten

Möglichkeiten der Projektierung

Zum Zugriff auf die Datenbanken mit WinCC OLE DB können Sie eigene Applikationen schreiben.

Für die Kommunikation mit dem WinCC OLE DB-Provider wird ADO DB in Applikationen eingesetzt, die z.B. mit Visual Basic, VBScript oder VBA erstellt wurden.

Hinweis

Sonderzeichen im Variablennamen

Beachten Sie bei Variablennamen, dass Programmiersprachen wie Visual Basic, VBScript oder VBA nur folgende Zeichen im Variablennamen erlauben: "A...Z", "a...z", "0...9" sowie "_".

Wenn Sie in WinCC in Variablennamen Sonderzeichen wie "," oder ";" verwenden, wird das Skript mit Fehlermeldung abgebrochen. Verwenden Sie in diesem Fall die "Variablen-ID", um eine Variable mit Sonderzeichen im Namen im Skript anzusprechen.

Voraussetzungen

- Für den Zugriff auf Archivdaten muss auf dem zugreifenden Rechner WinCC RT Professional oder WinCC V12 installiert sein.
- Bei ausgelagerten Archiven stellen Sie die Verbindung zwischen der SQL-Datenbank und den ausgelagerten Archiven mit dem WinCC Archive Connector her.

  Hinweis

  Die WinCC Runtime-Archive im Verzeichnis "<Projektverzeichnis> \ ArchiveManager" und den zugehörigen Unterverzeichnissen dürfen nicht mit dem Archive Connector verbunden oder getrennt werden, da ihre Verbindung mit dem SQL-Server durch WinCC RT Professional verwaltet wird.

Vorgehensweise

1. Stellen Sie die Verbindung zur Datenbank her, z.B. über MS Excel oder eine eigene Applikation.
2. Legen Sie die gewünschten Selektionskriterien fest.
3. Lesen Sie die Archivdaten aus.
   Das Abfrageergebnis können Sie z.B. in MS Excel darstellen oder als Csv-Datei exportieren.

5.2.3 Verbindung zur Archivdatenbank herstellen

Einleitung

Die Verbindung zwischen der Anwendung und der Archivdatenbank wird bei ActiveX-Datenobjekten (ADO) über das Connection-Objekt hergestellt. Ein wichtiger Parameter ist dabei der ConnectionString. Der ConnectionString enthält alle benötigten Angaben für den Zugriff über den OLE DB-Provider auf die Datenbank.

Aufbau des ConnectionString

```
"Provider = Name des OLE DB-Providers; Catalog=Datenbankname;Data
Source=Servername;"
```

Parameter	Beschreibung
Provider	Name des OLE DB-Providers: z.B. WinCCOLEDBProvider
Catalog	Name der WinCC-Datenbank Bei WinCC RT-Datenbanken verwenden Sie die Datenbanknamen, die mit "R" enden. <Datenbankname_R>. Es kann auch die Datenbank "CC_ExternalBrowsing" benutzt werden. Wenn Sie über den WinCC Archive Connector ausgelagerte WinCC-Archive mit dem SQL-Server verbunden haben, verwenden Sie den symbolischen Namen. **Hinweis** Geben Sie für den transparenten Zugriff für "Catalog" den Namen des WinCC-Projekts an, z. B.: "Catalog=WinCC_Project_Name". **Hinweis** Wenn Sie mit "CC_ExternalBrowsing" auf Meldearchive oder ausgelagerte Archive zugreifen, kann der Zugriff mehrere Minuten dauern.
Data Source	Servername Lokal: ".\WinCC" oder "<Rechnername>\WinCC" Remote: "<Rechnername>\WinCC" **Hinweis** Geben Sie beim transparenten Zugriff auf den Central Archive Server oder bei redundanten Servern über den OLE DB-Provider für "Data Source" Folgendes ein: <Symbolischer Rechnername>::\WinCC. **Hinweis** Wenn Sie direkt auf eine Archivvariable auf dem Central Archive Server "CAS" zugreifen, dann verwenden Sie den Namen der Archivvariablen. Der Central Archive Server "CAS" liefert als ID die CAS-ID zurück und nicht die ID der Archivvariablen: <SYMBOLISCHER RECHNERNAME>\\<Achive_Var_Name>

Beispiel Prozesswert- oder Meldearchiv:

Im folgenden Beispiel wird ein Connection-Objekt erzeugt und anschließend die Verbindung zur WinCC-Datenbank (Prozesswert- oder Meldearchiv) geöffnet.

```
Set conn = CreateObject("ADODB.Connection")
```

```
conn.open "Provider=WinCCOLEDBProvider.
1;Catalog=CC_OpenArch_03_05_27_14_11_46R;Data Source=.\WinCC"
```

Beispiel Anwenderarchiv:

Im folgenden Beispiel wird ein Connection-Objekt erzeugt und anschließend die Verbindung zum WinCC-Anwenderarchiv geöffnet.

```
Set conn = CreateObject("ADODB.Connection")
```

```
conn.open "Provider=SQLOLEDB.1; Integrated Security=SSPI; Persist
Security Info=false; Initial Catalog=CC_OpenArch_03_05_27_14_11_46R;
Data Source=.\WinCC"
```

Hinweis

Um die Performance bei lokalem Zugriff zu verbessern, tragen Sie bei Data Source "<Rechnername>\WinCC" ein anstelle von ".\WinCC" .

5.3 Anwendungsfälle

5.3.1 Anwendungsfall 1: Lokaler Zugriff auf WinCC RT-Datenbank

Voraussetzungen

Auf dem Bediengerät müssen folgende Lizenzen installiert sein:

- WinCC RT Professional

Prinzip

Eine Anwendung greift über den WinCC OLE DB-Provider auf die lokale WinCC Runtime-Datenbank zu.

Sie können lokal die Archivdaten auswerten und beispielsweise berechnen, wie groß die Standardabweichung eines Prozesswerts ist.

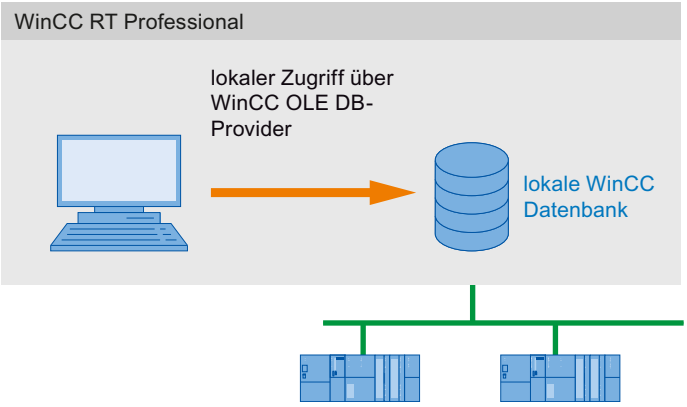

5.3.2 Anwendungsfall 2: Remote-Zugriff auf WinCC RT-Datenbank

Voraussetzungen

Auf dem Bediengerät müssen folgende Lizenzen installiert sein:

- WinCC RT Profesional

Der Zugriff kann unter verschiedenen Konstellationen des Connectivity Pack-Client erfolgen.

- Auf dem Client-Rechner ist eine WinCC-Software wie z.B. WinCC V12, Web Navigator-Server, DataMonitor-Server vorhanden.
  Der Client muss nicht explizit installiert werden. Die Lizenzierung erfolgt durch die WinCC-Lizenzen.

Prinzip

Der Client greift per Remote auf die WinCC Runtime-Datenbank eines Bediengeräts mit WinCC RT Professional zu. Über den WinCC OLE DB-Provider liest der Client die Daten der Prozesswert- und Meldearchive.

Da in diesem Anwendungsfall nicht auf ausgelagerte WinCC-Archive zugegriffen wird, müssen mit dem Archive Connector keine WinCC-Archive mit einem SQL-Server verbunden werden.

Sie können auf dem Client die Daten anzeigen, auswerten oder weiterverarbeiten z.B. durch Export in eine CSV-Datei.

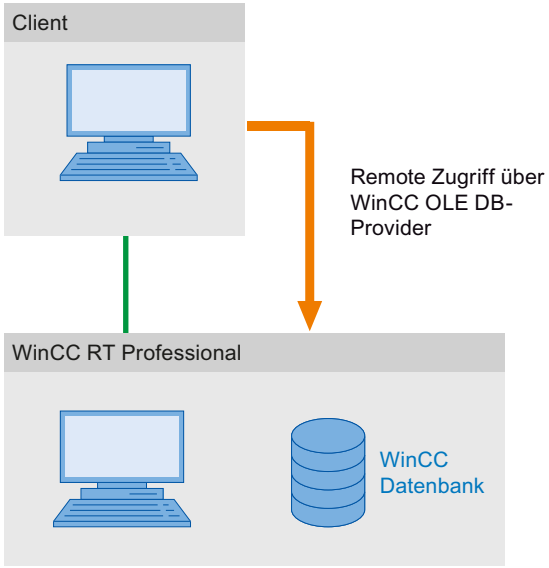

5.3.3 Anwendungsfall 3: Lokaler Zugriff auf WinCC-Archivdatenbanken

Voraussetzungen

Das Bediengerät benötigt folgende Installationen:

- WinCC V12
- WinCC RT Professional

Prinzip

Eine Anwendung greift über den WinCC OLE DB-Provider auf die lokale Archivdatenbank zu. Die älteren Archivdaten werden aus der WinCC RT-Datenbank in ein separates Verzeichnis auf dem gleichen Rechner kopiert.

Mit dem Archive Connector werden ausgelagerte WinCC-Archive wieder mit einem SQL-Server verbunden. Damit stehen die Archive für den Zugriff mit dem WinCC OLE DB-Provider zur Verfügung.

Sie können die lokalen Archivdaten anzeigen, durchsuchen oder auswerten, z.B. um Verfahrensfehler zu suchen oder Prozesse zu optimieren.

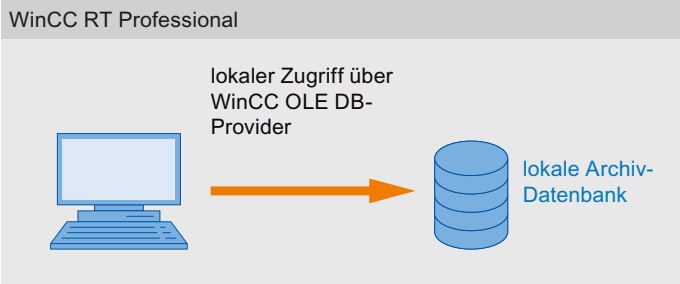

5.3.4 Anwendungsfall 4: Remote-Zugriff auf WinCC-Archivdatenbanken

Software-Voraussetzungen

Der Langzeitarchiv-Server benötigt folgende Installationen:

- Connectivity Pack-Server

Der Zugriff kann unter verschiedenen Konstellationen des Clients erfolgen.

- Auf dem Client-Rechner ist eine WinCC-Software wie z.B. WinCC V12 RT Professional, WinCC V12, Web Navigator-Server, DataMonitor-Server vorhanden.
  Der Client muss nicht explizit installiert werden.
  Die Lizenzierung erfolgt durch die WinCC-Lizenzen.

Prinzip

Einen Langzeitarchiv-Server setzen Sie ein, um die Datenbanken von Prozesswert- und Meldearchiven z.B. monatlich in einem Backup zu sichern.

Mit dem Archive Connector werden ausgelagerte WinCC-Archive wieder mit einem SQL-Server verbunden. Damit stehen die Archive für den Zugriff mit dem WinCC OLE DB-Provider zur Verfügung.

Der Client greift über den WinCC OLE DB-Provider auf die Archive zu. Sie können z.B. mit einer VB-Applikation die Archive auswerten und die Prozesswerte eines bestimmten Tages anzeigen lassen.

Hinweis

Wenn Sie mit "CC_ExternalBrowsing" auf Meldearchive oder ausgelagerte Archive zugreifen, kann der Zugriff mehrere Minuten dauern.

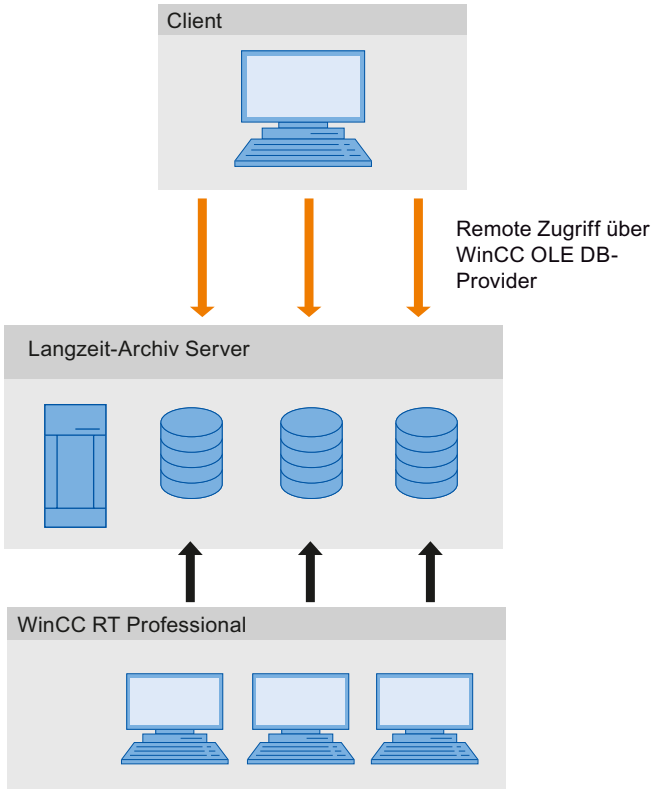

5.3.5 Anwendungsfall 5: Lokaler Zugriff auf WinCC-Anwenderarchive

Voraussetzungen

Das Bediengerät benötigt folgende Installationen:

- WinCC RT Professional

Prinzip

Eine Anwendung greift über den Microsoft OLE DB-Provider auf die lokalen WinCC-Anwenderarchive zu.

Sie können z. B. mit einer VB-Applikation, die lokalen Archivdaten anzeigen, durchsuchen und geänderte Werte zurückschreiben.

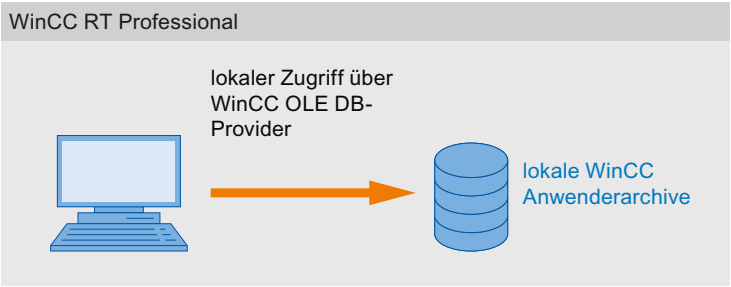

5.3.6 Anwendungsfall 6: Remote-Zugriff auf WinCC-Anwenderarchive

Voraussetzungen

Das Bediengerät benötigt folgende Installationen:

- WinCC RT Professional

Der Zugriff kann unter verschiedenen Konstellationen des Connectivity Pack-Clients erfolgen.

- Auf dem Client-Rechner ist eine WinCC-Software wie z.B. WinCC RT Professional, Web Navigator-Server, DataMonitor-Server vorhanden.
  Der Client muss nicht explizit installiert werden. Die Lizenzierung erfolgt durch die WinCC-Lizenzen.

Prinzip

Der Client greift über den MS OLE DB-Provider auf die WinCC-Anwenderarchive zu.

Sie können z.B. mit einer VB-Applikation die Archivdaten anzeigen, durchsuchen und geänderte Werte zurückschreiben.

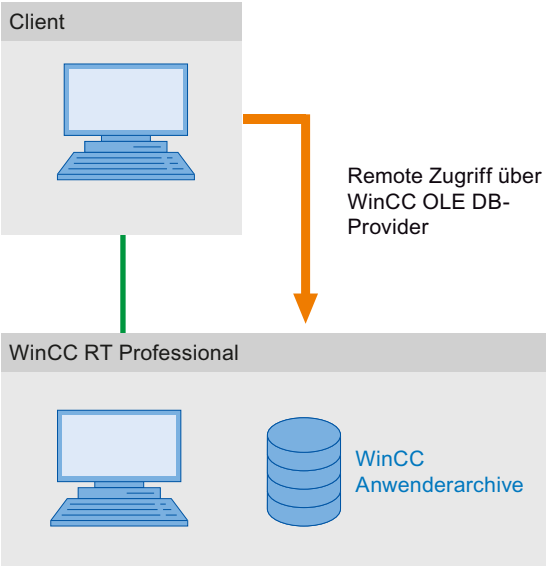

5.4 Abfragen der Archivdaten

5.4.1 Darstellung der Prozesswertarchive

Einleitung

Das Abfrageergebnis wird als Recordset zurückgegeben. In diesem Kapitel wird der Aufbau des Recordsets für Prozesswertarchive beschrieben.

Recordset-Aufbau

Feldname	Typ	Kommentar
ValueID	Integer 4 Byte oder Integer 8 Byte	Eindeutige Identifikation des Wertes. Die Länge hängt von der Art der Abfrage ab.
TimeStamp	Datetime	Zeitstempel
RealValue	Real 8 Byte	Variablenwert
Quality	Integer 4 Byte	QualityCode des Werts (z.B. "gut" oder "schlecht")
Flags	Integer 4 Byte	Interner Steuerungsparameter

5.4.2 Abfrage für Prozesswertarchive

Prinzip

Mit der folgenden Abfrage können Sie auf ein Prozesswertarchiv zugreifen. Über Filterkriterien können die Daten selektiert werden. Die Abfragen werden mit dem Command-Objekt an die Datenbank übergeben.

Hinweis

Die Länge der ValueID kann unterschiedlich sein.

- Bei Datenbanken, die auf einem zentralen Archivserver (CAS) verarbeitet wurden, ist die ValueID 8 Byte lang und enthält im HI-DWORD-Bereich eine Serverkennung sowie im LO-DWORD-Bereich die vom jeweiligen Server vergebene ValueID.
- Bei allen anderen Datenbanken ist die ValueID 4 Byte lang und enthält lediglich die vom WinCC Server vergebene, eindeutige ValueID.

Die 4-Byte-Abfrage über TAG:R ist zur Kompatibilität in jedem Fall weiterhin möglich. Die hier zurückgelieferte 4-Byte-ValueID ist aber im Fall von CAS-Datenbanken nicht mehr eindeutig.

Abfragen auf Prozesswertarchive sind begrenzt auf max. 20 Variablen mit jeweils max. 128 Zeichen pro Variable.

Syntax

Beachten Sie, dass die Abfrage keine Leerzeichen enthalten darf.

Abfrage von 8 Byte langen ValueIDs:

```
TAG_LLVID:R,<ValueID oder
ValueName>,<TimeBegin>,<TimeEnd>[,<SQL_clause>][,<TimeStep>]
```

Abfrage von 4 Byte langen ValueIDs:

```
TAG:R,<ValueID oder ValueName>,<TimeBegin>,<TimeEnd>[,<SQL_clause>]
[,<TimeStep>]
```

Parameter

Parameter	Beschreibung
ValueID	ValueID aus der Datenbank-Tabelle Mehrfachnennungen sind möglich, z.B. "TAG:R,(ValueID_1;ValueID_2;ValueID_x),<TimeBegin>,<TimeEnd>"
ValueName	ValueName im Format 'ArchiveName\Value_Name'. Der Parameter <ValueName> muss in einfache Anführungszeichen gesetzt werden. Mehrfachnennungen sind möglich, z.B. "TAG:R,('ValueName_1';'ValueName_2';'ValueName_x'), <TimeBegin>,<TimeEnd>" **Hinweis** Beachten Sie bei Variablennamen, dass Programmiersprachen wie Visual Basic, VBScript oder VBA nur folgende Zeichen im Variablennamen erlauben: "A...Z", "a...z", "0...9" sowie "_". Wenn Sie in WinCC in Variablennamen Sonderzeichen wie "," oder ";" verwenden, wird das Skript mit Fehlermeldung abgebrochen. Verwenden Sie in diesem Fall die "Variablen-ID", um eine Variable mit Sonderzeichen im Namen im Skript anzusprechen.
TimeBegin	Startzeitpunkt im Format 'YYYY-MM-DD hh:mm:ss.msc' Bei der Verwendung von <TimeStep> muss <TimeBegin> als absolute Zeit angegeben werden. Es ist keine relative Angabe oder "0000-00-00 00:00:00.000" möglich.
TimeEnd	Endzeitpunkt im Format 'YYYY-MM-DD hh:mm:ss.msc'
SQL_Clause	Filterkriterium in SQL-Syntax: [WHERE search_condition] [ORDER BY {order_expression [ASC|DESC] }] Das Kriterium "ORDER BY" kann nur mit angegebener Sortierfolge "{order_expression [ASC|DESC] }" benutzt werden. Beispiel: Die folgende Abfrage liefert alle Werte der Variablen "ValueName_1" und "ValueName_2" zurück, die unter 50 oder über 100 liegen. "TAG:R,('ValueName_1';'ValueName_2'),<TimeBegin>,<TimeEnd>, 'WHERE RealValue > 100 OR RealValue < 50'"

Parameter	Beschreibung		
TimeStep	Werte im angegebenen Zeitintervall werden zusammengefasst, beginnend mit dem Startzeitpunkt <Time-Begin> Format: 'TIMESTEP=x,y' x = Intervall in Sekunden y = Aggregationstyp, definiert das Intervall-Ergebnis Folgende Werte für den Aggregationstyp sind möglich:		
	Ohne Interpolation	Mit Interpolation	Bedeutung
	1 (FIRST) 2 (LAST) 3 (MIN) 4 (MAX) 5 (AVG) 6 (SUM) 7 (COUNT)	257 (FIRST_INTERPOLATED) 258 (LAST_INTERPOLATED) 259 (MIN_INTERPOLATED) 260 (MAX_INTERPOLATED) 261 (AVG_INTERPOLATED) 262 (SUM_INTERPOLATED) 263 (COUNT_INTERPOLATED)	Erster Wert Letzter Wert Minimalwert Maximalwert Mittelwert Summe Anzahl der Werte
	Ohne Interpolation bedeutet: Wenn im Intervall keine Werte vorhanden sind, wird kein Intervall-Ergebnis zurückgegeben. Mit Interpolation bedeutet: Wenn im Intervall keine Werte vorhanden sind, wird der Wert durch lineare Interpolation aus den Ergebnissen der nächsten nichtleeren Nachbarintervalle gebildet. Es erfolgt keine Extrapolation. Beispiel: Bei TIMESTEP=60,257 wird für jedes Intervall von 60 Sekunden der erste Wert dieses Intervalls oder - wenn es in dem Intervall keine Werte gibt - der linear interpolierte Wert aus den ersten Werten der Nachbarintervalle zurückgeliefert. "TAG:R,1,'2013-07-09 09:03:00.000','0000-00-00 00:10:00.000','TIMESTEP=60,257'"		

Hinweis

<TimeBegin> und <TimeEnd> dürfen nicht beide "ZERO" = '0000-00-00 00:00:00.000' sein.

Um die Performance zu verbessern, verwenden Sie in der Abfrage den Parameter "ValueID" anstelle von "ValueName". Die "ValueID" können Sie aus der Tabelle "Archive" ermitteln.

Einige Anwendungen können bei Prozesswerten die Zeit nicht in 1-ms-Schritte auflösen, was zu Ungenauigkeiten führen kann.
Ein Beispiel zur Auflösung der Millisekunden aus dem Zeitstempel von Prozesswerten finden Sie im VB-Skript "SplitDateTimeAndMs" im Kapitel "Beispiel: Prozesswertarchiv mit dem WinCC OLE DB-Provider lesen". Das Skript ist auch im Demoprojekt "OpConPack" implementiert.

Auswahl eines absoluten Zeitraums

Lesen vom Startzeitpunkt <TimeBegin> bis zum Endzeitpunkt <TimeEnd>.

Beispiel A1:

Liest Werte der ValueID 1 vom Startzeitpunkt 9:03 Uhr bis zum Endzeitpunkt 9:10 Uhr.

```
"TAG:R,1,'2013-07-09 09:03:00.000','2013-07-09 09:10:00.000'"
```

Auswahl eines relativen Zeitraums

Lesen vom Beginn der Aufzeichnung:

```
<TimeBegin> = '0000-00-00 00:00:00.000'
```

Lesen bis zum Ende der Aufzeichnung:

```
<TimeEnd> = '0000-00-00 00:00:00.000'
```

<TimeBegin> und <TimeEnd> dürfen jedoch nicht beide "ZERO" = '0000-00-00 00:00:00.000' sein.

Hinweis

Geben Sie einen relativen Zeitraum, den Sie aus einer verbundenen Archivdatenbank abfragen wollen, im folgenden Format an:

- 0000-00-DD hh:mm:ss.msc

Bei der Angabe eines Zeitraums in Monaten können die Inhalte fehlerhaft sein, da ein Monat zwischen 28 und 31 Tagen hat.

Beispiel B1:

Liest ab dem absoluten Zeitpunkt von "TimeBegin" bis zum Ende der Aufzeichnung, also dem letzten archivierten Wert.

```
<TimeBegin> = '2013-02-02 12:00:00.000', <TimeEnd> = '0000-00-00
00:00:00.000'
```

Beispiel B2:

Liest ab dem absoluten Zeitpunkt von "TimeBegin" 10 Sekunden weiter.

```
<TimeBegin> = '2013-02-02 12:00:00.000', <TimeEnd> = '0000-00-00
00:00:10.000'
```

Beispiel B3:

Liest 10 Sekunden zurück, ab dem absoluten Zeitpunkt von "TimeEnd".

```
<TimeBegin> = '0000-00-00 00:00:10.000', <TimeEnd> = '2013-02-02
12:00:00.000'
```

Beispiel B4:

Liest für mehrere ValueIDs (1;3;5;6) die Werte der letzten Stunde, von der Zeit des letzten archivierten Werts ausgehend.

```
"TAG:R,(1;3;5;6),'0000-00-00 01:00:00.000','0000-00-00
00:00:00.000'"
```

Beispiel B5:

Liest für die Variable "TAG_2" aus dem Archiv "ArTags" die Werte der letzten fünf Minuten, von der Zeit des letzten archivierten Werts ausgehend.

```
"TAG:R,'ArTags\TAG_2','0000-00-00 00:05:00.000','0000-00-00
00:00:00.000'"
```

Das folgende Bild zeigt ein mögliches Ergebnis dieses Beispiels. Die Abfrage erfolgte im Connectivity Pack Demo-Projekt.

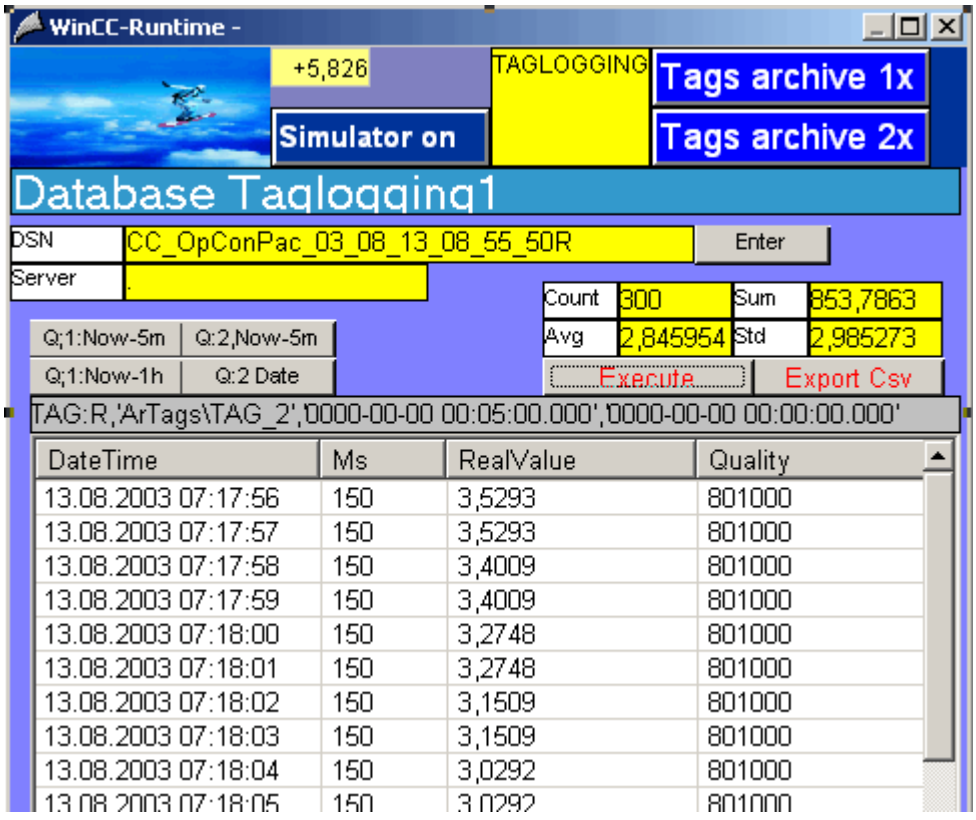

Mehrere Rückgabewerte zu einer Abfrage durch Filter auf Variablenwert

Beispiel C1:

Die folgende Abfrage benutzt zusätzlich den Parameter <SQL_Clause> und liefert alle Werte der Variablen mit der ValueID "3" und "6" zurück, die unter 50 oder über 100 liegen.

```
"TAG:R,(3;6),<TimeBegin>,<TimeEnd>,'WHERE RealValue > 100 OR
RealValue < 50'"
```

Abfrage mit Parameter <TimeStep>

Beispiel C2:

Die folgende Abfrage benutzt den Parameter <TimeStep> und liefert alle Werte der ValueID "1" zurück, vom Startzeitpunkt "TimeBegin" bis 5 Minuten später, in Intervallen von "60" Sekunden mit dem Aggregationstyp "5" = "Mittelwert ohne Interpolation".

```
"TAG:R,1,'2004-10-13 17:00:00.000','0000-00-00 00:05:00.000',
'TIMESTEP=60,5'"
```

Die folgende Grafik zeigt das Abfrageergebnis. In der linken Tabelle sind die Archivdaten dargestellt, die mit einem Archivierungszyklus von 30 Sekunden archiviert wurden. In der rechten Tabelle ist das Abfrageergebnis dargestellt. Es wird der Mittelwert aus jeweils zwei

Archivwerten der Sekunde "0" und der Sekunde "30" gebildet und mit dem ersten Zeitstempel des Berechnungszeitraums, der Sekunde "0", angezeigt.

TIMESTEP = 60,5

ValueID	Datetime	RealValue	Quality
1	2004-10-13 17:00:00.000	3.0	801000
1	2004-10-13 17:00:30.000	4.0	801000
1	2004-10-13 17:01:00.000	2.0	801000
1	2004-10-13 17:01:30.000	4.0	801000
1	2004-10-13 17:02:00.000	4.0	801000
1	2004-10-13 17:02:30.000	1.0	801000
1	2004-10-13 17:03:00.000	3.0	801000
1	2004-10-13 17:03:30.000	3.0	801000
1	2004-10-13 17:04:00.000	1.0	801000
1	2004-10-13 17:04:30.000	0.0	801000
1	2004-10-13 17:05:00.000	3.0	801000

ValueID	Datetime	RealValue	Quality
1	2004-10-13 17:00:00.000	3.5	801000
1	2004-10-13 17:01:00.000	3.0	801000
1	2004-10-13 17:02:00.000	2.5	801000
1	2004-10-13 17:03:00.000	3.0	801000
1	2004-10-13 17:04:00.000	0.5	801000
1	2004-10-13 17:05:00.000	3.0	801000

Beispiel C3:

Die folgende Abfrage benutzt den Parameter <TimeStep> und liefert alle Werte der ValueIDs "1" und "2" zurück, vom Startzeitpunkt "TimeBegin" bis 2 Minuten später, in Intervallen von "15" Sekunden mit dem Aggregationstyp "261" = "Mittelwert mit linearer Interpolation".

```
"TAG:R,(1;2),'2004-10-13 17:00:00.000','0000-00-00 00:02:00.000',
'TIMESTEP=15,261'"
```

Die folgende Grafik zeigt das Abfrageergebnis. In der linken Tabelle sind die Archivdaten dargestellt, die mit einem Archivierungszyklus von 30 Sekunden archiviert wurden. In der rechten Tabelle ist das Abfrageergebnis dargestellt. Die Archivwerte der Sekunden "0" und "30" werden unverändert mit ihrem Zeitstempel im Abfrageergebnis angezeigt. Für die Sekunde "15" wird der linear interpolierte Wert aus den Archivwerten der Sekunden "0" und "30" gebildet. Für die Sekunde "45" wird der linear interpolierte Wert aus den Archivwerten der Sekunde "30" der gleichen Minute und der Sekunde "0" der folgenden Minute gebildet.

TIMESTEP = 15, 261

ValueID	Datetime	RealValue	Quality
1	2004-10-13 17:00:00.000	3,0	801000
2	2004-10-13 17:00:00.000	12,0	801000
1	2004-10-13 17:00:30.000	4,0	801000
2	2004-10-13 17:00:30.000	14,0	801000
1	2004-10-13 17:01:00.000	2,0	801000
2	2004-10-13 17:01:00.000	16,0	801000
1	2004-10-13 17:01:30.000	4,0	801000
2	2004-10-13 17:01:30.000	17,0	801000
1	2004-10-13 17:02:00.000	4,0	801000
2	2004-10-13 17:02:00.000	14,0	801000
1	2004-10-13 17:02:30.000	1,0	801000
2	2004-10-13 17:02:30.000	13,0	801000
1	2004-10-13 17:03:00.000	3,0	801000
2	2004-10-13 17:03:00.000	13,0	801000

ValueID	Datetime	RealValue	Quality
1	2004-10-13 17:00:00.000	3,0	801000
1	2004-10-13 17:00:15.000	3,5	801000
1	2004-10-13 17:00:30.000	4,0	801000
1	2004-10-13 17:00:45.000	3,0	801000
1	2004-10-13 17:01:00.000	2,0	801000
1	2004-10-13 17:01:15.000	3,0	801000
1	2004-10-13 17:01:30.000	4,0	801000
1	2004-10-13 17:01:45.000	4,0	801000
1	2004-10-13 17:02:00.000	4,0	801000
2	2004-10-13 17:00:00.000	12,0	801000
2	2004-10-13 17:00:15.000	13,0	801000
2	2004-10-13 17:00:30.000	14,0	801000
2	2004-10-13 17:00:45.000	15,0	801000
2	2004-10-13 17:01:00.000	16,0	801000
2	2004-10-13 17:01:15.000	16,5	801000
2	2004-10-13 17:01:30.000	17,0	801000
2	2004-10-13 17:01:45.000	15,5	801000
2	2004-10-13 17:02:00.000	14,0	801000

5.4.3 Abfrage für Meldearchive

Einleitung

Mit der folgenden Abfrage können Sie auf das Meldearchiv zugreifen. Über Filterkriterien können die Daten selektiert werden. Die Abfrage wird mit dem Command-Objekt an die Datenbank übergeben.

Informationen zum Status von Meldungen finden Sie im WinCC Information System unter "Arbeiten mit WinCC > ANSI-C zum Erstellen von Funktionen und Aktionen > ANSI-C Funktionsbeschreibungen > Anhang > Strukturdefinitionen > Strukturdefinition MSG_RTDATA_STRUCT".

Bei Abfragen auf die Meldearchive wird das Ergebnis archivweise zusammengefasst, aber es erfolgt keine Sortierung über alle abgefragten Archivsegmente. Soll eine Sortierung erfolgen, so ist die Filterbedingung entsprechend zu erweitern z. B. für zeitfolgerichtige Sortierung mit "ORDER BY DateTime ASC, MS ASC".

Syntax

```
ALARMVIEW:SELECT * FROM <ViewName>[WHERE <Condition>...., optional]
```

Parameter

Parameter	Beschreibung
ViewName	Name der Datenbanktabelle. Die Tabelle muss in der gewünschten Sprache angegeben werden. Der "ViewName" für die fünf europäischen Sprachen lautet beispielweise: ALGVIEWDEU: deutsche Meldearchivdaten ALGVIEWENU: englische Meldearchivdaten ALGVIEWESP: spanische Meldearchivdaten ALGVIEWFRA: französische Meldearchivdaten ALGVIEWITA: italienische Meldearchivdaten Der "ViewName" für die asiatischen Sprachen lautet: ALGVIEWCHT: chinesische (simplified) Meldearchivdaten ALGVIEWCHT: chinesische (traditionell) Meldearchivdaten ALGVIEWJPN: japanische Meldearchivdaten ALGVIEWKOR: koreanische Meldearchivdaten Hinweis Es werden die Sprachen unterstützt, die im WinCC-Basissystem installiert bzw. in der WinCC Text Library projektiert sind. Informationen über die möglichen Abfrage-Sprachen bzw. den entsprechenden "ViewName" finden Sie im SQL-Server in den verbundenen Meldearchiven unter "Views". Darin werden alle Sprachen mit ihren Kennungen z.B. "ALGVIEWENU" angezeigt, die im jeweiligen Archiv unterstützt werden.
Condition	Filterkriterium z.B.: DateTime>'2003-06-01' AND DateTime<'2003-07-01' DateTime>'2003-06-01 17:30:00' MsgNr = 5 MsgNr in (4, 5) State = 2 Bei DateTime können nur absolute Zeitangaben verwendet werden.

Beispiel 1:
Liest alle Einträge der Meldung Nummer 5, die nach dem 05.07.2003 erfasst wurden.

```
"ALARMVIEW:SELECT * FROM ALGVIEWENU WHERE MsgNr = 5 AND
DateTime>'2003-07-05'"
```

Beispiel 2:
Liest alle Meldungen mit dem Zeitstempel zwischen 03.07.2003 und 05.07.2003.

```
"ALARMVIEW:SELECT * FROM ALGVIEWENU WHERE DateTime>'2003-07-03' AND
DateTime<'2003-07-05'"
```

Das nachfolgende Bild zeigt ein mögliches Ergebnis dieses Beispiels. Die Abfrage erfolgte im Connectivity Pack Demo-Projekt.

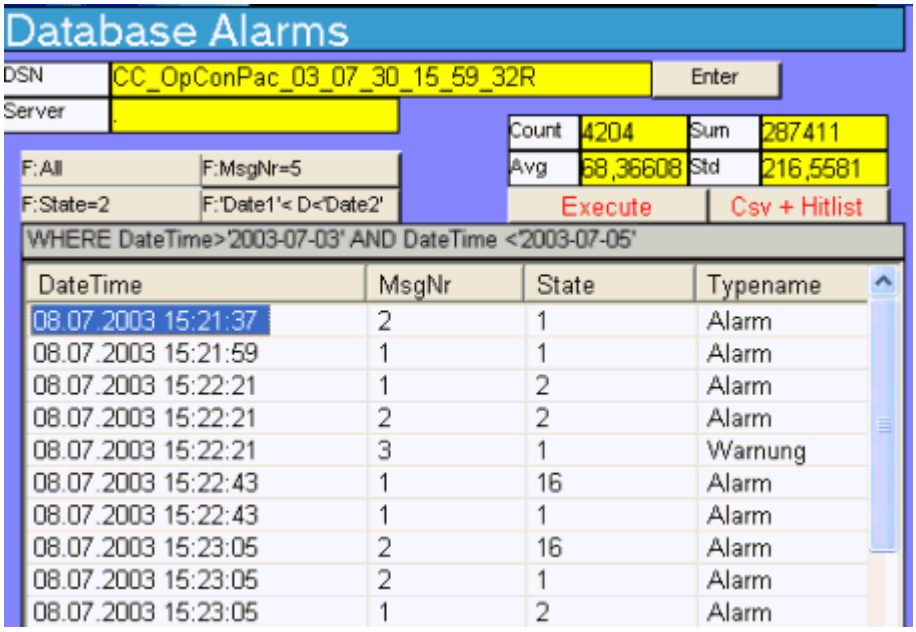

5.4.4 Darstellung der Meldearchive

Einleitung

Das Abfrageergebnis wird als Recordset zurückgegeben. In diesem Kapitel wird der Aufbau des Recordset für Meldearchive beschrieben.

Informationen zum Status von Meldungen finden Sie im WinCC Information System unter "Arbeiten mit WinCC > ANSI-C zum Erstellen von Funktionen und Aktionen > ANSI-C Funktionsbeschreibungen > Anhang > Strukturdefinitionen > Strukturdefinition MSG_RTDATA_STRUCT".

Recordset-Aufbau

Position	Feldname	Typ	Kommentar
1	MsgNr	Integer 4 Byte	Meldenummer
2	State	Small Integer 2 Byte	Status der Meldung
3	DateTime	DateTime 8 Byte	Zeitstempel der Meldung (Datum/Uhrzeit ohne Millisekunden)
4	Ms	Small Integer 2 Byte	Zeitstempel der Meldung (Millisekunden)
5	Instance	VarChar(255)	Instanzname der Meldung
6	Flags1	Integer 4 Byte	(nur interne Verwendung)
7	PValueUsed	Integer 4 Byte	Verwendete Prozesswerte

Position	Feldname	Typ	Kommentar
8 bis 17	PValue1 bis PVa-lue10	Real 8 Byte	numerischer Prozesswert 1 bis 10
18 bis 27	PText1 bis PText10	VarChar(255)	Prozesswerttext 1 bis 10
28	Computername	VarChar(255)	Rechnername
29	Application	VarChar(255)	Applikationsname
30	Comment	VarChar(255)	Kommentar
31	Username	VarChar(255)	Benutzername
32	Counter	Integer 4 Byte	Fortlaufender Meldezähler
33	TimeDiff	Integer 4 Byte	Zeitdifferenz zum Zustand "Gekommen"
34	Classname	VarChar(255)	Name der Meldeklasse
35	Typename	VarChar(255)	Name der Meldeart
36	Class	Small Integer 2 Byte	ID der Meldeklasse
37	Type	Small Integer 2 Byte	ID der Meldeart
38 bis 47	Text1 bis Text10	VarChar(255)	Meldetext 1 bis 10
48	AG_NR	Small Integer 2 Byte	Nummer des AG
49	CPU_NR	Small Integer 2 Byte	Nummer der CPU
50	CrComeFore	Integer 4 Byte	Vordergrundfarbe für Status "gekommen"
51	CrComeBack	Integer 4 Byte	Hintergrundfarbe für Status "gekommen"
52	CrGoFore	Integer 4 Byte	Vordergrundfarbe für Status "gegangen"
53	CrGoBack	Integer 4 Byte	Hintergrundfarbe für Status "gegangen"
54	CrAckFore	Integer 4 Byte	Vordergrundfarbe für Status "quittiert"
55	CrAckBack	Integer 4 Byte	Hintergrundfarbe für Status "quittiert"
56	LocaleID	Integer 4 Byte	Ort des Alarms
57	Priority	Integer 4 Byte	Priorität
58	AP_type	Integer 4 Byte	Loop in Alarm
59	AP_name	VarChar(255)	Loop in Alarm Funktionsname
60	AP_PAR	VarChar(255)	Loop in Alarm Bild
61	InfoText	VarChar(255)	Infotext
62	TxtCame	VarChar(255)	Text gekommen
63	TxtWent	VarChar(255)	Text gegangen
64	TxtCameNWent	VarChar(255)	Text gekommen und gegangen
65	TxtAck	VarChar(255)	Text quittiert
66	AlarmTag	Integer 4 Byte	Meldevariable
67	AckType	Small Integer 2 Byte	Quittiertyp
68	Params	Integer 4 Byte	Parameter
69	Servername	VarChar(255)	Servername

5.4.5 Abfrage für Anwenderarchive

Einleitung

Mit der folgenden Abfrage können Sie über MS OLE DB-Provider auf WinCC-Anwenderarchive zugreifen. Der Zugriff kann lesend und schreibend erfolgen, um die gespeicherten Daten auszuwerten oder auch zu ändern und zu speichern.

Über Filterkriterien können die Daten selektiert werden. Die Abfrage wird mit dem Command-Objekt an die Datenbank übergeben.

Hinweis

Beachten Sie beim Zugriff über den MS OLE DB-Provider auf WinCC-Anwenderarchive:

- Stellen Sie sicher, dass kein schreibender Zugriff gleichzeitig über MS OLE DB-Provider und über WinCC möglich ist. Dadurch vermeiden Sie Inkonsistenzen in den Archiven.
- Änderungen über MS OLE DB-Provider werden in WinCC Runtime erst nach einer Neuanwahl des User Archive Table Controls durch einen Bildwechsel angezeigt. Die aktuellen Daten der Anwenderarchive werden dabei neu eingelesen.
- Über MS OLE DB-Provider geänderte Anwenderarchive werden in einem redundanten System nicht abgeglichen.
- Beachten Sie, dass WinCC-Updates zu Änderungen im Datenbank-Schema führen können. Schema-Änderungen sind auch bei der Installation von Hotfixes und Service Packs möglich. In diesem Fall müssen Sie den lesenden und schreibenden Zugriff entsprechend anpassen.

Syntax

Lesen von Werten

```
SELECT * FROM UA#<ArchiveName>[WHERE <Condition>...., optional]
```

Schreiben von Werten

```
UPDATE UA#<ArchiveName> SET UA#<ArchiveName>.<Column_n> = <Value>
[WHERE <Condition>...., optional]
```

Einfügen eines Datensatzes

```
INSERT INTO UA#<ArchiveName> (ID,<Column_1>,<Column_2>,<Column_n>)
VALUES (<ID_Value>, Value_1,Value_2,Value_n)
```

Löschen eines Datensatzes

```
DELETE FROM UA#<ArchiveName> WHERE ID = <ID_Number>
```

Parameter

Parameter	Beschreibung
ArchiveName	Name des Anwenderarchivs.
Condition	Filterkriterium z.B.: LastAccess>'2004-06-01' AND LastAccess<'2004-07-01' DateTime>'2004-06-01 17:30:00' ID = 5 ID > 3

Beispiel 1:
Liest alle Daten im Anwenderarchiv "Test".

```
SELECT * FROM UA#Test
```

Beispiel 2:
Liest alle Daten im Anwenderarchiv "Test", die zwischen 01.06.2004 und 01.07.2004 geändert wurden.

```
SELECT * FROM UA#Test WHERE LastAccess>'2004-06-01' AND
LastAccess<'2004-07-01'
```

Beispiel 3:
Schreibt den Wert 'New_String' in das Feld F_STRING der ID 3.

```
UPDATE UA#TEST SET F_STRING = 'New_String' WHERE ID = 3
```

Beispiel 4:
Fügt einen Datensatz mit der ID 100 ein.

```
INSERT INTO UA#Test (ID,F_Integer,F_Float,F_Double,F_String) VALUES
(100,10,'10.0','AAAA')
```

Beispiel 5:
Löscht den Datensatz mit der ID 100.

```
DELETE FROM UA#Test WHERE ID = 100
```

5.4.6 Zugriff über den Wizard "SQL Server Import/Export" projektieren

Einleitung

Mit dem WinCC OLE DB-Provider ist der Zugriff auf WinCC-Datenbanken unter Einsatz des Wizards "SQL Server Import/Export" möglich. Die entpackten Daten können Sie mit Hilfe des Wizards in einer Zwischendatenbank ablegen, auf die Sie mit Standard-SQL-Abfragen zugreifen können.

- Auf WinCC-Rechnern ist der Zugriff lokal oder remote auf die Runtime- und die Archiv-Datenbanken möglich.
- Bei Langzeit-Archivservern kann nur auf die Archiv-Datenbanken lokal oder remote zugegriffen werden, da diese keine Runtime-Datenbank besitzen.

Vorgehensweise

1. Starten Sie "SQL Server Management Studio" und wählen Sie die gewünschte Datenbank.
2. Wählen Sie im Kontextmenü der Datenbank "Tasks > Export Date...".
   Der SQL Server Import/Export-Wizard öffnet sich.
3. Projektieren Sie die Datenquelle.
   Klicken Sie auf "Next". Wählen Sie im Feld "Data source" den Eintrag "WinCC OLE DB-Provider for Archives". Klicken Sie auf die Schaltfläche "Properties...". Der Dialog "Data Link Properties" öffnet sich.
4. Projektieren Sie die korrekten Einstellungen des Providers.
   Geben Sie im Feld "Data Source" folgenden Text als Datenquelle ein: ".\WinCC". Der Eintrag für "Location" bleibt frei.
   Tragen Sie unter "Enter the initial catalog to use" entweder die gewünschte Runtime-Datenbank oder den symbolischen Namen ein, der im "Archive Connector Tool" konfiguriert wurde. Die korrekte Schreibweise des Namens finden Sie im "SQL Server Management Studio" im Verzeichnis "Databases". Alternativ können Sie auch die Datenbank "CC_ExternalBrowsing" für Runtime-Daten und Archiv-Daten eintragen.
   Klicken Sie auf die Registerkarte "Advanced". Wählen Sie für die Eigenschaft "Connect timeout" die gewünschte Zeit in Sekunden. Wählen sie in der Eigenschaft "Access permissions" nur das Optionskästchen "ReadWrite".
   Schließen Sie den Dialog über die Schaltfläche "OK".

   Hinweis

   Um die Performance bei lokalem Zugriff zu verbessern, tragen Sie im Feld "Data Source" "<Rechnername>\WinCC" ein statt ".\WinCC".

5. Projektieren Sie das Ziel der Daten.
   Klicken Sie auf "Next". Wählen Sie im Feld "Destination" z. B. den Eintrag "SQL Native Client" aus. Als Servernamen können Sie jede SQL Server-Instanz verwenden. Im Feld "Database" können Sie jede selbst erstellte Ziel-Datenbank als Datenbank eintragen.

   Hinweis

   Sie können das Feld "Database" auch leer lassen. Dann werden keine Ziel-Tabellen erstellt.

6. Projektieren Sie die Abfragebedingungen.
   Klicken Sie auf "Next". Wählen Sie die Option "Write a query to specify the data to transfer". Klicken Sie auf "Next". Geben Sie die gewünschte Abfragebedingung ein. Zum Beispiel werden mit der Abfrage "Tag:R,1,'0000-00-00 00:10:00.000','0000-00-00 00:00:00.000'" von der ValueID "1" die Werte der letzten 10 Minuten der Archivierung gelesen. Weitere Informationen zur Syntax finden Sie im Kapitel "Abfrage für Prozesswertarchive".
7. Beenden Sie den Wizard und exportieren Sie die Daten.
   Klicken Sie auf "Next" und auf der letzten Seite des Wizards auf "Finish". Der Wizard führt den Datenexport zur Ziel-Datenbank durch. Wenn der Datenexport erfolgreich war, werden die unkomprimierten Daten in der Ziel-Datenbank in der neu erstellten Tabelle "dbo.Query" abgelegt. Sie können den Tabellennamen ändern. Auch wenn Sie den Tabellennamen nicht ändern, werden die Daten bei einem neuen Export nicht überschrieben. Der Wizard legt neue Tabellen mit den Namen "Query1", "Query2" etc. an.

5.4.7 Voraussetzungen für die Nutzung der Reporting Services schaffen

Einleitung

Mit WinCC können Sie die Reporting Services des SQL-Server 2014 nutzen. Sie können somit mit dem Microsoft Visual Studio erstellte Reports mit Archivdaten im Netz zur Verfügung stellen.

Voraussetzungen

Die Nutzung der Reporting Services mit WinCC erfordert folgende weitere Softwarevoraussetzungen:

- Internet Information Services
- Reporting Services des MS SQL Server 2014

Hinweis

Nehmen Sie die Installationsschritte in genau der angegebenen Reihenfolge vor.

Internet Information Services installieren

1. Wählen Sie "Einstellungen > Systemsteuerung > Software".
2. Klicken Sie links auf die Schaltfläche "Windows-Komponenten hinzufügen/entfernen".
3. Aktivieren Sie im Dialog "Assistent für Windows-Komponenten" das Kontrollkästchen "Internet-Informationsdienste (IIS)".

Reporting Services des MS SQL Server 2014 installieren

Sie installieren die Reporting Services von der WinCC-DVD oder von deren Platz im Dateisystem.

1. Wählen Sie "Einstellungen > Systemsteuerung > Programme und Funktionen".
2. Selektieren Sie "Microsoft SQL Server 2014" und klicken Sie auf "Deinstallieren/ändern".
3. Wählen Sie "Hinzufügen" und im WinCC Setup den Pfad "InstData > SQL > SQL2014STD > setup".
   Der Microsoft SQL Server Installation Wizard wird geöffnet. Folgen Sie den Anweisungen.
4. Wählen Sie unter "Installationstyp" die Option "Features zu einer vorhandenen SQL Server 2014-Instanz hinzufügen" und die Instanz "WINCC".
   Die Management Tools und damit die SQL Server Data Tools sind bereits installiert.
5. Aktivieren Sie "Reporting Services - Native" und folgen Sie den Anweisungen.
   Die Reporting Services werden installiert.
6. Starten Sie den Konfigurations-Manager für Reporting Services und konfigurieren Sie die Reporting Services.

Internet Information Services konfigurieren

1. Öffnen Sie die Computerverwaltung.
2. Wählen Sie unter "Dienste und Anwendungen" den "Internetinformationsdienste (IIS)-Manager".
3. Öffnen Sie das Feature "Authentifizierung" und wählen Sie im Kontextmenü von "Anonyme Athentifizierung" den Eintrag "Bearbeiten".
4. Wählen Sie die Option "Bestimmter Benutzer" und geben Sie den Benutzernamen und Passwort ein.
   Der Benutzername hat folgendes Format: <Domäne oder Rechnername>\<Benutzer>

Hinweis

Wir empfehlen, nach Abschluss der Installationen den Rechner neu zu starten.

Ergebnis

Damit sind die Voraussetzungen für die Nutzung der Reporting Services hergestellt. Sie können jetzt Reports erstellen und im Internet zur Verfügung stellen.

5.4.8 Abfragen der Archivdaten

Einleitung

Die Abfragen werden mit dem Command-Objekt an die Datenbank übergeben. Ein wichtiger Parameter neben dem ConnectionString ist dabei der CommandText. Der CommandText übergibt die Abfrage. Das Ergebnis wird als Recordset zurückgegeben.

Hinweis

Zeitbereich beim Abfragen der Archive für Meldungen und Prozesswerte

Wird beim Abfragen der Archive für Meldungen oder Prozesswerte ein Zeitbereich gewählt, in dem keine Meldungen oder Werte in den Archiven vorhanden sind, so erfolgt keine Hinweis-Meldung oder sonstige Anzeige dieses Zustands. Soll dieser Zustand angezeigt werden, so muss durch den Anwender eine Fehlerbehandlung erfolgen.

Eine einfache Ausführung dieser Fehlerbehandlung ist im Beispielskript unter dem Thema "Beispiel: Meldearchivdaten mit dem WinCC OLE DB-Provider lesen" gezeigt.

In den folgenden Beispielen wird jeweils ein Command-Objekt erzeugt und die Abfrage als CommandText übergeben.

Zu den folgenden Aufbaubeispielen für den CommandText gehört auch der ConnectionString, dessen Aufbau unter "Verbindung zur Archivdatenbank herstellen" beschrieben ist.

Aufbau des CommandText

Prozesswertarchive:

```
Set oRs = CreateObject("ADODB.Recordset")
Set oCom = CreateObject("ADODB.Command")
oCom.CommandType = 1
Set oCom.ActiveConnection = conn
oCom.CommandText = "TAG:R,'PVArchive\Tag1','0000-00-00
00:10:00.000','0000-00-00 00:00:00.000'"
```

Meldearchive

```
Set oRs = CreateObject("ADODB.Recordset")
Set oCom = CreateObject("ADODB.Command")
oCom.CommandType = 1
Set oCom.ActiveConnection = conn
oCom.CommandText = "ALARMVIEW:Select * FROM AlgViewEnu"
```

Anwenderarchive

```
Set oRs = CreateObject("ADODB.Recordset")
Set oCom = CreateObject("ADODB.Command")
```

```
oCom.CommandType = 1
Set oCom.ActiveConnection = conn
oCom.CommandText = "SELECT * FROM UA#Test"
```

Ort des Recordset festlegen

Um für die Abfrage von Archivdaten den Ort des Recordset festzulegen, müssen Sie für die Eigenschaft "CursorLocation" den Wert "3" verwenden, z.B. "conn.CursorLocation = 3". Das Recordset wird auf dem Client erstellt.

5.4.9 Darstellung der Anwenderarchive

Einleitung

Jedes Anwenderarchiv besteht aus Datenfeldern mit editierbaren Eigenschaften. Jedes Datenfeld besitzt Eigenschaften wie Name, Aliasname, Typ, Länge, Wert usw. Die Darstellung der Datenfelder und Eigenschaften im Editor User Archives erfolgt in Zeilen und Spalten. Deshalb wird anstelle von Datenfeldern auch von Zeilen und anstelle von Eigenschaften auch von Spalten gesprochen.

Nachfolgend wird als Beispiel der Aufbau des Anwenderarchivs "Test" beschrieben. Dieses Anwenderarchiv ist im Connectivity Pack Demo-Projekt "OPConPack" im Verzeichnis "\Samples\Connectivity Pack\DemoProject" enthalten.

Aufbau des Anwenderarchives "Test"

Feldname	Typ	Kommentar
ID	Integer	Eindeutige Identifikation des Wertes
F_Integer	Integer	Beispiel für Wert
F_Float	Float	Beispiel für Wert
F_Double	Double	Beispiel für Wert
F_String	String	Beispiel-Zeichenkette

5.5 Analysefunktionen für Meldungen und Prozesswerte

5.5.1 Analysefunktionen für Meldungen und Prozesswerte

Einleitung

Mit dem WinCC stehen verschiedene Analysefunktionen bei der Abfrage von archivierten Meldungen und Prozesswerten zur Verfügung.

Zur Analyse wird eine Abfrage gestartet mit Parametern für verschiedene Aggregatfunktionen. Die Berechnung der Aggregatfunktionen erfolgt auf dem Connectivity Pack-Server und nur das Ergebnis wird zum Client übertragen.

Weitere Informationen zu CommandText und ConnectionString, die in den folgenden Beispielen verwendet sind, finden Sie unter "Verbindung zur Archivdatenbank herstellen" und "Abfragen der Archivdaten".

Analysefunktionen für Meldungen

Die Analyseabfrage für Meldearchive liefert einen spezifischen Recordset zurück, der zu jeder Meldung Konfigurations- und Runtime-Daten und die Ergebnisse der Aggregatfunktionen enthält.

Der zurückgelieferte Recordset von Analyseabfragen von Meldearchiven ist nicht identisch mit dem Recordset normaler Abfragen von Meldearchiven. Weitere Informationen finden Sie im Kapitel "Darstellung von Meldearchiven bei Analyseabfragen".

Für jede Meldung werden folgende Aggregatfunktionen berechnet. Die Spaltenbezeichnungen in der Ergebnisliste sind in Klammern gesetzt.

- Summe Meldungshäufigkeit ("FreqOfAlarm")
- Kumulierte Dauer von "Meldung gekommen" bis "Meldung gegangen" ("CumDurationComeGo")
- Durchschnittliche Dauer von "Meldung gekommen" bis "Meldung gegangen" ("AvDurationComeGo")
- Kumulierte Dauer von "Meldung gekommen" bis Erstquittierung ("CumDurationComeAckn1")
- Durchschnittliche Dauer von "Meldung gekommen" bis Erstquittierung ("AvDurationComeAckn1")
- Kumulierte Dauer von "Meldung gekommen" bis Zweitquittierung ("CumDurationComeAckn2")
- Durchschnittliche Dauer von "Meldung gekommen" bis Zweitquittierung ("AvDurationComeAckn2")
- Kumulierte Dauer von "Meldung gekommen" bis "Meldung gekommen" ("CumDurationComeCome")
- Durchschnittliche Dauer von "Meldung gekommen" bis "Meldung gekommen" ("AvDurationComeCome")

Syntax

Zur Berechnung der Aggregatfunktionen für Meldungen wird der folgende Befehl an den WinCC OLE DB-Provider übergeben.

```
"AlarmHitView: SELECT * FROM <ViewName>[WHERE <Condition>]"
```

Dabei sind :
<ViewName> = Name der Datenbanktabelle in der gewünschten Sprache z.B. ALGVIEWENU für Englisch.
[WHERE <Condition>] = optionales Filterkriterium als WHERE-Bedingung in SQL-Syntax.

Weitere Informationen zur Syntax der Parameter finden Sie im Kapitel "Abfrage für Meldearchive".

Beispiel

Das Beispiel liefert aus der Datenbank "ALGVIEWENU" die Ergebnisse der Aggregatfunktionen für alle Meldungen im Zeitbereich zwischen 15.07.2004 12:00 Uhr und 12:15 Uhr.

ConnectionString:

```
"Provider=WinCCOLEDBProvider.
1;Catalog=CC_OpenArch_03_05_27_14_11_46R;Data Source=.\WinCC"
```

CommandText:

```
"AlarmHitView: SELECT * FROM ALGVIEWENU WHERE DateTime>'2004-07-15
12:00:00' AND DateTime<'2004-07-15 12:15:00'"
```

Analysefunktionen für Prozesswerte

Die Analyse für Prozesswerte liefert das Ergebnis einer Aggregatfunktion zurück. In einer Abfrage kann nur eine Aggregatfunktion berechnet werden.

Für Prozesswerte stehen folgende Aggregatfunktionen zur Verfügung.

- MIN (Minimum)
- MAX (Maximum)
- AVG (Durchschnitt)
- SUM (Summe aller Werte)
- COUNT (Anzahl der Prozesswerte)
- COUNTER (Anzahl der Einträge mit dem Wert "1", z.B. Abfrage binärer Variablen)
- STDEV (Statistische Standard-Abweichung)
- VAR (Statistische Varianz)

Syntax

Zur Berechnung einer Aggregatfunktion für Prozesswerte wird eine Abfrage an den MS SQL OLE DB-Provider übergeben und die Prozedur "cp_TagStatistic" aus der Datenbank "SQL Server Master" ausgeführt.

Hinweis

Die Analysefunktionen für Prozesswerte beim transparenten Zugriff funktioniert nur mit der Connectivity Station auf einem Client mit eigenem Projekt.

Der Prozedur "cp_TagStatistic" werden folgende Parameter übergeben.

```
cp_TagStatistic @P1,@P2,@P3[,@P4]
```

Dabei sind :
"@P1" = Datenbankname (z.B. WinCC Runtime-Datenbank oder der symbolische Name des Verzeichnisses mit den ausgelagerten Archiven). Verwenden Sie beim transparenten Zugriff statt des Datenbanknamens den WinCC-Projektnamen.
"@P2" = WinCC OLE DB-Provider String für Prozesswerte.
"@P3" = gewünschte Aggregatfunktion.
"@P4" = <Symbolischer Rechnername>::\WinCC (nur bei transparentem Zugriff erforderlich).

Weitere Informationen zur Syntax der Parameter "@P1" und "@P2" finden Sie im Kapitel "Abfrage für Prozesswertarchive".

Hinweis

Analysefunktionen für Prozesswerte mit asiatischen Archivvariablennamen

Wenn Sie Archivvariablennamen mit asiatischen Zeichensätzen für die Berechnung einer Analysefunktion verwenden, dann muss der Aufruf für Unicode-Zeichensatz angepasst werden.

Ergänzen Sie vor den ersten beiden Parametern das Prefix "N'".

Beispiel: cp_TagStatistic N'TestDB',N'TAG:R,17,"2004-05-17 12:00:00","2004-05-17 13:00:00'","AVG'

Beispiel

Diese Abfrage liefert von der ValueID "17" aus der Datenbank "TestDB" den Durchschnittswert der Prozesswerte im Zeitbereich zwischen 17.05.2004 12:00 Uhr und 13:00 Uhr.

ConnectionString:

```
"Provider=SQLNCLI11;Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=master ;Data Source=.\WinCC"
```

CommandText:

```
"cp_TagStatistic 'TestDB','TAG:R,17,''2004-05-17
12:00:00'',''2004-05-17 13:00:00''','AVG'"
```

Beispiel für transparenten Zugriff

Diese Abfrage liefert von der ValueID "7" aus dem Projekt "WinCCProj" den Durchschnittswert der Prozesswerte im Zeitbereich zwischen 14.09.2006 10:00 Uhr und 11:00 Uhr.

ConnectionString:

```
"Provider=SQLNCLI11;Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=master ;Data Source=.\WinCC"
```

CommandText:

```
"cp_TagStatistic 'WinCCProj','TAG:R,7,''2006-09-14
10:00:00'',''2006-09-14 11:00:00''','AVG','Symb_WinCCProj::\WinCC'"
```

5.5.2 Darstellung von Meldearchiven bei Analyseabfragen

Einleitung

Die Analyseabfrage für Meldearchive liefert einen spezifischen Recordset zurück, der zu jeder Meldung Konfigurations- und Runtime-Daten und die Ergebnisse der Aggregatfunktionen enthält.
Dieser Recordset ist nicht identisch mit dem Recordset normaler Abfragen von Meldearchiven.

Recordset-Aufbau bei der Analyse von Meldearchiven

Bei der Abfrage von Meldearchiven mit der Analysefunktion "AlarmHitView" wird das Ergebnis als Recordset mit dem folgenden Aufbau zurückgegeben.

Position	Feldname	Typ	Kommentar
1	MsgNr	Integer 4 Byte	Meldenummer
2	State	Small Integer 2 Byte	Status der Meldung
3	DateTime	DateTime 8 Byte	Zeitstempel der Meldung (Datum/Uhrzeit ohne Millisekunden)
4	Ms	Small Integer 2 Byte	Zeitstempel der Meldung (Millisekunden)
5	Instance	VarChar(255)	Instanzname der Meldung
6	Flags1	Integer 4 Byte	(nur interne Verwendung)
7	Counter	Integer 4 Byte	Fortlaufender Meldezähler
8	TimeDiff	Integer 4 Byte	Zeitdifferenz zum Zustand "Gekommen"
9	Classname	VarChar(255)	Name der Meldeklasse
10	Typename	VarChar(255)	Name der Meldeart
11	Class	Small Integer 2 Byte	ID der Meldeklasse
12	Type	Small Integer 2 Byte	ID der Meldeart
13 bis 22	Text1 bis Text10	VarChar(255)	Meldetext 1 bis 10
23	AG_NR	Small Integer 2 Byte	Nummer des AG

Position	Feldname	Typ	Kommentar
24	CPU_NR	Small Integer 2 Byte	Nummer der CPU
25	CrComeFore	Integer 4 Byte	Vordergrundfarbe für Status "gekommen"
26	CrComeBack	Integer 4 Byte	Hintergrundfarbe für Status "gekommen"
27	CrGoFore	Integer 4 Byte	Vordergrundfarbe für Status "gegangen"
28	CrGoBack	Integer 4 Byte	Hintergrundfarbe für Status "gegangen"
29	CrAckFore	Integer 4 Byte	Vordergrundfarbe für Status "quittiert"
30	CrAckBack	Integer 4 Byte	Hintergrundfarbe für Status "quittiert"
31	Priority	Integer 4 Byte	Priorität
32	AP_type	Integer 4 Byte	Loop in Alarm
33	AP_name	VarChar(255)	Loop in Alarm Funktionsname
34	AP_PAR	VarChar(255)	Loop in Alarm Bild
35	InfoText	VarChar(255)	Infotext
36	TxtCame	VarChar(255)	Text gekommen
37	TxtWent	VarChar(255)	Text gegangen
38	TxtCameNWent	VarChar(255)	Text gekommen und gegangen
39	TxtAck	VarChar(255)	Text quittiert
40	AckType	Small Integer 2 Byte	Quittiertyp
41	FreqOfAlarm	Integer 4 Byte	Summe Meldungshäufigkeit
42	CumDurationComeGo	Integer 4 Byte	Kumulierte Dauer von "Meldung gekommen" bis "Meldung gegangen"
43	AvDurationComeGo	Real 8 Byte	Durchschnittliche Dauer von "Meldung gekommen" bis "Meldung gegangen"
44	CumDurationComeAckn1	Integer 4 Byte	Kumulierte Dauer von "Meldung gekommen" bis Erstquittierung
45	AvDurationComeAckn1	Real 8 Byte	Durchschnittliche Dauer von "Meldung gekommen" bis Erstquittierung
46	CumDurationComeAckn2	Integer 4 Byte	Kumulierte Dauer von "Meldung gekommen" bis Zweitquittierung
47	AvDurationComeAckn2	Real 8 Byte	Durchschnittliche Dauer von "Meldung gekommen" bis Zweitquittierung
48	CumDuration ComeCome	Integer 4 Byte	Kumulierte Dauer von "Meldung gekommen" bis "Meldung gekommen"
49	AvDurationComeCome	Real 8 Byte	Durchschnittliche Dauer von "Meldung gekommen" bis "Meldung gekommen"

5.6 Grundlagen zu OLE DB

Einleitung

Über die von WinCC zur Verfügung gestellte OLE DB-Schnittstelle und den zugehörigen Datenbank-Provider haben Sie die Möglichkeit, auf Prozesswert- und Meldearchive zuzugreifen.

OLE DB

OLE DB ist ein offener Standard für einen schnellen Zugriff auf unterschiedliche Datenbanken. Dabei spielt es keine Rolle, ob die Datenbank relational ist oder nicht.

Die Verbindung zwischen der OLE DB-Ebene und der Datenbank wird durch einen Datenbank-Provider hergestellt.

OLE DB-Schnittstellen und -Provider werden von unterschiedlichen Herstellern angeboten.

WinCC OLE DB-Provider

Über den WinCC OLE DB-Provider können Sie direkt auf die, in der MS SQL Server-Datenbank abgelegten, WinCC-Archivdaten zugreifen. In Abhängigkeit von der Projektierung werden Prozessdaten von WinCC komprimiert abgelegt. Der WinCC OLE DB-Provider erlaubt einen transparenten Zugriff auch auf diese Daten.

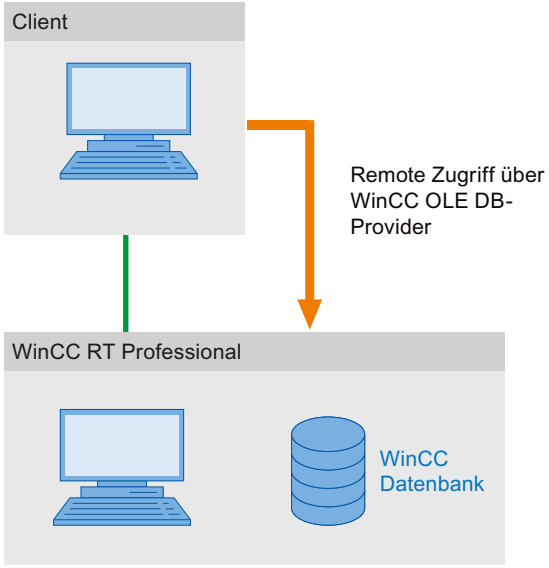

Microsoft OLE DB

Mit Microsoft OLE DB können Sie nur auf WinCC User Archives zugreifen.

Für den Zugriff auf WinCC User Archives über Microsoft OLE DB ist eine Connectivity Pack Lizenz auf dem WinCC-Rechner erforderlich, auf dessen User Archives zugegriffen werden soll. Für den Remote-Zugriff auf die MS SQL-Server-Datenbank wird die WinCC Client Access Licence benötigt.

Zum Schutz vor unberechtigten Zugriffen über MS OLE DB können vom Administrator der Datenbanken geeignete Maßnahmen getroffen werden. Weitere Informationen finden Sie im Kapitel "Sicherheitseinstellungen beim Zugriff auf SQL-Datenbanken über MS OLE DB".

Hinweis

Mit Microsoft OLE DB ist nur der Zugriff auf WinCC User Archives getestet und freigegeben, nicht jedoch auf Melde- und Prozesswertarchive.
Verwenden Sie für den Zugriff auf Melde- und Prozesswertarchive den WinCC OLE DB-Provider.

5.7 WinCC OLE DB-Provider

Einleitung

Über den WinCC OLE DB-Provider wird der Zugriff auf die Prozesswert- und Meldearchive möglich.

Weiterhin bietet der WinCC OLE DB-Provider Analysefunktionen wie beispielsweise Minimum, Maximum von Archivvariablen.

Zugriff auf Archivdaten

Der Zugriff auf Archivdaten kann unter verschiedenen Konstellationen erfolgen:

Gleichzeitiger Zugriff auf Archiv- und Runtime-Datenbanken

Die Funktion "ArchiveMonitor" fügt die Daten aus Runtime- und Archiv-Datenbanken des aktivierten WinCC-Projekts zusammen in einer SQL-Datenbank.

Die SQL-Datenbank heißt "CC_ExternalBrowsing" und erstellt darin die Views "AMT" und "ARCHIVE".

Die Views stellen alle erforderlichen Informationen für den WinCC OLE DB-Provider zusammen

Da diese Views alle erforderlichen Informationen für den WinCC OLE DB-Provider zur Verfügung stellen, kann die Datenbank "CC_ExternalBrowsing" als Datenquelle für den Provider benutzt werden.

Beim Deaktivieren von WinCC Runtime werden zusätzlich erstellte Views in "CC_ExternalBrowsing" wieder gelöscht.

Da auf einem Archivserver keine Runtime-Datenbanken vorhanden sind, ist der Zugriff auf die Archiv-Datenbanken über die Datenbank "CC_ExternalBrowsing" nicht möglich.

WinCC OLE DB-Provider als "Linked Server"

Hinweis

Bei der Projektierung von WinCC OLE DB-Provider als "Linked Server" muss im Dialog "Linked Server Properties" im Feld "Provider Name" der Eintrag "WinCC OLE DB-Provider for Archives" ausgewählt werden.

Im Dialog "Provider Options" muss das Kontrollkästchen "AllowInProcess" aktiviert werden. Weitere Informationen finden Sie hier: "SQL Server Books Online" unter "Configuring OLE DB Providers for Distributed Queries".

Der WinCC OLE DB-Provider kann im SQL-Server in die Liste der "Linked Servers" aufgenommen werden. Unabhängig davon, ob die Datenbank lokal oder remote abgelegt ist.

Beispiel:

WINCC OLE DB-Provider als Linked Server mit dem Server-Namen "WinCC":

```
select min(realvalue) from openquery(WINCC,'Tag:R,1,''0000-00-00
00:01:00.000'',''0000-00-00 00:00:00.000''')
select * from openquery(WINCC,'Tag:R,1,''0000-00-00
00:01:00.000'',''0000-00-00 00:00:00.000''')
```

5.8 Installation

Installation von WinCC OLE DB-Provider

Der WinCC OLE DB-Provider wird automatisch bei der Installation von WinCC mitinstalliert.

Nach der Installation von WinCC steht der WinCC OLE DB-Provider folgenden Systemen zur Verfügung:

- Engineering Station mit WinCC
- Bediengerät mit WinCC Runtime Professional

Mit Rezepturen arbeiten 6

6.1 Mit Rezepturen arbeiten

6.1.1 Grundlagen

6.1.1.1 Definition und Einsatzgebiete

Einleitung

In Rezepturen werden zusammengehörige Daten, z. B. Maschinenparametrierungen oder Produktionsdaten, zusammengefasst.

Beispiele:

- Maschinenparametrierungen, die notwendig sind, um die Produktion auf eine andere Produktvariante umzustellen.
- Produktbestandteile, die in unterschiedlichen Zusammensetzungen unterschiedliche Endprodukte ergeben.

Eine Rezeptur zeichnet sich durch eine feste Datenstruktur aus. Bei der Projektierung wird die Struktur einer Rezeptur festgelegt. Eine Rezeptur enthält Rezepturdatensätze, die sich in den Werten, nicht aber in der Struktur unterscheiden.

Rezepturen werden auf dem Bediengerät bzw. auf einem externen Speichermedium abgelegt. Ein Rezepturdatensatz wird grundsätzlich komplett in einem Arbeitsschritt zwischen Bediengerät und Steuerung übertragen. Zusätzlich können Sie Produktionsdaten in Runtime über eine CSV-Datei importieren.

Hinweis

Einschränkungen beim Import/Export

Das Exportieren bzw. Importieren von Rezepturen ist für folgende Bediengeräte nicht möglich:

- Basic Panels

Komplette Rezepturdaten, jedoch nicht einzelne Rezepturdatensätze können mit ProSave in das CSV-Format exportiert und importiert und auf das Bediengerät übertragen werden. Dabei wird die Runtime angehalten.

Einsatz von Rezepturen

Für den Einsatz von Rezepturen sind z. B. folgende Fälle möglich:

- Manuelle Produktion
  Sie wählen die gewünschten Rezepturdaten und zeigen die Rezepturdaten am Bediengerät an. Sie ändern bei Bedarf die Rezepturdaten und speichern Sie auf dem Bediengerät ab. Sie übertragen die Rezepturdaten zur Steuerung.
- Automatische Produktion
  Das Steuerungsprogramm stößt die Übertragung der Rezepturdaten zwischen Steuerung und Bediengerät an. Sie können die Übertragung auch durch eine Bedienung am Bediengerät starten. Die Produktion wird damit automatisch umgestellt. Eine Anzeige oder Änderung der Daten ist nicht zwingend notwendig.
- Teach-In-Betrieb
  An der Anlage optimieren Sie Produktionsdaten von Hand, z. B. Achspositionen oder Füllmengen. Die ermittelten Werte werden zum Bediengerät übertragen und in einem Rezepturdatensatz gespeichert. Zu einem späteren Zeitpunkt übertragen Sie die gespeicherten Rezepturdaten wieder zur Steuerung.

Eingeben und Ändern der Rezepturdaten

Sie geben die Daten in den einzelnen Rezepturdatensätzen ein und passen sie bei Bedarf an. Folgende Möglichkeiten stehen Ihnen dazu zur Verfügung:

- Eingabe der Daten bei der Projektierung
  Wenn die Produktionsdaten bereits feststehen, geben Sie die Daten bereits während der Projektierung der Rezeptur im Editor "Rezepturen" ein.
- Eingabe der Daten in Runtime
  Wenn Sie Produktionsdaten häufig anpassen müssen, ändern Sie die Daten folgendermaßen direkt in Runtime:
  - Sie geben die Daten direkt am Bediengerät ein.
  - Sie stellen die Parameter direkt an der Maschine ein. Sie übertragen dann die Daten von der Steuerung zum Bediengerät und speichern Sie in der Rezeptur.

6.1.1.2 Beispiele zum Einsatz von Rezepturen

Rezepturen werden z. B. in der Fertigungsindustrie und im Maschinenbau eingesetzt. Die folgenden Beispiele zeigen typische Anwendungsfälle, die Sie mit der Rezepturfunktionalität von WinCC umsetzen können:

- Maschinenparametrierung
  Ein Einsatzgebiet von Rezepturen ist die Maschinenparametrierung in der Fertigungsindustrie: Eine Maschine schneidet Holzplatten auf eine bestimmte Größe zu und bohrt Löcher. Je nach Holzplattengröße werden z. B. die Führungsschienen und Bohrer in neue Positionen gefahren. Die benötigten Positionsdaten sind als Datensätze in einer Rezeptur gespeichert. Im Teach-In-Betrieb parametrieren Sie die Maschine neu, wenn z. B. eine neue Holzplattengröße verarbeitet werden soll. Sie übertragen die neuen Positionsdaten direkt aus der Steuerung zum Bediengerät und speichern sie als neuen Datensatz.
- Chargenorientierte Produktion
  Ein anderes Einsatzgebiet von Rezepturen ist die chargenorientierte Produktion in der Nahrungsmittelindustrie: Eine Abfüllstation einer Fruchtsaftanlage produziert Saft, Nektar und Getränk in verschiedenen Geschmacksrichtungen. Die verwendeten Zutaten sind immer dieselben und unterscheiden sich nur in den Mischungsverhältnissen. Jede Geschmacksrichtung entspricht einer Rezeptur. Jedes Mischungsverhältnis entspricht einem Datensatz. Alle notwendigen Daten eines Mischungsverhältnisses können Sie mit einem Knopfdruck an die Maschinensteuerung übertragen.

6.1.1.3 Aufbau von Rezepturen

Einleitung

Der prinzipielle Aufbau einer Rezeptur wird anhand der Abfüllstation einer Fruchtsaftanlage dargestellt.

In einem Bediengerät können mehrere unterschiedliche Rezepturen vorhanden sein. Sie können eine Rezeptur z. B. vergleichen mit einem Karteikasten, der mehrere Karteikarten enthält. In diesem Karteikasten sind mehrere Herstellungsvarianten einer Produktfamilie enthalten. Die Daten von jeweils einer Herstellungsvariante sind komplett auf einer Karteikarte enthalten.

Beispiel:

Bei einer Anlage zur Getränkeherstellung wird eine Rezeptur für unterschiedliche Getränkevarianten benötigt. Es gibt z. B. die Getränkevarianten Fruchtsaftgetränk, Saft, Nektar.

Rezeptur

Die Rezeptur enthält alle Rezepturdatensätze, z. B. für die unterschiedlichen Getränkevarianten:

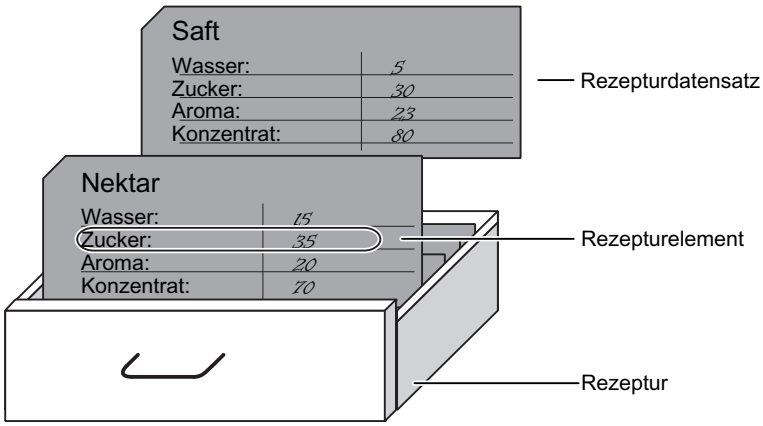

Rezepturdatensätze

Jede Karteikarte repräsentiert einen Rezepturdatensatz, der für die Herstellung einer Produktvariante notwendig ist.

Rezepturelemente

Jede Karteikarte einer Schublade hat den gleichen Aufbau. Alle Karteikarten enthalten Felder für die verschiedenen Zutaten. Jedes Feld entspricht einem Rezepturelement. Alle Datensätze einer Rezeptur enthalten damit die gleichen Elemente. Die Datensätze unterscheiden sich jedoch im Wert der einzelnen Elemente.

Beispiel:

Alle Getränke enthalten die folgenden Bestandteile:

- Wasser
- Konzentrat
- Zucker
- Aroma

Die Datensätze für Fruchtsaftgetränk, Saft oder Nektar unterscheiden sich jedoch z. B. in der Menge des Zuckers, der bei der Herstellung verwendet wird.

6.1.1.4 Anzeige der Rezepturen

Einleitung

Zur Anzeige von Rezepturen haben Sie folgende Möglichkeiten:

- Rezepturanzeige
- Rezepturbild

Eingaben in Rezepturanzeige und Rezepturbild

Im Rezepturbild oder in der Rezepturanzeige ändern Sie die Werte einer Rezeptur und beeinflussen so z. B. den Fertigungsprozess oder eine Maschine.

Rezepturanzeige und Rezepturbild können die gleiche Funktionalität für die Bedienung von Rezepturen haben. Sie unterscheiden sich in Folgendem:

- Darstellungsmöglichkeiten
- Bedienung
- Möglichkeiten der Datenübertragung zwischen Steuerung und Bediengerät

Rezepturanzeige

Die Rezepturanzeige ist für die Darstellung einfacher Rezepturen geeignet.

Die Rezepturanzeige ist ein fertiges Anzeige- und Bedienobjekt von WinCC zum Verwalten von Rezepturdatensätzen. Die Rezepturanzeige ist immer ein Bestandteil eines Bildes. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform. Die Darstellung und die Bedienungsmöglichkeiten passen Sie den jeweiligen Bedürfnissen an.

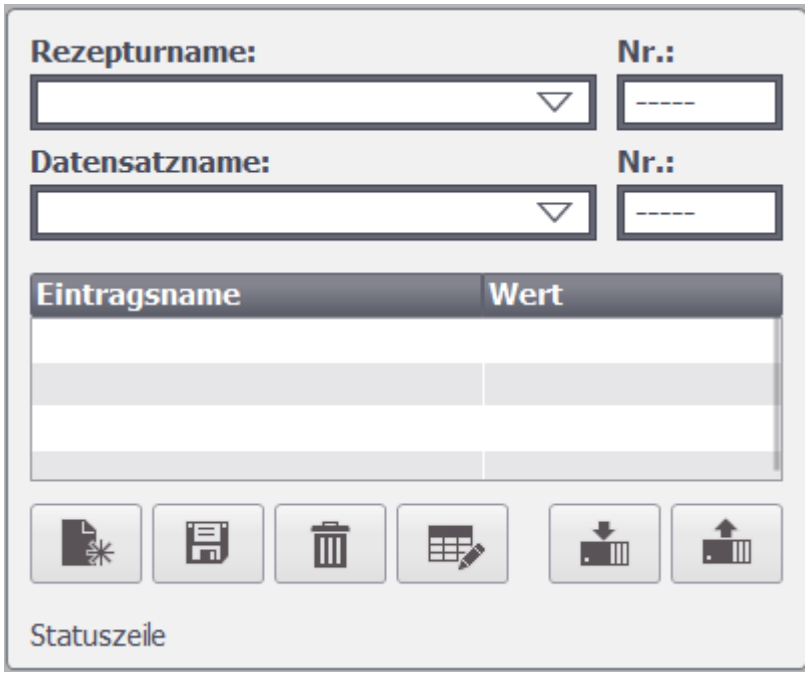

Wenn Sie in Ihrem Projekt Rezepturen mit einer Rezepturanzeige bearbeiten, werden die Werte in Rezepturdatensätzen gespeichert. Die Werte werden erst dann zwischen Bediengerät und Steuerung übertragen, wenn Sie das entsprechende Bedienobjekt bedienen.

Rezepturbild

Bei dem Rezepturbild handelt es sich um ein individuelles Bild, das Folgendes enthält:

- Eingabefelder für Rezepturvariablen
- Bedienobjekte zur Bedienung der Rezepturen, z. B. "SpeichereDatensatz"

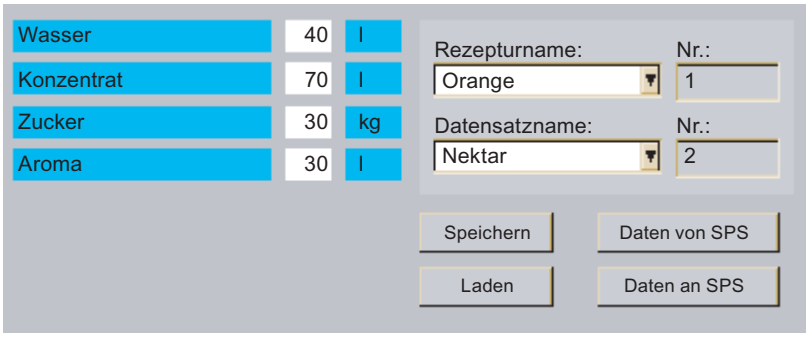

Das Rezepturbild ist in folgenden Fällen geeignet:

- Umfangreiche Rezepturen
- Zuordnung der Rezepturfelder zu der grafischen Darstellung des entsprechenden Anlagenbereichs
- Aufteilung der Rezepturdaten in mehrere Bilder

Hinweis

Die Werte von Rezepturvariablen werden abhängig von der Projektierung zu folgenden Zeitpunkten zwischen Steuerung und Rezepturbild übertragen:

- Sofort bei Änderung
- Bei Bedienung eines entsprechenden Bedienobjekts

Rezepturanzeige und Rezepturbild abgleichen

Wenn Sie Rezepturen in einer Rezepturanzeige und in einem Rezepturbild bearbeiten, können in Runtime Unterschiede zwischen den in der Rezepturanzeige angezeigten Werten und den in den zugehörigen Variablen gespeicherten Werten auftreten. Um dies zu verhindern, müssen Sie die Werte der Rezepturdatensätze mit den Werten der Rezepturvariablen abgleichen.

Es wird immer ein kompletter Rezepturdatensatz gespeichert bzw. abgegelichen.

Hinweis

Sie können die Rezepturvariablen nur mit der erweiterten Rezepturanzeige abgleichen. Ob der Abgleich durchgeführt wird, hängt davon ab, ob für die Rezeptur "Rezepturanzeige und Rezepturvariablen abgleichen " aktiviert ist.

6.1.1.5 Übertragen von Rezepturdatensätzen

Datenfluss bei Rezepturen

Das folgende Bild stellt den Datenfluss bei Rezepturen dar:

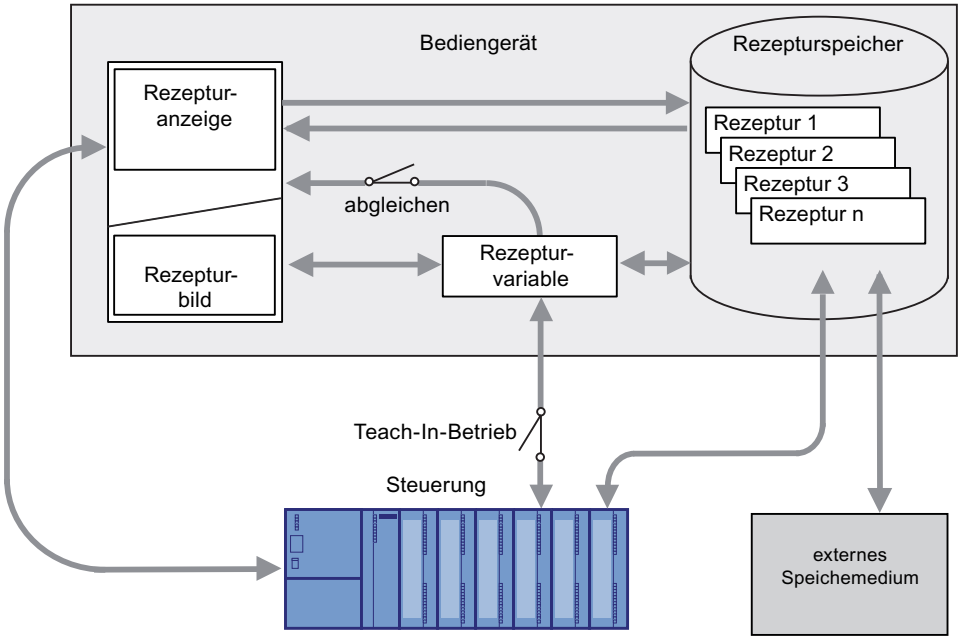

Zusammenwirken der Komponenten

In Runtime wirken folgende Komponenten zusammen:

- Rezepturanzeige / Rezepturbild
  Am Bediengerät werden Rezepturen in der Rezepturanzeige oder in einem Rezepturbild angezeigt und bearbeitet:
  - In der Rezepturanzeige werden die Rezepturdatensätze aus dem internen Speicher des Bediengeräts angezeigt und bearbeitet.
  - Im Rezepturbild werden die Werte der Rezepturvariablen angezeigt und bearbeitet.

  Abhängig von der Projektierung gleichen Sie die in der Rezepturanzeige angezeigten Werte mit den Werten der Rezepturvariablen ab.
- Rezepturspeicher des Bediengeräts
  Im Rezepturspeicher des Bediengeräts werden Rezepturen in Form von Rezepturdatensätzen gespeichert.
- Rezepturvariablen
  Die Rezepturvariablen enthalten Rezepturdaten. Wenn Sie Rezepturen in einem Rezepturbild bearbeiten, werden die Rezepturwerte in Rezepturvariablen abgelegt. Wann die Werte der Rezepturvariablen mit der Steuerung ausgetauscht werden, hängt von der Projektierung ab.

Hinweis

Sie können die Rezepturvariablen mit den Rezepturdatensätzen abgleichen, sodass in beiden dieselben Werte gespeichert sind.

Laden und Speichern von Rezepturdaten

In der Rezepturanzeige werden komplette Rezepturdatensätze vom Rezepturspeicher des Bediengeräts geladen bzw. in den Rezepturspeicher gespeichert.

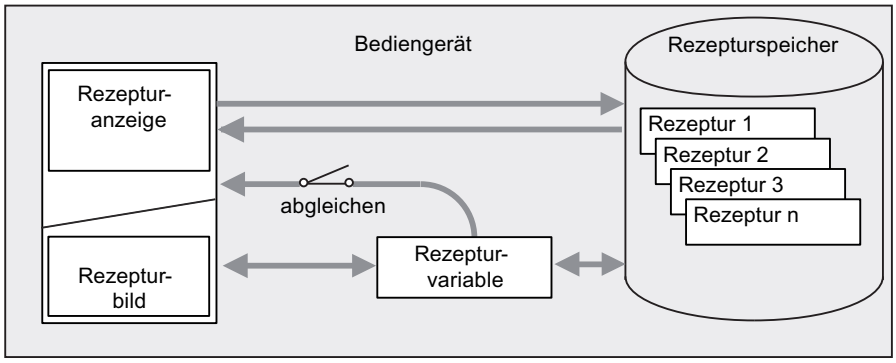

Im Rezepturbild werden die Werte des Rezepturdatensatzes vom Rezepturspeicher in die Rezepturvariablen geladen. Beim Speichern werden die Werte der Rezepturvariablen in einem Rezepturdatensatz in den Rezepturspeicher gespeichert.

Übertragen der Rezepturwerte zwischen Bediengerät und Steuerung

Zwischen Rezepturanzeige und Steuerung werden komplette Rezepturdatensätze übertragen.

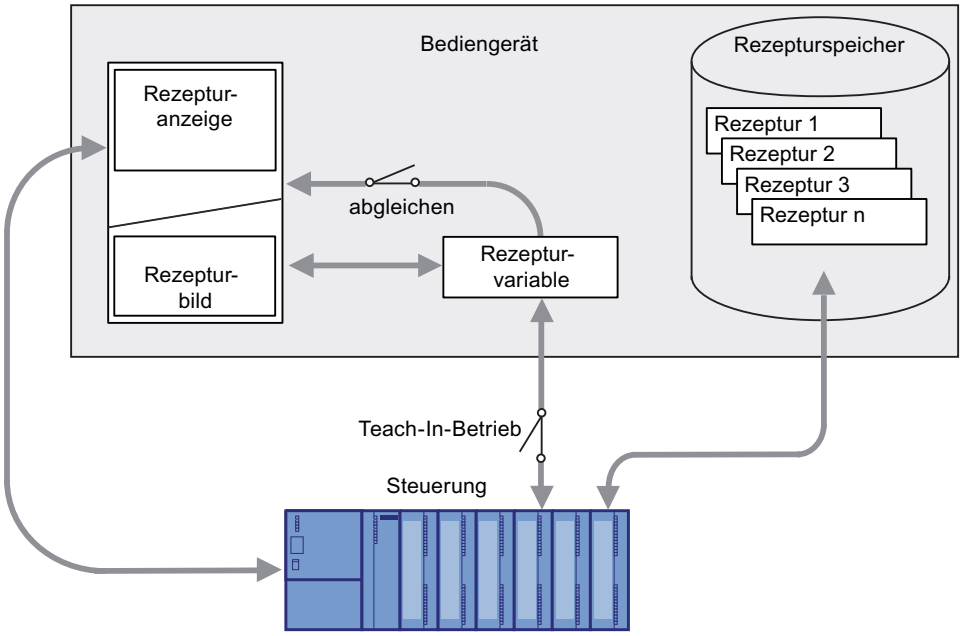

Zwischen Rezepturbild und Steuerung sind abhängig von der Projektierung folgende Übertragungen möglich:

- Übertragen von Rezepturdatensätzen zwischen Rezepturanzeige und Rezepturvariablen
- Sofortiges Übertragen von einzelnen geänderten Werten zwischen Rezepturvariablen und Steuerung. Folgende Einstellungen der Rezeptur sind dafür erforderlich:
  - "Rezepturanzeige und Rezepturvariablen abgleichen" ist aktiviert.
  - "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" ist deaktiviert.

Zwischen Bediengerät und Steuerung können Rezepturdatensätze direkt übertragen werden. Eine Anzeige am Bediengerät ist in diesen Fällen nicht zwingend erforderlich.

Exportieren und Importieren von Rezepturdatensätzen

Die Rezepturdatensätze werden aus dem Rezepturspeicher des Bediengeräts exportiert und auf dem externen Speichermedium in einer CSV-Datei gespeichert. Sie können die Datensätze vom Speichermedium wieder in den Rezepturspeicher importieren.

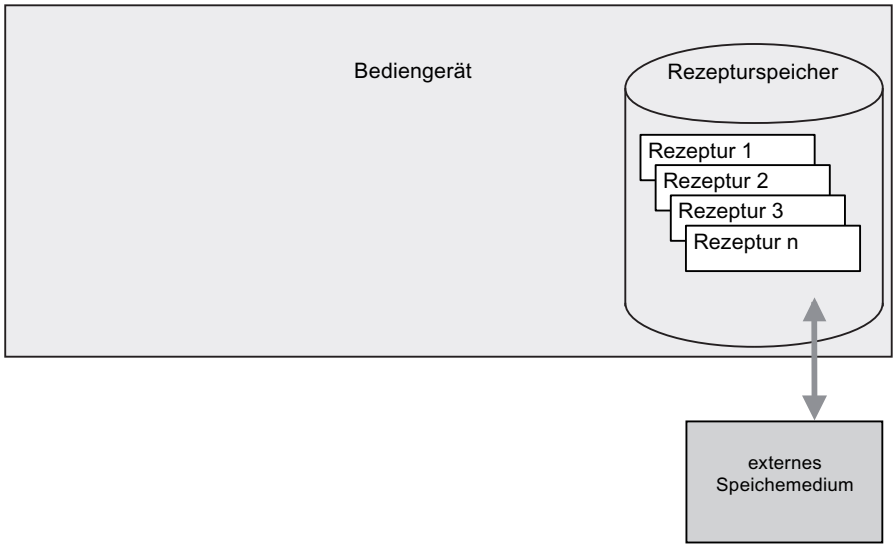

Folgende externe Speichermedien sind abhängig vom Bediengerät verfügbar:

- Kein Speichermedium z. B. bei Basic Panels
- Speicherkarte
- USB-Stick
- Festplatte

6.1.1.6 Konfiguration von Rezepturen

Einleitung

Je nach Einsatzzweck werden Rezepturen unterschiedlich konfiguriert:

- Wenn Sie in Ihrem Projekt Rezepturen mit einer Rezepturanzeige bearbeiten, werden die Werte nur in Rezepturdatensätzen gespeichert.
- Wenn Sie in Ihrem Projekt Rezepturen in einem Rezepturbild bearbeiten, werden die Werte in Rezepturvariablen gespeichert.

Die folgenden Einstellungsmöglichkeiten bestimmen das Zusammenwirken von Rezepturdatensätzen, Rezepturvariablen und Steuerung.

"Rezepturanzeige und Rezepturvariablen abgleichen" deaktiviert

Die Daten eines Rezepturdatensatzes werden nur in der Rezepturanzeige angezeigt und können auch nur dort bearbeitet werden. Wenn Sie diese Daten außerhalb der Rezepturanzeige verwenden, werden die Daten nicht mit der Rezepturanzeige abgeglichen.

"Rezepturanzeige und Rezepturvariablen abgleichen" aktiviert

Wenn Sie Rezepturen in einer Rezepturanzeige und in einem Rezepturbild bearbeiten, können in Runtime Unterschiede zwischen den in der Rezepturanzeige angezeigten Werten und den in den zugehörigen Variablen gespeicherten Werten auftreten. Um dies zu verhindern, müssen Sie die Werte der Rezepturdatensätze mit den Werten der Rezepturvariablen abgleichen.

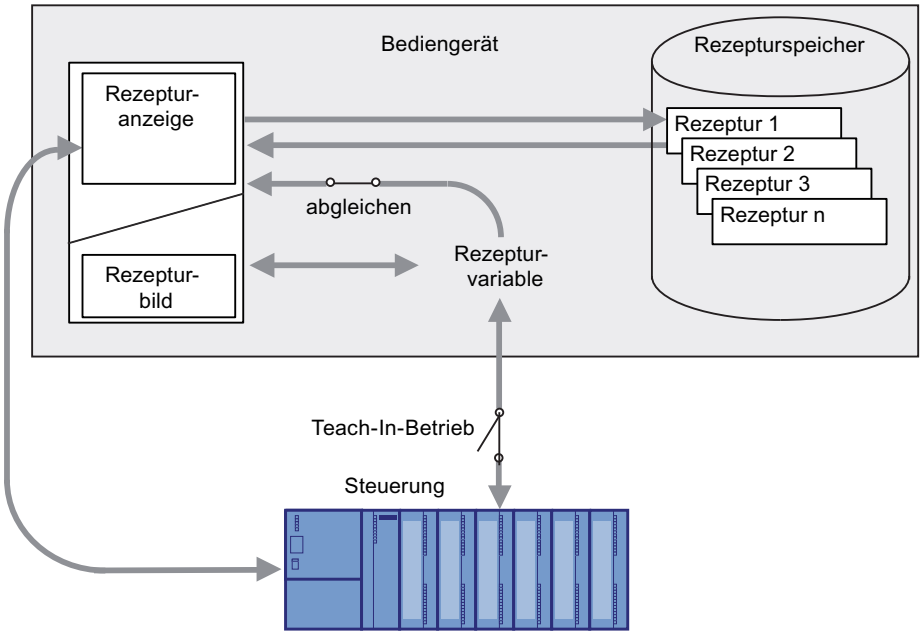

Hinweis

Sie können Rezepturvariablen nur mit der erweiterten Rezepturanzeige abgleichen.

Die Werte der Rezepturanzeige und der zugehörigen Rezepturvariablen werden nicht automatisch abgeglichen. Die Rezepturvariablen und die Rezepturanzeige werden erst abgeglichen, wenn Sie das Bedienobjekt mit der Funktion "RezepturanzeigeSynchronisiereDatensatzMitVariablen" bedienen.

"Rezepturanzeige und Rezepturvariablen abgleichen" aktiviert und "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert

Bei dieser Einstellung werden geänderte Rezepturwerte nicht unmittelbar zwischen den Rezepturvariablen im Rezepturbild des Bediengeräts und der Steuerung abgeglichen.

Zum Abgleichen der Werte muss ein Bedienobjekt mit der Funktion "SchreibeDatensatzInSteuerung" und "LeseDatensatzAusSteuerung" vorhanden sein.

Wenn Rezepturwerte in der Steuerung geändert werden, werden die geänderten Werte sofort im Rezepturbild angezeigt, wenn Sie das Bedienobjekt mit der Funktion "LeseDatensatzAusSteuerung" bedienen.

" Rezepturanzeige und Rezepturvariablen abgleichen" aktiviert und "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert

Bei dieser Einstellung werden geänderte Rezepturwerte unmittelbar zwischen den Rezepturvariablen im Bediengerät und der Steuerung abgeglichen.

Wenn Sie Rezepturwerte im Rezepturbild ändern, werden diese Änderungen sofort von der Steuerung übernommen und beeinflussen unmittelbar den Prozess.

Wenn Rezepturwerte in der Steuerung geändert werden, werden die geänderten Werte sofort im Rezepturbild angezeigt.

6.1.1.7 Besonderheiten bei einigen Geräten

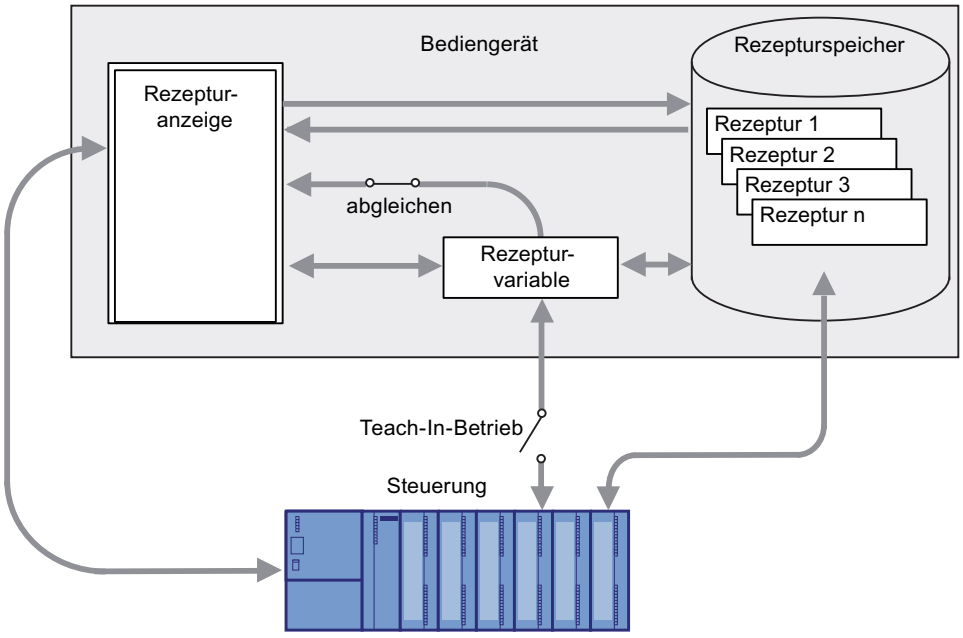

Basic Panels verhalten sich in folgenden Punkten unterschiedlich zu den restlichen Bediengeräten:

- Nur die einfache Rezepturanzeige wird unterstützt.
- Rezepturvariablen sind immer mit der Rezepturanzeige abgeglichen (siehe obige Abbildung): Wenn der Bediener einen Wert eines Rezepturelements in der Rezepturanzeige ändert, ändert sich auch der Wert der zugehörigen Rezepturvariablen.
- Rezepturvariablen werden bei Änderungen nicht automatisch zwischen Steuerung und Bediengerät übertragen. Rezepturvariablen müssen immer manuell übertragen werden im Teach-In-Betrieb.
  Zum Abgleichen der Werte mit der Steuerung muss ein Bedienobjekt mit der Funktion "SchreibeDatensatzInSteuerung" und "LeseDatensatzAusSteuerung" vorhanden sein.

Hinweis

Einschränkungen bei Rezepturvariablen

Bei Basic Panels können Sie Rezepturvariablen generell nicht außerhalb einer Rezeptur verwenden.

Hinweis

Einschränkungen beim Import/Export

Das Exportieren bzw. Importieren von Rezepturen ist für folgende Bediengeräte nicht möglich:

- Basic Panels

Komplette Rezepturdaten, jedoch nicht einzelne Rezepturdatensätze können mit ProSave in das CSV-Format exportiert und importiert und auf das Bediengerät übertragen werden. Dabei wird die Runtime angehalten.

ACHTUNG

Datenverlust bei mehreren Rezepturanzeigen im Bild

Gilt nur für Basic Panels: Wenn in einem Bild zwei oder mehr Rezepturanzeigen die gleiche Rezeptur anzeigen, kommt es zu einem Konflikt beim Datenzugriff.

Datenverlust und unvorhersehbarer Zustand von Rezepturdaten sind die Folge.

Stellen Sie sicher, dass der Bediener in unterschiedlichen Rezepturanzeigen nicht die gleiche Rezeptur auswählen und ändern kann.

- Zeigen Sie in einer Rezepturanzeige jeweils nur eine Rezeptur an.
- Zeigen Sie in jeder Rezepturanzeige jeweils eine andere Rezeptur an.

Hinweis

Speichern von Rezepturen auf Comfort Panels

Das Speichern von Rezepturen auf SD-Karte ist, wenn möglich, dem Speichern von Rezepturen im eingebauten Flash-Speicher vorzuziehen.

Zusammenwirken der Komponenten

In Runtime wirken folgende Komponenten zusammen:

- Rezepturanzeige
  Am Bediengerät werden Rezepturen in der Rezepturanzeige angezeigt und bearbeitet. In der Rezepturanzeige werden die Rezepturdatensätze aus dem internen Speicher des Bediengeräts angezeigt und bearbeitet.
- Rezepturspeicher des Bediengeräts
  Im Rezepturspeicher des Bediengeräts werden Rezepturen in Form von Rezepturdatensätzen gespeichert.
- Rezepturvariablen
  Die Rezepturvariablen enthalten die Werte der Rezepturelemente.

6.1.1.8 Synchronisation von Rezepturdatensätzen mit der Steuerung

Übersicht

Bei der Übertragung von Rezepturdatensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche zu.

Rezepturdatensätze werden immer direkt übertragen. Die Werte der Variablen werden direkt, ohne Zwischenablage, in die projektierten Adressen geschrieben, bzw. aus den projektierten Adressen gelesen.

Übertragungsarten

Für die Übertragung von Rezepturdatensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Koordinierung
- Koordinierte Übertragung über den Bereichszeiger "Datensatz".

Hinweis

Koordinierte Übertragung

Bei der koordinierten Übertragung verhindern Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten.

Voraussetzungen für die koordinierte Übertragung

Folgende Voraussetzungen gelten für die koordinierte Übertragung:

- Im Editor "Kommunikation > Verbindungen" ist der Bereichszeiger "Datensatz" für die gewünschte Verbindung eingerichtet.
- In den Eigenschaften der Rezeptur ist "Koordinierte Übertragung der Datensätze" aktiviert.
- In den Eigenschaften der Rezeptur ist die Verbindung zur Steuerung angegeben, mit der das Bediengerät die Übertragung koordiniert.

Koordinierte Übertragung

Bei der koordinierten Übertragung setzen sowohl Steuerung als auch Bediengerät Status-Bits im gemeinsam benutzten Datenfach.

Die koordinierte Übertragung von Rezepturdatensätzen bietet sich in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Rezepturdatensätzen.
- Die Steuerung wertet Informationen über Rezepturnummer und Rezepturname, sowie Rezepturdatensatznummer und Rezepturdatensatzname aus.
- Die Übertragung von Rezepturdatensätzen wird durch folgende Steuerungsaufträge gestartet:
  - "Datensatz_in_Steuerung_schreiben"
  - "Datensatz_aus_Steuerung_lesen"

6.1.1.9 Editor "Rezepturen"

Einleitung

Im Editor "Rezepturen" erstellen, konfigurieren und bearbeiten Sie Rezepturen, Rezepturelemente und Rezepturdatensätze. Darüber hinaus können Sie im Editor "Rezepturen" Werte in vorhandene Rezepturdatensätze eingeben.

Aufbau des Editors "Rezepturen"

Im oberen Teil des tabellarischen Editors legen Sie Rezepturen an und konfigurieren sie dort oder im Inspektorfenster.

Im unteren Teil des tabellarischen Editors finden Sie folgende Registerkarten:

- Elemente
  Hier definieren Sie über die Tabellenzellen die Rezepturelemente der ausgewählten Rezeptur. Rezepturelemente verschieben Sie innerhalb der Tabelle über die Kontextmenü-Einträge "Aufwärts" und "Abwärts".
- Datensätze
  Hier definieren Sie über die Tabellenzellen die Werte der Rezepturdatensätze der ausgewählten Rezeptur.

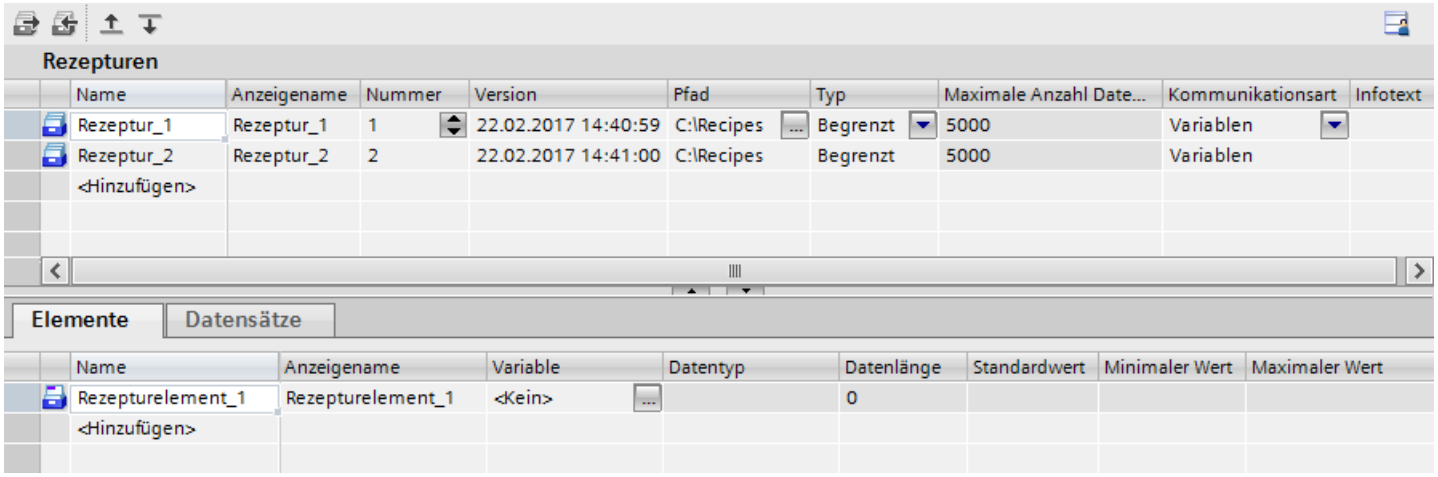

Im Inspektorfenster konfigurieren Sie die jeweils ausgewählte Rezeptur, das Rezepturelement oder den Rezepturdatensatz. Weitere Hinweise zur Projektierung der Bestandteile einer Rezeptur finden Sie unter "Rezepturen projektieren".

Einstellungen Rezepturen

Für die Rezepturen sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Name der Rezeptur	Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die der Bediener einer Rezeptur direkt zuordnen kann, z. B. "Fruchtsaft-Getränk".
Rezepturnummer	Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.
Version	Informationen zur Rezeptur. Voreingestellt ist das Datum und der Zeitpunkt der letzten Änderung der Rezeptur.
Pfad	Legt den Ablageort für die Rezepturen fest. Die Rezepturen werden als Datei abgelegt.
Größentyp [unveränderlich]	Die Rezepturdatensätze sind standardmäßig auf eine vorgegebene Anzahl begrenzt.
Anzahl Datensätze [unveränderlich]	Maximale Anzahl der Rezepturdatensätze einer Rezeptur in Runtime. Die Anzahl wird durch den Rezepturspeicher des Bediengeräts begrenzt.
Kommunikationsart [unveränderlich]	Die Rezepturdatensätze werden direkt in die Adressen der Rezepturvariablen geschrieben oder von dort gelesen.
Tooltip	Tooltipt zur Rezeptur, der dem Bediener in Runtime angezeigt wird.

Hinweis

Pfad

Der Ablageort hängt davon ab, mit welchen Speichermedien das Bediengerät ausgestattet ist.

Basic Panels

Diese Bediengeräte besitzen keinen externen Speicher. Rezepturen werden immer im internen Flash-Speicher abgelegt. Die Einstellung "Pfad" steht daher nicht zur Verfügung.

Einstellungen Rezepturelemente

In der Registerkarte "Elemente" sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Name des Rezepturelements	Identifiziert ein Rezepturelement innerhalb der Rezeptur eindeutig. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die Sie eindeutig zuordnen können, z. B. Achsenbezeichnungen an einer Maschine oder Zutaten wie "Aroma".
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die der Bediener direkt zuordnen kann, z. B. "Fruchtsaft- Aroma".
Rezepturvariable	Eine zugeordnete Variable speichert in Runtime den aktuellen Wert des Rezepturelements im Rezepturdatensatz.
Datentyp	Datentyp der Rezepturvariable.
Datenlänge [unveränderlich]	Datenlänge der Rezepturvariable, abhängig vom Datentyp.
Textliste	In einer Textliste wird einem Wert oder Wertebereich Text zugeordnet. Diesen Text können Sie z. B. in einem Ausgabefeld anzeigen lassen. Die zugeordnete Rezepturvariable muss den Datentyp einer Zahl besitzen. Der Variablenwert muss sich innerhalb des Wertebereichs der Textliste befinden.
Standardwert	Wird als Standardeintrag verwendet, wenn Sie einen neuen Rezepturdatensatz anlegen.
Minimaler Wert [unveränderlich]	Kleinster darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.
Maximaler Wert [unveränderlich]	Größter darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.
Nachkommastellen	Bestimmt, auf wie viele Stellen genau ein Dezimalwert gerundet wird z. B. 3 Nachkommastellen und umgekehrt mit welcher Zehnerpotenz ein Ganzzahlwert multipliziert wird z. B. 1000.
Tooltip	Tooltip zum Rezepturelement, der dem Bediener in Runtime angezeigt wird.

Einstellungen Rezepturdatensätze

In der Registerkarte "Datensätze" sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Name des Rezepturdatensatzes	Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Produktnummern, die der Bediener einem Produkt direkt zuordnen kann, z. B. "Gelber Fruchsaft E231".
Rezepturdatensatznummer	Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.
Rezepturelemente 1 bis n	Zu jedem Rezepturelement können Sie bereits während der Projektierung verschiedene Werte hinterlegen. Ein Wert bildet zusammen mit den Werten der anderen Rezepturelemente jeweils einen Rezepturdatensatz. Sie können mehrere Rezepturdatensätze hinterlegen. Wenn in den Transfereinstellungen aktiviert, werden beim Laden des Projekts auf das Bediengerät die Rezepturdatensätze mit übertragen und auf dem Bediengerät bereits existierende Datensätze überschrieben.
Kommentar	Kommentar zum Rezepturdatensatz

6.1.2 Rezepturen in Runtime

6.1.2.1 Rezepturbild und Rezepturanzeige

Grundlagen zum Rezepturbild

Einleitung

Das Rezepturbild ist ein Bild, das eine individuelle Eingabemaske für die Rezepturen enthält. Die Eingabemaske enthält E/A-Felder und andere Anzeige- und Bedienobjekte. Die Rezepturfunktionalität ist mit Systemfunktionen realisiert, z. B. das Speichern von Rezepturdatensätzen.

Das folgende Bild zeigt ein Beispiel für ein Rezepturbild:

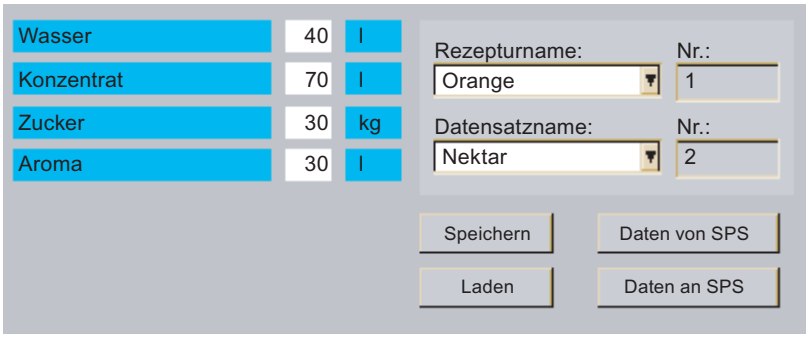

Hinweis

Verfügbarkeit

Bei Basic Panels und OP73, OP77A, TP177A (Portrait) werden keine Rezepturbilder unterstützt.

Rezepturbilder gestalten

Die Projektierung eines Rezepturbildes bietet Ihnen individuelle Gestaltungsmöglichkeiten. Sie können große Rezepturen themenspezifisch auf mehrere Bilder verteilen und anschaulich darstellen, z. B. mit grafischen Anzeige- und Bedienobjekten:

- Thematische Verteilung über mehrere Bilder
  - Sie können Rezepturdatensätze mit vielen Einträgen über mehrere Bilder verteilen. Z. B. projektieren Sie für jeden Anlagenbereich ein Bild mit den dazugehörigen Eingabemasken für die Rezepturdatensätze.

  Bei Bediengeräten mit kleinem Display ist die Verteilung auf mehrere Bilder sinnvoll, weil Sie damit in Runtime z. B. das Scrollen in einer Tabelle vermeiden.
- Visuelle Nachbildung der Maschine
  Sie bilden in einem Bild Ihre Maschine mit grafischen Bildobjekten visuell nach. Damit stellen Sie Parametrierungseinstellungen anschaulicher dar, indem Sie E/A-Felder direkt neben Maschinenelementen wie Achsen oder Führungsschienen platzieren. Sie stellen damit einen direkten Bezug zwischen den Werten und der Maschine her.

Rezepturwerte abgleichen

Damit Sie die Rezepturdatensatzwerte außerhalb der erweiterten Rezepturanzeige in den projektierten E/A-Feldern ändern können, aktivieren Sie in den Rezeptureigenschaften unter "Eigenschaften > Synchronisation" die Option "Rezepturanzeige und Rezepturvariablen abgleichen":

Die Option "Rezepturanzeige und Rezepturvariablen abgleichen" ist für die einfache Rezepturanzeige nicht verfügbar.

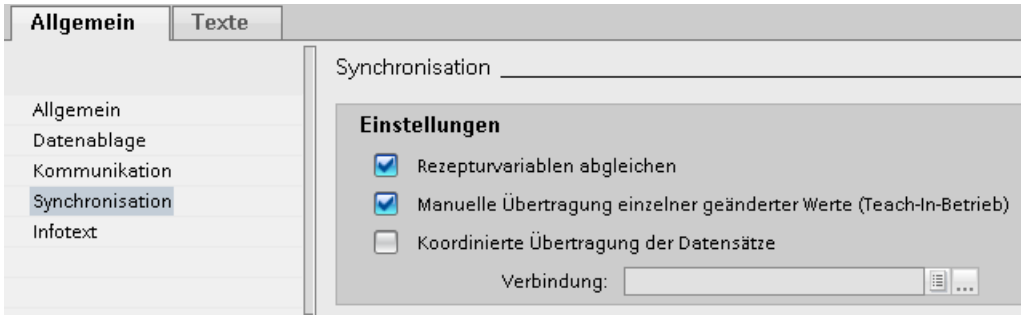

Bild 6-1 Rezepturen_Synchronisation_Advanced

Rezepturwerte automatisch übertragen

Wenn in Runtime die eingegebenen Rezepturwerte sofort an die angeschlossene Steuerung übertragen werden sollen, deaktivieren Sie unter "Eigenschaften > Extras" "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".

Wenn Sie in Runtime die sofortige Übertragung der eingegebenen Rezepturwerte einschalten und ausschalten wollen, projektieren Sie die Systemfunktion "SetzeRezepturvariablen".

Systemfunktionen

Die Systemfunktionen zum Laden, Speichern und Übertragen von Rezepturdatensätzen und Rezepturen finden Sie in der Gruppe "Rezepturen".

Für die Bedienung eines Rezepturbildes stehen Ihnen folgende Systemfunktionen zur Verfügung:

- ImportiereDatensaetze
- ExportiereDatensaetze
- LadeDatensatz
- LeseDatensatzAusSteuerung
- LöscheDatensatz
- SpeichereDatensatz
- SchreibeDatensatzInSteuerung
- SchreibeDatensatzvariablenInSteuerung
- SetzeRezepturvariablen
- LeseDatensatzvariablenAusSteuerung

Wenn Sie im Rezepturbild eine Rezepturanzeige verwenden, stehen Ihnen zur Bedienung der Rezepturanzeige folgende Systemfunktionen zur Verfügung:

- RezepturanzeigeSpeichereDatensatz
- RezepturanzeigeSpeichernUnterDatensatz
- RezepturanzeigeSynchronisiereDatensatzMitVariablen
- RezepturanzeigeLoescheDatensatz
- RezepturanzeigeDatensatzNeu
- RezepturanzeigeLeseDatensatzAusSteuerung
- RezepturanzeigeSchreibeDatensatzInSteuerung
- RezepturanzeigeUmbenennenDatensatz
- RezepturanzeigeZeigeInfotext
- RezepturanzeigeMenue (nur einfache Rezepturanzeige)
- RezepturanzeigeOeffnen (nur einfache Rezepturanzeige)
- RezepturanzeigeZurueck (nur einfache Rezepturanzeige)

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Grundlagen zur Rezepturanzeige

Einleitung

Die Rezepturanzeige ist ein Anzeige- und Bedienobjekt von WinCC. Mit der Rezepturanzeige können Sie Datensätze anzeigen, bearbeiten und verwalten.

Die Darstellung der Rezepturanzeigen ist abhängig von dem Gerät und der Geräteversion.

Bediengeräte mit einfacher Rezepturanzeige

Bediengerät	Typ
Basic Panels	KP300 Basic mono PN KP400 Basic color PN
	KTP400 Basic mono PN KTP400 Basic color PN KTP400 Basic mono PN Portrait KTP400 Basic color PN Portrait
	KTP600 Basic
	KTP1000 Basic
	TP1500 Basic

1) Die Darstellung der einfachen Rezepturanzeige ist auf die erweiterte Rezepturanzeige umstellbar

Bediengeräte mit erweiterter Rezepturanzeige vor Geräteversion V13

Bediengerät	Typ
Mobile Panels	Panel 177
	Panel 277 Panel 277 IWLAN V2 Panel 277F IWLAN V2 Panel 277F IWLAN (RFID Tag)

1) Die Darstellung der erweiterten Rezepturanzeige ist auf die einfache Rezepturanzeige umstellbar

Bediengeräte mit erweiterter Rezepturanzeige ab Geräteversion V13

Bediengerät	Typ
Basic Panels	KTP400 Basic PN KTP400 Basic PN Portrait
	KTP700 Basic
	KTP900 Basic
	KTP1200 Basic

Bediengerät	Typ
Comfort Panels	KP400 Comfort KTP400 Comfort
	KP700 Comfort TP700 Comfort
	KP900 Comfort TP900 Comfort
	KP1200 Comfort TP1200 Comfort
	KP1500 Comfort TP1500 Comfort
	TP1900 Comfort
	TP2200 Comfort
Mobile Panels	KTP400F Mobile
	KTP700 Mobile KTP700F Mobile
	KTP900 Mobile KTP900F Mobile

Siehe auch

Grundlagen zum Rezepturbild (Seite 1138)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

6.1.2.2 Einfache Rezepturanzeige

Beschreibung Einfache Rezepturanzeige

Rezepturanzeige

Die einfache Rezepturanzeige ist ein Anzeige- und Bedienobjekt, das zum Verwalten der Rezepturdatensätze verwendet wird. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform.

Die angezeigten Schaltflächen und Informationen in den Spalten sind einstellbar.

Die in der Rezepturanzeige angezeigten oder eingegebenen Werte werden in Rezepturdatensätzen gespeichert. Über Schaltflächen kann der angezeigte Rezepturdatensatz in die Steuerung geschrieben oder Werte von der Steuerung eingelesen werden.

Aufbau der Anzeige

Die einfache Rezepturanzeige besteht aus den folgenden Anzeigebereichen:

- Rezepturliste
- Datensatzliste
- Elementliste

Das folgende Bild zeigt eine Datensatzliste:

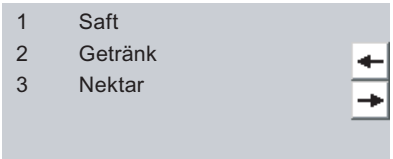

Jeder Anzeigebereich wird von der einfachen Rezepturanzeige separat am Bediengerät dargestellt. Jeden dieser Anzeigebereiche bedienen Sie mit einem kontextspezifischen Menü.

Standardmäßig startet die einfache Rezepturanzeige mit der Rezepturliste.

Bedienung

Für die Bedienung der einfachen Rezepturanzeige haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Rezepturdatensatz neu anlegen, ändern, kopieren oder löschen
- Rezepturdatensatz aus Steuerung lesen oder zur Steuerung übertragen

Anzeigebereich und Kontextmenü bedienen

Zur Bedienung der einfachen Rezepturanzeige wechseln Sie zwischen den Anzeigebereichen und den Kontextmenüs.

Tabelle 6-1 Bedienung des Anzeigebereichs

Schaltfläche	Taste	Funktion
	<Enter>	Der nächstniedrige Anzeigebereich, d. h. die Datensatzliste oder die Elementliste wird geöffnet.
←	<Esc>	Der vorherige Anzeigebereich wird geöffnet.
	<INS>	Legt einen neuen Datensatz zur selektierten Rezeptur an, wenn die Liste der Rezepturen oder Rezepturdatensätze angezeigt wird. Wechselt dann zur Liste der Rezepturelemente. Voraussetzung: "Egenschaften >Allgemein > Bearbeitungsmodus" ist aktiviert. Die Schaltfläche kann mit der Funktion "Taste SimulateSystemKey" simuliert werden auch auf Geräten ohne Tasten.
	<DEL>	Löscht den selektierten Rezepturdatensatz in der Liste der Rezepturdatensätze. Voraussetzung: "Egenschaften >Allgemein > Bearbeitungsmodus" ist aktiviert.

Schaltfläche	Taste	Funktion
	<Auf>/<Ab>	Wählt den vorherigen/nächsten Eintrag aus.
	<Bild Auf>/<Bild Ab>	Eine Anzeigeseite wird nach oben oder unten geblättert.
	<Pos 1>/<Ende>	Wählt den ersten/letzten Eintrag aus. Der erste/letzte Eintrag wird markiert.

Tabelle 6-2 Bedienung der Kontextmenüs

Schaltfläche	Taste	Funktion
→	<Rechts>	Das Kontextmenü des Anzeigebereichs wird geöffnet.
←	<Esc>	Das Menü wird geschlossen. Der Anzeigebereich wird geöffnet.
	Eingabe der Nummer des Menübefehls	Der Menübefehl wird ausgeführt.

Kontextspezifische Menüs der einfachen Rezepturanzeige

Zu jedem Anzeigebereich können Sie mit der Schaltfläche → eine Befehlsauswahl aufrufen. In der Befehlsauswahl werden die Befehle angezeigt, die für den Anzeigebereich verfügbar sind. Jedem Befehl ist eine Nummer zugeordnet. Der Befehl wird ausgeführt, sobald Sie diese Nummer eingeben. Alternativ selektieren Sie den Befehl und betätigen Sie die Taste <Return>.

Kontextspezifische Menüs der Rezepturliste

Menübefehl	Funktion
Neu	Für die markierte Rezeptur wird ein neuer Rezepturdatensatz erstellt. Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt.
Tooltip anzeigen	Der für die Rezeptur projektierte Tooltip wird angezeigt.
Öffnen	Die Datensatzliste der markierten Rezeptur wird geöffnet.

Kontextspezifische Menüs der Rezepturdatensatzliste

Menübefehl	Funktion
Neu	Ein neuer Rezepturdatensatz wird erstellt. Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt.
Löschen	Der markierte Datensatz wird gelöscht.
Speichern unter	Der markierte Datensatz wird unter einem anderen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.
Umbenennen	Der markierte Datensatz wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.
Öffnen	Die Elementliste des markierten Datensatzes wird geöffnet.
Zurück	Die Rezepturliste wird geöffnet.

Kontextspezifische Menüs der Rezepturelementliste

Menübefehl	Funktion
Speichern	Der markierte Datensatz mit dem Rezepturelement wird gespeichert.
Zur Steuerung	Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen.
Von Steuerung	Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.
Speichern unter	Der markierte Datensatz wird unter einem neuen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.
Tooltip anzeigen	Der für das Rezepturelement projektierte Tooltip wird angezeigt.
Umbenennen	Das markierte Rezepturelement wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.
Zurück	Die Datensatzliste wird geöffnet.

Kontextspezifische Menüs der Datensatzliste

Hinweis

Bediengeräteabhängigkeit

Bei Basic Panels sind zusätzlich die folgenden Menübefehle projektiert.

Menübefehl	Funktion
Zur Steuerung	Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen.
Von Steuerung	Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.

Werte aktualisieren und anzeigen

Hinweis

Bearbeiteter Rezepturdatensatz wird im Hintergrund geändert

Gilt nur für Basic Panels: Wenn ein Bediener einen Rezepturdatensatz geändert hat und ein Steuerungsauftrag einen beliebigen Rezepturdatensatz dieser Rezeptur lesen oder schreiben möchte, wird der Steuerungsauftrag angehalten und eine Systemmeldung ausgegeben. Hingegen wird der geänderte Wert sofort angezeigt, wenn nur der Steuerungauftrag und kein Bediener Rezepturdaten geändert hat.

Gilt nicht für Basic Panels: Wenn ein Bediener einen Rezepturdatensatz geändert hat und ein Steuerungsauftrag Werte des betreffenden Rezepturdatensatzes ändert, wird die Rezepturanzeige nicht automatisch aktualisiert. Um die Rezepturanzeige zu aktualisieren, müssen Sie den betreffenden Rezepturdatensatz neu anwählen.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Konfigurationsmöglichkeiten der einfachen Rezepturanzeige (Seite 1146)

Konfigurationsmöglichkeiten der einfachen Rezepturanzeige

Sie können das Verhalten der einfachen Rezepturanzeige im Inspektorfenster der Rezepturanzeige festlegen. Einige Geräte z. B. Basic Panels unterstützen nur die einfache Rezepturanzeige.

Bei Geräten mit einfacher Rezepturanzeige beachten Sie Folgendes:

1. Wählen Sie im Inspektorfenster unter "Eigenschaften > Darstellung > Modus" als "Anzeigentyp" "Einfache Ansicht".
2. Der Bereich "Eigenschaften > Einfache Ansicht" enthält zusätzliche Eigenschaften, die nur für die einfache Rezepturanzeige gelten.
3. Alle anderen Eigenschaften gelten auch für die erweiterte Rezepturanzeige.

Rezepturdatensatzwerte nur anzeigen

Um in einer Rezepturanzeige die Rezepturdaten nur zur Kontrolle anzuzeigen, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie in der Gruppe "Allgemein" "Bearbeitungsmodus".

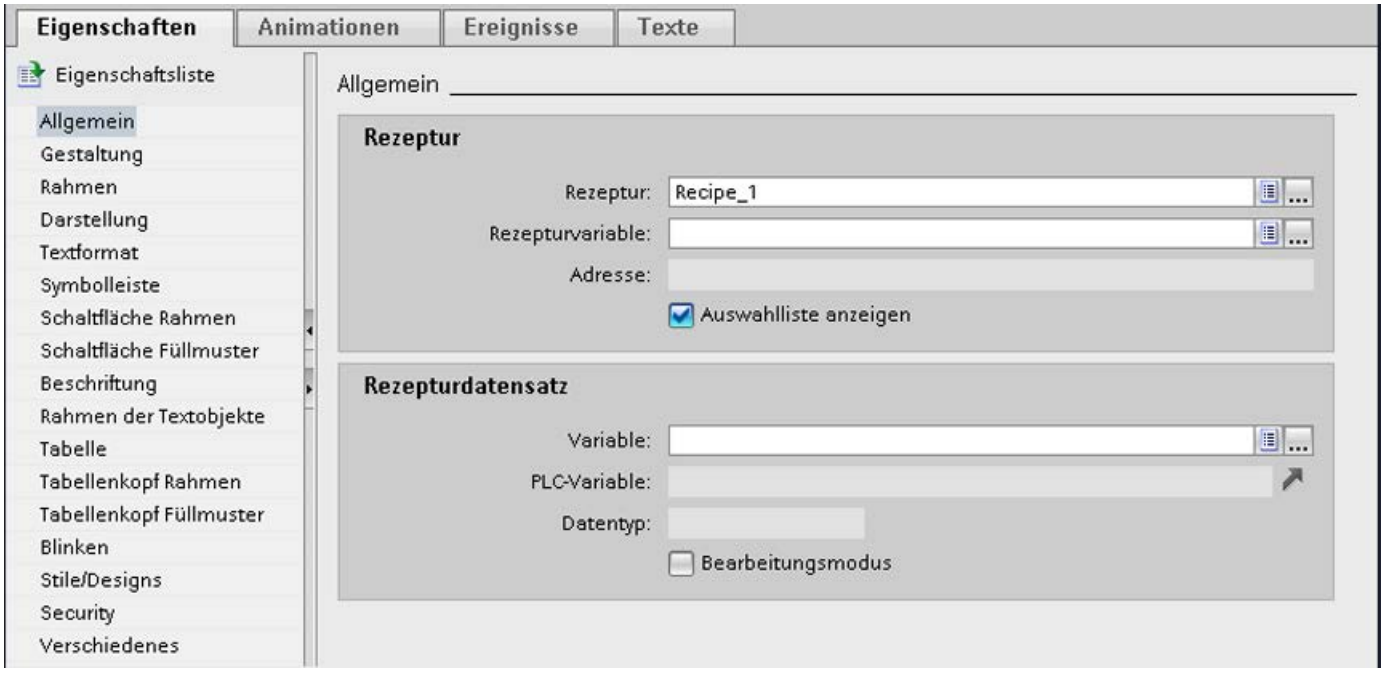

Sie können keine Rezepturdaten neu anlegen, umbenennen, editieren oder löschen.

Rezepturdatensatznummer/-name in Variable schreiben

Sie können in der erweiterten Rezepturanzeige an den Rezepturdatensatz jeweils eine Variable projektieren. Abhängig vom Datentyp "String" oder "Int" der Variablen, wird der Name oder die Nummer des Rezepturdatensatz in der Variablen gespeichert. Umgekehrt können Sie die Variable auch zur Auswahl des Rezepturdatensatzes verwenden, indem Sie den entsprechenden Wert vorgeben. Die Variablen können Sie z. B. als Parameter für eine Systemfunktion übergeben.

Gehen Sie folgendermaßen vor:

1. Geben Sie unter "Eigenschaften > Allgemein > Rezepturdatensatz" in das Feld "Variable" eine Variable vom Typ "Int" ein.

Die Nummer des Rezepturdatensatzes wird jeweils in einer Variablen gespeichert.

Ereignis an die Rezepturanzeige projektieren

Hinweis

Ereignisse und Schaltflächen

Wenn mindestens eine Schaltfläche aktiviert ist, wird das Register "Ereignisse" ausgeblendet.

Um ein Ereignis an die Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Bilder" die ins Bild eingefügte Rezepturanzeige.
   Die Eigenschaften der Rezepturanzeige werden im Inspektorfenster angezeigt.
2. Deaktivieren Sie unter "Eigenschaften > Symbolleiste" und "Eigenschaften > Einfache Ansicht" alle Schaltflächen.
3. Klicken Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" auf das Ereignis, das Sie projektieren möchten z. B. "Aktivieren".
4. Projektieren Sie eine Funktionsliste an das Ereignis.

Wenn der Bediener die Rezepturanzeige aktiviert, wird die Funktionsliste abgearbeitet.

Eigenschaften der Rezepturanzeige animieren

Um eine Animation einer Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Bilder" die ins Bild eingefügte Rezepturanzeige.
   Die Eigenschaften der Rezepturanzeige werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Animationen".

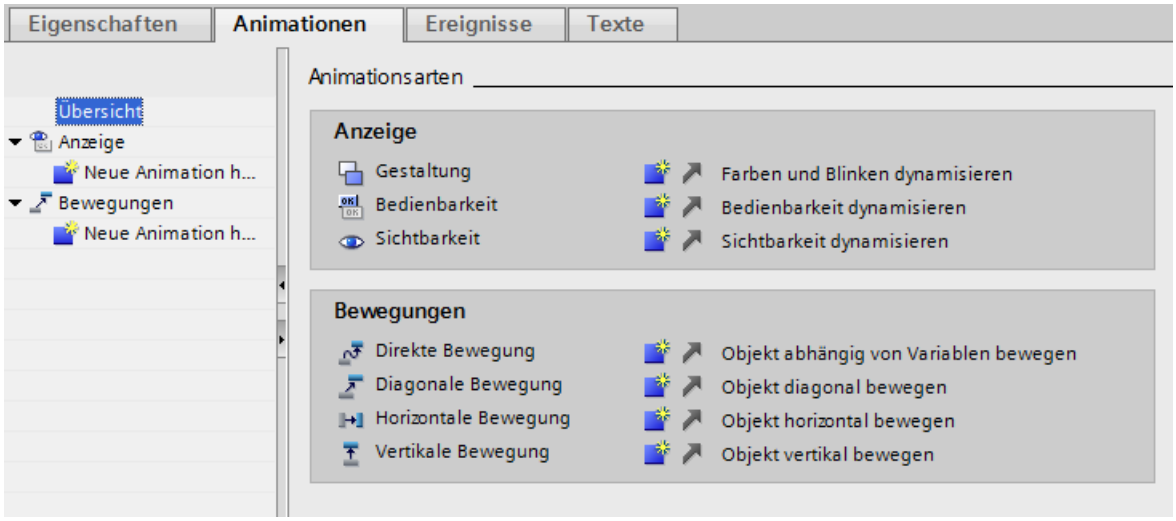

3. Binden Sie an eine oder mehrere der folgenden Eigenschaften eine Variable:
   - X-Position und y-Position
   - Gestaltung: Farben, Blinken
   - Bedienbarkeit
   - Sichtbarkeit
     Im Bereich "Animationen > Übersicht" sind alle Animationen tabellarisch zusammengefasst. Unter "Animationen > Variablenanbindungen > Variablenanbindung" können Sie neben Sichtbarkeit und Position auch an Höhe und Breite eine Variable anbinden.

Hinweis

Animationen und Schaltflächen

Beim Übersetzen des Projekts für Windows-CE-Bediengeräte erscheint eine Fehlermeldung, wenn nicht alle Schaltflächen deaktiviert sind.

Einschränkungen bei der einfachen Rezepturanzeige

Bei der einfachen Rezepturanzeige sind folgende Funktionen nicht möglich:

- Rezepturanzeige und Rezepturvariablen abgleichen
- Rezepturnummer/-name in Variable schreiben
- Statuszeile anzeigen
- Datensatznummer anzeigen

- Beschriftung anzeigen
- Tabelle anzeigen

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Beschreibung Einfache Rezepturanzeige (Seite 1142)

Konfiguration von Rezepturen (Seite 1130)

Einfache Rezepturanzeige bedienen

Einfache Rezepturanzeige mit Maus oder Touchpad bedienen

Um die einfache Rezepturanzeige mit Maus oder Touchpad zu bedienen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Rezepturanzeige die gewünschte Rezeptur.
2. Klicken Sie auf die Schaltfläche →.
   Das Kontextmenü wird geöffnet.
3. Wählen Sie den gewünschten Menübefehl.
   Der gewünschte Menübefehl wird ausgeführt.

Einfache Rezepturanzeige mit Tastatur bedienen

Um die einfache Rezepturanzeige mit der Tastatur zu bedienen, gehen Sie folgendermaßen vor:

1. Drücken Sie so oft die Taste <Tab>, bis die einfache Rezepturanzeige selektiert ist.
2. Wählen Sie mit den Cursortasten die gewünschte Rezeptur.
3. Drücken Sie die Taste <Rechts>.
   Das Kontextmenü wird geöffnet.
4. Drücken Sie so oft die Cursortaste <Ab>, bis der gewünschte Menübefehl ausgewählt ist.
5. Bestätigen Sie den Menübefehl mit der Taste <Enter>.

Tastaturkombinationen für die einfache Rezepturanzeige

Folgende Tastaturkombinationen sind in Runtime für die einfache Rezepturanzeige aktiviert, wenn im ES "Tastaturbedienung aktivieren" aktiviert ist.

Tastatenkombination	Auswirkung	Menübefehl
<Einfg>	Erzeugt einen neuen Rezepturdatensatz	Neu
<Entf>	Der angezeigt Rezepturdatensatz wird gelöscht.	Löschen

Rezepturanzeige mit Funktionstasten bedienen

Sie können die Rezepturanzeige über Funktionstasten bedienen, z. B., wenn das Bediengerät keine Touch-Funktionalität besitzt. An die Funktionstasten des Bediengeräts können Funktionen, z. B. "SpeichereDatensatz", projektiert sein.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Rezepturdatensätze verwalten

Rezepturdatensätze verwalten

Zur Verwaltung der einfachen Rezepturanzeige haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Neue Rezepturdatensätze erstellen
- Rezepturdatensätze kopieren
- Rezepturdatensätze ändern
- Rezepturdatensätze löschen

Neuen Rezepturdatensatz erstellen

Um einen neuen Rezepturdatensatz zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen neuen Rezepturdatensatz erstellen wollen.
2. Wählen Sie im Kontextmenü der Rezepturliste den Befehl "Neu".
   Ein neuer Datensatz mit der nächsten freien Nummer wird angelegt.
   Die Elementliste des neuen Rezepturdatensatzes wird geöffnet.
3. Geben Sie die Werte für die Elemente des Rezepturdatensatzes ein.
   Abhängig von der Projektierung sind die Werte des Rezepturdatensatzes bereits mit Standardwerten vorbelegt.
4. Wählen Sie im Kontextmenü der Elementliste den Befehl "Speichern".
   Der Dialog "Speichern unter" wird geöffnet.
5. Geben Sie Name und Nummer des Rezepturdatensatzes ein.
6. Klicken Sie auf die Schaltfläche "OK".
   Der Datensatz wird unter dem neuen Namen gespeichert.
   Wenn der Datensatz bereits vorhanden ist, wird ein Dialog geöffnet. Legen Sie in diesem Dialog fest, ob der vorhandene Datensatz überschrieben wird.

Rezepturdatensatz kopieren

Um einen Rezepturdatensatz zu kopieren, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz kopieren wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, von dem Sie eine Kopie speichern wollen.
3. Wählen Sie im Kontextmenü der Datensatzliste den Befehl "Speichern unter".
   Der Dialog "Speichern unter" wird geöffnet. Der Rezepturdatensatz erhält automatisch die nächste freie Rezepturdatensatznummer.
4. Geben Sie bei Bedarf Name und Nummer des Rezepturdatensatzes ein.
5. Klicken Sie auf die Schaltfläche "OK".

Rezepturdatensatz ändern

Um einen Rezepturdatensatz zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz ändern wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie ändern wollen.
3. Wählen Sie den gewünschten Rezepturdatensatz.
   Die Elementliste des Rezepturdatensatzes wird angezeigt.
4. Ersetzen Sie die vorhandenen Werte durch die neuen Werte.
5. Wählen Sie im Kontextmenü der Elementliste den Befehl "Speichern".

Rezepturdatensatz löschen

Um einen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, aus der Sie einen vorhandenen Rezepturdatensatz löschen wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie löschen wollen.
3. Wählen Sie im Kontextmenü der Datensatzliste den Befehl "Löschen".
4. Bestätigen Sie die Sicherheitsabfrage, um den Datensatz zu löschen.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Rezepturdatensatz aus Steuerung lesen

Einleitung

In Runtime können Sie direkt in der Anlage die Werte ändern, die auch in den Rezepturen im Bediengerät hinterlegt sind. Dies ist z. B. der Fall, wenn ein Ventil direkt an der Anlage weiter geöffnet wurde, als in der Rezeptur hinterlegt ist. Die Werte der im Bediengerät gespeicherten Rezepturdatensätze stimmen dann möglicherweise nicht mehr mit den Werten in der Steuerung überein.

Sie können die Werte der Rezepturvariablen aus der Steuerung lesen und in einen Rezepturdatensatz schreiben.

Die eingelesenen Werte werden in den Rezepturdatensatz geschrieben, der gerade am Bediengerät angezeigt wird.

Vorgehen

Um einen Rezepturdatensatz aus der Steuerung zu lesen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Rezeptur am Bediengerät.
   Die Datensatzliste wird geöffnet.
2. Öffnen Sie die Elementliste des Rezepturdatensatzes, in den Sie die Werte aus der Steuerung übernehmen wollen.
3. Wählen Sie im Kontextmenü der Elementliste den Befehl "Von Steuerung".
   Die Werte werden aus der Steuerung gelesen und im aktuellen Rezepturdatensatz angezeigt.
4. Wenn Sie die Werte speichern wollen, wählen Sie den Befehl "Speichern" oder "Speichern unter".

Ergebnis

Die Werte werden aus der Steuerung gelesen, am Bediengerät dargestellt und im Rezepturdatensatz gespeichert.

Hinweis

Basic Panels

Bei Basic Panels kann der Menübefehl "Von Steuerung" auch für die Datensatzliste projektiert worden sein. In diesem Fall können Sie den Menübefehl "Von Steuerung" auch in der Datensatzliste wählen.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Rezepturdatensatz zur Steuerung übertragen

Einleitung

Damit die Werte eines mit der Rezepturanzeige geänderten Datensatzes in der Steuerung gültig sind, müssen Sie die Werte zur Steuerung übertragen.

Es werden immer die in der Rezepturanzeige angezeigten Werte zur Steuerung übertragen.

Vorgehen

Um einen Rezepturdatensatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die gewünschte Rezeptur.
   Die Datensatzliste wird geöffnet.
2. Öffnen Sie die Elementliste des Rezepturdatensatzes, dessen Werte Sie zur Steuerung übertragen wollen.
3. Wählen Sie im Kontextmenü der Elementliste den Befehl "Zur Steuerung".

Ergebnis

Die Werte des Rezepturdatensatzes werden an die Steuerung übertragen.

Hinweis

Basic Panels

Bei Basic Panels kann der Menübefehl "Von Steuerung" auch für die Datensatzliste projektiert worden sein. In diesem Fall können Sie den Menübefehl "Von Steuerung" auch in der Datensatzliste wählen.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Synchronisation von Rezepturdatensätzen mit der Steuerung (Seite 1134)

Übertragen von Rezepturdatensätzen (Seite 1127)

6.1.2.3 Erweiterte Rezepturanzeige (vor V13)

Beschreibung Erweiterte Rezepturanzeige (vor V13)

Rezepturanzeige

Die erweiterte Rezepturanzeige ist ein Anzeige- und Bedienobjekt, das zum Verwalten der Rezepturdatensätze verwendet wird. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform.

Die angezeigten Schaltflächen, Überschriften und Informationen in den Spalten sind einstellbar.

Die in der Rezepturanzeige angezeigten oder eingegebenen Werte werden in Rezepturdatensätzen gespeichert.

Die Darstellung der Rezepturanzeigen ist abhängig von dem Gerät und der Geräteversion.

Verwendung

Mit der Rezepturanzeige können Sie Datensätze anzeigen, bearbeiten und verwalten.

Aufbau der Anzeige

Das folgende Bild zeigt ein Beispiel für die erweiterte Rezepturanzeige auf einem Bediengerät vor der Geräteversion V13.

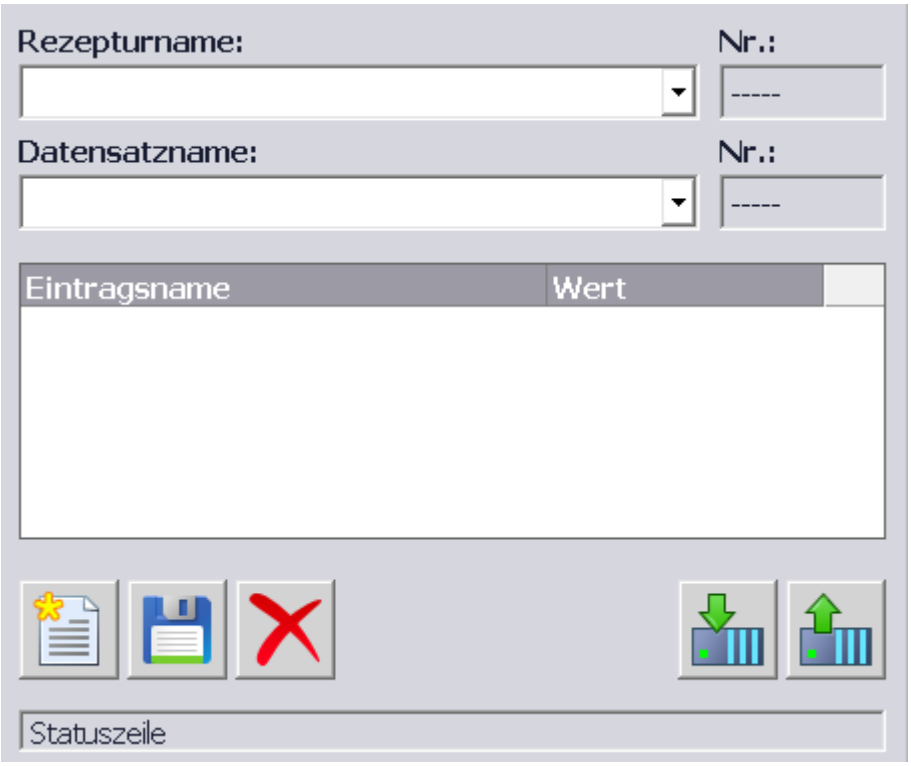

Bedienung

Abhängig von der Projektierung haben Sie folgende Möglichkeiten:

- Rezepturdatensätze anlegen, ändern, kopieren oder löschen
- Rezepturdatensätze aus der Steuerung lesen oder zur Steuerung übertragen
- Rezepturdatensätze mit den zugehörigen Rezepturvariablen abgleichen

Bedienobjekte

In der Rezepturanzeige können folgende Bedienobjekte projektiert sein:

Bedienobjekt	Tasten-kombination	Funktion
		Der projektierte Tooltip wird angezeigt.
	<Strg+Leertaste>	Ein neuer Rezepturdatensatz wird erstellt. Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt.
	<Strg+Enter>	Die angezeigten Werte des Rezepturdatensatzes werden gespeichert. Der Speicherort ist durch das Projekt vorgegeben.
	<Strg+*>	Der Rezepturdatensatz wird unabhängig von der Rezepturanzeige unter einem anderen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.
	<Strg+Entf>	Der angezeigte Rezepturdatensatz wird gelöscht.
	<Strg+=>	Wenn der Wert einer Rezepturvariablen einen aktuelleren Wert als die Rezepturanzeige besitzt, so wird dieser Wert in die Rezepturanzeige übernommen. Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Rezepturvariablen, so wird dieser Wert in die Rezepturvariable übernommen. Damit diese Funktion genutzt werden kann, muss bei den Rezeptureigenschaften "Rezepturanzeige und Rezepturvariablen abgleichen" aktiviert sein.
	<Strg+Ab>	Die in der Rezepturanzeige angezeigten Werte des eingestellten Rezepturdatensatzes werden in die Steuerung übertragen.
	<Strg+Auf>	Die Rezepturwerte aus der Steuerung werden in der Rezepturanzeige angezeigt.

Werte aktualisieren und anzeigen

Wenn durch einen Steuerungsauftrag Werte des betreffenden Rezepturdatensatzes geändert werden, dann wird die Rezepturanzeige nicht automatisch aktualisiert. Die Werte werden im Hintergrund geändert. Um die Rezepturanzeige zu aktualisieren, müssen Sie den betreffenden Rezepturdatensatz neu anwählen.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Erweiterte Rezepturanzeige bedienen (Seite 1159)

Rezepturdatensätze verwalten (Seite 1161)

Rezepturdatensatz abgleichen (Seite 1162)

Rezepturdatensatz aus Steuerung lesen (Seite 1163)

Rezepturdatensatz zur Steuerung übertragen (Seite 1164)

Konfigurationsmöglichkeiten der erweiterten Rezepturanzeige (Seite 1156)

Konfigurationsmöglichkeiten der erweiterten Rezepturanzeige

Sie können das Verhalten der erweiterten Rezepturanzeige im Inspektorfenster der Rezepturanzeige festlegen.

Die Darstellung der Rezepturanzeigen ist abhängig von dem Gerät und der Geräteversion.

Bei Geräten mit erweiterter Rezepturanzeige beachten Sie Folgendes:

1. Wählen Sie im Inspektorfenster unter "Eigenschaften > Darstellung > Modus" als "Anzeigentyp" " Erweitere Ansicht ".
2. Der Bereich "Eigenschaften > Einfache Ansicht" enthält Eigenschaften, die nur für die einfache Rezepturanzeige gelten.
3. Alle anderen Eigenschaften gelten für die erweiterte Rezepturanzeige.

Eine Rezeptur anzeigen

Um in einem Bild nur den Zugriff auf die Rezepturdatensätze einer bestimmten Rezeptur zuzulassen, gehen Sie folgendermaßen vor:

1. Geben Sie unter "Eigenschaften > Allgemein" in das Feld "Rezeptur" die gewünschte Rezeptur ein oder wählen Sie eine bestehende Rezeptur aus.
2. Wenn im Feld "Rezeptur" eine Rezeptur eingetragen ist, aktivieren Sie "Auswahlliste anzeigen", wenn der Rezepturname in Runtime angezeigt werden soll.

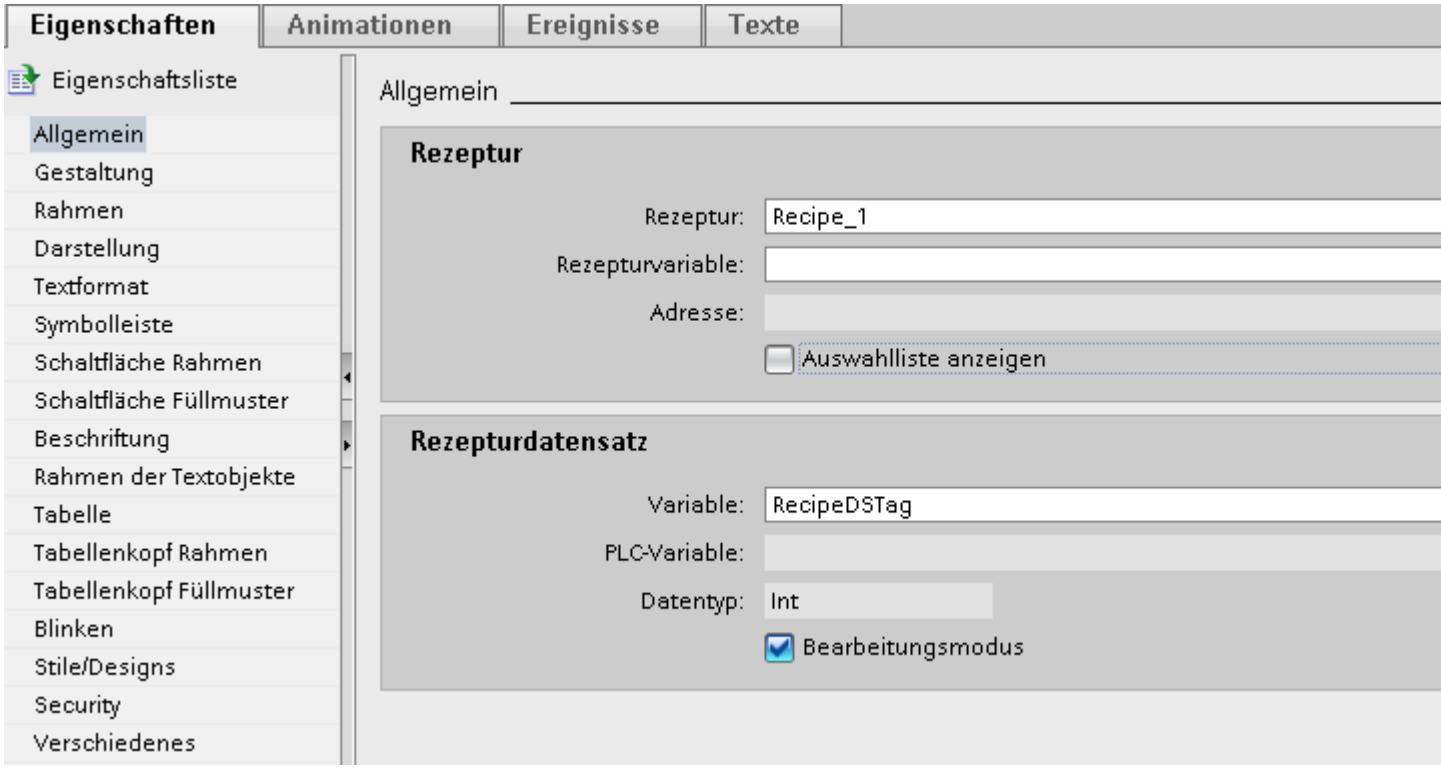

Die gewünschte Rezeptur wird in der Rezepturanzeige angegeben.

Rezeptur- und Rezepturdatensatznummer/-name in Variable schreiben

Sie können in der Rezepturanzeige sowohl an die Rezeptur als auch den Rezepturdatensatz jeweils eine Variable projektieren. Abhängig vom Datentyp "String" oder "Int" der Variablen, wird der Name oder die Nummer der Rezeptur oder des Rezepturdatensatz in der Variablen gespeichert. Umgekehrt können Sie die Variable auch zur Auswahl der Rezeptur oder des Rezepturdatensatzes verwenden, indem Sie den entsprechenden Wert vorgeben. Die Variablen können Sie z. B. als Parameter für eine Systemfunktion übergeben.

Gehen Sie folgendermaßen vor:

1. Geben Sie unter "Eigenschaften > Allgemein > Rezeptur" in das Feld "Rezepturvariable" eine Variable vom Typ "String" ein.
2. Geben Sie unter "Eigenschaften > Allgemein > Rezepturdatensatz" in das Feld "Variable" eine Variable vom Typ "Int" ein.

Der Name der Rezeptur und die Nummer des Rezepturdatensatzes werden jeweils in einer Variablen gespeichert.

Rezepturanzeige als Auswahlfeld verwenden

Um in einem Rezepturbild die Rezepturanzeige als Auswahlfeld für Rezepturen und Rezepturdatensätze zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie unter "Allgemein > Rezeptur > Rezepturvariable" die Variable für den Rezepturnamen.
2. Wählen Sie unter "Allgemein > Rezepturdatensatz > Variable" die Variable für den Rezepturdatensatznamen.
3. Deaktivieren Sie "Bearbeitungsmodus". Sie können keine Rezepturdaten neu anlegen, umbenennen, editieren oder löschen.
4. Um Rezepturen auszuwählen, aktivieren Sie "Auswahlliste anzeigen" und achten Sie darauf, dass unter "Eigenschaften > Allgemein > Rezeptur" keine Rezeptur ausgewählt ist.
5. Deaktivieren Sie unter "Eigenschaften > Symbolleiste" alle Schaltflächen.

Ereignis an die Rezepturanzeige projektieren

Um ein Ereignis an die Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Bilder" die ins Bild eingefügte Rezepturanzeige.
   Die Eigenschaften der Rezepturanzeige werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" auf das Ereignis, das Sie projektieren möchten.
3. Projektieren Sie eine Funktionsliste an das ausgewählte Ereignis.
   Wenn das projektierte Ereignis eintritt, wird die Funktionsliste abgearbeitet.

Siehe auch

Grundlagen zur Rezepturanzeige (Seite 1141)

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Konfiguration von Rezepturen (Seite 1130)

Erweiterte Rezepturanzeige bedienen

Einleitung

Die Rezepturanzeige können Sie sowohl mit Maus oder Touchpad als auch mit der Tastatur bedienen.

Die Darstellung der Rezepturanzeigen ist abhängig von dem Gerät und der Geräteversion.

Rezepturanzeige mit Maus oder Touchpad bedienen

Um die Rezepturanzeige mit Maus oder Touchpad zu bedienen, gehen Sie folgendermaßen vor:

1. Wählen Sie die gewünschte Rezeptur.
   Die Datensätze der Rezeptur werden angezeigt.
2. Wählen Sie den Datensatz, den Sie bearbeiten wollen.
3. Klicken Sie auf die Schaltfläche, deren Funktion Sie ausführen möchten.

Rezepturanzeige mit Tastatur bedienen

Um die Rezepturanzeige mit der Tastatur zu bedienen, gehen Sie folgendermaßen vor:

1. Drücken Sie so oft die Taste <Tab>, bis sich der Cursor im Feld zur Auswahl der Rezeptur befindet.
2. Drücken Sie die Taste <Enter>.
   Die Auswahlliste für die Rezepturen wird geöffnet.
3. Wählen Sie eine Rezeptur. Mit den Cursortasten <Links>, <Rechts>, <Auf> und <Ab> wechseln Sie in der Liste zum nächsten oder vorherigen Eintrag.
4. Wählen Sie einen Datensatz.
5. Drücken Sie so oft die Taste <Tab>, bis das Bedienobjekt selektiert ist, dessen Funktion Sie ausführen möchten.

Alternativ können Sie die Rezepturanzeige über die entsprechenden Tastenkombinationen bedienen.

Rezepturanzeige mit Funktionstasten bedienen

Sie können die Rezepturanzeige über Funktionstasten bedienen, z. B., wenn das Bediengerät keine Touch-Funktionalität besitzt. An die Funktionstasten des Bediengeräts können Funktionen, z. B. "SpeichereDatensatz", projektiert sein.

Tastaturkombinationen für die erweiterte Rezepturanzeige

Folgenden Tastaturkombinationen sind in Runtime für die erweiterte Rezepturanzeige aktiviert, wenn im ES "Tastaturbedienung aktivieren" aktiviert ist.

Tasten kombination	Auswirkung	Menübefehl	Schaltfläche
<HELP>	Ruft den projektierten Tooltip zur ausgewählten Rezeptur auf.	Tooltip	
<Strg +Space>	Erzeugt einen neuen Rezepturdatensatz Wenn ein Startwert konfiguriert ist, wird er im Eingabefeld angezeigt.	Datensatz hinzufügen	
<Strg+Return>	Speichert den geänderten Datensatz unter dem aktuellen Namen.	Speichern	
<Strg+*>	Speichert den geänderten Datensatz unter einem neuen Namen.	Speichern unter	
<Strg+Entf>	Der angezeigt Rezepturdatensatz wird gelöscht.	Datensatz löschen	
<Strg+=>	Vergleicht die Werte des markierten Datensatzes mit den Werten in der Steuerung. Wenn der Wert in der Rezepturanzeige aktueller ist als der aktuelle Wert der Rezepturvariablen, wird dieser Wert in die Rezepturvariable geschrieben.	Rezepturvariablen abgleichen	
<Strg+Ab>	Überträgt den aktuellen Wert in die Steuerung.	In PLC schreiben	
<Strg+Auf>	Liest den aktuellen Wert aus der Steuerung.	von PLC lesen	

Bildwechsel

Wenn Sie zu einem anderen Bild wechseln und in der Rezepturanzeige Änderungen an Rezepturdaten noch nicht gespeichert haben, werden Sie aufgefordert, die Rezepturdaten zu sichern. Damit Sie feststellen können, welche Rezepturdaten noch nicht gesichert waren, werden Ihnen der Rezepturname und der Name des Rezepturdatensatzes angezeigt.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Rezepturdatensätze verwalten

Rezepturdatensätze verwalten

Zur Verwaltung der Rezepturdatensätze haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Neue Rezepturdatensätze erstellen
- Rezepturdatensätze kopieren
- Rezepturdatensätze ändern
- Rezepturdatensätze löschen

Neuen Rezepturdatensatz erstellen

Um einen neuen Rezepturdatensatz zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen neuen Rezepturdatensatz erstellen wollen.
2. Klicken Sie auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg +Leertaste>.
   Ein neuer Rezepturdatensatz mit der nächsten freien Nummer wird angelegt.
   Wenn Sie die neue Datensatznummer in eine vorhandene Datensatznummer ändern, wird der bestehende Datensatz überschrieben.
3. Geben Sie Werte für die Elemente des Datensatzes ein.
   Abhängig von der Projektierung können die Elemente des Datensatzes mit Standardwerten belegt sein.
4. Klicken Sie auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+*>.
   Der Dialog "Speichern unter" wird geöffnet.
5. Geben Sie einen Namen für den Datensatz ein.
6. Bestätigen Sie Ihre Eingaben mit "OK".
   Der Datensatz wird unter dem neuen Namen gespeichert.
   Wenn der Datensatz bereits vorhanden ist, wird ein Dialog geöffnet. Legen Sie in diesem Dialog fest, ob der vorhandene Datensatz überschrieben wird.

Rezepturdatensatz kopieren

Um einen Rezepturdatensatz zu kopieren, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz kopieren wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie kopieren.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+*>.
   Der Dialog "Speichern unter" wird geöffnet.
4. Geben Sie einen Namen für den Datensatz ein.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Rezepturdatensatz ändern

Um einen Rezepturdatensatz zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz ändern wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie ändern wollen.
3. Ersetzen Sie die vorhandenen Werte durch die neuen Werte.
4. Klicken Sie in der Rezepturanzeige die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Enter>.

Rezepturdatensatz löschen

Um einen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz löschen wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie löschen wollen.
3. Klicken Sie in der Rezepturanzeige die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Entf>.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Rezepturdatensatz abgleichen

Einleitung

In Runtime können Unterschiede zwischen folgenden Werten auftreten:

- Den angezeigten Werten in der Rezepturanzeige
- Den tatsächlichen Werten der Rezepturvariablen

Abhängig von der Projektierung können Sie die in der Rezepturanzeige angezeigten Werte mit den Rezepturvariablen abgleichen. Der Abgleich umfasst immer alle Variablen, die zu einem Rezepturdatensatz gehören.

Hinweis

Variablenname geändert

Wenn der Variablenname der abzugleichenden Variable geändert wurde, lassen sich Variable und Wert des Rezepturelements nicht einander zuordnen. Die betreffenden Variablen werden nicht abgeglichen.

Hinweis

Sie können die Rezepturvariablen nur mit der erweiterten Rezepturanzeige abgleichen.

Voraussetzung

- Ein Rezepturdatensatz wird in der Rezepturanzeige angezeigt.
- Die Werte von Rezepturvariablen wurden z. B. durch Teachen geändert.

Vorgehen

Um einen Rezepturdatensatz zu synchronisieren, gehen Sie folgendermaßen vor:

- Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+=>.

Wenn der Wert einer Rezepturvariablen einen aktuelleren Wert als die Rezepturanzeige besitzt, so wird dieser Wert in die Rezepturanzeige übernommen.

Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Rezepturvariablen, so wird dieser Wert in die Rezepturvariable übernommen.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Synchronisation von Rezepturdatensätzen mit der Steuerung (Seite 1134)

Rezepturdatensatz aus Steuerung lesen

Einleitung

In Runtime können Sie direkt in der Anlage die Werte ändern, die auch in den Rezepturen im Bediengerät hinterlegt sind. Dies ist z. B. der Fall, wenn ein Ventil direkt an der Anlage weiter geöffnet wurde, als in der Rezeptur hinterlegt ist. Die Werte der im Bediengerät gespeicherten Rezepturdatensätze stimmen dann möglicherweise nicht mehr mit den Werten in der Steuerung überein.

Sie können die Werte der Rezepturvariablen aus der Steuerung lesen und in einen Rezepturdatensatz schreiben.

Die eingelesenen Werte werden in den Rezepturdatensatz geschrieben, der gerade am Bediengerät angezeigt wird.

Vorgehen

Um einen Rezepturdatensatz aus der Steuerung zu lesen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, dessen Werte Sie aus der Steuerung lesen wollen.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Auf>.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Rezepturdatensatz zur Steuerung übertragen

Einleitung

Damit die Werte eines mit der Rezepturanzeige geänderten Datensatzes in der Steuerung gültig sind, müssen Sie die Werte zur Steuerung übertragen.

Es werden immer die in der Rezepturanzeige angezeigten Werte zur Steuerung übertragen.

Vorgehen

Um einen Rezepturdatensatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, dessen Werte Sie in die Steuerung übertragen wollen.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Ab>.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Synchronisation von Rezepturdatensätzen mit der Steuerung (Seite 1134)

Übertragen von Rezepturdatensätzen (Seite 1127)

6.1.2.4 Erweiterte Rezepturanzeige (ab V13)

Beschreibung Erweiterte Rezepturanzeige (ab V13)

Einleitung

Die erweiterte Rezepturanzeige ist ein Anzeige- und Bedienobjekt, das zum Verwalten der Rezepturdatensätze verwendet wird. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform.

Die angezeigten Schaltflächen, Überschriften und Informationen in den Spalten sind einstellbar.

Die in der Rezepturanzeige angezeigten oder eingegebenen Werte werden in Rezepturdatensätzen gespeichert.

Die Darstellung der Rezepturanzeigen ist abhängig von dem Gerät und der Geräteversion.

Verwendung

Mit der Rezepturanzeige können Sie Datensätze anzeigen, bearbeiten und verwalten.

Aufbau der Anzeige

Das folgende Bild zeigt ein Beispiel für die erweiterte Rezepturanzeige auf einem Bediengerät ab der Geräteversion V13.

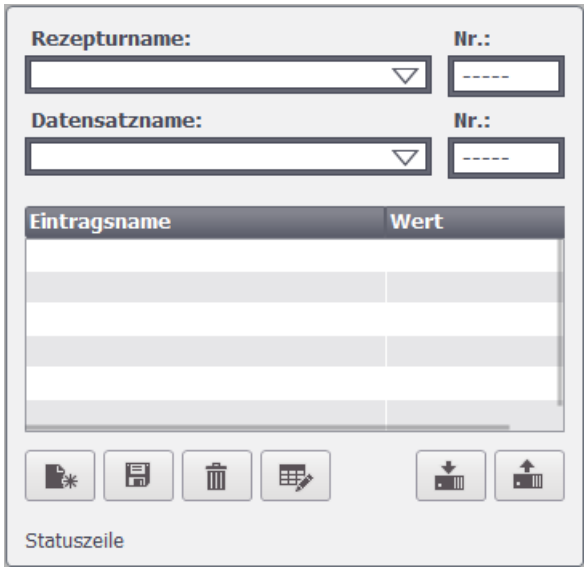

Bedienung

Abhängig von der Projektierung haben Sie folgende Möglichkeiten:

- Rezepturdatensätze anlegen, ändern, kopieren oder löschen
- Rezepturdatensätze aus der Steuerung lesen oder zur Steuerung übertragen
- Rezepturdatensätze mit den zugehörigen Rezepturvariablen abgleichen

Bedienelemente

In der Rezepturanzeige können folgende Bedienobjekte projektiert sein:

Bedienelemente	Tastenkombination	Funktion
	<HELP>	Der projektierte Tooltip wird angezeigt.
	<Strg+Leertaste>	Ein neuer Rezepturdatensatz wird erstellt. Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt.
	<Strg+Enter>	Die angezeigten Werte des Rezepturdatensatzes werden gespeichert. Der Speicherort ist durch das Projekt vorgegeben.
	<Strg+*>	Der Rezepturdatensatz wird unabhängig von der Rezepturanzeige unter einem anderen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.
	<Strg+Entf>	Der angezeigte Rezepturdatensatz wird gelöscht.
	<Strg+Shift+T>	Der markierte Rezepturdatensatz wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.

Bedienelemente	Tastenkombination	Funktion
	<Strg+=>	Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert ist, gilt Folgendes: • Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, wird dieser Wert in die Rezepturanzeige übernommen. • Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, wird dieser Wert in die Variable übernommen. Der Abgleich findet nur innerhalb des Bediengeräts statt. In diesem Fall wird die oben beispielhaft genannte Prozessvariable im EA-Feld nicht automatisch aus der Steuerung gelesen. [1)] Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert ist, gilt Folgendes: • Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, so wird dieser Wert in die Rezepturanzeige übernommen. • Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, so wird dieser Wert in die Variable übernommen. Der Abgleich findet auch mit der Steuerung statt. [2)]
	<Strg+Ab>	Die in der Rezepturanzeige angezeigten Werte des eingestellten Rezepturdatensatzes werden in die Steuerung übertragen.
	<Strg+Auf>	Die Rezepturwerte aus der Steuerung werden in der Rezepturanzeige angezeigt.

1) Für die Rezeptur sind unter "Allgemein > Synchronisation > Einstellungen" die Optionen "Rezepturvariablen abgleichen" und "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert.

2) Für die Rezeptur ist unter "Allgemein > Synchronisation > Einstellungen" die Option "Rezepturvariablen abgleichen" aktiviert und die Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert.

Werte aktualisieren und anzeigen

Wenn durch einen Steuerungsauftrag Werte des betreffenden Rezepturdatensatzes geändert werden, dann wird die Rezepturanzeige nicht automatisch aktualisiert. Die Werte werden im Hintergrund geändert. Um die Rezepturanzeige zu aktualisieren, müssen Sie den betreffenden Rezepturdatensatz neu anwählen.

Siehe auch

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Erweiterte Rezepturanzeige bedienen (Seite 1171)

Rezepturdatensätze verwalten (Seite 1174)

Rezepturdatensatz abgleichen (Seite 1176)

Rezepturdatensatz aus Steuerung lesen (Seite 1178)

Rezepturdatensatz zur Steuerung übertragen (Seite 1179)

Grundlagen zur Rezepturanzeige (Seite 1141)

Konfigurationsmöglichkeiten der erweiterten Rezepturanzeige (ab V13) (Seite 1168)

Konfigurationsmöglichkeiten der erweiterten Rezepturanzeige (ab V13)

Sie können das Verhalten der erweiterten Rezepturanzeige im Inspektorfenster der Rezepturanzeige festlegen.

Eine Rezeptur anzeigen

Um in einem Bild nur den Zugriff auf die Rezepturdatensätze einer bestimmten Rezeptur zuzulassen, gehen Sie folgendermaßen vor:

1. Geben Sie unter "Eigenschaften > Allgemein" in das Feld "Rezeptur" die gewünschte Rezeptur ein oder wählen Sie eine bestehende Rezeptur aus.
2. Wenn im Feld "Rezeptur" eine Rezeptur eingetragen ist, aktivieren Sie "Auswahlliste anzeigen", wenn der Rezepturname in Runtime angezeigt werden soll.

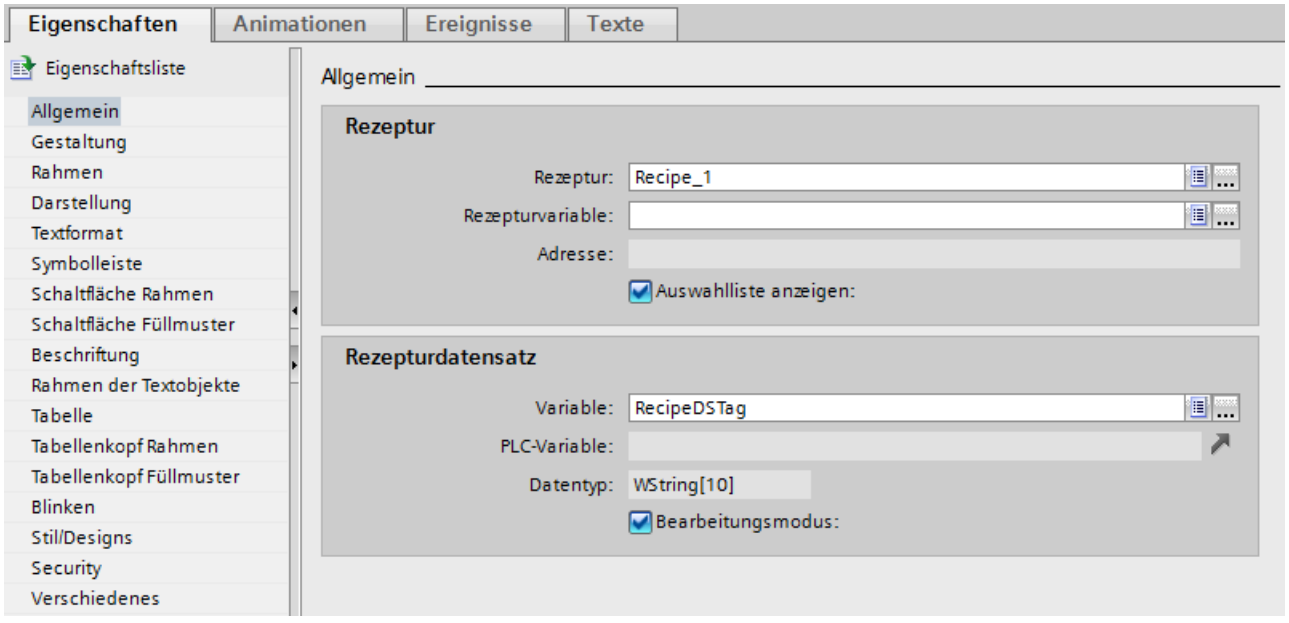

Die gewünschte Rezeptur wird in der Rezepturanzeige angegeben.

Rezeptur- und Rezepturdatensatznummer/-name in Variable schreiben

Sie können in der Rezepturanzeige sowohl an die Rezeptur als auch den Rezepturdatensatz jeweils eine Variable projektieren. Abhängig vom Datentyp "String" oder "Int" der Variablen, wird der Name oder die Nummer der Rezeptur oder des Rezepturdatensatz in der Variablen gespeichert. Umgekehrt können Sie die Variable auch zur Auswahl der Rezeptur oder des Rezepturdatensatzes verwenden, indem Sie den entsprechenden Wert vorgeben. Die Variablen können Sie z. B. als Parameter für eine Systemfunktion übergeben.

Gehen Sie folgendermaßen vor:

1. Geben Sie unter "Eigenschaften > Allgemein > Rezeptur" in das Feld "Rezepturvariable" eine Variable vom Typ "String" ein.
2. Geben Sie unter "Eigenschaften > Allgemein > Rezepturdatensatz" in das Feld "Variable" eine Variable vom Typ "Int" ein.

Der Name der Rezeptur und die Nummer des Rezepturdatensatzes werden jeweils in einer Variablen gespeichert.

Rezepturanzeige als Auswahlfeld verwenden

Um in einem Rezepturbild die Rezepturanzeige als Auswahlfeld für Rezepturen und Rezepturdatensätze zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie unter "Allgemein > Rezeptur > Rezepturvariable" die Variable für den Rezepturnamen.
2. Wählen Sie unter "Allgemein > Rezepturdatensatz > Variable" die Variable für den Rezepturdatensatznamen.
3. Deaktivieren Sie "Bearbeitungsmodus". Sie können keine Rezepturdaten neu anlegen, umbenennen, editieren oder löschen.

4. Um Rezepturen auszuwählen, aktivieren Sie "Auswahlliste anzeigen" und achten Sie darauf, dass unter "Eigenschaften > Allgemein > Rezeptur" keine Rezeptur ausgewählt ist.
5. Deaktivieren Sie unter "Eigenschaften > Symbolleiste" alle Schaltflächen.

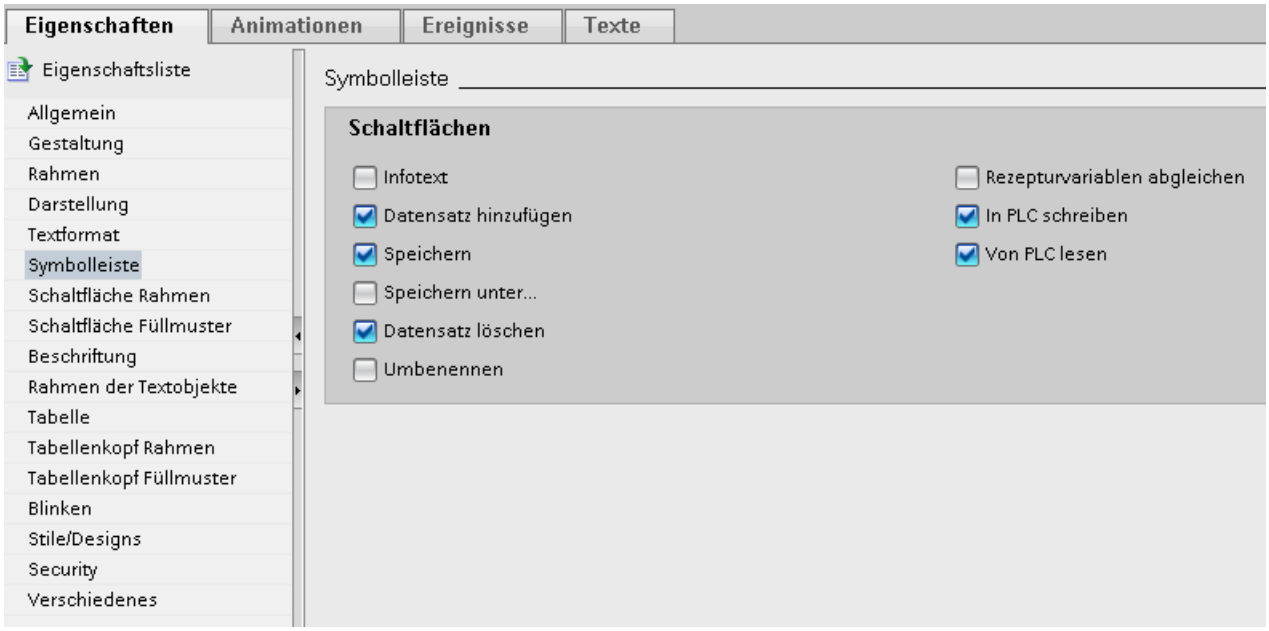

Ereignis an die Rezepturanzeige projektieren

Um ein Ereignis an die Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Bilder" die ins Bild eingefügte Rezepturanzeige.
   Die Eigenschaften der Rezepturanzeige werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" auf das Ereignis, das Sie projektieren möchten.
3. Projektieren Sie eine Funktionsliste an das ausgewählte Ereignis.
   Wenn das projektierte Ereignis eintritt, wird die Funktionsliste abgearbeitet.

Siehe auch

Beispiel für das Erstellen einer Rezeptur (Seite 1201)

Grundlagen zur Rezepturanzeige (Seite 1141)

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Konfiguration von Rezepturen (Seite 1130)

Erweiterte Rezepturanzeige bedienen

Einleitung

Die Rezepturanzeige können Sie sowohl mit Maus oder Touchpad als auch mit der Tastatur bedienen.

Rezepturanzeige mit Maus oder Touchpad bedienen

Um die Rezepturanzeige mit Maus oder Touchpad zu bedienen, gehen Sie folgendermaßen vor:

1. Wählen Sie die gewünschte Rezeptur.
   Die Datensätze der Rezeptur werden angezeigt.
2. Wählen Sie den Datensatz, den Sie bearbeiten wollen.
3. Klicken Sie auf die Schaltfläche, deren Funktion Sie ausführen möchten.

Rezepturanzeige mit Tastatur bedienen

Um die Rezepturanzeige mit der Tastatur zu bedienen, gehen Sie folgendermaßen vor:

1. Drücken Sie so oft die Taste <Tab>, bis sich der Cursor im Feld zur Auswahl der Rezeptur befindet.
2. Drücken Sie die Taste <Enter>.
   Die Auswahlliste für die Rezepturen wird geöffnet.
3. Wählen Sie eine Rezeptur. Mit den Cursortasten <Links>, <Rechts>, <Auf> und <Ab> wechseln Sie in der Liste zum nächsten oder vorherigen Eintrag.
4. Wählen Sie einen Datensatz.
5. Drücken Sie so oft die Taste <Tab>, bis das Bedienobjekt selektiert ist, dessen Funktion Sie ausführen möchten.

Alternativ können Sie die Rezepturanzeige über die entsprechenden Tastenkombinationen bedienen.

Rezepturanzeige mit Funktionstasten bedienen

Sie können die Rezepturanzeige über Funktionstasten bedienen, z. B., wenn das Bediengerät keine Touch-Funktionalität besitzt. An die Funktionstasten des Bediengeräts können Funktionen, z. B. "SpeichereDatensatz", projektiert sein.

Tastaturkombinationen für die erweiterte Rezepturanzeige

Folgenden Tastaturkombinationen sind in Runtime für die erweiterte Rezepturanzeige aktiviert, wenn im ES "Tastaturbedienung aktivieren" aktiviert ist.

Tasten kombination	Auswirkung	Menübefehl	Schaltfläche
<HELP>	Ruft den projektierten Tooltip zur ausgewählten Rezeptur auf.	Tooltip	
<Strg +Space>	Erzeugt einen neuen Rezepturdatensatz Wenn ein Startwert konfiguriert ist, wird er im Eingabefeld angezeigt.	Datensatz hinzufügen	
<Strg+Return>	Speichert den geänderten Datensatz unter dem aktuellen Namen.	Speichern	
<Strg+*>	Speichert den geänderten Datensatz unter einem neuen Namen.	Speichern unter	
<Strg+Entf>	Der angezeigte Rezepturdatensatz wird gelöscht.	Datensatz löschen	
<Strg+Shift +T>	Ändert den Namen des markierten Datensatzes.	Datensatz umbenennen	

Tasten kombination	Auswirkung	Menübefehl	Schaltfläche
<Strg+=>	Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert ist, gilt Folgendes: • Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, wird dieser Wert in die Rezepturanzeige übernommen. • Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, wird dieser Wert in die Variable übernommen. Der Abgleich findet nur innerhalb des Bediengeräts statt. In diesem Fall wird die oben beispielhaft genannte Prozessvariable im EA-Feld nicht automatisch aus der Steuerung gelesen. [1)] Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert ist, gilt Folgendes: • Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, so wird dieser Wert in die Rezepturanzeige übernommen. • Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, so wird dieser Wert in die Variable übernommen. Der Abgleich findet auch mit der Steuerung statt. [2)]	Rezepturanzeige und Rezepturvariablen abgleichen	

Tasten kombination	Auswirkung	Menübefehl	Schaltfläche
<Strg+Ab>	Überträgt den aktuellen Wert in die Steuerung.	Schreiben in Steuerung	
<Strg+Auf>	Liest den aktuellen Wert aus der Steuerung.	Lesen aus Steuerung	

1) Für die Rezeptur sind unter "Allgemein > Synchronisation > Einstellungen" die Optionen "Rezepturvariablen abgleichen" und "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert.

2) Für die Rezeptur ist unter "Allgemein > Synchronisation > Einstellungen" die Option "Rezepturvariablen abgleichen" aktiviert und die Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert.

Bildwechsel

Wenn Sie zu einem anderen Bild wechseln und in der Rezepturanzeige Änderungen an Rezepturdaten noch nicht gespeichert haben, werden Sie aufgefordert, die Rezepturdaten zu sichern. Damit Sie feststellen können, welche Rezepturdaten noch nicht gesichert waren, werden Ihnen der Rezepturname und der Name des Rezepturdatensatzes angezeigt.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Rezepturdatensätze verwalten

Rezepturdatensätze verwalten

Zur Verwaltung der Rezepturdatensätze haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Neue Rezepturdatensätze erstellen
- Rezepturdatensätze kopieren
- Rezepturdatensätze ändern
- Rezepturdatensätze löschen
- Rezepturdatensätze umbenennen

Neuen Rezepturdatensatz erstellen

Um einen neuen Rezepturdatensatz zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen neuen Rezepturdatensatz erstellen wollen.
2. Klicken Sie auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg +Leertaste>.
   Ein neuer Rezepturdatensatz mit der nächsten freien Nummer wird angelegt.
   Wenn Sie die neue Datensatznummer in eine vorhandene Datensatznummer ändern, wird der bestehende Datensatz überschrieben.
3. Geben Sie Werte für die Elemente des Datensatzes ein.
   Abhängig von der Projektierung können die Elemente des Datensatzes mit Standardwerten belegt sein.
4. Geben Sie einen Namen für den Datensatz ein.
5. Klicken Sie auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg +Enter>.
   Alternativ klicken Sie zum Speichern des Datensatzes auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+*>.
   Der Dialog "Speichern unter" wird geöffnet. Geben Sie den Namen für den Datensatz an und bestätigen Sie Ihre Eingaben mit "OK".
6. Der Datensatz wird unter dem neuen Namen gespeichert.
   Wenn der Datensatz bereits vorhanden ist, wird ein Dialog geöffnet. Legen Sie in diesem Dialog fest, ob der vorhandene Datensatz überschrieben wird.

Rezepturdatensatz kopieren

Um einen Rezepturdatensatz zu kopieren, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz kopieren wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie kopieren.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+*>.
   Der Dialog "Speichern unter" wird geöffnet.
4. Geben Sie einen Namen für den Datensatz ein.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Rezepturdatensatz ändern

Um einen Rezepturdatensatz zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz ändern wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie ändern wollen.

3. Ersetzen Sie die vorhandenen Werte durch die neuen Werte.
4. Klicken Sie in der Rezepturanzeige die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Enter>.

Rezepturdatensatz löschen

Um einen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz löschen wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie löschen wollen.
3. Klicken Sie in der Rezepturanzeige die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Entf>.

Rezepturdatensatz umbenennen

Um einen Rezepturdatensatz umzubenennen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz umbenennen wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie umbenennen wollen.
3. Klicken Sie in der Rezepturanzeige die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Shift+T>.
4. Geben Sie im Dialogfenster einen neuen Namen ein.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Rezepturdatensatz abgleichen

Einleitung

In Runtime können Unterschiede zwischen folgenden Werten auftreten:

- Den angezeigten Werten in der Rezepturanzeige
- Den tatsächlichen Werten der Rezepturvariablen

Abhängig von der Projektierung können Sie die in der Rezepturanzeige angezeigten Werte mit den Rezepturvariablen abgleichen. Der Abgleich umfasst immer alle Variablen, die zu einem Rezepturdatensatz gehören.

Hinweis

Variablenname geändert

Wenn der Variablenname der abzugleichenden Variable geändert wurde, lassen sich Variable und Wert des Rezepturelements nicht einander zuordnen. Die betreffenden Variablen werden nicht abgeglichen.

Hinweis

Sie können die Rezepturvariablen nur mit der erweiterten Rezepturanzeige abgleichen.

Voraussetzung

- Ein Rezepturdatensatz wird in der Rezepturanzeige angezeigt.
- Die Werte von Rezepturvariablen wurden z. B. durch Teachen geändert.

Vorgehen

Um einen Rezepturdatensatz zu synchronisieren, gehen Sie folgendermaßen vor:

- Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+=>.

Ergebnis

Das Ergebnis hängt von Ihren Einstellungen ab:

- Für die Rezeptur sind unter "Allgemein > Synchronisation > Einstellungen" die Optionen "Rezepturvariablen abgleichen" und "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert.
  Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert ist, gilt Folgendes:
  - Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, wird dieser Wert in die Rezepturanzeige übernommen.
  - Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, wird dieser Wert in die Variable übernommen. Der Abgleich findet nur innerhalb des Bediengeräts statt.

  In diesem Fall wird die oben beispielhaft genannte Prozessvariable im EA-Feld nicht automatisch aus der Steuerung gelesen.
- Für die Rezeptur ist unter "Allgemein > Synchronisation > Einstellungen" die Option "Rezepturvariablen abgleichen" aktiviert und die Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert.
  Wenn Variablen am Bediengerät in einer Rezeptur und an einer weiteren Stelle in der Projektierung verwendet werden (z. B. als Prozessvariablen in einem EA-Feld) und die rezeptureigene Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" deaktiviert ist, gilt Folgendes:
  - Wenn der Wert einer Variablen einen aktuelleren Wert als in der Rezepturanzeige besitzt, so wird dieser Wert in die Rezepturanzeige übernommen.
  - Wenn der angezeigte Wert in der Rezepturanzeige aktueller ist als der Wert der Variablen, so wird dieser Wert in die Variable übernommen.

  Der Abgleich findet auch mit der Steuerung statt.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Synchronisation von Rezepturdatensätzen mit der Steuerung (Seite 1134)

Rezepturdatensatz aus Steuerung lesen

Einleitung

In Runtime können Sie direkt in der Anlage die Werte ändern, die auch in den Rezepturen im Bediengerät hinterlegt sind. Dies ist z. B. der Fall, wenn ein Ventil direkt an der Anlage weiter geöffnet wurde, als in der Rezeptur hinterlegt ist. Die Werte der im Bediengerät gespeicherten Rezepturdatensätze stimmen dann möglicherweise nicht mehr mit den Werten in der Steuerung überein.

Sie können die Werte der Rezepturvariablen aus der Steuerung lesen und in einen Rezepturdatensatz schreiben.

Die eingelesenen Werte werden in den Rezepturdatensatz geschrieben, der gerade am Bediengerät angezeigt wird.

Vorgehen

Um einen Rezepturdatensatz aus der Steuerung zu lesen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, dessen Werte Sie aus der Steuerung lesen wollen.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Auf>.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Rezepturdatensatz zur Steuerung übertragen

Einleitung

Damit die Werte eines mit der Rezepturanzeige geänderten Datensatzes in der Steuerung gültig sind, müssen Sie die Werte zur Steuerung übertragen.

Es werden immer die in der Rezepturanzeige angezeigten Werte zur Steuerung übertragen.

Vorgehen

Um einen Rezepturdatensatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, dessen Werte Sie in die Steuerung übertragen wollen.
3. Klicken Sie in der Rezepturanzeige auf die Schaltfläche oder drücken Sie die Tastenkombination <Strg+Ab>

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Synchronisation von Rezepturdatensätzen mit der Steuerung (Seite 1134)

Übertragen von Rezepturdatensätzen (Seite 1127)

6.1.2.5 Rezepturdaten exportieren und importieren

Rezepturdatensatz exportieren und importieren

Einleitung

Sie können Rezepturdatensätze in eine CSV oder eine TXT-Datei exportieren, um sie z. B. in MS Excel zu bearbeiten, oder aus einer Datei importieren.

Das Exportieren und Importieren von Rezepturen ist abhängig von der Projektierung des Geräts und der Geräteversion.

Listentrennzeichen

Für die Trennung der Datensätze wird beim Import und Export ein Listentrennzeichen verwendet. Das als Standard verwendete Listentrennzeichen ist abhängig von den Einstellungen für Formate und Zahlen im Betriebssystem.

Die Einstellungen finden Sie unter "Start > Einstellungen > Systemsteuerung > Regions- und Sprachoptionen".

Wenn Sie Rezepturdatensätze importieren oder exportieren wollen, verwenden Sie dieses Listentrennzeichen nicht im Anzeigenamen der Rezepturdatensätze.

Für die Bedienung der Export-Importfunktion können z. B. folgende Bedienobjekte projektiert sein:

- Auswahlfeld für die Rezeptur
- Auswahlfeld für den Rezepturdatensatz
- Bedienobjekt mit der Funktionalität "ExportiereDatensaetze"
- Bedienobjekt mit der Funktionalität "ImportiereDatensaetze"

Rezepturdatensatz exportieren

Um einen Rezepturdatensatz in eine CSV-Datei zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in den Auswahlfeldern die gewünschte Rezeptur und den gewünschten Rezepturdatensatz.
2. Klicken Sie auf das Bedienobjekt mit der Funktionalität "ExportiereDatensaetze".

Rezepturdatensatz importieren

Um einen Rezepturdatensatz zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in den Auswahlfeldern die gewünschte Rezeptur und den gewünschten Rezepturdatensatz.
2. Klicken Sie auf das Bedienobjekt mit der Funktionalität "ImportiereDatensaetze".

Anzeige nach Import von Rezepturen

Wenn Sie während eines Imports von Rezepturdaten die Rezepturanzeige öffnen, werden nur die Rezepturdaten angezeigt, die bereits vollständig importiert wurden. Die Rezepturanzeige wird bei einem Datenimport nicht automatisch aktualisiert. Um eine vollständige Anzeige aller Rezepturdaten zu erhalten, öffnen Sie die Rezepturanzeige erst, nachdem die Systemmeldung über den erfolgreichen Import der Rezepturdaten eingegangen ist.

Alternativ aktualisieren Sie die Rezepturanzeige nach erfolgreichem Abschluss des Importvorgangs.

Einschränkungen beim Import/Export

Auf Basic Panels haben sie die Möglichkeit, mithilfe der Systemfunktionen "ExportiereDatensaetze" und "ImportiereDatensaetze" die Rezepturen in eine TXT-Datei zu exportieren und aus einer TXT-Datei zu importieren.

Hinweis

Exportieren und Importieren mit ProSave in das CSV-Format

Einzelne Rezepturdatensätze können nicht mit ProSave in das CSV-Format exportiert und importiert werden.

Nur komplette Rezepturdaten können mit ProSave in das CSV-Format exportiert und importiert und auf das Bediengerät übertragen werden.
Dabei wird die Runtime angehalten.

Siehe auch

Verhalten der Rezepturanzeige in Runtime (Seite 1182)

Verhalten beim Ändern der Rezepturstruktur

Einleitung

Unterschiedliche Rezepturstrukturen können z. B. in folgenden Fällen auftreten:

- Bei Änderungen während der Inbetriebnahme
- Beim Bearbeiten einer Maschine durch den Maschinenbauer (Retro-Fit)
- Beim Import von CSV-Dateien kann sich die Struktur der CSV-Datei von der Struktur der Rezeptur unterscheiden.

Sie können bereits erstellte Rezepturdatensätze trotzdem weiter verwenden.

ACHTUNG
Beim Ändern des Variablennamens geht die Zuordnung verloren.

Auswirkungen

Behandeln Sie Strukturabweichungen folgendermaßen:

- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei zusätzliche Werte enthält, werden diese Werte verworfen.
- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei Werte vom falschen Datentyp enthält, wird im Rezepturdatensatz der projektierte Standardwert verwendet.
  Beispiel: Der Rezepturdatensatz enthält Werte, die den Tankinhalt angeben und als Fließkommazahlen eingegeben wurden. Die entsprechende Rezepturvariable erwartet jedoch einen ganzzahligen Wert. In diesem Fall wird der übertragene Wert verworfen und der projektierte Standardwert verwendet.
- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei zu wenige Werte enthält, wird im Rezepturdatensatz ebenfalls der projektierte Standardwert verwendet.

Siehe auch

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Grundlagen zur Rezepturanzeige (Seite 1141)

Verhalten der Rezepturanzeige in Runtime

Bildwechsel

Wenn Sie zu einem anderen Bild wechseln und in der Rezepturanzeige Änderungen an Rezepturdaten noch nicht gespeichert haben, werden Sie aufgefordert, die Rezepturdaten zu sichern. Damit Sie feststellen können, welche Rezepturdaten noch nicht gesichert waren, werden Ihnen der Rezepturname und der Name des Rezepturdatensatzes angezeigt.

Rezepturdatensatz neu anlegen, ändern, kopieren oder löschen

Wenn Sie einen Rezepturdatensatz neu anlegen wollen und bereits ein Rezepturdatensatz vorhanden ist, wird am Bildschirm eine Systemmeldung ausgegeben.

Rezepturanzeige mit Funktionstasten bedienen

Sie können die Rezepturanzeige über Funktionstasten bedienen, z. B., wenn das Bediengerät keine Touch-Funktionalität besitzt. An die Funktionstasten des Bediengeräts können Funktionen, z. B. "SpeichereDatensatz", projektiert sein.

Anzeige nach Import von Rezepturdaten

Hinweis

Verfügbarkeit

Der Import und Export von Rezepturdaten steht für Basic Panels nicht zur Verfügung.

Wenn Sie während eines Imports von Rezepturdaten die Rezepturanzeige öffnen, werden nur die Rezepturdaten angezeigt, die bereits vollständig importiert wurden. Die Rezepturanzeige wird bei einem Datenimport nicht automatisch aktualisiert. Um eine vollständige Anzeige aller Rezepturdaten zu erhalten, öffnen Sie die Rezepturanzeige erst, nachdem die Systemmeldung über den erfolgreichen Import der Rezepturdaten eingegangen ist. Alternativ aktualisieren Sie die Rezepturanzeige nach erfolgreichem Abschluss des Importvorgangs.

Variable für Rezepturen und Rezepturdatensätze aktualisieren

Hinweis

Verfügbarkeit

Variablen für Rezepturen und Rezepturdatensätze stehen für Basic Panels nicht zur Verfügung.

Der aktuelle Rezepturdatensatz bzw. dessen Nummer kann abhängig von der Projektierung in einer Variablen gespeichert werden. Die Variable wird unter folgenden Bedingungen aktualisiert:

- Das Laden des Rezepturdatensatzes ist beendet.
- Das Bild mit der Rezepturanzeige wurde während des Ladens nicht verlassen.

Dieser Vorgang kann einige Zeit dauern.

Siehe auch

Rezepturdatensatz exportieren und importieren (Seite 1180)

Beschreibung Erweiterte Rezepturanzeige (vor V13) (Seite 1154)

Beschreibung Erweiterte Rezepturanzeige (ab V13) (Seite 1165)

Grundlagen zur Rezepturanzeige (Seite 1141)

6.1.3 Rezepturen projektieren

6.1.3.1 Rezepturen anlegen und bearbeiten

Allgemeine Vorgehensweise beim Projektieren

Wenn Sie eine Rezeptur neu anlegen, führen Sie folgende Projektierungsschritte aus:

Schritt	Beschreibung
1	Legen Sie die Struktur der Rezeptur fest.
2	Legen Sie entsprechend der Rezepturstruktur Variablen an. Weisen Sie diesen Variablen technologische Namen zu.
3	Legen Sie die Rezeptur an.
4	Geben Sie die gewünschten Eigenschaften der Rezeptur ein: • Sprachabhängiger Anzeigename der Rezeptur • Option "Koordinierte Übertragung der Datensätze" Nicht für Basic Panels: • Ablageort der Rezepturen • Option "Rezepturanzeige und Rezepturvariablen abgleichen" • Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
5	Legen Sie die Rezepturelemente an und geben Sie die gewünschten Eigenschaften ein: • Sprachabhängige Anzeigenamen der Rezepturelemente • Variablen-Anbindung der Rezepturelemente • Standardwerte und Nachkommastelle (Zehnerpotenz) für die Rezepturelemente
6	Legen Sie die Rezepturdatensätze an. Geben Sie die sprachabhängigen Anzeigenamen der Rezepturdatensätze ein.
7	Projektieren Sie ein Bild mit Rezepturanzeige oder ein Rezepturbild.

Hinweis

Basic Panels und OP77A, TP177A (Portrait)

Die Auswahl des Ablageorts steht für diese Geräte nicht zur Verfügung. Die Rezepturen werden immer im internen Flash-Speicher abgelegt.

Rezepturvariablen können Sie nicht außerhalb einer Rezeptur verwenden z. B. nicht in E/A-Feldern, nicht in Meldungen als Triggervariablen, nicht in Systemfunktionen als Parameter usw.

Hinweis

Einschränkungen Rezepturanzeige und Rezepturbild

Bei Basic Panels und OP77A, TP177A ist nur die einfache Rezepturanzeige verfügbar. Bei Basic Panels und OP73, OP77A, TP177A (Portrait) sind Rezepturbilder nicht verfügbar.

Neue Rezeptur anlegen

Einleitung

Zum Anlegen einer vollständigen Rezeptur legen Sie zunächst eine neue Rezeptur an, weisen der Rezeptur Rezepturelemente zu und legen in einem Rezepturdatensatz die entsprechenden Werte fest.

Hinweis

Importfähige Zeichenketten

Rezepturdaten, die ein Semikolon oder ein Komma enthalten, sind nicht importfähig.

Verwenden Sie für Rezepturen und Rezepturdatensätze ausschließlich Zeichenketten ohne Semikolon und ohne Komma.

Voraussetzung

- Die Variablen für die Rezeptur sind angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Rezeptur anlegen

Um eine Rezeptur anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "Rezepturen" in der ersten freien Tabellenzeile auf "Hinzufügen".
   Die neue Rezeptur wird angelegt und in einer Zeile angezeigt.

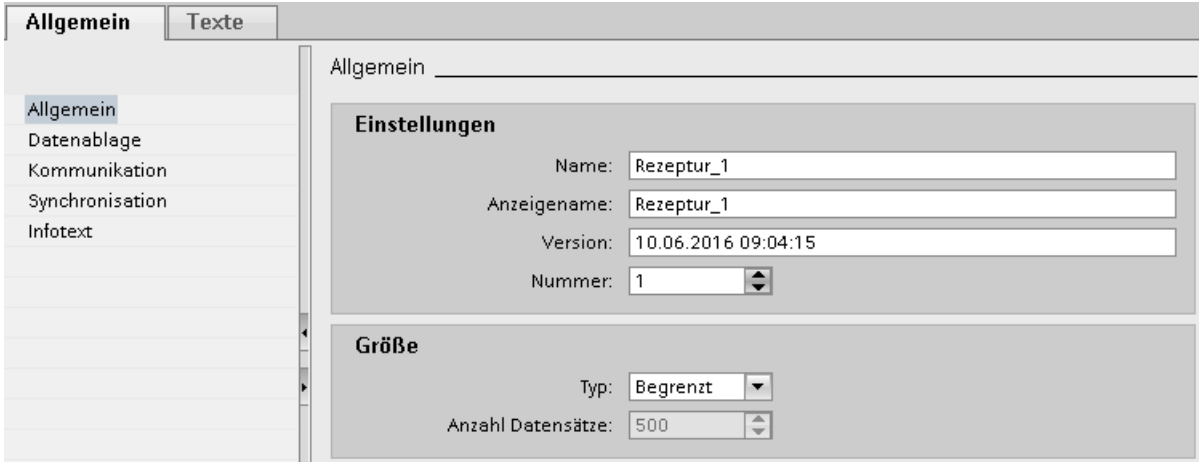

2. Geben Sie im Bereich "Allgemein" unter "Name" einen aussagekräftigen Namen für die Rezeptur ein.
   Der Name identifiziert die Rezeptur innerhalb eines Bediengeräts eindeutig.
3. Geben Sie unter "Anzeigename" den sprachabhängigen Namen ein, der in der Rezepturanzeige erscheint.

4. Wählen Sie unter "Nummer" eine Rezepturnummer aus.
   Die Nummer identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.
   Die Rezeptur erhält automatisch eine Version, die das Datum und den Zeitpunkt der letzten Änderung anzeigt. Alternativ geben Sie hier beliebige Informationen zur Rezeptur ein.
5. Legen Sie unter "Datenablage" den Ablageort für die Rezepturdatensätze fest. Die angebotenen Möglichkeiten sind abhängig vom verwendeten Bediengerät.

   Hinweis
   Basic Panels

   Die Auswahl des Ablageorts steht für diese Geräte nicht zur Verfügung. Die Rezepturen werden immer im internen Flash-Speicher abgelegt.

   Rezepturvariablen können Sie nicht außerhalb einer Rezeptur verwenden z. B. nicht in E/A-Feldern, nicht in Meldungen als Triggervariablen, nicht in Systemfunktionen als Parameter usw.

6. Geben Sie einen Tooltip ein, der dem Bediener in Runtime angezeigt wird.
7. Um in Runtime Rezepturvariablen, die in EA-Feldern projektiert sind, mit der Rezepturanzeige abzugleichen, aktivieren Sie im Inspektorfenster unter "Eigenschaften > Synchronisation" "Rezepturanzeige und Rezepturvariablen abgleichen".

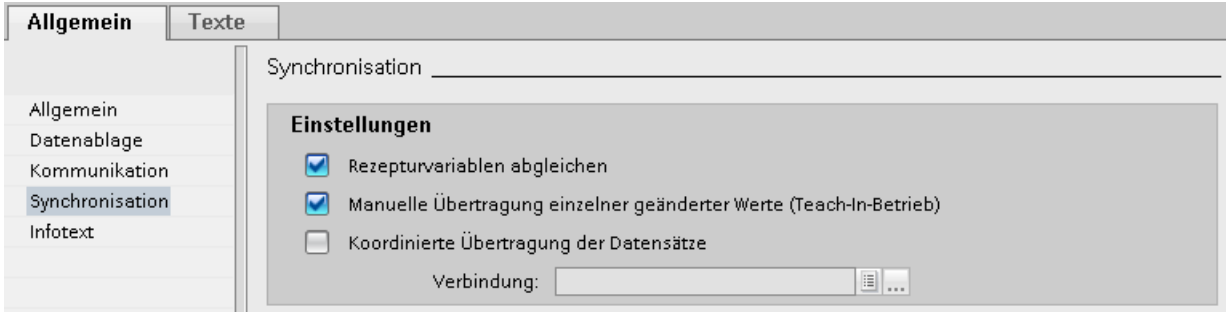

   Hinweis
   Basic Panels

   Da bei Basic Panels die Rezepturvariablen nicht zusätzlich in EA-Feldern in Bildern verwendet werden können, steht die Option "Rezepturanzeige und Rezepturvariablen abgleichen" nicht zur Verfügung, ebenso im Folgenden nicht die Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".

8. Um festzulegen, dass die Rezepturvariablen automatisch beim Editieren in den EA-Feldern in die Steuerung übertragen werden, deaktivieren Sie "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
9. Um in Runtime die Übertragung der Rezepturdaten über Bereichszeiger zu überwachen, aktivieren Sie "Koordinierte Übertragung der Datensätze".
10. Wählen Sie unter "Synchronisieren mit" die entsprechende Verbindung zur Steuerung für die koordinierte Übertragung.

Rezepturelement anlegen

Um Rezepturelemente anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Elemente".
2. Klicken Sie in der ersten freien Zeile des tabellarischen Editors auf "Hinzufügen".
   Ein neues Rezepturelement wird angelegt.
3. Geben Sie unter "Name" einen aussagekräftigen Namen für das Element ein.
   Der Name identifiziert das Element innerhalb der Rezeptur eindeutig.
4. Geben Sie unter "Anzeigename" einen sprachabhängigen Anzeigenamen für das Element ein.
   Der Anzeigename erscheint in Runtime, z. B. in der Rezepturanzeige.
5. Wählen Sie unter "Variable" die Variable aus, die Sie mit dem Rezepturelement verbinden wollen.
   In dieser Variablen wird in Runtime der Wert des Rezepturelements gespeichert, der in einem Rezepturdatensatz abgelegt ist.

Elemente | Datensätze

Name	Anzeigename	Variable	Datentyp	Datenlänge	Standardwert	Minimaler Wert	Maximaler Wert	Nachkommastellen	Infotext
Wasser	Wasser	LiterWater	Int	2	0	-32768	32767	0	
<Hinzufügen>									

6. Geben Sie einen Tooltip ein.
   Der Tooltip wird dem Bediener in Runtime angezeigt.
7. Geben Sie unter "Standardwert" den Wert ein, den Sie als Standardeintrag verwenden wollen, wenn Sie einen neuen Rezepturdatensatz anlegen.
8. Wenn Sie einem Wert oder einem Wertebereich Text zuordnen wollen, wählen Sie hier die entsprechende Textliste aus. Die zugeordnete Rezepturvariable muss den Datentyp einer Zahl besitzen. Der Variablenwert muss sich innerhalb des Wertebereichs der Textliste befinden.
   Der in der Textliste hinterlegte Text wird z. B. in einem Ausgabefeld in Runtime angezeigt.
9. Legen Sie in der Spalte "Nachkommastellen" fest, auf wie viele Stellen genau ein Dezimalwert gerundet wird z. B. 3 Nachkommastellen und umgekehrt mit welcher Zehnerpotenz ein Ganzzahlwert multipliziert wird z. B. 1000.
   Beispiele bei 3 Nachkommastellen: Die Eingabe "5" bei einem Rezepturelement mit dem Datentyp "Integer" liefert den Wert "5000". Die Eingabe "5,6789" bei einem Rezepturelement mit dem Datentyp "Real" liefert den Wert "5,679".
10. Legen Sie so viele Rezepturelemente an, wie Sie für die Rezeptur benötigen. Die maximal mögliche Anzahl von Rezepturelementen hängt vom verwendeten Bediengerät ab.

Elemente | Datensätze

Name	Anzeigename	Variable	Datentyp	Datenlänge	Standardwert
Wasser	Wasser	LiterWater	Int	2	0
Konzentrat	Konzentrat	LiterConcentrat	Int	2	0
Zucker	Zucker	KiloSugar	Int	2	0
Aroma	Aroma	GramAroma	Int	2	0
<Hinzufügen>					

Rezepturdatensatz mit bekannten Rezepturwerten anlegen

Um Rezepturelemente anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Datensätze".
2. Klicken Sie in der ersten freien Zeile des tabellarischen Editors auf "Hinzufügen". Ein neuer Rezepturdatensatz wird angelegt. Für jedes in der Rezeptur angelegte Rezepturelement enthält der Rezepturdatensatz eine eigene Spalte.

Elemente | Datensätze

Name	Anzeigename	Nummer	Wasser	Konzentrat	Zucker	Aroma	Kommentar
Rezepturdatensatz_1	Rezepturdatensatz_1	1	0	0	0	0	
<Hinzufügen>							

3. Geben Sie unter "Namen" einen aussagekräftigen Namen ein. Der Name identifiziert den Datensatz innerhalb der Rezeptur eindeutig.
4. Geben Sie unter "Anzeigenamen" einen sprachabhängigen Anzeigenamen ein. Der Anzeigename erscheint in Runtime, z. B. in der Rezepturanzeige.
5. Geben Sie unter "Nummer" eine Rezepturdatensatznummer ein. Die Rezepturdatensatznummer identifiziert den Rezepturdatensatz innerhalb der Rezeptur eindeutig.
6. Wenn Sie die Rezepturwerte bei der Projektierung bereits wissen, geben Sie für jedes Rezepturelement den entsprechenden Wert ein.

Elemente | Datensätze

Name	Anzeigename	Nummer	Wasser	Konzentrat	Zucker	Aroma	Kommentar
Getränk	Getränk	1	30	70	45	600	
<Hinzufügen>							

7. Legen Sie so viele Datensätze an, wie Sie für die Rezeptur benötigen.

Elemente | Datensätze

Name	Anzeigename	Nummer	Wasser	Konzentrat	Zucker	Aroma	Kommentar
Getränk	Getränk	1	30	70	45	600	
Nektar	Nektar	2	50	50	10	300	
Saft	Saft	3	5	95	3	100	
<Hinzufügen>							

Eingabe der Werte in Runtime

Um Werte in Runtime in die Rezepturdatensätze einzugeben, haben Sie folgende Möglichkeiten:

- Übertragung von Daten direkt aus der Steuerung (Teach-In-Betrieb)
- Import der Werte aus einer CSV-Datei
- Eingabe der Werte am Bediengerät

Hinweis

Basic Panels

Der Import von Werten steht für diese Geräte nicht zur Verfügung.

Ergebnis

Die komplette Rezeptur ist konfiguriert.

Rezepturdatensätze mit Datums- oder Zeitangaben

Wenn Sie Datums- oder Zeitangaben verwenden, achten Sie darauf, dass die System-Einstellung für Uhrzeit und Datum auf dem Projektierungsrechner mit der auf dem Zielsystem übereinstimmt. Beispiel: Sie laden um 13:55 einen Rezepturdatensatz auf das Zielsystem, in dem die Bearbeitungszeit 14 Uhr gespeichert ist. Wenn es auf dem Zielrechner bereits 14:05 ist, wird die Rezeptur nicht mehr bearbeitet. Wenn ein Bediener die Rezeptur bearbeitet, werden Änderungsinformationen nicht korrekt in die Datenbank zurückgeschrieben.

Überprüfen Sie nach dem Laden auf das Zielsystem Rezepturen mit Datums- oder Zeitangaben auf dem Zielsystem.

Rezeptur bearbeiten

Zweck

Sie wollen eine Rezeptur ändern, erweitern oder Teile löschen.

Voraussetzung

- Mindestens eine Rezeptur ist angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Rezeptureinstellungen ändern

Um die Rezeptureinstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Rezepturen" die Rezeptur, die Sie ändern wollen.
   Das Inspektorfenster wird geöffnet.
2. Ändern Sie die Konfiguration der Rezeptur im Inspektorfenster.

Rezepturelemente und Rezepturdatensätze ändern Sie auf die gleiche Weise.

Rezepturwerte ändern

Um Rezepturwerte zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, deren Werte Sie ändern wollen.
2. Klicken Sie auf die Registerkarte "Datensätze".
3. Geben Sie in den Wertespalten die neuen Werte ein.

Rezepturelement hinzufügen

Um einer Rezeptur weitere Rezepturelemente hinzuzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Rezepturen" die Rezeptur, der Sie weitere Elemente hinzufügen wollen.
2. Klicken Sie auf die Registerkarte "Elemente".
3. Klicken Sie in der ersten freien Zeile auf "Hinzufügen".
   Das Rezepturelement wird angelegt.
4. Konfigurieren Sie das Rezepturelement.

Rezepturdatensätze fügen Sie auf die gleiche Weise hinzu.

Rezepturen verwalten

Voraussetzung

- Eine Rezeptur mit Rezepturelementen und Rezepturdatensatz ist angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Rezepturen umbenennen

Bei Rezepturen, Rezepturelementen und Rezepturdatensätzen wird zwischen den internen Namen und den Anzeigenamen unterschieden.

Um Rezepturen umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie umbenennen wollen.
   Das Inspektorfenster wird geöffnet.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie den neuen Namen ein.
   Rezepturelemente und Rezepturdatensätze benennen Sie auf der entsprechenden Registerkarte auf die gleiche Weise um.

Hinweis

Die Anzeigenamen im Editor "Rezepturen" können Sie auch unter "Sprachen & Ressourcen > Projekttexte" umbenennen. Diese Möglichkeit bietet sich an, wenn Sie z. B. bereits in mehreren Sprachen projektiert haben.

Rezepturen kopieren und einfügen

Um Rezepturen zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie kopieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie in der ersten freien Tabellenzeile im Kontextmenü den Befehl "Einfügen".

Die kopierte Rezeptur wird in die Tabelle eingefügt. Mit der Rezeptur werden in der entsprechenden Registerkarte auch die Rezepturelemente und Rezepturdatensätze mitkopiert.

Zusätzlich kopieren Sie Rezepturelemente und Rezepturdatensätze auf der entsprechenden Registerkarte auf die gleiche Weise.

Wenn bereits ein Rezepturdatensatz mit dem selben Namen existiert, wird der Name des kopierten Rezepturdatensatzes um eine Ziffer erweitert. Damit ist die Eindeutigkeit der Namen gewährleistet. Rezepturdatensätze können nur innerhalb einer Rezeptur kopiert und eingefügt werden.

Rezeptur löschen

Um eine Rezeptur zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie löschen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Die Rezeptur wird gelöscht.

Rezepturelemente und Rezepturdatensätze löschen Sie auf der entsprechenden Registerkarte auf die gleiche Weise.

Hinweis

Beim Löschen einer Rezeptur werden die darin enthaltenen Rezepturdatensätze gelöscht.

Hinweis

Beim Löschen eines Rezepturelements werden die dazugehörigen Werte in den Rezepturdatensätzen ebenfalls gelöscht. Die zugeordneten Variablen bleiben erhalten.

Rezepturen in die Projektierung importieren und wieder exportieren

Einleitung

Sie können Rezepturen als CSV-Dateien exportieren und wieder in die Projektierung importieren.

Anwendungsfall

Sie exportieren Rezepturdaten, um die Rezepturdaten langfristig auf einem Rechner abzulegen und zu sichern.

Um Rezepturdaten zu vereinheitlichen und zu verteilen, exportieren Sie die Rezepturdaten auf einem Bediengerät. Sie ändern die CSV-Datei in Microsoft Excel und importieren die CSV-Datei in alle Bediengeräte, die die gleichen Rezepturdaten benötigen.

Sie tauschen Rezepturdaten zwischen verschiedenen Projekten aus. Sie ändern Rezepturdaten in Runtime. Um die geänderten Rezepturdaten nach WinCC ES zu übertragen, exportieren Sie die Rezepturdatensätze in Runtime und kopieren die CSV-Datei auf den Projektierungsrechner. Dort importieren Sie die CSV-Datei mit den Rezepturdatensätzen in die Rezeptur. Und in umgekehrter Richtung: in WinCC ES geänderte Rezepturdatensätze übertragen Sie in die Runtime auf dem Bediengerät.

Voraussetzung

- Die Rezeptur ist angelegt.
- Die Struktur der exportierten CSV-Datei und der Rezeptur im WinCC ES stimmen überein: Name, Anzahl und Datentyp der Rezepturelemente sind jeweils gleich.
- Der Editor "Rezepturen" ist geöffnet.
- In der Importdatei sind Semikolon und Kommazeichen ausschließlich als Trennzeichen verwendet.

CSV-Datei importieren

Um Rezepturdatensätze in eine Rezeptur zu importieren, gehen Sie wie folgt vor:

1. Markieren Sie in der Registerkarte "Rezepturen" die Zeile mit der gewünschten Rezeptur.
2. Wählen Sie im Kontextmenü den Befehl "Rezepturdaten importieren".
   Der Dialog "Import" wird geöffnet.

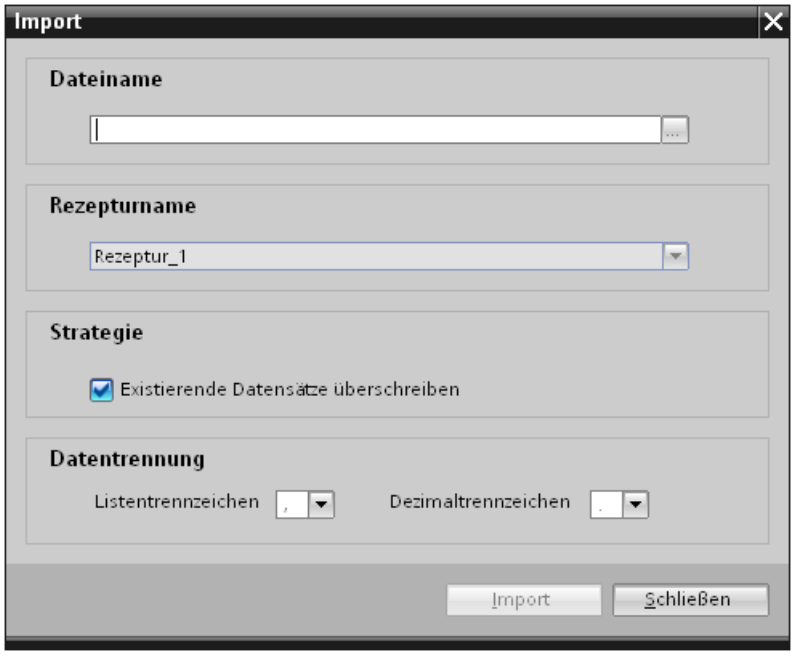

3. Wählen Sie unter "Dateiname" die gewünschte CSV-Datei.
4. Legen Sie unter "Strategie" fest, ob ein Rezepturdatensatz, der mit der gleichen Rezepturnummer in WinCC ES vorhanden ist, überschrieben wird.
5. Wählen Sie unter "Datentrennung" das Listen- und Dezimaltrennzeichen. Das Listentrennzeichen trennt in der CSV-Datei einzelne Rezepturelemente. Das Dezimaltrennzeichen trennt Ganzzahl und Nachkommastellen.

   Hinweis

   Verwenden Sie beim Import das gleiche Listentrennzeichen wie in der CSV-Datei beim Export.

6. Starten Sie den Vorgang mit der Schaltfläche "Import".

Ergebnis

Die Rezeptur wird um nicht vorhandene Rezepturdatensätze aus der CSV-Datei ergänzt. Falls gewählt, werden vorhandene Rezepturdatensätze überschrieben.

Wenn die Struktur der Rezeptur nicht mit der Struktur der CSV-Datei übereinstimmt, wird im Inspektorfenster unter "Info" eine Fehlermeldung ausgegeben.

CSV-Datei exportieren

1. Markieren Sie in der Registerkarte "Rezepturen" die Zeile mit der Rezeptur, die Sie exportieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Rezepturdaten exportieren".
   Der Dialog "Export" wird geöffnet.

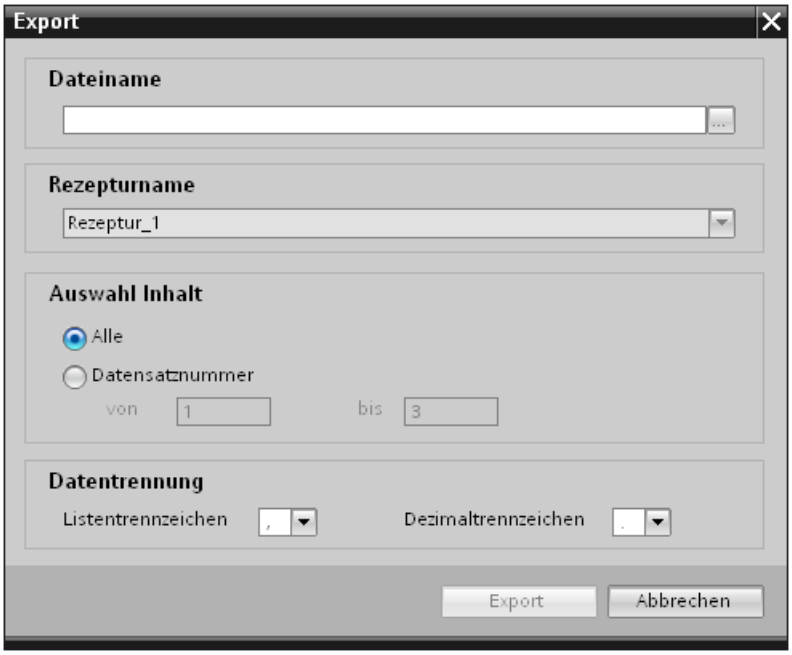

3. Geben Sie unter "Dateiname" den Pfad der CSV-Datei ein.
4. Wählen Sie unter "Auswahl Inhalt" alle Rezepturdatensätze oder schränken Sie die Auswahl auf bestimmte Rezepturdatensatznummern ein.
5. Wählen Sie unter "Datentrennung" das Listen- und Dezimaltrennzeichen. Das Listentrennzeichen trennt in der CSV-Datei einzelne Rezepturelemente. Das Dezimaltrennzeichen trennt Ganzzahl und Nachkommastellen.
6. Starten Sie den Vorgang mit der Schaltfläche "Export".

Ergebnis

Die Rezeptur wird als CSV-Datei exportiert.

6.1.3.2 Anzeige der Rezeptur projektieren

Einfache Rezepturanzeige projektieren

Voraussetzung

- Die Rezeptur ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Das Bild ist angelegt und geöffnet.

ACHTUNG
Datenverlust bei mehreren Rezepturanzeigen im Bild Gilt nur für Basic Panels, OP73, OP77A, TP177A und TP177A (Portrait): Wenn in einem Bild zwei oder mehr Rezepturanzeigen die gleiche Rezeptur anzeigen, kommt es zu einem Konflikt beim Datenzugriff. Datenverlust und unvorhersehbarer Zustand von Rezepturdaten sind die Folge. Stellen Sie sicher, dass der Bediener in unterschiedlichen Rezepturanzeigen nicht die gleiche Rezeptur auswählen und ändern kann. • Zeigen Sie in einer Rezepturanzeige jeweils nur eine Rezeptur an. • Zeigen Sie in jeder Rezepturanzeige jeweils eine andere Rezeptur an.

Vorgehen

Um eine einfache Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie die Rezepturanzeige in das Bild ein. Die Rezepturanzeige finden Sie in der Task Card "Werkzeuge" unter "Controls".
2. Nur bei Geräten, welche auch die erweitere Rezepturanzeige unterstützen: Aktivieren Sie unter "Eigenschaften > Darstellung > Modus" "Einfache Ansicht".

3. Wenn Sie in der Rezepturanzeige nur die Rezepturdatensätze einer bestimmten Rezeptur anzeigen wollen, wählen Sie unter "Eigenschaften > Allgemein > Rezeptur" die gewünschte Rezeptur aus.

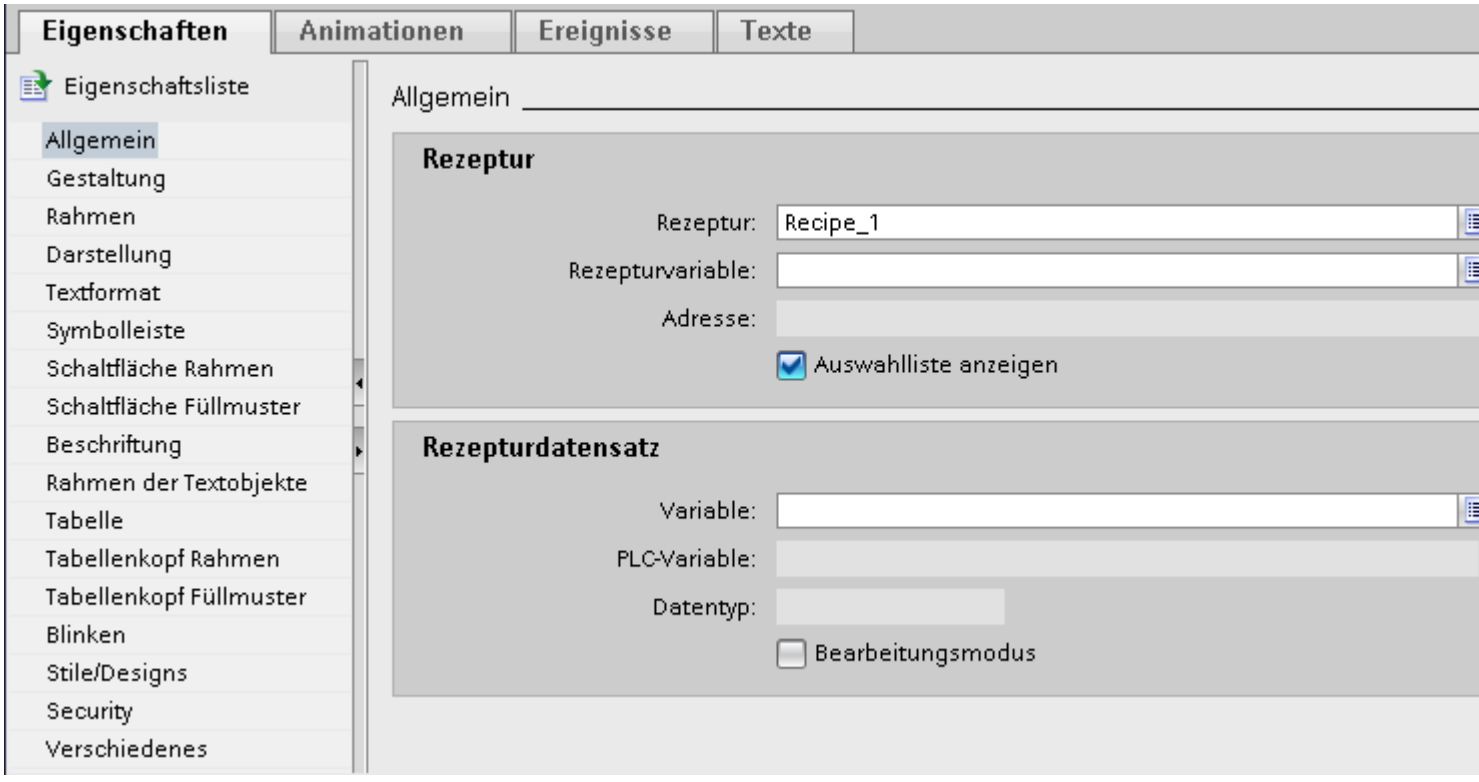

4. Wenn Sie in der Rezepturanzeige die Rezepturdaten nur anzeigen wollen, deaktivieren Sie im Bereich "Rezepturdatensatz" "Bearbeitungsmodus".
5. Legen Sie unter "Eigenschaften > Gestaltung" und "Eigenschaften > Darstellung" zusätzliche Anzeigeoptionen der Rezepturanzeige fest.
6. Legen Sie unter "Eigenschaften > Symbolleiste" fest, welche Menübefehle in Runtime in der Rezepturanzeige verfügbar sind.

Ergebnis

Die einfache Rezepturanzeige ist projektiert. In Runtime können Sie die Rezepturanzeige zum Anzeigen oder Bearbeiten der Rezepturdaten verwenden.

Das Deaktivieren des Bearbeitungsmodus unter "Eigenschaften > Eigenschaften > Allgemein" hat keine Auswirkung auf die Schaltflächen in der Symbolleiste. Die unter "Eigenschaften > Symbolleiste" aktivierten Schaltflächen sind auch bei deaktiviertem Bearbeitungsmodus bedienbar.

Erweiterte Rezepturanzeige projektieren

Voraussetzung

- Die Rezeptur ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Das Bild ist angelegt und geöffnet.

Vorgehen

Um eine erweiterte Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie die Rezepturanzeige in das Bild ein. Die Rezepturanzeige finden Sie in der Task Card "Werkzeuge > Controls".
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Darstellung > Modus > Erweiterte Ansicht".

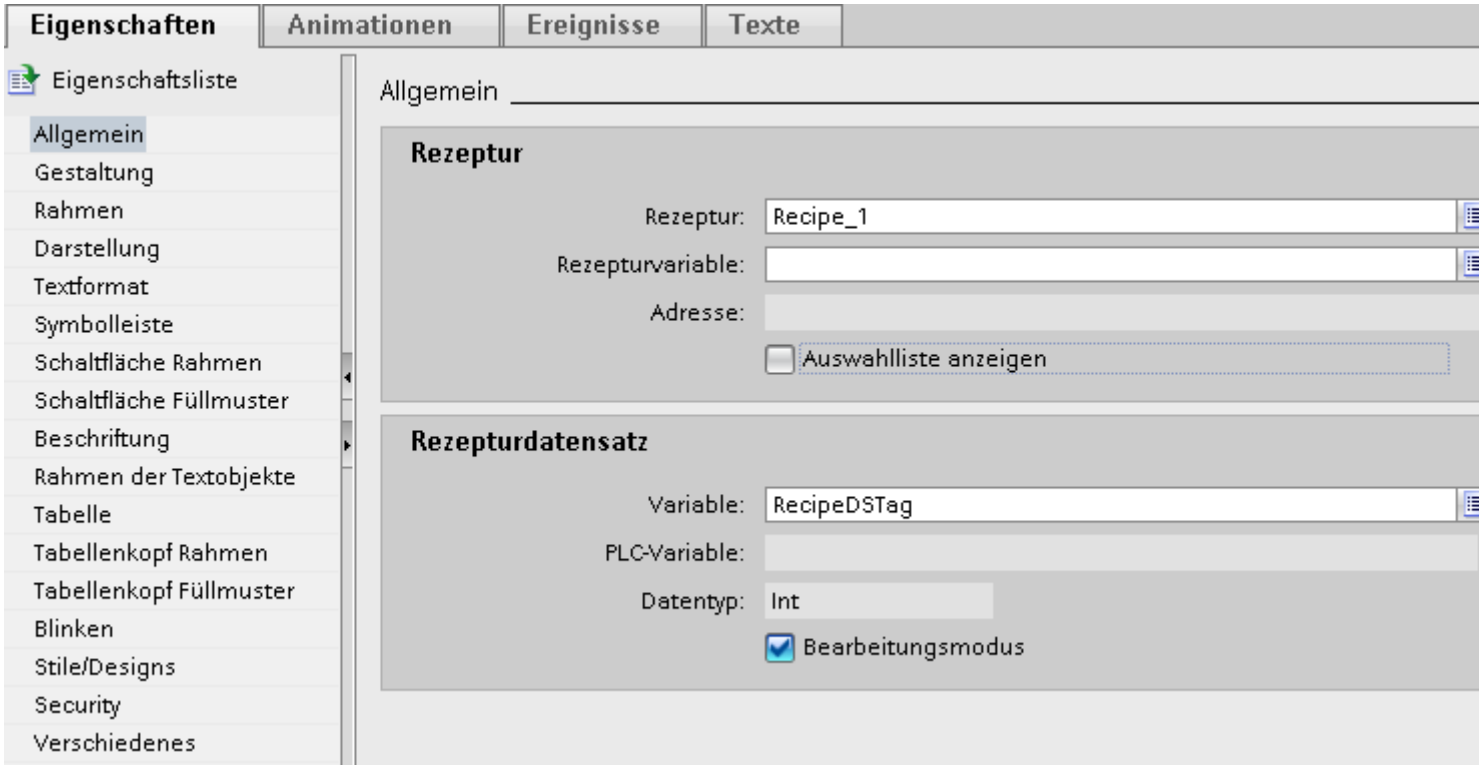

3. Geben Sie im Inspektorfenster in der Gruppe "Allgemein" die gewünschten Einstellungen ein.
   - Wenn Sie in der Rezepturanzeige nur die Rezepturdatensätze einer bestimmten Rezeptur anzeigen wollen, wählen Sie im Bereich "Rezeptur" unter "Rezeptur" die gewünschte Rezeptur aus.
   - Wenn Sie den Rezepturnamen oder die Rezepturnummer in einer Variablen speichern wollen, wählen Sie im Bereich "Rezeptur" unter "Rezepturvariable" die gewünschte Variable aus.
   - Wenn Sie den Rezepturdatensatznamen oder die Rezepturdatensatznummer in einer Variablen speichern wollen, wählen Sie im Bereich "Rezepturdatensatz" unter "Variable" die gewünschte Variable aus.
   - Wenn Sie in der Rezepturanzeige die Rezepturdaten nur anzeigen wollen, deaktivieren Sie "Bearbeitungsmodus".
   - Wenn Sie die Rezepturanzeige nur zum Auswählen von Rezepturen verwenden wollen, deaktivieren Sie unter "Eigenschaften > Tabelle > Tabelle anzeigen".
4. Legen Sie unter "Eigenschaften > Gestaltung" und "Eigenschaften > Darstellung" zusätzliche Anzeigeoptionen der Rezepturanzeige fest.

5. Wenn Sie die Beschriftung in der Rezepturanzeige ändern wollen, geben Sie unter "Eigenschaften > Beschriftung" eigene Texte ein.
6. Legen Sie unter "Eigenschaften > Symbolleiste" fest, welche Schaltflächen in Runtime in der Rezepturanzeige verfügbar sind.

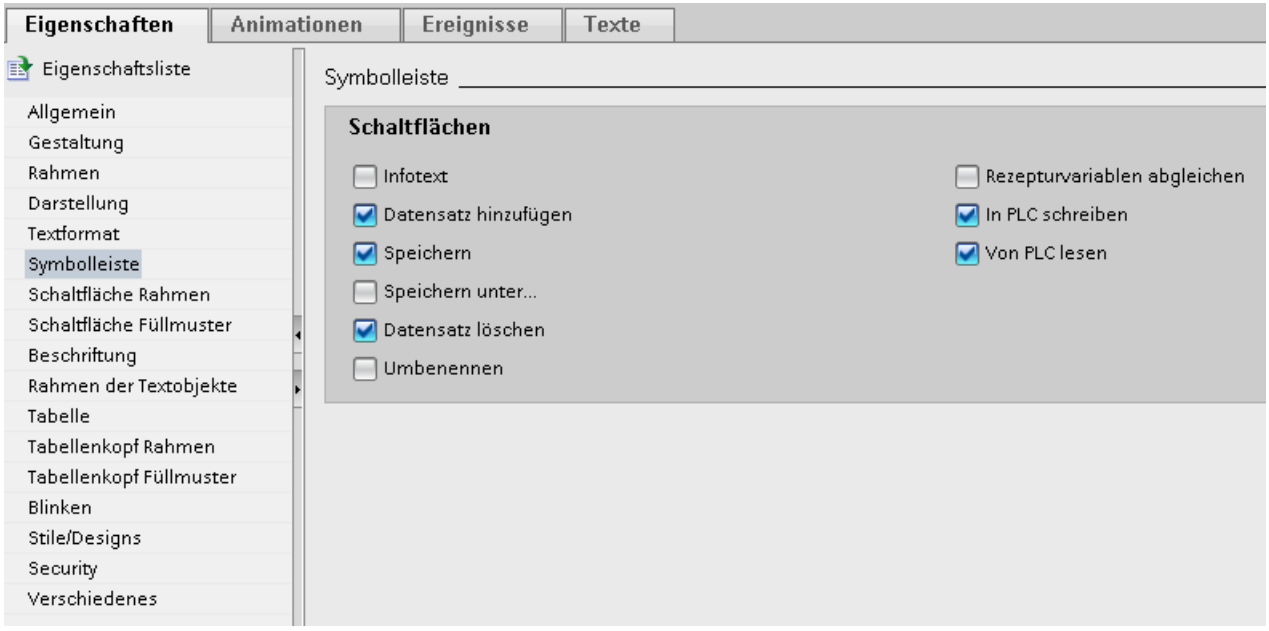

Hinweis

Wenn Sie im Kontextmenü der Rezepturanzeige den Befehl "Bearbeiten" wählen, wird die Rezepturanzeige aktiv geschaltet. Zur Aktivierung der Rezepturanzeige muss der Zoomfaktor auf 100 % eingestellt sein.

Im aktiven Modus können Sie die Spaltenbreite und die Position der Spalten "Eintragsname" und "Wert" einstellen.

Ergebnis

Die Rezepturanzeige ist projektiert. In Runtime können Sie die Rezepturanzeige zum Anzeigen oder Bearbeiten der Rezepturdaten verwenden.

Das Deaktivieren des Bearbeitungsmodus unter "Eigenschaften > Eigenschaften > Allgemein" hat keine Auswirkung auf die Schaltflächen in der Symbolleiste. Die unter "Eigenschaften > Symbolleiste" aktivierten Schaltflächen sind auch bei deaktiviertem Bearbeitungsmodus bedienbar.

Rezepturbild projektieren

Einleitung

Das Rezepturbild ist ein Bild, in dem Sie im Editor "Bilder" eine individuelle Eingabemaske projektieren. Die Eingabemaske erstellen Sie aus Ein-/Ausgabefeldern und anderen Anzeige- und Bedienobjekten. Mit Systemfunktionen projektieren Sie die Rezepturfunktionalität, z. B. das Speichern von Rezepturdatensätzen.

Hinweis

Verfügbarkeit

Rezepturbilder können nicht für Basic Panels und OP73, OP77A, TP177A (Portrait) erstellt werden.

Verwendung

Sie können Rezepturdatensätze mit vielen Einträgen über mehrere Bilder verteilen. Sie können z. B. für jeden Anlagenbereich ein Bild mit den dazugehörigen Eingabemasken für die Rezepturdatensätze projektieren.

Sie können in einem Bild Ihre Maschine mit grafischen Bildobjekten visuell nachbilden. Damit können Sie Parametrierungseinstellungen anschaulicher darstellen, indem Sie E/A-Felder direkt neben Maschinenelementen wie Achsen oder Führungsschienen platzieren. Sie können damit einen direkten Bezug zwischen den Werten und der Maschine herstellen.

Voraussetzung

- Die Rezeptur ist angelegt.
- Der Editor "Bilder" ist geöffnet.

Vorgehen

Um ein Rezepturbild zu projektieren, gehen Sie folgendermaßen vor:

1. Projektieren Sie das Bild und legen Sie darin die E/A-Felder für die Eingabemaske der Rezeptur an.
   Je nach Größe und Komplexität der Rezeptur können Sie auch mehrere Bilder anlegen.
2. Projektieren Sie die E/A-Felder mit den Variablen, die Sie mit den Rezepturelementen verbunden haben.
3. Projektieren Sie E/A-Felder zur Rezeptur- und Rezepturdatensatzauswahl.

Alternatives Vorgehen:

1. Projektieren Sie eine Rezepturanzeige als Auswahlliste für Rezepturdatensätze und Rezepturen.
2. Blenden Sie in der Rezepturanzeige die nicht benötigten Schaltflächen aus.
3. Projektieren Sie an die projektierten Bedienelemente die Systemfunktionen zum Bearbeiten der Rezepturdatensätze.
   Bedienelemente sind projektierte Schaltflächen im Bild oder Funktionstasten am Bediengerät.
   Die Systemfunktionen zum Bearbeiten von Rezepturdatensätzen finden Sie unter "Rezepturen" und "Tastaturbedienung für Bildobjekte".

Ergebnis

Das Rezepturbild ist angelegt.

6.1.4 Beispiele

6.1.4.1 Beispiel für das Erstellen einer Rezeptur

Aufgabe

In diesem Beispiel erstellen Sie drei Rezepturen für eine Fruchtsaftmischanlage. Mit der Fruchtsaftmischanlage werden die Geschmacksrichtungen Orange, Apfel und Tropical produziert. Für jede Geschmacksrichtung erstellen Sie jeweils eine Rezeptur.

Jede Rezeptur enthält für folgende Mischungsverhältnisse jeweils einen Rezepturdatensatz:

- Getränk
- Nektar
- Saft

Einstellungen

Die Einstellungen beziehen sich auf ein Bediengerät, das an eine SIMATIC S7-300 oder SIMATIC S7-400 gekoppelt ist.

Für das Beispiel benötigen Sie folgende Variablen, Rezepturen, Rezepturelemente und Rezepturdatensätze:

Variablen:

Tabelle 6-3

Name	Steuerungsanbindung	Adresse	Typ
LitreWater	Ja	DB 120, DBW 0	Integer
LitreConcentrate	Ja	DB 120, DBW 4	Integer

Name	Steuerungsanbindung	Adresse	Typ
KiloSugar	Ja	DB 120, DBW 8	Integer
GramAroma	Ja	DB 120, DBW 12	Integer

Rezepturen:

- Orange
- Apfel
- Tropical

Rezepturelemente:

Rezepturelement	Verbundene Variable
Liter Wasser	LitreWater
Liter Konzentrat	LitreConcentrate
Kilo Zucker	KiloSugar
Gramm Aroma	GramAroma

Rezepturdatensätze Getränk, Nektar und Saft:

Datensatzname	Liter Wasser	Liter Konzentrat	Kilo Zucker	Gramm Aroma
Getränk	30	70	45	600
Nektar	50	50	10	300
Saft	5	95	3	100

Vorgehen

Um die Rezeptur zu erstellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie die Variablen "LitreWater", "LitreConcentrate", "KiloSugar" und "GramAroma" mit den oben genannten Einstellungen.
2. Erstellen Sie die Rezepturen "Orange", "Apfel" und "Tropical" mit den oben genannten Einstellungen. Legen Sie in jeder Rezeptur die Rezepturelemente an.

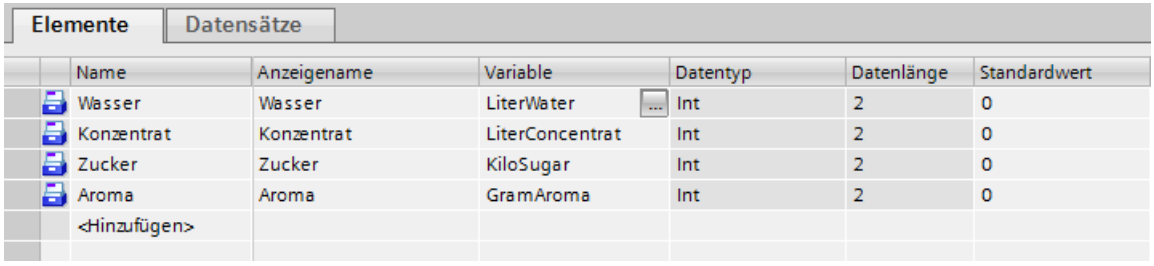

3. Nicht bei Basic Panels: Konfigurieren Sie jede Rezeptur so, dass die Rezepturdatensätze zwischen Rezepturbild und Rezepturanzeige abgegelichen werden. Die Werte der Rezepturvariablen sollen nicht automatisch an die Steuerung übertragen werden.
   Dazu sind im Eigenschaftsfenster der jeweiligen Rezeptur folgende Einstellungen notwendig:
   Unter "Eigenschaften > Extras":
   – Aktivieren Sie "Rezepturanzeige und Rezepturvariablen abgleichen".
   – Aktivieren Sie "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
4. Legen Sie in jeder Rezeptur die oben genannten Datensätze an. Geben Sie in jeden Datensatz die oben genannten Werte ein.

Elemente | Datensätze

Name	Anzeigename	Nummer	Wasser	Konzentrat	Zucker	Aroma	Kommentar
Getränk	Getränk	1	30	70	45	600	
Nektar	Nektar	2	50	50	10	300	
Saft	Saft	3	5	95	3	100	
<Hinzufügen>							

Ergebnis

Die drei Rezepturen "Orange", "Apfel" und "Tropical" sind angelegt.

Siehe auch

Konfigurationsmöglichkeiten der erweiterten Rezepturanzeige (ab V13) (Seite 1168)

6.1.4.2 Beispiel für das Projektieren eines Rezepturbildes

Aufgabe

In diesem Beispiel erstellen Sie ein Rezepturbild, in dem die Werte der Fruchtsaftmischanlage dargestellt werden. Um die Rezepturen und die dazugehörigen Rezepturdatensätze auszuwählen, verwenden Sie eine Rezepturanzeige.

Folgende Funktionen sollen über Schaltflächen bedienbar sein:

- Schaltfläche "Laden"
  Der ausgewählte Rezepturdatensatz wird vom Rezepturspeicher des Bediengeräts geladen und im Rezepturbild angezeigt.
- Schaltfläche "Speichern"
  Die angezeigten Rezepturvariablen werden in den Rezepturspeicher des Bediengeräts gespeichert.

- Schaltfläche "Daten an PLC"
  Die angezeigten Rezepturvariablen werden zur Steuerung übertragen.
- Schaltfläche "Daten von PLC"
  Der aktuelle Rezepturdatensatz in der Steuerung wird in die Rezepturvariablen übertragen und am Bediengerät angezeigt.

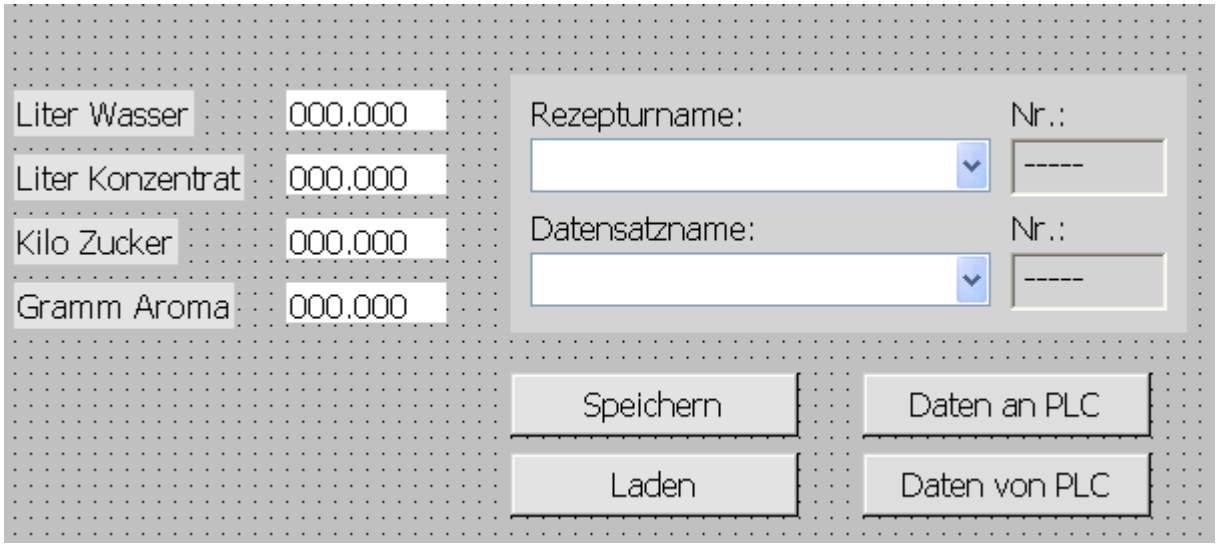

Voraussetzung

- Das Anwendungsbeispiel "Rezeptur erstellen" ist erstellt.
- Das Bild "Fruchtsaftmischanlage" ist angelegt und geöffnet.

Einstellungen

Für das Beispiel benötigen Sie folgende Variablen und Schaltflächen mit folgenden Einstellungen:

Variablen:

Name	Steuerungsanbindung	Typ
RecipeNumber	Nein	Integer
DataRecordNumber	Nein	Integer

Schaltflächen:

Beschriftung	Projektiertes Ereignis	Systemfunktion
Laden	Klicken	LadeDatensatz
Speichern	Klicken	SpeichereDatensatz
Daten an PLC	Klicken	SchreibeDatensatzvariablenInSteuerung
Daten von PLC	Klicken	LeseDatensatzvariablenAusSteuerung

Vorgehen

Um ein Rezepturbild zu projektieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus dem Objektfenster die Variablen "LitreWater", "LitreConcentrate", "KiloSugar" und "GramAroma" jeweils mit Drag&Drop in das Prozessbild "Fruchtsaftmischanlage".
   Vier EA-Felder werden erstellt, die mit den angegebenen Variablen verbunden sind.
2. Fügen Sie eine Rezepturanzeige ein, die nur die Auswahlfelder für Rezepturname und Datensatzname enthält.
   Nehmen Sie dazu im Inspektorfenster der Rezepturanzeige folgende Einstellungen vor:
   - Wählen Sie unter "Allgemein" den Anzeigetyp "Erweiterte Ansicht".
   - Deaktivieren Sie unter "Eigenschaften > Allgemein" "Bearbeitungsmodus" und unter "Eigenschaften > Tabelle" "Tabelle anzeigen".
   - Verbinden Sie das Feld "Rezepturvariable " mit der Variablen "RecipeNumber".
   - Verbinden Sie das Feld "Variable " mit der Variablen "DataRecordNumber".
   - Deaktivieren Sie unter "Eigenschaften > Schaltflächen" alle Schaltflächen.
3. Projektieren Sie vier Schaltflächen mit den oben genannten Einstellungen. Übergeben Sie als Parameter für Rezepturnummer und Rezepturdatensatznummer jeweils die Variablen "RecipeNumber" und "DataRecordNumber".

Ergebnis

In Runtime können Sie in der Rezepturanzeige die Rezeptur und den dazugehörigen Rezepturdatensatz auswählen und die Rezepturwerte ändern. Sie können mit den Schaltflächen die Rezepturdatensätze laden, speichern und übertragen.

6.1.4.3 Szenario für das Eingeben von Rezepturdatensätzen in Runtime

Ziel

Sie wollen Produktionsdaten am Bediengerät eingeben, ohne den aktiven Prozess zu stören. Die Produktionsdaten sollen deswegen nicht an die Steuerung übertragen werden.

Voraussetzung

- Die Rezeptur ist angelegt.
- Die Rezeptur hat folgende Einstellungen:
  - "Rezepturanzeige und Rezepturvariablen abgleichen" ist aktiviert oder deaktiviert.
  - Wenn "Rezepturanzeige und Rezepturvariablen abgleichen" aktiviert ist, muss " Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" aktiviert sein. Damit wird ein automatisches Übertragen der Rezepturvariablen zwischen Bediengerät und Steuerung verhindert.
- Ein Rezepturbild oder ein Bild mit Rezepturanzeige ist vorhanden.
- Ein Bedienobjekt zum Speichern der Rezepturdatensätze ist vorhanden.

Ablauf

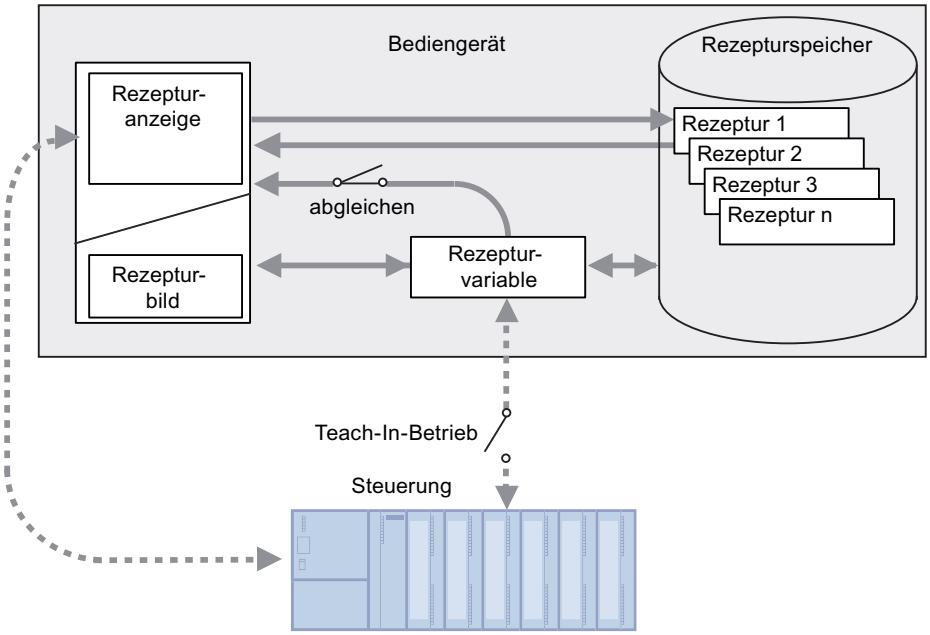

1. Geben Sie in der Rezepturanzeige oder dem Rezepturbild die Produktionsdaten ein.
2. Speichern Sie den geänderten Rezepturdatensatz.

Übertragen der Rezepturdaten zur Steuerung

Abhängig von der Projektierung können Bedienobjekte zum Übertragen der Rezepturdaten in die Steuerung vorhanden sein.

6.1.4.4 Szenatio für einen manuellen Produktionsablauf

Ziel

Ein an die Steuerung angeschlossenes Lesegerät liest am zu bearbeitenden Werkstück einen Barcode aus. Die Rezepturdatensatznamen entsprechen dabei den jeweiligen Barcodebezeichnungen. Damit kann die Steuerung den benötigten Rezepturdatensatz vom Speichermedium des Bediengeräts laden. Der Rezepturdatensatz wird zur Kontrolle am Bildschirm angezeigt.

Sie wollen die übertragenen Produktionsdaten gegebenenfalls online korrigieren können.

Voraussetzung

- Die Rezeptur ist angelegt.
- Die Rezeptur hat folgende Einstellungen:
  - "Rezepturanzeige und Rezepturvariablen abgleichen" ist aktiviert.
  - "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)" ist deaktiviert.

  Hinweis

  Die Änderungen werden sofort in die Steuerung übertragen.

- Ein Rezepturbild ist vorhanden.
  Bei Bedarf kann ein Bedienobjekt zum Abspeichern der Rezepturdatensätze im Rezepturbild vorhanden sein.

Ablauf

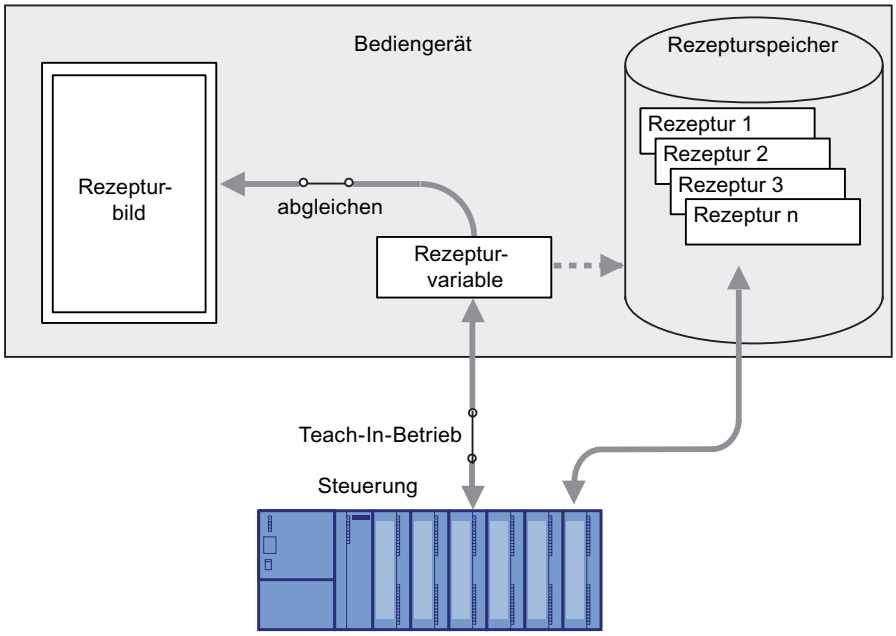

Verhalten bei Einsatz der Rezepturanzeige

Beim Einsatz der Rezepturanzeige ist ein sofortiges Übertragen von Änderungen nicht möglich. Sie müssen das Bedienobjekt zum Übertragen des Rezepturdatensatzes in die Steuerung bedienen.

6.1.4.5 Szenario für einen automatischen Produktionsablauf

Ziel

Sie wollen eine Produktion automatisiert ablaufen lassen. Die Produktionsdaten sollen entweder vom Rezepturspeicher des Bediengeräts oder von einem externen Speichermedium direkt an die Steuerung übertragen werden. Die Anzeige am Bildschirm ist nicht notwendig.

Voraussetzung

- Die Rezeptur ist angelegt.
- Die Rezeptur hat folgende Einstellungen:
  - "Koordinierte Übertragung der Datensätze" ist aktiviert.
    Da die Produktionsdaten zur Steuerung übertragen werden, ist die koordinierte Übertragung mit der Steuerung notwendig, um ein unkontrolliertes, gegenseitiges Überschreiben zu verhindern.

Ablauf

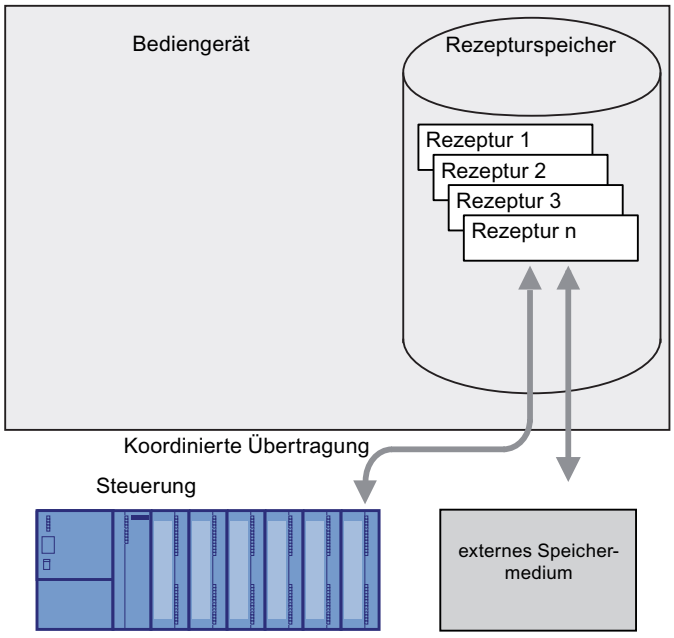

Realisierung

Um den Datenfluss zu steuern, haben Sie folgende Möglichkeiten:

- Das Steuerungsprogramm steuert die automatische Übertragung über Steuerungsaufträge oder bei Bedarf über Systemfunktionen von WinCC.
  Der Ablauf wird über die Statusinformationen im Datenfach und über Rückgabewerte der verwendeten Funktionen kontrolliert.
- Ein oder mehrere Skripte steuern die automatische Übertragung über Systemfunktionen von WinCC.
  Der Ablauf wird über die Rückgabewerte der verwendeten Funktionen kontrolliert.

Sie können den automatisierten Produktionsablauf mit den zur Verfügung stehenden Systemfunktionen realisieren:

- "ImportiereDatensaetze"
  Die Funktion lädt Datensätze aus einer *.CSV-Datei auf den Rezepturspeicher des Bediengeräts.
- "SchreibeDatensatzInSteuerung"
  Die Funktion überträgt einen Datensatz vom Rezepturspeicher des Bediengeräts zur Steuerung.

6.2 Mit Rezepturen arbeiten

6.2.1 Grundlagen

6.2.1.1 Definition und Einsatzgebiete

Einleitung

In Rezepturen werden zusammengehörige Daten, z. B. Maschinenparametrierung oder Produktionsdaten, zusammengefasst.

Beispiele:

- Maschinenparametrierungen, die notwendig sind, um die Produktion auf eine andere Produktvariante umzustellen.
- Produktbestandteile, die in unterschiedlichen Zusammensetzungen unterschiedliche Endprodukte ergeben.
- Daten aus technischen Prozessen, die in größeren Zeitabständen kontinuierlich abgespeichert werden.

Eine Rezeptur zeichnet sich durch eine feste Datenstruktur aus. Bei der Projektierung wird die Struktur einer Rezeptur einmalig festgelegt. Eine Rezeptur enthält Rezepturdatensätze, die sich in den Werten, nicht aber in der Struktur unterscheiden.

Rezepturen werden in einer SQL-Datenbank abgelegt. Ein Rezepturdatensatz wird grundsätzlich komplett in einem Arbeitsschritt zwischen Bediengerät und Steuerung übertragen. Zusätzlich können Sie Produktionsdaten in Runtime über eine CSV-Datei importieren.

Einsatz von Rezepturen

Für den Einsatz von Rezepturen sind z. B. folgende Fälle möglich:

- Manuelle Produktion
  Sie wählen die gewünschten Rezepturdaten und zeigen die Rezepturdaten am Bediengerät an. Sie ändern bei Bedarf die Rezepturdaten und speichern Sie auf dem Bediengerät ab. Sie übertragen die Rezepturdaten zur Steuerung.
- Automatische Produktion
  Das Steuerungsprogramm stößt die Übertragung der Rezepturdaten zwischen Steuerung und Bediengerät an. Sie können die Übertragung auch durch eine Bedienung am Bediengerät starten. Die Produktion wird damit automatisch umgestellt. Eine Anzeige oder Änderung der Daten ist nicht zwingend notwendig.
- Speicherung von Projektdaten
  Sie verwenden eine Rezeptur ohne Steuerungsanbindung zum Speichern von Projektdaten. In Runtime können Sie mit Skripten auf die in der Rezeptur gespeicherten Projektdaten zugreifen.

Sichten

Sichten beziehen Daten aus vorhandenen Rezepturen, die mindestens eine Gemeinsamkeit haben. Sie dienen der Zusammenfassung von Daten der Rezepturen.

Rezepturen und Sichten anzeigen

Um die Rezepturen und Sichten auf dem Bediengerät anzuzeigen und zu bearbeiten, gibt es folgende Möglichkeiten:

- Eine Rezepturanzeige innerhalb eines Prozessbildes
- Ein Rezepturbild

Eingeben und Ändern der Rezepturdaten

Sie geben die Daten in den einzelnen Rezepturdatensätzen ein und passen Sie bei Bedarf an. Folgende Möglichkeiten stehen Ihnen dazu zur Verfügung:

- Eingabe der Daten während der Projektierung
  Wenn die Produktionsdaten bereits feststehen, geben Sie diese Daten bereits während der Projektierung der Rezeptur im Editor "Rezepturen" ein.
- Eingabe der Daten in Runtime
  Wenn Sie Produktionsdaten häufig anpassen, ändern Sie die Daten direkt in Runtime.
  - Sie geben die Daten direkt am Bediengerät ein.
  - Sie stellen die Parameter direkt an der Maschine ein. Übertragen Sie dann die Daten von der Steuerung zum Bediengerät und speichern Sie in der Rezeptur.
  - Sie importieren die Rezepturdaten aus bereits ausgelagerten CSV-Dateien.

Siehe auch

Rezepturanzeige in Protokollen (Seite 1360)

Rezepturprotokoll (Seite 1361)

6.2.1.2 Aufbau von Rezepturen

Einleitung

Der prinzipielle Aufbau einer Rezeptur wird anhand einer Abfüllstation einer Fruchtsaftanlage dargestellt.

In einem Bediengerät können mehrere unterschiedliche Rezepturen vorhanden sein. Sie können eine Rezeptur z. B. mit einem Karteikasten, die mehrere Karteikarten enthält, vergleichen. In diesem Karteikasten sind mehrere Herstellungsvarianten einer Produktfamilie enthalten. Die Daten jeweils einer Herstellungsvariante sind komplett auf einer Karteikarte enthalten.

Beispiel:

Bei einer Anlage zur Getränkeherstellung wird eine Rezeptur für unterschiedliche Getränkevarianten benötigt. Es gibt z. B. die Getränkevarianten Fruchtsaftgetränk, Saft, Nektar.

Rezeptur

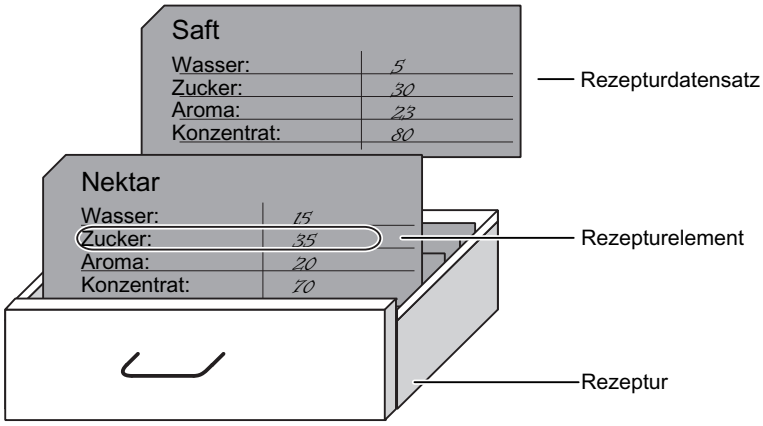

Rezepturdatensätze

Jede Karteikarte repräsentiert einen Rezepturdatensatz, der für die Herstellung einer Produktvariante notwendig ist.

Rezepturelemente

Jede Karteikarte einer Schublade ist gleich bedruckt. Alle Karteikarten enthalten Felder für die verschiedenen Zutaten. Jedes Feld entspricht einem Rezepturelement. Jeder Rezepturdatensatz enthält damit die gleichen Elemente. Die Datensätze unterscheiden sich jedoch im Wert der einzelnen Elemente.

Beispiel:

Alle Getränke enthalten die gleichen Elemente:

- Wasser
- Konzentrat
- Zucker
- Aroma

Die Datensätze für Fruchtsaftgetränk, Saft oder Nektar unterscheiden sich jedoch z. B. in der Menge des Zuckers, der bei der Herstellung verwendet wird.

Rezeptur	Rezepturdatensatz	Rezepturelemente	Wert
Getränke	1	Artikel	Saft
		Wasser	5
		Zucker	30
		Aroma	23
		Konzentrat	80
		Farbstoff	F30
	2	Artikel	Nektar
		Wasser	15
		Zucker	35
		Aroma	20
		Konzentrat	70
		Farbstoff	F35

Rezepturabfragen

Eine Rezepturabfrage ist eine Rezeptur, deren Elemente sich aus unterschiedlichen Rezepturen zusammensetzen. Beispiel:

```
Rezepturabfrage-Element1 = Rezeptur1_Rezepturelement3
Rezepturabfrage-Element2 = Rezeptur3_Rezepturelement7
Rezepturabfrage-Element3 = Rezeptur2_Rezepturelement10
```

Die verwendeten Rezepturen müssen bereits existieren.

Eine Rezepturabfrage wird in einer Rezepturanzeige angezeigt. Eine Rezepturabfrage besitzt jedoch keine eigenen Datensätze, sondern umfasst die Kombination aller Rezepturdatensätze der verwendeten Rezepturen. Beispiel: Wenn im obigen Beispiel zu jeder Rezeptur 10 Rezepturdatensätze existieren, umfasst die Rezepturabfrage 10*10*10 = 1000 Datensätze. Um diese Datensätze zu beschränken, können Sie in der Rezepturabfrage eine Relation definieren. Beispiel:

```
Rezepturabfrage-Element2 = Rezepturabfrage-Element3
```

Die Relation wirkt wie ein Filter für die Rezepturabfrage (siehe auch nachfolgendes Beispiel).

Beispiel

Für einen Etikettendruck wird aus den Rezepturen "Getränke" und "Bestellung" eine Rezepturabfrage "Etikett" erstellt.

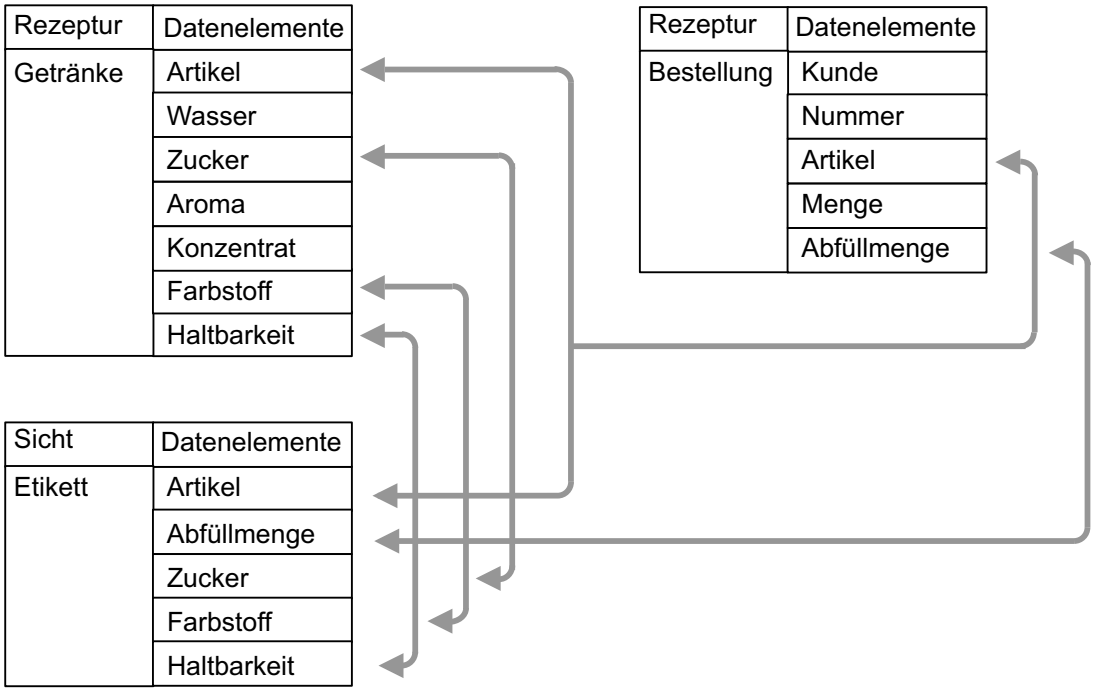

Die Abbildung zeigt unten die Rezepturabfrage "Etikett". Die benötigten Informationen für das Etikett sind in den Rezepturen "Getränke" und "Bestellung" enthalten. Aus der Rezeptur "Getränke" gehen die Rezepturelemente "Artikel", "Zucker", "Farbstoff" und "Haltbarkeit" in die Rezepturabfrage "Etikett" ein, aus der Rezeptur "Bestellung" die Rezepturelemente "Artikel" und "Abfüllmenge".

Das Rezepturelement "Artikel"' ist doppelt vorhanden:
`Getränke.Artikel, Bestellung.Artikel, Getränke.Wasser, Bestellung.Abfüllmenge...`

Die Rezepturanzeige zeigt zunächst jede Kombinationen dieser Daten in einer eigenen Zeile. Auf das Etikett müssen jedoch die Rezepturdaten des bestellten Artikels und nicht eines anderen Artikels, was folgende Bedingung sicher stellt:
`Getränke.Artikel = Bestellung.Artikel`

Die Rezepturanzeige wird gefiltert und Zeile für Zeile nur die bestellten Artikel mit ihren Rezepturdaten angezeigt.

6.2.1.3 Darstellung von Rezepturen in Runtime

Sie haben folgende Möglichkeiten, Rezepturdaten in Runtime darzustellen:

- In E/A-Feldern in beliebigen Positionen innerhalb von Anlagenbildern
- Innerhalb einer Rezepturanzeige in Tabellenform

Eine Rezepturanzeige verbinden Sie bei der Projektierung mit einer ausgewählten Rezeptur bzw. einer Rezepturabfrage. Die Rezepturanzeige kann nur auf diese Rezeptur bzw. auf diese Rezepturabfrage zugreifen.

Sie können für den Zugriff verschiedene Berechtigungen zuweisen.

6.2.1.4 Kommunikation mit der Steuerung

Einleitung

Zwischen Rezepturen und der Steuerung werden Daten über zwei Arten übertragen:

- Über Rezepturvariablen mit Hilfe von Steuervariablen
- Über Telegramme mit Hilfe von Rohdatenvariablen

Kommunikationsart "Steuervariablen"

Über Steuervariablen wird ein kompletter Rezepturdatensatz in die Rezepturvariablen übertragen und umgekehrt. Die Rezepturvariablen besitzen eine Kommunikationsverbindung zur Steuerung.

Kommunikationsart "Rohdatenvariable"

Steuerung und Bediengerät tauschen Rohdatenvariablen als Telegramme mit festgelegtem Aufbau untereinander aus.

Rohdatenvariablen bieten folgende Vorteile:

- Mehrere Datensätze zwischen Steuerung und Bediengerät übertragen.
- Teile eines Rezepturdatensatzes übertragen.
- Bei mehreren Rezepturen die gleiche Rohdatenvariable verwenden.

Siehe auch

Übersicht (Seite 1242)

6.2.1.5 Editor "Rezepturen"

Einleitung

Im Editor "Rezepturen" erstellen, konfigurieren und bearbeiten Sie Folgendes:

- Rezepturen und die zugehörigen Rezepturelemente und Rezepturdatensätze
- Rezepturabfragen und die zugehörigen Rezepturabfrage-Elemente und Relationen

Darüber hinaus können Sie im Editor "Rezepturen" Werte in vorhandene Rezepturdatensätze eingeben.

Aufbau des Editors "Rezepturen"

Der Editor "Rezepturen" enthält im oberen Teil des Arbeitsbereichs die Registerkarten "Rezepturen" und "Rezepturabfragen".

Auf den Registerkarten "Rezepturen" und "Rezepturabfragen" legen Sie Rezepturen bzw. Rezepturabfragen an und konfigurieren diese im Inspektorfenster.

Welche Registerkarten im unteren Teil des Arbeitsbereichs angezeigt werden, hängt davon ab, welche Registerkarte im oberen Teil des Arbeitsbereichs aufgerufen ist:

Registerkarte "Rezepturen"

- Rezepturelemente
  Hier definieren Sie über die Tabellenzellen die Rezepturelemente der ausgewählten Rezeptur. Rezepturelemente verschieben Sie innerhalb der Tabelle über die Kontextmenü-Einträge "Aufwärts" und "Abwärts".
- Rezepturdatensätze
  Hier definieren Sie über die Tabellenzellen die Werte der Rezepturdatensätze der ausgewählten Rezeptur.

Registerkarte "Rezepturabfragen"

- Rezepturabfrage-Elemente
  Hier definieren Sie über die Tabellenzellen die Rezepturabfrage-Elemente der ausgewählten Rezepturabfrage. Rezepturabfragen und Rezepturabfrage-Elemente verschieben Sie innerhalb der Tabelle über die Kontextmenü-Einträge "Aufwärts" und "Abwärts".
  Im Feld "Bedingungen" definieren Sie die Relationen zwischen den Elementen der Rezepturabfrage.

Inspektorfenster

Im Inspektorfenster konfigurieren Sie jeweils das Element, das in einer der Tabellen selektiert ist, z. B. eine Rezepturabfrage oder ein Rezepturdatensatz.

Einstellungen Rezepturen

Für Rezepturen sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Rezepturname	Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie sprechende Namen oder Bezeichnungen, die Sie einem Produkt direkt zuordnen können, z. B. "Fruchtsaft_Orange".
Rezepturnummer	Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig. Bei der Kommunkationsart "Rohdatenvariable" muss das Feld "PLC" im Telegramm mit einer gültigen Rezepturnummer übereinstimmen.
Version	Informationen zur Rezeptur. Voreingestellt ist hier das Datum und der Zeitpunkt der letzten Änderung der Rezeptur.
Pfad	Legt den Ablageort für die Rezepturen fest. Die Rezepturen werden nicht in einer Datei, sondern in einer Datenbank abgelegt.

Einstellung	Beschreibung
Typ	Sie können die Rezepturdatensätze auf eine selbst gewählte maximale Anzahl begrenzen.
Maximale Anzahl Datensätze	Maximale Anzahl der Rezepturdatensätze einer Rezeptur. Dadurch begrenzen Sie den benötigten Rezepturspeicher des Bediengeräts. Beim Größentyp "unbegrenzt" erscheint als Wert "0".
Kommunikationsart	Sie legen fest, wie Bediengerät und Steuerung standardmäßig Rezepturdaten austauschen: • Keine Kommunikation • Rohdatenvariablen • Variablen
Rohdatenvariable	Variable, die bei der Kommunikationsart "Rohdatenvariable" die Rezepturdaten zwischen Steuerung und Bediengerät als Telegramm überträgt.
Steuervariable Auftrag, ID, Feld, Wert	Vier Variablen, die bei der Kommunikationsart "Variablen" die Datenübertragung zwischen Rezeptur und Rezepturvariablen und damit der Steuerung regeln.
Zum Lesen Zum Schreiben	Name der Berechtigung, die in Runtime lesenden, schreibenden Zugriff auf die Rezeptur besitzt.
Letzter Zugriff Letzter Benutzer	Änderungsinformationen bzw. Zugriff auf eine Rezeptur werden in Runtime angezeigt.
Tooltip	Hilfetext zur Rezeptur für den Projekteur

Einstellungen Rezepturelemente

Für Rezepturelemente sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Rezepturelementname	Identifiziert ein Rezepturelement innerhalb der Rezeptur eindeutig. Vergeben Sie sprechende Namen oder Bezeichnungen, die Sie eindeutig zuordnen können, z. B. Achsenbezeichnungen an einer Maschine oder Zutaten wie "Aroma".
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie sprechende Namen oder Bezeichnungen, die Sie einem Produkt direkt zuordnen können, z. B. "Fruchtsaft_Orange".
Rezepturvariable	Eine zugeordnete Variable speichert in Runtime den aktuellen Wert des Rezepturelements im Rezepturdatensatz.
Datentyp	Datentyp der Rezepturvariable.
Datenlänge	Datenlänge der Rezepturvariable, abhängig vom Datentyp.
Standardwert	Wird als Standardeintrag verwendet, wenn Sie einen neuen Rezepturdatensatz anlegen.
Minimaler Wert	Kleinster darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.
Maximaler Wert	Größter darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.
Zum Lesen Zum Schreiben	Name der Berechtigung, die in Runtime lesenden, schreibenden Zugriff auf die Rezeptur besitzt.
Wert erforderlich	Das Rezepturelement hat einen Wert ungleich null.

Einstellung	Beschreibung
Eindeutige Werte	In allen Rezepturdatensätzen darf ein Wert für das Rezepturelement nur einmal vorkommen. Beispiel Rezepturelement "Bestellnummer".
Schnellsuche	Der Wert des Rezepturelements ist indiziert, was die Suche beschleunigt.
Tooltip	Hilfetext zum Rezepturelement für den Projekteur

Einstellungen Rezepturdatensätze

Für Rezepturdatensätze sind folgende Einstellungen möglich:

Einstellung	Beschreibung
Rezepturdatensatzname	Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.
Anzeigename	Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie sprechende Namen oder Bezeichnungen, die Sie einem Produkt direkt zuordnen können, z. B. Produktnummern.
Rezepturdatensatznummer	Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.
Rezepturelementname	Name des angelegten Rezepturelements. Unter diesem Namen sind die Werte abgelegt, die das Rezepturelement in den unterschiedlichen Rezepturdatensätzen (Zeilen) besitzt. Ein Wert bildet zusammen mit den Werten der anderen Rezepturelemente in einer Zeile jeweils einen Rezepturdatensatz. Die Werte können Sie bereits während der Projektierung hinterlegen.
Letzter Zugriff Letzter Benutzer	Änderungsinformationen bzw. Zugriff auf einen Rezepturdatensatz werden in Runtime angezeigt.
Kommentar	Kommentar zum Rezepturdatensatz

6.2.1.6 SQL-Sprache

SQL (Structured Query Language) ist eine leistungsstarke und verbreitete Datenbanksprache. In den Funktionen der WinCC-Skriptsprache wird die SQL-Sprache für Datenbankabfragen benutzt. Weiterführende Informationen entnehmen Sie bitte der entsprechenden Fachliteratur.

Bei einigen Standardfunktionen sowie bei einigen Funktionen im Editor "Rezepturen" müssen Sie Bedingungen in der Datenbanksprache SQL zur Spezifikation der zu bearbeitenden Datensätze angeben. Im folgenden sehen Sie einige Beispiele, wie eine SQL-Anweisung versorgt werden muß:

Hinweis

Schreibweise von Namen

Anhänge, Zusätze und die Namen einer Rezeptur, Rezepturabfrage oder eines Rezepturelements dürfen nur aus Buchstaben, Ziffern und Unterstrichen "_" bestehen und maximal 20 Zeichen lang sein. Das erste Zeichen muss immer ein Buchstabe sein.

- `FieldA > '1992-12-31 23:45:12.12'`
  Die Anweisung selektiert alle Datensätze, deren Wert in der Spalte "FieldA" größer als der angegebene Wert ist. FieldA ist vom Datentyp DB_TYP_TIME.
- `FieldB like 'tank%'`
  Damit werden zum Beispiel die Datensätze selektiert, die in der Spalte "FieldB" den Wert "tank1" oder "tank4" oder "tank12" usw. besitzen. FieldB ist vom Datentyp DB_TYP_CHAR.
- `FieldC > 100`
  Über eine Bedingung dieser Form werden alle Datensätze selektiert, die in der Spalte "FieldC" einen Wert größer 100 besitzen. FieldC ist vom Datentyp DB_TYP_INTEGER.
- `BETWEEN FieldC = 20 AND FieldC = 200`
  Das Statement selektiert alle Datensätze, deren Wert in der Spalte "FieldC" zwischen 20 und 200 liegt. FieldC ist vom Datentyp DB_TYP_INTEGER.
- `FieldD`
  Damit wird nach Spalte "FieldD" sortiert.
- `FieldE desc`
  Damit wird nach Spalte "FieldE" in umgekehrter alphabetischer Reihenfolge (descending order) sortiert.

Siehe auch

SQL-Schlüsselwörter (Seite 1219)

6.2.1.7 SQL-Schlüsselwörter

SQL-Schlüsselwörter

Hinweis

Verwenden Sie nachfolgende Begriffe und Schlüsselwörter der SQL-Sprache nicht als Name einer Rezeptur, Rezepturabfrage oder eines Rezepturelements:

- "Archive"
- "View"
- "Field"
- "ViewCol"

Schlüsselwörter der SQL-Sprache			
add	all	alter	and
any	as	asc	begin
between	binary	break	by
call	cascade	cast	char
char_convert	character	check	checkpoint
close	comment	commit	connect
constraint	continue	convert	create

Schlüsselwörter der SQL-Sprache			
cross	current	cursor	date
dba	dbspace	deallocate	dec
decimal	declare	default	delete
desc	distinct	do	double
drop	else	elseif	encrypted
end	endif	escape	exception
exec	execute	exists	fetch
first	float	for	foreign
from	full	goto	grant
group	having	holdlock	identified
if	in	index	inner
inout	insert	instead	int
integer	into	is	isolation
join	key	left	like
lock	long	match	membership
message	mode	modify	named
natural	noholdlock	not	null
numeric	of	off	on
open	option	options	or
order	others	out	outer
passthrough	precision	prepare	primary
print	privileges	proc	procedure
raiserror	readtext	real	reference
references	release	remote	rename
resource	restrict	return	revoke
right	rollback	save	savepoint
schedule	select	set	share
smallint	some	sqlcode	sqlstate
start	stop	subtrans	subtransaction
synchronize	syntax_error	table	temporary
then	time	tinyint	to
tran	trigger	truncate	tsequal
union	unique	unknown	update
user	using	validate	values
varbinary	varchar	variable	varying
view	when	where	while
with	work	writetext	

Siehe auch

SQL-Sprache (Seite 1218)

6.2.2 Rezepturen projektieren

6.2.2.1 Allgemeine Vorgehensweise beim Projektieren

Projektierungsschritte

Um eine Rezeptur anzulegen, sind folgende Projektierungsschritte notwendig:

Schritt	Aufgabe
1	Legen Sie die Struktur der Rezeptur fest.
2	Legen Sie die Kommunikationsart der Rezeptur fest.
3	Legen Sie abhängig von der Kommunikationsart die Steuervariablen oder die Rohdatenvariable an.
4	Legen Sie die Rezeptur an und legen Sie die Eigenschaften der Rezeptur fest.
5	Legen Sie die Rezepturelemente entsprechend der geplanten Struktur an.
6	Legen Sie die Rezepturdatensätze an.
7	Projektieren Sie bei Bedarf Sichten auf mehrere Rezepturen.
8	Projektieren Sie ein Bild mit einer Rezepturanzeige und verbinden Sie die Rezepturanzeige mit der Rezeptur oder der Sicht. Oder: Projektieren Sie ein Bild mit E/A-Feldern und Schaltflächen. Projektieren Sie an die Schaltflächen die Funktionen zum Lesen und Schreiben von Rezepturdatensatzwerten.

Hinweis

Runtime API

In der Runtime API stehen Ihnen umfangreiche uaArchive-Funktionen zur Verfügung, um Rezepturen und Rezepturabfragen zu bearbeiten (siehe Kapitel "Schnittstellen > Runtime API", Abschnitt "AUTOHOTSPOT").

6.2.2.2 Neue Rezeptur anlegen

Einleitung

Im Editor "Rezepturen" auf der Registerkarte "Rezepturen" legen Sie neue Rezepturen an.

Festlegung des Rezepturnamens

Verwenden Sie für den Rezepturnamen keine Schlüsselwörter oder reservierte Wörter der Datenbanksprache SQL (siehe Abschnitt "SQL-Schlüsselwörter (Seite 1219)").

Rezepturen mit Datums- oder Zeitangaben

Wenn Sie Datums- oder Zeitangaben verwenden, achten Sie darauf, dass die System-Einstellung für Uhrzeit und Datum auf dem Projektierungsrechner mit der auf dem Zielsystem übereinstimmt. Beispiel: Sie laden um 13:55 einen Rezepturdatensatz auf das Zielsystem, in dem die Bearbeitungszeit 14 Uhr gespeichert ist. Wenn es auf dem Zielrechner bereits 14:05 ist, wird die Rezeptur nicht mehr bearbeitet. Wenn ein Bediener die Rezeptur bearbeitet, werden Änderungsinformationen nicht korrekt in die Datenbank zurückgeschrieben.

Überprüfen Sie nach dem Laden auf das Zielsystem Rezepturen mit Datums- oder Zeitangaben auf dem Zielsystem.

Voraussetzung

- Die Steuervariablen oder die Rohdatenvariable sind angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Vorgehen

Um eine Rezeptur anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Editor "Rezepturen" auf der Registerkarte "Rezepturen" auf die erste freie Tabellenzeile.
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Einstellungen > Name" einen aussagekräftigen Namen für die Rezeptur ein.
   Der Name identifiziert die Rezeptur innerhalb des Projekts eindeutig.

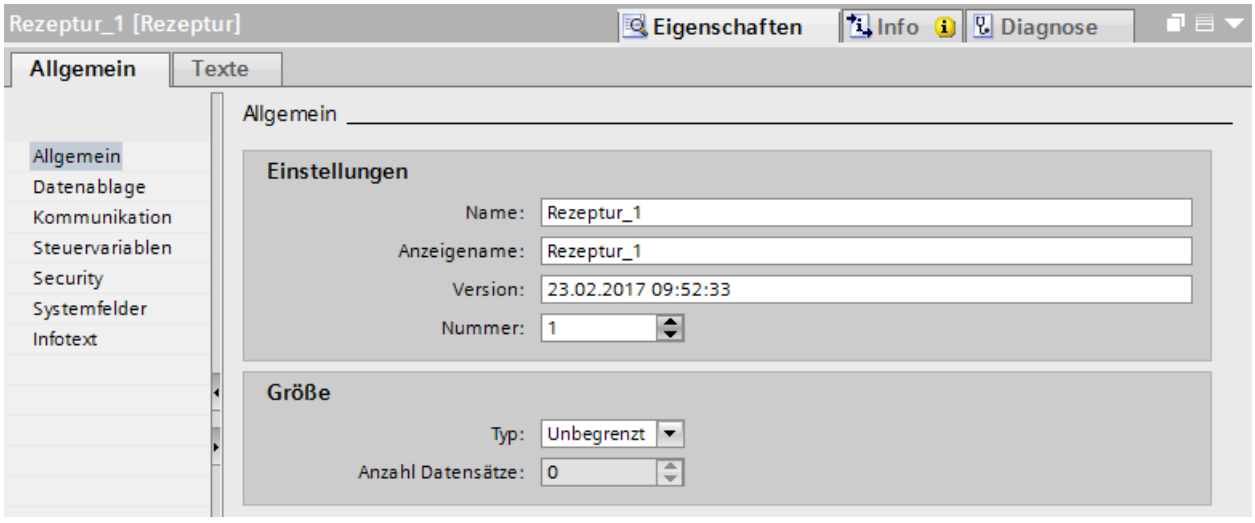

3. Geben Sie unter "Anzeigename" den sprachabhängigen Namen ein, der in Runtime angezeigt wird.
4. Legen Sie im Bereich "Größe" unter "Typ" fest, ob die Rezepturdatensätze auf eine vorgegebene maximale Anzahl begrenzt sind oder nicht.
5. Wenn der Größentyp begrenzt ist, geben Sie unter "Anzahl Datensätze" die maximale Anzahl der Rezepturdatensätze einer Rezeptur vor.

6. Legen Sie unter "Eigenschaften > Kommunikation" die "Kommunikationsart" fest:
   - Wenn Sie "Rohdatenvariable" gewählt haben, legen Sie zusätzlich eine Rohdatenvariable an und wählen sie unter "Rohdatenvariable" aus.
   - Wenn Sie die Rezeptur nur zum Speichern von Projektdaten verwenden wollen, wählen Sie "Keine Kommunikation".

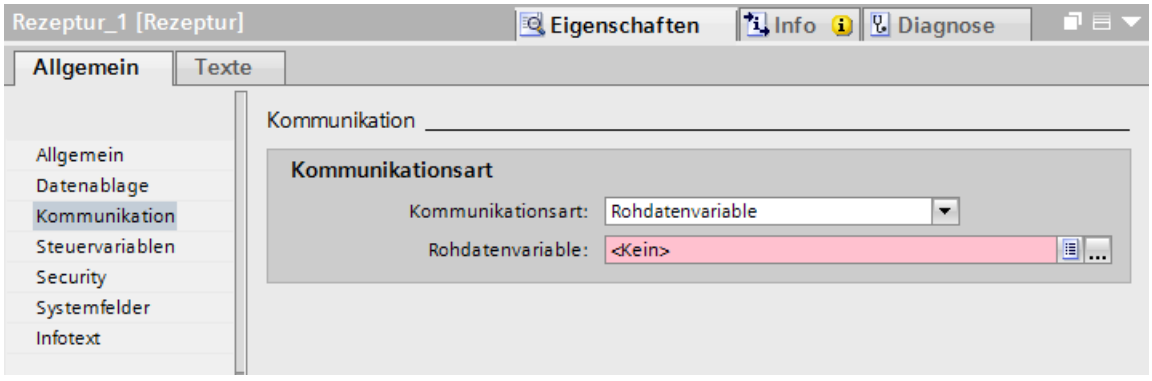

7. Wenn Sie als Kommunikationsart "Variablen" gewählt haben, legen Sie unter "Eigenschaften > Steuervariablen" die Steuervariablen für den Zugriff auf die Rezeptur fest.

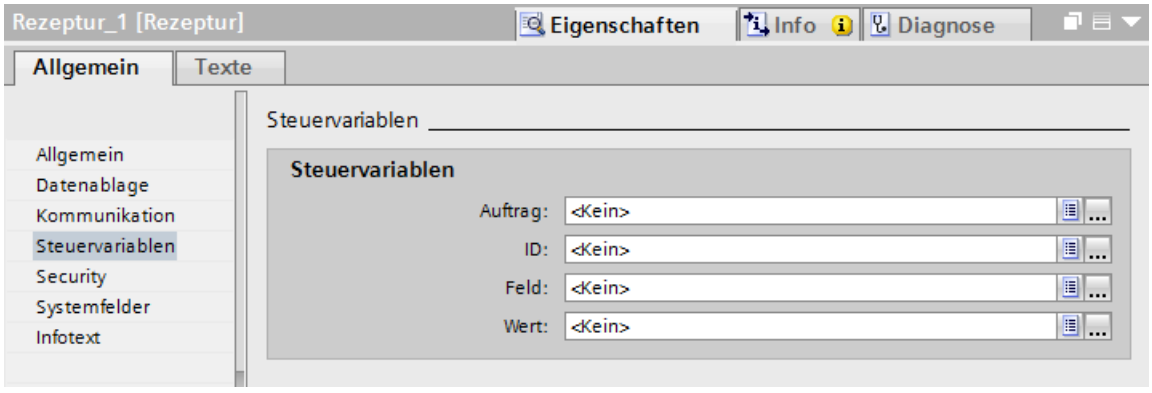

8. Legen Sie unter "Eigenschaften > Security" fest, welche Bedienberechtigungen lesenden und schreibenden Zugriff in Runtime auf die Rezeptur besitzen.

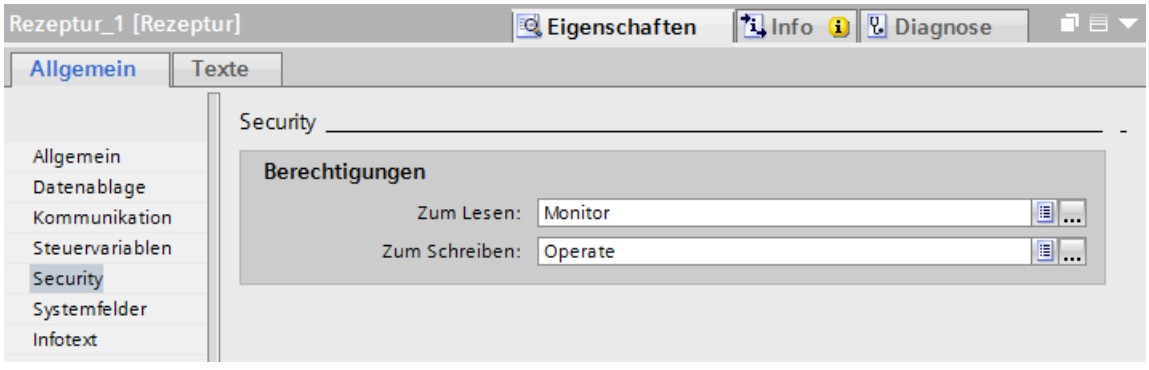

9. Legen Sie unter "Systemfelder" fest, ob die Einträge "Letzter Zugriff" und "Letzter Benutzer" in Runtime angezeigt werden.

Ergebnis

Die Rezeptur ist angelegt.

Siehe auch

SQL-Schlüsselwörter (Seite 1219)

6.2.2.3 Rezepturelemente und Datensätze anlegen

Einleitung

Im unteren Bereich des Editors "Rezepturen" legen Sie auf der Registerkarte "Rezepturelemente" Rezepturelemente für die selektierte Rezeptur an. Außerdem können Sie auf der Registerkarte "Rezepturdatensätze" Werte für die Rezepturelemente eingeben.

Voraussetzung

- Die Variablen für die Rezeptur sind angelegt.
- Die Rezeptur ist angelegt.
- Die Rezeptur ist im Editor "Rezepturen" selektiert.

Vorgehen

Um ein Rezepturelement anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf der Registerkarte "Rezepturelemente" auf die erste freie Tabellenzeile.
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Einstellungen > Name" einen aussagekräftigen Namen für das Rezepturelement ein. Innerhalb einer Rezeptur muss dieser Name eindeutig sein.
3. Geben Sie unter "Anzeigename" den sprachabhängigen Namen ein, der in Runtime erscheint.

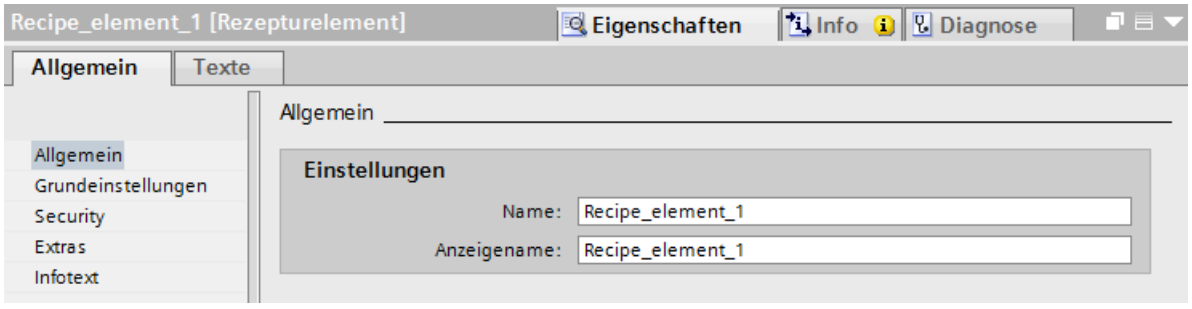

4. Wenn die Rezeptur die Kommunikationsart "Variablen" besitzt:
   Um das Rezepturelement mit einer Variablen zu verbinden, wählen Sie unter "Eigenschaften > Grundeinstellungen > Einstellungen" eine Variable aus. Darunter wird der Datentyp angezeigt.

5. Beim Datentyp "Zeichenkette" geben Sie zusätzlich unter "Datenlänge" die maximale Länge der Zeichenkette an. Der Bereich "Einstellungen" rechts zeigt den Wertebereich zahlenbasierten Datentypen, bei denen Sie zusätzlich einen Standardwert festlegen können.

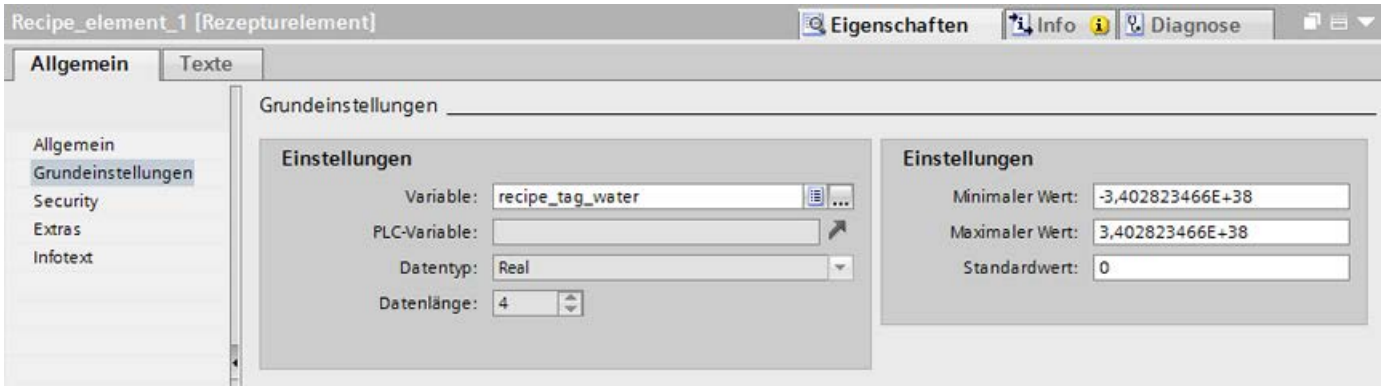

6. Wenn die Rezeptur die Kommunikationsart "Rohdatenvariable" oder "keine Kommunikation" besitzt: Wählen Sie den "Datentyp" und beim Datentyp "Zeichenkette" zusätzlich die "Datenlänge" des Rezepturelements.
7. Legen Sie unter "Eigenschaften > Security" fest, welche Berechtigungen in Runtime lesenden und schreibenden Zugriff auf das Rezepturelement besitzen.
8. Legen Sie unter "Eigenschaften > Extras" fest, welche Bedingungen die Werte des Rezepturelements erfüllen müssen. Zur schnelleren Suche können die Werte indiziert werden:

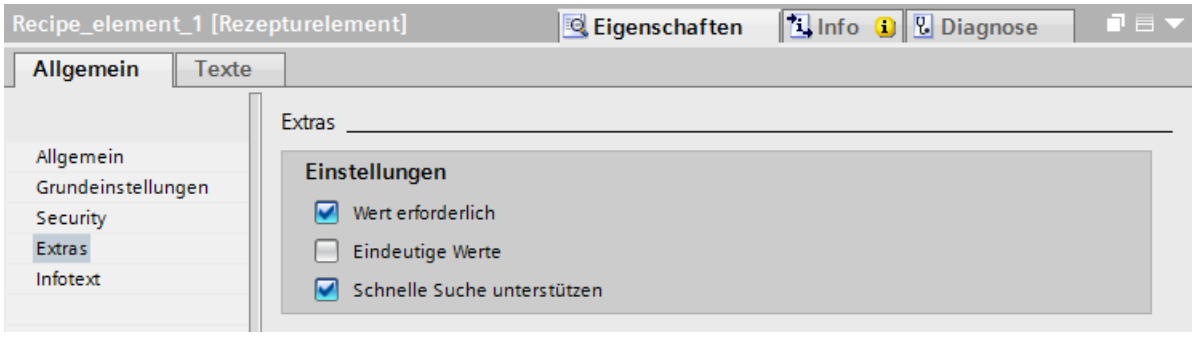

9. Um einen Rezepturdatensatz anzulegen, doppelklicken Sie auf der Registerkarte "Rezepturdatensätze" auf die erste freie Tabellenzeile.
10. Geben Sie die gewünschten Wert für das Rezepturelement im Tabelleneditor ein.

Ergebnis

Das Rezepturelement und ein entsprechender Rezepturdatensatz mit einem Wert sind angelegt.

6.2.2.4 Rezepturen in die Projektierung importieren und wieder exportieren

Einleitung

Sie können Rezepturen als CSV-Dateien exportieren und wieder in die Projektierung importieren.

Hinweis

Datensätze von Rezepturabfragen können nicht importiert oder exportiert werden.

Anwendungsfall

Sie exportieren Rezepturdaten, um die Rezepturdaten langfristig auf einem Rechner abzulegen und zu sichern.

Um Rezepturdaten zu vereinheitlichen und zu verteilen, exportieren Sie die Rezepturdaten auf einem Bediengerät. Sie ändern die CSV-Datei in Microsoft Excel und importieren die CSV-Datei in alle Bediengeräte, die die gleichen Rezepturdaten benötigen.

Sie tauschen Rezepturdaten zwischen verschiedenen Projekten aus. Sie ändern Rezepturdaten in Runtime. Um die geänderten Rezepturdaten nach WinCC ES zu übertragen, exportieren Sie die Rezepturdatensätze in Runtime und kopieren die CSV-Datei auf den Projektierungsrechner. Dort importieren Sie die CSV-Datei mit den Rezepturdatensätzen in die Rezeptur. Und in umgekehrter Richtung: in WinCC ES geänderte Rezepturdatensätze übertragen Sie in die Runtime auf dem Bediengerät.

Voraussetzung

- Die Rezeptur ist angelegt.
- Die Struktur der exportierten CSV-Datei und der Rezeptur im WinCC ES stimmen überein: Name, Anzahl und Datentyp der Rezepturelemente sind jeweils gleich.
- Der Editor "Rezepturen" ist geöffnet.

CSV-Datei importieren

Um Rezepturdatensätze in eine Rezeptur zu importieren, gehen Sie wie folgt vor:

1. Markieren Sie in der Registerkarte "Rezepturen" die Zeile mit der gewünschten Rezeptur.
2. Wählen Sie im Kontextmenü den Befehl "Rezepturdaten importieren".
   Der Dialog "Import" wird geöffnet.

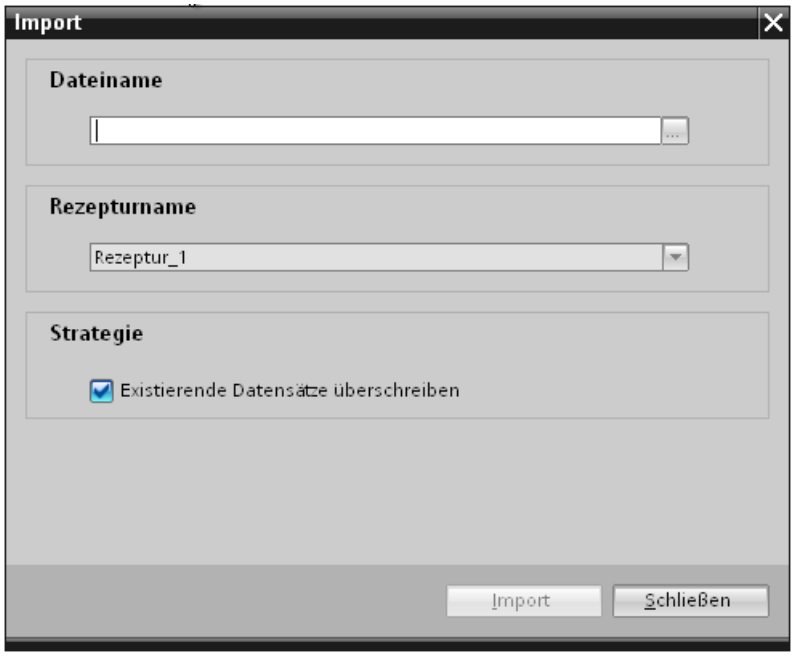

3. Wählen Sie unter "Dateiname" die gewünschte CSV-Datei mit den Rezepturdaten.
4. Legen Sie unter "Strategie" fest, ob ein Rezepturdatensatz in der CSV-Datei einen in WinCC mit der gleichen Rezepturdatensatznummer bereits vorhandenen Rezepturdatensatz überschreibt.
5. Starten Sie den Vorgang mit der Schaltfläche "Import".

Ergebnis

Die Rezeptur wird um nicht vorhandene Rezepturdatensätze aus der CSV-Datei ergänzt. Falls gewählt, werden vorhandene Rezepturdatensätze überschrieben.

Wenn die Struktur der Rezeptur nicht mit der Struktur der CSV-Datei übereinstimmt, wird im Inspektorfenster unter "Info" eine Fehlermeldung ausgegeben.

CSV-Datei exportieren

1. Markieren Sie in der Registerkarte "Rezepturen" die Zeile mit der Rezeptur, die Sie exportieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Rezepturdaten exportieren".
   Der Dialog "Export" wird geöffnet.

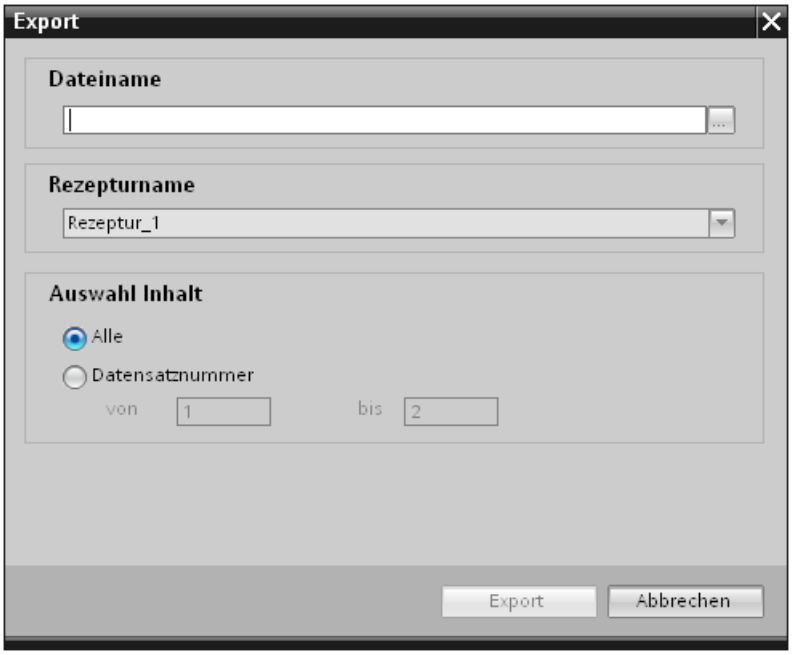

3. Wählen Sie unter "Dateiname" einen Pfad für die CSV-Datei.
4. Wählen Sie unter "Rezepturname" die gewünschten Rezeptur.
5. Wählen Sie unter "Auswahl Inhalt" alle Rezepturdatensätze oder schränken Sie die Auswahl auf bestimmte Rezepturdatensatznummern ein.
6. Starten Sie den Vorgang mit der Schaltfläche "Export".

Ergebnis

Die Rezeptur wird als CSV-Datei exportiert. Listentrennzeichen ist das Semikolon "Rezepturelement_1;Rezepturelement_2". Dezimaltrennzeichen ist der Punkt z. B. "3.14".

6.2.2.5 Vorgehen bei Änderung der Rezepturstruktur

Einleitung

Unterschiedliche Rezepturstrukturen können z. B. in folgenden Fällen auftreten:

- Bei Änderungen während der Inbetriebnahme
- Beim Bearbeiten einer Maschine durch den Maschinenbauer (Retro-Fit)
- Beim Import von CSV-Dateien kann sich die Struktur der CSV-Datei von der Struktur der Rezeptur unterscheiden.

Sie können bereits erstellte Rezepturdatensätze trotzdem weiter verwenden.

ACHTUNG
Beim Ändern des Variablennamens geht die Zuordnung verloren.

Auswirkungen

Wenn sich die Struktur der CSV-Datei von der Struktur der Rezeptur unterscheidet, werden Abweichungen wie folgt behandelt:

- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei zusätzliche Werte enthält, werden diese Werte verworfen.
- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei Werte vom falschen Datentyp enthält, wird im Rezepturdatensatz der projektierte Standardwert verwendet.
  Beispiel: Der Rezepturdatensatz enthält Werte, die den Tankinhalt angeben und als Fließkommazahlen eingegeben wurden. Die entsprechende Rezepturvariable erwartet jedoch einen ganzzahligen Wert. In diesem Fall wird der übertragene Wert verworfen und der projektierte Standardwert verwendet.
- Wenn der alte Rezepturdatensatz bzw. die CSV-Datei zu wenige Werte enthält, wird im Rezepturdatensatz ebenfalls der projektierte Standardwert verwendet.

6.2.2.6 Rezepturabfrage anlegen

Einleitung

Eine Rezepturabfrage ist ein Satz von Rezepturelementen aus unterschiedlichen Rezepturen. Beispiel: Rezepturabfrage "Wasserverbrauch":

- Wasser aus Rezeptur "Fruchtsaftgetränk"
- Wasser aus Rezeptur "Saft"
- Wasser aus Rezeptur "Nektar"

Rezepturelemente können Sie verknüpfen zu Relationen z. B. "Nektar.Wasser > Fruchtsaftgetränk.Wasser". In der Rezepturabfrage in Runtime werden dann nur solche Datensätze angezeigt, welche die Relation erfüllen, im Beispiel solche Datensätze, bei denen der Wasserbedarf der Rezeptrur "Nektar" größer ist als bei "Fruchtsaftgetränk".

Im Editor "Rezepturen" unter der Registerkarte "Rezepturabfragen" stellen Sie Rezepturabfragen aus Rezepturelementen zusammen. Aus den Rezepturen wählen Sie die Rezepturelemente, die in die Rezepturabfrage aufgenommen werden, und setzen Sie in Relation. Die Relation drückt eine Bedingung aus. In der Rezepturanzeige werden nur solche Kombinationen von Rezepturelementen angezeigt, welche der Bedingung genügen (wahr).

Die Daten einer Rezepturabfragen können Sie auch in Runtime editieren. Die geänderten Daten werden in die Ursprungsrezeptur in Runtime übernommen.

Folgendes ist innerhalb einer Rezepturabfragen nicht möglich:

- Anlegen von neuen Rezepturdaten
- Löschen von Rezepturdaten

Voraussetzung

- Mindestens zwei Rezepturen sind angelegt.
- Die Rezepturen befinden sich in der gleichen Datenbank auf dem gleichen Bediengerät.
- Die Rezepturen haben mindestens ein gleiches Rezepturelement im gleichen Datenformat.

Rezepturabfrage anlegen

Um eine Rezepturabfrage anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Editor "Rezepturen" in der Registerkarte "Rezepturabfragen" auf die erste freie Tabellenzeile.
   Eine neue Rezepturabfrage wird angelegt.
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Name" einen aussagekräftigen Namen für die Rezepturabfrage ein.

3. Geben Sie unter "Anzeigename" den sprachabhängigen Namen ein, der in Runtime erscheint.

Rezepturabfrage-Elemente und Relationen anlegen

Um ein neues Element für die Rezepturabfrage und die Relation anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Registerkarte "Rezepturabfrage-Elemente" auf die erste freie Tabellenzeile.
2. Wählen Sie unter "Eigenschaften > Allgemein > Rezepturelement" das Rezepturelement, das Sie in die Abfrage einfügen wollen.
3. Legen Sie bei Bedarf weitere Rezepturabfrage-Elemente an.
4. Selektieren Sie im Register "Rezepturabfragen" die Rezepturabfrage.

5. Wählen Sie im Inspektorfenster unter "Eigenschaften > Relation > Beziehung" als linker und rechter Operand ein Rezepturabfrage-Element und den gewünschten Operator.

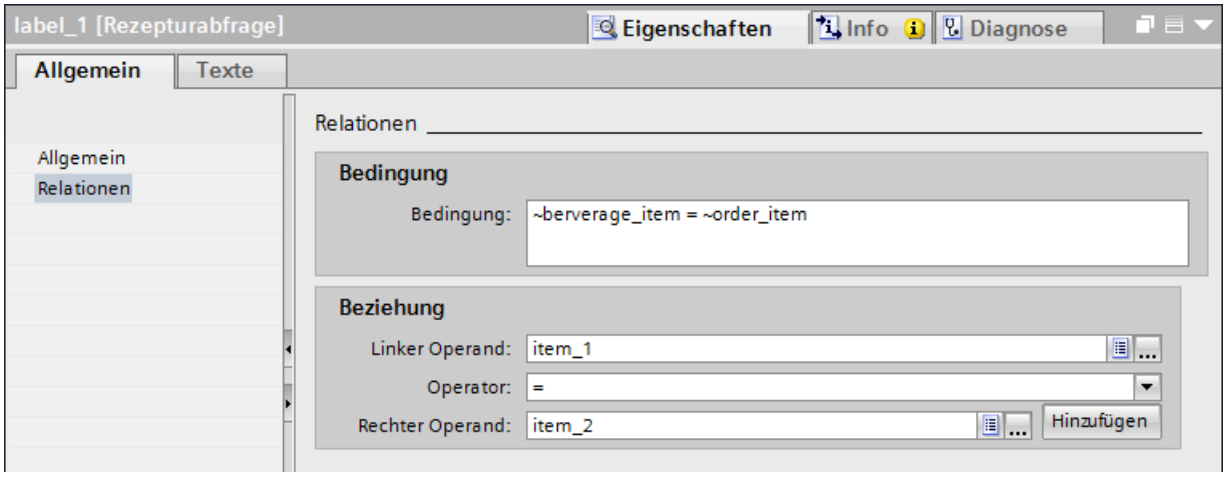

6. Klicken Sie auf die Schatlfläche "Hinzufügen".
   Die Beziehung erscheint unter "Bedingung". Wenn dort bereits eine Bedingung definiert wurde, wird die neue Beziehung mit dem logischen Operator "AND" hinzugefügt. Die Bedingung können Sie editieren.
7. Verändern Sie bei Bedarf die Reihenfolge der Rezepturabfrage-Elemente mit Drag&Drop.

Ergebnis

Die Rezepturabfrage ist angelegt und eine Bedingung definiert. Nur solche Kombinationen von Rezepturelementen, die der Bedingung genügen, werden angezeigt.

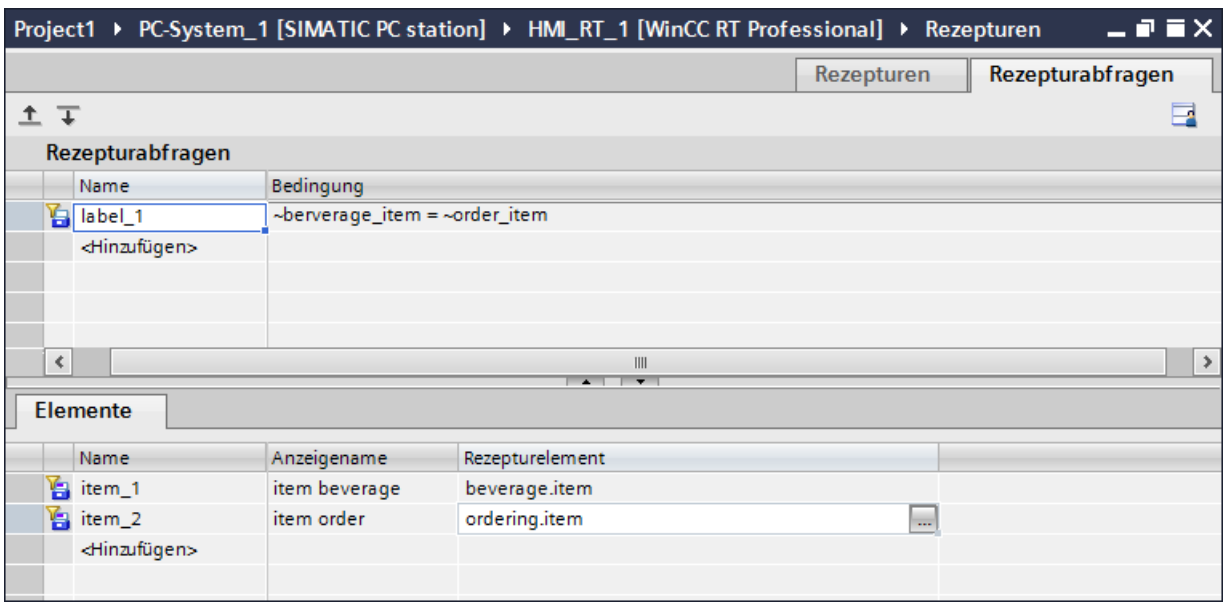

6.2.3 Anzeige von Rezepturen projektieren

6.2.3.1 Anzeige von Rezepturen

Einleitung

Rezepturen und Rezepturabfragen stellen Sie am Bediengerät folgendermaßen dar:

- In einer Rezepturanzeige
- In einzelnen E/A-Feldern

Rezepturanzeige

Die Rezepturanzeige dient der tabellarischen Darstellung der Rezepturen und Abfragen. Jeder Rezepturdatensatz belegt eine Zeile.

Das folgende Bild zeigt ein Beispiel für eine Rezepturanzeige.

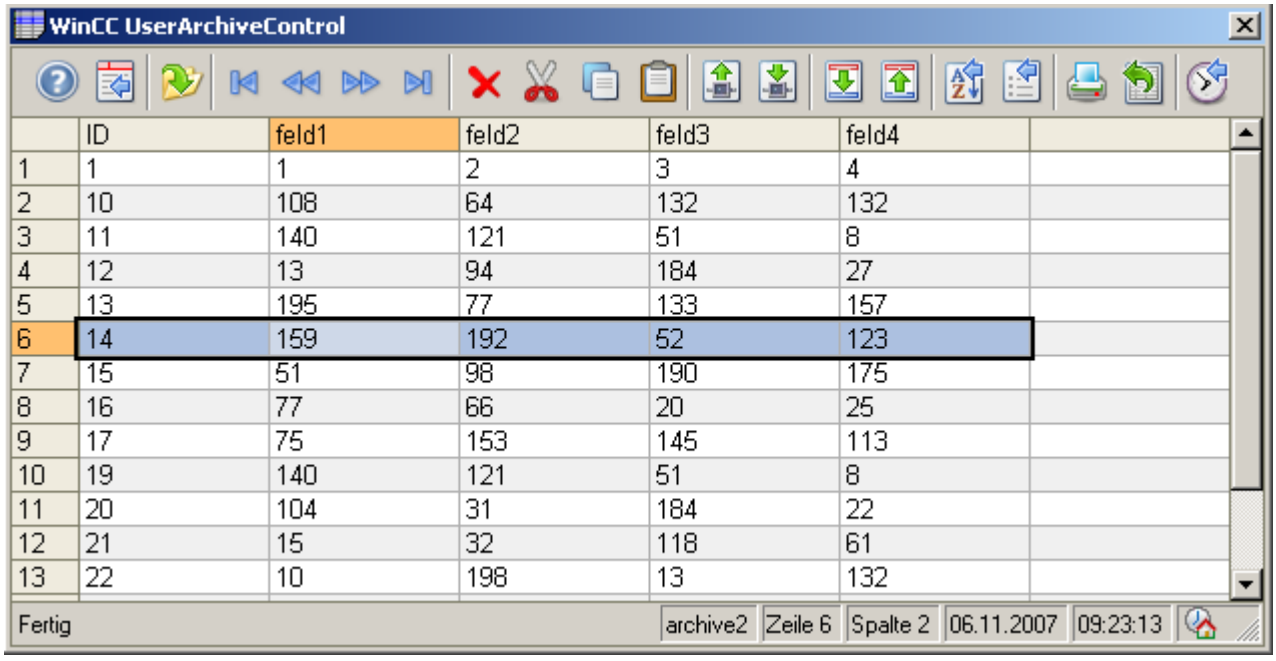

	ID	feld1	feld2	feld3	feld4
1	1	1	2	3	4
2	10	108	64	132	132
3	11	140	121	51	8
4	12	13	94	184	27
5	13	195	77	133	157
6	14	159	192	52	123
7	15	51	98	190	175
8	16	77	66	20	25
9	17	75	153	145	113
10	19	140	121	51	8
11	20	104	31	184	22
12	21	15	32	118	61
13	22	10	198	13	132

Der Bediener bedient die Rezepturanzeige in Runtime und legt z. B. neue Datensätze an oder importiert Rezepturen. Mit zwei Schaltflächen in der Rezepturanzeige übertragen Sie einen Rezepturdatensatz in die Rezepturvariablen und umgekehrt.

Hinweis

Bei der Projektierung einer Rezepturanzeige wählen Sie eine Rezeptur oder Rezepturabfrage aus. Die Auswahl kann der Bediener in Runtime ändern wie auch weitere Eigenschaften.

Rezeptur im Bild

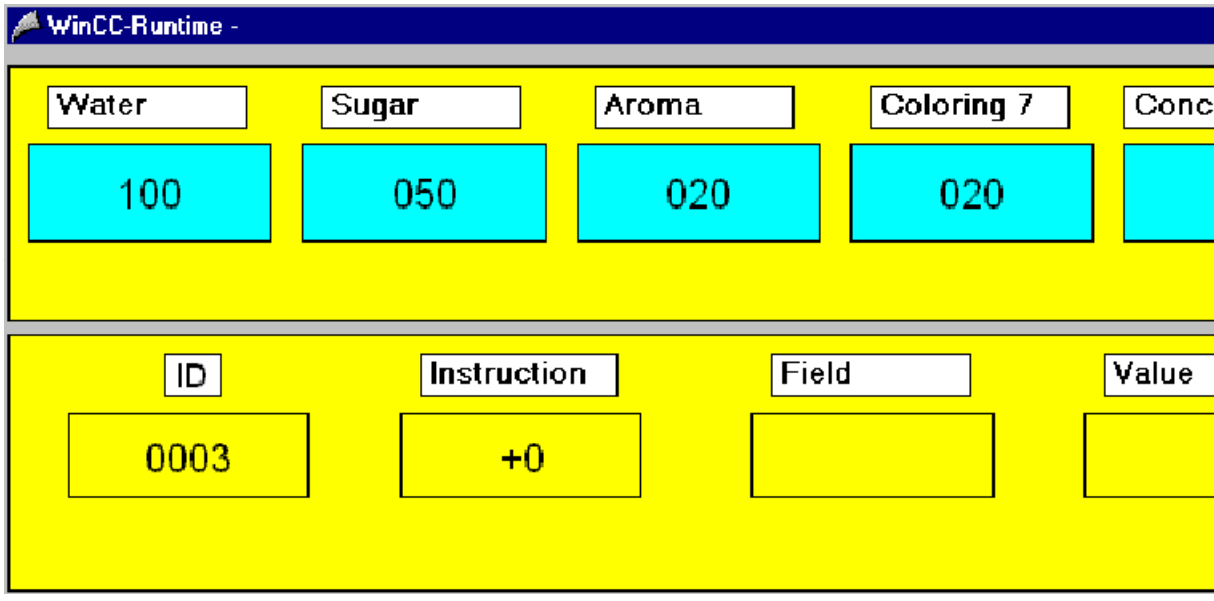

Die blauen Felder oben zeigen die Werte der Rezepturvariablen, die eine Kommunikationsverbindung zur Steuerung besitzen. Die gelben Felder unten zeigen den Inhalt der vier Steuervariablen. Im Beispiel wurden die blauen Rezepturwerte in den Rezepturdatensatz Nummer 3 geschrieben (Auftrag = 6) und die Anzahl der Fehler = 0 zurückgegeben.

Siehe auch

Steuervariablen (Seite 1243)

6.2.3.2 Eigenschaften der Rezepturanzeige

Einleitung

Die Rezepturanzeige zeigt Rezepturelemente in einem Bild in Tabellenform an. Der Bediener kann in Runtime in der Rezepturanzeige Rezepturdaten bearbeiten.

Konfiguration der Rezepturanzeige

Für die Rezepturanzeige haben Sie folgende Konfigurationsmöglichkeiten:

- Allgemein > Anzeige: Sie wählen als Datenquelle eine Rezeptur oder Rezepturabfrage und legen fest, ob Rezepturdatensätze bearbeitet, eingefügt oder gelöscht werden können..
- Allgemein > Druckauftrag
  Die Rezepturanzeige ist automatisch mit dem Systemdruckauftrag für Rezepturanzeigen verknüpft. Wenn Sie einen eigenen Druckauftrag oder einen angepassten Systemdruckauftrag verwenden wollen, verknüpfen Sie diesen mit der Rezepturanzeige.
- Allgemein > Zeit
  Sie legen die Zeitbasis für die Anzeige "Letzter Zugriff" fest, z. B. koordinierte Weltzeit. Sie können die gewählte Zeitbasis in Runtime mit ausgeben lassen.

- Gestaltung, Darstellung, Textformat
  Sie legen Farben und Design, Größe und Position der Rezepturanzeige, Schriftart und Schriftgröße fest.
- Fenster
  Sie legen die Schriftgröße, Überschrift des Fensters der Rezepturanzeige und weitere Fenster-Eigenschaften fest.
- Tabelle
  Sie legen das Erscheinungsbild der Spaltenüberschriften, Tabelle und Begrenzungslinien fest und das Verhalten beim Markieren und Sortieren von Datensätzen.
- Tabelle > Markierung
  Legen Sie fest, ob Zeilen oder nur Zellen mit der Maus markierbar sind.
  Projektieren Sie die Eigenschaften des Auswahlrechtecks, das um die markierten Zellen bzw. Zeilen der Tabelle dargestellt werden kann.
  Projektieren Sie bei Bedarf die Markierungsfarben für die markierbaren Zellen und/oder Zeilen. Mit der Eigenschaft "Automatische Farbgebung" wird für die Markierung die vom System festgelegte Farbe verwendet.
- Tabelle > Sortierung
  Bestimmen Sie, ob sortiert wird und wenn ja, mit Klick oder Doppelklick auf die Spaltenüberschrift. Legen Sie Reihenfolge der Sortierung fest, die mit jedem Klick oder Doppelklick auf die Spaltenüberschrift zyklisch durchlaufen wird: entweder Sortierung aufwärts - abwärts oder aufwärts - abwärts - keine Sortierung.
  Projektieren Sie Sortiersymbol und Sortierindex, die rechtsbündig in der Spaltenüberschrift dargestellt werden. Sie zeigen Sortierabfolge und Sortierreihenfolge der Spalten an.
  Wenn Sie die Option "Sortiertaste verwenden" aktivieren, wird über dem vertikalem Rollbalken das Sortiersymbol als Sortiertaste angezeigt. Über die Sortiertaste sortieren Sie mit einem Mausklick die ausgewählte Spalte in der projektierten Sortierreihenfolge. Wenn in der Tabelle kein vertikaler Rollbalken vorhanden ist, wird die Sortiertaste nicht angezeigt.
- Spalten
  Sie legen fest, welche Rezepturelemente in Runtime in welcher Reihenfolge angezeigt werden. Sie versehen bestimmte Elemente mit einem Schreibschutz und legen die Breite und Formatierung fest.
  Einige Spalten können den Inhalt und die Überschrift auch als Symbol darstellen. Bestimmen Sie im Feld "Darstellung", wie diese Spalten dargestellt werden. Text und Symbol können gemeinsam angezeigt werden.
- Filter
  Als Filterkriterium legen Sie eine Bedingung für ein Rezepturelement fest und verknüpfen mehrere Bedingungen. Nur wenn alle Filterkriterien erfüllt sind, wird der Datensatz angezeigt.

- Symbolleiste
  Sie legen neben der Erscheinung der Symbolleiste fest, welche Schaltflächen in Runtime in welcher Reihenfolge sichtbar und mit welcher Bedienberechtigung bedienbar sind.
  Aktivieren Sie in der Liste die Schaltflächen, die Sie zur Bedienung des WinCC-Controls in Runtime benötigen. Informationen zu der Funktion der einzelnen Schaltflächen finden Sie im Abschnitt "Bedienung in Runtime".
  Legen Sie bei Bedarf für eine Schaltfläche einen Hotkey fest.
  Wenn Sie Bedienberechtigungen den einzelnen Schaltflächen zuweisen, wird in Runtime die Schaltfläche nur für die Benutzer freigegeben, die die entsprechende Bedienberechtigung haben.
  Wenn Sie die Option "Aktiv" für eine Schaltfläche deaktivieren, wird die aktivierte Schaltfläche in Runtime angezeigt, ist aber nicht bedienbar.
- Statuszeile
  Sie legen neben der Erscheinung der Statuszeile fest, welche Informationen in Runtime in der Statuszeile angezeigt werden z. B. die Rezepturdatensatznummer.
- Datenexport
  Ein Standard-Dateiname und ein Standard-Verzeichnis sind bereits eingetragen durch Schaltervariablen. Sie können jedoch eigene Dateinamen und Verzeichnisnamen vergeben.
  Dabei können Sie folgende Schaltervariablen nutzen:
  @OBJECTNAME: Objektname der Rezepturanzeige, siehe unter "Eigenschaften > Verschiedenes"
  @PROJECTPATH: Projektverzeichnis
  @CURRENTDATE: aktuelles Datum
  @CURRENTTIME: aktuelle Zeit
- Security
  Sie legen fest, ob und wie Änderungen an den Eigenschaften der Rezepturanzeige in Runtime und nach einem Bildwechsel gespeichert werden und welche Berechtigung zum Ändern erforderlich ist.
- Verschiedenes
  Sie legen für die Rezepturanzeige die Sichtbarkeit und Ebene fest.

6.2.3.3 Rezepturanzeige projektieren

Voraussetzung

- Eine Rezeptur oder eine Rezepturabfrage ist angelegt.
- Ein Bild ist angelegt und in Editor "Bilder" geöffnet.

Vorgehen

Um eine Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Ziehen Sie das Objekt "Rezepturanzeige" aus der Palette "Controls" der Task Card "Werkzeuge" in das Bild.
2. Wählen Sie im Inspektorfenster unter "Allgemein > Anzeige" die gewünschte Rezeptur oder Rezepturabfrage.

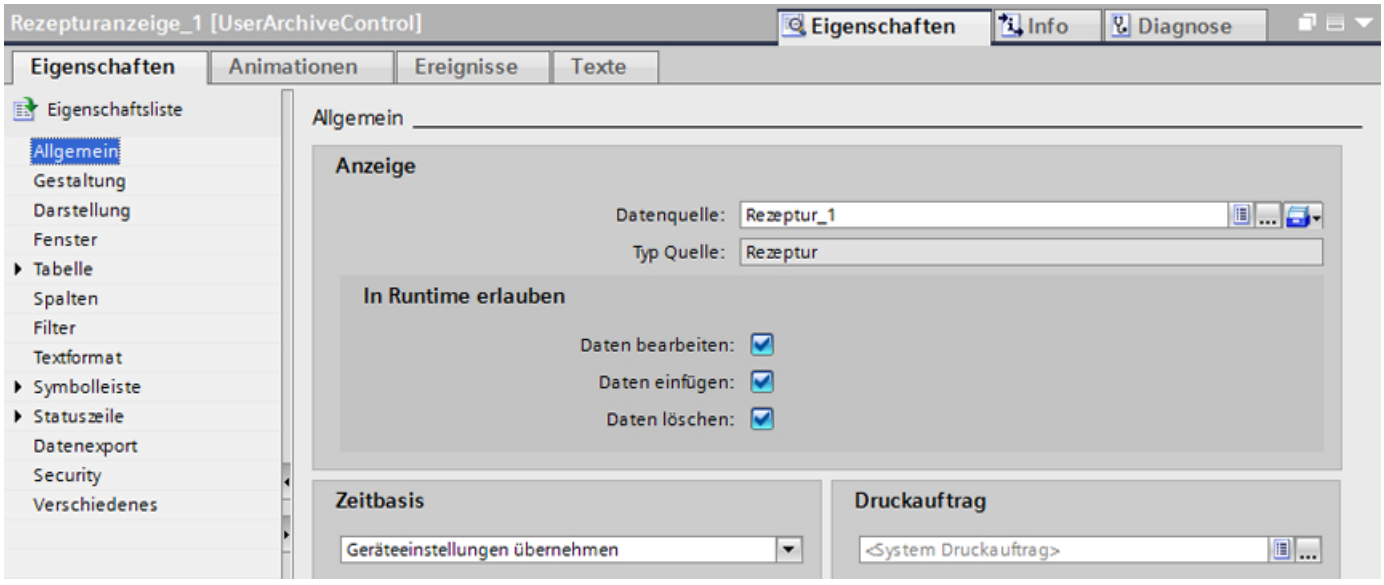

3. Legen Sie unter "Allgemein > In Runtime erlauben" die Schreib- und Leserechte für die Rezepturanzeige in Runtime fest.
4. Legen Sie unter "Gestaltung", "Darstellung" und "Textformat" zusätzliche Anzeigeoptionen der Rezepturanzeige fest.

5. Legen Sie unter "Eigenschaften > Spalten" fest, welche Rezepturelement in Runtime sichtbar und schreibgeschützt sind.

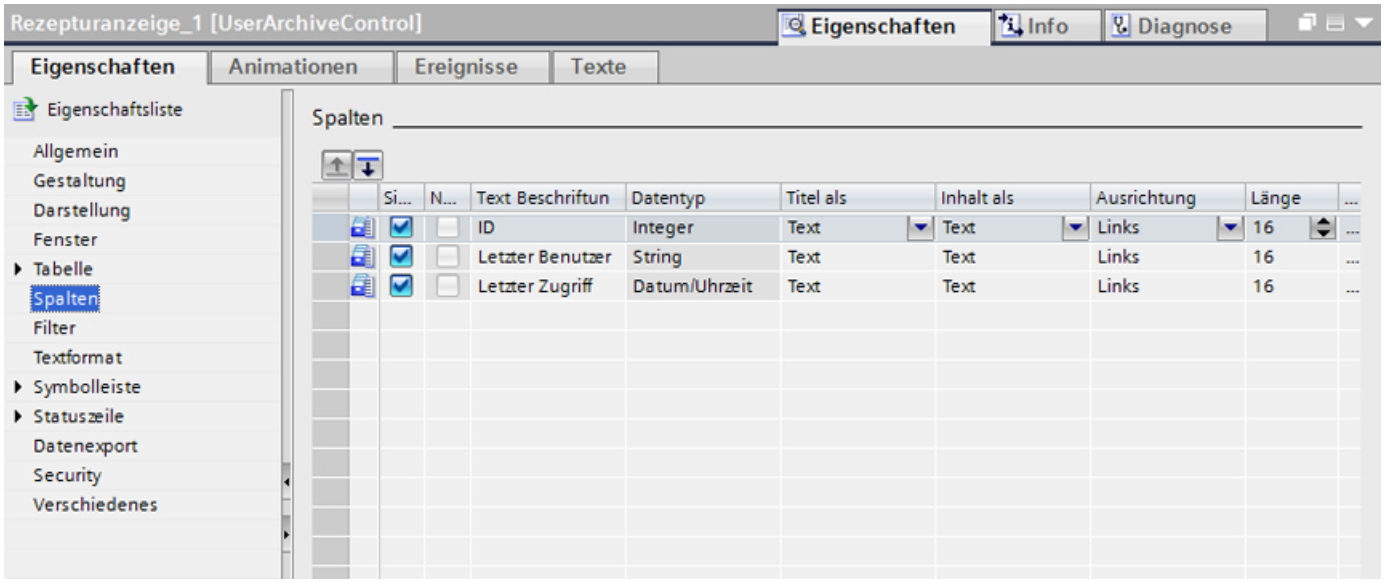

Hinweis

Einige Spalten können den Inhalt auch als Symbol darstellen. Bestimmen Sie unter "Inhalt als", wie diese Spalten dargestellt werden. Text und Symbol können auch gemeinsam angezeigt werden.

6. Legen Sie unter "Symbolleiste" fest, welche Schaltflächen in Runtime in der Rezepturanzeige verfügbar sind.

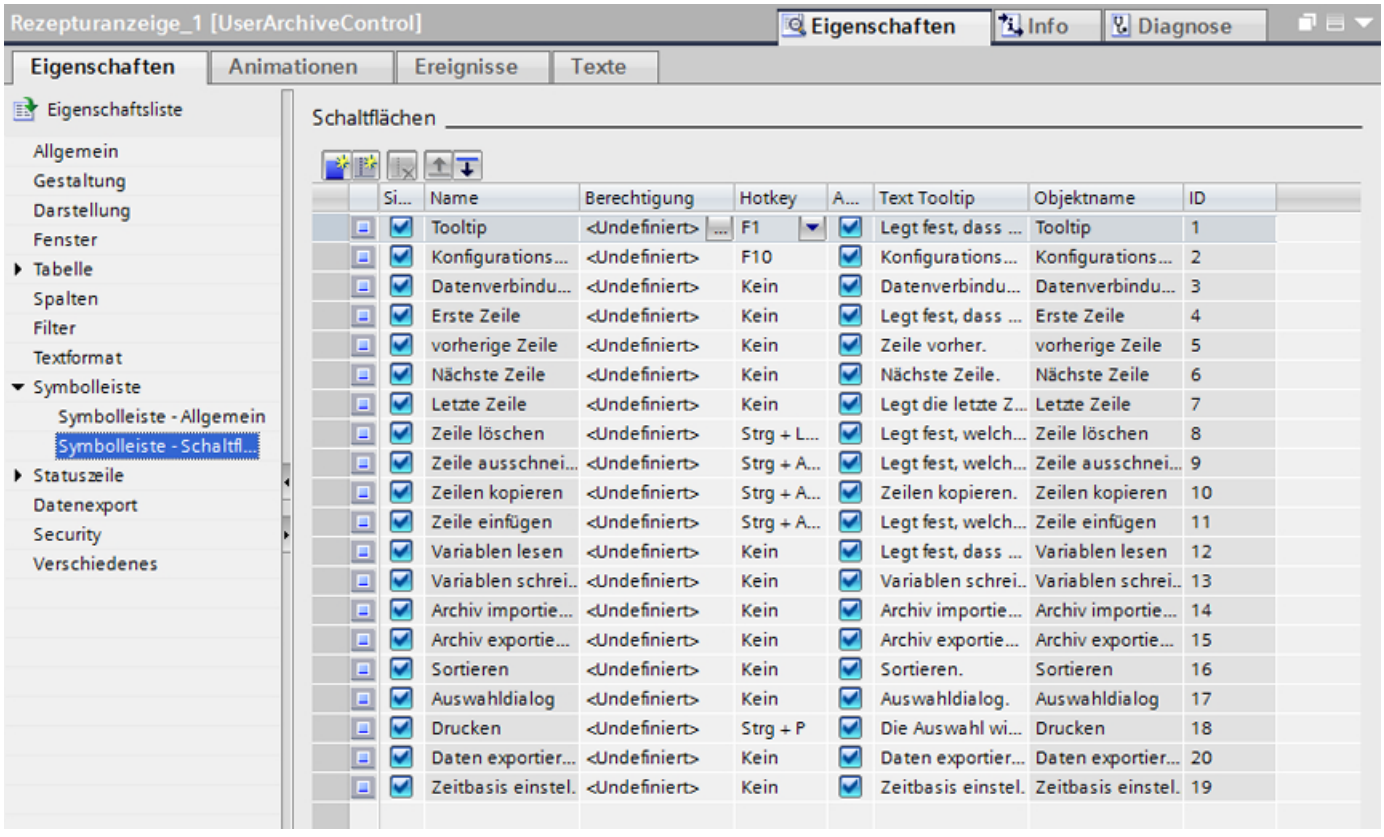

7. Legen Sie unter "Statuszeile" fest, welche Informationen in Runtime in der Statuszeile angezeigt werden.

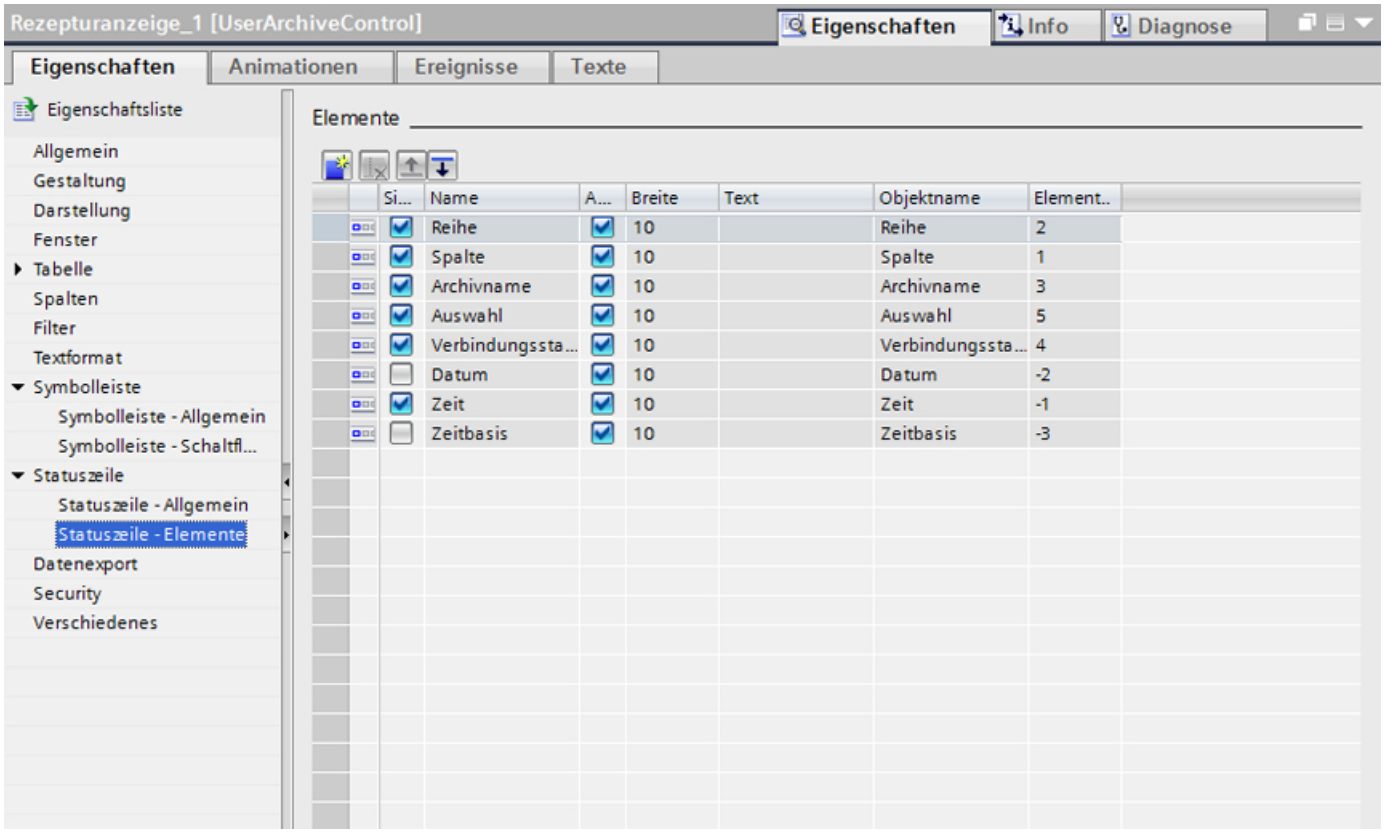

8. Geben Sie bei Bedarf unter "Filter" die Bedingungen ein, nach denen die Rezepturelemente in Runtime gefiltert werden.
9. Geben Sie bei Bedarf unter "Sortieren" die Kriterien ein, nach denen die Rezepturelemente in Runtime sortiert werden.

Ergebnis

Die Rezepturanzeige ist projektiert.

6.2.3.4 Rezepturdaten exportieren in Runtime

Einleitung

In Runtime exportieren Sie die Rezepturdaten über eine Schaltfläche in der Rezepturanzeige. Sie projektieren die Bedienung des Datenexports in Runtime.

So projektieren Sie die Bedienung des Datenexports

1. Legen Sie unter "Eigenschaften > Datenexport" einen Dateinamen und ein Verzeichnis für die Exportdatei fest.
   Der Dateiname kann sich aus selbst gewählten Namen und folgenden Schaltervariablen zusammensetzen:
   @OBJECTNAME: Objektname der Rezepturanzeige
   @PROJECTPATH: Projektverzeichnis
   @CURRENTDATE: aktuelles Datum
   @CURRENTTIME: aktuelle Zeit
1. Legen Sie die Dateiendung fest z. B. "CSV".
2. Bestimmen Sie den Umfang des Datenexports:
   – Alle Runtimedaten werden exportiert.
   – Ausgewählte Runtimedaten werden exportiert.
3. Projektieren Sie die Bedienung des Datenexportes in Runtime. Legen Sie fest:
   – In Runtime wird der Dialog "Datenexport-Voreinstellungen" angezeigt.
   – Der Benutzer darf den Dateinamen oder das Verzeichnis ändern.
4. Wenn "Dialog anzeigen" deaktiviert ist, werden die Daten bei Bedienung der Schaltfläche "Archiv exportieren" sofort in die festgelegte Exportdatei exportiert.
5. Speichern Sie das Projekt.
6. Aktivieren Sie unter "Eigenschaften > Symbolleiste" die Schaltfläche "Archiv exportieren".

Ergebnis

In Runtime können Sie alle oder ausgewählte Rezepturdaten über die Schaltfläche in eine festgelegte Datei exportieren.

6.2.3.5 Persistenz projektieren

Einleitung

Ein Benutzer kann in Runtime WinCC-Controls konfigurieren. Sie legen fest, wie sich die Online-Projektierung in Runtime auswirkt.

Die in Runtime projektierten Änderungen werden im Engineering System gesondert zum Bild gespeichert. Dadurch wird das Bild in seiner ursprünglichen Projektierung beibehalten.

Vorgehensweise

1. Wählen Sie den "Eigenschaften > Security > Persistenz > Online-Projektierung", den Sie nur im Engineering System bedienen können. In Runtime ist dann die Online-Konfigurierung unveränderbar festgelegt:
   - "Keine Persistenz". Die Online-Konfigurierungen werden in Runtime nicht beibehalten. Diese Voreinstellung lässt den Benutzer in Runtime keine Optionen. Die Online-Konfigurierungen gehen beim nächsten Bildwechsel und beim Deaktivieren/Aktivieren des Projekts verloren.
   - "Persistenz". Diese Voreinstellung lässt dem Benutzer in Runtime die Optionen "verwerfen", "beibehalten" oder "zurücksetzen". Mit der Option "beibehalten" werden die Online-Konfigurierungen beim nächsten Bildwechsel und beim Deaktivieren/Aktivieren des Projekts beibehalten.
   - "Persistenz in Runtime". Diese Voreinstellung lässt dem Benutzer in Runtime die Optionen "verwerfen", "beibehalten" oder "zurücksetzen". Mit der Option "beibehalten" werden die Online-Konfigurierungen beim nächsten Bildwechsel beibehalten, gehen aber beim Deaktivieren/Aktivieren des Projekts verloren.

   Hinweis

   Bei einem Benutzerwechsel werden die unterschiedlichen Konfigurierungen erst übernommen, nachdem Sie einen Bildwechsel durchgeführt haben.

2. Legen Sie fest, welche Bedienberechtigung der Benutzer haben muss, um Online konfigurieren zu können.
3. "Verhalten bei Bildwechsel" können Sie im Engineering System und in Runtime bei den Voreinstellungen "Persistenz" und "Persistenz während Runtime" bedienen. Die Option "zurücksetzen" ist nur in Runtime bedienbar, da sich im Engineering System die ursprüngliche Projektierung befindet.
   Wählen Sie eine der drei Auswirkungen der Online-Konfigurierung beim nächsten Bildwechsel:
   - Aktivieren Sie "Änderungen verwerfen", damit die Online-Konfigurierung nur solange besteht, wie das Bild angewählt ist. Beim nächsten Bildwechsel werden die Änderungen verworfen. Wenn Sie das Bild erneut anwählen, erscheint die Rezepturanzeige wie projektiert.
   - Aktivieren Sie "Änderungen beibehalten", damit die Online-Konfigurierung beim nächsten Bildwechsel nicht verloren geht.
   - Aktivieren Sie "zurücksetzen", damit das projektierte Bild erneut in Runtime übernommen wird. Alle online konfigurierten Änderungen gehen verloren.

   Hinweis

   Auch beim Speichern des Bildes oder bei der Delta-Übersetzung (Änderungsladen) wird das Bild in Runtime ersetzt. Alle online konfigurierten Änderungen gehen verloren.

4. Speichern Sie die Projektierung.

6.2.4 Datenübertragung zur Steuerung

6.2.4.1 Grundlagen zur Datenübertragung

Übersicht

Für die Datenübertragung von Rezepturwerten zwischen Steuerung und Bediengerät gibt es folgende Möglichkeiten:

- Keine Datenübertragung
- Kommunikationsart "Variablen"
- Kommunikationsart "Rohdatenvariable"

Keine Datenübertragung

Die Rezeptur dient nicht dazu, Rezepturdaten zur Steuerung zu übertragen, sondern als Datenspeicher und Anwenderarchiv: Zusammengehörige Werte legen Sie in Datensätzen ab, auf die Sie vielfältigen Zugriff haben. Im einfachsten Fall schaltet ein Bediener in einer Rezepturanzeige zwischen den Rezepturen um. Die jeweiligen Rezepturdaten werden in die Rezepturvariablen geladen und dort weiterverarbeitet.

Kommunikationsart "Variablen"

Rezepturvariablen übertragen die Rezepturwerte an die Steuerung. Umgekehrt werden die Werte aus der Steuerung über Rezepturvariablen in einen Rezepturdatensatz geschrieben. Dabei steuern Steuervariablen die Datenübertragung. Durch den gewünschten Eintrag in die entsprechenden Steuervariablen werden folgende Aufträge automatisch gestartet:

- Datensätze in die Rezepturvariablen schreiben und damit in die Steuerung
- Datensätze aus den Rezepturvariablen lesen und damit aus der Steuerung
- Datensätze löschen
- Datensätze neu anlegen

Hinweis

Bei der Kommunikationsart "Variablen" wird immer ein kompletter Datensatz übertragen. Die Übertragung von Rezepturdaten aus einer Rezepturabfrage heraus ist nicht möglich.

Kommunikationsart "Rohdatenvariable"

Das Automatisierungssystem sendet als aktiver Partner die Rohdatenvariable als Telegramm an das Bediengerät. Das Telegramm enthält Folgendes:

- Gewünschte Ziel-Rezeptur
- Gewünschte Ziel-Rezepturdatensätze
- Auftrag z. B. "Lesen", "Schreiben" oder "Löschen"
- Neue Rezepturwerte z. B. beim Auftrag "Schreiben"

Das Bediengerät liefert Rezeptur- und Quittierungsdaten als Telegramm zurück.

Für den Datenaustausch wird die Funktion "BSEND/BRCV" der S7-Kommunikation verwendet. Die laufende Nummer der Rezeptur dient dazu, die Rezeptur bei der Steuerung eindeutig zu identifizieren (PLCID).

Die Steuerung benötigt ein Anwenderprogramm, das in Runtime Folgendes ausführt:

- Daten zusammentragen bei Bedarf über E/A-Felder im Bediengerät.
- Daten prüfen.
- Rohdatenvariable als Telegramm erstellen.
- Telegramm senden.

Wenn Sie die Telegrammerzeugung vom Bediengerät aus starten wollen, projektieren Sie im Bild Funktionen, die das Anwenderprogramm starten.

Hinweis

Bei der Kommunikationsart "Rohdatenvariable" ist Folgendes möglich:

- Übertragen eines einzelnen Rezepturelements
- Übertragen eines ganzen Rezepturdatensatzes
- Übertragen mehrerer Rezepturdatensätze

Die Datenübertragung aus der Rezepturanzeige oder über Steuervariablen ist nicht möglich.

Steuervariablen

Einleitung

Bei der Kommunikationsart "Variablen" in der Rezeptur übertragen Sie die Rezepturdaten zwischen Steuerung und Bediengerät über Rezepturvariablen. Die Datenübertragung steuern vier Steuervariablen, die jeweils einer Rezeptur fest zugeordnet sind. Andere Steuervariablen können einer anderen Rezeptur zugeordnet sein.

Wenn der gewünschte Wert in die entsprechenden Steuervariablen eingetragen wird, wird die Datenübertragung automatisch gestartet.

Hinweis

Zur Gewährleistung der Funktion müssen Sie immer alle vier Steuervariablen einer Rezeptur angeben. Verändern Sie den Datentyp der Steuervariablen nicht.

Hinweis

Den Zugriffsschutz für die Steuervariablen einer geschützten Rezeptur müssen Sie über die Objekteigenschaften z. B. des Bildes, E/A-Feldes oder der Schaltfläche realisieren.

Steuervariablen

Steuervariable	Benötigter Datentyp	Bedeutung	
ID	Long	Nummer des Rezepturdatensatzes innerhalb der zugeordneten Rezeptur	
Auftrag	Long	Lesen aus den Rezepturvariablen, Schreiben in den Rezepturspeicher:	6
		Lesen aus dem Rezepturspeicher, Schreiben in die Rezepturvariablen:	7
		Löschen im Rezepturspeicher:	8
		Nach Ausführung des Auftrags steht in der Steuervariablen "Auftrag" eine Fehlerkennung:	
		Kein Fehler	0
		Fehler	-1
Feld	String	Der Name des Rezepturelements Achten Sie darauf, dass Sie nicht den Anzeigenamen verwenden.	
Wert	String	Der Wert des Rezepturelements	

Voraussetzung

- In der Rezeptur ist die Kommunikationsart "Variablen" eingestellt.
- Ein Prozessbild mit den benötigten E/A-Feldern für Prozessvariablen und Steuervariablen der Rezeptur ist projektiert.
  Beispiel:

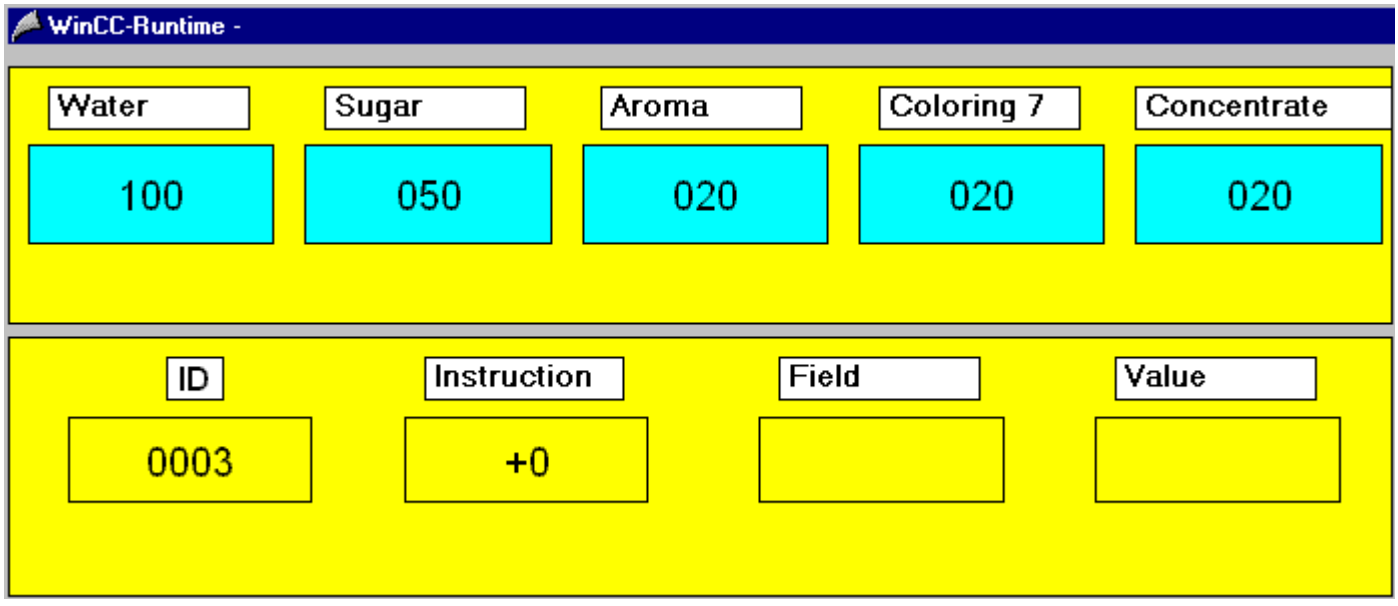

Einsetzen der Steuervariablen

Für die Steuervariablen sind folgende Kombinationen möglich:

- Steuervariablen "ID" und "Auftrag"
  Die Steuervariable "ID" gibt die Nummer des Rezepturdatensatzes an, der übertragen wird. Die Steuervariable "Auftrag" gibt an, ob der ausgewählte Rezepturdatensatz im Rezepturspeicher des Bediengeräts beschrieben, ausgelesen oder gelöscht wird (6, 7, 8).
- Steuervariablen "Auftrag", "Feld" und "Wert"
  Derjenige Rezepturdatensatz wird für die Datenübertragung ausgewählt, dessen Rezepturelement "Feld" den Rezepturwert "Wert" besitzt. Beispiel Feld = "Wasser", Wert = "5 Liter".
  Die Steuervariable "Auftrag" gibt an, ob der ausgewählte Rezepturdatensatz im Rezepturspeicher des Bediengeräts beschrieben, ausgelesen oder gelöscht wird (6, 7, 8).

Hinweis

Im Rezepturelement "Feld" darf in unterschiedlichen Datensätzen nicht der gleiche Rezepturwert stehen

Beim Rezepturelement muss unter "Eigenschaften > Extras" das Kontrollkästchen "Eindeutige Werte" aktiviert sein. Sonst ist keine eindeutige Zuordnung des Rezepturdatensatzes über den Wert möglich und es wird der erste Datensatz verwendet, auf den die Bedingung "Wert im Feld" zutrifft.

Wenn dem Element der Typ "String" zugeordnet ist, müssen Sie im E/A-Feld "Wert" die Eingabe in Hochkommas einschließen (z. B. 'Cola Light').

Weitere Wertekombinationen der Steuervariablen "ID" und "Auftrag"

ID	Auftrag = 6	Auftrag = 7	Auftrag = 8
-1	Datensatz aus den Rezepturvariablen wird gelesen und in der Rezeptur im Rezepturspeicher **hinten** angehängt.	-	Datensatz mit der **kleinsten** ID wird in der Rezeptur im Rezepturspeicher gelöscht.
-6	Datensatz aus den Rezepturvariablen wird gelesen und in den Rezepturdatensatz mit der **kleinsten** ID in der Rezeptur geschrieben.	Datensatz mit der **kleinsten** ID in der Rezeptur wird in die Rezepturvariablen geschrieben.	Datensatz mit der **kleinsten** ID wird in der Rezeptur im Rezepturspeicher gelöscht.
-9	Datensatz aus den Rezepturvariablen wird gelesen und in den Rezepturdatensatz mit der **größten** ID in der Rezeptur geschrieben.	Datensatz mit der **größten** ID in der Rezeptur wird in die Rezepturvariablen geschrieben.	Datensatz mit der **größten** ID wird in der Rezeptur im Rezepturspeicher gelöscht.

Siehe auch

Anzeige von Rezepturen (Seite 1232)

Rezepturbild bedienen (Seite 1257)

Unterschiede der Datenformate

Einleitung

In WinCC werden die Datenformate von Intel und Microsoft eingehalten. Grundsätzlich wird das niederwertige Byte zuerst und das höherwertige Byte zuletzt gespeichert. Dieses Datenformat ist sehr verbreitet und ist allgemein als "Intel-Format" bekannt.

Die Datenformate in WinCC unterscheiden sich grundsätzlich zu den Datenformaten in den Steuerungen.

Intel-Format

Beim Intel-Format wird die Dezimalzahl 300 folgendermaßen abgelegt:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binär	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0

Hex	0	1	2	C

Die Dezimalzahl 300 entspricht nach dem Intel-Format der Hexzahl 12C (1*256 + 2*16 + 12*1).

SIMATIC-Format

Beim SIMATIC-Format wird das niederwertige Byte auf höherwertigen Stellen abgelegt. Die Dezimalzahl 300 wird beim SIMATIC-Format folgendermaßen abgelegt:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binär	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1

Hex	2	C	0	1

Die Dezimalzahl 300 entspricht nach dem SIMATIC-Format der Hexzahl 2C01.

Wenn die Hexzahl 2C01 nach dem Intel-Format interpretiert wird, entspricht das der Dezimalzahl 11265.

Für die Steuerungen der SIMATIC S7-Reihe stehen Funktionsbausteine zur Verfügung, welche die Daten konvertieren. Sie müssen die Bausteine immer vor und nach Datenübertragungen zwischen Steuerung und WinCC aufrufen. Sie können die Funktionsbausteine im Internet vom Siemens Customer Support herunterladen. Es wird dann die komprimierte Datei ANSI_S5.EXE geladen. In ANSI_S5.EXE befindet sich der Funktionsbaustein "IEEE:GP".

Das aktive Senden ist in den Referenzhandbüchern der Steuerungen bzw. der CP's (Communication Processors) beschrieben.

6.2.4.2 Rohdatenvariable

Grundlagen zur Datenübertragung über Rohdatenvariable

Datenübertragung über Rohdatenvariable

Die folgenden Kapitel beschreiben die Datenübertragung der Rezepturdaten zwischen Bediengerät und Steuerung über Rohdatenvariablen. In der Steuerung wird die Funktion BSEND/BRCV benutzt. Das Telegramm, das die Rohdatenvariable enthält, wird von der Steuerung aktiv gesendet.

Die Telegramme können einen oder mehrere Schreib- bzw. Leseaufträge beinhalten.

Ablauf bei der Datenübertragung

1. Die Steuerung sendet das Telegramm zum Bediengerät.
2. Das Bediengerät sendet Folgendes zurück:
   - Daten falls angefordert
   - Eine Bearbeitungsquittung

Das Bediengerät sendet für jeden Auftrag eine eigene Quittung zurück.

Hinweis

Weil die Steuerung bei dieser Datenübertragung der aktive Partner ist, muss die Steuerung die gewünschte Funktion starten. Am Bediengerät können Sie die gewünschte Funktion z. B. durch eine Wertänderung einer externen Variablen und einer entsprechenden Auswertung in der Steuerung starten.

Verwenden Sie den Parameter "Auftragstyp" nicht zum Starten der Datenübertragung. Der im Auftrags- oder Quittungskopf verwendete Parameter "Auftragstyp" besitzt nur eine Funktionalität bezüglich der Rezepturen.

Auftragstypen und Fehlercodes

Beschreibung der Auftragstypen

Typ	Beschreibung
4	Die Rezeptur wird auf Vorhandensein geprüft.
5	Alle Datensätze werden aus der Rezeptur gelöscht.
6	Datensatz wird vom Bediengerät angefordert und in die Steuerung geschrieben.
7	Datensatz der Steuerung wird in den gewünschten Datensatz geschrieben.
8	Der Datensatz wird aus der Rezeptur gelöscht.

9	Das Element des gewünschten Datensatzes wird vom Bediengerät angefordert und in die Steuerung geschrieben.
10	Das Element des gewünschten Datensatzes wird in den gewünschten Datensatz geschrieben.

Beschreibung der Fehlercodes

Gruppe	Nr.	Beschreibung	Mögliche Fehlerursachen
Allgemein	0	Funktion ist ausgeführt worden.	--
Rezeptur	2	Nicht vorhandene Daten	Keine Rezeptur mit dieser PLCID projektiert.
Datensatz	101	Nicht zulässige Daten	Rezepturstruktur stimmt nicht überein, z. B. Anzahl oder Typ der Felder.
			Einfügen oder Aktualisieren des Datensatzes schlägt fehl. Mögliche Ursachen: • Rezeptur ist vom Typ "begrenzt". • Zu einem Feld ist ein Min- oder Max-Wert projektiert.
			Kriterium falsch
Datensatz	102	Nicht vorhandene Daten	Nur bei Auftragstyp 6: • Keine Daten vorhanden • Kriterium ist falsch.
Feld	201	Nicht zulässige Daten	Nur für Auftragstyp 10: Kriterium ist falsch. Mögliche Ursachen: • Feld ist nicht vorhanden. • Zu einem Feld ist ein Min- oder Max-Wert projektiert.
Feld	202	Nicht vorhandene Daten	Nur für Auftragstyp 9: • Kriterium ist falsch. • Kein Feld gefunden, das dem Kriterium entspricht.
Allgemein	254	Funktion nicht vorhanden	--
Allgemein	255	Undefinierter Fehler	--

Telegrammaufbau für Aufträge

Allgemeiner Telegrammaufbau

Aufbau des Telegramms zum Senden von Aufträgen:

	Länge
Telegrammkopf	16 Byte

	Länge
Auftragskopf 1	12 Byte
Daten des Auftrags 1	n Byte
Auftragskopf 2	12 Byte
Daten des Auftrags 2	n Byte
Auftragskopf n	12 Byte
Daten des Auftrags n	n Byte

Beispiel

Ein Telegramm mit einem Auftrag und Auftragsdaten von 10 Byte Länge hat eine Gesamtlänge von 38 Byte.

16 Byte+12 Byte +10 Byte = 38 Byte

Aufbau des Telegrammkopfs

Für den Aufbau eines Telegrammkopfs werden 16 Byte benötigt.

Funktion des Feldes	Kommentar	Byte
Telegrammlänge in Byte LSB [1])	Max Länge 4091 Bytes	1
Telegrammlänge in Byte		2
Telegrammlänge in Byte		3
Telegrammlänge in Byte MSB [2])		4
Transfertyp	"1" vom Bediengerät, "2" von der Steuerung	5
Reserviert		6
Anzahl der Aufträge im Telegramm LSB [1])		7
Anzahl der Aufträge im Telegramm MSB[2])		8
Nummer der Rezeptur (PLCID)	1. Zeichen, in ASCII	9
Nummer der Rezeptur (PLCID)	2. Zeichen, in ASCII	10
Nummer der Rezeptur (PLCID)	3. Zeichen, in ASCII	11
Nummer der Rezeptur (PLCID)	4. Zeichen, in ASCII	12
Nummer der Rezeptur (PLCID)	5. Zeichen, in ASCII	13
Nummer der Rezeptur (PLCID)	6. Zeichen, in ASCII	14
Nummer der Rezeptur (PLCID)	7. Zeichen, in ASCII	15
Nummer der Rezeptur (PLCID)	8. Zeichen, in ASCII	16

① Niederwertiges Byte

② Höherwertiges Byte

Hinweis

Die Nummer der Rezeptur wird bei der Projektierung im Editor "Rezepturen" automatisch vergeben, kann aber bei Bedarf geändert werden. Geben Sie im Telegramm die Nummer in ASCII an.

Aufbau eines Auftragskopfs

Für den Aufbau eines Auftragskopfs werden 12 Byte benötigt.

Funktion des Feldes	Kommentar	Byte
Auftragslänge in Byte LSB[1])		1
Auftragslänge in Byte MSB[2])		2
Auftragstyp	siehe "Beschreibung der Auftragstypen und Fehlercodes"	3
Reserviert		4
Feldnummer LSB[1])		5
Feldnummer MSB[2])		6
Datensatznummer LSB[1])		7
Datensatznummer		8
Datensatznummer		9
Datensatznummer MSB[2])		10
Selektionskriterium LSB[1])	Eintragswert des Elements oder Elementnummer, nach der selektiert wird.	11
Selektionskriterium MSB[2])	nicht bei 0	12

① Niederwertiges Byte
② Höherwertiges Byte

Die Feldnummer entspricht der laufenden Nummer des Rezepturelements.

Telegrammaufbau für Quittungen

Allgemeiner Aufbau eines Quittungstelegramm

Aufbau des Quittungstelegramms an die Steuerung::

	Länge
Quittungskopf	24 Byte
Quittungsdaten	

Aufbau eines Quittungskopfs

Für den Aufbau eines Quittungskopfs werden 24 Byte benötigt.

Funktion des Feldes	Kommentar	Byte
Telegrammlänge in Byte LSB [1])	Max Länge 4091 Bytes	1
Telegrammlänge in Byte		2
Telegrammlänge in Byte		3
Telegrammlänge in Byte MSB [2])		4
Transfertyp	"1" vom Bediengerät, "2" von der Steuerung	5
Reserviert		6
Auftragstyp	Siehe Beschreibung "Auftragstypen und Fehlercodes"	7
Fehlercode	Siehe Beschreibung "Auftragstypen und Fehlercodes"	8
Reserviert		9
Reserviert		10
Feldnummer LSB[1])		11
Feldnummer MSB[2])		12
Datensatznummer LSB[1])		13
Datensatznummer		14
Datensatznummer		15
Datensatznummer MSB[2])		16
Nummer der Rezeptur (PLCID)	1. Zeichen, in ASCII	17
Nummer der Rezeptur (PLCID)	2. Zeichen, in ASCII	18
Nummer der Rezeptur (PLCID)	3. Zeichen, in ASCII	19
Nummer der Rezeptur (PLCID)	4. Zeichen, in ASCII	20
Nummer der Rezeptur (PLCID)	5. Zeichen, in ASCII	21
Nummer der Rezeptur (PLCID)	6. Zeichen, in ASCII	22
Nummer der Rezeptur (PLCID)	7. Zeichen, in ASCII	23
Nummer der Rezeptur (PLCID)	8. Zeichen, in ASCII	24

① Niederwertiges Byte
② Höherwertiges Byte

Quittungsdaten

Die Quittung kann Folgendes enthalten:

- Den angeforderten Rezepturdatensatz
- Den angeforderten Inhalt des Elements
- Keine Daten

Rohdatenvariable projektieren

Rohdatenvariablen des Kanals "SIMATIC S7"

Einleitung

Eine Variable vom Typ "Rohdatentyp" entspricht einem Datentelegramm auf Transportebene. Der Inhalt der Rohdatenvariable ist nicht festgelegt und deshalb ist die Interpretation der übertragenen Daten nur dem Sender und Empfänger möglich. Für diesen Datentyp erfolgt in WinCC keine Formatanpassung. Die Länge beträgt max. 65535 Byte.

In WinCC werden zwei Arten von Rohdatenvariablen unterschieden: Rohdatenvariable für freie Anwendernutzung und Rohdatenvariable zur Abwicklung von S7-Funktionen.

Rohdatenvariable für freie Anwendernutzung

Rohdatenvariablen für die freie Anwendernutzung dienen zur Übertragung von Anwenderdatenblöcken zwischen WinCC und AS und hantieren nur die Nutzdaten. Dabei werden unterschieden:

- Rohdatenvariable als Byte-Array
- Rohdatenvariable für BSEND/BRCV-Funktionen

Rohdatenvariable zur Abwicklung von S7-Funktionen

Diese Rohdatenvariablen haben einen kanal-spezifischen Header und werden in der Regel vom Meldesystem und der Prozessdaten-Erfassung in WinCC genutzt.

Eine weiter gehende Beschreibung erfolgt hier nicht, da es sich um kanalinterne Variablen und Funktionen handelt.

Rohdatenvariable anlegen

Vorgehensweise

1. Legen Sie eine neue Verbindung an z. B. "SIMATICS7".
2. Aktivieren Sie auf der Registerkarte "Parameter" "WinCC RT Prof Verbindung > Rohdatenblock senden/empfangen".

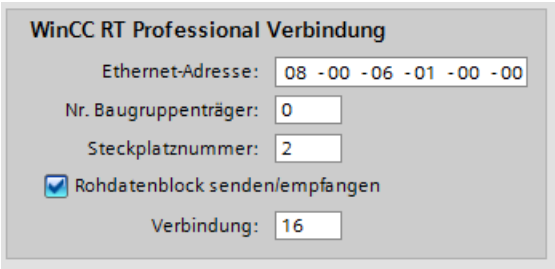

3. Geben Sie unter "Verbindung" den Hexadezimalwert der Verbindungs-ID der Verbindungsressource, die bei der Projektierung der Steuerung vergeben wurde.
4. Legen Sie eine neue Variable "Rohdatenvariable_1" an.

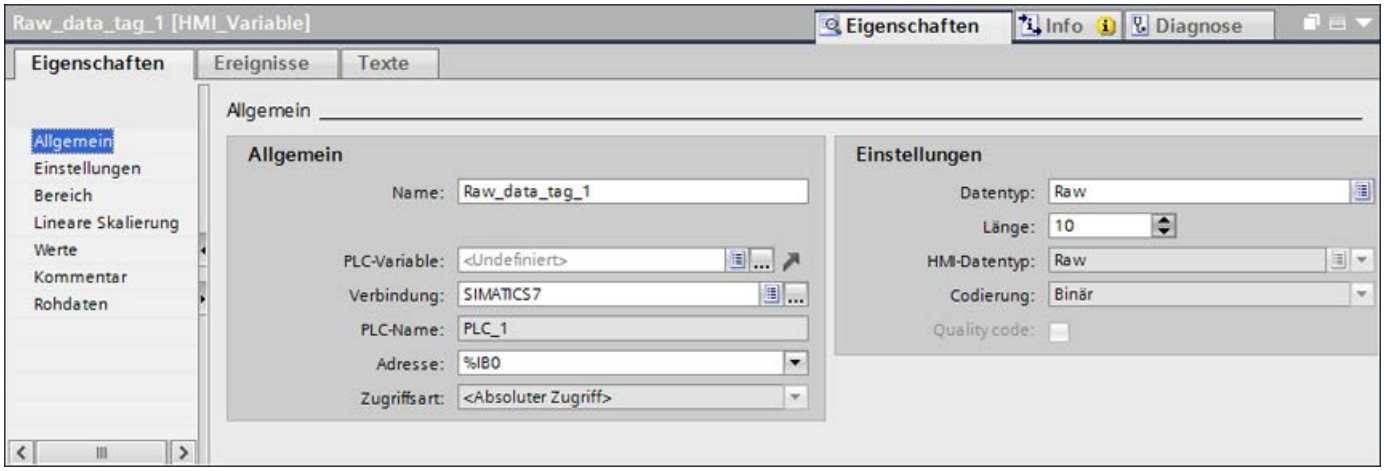

5. Wählen Sie als "Verbindung" "SIMATICS7".
6. Wählen Sie als "Datentyp" "Raw" und legen Sie eine ausreichende "Länge" in Bytes fest.
7. Aktivieren Sie unter "Rohdaten" "BSEND / BRCV"

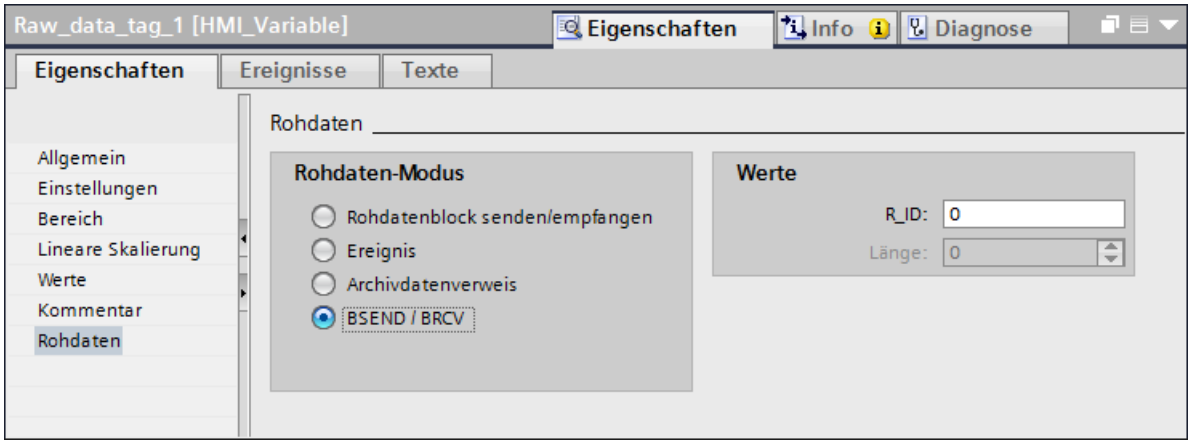

8. Geben Sie den Parameter "R_ID" an, der bei der Projektierung der Steuerung vergeben wurde.
9. Legen Sie eine neue Rezeptur an.
10. Wählen Sie im Inspektorfenster unter "Eigenschaften > Kommunikation" als Kommunikationsart "Rohdatenvariable" und als Rohdatenvariable "Rohdatenvariable_1".
11. Legen Sie Rezepturelemente und Rezepturdatensätze an. Beachten Sie dabei, dass die Rezepturnummer dem Feld "PLC" im Telegramm entspricht.

Rohdatenvariable für BSEND/BRCV-Funktionen der S7-Kommunikation

Einleitung

Rohdaten-Variable für "BSEND/BRCV"-Funktionen dienen der Übertragung von Anwender-Datenblöcken zwischen WinCC und AS und hantieren nur die Nutzdaten.

Über diesen Rohdatentyp können die Funktionen "BSEND/BRCV" der S7-Kommunikation genutzt werden. Die Funktionen sind bei einer S7-400 oder einer S7-300 mit CP343 verfügbar. Die Initiative der Datenübertragung liegt immer beim sendenden Partner, sodass über die "BSEND/BRCV"-Funktionen auch eine ereignisgesteuerte bzw. sporadische Datenblock-Übertragung realisiert werden kann.

Aus Ressourcengründen sollte die Anzahl der BSEND/BRCV-Rohdatenvariablen gering gehalten werden.

Ressourcenbegrenzung bei der Nutzung der S7-Funktionen "AR_SEND" und "BSEND/BRCV" zur Kommunikation mit S7-400

Die maximale Datenmenge, die zu einer Zeit gleichzeitig mit AR_SEND- und/oder BSEND/BRCV-Funktionen vom AS an WinCC gesendet werden können, ist begrenzt auf 16 kByte.

Beispiele:

- 1x BSEND mit max. 16 kByte
- 1x AR_SEND mit 8 kByte + 1x BSEND mit 8 kByte
- 1x AR_SEND mit 10 kByte + 1x AR_SEND mit 2 kByte + 1x BSEND mit 4 kByte

Hinweis

Wurde der Datenblock eines Schreibauftrages in das AS übertragen, dort aber noch nicht oder nicht vollständig aus dem Empfangspuffer gelöscht, wird ein weiterer Schreibauftrag mit einer Fehleranzeige zurückgewiesen. Schreibaufträge mit einer R_ID > 0x8000 0000 werden bei einer solchen Fehleranzeige in eine verbindungsspezifische Warteschlange eingetragen und es wird ca. 6 Sekunden lang versucht, den Schreibauftrag zu wiederholen.

Die zeitliche Koordination der Übertragung liegt beim Anwender und muss bei Schreibaufträgen mit kürzeren zeitlichen Abständen beachtet werden.

Projektierung einer PBK-Verbindung zur Nutzung von "BSEND/BRCV"-Funktionen

Die Nutzung von "BSEND/BRCV"-Funktionen ist nur über eine "festprojektierte Verbindung", eine sog. PBK-Verbindung (Programmierte-Baustein-Kommunikation) möglich. Zur Projektierung einer festprojektierten Verbindung ist in den Verbindungsparametern eine Verbindungsressource (Hex: 10 ... DF) anzugeben. Die Verbindungsressource wird bei der Projektierung der Verbindung im AS von STEP 7 vergeben. Die Verbindung muss im AS als passiver Verbindungsendpunkt projektiert werden.

Über eine festprojektierte Verbindung können auch "normale" Lese- und Schreibaufträge abgewickelt werden. Werden über die Verbindung sehr große Datenbereiche übertragen, so werden die Datenblöcke in mehreren PDUs übertragen. Aus Performancegründen kann es deshalb sinnvoll sein, für die "BSEND/BRCV"-Funktionen eine eigene Verbindung anzulegen.

Rohdatenvariable für "BSEND/BRCV"-Funktionen mit Parameter "R_ID"

Die Rohdaten-Variablen zum Übertragen von "BSEND/BRCV"-Datenblöcken werden als Rohdaten vom Typ "BSEND/BRCV" mit einer "R_ID" projektiert. Die Datenlänge ergibt sich implizit aus der gesendeten bzw. empfangenen Datenmenge.

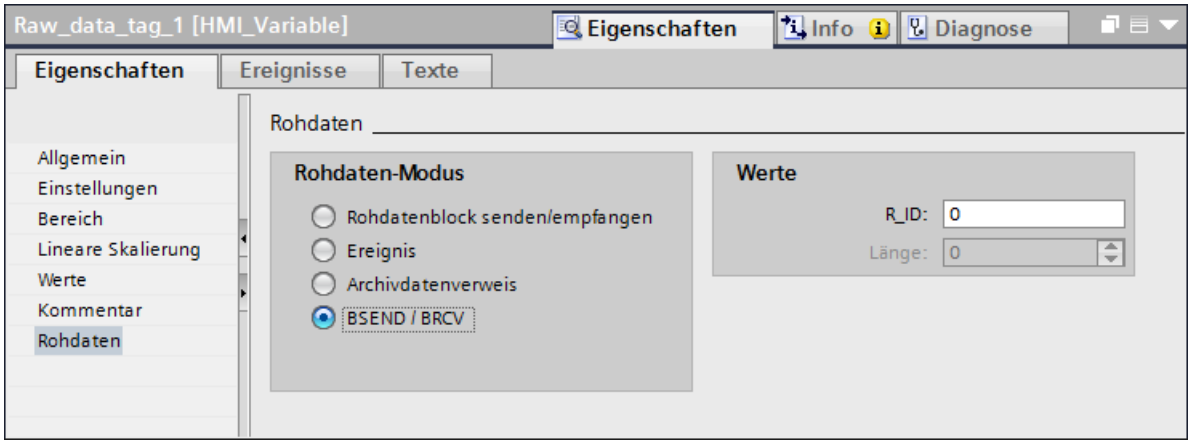

Als "R_ID" muss ein dezimaler Wert von 1 bis 999999 angegeben werden. Die R_ID wird bei der Projektierung im AS vergeben und dient der Unterscheidung von mehreren Datenblockübertragungen über eine Verbindung. An das unterlagerte Kommunikationssubsystem (SIMATIC Device Drivers) werden die Sende- und Empfangsaufrufe immer in Bezug auf diese R_ID angegeben. Eine Rohdaten-Variable ist somit immer einer eindeutigen R_ID zugeordnet.

Senden einer "BSEND/BRCV"- Rohdaten-Variable

Das Senden einer "BSEND/BRCV"-Rohdaten-Variable erfolgt wie das Schreiben einer "normalen" Prozess-Variablen. Nach dem Senden des Datenblocks und dem Empfang einer positiven Quittung vom AS wird der Datenblock in das Abbild des Datenmanagers übernommen.

Empfang einer "BSEND/BRCV"- Rohdaten-Variable

"BSEND/BRCV" Rohdaten werden sporadisch aufgrund der Initiative des AS an den Kanal gesendet. Ein explizites Lesen von S7 Rohdaten-Variablen ist deshalb nicht möglich.

Die Mechanismen des BSEND/BRCV beinhalten keine Funktionen für die Synchronisation. Wenn sich in der Anlaufphase noch kein Anwender zum Empfang der Daten angemeldet hat, werden die vom AS gesendeten Datenblöcke auf der Empfangsseite verworfen. Der Anwender muss daher selbst für die Synchronisation sorgen und z.B. die Senderichtung auf dem AS durch Setzen eines Flags in einem Datenwort freischalten.

6.2.5 Bedienung in Runtime

6.2.5.1 Rezepturanzeige

Funktionsumfang

Die Rezepturanzeige bietet Zugriff auf Anwenderarchive, den Rezepturen und Rezepturabfragen.

Sie können in Runtime:

- Rezepturanzeige online konfigurieren
  Die in Runtime projektierten Änderungen werden im Konfigurationssystem gesondert zum Bild benutzerabhängig gespeichert. Dadurch wird das Bild im Konfigurationssystem in seiner ursprünglichen Projektierung beibehalten.
- Datensätze anlegen, löschen oder ändern
- In Rezepturen blättern
- Variablen bei direkter Variablenanbindung lesen und schreiben
- Rezepturen importieren und exportieren
- Selektionskriterien definieren, um nur einen bestimmten Teil der Rezepturen anzuzeigen
- Sortierbedingungen für die angezeigten Rezepturelemente definieren

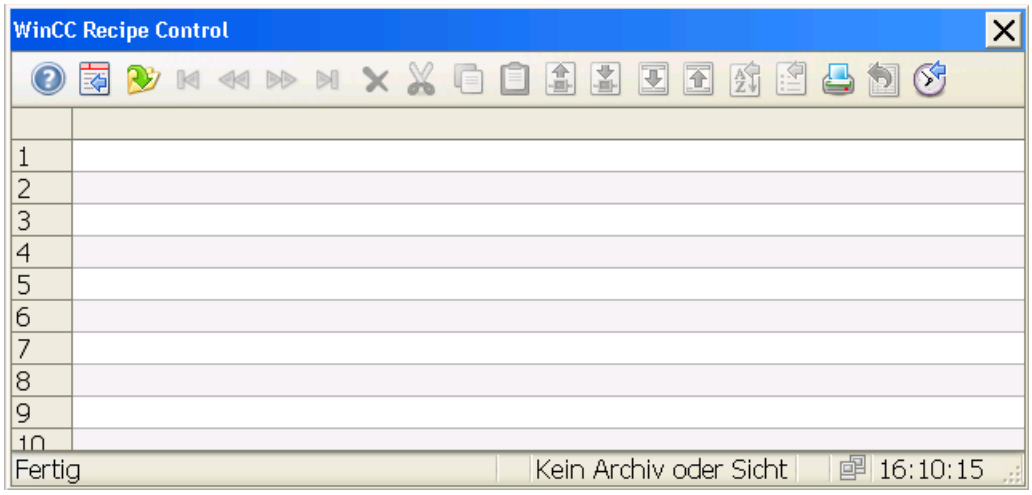

Eigenschaften

Eine Rezepturanzeige verbinden Sie bei der Projektierung mit einer Rezeptur oder Rezepturabfrage. Die Online-Konfigurierung und Änderung ausgewählter Rezepturwerte in Runtime können Sie jeweils durch eigene Bedienberechtigungen vor unberechtigtem Zugriff schützen. Wenn Sie den Zugriffsschutz aufheben, müssen Sie im Konfigurationsdialog die Rezepturanzeige neu mit der Rezeptur verbinden.

Der Zugriffsschutz wird beim Bildaufschlag der Rezepturanzeige abgefragt:

- Wenn der Benutzer keine Berechtigung zum Lesen eines Rezepturelements hat, wird die entsprechende Spalte in der Tabelle nicht angezeigt.
- Wenn der Benutzer keine Berechtigung zum Schreiben eines Rezepturelements hat, kann der Benutzer die Daten der entsprechenden Spalte in der Tabelle nicht bearbeiten.

Einen Zugriffsschutz für die Steuervariablen müssen Sie über die Objekteigenschaften z. B. eines Bildes, eines EA-Feldes oder einer Schaltfläche separat projektieren.

6.2.5.2 Rezepturbild bedienen

Einleitung

Sie bedienen das Rezepturbild mit den von Ihnen projektierten Bedienobjekten. Der Abschnitt beschreibt die Bedienung des Rezepturbildes mit Steuervariablen aus dem Abschnitt "Steuervariablen (Seite 1243)".

Hinweis

Runtime API

In der Runtime API stehen Ihnen umfangreiche uaArchive-Funktionen zur Verfügung, um Rezepturen und Rezepturabfragen in Runtime zu bearbeiten (siehe Kapitel "Schnittstellen > Runtime API", Abschnitt "Steuervariablen").

Voraussetzung

- In der Rezeptur ist die Kommunikationsart "Variablen" eingestellt.
- In den Eigenschaften für das Rezepturelement "Coloring 7" ist unter "Extras" das Kontrollkästchen "Eindeutige Werte" aktiviert.
- Rezepturvariablen und Steuervariablen sind eingerichtet.
- Ein Prozessbild mit den benötigten E/A-Feldern ist projektiert.
- Die E/A-Felder sind mit den Rezepturvariablen und Steuervariablen verbunden.

Beispiel:

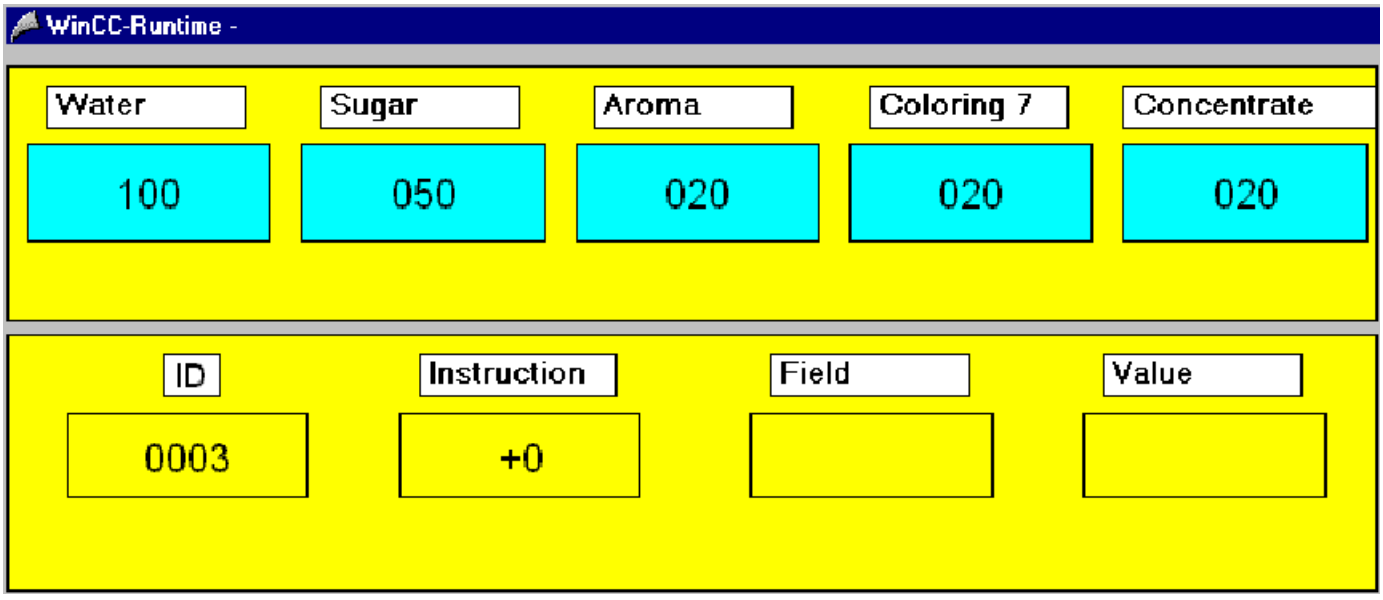

Vorgehensweise

Um Rezepturdaten in den Rezepturspeicher des Bediengeräts zu übertragen, gehen Sie wie folgt vor:

1. Geben Sie in den blauen Feldern "Water", "Sugar", "Aroma", "Coloring 7", "Concentrate" die abgebildeten Rezepturwerte ein. Die Rezepturwerte bilden den zu übertragenen Rezepturdatensatz.
2. Geben Sie als "ID" für den Ziel-Rezepturdatensatz die Nummer "3" ein.
3. Geben Sie als Auftrag im Feld "Instruction" zum Lesen "6" ein.

Die eingegebenen Rezepturdaten werden als kompletter Datensatz übertragen. Der Rezepturdatensatz Nummer 3 wird im Rezepturspeicher überschrieben. Der Steuerauftrag gibt in der Steuervariablen "Auftrag" den Wert "0" (kein Fehler) zurück.

Alternative Vorgehensweise

1. Geben Sie in den blauen Feldern " Water", "Sugar", "Aroma", "Coloring 7", "Concentrate" neue Rezepturwerte ein.
2. Geben Sie als "Field" das Rezepturelement "Coloring 7" ein.
3. Geben Sie als Wert im Feld "Value" des Rezepturelements "020" ein.
4. Geben Sie als Auftrag im Feld "Instruction" zum Lesen "6" ein.

Die eingegebenen Rezepturdaten werden als kompletter Datensatz übertragen. Der Rezepturdatensatz, dessen Rezepturelement " Coloring 7" den Rezepturwert "020" besitzt - im Beispiel der Rezepturdatensatz Nummer 3 im Rezepturspeicher, wird überschrieben. Der Steuerauftrag gibt in der Steuervariablen "Auftrag" den Wert "0" (kein Fehler) zurück.

Siehe auch

Anzeige von Rezepturen (Seite 1232)

Steuervariablen (Seite 1243)

6.2.5.3 Maus- und Tastatur-Bedienung

Einleitung

Sie bedienen die Rezepturanzeige in Runtime über Schaltflächen in der Symbolleiste. Wenn Sie nicht über die Symbolleiste bedienen möchten, können Sie über eine beliebige Dynamisierungsart die "ID" der gewünschten Schaltfläche in die Objekteigenschaft "ToolbarButtonClick" schreiben.

Übersicht

Symbol	Bechreibung	ID
	"Tooltip" Ruft die Hilfe zur Rezepturanzeige auf.	1
	"Konfigurationsdialog" Öffnet den Konfigurationsdialog, in dem Sie die Eigenschaften der Rezepturanzeige verändern.	2
	"Datenanbindung wählen" Öffnet einen Dialog zur Auswahl einer Rezeptur oder Rezepturabfrage. Die Rezepturdaten werden dann in der Tabelle der Rezepturanzeige dargestellt.	3
	"Erste Zeile" Über die Schaltfläche wird in der Tabelle der erste Datensatz dargestellt.	4
	"Vorherige Zeile" Über die Schaltfläche wird in der Tabelle der vorangehende Datensatz dargestellt.	5
	"Nächste Zeile" Über die Schaltfläche wird in der Tabelle der nachfolgende Datensatz dargestellt.	6
	"Letzte Zeile" Über die Schaltfläche wird in der Tabelle der letzte Datensatz dargestellt.	7
	"Zeile löschen" Der Inhalt der markierten Zeilen wird gelöscht.	8
	"Zeile ausschneiden" Der Inhalt der markierten Zeilen wird ausgeschnitten.	9
	"Zeile kopieren" Der Inhalt der markierten Zeilen wird kopiert.	10
	"Zeile einfügen" Der Inhalt der kopierten bzw. ausgeschnittenen Zeilen wird ab der markierten Zeile eingefügt.	11

	"Variablen lesen" Über die Schaltfläche werden die Inhalte der verbundenen WinCC-Variablen gelesen und in die Rezepturelemente geschrieben. Um die Schaltfläche nutzen zu können, muss bei der angezeigten Rezeptur die Kommunikationsart "Variablen" aktiviert sein. Die Rezepturelemente müssen mit Variablen verbunden sein.	12
	"Variablen schreiben" Über die Schaltfläche werden die Inhalte der Rezepturelemente in die verbundenen WinCC-Variablen geschrieben. Um die Schaltfläche nutzen zu können, muss bei der angezeigten Rezeptur die Kommunikationsart "Variablen" aktiviert sein. Die Rezepturelemente müssen mit Variablen verbunden sein.	13
	"Archiv importieren" Über die Schaltfläche wird eine CSV-Datei aus dem Verzeichnis "ua" des Projektordners in die Tabelle der Rezepturanzeige importiert.	14
	"Archiv exportieren" Der beim Laden ursprüngliche Inhalt der Tabelle wird mit Tabellenüberschriften über die Schaltfläche exportiert. Die Rezeptur wird als CSV-Datei im Verzeichnis "ua" des Projektordners abgelegt.	15
	"Sortieren" Öffnet einen Dialog zur Einstellung einer benutzerdefinierten Sortierung der angezeigten Spalten.	16
	"Auswahldialog" Legt Auswahlkriterien fest für die in der Tabelle darzustellenden Spalten der Rezepturanzeige.	17
	"Drucken" Startet den Ausdruck der angezeigten Werte. Den beim Druck verwendeten Druckauftrag legen Sie im Konfigurationsdialog auf der Registerkarte "Allgemeines" fest.	18
	"Zeitbasis einstellen" Öffnet einen Dialog zur Einstellung der Zeitbasis für die verwendeten Zeiten.	19
	"Daten exportieren" Über die Schaltfläche exportieren Sie die aktuellen Runtime-Daten einer Rezeptur oder Rezepturabfrage in eine CSV-Datei. Wenn die Option "Dialog anzeigen" aktiviert ist, öffnet sich ein Dialog, in dem Sie die Einstellungen zum Export sehen und den Export starten können. Mit den entsprechenden Bedienberechtigungen dürfen Sie auch die Datei und das Verzeichnis für den Export auswählen. Wenn kein Dialog angezeigt wird, startet sofort der Export der Daten in die voreingestellte Datei.	20
	"Benutzerdefiniert 1" Zeigt die erste, vom Benutzer erstellte Schaltfläche an. Die Funktion der Schaltfläche ist benutzerdefiniert.	1001

Mögliche Elemente der Statuszeile

In der Statuszeile der Rezepturanzeige können folgende Elemente erscheinen:

Symbol	Name	Beschreibung
Archiv: archive1	Archivname	Zeigt den Namen der ausgewählten Rezeptur oder Rezepturabfrage.
Zeile 1	Zeile	Zeigt die Nummer der markierten Zeile an.
Spalte 2	Spalte	Zeigt die Nummer der markierten Spalte an.
08.01.2008	Datum	Zeigt das Systemdatum.
09:02:20	Zeit	Zeigt die Systemzeit.
	Zeitbasis	Zeigt die Zeitbasis, die bei der Anzeige von Zeiten verwendet wird.

Navigation in der Tabelle

In der Tabelle können Sie wie folgt navigieren:

- Zur nächsten Zelle gelangen Sie mit der Taste "ENTER" oder mit der Cursortaste "Rechts".
- Zur vorherigen Zelle gelangen Sie mit der Taste "SHIFT+ENTER" oder mit der Cursortaste "Links".
- Zur nächsten Zeile gelangen Sie durch Mausklick in die Zeile oder mit der Cursortaste "Ab".
- Zur vorherigen Zeile gelangen Sie durch Mausklick in die Zeile oder mit der Cursortaste "Auf".

Hinweis

Wenn Sie beim Start von Runtime die Fehlermeldung "Error while connecting the data!" eingeblendet wird, hat die Rezepturanzeige keine Anbindung an eine Rezeptur oder Rezepturabfrage. Prüfen Sie Folgendes:

- Der Name ist korrekt angegeben.
- Die Projektierung ist nicht geändert.
- Die verbundene Rezeptur oder Rezepturabfrage existiert noch.

6.2.5.4 Rezepturdaten bearbeiten

Einleitung

Sie können in der Rezepturanzeige Daten bearbeiten. Folgende Möglichkeiten stehen zur Verfügung:

- Neue Daten eingeben
- Bestehende Daten ändern
- Zeilen löschen
- Zeilen ausschneiden, kopieren und wieder einfügen

Voraussetzung

- Sie haben im Konfigurationsdialog auf der Registerkarte "Allgemeines" das Bearbeiten zugelassen.
- Für die zu bearbeitende Spalte haben Sie im Konfigurationsdialog auf der Registerkarte "Spalten" die Eigenschaft "Schreibgeschützt" deaktiviert.
- Die Spalte "ID" können Sie nicht bearbeiten.
- Wenn die Rezepturanzeige mit einer Rezepturabfrage verbunden ist, können Sie eine Zeile nicht löschen und ausschneiden.

Neue Daten in die Tabelle eingeben

1. Klicken Sie auf , um zur letzten Zeile zu gelangen. Die Zeile ist markiert.
2. Doppelklicken Sie in die erste Zelle der markierten Zeile. Sie können auch in der Zelle auf "F2", "ALT+ENTER" oder "CTRL+ENTER" drücken.
3. Geben Sie nacheinander die Werte in die Zellen ein und bestätigen Sie jedesmal mit der Eingabetaste. Nachdem Sie alle Werte in die Zeile eingegeben haben und eine andere Zeile markieren, wird der neue Datensatz in die Rezeptur geschrieben. Zu einer anderen Zeile gelangen Sie durch Mausklick, mit der Taste "ENTER" oder mit den Cursortasten "Auf" und "Ab".
4. Sie können die Daten einer markierten Zelle mit "CTRL+C" bzw. "CTRL+X" in die Zwischenablage kopieren. Mit "CTRL+V" fügen Sie die kopierten Daten in eine markierte Zelle ein.

Bestehende Daten in der Tabelle ändern

1. Klicken Sie auf oder , um zur gewünschten Zeile zu gelangen. Sie können auch über den Rollbalken zur gewünschten Zeile gelangen.
2. Doppelklicken Sie in die gewünschte Zelle der markierten Zeile. Sie können auch in der Zelle auf "F2", "ALT+ENTER" oder "CTRL+ENTER" drücken.
3. Geben Sie nacheinander die Werte in die Zellen ein und bestätigen Sie jedesmal mit der Eingabetaste. Nachdem Sie alle Werte in die Zeile eingegeben haben und eine andere Zeile markieren, wird der geänderte Datensatz in die Rezeptur geschrieben.

Eine Zeile in der Tabelle löschen

1. Klicken Sie auf oder , um zur gewünschten Zeile zu gelangen. Sie können auch über den Rollbalken zur gewünschten Zeile gelangen.
2. Klicken Sie auf , um die markierte Zeile zu löschen.

Ausschneiden, Kopieren und Einfügen von Zeilen

1. Klicken Sie auf oder , um zur gewünschten Zeile zu gelangen. Sie können auch über den Rollbalken zur gewünschten Zeile gelangen.
2. Klicken Sie auf oder , um die Daten der Zeile auszuschneiden bzw. zu kopieren. Alternativ können Sie auch die Tastenkombination "CTLR+ALT+X" bzw. "CTLR+ALT+C" verwenden.
3. Gehen Sie zur gewünschten Zeile, in die Sie die Daten hinkopieren wollen. Klicken Sie auf , um die ausgeschnittenen bzw. kopierten Daten einzufügen. Wenn Sie die Daten der markierten Zeile nicht überschreiben wollen, gehen Sie zur letzten Zeile um die Daten einzufügen.

6.2.5.5 Daten auswählen und konfigurieren

Einleitung

Die Rezepturanzeige zeigt in einer Tabelle die Daten der verbundenen Rezeptur. Der angezeigte Inhalt der Tabelle wird bestimmt von den ausgewählten Spalten und der Selektion von Inhalten der Spalten.

Voraussetzung

- Sie haben ein oder mehrere Rezepturen bzw. Rezepturabfragen angelegt.
- Sie haben die Rezepturanzeige mit einer Rezeptur oder Rezepturabfrage verbunden.

Konfigurieren der Spalten

1. Wechseln Sie zur Registerkarte "Spalten".

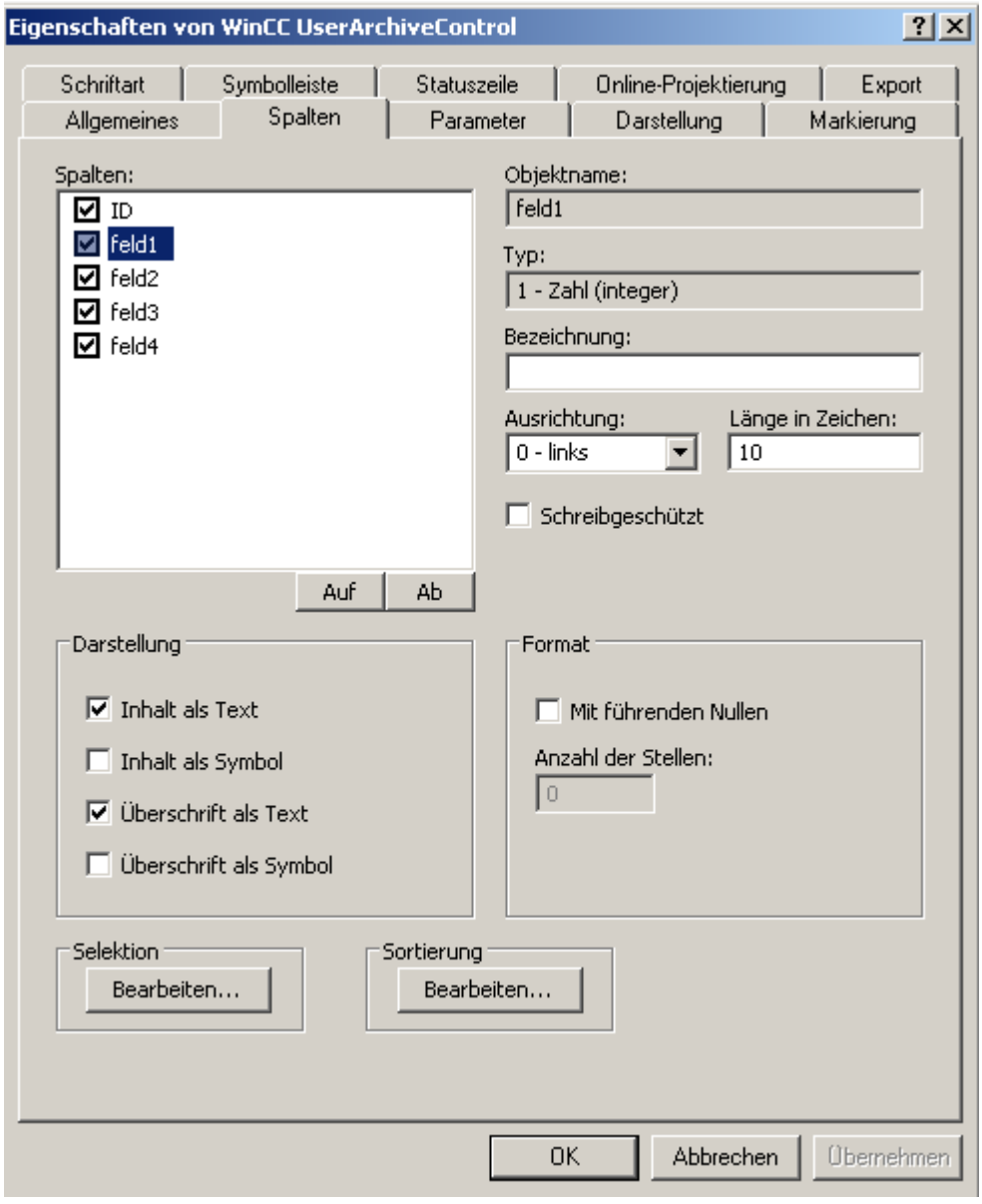

2. In der Liste "Spalten" sehen Sie die Felder der verbundenen Rezeptur bzw. Rezepturabfrage. Wenn ein Haken im Optionskästchen vor den Namen der Spalte gesetzt ist, wird die Spalte in der Tabelle angezeigt. Deaktivieren Sie mit der Maus das Optionkästchen, wenn die Spalte nicht dargestellt wird.
3. Legen Sie die Reihenfolge der Spalten in der Tabelle mit den Schaltflächen "Auf" bzw. "Ab" fest.
4. Wählen Sie eine Spalte aus, um die Eigenschaften und das Format zu konfigurieren.
5. Ändern Sie bei Bedarf die Breite der Spalte in der Tabelle. Geben Sie einen Wert in das Feld "Länge in Zeichen" ein.

6. Einige Spalten können den Inhalt und die Überschrift auch als Symbol darstellen. Bestimmen Sie im Feld "Darstellung", wie diese Spalten dargestellt werden. Text und Symbol können gemeinsam angezeigt werden.
7. Speichern Sie die Projektierung.

Selektion der Inhalte der Spalte, die in der Tabelle angezeigt werden

Im Bereich "Selektion" projektieren Sie Kriterien, die bei der Anzeige der Inhalte in den Spalten berücksichtigt werden.

Vorgehensweise

1. Klicken Sie auf "Bearbeiten...". Der Selektionsdialog wird geöffnet.

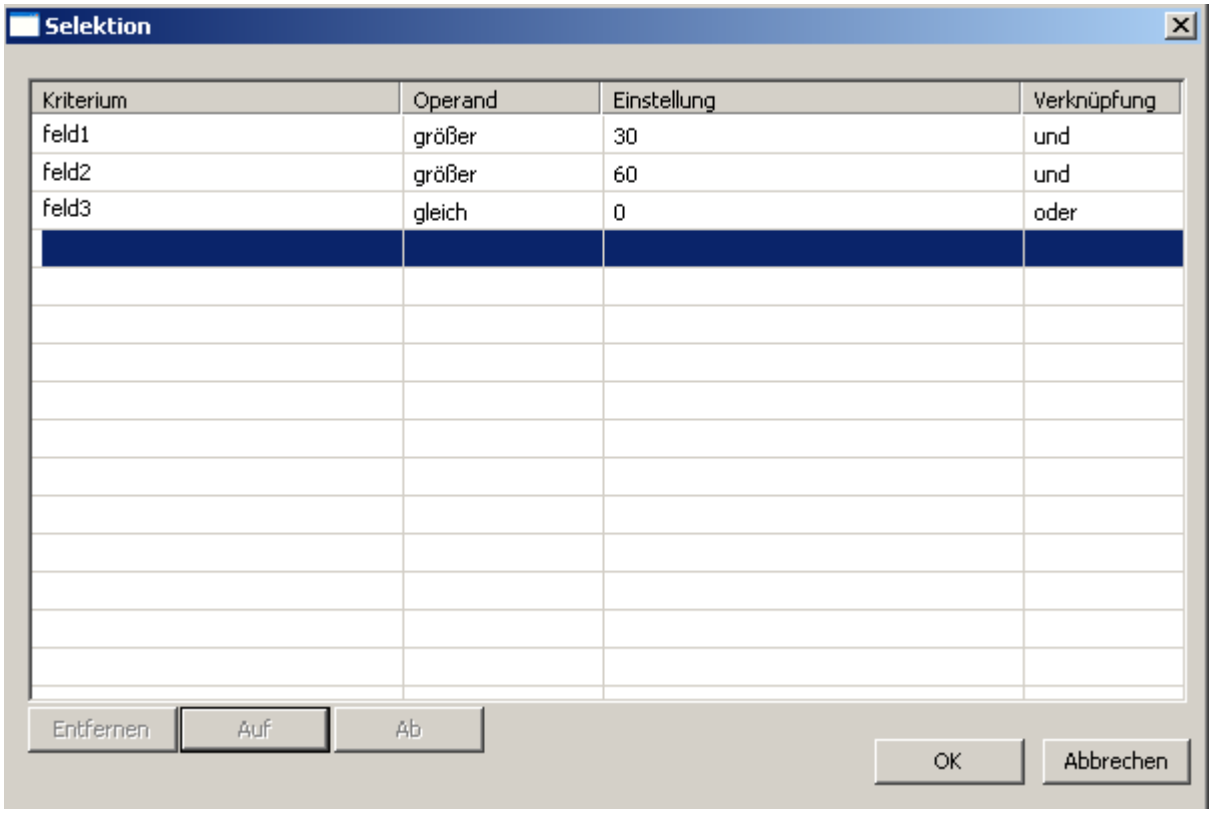

2. Legen Sie die Kriterien für Anzeige fest. Nähere Informationen zur Selektion von Spalten erfahren Sie unter Daten selektieren (Seite 1266) .
3. Beenden Sie den Selektionsdialog mit "OK". Die Selektion wird beim Start von Runtime in der Tabelle der Rezepturanzeige berücksichtigt.

Projektierung der Sortierung von Spalten

Im Bereich "Sortierung" projektieren Sie die Sortierung der Spalten in der Tabelle. Sie können auch in Runtime über Tastenfunktionen die Sortierkriterien festlegen.

1. Klicken Sie auf die Schaltfläche "Bearbeiten...". Der Sortier-Dialog öffnet sich.

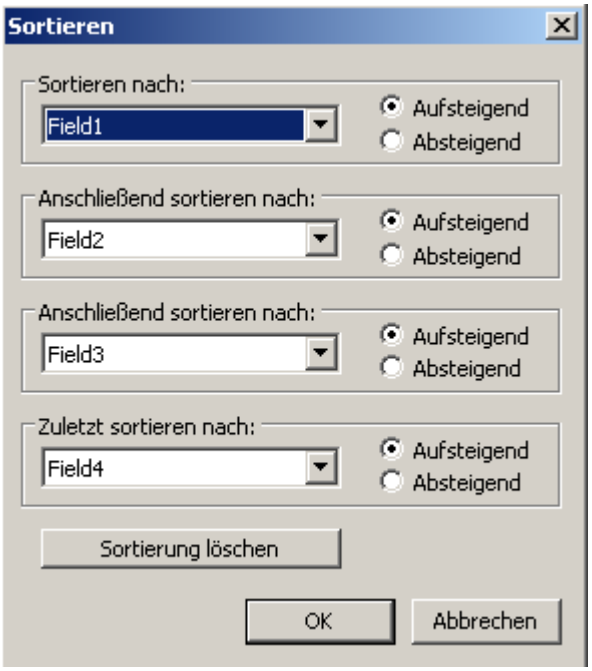

2. Legen Sie eine Sortierabfolge fest.
3. Beenden Sie den Sortier-Dialog mit "OK".
4. Speichern Sie die Projektierung der Inhalte.

Siehe auch

Daten selektieren (Seite 1266)

6.2.5.6 Daten selektieren

Einleitung

Welche Inhalte der Rezeptur Sie in der Tabelle der Rezepturanzeige darstellen oder exportieren wollen, können Sie in Runtime über den Selektionsdialog festlegen. Im Selektionsdialog definieren Sie Auswahlkriterien bezüglich der angezeigten Spalten der Rezeptur.

Voraussetzung

- Der Selektions-Dialog ist geöffnet (siehe Abschnitt "Daten auswählen und konfigurieren (Seite 1263)").

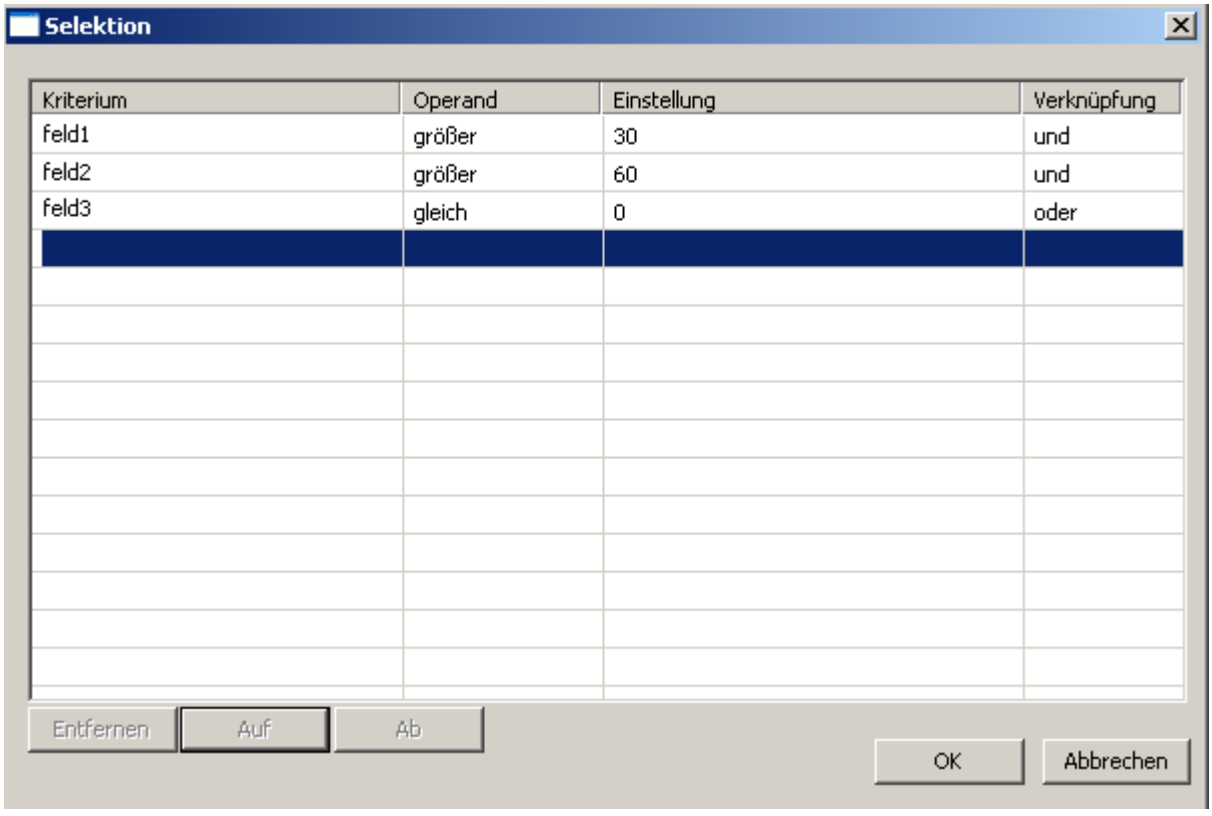

Vorgehensweise

Gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Spalte "Kriterium" in die erste leere Zeile. Die Liste mit den Spalten der Rezepturanzeige wird angezeigt. Wählen Sie die gewünschte Spalte aus, z. B. "feld1".
2. Doppelklicken Sie in die Spalte "Operand", um einen Operand auszuwählen.
3. Doppelklicken Sie in die Spalte "Einstellung", um einen Vergleichswert einzugeben.
4. Doppelklicken Sie in die Spalte "Verknüpfung", um eine "AND" oder "OR"-Verknüpfung auszuwählen.

5. Wiederholen Sie das Vorgehen, wenn Sie weitere Kriterien festlegen wollen.
6. Beenden Sie den Selektionsdialog mit "OK". Die Selektion wird in der Tabelle der Rezepturanzeige angezeigt.

Hinweis

Anzeige der Inhalte der Spalten sicherstellen

Achten Sie auf die richtige Verwendung der Einstellungen und Verknüpfung von Kriterien.

Falsche Verknüpfungen können dazu führen, dass in der Rezepturanzeige Daten der verbundenen Rezeptur nicht angezeigt werden.

Testen Sie deshalb vor der Verknüpfung jedes Kriterium separat und danach jeweils die verknüpften Kriterien. Prüfen Sie, dass auch in Kombination alle erwarteten Inhalte angezeigt werden.

Damit stellen Sie sicher, dass die Selektion in der Rezepturanzeige vollständig angezeigt wird.

Siehe auch

Daten auswählen und konfigurieren (Seite 1263)

6.2.5.7 Daten sortieren

Einleitung

Die Daten in der Rezepturanzeige können Sie in Runtime nach Spalten sortieren. Sie sortieren die Spalten entweder über die Tastenfunktion "Sortier-Dialog" oder direkt über die Überschrift der Spalten.

Hinweis

Sie können bereits bei der Projektierung in der Rezepturanzeige die Sortierkriterien festlegen, indem Sie auf der Registerkarte "Spalten" unter "Sortierung" auf die Schaltfläche "Bearbeiten..." klicken.

So sortieren Sie über den Sortier-Dialog

Voraussetzung

- Sie haben die Schaltfläche "Sortier-Dialog" auf der Registerkarte "Symbolleiste" des UserArchiveControl aktiviert.

Vorgehensweise

1. Klicken Sie auf die Schaltfläche "Sortier-Dialog" .
2. Wählen Sie aus der Liste "Sortieren nach" die Spalte der verbundenen Rezeptur aus, nach dem zuerst sortiert wird. Klicken Sie in ein Optionskästchen, um eine aufsteigende oder absteigende Sortierung festzulegen. Wenn Sie nach mehreren Spalten sortieren wollen, wählen Sie aus den Listen von "Anschließend sortieren nach" die weiteren Spalten in der gewünschten Reihenfolge aus.

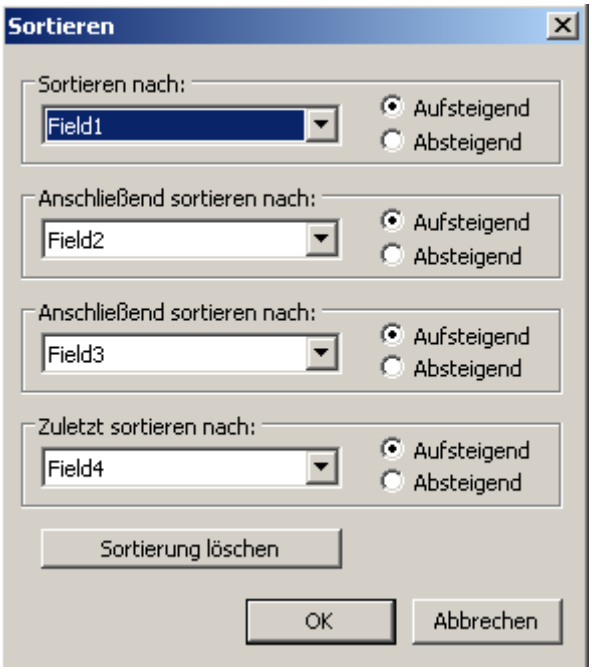

So sortieren Sie die Spalteninhalte über die Spaltenüberschrift

Beim Sortieren über die Spaltenüberschrift haben Sie die Möglichkeit eine Sortierreihenfolge für mehr als vier Spalten festzulegen. Ein Sortiersymbol und ein Sortierindex, rechtsbündig in der Spaltenüberschrift dargestellt, zeigen Sortierabfolge und Sortierreihenfolge der Spalteninhalte an.

Voraussetzung

- Sie haben auf der Registerkarte "Parameter" der Rezepturanzeige das Sortieren im Listenfeld "Sortieren über Spaltenüberschrift" per Klick oder Doppelklick zugelassen.
- Sie haben die Optionskästchen "Sortiersymbol anzeigen" und "Sortierindex anzeigen" aktiviert.

Vorgehensweise

1. Klicken Sie in die Spaltenüberschrift, dessen Spalte Sie als erste Spalte sortieren wollen. Der Sortierindex "1" wird angezeigt mit dem Sortiersymbol nach oben für die aufsteigende Sortierung.
2. Wenn Sie absteigend sortieren wollen, klicken Sie erneut in die Spaltenüberschrift.
3. Wenn Sie die Sortierabfolge mit "auf/ab/keine" festgelegt haben, können Sie mit einem dritten Klick die Sortierung der Spalte aufheben.
4. Wenn Sie mehrere Spalten sortieren wollen, klicken Sie in der gewünschten Reihenfolge in die jeweilige Überschrift der Spalte.

6.2.6 Beispiele

6.2.6.1 Beispiel: Rezepturen erstellen

Aufgabe

In diesem Beispiel erstellen Sie zwei Rezepturen.

- Eine Rezeptur "Beverage" für Getränke. In der Getränkemischanlage werden verschiedene Getränke produziert. Für jeden Artikel erstellen Sie jeweils einen Rezepturdatensatz.
- Eine Rezeptur "Orders" für Bestellungen. In den Bestellungen sind die aktuellen Bestelldaten hinterlegt.

Einstellungen

Die Einstellungen beziehen sich auf ein Bediengerät, das an eine SIMATIC S7-300 oder SIMATIC S7-400 gekoppelt ist.

Für das Beispiel benötigen Sie folgende Variablen, Rezepturen, Rezepturelemente und Rezepturdatensätze:

Variablen:

Name	Steuerungsanbindung	Adresse	Typ
article	Ja	DB 120, DBW 0	String
litrewater	Ja	DB 120, DBW 4	Integer
kilosugar	Ja	DB 120, DBW 8	Integer
gramaroma	Ja	DB 120, DBW 12	Integer
litreconcentrate	Ja	DB 120, DBW 16	Integer
coloring	Ja	DB 120, DBW 20	Integer
customer	Ja	DB 120, DBW 24	String
numbercustomer	Ja	DB 120, DBW 28	String
numberorder	Ja	DB 120, DBW 32	Integer
quantitiy	Ja	DB 120, DBW 36	Integer

Name	Steuerungsanbindung	Adresse	Typ
filling	Ja	DB 120, DBW 40	Integer
dateorder	Ja	DB 120, DBW 44	Date
status	Ja	DB 120, DBW 48	Integer

Rezeptur "Beverage":

Rezepturelement	Verbundene Variable
Article	article
Water	litrewater
Sugar	kilosugar
Aroma	gramaroma
Concentrate	litreconcentrate
Coloring	coloring

Datensatz ID	1	2	3
Article	Juice	Nectar	Cola Light
Water	5	50	72
Sugar	5	10	35
Aroma	100	300	240
Concentrate	95	50	35
Coloring	FS 0063	FS 0063	FS 1007

Rezeptur "Orders":

Rezepturelement	Verbundene Variable
Customer	customer
Number	numbercustomer
Order	numberorder
Article	article
Quantitiy	quantitiy
Filling	filling
Status	status
Date	dateorder

Vorgehensweise

1. Erstellen Sie die oben aufgeführten Variablen mit den oben genannten Einstellungen.
2. Erstellen Sie die Rezepturen "Beverage" und "Orders" mit den oben genannten Einstellungen.
3. Wählen Sie für die Rezepturen die folgenden Eigenschaften:
   – Unter "Allgemein > Größe" den Typ "Unbegrenzt"
   – Unter "Kommunikation > Kommunikationsart" die Kommunikationsart "Variablen"

4. Legen Sie in jeder Rezeptur die Rezepturelemente an.

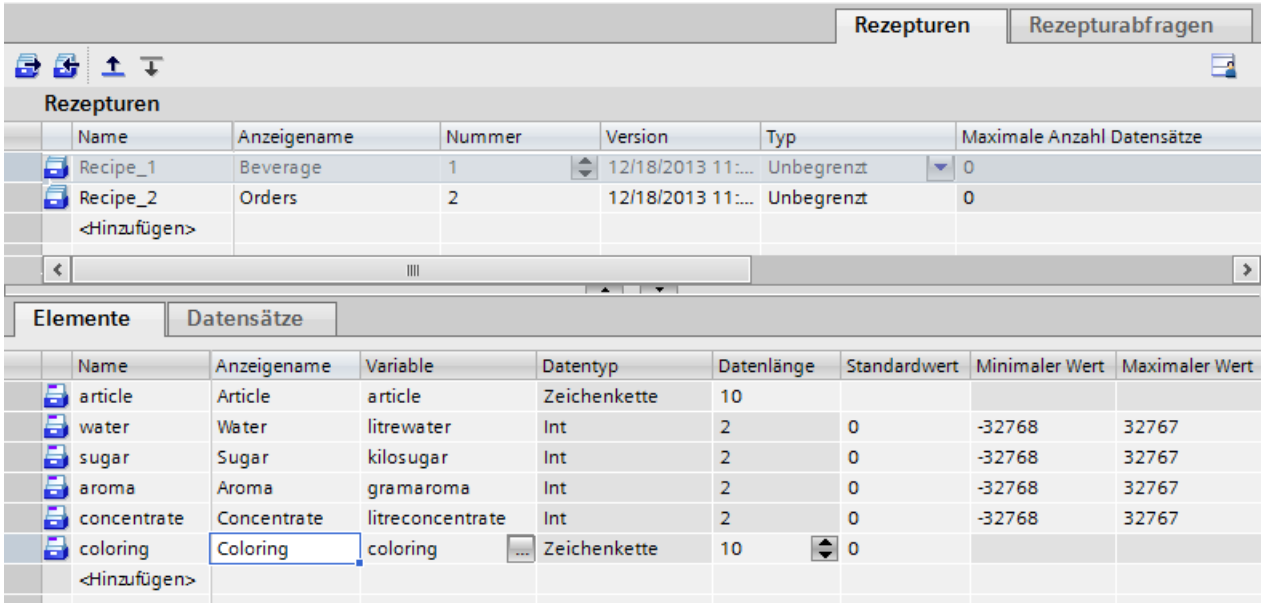

5. Aktivieren Sie für folgende Elemente im Eigenschaftsfenster unter "Extras" das Kontrollkästchen "Eindeutige Werte":
   - In der Rezeptur "Beverage" für das Rezepturelement "Article"
   - In der Rezeptur "Orders" für das Rezepturelement "Order"
6. Legen Sie in der Rezeptur "Beverage" die oben genannten Datensätze an. Geben Sie in jeden Datensatz die oben genannten Werte ein.

Ergebnis

Die Rezepturen "Beverage" und "Orders" sind angelegt.

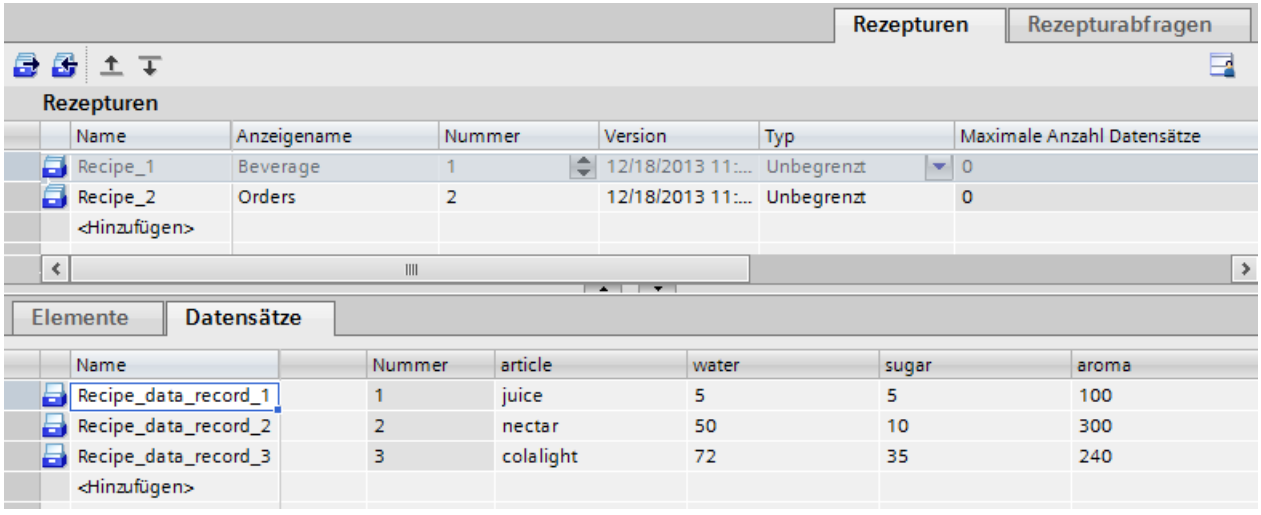

Siehe auch

Beispiel: Rezepturanzeige projektieren (Seite 1274)

6.2.6.2 Beispiel: Rezepturabfrage erstellen

Aufgabe

In diesem Beispiel erstellen Sie die Rezepturabfrage "Label". In der Rezepturabfrage werden folgende Rezepturelemente angezeigt:

Rezepturelement	Rezeptur
Article	Beverage, Orders
Customer	Orders
Status	Orders
Order	Orders
Filling	Orders
Sugar	Beverage
Coloring	Beverage

Das Rezepturelement "Article" ist in beiden Rezepturen als gemeinsames Element enthalten.

Voraussetzungen

Die Rezepturen "Beverage" und "Orders" sind erstellt.

Vorgehensweise

1. Erstellen Sie die Rezepturabfrage "Label". Legen Sie die oben aufgeführten Elemente an.

2. Geben Sie unter "Eigenschaften > Relationen" folgende Bedingungen ein:
   Beverage.Article = Orders.Article

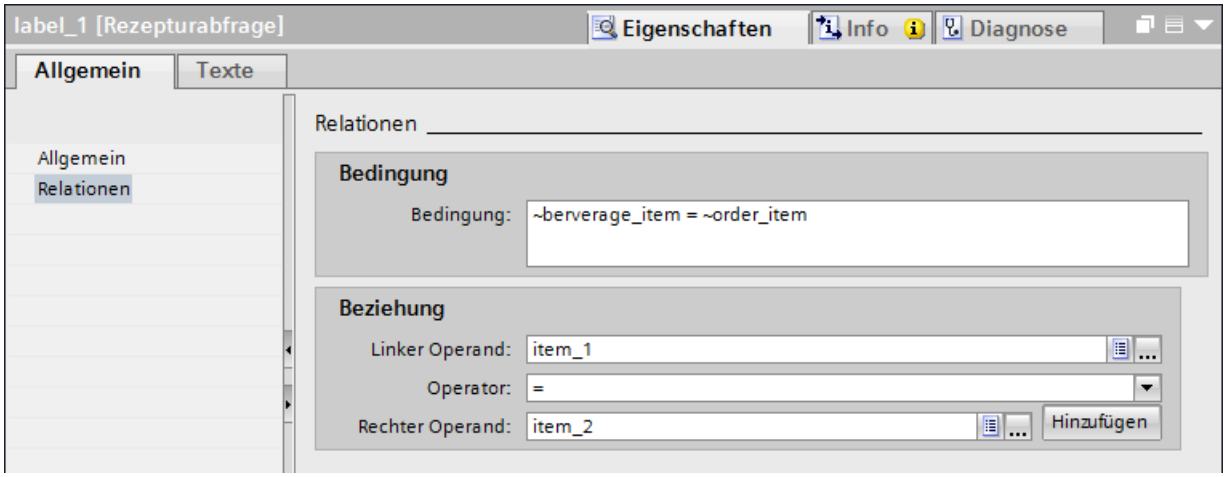

Ergebnis

Die Rezepturabfrage "Label" ist angelegt.

6.2.6.3 Beispiel: Rezepturanzeige projektieren

Aufgabe

In diesem Beispiel fügen Sie ein Anlagenbild mit Rezepturanzeige auf die Sicht "Etikett" ein.

Voraussetzung

Die Beispiele "Rezepturen erstellen" und "Rezepturabfrage erstellen" sind erstellt.

Das Prozessbild "Getränkemischanlage" ist angelegt und geöffnet.

Vorgehensweise

1. Fügen Sie die Rezepturanzeige in das Prozessbild ein. Die Rezepturanzeige finden Sie im Werkzeugfenster unter "Controls".
2. Wählen Sie Inspektorfenster in der Gruppe "Allgemein > Anzeige" im Eingabefeld "Datenquelle" über den Pfeil den Typ "Rezepturabfrage".
3. Wählen Sie die Rezepturabfrage "Etikett" aus.
4. Legen Sie im Inspektorfenster in der Gruppe "Eigenschaften" > "Gestaltung" zusätzliche Anzeigeoptionen der Rezepturanzeige fest.
5. Wechseln Sie im Inspektorfenster in die Gruppe "Eigenschaften" > "Spalten".
   - Aktivieren Sie für alle Rezepturelemente die Kontrollkästchen in den Spalten "Sichtbar" und "Geschützt".
6. Wechseln Sie im Inspektorfenster in die Gruppe "Eigenschaften" > "Symbolleiste". Aktivieren Sie unter "Symbolleiste - Schaltflächen" für folgende Schaltflächen die Kontrollkästchen in der Spalte "Sichtbar".
   - "Erste Zeile"
   - "Vorhergehende Zeile"
   - "Nächste Zeile"
   - "Letzte Zeile"
   - "Sortier-Dialog"
   - "Selektions-Dialog"
7. Wechseln Sie im Inspektorfenster in die Gruppe "Eigenschaften" > "Spalten". Geben Sie in der Spalte "Sortierung" folgende Sortierung ein:
   - Nach Auftragsnummer
   - Aufsteigend

Ergebnis

In Runtime wird auf im Prozessbild "Getränkemischanlage" die Rezepturanzeige mit der Rezepturabfrage "Etikett" geöffnet. Die Datensätze werden sortiert nach Auftragsnummer angezeigt. Sie können mit den Schaltflächen zwischen den Datensätzen wählen.

Über das Datenfeld "Status" können Sie sich den aktuellen Auftrag oder bereits abgearbeitete Aufträge anzeigen lassen.

Mit den Schaltflächen "Sortier-Dialog" und "Selektions-Dialog" können Sie die Rezepturanzeige der Datensätze nach Ihren Bedürfnissen anpassen.

Siehe auch

Beispiel: Rezepturen erstellen (Seite 1270)

6.2.6.4 Verwendung von Steuervariablen

Beispiel zur Verwendung der Steuervariablen:

Um in dem Beispiel mit den Steuervariablen zu arbeiten, müssen Sie die folgenden Schritte ausführen:

Voraussetzung

Die Rezeptur "Getränke" ist erstellt.

Eigenschaften der Rezeptur "Cola"	
Archivtyp	"Unbegrenzt"
Kommunikation	"Variablen"
Datentyp für Steuervariable "ID"	Long
Datentyp für Steuervariable "Auftrag"	Long
Datentyp für Steuervariable "Feld"	String (1 Zeichen)
Datentyp für Steuervariable "Wert"	String (1 Zeichen)

Vorgehensweise

1. Legen Sie die Steuervariablen mit folgenden Eigenschaften an:

Steuervariable	Datenformat	Ausgabeformat
ID	Dezimal	0999
Auftrag	Dezimal	s9
Feld	String	*
Wert	String	*

1. Wählen Sie für jede Steuervariable die Objekteigenschaft "Aktualisierung bei Änderung".
2. Verbinden Sie die Steuervariablen mit der Rezeptur "Getränke".
3. Erstellen Sie ein neues Bild und fügen Sie eine Rezepturanzeige ein.
   – Wählen Sie die Rezeptur "Getränke".
   – Aktivieren Sie unter "Allgemein" unter "Edit" die Schaltkästchen "Einfügen", "Ändern" und "Löschen".
4. Legen Sie für jedes projektierte Rezepturelement ein E/A-Feld an.
5. Verbinden Sie die E/A-Felder mit den zugehörigen Variablen.
   Z. B. das E/A-Feld für "Farbstoff" mit der Prozessvariablen "Coloring".
   – Legen Sie zu jedem projektierten E/A-Feld ein Textfeld für die Beschriftung an.
     Damit ordnen Sie in Runtime die E/A-Felder den einzelnen Rezepturelementen zu.

6. Legen Sie für jede Steuervariable ein E/A-Feld an.
   – Legen Sie zu jedem projektierten E/A-Feld ein Textfeld für die Beschriftung an. Damit ordnen Sie in Runtime die E/A-Felder den einzelnen Steuervariablen zu.
7. Aktivieren Sie die Runtime, bzw. laden Sie das Projekt auf das Bediengerät und starten Sie das Projekt.

Ergebnis

Die Projektierung für die Rezeptur ist fertig gestellt.

Zugriff auf die Rezeptur

Sie greifen auf folgende Weise auf die Rezeptur zu:

- Innerhalb der Tabellenansicht der Rezepturanzeige

ID	Artikel	LiterWasser	KiloZucker	GrammAroma	Liter
1	Calif Cola	90	10	15	75
2	Cola	80	30	30	80
3	Cola Light	100	20	50	60

- Über die projektierten E/A-Felder

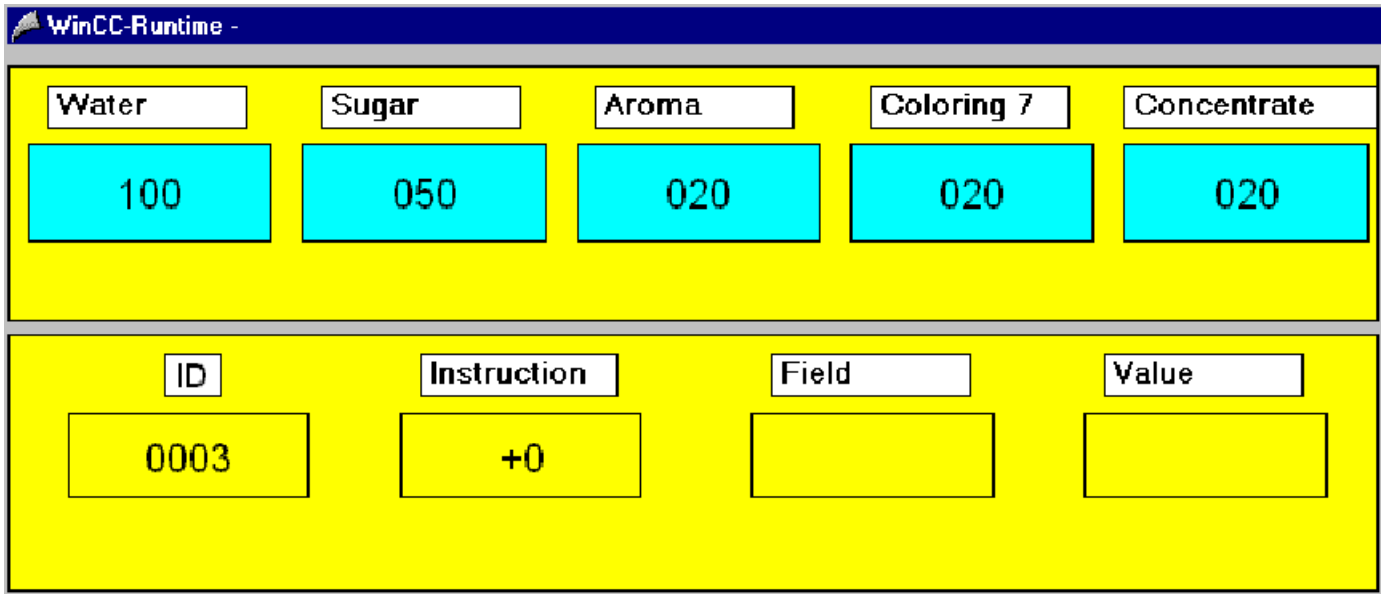

Datensatz über die ID in die Prozessvariablen schreiben

- Geben Sie in das E/A-Feld "ID" die ID "3" und in das E/A-Feld "Auftrag" eine "7" ein.
  Die Werte des Datensatzes "3" werden in die Prozessvariablen der E/A-Felder geschrieben und in den E/A-Feldern angezeigt.
  Wenn die Werte des Datensatzes fehlerfrei aus der Rezeptur ausgelesen wurden, wird im Feld "Auftrag" die Fehlernummer "0" angezeigt.
  Im Fehlerfall wird die Fehlernummer "-1" ausgegeben.
  Die Steuervariablen "Feld" und "Wert" werden nicht benötigt.

  Hinweis

  Durch Eingabe der ID "-1" und den Auftrag "6" werden die aktuellen Inhalte der Prozessvariablen in die Rezeptur eingelesen. Die neuen Werte werden an das Tabellenende angehängt. Die IDs der Datensätze werden fortlaufend hoch gezählt.

Lesen der Prozessvariablen über die ID

- Ändern Sie die Werte in den E/A-Feldern der Prozessvariablen und geben Sie im Feld "ID" eine "5" ein. Im E/A-Feld "Auftrag" geben Sie eine "6" ein.
  Die geänderten Werte der Prozessvariablen werden nun in den Datensatz "5" der Rezeptur geschrieben. Die vorher in diesem Datensatz enthaltenen Werte werden überschrieben.
  Die Steuervariablen "Feld" und "Wert" werden nicht benötigt.

Datensatz über die Steuervariablen "Feld" und "Wert" in die Prozessvariablen schreiben

Hinweis

Bei dem Element, das mit der Steuervariablen "Wert" referenziert wird, müssen Sie unter "Eigenschaften > Speicheroptionen" das Kontrollkästchen "Feld muss einen eindeutigen Wert besitzen" aktivieren. Eine eindeutige Zuordnung des Rezepturdatensatzes zu dem Wert im Element ist sonst nicht möglich.

- Geben Sie im E/A-Feld "Feld" das Wort "Artikel" ein und in das E/A-Feld "Wert" schreiben Sie "Cola Light". Geben Sie im E/A-Feld "Auftrag" eine "7" ein.
  Die Werte des Datensatzes "Cola Light " werden in die Prozessvariablen der E/A-Felder geschrieben und in den E/A-Feldern angezeigt.
  Die Steuervariable "ID" wird nicht benötigt und muss daher auf 0 gesetzt werden.

Mit Protokollen arbeiten 7

7.1 Grundlagen

7.1.1 Protokolle

Einleitung

Um Prozessdaten und abgearbeitete Zyklen der Produktion zu dokumentieren, verwenden Sie Protokolle. Sie haben die Möglichkeit, z. B. regelmäßige Schichtprotokolle zu erstellen, Chargendaten auszugeben oder den Fertigungsprozess für die Qualitätsprüfung zu dokumentieren.

Erstellen von Protokollen

Sie erstellen und bearbeiten Protokolle im Editor "Protokolle". In diesem Editor projektieren Sie für Protokolle Folgendes:

- Formale Gestaltung
  Im Inspektorfenster legen Sie die formale Gestaltung des Protokolls fest. Sie legen z. B. Seitenformat und Seitenränder fest, und ob das Protokoll eine Titelseite, Rückseite, Kopfzeilen oder Fußzeilen hat.
  Für Protokolle für Runtime Professional legen Sie darüber hinaus für die unterschiedlichen Seiten fest, ob sie mit oder ohne ein Wasserzeichen ausgegeben werden.
- Inhalt
  Im Arbeitsbereich legen Sie fest, welche Daten das Protokoll ausgibt, z. B. die Meldungen einer Schicht. Dazu fügen Sie entsprechende Objekte in die Detailseite(n) ein.
  Darüber hinaus gestalten Sie Titelseite, Rückseite, Kopf- und Fußzeilen nach Ihren Wünschen. Für Seiten mit Wasserzeichen gestalten Sie zusätzliche Seiten, die als "Hintergrundbild" mit ausgegeben werden.

Der modulare Aufbau ermöglicht es Ihnen, für jeden Anwendungsfall das passende Protokoll zu projektieren.

Hinweis

Bei Bediengeräten, die das Protokollieren nicht unterstützen, steht der Editor "Protokolle" nicht zur Verfügung.

Protokollausgabe

In Runtime wird ein Protokoll entweder zeit- oder ereignisgesteuert ausgegeben.

- Zeitgesteuert: automatischer Druck an einem bestimmten Termin, zu einer bestimmten Uhrzeit oder in Intervallen.
- Ereignisgesteuert: Druck startet bei Eintreten eines bestimmten Ereignisses, z. B. bei Klick auf eine Schaltfläche oder Überschreiten eines Grenzwerts.

Je nachdem für welche Runtime Sie die Ausgabe eines Protokolls projektieren, gehen Sie unterschiedlich vor.

In Runtime Advanced werden Protokolle auf dem Standarddrucker des Bediengeräts ausgegeben.

Für Runtime Professional legen Sie mithilfe eines Druckauftrags Folgendes fest:

- Ob nur einige Seiten oder alle Seiten des Protokolls ausgegeben werden
- Ob alle Daten ausgegeben werden oder nur die Daten eines bestimmten Zeitraums
- Ob das Protokoll auf einem Drucker oder in eine Datei ausgegeben wird
- Ob der Bediener vor dem eigentlichen Drucken einen Drucker auswählen oder Einstellungen wie den Seitenumfang ändern darf
- Ob das auszugebende Protokoll in der Druckauftragsliste eines entsprechend projektierten Applikationsfensters angezeigt wird

Druckfunktion von Bildobjekten

In einigen Bildobjekten der Runtime Professional, z. B. der Meldeanzeige, ist die Protokollausgabe über eine Schaltfläche (Druckersymbol) standardmäßig enthalten. WinCC verwendet für die Ausgabe vordefinierte Systemprotokolle, die Sie bei Bedarf an Ihre Bedürfnisse anpassen oder durch ein individuell gestaltetes Protokoll ersetzen.

Siehe auch

Aufbau von Protokollen (Seite 1281)

Protokoll erstellen (Seite 1294)

Aufbau von Protokollen (Seite 1283)

Grundlagen zur Protokollerstellung (Seite 1290)

Grundlagen zur Protokollausgabe (Seite 1309)

Grundlagen zu Systemprotokollen und Systemdruckaufträgen (Seite 1316)

7.1.2 Aufbau von Protokollen

Einleitung

Ein Protokoll in WinCC besteht aus mehreren Abschnitten, die Sie je nach Bedarf aktivieren oder deaktivieren.

Abschnitte eines Protokolls

Das folgende Bild zeigt die möglichen verschiedenen Abschnitte eines Protokolls im Editor "Protokolle":

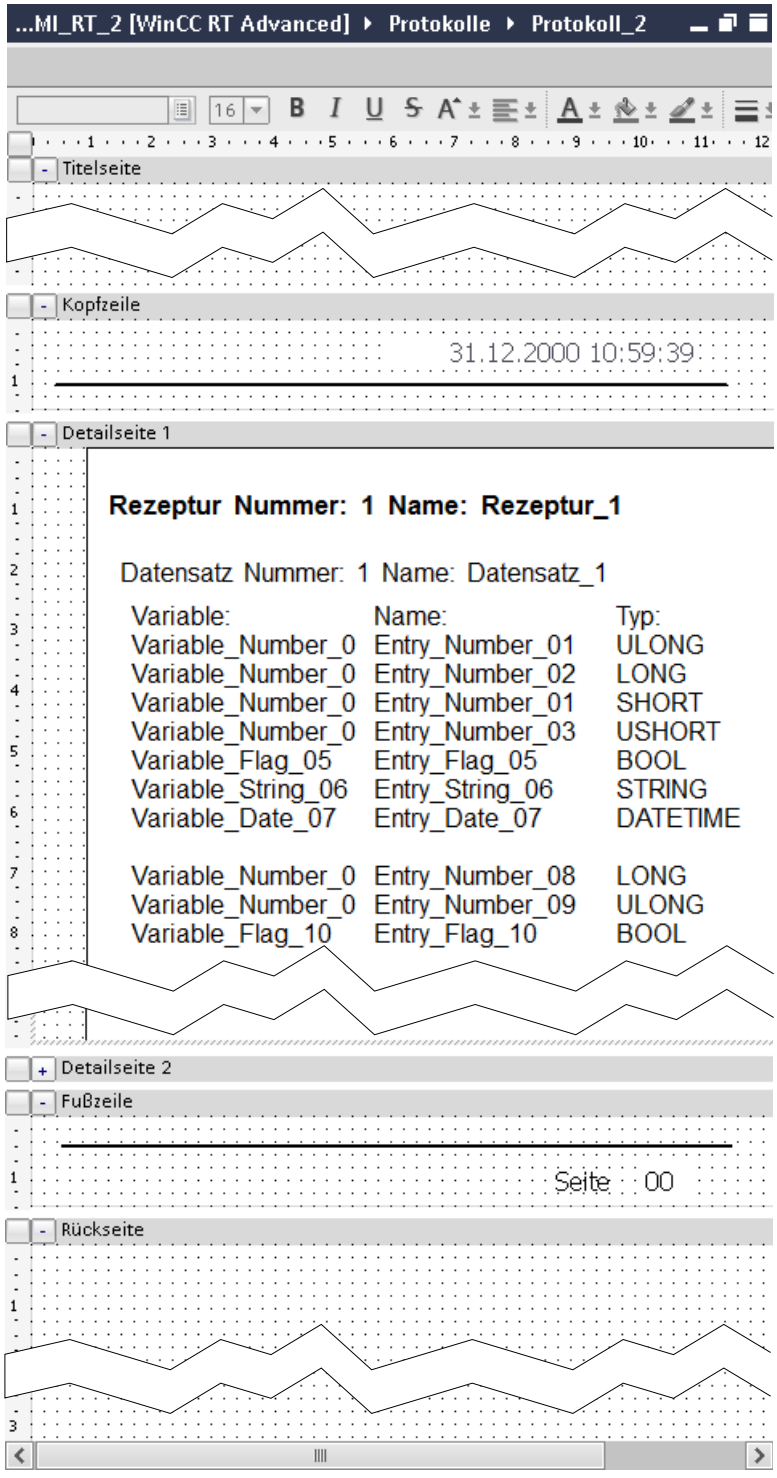

Titelseite und Rückseite

Auf der Titelseite tragen Sie aussagekräftige Informationen zum Protokollinhalt ein. Die Rückseite verwenden Sie z. B. als Impressum, auf der Sie Kontaktinformationen von Schichtleitern oder Servicetechnikern bereitstellen. Titelseite und Rückseite werden jeweils als eine Seite ausgegeben. Sie sind jeweils genau eine Seite lang und werden nicht umgebrochen.

Detailseite

Auf den Detailseiten des Protokolls projektieren Sie die Ausgabe der Runtime-Daten, z. B. ein Rezepturprotokoll oder Meldeprotokoll.

Um weitere Detailseiten einzufügen oder die Reihenfolge der Detailseiten zu ändern, verwenden Sie das Kontextmenü der Detailseite.

Kopfzeile und Fußzeile

Die Kopfzeile und Fußzeile wird auf jeder Detailseite des Protokolls ausgegeben. In der Kopfzeile oder Fußzeile fügen Sie typischerweise Seitennummern oder das Datum ein.

Siehe auch

Protokoll erstellen (Seite 1294)

Protokolle (Seite 1279)

Grundlagen zur Protokollerstellung (Seite 1287)

7.1.3 Aufbau von Protokollen

Einleitung

Ein Protokoll in WinCC besteht aus mehreren Abschnitten, die Sie je nach Bedarf aktivieren oder deaktivieren.

Abschnitte eines Protokolls

Das folgende Bild zeigt die möglichen verschiedenen Abschnitte eines Protokolls im Editor "Protokolle":

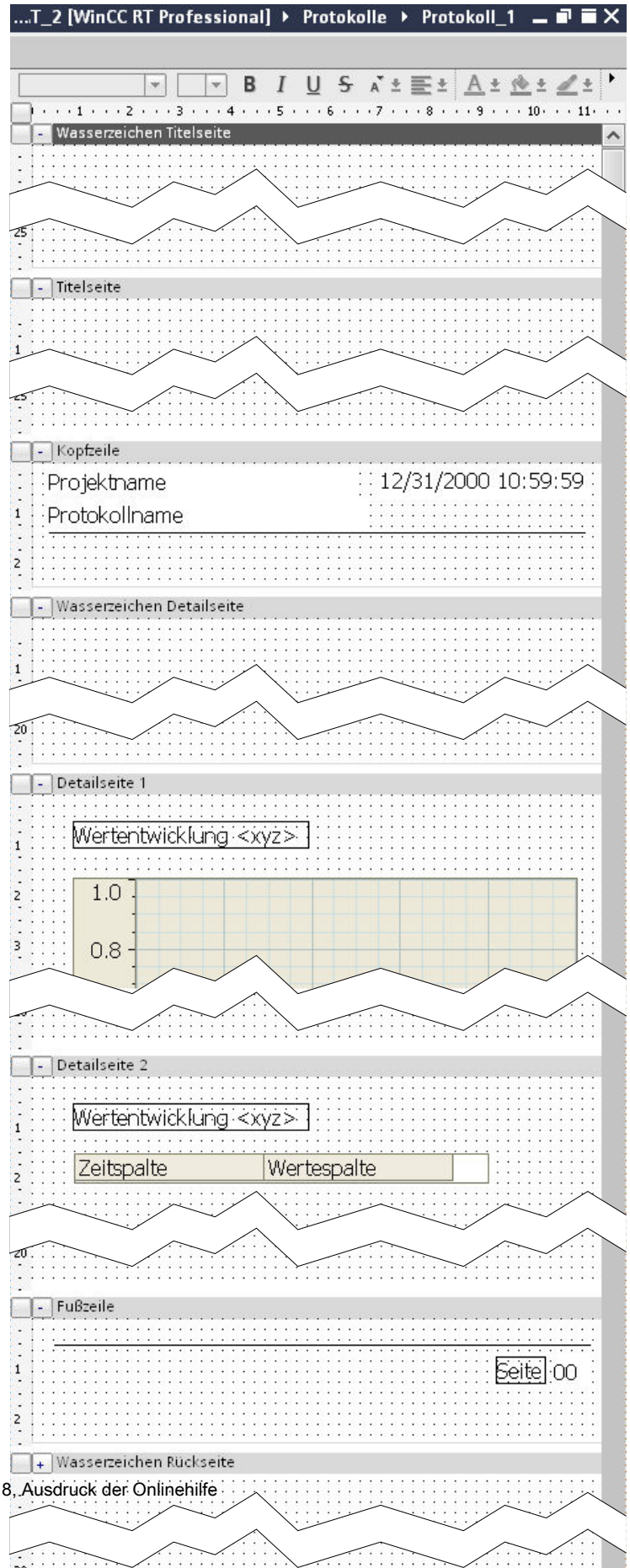
...T_2 [WinCC RT Professional] ▸ Protokolle ▸ Protokoll_1
Wasserzeichen Titelseite
Titelseite
Kopfzeile
Projektname
12/31/2000 10:59:59
Protokollname
Wasserzeichen Detailseite
Detailseite 1
Wertentwicklung <xyz>
1.0
0.8
Detailseite 2
Wertentwicklung <xyz>
Zeitspalte
Wertespalte
Fußzeile
Seite 00
Wasserzeichen Rückseite

Titelseite und Rückseite

Auf der Titelseite tragen Sie aussagekräftige Informationen zum Protokollinhalt ein. Die Rückseite verwenden Sie z. B. als Impressum, auf der Sie Kontaktinformationen von Schichtleitern oder Servicetechnikern bereitstellen. Titelseite und Rückseite werden jeweils als eine Seite ausgegeben. Sie sind jeweils genau eine Seite lang und werden nicht umgebrochen – auch nicht, wenn Sie z. B. ein Objekt mit dynamischer Länge auf die Seite projektieren.

Detailseite

Auf den Detailseiten des Protokolls projektieren Sie die Ausgabe der Runtime-Daten, z. B. ein Rezepturprotokoll oder Meldeprotokoll.

Um weitere Detailseiten einzufügen oder die Reihenfolge der Detailseiten zu ändern, verwenden Sie das Kontextmenü der Detailseite.

Kopfzeile und Fußzeile

Die Kopfzeile und Fußzeile wird auf jeder Detailseite des Protokolls ausgegeben. In der Kopf- und Fußzeile fügen Sie typischerweise Projektname, Protokollname, Seitennummern und das Datum ein.

Wasserzeichen

Titelseite, Detailseite und Rückseite können mit einem Wasserzeichen ausgegeben werden. Seite und zugehöriges Wasserzeichen werden übereinander gedruckt.

Mithilfe eines Wasserzeichens geben Sie ein Objekt "A" neben einem Objekt von variabler Länge "B" aus, ohne dass das Objekt "A" mit verschoben wird. Beispiel: Ein Ausgabefeld (A) soll auf jeder Seite neben einer Tabelle (B) ausgegeben werden.

Titel- und Rückseite sowie die Detailseiten haben jeweils ein separates Wasserzeichen.

Siehe auch

Protokolle (Seite 1279)

Grundlagen zur Protokollerstellung (Seite 1290)

Protokoll erstellen (Seite 1294)

7.2 Arbeiten mit Protokollen

7.2.1 Erstellen von Protokollen

7.2.1.1 Grundlagen zur Protokollerstellung

Einleitung

Ein Protokoll in WinCC besteht aus mehreren Abschnitten, die Sie je nach Bedarf aktivieren oder deaktivieren.

- Titelseite
- Rückseite
- Detailseiten
- Fuß- und Kopfzeile für Detailseiten

Hinweis

Ihnen stehen bis zu 35 Detailseiten pro Protokoll zur Verfügung.

Vorgehensweise

Das folgende Bild zeigt die allgemeine Vorgehensweise beim Erstellen eines Protokolls:

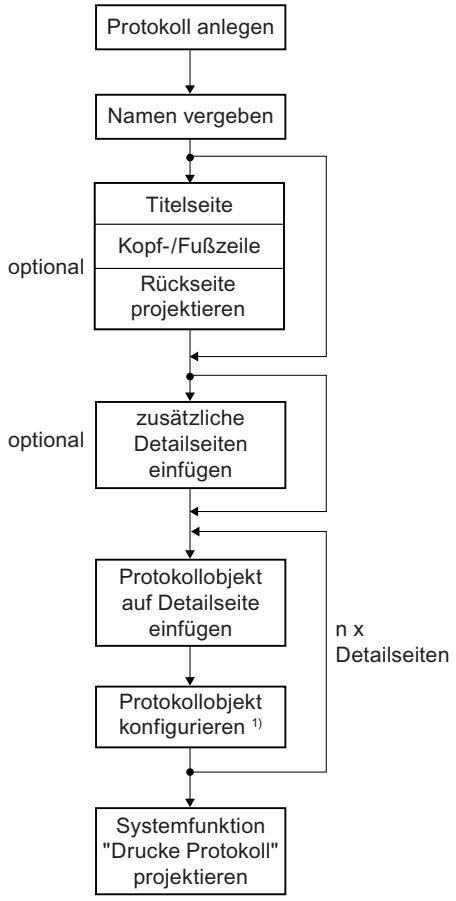

1) Einstellungen abhängig vom verwendeten Protokollobjekt.

Werkzeuge zur Gestaltung von Protokollen

Die Werkzeuge zur Gestaltung von Protokollen finden Sie in der Task Card "Werkzeuge".

- Zur *grafischen Gestaltung* fügen Sie in den verschiedenen Abschnitten des Protokolls "Basisobjekte", "Elemente" und "Grafiken" ein. Zum Beispiel für ein Logo in der Kopfzeile, Trennlinien und eine Seitenzahl in der Fußzeile.
- Die *Ausgabe der Runtime-Daten* projektieren Sie in den Detailseiten mithilfe der "Controls".

Hinweis

Zum Gestalten eines Protokolls stehen Ihnen viele Objekte des Editors "Bilder" zur Verfügung – jedoch nur mit einem eingeschränkten Funktionsumfang. Eigenschaften zur Bedienung sind nicht verfügbar. E/A-Felder in Protokollen geben z. B. ausschließlich Daten aus.

Position und Größe von Objekten

Platzieren Sie die Objekte an der Position, an der sie im Protokoll ausgegeben werden sollen. Objekte werden in der projektierten Größe ausgegeben – mit folgender Besonderheit:

Auf Detailseiten projektieren Sie "Rezepturprotokolle" oder "Meldeprotokolle", die Runtime-Daten in Tabellenform ausgeben.

- Die *Breite* der Tabelle wird automatisch an die Breite der Detailseite angepasst. Sie können die Breite nicht ändern.
- WinCC passt die *Höhe* der Tabelle bei der Ausgabe des Protokolls automatisch an die Menge der auszugebenden Daten an. Die Tabelle wird gegebenenfalls auf einer Folgeseite fortgesetzt.
  In der Detailseite verlängert WinCC Tabellen mit dynamischer Länge deshalb automatisch bis zum unteren Rand der Detailseite.

Das folgende Bild zeigt ein Beispiel für eine Detailseite im Protokoll und die Ausgabe in Runtime:

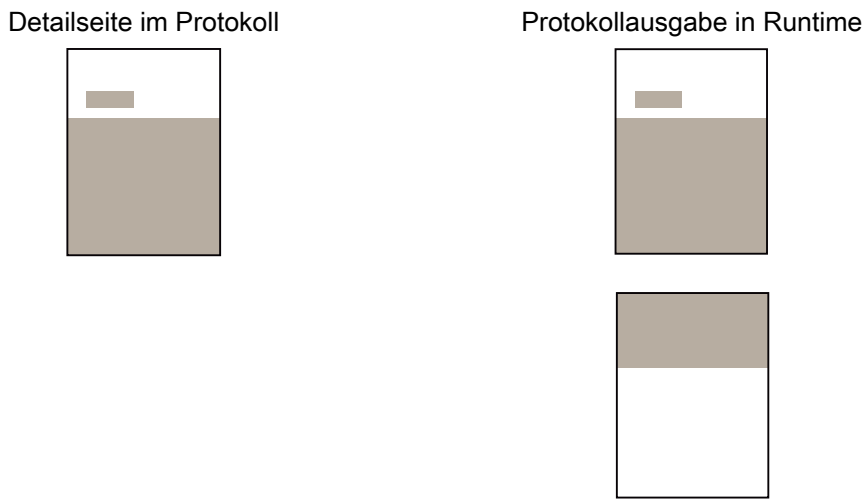

Zusätzliche Detailseiten

Um weitere Detailseiten einzufügen oder die Reihenfolge der Detailseiten zu ändern, verwenden Sie das Kontextmenü der Detailseite.

Siehe auch

Protokollieren von Meldungen projektieren (Seite 909)

Aufbau von Protokollen (Seite 1281)

Protokoll erstellen (Seite 1294)

Detailseiten verwalten (Seite 1306)

Objekte in Protokollen (Seite 1336)

7.2.1.2 Grundlagen zur Protokollerstellung

Einleitung

Ein Protokoll in WinCC besteht aus mehreren Abschnitten, die Sie je nach Bedarf aktivieren oder deaktivieren.

- Titelseite
- Rückseite
- Detailseiten
- Fuß- und Kopfzeile für Detailseiten
- Wasserzeichen für Titelseite, Rückseite, Detailseite

Hinweis

Ihnen stehen bis zu 35 Detailseiten pro Protokoll zur Verfügung.

Vorgehensweise

Das folgende Bild zeigt die allgemeine Vorgehensweise beim Erstellen eines Protokolls:

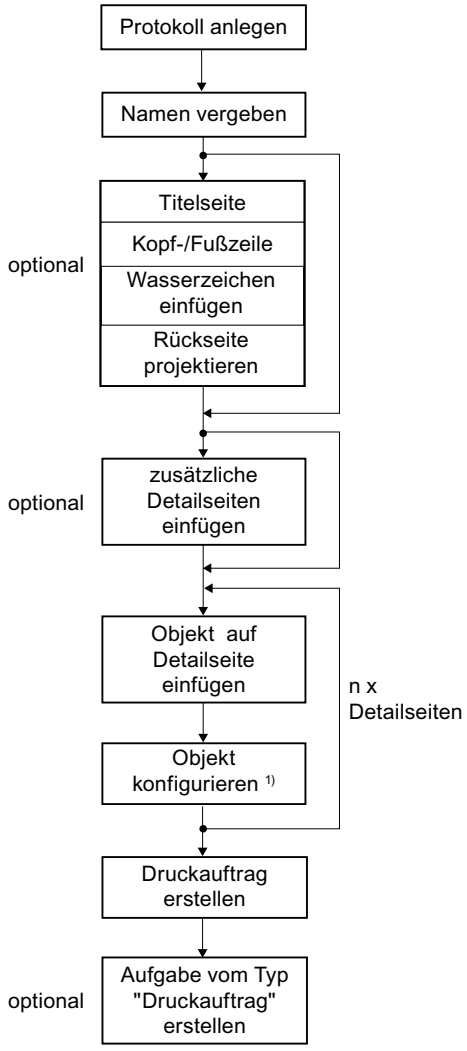

1) Einstellungen abhängig vom verwendeten Protokollobjekt.

Werkzeuge zur Gestaltung von Protokollen

Die Objekte zur Gestaltung von Protokollen finden Sie in der Task Card "Werkzeuge".

Hinweis

Zum Gestalten eines Protokolls stehen Ihnen viele Objekte des Editors "Bilder" zur Verfügung – jedoch nur mit einem eingeschränkten Funktionsumfang. Eigenschaften zur Bedienung sind nicht verfügbar. E/A-Felder in Protokollen geben z. B. ausschließlich Daten aus.

Positionierung von Objekten

Platzieren Sie die Objekte an der Position, an der sie im Protokoll ausgegeben werden sollen. Beachten Sie jedoch, dass WinCC Weißräume am Anfang der *Detailseite* bei der Ausgabe des Protokolls ignoriert. Das oberste Objekt markiert den Anfang der Detailseite im Ausdruck.

Das folgende Bild zeigt ein Beispiel für eine Detailseite im Protokoll und die Ausgabe in Runtime:

Detailseite im Protokoll

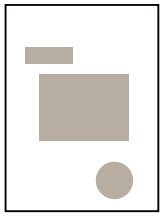

Protokollausgabe in Runtime

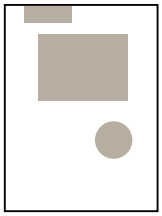

Wenn Sie einen Weißraum mit ausgeben wollen:, vergrößern Sie die Kopfzeile entsprechend. Alternativ können Sie auf der Detailseite ein "unsichtbares" Objekt an den gewünschten Anfang positionieren. Zum Beispiel eine weiße Linie, die auf weißem Papier nicht sichtbar ist.

Größe von Objekten

Objekte werden in der projektierten Größe ausgegeben.

Auf Detailseiten projektieren Sie "Controls" und "Protokollobjekte" für die Ausgabe der Runtime-Daten. WinCC passt die Runtime-Daten in den Rahmen des Objekts ein – mit einer Besonderheit: Wenn Sie ein *Objekt wie eine Tabelle mit dynamischer Länge* projektieren, passt WinCC die Höhe der Tabelle automatisch an die Menge der zu druckenden Daten an. Die Tabelle wird gegebenenfalls auf einer Folgeseite fortgesetzt. WinCC passt nur die Höhe des Objekts an und platziert alle nachfolgenden Objekte unterhalb des Objekts – so wie in der Detailseite projektiert.

Das folgende Bild zeigt ein Beispiel für eine Detailseite im Protokoll und die Ausgabe in Runtime:

Detailseite im Protokoll

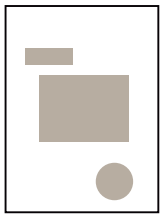

Protokollausgabe in Runtime

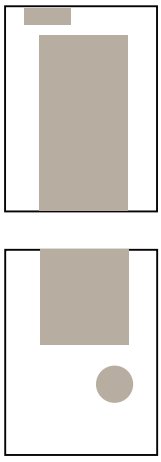

In der Regel haben die "Controls" und "Protokollobjekte" eine dynamische Länge, die Runtime-Daten in Tabellenform ausgeben. Dazu gehören folgende Objekte:

- Rezepturprotokoll
- Meldeprotokoll
- Tabellenanzeige
- ODBC-Datenbanktabelle
- Meldefolgeprotokoll
- CSV-Provider Tabellen
- Control Hardcopy
- Control Daten
- Control Informationen

Objekte neben dynamischen Objekten

Platzieren Sie in einer Detailseite neben Objekten mit dynamischer Länge keine anderen Objekte. Da die Position der Objekte nicht eindeutig zu bestimmen ist, gibt WinCC die Objekte möglicherweise nicht korrekt aus. Projektieren Sie die Objekte stattdessen als Wasserzeichen.

Das folgende Bild zeigt ein Beispiel für ein Wasserzeichen und eine Detailseite im Protokoll sowie die Ausgabe in Runtime:

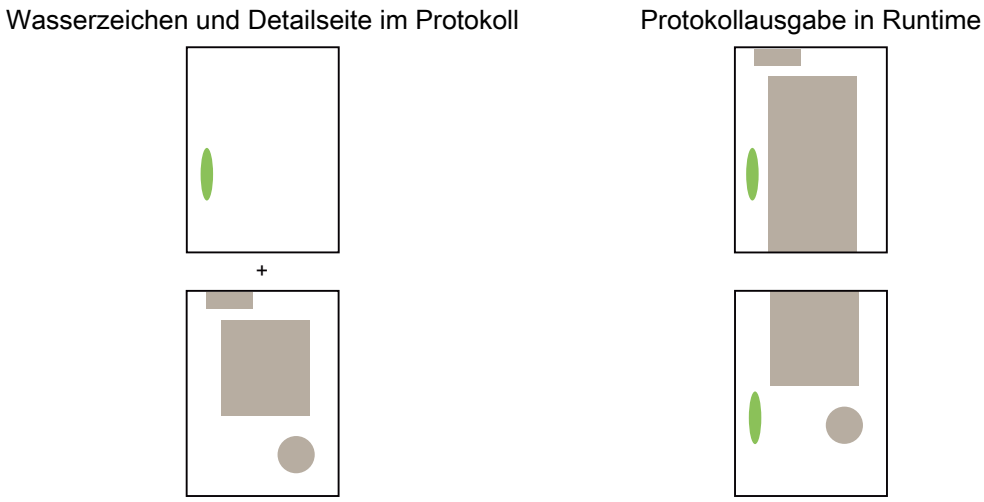

Titelseite und Rückseite

Titelseite und Rückseite werden jeweils als *eine* Seite ausgegeben. Wenn Sie ein Objekt mit dynamischer Länge auf die Titelseite oder Rückseite projektieren, setzt WinCC die Ausgabe nicht auf einer Folgeseite fort. Wenn das Objekt in Runtime mehr Daten enthält, als auf einer Seite ausgegeben werden können, werden nicht alle Daten im Protokoll ausgegeben.

Zusätzliche Detailseiten

Um weitere Detailseiten einzufügen oder die Reihenfolge der Detailseiten zu ändern, verwenden Sie das Kontextmenü der Detailseite.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

CSV-Datei zur Ausgabe von Kurven (Seite 1301)

CSV-Datei zur Ausgabe einer Tabelle (Seite 1303)

Fortlaufende Protokollierung von Meldungen (Seite 1300)

Protokolle (Seite 1279)

Aufbau von Protokollen (Seite 1283)

Protokoll erstellen (Seite 1294)

Detailseiten verwalten (Seite 1306)

Grundlagen zur Protokollausgabe (Seite 1309)

Objekte in Protokollen (Seite 1336)

7.2.1.3 Protokoll erstellen

Voraussetzung

Ein Projekt mit einem Bediengerät ist geöffnet.

Vorgehen

Um ein Protokoll zu erstellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter "Protokolle" auf "Neues Protokoll hinzufügen".
   Ein neues leeres Protokoll wird im Editor "Protokolle" angezeigt.
2. Wählen Sie im Kontextmenü des Protokolls den Befehl "Eigenschaften Protokoll".

3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" den Aufbau des Protokolls fest, z. B. ob Sie im Protokoll "Titelseite", "Rückseite", "Kopfzeile" und "Fußzeile" projektieren wollen.
   Die Abschnitte des Protokolls werden entsprechend aktualisiert.

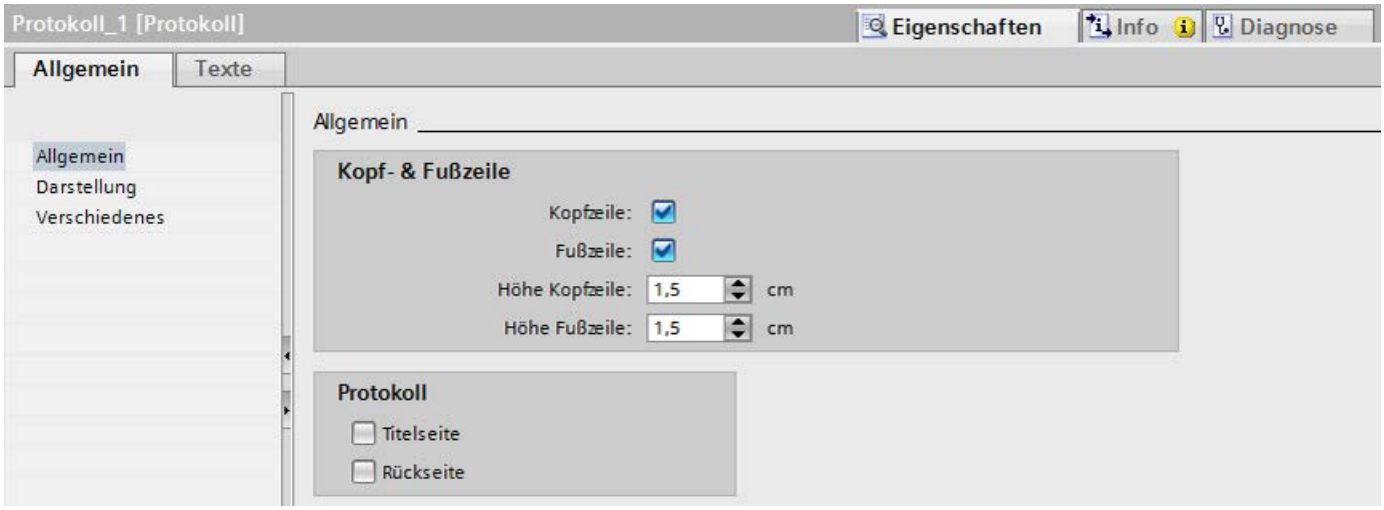

4. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Darstellung" das Format, die Seitenausrichtung und die Seitenränder des Protokolls.

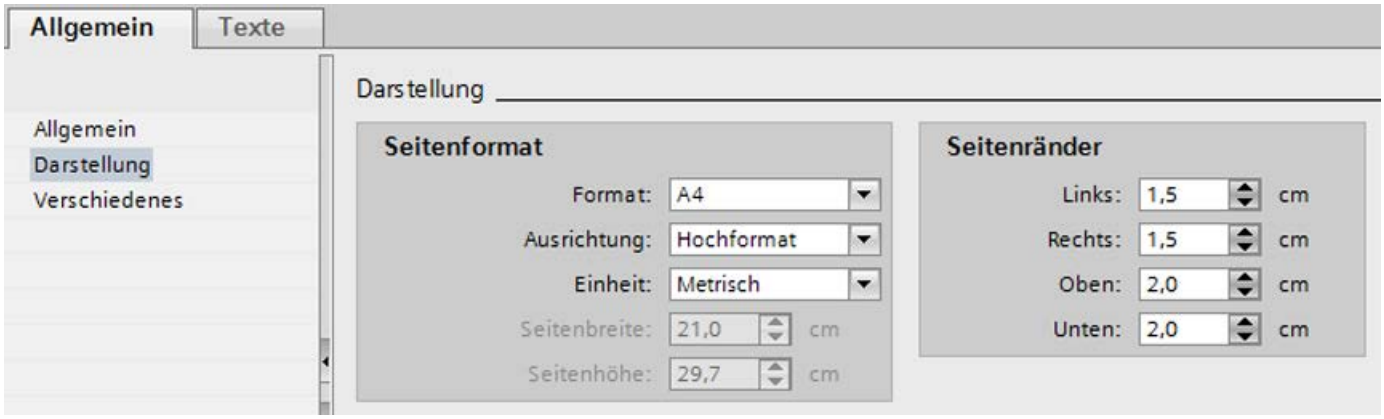

5. Tragen Sie unter "Eigenschaften > Eigenschaften > Verschiedenes" einen aussagekräftigen "Namen" für das Protokoll ein.
6. Gestalten Sie die einzelnen Abschnitte des Protokolls wie gewünscht.
   Ziehen Sie dazu die benötigten Basisobjekte, Elemente, Grafiken und Controls per Drag&Drop aus der Task Card "Werkzeuge" an die gewünschte Position.
   Alternativ können Sie auch bereits konfigurierte Objekte aus einem Bild in das Protokoll kopieren oder verschieben.
7. Konfigurieren Sie die Objekte im Inspektorfenster.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

CSV-Datei zur Ausgabe von Kurven (Seite 1301)

CSV-Datei zur Ausgabe einer Tabelle (Seite 1303)

Fortlaufende Protokollierung von Meldungen (Seite 1300)

Protokollieren von Meldungen projektieren (Seite 909)

Protokolle (Seite 1279)

Aufbau von Protokollen (Seite 1281)

Aufbau von Protokollen (Seite 1283)

Grundlagen zur Protokollerstellung (Seite 1290)

Arbeiten mit Objekten (Seite 1320)

Objekte in Protokollen (Seite 1336)

7.2.1.4 Werkzeuge zur Protokollgestaltung

Werkzeuge zur Gestaltung von Protokollen

Im Editor "Protokolle" verwenden Sie zur Gestaltung von Protokollen die Objekte der Task Card "Werkzeuge":

- Zur *grafischen Gestaltung* fügen Sie in den verschiedenen Abschnitten des Protokolls "Basisobjekte", "Elemente" und "Grafiken" ein. Zum Beispiel für ein Logo in der Kopfzeile, Trennlinien und eine Seitenzahl in der Fußzeile.
- Die *Ausgabe der Runtime-Daten* projektieren Sie in den Detailseiten mithilfe der "Controls".
- Wenn Sie anstelle eines Systemprotokolls für die Druckfunktion eines Bildobjekts eigene Protokolle verwenden wollen, projektieren Sie die Ausgabe der Runtime-Daten mithilfe der "Protokollobjekte".

Hinweis

Zum Gestalten eines Protokolls stehen Ihnen viele Objekte des Editors "Bilder" zur Verfügung – jedoch nur mit einem eingeschränkten Funktionsumfang. Eigenschaften zur Bedienung sind nicht verfügbar. E/A-Felder in Protokollen geben z. B. ausschließlich Daten aus.

Elemente

Die folgende Tabelle gibt einen Überblick über die Objekte des Panels "Elemente":

Objekt	Verwendung
Datum/Uhrzeit-Feld	Zur Ausgabe des aktuellen Datums und der aktuellen Uhrzeit Sie können das Ausgabeformat individuell festlegen.
E/A-Feld	Zur Ausgabe von Prozesswerten Verschiedene Ausgabeformate stehen zur Auswahl.
Projektname	Zur Ausgabe des Projektnamens Der Name kann mit und ohne Pfadangabe ausgegeben werden.

Objekt	Verwendung
Protokollname	Zur Ausgabe des Protokollnamens
Seitenzahl	Zur Ausgabe der Seitenzahl und der Gesamtanzahl Seiten Sie können das Ausgabeformat individuell festlegen.

Ausgabe von Daten in Tabellenform

Für eine zeitgesteuerte oder ereignisgesteuerte Ausgabe von Daten in Tabellenform projektieren Sie in Protokollen folgende Controls:

Control	Verwendung im Protokoll
Tabellenanzeige	Zur Ausgabe der Werte einer oder mehrerer Archivvariablen in Abhängigkeit von der Zeit Jede Tabellenzeile repräsentiert dabei die Werte der Archivvariablen zu einem bestimmten Zeitpunkt.
ODBC-Datenbanktabelle	Zur Ausgabe von Daten, die nicht aus WinCC stammen Die Daten werden über die ODBC-Schnittstelle als Tabelle ausgegeben.
CSV-Provider Tabellen	Zur Ausgabe von Daten einer CSV-Datei Die Daten der CSV-Datei müssen einer bestimmten Struktur entsprechen. Sie können eine CSV-Datei auswählen oder die Auswahl der CSV-Datei mit einer Variablen dynamisieren.

Im Editor "Bilder" ist in das Control "Tabellenanzeige" bereits eine Druckfunktion integriert (Schaltfläche in Runtime). Die "Tabellenanzeige" verwendet zum Drucken vorkonfigurierte Systemprotokolle. In den Systemprotokollen werden entweder alle Daten oder nur die aktuell im Objekt angezeigten Daten ausgegeben (Schnappschuss). Systemprotokolle passen Sie entweder an ihre Bedürfnisse an oder Sie verwenden anstelle der Systemprotokolle individuell gestaltete Protokolle.

Ausgabe von Daten als Kurven

Für eine zeitgesteuerte oder ereignisgesteuerte Ausgabe von Daten als Kurven projektieren Sie in Protokollen folgende Controls:

Controls	Verwendung im Protokoll
f(t)-Kurvenanzeige in Protokollen	Zur Ausgabe von Archivvariablen als Funktion der Zeit
f(x)-Kurvenanzeige in Protokollen	Zur Ausgabe von Archivvariablen als Funktion anderer Archivvariablen, z. B. der Temperaturverlauf in Abhängigkeit vom Druck Zusätzlich können Sie die Kurve mit einer Sollwertkurve vergleichen.
CSV-Provider Kurven	Zur Ausgabe von Daten einer CSV-Datei Die Daten der CSV-Datei müssen einer bestimmten Struktur entsprechen. Sie können eine CSV-Datei auswählen oder die Auswahl der CSV-Datei mit einer Variablen dynamisieren.

Im Editor "Bilder" ist in den Controls "f(t)-Kurvenanzeige" und "f(x)-Kurvenanzeige" bereits eine Druckfunktion integriert (Schaltfläche in Runtime). Diese Controls verwenden zum Drucken vorkonfigurierte Systemprotokolle, die Sie an Ihre Bedürfnisse anpassen können.

Ausgabe von Rezepturen

Für eine zeitgesteuerte oder ereignisgesteuerte Ausgabe von Rezepturdaten projektieren Sie in Protokollen folgendes Control:

Control	Verwendung im Protokoll
Rezepturanzeige	Zur Ausgabe von Rezepturelementen

Im Editor "Bilder" ist in dem Control "Rezepturanzeige" bereits eine Druckfunktion integriert (Schaltfläche in Runtime). Dieses Control verwendet zum Drucken vorkonfigurierte Systemprotokolle. In den Systemprotokollen werden entweder alle Daten oder nur die aktuell im Objekt angezeigten Daten ausgegeben (Schnappschuss). Systemprotokolle passen Sie entweder an ihre Bedürfnisse an oder Sie verwenden anstelle der Systemprotokolle individuell gestaltete Protokolle.

Ausgabe von Meldungen

WinCC bietet mehrere Möglichkeiten für die Protokollierung von Meldungen.

Für eine zeitgesteuerte oder ereignisgesteuerte Ausgabe von Meldungen projektieren Sie in Protokollen folgendes Control:

Control	Verwendung im Protokoll
Meldeanzeige	Zur Ausgabe von Meldungen, die während des Prozesses in einer Anlage auftreten Je nach Konfiguration gibt WinCC z. B. aktuelle Meldungen, unterdrückte Meldungen oder Meldungen aus dem Kurzzeitarchiv aus.

Im Editor "Bilder" ist in dem Control "Meldeanzeige" bereits eine Druckfunktion integriert (Schaltfläche in Runtime). Dieses Control verwendet zum Drucken vorkonfigurierte Systemprotokolle. In den Systemprotokollen werden entweder alle Meldungen oder nur die aktuell im Objekt angezeigten Meldungen ausgegeben (Schnappschuss). Systemprotokolle passen Sie entweder an ihre Bedürfnisse an oder Sie verwenden anstelle der Systemprotokolle individuell gestaltete Protokolle.

WinCC bietet darüber hinaus die Möglichkeit, Meldungen nach Start von Runtime fortlaufend auszugeben, z. B. auf einem Zeilendrucker. Die Ausgabe müssen Sie nur aktivieren. WinCC bietet zwei vorkonfigurierte Systemprotokolle, die Sie an Ihre Bedürfnisse anpassen: eines für die Ausgabe im Zeilenlayout und eines für die Ausgabe im Seitenlayout.

Ausgabe eines Schnappschusses von Runtime

Um einen Schnappschuss von Runtime in einem Protokoll auszugeben, projektieren Sie in Protokollen folgendes Control:

Control	Verwendung im Protokoll
Hardcopy	Zur Ausgabe eines Schnappschusses von Runtime Je nach Konfiguration gibt WinCC den Inhalt des gesamten Bildschirms, des aktiven Fensters oder eines individuell festgelegten Bildschirmbereichs aus.

Ausgabe von Daten über die ODBC-Schnittstelle

Um Daten von externen Quellen über die ODBC-Schnittstelle auszulesen und in einem Protokoll auszugeben, projektieren Sie in Protokollen folgende Controls.

Control	Verwendung im Protokoll
ODBC-Datenbankfeld	Zur Ausgabe von Daten aus einem Feld einer externen Datenbanktabelle
ODBC-Datenbanktabelle	Zur Ausgabe von externen Datenbanktabellen

Protokollobjekte

In einigen Bildobjekten ist die Protokollausgabe über eine Schaltfläche (Druckersymbol) standardmäßig enthalten, z. B. in der Meldeanzeige.

In Protokollen, die Sie anstelle eines vorkonfigurierten Systemprotokolls zum Drucken eines Bildobjekts verwenden wollen, projektieren Sie die Ausgabe der Runtime-Daten mitHilfe der "Protokollobjekte".

Die folgende Tabelle zeigt, welche Protokollobjekte Sie in Protokollen verwenden können:

Objekt	Verwendung im Protokoll
Control Hardcopy	Zur Ausgabe des Bildobjekts, so wie es in Runtime dargestellt ist (Schnappschuss)
Control Daten	Zur Ausgabe der Daten, die aktuell im Bildobjekt enthalten sind Die Daten werden in Tabellenform ausgegeben.
Control Informationen	Zur Ausgabe der Datenquelle, z. B. Fenstertitel, Objektname oder Name des Bildes

Siehe auch

Protokoll erstellen (Seite 1294)

Grundlagen zur Protokollerstellung (Seite 1290)

Control Daten (Seite 1337)

Control Hardcopy (Seite 1338)

Control Informationen (Seite 1339)

CSV-Provider Kurven (Seite 1340)

CSV-Provider Tabellen (Seite 1341)

Datum/Uhrzeit-Feld (Seite 1343)

f(t)-Kurvenanzeige in Protokollen (Seite 1346)

f(x)-Kurvenanzeige in Protokollen (Seite 1347)

Hardcopy (Seite 1350)

Meldeanzeige in Protokollen (Seite 1351)

ODBC-Datenbankfeld (Seite 1354)

ODBC-Datenbanktabelle (Seite 1356)

Projektname (Seite 1358)

Protokollname (Seite 1359)

Rezepturanzeige in Protokollen (Seite 1360)

Seitenzahl (Seite 1363)

Tabellenanzeige in Protokollen (Seite 1365)

Fortlaufende Protokollierung von Meldungen (Seite 1300)

7.2.1.5 Fortlaufende Protokollierung von Meldungen

Meldefolgeprotokoll

Ein Meldefolgeprotokoll gibt am Bediengerät auftretende Meldungen fortlaufend in chronologischer Reihenfolge aus, z. B. auf einem Zeilendrucker.

Die Verwendung des vorkonfigurierten Meldefolgeprotokolls müssen Sie nur aktivieren.

Wenn Sie z. B. nur bestimmte Meldungen ausgeben wollen, passen Sie das Protokoll an Ihre Bedürfnisse an.

Aktivierung und Konfiguration

Die Ausgabe des vorkonfigurierten Meldefolgeprotokolls ist in dem Systemdruckauftrag "Alarm sequence report" projektiert.

Um den Systemdruckauftrag bei Runtime-Start automatisch mit zu starten:

- Aktivieren Sie in den Runtime-Einstellungen des Bediengeräts "Dienste > Startsequenz von WinCC Runtime > Meldefolgeprotokoll".

Das Meldefolgeprotokoll wird standardmäßig auf einem Zeilendrucker im Zeilenlayout ausgegeben.

Um das Meldeprotokoll im Seitenlayout auszugeben:

- Öffnen Sie den Systemdruckauftrag "Alarm sequence report" und deaktivieren Sie unter "Eigenschaften > Eigenschaften > Allgemein" die Option "Zeilenprotokoll verwenden".

WinCC weist dem Systemdruckauftrag das entsprechende Systemprotokoll zu.

Bei Bedarf passen Sie die Systemprotokolle an Ihre Bedürfnisse an:

- "@AlarmSequenceReportForPagePrinter" für eine Ausgabe im Seitenlayout
- "@AlarmSequenceReportForLinePrinter" für eine Ausgabe im Zeilenlayout

Siehe auch

Grundlagen zur Protokollerstellung (Seite 1290)

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

Meldefolgeprotokoll (Seite 1352)

Grundlagen zu Systemprotokollen und Systemdruckaufträgen (Seite 1316)

Meldeanzeige in Protokollen (Seite 1351)

7.2.1.6 CSV-Datei zur Ausgabe von Kurven

Anforderungen an eine CSV-Datei für die Ausgabe von Kurven

Für die Ausgabe einer Tabelle müssen die Daten in der CSV-Datei in einer festgelegten Struktur vorliegen.

Das Objekt "CSV-Provider Kurven" stellt die Daten aus einer CSV-Datei im Protokoll als Kurven dar. Die Daten der CSV-Datei müssen der nachfolgend beschriebenen Struktur entsprechen.

Die CSV-Datei muss für die verschiedenen Teile des Grafen einen Absatz enthalten. Ein Absatz wiederum kann mehrere Zeilen enthalten. Folgende Absätze sind vorgeschrieben:

- "Graph": Enthält die Anzahl der Kurven und Angaben zur Darstellung des Grafen
- "Kurven": Enthält Angaben zur Kurvendarstellung
- "Kurvenwerte": Enthält die Werte der Kurven

Alle Absätze sind gleich aufgebaut. Die 1. Zeile markiert den Anfang eines Absatzes und beschreibt, wie die Daten des Absatzes interpretiert werden. Die anderen Zeilen eines Absatzes enthalten die eigentlichen Werte des Grafen oder der Kurven.

Trennen Sie die Werte mit einem Semikolon voneinander ab. Um das Ende einer Zeile zu markieren, fügen Sie Zeilenumbrüche ein.

Absatz "Graph"

Die 1. Zeile des Absatzes "Graph" muss wie folgt lauten:

#Trend_T; Name; Curves; DateFrom; DateTo; Common Y-Axis; Font; Fontsize

Die 2. Zeile muss die Werte für den Graph enthalten.

Name	Bedeutung	Anmerkung
Name	Name der Kurvenanzeige oder der Datei	
Curves	Anzahl der Kurven	
DateFrom	Anfang des Zeitbereichs	Notation: 2000-10-30 10:15:00.000
DateTo	Ende des Zeitbereichs	Notation: 2000-10-30 10:15:00.000
Common Y-Axis	Gemeinsame y-Achse	
Font	Schriftart	
Fontsize	Schriftgröße	

Beispiel:

Zeile	Inhalt
1	#Trend_T; Name; Curves; DateFrom; DateTo; Common Y-Axis; Font; Fontsize
2	"TrendControl1";3;"2001-10-10 16:30:00.000";"2001-10-10 16:40:00.000";0;"Arial";10

Absatz "Kurven"

Die 1. Zeile des Absatzes "Kurven" muss wie folgt lauten:

#Curve; Num; Name; Count; dMin; dMax; Color; Width; CurveType; Filling

Für jede Kurve, die im Grafen dargestellt wird, muss dieser Absatz eine separate Zeile mit den Werten für die jeweilige Kurve enthalten.

Name	Bedeutung	Anmerkung
Num	Nummer der Kurve	beginnend mit 0
Name	Name der Kurve	
Count	Anzahl der Werte	
dMin	Untergrenze der Kurve (zur Skalierung)	
dMax	Obergrenze der Kurve (zur Skalierung)	
Color	Farbe der y-Achse	in hexadezimaler Notation
Width	Linienstärke in Points	z .B. 1.5
CurveType	Kurventyp	LINE, DOTS, STEP
Filling	Füllfarbe für Flächen	0=nein, 1=ja

Beispiel:

Zeile	Inhalt
1	#Curve; Num; Name; Count; dMin; dMax; Color; Width; CurveType; Filling
2	0;"Temperatur";3;10;30;0x00ff0000;1;STEP;0
3	1;"Druck";5;0;50;0x0000ff00;2;DOTS;0
...	

Struktur Datensatz "Kurvenwerte"

Die 1. Zeile des Absatzes "Kurvenwerte" muss wie folgt lauten:

#Data; Num; Date; Value; Flags; Color

Für jeden Wert, der in den Kurven dargestellt wird, muss dieser Absatz eine separate Zeile mit den jeweiligen Koordinaten enthalten.

Name	Bedeutung	Anmerkung
Num	Nummer der Kurve	beginnend mit 0
Date	Wert auf der x-Achse der Kurve	Notation: 2000-10-30 10:15:00.000
Value	Wert auf der y-Achse der Kurve	

Name	Bedeutung	Anmerkung
Flags	Grenzwerte / Zeitüberlappung	
Color	Farbe des Kurvenabschnitts	In hexadezimaler Notation Wenn Sie für die Kurve keine Farbe festlegen, wird die Farbe der zugehörigen y-Achse übernommen.

Beispiel:

Zeile	Inhalt
1	#Data; Num; Date; Value; Flags; Color
2	0;"2001-10-10 16:30:00.000";22;0;0x000000FF
3	0;"2001-10-10 16:31:00.000";24;0;
...	

Siehe auch

Protokoll erstellen (Seite 1294)

Grundlagen zur Protokollerstellung (Seite 1290)

CSV-Provider Kurven (Seite 1340)

7.2.1.7 CSV-Datei zur Ausgabe einer Tabelle

Anforderungen an eine CSV-Datei für die Ausgabe als Tabelle

Das Objekt "CSV-Provider Tabelle" gibt die Daten aus einer CSV-Datei im Protokoll als Tabelle aus. Die Daten der CSV-Datei müssen der nachfolgend beschriebenen Struktur entsprechen.

Die CSV-Datei muss für die verschiedenen Teile der Tabelle einen Absatz enthalten. Ein Absatz wiederum kann mehrere Zeilen enthalten. Folgende Absätze sind vorgeschrieben:

- "Tabelle": Enthält die Anzahl der Spalten sowie Schriftart und -größe
- "Spalten": Enthält die Spaltenüberschriften, Spaltenbreiten und Ausrichtungen
- "Daten": Enthält die Daten

Alle Absätze sind gleich aufgebaut. Die 1. Zeile markiert den Anfang eines Absatzes. Die 1. Zeile beschreibt, wie die Daten des Absatzes interpretiert werden. Die anderen Zeilen eines Absatzes enthalten die eigentlichen Werte des Grafen oder der Kurven.

Trennen Sie die Werte mit einem Semikolon voneinander ab. Um das Ende einer Zeile zu markieren, fügen Sie Zeilenumbrüche ein.

Absatz "Tabelle"

Die 1. Zeile des Absatzes "Tabelle" muss wie folgt lauten:

#Table; Name; Columns; Font; Fontsize

Die 2. Zeile muss die Werte für die Tabelle enthalten.

Name	Bedeutung	Anmerkung
Name	Name der Tabellenanzeige	
Columns	Anzahl der Spalten	
Font	Schriftart	
Fontsize	Schriftgröße	

Beispiel:

Zeile	Inhalt
1	#Table; Name; Columns; Font; Fontsize
2	"Tab_9";4;"Arial";14

Absatz "Spalten"

Die 1. Zeile des Absatzes "Spalten" muss wie folgt lauten:

#Column; Num; Header; Width; Alignment

Für jede Spalte der Tabelle muss dieser Absatz eine separate Zeile mit den Werten für die jeweilige Spalte enthalten.

Name	Bedeutung	Anmerkung
Num	Nummer der Spalte	beginnend mit 0
Header	Spaltenüberschrift	
Width	Breite der Spalte im Verhältnis zu den anderen Spalten	Beispiel: Für eine Tabelle mit 3 Spalten legen Sie folgende Breiten fest: 2, 4 und 8. Als Ergebnis wird die 3. Spalte doppelt so breit wie die 2. Spalte und drei mal so breit wie die 1. Spalte ausgegeben. Spaltenbreite in Zeichen = [Tabellenbreite in Anzahl Zeichen] x [Width] / [Summe der Width-Werte aller Spalten]
Alignment	Ausrichtung	Left/Central/Right

Beispiel:

Zeile	Inhalt
1	#Column; Num; Header; Width; Alignment
2	1;"Nummer";6;L
3	…
4	…

Absatz "Werte"

Die 1. Zeile des Absatzes "Werte" muss wie folgt lauten:

#Data; Color; Col1; Col2; Col3; ...

Für jede Zeile der Tabelle muss dieser Absatz eine separate Zeile mit den Werten für die jeweiligen Spalten enthalten.

Name	Bedeutung	Anmerkung
Color	Farbe der Zeile	Notation wie in HTML als sechsstellige Hex-Zahl: 0xbbggrr (hex blue blue green green red red)
Col1	Daten der Spalte 1	
Col2	Daten der Spalte 2	
Col3	Daten der Spalte 3	

Beispiel:

Zeile	Inhalt
1	#Data; Color; Col1; Col2; Col3; Col4
2	0xFF00FF;"05/06/02";"15.55.53";86;"+/-"
3	0x32b400;"05/06/02";"15.55.54";87;"+/-"
4	0x32b400;"05/06/02";"16.00.00";82;"+/-"

Daten in Tabellenkalkulationsprogramm aufbereiten

Wie die Beispiele in obigen Abschnitten verdeutlichen, enthält die 1. Zeile eines Absatzes als 1. Wert eine Kennung, die den Absatz identifiziert, z. B. "#Data" zur Kennzeichnung des Absatzes "Werte". Die nachfolgenden Zeilen eines Absatzes enthalten diese Kennung nicht.

Wenn Sie Zeilen eines Absatzes (einer CSV-Datei) mit einem Tabellenkalkulationsprogramm wie Microsoft Excel verarbeiten, stimmen Spaltenüberschriften und Werte nicht miteinander überein.

Um dies zu vermeiden, fügen Sie in die Zeilen mit den eigentlichen Werten ein Semikolon ";" am Zeilenanfang als Platzhalter ein:

- Spaltenüberschriften und Werte stehen in Excel korrekt untereinander.
- WinCC ignoriert das Semikolon ";" am Zeilenanfang und gibt die Daten korrekt als Kurve/Tabelle aus.

Beispiel:

Zeile	Inhalt
1	#Data; Color; Col1; Col2; Col3; Col4
2	;0xFF00FF;"05/06/02";"15.55.53";86;"+/-"
3	;0x32b400;"05/06/02";"15.55.54";87;"+/-"

Steuerzeichen zur Formatierung von Werten

Mit Steuerzeichen legen Sie z. B. die Farbe, Schriftstil oder Ausrichtung von Werten fest. Die Steuerzeichen stehen immer vor einem Wert. Sie können mehrere Steuerzeichen kombinieren: "<B><U>Ausgabetext" gibt z. B. das Wort "Ausgabetext" fett und unterstrichen aus. Zwischen Groß- und Kleinschreibung wird nicht unterschieden.

Die nachfolgende Tabelle gibt einen Überblick über die Steuerzeichen:

<END>	Beendet eine Sequenz von Steuerzeichen. Der Rest des Texts wird so übernommen, wie er angegeben ist.
<COLOR=#rrggbb>	Schriftfarbe; Notation als sechsstellige Hex-Zahl mit folgender Bytefolge: #rrggbb (red red green green blue blue)
<BGCOLOR=#rrggbb>	Hintergrundfarbe in hexadezimaler Notation
<B>	Schriftstil fett
<U>	Schriftstil unterstrichen
<I>	Schriftstil kursiv
<STRIKE>	Schriftstil durchgestrichen
<ALIGN=left>	Textausrichtung linksbündig
<ALIGN=center>	Textausrichtung zentriert
<ALIGN=right>	Textausrichtung rechtsbündig

Hinweis

Unterschiedliche Notation bei Farbangaben

Farben werden als sechsstellige Hex-Zahl notiert.

Im Absatz "Werte" wird die Farbe einer Zeile wie in HTML mit folgender Bytefolge notiert:

0xbbggrr (blue blue green green red red).

Zur Formatierung eines einzelnen Werts werden die Farbwerte im Steuerzeichen <Color> in umgekehrter Reihenfolge notiert:

#rrggbb (red red green green blue blue).

Siehe auch

Protokoll erstellen (Seite 1294)

Grundlagen zur Protokollerstellung (Seite 1290)

CSV-Provider Tabellen (Seite 1341)

7.2.1.8 Detailseiten verwalten

Detailseite hinzufügen

1. Öffnen Sie das Kontextmenü einer Detailseite.
2. Wählen Sie den Befehl "Seiten > Seite davor einfügen" oder "Seiten > Seite danach einfügen".
   Abhängig vom gewählten Befehl wird die neue Detailseite entweder vor oder nach der existierenden Detailseite eingefügt.

Detailseite löschen

1. Öffnen Sie das Kontextmenü der Detailseite, die Sie löschen wollen.
2. Wählen Sie den Befehl "Seiten > Detailseite löschen".
   Die Detailseite wird gelöscht.

Detailseiten sortieren

1. Öffnen Sie das Kontextmenü der Detailseite, die Sie verschieben wollen.
2. Wählen Sie den Befehl "Seiten > Seite eine Position aufwärts" oder "Seiten > Seite eine Position abwärts".
   Abhängig vom gewählten Befehl wird die Detailseite entweder nach oben oder nach unten verschoben.

Abschnitte ein- und ausblenden

1. Wenn Sie einen bestimmten Abschnitt ein- oder ausblenden wollen, klicken Sie im Arbeitsbereich auf das Plus- oder Minuszeichen in der Titelzeile des Abschnitts.
2. Wenn Sie alle Abschnitte eines Protokolls ein- oder ausblenden wollen, wählen Sie im Kontextmenü eines beliebigen Abschnitts den Befehl "Alle Seiten einblenden" oder "Alle Seiten ausblenden".

Siehe auch

Aufbau von Protokollen (Seite 1281)

Aufbau von Protokollen (Seite 1283)

Grundlagen zur Protokollerstellung (Seite 1290)

Grundlagen zur Protokollerstellung (Seite 1287)

7.2.2 Ausgeben von Protokollen

7.2.2.1 Protokollausgabe in Runtime Advanced

Protokollausgabe in Runtime Advanced

Einleitung

In Runtime Advanced und in den Panels werden Protokolle entweder ereignisgesteuert oder zeitgesteuert ausgegeben.

Projektierungsschritte

Ereignisgesteuerte Ausgabe projektieren:

- Sie projektieren an eine Schaltfläche oder eine Variable an ein Ereignis die Systemfunktion "DruckeProtokoll".

Zeitgesteuerte Ausgabe projektieren:

- Sie legen im Aufgabenplaner eine Aufgabe vom Typ "Druckauftrag" an und weisen dem Auftrag das gewünschte Protokoll zu.
  In den Eigenschaften der Aufgabe legen Sie fest, wann oder wie oft das Protokoll ausgegeben wird.

Ausgabe in Runtime

Das Protokoll wird auf dem Drucker ausgegeben, der im Control Panel des Bediengeräts als Standarddrucker festgelegt ist.

Welche Standarddrucker für ein Bediengerät frei gegeben sind, entnehmen Sie der "Druckerliste". Weiterführende Informationen zur "Druckerliste" finden Sie auf der Internet-Seite des Siemens Customer Support und der Beitrags-ID "11376409".

Siehe auch

Protokollieren von Meldungen projektieren (Seite 909)

Überblick über den Protokollausgabe (Seite 1308)

7.2.2.2 Protokollausgabe in Runtime Professional

Überblick über den Protokollausgabe

Einleitung

In Runtime werden Protokolle entweder ereignisgesteuert oder zeitgesteuert ausgegeben.

Darüber hinaus können Sie die anstehenden Druckaufträge in Runtime als Liste anzeigen. Mit einer solchen Druckauftragsliste kann der Bediener selbst entscheiden, welche Protokolle ausgegeben werden.

Ereignisgesteuerte Ausgabe

Das Protokoll wird nach Eintreten eines Ereignisses ausgegeben.

Beispiele:

- Wert einer Variablen ändert sich oder überschreitet einen Grenzwert
- Meldung kommt, geht oder wird quittiert
- Bedienaktion durch den Bediener, z. B. Klicken auf eine Schaltfläche

Zeitgesteuerte Ausgabe

Das Protokoll wird automatisch ausgegeben.

Beispiele:

- Einmalig an einem festgelegten Datum, z. B. am 31.12.2010 um 6:30 Uhr
- In Intervallen, z. B. täglich um 8 Uhr oder montags um 18 Uhr

Projektierungsschritte

Je nachdem für welche Runtime Sie die zeit- oder ereignisgesteuerte Ausgabe eines Protokolls projektieren, gehen Sie unterschiedlich vor.

Um die Druckauftragsliste in Runtime anzuzeigen, projektieren Sie in Bildern das Control "Applikationsfenster". Je nach Projektierung zeigt das Applikationsfenster z. B. alle oder nur ausgewählte Druckaufträge an.

Siehe auch

Druckauftrag anlegen (Seite 1311)

Automatischen Start eines Druckauftrags im Aufgabenplaner projektieren (Seite 1315)

Protokollausgabe in Runtime Advanced (Seite 1307)

Aufgaben planen (Seite 1883)

Systemprotokolle und Systemdruckaufträge (Seite 1316)

Protokollausgabe in Runtime Professional (Seite 1308)

Grundlagen zur Protokollausgabe

Einleitung

In Runtime Professional werden Protokolle entweder ereignisgesteuert oder zeitgesteuert ausgegeben.

Projektierungsschritte

- Sie legen unter Protokolle einen Druckauftrag an und weisen diesem das gewünschte Protokoll zu.
  In den Eigenschaften des Druckauftrags legen Sie Seitenumfang, Zeitbereich und Ausgabeformat (Drucker oder Datei) des Protokolls fest.
- Um eine zeitgesteuerte Ausgabe zu projektieren, legen Sie im Aufgabenplaner eine Aufgabe vom Typ "Druckauftrag" an und ordnen dieser den gewünschten Druckauftrag zu. In den Eigenschaften der Aufgabe legen Sie fest, wie oft oder wann das Protokoll ausgegeben wird.
- Um eine ereignisgesteuerte Ausgabe zu projektieren, projektieren Sie am gewünschten Objekt an ein Ereignis ein C-Skript, das die Funktion "ReportJob" aufruft, z. B.
  - `ReportJob("<Name Druckauftrag>","PRINTJOB");` zur Ausgabe des Druckauftrags
  - `ReportJob("<Name Druckauftrag>","PREVIEW");` zur Ausgabe einer Druckvorschau

Hinweis

Bildobjekte mit vorkonfigurierter Protokollausgabe

In einigen Bildobjekten ist die Protokollausgabe über eine Schaltfläche (Druckersymbol) standardmäßig enthalten, z. B. in der Meldeanzeige. Bei Bedarf können Sie die vordefinierten Systemprotokolle ändern oder ein individuell gestaltetes Protokoll verwenden.

Ausgabe in Runtime

Das Protokoll wird je nach Projektierung wie folgt ausgegeben:

- Auf dem Drucker, der im Control Panel des Bediengeräts als Standarddrucker festgelegt ist
  Welche Standarddrucker für ein Bediengerät frei gegeben sind, entnehmen Sie der "Druckerliste". Weiterführende Informationen zur "Druckerliste" finden Sie auf der Internet-Seite des Siemens Service & Support unter der Beitrags-ID "11376409".
- Auf einem Netzwerkdrucker, der für das Bediengerät verfügbar ist
- In eine Datei im EMF-Format

Protokolle ausgeben

Die Ausgabe von Protokollen erfolgt im Grafikmodus. Wegen der anfallenden Datenmengen wird der Einsatz eines seriellen Druckers nicht empfohlen.

Damit das Protokoll richtig ausgedruckt wird, muss der angeschlossene Drucker das Papierformat und Layout des Protokolls unterstützen.

Hinweis

Der Wert einer Variablen im Protokoll wird erst unmittelbar im Moment des Drucks gelesen und ausgegeben. Bei mehrseitigen Protokollen kann zwischen der ersten und letzten Seite eine größere Zeitspanne liegen. Daher kann z. B. dieselbe Variable auf der letzten Seite mit einem anderen Wert ausgegeben werden als auf der ersten Seite.

Siehe auch

Protokolle (Seite 1279)

Druckauftrag anlegen (Seite 1311)

Automatischen Start eines Druckauftrags im Aufgabenplaner projektieren (Seite 1315)

Systemprotokolle und Systemdruckaufträge (Seite 1316)

Druckauftrag anlegen

Einleitung

Im Druckauftrag legen Sie Folgendes fest:

- Umfang des Druckauftrags: alle Seiten oder nur bestimmte Seiten
- Zeitraum, für den Protokolleinträge ausgegeben werden
- Drucker, auf dem das Protokoll gedruckt wird
- Datei, in die das Protokoll gespeichert wird

Voraussetzung

Der Editor "Druckaufträge" ist geöffnet.

Vorgehen

Um einen neuen Druckauftrag anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Spalte "Name" in die Zelle "<Hinzufügen>".
   Ein neuer Druckauftrag wird eingefügt.

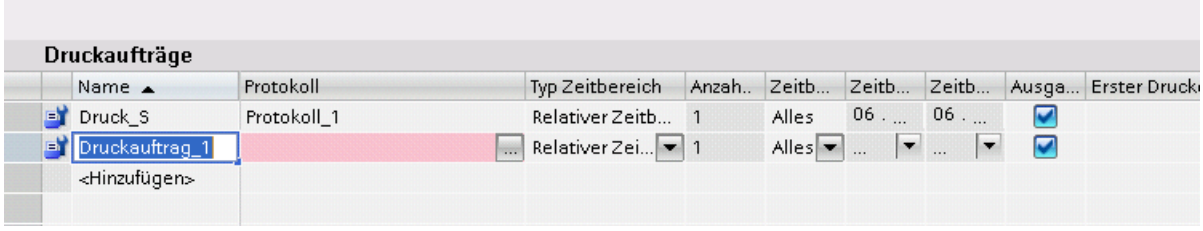

2. Geben Sie im Inspektorfenster unter " Eigenschaften > Eigenschaften > Allgemein" einen "Namen" für den Druckauftrag ein und wählen Sie das gewünschte "Protokoll" aus.
3. Wählen Sie unter "Dialogtyp" aus, ob der Bediener vor der Ausgabe des Protokolls Einstellungen ändern darf.

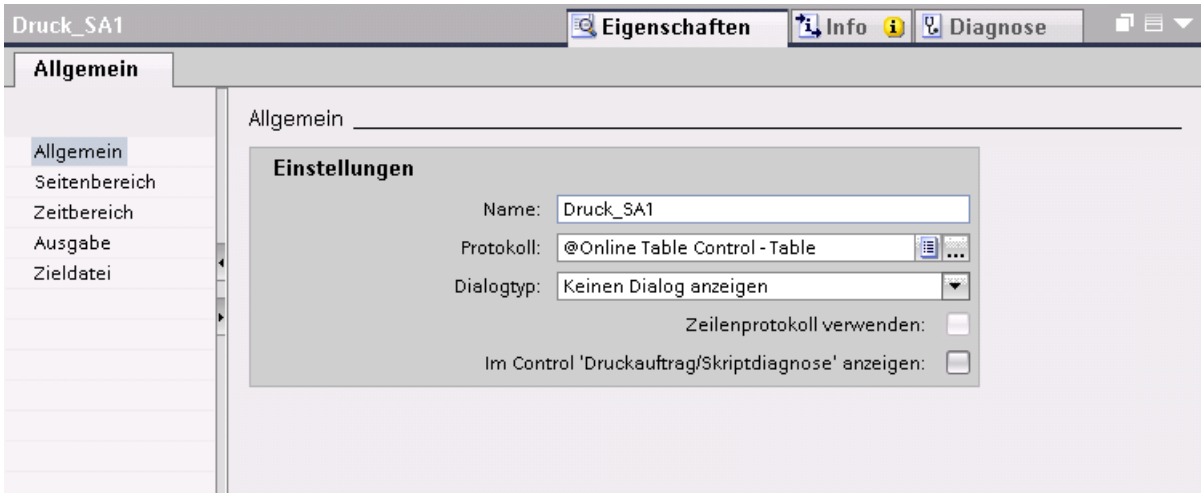

4. Legen Sie unter "Eigenschaften > Eigenschaften > Seitenbereich" fest, ob der Druckauftrag alle oder nur bestimmte Seiten ausgibt.

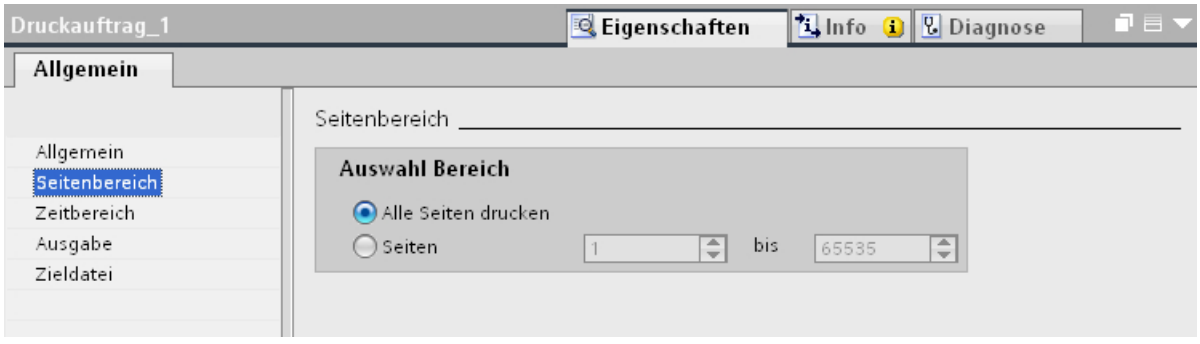

5. Legen Sie unter "Eigenschaften > Eigenschaften > Zeitbereich" fest, aus welchem Zeitraum Protokolleinträge ausgegeben werden.

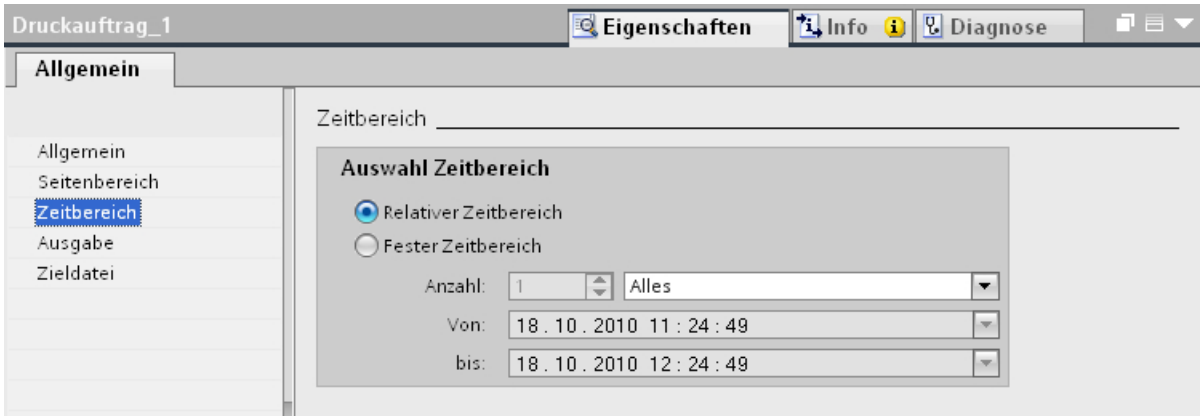

6. Wenn das Protokoll gedruckt werden soll, aktivieren Sie unter "Eigenschaften > Eigenschaften > Ausgabe > Druckausgabe" die Option "Drucker".
   Das Protokoll wird auf dem Standarddrucker des Bediengeräts ausgegeben.

7. Wenn das Protokoll auf einem anderen Drucker ausgegeben werden soll, wählen Sie bis zu drei Drucker aus.

8. Wenn das Protokoll in eine Datei ausgegeben werden soll, aktivieren Sie unter "Eigenschaften > Eigenschaften > Zieldatei > Ausgabe in Datei > In Datei drucken (*.emf)". Aus der Angabe bei "Präfix Zielverzeichnis" und dem Zeitstempel zum Druckzeitpunkt wird ein Ordnername gebildet. Die Protokolle werden im Projektverzeichnis der Runtime unter "PRT_OUT" in einem Ordner dieses Namens abgelegt.

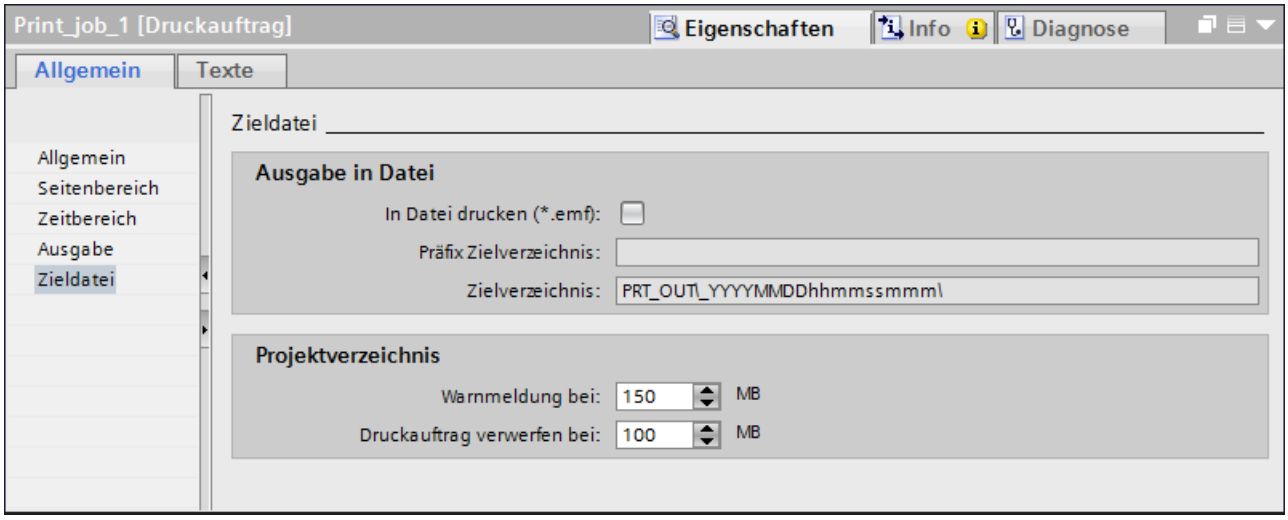

Ergebnis

Der Druckauftrag für das Protokoll ist angelegt.

- Für die zeitgesteuerte Ausgabe in Runtime legen Sie im Aufgabenplaner eine Aufgabe vom Typ "Druckauftrag" an. Ordnen Sie dieser Aufgabe den gewünschten Druckauftrag zu.
- Für eine ereignisgesteuerte Ausgabe projektieren Sie an das Objekt an das gewünschte Ereignis ein C-Skript, das die Funktion "PrintJob" aufruft.

Siehe auch

Grundlagen zur Protokollausgabe (Seite 1309)

Automatischen Start eines Druckauftrags im Aufgabenplaner projektieren (Seite 1315)

Automatischen Start eines Druckauftrags im Aufgabenplaner projektieren

Einleitung

Eine Aufgabe vom Typ "Druckauftrag" legt fest, wann oder wie oft ein Protokoll automatisch ausgegeben wird (zeitgesteuerte Ausgabe).

Voraussetzungen

- Ein Druckauftrag ist angelegt.

Vorgehen

Um eine Aufgabe für den automatischen Start eines Druckauftrags anzulegen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie in den "Runtime-Einstellungen" des Bediengeräts "Dienste > Startsequenz von WinCC Runtime > Geplante Druckaufträge in Runtime".
2. Legen Sie im Aufgabenplaner eine neue Aufgabe an.
3. Geben Sie unter "Eigenschaften > Eigenschaften > Allgemein" einen "Namen" für die Aufgabe ein.
4. Wählen Sie im Feld "Typ" die Einstellung "Druckauftrag".
5. Wählen Sie den gewünschten "Druckauftrag" aus.
6. Legen Sie unter "Startzeitpunkt" fest, wann oder wie oft das Protokoll ausgegeben werden soll, z. B. täglich um 17 Uhr:

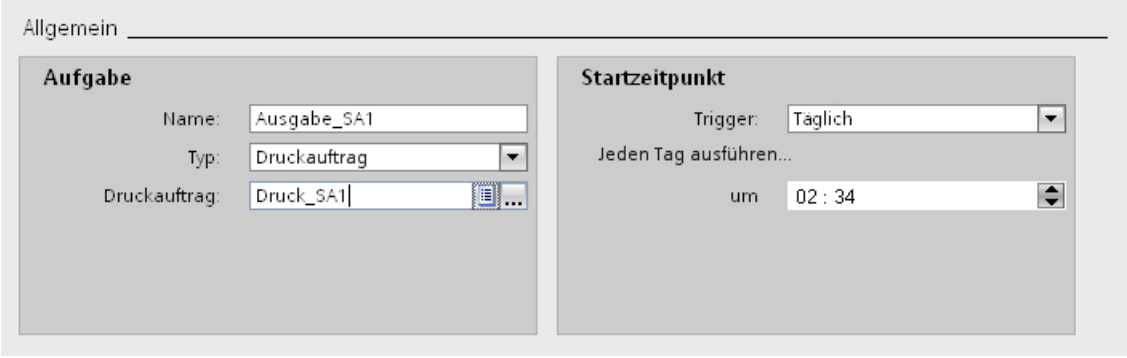

Ergebnis

Die Aufgabe für die zeitgesteuerte Ausgabe des Protokolls ist angelegt.

Siehe auch

Grundlagen zur Protokollausgabe (Seite 1309)

Druckauftrag anlegen (Seite 1311)

Ausdrucken eines Protokolls (Seite 1334)

7.2.2.3 Systemprotokolle und Systemdruckaufträge

Grundlagen zu Systemprotokollen und Systemdruckaufträgen

Systemprotokolle und Systemdruckaufträge für Druckfunktion von Bildobjekten

In einigen Bildobjekten ist die Protokollausgabe über eine Schaltfläche (Druckersymbol) standardmäßig enthalten, z. B. in der Meldeanzeige.

Die Schaltfläche "Drucken" des Bildobjekts ist mit einem vorkonfigurierten Systemdruckauftrag verbunden. Der Systemdruckauftrag wiederum gibt ein vorkonfiguriertes Systemprotokoll aus.

Systemprotokolle geben entweder einen "Schnappschuss" oder "alle Daten" des Bildobjekts aus.

- Schnappschuss: Das Protokoll gibt die Daten aus, die aktuell im Bildobjekt angezeigt werden
- Alle Daten: Das Protokoll gibt alle Daten aus, die aktuell im Bildobjekt enthalten sind.

Die folgende Tabelle zeigt, welche Systemprotokolle und welche Systemdruckaufträge für die Bildobjekte vorkonfiguriert sind.

Bildobjekt	Standard	Druckumfang	Systemdruckauftrag	Systemprotokolle
f(t)-Kurvenanzeige	X	Schnappschuss	Function Trend Control - Picture	@Function Trend Control - Picture
f(x)-Kurvenanzeige	X	Schnappschuss	Online Trend Control - Picture	@Online Trend Control - Picture
Tabellenanzeige		Schnappschuss	Online Table Control – Picture	@Online Table Control – Picture
	X	Alle Daten	Online Table Control - Table	@Online Table Control - Table
Meldeanzeige		Schnappschuss	Alarm Control – Picture	@Alarm Control – Picture
	X	Alle Daten	Alarm Control - Table	@Alarm Control - Table
Rezepturanzeige		Schnappschuss	Recipe Control - Picture	@Recipe Control - Picture
	X	Alle Daten	Recipe Control - Table	@Recipe Control - Table
Wertetabelle		Schnappschuss	Ruler Control - Picture	@Ruler Control - Picture
	X	Alle Daten	Ruler Control - Table	@Ruler Control - Table

In der Regel müssen Sie die Protokollausgabe nicht projektieren. Bei Bedarf passen Sie den Systemdruckauftrag oder das Systemprotokoll an Ihre Bedürfnisse an. Alternativ weisen Sie einem Bildobjekt einen anderen vorkonfigurierter Systemdruckauftrag oder einen Druckauftrag zu, den Sie für die Ausgabe eines individuell gestalteten Protokolls angelegt haben.

Systemprotokolle und Systemdruckauftrag für Meldefolgeprotokoll

Die Ausgabe des Meldefolgeprotokolls ist in einem Systemdruckauftrag projektiert. Wenn Sie "Eigenschaften > Eigenschaften > Allgemein > Zeilenprotokoll verwenden" aktiviert haben, verwendet WinCC das Meldefolgeprotokoll mit Zeilenlayout. Wenn Sie die Option nicht aktiviert haben, verwendet WinCC das Meldefolgeprotokoll mit Seitenlayout.

Lay-out	Systemdruckauftrag	Systemprotokoll
Seitenlayout	Alarm sequence report	@AlarmSequenceReportForPagePrinter
Zeilenlayout	Alarm sequence report	@AlarmSequenceReportForLinePrinter

Siehe auch

Fortlaufende Protokollierung von Meldungen (Seite 1300)

Grundlagen zur Protokollausgabe (Seite 1309)

Systemprotokolle und Systemdruckaufträge ändern

Systemprotokolle und Systemdruckaufträge in Projektnavigation anzeigen

Die Systemprotokolle und Systemdruckaufträge sind in der Projektvorlage gespeichert und werden in der Projektnavigation nicht automatisch angezeigt. Wenn Sie ein Systemprotokoll oder einen Systemdruckauftrag ändern wollen, blenden Sie diese Protokolle und Druckaufträge in der Projektnavigation ein.

Dazu gehen Sie folgendermaßen vor:

- Wählen Sie in der Projektnavigation im Kontextmenü des Ordners "Protokolle" den Befehl "Systemprotokolle hinzufügen".

Die Projektnavigation zeigt unter "Protokolle" die Systemprotokolle mit an.

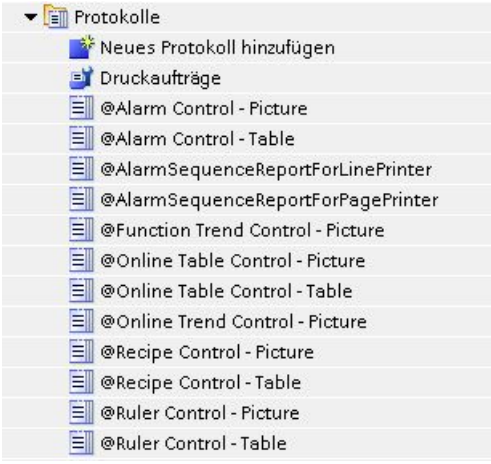

Die Liste der "Druckaufträge" ist um die Systemdruckaufträge erweitert.

Druckaufträge

Name	Protokoll	Erster Drucker	Komment
Alarm Control - Picture	@Alarm Control - Picture		
Alarm Control - Table	@Alarm Control - Table		
Alarm sequence report	@AlarmSequenceReportForPagePrinter		
Function Trend Control - Picture	@Function Trend Control - Picture		
Online Table Control - Picture	@Online Table Control - Picture		
Online Table Control - Table	@Online Table Control - Table		
Online Trend Control - Picture	@Online Trend Control - Picture		
Recipe Control - Picture	@Recipe Control - Picture		
Recipe Control - Table	@Recipe Control - Table		
Ruler Control - Picture	@Ruler Control - Picture		
Ruler Control - Table	@Ruler Control - Table		
<Hinzufügen>			

Systemdruckaufträge und Systemprotokolle ändern

Sie ändern Systemdruckaufträge und Systemprotokolle wie "normale" Druckaufträge und Protokolle – mit folgenden Einschränkungen:

- Bei Systemdruckaufträgen können weder Name noch die Standardzuordnung des Protokolls geändert werden.
- Systemprotokollen können keine Detailseiten hinzugefügt werden.

Gültigkeit der Änderungen

Änderungen an Systemprotokollen- und -druckaufträgen wirken sich im Projekt auf alle Objekte aus, denen das Systemprotokoll bzw. der Systemdruckauftrag zugewiesen ist.

Hinweis

Änderungen verwerfen

Um den von WinCC vorkonfigurierten Zustand wiederherzustellen, z. B. nachdem das Control zur Ausgabe der Daten versehentlich im Protokoll gelöscht wurde, löschen Sie zunächst das Systemprotokoll bzw. den -druckauftrag und wählen dann in der Projektnavigation im Kontextmenü des Ordners "Protokolle" den Befehl "Systemprotokolle hinzufügen".

Nicht benötigte Systemprotokolle und Systemdruckaufträge löschen

Um nur die im Projekt verwendeten Protokolle und Druckaufträge anzuzeigen (z. B. nach Migration), können Sie Systemprotokolle und -druckaufträge über das Kontextmenü löschen.

Wenn das gelöschte Protokoll bzw. der gelöschte Druckauftrag im Projekt verwendet wird, werden das von WinCC vorkonfigurierte Systemprotokoll und der vorkonfigurierte Systemdruckauftrag verwendet.

Um wieder alle Systemprotokolle und -druckaufträge anzuzeigen, wählen Sie in der Projektnavigation im Kontextmenü des Ordners "Protokolle" den Befehl "Systemprotokolle hinzufügen":

- Zuvor gelöschte Systemprotokolle und –druckaufträge werden wieder eingeblendet. Der von WinCC vorkonfigurierte Stand ist wiederhergestellt. Der Stand zum Zeitpunkt des Löschens wird nicht rekonstruiert.
- Alle anderen Systemprotokolle und -druckaufträge bleiben unverändert.

Siehe auch

Grundlagen zu Systemprotokollen und Systemdruckaufträgen (Seite 1316)

Protokollausgabe eines Bildobjekts individuell gestalten

Einleitung

Wenn Sie für die Protokollausgabe eines Bildobjekts nicht den standardmäßig verknüpften Systemdruckauftrag (inkl. Systemprotokoll) verwenden wollen, weisen Sie dem Bildobjekt einen anderen Systemdruckauftrag oder ein individuell gestaltetes Protokoll zu.

Anderen Systemdruckauftrag verwenden

Um dem Bildobjekt einen anderen Systemdruckauftrag zuzuweisen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Bildobjekt.
2. Wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Drucken" den gewünschten Systemdruckauftrag aus.

Individuell gestaltetes Protokoll verwenden

Um ein individuell gestaltetes Protokoll zu verwenden, gehen Sie folgendermaßen vor:

1. Erstellen Sie das Protokoll.
2. Legen Sie den Inhalt der Detailseite fest.
   - Um einen Schnappschuss der aktuell *im Bildobjekt angezeigten* Daten auszugeben, fügen Sie das Protokollobjekt "Control Hardcopy" ein.
   - Um alle Protokolldaten in Tabellenform auszugeben, die *aktuell im Bildobjekt enthalten* sind, fügen Sie das Protokollobjekt "Control Daten" ein.
   - Um die Datenquelle im Protokoll mit auszugeben, z. B. Fenstertitel, Objektname oder Name des Bildes, fügen Sie das Protokollobjekt "Control Screen" ein.
3. Erstellen Sie einen Druckauftrag und weisen Sie diesem das Protokoll zu.
4. Selektieren Sie das Bildobjekt und wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Drucken" den Druckauftrag aus.

7.2.3 Arbeiten mit Objekten

7.2.3.1 Objekt einfügen

Einleitung

Im Editor "Bilder" oder "Protokolle" fügen Sie die Objekte der Task Card "Werkzeuge" ein. Sie ziehen die Objekte mit der Maus in den Arbeitsbereich. Sie fügen die Objekte in der Standardgröße ein oder bestimmen die Größe beim Einfügen.

Darüber hinaus können Sie Objekte über die Zwischenablage von einem Editor in einen anderen Editor kopieren oder verschieben, z. B. um ein Bildobjekt in ein Protokoll zu übernehmen. Alternativ können Sie zum Kopieren und Verschieben statt der Zwischenablage auch die Maus verwenden:

- Kopieren: <Strg + Drag&Drop>
- Verschieben: Drag&Drop

Hinweis

Basic Panels

Der Editor "Protokolle" steht für Basic Panels nicht zur Verfügung.

Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

Objekt in Standardgröße einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt oder die gewünschte Grafik im WinCC Grafikordner.
   Wenn Sie den Mauszeiger im Arbeitsbereich bewegen, wird er zu einem Fadenkreuz mit angehängtem Objektsymbol.
2. Klicken Sie auf die Stelle im Arbeitsbereich, an der Sie das Objekt oder die Grafik einfügen wollen.
   Das Objekt wird in der Standardgröße an der gewünschten Position im Arbeitsbereich eingefügt.

Alternativ doppelklicken Sie in der Task Card "Werkzeuge" auf ein Objekt.

Objekt kopieren

1. Markieren Sie das gewünschte Objekt.
2. Wählen Sie im Kontextmenü "Kopieren".
3. Klicken Sie an die gewünschte Stelle und wählen Sie im Kontextmenü "Einfügen".

WinCC fügt eine Kopie des Objekts an der gewünschten Stelle ein. Sie können nur die Eigenschaften ändern, die in dem jeweiligen Kontext angemessen sind.

Beispiel: Im Editor "Bilder" legen Sie für E/A-Felder den Modus für Ein- und Ausgabe fest. Im Editor "Protokolle" ist der Modus auf "Ausgabe" festgelegt.

Original und Kopie sind nicht miteinander verbunden und werden unabhängig voneinander konfiguriert.

Linien einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt.
2. Klicken Sie an einer Stelle im Arbeitsbereich. Eine Linie wird in der Standardgröße eingefügt.

Polygon oder Polygonzug einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt "Polygonzug" oder "Polygon".
2. Klicken Sie an einer Stelle im Arbeitsbereich. Der Anfangspunkt des Objekts ist festgelegt.
3. Klicken Sie an eine andere Stelle im Arbeitsbereich. Ein Eckpunkt ist festgelegt.
4. Für jeden weiteren Eckpunkt klicken Sie im Arbeitsbereich an die entsprechende Stelle.
5. Doppelklicken Sie an einer Stelle im Arbeitsbereich. Der letzte Eckpunkt ist festgelegt. Alle Punkte des Polygons oder Polygonzugs sind festgelegt.

Hinweis

Basic Panels

Die Objekte "Polygonzug" und "Polygon" stehen für Basic Panels nicht zur Verfügung.

Hinweis

Um nacheinander mehrere Objekte gleichen Typs einzufügen, nutzen Sie die Funktion "Stempel". Sie vermeiden dadurch, das Objekt vor jedem Einfügen erneut in der Task Card "Werkzeuge" zu markieren. Wählen Sie dazu in der Symbolleiste der Task Card "Werkzeuge" das Symbol .

7.2.3.2 Objekt löschen

Einleitung

Sie löschen Objekte einzeln oder über eine Mehrfachauswahl.

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie löschen wollen.
   Um mehrere Objekte zu löschen, halten Sie die Taste <Shift> gedrückt und markieren Sie nacheinander die zu löschenden Objekte. Alternativ ziehen Sie mit der Maus einen Bereich um die gewünschten Objekte auf.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Ergebnis

Die selektierten Objekte werden gelöscht.

7.2.3.3 Mehrere Objekte eines Typs einfügen

Einleitung

WinCC bietet die Möglichkeit, mehrere Objekte des gleichen Typs direkt hintereinander zu "stempeln" d. h. einzufügen, ohne dabei jedesmal das Objekt neu auszuwählen. Außerdem haben Sie die Möglichkeit, ein bereits eingefügtes Objekt beliebig zu vervielfachen.

Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

Mehrere Objekte eines Typs einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das Objekt, das Sie einfügen wollen.
2. Klicken Sie in der Symbolleiste der Task Card "Werkzeuge" auf das Symbol .
   Die Funktion "Stempel" ist aktiviert.
3. Wenn Sie das Objekt in Standardgröße einfügen wollen, klicken Sie auf die gewünschte Position im Arbeitsbereich.
   Wenn Sie das Objekt in einer anderen Größe einfügen wollen, positionieren Sie den Mauszeiger auf die gewünschte Position im Arbeitsbereich. Ziehen Sie das Objekt bei gedrückter linker Maustaste auf die gewünschte Größe.
   Sobald Sie die Maustaste loslassen, ist das Objekt in den Arbeitsbereich eingefügt.
4. Um weitere Objekte desselben Typs einzufügen, wiederholen Sie Schritt 3.
5. Klicken Sie erneut auf das Symbol .
   Die Funktion "Stempel" ist deaktiviert.

Hinweis

Vorhandene Objekte kopieren Sie, indem Sie Drag&Drop nutzen und gleichzeitig die Taste <Strg> drücken. Das vorhandene Objekt wird dann nicht verschoben. Stattdessen wird eine Kopie davon an der neuen Position eingefügt.

Objekt einfügen und vervielfachen

1. Fügen Sie das gewünschte Objekt aus der Task Card "Werkzeuge" ein.
2. Drücken Sie die Taste <Strg> und positionieren Sie den Mauszeiger auf einen der im folgenden Bild dargestellten Anfasser.

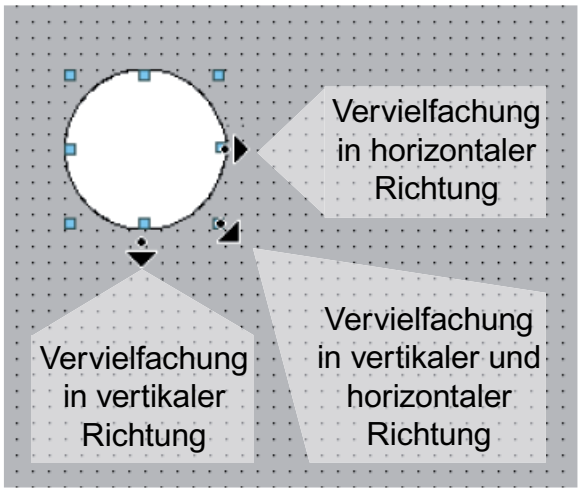

3. Ziehen Sie die Anfasser bei gedrückter linker Maustaste nach rechts und/oder unten.
4. Das Objekt wird dem verfügbaren Platz entsprechend vervielfacht, je weiter Sie den Mauszeiger bewegen.

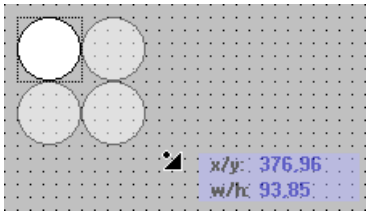

Ergebnis

Sie haben ein Objekt in einem Bild eingefügt und vervielfältigt.

7.2.3.4 Objekt positionieren

Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem Rechteck mit Anfassern umrahmt. Dieses Rechteck ist das Objektumfassende Rechteck. Die Position eines Objekts ist definiert durch die Koordinaten der linken oberen Ecke des Objektumfassenden Rechtecks.

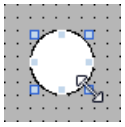

Hinweis

Wenn die Position außerhalb des Arbeitsbereichs liegt, wird das Objekt in Runtime nicht dargestellt.

Positionieren und Ausrichten

Sie haben die Möglichkeit im Arbeitsbereich ein Raster anzeigen zu lassen. Um Objekte einfacher zu positionieren, stehen Ihnen drei Optionen zur Verfügung:

- "Am Raster ausrichten": Wenn Sie Objekte bewegen, werden sie automatisch am Raster ausgerichtet und eingefügt. Wenn Sie gleichzeitig die Taste <Alt> gedrückt halten, wird das Objekt nicht am Raster ausgerichtet.
- "An Objekten ausrichten": Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.
- "Kein": Sie positionieren die Objekte an beliebiger Position.

Sie aktivieren und deaktivieren das Raster und die Optionen wie folgt:

- Im Menü "Extras > Einstellungen > Visualisierung > Bilder"
- In der Task Card "Layout > Raster"

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie verschieben wollen.
   Das markierte Objekt wird von einem Rechteck mit Anfassern umrahmt.

   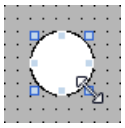

2. Klicken Sie auf das Objekt und halten Sie die linke Maustaste gedrückt.
3. Bewegen Sie den Mauszeiger auf die neue Position.
   Der Umriss des Objekts bewegt sich mit der Maus und zeigt die neue Position für das Objekt an.

   Das Objekt selbst behält zunächst seine ursprüngliche Position.
4. Lassen Sie die Maustaste los.
   Das Objekt wird auf die Position verschoben, die vorher durch den Objektumriss angezeigt wurde.

Alternatives Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die X-Werte und Y-Werte der Position ein.

Ergebnis

Das Objekt wird an der neuen Position angezeigt.

7.2.3.5 Objekt nach vorn oder hinten schieben (Basic, Advanced, Professional)

Einleitung

Mit den Funktionen "Reihenfolge" im Kontextmenü zu einem ausgewählten Objekt oder in der Symbolleiste schieben Sie das Objekt innerhalb einer Ebene vor oder hinter andere Objekte.

Hinweis

ActiveX-Controls liegen immer vor anderen Objekten einer Ebene (.NET-Eigenschaft).

Voraussetzung

Ein Bild mit mehreren Objekten auf einer Ebene ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, das Sie nach vorn oder nach hinten schieben wollen.
2. Wählen Sie im Kontextmenü den Befehl "Reihenfolge" und einen der folgenden Befehle:

Symbol	Beschreibung
	Schiebt das markierte Objekt vor alle anderen Objekte derselben Ebene
	Schiebt das markierte Objekt hinter alle anderen Objekte derselben Ebene
	Schiebt das markierte Objekt um eine Position nach vorn
	Schiebt das markierte Objekt um eine Position nach hinten

Alternatives Vorgehen

1. Öffnen Sie die Palette "Ebenen" der Task Card "Layout".
2. Navigieren Sie zum gewünschten Objekt.
3. Ziehen Sie das Objekt in der Baumstruktur mit gedrückter Maustaste an die gewünschte Position innerhalb der Ebene.
4. Lassen Sie die Maustaste los.

Ergebnis

Das Objekt wird nach vorn oder nach hinten geschoben.

7.2.3.6 Größe eines Objekts ändern

Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem objektumfassenden Rechteck mit Anfassern umrahmt. Sie haben folgende Möglichkeiten, die Größe eines Objekts zu ändern:

- Die Anfasser mit der Maus ziehen
- Eigenschaft "Größe" im Inspektorfenster ändern

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, dessen Größe Sie ändern wollen.
   Das objektumfassende Rechteck wird angezeigt. Das folgende Bild zeigt ein markiertes Objekt:

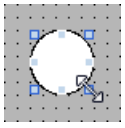

2. Ziehen Sie einen Anfasser des Rechtecks an eine neue Position.
   Die Größe des Objekts ändert sich.
   - Wenn die Funktion "Am Raster ausrichten" aktiviert ist, ändert sich die Objektgröße entsprechend den Rasterpunkten.
   - Wenn Sie beim Ziehen des Anfassers die Taste <Alt> gedrückt halten, wird diese Funktion nicht verwendet.
     Um das Objekt proportional zu skalieren, halten Sie die Taste <Shift> gedrückt, während Sie mit der Maus die Größe ändern.

Alternatives Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die Größe des Objekts ein.

Objektgröße vereinheitlichen

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der folgenden Schaltflächen: oder oder
   Die Größe der ausgewählten Objekte wird aneinander angepasst.

Das folgende Bild zeigt, wie Sie die ausgewählten Objekte an die Höhe des Referenzobjekts anpassen:

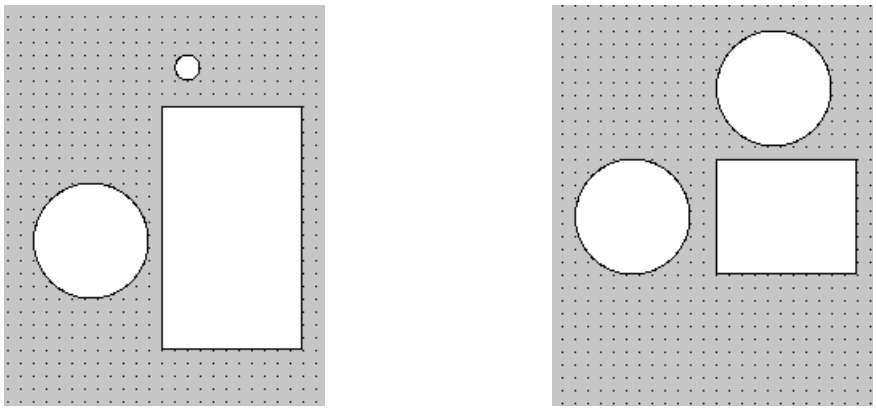

Symbol	Beschreibung
	Passt die ausgewählten Objekte an die Breite des Referenzobjekts an.
	Passt die ausgewählten Objekte an die Höhe des Referenzobjekts an.
	Passt die ausgewählten Objekte an die Breite und die Höhe des Referenzobjekts an.

Ergebnis

Das Objekt wird in der neuen Größe angezeigt.

7.2.3.7 Mehrere Objekte auswählen

Einleitung

Um die Eigenschaften von mehreren Objekten zugleich zu ändern oder die Objekte aneinander auszurichten, wählen Sie alle betreffenden Objekte aus. Dieser Vorgang heißt "Mehrfachauswahl".

Im Inspektorfenster erscheinen alle Eigenschaften der markierten Objekte.

Sie haben folgende Möglichkeiten, mehrere Objekte zu markieren:

- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte.
- Sie klicken mit gedrückter Taste <Shift> auf die gewünschten Objekte.

Auswahlrahmen einer Mehrfachauswahl

Der Auswahlrahmen umschließt alle Objekte einer Mehrfachauswahl. Der Auswahlrahmen ist vergleichbar mit dem objektumfassenden Rechteck eines einzelnen Objekts.

Der Auswahlrahmen ist nicht sichtbar. Nach Abschluss der Mehrfachauswahl werden folgende Rahmen angezeigt:

- Das Referenzobjekt wird mit dem objektumfassenden Rechteck angezeigt.
- Die anderen ausgewählten Objekte werden mit einem gestrichelten Rahmen angezeigt.

Referenzobjekt festlegen

Das Referenzobjekt ist das Objekt, an dem die anderen Objekte ausgerichtet werden. Das Referenzobjekt ist von einem Rechteck mit Anfassern umrahmt. Das folgende Bild zeigt ein Referenzobjekt mit zwei weiteren ausgewählten Objekten:

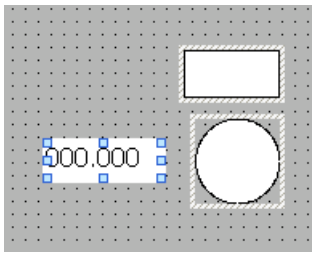

Um das Referenzobjekt festzulegen, haben Sie folgende Möglichkeiten:

- Sie wählen die Objekte über Mehrfachauswahl aus. Das zuerst ausgewählte Objekt ist dann das Referenzobjekt.
- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte. Das Referenzobjekt wird automatisch festgelegt. Wenn Sie ein anderes Objekt innerhalb der Auswahl als Referenzobjekt festlegen wollen, klicken Sie auf das gewünschte Objekt. Die Mehrfachauswahl wird dadurch nicht aufgehoben.

Voraussetzung

Der Arbeitsbereich mit mindestens zwei Objekten ist geöffnet.

Mehrere Objekte mit einem Auswahlrahmen auswählen

1. Positionieren Sie den Mauszeiger im Arbeitsbereich in der Nähe eines der auszuwählenden Objekte.
2. Ziehen Sie bei gedrückter Maustaste einen Auswahlrahmen um die auszuwählenden Objekte.

Oder:

1. Halten Sie die Taste <Shift> gedrückt.
2. Klicken Sie mit der Maus nacheinander auf die gewünschten Objekte.
   Alle ausgewählten Objekte werden durch Rahmen markiert.
   Das zuerst markierte Objekt wird als Referenzobjekt gekennzeichnet.

Hinweis

Sie entfernen ein Objekt aus der Mehrfachauswahl, indem Sie die Taste <Shift> gedrückt halten und das Objekt erneut anklicken.

Ergebnis

Mehrere Objekte sind markiert. Eines davon ist als Referenzobjekt gekennzeichnet. Sie können jetzt folgende Schritte ausführen:

- Die Objekteigenschaften aller Objekte verändern
- Die Größe aller Objekte um einen gleichen Faktor ändern, indem Sie mit der Maus den Auswahlrahmen entsprechend größer oder kleiner ziehen
- Alle Objekte gemeinsam verschieben
- Die Objekte am Referenzobjekt ausrichten

7.2.3.8 Position und Größe mehrerer Objekte verändern

Mögliche Änderungen

Nachdem Sie mehrere Objekte ausgewählt haben, bearbeiten Sie diese:

- Mit der Maus verschieben
  - Um die absolute Lage der markierten Objekte zu verändern, positionieren Sie den Mauszeiger über ein Objekt und verschieben bei gedrückter Maustaste die Mehrfachauswahl.
  - Um die Größe aller Objekte um den gleichen Faktor zu ändern, ziehen Sie mit der Maus an den Anfassern des Referenzobjekts.
- Mit den Symbolen der Symbolleiste über dem Arbeitsbereich verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen
- Mit den Befehlen im Kontextmenü des Arbeitsbereichs verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen

7.2.3.9 Objekte ausrichten

Vorgehen

1. Wählen Sie die gewünschten Objekte über Mehrfachauswahl aus.
2. Legen Sie ein Objekt als Referenzobjekt fest.
3. Wählen Sie in der Symbolleiste oder im Kontextmenü den gewünschten Befehl - siehe Tabelle unten.
   Die ausgewählten Objekte werden ausgerichtet.

Objekte bündig ausrichten

Die ausgewählten Objekte werden bündig am Referenzobjekt ausgerichtet.

Symbol	Beschreibung
	Richtet die ausgewählten Objekte an der linken Kante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der vertikalen Mittelachse des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der rechten Kante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der Oberkante des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der horizontalen Mittelachse des Referenzobjekts aus.
	Richtet die ausgewählten Objekte an der Unterkante des Referenzobjekts aus.
	Zentriert die ausgewählten Objekte auf dem Mittelpunkt des Referenzobjekts.
	Zentriert die ausgewählten Objekte vertikal im Bild.

An einem Objekt ausrichten

Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.

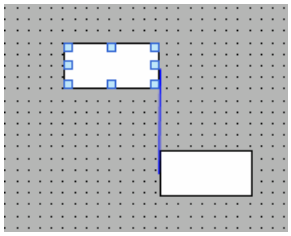

Wenn Sie mit der Tastatur arbeiten, drücken Sie die Alt-Taste. Wenn Sie das gewählte Objekt mit den Pfeiltasten bewegen, wird Ihnen der nächste Ankerpunkt angezeigt.

Objekte gleichmäßig verteilen

Sie benötigen mindestens drei ausgewählte Objekte. Ein Referenzobjekt wird nicht benötigt.

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der Schaltflächen "Horizontal gleichmäßig verteilen" oder "Vertikal gleichmäßig verteilen".
   Die ausgewählten Objekte werden in gleichmäßigen Abständen verteilt.

Das folgende Bild zeigt, wie Sie den vertikalen Abstand zwischen den ausgewählten Objekten vereinheitlichen:

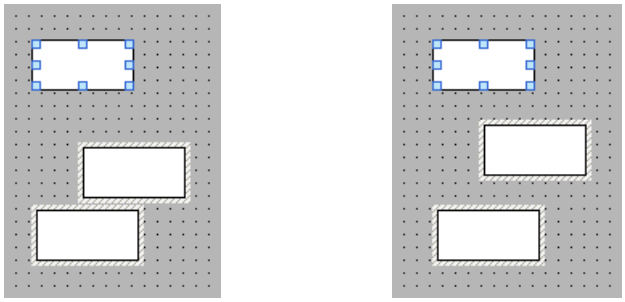

Symbol	Beschreibung
	Vereinheitlicht den horizontalen Abstand zwischen den Objekten. Die Lage der äußersten Objekte (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt.
	Vereinheitlicht den vertikalen Abstand zwischen den Objekten. Die Lage des obersten und untersten Objekts (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt.

7.2.3.10 Objekt drehen

Einleitung

Sie können ein geeignetes Objekt in Schritten von 90° im oder gegen den Uhrzeigersinn um sich selbst drehen.

Hinweis

Nicht alle Objekte sind drehbar. Einige Objekte, die in Bildern drehbar sind, können in Protokollen nicht gedreht werden.

Mithilfe der Mehrfachauswahl drehen Sie auch mehrere Objekte zugleich. Bestimmte WinCC Objekte, z. B. Schaltflächen, lassen sich nicht drehen.

In einem gedrehten Objekt ändert sich die Ausrichtung der im Objekt enthaltenen Elemente. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Drehen eines Objekts verhalten:

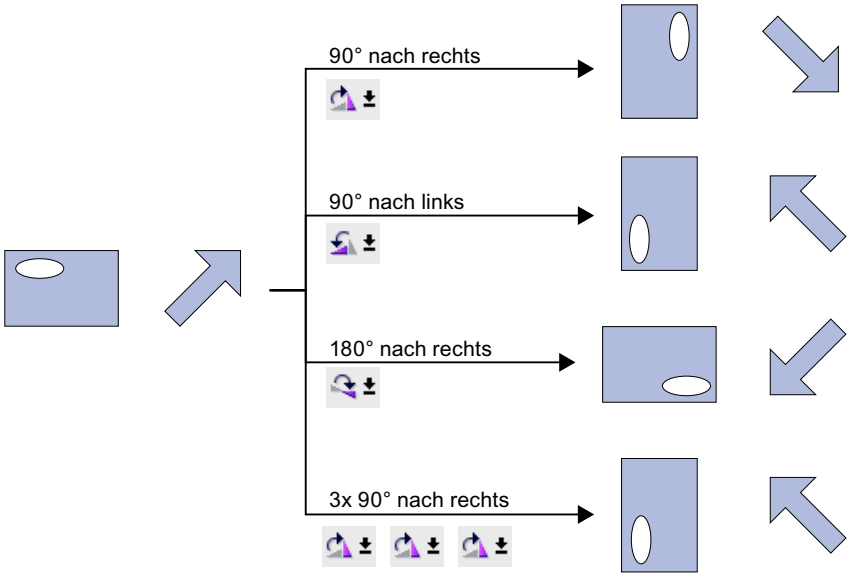

Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Selektieren Sie das Objekt, das Sie drehen wollen.
2. Klicken Sie in der Symbolleiste auf eines der folgenden Symbole:
   , um das Objekt im Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
   , um das Objekt gegen den Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
   , um das Objekt im Uhrzeigersinn um 180° zu drehen.

Ergebnis

Das Objekt wird gedreht angezeigt.

7.2.3.11 Objekt spiegeln

Einleitung

Sie können ein Objekt an seiner vertikalen oder horizontalen Mittelachse spiegeln. Wenn Sie ein Objekt spiegeln, ändert sich auch die Ausrichtung von im Objekt enthaltenen Elementen. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Spiegeln eines Objekts verhalten.

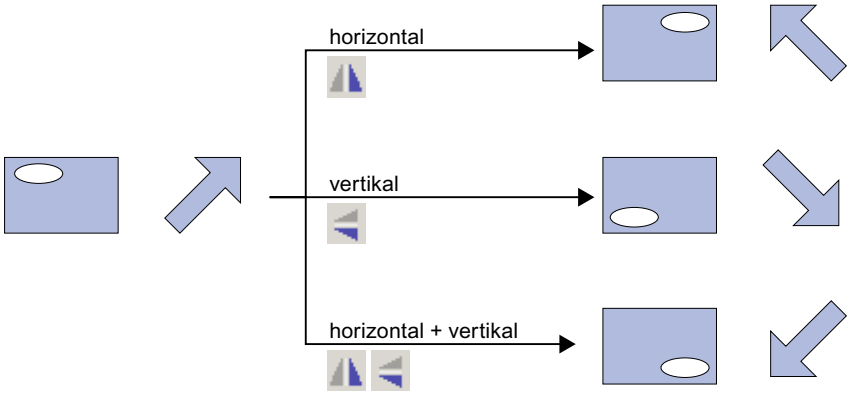

Voraussetzung

Ein Bild mit mindestens einem Objekt ist geöffnet.

Vorgehen

1. Wählen Sie das Objekt, das Sie spiegeln wollen.
2. Klicken Sie im Kontextmenü auf den Befehl "Spiegeln" und wählen Sie eine der angebotenen Möglichkeiten:
   – , um das markierte Objekt an seiner vertikalen Mittelachse zu spiegeln.
   – , um das markierte Objekt an seiner horizontalen Mittelachse zu spiegeln.

Ergebnis

Das Objekt wird gespiegelt angezeigt.

7.2.4 Bedienung in Runtime

7.2.4.1 Ausdrucken eines Protokolls

Ausdrucken eines Protokolls in Runtime

Wenn Sie ein Objekt mit der Systemfunktion "DruckeProtokoll" projektiert haben, kann der Bediener in Runtime ein Protokoll ausdrucken. Dazu drückt der Bediener eine Schaltfläche, z. B. "Protokoll drucken" in einem Bild. Der Bediener kann keine weiteren Einstellungen für das Protokoll festlegen oder ändern. Das Protokoll wird auf dem Standarddrucker ausgegeben.

7.2.4.2 Ausdrucken eines Protokolls

Start des Protokolldrucks durch den Bediener

Der Bediener startet in Runtime in folgenden Bildobjekten den Druck eines Protokolls über eine Schaltfläche "Drucken":

- f(x)-Kurvenanzeige
- f(t)-Kurvenanzeige
- Tabellenanzeige
- Meldeanzeige
- Rezepturanzeige

Wenn Sie im Druckauftrag festgelegt haben, dass ein Dialog zur Auswahl eines Druckers oder zum Festlegen von Eigenschaften angezeigt wird, ändert der Bediener diese Einstellungen in Runtime.

Wenn Sie eine Schaltfläche projektiert und mit einem entsprechenden lokalen Skript verknüpft haben, kann der Bediener den Protokolldruck über diese Schaltfläche starten.

Automatischer Start des Protokolldrucks

In folgenden Fällen wird der Protokolldruck in Runtime automatisch gestartet:

- Sie haben einen Druckauftrag mit einer Aufgabe von Typ "Druckauftrag" verknüpft.
- Sie haben ein globales Skript zum Protokolldruck mit einer Aufgabe verknüpft.
- Sie haben in den Runtime-Einstellungen des Bediengeräts unter "Dienste" den Eintrag "Geplante Druckaufträge in Runtime" aktiviert.

Druckauftrag/Skriptdiagnose

In Runtime werden abhängig von der Projektierung alle oder ausgewählte Druckaufträge im Objekt "Druckauftrag/Skriptdiagnose" angezeigt, wenn sie anstehen. Abhängig von der Projektierung kann der Bediener im Objekt "Druckauftrag/Skriptdiagnose" den Protokolldruck starten, Druckoptionen festlegen und eine Druckvorschau anzeigen.

Siehe auch

Grundlagen zur Protokollausgabe (Seite 1309)
Grundlagen zu Systemprotokollen und Systemdruckaufträgen (Seite 1316)

7.3 Objekte in Protokollen

7.3.1 Übersicht

Objekte im Editor "Protokolle"

Im Editor "Protokolle" stehen Ihnen verschiedene Objekte zur Verfügung:

- Objekte zur formalen Gestaltung eines Protokolls
  z. B. "Seitenzahl", "Linie" oder "Kreis"
- Objekte zur Ausgabe von Inhalten
  z. B. Objekte zur Meldeprotokollierung oder zur Ausgabe von Rezepturen

Im folgenden Abschnitt sind protokollspezifische Objekte beschrieben mit besonderen Eigenschaften. Einfache Gestaltungsobjekte werden nicht näher erläutert.

7.3.2 Audit-Protokoll

Verwendung

Mit dem Audit-Protokoll geben Sie den Inhalt des Audit Trails eines Bediengeräts als Protokoll aus.

DatensatzID	Zeitstempel	DeltaZuUTC	BenutzerID
0	18.11.2007 23:00 ObjektID: Anwendung Beschreibung: Dies ist ein Beispiel für eine Beschreibung. Sie darf nicht gekürzt werden und wird vollständig ausgedruckt.	-1:00	System

Voraussetzungen

Das Objekt "Audit-Protokoll" ist nur verfügbar, wenn folgende Voraussetzungen erfüllt sind:

- Das Bediengerät unterstützt eine GMP-konforme Projektierung.
- In den Runtime-Einstellungen des Bediengeräts ist die GMP-konforme Projektierung aktiviert.

Beachten Sie auch, dass die Bedienaktionen und Systemvorgänge nur dann aufgezeichnet werden, wenn ein entsprechendes Archiv projektiert ist.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Sichtbarkeit des Kommentars

7.3.3 Control Daten

Verwendung

Wenn Sie für die Protokollausgabe eines Bildobjekts ein individuell gestaltetes Protokoll projektieren wollen, benötigen Sie Protokollobjekte.

Das Protokollobjekt "Control Daten" gibt alle in dem Bildobjekt enthaltenen Daten in Tabellenform aus – auch die Daten, die in Runtime im Bildobjekt aktuell nicht sichtbar sind.

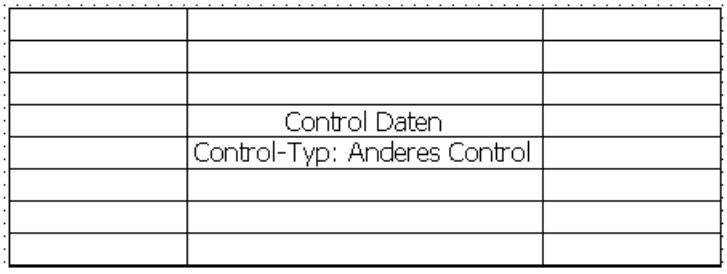

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Rahmen und Farben fest. Insbesondere passen Sie die folgenden Eigenschaften an:

- Control-Typ, dessen Inhalt ausgegeben wird
- Steuerungsobjekte des Bildobjekts, die mit abgebildet werden
- Position & Größe

Control-Typ

Im Inspektorfenster legen Sie unter " Eigenschaften > Eigenschaften > Allgemein > Control-Typ" fest, ob der Wert an der Linealposition, der Inhalt des Objekts oder das Objekt ausgegeben wird.

- Bei der "f(x)-Kurvenanzeige" und der "f(t)-Kurvenanzeige" wählen Sie zwischen der Option "Lineal" und "Anderes Control".
- Für alle anderen Bildobjekte selektieren Sie "Anderes Control".

Steuerungsobjekte des Bildobjekts

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Inhalt" fest, ob die Symbolleiste, die Statusleiste und die Bildlaufleiste mit abgebildet werden.

Größe & Position

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Layout" die Größe und Position des Protokollobjekts fest.

Im Protokoll wird die Tabelle in der Breite des Protokollobjekts ausgegeben. Projektieren Sie das Protokollobjekt in einer geeigneten Breite.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.4 Control Hardcopy

Verwendung

Wenn Sie für die Protokollausgabe eines Bildobjekts ein individuell gestaltetes Protokoll projektieren wollen, benötigen Sie Protokollobjekte.

Das Protokollobjekt "Control Hardcopy" gibt einen Schnappschuss des Bildobjekts aus. Dabei werden nur die Daten ausgegeben, die in Runtime im Bildobjekt sichtbar sind.

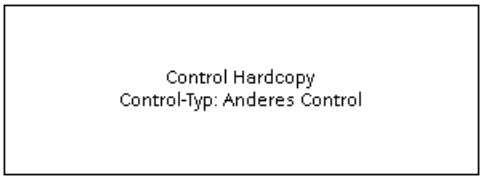

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Rahmen und Farben fest.

Insbesondere passen Sie die folgenden Eigenschaften an:

- Control-Typ, dessen Inhalt ausgegeben wird
- Welche Steuerungsobjekte des Bildobjekts mit abgebildet werden.
- Größe & Position

Control-Typ

Im Inspektorfenster legen Sie unter " Eigenschaften > Eigenschaften > Allgemein > Control-Typ" fest, ob der Wert an der Linealposition, der Inhalt des Objekts oder das Objekt ausgegeben wird.

- Bei der "f(x)-Kurvenanzeige" und der "f(t)-Kurvenanzeige" wählen Sie zwischen der Option "Lineal" und "Anderes Control".
- Für alle anderen Bildobjekte selektieren Sie "Anderes Control".

Steuerungsobjekte des Bildobjekts

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Inhalt" fest, ob die Symbolleiste, die Statusleiste und die Bildlaufleiste im Protokoll mit abgebildet werden.

Größe & Position

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Layout" die Größe und Position des Protokollobjekts fest.

Im Protokoll wird das Bildobjekt in den Rahmen des Protokollobjekts eingepasst. Projektieren Sie das Protokollobjekt in einer geeigneten Größe.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.5 Control Informationen

Verwendung

Wenn Sie für die Protokollausgabe eines Bildobjekts ein individuell gestaltetes Protokoll projektieren wollen, benötigen Sie Protokollobjekte.

Das Protokollobjekt "Control Informationen" gibt den Fenstertitel, den Namen des Bildobjekts oder den Namen des Bildes aus. Auf diese Weise geben Sie z. B. Datenquelle im Protokoll mit aus.

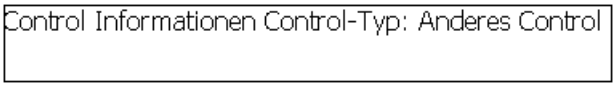

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Rahmen, Farben, Schriftart und Schriftgröße fest. Insbesondere passen Sie die folgenden Eigenschaften an:

- Control-Typ, dessen Inhalt ausgegeben wird
- Inhalt

Control-Typ

Im Inspektorfenster legen Sie unter " Eigenschaften > Eigenschaften > Allgemein > Control-Typ" fest, ob der Wert an der Linealposition, der Inhalt des Objekts oder das Objekt ausgegeben wird.

- Bei der "f(x)-Kurvenanzeige" und der "f(t)-Kurvenanzeige" wählen Sie zwischen der Option "Lineal" und "Anderes Control".
- Für alle anderen Bildobjekte selektieren Sie "Anderes Control".

Inhalt

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Inhalt" fest, welche Informationen das Protokollobjekt ausgibt:

- Fenstertitel
- Objektname
- Bildname

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.6 CSV-Provider Kurven

Verwendung

Mit dem Objekt "CSV-Provider Kurven" geben Sie Daten aus einer Datei im Format "*.csv" als Kurven im Protokoll aus.

Abhängig von der ausgegebenen Datenmenge werden im Protokoll automatisch Seitenumbrüche eingefügt. Die projektierte Breite wird im Protokoll nicht verändert. Nachfolgende Objekte werden automatisch verschoben.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Datenquelle

Datenquelle

Unter "Eigenschaften > Eigenschaften > Allgemein > Provider" legen Sie die Datenquelle fest. Sie können eine CSV-Datei auswählen oder die Auswahl der CSV-Datei mit einer internen Variablen dynamisieren. Auf diese Weise ändern Sie die Ausgabe der Daten direkt am Bediengerät. Die Variable muss vom Datentyp "String" oder "WString" sein.

Hinweis

Für die Darstellung von Kurven müssen die Daten in der CSV-Datei in einer festgelegten Struktur vorliegen.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.7 CSV-Provider Tabellen

Verwendung

Mit dem Objekt "CSV-Provider Tabellen" geben Sie Daten aus einer Datei im Format "*.csv" als Tabelle im Protokoll aus.

Abhängig von der ausgegebenen Datenmenge werden im Protokoll automatisch Seitenumbrüche eingefügt. Die projektierte Breite wird im Protokoll nicht verändert. Nachfolgende Objekte werden automatisch verschoben.

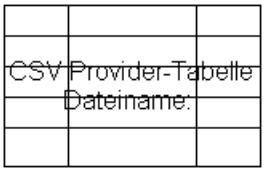

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Datenquelle
- Schrift aus Datenquelle übernehmen

Datenquelle

Unter "Eigenschaften > Eigenschaften > Allgemein > Provider" legen Sie die Datenquelle fest. Sie können eine CSV-Datei auswählen oder die Auswahl der CSV-Datei mit einer internen Variablen dynamisieren. Auf diese Weise ändern Sie die Ausgabe der Daten direkt am Bediengerät. Die Variable muss vom Datentyp "String" oder "WString" sein.

Hinweis

Für die Ausgabe einer Tabelle müssen die Daten in der CSV-Datei in einer festgelegten Struktur vorliegen.

Schrift aus Datenquelle übernehmen

Unter " Eigenschaften > Eigenschaften >Allgemein > Schriftart" legen Sie fest, ob bei der Ausgabe im Protokoll Schriftart und Schriftgröße aus der CSV-Datei übernommen werden.

Wenn Sie die Optionen "Schriftart aus Datenquelle" und "Schriftgröße aus Datenquelle" deaktivieren, werden die Einstellungen unter "Format Text > Stil" verwendet.

Siehe auch

CSV-Datei zur Ausgabe einer Tabelle (Seite 1303)

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.8 Datum/Uhrzeit-Feld

Verwendung

Das Objekt "Datum/Uhrzeit-Feld" zeigt die Systemzeit des Bediengeräts oder Datum und / oder Uhrzeit einer verbundenen Variablen an. Die Darstellung ist abhängig von der am Bediengerät eingestellten Sprache.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie unter "Eigenschaften > Eigenschaften > Allgemein" die folgenden Eigenschaften an:

- Langes Datum-Uhrzeit-Format: Legt fest, in welchem Format Datum und / oder Uhrzeit angezeigt werden.
- Systemzeit: Legt fest, ob die Systemzeit des Bediengeräts oder Datum und / oder Uhrzeit einer verbundenen Variable angezeigt werden.

Langes Datum-Uhrzeit-Format

Option	Beschreibung
"Aktiviert"	Datum und / oder Uhrzeit werden vollständig angezeigt, z. B. "Sunday, December 31, 2000 10:59:59 AM"
"Deaktiviert"	Datum und / oder Uhrzeit werden verkürzt angezeigt, z. B. "12/31/2000 10:59:59 AM"

Systemzeit

Option	Beschreibung
"Aktiviert"	Systemzeit des Bediengeräts wird angezeigt
"Deaktiviert"	Datum und / Uhrzeit der verbundenen Variable werden angezeigt Wählen Sie eine Variable vom Datentyp "DateTime", z. B. eine interne Variable. Informationen über Datentypen, die bei einer Verbindung zu anderen Steuerungen zur Verfügung stehen, finden Sie in der Dokumentation zu den entsprechenden Kommunikationstreibern.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.9 Datum/Uhrzeit-Feld

Verwendung

Das Objekt "Datum/Uhrzeit-Feld gibt Datum und / oder Uhrzeit in einem individuell konfigurierbaren Format im Protokoll aus.

12/31/2000 10:59:59 AM

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzeigeformat von Datum und / oder Uhrzeit

Anzeigeformat von Datum und Uhrzeit

Unter " Eigenschaften > Eigenschaften > Allgemein > Format" legen Sie fest, in welcher Form Datum und / oder Uhrzeit angezeigt werden. Sie können Platzhalter und beliebigen Text miteinander kombinieren.

Die folgende Tabelle zeigt, welche Platzhalter Sie verwenden können:

Platzhalter	Umsetzung	Beispiel
%a	Wochentag in abgekürzter Form	Sun
%A	Wochentag ausgeschrieben	Sunday
%b	Monat in abgekürzter Form	Dec
%B	Monat ausgeschrieben	December
%c	Datum und Zeit im Standardformat	12/31/2000 10:59:59 AM
%d	Tag als Dezimalzahl (01bis 31)	31
%H	Stunden im 24-Stunden-Format (00 bis 23)	22
%I	Stunden im 12-Stunden-Format (01 bis 12)	10
%j	Tag des Jahrs als Dezimalzahl (000 von 366)	366
%m	Monat als Dezimalzahl (01 bis 12)	12
%M	Minute als Dezimalzahl (00 bis 59)	59
%p	Zusatz PM/AM beim 12-Stunden-Format	AM
%S	Sekunde als Dezimalzahl (00 bis 59)	59
%U	Woche im Jahr als Dezimalzahl (01 bis 51)	50
%x	Datum im Standardformat	12/31/2000
%X	Zeit im Standardformat	10:59 AM
%y	Jahr ohne Jahrhundert als Dezimalzahl (00 bis 99)	00
%Y	Jahr mit Jahrhundert als Dezimalzahl	2000
%z	Zeitzone als Name	W. Europe Standard Time
%%	Prozentzeichen	%

Beispiele

Die folgende Tabelle zeigt, wie Sie Platzhalter und Texte miteinander kombinieren können:

Format	Beispiel Ausgabe
Datum: %x Uhrzeit: %H:%M	Datum: 12/31/2000 Uhrzeit: 22:59
Ausdruck vom %a %c	Ausdruck vom Sun 12/06/2010 10:59:59 AM

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.10 E/A-Feld

Verwendung

Das Objekt "E/A-Feld" dient der Ausgabe von Prozesswerten.

Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzeigeformat: Legt das Format fest, in dem die Werte im E/A-Feld ausgegeben werden.

Hinweis

Das Objekt "E/A-Feld" steht Ihnen auch im Editor "Bilder" zur Verfügung. In Protokollen gibt das Objekt Daten ausschließlich aus. Entsprechend konfigurieren Sie bei der Verwendung in Protokollen nur die Ausgabe von Daten.

Anzeigeformat

Das "Anzeigeformat" für die Ausgabe von Werten legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Format" fest.

Darstellung	
"Binär"	Ausgabe von Werten in binärer Schreibweise.
"Datum" *	Ausgabe von Datumsangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.
"Datum/Uhrzeit" *	Ausgabe von Datumsangaben und Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.
"Dezimal" *	Ausgabe von Werten in dezimaler Schreibweise.
"Hexadezimal"	Ausgabe von Werten in hexadezimaler Schreibweise.
"Uhrzeit" *	Ausgabe von Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.
"Zeichenkette"	Ausgaben von Zeichenketten.
*: nicht in Runtime Professional	

Überschneidungen vermeiden bei Ausgabefeldern

Wenn in einem Protokoll mehrere E/A-Felder als Ausgabefelder mit transparentem Hintergrund projektiert sind, überschneiden sich eventuell diese E/A-Felder. Der transparente Teil des einen Feldes überdeckt die Ziffern des anderen Feldes. Dabei kommt es möglicherweise zu Darstellungsproblemen im Protokoll. Um solche Überschneidungen zu vermeiden, setzen Sie in den Objekteigenschaften unter "Eigenschaften > Eigenschaften > Gestaltung" die Rahmen der E/A-Felder auf null. Aktivieren Sie "Eigenschaften > Eigenschaften > Darstellung > Objekt an Inhalt anpassen".

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.11 f(t)-Kurvenanzeige in Protokollen

Verwendung

Mit dem Objekt "f(t)-Kurvenanzeige" stellen Sie Variablenwerte aus dem Archiv in Form von Kurven als Funktionen der Zeit dar. Im Protokoll wird der Status des Kurvenverlaufs zu dem Zeitpunkt abgebildet, in dem der Druckauftrag gestartet wird.

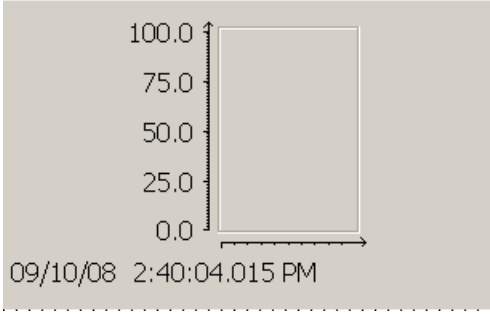

Hinweis

Reduzierter Funktionsumfang bei Ausgabe in Protokollen

Die Konfiguration des Objekts im Editor "Protokolle" unterscheidet sich nur geringfügig von der Konfiguration im Editor "Bilder".

Der Funktionsumfang des Objekts ist an die Verwendung zur Ausgabe in Protokollen angepasst: Zum Beispiel können Sie im Editor "Protokolle" die Bedienelemente in Runtime nicht projektieren.

Besonderheiten für die Ausgabe in Protokollen

Im Gegensatz zur Konfiguration im Bild können Sie die Eigenschaften jeder Kurve mit Variablen dynamisieren, z. B. die Darstellung und die Datenversorgung. Auf diese Weise können Sie z. B. ein Protokoll mehrfach verwenden.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.12 f(x)-Kurvenanzeige in Protokollen

Verwendung

Mit dem Objekt "f(x)-Kurvenanzeige" stellen Sie die Werte einer Variablen als Funktion einer anderen Variable dar. Auf diese Weise stellen Sie z. B. Temperaturverläufe als Funktion des Drucks dar. Im Protokoll wird der Status des Kurvenverlaufs zu dem Zeitpunkt abgebildet, in dem der Druckauftrag gestartet wird.

Zusätzlich können Sie die Kurve mit einer Sollwertkurve vergleichen.

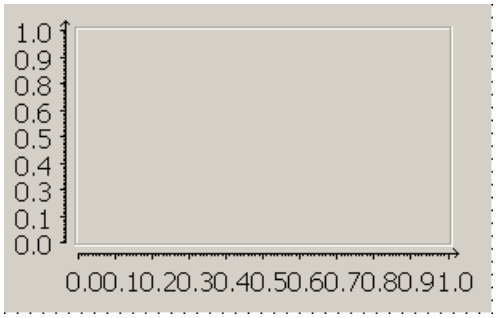

Hinweis

Reduzierter Funktionsumfang bei der Ausgabe in Protokollen

Die Konfiguration des Objekts im Editor "Protokolle" unterscheidet sich nur geringfügig von der Konfiguration im Editor "Bilder".

Der Funktionsumfang des Objekts ist an die Verwendung zur Ausgabe in Protokollen angepasst: Zum Beispiel können Sie im Editor "Protokolle" die Bedienelemente in Runtime nicht projektieren.

Besonderheiten für die Ausgabe in Protokollen

Im Gegensatz zur Konfiguration im Bild können Sie die Eigenschaften jeder Kurve mit Variablen dynamisieren, z. B. die Darstellung und die Datenversorgung. Auf diese Weise können Sie z. B. ein Protokoll mehrfach verwenden.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.13 Grafikanzeige

Verwendung

Das Objekt "Grafikanzeige" dient der Anzeige von Grafiken.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Grafik: Legt die Grafikdatei fest, die in dem Objekt angezeigt wird.
- Grafik anpassen: Legt die automatische Größenanpassung für Objekte mit Grafiken fest.

Grafik einfügen

Im Objekt "Grafikanzeige" verwenden Sie die folgenden Grafikformate: *.bmp, *.tif, *.png, *.ico, *.emf, *.wmf, *.gif, *.svg, *.jpg oder *.jpeg. Sie können Grafiken auch als OLE-Objekte in der Grafikanzeige verwenden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie die Grafik, die Sie einfügen wollen.
   Die Grafik wird rechts in der Vorschau angezeigt.
3. Um die Grafik in die Grafikanzeige einzufügen, klicken Sie auf "Übernehmen".

Grafik anpassen

Im Inspektorfenster wird festgelegt, ob die in einer Grafikanzeige dargestellte Grafik sich in Runtime an die Größe der Grafikanzeige anpasst oder nicht.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie im Bereich "Größenanpassung" eine der folgenden Möglichkeiten:
   - Nicht automatisch anpassen
   - Grafik an Objektgröße anpassen
   - Objektgröße an größte Grafik anpassen
   - Objekt anpassen, Grafik in Originalgröße

7.3.14 Grafisches E/A-Feld

Verwendung

Mit dem Objekt "Grafisches E/A-Feld" zeigen Sie abhängig vom Wert einer Variablen die Grafiken einer Grafikliste an. Variable und Grafikliste projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein".

Hinweis

Das Objekt "Grafisches E/A-Feld" steht Ihnen auch im Editor "Bilder" zur Verfügung. In Protokollen gibt das Objekt Daten ausschließlich aus. Entsprechend konfigurieren Sie bei der Verwendung in Protokollen nur die Ausgabe von Daten.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Farbe, Rahmen, sowie Position und Objektgröße. Insbesondere legen Sie die "Größenanpassung" fest, ob das Objekt bei der Ausgabe im Protokoll an die Grafik angepasst oder die Grafik in das Objekt eingepasst wird.

Siehe auch

Protokoll erstellen (Seite 1294)

7.3.15 Hardcopy

Verwendung

Mit dem Objekt "Hardcopy" geben Sie einen Schnappschuss des Bildschirminhalts im Protokoll aus.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Bereich festlegen

Bereich festlegen

Sie legen unter "Eigenschaften > Eigenschaften > Bereich" fest, welcher Bildschirminhalt im Protokoll ausgegeben wird:

- "Aktuelles Fenster (1)": aktives Fenster
- "Ausgewählter Teilbereich (2)": ausgewählter Bereich
- "Ganzer Bildschirm (0)": kompletter Inhalt

Sie können die Bereichsauswahl mit einer Variablen vom Typ "Zahl" dynamisieren. Die Einstellungen sind wie folgt zugeordnet:

- 0 = "Ganzer Bildschirm"
- 1 = "Aktuelles Fenster"
- 2 = "Ausgewählter Teilbereich"

Für einen "Ausgewählten Teilbereich(0)" legen Sie folgende Eigenschaften fest:

- X-Position des linken, oberen Punkts des Bereichs
- Y-Position des linken, oberen Punkts des Bereichs
- Breite in Pixel
- Höhe in Pixel

Sie können jede Eigenschaft mit einer Variablen dynamisieren.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.16 Meldeanzeige in Protokollen

Verwendung

Das Objekt "Meldeanzeige" zeigt Meldungen an, die während des Prozesses in einer Anlage auftreten. Die Meldeanzeige nutzen Sie auch, um Meldungen in Listen darzustellen. WinCC bietet verschiedene Anzeigen an, z. B. Anzeige "Aktuelle Meldungen" oder "Kurzzeitarchiv".

Im Protokoll wird der Status der Meldeanzeige zu dem Zeitpunkt abgebildet, in dem der Druckauftrag gestartet wird.

Datum	Uhrzeit	Numm	Anwendertext
27/08/10	14:48:44.000	88	TEXT
27/08/10	14:48:45.000	89	TEXT
27/08/10	14:48:46.000	90	TEXT
27/08/10	14:48:47.000	91	TEXT
27/08/10	14:48:48.000	92	TEXT
27/08/10	14:48:49.000	93	TEXT
27/08/10	14:48:50.000	94	TEXT
27/08/10	14:48:51.000	95	TEXT
27/08/10	14:48:52.000	96	TEXT
27/08/10	14:48:53.000	97	TEXT
27/08/10	14:48:54.000	98	TEXT
27/08/10	14:48:55.000	99	TEXT
27/08/10	14:48:56.000	100	TEXT

Hinweis

Reduzierter Funktionsumfang bei Ausgabe in Protokollen

Die Konfiguration des Objekts im Editor "Protokolle" unterscheidet sich nur geringfügig von der Konfiguration im Editor "Bilder".

Der Funktionsumfang des Objekts an die Verwendung zur Ausgabe in Protokollen angepasst: Zum Beispiel können Sie im Editor "Protokolle" die Bedienelemente in Runtime nicht projektieren.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Fortlaufende Protokollierung von Meldungen (Seite 1300)

Protokoll erstellen (Seite 1294)

7.3.17 Meldefolgeprotokoll

Verwendung

Das Objekt "Meldefolgeprotokoll" gibt alle Meldungen des Bediengeräts ab Start von Runtime fortlaufend in chronologischer Reihenfolge aus. Es wird nur im Systemprotokoll "@AlarmSequenceReportForPage" verwendet, das die Ausgabe des WinCC-Meldefolgeprotokolls im Seitenlayout festlegt.

In den Objekteigenschaften legen Sie fest, welche Meldungen ausgegeben werden.

Abhängig von der ausgegebenen Datenmenge werden im Protokoll automatisch Seitenumbrüche eingefügt. Die projektierte Breite wird im Protokoll nicht verändert. Nachfolgende Objekte werden automatisch verschoben.

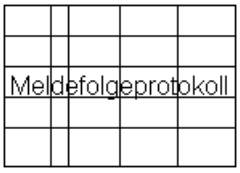

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Fortlaufende Protokollierung von Meldungen (Seite 1300)

Protokoll erstellen (Seite 1294)

Systemprotokolle und Systemdruckaufträge ändern (Seite 1317)

7.3.18 Meldeprotokoll

Verwendung

Mit dem Objekt "Meldeprotokoll" geben Sie Meldungen aus dem Meldepuffer oder dem Meldearchiv im Protokoll aus.

Nr.	Uhrzeit	Zustand	Datum		QGR		PLC
0	12:00:00	KGQ	01.01.1999	Class name	0	*	Device
	normal kursiv normal						
1	12:00:00	KGQ	01.01.1999	Class name	1	*	Device
	normal blinken normal						
2	12:00:00	KGQ	01.01.1999	Class name	2	*	Device
	Message Text						
3	12:00:00	KGQ	01.01.1999	Class name	3	*	Device
	normal bold normal						
4	12:00:00	KGQ	01.01.1999	Class name	4	*	Device
	normal underline normal						

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Quelle
- Meldeklassen
- Zeitbereich
- Weitere Einstellungen für die Anzeige
- Sichtbare Spalten

Quelle festlegen

Unter "Eigenschaften > Eigenschaften > Allgemein > Einstellungen > Quelle" legen Sie fest, ob Meldungen aus dem Meldepuffer oder aus dem Meldearchiv angezeigt werden. Das Meldearchiv wählen Sie unter "Meldearchiv" aus.

Hinweis

Die Quelle "Meldearchiv" ist nur verfügbar, wenn das Bediengerät Archive unterstützt.

Meldeklassen festlegen

Unter "Eigenschaften > Eigenschaften > Allgemein > Meldeklassen" aktivieren Sie die Meldeklassen, deren Meldungen im Protokoll ausgegeben werden.

Zeitbereich festlegen

Wenn Sie nur Meldungen aus einem bestimmten Zeitbereich ausgeben wollen, wählen Sie unter " Eigenschaften > Eigenschaften > Allgemein > Zeitbereich" je eine Variable für den Anfang und das Ende des Zeitbereichs. Die Variable muss vom Typ "DateTime" sein.

Weitere Einstellungen für die Anzeige

Unter " Eigenschaften > Eigenschaften > Allgemein > Einstellungen" legen Sie die folgenden Einstellungen für die Anzeige im Meldeprotokoll fest:

- Unter "Sortierung" legen Sie fest, ob die älteste oder die neueste Meldung zuerst angezeigt wird.
- Unter "Zeilen pro Eintrag" legen Sie fest, wie viele Zeilen pro Meldung zur Verfügung stehen. Die erforderliche Anzahl der Zeilen ist von folgenden Faktoren abhängig:
  - Anzahl und Breite der gewählten Spalten für die Ausgabe
  - verwendete Schriftgröße
  - Papierformat des Druckers
- Mit "Überschrift sichtbar" aktivieren Sie die Anzeige von Spaltenüberschriften.
- Mit "Millisek. anzeigen" aktivieren Sie die Anzeige von Millisekunden bei Zeitangaben.

Sichtbare Spalten

Unter " Eigenschaften > Eigenschaften > Darstellung > Sichtbare Spalten" aktivieren Sie die Spalten, die im Meldeprotokoll angezeigt werden.

Siehe auch

Protokollieren von Meldungen projektieren (Seite 909)

Protokoll erstellen (Seite 1294)

7.3.19 ODBC-Datenbankfeld

Verwendung

Mit dem Objekt "ODBC-Datenbankfeld" geben Sie über die ODBC-Schnittstelle Daten aus einer einzelnen Tabellenzelle im Protokoll aus. Über die ODBC-Schnittstelle stellen Sie Daten einheitlich in einem Protokoll dar, die nicht aus WinCC stammen.

Abhängig von der ausgegebenen Datenmenge werden im Protokoll automatisch Seitenumbrüche eingefügt. Die projektierte Breite wird im Protokoll nicht verändert. Nachfolgende Objekte werden automatisch verschoben.

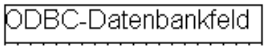

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Verbindungseinstellungen für die SQL-Verbindung
- Datenauswahl für das SQL-Statement

Textformat

Unter "Eigenschaften > Eigenschaften > Textformat > Ausrichtung" legen Sie die Ausrichtung für den angezeigten Text fest. Sie haben die Möglichkeit, den Text horizontal links, rechts oder zentriert darzustellen.

Hinweis

Die Einstellung "Ausrichtung > Vertikal" ist in Runtime nicht wirksam.

Verbindungseinstellungen

Unter "Eigenschaften > Eigenschaften > SQL-Verbindung" legen Sie Datenquelle, Benutzernamen und Kennwort für die SQL-Verbindung fest. Sie können alle Einstellungen mit Variablen dynamisieren. Auf diese Weise ändern Sie die Ausgabe der Daten direkt am Bediengerät.

Datenauswahl

Unter "Eigenschaften > Eigenschaften > SQL-Statement " legen Sie den Datenbereich fest, den Sie im Protokoll darstellen wollen. Geben Sie ein "Select-Statement" ein.

Geben Sie das gewünschte "Select-Statement" in das gleichnamige Textfeld ein oder wählen Sie die "Variable" aus, die das Select-Statement liefert. Die Variable muss vom Typ "String" oder "WString" sein.

Sie können das Select-Statement im gleichnamigen Eingabefeld mit Hilfe von Variablen dynamisieren. Dazu wählen Sie an der gewünschten Textstelle im Kontextmenü "Insert reportfield" und wählen dann die gewünschte Variable aus.

Hinweis

Geben Sie ausschließlich Select-Statements ein.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.20 ODBC-Datenbanktabelle

Verwendung

Mit dem Objekt "ODBC-Datenbanktabelle" geben Sie über die ODBC-Schnittstelle Daten aus einer Tabelle im Protokoll aus. Über die ODBC-Schnittstelle stellen Sie Daten einheitlich in einem Protokoll dar, die nicht aus WinCC stammen.

Abhängig von der ausgegebenen Datenmenge werden im Protokoll automatisch Seitenumbrüche eingefügt. Die projektierte Breite wird im Protokoll nicht verändert. Nachfolgende Objekte werden automatisch verschoben.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Verbindungseinstellungen für die SQL-Verbindung
- Datenauswahl für das SQL-Statement
- Darstellung der Tabelle, z. B. Anzahl Spalten und Umlaufliste

Verbindungseinstellungen

Unter "Eigenschaften > Eigenschaften > SQL-Verbindung" legen Sie Datenquelle, Benutzernamen und Kennwort für die SQL-Verbindung fest. Sie können alle Einstellungen mit Variablen dynamisieren. Auf diese Weise ändern Sie die Ausgabe der Daten direkt am Bediengerät.

Datenauswahl

Unter "Eigenschaften > Eigenschaften > SQL-Statement" legen Sie den Datenbereich fest, den Sie im Protokoll darstellen wollen.

Geben Sie das gewünschte "Select-Statement" in das gleichnamige Textfeld ein oder wählen Sie die "Variable" aus, die das Select-Statement liefert. Die Variable muss vom Typ "String" oder "WString" sein.

Sie können das Select-Statement im gleichnamigen Eingabefeld mit Hilfe von Variablen dynamisieren. Dazu wählen Sie an der gewünschten Textstelle im Kontextmenü "Insert reportfield" und wählen dann die gewünschte Variable aus.

Hinweis

Geben Sie ausschließlich Select-Statements ein.

Umlaufliste

Im Inspektorfenster legen Sie unter "Eigenschaften > Eigenschaften > Tabelle" mit der Option "Umlaufliste" fest, wie die Daten auf die Spalten der Tabelle verteilt werden.

- "Umlaufliste" deaktiviert
  Die Tabelle wird zeilenweise von oben nach unten gefüllt.
  Innerhalb einer Zeile wird in der 1. Spalte ein Wert ausgegeben. Die anderen Spalten bleiben leer.
- "Umlaufliste" aktiviert
  Die Tabelle wird zeilenweise von oben nach unten gefüllt.
  Innerhalb einer Zeile werden alle Spalten von links nach rechts gefüllt.

Hinweis

Die Option ist nur dann wirksam, wenn die Daten einspaltig geliefert werden und die projektierte Tabelle mehrspaltig ist.

Beispiel:

Sie projektieren eine Tabelle mit 3 Spalten. Im Bildobjekt sind zum Zeitpunkt der Protokollausgabe folgende Daten enthalten: A1, A2, A3, B1, B2, B3, C1, C3, D4

Die folgende Tabelle zeigt das Ergebnis mit deaktivierter Option "Umlaufliste":

Spalte 1	Spalte 2	Spalte 3
A1		
A2		
A3		
B1		
B2		
...		

Die folgende Tabelle zeigt das Ergebnis aktivierter Option "Umlaufliste":

Spalte 1	Spalte 2	Spalte 3
A1	A2	A3
B1	B2	B3
C1	C3	D4

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.21 Projektname

Verwendung

Mit dem Objekt "Projektname" geben Sie den Namen des aktuellen Projekts im Protokoll aus.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Format des Projektnamens

Format des Projektnamens

Unter " Eigenschaften > Eigenschaften > Format Text > Stil" legen Sie im Feld "Format" fest, in welcher Form der Projektname ausgegeben wird. Verwenden Sie Platzhalter für den Projektnamen und geben Sie bei Bedarf einen beliebigen Text ein.

Die folgende Tabelle zeigt, welche Platzhalter Sie verwenden können:

Platzhalter	Wert
"%R"	Pfad und Projektname
"%r"	Projektname (ohne Pfadangabe)

Beispiele

Die folgende Tabelle zeigt anhand einiger Beispiele, wie Sie Platzhalter und Texte miteinander kombinieren können:

Format	Beispiel Ausgabe
Projekt: %R	Projekt: D:\WinCC\Projekte\MP277_Mixing
Erstellt für Projekt "%r"	Erstellt für Projekt "MP277_Mixing"

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.22 Protokollname

Verwendung

Mit dem Objekt "Protokollname" geben Sie den Protokollnamen im Protokoll aus.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Format des Protokollnamens

Format des Protokollnamens

Unter " Eigenschaften > Eigenschaften > Format Text > Stil" legen Sie im Feld "Format" fest, in welcher Form der Protokollname ausgegeben wird. Verwenden Sie Platzhalter für den Protokollnamen und geben Sie bei Bedarf einen beliebigen Text ein.

Die folgende Tabelle zeigt, welche Platzhalter Sie verwenden können:

Platzhalter	Wert
"%L" oder "%l"	Protokollname

Beispiel

Die folgende Tabelle zeigt, wie Sie Platzhalter und Texte miteinander kombinieren können:

Format	Beispiel Ausgabe
Protokoll: %L	Protokoll: Meldungen

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.23 Rezepturanzeige in Protokollen

Verwendung

Mit dem Objekt "Rezepturanzeige" zeigen Sie Rezepturelemente in Tabellenform an. Im Protokoll wird der Status der Rezepturanzeige zu dem Zeitpunkt abgebildet, in dem der Druckauftrag gestartet wird.

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	

Hinweis

Reduzierter Funktionsumfang bei Ausgabe in Protokollen

Die Konfiguration des Objekts im Editor "Protokolle" unterscheidet sich nur geringfügig von der Konfiguration im Editor "Bilder".

Der Funktionsumfang des Objekts ist an die Verwendung zur Ausgabe in Protokollen angepasst: Zum Beispiel können Sie im Editor "Protokolle" die Bedienelemente in Runtime nicht projektieren.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Definition und Einsatzgebiete (Seite 1210)

Protokoll erstellen (Seite 1294)

7.3.24 Rezepturprotokoll

Verwendung

Mit dem Objekt "Rezepturprotokoll" geben Sie Elemente von Rezepturdatensätzen im Protokoll aus.

Rezeptur Nummer: 1 Name: Rezeptur_1

Datensatz Nummer: 1 Name: Datensatz_1

Variable:	Name:	Typ:	Wert:
Variable_Number_0	Entry_Number_01	ULONG	689874030
Variable_Number_0	Entry_Number_02	LONG	-99253259
Variable_Number_0	Entry_Number_01	SHORT	-9
Variable_Number_0	Entry_Number_03	USHORT	8172
Variable_Flag_05	Entry_Flag_05	BOOL	false
Variable_String_06	Entry_String_06	STRING	WinCC RT
Variable_Date_07	Entry_Date_07	DATETIME	01.01.1999 00:00:00
Variable_Number_0	Entry_Number_08	LONG	6968192
Variable_Number_0	Entry_Number_09	ULONG	9903155
Variable_Flag_10	Entry_Flag_10	BOOL	true

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie folgende Einstellungen an:

- Rezeptur
- Datensatz
- Format
- Sichtbare Einträge

Rezeptur auswählen

Unter "Eigenschaften > Eigenschaften > Allgemein > Rezeptur" legen Sie fest, welche Rezepturen im Rezepturprotokoll ausgegeben werden. Sie können die Rezepturen über den Rezepturnamen oder einen Bereich von Rezepturnummern festlegen. Unter "Erste Rezeptur" und "Letzte Rezeptur" geben Sie entweder einen Wert ein oder wählen eine Variable aus.

Alternativ lassen Sie alle Rezepturen anzeigen.

Datensatz auswählen

Unter "Eigenschaften > Eigenschaften > Allgemein > Datensatz" legen Sie fest, welche Datensätze der gewählten Rezepturen im Rezepturprotokoll ausgegeben werden. Sie können die Datensätze über den Datensatznamen oder einen Bereich von Datensatznummern festlegen. Unter "Erster Datensatz" und "Letzter Datensatz" geben Sie entweder einen Wert ein oder wählen eine Variable aus.

Alternativ lassen Sie alle Datensätze der gewählten Rezepturen anzeigen.

Format

Unter "Eigenschaften > Eigenschaften > Darstellung > Einstellungen" legen Sie im Feld "Format" fest, ob die Datensätze als Tabelle im Spaltenlayout oder als Zeilenprotokoll ausgegeben werden.

WinCC aktualisiert die Vorschau in der Detailseite entsprechend.

Sichtbare Einträge

Unter "Eigenschaften > Eigenschaften > Darstellung > Sichtbare Einträge" aktivieren Sie die Spalten, die im Rezepturprotokoll angezeigt werden. Mit "Überschriften anzeigen" aktivieren Sie die Anzeige von Spaltenüberschriften.

Siehe auch

Definition und Einsatzgebiete (Seite 1210)

Protokoll erstellen (Seite 1294)

7.3.25 Seitenzahl

Verwendung

Mit dem Objekt "Seitenzahl" geben Sie die aktuelle Seitennummer im Protokoll aus.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Farbe, Schriftart sowie Position und Objektgröße. Insbesondere legen Sie die "Größenanpassung" fest.

Größenanpassung

Unter "Eigenschaften > Eigenschaften > Darstellung > Größenanpassung" legen Sie mithilfe der Option "Objekt an Inhalt anpassen" fest, ob WinCC die Objektgröße an den Feldinhalt anpasst:

- Option "Objekt an Inhalt anpassen" aktiviert:
  WinCC passt die Objektgröße automatisch an das konfigurierte Format an. Schriftgröße und Feldlänge legen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Text" fest. Sie können das Objekt im Arbeitsbereich verschieben, jedoch nicht die Objektgröße ändern. Bei der Protokollausgabe wird der gesamte Feldinhalt zum Zeitpunkt der Protokollausgabe ausgegeben.
- Option "Objekt an Inhalt anpassen" nicht aktiviert:
  Sie legen die Feldgröße selbst fest. WinCC verändert die Größe des Feldes bei der Protokollausgabe nicht. Eventuell werden nicht alle Inhalte des Feldes im Protokoll ausgegeben. Projektieren Sie das Feld deshalb ausreichend groß.

Siehe auch

Protokoll erstellen (Seite 1294)

7.3.26 Seitenzahl

Verwendung

Mit dem Objekt "Seitenzahl" geben Sie die aktuelle Seitennummer oder die Gesamtanzahl der Seiten des Protokolls in der Kopf- oder Fußzeile des Protokolls aus.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Farbe, Schriftart, Rahmen, Ausrichtung, sowie Position und Objektgröße. Insbesondere legen Sie das Format der Seitenzahl fest.

Format

Unter "Eigenschaften > Allgemein > Text > Format" legen Sie fest, in welchem Format die Seitenzahl ausgegeben wird. Verwenden Sie Platzhalter für die Seitenzahlen und geben Sie bei Bedarf einen beliebigen Text ein.

Die folgende Tabelle zeigt, welche Platzhalter Sie verwenden können:

Platzhalter	Wert
"%N" oder "%n"	Aktuelle Seitennummer
"%T" oder "%t"	Gesamtanzahl der Seiten des Protokolls

Beispiele

Die folgende Tabelle zeigt anhand einiger Beispiele, wie Sie Platzhalter und Texte miteinander kombinieren können:

Format	Beispiel Ausgabe
Seite %n von %t	Seite 2 von 10
%n/%t	2/10

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.27 Symbolisches E/A-Feld

Verwendung

Mit dem Objekt "Symbolisches E/A-Feld" zeigen Sie abhängig vom Wert einer Variablen die Inhalte einer Textliste in Protokollen an. Variable und Textliste projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein".

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Farbe, Rahmen, Schriftart sowie Position und Objektgröße. Insbesondere legen Sie bei der "Größenanpassung" fest, ob die Objektgröße bei der Ausgabe im Protokoll an den Text angepasst wird oder nicht.

Hinweis

Das Objekt "symbolisches E/A-Feld" steht Ihnen auch im Editor "Bilder" zur Verfügung. In Protokollen gibt das Objekt Daten ausschließlich aus. Entsprechend konfigurieren Sie bei der Verwendung in Protokollen nur die Ausgabe von Daten.

Siehe auch

Protokoll erstellen (Seite 1294)

7.3.28 Tabellenanzeige in Protokollen

Verwendung

Mit dem Objekt "Tabellenanzeige" stellen Sie die Prozessdaten in einer Tabelle dar. Jede Tabellenzeile repräsentiert dabei den Status der ausgewählten Prozessvariablen zu einem bestimmten Zeitpunkt.

Im Protokoll wird der Status der Tabelle zu dem Zeitpunkt abgebildet, in dem der Druckauftrag gestartet wird.

Zeitspalte	Wertespalte	

Hinweis

Reduzierter Funktionsumfang bei Ausgabe in Protokollen

Die Konfiguration des Objekts im Editor "Protokolle" unterscheidet sich nur geringfügig von der Konfiguration im Editor "Bilder".

Der Funktionsumfang des Objekts ist an die Verwendung zur Ausgabe in Protokollen angepasst: Zum Beispiel können Sie im Editor "Protokolle" die Bedienelemente in Runtime nicht projektieren.

Besonderheiten für die Ausgabe in Protokollen

Im Gegensatz zur Konfiguration im Bild können Sie die Eigenschaften der Zeit- und Wertespalten mit Variablen dynamisieren. Auf diese Weise können Sie z. B. ein Protokoll mehrfach verwenden.

Siehe auch

Werkzeuge zur Protokollgestaltung (Seite 1296)

Protokoll erstellen (Seite 1294)

7.3.29 Textfeld

Verwendung

Das Objekt "Textfeld" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.

Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Text: Legt den Text für das Textfeld fest.
- Größe des Textfelds: Legt fest, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

Text

Der Text für das Textfeld wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie einen Text ein.
   Für mehrzeiligen Text können Sie einen Zeilenumbruch durch Drücken der Tastenkombination <Umschalttaste + Enter> setzen.

Größe des Textfelds

Im Inspektorfenster wird festgelegt, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Größenanpassung > Größe an Inhalt anpassen".

Eine Benutzerverwaltung projektieren 8

8.1 Grundlagen

8.1.1 Einsatzgebiet der Benutzerverwaltung

Prinzip

Der Zugriffsschutz regelt in Runtime den Zugriff auf Daten und Funktionen. So schützen Sie Ihre Anwendungen vor unbefugtem Bedienen. Sie schränken bereits beim Erstellen Ihres Projekts sicherheitsrelevante Bedienungen auf spezielle Benutzergruppen ein. Dazu richten Sie Benutzer und Benutzergruppen ein, die Sie mit charakteristischen Zugriffsrechten ausstatten, den Berechtigungen. An sicherheitsrelevanten Objekten projektieren Sie dann die zur Bedienung erforderlichen Berechtigungen. Bediener haben z. B. nur Zugriff auf bestimmte Bedienobjekte. Inbetriebsetzer haben z. B. uneingeschränkten Zugriff in Runtime.

Definition

Sie administrieren Benutzer, Benutzergruppen und Berechtigungen zentral in der Benutzerverwaltung von WinCC. Sie übertragen Benutzer und Benutzergruppen zusammen mit dem Projekt auf das Bediengerät. Die Benutzer und Kennwörter verwalten Sie an dem Bediengerät über die Benutzeranzeige.

Anwendungsbeispiel

Sie projektieren eine Berechtigung "Service", damit nur Servicetechniker Zugriff auf die Einstellparameter haben. Die Berechtigung ordnen Sie der Benutzergruppe "Servicetechniker" zu. Alle Mitglieder dieser Gruppe bedienen so die geschützten Einstellparameter.

ACHTUNG
Der Zugriffsschutz schützt nicht vor Fehlbedienungen. Sie müssen selbst dafür sorgen, dass nur entsprechend ausgebildetes und autorisiertes Personal Anlagen und Maschinen konstruiert, in Betrieb setzt, bedient und wartet. Der Zugriffsschutz ist nicht dazu geeignet, Arbeitsabläufe zu definieren und deren Einhaltung zu überwachen.

8.1.2 Aufbau der Benutzerverwaltung

Einleitung

Bei einem Projekt in der Fertigungstechnik unterscheidet sich die Umgebung beim Maschinenhersteller von der Umgebung beim Endkunden als Anlagenbetreiber.

Der Maschinenhersteller gestattet dem Benutzer z. B. Herr Meister einen bestimmten Zugriff innerhalb der Applikation bzw. dem Bediengerät. Beim Endkunden existiert jedoch kein Benutzer Meister. Die Benutzer des Endkunden und ihre Aufgaben sind während der Projektierung beim Maschinenhersteller noch nicht bekannt. Die endgültigen Benutzer werden in der Regel erst nach der Inbetriebnahme beim Endkunden festgelegt.

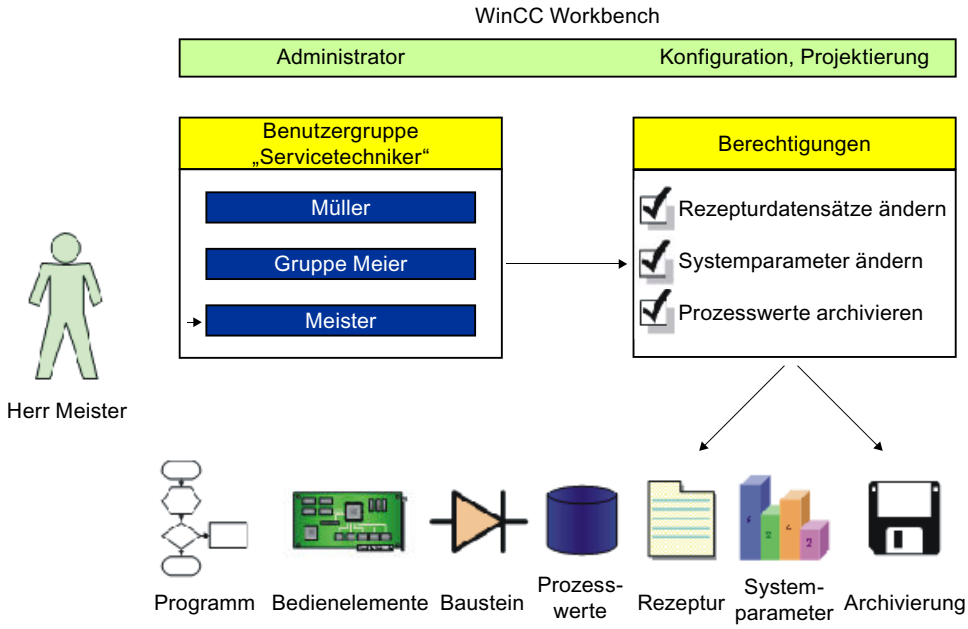

Prinzip

Um den Aufwand bei der Verwaltung gering zu halten, werden Berechtigungen nicht direkt einem einzelnen Benutzer zugewiesen, sondern sind über Benutzergruppen entkoppelt.

Eine Benutzergruppe fasst projektierte Berechtigungen aufgabengerecht zusammen. Z. B. werden alle notwendigen Berechtigungen für einen Serviceeinsatz in einer Gruppe "Servicetechniker" zusammengefasst. Wenn Sie einen Benutzer anlegen, der für den Service zuständig ist, weisen Sie ihn lediglich der Gruppe "Servicetechniker" zu.

Um Benutzer in Runtime zu verwalten, steht Ihnen die Benutzeranzeige zur Verfügung. Die Benutzeranzeige ermöglicht Ihnen in Runtime Benutzer anzulegen, zu löschen und ihnen eine Berechtigung zuzuweisen.

Die Benutzerverwaltung trennt die Administration der Benutzer von der Projektierung der Berechtigungen. Dadurch bietet sie Flexibilität beim Zugriffsschutz.

Eine Vorbelegung der Benutzerverwaltung kann bereits während der Projektierung im Engineering System erfolgen.

8.1.3 Benutzer

Einleitung

Benutzer legen Sie im Editor "Benutzerverwaltung" unter der Registerkarte "Benutzer" an und ordnen sie den Benutzergruppen zu. Die Registerkarte "Benutzer" ist Teil der Benutzerverwaltung von WinCC.

Öffnen

Sie öffnen die Registerkarte "Benutzer", indem Sie im Projektfenster auf "Benutzerverwaltung" doppelklicken.

Arbeitsbereich

Im Arbeitsbereich verwalten Sie die Benutzer:

- Sie erstellen oder löschen Benutzer.
- Sie ordnen Benutzer Benutzergruppen zu.

Hinweis

Sie können einen Benutzer nur einer Benutzergruppe zuordnen.

Hinweis

An einem Bediengerät ist die Anzahl auf 100 Benutzer und einen PLC-User begrenzt. Diese Grenze gilt nicht für PCs. An einem PC wird die maximale Anzahl der Benutzer durch den physikalischen Speicher begrenzt.

Inspektorfenster

Wenn Sie einen Benutzer markieren, können Sie in der Gruppe "Allgemein" das Kennwort ändern. Unter "Automatisches Abmelden" legen Sie fest, ob der Benutzer automatisch von dem Bediengerät abgemeldet wird, wenn nach der eingegebenen Zeit keine Eingabe erfolgt.

Sichern und Wiederherstellen

Die Benutzerdaten werden verschlüsselt und netzausfallsicher am Bediengerät gespeichert.

Die am Bediengerät eingerichteten Benutzer, Kennwörter, Gruppenzuordnungen und Abmeldezeiten können Sie sichern und wiederherstellen. Sie vermeiden so die erneute Eingabe der Daten an einem anderen Bediengerät.

Hinweis

Die aktuell gültigen Benutzerdaten werden in folgenden Fällen überschrieben:

- Abhängig von den Einstellungen bei einem erneuten Laden des Projekts
- Beim Wiederherstellen eines gesicherten Projekts
- Beim Importieren der Benutzerverwaltung über ein Bedienobjekt. Die erneut geladenen oder wiederhergestellten Benutzerdaten und Kennwörter sind sofort gültig.

Hinweis

Die Benutzerdaten aus früheren Runtime-Versionen können in der V14 gesichert und wiederhergestellt werden.

Die Benutzerdaten der V14 können in früheren Versionen nicht wiederhergestellt werden.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

8.1.4 Arbeitsbereich Benutzer

Einleitung

Der Arbeitsbereich "Benutzer" zeigt tabellarisch die Benutzer und Benutzergruppen. Sie administrieren die Benutzer und ordnen sie einer Benutzergruppe zu.

Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Benutzer" und "Benutzergruppen".

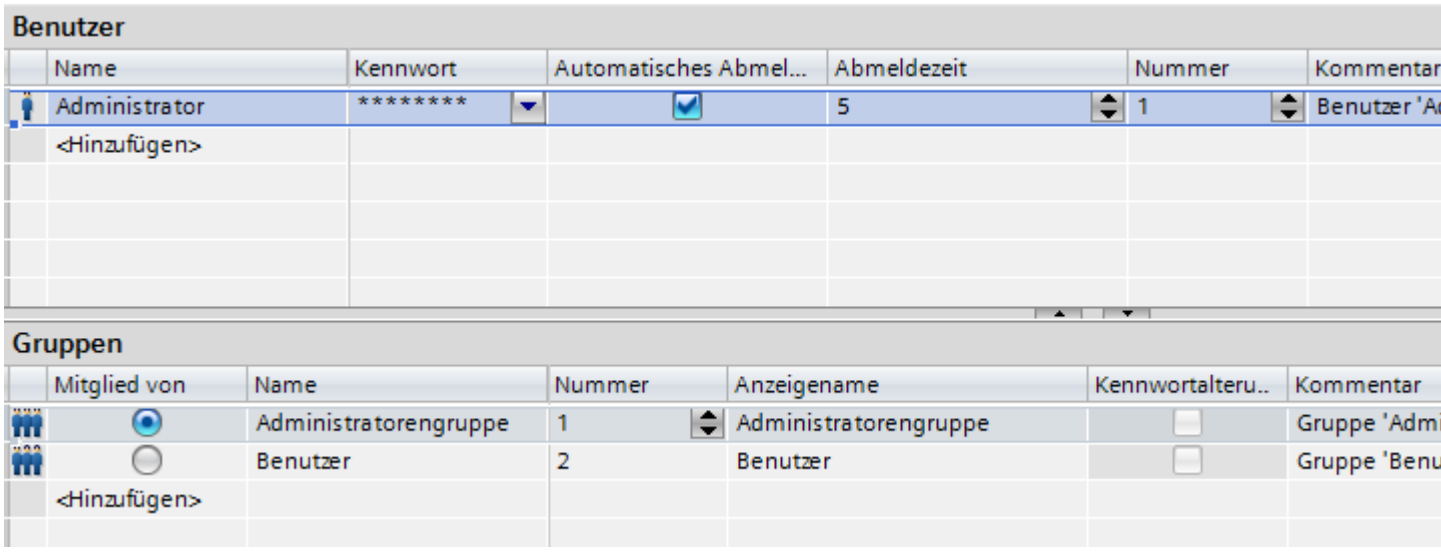

Die Tabelle "Benutzer" zeigt die angelegten Benutzer. Wenn Sie in dieser Tabelle einen Benutzer markieren, zeigt die Tabelle "Gruppen", welcher Benutzergruppe der Benutzer angehört.

Hinweis

Für den Benutzer "Administrator" ist das Kennwort "administrator" voreingestellt. Aus Sicherheitsgründen sollten Sie das Kennwort dieses Benutzers ändern.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Benutzergruppen (Seite 1371)

Arbeitsbereich Benutzergruppen (Seite 1372)

Einstellungen für die Benutzerverwaltung (Seite 1374)

Einstellungen für die Benutzerverwaltung (Seite 1381)

8.1.5 Benutzergruppen

Einleitung

Benutzergruppen und Berechtigungen legen Sie im Editor "Benutzerverwaltung" unter der Registerkarte "Benutzergruppen" an. Die Registerkarte "Benutzergruppen" ist Teil der Benutzerverwaltung von WinCC.

Öffnen

Doppelklicken Sie im Projektfenster auf "Benutzerverwaltung". Wählen Sie die Registerkarte "Benutzergruppen".

Arbeitsbereich

Im Arbeitsbereich verwalten Sie die Benutzergruppen und Berechtigungen:

- Sie legen neue Benutzergruppen und Berechtigungen an oder löschen diese.
- Sie weisen den Benutzergruppen die Berechtigungen zu.

Inspektorfenster

Wenn Sie eine Benutzergruppe oder eine Berechtigung markieren, können Sie in der Gruppe "Allgemein" den Namen editieren. Darüber hinaus fügen Sie in der Gruppe "Kommentar" eine kurze Beschreibung ein.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Arbeitsbereich Benutzer (Seite 1370)

Arbeitsbereich Benutzergruppen (Seite 1372)

Einstellungen für die Benutzerverwaltung (Seite 1374)

Einstellungen für die Benutzerverwaltung (Seite 1381)

8.1.6 Arbeitsbereich Benutzergruppen

Einleitung

Der Arbeitsbereich "Benutzergruppen" zeigt tabellarisch die Gruppen und ihre Berechtigungen. Sie administrieren die Benutzergruppen und weisen ihnen Berechtigungen zu.

Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Gruppen" und "Berechtigungen".

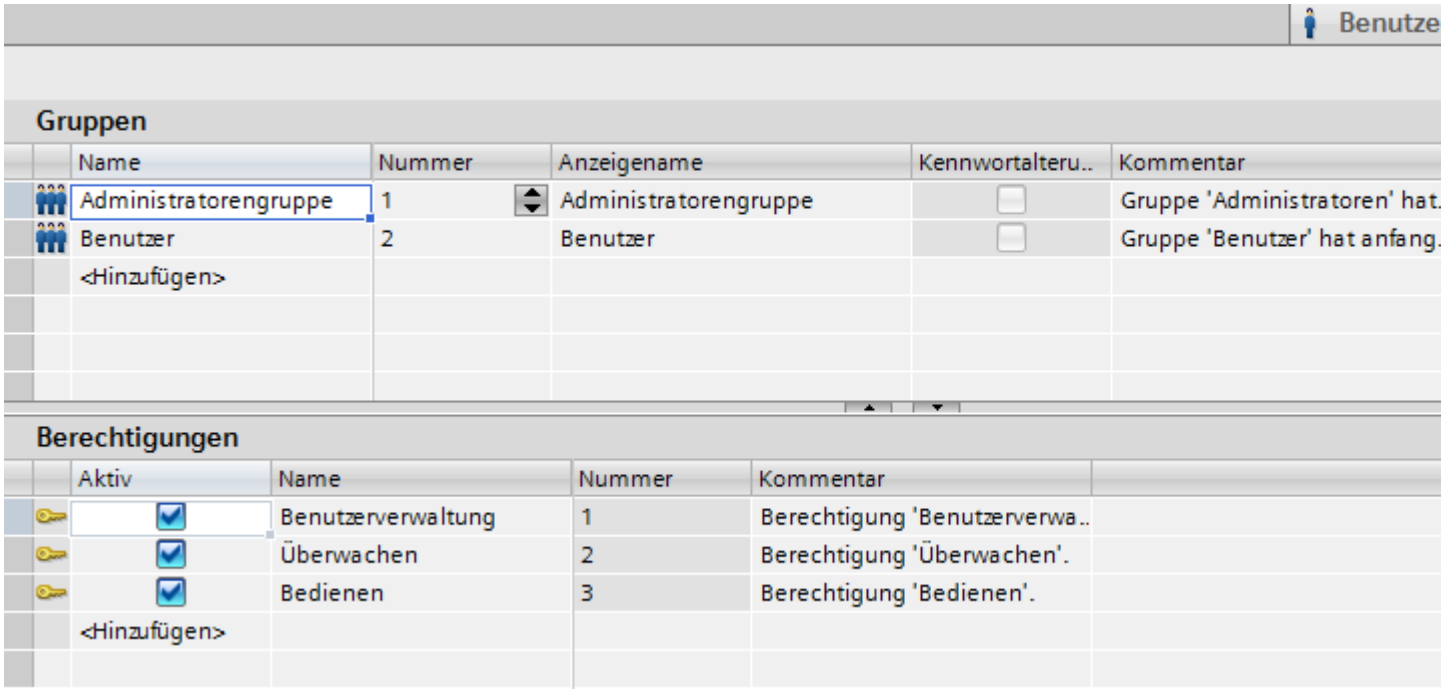

Die Tabelle "Gruppen" zeigt die angelegten Benutzergruppen. Wenn Sie in dieser Tabelle eine Benutzergruppe markieren, zeigt die Tabelle "Berechtigungen" in der Spalte "Aktiv", welche Berechtigungen zugewiesen wurden.

Die Nummer der Benutzergruppe und der Berechtigung vergibt die Benutzerverwaltung. Die Bezeichnungen und Beschreibungen vergeben Sie.

Die Nummern der vordefinierten Berechtigungen sind vorgegeben. Berechtigungen, die Sie erstellen, sind frei editierbar. Achten Sie bei der Vergabe jedoch auf die Eindeutigkeit der Nummern.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Arbeitsbereich Benutzer (Seite 1370)

Benutzergruppen (Seite 1371)

Einstellungen für die Benutzerverwaltung (Seite 1374)

Einstellungen für die Benutzerverwaltung (Seite 1381)

8.1.7 Einstellungen für die Benutzerverwaltung

Einleitung

Im Editor "Runtime-Einstellungen > Benutzerverwaltung" projektieren Sie Sicherheitseinstellungen für die Benutzer und ihre Kennwörter in Runtime.

Öffnen

Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen". Klicken Sie auf "Benutzerverwaltung".

Arbeitsbereich

Im Arbeitsbereich nehmen Sie Einstellungen für die Gültigkeit von Kennwörtern in Runtime vor. Sie bestimmen z. B. die Komplexität des Kennworts.

Hinweis

Wenn das Kennwort aus einem früheren Projekt nicht mehr der Richtlinie entspricht, wird der Benutzer aufgefordert, ein neues Kennwort zu hinterlegen.

Auswirkungen in Runtime

Abhängig von der Konfiguration haben die Sicherheitseinstellungen folgende Auswirkungen in Runtime.

- Gruppe "Allgemein"
  - Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert
    Die im Feld "Anzahl ungültiger Login-Versuche" eingetragene Zahl bestimmt, wie viele Anmeldeversuche ein Benutzer hat, bevor er der Gruppe "Unberechtigt" zugeordnet wird.
    Kontrollkästchen "Grenze für Login-Versuche aktivieren" nicht aktiviert
    Der Benutzer hat in Runtime unendlich viele Anmeldeversuche.
  - Feld "Anzahl ungültiger Login-Versuche"
    Wenn Sie im Feld z. B. "4" eingetragen haben und der vierte Anmeldeversuch des Benutzers fehlschlägt, wird er automatisch der Gruppe "Unberechtigt" zugewiesen.
    Sie können 1 - 9 Versuche eintragen.
  - Kontrollkästchen "Login nur mit Kennwort möglich"
    Wenn das Kontrollkästchen aktiv ist, findet eine Authentifizierung der Benutzer über das Kennwort statt. Der Benutzername ist nicht erforderlich.
    Um eine Zuordnung zu ermöglichen, dürfen Sie daher die Kennwörter nicht doppelt projektieren.

    Hinweis

    Wenn die Option "Login nur mit Kennwort möglich" aktiviert wird, setzt das System die Kennwörter für bereits angelegte Benutzer zurück. Daher ist empfehlenswert, die Option "Login nur mit Kennwort möglich" immer zu aktivieren, bevor Sie Benutzer anlegen.

    Wenn Sie die Benutzerdaten aus früheren Geräteversionen transferieren, in denen die Option "Login nur mit Kennwort möglich" deaktiviert war, ist es möglich, dass mehrere Benutzer mit identischen Kennwörtern hinterlegt wurden. Diese Benutzer werden in der Benutzeranzeige aufgelistet. Jedoch kann sich in Runtime nur derjenige Benutzer anmelden, deren Benutzername nach alphabetischer Sortierung zuerst kommt.

- Gruppe "Hierachieebene"
  - Kontrollkästchen "Gruppenspezifische Rechte für die Benutzerverwaltung"
    Mit aktiven Kontrollkästchen verwalten Administratoren nur die Benutzer, deren Gruppennummer kleiner oder gleich ihrer eigenen ist.
    Ein Administrator, dessen Gruppennummer z. B. 5 ist, kann nur Benutzer verwalten, deren Gruppennummer kleiner oder gleich 5 ist. D. h. auch, dass der Administrator Benutzer nur Gruppen zuordnen kann, die eine Gruppennummer kleiner oder gleich 5 haben.

- Gruppe "Kennwort"
  - Kontrollkästchen "Kennwortalterung aktivieren" aktiviert
    Das Kennwort läuft nach der im Feld "Gültigkeit für Kennwort (Tage)" eingetragenen Anzahl von Tagen ab.
    Im Editor "Benutzergruppen" wird zusätzlich die Spalte "Kennwortalterung" aktiviert. Damit bestimmen Sie gruppenspezifisch, ob Kennwörter ablaufen und ob Kennwortgenerationen gespeichert werden. Für Gruppen, deren Kennwortalterung nicht aktiviert ist, laufen die Kennwörter niemals ab.
  - Feld "Vorwarnzeit (Tage)"
    Der Benutzer wird die eingegebene Anzahl von Tagen vor Ablauf seines Kennworts davon unterrichtet, dass sein Kennwort abläuft.
  - Feld "Kennwortgenerationen"
    Wenn der Benutzer sein Kennwort ändert, muss sich das neue Kennwort von der eingegebenen Anzahl vorheriger Kennwörter unterscheiden. Die mögliche Anzahl der Kennwortgenerationen liegt zwischen 1 und 5.
- Gruppe "Kennwortkomplexität"
  - Kontrollkästchen "Muss Sonderzeichen enthalten" aktiviert
    Der Benutzer muss ein Kennwort mit mindestens einem Sonderzeichen eingeben, unabhängig an welcher Position.
  - Kontrollkästchen "Muss Ziffer enthalten" aktiviert
    Der Benutzer muss ein Kennwort mit mindestens einer Ziffer eingeben, unabhängig an welcher Position.
  - Feld "Minimale Kennwortlänge"
    Der Benutzer muss ein Kennwort mit einer Mindestlänge, von der im Feld "Minimale Kennwortlänge" eingetragenen Zahl, eingeben.
    Die minimale Länge des Kennworts ist 3 Zeichen.

Hinweis

Wenn die Option "Muss Sonderzeichen enthalten" aktiviert wird, setzt das System die Kennwörter für bereits angelegte Benutzer zurück. Daher ist es empfehlenswert, die Option "Muss Sonderzeichen enthalten" immer zu aktivieren, bevor Sie Benutzer anlegen.

Folgende Sonderzeichen sind für Kennwörter zugelassen:

-+.,/*:\"?_()=^!€%~`§'{}[]<>#$;&|@

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Arbeitsbereich Benutzer (Seite 1370)

Benutzergruppen (Seite 1371)

Arbeitsbereich Benutzergruppen (Seite 1372)

8.1.8 Einstellungen für die Benutzerverwaltung

Einleitung

Im Editor " Runtime-Einstellungen > Benutzerverwaltung" projektieren Sie Sicherheitseinstellungen für die Benutzer und ihre Kennwörter in Runtime.

Öffnen

Doppelklicken Sie im Projektfenster auf den Editor " Runtime-Einstellungen". Klicken Sie auf "Benutzerverwaltung".

Arbeitsbereich

Im Arbeitsbereich nehmen Sie Einstellungen für die Gültigkeit von Kennwörtern in Runtime vor. Sie bestimmen z. B. die Komplexität des Kennworts.

Hinweis

Wenn das Kennwort aus einem früheren Projekt nicht mehr der Richtlinie entspricht, wird der Benutzer aufgefordert, ein neues Kennwort zu hinterlegen.

Auswirkungen in Runtime

Abhängig von der Konfiguration haben die Sicherheitseinstellungen folgende Auswirkungen in Runtime.

- Gruppe "Allgemein"
  - Kontrollkästchen "Änderung des Initialkennworts" aktiviert
    Der Benutzer muss bei der ersten Anmeldung das für ihn vom Administrator vergebene Kennwort ändern.
  - Kontrollkästchen "Abmeldezeit ändern" aktiviert
    Für die Änderung der Abmeldezeit genügen einfache Benutzerrechte.
    Die Abmeldezeit ist die Zeitspanne, nach der die Benutzerverwaltung einen Benutzer automatisch abmeldet, wenn keine Eingabe am System erfolgt ist.
    Die Abmeldezeit für den SIMATIC Logon-Benutzer entspricht der des Standard-Benutzers "Administrator".
    Wenn der Benutzer die Abmeldezeit ändert, wird dies im Audit Trail archiviert.
  - Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert
    Die im Feld "Anzahl ungültiger Login-Versuche" eingetragene Zahl bestimmt, wie viele Anmeldeversuche ein Benutzer hat, bevor er der Gruppe "Unberechtigt" zugeordnet wird.
    Kontrollkästchen "Grenze für Login-Versuche aktivieren" nicht aktiviert
    Der Benutzer hat in Runtime unendlich viele Anmeldeversuche.
  - Feld "Anzahl ungültiger Login-Versuche"
    Wenn Sie im Feld z. B. "4" eingetragen haben und der vierte Anmeldeversuch des Benutzers fehlschlägt, wird er automatisch der Gruppe "Unberechtigt" zugewiesen.
    Sie können 1 - 9 Versuche eintragen.
  - Kontrollkästchen "Login nur mit Kennwort möglich"
    Wenn das Kontrollkästchen aktiv ist, findet eine Authentifizierung der Benutzer über das Kennwort statt. Der Benutzername ist nicht erforderlich.
    Um eine Zuordnung zu ermöglichen, dürfen Sie daher die Kennwörter nicht doppelt projektieren.

    Hinweis

    Wenn die Option "Login nur mit Kennwort möglich" aktiviert wird, setzt das System die Kennwörter für bereits angelegte Benutzer zurück. Daher ist empfehlenswert, die Option "Login nur mit Kennwort möglich" immer zu aktivieren, bevor Sie Benutzer anlegen.

    Wenn Sie die Benutzerdaten aus früheren Geräteversionen transferieren, in denen die Option "Login nur mit Kennwort möglich" deaktiviert war, ist es möglich, dass mehrere Benutzer mit identischen Kennwörtern hinterlegt wurden. Diese Benutzer werden in der Benutzeranzeige aufgelistet. Jedoch kann sich in Runtime nur derjenige Benutzer anmelden, deren Benutzername nach alphabetischer Sortierung zuerst kommt.

- Gruppe "Hierachieebene"

  - Kontrollkästchen "Gruppenspezifische Rechte für die Benutzerverwaltung"
    Mit aktiven Kontrollkästchen verwalten Administratoren nur die Benutzer, deren Gruppennummer kleiner oder gleich ihrer eigenen ist.
    Ein Administrator, dessen Gruppennummer z. B. 5 ist, kann nur Benutzer verwalten, deren Gruppennummer kleiner oder gleich 5 ist. D. h. auch, dass der Administrator Benutzer nur Gruppen zuordnen kann, die eine Gruppennummer kleiner oder gleich 5 haben.

- Gruppe "Kennwort"
  - Kontrollkästchen "Kennwortalterung aktivieren" aktiviert
    Das Kennwort läuft nach der im Feld "Gültigkeit für Kennwort (Tage)" eingetragenen Anzahl von Tagen ab.
    Im Editor "Benutzergruppen" wird zusätzlich die Spalte "Kennwortalterung" aktiviert. Damit bestimmen Sie gruppenspezifisch, ob Kennwörter ablaufen und ob Kennwortgenerationen gespeichert werden. Für Gruppen, deren Kennwortalterung nicht aktiviert ist, laufen die Kennwörter niemals ab.
  - Feld "Vorwarnzeit (Tage)"
    Der Benutzer wird die eingegebene Anzahl von Tagen vor Ablauf seines Kennworts davon unterrichtet, dass sein Kennwort abläuft.
  - Feld "Kennwortgenerationen"
    Wenn der Benutzer sein Kennwort ändert, muss sich das neue Kennwort von der eingegebenen Anzahl vorheriger Kennwörter unterscheiden. Die mögliche Anzahl der Kennwortgenerationen liegt zwischen 1 und 5.

- Gruppe "Kennwortkomplexität"
  - Kontrollkästchen "Muss Sonderzeichen enthalten" aktiviert
    Der Benutzer muss ein Kennwort mit mindestens einem Sonderzeichen eingeben, unabhängig an welcher Position.
  - Kontrollkästchen "Muss Ziffer enthalten" aktiviert
    Der Benutzer muss ein Kennwort mit mindestens einer Ziffer eingeben, unabhängig an welcher Position.
  - Feld "Minimale Kennwortlänge"
    Der Benutzer muss ein Kennwort mit einer Mindestlänge, von der im Feld "Minimale Kennwortlänge" eingetragenen Zahl, eingeben.
    Die minimale Länge des Kennworts ist 3 Zeichen.

    Hinweis

    Wenn die Option "Muss Sonderzeichen enthalten" aktiviert wird, setzt das System die Kennwörter für bereits angelegte Benutzer zurück. Daher ist empfehlenswert, die Option "Muss Sonderzeichen enthalten" immer zu aktivieren, bevor Sie Benutzer anlegen.

    Folgende Sonderzeichen sind für Kennwörter zugelassen:

    -+.,/*:\"?_()=^!€%~`§'{}[]<>#$;&|@

- Gruppe "SIMATIC Logon"
  - Kontrollkästchen "SIMATIC Logon aktivieren" aktiviert
    Eine Verbindung zum Server wird hergestellt. Die Autorisierung findet über SIMATIC Logon statt.
  - Kontrollkästchen "Verschlüsselte Übertragung" aktiviert
    Die Daten werden verschlüsselt an den Server übertragen.

Siehe auch

Zentrale Benutzerverwaltung mit SIMATIC Logon (Seite 1393)

Benutzer (Seite 1369)

Arbeitsbereich Benutzer (Seite 1370)

Benutzergruppen (Seite 1371)

Arbeitsbereich Benutzergruppen (Seite 1372)

Einstellungen für die Benutzerverwaltung (Seite 1374)

Einstellungen für die Benutzerverwaltung (Seite 1381)

8.1.9 Einstellungen für die Benutzerverwaltung

Einleitung

Im Editor "Runtime-Einstellungen > Benutzerverwaltung" legen Sie fest, ob sich Benutzer in Runtime dynamisch an einem Bediengerät anmelden. Sie können auch eine zentrale Benutzerverwaltung über SIMATIC Logon aktivieren.

Öffnen

Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen". Klicken Sie auf "Benutzerverwaltung".

Hinweis

Die Ansicht des Arbeitsbereichs ist bediengeräteabhängig.

Arbeitsbereich

Im Arbeitsbereich legen Sie fest, wie sich der Bediener an dem Bediengerät anmeldet:

- Dynamisches Anmelden
  Sie weisen beim dynamischen Anmelden jedem Bediengerät eine Variable zu. Ein Variablenwert wird einem Benutzer zugeordnet. Damit meldet sich der Benutzer in Runtime durch das Setzen des Variablenwerts an einem Bediengerät an, z. B. über einen Schlüsselschalter.
- SIMATIC Logon
  Wenn Sie SIMATIC Logon verwenden, legen Sie den Anzeigenamen fest, der an dem Bediengerät angezeigt wird. SIMATIC Logon unterstützt eine zentrale Benutzerverwaltung. Sobald Sie die Option "SIMATIC Logon" aktivieren, werden die im Editor "Benutzerverwaltung" angelegten Benutzer ignoriert. Der Arbeitsbereich "Benutzer" wird ausgeblendet.
  Nur Gruppen, deren Einstellungen mit den Windows-Gruppen namentlich übereinstimmen, sind von Bedeutung.

  Hinweis

  Wenn Sie SIMATIC Logon verwenden, können Sie die Option "Automatisch Anmelden" nicht nutzen.

Auswirkungen in Runtime

Beim dynamischen Anmelden wird der Benutzer an dem Bediengerät angemeldet, sobald die Variable den korrekten Wert enthält.

Die eingestellte automatische Abmeldezeit wirkt sich nicht auf die Benutzer aus, die sich über eine Login-Variable anmelden.

Bei SIMATIC Logon meldet sich der Bediener am Betriebssystem an. Die Benutzereinstellungen, z. B. Kennwortgültigkeit, werden von SIMATIC Logon aus der Benutzerverwaltung des Betriebssystems gelesen.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Arbeitsbereich Benutzer (Seite 1370)

Benutzergruppen (Seite 1371)

Arbeitsbereich Benutzergruppen (Seite 1372)

Benutzer dynamisch anmelden (Seite 1391)

Zentrale Benutzerverwaltung mit SIMATIC Logon (Seite 1393)

8.1.10 Zielgruppen in der Benutzerverwaltung

Prinzip

Die Zielgruppen stehen beispielhaft für unterschiedliche Personengruppen, die mit der Benutzerverwaltung arbeiten. Folgende vier Zielgruppen lassen sich in der Benutzerverwaltung unterscheiden:

1. Administrator OEM
2. Administrator RT
3. Projekteur
4. Bediener

Als Administrator OEM erstellen Sie z. B. beim Maschinenhersteller im Engineering System die Benutzergruppen, Benutzer und Berechtigungen für Runtime.

Als Administrator RT verwalten Sie in Runtime Benutzer über die "Benutzeranzeige".

Als Projekteur weisen Sie im Engineering System die Berechtigungen den Benutzergruppen zu. Des Weiteren projektieren Sie Berechtigungen an Objekte.

Als Bediener melden Sie sich in Runtime an. Sie haben nur Zugriff auf ein geschütztes Objekt, wenn Sie die erforderliche Berechtigung besitzen.

Hinweis

Die Zielgruppe Administrator RT ist in der Runtime Benutzerverwaltung als vordefinierte Benutzergruppe "Administratorengruppe" bereits angelegt. Im Folgenden werden zur besseren Veranschaulichung die vordefinierten Benutzergruppen und Berechtigungen nicht verwendet.

8.2 Benutzerverwaltung aufbauen

8.2.1 Benutzer für Runtime administrieren

8.2.1.1 Berechtigung erstellen

Einleitung

Sie erstellen eine Berechtigung und weisen sie einer oder mehreren Benutzergruppen zu.

Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Berechtigungen" auf "Hinzufügen".
2. Geben Sie als Name der Berechtigung "Runtime beenden" ein.
3. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

Siehe auch

Benutzergruppe anlegen (Seite 1383)
Benutzergruppen verwalten (Seite 1389)
Nummer der Benutzergruppe ändern (Seite 1391)

8.2.1.2 Benutzergruppe anlegen

Einleitung

Damit Sie nicht jedem Benutzer eine Berechtigung zuweisen, erstellen Sie Benutzergruppen. Sie legen eine Benutzergruppe an, weisen ihr Berechtigungen zu und ordnen ihr Benutzer zu.

Hinweis

Der Name der Benutzergruppe muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

Hinweis

Verwenden von SIMATIC Logon

Stellen Sie sicher, dass die Namen der Benutzergruppen in Windows und WinCC exakt übereinstimmen.

Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Gruppen" auf "Hinzufügen".
2. Geben Sie als "Name" der Benutzergruppe "Bediener" ein.
3. Bei Bedarf ändern Sie die "Nummer" der Benutzergruppe.
4. Geben Sie als "Anzeigename" der Benutzergruppe "Bediener" an.
5. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

In Runtime zeigt die Benutzeranzeige den Anzeigenamen der Benutzergruppe an. Der Anzeigename der Benutzergruppe ist sprachabhängig. Sie projektieren die Bezeichnung in mehreren Sprachen und schalten die Sprache in Runtime um.

Siehe auch

Berechtigung erstellen (Seite 1383)

8.2.1.3 Benutzergruppe anlegen

Einleitung

Damit Sie nicht jedem Benutzer eine Berechtigung zuweisen, erstellen Sie Benutzergruppen. Sie legen eine Benutzergruppe an, weisen ihr Berechtigungen zu und ordnen ihr Benutzer zu.

Hinweis

Verwenden von SIMATIC Logon

Stellen Sie sicher, dass die Namen der Benutzergruppen in Windows und WinCC exakt übereinstimmen.

Hinweis

Der Name der Benutzergruppe muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Gruppen" auf "Hinzufügen".
2. Geben Sie als Namen der Benutzergruppe "Bediener" ein.
3. Bei Bedarf ändern Sie die "Nummer" der Benutzergruppe.
4. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

8.2.1.4 Berechtigung zuweisen

Einleitung

Wenn Sie einer Benutzergruppe eine Berechtigung zuweisen, besitzen alle zugeordneten Benutzer diese Berechtigung.

Voraussetzungen

- Eine Berechtigung "Runtime beenden" ist erstellt.
- Eine Benutzergruppe "Bediener" ist angelegt.
- Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle "Gruppen" auf die Benutzergruppe "Bediener". Die Tabelle "Berechtigungen" zeigt alle Berechtigungen.
2. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Runtime beenden".

   Hinweis

   Die Berechtigung "Runtime beenden" ist nur eine Bezeichnung und hat keinen Bezug zur Funktion "StoppeRuntime". Sie müssen diesen Bezug selbst herstellen. Projektieren Sie dazu an eine Schaltfläche die Systemfunktionen "StoppeRuntime" und wählen als "Berechtigung" "Runtime beenden".

8.2.1.5 Benutzer anlegen

Einleitung

Sie legen einen Benutzer an, damit sich nach dem Laden auf das Gerät Personen mit ihrem Benutzernamen in Runtime anmelden.

Alternativ können Sie Benutzer über die "Benutzeranzeige" in Runtime anlegen und ändern.

Hinweis

Gilt nur für Runtime Professional:

Neue Benutzer, die über einen Web-Client oder einen WebUX-Client auf das System zugreifen sollen, können nur im Engineering System angelegt werden.

Wenn Sie einen Benutzer in Runtime mit der Benutzeranzeige anlegen, ist es nicht möglich, diesem Benutzer ein Web-Startbild zuzuweisen. Somit ist dieser Benutzer nicht für Web-Zugriffe freigeschaltet.

Bestehenden Benutzern können in Runtime ebenfalls keine Web-Zugriffsrechte erteilt werden.

Damit ein angelegter Benutzer Berechtigungen besitzt, müssen Sie ihn einer Benutzergruppe zuordnen und der Benutzergruppe Berechtigungen zuweisen.

Wenn der bei der Anmeldung eingegebene Benutzername mit einem Benutzer in Runtime übereinstimmt, ist die Anmeldung erfolgreich. Darüber hinaus muss das eingegebene Kennwort mit dem hinterlegten Kennwort des Benutzers übereinstimmen.

Hinweis

Beachten Sie die Groß- und Kleinschreibung.

Voraussetzungen

Der Arbeitsbereich "Benutzer" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
2. Geben Sie als Benutzername "Meister" ein.

   Hinweis

   Der Benutzername muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

   Hinweis

   Verwenden Sie bei der Eingabe eines Benutzernamen und eines Kennworts keine Sonderzeichen wie / " § $ % ? ' &.

3. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Ein Dialog zur Eingabe des Kennworts wird geöffnet.
4. Geben Sie das Kennwort des Benutzers ein.
5. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
6. Schließen Sie den Dialog über das Symbol .
7. Wenn der Benutzer nach einer bestimmten Zeitspanne abgemeldet werden soll, aktivieren Sie "Automatisches Abmelden".
8. Klicken Sie in die Spalte "Abmeldezeit". Als "Abmeldezeit " sind 5 Minuten voreingestellt.
9. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

8.2.1.6 Benutzer einer Benutzergruppe zuordnen

Einleitung

Wenn Sie einen Benutzer einer Benutzergruppe zuordnen, besitzt er die Berechtigungen der Benutzergruppe.

Hinweis

Sie müssen einen Benutzer genau einer Benutzergruppe zuordnen. Bei der Konsistenzprüfung und Generierung des Projekts wird die Zuordnung geprüft.

Voraussetzungen

- Der Benutzer "Meister" ist angelegt.
- Eine Benutzergruppe "Bediener" ist angelegt.
- Der Arbeitsbereich "Benutzer" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle "Benutzer" auf den Benutzer "Meister". Die Tabelle "Gruppen" zeigt alle Benutzergruppen.
2. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Bediener".

8.2.1.7 Benutzer verwalten

Einleitung

Im Arbeitsbereich verwalten Sie Benutzer und ordnen Sie Benutzergruppen zu.

Voraussetzungen

Der Arbeitsbereich "Benutzer" ist geöffnet.

Benutzername ändern

1. Doppelklicken Sie in der Tabelle "Benutzer" in der Spalte "Name" auf das Feld des zu ändernden Benutzernamens.
2. Ändern Sie den Benutzernamen.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Benutzernamen.

Kennwort des Benutzers ändern

1. Klicken Sie in der Tabelle "Benutzer" in der Spalte "Kennwort" auf die Schaltfläche .Ein Dialog zur Eingabe des Kennworts wird geöffnet.
2. Geben Sie im Feld "Kennwort eingeben" das neue Kennwort ein.
3. Geben Sie das neue Kennwort im Feld "Kennwort bestätigen" nochmals ein.
4. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" das Kennwort.

Nicht sichtbare Spalten anzeigen lassen

1. Bewegen Sie den Mauszeiger auf die Kopfzeile der Tabelle "Benutzer".
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und aktivieren Sie die Anzeige der Spalte z. B. "Abmeldezeit".

Abmeldezeit des Benutzers ändern

1. Doppelklicken Sie im Bereich "Benutzer" in der Spalte "Abmeldezeit" auf das Feld der zu ändernden Abmeldezeit.
2. Ändern Sie die Abmeldezeit.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Automatisches Abmelden" die Abmeldezeit.

Benutzer löschen

1. Markieren Sie die Zeile des zu löschenden Benutzers.
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

Hinweis

Vordefinierte Benutzer können Sie nicht löschen.

8.2.1.8 Benutzergruppen verwalten

Einleitung

Im Arbeitsbereich verwalten Sie die Benutzergruppen und vergeben Berechtigungen für die Verwendung in Runtime.

Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Namen der Benutzergruppe ändern

1. Doppelklicken Sie in der Tabelle "Gruppen" in der Spalte "Name" auf das Feld des zu ändernden Namens der Benutzergruppe.
2. Ändern Sie den Namen der Benutzergruppe.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Benutzergruppe im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen.

Hinweis

Vordefinierte Benutzergruppen können nicht gelöscht werden.

Nicht sichtbare Spalten anzeigen lassen

1. Bewegen Sie den Mauszeiger auf die Kopfzeile der Tabelle "Gruppen".
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und aktivieren Sie die Anzeige der Spalte z. B. "Anzeigename".

Anzeigename der Benutzergruppe ändern

1. Doppelklicken Sie in die Tabelle "Gruppen" in der Spalte "Anzeigename" auf das Feld des zu ändernden Anzeigenamens der Benutzergruppe.
2. Ändern Sie den Anzeigenamen der Benutzergruppe.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Benutzergruppe im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Anzeigenamen.

Benutzergruppe löschen

1. Markieren Sie die Zeile der zu löschenden Benutzergruppe
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

Hinweis

Vordefinierte Benutzergruppen können nicht gelöscht werden.

Namen der Berechtigung ändern

1. Doppelklicken Sie in der Tabelle "Berechtigungen" in der Spalte "Name" auf das Feld des zu ändernden Namens der Berechtigung.
2. Ändern Sie den Namen der Berechtigung.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Berechtigung im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen.

Berechtigung löschen

1. Markieren Sie die Zeile der zu löschenden Berechtigung
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

Hinweis

Vordefinierte Berechtigungen können nicht gelöscht werden.

Siehe auch

Berechtigung erstellen (Seite 1383)

8.2.1.9 Nummer der Benutzergruppe ändern

Einleitung

Um gruppenspezifische Rechte zu vergeben, editieren Sie die Nummer einer Benutzergruppe.

Voraussetzungen

- Im Editor "Runtime-Einstellungen > Benutzerverwaltung" ist "Gruppenspezifische Rechte für die Benutzerverwaltung" aktiviert.
- Die Benutzergruppe "Bediener" ist erstellt.
- Eine Berechtigung "Benutzerverwaltung" ist erstellt.
- Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle "Gruppen" auf die Benutzergruppe "Bediener"
2. Geben Sie in die Spalte "Nummer" der Benutzergruppe "6" ein.
3. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Benutzerverwaltung".

Ergebnis

Die Mitglieder der Benutzergruppe "Bediener" können nur Benutzer verwalten, deren Gruppennummer kleiner oder gleich 6 ist.

Siehe auch

Berechtigung erstellen (Seite 1383)

8.2.1.10 Benutzer dynamisch anmelden

Einleitung

Um "Dynamisches Anmelden" zu ermöglichen, sind zwei Projektierungsschritte erforderlich:

1. Ein Bediengerät einer projektierten Variable zuordnen.
2. Einen bestimmten Variablenwert einem Benutzer zuordnen.

Hinweis

Wenn Sie SIMATIC Logon aktiviert haben, ist eine dynamische Anmeldung nicht möglich.

Voraussetzungen

- Eine Benutzerverwaltung existiert.
- Eine Variable "Login" vom Typ "Byte" ist angelegt.

Vorgehensweise

Zuordnung Bediengerät - Variable

1. Doppelklicken Sie im Projektfenster auf "Runtime-Einstellungen > Benutzerverwaltung".
2. Doppelklicken Sie im Bereich "Dynamisches Anmelden" in die Tabelle auf "Hinzufügen".
3. Klicken Sie in der Spalte "Bediengerät" auf die Schaltfläche . Ein Dialog zur Auswahl des Bediengeräts wird geöffnet.
4. Wählen Sie ein Bediengerät.
5. Klicken Sie in der Spalte "Variable" auf die Schaltfläche . Ein Dialog zur Auswahl der Variable wird geöffnet.
6. Wählen Sie die Variable "Login".

Zuordnung Benutzer - Variablenwert

1. Öffnen Sie den Arbeitsbereich "Benutzer".
2. Bewegen Sie den Mauszeiger auf die Kopfzeile der Tabelle "Benutzer".
3. Öffnen Sie das Kontextmenü über die rechte Maustaste und aktivieren Sie die Anzeige der Spalten "Dynamisches Anmelden" und "Benutzerkennung".
4. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
5. Geben Sie als Benutzername "Meier" ein.
6. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Ein Dialog zur Eingabe des Kennworts wird geöffnet.
7. Geben Sie das Kennwort des Benutzers ein.
8. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
9. Schließen Sie den Dialog über das Symbol .
10. Aktivieren Sie die Spalte "Dynamisches Anmelden".

Ergebnis

Wenn die Variable den Wert 1 annimmt, wird der Benutzer "Meier" angemeldet.

Hinweis

Die benutzte Variable muss vom Typ "Binär" oder 8-, 16- oder 32-Bit-Wert sein.

8.2.2 Benutzer auf dem Server verwalten

8.2.2.1 Zentrale Benutzerverwaltung mit SIMATIC Logon

Einleitung

Um Benutzer und Benutzergruppen zentral für mehrere Applikationen bzw. Bediengeräte zu verwalten, aktivieren Sie SIMATIC Logon.

Prinzip

SIMATIC Logon ist ein Tool für die anlagenweite Benutzerverwaltung. Bei Verwendung von SIMATIC Logon werden die Benutzer außerhalb der Applikation bzw. des Bediengeräts zentral verwaltet.

Die Benutzergruppen und ihre Berechtigungen projektieren Sie wie von der lokalen Benutzerverwaltung gewohnt in WinCC. Sie benennen die Benutzergruppen auf dem Server und in WinCC identisch. Durch die Namensgleichheit wird in Runtime der Benutzergruppe eine Berechtigungen zugeordnet.

Die Benutzer hingegen müssen Sie nicht in WinCC anlegen, da diese dynamisch beim Anmeldeprozess vom Server übernommen werden. Jede Anmeldung oder Kennwortänderung wird von der Applikation bzw. dem Bediengerät an SIMATIC Logon weitergeleitet und bearbeitet.

Hinweis

SIMATIC Logon ist ein lizenzpflichtiges Produkt. Nähere Informationen zu SIMATIC Logon finden Sie im Internet unter <http://support.automation.siemens.com>.

Geben Sie in das Suchfeld die ID "34519648" ein und starten Sie die Suche. Das Handbuch "SIMATIC Logon - Electronic Signature" steht Ihnen zum Download zur Verfügung.

Anmeldeprozess über SIMATIC Logon Service

Die folgende Abbildung zeigt den Prozess, der automatisch abläuft, wenn ein Benutzer sich in Runtime anmeldet.

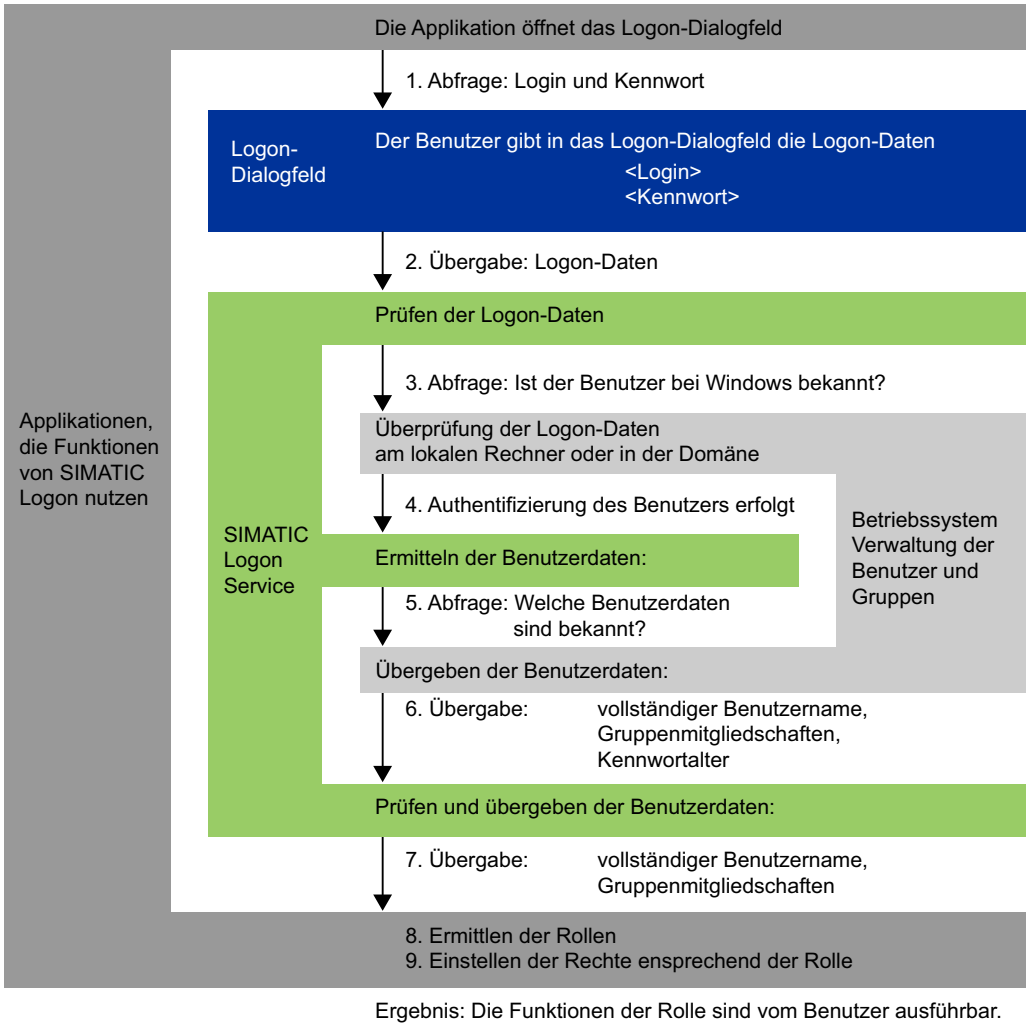

Ergebnis: Die Funktionen der Rolle sind vom Benutzer ausführbar.

Siehe auch

Einstellungen für die Benutzerverwaltung (Seite 1377)

Authentifizierung und Umgang mit Zertifikaten in SIMATIC Logon (Seite 1395)

8.2.2.2 Authentifizierung und Umgang mit Zertifikaten in SIMATIC Logon

Einleitung

SIMATIC Logon (ab V1.5 SP3) unterstützt sichere verschlüsselte Kommunikationsverbindungen zu folgenden Gerätefamilien ab der Geräteversion V13 SP1:

- Comfort Panels
- KTP Mobile Panels
- Runtime Advanced

Hinweis

Bediengeräte mit einer Geräteversion V13 oder älter können nur eine unsichere unverschlüsselte Verbindung zu einem SIMATIC Logon Server herstellen.

Grundlagen

Beim ersten Verbindungsaufbau wird das SIMATIC Logon Zertifikat mit dem lokalen Zertifikat des Bediengeräts abgeglichen. Das abgeglichene Zertifikat wird unter "SimaticLogon\rejected" abgelegt. Wenn Sie dem Zertifikat des SIMATIC Logon Servers vertrauen, kopieren Sie das abgelegte Zertifikat in den lokalen Zertifikatspeicher.

Den Zertifikatspeicher finden Sie unter dem folgenden Pfad:

- am PC normalerweise unter "C:\Programme\Siemens\CoRtHmiRtm\SimaticLogon\certs"
- an HMI Panels unter "\flash\simatic\SimaticLogon\certs"

Wenn der Abgleich erfolgreich durchgeführt ist, kann eine erfolgreiche sichere verschlüsselte Verbindung hergestellt werden.

Weitere Informationen zu getesteten Browsern finden Sie im Internet auf den Webseiten des Customer Support unter der Beitrags-ID=109480490:

- https://support.industry.siemens.com/cs/document/109480490/ (https://support.industry.siemens.com/cs/de/en/view/109480490)

Siehe auch

Zentrale Benutzerverwaltung mit SIMATIC Logon (Seite 1393)

8.2.2.3 Einstellungen auf dem Server vornehmen

Einleitung

Um einen einwandfreien Einsatz von SIMATIC Logon zu gewährleisten, führen Sie folgende Schritte auf Ihrem Server aus.

Vorgehensweise

1. Installieren Sie SIMATIC Logon Services.
2. Legen Sie die Benutzergruppen in der Benutzerverwaltung des Betriebssystems an.
3. Legen Sie die Benutzer in der Benutzerverwaltung des Betriebssystems an.

   Hinweis

   Die Benutzer von SIMATIC Logon müssen direkte Mitglieder einer Benutzergruppe des Betriebssystems sein. Mitglieder einer Untergruppe können nicht angemeldet werden.

4. Ordnen Sie die neuen Benutzer einer Gruppe zu.
5. Übertragen Sie die Lizenzen für jedes Bediengerät auf dem Server.

Hinweis

Informationen für die detaillierte Vorgehensweise erhalten Sie in der mitgelieferten Dokumentation von SIMATIC Logon.

Hinweis

Wenn eine Autorisierung über SIMATIC Logon stattfindet, gelten die Kennwortrichtlinien, die auf dem Server hinterlegt sind.

Hinweis

Die Abmeldezeit für SIMATIC Logon-Benutzer können Sie über "SIMATIC Logon konfigurieren" auf dem Server einstellen.

Automatische Abmeldezeit stellen Sie für alle Benutzer einheitlich ein, unabhängig von den Panels.

8.2.2.4 Anmelden über SIMATIC Logon

Voraussetzungen

- Eine Benutzerverwaltung existiert.
- Die in WinCC angelegten Gruppen stimmen mit den Windows-Gruppen auf dem Server überein.
- Ein Server mit installiertem SIMATIC Logon Service ist vorhanden.

Hinweis

Ein Benutzer darf nur Mitglied einer Benutzergruppe in WinCC sein. Wenn ein Benutzer Mitglied mehrerer Benutzergruppen auf dem Server ist, darf nur eine dieser Benutzergruppen in WinCC bekannt sein.

"Administratoren" und "Benutzer" sind sowohl in Windows als auch in WinCC vordefinierte Benutzergruppen. Wenn Sie in Windows einen neuen Benutzer anlegen, ist dieser automatisch Mitglied der Gruppe "Benutzer". Wenn Sie den Benutzer einer neuen Gruppe hinzufügen z. B. "SL-Benutzer", ist er Mitglied von zwei Gruppen.

Um eine Anmeldung über SIMATIC Logon zu ermöglichen, entfernen Sie den Benutzer aus der Gruppe "Benutzer" des Betriebssystems.

Eine erfolgreiche Anmeldung am Server wird in Runtime immer durch eine Systemmeldung bestätigt.

Hinweis

Wenn Sie SIMATIC Logon verwenden, um die Zugriffe auf ein Panel oder ein Gerät mit WinCC Runtime Advanced zu administrieren, müssen Sie beachten, dass in den Namen von Windows-Benutzergruppen und Windows-Benutzern die Zeichen '/' und '\' nicht verwendet werden dürfen.

Vorgehensweise

1. Öffnen Sie im Projektfenster den Editor "Runtime-Einstellungen > Benutzerverwaltung".
2. Aktivieren Sie "SIMATIC Logon aktivieren".
3. Je nachdem wo Sie Ihre Benutzer verwalten, wählen Sie "Windows-Domäne" oder "Windows-Computer".
4. Tragen Sie im Textfeld "Servername" den Namen oder die IP-Adresse des SIMATIC Logon Servers ein.

   Hinweis

   Achten Sie bei der Verwendung von IP-Adressen darauf, dass Ihr SIMATIC Logon Server immer dieselbe IP-Adresse hat.

5. Tragen Sie im Textfeld "Portnummer" die Portnummer der TCP/IP-Kommunikation ein. Der zulässige Wertebereich liegt zwischen 1024 - 49151. Die voreingestellte Portnummer ist "16389" und wird standardmäßig vom SIMATIC Logon Server verwendet.

6. Tragen Sie im Textfeld "Windows-Domäne" die Windows Domäne oder den Anmeldeserver mit den Benutzerdaten ein.
   Wenn Sie auf Benutzer des Anmeldeservers zugreifen, tragen Sie den Namen des Servers ein.
   Wenn der Anmeldeserver in einer Domäne ist und Sie wollen auf die Benutzer der Domäne zugreifen, tragen Sie den Namen der Domäne ein.

   Hinweis

   Wenn der Anmeldeserver in einer Domäne ist, werden lokale Windows Benutzer nicht akzeptiert.

7. Um die Daten verschlüsselt an den Server zu übertragen, aktivieren Sie das Kontrollkästchen "Verschlüsselte Übertragung".

Notfallbenutzer

Wenn der Server nicht erreichbar ist, können alle lokalen Benutzer, die in der Benutzerverwaltung von WinCC angelegt wurden, auch als Notfallbenutzer verwendet werden.

Notfallbenutzer haben die Rechte der Benutzergruppe, der sie zugewiesen sind.

Hinweis

Weitere Informationen erhalten Sie in der mitgelieferten Dokumentation von SIMATIC Logon.

Zu finden unter: [Installationsverzeichnis SIMATIC Logon] \manuals

8.2.2.5 Anmelden über SIMATIC Logon

Voraussetzungen

- Eine Benutzerverwaltung existiert.
- Die in WinCC angelegten Gruppen stimmen mit den Windows-Gruppen auf dem Server überein.
- Ein Server mit installiertem SIMATIC Logon Service ist vorhanden.

Hinweis

Wenn ein Benutzer Mitglied mehrerer Benutzergruppen ist, werden dem Benutzer die Summe der Berechtigungen der Gruppen zugeordnet. Eine Anmeldung ist möglich.

Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf "Runtime-Einstellungen".
2. Doppelklicken Sie im Arbeitsbereich auf "Benutzerverwaltung".
3. Aktivieren Sie "SIMATIC Logon."

8.2.3 Benutzer für Web Client administrieren

8.2.3.1 Benutzergruppe für Web Client administrieren

Einleitung

Für jede Benutzergruppe, die für die WebClients administriert werden soll, müssen Sie Folgendes konfigurieren:

- Ein beliebiges, für den Web-Zugriff aktiviertes Startbild
- Eine beliebige Sprache
- Anzahl der reservierten Lizenzen (für WebUX)

Sie machen damit den Benutzern verschiedene Bereiche eines Projekts sofort oder aber gar nicht zugänglich.

Hinweis

Der Name der Benutzergruppe muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

Voraussetzungen

- Das Bild "Start" ist angelegt und der Web-Zugriff ist aktiviert.
- Die gewünschte Projektsprache ist aktiviert.
- Die Benutzergruppe "Web-Bediener" ist angelegt.

Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen".
2. Klicken Sie auf "Benutzerverwaltung".
3. In der Registerkarte "Benutzergruppen" wählen Sie die Benutzergruppe "Web-Bediener" aus.
4. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Web-Optionen".
5. Unter "Einstellungen WebNavigator" in der Auswahlliste "Startbild" klicken Sie auf die Schaltfläche ...
   Ein Dialog zur Auswahl des Startbildes wird geöffnet.
6. Wählen Sie das Bild "Start" für Runtime.
7. Schließen Sie den Dialog über das Symbol .
8. Um ein Startbild für WebUX einzustellen, wiederholen Sie die Schritte 5 bis 7.

9. Um eine Sprache für die Benutzergruppe zu konfigurieren, klicken Sie unter "Optionen" in der Auswahlliste "Web-Sprache" auf die Schaltfläche
   Ein Dialog zur Auswahl der Sprache wird geöffnet.
10. Wählen Sie die gewünschte Sprache als Runtime-Sprache der Benutzergruppe.

Alternativ machen Sie diese Einstellungen über das Kontextmenü, indem Sie die Spalten "WebNavigator-Startbild", "WebUX-Startbild" und "Web-Sprache" aktivieren und die entsprechende Konfiguration vornehmen.

Hinweis

Der Name der Benutzergruppe ist sprachabhängig. Sie projektieren die Bezeichnung in mehreren Sprachen und schalten die Sprache in Runtime um.

8.2.3.2 Benutzer für Web Client administrieren

Einleitung

Für jeden Benutzer, der an einem WebClient (WebNavigator oder WebUX) angemeldet werden soll, müssen Sie Folgendes konfigurieren:

- Ein beliebiges, für den Web-Zugriff aktiviertes Startbild
- Eine beliebige Sprache
- Anzahl der reservierten Lizenzen (für WebUX)

Sie machen damit den Benutzern verschiedene Bereiche eines Projekts sofort oder aber gar nicht zugänglich.

Hinweis

Neue Benutzer, die über einen Web-Client oder einen WebUX-Client auf das System zugreifen sollen, können nur im Engineering System angelegt werden.

Wenn Sie einen Benutzer in Runtime mit der Benutzeranzeige anlegen, ist es nicht möglich, diesem Benutzer ein Web-Startbild zuzuweisen. Somit ist dieser Benutzer nicht für Web-Zugriffe freigeschaltet.

Bestehenden Benutzern können in Runtime ebenfalls keine Web-Zugriffsrechte erteilt werden.

Voraussetzungen

- Das Bild "Start" ist angelegt und der Web-Zugriff ist aktiviert.
- Die gewünschte Projektsprache ist aktiviert.

Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen".
2. Klicken Sie auf "Benutzerverwaltung".

3. In der Registerkarte "Benutzer" wählen Sie einen Benutzer aus, für den Sie die Einstellungen vornehmen möchten.
4. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Web-Optionen".
5. Unter "Einstellungen WebNavigator" in der Auswahlliste "Startbild" klicken Sie auf die Schaltfläche ...
   Ein Dialog zur Auswahl des Startbildes wird geöffnet.
6. Wählen Sie das Bild "Start".
7. Schließen Sie den Dialog über das Symbol
8. Um ein Startbild für WebUX einzustellen, wiederholen Sie die Schritte 5 bis 7.
9. Unter "Reservierte Lizenzanzahl" konfigurieren Sie die Anzahl der reservierten WebUX-Lizenzen für den ausgewählten Benutzer mithilfe der Schaltflächen

   Hinweis

   Wenn mehr reservierte Lizenzen konfiguriert werden als Lizenzen auf dem WebUX-Server verfügbar sind, werden nur die ersten unter der Benutzerverwaltung aufgeführten Benutzer berücksichtigt.

   Hinweis

   Wenn die Option "WebNavigator-Lizenzen für WebUX verwenden" aktiviert ist, können keine reservierten WebUX-Lizenzen garantiert werden.

10. Um eine Sprache für den Benutzer zu konfigurieren, klicken Sie unter "Optionen" in der Auswahlliste "Web-Sprache" auf die Schaltfläche .
    Ein Dialog zur Auswahl der Sprache wird geöffnet.
11. Wählen Sie die gewünschte Sprache als Runtime-Sprache des Web-Benutzers.

Alternativ machen Sie diese Einstellungen über das Kontextmenü, indem Sie die Spalten "WebNavigator-Startbild", "WebUX-Startbild", "Web-Sprache" und "Für WebUX reservierte Lizenzen" aktivieren und die entsprechende Konfiguration vornehmen.

Hinweis

Wenn der Benutzer die Gruppe wechselt, bleibt die Spracheinstellung erhalten. Das Startbild wird allerdings überschrieben.

Hinweis

Standardmäßig werden die WebNavigator-Lizenzen für WebUX genutzt. Wenn Sie diese Lizenzen jeweils separat nutzen möchten, deaktivieren Sie unter "Runtime-Einstellungen > Web-Zugriff" das Kontrollkästchen "WebNavigator-Lizenzen für WebUX verwenden".

8.2.4 Benutzer in Runtime administrieren

8.2.4.1 Benutzer in Runtime

Prinzip

Sie legen Benutzer und Benutzergruppen an und weisen ihnen Berechtigungen zu. Sie projektieren Objekte mit Berechtigungen. Nach dem Laden auf das Bediengerät sind in Runtime alle Objekte vor unberechtigtem Zugriff geschützt, die mit einer Berechtigung projektiert wurden.

Benutzeranzeige

Wenn Sie im Engineering System eine Benutzeranzeige projektieren, administrieren Sie nach dem Laden auf dem Bediengerät in der Benutzeranzeige die Benutzer.

ACHTUNG
Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert. Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige auf Nachfrage und in Abhängigkeit von den Einstellungen beim Laden überschrieben.

Benutzer, welche die Berechtigung "Benutzerverwaltung" haben, besitzen uneingeschränkten Zugriff auf die Benutzeranzeige. Damit administrieren sie alle Benutzer. Um sich selbst zu administrieren, hat jeder andere Benutzer nur einen eingeschränkten Zugriff auf die Benutzeranzeige.

Siehe auch

Benutzeranzeige (Seite 1402)

Benutzer mit der komplexen Benutzeranzeige verwalten (Seite 1406)

Sich als Benutzer anmelden (Seite 1410)

8.2.4.2 Benutzeranzeige

Zweck

Um auch in Runtime die Benutzer zu administrieren, projektieren Sie eine Benutzeranzeige im Engineering System.

Aufbau

Die Benutzeranzeige zeigt in jeder Zeile:

- Den Benutzer,
- Sein verschlüsseltes Kennwort,
- Die zugehörige Benutzergruppe,
- Die Abmeldezeit.

Wenn kein Benutzer angemeldet ist, ist die Benutzeranzeige leer. Nach dem Anmelden werden die Inhalte der einzelnen Felder angezeigt.

Benutzeranzeige eines Administrators

Benutzer	Kennwort	Gruppe	Abmeldezeit
Administrator	********	Administratorengruppe	5
Johnson	********	Benutzer	5
Meister	********	Group 2	5
PLC User	********	Unberechtigt	5

Wenn ein Administrator angemeldet ist, zeigt die Benutzeranzeige alle Benutzer. Der Administrator ändert den Benutzernamen und das Kennwort. Der Administrator legt neue Benutzer an und ordnet sie einer bestehenden Benutzergruppe zu.

Benutzeranzeige eines Benutzers

Benutzer	Kennwort	Gruppe	Abmeldezeit
Johnson	********	Benutzer	5

Wenn kein Administrator angemeldet ist, zeigt die Benutzeranzeige nur den angemeldeten Benutzer. Der Benutzer kann sein Kennwort und seine Abmeldezeit ändern.

Siehe auch

Benutzer in Runtime (Seite 1402)

Sich als Benutzer anmelden (Seite 1410)

8.2.4.3 Benutzeranzeige projektieren

Einleitung

Um die Benutzer auch im Runtime zu administrieren, projektieren Sie eine Benutzeranzeige im Engineering System.

Voraussetzungen

Ein Bild ist erstellt.

Vorgehensweise

1. Markieren Sie im Werkzeugfenster in der Kategorie "Controls" das Objekt "Benutzeranzeige"
2. Ziehen Sie per Drag&Drop das Objekt "Benutzeranzeige" in das Bild.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften".
4. Legen Sie das Erscheinungsbild der "Benutzeranzeige" fest.
5. Wählen Sie z. B. "Darstellung > Größenanpassung > Objekt an Inhalt anpassen".

Ergebnis

Sie haben eine Benutzeranzeige im Bild angelegt.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.4.4 Benutzer anlegen

Einleitung

Sie legen einen Benutzer an, damit sich Personen unter dem Benutzernamen in Runtime anmelden.

Alternativ können Sie Benutzer im Engineering System anlegen und auf das Bediengerät laden.

Hinweis

Gilt nur für Runtime Professional:

Neue Benutzer, die über einen Web-Client oder einen WebUX-Client auf das System zugreifen sollen, können nur im Engineering System angelegt werden.

Wenn Sie einen Benutzer in Runtime mit der Benutzeranzeige anlegen, ist es nicht möglich, diesem Benutzer ein Web-Startbild zuzuweisen. Somit ist dieser Benutzer nicht für Web-Zugriffe freigeschaltet.

Bestehenden Benutzern können in Runtime ebenfalls keine Web-Zugriffsrechte erteilt werden.

Nur wenn der bei der Anmeldung eingegebene Benutzername mit einem Benutzer in Runtime übereinstimmt, ist die Anmeldung erfolgreich. Darüber hinaus muss das bei der Anmeldung eingegebene Kennwort bei dem Benutzer übereinstimmen.

Hinweis

Beachten Sie die Groß- und Kleinschreibung.

Hinweis

Verwenden Sie bei der Eingabe eines Benutzernamen und eines Kennworts keine Sonderzeichen wie / " § $ % ? ' &.

Sie ordnen den Benutzer einer Benutzergruppe zu. Der Benutzer besitzt dann die Berechtigungen der Benutzergruppe.

Hinweis

In Runtime müssen Sie einen Benutzer einer Benutzergruppe zuordnen. Die Benutzergruppe wird im Engineering System angelegt. Die Bezeichnung der Benutzergruppe ist sprachabhängig.

Voraussetzungen

- Die Benutzeranzeige ist geöffnet.
- Eine Benutzergruppe "Group 2" ist angelegt.

Vorgehensweise

1. Klicken Sie in der Benutzeranzeige <Neuer Benutzer>. Ein Dialog wird geöffnet.
2. Geben Sie als Benutzernamen "Meister" ein.
3. Betätigen Sie die Taste <Return>.

4. Klicken Sie auf "Kennwort".
5. Geben Sie das Kennwort des Benutzers ein.
6. Betätigen Sie die Taste <Return>. Das Kennwort ist unsichtbar.
7. Klicken Sie in die Spalte "Gruppe".
8. Wählen Sie als "Gruppe" "Group 2".

Benutzer	Kennwort	Gruppe	Abmeldezeit
Administrator	********	Administra...	5
Johnson	********	Administra...	5
Meister	********	Group 2	5
PLC User	********	Unberechtigt	5

9. Betätigen Sie die Taste <Return>.
10. Klicken Sie in die Spalte "Abmeldezeit".
11. Geben Sie die Zeit ein, nachdem der Benutzer automatisch abgemeldet wird.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.4.5 Benutzer mit der komplexen Benutzeranzeige verwalten

Einleitung

Wenn Sie im Engineering System eine Benutzeranzeige projektieren haben, lassen sich die Benutzer und Benutzergruppen in Runtime verwalten.

ACHTUNG
Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert. Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige auf Nachfrage und in Abhängigkeit von den Einstellungen überschrieben.

Voraussetzungen

- Runtime ist aktiviert.
- Eine komplexe Benutzeranzeige ist angelegt.

- Das Bild mit der Benutzeranzeige ist geöffnet.
- Sie besitzen die vordefinierte Berechtigung "Benutzerverwaltung".

Hinweis

Wenn Sie die Berechtigung "Benutzerverwaltung" nicht besitzen, haben Sie nur das Recht Ihr Kennwort und Ihre Abmeldezeit zu ändern.

Benutzername ändern

1. Geben Sie in der Benutzeranzeige in der Spalte "Benutzer" einen neuen Benutzernamen ein.
2. Bestätigen Sie die Eingabe mit der Taste <Return>.

Hinweis

Der Benutzer kann sich nicht mehr mit seinem bisherigen Benutzernamen in Runtime anmelden. Wenn Sie den Namen löschen und die Taste <Return> betätigen, wird der Benutzer gelöscht.

Kennwort des Benutzers ändern

Die Verfügbarkeit der komplexen Benutzeranzeige ist geräteabhängig.

1. Geben Sie in der Benutzeranzeige in der Spalte "Kennwort" ein neues Kennwort ein.
2. Bestätigen Sie die Eingabe mit der Taste <Return>.

Hinweis

Der Benutzer kann sich nicht mehr mit seinem bisherigen Kennwort in Runtime anmelden.

Wenn Sie in der komplexen Benutzeranzeige das Kennwort löschen und mit <Return> betätigen, wird der Benutzer gelöscht.

Abmeldezeit des Benutzers ändern

1. Geben Sie in der Benutzeranzeige in der Spalte "Abmeldezeit" eine neue Abmeldezeit ein.
2. Bestätigen Sie die Eingabe mit der Taste <Return>.

Benutzer löschen

1. Klicken Sie auf den Namen des zu löschenden Benutzers.
2. Löschen Sie den Namen.
3. Betätigen Sie die Taste <Return>.

Hinweis

Der Benutzer kann sich nicht mehr in Runtime anmelden.

Benutzer einer anderen Benutzergruppe zuordnen

1. Aktivieren Sie für den entsprechenden Benutzer das Feld Benutzergruppe.
2. Wählen Sie eine Benutzergruppe aus.
3. Bestätigen Sie die Auswahl mit der Taste <Return>.

Siehe auch

Benutzer in Runtime (Seite 1402)

Sich als Benutzer anmelden (Seite 1410)

8.2.4.6 Gesperrte Benutzer entsperren

Gesperrte Benutzer entsperren

In den "Runtime-Einstellungen > Benutzerverwaltung" ist das Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert.
Im Feld "Anzahl der ungültigen Login-Versuche" ist die Zahl 3 eingetragen.

Wenn ein Benutzer bei der Anmeldung drei Fehlversuche erzeugt, z. B. durch eine falsche Kennworteingabe, wird er der Gruppe "Unberechtigt" zugewiesen. Der Benutzer verliert alle Berechtigungen. Der Benutzer kann sich zwar noch anmelden, hat aber keinerlei Berechtigungen mehr. Nur ein Benutzer mit Administratorrechten weist den unberechtigten Benutzer erneut einer Benutzergruppe zu.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.4.7 Sich als Benutzer anmelden

Einleitung

Sie melden sich als Benutzer in der Regel über eine spezielle Schaltfläche an. Dazu erscheint der Anmeldedialog.

Der Anmeldedialog erscheint standardmäßig beim Zugriff auf ein geschütztes Objekt, wenn

- Kein Benutzer in Runtime angemeldet ist
- Der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt.

Hinweis

Auf Basic Panels wird beim Bedienen einer Schaltfläche mit Zugriffsschutz immer der Anmeldedialog aufgeblendet.

Voraussetzungen

- Unter "Runtime-Einstellungen > Benutzerverwaltung" ist
  - das Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert.
  - im Feld "Anzahl der ungültigen Login-Versuche" die Zahl 3 eingetragen.
- Die Systemfunktion "ZeigeAnmeldedialog" ist an eine Schaltfläche mit der Beschriftung "Logon" projektiert.

Vorgehensweise

1. Klicken Sie auf die Schaltfläche "Logon". Der Anmeldedialog erscheint.

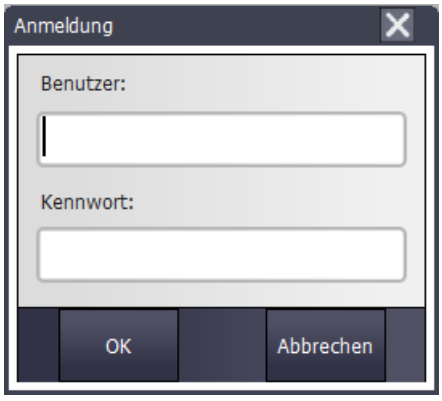

2. Geben Sie Ihren Benutzernamen ein, wie in der Benutzerverwaltung vergeben, z. B. "Meister".
   Wenn sich vor Ihnen ein Benutzer angemeldet hat, wird dessen Benutzername angezeigt.
3. Geben Sie das zugehörige Kennwort ein. Die Eingabe erfolgt verdeckt.

   Hinweis

   Beachten Sie beim Eingeben des Kennworts die Groß- und Kleinschreibung.

4. Schließen Sie den Dialog über die Schaltfläche "OK".

Hinweis

Wenn Sie das Bediengerät wechseln, müssen Sie das Kennwort nicht erneut eingeben, wenn auf allen Bediengeräten die erlaubten Sonderzeichen im Kennwort enthalten sind.

Folgende Sonderzeichen sind für Kennwörter zugelassen:

()€&@$%+#^[]|*-"<>!,.;:/_=\'? `~{}$

Wenn Sie von einer älteren Geräteversion zu V14 wechseln und für das Kennwort nicht zugelassene Sonderzeichen verwendet wurden, wird das Kennwort gelöscht. Sie werden durch eine Fehlermeldung darüber informiert und müssen ein neues Kennwort hinterlegen.

Anmeldung war erfolgreich

Wenn Sie als Benutzernamen "Meister" eingegeben haben und das eingegebene Kennwort mit dem hinterlegten Kennwort übereinstimmt, dann sind Sie in Runtime als Benutzer "Meister" angemeldet. Sie besitzen die Berechtigungen des Benutzers "Meister".

Nur wenn der Benutzer "Meister" beim Zugriff auf ein geschütztes Objekt, z. B. die Schaltfläche "Archivierung", die erforderliche Berechtigung besitzt, wird der Zugriff frei gegeben. Die projektierte Funktion wird sofort ausgeführt.

Wenn Sie nach der erfolgreichen Anmeldung nicht die erforderliche Berechtigung besitzen, erscheint eine entsprechende Fehlermeldung. Sie bleiben aber in Runtime angemeldet.

Anmeldung war nicht erfolgreich

Eine Fehlermeldung erscheint.

Aus Sicherheitsgründen besitzen Sie oder der zuvor angemeldete Benutzer keine Berechtigungen mehr. Der Zugriff auf nicht geschützte Objekte ist jedoch möglich. Die Benutzeranzeige zeigt jedoch keine Einträge. Benutzeranzeige und Berechtigungen ändern sich bei der nächsten erfolgreichen Anmeldung.

Wenn der dritte Anmeldeversuch fehlgeschlagen ist, wird der Benutzer der vordefinierten Gruppe "Unberechtigt" zugeordnet. Projektieren Sie daher keine Benutzergruppen mit diesem Anzeigenamen.

Wenn die Funktion "Abmelden" aufgerufen wird oder die Abmeldezeit des Benutzers abgelaufen ist, dann wird der Benutzer abgemeldet.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.4.8 Sich als Benutzer anmelden

Einleitung

Sie melden sich als Benutzer in der Regel über eine spezielle Schaltfläche an. Dazu erscheint der Anmeldedialog.

Der Anmeldedialog erscheint standardmäßig beim Zugriff auf ein geschütztes Objekt, wenn

- Kein Benutzer in Runtime angemeldet ist
- Der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt.

Voraussetzungen

- Die Systemfunktion "ZeigeAnmeldedialog" ist an eine Schaltfläche mit der Beschriftung "Logon" projektiert.

Vorgehensweise

1. Klicken Sie auf die Schaltfläche "Logon". Der Anmeldedialog erscheint.

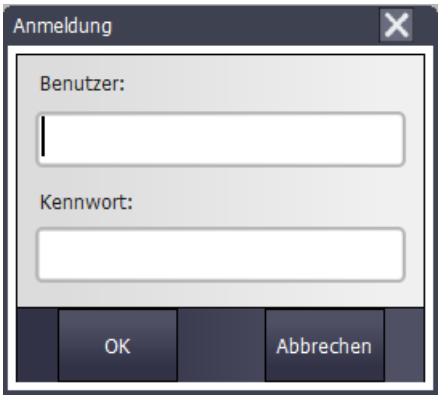

2. Geben Sie Ihren Benutzernamen ein, wie in der Benutzerverwaltung vergeben, z. B. "Meister".
   Wenn sich vor Ihnen ein Benutzer angemeldet hat, wird dessen Benutzername angezeigt.
3. Geben Sie das zugehörige Kennwort ein. Die Eingabe erfolgt verdeckt.
4. Schließen Sie den Dialog über die Schaltfläche "OK".

Anmeldung war erfolgreich

Wenn Sie als Benutzernamen "Meister" eingegeben haben und das eingegebene Kennwort mit dem hinterlegten Kennwort übereinstimmt, dann sind Sie in Runtime als Benutzer "Meister" angemeldet. Sie besitzen die Berechtigungen des Benutzers "Meister".

Nur wenn der Benutzer "Meister" beim Zugriff auf ein geschütztes Objekt, z. B. die Schaltfläche "Archivierung", die erforderliche Berechtigung besitzt, wird der Zugriff frei gegeben. Die projektierte Funktion wird sofort ausgeführt.

Wenn Sie nach der erfolgreichen Anmeldung nicht die erforderliche Berechtigung besitzen, erscheint eine entsprechende Fehlermeldung. Sie bleiben aber in Runtime angemeldet.

Anmeldung war nicht erfolgreich

Eine Fehlermeldung erscheint.

Aus Sicherheitsgründen besitzen Sie oder der zuvor angemeldete Benutzer keine Berechtigungen mehr. Der Zugriff auf nicht geschützte Objekte ist jedoch möglich. Die Benutzeranzeige zeigt jedoch keine Einträge. Benutzeranzeige und Berechtigungen ändern sich bei der nächsten erfolgreichen Anmeldung.

Wenn die Funktion "Abmelden" aufgerufen wird oder die Abmeldezeit des Benutzers abgelaufen ist, dann wird der Benutzer abgemeldet.

Siehe auch

Benutzer in Runtime (Seite 1402)

Benutzeranzeige (Seite 1402)

Benutzeranzeige projektieren (Seite 1404)
Benutzer anlegen (Seite 1404)
Benutzer mit der komplexen Benutzeranzeige verwalten (Seite 1406)
Gesperrte Benutzer entsperren (Seite 1408)
Sich als Benutzer anmelden (Seite 1408)
Benutzerverwaltung exportieren (Seite 1412)
Benutzerverwaltung importieren (Seite 1413)

8.2.4.9 Benutzerverwaltung exportieren

Einleitung

Die Benutzer und Benutzergruppen werden vom Engineering System auf das Bediengerät transferiert. Wenn Sie eine Benutzeranzeige projektiert haben, können Sie am Bediengerät die Benutzer über die Benutzeranzeige verwalten.

Wenn ein Benutzer auf mehrere Bediengeräte zugreift, müssen an jedem Bediengerät derselbe Benutzer und dasselbe Kennwort vorhanden sein. Bringen Sie die Benutzerverwaltung der unterschiedlichen Bediengeräte auf einen einheitlichen Stand. Exportieren Sie die an einem Bediengerät vorhandenen Benutzer und Kennwörter auf ein Speichermedium, z. B. eine Diskette, eine Speicherkarte oder ein Netzlaufwerk. An allen anderen Bediengeräten importieren Sie die Benutzer und Kennwörter von diesem Speichermedium.

Sie projektieren an eine Schaltfläche eine Funktion, welche die Benutzerdaten exportiert.

Die Systemfunktion "ExportiereImportiereBenutzerverwaltung" steht nicht auf allen Bediengeräten zur Verfügung.

ACHTUNG
Beim Export werden die Benutzerdaten vom Bediengerät verschlüsselt in die Zieldatei geschrieben. Die Zieldatei wird überschieben

Voraussetzungen

- Ein Bediengerät ist angelegt.
- Ein Bild ist angelegt.
- Eine Schaltfläche ist im Bild angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Klicken".

3. Klicken Sie in der Funktionsliste auf den Eintrag "Funktion hinzufügen".
4. Wählen Sie als Systemfunktion "ExportiereImportiereBenutzerverwaltung".
5. Wählen Sie als "Richtung" "Export". Beim Export wird die Zieldatei mit den Benutzerdaten überschrieben.
6. Geben Sie als "Dateiname" den vollständigen Pfad der Zieldatei ein z. B. "a:\test \usersview.txt".

Hinweis

Achten Sie auf die korrekte Schreibweise des Dateinamens. Wenn der Ordner, z. B. "\test \" nicht existiert, wird der Ordner ohne Rückfrage angelegt.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.4.10 Benutzerverwaltung importieren

Einleitung

Wenn ein Benutzer auf mehrere Bediengeräte zugreift, müssen an jedem Bediengerät derselbe Benutzer und dasselbe Kennwort vorhanden sein. Bringen Sie die Benutzerverwaltung der unterschiedlichen Bediengeräte auf einen einheitlichen Stand. Exportieren Sie die an einem Bediengerät vorhandenen Benutzer und Kennwörter auf ein Speichermedium, z. B. eine Diskette, eine Speicherkarte oder ein Netzlaufwerk. An allen anderen Bediengeräten importieren Sie die Benutzer und Kennwörter von diesem Speichermedium.

Sie projektieren an eine Schaltfläche eine Funktion, welche die Benutzerdaten importiert.

Die Systemfunktion "ExportiereImportiereBenutzerverwaltung" steht nicht auf allen Bediengeräten zur Verfügung.

ACHTUNG
Beim Import werden die Benutzerdaten von der Quelldatei eingelesen und in das Bediengerät geschrieben. Die aktuell gültigen Benutzer und Kennwörter im Bediengerät werden überschrieben. Die importierten Benutzer und Kennwörter sind sofort gültig.

Voraussetzungen

- Ein Bild ist erstellt.
- Eine Schaltfläche ist im Bild erstellt.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Klicken".
3. Klicken Sie in der Funktionsliste auf den Eintrag "Funktion hinzufügen".
4. Wählen Sie als Systemfunktion "ExportiereImportiereBenutzerverwaltung".
5. Wählen Sie als "Richtung" "Import". Beim Import werden die Benutzerdaten im Bediengerät überschrieben.
6. Geben Sie als "Dateiname" den vollständigen Pfad der Quelldatei an z. B. "a:\test \usersview.txt".

Hinweis

Achten Sie auf die korrekte Schreibweise des Dateinamens. Wenn der Ordner, z. B. "\test \" oder die Datei nicht existiert, erscheint eine Fehlermeldung.

Siehe auch

Sich als Benutzer anmelden (Seite 1410)

8.2.5 Zugriffsschutz projektieren

8.2.5.1 Zugriffsschutz

Einleitung

Um ein Objekt vor Zugriff zu schützen, projektieren Sie an das Objekt eine Berechtigung. Alle angemeldeten Benutzer, die diese Berechtigung besitzen, greifen dann auf das Objekt zu. Wenn ein Benutzer nicht die Bedienberechtigung eines Objekts besitzt, wird automatisch der Anmeldedialog aufgeblendet.

Hinweis

Zur Verarbeitung von Benutzer, Kennwort und Benutzergruppe, z. B. in der Steuerung, stehen Ihnen unter "Benutzerverwaltung" mehrere Systemfunktionen zur Verfügung.

8.2.5.2 Bedienberechtigung projektieren

Einleitung

Sie projektieren die Berechtigung "Runtime beenden" an eine Schaltfläche. Damit haben nur Benutzer darauf Zugriff, die diese Berechtigung besitzen, z. B. alle Benutzer der Benutzergruppe "Bediener".

Dadurch ist der Zugriff auf die Schaltfläche geschützt. Wenn ein angemeldeter Benutzer auf die Schaltfläche klickt, welcher der Benutzergruppe "Bediener" angehört und die erforderliche Berechtigung besitzt, wird Runtime beendet.

Wie Sie eine Schaltfläche mit Zugriffsschutz projektieren, ist in einem Beispiel ausführlich beschrieben.

Voraussetzungen

- Die Benutzergruppe "Bediener" ist angelegt.
- Die Berechtigung "Runtime beenden" ist erstellt.
- Ein Bild ist erstellt und geöffnet.
- Das Bild enthält eine Schaltfläche.

Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Security".
3. Wählen Sie als "Berechtigung" Runtime beenden".
4. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken"
5. Wählen Sie eine Systemfunktion aus der Funktionsliste, z.B. "StoppeRuntime".

Hinweis

Die Ereignisse "Aktivieren" und "Deaktivieren" dienen ausschließlich zur Erkennung, ob ein Objekt ausgewählt bzw. abgewählt wurde. Die Ereignisse lösen aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Aktivieren" bzw. "Deaktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

8.3 Referenz

8.3.1 Objekte mit Zugriffsschutz

Einleitung

Folgende Objekte sind mit einer Berechtigung projektierbar:

- E/A-Feld
- Schaltfläche
- Symbolisches E/A-Feld
- Grafisches E/A-Feld
- Datum/Uhrzeit-Feld
- Schalter
- Meldeanzeige
- Benutzeranzeige
- Rezepturanzeige
- System-Diagnoseanzeige

8.3.2 Objekte mit Zugriffsschutz

Einleitung

Folgende Objekte sind mit einer Berechtigung projektierbar:

- E/A-Feld
- Schaltfläche
- Symbolisches E/A-Feld
- Grafisches E/A-Feld
- Datum/Uhrzeit-Feld
- Schalter
- Symbolbibliothek
- Schieberegler
- Wirkbereichs-Name
- Wirkbereichs-Name (RFID)
- Meldeanzeige
- Meldefenster

- Benutzeranzeige
- HTML-Browser
- Beobachtungstabelle
- Sm@rtClient-Anzeige
- Rezepturanzeige
- f(x)-Kurvenanzeige
- System-Diagnoseanzeige
- System-Diagnosefenster
- PLC-Code-Anzeige
- GRAPH-Übersicht
- ProDiag-Übersicht

8.3.3 Objekte mit Zugriffsschutz

Einleitung

Folgende Objekte sind mit einer Berechtigung projektierbar:

- Verbinder
- Rohr
- Doppelt-T-Stück
- T-Stück
- Rohrbogen
- E/A-Feld
- Editierbares Textfeld
- Listenfeld
- Kombinationsfeld
- Schaltfläche
- Rundschaltfläche
- Symbolisches E/A-Feld
- Grafisches E/A-Feld
- Balken
- Schieberegler
- Bildlaufleiste
- Kontrollkästchen
- Optionsschaltfläche

- Zeigerinstrument
- Uhr
- Benutzeranzeige
- HTML-Browser
- Rezepturanzeige
- Meldeanzeige
- f(x)-Kurvenanzeige
- f(t)-Kurvenanzeige
- Tabellenanzeige
- Wertertabelle
- System-Diagnoseanzeige
- Media Player
- Kanal-Daignoseanzeige
- PLC-Code-Anzeige
- GRAPH-Übersicht
- ProDiag-Übersicht

Um folgende Objekte mit einem Zugriffsschutz zu projektieren, binden Sie zusätzlich die Systemfunktion "ZeigeAnmeldedialog" an ein Ereignis z. B. Klicken.

- Linie
- Polygonzug
- Polygon
- Ellipse
- Ellipsensegment
- Kreissegment
- Ellipsenbogen
- Kreisbogen
- Kreis
- Rechteck
- Textfeld
- Grafikanzeige
- Bildfenster
- Druckauftrag/Skriptdiagnose

8.3.4 Vordefinierte Benutzergruppen und Berechtigungen

Prinzip

Die vordefinierten Benutzergruppen und Berechtigungen haben folgende Nummern:

Benutzergruppe	Nummer
"Administratorengruppe"	1
"Benutzer"	2

Berechtigung	Nummer
"Benutzerverwaltung"	1
"Überwachen"	2
"Bedienen"	3

8.3.5 Vordefinierte Benutzergruppen und Berechtigungen

Prinzip

Die vordefinierten Benutzergruppen und Berechtigungen haben folgende Nummern:

Benutzergruppe:

- "Administratorengruppe"
- "Benutzer"

Berechtigung	Nummer
"Benutzerverwaltung"	1
"Überwachen"	2
"Bedienen"	3
"Fernbedienung aktivieren"	1000
"Web Zugriff - Nur beobachten"	1002

Die Berechtigungen 1000 "Fernbedienung aktivieren" ist für Client-Server-Projekte von Nutzen.

Die Berechtigung 1002 "Web Zugriff - Nur beobachten" ist in Verbindung mit dem Web Navigator von Bedeutung.

8.4 Beispiele

8.4.1 Beispiel: Schaltfläche mit Anmeldedialog projektieren

Aufgabe

In folgendem Beispiel projektieren Sie die Funktion "ZeigeAnmeldedialog" an eine Schaltfläche. Z. B. beim Schichtwechsel meldet sich ein anderer Benutzer in Runtime an. Dabei wird der zuvor angemeldete Benutzer abgemeldet.

Hinweis

In Runtime erscheint standardmäßig erst dann der Anmeldedialog, wenn Sie auf ein geschütztes Objekt zugreifen. Entweder ist kein Benutzer angemeldet oder der angemeldete Benutzer besitzt nicht die erforderliche Berechtigung.

Voraussetzungen

- Ein Bild ist erstellt.
- Eine Schaltfläche ist im Bild neu erstellt.

Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Loslassen".
3. Klicken Sie in der Tabelle "Funktionsliste" auf den Eintrag "Funktion hinzufügen".
4. Wählen Sie in der Gruppe "Benutzerverwaltung" die Systemfunktion "ZeigeAnmeldedialog".

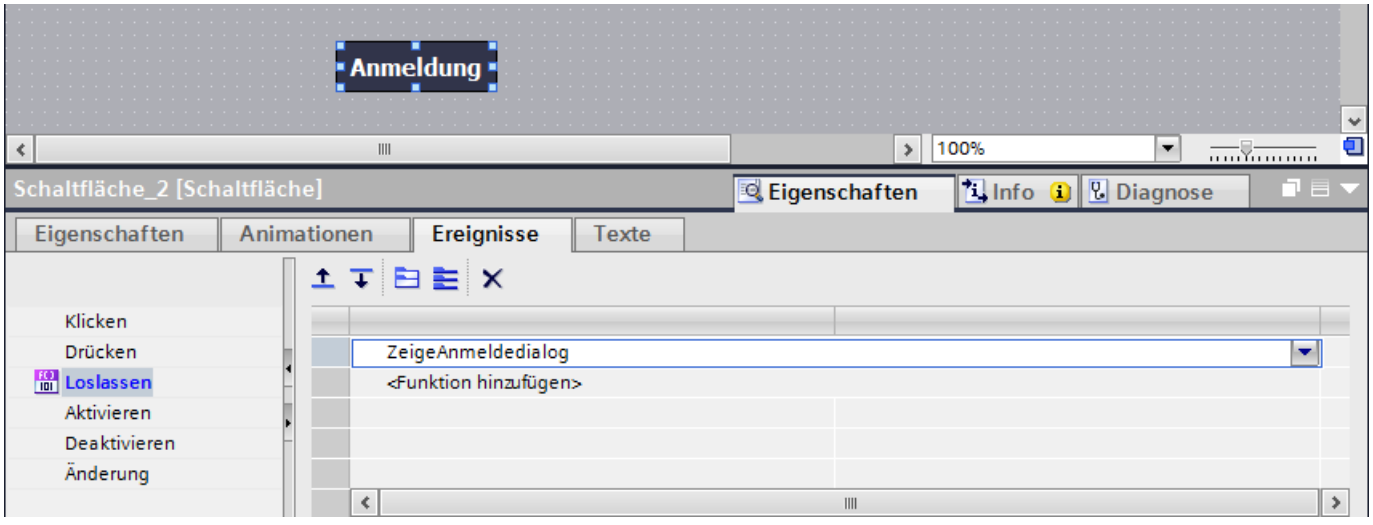

Ergebnis

Wenn der Benutzer in Runtime auf die Schaltfläche klickt, wird die Funktion "ZeigeAnmeldedialog" aufgerufen. Beim Aufruf der Funktion "ZeigeAnmeldedialog" erscheint der Anmeldedialog. Der Benutzer meldet sich mit dem Benutzernamen und seinem Kennwort an.

8.4.2 Beispiel: Anmelden und Abmelden protokollieren

Aufgabe

In folgendem Beispiel projektieren Sie die Funktion "VerfolgeBenutzeränderung" an das Ereignis "Benutzerwechsel".

Prinzip

Wenn sich ein Benutzer anmeldet oder abmeldet, wird die Funktion "VerfolgeBenutzeränderung" aufgerufen. Wenn eine Funktion aufgerufen wird, wird eine Systemmeldung mit Angabe des betreffenden Benutzers ausgegeben.

Diese Systemmeldung können Sie archivieren. Beim Archivieren wird die Systemmeldung mit einem Datumsstempel und Zeitstempel versehen. Dadurch können Sie zurückverfolgen, welcher Benutzer zu welcher Zeit und wie lange am Bediengerät angemeldet war.

Voraussetzungen

- Ein Bediengerät mit Runtime Advanced ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf "Aufgabenplaner".
2. Doppelklicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen".
3. Geben Sie als "Name" "Logon-Protocol" ein.
4. Wählen Sie als "Trigger" "Benutzerwechsel".
5. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".
6. Klicken Sie in der Tabelle "Funktionsliste" auf den Eintrag "Funktion hinzufügen".
7. Wählen Sie die Systemfunktion "VerfolgeBenutzeränderung".

Ergebnis

Wenn sich ein Benutzer anmeldet oder abmeldet, wird eine Systemmeldung ausgegeben.

8.4.3 Beispiel: Aufbau einer Benutzerverwaltung

Aufgabe

Im folgenden Beispiel richten Sie eine Benutzerverwaltung für unterschiedliche Benutzer und Benutzergruppen ein. Das Beispiel orientiert sich an einem typischen Anforderungsprofil aus der Fertigungstechnik.

Prinzip

Bei einer Anlage, einem Projekt sind ganz unterschiedliche Personengruppen beteiligt. Jede Personengruppe schützt jeweils ihre Daten und Funktionen vor dem Zugriff anderer. Dazu werden Benutzer angelegt und einer Benutzergruppe zugeordnet.

Sie können unterschiedliche Sichten durch Benutzergruppen abbilden.

Beispiel:

- Organisatorische Sicht: Inbetriebsetzer, Bediener, Schicht I, Schicht II
- Technologische Sicht: Achsensteuerung, Werkzeugwechsler, Anlage Nord, Anlage Süd

Das nachfolgende Beispiel orientiert sich an der organisatorischen Sicht.

Jede Benutzergruppe hat charakteristische Anforderungen an den Zugriffsschutz: Eine Benutzergruppe besitzt Berechtigungen für bestimmte Anwendungsfälle. Ein Programmierer ändert z. B. Rezepturdatensätze.

Im Beispiel werden die Benutzer Müller, Gruppe Meier und Meister angelegt und unterschiedlichen Benutzergruppen zugeordnet.

Frau Müller arbeitet als Programmierer mit dem Engineering System. Bei der Gruppe Meier handelt es sich um Inbetriebsetzer. Herr Meister ist Bediener.

Voraussetzungen

- Ein neues Projekt ist angelegt.
- Der Editor "Benutzerverwaltung" ist geöffnet.

Handlungsübersicht

Das Arbeiten mit der Benutzerverwaltung hat im Beispiel folgenden Ablauf:

1. Berechtigungen erstellen: Der Projekteur legt fest, welche Berechtigungen für den Zugriffsschutz benötigt werden.
2. Berechtigungen projektieren: Der Projekteur legt fest, welche Objekte bedient und welche Funktionen ausgeführt werden dürfen.

3. Benutzergruppen anlegen und Berechtigungen zuweisen: Der Administrator legt zusammen mit dem Projekteur die Benutzergruppen an. Der Projekteur legt über die Berechtigungen fest, wer Objekte bedienen und Änderungen an Parametern durchführen darf.
4. Benutzer anlegen und einer Benutzergruppe zuordnen: Der Administrator verwaltet die Benutzer.

Ergebnis

Das Ziel ist folgender Aufbau der Benutzerverwaltung aus Benutzern, Benutzergruppen und Berechtigungen:

Benutzer			Benutzergruppen	Berechtigungen			
Müller	Meier	Meister	Rollen	Rezepturdatensätze ändern	System-parameter ändern	Prozess-parameter ändern	Verwalten
			Administratorengruppe				x
X			Programmierer	X			
	X		Inbetriebsetzer	X	X	X	
		X	Bediener	x			

Der Benutzer "Meister", welcher der Benutzergruppe "Bediener" angehört, hat Zugriff auf die projektierte Schaltfläche "Zur Rezepturanzeige".

Hinweis

Alternativ können Sie als Bediener mehrere Benutzergruppen anlegen mit unterschiedlichen Bedienberechtigungen z. B. Bediener Stufe1, Bediener Stufe 2.

8.4.4 Beispiel: Berechtigung erstellen und projektieren

Aufgabe

Im folgenden Beispiel erstellen Sie die Berechtigungen.

Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzergruppen".
2. Doppelklicken Sie in der Tabelle "Berechtigungen" auf "Hinzufügen".
3. Geben Sie als "Name" der Berechtigung "Rezepturdatensätze ändern" ein.
4. Wiederholen Sie die Schritte 2 und 3, um die weiteren Berechtigungen zu erstellen: "Systemparameter ändern", "Prozessparameter ändern".

Ergebnis

Benutzer | Benutzergruppen

Gruppen

Name	Nummer	Anzeigename	Kennwortalterung	Kommentar
Administratorengruppe	1	Administrator group	☐	Gruppe 'Administratoren' hat an.
Benutzer	2	Benutzer	☐	Gruppe 'Benutzer' hat anfänglich
<Hinzufügen>				

Berechtigungen

Aktiv	Name	Anzeigename	Nummer	Kommentar
☐	Benutzerverwaltung	Benutzerverwaltung	1	Berechtigung 'Benutzerverwaltung'
☐	Überwachen	Überwachen	2	Berechtigung 'Überwachen'
☑	Bedienen	Bedienen	3	Berechtigung 'Bedienen'
☐	Rezepturdatensätze ändern	Rezepturdatensätze ändern	4	
☐	Systemparameter ändern	Systemparameter ändern	5	
☐	Prozessparameter ändern	Prozessparameter ändern	6	
<Hinzufügen>				

8.4.5 Beispiel: Schaltfläche mit Zugriffsschutz projektieren

Aufgabe

Im folgenden Beispiel erstellen Sie mithilfe einer Systemfunktion eine Schaltfläche für einen Bildwechsel. Sie schützen die Schaltfläche "Zur Rezepturanzeige" vor unbefugter Bedienung. Dazu projektieren Sie die Berechtigung "Rezepturdatensätze ändern" an die Schaltfläche "Zur Rezepturanzeige".

Voraussetzungen

- Eine Berechtigung "Rezepturdatensätze ändern" ist erstellt.
- Ein Bild "Rezepturen" ist erstellt.
- Ein Bild "Start" ist erstellt und geöffnet.
- Eine Schaltfläche ist im Bild "Start" angelegt und markiert.

Vorgehensweise

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie als Text "Zur Rezepturanzeige" ein.
3. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse >Klicken".

4. Klicken Sie in der Tabelle "Funktionsliste" in der ersten Zeile auf den Eintrag "Funktion hinzufügen".
5. Wählen Sie in der Gruppe "Bilder" die Systemfunktion "AktiviereBild".
6. Klicken Sie im Feld "Bildname" auf die Schaltfläche ... . Ein Dialog zur Auswahl des Bildes wird geöffnet.
7. Wählen Sie das Bild "Rezepturen" und schließen Sie den Dialog über die Schaltfläche .
8. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Security".
9. Wählen Sie als "Berechtigung" "Rezepturdatensätze ändern".

Ergebnis

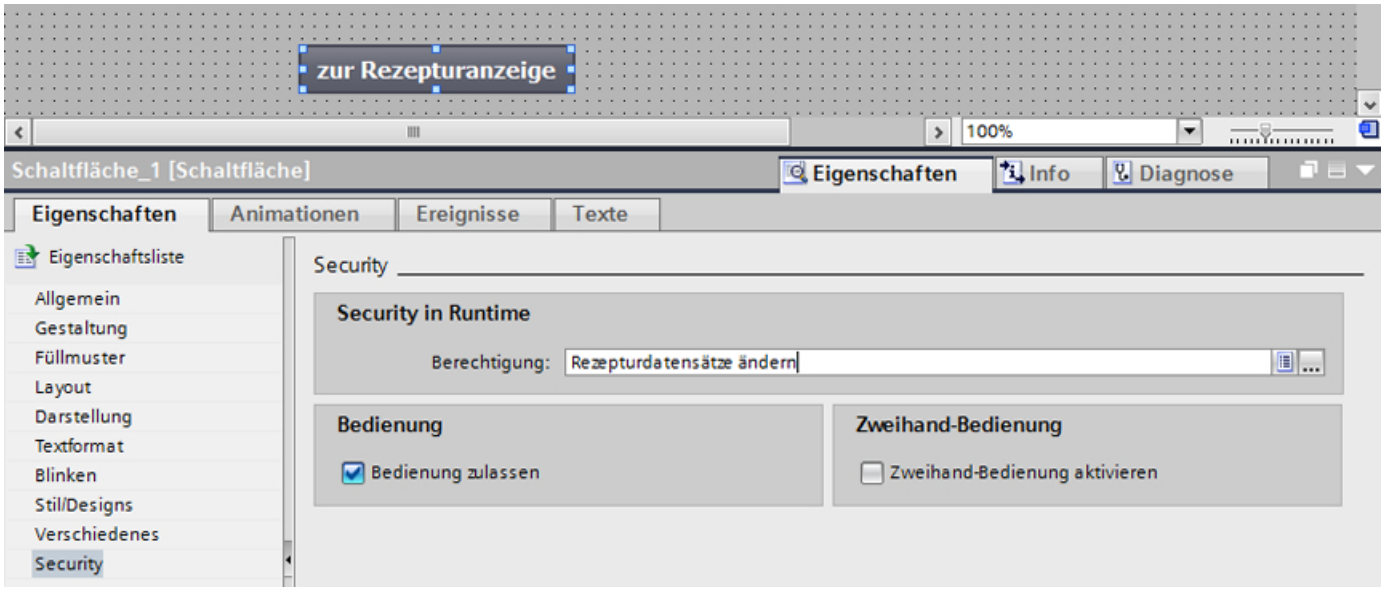

Der Zugriff auf die Schaltfläche "Zur Rezepturanzeige" ist geschützt. Wenn z. B. der Benutzer "Meier" in Runtime auf die Schaltfläche klickt, wird das Bild "Rezepturanzeige" geöffnet. Vorausgesetzt der Benutzer "Meier" ist korrekt angemeldet und hat die erforderliche Berechtigung. Das Bild "Rezepturen" enthält eine Rezepturanzeige und andere Bildobjekte.

Hinweis

Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, erscheint der Anmeldedialog. In Runtime Professional erscheint in diesem Fall die Meldung, dass für diesen Benutzer keine Bedienberechtigung vorliegt.

8.4.6 Beispiel: Benutzergruppen anlegen und Berechtigungen zuweisen

Aufgabe

Im folgenden Beispiel legen Sie die Benutzergruppen an und weisen ihnen Berechtigungen zu.

Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzergruppen".
2. Doppelklicken Sie in der Tabelle "Gruppen" auf "Hinzufügen".
3. Geben Sie als "Name" "Programmierer" ein.
4. Wiederholen Sie die Schritte 2 und 3, um die Benutzergruppen "Inbetriebsetzer" und "Bediener" anzulegen.
5. Klicken Sie in der Tabelle "Gruppen" auf "Administratorengruppe".
6. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Systemparameter ändern".

Zwischenergebnis

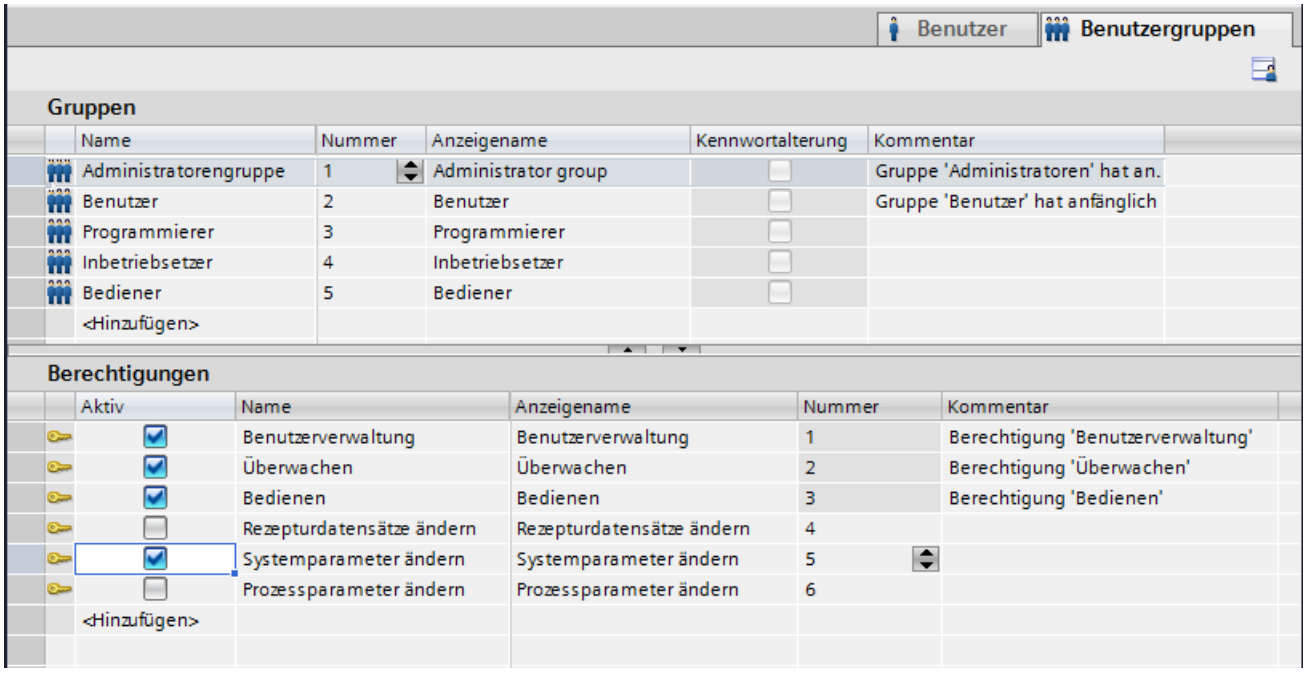

Vorgehensweise

1. Klicken Sie in der Tabelle "Gruppen" auf "Bediener".
2. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern".
3. Klicken Sie in der Tabelle "Gruppen" auf "Inbetriebsetzer".
4. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern", "Systemparameter ändern" und "Prozessparameter ändern".
5. Klicken Sie in der Tabelle "Gruppen" auf "Programmierer".
6. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern".

Ergebnis

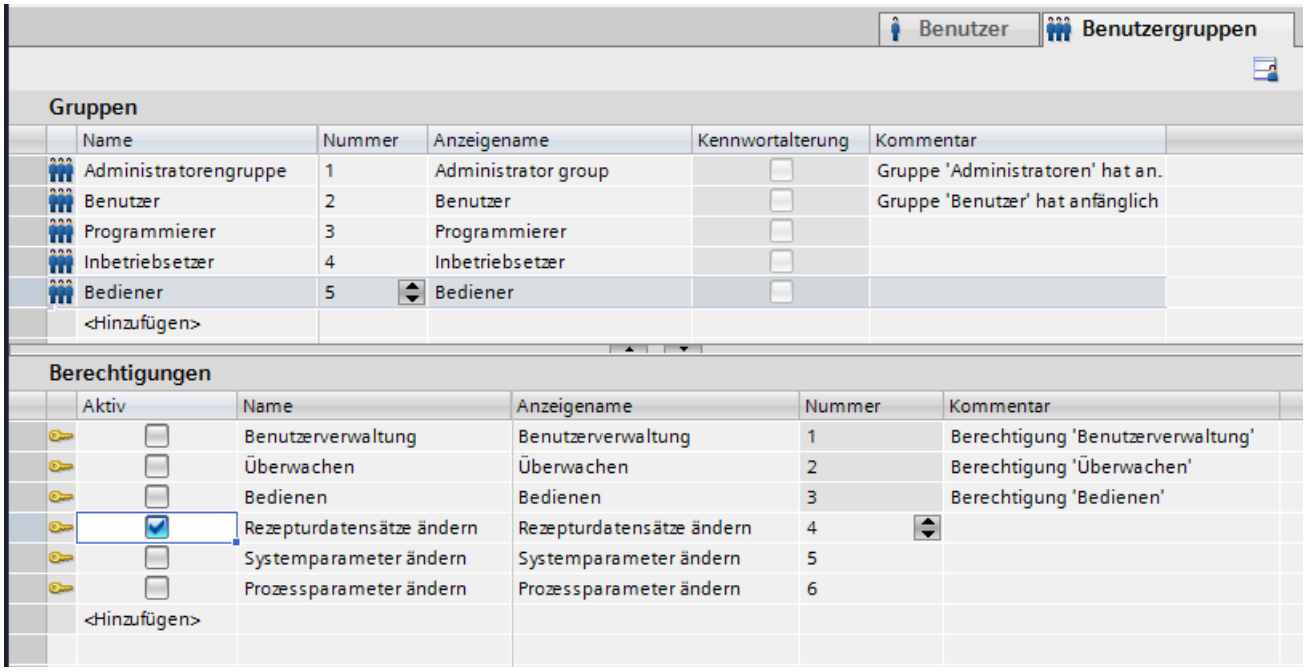

8.4.7 Beispiel: Benutzer anlegen und einer Benutzergruppe zuordnen

Aufgabe

Im folgenden Beispiel legen Sie die Benutzer an und ordnen sie ihren Benutzergruppen zu. Der Benutzer wird gleich nach der Eingabe des Namens alphabetisch einsortiert.

Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzer".
2. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
3. Geben Sie als Benutzernamen "Mueller" ein.
4. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Der Dialog zur Eingabe des Kennworts wird geöffnet.
5. Geben Sie als Kennwort "mueller" ein.
6. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
7. Schließen Sie den Dialog über das Symbol .
8. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Programmierer".

Zwischenergebnis

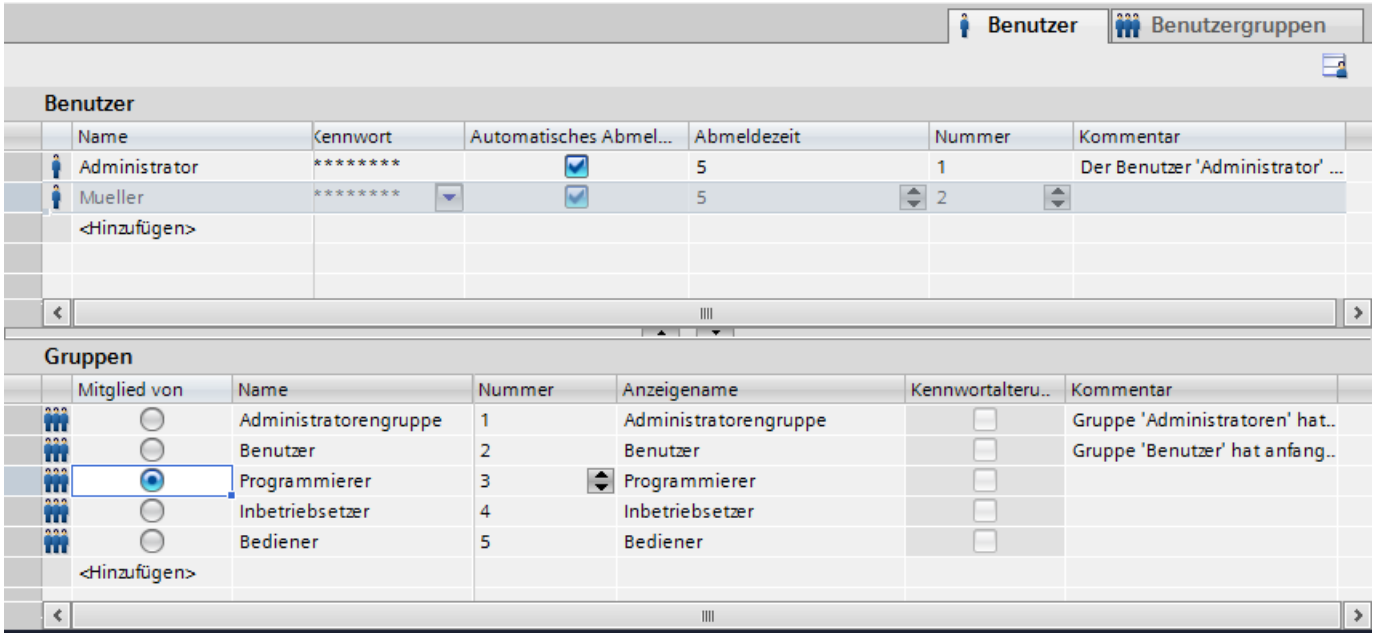

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
2. Geben Sie als Benutzernamen "Meier" ein.
3. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Der Dialog zur Eingabe des Kennworts wird geöffnet.
4. Geben Sie als Kennwort "meier" ein.
5. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
6. Schließen Sie den Dialog über das Symbol .
7. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Inbetriebsetzer".

8. Wiederholen Sie die Schritte 2 bis 6 für den Benutzer "Meister".
9. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Bediener".

Ergebnis

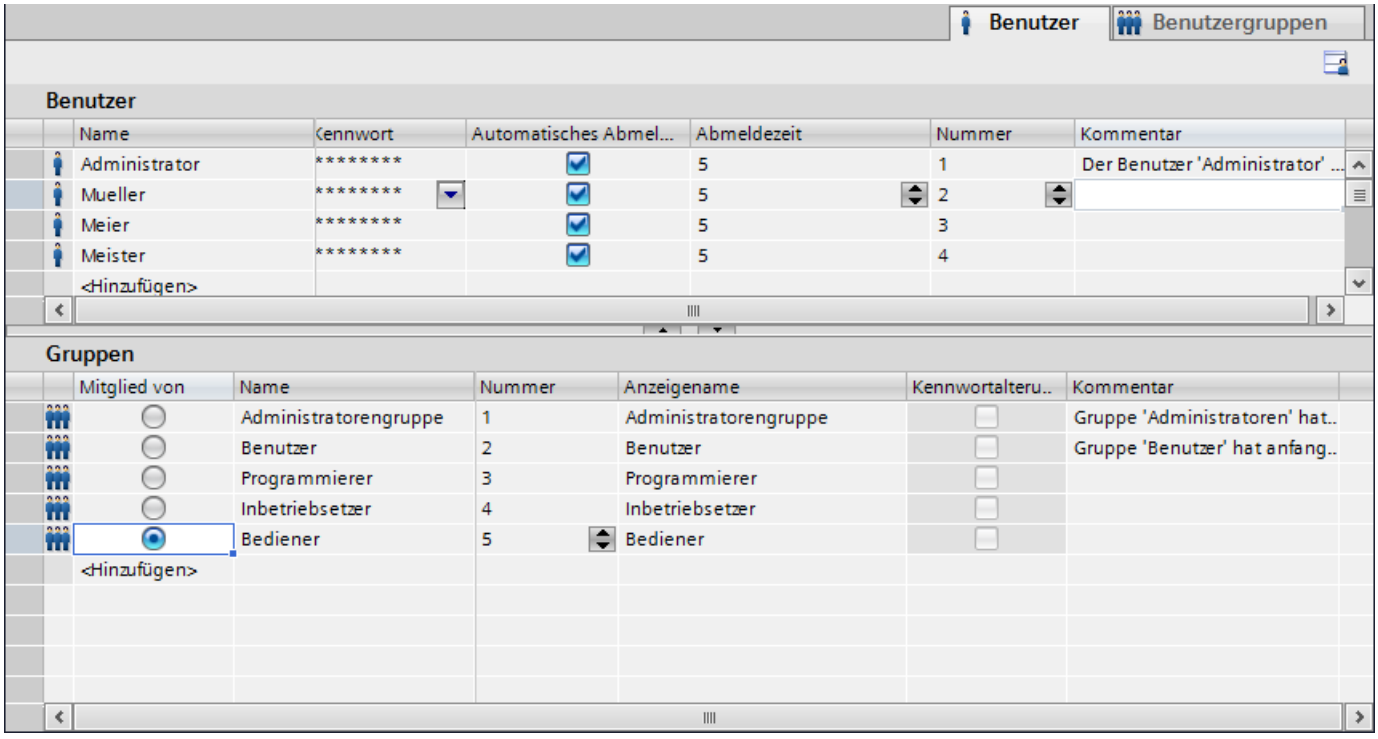

Mit Systemfunktionen und Runtime Scripting arbeiten 9

9.1 Grundlagen

9.1.1 Runtime Scripting

Begriffsdefinition

Folgende Begriffe werden in der Dokumentation verwendet:

Begriff	
Runtime Scripting	Oberbegriff für alle Tätigkeiten in benutzerdefinierte Funktionen und lokalen Skripten.
Funktion	Oberbegriff für Systemfunktionen und benutzerdefinierte Funktionen
Systemfunktionen	Systemfunktionen sind alle mit WinCC gelieferten Funktionen. Die Systemfunktionen werden entweder in Funktionslisten, benutzerdefinierten Funktionen oder lokalen Skripten verwendet.
Benutzerdefinierte Funktionen	Benutzerdefinierte Funktionen sind im Editor "Skripte" geschriebene Funktionen. Zur näheren Spezifikation werden in der Dokumentation die Begriffe "Benutzerdefinierte VB-Funktion" und "Benutzerdefinierte C-Funktion" verwendet.
Lokales Skript	Ein lokales Skript wird direkt an der Verwendungsstelle, z. B. einer Objekteigenschaft, erstellt und besitzt einen zyklischen oder azyklischen Trigger. Zur näheren Spezifikation werden in der Dokumentation die Begriffe "Lokales VB-Skript" und "Lokales C-Skript" verwendet
VBS/VBScript	Abk. für Visual Basic Script

Verfügbarkeit

	WinCC RT Advanced und Panels	WinCC RT Professional
Benutzerdefinierte VB-Funktionen	X	X
Benutzerdefinierte C-Funktionen	--	X
Lokale VB-Skripte	--	X
Lokale C-Skripte	--	X

Runtime Scripting anwenden

Die folgende Abbildung zeigt die Entscheidungshilfe vor anstehenden Programmieraufgaben:

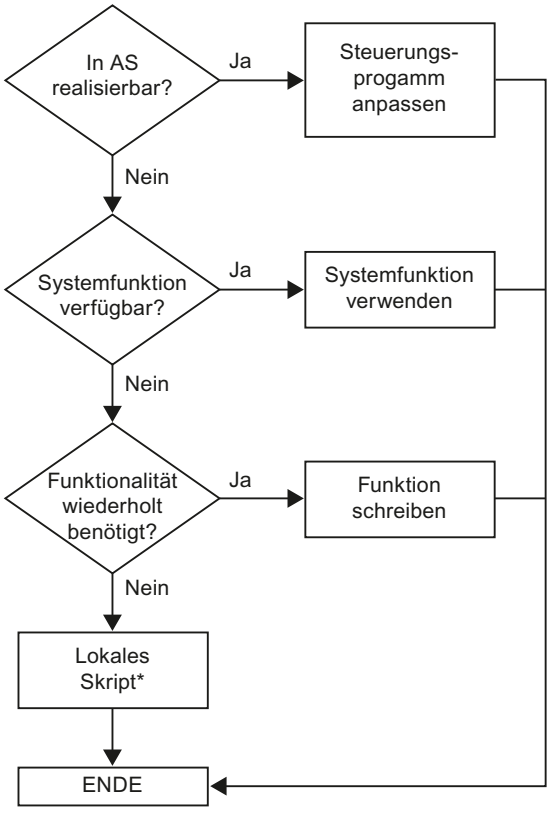

* In WinCC RT Advanced und Panels werden lokale Skripte nicht unterstützt.

Benutzerdefinierte Funktion oder lokales Skript?

Ob Sie Ihre Programmieraufgabe mit einer benutzerdefinierten Funktion oder einem lokalen Skript realisieren, hängt vom Einsatzzweck und letztendlich von der Anzahl der benutzerdefinierten Funktionen oder lokalen Skripte ab.

Verwenden Sie lokale Skripte für Hintergrundtätigkeiten, z. B. für den täglichen Ausdruck eines Protokolls, für die Überwachung von Variablen oder für die Ausführung von Berechnungen.

Benutzerdefinierte Funktionen verwenden Sie, um Code an mehreren Stellen Ihres Projekts verfügbar zu machen, aber nur einmal zu erstellen. Statt den Code mehrfach einzugeben, rufen Sie nur die entsprechende benutzerdefinierte Funktion auf. Ihr Code ist übersichtlicher und leichter zu pflegen.

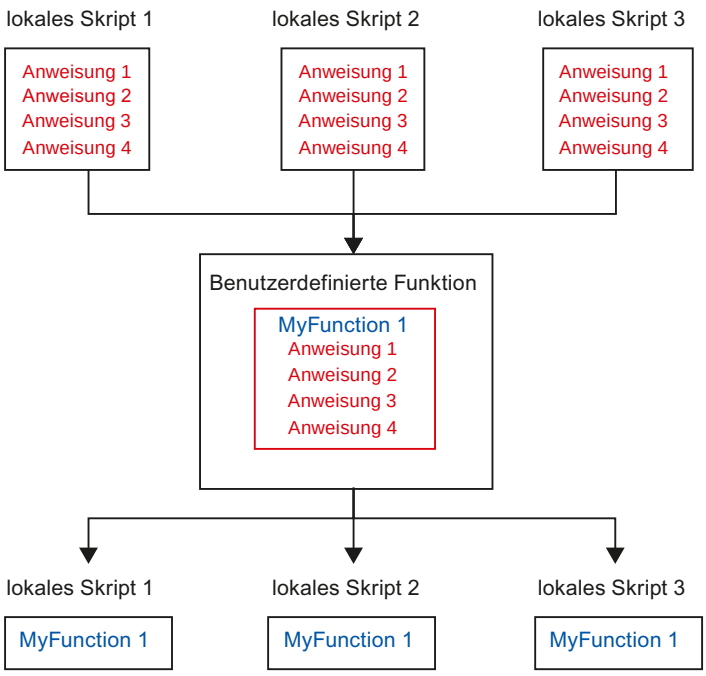

Siehe auch

Benutzerdefinierte Funktionen (Seite 1435)

Lokale Skripte (Seite 1437)

9.1.2 Systemfunktionen

Einsatzmöglichkeiten

- Funktionslisten
  Mit der Funktionsliste führen Sie beim Eintreten des Ereignisses die projektierten Systemfunktionen aus.
- Benutzerdefinierte Funktionen
  Wenn das Bediengerät benutzerdefinierte Funktionen unterstützt, verwenden Sie Systemfunktionen in Verbindung mit Anweisungen und Bedingungen im Code der benutzerdefinierten Funktion. Auf diese Weise führen Sie eine benutzerdefinierte Funktion in Abhängigkeit von einem bestimmten Systemzustand aus. Darüber hinaus können Sie z. B. Rückgabewerte von Systemfunktionen auswerten. Abhängig vom Rückgabewert führen Sie dann z. B. Prüffunktionen aus, die wiederum den Ablauf der Funktion beeinflussen.

Anwendungsbeispiele von Systemfunktionen

- Berechnungen, z. B. Variablenwert um einen bestimmten oder variablen Betrag erhöhen.
- Archivfunktionen, z. B. um ein Prozesswertarchiv starten.
- Einstellungen, z. B. um die Steuerung wechseln oder ein Bit in der Steuerung setzen.
- Meldungen, z. B. nach dem Ändern des Benutzers

Verwendung

Systemfunktionen verwenden Sie in einer Funktionsliste oder in einer benutzerdefinierten Funktion.

- Funktionsliste
  Wenn Sie eine Funktionsliste projektieren, wählen Sie die Systemfunktionen aus einer nach Kategorien geordneten Auswahlliste aus:

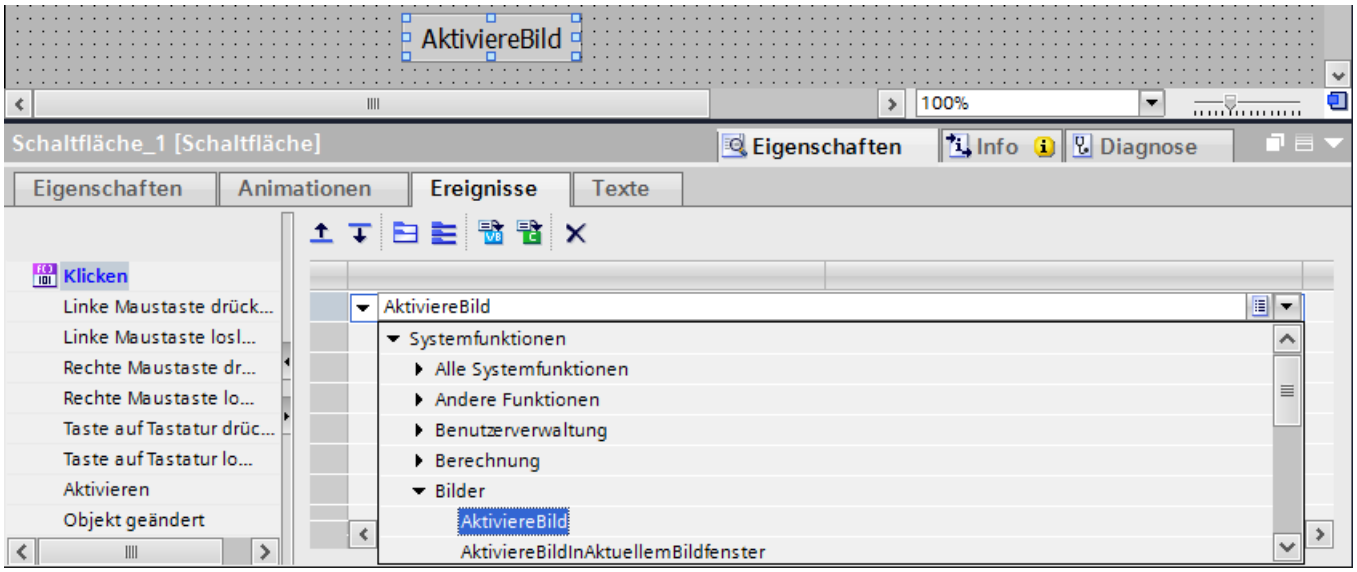

- Benutzerdefinierte Funktionen
  Wenn Sie eine Systemfunktion in einer benutzerdefinierten Funktion verwenden, wählen Sie diese aus einer Auswahlliste. Um die Auswahlliste zu öffnen, verwenden Sie die Tastenkombination <Strg+Leer> oder klicken Sie auf .

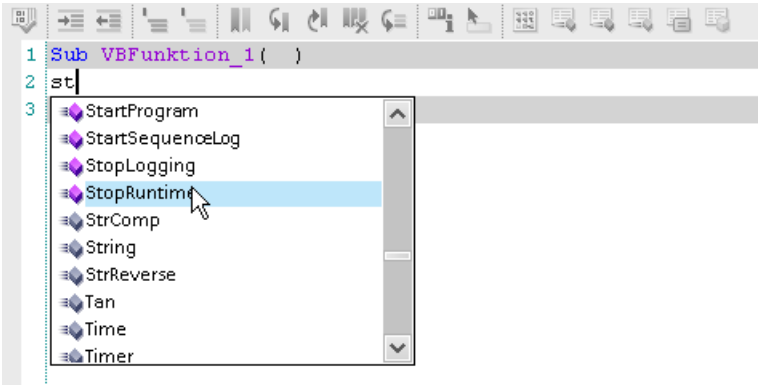

Sprachabhängigkeit

In der Funktionsliste sind die Namen der Systemfunktionen abhängig von der eingestellten Projektiersprache. Die Funktionalität ist damit für den Projekteur sofort erkennbar.

In benutzerdefinierten Funktionen verwenden Sie den englischen Namen der Systemfunktion. Den englischen Namen der Systemfunktion finden Sie in der Referenz.

Verfügbarkeit bei WinCC Runtime Advanced und Panels

Welche Systemfunktionen verfügbar sind, ist abhängig vom gewählten Bediengerät.

Bediengerätewechsel

Wenn Sie in einer Funktionsliste Systemfunktionen verwenden, die auf dem eingestellten Bediengerät nicht verfügbar sind, werden diese Systemfunktionen farblich gekennzeichnet.

Wenn Sie in einer benutzerdefinierten Funktion eine Systemfunktion verwenden, die auf dem eingestellten Bediengerät nicht verfügbar ist, erhalten Sie einen Warnhinweis. Zusätzlich wird die entsprechende Systemfunktion mit einer blauen Wellenlinie unterstrichen.

Siehe auch

Systemfunktionen für Archive (Seite 650)

9.1.3 Benutzerdefinierte Funktionen

Einsatzmöglichkeiten

Benutzerdefinierte Funktionen nutzen Sie z. B. für Folgendes:

- Funktionsliste projektieren
  In einer benutzerdefinierten Funktion haben Sie die Möglichkeit, benutzerdefinierte Funktionen und Systemfunktionen abhängig von Bedingungen auszuführen oder wiederholt ausführen zu lassen. Die benutzerdefinierte Funktion fügen Sie dann, z. B. einer Funktionsliste hinzu.
- Neue Funktionalitäten programmieren
  Benutzerdefinierte Funktionen sind nur in dem Projekt gültig, in dem sie definiert wurden. Für benutzerdefinierte Funktionen definieren Sie Übergabeparameter und Rückgabewerte,z. B. zum Umrechnen von Werten.

Hinweis

In einer Funktionsliste können Sie nur benutzerdefinierte Funktionen einer Programmiersprache verwenden. Mit der Auswahl der ersten benutzerdefinierten Funktion bestimmen Sie, ob VB-Funktionen oder C-Funktionen in der Funktionsliste wählbar sind.

Anwendungsbeispiele von benutzerdefinierten Funktionen

- Werte umrechnen
  Mithilfe von Funktionen rechnen Sie Werte zwischen verschiedenen Maßeinheiten um, z. B. Temperaturen.
- Variablenwerte statistisch auswerten
- Kurve in einer f(t)-Kurvenanzeige zeichen

Eigenschaften von benutzerdefinierten Funktionen

Eine benutzerdefinierte Funktion hat folgende Eigenschaften:

- Name
- Parameter (optional)
- Rückgabewert und Typ (optional)
- zentrale Änderbarkeit
- besitzt keinen Trigger. Sie müssen die benutzerdefinierte Funktion explizit aufrufen, z. B. in einer Funktionsliste

Organisation von benutzerdefinierten Funktionen

Benutzerdefinierte Funktionen erstellen Sie mit dem Editor "Skripte". Nähere Hinweise hierzu finden Sie unter "Editor "Skripte" (Seite 1444)".

Benutzerdefinierte Funktionen werden im Projekt gespeichert. Um benutzerdefinierte Funktionen zu schützen, richten Sie einen Know-how-Schutz ein. Nähere Hinweise hierzu finden Sie unter Know-how-Schutz einrichten (Seite 1477).

- Projektnavigation
  In der Projektnavigation werden die benutzerdefinierten Funktionen unter "VB-Skripte" oder unter "C-Skripte" aufgelistet.

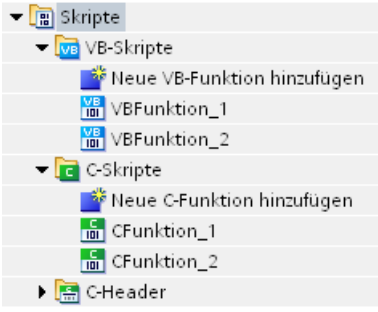

- Funktionsliste
  In einer Funktionsliste werden die benutzerdefinierten Funktionen unter "VB-Funktionen" oder "C-Funktionen" aufgelistet.
- Benutzerdefinierte Funktion
  Wenn Sie eine Systemfunktion in einer benutzerdefinierten Funktion verwenden, wählen Sie diese aus einer Auswahlliste. Um die Auswahlliste zu öffnen, verwenden Sie die Tastenkombination <Strg+Leer> oder klicken Sie auf .

Siehe auch

Runtime Scripting (Seite 1431)

Know-how-Schutz einrichten (Seite 1477)

Editor "Skripte" (Seite 1444)

9.1.4 Lokale Skripte

Einsatzmöglichkeiten

Lokale Skripte nutzen Sie z. B. für Folgendes:

- Dynamisierung von Objekten
  Sie verwenden ein lokales Skript, um den Wert einer Objekteigenschaft in Runtime zu bestimmen. Auf diese Weise ändern Sie z. B. die Farbe in Abhängigkeit von einem Variablenwert.
- Aufgaben zu einem beliebigen Zeitpunkt ausführen
  Sie verwenden ein lokales Skript zum Planen von Aufgaben, z. B. täglicher Ausdruck eines Protokolls bei Schichtende.

Anwendungsbeispiele von lokalen Skripten

- Sollwertvorgaben für Variablen an die Bedienung eines Grafikobjekts projektieren, um z. B. per Mausklick einen Wert für die Steuerung vorzugeben.
- Die Umschaltung der Runtime-Sprache an die Bedienung eines Grafikobjekts projektieren.
- Statusüberwachung von Variablen, z. B. um sicherzustellen, dass ein Wert gültig ist.

Eigenschaften von lokalen Skripten

Ein lokales Skript erstellen Sie direkt an der Verwendungsstelle. Ein lokales Skript hat folgende Eigenschaften:

- Vordefinierter Name
- Vordefinierte Parameter
- Vordefinierter Rückgabewert (optional)
- Änderbarkeit nur an Verwendungsstelle
- Besitzt einen Trigger
- Aufruf von benutzerdefinierten Funktionen, Systemfunktionen

Trigger

Trigger sind notwendig, um lokale Skripte in Runtime auszuführen. Der Trigger wird beim Anlegen eines lokalen Skripts entweder automatisch ermittelt oder vom Projekteur festgelegt.

Folgende Trigger werden in WinCC unterstützt:

- Azyklische Trigger, z. B. Sie planen ein einmaliges Beenden von Runtime für Wartungsarbeiten.
- Zyklische Trigger, z. B. Sie planen eine Aufgabe, die den täglichen Ausdruck eines Protokolls startet.
- Ereignisse, z. B. beim Klick auf die Schaltfläche.

Die folgende Abbildung zeigt unterschiedlichen Triggerarten:

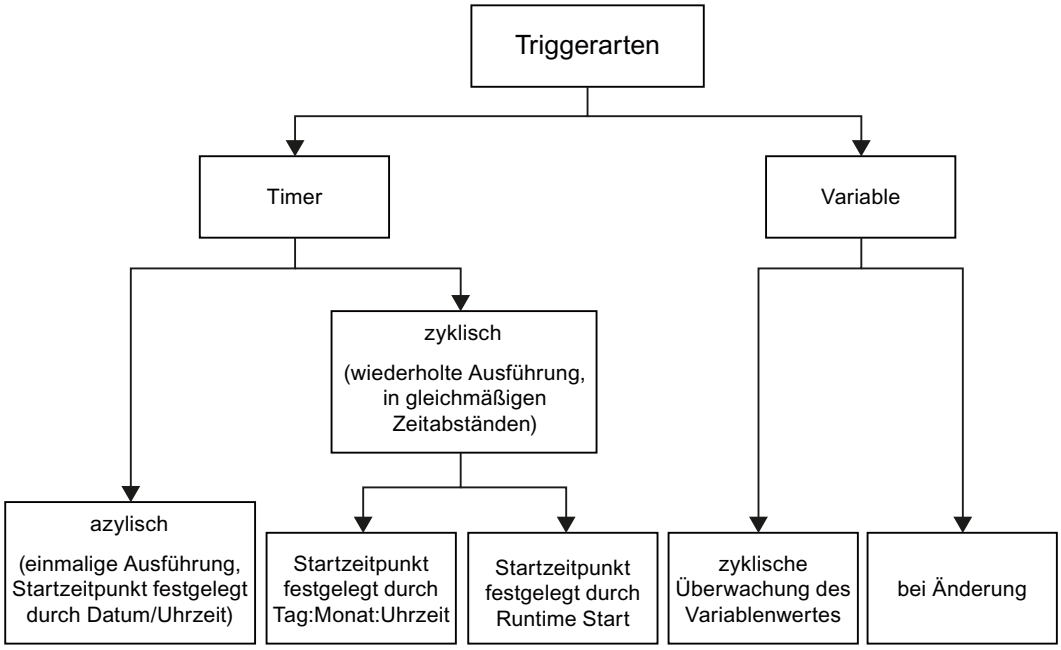

Trigger kommen in folgenden Editoren zu Einsatz:

- Aufgabenplaner
  Im Aufgabenplaner projektieren Sie zuerst den Trigger. Den Trigger verbinden Sie dann mit dem lokalen Skript oder einer Funktionsliste.
- Bilder
  Im Bild projektieren Sie zuerst in der Eigenschaftsliste ein lokales Skript, um eine Objekteigenschaft zu dynamisieren.
  Beim Anlegen des lokalen Skripts wird der Trigger automatisch ermittelt. Den Trigger können Sie ändern, z. B. in Variablentrigger.

Siehe auch

Runtime Scripting (Seite 1431)

9.2 Arbeiten mit Funktionslisten

9.2.1 Grundlagen zur Funktionsliste

Einleitung

Mit der Funktionsliste führen Sie beim Eintreten des projektierten Ereignisses eine oder mehrere Systemfunktionen und benutzerdefinierte Funktionen aus.

Prinzip

Die Funktionsliste projektieren Sie an ein Ereignis eines Objekts, z. B. an ein Bildobjekt oder an eine Aufgabe. An jedes Ereignis können Sie genau eine Funktionsliste projektieren. Welche Ereignisse zur Verfügung stehen, ist abhängig vom ausgewählten Objekt und Bediengerät.

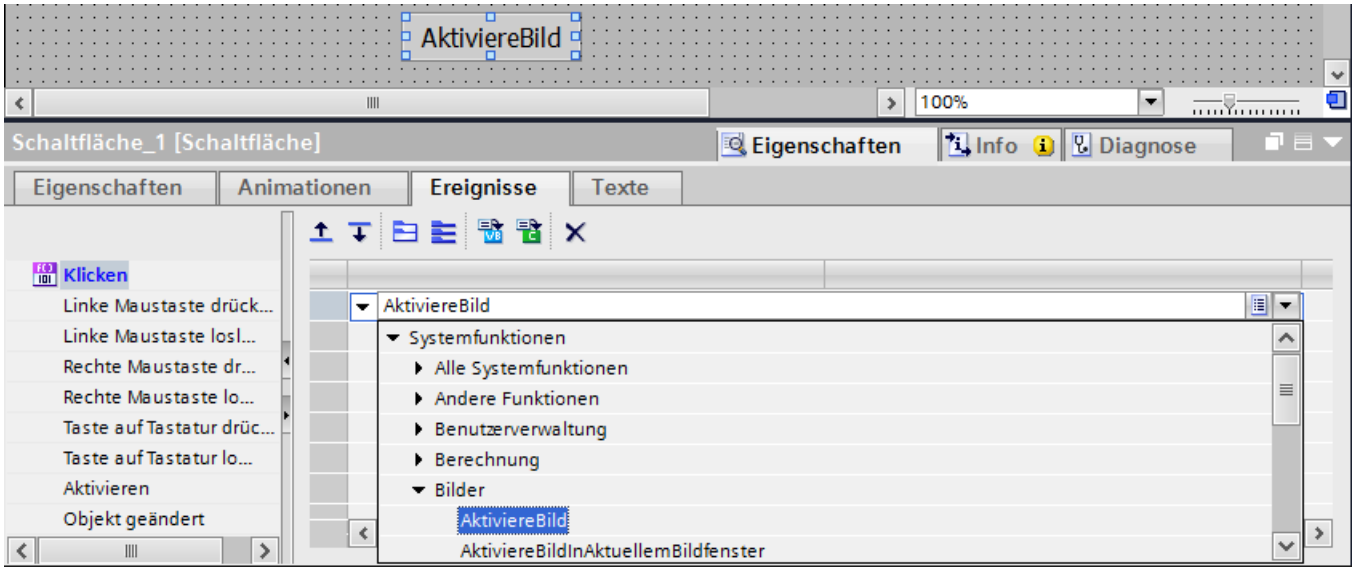

Ereignisse treten nur ein, während das Projekt in Runtime ist. Ereignisse sind z. B.:

- Ausführen einer Aufgabe
- Drücken einer Schaltfläche
- Quittieren einer Meldung

Hinweis

Die Auswahl der projektierbaren Systemfunktionen in einer Funktionsliste ist abhängig vom gewählten Bediengerät. Weitere Informationen finden Sie unter "Geräteabhängigkeit von Systemfunktionen".

Siehe auch

Funktionsliste (Seite 1878)

9.2.2 Eigenschaften einer Funktionsliste

Statusinformation

Während der Projektierung werden die Projektdaten im Hintergrund geprüft.

Bei folgenden Ursachen wird die Funktionsliste in Runtime nicht ausgeführt und die fehlerhaften Einträge rot markiert:

- Mindestens eine Systemfunktion oder eine benutzerdefinierte Funktion ist nicht vollständig mit Parametern versorgt.
- Mindestens eine Systemfunktion oder eine benutzerdefinierte Funktion ist enthalten, die vom gewählten Bediengerät nicht unterstützt wird, z. B. durch Wechsel des Gerätetyps

Abarbeitung von Systemfunktionen und benutzerdefinierten Funktionen (WinCC Runtime Advanced und Panels)

Eine Funktionsliste wird in Runtime von oben nach unten abgearbeitet. Wenn in der Funktionsliste Systemfunktionen mit längerer Laufzeit enthalten sind, werden diese asynchron abgearbeitet. Nähere Hinweise hierzu finden Sie unter "Aufruf von Systemfunktionen (Seite 1447)" und "Aufruf von benutzerdefinierten Funktionen (Seite 1449)".

Siehe auch

Aufruf von Systemfunktionen (Seite 1447)

Aufruf von benutzerdefinierten Funktionen (Seite 1449)

9.2.3 Funktionsliste projektieren

Einleitung

Sie projektieren eine Funktionsliste, indem Sie Systemfunktionen und benutzerdefinierte Funktionen aus einer Auswahlliste auswählen. Die Systemfunktionen sind in der Auswahlliste nach Kategorien geordnet.

In einer Funktionsliste verwenden Sie nur benutzerdefinierte Funktionen einer Programmiersprache. Mit der Auswahl der ersten benutzerdefinierten Funktion bestimmen Sie, ob benutzerdefinierte VB-Funktionen oder benutzerdefinierte C-Funktionen in der Funktionsliste wählbar sind. Welche Programmiersprachen verfügbar sind, ist abhängig vom gewählten Bediengerät.

Wenn Sie z. B. eine benutzerdefinierte VB-Funktion erstellt haben, finden Sie die benutzerdefinierte Funktion unter dem Eintrag "VB-Funktionen".

Voraussetzung

Objekt besitzt mindestens ein projektierbares Ereignis.

Vorgehen

Um eine Funktionsliste zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor, in dem sich das Objekt befindet.
2. Markieren Sie das Objekt.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Wählen Sie das Ereignis, an das Sie die Funktionsliste projektieren wollen.
4. Markieren Sie im Inspektorfenster in der Auswahlliste den Eintrag "<Funktion hinzufügen>".
5. Wählen Sie aus der Auswahlliste die gewünschte Systemfunktion oder die gewünschte benutzerdefinierte Funktion aus. Alternativ geben Sie den Namen der Systemfunktion oder den Namen der benutzerdefinierten Funktion ein.

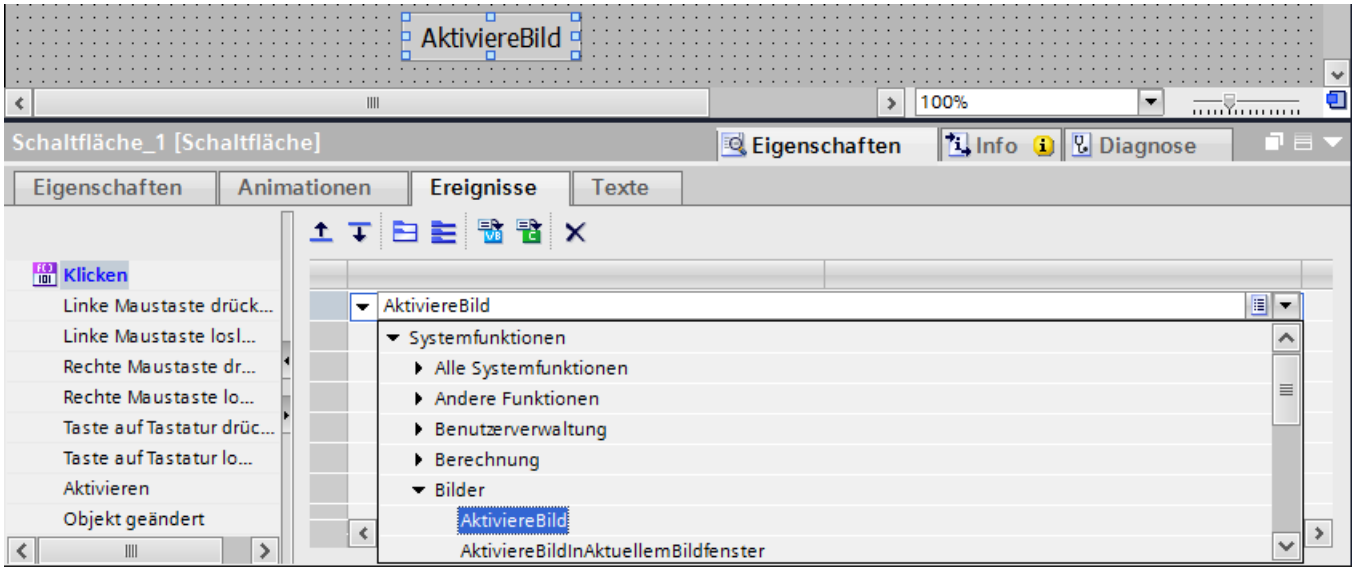

Die Systemfunktion oder die benutzerdefinierte Funktion wird in die Funktionsliste eingetragen.

6. Wenn die Systemfunktion oder die benutzerdefinierte Funktion Parameter besitzt, wählen Sie für die Parameter die entsprechenden Werte aus.

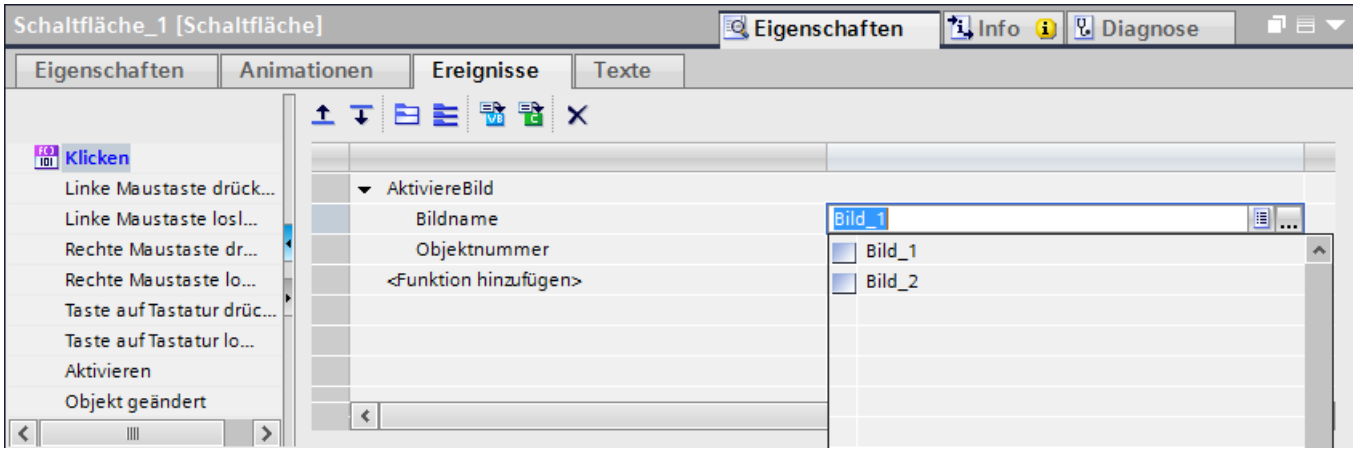

7. Wenn Sie der Funktionsliste weitere Systemfunktionen oder benutzerdefinierten Funktionen hinzufügen wollen, dann wiederholen Sie die Schritte 4.) bis 6.).

Ergebnis

Die Funktionsliste ist projektiert. Im Inspektorfenster wird neben dem projektierten Ereignis der Status der Funktionsliste angezeigt. Wenn in Runtime das projektierte Ereignis eintritt, wird die Funktionsliste von oben nach unten abgearbeitet.

Siehe auch

Beispiel: Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige (Seite 1509)

9.2.4 Funktionsliste bearbeiten

Einleitung

Sie können eine Funktionsliste wie folgt bearbeiten:

- Die Reihenfolge von Systemfunktionen und von benutzerdefinierte Funktionen ändern.
- Systemfunktionen oder benutzerdefinierte Funktionen entfernen.

Voraussetzung

Die Funktionsliste ist projektiert und geöffnet.

Reihenfolge einer Systemfunktion oder einer benutzerdefinierten Funktion ändern

1. Selektieren Sie in der Funktionsliste die gewünschte Systemfunktion oder benutzerdefinierte Funktion.
2. Klicken Sie dann so oft im Inspektorfenster auf den entsprechenden Richtungspfeil, bis die Systemfunktion oder benutzerdefinierte Funktion an der gewünschten Position steht.

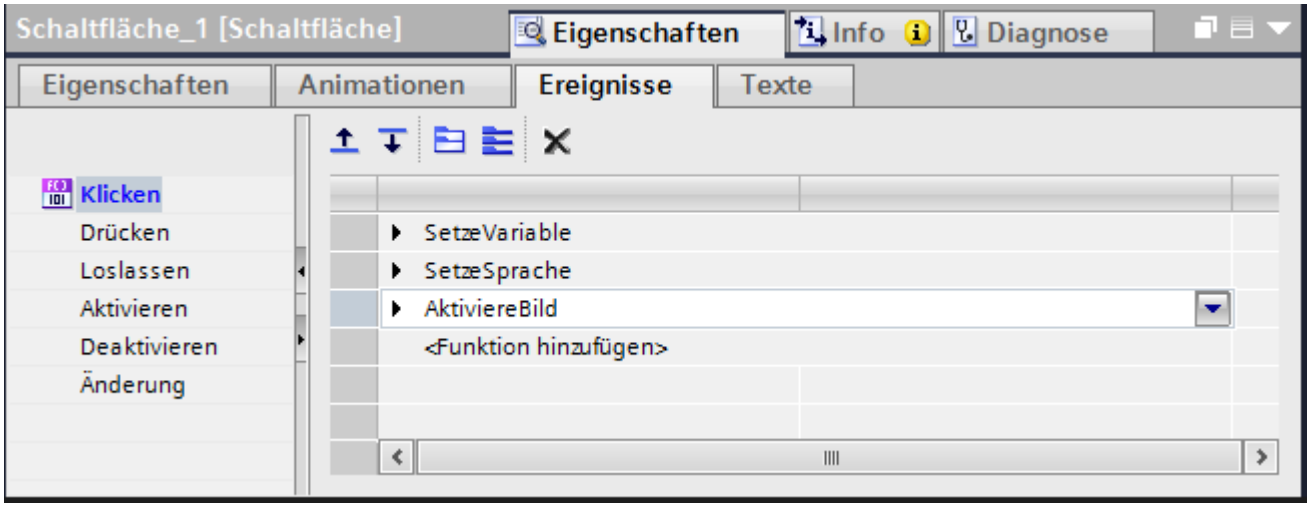

Reihenfolge mehrerer Systemfunktionen oder benutzerdefinierte Funktionen ändern

1. Halten Sie die Taste <Shift> gedrückt.
2. Klicken Sie mit der Maus nacheinander auf die gewünschten Systemfunktionen oder benutzerdefinierte Funktionen.
3. Verschieben Sie die Auswahl mit Drag&Drop an die gewünschte Position.

Systemfunktion oder benutzerdefinierte Funktion entfernen

1. Selektieren Sie in der Funktionsliste die gewünschte Systemfunktion oder die benutzerdefinierte Funktion.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

9.3 Arbeiten mit benutzerdefinierten Funktionen

9.3.1 Editor "Skripte"

Einleitung

Der Editor "Skripte" unterstützt Sie beim Erstellen von benutzerdefinierten Funktionen mit Funktionalitäten wie Autovervollständigung und Syntaxhervorhebung. Über die Task Card "Anweisungen" können Sie z. B. Code-Vorlagen für häufig benötigte Anweisungen einfügen.

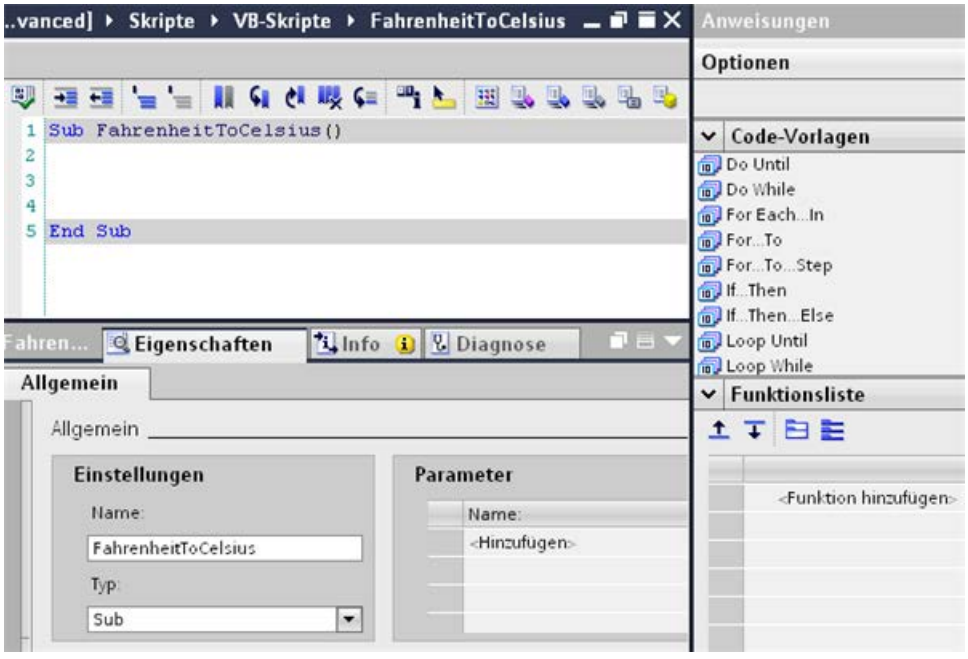

Autovervollständigung und Parametereingabe

Der Editor "Skripte" unterstützt Sie beim Programmieren kontextabhängig mit Autovervollständigung und bei der Parametereingabe.

Wenn Sie Systemfunktionen verwenden, werden Ihnen die Funktionsparameter als Tooltip angezeigt.

<Strg+J>

Mit der Tastenkombination <Strg+J> rufen Sie kontextabhängig die Objektliste auf. Verwenden Sie die Objektliste, um Systemfunktionen, Methoden und Eigenschaften mit Werten zu versorgen: Wählen Sie z. B. Bilder, Bildobjekte, Variablen oder Farben aus. Sie übernehmen ein selektiertes Objekt aus der Objektliste mit Doppelklick in den Code.

Hinweis

Variablen und Bilder können Sie auch mit Drag&Drop einfügen. Verwenden Sie Drag&Drop z. B. zum Belegen von Parametern.

<Strg+Leer>

Mit der Tastenkombination <Strg+Leer> rufen Sie eine Liste mit folgendem Inhalt auf:

- Konstanten und Funktionen der im Editor "Skripte" verwendeten Programmiersprache.
- Benutzerdefinierte Funktionen
- Systemfunktionen

Integrieren von Steuerzeichen in Zeichenketten

Wenn Sie Steuerzeichen wie Tabulatoren oder Zeilenumbrüche in Zeichenketten verwenden wollen, verwenden Sie die Funktion "Chr()". Als Parameter übergeben Sie den dezimalen ASCII-Code des gewünschten Steuerzeichens:

Beispiel:

- Chr(9) gibt "Tabulator" zurück.
- Chr(15) gibt "SI" (Shift in) zurück.
- Chr(20) gibt "DC4" (Device Control 4) zurück.

Weiterführende Informationen zu dieser VBS-Funktion finden Sie im Internet (https://msdn.microsoft.com/en-us/library/ws6aa3sf(v=vs.84).aspx).

Task Card "Anweisungen"

Die Task Card "Anweisungen" enthält die Paletten "Code-Vorlagen" und "Funktionsliste".

- Code-Vorlagen
  Um häufig verwendete Anweisungen einzufügen, doppelklicken Sie in der Palette "Code-Vorlagen" die gewünschte Anweisung. Ersetzen Sie die Platzhalter durch die verwendeten Ausdrücke.
- Funktionsliste
  In der Funktionsliste wählen Sie Systemfunktionen und benutzerdefinierte Funktionen aus einer Auswahlliste. Verfahren Sie wie bei der Projektierung einer Funktionsliste. Um die Funktionsliste in den Code zu übernehmen, klicken Sie auf die Schaltfläche "Übernehmen". Die Funktionsliste wird automatisch in die korrekte Syntax umgewandelt.

Hinweis

Nicht alle Systemfunktionen sind im Editor "Skripte" verfügbar. Nähere Hinweise hierzu finden Sie in der Referenz.

Editierfunktionen

Um den Code durch Einrückungen und Verwendung von Kommentaren lesbarer zu machen, verwenden Sie die Schaltflächen der Symbolleiste "Erweitertes Bearbeiten".

Lesezeichen helfen Ihnen, auch bei vielen Codezeilen, den Überblick zu behalten.

Weitere Informationen zu jeder Schaltfläche erhalten Sie in der dazugehörigen Direkthilfe.

Synchronisation von Variablen und Objekten

Wenn Sie im Projekt Variablennamen oder Objektnamen ändern, die Sie in einer benutzerdefinierten Funktion verwenden, werden die Namen in der benutzerdefinierten Funktion automatisch angepasst.

Wenn Sie in der Funktion den Variablennamen oder Objektnamen ändern, werden die Variablennamen oder Objektnamen im Projekt nicht angepasst. Beim Ausführen der benutzerdefinierten Funktion wird eine entsprechende Fehlermeldung ausgegeben.

Syntaxhervorhebung

Im Editor "Skripte" werden die Schlüsselwörter mit verschiedenen Farben hervorgehoben. Unbekannte Wörter werden mit einer roten Wellenlinie unterstrichen.

Die Tabelle zeigt die voreingestellten Farben für die wichtigsten Schlüsselwörter:

Farbe	Bedeutung	Beispiel
Blau	Schlüsselwort	If, Then
Dunkles Violett	Funktion	FahrenheitToCelsius
Schokolade	Systemfunktion	SetTag
Orange	HMI-Variable	Tag_1
Grün	Kommentar	'Dies ist ein Kommentar'

Anpassen von Anzeigeeigenschaften

Wenn Sie eine benutzerdefinierte Funktion geöffnet haben, können Sie die Anzeigeeigenschaften im Editor individuell anpassen. Wählen Sie dazu den Menübefehl "Extras > Einstellungen". Klicken Sie im Editor "Einstellungen" auf "Allgemein > Texteditoren". Im Arbeitsbereich ändern Sie z. B. die Farben zur Syntaxhervorhebung oder die Einzüge für den Code.

Siehe auch

Benutzerdefinierte VB-Funktion erstellen (Seite 1458)

Benutzerdefinierte C-Funktion erstellen (Seite 1469)

Lokale Skripte (Seite 1480)

Editor für lokale Skripte (Seite 1482)

9.3.2 Aufruf von Systemfunktionen

Einleitung

In einer benutzerdefinierten Funktion können Sie Systemfunktionen aufrufen. Die Syntax ist abhängig von der verwendeten Programmiersprache. Welche Programmiersprache unterstützt wird, ist abhängig vom gewählten Bediengerät.

Um in einer benutzerdefinierten Funktion eine Systemfunktion aufzurufen, haben Sie folgende Möglichkeiten:

- Mit <Strg+Leer> oder mit
- Direkte Eingabe
- Über die Palette "Funktionsliste"

Mit <Strg+Leer> oder mit

Öffnen Sie mit <Strg+Leer> die Auswahlliste und wählen Sie die gewünschte Systemfunktion aus.

Direkte Eingabe

Sie geben die Systemfunktionen direkt in den Code ein. Verwenden Sie den englischen Namen der Systemfunktion. Den englischen Namen der Systemfunktion finden Sie in der Systemfunktionsreferenz unter "Syntax". Die eingestellte Editiersprache wird nicht berücksichtigt.

Referenzen auf Objekte z. B. Bilder, Verbindungen und Archive übergeben Sie in Anführungszeichen:

- Aufruf von Systemfunktionen ohne Rückgabewert in VBS

```
SetTag "Tag1",64
```

- Aufruf von Systemfunktionen mit Rückgabewert in VBS

```
InverseLinearScaling "Xvalue","Tag1", 0, 1
```

- Aufruf von Systemfunktionen ohne Rückgabewert in C

```
SetTag ("Tag1",64);
```

- Aufruf von Systemfunktionen mit Rückgabewert in C

```
InverseLinearScaling ("Xvalue","Tag1", 0, 1);
```

Über die Palette "Funktionsliste"

Wählen Sie in der Palette "Funktionsliste" die gewünschte Systemfunktion aus einer Auswahlliste. Verfahren Sie wie bei der Projektierung einer Funktionsliste. Nähere Hinweise hierzu finden Sie unter "Funktionsliste projektieren".

Um die Liste in den Code zu übernehmen, klicken Sie auf die Schaltfläche "Übernehmen". Die Liste wird automatisch in die korrekte Syntax umgewandelt.

Bediengerätewechsel (Runtime Panels und Advanced)

Der Code einer benutzerdefinierten Funktion ist abhängig vom gewählten Bediengerät. Wenn Sie in der benutzerdefinierten Funktion Systemfunktionen verwenden, die vom gewählten Bediengerät nicht unterstützt werden, erhalten Sie im Ausgabefenster eine Fehlermeldung.

Siehe auch

Funktionsliste projektieren (Seite 1440)

Benutzerdefinierte VB-Funktion erstellen (Seite 1458)

Benutzerdefinierte VB-Funktion ausführen (Seite 1460)

Benutzerdefinierte C-Funktion ausführen (Seite 1473)

9.3.3 Aufruf von benutzerdefinierten Funktionen

Einleitung

In einer benutzerdefinierten Funktion können Sie benutzerdefinierte Funktionen der gleichen Programmiersprache einfügen, z. B. in einer benutzerdefinierten C-Funktion nur benutzerdefinierte C-Funktionen

Die Syntax ist abhängig von der verwendeten Programmiersprache. Welche Programmiersprache unterstützt wird, ist abhängig vom gewählten Bediengerät.

Um in einer benutzerdefinierten Funktion eine benutzerdefinierte Funktion aufzurufen, haben Sie folgende Möglichkeiten:

- Mit <Strg+Leer> oder mit
- Direkte Eingabe
- Über die Palette "Funktionsliste"

Mit <Strg+Leer> oder mit

Öffnen Sie mit <Strg+Leer> oder mit die Auswahlliste und wählen Sie die gewünschte benutzerdefinierte Funktion aus.

Direkte Eingabe

Sie geben die benutzerdefinierte Funktion direkt in den Code ein. Referenzen auf Objekte z. B. Bilder, Verbindungen und Archive übergeben Sie in Anführungszeichen.

Aufruf von benutzerdefinierten Funktionen ohne Rückgabewert in VBS

```
Average 4,10
```

Aufruf von benutzerdefinierten Funktionen mit Rückgabewert in VBS

```
Dim ValueA
ValueA = Average (4,10)
```

Wenn Sie den Rückgabewert nicht auswerten wollen, verwenden Sie den Aufruf wie für eine benutzerdefinierte Funktion ohne Rückgabewert.

Aufruf von benutzerdefinierten Funktionen ohne Rückgabewert in C

```
Average(4,10);
```

Aufruf von benutzerdefinierten Funktionen mit Rückgabewert in C

```
SetTagDouble("AverageValue",Average (4, 10));
```

Wenn Sie den Rückgabewert nicht auswerten wollen, verwenden Sie den Aufruf wie für eine benutzerdefinierte Funktion ohne Rückgabewert.

Über die Palette "Funktionsliste"

Wählen Sie in der Palette "Funktionsliste" die gewünschte benutzerdefinierte Funktion aus einer Auswahlliste.

Verfahren Sie wie bei der Projektierung einer Funktionsliste. Nähere Hinweise hierzu finden Sie unter "Funktionsliste projektieren (Seite 1440)".

Um die Liste in den Code zu übernehmen, klicken Sie auf die Schaltfläche "Übernehmen". Die Liste wird automatisch in die korrekte Syntax umgewandelt.

Siehe auch

Benutzerdefinierte VB-Funktion erstellen (Seite 1458)

Benutzerdefinierte VB-Funktion ausführen (Seite 1460)

Benutzerdefinierte C-Funktion ausführen (Seite 1473)

Funktionsliste projektieren (Seite 1440)

9.3.4 Zugriff auf HMI-Variablen

Einleitung

Mit VBS oder C können Sie auf die HMI-Variablen zugreifen, die Sie im Projekt angelegt haben. In Runtime ändern oder lesen Sie den Wert einer HMI-Variablen mit einer benutzerdefinierten Funktion.

Darüber hinaus können Sie lokale Variablen als Zähler oder als Zwischenspeicher anlegen. Um Fehler durch falsche Schreibweisen von lokalen Variablen zu verhindern, muss jede lokale Variable initialisiert werden.

HMI-Variablen

- WinCC Runtime Advanced und Panels
  In der benutzerdefinierten Funktion wird auf den Variablenwert zugegriffen, der im Variablenabbild abgelegt ist. Danach wird der Variablenwert mit der eingestellten Zykluszeit aktualisiert. Die benutzerdefinierte Funktion greift zunächst auf Variablenwerten zu, die zum vorherigen Zykluszeitpunkt gelesen wurden.
  Wenn der Variablenname im Projekt den VBS-Namenskonventionen entspricht, können Sie die Variable in der benutzerdefinierten Funktion folgendermaßen verwenden.
  ```
  'VBS_Example_03
   If BeltDriveOilTemperature > 100 Then [Anweisung]
  ```
  Wenn der Variablenname im Projekt nicht den VBS-Namenskonventionen entspricht, müssen Sie die Variable über die "SmartTags-Auflistung" referenzieren. Im folgenden Beispiel enthält der Variablenname das &-Zeichen, das nach den VBS-Namenskonventionen nicht erlaubt ist:
  ```
  'VBS_Example_04
  SmartTags("Test&Trial")= 2005
  ```
- WinCC Runtime Professional
  Sie haben die gleichen Zugriffsmöglichkeiten wie bei den Bediengeräten mit WinCC Runtime Advanced und Panels. Zusätzlich können Sie asynchron auf die Variablenwerte zugreifen. Beim asynchronen Zugriff wird direkt auf den Variablenwert in der Steuerung zugegriffen. In einem lokalen Skript oder einer benutzerdefinierten C-Funktion greifen Sie wie folgt auf den Variablenwert zu:
  - SmartTags-Auflistung (VBS)
  - Tags-Auflistung mit Read-/Write-Methode (VBS)
  - GetTag-Funktion (C)
  - SetTag-Funktion (C)

  Der Variablenwert wird beim Aufruf der Methode oder benutzerdefinierten Funktion einmalig aus der Steuerung gelesen. Im Gegensatz zum Lesen aus dem Variablenabbild wird der Variablenwert nicht regelmäßig aktualisiert.

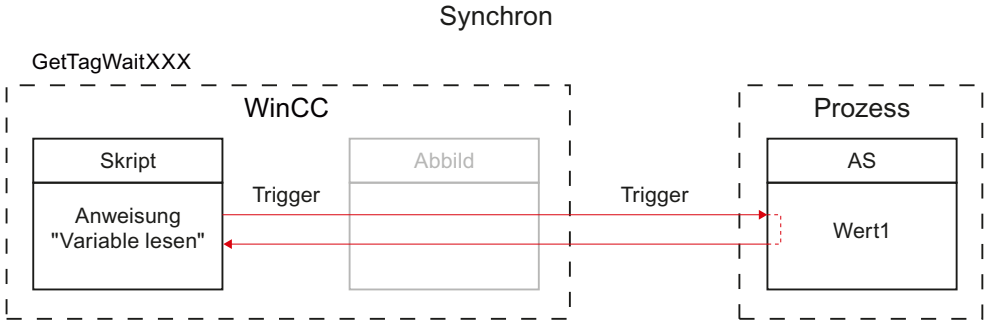

Bild 9-1 Synchron: Variablenwert direkt aus der Steuerung

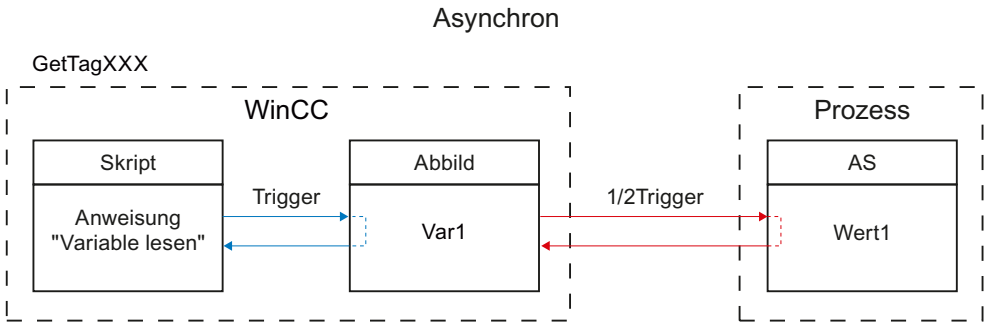

Bild 9-2 Asynchron: Variablenwert aus dem Variablenabbild

Hinweis

Runtime Scripting

Der Index von PLC-Arrayelementen kann mit einer beliebigen Zahl beginnen. In WinCC beginnt die Indexierung immer mit 0.

Eine PLC-Variable "Array[1..3] of Int" wird in WinCC z. B. mit "Array[0..2] of Int" abgebildet.

Wenn Sie in einem Skript auf ein Array zugreifen, achten Sie auf die korrekte Indexierung.

Lokale Variablen

Lokale Variablen können Sie nur innerhalb von benutzerdefinierten Funktionen oder lokalen Skripten verwenden.

Lokale Variablen verwenden Sie für Folgendes:

- Zwischenspeicher
- Zähler, z. B. For-Anweisung

Initialisieren Sie jede Variable entsprechend der Syntax der Programmiersprache.

Beispiel:. Dim objTagFillLevel

Initialisieren Sie jede Variable mit einem Wert, um Fehler durch leere HMI-Variablen zu vermeiden. Um den Wert der lokalen Variablen außerhalb zu nutzen, verknüpfen Sie diese mit einer internen oder externen HMI-Variable.

Zugriff auf HMI-Variablen mit dynamischem Namen (WinCC Runtime Advanced und Panels)

Die benutzerdefinierte VB-Funktion greift über den Variablennamen auf den Variablenwert zu. Sie können den Variablennamen so angeben, dass der Variablennamen zur Laufzeit der benutzerdefinierten Funktion zusammengesetzt wird.

Wenn Sie die benutzerdefinierte VB-Funktion nur in einem Bild aufrufen, in dem die HMI-Variable auch an anderer Stelle, z.B. an einem EA-Feld, verwendet wird, projektieren Sie aus Performancegründen die HMI-Variable mit der Erfassungsart "Zyklisch im Betrieb".

Wenn die benutzerdefinierte VB-Funktion aufgerufen wird und die HMI-Variable nicht in dem aktuell angezeigten Bild verwenden wird, projektieren Sie die HMI-Variable mit der Erfassungsart "Zyklisch fortlaufend". Dadurch ist sichergestellt, dass zu jedem Zeitpunkt der aktuelle Werte der Variable zur Verfügung steht.

9.3.5 Syntax von benutzerdefinierten Funktionen testen

Einleitung

Während der Programmierung wird der Code im Hintergrund geprüft. Syntaktische Fehler werden mit einer roten Wellenlinie gekennzeichnet. Dabei wird die benutzerdefinierte Funktion auf korrekte Syntax und auf gültige Objektreferenzierungen geprüft.

Um alle Fehlerstellen im Code zu identifizieren und die dazugehörigen Fehlermeldungen auszugeben, prüfen Sie im Editor "Skripte" die Syntax. Dabei werden Fehlermeldungen ausgegeben, die vom Skript-Parser erzeugt werden.

Um benutzerdefinierte VB-Funktionen auf logische Programmierfehler zu überprüfen, verwenden Sie einen separaten Debugger.

Fehlertypen

Folgende Fehlertypen werden bei der Syntaxprüfung erkannt und im "Ausgabefenster" ausgegeben:

- Zeichenfolge nicht bekannt (z. B. kein Schlüsselwort)
- Wertzuweisung an eine Konstante
- Der Name einer mit der Dim-Anweisung angelegten lokalen Variablen ist bereits bekannt, z. B.: als Parameter, Objekt oder Bestandteil des Objektmodells
- Syntaxfehler

Voraussetzungen

- Benutzerdefinierte Funktion ist geöffnet.

Vorgehen

Um die Syntax zu prüfen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf , um die Syntaxprüfung zu starten.
   Die Syntax wird überprüft. Die Ergebnisse der Syntaxprüfung werden im Inspektorfenster unter "Info > Übersetzen" angezeigt. Syntaxfehler werden mit der Zeilennummer ausgegeben.
2. Korrigieren Sie bei Bedarf die Fehler in der benutzerdefinierten Funktion.

Besonderheit bei der Syntaxprüfung von benutzerdefinierten Funktionen

Benutzerdefinierte Funktionen werden erst in Runtime interpretiert. Wenn ein Bediengerät übersetzt wird, werden dessen benutzerdefinierte Funktionen vom System auf ihre syntaktische Korrektheit überprüft. Unter bestimmten Umständen können jedoch Laufzeitfehler auftreten.

9.4 Erstellen von benutzerdefinierten VB-Funktionen

9.4.1 Zugriff auf Objekte mit VBS

Einleitung

In benutzerdefinierten VB-Funktionen stehen Ihnen die Objekte des VBS Objektmodells mit den dazugehörigen Eigenschaften und Methoden zur Verfügung.

In Runtime können Sie die Objekteigenschaften lesen und ändern.

Referenzierung von Objekten

In benutzerdefinierten VB-Funktionen referenzieren Sie Objekte über die dazugehörende Auflistung. Zur Identifizierung des Objekts verwenden Sie dessen Namen oder die Positionsnummer innerhalb der Auflistung.

Wenn Sie häufiger auf Eigenschaften eines Objekts zugreifen, erzeugen Sie eine Objektreferenz. Auf die Objekteigenschaften haben Sie mit und ohne Objektreferenz Zugriff.

Mit der folgenden Anweisung wird das erste Objekt im Bild "MainScreen" referenziert:

```
'VBS_Example_01
Dim objObject
'Change to Screen "MainScreen"
HMIRuntime.BaseScreenName = "MainScreen"
Set objObject = HMIRuntime.Screens(1).ScreenItems(1)
```

Mit der folgenden Anweisung wird ein Objekt über seinen Namen referenziert und eine Objekteigenschaft geändert. Dazu müssen Sie das Objekt mit diesem Namen im Bild angelegt haben.

```
'VBS_Example_02
Dim objCircle
HMIRuntime.BaseScreenName = "MainScreen"
Set objCircle = HMIRuntime.Screens(1).ScreenItems("Circle_01")
objCircle.BackColor = vbGreen
```

9.4.2 Verwendung von VB-Variablen

Definition VB-Variablen

VB-Variablen deklarieren Sie, indem Sie Definitionszeilen vor dem Funktionsnamen einer benutzerdefinierten VB-Funktion einfügen.

```
Dim objScreen
Dim objScrItem
```

Der Gültigkeitsbereich einer VB-Variablen hängt davon ab, wo sie deklariert wurde.

Benutzerdefinierten VB-Funktionen

Im Editor "Skripte" sind Zeilen 1 und 2 standardmäßig der Deklarationsbereich. Sie können beliebige Zeilen hinzufügen.

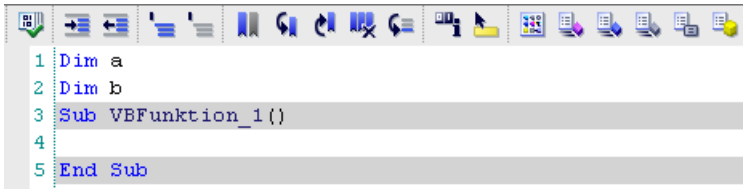

Jede so definierte VB-Variable können Sie in benutzerdefinierten VB-Funktionen verwenden. Wenn Sie im Editor "Skripte" die Syntax überprüfen, erhalten Sie eine Meldung, dass die VB-Variable nicht bekannt ist.

Die VB-Variable ist erst in Runtime bekannt. Sie wird bereits beim Start von Runtime angelegt, auch wenn die benutzerdefinierte VB-Funktion selbst nicht aufgerufen wird.

Um Inkonsistenzen zu vermeiden, müssen die VB-Variablen innerhalb der benutzerdefinierten VB-Funktionen eindeutig sein.

Lokale VB-Skripte

Der Deklarationsbereich eines lokalen Skrpits gehört zum Bild. Auf die darin deklarierten Variablen kann von jedem Skript innerhalb des Bildes zugegriffen werden. Wenn Sie ein Bildobjekt mit einem lokalen Skript in ein anderes Bild oder in die Bibliothek kopieren, wird auch das Skript, aber nicht der Deklarationsbereich kopiert.

Im Editor für lokale VB-Skripte können Sie den Deklarationsbereich ein- und ausblenden. Nähere Hinweise hierzu finden Sie unter "Editor für lokale Skripte (Seite 1482)".

- Lokale VB-Skripte mit azyklischem Trigger
  Jede so definierte VB-Variable wird automatisch in den Deklarationsbereich jedes lokalen Skripts mit azyklischem Trigger übernommen. Bei lokalen Skripten mit zyklischem Trigger bleibt der Deklarationsbereich leer.
- Lokale VB-Skripte mit zyklischem Trigger
  Jede so definierte VB-Variable wird automatisch in den Deklarationsbereich jedes lokalen VB-Skripts mit zyklischem Trigger übernommen. Bei lokalen Skripten mit azyklischem Trigger bleibt der Deklarationsbereich leer.

Beispiel

An die Eigenschaft "Farbe Hintergrund" wird ein lokales VB-Skript projektiert. Das lokale VB-Skript besitzt einen zyklischen Trigger.

Im Deklarationsbereich werden die VB-Variablen a und b deklariert.

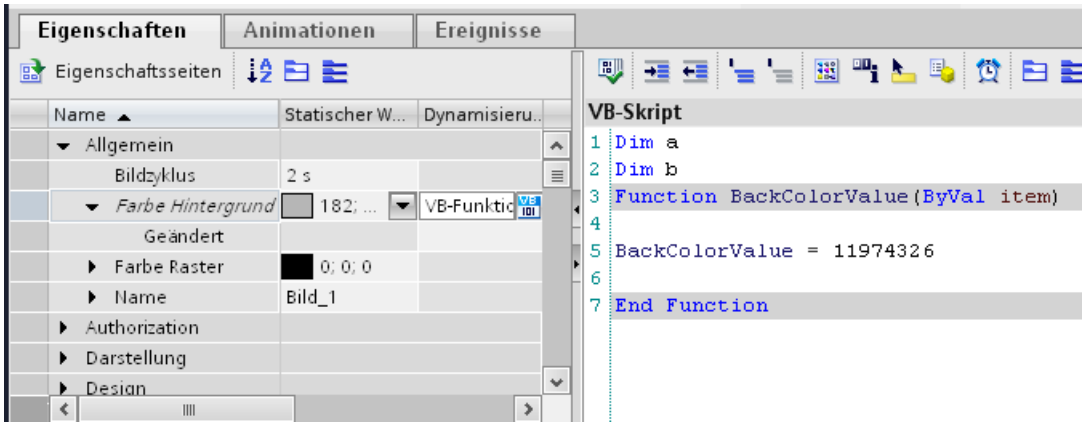

Wenn Sie an das Ereignis "Geändert" ein lokales VB-Skript projektieren, besitzt dieses VB-Skript einen azyklischen Trigger. Der Deklarationsbereich ist leer.

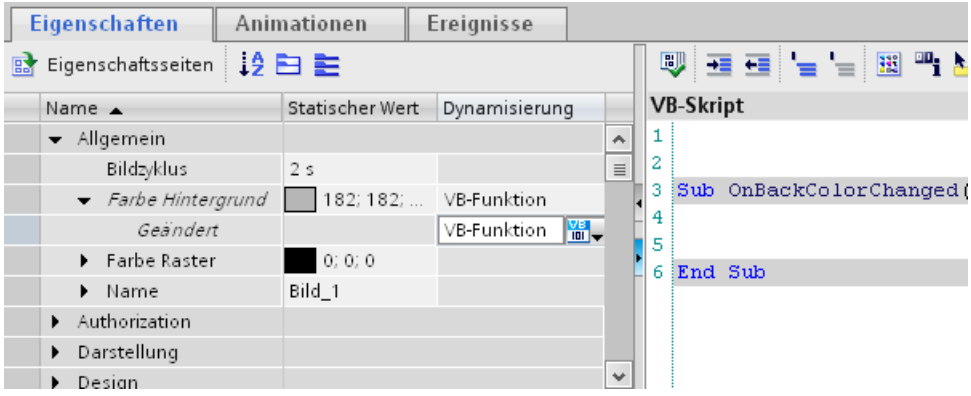

Wenn Sie an eine andere Eigenschaft ein lokales VB-Skript mit zyklischem Trigger projektieren, sind im Deklarationsbereich die VB-Variablen a und b bereits deklariert.

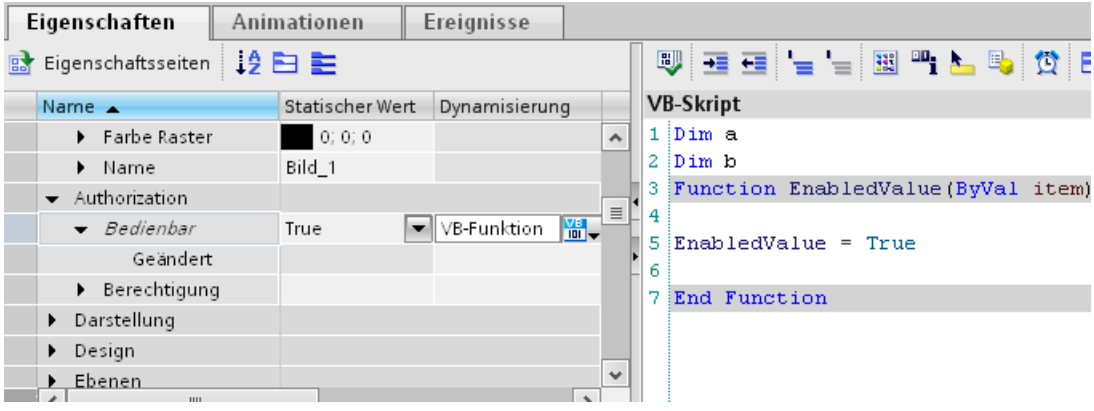

Siehe auch

Editor für lokale Skripte (Seite 1482)

9.4.3 Benutzerdefinierte VB-Funktion erstellen

Einleitung

Wenn Sie eine benutzerdefinierte VB-Funktion erstellen, legen Sie folgende Einstellungen fest:

- Den Namen, mit dem Sie die benutzerdefinierte VB-Funktion aufrufen.
- Den Typ der benutzerdefinierten VB-Funktion.
- Die Parameter, die der benutzerdefinierten VB-Funktion in Runtime übergeben werden.

Verwenden Sie als Namen ausschließlich folgende Zeichen:

- "A" bis "Z"
- "a" bis "z"
- "0" bis "9"
- "_" als Trennzeichen

Hinweis

Wenn Sie die Parameter oder den Funktionstyp ändern, wird an der Verwendungsstelle die Änderung farblich gekennzeichnet, z. B. in der Funktionsliste.

Vorgehen

Um eine benutzerdefinierte VB-Funktion zu erstellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation "Skripte"
2. Doppelklicken Sie in der Projektnavigation unter "VB-Skripte" auf "Neue VB-Funktion hinzufügen"

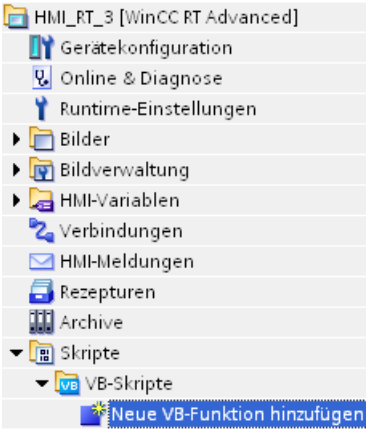

Die benutzerdefinierte VB-Funktion wird im Arbeitsbereich als neue Registerkarte angelegt. Im Inspektorfenster öffnet sich die Eingabemaske für die Konfigurationseinstellungen der benutzerdefinierten VB-Funktion.

3. Konfigurieren Sie im Inspektorfenster "Eigenschaften > Allgemein > Allgemein" die benutzerdefinierte VB-Funktion.

4. Geben Sie einen aussagekräftigen "Namen" für die benutzerdefinierte VB-Funktion ein.
5. Wählen Sie den "Typ" der benutzerdefinierten VB-Funktion.
   – "Function" besitzen einen Rückgabewert.
   – "Sub" besitzen keinen Rückgabewert.
6. Um Parameter hinzuzufügen, klicken Sie in der Liste "Parameter" auf "<Hinzufügen>". Geben Sie den Parameternamen ein und legen Sie den Parametertyp fest.
   Nähere Hinweise hierzu finden Sie in "Übergabe und Rückgabe von Werten in VBS (Seite 1462)".
   – "ByVal" übergibt den Parameterwert.
   – "ByRef" übergibt die Adresse des Parameters.
7. Geben Sie im Arbeitsbereich ihren Code ein.
   Nähere Hinweise hierzu finden Sie unter Editor "Skripte" (Seite 1444)".

Ergebnis

Die benutzerdefinierte VB-Funktion ist angelegt. In der Titelzeile werden der Typ, der Name und die Funktionsparameter angezeigt.

Anfang und Ende werden beim Anlegen der benutzerdefinierten Funktion automatisch eingefügt.

Wenn Sie benutzerdefinierte VB-Funktionen durch ein Kennwort schützen, werden diese verschlüsselt im Projekt gespeichert. Nähere Hinweise hierzu finden Sie unter "Schützen von benutzerdefinierten Funktionen (Seite 1477)".

Das folgende Bild zeigt eine benutzerdefinierte VB-Funktion, welche die Temperatur von "Fahrenheit" nach "Grad Celsius" umrechnet:

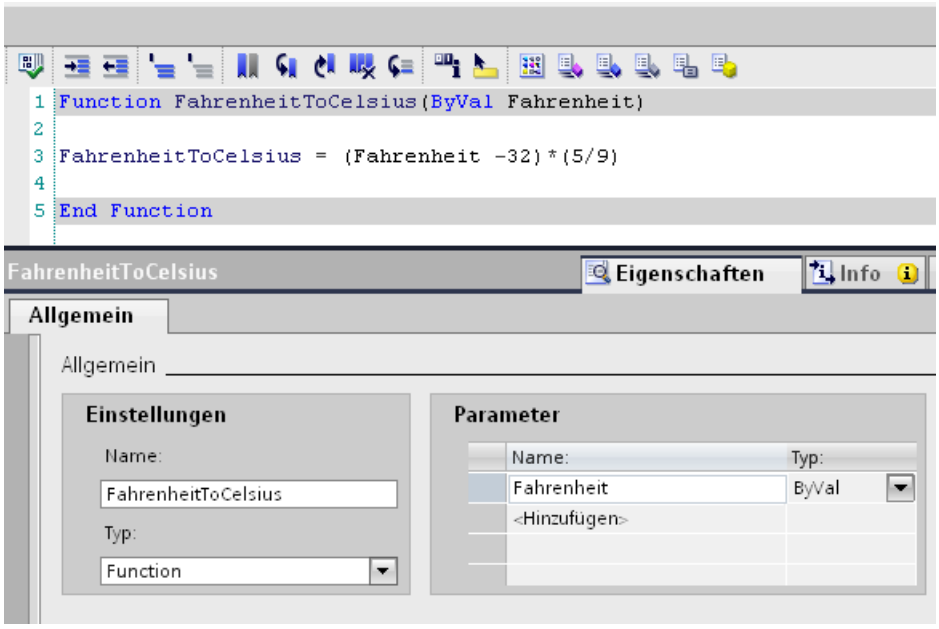

Siehe auch

Übergabe und Rückgabe von Werten in VBS (Seite 1462)

Editor "Skripte" (Seite 1444)

Schützen von benutzerdefinierten Funktionen (Seite 1477)

Aufruf von Systemfunktionen (Seite 1447)

Aufruf von benutzerdefinierten Funktionen (Seite 1449)

9.4.4 Benutzerdefinierte VB-Funktion umbenennen

Benutzerdefinierte VB-Funktion in der Projektnavigation umbenennen

Um eine benutzerdefinierte VB-Funktion in der Projektnavigation umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die gewünschte benutzerdefinierte VB-Funktion.
2. Wählen Sie im Kontextmenü der benutzerdefinierten VB-Funktion den Befehl "Umbenennen (F2)".
3. Geben Sie einen neuen Namen für die benutzerdefinierte VB-Funktion ein.

Benutzerdefinierte VB-Funktion im Inspektorfenster umbenennen

Um eine benutzerdefinierte VB-Funktion im Inspektorfenster umzubenennen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die benutzerdefinierte VB-Funktion im Editor "Skripte".
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
3. Geben Sie den neuen Funktionsnamen ein.

Ergebnis

Die benutzerdefinierte VB-Funktion ist umbenannt. Die Verwendungsstellen werden automatisch angepasst.

9.4.5 Benutzerdefinierte VB-Funktion ausführen

Einleitung

In WinCC gibt es folgende Möglichkeiten, eine benutzerdefinierte VB-Funktion auszuführen:

- Funktionsliste
- Benutzerdefinierte VB-Funktion
- Lokales VB-Skript (nur bei WinCC Runtime Professional)

Benutzerdefinierte VB-Funktion in einer Funktionsliste aufrufen

1. Fügen Sie die benutzerdefinierte VB-Funktion wie eine Systemfunktion zu einer Funktionsliste hinzu.
   Sie finden die benutzerdefinierten VB-Funktionen in der Auswahlliste unter "VB-Funktionen".
   Nähere Hinweise hierzu finden Sie in "Funktionsliste projektieren".
2. Wenn die benutzerdefinierte VB-Funktion Parameter besitzt, versorgen Sie die Parameter mit statischen Werten oder HMI-Variablen.
3. Wenn die benutzerdefinierte VB-Funktion einen Rückgabewert besitzt, wählen Sie eine HMI-Variable aus. An die HMI-Variable wird der zu verarbeitende Wert übergeben.

Hinweis

In einer Funktionsliste können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache verwenden. Mit der Auswahl der ersten benutzerdefinierten Funktion legen Sie die Programmiersprache fest.

Benutzerdefinierte VB-Funktion in einer benutzerdefinierten VB-Funktion aufrufen

1. Öffnen Sie die benutzerdefinierte VB-Funktion, in der Sie die benutzerdefinierte VB-Funktion aufrufen wollen.
2. Rufen Sie die benutzerdefinierte VB-Funktion in der Syntax der Programmiersprache auf. Nähere Hinweise hierzu finden Sie in "Aufruf von benutzerdefinierten Funktionen".
3. Wenn die benutzerdefinierte VB-Funktion einen Rückgabewert besitzt, weisen Sie die VB-Funktion einem Ausdruck zu, z. B. eine lokale Variable.

Hinweis

In einer benutzerdefinierten VB-Funktion können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache aufrufen.

Benutzerdefinierte VB-Funktion in einem lokalen VB-Skript aufrufen

1. Die Funktionsliste ist in ein lokales VB-Skript umgewandelt.
2. Rufen Sie die benutzerdefinierte VB-Funktion in der Syntax der Programmiersprache auf. Nähere Hinweise hierzu finden Sie in "Aufruf von benutzerdefinierten Funktionen".
3. Wenn die benutzerdefinierte VB-Funktion einen Rückgabewert besitzt, weisen Sie die VB-Funktion einem Ausdruck zu, z. B. eine lokale Variable.

Hinweis

In einem lokalen VB-Skript können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache aufrufen.

Siehe auch

Aufruf von Systemfunktionen (Seite 1447)

Aufruf von benutzerdefinierten Funktionen (Seite 1449)

Funktionsliste projektieren (Seite 1440)

9.4.6 Übergabe und Rückgabe von Werten in VBS

Übergabe eines Werts

Folgende Möglichkeiten gibt es bei der Übergabe von Parametern.

- "Call by Value" - ByVal
  ByVal übergibt den Parameterwert. Wenn Sie als Parameter eine Variable übergegeben, wird beim Ausführen der benutzerdefinierten Funktion der Variablenwert an die benutzerdefinierte Funktion übergeben.
- "Call by Reference" - ByRef
  ByRef übergibt die Adresse des Parameters. Wenn Sie als Parameter eine Variable übergegeben, wird beim Ausführen der benutzerdefinierten Funktion die Adresse der Variablen an die benutzerdefinierte Funktion übergeben.
  Beim Aufruf von benutzerdefinierten Funktionen und Systemfunktionen in benutzerdefinierten Funktionen wird der Parameter als ByRef übergeben.

Rückgabe eines Werts

Rückgabewerte können das Ergebnis einer Berechnung (z. B. Mittelwert von zwei Zahlen) zurückliefern. Ein Rückgabewert kann aber auch Aufschluss darüber geben, ob eine Anweisung korrekt ausgeführt wurde.

Deshalb besitzen auch solche Systemfunktionen einen Rückgabewert, die Dateioperationen wie Löschen ausführen.

Damit eine benutzerdefinierte Funktion einen Wert zurückgeben kann, müssen Sie den Typ "Function" einstellen. In der benutzerdefinierten Funktion weisen Sie dem Rückgabewert den Funktionsnamen zu:

```
1 Function Average( Value1 , Value2 )
2 'Check if Parameters are not numeric:
3 If IsNumeric (Value1) = False Then Value1 = 1
4 If IsNumeric (Value2) = False Then Value2 = 1
5 Average = (Value1+Value2)/2
6
7 End Function
```

Um einen Mittelwert aus zwei Zahlen zu bilden, rufen Sie die benutzerdefinierte VB-Funktion Average auf und übergeben das Ergebnis z. B. an die HMI-Variable AverageValue.

- In einer benutzerdefinierten VB-Funktion

```
SmartTags("AverageValue") = Average ("4", "10")
```

- In der Funktionsliste

Average	
Rückgabewert	AverageValue
Value1	10
Value2	4

Den Wert der HMI-Variablen AverageValue können Sie in einem E/A-Feld ausgeben.

Siehe auch

Benutzerdefinierte VB-Funktion erstellen (Seite 1458)

Beispiel: Umrechnen von Fahrenheit in Grad Celsius (Seite 1505)

Beispiel: Umrechnen von Inch in Meter (Seite 1507)

9.5 Erstellen von benutzerdefinierten C-Funktionen

9.5.1 Zugriff auf Objekte mit C

Einleitung

Mit folgenden Systemfunktionen greifen Sie auf die Eigenschaften von Objekten zu:

- "GetProp": Systemfunktionen, die mit "GetProp" beginnen, lesen einen Wert.
- "SetProp": Systemfunktionen, die mit "SetProp" beginnen, schreiben einen Wert.

Für jeden Datentyp gibt es eine separate Systemfunktion mit GetProp und SetProp.

Objekteigenschaft adressieren

Mit <Strg+J> öffnen Sie bei der Parametervergabe die Objektliste, aus der Sie die verfügbaren Objekte und Eigenschaften auswählen. Die Datentypen der übergebenen Parameter werden bei der Syntaxprüfung auf Plausibilität geprüft.

Objekteigenschaft schreiben

Das folgende Beispiel zeigt, wie ein Kreis in Ebene "4" verschoben wird:

```
SetPropLong("Screen_1", "Circle_1", "Layer", 4);
```

Siehe auch

Beispiel für das Zugreifen auf Objekte im Editor Bilder (Seite 1531)

9.5.2 C-Header erstellen

Einleitung

Verwenden Sie die Header-Datei "GlobalDefinitions.h", um dort Funktionen, Variablen und Konstanten zu definieren, die Sie in allen lokalen C-Skripten und benutzerdefinierten C-Funktionen benötigen. Auf diese Weise bleiben lokale C-Skripte und benutzerdefinierte C-Funktionen schlank und Sie vermeiden Redundanz. Änderungen führen Sie an zentraler Stelle in der Header-Datei aus.

Benutzerdefinierte C-Funktionen, die Sie im Projekt definiert haben, sind automatisch verfügbar. Diese C-Funktionen brauchen Sie nicht in der "GlobalDefinitions.h" definieren.

Die C-Bibliotheken wie "Math" sind standardmäßig referenziert.

Darüber hinaus können Sie:

- zusätzliche C-Header-Dateien anlegen.
- externen C-Header-Dateien einbinden.

Zusätzliche C-Header-Datei anlegen

1. Doppelklicken Sie in der Projektnavigation unter "Skripte >C-Header" auf "Neuen C-Header hinzufügen".
2. Geben Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" einen "Namen" für den C-Header ein.
3. Fügen Sie im Arbeitsbereich die benötigten Deklarationen von Variablen und Konstanten ein.

C-Header-Datei in einer C-Funktion zu nutzen

1. Um die Deklarationen der C-Header-Datei in einer C-Funktion zu nutzen, fügen Sie den C-Header im globalen Bereich der C-Funktion ein. C-Header wird mit einer #include-Anweisung eingebunden. Um Fehler beim Variablenzugriff zu vermeiden, dürfen C-Header-Dateien nicht mehrfach eingelesen werden. Durch das folgende Konstrukt wird z. B. das mehrfache Lesen von Header-Dateien verhindert.
```
#ifndef _myHeader_h
#define _myHeader_h
#include "myheader.h"
#endif
```

Include-Pfad für externe C-Header-Dateien festlegen

Um externe C-Header-Dateien zu verwenden, müssen Sie den Include-Pfad festlegen. Gehen Sie folgendermaßen vor:

1. Selektieren Sie das gewünschte Bediengerät.
2. Wählen Sie den Menübefehl "Extras >Einstellungen".
3. Wählen Sie in der Bereichsnavigation unter "Visualisierung > Runtime Scripting" den Pfad für den zusätzlichen C-Header aus.
4. Schließen Sie WinCC und starten sie es neu.
5. Wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software (komplett übersetzen)".

Hinweis

Kopieren eines Projekts in dem eine externe C-Header-Datei verwendet wird

Die Pfadangabe für die externe Header-Datei ist nicht im Projekt gespeichert und nur auf dem Computer gültig, auf dem sie gesetzt wurde. Wenn Sie dieses Projekt auf einen anderen Rechner kopieren, müssen sie den Include-Pfad für externe C-Header-Dateien im kopierten Projekt neu zuweisen.

9.5.3 Verwendung globaler C-Variablen

Definition globaler C-Variablen

Eine globale C-Variable definieren Sie, indem Sie die Definitionszeile vor den Funktionsnamen einer Funktion einfügen:

```
int global_a; //The tag global_a is defined as integer
void myfunction()
{
//insert the code starting here
}
```

Verwendung globaler C-Variablen

Globale C-Variablen verwenden Sie in benutzerdefinierten C-Funktionen oder lokalen C-Skripten, indem Sie die C-Variablen innerhalb der Header-Datei "GlobalDefinitions.h" als extern deklarieren:

GlobalDefinitions.h

```
extern int global_a;
```

Benutzerdefinierte C-Funktion

```
// Insert the header starting here
#include "GlobalDefinitions.h"
void myFunction()
{
//insert the code starting here
printf ("C2: %i\r\n", ++ global_a);
}
```

Dem Compiler wird damit gesagt, dass er die Variable global_a nicht selbst anlegen muss, sondern dass diese in Runtime an anderer Stelle angelegt wird.

Wird nun der Wert der Variablen global_a verändert, steht der geänderte Wert in allen benutzerdefinierten C-Funktionen und lokalen C-Skripten zur Verfügung.

Sie können globale C-Variablen auch direkt innerhalb einer Funktion deklarieren. Aus Gründen der Übersichtlichkeit und zur Vermeidung von Doppeldefinitionen empfiehlt es sich jedoch, globale C-Variablen nur an einer Stelle zu deklarieren.

Hinweis

Einer benutzerdefinierten Funktion und den mit ihr definierten globalen C-Variablen stehen maximal 64 KByte Speicherplatz zur Verfügung.

Gültigkeitsbereich

Jede in GlobalDefinitions.h als extern deklarierte Variable ist in Runtime in allen benutzerdefinierten C-Funktionen und lokalen C-Skripten bekannt. Sie wird bereits beim Start von Runtime angelegt, auch wenn die benutzerdefinierte Funktion selbst nicht aufgerufen wurde.

Hinweis

Im WinCC ServiceMode können zwischen benutzerdefinierten C-Funktionen im Aufgabenplaner und benutzerdefinierten Funktionen im Bild-Editor keine globalen C-Variablen ausgetauscht werden.

Wenn Sie die Variable über den WebNavigator ansprechen wollen, müssen Sie die C-Funktion mit der global deklarierten C-Variablen aufrufen.

9.5.4 Funktionen aus DLLs in benutzerdefinierter C-Funktion aufrufen

Einleitung

In benutzerdefinierten C-Funktionen können Sie Funktionen aus DLLs verwenden. Damit machen Sie den Funktionsumfang von vorhandenen DLLs nutzbar und vermeiden Redundanz.

Hinweis

Release- und Debug-DLLs

WinCC wird als Release-Version ausgeliefert und verwendet damit die Release-Versionen von System-DLLs. Wenn Sie eigene DLLs in der Debug-Version verwenden, kann zusätzlich auch die Release-DLL geladen werden. Um erhöhte Systemlast zu vermeiden, verwenden Sie beim Einsatz eigener DLLs immer die Release-Version der DLL.

Voraussetzung

DLL liegt im Ordner "..\Common Files\Siemens\bin" von WinCC oder in einem Ordner, der in der Systemvariablen PATH definiert ist. Diese Variable wird über die Systemeigenschaften des Betriebssystems festgelegt.

Vorgehen

Um Funktionen aus DLLs aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die benutzerdefinierte C-Funktion, in der Sie Funktionen aus einer DLL aufrufen wollen.
2. Fügen Sie am Anfang der benutzerdefinierten C-Funktion folgenden Code ein:

```
#pragma code("<Name>.dll")
//Declare functions stored in DLL:
<Return value> <Function name 1>(...);
<Return value> <Function name 2>(...);
...
<Return value> <Function name n>(...);
#pragma code
```

Ergebnis

Die Funktionen <Funktionsname 1> ... <Funktionsname n> aus <Name.dll> sind deklariert. Sie können die Funktionen in der benutzerdefinierten C-Funktion aufrufen.

Beispiel

Das folgende Beispiel zeigt, wie Sie die Datei "kernel32.dll" einbinden und die Funktion "GetLocalTime" aufrufen:

```
#pragma code("kernel32.dll")
VOID GetLocalTime(LPSYSTEMTIME lpSystemTime);
#pragma code()

SYSTEMTIME st;

GetLocalTime(&st);
```

9.5.5 Codepage für C-Funktion einrichten

Einleitung

Sie können Ihr WinCC-Projekt in mehreren Sprachen verwenden. Sie legen für jedes C-Skript fest, welche Codepage welcher Sprache zum Übersetzen verwendet wird. Die Codepage wirkt sich auf die Anzeige aller Texte aus, die im C-Skript in Runtime ausgegeben werden.

Sie wählen beim Anlegen einer C-Funktion, ob Sie die Einstellungen des Bediengeräts wählen oder eine der projektierten Runtime-Sprachen.

In den Runtime-Einstellungen des Bediengeräts stehen Ihnen folgende Optionen zur Verfügung:

- Betriebssystemsprache für Nicht-Unicode-Programme
  Die C-Funktion wird im ES in Englisch übersetzt.
  Texte werden in Runtime in der Codepage dieser Sprache ausgegeben.
- Aktuell eingestellte WinCC Runtime Sprache
  Die C-Funktion wird ES in Englisch übersetzt.
  Texte werden in Runtime in der Codepage dieser Sprache ausgegeben.
- Projektierte Runtime Sprache
  Die C-Funktion wird im ES in dieser Sprache übersetzt.

Voraussetzung

- Im Editor "Projektsprachen" sind die gewünschten Sprachen aktiviert.
- Im Editor "Sprache & Schriftart" sind die gewünschten Runtime-Sprachen aktiviert.

Codepage für das Bediengerät wählen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart".
3. Stellen Sie unter „Sprach-Einstellung für Skript > Sprache der C-Skripte mit „Geräteeeinstellung verwenden“"das Auswahlfenster über ein:
   - Auswahl aller Runtime Sprachen
   - Auswahl aller anderen Optionen
4. Wählen Sie die gewünschte Codepage.

Ergebnis

Wenn in Runtime über eine C-Funktion Texte ausgegeben werden, wird die für das Skript eingestellte Codepage verwendet.

9.5.6 Benutzerdefinierte C-Funktion erstellen

Einleitung

Wenn Sie eine benutzerdefinierte C-Funktion erstellen, legen Sie folgende Einstellungen fest:

- Den Namen, mit dem Sie die benutzerdefinierte C-Funktion aufrufen.
- Den Typ der benutzerdefinierten C-Funktion.

- Die Parameter, die der benutzerdefinierten C-Funktion in Runtime übergeben werden.

  Hinweis

  Wenn Sie die Parameter oder den Funktionstyp ändern, wird an der Verwendungsstelle die Änderung farblich gekennzeichnet, z. B. in der Funktionsliste.

- Die Codepage für die C-Funktion.

Verwenden Sie als Namen ausschließlich folgende Zeichen:

- "A" bis "Z"
- "a" bis "z"
- "0" bis "9"
- "_" als Trennzeichen

Hinweis

Beim Namen gibt es keine Unterscheidung zwischen Groß- und Kleinschreibung. Es ist nicht möglich, zwei benutzerdefinierte C-Funktionen mit dem gleichen Namen aber unterschiedlicher Schreibweise zu erstellen.

Hinweis

Unzulässige Variablennamen

Folgende Bezeichnungen sind in Runtime bereits dem Datentyp "char*" zugewiesen und dürfen deshalb in C-Scripting nicht als interne Variablen verwendet werden:

- "Object"
- "Property"
- "Tag"
- "Picture"

Vorgehen

Um eine neue benutzerdefinierte C-Funktion zu erstellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation "Skripte"
2. Doppelklicken Sie in der Projektnavigation unter "C-Skripte" auf "Neue C-Funktion hinzufügen"

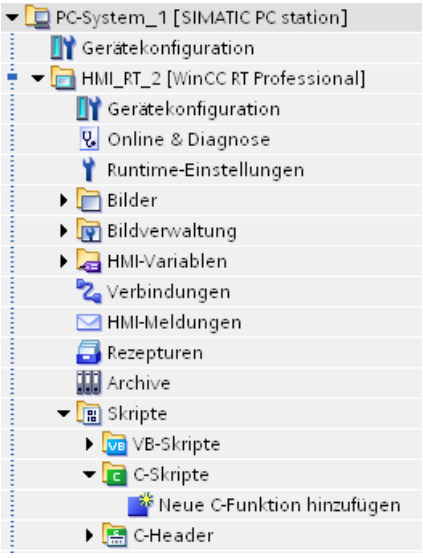

Die benutzerdefinierte C-Funktion wird im Arbeitsbereich als neue Registerkarte angelegt. Im Inspektorfenster öffnet sich die Eingabemaske für die Konfigurationseinstellungen der Funktion.

3. Wählen Sie unter "Codepage" die gewünschte Spracheinstellung.

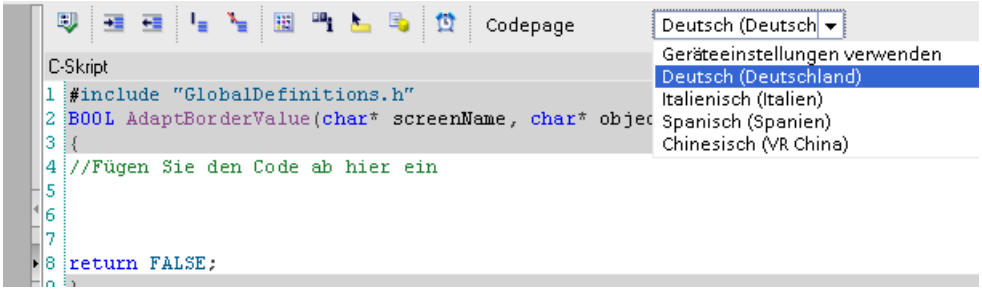

4. Konfigurieren Sie im Inspektorfenster "Eigenschaften > Allgemein > Allgemein" die Funktion.
5. Geben Sie einen aussagekräftigen "Namen" für die benutzerdefinierte C-Funktion ein. Wählen Sie den "Typ" der benutzerdefinierten C-Funktion. Nähere Hinweise hierzu finden Sie unter "Übergabe und Rückgabe von Werten in C".
   - Void: Funktion besitzt keinen Rückgabewert.
   - Variable: Rückgabewert wird als Variablenwert zurückgegeben.
   - Zeiger: Rückgabewert zeigt auf die Adresse der Variablen.

6. Um Parameter hinzuzufügen, doppelklicken Sie in der Liste "Parameter" "<Hinzufügen>". Geben Sie den Parameternamen ein. Nähere Hinweise hierzu finden Sie unter "Übergabe und Rückgabe von Werten in C".
   - Variable übergibt den Parameterwert.
   - Zeiger übergibt die Adresse des Parameters.
7. Schreiben Sie im Arbeitsbereich den Code.

Ergebnis

Die benutzerdefinierte C-Funktion ist angelegt. In der Titelzeile werden der Typ, der Name und die Funktionsparameter angezeigt.

Anfang und Ende werden beim Anlegen der benutzerdefinierten Funktion automatisch eingefügt.

Wenn Sie benutzerdefinierte C-Funktionen durch ein Kennwort schützen, werden diese verschlüsselt im Projekt gespeichert. Nähere Hinweise hierzu finden Sie unter "Schützen von benutzerdefinierten Funktionen (Seite 1477)".

Das folgende Bild zeigt eine benutzerdefinierte C-Funktion, welche die Temperatur von "Fahrenheit" nach "Grad Celsius" umrechnet:

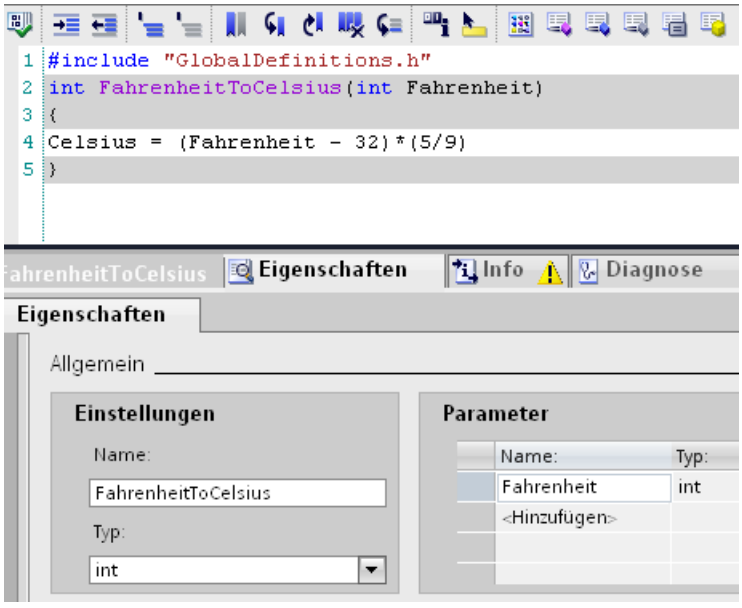

Siehe auch

Editor "Skripte" (Seite 1444)

Schützen von benutzerdefinierten Funktionen (Seite 1477)

9.5.7 Benutzerdefinierte C-Funktion umbenennen

Benutzerdefinierte C-Funktion in der Projektnavigation umbenennen

Um eine benutzerdefinierte C-Funktion in der Projektnavigation umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die benutzerdefinierte C-Funktion.
2. Wählen Sie im Kontextmenü der benutzerdefinierten C-Funktion den Befehl "Umbenennen (F2)".
3. Geben Sie einen neuen Namen für die benutzerdefinierte C-Funktion ein.

Benutzerdefinierte C-Funktion im Inspektorfenster umbenennen

Um eine benutzerdefinierte C-Funktion im Inspektorfenster umzubenennen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die benutzerdefinierte C-Funktion im Editor "Skripte".
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Allgemein > Allgemein".
3. Geben Sie den neuen Funktionsnamen ein.

Ergebnis

Die benutzerdefinierte C-Funktion ist umbenannt. Die Verwendungsstellen werden automatisch angepasst.

9.5.8 Benutzerdefinierte C-Funktion ausführen

Einleitung

In WinCC gibt es folgende Möglichkeiten, eine benutzerdefinierte C-Funktion auszuführen:

- Funktionsliste
- Benutzerdefinierte C-Funktion
- Lokales C-Skript

Benutzerdefinierte C-Funktion in einer Funktionsliste aufrufen

1. Fügen Sie die benutzerdefinierte C-Funktion wie eine Systemfunktion zu einer Funktionsliste hinzu.
   Sie finden die benutzerdefinierten C-Funktionen in der Auswahlliste unter "C-Funktionen". Nähere Hinweise hierzu finden Sie in "Funktionsliste projektieren".
2. Versorgen Sie die Parameter mit statischen Werten oder HMI-Variablen.
3. Wenn die benutzerdefinierte C-Funktion einen Rückgabewert besitzt, wählen Sie eine HMI-Variable aus. An die HMI-Variable wird der zu verarbeitende Wert übergeben.

Hinweis

In einer Funktionsliste können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache verwenden. Mit der Auswahl der ersten benutzerdefinierten Funktion legen Sie die Programmiersprache fest.

Benutzerdefinierte C-Funktion in einer benutzerdefinierten C-Funktion aufrufen

1. Öffnen Sie die benutzerdefinierte C-Funktion, in der Sie die benutzerdefinierte C-Funktion aufrufen wollen.
2. Rufen Sie die benutzerdefinierte C-Funktion in der Syntax der Programmiersprache auf. Nähere Hinweise hierzu finden Sie in "Aufruf von benutzerdefinierten Funktionen".
3. Wenn die benutzerdefinierte C-Funktion einen Rückgabewert besitzt, weisen Sie die Funktion einem Ausdruck zu, z. B. eine lokale Variable.

Hinweis

In einer benutzerdefinierten C-Funktion können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache aufrufen.

Benutzerdefinierte C-Funktion in einem lokalen C-Skript aufrufen

1. Die Funktionsliste ist in ein lokales C-Skript umgewandelt.
2. Rufen Sie die benutzerdefinierte C-Funktion in der Syntax der Programmiersprache auf. Nähere Hinweise hierzu finden Sie in "Aufruf von benutzerdefinierten Funktionen".
3. Wenn die benutzerdefinierte C-Funktion einen Rückgabewert besitzt, weisen Sie die C-Funktion einem Ausdruck zu, z. B. eine lokale Variable.

Hinweis

In einem lokalen C-Skript können Sie nur benutzerdefinierte Funktionen der gleichen Programmiersprache verwenden.

Siehe auch

Aufruf von Systemfunktionen (Seite 1447)

Aufruf von benutzerdefinierten Funktionen (Seite 1449)

Funktionsliste projektieren (Seite 1440)

9.5.9 Übergabe und Rückgabe von Werten in C

Übergabe eines Werts

Folgende Möglichkeiten gibt es bei der Übergabe von Parametern:

- Variable (Call by value)
  Um Werte in der Variablen zu speichern, wird der Variablennamen und der Datentyp festgelegt. Diese Variable wird aber nicht direkt übergeben, sondern es wird eine Kopie erstellt. Beim Ausführen der benutzerdefinierten C-Funktion wird der Variablenwert der Kopie an die benutzerdefinierte C-Funktion übergeben. Wenn Sie den Variablenwert ändern, ist die Änderung nur innerhalb der benutzerdefinierten C-Funktion gültig.
- Zeiger (Call by reference)
  Ein Zeiger ist eine Variable, die auf die Adresse einer Variablen zeigt. Zeiger werden wie Variablen definiert. Zusätzlich wird beim Datentyp das Zeichen * hinzugefügt, z. B. Double*. Die benutzerdefinierte C-Funktion greift auf die Variable zu, auf die der Zeiger zeigt. Wenn Sie den Variablenwert ändern, wird der geänderte Variablenwert in die Variable geschrieben.

Rückgabe eines Werts

Eine benutzerdefinierte C-Funktion kann einfache Anweisungen ausführen und nichts zurück geben. Der Rückgabewert ist dann vom Datentyp "void".

Wenn ein Rückgabewert benötigt wird, kann der Wert als Variable oder als Zeiger zurückgegeben werden.

Ein Rückgabewert kann aber auch Aufschluss darüber geben, ob eine Anweisung korrekt ausgeführt wurde. Deshalb besitzen auch solche Systemfunktionen einen Rückgabewert, die Dateioperationen wie das Schreiben in eine Datei ausführen.

Damit eine benutzerdefinierte C-Funktion einen Wert zurückgeben kann, müssen Sie z. B. den Typ "Double" einstellen. Mit der Anweisung return wird das Ergebnis an die benutzerdefinierte C-Funktion übergeben.

```
#include "GlobalDefinitions.h"
double Average(double Value1, double Value2)
{
double dbAverage;
dbAverage= (Value1+Value2)/2;
return dbAverage;
}
```

Um einen Mittelwert aus zwei Zahlen zu bilden, rufen Sie die benutzerdefinierte C-Funktion Average auf und übergeben das Ergebnis z. B. an die HMI-Variable AverageValue.

- In einer benutzerdefinierten C-Funktion

```
SetTagDouble("AverageValue",Average (4, 5));
```

- In der Funktionsliste

Average	
Rückgabewert	AverageValue
Value1	10
Value2	4

Den Wert der HMI-Variablen AverageValue können Sie in einem E/A-Feld ausgeben.

9.6 Schützen von benutzerdefinierten Funktionen

9.6.1 Know-how-Schutz einrichten

Vorgehen

Um einen Know-how-Schutz für benutzerdefinierte Funktionen einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die benutzerdefinierte Funktion ohne Know-how-Schutz, die Sie schützen möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Know-how-Schutz " wird geöffnet.
3. Klicken Sie auf "Definieren".
   Der Dialog "Passwort definieren" wird geöffnet.
4. Tragen Sie im Feld "Neu" ein Passwort ein.
5. Wiederholen Sie das Passwort im Feld "Bestätigen".
6. Bestätigen Sie die Eingabe mit "OK".
7. Schließen Sie den Dialog "Know-how-Schutz" mit "OK".

Ergebnis

Die benutzerdefinierte Funktion wird mit einem Know-how-Schutz versehen. In der Projektnavigation werden know-how-geschützte benutzerdefinierte Funktionen mit einem Schloss markiert. Das eingegebene Passwort ist für alle ausgewählten benutzerdefinierten Funktionen gültig.

9.6.2 Know-how-geschützte benutzerdefinierte Funktionen öffnen

Sie können mehrere know-how-geschützte benutzerdefinierte Funktionen nur dann gemeinsam öffnen, wenn sie mit dem gleichen Passwort geschützt werden.

Vorgehen

Um eine know-how-geschützte benutzerdefinierte Funktion zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie doppelt auf die benutzerdefinierte Funktion, die Sie öffnen möchten.
   Der Dialog "Zugriffsschutz" wird geöffnet.
2. Geben Sie das Passwort für die know-how-geschützte benutzerdefinierte Funktion ein.
3. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Wenn Sie das korrekte Passwort eingegeben haben, wird die know-how-geschützte benutzerdefinierte Funktion geöffnet. Der Know-how-Schutz wird jedoch nicht aufgehoben.

Nach dem Öffnen können Sie die benutzerdefinierte Funktion so lange bearbeiten, bis Sie die benutzerdefinierte Funktion oder das TIA Portal schließen. Beim nächsten Öffnen der benutzerdefinierten Funktion muss das Passwort wieder eingegeben werden. Wenn Sie den Dialog "Zugriffsschutz" mit "Abbrechen" oder "Schließen" schließen, wird die benutzerdefinierte Funktion zwar geöffnet, aber der Code nicht angezeigt. Sie können die benutzerdefinierte Funktion nicht bearbeiten.

9.6.3 Know-how-Schutz entfernen

Vorgehen

Um den Know-how-Schutz einer benutzerdefinierten Funktion zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die benutzerdefinierte Funktion, bei der Sie den Know-how-Schutz entfernen möchten.

   Hinweis

   Um für mehrere benutzerdefinierte Funktionen den Know-how-Schutz gleichzeitig zu entfernen, müssen die selektierten benutzerdefinierten Funktionen das gleiche Passwort haben.

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Know-how-Schutz" wird geöffnet.
3. Deaktivieren Sie das Optionskästchen "Code nicht anzeigen (Know-how-Schutz)".
4. Geben Sie das Passwort ein.
5. Bestätigen Sie die Eingabe mit "OK".

Ergebnis

Der Know-how-Schutz wird für die ausgewählte benutzerdefinierte Funktion aufgehoben.

9.6.4 Passwort ändern

Vorgehen

Um das Passwort zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die know-how-geschützte benutzerdefinierte Funktion, für die Sie das Passwort ändern möchten.

   Hinweis

   Um für mehrere benutzerdefinierte Funktionen das Passwort gleichzeitig zu ändern, müssen die selektierten benutzerdefinierten Funktionen das gleiche Passwort haben.

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".
   Der Dialog "Know-how-Schutz" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "Ändern".
4. Tragen Sie im Feld "Alt" das alte Passwort ein.
5. Tragen Sie im Feld "Neu" das neue Passwort ein.
6. Wiederholen Sie das neue Passwort im Feld "Bestätigen".
7. Bestätigen Sie die Eingabe mit "OK".
8. Schließen Sie den Dialog "Know-how-Schutz" mit "OK".

9.7 Erstellen von lokalen Skripten

9.7.1 Lokale Skripte

Definition

Lokale Skripte sind an ein einzelnes Objekt wie eine Objekteigenschaft gebunden und besitzen einen zyklischen oder azyklischen Trigger. Ein lokales Skript erstellen Sie direkt an der Verwendungsstelle:

- Aufgabenplaner
- Bilder
  - Ereignis (Umgewandelte Funktionsliste)
  - Objekteigenschaft (Eigenschaftsliste)
  - Eigenschaftsänderung (Eigenschaftsliste)
- Menüs und Symbolleisten
- Nur C-Skript: Protokolle
  - E/A-Feld an der Eigenschaft Prozesswert

Lokale Skripte werden nicht in der Projektnavigation angezeigt und sind nur an der Verwendungsstelle verfügbar.

Verwendung

Sie verwenden lokale Skripte für Folgendes:

- Eigenschaftsliste (Bildobjekt): Objekteigenschaft dynamisieren
  Verwenden Sie ein lokales Skript, um über dessen Rückgabewert eine Objekteigenschaft in Runtime zu dynamisieren. Beispiel: Objektposition in Abhängigkeit von mehreren Parametern ändern.
- Funktionsliste (Bildobjekt): Funktionsliste in lokales Skript umwandeln
  Wandeln Sie eine Funktionsliste in ein lokales Skript um, um bedingte Anweisungen oder Schleifen an ein Ereignis zu projektieren.
- Aufgabenplaner (Aufgabe):Lokales Skript zyklisch oder azyklisch ausführen
  Verwenden Sie den Aufgabenplaner, um ein Skript zu definierten Zeitpunkten auszuführen.

Lokale Skripte werden erst bei Verwendung in den Speicher des Bediengeräts geladen, z. B. beim Laden des Bildes. Lokale Skripte sparen damit Speicherplatz auf dem Bediengerät.

Programmierung

Wenn Sie ein lokales Skript an einer Objekteigenschaft erstellen, wird ein lokales Skript mit Rückgabewert erstellt. Der Rückgabewert liefert den Wert für die Objekteigenschaft.

Wenn Sie ein lokales Skript an einem Ereignis erstellen, wird ein lokales Skript ohne Rückgabewert erzeugt.

Die Parameter, die an das lokale Skript übergeben werden, hängen von der verwendeten Programmiersprache und der Verwendungsstelle ab.

Typische Parameter eines lokalen VB-Skripts

Item: Enthält die Objektreferenz:

- Aufgabenplaner (Aufgabe): Es wird kein Objekt übergeben.
- Menü & Symbolleisten: Das Objekt wird übergeben, welches das Ereignis auslöst.
- Bilder (Bildobjekt)
  - Eigenschaftsliste (Bildobjekt): Bei Dynamisierungen wird das Bildobjekt übergeben, dessen Eigenschaft Sie dynamisieren.
  - Umgewandelte Funktionsliste (Bildobjekt): Das Bildobjekt wird übergeben, dessen Ereignis Sie projektieren.

  Um auf die Eigenschaften des übergebenen Bildobjekts zuzugreifen, geben Sie "item." ein. Die verfügbaren Eigenschaften werden aufgelistet. Um auf ein anderes Objekt des Objektmodells zuzugreifen, verwenden Sie ScreenItems.
  Weiterführende Informationen zu Parametern von Ereignissen finden Sie in der Referenz.

Typische Parameter eines lokalen C-Skripts

Char* screenName: Enthält den Namen des Bildes, in dem das Objekt enthalten ist, dessen Eigenschaft Sie dynamisieren.

Char* objectName: Enthält den Objektnamen, dessen Eigenschaft Sie dynamisieren.

Char* propertyName: Enthält die Eigenschaft, die Sie dynamisieren.

Weiterführende Informationen zu Parametern von Ereignissen finden Sie in der Referenz.

Ausführen von lokalen Skripten

Lokale Skripte benötigen einen Trigger. Der Trigger legt fest, wann das Skript ausgeführt wird. Welcher Trigger verwendet wird, ist von der Verwendungsstelle des lokalen Skripts abhängig:

- Lokale Skripte aus einer umgewandelten Funktionsliste werden beim Eintritt des Ereignisses ausgeführt, z. B. Drücken.
- Lokale Skripte in der Eigenschaftsliste erhalten den Trigger automatisch. Sie können den Trigger ändern, z. B. in Variablentrigger.
- Lokale Skripte einer Aufgabe erhalten den vom Projekteur projektierten Trigger.

Siehe auch

Editor "Skripte" (Seite 1444)

9.7.2 Editor für lokale Skripte

Eigenschaften des Editors für lokale Skripte

Der Editor für lokale Skripte besitzt keine eigenen Einstellungsmöglichkeiten, sondern "erbt" die Einstellungen von "Runtime Scripting".

Der Editor bietet eine ähnliche Funktionalität wie der Editor "Skripte". Nähere Hinweise hierzu finden Sie in "Editor "Skripte" (Seite 1444)".

Folgende Funktionalitäten sind über die Symbolleiste nicht verfügbar:

- Auswahllisten für Eigenschaften und Funktionen

Im Editor für lokale VB-Skripte gibt es folgende zusätzlichen Symbole:

- Blendet Deklarationsbereich ein.
- Blendet Deklarationsbereich aus.

Im Deklarationsbereich legen Sie lokale VB-Variablen fest. Nähere Hinweise hierzu finden Sie unter "Verwendung von VB-Variablen (Seite 1455)".

Verfügbarkeit

Der Editor für lokale Skripte ist an folgenden Stellen aufrufbar:

- Aufgabenplaner
  - Ereignis: Ausführen
- Bilder
  - Ereignis
  - Objekteigenschaft (Eigenschaftsliste)
  - Eigenschaftsänderung (Eigenschaftsliste)
- Menüs und Symbolleisten
  - Ereignis: Klicken
- Nur C-Skript: Protokolle
  - E/A-Feld an der Eigenschaft Prozesswert

Siehe auch

Editor "Skripte" (Seite 1444)

Verwendung von VB-Variablen (Seite 1455)

9.7.3 Funktionsliste in lokales Skript umwandeln

Einleitung

Wenn Sie eine Funktionsliste in ein lokales Skript umwandeln, können Sie an ein Ereignis bedingte Anweisungen oder Schleifen projektieren. Eine Funktionsliste wandeln Sie entweder in ein lokales VB- oder C-Skript um.

Wenn in der Funktionsliste eine benutzerdefinierte Funktion enthalten ist, wird die Programmiersprache durch die Funktion bestimmt.

Ist z. B. eine C-Funktion enthalten, wird die Funktionsliste in ein lokales C-Skript umgewandelt.

Hinweis

Fehlermeldung im Global Script Diagnosefenster

Wenn in einer SetTag-Systemfunktion, die in ein C-Skript umgewandelt wurde, eine Variable vom Typ WString verwendet wird, wird beim Ausführen der Funktion eine Fehlermeldung im Global Script Diagnosefenster ausgegeben. Die Meldung hat keine Auswirkung auf die Ausführung der Funktion.

Folgende SetTag-Funktionen sind betroffen:

- SetTagWithOperatorEvent
- SetTagIndirect
- SetTagByTagIndirect
- SetTagIndirectByTagIndirect
- SetTagByProperty
- SetTagIndirectByProperty

Voraussetzung

Funktionsliste ist geöffnet.

Vorgehen

Um eine Funktionsliste in ein lokales Skript umzuwandeln, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ereignis, dessen Funktionsliste Sie in ein Skript umwandeln wollen.
2. Projektieren Sie bei Bedarf Systemfunktionen an das Ereignis.
3. Um die Funktionsliste in ein lokales Skript umzuwandeln, klicken Sie auf oder .
4. Schreiben Sie den Programmcode.

Alternativ verwenden Sie im Kontextmenü den Befehl "VB-Skript umwandeln" oder "C-Skript umwandeln".

Ergebnis

Wenn das Ereignis eintritt, wird das lokale Skript automatisch abgearbeitet.

Lokales Skript löschen

Um ein lokales Skript zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Ereignis mit dem lokalen Skript.
2. Wählen Sie im Kontextmenü des Ereignisses den Befehl "Funktionsliste".

Das lokale Skript wird gelöscht.

9.7.4 Objekteigenschaft mit lokalem Skript dynamisieren

Einleitung

Mit einem lokalen Skript dynamisieren Sie in Runtime eine Objekteigenschaft über den Rückgabewert des Skripts.

Voraussetzung

- Das Objekt mit der zu dynamisierenden Eigenschaft ist selektiert.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Klicken Sie im Inspektorfenster auf .
2. Selektieren Sie die zu dynamisierende Objekteigenschaft.
3. Wählen Sie in der Spalte "Dynamisierung" den Eintrag "VB-Skript" oder "C-Skript". Ein Dialog wird geöffnet.
4. Schreiben Sie den Programmcode.
5. Um den Trigger zu ändern, klicken Sie in der Symbolleiste auf " ".

Ergebnis

In Runtime ändert sich die Objekteigenschaft entsprechend der Projektierung in der Eigenschaftsliste.

Dynamisierung löschen

1. Selektieren Sie in der Eigenschaftsliste die dynamisierte Objekteigenschaft.
2. Wählen Sie in der Spalte "Dynamisierung" den Eintrag "Keine".

9.8 Debuggen

9.8.1 Debuggen von benutzerdefinierten VB-Funktionen

Übersicht

Um eine benutzerdefinierte VB-Funktion in Runtime zu testen, können Sie den z.B. den Debugger von "Microsoft Visual Studio 2008" verwenden.

Der Debugger wird mit dem WinCC-Setup mitgeliefert. Wenn bereits eine andere Visual Studio-Version installiert ist, z. B. Microsoft Visual Studio 2010, verwenden Sie den Debugger dieser Version.

Debugger für WinCC Runtime Professional

Der Debugger Microsoft Visual Studio 2008 Shell (Integrated Mode) befindet sich auf der DVD von WinCC Runtime Professional im Verzeichnis Support.

Debugger für WinCC Runtime Advanced

Sie können den Debugger Microsoft Visual Studio 2008 Shell (Integrated Mode) auf der Homepage von Microsoft kostenlos herunterladen. Achten Sie dabei darauf, dass sie die englisch-sprachige Variante verwenden.

Grundsätzlich sind zum Debuggen die aktuellen Versionen von Microsoft zu empfehlen.

Debugger für WinCC installieren

Vorgehensweise

1. Starten Sie das Setup von Microsoft Visual Studio 2008 Shell (Integrated Mode)
2. Folgen Sie den Anweisungen und übernehmen Sie die Standard-Einstellungen.
   Das folgende Programm wird installiert: Microsoft Visual Studio 2008 Shell (Integrated Mode) - ENU
3. Beim ersten Starten des Debuggers wird der Dialog "Visual Studio Just-In-Time Debugger" geöffnet.
   Wählen Sie den Eintrag "New instance of Visual Studio 2008".
   Um "Visual Studio 2008" als Standard-Debugger festzulegen, aktivieren Sie die Einstellung "Set the currently selected debugger as the default."

Microsoft Visual Studio 2008 Debugger

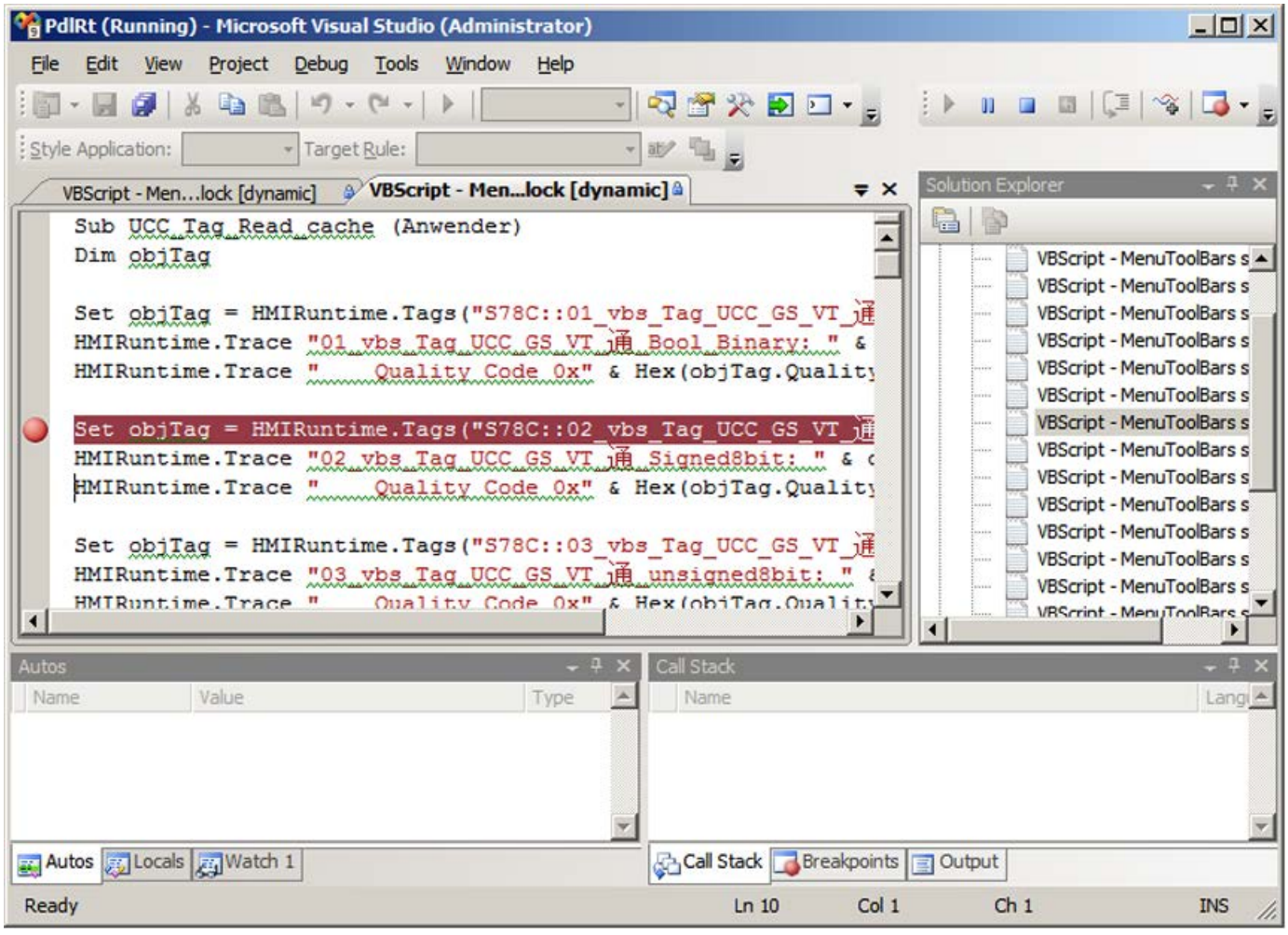

9.8.2 Grundlagen des Debuggens

Einführung

Mit einem Debugger testen Sie z. B., ob den Variablen die richtigen Werte übergeben werden und ob Abbruchbedingungen korrekt umgesetzt werden. Im Debugger können Sie

- den Quellcode der Funktion sichten
- die schrittweise Abarbeitung der Funktione kontrollieren
- Variablen und Eigenschaftenwerte anschauen und ändern
- den Funktionsablauf sehen und zu kontrollieren

Der Debugger ist nur dafür vorgesehen, Fehlerstellen in der benutzerdefinierten VB-Funktion zu finden. Dazu verwenden Sie z. B. Haltepunkte oder Sie führen die benutzerdefinierte VB-Funktion schrittweise aus. Nähere Hinweise hierzu finden Sie in der Dokumentation des verwendeten Debuggers. Um Fehler zu korrigieren, wechseln Sie nach WinCC in den Editor "Skripte". Nach nochmaligem Übersetzen und Laden prüfen Sie die benutzerdefinierte VB-Funktion erneut mit dem Debugger.

Hinweis

Beachten Sie, das im Debugger Ihr Code schreibgeschützt dargestellt wird. Sie können den Code nicht direkt im Debugger ändern, sondern nur die erforderlichen Änderungen testen.

Unterschied bei VBScript für Windows und Windows CE

Der Debugger überprüft die Syntax für VBScript für Windows. Wenn in der Funktion eine VBScript-Funktion für Windows CE enthalten ist, wird eine entsprechende Fehlermeldung ausgegeben. Einige VBScript-Funktionen sind unterschiedlich, z. B. CreateObject.

Fehlertypen

Folgende Fehlertypen werden beim Debuggen unterschieden:

- Syntaxfehler
  Syntaxfehler entstehen, wenn Sie z.B. ein Schlüsselwort falsch schreiben oder eine Klammer nicht schließen. Wenn Sie die Syntaxprüfung von WinCC einsetzen, können Sie Syntaxfehler bereits vor dem Test Ihrer Funktionen in Runtime ausschließen.
  Durch die Syntaxprüfung in WinCC wird eine Funktion geparst, ohne sie auszuführen. Unmittelbar vor der Ausführung in Runtime wird die Funktion nochmals geparst. Es werden alle Teile geparst, auch diejenigen, die z.B. erst nach Eintreten einer bestimmten Aktion zu einem späteren Zeitpunkt ausgeführt werden.
  Wenn in einer Funktion Syntaxfehler enthalten sind, können Sie Runtime nicht starten.

  Hinweis

  Für Runtime Professional können Sie in "Runtime-Einstellungen > Allgemein > Optionen für Skripte" festlegen, dass Runtime auch gestartet wird, wenn in einer Funktion Syntaxfehler enthalten sind.

- Laufzeitfehler
  Ein Laufzeitfehler tritt auf, wenn ein ungültiger/fehlerhafter Befehl ausgeführt werden soll, z.B. weil eine Variable nicht definiert ist. Um Laufzeitfehler abzufangen, können Sie in VBScript die Anweisung "On Error Resume Next" verwenden. Die Anweisung bewirkt, dass nach einem Laufzeitfehler die Folgeanweisung ausführt wird. In der Folgezeile kann dann der Fehlercode über das Err-Objekt überprüft werden. Um die Behandlung von Laufzeitfehlern im Skript wieder abzuschalten, verwenden Sie die Anweisung "On Error Goto 0".

- Logische Fehler
  Besonders hilfreich ist der Debugger zum Bereinigen von logischen Fehlern. Ein logischer Fehler tritt auf, wenn nicht das von Ihnen erwartete Ergebnis eintritt, weil z.B. eine Bedingung falsch überprüft wird. Um logische Fehler zu bereinigen, arbeiten Sie Ihre Funktion schrittweise ab, um den nicht funktionierenden Teil des Skriptes zu identifizieren.

Beispiel zur Fehlerbehandlung

```
Sub OnClick(ByVal Item)
'VBS27
  Dim objScreenItem
'
  'Activation of errorhandling:
  On Error Resume Next
  For Each objScreenItem In ScreenItems
    If "HMIRectangle" = objScreenItem.Type Then
    '
      '=== Property "RoundCornerHeight" only available for RoundRectangle
      objScreenItem.RoundCornerHeight = objScreenItem.RoundCornerHeight * 2
      If 0 <> Err.Number Then
        HMIRuntime.Trace objScreenItem.Name & ": no RoundedRectangle" & vbCrLf
        '
        'Delete error message
        Err.Clear
      End If
    End If
  Next
  On Error Goto 0  'Deactivation of errorhandling
End Sub
```

Grundlegendes Vorgehen

Wenn ein Fehler aufgetreten ist, und der Debugger geöffnet ist, erscheint das Skript schreibgeschützt in einem Fenster. Sie können durch das Skript-Dokument navigieren, Haltepunkte setzen, das Skript erneut in Runtime ausführen, und die Skripte schrittweise abarbeiten.

Sie können den Quellcode Ihrer Skripte nicht direkt im Debugger editieren. Wenn Sie einen Fehler gefunden haben, können Sie den Fehler in der Funktion in WinCC korrigieren, das Bild erneut laden und es im Debugger aktualisieren.

Bildwechsel beim Debuggen

Wenn Sie während des Debuggens einen Bildwechsel durchführen, bleibt das Skriptdokument des "alten" Bildes geöffnet, ist aber nicht mehr gültig. Es werden ggf. ungültige Fehler angezeigt, da die aufgerufenen Objekte nach dem Bildwechsel nicht mehr vorhanden sind.

9.8.3 Debugger aktivieren

Prinzip

Sie haben mehrere Möglichkeiten, den Debugger zu aktivieren:

- Automatischer Aufruf des Debuggers bei Auftreten eines Fehlers in Runtime.
- Einblenden einer Fehlerbox in Runtime, über die der Debugger aufgerufen werden kann.
- Starten des Debuggers aus dem Startmenü und öffnen eines laufenden Runtime-Skriptes.

Visual Studio mit Runtime verbinden

1. Starten Sie Runtime.
2. Öffnen Sie "Microsoft Visual Studio 2008".
3. Klicken Sie im Menü "File > New" auf "File".
4. Wählen Sie "Windows Script Host" und klicken Sie auf "Open".
5. Klicken Sie im Menü "Debug" auf "Attach to Process".
6. Wählen Sie:
   HmiRTm.exe: für ein Panel oder WinCC Runtime Advanced
   PdlRt.exe: für WinCC Runtime Professional
7. Doppelklicken Sie eine der Script-Dateien im "Solution Explorer".

Visual Studio als Just-In-Time-Debugger

1. Öffnen Sie "Microsoft Visual Studio 2008".
2. Klicken Sie im Menü "Tools" auf "Options".
3. Klicken Sie im Baum auf "Debugging" und "Just-In-Time".
4. Aktivieren Sie unter "Just-In-Time Debugging" die Option "Script".
5. Schließen Sie Visual Studio.
6. Starten Sie die Simulation über "Online > Simulation > Mit Skript-Debugger".

Wenn in Runtime eine fehlerhafte Funktion ausgeführt wird, öffnet sich ein Dialog, über den Sie den Debugger öffnen können.

Beenden des Debuggers

Sie können den Debugger beenden, ohne dass WinCC-Runtime beendet wird.

9.8.4 Aufbau von VBScript-Dateien

Aufbau der VBScript-Dateien

Alle in einem Gerät projektierten benutzerdefinierten Funktionen sind in einer Skript-Datei enthalten.

9.8.5 Aufbau von VBScript-Dateien

Prinzip

Um in Runtime Professional die gleichzeitige Abarbeitung von zyklischen und ereignisgesteuerten VBScript-Dateien zu ermöglichen, sind ereignisgesteuerte VBScript-Dateien und zyklische/variablengetriggerte VBScript-Dateien bei der Abarbeitung streng getrennt. So kann z. B. ein zyklisches lokales Skript nicht die Ausführung einer benutzerdefinierten Funktion behindern, die beim Drücken einer Schaltfläche ausgeführt werden soll.

Um dies zu gewährleisten, werden die ereignisgesteuerten lokalen Skripte und die zyklischen- sowie die variablengetriggerten lokalen Skripte in getrennten VBScript-Dateien abgelegt. Wenn Sie in VBScript-Dateien im Editor "Bilder" einen bildglobalen Teil definiert haben, wird dieser in beide VBScript-Dateien kopiert.

Hinweis

Da beide VBScript-Dateien getrennt behandelt werden, haben sie keinen gemeinsamen Datenbereich. Es erfolgt daher kein Abgleich globaler Variablen zwischen den beiden VBScript-Dateien. Wenn eine Synchronisation erforderlich ist, realisieren Sie diese über das DataSet-Objekt oder interne HMI-Variablen.

Aufbau der VBScript-Dateien

Wenn Sie VBScript-Dateien mit dem Debugger debuggen, öffnen sich immer die VBScript-Dateien der unterschiedlichen Runtime-Systeme.

Für das Grafische Runtime-System bedeutet dies, dass Sie pro Bild zwei Dateien erhalten:

- <Bildname>.pdl_events: Enthält die ereignisgesteuerten VBScript-Dateien
- <Bildname>.pdl_triggers: Enthält die zyklischen und variablengesteuerten VBScript-Dateien.

Im Folgenden sehen Sie, wie die VBScript-Dateien im grafischen Runtime-System aufgebaut sind:

- Inhalte der VBScript-Datei <bildname>.pdl_events
  Allgemeiner Code
  Ereignisgesteuerte Skripte
- Inhalte der VBScript-Datei <bildname>.pdl_triggers
  Allgemeiner Code
  Zyklische oder Variablengetriggerte Skripte

Hinweis

Beachten Sie, dass die lokalen Skripte und benutzerdefinierten Funktionen des grafischen Runtime-Systems nicht mit den Namen in der VBScript-Datei angezeigt werden, unter der Sie sie in WinCC gespeichert haben.

9.8.6 Namen im Debugger

Namen im Debugger

Die Namen von benutzerdefinierten Funktionen in den VBScript-Dateien im Debugger unterscheiden sich von den Namen, unter denen Sie Ihre benutzerdefinierte Funktionen in WinCC gespeichert haben.

Die Funktions- und Skriptnamen in den Skript-Dateien werden nach folgenden Regeln zusammengesetzt:

Aktionstyp	Name in der Skript-Datei
Zyklische oder variablengetriggerte Skripte an einer Eigenschaft	ObjektName_Eigenschaftsname_Trigger
Maus-Ereignisse	ObjektName_OnClick ObjektName_OnLButtonDown ObjektName_OnLButtonUp ObjektName_OnRButtonDown ObjektName_OnRButtonUp
Tastatur-Ereignisse	ObjektName_OnKeyDown ObjektName_OnKeyUp
Objekt-Ereignisse	ObjektName_OnObjectChanged ObjektName_OnSetFocus
Ereignisse an Eigenschaften	ObjektName_PropertyName_OnPropertyChanged ObjektName_PropertyName_OnPropertyState-Changed
Bildereignisse	Document_OnOpen Document_OnClosed

Zugelassene Länge der Skriptnamen

Die Namen der Skripte in den Skript-Dateien sind auf 255 Zeichen beschränkt. Jedes Sonderzeichen, das Sie in einem Objektnamen verwenden, wird in fünf Zeichen umgesetzt. Hinter einem führenden X wird das Sonderzeichen in vierstelligem Hexadezimalcode dargestellt. Wenn Sie z. B. auf Mausklick eine Aktion an eine Schaltfläche mit dem Namen "DrückMich" projektieren, erscheint das Skript in der Skript-Datei als "DrXFFFCckMich_OnClick".

Wenn der zusammengesetzte Objektname zu lang wird, wird bei der Syntaxprüfung in WinCC eine Fehlermeldung ausgegeben. Wegen dieser Einschränkung können Sie Ihre Grafikobjekte während der Projektierung nicht beliebig lang wählen.

Hinweis

Um den Namen eines Objekts in Runtime zu ermitteln, drücken Sie <Strg+Alt+Shift> und bewegen die Maus über dem entsprechenden Objekt. Der Bildname und der Objektname werden dann in einem Tooltip angezeigt.

9.8.7 Haltepunkte

Einführung

Um an bestimmten Stellen die Abarbeitung der VBScript-Datei zu stoppen und logische Fehler Schritt für Schritt einzugrenzen, setzen Sie Haltepunkte.

Wenn eine VBScript-Datei im Debugger aktualisiert wird, gehen alle Haltepunkte verloren.

Hinweis

Wenn Sie bei Runtime Professional in der Skript-Datei "<Bildname>.pdl_trigger" einen Haltepunkt setzen, werden alle triggergesteuerten Prozeduren angehalten.

Wenn Sie bei Runtime Professional in der Skript-Datei "<Bildname>.pdl_event" einen Haltepunkt setzen, werden alle eventgesteuerten Prozeduren angehalten.

Voraussetzung

- Runtime ist aktiviert.
- Das Bild, das Sie debuggen wollen, ist aktiv.

Haltepunkt setzen

1. Starten Sie den Debugger und wählen Sie die VBScript-Datei aus.
   Wenn Sie die automatische Aktivierung des Debuggers in WinCC gewählt haben, wird der Debugger aufgerufen, sobald ein fehlerhaftes Skript ausgeführt wird.
2. Setzen Sie den Cursor in die Zeile, in der Sie einen Haltepunkt setzen wollen.

3. Wählen Sie im Menü "Debug" den Befehl "Toggle Breakpoint".
   Die Zeile wird mit einem roten Punkt markiert.
4. Wechseln Sie in WinCC Runtime und führen Sie die VBScript-Datei aus, die Sie debuggen wollen.
   Der Debugger stoppt am ersten Haltepunkt. Die aktuelle Zeile wird gelb hinterlegt dargestellt.
   Sie gehen nun den Programmcode schrittweise durch.

Haltepunkt löschen

1. Setzen Sie den Curso in eine Zeile mit einem Haltepunkt
2. Wählen Sie im Menü "Debug" den Befehl "Toggle Breakpoint".

Um alle Haltepunkte in einer Datei zu löschen wählen Sie im Menü "Debug" den Befehl "Delete All Breakpoints".

9.8.8 Schrittweise arbeiten

Einführung

Sie verwenden den Debugger, um Ihre VBScript-Dateien (benutzerdefinierte Funktion oder lokales Skript) Schritt für Schritt abzuarbeiten und so z. B. logische Fehler systematisch einzugrenzen. Sie testen die Auswirkungen jeder einzelnen Codezeile in Runtime.

Voraussetzung

- Runtime ist aktiviert.
- Das Bild, das Sie debuggen wollen, ist aktiv.

Prinzipielle Vorgehensweise

1. Starten Sie den Debugger und wählen Sie die VBScript-Datei aus.
   Wenn Sie die automatische Aktivierung des Debuggers in WinCC gewählt haben, wird der Debugger aufgerufen, sobald ein fehlerhaftes Skript ausgeführt wird.
2. Setzen Sie in der VBScript -Datei einen Haltepunkt. Haltepunkte setzen Sie typischerweise vor Codezeilen, in denen Sie Fehler vermuten.
3. Wechseln Sie zu WinCC Runtime und lösen Sie ein Ereignis aus, das die VBScript-Dateien zum Ablaufen bringt.
   Der Debugger hält am ersten Haltepunkt und markiert die aktuelle Zeile.

4. Um schrittweise weiterzuarbeiten, wählen Sie einen der folgenden Menübefehle:
   – "Debug > Step Over": Geht zur nächsten Codezeile. Wenn in dieser Zeile eine benutzerdefinierte Funktion aufgerufen wird, überspringen Sie die gerufene benutzerdefinierte Funktion. Die benutzerdefinierte Funktion wird ausgeführt, aber der Debugger führt Sie nicht durch die einzelnen Zeilen der Funktion. Stattdessen springen Sie zur ersten Zeile, nachdem die Funktion ausgeführt wurde.
   – "Debug > Step Into": Geht zur nächsten Codezeile. Wenn in dieser Zeile eine benutzerdefinierte Funktion aufgerufen wird, springen Sie zu dieser Funktion. Sie arbeiten die gerufene Funktion dann schrittweise ab.
   – Um das schrittweise Abarbeiten einer aufgerufenen Funktion abzubrechen, wählen Sie den Menübefehl "Debug > Step Out". Die Funktion wird vollständig ausgeführt. Der Debugger springt zur ersten Zeile, nachdem die Funktion ausgeführt wurde.
5. Gehen Sie schrittweise bis zum Ende des Dokuments oder wählen Sie den Menübefehl "Debug > Run", um die VBScript-Datei bis zum Ende abzuarbeiten.

9.8.9 Befehle ausführen

Einführung

Im "Command Window" werden Befehle eingegeben und direkt ausgeführt. Sie können:

- Methoden aufrufen
- Prozeduren aufrufen
- Objekteigenschaften verändern

Sie führen im "Command Window" prinzipiell alle Befehle aus, die Sie auch in einem VB-Skript ausführen können.

Hinweis

Wenn Sie den Namen eines WinCC-Objekts in WinCC Runtime Professional ermitteln wollen, drücken Sie <Strg+Alt+Shift> und bewegen die Maus über das entsprechende Objekt. Der Bildname und der Objektname werden dann in einem Tooltip angezeigt.

Voraussetzung

- Die VBScript-Datei läuft in Runtime.
- Der Debugger ist geöffnet.

Vorgehen

1. Setzen Sie mindestens einen Haltepunkt in der VBScript-Datei.
2. Wechseln Sie nach WinCC Runtime und führen Sie eine Aktion aus, die die VBScript-Datei zur Ausführung bringt.
   Der Debugger stoppt am ersten Haltepunkt.

3. Wählen Sie "View > Other Windows > Command Window". Das "Command Window" wird geöffnet.
4. Geben Sie den gewünschten Befehl ein und bestätigen Sie mit <Return>.

Hinweis

Wenn Sie einen fehlerhaften Befehl im "Command Window" eingeben, wird in Runtime keine Fehlermeldung erzeugt. Stattdessen erscheint im Command Window die Meldung "<Script Error>".

Beispiel

```
Dim i
For i = 1 to 10
    HMIRuntime.Trace i & vbnewline
Next
```

Wenn dieses VBScript ausgeführt wird und der Debugger in der Schleife am Haltepunkt steht, können Sie mit

- ?i
  den Wert der Variablen i auslesen
- i = 5
  den Wert der Variablen i auf 5 setzen.

9.8.10 Variablen- oder Eigenschaftswerte ermitteln

Einführung

Im "Command Window" können Sie Werte von Variablen oder Eigenschaften ermitteln und ändern. So setzen Sie z. B. einen Prozesswert für die benutzerdefinierte Funktion wieder auf Null, ohne den Prozess anhalten zu müssen.

Hinweis

Wenn Sie den Namen eines WinCC-Objekts in WinCC Runtime Professional ermitteln wollen, drücken Sie <Strg+Alt+Shift> und bewegen die Maus über das entsprechende Objekt. Der Bildname und der Objektname werden dann in einem Tooltip angezeigt.

Voraussetzung

- Die VBScript-Datei läuft in Runtime.
- Der Debugger ist geöffnet.

Vorgehen

1. Setzen Sie mindestens einen Haltepunkt in der VBScript-Datei
2. Wechseln Sie nach WinCC Runtime und führen Sie eine Aktion aus, die die VBScript-Datei zur Ausführung bringt.
   Der Debugger stoppt am ersten Haltepunkt.
3. Wählen Sie "View > Other Window > Command Window". Das "Command Window" wird geöffnet.
4. Um den Wert einer Variablen oder Eigenschaft zu ermitteln, geben Sie folgende Zeichenkette ein
   – ? myTag
   – ? myObject

   Beachten Sie das Leerzeichen vor dem Objektnamen.
5. Drücken Sie <Return>.

9.8.11 Diagnose

9.8.11.1 Diagnose

Einleitung

Wenn Sie Ihre lokalen VB-Skripte und C-Skripte in Runtime ausführen und testen, geben Sie mit Hilfe des Objekts "Druckauftrag/Skriptdiagnose" schnell eine Analyse aus.

Diagnose-Werkzeuge

WinCC stellt Ihnen Werkzeuge zur Verfügung, um das Laufzeitverhalten in lokalen C-Skripten und VB- Skripten zu analysieren:

- Das Objekt Druckauftrag/Skriptdiagnose "GSC-Diagnose"
- Verwenden eines Debuggers

Verwendung

Um die GSC-Diagnose zu verwenden, fügen Sie das Objekt "Druckauftrag/Skriptdiagnose" in ein Bild ein. Verwenden Sie bei Vorlage "GSC-Diagnose".

Das Bild kann ein eigens zu Diagnosezwecken entworfenes Bild sein, das in Runtime aufgerufen wird.

GSC-Diagnose von VBS

GSC-Diagnose gibt die in den lokalen Skripten enthaltenen Trace-Methoden in der zeitlichen Reihenfolge ihres Aufrufs aus. Auch Trace-Anweisungen in benutzerdefinierten VB-Funktionen, die in lokalen Skripten aufgerufen werden, werden ausgegeben. Durch gezielten Einsatz von Trace-Anweisungen, z. B. zur Ausgabe von Variablenwerten, lässt sich so der Ablauf von Skripten und den darin aufgerufenen Funktionen verfolgen. Die Trace-Anweisung geben Sie in der Form "HMIRuntime.Trace(<Ausgabe>)" an. Auch Fehlersituationen, die den Aufruf der Funktion OnErrorExecute zur Folge haben, werden im Diagnosefenster angezeigt.

GSC-Diagnose von C

GSC-Diagnose gibt die in den lokalen Skripten enthaltenen printf-Methoden in der zeitlichen Reihenfolge ihres Aufrufs aus. Auch printf-Anweisungen in benutzerdefinierten C-Funktionen, die in lokalen Skripten aufgerufen werden, werden ausgegeben. Durch gezielten Einsatz von printf-Anweisungen, z. B. zur Ausgabe von Variablenwerten, lässt sich so der Ablauf von Skripten und den darin aufgerufenen Funktionen verfolgen.

Siehe auch

Beispiel für das Projektieren einer Diagnoseausgabe über Trace (Seite 1515)

9.8.11.2 GSC-Diagnose

Einleitung

GSC-Diagnose gibt die in den Skripten enthaltenen Trace-Anweisungen in der zeitlichen Reihenfolge ihres Aufrufs im Diagnosefenster aus. Auch Trace-Anweisungen in Funktionen, die in Skripten aufgerufen werden, werden ausgegeben. Durch gezielten Einsatz von Trace-Anweisungen, z. B. zur Ausgabe von Variablenwerten, lässt sich so der Ablauf von Skripten und den darin aufgerufenen Funktionen verfolgen.

Anwendung

Damit Sie GSC-Diagnose verwenden können, fügen Sie das Objekt "Druckauftrag/ Skriptdiagnose" vom Typ GSC-Diagnose in ein Bild ein. Über die Eigenschaften von GSC-Diagnose steuern Sie das Erscheinungsbild des GSC-Diagnosefensters.

Bei einem Bildwechsel wird der Inhalt des GSC-Diagnosefensters gelöscht.

Hinweis

Meldungen im Fenster "GSC-Diagnose" werden auch ausgegeben, wenn der Debugger aktiviert ist.

9.8.11.3 GSC-Diagnosefenster in ein Bild einfügen

Einleitung

Um die GSC-Diagnose zu verwenden, fügen Sie das Objekt "Druckauftrag/Skriptdiagnose" in ein Bild ein. Sie verwenden entweder ein beliebiges vorhandenes Bild oder erstellen ein Bild, das ausschließlich zu Diagnosezwecken dient. Im Inspektorfenster des Objekts "Druckauftrag/Skriptdiagnose" legen Sie fest, dass das Objekt zur GSC-Diagnose verwendet wird.

Voraussetzung

- Das gewünschte Bild ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

Vorgehen

Um das GSC-Diagnosefenster einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie das Objekt "Druckauftrag/Skriptdiagnose" aus der Palette "Controls" der Task Card "Werkzeuge" in das Bild ein.
2. Vergeben Sie unter "Verschiedenes > Name" einen Namen für das GSC-Diagnosefenster.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Inhalt > Fensterinhalt" die Option "Funktionen".
4. Wählen Sie unter "Vorlage" die Option "GSC-Diagnose".
5. Legen Sie unter "Allgemein", "Darstellung" und "Verschiedenes" die weiteren Einstellungen für das GSC-Diagnosefenster fest.

9.9 Laufzeitverhalten in Runtime

9.9.1 Abarbeitung von benutzerdefinierten Funktionen in Runtime

Prinzip

In Runtime wird immer nur eine benutzerdefinierte Funktion ausgeführt. Wenn mehrere benutzerdefinierte Funktionen zur Bearbeitung anstehen, werden diese in einer Warteschlange eingereiht und der Reihe nach abgearbeitet.

Hinweis

Eine Schleife in einer benutzerdefinierten Funktion blockiert damit die Ausführung der anderen Funktionen in der Warteschlange, auch wenn die Funktionen asynchron angestoßen wurden.

WinCC unterstützt eine maximale Schachtelungstiefe von acht benutzerdefinierten Funktionen. Beachten Sie, dass die Schachtelungstiefe nicht überprüft wird.

Hinweis

Wenn Sie an das Ereignis "Runtime-Stopp" eine benutzerdefinierte Funktion projektieren, dürfen Sie in dieser benutzerdefinierten Funktion nur die Systemfunktionen verwenden, die am Ereignis "Runtime-Stopp" verfügbar sind.

Stellen Sie sicher, dass das Beenden der Runtime nicht durch die Ausführung der benutzerdefinierten Funktion gestört wird.

Hinweis

Projektierung von benutzerdefinierten Funktionen

Beachten Sie, dass zu einem Zeitpunkt nicht übermäßig viele benutzerdefinierte Funktionen gleichzeitig angestoßen werden. Vermeiden Sie eine andauernde Systemauslastung von 100 %.

Um die Darstellung von Werten und die Bedienbarkeit nicht zu behindern, werden benutzerdefinierte Funktionen auf niedriger Priorität abgearbeitet. Bei extremer Systemauslastung werden die auszuführenden benutzerdefinierten Funktionen daher zunächst nur für die Ausführung vorgemerkt. Die maximale Größe der Vormerkliste ist bediengeräteabhängig und ist begrenzt durch die maximal zulässige Anzahl der benutzerdefinierten Funktionen. Nähere Informationen hierzu finden Sie in den Leistungsmerkmalen. Wenn zu einem Zeitpunkt, mehr benutzerdefinierte Funktionen angestoßen als vorgemerkt werden können, werden überzählige Aufrufe verworfen und eine Systemmeldung ausgegeben.

Bediengerätewechsel

Wenn Sie in einer benutzerdefinierten Funktion Systemfunktionen verwenden, die auf dem eingestellten Bediengerät nicht verfügbar sind, erhalten Sie einen Warnhinweis. Zusätzlich wird die entsprechende Systemfunktion in der benutzerdefinierten Funktion mit einer blauen Wellenlinie unterstrichen.

Siehe auch

Benutzerdefinierte Funktionen (Seite 1435)

9.9.2 Abarbeitung der Funktionsliste in Runtime

Prinzip

Eine Funktionsliste wird in Runtime von oben nach unten abgearbeitet. Damit es während der Abarbeitung nicht zu Wartezeiten kommt, wird zwischen synchroner Abarbeitung und asynchroner Abarbeitung unterschieden. Die Unterscheidung wird vom System getroffen, indem es die verschiedenen Laufzeiten der Systemfunktionen auswertet. Benutzerdefinierte Funktionen werden unabhängig von der Laufzeit stets synchron abgearbeitet. Wenn eine Systemfunktion einen Fehlerstatus zurückgibt, wird die Abarbeitung der Funktionsliste abgebrochen.

Synchrone Abarbeitung

Bei der synchronen Abarbeitung werden die Systemfunktionen in einer Funktionsliste nacheinander ausgeführt. Damit eine Systemfunktion ausgeführt wird, muss die vorhergehende Systemfunktion beendet sein.

Asynchrone Abarbeitung

Systemfunktionen, die Dateioperationen wie Speichern oder Lesen ausführen, haben eine längere Laufzeit als Systemfunktionen, die z. B. einen Variablenwert setzen.

Systemfunktionen mit längerer Laufzeit werden deshalb asynchron ausgeführt. Während eine Systemfunktion z. B. einen Rezepturdatensatz auf ein Speichermedium schreibt, wird bereits die nächste Systemfunktion ausgeführt. Durch die parallele Abarbeitung der Systemfunktionen werden Wartezeiten am Bediengerät vermieden.

9.9.3 Abarbeitung von Funktionen und lokalen Skripten in Runtime

Abarbeitung von lokalen Skripten in Runtime

Zwei lokale Skripte vom gleichen Typ können in Runtime nicht gleichzeitig ausgeführt werden. Damit z. B. zyklische Skripte nicht durch ein Skript behindert werden, das auf Mausklick ausgeführt wird, werden ereignisgetriggerte Skripte und zyklische/variablengetriggerte Skripte in Bildern unabhängig voneinander ausgeführt.

Hinweis

Die Synchronisation zwischen den Skriptarten in WinCC erfolgt ausschließlich durch das DataSet-Objekt oder interne HMI-Variablen. Durch die getrennte Abarbeitung besteht kein gemeinsamer Datenbereich zwischen ereignisgetriggerten und zyklischen/ variablengetriggerten lokalen Skripten.

Wenn die Abarbeitung zyklischer lokaler Skripte z. B. durch hohe Systemlast oder andere Skripte behindert wird, wird das lokale Skript bei der nächstmöglichen Gelegenheit wieder einmalig ausgeführt. Die nicht ausgeführten Zyklen werden nicht in einer Warteschleife gehalten, sondern verworfen.

Nach einem Bildwechsel noch laufende lokale Skripte werden 1 Minute nach dem Bildwechsel automatisch beendet.

Lokale Skripte, die beim Beenden von Runtime noch laufen, werden nach 5 Sekunden beendet.

Abarbeitung von Funktionen in Runtime

Funktionen werden in Runtime bei Auslösung nacheinander ausgeführt. Wird eine Funktion angestoßen, während gerade eine andere Funktion läuft, wird die zweite Funktion in einer Warteschleife gehalten, bis die Ausführung möglich ist.

Hinweis

Die Synchronisation zwischen lokalen Skripten und Funktionen erfolgt ausschließlich durch das DataSet-Objekt oder interne HMI-Variablen. Es besteht kein gemeinsamer Datenbereich zwischen lokalen Skripten und Funktionen.

Siehe auch

Benutzerdefinierte Funktionen (Seite 1435)

Lokale Skripte (Seite 1437)

9.9.4 Reihenfolge bei Abarbeitung von benutzerdefinierten Funktionen und Systemfunktionen

⚠ WARNUNG
Die erwartete Reihenfolge der Abarbeitung von benutzerdefinierten Funktionen und Systemfunktionen in Runtime kann nicht immer eingehalten werden. Dieser Sachverhalt führt eventuell zu unvorhergesehenen Betriebszuständen. Bitte beachten Sie die nachfolgende Beschreibung, in der Zusammenhänge erläutert werden. Unter "Abhilfe" erhalten Sie eine Anweisung zur Lösung.

Verwendung von benutzerdefinierten Funktionen an Ereignissen

Die beim Projektieren erwartete Reihenfolge der Bearbeitung der benutzerdefinierten Funktionen und Systemfunktionen kann unter Umständen nicht eingehalten werden, z. B. wenn die Bearbeitung einer benutzerdefinierten Funktion über den Auslösezeitpunkt eines nachfolgenden Ereignisses andauert. In diesem Fall kann es passieren, dass Systemfunktionen, die am nachfolgenden Ereignis projektiert sind, vor den Systemfunktionen, die am gleichen Ereignis mit der benutzerdefinierten Funktion projektiert wurden, ausgeführt werden.

Hintergrund

Benutzerdefinierte Funktionen werden in einer niederprioren Wartenschlange aufgereiht. Die Warteschlange wird sequenziell abgearbeitet. Werden an einem Ereignis Systemfunktionen und benutzerdefinierte Funktionen projektiert, werden alle an diesem Ereignis projektierten Systemfunktionen und benutzerdefinierten Funktionen in dieser Warteschlange ausgeführt.

Beispiel

An einer Taste haben Sie Folgendes projektiert:

	Ereignis	Funktionen
	Drücken	Benutzerdefinierte Funktion (1) SetzeBit(Variable) (2)
	Loslassen	RücksetzeBit(Variable) (3)

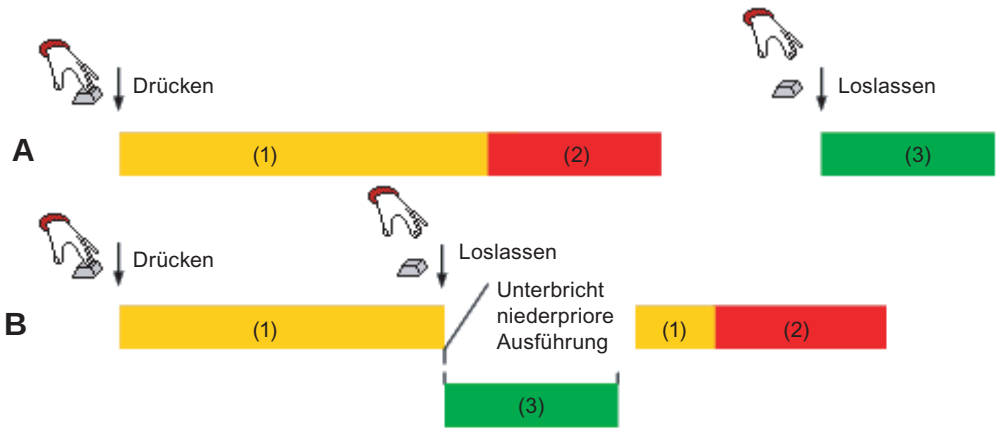

(1) Ausführungszeit der benutzerdefinierten Funktion

(2) Ausführungszeit der Systemfunktion SetzeBit(Variable)

(3) Ausführungszeit der Systemfunktion RücksetzeBit(Variable)

A: Erwartete Ausführung der benutzerdefinierten Funktion und der Systemfunktionen beim Projektieren.

Wenn Sie die Taste drücken, wird die benutzerdefinierte Funktion (1) und danach die Systemfunktion SetzeBit(Variable) (2) ausgeführt. Wenn Sie die Taste loslassen, wird die Systemfunktion RücksetzeBit(Variable) (3) ausgeführt.

B: Bearbeitung der benutzerdefinierten Funktion ist beim Loslassen der Taste noch nicht abgeschlossen.

Wenn Sie die Taste drücken, wird die benutzerdefinierte Funktion (1) ausgeführt. Die Ausführung der benutzerdefinierten Funktion (1) ist noch nicht beendet und Sie haben die Taste schon losgelassen. Die Systemfunktion RücksetzeBit(Variable) (3) unterbricht die Ausführung der Funktion. Somit wird die Systemfunktion RücksetzeBit(Variable) (3) vor der Systemfunktion SetzeBit(Variable) (2) ausgeführt. In der Konsequenz bleibt das Bit gesetzt.

Die Systemfunktion "SetzeBitWährendTasteGedrückt" zeigt das gleiche Verhalten, wenn die Systemfunktion zusammen mit benutzerdefinierten Funktionen projektiert wird.

Abhilfe

Die Ausführungsreihenfolge vom vorherigen Beispiel A wird eingehalten, wenn Sie an das Ereignis "Loslassen" zusätzlich eine benutzerdefinierte Funktion projektieren. Die benutzerdefinierte Funktion braucht dazu keine Funktionalität beinhalten.

	Ereignis	Funktionen
	Drücken	Benutzerdefinierte Funktion(1) SetzeBit(Variable) (2)
	Loslassen	Benutzerdefinierte Funktion_2 (3) RücksetzeBit(Variable) (4)

Hinweis

Es ist sicherzustellen, dass die Warteschlange der benutzerdefinierten Funktion nicht überläuft, sonst kann es dazu kommen, dass die Systemfunktion an dem nachfolgenden Ereignis nicht mehr ausgeführt werden.

Ein Überlauf ist erkennbar an "Überlast: Skript <Name> wird verworfen" oder "Die erwartete Reihenfolge der Abarbeitung von benutzerdefinierten Funktionen und Systemfunktionen in Runtime kann nicht immer eingehalten werden".

9.9.5 Dynamisieren von Objekteigenschaften in Runtime

Einleitung

Mit benutzerdefinierten Funktionen können Sie in Runtime auf Objekteigenschaften von Bildobjekten und auf Variablen zugreifen. Wenn Sie mit einer benutzerdefinierten Funktion Werte von Objekteigenschaften ändern, hat dies keine Auswirkung auf die Projektdaten.

Änderungen an Objekteigenschaften

Wenn Sie mit einer benutzerdefinierten Funktion in Runtime eine Objekteigenschaft eines Bildelements ändern, ist diese Änderung nur so lange wirksam, wie das Bild aktiv ist. Wenn Sie das Bild wechseln oder Sie das Bild neu laden, werden die projektierten Objekteigenschaften wieder angezeigt.

Sprachumschaltung

Wenn Sie in Runtime die Sprache umschalten, werden die fremdsprachigen Beschriftungen aus den Projektdaten geladen. Wenn Sie mit einer benutzerdefinierten Funktion Texte ändern, werden die von der benutzerdefinierten Funktion geänderten Texte wieder überschrieben.

9.10 Beispiele

9.10.1 Beispiele zu VBS

9.10.1.1 Beispiel: Umrechnen von Fahrenheit in Grad Celsius

Aufgabe

In diesem Beispiel erstellen Sie eine benutzerdefinierte VB-Funktion, die den Wert eines Temperaturmessfühlers von Fahrenheit in Grad Celsius umrechnet. Die Temperatur wird am Bediengerät in einem Ausgabefeld angezeigt.

Einstellungen

Für das Beispiel benötigen Sie zwei HMI-Variablen und eine benutzerdefinierte VB-Funktion mit folgenden Einstellungen:

HMI-Variablen:

Name	Steuerungsanbindung	Typ
Fahrenheit	ja	Real
Celsius	nein	Real

Benutzerdefinierte VB-Funktion:

Name	Typ	Parameter
FahrenheitToCelsius	Function	Fahrenheit

Vorgehen

1. Erstellen Sie die beiden HMI-Variablen "Fahrenheit" und "Celsius" mit den oben genannten Einstellungen.

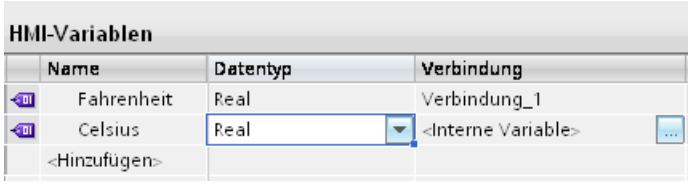

2. Erstellen Sie die benutzerdefinierte VB-Funktion "FahrenheitToCelsius" mit den oben genannten Einstellungen.

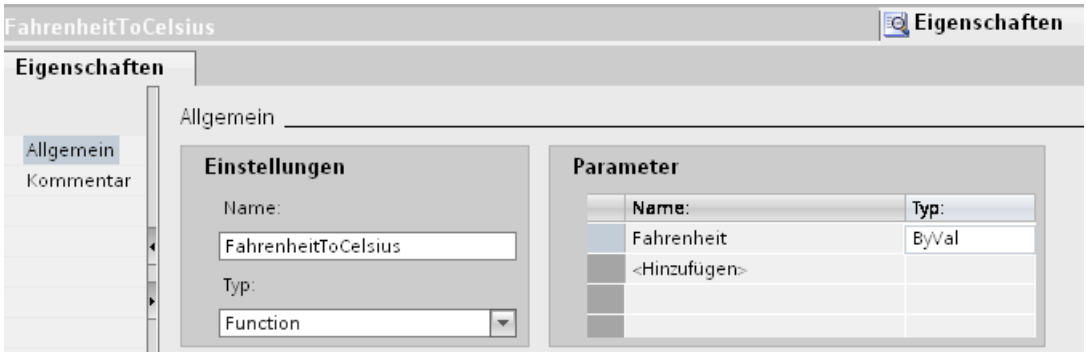

3. Schreiben Sie folgenden VBS-Code:
```
FahrenheitToCelsius = (Fahrenheit-32) * (5/9)
```

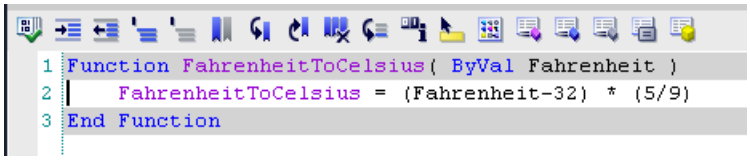

Zwischenergebnis

Die HMI-Variablen und die benutzerdefinierte VB-Funktion sind angelegt.

Vorgehen

1. Öffnen Sie den Editor "HMI-Variablen" und klicken Sie auf die Variable "Fahrenheit".
2. Klicken Sie im Inspektorfenster im Register "Eigenschaften > Ereignisse". Markieren Sie das Ereignis "Wertänderung".
3. Projektieren Sie die benutzerdefinierte VB-Funktion "FahrenheitToCelsius" an das Ereignis "Wertänderung".
4. Wählen Sie für den Parameter "Fahrenheit" die Variable "Fahrenheit" aus.
5. Wählen Sie für den Rückgabewert "FahrenheitToCelsius" die HMI-Variable "Celsius" aus.
6. Projektieren Sie in einem Bild ein Ausgabefeld und verbinden Sie es mit der Variablen "Celsius".

Alternatives Vorgehen

Statt einer benutzerdefinierten VB-Funktion vom Typ "Function" können Sie auch den Typ "Sub" verwenden. In diesem Fall weisen Sie den umgerechneten Wert direkt der HMI-Variablen "Celsius" zu:

```
SmartTags("Celsius") = (Fahrenheit-32) * (5/9)
```

Bei diesem Vorgehen entfällt der Rückgabewert der benutzerdefinierten VB-Funktion.

Ergebnis

Wenn sich in Runtime der Variablenwert von "Fahrenheit" ändert, wird die benutzerdefinierte VB-Funktion "FahrenheitToCelsius" ausgeführt. Der umgerechnete Temperaturwert wird an die HMI-Variable "Celsius" zurückgegeben und im Ausgabefeld angezeigt.

9.10.1.2 Beispiel: Umrechnen von Inch in Meter

Aufgabe

In diesem Beispiel erstellen Sie eine benutzerdefinierte VB-Funktion, die den Wert eines Längenmessgeräts von Inch in Meter umrechnet. Die Länge in Meter wird am Bediengerät in einem Ausgabefeld angezeigt.

Einstellungen

Für das Beispiel benötigen Sie zwei HMI-Variablen und eine benutzerdefinierte VB-Funktion mit folgenden Einstellungen:

HMI-Variablen:

Name	Steuerungsanbindung	Typ
VarInch	ja	Real
VarMeter	nein	Real

Benutzerdefinierte VB-Funktion

Name	Typ	Parameter
InchToMeter	Sub	Inch

Vorgehen

1. Erstellen Sie die beiden HMI-Variablen "VarInch" und "VarMeter" mit den oben genannten Einstellungen.

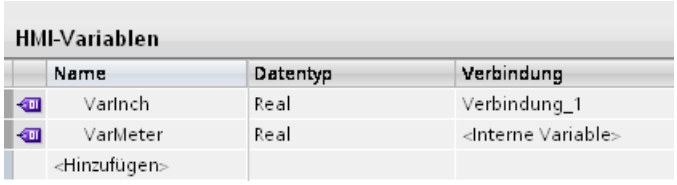

2. Erstellen Sie die benutzerdefinierte VB-Funktion "InchToMeter" mit den oben genannten Einstellungen.

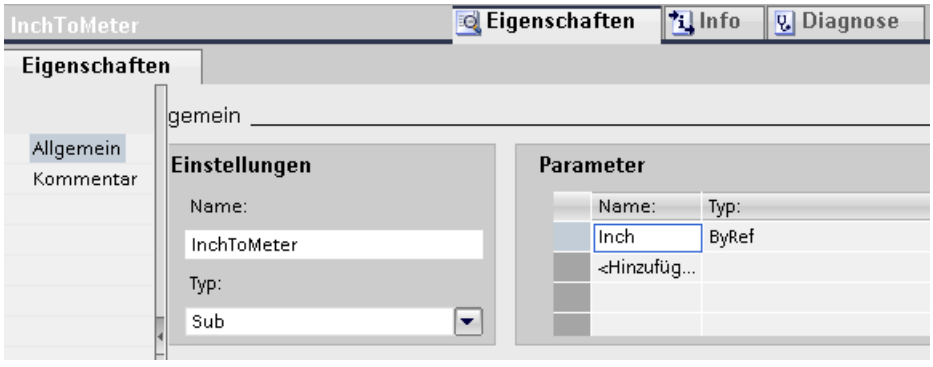

3. Schreiben Sie folgenden VBS-Code:
```
VarMeter = Inch*0.0254
```

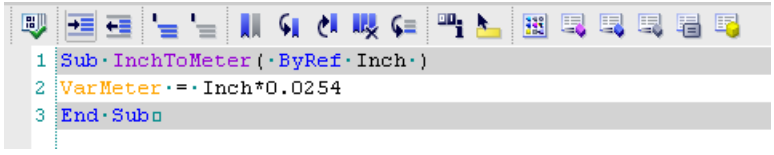

Zwischenergebnis

Die Variablen und die benutzerdefinierte VB-Funktion sind angelegt.

Vorgehen

1. Öffnen Sie den Editor "HMI-Variablen" und klicken Sie auf die Variable "VarInch".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Markieren Sie das Ereignis "Wertänderung".
3. Projektieren Sie die benutzerdefinierte VB-Funktion "InchToMeter" an das Ereignis "Wertänderung".
4. Wählen Sie für den Parameter "Inch" die HMI-Variable "VarInch" aus.
5. Projektieren Sie in einem Bild ein Ausgabefeld und verbinden Sie es mit der HMI-Variablen "VarMeter".

Ergebnis

Wenn sich in Runtime der Variablenwert von "VarInch" ändert, wird die benutzerdefinierte VB-Funktion "InchToMeter" ausgeführt. Der umgerechnete Wert wird in die HMI-Variable "VarMeter" geschrieben und im Ausgabefeld angezeigt.

9.10.1.3 Beispiel: Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige

Aufgabe

In diesem Beispiel verwenden Sie die Systemfunktion "SetzeBetriebsartGerät", um am Bediengerät zwischen den Betriebsarten "Online" und "Offline" zu wechseln. Sie lassen die aktuell eingestellte Betriebsart zusätzlich am Bediengerät anzeigen.

Voraussetzungen

Ein Prozessbild ist erstellt.

Einstellungen

Für das Beispiel benötigen Sie eine HMI-Variable und eine Textliste mit folgenden Einstellungen:

HMI-Variable:

Name	Steuerungsanbindung	Typ
OperatingMode	nein	Bool

Textliste:

Name	Enthält	Werte
ShowOperatingMode	Bit (0/1)	1: Betriebsart: "Online" 0: Betriebsart: "Offline"

Vorgehen

1. Erstellen Sie die oben genannte HMI-Variable "OperatingMode".

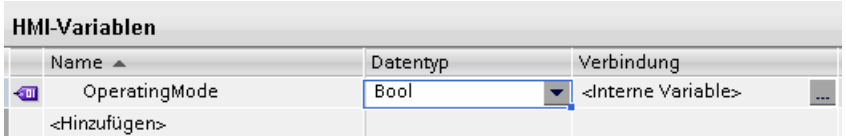

2. Erstellen Sie die oben genannte Textliste "ShowOperatingMode".

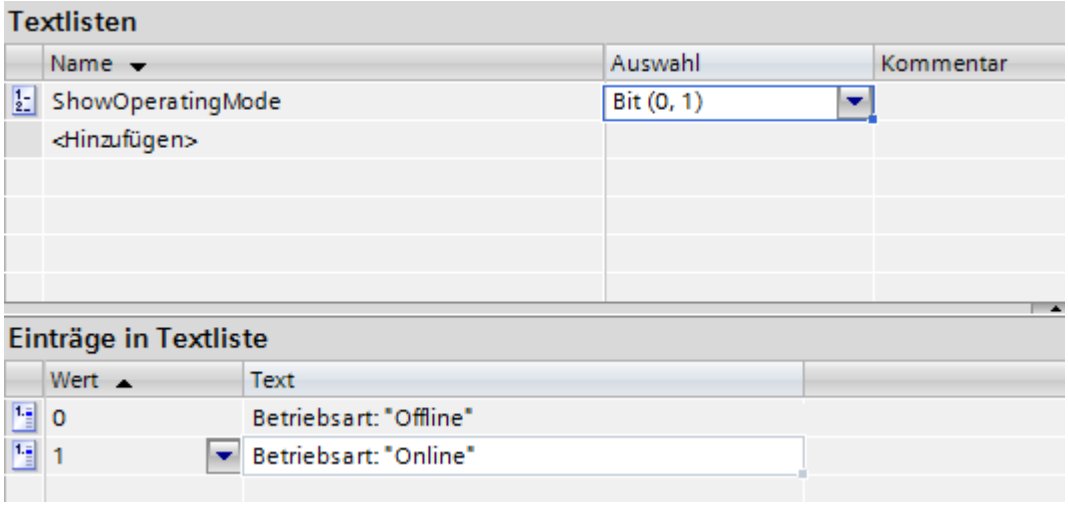

3. Öffnen Sie das Bild und fügen Sie eine Schaltfläche ein, an die Sie den Betriebsartenwechsel nach "Online" projektieren.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Markieren Sie das Ereignis "Drücken".
5. Projektieren Sie die Systemfunktion "SetzeBetriebsartGerät" an das Ereignis "Drücken". Die Systemfunktion finden Sie in der Auswahlliste unter "Einstellungen".
6. Wählen Sie für den Parameter "Betriebsart" den Eintrag "Online" aus.
7. Projektieren Sie die Systemfunktion "SetzeBit" an das Ereignis "Drücken". Die Systemfunktion finden Sie in der Auswahlliste unter "Bitverarbeitung".
8. Wählen Sie für den Parameter "Variable" aus der Auswahlliste die HMI-Variable "OperatingMode" aus.

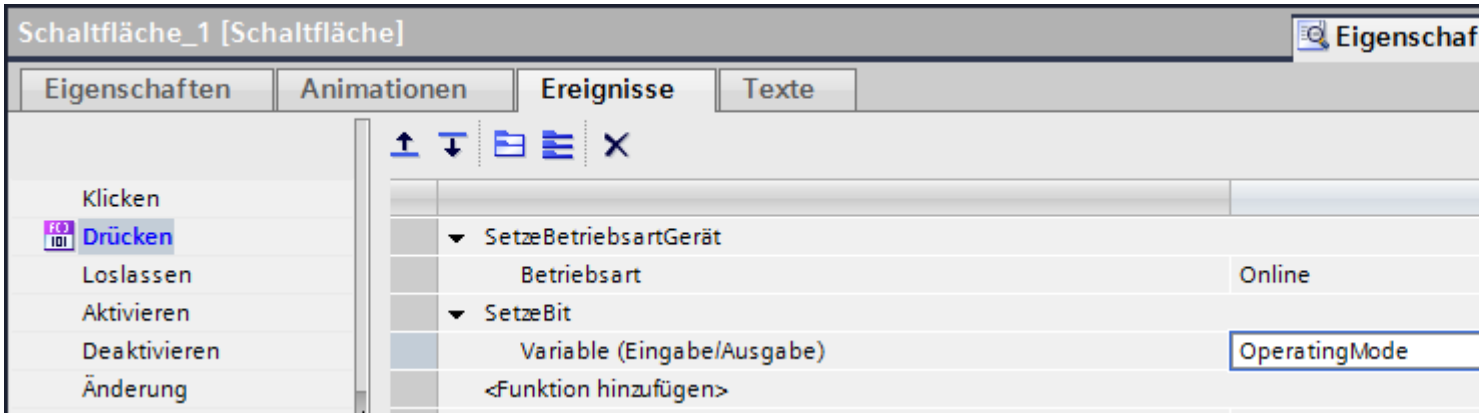

9. Fügen Sie im Prozessbild eine Schaltfläche ein, an die Sie den Betriebsartenwechsel nach "Offline" projektieren.
10. Wiederholen Sie die Schritte vier bis sieben. Wählen Sie für den Parameter "Betriebsart" den Eintrag "Offline". Projektieren Sie die Systemfunktion "RücksetzeBit" anstelle der Systemfunktion "SetzeBit".

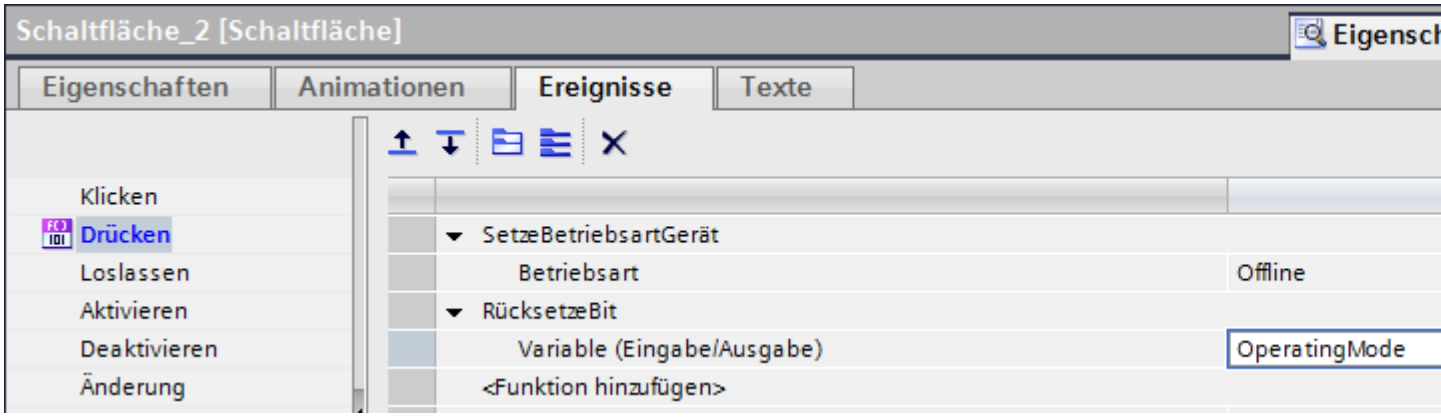

Zwischenergebnis

Sie können in Runtime mit den beiden Schaltflächen die Betriebsart des Bediengeräts umschalten.

Sie wollen die aktuell eingestellte Betriebsart in einem Ausgabefeld am Bediengerät anzeigen.

Vorgehen

1. Erstellen Sie im Prozessbild ein "Symbolisches E/A-Feld". Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften".
2. Nehmen sie in der Gruppe "Allgemein" folgende Einstellungen vor:
   - Wählen Sie als "Modus" "Ausgabe".
   - Wählen Sie als "Textliste" die Textliste "ShowOperatingMode".
   - Wählen Sie als "Variable" "OperatingMode".

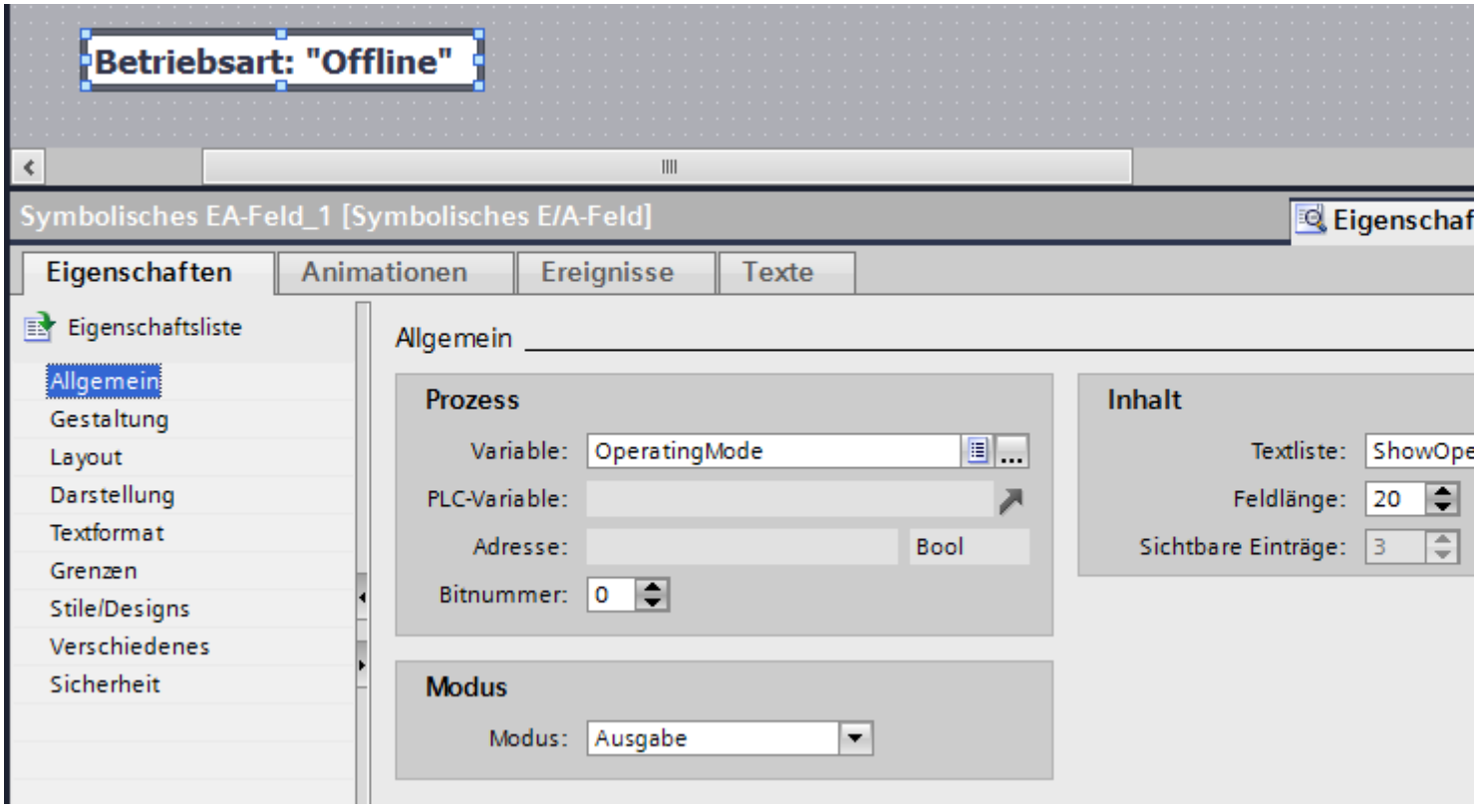

Ergebnis

Wenn Sie mit den Schaltflächen die Bedienart wechseln, wird stets die aktuell eingestellte Betriebsart am Bediengerät angezeigt.

Siehe auch

Funktionsliste projektieren (Seite 1440)

9.10.2 Beispiele zu VBS

9.10.2.1 Übersicht

Einleitung

Sie finden folgende Beispiele zu VBS:

- Zugriff auf Objekte im Editor "Bilder" (z. B. Farb- oder Textänderung)
- Farbe von Objekten bestimmen
- Sprachumschaltung projektieren
- Runtime deaktivieren
- Bildwechsel über Property projektieren
- Diagnoseausgabe über Trace projektieren
- Variablenwerte schreiben
- Variablenwerte lesen
- Objekteigenschaften schreiben
- Rezepturanzeige projektieren
- Meldeanzeige projektieren
- Kurve zeichnen
- Variablenwerte statistisch auswerten

Siehe auch

Beispiel für das Projektieren der Rezepturanzeige (Seite 1524)

Beispiel für das Projektieren der Meldeanzeige (Seite 1525)

Beispiel für das statistische Auswerten von Variablenwerten (Seite 1527)

9.10.2.2 Beispiel für das Zugreifen auf Objekte im Editor Bilder

Einleitung

Um die grafische Runtime-Umgebung zu dynamisieren, können Sie in WinCC mit VB-Scripting auf alle Objekte des Editors "Bilder" zugreifen. Dabei können Sie Grafikobjekte auf Bedienung dynamisieren, z. B. auf Mausklick oder in Abhängigkeit von Variablenwerten.

Vorgehen

Im folgenden Beispiel setzen Sie den Radius eines Kreises in Runtime bei Mausklick auf 20:

```
'VBS121
Dim objCircle
Set objCircle = ScreenItems("Circle1")
objCircle.Radius = 20
```

9.10.2.3 Beispiel für das Bestimmen von Farben von Objekten

Einleitung

Die Farben von Objekten sind über RGB-Werte definiert (Rot/Grün/Blau). Sie können Farbwerte für Objekte setzen oder auslesen.

Vorgehen

Das folgende Beispiel setzt die Füllfarbe des Bildes "ScreenWindow1" auf Blau:

```
'VBS122
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
objScreen.FillColor = RGB(0, 0, 255)
```

9.10.2.4 Beispiel für das Projektieren einer Sprachumschaltung

Einleitung

Sie können die Runtime-Sprache mit VBS umschalten. Verwenden Sie Schaltflächen mit entsprechenden Länderkennzeichnungen, die Sie auf der Startseite Ihres Projekts platzieren.

Die Runtime-Sprache geben Sie in VBS mit einer Länderkennung an, z. B. 1031 für Deutsch - Standard, 1033 für Englisch - USA etc. Eine Übersicht aller Länderkennungen finden Sie in den Grundlagen von VBScript unter dem Thema "Gebietsschema-ID (LCID)-Diagramm".

Vorgehen

Erstellen Sie am Ereignis "Mausklick" an einer Schaltfläche ein lokales VB-Skript und geben Sie folgenden Code ein, um die Runtime-Sprache auf Deutsch umzuschalten:

```
'VBS123
HMIRuntime.Language = 1031
```

9.10.2.5 Beispiel für das Deaktivieren der Runtime

Einleitung

Sie können mit VBS die Runtime beenden, z. B. auf Mausklick oder in Abhängigkeit von Variablenwerten oder anderen Ereignissen.

Vorgehen

Das folgende Beispiel beendet WinCC Runtime:

```
'VBS124
HMIRuntime.Stop
```

9.10.2.6 Beispiel für das Projektieren eines Bildwechsels über Property

Einleitung

Wenn Sie in Ihrer Projektierung aufgeteilte Bilder verwenden, z. B. in einem Grundbild Titel und Bedienleiste der Bedienoberfläche und ein eingebettetes Bildfenster für die eigentliche Bildanzeige, projektieren Sie den Bildwechsel über die Eigenschaften des Bildfensters.

Die Eigenschaft des Bildfensters "ScreenName" muss geändert werden, damit das andere Bild erscheint. Das lokale Skript und das Bildfenster müssen in demselben Bild projektiert sein.

Vorgehen

Im folgenden Beispiel zeigen Sie in einem Bildfenster "ScreenWindow" bei Ausführen des lokalen VB-Skripts das Bild "test" an:

```
'VBS126
Dim objScrWindow
Set objScrWindow = ScreenItems("ScreenWindow")
objScrWindow.ScreenName = "test"
```

9.10.2.7 Beispiel für das Projektieren einer Diagnoseausgabe über Trace

Einleitung

Wenn Sie das Objekt "Druckauftrag/Skriptdiagnose" in Ihr Bild eingefügt haben, können Sie Diagnoseausgaben mit dem Trace-Befehl in Runtime in dem Diagnosefenster ausgeben.

Druckauftrag/Skriptdiagnose gibt die in den benutzerdefinierten Funktionen enthaltenen Trace-Methoden in der zeitlichen Reihenfolge ihres Aufrufs aus. Das gilt auch für Trace-Anweisungen in Funktionen, die in benutzerdefinierten Funktionen aufgerufen werden. Durch gezielten Einsatz von Trace-Anweisungen, lässt sich der Ablauf von benutzerdefinierten Funktionen und

den darin aufgerufenen Funktionen verfolgen. Die Trace-Anweisung geben Sie in der Form "HMIRuntime.Trace <Ausgabe>" an.

In dem Ojekt "Druckauftrag/Skriptdiagnose" werden Trace-Ausgaben von C und VBS ausgegeben.

Text ausgeben

Das folgende Beispiel schreibt einen Text ins Diagnosefenster:

```
'VBS127
HMIRuntime.Trace "Customized error message"
```

Meldungen ausgeben

Das folgende Beispiel schreibt die Meldungen einer Meldeanzeige "Control2" in das Diagnosefenster:

```
Sub TraceMsg
  Dim objAlarmControl
  Dim lIndex
  Dim lCellIndex

  'create reference to the alarm control
  Set objAlarmControl = ScreenItems("Control2")

  'enumerate and trace out row numbers
  For lIndex = 1 To objAlarmControl.GetRowCollection.Count
     HMIRuntime.Trace "Row: " & (objAlarmControl.GetRow(lIndex).RowNumber) & "  "

     'enumerate and trace out column titles and cell texts
     For lCellIndex = 1 To objAlarmControl.GetRow(lIndex).CellCount
       HMIRuntime.Trace objAlarmControl.GetMessageColumn(lCellIndex -1).Name & " "
       HMIRuntime.Trace objAlarmControl.GetRow(lIndex).CellText(lCellIndex) & "  "
     Next
     HMIRuntime.Trace vbNewLine
  Next
End Sub
```

Informationen über HMI-Variablen ausgeben

Das folgende Beispiel schreibt Informationen über die Archivvariable "VarArch100" in das Diagnosefenster:

```
Sub ReadTagInfo
  Dim Tag
  Set Tag = HMIRuntime.Tags("TagLoggingfast\VarArch100")

  HMIRuntime.Trace "LastError = " + CStr(Tag.LastError) + vbNewLine
  HMIRuntime.Trace "Name = " + Tag.Name + vbNewLine
  HMIRuntime.Trace "QualityCode = " + CStr(Tag.QualityCode) + vbNewLine
End Sub
```

Siehe auch

Diagnose (Seite 1496)

9.10.2.8 Beispiel für das Schreiben von Variablenwerten

Einleitung

Sie können mit VBScript den Variablenwert in die Steuerung schreiben, z. B. auf Mausklick an einer Schaltfläche zur Sollwertvorgabe, oder interne Variablenwerte setzen, um andere benutzerdefinierte Funktionen auszulösen.

Nachfolgend werden verschiedene Schreibvarianten aufgeführt und erklärt.

Einfaches Schreiben

Im folgenden Beispiel wird ein Wert in die Variable "Tag1" geschrieben:

```
'VBS128
HMIRuntime.Tags("Tag1").Write 6
```

Dies ist die einfachste Form des Schreibens, da keine Objektreferenz erzeugt wird.

Schreiben mit Objektreferenz

Im folgenden Beispiel wird eine lokale Kopie des Tag-Objekts erzeugt und ein Wert in die Variable "Tag1" geschrieben:

```
'VBS129
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Write 7
```

Das Referenzieren bietet den Vorteil, dass Sie vor dem Schreiben mit dem Tag-Objekt arbeiten können. Sie können den Variablenwert lesen, Berechnungen ausführen und wieder schreiben:

```
'VBS130
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
objTag.Value = objTag.Value + 1
objTag.Write
```

Synchrones Schreiben

Standardmäßig wird der zu schreibende Wert an den Variablenhaushalt übergeben und die Abarbeitung des Skripts fortgesetzt. In manchen Fällen muss jedoch gewährleistet sein, dass der Wert auch tatsächlich geschrieben wurde, bevor das Skript weiter abgearbeitet wird.

Diese Schreibart realisieren Sie, indem Sie bei dem zusätzlichen, optionalen Parameter den Wert 1 angeben:

```
'VBS131
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Write 8,1
```

oder

```
'VBS132
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Value = 8
objTag.Write ,1
```

Hinweis

Beachten Sie, dass der Aufruf im Vergleich zum Standardaufruf länger dauert. Die Dauer ist unter anderem vom Kanal und dem AS abhängig.

Diese Schreibart entspricht dem SetTagXXXWait()-Aufruf aus dem C-Scripting.

Schreiben mit Statusbehandlung

Um sicherzustellen, dass ein Wert erfolgreich geschrieben wurde, führen Sie nach dem Schreiben eine Fehlerüberprüfung durch bzw. ermitteln den Status der Variablen.

Dies geschieht, indem nach dem Schreiben des Werts die Eigenschaft LastError überprüft wird. Wenn die Überprüfung erfolgreich war, d. h. der Auftrag erfolgreich abgesetzt werden konnte, wird der Status der Variablen überprüft.

Bei einem Schreibauftrag wird nicht der aktuelle Status aus dem Prozess ermittelt. Um diesen zu ermitteln ist es erforderlich, die Variable zu lesen. Der nach dem Lesevorgang in der Eigenschaft QualityCode enthaltene Wert gibt Aufschluss über den Status der Variablen und weist ggf. auf eine ausgefallene AS-Verbindung hin.

Im folgenden Beispiel wird die Variable "Tag1" geschrieben. Wenn es beim Schreiben zu einem Fehler kommt, werden Fehlerwert und Fehlerbeschreibung im Global Script Diagnosefenster ausgegeben. Anschließend wird der QualityCode überprüft. Wenn der QualityCode nicht OK (0x80) ist, wird er im Diagnosefenster ausgegeben.

```
'VBS133
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Write 9
If 0 <> objTag.LastError Then
  HMIRuntime.Trace "Error: " & objTag.LastError & vbCrLf & "ErrorDescription: " &
objTag.ErrorDescription & vbCrLf
Else
  objTag.Read
  If &H80 <> objTag.QualityCode Then
     HMIRuntime.Trace "QualityCode: 0x" & Hex(objTag.QualityCode) & vbCrLf
  End If
End If
```

Hinweis

Nach dem Schreiben einer Variablen wird die Eigenschaft QualityCode des lokalen Tag-Objekts auf "BAD Out of Service" gesetzt, da nicht bekannt ist, welchen QualityCode die Variable im Prozess führt. Durch ein erneutes Lesen kann der neue QualityCode bestimmt werden.

Der QualityCode kann aus VBS heraus nicht geschrieben werden.

9.10.2.9 Beispiel für das Lesen von Variablenwerten

Einleitung

Sie können mit VBS den Wert einer Variablen lesen und weiterverarbeiten. So können Sie z. B. auf Mausklick an einem Button etwas über den Zustand der Anlage erfahren oder Berechnungen ausführen.

Nachfolgend werden verschiedene Lesevarianten aufgeführt und erklärt.

Einfaches Lesen

Im folgenden Beispiel wird der Wert der Variablen "Tag1" gelesen und im Global Script Diagnosefenster ausgegeben:

```
'VBS134
HMIRuntime.Trace "Value: " & HMIRuntime.Tags("Tag1").Read & vbCrLf
```

Dies ist die einfachste Form des Lesens, da keine Objektreferenz erzeugt wird.

Lesen mit Objektreferenz

Im folgenden Beispiel wird eine lokale Kopie des Tag-Objekts erzeugt, der Variablenwert gelesen und Objekt "Druckauftrag/Skriptdiagnose" ausgegeben:

```
'VBS135
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
HMIRuntime.Trace "Value: " & objTag.Read & vbCrLf
```

Das Referenzieren bietet den Vorteil, dass Sie mit dem Tag-Objekt arbeiten können. Sie können den Variablenwert lesen, Berechnungen ausführen, und wieder schreiben:

```
'VBS136
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
objTag.Value = objTag.Value + 1
objTag.Write
```

Mit der Read-Methode gelesene Prozessvariablen werden dem Abbild hinzugefügt, sie werden von nun an zyklisch aus dem Automatisierungs-System (AS) angefordert. Wenn sich die Variable bereits im Abbild befindet, wird der darin enthaltene Wert geliefert.

Bei Bildabwahl werden die Variablen wieder abgemeldet.

Hinweis

Wenn eine Variable in einer Funktion angefordert wird, bleibt sie über die gesamte Runtime-Laufzeit von WinCC angemeldet.

Direktes Lesen

Standardmäßig werden Variablenwerte aus dem Variablenabbild gelesen. In manchen Situationen kann es jedoch erforderlich sein, den Wert direkt aus dem AS zu lesen, z. B. um schnelle Vorgänge zu synchronisieren.

Wenn Sie beim Lesen den optionalen Parameter auf 1 setzen, wird die Variable nicht zyklisch angemeldet, sondern der Wert wird einmalig aus dem AS angefordert.

```
'VBS137
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
HMIRuntime.Trace "Value: " & objTag.Read(1) & vbCrLf
```

Hinweis

Beachten Sie, dass der Aufruf im Vergleich zum Standardaufruf länger dauert. Die Dauer ist unter anderem vom Kanal und dem AS abhängig.

Vermeiden Sie diese Aufrufart in zyklischen C-Skripten, da dies der Hauptgrund für Performanceeinbußen ist.

Diese Leseart entspricht dem GetTagXXXWait()-Aufruf aus dem C-Scripting.

Lesen mit Statusbehandlung

Um sicherzustellen, dass ein Wert gültig ist, führen Sie nach dem Lesen eine Überprüfung durch. Dies geschieht, indem der QualityCode kontrolliert wird.

Im folgenden Beispiel wird die Variable "Tag1" gelesen und anschließend der QualityCode überprüft. Wenn der QualityCode nicht OK (0x80) entspricht, werden die Eigenschaften LastError, ErrorDescription und QualityCode im Global Script Diagnosefenster ausgegeben.

```
'VBS138
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
If &H80 <> objTag.QualityCode
  Then
    HMIRuntime.Trace "Error: " & objTag.LastError & vbCrLf & "ErrorDescription: " &
objTag.ErrorDescription & vbCrLf & "QualityCode: 0x" & Hex(objTag.QualityCode) &vbCrLf
  Else
    HMIRuntime.Trace "Value: " & objTag.Value & vbCrLf
End If
```

Hinweis

Wenn ein Fehler beim Lesen auftritt, wird der QualityCode auf "BAD Out of Service" gesetzt. Beim Lesen ist es daher ausreichend, den QualityCode zu prüfen.

9.10.2.10 Beispiel für das Schreiben von Objekteigenschaften

Einleitung

VBS bietet Zugriff auf die Eigenschaften aller Anzeige- und Bedienobjekte im Editor "Bilder". Sie können Eigenschaften auslesen oder zur Runtime verändern.

Die folgenden Beispiele zeigen Ihnen verschiedene Zugriffsformen.

Einfaches Setzen einer Eigenschaft

Im folgenden Beispiel wird die Hintergrundfarbe des im Bild enthaltenen Objekts "Rectangle1" auf rot gesetzt:

```
'VBS139
ScreenItems("Rectangle1").BackColor = RGB(255,0,0)
```

Dies ist die einfachste Form des Schreibens, da keine Objektreferenz erzeugt wird.

Hinweis

Wenn Sie ohne eine Objektreferenz arbeiten, werden von der Autovervollständigung nur die Standard-Eigenschaften aufgeführt.

Die in diesem Beispiel verwendete Ausdrucksweise ist nur im Editor "Bilder" ablauffähig. Analog adressieren Sie im Aufgabenplaner die Objekte über das HMIRuntime-Objekt.

Setzen einer Eigenschaft mit Objektreferenz

Im folgenden Beispiel wird eine Referenz auf das im Bild enthaltene Objekt "Rectangle1" erzeugt und mit Hilfe der VBS-Standardfunktion RGB() die Hintergrundfarbe auf rot gesetzt:

```
'VBS140
Dim objRectangle
Set objRectangle = ScreenItems("Rectangle1")
objRectangle.BackColor = RGB(255,0,0)
```

Nähere Hinweise zu RGB finden Sie in "VBScript für Windows".

Das Referenzieren bietet sich an, wenn Sie mehrere Eigenschaften eines Objekts ändern wollen. Auf diese Art bekommen Sie mit der Autovervollständigung alle Eigenschaften des Objekts aufgelistet.

Hinweis

Die in diesem Beispiel verwendete Ausdrucksweise ist nur im Editor "Bilder" ablauffähig. Analog adressieren Sie im Aufgabenplaner die Objekte über das HMIRuntime-Objekt.

Eigenschaften über Bildfenster hinweg setzen

VBS im Editor "Bilder" bietet zwei Möglichkeiten der bildübergreifenden Adressierung:

- über das Screen-Objekt eines Bildfensters mit "ScreenItems"
- vom Grundbild aus mit "HMIRuntime.Screens"

Referenzierung über das Bildfenster

Im folgendem Beispiel wird in einem unterlagerten Bildfenster die Farbe eines Rechtecks geändert. Die benuzterdefinierte VB-Funktion wird in dem Bild "BaseScreen" ausgeführt, in dem sich das Bildfenster "ScreenWindow1" befindet. Im Bildfenster wird ein Bild angezeigt, das ein Objekt vom Typ "Rechteck" mit dem Name "Rectangle1" enthält.

```
'VBS199
Sub OnLButtonUp(ByVal Item, ByVal Flags, ByVal x, ByVal y)
  Dim objRectangle
  Set objRectangle = ScreenItems("ScreenWindow1").Screen.ScreenItems("Rectangle1")
  objRectangle.BackColor = RGB(255,0,0)
End Sub
```

Referenzierung vom Grundbild aus

Sie können das Bild mit dem zu ändernden Objekt über HMIRuntime.Screens referenzieren. Die Angabe des Bildes erfolgt dabei relativ zum Grundbild über den folgenden Zugriffsschlüssel:

[<Grundbildname>.]<Bildfenstername>[:<Bildname>]... .<Bildfenstername>[:<Bildname>]

Im folgenden Beispiel wird eine Referenz auf das im Bild "Screen2" enthaltene Objekt "Rectangle1" erzeugt und die Hintergrundfarbe auf rot gesetzt.

Das Bild "Screen2" befindet sich dabei in "Screen1". "Screen1" wird im Grundbild "BaseScreen" angezeigt.

```
'VBS141
Dim objRectangle
Set objRectangle =
HMIRuntime.Screens("BaseScreen.ScreenWindow1:Screen1.ScreenWindow1:Screen2").ScreenItems("
Rectangle1")
objRectangle.BackColor = RGB(255,0,0)
```

Es ist nicht notwendig, die Bildnamen mit anzugeben. Über die Bildfensternamen ist es möglich, ein Bild eindeutig zu adressieren. Es reicht deshalb aus, die Namen der Bildfenster anzugeben, wie im nachfolgenden Beispiel:

```
'VBS142
Dim objRectangle
Set objRectangle =
HMIRuntime.Screens("ScreenWindow1.ScreenWindow2").ScreenItems("Rectangle1")
objRectangle.BackColor = RGB(255,0,0)
```

Über diese Art der Adressierung ist es möglich, Objekte in Bildfenstern mit wechselnden Bildern anzusprechen. Dies ist besonders für die Bildbausteintechnik interessant.

Eigenschaft über Rückgabewert dynamisieren

Sie können lokale Skripte an Eigenschaften nicht nur durch Ereignis-Trigger oder zyklische Trigger auslösen, sondern Eigenschaften auch direkt über ein lokales Skript dynamisieren.

Im folgenden Beispiel wird die Hintergrundfarbe eines Objekts über einen Rückgabewert dynamisiert. Der übergebene Wert kann z. B. aus einer Auswertung von Ereignissen in der Steuerung stammen und zur grafischen Darstellung eines Betriebszustandes eingesetzt werden:

```
'VBS146
Function BackColor_Trigger(ByVal Item)
  BackColor_Trigger = RGB(125,0,0)
End Function
```

Hinweis

Wenn Sie mit einer benutzerdefinierten Funktion eine Objekteigenschaft über den Rückgabewert einer benutzerdefinierten Funktion dynamisieren, wird der Wert der Objekteigenschaft nur dann geschrieben, wenn er sich im Vergleich zum letzten Funktions-Durchlauf geändert hat. Dabei wird nicht berücksichtigt, ob der Wert von einer anderen Stelle verändert wurde.

Deshalb dürfen Eigenschaften, die mit benutzerdefinierten Funktionen über den Rückgabewert dynamisiert wurden, nicht von anderer Stelle verändert werden, z. B. anderen benutzerdefinierten Funktionen. Wenn Sie dies nicht beachten, werden möglicherweise falsche Werte angezeigt.

9.10.2.11 Beispiel für das Projektieren der Rezepturanzeige

Einleitung

In der Rezepturanzeige werden Rezepturdatensätze ausgegeben.

Sie können Datensätze hinzufügen, kopieren, bearbeiten oder löschen.

Rezeptur laden und anzeigen

Das folgende Beispiel lädt die Rezeptur "Juice" in die Rezepturanzeige "Control1". Die Rezeptur wird in der Rezepturanzeige dargestellt:

```
Sub loadUserArchive()
  Dim objUAC
  Set objUAC = ScreenItems("Control1")
  objUAC.BorderColor = RGB(255,0,0)
  objUAC.BorderWidth = 3
  objUAC.Closeable = False
  objUAC.Datasource = "Juice"
End Sub
```

Rezepturdatensatz kopieren

Das folgende Beispiel kopiert den Rezepturdatensatz der Zeile 3. Die Kopie wird am Ende der Tabelle eingefügt.

```
Sub CopyRow
  Dim objUAC
  Set objUAC = ScreenItems("Control1")

  Dim iRowCount

  objUAC.AutoCompleteRows = True

  objUAC.SelectRow(3)
  objUAC.CopyRows

  iRowCount = objUAC.GetRowCollection.Count
  objUAC.SelectRow(iRowCount + 1)
  objUAC.PasteRows
  objUAC.SelectRow(1)
  objUAC.UnselectAll
End Sub
```

Siehe auch

Übersicht (Seite 1513)

9.10.2.12 Beispiel für das Projektieren der Meldeanzeige

Einleitung

Meldungen werden in der Meldeanzeige angezeigt. Sie können Meldungen ein- und ausblenden, sortieren und filtern.

Meldungen filtern

In diesem Beispiel werden in der Meldeanzeige "Control2" nur die Meldungen der Meldenummer 2 angezeigt.

```
Sub Filter
  Dim objAlarmControl

  'create reference to AlarmControl
  Set objAlarmControl = ScreenItems("Control2")

  'set / reset the filter and create a trace
  If (objAlarmControl.MsgFilterSQL = "") Then
     objAlarmControl.MsgFilterSQL = "MSGNR = 2"
  Else
     objAlarmControl.MsgFilterSQL = ""
  End If
End Sub
```

Meldungen entfernen

In diesem Beispiel werden die Meldungen 1-3 aus der Meldeanzeige entfernt.

```
Sub ResetAlarm
  Dim objAlarm
  'reset alarm 1 - 3
  Set objAlarm = HMIRuntime.Alarms(1)
  objAlarm.State = 2 'hmiAlarmStateGone
  objAlarm.Create
  Set objAlarm = HMIRuntime.Alarms(2)
  objAlarm.State = 2 'hmiAlarmStateGone
  objAlarm.Create
  Set objAlarm = HMIRuntime.Alarms(3)
  objAlarm.State = 2 'hmiAlarmStateGone
  objAlarm.Create
End Sub
```

Meldungen hinzufügen

In diesem Beispiel werden die Meldungen 1-3 in der Meldeanzeige hinzugefügt. Die Meldungen 1-3 müssen angelegt sein.

```
Sub SetAlarm
  Dim objAlarm

  'create Alarm 1 - 3
  Set objAlarm = HMIRuntime.Alarms(1)
  objAlarm.State = 1 'hmiAlarmStateCome
  objAlarm.Create
  Set objAlarm = HMIRuntime.Alarms(2)
  objAlarm.State = 1 'hmiAlarmStateCome
  objAlarm.Create
  Set objAlarm = HMIRuntime.Alarms(3)
  objAlarm.State = 1 'hmiAlarmStateCome
  objAlarm.Create
End Sub
```

Siehe auch

Übersicht (Seite 1513)

9.10.2.13 Beispiel für das statistische Auswerten von Variablenwerten

Einleitung

Mit der Wertetabelle können Sie die Koordinaten von Kurven und die statistische Auswertung von Variablenwerten tabellarisch darstellen. Dazu verbinden Sie die Wertetabelle mit dem entsprechenden Objekt. Die ausgewerteten Daten können ausgedruckt und exportiert werden.

Vorgehen

Im folgenden Beispiel werden die Variablenwerte der f(t)-Kurvenanzeige "Control2" statistisch ausgewertet. Um das Ergebnis im Statistikbereich-Fenster der Wertetabelle darzustellen, stellen Sie in Eigenschaften der Wertetabelle Folgendes ein:

- Datenquelle: "Control2"
- Mode: "Statistic area" window"

Fügen Sie in f(t)-Kurvenanzeige eine neue Schaltfläche hinzu und projektieren Sie daran die benutzerdefinierte VB-Funktion.

```
Sub OnToolbarButtonClicked(ByVal Item,  ByVal lId)
  Dim objTrend
  Set objTrend = ScreenItems("Control2")
  objTrend.CalculateStatistic()
End Sub
```

Siehe auch

Übersicht (Seite 1513)

9.10.2.14 Beispiel für das Starten eines Skripts am Server (Logging-Objekt)

Einleitung

In Mehrplatzprojekten funktioniert das Logging-Objekt derzeit nur am Server. In dem folgenden Beispiel wird gezeigt, wie Sie vom Client aus ein Skript am Server starten und somit auch am Client Archivsegmente einlagern und löschen können.

Das folgende Beispiel zeigt ein globales Skript, das mit einer Steuervariable gestartet wird. Der Inhalt der Steuervariable bestimmt, ob die Methode "Restore" oder die Methode "Remove" aufgerufen wird. Am Ende des Skripts wird die Steuervariable auf "0" gesetzt. Eine Abfrage verhindert, dass das Skript an Client-Rechnern gestartet wird. Pfad und Zeitraum werden durch interne Variablen übergeben.

```
'VBS180
'StartLoggingOnServer
Dim StartLogging
Dim SourcePath
Dim TimeFrom
Dim TimeTo
Dim RetVal
'Exit when running on client
If (Left(HMIRuntime.ActiveProject.Path, 1) = "\") Then
Exit Function
End If
'read parameters
StartLogging = HMIRuntime.Tags("StartLogging").Read
SourcePath = HMIRuntime.Tags("SourcePath").Read(1)
TimeFrom = HMIRuntime.Tags("TimeFrom").Read(1)
TimeTo = HMIRuntime.Tags("TimeTo").Read(1)
'restore or remove depends on the parameter
If (StartLogging = 1) Then
RetVal = HMIRuntime.Logging.Restore(SourcePath, TimeFrom, TimeTo, -1)
HMIRuntime.Tags("RetVal").Write RetVal, 1
HMIRuntime.Tags("StartLogging").Write 0,1
Elseif (StartLogging = 2) Then
RetVal = HMIRuntime.Logging.Remove(TimeFrom, TimeTo, -1)
HMIRuntime.Tags("RetVal").Write RetVal, 1
HMIRuntime.Tags("StartLogging").Write 0,1
End If
```

Als Pfadangabe kann auch eine Netzwerkfreigabe angegeben werden. Die einzulagernden Archivsegmente müssen also nicht lokal am Server liegen. Es muss jedoch gewährleistet sein, dass der Server korrekt auf den Pfad zugreifen kann.

Hinweis

Das Beispiel zeigt einen Lösungsvorschlag und ist bei Bedarf anzupassen.

Vorgehensweise

1. Legen Sie im WinCC Explorer die folgenden internen Variablen mit projektweiter Aktualisierung an:
   - StartLogging (Vorzeichenloser 8-Bit Wert)
   - SourcePath (Textvariable 8-Bit Zeichensatz)
   - TimeFrom (Textvariable 8-Bit Zeichensatz)
   - TimeTo (Textvariable 8-BBit Zeichensatz)
   - RetVal (Vorzeichenbehafteter 32-Bit Wert)
2. Erstellen Sie das Skript "StartLoggingOnServer".
3. Projektieren Sie an das Ereignis "Wertänderung" der Variablen "StartLogging" das Skript "StartLoggingOnServer".

Das Skript können auf einem Client z. B. mit folgendem Skript starten:

```
'VBS181
'Prerequisite: set parameters
HMIRuntime.Tags("SourcePath").Write "\\client_pc\temp",1
HMIRuntime.Tags("TimeFrom").Write "2004",1
HMIRuntime.Tags("TimeTo").Write "2004",1
'start action
HMIRuntime.Tags("StartLogging").Write 1,1
```

Hinweis

Die Variablen werden überwiegend im Modus "direkt" geschrieben und gelesen. Dadurch werden die Abläufe synchronisiert. Da es sich um interne Variablen handelt, kann dieser Modus bedenkenlos eingesetzt werden.

9.10.2.15 Beispiel für das Öffnen der PLC-Code-Anzeige aus der GRAPH-Übersicht

Einleitung

Mit VBS haben Sie die Möglichkeit, auch unter der Verwendung von Bildfenstern von der GRAPH-Übersicht in die PLC-Code-Anzeige zu wechseln.

Sie erstellen dafür einen VB-Skript an einer GRAPH-Übersicht, ein Bild mit einem Bildfenster sowie ein weiteres Bild mit einer PLC-Code-Anzeige, die den entsprechenden Programmcode anzeigt.

PLC-Code-Anzeige aus einem Bildfenster öffnen

Im folgenden Beispiel wird im Bildfenster "Screen window_1" aus der GRAPH-Übersicht die PLC-Code-Anzeige "PLC code view_1" im Bild "CodeViewScreen" mithilfe eines Ereignisses aufgerufen.

```
Sub OnPlcCodeViewButtonClick(ByVal item, ByVal codeViewerDiagnosticsContext)
  Dim objScreenWindow
  Set objScreenWindow = ScreenItems("Screen window_1")
  objScreenWindow.ScreenName = "CodeViewScreen"
  objScreenWindow.Screen.ScreenItems("PLC code view_1").NavigateTo =
codeViewerDiagnosticsContext
End Sub
```

9.10.3 Beispiele zu C

9.10.3.1 Übersicht

Einleitung

Sie finden Beispiele zu C-Funktionen:

- Zugriff auf Objekte im Editor "Bilder" (z. B. Farb- oder Textänderung)
- Farbe von Objekten bestimmen
- Sprachumschaltung projektieren
- Runtime deaktivieren
- Bildwechsel über Property projektieren
- Diagnoseausgabe über Trace projektieren
- Variablenwerte schreiben
- Variablenwerte lesen
- Objekteigenschaften schreiben

Siehe auch

Beispiel für das Zugreifen auf Objekte im Editor Bilder (Seite 1531)

Beispiel für das Bestimmen von Farben von Objekten (Seite 1531)

Beispiel für das Projektieren einer Sprachumschaltung (Seite 1532)

Beispiel für das Deaktivieren der Runtime (Seite 1532)

Beispiel für das Projektieren eines Bildwechsels über Property (Seite 1533)

Beispiel für das Projektieren einer Diagnoseausgabe über Trace (Seite 1533)

Beispiel für das Schreiben von Variablenwerten (Seite 1534)

Beispiel für das Lesen von Variablenwerten (Seite 1534)

Beispiel für das Schreiben von Objekteigenschaften (Seite 1535)

9.10.3.2 Beispiel für das Zugreifen auf Objekte im Editor Bilder

Einleitung

Um die grafische Runtime-Umgebung zu dynamisieren, können Sie in WinCC mit C-Scripting auf alle Objekte des Editors "Bilder" zugreifen. Dabei können Sie Grafikobjekte auf Bedienung dynamisieren, z. B. auf Mausklick oder in Abhängigkeit von Variablenwerten.

Vorgehen

Im folgenden Beispiel setzen Sie den Radius eines Kreises in Runtime bei Mausklick auf 10:

```
'C121
SetPropDouble(screenName,"Kreis_1","Radius",10);
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.3 Beispiel für das Bestimmen von Farben von Objekten

Einleitung

Die Farben von Objekten sind über RGB-Werte definiert (Rot/Grün/Blau). Sie können Farbwerte für Objekte setzen oder auslesen.

Vorgehen

Das folgende Beispiel setzt die Füllfarbe des Objekts "Roundbutton1" auf Blau:

```
'C122
SetPropertyByConstant (screenName, " Roundbutton1", "FillPatternColor", "65035");
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.4 Beispiel für das Projektieren einer Sprachumschaltung

Einleitung

Sie können die Runtime-Sprache mit C-Scripting umschalten. Verwenden Sie Schaltflächen mit entsprechenden Länderkennzeichnungen, die Sie auf der Startseite Ihres Projekts platzieren.

Die Runtime-Sprache geben Sie mit einer Länderkennung an, z. B. 1031 für Deutsch - Standard, 1033 für Englisch - USA etc. Eine Übersicht aller Länderkennungen finden Sie in den Grundlagen von VBScript unter dem Thema "Gebietsschema-ID (LCID)-Diagramm".

Vorgehen

Erstellen Sie am Ereignis "Mausklick" an einer Schaltfläche eine benutzerdefinierte C-Funktion und geben Sie folgenden Code ein, um die Runtime-Sprache auf Deutsch umzuschalten:

```
'C123
SetLanguageByLocaleID(1033);
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.5 Beispiel für das Deaktivieren der Runtime

Einleitung

Sie können mit C-Funktionen die Runtime beenden, z. B. auf Mausklick oder in Abhängigkeit von Variablenwerten oder anderen Ereignissen.

Vorgehen

Das folgende Beispiel beendet WinCC Runtime:

```
'C124
StopRuntime(hmiStopRuntime);
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.6 Beispiel für das Projektieren eines Bildwechsels über Property

Einleitung

Wenn Sie in Ihrer Projektierung aufgeteilte Bilder verwenden, z. B. in einem Grundbild Titel und Bedienleiste der Bedienoberfläche und ein eingebettetes Bildfenster für die eigentliche Bildanzeige, projektieren Sie den Bildwechsel über die Eigenschaften des Bildfensters.

Die Eigenschaft des Bildfensters "ScreenName" muss geändert werden, damit das andere Bild erscheint. Die C-Funktion und das Bildfenster müssen in demselben Bild projektiert sein.

Vorgehen

Im folgenden Beispiel zeigen Sie in einem Bildfenster "ScreenWindow" bei Ausführen der C-Funktion das Bild "test" an:

```
'C126
ActivateScreenInScreenWindow (screenName, "ScreenWindow", "test");
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.7 Beispiel für das Projektieren einer Diagnoseausgabe über Trace

Einleitung

Wenn Sie ein GSC-Diagnosefenster in Ihr Bild eingefügt haben, können Sie Diagnoseausgaben mit dem Trace-Befehl in Runtime in dem Diagnosefenster ausgeben.

GSC-Diagnose gibt die in den Funktionen enthaltenen Trace-Methoden in der zeitlichen Reihenfolge ihres Aufrufs aus. Das gilt auch für Trace-Anweisungen in Funktionen, die in benutzerdefinierten Funktionen aufgerufen werden. Durch gezielten Einsatz von Trace-Anweisungen lässt sich der Ablauf von benutzerdefinierten Funktionen und den darin aufgerufenen Funktionen verfolgen. Die Trace-Anweisung geben Sie in der Form "printf(<Ausgabe>)" an.

Im GSC-Diagnosefenster werden Trace-Ausgaben von C-Funktionen ausgegeben.

Text ausgeben

Das folgende Beispiel schreibt einen Text ins Diagnosefenster:

```
'C127
printf("Customized C script error message\r\n");
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.8 Beispiel für das Schreiben von Variablenwerten

Einleitung

Sie können mit C-Funktionen den Variablenwert in die Steuerung schreiben, z. B. auf Mausklick an einer Schaltfläche zur Sollwertvorgabe, oder interne Variablenwerte setzen, um andere Funktionen auszulösen.

Einfaches Schreiben

Im folgenden Beispiel wird ein Wert in die Variable "Tag1" geschrieben:

```
'C128
int value;
value = 7;
SetTagSDWord("Tag1", value);
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.9 Beispiel für das Lesen von Variablenwerten

Einleitung

Sie können mit C-Funktionen den Wert einer Variablen lesen und weiterverarbeiten. So können Sie z. B. auf Mausklick an einem Button etwas über den Zustand der Anlage erfahren oder Berechnungen ausführen.

Einfaches Lesen

Im folgenden Beispiel wird der Wert der Variablen "Tag1" gelesen und im Global Script Diagnosefenster ausgegeben:

```
'C134
int value;
value = GetTagGetTagSDWord("Tag1");
```

Siehe auch

Übersicht (Seite 1530)

9.10.3.10 Beispiel für das Schreiben von Objekteigenschaften

Einleitung

C-Funktionen bieten Ihnen Zugriff auf die Eigenschaften aller Anzeige- und Bedienobjekte im Editor "Bilder". Sie können Eigenschaften auslesen oder zur Runtime verändern.

Das folgende Beispiel zeigt Ihnen den Zugriff auf eine Eigenschaft.

Einfaches Setzen einer Eigenschaft

Im folgenden Beispiel wird die Hintergrundfarbe des im Bild enthaltenen Objekts "Button1" auf rot gesetzt:

```
'C139
SetPropLong("Screen_1","Button1","ForeColor",65333);
```

Siehe auch

Übersicht (Seite 1530)

9.11 Referenz

9.11.1 Funktionsliste

9.11.1.1 Geräteabhängigkeit

Systemfunktionen für Basic Panels

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Basic Panels.

Technische Änderungen vorbehalten.

Übersicht

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
Benutzerdefinierte Funktionen	nein	nein	nein	nein	nein
Abmelden (Seite 1612)	ja	ja	ja	ja	ja
AktiviereBild (Seite 1569)	ja	ja	ja	ja	ja
AktiviereBildMitNummer (Seite 1570)	ja	ja	ja	ja	ja
AktivierePutzbild (Seite 1717)	nein	nein	ja	ja	ja
AktiviereVorherigesBild (Seite 1571)	ja	ja	ja	ja	ja
AktualisiereVariable (Seite 1572)	ja	ja	ja	ja	ja
ÄndereKontrast (Seite 1651)	ja	nein	ja [1)]	ja	ja
Anmelden (Seite 1613)	ja	ja	ja	ja	ja
ArchiviereProtokolldatei (Seite 1603)	nein	nein	nein	nein	nein
ArchiviereVariable (Seite 1605)	nein	nein	nein	nein	nein
BearbeiteMeldung (Seite 1686)	ja	ja	ja	ja	nein
BildobjektCursorAbwärts (Seite 1721)	ja	ja	ja	ja	ja
BildobjektCursorAufwärts (Seite 1721)	ja	ja	ja	ja	ja
BildobjektCursorLinks (Seite 1723)	nein	nein	nein	nein	nein
BildobjektCursorRechts (Seite 1722)	nein	nein	nein	nein	nein
BildobjektSeiteAbwärts (Seite 1722)	ja	ja	ja	ja	ja
BildobjektSeiteAufwärts (Seite 1723)	ja	ja	ja	ja	ja
Codieren (Seite 1572)	nein	nein	nein	nein	nein
CodierenEx (Seite 1573)	nein	nein	nein	nein	nein
Direkttaste (Seite 1574)	nein	nein	nein	nein	nein

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
DirekttasteBildnummer (Seite 1577)	nein	nein	nein	nein	nein
DruckeBild (Seite 1638)	nein	nein	nein	nein	nein
DruckeProtokoll (Seite 1639)	nein	nein	nein	nein	nein
ErfasseBenutzeraktion (Seite 1577)	nein	nein	nein	nein	nein
ErhöheFokussiertenWert (Seite 1579)	ja	ja	ja	ja	nein
ErhöheVariable (Seite 1618)	ja	ja	ja	ja	ja
ExportiereDatensätze (Seite 1692)	ja	ja	ja	ja	ja
ExportiereDatensätzeMitPrüfsumme (Seite 1694)	nein	nein	nein	nein	nein
ExportiereImportiereBenutzerverwaltung (Seite 1613)	nein	nein	nein	nein	nein
GeheZumAnfang (Seite 1725)	ja	ja	ja	ja	nein
GeheZumEnde (Seite 1725)	ja	ja	ja	ja	nein
HardwareSicherEntfernen (Seite 1579)	nein	nein	nein	nein	nein
HTMLBrowserAbbrechen (Seite 1655)	nein	nein	nein	nein	nein
HTMLBrowserAbwärtsScrollen (Seite 1656)	nein	nein	nein	nein	nein
HTMLBrowserAktualisieren (Seite 1656)	nein	nein	nein	nein	nein
HTMLBrowserAufwärtsScrollen (Seite 1657)	nein	nein	nein	nein	nein
HTMLBrowserVorwärts (Seite 1660)	nein	nein	nein	nein	nein
HTMLBrowserZurück (Seite 1661)	nein	nein	nein	nein	nein
HTMLBrowserHeranzoomen (Seite 1657)	nein	nein	nein	nein	nein
HTMLBrowserHerauszoomen (Seite 1658)	nein	nein	nein	nein	nein
HTMLBrowserNachLinksScrollen (Seite 1658)	nein	nein	nein	nein	nein
HTMLBrowserNachRechtsScrollen (Seite 1658)	nein	nein	nein	nein	nein
HTMLBrowserSeiteNachOben (Seite 1659)	nein	nein	nein	nein	nein
HTMLBrowserSeiteNachUnten (Seite 1659)	nein	nein	nein	nein	nein
HTMLBrowserStartseite (Seite 1660)	nein	nein	nein	nein	nein
ImportiereDatensätze (Seite 1697)	ja	ja	ja	ja	ja
ImportiereDatensätzeMitPrüfsumme (Seite 1698)	nein	nein	nein	nein	nein
InvertiereBit (Seite 1624)	ja	ja	ja	ja	ja
InvertiereBitInVariable (Seite 1625)	ja	ja	ja	ja	ja
InvertiereLineareSkalierung (Seite 1619)	ja	ja	ja	ja	ja
KalibriereTouchscreen (Seite 1652)	nein	ja	ja	ja	ja
KopiereArchiv (Seite 1606)	nein	nein	nein	nein	nein
KurvenanzeigeBlättereVorwärts (Seite 1661)	ja	ja	ja	ja	ja
KurvenanzeigeBlättereZurück (Seite 1662)	ja	ja	ja	ja	ja
KurvenanzeigeErweitern (Seite 1662)	ja	ja	ja	ja	ja

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
KurvenanzeigeKomprimieren (Seite 1662)	ja	ja	ja	ja	ja
KurvenanzeigeLinealLinks (Seite 1663)	ja	ja	ja	ja	ja
KurvenanzeigeLinealRechts (Seite 1664)	ja	ja	ja	ja	ja
KurvenanzeigeSetzeLinealmodus (Seite 1664)	ja	ja	ja	ja	ja
KurvenanzeigeStartenStoppen (Seite 1665)	ja	ja	ja	ja	ja
KurvenanzeigeZurückZumAnfang (Seite 1665)	ja	ja	ja	ja	ja
LadeDatensatz (Seite 1700)	ja	ja	ja	ja	ja
LeseBenutzername (Seite 1615)	ja	ja	ja	ja	ja
LeseDatensatzAusSteuerung (Seite 1701)	ja	ja	ja	ja	ja
LeseDatensatzName (Seite 1702)	nein	nein	nein	nein	nein
LeseDatensatzvariablenAusSteuerung (Seite 1704)	ja	ja	ja	ja	ja
LeseGruppennummer (Seite 1615)	ja	ja	ja	ja	ja
LeseHelligkeit (Seite 1655)	nein	nein	nein	nein	nein
LeseKennwort (Seite 1616)	ja	ja	ja	ja	ja
LesePLCModus (Seite 1640)	ja	ja	ja	ja	ja
LineareSkalierung (Seite 1620)	ja	ja	ja	ja	ja
LöscheArchiv (Seite 1607)	nein	nein	nein	nein	nein
LöscheDatensatz (Seite 1705)	ja	ja	ja	ja	ja
LöscheDatensatzSpeicher (Seite 1706)	nein	nein	nein	nein	nein
LöscheMeldepuffer (Seite 1686)	ja	ja	ja	ja	ja
LöscheMeldepufferProTool (Seite 1687)	ja	ja	ja	ja	ja
MeldeanzeigeAktualisieren (Seite 1688)	nein	nein	nein	nein	nein
MeldeanzeigeLoopInAlarm (Seite 1688)	ja	ja	ja	ja	ja
MeldeanzeigeQuittiereMeldung (Seite 1689)	ja	ja	ja	ja	ja
MeldeanzeigeZeigeHilfetext (Seite 1690)	ja	ja	ja	ja	ja
ÖffneAlleArchive (Seite 1608)	nein	nein	nein	nein	nein
ÖffneBildschirmtastatur (Seite 1720)	nein	nein	nein	nein	nein
ÖffneDialogSystemsteuerung (Seite 1580)	ja [2)]	ja [2)]	ja [2)]	ja [2)]	ja [2)]
ÖffneEingabeaufforderung (Seite 1581)	nein	nein	nein	nein	nein
ÖffneInternetExplorer (Seite 1581)	nein	nein	nein	nein	nein
ÖffneSystemsteuerung (Seite 1582)	nein	nein	nein	nein	nein
ÖffneTaskManager (Seite 1719)	nein	nein	nein	nein	nein
QuittiereMeldung (Seite 1690)	ja	ja	ja	ja	nein
PDFAbwärtsScrollen (Seite 1666)	nein	nein	nein	nein	nein
PDFAufwärtsScrollen (Seite 1666)	nein	nein	nein	nein	nein
PDFAnpassenAufBreite (Seite 1667)	nein	nein	nein	nein	nein
PDFAnpassenAufHöhe (Seite 1667)	nein	nein	nein	nein	nein

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
PDFGeheZurErstenSeite (Seite 1668)	nein	nein	nein	nein	nein
PDFGeheZurLetztenSeite (Seite 1668)	nein	nein	nein	nein	nein
PDFGeheZurNächstenSeite (Seite 1669)	nein	nein	nein	nein	nein
PDFGeheZuSeite (Seite 1669)	nein	nein	nein	nein	nein
PDFGeheZurVorherigenSeite (Seite 1670)	nein	nein	nein	nein	nein
PDFHeranzoomen (Seite 1670)	nein	nein	nein	nein	nein
PDFHerauszoomen (Seite 1670)	nein	nein	nein	nein	nein
PDFLinksScrollen (Seite 1671)	nein	nein	nein	nein	nein
PDFRechtsScrollen (Seite 1671)	nein	nein	nein	nein	nein
PDFZoomOriginal (Seite 1672)	nein	nein	nein	nein	nein
RezepturanzeigeDatensatzNeu (Seite 1707)	ja	ja	ja	ja	ja
RezepturanzeigeLeseDatensatzAusSteuerung (Seite 1707)	ja	ja	ja	ja	ja
RezepturanzeigeLöscheDatensatz (Seite 1708)	ja	ja	ja	ja	ja
RezepturanzeigeMenü (Seite 1708)	ja	ja	ja	ja	ja
RezepturanzeigeÖffnen (Seite 1709)	ja	ja	ja	ja	ja
RezepturanzeigeSchreibeDatensatzInSteuerung (Seite 1710)	ja	ja	ja	ja	ja
RezepturanzeigeSpeichereDatensatz (Seite 1710)	ja	ja	ja	ja	ja
RezepturanzeigeSpeichernUnterDatensatz (Seite 1711)	ja	ja	ja	ja	ja
RezepturanzeigeSynchronisiereDatensatzMitVariablen (Seite 1711)	ja	ja	ja	ja	ja
RezepturanzeigeUmbenennenDatensatz (Seite 1712)	ja	ja	ja	ja	ja
RezepturanzeigeZeigeHilfetext (Seite 1712)	ja	ja	ja	ja	ja
RezepturanzeigeZurück (Seite 1712)	ja	ja	ja	ja	ja
RücksetzeBit (Seite 1627)	ja	ja	ja	ja	ja
RücksetzeBitInVariable (Seite 1629)	ja	ja	ja	ja	ja
SchaltflächeDrücken (Seite 1585)	ja	ja	ja	ja	nein
SchaltflächeLoslassen (Seite 1586)	ja	ja	ja	ja	nein
SchiebenUndMaskieren (Seite 1631)	ja	ja	ja	ja	ja
SchließeAlleArchive (Seite 1609)	nein	nein	nein	nein	nein
SchreibeDatensatzInSteuerung (Seite 1713)	ja	ja	ja	ja	ja
SchreibeDatensatzvariablenInSteuerung (Seite 1714)	ja	ja	ja	ja	ja
SeiteAbwärts (Seite 1724)	ja	ja	ja	ja	nein
SeiteAufwärts (Seite 1724)	ja	ja	ja	ja	nein

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
SendeEMail (Seite 1584)	nein	nein	nein	nein	nein
SetzeAkustischesSignal (Seite 1653)	nein	nein	nein	nein	nein
SetzeAnzeigemodus (Seite 1641)	nein	nein	nein	nein	nein
SetzeBetriebsartGerät (Seite 1642)	ja	ja	ja	ja	ja
SetzeBit (Seite 1633)	ja	ja	ja	ja	ja
SetzeBitInVariable (Seite 1634)	ja	ja	ja	ja	ja
SetzeBitWährendTasteGedrückt (Seite 1637)	ja	ja	ja	ja	ja
SetzeFarbeHintergrundbeleuchtung (Seite 1654)	ja	ja	nein	nein	nein
SetzeHelligkeit (Seite 1653)	nein	ja	ja	nein	nein
SetzeModusBildschirmtastatur (Seite 1642)	nein	nein	nein	nein	nein
SetzeProtokolliermodusMeldung (Seite 1639)	nein	nein	nein	nein	nein
SetzeRezepturvariablen (Seite 1715)	ja	ja	ja	ja	ja
SetzeSommerzeit (Seite 1584)	nein	nein	nein	nein	nein
SetzeSprache (Seite 1645)	ja	ja	ja	ja	ja
SetzeVariable (Seite 1621)	ja	ja	ja	ja	ja
SetzeVerbindungsmodus (Seite 1646)	ja	ja	ja	ja	ja
SetzeWebZugriff (Seite 1647)	nein	nein	nein	nein	nein
SichereRAMDateisystem (Seite 1587)	nein	nein	nein	nein	nein
SimuliereSystemtaste (Seite 1587)	ja	ja	ja	ja	nein
SimuliereVariable (Seite 1588)	ja	ja	ja	ja	ja
SmartClientAnzeigeAktualisieren (Seite 1679)	nein	nein	nein	nein	nein
SmartClientAnzeigeSchreibschutzAus (Seite 1680)	nein	nein	nein	nein	nein
SmartClientAnzeigeSchreibschutzEin (Seite 1680)	nein	nein	nein	nein	nein
SmartClientAnzeigeTrennen (Seite 1681)	nein	nein	nein	nein	nein
SmartClientAnzeigeVerbinden (Seite 1681)	nein	nein	nein	nein	nein
SmartClientAnzeigeVerlassen (Seite 1682)	nein	nein	nein	nein	nein
SpeichereDatensatz (Seite 1716)	ja	ja	ja	ja	ja
StarteArchivierung (Seite 1609)	nein	nein	nein	nein	nein
StarteNächstesArchiv (Seite 1610)	nein	nein	nein	nein	nein
StarteProgramm (Seite 1589)	nein	nein	nein	nein	nein
StatusSteuernLeseWerte (Seite 1591)	nein	nein	nein	nein	nein
StatusSteuernSchreibeWerte (Seite 1592)	nein	nein	nein	nein	nein
SteuereSmartServer (Seite 1592)	nein	nein	nein	nein	nein
SteuereWebServer (Seite 1593)	nein	nein	nein	nein	nein
StoppeArchivierung (Seite 1611)	nein	nein	nein	nein	nein

	KP300 Basic PN	KP400 Basic PN	KTP600 Basic PN / DP	KTP1000 Basic PN / DP	TP1500 Basic PN
StoppeRuntime (Seite 1593)	ja	ja	ja	ja	ja
SystemDiagnoseanzeigeAktualisierePLC-Buffer (Seite 1683)	ja	ja	ja	ja	ja
SystemdiagnoseAnzeigeAktivieren (Seite 1682)	nein	nein	nein	nein	nein
SystemDiagnoseanzeDetailansicht (Seite 1683)	ja	ja	ja	ja	ja
SystemDiagnoseanzeigeDiagnosepuffer (Seite 1684)	nein	nein	nein	nein	nein
SystemDiagnoseanzeigeGerätesicht (Seite 1684)	nein	nein	nein	nein	nein
SystemDiagnoseanzeigeZurück (Seite 1685)	ja	ja	ja	ja	ja
TextAbrufen (Seite 1594)	nein	nein	nein	nein	nein
TrenneVariableVonHandrad (Seite 1595)	nein	nein	nein	nein	nein
VerbindeVariableMitHandrad (Seite 1596)	nein	nein	nein	nein	nein
VerfolgeBenutzeränderung (Seite 1617)	ja	ja	ja	ja	ja
VerringereFokussiertenWert (Seite 1596)	ja	ja	ja	ja	nein
VerringereVariable (Seite 1623)	ja	ja	ja	ja	ja
WechseleVerbindung (Seite 1648)	ja	ja	ja	ja	ja
WechseleVerbindungEIP (Seite 1649)	nein	nein	nein	nein	nein
ZeigeAnmeldedialog (Seite 1617)	ja	ja	ja	ja	ja
ZeigeHilfetext (Seite 1597)	ja	ja	ja	ja	ja
ZeigeMeldefenster (Seite 1691)	ja	ja	ja	ja	ja
ZeigePopupBild (Seite 1598)	nein	nein	nein	nein	nein
ZeigePopupBildGrößeVeränderbar (Seite 1600)	nein	nein	nein	nein	nein
ZeigeSlideInBild (Seite 1602)	nein	nein	nein	nein	nein
ZeigeSoftwareVersion (Seite 1598)	nein	nein	nein	nein	nein
ZeigeSystemDiagnosefenster (Seite 1685)	nein	nein	nein	nein	nein
ZeigeSystemmeldung (Seite 1691)	nein	nein	nein	nein	nein

1) Nur für KTP600 Basic mono PN
2) Nur zum Ändern der IP-Einstellungen

Systemfunktionen für Basic Panels 2nd Generation

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Basic Panels.

Technische Änderungen vorbehalten.

Übersicht

	KTP400 Basic PN	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1200 Basic PN / DP
Benutzerdefinierte Funktionen	nein	nein	nein	nein
Abmelden (Seite 1612)	ja	ja	ja	ja
AktiviereBild (Seite 1569)	ja	ja	ja	ja
AktiviereBildMitNummer (Seite 1570)	ja	ja	ja	ja
AktivierePutzbild (Seite 1717)	ja	ja	ja	ja
AktiviereVorherigesBild (Seite 1571)	ja	ja	ja	ja
AktualisiereVariable (Seite 1572)	ja	ja	ja	ja
ÄndereKontrast (Seite 1651)	nein	nein	nein	nein
Anmelden (Seite 1613)	ja	ja	ja	ja
ArchiviereProtokolldatei (Seite 1603)	nein	nein	nein	nein
ArchiviereVariable (Seite 1605)	ja	ja	ja	ja
BearbeiteMeldung (Seite 1686)	ja	ja	ja	ja
BildobjektCursorAbwärts (Seite 1721)	ja	ja	ja	ja
BildobjektCursorAufwärts (Seite 1721)	ja	ja	ja	ja
BildobjektCursorLinks (Seite 1723)	ja	ja	ja	ja
BildobjektCursorRechts (Seite 1722)	ja	ja	ja	ja
BildobjektSeiteAbwärts (Seite 1722)	ja	ja	ja	ja
BildobjektSeiteAufwärts (Seite 1723)	ja	ja	ja	ja
Codieren (Seite 1572)	nein	nein	nein	nein
CodierenEx (Seite 1573)	nein	nein	nein	nein
Direkttaste (Seite 1574)	nein	nein	nein	nein
DirekttasteBildnummer (Seite 1577)	nein	nein	nein	nein
DruckeBild (Seite 1638)	nein	nein	nein	nein
DruckeProtokoll (Seite 1639)	nein	nein	nein	nein
ErfasseBenutzeraktion (Seite 1577)	nein	nein	nein	nein
ErhöheFokussiertenWert (Seite 1579)	ja	ja	ja	ja
ErhöheVariable (Seite 1618)	ja	ja	ja	ja
ExportiereDatensätze (Seite 1692)	ja	ja	ja	ja
ExportiereDatensätzeMitPrüfsumme (Seite 1694)	nein	nein	nein	nein
ExportiereImportiereBenutzerverwaltung (Seite 1613)	nein	nein	nein	nein
GeheZumAnfang (Seite 1725)	ja	ja	ja	ja
GeheZumEnde (Seite 1725)	ja	ja	ja	ja
HardwareSicherEntfernen (Seite 1579)	ja	ja	ja	ja
HTMLBrowserAbbrechen (Seite 1655)	ja	ja	ja	ja
HTMLBrowserAbwärtsScrollen (Seite 1656)	ja	ja	ja	ja
HTMLBrowserAktualisieren (Seite 1656)	ja	ja	ja	ja
HTMLBrowserAufwärtsScrollen (Seite 1657)	ja	ja	ja	ja

	KTP400 Basic PN	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1200 Basic PN / DP
HTMLBrowserHeranzoomen (Seite 1657)	ja	ja	ja	ja
HTMLBrowserHerauszoomen (Seite 1658)	ja	ja	ja	ja
HTMLBrowserNachLinksScrollen (Seite 1658)	ja	ja	ja	ja
HTMLBrowserNachRechtsScrollen (Seite 1658)	ja	ja	ja	ja
HTMLBrowserSeiteNachUnten (Seite 1659)	ja	ja	ja	ja
HTMLBrowserSeiteNachOben (Seite 1659)	ja	ja	ja	ja
HTMLBrowserStartseite (Seite 1660)	ja	ja	ja	ja
HTMLBrowserVorwärts (Seite 1660)	ja	ja	ja	ja
HTMLBrowserZurück (Seite 1661)	ja	ja	ja	ja
ImportiereDatensätze (Seite 1697)	ja	ja	ja	ja
ImportiereDatensätzeMitPrüfsumme (Seite 1698)	nein	nein	nein	nein
InvertiereBit (Seite 1624)	ja	ja	ja	ja
InvertiereBitInVariable (Seite 1625)	ja	ja	ja	ja
InvertiereLineareSkalierung (Seite 1619)	ja	ja	ja	ja
KalibriereTouchscreen (Seite 1652)	ja	ja	ja	ja
KopiereArchiv (Seite 1606)	nein	nein	nein	nein
KurvenanzeigeBlättereVorwärts (Seite 1661)	ja	ja	ja	ja
KurvenanzeigeBlättereZurück (Seite 1662)	ja	ja	ja	ja
KurvenanzeigeErweitern (Seite 1662)	ja	ja	ja	ja
KurvenanzeigeKomprimieren (Seite 1662)	ja	ja	ja	ja
KurvenanzeigeLinealLinks (Seite 1663)	ja	ja	ja	ja
KurvenanzeigeLinealRechts (Seite 1664)	ja	ja	ja	ja
KurvenanzeigeSetzeLinealmodus (Seite 1664)	ja	ja	ja	ja
KurvenanzeigeStartenStoppen (Seite 1665)	ja	ja	ja	ja
KurvenanzeigeZurückZumAnfang (Seite 1665)	ja	ja	ja	ja
LadeDatensatz (Seite 1700)	ja	ja	ja	ja
LeseBenutzername (Seite 1615)	ja	ja	ja	ja
LeseDatensatzAusSteuerung (Seite 1701)	ja	ja	ja	ja
LeseDatensatzName (Seite 1702)	nein	nein	nein	nein
LeseDatensatzvariablenAusSteuerung (Seite 1704)	ja	ja	ja	ja
LeseGruppennummer (Seite 1615)	ja	ja	ja	ja
LeseHelligkeit (Seite 1655)	nein	nein	nein	nein
LeseKennwort (Seite 1616)	nein [1)]	nein [1)]	nein [1)]	nein [1)]
LesePLCModus (Seite 1640)	ja	ja	ja	ja
LineareSkalierung (Seite 1620)	ja	ja	ja	ja
LöscheArchiv (Seite 1607)	ja	ja	ja	ja
LöscheDatensatz (Seite 1705)	ja	ja	ja	ja
LöscheDatensatzSpeicher (Seite 1706)	nein	nein	nein	nein
LöscheMeldepuffer (Seite 1686)	ja	ja	ja	ja

	KTP400 Basic PN	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1200 Basic PN / DP
LöscheMeldepufferProTool (Seite 1687)	ja	ja	ja	ja
MeldeanzeigeAktualisieren (Seite 1688)	nein	nein	nein	nein
MeldeanzeigeLoopInAlarm (Seite 1688)	ja	ja	ja	ja
MeldeanzeigeQuittiereMeldung (Seite 1689)	ja	ja	ja	ja
MeldeanzeigeZeigeHilfetext (Seite 1690)	ja	ja	ja	ja
ÖffneAlleArchive (Seite 1608)	ja	ja	ja	ja
ÖffneBildschirmtastatur (Seite 1720)	nein	nein	nein	nein
ÖffneDialogSystemsteuerung (Seite 1580)	ja [2)]	ja [2)]	ja [2)]	ja [2)]
ÖffneEingabeaufforderung (Seite 1581)	nein	nein	nein	nein
ÖffneInternetExplorer (Seite 1581)	nein	nein	nein	nein
ÖffneSystemsteuerung (Seite 1582)	nein	nein	nein	nein
ÖffneTaskManager (Seite 1719)	nein	nein	nein	nein
PDFAbwärtsScrollen (Seite 1666)	nein	nein	nein	nein
PDFAnpassenAufBreite (Seite 1667)	nein	nein	nein	nein
PDFAnpassenAufHöhe (Seite 1667)	nein	nein	nein	nein
PDFAufwärtsScrollen (Seite 1666)	nein	nein	nein	nein
PDFGeheZurErstenSeite (Seite 1668)	nein	nein	nein	nein
PDFGeheZurLetztenSeite (Seite 1668)	nein	nein	nein	nein
PDFGeheZurNächstenSeite (Seite 1669)	nein	nein	nein	nein
PDFGeheZurVorherigenSeite (Seite 1670)	nein	nein	nein	nein
PDFGeheZuSeite (Seite 1669)	nein	nein	nein	nein
PDFHeranzoomen (Seite 1670)	nein	nein	nein	nein
PDFHerauszoomen (Seite 1670)	nein	nein	nein	nein
PDFLinksScrollen (Seite 1671)	nein	nein	nein	nein
PDFRechtsScrollen (Seite 1671)	nein	nein	nein	nein
PDFZoomOriginal (Seite 1672)	nein	nein	nein	nein
QuittiereMeldung (Seite 1690)	ja	ja	ja	ja
RezepturanzeigeDatensatzNeu (Seite 1707)	ja	ja	ja	ja
RezepturanzeigeLeseDatensatzAusSteuerung (Seite 1707)	ja	ja	ja	ja
RezepturanzeigeLöscheDatensatz (Seite 1708)	ja	ja	ja	ja
RezepturanzeigeMenü (Seite 1708)	ja	ja	ja	ja
RezepturanzeigeÖffnen (Seite 1709)	ja	ja	ja	ja
RezepturanzeigeSchreibeDatensatzInSteuerung (Seite 1710)	ja	ja	ja	ja
RezepturanzeigeSpeichereDatensatz (Seite 1710)	ja	ja	ja	ja
RezepturanzeigeSpeichernUnterDatensatz (Seite 1711)	ja	ja	ja	ja
RezepturanzeigeSynchronisiereDatensatzMitVariablen (Seite 1711)	ja	ja	ja	ja

	KTP400 Basic PN	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1200 Basic PN / DP
RezepturanzeigeUmbenennenDatensatz (Seite 1712)	ja	ja	ja	ja
RezepturanzeigeZeigeHilfetext (Seite 1712)	ja	ja	ja	ja
RezepturanzeigeZurück (Seite 1712)	ja	ja	ja	ja
RücksetzeBit (Seite 1627)	ja	ja	ja	ja
RücksetzeBitInVariable (Seite 1629)	ja	ja	ja	ja
SchaltflächeDrücken (Seite 1585)	ja	ja	ja	ja
SchaltflächeLoslassen (Seite 1586)	ja	ja	ja	ja
SchiebenUndMaskieren (Seite 1631)	ja	ja	ja	ja
SchließeAlleArchive (Seite 1609)	ja	ja	ja	ja
SchreibeDatensatzInSteuerung (Seite 1713)	ja	ja	ja	ja
SchreibeDatensatzvariablenInSteuerung (Seite 1714)	ja	ja	ja	ja
SeiteAbwärts (Seite 1724)	ja	ja	ja	ja
SeiteAufwärts (Seite 1724)	ja	ja	ja	ja
SendeEMail (Seite 1584)	nein	nein	nein	nein
SetzeAkustischesSignal (Seite 1653)	nein	nein	nein	nein
SetzeAnzeigemodus (Seite 1641)	nein	nein	nein	nein
SetzeBetriebsartGerät (Seite 1642)	ja	ja	ja	ja
SetzeBit (Seite 1633)	ja	ja	ja	ja
SetzeBitInVariable (Seite 1634)	ja	ja	ja	ja
SetzeBitWährendTasteGedrückt (Seite 1637)	ja	ja	ja	ja
SetzeFarbeHintergrundbeleuchtung (Seite 1654)	nein	nein	nein	nein
SetzeHelligkeit (Seite 1653)	ja	ja	ja	ja
SetzeModusBildschirmtastatur (Seite 1642)	nein	nein	nein	nein
SetzeProtokolliermodusMeldung (Seite 1639)	nein	nein	nein	nein
SetzeRezepturvariablen (Seite 1715)	ja	ja	ja	ja
SetzeSommerzeit (Seite 1584)	nein	nein	nein	nein
SetzeSprache (Seite 1645)	ja	ja	ja	ja
SetzeVariable (Seite 1621)	ja	ja	ja	ja
SetzeVerbindungsmodus (Seite 1646)	ja	ja	ja	ja
SetzeWebZugriff (Seite 1647)	nein	nein	nein	nein
SichereRAMDateisystem (Seite 1587)	nein	nein	nein	nein
SimuliereSystemtaste (Seite 1587)	ja	ja	ja	ja
SimuliereVariable (Seite 1588)	ja	ja	ja	ja
SmartClientAnzeigeAktualisieren (Seite 1679)	nein	nein	nein	nein
SmartClientAnzeigeSchreibschutzAus (Seite 1680)	nein	nein	nein	nein
SmartClientAnzeigeSchreibschutzEin (Seite 1680)	nein	nein	nein	nein
SmartClientAnzeigeTrennen (Seite 1681)	nein	nein	nein	nein

	KTP400 Basic PN	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1200 Basic PN / DP
SmartClientAnzeigeVerbinden (Seite 1681)	nein	nein	nein	nein
SmartClientAnzeigeVerlassen (Seite 1682)	nein	nein	nein	nein
SpeichereDatensatz (Seite 1716)	ja	ja	ja	ja
StarteArchivierung (Seite 1609)	ja	ja	ja	ja
StarteNächstesArchiv (Seite 1610)	nein	nein	nein	nein
StarteProgramm (Seite 1589)	nein	nein	nein	nein
StatusSteuernLeseWerte (Seite 1591)	nein	nein	nein	nein
StatusSteuernSchreibeWerte (Seite 1592)	nein	nein	nein	nein
SteuereSmartServer (Seite 1592)	nein	nein	nein	nein
SteuereWebServer (Seite 1593)	nein	nein	nein	nein
StoppeArchivierung (Seite 1611)	ja	ja	ja	ja
StoppeRuntime (Seite 1593)	ja	ja	ja	ja
SystemdiagnoseAnzeigeAktivieren (Seite 1682)	nein	nein	nein	nein
SystemDiagnoseanzeigeAktualisierePLCBuffer (Seite 1683)	ja	ja	ja	ja
SystemDiagnoseanzeigeDetailansicht (Seite 1683)	ja	ja	ja	ja
SystemDiagnoseanzeigeDiagnosepuffer (Seite 1684)	nein	nein	nein	nein
SystemDiagnoseanzeigeGerätesicht (Seite 1684)	nein	nein	nein	nein
SystemDiagnoseanzeigeZurück (Seite 1685)	ja	ja	ja	ja
TextAbrufen (Seite 1594)	nein	nein	nein	nein
TrenneVariableVonHandrad (Seite 1595)	nein	nein	nein	nein
VerbindeVariableMitHandrad (Seite 1596)	nein	nein	nein	nein
VerfolgeBenutzeränderung (Seite 1617)	ja	ja	ja	ja
VerringereFokussiertenWert (Seite 1596)	ja	ja	ja	ja
VerringereVariable (Seite 1623)	ja	ja	ja	ja
WechseleVerbindung (Seite 1648)	ja	ja	ja	ja
WechseleVerbindungEIP (Seite 1649)	nein	nein	nein	nein
ZeigeAnmeldedialog (Seite 1617)	ja	ja	ja	ja
ZeigeHilfetext (Seite 1597)	ja	ja	ja	ja
ZeigeMeldefenster (Seite 1691)	ja	ja	ja	ja
ZeigePopUpBild (Seite 1598)	nein	nein	nein	nein
ZeigePopupBildGrößeVeränderbar (Seite 1600)	nein	nein	nein	nein
ZeigeSlideInBild (Seite 1602)	nein	nein	nein	nein
ZeigeSoftwareVersion (Seite 1598)	nein	nein	nein	nein
ZeigeSystemDiagnosefenster (Seite 1685)	nein	nein	nein	nein
ZeigeSystemmeldung (Seite 1691)	nein	nein	nein	nein

[1] Für Geräte ab der Geräteversion V14.0 nicht verfügbar
[2] Nur zum Ändern der IP-Einstellungen

Systemfunktionen für Comfort Panels

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Comfort Panels.

Technische Änderungen vorbehalten.

Übersicht

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
Benutzerdefinierte Funktionen	ja	ja	ja
Abmelden (Seite 1612)	ja	ja	ja
AktiviereBild (Seite 1569)	ja	ja	ja
AktiviereBildMitNummer (Seite 1570)	ja	ja	ja
AktivierePLCCodeAnzeige (Seite 1672)	nein	ja	ja
AktivierePutzbild (Seite 1717)	ja [1]	ja	nein
AktiviereVorherigesBild (Seite 1571)	ja	ja	ja
AktualisiereVariable (Seite 1572)	ja	ja	ja
ÄndereKontrast (Seite 1651)	nein	nein	nein
Anmelden (Seite 1613)	ja	ja	ja
ArchiviereProtokolldatei (Seite 1603)	ja	ja	ja
ArchiviereVariable (Seite 1605)	ja	ja	ja
BearbeiteMeldung (Seite 1686)	ja	nein	ja
BildobjektCursorAbwärts (Seite 1721)	ja	ja	ja
BildobjektCursorAufwärts (Seite 1721)	ja	ja	ja
BildobjektCursorLinks (Seite 1723)	nein	ja	ja
BildobjektCursorRechts (Seite 1722)	nein	ja	ja
BildobjektSeiteAbwärts (Seite 1722)	ja	ja	ja
BildobjektSeiteAufwärts (Seite 1723)	ja	ja	ja
Codieren (Seite 1572)	ja	ja	ja
CodierenEx (Seite 1573)	ja	ja	ja
Direkttaste (Seite 1574)	ja [1]	ja	nein

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
DirekttasteBildnummer (Seite 1577)	ja [1)]	ja	nein
DruckeBild (Seite 1638)	ja	ja	ja
DruckeProtokoll (Seite 1639)	ja	ja	ja
ErfasseBenutzeraktion (Seite 1577)	ja	ja	ja
ErhöheFokussiertenWert (Seite 1579)	ja	nein	ja
ErhöheVariable (Seite 1618)	ja	ja	ja
ExportiereDatensätze (Seite 1692)	ja	ja	ja
ExportiereDatensätzeMitPrüfsumme (Seite 1694)	ja	ja	ja
ExportiereImportiereBenutzerverwaltung (Seite 1613)	ja	ja	ja
GeheZumAnfang (Seite 1725)	ja	nein	ja
GeheZumEnde (Seite 1725)	ja	nein	ja
HardwareSicherEntfernen (Seite 1579)	ja	ja	ja
HTMLBrowserAbbrechen (Seite 1655)	nein	ja	ja
HTMLBrowserAbwärtsScrollen (Seite 1656)	nein	ja	ja
HTMLBrowserAktualisieren (Seite 1656)	nein	ja	ja
HTMLBrowserAufwärtsScrollen (Seite 1657)	nein	ja	ja
HTMLBrowserVorwärts (Seite 1660)	nein	ja	ja
HTMLBrowserZurück (Seite 1661)	nein	ja	ja
HTMLBrowserHeranzoomen (Seite 1657)	nein	ja	ja
HTMLBrowserHerauszoomen (Seite 1658)	nein	ja	ja
HTMLBrowserNachLinksScrollen (Seite 1658)	nein	ja	ja
HTMLBrowserNachRechtsScrollen (Seite 1658)	nein	ja	ja
HTMLBrowserSeiteNachUnten (Seite 1659)	nein	ja	ja
HTMLBrowserSeiteNachOben (Seite 1659)	nein	ja	ja
HTMLBrowserStartseite (Seite 1660)	nein	ja	ja
AUTOHOTSPOT	nein	ja	ja
ImportiereDatensätze (Seite 1697)	ja	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
ImportiereDatensätzeMitPrüfsumme (Seite 1698)	ja	ja	ja
InvertiereBit (Seite 1624)	ja	ja	ja
InvertiereBitInVariable (Seite 1625)	ja	ja	ja
InvertiereLineareSkalierung (Seite 1619)	ja	ja	ja
KalibriereTouchscreen (Seite 1652)	ja [1)]	ja	nein
KopiereArchiv (Seite 1606)	ja	ja	ja
KurvenanzeigeBlättereVorwärts (Seite 1661)	ja	ja	ja
KurvenanzeigeBlättereZurück (Seite 1662)	ja	ja	ja
KurvenanzeigeErweitern (Seite 1662)	ja	ja	ja
KurvenanzeigeKomprimieren (Seite 1662)	ja	ja	ja
KurvenanzeigeLinealLinks (Seite 1663)	ja	ja	ja
KurvenanzeigeLinealRechts (Seite 1664)	ja	ja	ja
KurvenanzeigeSetzeLinealmodus (Seite 1664)	ja	ja	ja
KurvenanzeigeStartenStoppen (Seite 1665)	ja	ja	ja
KurvenanzeigeZurückZumAnfang (Seite 1665)	ja	ja	ja
LadeDatensatz (Seite 1700)	ja	ja	ja
LeseBenutzername (Seite 1615)	ja	ja	ja
LeseDatensatzAusSteuerung (Seite 1701)	ja	ja	ja
LeseDatensatzname (Seite 1702)	ja	ja	ja
LeseDatensatzvariablenAusSteuerung (Seite 1704)	ja	ja	ja
LeseGruppennummer (Seite 1615)	ja	ja	ja
LeseHelligkeit (Seite 1655)	ja	ja	ja
LeseKennwort (Seite 1616)	nein [2)]	nein [2)]	nein [2)]
LesePLCModus (Seite 1640)	ja	ja	ja
LineareSkalierung (Seite 1620)	ja	ja	ja
LöscheArchiv (Seite 1607)	ja	ja	ja
LöscheDatensatz (Seite 1705)	ja	ja	ja
LöscheDatensatzSpeicher (Seite 1706)	ja	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
LöscheMeldepuffer (Seite 1686)	ja	ja	ja
LöscheMeldepufferProTool (Seite 1687)	ja	ja	ja
MeldeanzeigeAktualisieren (Seite 1688)	ja	ja	ja
MeldeanzeigeLoopInAlarm (Seite 1688)	ja	ja	ja
MeldeanzeigeQuittiereMeldung (Seite 1689)	ja	ja	ja
MeldeanzeigeZeigeHilfetext (Seite 1690)	ja	ja	ja
ÖffneAlleArchive (Seite 1608)	ja	ja	ja
ÖffneBildschirmtastatur (Seite 1720)	ja	ja	ja
ÖffneDateiBrowser (Seite 1718)	ja	ja	ja
ÖffneDialogSystemsteuerung (Seite 1580)	ja	ja	ja
ÖffneEingabeaufforderung (Seite 1581)	ja	ja	ja
ÖffneInternetExplorer (Seite 1581)	ja	ja	ja
ÖffneSystemsteuerung (Seite 1582)	ja	ja	ja
ÖffneTaskManager (Seite 1719)	ja	ja	ja
QuittiereMeldung (Seite 1690)	ja	nein	ja
PDFAbwärtsScrollen (Seite 1666)	ja	ja	ja
PDFAufwärtsScrollen (Seite 1666)	ja	ja	ja
PDFAnpassenAufBreite (Seite 1667)	ja	ja	ja
PDFAnpassenAufHöhe (Seite 1667)	ja	ja	ja
PDFGeheZurErstenSeite (Seite 1668)	ja	ja	ja
PDFGeheZurLetztenSeite (Seite 1668)	ja	ja	ja
PDFGeheZurNächstenSeite (Seite 1669)	ja	ja	ja
PDFGeheZuSeite (Seite 1669)	ja	ja	ja
PDFGeheZurVorherigenSeite (Seite 1670)	ja	ja	ja
PDFHeranzoomen (Seite 1670)	ja	ja	ja
PDFHerauszoomen (Seite 1670)	ja	ja	ja
PDFLinksScrollen (Seite 1671)	ja	ja	ja
PDFRechtsScrollen (Seite 1671)	ja	ja	ja
PDFZoomOriginal (Seite 1672)	ja	ja	ja
PLCCodeAnzeigeDetails (Seite 1675)	nein	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
PLCCodeAnzeigeNächstesNetzwerk (Seite 1675)	nein	ja	ja
PLCCodeAnzeigeZeigeSchrittModus (Seite 1676)	nein	ja	ja
PLCCodeAnzeigeSymbole (Seite 1676)	nein	ja	ja
PLCCodeAnzeigeTransitionInterlock (Seite 1677)	nein	ja	ja
PLCCodeAnzeigeHeranzoomen (Seite 1678)	nein	ja	ja
PLCCodeAnzeigeHerauszoomen (Seite 1678)	nein	ja	ja
PLCCodeAnzeigeVorherigesNetzwerk (Seite 1679)	nein	ja	ja
AUTOHOTSPOT	nein	ja	ja
RezepturanzeigeDatensatzNeu (Seite 1707)	ja	ja	ja
RezepturanzeigeLeseDatensatzAusSteuerung (Seite 1707)	ja	ja	ja
RezepturanzeigeLöscheDatensatz (Seite 1708)	ja	ja	ja
RezepturanzeigeMenü (Seite 1708)	ja	ja	ja
RezepturanzeigeÖffnen (Seite 1709)	ja	ja	ja
RezepturanzeigeSchreibeDatensatzInSteuerung (Seite 1710)	ja	ja	ja
RezepturanzeigeSpeichereDatensatz (Seite 1710)	ja	ja	ja
RezepturanzeigeSpeichernUnterDatensatz (Seite 1711)	ja	ja	ja
RezepturanzeigeSynchronisiereDatensatzMitVariablen (Seite 1711)	ja	ja	ja
RezepturanzeigeUmbenennenDatensatz (Seite 1712)	ja	ja	ja
RezepturanzeigeZeigeHilfetext (Seite 1712)	ja	ja	ja
RezepturanzeigeZurück (Seite 1712)	ja	ja	ja
RücksetzeBit (Seite 1627)	ja	ja	ja
RücksetzeBitInVariable (Seite 1629)	ja	ja	ja
SchaltflächeDrücken (Seite 1585)	ja	ja	ja
SchaltflächeLoslassen (Seite 1586)	ja	ja	ja
SchiebenUndMaskieren (Seite 1631)	ja	ja	ja
SchließeAlleArchive (Seite 1609)	ja	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
SchreibeDatensatzInSteuerung (Seite 1713)	ja	ja	ja
SchreibeDatensatzvariablenInSteuerung (Seite 1714)	ja	ja	ja
SeiteAbwärts (Seite 1724)	ja	nein	ja
SeiteAufwärts (Seite 1724)	ja	nein	ja
SendeEMail (Seite 1584)	ja	ja	ja
SetzeAkustischesSignal (Seite 1653)	ja [1)]	ja	nein
SetzeAnzeigemodus (Seite 1641)	nein	nein	nein
SetzeBetriebsartGerät (Seite 1642)	ja	ja	ja
SetzeBit (Seite 1633)	ja	ja	ja
SetzeBitInVariable (Seite 1634)	ja	ja	ja
SetzeBitWährendTasteGedrückt (Seite 1637)	ja	ja	ja
SetzeFarbeHintergrundbeleuchtung (Seite 1654)	nein	nein	nein
SetzeHelligkeit (Seite 1653)	ja	ja	ja
SetzeModusBildschirmtastatur (Seite 1642)	ja	ja	ja
SetzePLCModus (Seite 1644)	ja	ja	ja
SetzeProtokolliermodusMeldung (Seite 1639)	ja	ja	ja
SetzeRezepturvariablen (Seite 1715)	ja	ja	ja
SetzeSommerzeit (Seite 1584)	ja	ja	ja
SetzeSprache (Seite 1645)	ja	ja	ja
SetzeVariable (Seite 1621)	ja	ja	ja
SetzeVerbindungsmodus (Seite 1646)	ja	ja	ja
SetzeWebZugriff (Seite 1647)	ja	ja	ja
SichereRAMDateisystem (Seite 1587)	ja	ja	ja
SimuliereSystemtaste (Seite 1587)	ja	nein	ja
SimuliereVariable (Seite 1588)	ja	ja	ja
SmartClientAnzeigeAktualisieren (Seite 1679)	ja	ja	ja
SmartClientAnzeigeSchreibschutzAus (Seite 1680)	ja	ja	ja
SmartClientAnzeigeSchreibschutzEin (Seite 1680)	ja	ja	ja
SmartClientAnzeigeTrennen (Seite 1681)	ja	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
SmartClientAnzeigeVerbinden (Seite 1681)	ja	ja	ja
SmartClientAnzeigeVerlassen (Seite 1682)	ja	ja	ja
SpeichereDatensatz (Seite 1716)	ja	ja	ja
StarteArchivierung (Seite 1609)	ja	ja	ja
StarteNächstesArchiv (Seite 1610)	ja	ja	ja
StarteProgramm (Seite 1589)	ja	ja	ja
StatusSteuernLeseWerte (Seite 1591)	ja	ja	ja
StatusSteuernSchreibeWerte (Seite 1592)	ja	ja	ja
SteuereSmartServer (Seite 1592)	ja	ja	ja
SteuereWebServer (Seite 1593)	ja	ja	ja
StoppeArchivierung (Seite 1611)	ja	ja	ja
StoppeRuntime (Seite 1593)	ja	ja	ja
SystemdiagnoseAnzeigeAktivieren (Seite 1682)	ja	ja	ja
SystemDiagnoseanzeigeAktualisierePLCBuffer (Seite 1683)	ja	ja	ja
SystemDiagnoseanzeigeDetailansicht (Seite 1683)	ja	ja	ja
SystemDiagnoseanzeigeDiagnosepuffer (Seite 1684)	ja	ja	ja
SystemDiagnoseanzeigeGerätesicht (Seite 1684)	ja	ja	ja
SystemDiagnoseanzeigeZurück (Seite 1685)	ja	ja	ja
TextAbrufen (Seite 1594)	ja	ja	ja
TrenneVariableVonHandrad (Seite 1595)	nein	nein	nein
VerbindeVariableMitHandrad (Seite 1596)	nein	nein	Nein
VerfolgeBenutzeränderung (Seite 1617)	ja	ja	ja
VerringereFokussiertenWert (Seite 1596)	ja	nein	ja
VerringereVariable (Seite 1623)	ja	ja	ja
WechseleVerbindung (Seite 1648)	ja	ja	ja
WechseleVerbindungEIP (Seite 1649)	ja	ja	ja
ZeigeAnmeldedialog (Seite 1617)	ja	ja	ja

	KTP400 KP400	TP700 TP900 TP1200 TP1500 TP1900 TP2200	KP700 KP900 KP1200 KP1500
ZeigeHilfetext (Seite 1597)	ja	ja	ja
ZeigeMeldefenster (Seite 1691)	ja	ja	ja
ZeigePopupBild (Seite 1598)	ja	ja	ja
ZeigePopupBildGrößeVeränderbar (Seite 1600)	ja	ja	ja
ZeigeSlideInBild (Seite 1602)	ja	ja	ja
ZeigeSoftwareVersion (Seite 1598)	ja	ja	ja
ZeigeSystemDiagnosefenster (Seite 1685)	ja	ja	ja
ZeigeSystemmeldung (Seite 1691)	ja	ja	ja

1) Nur am KTP 400 Comfort
2) Für Geräte ab der Geräteversion V14.0 nicht verfügbar

Systemfunktionen für Mobile Panels

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen auf den Mobile Panels.

Technische Änderungen vorbehalten.

Übersicht

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Benutzerdefinierte Funktionen	nein	nein	ja	ja	ja	ja
Abmelden (Seite 1612)	ja	ja	ja	ja	ja	ja
AktiviereBild (Seite 1569)	ja	ja	ja	ja	ja	ja
AktiviereBildMitNummer (Seite 1570)	ja	ja	ja	ja	ja	ja
AktivierePLCCodeAnzeige (Seite 1672)	nein	nein	nein	nein	ja	ja
AktivierePutzbild (Seite 1717)	nein	nein	nein	nein	nein	nein
AktiviereVorherigesBild (Seite 1571)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
AktualisiereVariable (Seite 1572)	ja	ja	ja	ja	ja	ja
ÄndereKontrast (Seite 1651)	ja	ja	nein	nein	nein	nein
Anmelden (Seite 1613)	ja	ja	ja	ja	ja	ja
ArchiviereProtokolldatei (Seite 1603)	nein	nein	ja	ja	ja	ja
ArchiviereVariable (Seite 1605)	nein	nein	ja	ja	ja	ja
BearbeiteMeldung (Seite 1686)	ja	ja	ja	ja	ja	ja
BildobjektCursorAbwärts (Seite 1721)	ja	ja	ja	ja	ja	ja
BildobjektCursorAufwärts (Seite 1721)	ja	ja	ja	ja	ja	ja
BildobjektSeiteAbwärts (Seite 1722)	ja	ja	ja	ja	ja	ja
BildobjektSeiteAufwärts (Seite 1723)	ja	ja	ja	ja	ja	ja
BildobjektCursorLinks (Seite 1723)	ja	ja	ja	ja	ja	ja
BildobjektCursorRechts (Seite 1722)	ja	ja	ja	ja	ja	ja
Codieren (Seite 1572)	ja	ja	ja	ja	ja	ja
CodierenEx (Seite 1573)	ja	ja	ja	ja	ja	ja
Direkttaste (Seite 1574)	ja	ja	ja	ja	ja	ja
DirekttasteBildnummer (Seite 1577)	ja	ja	ja	ja	ja	ja
DruckeBild (Seite 1638)	nein	ja	ja	ja	ja	ja
DruckeProtokoll (Seite 1639)	nein	ja	ja	ja	ja	ja
ErfasseBenutzeraktion (Seite 1577)	nein	nein	ja	ja	ja	ja
ErhöheFokussiertenWert (Seite 1579)	ja	ja	ja	ja	ja	ja
ErhöheVariable (Seite 1618)	ja	ja	ja	ja	ja	ja
ExportiereDatensätze (Seite 1692)	ja	ja	ja	ja	ja	ja
ExportiereDatensätzeMitPrüfsumme (Seite 1694)	nein	nein	ja	ja	ja	ja
ExportiereImportiereBenutzerverwaltung (Seite 1613)	ja	ja	ja	ja	ja	ja
GeheZumAnfang (Seite 1725)	ja	ja	ja	ja	ja	ja
GeheZumEnde (Seite 1725)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
HardwareSicherEntfernen (Seite 1579)	nein	nein	ja	ja	ja	ja
HTMLBrowserAbbrechen (Seite 1655)	nein	nein	nein	ja	ja	ja
HTMLBrowserAbwärtsScrollen (Seite 1656)	nein	nein	nein	ja	ja	ja
HTMLBrowserAktualisieren (Seite 1656)	nein	nein	nein	ja	ja	ja
HTMLBrowserAufwärtsScrollen (Seite 1657)	nein	nein	nein	ja	ja	ja
HTMLBrowserHeranzoomen (Seite 1657)	nein	nein	nein	ja	ja	ja
HTMLBrowserHerauszoomen (Seite 1658)	nein	nein	nein	ja	ja	ja
HTMLBrowserNachLinksScrollen (Seite 1658)	nein	nein	nein	ja	ja	ja
HTMLBrowserNachRechtsScrollen (Seite 1658)	nein	nein	nein	ja	ja	ja
HTMLBrowserSeiteNachUnten (Seite 1659)	nein	nein	nein	ja	ja	ja
HTMLBrowserSeiteNachOben (Seite 1659)	nein	nein	nein	ja	ja	ja
HTMLBrowserStartseite (Seite 1660)	nein	nein	nein	ja	ja	ja
HTMLBrowserVorwärts (Seite 1660)	nein	nein	nein	ja	ja	ja
HTMLBrowserZurück (Seite 1661)	nein	nein	nein	ja	ja	ja
ImportiereDatensätze (Seite 1697)	ja	ja	ja	ja	ja	ja
ImportiereDatensätzeMitPrüfsumme (Seite 1698)	nein	nein	ja	ja	ja	ja
InvertiereBit (Seite 1624)	ja	ja	ja	ja	ja	ja
InvertiereBitInVariable (Seite 1625)	ja	ja	ja	ja	ja	ja
InvertiereLineareSkalierung (Seite 1619)	ja	ja	ja	ja	ja	ja
KalibriereTouchscreen (Seite 1652)	ja	ja	ja [1)]	ja	ja	ja
KopiereArchiv (Seite 1606)	nein	nein	ja	ja	ja	ja
KurvenanzeigeBlättereVorwärts (Seite 1661)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
KurvenanzeigeBlättereZurück (Seite 1662)	ja	ja	ja	ja	ja	ja
KurvenanzeigeErweitern (Seite 1662)	ja	ja	ja	ja	ja	ja
KurvenanzeigeKomprimieren (Seite 1662)	ja	ja	ja	ja	ja	ja
KurvenanzeigeLinealLinks (Seite 1663)	ja	ja	ja	ja	ja	ja
KurvenanzeigeLinealRechts (Seite 1664)	ja	ja	ja	ja	ja	ja
KurvenanzeigeSetzeLinealmodus (Seite 1664)	ja	ja	ja	ja	ja	ja
KurvenanzeigeStartenStoppen (Seite 1665)	ja	ja	ja	ja	ja	ja
KurvenanzeigeZurückZumAnfang (Seite 1665)	ja	ja	ja	ja	ja	ja
LadeDatensatz (Seite 1700)	ja	ja	ja	ja	ja	ja
LeseBenutzername (Seite 1615)	ja	ja	ja	ja	ja	ja
LeseDatensatzAusSteuerung (Seite 1701)	ja	ja	ja	ja	ja	ja
LeseDatensatzname (Seite 1702)	ja	ja	ja	ja	ja	ja
LeseDatensatzvariablenAusSteuerung (Seite 1704)	ja	ja	ja	ja	ja	ja
LeseGruppennummer (Seite 1615)	ja	ja	ja	ja	ja	ja
LeseHelligkeit (Seite 1655)	nein	nein	nein	ja	ja	ja
LeseKennwort (Seite 1616)	ja	ja	ja	nein [2)]	nein [2)]	nein [2)]
LöscheArchiv (Seite 1607)	nein	nein	ja	ja	ja	ja
LöscheDatensatz (Seite 1705)	ja	ja	ja	ja	ja	ja
LöscheDatensatzSpeicher (Seite 1706)	ja	ja	ja	ja	ja	ja
LöscheMeldepuffer (Seite 1686)	ja	ja	ja	ja	ja	ja
LöscheMeldepufferProTool (Seite 1687)	ja	ja	ja	ja	ja	ja
MeldeanzeigeAktualisieren (Seite 1688)	ja	ja	ja	ja	ja	ja
MeldeanzeigeLoopInAlarm (Seite 1688)	ja	ja	ja	ja	ja	ja
MeldeanzeigeQuittiereMeldung (Seite 1689)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
MeldeanzeigeZeigeHilfetext (Seite 1690)	ja	ja	ja	ja	ja	ja
ÖffneAlleArchive (Seite 1608)	nein	nein	ja	ja	ja	ja
ÖffneDateiBrowser (Seite 1718)	nein	nein	nein	ja	ja	ja
ÖffneBildschirmtastatur (Seite 1720)	nein	nein	ja	ja	ja	ja
ÖffneEingabeaufforderung (Seite 1581)	nein	nein	ja	ja	ja	ja
ÖffneDialogSystemsteuerung (Seite 1580)	nein	nein	nein	ja	ja	ja
ÖffneInternetExplorer (Seite 1581)	nein	ja	ja	ja	ja	ja
ÖffneTaskManager (Seite 1719)	nein	nein	ja	ja	ja	ja
ÖffneSystemsteuerung (Seite 1582)	nein	nein	ja [1)]	nein	ja/nein [3)]	ja/nein [3)]
QuittiereMeldung (Seite 1690)	ja	ja	ja	ja	ja	ja
PDFAbwärtsScrollen (Seite 1666)	nein	nein	nein	ja	ja	ja
PDFAufwärtsScrollen (Seite 1666)	nein	nein	nein	ja	ja	ja
PDFAnpassenAufBreite (Seite 1667)	nein	nein	nein	ja	ja	ja
PDFAnpassenAufHöhe (Seite 1667)	nein	nein	nein	ja	ja	ja
PDFGeheZurErstenSeite (Seite 1668)	nein	nein	nein	ja	ja	ja
PDFGeheZurLetztenSeite (Seite 1668)	nein	nein	nein	ja	ja	ja
PDFGeheZurNächstenSeite (Seite 1669)	nein	nein	nein	ja	ja	ja
PDFGeheZuSeite (Seite 1669)	nein	nein	nein	ja	ja	ja
PDFGeheZurVorherigenSeite (Seite 1670)	nein	nein	nein	ja	ja	ja
PDFHeranzoomen (Seite 1670)	nein	nein	nein	ja	ja	ja
PDFHerauszoomen (Seite 1670)	nein	nein	nein	ja	ja	ja
PDFLinksScrollen (Seite 1671)	nein	nein	nein	ja	ja	ja
PDFRechtsScrollen (Seite 1671)	nein	nein	nein	ja	ja	ja
PDFZoomOriginal (Seite 1672)	nein	nein	nein	ja	ja	ja
PLCCodeAnzeigeDetails (Seite 1675)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeNächstesNetzwerk (Seite 1675)	ja	ja	ja	nein	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
PLCCodeAnzeigeZeigeSchritt-Modus (Seite 1676)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeSymbole (Seite 1676)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeTransitionInterlock (Seite 1677)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeHeranzoomen (Seite 1678)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeHerauszoomen (Seite 1678)	ja	ja	ja	nein	ja	ja
PLCCodeAnzeigeVorheriges-Netzwerk (Seite 1679)	ja	ja	ja	nein	ja	ja
RezepturanzeigeDatensatzNeu (Seite 1707)	ja	ja	ja	ja	ja	ja
RezepturanzeigeLeseDatensatzAusSteuerung (Seite 1707)	ja	ja	ja	ja	ja	ja
RezepturanzeigeLöscheDatensatz (Seite 1708)	ja	ja	ja	ja	ja	ja
RezepturanzeigeMenü (Seite 1708)	ja	ja	ja	ja	ja	ja
RezepturanzeigeÖffnen (Seite 1709)	ja	ja	ja	ja	ja	ja
RezepturanzeigeSchreibeDatensatzInSteuerung (Seite 1710)	ja	ja	ja	ja	ja	ja
RezepturanzeigeSpeichereDatensatz (Seite 1710)	ja	ja	ja	ja	ja	ja
RezepturanzeigeSpeichernUnterDatensatz (Seite 1711)	ja	ja	ja	ja	ja	ja
RezepturanzeigeSynchronisiereDatensatzMitVariablen (Seite 1711)	ja	ja	ja	ja	ja	ja
RezepturanzeigeUmbenennen-Datensatz (Seite 1712)	ja	ja	ja	ja	ja	ja
RezepturanzeigeZeigeHilfetext (Seite 1712)	ja	ja	ja	ja	ja	ja
RezepturanzeigeZurück (Seite 1712)	ja	ja	ja	ja	ja	ja
RücksetzeBit (Seite 1627)	ja	ja	ja	ja	ja	ja
RücksetzeBitInVariable (Seite 1629)	ja	ja	ja	ja	ja	ja
SchaltflächeDrücken (Seite 1585)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
SchaltflächeLoslassen (Seite 1586)	ja	ja	ja	ja	ja	ja
SchiebenUndMaskieren (Seite 1631)	ja	ja	ja	ja	ja	ja
SchließeAlleArchive (Seite 1609)	nein	nein	ja	ja	ja	ja
SchreibeDatensatzInSteuerung (Seite 1713)	ja	ja	ja	ja	ja	ja
SchreibeDatensatzvariablenIn-Steuerung (Seite 1714)	ja	ja	ja	ja	ja	ja
SeiteAbwärts (Seite 1724)	ja	ja	ja	ja	ja	ja
SeiteAufwärts (Seite 1724)	ja	ja	ja	ja	ja	ja
SendeEMail (Seite 1584)	nein	ja	ja	ja	ja	ja
SetzeAkustischesSignal (Seite 1653)	nein	nein	nein	ja	ja	ja
SetzeAnzeigemodus (Seite 1641)	nein	nein	nein	nein	nein	nein
SetzeBetriebsartGerät (Seite 1642)	ja	ja	ja	ja	ja	ja
SetzeBit (Seite 1633)	ja	ja	ja	ja	ja	ja
SetzeBitInVariable (Seite 1634)	ja	ja	ja	ja	ja	ja
SetzeBitWährendTasteGedrückt (Seite 1637)	ja	ja	ja	ja	ja	ja
SetzeFarbeHintergrundbeleuchtung (Seite 1654)	nein	nein	nein	nein	nein	nein
SetzeHelligkeit (Seite 1653)	nein	nein	nein	ja	ja	ja
SetzeModusBildschirmtastatur (Seite 1642)	nein	nein	nein	ja	ja	ja
SetzeProtokolliermodusMeldung (Seite 1639)	nein	nein	ja	ja	ja	ja
SetzeRezepturvariablen (Seite 1715)	ja	ja	ja	ja	ja	ja
SetzeSommerzeit (Seite 1584)	nein	nein	ja	ja	ja	ja
SetzeSprache (Seite 1645)	ja	ja	ja	ja	ja	ja
SetzeVariable (Seite 1621)	ja	ja	ja	ja	ja	ja
SetzeVerbindungsmodus (Seite 1646)	ja	ja	ja	ja	ja	ja
SetzeWebZugriff (Seite 1647)	nein	ja	ja	ja	ja	ja
SichereRAMDateisystem (Seite 1587)	nein	nein	ja	ja	ja	ja
SimuliereSystemtaste (Seite 1587)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
SimuliereVariable (Seite 1588)	ja	ja	ja	ja	ja	ja
SmartClientAnzeigeAktualisieren (Seite 1679)	nein	ja	ja	ja	ja	ja
SmartClientAnzeigeSchreibschutzAus (Seite 1680)	nein	ja	ja	ja	ja	ja
SmartClientAnzeigeSchreibschutzEin (Seite 1680)	nein	ja	ja	ja	ja	ja
SmartClientAnzeigeTrennen (Seite 1681)	nein	ja	ja	ja	ja	ja
SmartClientAnzeigeVerbinden (Seite 1681)	nein	ja	ja	ja	ja	ja
SmartClientAnzeigeVerlassen (Seite 1682)	nein	ja	ja	ja	ja	ja
SpeichereDatensatz (Seite 1716)	ja	ja	ja	ja	ja	ja
StarteArchivierung (Seite 1609)	nein	nein	ja	ja	ja	ja
StarteNächstesArchiv (Seite 1610)	nein	nein	ja	ja	ja	ja
StarteProgramm (Seite 1589)	nein	nein	ja	ja	ja	ja
StatusSteuernLeseWerte (Seite 1591)	ja	ja	ja	ja	ja	ja
StatusSteuernSchreibeWerte (Seite 1592)	ja	ja	ja	ja	ja	ja
SteuereSmartServer (Seite 1592)	nein	ja	ja	ja	ja	ja
SteuereWebServer (Seite 1593)	nein	ja	ja	ja	ja	ja
StoppeArchivierung (Seite 1611)	nein	nein	ja	ja	ja	ja
StoppeRuntime (Seite 1593)	ja	ja	ja	ja	ja	ja
TextAbrufen (Seite 1594)	ja	ja	ja	ja	ja	ja
SystemdiagnoseAnzeigeAktivieren (Seite 1682)	nein	nein	nein	ja	ja	ja
TrenneVariableVonHandrad (Seite 1595)	ja	ja	ja	nein	nein	nein
VerbindeVariableMitHandrad (Seite 1596)	ja	ja	ja	nein	nein	nein
VerfolgeBenutzeränderung (Seite 1617)	ja	ja	ja	nein	nein	nein
VerringereFokussiertenWert (Seite 1596)	ja	ja	ja	ja	ja	ja
VerringereVariable (Seite 1623)	ja	ja	ja	ja	ja	ja
WechseleVerbindung (Seite 1648)	ja	ja	ja	ja	ja	ja

	Mobile Panel 177 DP	Mobile Panel 177 PN	Mobile Panel 277 Mobile Panel 277 IWLAN Mobile Panel 277 F IWLAN	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
WechseleVerbindungEIP (Seite 1649)	ja	ja	ja	ja	ja	ja
ZeigeAnmeldedialog (Seite 1617)	ja	ja	ja	ja	ja	ja
ZeigeHilfetext (Seite 1597)	ja	ja	ja	ja	ja	ja
ZeigeMeldefenster (Seite 1691)	ja	ja	ja	ja	ja	ja
ZeigePopupBild (Seite 1598)	nein	nein	nein	ja	ja	ja
ZeigePopupBildGrößeVeränderbar	nein	nein	nein	ja	ja	ja
ZeigeSlideInBild (Seite 1602)	nein	nein	nein	ja	ja	ja
ZeigeSoftwareVersion (Seite 1598)	nein	nein	ja	ja	ja	ja
ZeigeSystemDiagnosefenster (Seite 1685)	nein	nein	nein	ja	ja	ja
ZeigeSystemmeldung (Seite 1691)	ja	ja	ja	ja	ja	ja

1) Nur für Mobile Panel 277, Mobile Panel 277 IWLAN

2) Für Geräte ab der Geräteversion V14.0 nicht verfügbar

3) Nicht für F-Panels

Siehe auch

LineareSkalierung (Seite 1620)

PROFIsafeTrennen (Seite 1583)

Systemfunktionen für Runtime Advanced

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen bei WinCC RT Advanced.

Technische Änderungen vorbehalten.

Übersicht

	WinCC RT Advanced
Benutzerdefinierte Funktionen	ja
Abmelden (Seite 1612)	ja
AktiviereBild (Seite 1569)	ja
AktiviereBildMitNummer (Seite 1570)	ja
AktivierePLCCodeAnzeige (Seite 1672)	ja
AktivierePutzbild (Seite 1717)	ja
AktiviereVorherigesBild (Seite 1571)	ja
AktualisiereVariable (Seite 1572)	ja
ÄndereKontrast (Seite 1651)	nein
Anmelden (Seite 1613)	ja
ArchiviereProtokolldatei (Seite 1603)	ja
ArchiviereVariable (Seite 1605)	ja
BearbeiteMeldung (Seite 1686)	ja
BildobjektCursorAbwärts (Seite 1721)	ja
BildobjektCursorAufwärts (Seite 1721)	ja
BildobjektCursorLinks (Seite 1723)	ja
BildobjektCursorRechts (Seite 1722)	ja
BildobjektSeiteAbwärts (Seite 1722)	ja
BildobjektSeiteAufwärts (Seite 1723)	ja
Codieren (Seite 1572)	ja
CodierenEx (Seite 1573)	ja
Direkttaste (Seite 1574)	nein
DirekttasteBildnummer (Seite 1577)	nein
DruckeBild (Seite 1638)	ja
DruckeProtokoll (Seite 1639)	ja
ErfasseBenutzeraktion (Seite 1577)	ja
ErhöheFokussiertenWert (Seite 1579)	ja
ErhöheVariable (Seite 1618)	ja
ExportiereDatensätze (Seite 1692)	ja
ExportiereDatensätzeMitPrüfsumme (Seite 1694)	nein
ExportiereImportiereBenutzerverwaltung (Seite 1613)	ja
GeheZumAnfang (Seite 1725)	ja
GeheZumEnde (Seite 1725)	ja
HardwareSicherEntfernen (Seite 1579)	nein
HTMLBrowserAbbrechen (Seite 1655)	ja
HTMLBrowserAbwärtsScrollen (Seite 1656)	ja
HTMLBrowserAktualisieren (Seite 1656)	ja
HTMLBrowserAufwärtsScrollen (Seite 1657)	ja
HTMLBrowserHeranzoomen (Seite 1657)	ja
HTMLBrowserHerauszoomen (Seite 1658)	ja

	WinCC RT Advanced
HTMLBrowserNachLinksScrollen (Seite 1658)	ja
HTMLBrowserNachRechtsScrollen (Seite 1658)	ja
HTMLBrowserSeiteNachUnten (Seite 1659)	ja
HTMLBrowserSeiteNachOben (Seite 1659)	ja
HTMLBrowserStartseite (Seite 1660)	ja
HTMLBrowserVorwärts (Seite 1660)	ja
HTMLBrowserZurück (Seite 1661)	ja
ImportiereDatensätze (Seite 1697)	ja
ImportiereDatensätzeMitPrüfsumme (Seite 1698)	ja
InvertiereBit (Seite 1624)	ja
InvertiereBitInVariable (Seite 1625)	ja
InvertiereLineareSkalierung (Seite 1619)	ja
KalibriereTouchscreen (Seite 1652)	nein
KopiereArchiv (Seite 1606)	ja
KurvenanzeigeBlättereVorwärts (Seite 1661)	ja
KurvenanzeigeBlättereZurück (Seite 1662)	ja
KurvenanzeigeErweitern (Seite 1662)	ja
KurvenanzeigeKomprimieren (Seite 1662)	ja
KurvenanzeigeLinealLinks (Seite 1663)	ja
KurvenanzeigeLinealRechts (Seite 1664)	ja
KurvenanzeigeSetzeLinealmodus (Seite 1664)	ja
KurvenanzeigeStartenStoppen (Seite 1665)	ja
KurvenanzeigeZurückZumAnfang (Seite 1665)	ja
LadeDatensatz (Seite 1700)	ja
LeseBenutzername (Seite 1615)	ja
LeseDatensatzAusSteuerung (Seite 1701)	ja
LeseDatensatzname (Seite 1702)	ja
LeseDatensatzvariablenAusSteuerung (Seite 1704)	ja
LeseGruppennummer (Seite 1615)	ja
LeseHelligkeit (Seite 1655)	nein
LeseKennwort (Seite 1616)	nein [1)]
LineareSkalierung (Seite 1620)	ja
LöscheArchiv (Seite 1607)	ja
LöscheDatensatz (Seite 1705)	ja
LöscheDatensatzSpeicher (Seite 1706)	nein
LöscheMeldepuffer (Seite 1686)	ja
LöscheMeldepufferProTool (Seite 1687)	ja
MeldeanzeigeAktualisieren (Seite 1688)	ja
MeldeanzeigeLoopInAlarm (Seite 1688)	ja
MeldeanzeigeQuittiereMeldung (Seite 1689)	ja
MeldeanzeigeZeigeHilfetext (Seite 1690)	ja

	WinCC RT Advanced
ÖffneAlleArchive (Seite 1608)	ja
ÖffneBildschirmtastatur (Seite 1720)	ja
ÖffneDialogSystemsteuerung (Seite 1580)	nein
ÖffneEingabeaufforderung (Seite 1581)	nein
ÖffneInternetExplorer (Seite 1581)	nein
ÖffneSystemsteuerung (Seite 1582)	nein
ÖffneTaskManager (Seite 1719)	ja
QuittiereMeldung (Seite 1690)	ja
PDFAbwärtsScrollen (Seite 1666)	ja
PDFAufwärtsScrollen (Seite 1666)	ja
PDFAnpassenAufBreite (Seite 1667)	ja
PDFAnpassenAufHöhe (Seite 1667)	ja
PDFGeheZurErstenSeite (Seite 1668)	ja
PDFGeheZurLetztenSeite (Seite 1668)	ja
PDFGeheZurNächstenSeite (Seite 1669)	ja
PDFGeheZuSeite (Seite 1669)	ja
PDFGeheZurVorherigenSeite (Seite 1670)	ja
PDFHeranzoomen (Seite 1670)	ja
PDFHerauszoomen (Seite 1670)	ja
PDFLinksScrollen (Seite 1671)	ja
PDFRechtsScrollen (Seite 1671)	ja
PDFZoomOriginal (Seite 1672)	ja
PLCCodeAnzeigeDetails (Seite 1675)	ja
PLCCodeAnzeigeNächstesNetzwerk (Seite 1675)	ja
PLCCodeAnzeigeZeigeSchrittModus (Seite 1676)	ja
PLCCodeAnzeigeSymbole (Seite 1676)	ja
PLCCodeAnzeigeTransitionInterlock (Seite 1677)	ja
PLCCodeAnzeigeHeranzoomen (Seite 1678)	ja
PLCCodeAnzeigeHerauszoomen (Seite 1678)	ja
PLCCodeAnzeigeVorherigesNetzwerk (Seite 1679)	ja
RezepturanzeigeDatensatzNeu (Seite 1707)	ja
RezepturanzeigeLeseDatensatzAusSteuerung (Seite 1707)	ja
RezepturanzeigeLöscheDatensatz (Seite 1708)	ja
AUTOHOTSPOT	ja
RezepturanzeigeMenü (Seite 1708)	ja
RezepturanzeigeÖffnen (Seite 1709)	ja
RezepturanzeigeSchreibeDatensatzInSteuerung (Seite 1710)	ja
RezepturanzeigeSpeichereDatensatz (Seite 1710)	ja
RezepturanzeigeSpeichernUnterDatensatz (Seite 1711)	ja
RezepturanzeigeSynchronisiereDatensatzMitVariablen (Seite 1711)	ja
RezepturanzeigeUmbenennenDatensatz (Seite 1712)	ja

	WinCC RT Advanced
RezepturanzeigeZeigeHilfetext (Seite 1712)	ja
RezepturanzeigeZurück (Seite 1712)	ja
RücksetzeBit (Seite 1627)	ja
RücksetzeBitInVariable (Seite 1629)	ja
SchaltflächeDrücken (Seite 1585)	ja
SchaltflächeLoslassen (Seite 1586)	ja
SchiebenUndMaskieren (Seite 1631)	ja
SchließeAlleArchive (Seite 1609)	ja
SchreibeDatensatzInSteuerung (Seite 1713)	ja
SchreibeDatensatzvariablenInSteuerung (Seite 1714)	ja
SeiteAbwärts (Seite 1724)	ja
SeiteAufwärts (Seite 1724)	ja
SendeEMail (Seite 1584)	ja
SetzeAkustischesSignal (Seite 1653)	nein
SetzeAnzeigemodus (Seite 1641)	ja
SetzeBetriebsartGerät (Seite 1642)	ja
SetzeBit (Seite 1633)	ja
SetzeBitInVariable (Seite 1634)	ja
SetzeBitWährendTasteGedrückt (Seite 1637)	ja
SetzeFarbeHintergrundbeleuchtung (Seite 1654)	nein
SetzeHelligkeit (Seite 1653)	nein
SetzeModusBildschirmtastatur (Seite 1642)	ja
SetzeProtokolliermodusMeldung (Seite 1639)	ja
SetzeRezepturvariablen (Seite 1715)	ja
SetzeSommerzeit (Seite 1584)	ja
SetzeSprache (Seite 1645)	ja
SetzeVariable (Seite 1621)	ja
SetzeVerbindungsmodus (Seite 1646)	ja
SetzeWebZugriff (Seite 1647)	ja
SichereRAMDateisystem (Seite 1587)	nein
SimuliereSystemtaste (Seite 1587)	ja
SimuliereVariable (Seite 1588)	ja
SmartClientAnzeigeAktualisieren (Seite 1679)	ja
SmartClientAnzeigeSchreibschutzAus (Seite 1680)	ja
SmartClientAnzeigeSchreibschutzEin (Seite 1680)	ja
SmartClientAnzeigeTrennen (Seite 1681)	ja
SmartClientAnzeigeVerbinden (Seite 1681)	ja
SmartClientAnzeigeVerlassen (Seite 1682)	ja
SpeichereDatensatz (Seite 1716)	ja
StarteArchivierung (Seite 1609)	ja
StarteNächstesArchiv (Seite 1610)	ja

	WinCC RT Advanced
StarteProgramm (Seite 1589)	ja
StatusSteuernLeseWerte (Seite 1591)	ja
StatusSteuernSchreibeWerte (Seite 1592)	ja
SteuereSmartServer (Seite 1592)	ja
SteuereWebServer (Seite 1593)	ja
StoppeArchivierung (Seite 1611)	ja
StoppeRuntime (Seite 1593)	ja
SystemdiagnoseAnzeigeAktivieren (Seite 1682)	ja
TextAbrufen (Seite 1594)	ja
TrenneVariableVonHandrad (Seite 1595)	nein
VerbindeVariableMitHandrad (Seite 1596)	nein
VerfolgeBenutzeränderung (Seite 1617)	ja
VerringereFokussiertenWert (Seite 1596)	ja
VerringereVariable (Seite 1623)	ja
WechseleVerbindung (Seite 1648)	ja
WechseleVerbindungEIP (Seite 1649)	ja
AUTOHOTSPOT	ja
ZeigeAnmeldedialog (Seite 1617)	ja
ZeigeHilfetext (Seite 1597)	ja
ZeigeMeldefenster (Seite 1691)	ja
ZeigePopupBild (Seite 1598)	ja
ZeigePopupBildGrößeVeränderbar (Seite 1600)	ja
ZeigeSlideInBild (Seite 1602)	ja
ZeigeSoftwareVersion (Seite 1598)	ja
ZeigeSystemDiagnosefenster (Seite 1685)	ja
ZeigeSystemmeldung (Seite 1691)	ja

1) Für Geräte ab der Geräteversion V14.0 nicht verfügbar

Siehe auch

PROFIsafeTrennen (Seite 1583)

Systemfunktionen für Runtime Professional

Verfügbarkeit der Systemfunktionen

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Systemfunktionen bei WinCC Runtime Professional.

Technische Änderungen vorbehalten.

Übersicht

	WinCC RT Professional
Benutzerdefinierte Funktionen	ja
AktiviereBild (Seite 1744)	ja
AktiviereBildInAktuellemBildfenster (Seite 1745)	ja
AktiviereBildInBildfenster (Seite 1746)	ja
ErhöheVariable (Seite 1730)	ja
ExportiereImportiereBenutzerverwaltung (Seite 1728)	ja
InvertiereBit (Seite 1759)	ja
InvertiereBitInVariable (Seite 1761)	ja
InvertiereLineareSkalierung (Seite 1731)	ja
LineareSkalierung (Seite 1733)	ja
RücksetzeBit (Seite 1763)	ja
RücksetzeBitInVariable (Seite 1764)	ja
SetzeBit (Seite 1766)	ja
SetzeBitInVariable (Seite 1767)	ja
SetzeEigenschaftAktuellesFenster (Seite 1757)	ja
SetzeEigenschaftAktuellesFensterDurchEigenschaft (Seite 1757)	ja
SetzeEigenschaftAktuellesFensterDurchVariableIndirekt (Seite 1758)	ja
SetzeEigenschaftDurchEigenschaft (Seite 1749)	ja
SetzeEigenschaftDurchKonstante (Seite 1751)	ja
SetzeEigenschaftDurchVariable (Seite 1753)	ja
SetzeEigenschaftDurchVariableIndirekt (Seite 1755)	ja
SetzeVariable (Seite 1734)	ja
SetzeVariableDurchEigenschaft (Seite 1736)	ja
SetzeVariableDurchVariableIndirekt (Seite 1740)	ja
SetzeVariableMitBedienmeldung (Seite 1741)	ja
SetzeVariableIndirekt (Seite 1737)	ja
SetzeVariableIndirektDurchEigenschaft (Seite 1738)	ja
SetzeVariableIndirektDurchVariableIndirekt (Seite 1739)	ja
StoppeRuntime (Seite 1726)	ja
VerringereVariable (Seite 1742)	ja
ZeigeAnmeldedialog (Seite 1729)	ja
ZeigePLCCodeAnzeigeVonMeldung (Seite 1773)	ja
ZeigeBlockInTiaPortalVonMeldung (Seite 1772)	ja

9.11.1.2 Systemfunktionen

Andere Funktionen

AktiviereBild

Beschreibung

Führt einen Bildwechsel zu dem angegebenen Bild aus.

Um vom Grundbild in den Permanentbereich oder umgekehrt zu wechseln, verwenden Sie die Systemfunktion "AktiviereBildMitNummer".

Verwendung in der Funktionsliste

AktiviereBild (Bildname, Objektnummer)

Verwendung in benutzerdefinierten Funktionen

ActivateScreen Screen_name, Object_number

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit ".

Parameter

Bildname

Name des Bildes, zu dem gewechselt wird.

Objektnummer

Das Bedienelement, das nach dem Bildwechsel im angegebenen Bild den Fokus erhält. Die Nummer des Bedienelements legen Sie während der Projektierung durch die Tab-Reihenfolge fest.

Wenn Sie "0" angeben:

- Ist der Fokus beim Aufruf der Systemfunktion im Permanentbereich, behält der Permanentbereich den Fokus.
- Ist der Fokus beim Aufruf der Systemfunktion im Grundbild, erhält das erste Bedienelement im angegebenen Bild den Fokus.

Hinweis

Wenn die Systemfunktion "AktiviereBild" an das Ereignis "Rand erreicht" projektiert wird, so ist für den Parameter "Objektnummer" nur der Wert 0 zulässig. Das aktive Objekt wird nicht anhand der Objektnummer festgelegt, sondern anhand der X-Position des aktiven Objekts vor dem Bildwechsel.

Beispiel

Der folgende Programmcode aktiviert mit der Funktion ActivateScreen das Bild "Screen_2", z. B. beim Klicken auf eine Schaltfläche.

```
Sub ActivateScreen_2()

'Userdefined code
'' i. e. when pressing a button

ActivateScreen "Screen_2",0
```

Siehe auch

AktiviereBildMitNummer (Seite 1570)

AktiviereBildMitNummer

Beschreibung

Führt abhängig von einem Variablenwert einen Bildwechsel zu einem Bild aus.

Das Bild wird über seine Bildnummer identifiziert.

Verwendung in der Funktionsliste

AktiviereBildMitNummer (Bildnummer, Objektnummer)

Verwendung in benutzerdefinierten Funktionen

ActivateScreenByNumber Screen_number, Object_number

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Bildnummer

Variable, welche die Bildnummer enthält, zu dem gewechselt wird.

Wenn Sie vom Grundbild in den Permanentbereich oder umgekehrt wechseln wollen, geben Sie "0" oder "-1" an:

0 = Wechselt vom Grundbild in den Permanentbereich.

-1 = Wechselt vom Permanentbereich ins Grundbild

Objektnummer

Nummer des Bildobjekts, das nach dem Bildwechsel im angegebenen Bild den Fokus erhält. Die Nummer des Bedienelements legen Sie während der Projektierung durch die Tab-Reihenfolge fest.

Wenn Sie "0" angeben:

- Ist der Fokus beim Aufruf der Systemfunktion im Permanentbereich, behält der Permanentbereich den Fokus.
- Ist der Fokus beim Aufruf der Systemfunktion im Grundbild, erhält das erste Bedienelement im angegebenen Bild den Fokus.

Siehe auch

AktiviereBild (Seite 1569)

AktiviereVorherigesBild

Beschreibung

Führt einen Bildwechsel zu dem Bild aus, das vor dem aktuellen Bild aktiviert war. Der Bildwechsel wird nicht ausgeführt, wenn zuvor kein Bild aktiviert war.

Die 10 zuletzt aufgerufenen Bilder werden gespeichert. Wenn Sie zu einem Bild wechseln, das nicht mehr gespeichert ist, wird eine Systemmeldung ausgegeben.

Hinweis

Wenn Sie die Systemfunktion verwenden wollen, muss das Bild zu dem sie wechseln in der Navigationsstruktur verwendet werden.

Verwendung in der Funktionsliste

AktiviereVorherigesBild

Verwendung in benutzerdefinierten Funktionen

ActivatePreviousScreen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit ".

Parameter

--

AktualisiereVariable

Beschreibung

Liest für die Variable mit der angegebenen Aktualisierungskennung den aktuellen Wert aus der Steuerung.

Verwendung in der Funktionsliste

AktualisiereVariable (Aktualisierungskennung)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Aktualisierungskennung

Aktualisierungskennung, die der zu aktualisierenden Variablen zugeordnet ist.

Siehe auch

Abmelden (Seite 1612)

Codieren

Beschreibung

Passt den Datentyp "String" einer Variablen für Übergabe an das Automatisierungssystem (AS) an. Die Variable vom WinCC-Datentyp "String" wird in den AS-Datentyp "Array of Byte" umgewandelt. Das Ergebnis wird in eine Variable geschrieben.

Verwendung in der Funktionsliste

Codieren (Byte-Array, Zeichenfolge, Codieren)

Verwendung in benutzerdefinierten Funktionen

Encode Byte_array, String, Encoding

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Byte-Array

Die Variable, die den umgewandelten Wert enthält.

Hinweis

Das Byte-Array muss die doppelte Größe der Stringlänge + 2 haben. Wenn der String eine Länge von 120 Zeichen hat, muss das Byte-Array 242 Arrayelemente enthalten.

Wenn die Größe nicht ausreichend ist, werden Zeichen abgeschnitten bzw. nicht konvertiert.

Zeichenfolge

Die Variable vom Datentyp "String", die umgewandelt wird.

Codieren

0 (hmiEncodeUTF16LE) - String wird in UTF16LE (Unicode 16 Little Endian) codiert.

CodierenEx

Beschreibung

Passt den Datentyp "String" einer Variablen für Übergabe an das Automatisierungssystem (AS) an. Die Variable vom WinCC-Datentyp "String" wird in den AS-Datentyp "Array of Byte" umgewandelt. Das Ergebnis wird in eine Variable geschrieben.

Im Gegensatz zur Systemfunktion Codieren ist der Parameter Zeilenumbruch definierbar. Mit dem Parameter Zeilenumbruch löschen Sie die Zeilenumbrüche oder ersetzen die Zeilenumbrüche durch vordefinierte Zeichen.

Verwendung in der Funktionsliste

CodierenEx (Byte-Array, Zeichenkette, Kodierung, Zeilenumbruch)

Verwendung in benutzerdefinierten Funktionen

EncodeEx Byte_array, String, Encoding, Line_break

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Byte-Array

Die Variable, die den umgewandelten Wert enthält.

Hinweis

Das Byte-Array muss die doppelte Größe der Stringlänge +2 haben.Wenn der String eine Länge von 120 Zeichen besitzt, muss das Byte-Array 242 Arrayelemente enthalten.

Wenn die Größe nicht ausreichend ist, werden Zeichen abgeschnitten bzw. nicht konvertiert.

Zeichenkette

Die Variable vom Datentyp "String", die umgewandelt wird.

Kodierung

0 (hmiEncodeUTF16LE) - String wird in UTF16LE (Unicode 16 Little Endian) codiert.

Zeilenumbruch

Alle Zeilenumbrüche werden durch vordefinierte Zeichen ersetzt oder gelöscht. Zeilenumbrüche nicht ersetzen ist als Standardwert gesetzt.

0 (Ersetzen mit "\r\n' (0x000D, 0x000A)) - Zeilenumbruch wird durch "\r\n" ersetzt.

1 (Ersetzen mit "\n' (0x000A)) - Zeilenumbruch wird durch "\n" ersetzt.

2 (Nicht ersetzen) - Zeilenumbruch wird nicht ersetzt.

3 (Entfernen der Zeilenumbrüche) - Zeilenumbrüche werden gelöscht.

Direkttaste

Beschreibung

Ermöglicht am Touch-Display eines Bediengeräts schnelle Tastenbedienungen, ohne dass kommunikationsbedingte Verzögerungen auftreten.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. direkt vom Bediengerät aus Bits im E/A-Bereich einer SIMATIC S7 setzen wollen.

Hinweis

Wenn Sie WinCC integriert in SIMATIC STEP 7 betreiben und Sie einen DP-Bereich in SIMATIC STEP 7 konfiguriert haben: Beim Betätigen der Schaltfläche, an die Sie diese Systemfunktion projektiert haben, wird im E/A-Bereich der CPU ein Bit gesetzt. Beim Loslassen der Schaltfläche wird das Bit zurückgesetzt.

Hinweise zur Projektierung der Systemfunktion

Wenn Sie eine Schaltfläche mit der Systemfunktion "Direkttaste" projektieren, wird ein fester Bereich des Touch-Displays für die Direkttasten-Bedienung reserviert.

Bildobjekte, die sich in Runtime in diesem Bereich über der Schaltfläche mit der Systemfunktion "Direkttaste" befinden, verdecken die Schaltfläche zwar optisch. Die Bildobjekte verhindern jedoch nicht die Auslösung der Systemfunktion "Direkttaste".

Hinweis

Wird eine externe Applikation, wie Pocket Internet Explorer oder Control Panel, gestartet, so wird diese im Vordergrund aktiv und legt die Runtime in den Hintergrund. Das Bit für die Funktion "DirekttasteBildnummer" ist nicht mehr gesetzt und die Tasten oder Schaltflächen mit der projektierten Funktion "Direkttaste" lösen das zugehörige Bit in der Steuerung nicht mehr aus.

⚠ WARNUNG

Das unbeabsichtigte Auslösen der Systemfunktion "Direkttaste" führt zur Gefährdung von Personen oder Sachschäden an der Maschine.

Um diese Gefahren zu vermeiden, müssen Sie Folgendes beachten:

- Bei der Projektierung des Prozessbildes darf kein Bildobjekt die Schaltfläche mit der Systemfunktion "Direkttaste" überdecken.
- Die Dynamisierung der Position oder Anzeige (Freigabe) eines Bildobjekts, abhängig von Prozesswerten, darf in Runtime nicht zu einer Überdeckung der Schaltfläche mit der Systemfunktion "Direkttaste" führen.

  Hinweis

  Beachten Sie diese Vorgabe während der Projektierung. Überprüfen Sie auch bestehende Projektierungen und passen Sie diese gegebenenfalls unverzüglich an.

In folgenden Fällen führt die Überdeckung einer Schaltfläche mit der Systemfunktion "Direkttaste" nicht zu oben beschriebenem Problem:

- Bedienen des Meldefensters
- Bedienen der Rezepturanzeige
- Aufheben des Bildschirmschoners
- Verwenden der Bildschirmtastatur
- Ausführen der Systemfunktion "AktivierePutzbild"

Verhalten von DP-Direkttasten im Offline-Betrieb

Wenn die Systemfunktion "WechseleVerbindung" mit dem Parameter "Offline" am Bediengerät ausgeführt wird, so wird die Verbindung zur angegebenen Steuerung getrennt.

Hinweis

Beachten Sie sowohl auf SIMATIC STEP 7-Seite als auch auf WinCC-Seite Folgendes:

Die DP-Direkttasten sind in diesem Fall weiterhin aktiv. Wenn Sie in der Betriebsart "Offline" eine Schaltfläche mit der Systemfunktion "Direkttaste" oder bei einem Tastengerät die Direkttaste betätigen, wird das zugehörige Bit in der Steuerung gesetzt.

Verwendung in der Funktionsliste

Direkttaste (Bit)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bit

Legt das Bit fest, das gesetzt wird. Je nach Bediengerät ist hier Bitnummer 0 bis 31 oder 0 bis 39 möglich.

Anwendungsbeispiel

Ziel

Sie wollen über eine Schaltfläche direkt vom Bediengerät aus Bits im E/A-Bereich einer SIMATIC S7 setzen.

Voraussetzung

WinCC integriert in SIMATIC STEP 7 ist installiert.

Das Bediengerät ist während des Betriebs an eine SIMATIC S7 über PROFIBUS-DP angekoppelt.

Sie haben zum Zeitpunkt des Generierens Ihres Projekts WinCC integriert installiert.

Der Bitbereich für Direkttasten ist in SIMATIC STEP 7 festgelegt. Zur Projektierung siehe "Benutzerhandbuch Kommunikation".

Hinweise zur Projektierung

Projektieren Sie die Schaltfläche, die Sie als Direkttaste verwenden wollen. Ordnen Sie der Schaltfläche die Systemfunktion "Direkttaste" zu. Geben Sie als Parameter die Bitnummer an, die beim Drücken der Taste gesetzt wird.

Ablauf am Bediengerät

Das Bit wird mit dem Berühren der Direkttaste gesetzt und beim Loslassen der Taste oder beim Verlassen des Bildes wieder zurückgesetzt.

DirekttasteBildnummer

Beschreibung

Setzt innerhalb des angegebenen Bitbereichs einer Direkttaste das Bit und überträgt es zur S7-Steuerung, die mit dem Bediengerät verbunden ist. Damit ist eine eindeutige Zuordnung zwischen Steuerbit und Bildnummer jederzeit gewährleistet.

Ohne die Verwendung der Systemfunktion muss die S7-Steuerung die jeweilige Funktionalität über die Bildnummer unterscheiden. Dies verzögert die Aktualisierung der Bildnummer nach einem Bildwechsel.

Hinweis

Wird eine externe Applikation, wie der Pocket Internet Explorer oder das Control Panel, gestartet, so wird diese im Vordergrund aktiv und legt die Runtime in den Hintergrund. Dennoch bleibt das Bit für die Funktion "DirekttasteBildnummer" gesetzt und die Tasten oder Schaltflächen mit der projektierten Funktion "Direkttaste" lösen weiterhin das zugehörige Bit in der Steuerung aus.

Verwendung in der Funktionsliste

DirekttasteBildnummer (Bit)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bit

Legt das Bit fest, das gesetzt wird. Je nach Bediengerät ist hier Bitnummer 0 bis 31 oder 0 bis 39 möglich.

ErfasseBenutzeraktion

Beschreibung

Mit dieser Systemfunktion erfassen Sie Benutzeraktionen im Audit Trail, die nicht automatisch im Audit Trail eingetragen werden. Außerdem können Sie mithilfe dieser Systemfunktion vom Benutzer eine Quittierung oder eine elektronische Unterschrift für die Bedienaktion sowie einen

Kommentar einfordern. Voraussetzung für die Verwendung der Systemfunktion ist, dass unter "Runtime-Einstellungen > GMP" die GMP-konforme Projektierung aktiviert ist.

Wenn Sie in einer Funktion die Systemfunktion "NotifyUserAction" verwenden und die Eingabe über die Schaltfläche "Abbrechen" beenden, kann der Debugger aufgerufen werden. Um dieses Verhalten abzufangen, können Sie in einer Funktion die Anweisung "On Error Resume Next" verwenden. Diese Anweisung bewirkt, dass nach einem Laufzeitfehler die Folgeanweisung ausgeführt wird. Wenn Sie die Anweisung "On Error Resume Next" verwenden, wird auch die Ausgabe von Systemmeldungen unterdrückt.

Verwendung in der Funktionsliste

ErfasseBenutzeraktion (Bestätigungstyp, Kommentar erforderlich, Kategorie, Objektname, Beschreibung)

Verwendung in benutzerdefinierten Funktionen

NotifyUserAction Confirmation_type, Mandatory_commenting, Category, Object_name, Description

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Bestätigungstyp

Legt fest, wie die Aktion bestätigt werden muss

0 = (None): Keine Bestätigung erforderlich, ein Eintrag im Audit Trail wird erzeugt

1 = (Acknowledgement): Quittierung, der Benutzer muss die Aktion quittieren; ein Eintrag im Audit Trail wird erzeugt

2 = (Digital Signature): elektronische Unterschrift, ein Dialogfenster wird eingeblendet, in dem der Benutzer seine elektronische Unterschrift eingeben muss; ein Eintrag im Audit Trail wird erzeugt

Kommentar erforderlich

Legt fest, ob der Benutzer einen Kommentar eingeben muss. Der Kommentar wird im Audit Trail archiviert.

0 = (True): Wahr; ein Dialogfenster wird eingeblendet, in dem der Benutzer einen Kommentar eingeben muss

1 = (False): Falsch; kein Kommentar erforderlich

Kategorie

Kategorie oder Klassenname des geänderten Objekts

Objektname

Name des geänderten Objekts

Beschreibung

Text, der die zu archivierende Benutzeraktion beschreibt.

ErhöheFokussiertenWert

Beschreibung

Addiert den angegebenen Wert zum Wert der Variablen, die mit dem Eingabefeld (symbolisches Auswahlfeld, grafisches Auswahlfeld, Schieber) verbunden ist, das aktuell den Fokus besitzt.

Diese Systemfunktion kann nur an Funktionstasten verwendet werden.

Verwendung in der Funktionsliste

ErhöheFokussiertenWert (Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Wert

Der Wert, der zum Variablenwert addiert wird.

HardwareSicherEntfernen

Beschreibung

Prüft, ob auf dem externen Speichermedium schreibende oder lesende Zugriffe ausgeführt werden. Wenn keine Zugriffe ausgeführt werden, lässt sich das externe Speichmedium ohne Datenverlust entfernen.

Verwendung in der Funktionsliste

HardwareSicherEntfernen(Pfad, Ergebnis)

Verwendung in benutzerdefinierten Funktionen

SafelyRemoveHardware Path, Result

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Ergebnis

Variable, in der das Ergebnis eingetragen wird.

TRUE : Speichermedium kann sicher entfernt werden. Eine entsprechende Systemmeldung wird ausgegeben.

FALSE: Speichermedium kann nicht entfernt werden. Eine entsprechende Systemmeldung wird ausgegeben.

Pfad

Pfad des Speichermediums , z. B. \Storage Card USB\

ÖffneDialogSystemsteuerung

Beschreibung

Öffnet einen Dialog, über den ausgewählte Einstellungen der Systemsteuerung verändert werden können.

Mit dieser Systemfunktion stellen Sie unter anderem am Bediengerät Folgendes ein:

- Eigenschaften und Wert der IP-Adresse
- Identifikaktion des Benutzers im Netzwerk
- WinCC Internet Settings

Hinweis

Sicherheit im Projekt

Mit der Systemfunktion „ÖffneDialogSystemsteuerung“ umgehen Sie den SecureMode am Bediengerät. Um die Sicherheit Ihres Projekts zu gewähren, treffen Sie die entsprechenden Vorkehrungen.

Verwendung in der Funktionsliste

ÖffneDialogSystemsteuerung (Dialog)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Dialog

Legt den zu öffnenden Dialog der Systemsteuerung fest.

- PROFINET_X1: Einstellung der IP-Adresse und der Ethernet-Parameter
- PROFINET_X3: Einstellung der IP-Adresse und der Ethernet-Parameter; nur bei Comfort Panel KP 1500, TP 1500; TP1900, TP2200
- WinCC Internet Settings: Einstellung für Web Server, E-Mail-Benachrichtigung, sofern das Bediengerät diese Funktionen unterstützt
- Network ID: Einstellung zur Identifikation im Netzwerk, sofern das Bediengerät diese Funktionen unterstützt

ÖffneEingabeaufforderung

Beschreibung

Öffnet eine Windows-Eingabeaufforderung.

Verwenden Sie diese Systemfunktion, wenn Sie z.B. Dateien kopieren wollen oder eine andere Anwendung aufrufen wollen.

Verwendung in der Funktionsliste

ÖffneEingabeaufforderung

Verwendung in benutzerdefinierten Funktionen

OpenCommandPrompt

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

ÖffneInternetExplorer

Beschreibung

Öffnet den Internet Explorer am Bediengerät.

Wenn der Internet Explorer bereits geöffnet ist, wird der Internet Explorer mit Aufruf der Systemfunktion geschlossen und erneut geöffnet.

Hinweis

Der Internet Explorer speichert Daten temporär im DRAM-Dateisystem des Bediengeräts, z.B. zuletzt aufgerufene Web-Seiten.

Diese Daten können Sie mit der Systemfunktion "SichereRAMDateisystem" sichern, damit sie auch nach dem Neustart des Bediengeräts zur Verfügung stehen.

Verwendung in der Funktionsliste

ÖffneInternetExplorer (Startseite)

Verwendung in benutzerdefinierten Funktionen

OpenInternetExplorer Start_page

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Startseite

Die Seite, die beim Aufruf des Internet Explorer geladen wird, z.B. "www.siemens.com".

ÖffneSystemsteuerung

Beschreibung

Öffnet ein Fenster, das die Windows-CE-Systemsteuerung anzeigt. Auf einem PC können Sie diese Systemfunktion nicht verwenden.

Mit dieser Systemfunktion stellen Sie unter anderem am Windows CE-basierten Bediengerät Folgendes ein:

- Drucker auswählen
- Transfer-Eigenschaften wählen
- Bildschirmschoner konfigurieren
- Flash-Speicher konfigurieren

Hinweis

Kein Backup oder Restore während laufender Runtime

Backup- und Restore-Funktionen dürfen nur ausgeführt werden, wenn Runtime beendet ist. Andernfalls können unerwünschte Effekte auftreten, z. B. Anzeigefehler.

Verwendung in der Funktionsliste

ÖffneSystemsteuerung

Verwendung in benutzerdefinierten Funktionen

OpenControlPanel

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

PROFIsafeTrennen

Beschreibung

Trennt die PROFIsafe-Verbindung für den fehlersicheren Betrieb zwischen einem KTP Mobile Panel und der Steuerung.

Nach dem Ausführen der Systemfunktion "PROFIsafeTrennen" kann der Stecker des KTP Mobile Panels von der Steuerung getrennt werden, ohne dass die Anlage einen Fehler meldet.

Verwendung in der Funktionsliste

PROFIsafeTrennen

Verwendung in benutzerdefinierten Funktionen

TerminatePROFIsafe

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Siehe auch

Systemfunktionen für Mobile Panels (Seite 1554)

Geräteabhängigkeit (Seite 1536)

SendeEMail

Beschreibung

Schickt eine E-Mail vom Bediengerät aus an den angegebenen Empfänger.

Verwenden Sie diese Systemfunktion, wenn z.B. im Servicefall die Meldung direkt an den zuständigen Servicetechniker weitergeleitet werden soll.

Hinweis

Um Meldungen zusätzlich als E-Mail zu versenden, muss das HMI-System über einen E-Mail-Client verfügen.

Verwendung in der Funktionsliste

SendeEMail (Adresse, Betreff, Text, Antwortadresse)

Verwendung in benutzerdefinierten Funktionen

SendEMail Address, Subject, Text, Return_address

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Adresse

Die E-Mail-Adresse des Empfängers.

Betreff

Die Betreffzeile der E-Mail.

Text

Der Text, der in der E-Mail versendet wird.

Antwortadresse

Die E-Mail-Adresse, an die der Empfänger die Antwort auf diese E-Mail senden soll.

SetzeSommerzeit

Beschreibung

Die Systemfunktion "SetzeSommerzeit" schaltet die Uhrzeit des Bediengeräts auf Sommerzeit bzw. Winterzeit um.

Die Zeitumstellung wird unmittelbar nach Ausführung der Systemfunktion vorgenommen.

Hinweis

Die Systemfunktion "SetzeSommerzeit" unterstützt keine Zeitzonen ohne Sommerzeit.

Hinweis

Windows 7

Die Systemfunktion "SetzeSommerzeit" wird auf PC-basierten Bediengeräten unter Windows 7 nicht unterstützt.

Verwendung in der Funktionsliste

SetzeSommerzeit(Sommerzeit)

Verwendung in benutzerdefinierten Funktionen

SetDaylightSavingTime Daylight_saving_time

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Sommerzeit

Legt fest, ob am Bediengerät die Sommerzeit eingestellt wird:

0 = Sommerzeit ist nicht aktiviert

1 = Sommerzeit ist aktiviert

SchaltflächeDrücken

Beschreibung

Sie projektieren die Systemfunktion an die Funktionstasten eines Bediengeräts. Die Systemfunktion "SchaltflächeDrücken" löst am angegebenen Bildobjekt die Funktion aus, die dem Ereignis "Taste Drücken" zugewiesen ist.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. eine Schaltfläche im Bild über eine Funktionstaste des Bediengeräts bedienen wollen.

Hinweis

Sie müssen die Systemfunktionen "SchaltflächeDrücken" und "SchaltflächeLoslassen" immer gemeinsam projektieren. Wenn Sie also an das Ereignis "Taste Drücken" einer Funktionstaste die Systemfunktion "SchaltflächeDrücken" projektieren, dann projektieren Sie an das Ereignis "Loslassen" derselben Funktionstaste die Systemfunktion "SchaltflächeLoslassen".

Verwendung in der Funktionsliste

SchaltflächeDrücken (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, an dem das Ereignis ausgelöst wird.

SchaltflächeLoslassen

Beschreibung

Die Systemfunktion kann nur an die Funktionstasten eines Bediengeräts projektiert werden und löst am angegebenen Bildobjekt das Ereignis "Taste Loslassen" aus.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. eine Schaltfläche im Bild über eine Funktionstaste des Bediengeräts bedienen wollen.

Hinweis

Sie müssen die Systemfunktionen "SchaltflächeDrücken" und "SchaltflächeLoslassen" immer gemeinsam projektieren. Wenn Sie also an das Ereignis "Taste Drücken" einer Funktionstaste die Systemfunktion "SchaltflächeDrücken" projektieren, dann projektieren Sie an das Ereignis "Taste Loslassen" derselben Funktionstaste die Systemfunktion "SchaltflächeLoslassen".

Verwendung in der Funktionsliste

SchaltflächeLoslassen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, an dem das Ereignis ausgelöst wird.

SichereRAMDateisystem

Beschreibung

Sichert das RAM-Dateisystem auf dem Speichermedium des Bediengeräts.

Nach dem Neustart des Bediengeräts werden die Daten automatisch wieder in das RAM-Dateisystem geladen.

Anwendungen wie der Internet Explorer speichern Daten (z.B. zuletzt aufgerufene Web-Seiten) temporär im DRAM-Dateisystem des Bediengeräts.

Verwendung in der Funktionsliste

SichereRAMDateisystem

Verwendung in benutzerdefinierten Funktionen

BackupRAMFileSystem

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

SimuliereSystemtaste

Beschreibung

Simuliert das Verhalten einer Systemtaste. Verwenden Sie diese Systemfunktion, wenn auf dem Bediengerät eine Systemtaste nicht vorhanden ist. Z. B. die "ACK"-Taste, die "Eingabe"-Taste oder der Nummern-Block.

Verwendung in der Funktionsliste

SimuliereSystemtaste(Systemtaste)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Systemtaste

Systemtaste, deren Verhalten simuliert werden soll.

Systemtaste "+/-"

Bei der Systemfunktion SimuliereSystemtaste wird die Systemtaste "+/-" ausschließlich bei den folgenden Bediengeräten unterstützt:

- KP300 Basic
- KP400 Basic
- KTP400 Basic mono PN
- KTP400 Basic color PN
- KTP400 Basic color PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic color PN
- KTP600 Basic color DP
- KTP700 Basic PN/DP
- KTP900 Basic PN
- KTP1000 Basic PN
- KTP1000 Basic DP
- KTP1200 Basic PN/DP
- Comfort Panels

Bei allen anderen Bediengeräten verwenden Sie die Systemtasten "+" und "-" getrennt.

SimuliereVariable

Beschreibung

Simuliert das Verhalten von Variablen und dynamischen Objekten wie Textlisten, ohne dass das Bediengerät an eine Steuerung angeschlossen ist. Die Systemfunktion können Sie z. B. and das Ereignis "Aufgebaut" eines Bildes projektieren.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. die Funktionalität eines Projekts präsentieren wollen.

Zur Simulation können nur Variablen vom Datentyp Integer verwendet werden.

Hinweis

Wenn Sie auf einem Basic Panel die Systemfunktion "SimuliereVariable" in Verbindung mit einer kleinen Zykluszeit verwenden, kann das Bediengerät überlastet werden.

Verwendung in der Funktionsliste

SimuliereVariable (Variable, Zyklus, Maximaler Wert, Minimaler Wert, Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Variable

Die Variable, deren Wert verändert wird.

Zyklus

Der Faktor, mit dem der Basiszyklus von 200 Millisekunden multipliziert wird. Der Zyklus legt fest, wann der Variablenwert um den angegebenen Wert verändert wird. Möglicher Zyklus zwischen 1 und 32767.

Maximaler Wert

Der Maximalwert, den der Variablenwert bei der Simulation annehmen kann. Der Maximalwert muss größer als der Minimalwert sein, aber kleiner gleich 32767.

Minimaler Wert

Der Minimalwert, den der Variablenwert bei der Simulation annehmen kann. Der Minimalwert muss kleiner als der Maximalwert sein, aber größer gleich -32768.

Wert

Der Wert, um den der Variablenwert mit jedem Zyklus verändert wird. Möglicher Wert zwischen -32768 und 32767.

- Ein positiver Wert erhöht den Variablenwert. Wenn der Maximalwert erreicht wird, wird der Variablenwert nach dem nächsten Aktualisierungszyklus auf den Minimalwert gesetzt.
- Ein negativer Wert verringert den Variablenwert. Wenn der Minimalwert erreicht wird, wird der Variablenwert nach dem nächsten Aktualisierungszyklus auf den Maximalwert gesetzt.

StarteProgramm

Beschreibung

Startet das angegebene Programm auf dem Bediengerät.

Die Runtime-Software läuft im Hintergrund weiter. Meldungen werden weiterhin ausgegeben und Prozesswerte werden weiterhin aktualisiert.

Wenn Sie die angegebene Anwendung beenden, wird am Bediengerät das Bild angezeigt, das beim Ausführen der Systemfunktion aktiv war.

Verwenden Sie diese Systemfunktion, wenn Sie z.B. Rezepturdatensätze am Bediengerät in MS Excel bearbeiten wollen.

Hinweis

Wenn auf dem Bediengerät Windows CE installiert ist, prüfen Sie während der Projektierung, ob sich die gewünschte Anwendung mit dieser Systemfunktion starten lässt.

Sie können mit dieser Systemfunktion alle Anwendungen starten, die Sie im Dialog "Ausführen" von Windows CE starten können.

Die zu startende Anwendung muss auf dem Bediengerät installiert sein.

Hinweis

Wenn auf einem Mobile Panel oder einem Comfort Panel eine Anwendung über die Systemfunktion "StarteProgramm" nicht ausgeführt werden kann, prüfen Sie die Pfadangaben der Systemfunktion. Auf Mobile Panels und Comfort Panels befinden sich einige Anwendungen im Ordnern"\flash\addons".

Verwendung in der Funktionsliste

StarteProgramm (Programmname, Programmparameter, Darstellung, Auf Programmende warten)

Verwendung in benutzerdefinierten Funktionen

StartProgram Program_name, Program_parameters, Display_mode, Wait_for_program_to_end

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Programmname

Name und Pfad des Programms, das gestartet wird. Bei diesem Parameter wird die Groß- und Kleinschreibung beachtet.

Hinweis

Wenn der Pfad Leerzeichen enthält, wird das Programm nur dann korrekt gestartet, wenn der Pfad in Anführungszeichen gesetzt ist z. B. "C:\Program Files\START\start.exe".

Programmparameter

Die Parameter, die Sie bei Start des Programms übergeben z.B. eine Datei, die nach Programmstart geöffnet wird.

Die Beschreibung der benötigten Parameter finden Sie in der Dokumentation des zu startenden Programms.

Darstellung

Legt fest, wie das Programmfenster am Bediengerät dargestellt wird:

0 (hmiShowNormal) = Normal

1 (hmiShowMinimized) = Minimiert

2 (hmiShowMaximized) = Maximiert

3 (hmiShowMinimizedAndInactive) = Minimiert und inaktiv

Auf Programmende warten

Legt fest, ob nach Beenden des aufgerufenen Programms zurück zum Projekt gewechselt wird:

0 (hmiNo) = Nein: Es wird nicht zum Projekt gewechselt

1 (hmiYes) = Ja: Es wird zum Projekt gewechselt

Hinweis

Der Parameter "Auf Programmende warten" steht nur für Runtime Advanced und Panels zur Verfügung.

StatusSteuernLeseWerte

Beschreibung

Startet oder stoppt die Aktualisierung der Werte in der Beobachtungstabelle. Die Werte werden so lange aus der Steuerung gelesen, die mit dem Bediengerät verbunden ist, bis die Aktualisierung gestoppt wird.

Hinweis

Solange die Aktualisierung der Werte gestartet ist, sind in den Eingabefeldern der Beobachtungstabelle keine Eingaben möglich.

Verwendung in der Funktionsliste

StatusSteuernLeseWerte (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Beobachtungstabelle, in welche Daten aus der Steuerung geschrieben werden.

Siehe auch

Beobachtungstabelle (Seite 355)

StatusSteuernSchreibeWerte

Beschreibung

Schreibt die Werte aus der Beobachtungstabelle in die Steuerung, die mit dem Bediengerät verbunden ist

Verwendung in der Funktionsliste

StatusSteuernSchreibeWerte (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Beobachtungstabelle, aus der Daten in die Steuerung geschrieben werden.

Siehe auch

Beobachtungstabelle (Seite 355)

SteuereSmartServer

Beschreibung

Startet oder stoppt den Sm@rtServer.

Verwendung in der Funktionsliste

SteuereSmartServer (Modus)

Verwendung in benutzerdefinierten Funktionen

ControlSmartServer Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob der Sm@rtServer gestartet oder gestoppt wird.

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

0 (hmiStop) = Stoppen: Sm@rtServer wird gestoppt

1 (hmiStart) = Starten: Sm@rtServer wird gestartet

SteuereWebServer

Beschreibung

Startet oder stoppt den Webserver.

Verwendung in der Funktionsliste

SteuereWebServer (Modus)

Verwendung in benutzerdefinierten Funktionen

ControlWebServer Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob der Webserver gestartet oder gestoppt wird.

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

0 (hmiStop) = Stoppen: Webserver wird gestoppt

1 (hmiStart) = Starten: Webserver wird gestartet

StoppeRuntime

Beschreibung

Beendet die Runtime-Software und damit das laufende Projekt am Bediengerät.

Verwendung in der Funktionsliste

StoppeRuntime (Modus)

Verwendung in benutzerdefinierten Funktionen

StopRuntime Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob nach dem Beenden von Runtime auch das Betriebssystem heruntergefahren wird:

0 (hmiStopRuntime) = Runtime: Betriebssystem wird nicht heruntergefahren

1 (hmiStopRuntimeAndOperatingSystem) = Runtime und Betriebssystem: Betriebssystem wird heruntergefahren (bei WinCE nicht möglich)

Beispiel

Der folgende Programmcode fährt Runtime und Betriebssystem herunter.

```
{

//Stop runtime and shutdown
StopRuntime (hmiStopRuntimeAndOperationSystem);

}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

TextAbrufen

Beschreibung

Ermittelt den Listeneintrag aus einer Textliste. Das Ergebnis ist abhängig vom Wert und von der gewählten Runtime-Sprache. Das Ergebnis wird in eine Variable vom Datentyp "String" gespeichert.

Verwendung in der Funktionsliste

TextAbrufen (Ausgabetext, Index, Sprache, Textliste)

Verwendung in benutzerdefinierten Funktionen

LookupText Output_text, Index, Language, Text_list

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Ausgabetext

Die Variable, in der das Ergebnis geschrieben wird.

Index

Die Variable, die den Wert des Listeneintrags festlegt.

Sprache

Legt fest, in welcher Runtime-Sprache der Listeneintrag ermittelt wird.

- Runtime-Sprache
  Sprachkürzel gemäß der VBScript-Referenz , z. B. "de-DE" für Deutsch (Deutschland) oder "en-US" für Englisch (Vereinigte Staaten von Amerika). Die Auswahl ist abhängig von den aktivierten Runtime-Sprachen.
- Variable
  Die Variable, welche die Sprache enthält. Die Runtime-Sprache geben Sie als Dezimalwert der Länder-Kennung an, z. B. 1031 für Deutsch - Standard, 1033 für Englisch - USA. Nähere Hinweise hierzu finden Sie in den Grundlagen von VBScript bei "Gebietsschema-ID (LCID)-Diagramm.
- Integer
  Die Zahl, die der Reihenfolge der Runtime-Sprachen bei Sprachumschaltung entspricht. Die Auswahl ist abhängig von den aktivierten Runtime-Sprachen, z. B. "0" für die Sprache, die beim ersten Runtime-Start erscheint. Nähere Hinweise hierzu finden Sie im Thema "Sprachen in Runtime".

Textliste

Legt die Textliste fest. Aus der Textliste wird der Listeneintrag ausgelesen.

TrenneVariableVonHandrad

Beschreibung

Trennt die Variable, die mit dem Bedienelement Handrad verbunden ist, und verbindet das Handrad wieder mit der globalen Variablen.

Verwendung in der Funktionsliste

TrenneVariableVonHandrad

Verwendung in benutzerdefinierten Funktionen

-

Parameter

--

VerbindeVariableMitHandrad

Beschreibung

Verbindet die Variable mit dem Bedienelement Handrad. Wenn das Handrad bedient wird, ändert sich der Wert dieser Variablen. Die Verbindung kann mit der Systemfunktion "TrenneVariableVonHandrad" wieder getrennt werden.

Verwendung in der Funktionsliste

VerbindeVariableMitHandrad (Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Wert

Name der Variablen, die mit dem Bedienelement Handrad verbunden wird.

VerringereFokussiertenWert

Beschreibung

Subtrahiert den angegebenen Wert vom Wert der Variablen, die mit dem Bildobjekt verbunden ist, das aktuell den Fokus besitzt.

Diese Systemfunktion kann nur an Funktionstasten verwendet werden.

Verwendung in der Funktionsliste

VerringereFokussiertenWert (Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Wert

Der Wert, der vom Variablenwert subtrahiert wird.

ZeigeHilfetext

Verwendung

Zeigt den projektierten Tooltip des ausgewählten Objekts an.

Wenn die Systemfunktion an eine Funktionstaste projektiert ist, wird der Tooltip zu dem Bildobjekt angezeigt, das gerade den Fokus besitzt. Wenn gleichzeitig ein Tooltip zum Bild selbst projektiert ist, kann zu diesem mit der Taste <Enter> oder durch Doppelklick auf das Hilfefenster umgeschaltet werden.

Wenn die Systemfunktion an eine Schaltfläche projektiert ist, wird nur der Tooltip zum aktuellen Bild angezeigt. Wenn gleichzeitig ein Tooltip an die Schaltfläche selbst projektiert ist, wird zuerst der Tooltip zur Schaltfläche angezeigt. Mit der Taste <Enter> oder durch Doppelklick auf das Hilfefenster wird zum Tooltip des aktuellen Bilds umgeschaltet.

Hinweis

Solange das Hilfefenster geöffnet ist, lässt sich kein anderes Bildobjekt bedienen. Um die Bildobjekte zu bedienen, schließen Sie das Hilfefenster.

Schließen des Hilfefensters

Es gibt folgende Möglichkeiten das Hilfefenster zu schließen:

Bei Tasten-Bedienung:

- Erneutes Drücken der Taste <HELP>
- Drücken der Taste <ESC>

Bei Touch-Bedienung:

- Bedienen der Schaltfläche ❌

Verwendung in der Funktionsliste

ZeigeHilfetext (Darstellung)

Verwendung in benutzerdefinierten Funktionen

ShowOperatorNotes Display_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Darstellung

Legt fest, ob der projektierte Tooltip eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Projektierter Tooltip wird ausgeblendet

1 (hmiOn) = Ein: Projektierter Tooltip wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

ZeigeSoftwareVersion

Beschreibung

Blendet die Versionsnummer der Runtime-Software ein oder aus.

Verwenden Sie diese Systemfunktion, wenn Sie z.B. in einem Servicefall die verwendete Version der Runtime-Software benötigen.

Verwendung in der Funktionsliste

ZeigeSoftwareVersion (Darstellung)

Verwendung in benutzerdefinierten Funktionen

ShowSoftwareVersion Display_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Darstellung

Legt fest, ob die Versionsnummer eingeblendet wird:

0 (hmiOff) = Aus: Versionsnummer wird nicht eingeblendet

1 (hmiOn) = Ein: Versionsnummer wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

ZeigePopupBild

Beschreibung

Ruft das Pop-up-Bild auf, z. B.beim Betätigen einer Schaltfläche.

Als Koordinaten können Sie einen konstanten Wert eingeben oder eine Variable zuweisen. Falls das projektierte Pop-up-Bild nicht sichtbar oder nur teilweise sichtbar ist, werden die Koordinaten auf 0.0 gesetzt.

Unabhängig von der Größe des Permanentbereichs, ist die Anfangsposition der Koordinaten immer 0.0.

Verwendung in der Funktionsliste

ZeigePopupBild (Name des Bildes, Koordinate X, Koordinate Y, Anzeigemodus, Animation, Animationsgeschwindigkeit)

Verwendung in benutzerdefinierten Funktionen

ShowPopupScreen Screen_name, X_coordinate, Y_coordinate, Display_mode, Animation, Animation_speed

Parameter

Name des Bildes

Legt den Namen des Pop-up-Bildes fest, das in Runtime beim Betätigen der Schaltfläche erscheint.

Koordinate X

Position des Pop-up-Bildes im aktuellem Bild auf der X-Achse

Koordinate Y

Position des Pop-up-Bildes im aktuellem Bild auf der Y-Achse

Anzeigemodus

Legt den Modus für das Pop-up-Bild fest:

Umschalten

Aus

Ein

Animation

Legt die Richtung fest, aus der das Pop-up-Bild eingeblendet wird:

Aus

Links

Oben

Rechts

Unten

Animationsgeschwindigkeit

Legt die Geschwindigkeit fest, mit der das Pop-up-Bild eingeblendet wird:

Langsam

Mittel

Schnell

ZeigePopupBildGrößeVeränderbar

Beschreibung

Ruft das Pop-up-Bild mit einer definierten Größe auf. Mit dieser Systemfunktion können Sie ein Pop-up-Bild in einer anderen Größe aufrufen, für die Navigation im Inhalt des Pop-up-Bildes werden die Bildlaufleisten angezeigt.

Als Koordinaten können Sie einen konstanten Wert eingeben oder eine Variable zuweisen. Falls das projektierte Pop-up-Bild nicht sichtbar oder nur teilweise sichtbar ist, werden die Koordinaten auf 0.0 gesetzt. Unabhängig von der Größe des Permanentbereichs, ist die Anfangsposition der Koordinaten immer 0.0.

Außerdem definieren Sie die angezeigte Breite und Höhe des Pop-up-Bildes, die von den Größen des projektierten Pop-up-Bildes abweichen können. In diesem Fall werden im Pop-up-Bild die Bildlaufleisten angezeigt.

Hinweis

Bedienung auf Bediengeräten mit Tasten

In Pop-up-Bildern scrollen Sie mithilfe von Tastenkombination <ALT>+<Pfeiltaste>.

Nur eine Bildlaufleiste anzeigen

Falls Sie bei den Parametern der Systemfunktion nur eine der Größen kleiner als die Größe des projektierten Pop-up-Bilds angeben, werden im Pop-up-Bild die beiden Bildlaufleisten angezeigt. Sie haben zwei Möglichkeiten, um nur eine Bildlaufleiste im Pop-up-Bild zu konfigurieren:

- Verringern Sie die Breite/Höhe des projektierten Pop-up-Bildes um die Größe der gerätespezifischen Bildlaufleiste.
- Legen Sie für den einen Größenparameter den Wert fest, welcher der tatsächlichen Größe des Pop-up-Bildes entspricht und rechnen Sie dazu die gerätespezifische Größe der Bildlaufleiste. Den Wert für die andere Größe legen Sie nach Ihrem Bedarf fest.
  Beispiel:
  Sie möchten ein Pop-up-Bild mit der projektierten Größe von 500 x 420 in der Größe 500 x 300 mit einer vertikalen Bildlaufleiste aufrufen. Legen Sie für den Parameter Breite den Wert 534 fest, der der tatsächlichen Breite des projektierten Pop-up-Bilds zuzüglich der Größe der Bildlaufleiste von 34 entspricht. Geben Sie für die Höhe den Wert 300 an. In Runtime wird das Pop-up-Bild in der Größe 500 x 300 aufgerufen und nur die vertikale Bildlaufleiste wird angezeigt.

Gerätetyp (Version V14)	Breite	Höhe
Bediengeräte mit der Displaygröße von 4" Comfort Panels mit der Displaygröße von 19" Runtime Advanced	25	25
Bediengeräte mit der Displaygröße von 9" Comfort Panels mit der Displaygröße von 15" Comfort Panels mit der Displaygröße von 22"	27	27
Bediengeräte mit der Displaygröße von 7" Bediengeräte mit der Displaygröße von 12"	34	34

Verwendung in der Funktionsliste

ZeigePopupBildGrößeVeränderbar (Name des Bildes, Koordinate X, Koordinate Y, Breite, Höhe, Anzeigemodus, Animation, Animationsgeschwindigkeit)

Verwendung in benutzerdefinierten Funktionen

ShowPopupScreenSizable Screen_name, X_coordinate, Y_coordinate, Width, Height, Display_mode, Animation, Animation_speed

Parameter

Name des Bildes

Legt den Namen des Pop-up-Bildes fest, das in Runtime beim Betätigen der Schaltfläche erscheint.

Koordinate X

Position des Pop-up-Bildes im aktuellem Bild auf der X-Achse

Koordinate Y

Position des Pop-up-Bildes im aktuellem Bild auf der Y-Achse

Breite

Legt die Breite des Pop-up-Bildes fest. Die maximale Breite darf die Bildbreite für das projektierte Bediengerät nicht überschreiten.

Höhe

Legt die Höhe des Pop-up-Bildes fest. Die maximale Höhe darf die Bildbreite für das projektierte Bediengerät nicht überschreiten.

Anzeigemodus

Legt den Modus für das Pop-up-Bild fest:

Umschalten

Aus

Ein

Animation

Legt die Richtung fest, aus der das Pop-up-Bild eingeblendet wird:

Aus

Links

Oben

Rechts

Unten

Animationsgeschwindigkeit

Legt die Geschwindigkeit fest, mit der das Pop-up-Bild eingeblendet wird:

Langsam

Mittel

Schnell

Siehe auch

Systemfunktionen für Runtime Advanced (Seite 1562)

ZeigeSlideInBild

Beschreibung

Ruft das Slide-in-Bild auf, z. B. beim Betätigen einer Schaltfläche.

Verwendung in der Funktionsliste

ZeigeSlideInBild (Bildname, Modus)

Verwendung in benutzerdefinierten Funktionen

ShowSlideInScreen SlideInScreen_name, Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Bildname

Legt das Slide-in-Bild fest, das in Runtime beim Betätigen der Schaltfläche erscheint:

Slide-in-Bild oben

Slide-in-Bild unten

Slide-in-Bild links

Slide-in-Bild rechts

Modus

Legt den Modus für das Slide-in-Bild fest:

Umschalten

Aus

Ein

Archive

ArchiviereProtokolldatei

Beschreibung

Diese Systemfunktion verschiebt oder kopiert ein Archiv zur Langzeitarchivierung an einen anderen Speicherort.

Verwenden Sie bei Audit Trails immer den Modus "Verschieben (hmiMove)", da sonst durch doppelte Datenhaltung gegen die FDA-Richtlinie verstoßen wird.

Führen Sie vor "ArchiviereProtokolldatei" immer die Systemfunktion "SchließeAlleArchive" aus.

Führen Sie nach Abschluss dieser Systemfunktion die Systemfunktion "ÖffneAlleArchive" aus.

Im Modus "Archiv kopieren" werden die Archive erst wieder geöffnet, nachdem das Archiv erfolgreich kopiert wurde, oder wenn beim Kopiervorgang eine Zeitüberschreitung eintritt. Im Modus "Archiv verschieben" werden die zu verschiebenden Archive umbenannt und die neuen Archive werden sofort geöffnet. Für das Verschieben der umbenannten Archive wird ein Auftrag abgesetzt, der alle 300 Sekunden versucht diese zu verschieben, falls der Server nicht

erreichbar ist. Der Auftrag bleibt auch nach einem Neustart von Runtime bestehen, bis er ausgeführt ist.

Verwendung in der Funktionsliste

ArchiviereProtokolldatei (Archivtyp, Archiv, Verzeichnisname, Modus)

Verwendung in benutzerdefinierten Funktionen

ArchiveLogFile Log_type, Log, Directory_name, Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest:

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

2 (hmiAuditrailArchive)= Audit-Trail-Archiv Verfügbar bei GMP-konformen Projekten. Nähere Hinweise hierzu finden Sie unter "GMP-konforme Projektierung aktivieren".

Archiv

Name des Archivs, das archiviert wird.

Verzeichnisname

Pfad, in den das Archiv gespeichert wird.

Modus

0 (hmiCopy) = Archiv kopieren

1 (hmiMove) = Archiv verschieben

Anwendungsbeispiel

Sie wollen ein Archiv vom lokalen Speichermedium auf den Server verschieben, um das Archiv in regelmäßigen Abständen zu sichern.

Hinweise zur Projektierung

Legen Sie im Aufgabenplaner eine Aufgabe an, die täglich zu einer bestimmten Uhrzeit ausgeführt wird. Projektieren Sie an der Aufgabe folgende Funktionsliste:

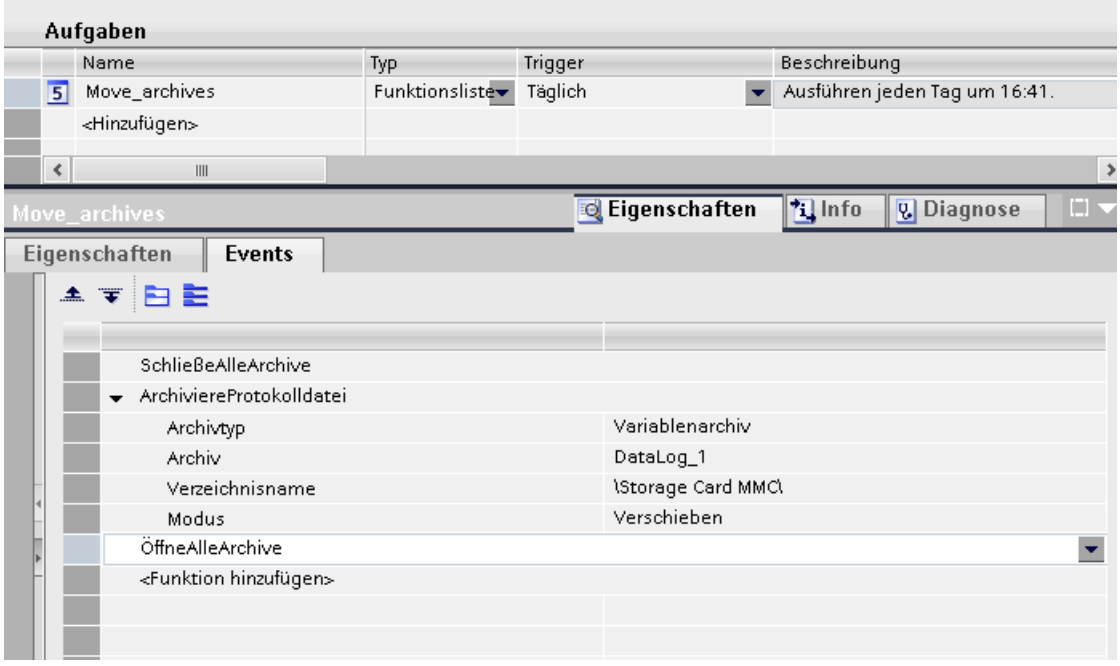

Ablauf am Bediengerät

- Alle geöffneten Archive werden geschlossen.
- Das angegebene Archiv wird auf den Server verschoben.
- Alle geschlossenen Archive werden wieder geöffnet.

ArchiviereVariable

Beschreibung

Speichert den Wert der angegebenen Variablen im angegebenen Variablenarchiv.

Verwenden Sie die Systemfunktion, wenn Sie einen Prozesswert zu einem bestimmten Zeitpunkt archivieren wollen.

Verwendung in der Funktionsliste

ArchiviereVariable (Variable)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Variable

Die Variable, deren Wert archiviert wird. Die Variable wird in dem Archiv abgelegt, das für die angegebene Variable projektiert ist.

KopiereArchiv

Beschreibung

Kopiert den Inhalt eines Archivs in ein anderes Archiv. Dabei können Sie Variablenwerte nur in andere Variablenarchive und Meldungen nur in andere Meldearchive kopieren.

Hinweis

Wenn Sie ein Archiv mit der Systemfunktion "KopiereArchiv" kopieren, kann es vorkommen, dass in der Kopie des Archivs nationale Sonderzeichen in den archivierten Meldetexten durch Fremdapplikationen nicht lesbar sind. WinCC Runtime ist davon nicht betroffen. Die kopierten Archive können von WinCC Runtime ohne Fehler gelesen werden.

Hinweis

Beim Kopieren der Umlaufarchive werden 80 % der Archiveinträge kopiert. 20 % der Einträge werden nicht mitkopiert, weil der Speicherplatz standardmäßig für Pufferüberlauf reserviert ist.

Verwendung in der Funktionsliste

KopiereArchiv (Archivtyp, Zielarchiv, Quellarchiv, Modus, Quellarchiv löschen)

Verwendung in benutzerdefinierten Funktionen

CopyLog Log_type, Destination log, Source_log, Mode, Delete_source_log

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest:

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

Zielarchiv

Name des Archivs, in das die Einträge kopiert werden (Zielarchiv).

Quellarchiv

Name des Archivs, aus dem die Einträge kopiert werden (Quellarchiv).

Modus

Legt fest, wie im Zielarchiv mit den kopierten Einträgen verfahren wird:

0 (hmiOverwrite) = Überschreiben: Vorhandene Einträge werden überschrieben.

2 (hmiAppend) = Anhängen: Die Einträge werden am Ende des Zielarchivs eingefügt. Wenn die projektierte Größe des Zielarchivs erreicht ist, wird das Zielarchiv wie ein Umlaufarchiv behandelt.

Quellarchiv löschen

Legt fest, ob das Quellarchiv nach dem Kopieren gelöscht wird.

0 (hmiNo) = Nein: Wird nicht gelöscht.

1 (hmiYes) = Ja: Wird gelöscht.

LöscheArchiv

Beschreibung

Löscht alle Datensätze in dem angegebenen Archiv.

Verwendung in der Funktionsliste

LoescheArchiv (Archivtyp, Archiv)

Verwendung in benutzerdefinierten Funktionen

ClearLog Log_type, Log

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest.

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

2 (hmiAuditTrailArchive) = Audit Trail-Archiv. Verfügbar bei GMP-konformen Projekten. Nähere Nähere Hinweise hierzu finden in "GMP-konforme Projektierung aktivieren".

Archiv

Name des Archivs, aus dem alle Einträge gelöscht werden.

ÖffneAlleArchive

Beschreibung

Stellt die Verbindung zwischen WinCC und den Archiven wieder her. Die Archivierung kann fortgesetzt werden.

Hinweis

Um die Archivierung wieder zu starten, führen Sie die Systemfunktion "StarteArchivierung" aus.

Verwendung in der Funktionsliste

ÖffneAlleArchive

Verwendung in benutzerdefinierten Funktionen

OpenAllLogs

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Anwendungsbeispiel

Sie wollen in Runtime den Datenträger wechseln, auf dem die Prozesswerte archiviert werden.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktionen "StoppeArchiverung" und "SchließeAlleArchive" an die Schaltfläche "Close Archive".

Projektieren Sie die Systemfunktionen "ÖffneAlleArchive" und "StarteArchivierung" an die Schaltfläche "Open Archive".

Übergeben Sie als Parameter jeweils den Namen des Archivs, das gestoppt und gestartet werden soll.

Ablauf am Bediengerät

Wenn Sie die Schaltfläche "Close Archive" drücken, wird das angegebene Archiv gestoppt und alle geöffneten Archive geschlossen. Sie können den Datenträger wechseln. Mit der Schaltfläche "Open Archive" öffnen Sie alle Archive. Die Archivierung in das angegebene Archiv wird fortgesetzt.

SchließeAlleArchive

Beschreibung

Trennt die Verbindung zwischen WinCC und allen Archiven.

Verwendung in der Funktionsliste

SchließeAlleArchive

Verwendung in benutzerdefinierten Funktionen

CloseAllLogs

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Anwendungsbeispiel

Sie wollen in Runtime den Datenträger wechseln, auf dem die Prozesswerte archiviert werden.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktion "SchließeAlleArchive" an die Schaltfläche "Close Archive".

Projektieren Sie die Systemfunktionen "ÖffneAlleArchive" und "StarteArchivierung" an die Schaltfläche "Open Archive".

Übergeben Sie als Parameter jeweils den Namen des Archivs, das gestoppt und gestartet werden soll.

Ablauf am Bediengerät

Wenn Sie die Schaltfläche "Close Archive" drücken, werden die geöffneten Archive geschlossen. Sie können den Datenträger wechseln. Während des Datenträgerwechsels wird die Archivierung fortgesetzt. Die zu archivierenden Prozesswerte werden gepuffert. Mit der Schaltfläche "Open Archive" öffnen Sie alle Archive. Die Archivierung wird in dem angegebenen Archiv fortgesetzt. Die gepufferten Prozesswerte werden in das Archiv nachgetragen.

StarteArchivierung

Beschreibung

Startet die Archivierung von Prozesswerten oder Meldungen in dem angegebenen Archiv. Die Funktion kann auch auf Audit-Trails angewendet werden.

Sie können die Archivierung in Runtime mit der Systemfunktion "StoppeArchivierung" unterbrechen.

Verwendung in der Funktionsliste

StarteArchivierung (Archivtyp, Archiv)

Verwendung in benutzerdefinierten Funktionen

StartLogging Log_type, Log

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest:

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

2 (hmiAuditrailArchive) = Audit-Trail

Archiv

Name des Archivs, das gestartet wird.

Siehe auch

Verhalten des Archivs bei Runtime-Start steuern (Seite 644)

StarteNächstesArchiv

Beschreibung

Stoppt die Archivierung von Prozesswerten oder Meldungen für das angegebene Archiv.

Die Archivierung wird in dem nächsten Archiv des segmentierten Umlaufarchivs fortgesetzt, das Sie für das angegebene Archiv projektiert haben.

Wenn Sie für das angegebene Archiv kein segmentiertes Umlaufarchiv projektiert haben, hat die Systemfunktion keine Wirkung.

Verwendung in der Funktionsliste

StarteNächstesArchiv (Archivtyp, Archiv)

Verwendung in benutzerdefinierten Funktionen

StartNextLog Log_type, Log

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest:

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

Archiv

Name des Archivs, für das die Archivierung gestoppt und im nächsten Archiv fortgesetzt wird.

StoppeArchivierung

Beschreibung

Stoppt die Archivierung von Prozesswerten oder Meldungen in dem angegebenen Archiv. Die Funktion kann auch auf Audit-Trails angewendet werden.

Sie können die Archivierung in Runtime mit der Systemfunktion "StarteArchivierung" wieder fortsetzen.

Hinweis

Wenn die Archivierung gestoppt ist, besteht immer noch eine Verbindung zwischen WinCC und den Archivdateien oder der Archivdatenbank. Um diese Verbindung zu trennen, verwenden Sie die Systemfunktion "SchließeAlleArchive".

Verwendung in der Funktionsliste

StoppeArchivierung (Archivtyp, Archiv)

Verwendung in benutzerdefinierten Funktionen

StopLogging Log_type, Log

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Archivtyp

Legt den Typ des Archivs fest:

0 (hmiTagArchive) = Variablenarchiv

1 (hmiAlarmArchive) = Meldearchiv

2 (hmiAuditrailArchive) = Audit-Trail

Archiv

Name des Archivs, das gestoppt wird.

Anwendungsbeispiel

Sie wollen in Runtime den Datenträger wechseln, auf dem die Prozesswerte archiviert werden.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktionen "StoppeArchivierung" und "SchließeAlleArchive" an die Schaltfläche "Close Archive".

Projektieren Sie die Systemfunktionen "ÖffneAlleArchive" und "StarteArchivierung" an die Schaltfläche "Open Archive".

Übergeben Sie als Parameter jeweils den Namen des Archivs, das gestoppt und gestartet werden soll.

Ablauf am Bediengerät

Wenn Sie die Schaltfläche "Close Archive" drücken, wird das angegebene Archiv gestoppt und alle geöffneten Archive geschlossen. Sie können den Datenträger wechseln. Mit der Schaltfläche "Open Archive" öffnen Sie alle Archive und setzen die Archivierung in dem angegebenen Archiv wieder fort.

Benutzerverwaltung

Abmelden

Beschreibung

Meldet den aktuellen Benutzer am Bediengerät ab.

Verwendung in der Funktionsliste

Abmelden

Verwendung in benutzerdefinierten Funktionen

Logoff

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Anmelden

Beschreibung

Meldet den aktuellen Benutzer am Bediengerät an.

Verwendung in der Funktionsliste

Anmelden (Kennwort, Benutzername)

Verwendung in benutzerdefinierten Funktionen

Logon Password, User_name

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Kennwort

Die Variable, aus der das Kennwort des anzumeldenden Benutzers gelesen wird. Wenn der Benutzer angemeldet ist, wird das Kennwort in der Variablen gelöscht.

Benutzername

Die Variable, aus welcher der Benutzername des anzumeldenden Benutzers gelesen wird.

ExportiereImportiereBenutzerverwaltung

Beschreibung

Exportiert alle Benutzer der Benutzerverwaltung des gerade aktiven Projekts in die angegebene Datei oder importiert die Benutzer aus der angegebenen Datei in das gerade aktive Projekt.

In der Benutzerverwaltung werden Benutzer, deren Passwörter und Rechte gespeichert.

Beim Import werden alle Benutzer überschrieben. Die importierten Benutzer sind sofort gültig.

Hinweis

Das Exportieren und Importieren der Benutzerverwaltung aus der V14 wird auf den Bediengeräten mit einer älteren Runtime-Version als V14 nicht unterstützt.

Verwendung in der Funktionsliste

ExportiereImportiereBenutzerverwaltung (Dateiname, Richtung)

Verwendung in benutzerdefinierten Funktionen

ExportImportUserAdministration File_name, Direction

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Dateiname

Name der Datei, welche die Passwörter enthält oder in der die Passwörter geschrieben werden. Geben Sie den Ablageort und die Dateiendung (*.txt) mit an, z. B. "C:\TEMP \Passwords.txt".

Hinweis

Wenn Sie als Ablageort eine Storage Card verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Dateiname>".

Richtung

Legt fest, ob Passwörter exportiert oder importiert werden:

0 (hmiExport) = Export: Passwörter werden exportiert.

1 (hmiImport) = Import: Passwörter werden importiert.

Siehe auch

Benutzerverwaltung exportieren (Seite 1412)

Benutzerverwaltung importieren (Seite 1413)

LeseBenutzername

Beschreibung

Schreibt den Benutzernamen des gerade am Bediengerät angemeldeten Benutzers in die angegebene Variable.

Wenn die angegebene Variable eine Steuerungsanbindung besitzt, steht der Benutzername auch in der Steuerung zur Verfügung. Mit dieser Systemfunktion können Sie z.B. eine benutzerabhängige Freischaltung bestimmter Funktionalitäten realisieren.

Verwendung in der Funktionsliste

LeseBenutzername (Variable)

Verwendung in benutzerdefinierten Funktionen

GetUserName Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in die der Benutzername geschrieben wird.

LeseGruppennummer

Beschreibung

Liest die Nummer der Gruppe, welcher der am Bediengerät angemeldete Benutzer angehört, und schreibt sie in die angegebene Variable.

Verwendung in der Funktionsliste

LeseGruppennummer (Variable)

Verwendung in benutzerdefinierten Funktionen

GetGroupNumber Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in welche die Nummer der Gruppe geschrieben wird.

LeseKennwort

Beschreibung

Schreibt das Kennwort des gerade am Bediengerät angemeldeten Benutzers in die angegebene Variable.

Hinweis

Stellen Sie sicher, dass der Wert der angegebenen Variable nicht an anderer Stelle im Projekt angezeigt wird.

Hinweis

Kennwörter von SIMATIC Logon-Benutzern können nicht ausgelesen werden.

Hinweis

Die Systemfunktion "LeseKennwort" ist für Geräte ab der Geräteversion 14.0.0.0 nicht verfügbar.

Wenn diese Systemfunktion in einem Projekt einer älteren Version als V14 verwendet wurde, wird sie nach der Migration in die V14 entfernt. Das mit der Systemfunktion verknüpfte Ereignis wird dadurch ungültig. In der Projektierung wird die Systemfunktion "LeseKennwort" als fehlerhaft angezeigt. Beim Übersetzen wird eine Warnung ausgegeben. Im übersetzten Runtime-Projekt ist die Systemfunktion "LeseKennwort" nicht mehr enthalten.

Verwendung in der Funktionsliste

LeseKennwort (Variable)

Verwendung in benutzerdefinierten Funktionen

GetPassword Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in die das Kennwort geschrieben wird.

VerfolgeBenutzeränderung

Beschreibung

Gibt eine Systemmeldung aus, die anzeigt, welcher Benutzer gerade am Bediengerät angemeldet ist.

Diese Systemfunktion kann nur im Aufgabenplaner verwendet werden.

Verwendung in der Funktionsliste

VerfolgeBenutzeränderung

Verwendung in benutzerdefinierten Funktionen

-

Parameter

--

ZeigeAnmeldedialog

Beschreibung

Öffnet einen Dialog am Bediengerät, mit dem sich der Benutzer am Bediengerät anmelden kann.

Verwendung in der Funktionsliste

ZeigeAnmeldedialog

Verwendung in benutzerdefinierten Funktionen

-

Parameter

--

Berechnung

ErhöheVariable

Beschreibung

Addiert den angegebenen Wert zum Wert der Variablen.

X = X + a

Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Werts verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen können Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zuweisen.

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Wert der Variablen in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

Verwendung in der Funktionsliste

ErhöheVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

IncreaseTag Tag, Value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, zu welcher der angegebene Wert addiert wird.

Wert

Der Wert, der addiert wird.

Siehe auch

SetzeVariable (Seite 1621)

InvertiereLineareSkalierung

Beschreibung

Weist der Variablen X einen Wert zu, der aus dem Wert der angegebenen Variablen Y über die lineare Funktion X = (Y - b) / a berechnet wird.

Die Variablen X und Y dürfen nicht identisch sein. Diese Systemfunktion ist die Umkehrfunktion zur Systemfunktion "LineareSkalierung".

Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

Verwendung in der Funktionsliste

InvertiereLineareSkalierung (X, Y, b, a)

Verwendung in benutzerdefinierten Funktionen

InverseLinearScaling X, Y, b, a

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

X

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

Y

Die Variable, deren Wert zur Berechnung verwendet wird.

b

Der Wert, der subtrahiert wird.

a

Der Wert, durch den dividiert wird.

Beispiel

Der folgende Programmcode weist der Variable varX einen Wert mit der Funktion InverseLinearScaling zu.

```
{
BYTE varX;
BYTE Yvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

//Inverse linear scaling
 InverseLinearScaling (varX, Yvalue, bvalue, avalue);

printf ("varX = %d\r\n, varX);
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

LineareSkalierung

Beschreibung

Weist der Variablen Y einen Wert zu, der aus dem Wert der angegebenen Variablen X über die lineare Funktion Y = (a * X) + b berechnet wird.

Die Umkehrfunktion ist die Systemfunktion "InvertiereLineareSkalierung".

Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

Verwendung in der Funktionsliste

LineareSkalierung (Y, a, X, b)

Verwendung in benutzerdefinierten Funktionen

LinearScaling Y, a, X, b

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Y

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

a

Der Wert, mit dem multipliziert wird.

X

Die Variable, deren Wert zur Berechnung verwendet wird.

b

Der Wert, der addiert wird.

Beispiel

Der folgende Programmcode weist mit der Funktion LinearScaling der Variable Yvar einen Wert zu.

```
{
BYTE Yvar;
BYTE Xvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

// linear scaling
 LinearScaling ( Yvar, avalue, Xvalue, bvalue);

printf ("Yvar = %d\r\n, Yvar);
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

SetzeVariable

Beschreibung

Weist der angegebenen Variablen einen neuen Wert zu.

Hinweis

Abhängig vom Variablentyp können Sie mit dieser Systemfunktion Zeichenketten und Zahlen zuweisen.

Verwendung in der Funktionsliste

SetzeVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

SetTag Tag, Value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, welcher der angegebene Wert zugewiesen wird.

Wert

Der Wert, welcher der angegebenen Variablen zugewiesen wird.

Hinweis

Die Systemfunktion "SetzeVariable" wird nur nach Aufbau einer Verbindung ausgeführt.

Beispiel

Der folgende Programmcode setzt mit der Funktion SetTag den Wert der Variablen gs_tag_bit auf TRUE und speichert den Rückgabewert in der Variable ok.

```
{
BOOL ok;
BOOL bvalue;

//Set the tag to true
ok = SetTag("gs_tag_bit", TRUE);
//error handling
if(ok)
{
  // succeeded
  printf ( "Function has run through.\r\n" );
bvalue = GetTagBit("gs_tag_bit");
  printf ("Value of gs_tag_bit: %d\r\n", bvalue);
}
else
{
  // failed
  printf ( "Error - function failed." );
}
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

VerringereVariable

Beschreibung

Subtrahiert den angegebenen Wert vom Wert der Variablen.

X = X - a

Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Wertes verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen weisen Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zu.

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Wert der Variablen in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

Verwendung in der Funktionsliste

VerringereVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

DecreaseTag Tag, Value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, von welcher der angegebene Wert subtrahiert wird.

Wert

Der Wert, der subtrahiert wird.

Bitbearbeitung

InvertiereBit

Beschreibung

Invertiert den Wert der angegebenen Variable vom Typ "Bool":

- Hat die Variable den Wert 1 (TRUE), wird sie auf 0 (FALSE) gesetzt.
- Hat die Variable den Wert 0 (FALSE), wird sie auf 1 (TRUE) gesetzt.

Verwendung in der Funktionsliste

InvertiereBit (Variable)

Verwendung in benutzerdefinierten Funktionen

InvertBit Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, dessen Bit gesetzt wird.

Beispiel

Der folgende Programmcode invertiert den Wert der boolschen Variable bStatus und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Get current value
bValue=myTag.Value

'Save current value
bSaved=bValue

'Invert Bit
InvertBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 0;
BOOL bSaved = bStatus;

//Invert variable
invertBit(bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
  ...
}
```

InvertiereBitInVariable

Beschreibung

Invertiert ein Bit in der angegebenen Variablen:

- Hat das Bit in der Variablen den Wert 1 (TRUE), wird es auf 0 (FALSE) gesetzt.
- Hat das Bit in der Variablen den Wert 0 (FALSE), wird es auf 1 (TRUE) gesetzt.

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der

Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "InvertiereBit".

Verwendung in der Funktionsliste

InvertiereBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

InvertBitInTag Tag, Bit

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der das angegebene Bit gesetzt wird.

Bit

Die Nummer des Bits, das gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in einer Variablen von rechts nach links gezählt. Die Zählung beginnt mit 0.

Beispiel

Der folgende Programmcode invertiert ein Bit an der angegebenen Position bitposition in der Variable bStatusWord und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Get current value
bValue=myTag.Value

'Save current value
bSaved=bValue

'Invert Bit in position
bitposition=2
InvertBit(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bsaved = bStatusWord;
BYTE bitposition = 2;

//Invert bit in bitposition
InvertBitInTag (bStatusWord, bitposition);
//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bsaved);
  ...
}
```

RücksetzeBit

Beschreibung

Setzt den Wert einer Variablen vom Typ "Bool" auf 0 (FALSE).

Verwendung in der Funktionsliste

RücksetzeBit (Variable)

Verwendung in benutzerdefinierten Funktionen

ResetBit Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable vom Typ BOOL, die auf 0 (FALSE) gesetzt wird.

Beispiel

Der folgende Programmcode setzt den Wert der boolschen Variable bStatus mit der Funktion ResetBit auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Set value
bValue=1
myTag.Value=bValue

'Save current value
bSaved=bValue

'Reset Bit
ResetBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 1;
BOOL bSaved = bStatus;

//Reset bit
  ResetBit (bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
...
}
```

Siehe auch

RücksetzeBitInVariable (Seite 1629)

RücksetzeBitInVariable

Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 0 (FALSE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie stattdessen die Systemfunktion "RücksetzeBit".

Verwendung in der Funktionsliste

RücksetzeBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

ResetBitInTag Tag, Bit

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der ein Bit auf 0 (FALSE) gesetzt wird.

Bit

Die Nummer des Bits, das auf 0 (FALSE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position bitposition der Variablen bStatusWord auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Save current value
bValue=myTag.Value
bSaved=bValue

'Reset Bit
bitposition=2
ResetBit(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bSaved = bStatusWord;
BYTE bitposition = 2;

//Reset bit in bitposition
ResetBitInTag (bStatusWord, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bSaved);
  ...
}
```

Siehe auch

RücksetzeBit (Seite 1627)

SchiebenUndMaskieren

Beschreibung

Die Systemfunktion wandelt das Eingangs-Bitmuster der Quellvariablen in ein Ausgangs-Bitmuster der Zielvariablen. Dabei werden Bits verschoben und maskiert.

Hinweis

Wenn Quellvariable und Zielvariable eine unterschiedliche Anzahl von Bits haben, kann die Anwendung der Systemfunktion in der Zielvariablen zu einer Wertebereichsüberschreitung führen.

Verwendung in der Funktionsliste

SchiebenUndMaskieren (Quellvariable, Zielvariable, Zu schiebende Bits, Zu maskierende Bits)

Verwendung in benutzerdefinierten Funktionen

ShiftAndMask Source_tag, Target_tag, Bits_to_shift, Bits_to_mask

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Quellvariable

Die Variable enthält das Eingangs-Bitmuster, erlaubt sind Variablen von ganzzahligem Typ z. B. "Byte", "Char", "Int", "UInt", "Long" und "ULong".

Beispiel: Die Quellvariable vom Typ 16 Bit Integer hat den aktuellen Wert 72: 0000000001001000.

Zielvariable

In die Variable wird das Ausgangs-Bitmuster gespeichert. Erlaubt sind Variablen von ganzzahligem Typ z. B. "Byte", "Char", "Int", "UInt", "Long" und "ULong".

Beispiel: Das verschobene Eingangs-Bitmuster wird mit der Bit-Maske multipliziert, Bit für Bit UND-verknüpft: 0000000000001001. Das Ergebnis hat den Dezimalwert "8" und wird in der Zielvariablen gespeichert.

Achten Sie auf Folgendes:

- Quell- und Zielvariable haben die gleiche Anzahl Bits.
- Die Anzahl der zu schiebenden Bits ist kleiner als die Anzahl der Bits der Quell- und Zielvariablen.
- Zu maskierende Bits hat nicht mehr Bits als Quell- und Zielvariable.

Zu schiebende Bits

Anzahl der Bits, um die das Eingangs-Bitmuster nach rechts verschoben wird. Ein negativer Wert verschiebt das Eingangs-Bitmuster nach links.

Beispiel: "Zu schiebende Bits" hat den Wert "+3". Dann wird beim Aufruf der Systemfunktion das Eingangs-Bitmuster um 3 Bits nach rechts verschoben: 0000000000001001.
Links wird mit "0" aufgefüllt. Rechts werden 3 Bits abgeschnitten. Neuer Dezimalwert ist "9".

Hinweis

Wenn die Quellvariable einen vorzeichenbehafteten Datentyp Integer besitzt mit dem Vorzeichen "-", ist das linke Bit "1". Beim Verschieben nach rechts wird dieses Vorzeichen-Bit mit "0" aufgefüllt. Das Vorzeichen wechselt nach "+".

Zu maskierende Bits

Eine Integer-Zahl dient als Bit-Maske. Mit deren Bitmuster wird das verschobene Eingangs-Bitmuster multipliziert. Beispiel: Zahl "2478" mit dem Bitmuster "0000100110101110".

Die Bit-Maske geben Sie auf 3 verschiedene Arten ein:

- Hexadezimal: Geben Sie zuerst als Präfix "0h" oder "0H" ein, gefolgt von einem optionalen Leerzeichen zur besseren Lesbarkeit. Anschließend gruppieren Sie das Bitmuster in Viererblöcken (0000)(1001)(1010)(1110) und stellen jeden Block im 16er System dar: (0)(9)(A)(E). Bei der Eingabe sind nur die Zeichen 0-9, A-F, a-f erlaubt: "0h 09AE".
- Binär: Geben Sie zuerst als Präfix "0b" oder "0B" ein, gefolgt von einem optionalen Leerzeichen zur besseren Lesbarkeit. Anschließend gruppieren Sie das binäre Bitmuster zur Kontrolle in Viererblöcken 0000 1001 1010 1110 mit Leerzeichen dazwischen. Bei der Eingabe sind nur die Zeichen "0" oder "1" erlaubt: "0b 0000 1001 1010 1110".
- Dezimal: Geben Sie direkt ohne Präfix den Wert "2478" ein.

Hinweis

Wenn Sie die Geräteversion des Ziel-Bediengeräts nach der Projektierung ändern (z. B. "13.1.0" auf "13.0.0" oder umgekehrt), müssen Sie die Parameter dieser Systemfunktion kontrollieren und testen.

Die Datentypen "Char" und "Word" können Sie für die Parameter "Quellvariable" und "Zielvariable" ab der Geräteversion 13.1.0 verwenden. In den Geräteversionen vor 13.1.0 müssen diesen Parametern andere Datentypen zugewiesen werden:

- Statt "Char" verwenden Sie "SInt"
- Statt "Word" verwenden Sie "Int"

Anderenfalls können unerwünschte Effekte auftreten, wie z. B. falsches oder unerwartetes Verhalten der projektierten Systemfunktionen.

SetzeBit

Beschreibung

Setzt den Wert einer Variablen vom Typ "Bool" auf 1 (TRUE).

Verwendung in der Funktionsliste

SetzeBit (Variable)

Verwendung in benutzerdefinierten Funktionen

SetBit Tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable vom Typ BOOL, die auf 1 (TRUE) gesetzt wird.

Beispiel

Der folgende Programmcode setzt den Wert der boolschen Variable bStatus mit der Funktion SetBit auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Set value
bValue=0
myTag.Value=bValue

'Save current value
bSaved=bValue

'Set Bit
SetBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 0;
BOOL bSaved = bStatus;

//Set bit
  SetBit (bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
...
}
```

SetzeBitInVariable

Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 1 (TRUE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der

Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "SetzeBit".

Verwendung in der Funktionsliste

SetzeBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

SetBitInTag Tag, Bit

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der ein Bit auf 1 (TRUE) gesetzt wird.

Bit

Die Nummer des Bits, das auf 1 (TRUE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

Hinweis

Eine notwendige Bedingung für eine zuverlässige Funktionalität ist eine garantierte Aktualisierung der verwendeten Variablen mit den aktuellen Prozesswerten. Projektieren Sie die Variable deswegen in einem E/A-Feld oder die Systemfunktion an einem Bildobjekt z. B. einer Schaltfläche.

Wenn Sie die Systemfunktion an ein kurzzeitiges Ereignis z. B. Kommen einer Meldung projektiert haben, dann erreichen Sie aktuelle Prozesswerte nur, wenn Sie die Variable auf dauerhaft lesen setzen.

Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position bitposition der Variablen bStatusWord auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Save current value
bValue=myTag.Value
bSaved=bValue

'Set Bit in tag
bitposition=2
SetBitInTag(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bSaved = bStatusWord;
BYTE bitposition = 2;

//Reset bit in bitposition
SetBitInTag (bStatusWord, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bSaved);
  ...
}
```

Siehe auch

SetzeBit (Seite 1633)

SetzeBitWährendTasteGedrückt

Beschreibung

Setzt ein Bit der angegebenen Variablen auf 1 (TRUE), solange der Benutzer die projektierte Taste gedrückt hält.

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde. Um Probleme mit sich zeitlich überschneidenden Zugriffen auf dieselbe Variable zu vermeiden, sollten Sie mit dieser Systemfunktion ausschließlich auf Variablen vom Typ BOOL zugreifen.

Hinweis

Durch einen für eine Taste projektierten Bildwechsel werden alle Funktionen am Ereignis "Loslassen" sofort ausgeführt, auch wenn die Taste noch gedrückt ist.

Wenn bei einer Funktionstaste die Systemfunktion "SetzeBitWährendTasteGedrückt" projektiert ist, wird sofort nach dem Ausführen des Bildwechsels das Bit zurückgesetzt. Dieses Verhalten ist notwendig, da sich die Tastenbelegung nach dem Bildwechsel ändert.

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "SetzeBit".

Verwendung in der Funktionsliste

SetzeBitWährendTasteGedrückt (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Variable

Die Variable, in der ein Bit zeitweilig auf 1 (TRUE) gesetzt wird. Verwenden Sie, soweit dies die Steuerung zulässt, ausschließlich Variablen vom Typ BOOL.

Bit

Die Nummer des Bits, das zeitweilig auf 1 (TRUE) gesetzt wird.

Hinweis

Eine notwendige Bedingung für eine zuverlässige Funktionalität ist eine garantierte Aktualisierung der verwendeten Variablen mit den aktuellen Prozesswerten. Projektieren Sie die Variable deswegen in einem IO-Feld oder die Funktion an einem Bildelement z. B. einer Schaltfläche.

Wenn Sie die Funktion an ein kurzzeitiges Ereignis z. B. Kommen einer Meldung projektiert haben, dann erreichen Sie aktuelle Prozesswerte nur, wenn Sie die Variable auf dauerhaft lesen setzen.

Siehe auch

SetzeBit (Seite 1633)

Drucken

DruckeBild

Beschreibung

Druckt das gerade am Bediengerät angezeigte Bild auf dem Drucker aus, der am Bediengerät angeschlossen ist.

Geöffnete Fenster werden ebenfalls mitgedruckt.

Hinweis

Die Druckereinstellungen werden aus den aktuellen Einstellungen im Windows-Betriebssystem übernommen.

Verwendung in der Funktionsliste

DruckeBild

Verwendung in benutzerdefinierten Funktionen

PrintScreen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

DruckeProtokoll

Beschreibung

Druckt das angegebene Protokoll auf dem Drucker aus, der am Bediengerät angeschlossen ist. Das Protokoll wird in der Sprache gedruckt, die Sie am Bediengerät eingestellt haben.

Hinweis

Wenn Runtime beendet wird, während ein mit Hilfe der Systemfunktion Archivdaten gedruckt werden, so wird das Protokoll ab dem Zeitpunkt zu dem Runtime beendet ist, nicht mehr mit Daten versorgt.

Verwendung in der Funktionsliste

DruckeProtokoll (Protokoll)

Verwendung in benutzerdefinierten Funktionen

PrintReport Report

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Protokoll

Name des Protokolls, das ausgedruckt wird.

Hinweis

Wenn Sie für die Funktion "DruckeProtokoll" ein neues Protokoll über den Dialog "Funktionsliste" anlegen, erscheint beim Kompilieren folgende Warnung: "Das Protokoll "Protokoll_1" hat keine Druckseiten."

Um die Warnung zu beheben, öffnen Sie "Protokoll_1" über das Projektfenster und kompilieren Sie das Projekt neu.

SetzeProtokolliermodusMeldung

Beschreibung

Schaltet das automatische Protokollieren von Meldungen auf dem Drucker ein oder aus.

Verwendung in der Funktionsliste

SetzeProtokolliermodusMeldung (Modus)

Verwendung in benutzerdefinierten Funktionen

SetAlarmReportMode Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob Meldungen automatisch auf dem Drucker protokolliert werden:

0 (hmiDisablePrinting) = Protokollierung Aus: Meldungen werden nicht automatisch gedruckt.

1 (hmiEnablePrinting) = Protokollierung Ein: Meldungen werden automatisch gedruckt.

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her.

Einstellungen

LesePLCModus

Beschreibung

Wertet den aktuellen Zustand der verbundenen Steuerung aus.

Die Systemfunktion "LesePLCModus" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7-1200
- SIMATIC S7-1500

Verwendung in der Funktionsliste

LesePLCModus (Verbindung, Modus)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Verbindung

Verbindung von Steuerung und Bediengerät.

Modus

Wertet den Zustand der verbundenen Steuerung aus.
8: RUN = Das Steuerungsprogamm wird abgearbeitet.
4: STOP = Das Steuerungsprogramm ist unterbrochen.
Wählen Sie für die Auswertung eine geeignete Anwendervariable aus.

SetzeAnzeigemodus

Beschreibung

Ändert die Einstellungen des Fensters, in dem die Runtime-Software läuft.

Standardmäßig läuft die Runtime-Software im Vollbildmodus. Die Taskumschaltung von Windows ist deaktiviert.

Verwendung in der Funktionsliste

SetzeAnzeigemodus (Darstellung)

Verwendung in benutzerdefinierten Funktionen

SetDisplayMode Display_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Darstellung

Legt die Einstellungen für das Fenster fest, in dem die Runtime-Software läuft:

1 (hmiScreenFull): Vollbild: Titelzeile des Fensters ist nicht sichtbar

2 (hmiScreenMaximized): Maximiert

3 (hmiScreenRestore): Wiederherstellen: Zuletzt verwendete Fenstereinstellung wird verwendet. Diese Darstellung kann nur verwendet werden, wenn das Fenster minimiert oder maximiert angezeigt wird.

4 (hmiScreenMinimized): Minimiert

5 (hmiScreenAutoAdjust): Automatisch: Die Größe des Fensters wird so eingestellt, dass alle enthaltenen Bildobjekte sichtbar sind

6 (hmiScreenOnTop): Vordergrund; abhängig von der Windows-Einstellung wird entweder das Fenster im Vordergrund eingeblendet oder das zum Fenster gehörige Programmsymbol blinkt in der Taskleiste. Die Einstellung kann in der Windows-Konfiguration geändert werden und gilt für alle Windows-Anwendungen.

SetzeBetriebsartGerät

Beschreibung

Schaltet die Betriebsart am Bediengerät um. Folgende Betriebsarten sind möglich: "Online", "Offline" und "Laden".

Verwendung in der Funktionsliste

SetzeBetriebsartGerät (Betriebsart)

Verwendung in benutzerdefinierten Funktionen

SetDeviceMode Operating_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Betriebsart

Legt die Betriebsart des Bediengeräts fest. Bei Geräten mit Windows CE kann dieser Parameter nicht über eine Variable vom Datentyp BOOL versorgt werden.

0 (hmiOnline) = Online: Die Verbindung zur Steuerung wird hergestellt. Dabei wird immer der projektierte Verbindungsstatus eingestellt. Die Stati, die zuletzt in Runtime verwendet wurden, werden nicht berücksichtigt.

1 (hmiOffline) = Offline: Die Verbindung zur Steuerung wird getrennt.

2 (hmiTransfer) = Laden: Sie können ein Projekt vom Projektierungsrechner auf das Bediengerät übertragen.

Hinweis

Wenn Sie als Bediengerät einen PC verwenden, wird beim Umschalten der Betriebsart nach "Laden" die Runtime-Software beendet.

Siehe auch

SetzeVerbindungsmodus (Seite 1646)

SetzeModusBildschirmtastatur

Beschreibung

Aktiviert oder deaktiviert das automatische Einblenden der Bildschirmtastatur am Bediengerät.

Verwenden Sie diese Systemfunktion, wenn Sie das Einblenden der Bildschirmtastatur verhindern wollen, weil Sie z.B. eine externe Tastatur am Bediengerät angeschlossen haben.

Hinweis

Wenn Sie die Systemfunktion "SetzeModusBildschirmtastatur" ("SetScreenKeyboardMode") auf einem nicht Touch-Bediengerät einsetzen, aktivieren Sie in den Geräteeinstellungen im Bereich "Einstellungen für Runtime" die Option "Bildschirmtastatur verwenden".

Verwendung in der Funktionsliste

SetzeModusBildschirmtastatur (Modus)

Verwendung in benutzerdefinierten Funktionen

SetScreenKeyboardMode Mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob die Bildschirmtastatur eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Bildschirmtastatur wird ausgeblendet

1 (hmiOn) = Ein: Bildschirmtastatur wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her.

SetzePLCDatumUhrzeit

Beschreibung

Ändert das Datum und die Uhrzeit der verbundenen Steuerung.

Die Systemfunktion "SetzePLCDatumUhrzeit" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7-1200
- SIMATIC S7-1500

Verwendung in der Funktionsliste

SetzePLCDatumUhrzeit (Verbindung, Zeit)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Verbindung

Verbindung von Steuerung und Bediengerät.

Zeit

Übergibt das Datum und die Uhrzeit des Bediengeräts an die Steuerung. Die Steuerung übernimmt das Datum und die Uhrzeit des Bediengeräts.

SetzePLCModus

Beschreibung

Schaltet die Betriebsart der Steuerung in einen der folgenden Zustände um:

- RUN
- STOP

Die Systemfunktion "SetzePLCModus" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7-1200
- SIMATIC S7-1500

Verwendung in der Funktionsliste

SetzePLCModus (Verbindung, Modus)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Verbindung

Verbindung von Steuerung und Bediengerät.

Modus

Legt die Betriebsart der Steuerung fest:

8: RUN = Die Steuerung wir in den Zustand RUN versetzt. Das Steuerungsprogamm wird abgearbeitet.

4: STOP = DieSteuerung wird in den Zustand STOP versetzt. Das Steuerungsprogramm wird unterbrochen.

Siehe auch

SystemDiagnoseanzeigeAktualisierePLCBuffer (Seite 1683)

SetzeSprache

Beschreibung

Schaltet die Sprache am Bediengerät um. Alle projektierten Texte und Systemmeldungen werden am Bediengerät in der neu eingestellten Sprache dargestellt.

Verwendung in der Funktionsliste

SetzeSprache (Sprache)

Verwendung in benutzerdefinierten Funktionen

SetLanguage Language

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Sprache

Legt fest, welche Sprache am Bediengerät eingestellt wird. Es sind folgende Angaben möglich

- -1 (hmiToggle) = Umschalten: Wechselt zur nächsten Sprache. Die Reihenfolge legen Sie während der Projektierung im Editor "Projektsprachen" fest.
- Nummer, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben. Wechselt zu der Sprache mit der angegebenen Nummer.
- Sprache, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben.
- Sprachkürzel gemäß der VBScript5-Referenz: Wechselt zur Sprache, die dem angegebenen Sprachkürzel entspricht, z. B. "de-DE" für Deutsch (Deutschland) oder "en-US" für Englisch (Vereinigte Staaten von Amerika).
  Eine Übersicht aller Sprachkürzel finden Sie in den Grundlagen von VBScript unter dem Thema "Gebietsschema-ID (LCID)-Diagramm".

SetzeVerbindungsmodus

Beschreibung

Baut die angegebene Verbindung auf oder ab.

Hinweis

Eine Verbindung zur Steuerung kann nur dann hergestellt werden, wenn Sie am Bediengerät die Betriebsart "Online" eingestellt haben. Verwenden Sie dazu die Systemfunktion "SetzeBetriebsartGerät".

Wenn die Verbindung im Modus „Offline“ geladen wurde, wird die Verbindung bei jedem „Offline“-Schalten der Betriebsart wieder getrennt. Um nach dem erneuten „Online“-Schalten der Betriebsart die Verbindung wieder aufzubauen, setzen Sie die Verbindung erneut in den Modus „Online“.

Verwendung in der Funktionsliste

SetzeVerbindungsmodus (Modus, Verbindung)

Verwendung in benutzerdefinierten Funktionen

SetConnectionMode Mode, Connection

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob die Verbindung mit der Steuerung hergestellt wird oder getrennt wird. Bei Geräten mit Windows CE kann dieser Parameter nicht über eine Variable vom Datentyp BOOL versorgt werden.

0 (hmiOnline) = Online: Verbindung wird hergestellt.

1 (hmiOffline) = Offline: Verbindung wird getrennt.

Verbindung

Die Steuerung, die mit dem Bediengerät verbunden wird. Den Namen der Steuerung legen Sie im Verbindungs-Editor fest.

Mehrfache Verwendung der Systemfunktion in einer benutzerdefinierten Funktion

Wenn Sie die Systemfunktion "SetzeVerbindungsmodus" für verschiedene Verbindungen verwenden, kann es vorkommen, dass nicht alle Systemfunktionen korrekt ausgeführt werden. Um dies zu vermeiden, gehen Sie wie folgt vor:

1. Legen Sie eine Variable mit dem Startwert "0" an.
2. Projektieren Sie an das Ereignis "Wertänderung" der HMI-Variablen die Systemfunktion "SetzeVerbindungsmodus". Wenn Sie z.B. 3 Verbindungen abbauen wollen, dann müssen die Systemfunktion dreimal projektieren.
3. Wenden Sie in der benutzerdefinierten Funktion die Systemfunktion "InvertBit" auf die HMI-Variable an.

Anwendungsbeispiel

Typische Anwendungsbeispiele für diese Systemfunktion sind die beiden folgenden:

- Test
  Solange keine Steuerung am Bediengerät angeschlossen ist, werden während des Tests am Bediengerät keine Fehlermeldungen ausgegeben. Wenn das Bediengerät an eine Steuerung angeschlossen ist, können Sie die Verbindung zur Steuerung auf Tastendruck herstellen.
- Inbetriebnahme
  Für eine Anlage sollen mehrere Steuerungen projektiert werden. Dazu projektieren Sie zunächst alle Steuerungen bis auf eine "offline". Nach der Inbetriebnahme der ersten Steuerung stellen Sie die Verbindung zu jeder weiteren Steuerung über Tastendruck her. Auf diese Weise nehmen Sie die anderen Steuerungen nacheinander in Betrieb.

SetzeWebZugriff

Beschreibung

Legt die Zugriffsart auf die Runtime-Applikation über das Internet fest.

Verwendung in der Funktionsliste

SetzeWebZugriff (Zugriffsart)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Zugriffsart

Legt die Zugriffsart auf die Runtime-Applikation fest:

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her.

0 (hmiReadOnly) = Nur Lesen.

1 (hmiReadWrite) = Lesen und Schreiben.

WechseleVerbindung

Beschreibung

Trennt die Verbindung zur gerade verwendeten Steuerung und stellt eine Verbindung zu einer Steuerung mit einer anderen Adresse her. Die neu verbundene Steuerung muss dabei zur gleichen Geräteklasse (S7-300, S7-400, usw) gehören.

Hinweis

Achten Sie beim Wechsel zu einer anderen Adresse darauf, dass diese Adresse nicht bereits von einem anderen Bediengerät verwendet wird.

Folgende Adressentypen werden unterstützt:

- IP-Adresse
- MPI-Adresse

Folgende Steuerungstypen werden unterstützt:

- SIMATIC S7 300/400
- SIMATIC S7 200
- SIMATIC S7 1200 (Geräteversion bis V11)
- SIMATIC S7 LOGO!
- SIMATIC S7 NC
- SIMOTION

Verwendung in der Funktionsliste

WechseleVerbindung (Verbindung, Adresse, Steckplatz, Baugruppenträger)

Verwendung in benutzerdefinierten Funktionen

ChangeConnection Connection, Address, Slot, Rack

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Verbindung

Name der Verbindung, die getrennt wird. Den Namen legen Sie während der Projektierung z.B. im Editor "Verbindungen" fest.

Adresse

MPI/PROFIBUS- oder IP-Adresse der Steuerung, zu der die Verbindung hergestellt wird.

Hinweis

Sie legen die Adresse über eine Variable fest. Die Objektliste zeigt Ihnen Variablen aller Datentypen an. Wählen Sie ausschließlich Variablen folgender Datentypen:

- Ethernet-Verbindung: Datentyp "String"
- MPI-Verbindung: Datentypen "Int"

Steckplatz

Steckplatz der Steuerung, zu der die Verbindung hergestellt wird.

Baugruppenträger

Baugruppenträger der Steuerung, zu der die Verbindung hergestellt wird.

Anwendungsbeispiel

Sie wollen ein Bediengerät an verschiedenen Maschinen betreiben. Dazu projektieren Sie im Projekt eine Steuerung. Beim Wechsel der Steuerung wird die Verbindung mit der gerade verwendeten Steuerung getrennt. Danach wird die Verbindung zur neuen Steuerung mit anderen Adressparametern wieder aufgebaut. Alle Variablen dieser Verbindung werden nun aus der neuen Steuerung aktualisiert.

Die Steuerung, die Sie beim Anlegen des Projekts angegeben haben, wird standardmäßig verwendet.

1. Geben Sie im Editor "Verbindungen" Namen und Adresse der Steuerung ein.
2. Projektieren Sie im Prozessbild eine Schaltfläche.
3. Projektieren Sie am Ereignis "Drücken" die Systemfunktion "WechseleVerbindung".
4. Geben Sie als Parameter den Namen der Verbindung und die Adresse der Steuerung an.

WechseleVerbindungEIP

Beschreibung

Trennt die Verbindung zur gerade verwendeten Allen Bradley-Steuerungen und stellt eine Verbindung zu einer anderen Allen Bradley-Steuerung in Runtime her. "EIP" steht für "Ethernet/IP".

Die neu verbundene Steuerung muss dabei zur gleichen Geräteklasse (Allen Bradley) gehören.

Damit die Verbindung zustande kommt, muss der CPU-Typ der zu verbindenden Steuerung dem CPU-Typ der bereits verbundenen Steuerung entsprechen, z. B. beide Steuerungen haben den CPU-Typ "SLC/Micrologix PLC".

Hinweis

Achten Sie beim Wechsel zu einer anderen Adresse darauf, dass diese Adresse nicht bereits von einem anderen Bediengerät verwendet wird.

Folgende Adressentypen werden unterstützt:

- IP-Adresse

Verwendung in der Funktionsliste

WechseleVerbindungEIP (Verbindung, Adresse, Kommunikationspfad)

Verwendung in benutzerdefinierten Funktionen

ChangeConnectionEIP Connection, Address, Communication_path

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Verbindung

Name der Verbindung, die getrennt wird. Den Namen legen Sie während der Projektierung z. B. im Editor "Verbindungen" fest.

Adresse

IP-Adresse der Steuerung, zu der die Verbindung hergestellt wird.

Hinweis

Sie legen die Adresse über eine Variable fest. Die Objektliste zeigt Ihnen Variablen aller Datentypen an. Wählen Sie ausschließlich Variablen des Datentyps "String".

Hinweis

Die hinterlegte IP-Adresse muss aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255 bestehen.

Kommunikationspfad

CIP-Pfad vom Ethernet-Modul zur Steuerung, zu der die Verbindung hergestellt wird.

Sie legen den Kommunikationspfad entweder über eine String-Variable oder eine HMI-Variable vom Datentyp "String".

Anwendungsbeispiel

Sie wollen ein Bediengerät an verschiedenen Maschinen betreiben. Dazu projektieren Sie im Projekt eine Steuerung. Beim Wechsel der Steuerung wird die Verbindung mit der gerade verwendeten Steuerung getrennt. Danach wird die Verbindung zur neuen Steuerung mit anderen Adressparametern wieder aufgebaut. Alle Variablen dieser Verbindung werden nun aus der neuen Steuerung aktualisiert.

1. Geben Sie im Editor "Verbindungen" Namen und Adresse der Steuerung ein.
2. Projektieren Sie im Prozessbild eine Schaltfläche.
3. Projektieren Sie am Ereignis "Drücken" die Systemfunktion "WechseleVerbindungEIP".
4. Geben Sie als Parameter den Namen der Verbindung und die Adresse der Steuerung an.

Funktionen des Displays

ÄndereKontrast

Beschreibung

Ändert den Kontrast des Displays am Bediengerät um eine Stufe.

Verwendung in der Funktionsliste

ÄndereKontrast (Änderung)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Änderung

Legt fest, wie der Kontrast geändert wird:

0 (hmiDecrease) = Verringern: Verringert den Kontrast um eine Stufe.

1 (hmiIncrease) = Erhöhen: Erhöht den Kontrast um eine Stufe.

Anwendungsbeispiel

Ziel

Sie wollen über jeweils eine Schaltfläche den Kontrast des Bildschirms erhöhen und verringern.

Hinweise zur Projektierung

Projektieren Sie zwei Schaltflächen und projektieren Sie jeweils an das Ereignis "Drücken" die Systemfunktion "ÄndereKontrast". Vergeben Sie als Parameter einmal "Verringern" und einmal "Erhöhen".

Ablauf am Bediengerät

Wenn Sie in Runtime eine der beiden Schaltflächen drücken, wird der Kontrast um eine Stufe erhöht oder verringert.

Siehe auch

Abmelden (Seite 1612)

KalibriereTouchscreen

Beschreibung

Ruft ein Programm zur Kalibrierung des Touch-Displays auf.

Während des Kalibriervorgangs werden Sie aufgefordert, fünf Positionen auf dem Touch-Display zu berühren. Um den Kalibriervorgang zu bestätigen, berühren Sie innerhalb von 30 Sekunden das Touch-Display. Wenn Sie das Touch-Display innerhalb der Zeitspanne nicht berühren, werden die Kalibriereinstellungen verworfen. Die Bedienerführung ist in englischer Sprache.

Verwenden Sie diese Systemfunktion, wenn Sie das Bediengerät zum ersten Mal in Betrieb nehmen.

Verwendung in der Funktionsliste

KalibriereTouchscreen

Verwendung in benutzerdefinierten Funktionen

CalibrateTouchScreen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Hinweis

Die Systemfunktion KalibriereTouchscreen kann nicht simuliert werden.

SetzeAkustischesSignal

Beschreibung

Konfiguriert die akustische Rückmeldung des Bediengeräts bei der Bedienung des Touch-Displays.

Hinweis

Beim Neustart des Bediengeräts wird die Konfiguration wieder hergestellt, die beim Ausschalten eingestellt war.

Verwendung in der Funktionsliste

SetzeAkustischesSignal (Lautstärke)

Verwendung in benutzerdefinierten Funktionen

SetAcousticSignal Volume

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Lautstärke

Legt fest, ob und wie laut ein akustisches Signal ausgegeben wird:

-1 (hmiToggle) = Umschalten: Schaltet die Ausgabe des akustischen Signals, wie folgt um: Stumm > Leise > Laut.

0 (hmiMuted) = Stumm: kein akustisches Signal

1 (hmiQuiet) = Leise: leises akustisches Signal

2 (hmiLoud) = Laut: lautes akustisches Signal

SetzeHelligkeit

Beschreibung

Legt die Helligkeit des Displays fest.

Hinweis

Beim Neustart des Bediengeräts wird die Konfiguration wieder hergestellt, die im Control Panel/ Start Center eingestellt ist.

Gilt für Basic Panels 2nd Generation, Mobile Panels und Comfort Panels:

Der Wert für die Systemfunktion "SetzeHelligkeit" ist zwischen 0 % und 100 % einstellbar. Der eingestellte Wert wird auf das Bediengerät übertragen. Die Helligkeitseinstellungen auf dem Bediengerät sind unter "Start Center > Settings > Display" einsehbar und editierbar. Die Bediengeräte unterstützen eine Helligkeitseinstellung zwischen 10 % und 100 %.

Wenn der Systemfunktion „SetzeHelligkeit“ ein Wert von 0 % zugewiesen ist, ist das Display des Bediengeräts in Runtime standardmäßig ausgeschaltet. Wenn der Bediener das Display berührt, wechselt das Display zur vorherigen Helligkeitseinstellung.

Wenn der Systemfunktion „SetzeHelligkeit“ ein Wert zwischen 1 % und 10 % zugewiesen ist und der Bediener die Display-Einstellungen im Start Center öffnet, stellt sich die Helligkeit auf 10 % um.

Verwendung in der Funktionsliste

SetzeHelligkeit (Wert)

Verwendung in benutzerdefinierten Funktionen

SetBrightness Value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Wert

Neuer Wert für die Helligkeit.

SetzeFarbeHintergrundbeleuchtung

Beschreibung

Legt die Farbe für die Hintergrundbeleuchtung fest.

Hinweis

Beim Neustart des Bediengeräts wird die Konfiguration wieder hergestellt, die beim Ausschalten eingestellt war.

Verwendung in der Funktionsliste

SetzeFarbeHintergrundbeleuchtung (Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Wert

Legt die Farbe für die Hintergrundbeleuchtung fest:

0 (hmiWhite) = Weiß: Keine Farbe

1 (hmiGreen) = Grün: Grüne Farbe

2 (hmiYellow) = Gelb: Gelbe Farbe

3 (hmiRed) = Rot: Rote Farbe

LeseHelligkeit

Beschreibung

Liest den Wert für die Helligkeit aus.

Verwendung in der Funktionsliste

LeseHelligkeit (Helligkeit)

Verwendung in benutzerdefinierten Funktionen

GetBrightness Brightness

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Helligkeit

Variable, in die der Wert geschrieben wird.

Funktionen des HTML-Browsers

HTMLBrowserAbbrechen

Beschreibung

Führt die Funktion "Abbrechen" des HTML-Browsers aus.

Verwendung in der Funktionsliste

HTMLBrowserAbbrechen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserAbwärtsScrollen

Beschreibung

Blättert im HTML-Browser abwärts.

Verwendung in der Funktionsliste

HTMLBrowserAbwärtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserAktualisieren

Beschreibung

Führt die Funktion "Aktualisieren" des HTML-Browsers aus.

Verwendung in der Funktionsliste

HTMLBrowserAktualisieren (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserAufwärtsScrollen

Beschreibung

Blättert im HTML-Browser aufwärts.

Verwendung in der Funktionsliste

HTMLBrowserAufwärtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserHeranzoomen

Beschreibung

Vergrößert die Darstellung im HTML-Browser um eine Zoomstufe.

Verwendung in der Funktionsliste

HTMLBrowserHeranzoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserHerauszoomen

Beschreibung

Verkleinert die Darstellung im HTML-Browser um eine Zoomstufe.

Verwendung in der Funktionsliste

HTMLBrowserHerauszoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserNachLinksScrollen

Beschreibung

Blättert im HTML-Browser nach links.

Verwendung in der Funktionsliste

HTMLBrowserNachLinksScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserNachRechtsScrollen

Beschreibung

Blättert im HTML-Browser nach rechts.

Verwendung in der Funktionsliste

HTMLBrowserNachRechtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserSeiteNachOben

Beschreibung

Führt im HTML-Browser zu einer seitenweisen Cursorbewegung nach oben.

Verwendung in der Funktionsliste

HTMLBrowserSeiteNachOben (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserSeiteNachUnten

Beschreibung

Führt im HTML-Browser zu einer seitenweisen Cursorbewegung nach unten.

Verwendung in der Funktionsliste

HTMLBrowserSeiteNachUnten (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserStartseite

Beschreibung

Wechselt zur Startseite, die für den HTML-Browsers hinterlegt ist.

Verwendung in der Funktionsliste

HTMLBrowserStartseite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserVorwärts

Beschreibung

Führt die Funktion "Vorwärts" des HTML-Browsers aus.

Verwendung in der Funktionsliste

HTMLBrowserVorwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

HTMLBrowserZurück

Beschreibung

Führt die Funktion " Zurück" des HTML-Browsers aus.

Verwendung in der Funktionsliste

HTMLBrowserZurück (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Objektname des HTML-Browsers, in dem der Befehl ausgeführt wird.

Funktionen der Kurvenanzeige

KurvenanzeigeBlättereVorwärts

Beschreibung

Blättert in der Kurvenanzeige um eine Anzeigenbreite vorwärts.

Verwendung in der Funktionsliste

KurvenanzeigeBlättereVorwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in welcher um eine Anzeigenbreite vorwärts geblättert wird.

KurvenanzeigeBlättereZurück

Beschreibung

Blättert in der Kurvenanzeige um eine Anzeigenbreite rückwärts.

Verwendung in der Funktionsliste

KurvenanzeigeBlättereZurück (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der um eine Anzeigenbreite rückwärts geblättert wird.

KurvenanzeigeErweitern

Beschreibung

Verkleinert den Zeitabschnitt, der in der Kurvenanzeige dargestellt wird.

Verwendung in der Funktionsliste

KurvenanzeigeErweitern (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in welcher der dargestellte Zeitabschnitt verkleinert wird.

KurvenanzeigeKomprimieren

Beschreibung

Vergrößert den Zeitabschnitt, der in der Kurvenanzeige dargestellt wird.

Verwendung in der Funktionsliste

KurvenanzeigeKomprimieren (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in welcher der dargestellte Zeitabschnitt vergrößert wird.

KurvenanzeigeLinealLinks

Beschreibung

Bewegt in der Kurvenanzeige die Leselinie rückwärts (nach links).

Hinweis

Damit Sie die Leselinie bewegen können, müssen Sie die Leselinie eingeschaltet haben. Verwenden Sie dazu die Systemfunktion "KurvenanzeigeSetzeLinealmodus".

Verwendung in der Funktionsliste

KurvenanzeigeLinealLinks (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der die Leselinie rückwärts bewegt wird.

Siehe auch

KurvenanzeigeSetzeLinealmodus (Seite 1664)

KurvenanzeigeLinealRechts

Beschreibung

Bewegt in der Kurvenanzeige die Leselinie vorwärts (nach rechts).

Hinweis

Damit Sie die Leselinie bewegen können, müssen Sie die Leselinie eingeschaltet haben. Verwenden Sie dazu die Systemfunktion "KurvenanzeigeSetzeLinealmodus".

Verwendung in der Funktionsliste

KurvenanzeigeLinealRechts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der die Leselinie vorwärts bewegt wird.

Siehe auch

KurvenanzeigeSetzeLinealmodus (Seite 1664)

KurvenanzeigeSetzeLinealmodus

Beschreibung

Blendet in der Kurvenanzeige die Leselinie ein oder aus. Die Leselinie zeigt den zu einem X-Wert gehörigen Y-Wert an.

Hinweis

Damit das Lineal eingeblendet wird, müssen Sie in den Eigenschaften der Kurvenanzeige die Einstellung "Lineal anzeigen" aktivieren.

Verwendung in der Funktionsliste

KurvenanzeigeSetzeLinealmodus (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der die Leselinie eingeblendet oder ausgeblendet wird.

Siehe auch

KurvenanzeigeLinealRechts (Seite 1664)

KurvenanzeigeLinealLinks (Seite 1663)

KurvenanzeigeStartenStoppen

Beschreibung

Hält in der Kurvenanzeige die Kurvenaufzeichnung an oder setzt die Kurvenaufzeichnung fort.

Verwendung in der Funktionsliste

KurvenanzeigeStartenStoppen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der die Kurvenaufzeichnung gestartet oder gestoppt wird.

KurvenanzeigeZurückZumAnfang

Beschreibung

Blättert in der Kurvenanzeige zum Beginn des Darstellungsbereichs zurück.

Verwendung in der Funktionsliste

KurvenanzeigeZurückZumAnfang (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Kurvenanzeige, in der zum Beginn des Darstellungsbereichs geblättert wird.

Siehe auch

Systemfunktionen für Runtime Advanced (Seite 1562)

Funktionen der PDF-Anzeige

PDFAbwärtsScrollen

Beschreibung

Blättert in der PDF-Datei in der PDF-Anzeige abwärts.

Verwendung in der Funktionsliste

PDFAbwärtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFAufwärtsScrollen

Beschreibung

Blättert in der PDF-Datei in der PDF-Anzeige aufwärts.

Verwendung in der Funktionsliste

PDFAufwärtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFAnpassenAufBreite

Beschreibung

Passt die Darstellung der PDF-Datei an die Breite des Fensters der PDF-Anzeige an.

Verwendung in der Funktionsliste

PDFAnpassenAufBreite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFAnpassenAufHöhe

Beschreibung

Passt die Darstellung der PDF-Datei an die Höhe des Fensters der PDF-Anzeige an.

Verwendung in der Funktionsliste

PDFAnpassenAufHöhe (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFGeheZurErstenSeite

Beschreibung

Wechselt zur ersten Seite der PDF-Datei in der PDF-Anzeige.

Verwendung in der Funktionsliste

PDFGeheZurErstenSeite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFGeheZurLetztenSeite

Beschreibung

Wechselt zur letzten Seite der PDF-Datei in der PDF-Anzeige.

Verwendung in der Funktionsliste

PDFGeheZurLetztenSeite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFGeheZurNächstenSeite

Beschreibung

Wechselt zur nächsten Seite der PDF-Datei in der PDF-Anzeige.

Verwendung in der Funktionsliste

PDFGeheZurNächstenSeite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFGeheZuSeite

Beschreibung

Wechselt zur eingegebenen Seite der PDF-Datei in der PDF-Anzeige.

Verwendung in der Funktionsliste

PDFGeheZuSeite (Bildobjekt, Seite)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

Seite

Seite im PDF-Dokument, die dargestellt wird.

PDFGeheZurVorherigenSeite

Beschreibung

Wechselt zur vorherigen Seite der PDF-Datei in der PDF-Anzeige.

Verwendung in der Funktionsliste

PDFGeheZurVorherigenSeite (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFHeranzoomen

Beschreibung

Vergrößert die Darstellung in der PDF-Datei in der PDF-Anzeige um eine Zoomstufe.

Verwendung in der Funktionsliste

PDFHeranzoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFHerauszoomen

Beschreibung

Verkleinert die Darstellung in der PDF-Datei in der PDF-Anzeige um eine Zoomstufe.

Verwendung in der Funktionsliste

PDFHerauszoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFLinksScrollen

Beschreibung

Blättert in der PDF-Datei in der PDF-Anzeige nach links.

Verwendung in der Funktionsliste

PDFLinksScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFRechtsScrollen

Beschreibung

Blättert in der PDF-Datei in der PDF-Anzeige nach rechts.

Verwendung in der Funktionsliste

PDFRechtsScrollen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

PDFZoomOriginal

Beschreibung

Ändert die Darstellung in der PDF-Datei in der PDF-Anzeige auf die Originalgröße.

Verwendung in der Funktionsliste

PDFZoomOriginal (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PDF-Anzeige, in der der Befehl ausgeführt wird.

Funktionen der PLC-Code-Anzeige

AktivierePLCCodeAnzeige

Beschreibung

Führt einen Bildwechsel zu dem angegebenen Bild mit der PLC-Code-Anzeige aus. Die PLC-Code-Anzeige zeigt den Programmcode des betroffenen Netzwerks.

Die Systemfunktion AktivierePLCCodeAnzeige projektieren Sie an das Ereignis "Schaltfläche PLC-Code-Anzeige klicken" einer GRAPH-Übersicht oder an ein Ereignis einer Schaltfläche.

Bildwechsel von der GRAPH-Übersicht

Sie projektieren die Systemfunktion an das Ereignis "Schaltfläche PLC-Code-Anzeige klicken" in einer GRAPH-Übersicht um die GRAPH-Schrittkette in der PLC-Code-Anzeige darzustellen. Wenn kein Fehler ansteht, wird in der PLC-Code-Anzeige der Ablauf der GRAPH-Kette dargestellt.

Wenn ein Fehler auftritt, wird in der PLC-Code-Anzeige nach dem Einsprung der gestörte Schritt angezeigt. Falls mehrere Fehler aufgetreten sind, wird nach dem Einsprung der erste gestörte Schritt der Ablaufkette angezeigt. Wenn Sie den Fehler behoben haben, wird in der Anzeige automatisch der nächste fehlerhaften Schritt angezeigt. In der Detailansicht wird je nach Fehlerart die Transition oder die Verriegelung (Interlock) angezeigt.

Bildwechsel von einer Meldung

Sie projektieren die Systemfunktion an ein Ereignis einer Schaltfläche. Beim Betätigen der Schaltfläche in Runtime überprüft die Systemfunktion, ob die zuletzt gewählte Meldung in der konfigurierten Meldeanzeige eine Überwachungsmeldung oder eine GRAPH-Meldung ist. Wenn für diese Meldung der Einsprung möglich ist, wird die konfigurierte PLC-Code-Anzeige mit dem entsprechenden Programmcode geöffnet.

Der Einsprung von einer Meldung in der Meldeanzeige in die PLC-Code-Anzeige wird für folgende Überwachungsmeldungen ermöglicht:

- Bei globalen Überwachungen nur für Verriegelungsüberwachungen (Interlock)
- Bei lokalen Überwachungen für alle Basisüberwachungen an Input-Parametern

Weitere Informationen zu Überwachungen finden Sie im Kapitel "PLC programmieren > Maschinen und Anlagen mit ProDiag überwachen".

Der Einsprung in die PLC-Code-Anzeige ist für alle GRAPH-Meldungen möglich. In der PLC-Code-Anzeige wird nach dem Einsprung die GRAPH-Ablaufkette und der fehlerhafte Schritt angezeigt.

Wenn gleichzeitig eine Überwachungsmeldung und eine Verriegelungsmeldung für den selben Schritt anstehen, wird nach dem Auslösen der Systemfunktion in der PLC-Code-Anzeige immer zuerst das Verriegelungs-Netzwerk angezeigt, unabhängig davon, ob die Überwachungs- oder die Verriegelungsmeldung selektiert war.

Für den Sprung von einer Überwachungsmeldung in die PLC-Code-Anzeige muss bei der Verwendung von unterstützten lokalen Operanden in einem Funktionsbaustein der Instanzname den folgenden Namenskonventionen entsprechen: <FB-Name>_DB.

Der Sprung in eine Funktion oder einen Organisationsbaustein ist nur dann möglich, wenn ausschließlich globale Operanden verwendet werden.

Hinweis

Der Einsprung in Verbindung mit dem Meldepuffer oder Meldearchiv wird nicht unterstützt.

Verwendung in der Funktionsliste

AktivierePLCCodeAnzeige(Bildname, Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

ActivatePLCCodeView Screen_name, Object_name

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Hinweis

Für den Einsprung in die GRAPH-Ablaufkette über die Systemfunktion "AktivierePLCCodeAnzeige" muss die Version des GRAPH-Bausteins mindestens V3.0 betragen.

Hinweis

Die Verwendung der Systemfunktion "AktivierePLCCodeAnzeige" in einem VB-Skript hat an einer GRAPH-Übersicht keine Wirkung.

Parameter

Bildname

Name des Bildes, das die PLC-Code-Anzeige enthält.

Hinweis

Bildwechsel zu einem Pop-up-Bild oder einem Slide-in-Bild wird nicht unterstützt.

Bildobjekt

Objektname der PLC-Code-Anzeige.

PLCCodeAnzeigeBildlaufleisteVerschieben

Beschreibung

Verschiebt die Bildlaufleiste in der PLC-Code-Anzeige. Mit dieser Systemfunktion haben Sie die Möglichkeit, auf bis zu sechs Bildlaufleisten in bestimmten Zeichenbereichen zuzugreifen.

Verwendung in der Funktionsliste

PLCCodeAnzeigeBildlaufleisteVerschieben (Bildobjekt, Bereich, Richtung)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

Bereich

Bereich, in dem die Bildlaufleiste bewegt wird:

Zeichenbereich

Symboltabelle

Detailansicht

Richtung

Richtung, in welche die Bildlaufleiste verschoben wird

Links

Rechts

Oben

Unten

PLCCodeAnzeigeDetails

Beschreibung

Zeigt die Detailansicht in der ausgewählten PLC-Code-Anzeige an.

Verwendung in der Funktionsliste

PLCCodeAnzeigeDetails (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeNächstesNetzwerk

Beschreibung

Navigiert zum nächsten Netzwerk in der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeNächstesNetzwerk (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeZeigeSchrittModus

Beschreibung

Wechselt zwischen der manuellen Schrittauswahl und automatischer Auswahl des aktiven Schrittes in der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeZeigeSchrittModus (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeSymbole

Beschreibung

Zeigt den Symbolbereich des Netzwerks in der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeSymbole (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeTransitionInterlock

Beschreibung

Schaltet zwischen der Anzeige von Transitions- und Interlock-Netzwerken in der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeTransitionInterlock (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeAktualOderErstwerte

Beschreibung

Schaltet zwischen der Anzeige von Aktualwerten und Erstwerten in der PLC-Code-Anzeige.

Hinweis

Voraussetzungen für die Erstwerterfassung

Die Erstwerterfassung ist nur für die GRAPH-Funktionsbausteine ab Version V4.0 möglich.

Für die Erstwerterfassung muss in den Eigenschaften des GRAPH-Bausteins der Haken "Erstwerterfassung" gesetzt werden.

Verwendung in der Funktionsliste

PLCCodeAnzeigeAktualOderErstwerte (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeHeranzoomen

Beschreibung

Vergrößert die Anzeige des Netzwerks im Netzwerkbereich der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeHeranzoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

PLCCodeAnzeigeHerauszoomen

Beschreibung

Verkleinert die Anzeige des Netzwerks im Netzwerkbereich der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeHerauszoomen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige

PLCCodeAnzeigeVorherigesNetzwerk

Beschreibung

Navigiert zum vorherigen Netzwerk in der PLC-Code-Anzeige.

Verwendung in der Funktionsliste

PLCCodeAnzeigeVorherigesNetzwerk (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der PLC-Code-Anzeige.

Funktionen der Sm@rtClient-Anzeige

SmartClientAnzeigeAktualisieren

Beschreibung

Aktualisiert in der Sm@rtClient-Anzeige den angezeigten Inhalt.

Verwendung in der Funktionsliste

SmartClientAnzeigeAktualisieren (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

SmartClientAnzeigeSchreibschutzAus

Beschreibung

Setzt in der Sm@rtClient-Anzeige den Nur-Lese-Zugriff auf "Aus".

Mit dieser Einstellung können Sie ein entferntes Bediengerät bedienen. Verwenden Sie die Systemfunktion "SmartClientAnzeigeSchreibschutzEin", um den Nur-Lese-Zugriff wieder einzuschalten.

Verwendung in der Funktionsliste

SmartClientAnzeigeSchreibschutzAus (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

SmartClientAnzeigeSchreibschutzEin

Beschreibung

Setzt in der Sm@rtClient-Anzeige den Nur-Lese-Zugriff auf "Ein".

Mit dieser Einstellung können Sie ein entferntes Bediengerät nur beobachten. Verwenden Sie die Systemfunktion "SmartClientAnzeigeSchreibschutzAus", um den Nur-Lese-Zugriff wieder auszuschalten.

Verwendung in der Funktionsliste

SmartClientAnzeigeSchreibschutzEin (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

SmartClientAnzeigeTrennen

Beschreibung

Führt in der Sm@rtClient-Anzeige den Befehl "Trennen" aus.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des Bildobjekts nicht benutzt werden soll.

Verwendung in der Funktionsliste

SmartClientAnzeigeTrennen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

SmartClientAnzeigeVerbinden

Beschreibung

Führt in der Sm@rtClient-Anzeige den Befehl "Verbinden" aus.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des Bildobjekts nicht benutzt werden soll. Die Sm@rtClient-Anzeige verbindet sich mit dem projektierten Bediengerät.

Hinweis

Es ist nicht möglich, eine verschlüsselte Verbindung zwischen dem Sm@rtClient und dem Sm@rtServer über die Systemfunktion "SmartClientAnzeigeVerbinden" herzustellen, wenn das Server-Zertifikat nicht im Vorfeld vom Sm@rtClient akzeptiert wurde. Um eine verschlüsselte Verbindung zwischen dem Sm@rtClient und dem Sm@rtServer herzustellen, nutzen Sie die in der Sm@rtClientAnzeige projektierte Schaltfläche "Verbinden" in Runtime.

Verwendung in der Funktionsliste

SmartClientAnzeigeVerbinden (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

SmartClientAnzeigeVerlassen

Beschreibung

Verlässt die Sm@rtClient-Anzeige und kehrt zur Bedienung des Bediengeräts zurück.

Die Verbindung zu dem Bediengerät, das in der Sm@rtClient-Anzeige projektiert ist, bleibt bestehen.

Verwendung in der Funktionsliste

SmartClientAnzeigeVerlassen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Sm@rtClient-Anzeige, in welcher der Befehl ausgelöst wird.

Funktionen der Systemdiagnose-Anzeige

SystemdiagnoseAnzeigeAktivieren

Beschreibung

Aktiviert die System-Diagnoseanzeige. Die System-Diagnoseanzeige zeigt die Detailansicht des betroffenen Geräts.

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeAktivieren(Bildname, Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

ActivateSystemDiagnosticsView Screen_name, Object_name

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Bildname

Name des Bildes, die die System-Diagnoseanzeige enthält.

Bildobjekt

Objektname der System-Diagnoseanzeige.

SystemDiagnoseanzeigeAktualisierePLCBuffer

Beschreibung

Aktualisiert, die in der System-Diagnoseanzeige, die Daten aus dem Diagnosepuffer der Steuerung.

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeAktualisierePLCBuffer (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der System-Diagnoseanzeige.

SystemDiagnoseanzeigeDetailansicht

Beschreibung

Öffnet in einer Systemdiagnose die Detailansicht zur Anzeige ausführlicher Informationen über die selektierte Verbindung und der anliegenden Fehler.

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeDetailansicht (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der System-Diagnoseanzeige, welche in die Detailansicht geschaltet wird.

SystemDiagnoseanzeigeDiagnosepuffer

Beschreibung

Öffnet in einer Systemdiagnose die Ansicht zur Anzeige der aktuellen Daten des Diagnosepuffers.

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeDiagnosepuffer (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der System-Diagnoseanzeige, welche in die Detailansicht geschaltet wird.

SystemDiagnoseanzeigeGerätesicht

Beschreibung

Öffnet in einer Systemdiagnose die Gerätesicht zur Anzeige alle verfügbaren Verbindungen.

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeGerätesicht (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der System-Diagnoseanzeige.

SystemDiagnoseanzeigeZurück

Beschreibung

Wechselt in einer Systemdiagnose zur übergeordneten Ansicht.

- Wenn in der System-Diagnoseanzeige die Diagnosepufferansicht dargestellt ist, öffnet die Funktion die Gerätesicht.
- Wenn in der System-Diagnoseanzeige die Detailansicht dargestellt ist, öffnet die Funktion die Diagnosepuffer-Ansicht..

Verwendung in der Funktionsliste

SystemDiagnoseanzeigeZurück (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der System-Diagnoseanzeige.

ZeigeSystemDiagnosefenster

Beschreibung

Blendet das System-Diagnosefenster am Bediengerät ein oder aus. Das System-Diagnosefenster ist nur im Globalen Bild für Comfort Panels und für WinCC Runtime Advanced verfügbar.

Verwendung in der Funktionsliste

ZeigeSystemDiagnosefenster (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

ShowSystemDiagnosticsWindow Object_name

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Bildobjekt

Name des System-Diagnosefensters, das eingeblendet oder ausgeblendet wird.

Meldungen

BearbeiteMeldung

Beschreibung

Löst das Ereignis "Bearbeiten" für die gewählte Meldung oder bei Mehrfachselektion für die zuletzt gewählte Meldung aus. Wenn Sie die zu bearbeitende Meldungen noch nicht quittiert haben, erfolgt mit Aufruf dieser Systemfunktion automatisch die Quittierung.

Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

Verwendung in der Funktionsliste

BearbeiteMeldung

Verwendung in benutzerdefinierten Funktionen

EditAlarm

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

LöscheMeldepuffer

Beschreibung

Löscht Meldungen aus dem Meldepuffer am Bediengerät.

Hinweis

Noch nicht quittierte Meldungen werden ebenfalls gelöscht.

Verwendung in der Funktionsliste

LöscheMeldepuffer (Nummer der Meldeklasse)

Verwendung in benutzerdefinierten Funktionen

ClearAlarmBuffer Alarm_class_number

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Nummer der Meldeklasse

Legt fest, welche Meldungen aus dem Meldepuffer gelöscht werden:

0 (hmiAll) = Alle Meldungen

1 (hmiAlarms) = Meldungen der Meldeklasse "Fehler"

2 (hmiEvents) = Meldungen der Meldeklasse "Warnungen"

3 (hmiSystem) = Meldungen der Meldeklasse "Systemmeldungen"

4 (hmiS7Diagnosis) = Meldungen der Meldeklasse "S7-Diagnosemeldung"

Hinweis
Geräteabhängigkeit

Meldungen der Meldeklasse "Diagnosis Events" stehen auf Basic Panels nicht zur Verfügung.

LöscheMeldepufferProTool

Beschreibung

Diese Systemfunktion existiert aus Kompatibilitätsgründen.

Sie hat die gleiche Funktionalität wie die Systemfunktion "LöscheMeldepuffer", verwendet aber die alte ProTool-Nummerierung.

Verwendung in der Funktionsliste

LöscheMeldepufferProTool (Nummer der Meldeklasse)

Verwendung in benutzerdefinierten Funktionen

ClearAlarmBufferProtoolLegacy Alarm_class_number

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Nummer der Meldeklasse

Nummer der Meldeklasse, deren Meldungen gelöscht werden sollen:

-1 (hmiAllProtoolLegacy) = Alle Meldungen

0 (hmiAlarmsProtoolLegacy) = Meldungen der Meldeklasse "Errors"

1 (hmiEventsProtoolLegacy) = Meldungen der Meldeklasse "Warnings"

2 (hmiSystemProtoolLegacy) = Meldungen der Meldeklasse "System"

3 (hmiS7DiagnosisProtoolLegacy) = Meldungen der Meldeklasse "Diagnosis Events"

Hinweis

Geräteabhängigkeit

Meldungen der Meldeklasse "Diagnosis Events" stehen auf Basic Panels nicht zur Verfügung.

MeldeanzeigeAktualisieren

Beschreibung

Aktualisiert die erweiterte Meldeanzeige.

Verwendung in der Funktionsliste

MeldeanzeigeAktualisieren (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Meldeanzeige, die aktualisiert wird.

MeldeanzeigeLoopInAlarm

Beschreibung

Löst das Ereignis "Loop-In-Alarm" für die gewählte Meldung oder bei Mehrfachselektion die zuletzt gewählte Meldung aus.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des ActiveX-Controls nicht benutzt werden soll.

An das Ereignis "Loop-In-Alarm" können Sie wiederum eine Systemfunktion projektieren. Beispielsweise können Sie zu dem Prozessbild wechseln, in dem die Meldung aufgetreten ist.

Hinweis

Wenn Sie die zu bearbeitenden Meldungen noch nicht quittiert haben, erfolgt mit Aufruf dieser Systemfunktion automatisch die Quittierung.

Verwendung in der Funktionsliste

MeldeanzeigeLoopInAlarm (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

MeldeanzeigeQuittiereMeldung

Beschreibung

Quittiert die Meldungen, die in der angegebenen Meldeanzeige angewählt sind.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des ActiveX-Controls nicht benutzt werden soll.

Verwendung in der Funktionsliste

MeldeanzeigeQuittiereMeldung (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

MeldeanzeigeZeigeHilfetext

Beschreibung

Zeigt den projektierten Tooltip der Meldung an, die in der angegebenen Meldeanzeige ausgewählt ist.

Verwendung in der Funktionsliste

MeldeanzeigeZeigeHilfetext (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

Siehe auch

Systemfunktionen für Runtime Advanced (Seite 1562)

QuittiereMeldung

Beschreibung

Quittiert alle ausgewählten Meldungen.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät keine ACK-Taste besitzt oder die integrierte Taste der Meldeanzeige nicht genutzt werden soll.

Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

Verwendung in der Funktionsliste

QuittiereMeldung

Verwendung in benutzerdefinierten Funktionen

AcknowledgeAlarm

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

ZeigeMeldefenster

Beschreibung

Blendet das Meldefenster am Bediengerät ein oder aus.

Verwendung in der Funktionsliste

ZeigeMeldefenster (Objektname, Darstellung)

Verwendung in benutzerdefinierten Funktionen

ShowAlarmWindow Object_name, Display_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Objektname

Name der Meldeanzeige, die eingeblendet oder ausgeblendet wird.

Darstellung

Legt fest, ob das Meldefenster eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Meldeanzeige wird ausgeblendet

1 (hmiOn) = Ein: Meldeanzeige wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

ZeigeSystemmeldung

Beschreibung

Zeigt den Wert des übergebenen Parameters als Systemmeldung am Bediengerät an.

Verwendung in der Funktionsliste

ZeigeSystemmeldung (Text/Wert)

Verwendung in benutzerdefinierten Funktionen

ShowSystemAlarm Text_or_value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Text/Wert

Der Text oder der Wert, der als Systemmeldung ausgegeben wird.

Rezepturen

ExportiereDatensätze

Beschreibung

Exportiert einen oder alle Datensätze einer Rezeptur in eine CSV-Datei oder eine TXT-Datei.

Pro Rezeptur wird eine Datei angelegt.

Verwendung in der Funktionsliste

ExportiereDatensätze (Rezepturnummer/-name, Datensatznummer/-name, Dateiname, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

ExportDataRecords Recipe_number/name, Data_record number/name, File_name, Overwrite, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der Rezepturdatensätze exportiert werden. Geben Sie "0" an, wenn Sie aus allen verfügbaren Rezepturen Rezepturdatensätze exportieren wollen.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der exportiert wird. Geben Sie "0" an, wenn Sie alle Rezepturdatensätze exportieren wollen.

Dateiname

Name der Datei, in welche die Rezepturdatensätze exportiert werden. Geben Sie den Namen einschließlich des Ablageorts und der Dateierweiterung (*.csv oder *.txt) an, z.B. "C:\TEMP \Orange.csv". Wenn ein Ordner nicht existiert, wird er beim Export angelegt.

Wenn Sie den Dateinamen nicht vollständig eingeben, ist das Verhalten abhängig von der anzahl der projektierten Rezepturen:

- Wenn mehrere Rezepturen projektiert sind und
  Sie nur einen Dateinamen und keinen Pfad angeben, wird die Datei in einem Systemverzeichnis, z.B. "C:\Documents and Settings\[User]", abgelegt.
  Sie nur einen Pfad und keinen Dateinamen angeben, wird der Dateiname aus dem Rezepturnamen erzeugt. Die Voraussetzung ist, dass der im Pfad angebene Ordner, z. B. "D:\Data\", angelegt ist. Wenn der Ordner "D:\Temp" nicht angelegt ist, wird der Ordnername als Präfix für den Dateinamen verwendet,Temp_Rezepturnamen.csv.
- Wenn nur eine Rezeptur projektiert ist und Sie nur einen Pfad und keinen Dateinamen angeben,
  wird, wenn der Ordner nicht existiert eine Datei mit dem Ordnernamen angelegt. Diese besitzt jedoch keine Dateierweiterung.
  wird, wenn der Ordner existiert, der Export mit einer Fehlermeldung abgebrochen.

Wenn Sie als Ablageort eine Speicherkarte verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Name>".

Für Basic Panel geben Sie den Dateinamen wie folgt an: "\USB_X60.1\<Name>"

Überschreiben

Legt fest, ob eine bereits vorhandene Export-Datei gleichen Namens überschrieben wird:

0 (hmiOverwriteForbidden) = Nein: Export-Datei wird nicht überschrieben. Der Exportvorgang wird nicht ausgeführt.

1 (hmiOverwriteAlways) = Ja: Export-Datei wird ohne Nachfrage überschrieben.

2 (hmiOverwriteWithPrompting) = Nach Bestätigung: Export-Datei wird erst nach Bestätigung überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach dem Export eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Exportformat der Rezepturdatensätze

Wenn für die Exportdatei die Dateierweiterung ".csv" gewählt wird, dann werden lediglich gültige Zeichen aus dem ANSI-Zeichensatz unterstützt. Dies gilt auch für Trennzeichen bei Dezimalzahlen und Listenelementen. Die verwendeten Trennzeichen werden in den Ländereinstellungen des Betriebssystems des exportierenden Computers festgelegt.

Für den Export kann auch das Dateiformat "Unicode Text" (".txt") gewählt werden. Dieses Dateiformat unterstützt alle in WinCC und WinCC Runtime verwendbaren Zeichen. Die verwendeten Trennzeichen werden auch hier in den Ländereinstellungen des Betriebssystems des exportierenden Computers festgelegt. Bei diesem Dateiformat wird als Trennzeichen für Listenelemente immer der Tabulator verwendet.

Die zugehörige Funktion zum Import der Daten unterstützt ebenfalls die beiden Dateiformate ".csv" und ".txt" (Unicode).

Anwendungsbeispiel

Sie wollen über eine Taste alle Datensätze exportieren.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktion "ExportiereDatensätze" an das Ereignis "Drücken" der gewünschten Taste. Übergeben Sie folgende Parameter:

- Rezepturnummer/-name = 1
- Datensatznummer/-name = 0
- Dateiname = c:\temp\orange.csv (für Basic Panels "\USB_X60.1\orange.csv")
- Überschreiben = 1
- Statusmeldung ausgeben = 1

An Stelle der Konstanten können Sie auch Variablen angeben. Je nach Projektierung kann der Bediener die gewünschten Werte in E/A-Felder eingeben oder aus der Steuerung lesen. Auf diese Weise kann der Bediener bestimmen, welche Rezepturdatensätze exportiert werden.

Ablauf am Bediengerät

Sobald die Taste betätigt wird, wird die Systemfunktion ausgelöst. Die Konstanten oder Variablen werden ausgewertet. Alle Datensätze der Rezeptur 1 werden in die Datei orange.csv exportiert. Wenn die Datei bereits vorhanden ist, wird sie überschrieben.

Nach dem Export der Datensätze wird eine Systemmeldung ausgegeben.

ExportiereDatensätzeMitPrüfsumme

Beschreibung

Exportiert einen oder alle Datensätze einer Rezeptur in eine CSV-Datei und generiert für jede Zeile in der Datei eine Prüfsumme.

Pro Rezeptur wird eine Datei angelegt.

Verwendung in der Funktionsliste

ExportiereDatensätzeMitPrüfsumme (Rezepturnummer/-name, Datensatznummer/-name, Dateiname, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

ExportDataRecordsWithChecksum Recipe_number_or_name, Data_record number_or_name, File_name, Overwrite, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der Rezepturdatensätze exportiert werden. Geben Sie "0" an, wenn Sie aus allen verfügbaren Rezepturen Rezepturdatensätze exportieren wollen.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der exportiert wird. Geben Sie "0" an, wenn Sie alle Rezepturdatensätze exportieren wollen.

Dateiname

Name der CSV-Datei, in welche die Rezepturdatensätze exportiert werden. Geben Sie den Pfad und die Dateiendung mit an z.B. "C:\TEMP\Orange.CSV".

Wenn Sie als Ablageort eine Speicherkarte verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Dateiname>".

Wenn Sie nur einen Dateinamen und keinen Pfad festlegen, wird die Datei in dem Verzeichnis abgelegt, aus dem die Runtime gestartet wurde. Wenn unter Betriebssystem Windows 7 keine Schreibrechte für das Verzeichnis bestehen, wird die Datei im Benutzerverzeichnis im Ordner "VirtualStore" abgelegt.

Wenn Sie nur den Pfad für den Export angeben, wird der Dateiname jeweils automatisch aus dem Rezepturnamen erzeugt. Die Voraussetzung ist, dass das Verzeichnis z. B. "D:\Temp\" angelegt ist. Wenn das Verzeichnis "D:\Temp" nicht angelegt ist, wird der Verzeichnisname als Präfix für den Dateinamen verwendet,Temp_Rezepturnamen.csv.

Überschreiben

Legt fest, ob eine bereits vorhandene CSV-Datei gleichen Namens überschrieben wird:

0 (hmiOverwriteForbidden) = Nein: CSV-Datei wird nicht überschrieben. Der Exportvorgang wird nicht ausgeführt.

1 (hmiOverwriteAlways) = Ja: CSV-Datei wird ohne Nachfrage überschrieben.

2 (hmiOverwriteWithConfirmation) = Nach Bestätigung: CSV-Datei wird erst nach Bestätigung überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach dem Export eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Exportformat der Rezepturdatensätze

Wenn für die Exportdatei die Dateierweiterung ".csv" gewählt wird, dann werden lediglich gültige Zeichen aus dem ANSI-Zeichensatz unterstützt. Dies gilt auch für Trennzeichen bei Dezimalzahlen und Listenelementen. Die verwendeten Trennzeichen werden in den Ländereinstellungen des Betriebssystems des exportierenden Computers festgelegt.

Für den Export kann auch das Dateiformat "Unicode Text" (".txt") gewählt werden. Dieses Dateiformat unterstützt alle in WinCC und WinCC Runtime verwendbaren Zeichen. Die verwendeten Trennzeichen werden auch hier in den Ländereinstellungen des Betriebssystems des exportierenden Computers festgelegt. Bei diesem Dateiformat wird als Trennzeichen für Listenelemente immer der Tabulator verwendet.

Die zugehörige Funktion zum Import der Daten unterstützt ebenfalls die beiden Dateiformate ".csv" und ".txt" (Unicode).

Anwendungsbeispiel

Sie wollen über eine Taste alle Datensätze exportieren und mit einer Prüfsumme versehen.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktion "ExportiereDatensätzeMitPrüfsumme" an das Ereignis "Drücken" der gewünschten Taste. Übergeben Sie folgende Parameter:

- Rezepturnummer/-name = 1
- Datensatznummer/-name = 0
- Dateiname = c:\temp\orange.csv
- Überschreiben = 1
- Statusmeldung ausgeben = 1

An Stelle der Konstanten können Sie auch Variablen angeben. Je nach Projektierung kann der Bediener die gewünschten Werte in E/A-Felder eingeben oder aus der Steuerung lesen. Auf diese Weise kann der Bediener bestimmen, welche Rezepturdatensätze exportiert werden.

Ablauf am Bediengerät

Sobald die Taste betätigt wird, wird die Systemfunktion ausgelöst. Die Konstanten oder Variablen werden ausgewertet. Alle Datensätze der Rezeptur 1 werden in die Datei orange.csv exportiert und mit Prüfsummen versehen. Wenn die Datei bereits vorhanden ist, wird sie überschrieben.

Nach dem Export der Datensätze wird eine Systemmeldung ausgegeben.

ImportiereDatensätze

Beschreibung

Importiert einen oder alle Datensätze einer Rezeptur aus einer CSV-Datei oder einer TXT-Datei.

Wenn ein Pfad angegeben wird, werden alle Dateien des angegebenen Verzeichnisses importiert.

Verwendung in der Funktionsliste

ImportiereDatensätze (Dateiname, Datensatznummer/-name, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

ImportDataRecords File_name, Data_record_number_or_name, Overwrite, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Dateiname

Name der Datei, aus der die Rezepturdatensätze importiert werden. Geben Sie den Ablageort und die Dateierweiterung (*.csv oder *.txt) mit an, z.B. "C:\TEMP\Orange.csv",

Für Basic Panel geben Sie den Dateinamen wie folgt an: "\USB_X60.1\<Name>"

Für andere Geräte gilt: Wenn Sie als Ablageort eine Speicherkarte verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Name>"

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der importiert wird. Geben Sie "0" an, wenn Sie alle Rezepturdatensätze importieren wollen.

Überschreiben

Legt fest, ob bereits vorhandene Rezepturdatensätze überschrieben werden:

0 (hmiOverwriteForbidden) = Nein: Rezepturdatensätze werden nicht überschrieben. Der Importvorgang wird nicht ausgeführt.

1 (hmiOverwriteAlways) = Ja: Rezepturdatensätze werden ohne Nachfrage überschrieben.

2 (hmiOverwriteWithPrompting) = Nach Bestätigung: Rezepturdatensätze werden erst nach Bestätigung überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach dem Import eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Projektierbare Objekte

Objekt	Ereignis
Variable	Wertänderung Obergrenze überschritten Untergrenze unterschritten
Funktionstaste (global)	Loslassen Drücken
Funktionstaste (lokal)	Loslassen Drücken
Bild	Aufgebaut Abgebaut
Bildobjekt	Drücken Loslassen Klicken Ändern (oder Umschalten beim Schalter) Einschalten Ausschalten Aktivieren Deaktivieren
Aufgabenplaner	Zeit abgelaufen

ImportiereDatensätzeMitPrüfsumme

Beschreibung

Importiert einen oder alle Datensätze einer Rezeptur aus einer CSV-Datei mit Prüfsumme und verifiziert die Prüfsumme.

Verwendung in der Funktionsliste

ImportiereDatensätzeMitPrüfsumme (Dateiname, Datensatznummer/-name, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus, Prüfsumme verifizieren)

Verwendung in benutzerdefinierten Funktionen

ImportDataRecordsWithChecksum File_name, Data_record_number_or_name, Overwrite, Output_status_message, Processing_status, Verify_checksum

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Dateiname

Name der CSV-Datei, aus der die Rezepturdatensätze importiert werden. Geben Sie den Pfad und die Dateierweiterung mit an, z. B. "C:\TEMP\Orange.CSV".

Wenn Sie als Speichermedium eine Storage Card verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Dateiname>".

Wenn Sie ein Verzeichnis statt einer einzelnen CSV-Datei angeben, werden alle Dateien des angegebenen Verzeichnisses importiert.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der importiert wird. Geben Sie "0" an, wenn Sie alle Rezepturdatensätze importieren wollen.

Überschreiben

Legt fest, ob bereits vorhandene Rezepturdatensätze überschrieben werden:

0 (hmiOverwriteForbidden) = Nein: Rezepturdatensätze werden nicht überschrieben. Der Importvorgang wird nicht ausgeführt.

1 (hmiOverwriteAlways) = Ja: Rezepturdatensätze werden ohne Nachfrage überschrieben.

2 (hmiOverwriteWithConfirmation) = Nach Bestätigung: Rezepturdatensätze werden erst nach Bestätigung überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach dem Import eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z. B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Prüfsumme verifizieren

Legt fest, ob die Prüfsumme beim Import verifiziert wird:

0 (hmiFalse) = Nein: Prüfsumme wird nicht verifiziert.

1 (hmiTrue) = Ja: Prüfsumme wird verifiziert.

LadeDatensatz

Beschreibung

Lädt den angegebenen Rezepturdatensatz vom Speichermedium des Bediengeräts in die Rezepturvariablen. Verwenden Sie diese Systemfunktion, um z. B. einen Rezepturdatensatz im Rezepturbild anzuzeigen.

Aktivieren Sie in den Einstellungen zur Synchronisation der Rezeptur die Option "Rezepturanzeige und Rezepturvariablen abgleichen". Wenn diese Option deaktiviert ist, hat die Systemfunktion keine Auswirkung.

Verwendung in der Funktionsliste

LadeDatensatz (Rezepturnummer/-name, Datensatznummer/-name, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

LoadDataRecord Recipe_number_or_name, Data_record_number_or_name, Confirmation, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der ein Rezepturdatensatz geladen wird.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der geladen wird.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

LeseDatensatzAusSteuerung

Beschreibung

Überträgt den angegebenen Rezepturdatensatz aus der Steuerung auf das Speichermedium des Bediengeräts.

Verwendung in der Funktionsliste

LeseDatensatzAusSteuerung (Rezepturnummer/-name, Datensatznummer/-name, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

GetDataRecordFromPLC Recipe_number_or_name, Data_record_number_or_name, Overwrite, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der Rezepturdatensätze übertragen werden.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der aus der Steuerung auf den Datenträger des Bediengeräts übertragen wird.

Überschreiben

Legt fest, ob ein bereits vorhandener Rezepturdatensatz gleichen Namens überschrieben wird:

0 (hmiOverwriteForbidden) = Nein: Rezepturdatensatz wird nicht überschrieben. Der Übertragungsvorgang wird nicht ausgeführt.

1 (hmiOverwriteAlways) = Ja: Rezepturdatensatz wird ohne Nachfrage überschrieben.

2 (hmiOverwriteWithPrompting) = Nach Bestätigung: Rezepturdatensatz wird erst nach Bestätigung überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach der Übertragung eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Anwendungsbeispiel

Sie wollen über eine Taste einen Datensatz aus der Steuerung auf den Datenträger des Bediengeräts übertragen.

Hinweise zur Projektierung

Projektieren Sie die Systemfunktion "LeseDatensatzAusSteuerung" an das Ereignis "Drücken" der gewünschten Taste. Übergeben Sie folgende Parameter:

Rezepturnummer/-name = 1

Datensatznummer/-name = 1

Überschreiben = 1

Statusmeldung ausgeben = 1

An Stelle der Konstanten können Sie auch Variablen angeben. Je nach Projektierung kann der Bediener die gewünschten Werte in E/A-Felder eingeben oder aus der Steuerung lesen. Auf diese Weise kann der Bediener bestimmen, welcher Rezepturdatensatz aus der Steuerung übertragen wird.

Ablauf am Bediengerät

Sobald die Taste betätigt wird, wird die Systemfunktion ausgelöst. Die Konstanten oder Variablen werden ausgewertet und der erste Datensatz der Rezeptur 1 wird aus der Steuerung auf den Datenträger des Bediengeräts übertragen. Ist der Rezepturdatensatz bereits vorhanden, wird er überschrieben.

Nach der Übertragung wird eine Systemmeldung ausgegeben.

LeseDatensatzname

Beschreibung

Schreibt die Namen der angegebenen Rezeptur und des angegebenen Rezepturdatensatzes in die angegebenen Variablen.

Hinweis

Wenn die Rezeptur oder der Rezepturdatensatz nicht existiert, werden Platzhalter ("###") in die Variablen geschrieben.

Hinweis

Als Variablen werden nur interne oder externe Variablen unterstützt.

Verwendung in der Funktionsliste

LeseDatensatzname (Rezepturnummer, Datensatznummer, Rezepturname, Datensatzname, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

GetDataRecordName Recipe_number, Data_record_number, Recipe_name, Data_record_name, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer

Nummer der Rezeptur, deren Name in die angegebene Variable geschrieben wird.

Datensatznummer

Nummer des Rezepturdatensatzes, dessen Name in die angegebene Variable geschrieben wird.

Rezepturname

Die Variable, in die der Rezepturname geschrieben wird. Die Variable muss vom Typ STRING sein.

Datensatzname

Die Variable, in die der Name des Rezepturdatensatzes geschrieben wird. Die Variable muss vom Typ STRING sein.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

Anwendungsbeispiel

Sie wollen die Namen der angezeigten Rezeptur und den Namen des angezeigten Rezepturdatensatzes am Bediengerät ausgeben.

Projektieren Sie folgende Variablen:

- "RecNumber" vom Typ INTEGER
- "RecDataNumber" vom Typ INTEGER

- "RecName" vom Typ STRING
- "RecDataName" vom Typ STRING

Projektieren Sie eine Rezepturanzeige mit den Variablen "RecNumber" für die Rezepturnummer und "RecDataNumber" für die Datensatznummer.

Projektieren Sie an das Ereignis "Drücken" einer Schaltfläche die Systemfunktion "LeseDatensatzname" und übergeben Sie folgende Parameter:

- Rezepturnummer: RecNumber
- Datensatznumer: RecDataNumber
- Rezepturname: RecName
- Datensatzname: RecDataName

Projektieren Sie zwei Ausgabefelder und verbinden Sie diese mit den Variablen "RecName" und "RecDataName".

Wählen Sie in der Rezepturanzeige eine Rezeptur- und eine Datensatznummer aus. Sobald Sie die Schaltfläche betätigen, wird die Systemfunktion ausgelöst und die Namen der Rezeptur und des Rezepturdatensatzes werden in den beiden Ausgabefeldern angezeigt.

LeseDatensatzvariablenAusSteuerung

Beschreibung

Überträgt die Werte des Rezepturdatensatzes, der in der Steuerung geladen ist, in die Rezepturvariablen.

Verwenden Sie diese Systemfunktion z.B. beim Teach-In-Betrieb an einer Maschine.

Verwendung in der Funktionsliste

LeseDatensatzvariablenAusSteuerung (Rezepturnummer/-name, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

GetDataRecordTagsFromPLC Recipe_number_or_name, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name des Rezepturdatensatzes, dessen Werte aus der Steuerung in die Variablen geschrieben werden.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

LöscheDatensatz

Beschreibung

Löscht einen Rezepturdatensatz.

Mehrere Datensätze können in einer oder mehreren Rezepturen gelöscht werden.

Verwendung in der Funktionsliste

LöscheDatensatz (Rezepturnummer/-name, Datensatznummer/-name, Bestätigung, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

ClearDataRecord

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der Rezepturdatensätze gelöscht werden. Wenn Sie aus allen verfügbaren Rezepturen Rezepturdatensätze löschen wollen, geben Sie "0" an.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der gelöscht wird. Wenn Sie alle Rezepturdatensätze löschen wollen, geben Sie "0" an.

Bestätigung

Legt fest, ob der Bediener das Löschen bestätigen soll:

0 (hmiOff) = Aus: Löschvorgang wird ohne Bestätigung gestartet.

1 (hmiOn) = Ein: Beginn des Löschvorgangs muss bestätigt werden.

Statusmeldung ausgeben

Legt fest, ob nach dem Löschen eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

LöscheDatensatzSpeicher

Beschreibung

Löscht alle Rezepturdatensätze vom angegebenen Speichermedium.

Verwendung in der Funktionsliste

LöscheDatensatzSpeicher (Speicherort, Nach Bestätigung, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

ClearDataRecordMemory Storage_location, Confirmation, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Speicherort

Legt den Speicherort fest:

0 (hmiFlashMemory) = Flash-Speicher: Interner Flash-Speicher des Bediengeräts

1 (hmiStorageCard) = Storage Card

2 (hmiStorageCard2) = Storage Card 2

3 (hmiStorageCard3) = Storage Card MultiMediaCard

4 (hmiStorageCard4) = Storage Card USB

Nach Bestätigung

Legt fest, ob der Bediener das Löschen bestätigen soll:

0 (hmiOff) = Aus: Löschvorgang wird ohne Bestätigung gestartet.

1 (hmiOn) = Ein: Beginn des Löschvorgangs muss bestätigt werden.

Statusmeldung ausgeben

Legt fest, ob nach dem Löschen eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

RezepturanzeigeDatensatzNeu

Beschreibung

Erstellt in der angegebenen Rezepturanzeige einen neuen Datensatz.

Verwendung in der Funktionsliste

RezepturanzeigeDatensatzNeu (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der neue Rezepturdatensatz angelegt wird.

RezepturanzeigeLeseDatensatzAusSteuerung

Beschreibung

Überträgt den Datensatz, der gerade in der Steuerung geladen ist, an das Bediengerät und zeigt ihn in der Rezepturanzeige an.

Verwendung in der Funktionsliste

RezepturanzeigeLeseDatensatzAusSteuerung (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der aus der Steuerung übertragene Rezepturdatensatz angezeigt wird.

RezepturanzeigeLöscheDatensatz

Beschreibung

Löscht den Datensatz, der gerade in der Rezepturanzeige angezeigt wird.

Verwendung in der Funktionsliste

RezepturanzeigeLöscheDatensatz (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der angezeigte Rezepturdatensatz gelöscht wird.

RezepturanzeigeMenü

Beschreibung

Öffnet das Menü der angegebenen einfachen Rezepturanzeige.

Verwenden Sie diese Systemfunktion nur bei einer einfachen Rezepturanzeige.

Verwendung in der Funktionsliste

RezepturanzeigeMenü (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in der das Menü geöffnet werden soll.

RezepturanzeigeÖffnen

Beschreibung

Zeigt in der angegebenen Rezepturanzeige die Datensatzwerte an oder wechselt zum nächsten Auswahlfeld. Die Systemfunktion hat keine Wirkung, wenn am Bediengerät das Auswahlfeld für die Rezepturdatensatzwerte angezeigt wird.

Bedienreihenfolge der Auswahlfelder in Runtime:

- Rezepturname
- Datensatzname
- Rezepturdatensatzwerte

Verwenden Sie diese Systemfunktion, wenn Sie die einfache Rezepturanzeige projektiert haben. In der einfachen Rezepturanzeige wird jeweils nur ein Auswahlfeld am Bediengerät angezeigt. Verwenden Sie die Systemfunktion "RezepturanzeigeZurück", um das vorherige Auswahlfeld anzuzeigen.

Verwendung in der Funktionsliste

RezepturanzeigeÖffnen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der Befehl ausgelöst wird.

Siehe auch

RezepturanzeigeZurück (Seite 1712)

RezepturanzeigeSchreibeDatensatzInSteuerung

Beschreibung

Überträgt den Rezepturdatensatz, der gerade in der Rezepturanzeige angezeigt wird, zur Steuerung.

Verwendung in der Funktionsliste

RezepturanzeigeSchreibeDatensatzInSteuerung (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, aus welcher der Rezepturdatensatz zur verbundenen Steuerung übertragen wird.

RezepturanzeigeSpeichereDatensatz

Beschreibung

Speichert den Rezepturdatensatz, der gerade in der Rezepturanzeige angezeigt wird.

Verwendung in der Funktionsliste

RezepturanzeigeSpeichereDatensatz (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der Rezepturdatensatz gespeichert wird.

RezepturanzeigeSpeichernUnterDatensatz

Beschreibung

Speichert den Datensatz, der gerade in der Rezepturanzeige angezeigt wird, unter einem neuen Namen.

Verwendung in der Funktionsliste

RezepturanzeigeSpeichernUnterDatensatz (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Speichert den Datensatz, der in der Rezepturanzeige angezeigt wird, unter einem neuen Namen und/oder eine andere Nummer.

RezepturanzeigeSynchronisiereDatensatzMitVariablen

Beschreibung

Synchronisiert die Werte des Datensatzes, der gerade in der Rezepturanzeige angezeigt wird, mit den dazugehörigen Rezepturvariablen. Verwenden Sie diese Systemfunktion nur bei der erweiterten Rezepturanzeige.

Während der Synchronisation werden alle Werte des Datensatzes in die dazugehörigen Rezepturvariablen geschrieben.

Verwendung in der Funktionsliste

RezepturanzeigeSynchronisiereDatensatzMitVariablen (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher die Werte mit den dazugehörigen Variablen synchronisiert werden.

RezepturanzeigeUmbenennenDatensatz

Beschreibung

Benennt in der angegebenen Rezepturanzeige den ausgewählten Datensatz um.

Verwendung in der Funktionsliste

RezepturanzeigeUmbenennenDatensatz (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der Rezepturdatensatz umbenannt wird.

RezepturanzeigeZeigeHilfetext

Beschreibung

Zeigt den projektierten Tooltip der angegebenen Rezepturanzeige an.

Verwendung in der Funktionsliste

RezepturanzeigeZeigeHilfetext (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, deren projektierter Tooltip angezeigt wird.

RezepturanzeigeZurück

Beschreibung

Kehrt in der angegebenen Rezepturanzeige zum vorherigen Auswahlfeld zurück.

Die Systemfunktion hat keine Wirkung, wenn am Bediengerät das Auswahlfeld für die Rezeptur angezeigt wird. Bedienreihenfolge der Auswahlfelder in Runtime:

- Rezepturname
- Datensatzname
- Rezepturdatensatzwerte

Verwenden Sie diese Systemfunktion, wenn Sie die einfache Rezepturanzeige projektiert haben. In der einfachen Rezepturanzeige wird jeweils nur ein Auswahlfeld am Bediengerät angezeigt. Verwenden Sie die Systemfunktion "RezepturanzeigeÖffnen", um die Rezepturdatensatzwerte oder das nächste Auswahlfeld anzuzeigen.

Verwendung in der Funktionsliste

RezepturanzeigeZurück (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name der Rezepturanzeige, in welcher der Befehl ausgelöst wird.

SchreibeDatensatzInSteuerung

Beschreibung

Überträgt den angegebenen Rezepturdatensatz direkt vom Datenträger des Bediengeräts an die Steuerung, die mit dem Bediengerät verbunden ist.

Hinweis

Die Werte des Rezepturdatensatzes müssen dabei nicht am Bediengerät angezeigt werden.

Verwendung in der Funktionsliste

SchreibeDatensatzInSteuerung (Rezepturnummer/-name, Datensatznummer/-name, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

SetDataRecordToPLC Recipe_number_or_name, Data_record_number_or_name, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der ein Rezepturdatensatz an die Steuerung übertragen wird.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der an die Steuerung übertragen wird.

Statusmeldung ausgeben

Legt fest, ob nach der Übertragung eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

SchreibeDatensatzvariablenInSteuerung

Beschreibung

Überträgt die Werte der Rezepturvariablen in die Steuerung. Die Rezepturvariablen enthalten die Werte des Datensatzes, der am Bediengerät angezeigt wird

Verwendung in der Funktionsliste

SchreibeDatensatzvariablenInSteuerung (Rezepturnummer/-name, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

SetDataRecordTagsToPLC Recipe_ number_or_name, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, aus der ein Rezepturdatensatz an die Steuerung übertragen wird.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

SetzeRezepturvariablen

Beschreibung

Ändert den Status der Rezepturvariablen von "Online" nach "Offline" und umgekehrt

Verwenden Sie diese Systemfunktion, wenn Sie z.B. bei der Inbetriebnahme einer Maschine noch die Rezepturdatensatzwerte fein abstimmen.

Verwendung in der Funktionsliste

SetzeRezepturvariablen (Rezepturnummer/-name, Status, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

SetRecipeTags Recipe_number_or_name, Status, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, in welcher der Rezepturdatensatz gespeichert ist.

Status

Legt den Status der Rezepturvariablen fest:

0 (hmiOnline) = Online: Wertänderungen an den Rezepturvariablen werden sofort an die mit dem Bediengerät verbundene Steuerung übertragen.

1 (hmiOffline) = Offline: Wertänderungen an den Rezepturvariablen werden erst an die mit dem Bediengerät verbundene Steuerung übertragen, wenn Sie z.B. die Systemfunktion "SchreibeDatensatzvariablenInSteuerung" ausführen.

Statusmeldung ausgeben

Legt fest, ob nach dem Speichern eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

SpeichereDatensatz

Beschreibung

Speichert die aktuellen Werte der Rezepturvariablen als Datensatz auf dem Speichermedium des Bediengeräts.

Verwenden Sie diese Systemfunktion, um z.B. einen Rezepturdatensatz im Rezepturbild zu speichern.

Verwendung in der Funktionsliste

SpeichereDatensatz (Rezepturnummer/-name, Datensatznummer/-name, Überschreiben, Statusmeldung ausgeben, Bearbeitungsstatus)

Verwendung in benutzerdefinierten Funktionen

SaveDataRecord Recipe_number_or_name, Data_record_number_or_name, Overwrite, Output_status_message, Processing_status

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Rezepturnummer/-name

Nummer oder Name der Rezeptur, in der ein Rezepturdatensatz gespeichert wird.

Datensatznummer/-name

Nummer oder Name des Rezepturdatensatzes, der gespeichert wird. Wenn in der Rezeptur ein Datensatz mit dieser Nummer oder diesem Namen nicht existiert, wird unabhängig vom Wert des Parameters "Überschreiben" ein neuer Datensatz angelegt.

Überschreiben

Legt fest, ob ein bereits vorhandener Datensatz überschrieben wird:

0 (hmiOverwriteForbidden) = Nein: Der Rezepturdatensatz wird nicht überschrieben, der Datensatz wird nicht gespeichert.

1 (hmiOverwriteAlways) = Ja: Der Rezepturdatensatz wird ohne Nachfrage überschrieben.

2 (hmiOverwriteWithConfirmation) = Nach Bestätigung: Der Rezepturdatensatz wird erst nach Bestätigung durch den Benutzer überschrieben.

Statusmeldung ausgeben

Legt fest, ob nach dem Speichern eine Statusmeldung ausgegeben wird:

0 (hmiOff) = Aus: Statusmeldung wird nicht ausgegeben.

1 (hmiOn) = Ein: Statusmeldung wird ausgegeben.

Bearbeitungsstatus

Gibt den Bearbeitungsstatus der Systemfunktion zurück. Verwenden Sie den Rückgabewert, um z.B. andere Systemfunktionen erst dann auszuführen, wenn diese Systemfunktion erfolgreich beendet wurde:

2 = Systemfunktion wird gerade ausgeführt.

4 = Systemfunktion wurde erfolgreich beendet.

12 = Systemfunktion wurde nicht ausgeführt, weil ein Fehler aufgetreten ist.

System

AktivierePutzbild

Beschreibung

Aktiviert das Putzbild am Bediengerät. Dabei wird das Display des Bediengeräts für den angegebenen Zeitraum deaktiviert.

Während das Display des Bediengeräts deaktiviert ist, können Sie es reinigen, ohne unbeabsichtigt Touch-Funktionen auszulösen.

Verwendung in der Funktionsliste

AktivierePutzbild (Zeitraum)

Verwendung in benutzerdefinierten Funktionen

--

Parameter

Zeitraum

Zeitraum, in dem das Display deaktiviert ist. Die verbleibende Zeit wird als Fortschrittsbalken angezeigt.

Wertebereich in Sekunden von 10 bis 300.

Hinweis

Die Systemfunktion AktivierePutzbild kann nicht simuliert werden.

Siehe auch

Abmelden (Seite 1612)

ÖffneDateiBrowser

Beschreibung

Öffnet einen Datei-Browserdialog.

Abhängig vom Modus kann der Benutzer eine Datei oder einen Ordner auswählen.

Verwendung in der Funktionsliste

ÖffneDateiBrowser (Maske, Ausgangsordner, Datei, Navigation Zurück, Navigation Aufwärts, Neuer Ordner, Pfad, Status)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Maske

Legt eine Maske fest, um die angezeigten Dateien zu filtern. Dieser Parameter wird nur im Modus "Datei" ausgewertet. Mehrere Filterkriterien müssen durch ; getrennt werden. Beispiel: *.htm;*.html

Ausgangsordner

Legt den Ordner fest, mit dem der Dateiauswahl-Dialog gestartet wird.

Datei

Legt den Modus fest, in dem der Dialog geöffnet wird:

0 = Modus zum Suchen eines Ordners

1 = Modus zum Suchen einer Datei

Navigation Zurück

Legt fest, ob die Schaltfläche "Zurück" bedient werden kann. Diese Schaltfläche ermöglicht eine einfache Navigation zu dem zuletzt besuchten Ordner.

Navigation Aufwärts

Legt fest, ob die Schaltfläche "Aufwärts" bedient werden kann. Diese Schaltfläche ermöglicht eine einfache Navigation zum übergeordneten Ordner.

Neuer Ordner

Legt fest, ob die Schaltfläche zum Anlegen eines neuen Ordners bedient werden kann.

Pfad

Enthält nach dem Beenden des Dialogs die Pfadangabe des zuletzt besuchten Ordners.

Status

Liefert eine Information, wie der Dialog beendet wurde

-1 = Dialog wurde abgebrochen

0 = Dialog wurde mit "OK" beendet.

ÖffneTaskManager

Beschreibung

Blendet den Task-Manager ein.

Mit dem Task-Manager können Sie am Bediengerät zu anderen geöffneten Anwendungen wechseln.

Hinweis

Das Aussehen des Task-Managers ist abhängig vom installierten Betriebssystem.

Verwendung in der Funktionsliste

ÖffneTaskManager

Verwendung in benutzerdefinierten Funktionen

OpenTaskManager

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Skripte unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

Tastatur

ÖffneBildschirmtastatur

Beschreibung

Blendet die Bildschirmtastatur ein oder aus.

Die Bildschirmtastatur bleibt so lange geöffnet, bis Sie die Bildschirmtastatur explizit schließen. Damit können Sie die Bildschirmtastatur auch in anderen Applikationen verwenden.

Verwendung in der Funktionsliste

ÖffneBildschirmtastatur (Darstellung)

Verwendung in benutzerdefinierten Funktionen

OpenScreenKeyboard Display_mode

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Darstellung

Legt fest, ob das Fenster mit der Bildschirmtastatur minimiert oder maximiert geöffnet wird:

0 (hmiScreenKeyboardMinimized) = Minimiert

1 (hmiScreenKeyboardMaximized) = Maximiert

Tastaturbedienung für Bildobjekte

BildobjektCursorAbwärts

Beschreibung

Führt zu einer zeilenweisen Cursorbewegung nach unten in der Tabelle des angegebenen Bildobjekts.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige
- NC-Teilprogramm-Anzeige
- PLC-Code-Anzeige

Verwendung in der Funktionsliste

BildobjektCursorAbwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

BildobjektCursorAufwärts

Beschreibung

Führt zu einer zeilenweisen Cursorbewegung nach oben in der Tabelle im angegebenen Bildobjekt.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige
- NC-Teilprogrammm-Anzeige
- PLC-Code-Anzeige

Verwendung in der Funktionsliste

BildobjektCursorAufwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

BildobjektCursorRechts

Beschreibung

Führt im angegebenen Bildobjekt zu einer zeilenweisen Cursorbewegung nach rechts.

Verwendung in der Funktionsliste

BildobjektCursorLinks (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem der Befehl ausgeführt wird.

BildobjektSeiteAbwärts

Beschreibung

Führt in der Tabelle des angegebenen Bildobjekts zu einer seitenweisen Cursorbewegung nach unten.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige
- NC-Teilprogramm-Anzeige

Verwendung in der Funktionsliste

BildobjektSeiteAbwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

BildobjektSeiteAufwärts

Beschreibung

Führt in der Tabelle des angegebenen Bildobjekts zu einer seitenweisen Cursorbewegung nach oben.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige
- NC-Teilprogramm-Anzeige

Verwendung in der Funktionsliste

BildobjektSeiteAufwärts (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

BildobjektCursorLinks

Beschreibung

Führt im angegebenen Bildobjekt zu einer zeilenweisen Cursorbewegung nach links.

Verwendung in der Funktionsliste

BildobjektCursorLinks (Bildobjekt)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Bildobjekt

Name des Bildobjekts, in dem der Befehl ausgeführt wird.

SeiteAbwärts

Beschreibung

Führt die Tastenfunktion <Bild Ab> am Bediengerät aus.

Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

Verwendung in der Funktionsliste

SeiteAbwärts

Verwendung in benutzerdefinierten Funktionen

PageDown

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

-

SeiteAufwärts

Beschreibung

Führt die Tastenfunktion <Bild Auf> am Bediengerät aus.

Diese Systemfunktion kann nur für Funktionstasten und für Aufgaben mit zeitlichen Trigger verwendet werden.

Verwendung in der Funktionsliste

SeiteAufwärts

Verwendung in benutzerdefinierten Funktionen

PageUp

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

-

GeheZumAnfang

Beschreibung

Führt die Tastenfunktion <Pos1> am Bediengerät aus.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät diese Funktionalität nicht standardmäßig besitzt.

Diese Systemfunktion ist nur für Funktionstasten verwendbar.

Verwendung in der Funktionsliste

GeheZumAnfang

Verwendung in benutzerdefinierten Funktionen

GoToHome

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

GeheZumEnde

Beschreibung

Führt die Tastenfunktion <Ende> am Bediengerät aus.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät diese Funktionalität nicht standardmäßig besitzt.

Diese Systemfunktion ist nur für Funktionstasten verwendbar.

Verwendung in der Funktionsliste

GeheZumEnde

Verwendung in benutzerdefinierten Funktionen

GoToEnd

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

--

9.11.1.3 Systemfunktionen

Andere Funktionen

StoppeRuntime

Beschreibung

Beendet die Runtime-Software und damit das laufende Projekt am Bediengerät.

Verwendung in der Funktionsliste

StoppeRuntime (Modus)

Verwendung in benutzerdefinierten Funktionen

StopRuntime (Mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob nach dem Beenden von Runtime auch das Betriebssystem heruntergefahren wird:

0 (hmiStopRuntime) = Runtime: Betriebssystem wird nicht heruntergefahren

1 (hmiStopRuntimeAndOperatingSystem) = Runtime und Betriebssystem: Betriebssystem wird heruntergefahren (bei WinCE nicht möglich)

Beispiel

Der folgende Programmcode fährt Runtime und Betriebssystem herunter.

```
{

//Stop runtime and shutdown
StopRuntime (hmiStopRuntimeAndOperationSystem);

}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

TextAbrufen

Beschreibung

Ermittelt den Listeneintrag aus einer Textliste. Das Ergebnis ist abhängig vom Wert und von der gewählten Runtime-Sprache. Das Ergebnis wird in eine Variable vom Datentyp "String" gespeichert.

Verwendung in der Funktionsliste

TextAbrufen (Ausgabetext, Index, Sprache, Textliste)

Verwendung in benutzerdefinierten Funktionen

LookupText (Output_text, Index, Language, Text_list)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Ausgabetext

Die Variable, in der das Ergebnis geschrieben wird.

Index

Die Variable, die den Wert des Listeneintrags festlegt.

Sprache

Legt fest, in welcher Runtime-Sprache der Listeneintrag ermittelt wird.

- Runtime-Sprache
  Sprachkürzel gemäß der VBScript-Referenz , z. B. "de-DE" für Deutsch (Deutschland) oder "en-US" für Englisch (Vereinigte Staaten von Amerika). Die Auswahl ist abhängig von den aktivierten Runtime-Sprachen.
- Variable
  Die Variable, welche die Sprache enthält. Die Runtime-Sprache geben Sie als Dezimalwert der Länder-Kennung an, z. B. 1031 für Deutsch - Standard, 1033 für Englisch - USA. Nähere Hinweise hierzu finden Sie in den Grundlagen von VBScript bei "Gebietsschema-ID (LCID)-Diagramm.
- Integer
  Die Zahl, die der Reihenfolge der Runtime-Sprachen bei Sprachumschaltung entspricht. Die Auswahl ist abhängig von den aktivierten Runtime-Sprachen, z. B. "0" für die Sprache, die beim ersten Runtime-Start erscheint. Nähere Hinweise hierzu finden Sie im Thema "Sprachen in Runtime".

Textliste

Legt die Textliste fest. Aus der Textliste wird der Listeneintrag ausgelesen.

Siehe auch

Geräteabhängigkeit (Seite 1536)

Benutzerverwaltung

ExportiereImportiereBenutzerverwaltung

Beschreibung

Exportiert alle Benutzer der Benutzerverwaltung des gerade aktiven Projekts in die angegebene Datei oder importiert die Benutzer aus der angegebenen Datei in das gerade aktive Projekt.

In der Benutzerverwaltung werden Benutzer, deren Passwörter und Rechte gespeichert.

Beim Import werden alle Benutzer überschrieben. Die importierten Benutzer sind sofort gültig.

Hinweis

Das Exportieren und Importieren der Benutzerverwaltung aus der V14 wird auf den Bediengeräten mit einer älteren Runtime-Version als V14 nicht unterstützt.

Verwendung in der Funktionsliste

ExportiereImportiereBenutzerverwaltung (Dateiname, Richtung)

Verwendung in benutzerdefinierten Funktionen

ExportImportUserAdministration (File_name, Direction)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Dateiname

Name der Datei, welche die Passwörter enthält oder in der die Passwörter geschrieben werden. Geben Sie den Ablageort und die Dateiendung (*.txt) mit an, z. B. "C:\TEMP \Passwords.txt".

Hinweis

Wenn Sie als Ablageort eine Storage Card verwenden, dann geben Sie den Ablageort wie folgt an: "\StorageCard\<Dateiname>".

Richtung

Legt fest, ob Passwörter exportiert oder importiert werden:

0 (hmiExport) = Export: Passwörter werden exportiert.

1 (hmiImport) = Import: Passwörter werden importiert.

ZeigeAnmeldedialog

Beschreibung

Öffnet einen Dialog am Bediengerät, mit dem sich der Benutzer am Bediengerät anmelden kann.

Verwendung in der Funktionsliste

ZeigeAnmeldedialog

Verwendung in benutzerdefinierten Funktionen

ShowLogonDialog

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)"

Parameter

--

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

Berechnung

ErhöheVariable

Beschreibung

Addiert den angegebenen Wert zum Wert der Variablen.

X = X + a

Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Wertes verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen können Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zuweisen.

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Variablenwert in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

Verwendung in der Funktionsliste

ErhöheVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

IncreaseTag (Tag, Value)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variable

Die Variable, zu welcher der angegebene Wert addiert wird.

Wert

Der Wert, der addiert wird.

Beispiel

Der folgende Programmcode erhöht den Wert der Variablen varX um den Wert in der Variable value. Der eingegebene Wert wird in der Variable old_value gesichert und zusammen mit dem neuen Wert varX ausgegeben.

```
{
BYTE varX;
BYTE value;

//user input
...
BYTE old_value = varX;

//Increase tag
IncreaseTag(varX, value);

//print original value and function result
printf ("User input: %i\r\n, Result of function IncreaseTag: %i\r\n", old_value, varX);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

InvertiereLineareSkalierung

Beschreibung

Weist der Variablen X einen Wert zu, der aus dem Wert der angegebenen Variablen Y über die lineare Funktion X = (Y - b) / a berechnet wird.

Die Variablen X und Y dürfen nicht identisch sein. Diese Systemfunktion ist die Umkehrfunktion zur Systemfunktion "LineareSkalierung".

Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

Verwendung in der Funktionsliste

InvertiereLineareSkalierung (X, Y, b, a)

Verwendung in benutzerdefinierten Funktionen

InvertLinearScaling (X, Y, b, a)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

X

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

Y

Die Variable, deren Wert zur Berechnung verwendet wird.

b

Der Wert, der subtrahiert wird.

a

Der Wert, durch den dividiert wird.

Beispiel

Der folgende Programmcode weist der Variable varX einen Wert mit der Funktion InverseLinearScaling zu.

```
{
BYTE varX;
BYTE Yvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

//Inverse linear scaling
 InverseLinearScaling (varX, Yvalue, bvalue, avalue);

printf ("varX = %d\r\n, varX);
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

LineareSkalierung

Beschreibung

Weist der Variablen Y einen Wert zu, der aus dem Wert der angegebenen Variablen X über die lineare Funktion Y = (a * X) + b berechnet wird.

Die Umkehrfunktion ist die Systemfunktion "InvertiereLineareSkalierung".

Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

Verwendung in der Funktionsliste

LineareSkalierung (Y, a, X, b)

Verwendung in benutzerdefinierten Funktionen

LinearScaling (Y, a, X, b)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Y

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

a

Der Wert, mit dem multipliziert wird.

X

Die Variable, deren Wert zur Berechnung verwendet wird.

b

Der Wert, der addiert wird.

Beispiel

Der folgende Programmcode weist mit der Funktion LinearScaling der Variable Yvar einen Wert zu.

```
{
BYTE Yvar;
BYTE Xvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

// linear scaling
 LinearScaling ( Yvar, avalue, Xvalue, bvalue);

printf ("Yvar = %d\r\n, Yvar);
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariable

Beschreibung

Weist der angegebenen Variablen einen neuen Wert zu.

Hinweis

Abhängig vom Variablentyp können Sie mit dieser Systemfunktion Zeichenketten und Zahlen zuweisen.

Verwendung in der Funktionsliste

SetzeVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

SetTag (Tag, Value)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, welcher der angegebene Wert zugewiesen wird.

Wert

Der Wert, welcher der angegebenen Variablen zugewiesen wird.

Hinweis

Die Systemfunktion "SetzeVariable" wird nur nach Aufbau einer Verbindung ausgeführt.

Beispiel

Der folgende Programmcode setzt mit der Funktion SetTag den Wert der Variablen gs_tag_bit auf TRUE und speichert den Rückgabewert in der Variable ok.

```
{
BOOL ok;
BOOL bvalue;

//Set the tag to true
ok = SetTag("gs_tag_bit", TRUE);
//error handling
if(ok)
{
  // succeeded
  printf ( "Function has run through.\r\n" );
bvalue = GetTagBit("gs_tag_bit");
  printf ("Value of gs_tag_bit: %d\r\n", bvalue);
}
else
{
  // failed
  printf ( "Error - function failed." );
}
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableDurchEigenschaft

Beschreibung

Legt einen Variablenwert durch den Wert einer Objekteigenschaft fest. Die Änderung wird zusätzlich im Meldesystem protokolliert

Verwendung in der Funktionsliste

SetzteVariableDurchEigenschaft (Variablenname, Bildname, Bildobjekt, Name der Eigenschaft, Mit oder ohne Bedienmeldung)

Verwendung in benutzerdefinierten Funktionen

SetTagByProperty (Tag_name, Screen_name, Screen_object, Property_name, With_or_without_operator_event)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variablenname

Name der Variablen, deren Wert durch die Objekteigenschaft festgelegt wird.

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Bildobjekt

Name des Objekts, dessen Eigenschaft den Variablenwert liefert.

Name der Eigenschaft

Name der Eigenschaft, die den Variablenwert liefert.

Mit oder ohne Bedienmeldung

0 (hmiWithoutOperatorEvent) = Ohne Bedienmeldung

1 (hmiWithOperatorEvent) = Mit Bedienmeldung

Beispiel

Der folgende Programmcode gibt beim Klick auf ein Kombinationsfeld den Wert des ausgewählten Textes zurück.

```
{
char* rt_value;

SetTagByProperty (rt_value, screenName, objectName, "SelectedText",
hmiWithoutOperatorEvent);

...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableIndirekt

Beschreibung

Schreibt einen Wert in eine indirekt adressierte Variable. Die als Ausgabe-Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert durch die Funktion geändert wird.Über eine Bedienmeldung kann die Änderung im Meldesystem protokolliert werden.

Verwendung in der Funktionsliste

SetzeVariableIndirekt (Variablenname, Wert, Mit oder ohne Bedienmeldung)

Verwendung in benutzerdefinierten Funktionen

SetTagIndirect (Tag_name, Value, With_or_without_operator_event)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variablenname

Name einer String-Variablen, die den Namen einer Variablen enthält, deren Wert geändert werden soll.

Wert

Wert, der geschrieben werden soll.

Mit oder ohne Bedienmeldung

0 (hmiWithoutOperatorEvent) = Ohne Bedienmeldung

1 (hmiWithOperatorEvent) = Mit Bedienmeldung

Beispiel

Der folgende Programmcode schreibt den Wert "value" in die Variable Tag3.

```
{
int value;

SetTag ("IndirectWrite", "Tag3");
SetTagIndirect ("IndirectWrite", "value", hmiWithoutOperatorEvent);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableIndirektDurchEigenschaft

Beschreibung

Schreibt den Wert einer Objekteigenschaft in eine indirekt adressierte Variable. Die als Ausgabe-Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert durch die Funktion geändert wird. Über eine Bedienmeldung kann die Änderung im Meldesystem protokolliert werden.

Verwendung in der Funktionsliste

SetzeVariableIndirektDurchEigenschaft (Variablenname, Bildname, Bildobjekt, Name der Eigenschaft, Mit oder ohne Bedienmeldung)

Verwendung in benutzerdefinierten Funktionen

SetTagIndirectByProperty (Tag_name, Screen_name, Screen_object, Property_name, With_or_without_operator_event)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variablenname

Name einer String-Variablen, die den Namen einer Variablen enthält, deren Wert geändert werden soll.

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Bildobjekt

Name des Objekts, dessen Eigenschaft den Wert liefert.

Name der Eigenschaft

Name der Eigenschaft, die den Wert liefert.

Mit oder ohne Bedienmeldung

0 (hmiWithoutOperatorEvent) = Ohne Bedienmeldung

1 (hmiWithOperatorEvent) = Mit Bedienmeldung

Beispiel

Der folgende Programmcode schreibt den Wert der Objekteigenschaft "Hintergrundfarbe" in die Variable Tag2.

```
{
SetTag ("IndirectWrite", "Tag2");
SetTagIndirectByProperty ("IndirectWrite", screenName, objectName, "BackColor",
hmiWithoutOperatorEvent);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableIndirektDurchVariableIndirekt

Beschreibung

Schreibt den Wert einer indirekt adressierten Variablen in eine indirekt adressierte Variable. Die als Ausgabe-Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert durch die Funktion geändert wird. Die als Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert gelesen wird. Über eine Bedienmeldung kann die Änderung im Meldesystem protokolliert werden.

Verwendung in der Funktionsliste

SetzeVariableIndirektDurchVariableIndirekt (Variablenname, Variablenname, Mit oder ohne Bedienmeldung)

Verwendung in benutzerdefinierten Funktionen

SetTagIndirectByTagIndirect Tag_name, Source_tag_name, With_or_without_operator_event

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variablenname

Name einer String-Variablen, die den Namen einer Variablen enthält, deren Wert geändert werden soll.

Variablenname

Name der indirekten Variablen, die den Variablenwert liefert.

Mit oder ohne Bedienmeldung

0 (hmiWithoutOperatorEvent) = Ohne Bedienmeldung

1 (hmiWithOperatorEvent) = Mit Bedienmeldung

Beispiel

Der folgende Programmcode schreibt den Wert der Variable "Tag4" in die Variable "Tag2".

```
{
SetTag ("IndirectWrite", "Tag2");
SetTag ("IndirectRead", "Tag4");
SetTagIndirectByTagIndirect ("IndirectWrite", "IndirectRead");
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableDurchVariableIndirekt

Beschreibung

Schreibt den Wert einer indirekt adressierten Variable in eine Variable. Die als Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert gelesen wird. Über eine Bedienmeldung kann die Änderung im Meldesystem protokolliert werden.

Verwendung in der Funktionsliste

SetzeVariableDurchVariableIndirekt (Name der Variablen, Variablenname, Mit oder ohne Bedienmeldung)

Verwendung in benutzerdefinierten Funktionen

SetTagByTagIndirect Tag_name, Source_tag_name, With_or_without_operator_event

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Name der Variablen

Name der Variable, deren Wert geändert werden soll.

Variablenname

Name einer String-Variablen, die den Namen einer Variablen enthält, die den Variablenwert liefert.

Mit oder ohne Bedienmeldung

0 (hmiWithoutOperatorEvent) = Ohne Bedienmeldung

1 (hmiWithOperatorEvent) = Mit Bedienmeldung

Beispiel

Der folgende Programmcode schreibt Wert der Variable Tag4 in die Variable Tag1.

```
{
SetTag ("IndirectRead", "Tag4");
SetTagByTagIndirect ("Tag1", "IndirectRead", hmiWithoutOperatorEvent);

...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeVariableMitBedienmeldung

Beschreibung

Legt den Wert für eine Variable fest. Die Änderung wird zusätzlich im Meldesystem protokolliert.

Verwendung in der Funktionsliste

SetzeVariableMitBedienmeldung (Name der Variablen, Wert)

Verwendung in benutzerdefinierten Funktionen

SetTagWithOperatorEvent Tag_name, Value

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Name der Variablen

Name der Variablen, deren Wert festgelegt wird.

Wert

Der Wert, der in die Variable geschrieben wird.

Beispiel

Der folgende Programmcode überträgt beim Klick auf eine Taste den Wert der Variable "value" auf die Variable "result" .

```
'Programming language: VBS
SetTagWithOperatorEvent result, value
...
```

```
//Programming language: C
{
SetTagWithOperatorEvent ("result", "value");
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

VerringereVariable

Beschreibung

Subtrahiert den angegebenen Wert vom Wert der Variablen.

X = X - a

Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Wertes verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen können Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zuweisen.

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Variablenwert in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

Verwendung in der Funktionsliste

VerringereVariable (Variable, Wert)

Verwendung in benutzerdefinierten Funktionen

DecreaseTag (Tag, Value)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Variable

Die Variable, von welcher der angegebene Wert subtrahiert wird.

Wert

Der Wert, der subtrahiert wird.

Beispiel

Der folgende Programmcode erniedrigt den Wert der Variablen varX um den Wert in der Variable value. Der eingegebene Wert wird in der Variable old_value gesichert und zusammen mit dem neuen Wert varX ausgegeben.

```
{
BYTE varX;
BYTE value;

//user input
...
BYTE old_value = varX;

//Decrease tag
DecreaseTag(varX, value);

//print original value and function result
printf ("User input: %i\r\n, Result of function DecreaseTag: %i\r\n", old_value, varX);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

Bilder

AktiviereBild

Beschreibung

Führt einen Bildwechsel zu dem angegebenen Bild aus.

Um vom Grundbild in den Permanentbereich oder umgekehrt zu wechseln, verwenden Sie die Systemfunktion "AktiviereBildMitNummer".

Verwendung in der Funktionsliste

AktiviereBild (Bildname, Objektnummer)

Verwendung in benutzerdefinierten Funktionen

ActivateScreen (Screen_name, Object_number)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit ".

Parameter

Bildname

Name des Bildes, zu dem gewechselt wird.

Objektnummer

Das Bedienelement, das nach dem Bildwechsel im angegebenen Bild den Fokus erhält. Die Nummer des Bedienelements legen Sie während der Projektierung durch die Tab-Reihenfolge fest.

Wenn Sie "0" angeben:

- Ist der Fokus beim Aufruf der Systemfunktion im Permanentbereich, behält der Permanentbereich den Fokus.
- Ist der Fokus beim Aufruf der Systemfunktion im Grundbild, erhält das erste Bedienelement im angegebenen Bild den Fokus.

Hinweis

Wenn die Systemfunktion "AktiviereBild" an das Ereignis "Rand erreicht" projektiert wird, so ist für den Parameter "Objektnummer" nur der Wert 0 zulässig. Das aktive Objekt wird nicht anhand der Objektnummer festgelegt, sondern anhand der X-Position des aktiven Objekts vor dem Bildwechsel.

Beispiel

Der folgende Programmcode aktiviert mit der Funktion ActivateScreen das Bild "Screen_2", z. B. beim Klicken auf eine Schaltfläche.

```
Sub ActivateScreen_2()

'Userdefined code
'' i. e. when pressing a button

ActivateScreen "Screen_2",0
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

AktiviereBildInAktuellemBildfenster

Beschreibung

Führt im aktuellen Bildfenster einen Bildwechsel zu dem angegebenen Bild aus.

Verwendung in der Funktionsliste

AktiviereBildInAktuellemBildfenster (Neuer Bildname)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Neuer Bildname

Name des Bildes, das im Bildfenster angezeigt wird.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

AktiviereBildInBildfenster

Beschreibung

Führt im angegebenen Bildfenster einen Bildwechsel zu dem angegebenen Bild aus.

Verwendung in der Funktionsliste

AktiviereBildInBildfenster (Bildname, Bildfenster, Neuer Bildname)

Verwendung in benutzerdefinierten Funktionen

ActivateScreenInScreenWindow (Screen_name, Screen_window, New_screen_name)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Parameter

Bildname

Name des Bildes, das im Bildfenster angezeigt wird.

Bildfenster

Name des Bildfensters, in dem das neue Bild angezeigt wird.

Neuer Bildname

Name des neuen Bildes, das im Bildfenster angezeigt wird.

Beispiel

Der folgende Programmcode aktiviert mit der Funktion ActivateScreenInScreenWindow beim Klicken auf eine Taste das Bild "Screen_2".

```
{
// User defined code
// i.e. when pressing a button
ActivateScreenInScreenWindow (GetParentScreen(screenName),
GetParentScreenWindow(screenName), "Screen_2");
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

LeseÜbergeordnetesBild

Beschreibung

Die Funktion ermittelt aus einem übergebenen Bildpfad den Namen des übergeordneten Bildes.

Verwendung in der Funktionsliste

LeseÜbergeordnetesBild (Bild)

Verwendung in benutzerdefinierten Funktionen

GetParentScreen Screen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit ".

Parameter

Bild

Der übergebene Parameter Bild muss den Aufbau haben, wie ihn das Grafiksystem für Bildpfade bildet:

<Bildname>.<Bildfenstername>:<Bildname>.<Bildfenstername>:<Bildname>...

Wenn Sie in der Funktionsliste zur Festlegung des Parameters die Objektliste verwenden, wird nicht der Bildpfad, sondern nur der Bildname eingetragen.

Hinweis

Das Zeichen "." dient der Unterscheidung von Bild- und Bildobjektnamen. Das Zeichen ":" dient der hierarchischen Strukturierung. Verwenden Sie als Trennzeichen in Namensbezeichnungen daher ausschließlich die Zeichen "-" oder "_".

Rückgabewert

Namen des übergeordneten Bildes.

Prinzip

In einem Bild "Screen_1" liegt ein Bildfenster "Screenwindow_1". Im Bildfenster wird das Bild "Screen_2" mit dem Bildfenster "Screenwindow_2" usw.

Bildpfad: Screen_1.Screenwindow_1:Screen_2.Screenwindow_2:Screen_3

GetParentScreen liefert:

- "Screen_2", wenn die Systemfunktion im Bild "Screen_3" aufgerufen wird
- "Screen_1", wenn die Systemfunktion im Bild "Screen_2" aufgerufen wird

GetParentScreenwindow liefert:

- "Screenwindow_2", wenn die Systemfunktion im Bild "Screen_3" aufgerufen wird
- "Screenwindow_1", wenn die Systemfunktion im Bild "Screen_2" aufgerufen wird

LeseÜbergeordnetesBildfenster

Beschreibung

Die Funktion ermittelt aus einem übergebenen Bildpfad den Namen des übergeordneten Bildfensters.

Verwendung in der Funktionsliste

LeseÜbergeordnetesBildfenster (Bild)

Verwendung in benutzerdefinierten Funktionen

GetParentScreenWindow Screen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit ".

Parameter

Bild

Der übergebene Parameter Bild muss den Aufbau haben, wie ihn das Grafiksystem für Bildpfade bildet:

<Bildname>.<Bildfenstername>:<Bildname>.<Bildfenstername>:<Bildname>...

Wenn Sie in der Funktionsliste zur Festlegung des Parameters die Objektliste verwenden, wird nicht der Bildpfad, sondern nur der Bildname eingetragen.

Hinweis

Das Zeichen "." dient der Unterscheidung von Bild- und Bildobjektnamen. Das Zeichen ":" dient der hierarchischen Strukturierung. Verwenden Sie als Trennzeichen in Namensbezeichnungen daher ausschließlich die Zeichen "-" oder "_".

Rückgabewert

Namen des übergeordneten Bildfensters.

Prinzip

In einem Bild "Screen_1" liegt ein Bildfenster "Screenwindow_1". Im Bildfenster wird das Bild "Screen_2" mit dem Bildfenster "Screenwindow_2" usw.

Bildpfad: Screen_1.Screenwindow_1:Screen_2.Screenwindow_2:Screen_3

GetParentScreen liefert:

- "Screen_2", wenn die Systemfunktion im Bild "Screen_3" aufgerufen wird
- "Screen_1", wenn die Systemfunktion im Bild "Screen_2" aufgerufen wird

GetParentScreenwindow liefert:

- "Screenwindow_2", wenn die Systemfunktion im Bild "Screen_3" aufgerufen wird
- "Screenwindow_1", wenn die Systemfunktion im Bild "Screen_2" aufgerufen wird

SetzeEigenschaftDurchEigenschaft

Beschreibung

Legt den Wert einer Objekteigenschaft durch eine andere Objekteigenschaft fest.

Verwendung in der Funktionsliste

SetzeEigenschaftDurchEigenschaft (Bildname, Objekt, Name der Eigenschaft, Ziel-Bildname, Ziel-Bildobjekt, Ziel Eigenschaftsname)

Verwendung in benutzerdefinierten Funktionen

SetPropertyByProperty Screen_name, Screen_object, Property_name, Source_screen_name, Source_screen_object, Source_property_name

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Wenn Sie die Eigenschaft eines Bildes durch eine andere Bildeigenschaft festlegen wollen, müssen die Parameter "Objekt" und "Ziel Objekt" leer sein. Verwenden Sie dazu z. B. folgende Syntax:

SetPropertyByProperty ("<Bildname>","","<Eigenschaftsname>","<ZIel_Bildname>","","<Ziel_Eigenschaftsname>")

Parameter

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Objekt

Name des Objekts, von dem ein Eigenschaftswert an das Zielobjekt übertragen wird.

Name der Eigenschaft

Name der Eigenschaft, deren Wert an das Zielobjekt übertragen wird.

Ziel-Bildname

Name des Bildes, in dem das Zielobjekt enthalten ist.

Ziel-Bildobjekt

Name des Zielobjekts, bei dem eine Eigenschaft geändert wird.

Ziel Eigenschaftsname

Name der Eigenschaft, die geändert wird.

Beispiel

Der folgende Programmcode überträgt mit der Funktion SetPropertyByProperty die Eigenschaft "ToolbarButtonClick" des Objekts "Control_1" im Ausgangsbild "Trend_1" : auf die entsprechende Eigenschaft im Zielbild "Trend_2".

```
'Programming language: VBS
'Name of source picture: Trend_1
'Name of target picture: Trend_2
'Name of the f(t) trend view control: Control_1

SetPropertyByProperty "Trend_1", "Control_1", "ToolbarButtonClick", "Trend_2",
"Control_2", "ToolbarButtonClick"

'User defined code
...
```

```
{
//Programming language: C
//Name of source picture: Trend_1
//Name of target picture: Trend_2
//Name of the f(t) trend view control: Control_1

SetPropertyByProperty ("Trend_1", "Control_1", "ToolbarButtonClick", "Trend_2",
"Control_2", "ToolbarButtonClick");

// User defined code
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftDurchKonstante

Beschreibung

Legt den Wert einer Objekteigenschaft als Zeichenkette fest.

Verwendung in der Funktionsliste

SetzeEigenschaftDurchKonstante (Bildname, Bildobjekt, Name der Eigenschaft, Wert)

Verwendung in benutzerdefinierten Funktionen

SetPropertyByConstant

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Wenn Sie die Eigenschaft eines Bildes ändern wollen, muss der Parameter "Objekt" leer sein. Verwenden Sie dazu z. B. folgende Syntax:

SetPropertyByConstant ("<Bildname>","","<Eigenschaftsname>","<Wert>")

Parameter

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Bildobjekt

Name des Objekts, dessen Eigenschaft geändert wird.

Name der Eigenschaft

Name der Eigenschaft, die geändert wird.

Wert

Der Wert, welcher der Eigenschaft zugewiesen wird.

Beispiel

Der folgende Programmcode ändert mit der Funktion SetPropertyByConstant die Eigenschaft eines Objekts: Im Bild "Trends" wird die Eigenschaft "ToolbarButtonClick" des darin enthaltenen Objekts "Control_1" auf den Wert 26 gesetzt.

```
'Programming language: VBS
'Name of the picture: Trends
'Name of the f(t) trend view control: Control_1

SetPropertyByConstant "Trends", "Control_1", "ToolbarButtonClick", "26"

'User defined code
...
```

```
{
//Programming language: C
//Name of the picture: Trends
//Name of the f(t) trend view control: Control_1

SetPropertyByConstant ("Trends", "Control_1", "ToolbarButtonClick", "26");

// User defined code
...
}
```

Beispiel: Bildeigenschaft ändern

Der folgende Programmcode ändert mit der Funktion SetPropertyByConstant die Eigenschaft eines Bilds: Im Bild "Trends" wird die Eigenschaft "Backcolor" auf den Wert 255 gesetzt.

```
'Programming language: VBS
'Name of the picture: Trends

SetPropertyByConstant "Trends", "Trends", "Backcolor", "255"

'User defined code
...
```

```
{
//Programming language: C
//Name of the picture: Trends

SetPropertyByConstant ("Trends", "Trends", "Backcolor", "255");

// User defined code
...
}
```

Alternativ verwenden Sie anstelle des zweiten Parameters (Objekt) das Kennwort NULL oder einen Leerzeichen-String.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftDurchVariable

Beschreibung

Legt den Wert einer Objekteigenschaft durch einen Variablenwert fest.

Verwendung in der Funktionsliste

SetzeEigenschaftDurchVariable (Bildname, Bildobjekt, Name der Eigenschaft, Variablenname)

Verwendung in benutzerdefinierten Funktionen

SetPropertyByTag (Screen_name, Screen_object, Property_name, Tag_name)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Wenn Sie die Eigenschaft eines Bildes durch einen Variablenwert festlegen wollen, muss der Parameter "Objekt" leer sein. Verwenden Sie dazu z. B. folgende Syntax:

SetPropertyByTag ("<Bildname>","","<Eigenschaftsname>","<Variablenname>")

Parameter

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Bildobjekt

Name des Objekts, dessen Eigenschaft durch den Variablenwert gesetzt wird.

Name der Eigenschaft

Name der Eigenschaft, die durch den Variablenwert gesetzt wird.

Variablenname

Name der Variablen, die den Wert der Eigenschaft enthält.

Beispiel

Der folgende Programmcode ändert mit der Funktion SetPropertyByTag die Eigenschaft eines Objekts: Beim Klicken auf das Objekt werden der Objektname und das Bild, in dem sich das Objekt befindet, übergeben. Die Überschrift CaptionText im Bildfenster erhält den Wert der Variable HMI_value_1.

```
'Programming language: VBS
SetPropertyByTag screenName, objectName, "CaptionText", "HMI_value_1"

'User defined code
...
```

```
{
//Programming language: C
SetPropertyByTag (screenName, objectName, "CaptionText", "HMI_value_1");

// User defined code
...
}
```

Beispiel

Der folgende Programmcode ändert mit der Funktion SetPropertyByTag die Eigenschaft eines Objekts: Im Bild "Trends" wird die Eigenschaft "ToolbarButtonClick" des darin enthaltenen Objekts "Control_1" auf den Wert 26 gesetzt.

```
'Programming language: VBS
'Name of the picture: Trends
'Name of the f(t) trend view control: Control_1

SetPropertyByConstant "Trends", "Control_1", "ToolbarButtonClick", "26"

'User defined code
...
```

```
{
//Programming language: C
//Name of the picture: Trends
//Name of the f(t) trend view control: Control_1

SetPropertyByConstant ("Trends", "Control_1", "ToolbarButtonClick", "26");

// User defined code
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftDurchVariableIndirekt

Beschreibung

Schreibt den Wert einer einer indirekt adressierten Variable in eine Objekteigenschaft fest.Die als Parameter übergebene Variable enthält den Namen einer zweiten Variable, deren Wert gelesen wird.

Verwendung in der Funktionsliste

SetzeEigenschaftDurchVariableIndirekt (Bildname, Bildobjekt, Name der Eigenschaft, Variablenname)

Verwendung in benutzerdefinierten Funktionen

SetPropertyByTagIndirect (Screen_name, Screen_object, Property_name, Tag_name)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional (Seite 1567)".

Wenn Sie die Eigenschaft eines Bildes indirekt durch eine Variable festlegen wollen, muss der Parameter "Objekt" leer sein. Verwenden Sie dazu z. B. folgende Syntax:

SetPropertyByTagIndirect ("<Bildname>","","<Eigenschaftsname>","<Variablenname>")

Parameter

Bildname

Name des Bildes, in dem das Objekt enthalten ist.

Bildobjekt

Name des Objekts, dessen Eigenschaft geändert werden soll.

Name der Eigenschaft

Name der Eigenschaft, die geändert werden soll.

Variablenname

Name der Variablen, die wiederum den Namen der Variablen enthält, deren Wert gelesen wird.

Beispiel

Der folgende Programmcode ändert mit der Funktion SetPropertyByTagIndirect die Eigenschaft eines Objekts: .

```
'Programming language: VBS
SetPropertyByTagIndirect GetParentScreen(screenName), GetParentScreenWindow(screenName),
"ScreenName", "HMI_value_1"

'User defined code
...
```

```
{
//Programming language: C
SetPropertyByTagIndirect (GetParentScreen(screenName), GetParentScreenWindow(screenName),
"ScreenName", "HMI_value_1");

// User defined code
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftAktuellesFensterDurchEigenschaft

Beschreibung

Legt im aktuellen Fenster den Wert einer Objekteigenschaft durch eine andere Objekteigenschaft fest..

Verwendung in der Funktionsliste

SetzeEigenschaftAktuellesFensterDurchEigenschaft (Name der Eigenschaft, Bildname, Bildobjekt, Name der Eigenschaft)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Name der Eigenschaft

Name der Eigenschaft, dessen Eigenschaft an das Zielobjekt übertragen werden soll.

Bildname

Name des Bildes, in dem das Bildobjekt enthalten ist.

Bildobjekt

Name des Bildobjekts, an das die Eigenschaft übertragen wird.

Name der Eigenschaft

Name der Eigenschaft, die geändert wird.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftAktuellesFenster

Beschreibung

Legt im aktuellen Fenster den Wert einer Objekteigenschaft durch einen Variablenwert fest.

Verwendung in der Funktionsliste

SetzeEigenschaftAktuellesFenster (Name der Eigenschaft, Wert)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Name der Eigenschaft

Name der Eigenschaft, die durch den Variablenwert gesetzt wird.

Wert

Name der Variablen, die den Wert der Eigenschaft enthält.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeEigenschaftAktuellesFensterDurchVariableIndirekt

Beschreibung

Legt im aktuellen Fenster den Wert einer Objekteigenschaft durch eine Variable fest. Die Variable enthält den Variablennamen, der die Objekteigenschaft festlegt.

Verwendung in der Funktionsliste

SetzeEigenschaftAktuellesFensterDurchVariableIndirekt (Name Zieleigenschaft, Variablenname)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Name Zieleigenschaft

Name der Eigenschaft, die durch den Variablenwert gesetzt wird.

Variablenname

Name der Variablen, die wiederum den Namen der Variablen enthält, welche die Objekteigenschaft festlegt.

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

Bitbearbeitung

InvertiereBit

Beschreibung

Invertiert den Wert der angegebenen Variable vom Typ "Bool":

- Hat die Variable den Wert 1 (TRUE), wird sie auf 0 (FALSE) gesetzt.
- Hat die Variable den Wert 0 (FALSE), wird sie auf 1 (TRUE) gesetzt.

Verwendung in der Funktionsliste

InvertiereBit (Variable)

Verwendung in benutzerdefinierten Funktionen

InvertBit (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, dessen Bit gesetzt wird.

Beispiel

Der folgende Programmcode invertiert den Wert der boolschen Variable bStatus und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Get current value
bValue=myTag.Value

'Save current value
bSaved=bValue

'Invert Bit
InvertBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 0;
BOOL bSaved = bStatus;

//Invert variable
invertBit(bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
  ...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

InvertiereBitInVariable

Beschreibung

Invertiert ein Bit in der angegebenen Variablen:

- Hat das Bit in der Variablen den Wert 1 (TRUE), wird es auf 0 (FALSE) gesetzt.
- Hat das Bit in der Variablen den Wert 0 (FALSE), wird es auf 1 (TRUE) gesetzt.

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "InvertiereBit".

Verwendung in der Funktionsliste

InvertiereBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

InvertBitInTag (Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der das angegebene Bit gesetzt wird.

Bit

Die Nummer des Bits, das gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in einer Variablen von rechts nach links gezählt. Die Zählung beginnt mit 0.

Beispiel

Der folgende Programmcode invertiert ein Bit an der angegebenen Position bitposition in der Variable bStatusWord und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Get current value
bValue=myTag.Value

'Save current value
bSaved=bValue

'Invert Bit in position
bitposition=2
InvertBit(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bsaved = bStatusWord;
BYTE bitposition = 2;

//Invert bit in bitposition
InvertBitInTag (bStatusWord, bitposition);
//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bsaved);
  ...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

RücksetzeBit

Beschreibung

Setzt den Wert einer Variablen vom Typ "Bool" auf 0 (FALSE).

Verwendung in der Funktionsliste

RücksetzeBit (Variable)

Verwendung in benutzerdefinierten Funktionen

ResetBit (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable vom Typ BOOL, die auf 0 (FALSE) gesetzt wird.

Beispiel

Der folgende Programmcode setzt den Wert der boolschen Variable bStatus mit der Funktion ResetBit auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Set value
bValue=1
myTag.Value=bValue

'Save current value
bSaved=bValue

'Reset Bit
ResetBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 1;
BOOL bSaved = bStatus;

//Reset bit
  ResetBit (bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

RücksetzeBitInVariable

Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 0 (FALSE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie stattdessen die Systemfunktion "RücksetzeBit".

Verwendung in der Funktionsliste

RücksetzeBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

ResetBitInTag (Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der ein Bit auf 0 (FALSE) gesetzt wird.

Bit

Die Nummer des Bits, das auf 0 (FALSE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position bitposition der Variablen bStatusWord auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Save current value
bValue=myTag.Value
bSaved=bValue

'Reset Bit
bitposition=2
ResetBit(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bSaved = bStatusWord;
BYTE bitposition = 2;

//Reset bit in bitposition
ResetBitInTag (bStatusWord, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bSaved);
  ...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeBit

Beschreibung

Setzt den Wert einer Variablen vom Typ "Bool" auf 1 (TRUE).

Verwendung in der Funktionsliste

SetzeBit (Variable)

Verwendung in benutzerdefinierten Funktionen

SetBit (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable vom Typ BOOL, die auf 1 (TRUE) gesetzt wird.

Beispiel

Der folgende Programmcode setzt den Wert der boolschen Variable bStatus mit der Funktion SetBit auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, strResult

Set myTag = SmartTags("bStatus")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Set value
bValue=0
myTag.Value=bValue

'Save current value
bSaved=bValue

'Set Bit
SetBit(myTag)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BOOL bStatus = 0;
BOOL bSaved = bStatus;

//Set bit
  SetBit (bStatus);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatus, bSaved);
...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

SetzeBitInVariable

Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 1 (TRUE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "SetzeBit".

Verwendung in der Funktionsliste

SetzeBitInVariable (Variable, Bit)

Verwendung in benutzerdefinierten Funktionen

SetBitInTag(Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Variable

Die Variable, in der ein Bit auf 1 (TRUE) gesetzt wird.

Bit

Die Nummer des Bits, das auf 1 (TRUE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

Hinweis

Eine notwendige Bedingung für eine zuverlässige Funktionalität ist eine garantierte Aktualisierung der verwendeten Variablen mit den aktuellen Prozesswerten. Projektieren Sie die Variable deswegen in einem E/A-Feld oder die Systemfunktion an einem Bildobjekt z. B. einer Schaltfläche.

Wenn Sie die Systemfunktion an ein kurzzeitiges Ereignis z. B. Kommen einer Meldung projektiert haben, dann erreichen Sie aktuelle Prozesswerte nur, wenn Sie die Variable auf dauerhaft lesen setzen.

Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position bitposition der Variablen bStatusWord auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bSaved aus.

```
'Programming language: VB
Dim myTag
Dim myOutputField
Dim bValue, bSaved, bitposition, strResult

Set myTag = SmartTags("bStatusWord")
Set myOutputField=HMIRuntime.Screens("MyScreen").ScreenItems("objTextField")

'Save current value
bValue=myTag.Value
bSaved=bValue

'Set Bit in tag
bitposition=2
SetBitInTag(myTag, bitposition)
bValue=myTag.Value

'Output result old and new value:
strResult="Old Value: "&bSaved &Chr(13)&"New Value: "&bValue
myOutputField.Text=strResult
```

```
//Programming language: C
{
BYTE bStatusWord;
BYTE bSaved = bStatusWord;
BYTE bitposition = 2;

//Reset bit in bitposition
SetBitInTag (bStatusWord, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bStatusWord, bSaved);
  ...
}
```

Siehe auch

Systemfunktionen für Runtime Professional (Seite 1567)

Einstellungen

LesePLCModus

Beschreibung

Wertet den aktuellen Zustand der verbundenen Steuerung aus.

Die Systemfunktion "LesePLCModus" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7-1200
- SIMATIC S7-1500

Verwendung in der Funktionsliste

LesePLCModus (Verbindung, Modus)

Verwendung in benutzerdefinierten Funktionen

-

Parameter

Verbindung

Verbindung von Steuerung und Bediengerät.

Modus

Wertet den Zustand der verbundenen Steuerung aus.
8: RUN = Das Steuerungsprogamm wird abgearbeitet.
4: STOP = Das Steuerungsprogramm ist unterbrochen.
Wählen Sie für die Auswertung eine geeignete Anwendervariable aus.

SetzeSprache

Beschreibung

Schaltet die Sprache am Bediengerät um. Alle projektierten Texte und Systemmeldungen werden am Bediengerät in der neu eingestellten Sprache dargestellt.

Verwendung in der Funktionsliste

SetzeSprache (Sprache)

Verwendung in benutzerdefinierten Funktionen

SetLanguage (Language)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Sprache

Legt fest, welche Sprache am Bediengerät eingestellt wird. Es sind folgende Angaben möglich

- -1 (hmiToggle) = Umschalten: Wechselt zur nächsten Sprache. Die Reihenfolge legen Sie während der Projektierung im Editor "Projektsprachen" fest.
- Nummer, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben. Wechselt zu der Sprache mit der angegebenen Nummer.
- Sprache, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben.
- Sprachkürzel gemäß der VBScript5-Referenz: Wechselt zur Sprache, die dem angegebenen Sprachkürzel entspricht, z. B. "de-DE" für Deutsch (Deutschland) oder "en-US" für Englisch (Vereinigte Staaten von Amerika).
  Eine Übersicht aller Sprachkürzel finden Sie in den Grundlagen von VBScript unter dem Thema "Gebietsschema-ID (LCID)-Diagramm".

Tastatur

SetzeModusBildschirmtastatur

Beschreibung

Aktiviert oder deaktiviert das automatische Einblenden der Bildschirmtastatur am Bediengerät.

Verwenden Sie diese Systemfunktion, wenn Sie das Einblenden der Bildschirmtastatur verhindern wollen, weil Sie z.B. eine externe Tastatur am Bediengerät angeschlossen haben.

Hinweis

Wenn Sie die Systemfunktion "SetzeModusBildschirmtastatur" ("SetScreenKeyboardMode") auf einem nicht Touch-Bediengerät einsetzen, aktivieren Sie in den Geräteeinstellungen im Bereich "Einstellungen für Runtime" die Option "Bildschirmtastatur verwenden".

Verwendung in der Funktionsliste

SetzeModusBildschirmtastatur (Modus)

Verwendung in benutzerdefinierten Funktionen

SetScreenKeyboardMode (Mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

Parameter

Modus

Legt fest, ob die Bildschirmtastatur eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Bildschirmtastatur wird ausgeblendet

1 (hmiOn) = Ein: Bildschirmtastatur wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her.

Tastaturbedienung für Bildobjekte

ZeigeBlockInTiaPortalVonMeldung

Beschreibung

Führt einen Wechsel von der Meldeanzeige zum betroffenen Programmcode des ausgewählten Projekts im Engineering System aus. Mit dem Wechsel von der Meldeanzeige zum Projekt im TIA Portal wird die weitere Analyse des Programmcodes ermöglicht, z. B. Suche nach einem fehlenden Kontakt im Interlock.

Diese Systemfunktion projektieren Sie an ein Bildobjekt, z. B. eine Schaltfläche, oder an eine Schaltfläche der Symbolleiste in einer Meldeanzeige. Beim Klicken der Schaltfläche wird der notwendige Kontext vom System vorgemerkt und nach dem Sprung ins Engineering System wird automatisch die entsprechende Stelle im Programmcode angezeigt. Über den Parameter Schreibschutz legen Sie fest, mit welcher Berechtigung der Wechsel ins Engineering System aktiviert wird.

Wenn Sie die Systemfunktion an eine Schaltfläche projektiert haben, wird das TIA Portal in dem Modus gestartet, der über den Parameter Schreibschutz festgelegt wurde. Um unterschiedliche Sicherheitszugriffe zu ermöglichen, projektieren Sie die Schaltflächen mit zwei verschiedenen Berechtigungen:

- Für die erste Schaltfläche projektieren Sie die Berechtigung zum uneingeschränkten Zugriff, indem Sie den Parameter Schreibschutz auf FALSE setzen.
- Für die zweite Schaltfläche projektieren Sie die Berechtigung, die keine Änderung der Daten erlaubt, indem Sie den Parameter Schreibschutz auf TRUE setzen.

Verwendung in der Funktionsliste

ZeigeBlockInTIAPortalVonMeldung (Schreibschutz, Offlinemodus, TIA Portal Projektpfad, Bildname der Meldeanzeige, Name der Meldeanzeige, Fehlervariable)

Verwendung in benutzerdefinierten Funktionen

ShowBlockInTIAPortalFromAlarm ReadOnly, Offline_mode, TiaPortalProject_path, AlarmScreen_name, AlarmView_name, Error_tag

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional" (Seite 1567).

Parameter

Schreibschutz

Legt fest, ob das Projekt im Nur-Lese-Modus geöffnet wird.

TRUE (Standard): TIA Portal wird im Lesemodus geöffnet, der Benutzer kann keine Änderung vornehmen.

FALSE: TIA Portal wird mit der Lese- und Schreibberechtigung geöffnet.

Offlinemodus

Legt fest, ob das Projekt im Offline-Modus geöffnet wird.

TRUE: Das TIA Portal geht nach dem Öffnen des Bausteins nicht in den Online-Modus.

FALSE (Standard): Nach dem Öffnen des Bausteins wird der Online-Modus gestartet.

TIA Portal Projektpfad

Pfad und Dateiname des Projekts im TIA Portal, zu dem aus der Meldeanzeige gewechselt wird.

Bildname der Meldeanzeige

Name des Bildes, das die Meldeanzeige enthält.

Name der Meldeanzeige

Objektname der Meldeanzeige, aus der zum TIA Portal gewechselt wird.

Fehlervariable

Im Fehlerfall wird über die Fehlervariable eine Information zum aufgetretenen Fehler ausgegeben.

ZeigePLCCodeAnzeigeVonMeldung

Beschreibung

Führt einen Wechsel von der Meldeanzeige mit der ausgewählten ProDiag-Meldung bzw. der ausgewählten GRAPH-Meldung zur PLC-Code-Anzeige mit dem betroffenen Netzwerk aus. Das Bild, das die PLC-Code-Anzeige enthält, wird entweder in einem Grundbild oder in einem Bildfenster geöffnet.

Verwendung in der Funktionsliste

ZeigePLCCodeAnzeigeVonMeldung (Meldungsbildname, Meldeanzeigename, Grundbildname, Bildfenstername, Bildname der PLC-Code-Anzeige, Name der PLC-Code-Anzeige, Fehlervariable)

Verwendung in benutzerdefinierten Funktionen

ShowPLCCodeViewFromAlarm (Screen, Alarmview, Screen, ScreenWindow, Screen, PLCCodeView, Error tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Systemfunktionen für Runtime Professional" (Seite 1567).

Parameter

Meldungsbildname

Name des Bildes, in dem die Meldeanzeige enthalten ist.

Meldeanzeigename

Objektname der Meldeanzeige, aus der die ausgewählte Meldung übertragen wird.

Grundbildname

Name des Bilds, das ein Bildfenster enthält, in dem das Bild dargestellt werden kann.

Bildfenstername

Name des Bildfensters, in dem das Bild mit der PLC-Code-Anzeige dargestellt werden kann.

Bildname der PLC-Code-Anzeige

Name des Bildes, das die PLC-Code-Anzeige enthält.

Name der PLC-Code-Anzeige

Objektname der PLC-Code-Anzeige

Fehlervariable

Im Fehlerfall wird über die Fehlervariable eine Information zum aufgetretenen Fehler ausgegeben..

Fehlermeldungen

Folgende Fehler können über Fehlervariable zurückgemeldet werden:

KOPAPI_E_TPSF_ALGCTRL_NOTSELECTED	Es muss genau eine Meldung selektiert sein
KOPAPI_E_TPSF_ONLYONLINELISTALLOWED	Bitte nur Meldungen aus der Meldeliste der Meldeanzeige selektieren

KOPAPI_E_TPSF_ONALARMWRONGPRODUCERID	Es ist keine GRAPH/ProDiag-Meldung selektiert
KOPAPI_E_TPSF_ONALARMNOSD0JUMPFLAG	Sprung über diese GRAPH/ProDiag-Meldung nicht möglich. Der Einsprung von einer Meldung in der Meldeanzeige in die PLC-Code-Anzeige wird für folgende Überwachungsmeldungen ermöglicht: • Bei globalen Überwachungen nur für Verriegelungsüberwachungen (Interlock) • Bei lokalen Überwachungen für alle Basisüberwachungen an Input-Parametern

9.11.2 Ereignisse

9.11.2.1 Übersicht

Editoren

Einleitung

Die folgende Tabelle zeigt welche Ereignisse in welchem Editor eintreten.

Technische Änderungen vorbehalten.

Symbol	Editor
	Bilder
	HMI-Meldungen
	HMI-Variablen
	Aufgabenplaner

Abgebaut (Seite 1799)	X	--	--	--
Aktivieren (Seite 1800)	--	--	--	--
Änderung (Seite 1800)	--	--	--	--
Aufgebaut (Seite 1800)	X	--	--	--
Ausführen (Seite 1801)	--	--	--	X
Auswahl geändert (Seite 1801)	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	X	--
Bei Unterschreitung (Seite 1801)	--	--	X	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	X

Beim Schließen eines Dialogs (Seite 1802)	--	--	--	X
Benutzerwechsel (Seite 1803)	--	--	--	X
Bildwechsel (Seite 1803)	--	--	--	X
Deaktivieren (Seite 1804)	--	--	--	X
Doppelklicken (Seite 1804)	--	--	--	--
Drücken (Seite 1805)	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--
Gegangen (Seite 1812)	--	X	--	--
Gekommen (Seite 1812)	--	X	--	--
Klicken (Seite 1813)	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	X	--	--
Loslassen (Seite 1817)	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	X
Quittieren (Seite 1819)	--	X	--	--
Rand erreicht (Seite 1820)	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	X
Taste drücken (Seite 1825)	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--
Wertänderung (Seite 1830)	--	--	X	--
Zeit abgelaufen (Seite 1831)	--	--	--	--

Basisobjekte

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Linie
	Ellipse
	Kreis
	Rechteck

Symbol	Objekt
A	Textfeld
	Grafikanzeige

					A	
Abgebaut (Seite 1799)	--	--	--	--	--	--
Aktivieren (Seite 1800)	--	--	--	--	--	--
Änderung (Seite 1800)	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--
Deaktivieren (Seite 1804)	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--
Klicken (Seite 1813)	--	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--
Wertänderung (Seite 1830)	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--

Elemente

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	EA-Feld
	Schaltfläche
	Symbolisches EA-Feld
	Grafisches EA-Feld
	Datum/Uhrzeit-Feld
	Balken
	Schalter

Abgebaut (Seite 1799)	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	X	X	X	--	--	X
Änderung (Seite 1800)	--	X	X	--	--	--	X
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--	--
Deaktivieren (Seite 1804)	X	X	X	X	--	--	X
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	X	--	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--
Klicken (Seite 1813)	--	X	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	X	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--	--

Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--	X
Umschalten EIN (Seite 1828)	--	--	--	--	--	--	X
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--	--
Wertänderung (Seite 1830)	--	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--	--

Controls

Einleitung

Die folgende Tabelle zeigt welche Ereignisse bei welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Meldeanzeige/Meldefenster
	Meldeindikator
	Kurvenanzeige
	Benutzeranzeige
	HTML-Browser (nur Basic Panel 2nd Generation)
	Rezepturanzeige
	System-Diagnoseanzeige
	Hilfeindikator

Abgebaut (Seite 1799)	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	--	X	X	X	--	--	X
Änderung (Seite 1800)	--	--	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--	--	--

Deaktivieren (Seite 1804)	X	--	X	X	X	--	--	X
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)								
Klicken (Seite 1813)	--	X	--	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	X	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--	--	--
Wertänderung (Seite 1830)	--	--	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--	--	--

9.11.2.2 Übersicht

Editoren

Einleitung

Die folgende Tabelle zeigt welche Ereignisse in welchem Editor eintreten.

Technische Änderungen vorbehalten.

Symbol	Editor
	Bilder
	HMI-Meldungen
	HMI-Variablen
	Archive
	Aufgabenplaner
	Audit Trail

Abgebaut (Seite 1799)	X	--	--	--	--	--
Aktivieren (Seite 1800)	--	--	--	--	--	--
Änderung (Seite 1800)	--	--	--	--	--	--
Aufgebaut (Seite 1800)	X	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	X	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	X	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	X	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	X	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	X	--
Benutzerwechsel (Seite 1803)	--	--	--	--	X	--
Bildwechsel (Seite 1803)	--	--	--	--	X	--
Deaktivieren (Seite 1804)	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--
Gegangen (Seite 1812)	--	X	--	--	--	--
Gekommen (Seite 1812)	--	X	--	--	--	--
Klicken (Seite 1813)	--	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	X	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	X	--
Quittieren (Seite 1819)	--	X	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	X	--
Taste drücken (Seite 1825)	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	X	--	--
Umschalten (Seite 1827)	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	X	--	X
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	X	--	X
Wertänderung (Seite 1830)	--	--	X	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--

Basisobjekte

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Linie
	Polygonzug
	Polygon
	Ellipse
	Kreis
	Rechteck
A	Textfeld
	Grafikanzeige

							A	
Abgebaut (Seite 1799)	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	--	--	--	--	--	--	--	--
Änderung (Seite 1800)	--	--	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--	--	--
Deaktivieren (Seite 1804)	--	--	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--
Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--
Klicken (Seite 1813)	--	--	--	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--	--	--

Quittieren (Seite 1819)	--	--	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--	--	--	--	--
Umschalten (Seite 1827)	--	--	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--	--	--
Wertänderung (Seite 1830)	--	--	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--	--	--

Elemente

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	EA-Feld
	Schaltfläche
	Symbolisches EA-Feld
	Grafisches EA-Feld
	Datum/Uhrzeit-Feld
	Balken
	Schalter
	Symbolbibliothek
	Schieberegler
	Gauge
	Uhr

Abgebaut (Seite 1799)	--	--	--	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	X	X	X	--	--	X	X	X	--	--
Änderung (Seite 1800)	--	X	X	--	--	--	X	--	X	X	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--

Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--	--	--	--	--	--
Deaktivieren (Seite 1804)	X	X	X	X	--	--	X	X	X	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	X	--	--	--
Drücken (Seite 1805)	--	X	--	--	--	--	--	X	--	--	--
Eingabe abgeschlossen (Seite 1810)	X		X	X	--	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--
Klicken (Seite 1813)	--	X	--	--	--	--	--	X	--	--	--
Klicken bei Blinken (Seite 1814)	--	--	--	--	--	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	X	--	--	--	--	--	X	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--	--	--	--	--	--
Taste loslassen (Seite 1825)	--	--	--	--	--	--	--	--	--	--	--
Überlauf (Seite 1826)	--	--	--	--	--	--	--	--	--	--	--
Umschalten (Seite 1827)	--	--	--	--	--	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--	X	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--	X	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--	--	--	--	--	X
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--	--	--	--	--	X
Wertänderung (Seite 1830)	--	--	--	--	--	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--	--	--	--	--	--

Controls

Einleitung

Die folgende Tabelle zeigt welche Ereignisse bei welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Meldeanzeige/Meldefenster
	Meldeindikator

Symbol	Objekt
	Kurvenanzeige
	Benutzeranzeige
	HTML-Browser
	Beobachtungstabelle
	Sm@rtClient-Anzeige
	Rezepturanzeige
	f(x)-Kurvenanzeige
	System-Diagnoseanzeige/System-Diagnosefenster
	Media Player[1]
	PLC-Code-Anzeige
	GRAPH-Übersicht
	ProDiag-Übersicht
	Kriterienanalyse-Anzeige
	Kamera-Anzeige[1]
	PDF-Anzeige[1]
	Funktionstaste

[1] Nur bei Comfort-Bediengeräten

Abgebaut (Seite 1799)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	--	X	X	X	X	X	X	--	X	--	X	X	X	X	X	X	--
Änderung (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	X	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Bei Überschreitung (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Bei Unterschreitung (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Beim Öffnen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Beim Schließen eines Dialogs (Seite 1802)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Benutzerwechsel (Seite 1803)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Bildwechsel (Seite 1803)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Deaktivieren (Seite 1804)	X	--	X	X	X	X	X	X	--	--	--	X	X	X	X	X	X	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Eingabe abgeschlossen (Seite 1810)	--	--	--	--	--	--	--	--		--	--	--	--	--	--	--	--	--
ESC doppelt drücken (Seite 1811)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Klicken (Seite 1813)	--	X	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Klicken bei Blinken (Seite 1814)	--	X	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Meldungspufferüberlauf (Seite 1817)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Rand erreicht (Seite 1820)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Meldeanzeige klicken (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	X	X	--	--	--	--
Schaltfläche PLC-Code-Anzeige klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Taste drücken (Seite 1825)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X
Taste loslassen (Seite 1825)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X
Überlauf (Seite 1826)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Umschalten (Seite 1827)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Umschalten AUS (Seite 1827)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Umschalten EIN (Seite 1828)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Wenig freier Speicher (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Wenig freier Speicher, kritisch (Seite 1830)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Wertänderung (Seite 1830)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Zeit abgelaufen (Seite 1831)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

9.11.2.3 Übersicht

Editoren

Einleitung

Die folgende Tabelle zeigt welche Ereignisse in welchem Editor eintreten.

Technische Änderungen vorbehalten.

Symbol	Editor
	Bilder
	HMI-Meldungen
	Archive
	Aufgabenplaner

Abspielen beendet (Seite 1799)	--	--	--	--
Aktivieren (Seite 1800)	X	--	--	--
Änderung (Seite 1800)	--	--	--	--
Aufgebaut (Seite 1800)	X	--	--	--
Ausführen (Seite 1801)	--	--	--	X
Auswahl geändert (Seite 1801)	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--
Drücken (Seite 1805)	--	--	--	--
Eigenschaft Bedienmeldung geändert (Seite 1805)	--	--	--	--
Eigenschaft Block geändert (Seite 1805)	--	--	--	--
Eigenschaft Diagramm geändert (Seite 1806)	--	--	--	--
Eigenschaft geändert (Seite 1806)	--	--	--	--
Eigenschaft Kurve geändert (Seite 1806)	--	--	--	--
Eigenschaft Spalte geändert (Seite 1807)	--	--	--	--
Eigenschaft Spalte Meldestatistik geändert (Seite 1807)	--	--	--	--
Eigenschaft Spaltenname Statistikbereich geändert (Seite 1807)	--	--	--	--
Eigenschaft Statuszeilenelement ändern (Seite 1808)	--	--	--	--
Eigenschaft Symbolleiste Schaltfläche ändern (Seite 1808)	--	--	--	--
Eigenschaft Wertachse ändern (Seite 1808)	--	--	--	--
Eigenschaft Wertspalte ändern (Seite 1809)	--	--	--	--
Eigenschaft X-Achse geändert (Seite 1809)	--	--	--	--
Eigenschaft Y-Achse geändert (Seite 1809)	--	--	--	--
Eigenschaft Zeitachse ändern (Seite 1810)	--	--	--	--
Eigenschaft Zeitspalte ändern (Seite 1810)	--	--	--	--
Fehler (Seite 1811)	--	--	--	--
Gegangen (Seite 1812)	--	X	--	--

Gekommen (Seite 1812)	--	X	--	--
Im Meldebereich (Seite 1812)	--	--	--	--
Im Toleranzbereich (Seite 1813)	--	--	--	--
Im Warnbereich (Seite 1813)	--	--	--	--
Klicken (Seite 1813)	X	--	--	--
Kontextmenü (Seite 1814)	--	--	--	--
Lautstärke (Seite 1815)	--	--	--	--
Layout ändern (Seite 1815)	--	--	--	--
Lineal ändern (Seite 1815)	--	--	--	--
Linke Maustaste drücken (Seite 1816)	X	--	--	--
Linke Maustaste loslassen (Seite 1816)	X	--	--	--
Loop-In-Alarm (Seite 1816)	--	X	--	--
Loslassen (Seite 1817)	--	--	--	--
Maus bewegen (Seite 1818)	--	--	--	--
Medium geöffnet (Seite 1818)	--	--	--	--
Medium geschlossen (Seite 1818)	--	--	--	--
Objekt geändert (Seite 1819)	X	--	--	--
Pause (Seite 1819)	--	--	--	--
Quittieren (Seite 1819)	X	X	--	--
Rechte Maustaste drücken (Seite 1820)	X	--	--	--
Rechte Maustaste loslassen (Seite 1821)	X	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--
Schaltfläche Meldeanzeige klicken (Seite 1821)	--	--	--	--
Schaltfläche PLC-Code-Anzeige klicken (Seite 1822)	--	--	--	--
Schaltfläche Symbolleiste klicken (Seite 1822)	--	--	--	--
Schieber verändern (Seite 1822)	--	--	--	--
SchrittVor (Seite 1823)	--	--	--	--
SchrittZurück (Seite 1823)	--	--	--	--
Sperre geändert (Seite 1823)	--	--	X	--
Status geändert (Seite 1824)	--	X	--	--
Stopp (Seite 1824)	--	--	--	--
Symbolleiste umschalten (Seite 1824)	--	--	--	--
Taste auf Tastatur drücken (Seite 1826)	X	--	--	--
Taste auf Tastatur loslassen (Seite 1826)	X	--	--	--
Videoformat ändern (Seite 1828)	--	--	--	--
Vollbild (Seite 1829)	--	--	--	--
Vollbild verlassen (Seite 1829)	--	--	--	--
Wiedergabe (Seite 1831)	--	--	--	--

Basisobjekte

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Linie
	Polygonzug
	Polygon
	Ellipse
	Ellipsensegment
	Kreissegment
	Ellipsenbogen
	Kreisbogen
	Kreis
	Reckteck
	Verbinder
A	Textfeld
	Grafikanzeige
	Rohr
	Doppel-T-Stück
	T-Stück
	Rohrbogen

												A					
Abspielen beendet (Seite 1799)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Änderung (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Bedienmeldung geändert (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Block geändert (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Eigenschaft Diagramm geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Kurve geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spalte geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spalte Meldestatistik geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spaltenname Statistikbereich geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Statuszeilenelement ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Symbolleiste Schaltfläche ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Wertachse ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Wertspalte ändern (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft X-Achse geändert (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Y-Achse geändert (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitachse ändern (Seite 1810)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitspalte ändern (Seite 1810)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Fehler (Seite 1811)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Meldebereich (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Toleranzbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Warnbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Klicken (Seite 1813)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Kontextmenü (Seite 1814)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Lautstärke (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Layout ändern (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Lineal ändern (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Linke Maustaste drücken (Seite 1816)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Linke Maustaste loslassen (Seite 1816)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Maus bewegen (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Medium geöffnet (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Medium geschlossen (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Objekt geändert (Seite 1819)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Pause (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Rechte Maustaste drücken (Seite 1820)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Rechte Maustaste loslassen (Seite 1821)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Meldeanzeige klicken (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche PLC-Code-Anzeige klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Symbolleiste klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schieber verändern (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
SchrittVor (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
SchrittZurück (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Sperre geändert (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Status geändert (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Stopp (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Symbolleiste umschalten (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Taste auf Tastatur drücken (Seite 1826)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Taste auf Tastatur loslassen (Seite 1826)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Vollbild (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Vollbild verlassen (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Videoformat ändern (Seite 1828)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Wiedergabe (Seite 1831)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Elemente

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchen Elementen eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	EA-Feld
	Editierbares Textfeld
	Listenfeld
	Kombinationsfeld
	Schaltfläche
	Rundschaltfläche
	Symbolisches EA-Feld
	Grafisches EA-Feld
	Balken
	Symbolbibliothek
	Schieberegler
	Bildlaufleiste
	Kontrollkästchen
	Optionsschaltfläche
	Zeigerinstrument
	Uhr
	Speicherplatzanzeige

Abspielen beendet (Seite 1799)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aktivieren (Seite 1800)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Änderung (Seite 1800)	--	--	--	--	--	--	--	--	--	--	X	--	--	--	X	--	--
Aufgebaut (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ausführen (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--
Eigenschaft Bedienmeldung geändert (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Block geändert (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Diagramm geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Kurve geändert (Seite 1806)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spalte geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spalte Meldestatistik geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spaltenname Statistikbereich geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Statuszeilenelement ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Symbolleiste Schaltfläche ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Wertachse ändern (Seite 1808)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Wertspalte ändern (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft X-Achse geändert (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Y-Achse geändert (Seite 1809)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitachse ändern (Seite 1810)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitspalte ändern (Seite 1810)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Fehler (Seite 1811)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Im Meldebereich (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X
Im Toleranzbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X
Im Warnbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X
Klicken (Seite 1813)	X	X	X	X	X	X	X	X	X	X	--	X	X	X	--	--	--
Kontextmenü (Seite 1814)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Lautstärke (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Layout ändern (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Lineal ändern (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Linke Maustaste drücken (Seite 1816)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Linke Maustaste loslassen (Seite 1816)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--
Maus bewegen (Seite 1818)	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--
Medium geöffnet (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Medium geschlossen (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Objekt geändert (Seite 1819)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Pause (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Rechte Maustaste drücken (Seite 1820)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Rechte Maustaste loslassen (Seite 1821)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Meldeanzeige klicken (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche PLC-Code-Anzeige klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Symbolleiste klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Schieber verändern (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
SchrittVor (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
SchrittZurück (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Sperre geändert (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Status geändert (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Stopp (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Symbolleiste umschalten (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Taste auf Tastatur drücken (Seite 1826)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Taste auf Tastatur loslassen (Seite 1826)	X	X	X	X	X	X	X	X	X	--	--	X	X	X	--	--	--
Videoformat ändern (Seite 1828)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Vollbild (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Vollbild verlassen (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Wiedergabe (Seite 1831)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Controls

Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchen Controls eintreten.

Technische Änderungen vorbehalten.

Symbol	Objekt
	Bildfenster
	Benutzeranzeige
	HTML-Browser
	Druckauftrag/Skriptdiagnose
	Rezepturanzeige
	Meldeanzeige
	f(x)-Kurvenanzeige
	f(t)-Kurvenanzeige
	Tabellenanzeige
	Wertetabelle
	System-Diagnoseanzeige
	Media Player

Symbol	Objekt
	Kanaldiagnose-Anzeige
	PLC-Code-Anzeige
	GRAPH-Übersicht
	ProDiag-Übersicht
	Kriterienanlyse-Anzeige

Abspielen beendet (Seite 1799)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Aktivieren (Seite 1800)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Änderung (Seite 1800)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Aufgebaut (Seite 1800)	--	--	--	--	X	X	X	X	X	X	--	--	--	--	--	--	--
Ausführen (Seite 1801)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Auswahl geändert (Seite 1801)	--	--	--	--	X	--	--	--	--	--	--	--	--	--	--	--	--
Doppelklicken (Seite 1804)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Drücken (Seite 1805)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Bedienmeldung geändert (Seite 1805)	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Block geändert (Seite 1805)	--	--	--	--	--	X	--	--	--	X	--	--	--	--	--	--	--
Eigenschaft Diagramm geändert (Seite 1806)	--	--	--	--	--	--	X	X	--	--	--	--	--	--	--	--	--
Eigenschaft geändert (Seite 1806)	--	--	--	--	X	X	X	X	X	X	--	--	--	--	--	--	--
Eigenschaft Kurve geändert (Seite 1806)	--	--	--	--	--	--	X	X	--	--	--	--	--	--	--	--	--
Eigenschaft Spalte geändert (Seite 1807)	--	--	--	--	X	X	--	--	--	X	--	--	--	--	--	--	--
Eigenschaft Spalte Meldestatistik geändert (Seite 1807)	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--	--	--
Eigenschaft Spaltenname Statistikbereich geändert (Seite 1807)	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--
Eigenschaft Statuszeilenelement ändern (Seite 1808)	--	--	--	--	X	X	X	X	X	X	--	--	--	--	--	--	--
Eigenschaft Symbolleiste Schaltfläche ändern (Seite 1808)	--	--	--	--	X	X	X	X	X	X	--	--	--	--	--	--	--

Eigenschaft Wertachse ändern (Seite 1808)	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--
Eigenschaft Wertspalte ändern (Seite 1809)	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--
Eigenschaft X-Achse geändert (Seite 1809)	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--	--
Eigenschaft Y-Achse geändert (Seite 1809)	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitachse ändern (Seite 1810)	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--
Eigenschaft Zeitspalte ändern (Seite 1810)	--	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--
Fehler (Seite 1811)	--	--	--	--	--	--	--	--	--	--	--	--	--	X	--	X	X
Gegangen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Gekommen (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Meldebereich (Seite 1812)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Toleranzbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Im Warnbereich (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Klicken (Seite 1813)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X	--	--
Kontextmenü (Seite 1814)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Lautstärke (Seite 1815)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Layout ändern (Seite 1815)	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--
Lineal ändern (Seite 1815)	--	--	--	--	--	--	--	X	--	--	--	--	--	--	--	--	--
Linke Maustaste drücken (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Linke Maustaste loslassen (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loop-In-Alarm (Seite 1816)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Loslassen (Seite 1817)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Maus bewegen (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Medium geöffnet (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Medium geschlossen (Seite 1818)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Objekt geändert (Seite 1819)	X	--	--	X	--	--	--	--	--	--	--	--	--	--	--	--	--

Pause (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Quittieren (Seite 1819)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Rechte Maustaste drücken (Seite 1820)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Rechte Maustaste loslassen (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Runtime-Stopp (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Schaltfläche Meldeanzeige klicken (Seite 1821)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X	X	--
Schaltfläche PLC-Code-Anzeige klicken (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	X	--	--
Schaltfläche Symbolleiste klicken (Seite 1822)	--	--	--	--	X	X	X	X	X	X	--	--	--	--	--	--	--
Schieber verändern (Seite 1822)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
SchrittVor (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
SchrittZurück (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Sperre geändert (Seite 1823)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Status geändert (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Stopp (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Symbolleiste umschalten (Seite 1824)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Taste auf Tastatur drücken (Seite 1826)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Taste auf Tastatur loslassen (Seite 1826)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
Videoformat ändern (Seite 1828)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Vollbild (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Vollbild verlassen (Seite 1829)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--
Wiedergabe (Seite 1831)	--	--	--	--	--	--	--	--	--	--	--	X	--	--	--	--	--

9.11.2.4 Ereignisse

Abgebaut

Beschreibung

Tritt ein, wenn das aktive Bild am Bediengerät abgebaut ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Abspielen beendet

Beschreibung

Tritt ein, wenn die Video- oder Audiodatei abgespielt ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Aktivieren

Beschreibung

Tritt ein, wenn der Benutzer ein Anzeige- und Bedienobjekt über die projektierte Tab-Reihenfolge auswählt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Hinweis

Wenn der Benutzer z. B. eine Schaltfläche mit der Maus anklickt, wird das Ereignis "Klicken" ausgelöst. Um das Ereignis "Aktivieren" auszulösen, wählt der Benutzer die Schaltfläche über die Tab-Reihenfolge aus.

Das Ereignis "Aktivieren" dient ausschließlich zur Erkennung, ob ein Objekt ausgewählt wurde. Das Ereignis löst aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Aktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

Änderung

Beschreibung

Tritt ein, wenn sich der Zustand eines Anzeige- und Bedienobjekts ändert.

Wenn der Benutzer z. B.einen Schalter betätigt, ändert sich der Zustand eines Objekts.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Aufgebaut

Beschreibung

Tritt ein, wenn ein Anzeige- und Bedienobjekt in Runtime vollständig geladen ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Ausführen

Beschreibung

Tritt ein, wenn die geplante Aufgabe ausgeführt wird.

Auswahl geändert

Beschreibung

Tritt ein, wenn sich die Auswahl ändert.

Bei Überschreitung

Beschreibung

Tritt ein, wenn der obere Grenzwert einer Variablen überschritten wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Bei Unterschreitung

Beschreibung

Tritt ein, wenn der untere Grenzwert einer Variablen unterschritten wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Beim Öffnen eines Dialogs

Beschreibung

Tritt ein, wenn sich ein modaler Dialog öffnet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Beim Schließen eines Dialogs

Beschreibung

Tritt ein, wenn sich ein modaler Dialog schließt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Benutzerwechsel

Beschreibung

Tritt ein, wenn sich ein Benutzer am Bediengerät abmeldet oder sich ein anderer Benutzer am Bediengerät anmeldet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Bildwechsel

Beschreibung

Tritt ein, wenn nach einem Bildwechsel alle projektierten Anzeige- und Bedienobjekte im Bild geladen sind.

Um bei einem Bildwechsel zu einem bestimmten Bild andere Systemfunktionen auszuführen, verwenden Sie das Ereignis "Aufgebaut".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Deaktivieren

Beschreibung

Tritt ein, wenn der Benutzer den Fokus von einem Anzeige- und Bedienobjekt den Fokus nimmt.

Sie desaktivieren ein Bildobjekt über die projektierte Tab-Reihenfolge oder wenn Sie mit der Maus eine andere Aktion ausführen.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Hinweis

Systemfunktionen oder benutzerdefinierte Funktionen am Ereignis "Deaktivieren" eines Bilds werden bei Bildabwahl nicht ausgeführt.

Das Ereignis "Deaktivieren" dient ausschließlich zur Erkennung, ob ein Objekt abgewählt wurde. Das Ereignis löst aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Deaktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Doppelklicken

Beschreibung

Tritt ein, wenn der Benutzer auf ein Objekt aus der Symbolbibliothek doppelklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Drücken

Beschreibung

Tritt ein, wenn der Benutzer auf eine Schaltfläche mit der linken Maustaste klickt, die Taste <RETURN> oder <SPACE> drückt.

Tritt auch ein, wenn der Benutzer auf ein Objekt der Symbolbibliothek mit der rechten Maustaste klickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Bedienmeldung geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft einer Bedienmeldung ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Block geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft eines Blocks ändert, z. B. Meldeblock.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Diagramm geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft des Kurvendiagramms ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft geändert

Beschreibung

Tritt ein, wenn sich eine Eigenschaft ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Kurve geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Kurve ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Spalte geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft einer Spalte ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Spalte Meldestatistik geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Spalte Meldestatistik ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Spaltenname Statistikbereich geändert

Beschreibung

Tritt ein, wenn sich der Spaltenname des Statistikbereichs ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Statuszeilenelement ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft eines Elements in der Statuszeile ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Symbolleiste Schaltfläche ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft einer Schaltfläche in der Symbolleiste ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Wertachse ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Wertachse ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Wertspalte ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Wertespalte ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft X-Achse geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft der X-Achse ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Y-Achse geändert

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Y-Achse ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Zeitachse ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Zeitachse ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eigenschaft Zeitspalte ändern

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Zeitspalte ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Eingabe abgeschlossen

Beschreibung

Tritt ein, wenn die Eingabe in einem EA-Feld mit der Enter-Taste, mit der Maus oder per Touchbedienung bestätigt wird.

Das Ereignis "Eingabe abgeschlossen" wird auch gestartet, wenn sich der Wert einer Variable nicht ändert, z. B. bei einer Wertüberschreitung, oder wenn ein Benutzer den Dialog zum Quittieren einer quittierpflichtigen Variablen, z. B. beim Optionspaket Audit, abbricht.

Beim Anmelden als Benutzer und bei Eingabefeldern, die mit einer Berechtigung projektiert wurden, wird das Ereignis hingegen nicht ausgelöst.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)
Basisobjekte (Seite 1782)
Elemente (Seite 1783)
Controls (Seite 1784)

ESC doppelt drücken

Beschreibung

Tritt ein, wenn der Bediener die Taste <ESC> am Bediengerät zweimal drückt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)
Basisobjekte (Seite 1782)
Elemente (Seite 1783)
Controls (Seite 1784)

Fehler

Beschreibung

Tritt ein, wenn ein Fehler während des Aufbaus der Anzeige eines Programmnetzwerkes in der PLC-Code-Anzeige auftritt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Gegangen

Beschreibung

Tritt ein, wenn eine Meldung geht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Gekommen

Beschreibung

Tritt ein, wenn eine Meldung ausgelöst wurde und in der Meldeanzeige angezeigt wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Im Meldebereich

Beschreibung

Tritt ein, wenn der verfügbare Speicherplatz die projektierte Überwachungsgrenze "Alarm" erreicht hat. Dieses Ereignis projektieren Sie im Anzeige- und Bedienobjekt "Speicherplatzanzeige".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Im Toleranzbereich

Beschreibung

Tritt ein, wenn der verfügbare Speicherplatz die projektierte Überwachungsgrenze "Toleranz" erreicht hat. Dieses Ereignis projektieren Sie im Anzeige- und Bedienobjekt "Speicherplatzanzeige".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Im Warnbereich

Beschreibung

Tritt ein, wenn der verfügbare Speicherplatz die projektierte Überwachungsgrenze "Warnung" erreicht hat. Dieses Ereignis projektieren Sie im Anzeige- und Bedienobjekt "Speicherplatzanzeige".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Klicken

Beschreibung

Tritt ein, wenn der Benutzer ein Anzeige- und Bedienobjekt mit der Maus anklickt oder am Touch-Gerät mit dem Finger berührt.

Wenn Sie ein falsches Objekt angeklickt haben, verhindern Sie die Abarbeitung der projektierten Funktionsliste wie folgt:

- Bewegen Sie den Mauszeiger mit gedrückter Maustaste vom Objekt weg. Sobald der Mauszeiger das Objekt verlassen hat, lassen Sie die Maustaste los. Die Funktionsliste wird dann nicht abgearbeitet.
- Bei Touch-Geräten berühren Sie das Display so lange mit dem Finger bis eine Aktion ausgeführt wird, z. B. ein Bildwechsel.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Klicken bei Blinken

Beschreibung

Tritt ein, wenn der Benutzer einen blinkenden Meldeindikator mit der Maus anklickt oder mit dem Finger berührt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Kontextmenü

Beschreibung

Tritt ein, wenn sich das Kontextmenü öffnet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Lautstärke

Beschreibung

Tritt ein, wenn sich die Lautstärke ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Layout ändern

Beschreibung

Tritt ein, wenn sich eine Eigenschaft ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Lineal ändern

Beschreibung

Tritt ein, wenn sich eine Eigenschaft ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Linke Maustaste drücken

Beschreibung

Tritt ein, wenn der Bediener mit der linken Maustaste ein Anzeige- und Bedienobjekt anklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Linke Maustaste loslassen

Beschreibung

Tritt ein, wenn der Benutzer die linke Maustaste nach dem Klicken loslässt.

Solange der Benutzer die linke Maustaste gedrückt hält, tritt dieses Ereignis nicht ein.

Wenn der Bediener die Maustaste nach dem Klicken auf eine Schaltfläche loslässt, tritt dieses Ereignis auch ein, wenn sich der Mauszeiger nicht mehr über der Schaltfläche befindet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Loop-In-Alarm

Beschreibung

Das Ereignis "Loop-In-Alarm" tritt ein, wenn der Benutzer eine Meldung in der Meldeanzeige markiert und auf die Schaltfläche "Loop-In-Alarm" klickt oder auf die Meldung doppelklickt.

Das Ereignis "Loop-In-Alarm" ist bei Bitmeldungen und bei Analogmeldungen projektierbar. An das Ereignis "Loop-In-Alarm" kann eine Systemfunktion projektiert werden, z. B. wechseln zu einem Bild, in dem Sie nähere Hinweise zur aufgetretenen Meldung geben. Ist beim Ereignis Loop-In-Alarm eine Funktion projektiert und die markierte Meldung ist quittierpflichtig, dann wird diese Meldung beim ersten Auftreten des Ereignisses ebenfalls quittiert.

Wenn kein Loop-In-Alarm für die markierte Meldung projektiert ist, hat das Betätigen der Schaltfläche "Loop-In-Alarm" keine Auswirkung.

An das Ereignis "Loop-In-Alarm" können Sie bei einer Runtime Professional keine lokalen Skripte projektieren.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Loslassen

Beschreibung

Tritt ein, wenn der Benutzer eine Schaltfläche loslässt.

Solange Sie die Schaltfläche gedrückt halten, tritt dieses Ereignis nicht ein.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Meldungspufferüberlauf

Beschreibung

Tritt ein, wenn die projektierte Größe des Meldepuffers erreicht ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Maus bewegen

Beschreibung

Tritt ein, wenn der Bediener die Maus bewegt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Medium geöffnet

Beschreibung

Tritt ein, wenn das Medium geöffnet ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Medium geschlossen

Beschreibung

Tritt ein, wenn das Medium geschlossen ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Objekt geändert

Beschreibung

Tritt ein, wenn der Bediener in Runtime eine Eigenschaft eines Anzeige- und Bedienobjekts ändert.

Hinweis

Jede von diesem Ereignis ausgelöste Aktion wird einzeln angemeldet und beim Bildwechsel abgemeldet. Dieses Verhalten belastet den Arbeitsspeicher. Verwenden Sie dieses Ereignis nur für wichtige Änderungen von Eigenschaften, z. B. für ein E/A-Feld.

Hinweis

Die Verfügbarkeit des Ereignisses ist abhängig von Bediengerät und Objekttyp.

Pause

Beschreibung

Tritt ein, wenn das Abspielen der Video- oder Audiodatei angehalten wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Quittieren

Beschreibung

Tritt ein, wenn der Benutzer eine Meldung quittiert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Rand erreicht

Beschreibung

Tritt ein, wenn der Benutzer den Anfang oder das Ende eines scrollbaren Bereichs erreicht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Hinweis

An das Ereignis "Rand erreicht" darf keine benutzerdefinierte Funktion projektiert werden.

Projektierbare Objekte

Das Ereignis können Sie nur an die Tasten <Auf> und <Ab> projektieren oder an die Tasten, an die Sie die Systemfunktionen "BildobjektSeiteAufwärts" oder "BildobjektSeiteAbwärts" projektiert haben.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Rechte Maustaste drücken

Beschreibung

Tritt ein, wenn der Benutzer ein Anzeige- und Bedienobjekt mit der rechten Maustaste anklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Rechte Maustaste loslassen

Beschreibung

Tritt ein, wenn der Benutzer die rechte Maustaste nach dem Klicken loslässt.

Solange der Benutzer die rechte Maustaste gedrückt hält, tritt dieses Ereignis nicht ein.

Wenn der Bediener die Maustaste nach dem Klicken auf eine Schaltfläche loslässt, tritt dieses Ereignis auch ein, wenn sich der Mauszeiger nicht mehr über der Schaltfläche befindet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Runtime-Stopp

Beschreibung

Tritt ein, wenn der Benutzer die Runtime-Software am Bediengerät beendet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Hinweis

An das Ereignis "Runtime-Stopp" darf keine benutzerdefinierte Funktion projektiert werden.

Schaltfläche Meldeanzeige klicken

Beschreibung

Tritt ein, wenn der Benutzer die Schaltfläche "Meldeanzeige" anklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Schaltfläche PLC-Code-Anzeige klicken

Beschreibung

Tritt ein, wenn der Benutzer die Schaltfläche "PLC-Code-Anzeige" anklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Schaltfläche Symbolleiste klicken

Beschreibung

Tritt ein, wenn der Benutzer eine Schaltfläche in der Symbolleiste anklickt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Schieber verändern

Beschreibung

Tritt ein, wenn sich der Schieber der Lautstärke verändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

SchrittVor

Beschreibung

Tritt ein, wenn die Video- oder Audiodatei zur nächsten Sequenz geht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

SchrittZurück

Beschreibung

Tritt ein, wenn die Video- oder Audiodatei zur vorherigen Sequenz geht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Sperre geändert

Beschreibung

Tritt ein, wenn sich beim Archiv der Schreibzugriff ändert, z. B. von gesperrt nach frei gegeben.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Status geändert

Beschreibung

Tritt ein, wenn sich der Status einer Meldung ändert, z. B. von Gekommen nach Gegangen.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Stopp

Beschreibung

Tritt ein, wenn das Abspielen der Video- oder Audiodatei stoppt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Symbolleiste umschalten

Beschreibung

Tritt ein, wenn sich die Eigenschaft der Zeitachse ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Taste drücken

Beschreibung

Tritt ein, wenn der Benutzer eine Funktionstaste drückt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Taste loslassen

Beschreibung

Tritt ein, wenn der Benutzer eine Funktionstaste loslässt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Taste auf Tastatur drücken

Beschreibung

Tritt ein, wenn der Benutzer eine Taste auf der Tastatur drückt.

Die Tasten <F10> und <ALT+DRUCK> verwendet der Bediener für die Prozessbedienung nicht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Taste auf Tastatur loslassen

Beschreibung

Tritt ein, wenn der Benutzer eine Taste auf der Tastatur loslässt.

Die Tasten <F10> und <ALT+DRUCK> verwendet der Bediener für die Prozessbedienung nicht.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Überlauf

Beschreibung

Tritt ein, wenn die projektierte Größe eines Archivs erreicht ist. Sie verwenden den Archivtyp "Ereignis auslösen".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Umschalten

Beschreibung

Tritt ein, wenn der Benutzer einen Anzeige- und Bedienobjekt umschaltet, z. B. einen Schalter von "Ein" nach "Aus".

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Umschalten AUS

Beschreibung

Tritt ein, wenn der Benutzer das Anzeige- und Bedienobjekt "Schalter" in die Stellung AUS bewegt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Umschalten EIN

Beschreibung

Tritt ein, wenn der Benutzer das Anzeige- und Bedienobjekt "Schalter" in die Stellung EIN bewegt.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Verbunden

Beschreibung

Tritt ein, wenn der Benutzer ein Mobile Panels mit einer Anschluss-Box verbindet.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät ist.

Videoformat ändern

Beschreibung

Tritt ein, wenn sich das Videoformat ändert.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Vollbild

Beschreibung

Tritt ein, wenn der Vollbildmodus aktiviert wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Vollbild verlassen

Beschreibung

Tritt ein, wenn der Vollbildmodus verlassen wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Wenig freier Speicher

Beschreibung

Tritt ein, wenn auf dem Speichermedium, auf dem ein Audit Trail gespeichert wird, weniger Speicherplatz als die konfigurierte Mindestmenge zur Verfügung steht.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Wenig freier Speicher, kritisch

Beschreibung

Tritt ein, wenn hardwarebedingt auf dem Speichermedium, auf dem ein Audit Trail gespeichert wird, zu wenig Speicherplatz zur Verfügung steht.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Wertänderung

Beschreibung

Tritt ein, wenn sich der Wert einer Variablen oder der Wert eines Arrayelements ändert.

Die Wertänderung einer Variablen wird von der Steuerung ausgelöst oder vom Benutzer, z. B. wenn er einen neuen Wert eingibt. Wenn eine Systemfunktion den Wert ändert, wird kein Ereignis ausgegeben.

Hinweis

Beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

Wiedergabe

Beschreibung

Tritt ein, wenn die Video- oder Audiodatei wiedergegeben wird.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Elemente (Seite 1792)

Zeit abgelaufen

Beschreibung

Tritt ein, wenn die projektierte Zeit im Aufgabenplaner abgelaufen ist.

Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

Siehe auch

Editoren (Seite 1780)

Basisobjekte (Seite 1782)

Elemente (Seite 1783)

Controls (Seite 1784)

9.11.3 WinCC Engineering V15.1 - Programmierreferenz

WinCC Programmierreferenz

Eine detaillierte Beschreibung zu VB-Scripting, C-Scripting und Runtime API finden Sie im Handbuch "WinCC Engineering V15.1 - Programmierreferenz".

Mobile Panels 10

10.1 Grundlagen

10.1.1 Einsatzgebiet Mobile Panel Wireless

Einleitung

Mit dem Mobile Panel Wireless bewegen Sie sich uneingeschränkt durch die Anlage, wahlweise mit oder ohne Sicherheitstechnik. Sie bedienen und verfolgen situationsabhängig bewegliche Maschinenteile oder die gesamte Anlage. Sie planen, konfigurieren, simulieren und erweitern Funknetze schnell und sicher.

Aufbauend auf der standardisierten WLAN-Technologie sind die Mobile Panels Wireless speziell für die Anforderungen in der Automatisierung zugeschnitten.

WLAN-Bereich

Der WLAN-Bereich ist der Bereich der Anlage, in dem das Bediengerät über ein drahtloses lokales Netzwerk mit der Steuerung kommuniziert. Die Funkabdeckung des Funknetzes verfügt in diesem Bereich über eine ausreichende Signalstärke.

Das WLAN einer Anlage wird durch den Funkbereich eines oder mehrerer Access Points gebildet.

Access Point

Der Access Point dient als Netzübergang zwischen dem drahtlosen und dem kabelgebundenen Netzwerk. Das Bediengerät kommuniziert über WLAN mit einem Access Point. Sie bedienen ohne störendes Kabel die verschiedenen Maschinen oder Anlagen. Über den Access Point ist das Bediengerät mit einem Netzwerk verbunden, in dem es mit einer Steuerung kommuniziert.

Wirkbereich

Sicherheitsrelevante Bedienungen sind nur in einem begrenzten Teil eines WLAN-Bereichs vor einer Maschine oder Anlage möglich, dem Wirkbereich. Eine Ausnahme ist der Nothalt, der im gesamten WLAN-Bereich funktioniert.

Im Bediengerät sind die projektierten Wirkbereiche hinterlegt. Die Anmeldung des Bediengeräts an einer Maschine ist nur innerhalb des Wirkbereichs möglich. Vorausgesetzt kein anderes Bediengerät ist am selben Wirkbereich angemeldet.

Damit Anmeldung und Maschinenbedienung eindeutig sind, darf sich ein Wirkbereich nicht mit einem anderen Wirkbereich überschneiden.

Tags

Wirkbereiche formen Sie durch die räumliche Verteilung von Tags an einer Maschine oder Anlage. Durch Tags wird ein geometrischer Raum aufgespannt, der für die Anmeldung an einer Maschine relevant ist. Nach der Anmeldung bedienen Sie die Maschine mit Zustimmtaster.

Zwei Arten von Tags werden unterschieden:

- Transponder
- RFID Tag

Transponder

Nach der erfolgreichen Anmeldung an einer Maschine findet eine Überwachung des aufgespannten geometrischen Raumes statt. Eine Abstandsmessung zwischen Bediengerät und Transponder ist nur möglich, wenn sich beide Geräte jeweils im Empfangsbereich des anderen befinden. Wenn Sie den Wirkbereich verlassen, dann wird die sicherheitsrelevante Bedienung der Maschine mit Zustimmtaster verhindert.

Zonen im Transpondersystem

Zonen sind projektierbare Objekte innerhalb eines Transpondersystems. Eine Zone wird definiert durch den maximalen Abstand zu einem oder mehreren Transpondern. Bei einem kabelgebundenen Mobile Panel definiert die Länge des Anschlusskabels eine Zone um eine Anschluss-Box.

Zonen sind Bedienbereiche, in denen sicherheitsbezogenes Bedienen nicht möglich ist. Für den Eintritt in eine Zone projektieren Sie z. B. einen Bildwechsel zum passenden Anlagenbild.

RFID Tag

Nach erfolgreicher Anmeldung an einer Maschine findet keine Überwachung des aufgespannten geometrischen Raums statt. Die Überwachung erfordert daher eine zusätzliche organisatorische Maßnahme z. B.: eine Lichtschranke oder einen Schutzzaun. Durch ein Sicherheitssystem ist der Schutzbereich von der Anlage abgegrenzt. Der Schutzbereich ist der Bereich in der Anlage, in dem Sie eine oder mehrere Maschinen im fehlersicheren Betrieb bedienen. Innerhalb von Schutzbereichen sind ein oder mehrere RFID Tags verteilt. Der Schutzbereich ist kein projektierbares Objekt.

10.1.2 Funktionsweise des Transpondersystems

Übersicht

Zum grundlegenden Verständnis beschreibt der Abschnitt folgendes Zusammenspiel:

- Bediengerät
- Zone

- Wirkbereich
- Transponder

Hinweis

Schützen Sie das Projekt, das ein Mobile Panel Wireless für den fehlersicheren Betrieb enthält, immer vor unbefugtem Zugriff durch Vergabe von Administratorrechten oder Verschlüsselung.

Funktionsweise im Transpondesystem

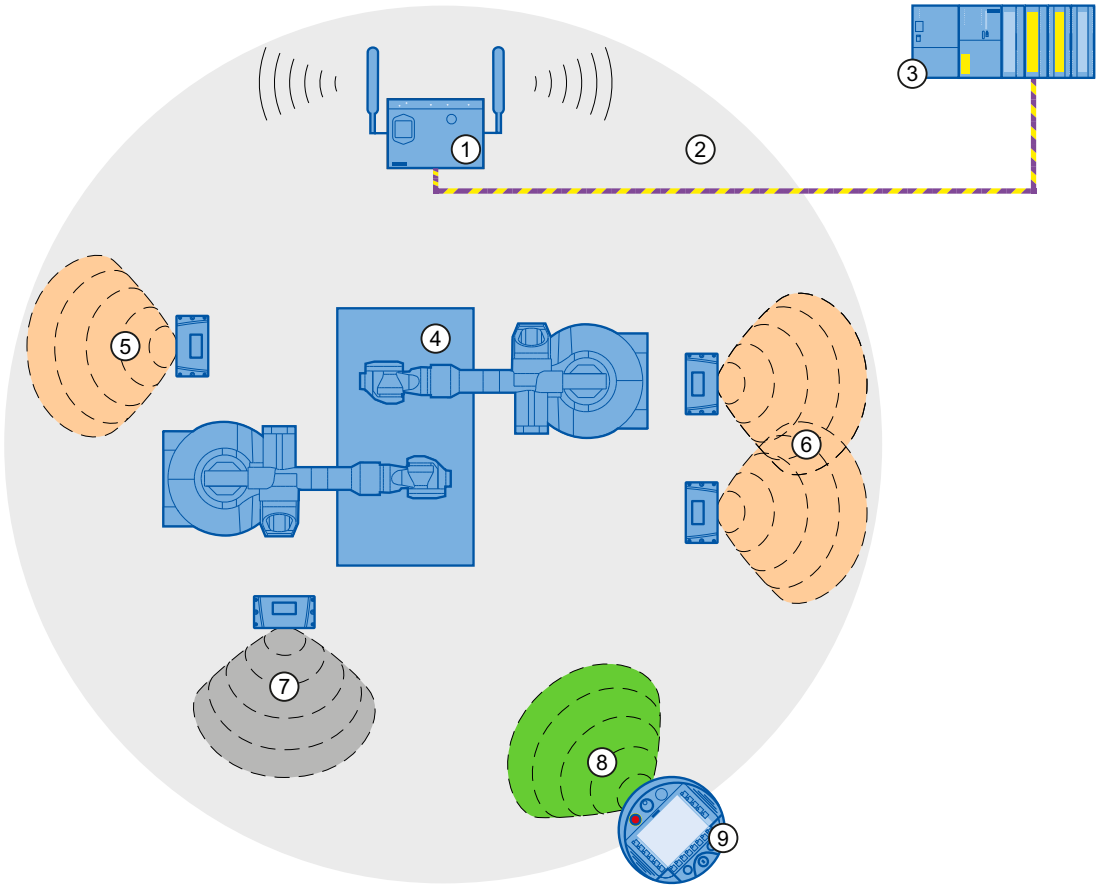

1 Access Point
2 WLAN des Access Point
3 Steuerung
4 Anlage
5 Wirkbereich "Roboter 1" mit einem Transponder
6 Wirkbereich "Roboter 2" mit zwei Transpondern
7 Zone "Zone 1" mit einem Transponder
8 Funkbereich des Bediengeräts
9 Bediengerät

Beispiel:

- Für die Zone "Zone 1" ist ein Bildwechsel projektiert.
- Der Transponder1 ist dem Wirkbereich "Roboter 2" zugeordnet.
- Der Bediener betritt den Funkkegel des Transponders, um die Transponder-Kennung zu empfangen.

- Zur Anmeldung an der Maschine "Roboter 2" drückt der Bediener das Objekt "Wirkbereichs-Name".
- Der Dialog "Wirkbereichs-Anmeldung" erscheint. Der Bediener gibt die Wirkbereich-ID ein und bestätigt mit "OK". Der Dialog wird geschlossen.
- Wenn die Anmeldung erfolgreich war, sind die Zustimmtaster aktiv.
- Das Bediengerät misst die Entfernung zum Transponder, die kleiner ist als der projektierte Abstand des Wirkbereichs.

Ergebnis:

Der Bediener ist an der Maschine "Roboter 2" angemeldet. Das Bedienobjekt "Wirkbereichs-Name" ist grün und zeigt den Namen des Wirkbereichs "Roboter 2".

Wenn das Bediengerät am Wirkbereich angemeldet ist, gilt Folgendes:

- Der Bediener darf den Wirkbereich nicht ohne Abmeldung verlassen. Wenn der Bediener den Wirkbereich ohne Abmeldung für länger als 30 Sekunden verlässt, dann erfolgt ein lokaler Rampdown.
- Es kann sich kein weiteres Bediengerät an der Maschine anmelden.

Um sich von der Maschine abzumelden, drückt der Bediener das Bedienobjekt "Wirkbereichs-Name". Ein Dialog zur Abmeldung wird eingeblendet. Der Bediener bestätigt den Dialog.

Nach erfolgreicher Abmeldung kann sich ein anderer Bediener an der Maschine "Roboter2" anmelden.

Wenn der Bediener die Zone "Zone 1" betritt, findet ein Bildwechsel statt.

10.1.3 Funktionsweise des RFID-Systems

Übersicht

Die Abkürzung RFID steht für Radiofrequenz-Identifikation. RFID bietet die Möglichkeit, Objekte berührungslos und ohne Sichtkontakt per Funk zu identifizieren.

Zum grundlegenden Verständnis beschreibt der Abschnitt folgendes Zusammenspiel:

- Bediengerät
- Wirkbereich (RFID)
- RFID Tag

Hinweis

RFID Tags im Anlagenbereich

RFID Tags dürfen Sie nur in abgegrenzten Anlagenbereichen, die durch zusätzliche Schutzmaßnahmen z. B. Schutzzäune gesichert sind, anbringen.

Funktionsweise im RFID-System

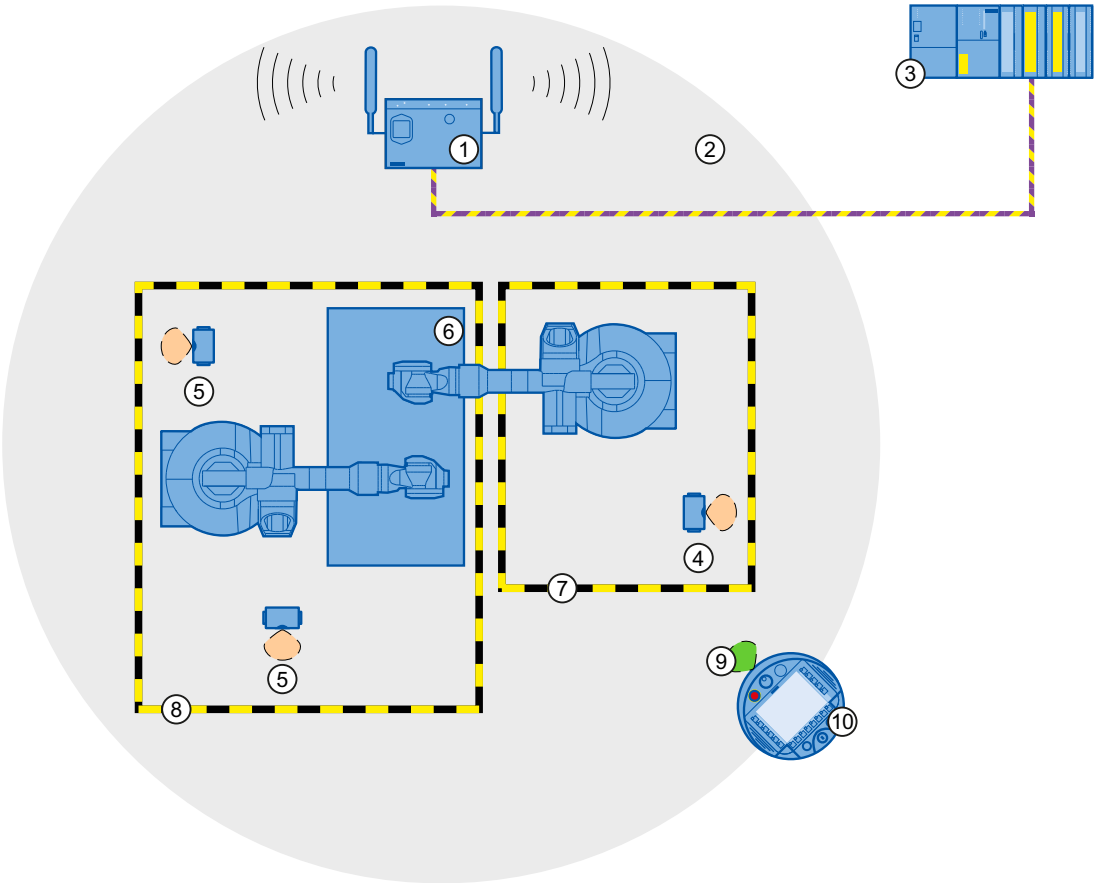

① Access Point
② WLAN des Access Point
③ Steuerung
④ RFID-Tag mit Wirkbereich (RFID) zum Anmelden am "Roboter1"
⑤ RFID-Tag mit Wirkbereich (RFID) zum Anmelden am "Roboter2"
⑥ Anlage
⑦ Schutzbereichsgrenze der Maschine "Roboter1"
⑧ Schutzbereichsgrenze der Maschine "Roboter2"
⑨ Funkbereich des Bediengeräts
⑩ Bediengerät

Beispiel:

- "RFID Tag1" ist dem Wirkbereich (RFID) "Roboter1" zugeordnet.
- Ein Bediener betritt den abgegrenzten Anlagenbereich. Der Bediener wechselt manuell in das Bild mit dem projektierten Bedienobjekt "Wirkbereichs-Name (RFID)".

- Am "RFID Tag1" drückt er die Schaltfläche "Scan".
  Während des Anmeldevorgangs ist die Schaltfläche gelb und zeigt den Text "Scanning…".
- Der Dialog "Wirkbereichs-Anmeldung" erscheint. Der Bediener gibt die Wirkbereich (RFID)-ID ein und bestätigt mit "OK". Der Dialog wird geschlossen.
- Der Dialog "Bestätigung der Anmeldung" erscheint. Der Bediener bestätigt die Anmeldung mit Zustimmtaster.
- Wenn die Anmeldung erfolgreich war, sind die Zustimmtaster aktiv.

Ergebnis:

Der Bediener ist an der Maschine "Roboter1" angemeldet und bedient die Maschine innerhalb des Schutzbereichs im fehlersicheren Betrieb. Das Bedienobjekt "Wirkbereichs-Name (RFID)" ist grün und zeigt den Namen des Wirkbereichs (RFID) "Roboter1".

Wenn das Bediengerät an der Maschine angemeldet ist, gilt Folgendes:

- Der Bediener darf den Schutzbereich nicht verlassen, ohne sich von der Maschine abzumelden. Wenn der Bediener den Schutzbereich ohne Abmeldung verlässt, dann erfolgt ein lokaler Rampdown.
- Es kann sich kein weiteres Bediengerät an dieser Maschine anmelden.

Um sich von der Maschine abzumelden, drückt der Bediener das Bedienobjekt "Wirkbereichs-Name (RFID)". Ein Dialog zur Abmeldung wird eingeblendet. Der Bediener bestätigt den Dialog.

Nach erfolgreicher Abmeldung kann sich ein anderer Bediener an der Maschine "Roboter1" anmelden.

10.1.4 Zonen

Einleitung

Der folgende Abschnitt gilt für folgende Mobile Panels:

- Mobile Panels Wireless z. B. Mobile Panel 277 IWLAN V2 und Mobile Panel 277F IWLAN V2
- KTP Mobile Panels, kabelgebunden, z. B. Mobile Panel KTP700

Nur bei diesen Bediengeräten ist ein Arbeitsbereich "Zonen" sichtbar.

Hinweis

Nähere Hinweise finden Sie in Betriebsanleitung des Bediengeräts:

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

Öffnen

Abhängig vom gewählten Bediengerät, öffnen Sie den Arbeitsbereich im Projektfenster durch einen Doppelklick auf "Zonen" bzw. "Zonen und Wirkbereiche".

Arbeitsbereich

Für Mobile Panels Wireless zeigt der Arbeitsbereich "Zonen" die eingerichteten Zonen und ihre zugeordneten Transponder.

Für kabelgebundene Mobile Panels zeigt der Arbeitsbereich "Zonen" eingerichteten Zonen und die dazugehörige Box-ID der Anschluss-Box.

Inspektorfenster

Wenn eine Zone markiert ist, editieren Sie unter "Allgemein" den Namen und Anzeigenamen.

Bei Mobile Panels Wireless projektieren Sie die Grenze der Zone um den Transponder. Die Zone eines Transponders besitzt die Ereignisse "Beim Eintritt" und "Beim Austritt".

Bei kabelgebundenen Mobile Panels tragen Sie die Box-ID der Anschluss-Box ein. Die Zone einer Anschluss-Box besitzt das Ereignis "Verbunden".

An die Ereignisse projektieren Sie die Systemfunktion "AktiviereBild".

10.1.5 Arbeitsbereich Zonen an Transpondern

Einleitung

Der Arbeitsbereich "Zonen" zeigt tabellarisch die Zonen und ihre Transponder. Sie erstellen eine Liste der Transponder und ordnen einer Zone bestimmte Transponder zu. Die Grenze der Zone definieren Sie durch einen maximalen Abstand zu den Transpondern.

Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Zonen" und "Transponder".

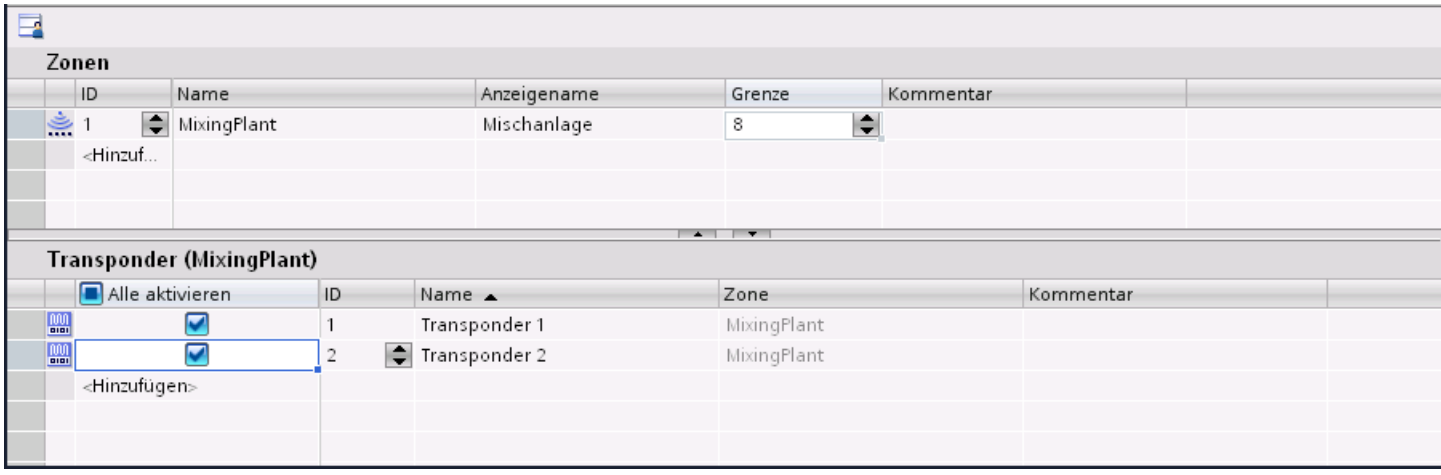

Wenn Sie in der Tabelle "Zonen" eine Zone markieren, zeigt die Tabelle "Transponder" Folgendes:

- Transponder aktiviert: Der Transponder ist der markierten Zone zugeordnet.
- Transponder deaktiviert: Der Transponder ist noch keiner Zone zugeordnet.
- Transponder nicht verfügbar: Der Transponder ist bereits einer anderen Zone zugeordnet. Um die Zuordnung rückgängig zu machen, wechseln Sie zu der betreffenden Zone und deaktivieren Sie den Transponder.

Die IDs der Zonen und Transponder werden zunächst automatisch vergeben. Sie können sie jedoch ändern.

Achten Sie dabei auf Folgendes:

- Maximal 254 Zonen sind projektierbar.
- Die Zonen-ID muss eindeutig sein und aus dem Wertebereich 1-254.
- Sie können zunächst Transponder projektieren, ohne sie einer Zone zuzuordnen.
- Insgesamt existieren höchstens 255 Transponder, die einer Zone zugeordnet sind.
- Die Transponder-ID muss eindeutig sein und aus dem Wertebereich 1-65534.

Die Transponder-ID stellen Sie am Transponder ein.

10.1.6 Arbeitsbereich Zonen an Anschluss-Boxen

Einleitung

Der Arbeitsbereich "Zonen" zeigt tabellarisch die Box-IDs der Anschluss-Boxen und ihre Zonen.

Sie erstellen eine Liste der Box-IDs der Anschluss-Boxen und ordnen einer Zone bestimmte Anschluss-Boxen zu. Die Grenze der Zone wird definiert durch die Länge des Anschlussklabels an IhrMobile Panel.

Prinzip

Der Arbeitsbereich besteht aus der Tabelle "Zonen".

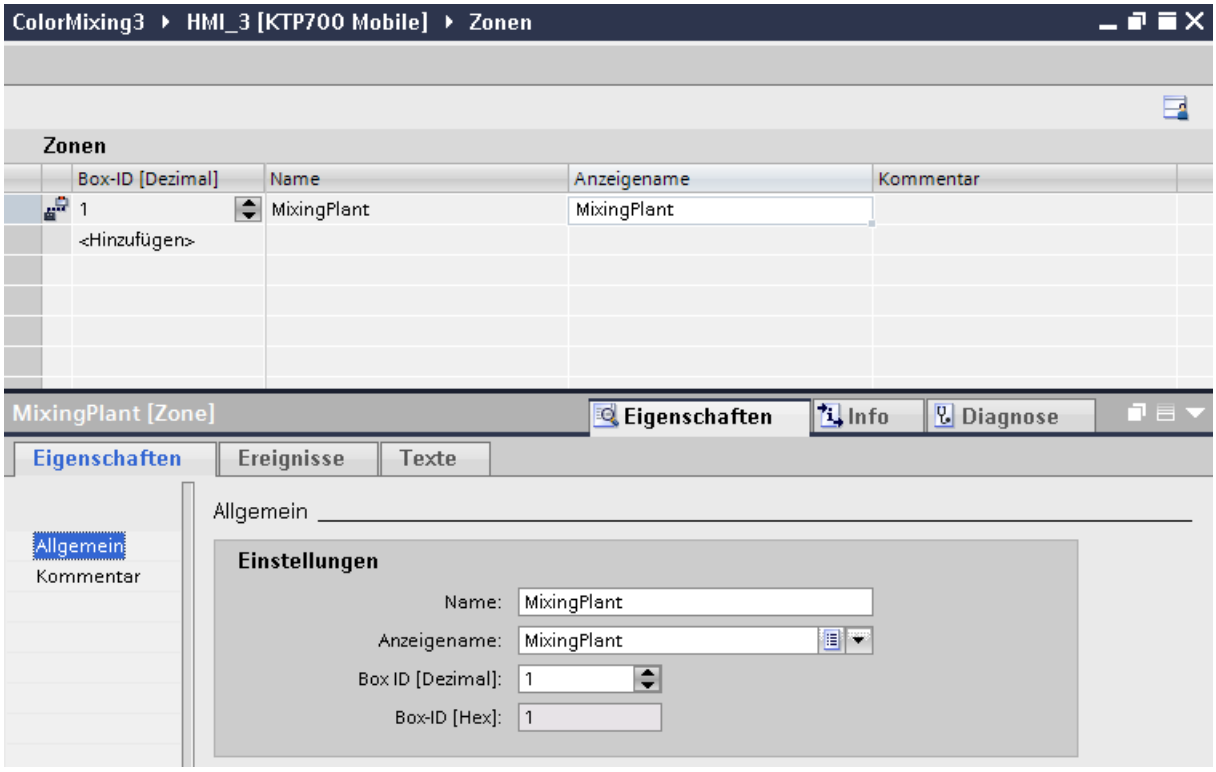

Die Box-ID wird an der Anschluss-Box eingestellt und kann vom Bediengerät ausgelesen werden.

10.1.7 Wirkbereiche

Einleitung

Der folgende Abschnitt gilt nur für die Mobile Panels Wireless, die den fehlersicheren Betrieb unterstützen z. B. Mobile Panel 277F IWLAN V2. Nur bei diesen Bediengeräten ist der Arbeitsbereich "Wirkbereiche" sichtbar.

Um sicherheitsrelevante Bedienungen zu kontrollieren, richten Sie Wirkbereiche ein. Ein Wirkbereich wird definiert durch den maximalen Abstand zu einem oder mehreren Transpondern.

Hinweis

Nähere Hinweise finden Sie in Betriebsanleitung des Bediengeräts:

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

Öffnen

Sie öffnen den Arbeitsbereich im Projektfenster durch einen Doppelklick auf "Zonen & Wirkbereiche". Klicken Sie auf die Registerkarte "Wirkbereiche".

Arbeitsbereich

Der Arbeitsbereich "Wirkbereiche" zeigt die eingerichteten Wirkbereiche und ihre zugeordneten Transponder.

Inspektorfenster

Wenn ein Wirkbereich markiert ist, editieren Sie in der Kategorie "Allgemein" den Namen, Anzeigenamen und die Grenze des Wirkbereichs.

Wenn ein Transponder markiert ist, sehen Sie neben dem Wirkbereich auch die Zone, welcher der Transponder zugeordnet ist. Damit sich die Wirkbereiche nicht überschneiden, können Sie einen Transponder nur einem Wirkbereich zuordnen.

10.1.8 Arbeitsbereich Wirkbereiche

Einleitung

Der Arbeitsbereich "Wirkbereiche" zeigt tabellarisch die Wirkbereiche und ihre Transponder. Sie erstellen eine Liste der Transponder und ordnen einem Wirkbereich bestimmte Transponder zu. Die Grenze des Wirkbereichs definieren Sie durch einen maximalen Abstand zu den Transpondern.

Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Wirkbereiche" und "Transponder".

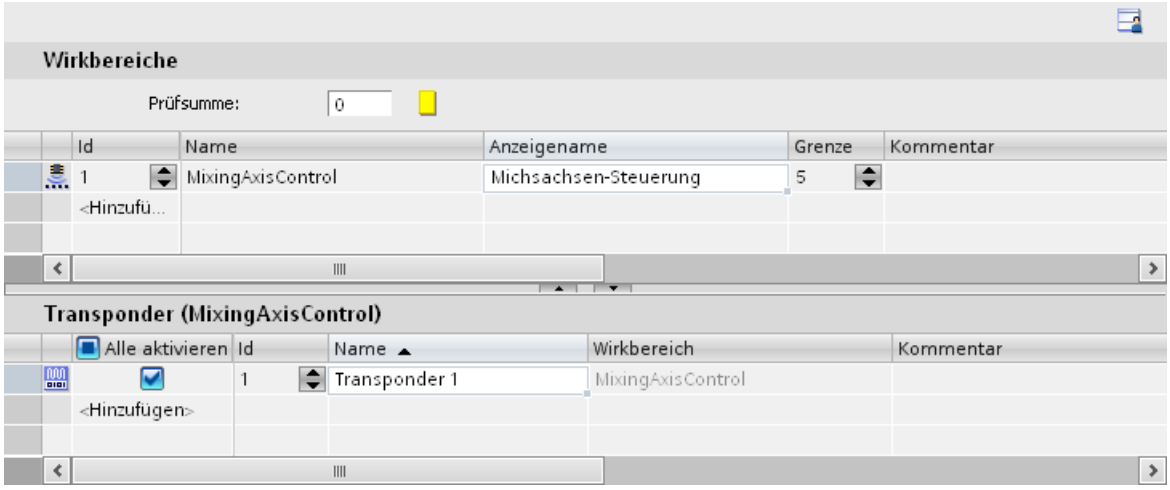

Damit die projektierten Wirkbereiche und Transponder sicher mit denen vor Ort an der Maschine übereinstimmen, berechnet das Bediengerät aus den Daten vor Ort eine Prüfsumme. Am Bediengerät lässt sich das Projekt nur starten, wenn die vor Ort berechnete Prüfsumme mit der im Editor "Wirkbereiche" hinterlegten Prüfsumme übereinstimmt.

Wenn Sie in der Tabelle "Wirkbereiche" einen Wirkbereich markieren, zeigt die Tabelle "Transponder" Folgendes:

- Transponder aktiviert: Der Transponder ist dem markierten Wirkbereich zugeordnet.
- Transponder deaktiviert: Der Transponder ist noch keinem Wirkbereich zugeordnet.
- Transponder nicht verfügbar: Der Transponder ist bereits einem anderen Wirkbereich zugeordnet. Um die Zuordnung rückgängig zu machen, wechseln Sie zu dem betreffenden Wirkbereich und deaktivieren Sie den Transponder.
- Neben dem Wirkbereich wird auch die Zone angezeigt, welcher der Transponder zugeordnet ist.

Die IDs der Wirkbereiche und Transponder werden zunächst automatisch vergeben. Sie können sie jedoch ändern.

Achten Sie dabei auf Folgendes:

- Maximal 127 Wirkbereiche sind projektierbar.
- Die Wirkbereich-ID muss eindeutig sein und aus dem Wertebereich 1-127.
- Der Anzeigename eines Wirkbereichs darf nicht mit seiner ID übereinstimmen.
- Sie können zunächst Transponder projektieren, ohne sie einem Wirkbereich zuzuordnen.
- Insgesamt existieren höchstens 127 Transponder, die einem Wirkbereich zugeordnet sind.
- Die Transponder-ID muss eindeutig sein und aus dem Wertebereich 1-65534.

Die Transponder-ID stellen Sie am Transponder ein.

Wirkbereich in Runtime

Um sich in Runtime am Wirkbereich anzumelden, wird dem Bediener in der Runtime-Sprache nur der Anzeigename des Wirkbereichs angezeigt. Der Bediener liest in der Anlage die Wirkbereich-ID ab und gibt sie am Bediengerät ein. Dadurch wird sichergestellt, dass die richtige Maschine bedient wird. Nach erfolgreicher Anmeldung sind die Zustimmtaster bedienbar.

10.1.9 Wirkbereiche (RFID)

Einleitung

Der folgende Abschnitt gilt nur für das Mobile Panel 277F IWLAN (RFID Tag), das den fehlersicheren Betrieb mit RFID unterstützt. Nur bei diesen Bediengeräten ist der Arbeitsbereich "Wirkbereiche (RFID)" sichtbar.

Um sicherheitsrelevante Bedienungen zu kontrollieren, richten Sie Wirkbereiche (RFID) ein.

Hinweis

Nähere Hinweise finden Sie in Betriebsanleitung des Bediengeräts:

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

Öffnen

Sie öffnen den Arbeitsbereich im Projektfenster durch einen Doppelklick auf "Wirkbereiche (RFID)".

Arbeitsbereich

Der Arbeitsbereich "Wirkbereiche (RFID)" zeigt die eingerichteten Wirkbereiche und ihre zugeordneten RFID Tags.

Inspektorfenster

Wenn ein Wirkbereich (RFID) markiert ist, editieren Sie in der Kategorie "Allgemein" den Namen, Anzeigenamen und ID des Wirkbereichs.

10.1.10 Arbeitsbereich Wirkbereiche (RFID)

Einleitung

Der Arbeitsbereich "Wirkbereiche (RFID)" zeigt tabellarisch die Wirkbereiche (RFID) und ihre RFID Tags. Sie erstellen eine Liste der RFID Tags und ordnen einem Wirkbereich bestimmte RFID Tags zu.

Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Wirkbereiche (RFID)" und "RFID Tags".

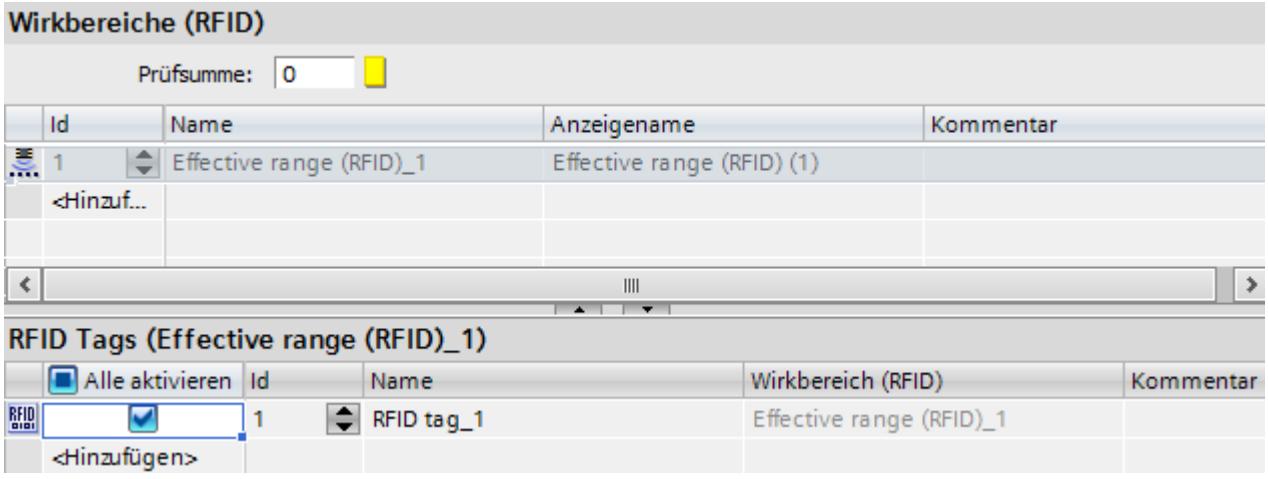

Damit die projektierten Wirkbereiche (RFID) und RFID Tags sicher mit denen vor Ort an der Maschine übereinstimmen, berechnet das Bediengerät aus den Daten vor Ort eine Prüfsumme. Am Bediengerät lässt sich das Projekt nur starten, wenn die vor Ort berechnete Prüfsumme mit der im Editor "Wirkbereiche (RFID)" hinterlegten Prüfsumme übereinstimmt.

Wenn Sie in der Tabelle "Wirkbereiche (RFID)" einen Wirkbereich (RFID) markieren, zeigt die Tabelle "RFID Tags" Folgendes:

- RFID Tag aktiviert: Das RFID Tag ist dem markierten Wirkbereich (RFID) zugeordnet.
- RFID Tag deaktiviert: Das RFID Tag ist noch keinem Wirkbereich (RFID) zugeordnet.
- RFID Tag nicht verfügbar: Das RFID Tag ist bereits einem anderen Wirkbereich (RFID) zugeordnet. Um die Zuordnung rückgängig zu machen, wechseln Sie zu dem betreffenden Wirkbereich (RFID) und deaktivieren Sie das RFID Tag.

Die IDs der Wirkbereiche (RFID) und RFID Tags werden zunächst automatisch vergeben. Sie können sie jedoch ändern.

Achten Sie dabei auf Folgendes:

- Maximal 127 Wirkbereiche (RFID) sind projektierbar.
- Die Wirkbereich (RFID)-ID muss eindeutig sein und aus dem Wertebereich 1-127.
- Der Anzeigename eines Wirkbereichs (RFID) darf nicht mit seiner ID übereinstimmen.

- Sie können zunächst RFID Tags projektieren, ohne sie einem Wirkbereich (RFID) zuzuordnen.
- Insgesamt existieren höchstens 127 RFID Tags, die einem Wirkbereich (RFID) zugeordnet sind.
- Die RFID Tag-ID muss eindeutig sein und aus dem Wertebereich 1-65534.

Die RFID Tag-ID wird mit dem Bediengerät in das RFID-Tag geschrieben.

Wirkbereich in Runtime

Um sich in Runtime am Wirkbereich (RFID) anzumelden, wird dem Bediener in der Runtime-Sprache nur der Anzeigename des Wirkbereichs (RFID) angezeigt. Der Bediener liest in der Anlage die Wirkbereich (RFID)-ID ab und gibt sie am Bediengerät ein. Dadurch wird sichergestellt, dass die richtige Maschine bedient wird. Nach erfolgreicher Anmeldung sind die Zustimmtaster bedienbar.

10.2 Projektierung der Mobile Panels V2 für den fehlersicheren Betrieb (ab V12 SP1)

10.2.1 Projektierungsübersicht

Einleitung

Für folgende Mobile Panels V2 projektieren Sie im TIA Portal ab Version V12 SP1 den fehlersicheren Betrieb im Editor "Geräte & Netze":

- Mobile Panel 277F IWLAN V2
- Mobile Panel 277F IWLAN (RFID Tag)

Handlungsübersicht

1. PLC für den fehlersicheren Betrieb anlegen.
2. Mobile Panel V2 für den fehlersicheren Betrieb anlegen.
3. Verbindung zwischen dem Mobile Panel und PLC herstellen.

Siehe auch

Mobile Panel anlegen (Seite 1849)

PLC für den fehlersicheren Betrieb anlegen (Seite 1848)

Verbindung zwischen Mobile Panel und PLC herstellen (Seite 1850)

10.2.2 PLC für den fehlersicheren Betrieb anlegen

Voraussetzung

Ein Projekt ist geöffnet

Vorgehen

1. Klicken Sie auf der Projektnavigation auf "Neues Gerät hinzufügen".
2. Klicken Sie im Dialog "Neues Gerät hinzufügen" auf "Controller".
3. Wählen Sie eine PLC für den fehlersicheren Betrieb, z. B. CPU 315F-2 PN/DP.
   Der Editor "Geräte & Netze" wird geöffnet.
4. Selektieren Sie das Gerät in der "Geräteübersicht".
5. Klicken Sie im Inspektorfenster "Eigenschaften > Allgemein > PROFINET-Schnittstelle [X2] > Ethernet-Adressen > Schnittstelle vernetzt mit > Neues Subnetz hinzufügen".

6. Vergeben Sie die IP-Adresse unter "IP-Protokoll > IP-Adresse im Projekt einstellen".
7. Geben Sie für "Eigenschaften > PROFINET-Schnittstelle [X2] > F-Parameter > Default F-Überwachungszeit für F-Peripherie dieser Schnittstelle" eine Zeit ein.
8. Legen Sie unter "Eigenschaften > Fehlersicherheit" die Eigenschaften für den fehlersicheren Betrieb fest:
   – "F-Fähigkeit aktiviert"
   – "Basis für PROFIsafe-Adressen"
9. Legen Sie die weiteren Eigenschaften für den fehlersicheren Betrieb wie gewohnt fest, z. B. Schutzstufe.

Ergebnis

Sie haben eine PLC im Projekt angelegt und die Einstellungen für den fehlersicheren Betrieb vorgenommen.

Siehe auch

Projektierungsübersicht (Seite 1848)

10.2.3 Mobile Panel anlegen

Voraussetzung

- Das Inspektorfenster ist geöffnet.
- Eine PLC für den fehlersicheren Betrieb ist im Projekt angelegt z.B. CPU-315F-2PN/DP.

Mobile Panel im Projekt einfügen.

1. Klicken Sie auf der Projektnavigation auf "Neues Gerät hinzufügen".
2. Klicken Sie im Dialog "Neues Gerät hinzufügen" auf "HMI".
3. Wählen Sie ein Mobile Panel V2 für den fehlersicheren Betrieb, z. B. Mobile Panel 277F IWLAN V2.
   Das Gerät wird im Projekt eingefügt.

Eigenschaften festlegen

1. Öffnen Sie den Editor "Geräte und Netze".
2. Markieren Sie das Mobile Panel im Arbeitsbereich.
3. Öffnen Sie im Inspektorfenster "Eigenschaften > Allgemein".

4. Um eine leichtere Zuordnung im Projekt zu ermöglichen, geben Sie unter "Name" den Namen des zugehörigen Moduls an.
5. Vergeben Sie die IP-Adresse unter "Ethernet-Adressen > IP-Protokoll > IP-Adresse im Projekt einstellen".

Ergebnis

Sie haben im Projekt ein Mobile Panel für den fehlersicheren Betrieb angelegt.

Siehe auch

Projektierungsübersicht (Seite 1848)

10.2.4 Verbindung zwischen Mobile Panel und PLC herstellen

Voraussetzungen

- Eine PLC für den fehlersicheren Betrieb ist im Projekt angelegt z.B. CPU-315F-2PN/DP.
- Ein Gerät ist angelegt.
- Der Editor "Geräte und Netze" ist geöffnet.

Vorgehen

1. Öffnen Sie die "Netzsicht".
2. Markieren Sie das Mobile Panel im Arbeitsbereich.
3. Aktivieren Sie im Inspektorfenster "Eigenschaften > Allgemein > Betriebsart > IO-Device".
4. Weisen Sie unter "Zugewiesener IO-Controller" die fehlersichere PLC zu.
   PROFINET ist aktiviert und die Transferbereiche der I-Device-Kommunikation werden angezeigt.
5. Klicken Sie auf die Schaltfläche "Verbindungen" und wählen Sie den Verbindungstyp "HMI-Verbindung".
   Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.
6. Klicken Sie in die Schnittstelle der PLC und ziehen Sie per Drag&Drop eine Verbindung zur Schnittstelle des Bediengeräts.
   HMI-Verbindung wird erstellt.
7. Markieren Sie das Mobile Panel im Arbeitsbereich.
8. Konfigurieren Sie die F-Parameter unter "Eigenschaften > Allgemein > F-Parameter".
   Um eine andere als von der PLC vorgegebene F-Überwachungszeit einzustellen, aktivieren Sie "Manuelle Vergabe der F-Überwachungszeit".
   Geben Sie eine für Ihr Bediengerät und Ihre Anlage passende F-Überwachungszeit ein.

Ergebnis

Das Bediengerät ist mit der PLC verbunden und die Einstellungen für den fehlersicheren Betrieb vorgenommen.

Siehe auch

Projektierungsübersicht (Seite 1848)

10.3 Projektierung der KTP Mobile Panels für den fehlersicheren Betrieb

10.3.1 KTP Mobile Panel für den fehlersicheren Betrieb anlegen

Einleitung

Für folgende KTP Mobile Panels projektieren Sie den fehlersicheren Betrieb im Editor "Geräte & Netze":

- KTP400F Mobile Panel
- KTP700F Mobile Panel
- KTP900F Mobile Panel

Voraussetzung

- Das Inspektorfenster ist geöffnet.
- Eine PLC für den fehlersicheren Betrieb ist im Projekt angelegt, z. B. CPU-1517F-2PN/DP.

Mobile Panel im Projekt einfügen

1. Klicken Sie auf der Projektnavigation auf "Neues Gerät hinzufügen".
2. Klicken Sie im Dialog "Neues Gerät hinzufügen" auf "HMI".
3. Wählen Sie ein KTP Mobile Panel für den fehlersicheren Betrieb, z. B. Mobile Panel KTP700F. Das Gerät wird im Projekt eingefügt.

Fehlersicheren Betrieb für das Mobile Panel projektieren

1. Öffnen Sie im Editor "Geräte und Netze" die Netzsicht.
2. Markieren Sie das Mobile Panel im Arbeitsbereich.
3. Aktivieren Sie im Inspektorfenster "Eigenschaften > Allgemein > Betriebsart > IO-Device".
4. Weisen Sie unter "Zugewiesener IO-Controller" die fehlersichere PLC zu.
   PROFINET ist aktiviert und die Transferbereiche der I-Device-Kommunikation werden angezeigt.
5. Klicken Sie im Inspektorfenster "Eigenschaften > Allgemein > PROFIsafe" auf "PROFIsafe aktivieren"
   PROFIsafe ist aktiviert und unter "Eigenschaften > Allgemein > PROFIsafe Parameter > F-Parameter" werden die F-Parameter angezeigt.
6. Um eine neue F-Überwachungszeit einzugeben, aktivieren Sie "Manuelle Vergabe der F-Überwachungszeit".
7. Geben Sie eine für Ihr Bediengerät und Ihre Anlage passende F-Überwachungszeit ein.

Ergebnis

Der fehlersichere Betrieb für das KTP Mobile Panel ist in WinCC projektiert. Erstellen Sie nun in STEP 7 die entsprechenden F-Bausteine und verschalten Sie diese in der Steuerung. Am Mobile Panel muss die PROFIsafe-Adresse eingestellt werden.

Weitere Hinweise zur Parametrierung und F-Bausteinprogrammierung finden Sie im Handbuch "SIMATIC Safety - Projektieren und Programmieren" der STEP 7 Dokumentation und im Gerätehandbuch Ihres Mobile Panels.

10.3.2 Abbau der PROFIsafe Verbindung projektieren

Vorgehen

Bevor Sie Ihr Mobile Panel von einer Anschluss-Box abstecken, müssen Sie die Verbindung zu PROFIsafe trennen.

Um die PROFIsafe-Verbindung zu trennen, verwenden Sie die Systemfunktion "PROFIsafeTrennen". Die Systemfunktion leitet Ihren Aufruf an das fehlersichere PROFINET I/O Modul weiter.

Voraussetzungen

- Ein Mobile Panel für den fehlersicheren Betrieb ist angelegt, z. B. KTP700F Mobile Panel.
- Ein Bild ist angelegt und geöffnet.

Vorgehen

Um den PROFIsafe Verbindungsaufbau und Verbindungsabbau zu projektieren, gehen Sie folgendermaßen vor:

- Projektieren Sie Systemfunktion "PROFIsafeTrennen", z. B. an eine Schaltfläche oder verwenden Sie die Systemfunktion in einem Skript.

Ergebnis

Wenn Sie in Runtime den projektieren Trigger zum Trennen der PROFIsafe-Verbindung auslösen, wird ein Dialog zum Beenden der PROFIsafe-Verbindung geöffnet. Benutzer hat die Möglichkeit "Ja" oder "Nein" auszuwählen.

10.4 Arbeiten mit Zonen an Transpondern

10.4.1 Zone im Transpondersystem projektieren

Gültigkeit

Der folgende Abschnitt gilt für alle Mobile Panels Wireless z. B. Mobile Panel 277 IWLAN.

Einleitung

Um anlagenspezifisch zu bedienen und zu beobachten, richten Sie eine Zone ein.

Voraussetzungen

- Im Editor "Projektsprachen" ist die geeignete Bearbeitungssprache eingestellt z. B. "Deutsch".
- Der Arbeitsbereich "Zonen" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Zonen" auf "Hinzufügen…".
2. Geben Sie für "Name" der Zone "MixingPlant" ein.
3. Geben Sie für "Anzeigename" der Zone in Runtime "Mischanlage" ein. Geben Sie für "ID" der Zone "1" ein.
4. Geben Sie für "Grenze" der Zone "8" Meter ein.
5. Doppelklicken Sie in der Tabelle "Transponder" auf "Hinzufügen…".
6. Geben Sie für "Name" des Transponders "Transponder 1" ein.
7. Geben Sie für "ID" des Transponders "1" ein.
   Die Transponder-ID stellen Sie dann am Transponder ein.
8. Aktivieren Sie die Zuordnung von "Transponder 1".
   "Transponder 1" ist der Zone "MixingPlant" zugeordnet.
9. Wiederholen Sie die Schritte 6-9 entsprechend für "Transponder 2".
   "Transponder 2" ist der Zone "MixingPlant" zugeordnet.

Ergebnis

Unabhängig von den Wirkbereichen ist die Zone "MixingPlant" projektiert als zwei kegelförmige Bereiche mit Abstand 8 Meter: einen Bereich um "Transponder 1" und einen um "Transponder 2".

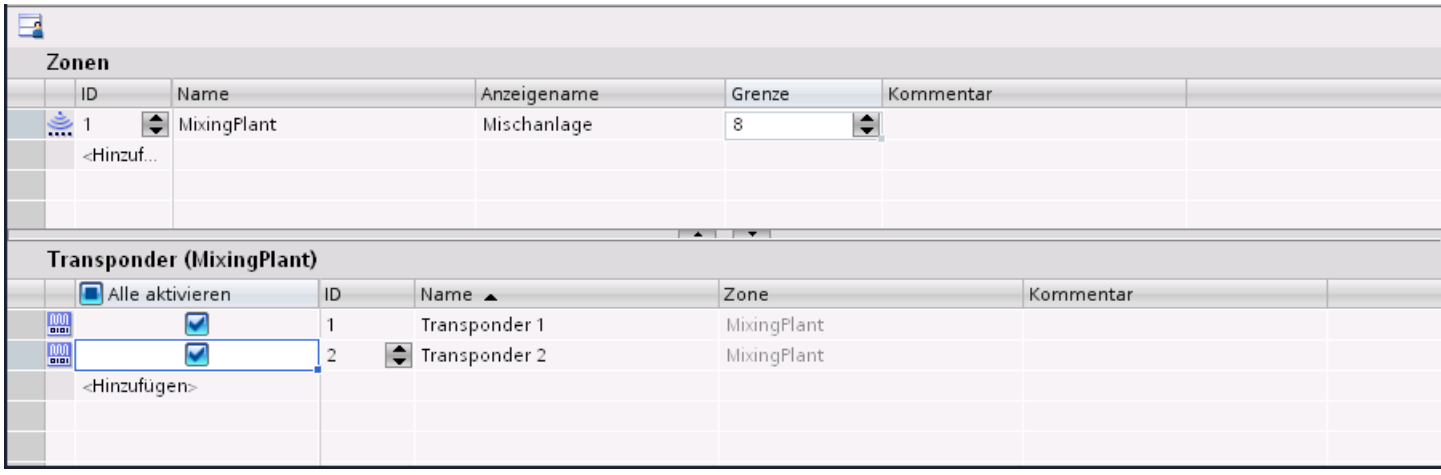

Der Anzeigename bezeichnet die Zone in der eingestellten Bearbeitungssprache. Eine Übersicht über die Anzeigenamen finden Sie im Editor "Projekttexte".

10.4.2 Bild bei Zoneneintritt anzeigen

Einleitung

Sie projektieren einen Bildwechsel beim Eintritt und Austritt aus der Zone.

Voraussetzungen

- Ein Mobile Panel, das Zonen unterstützt, ist angelegt z. B. Mobile Panel 277 IWLAN.
- Die Zone "MixingPlant" ist projektiert.
- Das Bild "Plant_1" ist angelegt und geöffnet.
- Das Bild "MixingPlant_1" ist angelegt und geöffnet..
- Der Arbeitsbereich "Zonen" ist geöffnet.

Vorgehensweise

1. Markieren Sie in der Tabelle "Zonen" die Zone "MixingPlant".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Klicken Sie auf das Ereignis "Beim Eintritt".
4. Wählen Sie in der "Funktionsliste" die Systemfunktion "AktiviereBild".

5. Wählen Sie als "Bildname" "MixingPlant_1".
6. Klicken Sie auf das Ereignis "Beim Austritt".
7. Wählen Sie in der Funktionsliste die Systemfunktion "AktiviereBild".
8. Wählen Sie als "Bildname" "Plant_1".

Ergebnis

Beim Eintritt in die Zone "MixingPlant", wird am Display des Bediengeräts das Bild "MixingPlant_1" angezeigt. Beim Austritt aus der Zone erscheint wieder das Bild "Plant_1".

10.4.3 Objekt abhängig von der Zone anzeigen

Gültigkeit

Das folgende Beispiel ist unabhängig von WLAN und gilt für alle Mobile Panels.

Einleitung

Sie projektieren ein E/A-Feld, das nur innerhalb einer bestimmten Zone sichtbar ist.

Voraussetzungen

- Ein Mobile Panel, das Zonen unterstützt, ist angelegt z.B. Mobile Panel 277 IWLAN.
- Die Variable "Zone_id" vom Datentyp "Int" ist angelegt.
- Ein Bild "Plant_1_Overview" ist angelegt.

Vorgehensweise

1. Öffnen Sie den Editor "Runtime-Einstellungen > Allgemein".
2. Wählen Sie als "ID Zone/Wirkbereich" die Variable "Zone_id".
3. Öffnen Sie das Bild "Plant_1_Overview".
4. Ziehen Sie per Drag&Drop das Objekt "EA-Feld" aus dem Werkzeugfenster in das Bild.
5. Klicken Sie im Inspektorfenster auf "Eigenschaften > Animationen".
6. Klicken Sie auf "Darstellung > Sichtbarkeit".
7. Wählen Sie als "Variable" "Zone_id".
8. Wählen Sie als "Bereich" "von" 1 "bis" 1.
9. Aktivieren Sie "sichtbar".

Ergebnis

Das EA-Feld ist in Runtime nur unter folgenden Bedingungen sichtbar:

- Das kabelgebundene Mobile Panel ist an eine Anschluss-Box mit der ID = 1 angeschlossen.
- Das Mobile Panel Wireless befindet sich in einer Zone mit der ID = 1 ein z. B. die Zone "MixingPlant".

10.5 Arbeiten mit Zonen an Anschluss-Boxen

10.5.1 Zone einer Anschluss-Box projektieren

Gültigkeit

Der folgende Abschnitt gilt für alle KTP Mobile Panels z. B. Mobile Panel KTP700.

Einleitung

Um anlagenspezifisch zu bedienen und zu beobachten, richten Sie eine Zone ein.

Voraussetzungen

- Im Editor "Projektsprachen" ist die geeignete Bearbeitungssprache eingestellt z. B. "Deutsch".
- Der Arbeitsbereich "Zonen" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Zonen" auf "Hinzufügen…".
2. Geben Sie für "ID" die Box-ID Ihrer Anschluss-Box ein.
3. Geben Sie für "Name" der Zone "MixingPlant" ein.

4. Geben Sie für "Anzeigename" der Zone in Runtime "Mischanlage" ein.

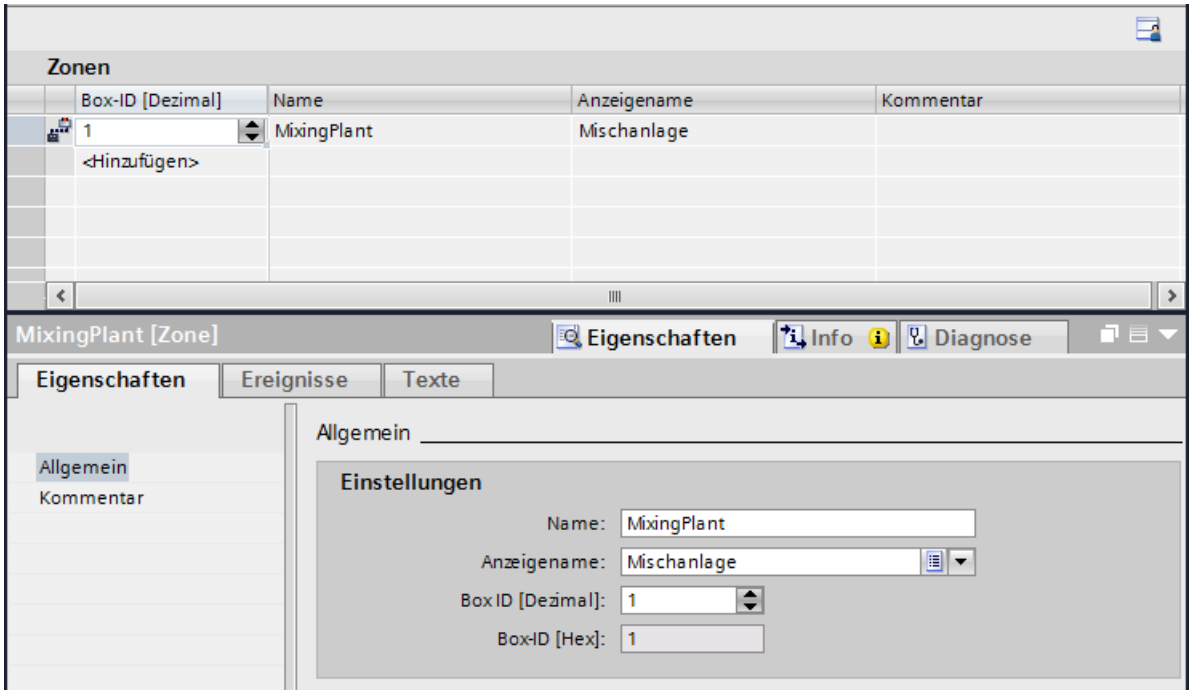

5. Wiederholen Sie diese Schritte für alle benötigten Anschluss-Boxen im Projekt.

Ergebnis

Die Zone "MixingPlant" ist projektiert. Die Reichweite der Zone ist von der Länge des Anschlusskabels des Mobile Panels definiert.

Der Anzeigename bezeichnet die Zone in der eingestellten Bearbeitungssprache. Eine Übersicht über die Anzeigenamen finden Sie im Editor "Projekttexte".

10.5.2 Systemfunktionen bei der Verbindung zur Anschluss-Box aufrufen

Einleitung

Sie haben die Möglichkeit, beim Eintritt in eine Zone verschiedene Systemfunktionen auszulösen, z.B. ein projektiertes Pop-up-Bild aufzurufen oder den aktuellen Benutzer ein- und abzumelden. Zusätzlich können Sie benutzerdefinierte VB-Funktionen verwenden, die Sie auch aus der Funktionsliste auswählen können.

Im folgenden Beispiel projektieren Sie das Bild, das bei Verbindung eines Mobile Panels mit einer Anschluss-Box angezeigt wird.

Voraussetzungen

- Ein Mobile Panel, das Zonen unterstützt, ist angelegt, z. B. Mobile Panel KTP700.
- Die Zone "MixingPlant" ist projektiert.
- Das Bild "Plant_1" ist angelegt und geöffnet.
- Das Bild "MixingPlant_1" ist angelegt und geöffnet..
- Der Arbeitsbereich "Zonen" ist geöffnet.

Vorgehensweise

1. Markieren Sie in der Tabelle "Zonen" die Zone "MixingPlant".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Klicken Sie auf das Ereignis "Verbunden".
4. Wählen Sie in der "Funktionsliste" die Systemfunktion "AktiviereBild".
5. Wählen Sie als "Bildname" "MixingPlant_1".

Ergebnis

Wenn Sie Ihr Mobile Panel mit der Anschluss-Box der Zone "MixingPlant" verbinden, wird am Display des Bediengeräts das Bild "MixingPlant_1" angezeigt.

10.6 Arbeiten mit Wirkbereichen (Transponder)

10.6.1 Übersicht

Einleitung

Der folgende Projektierungsleitfaden beschreibt, welche Schritte ein Projekteur unternimmt, um an einem Mobile Panel Wireless für den fehlersicheren Betrieb einen Wirkbereich einzurichten.

Handlungsübersicht

1. Wirkbereich projektieren:
   Sie projektieren den Wirkbereich "MixingAxisControl" als einen kegelförmigen Bereich um den "Transponder 1" mit Abstand 5 Meter.
2. Wirkbereichs-Name projektieren:
   Damit eine Anmeldung am Wirkbereich stattfinden kann, projektieren Sie das Objekt "Wirkbereichs-Name".
3. Weitere Wirkbereich-Objekte projektieren:
   Sie projektieren weitere Objekte, um die Position und Signalstärke in einem Wirkbereich anzuzeigen.
4. Parameter für das Laden einstellen:
   - PROFIsafe-Kommunikation
   - WLAN-Funknetz
   - Powermanagement
   - Schnittstelle
5. Datenkanal parametrieren
6. Netzwerkbetrieb konfigurieren
7. Transponder einstellen
8. Wirkbereiche in Betrieb nehmen
9. Bediengerät einschalten und testen
10. Laden manuell starten
11. Wirkbereiche an der Anlage abnehmen:
    Sie quittieren die Wirkbereiche mit ihren Transpondern.
12. Prüfsumme ermitteln

13. Projekt mit Prüfsumme erneut laden:
    Sie tragen im Projekt die ermittelte Prüfsumme ein und laden das Projekt erneut.
14. Wirkbereich testen

Hinweis

Nähere Hinweise zu den Punkten 5 bis 14 finden Sie in der Betriebsanleitung des Bediengeräts.

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

10.6.2 Wirkbereich (Transponder) projektieren

Einleitung

Um sicherheitsrelevante Bedienungen vor einer Maschine oder Anlage zu kontrollieren, richten Sie einen Wirkbereich ein.

Voraussetzungen

- Im Editor "Projektsprachen" ist die geeignete Bearbeitungssprache eingestellt z. B. "Deutsch".
- Der Arbeitsbereich "Zonen & Wirkbereiche" ist geöffnet.
- Die Registerkarte "Wirkbereiche" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Wirkbereiche" auf "Hinzufügen...".
2. Geben Sie für "Name" des Wirkbereichs "MixingAxisControl" ein.
3. Geben Sie für "Anzeigename" des Wirkbereichs in Runtime "Mischachsen-Steuerung" ein.
4. Geben Sie für "ID" des Wirkbereichs "1" ein.
   Achten Sie darauf, dass die Wirkbereich-ID mit der Kennung an der Maschine übereinstimmt.
5. Geben Sie für "Grenze" des Wirkbereichs "5" Meter ein.
6. Doppelklicken Sie in der Tabelle "Transponder" auf "Hinzufügen...".
7. Geben Sie für "Name" des Transponders "Transponder 1" ein.
8. Geben Sie für "ID" des Transponders "1" ein.
   Die Transponder-ID stellen Sie dann am Transponder ein.
9. Aktivieren Sie die Zuordnung von "Transponder 1".
   "Transponder 1" ist dem Wirkbereich "MixingAxisControl" zugeordnet.

Ergebnis

Unabhängig von den Zonen ist der Wirkbereich "MixingAxisControl" projektiert als ein kegelförmiger Bereich um den "Transponder 1" mit Abstand 5 Meter.

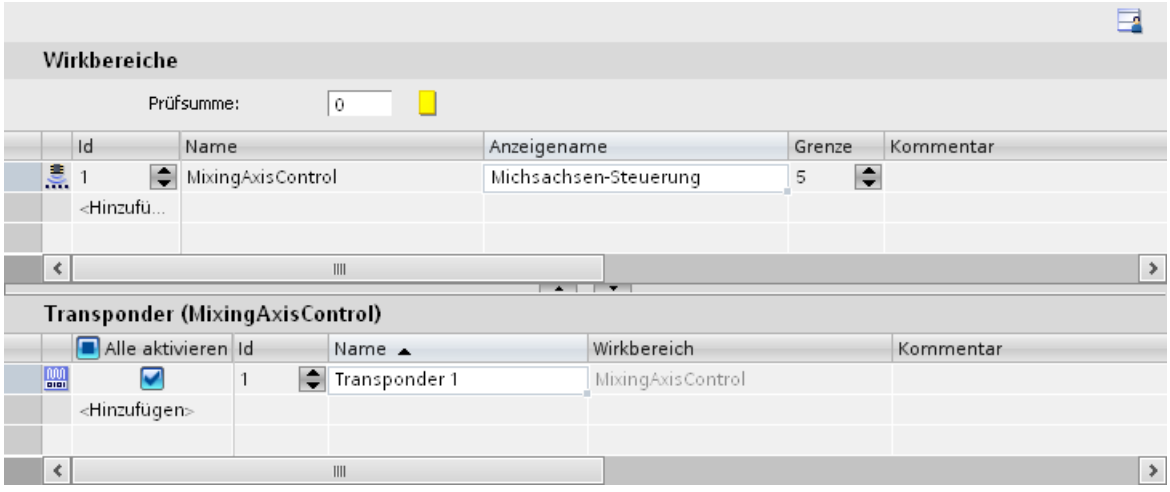

Der Anzeigename bezeichnet den Wirkbereich in der eingestellten Bearbeitungssprache. Eine Übersicht über die Anzeigenamen finden Sie im Editor "Projekttexte".

10.6.3 Wirkbereichs-Name projektieren

Einleitung

Damit sich ein Bediener über einen Wirkbereich an einer Maschine anmelden und abmelden kann, projektieren Sie mindestens das Objekt "Wirkbereichs-Name". Wenn Sie bei einem Mobile Panel Wireless für den fehlersicheren Betrieb kein Wirkbereich-Objekt projektieren, erscheint beim Generieren eine Warnung.

Voraussetzungen

Ein Bild ist angelegt und geöffnet.

Vorgehensweise

1. Markieren Sie im Werkzeugfenster in der Kategorie "Elemente" das Objekt "Wirkbereichs-Name".
2. Ziehen Sie mit Drag&Drop das Objekt "Wirkbereichs-Name" aus dem Werkzeugfenster in das Bild.

Ergebnis

Wenn der Bediener in Runtime auf das Objekt klickt, startet er das Anmelden oder Abmelden des Bediengeräts an einer Maschine, die einem Wirkbereich zugeordnet ist.

10.6.4 Weitere Objekte der Mobile Wireless projektieren

Einleitung

Um weitere Informationen über den Wirkbereich anzuzeigen, projektieren Sie weitere Objekte, z.B. Ladezustand oder WLAN-Empfang.

Voraussetzungen

- Ein Bild ist angelegt und geöffnet.
- Wirksbereichs-Name ist im Bild projektiert.

Vorgehensweise

1. Markieren Sie im Werkzeugfenster in der Kategorie "Elemente" das Objekt "Wirkbereichs-Signal".
2. Ziehen Sie mit Drag&Drop das Objekt "Wirkbereichs-Signal" aus dem Werkzeugfenster in das Bild.
3. Wiederholen Sie die Schritte 1 und 2 mit dem Objekt "WLAN-Empfang".

Ergebnis

Das Objekt "Wirkbereichs-Signal" zeigt an, wie gut sich das Mobile Panel Wireless in einem Wirkbereich befindet. Das Objekt "WLAN-Empfang" Qualität" zeigt die Signalstärke der WLAN-Funkverbindung an.

Siehe auch

WLAN-Empfang (Seite 531)

Wirkbereichs-Signal (Seite 529)

Wirkbereichs-Name (Seite 525)

10.7 Arbeiten mit Wirkbereichen (RFID)

10.7.1 Übersicht

Gültigkeit

Das folgende Kapitel gilt nur für Mobile Panels Wireless mit RFID, z. B.
Mobile Panel 277F IWLAN (RFID Tag).

Einleitung

Der folgende Projektierungsleitfaden beschreibt, welche Schritte ein Projekteur unternimmt, um an einem Mobile Panel Wireless mit RFID für den fehlersicheren Betrieb einen Wirkbereich (RFID) einzurichten.

Handlungsübersicht

1. Wirkbereich (RFID) projektieren:
   Sie projektieren den Wirkbereich (RFID) "MixingAxisControl".
2. Wirkbereichs-Name (RFID) projektieren:
   Damit eine Anmeldung am Wirkbereich (RFID) stattfinden kann, projektieren Sie das Objekt "Wirkbereichs-Name (RIFD)".
3. Parameter für das Laden einstellen
4. Datenkanal parametrieren
5. Netzwerkbetrieb konfigurieren
6. RFID Tag in der Anlage installieren
7. Bediengerät einschalten und Lademodus aktivieren
8. Projekt laden
9. RFID Tags in der Anlage mit entsprechender ID beschreiben
10. Prüfsumme ermitteln
11. Projekt mit Prüfsumme erneut laden:
    Sie tragen im Projekt die ermittelte Prüfsumme ein und laden das Projekt erneut.
12. Wirkbereich (RFID) testen

Hinweis

Nähere Hinweise zu den Punkten 4 bis 12 finden Sie in der Betriebsanleitung des Bediengeräts.

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

10.7.2 Wirkbereich (RFID) projektieren

Einleitung

Um sicherheitsrelevante Bedienungen vor einer Maschine oder Anlage zu kontrollieren, richten Sie einen Wirkbereich (RFID) ein.

Voraussetzungen

- Im Editor "Projektsprachen" ist die geeignete Bearbeitungssprache eingestellt z. B. "Deutsch".
- Der Arbeitsbereich "Wirkbereiche (RFID)" ist geöffnet.

Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Wirkbereiche (RFID)" auf "Hinzufügen...".
2. Geben Sie als "Name" des Wirkbereichs (RFID) "MixingAxisControl" ein.
3. Geben Sie als "Anzeigename" des Wirkbereichs (RFID) "Mischachsen-Steuerung" ein.
4. Geben Sie für "ID" des Wirkbereichs (RFID) "1" ein.
   Achten Sie darauf, dass die Wirkbereich (RFID)-ID mit der Kennung an der Maschine übereinstimmt.
5. Doppelklicken Sie in der Tabelle "RFID Tags" auf "Hinzufügen...".
6. Geben Sie für "Name" des RFID Tags "RFID Tag 1" ein.
7. Geben Sie für "ID" des RFID Tags "1" ein. Die RFID Tag-ID stellen Sie dann am RFID Tag ein.
8. Aktivieren Sie die Zuordnung von "RFID Tag 1".
   "RFID Tag 1 " ist dem Wirkbereich (RFID) "MixingAxisControl" zugeordnet.

Ergebnis

Der Wirkbereich (RFID) "MixingAxisControl" ist projektiert.

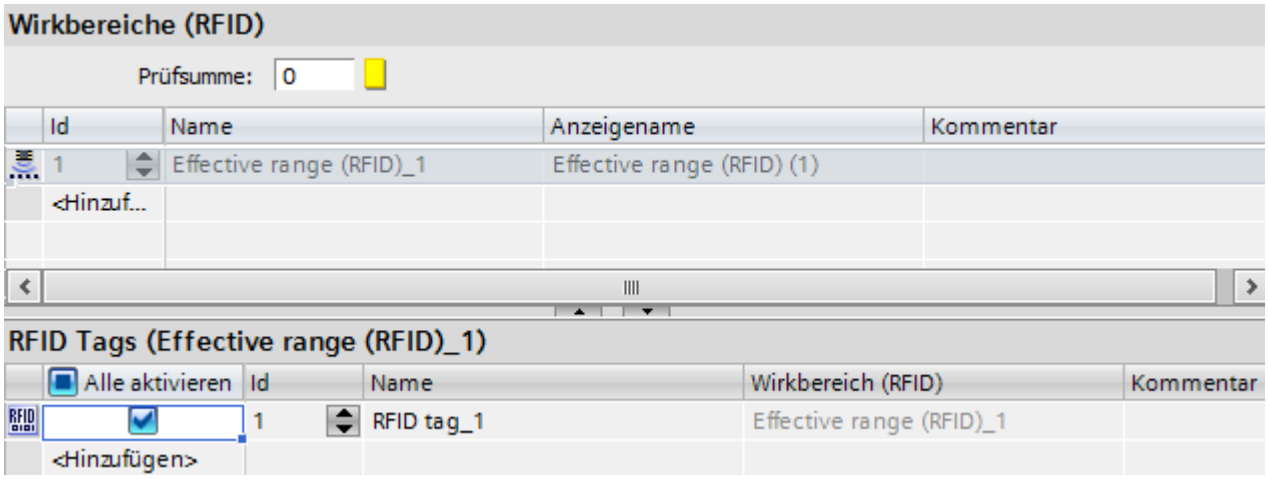

Der Anzeigename bezeichnet den Wirkbereich (RFID) in der eingestellten Bearbeitungssprache. Eine Übersicht über die Anzeigenamen finden Sie im Editor "Projekttexte".

10.7.3 Wirkbereichs-Name projektieren (RFID)

Einleitung

Damit sich ein Bediener über einen Wirkbereich (RFID) an einer Maschine anmelden und abmelden kann, projektieren Sie das Objekt "Wirkbereichs-Name". Wenn Sie bei einem Mobile Panel Wireless mit RFID das Objekt "Wirkbereichs-Name (RFID)" nicht projektieren, erscheint beim Generieren eine Warnung.

Voraussetzungen

Ein Bild ist angelegt und geöffnet.

Vorgehensweise

1. Wählen Sie im Werkzeugfenster unter "Elemente" das Objekt "Wirkbereichs-Name (RFID)".
2. Ziehen Sie per Drag&Drop das Objekt "Wirkbereichs-Name (RFID)" aus dem Werkzeugfenster in das Bild.

Ergebnis

Wenn der Bediener in Runtime auf das Objekt klickt, startet er das Anmelden oder Abmelden des Bediengeräts an einer Maschine, die einem Wirkbereich (RFID) zugeordnet ist.

10.8 Referenz

10.8.1 PROFIsafe-Adresse

Gültigkeit

Die PROFIsafe-Adresse ist für Mobile Panels Wireless und KTP F-Geräte verfügbar.

Hinweis

Nähere Hinweise zum fehlersicheren Betrieb finden Sie der Betriebsanleitung des Bediengeräts:

Gesamtdokumentation der Mobile Wireless (http://support.automation.siemens.com/WW/view/de/26268960)

PROFIsafe-Adresse

Die PROFIsafe-Adresse ist die netzweit und stationsweit eindeutige Adresse für die F-Peripherie in der PROFIsafe-Kommunikation. Bei den Mobile Panels Wireless hinterlegen Sie im Projekt einen Standardwert und laden ihn auf das Bediengerät. Auch am Bediengerät können Sie eine PROFIsafe-Adresse hinterlegen.

Die PROFIsafe-Adresse muss aus dem Wertebereich 1 bis 65534 sein. Beide PROFIsafe-Adressen im Engineering System und im Bediengerät werden auf formale Gültigkeit geprüft.

Beim Start der Runtime wird die PROFIsafe-Adresse folgendermaßen geladen:

- Im Control Panel ist eine gültige PROFIsafe-Adresse parametriert:
  Das Bediengerät lädt die im Control Panel eingestellte PROFIsafe-Adresse.

10.8.2 Power Management

Gültigkeit

Im Editor "Runtime-Einstellungen > Power Management" projektieren Sie den Energiesparmodus eines Mobile Panels.

Power Management

Das Mobile Panel Wireless kann auf folgende Arten Energie sparen:

- Helligkeit verringern, leichte Einsparung
- Bildschirm ausschalten, hohe Einsparung

Sie geben jeweils an, nach welcher Zeit ohne eine Bedienung das Power Management aktiviert wird. Die Kommunikationsverbindung bleibt dabei bestehen. Die Zeit für "Bildschirm ausschalten" ist größer als die Zeit für "Helligkeit verringern".

10.8.3 Zonen-ID/Anschlusspunkt-ID

Gültigkeit

Im Editor "Runtime-Einstellungen > Allgemein" projektieren Sie eine interne Variable, die abhängig vom Wert z. B. spezifische Informationen am Display des Bediengeräts einblendet.

Hinweis

Mobile Panels Wireless mit RFID

Für das Mobile Panels Wireless mit RFID ist der Bereich "Zonen-ID/Anschlusspunkt-ID" nicht verfügbar.

Zonen-ID/Anschlusspunkt-ID

An dieses Feld projektieren Sie eine interne Variable, die in Abhängigkeit des Bediengerätetyps folgenden Wert zur Laufzeit enthält:

- Die ID der Anschluss-Box bzw. des Wirkbereichs, an die das kabelgebundene Mobile Panel angeschlossen ist.
- Die ID der Zone, in der sich das Mobile Panel Wireless befindet. Bei einem Wert von 255 befindet sich das Bediengerät in keiner Zone.

Aufgaben planen 11

11.1 Grundlagen

11.1.1 Einsatzgebiet des Aufgabenplaners

Definition

Im Aufgabenplaner projektieren Sie Aufgaben, die unabhängig vom Bild im Hintergrund ausgeführt werden. Sie erstellen die Aufgaben, indem Sie Systemfunktionen oder Skripte an einen Trigger knüpfen. Wenn das auslösende Ereignis eintritt, folgt der Aufruf der verknüpften Funktionen.

Anwendungsbeispiel

Der Aufgabenplaner dient dazu, automatisiert ereignisgesteuerte Aufgaben auszuführen. Mit einer Aufgabe automatisieren Sie z. B.:

- Regelmäßige Auslagerung von Archivdaten
- Ausdruck eines Meldeprotokolls bei Meldepufferüberlauf
- Ausdruck eines Protokolls am Schichtende
- Überwachung einer Variablen
- Überwachung eines Benutzerwechsels

Hinweis

Die Verfügbarkeit der genannten Beispiele ist abhängig vom Bediengerät.

11.1.2 Arbeiten mit Aufgaben und Triggern

Einleitung

Eine Aufgabe besteht aus einem Trigger und einem Aufgabentyp.

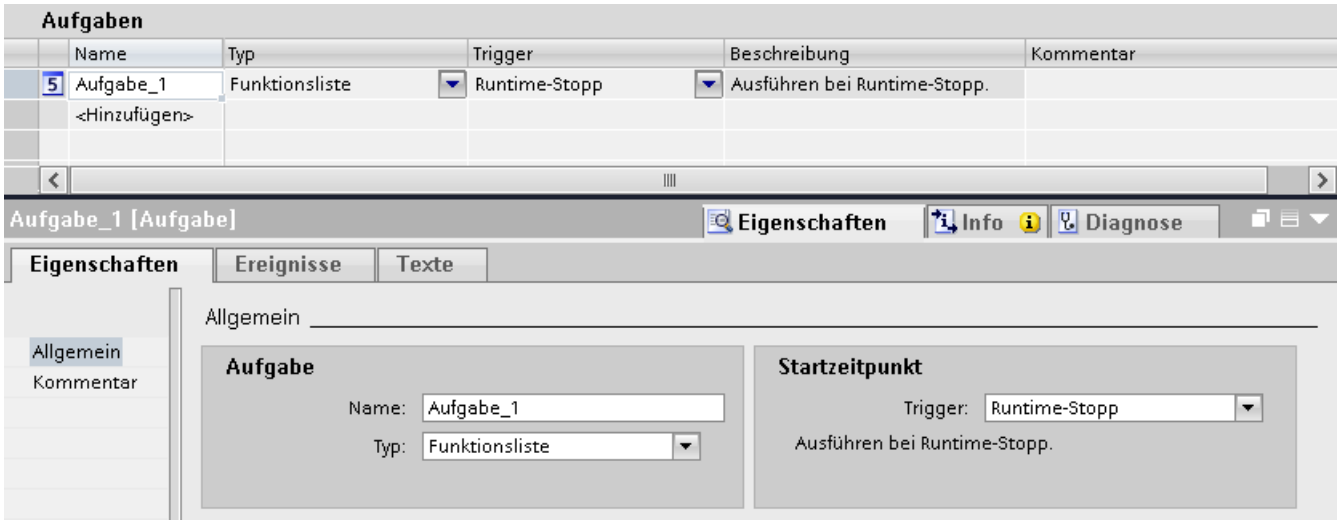

Starten einer Aufgabe

Gesteuert durch einen Trigger startet der Aufgabenplaner die Aufgabe, die zu dem Trigger gehört.

11.1.3 Arbeitsbereich des Editors Aufgabenplaner

Einleitung

Sie öffnen den Aufgabenplaner im Projektfenster durch einen Doppelklick auf "Aufgabenplaner". Der Arbeitsbereich zeigt die geplanten Aufgaben, die aus dem auslösenden Trigger und dem Aufgabentyp bestehen z. B. der Funktionsliste.

Aufbau

Der Arbeitsbereich besteht aus der Tabelle der Aufgaben.

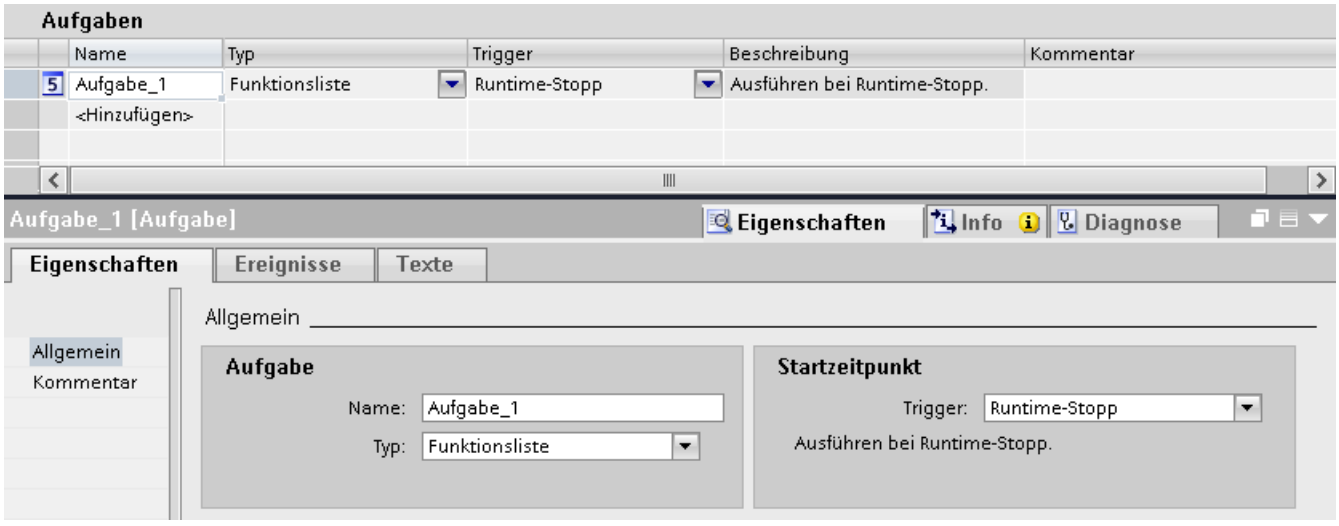

Die Tabelle der Aufgaben zeigt die festgelegten Aufgaben mit ihren Eigenschaften z. B. dem auslösenden Trigger. Sie wählen einen Aufgabentypen und einen Trigger aus. Sie vergeben einen Namen und einen Kommentar für die Aufgabe. Die Beschreibung fasst textuell die Aufgabe mit dem geplanten Zeitpunkt zusammen.

Inspektorfenster

Die Registerkarte "Eigenschaften" des Inspektorfensters teilt sich in zwei Bereiche.

Der Bereich "Aufgabe" zeigt ebenfalls den Namen der Aufgabe und den Aufgabentyp. Der Bereich "Startzeitpunkt" zeigt den auslösenden Trigger. Der Bereich unterscheidet sich je nach Auswahl des Triggers.

In der Registerkarte "Ereignisse" projektieren Sie die Funktionsliste mit Systemfunktionen, die in der Aufgabe ausgeführt werden.

Hinweis

Nähere Informationen zu Objekten in der Oberfläche erhalten Sie über Tooltips. Bewegen Sie dazu den Mauszeiger auf das gewünschte Objekt oder drücken Sie <F1>, wenn das Objekt selektiert ist.

11.1.4 Trigger

Einleitung

Ein Trigger wird mit einer Aufgabe verbunden und bildet somit das auslösende Ereignis für den Aufruf dieser Aufgabe. Wenn der Trigger eintritt, wird die Aufgabe ausgeführt.

Ereignis-Trigger

Wenn eine Aufgabe an ein Systemereignis geknüpft ist, dann wird die Aufgabe ereignisgesteuert ausgelöst. Systemereignisse sind z. B. Runtime-Stopp, Bildwechsel, Benutzerwechsel usw.

Jedes Systemereignis ist pro Bediengerät nur einmal projektiertbar.

Aufgabe deaktivieren

Wenn Sie eine Aufgabe vorübergehend nicht benötigen, deaktivieren Sie die Aufgabe im Engineering System. Mit dem Trigger "Deaktiviert" machen Sie auch ein bereits projektiertes Systemereignis wieder verfügbar.

Beispiel: Sie planen eine Aufgabe "A" mit dem Systemereignis "Runtime-Stopp". Dann ist dieses Systemereignis für eine andere Aufgabe "B" nicht mehr verfügbar. Um das Systemereignis "Runtime-Stopp" wieder verfügbar zu machen, wählen Sie bei der Aufgabe "A" als Trigger "Deaktiviert".

Hinweis

Welche Trigger Ihnen zur Verfügung stehen ist abhängig vom Bediengerät.

11.1.5 Trigger

Einleitung

Ein Trigger wird mit einer Aufgabe verbunden und bildet somit das auslösende Ereignis für den Aufruf dieser Aufgabe. Wenn der Trigger eintritt, wird die Aufgabe ausgeführt.

Triggerarten

Der Aufgabenplaner unterscheidet verschiedene Triggerarten:

Azyklische Trigger

Sie bestehen aus der Angabe des Datums und der Uhrzeit. Aufgaben mit azyklischem Trigger führt der Aufgabenplaner einmal am angegebenen Datum zur angegebenen Uhrzeit aus.

Zyklische Trigger

Aufgaben, die an einen zyklischen Trigger geknüpft sind, treten regelmäßig zu einem bestimmten Zeitpunkt ein.

- Standardzyklus: Beim Standardzyklus fällt der Beginn des ersten Zeitintervalls mit dem Start von Runtime zusammen. Die Länge des Intervalls wird durch den Zyklus bestimmt.
- Trigger mit definiertem Startzeitpunkt: Sie legen einen Startzeitpunkt und ein Zeitintervall fest. Mit zyklischen Triggern führen Sie Aufgaben z. B. täglich oder wöchentlich aus. Sie wählen z. B. "Wöchentlich" als Trigger, damit ist die Länge des Intervalls eine Woche. Den Beginn des Zeitintervalls legen Sie mit dem Wochentag und der Uhrzeit fest.

Hinweis

Anwenderdefinierte Zyklen

Sie haben eigene Zyklen definiert. Im Aufgabenplaner können Sie nur Zyklen verwenden, deren Werte bei maximal einer Stunde liegen.

Ereignis-Trigger

Wenn eine Aufgabe an ein Systemereignis geknüpft ist, dann wird die Aufgabe ereignisgesteuert ausgelöst. Systemereignisse treten im Gegensatz zu zyklischen Triggern meist azyklisch auf. Ein Systemereignis ist z. B. Runtime-Stopp.

Jedes Systemereignis ist pro Bediengerät nur einmal projektiertbar.

Aufgabe deaktivieren

Wenn Sie eine Aufgabe vorübergehend nicht benötigen, deaktivieren Sie die Aufgabe im Engineering System. Mit dem Trigger "Deaktiviert" machen Sie auch ein bereits projektiertes Systemereignis wieder verfügbar.

Beispiel: Sie planen eine Aufgabe "A" mit dem Systemereignis "Benutzerwechsel". Dann ist dieses Systemereignis für eine andere Aufgabe "B" nicht mehr verfügbar. Um das Systemereignis "Benutzerwechsel" wieder verfügbar zu machen, wählen Sie bei der Aufgabe "A" als Trigger "Deaktiviert".

Hinweis

Welche Trigger Ihnen zur Verfügung stehen ist abhängig vom Bediengerät.

11.1.6 Timer bei zyklischen Triggern

Timer bei zyklischen und azyklischen Triggern

Sie haben die Möglichkeit, den Startzeitpunkt einer Aufgabe in Runtime dynamisch zu ändern. Als Timer wählen Sie eine HMI-Variable. Der Wert der Variablen bestimmt in Runtime den Zeitpunkt, an dem die Aufgabe startet.

Hinweis

Eine HMI-Variable muss vom Typ "DateTime" sein. Eine PLC-Variable muss vom Typ "Date and Time" bzw. "DTL" sein.

Solange nach Runtime-Start der Bediener die Variable nicht ändert, gilt als Startzeitpunkt der in der Aufgabe projektierte Zeitpunkt. Sobald der Bediener die Variable erstmals per Eingabe ändert, ist immer der aktuelle Wert der Variablen der Startzeitpunkt. Um wieder den projektierten Zeitpunkt als Startzeitpunkt zu setzen, gibt der Bediener den projektierten Zeitpunkt ein.

11.1.7 Variablentrigger

Variablentrigger

Neben azyklischen, zyklischen und Ereignis-Triggern bietet WinCC auch den Variablentrigger an.

Variablentrigger bestehen aus der Angabe einer oder mehrerer Variablen. Nur wenn sich ein Variablenwert ändert, wird die Aufgabe ausgeführt, die mit solch einem Trigger verbunden ist. Damit wird eine Reduzierung der Systemlast erreicht. In welchem Zyklus die Variablenwerte abgefragt werden, ist für jede Variable individuell einstellbar. Für die Abfragezyklen wählen Sie zwischen "250 ms" und ein Mal pro Stunde. Zusätzlich stehen Ihnen selbst definierte Anwenderzyklen zur Verfügung.

Regeln für die Auswahl des Triggers

Verwenden Sie bei der Verarbeitung von Variablen den Variablentrigger anstatt des zyklischen Triggers. Damit reduziert sich die Systemlast.

- Variablentrigger
  Nur wenn sich der Wert der Variablen ändert, wird die Aufgabe ausgeführt.
  = Aufgabe wird ausgeführt.

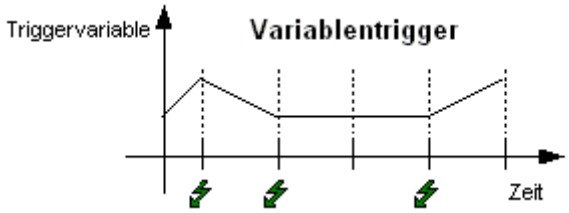

- Zyklischer Trigger
  Als zyklischen Trigger haben Sie z. B. "10 Sekunden" gewählt. Wenn der Trigger eintritt, wird eine Aufgabe ausgeführt.
  = Aufgabe wird ausgeführt.

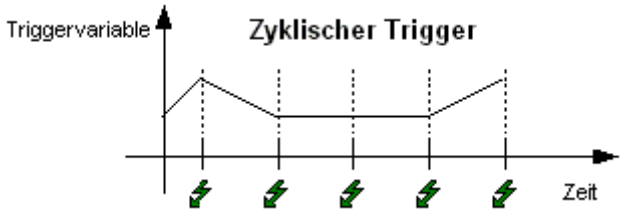

11.1.8 Funktionsliste

Funktionsliste

Ein Trigger startet die Funktionsliste. Die Funktionsliste wird Zeile für Zeile abgearbeitet. Jede Zeile enthält eine Systemfunktion. An jede Aufgabe projektieren Sie genau eine Funktionsliste.

Hinweis

Die Auswahl der projektierbaren Systemfunktionen in einer Funktionsliste ist abhängig vom gewählten Trigger und dem Bediengerät.

Siehe auch

Anbindung von lokalen VB-Skripten (Seite 1880)

Anbindung von lokalen C-Skripten (Seite 1881)

11.1.9 Funktionsliste

Funktionsliste

Ein Trigger startet die Funktionsliste. Die Funktionsliste wird Zeile für Zeile abgearbeitet. Jede Zeile enthält eine Systemfunktion oder ein lokales Skript. An jede Aufgabe projektieren Sie genau eine Funktionsliste.

Hinweis

Die Auswahl der projektierbaren Systemfunktionen in einer Funktionsliste ist ist abhängig vom gewählten Trigger und dem Bediengerät.

Abarbeitung von Systemfunktionen und Skripten

In der Regel werden Systemfunktionen und Skripte in einer Funktionsliste in Runtime sequenziell von oben nach unten abgearbeitet.

Um nicht sequenzielle und bedingte Abläufe zu programmieren, verwenden Sie ein Skript mit Schleifen, bedingten Anweisungen und Abbruchbedingungen.

11.1.10 Druckauftrag

Einleitung

WinCC Runtime Professional bietet neben der "Funktionsliste" auch den Aufgabentyp "Druckauftrag" an.

Druckauftrag

Um das Protokoll zu drucken, erstellen Sie einen Druckauftrag. Tritt ein zyklischer oder azyklischer Trigger ein, startet der Aufgabenplaner den Ausdruck eines Protokolls. Im Editor "Druckaufträge" definieren Sie vorab den Seitenumfang und Zeitbereich des Protokolls. Weiterhin bestimmen Sie, ob das Protokoll über den Drucker ausgegeben oder in eine Datei geschrieben wird.

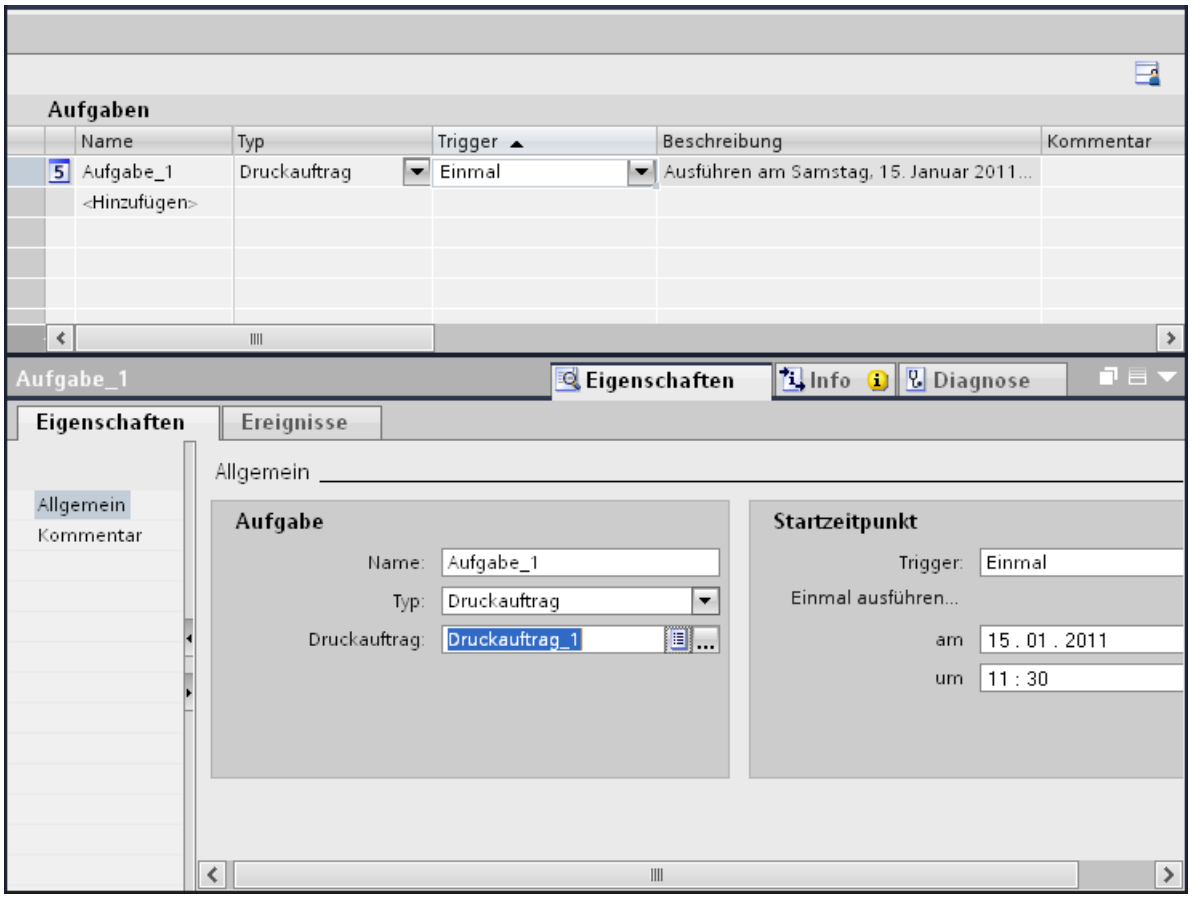

Wenn Sie als Aufgabentyp "Druckauftrag" gewählt haben, erscheint im Bereich "Aufgabe" ein zusätzliches Feld. Im Feld "Druckauftrag" wählen Sie den vorab definierten Druckauftrag für die Aufgabe aus.

Hinweis

Um den Aufgabentyp "Druckauftrag" zu verwenden, aktivieren Sie in der Startsequenz von WinCC Runtime "Geplante Druckaufträge in Runtime".

Klicken Sie im Editor "Runtime-Einstellungen>Dienste > Geplante Druckaufträge in Runtime ".

11.1.11 Anbindung von lokalen VB-Skripten

Einleitung

Runtime Professional bietet Ihnen neben den Aufgabentypen "Funktionsliste" und "Druckauftrag" auch eine Anbindung von lokalen VB-Skripten.

Anbindung lokaler VB-Skripte

Lokale Skripte werden von Ihnen erstellt und geändert. Auch die in der Funktionsliste angebotenen Systemfunktionen können Sie in ein lokales Skript konvertieren und den Code weiter bearbeiten. Das lokale VB-Skript wird mit der Aufgabe gespeichert.

Ein Trigger startet das Skript in Runtime. Wenn Sie eine Funktionsliste in ein VB-Skript konvertieren, haben Sie keinen Zugriff mehr auf die Funktionsliste für diese Aufgabe. Um wieder auf die Funktionsliste zuzugreifen, löschen Sie das VB-Skript.

Der Aufgabenplaner bietet Ihnen die Möglichkeit unabhängig von Bildern VB-Skripte zu starten. In den VB-Skripten haben Sie Zugriff auf Variablen und Bilder in Runtime.

Es besteht kein gemeinsamer Datenbereich zwischen den Skripte im Aufgabenplaner und denen in Bildern. Eine Synchronisation von Skripten im Aufgabenplaner und den Bildern erfolgt nur über das DataSet-Objekt oder Variablen. Der Datentyp von VBS-Variablen ist immer VARIANT.

Das lokale VB-Skript wird nicht parallel abgearbeitet. Das Skript, das zuletzt angestoßen wurde, kommt in eine Warteschleife, bis es abgearbeitet werden kann.

Hinweis

Beim Anlegen eines Skripts wird die Anweisung "Option explicit" automatisch und nicht löschbar in den Deklarationsbereich eingetragen. Die Anweisung ist erforderlich, um Fehler durch falsche Schreibweise von Variablen ohne Deklaration zu vermeiden.

Die Anweisung erfordert, dass Sie Variablen immer mit der Anweisung "Dim" in Ihrem Code definieren. Verwenden Sie die Anweisung "Option explicit" nicht in Ihrem Code, da es sonst zu Lauzeitfehlern kommen kann.

Voraussetzung

Um die Funktionsliste und lokale Skripte zu verwenden, müssen Sie in der Startsequenz von WinCC "Aufgaben in Runtime" aktivieren.

Klicken Sie im Editor "Runtime-Einstellungen > Dienste > Aufgaben in Runtime".

Hinweis

Sie können C-Skripte nicht in VB-Skripten aufrufen und umgekehrt.

Anwenderzyklen als Trigger

Wenn Sie Anwenderzyklen als Trigger für VB-Skripte verwenden, werden die Skripte in Runtime immer mit einer Granularität von 250 ms ausgeführt.

Siehe auch

Funktionsliste (Seite 1877)

11.1.12 Anbindung von lokalen C-Skripten

Einleitung

Runtime Professional bietet Ihnen neben den Aufgabentypen "Funktionsliste" und "Druckauftrag" auch eine Anbindung von lokalen C-Skripten.

Anbindung lokaler C- Skripte

Lokale Skripte werden von Ihnen erstellt und geändert. Auch die in der Funktionsliste angebotenen Systemfunktionen können Sie in ein lokales C-Skript konvertieren und den Code weiter bearbeiten. Das lokale C-Skript wird mit der Aufgabe gespeichert.

Ein Trigger startet das Skript in Runtime. Wenn Sie eine Funktionsliste in ein C-Skript konvertieren, haben Sie keinen Zugriff mehr auf die Funktionsliste für diese Aufgabe. Um wieder auf die Funktionsliste zuzugreifen, löschen Sie das C-Skript.

Ein Trigger startet das lokale C-Skript in Runtime.

Das lokale C-Skript wird nicht parallel abgearbeitet. Das Skript, das zuletzt angestoßen wurde, kommt in eine Warteschleife, bis es abgearbeitet werden kann.

Voraussetzung

Um die Funktionsliste und lokale Skripte zu verwenden, müssen Sie in der Startsequenz von WinCC "Aufgaben in Runtime" aktivieren.

Klicken Sie im Editor "Runtime-Einstellungen > Dienste > Aufgaben in Runtime".

Hinweis

Sie können C-Skripte nicht in VB-Skripten aufrufen und umgekehrt.

Anwenderzyklen als Trigger

Wenn Sie Anwenderzyklen als Trigger für C-Skripte verwenden, werden die Skripte in Runtime immer mit einer Granularität von 250 ms ausgeführt.

Siehe auch

Funktionsliste (Seite 1877)

11.2 Aufgaben planen

11.2.1 Aufgabe planen mit azyklischem Trigger

Einleitung

Sie planen ein einmaliges Beenden von Runtime für Wartungsarbeiten. Die Aufgabe startet eine Funktion, die Runtime beendet.

Voraussetzungen

- Der Editor "Aufgabenplaner" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen ...".
2. Geben Sie als "Name" "Wartung zum Ende des Jahres" ein.
3. Wählen Sie als "Trigger" "Einmal".
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > am" den "31.12.2008" aus.
5. Geben Sie im Feld "um" die Uhrzeit "18:00" ein.
6. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".

7. Wählen Sie in der Funktionsliste die Systemfunktion "StoppeRuntime".
8. Wählen Sie als "Modus" "Runtime".

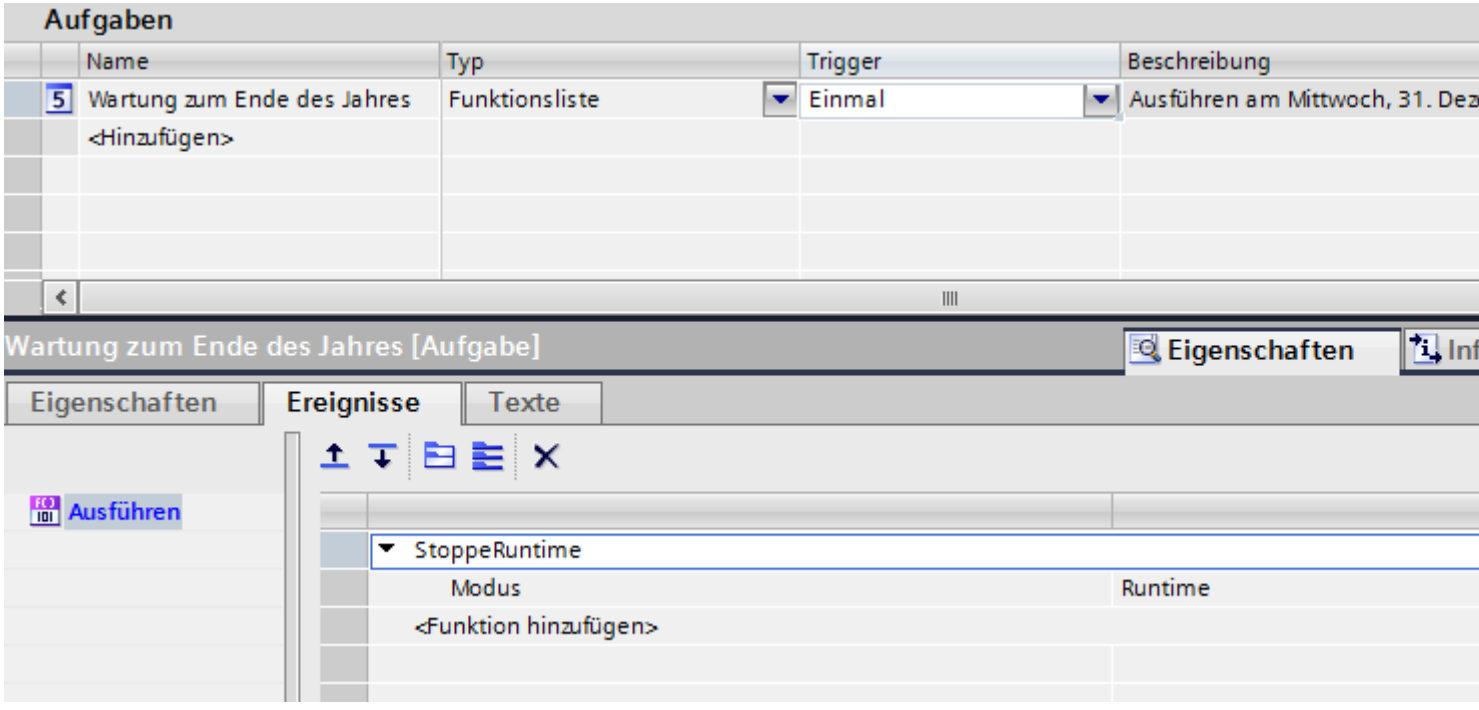

Ergebnis

Die Aufgabe wird einmalig ausgeführt. Am 31.12.2008 um 18:00 Uhr wird die Runtime gestoppt.

Hinweis

Ereignis "StoppeRuntime" auf RT Professional

Beachten Sie, dass beim Ereignis "StoppeRuntime" die Runtime im Begriff ist, sich zu beenden und daher nicht mehr alle Funktionalitäten zur Verfügung stehen.

11.2.2 Aufgabe planen mit zyklischem Trigger

Einleitung

Sie planen eine Aufgabe, die den täglichen Ausdruck eines Protokolls startet.

Voraussetzungen

- Ein Bediengerät mit Runtime Professional ist angelegt.
- Der Druckauftrag "Tägliches Meldeprotokoll" wurde erstellt.
- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Wählen Sie "Typ" "Druckauftrag".
3. Geben Sie als "Name" "Täglicher Ausdruck des Meldeprotokolls" ein.
4. Wählen Sie als "Trigger" "Täglich".
5. Geben Sie im Inspektorfenster " Eigenschaften > Eigenschaften > Allgemein > um" die Uhrzeit "18:00" ein.

6. Wählen Sie im Bereich "Aufgabe" den Druckauftrag "Tägliches Meldeprotokoll" aus.

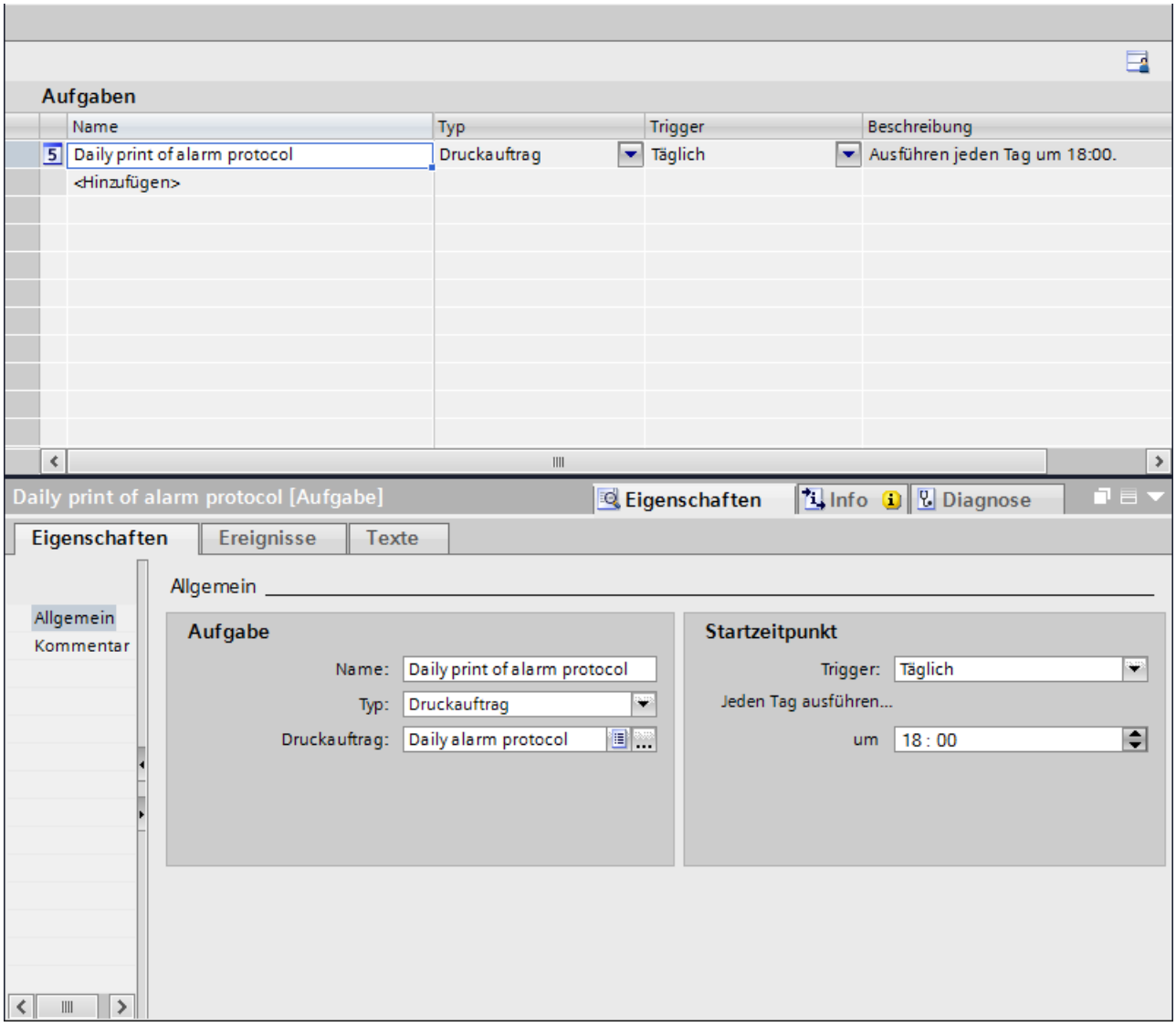

7. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".
8. Aktivieren Sie "Startsequenz von WinCC Runtime > Geplante Druckaufträge in Runtime".

Ergebnis

Um 18 Uhr wird täglich das Protokoll "Tägliches Meldeprotokoll" gedruckt.

11.2.3 Aufgabe planen mit Ereignis-Trigger

Einleitung

Sie planen eine Aufgabe, die beim Benutzerwechsel einen Bildwechsel auslöst.

Voraussetzungen

- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.
- Das Bild "Start" ist angelegt.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Geben Sie als "Name" "Bildwechsel bei Benutzerwechsel" ein.
3. Wählen Sie als "Trigger" "Benutzerwechsel".
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
5. Wählen Sie in der Funktionsliste die Systemfunktion "Bilder/AktiviereBild".
6. Wählen Sie im Feld Bildname das Bild "Start".

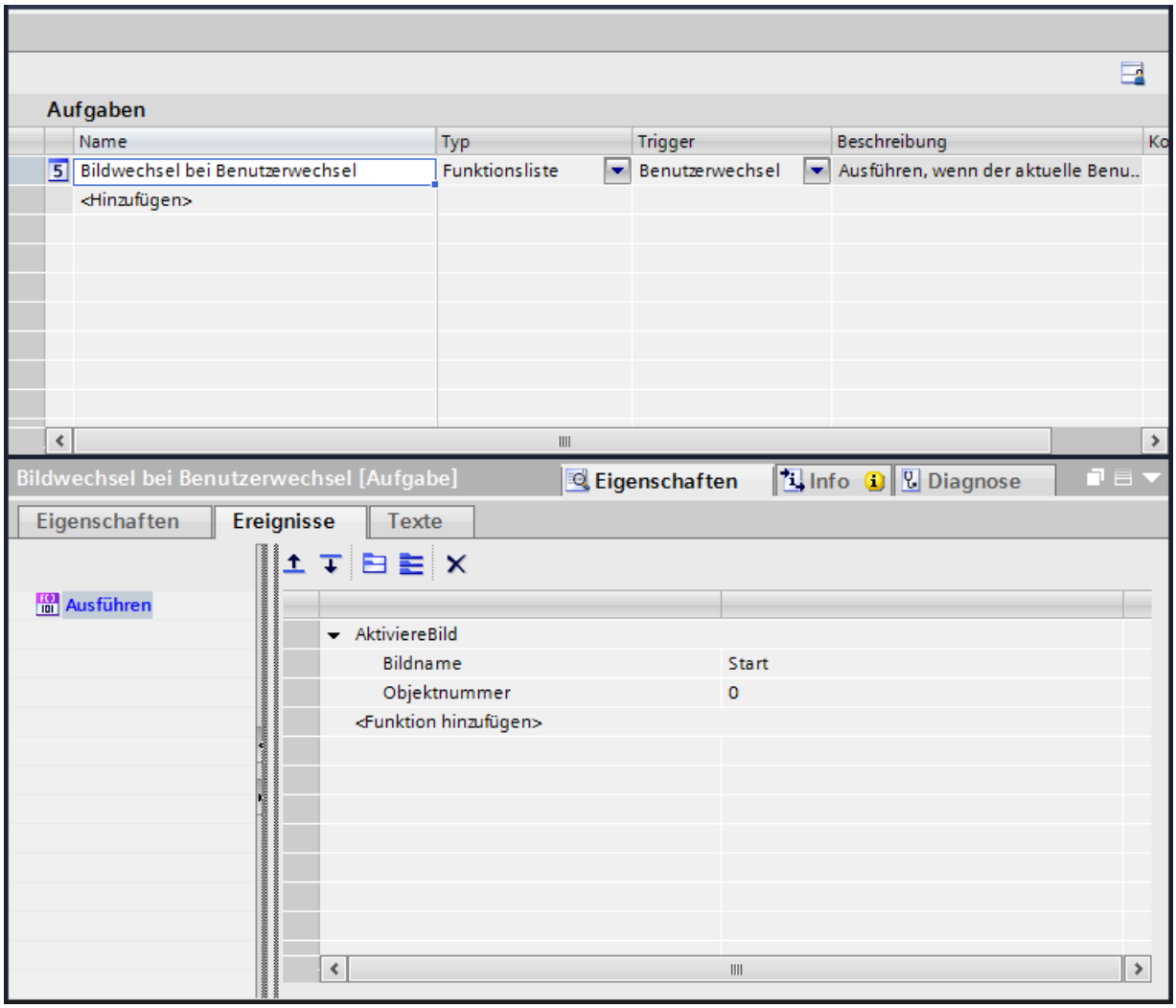

Ergebnis

Die Aufgabe wird beim Ereignis "Benutzerwechsel" ausgeführt. Wenn sich ein neuer Benutzer anmeldet, wird das Bild "Start" eingeblendet.

11.2.4 Aufgabe verwalten

Bezeichnung ändern

1. Doppelklicken Sie im Arbeitsbereich auf das Feld in der Spalte "Name".
2. Ändern Sie den Namen der Aufgabe.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ ändern Sie die Bezeichnung im Inspektorfenster "Aufgabe > Name".

Trigger ändern

1. Markieren Sie im Arbeitsbereich das Feld in der Spalte "Trigger".
2. Öffnen Sie die Auswahlliste über die Schaltfläche und wählen Sie den gewünschten Trigger aus.

Alternativ ändern Sie den Trigger im Inspektorfenster im Bereich "Startzeitpunkt".

Aufgabe löschen

1. Markieren Sie eine oder mehrere Zeilen der zu löschenden Aufgaben.
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Menübefehl "Löschen".

Alternativ löschen Sie eine oder mehrere Aufgaben mithilfe der Schaltfläche <ENTF>.

11.3 Beispiele für Runtime Professional

11.3.1 Beispiel: Täglich Startbild aktivieren

Einleitung

Sie planen eine Aufgabe, die täglich vor Schichtbeginn das Startbild mit dem Anmeldedialog aktiviert.

Voraussetzungen

- Ein Bild "Startbild" ist erstellt.
- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle "Aufgaben" auf "Hinzufügen..." .
2. Geben Sie als "Name" "Startbild aktivieren" ein.
3. Wählen Sie als "Typ" "Funktionsliste"
4. Wählen Sie als "Trigger" "Täglich".
5. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein >um" die Uhrzeit "05:30" ein.
6. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".
7. Wählen Sie in der Funktionsliste in der Gruppe "Bilder" die Systemfunktion "AktiviereBild".

8. Wählen Sie als "Bildname" "Startbild" aus.

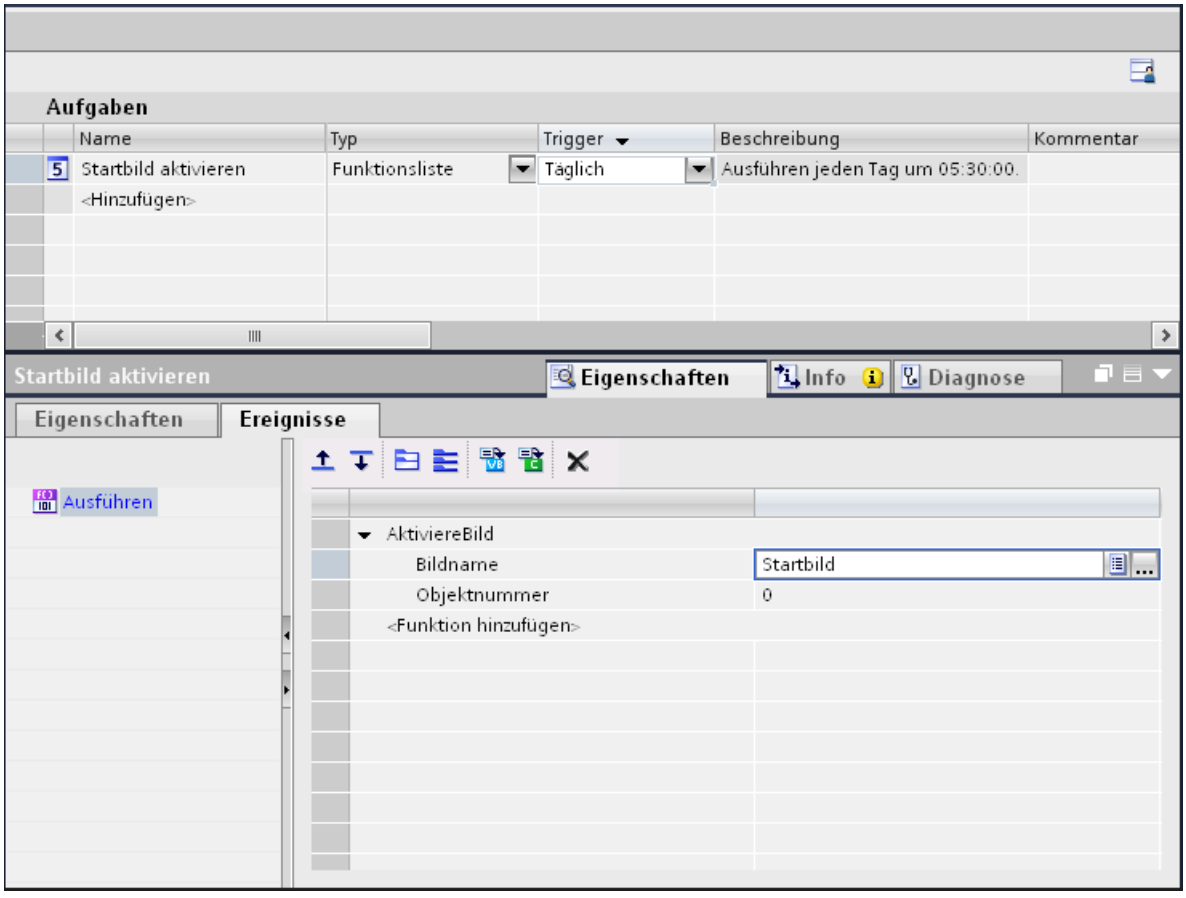

9. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".
10. Aktivieren Sie "Aufgaben in Runtime".

Ergebnis

Täglich um 5:30 wird das Bild "Startbild" aktiviert.

11.3.2 Beispiel: Einmaliger Ausdruck eines Protokolls

Aufgabe

Sie planen eine Aufgabe, die einmalig ein Protokoll am Jahresende ausgibt.

Voraussetzung

- Der Druckauftrag "Jahresübersicht" ist angelegt.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen..." .
2. Geben Sie als "Name" "Jahresabschluss" ein.
3. Wählen Sie als "Typ" "Druckauftrag"
4. Wählen Sie als "Trigger" "Einmal".
5. Wählen Sie im Inspektorfenster "Eigenschaften >Eigenschaften > Allgemein >am" das Datum "31.12.2010" aus.
6. Geben Sie im Feld "um" die Uhrzeit "22:00" ein.
7. Wählen Sie im Bereich "Aufgabe" den Druckauftrag "Jahresübersicht" aus.

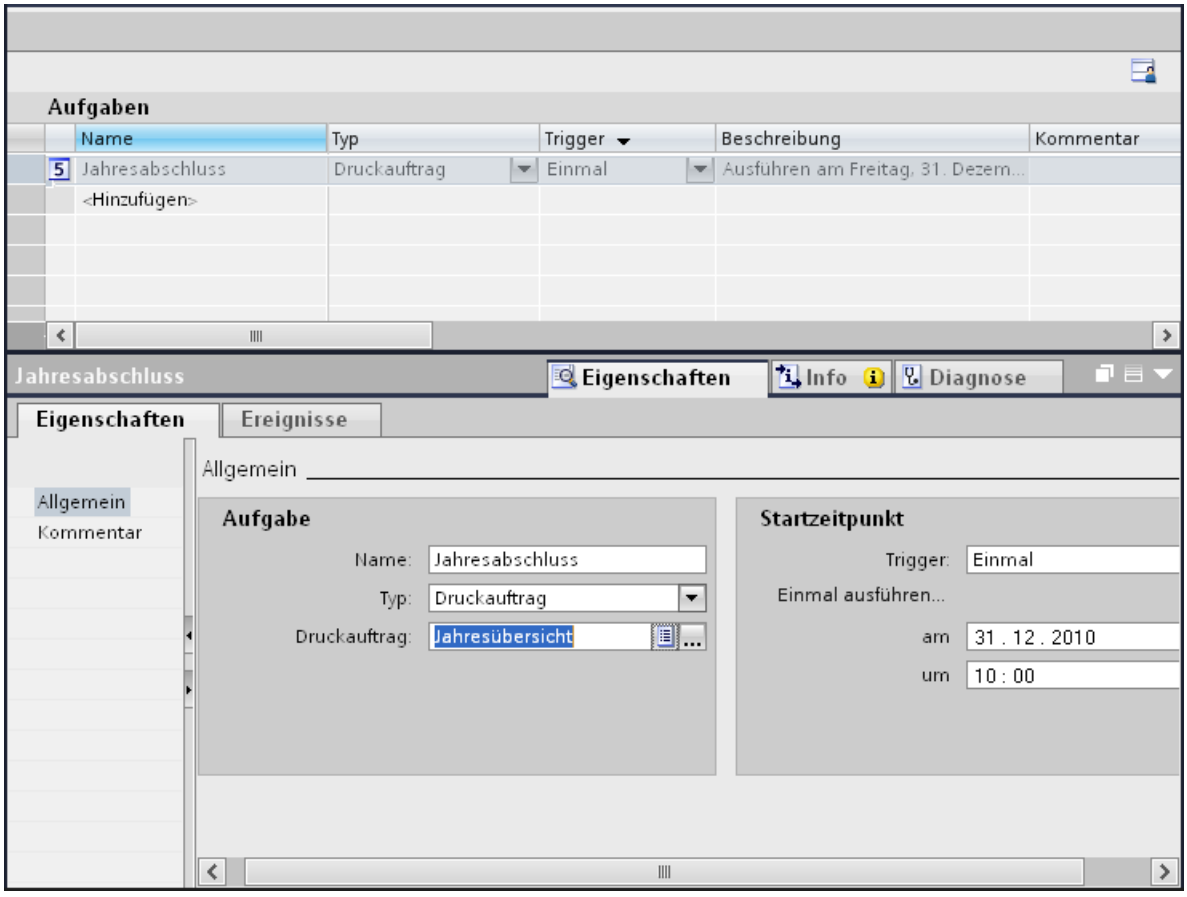

8. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".
9. Aktivieren Sie "Startsequenz von WinCC Runtime > Geplante Druckaufträge in Runtime".

Ergebnis

Am 31.12.2008 um 22:00 Uhr wird der Druckauftrag "Jahresübersicht" gestartet.

11.3.3 Beispiel: Lesen des aktuellen Benutzernamens

Einleitung

Sie planen eine Aufgabe, die den aktuellen Benutzer in ein Textfeld schreibt.

Voraussetzungen

- Eine Bild "Screen_1" ist erstellt.
- Ein Textfeld "Textfield_1" ist im Bild "Screen_1" angelegt.
- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Geben Sie als "Name" "Lesen des Benutzernamens" ein.
3. Wählen Sie als "Trigger" "Variablentrigger".
4. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Trigger hinzufügen".
5. Wählen Sie die Systemvariable "@CurrentUser".
6. Wählen Sie als "Zyklus" "1 min".
7. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".
8. Klicken Sie im Bereich "Funktionsliste" auf .
9. Geben Sie folgendes VB-Skript ein:

```
Sub Action ()
    Dim user
    user = HMIRuntime.Tags("@CurrentUser").Read
    HMIRuntime.Screens("Screen_1").ScreenItems("Textfield_1").Text = user
End Sub
```

1. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".
2. Aktivieren Sie "Aufgaben in Runtime".

Ergebnis

Bei Änderung des Variablenwerts wird der aktuelle Benutzer in das Textfeld "Textfield_1" geschrieben.

11.3.4 Beispiel: Eine Variable hochzählen

Einleitung

Sie planen eine Aufgabe, die einmal pro Stunde den Wert einer Variablen um eins hochzählt.

Voraussetzungen

- Ein Bediengerät mit Runtime Professional ist angelegt.
- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.
- Die Variable "Tag1" ist angelegt.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Wählen Sie "Typ" "Funktionsliste".
3. Geben Sie als "Name" "Variable hochzählen" ein.
4. Wählen Sie als "Trigger" "1 Stunde".
5. Geben Sie Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Startzeitpunkt > zu Minute" den Wert "35:00" ein.
6. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".
7. Klicken Sie im Bereich "Funktionsliste" auf .
8. Geben Sie folgen Programmcode in das Eingabefeld ein:

```
Sub Action ()
    Dim objTag1
    Dim lngValue
    Set objTag1 = HMIRuntime.Tags("Tag1")
    lngValue = objTag1.Read
    objTag1.Write lngValue + 1
End Sub
```

1. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".
2. Aktivieren Sie "Aufgaben in Runtime".

Ergebnis

Die Variable wird zyklisch hochgezählt.

11.4 Beispiele für Runtime Advanced und Panels

11.4.1 Beispiel: Benutzer bei Benutzerwechsel aktualisieren

Aufgabe

Sie projektieren ein E/A-Feld, welches den angemeldeten Benutzer anzeigt. Sie planen eine Aufgabe, die beim Wechsel des angemeldeten Benutzers das E/A-Feld aktualisiert.

Voraussetzungen

- Eine Variable "CurrentUser" vom Typ "String" ist angelegt.
- Ein Bild ist angelegt und geöffnet.
- Ein E/A-Feld ist im Bild angelegt.

Vorgehensweise

1. Klicken Sie auf das Objekt "E/A-Feld".
2. Klicken Sie im Inspektorfenster "Eigenschaften >Eigenschaften > Allgemein":
   - Wählen Sie als "Anzeigeformat" "Zeichenkette".
   - Wählen Sie als "Variable" "CurrentUser".
   - Wählen Sie als Modus "Ausgabe".
3. Wechseln Sie in den Arbeitsbereich des Aufgabenplaners.
4. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
5. Geben Sie als "Name" "CurrentUser" ein.
6. Wählen Sie als "Trigger" "Benutzerwechsel".
7. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".

8. Wählen Sie in der Funktionsliste in der Gruppe "Benutzerverwaltung" die Systemfunktion "LeseBenutzername".
9. Wählen Sie als "Variable" "CurrentUser".

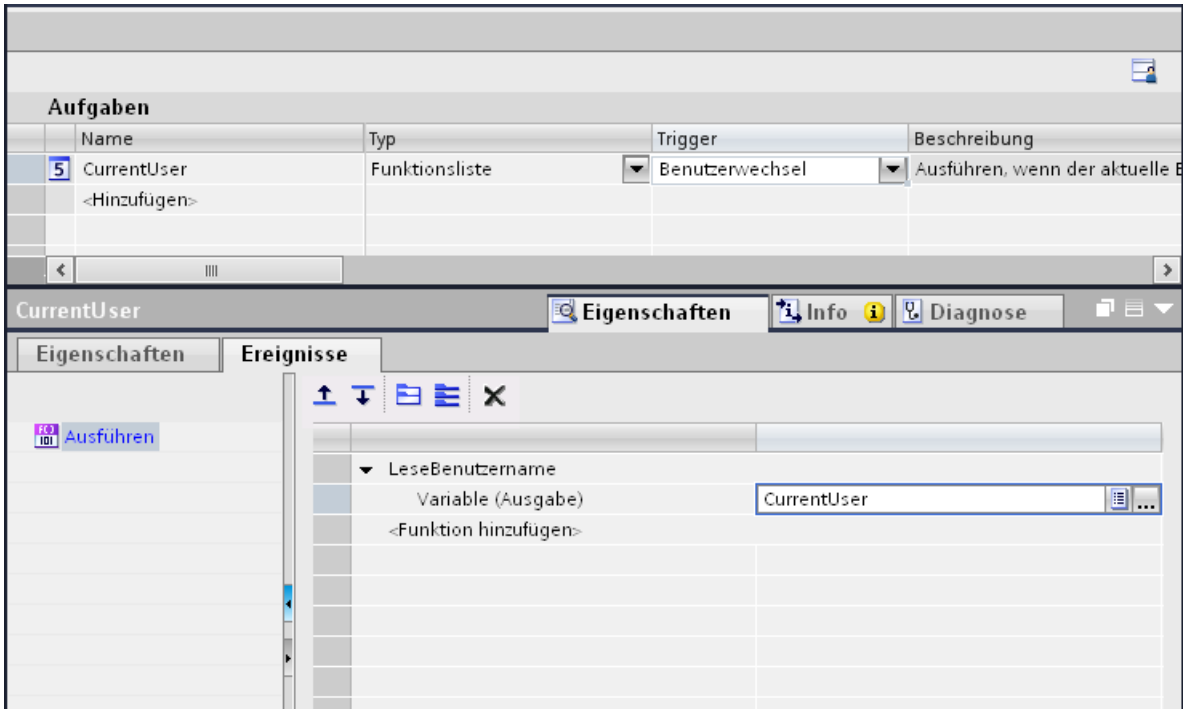

Ergebnis

Wenn sich ein neuer Benutzer erfolgreich anmeldet, wird die Systemfunktion "LeseBenutzername" aufgerufen. Die Variable "CurrentUser" wird aktualisiert und im E/A-Feld der neue angemeldete Benutzer angezeigt.

Wenn sich ein Benutzer nicht erfolgreich anmeldet, wird der angemeldete Benutzer abgemeldet. Das E/A-Feld zeigt weiterhin den ehemals angemeldeten Benutzer, bis sich ein neuer Benutzer erfolgreich anmeldet.

11.4.2 Beispiel: Wöchentlicher Ausdruck eines Protokolls

Einleitung

Sie planen eine Aufgabe, die den wöchentlichen Ausdruck eines Protokolls startet.

Voraussetzungen

- Ein Protokoll "Wöchentliches Protokoll" wurde erstellt.
- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Geben Sie als "Name" "Wöchentlicher Protokollausdruck zum Schichtende" ein.
3. Wählen Sie als "Trigger" "Wöchentlich".
4. Wählen Sie im Inspektorfenster "Eigenschaften >Eigenschaften > Allgemein > am" "Freitag".
5. Geben Sie im Feld "um" die Uhrzeit "18:00" ein.

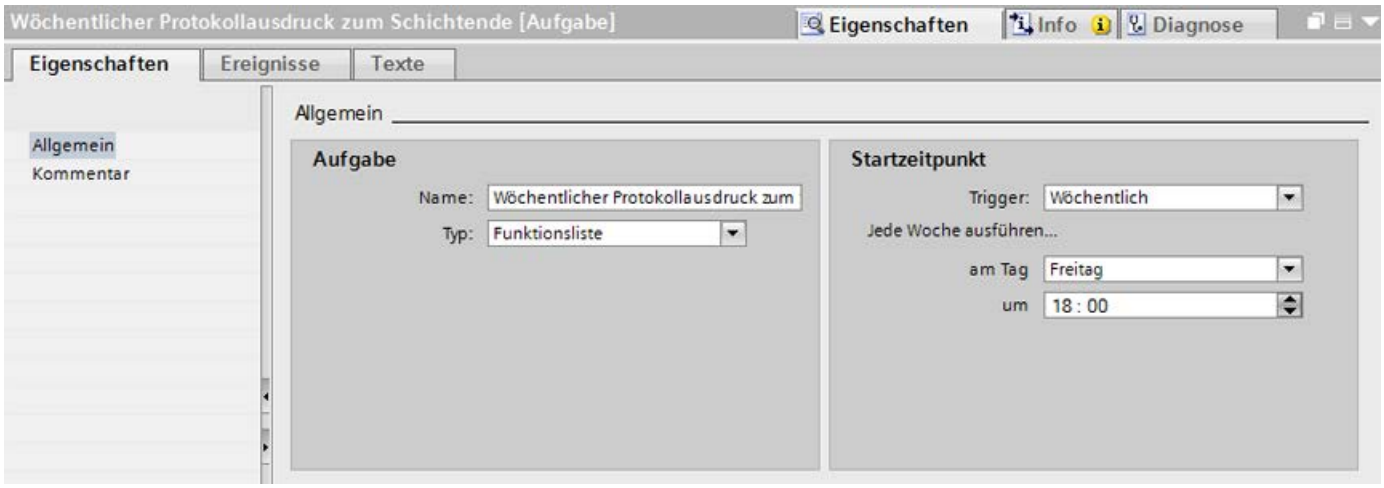

6. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
7. Wählen Sie in der Funktionsliste die Systemfunktion "Drucken/DruckeProtokoll".
8. Wählen Sie als Protokoll "Wöchentliches Protokoll" aus.

Ergebnis

Der wöchentliche Druckauftrag startet am Freitag um 18:00h.

11.4.3 Beispiel: Startzeitpunkt einer Aufgabe in Runtime ändern

Einleitung

Ein zyklischer Trigger "Täglicher Start um 18:00 Uhr" wird im Engineering System projektiert. Die Uhrzeit "18:00" ist als Konstante fest. Um jedoch den Startzeitpunkt in Runtime zu ändern, wählen Sie als "Timer" eine Variable. Der Wert der Variable bestimmt in Runtime den Zeitpunkt, an dem die Aufgabe startet.

Aufgabe

Sie planen eine Aufgabe, die täglich Archivdaten auslagert und als Timer die Variable "LogTime" besitzt. Sie projektieren eine Variable "LogTime" an ein Eingabefeld.

Voraussetzungen

- Das Quellarchiv "DataLogSource" ist als Variablenarchiv erstellt.
- Das Zielarchiv "DataLogDestination" ist als Variablenarchiv erstellt.
- Ein Bild "ChangeLogTime" ist angelegt.

Handlungsübersicht

1. Timervariable anlegen: Sie legen eine Variable an, mit der Sie in Runtime den Startzeitpunkt der Archivsauslagerung ändern.
2. Datum/Uhrzeit-Feld projektieren: Der Bediener ändert mithilfe des Datum/Uhrzeit-Felds in Runtime den Startzeitpunkt der Aufgabe.
3. Aufgabe mit Timervariable projektieren: Sie legen eine Aufgabe an, deren Startzeitpunkt in Runtime dynamisch änderbar ist.

Ergebnis

Wenn im Eingabefeld "LogDataTimeField" nichts eingegeben wird, startet standardmäßig die Aufgabe täglich um 18:00 Uhr. Die Archivdaten werden auslagert. Wenn ein Bediener im Eingabefeld "LogDataTimeField" 19:00 Uhr eingibt, werden die Archivdaten um 19:00 ausgelagert. Vorausgesetzt der Wert der Variable ändert sich nicht bis zum Start der Aufgabe.

11.4.4 Beispiel: 1. Variable für Runtime projektieren

Aufgabe

Im folgenden Beispiel projektieren Sie die Variable "LogTime".

Vorgehensweise

1. Öffnen Sie den Editor "HMI-Variablen".
2. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
3. Geben Sie als "Name" der Variablen "LogTime" ein.
4. Wählen Sie in der Spalte "Verbindung" "Interne Variable".
5. Wählen Sie in der Spalte "Datentyp" "DateTime ".

Ergebnis

Eine Variable vom Typ "DateTime" wurde erstellt.

11.4.5 Beispiel: 2. Datum/Uhrzeit-Feld projektieren

Aufgabe

Im folgenden Beispiel projektieren Sie ein Datum/Uhrzeit-Feld. Um in Runtime den Startzeitpunkt der Aufgabe zu ändern, knüpfen Sie die Variable "LogTime" an das Datum/Uhrzeit-Feld.

Voraussetzungen

- Die Variable "LogTime" vom Typ "DateTime" ist angelegt.
- Das Bild "ChangeLogTime" ist geöffnet.

Vorgehensweise

1. Ziehen Sie im Werkzeugfenster aus der Kategorie "Einfache Objekte" ein "Datum/Uhrzeit-Feld" per Drag&Drop in das Bild.
2. Klicken Sie im Insperktorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
3. Deaktivieren Sie "Systemzeit".
4. Klicken Sie auf "Variable" und wählen Sie "LogTime".
5. Deaktivieren Sie "Datum anzeigen" und aktivieren Sie "Uhrzeit anzeigen".
6. Wählen Sie als "Modus" "Eingabe/Ausgabe".
7. Klicken Sie "Eigenschaften > Verschiedenes".
8. Geben Sie als "Name" "LogDataTimeField" ein.

Ergebnis

Der Bediener gibt mit dem angelegten Datum/Uhrzeit-Feld eine Uhrzeit ein.

11.4.6 Beispiel: 3. Archivierung mit Timervariable projektieren

Aufgabe

Im folgenden Beispiel projektieren Sie eine Aufgabe mit einer Timervariablen, mit der Sie in Runtime den Startzeitpunkt der Aufgabe ändern.

Vorausetzungen

- Die Variable "LogTime" ist angelegt
- Das Quellarchiv "DataLogSource" ist als Variablenarchiv erstellt.

- Das Zielarchiv "DataLogDestination" ist als Variablenarchiv erstellt.
- Der Arbeitsbereich des Aufgabenplaners ist geöffnet.

Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Geben Sie als "Name" "Tägliche Auslagerung eines Archivs zu variabler Zeit" ein.
3. Wählen Sie als "Trigger" "Täglich".
4. Geben Sie im Inspektorfenster "Eigenschaften >Eigenschaften > Allgemein > um" die Uhrzeit "18:00" ein.
5. Wählen Sie als "Standard-Timer" die Variable "LogTime".
6. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".

7. Wählen Sie in der Funktionsliste die Systemfunktion "Systemfunktionen/Archive/KopiereArchiv".
8. Wählen Sie folgende Einstellungen:
   - Wählen Sie als "Archivtyp" den Typ "Variablenarchiv".
   - Wählen Sie als "Zielarchiv" "DataLogDestination".
   - Wählen Sie als "Quellarchiv" "DataLogSource".
   - Wählen Sie als "Modus" "Überschreiben".
   - Wählen Sie als "Quellarchiv löschen" den Wert "Ja".

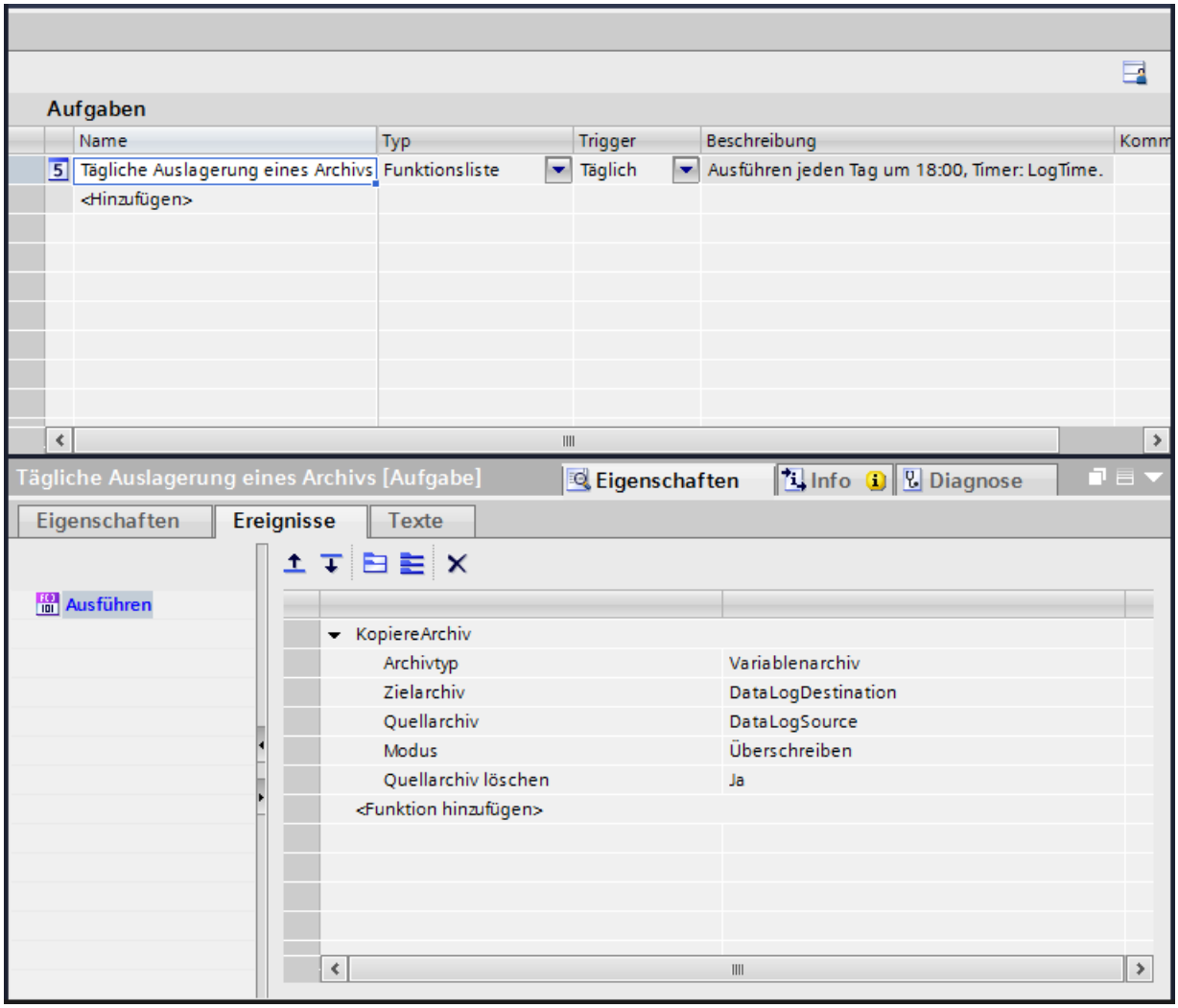

Ergebnis

Wenn im Eingabefeld "LogDataTimeField" nichts eingegeben wird, startet standardmäßig die Aufgabe täglich um 18:00 Uhr. Die Archivdaten werden auslagert. Wenn ein Bediener im Eingabefeld "LogDataTimeField" 19:00 Uhr eingibt, werden die Archivdaten um 19:00 ausgelagert. Vorausgesetzt der Wert der Variable ändert sich nicht bis zum Start der Aufgabe.

Diagnosefunktionen nutzen 12

12.1 Systemdiagnose projektieren

12.1.1 Systemdiagnose

12.1.1.1 Grundlagen für Basic Panels

Grundlagen der Systemdiagnose

Einleitung

Mit der Systemdiagnose können Sie Meldungen aus dem Diagnosepuffer aller integrierten Verbindungen anzeigen lassen.

System-Diagnoseanzeige

Die System-Diagnoseanzeige ist ein Bedien- und Anzeigeobjekt, das Sie in einem Bild verwenden.

Sie navigieren direkt zur Fehlerursache und zur zugehörigen Verbindung. Sie haben Zugriff zu allen integrierten Verbindungen, die Sie im Editor "Geräte & Netze" konfiguriert haben.

Ansichten in der Systemdiagnose

Einleitung

In der einfachen System-Diagnoseanzeige stehen drei verschiedene Ansichten zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht

Geräteansicht

Die Geräteansicht wird nur angezeigt, wenn mehr als eine integrierte Verbindung projektiert wurde.

Die Geräteansicht zeigt in tabellarischer Sicht alle verfügbaren Verbindungen. Durch einen Doppelklick auf eine Verbindung öffnen sich die Diagnosepuffer-Ansicht.

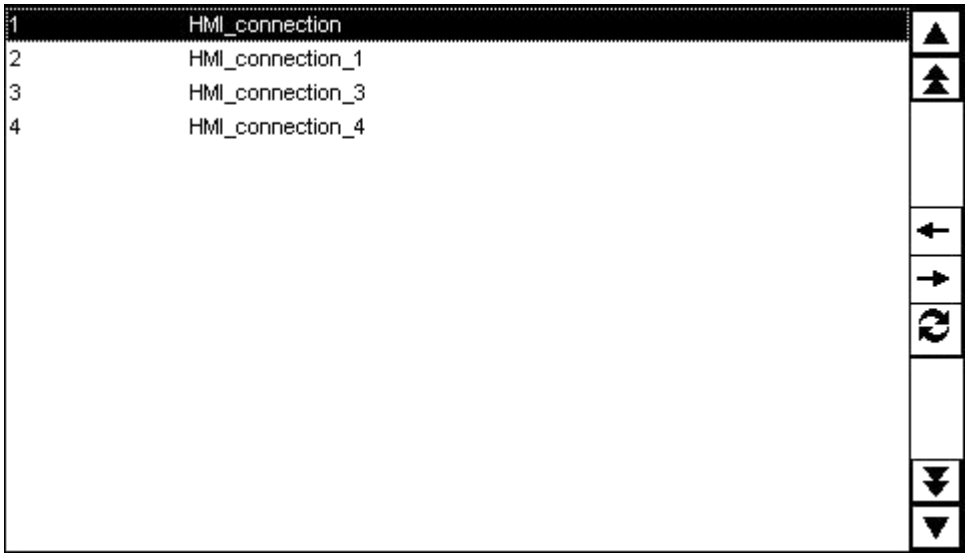

Diagnosepuffer-Ansicht

Die Diagnosepuffer-Ansicht zeigt die aktuellen Daten aus dem Diagnosepuffer.

Um die Diagnosepuffer-Ansicht zu aktualisieren, wählen Sie die Schaltfläche .

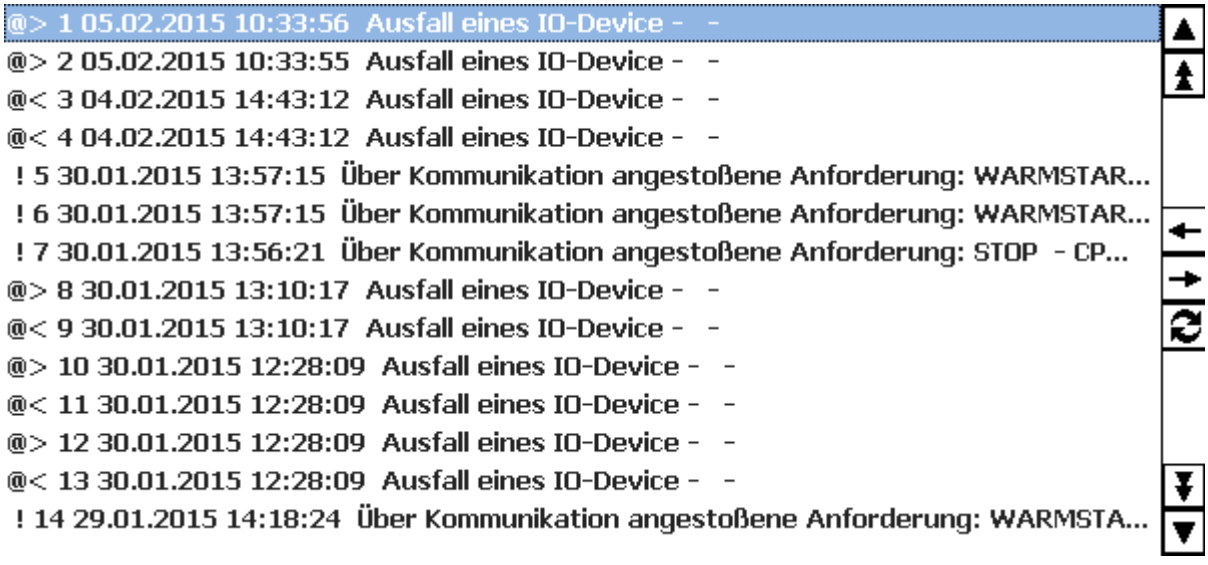

Detailansicht

Die Detailansicht zeigt ausführliche Informationen über die selektierte Verbindung und die anliegenden Fehler. In der Detailansicht prüfen Sie, ob die Daten korrekt sind.

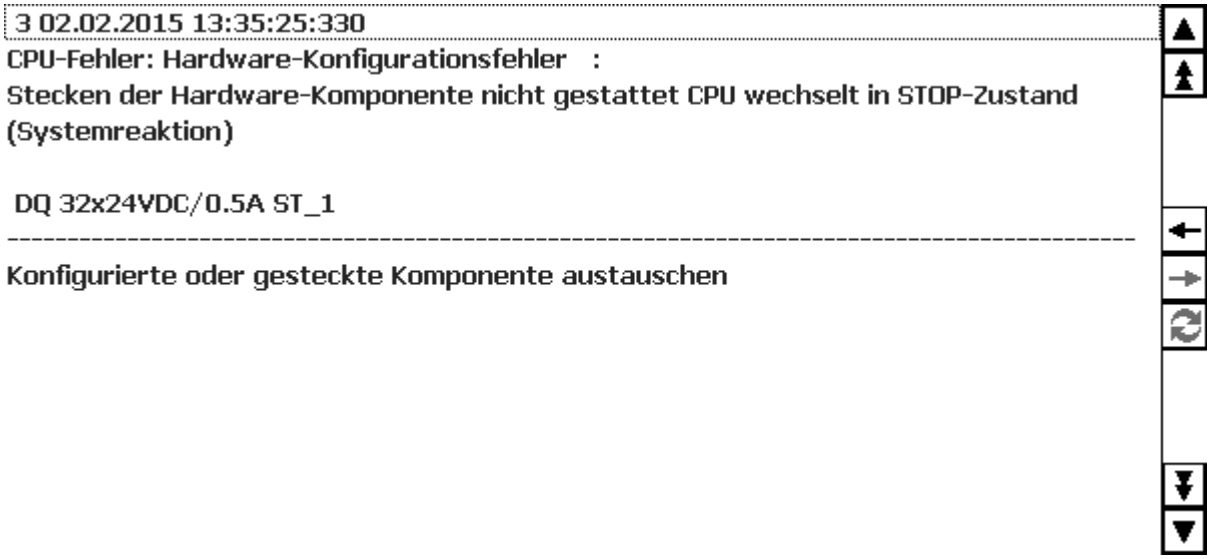

Hinweis

Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbinungen mit den Steuerungen S7-1200 und S7-1500 zur Verfügung.

Navigationsschaltflächen

Schaltfläche	Taste	Funktion
→	Enter-Taste	In der Geräteansicht: Öffnet die Diagnosepuffer-Ansicht des gewählten Geräts. In der Diagnosepuffer-Ansicht: Öffnet die Detailansicht.
←	Esc-Taste	In der Diagnosepuffer-Ansicht : Öffnet die Geräteansicht. In der Detailansicht: Öffnet die Diagnosepuffer-Ansicht.
	Projektierte Funktionstaste z. B. F1.	Aktualisiert die Diagnosepuffer-Ansicht..

12.1.1.2 Grundlagen für Panels und Runtime Advanced

Grundlagen der Systemdiagnose

Einleitung

Mit der Systemdiagnose können Sie Störungen und Fehler in Ihrer gesamten Anlage erkennen. WinCC bietet Ihnen zwei Anzeige- und Bedienobjekte für schnelle Fehlerlokalisierung.

Damit die Steuerung Fehlermeldungen an die System-Diagnoseanzeige oder das System-Diagnosefenster übergeben kann, müssen Sie für einige Steuerungen in den Einstellungen die Systemdiagnose aktivieren.

Bei der Kommunikation mit den Steuerungen SIMATIC S7-1500 und SIMATIC S7-1200 ist die Systemdiagnose automatisch aktiviert.

Auf dem Bediengerät ist unter "Runtime-Einstellungen > Meldungen > Steuerungsmeldungen" für die jeweilige HMI-Verbindung die Option "Systemdiagnose" automatisch aktiviert.

System-Diagnoseanzeige

Während Sie mit der Meldeanzeige den Zustand einer Steuerung betrachteten, bietet die System-Diagnoseanzeige einen Überblick über alle verfügbaren Geräte in Ihrer Anlage: Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

System-Diagnosefenster

Das System-Diagnosefenster ist ein Bedien- und Anzeigeobjekt, das Sie nur im Globalen Bild verwenden können.

Die Funktionen des System-Diagnosefensters unterscheiden sich nicht von denen der System-Diagnoseanzeige. Da das System-Diagnosefenster im Globalen Bild projektiert wird, können Sie z. B. zusätzlich festlegen, ob das Objekt in Runtime schließbar ist.

Hinweis

Systemdiagnose auf Basic Panels

Das Objekt "System-Diagnosefenster" steht auf Basic Panels nicht zur Verfügung.

Ansichten in der Systemdiagnose

Einleitung

In der System-Diagnoseanzeige und im System-Diagnosefenster stehen vier verschiedene Ansichten zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht
- Matrixansicht (nur für Mastersysteme, PROFIBUS, PROFINET)

Geräteansicht

Die Geräteansicht zeigt in tabellarischer Sicht alle verfügbaren Geräte einer Ebene. Durch einen Doppelklick auf ein Gerät öffnen sich entweder die untergeordneten Geräte oder die Detailansicht. Symbole in der ersten Spalte geben Auskunft über den aktuellen Zustand des Geräts.

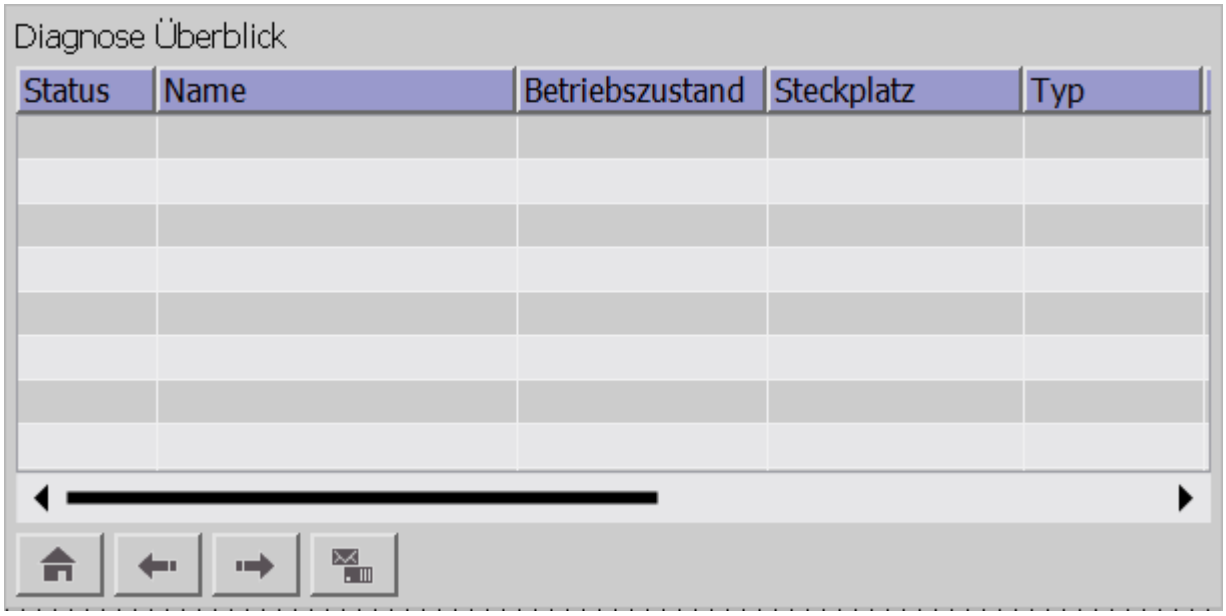

Diagnosepuffer-Ansicht

In der Diagnosepuffer-Ansicht werden die aktuellen Daten aus dem Diagnosepuffer des Geräts angezeigt. Die Diagnosepuffer-Ansicht ist nur in der Geräteansicht aufrufbar.

Um die Diagnosepuffer-Ansicht zu aktualisieren, wählen Sie die Schaltfläche .

319_194 \ Rail_0 \ 319_194 \ Diagnosepuffer-Ansicht

Nr.	Datum	Zeit	Ereignis
1	05.02.2015	11:33:47	Baugruppe ok
2	05.02.2015	11:33:45	Baugruppe gestört oder Wartung erforderlich
3	04.02.2015	15:43:03	Baugruppe gestört oder Wartung erforderlich
4	04.02.2015	15:43:03	Baugruppe gestört oder Wartung erforderlich
5	04.02.2015	15:43:02	Baugruppe gestört oder Wartung erforderlich
6	02.02.2015	13:41:59	Betriebszustandsübergang von ANLAUF nach RUN
7	02.02.2015	13:41:59	Manuelle Neustart (Warmstart)-Anforderung
8	02.02.2015	13:41:59	Betriebszustandsübergang von STOP nach ANLAUF
9	02.02.2015	13:41:52	Dezentrale Peripherie: Station Wiederkehr
10	02.02.2015	13:41:52	Dezentrale Peripherie: Station Wiederkehr

Detailansicht

Die Detailansicht zeigt ausführliche Informationen über das selektierte Gerät und die anliegenden Fehler. In der Detailansicht prüfen Sie, ob die Daten korrekt sind. Fehlertexte können Sie in der Detailansicht nicht sortieren.

1516_196 \ 1516_196

Status	
Name	1516_196
Betriebszustand	
Baugruppenträger	0
Steckplatz	1
Typ	CPU 1516-3 PN/DP
Bestellnummer	6ES7 516-3AN00-0AB0
Adresse	49*
Anlagenbezeichnung	PI 1516_196
Positionskennung	LI 1516_196
Hersteller ID	SIEMENS AG
Hardware Version	81
Profil ID	
spezifische Profildaten	
I&M Datenversion	1.1
Fehlertext	CPU nicht in RUN

Hinweis

Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbinungen mit den Steuerungen S7-1200 und S7-1500 zur Verfügung.

Dezentrale I/O Ansicht

Die Dezentrale I/O Ansicht steht bei Mastersystemen und IO-Systemen zur Verfügung. In dieser Ansicht sehen Sie den Status der Subgeräte des Systems. Jedes Element in der Ansicht zeigt den Gerätenamen, den Gerätetyp und die IP- oder die PROFIBUS-Adresse.

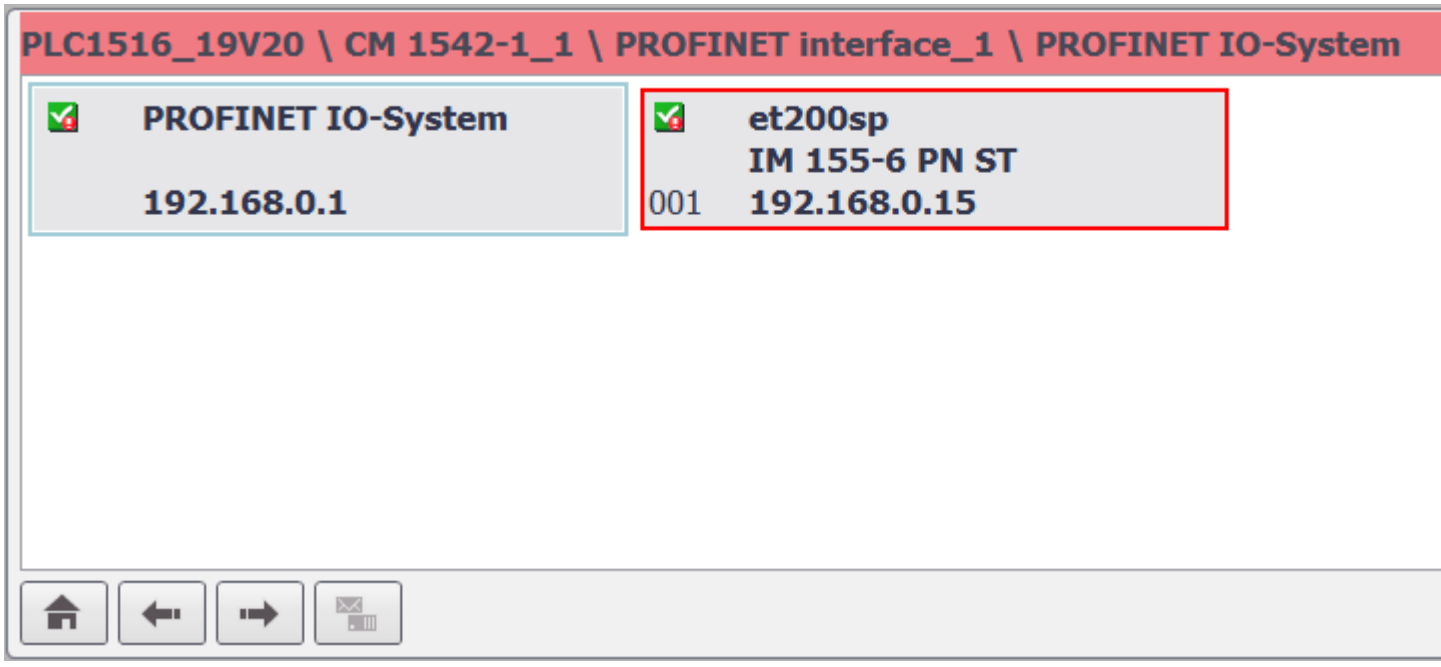

Im Fall einer Störung wird das betroffene Element visuell hervorgehoben:

- Rote Umrandung deutet auf eine Störung in einem untergeordneten Element
- Roter Hintergrund deutet auf eine Störung im Element selbst, z. B. das Gerät ist nicht erreichbar.

Navigationsschaltflächen

Schaltfläche	Taste	Funktion
	Enter-Taste	Öffnet die untergeordneten Geräte bzw. die Detailansicht, wenn keine Geräte untergeordnet sind.
	Esc-Taste	Öffnet das übergeordnete Gerät oder die Geräteansicht, falls kein übergeordnetes Gerät besteht.

Schaltfläche	Taste	Funktion
	Home-Taste.	Öffnet die Geräteansicht.
	Projektierte Funktionstaste z. B. F1.	Öffnet die Diagnosepuffer-Ansicht. Nur sichtbar in der Geräteansicht.
	Projektierte Funktionstaste z. B. F2.	Aktualisiert die Diagnosepuffer-Ansicht.

Hinweis

Navigation in der Matrixansicht bei Comfort Panels mit Funktionstasten

Für die Auswahl von PNIO- oder Profibus- Slaves in der Matrixansicht verwenden Sie zusätzlich die Umschalt-Taste.

Sprachumschaltung in Runtime

Schalten Sie die Sprache in Runtime um, bevor Sie die Detailansicht öffnen.

Falls Sie bei geöffneter Detailansicht die Runtimesprache umschalten, wechseln Sie zurück in die Diagnosepuffer-Ansicht.

Öffnen Sie erneut die Detailansicht. Die gewählte Runtimesprache wird nun korrekt dargestellt.

12.1.1.3 Objekte der Systemdiagnose projektieren

Systemdiagnose in der Steuerung aktivieren

Einleitung

Bei der Kommunikation mit den Steuerungen SIMATIC S7-1500 und SIMATIC S7-1200 ist die Systemdiagnose automatisch aktiviert.

Für andere Steuerungen sind die folgenden Einstellungen in der Steuerung erforderlich, damit die Steuerung Fehlermeldungen an die System-Diagnoseanzeige übergeben kann.

Voraussetzungen

- Eine Steuerung z. B. SIMATIC S7-300/400 oder ET 200 ist angelegt.
- Der Editor "Geräte & Netze" ist in der "Gerätesicht" geöffnet.

Vorgehen

1. Selektieren Sie in der Auswahlliste die Steuerung.
2. Klicken Sie auf die Steuerung und wählen Sie im Kontextmenü "Eigenschaften". Im Inspektorfenster werden die Eigenschaften der Steuerung angezeigt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Allgemein > Systemdiagnose > Allgemein".
4. Aktivieren Sie "Systemdiagnose für diese CPU aktivieren".
5. Selektieren Sie die Steuerung und wählen Sie im Kontextmenü "Übersetzen > Hardwarekonfiguration".

Hinweis

Besonderheit Systemdiagnosemeldungen SIMATIC S7-300/400

Wenn Sie unter "Runtime-Einstellungen > Meldungen > Steuerungsmeldungen > Anzeigeklassen" für die jeweilige HMI-Verbindung die Anzeigeklasse 0 deaktivieren, werden die Systemdiagnosemeldungen von der Steuerung nicht an das Bediengerät übertragen, auch wenn die Option "Systemdiagnose" in der Steuerung und auf dem Bediengerät aktiviert ist.

Ergebnis

Nach dem Übersetzen sind die Einstellungen wirksam. Die Steuerung kann Fehlermeldungen an die System-Diagnoseanzeige übergeben.

System-Diagnoseindikator einfügen

Einleitung

Der System-Diagnoseindikator ist ein vordefiniertes grafisches Symbol der Bibliothek, das Sie auf Fehler in Ihrer Anlage hinweist.

Wenn ein Fehler auftritt ändert sich das Aussehen des Objekts. Das Bibliotheksobjekt kann zwei Zuständeanzeigen:

- Kein Fehler

- Fehler

Voraussetzung

- Die Task Card "Bibliotheken" ist geöffnet.
- Die globale Bibliothek "Buttons and Switches > Kopiervorlagen > DiagnosticsButtons" ist geöffnet.
- Ein Bild ist geöffnet.
- Ein System-Diagnosefenster ist im Globalen Bild angelegt.

Vorgehen

1. Selektieren Sie in der Bibliothek das Objekt "DiagnosticsIndicator".

   Hinweis

   System-Diagnoseindikator für unterschiedliche Gerätetypen

   Achten Sie bei der Auswahl des Objekts darauf, dass Sie das richtige Objekt für den verwendeten Gerätetyp im Projekt verwenden .

2. Ziehen Sie das Bibliotheksobjekt per Drag&Drop an die Stelle des Arbeitsbereichs, an der Sie das Objekt einfügen wollen.
   Das Bibliotheksobjekt wird eingefügt.

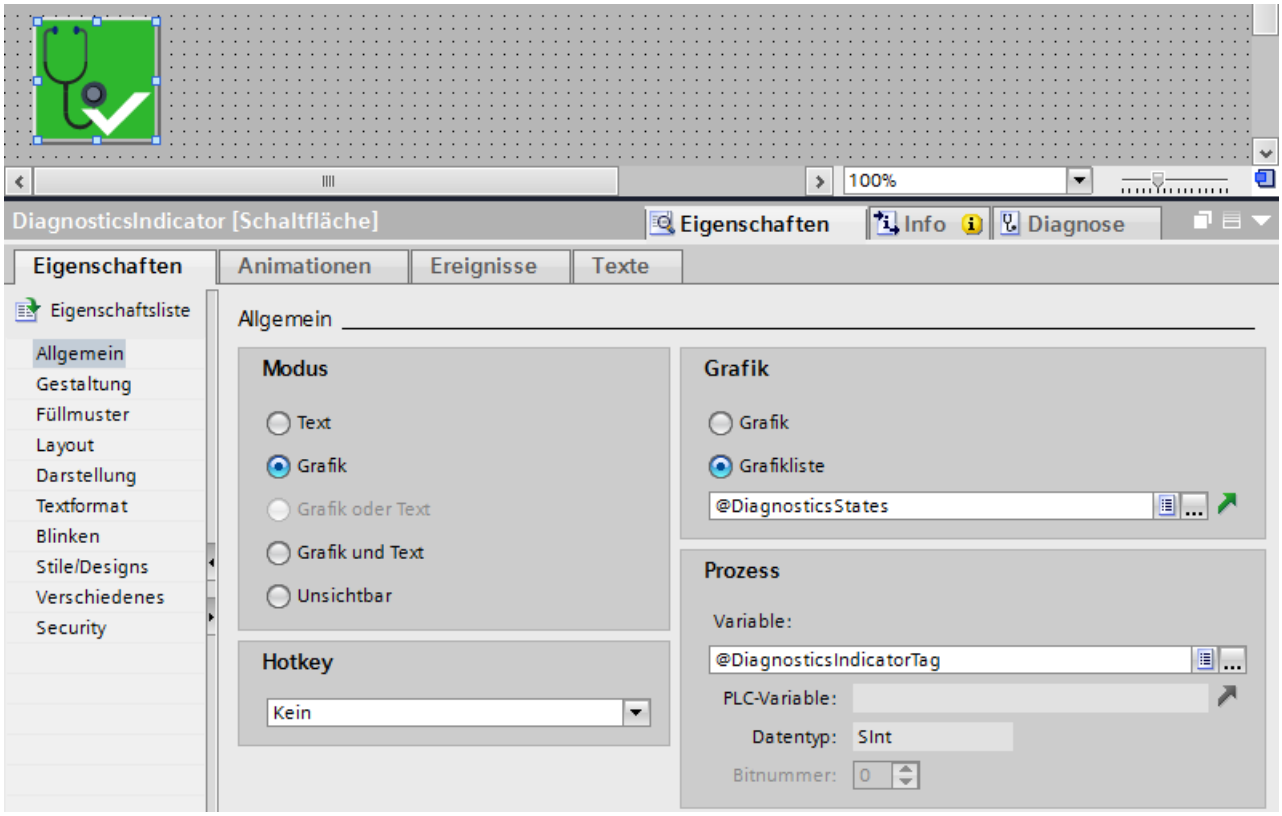

3. Selektieren Sie das Bibliotheksobjekt.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
   Für das Ereignis "Klicken" ist die Systemfunktion "ZeigeSystemdiagnosefenster" voreingestellt.

Ergebnis

Der Status-Diagnoseindikator ist im Projekt eingefügt und mit dem System-Diagnosefenster verbunden.

Wenn in Runtime eine Fehlermeldung auftritt, ändert der System-Diagnoseindikator sein Aussehen. Wenn Sie auf den System-Diagnoseindikator klicken, öffnet sich das System-Diagnosefenster. Das System-Diagnosefenster zeigt die Detailansicht des betroffenen Geräts.

Zugriffsschutz für System-Diagnosefenster projektieren

Um die System-Diagnosefenster vor unberechtigtem Zugriff zu schützen, projektieren Sie an den System-Diagnoseindikator einen Zugriffsschutz.

1. Selektieren Sie in das Objekt "DiagnosticsIndicator" im Bild.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Security in Runtime" eine Berechtigung.

Wenn Sie in Runtime auf den System-Diagnoseindikator klicken, öffnet sich ein Anmeldedialog. Nur wenn Sie die erforderliche Berechtigung haben, öffnet sich das System-Diagnosefenster.

System-Diagnosefenster im Globalen Bild projektieren

Einleitung

Das System-Diagnosefenster bietet Ihnen einen Überblick aller verfügbaren Geräte in Ihrer Anlage. Das System-Diagnosefenster verhält sich wie die System-Diagnoseanzeige, ist allerdings nur im Globalen Bild verfügbar.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Ein Comfort Panel oder WinCC RT Advanced ist angelegt.
- Das Globale Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnosefenster". Das Objekt wird im Globalen Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten > Geräte/ Detailansicht".
3. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. : Zustand, Name, Adresse.
4. Aktivieren Sie die Eigenschaften, die Sie in der Detailansicht für Runtime benötigen, z. B. : Zustand, Name, Anlagenkennzeichen.
5. Aktivieren Sie die Spalten, die Sie in der Diagnosepuffer-Ansicht benötigen, z. B. Zustand, Name.
6. Passen Sie bei Bedarf die Überschriften für die Spalten an.

7. Um die Spalten in Runtime zu verschieben, aktivieren Sie "Eigenschaften > Eigenschaften > Layout > Spalteneinstellungen > Spalten verschiebbar".
8. Damit das System-Diagnosefenster in Runtime geschlossen werden kann, aktivieren Sie "Eigenschaften > Eigenschaften > Fenster > Schließbar".

Ergebnis

Das System-Diagnosefenster ist im Globalen Bild eingefügt. Wenn eine Fehlermeldung in der Anlage auftritt, reagiert das System-Diagnosefenster und zeigt das betroffene Gerät an.

Projektieren Sie in einem Bild einen System-Diagnoseindikator, um das System-Diagnosefenster zu öffnen.

Schaltfläche als System-Diagnoseindikator projektieren

Einleitung

Alternativ zum Objekt "DiagnosticsIndicator" aus der Bibliothek, können Sie z. B. eine Schaltfläche projektieren, die Sie auf Fehler in Ihrer Anlage hinweist.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Die Task Card "Werkzeuge" ist geöffnet.
- Eine Bit-Grafikliste ist angelegt mit zwei verschiedenen Grafiken für die Zustände.
- Ein Bild ist geöffnet.
- Eine System-Diagnoseanzeige ist angelegt.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "Schaltfläche". Eine Schaltfläche wird im Bild eingefügt.
2. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Modus > Grafik".
3. Wählen Sie unter "Grafik > Grafikliste" die Bit-Grafikliste aus.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Variable" die Variable "@DiagnosticsIndicatorTag" aus.

Systemfunktion an die Schaltfläche projektieren

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
2. Wählen Sie das Ereignis "Klicken".
3. Klicken Sie in der Tabelle auf "Funktion hinzufügen".

4. Wählen Sie die Systemfunktion "SystemdiagnoseanzeigeAktivieren".
5. Wählen Sie die System-Diagnoseanzeige.

Ergebnis

Sie haben eine Schaltfläche projektiert, die auf Fehlermeldungen aus der Steuerung reagiert. Wenn in Runtime eine Fehlermeldung auftritt, ändert sich die Schaltfläche.

Die Schaltfläche hat zwei Zustände.

- Fehler
  Wenn Sie auf die Schaltfläche klicken, öffnet sich die System-Diagnoseanzeige. Die System-Diagnoseanzeige zeigt die Detailansicht des betroffenen Geräts.
- Kein Fehler
  Wenn Sie auf die Schaltfläche klicken, öffnet sich die System-Diagnoseanzeige. Die System-Diagnoseanzeige zeigt die Geräteansicht.

System-Diagnoseanzeige projektieren

Einleitung

Sie fügen eine System-Diagnoseanzeige in Ihrem Projekt ein, um einen Überblick über alle verfügbaren Geräte in Ihrer Anlage haben.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Ein Comfort Panel oder WinCC RT Advanced ist angelegt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten > Geräte/ Detailansicht".
3. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. Zustand, Name, Steckplatz.
4. Aktivieren Sie die Spalten, die Sie in der Detailansicht für Runtime benötigen, z. B. Zustand, Name, Anlagenkennzeichen.
5. Aktivieren Sie die Spalten, die Sie in der Diagnosepuffer-Ansicht benötigen z. B. Zustand, Name.

6. Um die Spalten in Runtime zu verschieben, aktivieren Sie "Eigenschaften > Eigenschaften > Layout > Spalteneinstellungen > Spalten verschiebbar".
7. Unter "Eigenschaften > Eigenschaften > Spaltenköpfe" können Sie bei Bedarf die Überschriften für die Spalten ändern.

Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt. In Runtime werden Fehler der gesamten Anlage in der System-Diagnoseanzeige angezeigt.

System-Diagnoseanzeige projektieren

Einleitung

Um einen Überblick über alle integrierten Verbindungen zu haben, fügen Sie eine System-Diagnoseanzeige in Ihrem Projekt ein.

Voraussetzungen

- Eine Steuerung ist angelegt.
- Ein Basic Panel ist angelegt.
- Eine integrierte Verbindung wurde im Editor "Geräte & Netze" erstellt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Darstellung".
3. Wählen Sie für "Zeilen pro Eintrag" z. B. 5.

Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt.

Um die aktuellen Meldungen zu erhalten, aktualisieren Sie den Diagnosepuffer.

Anzeige in der System-Diagnoseanzeige teilen

Einleitung

Die System-Diagnoseanzeige bietet auch eine geteilte Ansicht der Anzeige an. So haben Sie die Möglichkeit, die Geräte und die zugehörigen Details auf einen Blick zu sehen.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Ein Bediengerät ist angelegt z. B. ein Comfort Panel.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Layout".
3. Aktivieren Sie "Anzeige teilen".
4. Aktivieren Sie "Größenanpassung > Automatisch anpassen".
5. Bei Bedarf ändern Sie die Anzahl der Zeilen für die "Geräteansicht" und die "Detailansicht".

Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt. Die System-Diagnoseanzeige wird in zwei Bereiche geteilt.

Der obere Bereich zeigt die Geräteansicht. Der untere Bereich zeigt die Detailansicht.

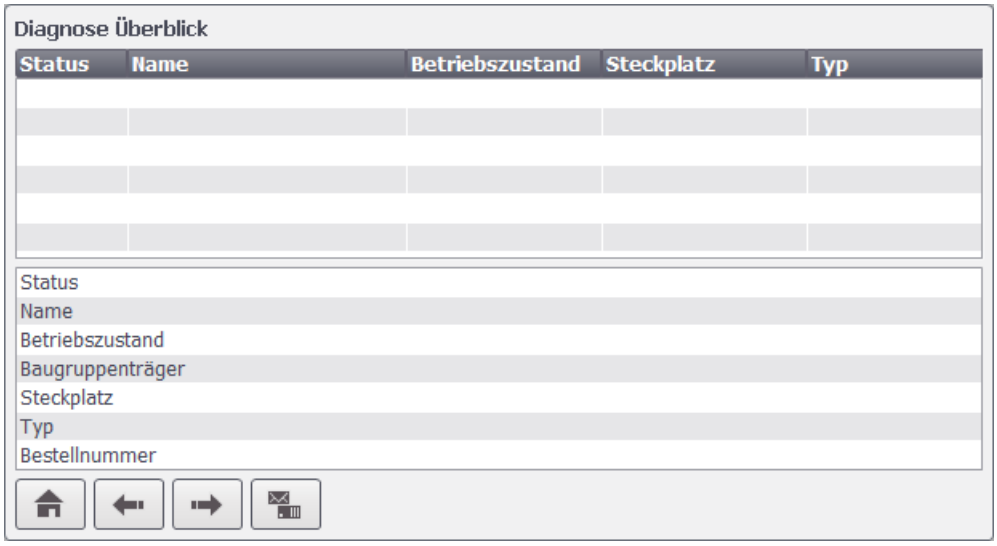

Wenn Sie die Diagnosepuffer-Ansicht öffnen, zeigt der obere Bereich die Übersicht des Diagnosepuffers und der untere Bereich die Details.

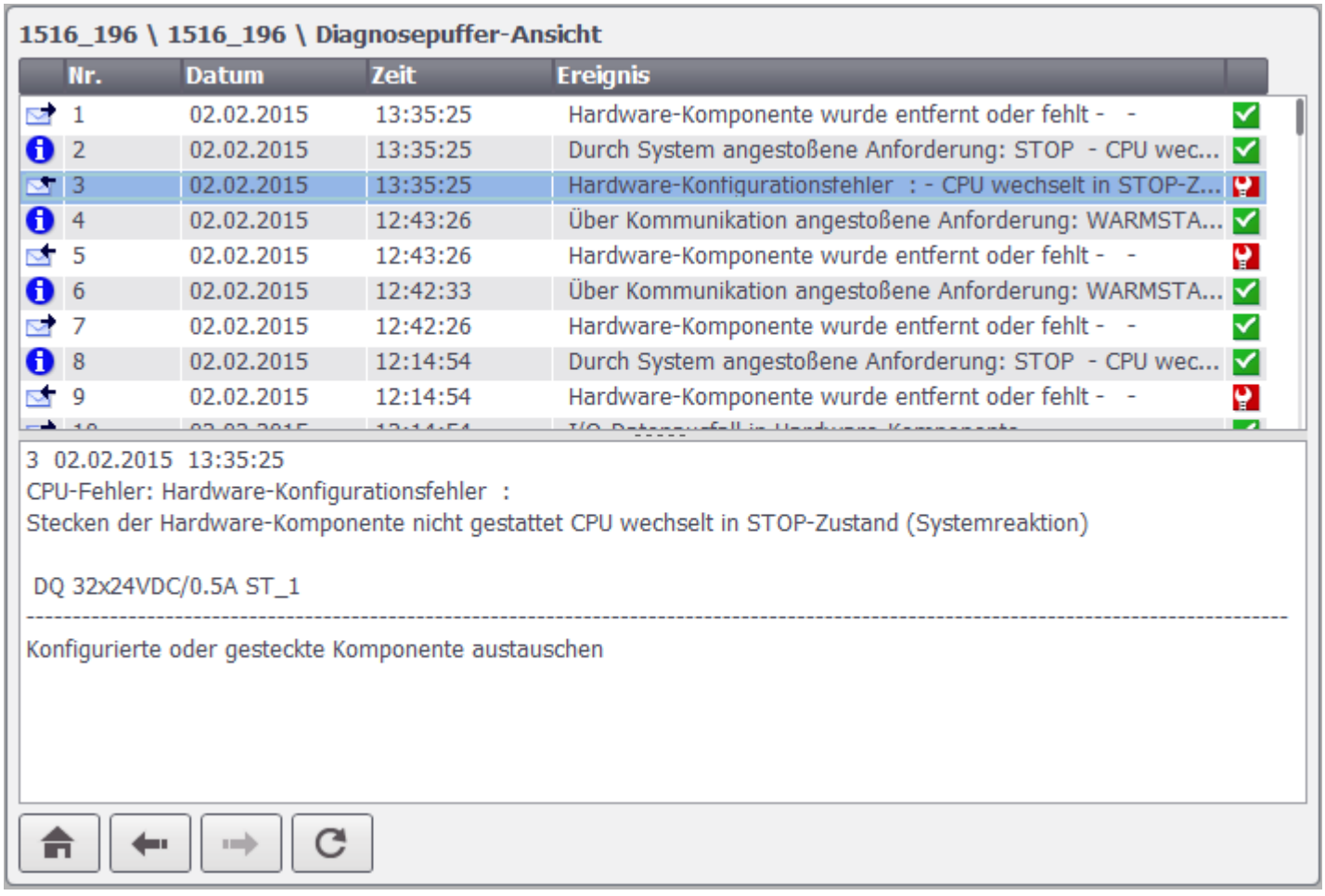

Hinweis

Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbinungen mit den Steuerungen S7-1200 und S7-1500 zur Verfügung.

12.1.2 Systemdiagnose

12.1.2.1 Grundlagen für Runtime Professional

Grundlagen der Systemdiagnose

Einleitung

Mit der Systemdiagnose können Sie Störungen und Fehler in Ihrer gesamten Anlage erkennen. WinCC bietet Ihnen Bedienobjekte für schnelle Fehlerlokalisierung.

Damit die Steuerung Fehlermeldungen an die System-Diagnoseanzeige oder das System-Diagnosefenster übergeben kann, müssen Sie für einige Steuerungen in den Einstellungen die Systemdiagnose aktivieren.

Bei der Kommunikation mit den Steuerungen SIMATIC S7-1500 und SIMATIC S7-1200 ist die Systemdiagnose automatisch aktiviert.

Auf dem Bediengerät ist unter "Runtime-Einstellungen > Meldungen > Steuerungsmeldungen" für die jeweilige HMI-Verbindung die Option "Systemdiagnose" automatisch aktiviert.

System-Diagnoseanzeige

Während Sie mit der Meldeanzeige den Zustand einer Steuerung betrachteten, bietet die System-Diagnoseanzeige einen Überblick über alle verfügbaren Geräte in Ihrer Anlage: Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

System-Diagnoseindikator

Der System-Diagnoseindikator ist ein vordefiniertes grafisches Symbol der Bibliothek, das Sie auf Fehler in Ihrer Anlage hinweist.

Beim Auftreten eines Fehlers ändert der System-Diagnoseindikator sein Aussehen.

GoTo-Schaltfläche

Die GoTo-Schaltfläche ist ein vordefiniertes grafisches Symbol der Bibliothek. Wenn Diagnosemeldungen in der Meldeanzeige erscheinen, können Sie mithilfe der GoTo-Funktion an die Fehlerstelle in der System-Diagnoseanzeige springen.

Siehe auch

Systemdiagnose in der Steuerung aktivieren (Seite 1920)

Ansichten in der Systemdiagnose

Einleitung

In der System-Diagnoseanzeige und im System-Diagnosefenster stehen vier verschiedene Ansichten zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht
- Matrixansicht (nur für Mastersysteme, PROFIBUS, PROFINET)

Geräteansicht

Die Geräteansicht zeigt in tabellarischer Sicht alle verfügbaren Geräte einer Ebene. Durch einen Doppelklick auf ein Gerät öffnen sich entweder die untergeordneten Geräte oder die Detailansicht. Symbole in der ersten Spalte geben Auskunft über den aktuellen Zustand des Geräts.

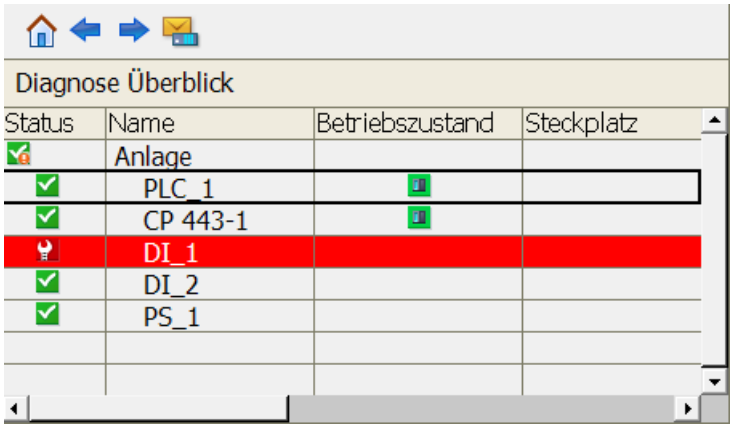

Diagnosepuffer-Ansicht

In der Diagnosepuffer-Ansicht werden die aktuellen Daten aus dem Diagnosepuffer des Geräts angezeigt. Die Diagnosepuffer-Ansicht ist nur in der Geräteansicht aufrufbar.

Um die Diagnosepuffer-Ansicht zu aktualisieren, wählen Sie die Schaltfläche "Aktualisieren" .

319_194 \ Rail_0 \ 319_194 \ Diagnosepuffer-Ansicht

Nr.	Datum	Zeit	Ereignis
1	05.02.2015	11:33:47	Baugruppe ok
2	05.02.2015	11:33:45	Baugruppe gestört oder Wartung erforderlich
3	04.02.2015	15:43:03	Baugruppe gestört oder Wartung erforderlich
4	04.02.2015	15:43:03	Baugruppe gestört oder Wartung erforderlich
5	04.02.2015	15:43:02	Baugruppe gestört oder Wartung erforderlich
6	02.02.2015	13:41:59	Betriebszustandsübergang von ANLAUF nach RUN
7	02.02.2015	13:41:59	Manuelle Neustart (Warmstart)-Anforderung
8	02.02.2015	13:41:59	Betriebszustandsübergang von STOP nach ANLAUF
9	02.02.2015	13:41:52	Dezentrale Peripherie: Station Wiederkehr
10	02.02.2015	13:41:52	Dezentrale Peripherie: Station Wiederkehr

Detailansicht

Die Detailansicht zeigt ausführliche Informationen über das selektierte Gerät und die anliegenden Fehler. In der Detailansicht prüfen Sie, ob die Daten korrekt sind. Fehlertexte können Sie in der Detailansicht nicht sortieren.

1516_196 \ 1516_196

Status	
Name	1516_196
Betriebszustand	
Baugruppenträger	0
Steckplatz	1
Typ	CPU 1516-3 PN/DP
Bestellnummer	6ES7 516-3AN00-0AB0
Adresse	49*
Anlagenbezeichnung	PI 1516_196
Positionskennung	LI 1516_196
Hersteller ID	SIEMENS AG
Hardware Version	81
Profil ID	
spezifische Profildaten	
I&M Datenversion	1.1
Fehlertext	CPU nicht in RUN

Hinweis

Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbindungen mit den Steuerungen S7-1200 und S7-1500 zur Verfügung.

Navigationsschaltflächen

Schaltfläche	Taste	Funktion
	Enter-Taste	Öffnet die untergeordneten Geräte bzw. die Detailansicht, wenn keine Geräte untergeordnet sind.
	Esc-Taste	Öffnet das übergeordnete Gerät oder die Geräteansicht, falls kein übergeordnetes Gerät besteht.
	Home-Taste.	Öffnet die Geräteansicht.
	Funktionstaste F4.	Öffnet die Diagnosepuffer-Ansicht. Nur sichtbar in der Geräteansicht.
	Funktionstaste F5.	Aktualisiert die Diagnosepuffer-Ansicht.

12.1.2.2 Objekte der Systemdiagnose projektieren

Systemdiagnose in der Steuerung aktivieren

Einleitung

Bei der Kommunikation mit den Steuerungen SIMATIC S7-1500 und SIMATIC S7-1200 ist die Systemdiagnose automatisch aktiviert.

Für andere Steuerungen sind die folgenden Einstellungen in der Steuerung erforderlich, damit die Steuerung Fehlermeldungen an die System-Diagnoseanzeige übergeben kann.

Voraussetzungen

- Eine Steuerung z. B. SIMATIC S7-300/400 oder ET 200 ist angelegt.
- Der Editor "Geräte & Netze" ist in der "Gerätesicht" geöffnet.

Vorgehen

1. Selektieren Sie in der Auswahlliste die Steuerung.
2. Klicken Sie auf die Steuerung und wählen Sie im Kontextmenü "Eigenschaften". Im Inspektorfenster werden die Eigenschaften der Steuerung angezeigt.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Allgemein > Systemdiagnose > Allgemein".
4. Aktivieren Sie "Systemdiagnose für diese CPU aktivieren".
5. Selektieren Sie die Steuerung und wählen Sie im Kontextmenü "Übersetzen > Hardwarekonfiguration".

Hinweis

Besonderheit Systemdiagnosemeldungen SIMATIC S7-300/400

Wenn Sie unter "Runtime-Einstellungen > Meldungen > Steuerungsmeldungen > Anzeigeklassen" für die jeweilige HMI-Verbindung die Anzeigeklasse 0 deaktivieren, werden die Systemdiagnosemeldungen von der Steuerung nicht an das Bediengerät übertragen, auch wenn die Option "Systemdiagnose" in der Steuerung und auf dem Bediengerät aktiviert ist.

Ergebnis

Nach dem Übersetzen sind die Einstellungen wirksam. Die Steuerung kann Fehlermeldungen an die System-Diagnoseanzeige übergeben.

Siehe auch

Grundlagen der Systemdiagnose (Seite 1917)

System-Diagnoseanzeige projektieren

Einleitung

Sie fügen eine System-Diagnoseanzeige in Ihrem Projekt ein, um einen Überblick über alle verfügbaren Geräte in Ihrer Anlage haben.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Eine PC Station mit RT Professional ist angelegt.
- Eine HMI-Verbindung zwischen Steuerung und RT Professional ist angelegt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten > Geräte/ Detailansicht".
3. Aktivieren Sie die Spalten, die Sie in der Geräteansicht für Runtime benötigen, z. B. Zustand, Name, Steckplatz.
4. Aktivieren Sie die Spalten, die Sie in der Detailansicht für Runtime benötigen, z. B. Zustand, Name, Anlagenkennzeichen.
5. Aktivieren Sie die Spalten, die Sie in der Diagnosepuffer-Ansicht benötigen z. B. Zustand, Name.
6. Um die Spalten in Runtime zu verschieben, aktivieren Sie "Eigenschaften > Eigenschaften > Layout > Spalteneinstellungen > Spalten verschiebbar".
7. Unter "Eigenschaften > Eigenschaften > Spaltenköpfe" können Sie bei Bedarf die Überschriften für die Spalten ändern.

Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt. In Runtime werden Fehler der gesamten Anlage in der System-Diagnoseanzeige angezeigt.

Anzeige in der System-Diagnoseanzeige teilen

Einleitung

Die System-Diagnoseanzeige bietet auch eine geteilte Ansicht der Anzeige an. So haben Sie die Möglichkeit, die Geräte und die zugehörigen Details auf einen Blick zu sehen.

Voraussetzungen

- Mindestens eine Steuerung ist angelegt.
- In jeder Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Eine PC Station mit RT Professional ist angelegt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Layout".

3. Aktivieren Sie "Anzeige teilen".
4. Bei Bedarf ändern Sie die Anzahl der Zeilen für die "Geräteansicht" und die "Detailansicht".

Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt. Die System-Diagnoseanzeige wird in zwei Bereiche geteilt.

Der obere Bereich zeigt die Geräteansicht. Der untere Bereich zeigt die Detailansicht.

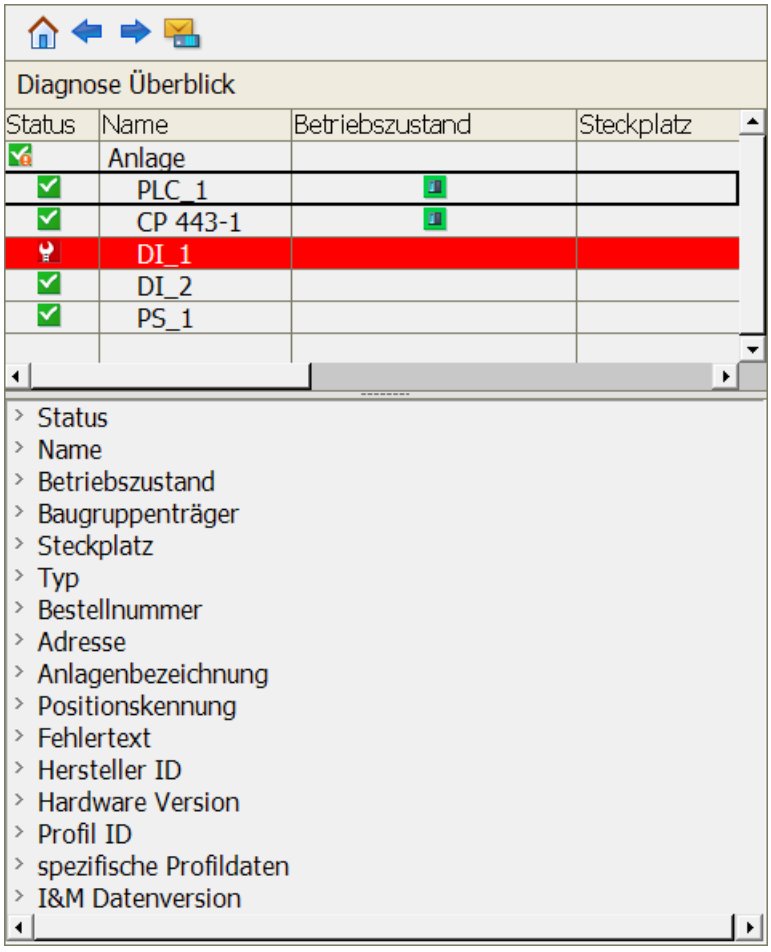

Wenn Sie die Diagnosepuffer-Ansicht öffnen, zeigt der obere Bereich die Übersicht des Diagnosepuffers und der untere Bereich die Details.

Hinweis

Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbinungen mit den Steuerungen S7-1200 und S7-1500 zur Verfügung.

12.1.2.3 Beispiel: Systemdiagnose mit allen Objekten

Beispiel: Handlungsübersicht

Einleitung

- Eine Steuerung ist angelegt.
- In der Steuerung ist die Option "Systemdiagnose für diese CPU aktivieren" aktiviert.
- Eine Verbindung zwischen Steuerung und PC Station ist angelegt.

Projektierungsschritte

Um eine schnelle Sicht auf Fehler zu erhalten, erstellen Sie ein Übersichtsbild mit verschiedenen Objekten zur Anzeige von Diagnosemeldungen.

Das folgende Beispiel zeigt, wie Sie die Objekte der Task Card "Werkzeuge" bzw. "Bibliotheken" in Ihrem Projekt effizient einsetzten.

Das Beispiel ist in mehrere Schritte aufgeteilt:

- Bilder erstellen
  Der Projekteur erstellt mehrere Bilder für die Systemdiagnose:
  - Übersichtbild mit allen Objekten für die Systemdiagnose
  - Bild für die Meldeanzeige
  - Bild für die System-Diagnoseanzeige
- Objekte in Bilder einfügen
  Der Projekteur fügt verschiedene Objekte in die Bilder ein:
  - Meldeanzeige in das Bild "Alarm"
  - Bildfenster für die Anzeige von Meldeanzeige und System-Diagnoseanzeige
- Objekte konfigurieren
  Der Projekteur verschaltet die Objekte, um eine gezielte Navigation zur Fehlerursache in Runtime zu ermöglichen.
  - System-Diagnoseindikator mit Bildfenster der System-Diagnoseanzeige
  - Goto-Schaltfläche mit dem Bildfenster der Meldeanzeige und System-Diagnoseanzeige.

Siehe auch

Beispiel: Bilder erstellen (Seite 1925)

Beispiel: Objekte in Bilder einfügen (Seite 1925)

Beispiel: System-Diagnoseindikator einfügen (Seite 1926)

Beipsiel: Goto-Schaltfläche einfügen (Seite 1928)

Beispiel: Bilder erstellen

Einleitung

Im Folgenden erstellen Sie drei Bilder für die Anzeige verschiedener Objekte.

Voraussetzung

Ein Projekt ist geöffnet.

Ein PC Station mit RT Professional ist angelegt.

Vorgehen

1. Öffnen Sie die Projektnavigation.
2. Klicken Sie im Ordner Bilder auf "Neues Bild hinzufügen".
3. Öffnen Sie das Kontextmenü des Bildes und vergeben Sie den Namen "Overview SysDiag".
4. Erstellen Sie ein weiteres Bild mit dem Namen "Alarm".
5. Öffnen Sie die Task Card "Bibliotheken".
6. Öffnen Sie die Globale Bibliothek "Buttons and Switches > Kopiervorlagen > DiagnosticsButtons > RTProfessional".
7. Klicken Sie in der globalen Bibliothek auf das Objekt "DiagnosticScreen".
8. Ziehen Sie per Drag&Drop das Objekt in die Projektnavigation in den Ordner "Bilder".

Ergebnis

Sie haben drei Bilder in der Projektnavigation angelegt:

- Overview SysDiag
- Alarm
- SystemDiagnostics

Im Bild "SystemDiagnostics" ist eine System-Diagnoseanzeige angelegt.

Siehe auch

Beispiel: Handlungsübersicht (Seite 1924)

Beispiel: Objekte in Bilder einfügen

Einleitung

Im folgenden Beispiel fügen Sie verschieden Objekte aus der Task Card "Werkzeuge" in die Bilder ein.

Voraussetzung

Die folgenden drei Bilder sind angelegt:

- Overview SysDiag
- Alarm
- SystemDiagnostics

Die Task Card "Werkzeuge" ist geöffnet.

Vorgehen

1. Öffnen Sie das Bild "Alarm".
2. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" eine "Meldeanzeige" in das Bild.
3. Öffnen Sie das Bild "Overview SysDiag".
4. Ziehen per Drag&Drop aus der Task Card "Werkzeuge" ein "Bildfenster" in das Bild.
5. Vergeben Sie im Inspektofenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" den Namen "Alarm".
6. Wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Inhalt > Angezeigtes Bild" das Bild "Alarm" aus.
7. Fügen Sie ein zweites Bildfenster ein mit dem Namen "SysDiag".
8. Wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein > Inhalt > Angezeigtes Bild" das Bild "SystemDiagnostics".

Ergebnis

Sie haben im Bild "Alarm" eine Meldeanzeige eingefügt. Die Meldeanzeige hat den unveränderten Namen "Meldeanzeige_1".

Im Bild "Overview SysDiag" haben Sie zwei Bildfenster angelegt. Das erste Bildfenster hat den Namen "Alarm" und zeigt die Meldeanzeige an. Das zweite Bildfenster hat den Namen "SysDiag" und zeigt die System-Diagnoseanzeige an.

Siehe auch

Beispiel: Handlungsübersicht (Seite 1924)

Beispiel: System-Diagnoseindikator einfügen

Einleitung

Der System-Diagnoseindikator ist ein vordefiniertes grafisches Symbol der Bibliothek, das Sie auf Fehler in Ihrer Anlage hinweist.

Wenn ein Fehler auftritt, ändert sich das Aussehen des Objekts. Das Bibliotheksobjekt kann zwei Zustände anzeigen:

- Kein Fehler
- Fehler

Voraussetzung

- Die Task Card "Bibliotheken" ist geöffnet.
- Die globale Bibliothek "Buttons and Switches > Kopiervorlagen > DiagnosticsButtons > RT Professional" ist geöffnet.
- Die folgenden drei Bilder sind angelegt:
  - Overview SysDiag
  - Alarm
  - SystemDiagnostics
- Das Bildfenster "SysDiag" ist angelegt und mit dem Bild "DiagnosticScreen" verbunden.

Vorgehen

1. Selektieren Sie in der Bibliothek das Objekt "DiagnosticsIndicator".
2. Ziehen Sie das Bibliotheksobjekt per Drag&Drop in das Bild "Overview SysDiag". Das Bibliotheksobjekt wird eingefügt.
3. Selektieren Sie das Bibliotheksobjekt.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Für das Ereignis "Klicken" sind drei Systemfunktionen voreingestellt.

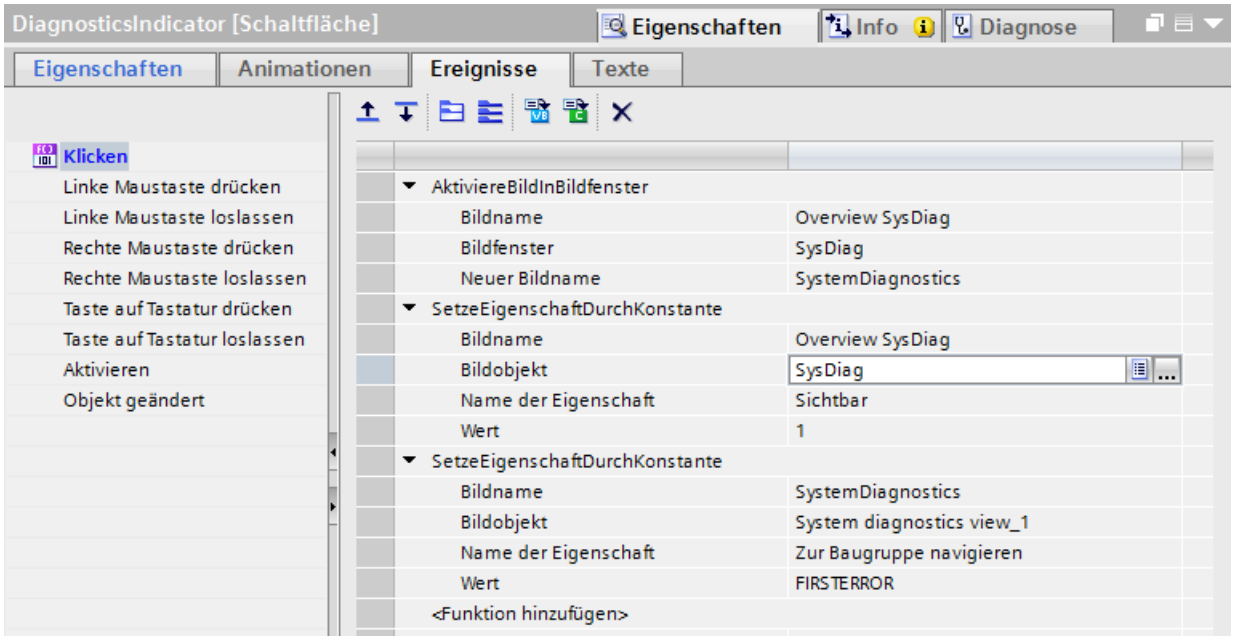

5. Wählen Sie in der Systemfunktion "AktiviereBildInBildfenster" für den Parameter "Bildname" "SysDiag".
6. Wählen Sie in der Systemfunktion "SetzeEigenschaftDurchKonstante" für den Parameter "Bildobjekt" "SysDiag"

Ergebnis

Der System-Diagnoseindikator ist im Projekt eingefügt und mit dem Bildfenster "SysDiag" verbunden.

Wenn in Runtime eine Fehlermeldung auftritt, ändert der System-Diagnoseindikator sein Aussehen. Wenn Sie auf den System-Diagnoseindikator klicken, öffnet sich das Bildfenster SysDiag, in dem eine System-Diagnoseanzeige projekiert ist. Die System-Diagnoseanzeige zeigt die Detailansicht des betroffenen Geräts.

Siehe auch

Beispiel: Handlungsübersicht (Seite 1924)

Beipsiel: Goto-Schaltfläche einfügen

Voraussetzungen

- Die Task Card "Bibliotheken" ist geöffnet.
- Die globale Bibliothek "Buttons and Switches > Kopiervorlagen > DiagnosticsButtons > RT Professional" ist geöffnet.
- Die folgenden drei Bilder sind angelegt:
  - Overview SysDiag
  - Alarm
  - SystemDiagnostics
- Das Bildfenster "Alarm" ist angelegt und mit dem Bild "Alarm" verbunden.
- Das Bildfenster "SysDiag" ist angelegt und mit dem Bild "DiagnosticScreen" verbunden.

Goto-Schaltfläche im Bild projektieren

1. Öffnen Sie das Bild "SystemDiagnostic".
2. Öffnen Sie die globale Bibliothek "Buttons and Switches > Kopiervorlagen > DiagnosticsButtons".
3. Selektieren Sie in der Bibliothek das Objekt "DiagnosticsGoto".
4. Ziehen Sie das Bibliotheksobjekt per Drag&Drop in das Bild "Overview SysDiag". Das Bibliotheksobjekt wird eingefügt.
5. Selektieren Sie das Bibliotheksobjekt.

6. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
   Für das Ereignis sind vier Systemfunktionen voreingestellt.

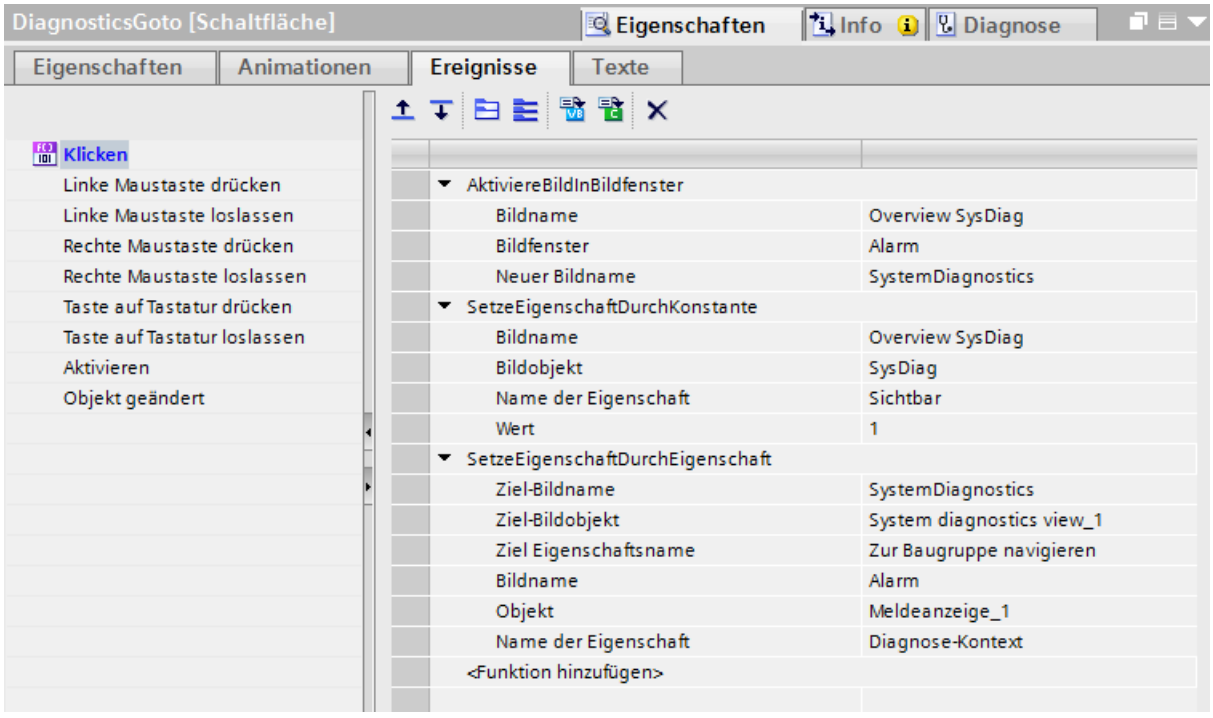

7. Wählen Sie in der Systemfunktion "AktiviereBildInBildfenster" für den Parameter "Bildname" "SysDiag".
8. Wählen Sie in der Systemfunktion "SetzeEigenschaftDurchKonstante" für den Parameter "Bildobjekt" "SysDiag"
9. Wählen Sie in der Systemfunktion "SetzteEigenschaftDurchEigenschaft" folgende Einstellungen:

- Bildname: Alarm
- Objekt: Meldeanzeige_1

Ergebnis

Sie haben die Goto-Schaltfläche projektiert und mit der Meldanzeige und der System-Diagnoseanzeige verknüpft.

Wenn in Runtime im Bildfenster "Alarms" die Meldeanzeige eine Fehlermeldung ausgibt, klicken Sie auf die Goto-Schaltfläche.

Im Bildfenster "SysDiag" wird die System-Diagnoseanzeige angezeigt. Das Gerät mit dem zugehörigen Fehler wird in der Detailansicht geöffnet.

Siehe auch

Beispiel: Handlungsübersicht (Seite 1924)

12.2 Überwachungsfunktionen für SIMATIC IPCs

Hardware-Überwachung für SIMATIC IPCs

Alle SIMATIC IPCs unterstützen Überwachungsfunktionen. Ein Teil der Überwachungsfunktionen kann im TIA Portal angezeigt werden, wenn Sie sich online mit dem IPC verbinden. Im Diagnose-Editor erhalten Sie Informationen zu Betriebszuständen folgender Gruppen:

- Lüfter
- Temperaturen
- Spannungen
- Laufwerke/Speichermedien

Das folgende Bild zeigt die Übersicht über alle Gruppen im Diagnose-Editor.

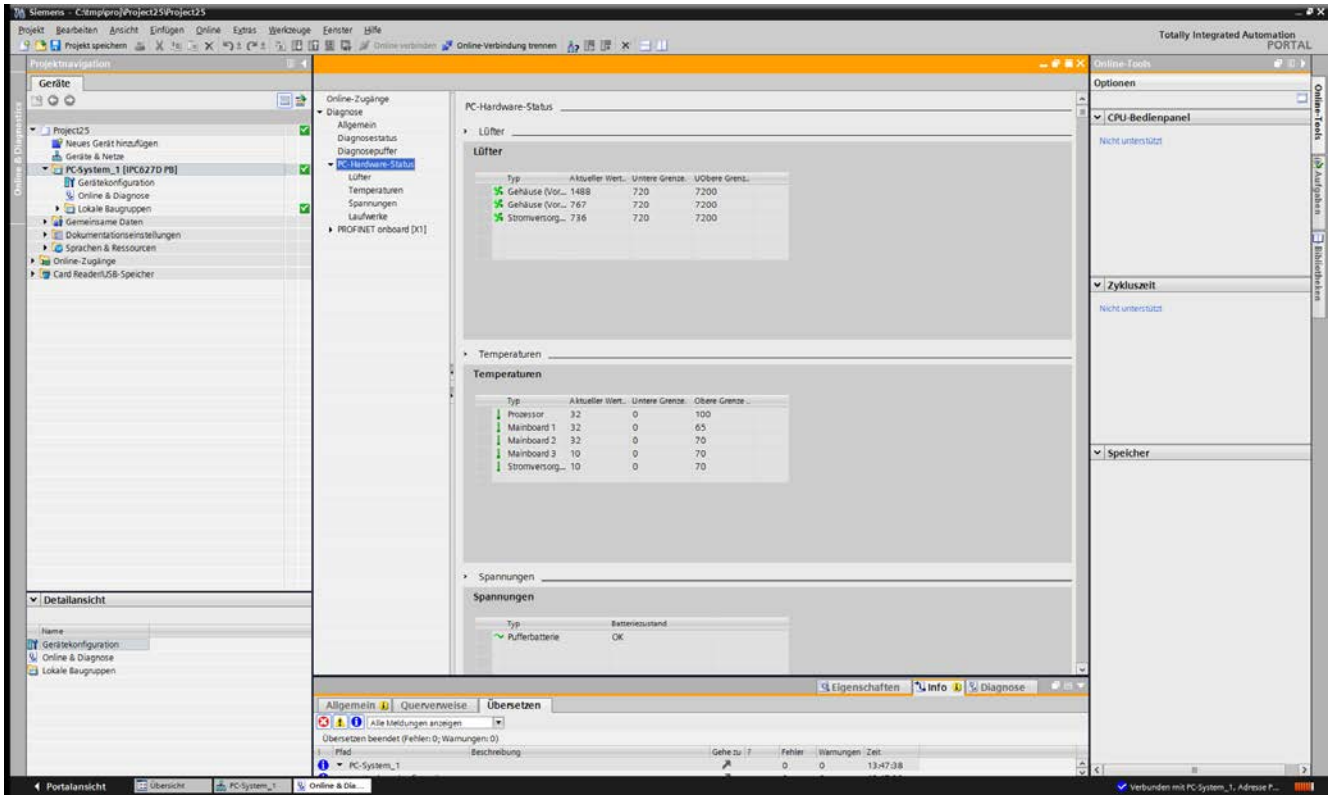

Sie können im Diagnose-Editor zwischen der Gesamtansicht und der Einzelansicht wählen.

Gruppe "Temperaturen"

Die Gruppe "Temperaturen" zeigt die Temperaturen verschiedener Komponenten an.

Gruppe "Lüfter"

Die Gruppe "Lüfter" zeigt die Zustände der vorhandenen Lüfter (geräteabhängig) an. Die Drehzahlen sind in Umdrehungen pro Minute (U/Min) angegeben.

Gruppe "Spannungen"

Die Gruppe "Spannungen" zeigt die Zustände der vorhandenen Spannungssensoren (geräteabhängig) an. Es wird die Spannung der CMOS-Batterie überwacht. Dieser Sensor liefert eine Aussage darüber, ob die Spannung in Ordnung (=OK), kritisch oder fehlerhaft ist.

Gruppe "Laufwerke"

Die Gruppe "Laufwerke" zeigt die Zustände der vorhandenen Laufwerke/Speichermedien (geräteabhängig) an. Hierbei werden die folgenden Speichermedien unterstützt:

- ATA-Festplatten
- CompactFlash Karten
- SCSI-Festplatten
- USB-Festplatten bzw. USB-Sticks
- Festplatten an einem INTEL RAID-Adapter

Für jedes Laufwerk ist in den Eigenschaften angegeben, ob es ein S.M.A.R.T.-fähiges und/oder RAID-Laufwerk ist. In diesem Fall wird eine Statusinformation angezeigt.

Grenzwerte

Für die beiden Gruppen Lüfter und Temperaturen sind jeweils untere und obere Grenzwerte definiert. Die Grenzwerte sind vom IPC abhängig und können nicht verändert werden.

Farbcode

Die Icons neben dem jeweiligen Anzeigeobjekt ändern abhängig vom Zustand des Objekts die Farbe. Für alle Gruppen gilt:

- Grün = OK
- Rot = Grenzwert über- bzw. unterschritten bzw. Zustand des Objekts kritisch.

Verhalten bei Abweichungen

Wird der Grenzwert über- bzw. unterschritten oder erreicht eine der Gruppen einen kritischen Zustand, wird ein Eintrag in den Diagnosepuffer ausgelöst. Gleichzeitig wechselt das Icon die Farbe auf Rot.

Weitere Informationen zu Diagnosefunktionen der IPCs finden Sie im SIOS Portal (https://support.industry.siemens.com/cs/de/de/view/39129913)

12.3 Maschinen und Anlagen mit ProDiag überwachen

12.3.1 Grundlagen der Überwachung mit ProDiag

Einleitung

Die TIA Portal Funktionalität ProDiag dient zur Überwachung und Ermittlung von Fehlern, die in Ihrer Anlage oder Maschine auftreten. Mit ProDiag zeigen Sie die Fehlerart, die Fehlerursache und den Fehlerort am Bediengerät an.

Verwendung

Sie verwenden ProDiag-Funktionen, um Ihre Anlage zu überwachen und auf einem Bediengerät zu visualisieren. Hauptziel von ProDiag ist die Reduzierung von Stillstandszeiten und Produktionsausfällen nach dem Auftreten eines Fehlers, sowie die Vermeidung von Fehlern durch rechtzeitige Warnungen. Diagnose- und Anzeigeobjekte liefern gezielte Hinweise für den Bediener bei der Fehlerbehebung und zeigen die Abläufe auf einem Bediengerät vor Ort.

Prinzip

In STEP 7 legen Sie die Überwachungen für die Operanden an und projektieren die Einstellungen nach Ihren Anforderungen. Beim Auftreten eines Fehlers wird eine Überwachungsmeldung nach den von Ihnen projektierten Kriterien ausgegeben. Die projektierten Überwachungen werden im voreingestellten ProDiag-Funktionsbaustein gespeichert. Sie können die automatisch erzeugten ProDiag-FBs nutzen oder eigene ProDiag-FBs erstellen und konfigurieren.

Mit WinCC visualisieren Sie die ProDiag-Funktionen für die Anzeige auf einem HMI-Bediengerät. Sie projektieren Bildobjekte, die einen Überblick über den aktuellen Status der Anlage ermöglichen und im Fall eines Fehlers den betroffenen Programmcode, bzw. den Schrittablauf, anzeigen.

12.3.2 Grundlagen der Überwachung von Operanden

Überwachung von Operanden

Mit einer Überwachung überwachen Sie den Zustand einer booleschen Variablen. Wenn in Ihrer Anlage oder Maschine ein Ereignis auftritt, wird dadurch die projektierte Überwachung ausgelöst und der Zustand des Operanden wird in ein Fehlerbit in einem ProDiag-Baustein geschrieben. Es stehen Ihnen folgende sieben Überwachungsarten zur Verfügung:

- Operandenüberwachung
- Verriegelungsüberwachung
- Aktionsüberwachung

- Reaktionsüberwachung
- Positionsüberwachung
- Textmeldung
- Fehlermeldung

Kategorien und Unterkategorien

Für jede Überwachung können Sie Kategorien und Unterkategorien festlegen. Mithilfe von acht Kategorien stufen Sie die auftretenden Ereignisse nach ihrer Dringlichkeit ein. Unterkategorien dienen dazu, Informationen noch weiter zu untergliedern. Kategorien und Unterkategorien können Sie in einer Überwachung miteinander kombinieren.

Kategorien, Unterkategorien und Meldetexte für die Überwachungen konfigurieren Sie in der Projektnavigation unter "Gemeinsame Daten > Überwachungseinstellungen". Weitere Informationen zu Überwachungen und Kategorien finden Sie im Kapitel "PLC programmieren > Maschinen und Anlagen mit ProDiag überwachen".

Überwachung in HMI verfolgen

Verfolgen Sie in WinCC den Zustand der projektierten Überwachungen mithilfe des Objekts "ProDiag-Übersicht".

Wenn ein Ereignis in Ihrer Anlage auftritt, leuchten in der ProDiag-Übersicht die entsprechenden Symbole der Überwachungskategorie und der Überwachungsart. Über die konfigurierten Schaltflächen springen Sie direkt zur Fehlermeldung oder zur Anzeige des betroffenen Netzwerks in der PLC-Code-Anzeige.

Den Status der GRAPH-Ablaufkette eines GRAPH-Datenbausteins zeigen Sie im Objekt "GRAPH-Übersicht" an. Falls in einem der Schritte ein GRAPH-Überwachungsfehler auftritt, leuchtet das Fehlersymbol in der GRAPH-Übersicht und der gestörte Schritt wird visuell hervorgehoben.

Mit dem Objekt "PLC-Code-Anzeige" zeigen Sie das Anwenderprogramm in Programmiersprachen KOP, FUP und GRAPH an Ihrem Bediengerät an.

Die Funktionen Erstwerterfassung und Kriterienanalyse ermöglichen Ihnen die Aufzeichnung der Werte zum Zeitpunkt der Störung und schnelle Erkennung der fehlerhaften Operanden mit dem Objekt "Kriterienanalyse-Anzeige".

Vorteile

Mit ProDiag haben Sie die Möglichkeit, Überwachungen zu projektieren und Ihre Anlage zu überwachen, ohne den Code des Anwenderprogramms zu verändern.

Die Anlagendiagnose führen Sie an Ihrem Bediengerät durch. Die Daten werden automatisch synchronisiert um die Anzeige an Ihrem Bediengerät immer aktuell zu halten.

Weitere Informationen zur integrierten Diagnose Ihrer Maschinen und Anlagen mit SIMATIC ProDiag finden Sie im Siemens YouTube Channel:

Erhöhte Verfügbarkeit Ihrer Maschinen und Anlagen mit SIMATIC ProDiag (https://youtu.be/cS-KVa9AVYg)

Ein Anwenderbeispiel für die Maschinen- und Anlagendiagnose mit ProDiag finden Sie im Siemens Industry Online Support mit der Beitrags-ID 109740151:

Maschinen- und Anlagendiagnose mit ProDiag (https://support.industry.siemens.com/cs/ww/de/view/109740151)

Siehe auch

Maschinen- und Anlagendiagnose mit ProDiag (https://support.industry.siemens.com/cs/ww/de/view/109740151)

12.3.3 Voraussetzungen und Lizenzierung

Einleitung

Sie projektieren die ProDiag-Überwachungen im TIA Portal mit STEP 7 Professional und erstellen die Bildobjekte zur Überwachung und Diagnose mit WinCC. Um die Funktionalität ProDiag und entsprechende Bildobjekte nutzen zu können, benötigen Sie eine Lizenz.

Softwarevoraussetzungen

Zum Projektieren von ProDiag-Überwachungen benötigen Sie folgende Produkte:

- TIA Portal STEP 7 Professional
- TIA Portal WinCC
  - Comfort/Advanced ab V14 oder
  - Professional ab V14

Hardwarevoraussetzungen

ProDiag-Funktionalität steht für die CPUs der Baureihe S7-1500 mit der Firmware ab Version 2.0 zur Verfügung.

Die Objekte zur Überwachung und Diagnose von Anlagen stehen für folgende HMI-Geräte zur Verfügung:

- Comfort Panels und KTP Mobile Panels*
- RT Advanced
- RT Professional

* Das Objekt "PLC-Code-Anzeige" steht auf Comfort Panels und KTP Mobile Panels ab Displaygröße 7" zur Verfügung

Hinweis

Die Objekte zur Überwachung und Diagnose von Anlagen können Sie unter den in der CPU projektierten Schutzstufen "Vollzugriff" und "Lesezugriff" verwenden.

Die Verwendung von ProDiag-Objekten unter den Schutzstufen "HMI-Zugriff" und "Kein Zugriff“ ist nicht möglich.

Lizenzierung von ProDiag-Überwachungen

Die Anzahl der ProDiag-Überwachungen, die Sie mit STEP 7 Professional projektieren, wird lizenziert. Für die ersten 25 Überwachungen benötigen Sie keine Lizenz, für weitere Überwachungen müssen Lizenzen genutzt werden.

Anzahl der Überwachungen	<= 25	<= 250	<= 500	<= 750	<= 1000	> 1000
Anzahl der Lizenzen	Keine	1	2	3	4	5

Lizenzierung von ProDiag-Objekten

Um die Objekte zur Diagnose und Überwachung in Verbindung mit der ProDiag-Überwachung in Ihrem Programm zu nutzen, benötigen Sie eine ProDiag-Lizenz.

12.3.4 Überwachungen visualisieren

12.3.4.1 Objekte zur Überwachung und Diagnose von Anlagen

Einleitung

WinCC bietet Ihnen folgende Objekte für die Anzeige des aktuellen Zustands von Überwachungen und Diagnose von Fehlern im Programmcode:

- ProDiag-Übersicht
- GRAPH-Übersicht
- PLC-Code-Anzeige
- Kriterienanalyse-Anzeige

ProDiag-Übersicht

Das Objekt "ProDiag-Übersicht" bietet Ihnen einen Überblick über den aktuellen Zustand der projektierten Überwachungen an.
Die ProDiag-Übersicht steht auf Comfort Panels, KTP Mobile Panels, RT Advanced und RT Professional zur Verfügung.

GRAPH-Übersicht

Das Objekt „GRAPH-Übersicht“ dient zur Anzeige des aktuellen Programmstatus für ausgeführte Schritte der GRAPH-Ablaufkette.
Die GRAPH-Übersicht steht auf Comfort Panels, KTP Mobile Panels, RT Advanced und RT Professional zur Verfügung.

PLC-Code-Anzeige

Das Objekt "PLC-Code-Anzeige" dient zur Anzeige des aktuellen Programmstatus von Anwenderprogrammen, die in den Programmiersprachen KOP, FUP oder GRAPH programmiert worden sind.

Die PLC-Code-Anzeige steht auf Comfort Panels, KTP Mobile Panels ab Displaygröße 7", RT Advanced und RT Professional zur Verfügung.

Kriterienanalyse-Anzeige

Das Objekt "Kriterienanalyse-Anzeige" dient zur Anzeige der fehlerhaften Operanden im Anwenderprogramm, die für eine ausgewählte Meldung durch Kriterienanalyse ermittelt wurden.
Die Kriterienanalyse-Anzeige steht auf Comfort Panels, KTP Mobile Panels, RT Advanced und RT Professional zur Verfügung.

12.3.4.2 Zusammenwirkung der Objekte zur Überwachung und Diagnose

Einleitung

Mit den Objekten zur Überwachung und Diagnose von Anlagen haben Sie folgende Möglichkeiten:

- Sie stellen den Status der im Anwenderprogramm projektierten Überwachungen auf einem HMI-Gerät dar,
- Sie werten die aktuellen Meldungen aus und zeigen das Netzwerk an, welches die Meldungen verursacht,
- Sie zeigen den Programmcode am HMI-Gerät an.

Je nach Anwendung zeigen Sie auf dem Bediengerät folgende Informationen an:

- Mit dem Objekt "ProDiag-Übersicht" zeigen Sie den aktuellen Status der ProDiag-Überwachungen an.
- Mit dem Objekt "GRAPH-Übersicht" zeigen Sie den aktuellen Status der GRAPH-Ablaufkette an.
- Mit dem Objekt "Kriterienanalyse-Anzeige" zeigen Sie die fehlerhaften Operanden zu einer ausgewählten GRAPH- oder ProDiag-Meldung.

Zusammenwirkung der ProDiag-Objekte

Die nachfolgende Abbildung stellt die Zusammenwirkung von Objekten zur Visualisierung der ProDiag Überwachungen dar:

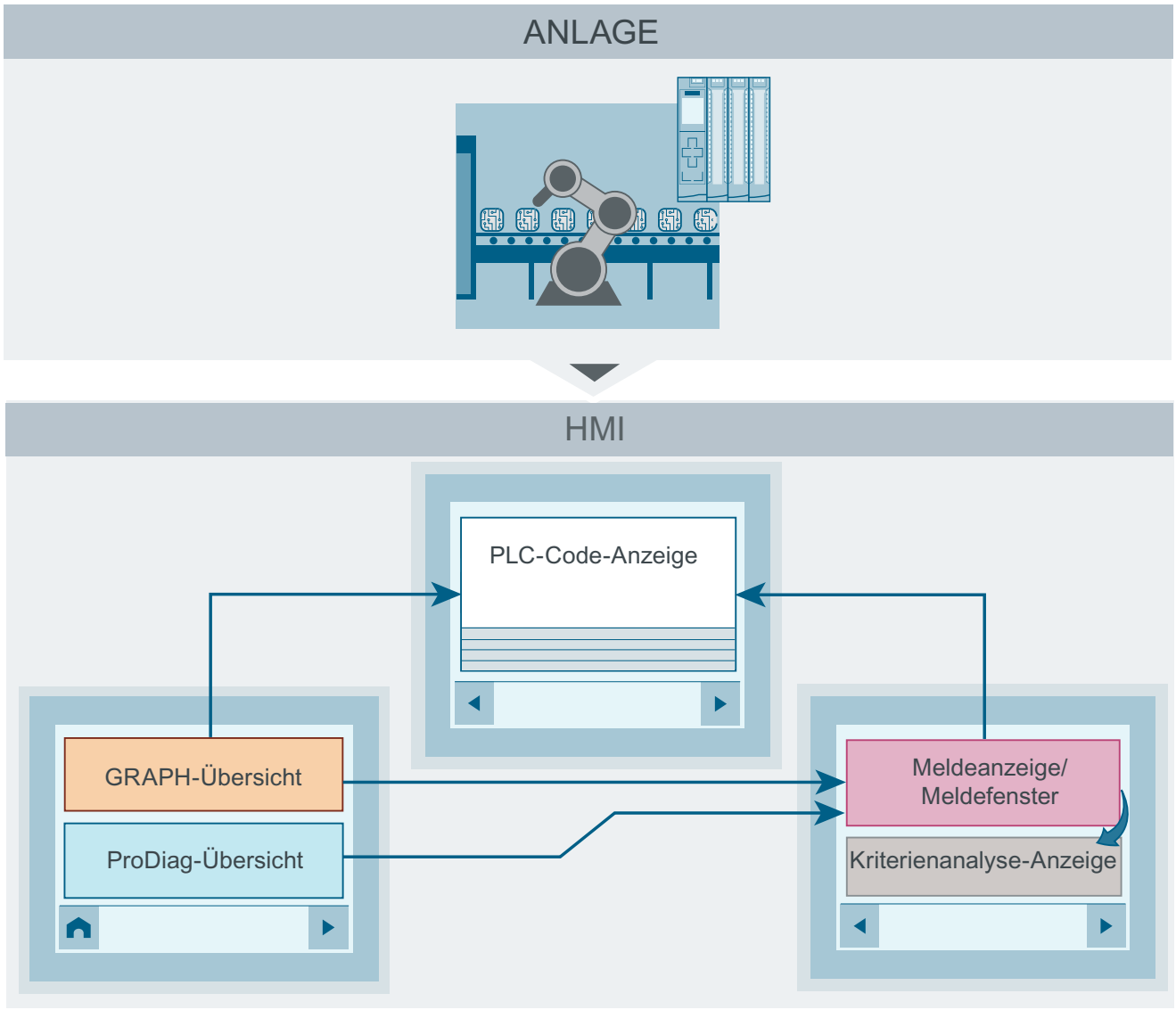

Die Visualisierung der Funktionalität ProDiag am Bediengerät setzt sich aus folgenden Aktionen zusammen:

1. Programmstatus anzeigen
   - Status der Überwachungen mit der ProDiag-Übersicht anzeigen
     Am Bediengerät wird die Verbindung zur Steuerung und dem verbundenen ProDiag-FB aufgebaut und der aktuelle Status der Überwachungen angezeigt. Sobald ein Ereignis auftritt, leuchtet das entsprechende Symbol in der ProDiag-Übersicht.
     Die Schaltfläche "Meldeanzeige" ermöglicht den Sprung in die Meldeanzeige oder Meldefenster mit den Details zum aufgetretenen Ereignis. Um in der Meldeanzeige/im Meldefenster die überwachungsrelevanten Meldungen leichter zu finden, ist es empfehlenswert, die angezeigten Meldungen mithilfe eines Filters einzugrenzen.
   - Ablauf der GRAPH-Kette mit der GRAPH-Übersicht anzeigen
     Sobald die Verbindung zur Steuerung aufgebaut ist, wird in Runtime der aktuell ausgeführte Schritt der GRAPH-Ablaufkette dargestellt. Mithilfe von Bedienschaltflächen navigieren Sie zum weiteren Schritt oder zum Initialschritt der Ablaufkette.
     Die Schaltfläche "Meldeanzeige" ermöglicht den Sprung in die Meldeanzeige oder Meldefenster mit den Details zum aufgetretenen Ereignis.
     Die Schaltfläche "PLC-Code-Anzeige" ermöglicht den Sprung in die PLC-Code-Anzeige, z. B. um einen Fehler im Ablauf der GRAPH-Kette anzuzeigen.
2. Aktuelle Meldungen in der Meldeanzeige oder im Meldefenster anzeigen
   In der Meldeanzeige oder im Meldefenster werden in Runtime die auftretenden Meldungen angezeigt.

   Hinweis

   Anzeige von Meldungen der Meldeklassen "Acknowledgement" und "No Acknowledgement"

   Um die Überwachungsmeldungen der systemeigenen oder selbstdefinierten Meldeklassen in Ihrer Meldeanzeige oder in Ihrem Meldefenster anzuzeigen, aktivieren Sie die Anzeige dieser Meldeklassen im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein".

   In der Meldeanzeige/im Meldefenster sind die Meldeklassen der Überwachungsmeldungen erst nach dem Projektieren der Überwachungen und Übersetzen der Steuerung aktivierbar.

   Die Systemfunktion "AktivierePLCCodeAnzeige" ermöglicht den Sprung von der letzten aktiven oder einer ausgewählten Meldung zur PLC-Code-Anzeige mit dem betroffenen Programmcode.
   Zusätzlich haben Sie mit den Funktionen Erstwerterfassung und Kriterienanalyse die Möglichkeit, die Werte zum Zeitpunkt der Störung zu erfassen und die fehlerhaften Operanden im Programm im Objekt "Kriterienanalyse-Anzeige" auszugeben.
3. Programmcode in der PLC-Code-Anzeige anzeigen
   Sobald die Verbindung zur Steuerung aufgebaut wurde, zeigt die PLC-Code-Anzeige je nach Projektierung den aktuellen Status des Netzwerks an:
   - In Runtime wird das KOP/FUP Netzwerk und der Programmcode angezeigt. In der Symboltabelle werden die Symbolnamen, absolute Adressen der Symbole und zugehörige Kommentare dargestellt.
   - In Runtime wird die GRAPH-Ablaufkette dargestellt. In der Detailansicht werden die Transitionen der Schritte angezeigt.

Sie haben die Möglichkeit, eine Schaltfläche zu projektieren, die im Fall eines Fehlers ihr Aussehen verändert und Ihnen den Einsprung in die projektierte PLC-Code-Anzeige ermöglicht. Legen Sie dafür eine Schaltfläche fest und projektieren Sie daran die Systemfunktion "AktivierePLCCodeAnzeige ". Dynamisieren Sie die Gestaltung (z.B. Hintergrundfarbe) der Schaltfläche mithilfe der Steuervariable für PLC-Code-Anzeige, die Sie zuvor mit der Meldeanzeige verschaltet haben, sodass z.B. beim Kommen einer ProDiag-Meldung die Schaltfläche ihre Farbe verändert.

Übertragung von Sprachen von CPU auf HMI-Bediengerät

Im TIA Portal werden die Texte für die ProDiag-Funktionalität, z.B. Symbolnamen und Kommentare in einer PLC-Code-Anzeige sowie die Texte der Überwachungsmeldungen, aus dem Anwenderprogramm der Steuerung auf das Bediengerät automatisch geladen. Achten Sie bei den mehrsprachigen Projekten darauf, dass die Sprachen, die in der Steuerung zur Verfügung stehen und die Runtime-Sprachen des Bediengeräts übereinstimmen. Nur wenn die Sprachen übereinstimmen, können die Meldetexte auf dem Bediengerät in den gewünschten Sprachen angezeigt werden.

Weitere Informationen zu mehrsprachigen Projekten finden Sie unter "Geräte und Netzte konfigurieren > Zusatzinformationen zu Konfigurationen > Arbeitsweise von S7-1500 CPUs > Einstellen des Betriebsverhaltens > CPU-Eigenschaften > Mehrsprachigkeit".

12.3.4.3 Zusammenwirkung der Objekte zur Überwachung und Diagnose

Einleitung

Mit den Objekten zur Überwachung und Diagnose von Anlagen haben Sie folgende Möglichkeiten:

- Sie stellen den Status der im Anwenderprogramm projektierten Überwachungen auf einem HMI-Gerät dar,
- Sie werten die aktuellen Meldungen aus und zeigen das Netzwerk an, welches die Meldungen verursacht,
- Sie zeigen den Programmcode am HMI-Gerät an.

Je nach Anwendung zeigen Sie auf dem Bediengerät folgende Informationen an:

- Mit dem Objekt "ProDiag-Übersicht" zeigen Sie den aktuellen Status der ProDiag-Überwachungen an.
- Mit dem Objekt "GRAPH-Übersicht" zeigen Sie den aktuellen Status der GRAPH-Ablaufkette an.
- Mit dem Objekt "Kriterienanalyse-Anzeige" zeigen Sie die fehlerhaften Operanden zu einer ausgewählten GRAPH- oder ProDiag-Meldung.

Zusammenwirkung der ProDiag-Objekte

Die nachfolgende Abbildung stellt die Zusammenwirkung von Objekten zur Visualisierung der ProDiag Überwachungen dar:

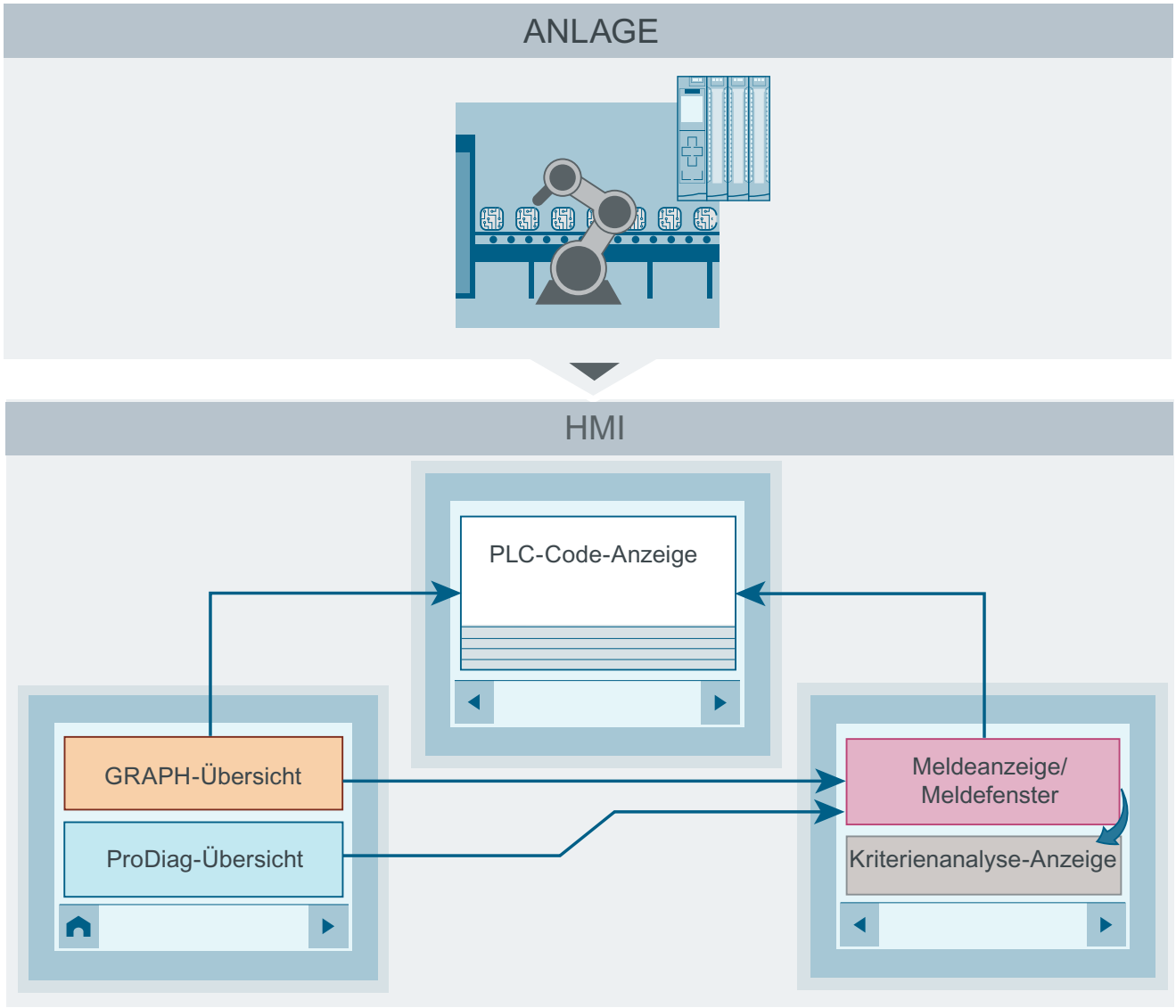
ANLAGE
HMI
PLC-Code-Anzeige
GRAPH-Übersicht
ProDiag-Übersicht
Meldeanzeige/
Meldefenster
Kriterienanalyse-Anzeige

Die Visualisierung der Funktionalität ProDiag am Bediengerät setzt sich aus folgenden Aktionen zusammen:

1. Programmstatus anzeigen
   - Status der Überwachungen mit der ProDiag-Übersicht anzeigen
     Am Bediengerät wird die Verbindung zur Steuerung und dem verbundenen ProDiag-FB aufgebaut und der aktuelle Status der Überwachungen angezeigt. Sobald ein Ereignis auftritt, leuchtet das entsprechende Symbol in der ProDiag-Übersicht.
     Die Schaltfläche "Meldeanzeige" ermöglicht den Sprung in die Meldeanzeige oder Meldefenster mit den Details zum aufgetretenen Ereignis. Um in der Meldeanzeige/im Meldefenster die überwachungsrelevanten Meldungen leichter zu finden, ist es empfehlenswert, die angezeigten Meldungen mithilfe eines Filters einzugrenzen.
   - Ablauf der GRAPH-Kette mit der GRAPH-Übersicht anzeigen
     Sobald die Verbindung zur Steuerung aufgebaut ist, wird in Runtime der aktuell ausgeführte Schritt der GRAPH-Ablaufkette dargestellt. Mithilfe von Bedienschaltflächen navigieren Sie zum weiteren Schritt oder zum Initialschritt der Ablaufkette.
     Die Schaltfläche "Meldeanzeige" ermöglicht den Sprung in die Meldeanzeige oder Meldefenster mit den Details zum aufgetretenen Ereignis.
     Die Schaltfläche "PLC-Code-Anzeige" ermöglicht den Sprung in die PLC-Code-Anzeige, z. B. um einen Fehler im Ablauf der GRAPH-Kette anzuzeigen.
2. Aktuelle Meldungen in der Meldeanzeige oder im Meldefenster anzeigen
   In der Meldeanzeige oder im Meldefenster werden in Runtime die auftretenden Meldungen angezeigt.

   Hinweis

   Anzeige von Meldungen der Meldeklassen "Acknowledgement" und "No Acknowledgement"

   Um die Überwachungsmeldungen der systemeigenen oder selbstdefinierten Meldeklassen in Ihrer Meldeanzeige oder in Ihrem Meldefenster anzuzeigen, aktivieren Sie die Anzeige dieser Meldeklassen im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein".

   In der Meldeanzeige/im Meldefenster sind die Meldeklassen der Überwachungsmeldungen erst nach dem Projektieren der Überwachungen und Übersetzen der Steuerung aktivierbar.

   Die Systemfunktion "ZeigePLCCodeAnzeigeVonMeldung" ermöglicht den Sprung von der letzten aktiven oder einer ausgewählten Meldung zur PLC-Code-Anzeige mit dem betroffenen Programmcode.
   Zusätzlich haben Sie mit den Funktionen Erstwerterfassung und Kriterienanalyse die Möglichkeit, die Werte zum Zeitpunkt der Störung zu erfassen und die fehlerhaften Operanden im Programm im Objekt "Kriterienanalyse-Anzeige" auszugeben.
3. Programmcode in der PLC-Code-Anzeige anzeigen
   Sobald die Verbindung zur Steuerung aufgebaut wurde, zeigt die PLC-Code-Anzeige je nach Projektierung den aktuellen Status des Netzwerks an:
   - In Runtime wird das KOP/FUP Netzwerk und der Programmcode angezeigt. In der Symboltabelle werden die Symbolnamen, absolute Adressen der Symbole und zugehörige Kommentare dargestellt.
   - In Runtime wird die GRAPH-Ablaufkette dargestellt. In der Detailansicht werden die Transitionen der Schritte angezeigt.

Sie haben die Möglichkeit, eine Schaltfläche zu projektieren, die im Fall eines Fehlers Ihnen den Einsprung in die projektierte PLC-Code-Anzeige ermöglicht. Legen Sie dafür eine Schaltfläche fest und projektieren Sie daran die Systemfunktion "ZeigePLCCodeAnzeigeVonMeldung".

Übertragung von Sprachen von CPU auf HMI-Bediengerät

Im TIA Portal V14 werden die Texte für die ProDiag-Funktionalität, z.B. Symbolnamen und Kommentare in einer PLC-Code-Anzeige sowie die Texte der Überwachungsmeldungen, aus dem Anwenderprogramm der Steuerung auf das Bediengerät automatisch geladen. Achten Sie bei den mehrsprachigen Projekten darauf, dass die Sprachen, die in der Steuerung zur Verfügung stehen und die Runtime-Sprachen des Bediengeräts übereinstimmen. Nur wenn die Sprachen übereinstimmen, können die Meldetexte auf dem Bediengerät in den gewünschten Sprachen angezeigt werden.

Weitere Informationen zu mehrsprachigen Projekten finden Sie unter "Geräte und Netzte konfigurieren > Zusatzinformationen zu Konfigurationen > Arbeitsweise von S7-1500 CPUs > Einstellen des Betriebsverhaltens > CPU-Eigenschaften > Mehrsprachigkeit".

12.3.4.4 ProDiag-Übersicht projektieren

Übersicht über das Projektieren von Überwachungen

Einleitung

Im TIA Portal projektieren Sie ProDiag-Überwachungen in Ihrem Anwenderprogramm in STEP 7 und konfigurieren die Visualisierung für die HMI-Bediengeräte in WinCC.

Voraussetzungen

- Es ist eine S7-1500 Steuerung angelegt
- Es existiert mindestens eine boolesche Variable, entweder in einem Datenbaustein, Variablentabelle oder in einer Bausteinschnittstelle
- Es ist ein HMI-Bediengerät angelegt
- Ein Bild ist erstellt worden

Arbeitsschritte zur Projektierung von Überwachungen mit ProDiag

Sie projektieren die Funktionalität ProDiag im TIA Portal in folgenden Arbeitsschritten:

In STEP 7

1. ProDiag-Funktionsbaustein für die Überwachungen anlegen.
2. Einstellungen an dem ProDiag-Funktionsbaustein definieren.

3. ProDiag-Überwachungseinstellungen festlegen.
   - Sie legen die Überwachungsart fest
   - Sie stellen die Verzögerungszeit ein
   - Sie wählen die Bedingungen, bei welchen eine Überwachungsmeldung ausgelöst wird
   - Sie legen die Kategorien und Unterkategorien fest
4. Überwachung für eine boolesche Variable hinzufügen.
   Sie legen entweder eine globale oder eine lokale Überwachung für eine boolesche Variable an.
5. ProDiag-Funktionsbaustein aufrufen.
6. Projekt übersetzten und laden.

In WinCC

1. Das Objekt "ProDiag-Übersicht" in das HMI-Bild einfügen.
2. Zustandsvariable auswählen.
3. Die Anzeige und die Darstellung des Objekts in den Einstellungen konfigurieren.
4. Den Sprung in die Meldeanzeige konfigurieren.

Zusätzlich können Sie den Programmablauf im Objekt "GRAPH-Übersicht", sowie den aktuellen Programmstatus im Objekt "PLC-Code-Anzeige", anzeigen lassen.

Zuordnung von Symbolen in der ProDiag-Übersicht

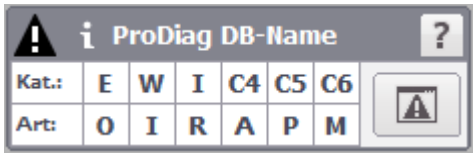

Die Zustandsvariable "State" vom PLC-Datentyp "SV_FB_State" des ProDiag-Funktionsbausteins enthält die Sammelfehlerbits für die Überwachungsarten und Kategorien, die sie mithilfe der ProDiag-Übersicht visualisieren.

Sie können zwei Arten von Sammelfehlerbits im Objekt "ProDiag-Übersicht" visualisieren:

- Sammelfehlerbit für die Überwachungen (All, O, I, R, A, P, Merr, Mtxt)
- Sammelfehlerbit für die Kategorien (C1 ... C6)

Nachfolgende Tabelle zeigt die Zuordnung der Sammelfehlerbits der Zustandsvariable "State" zu den Symbolen in der ProDiag-Übersicht.

ProDiag-Übersicht	Elemente im ProDiag-FB	Beschreibung
Symbole		
	All	Sammelfehlerbit für alle Überwachungsarten, ausgenommen von Textmeldung Wenn Wert von All = TRUE, wurde mindestens eine Überwachung ausgelöst und das Symbol wird in der ProDiag-Übersicht optisch hervorgehoben.
	SV_Types	Byte für alle Sammelfehlerbits der Überwachungsarten
	Mtxt (Message text)	Sammelfehlerbit für alle Meldungen
	-	Tooltip mit Informationen zu dieser ProDiag-Übersicht
Kategorien		
E (Error)	C1 (Category)	Fehler
W (Warning)	C2 (Category)	Warnung
	Categories	Sammelfehlerbyte für alle Kategorien.
I (Info)	C3 (Category)	Information
C4 ... C6	C4 ... C6 (Category)	Zusätzliche Kategorien
Überwachungsarten		
O	O (Operand)	Sammelfehlerbit für Operandenüberwachungen
I	I (Interlock)	Sammelfehlerbit für Verriegelungsüberwachungen
A	A (Action)	Sammelfehlerbit für Aktionsüberwachungen
R	R (Reaction)	Sammelfehlerbit für Reaktionsüberwachungen
P	P (Position)	Sammelfehlerbit für Positionsüberwachungen
M	Merr (Message error)	Sammelfehlerbit für Fehlermeldungen

ProDiag-Übersicht projektieren

Einleitung

ProDiag-Übersicht dient zur Überwachung Ihrer Maschine oder Anlage zur Laufzeit und Ermittlung von Diagnoseinformationen im Fall eines auftretenden Fehlers.

Nachdem Sie die Zustandsvariable im Objekt festgelegt haben und die Verbindung zu einem ProDiag-FB aufgebaut wurde, wird der Status der Zustandsvariablen "State" des entsprechenden PLC-Datentyps abgefragt. In Runtime werden in der ProDiag-Übersicht die Zustände der überwachten Operanden als Symbole, ähnlich wie Signalfarben einer Ampel, dargestellt.

In WinCC konfigurieren Sie die Anzeige und die Darstellung der Kategorien und Überwachungsarten, die im Objekt "ProDiag-Übersicht" dargestellt werden, unabhängig von den Überwachungseinstellungen in STEP 7.

Voraussetzungen

- Mindestens eine S7-1500 Steuerung ist angelegt.
- Mindestens eine Überwachung ist projektiert.
- Es existieren ein ProDiag-FB und ProDiag-DB.
- Eine PC-Station oder ein Bediengerät, das die Funktionalität ProDiag unterstützt, ist angelegt.
- Eine HMI-Verbindung zwischen Steuerung und Bediengerät ist angelegt.
- Ein Bild ist angelegt und das Inspektorfenster ist geöffnet.

Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie unter der Eigenschaft "Variable" die Auswahlschaltfläche.
3. Selektieren Sie die Zustandsvariable des ProDiag-FB.
   Alternativ können Sie die entsprechende Zustandsvariable aus der Detailansicht per Drag&Drop hinzufügen.

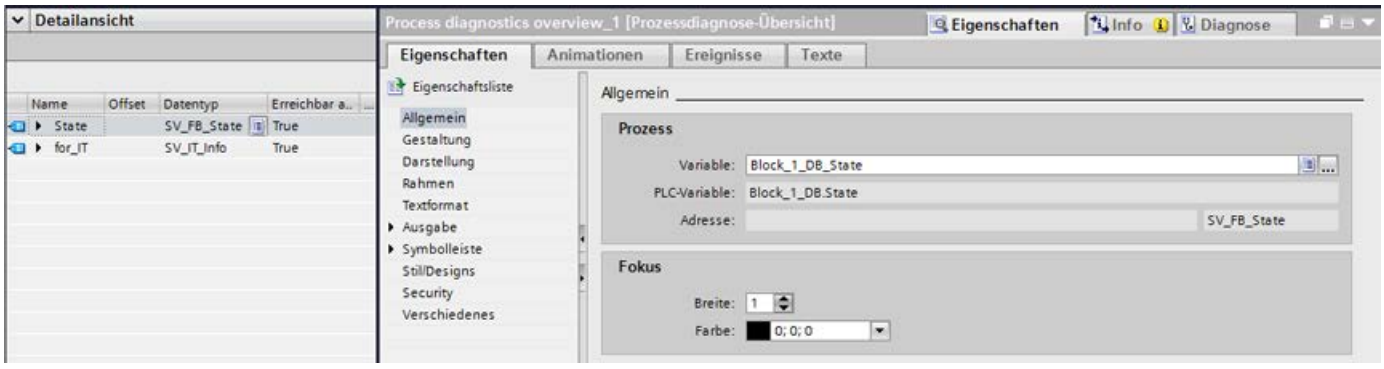

4. Legen Sie unter "Ausgabe > Kategorien" die Bezeichnungen und Farben für die Überwachungskategorien fest.
5. Legen Sie unter "Ausgabe > Überwachungsarten" die Bezeichnungen und Farben für die Überwachungsarten fest.
6. Legen Sie unter "Darstellung" fest, welche Symbole und Schaltflächen im Objekt angezeigt werden.

7. Wählen Sie unter "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.
8. Unter "Eigenschaften > Ereignisse" können Sie an das Ereignis "Schaltfläche Meldeanzeige klicken" eine Systemfunktion projektieren, um in Runtime aus der ProDiag-Übersicht in die Meldeanzeige zu springen.

Ergebnis

Die ProDiag-Übersicht ist im Bild eingefügt. In Runtime werden die aktuelle Zustände zu überwachten Ereignissen angezeigt.

Sprung in die Meldeanzeige projektieren

Einleitung

Sie haben die Möglichkeit, nach dem Auftreten eines Fehlers von der ProDiag-Übersicht direkt in ein Meldefenster oder ein Bild mit einer Meldeanzeige zu springen.

Voraussetzung

- Ein Bild ist angelegt.
- Eine Meldeanzeige oder ein Meldefenster wurde erstellt.
- Das Objekt "ProDiag-Übersicht" wurde erstellt.
- Das Inspektorfenster ist geöffnet.

Sprung in die Meldeanzeige Projektieren

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
2. Klicken Sie auf das Ereignis "Meldeanzeige Schaltfläche Klicken".
3. Wählen Sie die Systemfunktion "ZeigeMeldefenster" unter "<Funktion hinzufügen>".
4. Geben Sie den Objektnamen des Meldefensters an.
5. Alternativ rufen Sie das Bild mit der projektierten Meldeanzeige mithilfe der Funktion "AktiviereBild" auf.
   Geben Sie den Bildnamen des Bilds an, welches die projektierte Meldeanzeige enthält.

Ergebnis

Wenn Sie in Runtime die Schaltfläche "Meldeanzeige" in der projektierten ProDiag-Übersicht betätigen, wird das Bild mit der entsprechenden Meldeanzeige oder ein Meldefenster automatisch geöffnet.

12.3.4.5 GRAPH-Übersicht projektieren

Status der GRAPH-Ablaufkette anzeigen

Einleitung

Mithilfe von Objekt "GRAPH-Übersicht" visualisieren Sie den Status der GRAPH-Ablaufkette.

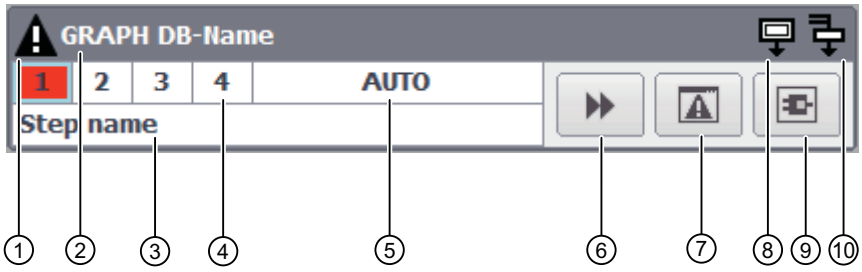

① Fehlersymbol

② Name des GRAPH-Datenbausteins

③ Name des aktuell ausgeführten Schrittes

④ Aktiver Schritt der Ablaufkette

- Falls in einem Schritt eine Störung auftritt, wird die Schrittnummer rot markiert.

⑤ Betriebsmodus

⑥ Schaltfläche "Schritt"

- Wenn die Kette sich im letzten Schritt befindet, wird die Schaltfläche "Initialschritt" angezeigt.

⑦ Schaltfläche "Meldeanzeige"

⑧ Initialschrittsymbol

⑨ Schaltfläche "PLC-Code-Anzeige".

⑩ Simultanschrittsymbol

Status der GRAPH-Ablaufkette anzeigen

Über die GRAPH-DB-Variable verbinden Sie die GRAPH-Übersicht mit der Steuerung. Nachdem die Verbindung zur Steuerung aufgebaut wurde, zeigt das Objekt die Schrittnummer der Schritte der GRAPH-Ablaufkette. Der Name des aktuell ausgeführten Schritts wird unter der Schrittkette ausgegeben. Falls es neben dem aktuell ausgeführten Schritt noch weitere Simultanschritte gibt, leuchtet das Simultanschrittsymbol in der GRAPH-Übersicht.

Je nach Anwendung können Sie zwischen den vier Betriebsmodi für den Ablauf der GRAPH-Kette wechseln. Der aktuell ausgewählte Modus wird in der GRAPH-Übersicht angezeigt.

Wenn bei der Ausführung eines Schritts ein Fehler auftritt, wird der fehlerhafte Schritt und das Fehlersymbol visuell hervorgehoben.

Im Fehlerfall haben Sie die Möglichkeit, zur Meldung in der konfigurierten Meldeanzeige zu springen oder das betroffene Programm in der PLC-Code-Anzeige anzusehen.

Einzeiliger Modus

Sie haben die Möglichkeit, die GRAPH-Übersicht in der Standard-Ansicht oder im einzeiligen Modus darzustellen. Sie können, zum Beispiel, mehrere GRAPH-Übersichten im einzeiligen Modus in einem Bild projektieren. Damit haben Sie den Überblick über den aktuellen Status von mehreren Fertigungsbereichen, deren Überwachungen in entsprechenden GRAPH-FBs zusammengefasst sind.

Anzeige von Schritten und Kriterienanalyse

Im erweiterten Modus aktivieren Sie die zusätzlichen Zeilen für die Anzeige von vorherigen und nächsten Schritt sowie den ersten fehlerhaften Operanden der Kriterienanalyse.

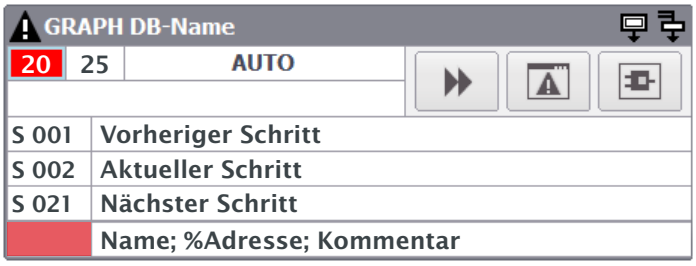

Siehe auch

GRAPH-Übersicht (Seite 396)

GRAPH-Übersicht projektieren (Seite 1951)

Betriebsmodus konfigurieren (Seite 1953)

Zusammenwirkung der Objekte zur Überwachung und Diagnose (Seite 1937)

Erstwerterfassung und Kriterienanalyse (Seite 1996)

Schrittverlauf anzeigen

Einleitung

Sie haben die Möglichkeit, in der GRAPH-Übersicht zusätzlich zum aktuellen Schritt den vorherigen und den nächsten Schritt der GRAPH-Kette anzuzeigen. Dafür werden nach dem Aufruf eines Bildes mit einer GRAPH-Übersicht die aktiven Schritte intern aufgezeichnet. Wenn in einem Schritt eine Störung auftritt, sehen Sie in der GRAPH-Übersicht, welche Aktion vor dem fehlerhaften Schritt ausgeführt wurde und welche Aktion danach kommt.

Die mehrsprachigen Schrittnamen, die in der GRAPH-Übersicht angezeigt werden, legen Sie zuvor im Anwenderprogramm in einer mehrsprachigen Textliste fest.

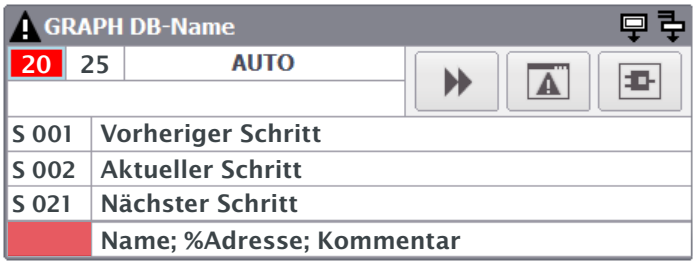

Kennzeichnung der Schritte

Die GRAPH-Ablaufkette wird von links nach rechts ausgewertet. Die Schrittnummern in der GRAPH-Übersicht werden zusätzlich gekennzeichnet, ob direkt nach oder vor diesem Schritt ein Simultan- oder ein Alternativzweig vorhanden ist. Damit sehen Sie gleich auf einen Blick folgende Informationen zum Ablauf des Programms:

- Pluszeichen
  - beim vorherigen Schritt: vor dem aktiven Schritt ist ein Simulatzweig vorhanden. Es waren mehrere Schritte gleichzeitig aktiv.
  - beim nächsten Schritt: nach dem aktiven Schritt ist ein Simultanzweig vorhanden. Damit wird mehr als ein Schritt aktiviert.
- Fragezeichen
  - beim vorherigen Schritt: vor dem aktiven Schritt konnte ein anderer Schritt als der erste (linke) Schritt der Kette aktiv gewesen sein.
  - beim nächsten Schritt: nach dem aktiven Schritt ist ein Alternativzweig vorhanden. Damit kann auch ein anderer Schritt als der erste (linke) Schritt der Kette aktiviert werden.
- Kein Zeichen: der vorheriger oder der nächster Schritt ist eindeutig ermittelt.

Hinweis

Einschränkungen bei der Schrittermittlung

Beim erstmaligen Aufruf der GRAPH-Übersicht werden die vorherigen und die nächsten Schritte aus dem GRAPH-Baustein in der Reihenfolge der programmierten Kette ermittelt.

Beim erstmaligen Aufruf der GRAPH-Übersicht und nach einem Bildwechsel existiert keine Aufzeichnung der aktiven Schritte und eine eindeutige Ermittlung des vorherigen Schrittes ist damit nicht möglich.

GRAPH-Übersicht aktualisiert die Anzeige in regelmäßigen Zeitabschnitten. Wenn die Kette schnell durchläuft oder Sie manuell häufig zwischen den Schritten schalten, kann es vorkommen, dass ein anderer Schritt der Kette ermittelt wird, als der Schritt, der aktiv war.

Vorherigen und nächsten Schritt anzeigen

Um in der GRAPH-Übersicht den vorherigen und den nachfolgenden Schritt anzuzeigen, aktivieren Sie unter "Eigenschaften > Gestaltung > Optionen" die Option "Vorheriger und nächster Schritt".

Ergebnis der Kriterienanalyse anzeigen

Sie haben die Möglichkeit, in der GRAPH-Übersicht bei einem anstehenden Fehler das Ergebnis der Kriterienanalyse anzuzeigen. Dabei wird in der letzten Zeile der GRAPH-Übersicht der Symbolname, die absolute Adresse, der Wert des ersten fehlerhaften Operanden und der Kommentar angezeigt.

Um den ersten fehlerhaften Operanden der Kriterienanalyse anzuzeigen, aktivieren Sie unter "Eigenschaften > Gestaltung > Optionen" die Option "Kriterienanalyse".

Hinweis

Für die Kriterienanalyse werden in folgenden Fällen keine Informationen in der GRAPH-Übersicht ausgeben:

- Die Version des GRAPH-Bausteins ist kleiner als 4.0.
- Die Erstwerterfassung wurde in den Bausteineigenschaften nicht aktiviert oder die Erstwerte wurden nicht erfasst.
- Im aktuellen Schritt steht kein Fehler an.

GRAPH-Übersicht projektieren

Einleitung

Mit der GRAPH-Übersicht betrachten Sie den aktuellen Programmstatus für ausgeführte Schritte einer GRAPH-Ablaufkette.

Voraussetzung

- Eine PLC mit einem GRAPH-Instanzdatenbaustein ist angelegt.
- GRAPH-Instanzdatenbaustein enthält mindestens eine Variable, die in HMI sichtbar und aus HMI erreichbar ist.

  Hinweis

  Die Prozessvariable, die Sie für die GRAPH-Übersicht verwenden, muss aus HMI sichtbar und aus HMI erreichbar sein.

  Um die Variablen des GRAPH-Datenbausteins für HMI sichtbar und erreichbar zu kennzeichnen, öffnen Sie den GRAPH-Funktionsbaustein, selektieren Sie den Baustein im Arbeitsbereich und wählen Sie in der Menüleiste "Bearbeiten > Interne Parameter sichtbar/erreichbar aus HMI". Übersetzten Sie anschließend die Programmbausteine.

- Ein Bediengerät ist angelegt.
- Ein Bild ist angelegt.
- Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie per Drag&Drop die GRAPH-Übersicht aus dem Werkzeugfenster in das projektierte Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie die Auswahlschaltfläche unter "Prozess > Variable".
   Ein Dialog "Neues Objekt hinzufügen" öffnet sich.
4. Wählen Sie im Ordner "Programmbausteine" die entsprechende PLC.
5. Selektieren Sie die entsprechende PLC-Variable des GRAPH-Instanzdatenbausteins.

   Hinweis

   Wenn zwischen dem HMI-Gerät und der gewählten PLC keine Verbindung projektiert wurde, wird automatisch eine Verbindung erstellt.

   Außerdem wird eine HMI-Variable erstellt, die mit der PLC-Variablen verbunden ist.

6. Legen Sie unter "Eigenschaften > Eigenschaften > Darstellung > Optionen" die Symbole fest, die in der GRAPH-Übersicht angezeigt werden.
   Um die GRAPH-Übersicht im Kompatibilitätsmodus ohne Schaltflächen der Symbolleiste und Betriebsmodusanzeige darzustellen, aktivieren Sie die Eigenschaft "Einzeiliger Modus".

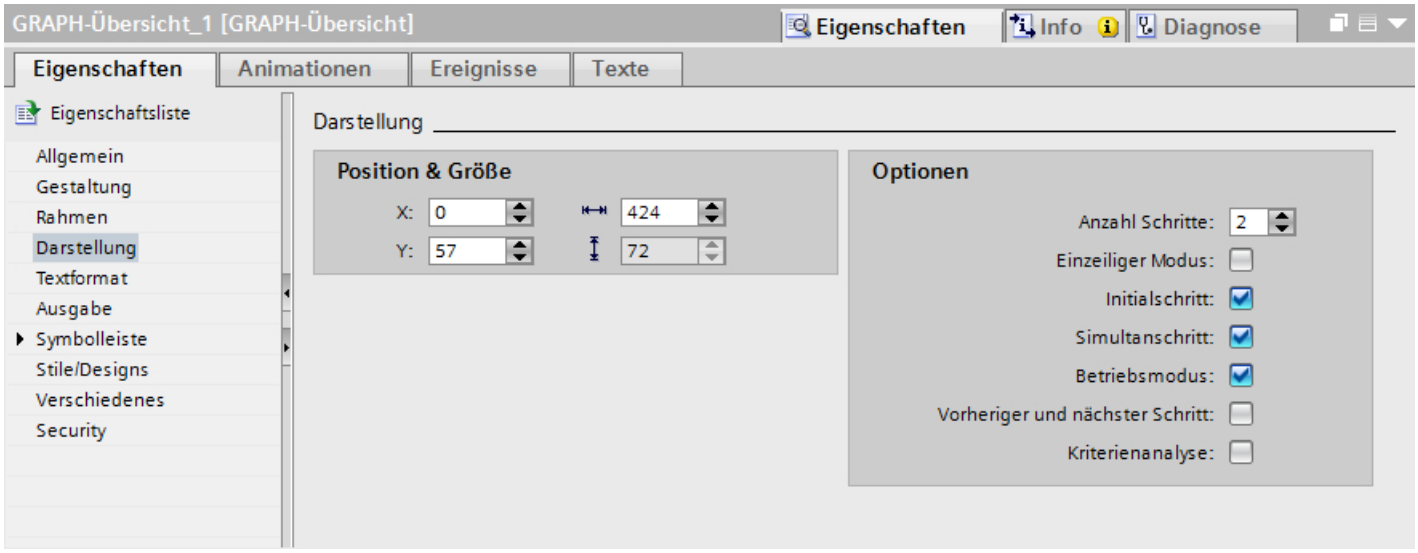

7. Legen Sie unter "Eigenschaften > Eigenschaften > Symbolleiste > Allgemein" die Schaltflächen fest, die im Objekt angezeigt werden.
8. Sie haben die Möglichkeit, die Eigenschaften im Objekt "GRAPH-Übersicht" mit Variablen zu dynamisieren und Zustände farbig zu kennzeichnen.
   Eine ausführliche Beschreibung zum Dynamisieren einer Eigenschaft finden Sie im Kapitel Dynamisieren von Bildern (Seite 210)
9. Unter "Eigenschaften > Ereignisse" können Sie den Schaltflächen in der GRAPH-Übersicht Systemfunktionen zuweisen, um in Runtime in die Meldeanzeige und in die PLC-Code-Anzeige zu springen.

Ergebnis

Die GRAPH-Übersicht ist im Bild eingefügt. In Runtime wird der aktuelle Zustand der GRAPH-Schrittkette angezeigt.

Siehe auch

Status der GRAPH-Ablaufkette anzeigen (Seite 1948)

Dynamisieren von Bildern (Seite 210)

Grundlagen zur Dynamisierung von Bildern (Seite 210)

Betriebsmodus konfigurieren

Einleitung

Für die Ausführung des GRAPH-Bausteins stehen Ihnen vier verschiedene Bedtriebsmodi zur Verfügung:

- Auto (Standardeinstellung)
  Schaltet automatisch in den nächsten Schritt, wenn die Transition erfüllt ist.
- TAP
  Schaltet in den nächsten Schritt, wenn die Transition erfüllt ist und am Parameter T_PUSH ein Flankenwechsel von "0" auf "1" vorliegt.
- TOP
  Schaltet in den nächsten Schritt, wenn die Transition erfüllt ist oder am Parameter T_PUSH ein Flankenwechsel von "0" auf "1" vorliegt.
- Manuell
  Beim Erfüllen der Transition wird der nächste Schritt nicht automatisch aktiviert. Die Schritte können manuell ausgewählt und abgewählt werden.

Hinweis

Betriebsmodus setzen Sie durch Modifizieren der Schnittstellenparameter des GRAPH-Blocks in Ihrem Steuerungsprogramm.

Hinweis

Der Name des Betriebsmodus ist sprachabhängig und kann für die gewünschten Runtime-Sprachen übersetzt werden.

In WinCC Runtime Professional haben Sie die Möglichkeit, den Betriebsmodus, der in der GRAPH-Übersicht angezeigt wird, zu umbenennen.

Beschriftungen für die Betriebsmodi konfigurieren

Ein Bild mit der projektierten GRAPH-Übersicht ist geöffnet.

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Ausgabe".
2. Konfigurieren Sie die Beschriftungen für die vier Betriebsmodi unter "Ausgabe > Beschriftungen".

12.3.4.6 PLC-Code-Anzeige projektieren

PLC-Code-Anzeige projektieren

Einleitung

Sie fügen eine PLC-Code-Anzeige in Ihrem Projekt ein, um die PLC Programmnetzwerke in den Programmiersprachen KOP, FUP und GRAPH in Runtime darzustellen.

Voraussetzung

- Mindestens eine Steuerung ist angelegt.
- Ein Bediengerät ist angelegt.
- Eine HMI-Verbindung zwischen Steuerung und Bediengerät ist angelegt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie per Drag&Drop die PLC-Code-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste".

3. Aktivieren Sie die Schaltflächen, die Sie in Runtime benötigen, z. B.: Nächstes Netzwerk, Vorheriges Netzwerk, Schrittmodus.

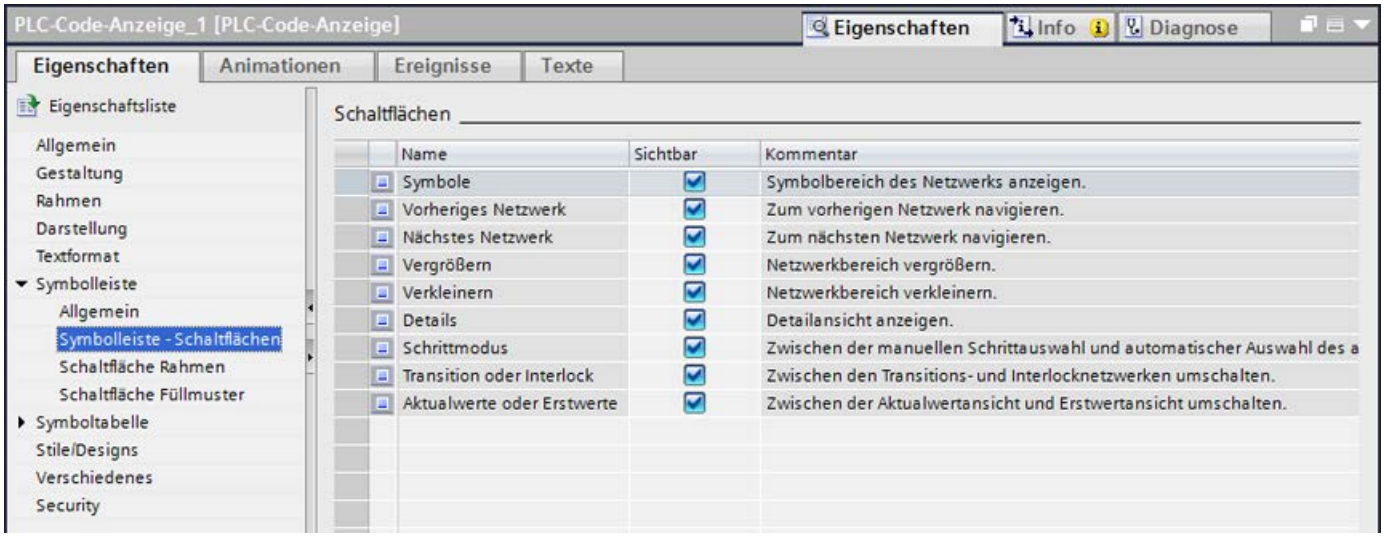

4. Konfigurieren Sie bei Bedarf die Darstellung der Schaltflächen, z. B. Hintergrundfarbe oder Füllmuster.
5. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Spalten".
6. Passen Sie bei Bedarf die Überschriften oder die Reihenfolge der Spalten an.
7. Wählen Sie "Eigenschaften > Eigenschaften > Security" eine Berechtigung für die Bedienung aus.

Ergebnis

Die PLC-Code-Anzeige ist im Bild eingefügt. In Runtime können PLC-Anwenderprogramme angezeigt werden, die in den Programmiersprachen KOP, FUP oder GRAPH programmiert worden sind.

Ansichten in der PLC-Code-Anzeige

Einleitung

In der PLC-Code-Anzeige zeigen Sie verschiedene Informationen zum Anwendeprogramm an:

- Informationsbereich
- Symboltabelle
- Detailansicht
- Transition/Interlock-Ansicht
- Erstwert/Aktualwert-Ansicht

Informationsbereich

Der Informationsbereich zeigt den KOP/FUP Programmcode Ihres Anwenderprogramms bzw. die GRAPH-Ablaufkette eines GRAPH-Funktionsbausteins. Nachdem die Verbindung zur Steuerung aufgebaut wurde, wird im Informationsbereich der Netzwerkstatus hervorgehoben über dem Programmcode dargestellt.

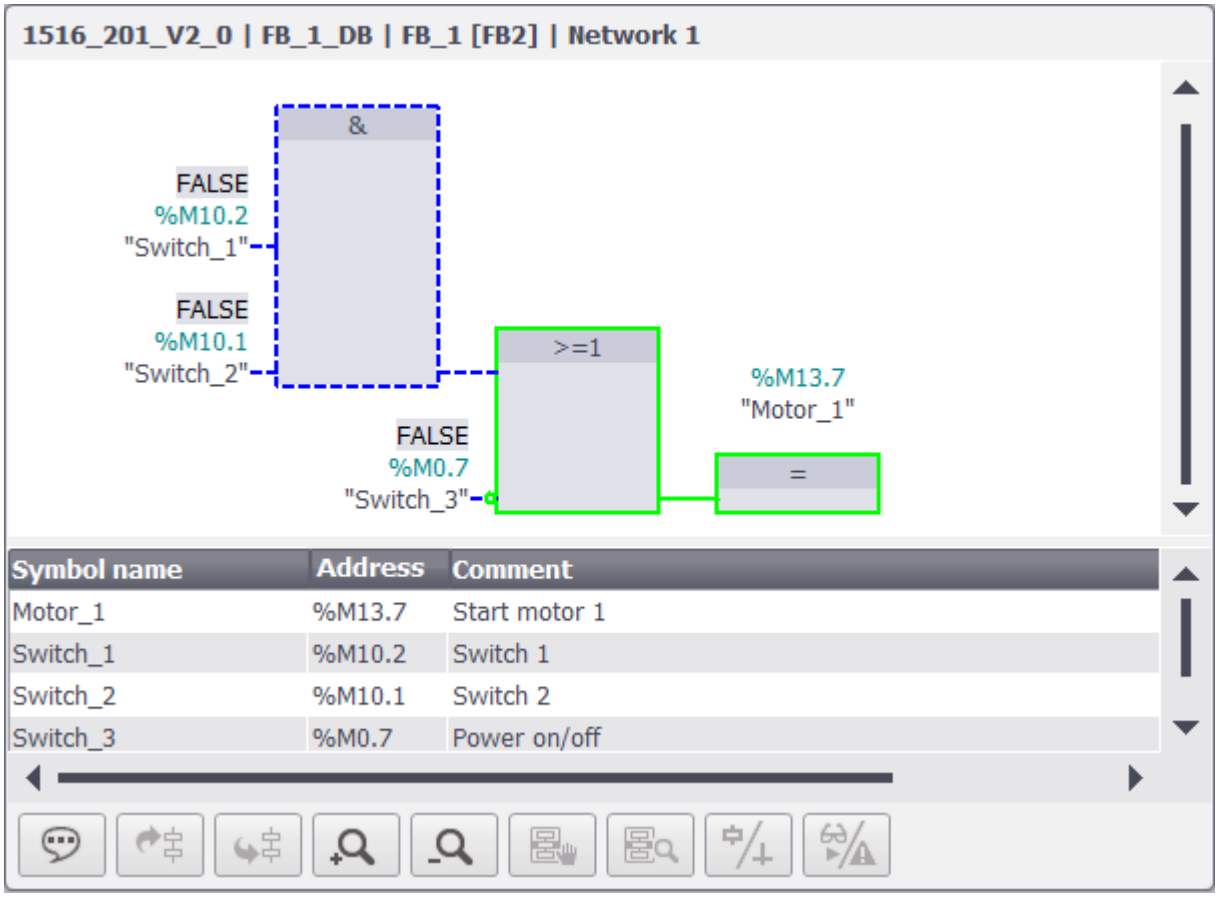

Symboltabelle

Symboltabelle zeigt die Symbole an, die im angezeigten Netzwerk verwendet sind. Die drei Spalten der Symboltabelle zeigen den Symbolnamen der Variablen, die absolute Adresse und die Kommentare an. Da die Kommentare aus dem Anwenderprogramm der Steuerung geladen werden, werden diese in der Sprache angezeigt, die in die Steuerung geladen ist.

Die Höhe und die Breite der Symboltabelle können Sie in Runtime dynamisch verändern.

Detailansicht

Für die Darstellung Ihrer GRAPH-Ablaufkette steht Ihnen zusätzlich die Detailansicht zur Verfügung.

Detailansicht zeigt ausführliche Informationen über das selektierte Netzwerk und die anliegenden Fehler. In der Detailansicht sehen Sie, z. B. die Transition des ersten aktiven Schritts der GRAPH-Ablaufkette. Sie rufen die Detailansicht in Runtime mithilfe der Schaltfläche "Detailansicht" auf.

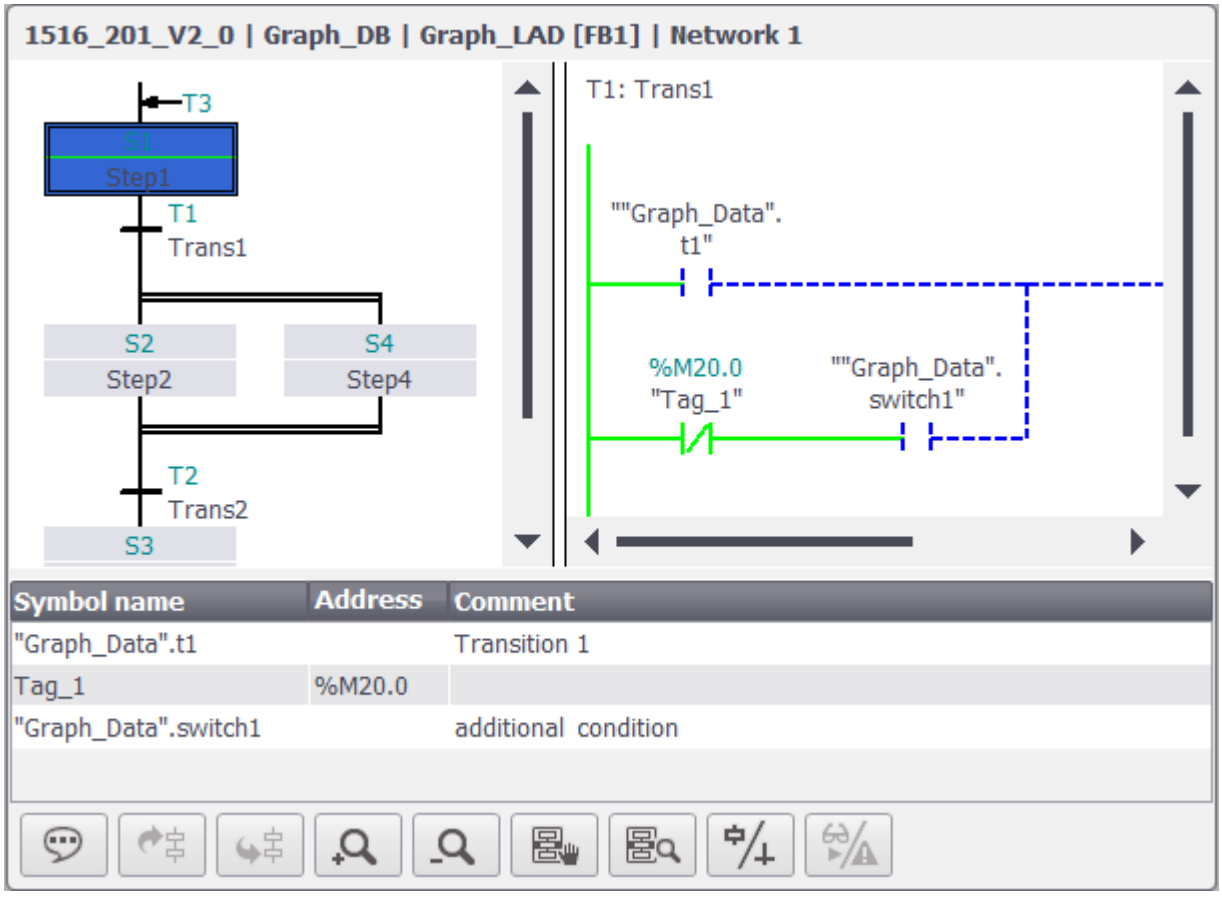

Transition/Interlock-Ansicht

Sie haben die Möglichkeit, bei der Anzeige Ihres Anwenderprogramms zwischen der Transition und Interlock-Ansichten umzuschalten. Die Transition/Interlock-Ansicht steht zur Verfügung, wenn die Detailansicht angezeigt wird.

Sie schalten zwischen den Transition/Interlock-Ansichten in Runtime mithilfe der Schaltfläche "Transition/Interlock". Beim Aktivieren der Schaltfläche wird das Interlock-Netzwerk angezeigt, beim Deaktivieren wird das Transition-Netzwerk angezeigt.

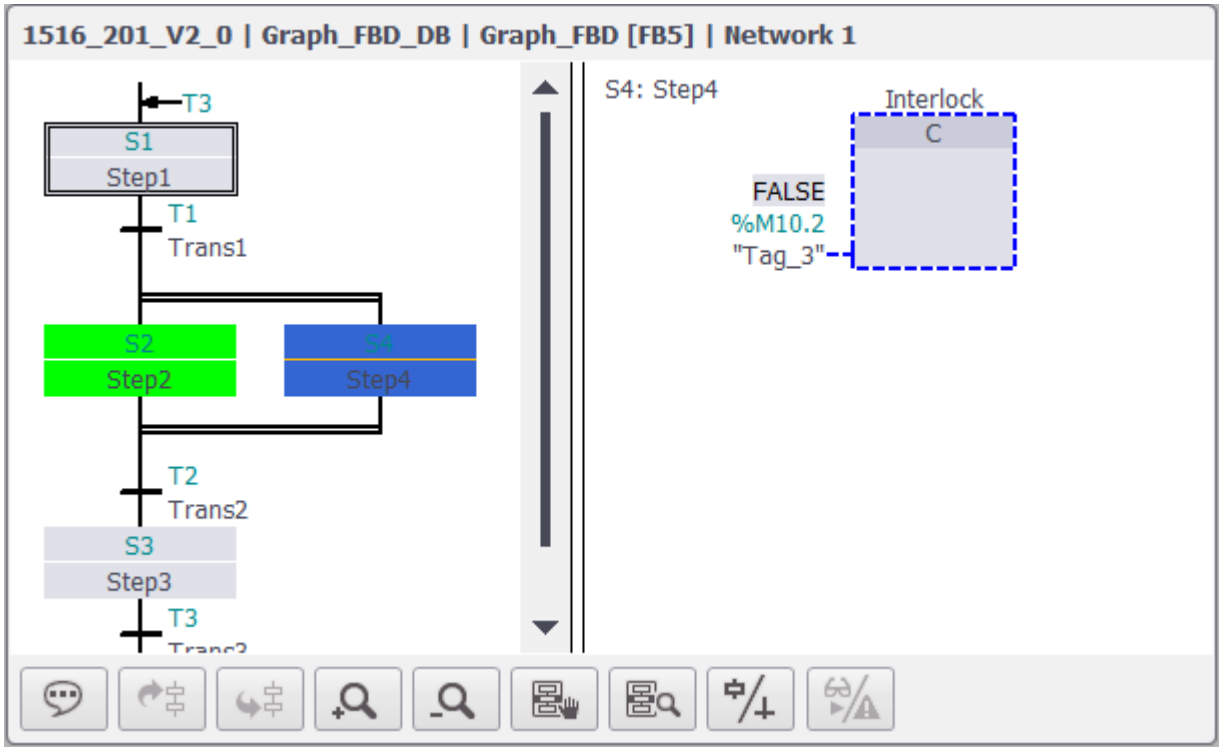

Erstwert/Aktualwert-Ansicht

Wenn sie im Anwenderprogramm für den entsprechenden Funktionsbaustein die Erstwerterfassung aktiviert haben, können Sie in der PLC-Code-Anzeige zwischen der Erstwert-Ansicht und Aktualwert-Ansicht wechseln. Die Aktualwert-Ansicht nutzt die aktuellen Werte aus der Steuerung und zeigt den aktuellen Programmstatus an. Die Erstwert-Ansicht nutzt die Werte, die zum Zeitpunkt des Fehlers aufgezeichnet wurden, und zeigt die fehlerhaften Operanden und Kriterien an. Die fehlerhaften Operanden werden in der Erstwert-Ansicht visuell hervorgehoben.

Sie schalten zwischen den Aktualwerten und den Erstwerten mithilfe der Schaltfläche "Erstwerte oder Aktualwerte". Bei der aktivierten Erstwerterfassung wird in der PLC-Code-Anzeige nach dem Einsprung standardmäßig die Erstwert-Ansicht angezeigt.

In der Erstwertansicht werden im Fehlerfall die fehlerhaften Operanden in Runtime hervogehoben.

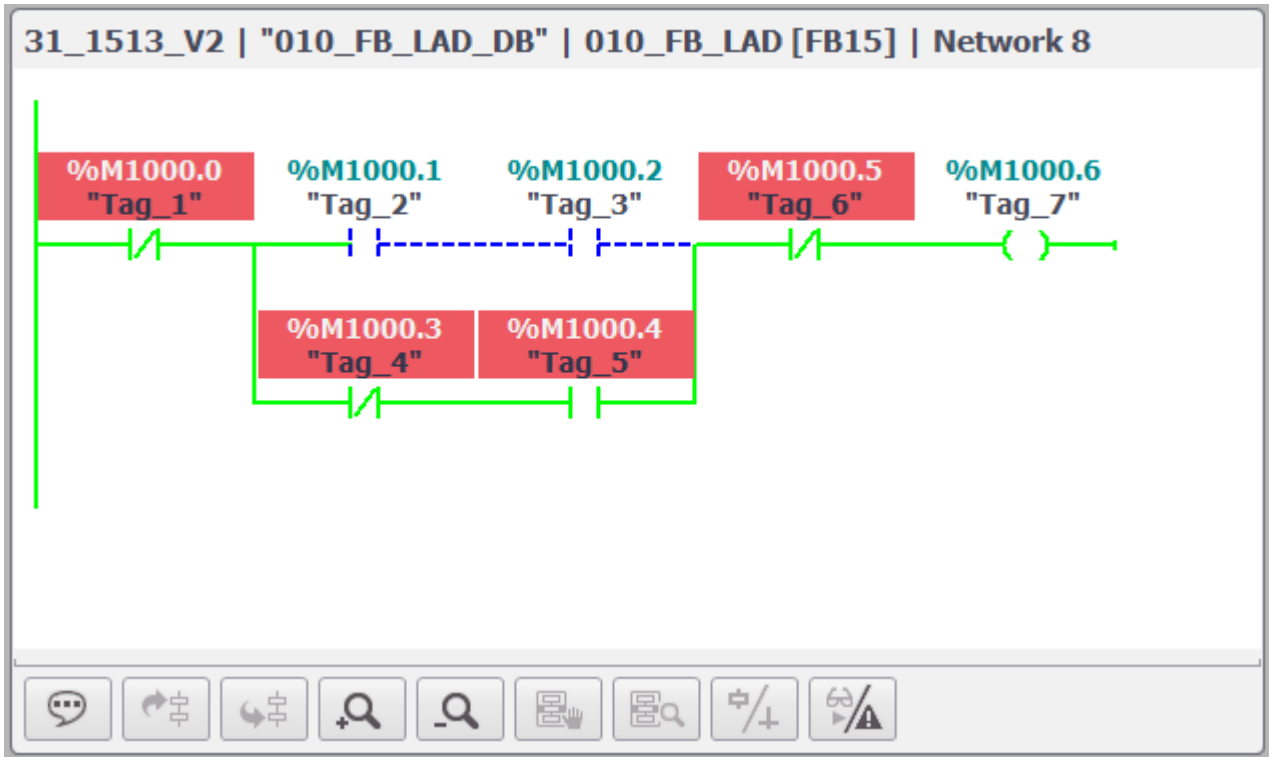

Schaltflächen der Symbolleiste

Die nachfolgende Tabelle zeigt die Schaltflächen der Symbolleiste und ihre Bedeutung.

Bedienelement	Bezeichnung	Funktion
	"Symbolbereich"	Zeigt den Symbolbereich an.
	"Vorheriges Netzwerk"	Navigiert zum vorherigen Netzwerk.
	"Nächstes Netzwerk"	Navigiert zum nächsten Netzwerk.
	"Vergrößern"	Vergrößert den Informationsbereich.
	"Verkleinern"	Verkleinert den Informationsbereich.
	"Detail"	Zeigt die Detailansicht an.
	"Schrittmodus"	Wechselt zwischen der manuellen Schrittauswahl und automatischer Auswahl des aktiven Schrittes.

Bedienelement	Bezeichnung	Funktion
	"Transition oder Interlock"	Wechselt zwischen den Darstellungen von Transitions- und Interlocknetzwerken.
	"Aktualwerte oder Erstwerte"	Wechselt zwischen Aktualwert-Ansicht und Erstwert-Ansicht.

Darstellung in der PLC-Code-Anzeige konfigurieren

Einleitung

Die PLC-Code-Anzeige bietet für die Darstellung der GRAPH-Ablaufkette auch eine geteilte Ansicht der Anzeige an. So haben Sie die Möglichkeit, das Programm und die zugehörigen Details auf einem Blick zu sehen. Die Detailansicht rufen Sie in Runtime über die Schaltfläche "Detailansicht" auf.

In WinCC können Sie das Größenverhältnis des Informationsbereichs zur Detailansicht festlegen.

Voraussetzung

- Ein Bild ist angelegt.
- Eine PLC-Code-Anzeige ist im Bild angelegt.
- Das Inspektorfenster ist geöffnet.

Breite der Detailansicht festlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Legen sie die Breite der Detailansicht im Verhältnis zum Informationsbereich mithilfe der Option "GRAPH Detailansicht Breite (%)" fest.

Ergebnis

Die PLC-Code-Anzeige wird für die Darstellung der GRAPH-Ablaufkette in Runtime in zwei Bereiche geteilt. Der linke Bereich zeigt die GRAPH-Ablaufkette und der rechte Bereich zeigt die Detailansicht mit der Transition des aktiven Schritts.

Unterstützte Anweisungen

Einleitung

Das Objekt "PLC-Code-Anzeige" unterstützt Anweisungen in den Programmiersprachen FUP und KOP.

Anweisungen

Folgende Anweisungen werden in der PLC-Code-Anzeige unterstützt:

- Bitverknüpfungen
- Vergleicher
- Zeiten
- Zähler

Bitverknüpfungen

Anweisung	KOP	FUP
Öffnerkontakt		“Tag_1“
Schließerkontakt		“Tag_1“
Ausgang setzen	“Tag_1“ S	“Tag_1“ S
Ausgang rücksetzen	“Tag_1“ R	“Tag_1“ R
Zuweisung	“Tag_1“	“Tag_1“ =
NOT: VKE invertieren	NOT	“Tag_1“ -
Negierter Kontakt	“Tag_1“	“Tag_1“ /=

Anweisung	KOP	FUP
SR: Flipflop setzen/rücksetzen	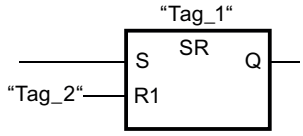	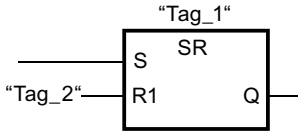
RS: Flipflop rücksetzen/setzen	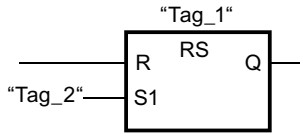	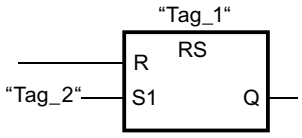
UND	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	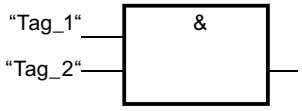
ODER	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	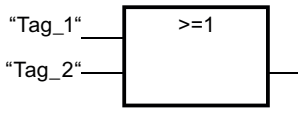
EXKLUSIV ODER	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	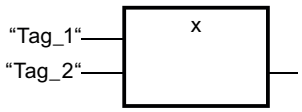

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Vergleicher

Anweisung	KOP	FUP
CMP ==: Gleich	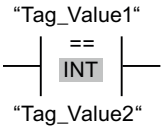	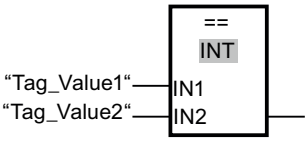
CMP <>: Ungleich	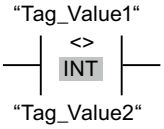	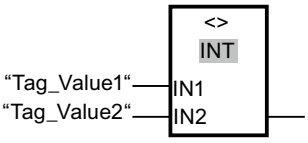
CMP >=: Größer gleich	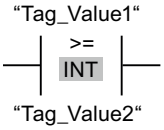	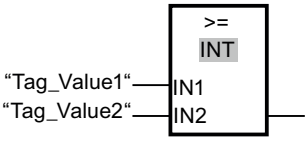

Anweisung	KOP	FUP
CMP <=: Kleiner gleich	“Tag_Value1“ <= INT “Tag_Value2“	<= INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP >: Größer	“Tag_Value1“ > INT “Tag_Value2“	> INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP <: Kleiner	“Tag_Value1“ < INT “Tag_Value2“	< INT “Tag_Value1“ IN1 “Tag_Value2“ IN2

Hinweis

PLC-Code-Anzeige unterstützt bei der Anzeige der Vergleicher folgende Datentypen: Binärzahlen, Ganzzahlen, Gleitpunktzahlen und Zeiten mit Ausnahme von S5TIME.

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Zeiten

Anweisung	KOP	FUP
TP: Impuls erzeugen	“TP_DB“ TP TIME “Tag_Start“ IN Q “Tag_Status“ “Tag_PresetTime“ PT ET “Tag_ElapsedTime“	“TP_DB“ TP TIME IN Q “Tag_Status“ PT ET “Tag_ElapsedTime“
TON: Einschaltverzögerung erzeugen	“TON_DB“ TON TIME “Tag_Start“ IN Q “Tag_Status“ “Tag_PresetTime“ PT ET “Tag_ElapsedTime“	“TON_DB“ TON TIME IN Q “Tag_Status“ PT ET “Tag_ElapsedTime“

Anweisung	KOP	FUP
TOF: Ausschaltverzögerung erzeugen	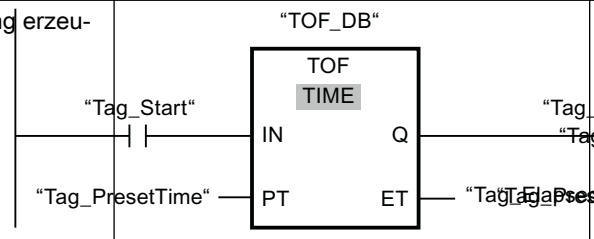	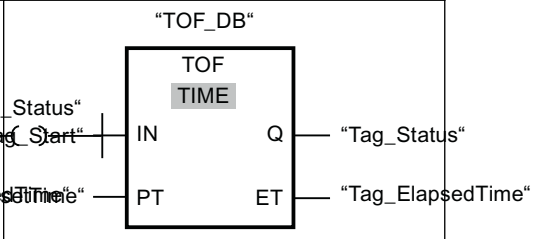
TONR: Zeit akkumulieren	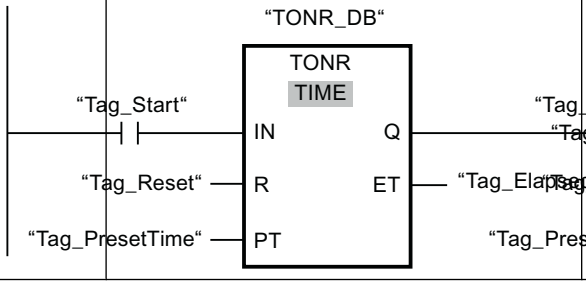	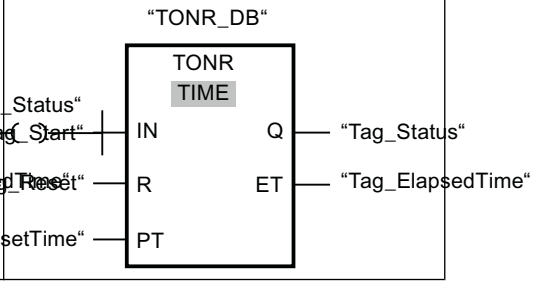

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Zähler*

* Zähler unterstützen die Ganzzahlen-Datentypen

Anweisung	KOP	FUP
CTU: Vorwärts zählen	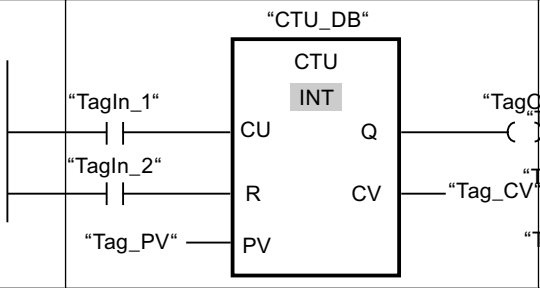	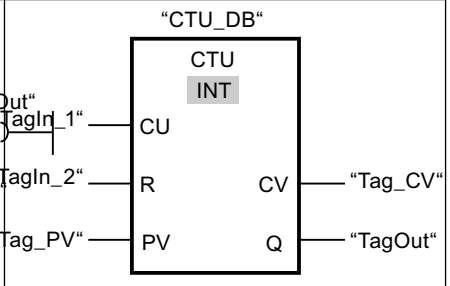
CTD: Rückwärts zählen	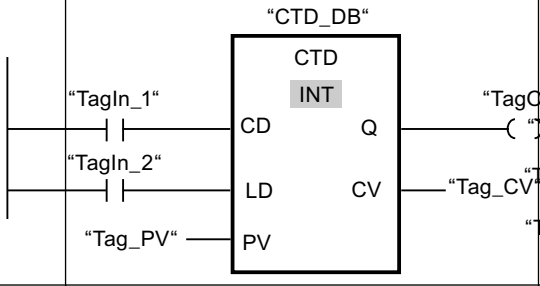	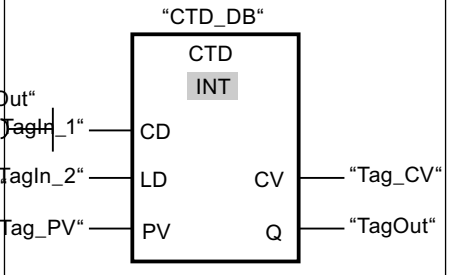
CTUD: Vorwärts und rückwärts zählen	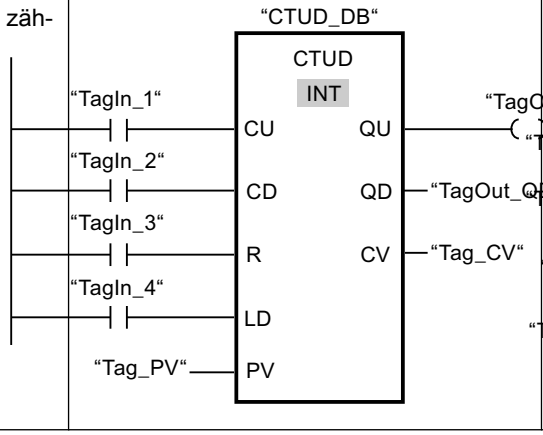	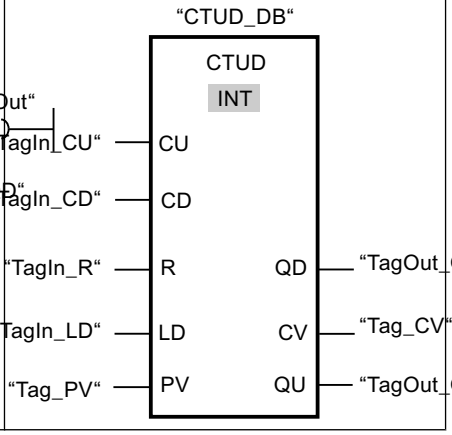

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Unterstützte Datentypen

Unterstützte Datentypen

Die folgende Tabelle zeigt die im Objekt "PLC-Code-Anzeige" unterstützten Datentypen und ihre Darstellungsformate:

Datentyp	Länge	Format
Binärzahlen		
BOOL	1 Bit	Bool

Datentyp	Länge	Format
BYTE	8 Bit	Hex
WORD	16 Bit	Hex
DWORD	32 Bit	Hex
LWORD	64 Bit	Hex
Ganzzahlen		
SINT	8 Bit	Dec
USINT	8 Bit	Dec
INT	16 Bit	Dec
UINT	16 Bit	Dec
DINT	32 Bit	Dec
UDINT	32 Bit	Dec
LINT	64 Bit	Dec
ULINT	64 Bit	Dec
Gleitpunktzahlen		
REAL	32 Bit	Floating point number
LREAL	64 Bit	Floating point number
Zeiten		
TIME	32 Bit	Time
LTIME	64 Bit	LTime
Datum und Uhrzeit		
DTL**	12 Byte	Date and Time
* Nur bei den booleschen Operanden ** Komplette DTL-Strukturen werden nicht unterstützt. Nur einzelne Elemente von DTL-Strukturen werden unterstützt.		

Hinweis

Weitere Informationen zu den PLC-Datentypen finden Sie unter "PLC programmieren > Datentypen".

Hinweis

Die 64-Bit PLC-Datentypen LINT, ULINT und LWORD werden im HMI-Kanal auf den HMI-Datentyp LREAL abgebildet. Dabei ist mit einem Genauigkeitsverlust zu rechnen, wenn der Wert die Größe 2^50 überschreitet.

Einschränkungen für die PLC-Code-Anzeige

Operanden und UDTs

Operanden, die im "#Temp"- oder "#InOut"-Bereich deklariert sind, werden von der PLC-Code-Anzeige generell nicht unterstützt. Dies gilt sowohl für elementare Datentypen als auch für Datentypen, die in UDTs enthalten sind. Datentypen eines UDTs können im "#In"- und "#Out"-Bereich deklariert und in der PLC-Code-Anzeige dargestellt werden. Für die Datentypen des UDTs gelten dabei die gleichen Beschränkungen wie für elementare Datentypen.

Einschränkung zu den Datentypen

Die Verwendung der Datentypen STRING, WSTRING, CHAR, WCHAR, S5TIME führt dazu, dass das Netzwerk in der PLC-Code-Anzeige nicht angezeigt werden kann.

Verwendung von 64-Bit Datentypen

Bei der Verwendung von 64-Bit Datentypen kann es zu einem geringen Genauigkeitsverlust kommen, da diese Datentypen im HMI-Kanal auf den Datentyp Double abgebildet werden. Somit kann es vorkommen, dass ganzzahlige Datentypen mit Nachkommastellen angezeigt werden.

Adressierung von Variablen

Beachten Sie bei der Adressierung der Variablen folgende Einschränkungen:

- Das Objekt "PLC-Code-Anzeige" unterstützt nur symbolische Adressierung von Variablen. Wenn der Operand nicht symbolisch adressiert wird, kann das Netzwerk mit dem Operanden nicht angezeigt werden und es wird eine Fehlermeldung ausgegeben.
- Die Verwendung von Array-Elementen mit einer Variable als Index wird nicht unterstützt, z.B. #myArray[MyTag]. Verwenden Sie bei der Adressierung der Array-Elemente numerische Werte, z.B. #myArray[6].
- Mit Slice-Zugriffen adressieren Sie gezielt Bereiche innerhalb deklarierter Variablen. PLC-Code-Anzeige unterstützt bei den Slice-Zugriffen nur die Zugriffsbreite "Bit" auf die booleschen Variablen.

Sprung in die PLC-Code-Anzeige

Für den Sprung von einer Überwachungsmeldung in die PLC-Code-Anzeige muss bei der Verwendung von unterstützten lokalen Operanden in einem Funktionsbaustein der Instanzname den folgenden Namenskonventionen entsprechen: <FB-Name>_DB.

Der Sprung in eine Funktion oder einen Organisationsbaustein ist nur dann möglich, wenn ausschließlich globale Operanden verwendet werden.

12.3.4.7 PLC-Code-Anzeige projektieren

PLC-Code-Anzeige projektieren

Einleitung

Sie fügen eine PLC-Code-Anzeige in Ihrem Projekt ein, um die PLC Programmnetzwerke in den Programmiersprachen KOP, FUP und GRAPH in Runtime darzustellen.

Voraussetzung

- Mindestens eine Steuerung ist angelegt.
- Eine PC Station ist angelegt.
- Eine HMI-Verbindung zwischen Steuerung und Bediengerät ist angelegt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie per Drag&Drop die PLC-Code-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
3. Unter "Darstellung > Symboltabelle Linien" passen Sie die Anzahl der Zeilen an, die in der Symboltabelle angezeigt werden.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste".

5. Weisen Sie den Schaltflächen, die Sie in Runtime benötigen, z.B. Schrittmodus, einen Hotkey zu.

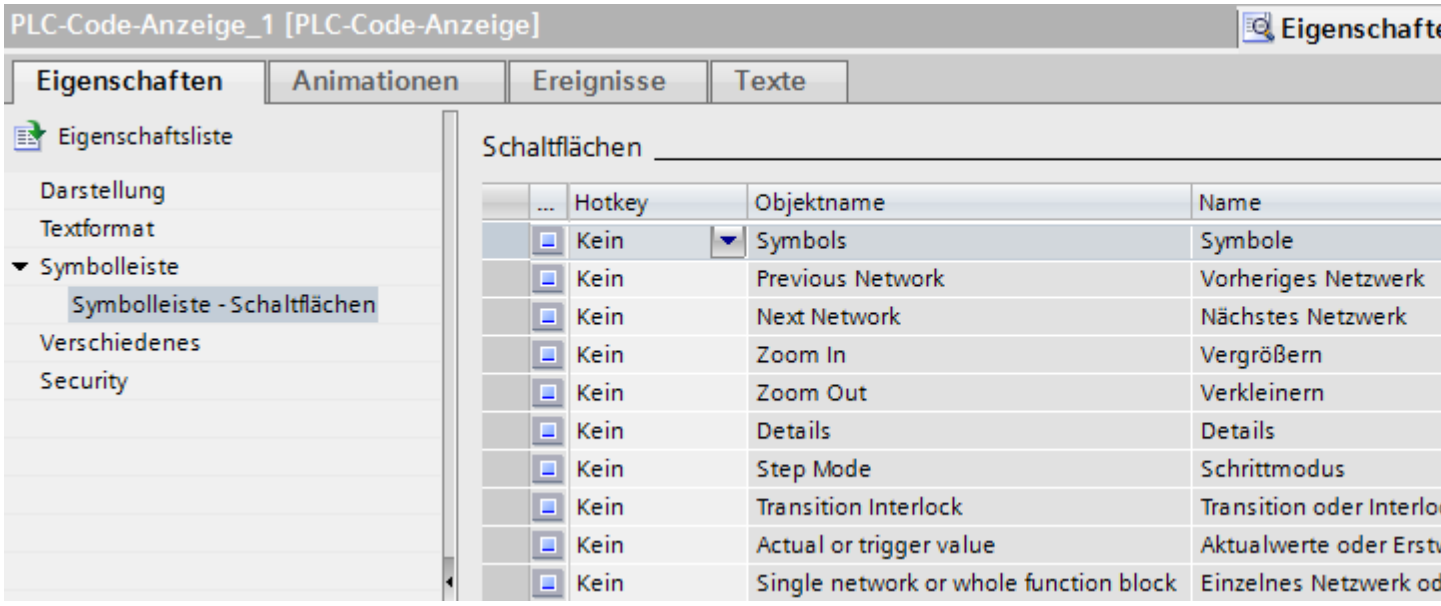

6. Um Fehler beim Aufbau der Anzeige eines Netzwerkes in Runtime anzuzeigen oder zu behandeln, projektieren Sie unter Eigenschaften > Ereignisse > Fehler" ein entsprechendes Skript.

Ergebnis

Die PLC-Code-Anzeige ist im Bild eingefügt. In Runtime können PLC-Anwenderprogramme angezeigt werden, die in den Programmiersprachen KOP, FUP oder GRAPH programmiert worden sind.

Ansichten in der PLC-Code-Anzeige

Einleitung

In der PLC-Code-Anzeige zeigen Sie verschiedene Informationen zum Anwenderprogramm an:

- Informationsbereich
- Symboltabelle
- Detailansicht
- Transition/Interlock-Ansicht
- Erstwert/Aktualwert-Ansicht

Informationsbereich

Der Informationsbereich zeigt den KOP/FUP Programmcode Ihres Anwenderprogramms, bzw. die GRAPH-Ablaufkette eines GRAPH-Datenbausteins. Nachdem die Verbindung zur Steuerung aufgebaut wurde, wird im Informationsbereich der Netzwerkstatus hervorgehoben über dem Programmcode dargestellt.

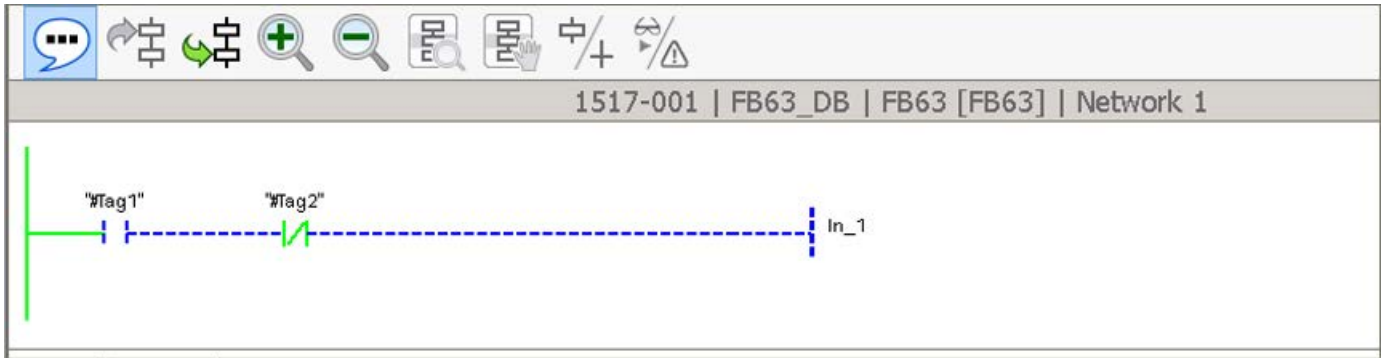

Name	Address	Comment
#Tag1		Tag 1
#Tag2		Tag 2
In_1		Input pin 1

Symboltabelle

Die Symboltabelle zeigt die Symbole an, die im angezeigten Netzwerk verwendet sind. Die drei Spalten der Symboltabelle zeigen den Symbolnamen der Variablen, die absolute Adresse und die Kommentare an. Da die Kommentare aus dem Anwenderprogramm der Steuerung geladen werden, werden diese in der Sprache angezeigt, die in die Steuerung geladen ist.

Die Höhe und die Breite der Symboltabelle können Sie in Runtime dynamisch verändern.

Detailansicht

Die Detailansicht zeigt ausführliche Informationen über das selektierte Netzwerk und die anliegenden Fehler. In der Detailansicht sehen Sie z. B. die Transition des ersten aktiven Schritts der GRAPH-Ablaufkette. Sie rufen die Detailansicht in Runtime mithilfe der Schaltfläche "Detailansicht" auf.

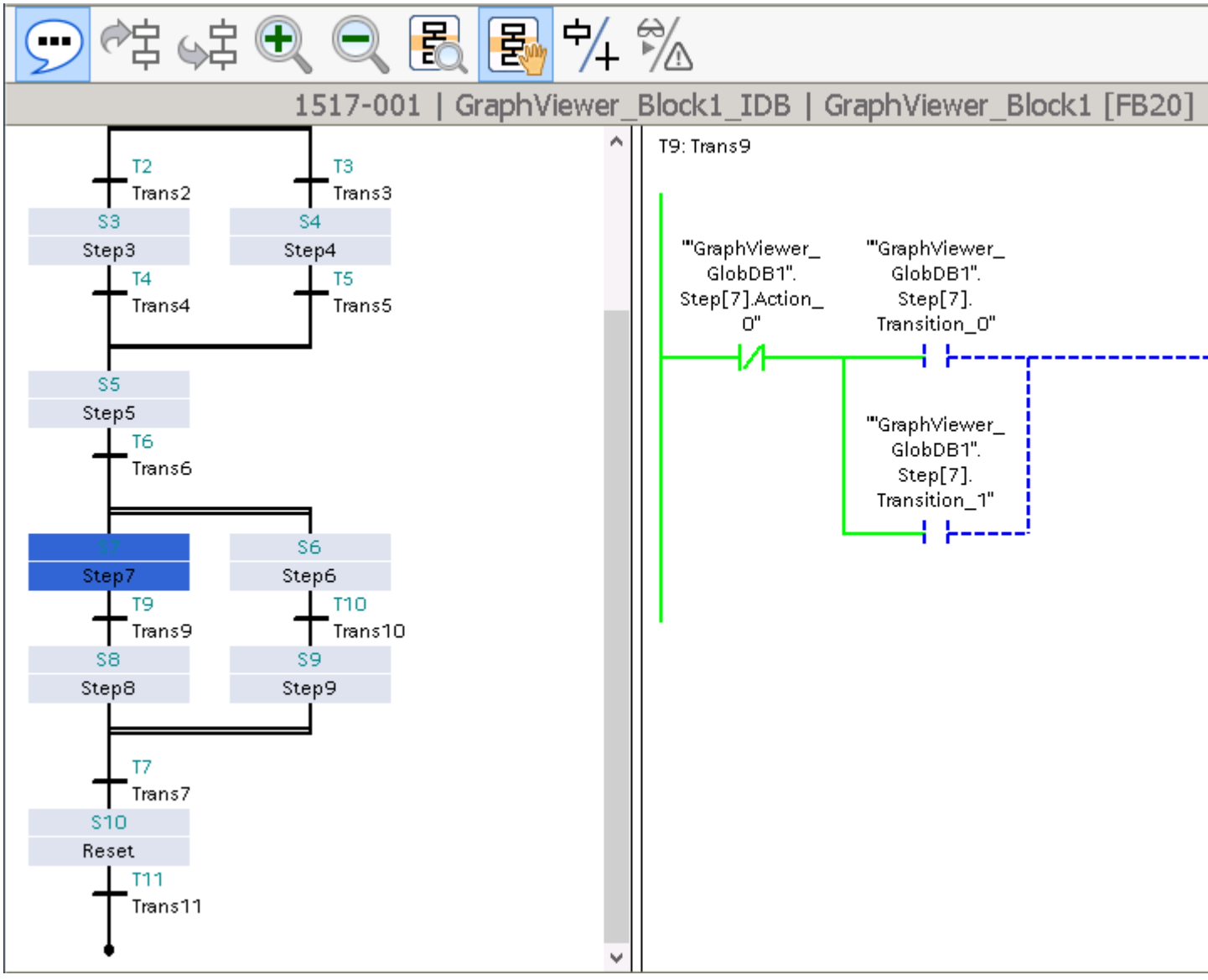

Transition/Interlock-Ansicht

Sie haben die Möglichkeit, bei der Anzeige Ihres KOP/FUP-Anwenderprogramms zwischen der Transition und Interlock-Ansichten umzuschalten. Die Transition/Interlock-Ansicht steht zur Verfügung, wenn die Detailansicht angezeigt wird.

Sie schalten zwischen den Transition/Interlock-Ansichten in Runtime mithilfe der Schaltfläche "Transition/Interlock". Beim Aktivieren der Schaltfläche wird das Interlock-Netzwerk angezeigt, beim Deaktivieren wird das Transition-Netzwerk angezeigt.

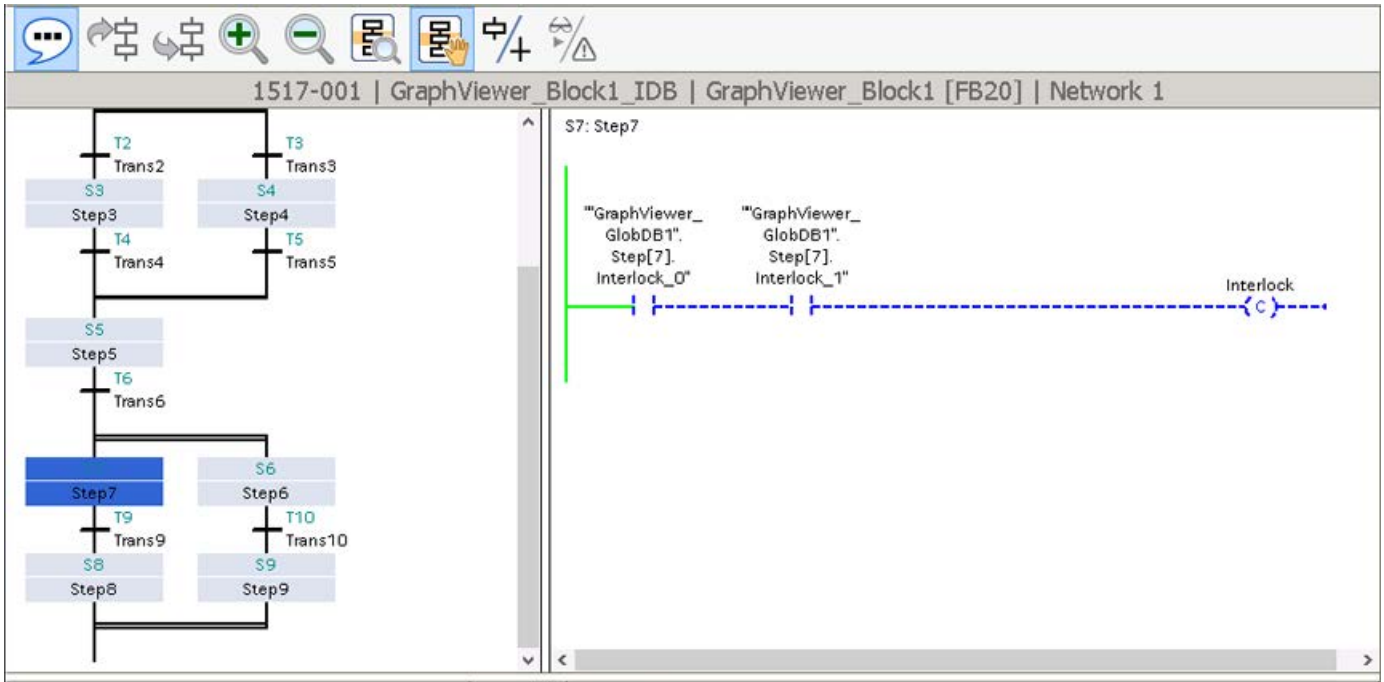

Name	Address	Comment
"GraphViewer_GlobDB1".Step[7].Interlock_0		Interlock_0
"GraphViewer_GlobDB1".Step[7].Interlock_1		Interlock 1

Erstwert/Aktualwert-Ansicht

Wenn sie im Anwenderprogramm für den entsprechenden Funktionsbaustein die Erstwerterfassung aktiviert haben, können Sie in der PLC-Code-Anzeige zwischen der Erstwert-Ansicht und Aktualwert-Ansicht wechseln. Die Aktualwert-Ansicht nutzt die aktuellen Werte aus der Steuerung und zeigt den aktuellen Programmstatus an. Die Erstwert-Ansicht nutzt die Werte, die zum Zeitpunkt des Fehlers aufgezeichnet wurden, und zeigt die fehlerhaften Operanden und Kriterien an. Die fehlerhaften Operanden werden in der Erstwert-Ansicht visuell hervorgehoben.

Sie schalten zwischen den Aktualwerten und den Erstwerten mithilfe der Schaltfläche "Erstwerte oder Aktualwerte". Bei der aktivierten Erstwerterfassung wird in der PLC-Code-Anzeige nach dem Einsprung standardmäßig die Erstwert-Ansicht angezeigt.

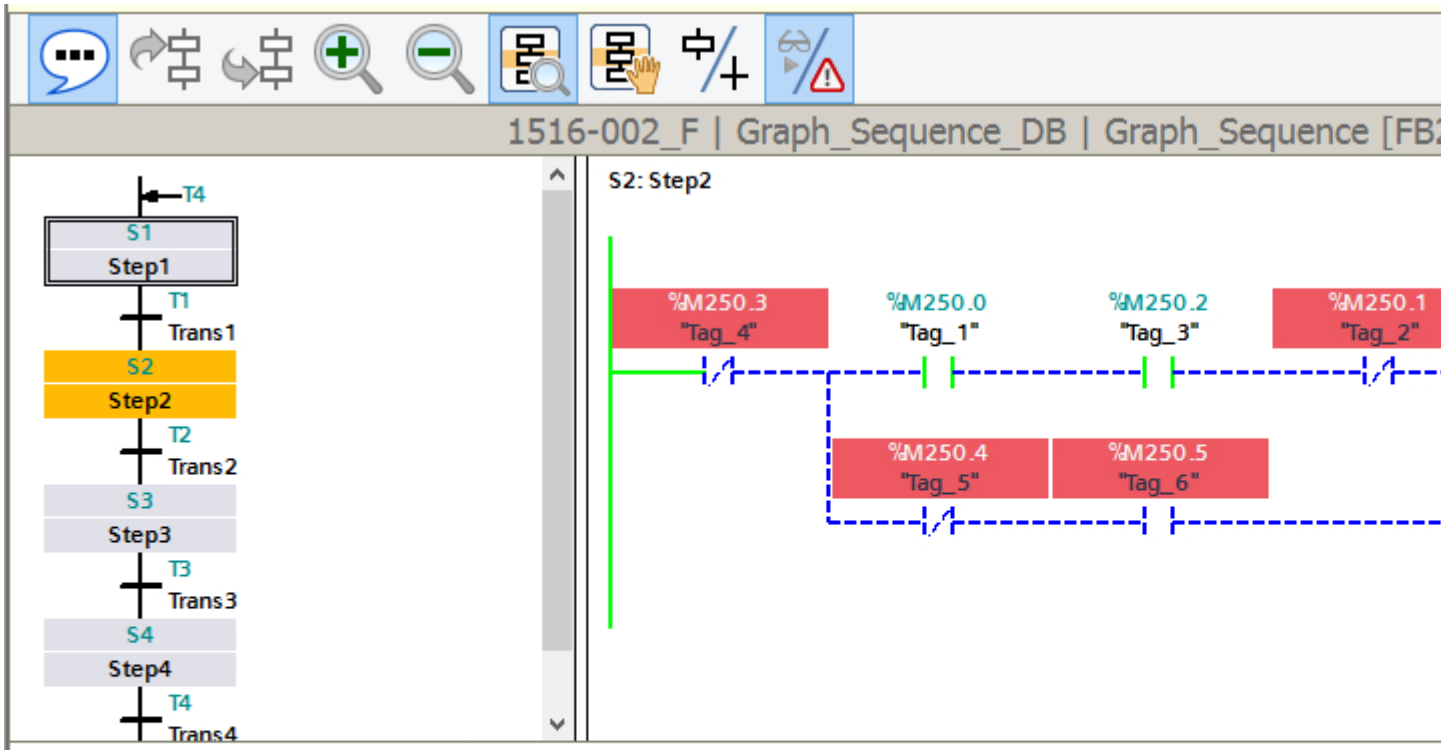

Name	Address	Comment
Tag_4	%M250.3	Interlock_2
Tag_2	%M250.1	Interlock_1
Tag_5	%M250.4	Hand mode
Tag_6	%M250.5	Hand switch

Schaltflächen der Symbolleiste

Die nachfolgende Tabelle zeigt die Schaltflächen der Symbolleiste und ihre Bedeutung.

Bedienelement	Bezeichnung	Funktion
	"Symbolbereich"	Zeigt den Symbolbereich an.
	"Vorheriges Netzwerk"	Navigiert zum vorherigen Netzwerk.
	"Nächstes Netzwerk"	Navigiert zum nächsten Netzwerk.
	"Vergrößern"	Vergrößert den Informationsbereich.

Bedienelement	Bezeichnung	Funktion
	"Verkleinern"	Verkleinert den Informationsbereich.
	"Detail"	Zeigt die Detailansicht an.
	"Schrittmodus"	Wechselt zwischen der manuellen Schrittauswahl und automatischer Auswahl des aktiven Schrittes.
	"Transtition oder Interlock"	Wechselt zwischen Transition-Ansicht und Interlock-Ansicht.
	"Aktualwerte oder Erstwerte"	Wechselt zwischen Aktualwert-Ansicht und Erstwert-Ansicht.

Unterstützte Anweisungen

Einleitung

Das Objekt "PLC-Code-Anzeige" unterstützt Anweisungen in den Programmiersprachen FUP und KOP.

Anweisungen

Folgende Anweisungen werden in der PLC-Code-Anzeige unterstützt:

- Bitverknüpfungen
- Vergleicher
- Zeiten
- Zähler

Bitverknüpfungen

Anweisung	KOP	FUP
Öffnerkontakt		“Tag_1“
Schließerkontakt		“Tag_1“

Anweisung	KOP	FUP
Ausgang setzen	“Tag_1“ (S)	“Tag_1“ S
Ausgang rücksetzen	“Tag_1“ (R)	“Tag_1“ R
Zuweisung	“Tag_1“ ( )	“Tag_1“ =
NOT: VKE invertieren	NOT	“Tag_1“ -
Negierter Kontakt	“Tag_1“ (/)	“Tag_1“ /=
SR: Flipflop setzen/rücksetzen	“Tag_1“ SR S Q “Tag_2“ R1	“Tag_1“ SR S “Tag_2“ R1 Q
RS: Flipflop rücksetzen/setzen	“Tag_1“ RS R Q “Tag_2“ S1	“Tag_1“ RS R “Tag_2“ S1 Q
UND	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	“Tag_1“ “Tag_2“ &
ODER	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	“Tag_1“ “Tag_2“ >=1
EXKLUSIV ODER	Wird durch entsprechende Verschaltung von Öffner- bzw. Schließerkontakten realisiert.	“Tag_1“ “Tag_2“ x

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Vergleicher

Anweisung	KOP	FUP
CMP ==: Gleich	“Tag_Value1“ == INT “Tag_Value2“	== INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP <>: Ungleich	“Tag_Value1“ <> INT “Tag_Value2“	<> INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP >=: Größer gleich	“Tag_Value1“ >= INT “Tag_Value2“	>= INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP <=: Kleiner gleich	“Tag_Value1“ <= INT “Tag_Value2“	<= INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP >: Größer	“Tag_Value1“ > INT “Tag_Value2“	> INT “Tag_Value1“ IN1 “Tag_Value2“ IN2
CMP <: Kleiner	“Tag_Value1“ < INT “Tag_Value2“	< INT “Tag_Value1“ IN1 “Tag_Value2“ IN2

Hinweis

PLC-Code-Anzeige unterstützt bei der Anzeige der Vergleicher folgende Datentypen: Binärzahlen, Ganzzahlen, Gleitpunktzahlen und Zeiten mit Ausnahme von S5TIME.

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Zeiten

Anweisung	KOP	FUP
TP: Impuls erzeugen	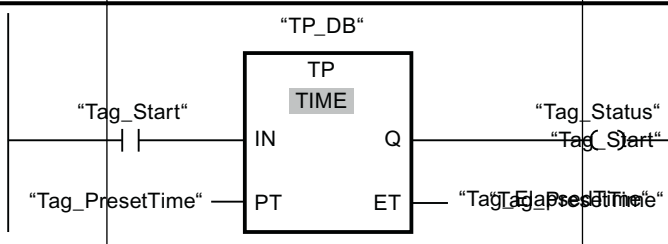	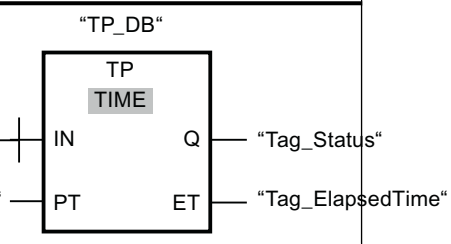
TON: Einschaltverzögerung erzeugen	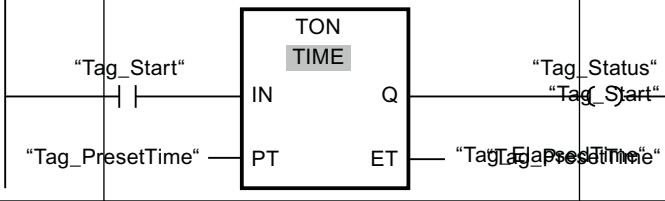	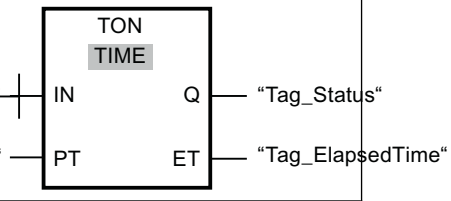
TOF: Ausschaltverzögerung erzeugen	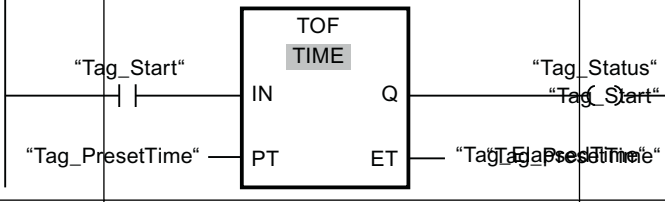	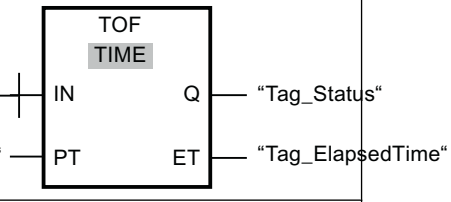
TONR: Zeit akkumulieren	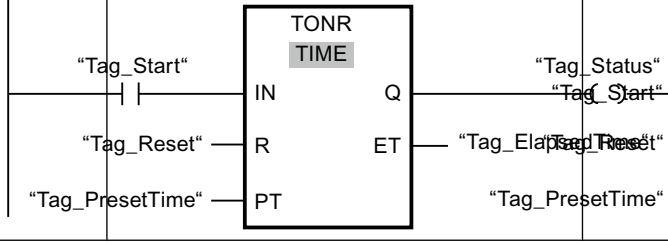	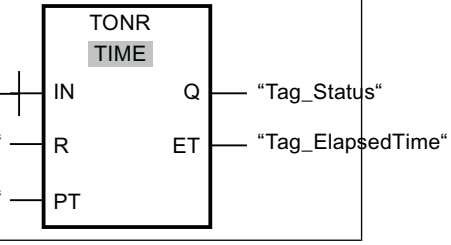

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Zähler*

* Zähler unterstützen die Ganzzahlen-Datentypen

Anweisung	KOP	FUP
CTU: Vorwärts zählen	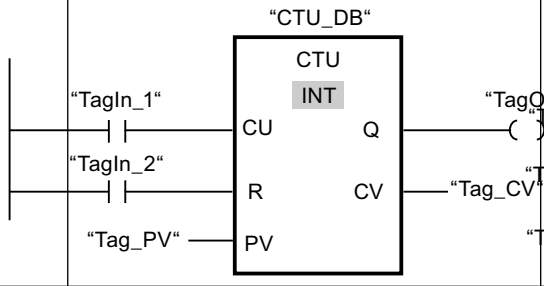	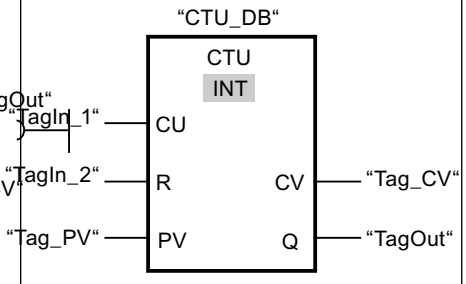
CTD: Rückwärts zählen	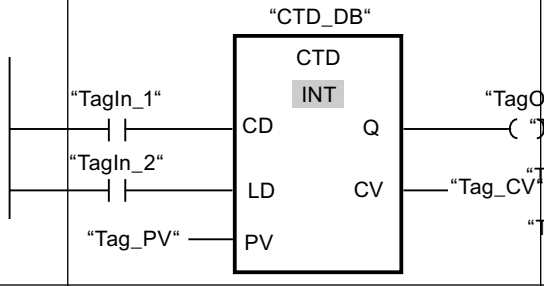	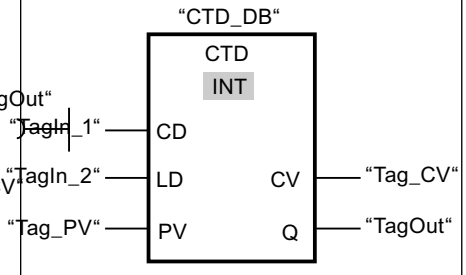
CTUD: Vorwärts und rückwärts zählen	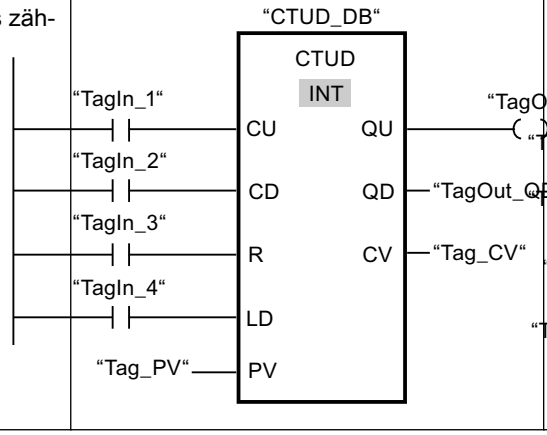	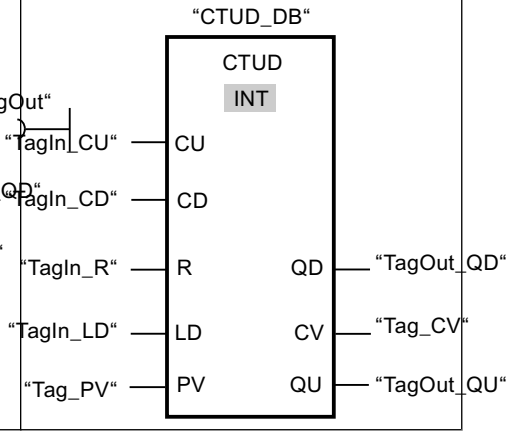

Weitere Informationen zu Anweisungen finden Sie im Kapitel "PLC programmieren > Anweisungen".

Unterstützte Datentypen

Unterstützte Datentypen

Die folgende Tabelle zeigt die im Objekt "PLC-Code-Anzeige" unterstützten Datentypen und ihre Darstellungsformate:

Datentyp	Länge	Format
Binärzahlen		
BOOL*	1 Bit	Bool

Datentyp	Länge	Format
BYTE	8 Bit	Hex
WORD	16 Bit	Hex
DWORD	32 Bit	Hex
LWORD	64 Bit	Hex
Ganzzahlen		
SINT	8 Bit	Dec
USINT	8 Bit	Dec
INT	16 Bit	Dec
UINT	16 Bit	Dec
DINT	32 Bit	Dec
UDINT	32 Bit	Dec
LINT	64 Bit	Dec
ULINT	64 Bit	Dec
Gleitpunktzahlen		
REAL	32 Bit	Floating point number
LREAL	64 Bit	Floating point number
Zeiten		
S5TIME	16 Bit	SIMATIC_Timer
TIME	32 Bit	Time
LTIME	64 Bit	LTime
Datum und Uhrzeit		
DTL**	12 Byte	Date and Time
* Nur bei den booleschen Operanden ** Komplette DTL-Strukturen werden nicht unterstützt. Nur einzelne Elemente von DTL-Strukturen werden unterstützt.		

Hinweis

Weitere Informationen zu den PLC-Datentypen finden Sie unter "PLC programmieren > Datentypen".

Hinweis

Operanden und UDTs

Operanden, die im "#Temp"- oder "#InOut"-Bereich deklariert sind, werden von der PLC-Code-Anzeige generell nicht unterstützt. Dies gilt sowohl für elementare Datentypen als auch für Datentypen, die in UDTs enthalten sind. Datentypen eines UDTs können im "#In"- und "#Out"-Bereich deklariert und in der PLC-Code-Anzeige dargestellt werden. Für die Datentypen des UDTs gelten dabei die gleichen Beschränkungen wie für elementare Datentypen.

Hinweis

Das Vorverschaltungsnetzwerk darf keine Variablen aus dem Temp- oder InOut-Bereich eines FBs entahlten.

Folgende Datentypen dürfen nicht für Variablen verwendet werden: STRING, WSTRING, CHAR, WCHAR, S5TIME

Hinweis

Die 64-Bit PLC-Datentypen LINT, ULINT und LWORD werden im HMI-Kanal auf den HMI-Datentyp LREAL abgebildet. Dabei ist mit einem Genauigkeitsverlust zu rechnen, wenn der Wert die Größe 2^50 überschreitet.

Siehe auch

Datentypen für SIMATIC S7-1500 (Seite 632)

Einschränkungen für die PLC-Code-Anzeige

Operanden und UDTs

Operanden, die im "#Temp"- oder "#InOut"-Bereich deklariert sind, werden von der PLC-Code-Anzeige generell nicht unterstützt. Dies gilt sowohl für elementare Datentypen als auch für Datentypen, die in UDTs enthalten sind. Datentypen eines UDTs können im "#In"- und "#Out"-Bereich deklariert und in der PLC-Code-Anzeige dargestellt werden. Für die Datentypen des UDTs gelten dabei die gleichen Beschränkungen wie für elementare Datentypen.

Einschränkung zu den Datentypen

Die Verwendung der Datentypen STRING, WSTRING, CHAR, WCHAR, S5TIME führt dazu, dass das Netzwerk in der PLC-Code-Anzeige nicht angezeigt werden kann.

Verwendung von 64-Bit Datentypen

Bei der Verwendung von 64-Bit Datentypen kann es zu einem geringen Genauigkeitsverlust kommen, da diese Datentypen im HMI-Kanal auf den Datentyp Double abgebildet werden. Somit kann es vorkommen, dass ganzzahlige Datentypen mit Nachkommastellen angezeigt werden.

Adressierung von Variablen

Beachten Sie bei der Adressierung der Variablen folgende Einschränkungen:

- Das Objekt "PLC-Code-Anzeige" unterstützt nur symbolische Adressierung von Variablen. Wenn der Operand nicht symbolisch adressiert wird, kann das Netzwerk mit dem Operanden nicht angezeigt werden und es wird eine Fehlermeldung ausgegeben.
- Die Verwendung von Array-Elementen mit einer Variable als Index wird nicht unterstützt, z.B. #myArray[MyTag]. Verwenden Sie bei der Adressierung der Array-Elemente numerische Werte, z.B. #myArray[6].
- Mit Slice-Zugriffen adressieren Sie gezielt Bereiche innerhalb deklarierter Variablen. PLC-Code-Anzeige unterstützt bei den Slice-Zugriffen nur die Zugriffsbreite "Bit" auf die booleschen Variablen.

Sprung in die PLC-Code-Anzeige

Für den Sprung von einer Überwachungsmeldung in die PLC-Code-Anzeige muss bei der Verwendung von unterstützten lokalen Operanden in einem Funktionsbaustein der Instanzname den folgenden Namenskonventionen entsprechen: <FB-Name>_DB.

Der Sprung in eine Funktion oder einen Organisationsbaustein ist nur dann möglich, wenn ausschließlich globale Operanden verwendet werden.

Einsprung von der Meldung konfigurieren

Einleitung

Sie haben die Möglichkeit, einen Einsprung aus der Meldeanzeige in die PLC-Code-Anzeige oder ins Projekt im TIA Portal aus Runtime zu konfigurieren. Ihnen stehen zwei Funktionen zur Verfügung:

- Mithilfe der Systemfunktion "ZeigePLCCodeAnzeigeVonMeldung (Seite 1773)" springen Sie in die PLC-Code-Anzeige.
- Mithilfe der Systemfunktion "ZeigeBlockInTIAPortalVonMeldung (Seite 1772)" springen Sie aus Runtime in die Projektierung im TIA Portal.

In diesem Beispiel konfigurieren Sie den Einsprung in die PLC-Code-Anzeige aus der entsprechenden Meldung in der Meldeannzeige.

Voraussetzung

- Das Bild "Alarm" ist geöffnet.
- Die Meldeanzeige "ProDiag_Alarm" ist angelegt.
- Die PLC-Code-Anzeige "PLC code view" ist im Bild "Program code" angelegt.

Schaltfläche für den Einsprung projektieren

1. Ziehen Sie eine Schaltfläche aus der Task Card "Werkzeuge" in das offene Bild "Alarm".
2. Vergeben Sie unter "Verschiedenes > Objekt > Name" einen Namen, z. B. "Jump to PLC Code".
3. Wechseln Sie zu "Ereignisse > Klicken".
4. Klicken Sie auf "Funktion hinzufügen".
5. Selektieren Sie aus der Auswahlliste die Funktion "ZeigePLCCodeAnzeigeVonMeldung".
6. Geben Sie unter "Meldungsbildname" den Namen des Bilds, in welchem die Meldeanzeige projektiert ist, "Alarm".
7. Geben Sie unter "Meldeanzeigenname" den Namen der Meldeanzeige an, "ProDiag_Alarm".
8. Geben Sie unter "Bildname der PLC-Code-Anzeige" "Program code" an.
9. Geben Sie unter "Name der PLC-Code-Anzeige" "PLC code view" an.

Einsprungsmöglichkeit prüfen

1. Selektieren Sie die Meldeanzeige "ProDiag-Alarm".
2. Wechseln Sie zum Register "Ereignisse".
3. Klicken Sie auf das Ereignis "Auswahl geändert".
4. Erstellen Sie ein benutzerdefiniertes C-Skript an diesem Ereignis:

```
#include "GlobalDefinitions.h"
void OnSelectedIdChanged(char* screenName, char* objectName, long
id) {
#pragma code("KOPAPI.dll")
#include "kopapi.h"
#pragma code()
CMN_ERROR errorStruct;
BOOL bResult;
bResult = IsJumpableProDiagAlarm (screenName, objectName, id, &err
orStruct);
SetPropBOOL (screenName, "Jump to PLC Code", "Enabled", bResult);
```

Ergebnis

Sie haben eine Schaltfläche zum Springen in die PLC-Code-Anzeige projektiert. Nur dann, wenn eine ProDiag-relevante Meldung in der Meldeanzeige erscheint, wird die Schaltfläche angezeigt, die den Einsprung ermöglicht.

Siehe auch

In die Projektierung springen (Seite 2420)

12.3.4.8 Darstellungsprobleme in der PLC-Code-Anzeige beheben

Einleitung

Wenn bei der Visualisierung des Programmcodes ein Fehler auftritt, wird in der PLC-Code-Anzeige eine Fehlermeldung mit der dazugehörigen Fehlernummer ausgegeben. Um den Anzeigefehler zu beheben, arbeiten Sie zunächst die vorgeschlagenen ersten Maßnahmen durch.

Wenn der Fehler weiterhin besteht, setzen Sie sich in Verbindung mit dem Customer Support.

Erste Maßnahmen zur Fehlerbehebung

Beachten Sie die nachfolgenden allgemeinen Hinweise für die Fehlerbehebung:

1. Stellen Sie sicher, dass die aktuelle Projektierung auf das Bediengerät geladen wurde.
2. Wenn Sie Programmcode in der PLC-Code-Anzeige mithilfe einer Runtime-API Funktion darstellen, überprüfen Sie die Schreibweise und die Reihenfolge der Parameter des Aufrufs.
3. Starten Sie Runtime auf Ihrem Bediengerät erneut.

Hinweise für Fehlermeldungen

Die nachfolgende Tabelle zeigt die Fehlermeldungen und die Hinweise zur Behebung einiger spezifischen Fehler bei der Anzeige des Programmcodes (sofern vorhanden):

Fehlernummer	Fehlermeldung	Hinweise
3101, 3107, 3109, 3113, 3402, 3501, 3502, 3601, 3602, 4000 - 4107, 4110 - 4113, 4115 - 4121 4200 - 4203, 4300 - 4604	Bei der Ermittlung der Anzeigedaten ist ein allgemeiner Fehler aufgetreten.	
3102 4114, 4122, 4123	Die Parameter des Aufrufs sind nicht korrekt.	Überprüfen Sie, ob alle Parameter des Aufrufs in der vorgegebenen Reihenfolge angegeben wurden. Stellen Sie sicher, dass die Angaben vollständig und korrekt sind.
3103	Die PLC wurde nicht gefunden oder der Name der PLC ist falsch.	Überprüfen Sie den angegebenen PLC-Namen.
3104	Die Adresse des Codebausteins konnte nicht ermittelt werden.	
3105	Die Adresse des Instanz-Datenbausteins konnte nicht ermittelt werden.	Stellen Sie sicher, dass im Parameter der Name des Instanz-Datenbausteins und nicht der Name des Funktionsbausteins angegeben wurde.
3106	Der Baustein ist Know-how-geschützt.	

Fehlernummer	Fehlermeldung	Hinweise
3108	Die Version des Engineering System ist nicht kompatibel.	Stellen Sie sicher, dass das Anwenderprogramm und die Visualisierung mit der entsprechenden Version des TIA Portals projektiert wurden. Die PLC-Code-Anzeige mit WinCC V14 unterstützt nur die Projektierung mit Engineering System des TIA Portals V14.
3112	Das Netzwerk enthält einen Operanden, der nicht unterstützt wird.	
3117	Die Netzwerksprache wird nicht unterstützt.	
3201	Der Operand wurde nicht gefunden.	
3301	Der aufgerufene Baustein wurde nicht gefunden.	
3302	Es wurde kein Baustein gefunden, an dem dieser UDT verwendet wird.	
3303	Es wurden keine passenden Eingänge gefunden, oder die Eingänge enthalten nicht unterstützte Datentypen.	
3401	Bei der Ermittlung der Daten eines GRAPH-Bausteins ist ein Fehler aufgetreten.	Stellen Sie sicher, dass der angegebene Baustein ein GRAPH-Baustein ist.
3114, 3110, 3111 4103	Das Netzwerk enthält ein Element, das nicht unterstützt wird.	
4106	Bei der Berechnung des Online-Status ist ein Fehler aufgetreten.	Trennen Sie das Bediengerät vom Netzwerk, indem Sie das Netzwerkkabel ausstecken. Warten Sie einen Moment und stecken Sie das Netzwerkkabel wieder ein.
4108	Der PLC-Name wurde nicht festgelegt.	
4109	Der DB-Name wurde nicht festgelegt.	
4202	Der ProDiag UDT wurde nicht festgelegt.	

12.3.5 Beispiel: Visualisierung der ProDiag-Funktionalität

12.3.5.1 Handlungsübersicht

Einleitung

Mit ProDiag haben Sie die Möglichkeit, Störungen in Ihrer Anlage zu überwachen und Informationen zu den Überwachungen an Ihrem Bediengerät anzuzeigen.

Folgendes Beispiel zeigt, wie Sie mit WinCC Objekte zur Überwachung und Diagnose von Anlagen einsetzen.

Beispiel

In diesem Beispiel projektieren Sie die Visualisierung für die Überwachung des Betriebs eines Förderbandes.

Mit einer zuvor erstellen ProDiag-Überwachung verfolgen Sie die Bewegung der Produktionsstücke auf dem Förderband.

In STEP 7 erstellen Sie Ihr Anwenderprogramm und legen ProDiag-Überwachungen für die Operanden an. In WinCC projektieren Sie dann die Bilder und Objekte für die Anzeige des Programms, des aktuellen Programmstatus und die Ausgabe von Fehlermeldungen.

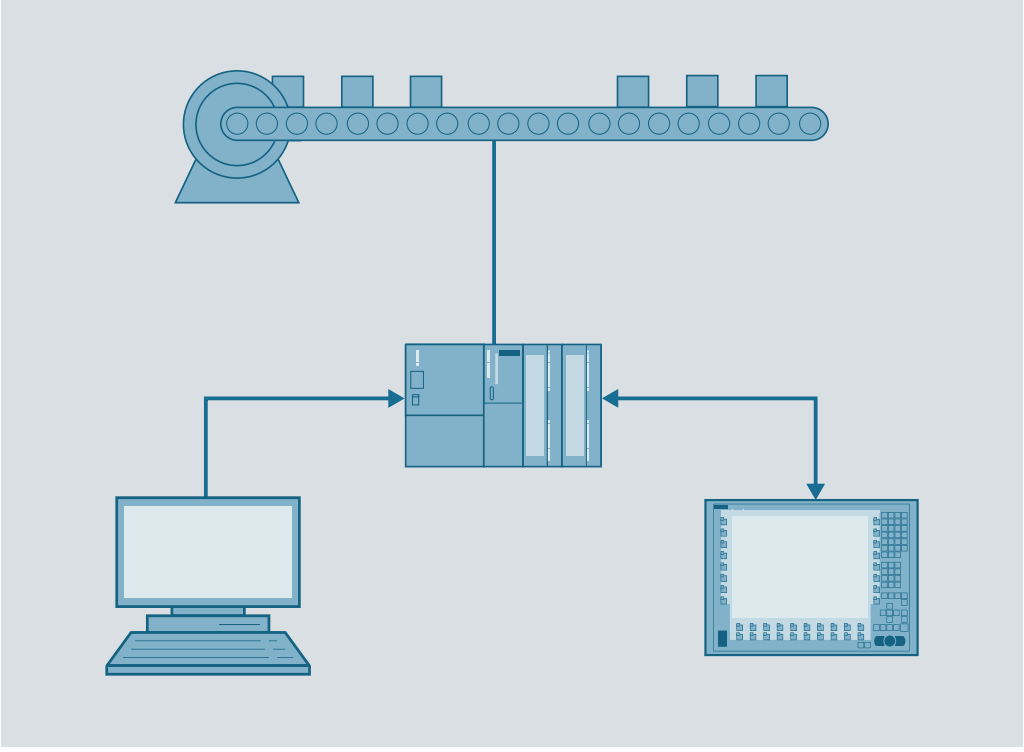

Projektierungsschritte

Das Beispiel ist in mehrere Schritte aufgeteilt:

- Bilder erstellen
  - Übersichtbild mit den ProDiag-Bildobjekten
  - Bild für die Meldeanzeige
  - Bild für die PLC-Code-Anzeige
- Objekte in Bilder einfügen
  - ProDiag-Übersicht für die Anzeige der ProDiag-Überwachungen
  - GRAPH-Übersicht für die Anzeige der GRAPH-Ablaufkette
  - Meldeanzeige für die Anzeige der ProDiag-Meldungen
  - PLC-Code-Anzeige für die Anzeige des Programcode in den Sprachen KOP, FUP und GRAPH

- Objekte konfigurieren
  Die Objekte verschalten, um eine gezielte Navigation zur Fehlerursache in Runtime zu ermöglichen.

12.3.5.2 Voraussetzungen

Voraussetzungen

- Eine S7-1500 Steuerung mit der Firmware ab Version 2.0 ist angelegt.
- Ein Anwenderprogramm zum Steuern des Förderbandes wurde erstellt.
- Das Anwenderprogramm enthält einen GRAPH-Funktionsbaustein mit dem Bewegungsablauf des Förderbandes.
- Das Anwenderprogramm enthält einen ProDiag-Funktionsbaustein mit den projektierten Operandenüberwachungen.
  Weitere Informationen zur Erstellung von ProDiag-Überwachungen finden Sie im Kapitel "PLC programmieren > Maschinen und Anlagen mit ProDiag überwachen".
- Das Bediengerät TP1200 Comfort ist angelegt.
- Eine Verbindung zwischen der Steuerung und Ihrem Bediengerät ist angelegt.

Steuerungsprogramm

Zuvor wurde in STEP 7 ein Programm für den Ablauf des Fließbandes sowie ein ProDiag-Funktionsbaustein mit Operandenüberwachungen erstellt. Nachfolgende Tabelle zeigt die Bausteine und Variablen, die in diesem Beispiel der Visualisierung mit WinCC verwendet werden:

Name	Bezeichnung/Typ	Erklärung
Conv_Seq [FB]	GRAPH-Funktionsbaustein	Beinhaltet den Programmablauf der Bewegung des Fließbandes.
Conv_Seq_DB	GRAPH-Datenbaustein	Zugehöriger GRAPH-Instanzdatenbaustein.
OFF_SQ	Variable vom Typ Bool aus dem GRAPH-Datenbaustein	PLC-Variable, die als Prozessvariable für die Anzeige von Status der GRAPH-Ablaufkette verwendet wird.
Conveyor [FB]	ProDiag-Funktionsbaustein	ProDiag-Funktionsbaustein beinhaltet die projektierten ProDiag-Überwachungen für die Operanden, die den Betrieb des Fließbandes ansteuern.
Conveyor_DB	ProDiag-Datenbaustein	Zugehöriger ProDiag-Instanzdatenbaustein.
State	Variable vom Typ SV_FB_State aus dem ProDiag-Datenbaustein	Zustandsvariable vom PLC-Datentyp "SV_FB_State", beinhaltet die Sammelfehlerbits aller Überwachungen, die dem ProDiag-Funktionsbaustein zugeordnet sind.

12.3.5.3 Bilder erstellen

Einleitung

Sie erstellen drei Bilder für die Anzeige verschiedener Objekte zum Visualisieren der ProDiag-Überwachungen in Ihrer Anlage.

Vorgehensweise

1. Öffnen Sie die Projektnavigation.
2. Klicken Sie im Ordner Bilder auf "Neues Bild hinzufügen".
3. Öffnen Sie das Kontextmenü des Bildes und vergeben Sie den Namen "Overview".
4. Erstellen Sie zwei weitere Bilder mit den Namen "Alarm" und "Program code".

Ergebnis

Sie haben ein Übersichtsbild und zwei weitere Bilder für die Visualisierung der Überwachungen in Runtime erstellt.

12.3.5.4 Diagnoseobjekte einfügen

Einleitung

Um den Programmcode und den Programmstatus in Runtime auf Ihrem Bediengerät zu visualisieren, fügen Sie die Objekte zur Diagnose in die erstellten Bilder ein.

In der GRAPH-Übersicht sehen Sie in Runtime die aktuell ausgeführten Schritte der GRAPH-Ablaufkette, die sie projektiert haben.

ProDiag-Übersicht zeigt Ihnen, ob eine Störung aufgetreten ist und eine ProDiag-Überwachung ausgelöst wurde.

Voraussetzung

- Die Bilder "Program code", "Overview" und "Alarm" sind angelegt.
- Die Task Card "Werkzeuge" ist geöffnet.

Vorgehensweise

1. Öffnen Sie das Bild "Program code".
2. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" eine "PLC-Code-Anzeige" in das Bild.
3. Vergeben Sie im Inspektofenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" den Objektnamen "PLC code view".
4. Öffnen Sie das Bild "Overview".

5. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" das Objekt "ProDiag-Übersicht" in das Bild.
   Falls Sie mehrere ProDiag-Bausteine aus Ihrem Programm überwachen möchten, legen Sie für jeden Baustein eine ProDiag-Übersicht an.
6. Vergeben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" sinnvollen Objektnamen, z. B. "Overview_Conveyor".
7. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" eine "GRAPH-Übersicht" in das Bild.
8. Vergeben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" den Namen "GRAPH_Sequence".
9. Öffnen Sie das Bild "Alarm".
10. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" eine Meldeanzeige in das Bild.
11. Vergeben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" den Namen "ProDiag_Alarm".

Ergebnis

Sie haben die Objekte zur Überwachung und Diagnose in den jeweiligen Bildern angelegt.

- Objekte zur Anzeige des aktuellen Programmstatus im Bild "Overview".
- PLC-Code-Anzeige zur Anzeige von Programmcode in Sprachen KOP, FUP und GRAPH im Bild "Program code".
- Meldeanzeige zur Anzeige von ProDiag-Meldungen im Bild "Alarm".

12.3.5.5 Diagnoseobjekte konfigurieren

Einleitung

In diesem Beispiel verbinden Sie die Objekte zur Visualisierung von Programmcode und Überwachungen mit Ihrem Steuerungsprogramm. Zuvor haben Sie in STEP 7 ein Programm zum Ablauf des Transports der Produktionsstücke am Fließband erstellt. Außerdem haben Sie einen ProDiag-Funktionsbaustein erstellt, in dem Sie die Überwachungen für die Operanden angelegt haben.

Voraussetzung

- Das Bild "Overview" ist angelegt.
- Im Bild ist eine GRAPH-Übersicht und eine ProDiag-Übersicht angelegt.
- Ein ProDiag-Funktionsbaustein wurde im Steuerungsprogramm angelegt.
- Ein GRAPH-Instanzdatenbaustein "Conv_Seq_DB" wurde im Steuerungsprogramm angelegt.
- GRAPH-Instanzdatenbaustein "Conv_Seq_DB" enthält die Variable "OFF_SQ", die in HMI sichtbar und aus HMI erreichbar ist.

Vorgehensweise

1. Selektieren Sie die GRAPH-Übersicht.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Prozess > Variable" auf die Auswahlschaltfläche.
   Ein Dialog wird geöffnet.
3. Klicken Sie im Ordner "Programmbausteine" auf den GRAPH-Instanzdazenbaustein "Conv_Seq_DB".
4. Selektieren Sie die PLC-Variable "OFF_SQ" des GRAPH-Instanzdatenbausteins und bestätigen Sie Ihre Auswahl.

   Hinweis

   Die Prozessvariable, die Sie für die GRAPH-Übersicht verwenden, muss aus HMI sichtbar und aus HMI erreichbar sein.

   Um die Variablen des GRAPH-Datenbausteins für HMI sichtbar und erreichbar zu kennzeichnen, öffnen Sie den GRAPH-Funktionsbaustein, selektieren Sie den Baustein im Arbeitsbereich und wählen Sie in der Menüleiste "Bearbeiten > Interne Parameter sichtbar/erreichbar aus HMI". Übersetzten Sie anschließend die Programmbausteine.

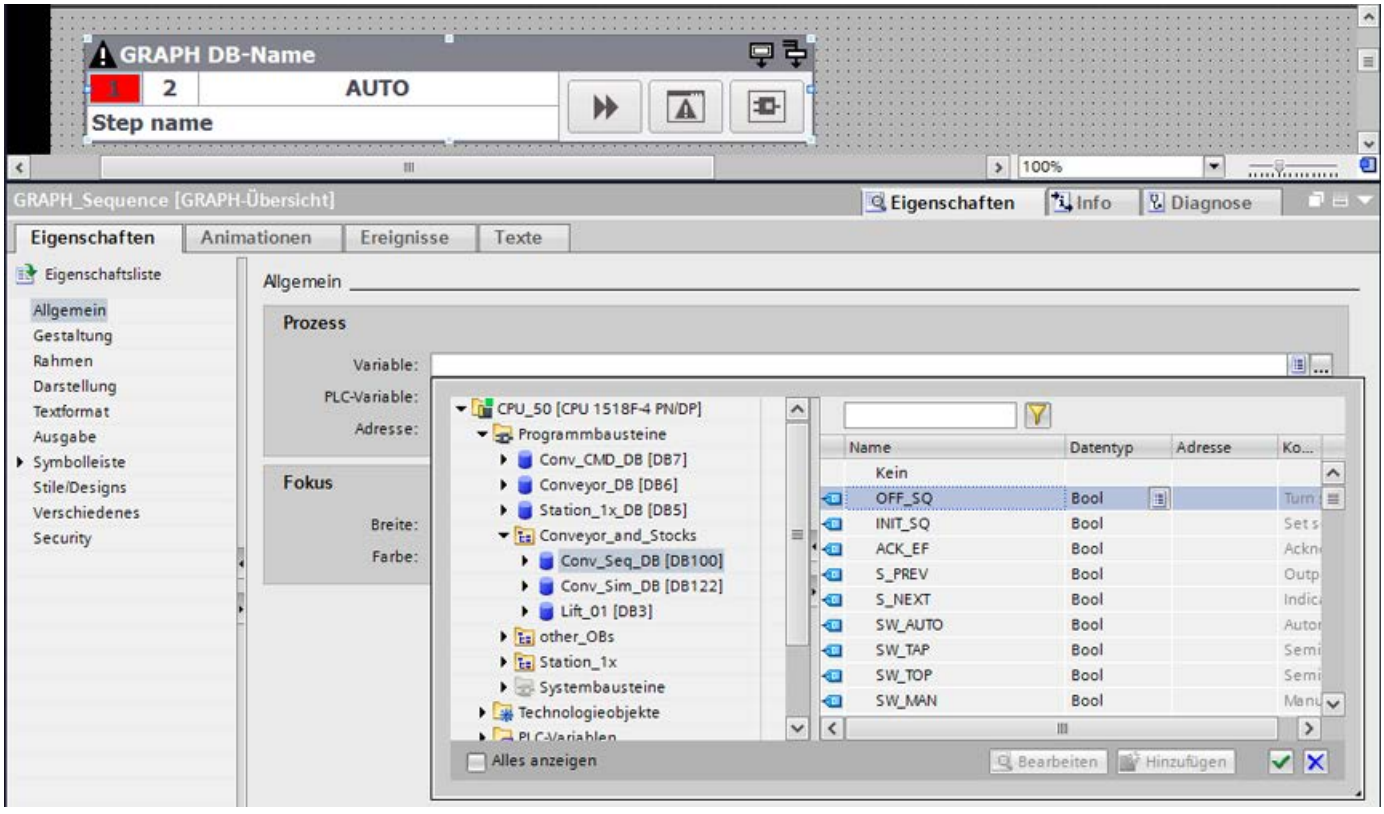

   Der GRAPH-Instanzdatenbaustein und das Objekt "GRAPH-Übersicht" sind verbunden.

5. Selektieren Sie die ProDiag-Übersicht "Overview_Conveyor".

6. Klicken Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Prozess > Variable" auf die Auswahlschaltfläche.
   Ein Dialog wird geöffnet.
7. Klicken Sie im Ordner "Programmbausteine" auf den Datenbaustein "Coveyor_DB".
8. Selektieren Sie die Zustandsvariable "State" des Datenbausteins "Conveyor_DB".

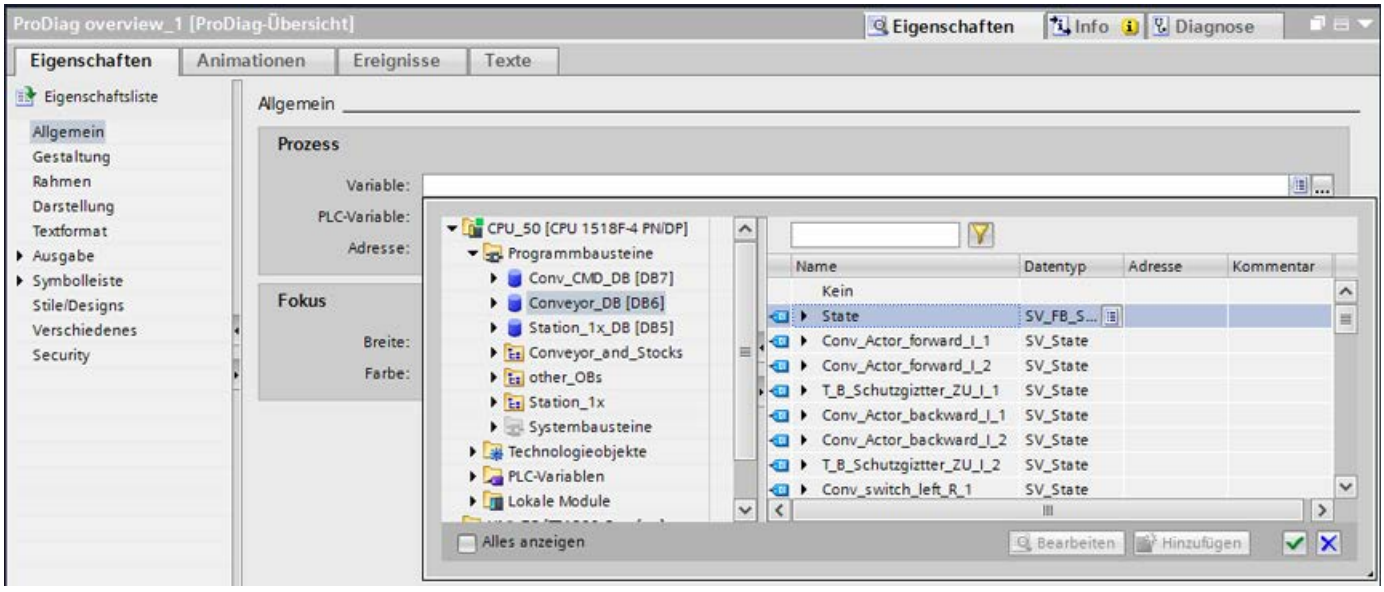

Der ProDiag-Datenbaustein und das Objekt "ProDiag-Übersicht" sind verbunden.

Ergebnis

GRAPH-Übersicht und ProDiag-Übersicht sind mit den Datenbausteinen des Steuerungsprogramms verbunden. In Runtime zeigen die Objekte die kommenden Überwachungsfehler und den aktuellen Status der GRAPH-Ablaufkette.

12.3.5.6 Meldeanzeige für ProDiag konfigurieren

Einleitung

In diesem Beispiel konfigurieren Sie die Meldeanzeige für die Darstellung der Überwachungsmeldungen und den Einsprung in die PLC-Code-Anzeige mit dem Programmcode.

Voraussetzung

- Das Bild "Alarm" ist angelegt.
- Die Meldeanzeige "ProDiag_Alarm" ist angelegt.
- Die PLC-Code-Anzeige "PLC code view" ist im Bild "Program code" angelegt.

Vorgehensweise

1. Öffnen Sie das Bild "Alarm" und selektieren Sie die Meldeanzeige.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Allgemein".
3. Aktivieren Sie die Meldeklassen von Meldungen, die in der Meldeanzeige dargestellt werden, z.B. die systemeigenen Meldeklassen "Acknowledgement" und "No Acknowledgement".
   Sie haben die Möglichkeit, auch Meldungen der Meldeklassen anzuzeigen, die Sie selbst zuvor in STEP 7 definiert haben.

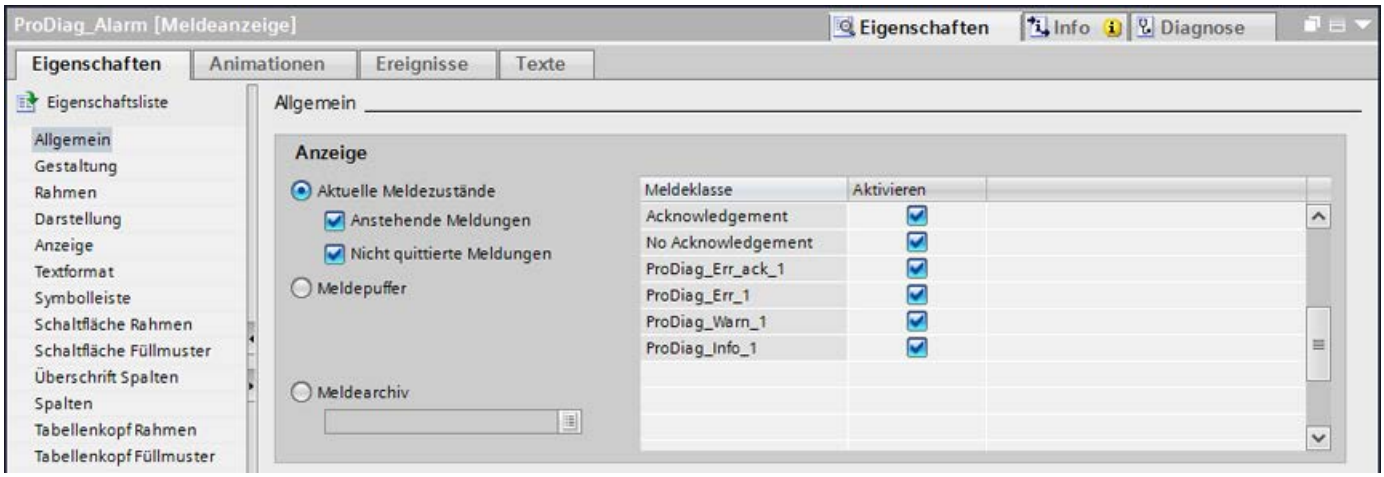

4. Legen Sie unter "Eigenschaften > Anzeige > Steuervariable für PLC-Code-Anzeige" eine boolesche Variable fest, die den Einsprung von der letzten aktiven ProDiag-Meldung in die PLC-Code-Anzeige steuern wird.
   Diese Variable dient dazu es auszuwerten, ob der Einsprung von der selektierten Meldung in die PLC-Code-Anzeige möglich ist.

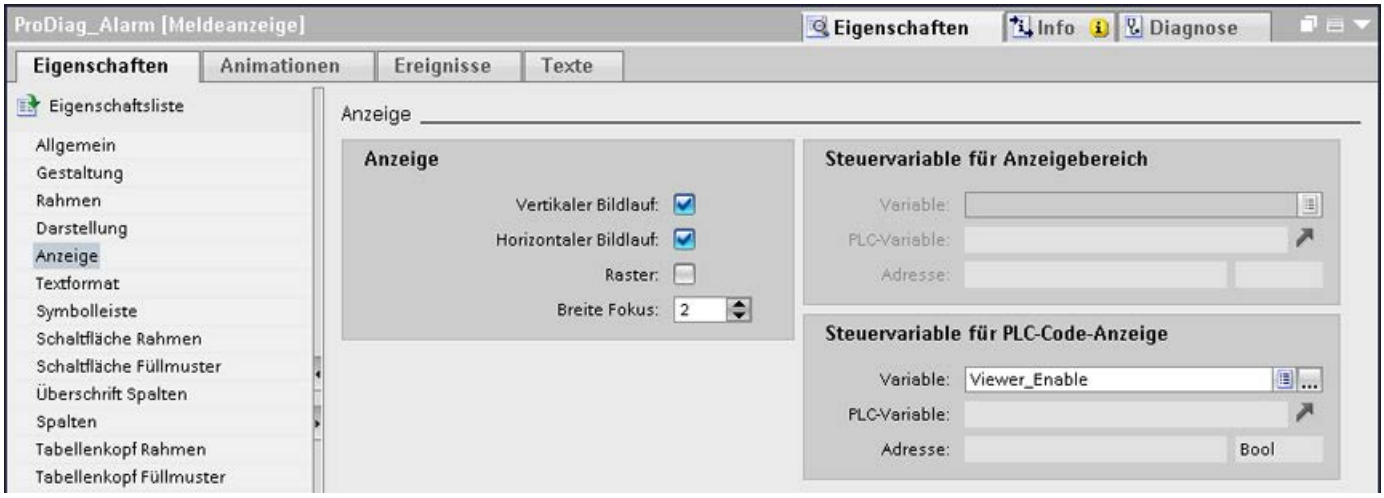

5. Ziehen Sie per Drag&Drop aus der Task Card "Werkzeuge" eine Schaltfläche in das Bild "Alarm".
   Vergeben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Verschiedenes > Name" den Namen "Program code".
6. Klicken Sie im Inspektorfenster "Ereignisse > Klicken".

7. Wählen Sie unter "Funktion hinzufügen" die Systemfunktion "AktivierePLCCodeAnzeige".
8. Geben Sie als Parameter den Bildnamen "Program code" und den Objektnamen "PLC code view" ein.
9. Wählen Sie unter "Eigenschaften > Animationen > Sichtbarkeit > Variable" die zuvor festgelegte boolesche Steuervariable für PLC-Code-Anzeige.

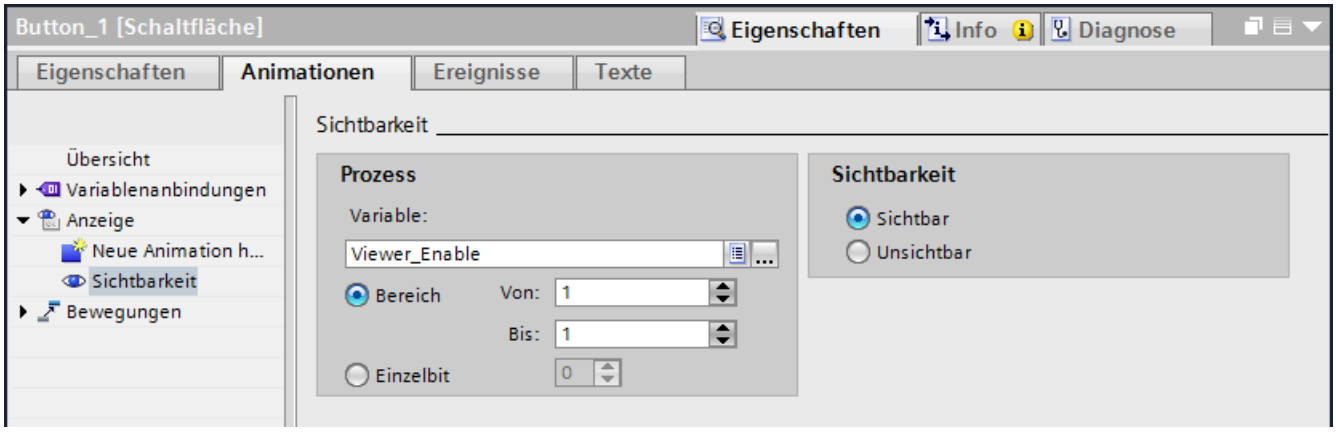

Wenn der Sprung von der gekommenen ProDiag-Meldung in die PLC-Code-Anzeige möglich ist, wird die Schaltfläche "Program code" sichtbar.

Ergebnis

Sie haben die Anzeige von Meldungen der definierten Meldeklassen in der Meldeanzeige aktiviert. In Runtime werden in der Meldeanzeige die Meldungen der entsprechenden Meldeklassen angezeigt.

Über die projektierte Schaltfläche springen Sie in Runtime in der PLC-Code-Anzeige mit der Programmstelle, die die ProDiag-Meldung ausgelöst hat.

12.3.5.7 Einsprung von der GRAPH-Übersicht projektieren

Einleitung

In diesem Schritt konfigurieren Sie die einfache Navigation zwischen der GRAPH-Übersicht, der Meldeanzeige und der PLC-Code-Anzeige.

Einsprung von der GRAPH-Übersicht projektieren

1. Selektieren Sie die GRAPH-Übersicht "GRAPH_Sequence".
2. Klicken Sie im Inspektorfenster "Ereignisse > Schaltfläche Meldeanzeige klicken".
3. Wählen Sie unter "Funktion hinzufügen" die Systemfunktion "AktiviereBild".
4. Geben Sie den Namen des Bilds "Alarm" an, welches die Meldeanzeige enthält.
5. Klicken Sie im Inspektorfenster "Ereignisse > Schaltfläche PLC-Code-Anzeige klicken".

6. Wählen Sie unter "Funktion hinzufügen" die Systemfunktion "AktivierePLCCodeAnzeige".
7. Geben Sie den Namen des Bilds "Program code" und der PLC-Code-Anzeige "PLC code view" an.

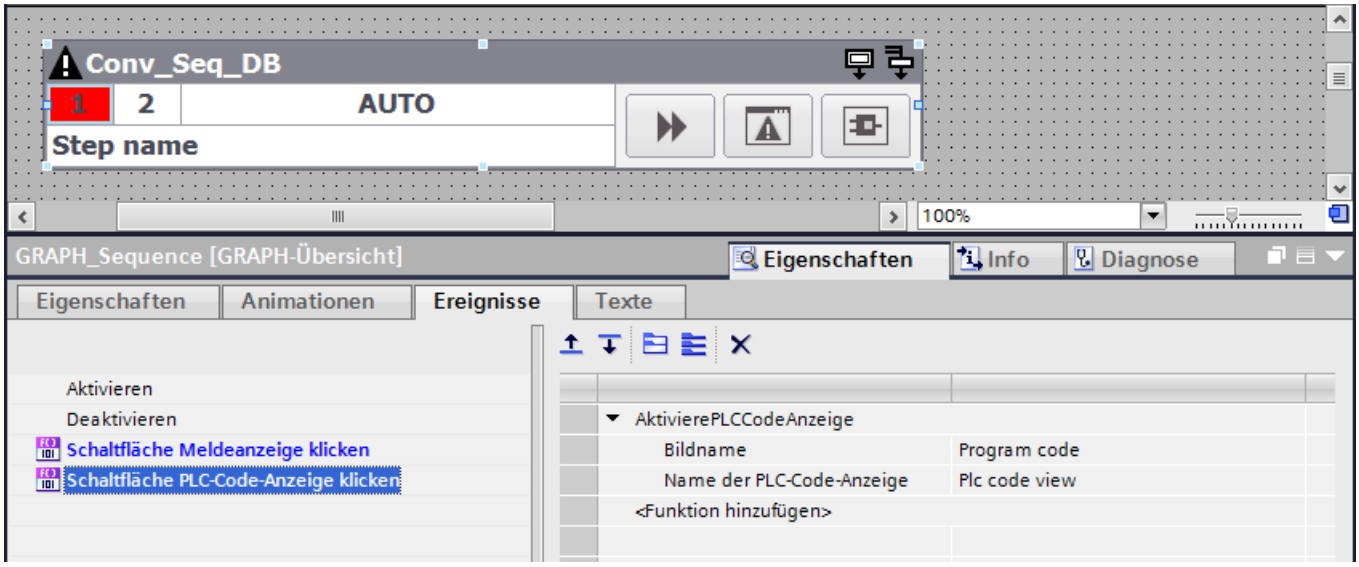

Sie haben die GRAPH-Übersicht mit der PLC-Code-Anzeige verbunden. Durch Klicken der Schaltfläche "PLC-Code-Anzeige" wird in Runtime das Bild mit der Anzeige der GRAPH-Ablaufkette eingeblendet.

Ergebnis

In Runtime überwachen Sie mithilfe der GRAPH-Übersicht den aktuellen Status der Ablaufkette. Sie haben die Möglichkeit, die GRAPH-Ablaufkette jederzeit in der PLC-Code-Anzeige anzusehen.

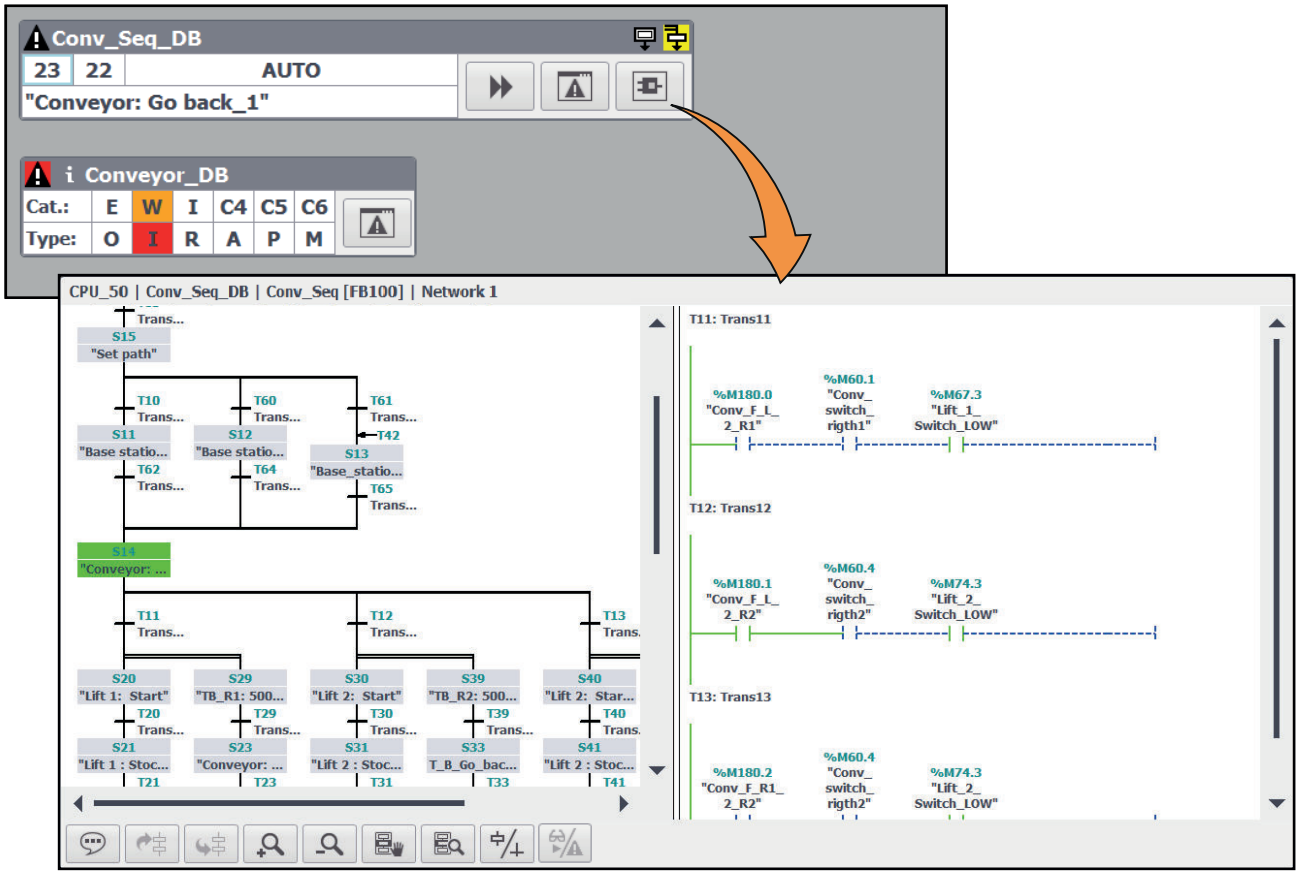

Im Fall einer Störung springen Sie mithilfe der konfigurierten Schaltflächen zur entsprechenden Meldung in der Meldeanzeige oder direkt in die PLC-Code-Anzeige mit der Darstellung der Schrittkette.

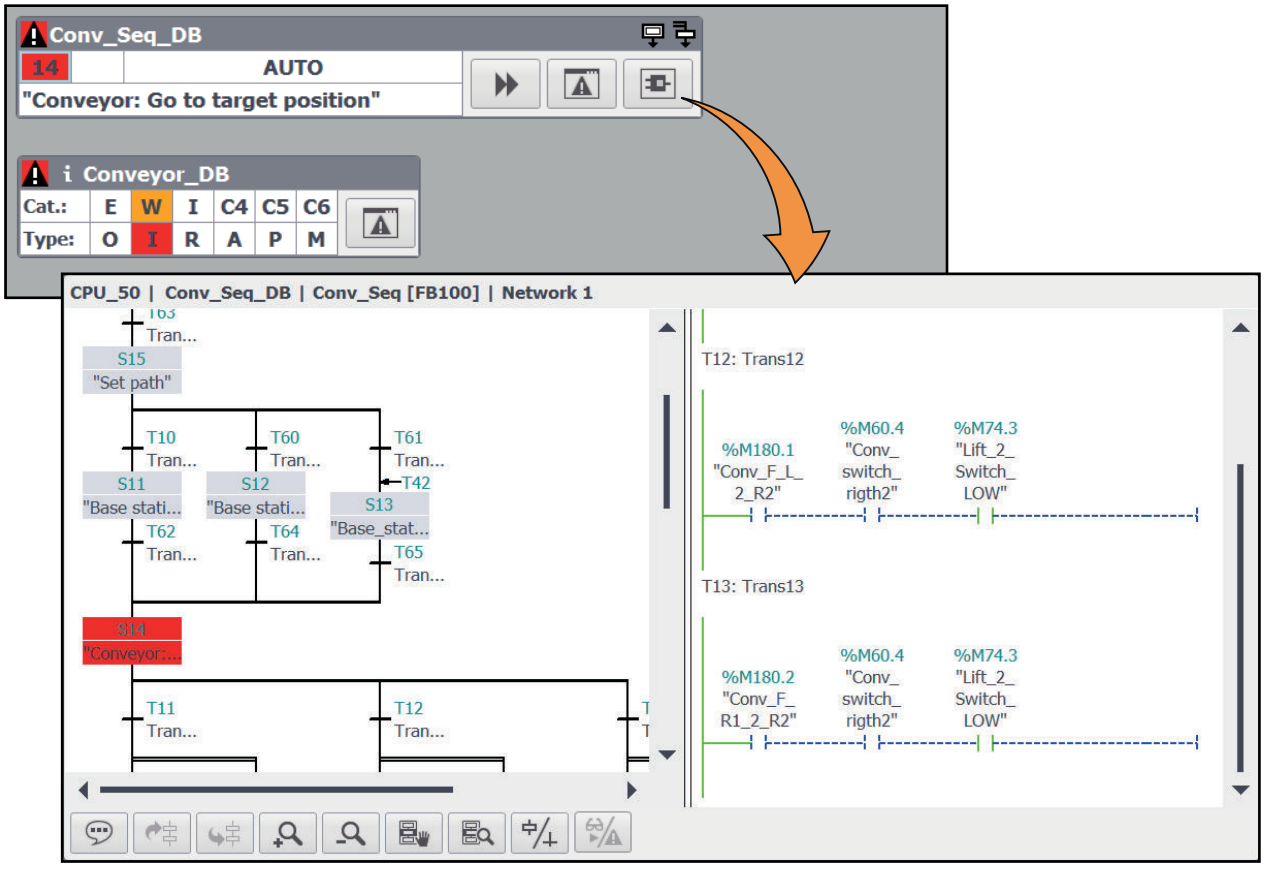

12.3.5.8 Einsprung von der ProDiag-Übersicht projektieren

Einleitung

In diesem Schritt konfigurieren Sie die einfache Navigation zwischen der ProDiag-Übersicht, der Meldeanzeige und der PLC-Code-Anzeige.

Einsprung von der ProDiag-Übersicht projektieren

1. Selektieren Sie die ProDiag-Übersicht "Overview_Conveyor".
2. Klicken Sie im Inspektorfenster "Ereignisse > Schaltfläche Meldeanzeige klicken".
3. Wählen Sie unter "Funktion hinzufügen" die Systemfunktion "AktiviereBild".
4. Geben Sie den Namen des Bilds "Alarm" an, welches die Meldeanzeige enthält.

Ergebnis

Sie haben die Diagnoseobjekte miteinander verbunden.

In Runtime überwachen Sie mithilfe der ProDiag-Übersicht den Status der ProDiag-Überwachungen. Im Fall eines Fehlers wird das entsprechende Überwachungssymbol visuell hervorgehoben.

Durch Klicken der Schaltfläche "Meldeanzeige" wird das Bild mit der Meldeanzeige und aktuell anstehenden Meldungen eingeblendet. Mithilfe der projektierten Schaltfläche springen Sie z.B. von einer ProDiag-Meldung zur Anzeige des Netzwerks in der PLC-Code-Anzeige.

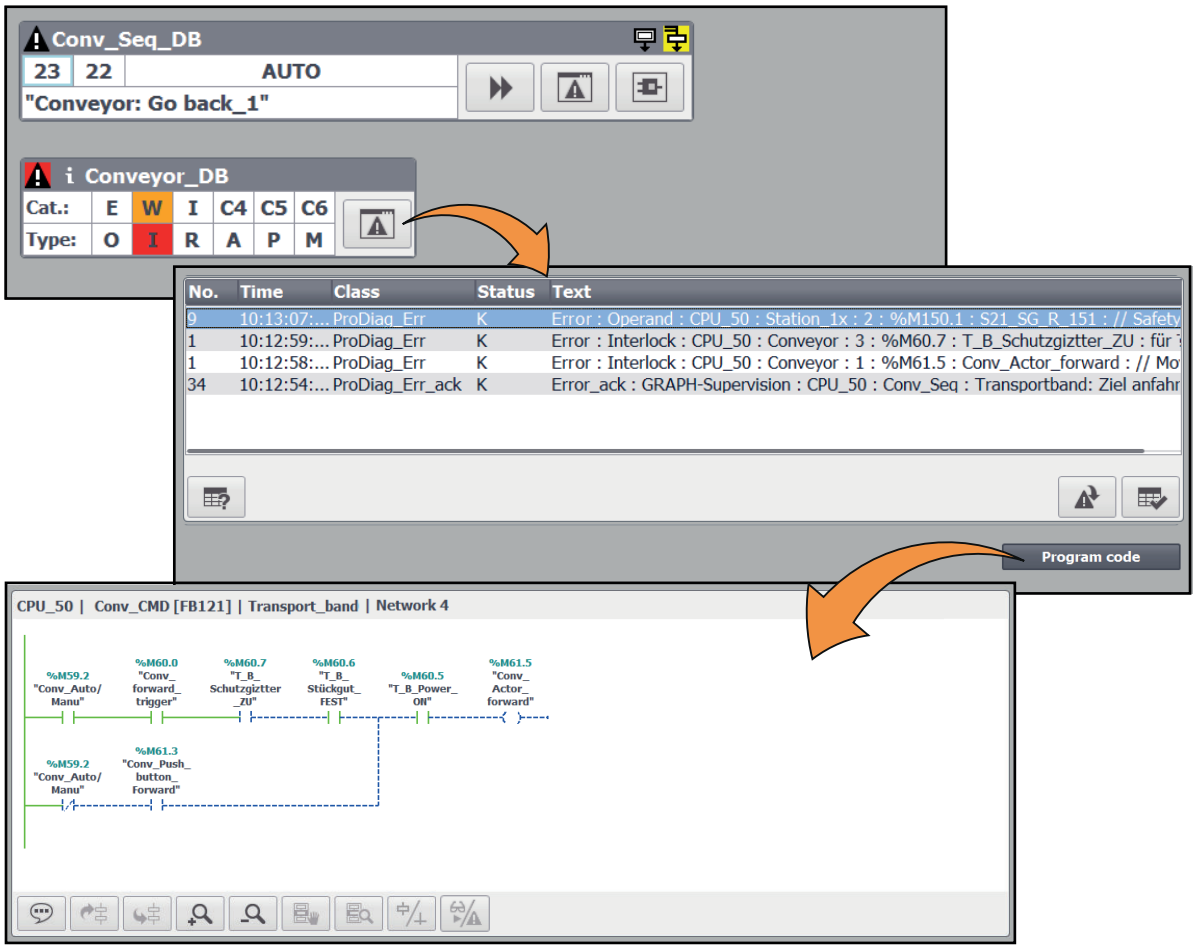

12.3.6 Erstwerterfassung und Kriterienanalyse

12.3.6.1 Überblick über die Erstwerterfassung und Kriterienanalyse

Einleitung

Im TIA Portal haben Sie die Möglichkeit, den Ablauf des Anwenderprogramms auf einem HMI-Gerät zu verfolgen. Dabei werden die Daten und Werte auf dem Bediengerät stetig mit der PLC abgeglichen und aktualisiert. Somit sehen Sie den aktuellen Programmstatus mit den Aktualwerten der Signalzustände auf dem Bediengerät.

Wenn in Ihrer Anlage ein Fehler auftritt, haben Sie die Möglichkeit aus der entsprechenden Fehlermeldung zum Programmcode zu springen und im Objekt "PLC-Code-Anzeige" die Fehlerstelle im Netzwerk anzeigen zu lassen. Im Objekt "Kriterienanalyse-Anzeige" sehen Sie die fehlerhaften Operanden zu einer ausgewählten GRAPH- oder ProDiag-Meldung auf einen Blick.

Die Funktionen Erstwerterfassung und Kriterienanalyse ermöglichen Ihnen die Aufzeichnung der Werte zum Zeitpunkt der Störung und schnelle Erkennung der fehlerhaften Operanden im Programm.

Die Funktionen Erstwerterfassung und Kriterienanalyse stehen für GRAPH-Funktionsbausteine, ProDiag-Funktionsbausteine und Sicherheitsprogramme (F-Bausteine) zur Verfügung.

Voraussetzung

- Die Erstwerterfassung ist im TIA Portal für folgende Bausteine aktivierbar:
  - Ab Version V14 SP1 für die GRAPH-Funktionsbausteine ab Version 4.0.
  - Ab Version V15 für die ProDiag-Funktionsbausteine ab Version 2.0.
- Maximal 32 Zustände können aufgezeichnet werden. Die Erstwerte für ein Netzwerk, das mehr als 32 Elemente enthält, werden nicht aufgezeichnet.

Erstwerterfassung

Mithilfe der Erstwerterfassung können Sie die Werte zum Zeitpunkt des Fehlers in der Steuerung erfassen, diese in der PLC-Code-Anzeige auf Ihrem Bediengerät anzeigen lassen und mit den Aktualwerten vergleichen. Mit der Erstwerterfassung zeichnen Sie die Signalzustände von booleschen Operanden und Ergebnisse von Vergleichern in Transitionen und Interlocks kontinuierlich auf.

Die Signalzustände werden nach einer definierten Reihenfolge von links oben nach rechts unten aufgezeichnet:

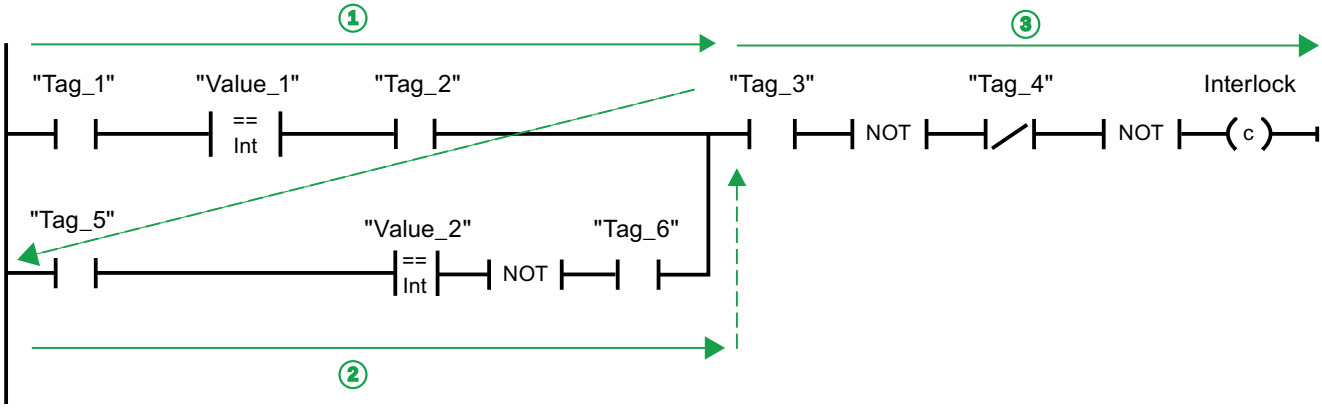

Erstwerterfassung aktivieren Sie einzeln für jeden GRAPH-Baustein im Anwenderprogramm. Es können maximal 32 Signalzustände von booleschen Operanden pro Interlock bzw. Transition eines GRAPH-Schritts aufgezeichnet werden. Jeder einzelne Signalzustand belegt ein Bit. Die Werte werden in einem DWORD gespeichert.

Im folgenden Beispiel sehen Sie das Prinzip und die Reihenfolge, nach der die Erstwerte in einem Interlock erfasst werden:

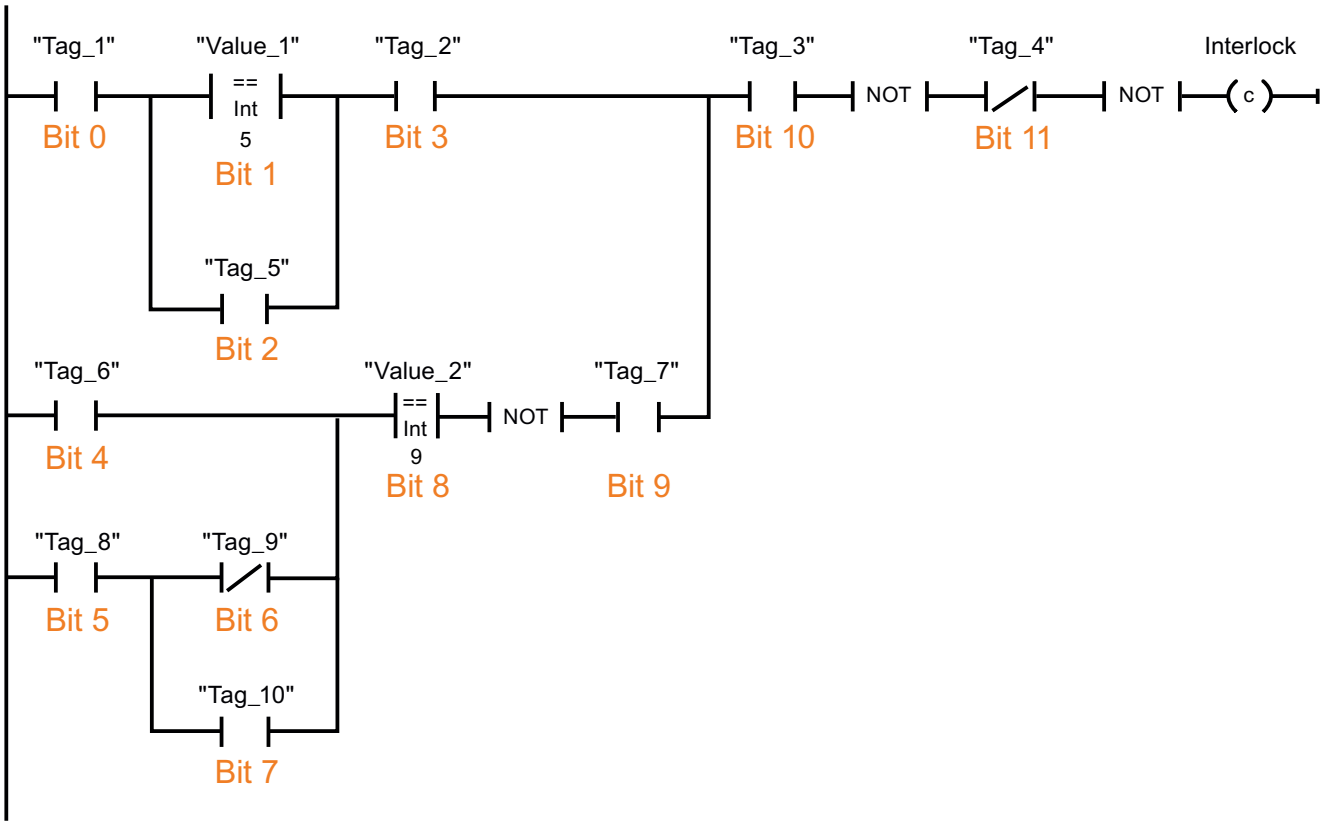

Kriterienanalyse

Die Erstwerterfassung in der PLC ermöglicht die Analyse der Kriterien und fehlerhaften Operanden im Programm. Die Auswertung der Kriterienanalyse sehen Sie auf Ihrem Bediengerät in der PLC-Code-Anzeige. Zusätzlich konnen Sie im Objekt "Kriterienanalyse-Anzeige" die fehlerhaften Operanden zu einer ausgewählten GRAPH- oder ProDiag-Meldung mit Kriterienanalyse anzeigen lassen.

Für die Bausteine, für welche Sie die Erstwerterfassung aktiviert haben, wird nach dem Einsprung standardmäßig die Erstwertansicht in der PLC-Code-Anzeige angezeigt. Zusätzlich werden in der Erstwertansicht die fehlerhaften Operanden und Kriterien visuell hervorgehoben.

Die Namen, die absolute Adressen und die Kommentare der fehlerhaften Operanden sehen Sie in der Symboltabelle. Sie können zwischen der Anzeige der Aktualwerte und der Anzeige der Erstwerte hin und her schalten.

Bei einem Fehler in einem Vergleicher werden beide Operanden als fehlerhaft gekennzeichnet. In der Erstwerte-Ansicht werden nur die aufgezeichneten Werte dargestellt. Um die Aktualwerte der Variablen zu sehen, wechseln Sie zur Aktualwerte-Ansicht.

Hinweis

Auswertungsreihenfolge bei der Programmierung beachten

Da die Operanden in der Symboltabelle nach der Regel der Erstwerterfassung aufgelistet werden, wird bei der Kriterienanalyse im Fehlerfall zuerst der linke oberste Operand in der Symboltabelle ausgegeben. Beachten Sie diese Auswertungsreihenfolge bei der Programmierung des Netzwerks, um die wichtigsten Operanden in der PLC-Code-Anzeige zuerst zu sehen.

Weitere Informationen zur Reihenfolge der Auflistung finden Sie im Beispiel für die Reihenfolge der Kriterienanalyse (Seite 2001)

Siehe auch

Beispiel für die Reihenfolge der Kriterienanalyse (Seite 2001)

12.3.6.2 Unterstützte Anweisungen

Einleitung

Die Erstwerte und die Ergebnisse der Kriterienanalyse sehen Sie im Objekt "PLC-Code-Anzeige".

Bei globalen Überwachungen werden die Erstwerte aller booleschen Operanden aufgezeichnet, die sich im Netzwerk befinden. Wenn das Netzwerk mehrere einzelne Stromschienen enthält, die nicht miteinander verbunden sind, dann werden nur die Erstwerte der jeweils betroffenen Stromschiene aufgezeichnet.

Bei lokalen Überwachungen werden nur die Erstwerte aufgezeichnet, die beim Bausteinaufruf als Bedingungen für den überwachten Parameter angegeben sind.

Unterstützte Anweisungen

Für die Erstwerterfassung werden folgende Anweisungen in KOP und FUP unterstützt:

Anweisungen	Anzeige auf dem Bediengerät (HMI)
Bitverknüpfungen	
Schließerkontakt	Erstwerte und Kriterienanalyse
Öffnerkontakt	
VKE invertieren	Die Anweisung wird unterstützt, ist aber nicht relevant für die Erstwerte bzw. die Kriterienanalyse.
Zuweisung	
Zuweisung negieren	

Anweisungen	Anzeige auf dem Bediengerät (HMI)
Ausgang rücksetzen	Erstwerte
Ausgang setzen	
Flipflop setzen/rücksetzen	
Flipflop rücksetzen/setzen	
Vergleicher	
Gleich	Erstwerte und Kriterienanalyse
Ungleich	
Größer gleich	
Kleiner gleich	
Größer	
Kleiner	
Zeiten	
TP	Erstwerte
TON	Erstwerte und Kriterienanalyse
TOF	Erstwerte
TONR	
Zähler	
CTU	Erstwerte
CTD	
CTUD	

Bei den Bitverknüpfungen wird der Zustand des Operanden und bei den Vergleichern das Ergebnis des Vergleichs aufgezeichnet.

Bei den Flipflops werden beide Eingänge (R und S) aufgezeichnet, wenn sie verschaltet sind.

Bei den Zeiten und Zählern wird der Zustand des Operanden am Ausgang und zusätzlich die Eingänge, wenn sie verschaltet sind, aufgezeichnet. (Zum Beispiel bei CTUD: CU, CD, R, LD)

Hinweis

Die FUP Anweisung EXKLUSIV ODER wird von der Erstwerterfassung und Kriterienanalyse nicht unterstützt.

Siehe auch

Unterstützte Datentypen (Seite 1978)

Unterstützte Datentypen (Seite 1965)

12.3.6.3 Beispiel für die Reihenfolge der Kriterienanalyse

Einleitung

Die Auswertung der Kriterienanalyse in der PLC-Code-Anzeige läuft nach einer fest definierten Reihenfolge ab. In diesem Beispiel sehen Sie wie die Erstwerterfassung und die Kriterienanalyse auf einem Bediengerät funktionieren.

Erstwerterfassung

Sie erfassen den Signalzustand einer Verriegelung im Automatik- und Handbetrieb im Anwenderprogramm und haben folgende Operanden projektiert:

Symbolname	Kommentar	Bedeutung
Tag_1	Auto mode	Automatikbetrieb
Tag_2	Interlock_1	Erste Verriegelung
Tag_3	Motor On	Motor einschalten
Tag_4	Interlock_2	Zweite Verriegelung
Tag_5	Hand mode	Handbetrieb
Tag_6	Hand switch	Handtaster

Die Signalzustände der Operanden werden in der folgenden Reihenfolge erfasst und in das DWORD geschrieben:

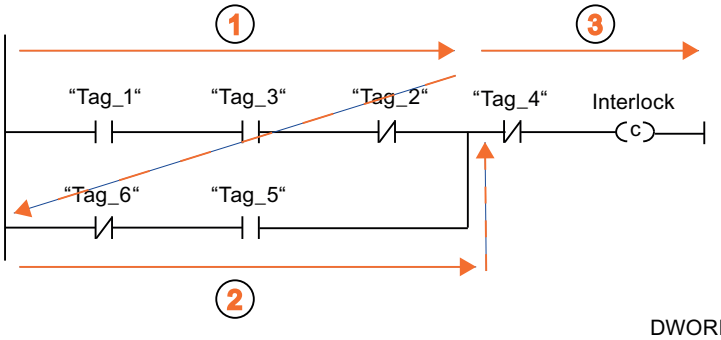

DWORD

Bit 31	Bit ...	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	Tag_4	Tag_5	Tag_6	Tag_2	Tag_3	Tag_1

Kriterienanalyse im Fehlerfall

Wenn im Interlock ein Fehler auftritt, nutzen Sie die projektierte Schaltfläche, um von der Fehlermeldung in die PLC-Code-Anzeige zu springen und die Fehlerstelle im Programm zu sehen. Dadurch, dass Sie die Erstwerterfassung bereits aktiviert haben, wird in der PLC-Code-Anzeige automatisch die Erstwert-Ansicht mit der Kriterienanalyse eingeblendet.

Die fehlerauslösenden Operanden werden farblich gekennzeichnet. In der Symboltabelle der PLC-Code-Anzeige sehen Sie nur die Namen, absolute Adressen und Kommentare der fehlerhaften Operanden nach der Reihenfolge der Erstwerterfassung.

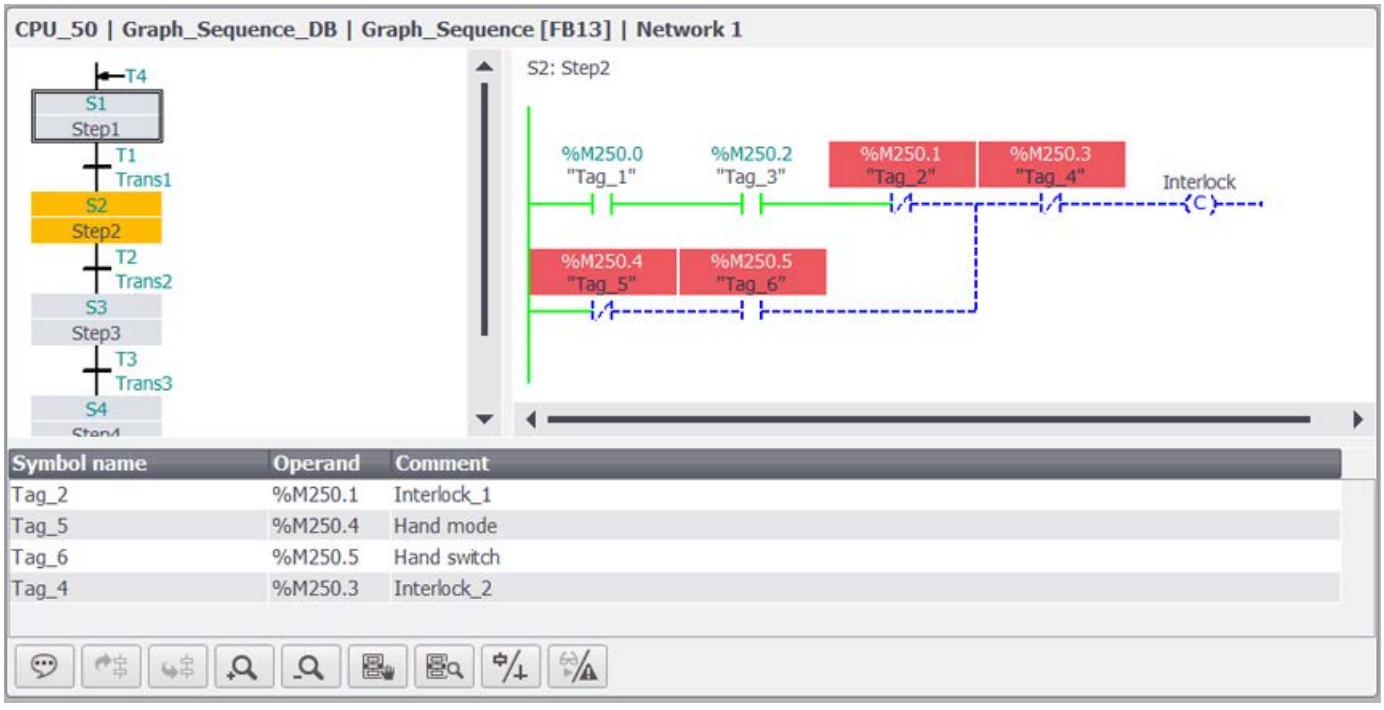

Sie haben die Möglichkeit, in den Hand-Modus zu schalten und Kriterienanalyse zu aktivieren. Dadurch können Sie in der Detail-Ansicht das fehlerauslösende Netzwerk auch dann sehen, wenn die Fehlersituation zwischenzeitlich wieder behoben wurde.

Optimierte Programmierung des Interlocks

Damit Sie schnell auf den ersten Blick erkennen können, dass die Lichtschranke für den Fehler verantwortlich ist, ist es in diesem Beispiel von Vorteil, den Operanden Tag_4 an erster Stelle zu programmieren. Damit wird die Information zur offenen Lichtschranke in der ersten Zeile der Symboltabelle dargestellt.

Das folgende Beispiel zeigt die Programmierung für die optimale Ansicht im Objekt "PLC-Code-Anzeige":

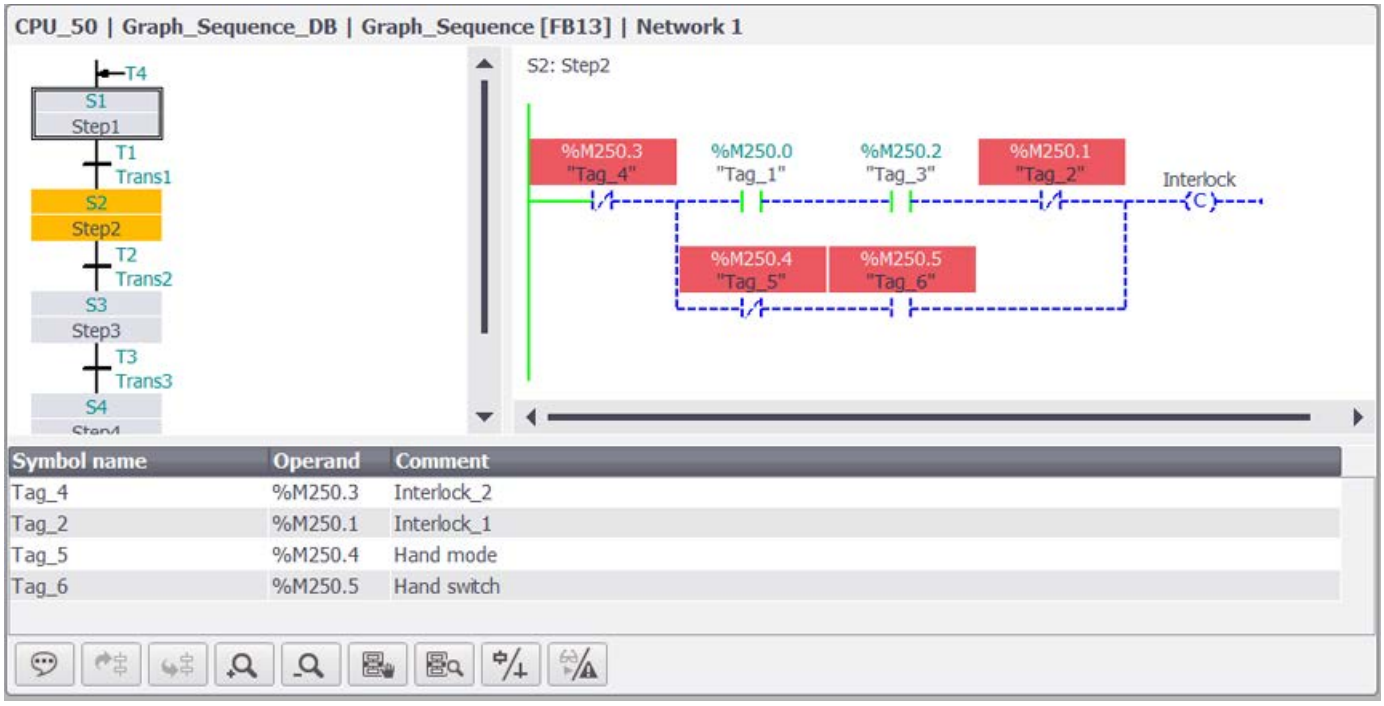

12.3.6.4 Kriterienanalyse-Anzeige projektieren

Verwendung

Das Objekt "Kriterienanalyse-Anzeige" zeigt Ihnen die fehlerhaften Operanden des Anwenderprogramms, die eine ausgewählte ProDiag-Meldung oder eine GRAPH-Meldung ausgelöst haben. Um die Verbindung zu den entsprechenden Meldungen herzustellen, projektieren Sie an den beiden Objekten Steuervariablen.

Wenn Sie in Runtime die gekommene ProDiag- oder GRAPH-Meldung in der Meldeanzeige auswählen, sehen Sie in der Kriterienanalyse-Anzeige die Operanden, die in der Kriterienanalyse ermittelt wurden. Die Meldungen und die Operanden sehen Sie in Runtime auf einen Blick, wenn Sie die Kriterienanalyse-Anzeige und die damit verbundene Meldeanzeige im gleichen Bild projektieren.

Sie haben die Möglichkeit, mehrere Meldeanzeigen mit einer Kriterienanalyse-Anzeige zu verbinden und umgekehrt. Auf KTP Mobile Panels, Comfort Panels und RT Advanced können Sie die Kriterienanalyse-Anzeige in einem Pop-up-Bild oder in einem Slide-in-Bild zu projektieren. Somit können Sie z. B. mehrere Meldeanzeigen, die in verschiedenen Bildern projektiert sind, mit einer Kriterienanalyse-Anzeige verbinden, die Sie in einem Slide-In-Bild bei Bedarf einblenden.

Hinweis

Das Objekt "Kriterienanalyse-Anzeige" steht im Globalen Bild nicht zur Verfügung.

Voraussetzung

- Das Bediengerät ist mit der Steuerung verbunden
- Die Steuerung enthält ein ProDiag-Programm mit der Version 2.0 oder höher oder ein GRAPH-Programm mit der Version 4.0 oder höher
- Die Erstwerterfassung ist bei den Funktionsbausteinen aktiviert
- Eine Meldeanzeige ist projektiert

Vorgehen

1. Ziehen Sie per Drag&Drop die Kriterienanalyse-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie unter der Eigenschaft "Variable" die Auswahlschaltfläche.
4. Selektieren Sie unter "Prozess > Variable" die Zustandsvariable der entsprechenden Meldeanzeige.
5. Klicken Sie auf die Meldeanzeige.
6. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Anzeige".
7. Selektieren Sie unter "Steuervariable für Kriterienanalyse" die Zustandsvariable der Kriterienanalyse-Anzeige.

Hinweis

Datentyp und Länge der Zustandsvariablen

Die Zustandsvariable muss vom Datentyp WString sein und die Länge von 50 betragen.

Ergebnis

Die Kriterienanalyse-Anzeige ist im Bild projektiert und mit der Meldeanzeige verbunden. In Runtime können Sie für eine ausgewählte Meldung die detaillierten Informationen über die Operanden sehen, die diese Meldung ausgelöst haben.

12.3.6.5 Kriterienanalyse-Anzeige projektieren

Verwendung

Das Objekt "Kriterienanalyse-Anzeige" zeigt Ihnen die fehlerhaften Operanden des Anwenderprogramms, die eine ausgewählte ProDiag-Meldung oder eine GRAPH-Meldung ausgelöst haben. Um die Verbindung zu den entsprechenden Meldungen herzustellen, projektieren Sie an den beiden Objekten Steuervariablen.

Wenn Sie in Runtime die gekommene ProDiag- oder GRAPH-Meldung in der Meldeanzeige auswählen, sehen Sie in der Kriterienanalyse-Anzeige die Operanden, die durch die Kriterienanalyse ermittelt wurden.

Die Kriterienanalyse-Anzeige und die damit verbundene Meldeanzeige projektieren Sie im gleichen Bild.

Voraussetzung

- Das Bediengerät ist mit der Steuerung verbunden
- Die Steuerung enthält ein ProDiag-Programm mit der Version 2.0 oder höher oder ein GRAPH-Programm mit der Version 4.0 oder höher
- Die Erstwerterfassung ist bei den Funktionsbausteinen aktiviert
- Eine Meldeanzeige ist projektiert

Vorgehen

1. Ziehen Sie per Drag&Drop die Kriterienanalyse-Anzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie unter der Eigenschaft "Datenquelle" die Auswahlschaltfläche.
4. Selektieren Sie die projektierte Meldeanzeige.

Ergebnis

Die Kriterienanalyse-Anzeige ist im Bild projektiert und mit der Meldeanzeige verbunden. In Runtime können Sie für eine ausgewählte Meldung die detaillierten Informationen über die Operanden sehen, die diese Meldung ausgelöst haben.

12.3.6.6 Meldungen mit Kriterien ausgeben

Einleitung

Bei der aktivierten Erstwerterfassung werden die Werte des fehlerhaften Operanden zum Zeitpunkt der Störung aufgezeichnet und fehlende Kriterien analysiert und ermittelt.

Sie haben die Möglichkeit, die GRAPH- und ProDiag-Meldungen mit Zusatzinformationen zu fehlerhaften Operanden zu erweitern und auf Ihrem Bediengerät auszugeben. Wenn in Runtime eine Störung im Programmablauf auftritt, zeigt die Fehlermeldung auch den ersten fehlerhaften Operanden im gestörten Netzwerk. Die detaillierten Informationen zu allen fehlerhaften Operanden der Fehlermeldung sehen Sie dann im Objekt "Kriterienanalyse-Anzeige".

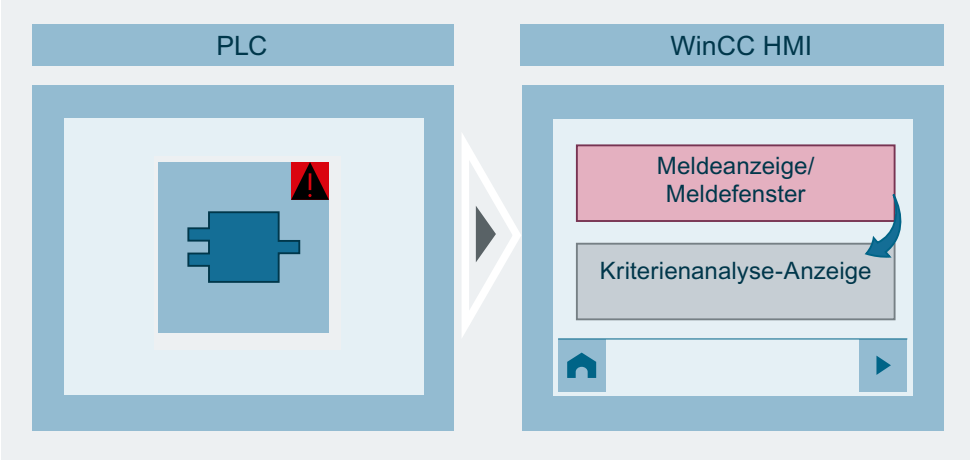

Folgende Informationen können zu einer Meldung hinzugefügt werden:

- Symbolname: Der symbolische Name des ersten Operanden, der die ausgewählte Fehlermeldung ausgelöst hat
- Absolute Adresse: Die Adresse des ersten fehlerhaften Operanden
- Wert des Operanden: Der Wert des ersten fehlerhaften Operanden zum Zeitpunkt der Störung
- Kommentar: Mehrsprachiger Kommentar, der in dem Anwenderprogramm projektiert wurde

Die Zusatzinformationen werden in der Meldung nach dem Meldetext angezeigt, jeweils getrennt durch Semikolon und Leerzeichen.

Alternativ können Sie festlegen, dass die Zusatzinformationen im Infotext angezeigt werden.

In WinCC Runtime Professional haben Sie die Möglichkeit, auch die zusätzlichen Meldetexte der Überwachungsmeldungen zu erweitern. Wählen Sie dafür unter "Runtime-Einstellungen > Meldungen > Kriterienanalyse > Text erweitern" den entsprechenden zusätzlichen Meldetext aus.

Hinweis

Die Reihenfolge der Zusatzinformationen, die zu der Meldung hinzugefügt werden, ist vorgegeben und kann nicht geändert werden.

Hinweis

Um die Meldungen aus der Steuerung auf dem HMI-Gerät vollständig anzuzeigen, muss Sie die Option "Automatische Aktualisierung" unter "Runtime-Einstellungen Meldungen > Steuerungsmeldungen" für die jeweilige Verbindung aktiviert sein. Weitere Informationen zu vollständigen Meldungen finden Sie unter "Vollständige Meldung von der Steuerung an das Bediengerät senden (Seite 987)".

Kriterienanalyse im Meldesystem

Sie visualisieren die Meldungen zur Kriterienanalyse in folgenden Schritten:

- Sie aktivieren Erstwerterfassung in den Eigenschaften des ProDiag- oder GRAPH-Funktionsbausteins des Anwenderprogramms
- Sie aktivieren die Optionen zum Erweitern der Meldetexte in den Runtime-Einstellungen des HMI-Geräts
- Sie projektieren eine Meldeanzeige und eine Kriterienanalyse-Anzeige und verbinden beide Objekte miteinander

Meldungen erweitern

1. Öffnen Sie den Editor "Runtime-Einstellungen" des HMI-Geräts.
2. Klicken Sie "Runtime Einstellungen > Meldungen > Kriterienanalyse".
3. Wählen Sie unter "Kriterienanalyse > Text erweitern" welche Meldungen Sie erweitern möchten.
4. Aktivieren Sie die Zusatzinformationen, die in der Meldung zum Meldetext hinzugefügt werden, z. B. Symbolname und Adresse des ersten fehlerhaften Operanden.

Ergebnis

Im Fehlerfall sehen Sie in der Meldeanzeige nicht nur den Meldetext, sondern auch den Operanden, der die Fehlermeldung ausgelöst hat. Sie selektieren die entsprechende Meldung und sehen in der Kriterienanalyse-Anzeige die fehlerhaften Operanden, die durch Kriterienanalyse ermittelt wurden.

Mit Steuerungen kommunizieren 13

13.1 WinCC Engineering V15.1 - Kommunikation

Mit Steuerungen kommunizieren

Eine detaillierte Beschreibung zur Kommunikation mit Steuerungen finden Sie im Handbuch "WinCC Engineering V15.1 - Kommunikation".

Übergreifende Funktionen nutzen 14

14.1 Mit dem Bediengeräte-Assistent arbeiten

14.1.1 Grundlagen zum Bediengeräte-Assistenten

Einleitung

Wenn Sie in Ihrem Projekt ein neues Bediengerät anlegen, startet automatisch der Bediengeräte-Assistent.

Bediengeräte-Assistent

Der Bediengeräte-Assistent führt Sie Schritt für Schritt durch die einzelnen Dialoge und unterstützt Sie beim Anlegen eines Geräts. Mit dem Bediengeräte-Assistenten legen Sie grundsätzliche Einstellungen für ihr Bediengerät fest, z. B. die Bilddarstellung oder die Verbindung zu Ihrer PLC.

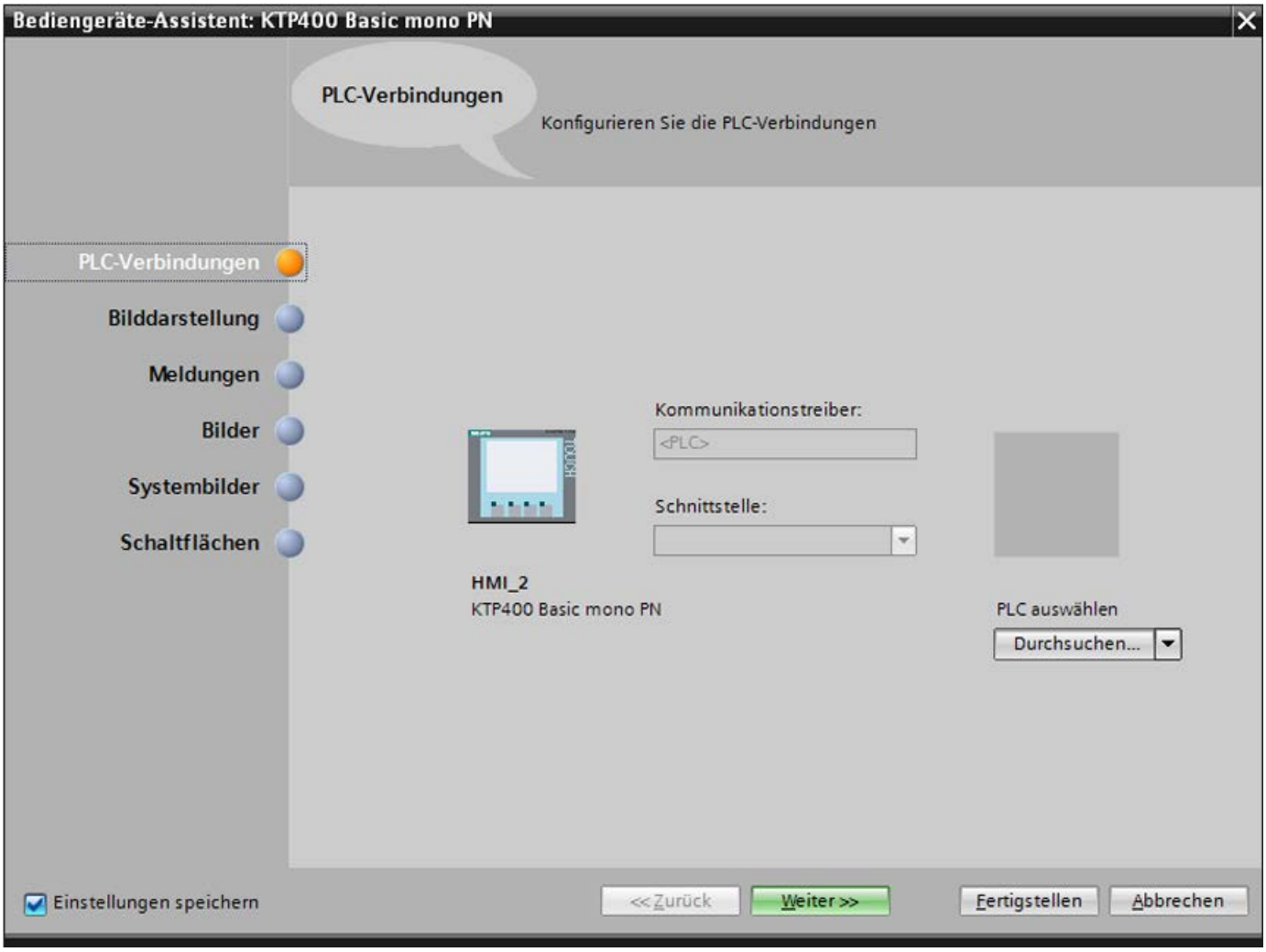

Hinweis

Wenn Sie mit dem Bediengeräte-Assistenten ein Gerät mit einem Farb-Display anlegen, kann es vorkommen, dass die Grafiken der Navigationsschaltflächen in schwarz-weiß angezeigt werden. Dieser Fehler tritt aber nur auf, wenn das neue Gerät mit dem gleichen Namen angelegt wird, den ein zwischenzeitlich gelöschtes Gerät mit monochromem Display besessen hat.

Sie können den Fehler vermeiden, indem Sie immer, wenn Sie ein Gerät aus dem Projekt löschen, auch die zugehörigen Grafiken in der Grafiksammlung löschen.

Hinweis

Mehrsprachigkeit im Bediengeräte-Assistenten

Wenn Sie mit dem Bediengerät-Assistenten die Bildnavigation mehrsprachig anlegen, verwendet WinCC für benutzerdefinierte Bezeichnungen in allen Projektsprachen die Oberflächensprache.

Wenn der Zeichensatz der Oberflächensprache in einer Projektsprache nicht unterstützt wird, dann werden diese Bezeichnungen in dieser Projektsprache nicht angezeigt, z. B. bei der Oberflächensprache Chinesisch und der Projektsprache Englisch.

Hinweis

Schnittstellentyp der Verbindung

Wenn möglich, erstellt der Bediengerät-Assistent die Verbindung zwischen Bediengerät und Steuerung mit dem Standard-Schnittstellentyp des Bediengeräts.

Prüfen Sie nach der Projektierung mit dem Bediengerät-Assistenten die Verbindung im Editor "Geräte & Netze".

Siehe auch

Grundlagen zum Bediengeräte-Assistenten (Seite 2014)

Allgemeine Informationen zu Querverweisen (Seite 2086)

Sprachbegriffe in WinCC (Seite 2095)

14.1.2 Grundlagen zum Bediengeräte-Assistenten

Einleitung

Der Bediengeräte-Assistent führt Sie Schritt für Schritt durch die einzelnen Dialoge und unterstützt Sie beim Anlegen eines Geräts. Mit dem Bediengeräte-Assistenten legen Sie grundsätzliche Einstellungen für ihr Bediengerät fest, z. B. die Bilddarstellung oder die Verbindung zu Ihrer PLC.

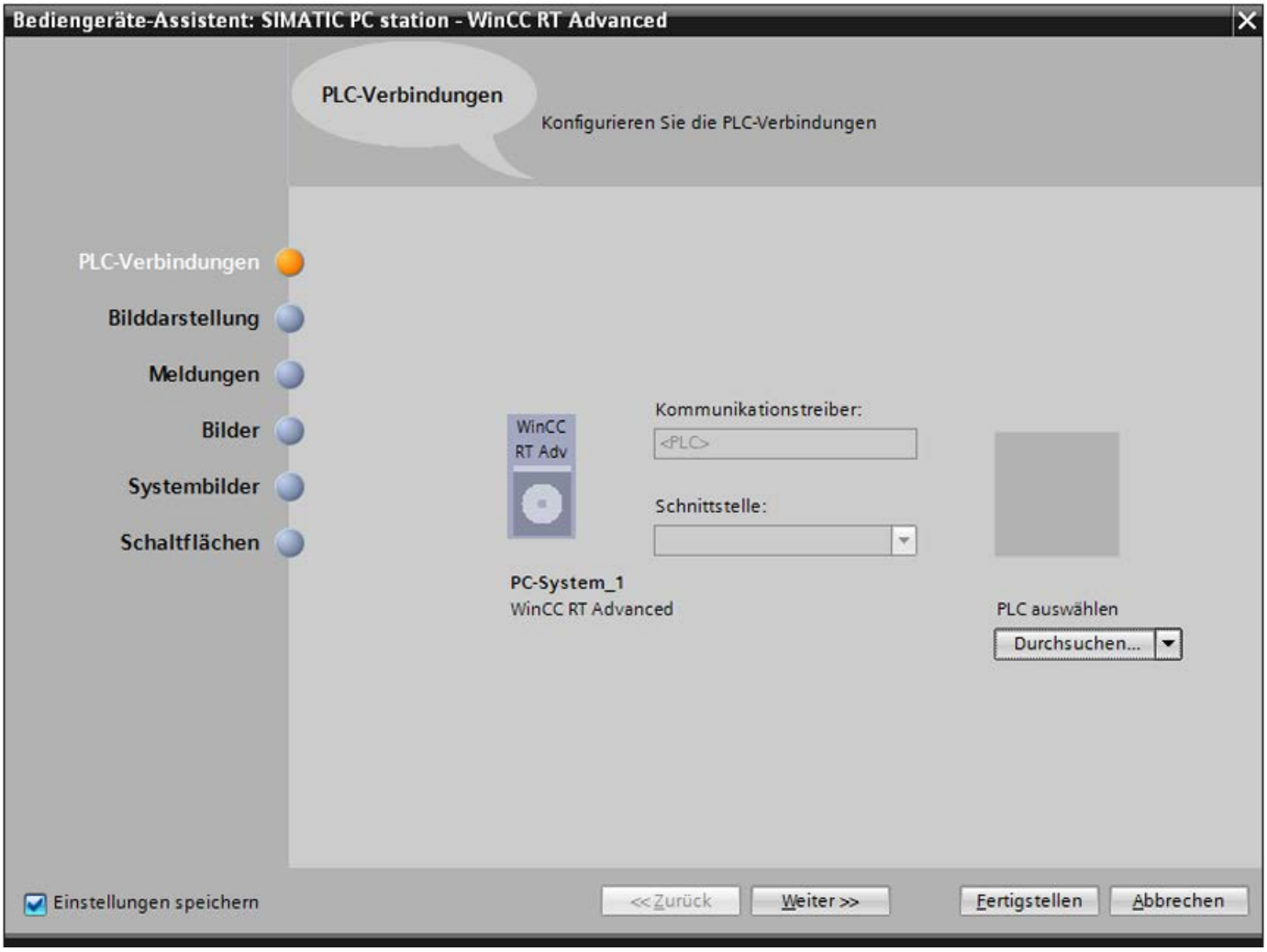

Bediengeräte-Assistenten öffnen

1. Fügen Sie ein neues Gerät in Ihrem Projekt ein, z. B. RT Advanced.
2. Klicken Sie in der Projektnavigation auf das Gerät.
3. Wählen Sie im Kontextmenü "Bediengeräte-Assistent starten". Der "Bediengeräte-Assistent wird geöffnet.

Hinweis

Wenn Sie nach dem Einfügen des Geräts Änderungen machen z. B. ein neues Bild hinzufügen, lässt sich der Bediengeräte-Assistent nicht mehr öffnen.

Hinweis

Wenn Sie mit dem Bediengeräte-Assistenten ein Gerät mit einem Farb-Display anlegen, kann es vorkommen, dass die Grafiken der Navigationsschaltflächen in schwarz-weiß angezeigt werden. Dieser Fehler tritt aber nur auf, wenn das neue Gerät mit dem gleichen Namen angelegt wird, den ein zwischenzeitlich gelöschtes Gerät mit monochromem Display besessen hat.

Sie können den Fehler vermeiden, indem Sie immer, wenn Sie ein Gerät aus dem Projekt löschen, auch die zugehörigen Grafiken in der Grafiksammlung löschen.

Hinweis

Geräteabhängigkeit - Anführungszeichen in Projektnamen

WinCC Professional Projekte, deren Namen Anführungszeichen enthalten, werden nicht übersetzt.

Wenn Sie ein Bediengerät für WinCC RT Professional anlegen, verwenden Sie keine Anführungszeichen innerhalb des Projektnamens.

Hinweis

Mehrsprachigkeit im Bediengeräte-Assistenten

Wenn Sie mit dem Bediengerät-Assistenten die Bildnavigation mehrsprachig anlegen, verwendet WinCC für benutzerdefinierte Bezeichnungen in allen Projektsprachen die Oberflächensprache.

Wenn der Zeichensatz der Oberflächensprache in einer Projektsprache nicht unterstützt wird, dann werden diese Bezeichnungen in dieser Projektsprache nicht angezeigt, z. B. bei der Oberflächensprache Chinesisch und der Projektsprache Englisch.

Siehe auch

Grundlagen zum Bediengeräte-Assistenten (Seite 2011)

Dateien über den Editor "HMI Dateien" laden (Seite 2193)

14.2 Arbeiten mit Bibliotheken

14.2.1 Grundlagen zu Bibliotheken

Einleitung

In den Bibliotheken legen Sie Objekte zentral ab, die Sie häufig benötigen. Ein in der Bibliothek abgelegtes Objekt müssen Sie nur einmal konfigurieren. Danach können Sie es beliebig oft wiederverwenden. Bibliotheksobjekte erweitern die Menge der verfügbaren Bildobjekte und steigern durch Mehrfachverwendung von Fertigobjekten die Effektivität beim Projektieren.

Mit WinCC werden umfangreiche Bibliotheken mitgeliefert z. B. zu den Gebieten "Motoren" oder "Ventile". Sie können aber auch Bibliotheksobjekte selbst definieren.

Bibliotheken verwalten Sie In der Task Card "Bibliotheken" oder in der Bibliotheksansicht. Folgende Bibliotheken stehen Ihnen zur Verfügung:

- Projektbibliothek
- Globale Bibliotheken

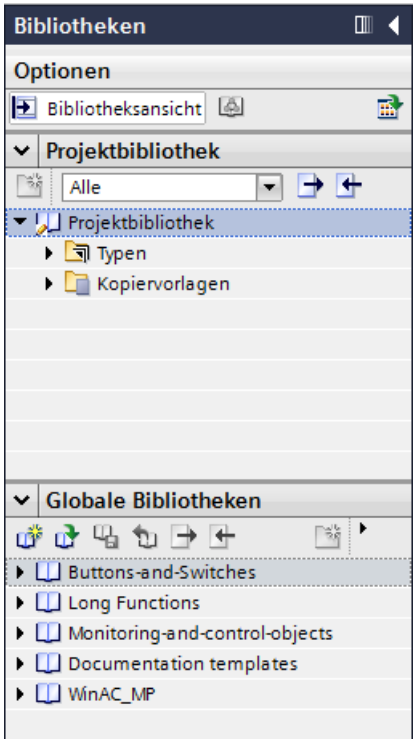

Hinweis

Eine Symbolbibliothek befindet sich in der Task Card "Werkzeuge" in der Palette "Grafiken".

Projektbibliothek

In jedem Projekt gibt es genau eine Projektbibliothek. Die Objekte der Projektbibliothek werden mit den Projektdaten gespeichert und stehen nur für das Projekt zur Verfügung, in dem die Bibliothek angelegt wurde. Wenn das Projekt auf einen anderen PC verschoben wird, wird eine darin angelegte Projektbibliothek mitverschoben.

Um das Bibliotheksobjekt der Projektbibliothek auch in anderen Projekten nutzen zu können, verschieben oder kopieren Sie das Objekt in eine Globale Bibliothek.

Globale Bibliotheken

Eine Globale Bibliothek wird unabhängig von den Projektdaten in einer eigenen Datei mit der Dateinamenerweiterung ".alxx" gespeichert, dabei steht "xx" für die aktuelle WinCC Versionsnummer.

Ein Projekt kann auf mehrere Globale Bibliotheken zugreifen. Dieselbe Globale Bibliothek kann von mehreren Projekten gleichzeitig verwendet werden.

Wenn ein Bibliotheksobjekt von einem Projekt geändert wurde, wird die Bibliothek danach auch in den Projekten geändert, die diese Bibliothek geöffnet haben.

Bibliotheksobjekte

Eine Bibliothek kann alle WinCC-Objekte enthalten. Beispiele:

- Ganzes Bediengerät
- Bilder
- Anzeige- und Bedienobjekte einschließlich Variablen und Funktionen
- Grafiken
- Variablen
- Meldungen
- Text- und Grafiklisten
- Bildbausteine
- Anwenderdatentypen
- Stile und Formatvorlagen

Siehe auch

Grundlagen zu Bildern (Seite 35)

Bibliotheksobjekte anzeigen (Seite 2035)

Objekt in einer Bibliothek hinterlegen (Seite 2038)

Bibliotheksobjekt einfügen (Seite 2039)

Globale Bibliothek erstellen (Seite 2026)

Globale Bibliothek speichern (Seite 2027)

Globale Bibliothek öffnen (Seite 2028)

Kopiervorlagen und Typen (Seite 2020)

14.2.2 Übersicht über die Bibliotheksansicht

Funktion der Bibliotheksansicht

Die Bibliotheksansicht vereinigt die Funktionalitäten der Task Card "Bibliotheken" und dem Übersichtsfenster. In der Bibliotheksansicht zeigen Sie die Elemente einer Bibliothek in unterschiedlichen Ansichten an und sehen z. B. in der Detailansicht weitere Eigenschaften der einzelnen Elemente. Außerdem bearbeiten und versionieren Sie Typen in der Bibliotheksansicht.

Aufbau der Bibliotheksansicht

Das folgende Bild zeigt die Komponenten der Bibliotheksansicht:

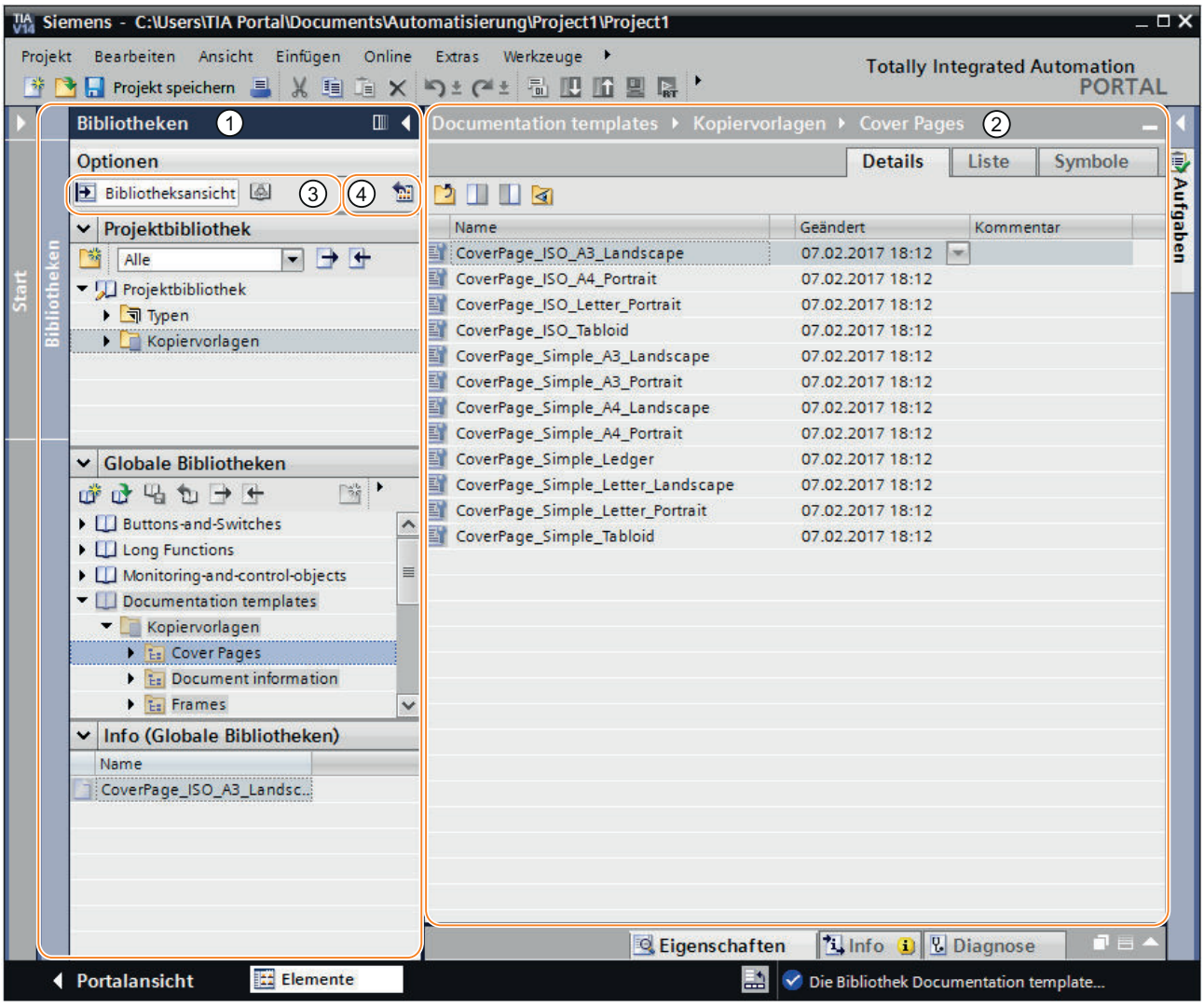

1 Bibliotheksnavigation

2 Bibliotheksübersicht

3 Schaltfläche "Bibliotheksansicht"

4 Schaltfläche "Bibliotheksübersicht öffnen oder schließen"

Bibliotheksnavigation

Die Bibliotheksnavigation gleicht bis auf kleinere Abweichungen der Task Card "Bibliotheken". Im Gegensatz zur Task Card gibt es keine Palette "Elemente", da die Elemente in der Bibliotheksübersicht angezeigt werden. Zudem können Sie in der Bibliotheksnavigation die Bibliotheksansicht schließen oder die Bibliotheksübersicht öffnen oder schließen.

Bibliotheksübersicht

Die Bibliotheksübersicht entspricht dem Übersichtsfenster und zeigt die Elemente des aktuell in der Bibliotheksnavigation selektierten Objekts an. Sie können die Elemente in drei unterschiedlichen Ansichten anzeigen lassen. Außerdem können Sie in der Bibliotheksübersicht z. B. folgende Aktionen durchführen:

- Elemente kopieren
- Elemente verschieben
- Typen versionieren
- Bildbausteine und HMI Anwenderdatentypen bearbeiten
- Instanzen von Typen bearbeiten

14.2.3 Kopiervorlagen und Typen

Einleitung

Sowohl die "Projektbibliothek" als auch die "Globale Bibliothek" enthalten die beiden Ordner "Kopiervorlagen" und "Typen". Sie können die Bibliotheksobjekte entweder als Kopiervorlage oder als Typ erzeugen bzw. verwenden.

Kopiervorlagen

Um voneinander unabhängige Kopien des Bibliotheksobjekts zu erzeugen, verwenden Sie Kopiervorlagen.

Typen

Von Objekten des Ordners "Typen" erstellen Sie Instanzen und verwenden diese in Ihrem Projekt. Die Instanzen sind an ihren jeweiligen Typ gebunden.

Verwalten der Bibliotheksobjekte

Sie kopieren und verschieben Bibliotheksobjekte in andere Bibliothek. Sie kopieren Kopiervorlagen nur in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen". Auch Typen können Sie nur in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" einfügen.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

Zustand von Typ-Versionen (Seite 2040)

14.2.4 Bibliotheken in WinCC

Einleitung

Mit WinCC werden umfangreiche Bibliotheken mitgeliefert. Thematisch in Ordnern gegliedert, finden Sie vorgefertigte grafische Objekte, die Sie in Bildern für das Bedienen und Beobachten Ihrer Anlage verwenden können.

Globale Bibliothek "Buttons and Switches"

Die Bibliotheken "Buttons and Switches" bietet Ihnen eine große Auswahl an Schaltern und Schaltflächen.

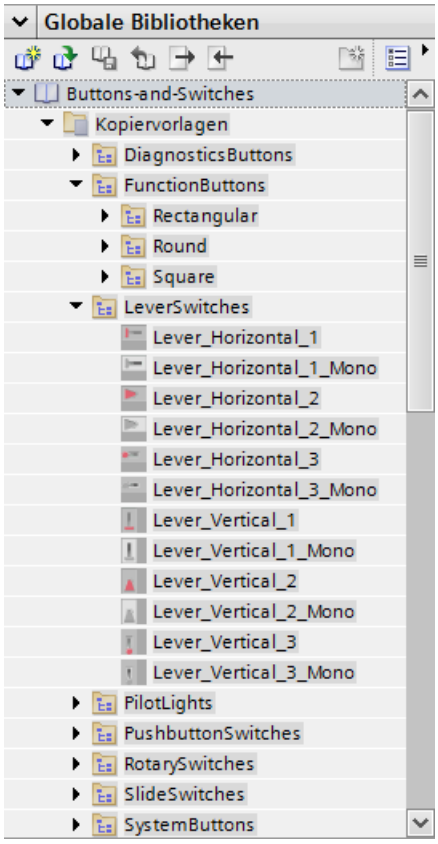

Die Ordner gliedern Schalter bzw. Schaltflächen in Kategorien. Im Ordner "DiagnosticsButtons" finden Sie z. B. das Objekt "System-Diagnoseindikator". Das Objekt "System-Diagnoseindikator" verwenden Sie für die Systemdiagnose in Ihrer Anlage.

Hinweis

Die Objekte im Ordner "DiagnosticsButtons" können Sie nur auf Comfort Panels nutzen.

Die Objekte mit der Bezeichnung "Switch" im Objektnamen bzw. im zugehörigen Ordnernamen können Sie nicht in Runtime-Professional einsetzen.

Globale Bibliothek "Monitoring and Control objects"

Die Bibliothek "Monitoring and Control objects" bietet Ihnen komplexere Bedien- und Anzeigeobjekte in mehreren Designs sowie dazu passende Kontrolllampen, Schaltflächen und Schalter.

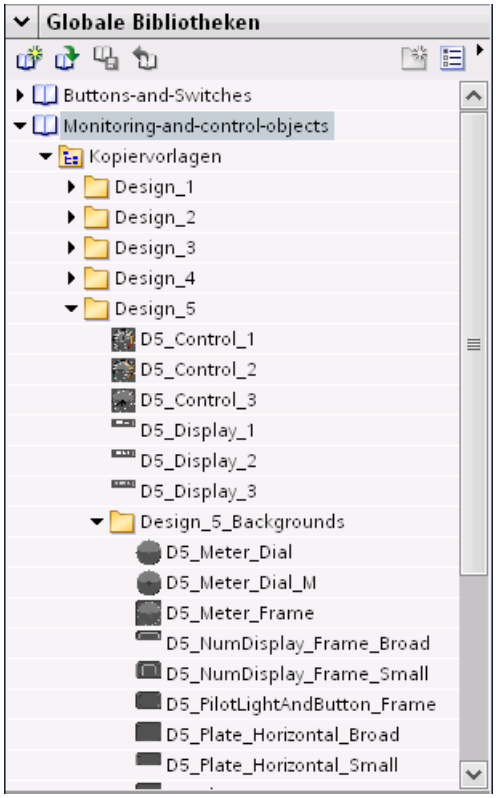

Zusätzlich sind zu den Designs passende Grafikanzeigen in den Ordnern "Design_Backgrounds" abgelegt, die zur kundenspezifischen Erweiterung des Bibliotheksumfangs als Objekthintergründe genutzt werden können.

Hinweis

Die Objekte mit der Bezeichnung "Switch" im Objektnamen können Sie nicht in Runtime Professional einsetzen. Das Gleiche gilt für das Objekt "D5_Display_3" mit dem darin enthaltenen Datum/Uhrzeit-Feld.

14.2.5 Bibliotheken verwalten

14.2.5.1 Übersicht über die Bibliotheksverwaltung

Funktion der Bibliotheksverwaltung

Kopiervorlagen und Typen mit Abhängigkeiten zu anderen Bibliothekselementen unterliegen einigen Funktionseinschränkungen. Sie lassen sich beispielsweise nicht löschen, so lange noch Abhängigkeiten bestehen. Auf diese Weise wird verhindert, dass andere Bibliothekselemente unbrauchbar werden. Mit der Bibliotheksverwaltung identifizieren Sie Abhängigkeiten und verschaffen sich einen Überblick über den Arbeitsfortschritt.

Die Bibliotheksverwaltung bietet folgende Funktionen:

- Darstellung der Zusammenhänge von Typen und Kopiervorlagen
  Wird ein Typ in anderen Typen oder Kopiervorlagen referenziert, dann werden die Zusammenhänge in der Bibliotheksverwaltung angezeigt. Ebenso wird angezeigt, welche Bibliothekselemente ein Typ oder eine Kopiervorlage referenziert.
- Anzeige der Verwendungsstellen von Typen im Projekt
- Filtern aller Typen, die sich im Zustand "im Test" oder "in Bearbeitung" befinden

Aufbau der Bibliotheksverwaltung

Das Bild zeigt Bibliotheksverwaltung, die aus den folgenden Komponenten besteht:

- Funktionsleiste der Bibliotheksverwaltung
- Bereich "Typen"
- Bereich "Verwendung"

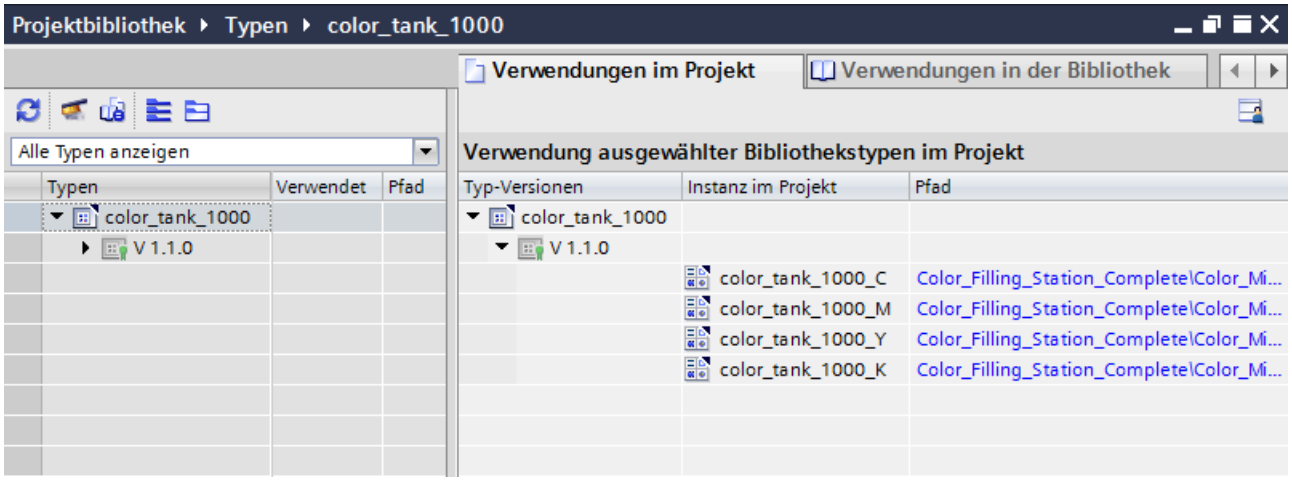

Funktionsleiste der Bibliotheksverwaltung

Über die Funktionsleiste der Bibliotheksverwaltung können Sie folgende Aktionen durchführen:

- Ansicht aktualisieren
  Falls das Projekt geändert wurde, aktualisieren Sie die Ansicht der Bibliotheksverwaltung.
- Bibliothek aufräumen
  Sie können die Projektbibliothek und globale Bibliotheken aufräumen. Mit dem Aufräumen einer Bibliothek werden alle Typen und Typ-Versionen gelöscht, die mit keiner Instanz im Projekt verküpft sind.
- Projekt harmonisieren
  Durch das Harmonisieren eines Projekts werden die Namen und die Pfadstrukuren von Typverwendungen im Projekt an die entsprechenden Namen und Pfadstukturen der Typen innerhalb einer Bibliothek angepasst.

Bereich "Typen"

Im Bereich "Typen" werden die Inhalte des Ordners angezeigt, den Sie in der Bibliotheksansicht selektiert haben. Über die Schaltflächen in der Funktionsleiste des Bereichs "Typen" können Sie alle Typen auf- oder zuklappen. Außerdem können Sie mit der Klappliste "Filter" die Ansicht filtern und ausschließlich alle Typen im Zustand "im Test" oder "in Bearbeitung" anzeigen. Zu jedem Typen werden die Typen angezeigt, die von ihm referenziert werden.

Bereich "Verwendungen"

Im Bereich "Verwendungen" sehen Sie eine Übersicht über die Verwendungsstellen der selektierten Typen und Kopiervorlagen. Der Bereich "Verwendungen" ist in zwei Register unterteilt:

- Register "Verwendungen im Projekt"
  Im Register "Verwendungen im Projekt" zeigen Sie die Instanzen von Typ-Versionen sowie ihre jeweilige Verwendungsstelle im Projekt an. Wenn Sie eine Instanz selektieren, können Sie im Inspektorfenster die Querverweise der Instanz im Projekt anzeigen.
- Register "Verwendungen in der Bibliothek"
  Im Register "Verwendungen in der Bibliothek" zeigen Sie alle Stellen innerhalb der Bibliothek an, an denen ein Typ oder eine Kopiervorlage verwendet werden.

14.2.5.2 Bibliotheksverwaltung öffnen

Vorgehen

Um die Bibliotheksverwaltung zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksansicht.
2. Selektieren Sie einen Typen oder einen beliebigen Ordner, der Typen enthält.
3. Wählen Sie im Kontextmenü den Befehl "Bibliotheksverwaltung".

Ergebnis

Die Bibliotheksverwaltung wird geöffnet und die Typen werden mit ihren Versionen angezeigt.

14.2.5.3 Typen in der Bibliotheksverwaltung filtern

Einleitung

Mit der Filterfunktion in der Bibliotheksverwaltung haben Sie die Möglichkeit, die angezeigten typen einzugrenzen. Folgende Filter stehen Ihnen zur Verfügung:

- Anzeige aller Typen, die eine Version im Zustand „in Bearbeitung“ besitzen
- Anzeige aller Typen, die keine Instanzen im Projekt besitzen
- Anzeige aller Typen, die mehr als eine Version besitzen
- Anzeige aller freigegebenen Typen

Voraussetzung

Mindestens ein Typ ist angelegt.

Filtern nach allen Typen mit ausstehenden Änderungen

1. Selektieren Sie in der Projektbibliothek den Ordner "Typen".
2. Wählen Sie im Kontextmenü des Ordners "Typen" den Eintrag "Bibliotheksverwaltung“. Die Bibliotheksverwaltung wird geöffnet.
3. Wählen Sie in der Klappliste "Filter" den Eintrag "Typen mit ausstehenden Änderungen". Im Bereich "Typen" werden ausschließlich die Typen angezeigt, die sich im Zustand "in Bearbeitung“ befinden.

Filtern nach allen Typen, die keine Instanzen im Projekt besitzen

1. Selektieren Sie in der Projektbibliothek den Ordner "Typen".
2. Wählen Sie im Kontextmenü des Ordners "Typen" den Eintrag "Bibliotheksverwaltung“. Die Bibliotheksverwaltung wird geöffnet.
3. Wählen Sie in der Klappliste "Filter" den Eintrag "Typen ohne Verwendung im Projekt". Im Bereich "Typen" werden ausschließlich die Typen angezeigt, die keine Instanzen im Projekt besitzen.

Filtern nach Typen mit mehreren Versionen

1. Selektieren Sie in der Projektbibliothek den Ordner "Typen".
2. Wählen Sie im Kontextmenü des Ordners "Typen" den Eintrag "Bibliotheksverwaltung“. Die Bibliotheksverwaltung wird geöffnet.
3. Wählen Sie in der Klappliste "Filter" den Eintrag "Typen mit mehreren Versionen". Im Bereich "Typen" werden ausschließlich die Typen angezeigt, die mehr als eine Version haben.

Filtern nach allen freiegebenen Typen

1. Selektieren Sie in der Projektbibliothek den Ordner "Typen".
2. Wählen Sie im Kontextmenü des Ordners "Typen" den Eintrag "Bibliotheksverwaltung“. Die Bibliotheksverwaltung wird geöffnet.
3. Wählen Sie in der Klappliste "Filter" den Eintrag "Freiegebene Typen". Im Bereich "Typen" werden ausschließlich die Typen angezeigt, die nur freigegebene Versionen haben.

Filtern nach Typen mit neuen Versionen für das Hochrüsten anderer Typen

1. Selektieren Sie in der Projektbibliothek den Ordner "Typen".
2. Wählen Sie im Kontextmenü des Ordners "Typen" den Eintrag "Bibliotheksverwaltung“. Die Bibliotheksverwaltung wird geöffnet.
3. Wählen Sie in der Klappliste "Filter" den Eintrag "Typen mit neuen Versionen für das Hochrüsten anderer Typen". Im Bereich "Typen" werden ausschließlich die Typen angezeigt, die potenziell als Quelle fürs Hochrüsten anderer Typen verwendet werden können.

Siehe auch

Bibliotheksverwaltung öffnen (Seite 2024)

Bildbaustein-Typ erzeugen (Seite 278)

14.2.5.4 Globale Bibliothek erstellen

Einleitung

In Bibliotheken hinterlegen Sie konfigurierte Objekte, die Sie bei der Projektierung mehrfach verwenden wollen. Um Objekte in mehreren Projekten zu nutzen, legen Sie eine globale Bibliothek an.

Voraussetzung

- Ein Projekt ist geöffnet.
- Die Task Card "Bibliotheken" oder die "Bibliotheksansicht" ist geöffnet.

Vorgehen

1. Klicken Sie unter "Globale Bibliotheken" auf das Symbol .
   Der Dialog "Neue globale Bibliothek erstellen" wird geöffnet.

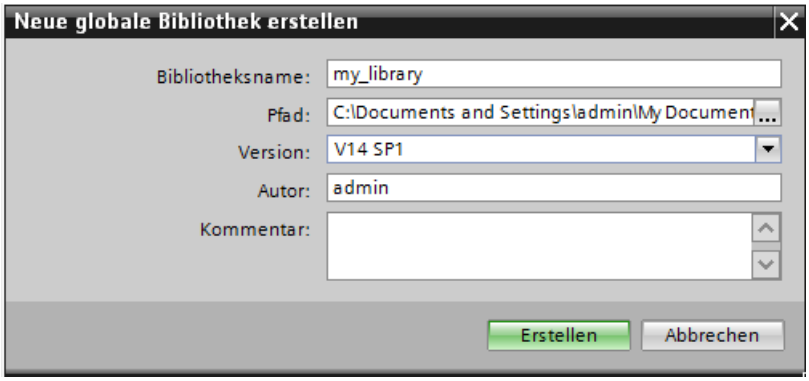

2. Geben Sie einen Name ein.
3. Wählen Sie den Pfad, wo die neue Bibliothek gespeichert werden soll.
4. Klicken Sie auf "Erstellen".

Ergebnis

Die neue Bibliothek wird in der Palette "Globale Bibliotheken" angezeigt. Die globale Bibliothek enthält die Ordner "Typen", "Kopiervorlagen" und "Gemeinsame Daten". Unter den "Gemeinsamen Daten" finden Sie die Protokolle zu der globalen Bibliothek.

Am Speicherort der globalen Bibliothek wird im Dateisystem ein Ordner mit dem Namen der globalen Bibliothek angelegt. Die eigentliche Bibliotheksdatei erhält die Dateinamenerweiterung ".alxx", dabei steht "xx" für die aktuelle WinCC Versionsnummer.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.5.5 Globale Bibliothek speichern

Einleitung

Eine Globale Bibliothek wird als separate Datei auf Ihrer Festplatte gespeichert. Die Datei enthält die Objekte der Globalen Bibliothek einschließlich der referenzierten Objekte. Z. B. wird die Referenz einer Variablen, die an ein E/A-Feld projektiert wurde, ebenfalls in der Bibliothek gespeichert.

Wenn Sie Ihr Projekt oder WinCC schließen, ohne zu speichern, fordert WinCC Sie auf, die Globalen Bibliotheken zu speichern. Sie können auch während der Projektierung die Globale Bibliothek speichern, ohne dafür das gesamte Projekt zu speichern.

Voraussetzung

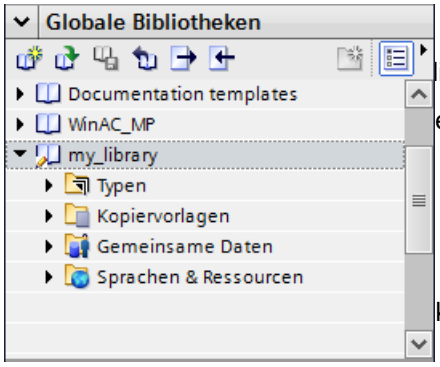

- liothek ist geöffnet.
- e Bibliotheksansicht ist geöffnet.
-

Vorgehen

1. k, die Sie speichern möchten.
2. Klicken Sie in in der Palette "Globale Bibliothek" auf das Symbol .

Alternativ verwenden Sie im Kontextmenü "Bibliothek speichern".

Wenn Sie die Globale Bibliothek in einem anderen Ordner speichern wollen, wählen Sie im Kontextmenü "Speichern unter". Wählen Sie den Pfad, in dem Sie die neue Bibliothek speichern wollen, und geben Sie einen Dateinamen an.

Ergebnis

Die Globalen Bibliotheken werden unter dem aktuellen oder dem neu vergebenen Dateinamen gespeichert.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.5.6 Globale Bibliothek öffnen

Einleitung

In WinCC sind Globale Bibliotheken als separate Dateien abgelegt. Sie können eine Globale Bibliothek in jedem Projekt verwenden.

Voraussetzung

- Eine Globale Bibliothek ist gespeichert.
- Ein Projekt ist geöffnet.
- Die Task Card "Bibliotheken" oder die Bibliotheksansicht ist geöffnet.

Vorgehen

1. Klicken Sie in der Palette "Globale Bibliotheken" auf das Symbol .
   Der Dialog "Globale Bibliothek öffnen" wird geöffnet.
2. Wählen Sie den Pfad, in dem die Bibliothek gespeichert ist.
3. Klicken Sie auf "Öffnen".

Hinweis

Um vom mehreren Projekten aus Zugriff auf eine globale Bibliothek zu haben, öffnen Sie die globale Bibliothek schreibgeschützt. Wenn eine globale Bibliothek nicht schreibgeschützt geöffnet ist, wird der Zugriff von den Projekten blockiert.

Ergebnis

WinCC zeigt die geöffnete Globale Bibliothek in der Palette "Globale Bibliotheken" an.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.5.7 Protokolle zu globalen Bibliotheken anzeigen

Bei der Aktualisierung von globalen Bibliotheken werden Protokolle angelegt, in denen alle durchgeführten Änderungen an der globalen Bibliothek aufgelistet sind. Die Protokolle werden zusammen mit der globalen Bibliothek gespeichert und sind immer verfügbar, nachdem Sie die globale Bibliothek geöffnet haben.

Vorgehen

Um die Protokolle einer globalen Bibliotheken aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek in der Task-Card "Bibliotheken" oder in der Bibliotheksansicht.
2. Öffnen Sie den unterlagerten Ordner "Gemeinsame Daten > Protokolle".
3. Doppelklicken Sie auf das gewünschte Protokoll.
   Das Protokoll wird im Arbeitsbereich geöffnet.

14.2.5.8 Projekt mit den Inhalten einer Bibliothek aktualisieren

Einleitung

Nachdem Sie mehrere Typen in der Projektbibliothek bearbeitet haben, aktualisieren Sie alle Instanzen im Projekt auf die neuste Version der Typen.

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Einzelne Ordner innerhalb einer Bibliothek
- Einzelne Typen

Voraussetzung

Die Task Card "Bibliothken" oder die Bibliotheksansicht ist geöffnet.

Vorgehen

1. Selektieren Sie einen Ordner innerhalb der Projektbibliothek oder einzelne Typen.
2. Wählen Sie im Kontextmenü "Aktualisieren > Projekt...".
   Ein Dialog wird geöffnet.
3. Wählen Sie entweder das gesamte Projekt oder einzelne Geräte für die Aktualisierung aus.
4. Um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen, aktivieren Sie das Optionskästchen "Alle nicht verwendeten Versionen aus der Projektbibliothek löschen".
5. Bestätigen Sie mit "Ok".

Ergebnis

Im Projekt werden alle Instanzen der Typen auf die neuste Version der gewählten Typen in der Projektbibliothek.

In der Projektnavigation unter "Gemeinsame Daten" finden Sie ein Protokoll des Aktualisierungsvorgangs.

14.2.5.9 Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren

Folgende Möglichkeiten zur Aktualisierung von Bibliotheken stehen Ihnene zur Verfügung:

- Aktualisieren einer globalen Bibliothek mit Typen aus einer anderen globalen Bibliothek
- Aktualisieren der Projektbibliothek mit Typen aus einer globalen Bibliothek

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Einzelne Ordner innerhalb einer Bibliothek
- Einzelne Typen

Voraussetzung

Um eine globale Bibliothek zu aktualisieren, öffnen Sie die Bibliothek mit Schreibrechten.

Voraussetzung

Die Task Card "Bibliothken" oder die Bibliotheksansicht ist geöffnet.

Vorgehen

Um eine Bibliothek mit Inhalten einer anderen Bibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Ordner innerhalb der Bibliothek oder einzelne Typen.
2. Klicken Sie mit der rechten Maustauste auf die Quelle und wählen Sie im Kontextmenü den Befehl "Aktualisieren > Bibliothek...".
   Der Dialog "Bibliothek aktualisieren" wird geöffnet.
3. Wählen Sie aus, welche Art von Bibliothek Sie aktualisieren möchten:
   – Wählen Sie "Projektbibliothek aktualisieren", um die Projektbibliothek mit Typen aus einer globalen Bibliothek zu aktualisieren.
   – Wählen Sie "Globale Bibliothek aktualisieren", wenn Sie eine globale Bibliothek aktualisieren möchten.
4. Optional: Wählen Sie aus der Klappliste die globale Bibliothek, die Sie aktualisieren möchten.
5. Um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen, aktivieren Sie das Optionskästchen "Alle nicht verwendeten Versionen aus der Projektbibliothek löschen".
6. Bestätigen Sie mit "OK".

Ergebnis

- Typen, die in der Zielbibliothek noch nicht vorhanden sind, werden dort mit all ihren Versionen ergänzt. Typen, die bereits in der Zielbibliothek vorhanden sind, werden um die neueren Versionen ergänzt.
  Wenn in der Zielbibliothek bereits eine neuere Version eines Typen vorhanden ist, wird die neueste Version aus der Quellbibliothek trotzdem kopiert und automatisch eine neuere Versionsnummer vergeben.
- Für den Aktualisierungsvorgang wird ein Protokoll angelegt, das alle durchgeführten Änderungen an der Zielbibliothek enthält.
  Wenn Sie die Projektbibliothek aktualisiert haben, finden Sie das Protokoll in der Projektnavigation unter "Gemeinsame Daten > Protokolle".
  Wenn Sie eine globale Bibliothek aktualisiert haben, finden Sie das Protokoll, der globalen Bibliothek unterlagert, im Ordner "Gemeinsame Daten > Protokolle".

14.2.5.10 Typen einer Bibliothek hochrüsten

Beschreibung

In Ihrem Projekt sind Objekte enthalten, die andere Objekte, z. B. Anwenderdatentypen verwenden oder referenzieren. Sie haben eine Projektbibliothek erstellt und die verwendeten Bibliothekstypen in der Projektbibliothek gespeichert. Die Typen haben Sie außerhalb des Projekts weiter bearbeitet. Die neuen Versionen der bearbeiteten Typen integrieren Sie wieder in die Projektbibliothek.

Sie haben die Möglichkeit, mithilfe der Funktion "Hochrüsten von Typen" alle referenzierten Typen, die von einem geänderten Typ abhängig sind, zu erkennen und diese automatisch im

gesamten Projekt hochzurüsten. Die hochgerüsteten Typen werden in der Projektbibliothek in den Zustand "in Bearbeitung" oder "im Test" versetzt. Somit sehen Sie auf einen Blick, welche Typen geändert wurden, um diese zu testen und frei zu geben.

Voraussetzung

- Die Projektbibliothek wurde mit den Typen der Globalen Bibliothek aktualisiert
- Die neuen Typ-Versionen wurden in die bestehende Projektbibliothek integriert

Vorgehen

1. Öffnen Sie die Bibliotheksverwaltung.
2. Wählen Sie im Bereich "Typen" aus der Klappliste "Typen mit neuen Versionen für das Hochrüsten anderer Typen" aus.
   Alle Typen, für die eine neue Version eines Typs in der Projektbibliothek existiert werden angezeigt.
3. Klicken Sie den Ordner "Typen" an und wählen Sie im Kontextmenü "Hochrüstbare Typen angeigen" aus.
   Alle referenzierten Typen werden im Register "Hochrüsten von Typen" angezeigt.
4. Wählen Sie unter "Hochzurüstende Typen" eine der Optionen aus:
   – "Nur mit Testumgebung", um alle Typen anzuzeigen, die eine Instanz in der Testumgebung referenzieren.
   – "Kompatibel mit gewählter Testumgebung", um alle Typen anzuzeigen, die eine Instanz der Testumgebung referenzieren können. Die Instanz wurde aber noch nicht erstellt.
5. Wählen Sie eine Testumgebung aus.
   Alle Typen, die hochgerüstet werden müssen, werden angezeigt.

   Hinweis

   Wenn einer der hochzurüstenden Typen den Status "Fehler" hat, können die Typen nicht hochgerüstet werden. Beheben Sie zuerst den Fehler und starten Sie den Vorgang erneut.

6. Klicken Sie auf "Starten".
   Nach erfolgreichem Hochrüsten wird der Dialog "Hochrüsten von Typen" geöffnet.
7. Überprüfen Sie die hochgerüsteten Typen und geben Sie diese anschließend frei.

Ergebnis

Alle Typen in Ihrem Projekt wurden hochgerüstet und referenzieren die aktuellen Typ-Versionen.

Siehe auch

Projekt mit den Inhalten einer Bibliothek aktualisieren (Seite 2029)

Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren (Seite 2030)

14.2.5.11 Bibliothekstexte exportieren und importieren

Einleitung

Sie haben die Möglichkeit, die Texte der Bibliotheksobjekte in eine .xlsx Datei zu exportieren, um sie z. B. in MS Excel zu bearbeiten oder zur Übersetzung auszuleiten.

Sie exportieren und importieren Texte folgender Objekte in der Bibliothek:

- Einzelne Bibliothekstypen und Kopiervorlagen
- Mehrere Bibliothekstypen und Kopiervorlagen
- Alle Bibliotheksobjekte der Projektbibliothek oder einer Globalen Bibliothek

Nach der Bearbeitung oder externen Übersetzung importieren Sie die Texte in das TIA Portal.

Beim Importieren der Texte werden alle Texte aus der Importdatei für die gesamte Bibliothek importiert, auch wenn Sie nur ein Bibliotheksobjekt selektiert haben. Die Zielsprachen der Importdatei müssen im Projekt aktiviert sein.

Beim Importieren in eine Kopiervorlage werden die Texte der Vorlage in der Bibliothek mit den neuen Texten aus der Import-Datei überschrieben. Beim Importieren der Texte in einen Bibliothekstyp wird in der Bibliothek die aktuellste Version mit den neuen Texten aus der Import-Datei überschrieben. Wenn in einer Projektbibliothek eine Version eines Typs noch nicht freigegeben ist, können für die gesamte Projektbibliothek keine Texte importiert werden.

Hinweis

Einschränkungen für den Textimport

Die Texte, die zu folgenden Bibliotheksobjekten gehören, können nicht importiert werden:

- Typ-Instanzen, die in einer Kopiervorlage enthalten sind
- Bibliothekstypen, deren Versionen noch nicht freigegeben sind und den Zustand "In Bearbeitung" oder "In Test" haben
- Schreibgeschützte Globale Bibliothek

Quellsprache und Zielsprache definieren

Im Export-Dialog legen Sie die Quellsprache und die Zielsprache für den Export der Texte fest. Die Auswahl der verfügbaren Sprachen hängt von den definierten Projektsprachen ab.

Bei der Globalen Bibliothek hängt die Auswahl der verfügbaren Quell- und Zielsprache von den Sprachen ab, die der Ersteller der Bibliothek festgelegt hat. Um die verfügbaren Sprachen der Globalen Bibliothek zu sehen, doppelklicken Sie im Projektordner "Sprachen & Ressourcen" der jeweiligen Bibliothek auf den Eintrag "Bibliothekssprachen".

Texte exportieren

Um die Texte eines einzelnen oder mehrerer Bibliotheksobjekte zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen sie die Bibliothek.
2. Wählen Sie in der Bibliothek das gewünschte Bibliotheksobjekt.

3. Wählen Sie im Kontextmenü des Objekts den Befehl "Bibliothekstexte exportieren".
   Alternativ klicken Sie in der Funktionsleiste auf die Schaltfläche "Bibliothekstexte exportieren".
   Der Dialog "Exportieren" wird geöffnet.
4. Wählen Sie im Dialog die Quellsprache und die Zielsprache für den Export.
5. Geben Sie den Namen und den Pfad für die Export-Datei ein.

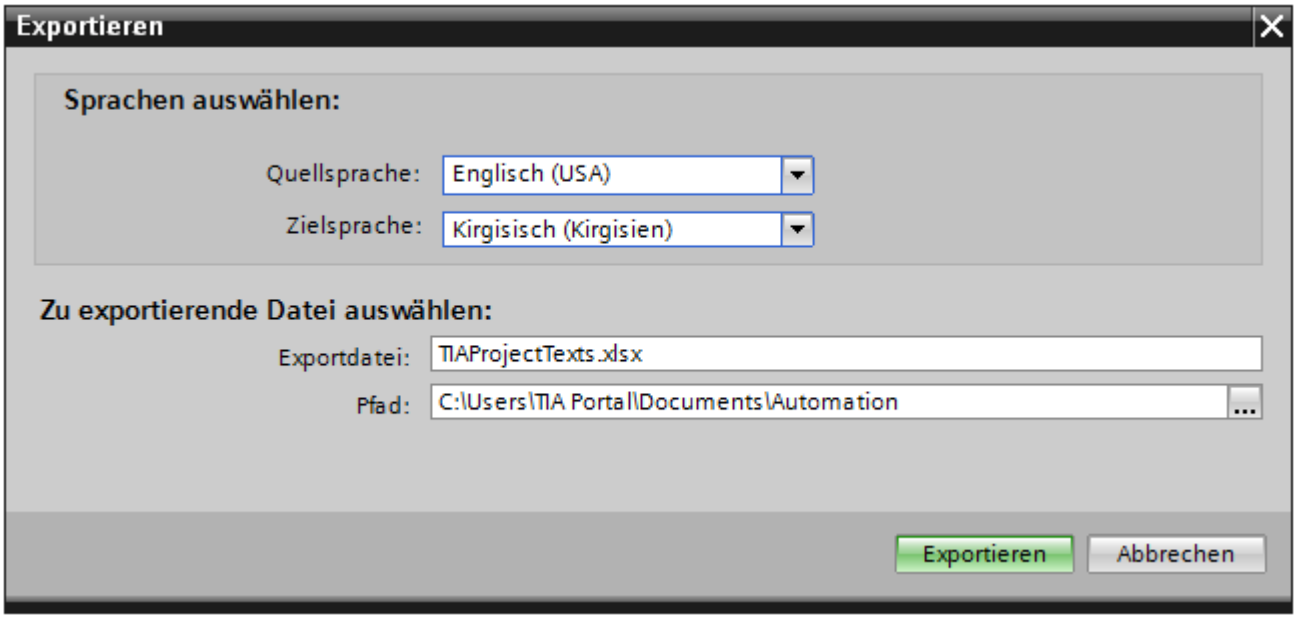

6. Klicken Sie auf "Exportieren".
   Nachdem der Export erfolgreich abgeschlossen wurde, wird unter dem angegebenen Pfad die Export-Datei abgelegt.

Texte importieren

Um die Texte nach der Bearbeitung oder Übersetzung wieder in das TIA Portal zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen sie die Projektbibliothek oder die Globale Bibliothek.
2. Wählen Sie im Kontextmenü des Objekts den Befehl "Bibliothekstexte importieren".
   Alternativ klicken Sie in der Funktionsleiste auf die Schaltfläche "Bibliothekstexte importieren".
   Der Dialog "Importieren" wird geöffnet.
3. Wählen Sie im Feld "Zu importierende Datei auswählen" den Pfad und den Namen der Importdatei aus.
   Wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten, aktivieren Sie das Optionskästchen "Quellsprache importieren".
4. Klicken Sie auf "Importieren".

14.2.6 Objekte in einer Bibliothek verwalten

14.2.6.1 Bibliotheksobjekte anzeigen

Einleitung

Die Bibliotheken werden als Dateiordner in der entsprechenden Palette angezeigt. Die in der Bibliothek enthaltenen Elemente werden im Dateiordner und in der Palette "Elemente" angezeigt.

Voraussetzung

- Mindestens ein Bibliothekobjekt ist in einer Bibliothek angelegt.
- Die Task Card "Bibliotheken" ist geöffnet.

Vorgehen

1. Selektieren Sie die Bibliothek in der entsprechenden Palette, deren Bibliotheksobjekte Sie anzeigen wollen

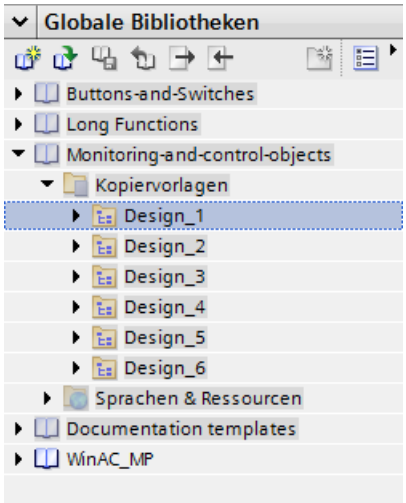

2. Klicken Sie auf das Symbol .
   Die enthaltenen Bibliotheksobjekte werden in der Palette "Elemente" angezeigt.

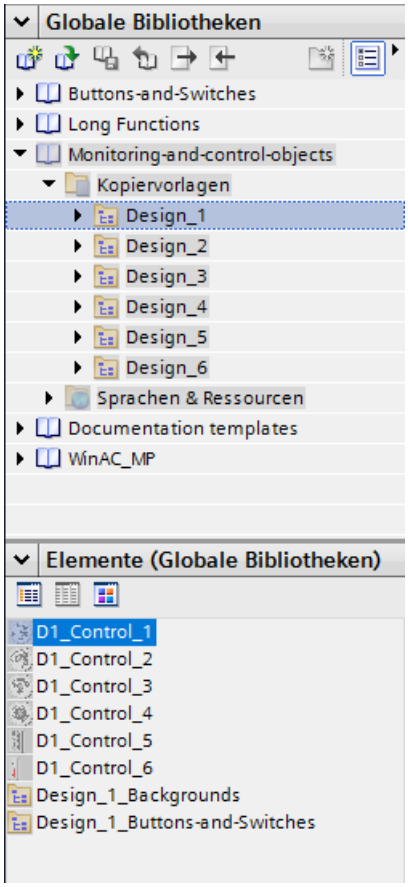

3. Klicken Sie auf eines der folgenden Symbole:

Symbol	Beschreibung
	Elementansicht im Detailmodus
	Elementsansicht im Listenmodus
	Elementansicht im Übersichtsmodus mit Symbolen

Wenn der Bibliothek über Mehrfachauswahl mehreren Objekten zugeordnet werden, wird in der Palette "Elemente" nur eines der Objekte angezeigt. Die einzelnen Bestandteile dieses Elements werden in der Palette "Teile" angezeigt.

Teile der Bibliotheksobjekte anzeigen

1. Selektieren Sie die Bibliothek in der entsprechenden Palette, aus der Sie die Bestandteile eines Elements sehen wollen.
2. Klicken Sie auf das Symbol .
3. Die enthaltenen Bibliothekobjekte werden in der Palette "Elemente" angezeigt.
4. Selektieren Sie das Element.
   In der Palette "Teile" sehen Sie die Objekte, aus denen das Element besteht.

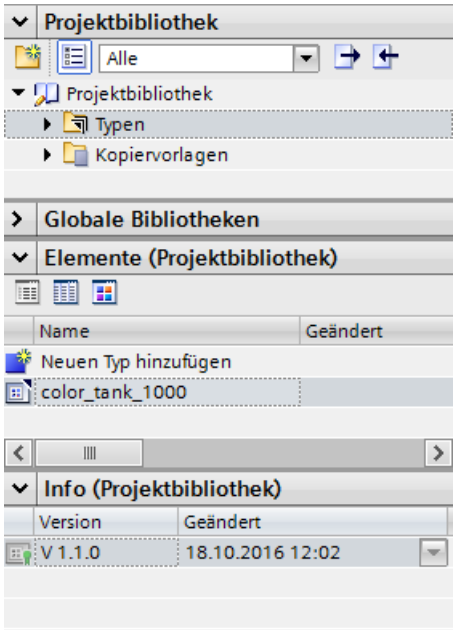

Ergebnis

Die Bibliothekobjekte werden entsprechend der vorgenommenen Konfiguration angezeigt. Die Bestandteile der Bildbausteine werden angezeigt.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.6.2 Objekt in einer Bibliothek hinterlegen

Einleitung

In Bibliotheken können Sie alle WinCC-Objekte, z. B. Bilder, Variablen, Grafikobjekte oder Meldungen hinterlegen. Sie ziehen dazu per Drag&Drop aus dem Arbeitsbereich, der Projektfenster oder der Detailansicht das entsprechende Objekt in die Bibliothek. Wenn Sie die Bibliothek in Kategorien eingeteilt haben, können Sie ein Objekt auch direkt in eine bestimmte Kategorie einfügen.

Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Ein Bildobjekt ist im Arbeitsbereich des Bildes angelegt.
- Die angelegten Bibliotheken werden angezeigt.

Vorgehen

1. Selektieren Sie das Objekt im Arbeitsbereich des Editors "Bilder".
2. Ziehen Sie das Objekt per Drag&Drop aus dem Arbeitsbereich in die gewünschte Bibliothek. Dem Mauszeiger verwandelt sich in ein Fadenkreuz mit angehängtem Objektsymbol.

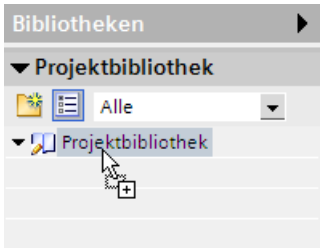

Ergebnis

Das Objekt ist in der Bibliothek hinterlegt und kann beim Projektieren mehrfach verwendet werden.

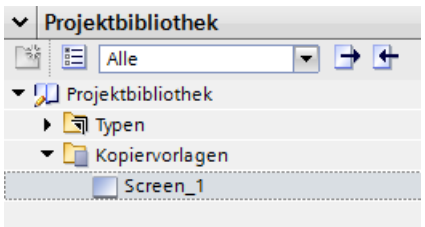

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.6.3 Bibliotheksobjekt einfügen

Einleitung

Beim Einfügen eines Bibliotheksobjekts wird der Objektname standardmäßig aus der Bezeichnung des Objekttyps und einer fortlaufenden Nummer zusammengesetzt.

Wenn das eingefügte Objekt bereits vorhanden ist, haben Sie die Möglichkeit, das Objekt zu ersetzten oder unter einem neuen Namen zu speichern.

Bibliotheksobjekte, die vom Bediengerät nicht unterstützt werden, können nicht eingefügt werden.

Hinweis

Wenn Sie ein Bild mit verknüpfter Vorlage aus der Bibliothek einfügen, wird auch die Vorlage eingefügt. Eine eventuell vorhandene, passende Vorlage wird nicht genutzt.

Voraussetzung

- Die Task Card "Bibliotheken" ist geöffnet.
- Der Editor, in den Sie das Bibliotheksobjekt einfügen wollen, ist geöffnet.

Vorgehen

1. Selektieren Sie ein Bibliotheksobjekt aus der Bibliothek.
2. Ziehen Sie das Bibliotheksobjekt per Drag&Drop an die Stelle des Arbeitsbereichs, an der Sie das Objekt einfügen wollen.
   Das Bibliotheksobjekt wird eingefügt.

Ergebnis

Wenn das Objekt im Ordner "Kopiervorlagen" enthalten war, haben Sie eine unabhängige Kopie des Bibliotheksobjekts im Editor eingefügt.

Wenn das Objekt im Ordner "Typen" enthalten war, haben Sie eine Instanz des Bibliotheksobjekts im Editor eingefügt.

Siehe auch

Grundlagen zu Bibliotheken (Seite 2016)

14.2.7 Typen und ihre Versionen verwenden

14.2.7.1 Zustand von Typ-Versionen

Einleitung

Abhängig von der Verwendungsstelle hat die Version eines Typen verschiedene Zustände.

Freigegebene Version

Der Zustand "freigebenene Version" ist für alle Typen verfügbar, unabhängig von der Verwendungsstelle.

Wenn Sie eine freigegebene Version bearbeiten möchten, müssen Sie zunächst eine neue Testversion oder einen Arbeitsstand erzeugen.

Freigegebene Typ-Versionen von Skripten und Bildern lassen sich an ihrer Instanz öffnen und betrachten.

Version "in Bearbeitung"

Der Zustand "in Bearbeitung" ist für folgende Typen möglich:

- Bildbausteine
- Anwenderdatentypen (HMI UDT)
- Stile
- Formatvorlagen

Wenn Sie einen neuen Typen erstellen, oder eine neue Version eines freigebenen Typen, wird der Typ in den Zustand "in Bearbeitung" gesetzt.

Typen mit dem Zustand "in Bearbeitung" lassen sich in der Bibliotheksansicht bearbeiten, ohne dass ein Bezug zu einer Instanz im Projekt vorhanden sein muss. Die Kompatibiltiät des Typen wird bei der Freigabe mit einer Konsistenzprüfung geprüft.

Version "im Test"

Nur Versionen von Skripten und HMI BIldern haben den Zustand "im Test".

Wenn Sie eine neue Version eines Typen erstellen, wird der Typ in den Zustand "im Test" gesetzt.

Eine Version, die sich im "im Test" befindet, ist mit einer Instanz im Projekt verknüpft. Es ist immer nur möglich, eine einzige Version je Typ in den Zustand "im Test" zu versetzen.

Eine Version im Test darf nur mit einer einzigen Instanz im Projekt verknüpft sein. Daher ist es nicht möglich eine Instanz in die Zwischenablage zu kopieren, sie zu duplizieren oder einen weiteren Typen aus der Instanz zu erstellen, solange diese sich im Zustand "im Test" befindet.

Siehe auch

Skript als Typ erzeugen (Seite 2041)
Bild als Typ erzeugen (Seite 2042)
Neue Version eines Typen erstellen (Seite 2043)
Kopiervorlagen und Typen (Seite 2020)
Bildbaustein-Typ erzeugen (Seite 278)

14.2.7.2 Skript als Typ erzeugen

Voraussetzung

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt und geöffnet.
- Die Projektnavigation ist geöffnet.
- Die Task Card „Bibliotheken“ ist geöffnet.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Editor „Skripte“.
2. Erstellen Sie ein neues Skript.
3. Selektieren Sie das Skript in der Projektnavigation.
4. Ziehen Sie das Skript per Drag & Drop in eine Bibliothek in der Task Card "Bibliohtken". Ein Dialog wird geöffnet.
5. Geben Sie einen Namen ein.
6. Vergeben Sie einen Kommentar.

Ergebnis

Sie haben von einem Skript einen Typ-Version in der Bibliothek erzeugt. Der erzeugte Typ wird als freigegebene Version in der Bibliothek gespeichert. Im Projekt wird eine Instanz des Typen verwendet.

Um das Skript zu ändern, erstellen Sie eine neue Version des Skripts.

Siehe auch

Zustand von Typ-Versionen (Seite 2040)

14.2.7.3 Bild als Typ erzeugen

Voraussetzung

Ein Projekt ist geöffnet.

Ein Bediengerät ist angelegt und geöffnet.

Die Projektnavigation ist geöffnet.

Die Task Card „Bibliotheken“ ist geöffnet.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Editor „Bilder“.
2. Erstellen Sie ein neues Bild.
3. Selektieren Sie das Bild in der Projektnavigation.
4. Ziehen Sie das Bild per Drag & Drop in eine Bibliothek in der Task Card "Bibliohtken". Ein Dialog wird geöffnet.
5. Geben Sie einen Namen ein.
6. Vergeben Sie einen Kommentar

Ergebnis

Sie haben von einem Bild einen Typen in der Bibliothek erzeugt.

Der erzeugte Typ wird als freigegebene Version in der Bibliothek gespeichert. Im Projekt wird eine Instanz des Typen verwendet.

Um das Bild zu ändern, erstellen Sie eine neue Version des Bilds.

Siehe auch

Zustand von Typ-Versionen (Seite 2040)

14.2.7.4 Stil als Typ erzeugen

Einleitung

Um einen neuen Stil zu definieren, fügen Sie in der Projektbibliothek einen neuen Typen hinzu.

Neuen Stil hinzufügen

1. Öffnen Sie auf die Task Card "Bibliotheken".
2. Wählen Sie im Kontextmenü der Projektbibliothek unter "Typen" den Befehl "Neuen Typ hinzufügen". Ein Dialog wird geöffnet.
3. Wählen Sie "HMI Stil".

Ergebnis

Der neue Stil wird erzeugt und unter dem gewählten Namen in der Projektbibliothek angezeigt. Der HMI Stil-Typ erhält den Status "In Bearbeitung" und die Version 0.0.1.

14.2.7.5 Formatvorlage als Typ erzeugen

Einleitung

Um eine neue Formatvorlage zu definieren, fügen Sie in der Projektbibliothek einen neuen Typen hinzu.

Neue Formatvorlage hinzufügen

1. Öffnen Sie auf die Task Card "Bibliotheken".
2. Wählen Sie im Kontextmenü der Projektbibliothek unter "Typen" den Befehl "Neuen Typ hinzufügen". Ein Dialog wird geöffnet.
3. Wählen Sie "HMI Formatvorlage".
4. Vergeben Sie der neuen Formatvorlage einen aussagekräftigen Namen.
5. Wählen Sie die Kategorie der Formatvorlage aus der Liste.

Ergebnis

Die neue Formatvorlage wird erzeugt und unter dem gewählten Namen in der Projektbibliothek angezeigt.

Der HMI Formatvorlagen-Typ erhält den Status "In Bearbeitung" und die Version 0.0.1.

14.2.7.6 Neue Version eines Typen erstellen

Prinzip

Wenn Sie eine neue Version eines Typen erstellen, bestimmt die Verwendungsstelle des Typen welchen Zustand die neu erstellte Version erhält.

Voraussetzung

Die Task Card "Bibliotheken" ist geöffnet.

Ein Typ ist erstellt und freigegeben.

Vorgehen

1. Selektieren Sie den freigegebenen Typen.
2. Wählen Sie im Kontextmenü "Typ editieren".

Ergebnis für Typen von Bildbausteinen, Anwenderdaten und Stilen

Eine neue Version des Typen wird erstellt.

Die Version hat den Zustand "in Bearbeitung". Die Bibliotheksansicht wird geöffnet.

Ergebnis für Typen von Skripten und Bildern

Ein Dialog wird geöffnet.

Nachdem Sie die Einstellungen im Dialog gewählt haben, wird die Version in den Zustand "in Test" gesetzt. Die verwendete Instanz im Projekt wird in den Zustand "in Test" gesetzt. Die Bibliotheksansicht wird geöffnet.

Siehe auch

Zustand von Typ-Versionen (Seite 2040)

14.3 Projektdaten importieren und exportieren

14.3.1 Datenaustausch zwischen Projekten und externen Anwendungen

Einleitung

WinCC bietet Ihnen die Möglichkeit, Projektdaten zwischen den Projekten auszutauschen oder in externe Anwendungen zu kopieren.

Export und Import zwischen Projekten

Sie können folgende Projektdaten aus einem Projekt exportieren und in ein anderes Projekt importieren.

- Rezepturdatensätze
- Meldungen
- Variablen
- Textlisten
- Projekttexte

Durch den Export und Import reduzieren Sie den Arbeitsaufwand. Anstatt neue Datensätze anzulegen, verwenden Sie Daten, die Sie bereits in früheren Projekten erstellt haben.

Bearbeitung der Exportdatei

Je nach Editor stehen Ihnen folgende Dateiformate für den Export bzw. Import zur Verfügung:

- *.xlsx für Meldungen, Variablen, Projekttexte und Textlisten
- *.csv für Rezepturdatensätze

Zur Bearbeitung der Importdatei verwenden Sie z. B. Excel.

Dateiformat XLSX

Das Format XLSX ist ein Dateiformat für Excel-Tabellen auf der Grundlage des Open-XML-Formats. XLSX-Dateien sind für Microsoft Excel 2007 optimiert.

In der XLXS-Datei können Sie die Spalten beliebig sortieren.

Dateiformat CSV

CSV steht für Comma Separated Value. In diesem Format werden die Tabellenspalten, die den Namen und den Wert des Eintrags enthalten, durch Semikola getrennt. Jede Tabellenzeile wird durch eine Zeilenumbruch abgeschlossen. Zur Bearbeitung können Sie die CSV-Datei auch in Excel öffnen.

Import von Projektdaten

Bei einem Import der Projektdaten werden die Objekte im Projekt erzeugt.

Beim Import von Dateien wird die Syntax der Importdatei geprüft. Die Korrektheit der importierten Werte oder Abhängigkeiten zwischen den importierten Werten werden nicht geprüft.

Wenn die importierten Daten Fehler aufweisen, so werden diese beim Übersetzen des Projekts gemeldet.

Kopieren im Excel-Format

In allen Tabelleneditoren können Sie die Inhalte im Excel-Format in den Zwischenspeicher Ihres PCs kopieren. Anschließend fügen Sie die Projektdaten im Excel-Format direkt in eine beliebige Anwendung außerhalb des TIA Portals ein. Sie nutzen dafür den entsprechenden Befehl im Kontextmenü des Arbeitsbereichs:

- Wenn Sie den Befehl im Kontextmenü des Zeilenkopfs wählen, wird die ganze Zeile in den Zwischenspeicher kopiert.
- Wenn Sie den Befehl im Kontextmenü einer Zelle wählen, wird nur der Zelleninhalt in den Zwischenspeicher kopiert.
- Wenn Sie mehrere Zeilen markieren und den Befehl wählen, werden immer alle markierten Daten in den Zwischenspeicher kopiert.

Dieser Datenaustausch ist nur als Export möglich.

Siehe auch

Grundlagen zum Bediengeräte-Assistenten (Seite 2011)

14.3.2 Rezepturdatensätze importieren und exportieren

14.3.2.1 Rezepturdatensätze exportieren

Einleitung

Für den Export von Datensätzen aus Rezepturen steht Ihnen in WinCC eine Exportfunktion zur Verfügung.

Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Rezepturen angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Rezepturen exportieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, deren Datensätze Sie exportieren wollen.
2. Klicken Sie auf das Symbol .
   Der Dialog "Export" wird geöffnet.

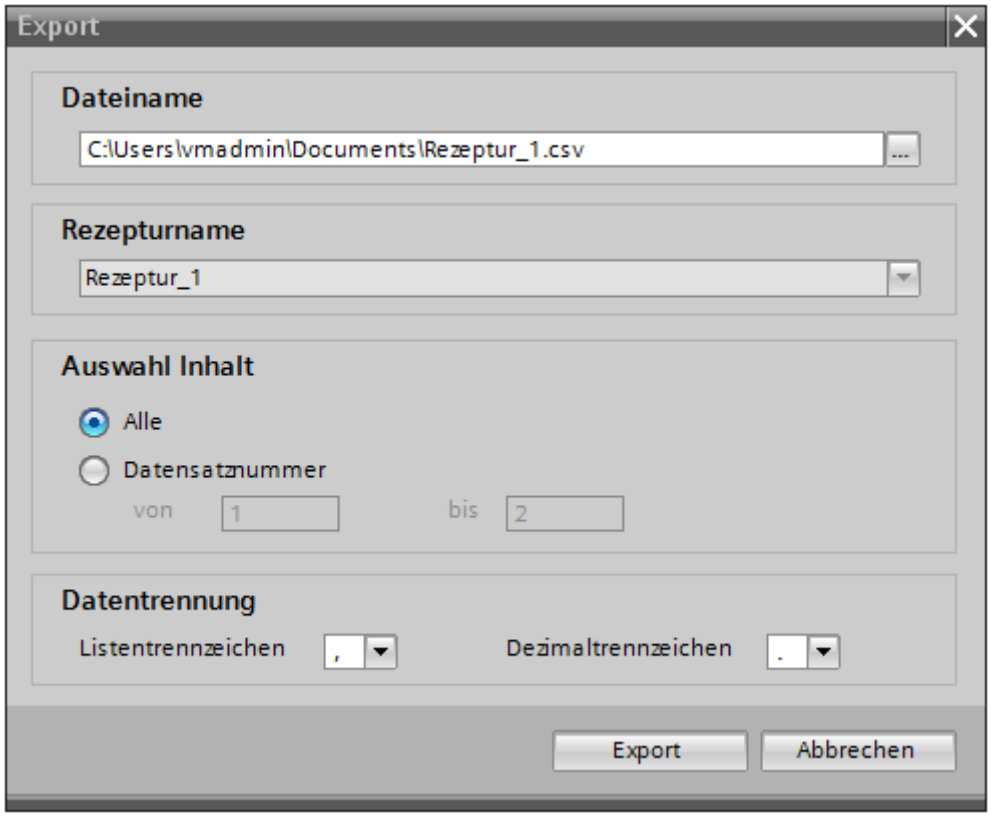

   Unter "Rezepturauswahl" wird die markierte Rezeptur angezeigt.
3. Legen Sie unter "Auswahl Inhalt" fest, ob Sie alle oder nur ausgewählte Datensätze exportieren.
4. Legen Sie unter "Dateiauswahl" fest, in welche Datei die Rezepturdaten gespeichert werden.
5. Legen Sie unter "Datentrennung" Listentrennzeichen und Dezimaltrennzeichen fest.
6. Klicken Sie auf "Export".
   Der Export wird gestartet.

Ergebnis

Die exportierten Daten wurden in eine CSV-Datei geschrieben. Die CSV-Datei wird im angegebenen Verzeichnis abgelegt.

Siehe auch

Rezepturdatensätze importieren (Seite 2048)

14.3.2.2 Rezepturdatensätze importieren

Einleitung

Rezepturen werden über ihre Namen identifiziert. Die Rezepturname müssen daher eindeutig sein. Zur Kontrolle der Datenstruktur in der Importdatei öffnen Sie die Datei in einem einfachen Texteditor.

Beim Import wählen Sie, ob bereits bestehende Datensätze mit gleichen Namen überschrieben werden.

Voraussetzungen

- Eine CSV-Datei mit mindestens einer Rezeptur ist angelegt.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "Rezepturen" mit mindestens einer Rezeptur ist geöffnet.

Rezeptur importieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, in die Sie Datensätze importieren wollen.
2. Klicken Sie auf das Symbol .
   Der Dialog "Import" wird geöffnet.

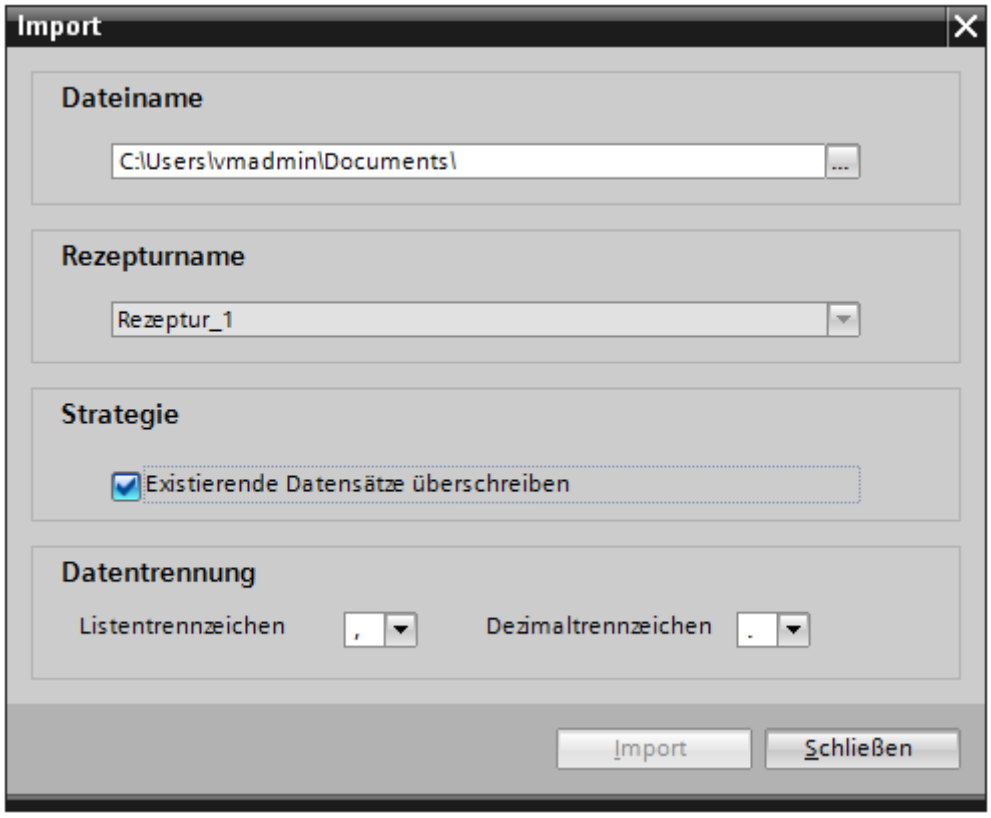

   Unter "Auswahl Rezeptur" wird die markierte Rezeptur angezeigt.
3. Wählen Sie unter "Dateiauswahl" die Datei, die Sie importieren wollen.
4. Legen Sie unter "Strategie" fest, ob Sie vorhandene gleichnamige Datensätze überschreiben wollen.
5. Wählen Sie unter "Datentrennung" das Listentrennzeichen und das Dezimaltrennzeichen, das in der CSV-Datei verwendet wurde.
6. Klicken Sie auf "Import".
   Der Import wird gestartet.

Ergebnis

Die Datensätze werden in der markierten Rezeptur angelegt. Abhängig von der Einstellung unter "Strategie" werden vorhandene Datensätze mit gleichnamigen Datensätzen aus der CSV-Datei überschrieben.

Wenn Sie die Option "Existierende Datensätze überschreiben" deaktivieren, werden gleichnamigen Datensätzen aus der CSV-Datei zusätzlich importiert.

Siehe auch

Rezepturdatensätze exportieren (Seite 2046)

14.3.2.3 Rezepturdatensätze exportieren

Einleitung

Für den Export von Datensätzen aus Rezepturen steht Ihnen in WinCC eine Exportfunktion zur Verfügung.

Voraussetzungen

- Das WinCC-Projekt für den Export ist geöffnet.
- In dem Projekt sind Rezepturen angelegt.
- Der Editor "Rezepturen" ist geöffnet.

Rezepturen exportieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, deren Datensätze Sie exportieren wollen.
2. Klicken Sie auf das Symbol .
   Der Dialog "Export" wird geöffnet.

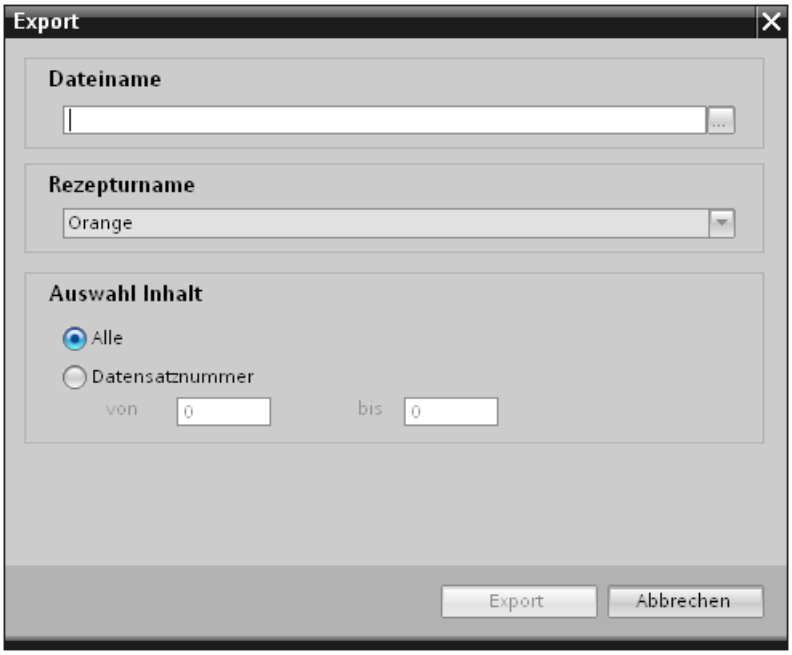

   Unter "Rezepturname" wird die markierte Rezeptur angezeigt.
3. Legen Sie unter "Dateiname" fest, in welche Datei die Rezepturdaten gespeichert werden.

4. Legen Sie unter "Auswahl Inhalt" fest, ob Sie alle oder nur ausgewählte Datensätze exportieren.
5. Klicken Sie auf "Export".
   Der Export wird gestartet.

Ergebnis

Die exportierten Daten wurden in eine CSV-Datei geschrieben. Die CSV-Datei wird im angegebenen Verzeichnis abgelegt.

14.3.2.4 Rezepturdatensätze importieren

Einleitung

Rezepturen werden über ihre Namen identifiziert. Die Rezepturname müssen daher eindeutig sein. Zur Kontrolle der Datenstruktur in der Importdatei öffnen Sie die Datei in einem einfachen Texteditor.

Beim Import wählen Sie, ob bereits bestehende Datensätze mit gleichen Namen überschrieben werden.

Voraussetzungen

- Eine CSV-Datei mit mindestens einer Rezeptur ist angelegt.
- Das WinCC-Projekt für den Import ist geöffnet.
- Der Editor "Rezepturen" mit mindestens einer Rezeptur ist geöffnet.

Rezeptur importieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, in die Sie Datensätze importieren wollen.
2. Klicken Sie auf das Symbol .
   Der Dialog "Import" wird geöffnet.

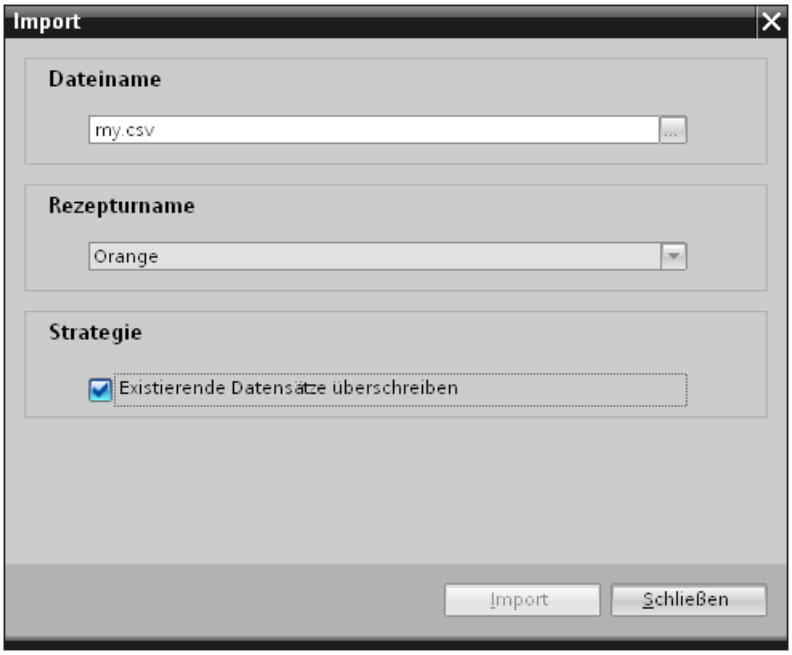

   Unter "Rezepturname" wird die markierte Rezeptur angezeigt.
3. Wählen Sie unter "Dateiauname" die Datei, die Sie importieren wollen.
4. Legen Sie unter "Strategie" fest, ob Sie vorhandene gleichnamige Datensätze überschreiben wollen.
5. Klicken Sie auf "Import".
   Der Import wird gestartet.

Ergebnis

Die Datensätze werden in der markierten Rezeptur angelegt. Abhängig von der Einstellung unter "Strategie" werden vorhandene Datensätze mit Datensätzen aus der CSV-Datei überschrieben, wenn die Namen sich gleichen.

14.3.2.5 Bearbeitung von exportierten Rezepturdatensätzen

Einleitung

Sie können die exportierten Rezepturdatensätze bearbeiten und wieder importieren. So können Sie z. B. die Datei um neue Datensätze erweitern oder die vorhandenen Datensätze ändern. Die exportierten Rezepturdatensätze in CSV-Dateien bearbeiten Sie in Microsoft Excel.

Vorgehensweise

1. Starten Sie Microsoft Excel.
2. Im Menü "Datei" klicken Sie auf "Öffnen".
   Der Dialog "Datei Öffnen" wird geöffnet.
3. Wählen Sie den Pfad, in dem die exportierte Datensatzdatei gespeichert ist.
   Im Dialog "Öffnen" wählen Sie in der Klappliste die Option "Alle Dateien (*.*)" , damit die exportierte Datei erkannt wird.
4. Klicken Sie auf "Öffnen".
   Der Dialog "Textkonvertierungs-Assistent" wird geöffnet.
5. Im ersten Schritt klicken Sie auf die Schaltfläche "Weiter".
6. Im zweiten Schritt wählen Sie den Semikolon als Trennzeichen.
   Klicken Sie auf die Schaltfläche "Weiter".
7. Beenden Sie den Assistenten über die Schaltfläche "Fertigstellen".
8. Führen Sie die gewünschten Änderungen in der Datei aus.
9. Speichern Sie die geänderte Datei im Format CSV (Trennzeichen-getrennt) wieder.
10. Importieren Sie die geänderte Datei.
    Wenn Sie die Datei unter einem anderen Namen gespeichert haben, passen Sie vor dem Importieren den Dateinamen im Dialog "Import" an.

Ergebnis

Die exportierten Rezepturdatensätze wurden bearbeitet und in einer Rezeptur angelegt.

Siehe auch

Rezepturdatensätze importieren (Seite 2051)

14.3.2.6 Format der Daten von Rezepturdatensätze

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Rezepturen. Die Datei mit den Daten der Rezepturen muss im Dateiformat "*.csv" vorliegen. :

Struktur der Rezepturdaten

Der Aufbau der Importdatei ist zwingend vorgeschrieben. Das folgende Beispiel zeigt die Struktur für eine Rezeptur, die zwei Rezepturelemente mit je zwei Datensätzen enthält:

```
"ID";<Name Rezepturelement 1>;< Name
Rezepturelement 2>;"LastUser";"LastAccess[UTC]"<Zeilenumbruch>
<Rezepturdatensatznummer 1>;<Rezepturdatensatz 1
Wert 1>;<Rezepturdatensatz 1 Wert 2>;"<Letzter Benutzer 1>";<Datum
und Uhrzeit 1><Zeilenumbruch>
```

```
<Rezepturdatensatznummer 2>;<Rezepturdatensatz 2
Wert 1>;<Rezepturdatensatz 2 Wert 2>;"<Letzter Benutzer 2>";<Datum
und Uhrzeit 2><Zeilenumbruch>
```

ID der Sprache

Die erste Zeile, die immer vorhanden ist, beschreibt die Spaltenüberschriften als Kopfzeile. Nur die Namen der Rezepturelemente werden dynamisch eingefügt.

Jede weitere Zeile entspricht einem Rezepturdatensatz mit seinen jeweiligen Werten der Rezepturelemente und den Änderungsinformationen.

14.3.3 Meldungen importieren und exportieren

14.3.3.1 Meldungen exportieren

Einleitung

WinCC stellt eine Exportfunktion für Meldungen zur Verfügung.

Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Meldungen angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.

Meldungen exportieren

1. Klicken Sie in "Bitmeldungen" oder "Analogmeldungen" auf die Schlaltfläche .
   Der Dialog "Export" wird geöffnet.

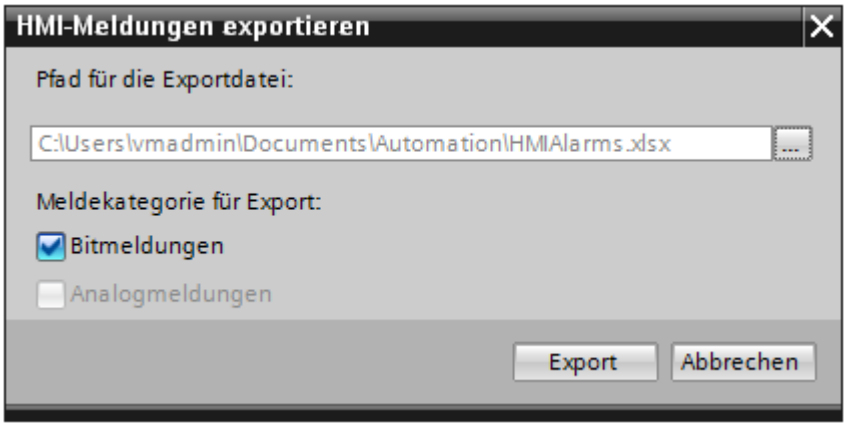

2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.

3. Legen Sie fest, ob Sie "Bitmeldungen" oder "Analogmeldungen" exportieren wollen.
4. Klicken Sie auf "Export". Der Export wird gestartet.

Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Siehe auch

Format der Daten von Analogmeldungen (Seite 2063)

Format der Daten von Bitmeldungen (Seite 2067)

14.3.3.2 Meldungen importieren

Einleitung

Meldungen werden über ihre Meldenummer identifiziert. Die Meldenummern müssen innerhalb der Meldetypen Analogmeldung und Bitmeldung eindeutig sein. Meldungen mit identischen Meldenummern werden daher überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.

Wenn eine xlsx-Datei leere Listeneinträge für bereits bestehende Meldungen enthält, werden diese beim Import ignoriert. Die Einträge der bestehenden Meldung bleiben erhalten und werden nicht durch leere ersetzt.

Voraussetzungen

- Eine xlsx-Datei mit Meldungen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "HMI-Meldungen" ist geöffnet.

Meldungen importieren

1. Klicken Sie in "Bitmeldungen" oder "Analogmeldungen" auf die Schaltfläche . Der Dialog "Import" wird geöffnet.

2. Klicken Sie auf die Schaltfläche "..." und wählen Sie die Datei, die Sie importieren wollen.
3. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet. Ein Laufbalken zeigt den Fortschritt des Importvorgangs an.

Ergebnis

Mit den importierten Daten wurden die entsprechenden Meldungen einschließlich der Meldetexte in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden sind, z. B. dynamische Parameter wie Variablen. Diese Parameter müssen jeweils in allen zu importierenden Sprachen vorhanden sein.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Dadurch kann einer Meldung eine Triggervariable vom falschen Typ, z. B. String, zugewiesen werden. Beim Übersetzen wird ein Fehler gemeldet.

Siehe auch

Format der Daten von Analogmeldungen (Seite 2063)

Format der Daten von Bitmeldungen (Seite 2067)

14.3.3.3 Meldungen importieren

Einleitung

Meldungen werden über den Namen der Meldung identifiziert. Wenn der Name der Meldung im Projekt bereits besteht, so wird die bestehende Meldung mit den Daten aus der xlsx-Datei überschrieben. Wenn die Meldung noch nicht besteht, wird eine neue Meldung angelegt.

Die Meldenummern müssen Projektweit eindeutig sein. Wenn eine Meldenummer durch den Import mehrfach vergeben wird, wird der Import der Meldung durchgeführt und die Meldenummer danach als ungültig markiert.

Voraussetzungen

- Eine xlsx-Datei mit Meldungen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "HMI-Meldungen" ist geöffnet.

Meldungen importieren

1. Klicken Sie in "Bitmeldungen" oder "Analogmeldungen" auf die Schaltfläche . Der Dialog "Import" wird geöffnet.

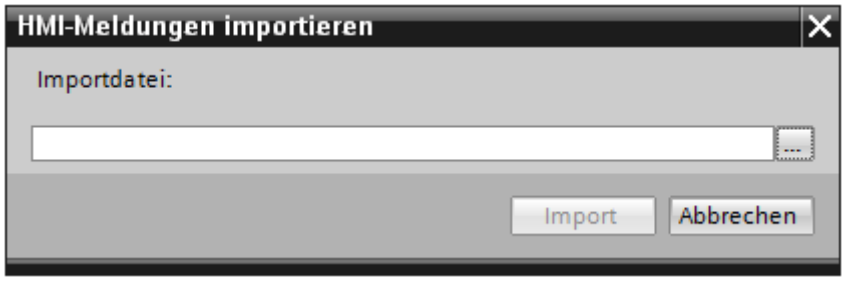

2. Klicken Sie auf die Schaltfläche "..." und wählen Sie die Datei, die Sie importieren wollen.
3. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet. Ein Laufbalken zeigt den Fortschritt des Importvorgangs an.

Hinweis

Für den Meldungsimport können sie keine Optionseinstellungen wählen.

Ergebnis

Mit den importierten Daten wurden die entsprechenden Meldungen einschließlich der Meldetexte in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Es ist möglich, eine Triggervariable vom falschen Typ z. B. String einer Meldung zuzuweisen. Beim Übersetzen wird ein Fehler gemeldet.

14.3.3.4 Format der Daten von Analogmeldungen

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Analogmeldungen. Die Datei mit den Daten der Analogmeldungen muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Meldungsdaten

Die Importdatei besteht in Microsoft Excel aus Tabellenblättern:

- Analog alarms (Analogmeldungen)
- Limits (Grenzen)

Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Analogmeldungen muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "Analog alarms"

	A	B	C	D	E
1	ID	Name	Event text [en-US], Alarm text	FieldInfo [Alarm text]	Class
2	1	Analog_alarm_1	AA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz		Errors
3	2	Analog_alarm_2	AA2 Warning-AC this text should be bold		Warnings
4	3	Analog_alarm_3	AA3 SDm-AC this text should be italic		SDm
5	4	Analog_alarm_4	AA4 SDo-AC <u>this text should be underlined</u>		SDo
6	5	Analog_alarm_5	AA5 SystemAcknowledgement-AC <blink>this text should be flashing<		System_Ackn
7	25	Analog_alarm_25	Internal AA23 switchDT: Deadband mode in case of violation - value HL		AADT-Interna
8	26	Analog_alarm_26	Internal AA23 switchDT: Deadband mode in case of violation - percent		AADT-Interna
9	31	Analog_alarm_31	Internal AA23 switchDT: Low limit violation static		AADT-Interna
10	32	Analog_alarm_32	Internal AA23 switchDT: High limit violation static		AADT-Interna
11	33	Analog_alarm_33	Internal AA23 switchDT: Low limit violation dynamic		AADT-Interna
12	34	Analog_alarm_34	Internal AA23 switchDT: High limit violation dynamic		AADT-Interna
13	35	Analog_alarm_35	Internal AA23 switchDT: delay 3 seconds High limit violation		AADT-Interna
14	42	Analog_alarm_40	Internal AA23 switchDT: delay 3 seconds Low limit violation LLV		AADT-Interna
15	23	Analog_alarm_41	AA23 DT in event text: <field ref="0" /> Bool, <field ref="1" /> Byte, <fi	<ref id = 0; type = AlarmTag; T	ACAT
16	24	Analog_alarm_42	AA24 DT in event text: <field ref="0" /> Timer, <field ref="1" /> Counte	<ref id = 0; type = AlarmTag; T	ACAT

Tabelle 14-1 Bedeutung der Einträge

Listeneintrag	Bedeutung
ID	Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.
Name	Name der Analogmeldung
Event text [de-DE], Alarm text	Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein Wenn ein dynamischer Paramter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet.
FeldInfo	Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt. Beispiel für dynamische Parameter: Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;> Textliste: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>
Class	Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.
Group	Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.
Trigger tag	Gibt die Variable an, die auf Grenzwert-Verletzung überwacht wird.

Listeneintrag	Bedeutung
Delay time value	Gibt die Verzögerungszeit an. Die Meldung wird erst dann ausgelöst, wenn die Grenzwert-Verletzung die angegebene Verzögerungszeit andauert.
Delay time unit	Gibt die Zeiteinheit für die Verzögerung an.
Report	Aktiviert die Protokollierung der jeweiligen Meldung auf einem Drucker. True oder "1" = Protokollierung aktiviert False oder "0" = Protokollierung deaktiviert. Die Protokollierung muss zusätzlich im Projekt global aktiviert werden.
Info text [de-DE], Info text	Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt. Die Bezeichnung des Feldes enthält eine Sprachkennung.

Beispiel für das Tabellenblatt "Limits"

	A	B	C	D	E	F	G
1	Alarm ID	Limit type	Limit value	Limit mode	Deadband mo	Deadband valu	Deadband in percent
2	1	Constant	0	Upper limit	Off	0	False
3	2	Constant	1	Upper limit	Off	0	False
4	3	Constant	2	Upper limit	Off	0	False
5	4	Constant	3	Upper limit	Off	0	False
6	5	Constant	4	Upper limit	Off	0	False
7	25	Constant	50	Upper limit	On both	5	False
8	26	Constant	50	Upper limit	On both	10	True
9	31	Constant	50	Lower limit	Off	0	False
10	32	Constant	50	Upper limit	Off	0	False
11	33	Tag	AASDTdyn	Lower limit	Off	0	False
12	34	Tag	AASDT1dyn	Upper limit	Off	0	False
13	35	Constant	50	Upper limit	Off	0	False
14	36	Constant	50	Lower limit	On both	5	False

Tabelle 14-2 Bedeutung der Einträge

Listeneintrag	Bedeutung
Alarm ID	Meldenummer Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.
Limit mode	Triggermodus Gibt die Methode an, mit welcher der Grenzwert überwacht wird.
Limit type	Gibt die Grenze an, die überwacht wird. Als Grenzwert ist sowohl eine Variable als auch eine Konstante möglich.

Listeneintrag	Bedeutung
Limit value	Grenzwert Gibt die Variable bzw. Konstante an, die auf Grenzwert-Verletzung überwacht wird.
Deadband mode	Hysteresemodus Gibt an, ob und in welchen Fällen eine Hysterese angewendet wird. Bei "Gehen" Bei "Kommen" Bei "Kommen" und "Gehen"
Deadband in percent	0 = Der bei "Hysterese" angegebene Wert wird als Absolutwert betrachtet. 1 = Der bei "Hysterese" angegebene Wert wird prozentual auf den Grenzwert bezogen.
Deadband mode	Hysterese Gibt eine Konstante als Wert der Hysterese an.

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

Siehe auch

Meldungen exportieren (Seite 2054)

Meldungen importieren (Seite 2055)

Format der Daten von Bitmeldungen (Seite 2061)

14.3.3.5 Format der Daten von Bitmeldungen

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Bitmeldungen. Die Datei mit den Daten der Bitmeldungen muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Meldungsdaten

Die Importdatei besteht in Microsoft Excel aus dem Tabellenblättern "Discrete alarms" (Bitmeldungen). Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Bitmeldungen muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "Discrete alarms"

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	ID	Name	Event text [en-US], Alarm text	FieldInfo [Alarm text]	Class	Trigger tag	Trigge	Acknowle	Acknow	PLC ackno	PLC ackn	Group	Report	Info text [en-US], Info text
2	1	Discrete_alarm_1	DA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz bou\n		Errors	HMI_AC-Test	0	<No value	0	<No value	0	<No value	False	Infotext DA1 Error
3	2	Discrete_alarm_2	DA2 Warning-AC this text should be bold		Warnings	HMI_AC-Test	1	<No value	0	<No value	0	<No value	False	Infotext DA2 Warning
4	3	Discrete_alarm_3	DA3 SDm-AC this text should be italic		SDm	HMI_AC-Test	2	<No value	0	<No value	0	<No value	False	Infotext DA3 SDm
5	4	Discrete_alarm_4	DA4 SDo-AC <u>this text should be underlined</u>		SDo	HMI_AC-Test	3	<No value	0	<No value	0	<No value	False	Infotext DA4 SDo
6	5	Discrete_alarm_5	DA5 SystemAcknowledgement-AC <blink>this text should be flashing</blink		System_Ackn	HMI_AC-Test	4	<No value	0	<No value	0	<No value	False	Infotext DA5 SystemAcknowledgement
7	6	Discrete_alarm_6	DA6 SystemNoAcknowledgement-AC mixed test: Bold, Italic, <i		System_No_A	HMI_AC-Test	5	<No value	0	<No value	0	<No value	False	Infotext DA6 SystemNoAcknowledgement
8	7	Discrete_alarm_7	DA7 DT in event text: <field ref="0" /> Integer, <field ref="1" /> Real, <field re	<ref id = 0; type = AlarmTag; Tag = PL	ACAT	HMI_TriggerAT	0	<No value	0	<No value	0	<No value	False	
9	8	Discrete_alarm_8	DA8 DT in event text: <field ref="0" /> S5Time, <field ref="1" /> Timer, <field r	<ref id = 0; type = AlarmTag; Tag = PL	ACAT	HMI_TriggerAT	1	<No value	0	<No value	0	<No value	False	
10	11	Discrete_alarm_9	DA11 DT in event text: <field ref="0" /> Int, <field ref="1" /> Real, <field ref="2	<ref id = 0; type = AlarmTag; Tag = Ir	ACAT	HMI_TriggerInte	0	<No value	0	<No value	0	<No value	False	
11	12	Discrete_alarm_10	DA12 DT in event text: <field ref="0" /> UDint, <field ref="1" /> UInt;	<ref id = 0; type = AlarmTag; Tag = U	ACAT	HMI_TriggerInte	1	<No value	0	<No value	0	<No value	False	
12	13	Discrete_alarm_11	DA13 Textformat: Integer: <field ref="0" /> decimal, <field ref="1" /> binary, <	<ref id = 0; type = AlarmTag; Tag = H	ACAT	HMI_TFAuD	0	<No value	0	<No value	0	<No value	False	

Tabelle 14-3 Bedeutung der Einträge

Listeneintrag	Bedeutung
ID	Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.
Name	Name der Analogmeldung
Event text [de-DE], Alarm text	Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein. Wenn ein dynamischer Paramter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet.
FeldInfo	Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt. Beispiel für dynamische Parameter: Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;> Textliste: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>
Class	Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.
Group	Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.
Trigger tag	Gibt die Variable an, die das Bit enthält durch das die Meldung ausgelöst wird.
Trigger bit	Gibt die Nummer des Bits an, durch das die Meldung ausgelöst wird.
Acknowledge tag	Gibt die Variable an, die das Bit enthält, das beim Quittieren durch den Bediener gesetzt wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.
Acknowledgment bit	Gibt die Nummer des Bits an, das gesetzt wird, wenn der Bediener die Meldung quittiert.
PLC acknowledgement tag	Gibt die Variable an, die das Bit enthält, durch das die Meldung vom Steuerungsprogramm quittiert wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.
PLC acknowledgment bit	Gibt die Nummer des Bits an, durch das die Meldung vom Steuerungsprogramm quittiert wird.

Listeneintrag	Bedeutung
Delay time value	Gibt die Verzögerungszeit an. Die Meldung wird erst dann ausgelöst, wenn die Grenzwert-Verletzung die angegebene Verzögerungszeit andauert.
Delay time unit	Gibt die Zeiteinheit für die Verzögerung an.
Report	Aktiviert die Protokollierung der jeweiligen Meldung auf einem Drucker. True oder "1" = Protokollierung aktiviert False oder "0" = Protokollierung deaktiviert. Die Protokollierung muss zusätzlich im Projekt global aktiviert werden.
Info text [de-DE], Info text	Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt. Die Bezeichnung des Feldes enthält eine Sprachkennung.

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

Siehe auch

Meldungen exportieren (Seite 2054)

Meldungen importieren (Seite 2055)

Format der Daten von Analogmeldungen (Seite 2058)

14.3.3.6 Format der Daten von Analogmeldungen

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Analogmeldungen. Die Datei mit den Daten der Analogmeldungen muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Meldungsdaten

Die Importdatei besteht in Microsoft Excel aus zwei Tabellenblättern:

- Analog alarms (Analogmeldungen)
- Limits (Grenzen)

Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Analogmeldungen muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "Analog alarms"

ID	Name	Class	Group	Priority	CPU number	PLC number	Trigger tag	Delay value	Delay time unit
1	AnalogAlarm_1	Errors	AlarmGroup_1	1	1	1	Tag_1	5	Second
2	AnalogAlarm_2	Errors	AlarmGroup_2	1	1	1	Tag_2	5	Second
3	AnalogAlarm_3	Warnings		1	1	1	Tag_3	0	
4	AnalogAlarm_4	Warnings		1	1	1	Tag_4	0	

Acknowledgement tag	Acknowledgement bit	Single acknowledgement	Status tag	Status bit	Alarm annunciation	Display suppression mask
Tag_2	0	true	Tag_3	0	false	0x00000001
Tag_2	1	true	Tag_3	1	false	0x00000001
			<No Value>		false	
			<No Value>		false	

Event text [de-DE], Alarm text 1	Additional text 1 [de-DE], Alarm text 2	Additional text 3 [de-DE], Alarm text 4	Info text [de-DE], Info text
Anwendertext 1	Zusatztext 1		Infotext 1
Anwendertext 2		Zusatztext 2	Infotext 2
Anwendertext 3			Infotext 3
Anwendertext 4 <ref domaintype="AnalogAlarmParameterFieldInfo">\<link reference="1" valuetype="ID"/\> \<format displaytype="" length="5" alignment="Right" precision="0" zeropadding="False"/\> \</ref\>			Infotext 4

Tabelle 14-4 Bedeutung der Einträge

Listeneintrag	Bedeutung
ID	Zeigt die Nummer der Analogmeldung. Die Meldenummer muss projektweit eindeutig sein.
Name	Zeigt den Namen der Analogmeldung. Der Name dient zum Referenzieren einer Meldung. Meldungen mit identischem Namen werden daher beim Import überschrieben. Eine Meldung mit nicht existentem Namen wird neu angelegt.
Event text [de-DE], Alarm text	Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein. Wenn ein dynamischer Paramter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet.
FeldInfo	Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt. Beispiel für dynamische Parameter: Parameter: <ref id = 0; type = AlarmAnalogParameter; Parameter = Limit; DisplayType = Decimal; Length = 1; Precision = 0; Alignment = Right; ZeroPadding = False;> Systemparameter: <ref id = 0; type = CommonSystemParameter; SystemParameter = UserName;>
Class	Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.

Listeneintrag	Bedeutung
Group	Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.
Priority	Gibt die Priorität der Meldung an. Über die Priorität sortieren Sie die Meldungen in der Meldeanzeige. Wertebereich: 0 -16
CPU number	Gibt die CPU-Nummer an. Diese Option ist nur dann aktiv, wenn im Editor "HMI-Meldungen" auf der Registerkarte "Meldeblöcke > Systemblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
PLC number	Gibt die Nummer der PLC an. Diese Option ist nur dann aktiv, wenn im Editor "HMI-Meldungen" auf der Registerkarte "Meldeblöcke > Systemblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
Trigger tag	Gibt die Variable an, die auf Grenzwert-Verletzung überwacht wird.
Verzögerung	Gibt die Verzögerungszeit an. Die Meldung wird erst dann ausgelöst, wenn die Grenzwert-Verletzung die angegebene Verzögerungszeit andauert.
Delay time unit	Gibt die Zeiteinheit für die Verzögerung an.
Acknowledge tag	Gibt die Variable an, die das Bit enthält, das beim Quittieren durch den Bediener gesetzt wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.
Acknowledgment bit	Gibt die Nummer des Bits an, das gesetzt wird, wenn der Bediener die Meldung quittiert.
Single acknowledgment	Gibt an, ob jede einzelne Meldung quittiert werden muss. True = Meldung muss einzeln quittiert werden.
Status tag	Zeigt die Variable, welche die Meldezustände Gekommen, Gegangen und die Quittieranforderung speichert.
Status bit	Gibt die Nummer des Bits an, das gesetzt wird, das die Meldung quittiert.
Alarm annonciation	Gibt an, ob eine Meldung unterdrückt wird. True = Meldung wird unterdrückt.
Display suppresion mask	Gibt die Methode zum Unterdrücken der Anzeige an.
Additonal Text	Zeigt den zusätzlichen Text der Meldung.
Info text [de-DE], Info text	Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt. Die Bezeichnung des Feldes enthält eine Sprachkennung.

Beispiel für das Tabellenblatt "Limits"

Alarm ID	Name	Limit mode	Limit type	Limit value	Suppression	Deadband mode	Deadband in percent	Deadband value
1	AnalogAlarmLimit_1	Upper limit	Constant	100	false	On above thresold	false	105
1	AnalogAlarmLimit_2	Upper limit	Constant	110	false	On above thresold	false	115
2	AnalogAlarmLimit_1	Upper limit	Constant	100	false	On above thresold	false	105
2	AnalogAlarmLimit_2	Upper limit	Constant	110	false	On above thresold	false	115
3	AnalogAlarmLimit_1	Lower limit	Tag	LimitTag_1	false	Off		
4	AnalogAlarmLimit_1	Lower limit	Tag	LimitTag_2	false	Off		

Tabelle 14-5 Bedeutung der Einträge

Listeneintrag	Bedeutung
Alarm ID	Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Durch die Meldenummer werden die Grenzen zu einer Meldungen zugeordnet.
Name	Zeigt den Namen der Analogmeldung.
Limit mode	Gibt die Methode an, mit welcher der Grenzwert überwacht wird.
Grenzen Limit type	Gibt die Grenze an, die überwacht wird. Als Grenzwert ist sowohl eine Variable als auch eine Konstante möglich.
Limit value	Gibt die Variable bzw. Konstante an, die auf Grenzwert-Verletzung überwacht wird.
Suppression	Gibt an, ob eine Meldung unterdrückt wird. True = Die Meldung wird unterdrückt.
Deadband mode	Gibt an, ob und in welchen Fällen eine Hysterese angewendet wird. Bei "Gehen" Bei "Kommen" Bei "Kommen" und "Gehen"
Deadband in percent	0 = Der bei "Hysterese" angegebene Wert wird als Absolutwert betrachtet. 1 = Der bei "Hysterese" angegebene Wert wird prozentual auf den Grenzwert bezogen.
Deadband mode	Gibt eine Konstante als Wert der Hysterese an.

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

Siehe auch

Meldungen exportieren (Seite 2054)

Meldungen importieren (Seite 2055)

14.3.3.7 Format der Daten von Bitmeldungen

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Bitmeldungen. Die Datei mit den Daten der Bitmeldungen muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Meldungsdaten

Die Importdatei besteht in Microsoft Excel aus dem Tabellenblatt "Discrete alarms" (Bitmeldungen). Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Bitmeldungen muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "Discrete alarms"

ID	Name	Class	Group	Priority	CPU number	PLC number	Trigger tag	Trigger bit	Trigger mode	Alarm parameter 1	Alarm
1	DiscreteAlarm_1	Errors	AlarmGroup_1	1			Tag_1	0	On rising edge	Tag_1	<No V
2	DiscreteAlarm_2	Errors	AlarmGroup_2	1			Tag_1	0	On rising edge	Tag_1	<No V
3	DiscreteAlarm_3	Warning		1			Tag_1	0	On falling edge	<No Value>	<No V
4	DiscreteAlarm_4	Warning		1			Tag_1	0	On falling edge	<No Value>	<No V

Alarm parameter 4	Alarm parameter 5	Alarm parameter 6	Alarm parameter 7	Alarm parameter 8	Alarm parameter 9	Alarm parameter 10
<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>
<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>
<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>
<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>	<No Value>

Acknowledgement tag	Acknowledgement bit	Single acknowledgement	Status tag	Status bit	Alarm annunciation	Display suppression m
Tag_2	0	true	Tag_3	0	false	0x00000000
Tag_2	1	true	Tag_3	1	false	0x00000000
			<No Value>		false	
			<No Value>		false	

Event text [de-DE], Alarm text 1	Additional text 1 [de-DE], Alarm text 2	Additional text 3 [de-DE], Alarm text 4	Info te
Anwendertext 1	Zusatztext 1		Infotex
Anwendertext 2		Zusatztext 2	Infotex
Anwendertext 3			Infotex
Anwendertext 4 <ref domaintype="ParameterFieldInfo">\<link reference="MyGroup.TagTable1.Tag_1" valuetype="ID"/\> \<format displaytype="Decimal" length="5" alignment="Right" precision="2" zeropadding="0"/\> \</ref\>			Infotex

Tabelle 14-6 Bedeutung der Einträge

Listeneintrag	Bedeutung
ID	Zeigt die Nummer der Bitmeldung. Die Meldenummer ist eindeutig.
Name	Zeigt den Namen der Bitmeldung. Der Name dient zum Referenzieren einer Meldung. Meldungen mit identischem Namen werden daher beim Import überschrieben. Eine Meldung mit nicht existentem Namen wird neu angelegt.
Event text [de-DE], Alarm text	Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein. Wenn ein dynamischer Paramter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet.
FeldInfo	Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt. Beispiel für dynamische Parameter: Parameter: <ref id = 0; type = AlarmDiscreteParameter; Parameter = Parameter 1; Tag = HMI_Variable_1; DisplayType = Decimal; Length = 5; Precision = 0; Alignment = Right; ZeroPadding = False;> Systemparamter: <ref id = 0; type = CommonSystemParameter; SystemParameter = UserName;>
Class	Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.
Group	Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.
Priority	Gibt die Priorität der Meldung an. Über die Priorität sortieren Sie die Meldungen in der Meldeanzeige. Wertebereich: 0 -16
CPU number	Gibt die CPU-Nummer an. Diese Option ist nur dann aktiv, wenn im Editor "HMI-Meldungen" auf der Registerkarte "Meldeblöcke > Systemblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
PLC number	Gibt die Nummer der PLC an. Diese Option ist nur dann aktiv, wenn im Editor "HMI-Meldungen" auf der Registerkarte "Meldeblöcke > Systemblöcke" der Meldeblock "CPU/PLC-Nummer" aktiviert ist.
Trigger tag	Gibt die Variable an, die das Bit enthält, durch das die Meldung ausgelöst wird.
Trigger bit	Gibt die Nummer des Bits an, durch das die Meldung ausgelöst wird.
Trigger mode	Gibt die Methode an, mit welcher der Grenzwert überwacht wird.
Alarm parameter	Gibt die Variable an, die einem Parameterblock zugeordnet ist, um Prozesswerte in Meldetexten auszugeben.
Acknowledge tag	Gibt die Variable an, die das Bit enthält, das beim Quittieren durch den Bediener gesetzt wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.
Acknowledgment bit	Gibt die Nummer des Bits an, das gesetzt wird, wenn der Bediener die Meldung quittiert.
Single acknowledgment	Gibt an, ob jede einzelne Meldung quittiert werden muss. True = Meldung muss einzeln quittiert werden.
Status tag	Zeigt die Variable, welche die Meldezustände Gekommen, Gegangen und die Quittieranforderung speichert.

Listeneintrag	Bedeutung
Status bit	Gibt die Nummer des Bits an, das gesetzt wird, das die Meldung quittiert.
Alarm annonciation	Gibt an, ob eine Meldung unterdrückt wird. True = Meldung wird unterdrückt.
Display suppresion mask	Gibt die Methode zum Unterdrücken der Anzeige an.
Additonal Text	Zeigt den zusätzlichen Text der Meldung.
Info text [de-DE], Info text	Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt. Die Bezeichnung des Feldes enthält eine Sprachkennung.

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

Siehe auch

Meldungen exportieren (Seite 2054)

Meldungen importieren (Seite 2055)

14.3.4 Variablen importieren und exportieren

14.3.4.1 Variablen exportieren

Einleitung

WinCC stellt eine Exportfunktion für Variablen zur Verfügung.

Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Variablen angelegt.
- Der Editor "HMI-Variablen" ist geöffnet.

Variablen exportieren

1. Klicken Sie in der der Registerkarte "HMI-Variablen" auf die Schlaltfläche .
   Der Dialog "Export" wird geöffnet.

2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.
3. Klicken Sie auf "Export". Der Export wird gestartet.

Hinweis

Die Versionsnummer der xlsx-Datei mit den exportierten Variablen hängt von der TIA Portal-Version ab. Wenn die xlsx-Datei aus einem Projekt in WinCC V13 SP1 exportiert wurde, hat sie die Versionsnummer 1.2. Wenn die xlsx-Datei aus einem Projekt vor WinCC V13 SP1 exportiert wurde, hat sie die Versionsnummer 1.1.

Hinweis

Es ist nicht möglich, HMI-Variablen des Datentyps "UDT", die strukturierte Elemente enthalten, über Excel zur nachfolgenden Bearbeitung zu exportieren.

Nach dem Export erscheint im Excel nur die übergeordnete HMI-Variable. Ihre untergeordneten Elemente können nicht bearbeitet werden.

Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

14.3.4.2 Variablen importieren (ab WinCC V13 SP1)

Einleitung

Variablen werden über den Namen der Variablen identifiziert. Wenn der Name der Variablen im Projekt bereits besteht, so wird die bestehende Variable mit den Daten aus der xlsx-Datei überschrieben. Wenn die Variable noch nicht besteht, wird eine neue Variable angelegt.

Voraussetzungen

- Eine xlsx-Datei mit Variablen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.

Variablen importieren

1. Klicken Sie in der Projektnavigation auf "HMI-Variablen".
2. Doppelklicken Sie "Alle Variablen anzeigen". Der Editor "HMI-Variablen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche . Der Dialog "Import" wird geöffnet.

4. Klicken Sie auf die Schaltfläche "..." und wählen Sie die Datei, die Sie importieren wollen.
5. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet.

Ergebnis

Die entsprechenden Variablen wurden in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Es ist möglich, eine Triggervariable vom falschen Typ z. B. String einer Variable zuzuweisen. Beim Übersetzen wird ein Fehler gemeldet.

14.3.4.3 Variablen importieren (vor WinCC V13 SP1)

Einleitung

Variablen werden über den Namen der Variablen identifiziert. Wenn der Name der Variablen im Projekt bereits besteht, so wird die bestehende Variable mit den Daten aus der xlsx-Datei überschrieben. Wenn die Variable noch nicht besteht, wird eine neue Variable angelegt.

Die Versionsnummer der xlsx-Datei hängt von der TIA Portal-Version ab. Wenn die xlsx-Datei aus einem Projekt in WinCC V13 SP1 exportiert wurde, hat sie die Versionsnummer 1.2. Wenn die xlsx-Datei aus einem Projekt vor WinCC V13 SP1 exportiert wurde, hat sie die Versionsnummer 1.1.

Um Dateien mit der Versionsnummer 1.2 in ein Projekt zu importieren, das vor WinCC V13 SP1 erstellt wurde, müssen Sie die Dateien aufbereiten.

Voraussetzungen

- Eine xlsx-Datei mit Variablen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.

Dateien für den Import aufbereiten

1. Öffnen Sie die betreffende Datei im Microsoft Excel.
2. Wählen Sie "Datei > Eigenschaften > Erweiterte Eigenschaften".
   Der Dialog "Eigenschaften" wird geöffnet.
3. Wechseln Sie zu der Task Card "Anpassen".
4. Selektieren Sie im Bereich "Eigenschaften" "TIA_Version".
5. Ändern Sie in der Spalte "Wert" den Wert von "1.2" auf "1.1".
6. Klicken Sie auf die Schaltfläche "Ändern".
7. Klicken Sie auf "OK".
8. Löschen Sie in der Datei die rechte Spalte "Synchronization".
9. Speichern Sie die Datei.

Variablen importieren

1. Klicken Sie in der Projektnavigation auf "HMI-Variablen".
2. Doppelklicken Sie "Alle Variablen anzeigen". Der Editor "HMI-Variablen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche . Der Dialog "Import" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "..." und wählen Sie die Datei, die Sie importieren wollen.
5. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet.

Ergebnis

Die entsprechenden Variablen wurden in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Es ist möglich, eine Triggervariable vom falschen Typ z. B. String einer Variable zuzuweisen. Beim Übersetzen wird ein Fehler gemeldet.

14.3.4.4 Format der Variablendaten

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei mit den Variablendaten für den Variablenimport. Die Datei mit den Variablendaten muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Variablendaten

Die Importdatei besteht in Microsoft Excel aus Tabellenblättern:

- HMI Tags (HMI-Variablen)
- Multiplexing (Multiplex Variablen)

Jede Variable erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Variablendaten muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "HMI Tags"

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Name	Path	Connection	PLC tag	DataType	Length	Address	Access Meth	Indirect ac	Index tag	Start value	ID tag	Coding	Display na	Comment	Acquisitio	Acquis	High High
2	HMI_Int	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
3	Mux_Tag_1	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	True	Mux_Tag_	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
4	Mux_Tag_2	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	True	Mux_Tag_	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
5	Mux_11	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
6	Mux_21	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
7	Mux_13	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
8	Mux_12	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
9	Mux_23	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
10	Mux_22	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
11	Mux_Tag_1_Index	Internal Tags	<No Value>	<No Value>	UInt	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
12	Mux_Tag_12	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Value>	True	Mux_Tag_	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
13	Mux_Tag_11	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Value>	True	Mux_Tag_	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
14	Mux_Tag_13	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Value>	True	Mux_Tag_	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
15	HMI_UDInt	Internal Tags	<No Value>	<No Value>	UDInt	4	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
16	Gauge_Process	Default tag table	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
17	Button_Tag_4	Default tag table	<No Value>	<No Value>	Int	2	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
18	HMI_USInt	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Value>	False	<No Value	<No Value>	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit
19	Data_block_2_PLC_DateTime_2	Default tag table	HMI_connection_1	Data_block_2.PLC_I	Date_And_Time	8	%DB28.DBX598.(	<absolute ac	False	<No Value	01.01.1999 12:00	0	Binary	<No Value	<No Value	Cyclic in o	1 s	NoLimit

Tabelle 14-7 Bedeutung der Einträge

Listeneintrag	Bedeutung
Name	Gibt den projektierten Namen einer HMI-Variable an.
Path	Gibt an, in welchem Ordnen in der Projektnavigation die Variable abgelegt wird. Die Ordnerstruktur wird durch "\" dargestellt: "Ordnername1\Ordnername2\Variablenname".
PLC Tag	Gibt an, ob die Variable mit einer PLC-Variable verbunden ist.
Connection	Gibt den Namen der Verbinung zur PLC an.
Data type	Gibt den Datentyp der Variable an. Die erlaubten Datentypen sind abhängig vom verwendeten Kommunikationstreiber. Weitere Informationen zu den erlaubten Datentypen der verschiedenen Kommunikationstreiber finden Sie in der Dokumentation im Kapitel "Kommunikation".
Length	Gibt die Länge der Variable an. Der Eintrag ist nur für Datentypen mit dynamischer Länge hilfreich z.B. String. Bei anderen Datentypen bleibt der Eintrag leer.
Address	Gibt die Variablenadresse in der PLC an. Die Variablenadresse muss genau so geschrieben sein, wie sie in WinCC dargestellt wird, z. B. "%DB1.DBW0". Für interne Variablen bleibt die Variablenadresse leer.
Multiplexing	Gibt an, ob Multiplexing verwendet wird.
Index tag	Zeigt den Namen der Indexvariable für das Multiplexen an. In Runtime wird zunächst der Wert der Indexvariable ausgelesen. Dann wird auf die Variable zugegriffen, die an der entsprechenden Stelle der Variablenliste steht.
StartValue	Gibt den Startwert der Variable an.
ID tag	Über die Aktualisierungskennung aktualisieren Sie den Wert einer Variablen mit Hilfe einer Funktion oder eines PLC-Auftrags. Innerhalb eines Bediengeräts muss die Aktualisierungskennung eindeutig sein.
Coding	Zeigt die Methode zur Codierung an.
DiplayName [de_DE]	Zeigt den Anzeigenamen einer HMI-Variable. Die Bezeichnung des Feldes enthält eine Sprachkennung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Anzeigenamen müssen für den Import mit einer Sprachkennung versehen sein. Beim Import wird der Text in die zugehörige Projektsprache importiert.
Acquisition mode	Gibt die Erfassungsart der Variable an.

Listeneintrag	Bedeutung
Acquisition cycle	Gibt den Erfassungszyklus der Variable an. Der Erfassungszyklus muss genau so geschrieben werden, wie die Darstellung in WinCC. Der Wert ist sprachunabhängig und daher in jeder Sprache gleich anzugeben. Der Standardwert ist "1 s". Ist die Erfassungsart der Variable "auf Anforderung", dann ist der Erfassungszyklus undefiniert. Benutzerdefinierte Erfassungszyklen müssen vorher angelegt sein, da die Datei sonst nicht importiert wird.
High High Limit type	Gibt an, ob der Grenzwert "High high" mit einer Konstanten, Variablen oder gar nicht überwacht wird.
High High Limit	Zeigt den Grenzwert "High High" an.
High Limit type	Gibt an, ob der Grenzwert "High" mit einer Konstanten, Variablen oder gar nicht überwacht wird.
High Limit	Zeigt den Grenzwert "High" an.
Low Limit type	Gibt an, ob der Grenzwert "Low" mit einer Konstanten, Variablen oder gar nicht überwacht wird.
Low Limit	Zeigt den Grenzwert "Low" an.
Low Low Limit type	Gibt an, ob der Grenzwert "Low Low" mit einer Konstanten, Variablen oder gar nicht überwacht wird.
Low Low Limit	Zeigt den Grenzwert "Low Low" an.
Linear scaling	Gibt an, ob die lineare Skalierung aktiviert ist. Der Eintrag kann nur für externe Variablen verwendet werden.
End value PLC	Gibt den Endwert der PLC-Variable an.
Start value PLC	Gibt den Startwert der PLC-Variable an.
End value HMI	Gibt den Endwert der HMI-Variable an.
Start value HMI	Gibt den Startwert der HMI-Variable an.

Beispiel für das Tabellenblatt "Multiplexing"

	A	B	C
1	HMI Tag name	Multiplex Tag	Index
2	Mux_Tag_1	Mux_11	0
3	Mux_Tag_1	Mux_12	1
4	Mux_Tag_1	Mux_13	2
5	Mux_Tag_2	Mux_21	0
6	Mux_Tag_2	Mux_22	1
7	Mux_Tag_2	Mux_23	2
8	Mux_Tag_12	HMI_Array_Mux2	-1
9	Mux_Tag_11	HMI_Array_Mux1	-1
10	Mux_Tag_13	HMI_Array_Mux3	-1

Tabelle 14-8 Bedeutung der Einträge

Listeneintrag	Bedeutung
Name	Gibt den projektierten Namen einer HMI-Variable an, die eine indirekte Adressierung verwendet. Die HMI-Variable muss im Tabellenblatt "HMI Tags" vorhanden sein.
Index	Zeigt den Wert an, von dem die Auswahl der Variable abhängt.
Multiplex Tag	Zeigt die zum Indexwert zugehörige Variable aus der Variablenliste.

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Variablen.

14.3.4.5 Format der Variablendaten

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei mit den Variablendaten für den Variablenimport. Die Datei mit den Variablendaten muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Variablendaten

Die Importdatei besteht in Microsoft Excel aus Tabellenblättern:

- HMI Tags (Variablen)
- SubstituteValueUsage (Ersatzwert)

Jede Variable erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Variablendaten muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "HMI Tags"

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Name	Path	Connection	PLC tag	DataType	HMI DataTy	Length	Address	Access Me	Start valu	Quality Cc	Persisten	Substitute	Tag value	Update M	Coding	Comment	High High
2	Tag_1	Default table	HMI_Connection_1	Data_block_1.Static_1	Real	Float	4	%DB1.DBD	<No Value	<No Value	True	False	25	<No Value	Client/Ser	Binary	<No Value	NoLimit
3	Tag_2	Default table	<No Value>	<No Value>	Int	Int	2	<No Value>	<No Value	<No Value	False	False	<No Value	<No Value	Client/Ser	Binary	<No Value	NoLimit

Tabelle 14-9 Bedeutung der Einträge

Listeneintrag	Bedeutung
Name	Gibt den projektierten Namen einer HMI-Variable an.
Path	Gibt an, in welchem Ordnen in der Projektnavigation die Variable abgelegt wird. Die Ordnerstruktur wird durch "," dargestellt: "Ordnername1,Ordnername2,Variablenname".
PLC Tag	Gibt an, welche PLC-Variable mit der HMI-Variable verbunden ist.
Connection	Gibt den Namen der Verbindung an.
Date type	Gibt den Datentyp der PLC-Variable an. Die erlaubten Datentypen sind abhängig vom verwendeten Kommunikationstreiber. Weitere Informationen zu den erlaubten Datentypen der verschiedenen Kommunikationstreiber finden Sie in der Dokumentation im Kapitel "Kommunikation".
HMI Data type	Gibt den Datentyp der HMI-Variable an. Die erlaubten Datentypen sind abhängig vom verwendeten Kommunikationstreiber. Weitere Informationen zu den erlaubten Datentypen der verschiedenen Kommunikationstreiber finden Sie in der Dokumentation im Kapitel "Kommunikation".
Length	Gibt die Länge der Variable in Bytes an. Der Eintrag ist nur für Datentypen mit dynamischer Länge hilfreich z.B. String. Bei anderen Datentypen bleibt der Eintrag leer.
Address	Gibt die Variablenadresse in der PLC an. Die Variablenadresse muss genau so geschrieben sein, wie sie in WinCC dargestellt wird, z. B. "%DB1.DBW0". Für interne Variablen bleibt die Variablenadresse leer.
Start Value	Gibt den Startwert der Variable an.
Substitute Value	Gibt den Ersatzwert an. Der Ersatzwert wird eingesetzt, wenn ein fehlerhafter Prozesswert gelesen wird.
ID tag	Über die Aktualisierungskennung aktualisieren Sie den Wert einer Variablen mit Hilfe einer Funktion oder eines PLC-Auftrags innerhalb eines Bediengeräts muss die Aktualisierungskennung eindeutig sein.
High High Limit type	Gibt an, ob der Grenzwert "High high" mit einer Konstanten oder gar nicht überwacht wird.
High High Limit	Zeigt den Grenzwert "High High" an.
High Limit type	Gibt an, ob der Grenzwert "High" mit einer Konstanten oder gar nicht überwacht wird.
High Limit	Zeigt den Grenzwert "High" an.
Low Limit type	Gibt an, ob der Grenzwert "Low" mit einer Konstanten oder gar nicht überwacht wird.
Low Limit	Zeigt den Grenzwert "Low" an.
Low Low Limit type	Gibt an, ob der Grenzwert "Low Low" mit einer Konstanten oder gar nicht überwacht wird.
Low Low Limit	Zeigt den Grenzwert "Low Low" an.
Linear scaling	Gibt an, ob die lineare Skalierung aktiviert ist. Der Eintrag kann nur für externe Variablen verwendet werden.
End value PLC	Gibt den Endwert der PLC-Variable an.
Start value PLC	Gibt den Startwert der PLC-Variable an.
End value HMI	Gibt den Endwert der HMI-Variable an.
Start value HMI	Gibt den Startwert der HMI-Variable an.

Beispiel für das Tabellenblatt "SubstituteValueUsage"

	A	B
1	HMI Tag name	Substitute Value Usage
2	Tag_1	StartValue
3	Tag_1	ConnectionError

Tabelle 14-10 Bedeutung der Einträge

Listeneintrag	Bedeutung
Parent	Gibt die Variable an, für die ein Ersatzwert definiert wird. Die Variable muss im Tabellenblatt "HMI-Tags" vorhanden sein.
Substitute Value Usage	Gibt den Ersatzwert an. Den Ersatzwert verwenden Sie in folgenden Fällen: • Als Startwert • Bei Kommunikationsfehlern • High High Grenzwert • Low Low Grenzwert

Hinweis

"No Value" in der Tabelle

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Variablen.

14.3.5 Textlisten importieren und exportieren

14.3.5.1 Textlisten exportieren

Einleitung

WinCC stellt eine Exportfunktion für Textlisten zur Verfügung.

Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Textlisten angelegt.
- Der Editor "Text- & Grafiklisten" ist geöffnet.

Textlisten exportieren

1. Klicken Sie in der der Registerkarte "Textlisten" auf die Schlaltfläche .
   Der Dialog "Export" wird geöffnet.

2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.
3. Standardmäßig werden Texte in allen definierten Projektsprachen exportiert.
   Wenn Sie einzelne Sprachen nicht exportieren möchten, deaktivieren Sie die nicht benötigten Sprachen im Dialog.
4. Klicken Sie auf "Export". Der Export wird gestartet.

Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Siehe auch

Textlisten importieren (Seite 2079)

Format der Daten für Textlisten (Seite 2080)

14.3.5.2 Textlisten importieren

Einleitung

Im Folgenden importieren Sie Textlisten aus einer xlsx-Datei in WinCC.

Voraussetzungen

- Eine xlsx-Datei mit Textlisten ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.

- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "Text- & Grafiklisten" ist geöffnet.

Textlisten importieren

1. Klicken Sie in der Registerkarte "Textlisten" auf die Schaltfläche .
   Der Dialog "Import" wird geöffnet.

2. Wählen Sie unter "Dateiauswahl" die Datei, die Sie importieren wollen.
3. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet.

Ergebnis

Sie haben die Textlisten importiert. Die entsprechenden Textlisten wurden in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

Siehe auch

Textlisten exportieren (Seite 2078)

14.3.5.3 Format der Daten für Textlisten

Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei mit den Textlisten für den Import. Die Datei mit den Textlistendaten muss im Dateiformat "*.xlsx" vorliegen.

Struktur der Variablendaten

Die Importdatei besteht in Microsoft Excel aus zwei Tabellenblättern:

- TextList (Textlisten)
- TextListEntry (Textlisteneintrag)

Jede Textliste erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Daten muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "TextList"

	A	B	C
1	Name	ListRange	Comment
2	TLValue/Range	Decimal	
3	TLBit	Bit	
4	TLBitnumber	Binary	
5	TLall_1	Decimal	
6	TLall_2	Decimal	

Tabelle 14-11 Bedeutung der Einträge

Listeneintrag	Bedeutung
Name	Zeigt den Namen der Textliste.
ListRange	Gibt den Bereich der Textliste an: Nummer = Bitnummer (0-31) Range = Wert/Bereich Bit = Bit (0;1)
Comment	Beliebiger Kommentar zu der Textliste. Sie können maximal 500 Zeichen verwenden

Beispiel für das Tabellenblatt "TextListEntry"

	A	B	C	D	E	F
1	Name	Parent	DefaultEntry	Value	Text	FieldInfos
2	Text_list_entry_1	TLValue/Range	TRUE	0 - 1	Default entry TLValue/Range ->	
3	Text_list_entry_2	TLValue/Range		2 - 3	TLValue/Range = 2-3 ->	
4	Text_list_entry_3	TLValue/Range		1	TLValue Range - single value = 1 ->	
5	Text_list_entry_1	TLBit		0	TLBit = 0 ->	
6	Text_list_entry_2	TLBit		1	TLBit = 1 ->	
7	Text_list_entry_1	TLBitnumber	TRUE	0	Default entry TLBitnumber; ->	
8	Text_list_entry_2	TLBitnumber		0	TLBitnumber - Bitnumber 0 is set ->	
9	Text_list_entry_3	TLBitnumber		1	TLBitnumber - Bitnumber 1 is set ->	
10	Text_list_entry_4	TLBitnumber		2	TLBitnumber - Bitnumber 2 is set ->	
11	Text_list_entry_5	TLBitnumber		3	TLBitnumber - Bitnumber 3 is set ->	
12	Text_list_entry_1	TL1	TRUE	0 - 1	Default entry TL1	
13	Text_list_entry_2	TL1		1 - 3	TL1 Value between 1 - 3 ->	
14	Text_list_entry_3	TL1		4 - 6	TL1 Value between 4 - 6 ->	
15	Text_list_entry_4	TL1		7	TL1 Single value = 7 ->	
16	Text_list_entry_1	TL2	TRUE	0 - 1	<field ref="0" /> ->	<ref id = 0; type = CommonTextList; TextList = TL1; Tag = HMI_TL1control; Length = 100;>
17	Text_list_entry_2	TL2		1	TL2 Single value = 1 ->	
18	Text_list_entry_3	TL2		2 - 3	TL2 Range between 2 - 3 ->	
19	Text_list_entry_1	TLMultilined	TRUE	0 - 1	Default entry TLMultilined; last row	
20	Text_list_entry_2	TLMultilined		0 - 3	TLMultilined Value between 0-3\nwith test of"\n"	
21	Text_list_entry_1	TLall_1	TRUE	0 - 1	Default entry TLall_1	
22	Text_list_entry_1	TLall_2	TRUE	0 - 1	Default entry TLall_2	

Tabelle 14-12 Bedeutung der Einträge

Listeneintrag	Bedeutung
Name	Zeigt den Namen des Textlisteneintrags.
Parent	Zeigt den Namen der zugehörigen Texliste.
DefaultEntry	Gibt an, ob der Textlisteneintrag ein Standardeintrag ist. Der Standardeintrag erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt.
Value	Gibt die ganzzahligen Werte bzw. Wertebereiche einer Variablen an, die den Texteinträge aus der Textliste zugeordnet sind.
Text	Zeigt den Textlisteneintrag. Die Bezeichnung des Feldes enthält eine Sprachkennung. Textlisteneinträge müssen für den Import mit einer Sprachkennung versehen sein. Wenn ein dynamischer Paramter im Textlisteneintrag vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Textlisteneintrag zugeordnet.
FeldInfo	Gibt an, ob dynamische Parameter in der Textliste enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt. Beispiel für dynamische Parameter: Variable: <ref id = 0; type = CommonTagDisplayFormat; Tag = tag 1; DisplayType = Decimal; DisplayFormat = 9;> Textliste: <ref id = 1; type = CommonTextList; TextList = Textliste_1; Tag = tag 2; Length = 5;> Steuerungsvariable: <ref id = 0; type = CommonControlTagDisplayFormat; DisplayType = Decimal; DisplayFormat = 9;>

Siehe auch

Textlisten exportieren (Seite 2078)

14.3.6 Projekttexte importieren und exportieren

14.3.6.1 Projekttexte importieren

Sie bearbeiten die xlsx-Datei oder geben sie an einen Übersetzer weiter. Nach abgeschlossener Übersetzung importieren Sie die übersetzten Texte. Die Fremdsprachen werden im Projekt an das entsprechende Objekt importiert.

Hinweis

In WinCC importieren Sie die zuvor exportierten Projekttexte nur in dasselbe Projekt. Der Import in ein anderes Projekt wird nicht unterstützt.

Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Klicken Sie auf die Schaltfläche .Der Dialog "Import" wird geöffnet.
4. Wählen Sie aus dem Feld "Importdatei" den Pfad und den Dateinamen der Importdatei aus.
5. Aktivieren Sie das Optionskästchen "Quellsprache importieren", wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
6. Klicken Sie auf "Importieren".

Ergebnis

Sie haben die Projekttexte importiert.

14.3.6.2 Projekttexte exportieren

Sie exportieren Projekttexte zur Übersetzung. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann z. B. in Microsoft Excel bearbeitet werden.

Sie können die Datei mit den Übersetzern austauschen und nach der Übersetzung direkt in das Projekt wieder importieren.

Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Selektieren Sie die Texte, die Sie exportieren möchten.
4. Klicken Sie auf die Schaltfläche .Der Dialog "Export" wird geöffnet.

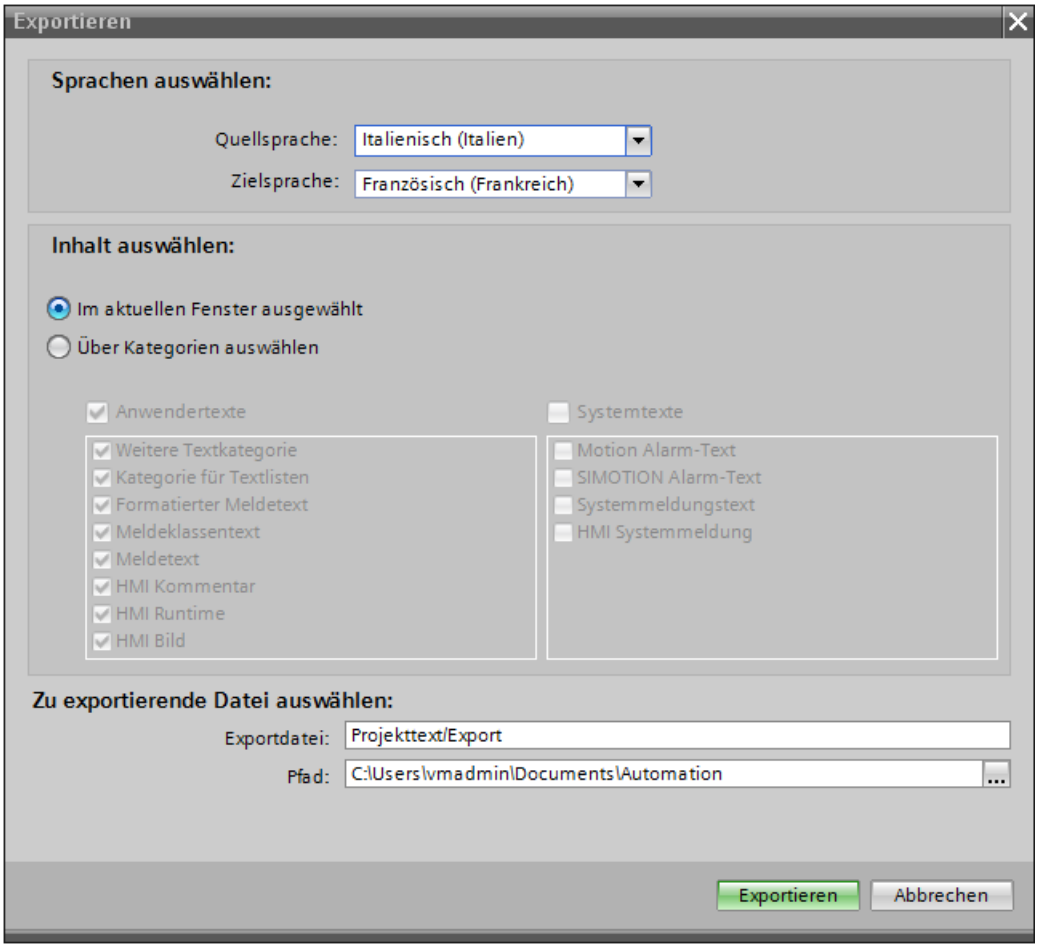

5. Wählen Sie aus der Klappliste "Quellsprache" die Sprache, aus der Sie die Übersetzung vornehmen möchten, z. B. Italienisch.
6. Wählen Sie aus der Klappliste "Zielsprache" die Sprache, in die Sie die Texte übersetzen möchten, z. B. Französisch.

7. Geben Sie im Eingabefeld "Exportdatei" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
8. Klicken Sie auf "Export".

Ergebnis

Die im Editor "Projekttexte" selektierten Texte wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Alternativ können Sie auch alle Projekttexte über Kategorien auswählen und exportieren. Aktivieren Sie im Dialog "Export" die Optionen "Anwendertexte" oder "Systemtexte", abhängig davon, welche Texte Sie exportieren möchten. Der Export kann in diesem Fall zusätzlich nach Kategorien eingegrenzt werden.

Hinweis

Projekttexte in Bibliotheksobjekten können nicht exportiert werden.

14.4 Querverweise nutzen

14.4.1 Allgemeine Informationen zu Querverweisen

Einleitung

Die Querverweisliste bietet einen Überblick über die Verwendung von Objekten und Geräten innerhalb des Projekts und der Projektbibliothek. Anhand der Querverweisliste lassen sich die Beziehungen und die Abhängigkeiten der Objekte untereinander anzeigen.

Nutzen der Querverweisliste

Die Querverweisliste bietet Ihnen folgende Vorteile:

- Bei der Projekterstellung und bei Änderungen behalten Sie die Übersicht über verwendete Geräte, Objekte, Variablen, Meldungen, Skripte usw.
- Von den Querverweisen aus können Sie direkt an die jeweilige Verwendungsstelle von Objekten springen.
- Sie sehen ob das jeweilige Objekt andere Objekte verwendet oder selbst verwendet wird.
- Bei der Fehlersuche erfahren Sie z. B. Folgendes:
  - Welche Objekte werden in welchem Bild und auf welchen Geräten verwendet.
  - Welche Meldungen und Rezepturen werden in welcher Anzeige angezeigt.
  - Welche Variable wird in welcher Meldung oder welchem Objekt verwendet.
- Als Teil der Projektdokumentation liefern die Querverweise einen umfassenden Überblick über alle verwendeten Objekte, Meldungen, Rezepturen, Variablen, Bilder usw.

Weitere Information zum Arbeiten mit der Querverweisliste finden Sie unter "Querverweise anzeigen".

Siehe auch

Grundlagen zum Bediengeräte-Assistenten (Seite 2011)

14.4.2 Besonderheiten beim Arbeiten mit Querverweisliste in WinCC

Mit Querverweisen in WinCC arbeiten

In WinCC gibt es einige Besonderheiten, die Sie beim Arbeiten mit Querverweisen berücksichtigen müssen.

Weitere Information zum Arbeiten mit der Querverweisliste finden Sie unter "Querverweise anzeigen".

Dialog "Querverweise ändern"

Über den Dialog "Objektverweise ändern" ändern Sie Referenzen von Variablen an den Eigenschaften von Bildobjekten.

Textuelle Querverweise beim Arbeiten mit VB-Scripting nutzen

Beim Arbeiten mit VB-Scripting können Sie mit Hilfe von textuellen Querverweisen Variablen direkt referenzieren, ohne auf die SmartTags-Auflistung zuzugreifen.

Neues Objekt in der Querverweisliste hinzufügen

Die Autovervollständigung bietet nur HMI-Variablen zur Auswahl an. Arrayelemente einer HMI-Variablen werden in der Autovervollständigung nicht angezeigt. Sie können Arrayelemente nur hinzufügen, indem Sie den Namen des Arrayelements eingeben.

Querverweise im Inspektorfenster anzeigen

Wenn Sie im Bild ein Objekt auswählen und eine HMI-Variable an dem Objekt als Prozessvariable verwendet wird, werden in den Querverweisen das Objekt und die verschaltete HMI-Variable angezeigt. Zusätzlich werden alle Verwendungsstellen des Objekts und der HMI-Variablen aufgelistet.

Wenn die HMI-Variable mit einer PLC-Variablen oder einer DB-Variablen verschaltet ist, werden auch die Verwendungsstellen der verschalteten PLC-Variablen bzw. DB-Variablen angezeigt.

Kontextmenü in der Querverweisliste

Das Kontextmenü ist objektbezogen. Das Kontextmenü ist dann für die jeweilige Zeile verfügbar, wenn in der Spalte "Objekt" tatsächlich ein Objekt enthalten ist.

14.4.3 Textuelle Querverweise

Einleitung

Textuelle Querverweise sind Querverweise, die als reiner Text hinterlegt sind. Mit Hilfe von textuellen Querverweisen können Sie nicht zu einem anderen Objekt navigieren. In der Querverweisliste werden solche Querverweise rot dargestellt.

Textuelle Querverweise entstehen folgenderweise:

- Sie legen einen textuellen Querverweis für ein nicht existierendes Objekt manuell an, indem Sie den Objektnamen unter Eigenschaft eintragen.
- Das verwendete Objekt wurde bereits gelöscht und die Verbindung dazu wird als textueller Querverweis dargestellt.

Grundlagen

Wenn Referenzen und untelagerte Objekte eines Quellobjekts gelöscht werden, werden sie in textuelle Querverweise umgewandelt. Wenn das Quellobjekt selbst gelöscht wird, wird kein textueller Querverweis hinterlegt.

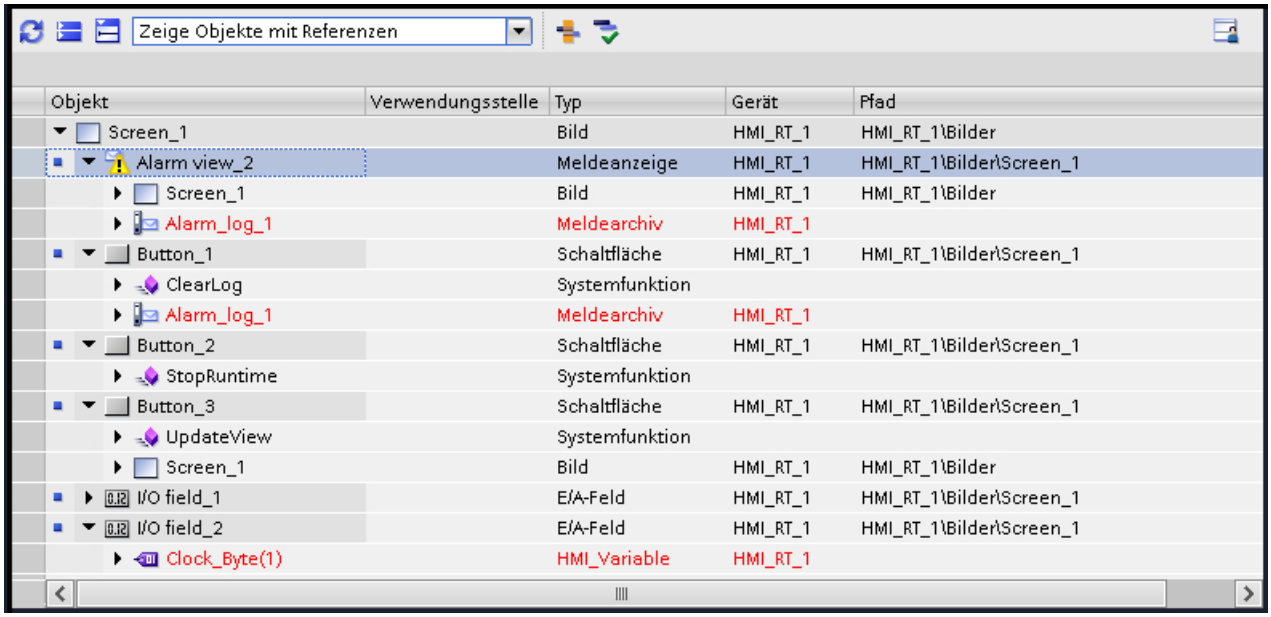

Hinweis

Nahezu alle Objekte unterstützen textuelle Querverweise mit Ausnahme von Variablenarchiven.

Hinweis

Textuelle Querverweise können nicht gelöscht werden.

Sie können einen textuellen Querverweis auch manuell anlegen, um eine Verbindung zu einem nicht existierenden Objekt herzustellen. Diese Vorgehensweise kann hilfreich sein, wenn Sie in VB-Scripting ohne SmartTags auf Variablen direkt zugreifen möchten.

Hinweis

Textuelle Querverweise für Variablen in RT Professional

Wenn einem Bild kein Anwenderdatentyp (UDT) zugewiesen ist bzw. die Variablen als Anwenderdatentyp-Element mit @NOP, @NOTP, @NOSP referenziert sind, erscheinen für die Variablen textuelle Querverweise.

Textuelle Querverweise beim Arbeiten mit VB-Scripting nutzen

Beim Arbeiten mit VB-Scripting können Sie mit Hilfe von textuellen Querverweisen Variablen direkt referenzieren, ohne auf die SmartTags-Auflistung zuzugreifen.

Wenn Sie einen Objektnamen eingeben, sucht der Skript-Parser das referenzierte Objekt.

Zum Beispiel: `DIM x`

```
x = MyObject
```

Wenn unter diesem Namen ein VB-Skript oder eine Variable existiert, wird darauf referenziert. Wenn unter diesem Namen ein VB-Skript und eine Variable existieren, wird der Skript-Parser die Referenz zum VB-Skript herstellen.

Wenn weder eine Variable noch ein VB-Skript unter diesem Namen existieren, legt das System zwei textuelle Querverweise an: einen für die Variable und einen für das VB-Skript.

Dem ersten Objekt, das unter diesem Namen angelegt wird (eine Variable oder ein VB-Skript), wird der eine von den beiden textuellen Querverweisen zugewiesen. Der textuelle Querverweis wird zum richtigen Querverweis. Der zweite textuelle Querverweis wird gelöscht.

Hinweis

In RT Professional wird nur ein textueller Querverweis für das VB-Skript angelegt, da direkte Adressierung von Variablen in diesem Zusammenhang nicht möglich ist.

Siehe auch

Ungültige Querverweise (Seite 2089)

14.4.4 Ungültige Querverweise

Einleitung

Bedingt durch bestimmte Projektierungsänderungen kann ein Querverweis ungültig werden. Ungültige Querverweise werden in der Querverweisliste und im Inspektorfenster ausgegraut dargestellt. Neben dem Querverweistext erscheint ein Fragezeichen. Wenn Sie auf das Fragezeichen klicken, öffnet sich die Onlinehilfe.

Ungültige Querverweise haben keine negativen Auswirkungen auf Ihr Projekt.

Grundlagen

Ein Querverweis kann aus folgenden Gründen ungültig werden:

- Änderung des Gerätetyps
- Kopieren eines Querverweises aus einem anderen Bediengerät auf ein Bediengerät, das dieses Objekt oder diese Objekteigenschaft nicht unterstützt
- Änderung einer Objekteigenschaft bzw. Zuweisung von ungültigen Werten

In der Querverweisliste können folgende Objekte ungültig sein:

- Quellobjekt und Zielobjekt
- Unterlagerte Objekte und ihre übergeordneten Objekte
- Verwendungsstelle

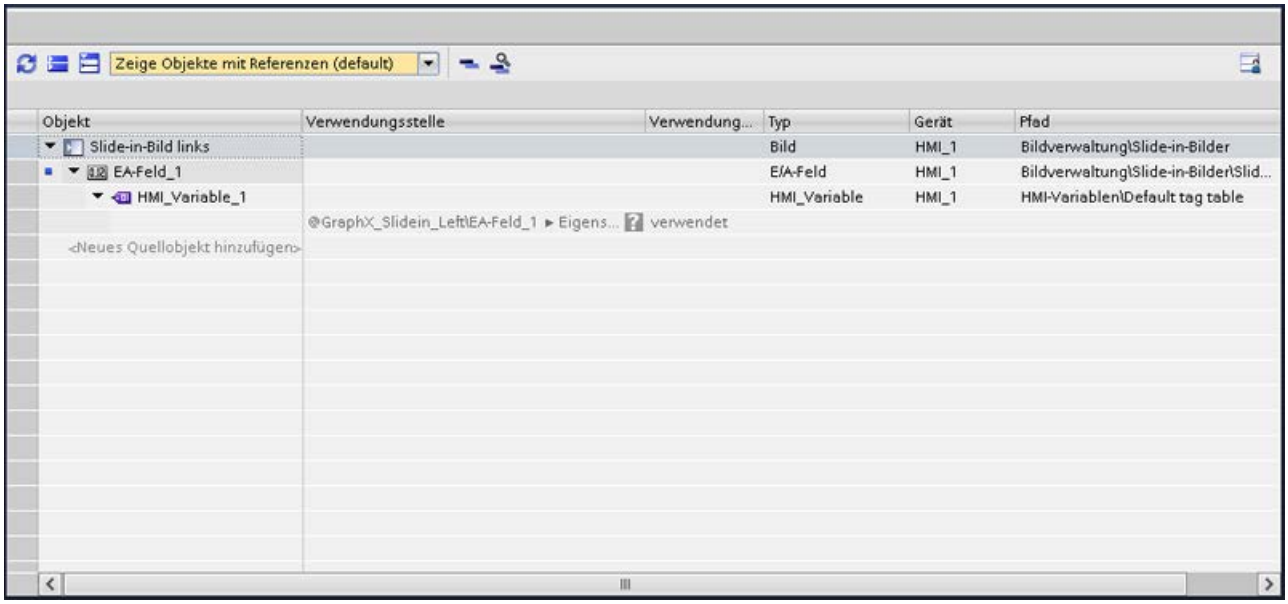

Wenn ein Objekt oder eine Verwendungsstelle ungültig ist, können Sie zu diesem Objekt oder dieser Verwendungsstelle nicht navigieren. Sie können die Eigenschaften einer ungültigen Verwendungsstelle nicht ändern. Bei ungültigen Objekten können Sie über das Kontextmenü die Befehle "Löschen" "Eigenschaften" aufrufen. Alle anderen Kontextmenübefehle sind nicht verfügbar. Bei ungültigen Verwendungsstellen ist kein Kontextmenü verfügbar.

Hinweis

Textueller Querverweis ungültig

Ein Objekt kann gleichzeitig als ein ungültiges Objekt und ein textueller Querverweis in der Querverweisliste fungieren. In diesem Fall wird der ungültige Querverweis wie ein textueller Querverweis rot dargestellt.

Beispiele

Quellobjekt ungültig

Sie haben ein Bild für ein Comfort Panel angelegt und eine VB-Scripting-Referenz in die Funktionsliste eingebaut. Sie ändern den Gerätetyp von "SIMATIC Comfort Panel" auf "SIMATIC Basic Panel". Da das VB-Scripting für Basic Panels nicht verfügbar ist, ist der Querverweis für das VB-Skript ungültig.

Zielobjekt ungültig

Sie haben für ein Comfort Panel ein Bild angelegt, das über einen Querverweis auf das Audit Trail-Archiv über die Systemfunktion "LöscheArchiv" verfügt. Sie kopieren dieses Bild in

Runtime Professional. Da das Audit Trail-Archiv für Runtime Professional nicht verfügbar ist, ist der Querverweis ungültig.

Eigenschaft, die das Quellobjekt und das Zielobjekt miteinander verbindet, ist ungültig

Sie haben in Runtime Professional einen Eintrag in der Grafikliste erstellt, der über einen Querverweis auf die Eigenschaft "Blinken" verfügt. Sie kopieren diese Grafikliste auf ein Comfort Panel. Da diese Eigenschaft auf Control Panels nicht verfügbar ist, ist der Querverweis von der Grafikliste zur Eigenschaft "Blinken" ungültig.

Eigenschaft ungültig, wenn Quellobjekt ungültig

Sie haben für ein Comfort Panel ein Slide-in-Bild abgelegt, das über einen Querverweis auf eine Variable über ein E/A-Feld verfügt. Sie ändern den Gerätetyp von "SIMATIC Comfort Panel" auf "SIMATIC Basic Panel". Da das Slide-in-Bild auf Basic Panels nicht verfügbar ist, ist der Querverweis ungültig.

Textueller Querverweis ungültig

Sie haben für ein Comfort Panel eine Variable angelegt und eine VB-Funktion "SetzeVariable" mit der angelegten Variablen verbunden. Sie haben ebenfalls ein Bild angelegt, das über Ereignis "Aufgebaut" mit der VB-Funktion "SetzeVariable" verbunden ist. Sie löschen die VB-Funktion. Für die gelöschte VB-Funktion wird ein textueller Querverweis erstellt. Sie ändern den Gerätetyp von "SIMATIC Comfort Panel" auf "SIMATIC Basic Panel". Da das VB-Scripting für Basic Panels nicht verfügbar ist, ist der textuelle Querverweis für das VB-Skript ungültig. Der ungültige Querverweis wird rot dargestellt.

Siehe auch

Textuelle Querverweise (Seite 2087)

14.4.5 Variablen in den Bildern umverdrahten

Einleitung

Über den Dialog "Objektverweise ändern" ändern Sie Referenzen von Variablen an den Eigenschaften von Bildobjekten.

Mit dieser Funktion können Sie eine große Anzahl von Variablen an verschiedenen Bildobjekten austauschen. Das Austauschen von Referenzen ist hilfreich, wenn Sie z. B. ein Bild per Copy&Paste angelegt haben und seine Referenzen anpassen müssen.

Sie können einzelne oder mehrere Zeichen in den Variablennamen suchen und ersetzen.

Den Dialog "Objektverweise ändern" öffnen Sie über den Befehl im Kontextmenü. Sie können den Dialog für einzelne oder mehrere Bildobjekte oder ein Bild öffnen.

Der Dialog "Objektverweise ändern" ist für folgende Bilder verfügbar:

- Übersichtsbild (Permanentbereich)
- Vorlagen
- Pop-up-Bilder

- Slide-in-Bilder
- Globales Bild

So ersetzen Sie eine Variable

1. Wählen Sie ein Objekt im Bild, das eine Variable enthält, die Sie ändern wollen.
2. Wählen Sie im Kontextmenü oder im Menü "Bearbeiten" den Befehl "Objektverweise ändern".
   Der Dialog "Objektverweise ändern" öffnet sich. In der Spalte "Objekt" sehen Sie die ausgewählten Bildobjekte und die verwendeten Variablen.

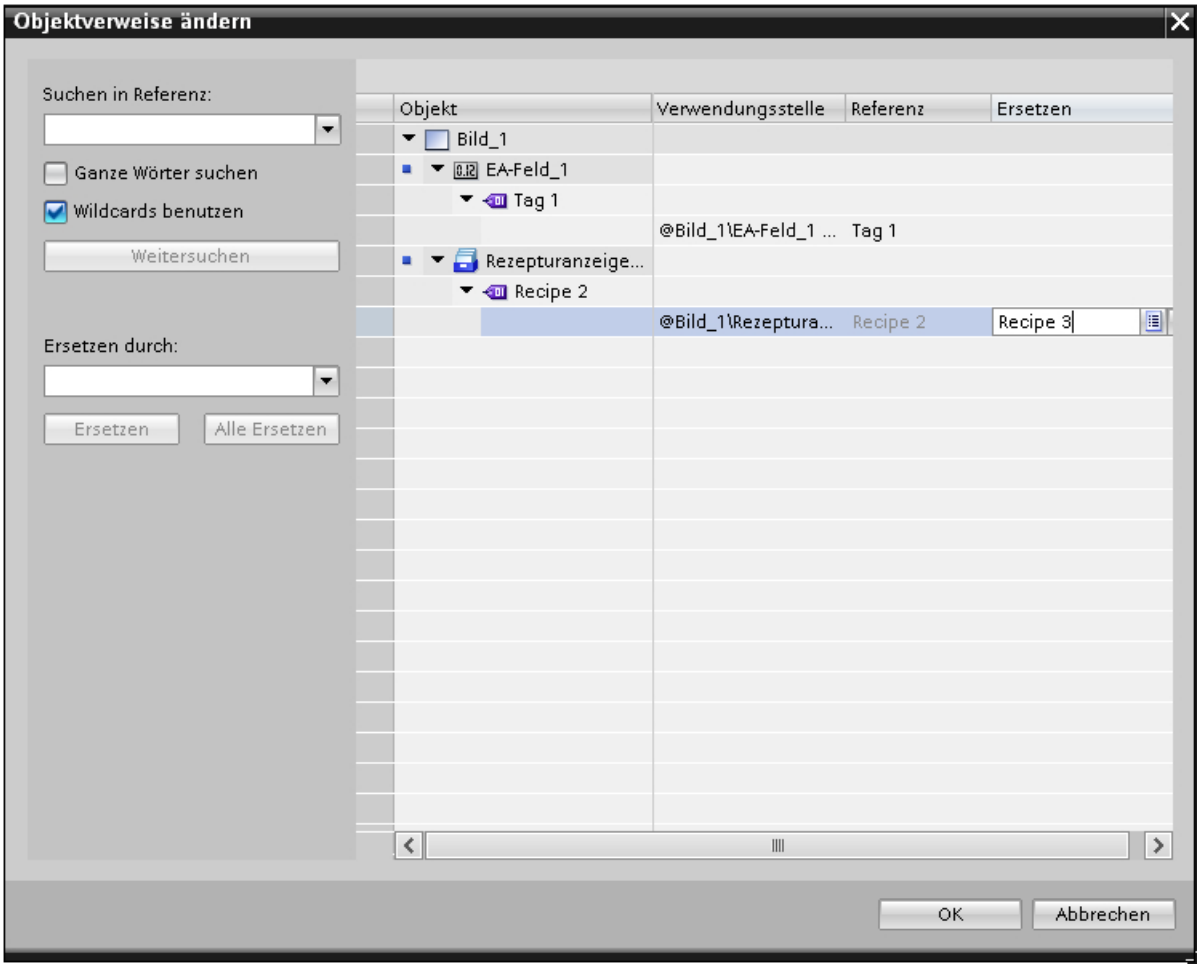

3. Geben Sie im Eingabefeld "Suchen in Referenz" den Namen oder den Teil des Namens ein, den Sie suchen wollen.
4. Mit der Schaltfläche "Weitersuchen" starten Sie den Suchvorgang. Wenn eine Referenz gefunden wird, wird sie in der Spalte „Referenz“ markiert.
   Durch einen weiteren Klick auf die Schaltfläche „Weitersuchen“ fahren Sie mit dem Suchvorgang fort bis die Liste vollständig durchsucht wurde.
   Danach startet der Suchvorgang wieder vom Anfang der Liste.
5. Geben Sie im Eingabefeld "Ersetzen durch" den Namen oder den Teil des Namens ein, mit dem Sie den gefundenen Namen ersetzen wollen.

6. Klicken Sie auf die Schaltfläche "Ersetzen" um die aktuell ausgewählte Variable zu ersetzen.
   Mit der Schaltfläche "Alles Ersetzen" werden alle gefundenen Referenzen ersetzt. So können Sie eine große Anzahl von Referenzen durch eine andere Referenz ersetzen.
7. Um eine Variable aus der Liste aller im Projekt angelegten Variablen auszuwählen, klicken Sie in der Spalte "Ersetzen" auf die Schaltfläche .
8. Um eine Variable aus der Objektliste auszuwählen oder eine neue Variable anzulegen, klicken Sie in der Spalte "Ersetzen" auf die Schaltfläche .
9. Bestätigen Sie Ihre Eingabe mit "OK".

Die Umverdrahtung der Variablen wird in das Projekt übernommen.

Alternativ können Sie in der Spalte "Ersetzen" direkt eine Variable eingeben, durch die die existierende Variable ersetzt werden soll. Dabei werden Sie durch die Autovervollständigung unterstützt.

Hinweis

Wenn Sie in der Spalte "Ersetzen" nach einer Variablen suchen, werden Sie durch die Autovervollständigung unterstützt. Nach der Eingabe der ersten Buchstaben des Variablennamens in der Spalte "Ersetzen", zeigt die Autovervollständigungsliste die Variablen an, deren Namen diese Buchstaben enthalten. Die Autovervollständigungsliste enthält kontextabhängig nur diejenigen Variablen, die für den vorgesehenen Zweck zulässig sind.

Hinweis

Die in der Spalte angezeigten Variablen werden als schreibgeschützt angezeigt, wenn das Bezugsobjekt schreibgeschützt ist, z. B. eine Variable in einem Bild, das in der Bibliothek abgelegt wurde.

Hinweis

Es ist möglich, eine Variable, die noch nicht existiert, manuell anzulegen. Diese Variable legen Sie nachträglich an.

Zusatzoptionen für die Suche

Sie können die Suche verfeinern, indem Sie folgende Optionen aktivieren:

- Wildcards benutzen
  Geben Sie * für eine beliebige Anzahl von Zeichen ein. Beispiel: Sie Suchen nach allen Variablen, die mit "HMI" beginnen. Geben Sie "HMI*" in das Suchfeld ein.
  Geben Sie ? für ein einzelnes Zeichen ein.
- Ganze Wörter suchen
  Um ausschließlich nach vollständigen Namen zu suchen, aktivieren Sie die Option "Ganze Wörter suchen".

Ersetzen von Arrays und anwenderdefinierten Variablen

Variablen, deren Namen Sonderzeichen . [ ] enthalten, werden in Anführungszeichen dargestellt.

Variablen vom Typ Array und Variablen von anwenderdefinierten Datentypen (UDT) enthalten Sonderzeichen um die Variable von ihren Elementen zu trennen. Sie werden aber nicht in Anführungszeichen dargestellt. Um ein Array oder eine UDT durch eine HMI-Variable mit Sonderzeichen zu ersetzen, geben Sie den Namen der HMI-Variable in Anführungszeichen in das Eingabefeld "Ersetzen" ein. Wenn Sie den Namen ohne Anführungszeichen eingeben, wird eine Referenz auf ein nicht existierendes Array oder UDT erstellt, anstatt auf die gewünschte HMI-Variable mit Sonderzeichen zu verweisen.

Korrigieren von Eingaben

Bei einer falschen Eingabe im Eingabefeld "Ersetzen durch" können die bestehenden Referenzen durch Referenzen auf nicht existierende Variablen ersetzt werden.

Wenn sie eine falsche Eingabe erkennen, bevor Sie mit "OK" bestätigen, können Sie den Fehler rückgängig machen. Dafür markieren Sie die Spalte "Ersetzen" und drücken die Taste "Entfernen". Alle falschen Eingaben in der Spalte "Referenz" werden gelöscht.

Wenn Sie den Dialog mit "OK" bestätigt haben, werden die betroffenen Eigenschaften der Bildobjekte rot markiert. Im Dialog "Querverweise ändern" werden nicht existierende Variablen rot markiert dargestellt.

Wenn die neue Variable noch nicht existiert, müssen Sie die Variable nachträglich anlegen.

14.5 Sprachen verwalten

14.5.1 Sprachbegriffe in WinCC

Oberflächensprache und Projektsprachen

In WinCC wird zwischen zwei Sprachebenen unterschieden:

- Oberflächensprache
  In der Oberflächensprache werden beim Projektieren die Texte in den Menüs und Dialogen von WinCC angezeigt. Die Oberflächensprache beeinflusst auch die Beschriftung der Bedienobjekte, die Parameter der Systemfunktionen, die Onlinehilfe usw.
- Projektsprachen
  Als Projektsprachen werden alle Sprachen bezeichnet, in denen ein Projekt später bearbeitet werden soll. Projektsprachen werden verwendet, um ein Projekt in mehreren Sprachen zu erstellen.

Beide Sprachebenen sind vollkommen unabhängig voneinander. Zum Beispiel können Sie mit einer deutschen Oberflächensprache jederzeit englische Projekte erstellen und umgekehrt.

Projektsprachen

Innerhalb der Projektsprachen werden folgende Sprachen unterschieden:

- Referenzsprache
  Die Referenzsprache ist die Sprache, in der Sie das Projekt zunächst projektieren. Während der Projektierung wählen Sie eine der Projektsprachen als Referenzsprache. Die Referenzsprache verwenden Sie als Vorlage für Übersetzungen. Sie erstellen zunächst alle Texte des Projekts in der Referenzsprache und übersetzen diese anschließend. Während Sie übersetzen, lassen Sie sich gleichzeitig die Texte der Referenzsprache anzeigen.
- Editiersprache
  In der Editiersprache erstellen Sie die Übersetzungen der Texte.
  Nachdem Sie Ihr Projekt in der Referenzsprache erstellt haben, übersetzen Sie die Texte in die übrigen Projektsprachen. Wählen Sie jeweils eine Projektsprache als Editiersprache und bearbeiten Sie die Texte für die entsprechende Sprachvariante. Sie können die Editiersprache jederzeit wechseln.

  Hinweis

  Beim Umschalten der Projektsprache wird auch die Tastenbelegung der Tastatur geändert. Bei einigen Sprachen (z. B. Spanisch) ist die Umstellung der Tastenbelegung betriebssystembedingt nicht möglich. In diesem Fall wird die Tastenbelegung auf Englisch umgeschaltet.

- Runtime-Sprachen
  Die Runtime-Sprachen sind die Projektsprachen, die auf das Bediengerät übertragen werden. Abhängig von den Erfordernissen Ihres Projekts entscheiden Sie, welche der Projektsprachen auf das Bediengerät übertragen werden.
  Damit der Bediener in Runtime zwischen diesen Sprachen umschalten kann, stellen Sie ihm entsprechende Bedienelemente zur Verfügung.

Siehe auch

Einstellungen für Sprachen im Betriebssystem (Seite 2097)

Einstellungen im Betriebssystem für asiatische Sprachen (Seite 2097)

Oberflächensprache wählen (Seite 2098)

Editor "Grafiksammlung" (Seite 2109)

Sprachen und Schriftarten in Runtime (Seite 2114)

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 2122)

Grundlagen zum Bediengeräte-Assistenten (Seite 2011)

14.5.2 Einstellungen für Sprachen im Betriebssystem

Einleitung

Die Einstellungen im Betriebssystem des Projektierungs-PCs haben in folgenden Bereichen Einfluss auf die Sprachverwaltung von WinCC:

- Auswahl der Projektsprachen
- Sprachabhängige Darstellung von Datumsangaben, Uhrzeiten, Geldbeträgen und Zahlen
- Darstellung von ASCII-Zeichen

Auswahl der Projektsprache

Eine Sprache steht nur dann als Projektsprache zur Verfügung, wenn sie im Betriebssystem installiert ist.

Sprachabhängige Darstellung von Datumsangaben, Uhrzeiten, Geldbeträgen und Zahlen

Die Darstellung von Datumsangaben und Uhrzeiten wird im Datum-Uhrzeit-Feld durch WinCC für die eingestellte Projektsprache und Runtime-Sprache fest vorgegeben.

Damit Datumsangaben, Uhrzeiten und Zahlen in der gewählten Bearbeitungssprache formell korrekt dargestellt werden, müssen Sie diese Sprache in der Systemsteuerung über die Ländereinstellungen einstellen.

Darstellung von ASCII-Zeichen

Bei Textausgabefeldern ist die Darstellung der ASCII-Zeichen ab 128 abhängig von der eingestellten Sprache und von dem verwendeten Betriebssystem.

Wenn auf verschiedenen PCs die gleichen Sonderzeichen dargestellt werden sollen, dann müssen auf den PCs das verwendete Betriebssystem und die verwendeten Ländereinstellung gleich sein.

Siehe auch

Sprachbegriffe in WinCC (Seite 2095)

Einstellungen im Betriebssystem für asiatische Sprachen (Seite 2097)

14.5.3 Einstellungen im Betriebssystem für asiatische Sprachen

Einstellungen auf westlichen Betriebssystemen

Wenn Sie asiatische Zeichen eingeben wollen, müssen Sie im Betriebssystem die Unterstützung für diese Sprache aktivieren.

Zur Projektierung von asiatischen Texten steht Ihnen unter Windows der Input Method Editor (IME) zur Verfügung. Ohne diesen Editor können Sie asiatische Texte nur anzeigen, aber nicht bearbeiten. Weitere Informationen zum Input Method Editor finden Sie in der Windows-Dokumentation. Zur Eingabe von asiatischen Zeichen bei der Projektierung wechseln Sie im "Input Methode Editor" zu der asiatischen Eingabemethode.

Um sprachabhängige Projekttexte wie Meldetexte im Simulator in asiatischen Zeichen angezeigen zu lassen, stellen Sie das Betriebssystem auf die gewünschte Sprache um.

Einstellungen auf asiatischen Betriebssystemen

Wenn Sie auf einem asiatischen Betriebssystem projektieren, müssen Sie für die Eingabe von ASCII-Zeichen, z. B. für Objektnamen, auf das englische Eingabegebietsschema wechseln. Da das englische Eingabegebietsschema in der Grundinstallation des Betriebssystems enthalten ist, brauchen Sie kein zusätzliches Eingabegebietsschema installieren.

Sprachunterstützung aktivieren

1. Öffnen Sie die Systemsteuerung.
2. Wählen Sie "Regions- und Sprachoptionen".
3. Aktivieren Sie auf der Registerkarte "Sprachen" das Kontrollkästchen "Dateien für ostasiatische Sprachen installieren".
4. Klicken Sie anschließend im Bereich "Textdienste und Eingabesprachen" auf die Schaltfläche "Details". Der Dialog "Textdienste und Eingabesprachen" wird geöffnet.
5. Auf der Registerkarte "Einstellungen" fügen Sie im Bereich "Installierte Dienste" das benötigte Eingabegebietsschema hinzu.
6. Auf der Registerkarte "Erweitert" wählen Sie im Bereich "Sprache für Programme, die Unicode nicht unterstützen" die Sprache des Betriebssystems aus.

Siehe auch

Sprachbegriffe in WinCC (Seite 2095)

Einstellungen für Sprachen im Betriebssystem (Seite 2097)

14.5.4 Projektsprachen einstellen

14.5.4.1 Oberflächensprache wählen

Einleitung

Die Oberflächensprache wird für die Anzeige von Menüeinträgen, Titelleisten, Hilfetexten, Texten in Dialogfenstern und weiteren Bezeichnungen in der Benutzeroberfläche von WinCC verwendet.

Während der Projektierung können Sie zwischen den installierten Oberflächensprachen wechseln. Die Bezeichnung der Bedienobjekte bleibt auch nach dem Wechsel der Oberflächensprache in der Sprache erhalten, die Sie beim Einfügen des Objekts eingestellt hatten.

Vorgehen

1. Wählen Sie im Menü "Extras > Einstellungen".
   Der Dialog "Einstellungen" wird geöffnet.
2. Wählen Sie unter "Allgemein > Allgemeine Einstellungen" die gewünschte Oberflächensprache aus.

Ergebnis

Die gewählte Sprache wird von WinCC als Oberflächensprache verwendet.

Siehe auch

Sprachbegriffe in WinCC (Seite 2095)

Projektsprachen aktivieren (Seite 2099)

Referenzsprache und Editiersprache wählen (Seite 2100)

14.5.4.2 Projektsprachen aktivieren

Einleitung

Im Editor "Projektsprachen" stellen Sie die Projektsprachen ein. Sie legen fest, welche der Projektsprachen als Referenzsprache und welche als Editiersprache dient.

Projektsprachen aktivieren

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".
   Im Arbeitsbereich werden die möglichen Projektsprachen angezeigt.
3. Aktivieren Sie die Sprachen, die als Projektsprachen zur Verfügung stehen sollen.

Hinweis

Kopieren mehrsprachig projektierter Objekte

Beim Kopieren eines mehrsprachig projektierten Objekts in ein anderes Projekt werden nur Texte in den Projektsprachen mitkopiert, die im Zielprojekt aktiviert sind. Aktivieren Sie im Zielprojekt alle Projektsprachen, deren Texte Sie beim Kopieren mit in das Zielprojekt übernehmen wollen.

Projektsprachen deaktivieren

1. Deaktivieren Sie die Sprachen, die nicht als Projektsprachen zur Verfügung stehen sollen.

ACHTUNG
Wenn Sie eine Projektsprache deaktivieren, werden im aktuellen Projekt alle in dieser Sprache bereits erstellten Texte und Grafiken gelöscht.

Siehe auch

Oberflächensprache wählen (Seite 2098)

Referenzsprache und Editiersprache wählen (Seite 2100)

14.5.4.3 Referenzsprache und Editiersprache wählen

Einleitung

Im Editor "Projektsprachen" stellen Sie die Projektsprachen ein. Sie legen fest, welche der Projektsprachen als Referenzsprache und welche als Editiersprache dient. Die Editiersprache können Sie jederzeit ändern.

Voraussetzungen

Der Editor "Projektsprachen" ist geöffnet.

Mehrere Projektsprachen sind aktiviert.

Referenzsprache und Editiersprache wählen

1. Klicken Sie im Bereich "Allgemein > Editiersprache" in der Auswahlliste auf das Pfeilsymbol.
2. Klicken Sie in der Auswahlliste auf die gewünschte Sprache z. B. Deutsch.
3. Klicken Sie im Bereich "Allgemein > Referenzsprache" in der Auswahlliste auf das Pfeilsymbol.
4. Klicken Sie in der Auswahlliste auf die gewünschte Sprache z. B. Englisch.

Die ausgewählte Sprache wird im Listenfeld angezeigt.

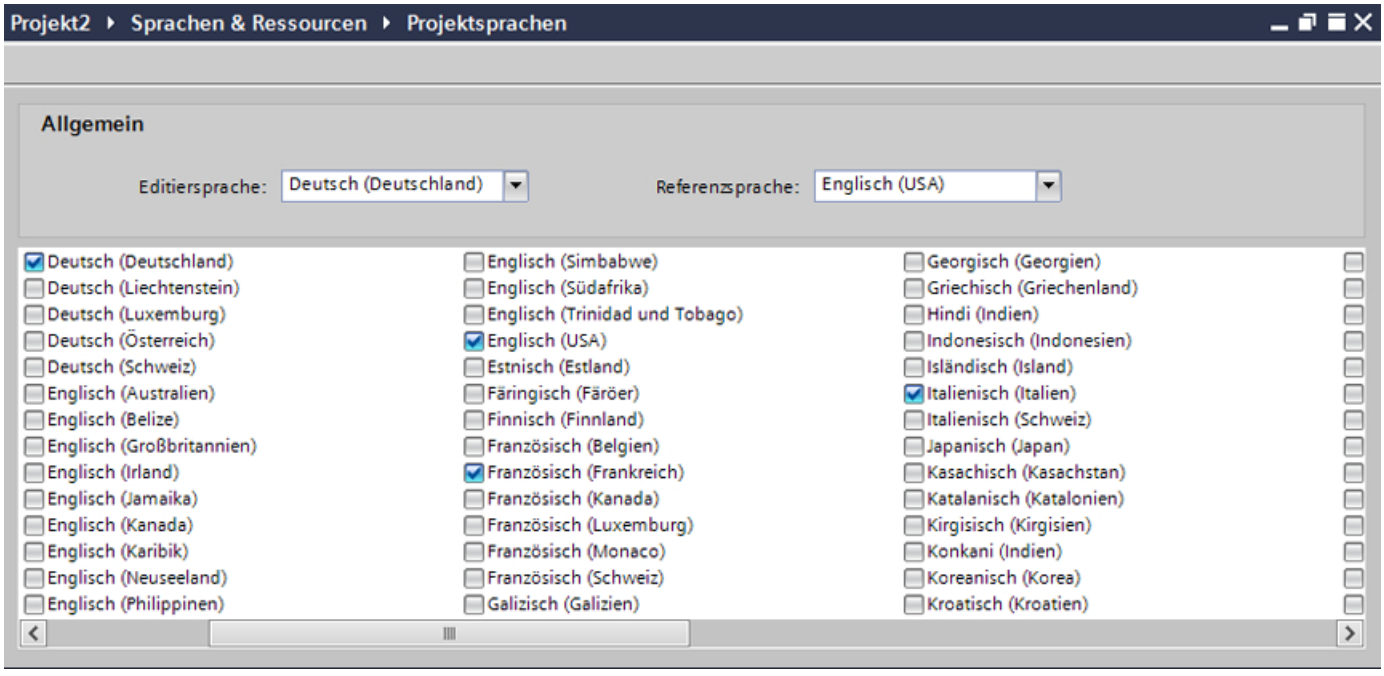

Ergebnis

Sie haben die Editiersprache und die Referenzsprache gewählt.

Wenn Sie die Editiersprache ändern, werden alle zukünftigen Texteingaben in der neuen Editiersprache gespeichert.

Siehe auch

Oberflächensprache wählen (Seite 2098)

Projektsprachen aktivieren (Seite 2099)

14.5.5 Ein Projekt in mehreren Sprachen erstellen

14.5.5.1 Arbeiten mit mehreren Sprachen

Mehrsprachige Projektierung in WinCC

Mit WinCC projektieren Sie Ihre Projekte auch in mehreren Sprachen. Die Erstellung eines Projekts in mehreren Sprachen kann verschiedene Ursachen haben:

- Sie setzen ein Projekt in mehreren Ländern ein.
  Sie erstellen das Projekt in mehreren Sprachen; bei der Inbetriebnahme des Bediengeräts wird nur die Sprache auf das Bediengerät übertragen, die von den Bedienern am jeweiligen Aufstellungsort gesprochen wird.
- Die Bediener einer Anlage sprechen verschiedene Sprache.
  Beispiel: Ein Bediengerät wird in China eingesetzt, das Servicepersonal versteht jedoch nur Englisch.

Projekttexte übersetzen

Mit WinCC können Sie Projekttexte in mehreren Sprachen in verschiedenen Editoren direkt eingeben, z.B. im Editor "Projekttexte". Weiterhin stellt Ihnen WinCC zum Übersetzen Ihrer Projektierung Export- und Importmöglichkeiten zur Verfügung. Dies ist besonders von Vorteil, wenn Sie Projekte mit einem hohen Textanteil projektieren und übersetzen lassen möchten.

Sprachverwaltung und Übersetzung in WinCC

Folgende Editoren dienen in WinCC der Verwaltung von Sprachen und zur Übersetzung von Texten:

Editor	Kurzbeschreibung
Projektsprachen	Auswahl der Projektsprachen, der Editiersprache und der Referenzsprache.
Sprachen und Schriften	Auswahl der Runtime-Sprachen und der am Bediengerät verwendeten Schriftarten.
Projekttexte	Zentrale Verwaltung der projektierten Texte in allen Projektsprachen.
Grafiken	Grafiksammlung für die Verwaltung der Grafiken und deren sprachabhängige Varianten.

Siehe auch

Grundlagen zu Projekttexten (Seite 2103)

Texte direkt übersetzen (Seite 2104)

Texte mithilfe von Referenztexten übersetzen (Seite 2106)

Projekttexte exportieren (Seite 2106)

Projekttexte importieren (Seite 2108)

14.5.5.2 Grundlagen zu Projekttexten

Texte in unterschiedlichen Sprachen im Projekt

Texte, die während der Prozessbearbeitung auf Anzeigegeräten ausgegeben werden, werden üblicherweise in der Sprache eingegeben, in der die Automatisierungslösung programmiert ist. Ebenso werden Kommentare und Bezeichnungen von Objekten nur in dieser Sprache eingegeben.

Beherrscht ein Bediener die Sprache nicht, benötigt er eine Übersetzung aller bedienerrelevanten Texte in seine Muttersprache. Sie können daher sämtliche Texte in jede beliebige Sprache übersetzen. So können Sie sicherstellen, dass jeder, der zu einem späteren Zeitpunkt mit den Texten im Projekt konfrontiert wird, diese Texte in der von ihm gewünschten Sprache angezeigt bekommt.

Anwendertexte und Systemtexte

Um die Übersichtlichkeit zu verbessern, wird zwischen Anwendertexten und Systemtexten unterschieden:

- Anwendertexte sind Texte, die vom Nutzer angelegt worden sind.
- Systemtexte sind Texte, die automatisch in Abhängigkeit von der Konfiguration im Projekt angelegt werden.

Die Projekttexte werden im Projekttexte-Editor verwaltet. Dieser ist in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte" zu finden.

Beispiele für mehrsprachige Projekttexte

Sie können z. B. die folgenden Textarten mehrsprachig verwalten:

- Anzeigetexte
- Meldungstexte
- Kommentare in Tabellen
- Beschriftungen von Bildobjekten
- Textlisten

Übersetzen von Texten

Um Texte zu übersetzen, gibt es zwei unterschiedliche Vorgehensweisen.

- Texte direkt übersetzen
  Sie können die Übersetzungen für die einzelnen Projektsprachen direkt im Editor "Projekttexte" eingeben.
- Texte mithilfe von Referenztexten übersetzen
  Für kleinere Textmengen können Sie die Editiersprache wechseln. Während die Texte der Referenzsprache eingeblendet bleiben, geben Sie in die neuen Texte in der Editiersprache ein.

Siehe auch

Arbeiten mit mehreren Sprachen (Seite 2102)

Texte direkt übersetzen (Seite 2104)

Texte mithilfe von Referenztexten übersetzen (Seite 2106)

Projekttexte exportieren (Seite 2106)

Projekttexte importieren (Seite 2108)

14.5.5.3 Texte direkt übersetzen

Texte übersetzen

Wenn Sie in Ihrem Projekt mehrere Sprachen verwenden, können Sie einzelne Texte direkt übersetzen. Sobald Sie die Sprache der Softwareoberfläche ändern, stehen die übersetzten Texte in der jeweiligen Sprache zur Verfügung.

Voraussetzungen

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Sie haben mindestens zwei weitere Projektsprache gewählt.

Vorgehen

Um einzelne Texte zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte".
   Im Arbeitsbereich erscheint eine Liste mit den Texten im Projekt. Für jede Projektsprache existiert eine eigene Spalte.

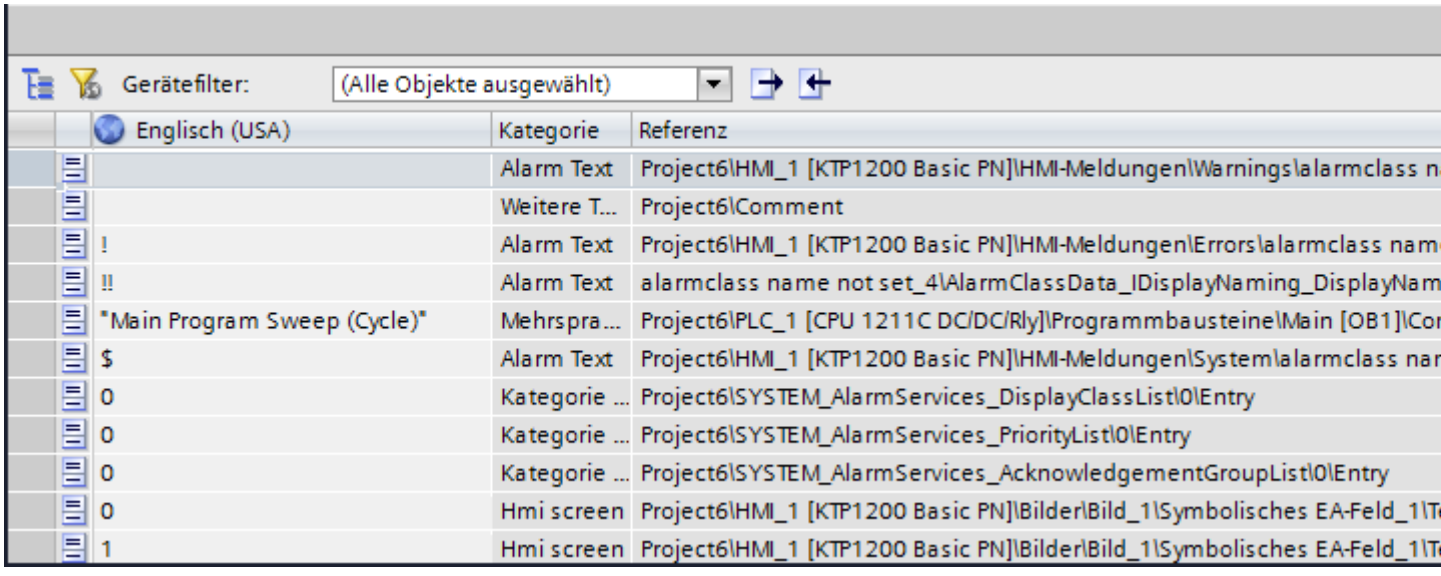

3. Um identische Texte zu gruppieren und sie auf einmal zu übersetzen, klicken Sie in der Funktionsleiste auf die Schaltfläche .
4. Um Texte mit fehlenden Übersetzungen auszublenden, klicken Sie in der Funktionsleiste auf die Schaltfläche .
5. Klicken Sie in eine leere Spalte und geben Sie die Übersetzung ein.

Ergebnis

Sie haben einzelne Texte im Editor "Projekttexte" übersetzt. In Runtime werden die die Texte in der jeweiligen Runtime-Sprache angezeigt.

Siehe auch

Arbeiten mit mehreren Sprachen (Seite 2102)

Grundlagen zu Projekttexten (Seite 2103)

Projekttexte exportieren (Seite 2106)

Projekttexte importieren (Seite 2108)

14.5.5.4 Texte mithilfe von Referenztexten übersetzen

Einführung

Nach dem Wechseln der Editiersprache werden alle Texte in Eingabefeldern in der neuen Editiersprache angezeigt. Sind für diese Sprache noch keine Übersetzungen vorhanden, sind die Eingabefelder leer oder mit den Standardwerten befüllt.

Wenn Sie in ein entsprechendes Eingabefeld erneut Text eingeben, wird dieser in der aktuellen Editiersprache gespeichert. Im Anschluss existieren für dieses Eingabefeld Texte in zwei Projektsprachen - in der ehemaligen Editiersprache und in der aktuellen Editiersprache. Auf diese Weise ist es möglich, Texte in mehreren Projektsprachen anzulegen.

Sie können sich bei einem Eingabefeld bereits bestehende Übersetzungen in anderen Projektsprachen anzeigen lassen. Diese dienen als Vergleich für die Texteingabe in der aktuellen Editiersprache und werden als Referenzsprache bezeichnet.

Voraussetzung

Für ein Eingabefeld existiert mindestens eine Übersetzung in einer weiteren Projektsprache.

Vorgehen

Um sich für ein Eingabefeld die Übersetzung in einer Referenzsprache anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Task Card "Aufgaben > Sprachen & Ressourcen ".
2. Wählen Sie aus der Klappliste "Referenzsprache" eine Referenzsprache aus.

Ergebnis

Die Referenzsprache ist voreingestellt. Wenn Sie in ein Textfeld klicken, werden in der Task Card "Aufgaben > Referenztext" Übersetzungen angezeigt, die bereits in anderen Projektsprachen vorhanden sind.

Siehe auch

Arbeiten mit mehreren Sprachen (Seite 2102)

Grundlagen zu Projekttexten (Seite 2103)

Projekttexte exportieren (Seite 2106)

Projekttexte importieren (Seite 2108)

14.5.5.5 Projekttexte exportieren

Sie exportieren Projekttexte zur Übersetzung. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann z. B. in Microsoft Excel bearbeitet werden.

Sie können die Datei mit den Übersetzern austauschen und nach der Übersetzung direkt in das Projekt wieder importieren.

Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Deutsch und Englisch.

Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Selektieren Sie die Texte, die Sie exportieren möchten.
4. Klicken Sie auf die Schaltfläche .Der Dialog "Exportieren" wird geöffnet.

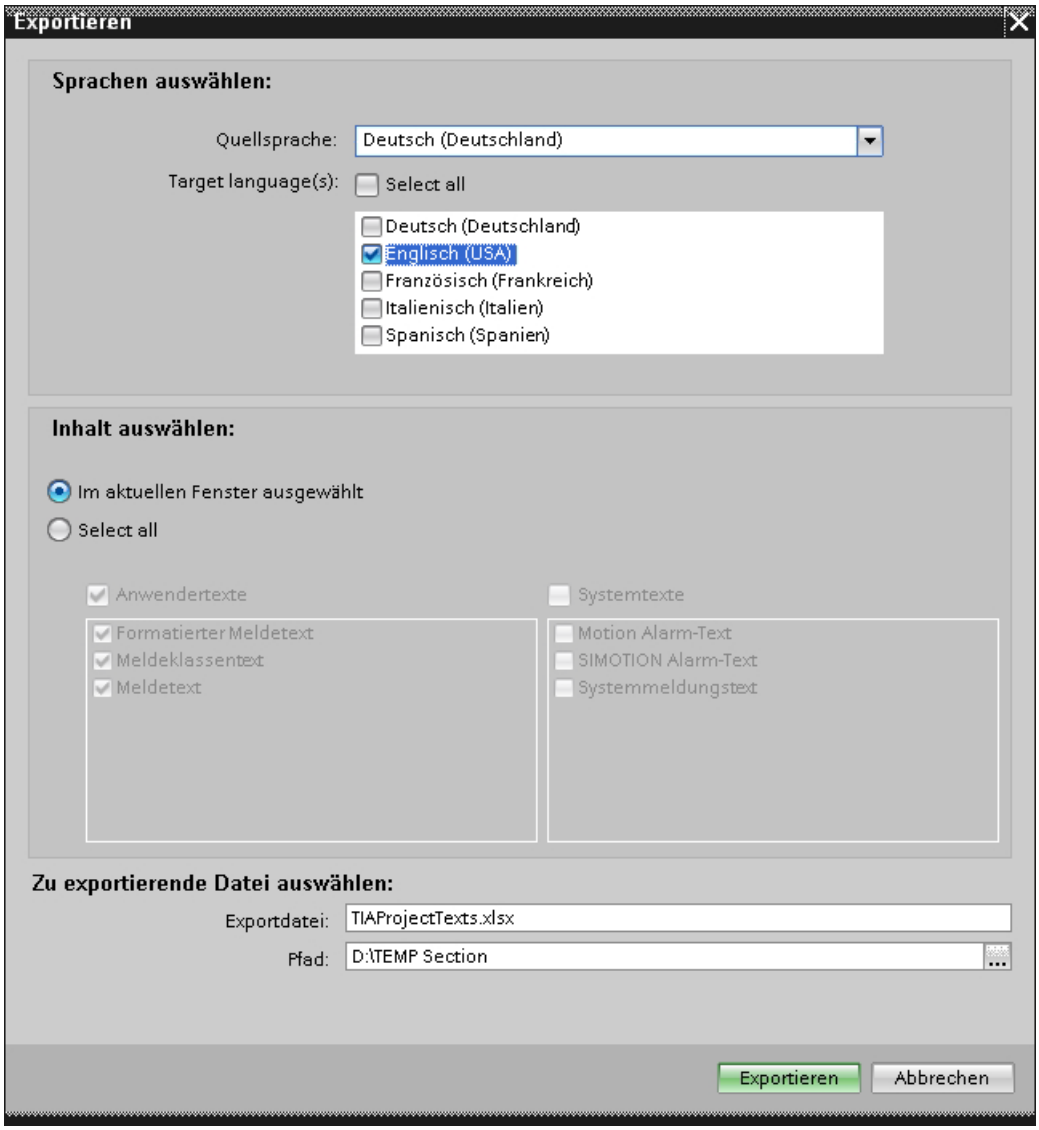

5. Wählen Sie die gewünschte Quellsprache aus.
6. Wählen Sie die gewünschten Zielsprachen aus.
   Wenn Sie alle Zielsprachen exportieren wollen, wählen Sie die Option "Alles auswählen".
7. Geben Sie der Exportdatei den gewünschten Namen.
8. Stellen Sie den gewünschten Pfad zum Ablageort der Exportdatei ein.
9. Klicken Sie auf "Exportieren".

Ergebnis

Die im Editor "Projekttexte" selektierten Texte wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Alternativ können Sie auch alle Projekttexte über Kategorien auswählen und exportieren. Aktivieren Sie im Dialog "Exportieren" die Optionen "Anwendertexte" oder "Systemtexte", abhängig davon, welche Texte Sie exportieren möchten. Der Export kann in diesem Fall zusätzlich nach Kategorien eingegrenzt werden.

Hinweis

Projekttexte in Bibliotheksobjekten können nicht exportiert werden.

Siehe auch

Arbeiten mit mehreren Sprachen (Seite 2102)

Grundlagen zu Projekttexten (Seite 2103)

Texte direkt übersetzen (Seite 2104)

Texte mithilfe von Referenztexten übersetzen (Seite 2106)

Projekttexte importieren (Seite 2108)

14.5.5.6 Projekttexte importieren

Sie bearbeiten die xlsx-Datei oder geben sie an einen Übersetzer weiter. Nach abgeschlossener Übersetzung importieren Sie die übersetzten Texte. Die Fremdsprachen werden im Projekt an das entsprechende Objekt importiert.

Hinweis

In WinCC importieren Sie die zuvor exportierten Projekttexte nur in dasselbe Projekt. Der Import in ein anderes Projekt wird nicht unterstützt.

Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".
   Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Klicken Sie auf die Schaltfläche .Der Dialog "Import" wird geöffnet.
4. Wählen Sie aus dem Feld "Importdatei" den Pfad und den Dateinamen der Importdatei aus.
5. Aktivieren Sie das Optionskästchen "Quellsprache importieren", wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
6. Klicken Sie auf "Importieren".

Ergebnis

Sie haben die Projekttexte importiert.

Siehe auch

Arbeiten mit mehreren Sprachen (Seite 2102)

Grundlagen zu Projekttexten (Seite 2103)

Texte direkt übersetzen (Seite 2104)

Texte mithilfe von Referenztexten übersetzen (Seite 2106)

Projekttexte exportieren (Seite 2106)

14.5.6 Sprachabhängige Grafiken einsetzen

14.5.6.1 Editor "Grafiksammlung"

Einleitung

Im Editor "Grafiksammlung" verwalten Sie die projektierten Grafikobjekte in mehreren Sprachversionen. Mehrsprachige Projekte erfordern teilweise auch sprachabhängige Varianten der Grafiken, z. B. wenn:

- Die Grafiken Text enthalten,
- In den Grafiken kulturelle Gegebenheiten eine Rolle spielen.

Öffnen des Editors "Grafiksammlung"

Doppelklicken Sie in der Projektnavigation auf "Sprachen und Ressourcen > Grafiksammlung".

Arbeitsbereich

Der Arbeitsbereich zeigt Ihnen alle projektierten Grafikobjekte in einer Tabelle. Für jede Projektsprache ist eine eigene Tabellenspalte angelegt. Die Tabellenspalten enthalten die Grafikvarianten einer Sprache.

Zusätzlich können Sie für jede Grafik eine Standard-Grafik festlegen, die immer dann angezeigt wird, wenn es für eine Projektsprache keine sprachspezifische Grafik gibt.

Vorschau

In der Vorschau lassen Sie sich die Darstellungen der Grafiken auf verschiedenen Bediengeräten in einer Vorschau anzeigen.

Siehe auch

Sprachbegriffe in WinCC (Seite 2095)

Grafik in der Grafiksammlung hinterlegen (Seite 2110)

Externe Grafik in der Grafiksammlung hinterlegen (Seite 2112)

14.5.6.2 Grafik in der Grafiksammlung hinterlegen

Einleitung

Mit dem Editor "Grafiken" importieren Sie Grafiken, die Sie in Bildobjekten im Editor "Bilder" verwenden. Zusätzlich verwalten Sie die für sprachabhängige Grafikvarianten. Eine Vorschau zeigt Ihnen die Darstellungen einer Grafik auf verschiedenen Bediengeräten an.

Hinweis

Dateinamen für sprachabhängige Grafiken

In den Dateinamen sprachabhängiger Grafiken ist die Groß- und Kleinschreibung relevant. Achten Sie für allen Sprachen auf eine einheitliche Schreibweise.

Voraussetzung

- Die sprachabhängigen Varianten einer Grafik stehen zur Verfügung.
- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Der Editor "Grafiken" ist geöffnet.

Grafik einfügen

1. Klicken Sie in der Tabelle "Grafiksammlung" auf "Hinzufügen". Ein Dialog wird geöffnet.
2. Wählen Sie die gewünschte Grafikdatei.
3. Klicken Sie im Dialogfenster auf "Öffnen".
   Die Grafik wird in das Projekt eingefügt und im Editor "Grafiken" in allen Zellen dieser Zeile angezeigt.
4. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
5. Wählen Sie im Kontextmenü den Eintrag "Grafik einfügen". Ein Dialog wird geöffnet
6. Wählen Sie die gewünschte Grafikdatei und klicken Sie auf "Öffnen".
   An Stelle der Grafik der Referenzsprache wird die sprachabhängige Variante in die Tabelle eingefügt.
7. Fügen Sie abschließend in die Spalte "Standardgrafik" eine Grafik ein, die in Runtime für diejenigen Sprachen angezeigt wird, für die keine sprachspezifische Grafik vorhanden ist.

Alternativ können Sie auch per Drag&Drop eine Grafik aus dem Windows Explorer, an die gewünschte Stelle in der Tabelle "Grafiksammlung" ziehen.

Grafiken in der Bediengerätevorschau anzeigen

1. Klicken Sie in der Tabelle auf eine Grafik.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Grafikeinstellungen > Vorschau Bediengerät" das gewünschte Bediengerät aus.
   Die Grafik wird Ihnen in der Vorschau so angezeigt, wie es in Runtime am ausgewählten Bediengerät dargestellt wird.

Ergebnis

Die eingefügten Grafiken stehen im Editor "Grafiken" zur Verfügung. Beim Editieren wird die Grafik angezeigt, die der jeweiligen Editiersprache zugeordnet ist. In allen Editiersprachen, für die kein Bild importiert wurde, wird das Standardbild angezeigt.

In Runtime werden die der jeweiligen Runtime-Sprache zugeordneten Bilder angezeigt. In allen Runtime-Sprachen, für die kein Bild importiert wurde, wird das Standardbild angezeigt.

Hinweis

Wenn Sie eine Projektsprache deaktivieren, werden im aktuellen Projekt alle in dieser Sprache bereits erstellten Grafiken gelöscht.

Siehe auch

Editor "Grafiksammlung" (Seite 2109)

14.5.6.3 Externe Grafik in der Grafiksammlung hinterlegen

Einleitung

Um in einem externen Grafikprogramm erstellte Grafiken in Ihren Bildern anzuzeigen, hinterlegen Sie diese vorher in der Grafiksammlung des WinCC Projekts.

Voraussetzung

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Der Editor "Grafiken" ist geöffnet.
- Eine Grafik ist im Editor "Grafiken" vorhanden.

Neue Grafik als OLE-Objekt erstellen und einfügen

1. Klicken Sie in der Tabelle "Grafiksammlung auf "Hinzufügen". Ein Dialog wird geöffnet.
2. Navigieren Sie zum Ordner, in dem die Grafik gespeichert ist.
3. Klicken Sie im Dialogfenster auf "Öffnen".
   Die Grafik wird in das Projekt eingefügt und im Editor "Grafiken" in allen Zellen dieser Zeile angezeigt.
4. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
5. Wählen Sie im Kontextmenü den Eintrag "Objekt einfügen". Der Dialog "Objekt einfügen" wird geöffnet.

   Hinweis

   Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

6. Wählen Sie im Dialog "Objekt einfügen > Neu erstellen" und einen Objekttyp.
7. Klicken Sie auf "OK". Das dazugehörende Grafikprogramm wird geöffnet.
8. Wenn Sie die Grafik erstellt haben, schließen Sie das Grafikprogramm.
   Die Grafik wird im Standardformat des Grafikprogramms abgespeichert und in der Grafiksammlung angezeigt.

Erstellte Grafik in WinCC einfügen

1. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
2. Wählen Sie im Kontextmenü den Eintrag "Objekt einfügen". Der Dialog "Objekt einfügen" wird geöffnet.

   Hinweis

   Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

3. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
4. Klicken Sie auf die Schaltfläche "Durchsuchen".
5. Navigieren Sie zu der erstellten Grafik und wählen Sie diese aus.

   Hinweis

   Um Grafikdateien zu importieren, beachten Sie folgende Größenbeschränkungen:

   *.bmp, *.tif, *.emf, *.wmf ≤4 MByte

   *.jpg, *.jpeg, *.ico, *.gif "*≤1 MByte

Ergebnis

Die eingefügten OLE-Objekt stehen im Editor "Grafiken" zur Verfügung.

Im Editor "Bilder" wird die Grafikvariante der aktuell eingestellten Editiersprache angezeigt. In allen Editiersprachen, für die kein Bild importiert wurde, wird die Standardgrafik angezeigt.

n Runtime wird jeweils die Grafikvariante der aktuellen Runtime-Sprache angezeigt. In allen Runtime-Sprachen, für die keine Grafik eingefügt wurde, wird die Standardgrafik angezeigt.

Die in der Grafiksammlung hinterlegten OLE-Objekte öffnen Sie durch Doppelklicken mit dem dazugehörigen Grafikprogramm, um sie zu bearbeiten.

Siehe auch

Editor "Grafiksammlung" (Seite 2109)

14.5.7 Sprachen in Runtime

14.5.7.1 Sprachen und Schriftarten in Runtime

Mehrere Runtime-Sprachen einsetzen

Sie bestimmen, welche Projektsprachen in Runtime auf einem bestimmten Bediengerät zum Einsatz kommen. Die Anzahl der Runtime-Sprachen, die gleichzeitig auf dem Bediengerät zur Verfügung stehen, ist geräteabhängig. Um dem Bediener die Sprachumschaltung in Runtime zu ermöglichen, müssen Sie ein entsprechendes Bedienelement projektieren.

Bei Runtime-Start wird das Projekt in der zuletzt eingestellten Sprache angezeigt. Beim ersten Runtime-Start erscheint die Sprache mit der niedrigsten Nummer in "Reihenfolge für Sprachumschaltung".

Runtime-Einstellung "Sprache & Schriftart"

Unter "Sprache & Schriftart" projektieren Sie Folgendes:

- Projektsprachen, welche als Runtime-Sprachen für den jeweiligen Bediengerätetyp zur Verfügung stehen
- Reihenfolge, in der die Sprachen bei einem Sprachwechsel umgeschaltet werden
- Standardschriftart für eine Runtime-Sprache

Zusätzlich wird im Bereich "Schriftart herunterladen" der Lizenzstatus der Schriftarten angezeigt. Wenn für eine geschützte Schriftart eine spezielle Lizenz erforderlich ist, wird es in der Spalte "Lizenzstatus" angezeigt.

Hinweis

Schriftarten nachinstallieren

Wenn eine projektierte Schriftart auf dem Projektierungsrechner fehlt, wird das Projekt nicht übersetzt. Eine Nachinstallation ist erforderlich.

Starten Sie nach der Installation einer Schriftart das TIA Portal neu und übersetzen Sie Ihr Projekt vollständig. Erst dann wird das Bediengerät fehlerfrei übersetzt.

Hinweis

Runntime-Sprache in Controls (RT Professional)

Wenn Sie im Engineering System ein Control in einer spezifischen Runtime-Sprache projektiert haben, die keine Oberflächen-Sprache ist, kann es vorkommen, dass manche Texte, z.B. Systemtexte in der Statuszeile, in englischer Sprache erscheinen.

Hinweis

Runtime-Sprachen als Eingabesprachen festlegen

Um Daten in Runtime eingeben und bearbeiten zu können, konfigurieren Sie die festgelegten Runtime-Sprachen auch als Eingabesprachen für die Tastatur von Ihrem PC.

Siehe auch

Sprachbegriffe in WinCC (Seite 2095)

Methoden der Sprachumschaltung (Seite 2115)

Runtime-Sprache aktivieren (Seite 2116)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 2117)

Standardschrift für eine Runtime-Sprache einstellen (Seite 2119)

Archivsprache wählen (Seite 2120)

14.5.7.2 Methoden der Sprachumschaltung

Einleitung

Damit auf dem Bediengerät mehrere Runtime-Sprachen zur Verfügung stehen, projektieren Sie eine Sprachumschaltung. Nur so ist es dem Bediener möglich, zwischen verschiedenen Runtime-Sprachen zu wählen.

Methoden der Sprachumschaltung

Sie können folgende Methoden der Sprachumschaltung projektieren:

- Direkte Sprachwahl
  Jede Sprache wird über eine eigene Schaltfläche eingestellt. In diesem Fall erstellen Sie für jede Runtime-Sprache eine Schaltfläche.
- Sprachumschaltung
  Der Bediener schaltet die Sprachen über eine Schaltfläche um.

Die Beschriftungen der Schaltflächen müssen bei beiden Methoden in jede verwendete Sprache übersetzt werden. Sie können zusätzlich ein Ausgabefeld projektieren, das die gerade gewählte Sprache anzeigt.

Siehe auch

Sprachen und Schriftarten in Runtime (Seite 2114)

Runtime-Sprache aktivieren (Seite 2116)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 2117)

Standardschrift für eine Runtime-Sprache einstellen (Seite 2119)

Archivsprache wählen (Seite 2120)

14.5.7.3 Runtime-Sprache aktivieren

Einleitung

Der Editor "Sprache & Schriftart" zeigt Ihnen alle im Projekt verfügbaren Projektsprachen an. Sie wählen, welche Projektsprachen als Runtime-Sprachen auf dem Bediengerät zur Verfügung stehen sollen.

Voraussetzungen

Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart".
3. Aktivieren Sie die folgenden Sprachen:
   - Englisch
   - Französisch
   - Italienisch

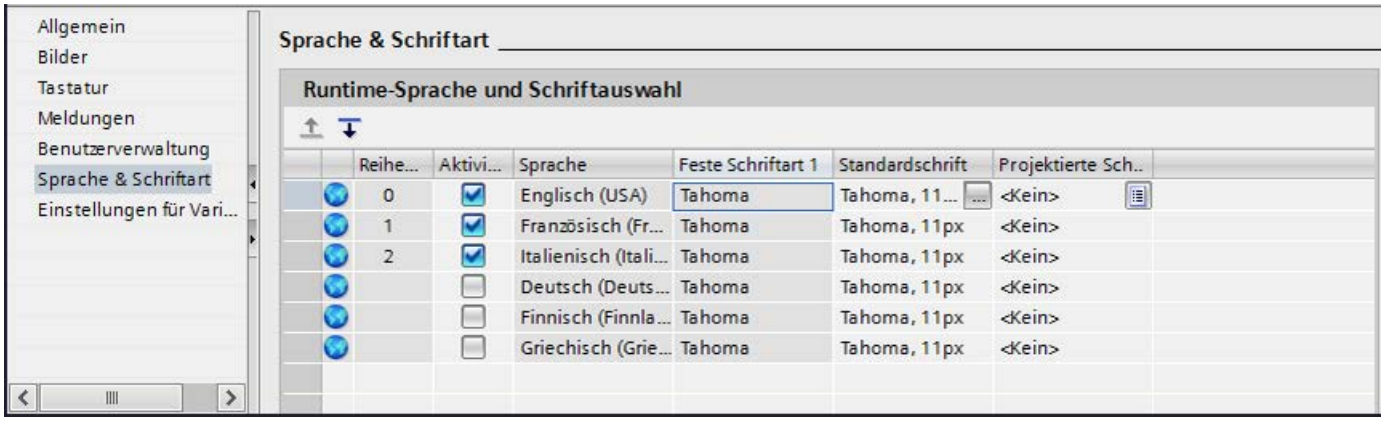

Ergebnis

Sie haben drei Runtime-Sprachen aktiviert. In der Spalte "Reihenfolge" wird für jede Sprache automatisch eine Nummer vergeben. Die aktivierten Runtime-Sprachen werden mit dem übersetzten Projekt auf das Bediengerät geladen.

Wenn Sie mehr Sprachen auswählen, als auf das Bediengerät übertragen werden können, werden Sie durch eine farbige Hinterlegung in der Tabelle darauf aufmerksam gemacht.

Siehe auch

Sprachen und Schriftarten in Runtime (Seite 2114)

Methoden der Sprachumschaltung (Seite 2115)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 2117)

Archivsprache wählen (Seite 2120)

14.5.7.4 Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen

Einleitung

Sie legen fest, in welcher Reihenfolge die Runtime-Sprachen bei einem Sprachwechsel umgeschaltet werden. Beim ersten Runtime-Start erscheint das Projekt in der Sprache mit der niedrigsten Nummer in der Spalte "Reihenfolge".

Voraussetzungen

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Der Editor "Sprache & Schriftart" ist geöffnet und drei Runtime-Sprachen sind aktiviert in folgender Reihenfolge:
  1. Englisch
  2. Italienisch
  3. Französisch

Vorgehen

1. Selektieren Sie die Runtime-Sprache "Englisch".
2. Klicken Sie auf das Schaltfläche . Die Runtime-Sprache "Englisch" wird nach unten verschoben. In der Spalte "Reihenfolge" wird die Nummer automatisch auf "1" geändert.

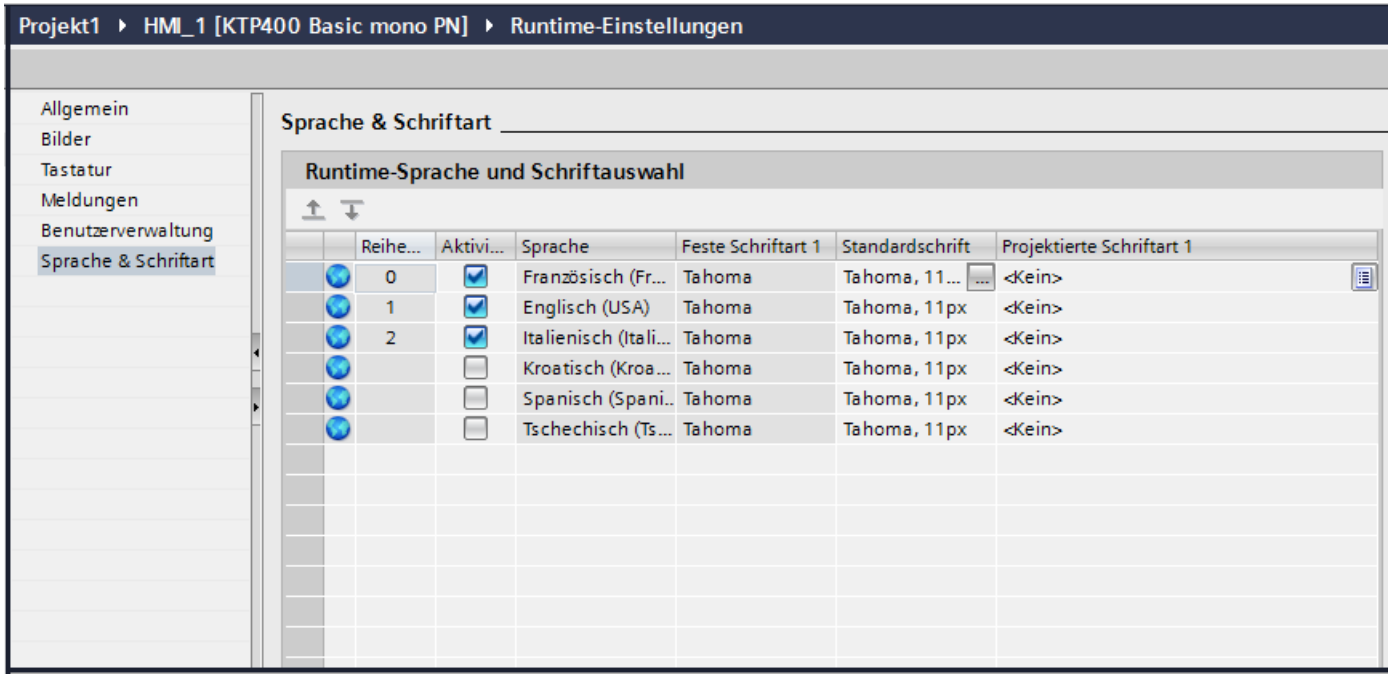

Ergebnis

Sie haben die Reihenfolge der Runtime-Sprachen geändert. Beim ersten Start von Runtime erscheint das Projekt in der Sprache mit der niedrigsten Nummer. Bei einer Sprachumschaltung wechselt die Sprache in der Reihenfolge der vergebenen Nummern.

Siehe auch

Sprachen und Schriftarten in Runtime (Seite 2114)

Methoden der Sprachumschaltung (Seite 2115)

Runtime-Sprache aktivieren (Seite 2116)

Standardschrift für eine Runtime-Sprache einstellen (Seite 2119)

Archivsprache wählen (Seite 2120)

14.5.7.5 Standardschrift für eine Runtime-Sprache einstellen

Einleitung

Im Editor "Sprache & Schriftart" legen Sie für jede Runtime-Sprache fest, in welcher Schriftart die Texte auf dem Bediengerät angezeigt werden. Die Standardschriftart wird für alle Texte verwendet, bei denen Sie keine Schriftart festlegen können, z. B. Dialogtexte.

WinCC bietet hierbei nur die Schriftarten zur Auswahl an, die vom Bediengerät unterstützt werden.

Hinweis

Schriftstile "Fett" und "Kursiv"

Die Schriftstile "Fett" und "Kursiv" werden in Runtime nicht auf allen Bediengeräten unterstützt.

Voraussetzungen

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Im Editor "Sprache & Schriftart" sind drei Runtime-Sprachen aktiviert.
  1. Chinesisch
  2. Deutsch
  3. Französisch

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart". Die Tabelle zeigt die aktivieren Runtime-Sprachen und die Schriftarten.
3. Klicken Sie in der Zeile für "Französich" auf die Spalte "Standardschrift".
4. Wählen Sie die Schriftart aus, die standardmäßig verwendet wird, wenn Sie für einen Text keine Schriftart wählen können.

Ergebnis

Die Projekttexte der Runtime-Sprache "Französisch" werden auf dem Bediengerät in der gewählten Schriftart angezeigt.

Diese Schriftarten werden beim Transfer zusätzlich zum Bediengerät übertragen.

Die Standardschriftart wird auch für die Darstellung der Dialoge aus dem Betriebssystem des Bediengeräts verwendet. Wenn Dialogtexte oder Dialogüberschriften nicht vollständig dargestellt werden, wählen Sie eine kleinere Schriftgröße für die Standardschriftart.

Siehe auch

Sprachen und Schriftarten in Runtime (Seite 2114)

Methoden der Sprachumschaltung (Seite 2115)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 2117)

Archivsprache wählen (Seite 2120)

14.5.7.6 Schriftart für alle Sprachen vereinheitlichen

Einleitung

Die Option "Gleiche Schriftart für alle Sprachen verwenden" ermöglicht es Ihnen, die Schriftart für alle Projektsprachen beim Projektieren zu vereinheitlichen.

Voraussetzung

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Im Editor "Sprache & Schriftart" sind mehrere Runtime-Sprachen aktiviert.
- Für die aktivierten Runtime-Sprachen ist dieselbe Schriftart unter "Projektierte Schriftart" festgelegt.

Vorgehen

1. Im Menü "Extras > Einstellungen > Visualisierung > Allgemein" aktivieren Sie das Optionsfeld "Gleiche Schriftart für alle Sprachen verwenden".

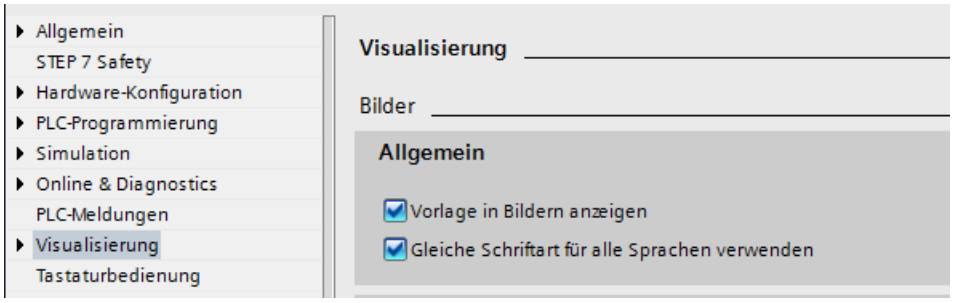

Ergebnis

Sie haben die Option "Gleiche Schriftart für alle Sprachen verwenden" aktiviert. Wenn Sie beim Projektieren die Schriftart eines Objekts in einer Sprache ändern, wird diese Schriftart für alle aktiven Sprachen übernommen.

14.5.7.7 Archivsprache wählen

Einleitung

Im Editor "Runtime-Einstellungen > Allgemein" wählen Sie die Sprache, die verwendet wird, um Archive in Runtime zu schreiben.

Voraussetzungen

- Im Editor "Projektsprachen" sind die Sprachen aktiviert, die Sie in Ihrem Projekt verwenden z. B. "Deutsch" und "Englisch".

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart".
3. Aktivieren Sie die Runtime-Sprachen z. B. "Deutsch" und "Englisch".
4. Legen Sie die "Reihenfolge" fest:
   - 1 Deutsch
   - 2 Englisch
5. Klicken Sie "Runtime-Einstellungen > Allgemein".
6. Wählen Sie für "Archive > Archivierungssprache" die Sprache "Deutsch".

Ergebnis

Nach dem Laden startet das Projekt in der Runtime-Sprache "Deutsch". Die Archive werden in Deutsch geschrieben. Während Runtime schaltet der Bediener die Runtime-Sprache auf Englisch. Die Archive werden jedoch weiterhin in Deutsch geschrieben.

Siehe auch

Sprachen und Schriftarten in Runtime (Seite 2114)

Methoden der Sprachumschaltung (Seite 2115)

Runtime-Sprache aktivieren (Seite 2116)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 2117)

Standardschrift für eine Runtime-Sprache einstellen (Seite 2119)

14.5.7.8 Besonderheiten asiatischer und östlicher Sprachen in Runtime

Einleitung

Beachten Sie folgende Besonderheiten für den Betrieb in Runtime bei Projekten für asiatische Sprachen.

Hinweis

Bei der Projektierung verwenden Sie ausschließlich die asiatischen Schriftarten, die Ihr Projektierungsrechner unterstützt.

Speicherbedarf für asiatische Zeichensätze

Der Speicherbedarf ist bei Verwendung von asiatischen Sprachen größer. Achten Sie daher beim Übersetzen des Projekts auf entsprechende Fehlermeldungen.

Schriftgröße für asiatische Zeichensätze

Wenn Sie ein Projekt für asiatische Sprachen erstellen, verwenden Sie für die Darstellung in Runtime mindestens Schriftgröße 10 Punkt. Bei der Verwendung einer kleineren Schriftgröße werden asiatische Schriftzeichen unleserlich. Dies gilt auch für die Standardschriftart in den Runtime Einstellungen unter "Sprache & Schriftart".

Textfeldlänge bei asiatischen Sprachen

Wenn Sie mehrsprachige Projekte mit asiatischen Sprachen bearbeiten, achten Sie auf die Länge von Textfeldern. Abhängig von der Schriftart und -größe werden Felder unvollständig angezeigt.

1. Öffnen Sie im Inspektorfenster des Textfeldes "Eigenschaften > Darstellung".
2. Deaktivieren Sie unter "Größenanpassung" die Option "Automatisch anpassen".
3. Prüfen Sie, ob die Anzeige in Runtime korrekt ist.

14.5.8 Beispiel zur mehrsprachigen Projektierung

14.5.8.1 Beispiel: Schaltfläche zur Sprachumschaltung projektieren

Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, mit der Sie in Runtime zwischen mehreren Runtime-Sprachen umschalten.

Voraussetzungen

- Sie haben das Beispiel "Mehrsprachige Beschriftung einer Schaltfläche projektieren" ausgeführt.
- Das Bild "Screen_1" ist geöffnet.
- Die Schaltfläche ist im Bild selektiert.

Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
2. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
3. Wählen Sie die Systemfunktion "SetzeSprache" und die Einstellung "Umschalten".

Ergebnis

Sie haben der Schaltfläche die Funktion "SetzeSprache" zugewiesen. Wenn Sie in Runtime die Schaltfläche drücken, dann schaltet die Runtime-Sprache um. Die Reihenfolge der Runtime-Sprachen wird so umgesetzt, wie sie im Editor "Sprache & Schriftart" durch die Nummerierung der Sprachen vorgegeben ist.

Siehe auch

Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren (Seite 2123)

Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren (Seite 2124)

Sprachbegriffe in WinCC (Seite 2095)

14.5.8.2 Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren

Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, die in Deutsch mit "Sprache umschalten" und in Englisch mit "Switch Language" beschriftet ist.

Voraussetzungen

- Im Editor "Projektsprachen" sind die Sprachen "Deutsch" und "Englisch" aktiviert.
- Deutsch ist als Editiersprache und Referenzsprache gewählt.
- Das Bild "Screen_1 "ist angelegt und geöffnet.
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Ziehen Sie eine Schaltfläche per Drag&Drop aus der Task Card "Werkzeuge" in das Bild. Die Schaltfläche wird im Bild eingefügt.
2. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Geben Sie unter "Text > Aus" die Beschriftung "Sprache umschalten" ein.
4. Bestätigen Sie die Eingabe mit <Return>. Die Schaltfläche ist beschriftet.
5. Öffnen Sie die Task Card "Aufgaben".
6. Wählen Sie "Sprachen & Ressourcen > Editiersprache" die Sprache "Englisch" aus.
7. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Text > Aus" die Beschriftung "Switch Language" ein.

Ergebnis

Die Beschriftung der Schaltfläche ist in deutscher und englischer Sprache projektiert. In Runtime wird die der aktuellen Runtime-Sprache entsprechende Beschriftung angezeigt.

Siehe auch

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 2122)

Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren (Seite 2124)

14.5.8.3 Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren

Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, die in Deutsch mit "Sprache umschalten" und in Englisch mit "Switch Language" beschriftet ist.

Voraussetzungen

- Im Editor "Projektsprachen" sind folgende Sprachen aktiviert
  - Deutsch
  - Englisch
  - Italienisch
- Alle Sprachen sind im Editor "Runtime-Einstellungen > Sprache & Schriftart" als Runtime-Sprache aktiviert.
- Das Bild "Screen_1 "ist angelegt und geöffnet.
- Im Bild sind drei Schalflächen angelegt:
  - Schaltfläche_1 mit der Beschriftung "Deutsch"
  - Schaltfläche_2 mit der Beschriftung "English"
  - Schaltfläche_3 mit der Beschriftung "Italiano"
- Das Inspektorfenster ist geöffnet.

Vorgehen

1. Selektieren Sie "Schalftläche_1".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
3. Klicken Sie in der Tabelle auf "<Funktion hinzufügen>".
4. Wählen Sie die Systemfunktion "SetzeSprache".
5. Klicken Sie auf das Feld "Umschalten".
6. Klicken Sie auf die Schaltfläche .
7. Wählen Sie "Runtime-Sprache". Das Feld wird rot markiert.
8. Wählen Sie in der Aufklappliste "Deutsch".
9. Wiederholen Sie die Schritte 1 - 8 für die anderen beiden Schlaltflächen und wählen Sie die entsprechende Runtime-Sprache aus.

Ergebnis

Sie haben drei Schaltflächen für die Sprachumschaltung in Runtime projektiert. Jede Schaltfläche schaltet in eine andere Runtime-Sprache um. Wenn Sie in Runtime z. B. auf die Schaltfläche "English" klicken, dann schalten Sie Runtime-Sprache auf Englisch um.

Siehe auch

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 2122)

Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren (Seite 2123)

14.6 Gerät tauschen

14.6.1 Grundlagen

Einleitung

Mit dem Tausch der Geräte können Sie bereits vorhandene Projektierungen für Ihre neuen Geräte verwenden und mit geringem manuellen Aufwand optimieren.

Alle von Ihnen projektierten Daten bleiben in den Projektierungsdaten enthalten. Daher ist es nicht erforderlich, einzelne Objekte eines Geräts zu kopieren und in ein anderes Gerät einzufügen.

Prinzip

Beim Tausch eines Geräts gelten folgende Grundsätze:

- Nur vom neuen Gerät unterstützte Funktionen sind verfügbar. Nur Projektierungsdaten, die vom neuen Gerät unterstützt werden, werden angezeigt.
  Betroffen sind z. B.
  - Rezepturen,
  - verfügbare Objekte in Bildern,
  - verfügbare Systemfunktionen,
  - verfügbare Kommunikationsprotokolle.
- Auf dem neuen Gerät kann die Anzahl der unterstützten Objekte z. B. Bilder oder Variablen begrenzt sein. Wenn die vorhandenen Objekte die Beschränkungen auf dem neuen Gerät überschreiten, werden die Objekte vollständig angezeigt. Die Objekte werden allerdings in den einzelnen Editoren farblich gekennzeichnet. Beim Übersetzen der Projektierungsdaten wird ein Fehler ausgegeben.
  Daher ist es erforderlich, dass Sie bei einem Tausch auf ein Gerät mit geringerem Leistungsumfang manuell nachbearbeiten.
  Beispiel: Mengenbeschränkung bei Verbindungen
  Wenn auf dem neuen Gerät weniger Verbindungen unterstützt werden als projektiert wurden, werden alle Verbindungen farbig als ungültig markiert. Löschen Sie überzählige Verbindungen.
- Nach einem Bediengerätewechsel sollten Sie das Aussehen der projektierten Bilder überprüfen. Aufgrund geänderter Displaygrößen kann sich Position und Aussehen von Bildobjekten, z. B. Rezepturanzeige und Meldeanzeige, verändern.

Hinweis

Wenn Sie ein Gerät z. B. ein Panel tauschen und als neues Gerät eine PC Station wählen, wird in der Projektnavigation WinCC Runtime Advanced automatisch unter die PC Station verschoben.

Hinweis

Steuerungsmeldungen nach einem Basic Panel-Gerätetausch

Damit nach einem Gerätetausch ("Gerätesicht > Gerät tauschen") von einem Basic Panel auf ein Comfort Panel, Mobile Panel oder auf WinCC RT Advanced die Steuerungsmeldungen (PLC-Überwachungsmeldungen, Programmmeldungen oder Systemmeldungen) angezeigt werden, müssen Sie unter "Runtime Einstellungen > Meldungen > Steuerungsmeldungen“ für die jeweiligen Verbindungen die Einstellung "Anzeigeklassen“ aktivieren.

Siehe auch

Geräteabhängige Funktionen (Seite 2128)

Beispiel: Gerät tauschen (Seite 2137)

Grundlagen zum Bediengeräte-Assistenten (Seite 2011)

Basic Panel (Seite 2455)

Mobile Panel (Seite 2462)

Comfort Panel (Seite 2467)

WinCC Runtime Advanced (Seite 2473)

Projektierung der Verbindungen anpassen (Seite 2127)

14.6.2 Projektierung der Verbindungen anpassen

Einleitung

Bei einem Bediengerätewechsel können Fehlermeldungen vom Typ "... wird in der neuen Konfiguration nicht unterstützt. Es wird entfernt." auftreten. Diese Meldungen beziehen sich auf projektierte Verbindungen des Geräts und werden z. B. durch eine unterschiedliche Anzahl von Schnittstellen der Bediengeräte ausgelöst. Nach dem Gerätewechsel werden diese Verbindungen rot markiert. Wenn Sie diese Verbindungen weiterhin benutzen möchten, müssen Sie die Projektierung der Verbindung anpassen.

Vorgehen

1. Öffnen Sie den Editor "Geräte und Netze".
2. Klicken Sie in der Funktionsleiste der Netzsicht auf "Vernetzen".
3. Vernetzen Sie die Schnittstelle des Bediengeräts mit der Schnittstelle der CPU.
4. Klicken Sie im tabellarischen Bereich der Netzsicht auf die Tabelle "Verbindungen".

5. Selektieren Sie die rot markierte Verbindung.
6. Stellen Sie im Inspektorfenster "Eigenschaften > Allgemein > Schnittstelle" die neue Schnittstelle ein.

Siehe auch

Grundlagen (Seite 2126)

14.6.3 Geräteabhängige Funktionen

14.6.3.1 Geräteabhängige Funktionen

Vom Gerät abhängige Funktionen

Vom Gerät abhängige Funktionen werden folgendermaßen umgesetzt:

- Farben
  Wenn Sie von einem Gerät mit Farbanzeige gegen ein Gerät tauschen, das einen geringeren Farbumfang besitzt, wird die Farbe automatisch geändert.
  Wenn Sie die Farbe manuell ändern und wieder zu einem Gerät mit größerem Farbumfang zurückwechseln, bleibt der reduzierte Farbumfang erhalten.
- Schriftarten
  Wenn eine projektierte Schriftart auf einem Gerät nicht vorhanden ist, wird sie durch eine ähnliche Schriftart oder die projektierte Standardschriftart ersetzt. Die Standardschriftart hängt vom ausgewählten Gerät ab.
- Schriftgröße
  Um den Text auf Geräten anzuzeigen, verwenden Sie kleine Windows-Schriftarten. Wenn Sie große Windows-Schriftart verwenden, dann wird, abhängig von der Größe des Displays, der Text nicht vollständig angezeigt.
  In asiatische Sprachen ist der Zeichenumfang deutlich höher. Daher wirkt sich die Verwendung unterschiedlicher Schriftgrößen massiv auf den Speicherbedarf auf allen Geräten aus.
  Für eine performante Projektierung verwenden Sie für alle großen Schriften im gesamten Projekt die gleiche Schriftart.
- Bilder und Bildobjekte
  Wenn beim Tausch eines Geräts, das neue Gerät eine andere Auflösung unterstützt als das vorherige Gerät, haben Sie mehrere Möglichkeiten um die Bilder anzupassen.
  Im Menü "Extras > Einstellungen > Visualisierung > Bilder Größenanpassung" passen Sie die Größe der Bilder an das neue Gerät an.

Siehe auch

Grundlagen (Seite 2126)

Möglichkeiten der Bildanpassung (Seite 2131)

14.6.3.2 Tastenbelegung beim Tausch eines Geräts

Einleitung

Da die verfügbaren Geräte mit unterschiedlichen Funktionstasten ausgestattet sind, werden die darauf projektierten Funktionen bei dem Tausch des Gerätes auf die verfügbaren Funktionstasten des neuen Geräts abgebildet.

Abbildung der Funktionstasten

Die Funktionstasten unterhalb des Displays werden von links beginnend nach rechts auf das neue Gerät abgebildet. Hat das neue Gerät weniger Tasten, so werden die fehlenden Tasten nicht abgebildet.

Hinweis

Bediengeräte ohne LED-Tasten

Wenn Sie im Engineering System ein Gerät mit projektierten LED-Tasten gegen ein Gerät ohne LED-Tasten tauschen, startet nach dem Transfer der Projektdaten auf das Gerät die Runtime nicht.

Löschen Sie die LED-Tasten-Projektierung, bevor Sie den Gerätetausch durchführen.

Beispiel: Tausch KTP1000 Basic gegen KTP600 Basic

Sie haben am KTP1000 Basic auf die Taste F2 eine Funktion projektiert. Die Funktion wird nach dem Tausch auf ein KTP600 Basic mit der Taste F2 ausgelöst.

Haben Sie bei einem KTP1000 Basic die Taste F7 verwendet, steht diese Funktion nach dem Tausch auf ein KTP600 Basic nicht mehr zur Verfügung.

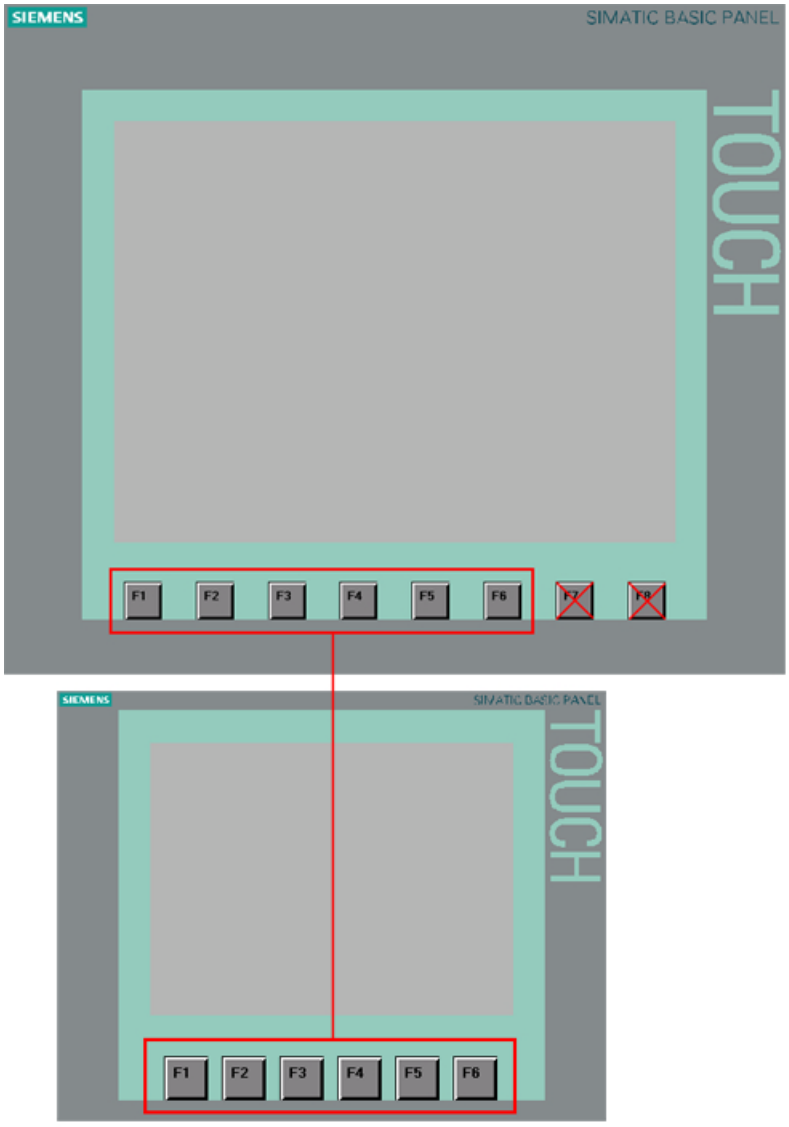

Abbildung der Steuertasten und Cursortasten

Die folgenden Tasten werden nur auf die gleichen Tasten des neuen Geräts abgebildet:

- HELP
- ESC
- ACK
- ENTER
- PAGE UP
- PAGE DOWN
- CURSOR UP
- CURSOR DOWN

Siehe auch

Geräteabhängige Funktionen (Seite 2128)

14.6.4 Bilder an das neue Gerät anpassen

14.6.4.1 Möglichkeiten der Bildanpassung

Einleitung

Vor dem Tausch eines Geräts wählen Sie die Größenanpassung der Bilder. Die Größenanpassung ist vor allem bei einem Tausch von Geräten mit verschiedenen Auflösungen des Displays von Bedeutung.

Bei Objekten wie Grafikanzeigen oder Textfeldern kann die Objektanpassung an den Inhalt unterbunden werden.

Hinweis

Wenn Sie ein Gerät mit einem Display im Querformat gegen ein Gerät mit Hochformat tauschen, werden die Objekte verzerrt dargestellt. Durch das veränderte Displayformat werden z. B. die Beschriftungen der Objekte abgeschnitten oder der Inhalt nicht an das Objekt angepasst. Passen Sie daher nach dem Tausch des Geräts die Bilder auf dem neuen Gerät an.

Bildanpassung beim Tausch eines Geräts

Im Menü "Extras > Einstellungen > Visualisierung >Anpassungen Bilder und Bildobjekte" passen Sie die Größe der Bilder an das neue Gerät an.

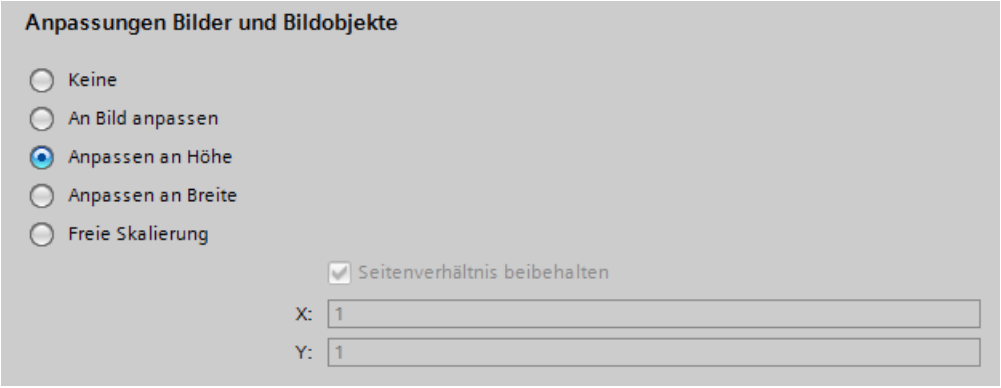

Wählen Sie eine der folgenden Einstellungen.

Keine (Standardeinstellung)

Die Bilder werden nicht skaliert. Die Objekte behalten ihre Position und Größe im Bild. Verwenden Sie diese Einstellung als ersten Test zum Prüfen eines möglichen Tauschergebnisses, da beim Hin- und Rücktausch keine Rundungsverluste auftreten.

Wenn das Display des neuen Geräts kleiner ist, kann diese Option dazu führen, dass sich Objekte außerhalb des projektierbaren Bereichs befinden.

An Bild anpassen

Position und Objektgröße werden an die neue Displaygröße angepasst. Die Anpassung findet auf der x- und y-Achse statt. Grafiken und Schriftgrößen werden entsprechend angepasst.

Anpassen an Höhe

Das Seitenverhältnis wird beibehalten und die Bilder an die Höhe des neuen Geräts angepasst.

Verwenden Sie diese Option, wenn Sie z. B. ein Gerät mit Displayformat 4:3 gegen ein Gerät mit Widescreen tauschen.

Anpassen an Breite

Das Seitenverhältnis wird beibehalten und die Bilder an die Breite des neuen Geräts angepasst.

Verwenden Sie diese Option, wenn Sie z. B. ein Gerät mit Widescreen gegen ein Gerät mit Displayformat 4:3 tauschen.

Freie Skalierung

Sie bestimmen einen freien Skalierungsfaktor für die Bildanpassung. Sie können sowohl einen Faktor für die x-Achse als auch für die y-Achse bestimmen.

Wenn Sie bei der freien Skalierung einen Faktor < 1 verwenden, kann es zu Verzerrungen der Objekte führen. Z. B. wird die Beschriftung der Objekte abgeschnitten oder der Inhalt nicht an das Objekt angepasst.

Passen Sie daher nach dem Tausch des Geräts die Bilder auf dem neuen Gerät an.

Hinweis

Bei Objekten mit einem festen Seitenverhältnis z. B. Zeigerinstrument, Kreis, findet keine Anpassung des Seitenverhältnisses statt. Die Objekte werden auf dem neuen Gerät mit gleichem Seitenverhältnis dargestellt, wie vor dem Tausch.

Siehe auch

Position der Bildobjekte festlegen (Seite 2135)

14.6.4.2 Objekte an Inhalt anpassen

Einleitung

Für einige Objekte können Sie im Inspektorfenster festlegen, dass sich ihre Größe an den jeweiligen Inhalt anpasst, z. B.:

- Textfeld: Größe an den enthaltenen Text anpassen.
- E/A-Feld: Größe an den enthaltenen Text anpassen.
- Symbolisches E/A-Feld: Größe an den enthaltenen Text oder die Textliste anpassen.
- Grafikanzeige: Größe an die enthaltene Grafik anpassen.

Größenanpassung für Textobjekte und Grafikobjekte

Im Menü "Extras > Einstellungen > Visualisierung > Anpassungen Bilder und Bildobjekte > Objekte an Inhalt anpassen" unterbinden Sie die automatische Größenanpassung der einzelnen Objekte. Damit werden die Objekte so skaliert, wie Sie es unter "Extras > Einstellungen > Visualisierung > Anpassungen Bilder und Bildobjekte" festgelegt haben.

Wählen Sie, für welche Objekte die automatische Größenanpassung unterbunden wird.

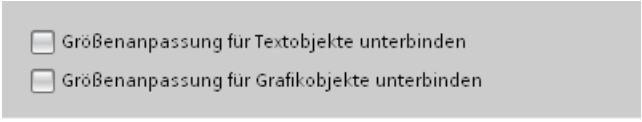

- Wenn "Größenanpassung für Textobjekte unterbinden" aktiviert ist, wird in den Eigenschaften der Textobjekte die automatische Größenanpassung ignoriert.
  Wenn Sie z. B. "Höhe an neues Gerät anpassen" aktiviert haben, wird das Textfeld ebenso wie die anderen Objekte, entsprechend der Höhe des neuen Geräts skaliert.
- Wenn "Größenanpassung für Grafikobjekte unterbinden" aktiviert ist, wird in den Eigenschaften der Grafikobjekte die automatische Größenanpassung ignoriert.
  Wenn Sie z. B. "Breite an neues Gerät anpassen" aktiviert haben, wird die Grafikanzeige ebenso wie die anderen Objekte, entsprechend der Breite des neuen Geräts skaliert.

  Hinweis

  Die Einstellungen haben keine Auswirkung auf Bildobjekte, deren Größe nicht geändert werden kann z. B. Meldeindikator oder Bildobjekte mit festem Seitenverhältnis.

"Größenanpassung für Textobjekte unterbinden" und "Größenanpassung für Textobjekte unterbinden" haben keine Auswirkung, wenn:

- Sie "Anpassung Bilder und Bildobjekte > Keine" aktiviert haben.

- Sie "Breite und Höhe an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.

- Sie "Höhe an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.

- Sie "Breite an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.

- Sie "Anpassung Bilder und Bildobjekte > Keine" und Position "Oben Links" gewählt haben, weil somit keine Anpassung notwendig ist.

Beispiel

Die folgende Abbildung zeigt die Auswirkungen der automatischen Größenanpassung am Beispiel eines Grafikobjekts, an dem zwei Schaltflächen ausgerichtet sind:

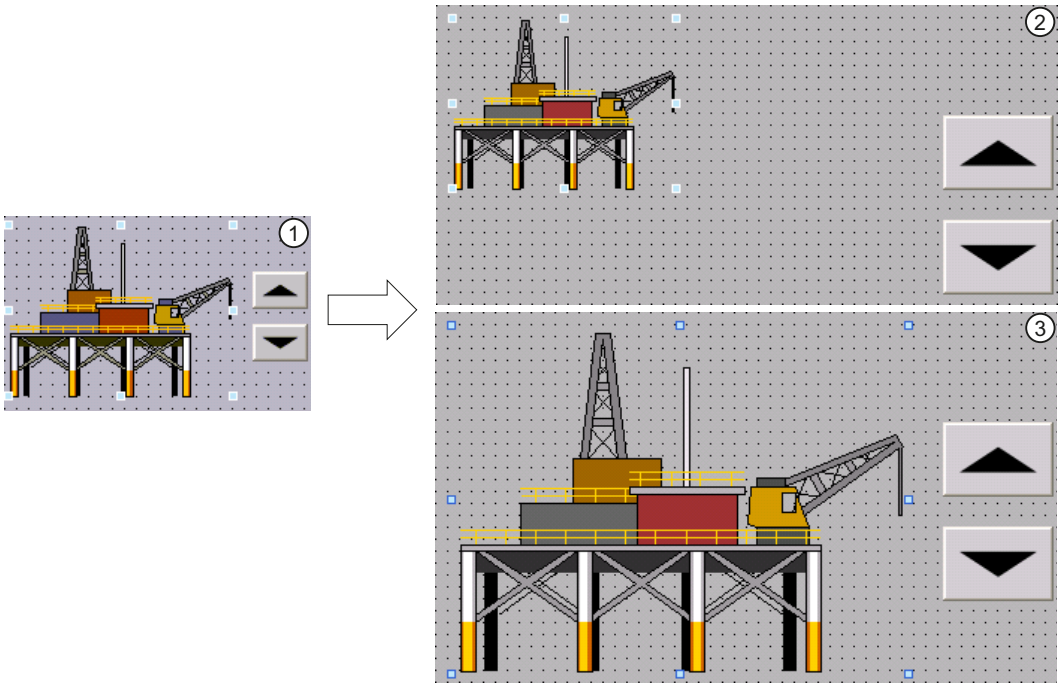

① Ausgangslage:

- Zwei Schaltflächen sind an einem Grafikobjekt ausgerichtet.
- In den Objekteigenschaften des Grafikobjekts ist unter "Darstellung > Größenanpassung" die Option "Objektgröße an Grafik anpassen" oder "Objektgröße an Grafik angleichen" aktiviert.

② Möglichkeit 1: Nach der Bediengerätumschaltung sollen die ursprünglichen Eigenschaften des Grafikobjekts beibehalten werden.

- Deaktivieren Sie in den Einstellungen unter "Größenanpassung der Objekte" die Option "Größenanpassung für Grafikobjekte unterbinden".

Auswirkung: Das Grafikobjekt wird nach der Bediengerätumschaltung in seiner ursprünglichen Größe beibehalten. Die Ausrichtung zu den Schaltflächen geht verloren.

③ Möglichkeit 2: Nach der Bediengerätumschaltung soll das Grafikobjekt relativ zur neuen Bildschirmauflösung platziert werden.

- Aktivieren Sie in den Einstellungen unter "Größenanpassung der Objekte" die Option "Größenanpassung für Grafikobjekte unterbinden".

In den Objekteigenschaften des Grafikobjekts wird die Option "Grafik an Objektgröße anpassen" automatisch aktiviert. Die beiden Schaltflächen sind auch nach der Bediengerätumschaltung korrekt am Grafikobjekt ausgerichtet.

Siehe auch

Position der Bildobjekte festlegen (Seite 2135)

14.6.4.3 Position der Bildobjekte festlegen

Einleitung

Um die Position der Bildobjekte an das neue Gerät anzupassen, haben Sie verschiedene Möglichkeiten.

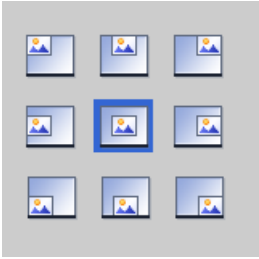

Position wählen

Im Menü "Extras > Einstellungen > Visualisierung > Bilder Größenanpassung > Position wählen" passen Sie die Position der Bildobjekte an das neue Gerät an.

Beispiel

Die folgende Option richtet die Objekte an der oberen linken Kante aus.

Die folgende Option zentriert die Objekte in der Mitte des Bildschirms.

Siehe auch

Möglichkeiten der Bildanpassung (Seite 2131)

Objekte an Inhalt anpassen (Seite 2133)

14.6.5 Beispiel: Gerät tauschen

14.6.5.1 Beispiel: Handlungsübersicht

Einleitung

Im folgenden Beispiel tauschen Sie ein KTP700 Basic 7" PN gegen ein TP1500 Comfort PRO.

Handlungsübersicht

Der Tausch eines Geräts hat im Beispiel folgenden Ablauf.

1. Bilder an das neue Gerät anpassen
2. Gerät tauschen

Siehe auch

Beispiel: Bilder anpassen (Seite 2136)

Beispiel: Gerät tauschen (Seite 2137)

14.6.5.2 Beispiel: Bilder anpassen

Einleitung

Bevor Sie ein Gerät tauschen, führen Sie noch Bildanpassungen durch. Die unterschiedlichen Formate der Displays erfordern eine Bildanpassung.

Das KTP700 Basic hat das Format 800 x 480 Pixel, während das TP1500 Comfort PRO das Format 1366 x 768 hat.

Voraussetzungen

- Ein Projekt ist geöffnet.
- Im Projekt wird das Gerät KTP700 Basic 7" PN verwendet.

Bilder anpassen

1. Öffnen Sie das Menü "Extras > Einstellungen".
2. Klicken Sie auf "Visualisierung > Anpassungen Bilder und Bildobjekte".

3. Aktivieren Sie "Anpassen an Höhe".

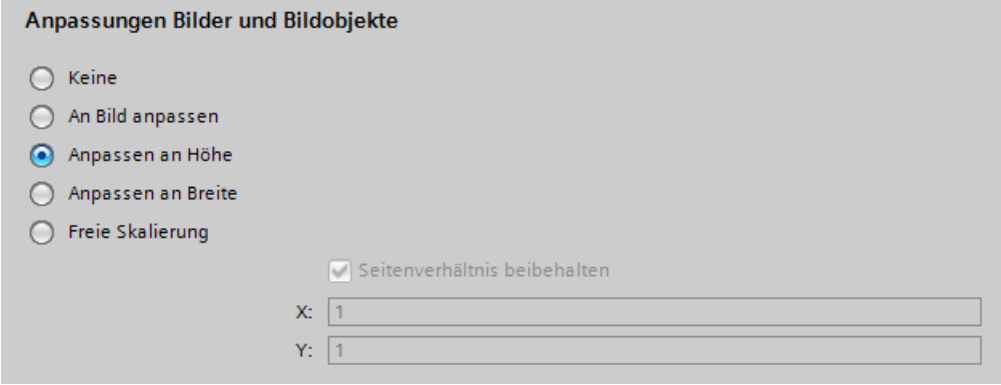

4. Aktivieren Sie "Größenanpassung für Textobjekte unterbinden".
5. Aktivieren Sie "Größenanpassung für Grafikobjekte unterbinden".

Ergebnis

Sie haben die Bildanpassung als Vorbereitung für den Tausch des Geräts vorgenommen.

Siehe auch

Beispiel: Handlungsübersicht (Seite 2136)

Beispiel: Gerät tauschen (Seite 2137)

14.6.5.3 Beispiel: Gerät tauschen

Einleitung

Das folgende Beispiel zeigt, wie Sie ein Gerät tauschen.

Voraussetzungen

- Ein Projekt ist angelegt und geöffnet.
- Im Projekt wird das Gerät KTP700 Basic 7" PN verwendet.

Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Geräte & Netze". Der Editor wird geöffnet.
2. Klicken Sie auf das Gerät "KTP700 Basic PN".
3. Wählen Sie im Kontextmenü des Geräts den Eintrag "Gerät/Version ändern". Ein Dialog wird geöffnet.
4. Selektieren Sie das Gerät TP1500 Comfort PRO.

5. Wählen Sie unter "Version" das zur Projektierung passende Image für das Bediengerät. Wenn Sie das Image ändern, wird beim nächsten Laden das neue Image auf das Zielsystem übertragen. Dabei werden alle Runtime-Daten gelöscht.
   In der "Kompatibilitätsinformation" erhalten Sie Informationen über Unterschiede in der Hardware.

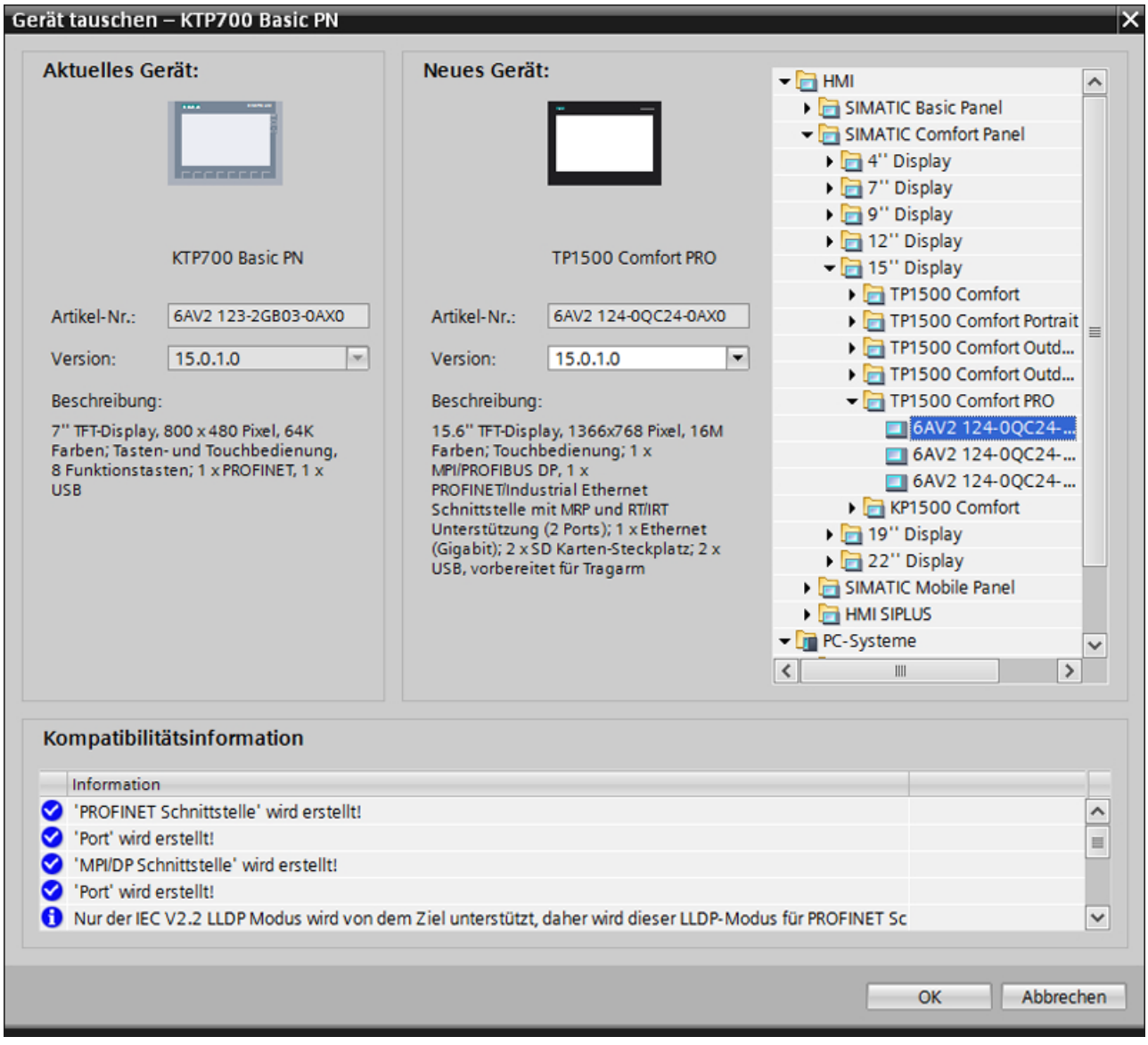

6. Klicken Sie auf "OK". Der Tausch des Geräts wird gestartet.

Ergebnis

Sie haben das im Projekt verwendete Gerät KTP700 Basic getauscht. Sie verwenden nun das Gerät TP1500 Comfort PRO.

Siehe auch

Beispiel: Handlungsübersicht (Seite 2136)
Aktualisierung des Betriebssystems (Seite 2432)
Geräteversion ändern (Seite 2188)
Wechsel zwischen Geräteversionen (Seite 2187)
Grundlagen (Seite 2126)
Beispiel: Bilder anpassen (Seite 2136)

14.7 Zwischen Bediengeräten und Editoren kopieren

14.7.1 Grundlagen

14.7.1.1 Grundlagen

Kopieren und Einfügen innerhalb eines Bediengeräts

Innerhalb eines Bediengeräts kopieren und fügen Sie Objekte ein, z. B. ein Anzeigeobjekt. Wenn das Objekt bereits in dem Editor angelegt ist, dann wird beim Einfügen dem Objektnamen automatisch eine Nummerierung nach folgendem Prinzip angehängt:

- "<Objektname>_1" wird umbenannt zu "<Objektname>_2".
- "<Objektname>_2" wird umbenannt zu "<Objektname>_3".

Kopieren und Einfügen zwischen Bediengeräten

Sie können auch zwischen Bediengeräten kopieren und einfügen. Wenn bereits ein Objekt mit demselben Namen existiert, haben Sie folgende Möglichkeiten:

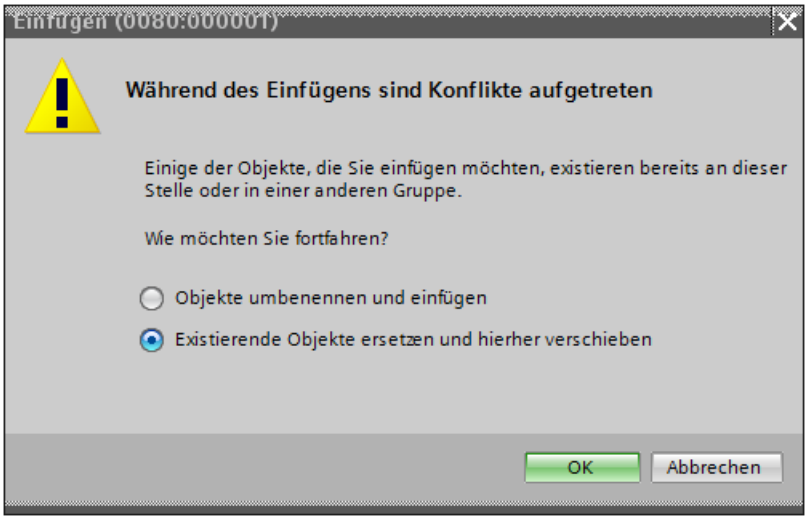

Hinweis

Ausnahme von dieser Grundregel

Das Kopieren und Einfügen einer Meldeklasse, die aus einer projektweiten Meldeklasse erzeugt wurde, wird abweichend von dieser Grundregel behandelt. Wenn die kopierte Meldeklasse im Ziel-Bediengerät innerhalb des gleichen Projekts bereits vorhanden ist, wird der Befehl "Einfügen" nicht ausgeführt.

Benutzerdefinierte Ordner kopieren

In der Projektnavigation legen Sie benutzerdefinierte Ordner für Editoren z. B. für HMI-Variablen, Bilder usw. an.

Sie kopieren benutzerdefinierte Ordner und fügen sie in ein anderes Bediengerät ein. Die in einem benutzerdefinierten Ordner enthaltenen Objekte können die für ein anderes Bediengerät bestehenden Einschränkungen überschreiten, z. B. bei der Anzahl der unterstützten Bilder. Nach dem Einfügen werden alle Objekte angezeigt. Beim Übersetzen der Projektdaten wird ein Fehler ausgegeben.

Systemordner sind grundsätzlich nicht kopierbar.

Siehe auch

Nicht unterstützte Objekte und Funktionalitäten (Seite 2142)

Kopieren von Objekten mit verknüpften Objekten (Seite 2144)

Automatisch kopierte verknüpfte Objekte (Seite 2144)

Kopieren von Bildern (Seite 2147)

Grundlagen (Seite 2150)

14.7.1.2 Möglichkeiten für das Kopieren und Einfügen

Möglichkeiten für das Kopieren und Einfügen

Durch das Kopieren und Einfügen von einzelnen Objekten vermindern Sie den Projektierungsaufwand.

In WinCC haben Sie verschiedene Möglichkeiten Objekte zu kopieren und einzufügen.

Kontextmenü

Um Objekte über das Kontextmenü zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein Objekt, z. B. eine Schaltfläche.
2. Wählen Sie im Kontextmenü "Kopieren".
3. Positionieren Sie den Mauszeiger an die Stelle im Bild, an der Sie eine Kopie der Schaltfläche einfügen wollen.
4. Wählen Sie im Kontextmenü "Einfügen".
   Die Schaltfläche wird eingefügt mit allen bereits festgelegten Eigenschaften.

Drag&Drop

Um Objekte per Drag&Drop zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation eines Geräts auf "Bilder > Start".
2. Ziehen Sie das Bild "Start" per Drag&Drop auf den Ordner Bilder eines anderen Geräts.

3. Wenn ein gleichnamiges Bild auf dem zweiten Gerät existiert, wird ein Dialog eingeblendet.
4. Wählen Sie, ob das vorhandene Bild ersetzt wird, bzw. ob das Bild "Screen" umbenannt wird.

14.7.1.3 Nicht unterstützte Objekte und Funktionalitäten (Professional)

Einleitung

Beim Kopieren eines Objekts werden alle seine Eigenschaften und Einstellungen vom Ziel-Bediengerät übernommen.

Nicht unterstützte Objekte

Objekte, die im Ziel-Bediengerät nicht unterstützt werden, können nicht eingefügt werden.

Hinweis

Wenn Sie ein Bild kopieren, das Objekte enthält, die vom Ziel-Bediengerät nicht unterstützt werden, bleiben die Objekte im Hintergrund erhalten. Wenn Sie das Bild erneut kopieren und das neue Gerät die Objekte unterstützt, werden Sie wieder angezeigt.

Ungültige Objekte

Folgende Objekte werden nach dem Einfügen im Ziel-Bediengerät ungültig:

- Referenzierte Objekte, die im Ziel-Bediengerät nicht vorhanden sind.
- Objekte mit Einstellungen, die im Ziel-Bediengerät nicht unterstützt werden.
- An Objekte projektierte Systemfunktionen, die im Ziel-Bediengerät nicht unterstützt werden.

Ungültige Objekte werden durch eine farbige Markierung gekennzeichnet. Wählen Sie ein unterstütztes Objekt oder legen Sie ein neues Objekt an. Wenn Sie ein ungültiges Objekt beibehalten, wird beim Übersetzen der Projektdaten ein Fehler ausgegeben.

Farben und Schriftarten

Farben und Schriftarten werden von Bediengeräten in unterschiedlichem Umfang unterstützt. Beim Einfügen werden nicht unterstützte Farben und Schriftarten durch unterstützte Farben und Schriftarten ersetzt. Wenn Sie dasselbe Objekt zurück in das Quell-Bediengerät einfügen, sind die ursprünglich vorhandenen Einstellungen wieder aktiv.

14.7.1.4 Nicht unterstützte Objekte und Funktionalitäten

Einleitung

Beim Kopieren eines Objekts werden alle seine Eigenschaften und Einstellungen vom Ziel-Bediengerät übernommen.

Nicht unterstützte Objekte

Objekte, die im Ziel-Bediengerät nicht unterstützt werden, können nicht eingefügt werden.

Hinweis

Wenn Sie ein Bild kopieren, das Objekte enthält, die vom Ziel-Bediengerät nicht unterstützt werden, bleiben die Objekte im Hintergrund erhalten. Wenn Sie das Bild erneut kopieren und das neue Gerät die Objekte unterstützt, werden Sie wieder angezeigt.

Ungültige Objekte

Folgende Objekte werden nach dem Einfügen im Ziel-Bediengerät ungültig:

- Referenzierte Objekte, die im Ziel-Bediengerät nicht vorhanden sind.
- Objekte mit Einstellungen, die im Ziel-Bediengerät nicht unterstützt werden.
- An Objekte projektierte Systemfunktionen, die im Ziel-Bediengerät nicht unterstützt werden.

Ungültige Objekte werden durch eine farbige Markierung gekennzeichnet. Wählen Sie ein unterstütztes Objekt oder legen Sie ein neues Objekt an. Wenn Sie ein ungültiges Objekt beibehalten, wird beim Übersetzen der Projektdaten ein Fehler ausgegeben.

Farben und Schriftarten

Farben und Schriftarten werden von Bediengeräten in unterschiedlichem Umfang unterstützt. Beim Einfügen werden nicht unterstützte Farben und Schriftarten durch unterstützte Farben und Schriftarten ersetzt. Wenn Sie dasselbe Objekt zurück in das Quell-Bediengerät einfügen, sind die ursprünglich vorhandenen Einstellungen wieder aktiv.

Meldeklassen

Nur ein Teil der aktivierten Meldeklassen wird aktiviert.

Wenn Sie in Projekt_1 z. B. ein Meldefenster im Globalen Bild projektieren, kopieren Sie das Meldefenster und fügen es in das Globale Bild in Projekt_2 ein.

Nur ein Teil der aktivierten Meldeklassen wird nach dem Einfügen aktiviert.

Das Verhalten tritt bei folgenden Anzeigeobjekten auf:

- Meldefenster
- Meldeindikator
- Meldeanzeige

Siehe auch

Grundlagen (Seite 2140)

14.7.2 Kopieren und Einfügen

14.7.2.1 Kopieren von Objekten mit verknüpften Objekten

Einleitung

Ein Objekt wird z. B. in folgenden Fällen mit einem anderen Objekt verknüpft:

- Sie legen für eine Meldung eine Variable als Triggervariable fest.
  Die Meldung ist das Objekt. Die Variable ist das verknüpfte Objekt.
- Sie legen für eine externe Variable eine Verbindung fest.
  Die Variable ist das Objekt. Die Verbindung ist das verknüpfte Objekt.

Das Objekt wird beim Kopieren und Einfügen immer vollständig eingefügt. Ob das verknüpfte Objekt eingefügt wird, hängt vom benutzten Befehl für das Einfügen ab.

Einfaches Einfügen

Das verknüpfte Objekt wird nicht kopiert. Die Verknüpfung zum Objekt wird übernommen und im Ziel-Bediengerät wie folgt umgesetzt:

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt mit seinen Einstellungen genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, wird der Objektname angezeigt. Das Objekt wird ungültig.

Bei einigen Objekten werden verknüpfte Objekte beim einfachen Einfügen automatisch mit kopiert und eingefügt.

Siehe auch

Automatisch kopierte verknüpfte Objekte (Seite 2144)

Grundlagen (Seite 2140)

14.7.2.2 Automatisch kopierte verknüpfte Objekte

Kopieren verknüpfter Objekte

Die folgende Tabelle zeigt die Objekte, bei denen verknüpfte Objekte beim einfachen Einfügen automatisch mit eingefügt werden.

Objekt	Verknüpftes Objekt
Bild	Vorlage
Symbolisches E/A-Feld	Textliste
Grafisches E/A-Feld	Grafikliste
Grafikanzeige	Grafik

Objekt	Verknüpftes Objekt
Variable	Meldung
	Zyklus
Rezepturelement	Textliste
Aufgabenplaner	Trigger

Siehe auch

Kopieren von Objekten mit verknüpften Objekten (Seite 2144)

Grundlagen (Seite 2140)

Grundlagen (Seite 2150)

14.7.2.3 Automatisch kopierte verknüpfte Objekte

Kopieren verknüpfter Objekte

Die folgende Tabelle zeigt die Objekte, bei denen verknüpfte Objekte beim einfachen Einfügen automatisch mit eingefügt werden.

Objekt	Verknüpftes Objekt
Bild	Zyklus
	Vorlage
Symbolisches E/A-Feld	Textliste
Grafisches E/A-Feld	Grafikliste
Grafikanzeige	Grafik
Variable	Meldung
	Archivvariable
	Zyklus
Archiv	Archivvariable
Archivvariable	Archivtyp
Rezepturelement	Textliste
Aufgabenplaner	Trigger

14.7.2.4 Kopieren von Bediengeräten mit HMI-Verbindungen

Wenn Sie ein Bediengerät mit HMI-Verbindungen zu einer Steuerung kopieren, wird die HMI-Verbindung im neuen Bediengerät nicht automatisch mit einer vorhandenen namensgleichen Steuerung verbunden. Dies gilt sowohl für das Kopieren innerhalb eines Projekts als auch für das projektübergreifende Kopieren.

Projektierung der HMI-Verbindung vervollständigen

Um auch im neuen Bediengerät über eine HMI-Variable auf die Steuerungsvariable zugreifen zu können, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "Geräte & Netze".
2. Verbinden Sie das neue Bediengerät mit dem gewünschten Netz.
3. Öffnen Sie die Verbindungstabelle.
4. Wählen Sie die HMI-Verbindung des neuen Bediengerätes aus.
5. Wählen Sie unter "Partner" die gewünschte Steuerung aus.

Hinweis

Wenn Sie zwischen Kopieren des HMI-Geräts und Vervollständigen der Verbindung das neue Bediengerät übersetzen oder weitere PLC-Variablen verbinden, wird in manchen Fällen eine zusätzliche HMI-Verbindung zur gleichen Steuerung erzeugt. Dies ist insbesondere der Fall, wenn HMI-Variablen mit DB-Array-Elementen verbunden sind.

14.7.2.5 Mit Anwenderdatentypen verknüpfte Bildobjekte kopieren

Einleitung

Wenn Sie ein Bildobjekt von einem Bild, das einen Anwenderdatentyp verwendet, zu einem Bild kopieren, das keinen Anwenderdatentyp verwendet, werden die Variablenreferenzen im Zielbild mit dem Präfix "@NOP::" angezeigt.

Verschaltung des Anwenderdatentyps

Damit die Variablenreferenzen in Runtime funktionieren, verschalten Sie am Zielbild temporär einen Anwenderdatentyp. Anschließend kann die Verschaltung des Anwenderdatentyps am Zielbild wieder gelöscht werden.

14.7.2.6 Drag&Drop aus der Detailansicht

Einleitung

Mit einfachen Handgriffen erhöhen Sie die Effizienz Ihrer Projektierung. Im Folgenden sehen Sie einige Beispiele für eine effiziente Projektierung.

Objekte aus der Detailansicht in ein Bild einfügen

Sie ziehen aus der Detailansicht Objekte verschiedener Editoren in andere Editoren.

Symbolische E/A-Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "Text - und Grafiklisten". In der Detailansicht werden alle verfügbaren Text- und Grafiklisten angezeigt.
3. Klicken Sie in der Detailansicht auf eine Textliste z. B. "Textlist1".
4. Ziehen Sie per Drag&Drop eine Textliste aus der Detailansicht in ein Bild. Éin Symbolisches E/A-Feld wurde angelegt und mit der Textliste "Textlist1" verschaltet.

Grafisches E/A - Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "Text - und Grafiklisten". In der Detailansicht werden alle verfügbaren Text- und Grafiklisten angezeigt.
3. Klicken Sie in der Detailansicht auf eine Grafikliste z. B. "Graficlist1".
4. Ziehen Sie per Drag&Drop eine Grafikliste aus der Detailansicht in ein Bild. Éin Grafisches E/A-Feld wurde angelegt und mit der Grafikliste "Graficlist1" verschaltet.

E/A - Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "HMI-Variablen". In der Detailansicht werden alle verfügbaren HMI-Variablen angezeigt.
3. Klicken Sie in der Detailansicht auf HMI-Variable z. B. "Tag1".
4. Ziehen Sie per Drag&Drop die HMI-Variable aus der Detailansicht in ein Bild. Éin E/A-Feld wurde angelegt und mit der HMI-Variable "Tag1" verschaltet.

14.7.2.7 Kopieren von Bildern

Einleitung

Sie kopieren ein oder mehrere Bilder aus dem Ordner "Bilder" und fügen sie in den Ordner "Bilder" eines anderen Bediengeräts ein.

Art und Größe des Displays

Bei Bediengeräten mit Tasten werden die zur Verfügung stehenden Tasten automatisch im Bild angezeigt. Beim Kopieren eines Bildes zwischen den Bediengeräten werden die Tasten ein- bzw. ausgeblendet. An Funktionstasten projektierte Funktionen werden nicht übernommen.

Wenn im Ziel-Bediengerät weniger Platz für das Bild zur Verfügung steht als im Quell-Bediengerät, können Sie die Größe der vorhandenen Objekte und die Abstände zwischen den vorhandenen Objekten anpassen.

Automatische Größenanpassung der Objekte

1. Wählen Sie im Menü "Extra > Einstellungen > Visualisierung > Anpassung Bilder und Bildobjekte".
2. Aktivieren Sie z. B. "Anpassen an Höhe".

Siehe auch

Grundlagen (Seite 2140)

14.7.2.8 Kopieren von Rezepturen innerhalb eines Bediengeräts

Editor "Rezepturen"

Sie kopieren Rezepturen, Rezepturelemente und Rezepturdatensätze innerhalb der jeweiligen Tabelle. Sie kopieren ein Rezepturelement in eine andere Rezeptur.

Nur WinCC Runtime Professional: Sie können ein Rezeptursicht-Element in eine andere Rezeptursicht kopieren. Wenn ein Rezeptursicht-Element mit dem gleichen Namen bereits existiert, wird ein Konflikt-Dialog angezeigt. Sie können wählen, das Rezepturelement zu ersetzen oder umzubenennen. Sie können Rezepturelemente kopieren in die erste leere Zeile des Editors "Rezeptursichten", Register "Elemente".

Sie können einen Rezepturdatensatz in eine andere Rezeptur kopieren, falls die andere Rezeptur die gleiche Anzahl Rezepturelemente enthält. Wenn die Datentypen unterschiedlich sind, wird der Wert zwar in den Ziel-Datensatz kopiert, erhält jedoch eine Fehlermarkierung.

Editor "Variablen"

Sie können eine Variable per Drag&Drop auf ein Rezepturelement in die Spalte "Variable" ziehen. Die Variable wird mit dem Rezepturelement verbunden. Wenn schon eine Variable verbunden ist, wird eine Fehlermeldung ausgegeben.

Editor "Bilder"

Wenn Sie eine Rezeptur per Drag&Drop auf ein Bild ziehen, wird eine neue Rezepturanzeige erstellt und mit der Rezeptur verbunden.

14.7.2.9 Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Überblick

Die folgende Tabelle zeigt die Verfügbarkeit der unterschiedlichen Funktionen in Runtime Advanced und Panels und Runtime Professional.

Funktion in WinCC	Runtime Advanced und Panels	Runtime Professional
Archive	Ja Variablenarchiv Meldearchiv	Ja Variablenarchiv Verdichtungsarchiv Meldearchiv
Aufgabenplaner	Ja	Ja
Benutzerverwaltung	Ja	Ja
Bilder	Ja	Ja
Einstellungen	Ja	Ja
Globale Elemente	Ja	Nein
Meldungen	Ja Meldeindikator Meldefenster Meldeanzeige	Ja Meldeanzeige mit mehr Funktionen als in Runtime Advanced und Panels
Menüs und Symbolleisten	Nein	Ja
Protokolle	Ja Ausdruck über Systemfunktion "Drucke-Protokoll"	Ja Ausdruck über Druckauftrag und Aufgabenplaner
Rezepturen	Ja	User Archives aus WinCC V6.2
Skripte	Ja VBS	Ja VBS C
Strukturen	Ja	Ja
Text- und Grafiklisten	Ja	Ja
Variablen	Ja	Ja
Verbindungen	Ja Unterschiedliche Kommunikationstreiber	Ja Unterschiedliche Kommunikationstreiber
Vorlagen	Ja	Nein
Zyklen	Ja	Ja

14.7.3 Kopieren und Einfügen zwischen Runtime Advanced und Panels und Runtime Professional

14.7.3.1 Grundlagen

Einleitung

Sie können Projektdaten, z. B. Bilder, Objekte oder Variablen, zwischen Bediengeräten kopieren und einfügen, die zu unterschiedlichen Runtimes gehören.

Eigenschaften und Einstellungen

Objekte, die in beiden Runtimes unterstützt werden, können unterschiedliche Eigenschaften und Einstellungen haben. Beim Kopieren und Einfügen werden diese Unterschiede wie folgt umgesetzt:

- Eigenschaften, die vom Ziel-Bediengerät nicht unterstützt werden, werden nicht angezeigt.
- Für Eigenschaften, die ausschließlich vom Ziel-Bediengerät unterstützt werden, werden Standardeinstellungen festgelegt.
- Einstellungen, die vom Ziel-Bediengerät nicht unterstützt werden, werden durch Standardeinstellungen ersetzt.

Wenn Sie ein Objekt zurück in das Quell-Bediengerät kopieren, ohne dass Sie manuell Einstellungen im Ziel-Bediengerät verändert haben, sind alle ursprünglich im Quell-Bediengerät festgelegten Eigenschaften und Einstellungen wieder aktiv.

Nicht unterstützte Ereignisse und Animationen

Wenn Sie ein Objekt in einer anderen Runtime einfügen, sind folgende Fälle möglich:

- Ereignisse oder Animationen werden vom Ziel-Bediengerät für das kopierte Objekt grundsätzlich nicht unterstützt.
  Ereignisse oder Animationen, die an das kopierte Objekt projektiert wurden, werden im Ziel-Bediengerät nicht angezeigt.
- Ereignisse oder Animationen werden für das kopierte Objekt grundsätzlich nur vom Ziel-Bediengerät unterstützt.
- Bestimmte Ereignisse oder Animationen werden vom Ziel-Bediengerät für das kopierte Objekt nicht unterstützt.
  Ereignisse oder Animationen, die an das kopierte Objekt projektiert wurden und vom Ziel-Bediengerät nicht unterstützt werden, werden im Ziel-Bediengerät nicht angezeigt.
- Bestimmte Ereignisse oder Animationen werden nur vom Ziel-Bediengerät für das kopierte Objekt unterstützt.

Wenn Sie ein Objekt zurück in das Quell-Bediengerät kopieren, sind alle ursprünglich vorhandenen Ereignisse und Animationen wieder aktiv.

Siehe auch

Automatisch kopierte verknüpfte Objekte (Seite 2144)

Einschränkungen beim Kopieren (Seite 2152)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2149)

Strukturelemente (Seite 2158)

Objekte (Seite 2172)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2174)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2175)

Kopieren von Variablen (Seite 2156)

Grundlagen (Seite 2140)

14.7.3.2 Systemfunktionen

Unterstützte Systemfunktionen

Folgende Systemfunktionen werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

- AktiviereBild
- ErhoeheVariable
- InvertiereBit
- InvertiereBitInVariable
- InvertiereLineareSkalierung
- LineareSkalierung
- RuecksetzeBit
- RuecksetzeBitInVariable
- SetzeBit
- SetzeBitInVariable
- SetzeVariable
- StoppeRuntime
- VerringereVariable
- ZeigeAnmeldedialog

In Runtime Advanced und Panels werden zusätzlich weitere Systemfunktionen unterstützt.

Siehe auch

Bildobjekte (Seite 2153)

14.7.3.3 Einschränkungen beim Kopieren

Unterschiede zwischen den Runtimes

Einige Funktionalitäten werden ausschließlich in einer Runtime unterstützt. Einige Funktionalitäten werden in beiden Runtimes unterstützt, sind aber unterschiedlich konzipiert. Diese Funktionalitäten können Sie nicht zwischen den Runtimes kopieren und einfügen.

Nur in Runtime Advanced und Panels unterstützte Funktionalitäten

- Vorlagen
- Globale Elemente

Diese Funktionalitäten können Sie nicht nach Runtime Professional kopieren.

Nur in Runtime Professional unterstützte Funktionalitäten

- Menüs und Symbolleisten
- C-Skripte

Diese Funktionalität können Sie nicht nach Runtime Advanced und Panels kopieren.

Unterschiedlich konzipierte Funktionen

- Konfiguration
- Rezepturen

Diese Funktionalitäten können Sie nicht zwischen den Runtimes kopieren.

Siehe auch

Grundlagen (Seite 2150)

14.7.3.4 Objekte in Bildern

Einleitung

Wenn Sie ein Objekt aus einem Bild kopieren und in ein anderes Bild einfügen, wird das Objekt an der gleichen Stelle im Bild eingefügt.

Text- und Grafiklisten

Wenn Sie ein symbolisches E/A-Feld mit einer Textliste oder ein grafisches E/A-Feld mit einer Grafikliste kopieren und in die andere Runtime einfügen, wird die Textliste oder Grafikliste in das Ziel-Bediengerät kopiert.

Objekte aus Protokollen kopieren

Sie können Objekte, die im Bild unterstützt werden, aus einem Protokoll in ein Bild kopieren. Dabei werden nicht unterstützte Einstellungen durch unterstützte Einstellungen ersetzt. Wenn Sie ein kopiertes Objekt wieder in das Protokoll kopieren, bleiben die geänderten Einstellungen aktiv.

Rezepturanzeige und Meldeanzeige

Die Objekte "Rezepturanzeige" und "Meldeanzeige" sind konzeptionell so unterschiedlich angelegt, dass Sie diese Objekte nicht zwischen den beiden Runtimes kopieren und einfügen können.

Siehe auch

Bildobjekte (Seite 2153)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2155)

Objekte (Seite 2172)

14.7.3.5 Bildobjekte

Unterstützte Objekte

Die folgende Tabelle zeigt, welche Objekte in Bildern in welcher Runtime unterstützt werden:

Tabelle 14-13 Basisobjekte

Objekt	Runtime Advanced und Panels	Runtime Professional
Linie	Ja	Ja
Polygonzug	Ja	Ja
Polygon	Ja	Ja
Ellipse	Ja	Ja
Ellipsensegment	Nein	Ja
Kreissegment	Nein	Ja
Ellipsenbogen	Nein	Ja
Kreisbogen	Nein	Ja
Kreis	Ja	Ja
Rechteck	Ja	Ja
Verbinder	Nein	Ja

Objekt	Runtime Advanced und Panels	Runtime Professional
Textfeld	Ja	Ja
Grafikanzeige	Ja	Ja
Rohr	Nein	Ja
Doppel-T-Stück	Nein	Ja
T-Stück	Nein	Ja
Rohrbogen	Nein	Ja

Tabelle 14-14 Elemente

Objekt	Runtime Advanced und Panels	Runtime Professional
E/A-Feld	Ja	Ja
Editierbares Textfeld	Nein	Ja
Listenfeld	Nein	Ja
Kombinationsfeld	Nein	Ja
Schaltfläche	Ja	Ja
Datum/Uhrzeit-Feld	Ja	Nein
Rundschaltfläche	Nein	Ja
Symbolisches E/A-Feld	Ja	Ja
Grafisches E/A-Feld	Ja	Ja
Balken	Ja	Ja
Schalter	Ja	Nein
Symbolbibliothek	Ja	Ja
Schieberegler	Ja	Ja
Bildlaufleiste	Nein	Ja
Kontrollkästchen	Nein	Ja
Optionsschaltflächen	Nein	Ja
Zeigerinstrument	Ja	Ja
Uhr	Ja	Ja
Speicherplatzanzeige	Nein	Ja
Ladezustand	Ja	Nein
Wirkbereichs-Name	Ja	Nein
Wirkbereichs-Name (RFID)	Ja	Nein
Wirkbereichs-Signal	Ja	Nein
Zonen-Name	Ja	Nein
Zonen-Signal	Ja	Nein

Tabelle 14-15 Controls

Objekt	Runtime Advanced und Panels	Runtime Professional
Bildfenster	Nein	Ja
Kurvenanzeige	Ja	Nein
Benutzeranzeige	Ja	Ja

Objekt	Runtime Advanced und Panels	Runtime Professional
HTML-Browser	Ja	Ja
Druckauftrag/Skriptdiagnose	Nein	Ja
Sm@rtClient-Anzeige	Ja	Nein
Beobachtungstabelle	Ja	Nein
Rezepturanzeige	Ja	Ja
Meldeanzeige	Ja	Ja
f(x)-Kurvenanzeige	Ja	Ja
f(t)-Kurvenanzeige	Nein	Ja
Tabellenanzeige	Nein	Ja
Wertetabelle	Nein	Ja
System-Diagnoseanzeige	Ja	Ja
Media Player	Nein	Ja
Kanaldiagnose-Anzeige	Nein	Ja
PLC-Code-Anzeige	Ja	Ja
GRAPH-Übersicht	Ja	Ja
ProDiag-Übersicht	Ja	Ja
Kriterienanalyse-Anzeige	Ja	Ja
Kamera-Anzeige	Ja	Nein
PDF-Anzeige	Ja	Nein

Siehe auch

Objekte in Protokollen (Seite 2173)

Ereignisse für Aufgabenplaner (Seite 2177)

Datentypen und Eigenschaften für Strukturen (Seite 2160)

Datentypen und Eigenschaften für Variablen (Seite 2157)

Systemfunktionen (Seite 2151)

14.7.3.6 Verbindungen

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Kommunikationstreiber

Die in den Runtimes möglichen Kommunikationstreiber entnehmen Sie der Referenz im Handbuch Kommunikation. Runtime Advanced und Panels und Runtime Professional unterstützen unterschiedliche Kommunikationstreiber. Es besteht kein gemeinsamer Kommunikationstreiber für beide Runtimes.

Bereichszeiger

In Runtime Professional werden die Bereichszeiger nicht unterstützt.

Siehe auch

Objekte in Bildern (Seite 2152)

Grundlagen (Seite 2150)

Kopieren von Verbindungen (Seite 2156)

Kopieren von Verbindungen

Einleitung

Sie können eine Verbindung im Editor "Verbindungen" kopieren und im Editor "Verbindungen" einer anderen Runtime einfügen.

Integrierte Verbindungen

Integrierte Verbindungen werden unter "Geräte & Netze" in der Netzsicht erstellt. Eine integrierte Verbindung wird im Editor "Verbindungen" angezeigt. Sie können die integrierte Verbindung im Editor "Verbindungen" aber nicht bearbeiten. Wenn Sie eine integrierte Verbindung aus dem Editor "Verbindungen" in eine andere Runtime kopieren und einfügen, wird diese Verbindung im Ziel-Bediengerät ungültig, da der Kommunikationspartner fehlt. Sie können die kopierte integrierte Verbindung ausschließlich in der Netzsicht des Ziel-Bediengeräts bearbeiten.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2155)

14.7.3.7 Kopieren von Variablen

Einleitung

Sie können eine Variable im Editor "Variablen" kopieren und im Editor "Variablen" einer anderen Runtime einfügen.

Nicht unterstützte Datentypen

Wenn Sie eine Variable mit einem Datentyp kopieren, der in der anderen Runtime nicht unterstützt wird, wird der Datentyp ungültig.

Archivvariablen

Wenn Sie eine Archivvariable mit der Variable verbunden haben, wird diese Archivvariable beim Kopieren der Variable mit in das Ziel-Bediengerät kopiert. Eine Archivvariable muss einem Variablenarchiv zugeordnet sein. Wenn im Ziel-Bediengerät kein entsprechendes Variablenarchiv vorhanden ist, wird ein neues Archiv angelegt.

Meldungen

Wenn Sie eine Analogmeldung oder eine Bitmeldung an eine Variable projektiert haben, wird diese Meldung beim Kopieren der Variable mit in das Ziel-Bediengerät kopiert.

Systemvariablen

Systemvariablen werden ausschließlich in Runtime Professional unterstützt.

Siehe auch

Grundlagen (Seite 2150)

Datentypen und Eigenschaften für Variablen (Seite 2157)

Bitmeldungen (Seite 2162)

Analogmeldungen (Seite 2163)

Archivvariablen (Seite 2168)

14.7.3.8 Datentypen und Eigenschaften für Variablen

Datentypen

Die folgende Tabelle zeigt, welche Datentypen in welcher Runtime unterstützt werden.

Datentyp	Runtime Advanced und Panels	Runtime Professional
Boolean	Ja	Ja
Byte (Signed Byte)	Ja	Ja
UByte (Unsigned Byte)	Ja	Ja
Short (Signed Int)	Ja	Ja
UShort (Unsigned Int)	Ja	Ja
Long (Signed Long)	Ja	Ja
ULong (Unsigned Long)	Ja	Ja
Float	Ja	Ja
Double	Ja	Ja
String	Ja	Ja
UString (Unicode String)	Nein	Ja
Raw (Raw Data Type)	Nein	Ja
Ref (Text Reference)	Nein	Ja
Date Time	Ja	Ja

Eigenschaften

Folgende Eigenschaften werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

- Adresse
- Inhalt (Kommentar)
- Verbindung
- Steuerungsvariable
- Datentyp
- Länge
- Grenzwerte:
  - Oberer Fehler
  - Analogmeldung erzeugen für Oberen Fehler
  - Obere Warnung
  - Analogmeldung erzeugen für Obere Warnung
  - Niedrig
  - Analogmeldung erzeugen für Untere Warnung
  - Unterer Fehler
  - Analogmeldung erzeugen für Unteren Fehler
- Lineare Skalierung
- Name
- Endwert Bediengerät
- Anfangswert Bediengerät
- Endwert PLC
- Anfangswert PLC
- Startwert

Siehe auch

Bildobjekte (Seite 2153)

14.7.3.9 Strukturen

Strukturelemente

Einleitung

Sie können ein Strukturelement im Editor "Strukturen" kopieren und im Editor "Strukturen" einer anderen Runtime einfügen.

Nicht unterstützte Datentypen

Wenn Sie ein Strukturelement mit einem Datentyp, der in der anderen Runtime nicht unterstützt wird, kopieren, wird der Datentyp im Editor "Struktur" ungültig.

Siehe auch

Strukturtypen (Seite 2159)

Datentypen und Eigenschaften für Strukturen (Seite 2160)

Grundlagen (Seite 2150)

Strukturtypen

Einleitung

Sie können einen Strukturtyp aus dem Ordner "Strukturen" kopieren und in den Ordner "Strukturen" in einer anderen Runtime einfügen.

Nicht unterstützte Steuerungstypen

Da kein Steuerungstyp in beiden Runtimes unterstützt wird, wird der Steuerungstyp im Ziel-Bediengerät ungültig.

Gerätefamilie

Die Gerätefamilie wird beim Kopieren an das Ziel-Bediengerät angepasst. Wenn Sie z. B. einen Strukturtyp aus Runtime Advanced und Panels nach Runtime Professional kopieren, wird der Eintrag "SmartCE" automatisch zu "SCADA".

Interne und externe Datentypen

In Runtime Advanced und Panels kann ein Strukturtyp entweder ausschließlich interne oder ausschließlich externe Datentypen enthalten. In Runtime Professional können in einem Strukturtyp interne und externe Datentypen enthalten sein.

Wenn Sie einen Strukturtyp, der interne und externe Datentypen enthält, von Runtime Professional nach Runtime Advanced und Panels kopieren, werden alle internen Datentypen ungültig. Da kein Kommunikationspartner in beiden Runtimes unterstützt wird, werden alle externen Datentypen ebenfalls ungültig.

Siehe auch

Strukturelemente (Seite 2158)

Datentypen und Eigenschaften für Strukturen

Datentypen

Die folgende Tabelle zeigt, welche Datentypen in welcher Runtime unterstützt werden.

Datentyp	Runtime Advanced und Panels	Runtime Professional
Boolean	Ja	Ja
Byte (Signed Byte)	Ja	Ja
UByte (Unsigned Byte)	Ja	Ja
Short (Signed Int)	Ja	Ja
UShort (Unsigned Int)	Ja	Ja
Long (Signed Long)	Ja	Ja
ULong (Unsigned Long)	Ja	Ja
Float	Ja	Ja
Double	Ja	Ja
String	Ja	Ja
UString (Unicode String)	Nein	Ja
Raw (Raw Data Type)	Nein	Ja
Ref (Text Reference)	Nein	Ja
Date Time	Ja	Nein

Eigenschaften

Folgende Eigenschaften werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

Strukturtypen:

- Kommentar: Inhalt
- Name
- Steuerungstyp
- Gerätefamilie

Strukturelemente:

- Kommentar: Inhalt
- Datentyp
- Länge
- Lineare Skalierung
- Name
- Endwert Bediengerät
- Anfangswert Bediengerät
- Endwert PLC
- Anfangswert PLC

- Fehler Obergrenze
- Fehler Untergrenze
- Startwert
- Bit Offset
- Offset

Siehe auch

Bildobjekte (Seite 2153)

14.7.3.10 Meldungen

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Unterstützte Objekte im Bild

In Runtime Advanced und Panels können Sie folgende Objekte zur Anzeige von Meldungen in ein Bild projektieren:

- Meldeanzeige
- Meldefenster
- Meldeindikator

In Runtime Professional können Sie ausschließlich eine Meldeanzeige projektieren.

Die Meldeanzeige ist in den Runtimes so unterschiedlich konzipiert, dass ein Kopieren einer Meldeanzeige zwischen den Runtimes nicht möglich ist.

Meldeblöcke

Die Meldeblöcke werden ausschließlich in Runtime Professional unterstützt.

Siehe auch

Bitmeldungen (Seite 2162)

Analogmeldungen (Seite 2163)

Steuerungsmeldungen und Systemmeldungen (Seite 2164)

Meldeklassen (Seite 2165)

Meldegruppen (Seite 2166)

Grundlagen (Seite 2150)

Grundlagen (Seite 2140)

Bitmeldungen

Einleitung

Sie können eine Bitmeldung im Editor "Meldungen" in der Registerkarte "Bitmeldungen" kopieren und in einer anderen Runtime einfügen. Wenn Sie eine Variable mit projektierten Bitmeldungen im Editor "Variablen" einfügen, werden die dazu gehörenden Bitmeldungen im Editor "Meldungen" eingefügt.

Unterstützte Eigenschaften

Folgende Eigenschaften werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

- ID
- Meldeklasse
- Meldegruppe
- Quittierung: Variable
- Quittierung: Bit
- Tooltip
- Trigger: Variable
- Trigger: Bit
- Ereignistext

Meldenummer

Wenn eine Meldenummer im Ziel-Bediengerät noch nicht existiert, dann wird sie beim Kopieren unverändert übernommen. Wenn die Meldenummer bereits existiert, wird automatisch die nächste freie Zahl als Meldenummer für die Bitmeldung vergeben.

Tooltip

Wenn Sie einen Tooltip mit mehr als 255 Zeichen nach Runtime Professional kopieren, wird der gesamte Tooltip ungültig.

Ereignistext

Der Ereignistext wird in folgenden Dialogen des Inspektorfensters angezeigt:

- Runtime Panels und Advanced:
  "Eigenschaften > Eigenschaften > Allgemein > Einstellungen > Ereignistext"
- Runtime Professional:
  "Eigenschaften > Eigenschaften >Texte > Einstellungen > Ereignistext"

Die Ereignistexte werden beim Kopieren mit allen Formatierungen in die entsprechenden Dialoge übernommen.

In folgenden Fällen werden die Ereignistexte ungültig:

- Wenn Sie einen Ereignistext mit mehr als 254 Zeichen nach Runtime Advanced und Panels kopieren.
- Wenn Sie einen Ereignistext mit einem dynamischen Parameter nach Runtime Panels und Advanced kopieren.
- Wenn Sie einen Ereignistext mit Textliste von Runtime Panels und Advanced nach Runtime Professional kopieren.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2161)

Analogmeldungen

Einleitung

Sie können eine Analogmeldung im Editor "Meldungen" in der Registerkarte "Analogmeldungen" kopieren und in einer anderen Runtime einfügen. Wenn Sie eine Variable mit projektierten Analogmeldungen im Editor "Variablen" einfügen, werden die dazu gehörenden Analogmeldungen im Editor "Meldungen" eingefügt.

Unterstützte Eigenschaften

Folgende Eigenschaften werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

- Meldenummer "ID"
- Ereignistext
- Meldeklasse
- Meldegruppe
- Tooltip
- Trigger: Variable
- Verzögerung
- Grenzwert
- Triggermodus
- Grenzwert synchronisieren
- Hysteresemodus
- Hysterese
- Hysterese in Prozent

Meldenummer "ID"

Wenn eine Meldenummer im Ziel-Bediengerät noch nicht existiert, dann wird sie beim Kopieren unverändert übernommen. Wenn die Meldenummer bereits existiert, wird automatisch die nächste freie Zahl als Meldenummer für die Analogmeldung vergeben.

Ereignistext

Der Ereignistext wird in folgenden Dialogen des Inspektorfensters angezeigt:

- Runtime Advanced und Panels:
  "Eigenschaften > Eigenschaften > Allgemein > Einstellungen > Ereignistext"
- Runtime Professional:
  "Eigenschaften > Eigenschaften > Texte > Einstellungen > Ereignistext"

Die Ereignistexte werden beim Kopieren mit allen Formatierungen in die entsprechenden Dialoge übernommen.

In folgenden Fällen werden die Ereignistexte ungültig:

- Wenn Sie einen Ereignistext mit mehr als 254 Zeichen nach Runtime Advanced und Panels kopieren.
- Wenn Sie einen Ereignistext mit einem dynamischen Parameter nach Runtime Panels und Advanced kopieren.
- Wenn Sie einen Ereignistext mit Textliste von Runtime Panels und Advanced nach Runtime Professional kopieren.

Tooltip

Wenn Sie einen Tooltip mit mehr als 255 Zeichen nach Runtime Professional kopieren, wird der gesamte Tooltip ungültig.

Verzögerung

Wenn Sie eine Verzögerung mit der Einheit "Minute" oder "Stunde" von Runtime Professional nach Runtime Advanced und Panels kopieren, wird die Einstellung ungültig.

Siehe auch

Kopieren von Variablen (Seite 2156)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2161)

Steuerungsmeldungen und Systemmeldungen

Steuerungsmeldungen

Da Steuerungsmeldungen in der Steuerung projektiert werden, sind Steuerungsmeldungen grundsätzlich nicht kopierbar.

Systemmeldungen

Da Systemmeldungen bediengeräteabhängig sind, sind Systemmeldungen grundsätzlich nicht kopierbar.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2161)

Meldeklassen

Einleitung

Sie können eine Meldeklasse im Editor "Meldungen" in der Registerkarte "Meldeklassen" kopieren und in einer anderen Runtime einfügen.

Vordefinierte Meldeklassen

Sie können vordefinierte Meldeklassen nicht kopieren und einfügen.

Archiv

In Runtime Advanced und Panels projektieren Sie ein Archiv, in das die Meldung abgelegt wird. In Runtime Professional legen Sie für jede Meldung einer Meldeklasse fest, ob die Meldung archiviert wird.

Beim Kopieren werden die Festlegungen zur Archivierung folgendermaßen umgesetzt:

- Kopieren von Runtime Advanced und Panels nach Runtime Professional:
  Wenn in Runtime Advanced und Panels ein Archiv festgelegt wurde, wird in Runtime Professional "Archiv" aktiviert.
  Wenn in Runtime Advanced und Panels kein Archiv festgelegt wurde, wird in Runtime Professional "Archiv" deaktiviert.
- Kopieren von Runtime Professional nach Runtime Advanced und Panels:
  Wenn in Runtime Professional "Archiv" aktiviert ist, müssen Sie in Runtime Advanced und Panels ein Archiv festlegen.
  Wenn in Runtime Professional "Archiv" deaktiviert ist, wird in Runtime Advanced und Panels der Eintrag <Kein Archiv> festgelegt.

Blinken

Die Einstellungen für das Blinkverhalten werden beim Kopieren nicht übernommen.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2161)

Meldegruppen

Kopieren und Einfügen

Sie können eine Meldegruppe im Editor "Meldungen" in der Registerkarte "Meldegruppen" kopieren und in einer anderen Runtime einfügen.

Meldegruppen aus Meldeklassen

Meldegruppen aus Meldeklassen werden ausschließlich in Runtime Professional unterstützt und können nicht kopiert und in Runtime Advanced und Panels eingefügt werden.

Hierarchisierung

Die Hierarchisierung von Meldegruppen wird ausschließlich in Runtime Professional unterstützt. Beim Kopieren zwischen den Runtimes wird die Hierarchisierung folgendermaßen umgesetzt:

- Kopieren von Runtime Advanced und Panels nach Runtime Professional
  Wenn Sie eine Meldegruppe in die Registerkarte "Meldegruppen" des Editors "Meldungen" kopieren, wird die Meldegruppe unabhängig von bereits vorhandenen Meldegruppen eingefügt.
  Wenn Sie eine Meldegruppe in eine Zeile mit einer vorhandenen Meldegruppe kopieren, wird die Meldegruppe als in der Hierarchie untergeordnete Meldegruppe eingefügt.
- Kopieren von Runtime Professional nach Runtime Advanced und Panels
  In Runtime Advanced und Panels kann keine Hierarchie abgebildet werden. Meldegruppen aus allen Hierarchieebenen werden unabhängig voneinander eingefügt.

ID der Meldegruppe

Wenn eine Meldenummer im Ziel-Bediengerät noch nicht existiert, dann wird sie beim Kopieren unverändert übernommen.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2161)

14.7.3.11 Archive

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Archive

Variablenarchive werden in beiden Runtimes unterstützt. Meldearchive und Verdichtungsarchive können Sie zwischen den Runtimes kopieren und einfügen.

Eigenschaften

Folgende Eigenschaften werden in Runtime Advanced und Panels und in Runtime Professional unterstützt:

Variablenarchive

- Name
- Kommentar
- Ablageort

Archivvariablen

- Name
- Prozessvariable
- Trigger
- Archivierungszyklus
- Oberer Grenzwert
- Unterer Grenzwert
- Kommentar

Ereignisse

Projektierte Ereignisse können nicht zwischen Runtime Advanced und Panels und Runtime Professional kopiert werden.

Siehe auch

Variablenarchive (Seite 2167)

Archivvariablen (Seite 2168)

Grundlagen (Seite 2150)

Variablenarchive

Kopieren und Einfügen

Sie können ein Variablenarchiv im Editor "Archive" in der Registerkarte "Variablenarchive" kopieren und in einer anderen Runtime einfügen. Alle Archivvariablen, die das Variablenarchiv enthält, werden automatisch mit eingefügt.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2166)

Archivvariablen

Einleitung

Sie können eine Archivvariable im Editor "Archive" in der Registerkarte "Variablenarchive" kopieren und in einer anderen Runtime einfügen.

Zusätzlich können Sie in Runtime Professional eine Archivvariable im Editor "Variablen" kopieren und in den Editor "Archive" in Runtime Advanced und Panels einfügen. Um eine Archivvariable in Runtime Professional zu kopieren, öffnen Sie den Editor "Variablen" und wählen Sie im oberen Arbeitsbereich die Registerkarte "Meldungen & Archive". Kopieren Sie im unteren Arbeitsbereich in der Registerkarte "Variablenarchive" die gewünschte Archivvariable.

In Archivvariablen verwendete Variablen

Wenn Sie eine Archivvariable mit einer Variable verbunden haben, wird die Archivvariable beim Kopieren dieser Variable mit in das Ziel-Bediengerät kopiert. Wenn im Ziel-Bediengerät kein entsprechendes Variablenarchiv vorhanden ist, wird ein neues Archiv angelegt. Wenn viele dieser zusätzlichen Daten mitkopiert werden, führt dies zu Performanceeinbußen.

Mehrfacharchivierung von Variablen

Beim Kopieren aus Runtime Professional nach Runtime Advanced und Panels wird die zu einer Archivvariablen gehörige Prozessvariable in folgenden Fällen ungültig:

- Die Variable wurde in Runtime Professional mehrmals archiviert.
- Mehrere Archivvariablen sind mit derselben Prozessvariable verbunden.

Trigger

Wenn Sie eine Archivvariable mit einem Zyklus einfügen, wird der Zyklus im Ziel-Bediengerät mit eingefügt.

In Runtime Professional projektieren Sie die Archivierungszyklen von Archivvariablen in der Form "Faktor x Zyklus", z. B. zweimal stündlich. In Runtime Advanced und Panels wird die Eigenschaft "Faktor" nicht unterstützt. Daher wird beim Kopieren von Runtime Advanced und Panels nach Runtime Professional die Eigenschaft Faktor auf die Standardeinstellung "1" gesetzt. Beim Kopieren von Runtime Professional nach Runtime Advanced und Panels ist in Runtime Advanced und Panels nur der Zyklus sichtbar und der Faktor wird nicht berücksichtigt. Wenn Sie z. B. einen Zyklus von "3 x 1h" aus Runtime Professional nach Runtime Advanced und Panels kopieren, wird dieser in Runtime Advanced und Panels als "1h" umgesetzt.

Archivvariablen aus dem Editor "Variablen" kopieren

Wenn Sie eine Archivvariable aus dem Editor "Variablen" in Runtime Professional in den Editor "Archive" in Runtime Advanced und Panels kopieren, wird die Eigenschaft "Prozessvariable" mit einer vorhandenen, gleichnamigen Variable verbunden. Wenn keine gleichnamige Variable vorhanden ist, wird die Eigenschaft "Prozessvariable" ungültig. Die Eigenschaft "Variablenarchiv" wird mit dem selektierten Variablenarchiv verbunden.

Wenn Sie eine Archivvariable innerhalb Runtime Professional aus dem Editor "Variablen" in den Editor "Variablen" eines anderen Bediengeräts kopieren, wird die Eigenschaft "Variablenarchiv" nicht mit übernommen.

Archivvariablen aus dem Editor "Archive" kopieren

Wenn Sie eine Archivvariable aus dem Editor "Archive" in Runtime Advanced und Panels in den Editor "Variablen" in Runtime Professional kopieren, wird die Eigenschaft "Variablenarchiv" nicht mit übernommen.

Wenn Sie eine Archivvariable aus dem Editor "Archive" in den Editor "Archive" eines anderen Bediengeräts kopieren, wird die Eigenschaft "Prozessvariable" mit einer vorhandenen, gleichnamigen Variable verbunden. Wenn keine gleichnamige Variable vorhanden ist, wird die Eigenschaft "Prozessvariable" ungültig. Die Eigenschaft "Variablenarchiv" wird mit dem selektierten Variablenarchiv verbunden.

In Verdichtungsarchivvariablen verwendete Archivvariablen

Wenn Sie eine Archivvariable, die in Runtime Professional in einer Verdichtungsarchivvariablen verwendet wird, nach Runtime Advanced und Panels kopieren, werden die Einstellungen für die Verdichtungsarchivvariable im Hintergrund mitkopiert. Wenn viele dieser nicht benötigten Einstellungen nach Runtime Advanced und Panels kopiert werden, führt dies zu Performanceeinbußen.

Siehe auch

Kopieren von Variablen (Seite 2156)

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2166)

14.7.3.12 Kopieren von VB-Skripten

Objekttypen, Eigenschaften und Methoden in VB-Skript

Die in Runtime Advanced und Panels und Runtime Professional unterstützten Objekte, Objekttypen, Eigenschaften und Methoden finden Sie in der VBS-Referenz.

Einfügen

Sie kopieren VB-Skripte

- über das Kontextmenü oder per Drag&Drop in der Projektnavigation,
- aus dem Code eines geöffneten VB-Skripts in ein anderes geöffnetes VB-Skript.

Objekte im VB-Skript aufgerufen

Sie kopieren ein VB-Skript. Im VB-Skript werden Objekte und deren Eigenschaften aufgerufen, die im Ziel-Bediengerät nicht vorhanden sind oder nicht unterstützt werden. Die Syntaxüberprüfung gibt einen Fehler aus. Korrigieren Sie die ungültigen Objektreferenzierungen.

Im VB-Skript aufgerufene Variablen

Wenn im kopierten VB-Skript Variablen aufgerufen werden, die im Ziel-Bediengerät nicht oder mit einem anderen Datentyp vorhanden sind, ergibt die Syntaxüberprüfung Fehler. Ersetzen Sie die ungültigen Variablen durch Variablen, die im Ziel-Bediengerät vorhanden sind.

Wenn Sie planen, Skripte in verschiedene Runtimes zu kopieren, verwenden Sie in den Runtimes die gleichen Variablen.

14.7.3.13 Kopieren von Rezepturen zwischen Runtime Advanced und Panels und Runtime Professional

Rezeptur und Rezepturelement

Beim Kopieren einer Rezeptur werden die Rezepturelemente und Rezepturdatensätze ebenfalls kopiert.

Hinweis

Inkonsistente Daten

Wenn das Bediengerät Eigenschaften nicht unterstützt z. B. den Datentyp, werden die Eigenschaften zwar formal übernommen, erhalten jedoch eine Fehlermarkierung.

Kontrollieren Sie die Eigenschaften und weisen Sie die markierten Eigenschaften neu zu.

Sie können ein Rezepturelement per Drag&Drop in die erste leere Zeile des Editors "Rezepturelemente" eines anderen Bediengeräts ziehen. Das Rezepturelement wird in die andere Rezeptur eingefügt. Wenn die Rezeptur nicht die Kommunikationsart "Variablen" besitzt, wird die verbundene Variable zwar übernommen, erhält jedoch eine Fehlermarkierung.

Besonderheiten beim Kopieren von WinCC Runtime Advanced und Panels nach WinCC Runtime Professional

Folgende Eigenschaften werden wie folgt vorbelegt und können anschließend editiert werden:

- Kommunikationsart "Variablen"
- Größentyp "begrenzt"
- Anzahl Datensätze

Der Ablageort der Rezeptur wird in "Datenbank" geändert.

Mit einem Rezepturelement wird die Verknüpfung kopiert, nicht jedoch die verknüpfte Variable. Sie müssen die Variable separat kopieren oder eine neue Variable gleichen Namens anlegen. Das gleiche gilt für die Textliste.

Hinweis

Inkonsistente Daten

Arrayvariablen werden in WinCC Runtime Professional nicht unterstützt. Legen Sie nach dem Kopieren der Rezeptur so viele Rezepturelemente und Variablen an, wie die verknüpfte Arrayvariable Arrayelemente enthält. Verknüpfen Sie die neuen Rezepturelemente und Variablen. Kopieren Sie die Werte der Quell-Rezepturdatensätze in die Ziel-Rezepturdatensätze.

Besonderheiten beim Kopieren von WinCC Runtime Professional nach WinCC Runtime Advanced und Panels

Folgende Eigenschaften werden geändert und sind nicht editierbar:

- Kommunikationsart "Variablen"
- Größentyp "begrenzt"
- Anzahl Datensätze je nach Bediengerät
- Ablageort "Datei" mit Standard-Pfad

Rezepturdatensatz und Rezepturabfrage

Sie können einen Rezepturdatensatz in eine andere Rezeptur kopieren, falls die andere Rezeptur die gleiche Anzahl Rezepturelemente enthält. Wenn die Datentypen unterschiedlich sind, wird der Wert zwar in den Ziel-Datensatz kopiert, erhält jedoch eine Fehlermarkierung.

Besonderheiten bei WinCC Runtime Professional

Sie ziehen eine Rezepturabfrage in die erste leere Zeile des Editors "Rezepturabfragen" eines anderen Bediengeräts. Die Rezepturabfrage wird kopiert, nicht jedoch die verknüpften Rezepturelemente und ihre Rezepturen.

14.7.3.14 Kopieren von Zyklen

Einleitung

Sie können einen Zyklus im Editor "Zyklen" kopieren und im Editor "Zyklen" einer anderen Runtime einfügen. Auch beim Kopieren von Variablen, Archivvariablen und Aufgaben werden die referenzierten Zyklen kopiert und im Editor "Zyklen" angezeigt.

Unterstützte Eigenschaften

Das Datum und die Uhrzeit der letzten Änderung werden beim Einfügen aktualisiert.

Eigenschaften

Die Eigenschaft "Startpunkt" wird ausschließlich in Runtime Professional unterstützt.

Zykluseinheiten

Runtime Professional und Runtime Advanced und PanelsIn unterstützen unterschiedliche Zykluseinheiten.

Siehe auch

Grundlagen (Seite 2150)

14.7.3.15 Protokolle

Objekte

Einleitung

Wenn Sie ein Objekt aus einem Protokoll kopieren und in ein anderes Protokoll einfügen, wird das Objekt an der gleichen Stelle im Protokoll eingefügt.

Unterstützte Eigenschaften

Die Objekte "Rezepturanzeige" und "Meldeanzeige" sind konzeptionell so unterschiedlich angelegt, dass sie nicht zwischen den verschiedenen Runtimes kopiert und eingefügt werden können.

Text- und Grafiklisten

Beim Kopieren symbolischer bzw. grafischer E/A-Felder werden automatisch Text- oder Grafiklisten mitkopiert und im Editor "Text- und Grafiklisten" angezeigt.

Objekte aus Bildern kopieren

Sie kopieren Objekte, die im Protokoll unterstützt werden, aus einem Bild in ein Protokoll. Dabei werden nicht unterstützte Einstellungen durch unterstützte Einstellungen ersetzt. Wenn Sie ein kopiertes Objekt wieder in das Bild kopieren, bleiben die geänderten Einstellungen aktiv.

Siehe auch

Objekte in Bildern (Seite 2152)

Kopieren eines Protokolls (Seite 2173)

Objekte in Protokollen (Seite 2173)

Grundlagen (Seite 2150)

Kopieren eines Protokolls

Einleitung

Sie kopieren ein Protokoll aus dem Ordner "Protokolle" und fügen es in einen anderen Ordner "Protokolle" ein.

Wasserzeichen

Wasserzeichen werden ausschließlich in Runtime Professional unterstützt. Wenn Sie ein Protokoll aus Runtime Professional nach Runtime Advanced und Panels kopieren, werden die Bereiche für die Wasserzeichen nicht mehr angezeigt.

Druckaufträge

Druckaufträge im Ordner "Protokolle" werden ausschließlich in Runtime Professional unterstützt und können nicht zwischen den Runtimes kopiert und eingefügt werden.

Siehe auch

Objekte (Seite 2172)

Objekte in Protokollen

Unterstützte Objekte

Die folgende Tabelle zeigt, welche Objekte in welcher Runtime unterstützt werden:

Basisobjekt	Runtime Advanced und Panels	Runtime Professional
Linie	Ja	Ja
Polygonzug	Ja	Ja
Ellipse	Ja	Ja
Kreis	Ja	Ja
Ellipsensegment	Nein	Ja
Kreissegment	Nein	Ja
Ellipsenbogen	Nein	Ja
Kreisbogen	Nein	Ja
Rechteck	Ja	Ja
Polygon	Ja	Ja
Textfeld	Ja	Ja
E/A-Feld	Ja	Ja
Symbolisches E/A-Feld	Ja	Nein
Datum/Uhrzeit-Feld	Ja	Nein
Grafisches E/A-Feld	Ja	Nein
Meldeanzeige	Nein	Ja

Basisobjekt	Runtime Advanced und Panels	Runtime Professional
Rezepturanzeige	Nein	Ja
Tabellenanzeige	Nein	Ja
f(t)-Kurvenanzeige	Nein	Ja
f(x)-Kurvenanzeige	Nein	Ja
Seitenzahl	Ja	Ja
Protokollname	Nein	Ja
Projektname	Nein	Ja
Hardcopy	Nein	Ja
ODBC-Datenbankfeld	Nein	Ja
ODBC-Datenbanktabelle	Nein	Ja
CSV-Provider Tabellen	Nein	Ja
CSV-Provider Kurven	Nein	Ja
Meldeprotokoll	Ja	Nein
Rezepturprotokoll	Ja	Nein
Control Daten	Nein	Ja
Control Hardcopy	Nein	Ja
Control Information	Nein	Ja

Siehe auch

Bildobjekte (Seite 2153)

14.7.3.16 Text- und Grafiklisten

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Bereiche bei Textlisten

Bei der Festlegung eines Bereichs werden folgende Einstellungen in beiden Runtimes unterstützt:

- Einzelner Wert
- Bereich

In Runtime Professional werden zusätzlich die folgenden Einstellungen unterstützt:

- Bis (<=)
- Von (>=)

Siehe auch

Kopieren von Text- und Grafiklisten (Seite 2175)

Grundlagen (Seite 2150)

Kopieren von Text- und Grafiklisten

Einleitung

Sie kopieren eine Text- oder Grafikliste im Editor "Text- und Grafiklisten" und fügen sie im gleichnamigen Editor einer anderen Runtime ein. Auch beim Kopieren symbolischer bzw. grafischer E/A-Felder werden automatisch Text- oder Grafiklisten mitkopiert und im Editor "Text- und Grafiklisten" angezeigt.

Listeneinträge

Beim Kopieren einer Text- oder Grafikliste werden die zugehörigen Listeneinträge mitkopiert. Sie können die Listeneinträge auch einzeln kopieren und einfügen.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2174)

14.7.3.17 Benutzerverwaltung

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional

Benutzer

Eigenschaften für das dynamische Anmelden und das Administrieren für Web Client werden ausschließlich in Runtime Professional unterstützt.

Benutzergruppen

Die Eigenschaften "Anzeigename" und "Nummer" werden nur in Runtime Advanced und Panels unterstützt.

Eigenschaften für das Administrieren für Web Client werden ausschließlich in Runtime Professional unterstützt.

Siehe auch

Kopieren von Benutzern, Benutzergruppen und Berechtigungen (Seite 2175)

Grundlagen (Seite 2140)

Kopieren von Benutzern, Benutzergruppen und Berechtigungen

Kopieren und Einfügen

Im Editor "Benutzerverwaltung" kopieren Sie Benutzer, Benutzergruppen und Berechtigungen.

Wenn Sie einen Benutzer kopieren, wird die zugehörige Benutzergruppe nicht kopiert.

Wenn Sie eine Benutzergruppe kopieren, werden die zugehörigen Berechtigungen nicht kopiert.

Standardeinträge und Systemberechtigungen

Folgende Benutzer, Benutzergruppen und Berechtigungen sind vordefiniert und können nicht gelöscht und nur eingeschränkt bearbeitet werden:

- Vordefinierter Benutzer "Administrator"
- Vordefinierte Gruppen "Administrator" und "Bediener"
- Vordefinierte Berechtigungen, z. B. "Bedienen", "Enable remote control"

Die vordefinierten Einträge kopieren und fügen Sie in eine andere Runtime ein. Die kopierten Einträge werden als benutzerdefinierte Einträge eingefügt und können bearbeitet werden.

Siehe auch

Unterschiede zwischen Runtime Advanced und Panels und Runtime Professional (Seite 2175)

14.7.3.18 Aufgabenplaner

Kopieren von Aufgaben

Einleitung

Sie kopieren eine Aufgabe im Editor "Aufgabenplaner" und fügen sie im Editor "Aufgabenplaner" eines anderen Geräts ein.

Aufgabentyp

Der Aufgabentyp "Druckauftrag" wird ausschließlich in Runtime Professional unterstützt. Typen, die nicht in beiden Runtimes unterstützt werden, werden nach dem Einfügen ungültig.

Siehe auch

Ereignisse für Aufgabenplaner (Seite 2177)

Grundlagen (Seite 2150)

Systemfunktionen (Seite 2151)

Ereignisse für Aufgabenplaner

Ereignisse

Die folgende Tabelle zeigt, welche Ereignisse in welcher Runtime unterstützt werden:

Ereignis	Runtime Advanced und Panels	Runtime Professional
Bildwechsel	Ja	Nein
Benutzerwechsel	Ja	Nein
Meldepufferüberlauf	Ja	Nein
Runtime-Stopp	Ja	Ja
Beim Öffnen eines Dialogs	Ja	Nein
Beim Schließen eines Dialogs	Ja	Nein
Deaktiviert	Ja	Ja
Einmal	Ja	Ja
250 Millisekunde	Nein	Ja
500 Millisekunde	Nein	Ja
1 Sekunde	Nein	Ja
2 Sekunde	Nein	Ja
5 Sekunde	Nein	Ja
10 Sekunde	Nein	Ja
1 Minute	Ja	Ja
5 Minute	Nein	Ja
10 Minute	Nein	Ja
1 Stunde	Ja	Ja
Täglich	Ja	Ja
Wöchentlich	Ja	Ja
Monatlich	Ja	Ja
Jährlich	Ja	Ja
Variablentrigger	Nein	Ja

Siehe auch

Bildobjekte (Seite 2153)

14.7.4 Kopieren zwischen unterschiedlichen RT- und ES-Versionen

Einleitung

Sie können Projektdaten, z. B. Bilder, Objekte oder Variablen, zwischen Projekten mit unterschiedlichen WinCC-Versionen kopieren und einfügen.

Wenn Sie zwischen verschiedenen WinCC-Versionen kopieren, bleiben alle Projektierungen erhalten, die auch in der Ziel-Version unterstützt werden. Projektierungen, welche die Ziel-Version nicht unterstützt, werden farbig als ungültig markiert.

Kopieren zwischen von Bediengeräten unterschiedlicher Geräteversionen

WinCC unterstützt alle Projektierungen aus einer vorhergehenden WinCC-Version.

Beim Kopieren innerhalb unterschiedlicher ES-Versionen gilt Folgendes:

- Alle Projektierungen bleiben erhalten, die auch in der jeweiligen RT-Version unterstützt werden.
- Für Projektierungen die ausschließlich in der WinCC-Version des Ziel-Projekts unterstützt werden, werden Standardeinstellungen festgelegt.
- Projektierungen, welche die jeweilige RT-Version nicht unterstützt, werden farbig als ungültig markiert oder nicht angezeigt.
  Wenn Sie ein Objekt unverändert zurück in das Quell-Bediengerät kopieren, sind alle ursprünglich im Quell-Bediengerät festgelegten Eigenschaften und Einstellungen wieder aktiv.

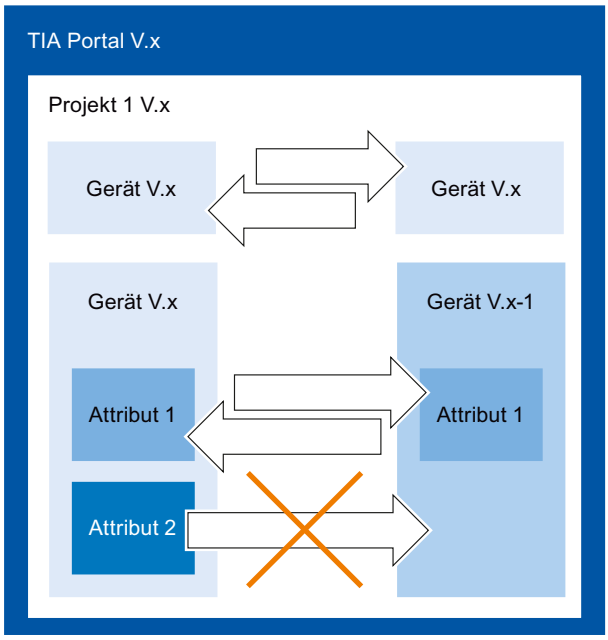

- Das Bediengerät muss für die aktuelle Runtime-Version gültig sein.

Kopieren zwischen unterschiedlichen ES-Versionen

Um zwischen zwei TIA-Projekten zu kopieren, öffnen Sie eine zweite Instanz Ihres TIA Portals. Sie können nur zwischen Projekten der gleichen ES-Version kopieren. Die ES-Version eines Projekts erkennen Sie an der Dateiendung *.ap<Versionsnummer>.

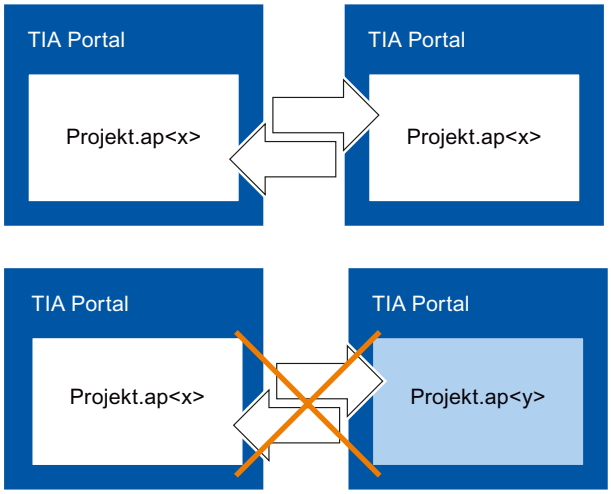

14.8 WinCC-Versionskompatibilität nutzen

14.8.1 Grundlagen zur Versionskompatibilität

Einleitung

Bestehende Projekte bearbeiten Sie wie folgt mit WinCC:

- Sie bearbeiten, übersetzen und laden bestehende Projekte mit dem Funktionsumfang der WinCC-Vorgängerversion. Diese Projekte können Sie anschließend mit der WinCC-Vorgängerversion weiter bearbeiten.
- Sie rüsten bestehende Projekte hoch und nutzen die Funktionen der aktuellen WinCC-Version.

Hinweis

WinCC-Funktionen

Wenn Sie ein Projekt einer vorhergehenden WinCC-Version bearbeiten, stehen Ihnen nur die Funktionen und Bediengeräte dieser vorhergehenden WinCC-Version zur Verfügung.

Versionen in WinCC

In WinCC arbeiten Sie mit unterschiedlichen Arten von Versionen:

- WinCC-Version
  Die am Projektierungs-PC installierte WinCC-Version, z. B. WinCC V12.
- Projektversion
  Projekte werden mit der WinCC-Version erstellt, die auf dem Projektierungs-PC installiert ist.
  Wenn Sie z. B. mit der aktuellen WinCC-Version ein WinCC-Projekt aus der Vorgängerversion bearbeiten, wird in der Projektnavigation die Versionsbezeichnung hinter dem Projektnamen angezeigt.
- Runtime-Version
  In WinCC können Sie Bediengeräte mit unterschiedlichen Runtime-Versionen projektieren. Die Runtime-Version legen Sie einmal für ein Bediengerät fest. Die Geräteversion muss zur Runtime-Version passen.
- Geräteversionen
  Abhängig vom verwendeten Bediengerät setzt sich das Image aus Betriebssystem und / oder Runtime-Software zusammen. Für jedes Bediengerät stellt WinCC verschiedene Images zur Verfügung, die entsprechend der Projektierung bei Bedarf auf das Bediengerät geladen werden. Die Geräteversion entspricht einem bestimmten Image. Die Geräteversion muss zur Projektierung passen.

Kompatibilität von WinCC-Versionen, Runtime-Versionen und Geräteversionen

Die folgende Abbildung zeigt das Zusammenwirken der Versionen im TIA Portal:

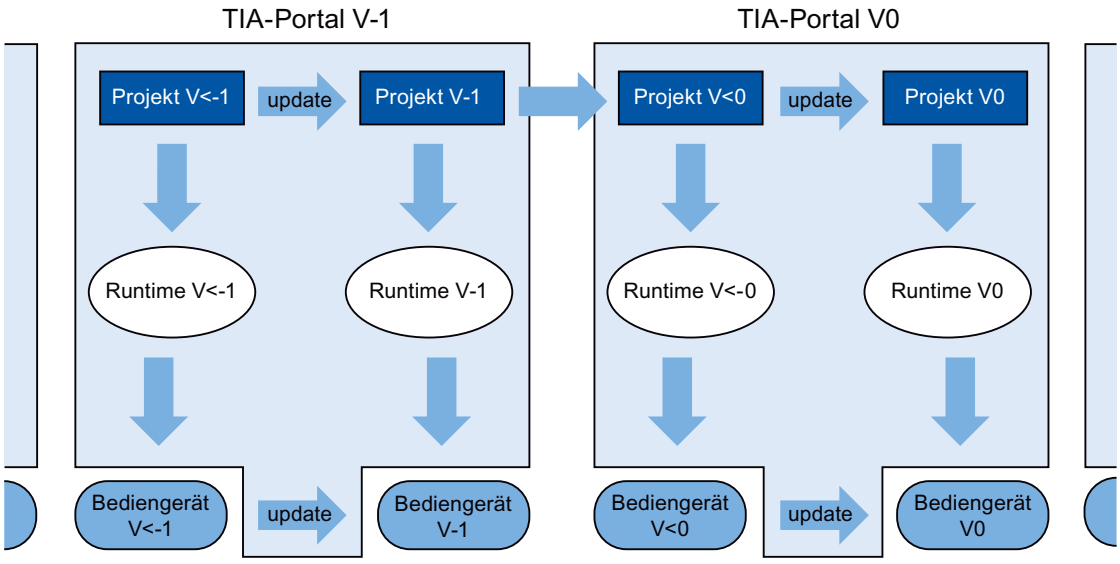

Anlegen von Projekten

Wenn Sie ein Projekt in WinCC neu anlegen, öffnen und bearbeiten Sie es mit der WinCC-Version, in der Sie es angelegt haben.

Speichern

Um ein Projekt einer vorhergehenden WinCC-Version für diese Version wieder abzuspeichern, speichern Sie wie gewohnt. Erst wenn Sie manuell das Projekt auf Ihre WinCC-Version hochrüsten, können Sie das Projekt in der vorhergehenden WinCC-Version nicht mehr öffnen.

Um ein Projekt einer vorhergehenden WinCC-Version in der aktuellen Version zu speichern, rüsten Sie das Projekt auf Ihre WinCC-Version hoch. Sie können dieses Projekt dann nicht mehr mit einer vorhergehenden WinCC-Version bearbeiten.

Übersetzen, Simulieren und Laden

Wenn Sie ein Projekt einer WinCC-Vorgängerversion verwenden, können Sie mit Ihrer aktuellen Version Runtime-Daten dieser Vorgängerversion erzeugen. Dadurch laden Sie auch Bediengeräte, die mit Ihrer WinCC-Version nicht mehr kompatibel sind.

Kopieren innerhalb von Projekten mit verschiedenen WinCC-Versionen

Wenn Objekte und Projektierungen in der Ziel-Version auch zur Verfügung stehen, kopieren Sie diese über die Zwischenablage oder Drag&Drop nach Bedarf.

Öffnen, Bearbeiten und Speichern von Projekten einer vorhergehenden WinCC-Version

Sie öffnen und bearbeiten Projekte vorhergehender WinCC-Versionen nach Bedarf. Sie nutzen dabei nur die Funktionen der vorhergehenden WinCC-Version. Nach Ihrer Bearbeitung kann das Projekt wieder in der vorhergehenden WinCC-Version bearbeitet und gespeichert werden.

Übersetzen, Laden und Simulieren von Projekten mit vorhergehenden WinCC-Versionen

Sie übersetzen, laden und simulieren Projekte vorhergehender WinCC-Versionen nach Bedarf. Dabei stellt Ihre aktuelle WinCC-Version die Runtimes und Geräteversionen der entsprechenden WinCC-Version zur Verfügung.

14.8.2 Grundlagen zur Versionskompatibilität

Einleitung

In der aktuellen WinCC-Version können Sie Projekte der WinCC-Vorgängerversion öffnen und hochrüsten.

Versionen in WinCC

In WinCC arbeiten Sie mit unterschiedlichen Arten von Versionen:

- WinCC-Version
  Die am Projektierungs-PC installierte WinCC-Version, z. B. WinCC V14.
- Projektversion
  Projekte werden mit der WinCC-Version erstellt, die auf dem Projektierungs-PC installiert ist.
  Wenn Sie z. B. mit der aktuellen WinCC-Version ein WinCC-Projekt aus der Vorgängerversion öffnen, wird in der Projektnavigation die Versionsbezeichnung hinter dem Projektnamen angezeigt.
- Runtime-Version
  Abhängig von der Projektversion wird das Runtime-Projekt für eine bestimmte Runtime-Version erzeugt. Die Geräteversion muss zur Runtime-Version passen.

Kompatibilität von WinCC-Versionen

Projekte aus einer WinCC-Vorgängerversion können Sie in der aktuellen WinCC-Version ausschließlich öffnen und hochrüsten. Um das Projekt einer WinCC - Vorgängerversion für die entsprechende Runtime-Version zu projektieren, verwenden Sie die passende WinCC-Version auf Ihrem Projektierungs-PC.

Übersetzen, Simulieren und Laden

Ein Projekt einer WinCC-Vorgängerversion können Sie in der aktuellen WinCC-Version nur übersetzen, simulieren und laden, wenn Sie das Projekt auf Ihre WinCC-Version hochrüsten.

Um das Projekt einer WinCC - Vorgängerversion für die entsprechende Runtime-Version zu übersetzen, simulieren und zu laden, verwenden Sie die passende WinCC-Version auf Ihrem Projektierungs-PC.

Siehe auch

Projekt hochrüsten (Seite 2183)

Geräteversion ändern (Seite 2188)

Wechsel zwischen Geräteversionen (Seite 2187)

Kopieren zwischen unterschiedlichen RT- und ES-Versionen (Seite 2177)

14.8.3 Projekt hochrüsten

Einleitung

Wenn die Projektversion älter als die WinCC-Version ist, wird in der Projektnavigation die Versionsbezeichnung angezeigt. Ihre WinCC-Version enthält auch die Vorgängerversion, mit der Sie bei Bedarf Projekte bearbeiten. Um in einem Projekt die Funktionen und Möglichkeiten Ihrer WinCC-Version zu nutzen, rüsten Sie das Projekt auf Ihre WinCC-Version hoch. Ändern Sie im Anschluss auch Ihre Geräte- oder Runtime-Version/-en entsprechend der neuen Projektversion.

Hinweis

WinCC-Versionskompatibilität

Wenn Sie ein Projekt auf Ihre WinCC-Version hochrüsten, können Sie das Projekt nicht mehr mit der vorhergehenden Version bearbeiten.

Voraussetzungen

- Die Projektversion ist ein Vorgänger Ihrer WinCC-Version.
- Sie haben Schreibrechte auf Ihrem Projektlaufwerk.
- Die Speicherkapazität auf Ihrem Projektlaufwerk ist ausreichend für ein weiteres Projekt dieser Größe.

Vorgehen

Um ein Projekt auf Ihre WinCC-Version hochzurüsten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Projekt hochrüsten".
   Ein Dialog wird geöffnet.

3. Klicken Sie "Bestätigen".
   Das Projekt wird geschlossen und ein Fortschrittsbalken angezeigt.

   Hinweis

   Wenn im Projekt nicht unterstütze Geräte projektiert sind, wird es nicht hochgerüstet. Eine Fehlermeldung informiert Sie darüber.

   Um ein Projekt dennoch hochzurüsten, ändern Sie die Geräteversion des betroffenen Geräts oder das Gerät selbst über den Menübefehl "Gerät/Version ändern".

   Wenn das Projekt hochgerüstet ist, wird eine Meldung ausgegeben.

4. Übersetzen Sie anschließend jedes Gerät im Projekt mit dem Befehl "Bearbeiten > Übersetzen".

   Hinweis

   Reihenfolge der Übersetzung

   Übersetzen Sie zuerst die Steuerungen und danach die Bediengeräte in Ihrem Projekt. Somit stellen Sie sicher, dass die Steuerungsdaten, die für das Übersetzen der Bediengeräte notwendig sind, zur Verfügung stehen.

Ergebnis

- Im Projektlaufwerk wurde das Projekt in der vorhergehenden WinCC-Version mit der entsprechenden Dateiendung gespeichert.
- Im Projektlaufwerk wird das Projekt mit der aktuellen WinCC-Version und der entsprechenden Dateiendung angezeigt.
- Das Projekt wird in der Projektnavigation ohne WinCC-Versionsbezeichnung angezeigt.

Um die Funktionen der neuen WinCC-Version zu nutzen, ändern Sie im nächsten Schritt Ihre Geräte- oder Runtime-Version/-en entsprechend der neuen Projektversion.

Siehe auch

Grundlagen zur Versionskompatibilität (Seite 2182)

14.8.4 Globale Bibliothek hochrüsten

Einleitung

Um Objekte einer Globalen Bibliothek in einem Projekt bearbeiten zu können, muss die Globale Bibliothek in der Produktversion des Projekts vorliegen. Sie können jede Globale Bibliothek aus einer früheren Produktversion auf die aktuelle Produktversion hochrüsten. Eine entsprechende Aufforderung erhalten Sie beim Öffnen der globalen Bibliothek.

Voraussetzungen

- Die Version der Globalen Bibliothek ist ein Vorgänger Ihrer WinCC-Version.
- Sie haben Schreibrechte auf Ihrem Projektlaufwerk.
- Alle Typen in der Bibliothek sind freigegeben.

Hinweis

Hochrüsten einer Anwenderbibliothek

Wenn Sie eine Anwenderbibliothek aus einer früheren Version von WinCC verwenden möchten, müssen Sie diese hochrüsten. Stellen Sie sicher, dass alle Typen in der Bibliothek freigegeben sind. Für Hochrüsten darf kein in Bibliothek enthaltener Typ im Zustand "in Bearbeitung" sein.

Vorgehen

Um eine globale Bibliothek aus dem TIA Portal V12.x oder früher hochzurüsten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek.
   Der Dialog "Globale Bibliothek hochrüsten" wird geöffnet.
2. Klicken Sie auf "OK".

Eine Kopie der globalen Bibliothek wird erstellt und hochgerüstet. Die Kopie der globalen Bibliothek erhält den Namenszusatz "_V13". Die globale Bibliothek wird geöffnet.

Ergebnis

Die Globale Bibliothek ist mit der entsprechenden Dateiendung gespeichert.

Siehe auch

Grundlagen zur Versionskompatibilität (Seite 2182)

Bibliotheken verwalten (Seite 2023)

14.8.5 Projekt einer WinCC-Vorgängerversion bearbeiten

Einleitung

WinCC bietet die Möglichkeit, Projekte einer vorhergehenden WinCC-Version zu bearbeiten. Während Sie ein Projekt einer vorhergehenden WinCC-Version bearbeiten, stehen Ihnen nur die Funktionen dieser Version zur Verfügung. Um die Funktionen Ihrer aktuellen WinCC-Version für dieses Projekt zu nutzen, rüsten Sie das Projekt auf Ihre WinCC-Version hoch.

Hinweis

Wenn Sie ein Projekt hochrüsten, werden die Inhalte in einem neuen Projekt mit der aktuellen Projektversion gespeichert. Das Ursprungsprojekt wird nicht überschrieben und kann weiterhin mit einer kompatiblen älteren Version des TIA Portals bearbeitet werden.

Voraussetzung

- Ein Projekt einer vorhergehenden WinCC-Version ist angelegt.
- Auf dem Projektierungs-PC ist die aktuelle WinCC-Version installiert.

Vorgehen

Um ein Projekt einer vorhergehenden WinCC-Version zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Projekt.
2. Bearbeiten Sie das Projekt mit den Funktionen der vorhergehenden WinCC-Version.
3. Speichern Sie das Projekt.
4. Übersetzen Sie das Projekt.
5. Laden und Simulieren Sie das Projekt.
6. Öffnen Sie das Projekt bei Bedarf mit der vorhergehenden WinCC-Version und bearbeiten Sie es dort weiter.

Ergebnis

Die geänderten Projektdaten können an einem anderen Projektierungs-PC mit der vorhergehenden WinCC-Version weiterbearbeitet werden. Das Runtime-Projekt wurde in der entsprechenden Runtime-Version erzeugt und geladen.

14.8.6 Wechsel zwischen Geräteversionen

Auswahl der Geräteversion

Wenn Sie ein neues Bediengerät projektieren, wählt WinCC automatisch die neueste Geräteversion aus.

Wenn Sie auf einem Bediengerät eine andere Geräteversion verwenden wollen als in WinCC eingestellt wurde, übertragen Sie ein Image auf das Bediengerät. WinCC liefert die benötigten Images der unterstützten Bediengeräte mit.

Informationen zu den in WinCC verwendeten Geräteversionen finden Sie als FAQ im Internet unter der Beitrags-ID 21742389.

ACHTUNG
Das Ändern der Geräteversion löscht alle Daten auf dem Bediengerät. Wenn Sie die Geräteversion ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb vorhandene Runtime-Daten und ändern Sie die Geräteversion nachträglich.

Aufbau Bediengerät

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts:

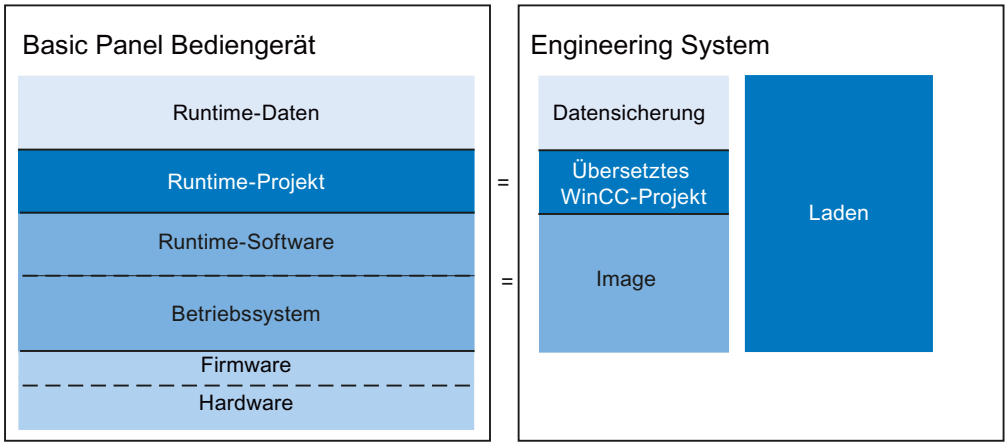

Siehe auch

Grundlagen zur Versionskompatibilität (Seite 2182)

14.8.7 Geräteversion ändern

Einleitung

Abhängig von der gewünschten Runtime-Version wählen Sie für Ihre Projektierung die passende Geräteversion.

Hinweis

Auswahl Geräteversionen

Die bereitgestellte Auswahl von Geräteversionen ist abhängig von der Projektversion.

Voraussetzungen

- Ein Projekt ist angelegt und geöffnet.
- Das Projekt enthält ein Bediengerät.

Vorgehen

Um die Geräteversion zu ändern, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Geräte & Netze".
   Der Editor wird geöffnet.
2. Selektieren Sie das gewünschte Bediengerät in der Gerätesicht.
3. Wählen Sie im Kontextmenü des Bediengeräts den Eintrag "Gerät / Version ändern".
   Ein Dialog wird geöffnet.
4. Selektieren Sie das gewünschte Bediengerät.
5. Wählen Sie unter "Version" die gewünschte Geräteversion.
6. Betätigen Sie Ihre Auswahl mit "OK".

Ergebnis

Sie haben die Geräteversion des Gerätes im WinCC-Projekt geändert.

ACHTUNG
Das Ändern der Geräteversion löscht beim Laden alle Daten auf dem Bediengerät. Wenn Sie die Geräteversion ändern und das Projekt übersetzen und Laden, werden auf dem Bediengerät Daten gelöscht. Sichern Sie deshalb vor dem Laden vorhandene Runtime-Daten.

Siehe auch

Grundlagen zur Versionskompatibilität (Seite 2182)

14.9 Farben zentral verwalten

14.9.1 Grundlagen zur Farbverwaltung

Einführung

WinCC bietet die Möglichkeit, die innerhalb eines Projekts verwendeten Farben zentral zu ändern. Der Dialog "Farbreferenz ändern" enthält eine hierarchische Übersicht aller im gewählten Objekt enthaltenen farbrelevanten Eigenschaften. In der Darstellung können Sie innerhalb der angezeigten Anzeige- und Bedienobjekte navigieren und erhalten damit einen Überblick über alle verwendeten Farben. Über Suche und Filter spezifizieren Sie eine Farbauswahl und ersetzen diese bei Bedarf mit anderen Farben.

Unterstützte Objekte

Über den Dialog "Farbreferenz ändern" haben Sie Zugriff auf alle im Projekt verwendeten und projektierbaren Farben. Ausgenommen davon sind Farben, die wie folgt verwendet werden:

- In Typen und Instanzen aus einer Bibliothek
- In Bildbausteinen
- In Skripten
- In Designs
- In Bildern mit Schreibschutz

14.9.2 Farben suchen und ersetzen

Einführung

Der Umfang, der im Dialog "Farbreferenzen ändern" angezeigten Objekte ist abhängig vom Ort, an dem der Dialog aufgerufen wird:

- Wenn Sie ein Bediengerät selektieren und den Dialog aufrufen, werden alle am Bediengerät verwendeten Farbreferenzen angezeigt.
- Wenn Sie ein Anzeigeobjekt innerhalb eines Bildes selektieren und den Dialog aufrufen, werden nur die Farbreferenzen angezeigt, die im Anzeigeobjekt enthalten sind.

Voraussetzung

- Ein Projekt ist angelegt.
- Bilder sind angelegt.

Vorgehen

1. Selektieren Sie das Objekt, das die gewünschten Farbreferenzen enthält.
2. Wählen Sie im Kontextmenü "Objektfarbe ändern".
   Ein Dialog wird geöffnet.

3. Wählen Sie die Farbe, die Sie ändern wollen:
   - Klicken Sie auf das Farbfeld im Suchfeld.
     Die Farbauswahl des Projekts wird geöffnet.
   - Um eine Standardfarbe oder eine benutzerdefinierte Farbe zu wählen, klicken Sie auf "Mehr Farben".
   - Um eine Farbe direkt aus dem selektierten Objekt zu verwenden, ziehen Sie ein Farbfeld per Drag&Drop aus der Übersichtstabelle auf das Suchfeld.
   - Um auch ähnliche Farben auszuwählen, stellen Sie eine Toleranz ein.

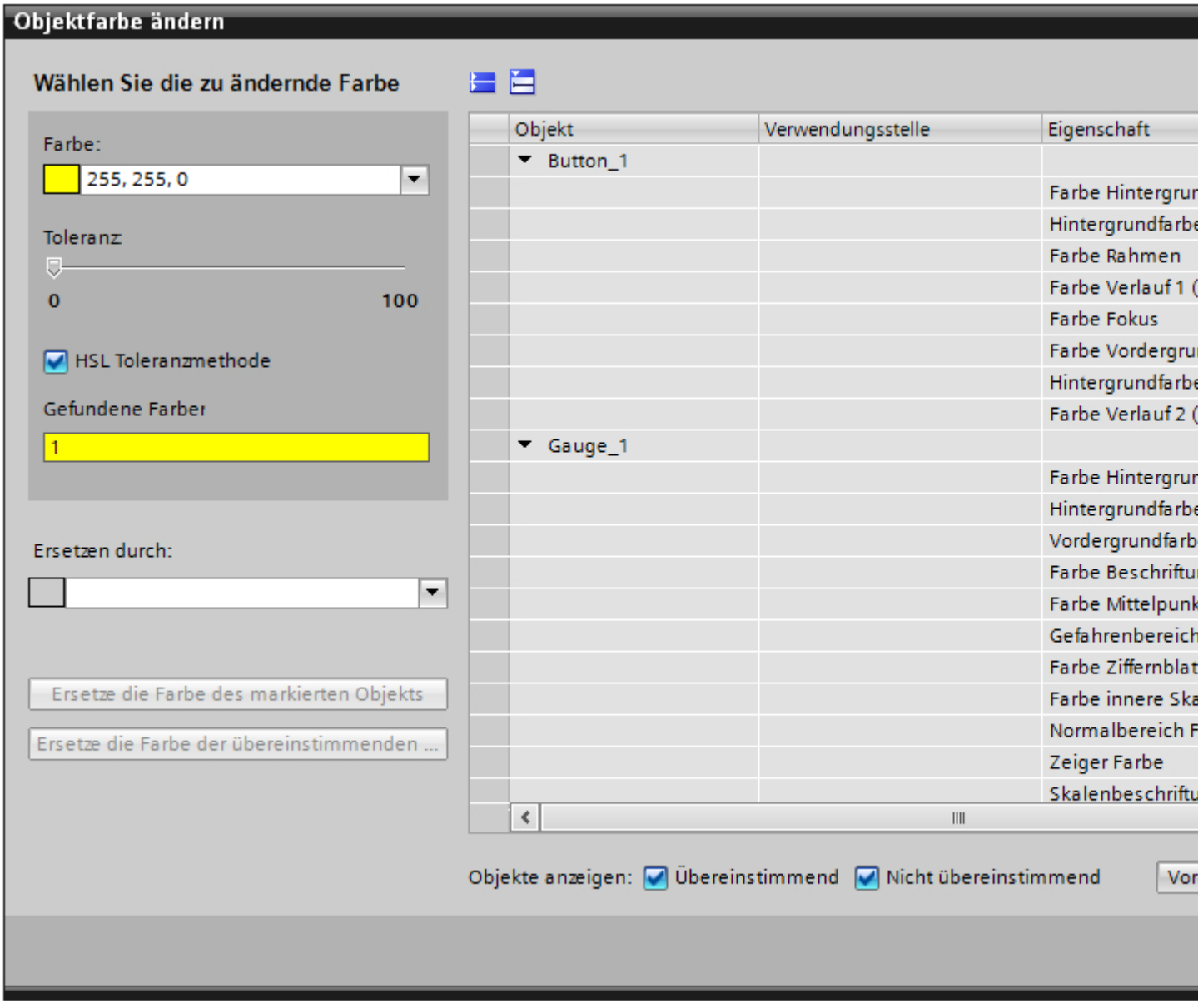

4. Filtern Sie die angezeigte Tabelle.

5. Wählen Sie in der Tabellenspalte "Ersetzen mit" für die einzelnen Eigenschaften die jeweils neue Farbe.
6. Klicken Sie "OK".

Ergebnis

Im selektieren Objekt sind die neuen Farbreferenzen projektiert.

14.10 Dateien über den Editor "HMI Dateien" laden

Einleitung

Mithilfe des Editors "HMI Dateien" laden Sie Dateien beliebigen Datenformats von Ihrem Computer in Ihr Projekt. Diese Dateien werden beim "Laden in Gerät" auf den PC übertragen. So verknüpfen Sie nützliche Zusatzinformationen wie Systemhandbücher, Videotutorials usw. mit dem jeweiligen Projekt oder verteilen sie verschiedene Geräte. Nach Bedarf überschreiben Sie die geladenen Dateien jederzeit mit aktuellen Dateien oder löschen sie aus dem Projekt.

Um den Arbeitsbereich des Editors "HMI Dateien" übersichtlich zu gestalten, legen Sie Ordner an. Die Dateien legen Sie dann in den Ordnern ab.

Wenn Sie diese Dateien oder Ordner auf Ihrem Computer ändern, erscheinen die Änderungen auch im Editor "HMI Dateien".

Hinweis

Der Editor "HMI Dateien" ist nur in der Projektnavigation sichtbar, wenn im Projekt ein Gerät mit Runtime Professional angelegt ist.

Voraussetzung

Die Dateien, die Sie in das Projekt laden möchten, befinden sich auf Ihrem Computer.

Dateien laden

1. Doppelklicken Sie in der Projektnavigation auf "Gemeinsame Daten".
2. Doppelklicken Sie auf den Editor "HMI Dateien".
   Der Editor wird geöffnet.

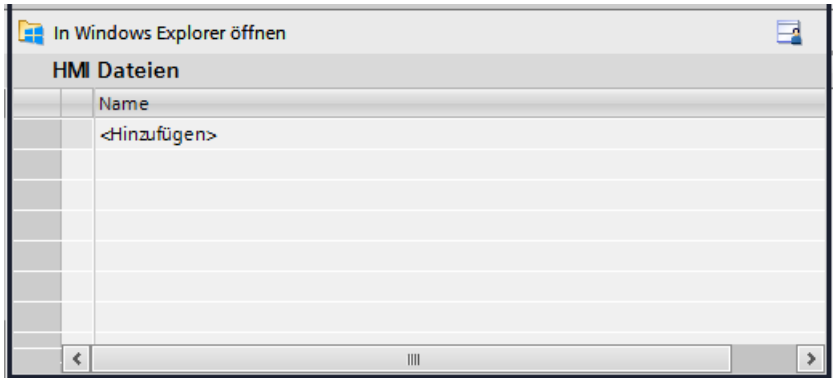

3. Klicken Sie im Arbeitsbereich auf die Schaltfläche "Hinzufügen".
   Der Windows Explorer wird geöffnet.
4. Navigieren Sie zum Verzeichnis, in dem die Dateien gespeichert sind, die Sie in das Projekt laden möchten.

5. Selektieren Sie eine oder mehrere Dateien.
6. Klicken Sie auf "Öffnen".
   Die Dateien erscheinen im Arbeitsbereich nummeriert in alphabetischer Reihenfolge und werden unter "<TIAProjectname>\UserFiles\HMI_globalFiles" gespeichert.

Alternativ können Sie den Windows Explorer über die Schaltfläche "In Windows Explorer öffnen" aufrufen und die gewünschten Dateien per Drag&Drop aus dem Windows Explorer in den Arbeitsbereich ziehen.

Hinweis

Um direkt in den Ablageordner "<TIAProjectname>\UserFiles\HMI_globalFiles" zu gelangen, klicken Sie auf die Schaltfläche "In Windows Explorer öffnen".

Dateien überschreiben

Sie überschreiben Dateien, indem Sie gleichnamige Dateien in die Projektablage laden.

1. Laden Sie die Dateien wie unter "Dateien laden" beschrieben.
2. Bestätigen Sie die Abfrage mit "Ja", damit die Dateien überschrieben werden.

Dateien löschen

1. Markieren Sie im Arbeitsbereich die Zeile der zu löschenden Datei.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Sie löschen mehrere Dateien, indem Sie mehrere Zeilen markieren und den Befehl "Löschen" aus dem Kontextmenü ausführen.

Ordner anlegen

Sie müssen zuerst Ordner anlegen, um sie dann mit Dateien zu füllen.

1. Markieren Sie eine leere Zeile im Arbeitsbereich.
2. Wählen Sie im Kontextmenü den Befehl "Neuen Ordner anlegen".
   Der Dialog "Neuen Ordner anlegen" wird geöffnet.
3. Geben Sie einen Namen für den Ordner ein.
4. Bestätigen Sie Ihre Eingaben mit "OK".
   Der Ordner erscheint im Arbeitsbereich.

Hinweis

Sie legen Unterordner an, indem Sie die Zeile mit dem jeweiligen Ordner selektieren und den Befehl "Neuen Ordner anlegen" aus dem Kontextmenü ausführen.

Dateien in Ordnern ablegen

1. Selektieren Sie im Arbeitsbereich einen Ordner, den Sie mit Dateien füllen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Objekt einfügen".
   Der Windows Explorer wird geöffnet.
3. Laden Sie die Dateien wie unter "Dateien laden" beschrieben.
   Die Dateien werden im Ordner abgelegt.

Ergebnis

Die Dateien und Ordner sowie deren Änderungen werden an alle Geräte im Projekt nach folgenden Aktionen übertragen:

- Übersetzen des Projekts
- Laden des Projekts
- Starten der Runtime
- Starten der Simulation

Dabei wird der Inhalt des Ordners "<TIAProjectname>\UserFiles\HMI_globalFiles" unter "<RTProject>\AdditionalFiles" gespiegelt.

Hinweis

Wenn eine Datei gerade in Runtime verwendet wird, wird diese Datei im Ordner "<RTProject>\AdditionalFiles" nicht überschrieben. Eine Warnung informiert Sie darüber.

Siehe auch

Übersicht zum Übersetzen und Laden von Projekten (Seite 2363)

14.11 Austausch von Steuerungsdaten aus anderen Projekten

Inter-Project-Engineering?

Mit der Funktion IPE (Inter Project Engineering) können Sie Steuerungsdaten aus anderen TIA Portal-Projekten einlesen und zur Projektierung verwenden.

Mit IPE können Sie die Kommunkation von HMI-Projektierung und PLC-Programmierung in verschiedenen TIA Portal-Projekten umsetzen und zusammenführen.

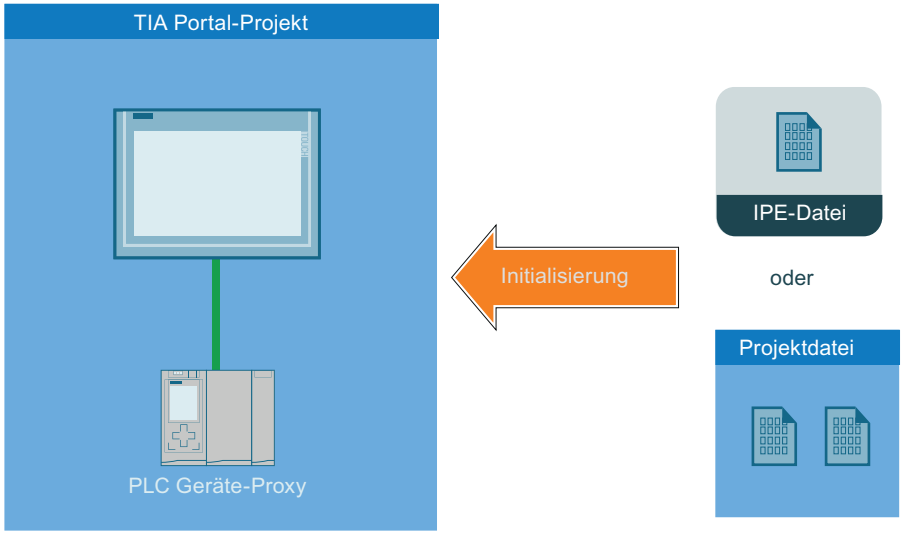

Nähere Hinweise und Vorgehensweisen zur Projektierung finden Sie unter: Grundlagen zu Inter Project Engineering (IPE)

14.12 Projektstatus am SIMATIC WinCC RT Symbol erkennen

Einleitung

In der Taskleiste im Benachrichtigungsbereich zeigt WinCC das Symbol "SIMATIC WinCC RT" an. Über dieses Symbol erhalten Sie Informationen über den Status der Runtime.

Projektstatus

Die folgende Tabelle zeigt, welcher Projektstatus welchem Symbol entspricht:

Symbol	Status
	WinCC Runtime ist gestartet.
	WinCC Runtime ist nicht aktiv. Kein Runtime-Projekt ist geöffnet.
	WinCC Runtime ist aktiv.
	WinCC Runtime ist aktiv und der Server hat den Status "Fault".
	WinCC wechselt den Status.
	WinCC öffnet ein Runtime-Projekt.
	WinCC aktiviert ein Runtime-Projekt.
	WinCC deaktiviert ein Runtime-Projekt.
	WinCC schließt ein Runtime-Projekt.

Bedienmöglichkeiten über das Kontextmenü

Sie können ein geöffnetes Runtime-Projekt in Abhängigkeit seines Status bedienen. Über das Kontextmenü des Symbols "SIMATIC WinCC RT" haben Sie folgende Bedienmöglichkeiten:

- Runtime aktivieren/Runtime deaktivieren
- Graphics Runtime beenden/Graphics Runtime starten
- Diagnosefenster öffnen
- Runtime Start Optionen
- WinCC Lizenz Analyse

Wenn Sie auf das Symbols "SIMATIC WinCC RT" klicken, sehen Sie den aktuellen Projektstatus und bei einem redundanten Projekt auch den Status des Partnerservers oder Clients.

Übersetzen und Laden 15

15.1 Herstellen der Verbindung zum Bediengerät

Einleitung

Um ein WinCC Projekt auf ein Bediengerät zu laden, muss die Verbindung zwischen dem Projektierungs-PC und dem Bediengerät aufgebaut und richtig konfiguriert sein. Wenn die Verbindung während des Ladens nicht zu Stande kommt, wird das Laden abgebrochen.

Verbindung zwischen Projektierungs-PC und Bediengerät herstellen

1. Kontrollieren Sie die Kabelverbindung zwischen Bediengerät und Projektierungs-PC.
2. Öffnen Sie in WinCC den Editor "Geräte & Netze" und aktivieren Sie die Netzsicht.
3. Selektieren Sie in der Netzsicht das Subnetz und prüfen Sie die Einstellungen für das Subnetz.
4. Selektieren Sie in der Netzsicht oder in der Gerätesicht die Schnittstelle des Bediengeräts und prüfen Sie im Inspektorfenster die Verbindungsparameter.
5. Schalten Sie das Bediengerät ein und tippen Sie im Loader auf die Schaltfläche "Control Panel".
   Das Control Panel wird geöffnet.
6. Tippen Sie im Control Panel zwei Mal auf Transfer.
   Der Dialog "Transfer Settings" wird geöffnet.
7. Kontrollieren Sie die Einstellungen und tippen Sie anschließend auf "Advanced".
   Der Dialog "[Protokoll*] Settings" wird geöffnet.
   *: Der Titel des Dialoges ist abhängig vom verwendeten Protokoll, z. B. "PROFIBUS Settings".
8. Kontrollieren Sie die erweiterten Einstellungen und schließen Sie den Dialog mit "OK".

Wichtige Einstellungen

Wenn Sie die Verbindungseinstellungen kontrollieren, prüfen Sie insbesondere folgende Parameter:

- Netzwerkadressen und Stationsadressen
- Eingestellte Übertragungsgeschwindigkeit
- Master am Bus; generell ist nur ein Master zulässig.

Hinweis

Verbindungen über Ethernet

Beachten Sie bei Ethernet-basierter Kommunikation die erforderlichen Netzwerkeinstellungen. Weitere Hinweise dazu finden Sie unter "Liesmich > WinCC > Security-Hinweise".

Wenn Sie für die Verbindung einen einstellbaren Adapter verwenden, prüfen Sie am Adapter ebenfalls die Einstellungen, z. B. Übertragungsgeschwindigkeit, Master am Bus.

Siehe auch

Projekt laden (Seite 2273)

15.2 Basic Panels

15.2.1 Einstellungen für Runtime

15.2.1.1 Einstellungen für Runtime

Einleitung

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen".

Hinweis

Geräteabhängigkeit

Die Auswahl an Optionen in den Einstellungen für Runtime ist abhängig vom Bediengerät.

Übersicht

In den Runtime-Einstellungen projektieren Sie Optionen aus verschiedenen Bereichen Ihres Projekts:

- Darstellung in Runtime
  - Startbild
  - Standardvorlage
  - Standardstil des Projekts und Stil des Bediengeräts
  - Bildschirmauflösung
  - Vollbildmodus
  - Grafiksammlung
  - Bitselektion für Farbe und Blinken
  - Bitselektion für Text- und Grafiklisten
- Projektkennung zur Kompatibilitätsprüfung mit der Steuerung
- Archivierungssprache
- Fernbedienung über Sm@rt Server im Bereich "Dienste"
  Weitere Hinweise dazu finden Sie in der WinCC Online-Hilfe unter "Optionen > Sm@rt Optionen"

- Bedienung in Runtime
  - Taskumschaltung
  - Zusätzliche Daten zum Debuggen von Skripten laden: Objektnamen und Kommentare in Skripten
  - Grenzwerte als Tooltipp anzeigen
  - Bildschirmtastatur
  - Funktionstasten deaktivieren
  - Loslassen von Schaltflächen beim Verlassen der Schaltfläche
  - Tastaturbelegung und Tastenkombinationen
- GMP-konforme Projektierung
  Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Optionen > WinCC Audit".
- Meldungen
  - Pufferüberlauf
  - Text der Quittiergruppe
  - Protokoll
  - Farben der Meldeklassen
  - Konfiguration der Systemmeldungen
  - Persistenter Meldepuffer
- Benutzerverwaltung
  - Konfiguration Kennwort und Abmeldezeit
  - Grenze für Login-Versuche
  - Hierarchieebenen
  - Konfiguration SIMATIC Logon
- Sprache und Schriftart
  - Verfügbare Runtime-Sprachen
  - Reihenfolge der Sprachumschaltung
  - Standardschrift
  - Schriftart mit gültiger Lizenz herunterladen
- Konfiguration des OPC UnifiedArchitecture Server
- Einstellungen zum Synchronisieren von PLC- und HMI-Variablen

Siehe auch

Darstellung in Runtime projektieren (Seite 2203)

Bedienung in Runtime projektieren (Seite 2204)

Kompatibilitätsprüfung mit der Steuerung einstellen (Seite 2206)

Systemmeldungen bearbeiten (Seite 836)

Meldeklassen anlegen (Seite 818)

Meldepufferüberlauf projektieren (Seite 927)

Einstellungen zum Synchronisieren (Seite 564)

Einstellungen für die Benutzerverwaltung (Seite 1374)

Sprachen verwalten (Seite 2095)

Arbeiten mit Archiven (Seite 1071)

15.2.1.2 Darstellung in Runtime projektieren

Einleitung

In den Runtime-Einstellungen legen Sie fest, wie Bilder, Meldungen und Sprache dargestellt werden.

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen". Die Auswahl an Optionen ist abhängig vom Bediengerät.

Anzeige am Zielsystem

In WinCC stellen Sie ein, wie das generierte Projekt in Runtime dargestellt wird:

- Wählen Sie, ob das Projekt als Vollbild oder in einem Fenster angezeigt wird. Das Fenster ist kleiner als der Bildschirm. Im Vollbildmodus beansprucht das Projekt den gesamten Bildschirm. Ein Fenster und Bedienelemente für das Fenster sind nicht mehr vorhanden. Um die Darstellung Vollbild beim Start zu verwenden, klicken Sie im Editor "Einstellungen" auf "Allgemein". Aktivieren Sie im Bereich "Bild" das Kontrollkästchen "Vollbildmodus".

  Hinweis

  Darstellung nicht vollständig

  Wenn der Bildschirm des Bediengeräts nicht die projektierte Größe (in Pixeln) hat, wird das Projekt im Vollbildmodus nur auf einem Teil des Bildschirms angezeigt.

- Unter "Allgemein > Bild" legen Sie fest, ob das Bediengerät den Standardstil des Projekts oder einen anderen vordefinierten Stil verwendet.

Einstellungen zu Bildern

In den Runtime-Einstellungen legen Sie Startbild und Standard-Vorlage fest. Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Arbeiten mit Bildern".

Einstellungen zur Sprache

- Archivierungssprache
  Unter "Allgemein > Archive" legen Sie fest, in welcher Sprache die Archive geschrieben werden. Die gewählte Archivsprache bleibt immer gleich, auch wenn der Bediener in Runtime die Sprache umschaltet.
  Wenn Sie "Anlaufsprache" wählen, bleibt die Archivsprache bis zum Runtime-Stopp gleich, auch wenn der Bediener die Sprache umgestellt hat. Beim nächsten Runtime-Start jedoch wird die zuletzt eingestellte Runtime-Sprache als Archivsprache verwendet.
- Runtime-Sprache
  Unter "Sprache & Schriftart" legen Sie alle Runtime-Sprachen und Ihre Darstellung fest. Wenn Sie die Schriftgröße bzw. Schriftstil ungünstig wählen, kann es dazu führen, dass Texte abgeschnitten dargestellt werden. Dabei sind z. B. folgende Texte betroffen:
  - Tooltips
  - langer Meldetext
  - Texte in Dialogen

Einstellung für bestimmte Bildobjekte

- Bitselektion festlegen
  Unter "Bilder > Bitselektion" legen Sie fest, ob die Bitauswahl auf diesem Bediengerät für Folgendes verwendet wird:
  - Text und Grafik (Text- und Grafiklisten)
  - Farbe und Blinken (Gestaltungsanalyse)
    Wenn Sie die jeweilige Option aktivieren, wird jeweils die Auswahl angezeigt, die beim gesetzten Bit mit der kleinsten Wertigkeit projektiert ist:
    Wenn Sie die jeweilige Option deaktivieren und mehrere Bits gesetzt sind, wird die Auswahl angezeigt, die nur beim gesetzten Bit projektiert ist.
- Tooltipps bei E/A-Feldern
  Ob projektierte Grenzwerte bei der Eingabe in E/A-Felder in Runtime als Tooltip angezeigt werden, legen Sie unter "Bilder > Anzeige" fest.
- Farbige Objekte
  Unter "Allgemein > Bilder > Farbtiefe" legen Sie die Farbtiefe Ihres Bediengeräts fest. Bildobjekte gestalten Sie bei Bedarf farbig. Die Anzahl der möglichen Farben hängt davon ab, wie viele Farben das eingestellte Bediengerät unterstützt.

15.2.1.3 Bedienung in Runtime projektieren

Einleitung

In den Runtime-Einstellungen legen Sie fest, wie Bilder bedient werden und Sie in Runtime mit Tasten bedienen.

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen". Die Auswahl an Optionen ist abhängig vom Bediengerät.

Taskumschaltung sperren

Abhängig vom Bediengerät, haben Sie die Möglichkeit, die Taskumschaltung am Bediengerät zu sprerren. Damit verhindern Sie, dass der Bediener in Runtime andere Anwendungen aufruft.

Aktivieren Sie in den Runtime-Einstellungen des Bediengeräts unter "Allgemein > Bild" die Option "Taskumschaltung sperrens".

Hinweis

Runtime beenden

Wenn Sie die Programmumschaltung sperren, projektieren Sie in Ihrem Projekt unbedingt die Systemfunktion "StoppeRuntime" an ein Objekt, z. B. an eine Funktionstaste oder eine Schaltfläche.

Einstellungen zum Datentransfer auf das Bediengerät

- Namen der Bildobjekte in Skripten transferieren
  Wenn Sie unter "Allgemein > Bild" die Option "Namen laden" aktivieren, werden anstelle von codierten Adressierungsinformationen die Objektnamen der Bildobjekte transferiert. Wenn Sie die Bildobjekte in einem Skript über den Objektnamen ansprechen wollen, benötigen Sie Objektnamen . Wenn Sie ein Skript im Debugger testen, wird der Code durch die Darstellung der Objektnamen übersichtlicher.
- Kommentare in Skripten anzeigen
  Unter "Bilder > Anzeige" legen Sie fest, ob die Kommentare in Skripten mit auf das Bediengerät transferiert werden.
  Wenn Sie die Kommentare nicht transferieren, sparen Sie Speicherplatz auf Ihrem Bediengerät.
  Wenn Sie die Kommentare transferieren, ist das Skript beim Testen im Debugger verständlicher.

Mit Funktionstasten bedienen

Um bei Tastengeräten die Funktion einer Funktionstaste zu symbolisieren, projektieren Sie im Display neben der Funktionstaste eine Grafik. Unter "Bilder > Funktionstasten > Benutzerdefinierte Piktogramm-Größe" legen Sie fest, ob Sie eine zum Standard abweichende Grafikgröße verwenden wollen.

Mit Tastatur bedienen

Unter "Tastatur > Allgemein > Bildschirmtastatur verwenden" legen Sie fest, ob auf Ihrem Bediengerät die Bildschirmtastatur zu Verfügung steht.

Unter "Tastatur > Allgemein > Schaltfläche beim Verlassen loslassen" legen Sie fest, ob das Ereignis "Loslassen" getriggert wird, wenn der Benutzer eine Schaltfläche verlässt, ohne sie loszulassen.

Für Tastenbediengeräte legen Sie unter "Tastatur > Deaktiviere Funktionstasten in Dialogen" fest, ob die Funktionstasten für die Dauer des angezeigten Dialogs deaktiviert werden. Große Dialoge können an Funktionstasten angrenzende oder Funktionstasten beschreibende Bereiche oder Grafiken verdecken. Funktionstasten könnten so versehentlich bedient werden.

15.2.1.4 Kompatibilitätsprüfung mit der Steuerung einstellen

Einleitung

Über den Bereichszeiger "Projektkennung" projektieren Sie die Projektkennung zur Überprüfung der Konsistenz des Projektes mit der Steuerung. Der Bereichszeicher liest die Projektversion der Steuerung aus. Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Mit Steuerungen kommunizieren".

Kompatibilitätsprüfung mit der Steuerung einstellen

Sie wählen die Projektkennung in den Runtime-Einstellungen des Bediengeräts unter "Allgemein > Identifizierung > Projektkennung".

Runtime überprüft beim Start, ob die Projektkennung des Projekts mit der Projektversion der Steuerung übereinstimmt. Nur wenn beide Werte übereinstimmen wird Runtime gestartet.

Bei Verbindungen ist ein Bereichszeiger "Projektkennung" angelegt, dessen Adresse Sie in der Steuerung festlegen. Der Wert, den Sie an dieser Adresse hinterlegen, ist dann die Projektkennung.

Die Projektkennung ist nicht auf allen Bediengeräten verfügbar.

15.2.1.5 Kennwortschutz in Runtime

Kennwortschutz in Runtime: Grundlagen

Übersicht

Der Verbindungsaufbau kann im TIA Portal durch Kennwörter geschützt werden. Die konfigurierten Kennwörter werden verschlüsselt gespeichert. Beim Transfer ins Zielsystem werden die Kennwörter ebenfalls gegen nicht autorisierte Zugriffe geschützt und können nicht gelesen werden. Auf den Bediengeräten in Runtime werden diese Kennwörter sicher entschlüsselt.

Der Kennwortschutz kann für folgende Verbindungen angelegt werden:

- SIMATIC S7-1200
- SIMATIC S7-1500
- SIMATIC S7-1500 Software Controller (WinAC)
- PC Station 2.0 (Stationmanager)

Hinweis

Beim Wechsel zu einer älteren Version werden die Kennwörter ungültig.

Zertifikat herunterladen

Das Zertifikat zum Entschlüsseln der Kennwörter wird im Zertifikatspeicher des Bediengeräts gespeichert. Eine Systemmeldung informiert Sie darüber, falls das Zertifikat nicht vorhanden ist.

Das Zertifikat stellen Sie wieder her, indem Sie die Konfiguration erneut herunterladen.

Siehe auch

Kennwort für den Transport konfigurieren (Seite 2207)

Kennwort für den Transport zurücksetzen (Seite 2208)

Kennwort für den Transport konfigurieren

Einleitung

Um den privaten Kennwortschlüssel beim Transport zu schützen, können Sie ein Kennwort für den Transport unter "Runtime-Einstellungen" konfigurieren.

Kennwort anlegen

1. Öffnen Sie im Projektfenster den Editor "Runtime-Einstellungen > Allgemein".
2. Tragen Sie im Bereich "Transportkennwort" im Textfeld "Kennwort eingeben" das Kennwort ein.
3. Bestätigen Sie das Kennwort im Feld "Kennwort bestätigen".

Hinweis

Wenn kein altes Kennwort existiert, ist das Feld "Altes Kennwort" deaktiviert.

Ergebnis

Sie haben das Kennwort für den Transport konfiguriert. Wenn ein Kennwort für den Transport konfiguriert ist und Sie das Projekt auf das Bediengerät übertragen, erscheint in Runtime einmalig ein Dialog mit der Kennwortabfrage.

Wenn das eingegebene Kennwort korrekt ist, wird das Zertifikat im Zertifikatspeicher des Bediengeräts gespeichert.

Wenn Sie das Kennwort ändern, laden Sie das Projekt mit dem geänderten Kennwort auf das Bediengerät und starten Runtime. Es erscheint keine Kennwortabfrage.

Siehe auch

Kennwortschutz in Runtime: Grundlagen (Seite 2206)

Kennwort für den Transport zurücksetzen (Seite 2208)

Kennwort für den Transport zurücksetzen

Einleitung

Wenn Sie beispielsweise das Transportkennwort verlegt haben, können Sie ein neues Sicherheitszertifikat erstellen. Dabei setzen Sie das Transportkennwort zurück.

Kennwort zurücksetzen

1. Klicken Sie auf die Schaltfläche "Kennwort zurücksetzen".
   Das Kennwort wird zurückgesetzt. Ein neues Zertifikat ohne Kennwort wird generiert.

   Hinweis

   Andere Kennwörter wie z. B. das Kennwort von der PLC, werden ungültig. Beim nächsten Übersetzen informiert eine Fehlermeldung Sie darüber, dass diese Kennwörter nicht mehr gültig sind.

2. Geben Sie das neue Kennwort ein und bestätigen Sie es im Feld "Kennwort bestätigen".

   Hinweis

   Solange Sie kein neues Kennwort für den Transport eingeben, ist das Zertifikat nicht kennwortgeschützt.

   Nachdem Sie das Projekt auf das Bediengerät geladen haben, erscheint die Kennwortabfrage.

Siehe auch

Kennwortschutz in Runtime: Grundlagen (Seite 2206)

Kennwort für den Transport konfigurieren (Seite 2207)

15.2.2 Übersicht zum Übersetzen und Laden von Projekten

Übersicht

Bereits während Sie ein Projekt in WinCC projektieren, wird das Projekt laufend im Hintergrund übersetzt. Dadurch werden die Zeiten für das endgültige Übersetzen verkürzt. Wenn Sie das Übersetzen starten, entsteht eine Datei, die am jeweiligen Bediengerät ablauffähig ist.

Wenn beim Übersetzen Fehler auftreten, unterstützt WinCC Sie dabei, die Fehler zu finden und zu beheben.

Nach der Fehlerbehebung laden Sie das übersetzte Projekt auf die Bediengeräte, auf denen Sie das Projekt einsetzen wollen. Wenn der Projektierungs-PC nicht mit dem Bediengerät verbunden ist, speichern Sie das übersetzte Projekt auf einem Datenträger ihrer Wahl. Das übersetzte Projekt übertragen Sie dann von einem mit dem Bediengerät verbundenen PC auf das Bediengerät.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten S7-Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

Projekt

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Das folgende Bild zeigt den Zusammenhang zwischen WinCC-Projekt und Runtime-Projekt am Beispiel des Vorgangs von "Übersetzen und Laden":

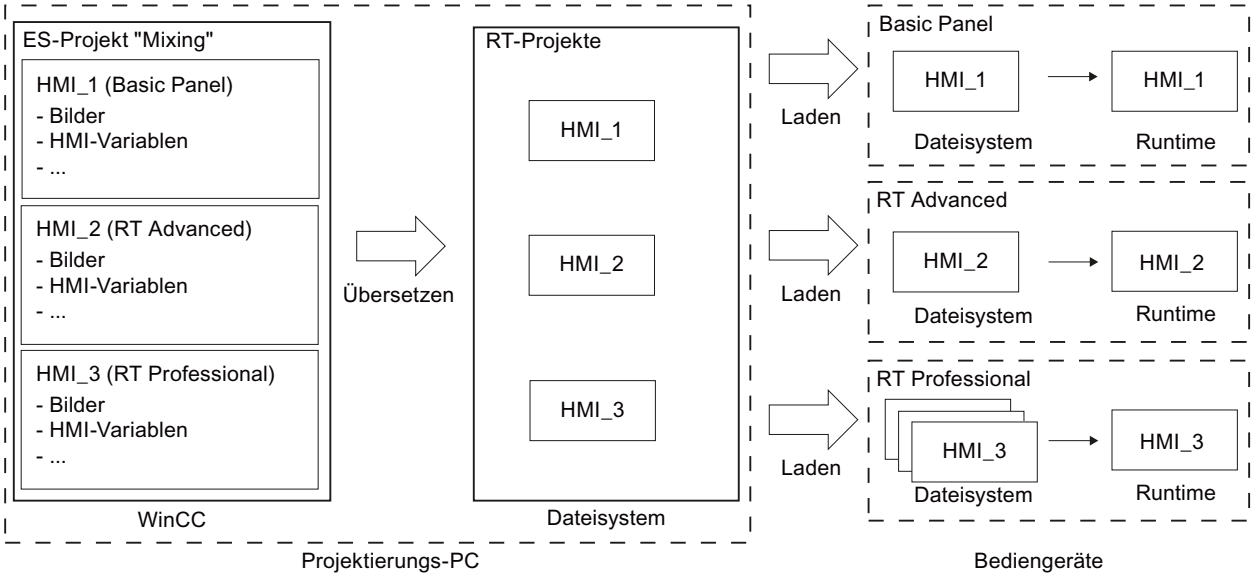

Runtime

Runtime ist die Software zur Prozessvisualisierung. In Runtime führen Sie das Projekt im Prozessbetrieb aus.

Es wird zwischen zwei Arten von Runtime unterschieden:

1. Runtime auf einem Panel
   Bevor Sie ein Runtime-Projekt auf einem Panel ausführen, müssen Sie das Runtime-Projekt vor dem Start auf das Panel übertragen.
2. Runtime auf einem PC
   Wenn Runtime auf dem Projektierungs-PC installiert ist, dann können Sie das Runtime-Projekt direkt auf dem Projektierungs-PC ausführen.
   Wenn Sie das Runtime-Projekt auf einem anderen PC ausführen wollen, müssen Sie das Runtime-Projekt vor dem Start auf den PC übertragen.

Runtime-Version

Die Runtime-Version ist abhängig vom Image des projektierten Bediengeräts. Die Runtime-Version des übersetzten Projekts wird im Inspektorfenster unter "Info" angezeigt.

Simulation

Mit einer Simulation testen Sie ihre Projektierung. Eine Simulation können Sie ohne eine Anbindung an den laufenden Prozess starten.

In einer Simulation testen Sie beispielsweise projektierte Variablen oder Bildwechsel. Während der Simulation können die projektierten Variablen mithilfe des Variablensimulators manipuliert, aktiviert und deaktiviert werden.

Es wird zwischen zwei Simulationsarten unterschieden:

1. Simulation eines Panels
   Wenn Sie ein Panel in ihrem Projekt angelegt haben, wird das Panel in der Simulation angezeigt. Mithilfe dieser Simulationsart können Sie ihre Projektierung am Bediengerät testen, ohne das Projekt auf das Panel zu übertragen.
2. Simulation einer Runtime
   Bei der Simulation einer Runtime testen Sie das Projekt direkt am Projektierungs-PC.

15.2.3 Projekt übersetzen

Einleitung

Bereits während Sie ein Projekt in WinCC projektieren, werden die Änderungen am Projekt laufend im Hintergrund übersetzt. Wenn Sie ein Projekt laden, wird das Projekt automatisch übersetzt. Auf diese Weise wird sichergestellt, dass immer der neueste Stand des Projekts geladen wird.

Während des Übersetzens überprüft WinCC das Projekt auf Konsistenz. Im Inspektorfenster werden alle fehlerhaften Stellen im Projekt aufgelistet. Über den Eintrag im Inspektorfenster springen Sie direkt zur Fehlerursache. Sie überprüfen die festgestellten Fehler und beheben diese.

Umfang des Übersetzens

Sobald Sie mit der Projektierung eines Bediengeräts beginnen, werden die Projektierungsdaten laufend im Hintergrund übersetzt. Wenn Sie ein Projekt manuell übersetzen, werden nur die Änderungen in der Projektierung seit dem letzten Übersetzen im Hintergrund übersetzt.

Um z. B. die Konsistenz der projektierten Daten zu testen, können Sie ein komplettes Übersetzen des Projekts jederzeit manuell starten.

Voraussetzung

- Ein Projekt ist geöffnet.

Vorgehen

Um ein Projekt zu übersetzen, gehen Sie folgendermaßen vor:

1. Wenn Sie mehrere Bediengeräte gleichzeitig übersetzen wollen, wählen Sie alle gewünschten Bediengeräte mit der Mehrfachauswahl in der Projektnavigation aus.
2. Übersetzen Sie das Projekt:
   - Um nur Änderungen im Projekt zu übersetzen, wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software (nur Änderungen)".
   - Um das Projekt komplett zu übersetzen, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (komplett übersetzen)".

Ergebnis

Die Projektierungsdaten aller ausgewählten Bediengeräte werden übersetzt. Wenn beim Übersetzen Fehler auftreten, werden diese im Inspektorfenster angezeigt.

15.2.4 Simulieren von Projekten

15.2.4.1 Grundlagen zur Simulation

Einleitung

Mit dem Simulator testen Sie das Verhalten Ihrer Projektierung am Projektierungs-PC. Auf diese Weise finden Sie frühzeitig und vor dem Produktivbetrieb logische Projektierungsfehler.

Sie können den Simulator auf folgende Arten starten:

- Im Kontextmenü des Bediengeräts oder eines Bildes: "Simulation starten"
- Menübefehl "Online > Simulation > [Starten|Mit Variablen-Simulator|Mit Skript-Debugger]"
- In der Portalansicht unter "Visualisierung > Gerät simulieren"

Voraussetzung

Auf dem Projektierungs-PC ist die Komponente Simulation/Runtime installiert.

Einsatzgebiet

Sie können mit dem Simulator z. B. folgende Funktionen des Bedien- und Beobachtungssystems testen:

- Überprüfung von Grenzwertstufen und Meldeausgaben
- Durchgängigkeit von Alarmen
- Projektierte Alarmsimulationen
- Projektierte Warnungen
- Projektierte Fehlermeldungen
- Kontrolle von Statusanzeigen
- Verschaltung und Layout von Bildern

Siehe auch

Projekt simulieren (Seite 2212)

15.2.4.2 Projekt simulieren

Einleitung

Sie simulieren Ihr Projekt auf eine der folgenden zwei Arten:

- Ohne angeschlossene Steuerung
  Die Werte von Bereichszeigern und Variablen verändern Sie in einem Variablen-Simulator, der bei der Simulation von WinCC Runtime gelesen wird.
- Mit angeschlossener Steuerung ohne laufenden Prozess
  Sie simulieren Ihr Projekt, indem Sie es direkt in Runtime laufen lassen. Die Variablen und Bereichzeiger werden aktiv. So erreichen Sie in Runtime eine authentische Simulation Ihres projektierten Bediengeräts.

Hinweis

Einschränkungen bei der Simulation

Folgende Systemfunktionen können nicht simuliert werden:

- KalibriereTouchscreen

Eine Simulation des Media Players ist nicht möglich. Im Simulationsfenster wird anstelle des Media Players ein statisches Bild angezeigt.

Bei Bediengeräten mit Windows CE sind Dateizugriffe über Skripte nicht möglich.

Voraussetzung

- Simulation ohne angeschlossene Steuerung: Variablen sind erstellt
- Simulation mit angeschlossener Steuerung ohne laufenden Prozess: Ein Projekt mit Variablen und Bereichszeigern ist erstellt

Vorgehen

Um ein Projekt über den Variablen-Simulator zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie am Projektierungs-PC das Projekt.
2. Wählen Sie den Menübefehl "Online > Simulation > Mit Variablen-Simulator".
   Wenn Sie das Projekt zum 1. Mal simulieren, wird der Simulator mit einer neuen, leeren Tabelle gestartet. Gleichzeitig wird das Projekt in Runtime geöffnet.
   Zwischen Variablen-Simulator und Runtime schalten Sie mit <Alt+Tab> um.
3. Um einen Prozesswert zu simulieren, wählen Sie im Variablen-Simulator die gewünschte "Variable" aus.
   Die Auswahl enthält alle projektierten Variablen. Sie können maximal 300 Variablen gleichzeitig simulieren.
4. Wählen Sie in der Spalte "Simulation" die gewünschte Simulationsart aus.
5. Verändern Sie in den entsprechenden Spalten die Werte für die Variablen und die Bereichszeiger.
6. Um die Simulation für diese Variable zu starten, aktivieren Sie das Kontrollkästchen "Start".
7. Um die Simulation zu speichern, wählen Sie den Menübefehl "Datei > Speichern" und geben sie einen sprechenden Namen ein, z. B. "Mixing".
   Die Datei erhält die Endung "*.cors".

Ergebnis

Die Prozesswerte werden in Runtime simuliert. Abhängig von der Simulationsart werden die Variablenwerte z. B. zufällig erzeugt oder inkrementiert.

Um Variablenwerte vorzugeben, ändern Sie die Simulationsart auf "<Anzeige>" und geben Sie unter "Wert setzen" den gewünschten Wert ein.

Das folgende Bild zeigt einen Variablen-Simulator mit vier Variablen, deren Werte im Wertebereich von 10 bis 1000 zufällig ermittelt werden:

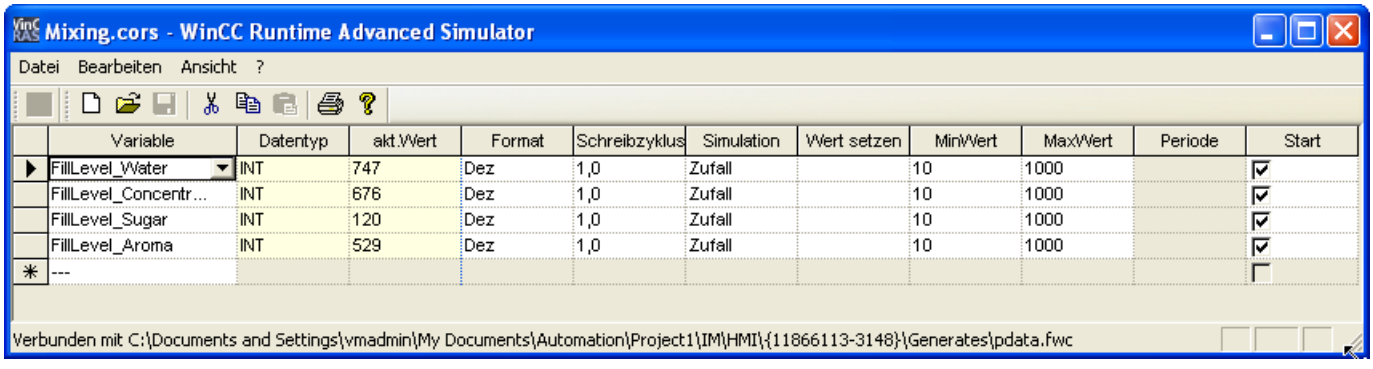

Simulationsdaten verwalten

Wenn Sie die Daten einer früheren Simulation gespeichert haben, können Sie die Datei zu einem späteren Zeitpunkt öffnen und Ihr Projekt erneut simulieren. Voraussetzung dafür ist, dass die im Variablen-Simulator enthaltenen Variablen und Bereichszeiger noch im Projekt enthalten sind.

Um eine Simulationsdatei zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator".
2. Wählen Sie im Variablen-Simulator den Menübefehl "Datei > Öffnen".
3. Wählen Sie die entsprechende Simulationsdatei aus und klicken Sie auf "Öffnen".
   Der Simulator übernimmt die gespeicherten Daten.

Variablen aktivieren und deaktivieren

Damit der Übergang von der Offline-Projektierung zur Online-Projektierung reibungslos möglich ist, starten und stoppen Sie die Simulation jeder Variable einzeln. Aktivieren Sie dazu in der entsprechenden Zeile "Start".

Wenn eine Variable aktiviert ist, werden die Simulationswerte berechnet und an den WinCC-Simulator übergeben.

Variable löschen

Um eine Variable aus dem Variablen-Simulator zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zelle mit dem Variablennamen.
2. Wählen Sie den Menübefehl "Bearbeiten > Ausschneiden".
   Die Variable wird aus der Tabelle entfernt.

Siehe auch

Grundlagen zur Simulation (Seite 2211)

Arbeiten mit dem Variablen-Simulator (Seite 2214)

15.2.4.3 Arbeiten mit dem Variablen-Simulator

Erläuterungen zum Variablen-Simulator

Der Variablen-Simulator besteht aus folgenden Spalten:

Spalte	Beschreibung
Variable	Legt die Variablen für die Simulation fest.
Datentyp	Zeigt den Datentyp der ausgewählten Variable.
akt. Wert	Zeigt den simulierten Wert der eingetragenen Variablen.

Spalte	Beschreibung
Format	Legt das gewählte Format fest, in welchem die Variablenwerte simuliert werden: • Dezimal (1, 2, 3, 4, ...) • Hexadezimal (03CE, 01F3, ...) • Binär (0 und 1)
Schreibzyklus	Legt den gewählten Zeitabstand fest, in dem die aktuellen Variablenwerte simuliert werden. Wenn Sie z. B. den Wert "2" eintragen, wird alle 2 Sekunden der aktuelle Variablenwert angezeigt.
Simulation	Zeigt, auf welche Art die Variablenwerte während der Simulation verarbeitet werden.
Wert setzen	Legt den gewählten Wert für die entsprechende Variable fest. Die Simulation startet bei dem eingesetzten Wert.
minWert maxWert	Legt den Wertebereich der Variable fest. Sie legen dazu einen Minimal- und einen Maximalwert an. Standardmäßig liegen die Voreinstellungen bei -32768 für den Minimalwert und 32767 für den Maximalwert.
Periode	Enthält die Periode, in der sich der Variablenwert bei den Simulationsarten "Inkrement" und "Dekrement" wiederholt.
Start	Startet die Simulation der Variable nach den vorangegangenen Eingaben.

Simulationsarten

Der Simulator stellt sechs verschiedene Simulationsarten zur Verfügung. Während der Simulation werden die projektierten Variablen mit realitätsnahen Werten versorgt.

Simulationsart	Beschreibung
Sinus	Verändert den Variablenwert in Form einer Sinuskurve. Die Darstellung erfolgt als periodische, nicht lineare Funktion.
Zufall	Stellt zufällig erzeugte Werte zur Verfügung. Der Variablenwert wird über eine Zufallsfunktion verändert.
Inkrement	Erhöht den Variablenwert kontinuierlich bis zu einem maximal angegebenen Wert. Beginnt nach Erreichen des Maximums wieder beim Minimum. Der Verlauf der Werte entspricht einer positiven Sägezahnkurve.
Dekrement	Verringert den Variablenwert kontinuierlich bis zu einem minimal angegebenen Wert. Beginnt nach Erreichen des Minimums wieder beim Maximum. Der Verlauf der Werte entspricht einer negativen Sägezahnkurve.
Bit schieben	Schiebt ein gesetztes Bit kontinuierlich um eine Position weiter. Die vorherige Position wird jeweils zurückgesetzt. Damit können Sie z. B. die Meldungen für ein Bediengerät testen.
<Anzeige>	Der aktuelle Variablenwert wird statisch angezeigt.

Beispiel: Variablen simulieren mit der Simulationsart "Bit schieben"

Um die Variablen mit der Simulationsart "Bit schieben" zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das zu simulierende Projekt.
2. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator". Der Variablen-Simulator wird geöffnet.
3. Wählen Sie in der Spalte "Variable" eine Variable aus Ihrem Projekt aus.

4. Wählen Sie in der Spalte "Format" "Bin" aus.
5. Geben Sie in der Spalte "Schreibzyklus" den Wert "1" ein.
6. Wählen Sie in der Spalte "Simulation" die Simulationsart "Bit schieben" aus.
7. Geben Sie in der Spalte "Wert setzen" den Wert "1" ein.
8. Aktivieren Sie die Variable mit dem Kontrollkästchen "Start".

Ergebnis

Der Simulator testet die ausgewählte Variable bitweise folgendermaßen:

Simulationswerte	Byte für Meldungen
Gesetzter Anfangswert	00000001
1. Simulationswert	00000010
2. Simulationswert	00000100
3. Simulationswert	00001000
....	...

Sie sehen in Runtime, ob bei dem jeweiligen Wert z. B. die gewünschte Meldung ausgegeben wird.

15.2.4.4 Einschränkungen bei der Simulation

Meldungen mit dynamischen Parametern

Wenn Sie Variablen oder Textlisten als externe Variablen für Meldungen verwenden, dann werden die dynamischen Parameter der Meldungen nicht angezeigt.

Für die Simulation von Meldungen im Variablen-Simulator sind nur interne Variablen freigegeben.

Verwenden Sie PLCSim, um dynamische Parameter zu simulieren.

Comfort Panels

Wenn Sie PLCSim mit der Version V14 oder höher für die Simulation mit einem Comfort Panel verwenden möchten, muss die Geräteversion des Comfort Panels 14.0.0.0 oder höher sein.

15.2.5 Laden von Projekten

15.2.5.1 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum 1. Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Wenn das Image des Ziel-Bediengeräts nicht dem Image aus der Projektierung entspricht, wird abgefragt, ob Sie das Image jetzt ändern wollen.

ACHTUNG
Das Ändern des Images löscht alle Daten auf dem Bediengerät. Wenn Sie das Image ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb bei Bedarf zuerst folgende Daten: • Benutzerverwaltung • Rezepturen

   - Voreinstellungen, die beim Laden wirksam werden. Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional)
     Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional)
     Sie können das Projekt nicht laden.
     Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- Ethernet

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf ein externes Speichermedium, z. B. einen USB-Stick, und laden es dann auf das Bediengerät.

Diese Funktion ist in Verbindung mit einem Bediengeräte-Image verfügbar, das mit der TIA Portal Version V14 oder höher kompatibel ist.

Sie erzeugen die notwendigen Projektdaten in WinCC, indem Sie das Bediengerät projektieren und anschließend den Ordner des Bediengeräts (z. B. "HMI_1 [<*Gerätetyp*>]") per Drag and Drop auf ein externes Speichermedium unter "Card Reader/USB-Speicher" ziehen.

Übertragen von Runtime-Addons

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

Siehe auch

Betriebssystem des Bediengeräts aktualisieren (Seite 2434)

15.2.5.2 Projekt laden

Einleitung

Damit ein Projekt auf einem Bediengerät abläuft, laden Sie das Projekt auf das Bediengerät. Beim Laden legen Sie insbesondere fest, ob auf dem Bediengerät vorhandene Daten wie "Benutzerverwaltung" oder "Rezepturdaten" überschrieben werden.

Wenn das Bediengerät PROFINET unterstützt, wird der in der Projektnavigation eingetragene Name des Bediengeräts als Gerätename für die PROFINET-Kommunikation verwendet. Der Name wird beim Laden in das Bediengerät geschrieben. Wenn im Bediengerät bereits ein Gerätename für die PROFINET-Kommunikation eingetragen war, wird dieser überschrieben.

Generell kann auf einem Bediengerät immer nur ein Projekt in Runtime sein. Standardmäßig ist ein Bediengerät so konfiguriert, dass eine laufende Runtime beim Laden automatisch beendet wird. Ansonsten müssen Sie Runtime am Bediengerät manuell beenden.

Wenn das Image des Ziel-Bediengeräts nicht dem Image aus der Projektierung entspricht, wird abgefragt, ob Sie das Image jetzt ändern wollen.

ACHTUNG
Das Ändern des Images löscht alle Daten auf dem Bediengerät. Wenn Sie das Image ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb bei Bedarf zuerst folgende Daten: • Benutzerverwaltung • Rezepturen

Transferverhalten am Bediengerät steuern

Generell kann auf einem Bediengerät immer nur ein Projekt in Runtime sein. Standardmäßig ist ein Bediengerät so konfiguriert, dass eine laufende Runtime beim Laden automatisch beendet wird. Sonst müssen Sie Runtime am Bediengerät manuell beenden.

Am Bediengerät definieren Sie im "Start Center" unter "Einstellungen", wie sich das Bediengerät beim Laden eines Projekts verhält:

Transfermodus	Auswirkung
Aus	Das Laden eines Projekts auf das Bediengerät ist nicht möglich.
Manuell	Das Laden eines Projekts auf das Bediengerät ist nur möglich, wenn die folgenden Voraussetzungen erfüllt sind: • Runtime läuft nicht • Das Bediengerät ist in der Betriebsart "Transfer".
Automatisch	Das Laden eines Projekts auf das Bediengerät ist immer möglich. Wenn am Projektierungs-PC ein Transfer gestartet wird und am Bediengerät ein Projekt in Runtime ist, wird das laufende Projekt automatisch beendet. Für Mobile Panels ist dieser Transfermodus aus Sicherheitsgründen deaktiviert.

Hinweis

Automatisches Beenden der Runtime

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im "Start Center".

Weiterführende Informationen zu den Transfereinstellungen finden Sie in der Dokumentation des verwendeten Bediengeräts.

Voraussetzung

- Im Projekt ist ein Bediengerät angelegt.
- Das Bediengerät ist mit dem Projektierungs-PC verbunden.
- Am Bediengerät ist das "Start Center" gestartet.
- Am Bediengerät ist im "Start Center" unter "Einstellungen" das Protokoll eingestellt, über welches das Projekt geladen wird.
- Am Bediengerät ist als Transfermodus "Automatisch" oder "Manuell" eingestellt.

Vorgehen

Um ein Projekt zu laden, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Projekt auf mehrere Bediengeräte gleichzeitig laden wollen, wählen Sie in der Projektnavigation alle gewünschten Bediengeräte mit der Mehrfachauswahl aus.
2. Wählen Sie im Kontextmenü eines Bediengeräts den Befehl "Laden in Gerät > Software".
3. Wenn der Dialog "Erweitertes Laden" geöffnet wird, konfigurieren Sie die "Einstellungen für das Laden". Achten Sie dabei darauf, dass die "Einstellungen für das Laden" den "Transfereinstellungen am Bediengerät" entsprechen:
   - Wählen Sie das verwendete Protokoll, z. B. Ethernet oder HTTP.
   - Konfigurieren Sie die entsprechenden Schnittstellenparameter am Projektierungs-PC.
   - Stellen Sie am Bediengerät bei Bedarf schnittstellen- oder protokollspezifische Einstellungen ein.
   - Klicken Sie auf "Laden".

   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet. Gleichzeitig wird das Projekt übersetzt. Das Ergebnis wird im Dialog "Vorschau laden" angezeigt.
4. Überprüfen Sie die angezeigten Voreinstellungen und ändern Sie diese bei Bedarf.
5. Klicken Sie auf "Laden".

Ergebnis

Das Projekt wird auf die ausgewählten Bediengeräte geladen. Ein vorhandenes Projekt wird ersetzt. Abhängig von den Einstellungen im Dialog "Vorschau laden" werden die Daten für die Benutzerverwaltung und / oder Rezepturen ersetzt.

Während des Ladens können Sie verfolgen, welche Dateien transferiert werden.

Wenn Fehler oder Warnungen beim Laden auftreten, werden im Inspektorfenster unter "Info > Laden" entsprechende Meldungen ausgegeben.

Nach dem erfolgreichen Laden ist das Projekt auf dem Bediengerät ablauffähig.

Hinweis

Falls die Übertragung unterbrochen wird, sichert WinCC automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Siehe auch

Daten des Bediengeräts sichern und wiederherstellen (Seite 2431)

Betriebssystem des Bediengeräts aktualisieren (Seite 2434)

Fehlermeldungen beim Laden von Projekten (Seite 2291)

Anpassen des Projekts für ein anderes Bediengerät (Seite 2229)

Herstellen der Verbindung zum Bediengerät (Seite 2199)

15.2.5.3 Projekt von externem Speichermedium laden

Einleitung

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf ein externes Speichermedium, z. B. einen USB-Stick, und laden es vom USB-Stick auf das Bediengerät.

Diese Funktion ist in Verbindung mit einem Bediengeräte-Image verfügbar, das mit der TIA Portal Version V14 oder höher kompatibel ist.

Diese Funktion steht für Basic Panels 2nd Generation zur Verfügung.

ACHTUNG
Datenverlust Wenn Sie ein Projekt mit der Option "Upgrade or downgrade Firmware" laden, dann wird das Betriebssystem des Bediengeräts aktualisiert. Dabei werden auf dem Bediengerät vorhandene Daten inklusive Bediengeräte-Kennwort gelöscht. Einstellungen im Start Center bleiben erhalten, Lizenzschlüssel werden vor dem Aktualisieren des Betriebssystems auf dem externen Speichermedium gesichert. Sichern Sie bei Bedarf Daten vor dem Laden.

Sie erzeugen die notwendigen Projektdaten in WinCC, indem Sie das Bediengerät projektieren und anschließend den Ordner des Bediengeräts (z. B. "HMI_1 [<Gerätetyp>]") per Drag&Drop oder Copy&Paste auf ein externes Speichermedium unter "Card Reader/USB-Speicher" ziehen.

Voraussetzungen

- Sie haben das Start Center am Bediengerät gestartet.
- Das Speichermedium mit dem gesicherten Projekt steckt im Bediengerät.

Vorgehen

1. Klicken Sie auf "Settings".
2. Öffnen Sie den Dialog "Service & Commissioning".
3. Betätigen Sie die Schaltfläche "Project Download".
   Mit "Project Download" haben Sie die Möglichkeit, ein einzelnes Projekt von einem externen Speichermedium in das Bediengerät zu laden.
4. Wählen Sie das Speichermedium aus, auf dem die Projektdaten gesichert sind.
5. Wählen Sie die Projektdatei aus.

   Hinweis

   Wenn kein Speichermedium am Bediengerät gesteckt ist, ist die Liste "Projects" leer.

6. Wählen Sie die Optionen für das Laden des Projekts:
   - "User administration:"
     Benutzerverwaltung auf dem Bediengerät mit der Benutzerverwaltung des Projekts überschreiben
   - "Recipe data:"
     Rezepturen des Bediengeräts mit den Rezepturen des Projekts überschreiben
   - "Upgrade or downgrade Firmware:"
     Firmware auf dem Bediengerät aktualisieren, falls die Firmware-Version des Bediengeräts zur Firmware-Version des Projekts inkompatibel ist

   Hinweis

   Sie können die Einstellungen zum Überschreiben der Benutzerverwaltung oder Rezepturen nur dann verändern, wenn sie vorher in der Projektierung nicht verriegelt wurden.

7. Bestätigen Sie mit der Schaltfläche "Accept".

Ergebnis

Nach dem Ladevorgang wird das neue Projekt auf dem Bediengerät gestartet.

15.2.6 Übersetzen und Laden mit Mulltiuser Engineering

15.2.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)

Einleitung

Beim Einsatz von Multiuser Engineering für die Bearbeitung Ihrer Projekte sollten Sie das Verhalten beim Übersetzen der Runtime-Projekte und das Verhalten beim Laden auf die Bediengeräte berücksichtigen.

Das Übersetzen und Laden auf ein Bediengerät ist sowohl in der Serverprojekt-Ansicht als auch in der lokalen Session möglich.

Weitere Informationen zum Thema "Multiuser Engineering" finden Sie unter "Multiuser Engineering einsetzen".

Grundlagen

Für Bediengeräte und RT Advanced sind im Multiuser Engineering folgende Szenarien möglich:

- Übersetzen in der Serverprojekt-Ansicht
- Übersetzen in der lokalen Session
- Laden aus der Serverprojekt-Ansicht
- Laden aus der lokalen Session

Hinweis

Das Komplettladen aus der Serverprojekt-Ansicht oder aus der lokalen Session unterscheidet sich nicht vom Komplettladen in einem Single-User-Projekt. Beim Komplettladen wird das aktuelle Runtime-Projekt aus der geraden aktiven Ansicht auf ein Bediengerät geladen.

Hinweis

Das Übersetzen und Laden von Projekten in einer lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Grundsätzlich können Sie in Multiuser Engineering-Projekten alle Befehle zum Übersetzen und Laden ausführen:

- "Software (komplett übersetzen)"
- "Übersetzen > Software (nur Änderungen)"
- "Software (komplett laden)"

Der Begriff "Projekt"

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen

bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Regeln

Folgende grundsätzliche Regeln gelten für Übersetzen und Laden in Multiuser Engineering:

- Das Runtime-Projekt, das in einer lokalen Session übersetzt wurde, bleibt immer lokal und wird nicht auf den Multiuser-Server hochgeladen. Es kann im Multiuser-Serverprojekt nicht gespeichert werden.
- Nur die Runtime-Projekte, die in der Serverprojekt-Ansicht übersetzt wurden, können im Multiuser-Serverprojekt gespeichert werden.

Weitere Informationen zu Multiuser Engineering finden Sie im Siemens YouTube Channel: Multiuser Engineering -ein Team arbeitet gleichzeitig an einem Projekt (https://www.youtube.com/watch?v=6Kq-4YWHJq4).

Siehe auch

Übersetzen in der Serverprojekt-Ansicht (Seite 2224)

Übersetzen in der lokalen Session (Seite 2225)

15.2.6.2 Übersetzen in der Serverprojekt-Ansicht

Grundlagen

Das Übersetzen und Laden von Projekten in der Serverprojekt-Ansicht unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Während Sie ein Projekt in der Serverprojekt-Ansicht übersetzen, wird das Multiuser-Serverprojekt blockiert. Andere Benutzer können in dieser Zeit keine Änderungen an diesem Serverprojekt durchführen. Das in der Serverprojekt-Ansicht übersetzte Runtime-Projekt wird zusammen mit dem Engineering Projekt im zentralen Multiuser-Server gespeichert. Durch das Blockieren des Multiuser-Serverprojekts ist gewährleistet, dass die Projektierungsdaten und das Runtime-Projekt synchron bleiben.

Hinweis

Wenn Sie in der Serverprojekt-Ansicht übersetzen und speichern, erhalten andere Benutzer beim "Aktualisieren" ihrer lokalen Session das von Ihnen aktualisierte Runtime-Projekt zusammen mit dem Engineering Projekt. Andere Benutzer müssen dann nach einer Aktualisierung nicht erneut die von Ihnen eingebrachten Änderungen übersetzen.

Beispiel: Übersetzen während des Eincheckens

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Wenn keine Übersetzungsfehler vorliegen, werden über "Änderungen speichern" beide Projekte, das modifizierte Engineering Projekt (mit den geänderten Variablen) und das übersetzte Runtime-Projekt, im zentralen Multiuser-Serverprojekt gespeichert.

Wenn Sie das Übersetzen während des Eincheckens auslassen, enthält das Projekt die Änderungen, die auf dem Server gespeichert wurden.

Der nächste Benutzer, der aus dem Serverprojekt eine lokale Session erstellt oder eine vorhandene lokale Session aktualisiert, muss zusätzlich zu seinen Änderungen noch Ihre zwei Änderungen übersetzen.

Hinweis

Das über mehrere lokale Sessions verteilte Arbeiten an einem gemeinsamen Projekt erhöht die Fehlerwahrscheinlichkeit. Daher ist es empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2223)

Übersetzen in der lokalen Session (Seite 2225)

15.2.6.3 Übersetzen in der lokalen Session

Grundlagen

Das Übersetzen und Laden von Projekten in der lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Da die lokale Session eine Kopie des Serverprojekts ist, ist der erste Übersetzungsstatus der lokalen Session mit dem des Serverprojekts identisch. Wenn im Serverprojekt unübersetzte Inhalte oder Fehlermeldungen beim Übersetzen vorhanden waren, werden sie in die lokale Session übertragen.

Hinweis

Es ist empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer und vermeiden Fehlerverbreitung.

Aktualisieren in der lokalen Session

Wenn Sie ein Projekt in der lokalen Session aktualisieren, wird die lokale Session komplett - inklusive des Übersetzungsstatus - durch den Inhalt des Serverprojekts ersetzt. Nur die für das Einchecken markieren Änderungen werden in der aktualisierten lokalen Session beibehalten und erzeugen zusätzliche Übersetzungsschritte in der lokalen Session.

Beispiel: Aktualisieren der lokalen Session

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Sie aktualisieren die Inhalte der lokalen Session, indem Sie auf die Schaltfläche "Aktualisieren" klicken. Nach dem Aktualisieren bekommt die lokale Session den Übersetzungsstatus des Serverprojekts. Zusätzlich sind Übersetzungsaufgaben vorhanden, die durch die Übernahme der geänderten Variablen begründet sind.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2223)

Übersetzen in der Serverprojekt-Ansicht (Seite 2224)

15.2.7 Runtime starten

15.2.7.1 Runtime auf dem Bediengerät starten

Einleitung

Sobald Sie das Projekt auf Bediengerät geladen haben, können Sie das Projekt in Runtime starten. Das Projekt wird auf dem Bediengerät in einer Datei mit folgender Endung gespeichert:

- Basic Panels: "*.srt"

Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben.

Die Programme zum Starten von Projekten am Bediengerät finden Sie im Installationsverzeichnis von Runtime.

Hinweis

Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im "Start Center".

Voraussetzung

- Auf dem Bediengerät ist WinCC Runtime installiert.
- Das Projekt ist auf das Bediengerät geladen.
- Das "Start Center" ist gestartet

Vorgehen

Auf einem Panel wird das Projekt in einem Ordner abgelegt, den Sie in den Transfereinstellungen des Bediengeräts angeben. Auf einem Panel wird die Anwendung "Start Center" gestartet. Abhängig von der Projektierung startet nach einer Verzögerungszeit das geladene Projekt automatisch.

Wenn das Projekt nicht automatisch startet:

1. Um das geladene Projekt zu starten, klicken Sie im "Start Center" auf "Start".

Weiterführende Informationen zum Starten von Projekten finden Sie in der Dokumentation des Bediengeräts.

15.2.8 Fehlermeldungen beim Laden von Projekten (Basic)

Mögliche Probleme beim Laden

Während ein Projekt auf ein Bediengerät geladen wird, werden im Ausgabefenster Statusmeldungen über den Fortschritt des Ladevorgangs ausgegeben.

Wenn beim Laden des Projekts auf das Bediengerät Probleme auftreten, ist die Ursache in den meisten Fällen eine der Folgenden:

- Falsche Einstellungen für Laden auf dem Bediengerät
- Falscher Bediengerättyp im Projekt
- Bediengerät ist nicht mit Projektierungs-PC verbunden.

Nachfolgend sind die häufigsten Fälle für einen Abbruch des Ladens mit möglicher Ursache und Abhilfe aufgeführt.

Das serielle Laden bricht ab

Mögliche Abhilfe: Wählen Sie eine kleinere Baudrate.

Das Laden wird wegen eines Kompatibilitätskonflikts abgebrochen

Mögliche Ursache	Abhilfe
Der Projektierungs-PC ist mit einem falschen Gerät verbunden, z. B. einer Steuerung	Kontrollieren Sie die Verkabelung. Korrigieren Sie die Kommunikationsparameter

Projekt wird nicht geladen

Mögliche Ursache	Abhilfe
Verbindung zum Bediengerät kann nicht aufgebaut werden (Meldung im Ausgabefenster)	Prüfen Sie die physikalische Verbindung zwischen Projektierungs-PC und Bediengerät. Prüfen Sie, ob sich das Bediengerät im Transfermodus befindet. Ausnahme: Fernsteuerung
Im Gerätemanager von Windows ist der Standard-Kommunikationstreiber nicht eingetragen	Kontrollieren Sie im Eigenschaftsfenster des Gerätemanagers den Gerätestatus des COM-Anschlusses.

Laden über die MPI/DP-Schnittstelle funktioniert nicht

Mögliche Ursache	Abhilfe
Am CP ist der "projektierte Betrieb" eingestellt, z. B. bei Verwendung der SIMATIC NET CD.	Stellen Sie mit der Anwendung "PC-Station einstellen" für den CP den "PG-Betrieb" ein. Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse". Laden Sie das Projekt aus WinCC auf den CP. Stellen Sie für den CP wieder den "projektierten Betrieb" ein.
Im PG/PC-Panel steht der Zugangspunkt "S7ONLINE" nicht auf einem physikalischen Gerät wie CP5611 (MPI). Ursache kann die Installation von "SIMATIC NET CD 7/2001" sein.	Stellen Sie mit der Anwendung "PG/PC-Panel" oder "PC-Station einstellen" den Zugangspunkt "S7ONLINE" auf ihr gewünschtes Gerät ein. Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse". Laden Sie das Projekt aus WinCC auf das Bediengerät. Stellen Sie den Zugangspunkt "S7ONLINE" wieder zurück auf das ursprüngliche Gerät.

Komplexität der Projektierung ist zu groß

Mögliche Ursache	Abhilfe
Ihre Projektierung enthält zu viele verschiedene Objekte oder Optionen für das ausgewählte Bediengerät.	Entfernen Sie alle Objekte eines Typs, z. B. alle Grafikanzeigen

15.2.9 Anpassen des Projekts für ein anderes Bediengerät

Einleitung

Wenn Sie ein WinCC Projekt auf ein Bediengerät laden, prüft WinCC, ob das Bediengerät mit dem im Projekt verwendeten Bediengerätetyp übereinstimmt. Wenn die Bediengerätetypen nicht übereinstimmen, erhalten Sie beim Ladevorgang eine Meldung.

Der Ladevorgang wird abgebrochen.

Anpassen des Projekts für das Bediengerät

Um das Projekt auf das angeschlossene Bediengerät zu laden, müssen Sie das Projekt entsprechend anpassen.

- Fügen Sie in der Projektnavigation ein neues Bediengerät ein. Wählen Sie in der Bediengeräteauswahl den richtigen Bediengerätetyp aus.
- Kopieren Sie die projektierten Bestandteile aus dem bisherigen Bediengerät in das neu angelegte Bediengerät.
  Viele Bestandteile kopieren Sie direkt in der Projektnavigation und der Detailansicht. Kopieren Sie z. B. den kompletten Ordner "Bilder" mithilfe des Kontextmenüs in den Ordner Bilder des neuen Bediengeräts.
- Bei einigen Einträgen in der Projektnavigation, für die im Kontextmenü der Befehl "Kopieren" nicht verfügbar ist, kopieren Sie die Inhalte über die Detailansicht.
- Selektieren Sie z. B. den Eintrag "Rezepturen" in der Projektnavigation. Die Rezepturen werden in der Detailansicht angezeigt.
- Selektieren Sie die Rezepturen in der Detailansicht und ziehen Sie diese per Drag&Drop auf den Eintrag "Rezepturen" des neuen Bediengeräts. Die Rezepturen werden kopiert. Eine Mehrfachselektion in der Detailansicht ist möglich.
- Projektieren Sie nicht kopierbare Bestandteile neu, z. B. Verbindungen, Bereichszeiger und Meldungen.
- Speichern Sie das Projekt auch zwischendurch.
- Übersetzen Sie das Projekt komplett.
- Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

Verknüpfung von Referenzen

Wenn Sie Objekte kopieren, werden die Referenzen zu verbundenen Objekten mitkopiert. Nach dem Kopieren der verbundenen Objekte werden die Referenzen wieder miteinander verknüpft.

Beispiel:

Sie kopieren ein Bild, in dem Objekte mit Variablen verknüpft sind. Nach dem Einfügen des Bilds in das neue Bediengerät sind die Variablennamen an den einzelnen Objekten eingetragen. Die Variablennamen sind rot markiert, da die Referenzen offen sind. Wenn Sie anschließend die Variablen kopieren und in das neue Bediengerät einfügen, werden die offenen Referenzen geschlossen. Die rote Markierung der Variablennamen wird aufgehoben.

Damit Referenzen zu verbundenen Objekten in der Steuerung vervollständigt werden, müssen Sie zuerst eine Verbindung zu der Steuerung projektieren.

Verwendung des Infobereichs

Wenn Sie das Projekt für das neue Bediengerät übersetzen, werden im Inspektorfenster auf der Registerkarte "Info" Fehler und Warnungen ausgegeben. Über den Kontextmenübefehl "Gehe zu" gelangen Sie direkt zur betreffenden Stelle, um den Fehler oder die Warnung zu korrigieren.

Arbeiten Sie die Liste der Fehler und Warnungen nacheinander ab.

Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

15.2.10 Projektgröße reduzieren

Einleitung

Beim Laden eines umfangreichen Projekts auf ein HMI-Bediengerät kann es vorkommen, dass der Speicherplatz des Bediengeräts für das Projekt nicht ausreicht. Sie haben mehrere Möglichkeiten, die Größe Ihres Projekts zu verringern.

Möglichkeiten zur Reduktion der Projektgröße

Sie haben mehrere Möglichkeiten, die Projektgröße zu reduzieren und Speicherplatz zu sparen:

- Anzahl der verfügbaren Runtime-Sprachen verringern
  Überprüfen Sie, ob alle selektierten Runtime-Sprachen tatsächlich benötigt werden. Deaktivieren Sie ggf. unter "Runtime Einstellungen > Sprache & Schriftart > Runtime-Sprache und Schriftauswahl" die Sprachen, die Sie nicht benötigen.
- Auf die Hilfetexte der S7 Diagnosemeldungen verzichten
  Um die Größe des Projekts zu verkleinern, können Sie das Herunterladen der Hilfetexte für die S7 Diagnosemeldungen deaktivieren. Um die Hilfetexte nicht auf das Bediengerät zu laden, deaktivieren Sie unter "Runtime-Einstellungen > Meldungen > Allgemein" die Option "S7 Diagnosehilfetexte herunterladen".
- Software komplett übersetzen
  Um die Projektdaten zu optimieren und veraltete Änderungen zu bereinigen, übersetzen Sie Ihr Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.
- Darstellung mithilfe von Stilen harmonisieren (ab WinCC V13)
  Es wird empfohlen, Bildobjekte mithilfe von Stilen zu harmonisieren. Vereinheitlichen Sie das Aussehen von Bildobjekten in Ihrem Projekt, um Projektdaten zu optimieren. Nutzen Sie durchgängig einen vorgegebenen oder einen benutzerdefinierten Stil bei der Projektierung der Bildobjekte.
- Automatische Aktualisierung von PLC Meldungen aktivieren (für S7 1500 Steuerungen und V14 HMI-Bediengeräte)
  Um den Speicherplatz zu sparen, legen Sie fest, dass die PLC-Meldetexte nur bei Bedarf in Runtime geladen werden. Aktivieren Sie dafür unter "Runtime Einstellungen > Meldungen > Steuerungsmeldungen" die Option "Automatische Aktualisierung". Stellen Sie sicher, dass in der entsprechenden Steuerung automatische Aktualisierung von Meldungen auch aktiviert ist.
  Die Option "Automatische Aktualisierung" steht bei Basic Panels nicht zur Verfügung.
  Eingesparter Speicherplatz hängt von der Anzahl der PLC-Meldungen und der Anzahl der Runtime-Sprachen ab.

- Anzahl der geladenen Schriftarten verringern
  Überprüfen Sie, ob die Anzahl der benutzerdefinierten heruntergeladenen Schriftarten verkleinert werden kann. Legen Sie ggf. unter "Runtime Einstellungen > Sprache & Schriftart > Runtime-Sprache und Schriftauswahl" für die benötigten Runtime-Sprachen nur die Standard-Schriftarten fest.
  Um Speicherplatz zu sparen, nutzen Sie bei der Projektierung weniger unterschiedlichen Schriftfamilien.
- Größe der Grafiken verringern
  Überprüfen Sie die Größe der Grafiken, die Sie im Projekt verwenden. Reduzieren Sie ggf. die Größe der Grafiken, indem Sie die Auflösung verringern oder ein stärkeres Komprimierungsformat ohne merkbaren Qualitätsverlust für die Projektgrafiken wählen.
  Um bei den Basic-Bediengeräten die Projektgröße kleiner zu halten, harmonisieren Sie die Größen, in welchen die Projektgrafiken verwendet werden.
  Wählen Sie passende Grafikformate für Ihre Bilder: nutzen Sie PNG-Grafiken für Zeichnungen, die keine Vektorgrafiken sind und JPEG-Grafiken für Fotos.

15.2.11 Grundlagen zur Bedienung in Runtime

15.2.11.1 Übersicht

Überblick über die Bedienung eines Projekts

Je nach Bediengerät stehen bei Basic-Bediengeräten folgende Möglichkeiten für die Eingabe zur Verfügung:

- Touch-Screen
- Funktionstasten
- Maus und Tastatur

Über den Touch-Screen, Funktionstasten oder Maus und Tastatur bedienen Sie das Control Panel/Start Center oder das Projekt, das auf Ihrem Bediengerät läuft.

Hinweis

Fehlbedienung

Innerhalb des Projekts können Bedienhandlungen enthalten sein, die umfassende anlagenspezifische Kenntnisse des Bedieners voraussetzen.
Stellen Sie sicher, dass nur geschultes Fachpersonal die Anlage bedient.

Bedienvarianten eines Bediengeräts

Abhängig von Ihrem Bediengerät bedienen Sie Ihre Anlage folgendermaßen:

- Bedienung über Touch-Screen
  Das Display des Geräts ist berührungssensitiv. Sie bedienen die Bedienobjekte im Display mit dem Finger oder einen Touch-Stift.
- Bedienung über Steuer- und Funktionstasten
  Im Gehäuse des Geräts sind Steuer- und Funktionstasten integriert.
  - Steuertasten haben eine festgelegte Funktion, z. B. Navigieren oder Quittieren von Meldungen.
  - Funktionstasten sind frei belegbar und ihre Funktionen daher projektspezifisch.
- Bedienung mit Maus und Tastatur
  Über eine integrierte USB-Schnittstelle werden Maus und Tastatur angeschlossen. Sie bedienen die Bedienobjekte mit Maus und Tastatur.

Individuell projektierte Bedienung

Dem Projekteur stehen unterschiedliche Möglichkeiten zur Verfügung, die Bedienung einzurichten.

Beispiele für Aktionen, deren Ausführung immer projektspezifisch festgelegt ist:

- Bildwechsel
- Protokollierung
- Runtime-Sprache umstellen

Für die Ausführung bestimmter Funktionen gibt es keine speziellen Bedienobjekte. Der Projekteur legt die Ausführung projektspezifisch fest. Der Bildwechsel kann z. B. über eine Schaltfläche oder eine Funktionstaste ausgelöst werden.

Informationen zu projektspezifischen Bedienungen finden Sie in der Anlagendokumentation.

15.2.11.2 Bedienung über Touchscreen

Überblick zur Bedienung über Touchscreen

Über den Touch-Screen bedienen Sie das Bediengerät oder das Projekt, das auf Ihrem Bediengerät läuft.

Touch-Screen bedienen

ACHTUNG
Beschädigung des Touch-Screens Berühren Sie den Touch-Screen nicht mit spitzen oder scharfen Gegenständen. Verzichten Sie auf schlagartige Berührungen des Touchscreen mit harten Gegenständen und vermeiden Sie eine dauerhafte Bedienung mit Gesten. Beides kann zu einer erheblichen Reduzierung der Lebensdauer bis hin zum Totalausfall des Touchscreen führen. **Auslösen unbeabsichtigter Aktionen** Das gleichzeitige Berühren mehrere Bedienobjekte kann unbeabsichtigte Aktionen auslösen. Berühren Sie immer nur ein Bedienobjekt am Bildschirm. Bedienobjekte sind berührungssensitive Darstellungen am Bildschirm des Bediengeräts.

Besonderheiten beim Bedienen über Touch-Screen

Die Bedienung über Touch-Screen zeichnet sich durch folgende Besonderheiten aus:

- Aktivieren
  Um ein Bedienelement zu aktivieren, berühren Sie den Touch-Screen mit dem Finger oder einen Touch-Stift. Um einen Doppelklick zu erzeugen, berühren Sie das Bedienelement zweimal kurz hintereinander.
- Werteingabe
  Zahlen und Buchstaben geben Sie am Touch-Screen über eine Bildschirmtastatur ein.
- Achtsame Bedienung
  Wenn Sie gleichzeitig mehrere Bedienelemente berühren, lösen Sie damit unbeabsichtigte Aktionen aus.

Werteingabe über Bildschirmtastatur

Die Bildschirmtastatur wird am Display angezeigt, sobald Sie ein Bildobjekt bedienen, das eine Eingabe erfordert. Abhängig vom Bediengerät und vom projektierten Bedienobjekt werden unterschiedliche Bildschirmtastaturen für numerische oder alphanumerische Werte angezeigt. Beim Beenden der Eingabe wird die Bildschirmtastatur wieder ausgeblendet.

Dateibrowser (Windows 8 oder höher)

Auf einem PC mit Windows 8 oder höher mit einem Touch-Screen kann der Dateibrowser-Dialog nur mit der Maus, Tastatur oder Bildschirmtatstatur ohne Verwendung von Touch-Funktion bedient werden. Auf einem Touch-Screen PC mit Windows 8 oder höher nutzen Sie den Dateibrowser-Dialog von Windows-Betriebssystem mithilfe eines Skripts.

Bildschirmtastatur

Darstellung

Die folgende Abbildung zeigt den prinzipiellen Aufbau einer Bildschirmtastatur auf einem TP1500 Basic.

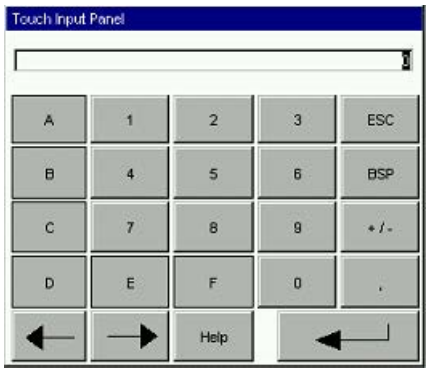

Zahlen

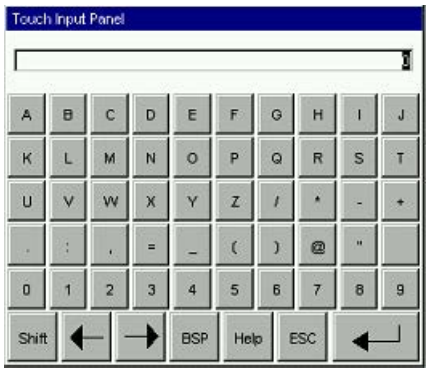

Buchstaben

Bedienelemente

Folgende Tasten sind auf der Bildschirmtastatur aller Bediengeräte vorhanden:

Schaltfläche	Name	Funktion
←	Cursor nach links	Navigiert nach links
→	Cursor nach rechts	Navigiert nach rechts
BSP	Backspace	Löscht ein Zeichen
ESC	Escape	Bricht die Eingabe ab
↵	Enter	Bestätigt die Eingabe
Help	Help	Zeigt den Hilfetext an Nur wenn für das Bedienobjekt ein Hilfetext projektiert wurde, wird diese Taste angezeigt.

Bildschirmtastatur bei Basic Panels 2nd Generation

Darstellung

Die folgende Abbildung zeigt den prinzipiellen Aufbau einer Bildschirmtastatur auf einem Basic panel 2nd Generation.

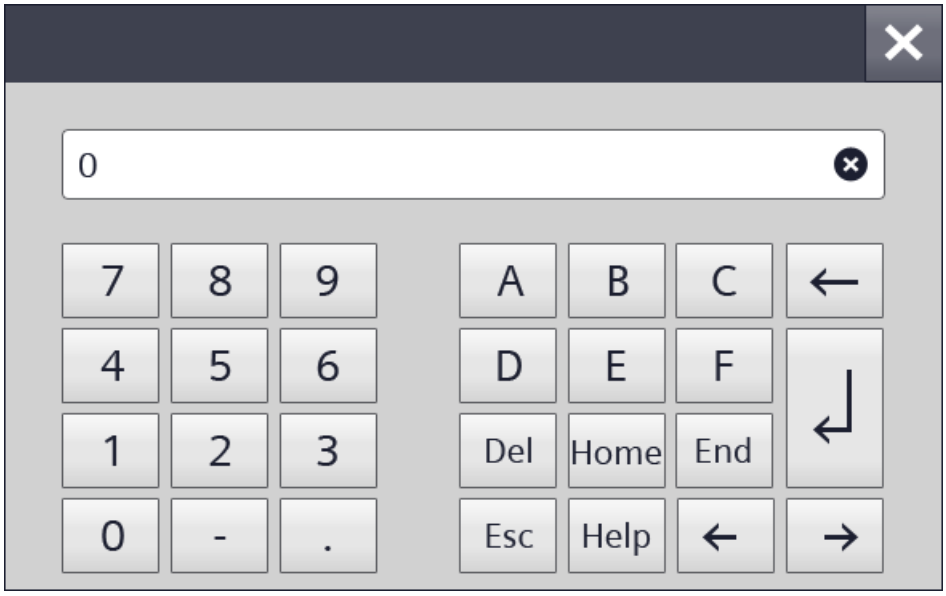

Zahlen

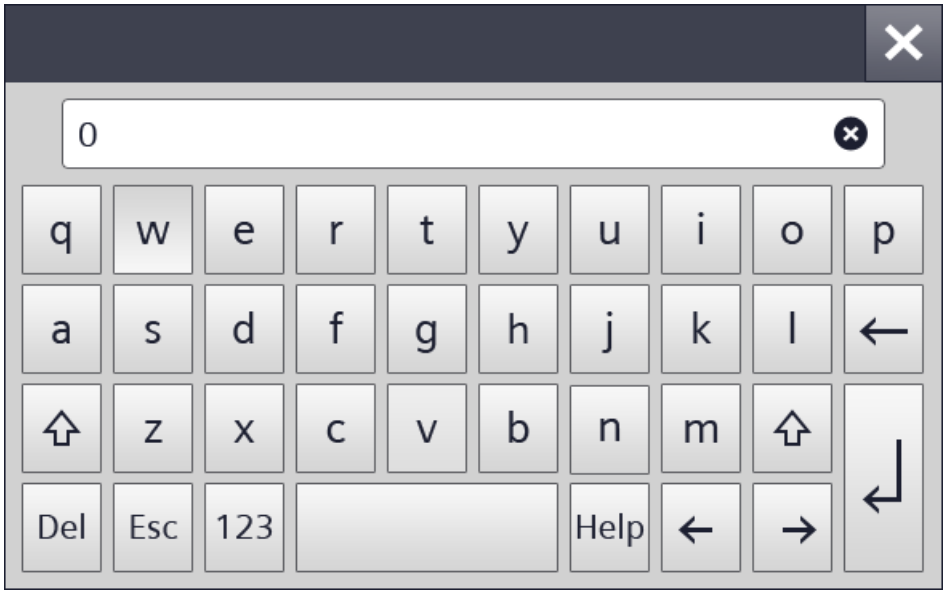

Buchstaben

Bedienelemente

Folgende Tasten sind auf der Bildschirmtastatur aller Bediengeräte vorhanden:

Schaltfläche	Name	Funktion
←	Cursor nach links	Navigiert nach links
→	Cursor nach rechts	Navigiert nach rechts
←	Backspace	Löscht ein Zeichen
Esc	Escape	Bricht die Eingabe ab
↵	Enter	Bestätigt die Eingabe
Help	Help	Zeigt den Hilfetext an Nur wenn für das Bedienobjekt ein Hilfetext projektiert wurde, wird diese Taste angezeigt.

15.2.11.3 Bedienung über Tasten

Überblick zur Bedienung über Tasten

Einleitung

Mit dem Tasten des Bediengeräts bedienen Sie das Control Panel/Start Center Ihres Geräts oder das Projekt, das auf Ihrem Gerät läuft. Abhängig vom Gerät stehen Ihnen Steuertasten und Funktionstasten zur Verfügung.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

Steuertasten und Tastenkombinationen

Einleitung

Die folgenden Tabellen zeigen die Steuertasten, mit denen Sie das Projekt bedienen.

Hinweis

Die Verfügbarkeit der Steuertasten ist abhängig vom verwendeten Bediengerät.

Funktionen an Tasten-Bediengeräten lösen Sie entweder über eine Taste oder Tastenkombination aus. Bei Tastenkombinationen halten Sie die erste Taste gedrückt. Danach drücken Sie die zweite Taste.

Im Display navigieren

Taste oder Tastenkombination		Funktion	Beschreibung
TAB		Tabulator	Wählt das nächste Bedienobjekt in der Tab-Reihenfolge aus
SHIFT	TAB	Tabulator	Wählt das vorherige Bedienobjekt in der Tab-Reihenfolge aus
◀ ▶ ▼ ▲		Cursortasten	Wählen das nächste Bedienobjekt, links, rechts, oberhalb oder unterhalb des aktuellen Bildobjekts aus Navigieren im Bedienobjekt

Bedienobjekte bedienen

Taste oder Tastenkombination		Funktion	Beschreibung
ENTER		Eingabetaste	• Bedient Schaltflächen. • Übernimmt und beendet eine Eingabe • Öffnet eine Auswahlliste • Schaltet innerhalb eines Eingabefelds zwischen Zeichenmodus und Normalmodus um Im Zeichenmodus ist ein einzelnes Zeichen markiert. In diesem Modus können Sie mit den Cursortasten im Zeichensatz weiterschalten.
SHIFT	▶ ◀ ▲ ▼	Cursor positionieren	Positionieren den Cursor innerhalb eines Bedienobjekts, z. B. im E/A-Feld
←		Zeichen löschen	Löscht das Zeichen links von der aktuellen Cursor-Position
INS DEL		Zeichen löschen	Löscht das Zeichen rechts von der aktuellen Cursor-Position
ESC		Abbrechen	• Löscht eingegebene Zeichen einer Werteingabe und stellt den ursprünglichen Wert wieder her • Schließt den aktiven Dialog.
FN	HOME	Zum Anfang blättern	Blättert in einer Liste zum Anfang
HOME		Zurückblättern	Blättert in einer Liste um eine Seite zurück
FN	END	Zum Ende blättern	Blättert in einer Liste zum Ende.
END		Weiterblättern	Blättert in einer Liste um eine Seite weiter
ALT	▼	Auswahlliste öffnen	Öffnet eine Auswahlliste
CTRL	ENTER	Wert übernehmen	Übernimmt den ausgewählten Wert in der Auswahlliste, ohne die Liste zu schließen

Tastenkombination eingeben

Taste	Funktion	Zweck
A-Z	Umschalten (Ziffern/Buchstaben)	Schaltet die Belegung der Eingabetasten von Ziffern auf Buchstaben um • Keine LED leuchtet: Die Ziffern-Belegung ist aktiv. Einmaliges Drücken der Taste schaltet auf Buchstaben-Belegung um. • Die rechte oder die linke LED leuchtet: Die linke oder rechte Buchstaben-Belegung ist aktiv. Jedes Drücken der Taste schaltet zwischen der linken Buchstaben-Belegung, der rechten Buchstaben-Belegung und der Ziffern-Belegung um.
SHIFT	Umschalten (Groß-/Kleinschreibung)	Verwendung in Tastenkombinationen, z. B. Umschalten auf Großbuchstaben
FN	Auf zusätzliche Tastenbelegung umschalten	Auf einigen Tasten finden Sie in der linken unteren Ecke blaue Sonderzeichen, z. B. das Prozentzeichen "%". Um diese Zeichen einzugeben, drücken Sie gleichzeitig die links abgebildete Sonderzeichen-Taste.
CTRL	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. Navigation in Kurvendarstellung
ALT	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. bei Bildobjekt "Beobachtungstabelle"

Meldungen quittieren

Taste	Funktion	Zweck
ACK	Quittieren	Quittiert die aktuell angezeigte Meldung oder alle Meldungen einer Meldegruppe Die LED leuchtet, solange unquittierte Meldungen anstehen.

Hilfetext anzeigen

Taste	Funktion	Beschreibung
HELP	Hilfetext anzeigen	Öffnet zum markierten Objekt ein Fenster mit dem projektierten Hilfetext, , z. B. für eine Meldung oder ein E/A-Feld Wenn zum markierten Objekt ein Hilfetext existiert, leuchtet die LED.

Taste oder Tastenkombination

Funktionstasten

Die Belegung der Funktionstasten (F1, F2, F3 …) wird bei der Projektierung festgelegt.

Funktionstasten mit globaler Funktionsbelegung

Eine global belegte Funktionstaste löst unabhängig vom angezeigten Bild immer dieselbe Aktion am Bediengerät oder in der Steuerung aus. Eine solche Aktion ist z. B. das Aktivieren eines Bildes oder das Schließen eines Meldefensters.

Funktionstasten mit lokaler Funktionsbelegung

Eine Funktionstaste mit lokaler Funktionsbelegung ist bildspezifisch und damit nur innerhalb des aktiven Bildes wirksam.

Die Funktion einer lokal belegten Funktionstaste kann von Bild zu Bild unterschiedlich sein.

Innerhalb eines Bildes hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

Funktionstasten bedienen

Hinweis

Funktionstaste nach einem Bildwechsel bedienen

Wenn Sie nach einem Bildwechsel eine Funktionstaste bedienen, kann die zugehörige Funktion im neuen Bild bereits angestoßen werden, bevor das Bild vollständig aufgebaut ist.

15.2.11.4 Navigieren im Display (BS)

Einleitung

Sie navigieren auf dem Display Ihres Bediengeräts folgendermaßen:

- Zwischen den projektierten Bildobjekten
- Innerhalb von Bildobjekten
  Wenn Sie ein komplexes Bildobjekt aktivieren, wechselt der Cursorfokus in das Bildobjekt und folgt der dortigen Tab-Reihenfolge.
- In Tabellen von Bildobjekten

Vorgehen

- Um in der festgelegten Tab-Reihenfolge zu navigieren, drücken Sie die Taste <TAB>.
- Um frei zwischen den Bedienelementen zu navigieren, drücken Sie die Cursortasten.

Abhängig von der Projektierung Ihres Bediengeräts, verwenden Sie zum Navigieren auch Funktionstasten oder Tastenkombinationen.

Wenn Sie Ihr Bediengerät über den Touchscreen oder mit der Maus bedienen, navigieren Sie implizit durch das Auslösen einer gewünschten Aktion. Dazu berühren oder klicken Sie das Bedienelement.

Ergebnis

Die Bedienelemente erhalten entsprechend der gewählten Reihenfolge den Cursorfokus. Sie können an dem ausgewählten Bedienelement eine Aktion auslösen.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.2.11.5 Aktion auslösen

Einleitung

Eine Aktion an einem Bedienelement auslösen kann Folgendes bedeuten:

- Ein Befehl wird ausgeführt.
  Beispiel: Um ein Skript zu triggern oder eine vordefinierte Funktion auszuführen, klicken Sie auf eine Schaltfläche.
- Ein Objekt wird aktiviert.
  Beispiel: Um einen Wert einzugeben, aktivieren Sie eine Tabellenzelle mit der Taste <Enter>.

Voraussetzung

- Sie sind zum Bedienelement navigiert, an dem Sie die Aktion auslösen wollen.
- Das Bedienelement hat den Cursorfokus.

Vorgehen

- Drücken Sie die Taste <Enter>.
  Oder
- Berühren Sie das Bedienelement am Touchscreen einfach oder zweimal kurz hintereinander.
  Oder
- Klicken oder doppelklicken Sie das Bedienelement mit der Maus.

Ergebnis

Folgende Ergebnisse sind z. B. möglich:

- Der gewünschte Befehl wird ausgeführt.
- Die Bildschirmtastatur wird geöffnet und/oder der Cursor blinkt im Eingabebereich des Bedienelements.
- Das Element ist ausgewählt und kann bewegt werden.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.2.11.6 Wert eingeben

Einleitung

Je nach Eingabeformat geben Sie numerische oder alphanumerische Werte in ein Eingabefeld ein.

Abhängig von der vorhandenen Hardware geben Sie diese Werte über die Bildschirmtastatur, die Steuertasten des Bediengerätes oder eine externe Tastatur ein.

Voraussetzung

- Das Objekt ist ein Eingabe- oder Tabellenfeld.
- Das Bedienelement ist aktiviert.

Wert eingeben

1. Geben Sie den gewünschten Wert ein.
2. Um den Wert zu bestätigen und das Feld zu verlassen, drücken Sie die Taste <Enter>.
3. Um den Wert zu verwerfen und das Feld zu verlassen, drücken Sie die Taste <Esc>.

Ergebnis

Ein Wert ist eingegeben oder verworfen. Sie navigieren bei Bedarf zum nächsten Bedienelement.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.2.11.7 Bedienelemente bewegen

Einleitung

Sie bewegen in Runtime bei Bedarf die beweglichen Bedienelemente eines Bildobjekts mit der Maus oder über den Touchscreen, z. B. einen Schieberegler oder einen Scrollbalken. Im Folgenden ist die Bedienung mit Tastatur beschrieben.

Voraussetzung

- Ein bewegliches Bedienelement ist aktiviert.

Vorgehen

- Um das Bedienelement zu bewegen, gehen Sie abhängig vom Bedienobjekt folgendermaßen vor:
  - Standard für Touchscreen:: Drücken Sie die Cursortasten.
  - Standard für Tastengeräte:: Drücken Sie <SHIFT> und die Cursortasten.
  - Schalter: Drücken Sie <ENTER>
  - Schieberegler: Drücken Sie <Bild Ab> oder <Bild Auf>

1. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Vorgehen beim Schieberegler

1. Um das Bedienelement zu bewegen, drücken Sie die Tasten " Cursortasten.
2. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Ergebnis

Die Position des beweglichen Bedienelements und die Anzeige im Bildobjekt haben sich verändert.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.2.11.8 Hilfetext anzeigen

Einleitung

Abhängig von der Projektierung stehen Ihnen als Hilfetext zusätzliche Informationen und Bedienhinweise zur Verfügung. Der Hilfetext ist einem Bedienobjekt, einer Meldung oder dem geöffnetem Bild zugeordnet. Der Hilfetext eines E/A-Feldes kann z. B. Hinweise zum einzugebenden Wert enthalten.

Alternativ zur Taste <Help> des Bediengeräts verwenden Sie bei Eingabeobjekten die Taste <Help> der Bildschirmtastatur.

Voraussetzung

- Ein Hilfetext ist am Bedienobjekt, am Bild oder an einer Meldung projektiert.

Hilfetext aufrufen

1. Aktivieren Sie das gewünschte Bedienobjekt.
2. Drücken Sie die Taste <Help> des Bediengeräts.
   Der Hilfetext zum Bedienobjekt wird angezeigt.

Wenn Sie ein Eingabeobjekt über den Touchscreen bedienen, wird die Bildschirmtastatur geöffnet. Wenn die Taste <Help> angezeigt wird, ist für das Bedienobjekt oder das aktuelle Bild ein Hilfetext projektiert.

Wenn zum markierten Bildobjekt kein Hilfetext existiert, wird der Hilfetext für das aktuelle Bild angezeigt, falls dieser projektiert wurde.

Bei langen Hilfetexten verwenden Sie den Scrollbalken.

Abhängig von der Projektierung rufen Sie den Hilfetext auch über ein projektiertes Bedienobjekt auf.

Zwischen Hilfetexten wechseln

- Um zwischen den Hilfetexten des Bedienobjekts und des Bildes zu wechseln, aktivieren Sie das Hilfetextfenster.

Hilfetext ausblenden

- Um den Hilfetext auszublenden, drücken Sie die Taste <Esc> oder erneut die Taste <Help>.

15.2.11.9 Runtime-Sprache umstellen

Einleitung

Das Projekt auf dem Bediengerät kann mehrsprachig sein. Zum Wechseln der am Bediengerät eingestellten Sprache während des laufenden Betriebs ist ein entsprechendes Bedienobjekt projektiert.

Nach dem Starten des Projekts ist immer die zuletzt aktive Sprache eingestellt.

Voraussetzung

- Die gewünschte Sprache für das Projekt ist am Bediengerät verfügbar.
- Die Funktion für die Sprachumschaltung ist mit einem Bedienobjekt verknüpft, z. B. mit einer Schaltfläche.

Sprache auswählen

Zwischen den Sprachen können Sie jederzeit umschalten. Nach dem Aufruf der Funktion für die Sprachumschaltung werden die sprachabhängigen Objekte in der neuen Sprache angezeigt.

Um die Sprache in Runtime umzuschalten, haben Sie folgende Möglichkeiten:

- Über ein projektiertes Bedienobjekt schalten Sie in einer Auswahl von Sprachen weiter.
- Über ein projektiertes Bedienobjekt wählen Sie die gewünschte Sprache direkt aus.

15.2.12 Barcodes mit optischen Handlesegeräten eingeben

Einleitung

Mit den optischen Handlesegeräten haben Sie die Möglichkeit, Bauteile, Maschinen und andere Objekte optisch zu identifizieren und auf Ihrem Bediengerät in einige Bedienobjekte direkt einzugeben.

Optische Handlesegeräte erfassen zweidimensionale Data Matrix Codes, eindimensionale Barcodes und postalische Barcodes.

Folgende optische Handlesegeräte werden unterstützt:

- SIMATIC MV320
- SIMATIC MV340

Die Vorlagen für die Einstellungen und Anweisungen zur Konfiguration finden Sie im Handbuch Ihres optischen Handlesegeräts.

Hinweis

Das optische Handlesegerät wird an die USB-Schnittstelle des HMI-Bediengeräts angeschlossen. Nur ein Gerät kann die USB-Schnittstelle zu einer Zeit nutzen. Daher ist die gleichzeitige Verwendung von USB-Tastatur und optischen Lesegeräts oder zwei optischen Lesegeräten nicht möglich.

Hinweis

Wenn Sie das optische Lesegerät an einem Projektierungs-PC anschließen, stellen Sie sicher, dass die USB-Schnittstelle für das Lesegerät ausreichend Strom zur Verfügung stellt.

Vorgehen

Sie lesen die Markierung eines Objekts mithilfe des angeschlossenen optischen Lesegeräts ein. Der Wert wird auf dem HMI-Bediengerät in das Objekt angegeben, der in Runtime den Fokus erhält.

Nach dem Einlesen bestätigen Sie den Wert mit der Eingabetaste oder mit dem "Suffix - Enter", den Sie zuvor in den Einstellungen Ihres optischen Lesegeräts konfiguriert haben.

Basic Panels, Comfort Panels, Mobile Panels, RT Advanced

Folgende Objekte unterstützen die Eingabe mit optischen Handlesegeräten:

Objekt	Vorbedingung für die Eingabe
E/A-Feld Datum/Uhrzeit-Feld	Der entsprechende Datentyp ist ausgewählt. Das Objekt und die Variablenlänge sind entsprechend konfiguriert. Das Bedienobjekt hat den Cursorfokus.
Rezepturanzeige	Rezepturdatensatz hat den Cursorfokus.
Benutzeranzeige	Anmeldedialog ist geöffnet. Benutzername oder Passwort haben den Cursorfokus.
HTML-Browser	Das Bedienobjekt hat den Cursorfokus.
Runtime Dialoge, die die Tastatureingabe unterstützen	Dialog ist geöffnet und das entsprechende Eingabefeld hat den Cursorfokus.

Comfort Panels, Mobile Panels, RT Advanced

Folgende Objekte unterstützen die Eingabe mit optischen Handlesegeräten:

Objekt	Vorbedingung für die Eingabe
Sm@rtClient-Anzeige	Für Sm@tClient ist Schreibzugriff konfiguriert. Das Bedienobjekt hat den Cursorfokus.
File Browser	Das Feld "File path" hat den Cursorfokus.
Beobachtungstabelle	Das Eingabefeld "Steuerwert" hat den Cursorfokus.
PDF-Anzeige	Das Suchfeld hat den Cursorfokus.
SINUMERIK NC-Bedienanzeige	Das Bedienobjekt hat den Cursorfokus.

Ergebnis

Der Code wurde erfasst und in das entsprechende Eingabefeld eingegeben.

15.2.13 Sicherheit auf dem Bediengerät

15.2.13.1 Grundlagen zu Signaturen

Übersicht

Die Bediengerätesoftware sowie die Runtime- und Geräteoptionen werden durch Signaturen gegen Manipulation geschützt. Auch die Geräteoptionen, die über ProSave angeboten werden, sind ebenfalls signiert.

Bediengerätesignatur

Die Gültigkeit der Bediengerätesignatur wird beim Laden der Bediengerätesoftware geprüft. Wenn die Bediengerätesignatur ungültig ist, wird das Laden der Bediengerätesoftware mit einer Fehlermeldung abgebrochen.

Signatur für Runtime- und Geräteoptionen

Wenn bei der Signaturprüfung die Signatur als ungültig erkannt wurde, informiert eine Fehlermeldung Sie darüber. In ProSave wird ebenfalls eine Fehlermeldung ausgegeben, die auf die Ungültigkeit der Signatur hinweist.

Hinweis

Um das Laden der Bediengerätesoftware oder der Optionen ohne Signatur (für Versionen vor V14) dennoch zu ermöglichen, müssen Sie die Überprüfung der Signatur ausschalten.

Weitere Vorgehensweise finden Sie unter Überprüfung der Signatur ausschalten (Seite 2248).

Siehe auch

Überprüfung der Signatur ausschalten (Seite 2248)

15.2.13.2 Überprüfung der Signatur ausschalten

Vorgehensweise

1. Tippen Sie im Control Panel des Bediengeräts zwei Mal auf "Transfer".
   Der Dialog "Transfer Settings" wird geöffnet.
   Standardmäßig ist die Option "Validate signatures" aktiviert.

   Hinweis

   Wenn Sie ein älteres WinCC Projekt auf ein Bediengerät laden, müssen Sie die Einstellungen zur Überprüfung der Signatur beachten.

2. Um das Laden der Bediengerätesoftware ohne Signatur oder das Herunterladen einer vertrauenswürdigen Option eines Drittanbieters zu ermöglichen, deaktivieren Sie die Option "Validate signatures".

Siehe auch

Grundlagen zu Signaturen (Seite 2247)

15.3 Runtime Advanced und Panels

15.3.1 Einstellungen für Runtime

15.3.1.1 Einstellungen für Runtime

Einleitung

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen".

Hinweis

Geräteabhängigkeit

Die Auswahl an Optionen in den Einstellungen für Runtime ist abhängig vom Bediengerät.

Übersicht

In den Runtime-Einstellungen projektieren Sie Optionen aus verschiedenen Bereichen Ihres Projekts:

- Darstellung in Runtime
  - Startbild
  - Standardvorlage
  - Standardstil des Projekts und Stil des Bediengeräts
  - Bildschirmauflösung
  - Vollbildmodus
  - Grafiksammlung
  - Bitselektion für Farbe und Blinken
  - Bitselektion für Text- und Grafiklisten
- Projektkennung zur Kompatibilitätsprüfung mit der Steuerung
- Archivierungssprache
- Fernbedienung über Sm@rt Server im Bereich "Dienste"
  Weitere Hinweise dazu finden Sie in der WinCC Online-Hilfe unter "Optionen > Sm@rt Optionen"

- Bedienung in Runtime
  - Taskumschaltung
  - Zusätzliche Daten zum Debuggen von Skripten laden: Objektnamen und Kommentare in Skripten
  - Grenzwerte als Tooltipp anzeigen
  - Bildschirmtastatur
  - Funktionstasten deaktivieren
  - Loslassen von Schaltflächen beim Verlassen der Schaltfläche
  - Tastaturbelegung und Tastenkombinationen
- GMP-konforme Projektierung
  Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Optionen > WinCC Audit".
- Meldungen
  - Pufferüberlauf
  - Text der Quittiergruppe
  - Protokoll
  - Farben der Meldeklassen
  - Konfiguration der Systemmeldungen
  - Konfiguration der Steuerungsmeldungen
  - Persistenter Meldepuffer
- Benutzerverwaltung
  - Konfiguration Kennwort und Abmeldezeit
  - Grenze für Login-Versuche
  - Hierarchieebenen
  - Konfiguration SIMATIC Logon
- Sprache und Schriftart
  - Verfügbare Runtime-Sprachen
  - Reihenfolge der Sprachumschaltung
  - Standardschrift
  - Lizenzstatus der Schriftarten
- Konfiguration des OPC UnifiedArchitecture Server
- Einstellungen zum Synchronisieren von PLC- und HMI-Variablen

Siehe auch

Einstellungen zum Synchronisieren (Seite 564)

Darstellung in Runtime projektieren (Seite 2251)

Bedienung in Runtime projektieren (Seite 2253)

Meldepufferüberlauf projektieren (Seite 927)

Meldeklassen anlegen (Seite 818)

Systemmeldungen bearbeiten (Seite 836)

Steuerungsmeldungen bearbeiten (Seite 837)

Protokollieren von Meldungen projektieren (Seite 909)

Einstellungen für die Benutzerverwaltung (Seite 1377)

Kompatibilitätsprüfung mit der Steuerung einstellen (Seite 2255)

Sprachen verwalten (Seite 2095)

Arbeiten mit Archiven (Seite 1071)

15.3.1.2 Darstellung in Runtime projektieren

Einleitung

In den Runtime-Einstellungen legen Sie fest, wie Bilder, Meldungen und Sprache dargestellt werden.

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen". Die Auswahl an Optionen ist abhängig vom Bediengerät.

Anzeige am Zielsystem

In WinCC stellen Sie ein, wie das generierte Projekt in Runtime dargestellt wird:

- Wählen Sie, ob das Projekt als Vollbild oder in einem Fenster angezeigt wird. Das Fenster ist kleiner als der Bildschirm. Im Vollbildmodus beansprucht das Projekt den gesamten Bildschirm. Ein Fenster und Bedienelemente für das Fenster sind nicht mehr vorhanden. Um die Darstellung Vollbild beim Start zu verwenden, klicken Sie im Editor "Einstellungen" auf "Allgemein". Aktivieren Sie im Bereich "Bild" das Kontrollkästchen "Vollbildmodus".

  Hinweis

  Darstellung nicht vollständig

  Wenn der Bildschirm des Bediengeräts nicht die projektierte Größe (in Pixeln) hat, wird das Projekt im Vollbildmodus nur auf einem Teil des Bildschirms angezeigt.

- Um die Taskleiste auszublenden, deaktivieren Sie die Taskleiste unter Windows.
- Unter "Allgemein > Bild" legen Sie fest, ob das Bediengerät den Standardstil des Projekts oder einen anderen vordefinierten Stil verwendet.

Einstellungen zur Sprache

- Archivierungssprache
  Unter "Allgemein > Archive" legen Sie fest, in welcher Sprache die Archive geschrieben werden. Die gewählte Archivsprache bleibt immer gleich, auch wenn der Bediener in Runtime die Sprache umschaltet.
  Wenn Sie "Anlaufsprache" wählen, bleibt die Archivsprache bis zum Runtime-Stopp gleich, auch wenn der Bediener die Sprache umgestellt hat. Beim nächsten Runtime-Start jedoch wird die zuletzt eingestellte Runtime-Sprache als Archivsprache verwendet.
- Runtime-Sprache
  Unter "Sprache & Schriftart" legen Sie alle Runtime-Sprachen und Ihre Darstellung fest.

Hinweis

Schriftstile "Fett" und "Kursiv"

Die Schriftstile "Fett" und "Kursiv" werden in Runtime nicht auf allen Bediengeräten unterstützt.

Hinweis

Basic Panels

Wenn Sie die Schriftgröße bzw. Schriftstil ungünstig wählen, kann es dazu führen, dass Texte abgeschnitten dargestellt werden. Dabei sind z. B. folgende Texte betroffen:

- Tooltips
- langer Meldetext
- Texte in Dialogen

Einstellung für bestimmte Bildobjekte

- Bitselektion festlegen
  Unter "Bilder > Bitselektion" legen Sie fest, ob die Bitauswahl auf diesem Bediengerät für Folgendes verwendet wird:
  - Text und Grafik (Text- und Grafiklisten)
  - Farbe und Blinken (Gestaltungsanalyse)
    Wenn Sie die jeweilige Option aktivieren, wird jeweils die Auswahl angezeigt, die beim gesetzten Bit mit der kleinsten Wertigkeit projektiert ist:
    Wenn Sie die jeweilige Option deaktivieren und mehrere Bits gesetzt sind, wird die Auswahl angezeigt, die nur beim gesetzten Bit projektiert ist.
- Tooltipps bei E/A-Feldern
  Ob projektierte Grenzwerte bei der Eingabe in E/A-Felder in Runtime als Tooltip angezeigt werden, legen Sie unter "Bilder > Anzeige" fest.
- Farbige Objekte
  Unter "Allgemein > Bilder > Farbtiefe" legen Sie die Farbtiefe Ihres Bediengeräts fest. Bildobjekte gestalten Sie bei Bedarf farbig. Die Anzahl der möglichen Farben hängt davon ab, wie viele Farben das eingestellte Bediengerät unterstützt.

Einstellung zum Design in der Windows-Systemsteuerung

Wählen Sie in den Systemeinstellungen von Windows für den Projektierungs-PC und den Runtime-PC das Design "Windows 7". Mit dieser Einstellung stellen Sie die korrekte Darstellung der Dialoge und der Bildschirmtastatur in Runtime sicher.

15.3.1.3 Bedienung in Runtime projektieren

Einleitung

In den Runtime-Einstellungen legen Sie fest, wie Bilder bedient werden und Sie in Runtime mit Tasten bedienen.

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen". Die Auswahl an Optionen ist abhängig vom Bediengerät.

Taskumschaltung sperren

Abhängig vom Bediengerät, haben Sie die Möglichkeit, die Taskumschaltung am Bediengerät zu sperren. Damit verhindern Sie, dass der Bediener in Runtime andere Anwendungen aufruft.

Aktivieren Sie in den Runtime-Einstellungen des Bediengeräts unter "Allgemein > Bild" die Option "Taskumschaltung sperren".

Um die Option "Taskumstellung sperren" auf einem Runtime PC zu verwenden, deaktivieren Sie das Aero-Design in Windows 7 und Windows 8.x. Um Aero-Design zu deaktivieren, klicken Sie mit der rechten Maustaste auf dem Desktop und wählen Sie "Anpassen". Wählen Sie im Menü „Darstellung und Anpassung" die Designs "Windows-Basis" oder "Windows - klassisch".

Hinweis

Runtime beenden

Wenn Sie die Programmumschaltung sperren, projektieren Sie in Ihrem Projekt unbedingt die Systemfunktion "StoppeRuntime" an ein Objekt, z. B. an eine Funktionstaste oder eine Schaltfläche.

Hinweis

Die Option "Taskumschaltung sperren" wirkt auch auf die Runtime Advanced-Bildschirmtastatur, aber nicht auf die Windows-Bildschirmtastatur.

Windows-Bildschirmtastatur deaktivieren

Um die Taskumschaltung für die Windows-Bildschirmtastatur zu sperren, müssen Sie die Windows-Bildschirmtastatur komplett deaktivieren.

Um die Windows-Bildschirmtastatur zu deaktivieren, gehen Sie folgenderweise vor:

1. Rufen Sie die Bildschirmtastatur auf.
2. Rufen Sie über "Extras > Optionen" den Dialog "Optionen" auf.
3. In der Registerkarte "Initialisierung" deaktivieren Sie die Optionen "Bei Fingereingabe, Symbol neben dem Textfeld anzeigen", "Symbol in der Taskleiste anzeigen", "Registerkarte "Eingabebereich" verwenden".

Die Runtime Advanced-Bildschirmtastatur ist von diesen Änderungen nicht betroffen und kann weiterhin aufgerufen und bedient werden.

Hinweis

Um die Windows-Bildschirmtastatur wieder zu aktivieren, rufen Sie "Start > Programme > Zubehör > Eingabehilfen > Bildschirmtastatur" auf.

Einstellungen zum Datentransfer auf das Bediengerät

- Namen der Bildobjekte in Skripten transferieren
  Wenn Sie unter "Allgemein > Bild" die Option "Namen laden" aktivieren, werden anstelle von codierten Adressierungsinformationen die Objektnamen der Bildobjekte transferiert. Die Objektnamen benötigen Sie, wenn Sie die Bildobjekte in einem Skript über den Objektnamen ansprechen wollen. Wenn Sie ein Skript im Debugger testen, wird der Code durch die Darstellung der Objektnamen übersichtlicher.
- Kommentare in Skripten anzeigen
  Unter "Bilder > Anzeige" legen Sie fest, ob die Kommentare in Skripten mit auf das Bediengerät transferiert werden.
  Wenn Sie die Kommentare nicht transferieren, sparen Sie Speicherplatz auf Ihrem Bediengerät.
  Wenn Sie die Kommentare transferieren, ist das Skript beim Testen im Debugger verständlicher.

Mit Funktionstasten bedienen

Um bei Tastengeräten die Funktion einer Funktionstaste zu symbolisieren, projektieren Sie im Display neben der Funktionstaste eine Grafik. Unter "Bilder > Funktionstasten > Benutzerdefinierte Piktogramm-Größe" legen Sie fest, ob Sie eine zum Standard abweichende Grafikgröße verwenden wollen.

Mit Tastatur bedienen

Unter "Tastatur > Allgemein > Bildschirmtastatur verwenden" legen Sie fest, ob auf Ihrem Bediengerät die Bildschirmtastatur zu Verfügung steht.

Unter "Tastatur > Allgemein > Schaltfläche beim Verlassen loslassen" legen Sie fest, ob das Ereignis "Loslassen" getriggert wird, wenn der Benutzer eine Schaltfläche verlässt, ohne sie loszulassen.

Für Tastenbediengeräte legen Sie unter "Tastatur > Deaktiviere Funktionstasten in Dialogen" fest, ob die Funktionstasten für die Dauer des angezeigten Dialogs deaktiviert werden. Große

Dialoge können an Funktionstasten angrenzende oder Funktionstasten erklärende Bereiche oder Grafiken verdecken. Funktionstasten könnten so versehentlich bedient werden.

15.3.1.4 Kompatibilitätsprüfung mit der Steuerung einstellen

Einleitung

Über den Bereichszeiger "Projektkennung" projektieren Sie die Projektkennung zur Überprüfung der Konsistenz des Projektes mit der Steuerung. Der Bereichszeicher liest die Projektversion der Steuerung aus. Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Mit Steuerungen kommunizieren".

Kompatibilitätsprüfung mit der Steuerung einstellen

Sie wählen die Projektkennung in den Runtime-Einstellungen des Bediengeräts unter "Allgemein > Identifizierung > Projektkennung".

Runtime überprüft beim Start, ob die Projektkennung des Projekts mit der Projektversion der Steuerung übereinstimmt. Nur wenn beide Werte übereinstimmen wird Runtime gestartet.

Bei Verbindungen ist ein Bereichszeiger "Projektkennung" angelegt, dessen Adresse Sie in der Steuerung festlegen. Der Wert, den Sie an dieser Adresse hinterlegen, ist dann die Projektkennung.

Die Projektkennung ist nicht auf allen Bediengeräten verfügbar.

15.3.1.5 Uhrzeitbasis einstellen

Zeitbasis für Uhrzeit einstellen

Die Uhrzeit stellen Sie im Control Panel Ihres Bediengeräts ein. Weitere Hinweise dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

Datum und Uhrzeit mit der Steuerung synchronisieren

Weitere Hinweise dazu finden Sie in der Online-Hilfe von WinCC unter "Mit der Steuerung kommunizieren".

15.3.1.6 Drucken in Runtime

Druckfunktionen

Im Online-Betrieb stehen Ihnen die folgenden Druckfunktionen zur Verfügung:

- Hardcopy
  Über ein Bedienobjekt, das die Systemfunktion "DruckeBild" triggert, drucken Sie das aktuell angezeigte Bild aus.
- Meldungen drucken
  Jede aufgetretene Meldung und deren Zustandsänderungen werden auf einem Drucker mitprotokolliert.
- Protokolle drucken
  Protokolle werden im Grafikmodus ausgegeben. Wegen der anfallenden Datenmengen wird der Einsatz eines seriellen Druckers nicht empfohlen.
  Damit das Protokoll richtig ausgedruckt wird, muss der angeschlossene Drucker das Papierformat und Layout des Protokolls unterstützen.

Hinweis

Der Wert einer Variablen im Protokoll wird erst unmittelbar im Moment des Drucks gelesen und ausgegeben. Bei mehrseitigen Protokollen kann zwischen der ersten und letzten Seite eine größere Zeitspanne liegen. Daher kann z. B. dieselbe Variable auf der letzten Seite mit einem anderen Wert ausgegeben werden als auf der ersten Seite.

15.3.1.7 Kennwortschutz in Runtime

Kennwortschutz in Runtime (Advanced)

Übersicht

Der Verbindungsaufbau kann im TIA Portal durch Kennwörter geschützt werden. Die konfigurierten Kennwörter werden verschlüsselt gespeichert. Beim Transfer ins Zielsystem werden die Kennwörter ebenfalls gegen nicht autorisierte Zugriffe geschützt und können nicht gelesen werden. Auf den Bediengeräten in Runtime werden diese Kennwörter sicher entschlüsselt.

Der Kennwortschutz kann für folgende Verbindungen angelegt werden:

- SIMATIC S7-1200
- SIMATIC S7-1500
- SIMATIC S7-1500 Software Controller (WinAC)
- PC Station 2.0 (Stationmanager)
- SIMATIC HMI HTTP

Der Kennwortschutz kann zusätzlich für die folgende Systemfunktion angelegt werden:

- Systemfunktion "SendeEMail"

Hinweis

Beim Wechsel zu einer älteren Version werden die Kennwörter ungültig.

Zertifikat herunterladen

Das Zertifikat zum Entschlüsseln der Kennwörter wird im Zertifikatspeicher des Bediengeräts gespeichert. Eine Systemmeldung informiert Sie darüber, falls das Zertifikat nicht vorhanden ist.

Das Zertifikat stellen Sie wieder her, indem Sie die Konfiguration erneut herunterladen.

Kennwort für den Transport konfigurieren

Einleitung

Um den privaten Kennwortschlüssel beim Transport zu schützen, können Sie ein Kennwort für den Transport unter "Runtime-Einstellungen" konfigurieren.

Kennwort anlegen

1. Öffnen Sie im Projektfenster den Editor "Runtime-Einstellungen > Allgemein".
2. Tragen Sie im Bereich "Transportkennwort" im Textfeld "Kennwort eingeben" das Kennwort ein.
3. Bestätigen Sie das Kennwort im Feld "Kennwort bestätigen".

Hinweis

Wenn kein altes Kennwort existiert, ist das Feld "Altes Kennwort" deaktiviert.

Ergebnis

Sie haben das Kennwort für den Transport konfiguriert. Wenn ein Kennwort für den Transport konfiguriert ist und Sie das Projekt auf das Bediengerät übertragen, erscheint in Runtime einmalig ein Dialog mit der Kennwortabfrage.

Wenn das eingegebene Kennwort korrekt ist, wird das Zertifikat im Zertifikatspeicher des Bediengeräts gespeichert.

Wenn Sie das Kennwort ändern, laden Sie das Projekt mit dem geänderten Kennwort auf das Bediengerät und starten Runtime. Es erscheint keine Kennwortabfrage.

Kennwort für den Transport zurücksetzen

Einleitung

Wenn Sie beispielsweise das Transportkennwort verlegt haben, können Sie ein neues Sicherheitszertifikat erstellen. Dabei setzen Sie das Transportkennwort zurück.

Kennwort zurücksetzen

1. Klicken Sie auf die Schaltfläche "Kennwort zurücksetzen".
   Das Kennwort wird zurückgesetzt. Ein neues Zertifikat ohne Kennwort wird generiert.

   Hinweis

   Andere Kennwörter wie z. B. das Kennwort von der PLC, werden ungültig. Beim nächsten Übersetzen informiert eine Fehlermeldung Sie darüber, dass diese Kennwörter nicht mehr gültig sind.

2. Geben Sie das neue Kennwort ein und bestätigen Sie es im Feld "Kennwort bestätigen".

   Hinweis

   Solange Sie kein neues Kennwort für den Transport eingeben, ist das Zertifikat nicht kennwortgeschützt.

   Nachdem Sie das Projekt auf das Bediengerät geladen haben, erscheint die Kennwortabfrage.

15.3.2 Übersicht zum Übersetzen und Laden von Projekten

Übersicht

Bereits während Sie ein Projekt in WinCC projektieren, wird das Projekt laufend im Hintergrund übersetzt. Dadurch werden die Zeiten für das endgültige Übersetzen verkürzt. Wenn Sie das Übersetzen starten, entsteht eine Datei, die am jeweiligen Bediengerät ablauffähig ist.

Wenn beim Übersetzen Fehler auftreten, unterstützt WinCC Sie dabei, die Fehler zu finden und zu beheben.

Nach der Fehlerbehebung laden Sie das übersetzte Projekt auf die Bediengeräte, auf denen Sie das Projekt einsetzen wollen. Wenn der Projektierungs-PC nicht mit dem Bediengerät verbunden ist, speichern Sie das übersetzte Projekt auf einem Datenträger ihrer Wahl. Das übersetzte Projekt übertragen Sie dann von einem mit dem Bediengerät verbundenen PC auf das Bediengerät.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten S7-Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

Projekt

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Das folgende Bild zeigt den Zusammenhang zwischen WinCC-Projekt und Runtime-Projekt am Beispiel des Vorgangs von "Übersetzen und Laden":

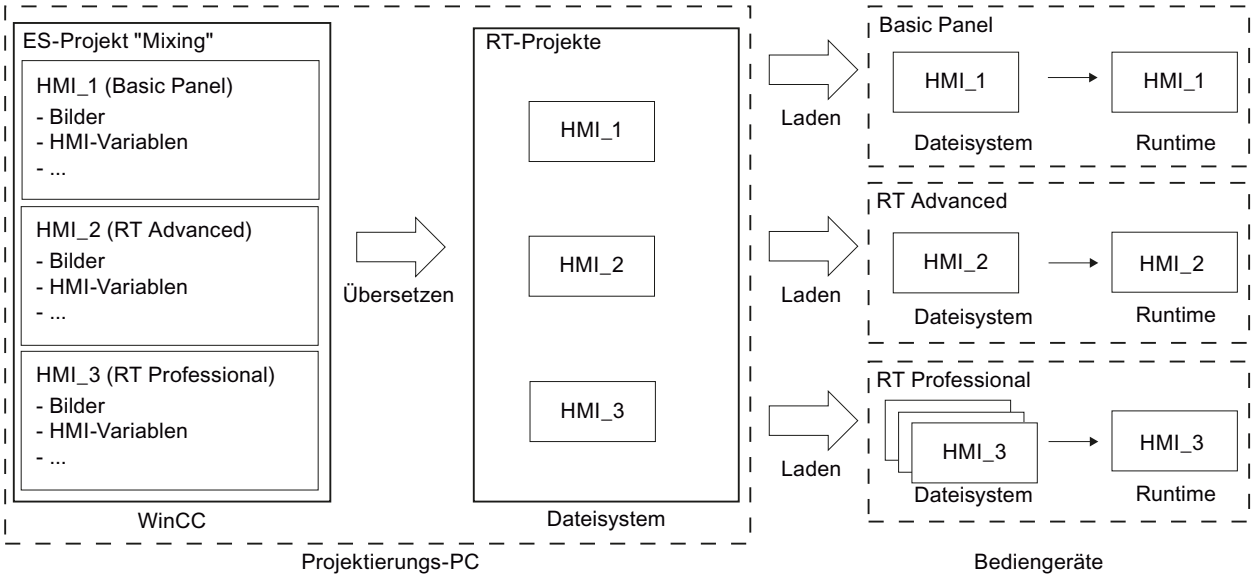

Runtime

Runtime ist die Software zur Prozessvisualisierung. In Runtime führen Sie das Projekt im Prozessbetrieb aus.

Es wird zwischen zwei Arten von Runtime unterschieden:

1. Runtime auf einem Panel
   Bevor Sie ein Runtime-Projekt auf einem Panel ausführen, müssen Sie das Runtime-Projekt vor dem Start auf das Panel übertragen.
2. Runtime auf einem PC
   Wenn Runtime auf dem Projektierungs-PC installiert ist, dann können Sie das Runtime-Projekt direkt auf dem Projektierungs-PC ausführen.
   Wenn Sie das Runtime-Projekt auf einem anderen PC ausführen wollen, müssen Sie das Runtime-Projekt vor dem Start auf den PC übertragen.

Runtime-Version

Die Runtime-Version ist abhängig vom Image des projektierten Bediengeräts. Die Runtime-Version des übersetzten Projekts wird im Inspektorfenster unter "Info" angezeigt.

Simulation

Mit einer Simulation testen Sie ihre Projektierung. Eine Simulation können Sie ohne eine Anbindung an den laufenden Prozess starten.

In einer Simulation testen Sie beispielsweise projektierte Variablen oder Bildwechsel. Während der Simulation können die projektierten Variablen mithilfe des Variablensimulators manipuliert, aktiviert und deaktiviert werden.

Es wird zwischen zwei Simulationsarten unterschieden:

1. Simulation eines Panels
   Wenn Sie ein Panel in ihrem Projekt angelegt haben, wird das Panel in der Simulation angezeigt. Mithilfe dieser Simulationsart können Sie ihre Projektierung am Bediengerät testen, ohne das Projekt auf das Panel zu übertragen.
2. Simulation einer Runtime
   Bei der Simulation einer Runtime testen Sie das Projekt direkt am Projektierungs-PC.

15.3.3 Projekt übersetzen

Einleitung

Bereits während Sie ein Projekt in WinCC projektieren, werden die Änderungen am Projekt laufend im Hintergrund übersetzt. Wenn Sie ein Projekt laden, wird das Projekt automatisch übersetzt. Auf diese Weise wird sichergestellt, dass immer der neueste Stand des Projekts geladen wird.

Während des Übersetzens überprüft WinCC das Projekt auf Konsistenz. Im Inspektorfenster werden alle fehlerhaften Stellen im Projekt aufgelistet. Über den Eintrag im Inspektorfenster springen Sie direkt zur Fehlerursache. Sie überprüfen die festgestellten Fehler und beheben diese.

Umfang des Übersetzens

Sobald Sie mit der Projektierung eines Bediengeräts beginnen, werden die Projektierungsdaten laufend im Hintergrund übersetzt. Wenn Sie ein Projekt manuell übersetzen, werden nur die Änderungen in der Projektierung seit dem letzten Übersetzen im Hintergrund übersetzt.

Um z. B. die Konsistenz der projektierten Daten zu testen, können Sie ein komplettes Übersetzen des Projekts jederzeit manuell starten.

Voraussetzung

- Ein Projekt ist geöffnet.

Vorgehen

Um ein Projekt zu übersetzen, gehen Sie folgendermaßen vor:

1. Wenn Sie mehrere Bediengeräte gleichzeitig übersetzen wollen, wählen Sie alle gewünschten Bediengeräte mit der Mehrfachauswahl in der Projektnavigation aus.
2. Übersetzen Sie das Projekt:
   - Um nur Änderungen im Projekt zu übersetzen, wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software (nur Änderungen)".
   - Um das Projekt komplett zu übersetzen, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (komplett übersetzen)".

Ergebnis

Die Projektierungsdaten aller ausgewählten Bediengeräte werden übersetzt. Wenn beim Übersetzen Fehler auftreten, werden diese im Inspektorfenster angezeigt.

15.3.4 Simulieren von Projekten

15.3.4.1 Grundlagen zur Simulation

Einleitung

Mit dem Simulator testen Sie das Verhalten Ihrer Projektierung am Projektierungs-PC. Auf diese Weise finden Sie frühzeitig und vor dem Produktivbetrieb logische Projektierungsfehler.

Sie können den Simulator auf folgende Arten starten:

- Im Kontextmenü des Bediengeräts oder eines Bildes: "Simulation starten"
- Menübefehl "Online > Simulation > [Starten|Mit Variablen-Simulator|Mit Skript-Debugger]"
- In der Portalansicht unter "Visualisierung > Gerät simulieren"

Voraussetzung

Auf dem Projektierungs-PC ist die Komponente Simulation/Runtime installiert.

Einsatzgebiet

Sie können mit dem Simulator z. B. folgende Funktionen des Bedien- und Beobachtungssystems testen:

- Überprüfung von Grenzwertstufen und Meldeausgaben
- Durchgängigkeit von Alarmen
- Projektierte Alarmsimulationen
- Projektierte Warnungen

- Projektierte Fehlermeldungen
- Kontrolle von Statusanzeigen

15.3.4.2 Simulation von Runtime Advanced

Einleitung

Bei der Simulation für Runtime Advanced wird zwischen folgenden Simulationsarten unterschieden:

- Gerätesimulation
- Variablensimulation

Beide Simuationsarten simulieren das Projekt ohne Prozessanbindung direkt am Projektierungs-PC. Während der Simulation erzeugte Daten wie Archive oder Rezepturen werden nicht gelöscht. Diese Daten werden am Projektierungs-PC in den im Projekt konfigurierten Pfaden gespeichert.

Gerätesimulation

Um die Bedienung des Bediengeräts zu simulieren, verwenden Sie die Gerätesimulation. Mit der Gerätesimulation testen Sie z. B. Bildwechsel.

Variablensimulation

Um die projektierten Prozessvariablen zu simulieren, verwenden Sie die Variablensimulation. Über eine Simulationstabelle lassen Sie Variablenwerte entweder automatisch erzeugen oder Sie geben Variablenwerte vor.

15.3.4.3 Projekt simulieren

Einleitung

Sie simulieren Ihr Projekt auf eine der folgenden zwei Arten:

- Ohne angeschlossene Steuerung
  Die Werte von Bereichszeigern und Variablen verändern Sie in einem Variablen-Simulator, der bei der Simulation von WinCC Runtime gelesen wird.
- Mit angeschlossener Steuerung ohne laufenden Prozess
  Sie simulieren Ihr Projekt, indem Sie es direkt in Runtime laufen lassen. Die Variablen und Bereichzeiger werden aktiv. So erreichen Sie in Runtime eine authentische Simulation Ihres projektierten Bediengeräts.

Hinweis

Einschränkungen bei der Simulation

Folgende Systemfunktionen können nicht simuliert werden:

- KalibriereTouchscreen

Eine Simulation des Media Players ist nicht möglich. Im Simulationsfenster wird anstelle des Media Players ein statisches Bild angezeigt.

Bei Bediengeräten mit Windows CE sind Dateizugriffe über Skripte nicht möglich.

Voraussetzung

- Simulation ohne angeschlossene Steuerung: Variablen sind erstellt
- Simulation mit angeschlossener Steuerung ohne laufenden Prozess: Ein Projekt mit Variablen und Bereichszeigern ist erstellt

Vorgehen

Um ein Projekt über den Variablen-Simulator zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie am Projektierungs-PC das Projekt.
2. Wählen Sie den Menübefehl "Online > Simulation > Mit Variablen-Simulator".
   Wenn Sie das Projekt zum 1. Mal simulieren, wird der Simulator mit einer neuen, leeren Tabelle gestartet. Gleichzeitig wird das Projekt in Runtime geöffnet.
   Zwischen Variablen-Simulator und Runtime schalten Sie mit <Alt+Tab> um.
3. Um einen Prozesswert zu simulieren, wählen Sie im Variablen-Simulator die gewünschte "Variable" aus.
   Die Auswahl enthält alle projektierten Variablen. Sie können maximal 300 Variablen gleichzeitig simulieren.
4. Wählen Sie in der Spalte "Simulation" die gewünschte Simulationsart aus.
5. Verändern Sie in den entsprechenden Spalten die Werte für die Variablen und die Bereichszeiger.

6. Um die Simulation für diese Variable zu starten, aktivieren Sie das Kontrollkästchen "Start".
7. Um die Simulation zu speichern, wählen Sie den Menübefehl "Datei > Speichern" und geben sie einen sprechenden Namen ein, z. B. "Mixing".
   Die Datei erhält die Endung "*.cors".

Ergebnis

Die Prozesswerte werden in Runtime simuliert. Abhängig von der Simulationsart werden die Variablenwerte z. B. zufällig erzeugt oder inkrementiert.

Um Variablenwerte vorzugeben, ändern Sie die Simulationsart auf "<Anzeige>" und geben Sie unter "Wert setzen" den gewünschten Wert ein.

Das folgende Bild zeigt einen Variablen-Simulator mit vier Variablen, deren Werte im Wertebereich von 10 bis 1000 zufällig ermittelt werden:

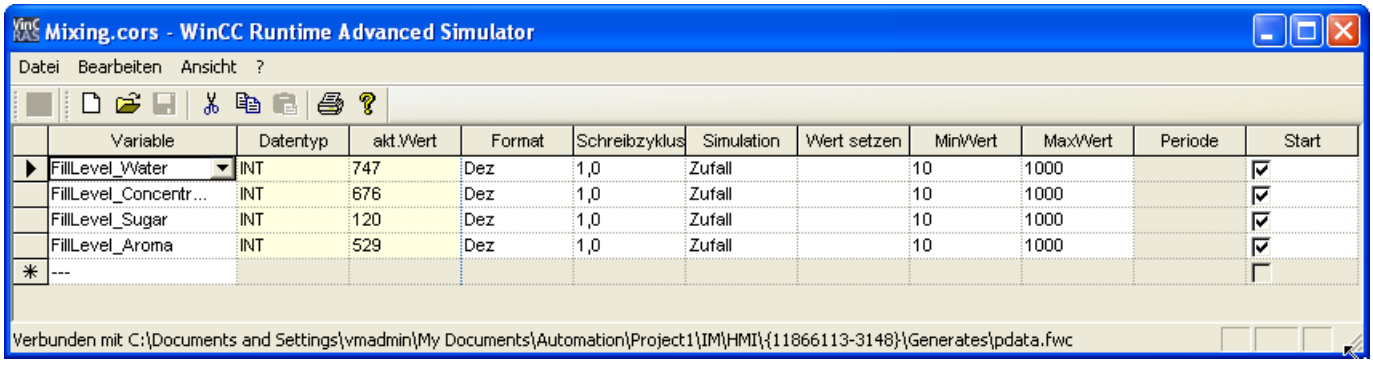

Simulationsdaten verwalten

Wenn Sie die Daten einer früheren Simulation gespeichert haben, können Sie die Datei zu einem späteren Zeitpunkt öffnen und Ihr Projekt erneut simulieren. Voraussetzung dafür ist, dass die im Variablen-Simulator enthaltenen Variablen und Bereichszeiger noch im Projekt enthalten sind.

Um eine Simulationsdatei zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator".
2. Wählen Sie im Variablen-Simulator den Menübefehl "Datei > Öffnen".
3. Wählen Sie die entsprechende Simulationsdatei aus und klicken Sie auf "Öffnen".
   Der Simulator übernimmt die gespeicherten Daten.

Variablen aktivieren und deaktivieren

Damit der Übergang von der Offline-Projektierung zur Online-Projektierung reibungslos möglich ist, starten und stoppen Sie die Simulation jeder Variable einzeln. Aktivieren Sie dazu in der entsprechenden Zeile "Start".

Wenn eine Variable aktiviert ist, werden die Simulationswerte berechnet und an den WinCC-Simulator übergeben.

Variable löschen

Um eine Variable aus dem Variablen-Simulator zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zelle mit dem Variablennamen.
2. Wählen Sie den Menübefehl "Bearbeiten > Ausschneiden".
   Die Variable wird aus der Tabelle entfernt.

15.3.4.4 Bild simulieren

Einleitung

Wenn Sie nur Änderungen in einem Bild gemacht haben, können Sie dieses Bild temporär als Startbild für die Simulation festlegen. Auf diese Weise können Sie eine Änderung testen, ohne das Startbild zu ändern oder am Bediengerät zu dem Bild zu navigieren.

Voraussetzung

Ein Projekt mit mindestens einem Bild ist erstellt.

Vorgehen

Um ein Bild temporär als Startbild für die Simulation festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Bild, das in der Simulation als Startbild angezeigt werden soll.
2. Wählen Sie im Kontextmenü des Bildes den Befehl "Starte Simulation"

Ergebnis

Die Simulation des Projekts wird gestartet. Im Simulationsfenster wird statt des projektierten Startbildes das in der Projektnavigation selektierte Bild angezeigt.

15.3.4.5 Arbeiten mit dem Variablen-Simulator

Erläuterungen zum Variablen-Simulator

Der Variablen-Simulator besteht aus folgenden Spalten:

Spalte	Beschreibung
Variable	Legt die Variablen für die Simulation fest.
Datentyp	Zeigt den Datentyp der ausgewählten Variable.
akt. Wert	Zeigt den simulierten Wert der eingetragenen Variablen.

Spalte	Beschreibung
Format	Legt das gewählte Format fest, in welchem die Variablenwerte simuliert werden: • Dezimal (1, 2, 3, 4, ...) • Hexadezimal (03CE, 01F3, ...) • Binär (0 und 1)
Schreibzyklus	Legt den gewählten Zeitabstand fest, in dem die aktuellen Variablenwerte simuliert werden. Wenn Sie z. B. den Wert "2" eintragen, wird alle 2 Sekunden der aktuelle Variablenwert angezeigt.
Simulation	Zeigt, auf welche Art die Variablenwerte während der Simulation verarbeitet werden.
Wert setzen	Legt den gewählten Wert für die entsprechende Variable fest. Die Simulation startet bei dem eingesetzten Wert.
minWert maxWert	Legt den Wertebereich der Variable fest. Sie legen dazu einen Minimal- und einen Maximalwert an. Standardmäßig liegen die Voreinstellungen bei -32768 für den Minimalwert und 32767 für den Maximalwert.
Periode	Enthält die Periode, in der sich der Variablenwert bei den Simulationsarten "Inkrement" und "Dekrement" wiederholt.
Start	Startet die Simulation der Variable nach den vorangegangenen Eingaben.

Simulationsarten

Der Simulator stellt sechs verschiedene Simulationsarten zur Verfügung. Während der Simulation werden die projektierten Variablen mit realitätsnahen Werten versorgt.

Simulationsart	Beschreibung
Sinus	Verändert den Variablenwert in Form einer Sinuskurve. Die Darstellung erfolgt als periodische, nicht lineare Funktion.
Zufall	Stellt zufällig erzeugte Werte zur Verfügung. Der Variablenwert wird über eine Zufallsfunktion verändert.
Inkrement	Erhöht den Variablenwert kontinuierlich bis zu einem maximal angegebenen Wert. Beginnt nach Erreichen des Maximums wieder beim Minimum. Der Verlauf der Werte entspricht einer positiven Sägezahnkurve.
Dekrement	Verringert den Variablenwert kontinuierlich bis zu einem minimal angegebenen Wert. Beginnt nach Erreichen des Minimums wieder beim Maximum. Der Verlauf der Werte entspricht einer negativen Sägezahnkurve.
Bit schieben	Schiebt ein gesetztes Bit kontinuierlich um eine Position weiter. Die vorherige Position wird jeweils zurückgesetzt. Damit können Sie z. B. die Meldungen für ein Bediengerät testen.
<Anzeige>	Der aktuelle Variablenwert wird statisch angezeigt.

Beispiel: Variablen simulieren mit der Simulationsart "Bit schieben"

Um die Variablen mit der Simulationsart "Bit schieben" zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das zu simulierende Projekt.
2. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator". Der Variablen-Simulator wird geöffnet.
3. Wählen Sie in der Spalte "Variable" eine Variable aus Ihrem Projekt aus.

4. Wählen Sie in der Spalte "Format" "Bin" aus.
5. Geben Sie in der Spalte "Schreibzyklus" den Wert "1" ein.
6. Wählen Sie in der Spalte "Simulation" die Simulationsart "Bit schieben" aus.
7. Geben Sie in der Spalte "Wert setzen" den Wert "1" ein.
8. Aktivieren Sie die Variable mit dem Kontrollkästchen "Start".

Ergebnis

Der Simulator testet die ausgewählte Variable bitweise folgendermaßen:

Simulationswerte	Byte für Meldungen
Gesetzter Anfangswert	00000001
1. Simulationswert	00000010
2. Simulationswert	00000100
3. Simulationswert	00001000
....	...

Sie sehen in Runtime, ob bei dem jeweiligen Wert z. B. die gewünschte Meldung ausgegeben wird.

15.3.4.6 Einschränkungen bei der Simulation

Meldungen mit dynamischen Parametern

Wenn Sie Variablen oder Textlisten als externe Variablen für Meldungen verwenden, dann werden die dynamischen Parameter der Meldungen nicht angezeigt.

Für die Simulation von Meldungen im Variablen-Simulator sind nur interne Variablen freigegeben.

Verwenden Sie PLCSim, um dynamische Parameter zu simulieren.

Comfort Panels

Wenn Sie PLCSim mit der Version V14 oder höher für die Simulation mit einem Comfort Panel verwenden möchten, muss die Geräteversion des Comfort Panels 14.0.0.0 oder höher sein.

15.3.5 Laden von Projekten

15.3.5.1 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum ersten Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Wenn sich das Bediengerät z. B. in einem Subnetz befindet, wählen Sie zusätzlich das Subnetz und das erste Gateway aus.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Wenn die Geräteversion des Ziel-Bediengeräts nicht der projektierten Geräteversion entspricht, wird abgefragt, ob Sie die Geräteversion jetzt ändern wollen.
     Solange das Projekt mit der eingestellten Geräteversion des Ziel-Bediengeräts nicht lauffähig ist, können Sie das Laden nicht starten.

     ACHTUNG

     Das Ändern der Geräteversion löscht alle Daten auf dem Bediengerät

     Wenn Sie die Geräteversion ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb zuerst folgende Daten:
     - Benutzerverwaltung
     - Rezepturen
     - License Keys

     Bei den Comfort-Geräten werden die Lizenzen nur beim Zurücksetzen auf Werkseinstellungen gelöscht. Sichern Sie vor dem Zurücksetzen auf Werkseinstellungen die License Keys.
   - Voreinstellungen, die beim Laden wirksam werden. Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional). Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional). Sie können das Projekt nicht laden. Beheben Sie zunächst die Fehler und laden Sie das Projekt erneut.
     Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Hinweis

Unterbrechung des Ladens

Falls das Laden unterbrochen wird, sichert WinCC automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf ein externes Speichermedium, z. B. einen USB-Stick, und laden es vom USB-Stick auf das Bediengerät.

Diese Funktion ist in Verbindung mit einem Bediengeräte-Image verfügbar, das mit der TIA Portal Version V14 oder höher kompatibel ist.

Sie erzeugen die notwendigen Projektdaten in WinCC, indem Sie das Bediengerät projektieren und anschließend den Ordner des Bediengeräts (z. B. "HMI_1 [<*Gerätetyp*>]") per Drag and Drop auf ein externes Speichermedium unter "Card Reader/USB-Speicher" ziehen.

Laden von Projekten ohne Rezepturdatensätze

In einem Projekt verwenden Sie Rezepturen. Sie übertragen das Projekt auf ein Basic Panel jedoch ohne die Rezepturdatensätze.

Wenn Sie die Struktur der Rezeptur im Engineering System geändert haben und auf dem Gerät bereits Rezepturdatensätze vorhanden waren, dann kann es zu Inkonsistenzen kommen.

Überprüfen Sie in diesem Fall die Konsistenz der Datensätze. Auf dem Gerät wird nicht bei allen strukturellen Änderungen ein Hinweis ausgegeben.

Pack&Go

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf einen PC, der mit dem Bediengerät verbunden ist, z. B. über Netzwerk. Von diesem PC aus laden Sie das Projekt auf das Bediengerät .

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- PN/IE

Übertragen von Runtime-Addons in WinCC

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

Siehe auch

Projekt von externem Speichermedium laden (Seite 2280)

Projekt aus "Pack&Go"-Datei auf Bediengerät laden (Seite 2278)

Projekt auf ein Bediengerät laden (Seite 2378)

"Pack&Go"-Datei erzeugen (Seite 2277)

Runtime Advanced und Panel Runtime starten (Seite 2290)

15.3.5.2 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Verbindungsaufbau

Bitte beachten Sie, dass die im Engineering System projektierte Geräteversion mit der Version der installierten Runtime übereinstimmt, da ansonsten die Kommunikation beim Laden des Projektes nicht oder nur verzögert aufgebaut werden kann.

Wenn die Geräteversion nicht mit der Version der installierten Runtime übereinstimmt, haben Sie folgende Möglichkeiten:

1. Ändern Sie die im Engineering System projektierte Geräteversion.
2. Installieren Sie die zur projektierten Geräteversion passende Version von WinCC Runtime Advanced.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum ersten Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Wenn sich das Bediengerät z. B. in einem Subnetz befindet, wählen Sie zusätzlich das Subnetz und das erste Gateway aus.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Prüfung der Runtime-Version des Ziel-Bediengeräts
     Wenn die Version der auf dem Zielgerät installierten WinCC Runtime Advanced nicht mit der projektierten Geräteversion übereinstimmt, können Sie das Projekt nicht laden.
   - Voreinstellungen, die beim Laden wirksam werden
     Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional)
     Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional)
     Sie können das Projekt nicht laden. Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Hinweis

Unterbrechung des Ladens

Falls das Laden unterbrochen wird, sichert WinCC automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren das übersetzte Projekt auf das Bediengerät.

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- PN/IE

Übertragen von Runtime-Addons in WinCC

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

Siehe auch

Projekt laden (Seite 2273)

15.3.5.3 Projekt laden

Einleitung

Damit ein Projekt auf einem Bediengerät abläuft, laden Sie das Projekt auf das Bediengerät. Beim Laden legen Sie insbesondere fest, ob auf dem Bediengerät vorhandene Daten wie "Benutzerverwaltung" oder "Rezepturdaten" überschrieben werden.

Wenn das Bediengerät PROFINET unterstützt, wird der Name, der in der Projektnavigation eingetragen ist, für die PROFINET-Kommunikation verwendet. Die Verwendung des Namen entspricht der default-Einstellungen der PROFINET-Schnittstelle des Bediengeräts. Bei Geräten mit mehr als einer PROFINET-Schnittstelle wird der Name des IE-CPs automatisch mit Trennpunkt dem Gerätenamen hinzugefügt. Der Name wird beim Laden in das Bediengerät geschrieben. Wenn im Bediengerät bereits ein Gerätename für die PROFINET-Kommunikation eingetragen war, wird dieser überschrieben.

Weitere Informationen zu diesen Einstellungen finden sie im Informationssystem im Kapitel "Gerätename und IP-Adresse zuweisen".

Wenn die Geräteversion des Ziel-Bediengerätes nicht der projektierten Geräteversion entspricht, wird abgefragt, ob Sie die Geräteversion jetzt ändern wollen.

ACHTUNG

Das Ändern der Geräteversion löscht alle Daten auf dem Bediengerät.

Wenn Sie die Geräteversion ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb zuerst folgende Daten:

- Benutzerverwaltung
- Rezepturen
- Lizenzen

Zurücksetzen auf Werkseinstellungen löscht darüber hinaus die License Keys. Sichern Sie vor dem Zurücksetzen auf Werkseinstellungen auch die License Keys.

Bei den Comfort-Geräten werden die Lizenzen nur beim Zurücksetzen auf Werkseinstellungen gelöscht. Sichern Sie vor dem Zurücksetzen auf Werkseinstellungen die Lizenzen und die License Keys.

Hinweis

Transfer auf Bediengerät mit PC Station V1.0 oder PC Station V2.0

Die in den Geräteeinstellungen des Bediengeräts eingestellte Version der PC-Station muss der installierten Version auf dem Bediengerät entsprechen.

Weiterführende Informationen zu Konfigurationseinstellungen für "PC Station" am Bediengerät finden Sie in der Dokumentation zu "SIMATIC NET".

Hinweis

Vorhandene Daten beim Laden überschreiben

Beim Laden des Projekts legen Sie fest, ob die auf dem Bediengerät vorhandene Daten überschrieben werden. Im Dialog "Laden Vorschau" können Sie durch Aktivieren der Optionskästchen folgende Daten immer beim Laden überschreiben:

- Bestehende Daten der Benutzerverwaltung auf dem Gerät
- Bestehende Daten der Rezepturen auf dem Gerät

Verbindungsaufbau mit WinCC Runtime Advanced

Bitte beachten Sie, dass die im Engineering System projektierte Geräteversion mit der Version der installierten Runtime übereinstimmt, da ansonsten die Kommunikation beim Laden des Projektes nicht aufgebaut werden kann oder verzögert erfolgt.

Wenn die Geräteversion nicht mit der Version der installierten Runtime übereinstimmt, haben sie folgende Möglichkeiten:

1. Ändern Sie die im Engineering System projektierte Geräteversion.
2. Installieren sie die zur projektierten Geräteversion passende Version von WinCC Runtime Advanced.

Hinweis

Wenn Ihr Bediengerät ein PC ist, wird die Geräteversion des Zielsystems beim Laden nicht automatisch aktualisiert.

Überprüfen Sie vor dem Übersetzen und Laden Ihres Projekts, ob die projektierte Geräteversion der des Ziel-Bediengerätes entspricht. Installieren Sie bei Bedarf die passende Runtime-Version auf Ihrem Bediengerät oder ändern Sie die Geräteversion manuell über die Eigenschaften des Bediengerätes.

Transferverhalten am Bediengerät steuern

Generell kann auf einem Bediengerät immer nur ein Projekt in Runtime sein. Standardmäßig ist ein Bediengerät so konfiguriert, dass eine laufende Runtime beim Laden automatisch beendet wird. Sonst müssen Sie Runtime am Bediengerät manuell beenden.

Am Bediengerät definieren Sie im "Start Center" unter "Einstellungen", wie sich das Bediengerät beim Laden eines Projekts verhält:

Transfermodus	Auswirkung
Aus	Das Laden eines Projekts auf das Bediengerät ist nicht möglich.
Manuell	Das Laden eines Projekts auf das Bediengerät ist nur möglich, wenn die folgenden Voraussetzungen erfüllt sind: • Runtime läuft nicht • Das Bediengerät ist in der Betriebsart "Transfer".
Automatisch	Das Laden eines Projekts auf das Bediengerät ist immer möglich. Wenn am Projektierungs-PC ein Transfer gestartet wird und am Bediengerät ein Projekt in Runtime ist, wird das laufende Projekt automatisch beendet. Für Mobile Panels ist dieser Transfermodus aus Sicherheitsgründen deaktiviert.

Hinweis

Automatisches Beenden der Runtime

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im "Start Center".

Weiterführende Informationen zu den Transfereinstellungen finden Sie in der Dokumentation des verwendeten Bediengeräts.

Voraussetzung

- Im Projekt ist ein Bediengerät angelegt.
- Das Bediengerät ist mit dem Projektierungs-PC verbunden.

- Am Bediengerät ist das "Start Center" gestartet.
- Am Bediengerät ist im "Start Center" unter "Einstellungen" das Protokoll eingestellt, über welches das Projekt geladen wird.
- Am Bediengerät ist als Transfermodus "Automatisch" oder "Manuell" eingestellt.

Vorgehen

Um ein Projekt zu laden, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Projekt auf mehrere Bediengeräte gleichzeitig laden wollen, wählen Sie in der Projektnavigation alle gewünschten Bediengeräte mit der Mehrfachauswahl aus.
2. Wählen Sie im Kontextmenü eines Bediengeräts den Befehl "Laden in Gerät > Software".
3. Wenn der Dialog "Erweitertes Laden" geöffnet wird, konfigurieren Sie die "Einstellungen für das Laden". Achten Sie dabei darauf, dass die "Einstellungen für das Laden" den "Transfereinstellungen am Bediengerät" entsprechen:
   - Wählen Sie das verwendete Protokoll, z. B. Ethernet oder HTTP.
   - Konfigurieren Sie die entsprechenden Schnittstellenparameter am Projektierungs-PC.
   - Stellen Sie am Bediengerät bei Bedarf schnittstellen- oder protokollspezifische Einstellungen ein.
   - Klicken Sie auf "Laden".

   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet. Gleichzeitig wird das Projekt übersetzt. Das Ergebnis wird im Dialog "Vorschau laden" angezeigt.
4. Überprüfen Sie die angezeigten Voreinstellungen und ändern Sie diese bei Bedarf.
5. Klicken Sie auf "Laden".

Ergebnis

Das Projekt wird auf die ausgewählten Bediengeräte geladen. In WinCC V13 wird das Projekt mit darin enthaltenen Runtime-Addons auf die ausgewählten Bediengeräte geladen.

Während des Ladens können Sie verfolgen, welche Dateien transferiert werden.

Wenn Fehler oder Warnungen beim Laden auftreten, werden im Inspektorfenster unter "Info > Laden" entsprechende Meldungen ausgegeben.

Nach dem erfolgreichen Laden ist das Projekt auf dem Bediengerät ablauffähig.

Hinweis

Falls die Übertragung unterbrochen wird, sichert WinCC V13 automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Siehe auch

Runtime auf dem Projektierungs-PC starten (Seite 2289)

Runtime Advanced und Panel Runtime starten (Seite 2290)

Runtime Professional starten (Seite 2397)

Betriebssystem des Bediengeräts aktualisieren (Seite 2434)

Übersicht zum Laden von Projekten (Seite 2271)

15.3.5.4 "Pack&Go"-Datei erzeugen

Einleitung

Wenn Sie das Bediengerät nicht mit dem Projektierungs-PC verbinden können, erzeugen Sie eine "Pack&Go"-Datei. Die "Pack&Go"-Datei ist eine ZIP-Datei, die Folgendes enthält:

- Das übersetzte Projekt
- Ein Programm zum Übertragen des Projekts auf das Bediengerät
- Image für das projektierte Bediengerät

Typischer Anwendungsfall: Ein Projektierungsbüro erstellt eine Projektvariante für ein neues Bediengerät. Der Projekteur des Projektierungsbüros hat keinen direkten Zugang zur Anlage. Deshalb sendet der Projekteur die "Pack&Go"-Datei per E-Mail an seinen Ansprechpartner vor Ort. Der Ansprechpartner entpackt die "Pack&Go"-Datei auf einem PC, der über Netzwerk mit dem Bediengerät verbunden ist. Danach überträgt er das Projekt vom PC auf das Bediengerät.

Voraussetzung

- Im Projekt ist ein Bediengerät angelegt.

Vorgehen

Um eine "Pack&Go"-Datei zu erzeugen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Bediengerät.
2. Wählen Sie im Menü "Online > Bediengeräte Wartung" den Befehl "Pack&Go".
3. Der Dialog "Pack&Go-Datei erzeugen" wird geöffnet.
4. Prüfen Sie, ob die angezeigte Geräteversion der des Ziel-Bediengerätes entspricht.
5. Wählen Sie als Übertragungsmodus ein Protokoll aus.
6. Wählen Sie unter "Pack&Go-Datei" den Speicherort aus und geben Sie den Dateinamen ein.
7. Wählen Sie bei Bedarf aus, ob die "Pack&Go"-Datei in mehrere Dateien aufgeteilt wird.
8. Klicken Sie auf "Erstellen".

Hinweis

Zum Entpacken einer auf mehrere Dateien aufgeteilten "Pack&Go"-Datei benötigen Sie ein Packprogramm. Das im Betriebssystem integrierte Packprogramm ist nicht ausreichend, um aufgeteilte Dateien zu entpacken.

Ergebnis

Die "Pack&Go"-Datei wird erzeugt und im angegebenen Ordner im Dateisystem abgelegt. Kopieren Sie anschließend die "Pack&Go"-Datei auf den PC, der mit dem Bediengerät verbunden ist.

15.3.5.5 Projekt aus "Pack&Go"-Datei auf Bediengerät laden

Einleitung

Pack&Go ist eine 64 Bit Applikation, und nur auf einem 64 Bit Betriebssystem lauffähig.

Wenn auf Ihrem PC das TIA Portal nicht installiert ist, installieren Sie Microsoft Visual C++ 2015 Redisitributable Package (x64) und .Net Framework 4.6.1 um die Anwendung Pack&Go ausführen zu können.

- Microsoft Visual C++ 2015 Redisitributable Package (x64) finden Sie im Download-Center von Microsoft unter der folgenden URL:
  http://www.microsoft.com/en-us/download/details.aspx?id=48145
- .Net Framework 4.6.1 finden Sie unter der folgenden URL:
  https://www.microsoft.com/en-us/download/details.aspx?id=49982

Voraussetzung

- .NET Framework mit der Version 4.6.1
- Auf dem PC ist ProSave installiert
- Die "Pack&Go"-Datei ist im Dateisystem eines PC abgelegt
- Der PC ist mit dem Bediengerät verbunden
- Geräteversion des Bediengeräts entspricht der Geräteversion im Projekt

Hinweis

Zum Entpacken einer auf mehrere Dateien aufgeteilten "Pack&Go"-Datei benötigen Sie ein Packprogramm. Das im Betriebssystem integrierte Packprogramm ist nicht ausreichend, um aufgeteilte Dateien zu entpacken.

Vorgehen

Um das Projekt aus einer "Pack&Go"-Datei auf ein Bediengerät zu laden, gehen Sie folgendermaßen vor:

1. Entpacken Sie die "Pack&Go"-Datei im Dateisystem des PC in einen beliebigen Ordner.
2. Starten Sie in diesem Ordner im Unterverzeichnis "PackNGo" die Anwendung "Siemens.Simatic.Hmi.PackNgo.exe".
   Der Dialog "Pack'n Go" wird geöffnet. Die Einstellungen für das Laden sind bereits voreingestellt. Ausnahme: Wenn Sie eine S7Usb-Verbindung verwenden, müssen Sie in jedem Fall das Zielgerät wählen.
3. Wenn die Einstellungen für Laden von den am PC verfügbaren Schnittstellen oder Protokollen abweichen, passen Sie die Einstellungen entsprechend an.
4. Legen Sie fest, ob die Daten der Benutzerverwaltung und der Rezepturen, die bereits auf dem Bediengerät gespeichert sind, überschrieben werden sollen.
5. Klicken Sie auf "Transfer".

Ergebnis

Die Verbindung zum Bediengerät wird über die gewählte Verbindung hergestellt. Wenn sich die verwendete Geräteversion im Projekt und die vom Bediengerät unterscheiden, wird eine Meldung ausgegeben.

Aktualisieren Sie in diesem Fall das Betriebssystem auf dem Bediengerät. Um das Betriebssystem auf dem Bediengerät zu aktualisieren, benötigen Sie ProSave. Wenn ProSave installiert ist und die gewählte Verbindung das Aktualisieren des Betriebssystems unterstützt, wird ihnen das Aktualisieren des Betriebssystems automatisch angeboten.

Siehe auch

Übersicht zum Laden von Projekten (Seite 2268)

15.3.5.6 Laden über die USB-Schnittstelle

Einleitung

Wenn Sie ein Projekt über eine USB-Schnittstelle laden, verbinden Sie den Projektierungs-PC und das Bediengerät mit einem USB-Kabel.

Voraussetzungen für den Transfer über USB

Um zu gewährleisten, dass die Datenübertragung über die USB-Schnittstelle erfolgreich verläuft, müssen folgende Bedingungen erfüllt sein:

- Sie verwenden ein USB Host-to-Host Kabel, USB 2.0 Standard.
- Sie haben den bereitgestellten USB-Treiber installiert.
  Sie finden den Treiber auf der WinCC Produkt-DVD unter "Support\DeviceDriver\USB".
- Das verwendete Bediengerät basiert auf Windows CE und besitzt eine USB-Schnittstelle.

Weiterführende Informationen zu den verwendeten Kabeln und den Herstellern/Lieferanten finden Sie im Internet unter:

- http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/de/19142034)

Hinweis

Treiberinstallation

Um Probleme bei der Übertragung der Projekte zu vermeiden, verwenden Sie ausschließlich den bereitgestellten USB-Treiber auf der WinCC Produkt-DVD!

Hinweis

Comfort-Bediengeräte

Bei Comfort-Bediengeräten verwenden Sie zum Laden von Projekten über USB die Mini-USB Schnittstelle. Dieser Treiber wird automatisch installiert.

Einschränkungen

Ein Projekttransfer ist immer für ein Bediengerät möglich. Ein Parallel-Transfer auf mehrere Bediengeräte gleichzeitig ist nicht möglich.

15.3.5.7 Projekt von externem Speichermedium laden

Einleitung

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf ein externes Speichermedium, z. B. einen USB-Stick, eine Memory-Card oder in einen Netzwerkordner und laden es dann auf Ihr Bediengerät.

Diese Funktion ist in Verbindung mit einem Bediengeräte-Image verfügbar, das mit der TIA Portal Version V14 oder höher kompatibel ist.

ACHTUNG

Datenverlust

Wenn Sie die Optionen "Firmware upgrade" oder "Firmware downgrade" aktivieren, dann wird das Betriebssystem des Bediengeräts aktualisiert. Dabei werden auf dem Bediengerät vorhandene Daten inklusive Bediengeräte-Kennwort gelöscht. Einstellungen im Start Center bleiben erhalten, Lizenzschlüssel werden vor dem Aktualisieren des Betriebssystems auf dem externen Speichermedium gesichert.

Sichern Sie bei Bedarf Daten vor dem Laden.

Sie erzeugen die notwendigen Projektdaten in WinCC, indem Sie das Bediengerät projektieren und anschließend den Ordner des Bediengeräts (z. B. "HMI_1 [<*Gerätetyp*>]") per Drag&Drop oder Copy&Paste auf ein externes Speichermedium unter "Card Reader/USB-Speicher" ziehen.

Voraussetzungen

- Sie haben das Start Center am Bediengerät gestartet.
- Das Speichermedium mit dem gesicherten Projekt steckt im Bediengerät.
- Für Runtime Advanced haben Sie die Dateien manuell in ein Verzeichnis kopiert und den Konfigurationspfad unter "Start Center > Settings" angepasst.

Vorgehensweise

1. Klicken Sie auf "Settings".
2. Öffnen Sie den Dialog "Service & Commissioning".
3. Wählen Sie die Registerkarte "Load Project".
4. Betätigen Sie "Next".
   Der Dialog "Load from external memory device" wird geöffnet.
5. Wählen Sie in der Gruppe "Accessible devices" das Speichermedium aus, auf dem die Projektdaten gesichert sind.
6. Betätigen Sie "Next".
   Es wird eine Übersicht aller mit dem Bediengerät kompatiblen Projekte angezeigt, die sich auf dem externen Speichermedium befinden.
7. Wählen Sie das Projekt aus, das Sie in das Bediengerät laden wollen.

   Hinweis

   Über "Details" erhalten Sie weitere Informationen zum gewählten Projekt.

8. Betätigen Sie "Next".
   Das Bediengerät prüft, ob die Projektdaten geladen werden können. Das Ergebnis der Prüfung wird im Dialog "Load Preview" angezeigt.
   Das Projekt kann in das Bediengerät geladen werden, wenn keine Meldungen vom Typ "Fehler" auftreten.

   Hinweis

   Falls ein Downgrade oder ein Upgrade notwendig ist, informiert Sie der Dialog "Update OS Image" über den möglichen Verlust der Daten und gibt Ihnen weitere Anweisungen.

9. Betätigen Sie "Load", um die Projektdaten mit den gewählten Optionen auf das Bediengerät zu übertragen.

Nach dem Ladevorgang wird das neue Projekt auf dem Bediengerät gestartet.

Meldungen im Dialog "Load Preview"

Im Dialog "Load Preview" können folgende Meldungen angezeigt werden:

- Meldungen vom Typ "Information"
- Meldungen vom Typ "Warnung", mit Optionen
- Meldungen vom Typ "Fehler", mit Optionen

Die folgende Tabelle zeigt Meldungen vom Typ "Information":

Symbol	Status	Meldung	Bedeutung
	Info	Firmware version ... Runtime version ...	Firmware und Runtime-Version auf dem Bediengerät
	Info	Ready For Loading	Projektdaten sind für das Bediengerät geeignet.

Die folgende Tabelle zeigt Meldungen vom Typ "Warnung", mit Optionen:

Symbol	Status	Meldung	Bedeutung
	Overwrite	Select project data	Die folgenden Zeilen enthalten Optionen zum Überschreiben der Daten auf dem Bediengerät.
		Recipes	Rezepturen des Bediengeräts mit den Rezepturen des Projekts überschreiben, optional.
		User administration data	Benutzerverwaltung auf dem Bediengerät mit der Benutzerverwaltung des Projekts überschreiben, optional.
	Upgrade	Runtime upgrade	Runtime-Version auf dem Bediengerät ist älter als Runtime-Version des Projekts, Versionen kompatibel, Upgrade der Runtime-Version auf dem Bediengerät ist optional.
	Upgrade	Firmware upgrade	Firmware-Version auf dem Bediengerät ist älter als Firmware-Version des Projekts, Versionen kompatibel, Upgrade der Firmware auf dem Bediengerät ist optional.
	Downgrade	Runtime downgrade	Runtime-Version auf dem Bediengerät ist neuer als Runtime-Version des Projekts, Versionen kompatibel, Downgrade der Runtime-Version auf dem Bediengerät ist optional.
	Downgrade	Firmware downgrade	Firmware-Version auf dem Bediengerät ist neuer als Firmware-Version des Projekts, Versionen kompatibel, Downgrade der Firmware auf dem Bediengerät ist optional.

Die folgende Tabelle zeigt Meldungen vom Typ "Fehler", mit Option:

Symbol	Status	Meldung	Bedeutung
	Upgrade	Runtime upgrade	Runtime-Version auf dem Bediengerät ist älter als Runtime-Version des Projekts, Versionen inkompatibel, Upgrade der Runtime-Version auf dem Bediengerät ist erforderlich.
	Upgrade	Firmware upgrade	Firmware-Version auf dem Bediengerät ist älter als Firmware-Version des Projekts, Versionen inkompatibel, Upgrade der Firmware auf dem Bediengerät ist erforderlich.
	Downgrade	Runtime downgrade	Runtime-Version auf dem Bediengerät ist neuer als Runtime-Version des Projekts, Versionen inkompatibel, Downgrade der Runtime-Version auf dem Bediengerät ist erforderlich.

Symbol	Status	Meldung	Bedeutung
	Downgrade	Firmware downgrade	Firmware-Version auf dem Bediengerät ist neuer als Firmware-Version des Projekts, Versionen inkompatibel, Downgrade der Firmware auf dem Bediengerät ist erforderlich.
	Download	Runtime download	Auf dem Bediengerät befindet sich keine Runtime-Software, z. B. nach der Aktualisierung des Betriebssystems. Download der Runtime-Software ist erforderlich.

Hinweis

Beachten Sie beim Laden von Projekten auf Ihr Bediengerät, dass es kompatible und inkompatible Firmware- und Runtime-Versionen gibt.

Wenn Sie eine kompatible Firmware- und Runtime-Version des Projekts auf Ihr Bediengerät laden, ist ein Upgrade bzw. ein Downgrade optional. Sie können das Projekt auf das Bediengerät ohne ein Upgrade oder ein Downgrade laden. Die Meldung vom Typ "Warnung" können Sie in diesem Fall ignorieren.

Wenn Sie eine inkompatible Firmware- und Runtime-Version des Projekts auf Ihr Bediengerät laden, müssen Sie ein Upgrade oder ein Downgrade ausführen. Anderenfalls können Sie das Projekt nicht auf das Bediengerät laden. Die Meldung vom Typ "Fehler" erscheint.

Hinweis

Die Meldungen vom Typ "Warnung" erscheinen auch bei potenzieller Überschreibung von Benutzerdaten und Rezepturen, wenn diese Einstellung in der Projektierung nicht gesperrt wurde.

Siehe auch

Übersicht zum Laden von Projekten (Seite 2271)

Übersicht zum Laden von Projekten (Seite 2268)

15.3.5.8 USB-Treiber unter Windows 7 installieren

Einleitung

Um ein Projekt über die USB-Schnittstelle zu laden, installieren Sie den auf der WinCC Produkt-DVD bereitgestellten USB-Treiber.

Hinweis

Treiberinstallation

Um Probleme bei der Übertragung der Projekte zu vermeiden, verwenden Sie ausschließlich den bereitgestellten USB-Treiber auf der WinCC Produkt-DVD!

Hinweis

Comfort-Bediengeräte

Bei Comfort-Bediengeräten verwenden Sie zum Laden von Projekten über USB die Mini-USB Schnittstelle. Dieser Treiber wird automatisch installiert.

Voraussetzung

- Die WinCC Produkt-DVD ist vorhanden.
- Sie verwenden ein USB Host-to-Host Kabel, USB-Standard 2.0.
- Das Bediengerät besitzt eine USB-Schnittstelle.

Vorgehen

Um den USB-Treiber unter Windows 7 zu installieren, gehen Sie folgendermaßen vor:

1. Schließen Sie das USB Host-to-Host Kabel an den USB-Port des PC an.
   Das USB-Kabel wird erkannt, aber der Assistent für die Treiberinstallation findet den Treiber nicht.

   Hinweis

   Erst wenn der Treiber auf dem PC fertig installiert ist, schließen Sie das Bediengerät an.

2. Öffnen Sie in der Systemsteuerung den Gerätemanager.
3. Wählen Sie "Andere Geräte > USB Host-to-Host Kabel".
4. Markieren Sie "USB Host-to-Host Kabel" und wählen im Kontextmenü "Treibersoftware aktualisieren...".
5. Wählen Sie bei der Abfrage "Wie möchten Sie nach Treibersoftware suchen?" "Auf dem Computer nach Treibersoftware suchen".
6. Klicken Sie "Treibersoftware aktualisieren > Durchsuchen...".
7. Legen Sie die WinCC Produkt-DVD als Quelle der Treiberinstallation fest.

8. Aktivieren Sie "Unterordner einbeziehen" und klicken Sie auf "Weiter".
   Die WinCC Produkt-DVD wird nach dem passenden Treiber für das USB Host-to-Host Kabel durchsucht.
   Die verfügbaren Treiber auf der WinCC Produkt-DVD werden angezeigt. Der Assistent markiert den passenden Treiber für Windows 7.
9. Kontrollieren Sie die Auswahl und klicken Sie auf "Weiter".
10. Um mit der Installation fortzufahren, klicken Sie "Hardwareinstallation > Installation fortsetzen".
    Der Treiber wird installiert.
11. Klicken Sie auf "Fertig stellen".
    Die Installation wird beendet.

Ergebnis

Der benötigte Treiber für das USB Host-to-Host Kabel wurde installiert. Um das Projekt zu laden, schließen Sie jetzt das Bediengerät an das USB-Kabel an.

15.3.6 Übersetzen und Laden mit Multiuser Engineering

15.3.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)

Einleitung

Beim Einsatz von Multiuser Engineering für die Bearbeitung Ihrer Projekte sollten Sie das Verhalten beim Übersetzen der Runtime-Projekte und das Verhalten beim Laden auf die Bediengeräte berücksichtigen.

Das Übersetzen und Laden auf ein Bediengerät ist sowohl in der Serverprojekt-Ansicht als auch in der lokalen Session möglich.

Weitere Informationen zum Thema "Multiuser Engineering" finden Sie unter "Multiuser Engineering einsetzen".

Grundlagen

Für Bediengeräte und RT Advanced sind im Multiuser Engineering folgende Szenarien möglich:

- Übersetzen in der Serverprojekt-Ansicht
- Übersetzen in der lokalen Session
- Laden aus der Serverprojekt-Ansicht
- Laden aus der lokalen Session

Hinweis

Das Komplettladen aus der Serverprojekt-Ansicht oder aus der lokalen Session unterscheidet sich nicht vom Komplettladen in einem Single-User-Projekt. Beim Komplettladen wird das aktuelle Runtime-Projekt aus der geraden aktiven Ansicht auf ein Bediengerät geladen.

Hinweis

Das Übersetzen und Laden von Projekten in einer lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Grundsätzlich können Sie in Multiuser Engineering-Projekten alle Befehle zum Übersetzen und Laden ausführen:

- "Software (komplett übersetzen)"
- "Übersetzen > Software (nur Änderungen)"
- "Software (komplett laden)"

Der Begriff "Projekt"

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Regeln

Folgende grundsätzliche Regeln gelten für Übersetzen und Laden in Multiuser Engineering:

- Das Runtime-Projekt, das in einer lokalen Session übersetzt wurde, bleibt immer lokal und wird nicht auf den Multiuser-Server hochgeladen. Es kann im Multiuser-Serverprojekt nicht gespeichert werden.
- Nur die Runtime-Projekte, die in der Serverprojekt-Ansicht übersetzt wurden, können im Multiuser-Serverprojekt gespeichert werden.

Weitere Informationen zu Multiuser Engineering finden Sie im Siemens YouTube Channel: Multiuser Engineering -ein Team arbeitet gleichzeitig an einem Projekt (https://www.youtube.com/watch?v=6Kq-4YWHJq4).

Siehe auch

Übersetzen in der Serverprojekt-Ansicht (Seite 2287)

Übersetzen in der lokalen Session (Seite 2288)

15.3.6.2 Übersetzen in der Serverprojekt-Ansicht

Grundlagen

Das Übersetzen und Laden von Projekten in der Serverprojekt-Ansicht unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Während Sie ein Projekt in der Serverprojekt-Ansicht übersetzen, wird das Multiuser-Serverprojekt blockiert. Andere Benutzer können in dieser Zeit keine Änderungen an diesem Serverprojekt durchführen. Das in der Serverprojekt-Ansicht übersetzte Runtime-Projekt wird zusammen mit dem Engineering Projekt im zentralen Multiuser-Server gespeichert. Durch das Blockieren des Multiuser-Serverprojekts ist gewährleistet, dass die Projektierungsdaten und das Runtime-Projekt synchron bleiben.

Hinweis

Wenn Sie in der Serverprojekt-Ansicht übersetzen und speichern, erhalten andere Benutzer beim "Aktualisieren" ihrer lokalen Session das von Ihnen aktualisierte Runtime-Projekt zusammen mit dem Engineering Projekt. Andere Benutzer müssen dann nach einer Aktualisierung nicht erneut die von Ihnen eingebrachten Änderungen übersetzen.

Beispiel: Übersetzen während des Eincheckens

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Wenn keine Übersetzungsfehler vorliegen, werden über "Änderungen speichern" beide Projekte, das modifizierte Engineering Projekt (mit den geänderten Variablen) und das übersetzte Runtime-Projekt, im zentralen Multiuser-Serverprojekt gespeichert.

Wenn Sie das Übersetzen während des Eincheckens auslassen, enthält das Projekt die Änderungen, die auf dem Server gespeichert wurden.

Der nächste Benutzer, der aus dem Serverprojekt eine lokale Session erstellt oder eine vorhandene lokale Session aktualisiert, muss zusätzlich zu seinen Änderungen noch Ihre zwei Änderungen übersetzen.

Hinweis

Das über mehrere lokale Sessions verteilte Arbeiten an einem gemeinsamen Projekt erhöht die Fehlerwahrscheinlichkeit. Daher ist es empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2285)

Übersetzen in der lokalen Session (Seite 2288)

15.3.6.3 Übersetzen in der lokalen Session

Grundlagen

Das Übersetzen und Laden von Projekten in der lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Da die lokale Session eine Kopie des Serverprojekts ist, ist der erste Übersetzungsstatus der lokalen Session mit dem des Serverprojekts identisch. Wenn im Serverprojekt unübersetzte Inhalte oder Fehlermeldungen beim Übersetzen vorhanden waren, werden sie in die lokale Session übertragen.

Hinweis

Es ist empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer und vermeiden Fehlerverbreitung.

Aktualisieren in der lokalen Session

Wenn Sie ein Projekt in der lokalen Session aktualisieren, wird die lokale Session komplett - inklusive des Übersetzungsstatus - durch den Inhalt des Serverprojekts ersetzt. Nur die für das Einchecken markieren Änderungen werden in der aktualisierten lokalen Session beibehalten und erzeugen zusätzliche Übersetzungsschritte in der lokalen Session.

Beispiel: Aktualisieren der lokalen Session

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Sie aktualisieren die Inhalte der lokalen Session, indem Sie auf die Schaltfläche "Aktualisieren" klicken. Nach dem Aktualisieren bekommt die lokale Session den Übersetzungsstatus des Serverprojekts. Zusätzlich sind Übersetzungsaufgaben vorhanden, die durch die Übernahme der geänderten Variablen begründet sind.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2285)

Übersetzen in der Serverprojekt-Ansicht (Seite 2287)

15.3.7 Runtime starten

15.3.7.1 Runtime auf dem Projektierungs-PC starten

Einleitung

Wenn auf dem Projektierungs-PC Runtime installiert ist, können Sie auf dem Projektierungs-PC ein Projekt in Runtime starten. Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben.

Hinweis

Bediengeräte mit Runtime Panels können Sie auf dem Projektierungs-PC nur simulieren. Wählen Sie in diesem den Menübefehl "Online > Simulation starten".

Hinweis

Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Start Center.

Voraussetzung

- Auf dem Projektierungs-PC ist ein Projekt geöffnet.
- Auf dem Projektierungs-PC ist Runtime installiert.
- Auf dem Projektierungs-PC ist kein Projekt in Runtime.

Vorgehen

Um die Runtime am Projektierungs-PC zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie das gewünschte Bediengerät in der Projektnavigation aus.
2. Wählen Sie den Menübefehl "Online > Runtime starten".

Ergebnis

Auf dem Projektierungs-PC wird Runtime gestartet und das Projekt angezeigt.

15.3.7.2 Runtime Advanced und Panel Runtime starten

Einleitung

Sobald Sie das Projekt auf das Bediengerät geladen haben, können Sie das Projekt in Runtime starten. Standardmäßig wird das Projekt am Bediengerät automatisch gestartet.

Bei Bediengeräten mit Runtime Advanced können im Dateisystem des Bediengeräts mehrere Projekte liegen. Sie können eines dieser Projekte in Runtime starten.

Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben. Achten Sie bei den Runtime-Einstellungen "Taskumschaltung sperren" und "Vollbild" darauf, dass Sie Runtime wieder beenden können. Projektieren Sie dazu z. B. eine Schaltfläche mit der Systemfunktion "StoppeRuntime".

Hinweis

Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im "Start Center".

Hinweis

Verschlüsselte Kommunikationsverbindungen in Runtime nutzen

Ein Runtime-Projekt mit verschlüsselten Kommunikationsverbindungen zu S7-Steuerungen muss für jeden Windows-Benutzer geladen werden, der in Runtime diese Verbindungen nutzen möchte.

Voraussetzung

- Auf dem Bediengerät ist WinCC Runtime Advanced oder die Panel Runtime installiert.
- Das Projekt ist auf das Bediengerät geladen.
- Das "Start Center" ist gestartet

Runtime auf einem PC starten

Auf einem PC wird das übersetzte Projekt frei zugänglich im Dateisystem mit der Endung "*.fwc" abgelegt.

1. Tragen Sie im "Start Center" unter "Einstellungen > Konfigurationspfad" die Projektdatei mit vollständiger Pfadangabe ein.
2. Um das Projekt nach Booten des Bediengeräts automatisch zu starten:
   – Wählen Sie unter "Warten bis Autostart" die gewünschte Verzögerungszeit in Sekunden.
   – Fügen Sie die Anwendung "Start Center" im Windows-Startmenü der Gruppe "Autostart" hinzu.
3. Um das Projekt am Bediengerät manuell zu starten, klicken Sie im "Start Center" auf "Start".

Runtime auf einem Panel starten

Auf einem Panel wird das Projekt in einem Ordner abgelegt, den Sie in den Transfereinstellungen des Bediengeräts angeben. Auf einem Panel wird die Anwendung "Start Center" gestartet. Abhängig von der Projektierung startet nach einer Verzögerungszeit das geladene Projekt automatisch.

Wenn das Projekt nicht automatisch startet:

1. Um das geladene Projekt zu starten, klicken Sie im "Start Center" auf "Start".

Weiterführende Informationen zum Starten von Projekten finden Sie in der Dokumentation des Bediengeräts.

Ergebnis

Am Bediengerät wird die Runtime gestartet.

15.3.8 Fehlermeldungen beim Laden von Projekten

Mögliche Probleme beim Laden

Während ein Projekt auf ein Bediengerät geladen wird, werden im Ausgabefenster Statusmeldungen über den Fortschritt des Ladevorgangs ausgegeben.

Wenn beim Laden des Projekts auf das Bediengerät Probleme auftreten, ist die Ursache in den meisten Fällen eine der Folgenden:

- Falsche Version des Betriebssystems auf Bediengerät
- Falsche Einstellungen für Laden auf dem Bediengerät
- Falscher Bediengerättyp im Projekt
- Bediengerät ist nicht mit Projektierungs-PC verbunden.

Nachfolgend sind die häufigsten Fälle für einen Abbruch des Ladens mit möglicher Ursache und Abhilfe aufgeführt.

Das serielle Laden bricht ab

Mögliche Abhilfe: Wählen Sie eine kleinere Baudrate.

Das Laden wird wegen eines Kompatibilitätskonflikts abgebrochen

Mögliche Ursache	Abhilfe
Konflikt zwischen den Versionen der verwendeten Projektierungs-Software und dem Betriebssystem des Bediengeräts	Synchronisieren Sie das Betriebssystem des Bediengeräts mit der verwendeten Version der Projektierungs-Software. Um das Betriebssystem des Bediengeräts zu aktualisieren, wählen Sie in WinCC im Menü "Online > Bediengeräte-Wartung" den Befehl "Betriebssystem aktualisieren". Alternativ können Sie auch ProSave verwenden. Weitere Informationen erhalten Sie in der Betriebsanleitung des Bediengeräts.
Der Projektierungs-PC ist mit einem falschen Gerät verbunden, z. B. einer Steuerung	Kontrollieren Sie die Verkabelung. Korrigieren Sie die Kommunikationsparameter

Projekt wird nicht geladen

Mögliche Ursache	Abhilfe
Verbindung zum Bediengerät kann nicht aufgebaut werden (Meldung im Ausgabefenster)	Prüfen Sie die physikalische Verbindung zwischen Projektierungs-PC und Bediengerät. Prüfen Sie, ob sich das Bediengerät im Transfermodus befindet. Ausnahme: Fernsteuerung
Im Gerätemanager von Windows ist der Standard-Kommunikationstreiber nicht eingetragen	Kontrollieren Sie im Eigenschaftsfenster des Gerätemanagers den Gerätestatus des COM-Anschlusses.

Laden über die MPI/DP-Schnittstelle funktioniert nicht

Mögliche Ursache	Abhilfe
Am CP ist der "projektierte Betrieb" eingestellt, z. B. bei Verwendung der SIMATIC NET CD.	Stellen Sie mit der Anwendung "PC-Station einstellen" für den CP den "PG-Betrieb" ein. Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse". Laden Sie das Projekt aus WinCC auf den CP. Stellen Sie für den CP wieder den "projektierten Betrieb" ein.
Im PG/PC-Panel steht der Zugangspunkt "S7ONLINE" nicht auf einem physikalischen Gerät wie CP5611 (MPI). Ursache kann die Installation von "SIMATIC NET CD" sein.	Stellen Sie mit der Anwendung "PG/PC-Panel" oder "PC-Station einstellen" den Zugangspunkt "S7ONLINE" auf ihr gewünschtes Gerät ein. Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse". Laden Sie das Projekt aus WinCC auf das Bediengerät. Stellen Sie den Zugangspunkt "S7ONLINE" wieder zurück auf das ursprüngliche Gerät.

Komplexität der Projektierung ist zu groß

Mögliche Ursache	Abhilfe
Ihre Projektierung enthält zu viele verschiedene Objekte oder Optionen für das ausgewählte Bediengerät.	Entfernen Sie alle Objekte eines Typs, z. B. alle HTML-Browser. Entfernen Sie alternativ Optionen wie Sm@rtServer oder OPC-Server.

15.3.9 Anpassen des Projekts für ein anderes Bediengerät

Einleitung

Wenn Sie ein WinCC Projekt auf ein Bediengerät laden, prüft WinCC, ob das Bediengerät mit dem im Projekt verwendeten Bediengerätetyp übereinstimmt. Wenn die Bediengerätetypen nicht übereinstimmen, erhalten Sie beim Ladevorgang eine Meldung.

Der Ladevorgang wird abgebrochen.

Anpassen des Projekts für das Bediengerät

Um das Projekt auf das angeschlossene Bediengerät zu laden, müssen Sie das Projekt entsprechend anpassen.

- Fügen Sie in der Projektnavigation ein neues Bediengerät ein. Wählen Sie in der Bediengeräteauswahl den richtigen Bediengerätetyp aus.
- Kopieren Sie die projektierten Bestandteile aus dem bisherigen Bediengerät in das neu angelegte Bediengerät.
  Viele Bestandteile kopieren Sie direkt in der Projektnavigation und der Detailansicht. Kopieren Sie z. B. den kompletten Ordner "Bilder" mithilfe des Kontextmenüs in den Ordner Bilder des neuen Bediengeräts.
- Bei einigen Einträgen in der Projektnavigation, für die im Kontextmenü der Befehl "Kopieren" nicht verfügbar ist, kopieren Sie die Inhalte über die Detailansicht.
- Selektieren Sie z. B. den Eintrag "Rezepturen" in der Projektnavigation. Die Rezepturen werden in der Detailansicht angezeigt.
- Selektieren Sie die Rezepturen in der Detailansicht und ziehen Sie diese per Drag&Drop auf den Eintrag "Rezepturen" des neuen Bediengeräts. Die Rezepturen werden kopiert. Eine Mehrfachselektion in der Detailansicht ist möglich.
- Projektieren Sie nicht kopierbare Bestandteile neu, z. B. Verbindungen, Bereichszeiger und Meldungen.
- Speichern Sie das Projekt auch zwischendurch.
- Übersetzen Sie das Projekt komplett.
- Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

Verknüpfung von Referenzen

Wenn Sie Objekte kopieren, werden die Referenzen zu verbundenen Objekten mitkopiert. Nach dem Kopieren der verbundenen Objekte werden die Referenzen wieder miteinander verknüpft.

Beispiel:

Sie kopieren ein Bild, in dem Objekte mit Variablen verknüpft sind. Nach dem Einfügen des Bilds in das neue Bediengerät sind die Variablennamen an den einzelnen Objekten eingetragen. Die Variablennamen sind rot markiert, da die Referenzen offen sind. Wenn Sie anschließend die Variablen kopieren und in das neue Bediengerät einfügen, werden die offenen Referenzen geschlossen. Die rote Markierung der Variablennamen wird aufgehoben.

Damit Referenzen zu verbundenen Objekten in der Steuerung vervollständigt werden, müssen Sie zuerst eine Verbindung zu der Steuerung projektieren.

Verwendung des Infobereichs

Wenn Sie das Projekt für das neue Bediengerät übersetzen, werden im Inspektorfenster auf der Registerkarte "Info" Fehler und Warnungen ausgegeben. Über den Kontextmenübefehl "Gehe zu" gelangen Sie direkt zur betreffenden Stelle, um den Fehler oder die Warnung zu korrigieren.

Arbeiten Sie die Liste der Fehler und Warnungen nacheinander ab.

Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

Siehe auch

Projekt laden (Seite 2273)

15.3.10 Projektgröße reduzieren

Einleitung

Beim Laden eines umfangreichen Projekts auf ein HMI-Bediengerät kann es vorkommen, dass der Speicherplatz des Bediengeräts für das Projekt nicht ausreicht. Sie haben mehrere Möglichkeiten, die Größe Ihres Projekts zu verringern.

Möglichkeiten zur Reduktion der Projektgröße

Sie haben mehrere Möglichkeiten, die Projektgröße zu reduzieren und Speicherplatz zu sparen:

- Anzahl der verfügbaren Runtime-Sprachen verringern
  Überprüfen Sie, ob alle selektierten Runtime-Sprachen tatsächlich benötigt werden. Deaktivieren Sie ggf. unter "Runtime Einstellungen > Sprache & Schriftart > Runtime-Sprache und Schriftauswahl" die Sprachen, die Sie nicht benötigen.
- Auf die Hilfetexte der S7 Diagnosemeldungen verzichten
  Um die Größe des Projekts zu verkleinern, können Sie das Herunterladen der Hilfetexte für die S7 Diagnosemeldungen deaktivieren. Um die Hilfetexte nicht auf das Bediengerät zu laden, deaktivieren Sie unter "Runtime-Einstellungen > Meldungen > Allgemein" die Option "S7 Diagnosehilfetexte herunterladen".
- Software komplett übersetzen
  Um die Projektdaten zu optimieren und veraltete Änderungen zu bereinigen, übersetzen Sie Ihr Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.
- Darstellung mithilfe von Stilen harmonisieren (ab WinCC V13)
  Es wird empfohlen, Bildobjekte mithilfe von Stilen zu harmonisieren. Vereinheitlichen Sie das Aussehen von Bildobjekten in Ihrem Projekt, um Projektdaten zu optimieren. Nutzen Sie durchgängig einen vorgegebenen oder einen benutzerdefinierten Stil bei der Projektierung der Bildobjekte.
- Automatische Aktualisierung von PLC Meldungen aktivieren (für S7 1500 Steuerungen und V14 HMI-Bediengeräte)
  Um den Speicherplatz zu sparen, legen Sie fest, dass die PLC-Meldetexte nur bei Bedarf in Runtime geladen werden. Aktivieren Sie dafür unter "Runtime Einstellungen > Meldungen > Steuerungsmeldungen" die Option "Automatische Aktualisierung". Stellen Sie sicher, dass in der entsprechenden Steuerung automatische Aktualisierung von Meldungen auch aktiviert ist.
  Die Option "Automatische Aktualisierung" steht bei Basic Panels nicht zur Verfügung.
  Eingesparter Speicherplatz hängt von der Anzahl der PLC-Meldungen und der Anzahl der Runtime-Sprachen ab.

- Anzahl der geladenen Schriftarten verringern
  Überprüfen Sie, ob die Anzahl der benutzerdefinierten heruntergeladenen Schriftarten verkleinert werden kann. Legen Sie ggf. unter "Runtime Einstellungen > Sprache & Schriftart > Runtime-Sprache und Schriftauswahl" für die benötigten Runtime-Sprachen nur die Standard-Schriftarten fest.
  Um Speicherplatz zu sparen, nutzen Sie bei der Projektierung weniger unterschiedlichen Schriftfamilien.
- Größe der Grafiken verringern
  Überprüfen Sie die Größe der Grafiken, die Sie im Projekt verwenden. Reduzieren Sie ggf. die Größe der Grafiken, indem Sie die Auflösung verringern oder ein stärkeres Komprimierungsformat ohne merkbaren Qualitätsverlust für die Projektgrafiken wählen.
  Um bei den Basic-Bediengeräten die Projektgröße kleiner zu halten, harmonisieren Sie die Größen, in welchen die Projektgrafiken verwendet werden.
  Wählen Sie passende Grafikformate für Ihre Bilder: nutzen Sie PNG-Grafiken für Zeichnungen, die keine Vektorgrafiken sind und JPEG-Grafiken für Fotos.

15.3.11 Unterstützte Netzwerkkameras und Einstellungen

Empfohlene Kameras und Einstellungen

Im Objekt "Kamera-Anzeige" geben Sie Bilder einer angeschlossenen Netzwerkkamera auf Ihrem Bediengerät aus. Verwenden Sie das Objekt "Kamera-Anzeige" mit den Netzwerkkameras, die das Streaming-Protokoll RTP/RTPS und Codecs MJPEG, H.264 und MPEG4 unterstützen.

Je nach Kamera und Bediengerät unterscheiden sich die Einstellungen, die die optimale Bildqualität liefern.

Hinweis

Geräteabhängigkeit des Objekts "Kamera-Anzeige"

Das Objekt "Kamera-Anzeige" ist für Comfort Panels, KTP Mobile Panels (ab Geräteversion V13) und RT Advanced verfügbar.

Folgende Tabellen zeigen Ihnen die Einstellungen für die getesteten Kameras und die daraus resultierenden Empfehlungen für die optimale Darstellung im Objekt "Kamera-Anzeige":

Siemens CCMS2025

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	MJPEG	MJPEG	MPEG-4	H.264	H.264	MJPEG
Auflösung Kamera-Anzeige	320x240	320x240	640x480	640x480	320x240	800x600
Auflösung Kamera	320x240	320x240	640x480	640x480	320x240	800x600 SVGA
Bildfrequenz (fps)	10	10	10	10	10	nicht relevant

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Bildverzögerung (Sekunden)	<1	<1	<1	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	geeignet	geeignet	geeignet	geeignet	geeignet	geeignet

AXIS M1011

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	MJPEG	H.264/MJPEG	H.264	MJPEG	MJPEG	MJPEG
Auflösung Kamera-Anzeige	320x240	320x240	640x480	640x480	480x360	640x480
Auflösung Kamera	320x240	320x240	640x480	640x480	480x360	640x480
Bildfrequenz (fps)	10	10	10	10	10	nicht relevant
Bildverzögerung (Sekunden)	>1	<1	<1	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	mit Einschränkungen geeignet	geeignet	geeignet	geeignet	geeignet	geeignet

AXIS M1013

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	MJPEG	MJPEG	MJPEG	MJPEG	MJPEG	MJPEG
Auflösung Kamera-Anzeige	320x240	320x240	800x600	800x600	480x360	800x600
Auflösung Kamera	320x240	320x240	800x600	800x600	480x360	800x600
Bildfrequenz (fps)	10	10	10	10	10	nicht relevant
Bildverzögerung (Sekunden)	<1	<1	>1	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	geegnet	geeignet	mit Einschränkungen geeignet	geeignet	geeignet	geeignet

D-Link DCS-942L

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	MJPEG	MJPEG	MJPEG	MJPEG	MPEG4	MPEG4
Auflösung Kamera-Anzeige	320x240	320x240	320x240	640x480	320x240	640x480
Auflösung Kamera	320x240	320x240	320x240	640x480	320x240	640x480
Bildfrequenz (fps)	10	10	10	10	10	nicht relevant
Bildverzögerung (Sekunden)	<1	<1	<1	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	geeignet	geeignet	nicht geeignet	geeignet	geeignet	geeignet

D-Link 4701E

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"		Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	H.264	MJPEG	H.264/MJPEG	H.264	MJPEG	MJPEG	MJPEG
Auflösung Kamera-Anzeige	320x240	320x240	320x240	640x480	640x480	320x240	960x720
Auflösung Kamera	320x240	320x240	320x240	640x480	640x480	320x240	960x720
Bildfrequenz (fps)	25	25	25	7	25	25	nicht relevant
Bildverzögerung (Sekunden)	<1	<1	<1	<2	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	geeignet	geeignet	geeignet	geeignet	geeignet	geeignet	geeignet

D-Link DCS-4201

	Comfort Panels 15" - 22"	RT Advanced
Codec	MJPEG	MJPEG
Auflösung Kamera-Anzeige	640x480	960x720
Auflösung Kamera	640x480	960x720
Bildverzögerung (Sekunden)	<1	<1
Qualität	Standard	Standard
Empfehlung mit Panel	geeignet	geeignet

TL-SC2020

	RT Advanced
Codec	MJPEG
Auflösung Kamera-Anzeige	640x480
Auflösung Kamera	640x480
Bildverzögerung (Sekunden)	<1
Qualität	Standard
Empfehlung mit Panel	geeignet

VANDERBILT CCMW3025

	Comfort Panels 4"	Mobile F Panels 4"	Comfort Panels 7" - 12"	Comfort Panels 15" - 22"	KTP Mobile Panels 7" - 9"	RT Advanced
Codec	H.264	MJPEG	MJPEG	MJPEG	MJPEG	MJPEG
Auflösung Kamera-Anzeige	320x240 (QVGA)	320x240 (QVGA)	640x480	1280x720	640x480 (VGA)	1280x720
Auflösung Kamera	320x240 (QVGA)	320x240 (QVGA)	320x240	1280x720	640x480 (VGA)	1280x720
Bildfrequenz (fps)	5	5	5	5	5	5
Bildverzögerung (Sekunden)	<1	<1	<1	<1	<1	<1
Qualität	Standard	Standard	Standard	Standard	Standard	Standard
Empfehlung mit Panel	geeignet	geeignet	geeignet	geeignet	geeignet	geeignet

Hinweis

Testergebnisse

Die optimale Einstellungen für RT Advanced wurden in Tests auf einem IPC427D mit dem Betriebssystem Windows 7 Ultimate und jeweiligen Kameras mit Standardeinstellungen ermittelt.

Die Testergebnisse sind konfigurationsspezifisch und hängen von Hardware, Betriebssystem und installierten Codecs ab.

Gegebenenfalls erhalten Sie die fehlende Codecs z. B. von dem Hersteller Ihrer Kamera.

15.3.12 Daten einer Speicherkarte anzeigen

15.3.12.1 Grundlagen

Einleitung

WinCC bietet Ihnen die Möglichkeit Daten Ihrer Speicherkarte anzeigen zu lassen. Dabei können Sie sowohl Speicherkarten des Bediengeräts als auch der CPU verwenden.

Sie haben folgende Möglichkeiten:

Backup anzeigen (Seite 2300)

Backup umbenennen und löschen (Seite 2302)

Bediengeräte-Image anzeigen (Seite 2303)

Bediengeräte-Image löschen (Seite 2304)

Bediengeräte-Image auf der Speicherkarte erstellen (Seite 2305)

Siehe auch

Backup anzeigen (Seite 2300)

Backup umbenennen und löschen (Seite 2302)

Bediengeräte-Image anzeigen (Seite 2303)

Bediengeräte-Image löschen (Seite 2304)

Bediengeräte-Image auf der Speicherkarte erstellen (Seite 2305)

15.3.12.2 Arbeiten mit Backups

Backup anzeigen

Einleitung

Wenn Sie auf einer Speicherkarte das Backup eines Basic Panels speichern, können Sie sich das Backup im TIA Portal anzeigen lassen.

Voraussetzungen

- WinCC ist installiert.
- Eine Speicherkarte mit einem Backup ist vorhanden.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
- Die Projektansicht ist geöffnet.

Backup auf der Speicherkarte im Kartenleser

1. Legen Sie die Speicherkarte in den Kartenleser.
2. Öffnen Sie in der Projektnavigation "SIMATIC Card Reader".
3. Wählen Sie das Laufwerk des Kartenlesers.
   Der Ordner "Online Card Data" wird angezeigt
4. Öffnen Sie den Ordner "Online Card Data "
5. Klicken Sie auf das Backup und öffnen Sie das Kontextmenü.
6. Wählen Sie "Eigenschaften".

Backup auf der Speicherkarte der PLC

Wenn das Backup auf der Speicherkarte der PLC abgelegt wurde, gehen Sie wie folgt vor:

1. Verbinden Sie die PLC mit dem Projektierungs-PC.
2. Klicken Sie in der Projektnavigation auf die PLC.
3. Wählen Sie im Kontextmenü "Online verbinden".
   Eine Verbindung zur PLC wird hergestellt.
   Wenn die PLC verbunden ist, wird der Ordner "Online Card Data" angezeigt.
4. Öffnen Sie den Ordner "Online Card Data".

   Hinweis

   Zugriff auf eine kennwortgeschützte PLC

   Wenn Sie auf eine PLC zugreifen, die durch ein Kennwort geschützt ist, werden Sie aufgefordert, ein Kennwort einzugeben.

   Um die Daten der Speicherkarte anzeigen zu lassen, benötigen Sie mindestens einen Lesezugriff.

5. Klicken Sie auf das Backup und öffnen Sie das Kontextmenü.
6. Wählen Sie "Eigenschaften".

Ergebnis

Die Eigenschaften des Backups werden in einem separaten Dialog angezeigt.

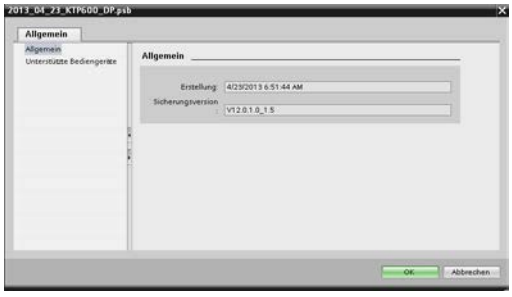

Backup umbenennen und löschen

Einleitung

Sie haben die Möglichkeit die Backups einer Speicherkarte in der Projektnavigation des TIA Portals umzubenennen und zu löschen.

Voraussetzungen

- WinCC ist installiert.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
  Bzw. Die PLC ist mit dem Projektierungs-PC online verbunden.
- Eine Speicherkarte mit einem Backup ist vorhanden.
- Die Projektansicht ist geöffnet.
- Das Backup wird in der Projektnavigation angezeigt.

Hinweis

Zugriff auf eine kennwortgeschützte PLC

Wenn Sie auf eine PLC zugreifen, die durch ein Kennwort geschützt ist, werden Sie aufgefordert, ein Kennwort einzugeben.

Um die Daten der Speicherkarte zu löschen bzw. umzubenennen, benötigen Sie einen Schreibzugriff.

Vorgehen

1. Klicken Sie in der Projektnavigation auf das Backup.
2. Öffnen Sie das Kontextmenü.
3. Um die Datei umzubenennen, wählen Sie "Umbenennen".
4. Geben Sie einen neuen Namen ein.
5. Um die Datei zu löschen, wählen Sie "Löschen".

Ergebnis

Die Backup-Datei wurde umbenannt bzw. gelöscht.

15.3.13 Arbeiten mit Bediengeräte-Images

15.3.13.1 Bediengeräte-Image anzeigen

Einleitung

Wenn Sie auf einer Speicherkarte ein Bediengeräte-Image eines Comfort Panels speichern, können Sie sich das Bediengeräte-Image im TIA Portal anzeigen lassen.

Voraussetzungen

- WinCC ist installiert.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
- Eine Speicherkarte mit dem Bediengeräte-Image ist vorhanden.
- Die Projektansicht ist geöffnet

Vorgehen

1. Legen Sie die Speicherkarte in den Kartenleser.
2. Öffnen Sie in der Projektnavigation "SIMATIC Card Reader".
3. Wählen Sie das Laufwerk des Kartenlesers.
   Der Ordner "Online Card Data" wird angezeigt.
4. Öffnen Sie den Ordner "Online Card Data".
   Die verfügbaren Images des Bediengeräts werden weiteren Ordnern angezeigt.
5. Klicken Sie auf das gewünschte Bediengeräte-Image.
6. Wählen Sie im Kontextmenü "Eigenschaften".

Ergebnis

Die Eigenschaften des Bediengeräte-Image werden in einem separaten Dialog angezeigt.

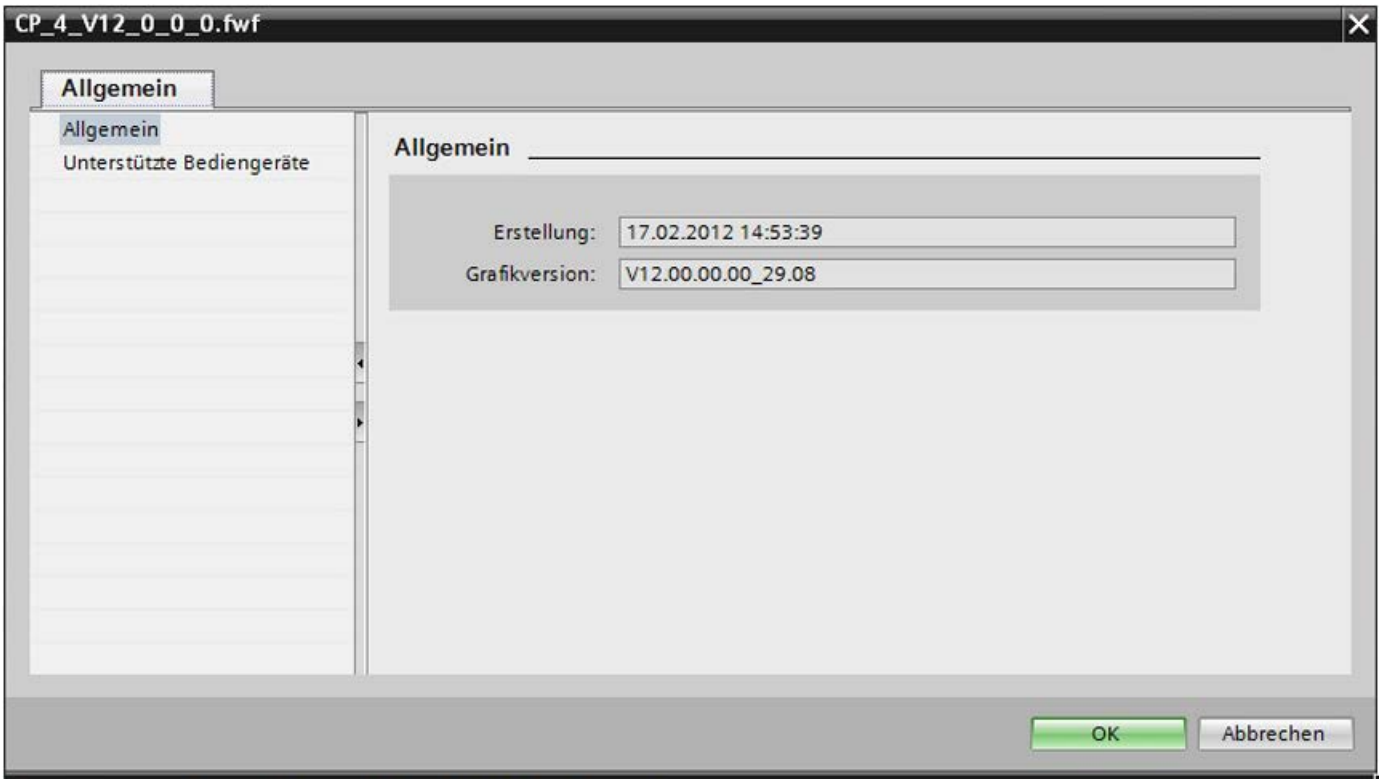

Siehe auch

Bediengeräte-Images ab V15 (Seite 2307)

15.3.13.2 Bediengeräte-Image löschen

Einleitung

Sie haben die Möglichkeit das Bediengeräte-Image eines Comfort Panels in der Projektnavigation des TIA Portals zu löschen.

Voraussetzungen

- WinCC ist installiert.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
- Eine Speicherkarte mit einem Bediengeräte-Image ist vorhanden.
- Die Projektansicht ist geöffnet.
- Das Bediengeräte-Image wird in der Projektnavigation angezeigt.

Vorgehen

1. Klicken Sie in der Projektnavigation auf das Bediengeräte-Image.
2. Öffnen Sie das Kontextmenü.
3. Um die Datei zu löschen, wählen Sie "Löschen".

Ergebnis

Das Bediengeräte-Image wurde gelöscht.

Siehe auch

Bediengeräte-Images ab V15 (Seite 2307)

15.3.13.3 Bediengeräte-Image auf der Speicherkarte erstellen

Einleitung

Sie können das Bediengeräte-Image eines Comfort Panels ändern, ohne das Comfort Panel mit dem Projektierungs-PC zu verbinden.

Sie erstellen ein Bediengeräte-Image auf einer externen Speicherkarte oder einem USB-Stick und übertragen das Bediengeräte-Image später auf das Comfort Panel.

Voraussetzungen

- WinCC ist installiert.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
- Die Projektansicht ist geöffnet.

Vorgehen

1. Legen Sie die Speicherkarte in den Kartenleser.
2. Klicken Sie in der Projektnavigation auf die Speicherkarte.
3. Öffnen Sie das Kontextmenü.

4. Wählen Sie den Eintrag "Create HMI OS image on memory card".
   Ein Dialog wird geöffnet.

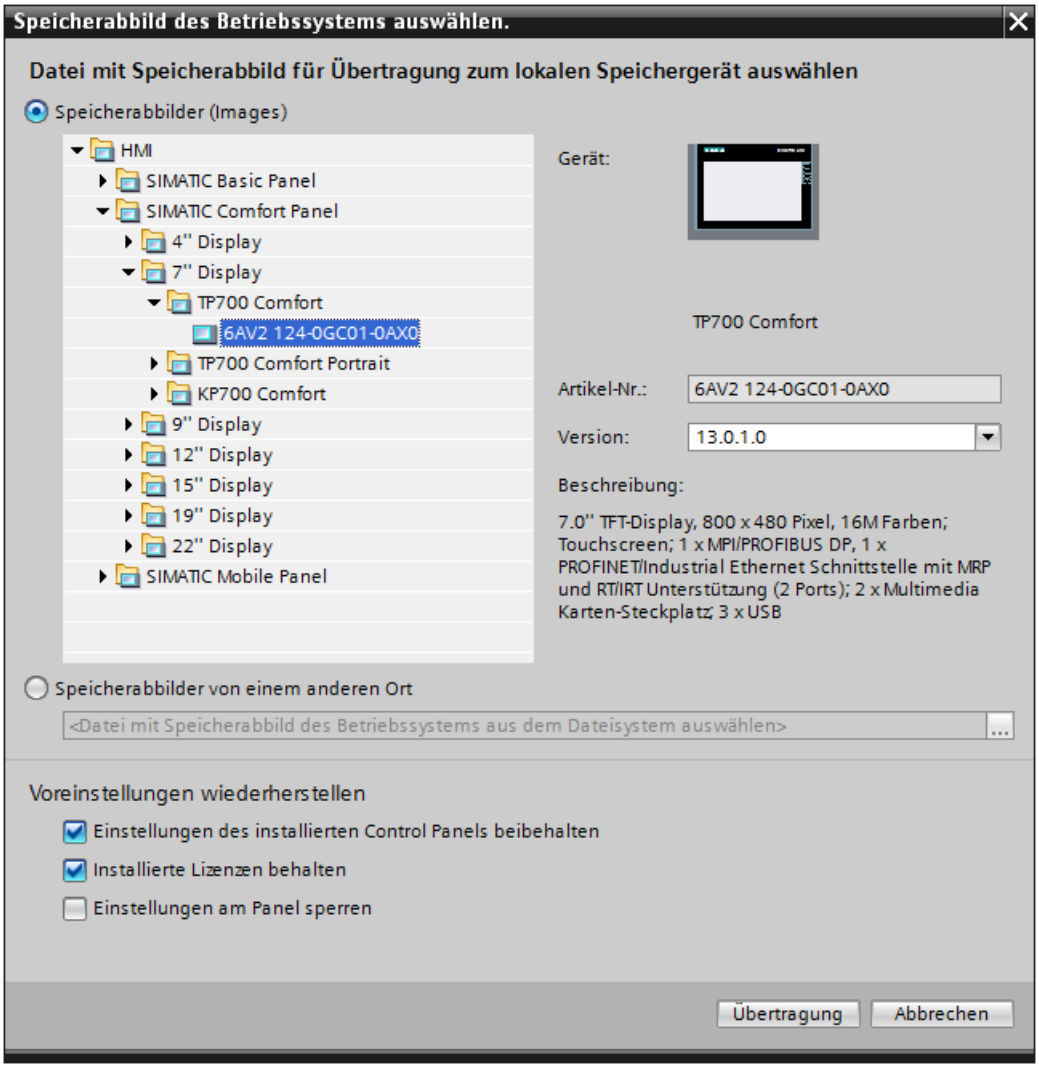

5. Wählen Sie ein Bediengeräte-Image aus.
6. Wählen Sie Einstellungen für die Wiederherstellung aus.
   - "Installierte Einstellungen des Control Panels beibehalten" aktiviert:
     Die von Ihnen vorgenommenen Einstellungen auf dem Comfort Panel bleiben erhalten.
   - "Installierte Lizenzen beihalten" aktviert:
     Die Lizenzen bleiben auf dem Panel erhalten.
   - "Einstellungen am Panel sperren" aktviert:
     Die gewählten Einstellungen für "Installierte Einstellungen des Control Panels beibehalten" und "Installierte Lizenzen beihalten" können Sie auf dem Comfort Panel nicht mehr ändern.

Ergebnis

Sie haben ein Bediengeräte-Image auf der Speicherkarte erstellt. Alternativ verwenden Sie anstatt der Speicherkarten ein USB-Stick.

Siehe auch

Bediengeräte-Images ab V15 (Seite 2307)

15.3.13.4 Bediengeräte-Images ab V15

Einleitung

Ab TIA Portal V15 sind nicht alle Bediengeräte-Images der vorhergehenden Versionen in der Standard WinCC-Installation enthalten. Auf diese Weise wird die Größe des TIA Portal-Installationspakets und der nachfolgenden Updates im Vergleich zu den vorhergehenden Versionen reduziert. Die Bediengeräte-Images, die nicht mehr in der WinCC-Installation enthalten sind, werden auf zwei zusätzlichen DVDs, DVD 1 und DVD 3, mitgeliefert und können jederzeit dazu installiert werden.

Außerdem haben Sie die Möglichkeit, das jeweilige Bediengeräte-Image zur Installation aus SIOS herunterzuladen.

Verfügbarkeit der Bediengeräte-Images

Die folgende Tabelle zeigt die Verfügbarkeit der Bediengeräte-Images in Kombination mit der Geräteversion:

Bediengeräte	V12.0	V13.0	V13.0.1	V13.0.2	V14.0	V14.0.1	V15.0
Basic Panels	DVD 1	-	-	-	-	-	-
Mobile Panels	DVD 1	-	-	-	-	-	-
Basic Panels 2nd Generation	-	DVD 3	DVD 3	DVD 3	DVD 3	DVD 1	DVD 1
Mobile Panels 2nd Generation	-	-	DVD 3	DVD 3	DVD 3	DVD 1	DVD 1
Comfort Panels	DVD 1	DVD 3	DVD 3	DVD 3	DVD 3	DVD 1	DVD 1

Optionen für die Installation

Beachten Sie bei der Installation von TIA Portal die folgenden auswählbaren Optionen:

- "SIMATIC WinCC Panel Images (aktuell)": Auswählbar bei der Installation der DVD 1
- "SIMATIC WinCC Legacy Panel Images für V15.0": Auswählbar bei der Installation der DVD 3

Die folgende Tabelle zeigt die Verfügbarkeit der Geräte-Versionen in Kombination mit den beiden bei der Installation auswählbaren Optionen:

DVD	Option für die Installation	Verfügbare Geräteversion
DVD 1	"SIMATIC WinCC Panel Images (aktuell)" aktiviert	V12.0, V14.01., V15.0
DVD 1	"SIMATIC WinCC Panel Images (aktuell)" deaktiviert	V12.0
DVD 1 und DVD 3	"SIMATIC WinCC Panel Images (aktuell)" deaktiviert "SIMATIC WinCC Legacy Panel Images für V15.0" aktiviert	V12.0, V13.0, V13.0.1, V14.0, V14.0.1, V15
DVD 1 und DVD 3	"SIMATIC WinCC Panel Images (aktuell)" aktiviert "SIMATIC WinCC Legacy Panel Images für V15.0" deaktiviert	V12.0, V13.0, V13.0.1, V14.0

Wenn bei der WinCC-Installation das jeweilige Bediengeräte-Image nicht mitinstalliert wurde, können Sie die folgenden Aktionen erst nach der nachträglichen Installation des fehlenden Bediengeräte-Images durchführen:

- Eine "Pack&Go"-Datei für das betreffende Bediengerät erzeugen.
- Eine OS-Aktualisierung für das betreffende Bediengerät anstoßen.
- Ein Projekt offline über einen USB-Datenträger auf das betreffende Bediengerät laden.
- Ein Projekt in das betreffende Bediengerät laden.
- Ein Bediengeräte-Image auf der Speicherkarte für das betreffende Bediengerät erstellen.

Eine Meldung informiert Sie darüber, dass das fehlende Bediengeräte-Image zuerst installiert werden muss.

Siehe auch

Bediengeräte-Image anzeigen (Seite 2303)

Bediengeräte-Image löschen (Seite 2304)

Bediengeräte-Image auf der Speicherkarte erstellen (Seite 2305)

15.3.14 Grundlagen zur Bedienung in Runtime

15.3.14.1 Übersicht

Bedienvarianten eines Bediengeräts

Abhängig von Ihrem Bediengerät bedienen Sie Ihre Anlage folgendermaßen:

- Bedienung über Touchscreen
  Das Display des Geräts ist berührungssensitiv. Sie bedienen die Bedienobjekte im Display mit dem Finger oder einen Touch-Stift.
- Bedienung über Steuer- und Funktionstasten
  Im Gehäuse des Geräts sind Steuer- und Funktionstasten integriert.
  - Steuertasten haben eine festgelegte Funktion, z. B. Navigieren oder Quittieren von Meldungen.
  - Funktionstasten sind frei belegbar und ihre Funktionen daher projektspezifisch.
- Bedienung über Maus und Tastatur
  Sie schließen Sie Maus und Tastatur an das Bediengerät an und bedienen die Bedienobjekte im Display wie bei einem PC.

Individuell projektierte Bedienung

Dem Projekteur stehen unterschiedliche Möglichkeiten zur Verfügung, die Bedienung einzurichten.

Beispiele für Aktionen, deren Ausführung immer projektspezifisch festgelegt ist:

- Bildwechsel
- Protokollierung
- Runtime-Sprache umstellen

Für die Ausführung bestimmter Funktionen gibt es keine speziellen Bedienobjekte. Der Projekteur legt die Ausführung projektspezifisch fest. Der Bildwechsel kann z. B. über eine Schaltfläche oder eine Funktionstaste ausgelöst werden.

Informationen zu projektspezifischen Bedienungen finden Sie in der Anlagendokumentation.

15.3.14.2 Bedienung über Touchscreen

Überblick zur Bedienung über Touchscreen

Über den Touch-Screen bedienen Sie das Bediengerät oder das Projekt, das auf Ihrem Bediengerät läuft.

Touch-Screen bedienen

ACHTUNG
Beschädigung des Touch-Screens Berühren Sie den Touch-Screen nicht mit spitzen oder scharfen Gegenständen. Verzichten Sie auf schlagartige Berührungen des Touchscreen mit harten Gegenständen und vermeiden Sie eine dauerhafte Bedienung mit Gesten. Beides kann zu einer erheblichen Reduzierung der Lebensdauer bis hin zum Totalausfall des Touchscreen führen. **Auslösen unbeabsichtigter Aktionen** Das gleichzeitige Berühren mehrere Bedienobjekte kann unbeabsichtigte Aktionen auslösen. Berühren Sie immer nur ein Bedienobjekt am Bildschirm. Bedienobjekte sind berührungssensitive Darstellungen am Bildschirm des Bediengeräts.

Besonderheiten beim Bedienen über Touch-Screen

Die Bedienung über Touch-Screen zeichnet sich durch folgende Besonderheiten aus:

- Aktivieren
  Um ein Bedienelement zu aktivieren, berühren Sie den Touch-Screen mit dem Finger oder einen Touch-Stift. Um einen Doppelklick zu erzeugen, berühren Sie das Bedienelement zweimal kurz hintereinander.
- Werteingabe
  Zahlen und Buchstaben geben Sie am Touch-Screen über eine Bildschirmtastatur ein.
- Achtsame Bedienung
  Wenn Sie gleichzeitig mehrere Bedienelemente berühren, lösen Sie damit unbeabsichtigte Aktionen aus.

Werteingabe über Bildschirmtastatur

Die Bildschirmtastatur wird am Display angezeigt, sobald Sie ein Bildobjekt bedienen, das eine Eingabe erfordert. Abhängig vom Bediengerät und vom projektierten Bedienobjekt werden unterschiedliche Bildschirmtastaturen für numerische oder alphanumerische Werte angezeigt. Beim Beenden der Eingabe wird die Bildschirmtastatur wieder ausgeblendet.

Dateibrowser (Windows 8 oder höher)

Auf einem PC mit Windows 8 oder höher mit einem Touch-Screen kann der Dateibrowser-Dialog nur mit der Maus, Tastatur oder Bildschirmtatstatur ohne Verwendung von Touch-Funktion bedient werden. Auf einem Touch-Screen PC mit Windows 8 oder höher nutzen Sie den Dateibrowser-Dialog von Windows-Betriebssystem mithilfe eines Skripts.

Bildschirmtastatur

Darstellung

Die folgende Abbildung zeigt den prinzipiellen Aufbau einer Bildschirmtastatur auf einem Runtime Advanced Bediengerät.

Alphanumerische Bildschirmtastatur

Hinweis

Sprachumschaltung

Die Sprachumschaltung im Projekt hat keinen Einfluss auf die alphanumerische Bildschirmtastatur. Die Eingabe von kyrillischen oder asiatischen Zeichen ist deshalb nicht möglich.

Numerische Bildschirmtastatur

Bedienelemente

Folgende Tasten sind auf der Bildschirmtastatur aller Bediengeräte vorhanden:

Schaltfläche	Name	Funktion
←	Cursor nach links	Navigiert nach links
→	Cursor nach rechts	Navigiert nach rechts
BSP	Backspace	Löscht ein Zeichen
ESC	Escape	Bricht die Eingabe ab
↵	Enter	Bestätigt die Eingabe
Help	Help	Zeigt den Hilfetext an Nur wenn für das Bedienobjekt ein Hilfetext projektiert wurde, wird diese Taste angezeigt.

Multitouch-Funktionen nutzen

Multitouch-Bedienung

Einleitung

RT Advanced unterstützt unter Windows 7, Windows 8 und Windows 10 Multitouch-Bedienung. Sie bedienen die Objekte am Multitouch-Screen durch Berühren mit einem oder zwei Fingern. Multitouch-Bedienung wird von TIA Portal ab V13 unterstützt. Versionen V12 oder älter unterstützen nur Singletouch-Bedienung.

Voraussetzung

Für die Nutzung der Multitouch-Funktionen muss der verwendete Monitor multitouchfähig sein.

Scrollen in Listen und Controls

Sie können in Listen und Controls durch Ziehen am Bildschirm scrollen.

Durch horizontales Ziehen verschieben Sie den Inhalt des Bilds nach links bzw. nach rechts. Durch vertikales Ziehen scrollen Sie in der Ansicht nach oben oder nach unten. Während Sie scrollen, wird Ihnen ein Anzeiger als Hinweis für Position angezeigt. Wenn Sie in einer Liste diagonal am Bildschirm ziehen, verschiebt sich der Inhalt gleichzeitig horizontal und vertikal.

Im Objekt "Kurvenanzeige" wird horizontales Scrollen nicht unterstützt.

Sie scrollen auf einer Seite nach oben oder unten fünf mal schneller, indem Sie mit zwei Fingern gleichzeitig nach oben oder nach unten ziehen.

Hinweis

Scrollen in Windows 8.1

Auf den Multitouch-Bediengeräten mit Windows 8.1 scrollen Sie fortlaufend, indem Sie auf dem Display doppelt Tippen und Ziehen.

Hinweis

Aktuelle Ansicht ist nicht persistent

Die durch Scrollen veränderte Position im Kurvenfenster wird nicht gespeichert.
Bei einem Bildwechsel wird die Kurvenansicht in die Standardeinstellung zurückgesetzt.

Ansicht vergrößern und verkleinern

Sie vergrößern oder verkleinern die Ansicht in Objekten "Kurvenanzeige" und "f(x)-Kurvenanzeige" durch Auf- und Zuziehen von zwei Fingern.

Mit Doppeltippen schalten Sie von der vergrößerten Kurvenanzeige zurück zur Normalansicht.

Im Objekt "Kurvenanzeige" ist die Zoomfunktion auf die Zeitachse beschränkt.

Wenn Sie bei der Projektierung der Werteachsen in f(x)-Kurvenanzeige die Option "Bereich > Automatisch anpassen" aktiviert haben, werden die Achsen während des Zoomens fortlaufend berechnet.

Hinweis

Aktuelle Ansicht ist nicht persistent

Die Änderungen von Zoomfaktor werden nicht gespeichert.

Bei einem Bildwechsel wird die Kurvenansicht in die Standardeinstellung zurückgesetzt.

Unterstützte Gesten

Unterstützte Gesten

Folgende Gesten werden unterstützt.

Geste		Funktion
	Tippen	Durch das Tippen wählen Sie z.B. eine Schaltfläche aus.
	Ziehen	Mit einem Finger können Sie z. B. in Listen horizontal und vertikal scrollen. Durch diagonales Ziehen in Listen scrollen Sie gleichzeitig horizontal und vertikal.
		Mit zwei Fingern scrollen Sie schnell auf einer Seite nach oben oder unten.
	Skalieren	Durch Auf- und Zuziehen können Sie die Anzeige des Controls vergrößern bzw. verkleinern.
	Doppeltippen	Durch das doppelte Tippen auf das Display setzen Sie die Ansicht wieder auf den Standard Zoomfaktor von 100% zurück.

Hinweis

Bedienung mit drei oder mehr Fingern

Verwenden Sie bei der Bedienung nicht drei oder mehr Finger. Das kann zu Fehlbedienungen führen.

Zweihand-Bedienung von Bedienelementen

Zweihand-Bedienung von Bedienelementen

Einleitung

WinCC unterstützt die Zweihand-Bedienung von Bedienlementen mit RT Advanced. Damit wird eine sichere Bedienung von solchen Bedienelementen ermöglicht, mit denen kritische Systemeinstellungen geändert werden können, z. B. Steuervariablen mit Maschinengrenzwerten.

Gesperrte und entsperrte Bedienelemente

Für die Zweihand-Bedienung von Bedienelementen definieren Sie bestimmte Bedienelemente als "gesperrte Bedienelemente". Gesperrte Bedienelemente sind in Runtime gewöhnlich nicht bedienbar. Der Bediener kann die gesperrten Bedienelemente nur dann bedienen, wenn er gleichzeitig eine dafür vorgesehene Freigabeschaltfläche drückt.

In Runtime sind die gesperrten Bedienelemente nur dann über die Tab-Reihenfolge erreichbar, wenn gleichzeitig eine Freigabeschaltfläche gedrückt wird.

Gesperrte Bedienelemente und Freigabeschaltflächen

Die folgenden Bedienelemente können Sie als gesperrte Bedienelemente konfigurieren:

- EA-Feld als Eingabefeld oder Eingabe/Ausgabefeld
- Schaltfläche (sichtbar)
- Schaltfläche (unsichtbar)
- Schieberegler
- Datum/Uhrzeit-Feld

Wenigstens eine Schaltfläche im Bild müssen Sie als Freigabeschaltfläche konfigurieren. Das kann eine beliebige nicht gesperrte Schaltfläche sein.

Hinweis

Im Permanentbereich können Sie weder gesperrte Bedienelemente noch Freigabeschaltflächen projektieren.

Vorlagen mit gesperrten Bedienelementen

Sie können gesperrte Bedienelemente auch in Vorlagen für Bilder verwenden. Um Bedienelemente in Vorlagen zu sperren oder zu entsperren, gehen Sie genauso vor wie bei Bedienelementen in Bildern.

In der Vorlage können Sie auch eine Freigabeschaltfläche vorsehen. Dafür gehen Sie genauso vor wie bei Freigabeschaltflächen in Bildern.

In einem aus der Vorlage erstellten Bild sind dann die Freigabeschaltfläche aus der Vorlage und eine eventuelle Freigabeschaltfläche des Bildes wirksam.

Darstellung in Runtime

In Runtime sind die gesperrten Bedienelemente gedimmt dargestellt. Sobald der Bediener auf ein gesperrtes Bedienelement drückt, werden die Freigabeschaltflächen optisch hervorgehoben. Die gesperrten Bedienelemente sind dann normal sichtbar, wenn diese über die Freigabeschaltfläche entsperrt sind.

Sie können die Freigabeschaltflächen daher auch als unsichtbare Bedienelemente konfigurieren.

Simulation von Projekten mit Multitouch-Funktionen

WinCC unterstützt die Simulation der projektierten Multitouch-Funktionen. Voraussetzung ist, dass Sie einen multitouchfähigen Monitor einsetzen.

Bedienelemente sperren und entsperren

In Projekten für Multitouchgeräte können Sie Bedienelemente sperren und entsperren. Gesperrte Bedienelemente sind in Runtime nur dann bedienbar, wenn der Bediener gleichzeitig eine Freigabeschaltfläche drückt.

Sie können ein einzelnes oder gleichzeitig mehrere Bedienelemente sperren und entsperren.

Vorgehensweise

1. Projektieren Sie die Bedienelemente vom Typ EA-Feld, Schaltfläche oder Schieberegler.
2. Selektieren Sie das oder die gewünschten Bedienelemente.

3. Um die Bedienelemente zu sperren, aktivieren Sie unter "Eigenschaften > Eigenschaften > Security" die Option "Zweihand-Bedienung aktivieren".

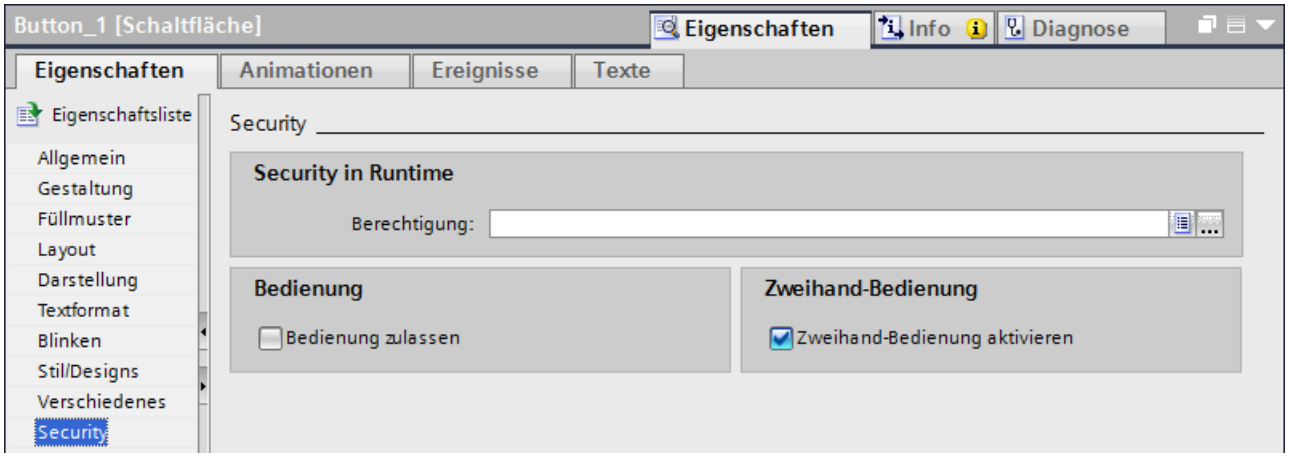

4. Um die Bedienelemente zu entsperren, deaktivieren Sie unter "Eigenschaften > Eigenschaften > Security" die Option "Zweihand-Bedienung aktivieren".

In Runtime sind die gesperrten Bedienelemente nur dann bedienbar, wenn gleichzeitig eine Freigabeschaltfläche gedrückt wird.

Freigabeschaltfläche definieren

Damit die gesperrten Bedienelemente benutzt werden können, müssen Sie im selben Bild wenigstens eine Freigabeschaltfläche konfigurieren.

Wenn keine Freigabeschaltfläche projektiert ist, macht WinCC Sie beim Übersetzen des Projekts darauf aufmerksam.

Freigabeschaltfläche in Bild oder Vorlage projektieren

Damit der Bediener gesperrte Bedienelemente auf Multitouchgeräten bedienen kann, muss in demselben Bild wenigstens eine Freigabeschaltfläche projektiert sein.

Wenn das Bild auf einer Vorlage beruht und in der Vorlage bereits eine Freigabeschaltfläche definiert ist, wirkt die Freigabeschaltfläche aus der Vorlage automatisch als Freigabeschaltfläche des Bildes. Sie können dann eine weitere Freigabeschaltfläche im Bild selbst projektieren.

Voraussetzung

In dem Bild oder in der Vorlage ist mindestens eine nicht gesperrte Schaltfläche projektiert.

Vorgehensweise

1. Wählen Sie unter "Eigenschaften > Eigenschaften > Allgemein" unter "Freigabe-Schaltfläche" die gewünschte Schaltfläche des Bildes oder der Vorlage aus.

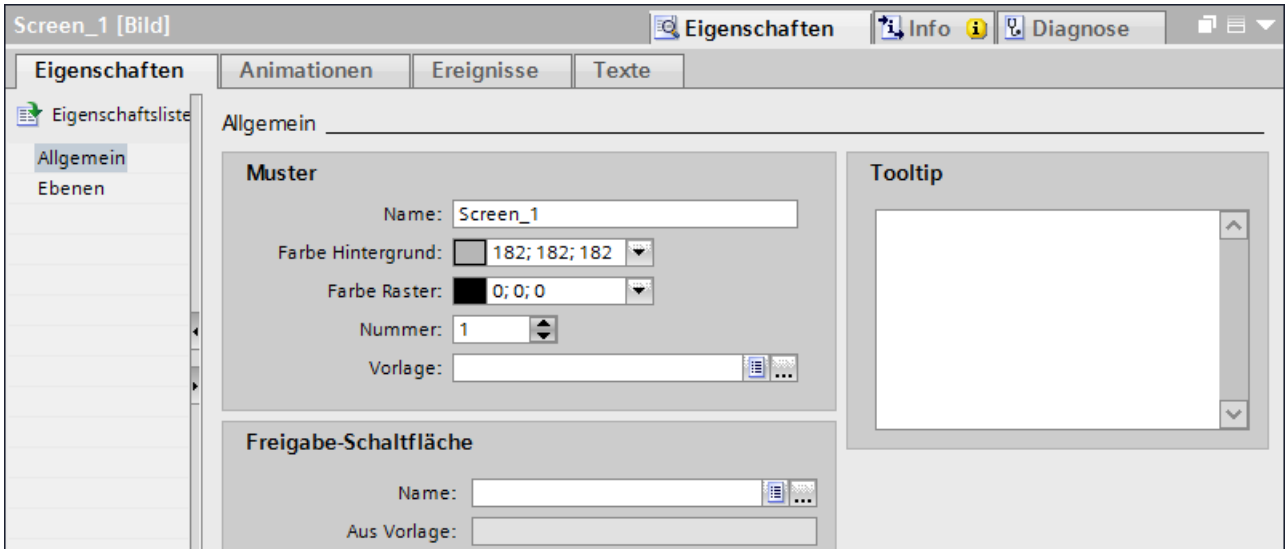

Hinweis

Sie können hierfür auch eine Schaltfläche aus einer Gruppe auswählen. Schaltflächen aus Bildbausteinen stehen dagegen nicht zur Verfügung.

Die ausgewählte Schaltfläche kann auch unsichtbar dargestellt sein. In Runtime wird sie beim Drücken eines gesperrten Bedienelements optisch hervorgehoben.

Wenn das Bild auf einer Vorlage beruht und in dieser Vorlage bereits eine Freigabeschaltfläche definiert ist, fungiert diese Schaltfläche automatisch ebenfalls als Freigabeschaltfläche des Bildes.

In Runtime sind die gesperrten Bedienelemente nur dann bedienbar, wenn der Bediener gleichzeitig eine der projektierten Freigabeschaltflächen drückt.

Einschränkungen im Sm@rtServer/Client Betrieb

Multitouch-Server und Singletouch-Client

Bedienung durch Berührung mit zwei oder mehr Fingern sowie Multitouch-Funktionen werden vom Singletouch Client-Gerät nicht unterstützt. Singletouch-Client kann mit den Gesten des Multitouch Sm@rtServers nicht bedient werden.

Singletouch-Gesten funktionieren genauso wie in WinCC V12.

Singletouch-Server und Multitouch-Client

Bedienung durch Berührung mit zwei oder mehr Fingern sowie Multitouch-Funktionen werden vom Singletouch Server-Gerät nicht unterstützt. Multitouch-Client kann mit den Gesten des Singletouch Sm@rtServers nicht bedient werden.

Singletouch-Gesten funktionieren genauso wie in WinCC V12.

15.3.14.3 Bedienung über Tasten

Überblick zur Bedienung über Tasten

Einleitung

Mit dem Tasten des Bediengeräts bedienen Sie das Control Panel/Start Center Ihres Geräts oder das Projekt, das auf Ihrem Gerät läuft. Abhängig vom Gerät stehen Ihnen Steuertasten und Funktionstasten zur Verfügung.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

Steuertasten und Tastenkombinationen

Einleitung

Die folgenden Tabellen zeigen die Steuertasten, mit denen Sie das Projekt bedienen.

Hinweis

Die Verfügbarkeit der Steuertasten ist abhängig vom verwendeten Bediengerät.

Funktionen an Tasten-Bediengeräten lösen Sie entweder über eine Taste oder Tastenkombination aus. Bei Tastenkombinationen halten Sie die erste Taste gedrückt. Danach drücken Sie die zweite Taste.

Im Display navigieren

Taste oder Tastenkombination		Funktion	Beschreibung
TAB		Tabulator	Wählt das nächste Bedienobjekt in der Tab-Reihenfolge aus
SHIFT	TAB	Tabulator	Wählt das vorherige Bedienobjekt in der Tab-Reihenfolge aus
◀ ▶ ▼ ▲		Cursortasten	Wählen das nächste Bedienobjekt, links, rechts, oberhalb oder unterhalb des aktuellen Bildobjekts aus Navigieren im Bedienobjekt

Bedienobjekte bedienen

Taste oder Tastenkombination		Funktion	Beschreibung
ENTER		Eingabetaste	• Bedient Schaltflächen. • Übernimmt und beendet eine Eingabe • Öffnet eine Auswahlliste • Schaltet innerhalb eines Eingabefelds zwischen Zeichenmodus und Normalmodus um Im Zeichenmodus ist ein einzelnes Zeichen markiert. In diesem Modus können Sie mit den Cursortasten im Zeichensatz weiterschalten.
SHIFT	▶ ◀ ▲ ▼	Cursor positionieren	Positionieren den Cursor innerhalb eines Bedienobjekts, z. B. im E/A-Feld
←		Zeichen löschen	Löscht das Zeichen links von der aktuellen Cursor-Position
INS DEL		Zeichen löschen	Löscht das Zeichen rechts von der aktuellen Cursor-Position
ESC		Abbrechen	• Löscht eingegebene Zeichen einer Werteingabe und stellt den ursprünglichen Wert wieder her • Schließt den aktiven Dialog.
FN	HOME	Zum Anfang blättern	Blättert in einer Liste zum Anfang
HOME		Zurückblättern	Blättert in einer Liste um eine Seite zurück
FN	END	Zum Ende blättern	Blättert in einer Liste zum Ende.
END		Weiterblättern	Blättert in einer Liste um eine Seite weiter
ALT	▼	Auswahlliste öffnen	Öffnet eine Auswahlliste
CTRL	ENTER	Wert übernehmen	Übernimmt den ausgewählten Wert in der Auswahlliste, ohne die Liste zu schließen

Tastenkombination eingeben

Taste	Funktion	Zweck
A-Z	Umschalten (Ziffern/Buchstaben)	Schaltet die Belegung der Eingabetasten von Ziffern auf Buchstaben um • Keine LED leuchtet: Die Ziffern-Belegung ist aktiv. Einmaliges Drücken der Taste schaltet auf Buchstaben-Belegung um. • Die rechte oder die linke LED leuchtet: Die linke oder rechte Buchstaben-Belegung ist aktiv. Jedes Drücken der Taste schaltet zwischen der linken Buchstaben-Belegung, der rechten Buchstaben-Belegung und der Ziffern-Belegung um.
SHIFT	Umschalten (Groß-/Kleinschreibung)	Verwendung in Tastenkombinationen, z. B. Umschalten auf Großbuchstaben
FN	Auf zusätzliche Tastenbelegung umschalten	Auf einigen Tasten finden Sie in der linken unteren Ecke blaue Sonderzeichen, z. B. das Prozentzeichen "%". Um diese Zeichen einzugeben, drücken Sie gleichzeitig die links abgebildete Sonderzeichen-Taste.
CTRL	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. Navigation in Kurvendarstellung
ALT	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. bei Bildobjekt "Beobachtungstabelle"

Meldungen quittieren

Taste	Funktion	Zweck
ACK	Quittieren	Quittiert die aktuell angezeigte Meldung oder alle Meldungen einer Meldegruppe Die LED leuchtet, solange unquittierte Meldungen anstehen.

Hilfetext anzeigen

Taste	Funktion	Beschreibung
HELP	Hilfetext anzeigen	Öffnet zum markierten Objekt ein Fenster mit dem projektierten Hilfetext, , z. B. für eine Meldung oder ein E/A-Feld Wenn zum markierten Objekt ein Hilfetext existiert, leuchtet die LED.

Taste oder Tastenkombination

Funktionstasten

Die Belegung der Funktionstasten (F1, F2, F3 …) wird bei der Projektierung festgelegt.

Funktionstasten mit globaler Funktionsbelegung

Eine global belegte Funktionstaste löst unabhängig vom angezeigten Bild immer dieselbe Aktion am Bediengerät oder in der Steuerung aus. Eine solche Aktion ist z. B. das Aktivieren eines Bildes oder das Schließen eines Meldefensters.

Funktionstasten mit lokaler Funktionsbelegung

Eine Funktionstaste mit lokaler Funktionsbelegung ist bildspezifisch und damit nur innerhalb des aktiven Bildes wirksam.

Die Funktion einer lokal belegten Funktionstaste kann von Bild zu Bild unterschiedlich sein.

Innerhalb eines Bildes hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

Funktionstasten bedienen

Hinweis

Funktionstaste nach einem Bildwechsel bedienen

Wenn Sie nach einem Bildwechsel eine Funktionstaste bedienen, kann die zugehörige Funktion im neuen Bild bereits angestoßen werden, bevor das Bild vollständig aufgebaut ist.

Direkttasten (Advanced)

Einleitung

Direkttasten setzen direkt vom Bediengerät aus Bits im Peripheriebereich einer SIMATIC S7.

Direkttasten ermöglichen Tastenbedienungen mit kurzen Reaktionszeiten, die z. B. Voraussetzung für den Tippbetrieb sind.

Hinweis

Die Direkttasten sind auch aktiv, wenn sich das Bediengerät in der Betriebsart "Offline" befindet.

Hinweis

Wenn Sie bei laufendem Projekt eine Funktionstaste mit Direkttastenfunktionalität bedienen, wird die Direkttastenfunktion unabhängig vom aktuellen Bildschirminhalt immer ausgeführt.

Hinweis

Sie können Direkttasten nur bei Kopplung über PROFIBUS DP oder PROFINET IO verwenden.

Direkttasten führen zu einer zusätzlichen Grundlast im Bediengerät.

Direkttasten

Folgende Objekte können als Direkttaste projektiert sein:

- Schaltflächen
- Funktionstasten

Bei Bediengeräten mit Touch-Bedienung können Sie zusätzlich Bildnummern definieren. Damit kann der Projekteur die Direkttasten bildspezifisch projektieren.

Nähere Hinweise zur Projektierung von Direkttasten finden Sie unter "WinCC Kommunikation".

Bedienelemente am Mobile Panel

Bedienelemente am Mobile Panel

Abhängig vom Typ Ihres Mobile Panels stehen Ihnen folgende Bedienelemente zur Verfügung:

- Funktionstasten
  Anzahl und Anordnung der Funktionstasten sind abhängig vom gewählten Gerätetyp. Die jeweilige Funktion ist projektspezifisch festgelegt.
- Handrad (optional)
  Das Handrad ist ohne Anschlag drehbar und hat keine Null-Stellung. Im laufenden Projekt geben Sie mit dem Handrad Inkrementalwerte ein.
- Schlüsselschalter (optional)
  Der Schlüsselschalter dient der Verriegelung von Funktionen, die Sie über das Bediengerät auslösen können.
- Leuchtdrucktaster
  Die Funktion des Leuchtdrucktasters ist durch das laufende Projekt vorgegeben.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts und in der Anlagendokumentation.

15.3.14.4 Navigieren im Display (BS)

Einleitung

Sie navigieren auf dem Display Ihres Bediengeräts folgendermaßen:

- Zwischen den projektierten Bildobjekten
- Innerhalb von Bildobjekten
  Wenn Sie ein komplexes Bildobjekt aktivieren, wechselt der Cursorfokus in das Bildobjekt und folgt der dortigen Tab-Reihenfolge.
- In Tabellen von Bildobjekten

Vorgehen

- Um in der festgelegten Tab-Reihenfolge zu navigieren, drücken Sie die Taste <TAB>.
- Um frei zwischen den Bedienelementen zu navigieren, drücken Sie die Cursortasten.

Abhängig von der Projektierung Ihres Bediengeräts, verwenden Sie zum Navigieren auch Funktionstasten oder Tastenkombinationen.

Wenn Sie Ihr Bediengerät über den Touchscreen oder mit der Maus bedienen, navigieren Sie implizit durch das Auslösen einer gewünschten Aktion. Dazu berühren oder klicken Sie das Bedienelement.

Ergebnis

Die Bedienelemente erhalten entsprechend der gewählten Reihenfolge den Cursorfokus. Sie können an dem ausgewählten Bedienelement eine Aktion auslösen.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.3.14.5 Aktion auslösen

Einleitung

Eine Aktion an einem Bedienelement auslösen kann Folgendes bedeuten:

- Ein Befehl wird ausgeführt.
  Beispiel: Um ein Skript zu triggern oder eine vordefinierte Funktion auszuführen, klicken Sie auf eine Schaltfläche.
- Ein Objekt wird aktiviert.
  Beispiel: Um einen Wert einzugeben, aktivieren Sie eine Tabellenzelle mit der Taste <Enter>.

Voraussetzung

- Sie sind zum Bedienelement navigiert, an dem Sie die Aktion auslösen wollen.
- Das Bedienelement hat den Cursorfokus.

Vorgehen

- Drücken Sie die Taste <Enter>.
  Oder
- Berühren Sie das Bedienelement am Touchscreen einfach oder zweimal kurz hintereinander.
  Oder
- Klicken oder doppelklicken Sie das Bedienelement mit der Maus.

Ergebnis

Folgende Ergebnisse sind z. B. möglich:

- Der gewünschte Befehl wird ausgeführt.
- Die Bildschirmtastatur wird geöffnet und/oder der Cursor blinkt im Eingabebereich des Bedienelements.
- Das Element ist ausgewählt und kann bewegt werden.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.3.14.6 Wert eingeben

Einleitung

Je nach Eingabeformat geben Sie numerische oder alphanumerische Werte in ein Eingabefeld ein.

Abhängig von der vorhandenen Hardware geben Sie diese Werte über die Bildschirmtastatur, die Steuertasten des Bediengerätes oder eine externe Tastatur ein.

Voraussetzung

- Das Objekt ist ein Eingabe- oder Tabellenfeld.
- Das Bedienelement ist aktiviert.

Wert eingeben

1. Geben Sie den gewünschten Wert ein.
2. Um den Wert zu bestätigen und das Feld zu verlassen, drücken Sie die Taste <Enter>.
3. Um den Wert zu verwerfen und das Feld zu verlassen, drücken Sie die Taste <Esc>.

Ergebnis

Ein Wert ist eingegeben oder verworfen. Sie navigieren bei Bedarf zum nächsten Bedienelement.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.3.14.7 Bedienelemente bewegen

Einleitung

Sie bewegen in Runtime bei Bedarf die beweglichen Bedienelemente eines Bildobjekts mit der Maus oder über den Touchscreen, z. B. einen Schieberegler oder einen Scrollbalken. Im Folgenden ist die Bedienung mit Tastatur beschrieben.

Voraussetzung

- Ein bewegliches Bedienelement ist aktiviert.

Vorgehen

- Um das Bedienelement zu bewegen, gehen Sie abhängig vom Bedienobjekt folgendermaßen vor:
  - Standard für Touchscreen:: Drücken Sie die Cursortasten.
  - Standard für Tastengeräte:: Drücken Sie <SHIFT> und die Cursortasten.
  - Schalter: Drücken Sie <ENTER>
  - Schieberegler: Drücken Sie <Bild Ab> oder <Bild Auf>

1. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Vorgehen beim Schieberegler

1. Um das Bedienelement zu bewegen, drücken Sie die Tasten " Cursortasten.
2. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Ergebnis

Die Position des beweglichen Bedienelements und die Anzeige im Bildobjekt haben sich verändert.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.3.14.8 Hilfetext anzeigen

Einleitung

Abhängig von der Projektierung stehen Ihnen als Hilfetext zusätzliche Informationen und Bedienhinweise zur Verfügung. Der Hilfetext ist einem Bedienobjekt, einer Meldung oder dem geöffnetem Bild zugeordnet. Der Hilfetext eines E/A-Feldes kann z. B. Hinweise zum einzugebenden Wert enthalten.

Alternativ zur Taste <Help> des Bediengeräts verwenden Sie bei Eingabeobjekten die Taste <Help> der Bildschirmtastatur.

Voraussetzung

- Ein Hilfetext ist am Bedienobjekt, am Bild oder an einer Meldung projektiert.

Hilfetext aufrufen

1. Aktivieren Sie das gewünschte Bedienobjekt.
2. Drücken Sie die Taste <Help> des Bediengeräts.
   Der Hilfetext zum Bedienobjekt wird angezeigt.

Wenn Sie ein Eingabeobjekt über den Touchscreen bedienen, wird die Bildschirmtastatur geöffnet. Wenn die Taste <Help> angezeigt wird, ist für das Bedienobjekt oder das aktuelle Bild ein Hilfetext projektiert.

Wenn zum markierten Bildobjekt kein Hilfetext existiert, wird der Hilfetext für das aktuelle Bild angezeigt, falls dieser projektiert wurde.

Bei langen Hilfetexten verwenden Sie den Scrollbalken.

Abhängig von der Projektierung rufen Sie den Hilfetext auch über ein projektiertes Bedienobjekt auf.

Zwischen Hilfetexten wechseln

- Um zwischen den Hilfetexten des Bedienobjekts und des Bildes zu wechseln, aktivieren Sie das Hilfetextfenster.

Hilfetext ausblenden

- Um den Hilfetext auszublenden, drücken Sie die Taste <Esc> oder erneut die Taste <Help>.

15.3.14.9 Runtime-Sprache umstellen

Einleitung

Das Projekt auf dem Bediengerät kann mehrsprachig sein. Zum Wechseln der am Bediengerät eingestellten Sprache während des laufenden Betriebs ist ein entsprechendes Bedienobjekt projektiert.

Nach dem Starten des Projekts ist immer die zuletzt aktive Sprache eingestellt.

Voraussetzung

- Die gewünschte Sprache für das Projekt ist am Bediengerät verfügbar.
- Die Funktion für die Sprachumschaltung ist mit einem Bedienobjekt verknüpft, z. B. mit einer Schaltfläche.

Sprache auswählen

Zwischen den Sprachen können Sie jederzeit umschalten. Nach dem Aufruf der Funktion für die Sprachumschaltung werden die sprachabhängigen Objekte in der neuen Sprache angezeigt.

Um die Sprache in Runtime umzuschalten, haben Sie folgende Möglichkeiten:

- Über ein projektiertes Bedienobjekt schalten Sie in einer Auswahl von Sprachen weiter.
- Über ein projektiertes Bedienobjekt wählen Sie die gewünschte Sprache direkt aus.

15.3.14.10 Webbrowser der WebKit-Engine: Übersicht

Einleitung

Im Objekt "HTML-Browser" haben Sie in RT Advanced die Möglichkeit, sich zwischen zwei Webbrowser-Typen zu entscheiden. Den Typ des Webbrowsers legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Typ HTML-Browser" fest.

Auf Panels ist nur der Webbrowser der WebKit Engine verfügar.

Der Webbrowser, der auf einer WebKit Engine basiert, bietet keine Active X Unterstützung.

HTML5-Funktionen

Folgende HTML5-Standardfunktionen werden vom Webbrowser der WebKit Engine vollständig oder teilweise unterstützt:

- Parsing-Regeln
- Elemente
- Formen und Felder
- Ausgabe
- Kommunikation
- Benutzerinteraktionen
- Performance
- Sicherheit
- History und Navigation
- 2D-Grafiken
- Speicher
- Animationen
- Web-Anwendungen
- Speicher
- Dateien und Dateisysteme

Hinweis

Microdata, Eingabe, Peer to Peer, Position und Orientierung, Video, Audio, Responsive Images, 3D-Grafiken, Streams und Web-Komponenten werden im Webbrowser der WebKit-Engine nicht unterstützt.

Die nachfolgenden Tabellen zeigen die Verfügbarkeit der HTML5-Funktionen im Webbroser der WebKit-Engine im Detail:

Parsing-Regeln	Verfügbar
<!DOCTYPE html> löst den Standard-Modus aus	Ja
HTML5 Tokenizer	Ja
HTML 5 Tree Building	Ja
Parsing Inline SVG	Ja
Parsing Inline MathML	Ja

Elemente	Verfügbar
Eingebettete unsichtbare Daten	Ja
Neue oder modifizierte Elemente	
Abschnittselemente	Ja
Gruppieren zusammengehöriger Inhaltselemente	Teilweise
Semantische Elemente der Textebene	Teilweise
Interaktive Elemente	Teilweise
Globale Attribute und Methoden	
Versteckte Attribute	Ja
Dynamische Markups einfügen	Ja

Formen und Felder	Verfügbar
Feldtypen	
type = text	Ja
type = search	Ja
type = tel	Ja
type = URL	Ja
type = email	Ja
type = date	Nein
type = month	Nein
type = week	Nein
type = time	Nein
type = datetime	Nein
type = datetime-local	Nein
type = number	Ja
type = range	Ja
type = color	Ja
type = checkbox	Ja
type = image	Ja
type = file	Ja
textarea	Ja
select	Ja

Formen und Felder	Verfügbar
fieldset	Ja
datalist	Ja
keygen	Ja
output	Ja
progress	Ja
meter	Ja
Felder	
Feldvalidierung	Ja
Zuordnung von Formen und Controls	Ja
Andere Attribute	Ja
CSS-Selektoren	Ja
Ereignisse	Ja
Formen	
Formenvalidierung	Ja

Ausgabe	Verfügbar
Full-Screen-Unterstützung	Nein
Web-Benachrichtigungen	Ja

Kommunikation	Verfügbar
Server-Sent-Ereignisse	Ja
Web-Beacons	Nein
XML HttpRequest Level 2	
Datei-Upload	Ja
Response Type	Ja
WebSocket	
Basic Socket Communication	Ja
ArrayBuffer and Blob	Ja

Benutzerinteraktionen	Verfügbar
Drag&Drop	
Attribute	Ja
Ereignisse	Ja
Editieren von HTML	
Editieren von Elementen	Ja
Editieren von Dokumenten	Ja
CSS-Selektoren	Nein
APIs	Ja
Zwischenablage	

Benutzerinteraktionen	Verfügbar
Zwischenablage für API und Ereignisse	Nein
Rechtschreibprüfung	
Rechtschreibattribute	Ja

Performance	Verfügbar
Native binäre Daten	Ja
Workers	
Web Workers	Ja
Shared Workers	Ja

Sicherheit	Verfügbar
Web Cryptography API	Nein
Content Security Policy 1.0	Ja
Content Security Policy 1.1	Nein
Cross-Origin Resource Sharing	Ja
Cross-Document Messaging	Ja
Iframes	
Sandboxes Iframe	Ja
Seamless Iframe	Ja
Iframe mit Inline-Inhalten	Ja

History und Navigation	Verfügbar
Sitzungsverlauf	Ja

2D-Grafiken	Verfügbar
Canvas 2D-Grafiken	Ja
2D-Primitiven	
Texteingabe in Graifken	Ja
Pfadeingabe in Grafiken	Nein
Zeichnen einer Ellipse in Grafiken	Nein
Zeichnen einer gestrichelten Linie in Grafiken	Ja
System Focus Ring	Nein
Funktionen	
Hit Testing	Nein
Blendemodus	Nein
Formate für Bildexport	
PNG	Ja
JPEG	Ja

2D-Grafiken	Verfügbar
JPEG-XR	Nein
WebP	Nein

Animation	Verfügbar
window.requestAnimationFrame	Ja

Web-Anwendungen	Verfügbar
Offline-Ressourcen	
Anwendungs-Cache	Ja
Service Workers	Nein
Content and Scheme Handlers	Nein

Speicher	Verfügbar
Key Value-Speicherung	
Sessionspeicherung	Ja
Lokale Speicherung	Nein
Datenbankspeicherung	
IndexedDB	Nein
Objectstore Blob	Nein
Objectstore ArrayBuffer	Nein
Web SQL-Datenbank	Ja

Dateien und Dateisysteme	Verfügbar
Lesen von Dateien	
Basisunterstützung für Lesen von Dateien	Ja
Aus einer Datei ein Blob erstellen	Ja
Aus einem Blob eine Daten-URL erstellen	Ja
Ein ArrayBuffer aus einem Blob erstellen	Ja
Eine Blob-URL aus einem Blob erstellen	Ja
Auf ein Dateisystem zugreifen	
Dateisystem API	Nein
Datei-API: Ordner und System	Nein

Weitere Funktionen	Verfügbar
Stile	
Stilelemente	Nein
Skripte	

Weitere Funktionen	Verfügbar
Asynchrone Skriptausführung	Ja
Meldung der Skripte-Fehler in Runtime	Ja
Ereignisse zur Skriptausführung	Nein
Base 64 Kodierung und Dekodierung	Ja
JSON Kodierung und Dekodierung	Ja
URL API	Ja
MutationObserver	Ja (voreingestellt)
Promises	Nein
Seitensichtbarkeit	Ja (voreingestellt)
Textauswahl	Ja
Scrollen (Scroll into View)	Ja

Siehe auch

HTML-Browser (Seite 402)

15.3.15 Barcodes mit optischen Handlesegeräten eingeben

Einleitung

Mit den optischen Handlesegeräten haben Sie die Möglichkeit, Bauteile, Maschinen und andere Objekte optisch zu identifizieren und auf Ihrem Bediengerät in einige Bedienobjekte direkt einzugeben.

Optische Handlesegeräte erfassen zweidimensionale Data Matrix Codes, eindimensionale Barcodes und postalische Barcodes.

Folgende optische Handlesegeräte werden unterstützt:

- SIMATIC MV320
- SIMATIC MV340

Die Vorlagen für die Einstellungen und Anweisungen zur Konfiguration finden Sie im Handbuch Ihres optischen Handlesegeräts.

Hinweis

Das optische Handlesegerät wird an die USB-Schnittstelle des HMI-Bediengeräts angeschlossen. Nur ein Gerät kann die USB-Schnittstelle zu einer Zeit nutzen. Daher ist die gleichzeitige Verwendung von USB-Tastatur und optischen Lesegeräts oder zwei optischen Lesegeräten nicht möglich.

Hinweis

Wenn Sie das optische Lesegerät an einem Projektierungs-PC anschließen, stellen Sie sicher, dass die USB-Schnittstelle für das Lesegerät ausreichend Strom zur Verfügung stellt.

Vorgehen

Sie lesen die Markierung eines Objekts mithilfe des angeschlossenen optischen Lesegeräts ein. Der Wert wird auf dem HMI-Bediengerät in das Objekt angegeben, der in Runtime den Fokus erhält.

Nach dem Einlesen bestätigen Sie den Wert mit der Eingabetaste oder mit dem "Suffix - Enter", den Sie zuvor in den Einstellungen Ihres optischen Lesegeräts konfiguriert haben.

Basic Panels, Comfort Panels, Mobile Panels, RT Advanced

Folgende Objekte unterstützen die Eingabe mit optischen Handlesegeräten:

Objekt	Vorbedingung für die Eingabe
E/A-Feld Datum/Uhrzeit-Feld	Der entsprechende Datentyp ist ausgewählt. Das Objekt und die Variablenlänge sind entsprechend konfiguriert. Das Bedienobjekt hat den Cursorfokus.
Rezepturanzeige	Rezepturdatensatz hat den Cursorfokus.
Benutzeranzeige	Anmeldedialog ist geöffnet. Benutzername oder Passwort haben den Cursorfokus.
HTML-Browser	Das Bedienobjekt hat den Cursorfokus.
Runtime Dialoge, die die Tastatureingabe unterstützen	Dialog ist geöffnet und das entsprechende Eingabefeld hat den Cursorfokus.

Comfort Panels, Mobile Panels, RT Advanced

Folgende Objekte unterstützen die Eingabe mit optischen Handlesegeräten:

Objekt	Vorbedingung für die Eingabe
Sm@rtClient-Anzeige	Für Sm@tClient ist Schreibzugriff konfiguriert. Das Bedienobjekt hat den Cursorfokus.
File Browser	Das Feld "File path" hat den Cursorfokus.
Beobachtungstabelle	Das Eingabefeld "Steuerwert" hat den Cursorfokus.
PDF-Anzeige	Das Suchfeld hat den Cursorfokus.
SINUMERIK NC-Bedienanzeige	Das Bedienobjekt hat den Cursorfokus.

Ergebnis

Der Code wurde erfasst und in das entsprechende Eingabefeld eingegeben.

15.3.16 Sicherheit auf dem Bediengerät

15.3.16.1 Grundlagen zu Signaturen

Übersicht

Die Bediengerätesoftware sowie die Runtime- und Geräteoptionen werden durch Signaturen gegen Manipulation geschützt. Auch die Geräteoptionen, die über ProSave angeboten werden, sind ebenfalls signiert.

Bediengerätesignatur

Die Gültigkeit der Bediengerätesignatur wird beim Laden der Bediengerätesoftware geprüft. Wenn die Bediengerätesignatur ungültig ist, wird das Laden der Bediengerätesoftware mit einer Fehlermeldung abgebrochen.

Signatur für Runtime- und Geräteoptionen

Wenn bei der Signaturprüfung die Signatur als ungültig erkannt wurde, informiert eine Fehlermeldung Sie darüber. In ProSave wird ebenfalls eine Fehlermeldung ausgegeben, die auf die Ungültigkeit der Signatur hinweist.

Hinweis

Um das Laden der Bediengerätesoftware oder der Optionen ohne Signatur (für Versionen vor V14) dennoch zu ermöglichen, müssen Sie die Überprüfung der Signatur ausschalten.

Weitere Vorgehensweise finden Sie unter Überprüfung der Signatur ausschalten (Seite 2336).

Siehe auch

Überprüfung der Signatur ausschalten (Seite 2336)

15.3.16.2 Herunterladen von Optionen der Drittanbieter

Grundlagen

Standardmäßig ist das Herunterladen der Optionen der Drittanbieter aufgrund einer fehlenden Signatur nicht erlaubt. Um das Herunterladen der Optionen der Drittanbieter dennoch zu ermöglichen, müssen Sie die Überprüfung der Signatur für die Zeit des Herunterladens ausschalten.

Weitere Vorgehensweise finden Sie unter Überprüfung der Signatur ausschalten (Seite 2336).

Hinweis

Es wird empfohlen, nach dem Herunterladen der Optionen die Signaturüberprüfung wieder einzuschalten.

Siehe auch

Überprüfung der Signatur ausschalten (Seite 2336)

15.3.16.3 Überprüfung der Signatur ausschalten

Vorgehensweise

1. Tippen Sie im Control Panel des Bediengeräts zwei Mal auf "Transfer".
   Der Dialog "Transfer Settings" wird geöffnet.
   Standardmäßig ist die Option "Validate signatures" aktiviert.

   Hinweis

   Wenn Sie ein älteres WinCC Projekt auf ein Bediengerät laden, müssen Sie die Einstellungen zur Überprüfung der Signatur beachten.

2. Um das Laden der Bediengerätesoftware ohne Signatur oder das Herunterladen einer vertrauenswürdigen Option eines Drittanbieters zu ermöglichen, deaktivieren Sie die Option "Validate signatures".

Siehe auch

Herunterladen von Optionen der Drittanbieter (Seite 2335)

Grundlagen zu Signaturen (Seite 2335)

15.4 Runtime Professional

15.4.1 Einstellungen für Runtime

15.4.1.1 Einstellungen der Runtime-Software

Einleitung

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen".

Hinweis

Geräteabhängigkeit

Die Auswahl an Optionen in den Einstellungen für Runtime ist abhängig vom Bediengerät.

Übersicht

In den Runtime-Einstellungen projektieren Sie Optionen aus verschiedenen Bereichen Ihres Projekts:

- Darstellung in Runtime
  - Startbild
  - Menüs & Symbolleisten
  - Unabhängige Bildfenster
  - Designs
  - Eigenschaften von Fenstern
  - Bitselektion für Farbe und Blinken
  - Bildpuffer
  - Grafiksammlung
- Skripte in Runtime
  - Starten auch mit fehlerhaftem Skript
  - Wertequelle für SmartTags
  - Sprache der C-Skripte mit "Geräteeinstellung verwenden"
- Optionen zum Übersetzen von Projekten
  - Gelöschte HMI-Variablen in Bildern nicht prüfen: Wenn deaktiviert, wird bei gelöschten HMI-Variablen in Bildern eine Warnung beim Übersetzen ausgegeben.
  - WebUX-Warnungen ausblenden

- Startsequenz
  Über die Startfrequenz legen Sie fest, welche Anwendungen beim Aktivieren des Projekts gestartet werden.
- Bedienung in Runtime
  - Fenstersteuerung
  - Verfügbarkeit der Online-Hilfe in Runtime
  - Bildschirmtastatur
  - Tastaturbelegung und Tastenkombinationen
  - Cursorsteuerung
  - Zoomfunktionen
- Meldungen
  - Dauer der Anzeigeunterdrückung
  - Meldelisteneinträge nach Stromausfall
  - OPC Meldungen und Ereignisse
  - Steuerungsmeldungen
- Archivierung
  - Aktualisierungszyklus
  - Eigenschaften der Archivsegmente
- Konfiguration der WinCC-OPC-Server
- Benutzerverwaltung
  - Dynamisches Anmelden
  - SIMATIC Logon
- Sprache und Schriftart
  - Verfügbare Runtime-Sprachen
  - Runtime-Sprache für nicht-multilinguale Objekte
  - Reihenfolge der Sprachumschaltung
  - Standardschrift
  - Zeichensatz für Menüs
- Web-Zugriff
  - Gestaltung
  - Auslastung des Servers
  - Lokale Benutzergruppen
  - Meldungen in der Ereignisanzeige
  - Lizenzen

  Weitere Informationen zu den Runtime-Einstellungen finden Sie in der Online-Hilfe zu den Optionen "WebNavigatior", "WebUX" und "DataMonitoring".

- Einstellungen zum Synchronisieren von PLC- und HMI-Variablen
- Redundanz
  - Redundante Server und Partner
  - Client-Umschaltung
  - Ausfallszeitraum für den Abgleich
  - Abgleich für Archive und Meldungen
  - Rezepturen

  Weitere Informationen zur Konfiguration von redundanten Servern finden Sie in der Online-Hilfe zur Option "Redundancy"
- Process Historian
  - Serverauswahl
  - Zwischenspeicherungspfad

  Weitere Informationen zu den Runtime-Einstellungen finden Sie in der Online-Hilfe zur Option "Process Historian und Information Server".

Siehe auch

Darstellung in Runtime projektieren (Professional) (Seite 2339)

Startsequenz einrichten (Seite 2349)

Dauer der Anzeigeunterdrückung einstellen (Seite 899)

Einstellungen für die Benutzerverwaltung (Seite 1381)

Einstellungen zum Synchronisieren (Seite 564)

Bedienung in Runtime projektieren (Seite 2341)

Arbeiten mit Archiven (Seite 1077)

Sprachen verwalten (Seite 2095)

15.4.1.2 Darstellung in Runtime projektieren (Professional)

Einleitung

In den Runtime-Einstellungen legen Sie fest, wie Startbild, unabhängige Bildfenster, Bilder, Menüs und Symbolleisten usw. dargestellt werden.

Um die Runtime-Einstellungen für Ihr Bediengerät zu bearbeiten, wählen Sie in der Projektnavigation unterhalb Ihres Bediengeräts "Runtime-Einstellungen". Die Auswahl an Optionen ist abhängig vom Bediengerät.

Einstellungen zum Startbild

Das Startbild ist das erste Bild, das nach Runtime-Start angezeigt wird. Die Bildschirmauflösung wird bei der Auswahl des Bildes automatisch an das Bediengerät angepasst. Startbild legen Sie in den Runtime-Einstellungen des Bediengeräts unter "Allgemein > Einstellungen" fest.

Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Startbild des Projekts definieren".

Einstellungen zum Design in den Runtime-Einstellungen

Unter "Allgemein > Einstellungen" legen Sie fest, ob das Bediengerät das Standard-Design des Projekts oder ein anderes vordefiniertes Design verwendet. Weitere Informationen dazu finden Sie in "Arbeiten mit Designs".

Einstellung zum Design in der Windows-Systemsteuerung

Wählen Sie in den Systemeinstellungen von Windows für den Projektierungs-PC und den Runtime-PC das Design "Windows 7". Mit dieser Einstellung stellen Sie die korrekte Darstellung der Dialoge und der Bildschirmtastatur in Runtime sicher.

Einstellungen zu Menüs und Symbolleisten

Unter "Bildverwaltung" projektieren Sie im Editor "Menüs & Symbolleisten" benutzerdefinierte Menüs und Symbolleisten.

Für Menüs und Symbolleisten nehmen Sie folgende Einstellungen vor:

- Text
- Grafik
- Daten: Diese Daten werden als Anwenderdaten an die Funktion übergeben.

Für Symbolleisten stehen Ihnen zusätzlich folgende Einstellungsmöglichkeiten zur Verfügung:

- Schaltflächentyp: Sie stellen ein, ob die Schaltfläche nur aus einer Grafik, nur aus Text oder aus beidem besteht.
- Bildgröße
- Transparente Farben: Sie legen den Farbwert fest, der in der ausgewählten Grafik durchsichtig angezeigt wird.

Weitere Informationen dazu finden Sie in der Online-Hilfe von WinCC unter "Übersicht zum Arbeiten mit Menüs und Symbolleisten".

Einstellungen zu unabhängigen Bildfenstern

Wenn Sie die Option "Haupt-Fenster ausblenden" unter "Allgemein > Unabhängige Bildfenster" aktivieren, legen Sie fest, dass Runtime im Hintergrund läuft. Das Fenster der Runtime wird ausgeblendet und ist auch nicht in der Taskleiste sichtbar.

Einstellungen für Bilder

Unter "Bilder" ändern Sie die Eigenschaften des Runtime-Fensters und legen Sie fest, wie zusätzlichen Funktionen verwendet werden.

Außerdem nehmen Sie allgemeine Einstellungen vor. Zu den allgemeinen Einstellungen zählen:

- In Runtime Hintergrundeffekte verwenden
- In Runtime Schatten für Bildobjekte verwenden
- In Runtime WinCC Klassik Design verwenden
- Bitselektion für die Gestaltung verwenden

Außerdem legen Sie unter Bildpuffer fest, wie viele Bilder zwischengespeichert werden. Die im Bildpuffer abgelegten Bilder werden mit den Bildnavigationstasten angezeigt.

Grafikeinstellungen

Unter "Grafikeinstellungen > Geladene Grafiksammlung" legen Sie fest, welche Grafiken aus der Grafiksammlung auf das Bediengerät geladen werden.

15.4.1.3 Bedienung in Runtime projektieren

Bedienung in Runtime projektieren (Professional)

Einleitung

Für die Bedienung in Runtime stehen Ihnen zahlreiche Funktionen zur Verfügung. Für einige dieser Funktionen können Sie bestimmte Tastenkombinationen definieren.

Tasten für die Fenstersteuerung

Die Änderung der Tastenkombination wird erst nach Neustart des Systems wirksam.

Fenster umschalten

Dieser Hotkey ermöglicht die Navigation zwischen mehreren Bildfenstern, die in einem Hauptbild projektiert sind. Durch Drücken des Hotkeys wird das jeweils nächste Bildfenster für eine Bedienung aktiviert.

Hinweis

Die Tasten <Entf> und <Pos1> können nicht als Hotkey verwendet werden.

Hilfe in Runtime verfügbar machen

Unter Runtime-Einstellungen > Allgemein legen Sie fest, dass die Online-Hilfe in Runtime verfügbar ist. Dazu aktivieren Sie die Option "Hilfe in Runtime verfügbar machen".

Einige Objekte in Runtime verfügen über eine Online-Hilfe. Schalten Sie den Aufruf der Online-Hilfen zentral ab. Dadurch verhindern Sie, dass in Runtime über die Online-Hilfe auf die Festplatte zugegriffen wird.

Zugriff auf Windows in Runtime verhindern

Um zu verhindern, dass der Anwender über bekannte Tastenkombinationen, z. B. <Strg> +<Alt>+<Entf>, auf die Windows-Oberfläche zugreift, aktivieren Sie unter "Runtime-Einstellungen > Tastatur > Tastatur-Befehle des Betriebssystems" die Funktion "Tastatur-Befehle deaktivieren". Deaktivieren Sie in Windows zusätzlich die Einstellung "Taskleiste immer im Vordergrund halten".

Bildschirmtastatur verwenden

Wenn Sie unter Allgemein die Option "Bildschirmtastatur verwenden" aktivieren, wird in Runtime die Bildschirmtastatur dargestellt.

Um Daten in Runtime einzugeben, wird am Bildschirm eine Bildschirmtastatur eingeblendet. Der Bediener drückt die einzelnen Tasten mit der Maus oder direkt über Touch-Screen.

Hinweis

Die Bildschirmtastatur ist nicht bei allen Bediengeräten verfügbar.

Tastatur-Befehle deaktivieren

Die Funktion „Tastatur-Befehle deaktivieren" wird von der über Windows aktivierbaren Bildschirmtastatur nicht unterstützt. Verwenden Sie stattdessen die von WinCC Runtime Professional angebotene Bildschirmtastatur.

Auch mit fehlerhaften Skripten starten

Mithilfe der Option "Auch mit fehlerhaften Skripten starten" legen Sie fest, dass Runtime Professional gestartet wird, auch wenn eine Funktion im Projekt fehlerhaft ist.

Namen laden

Unter Allgemein legen Sie mithilfe der Option "Namen laden" fest, dass die Namensinformationen der Objekte mit geladen werden.

Wenn das Optionskästchen aktiviert ist, werden die Namensinformationen von Objekten auf das Bediengerät transferiert.

Wenn das Optionskästchen deaktiviert ist, werden anstelle der Namensinformationen lediglich codierte Adressierungsinformationen auf das Bediengerät transferiert.

Die Namensinformationen der Objekte benötigen Sie, wenn Sie die Objekte in einem Skript über den Objektnamen ansprechen wollen. Wenn Sie ein Skript im Debugger testen, wird der Code durch die Darstellung der Objektnamen übersichtlicher.

Hauptfenster ausblenden

Mit dem Objekt "Bildfenster" werden andere Bilder des Projekts im aktuellen Bild dargestellt. Unter "Allgemein > Unabhängige Bildfenster" finden Sie die Funktion "Hauptfenster ausblenden". Mit dieser Option legen Sie fest, dass Runtime im Hintergrund läuft. Das Fenster der Runtime wird ausgeblendet und ist auch nicht in der Taskleiste sichtbar.

Tastaturbelegung wählen (Professional)

Einleitung

Zum Bedienen des Projekts können Sie Tastenkombinationen einstellen. Die Tastenkombinationen lassen sich als Alternative zur Mausbedienung nutzen.

- Tastatur-Befehle des Betriebssystems
  Im Bereich "Tastatur-Befehle des Betriebssystems" legen Sie fest, ob mit Tastenkombinationen auf das Betriebssystem zugegriffen wird.
- Tasten für Aktionen
  - "Anmelden" öffnet ein Fenster zum Anmelden eines Benutzers in Runtime.
  - "Abmelden" öffnet ein Fenster zum Abmelden eines Benutzers in Runtime.
  - "Hardcopy" öffnet den Dialog zum Drucken des Bildschirms in Runtime.
- Windows-Steuerungstasten
  Im Bereich "Windows-Steuerungstasten" legen Sie fest, ob mithilfe von einer festgelegten Tastenkombination in Runtime das nächste Bildfenster aktiviert wird.
- Bildnavigationstasten
  - "Vorwärts" wechselt zum nächsten Bild in der projektierten Bildreihenfolge.
  - "Zurück" wechselt zum vorherigen Bild in der projektierten Bildreihenfolge.
  - "Bild merken" speichert das aktuell angezeigte Bild.
  - "Gemerktes Bild aufrufen" wechselt zum zuletzt gespeicherten Bild.
  - "Startbild" wechselt zum Startbild.
- Tasten für TAB-Reihenfolge
  - "Modus für TAB-Reihenfolge" schaltet den Cursor-Modus um.
  - "Fokusanzeige des aktiven Objekts" aktiviert den Runtime-Cursor.
- Cursor-Steuerungstasten
  - "Hotkeys in Runtime aktivieren" aktiviert die Tastenkombinationen für die Cursorsteuerung in Runtime.
  - "Auf" führt zu einer Cursorbewegung nach oben.
  - "Ab" führt zu einer Cursorbewegung nach unten.
  - "Links" führt zu einer Cursorbewegung nach links.
  - "Rechts" führt zu einer Cursorbewegung nach rechts.

Voraussetzung

Der Editor "Runtime-Einstellungen" ist geöffnet.

Vorgehen

Um die Tastenkombinationen einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie "Runtime-Einstellungen > Tastatur.
2. Wählen Sie z. B. im Bereich "Tasten für Aktionen" die gewünschte Aktion.

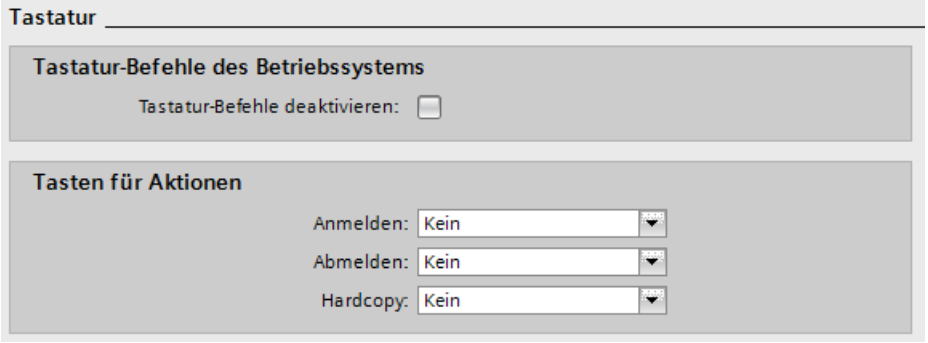

3. Öffnen Sie die Klappliste.
   Der Dialog zum Festlegen der Tastenkombination wird geöffnet.
4. Legen Sie die gewünschte Tastenkombination fest.

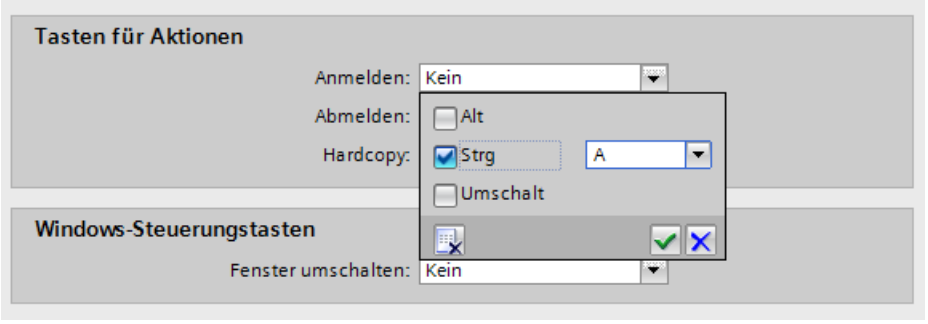

5. Übernehmen Sie die Einstellungen.
   Der Dialog zum Festlegen der Tastenkombination schließt sich.

Tastenkombination für Bedienung und Bildnavigation festlegen

Einleitung

Bilder können auch dann bedient werden, wenn eine Steuerung per Maus nicht verfügbar ist. Dazu legen Sie für die benötigten Bedienfunktionen Tastenkombinationen fest. Wenn der Bediener die Tastenkombination betätigt, wird die gewünschte Funktion aufgerufen.

Die wichtigsten Tastenkombinationen für die mauslose Bedienung in Runtime legen Sie im Editor "Runtime-Einstellungen > Tastatur" fest.

Vorgehen

Um die Tastenkombinationen für die Bedienung festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Eingabefeld der gewünschten Funktion.
2. Klicken Sie im Auswahlfeld auf die Schaltfläche . Der Dialog zum Festlegen der Tastenkombination wird geöffnet.
3. Legen Sie die gewünschte Tastenkombination fest.
4. Schließen Sie den Dialog über die Schaltfläche .

Alternativ können Sie statt einer Tastenkombination auch eine der Funktionstasten <F1> - <F12> angegeben.

Hinweis

Die Funktionstaste F12 darf nicht als systemweiter Hotkey projektiert werden.

Tasten für die Fenstersteuerung

Die Änderung der Tastenkombination wird erst nach Neustart des Systems wirksam.

Fenster umschalten

Dieser Hotkey ermöglicht die Navigation zwischen mehreren Bildfenstern, die in einem Hauptbild projektiert sind. Durch Betätigung des Hotkeys wird das jeweils nächste Bildfenster für eine Bedienung aktiviert.

Hinweis

Die Tasten <ENTF> und <POS1> können nicht als Hotkey verwendet werden.

Bildnavigationstasten

Die Schaltflächen "Vorwärts" und "Rückwärts" erlauben in gängigen Internet-Browsern das Umschalten zwischen Internetseiten, die kurz zuvor aufgerufen worden sind. Mit einer vergleichbaren Technik realisieren Sie in WinCC die Navigation zwischen einzelnen Prozessbildern.

Im Bereich "Bildauswahl" projektieren Sie Tastenkombinationen, die Ihnen in Runtime einen einfachen Wechsel zwischen den zuletzt aufgerufenen Bildern ermöglichen. Bilder werden bei Aufruf in einer temporären Liste registriert. Mit den Funktionen "Vorwärts" oder "Rückwärts" blättern Sie schrittweise in dieser Liste, z. B. die letzten fünf Bilder erneut aufzurufen. Die maximal erlaubte Anzahl von Einträgen in dieser Liste wird in dem Feld "Speicher Bildauswahl"

festgelegt. Wird diese Anzahl überschritten, ersetzt der Aufruf eines neuen Bilds den jeweils ältesten Eintrag.

Hinweis

Mit der Default-Einstellung für die "Speicher Bildauswahl" werden maximal 30 Bildaufrufe registriert. In der Praxis ist es meist nicht sinnvoll, diesen Wert wesentlich zu erhöhen, da die Umschaltung nur in Einzelschritten erfolgen kann. Eine noch effizientere Bildnavigation projektieren Sie mit der Bildbausteintechnik.

Cursorsteuerung projektieren (Professional)

Einleitung

In Bildern mit tabellenartiger Anordnung der Objekte kann mithilfe einer definierten Cursorsteuerung eine mauslose Navigation zwischen den projektierten Objekten erreicht werden.

Die Hotkeys für die Cursorsteuerung werden im Editor "Runtime-Einstellungen > Tastatur" festgelegt. Das Verhalten des Cursors in Tabellen legen Sie unter "Bilder > Cursor-Verhalten in Tabellen" fest.

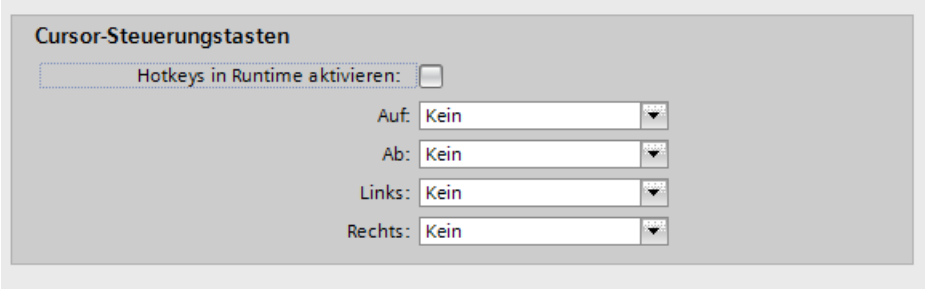

Vorgehen

Um die Tastenkombinationen für die Bedienung festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Eingabefeld der gewünschten Funktion.
2. Öffnen Sie die Klappliste.
   Der Dialog zum Festlegen der Tastenkombination wird geöffnet.
3. Legen Sie die gewünschte Tastenkombination fest.
4. Übernehmen Sie die Einstellungen.
   Der Dialog zum Festlegen der Tastenkombination schließt sich.

Alternativ können Sie statt einer Tastenkombination auch eine der Funktionstasten <F1> - <F12> angegeben.

Zoomfunktionen aktivieren (Professional)

Einleitung

Das Zoomen in Runtime wird durch drei Techniken unterstützt:

- Aufräumen
  Ebenen und die darin abgelegten Objekte können eingeblendet und ausgeblendet werden. Nähere Hinweise hierzu finden Sie unter "Sichtbarkeit eines Objekts dynamisieren (Seite 226)".
- Erweitertes Zoomen
  Die Ansicht eines Bildes in Runtime kann durch Einsatz des Mausrades vergrößert oder verkleinert werden. Halten Sie dazu <Strg> gedrückt und bewegen Sie das Mausrad in die gewünschte Richtung.
- Panning
  Wenn ein Bild mit einem Zoomfaktor versehen ist, der das Bild mit Bildlaufleisten anzeigt, können Sie einen Bildausschnitt im Dokument verschieben. Bei einem Klick auf das Mausrad erscheint ein Navigationskreuz. Durch Bewegen des Mauspfeils wird dann in die gewünschte Richtung gescrollt. Dabei bestimmt der Abstand des Mauspfeils zum Navigationskreuz die Scrollgeschwindigkeit. Bei einem weiteren Klick wird das Panning wieder deaktiviert.

Voraussetzung

- Einen Maustreiber von Logitech oder Microsoft-Intellimouse
- Das Mausrad muss auf "Autoscroll" eingestellt sein.
- Der Editor "Runtime-Einstellungen" ist geöffnet.

Zoomfunktion aktivieren

1. Klicken Sie iauf "Bilder".
2. Deaktivieren Sie "Aufräumen" und "Erweitertes Zoom".
   Wenn Sie "Aufräumen" und "Erweitertes Zoomen" aktivieren, sind die Funktionen ausgeschaltet.

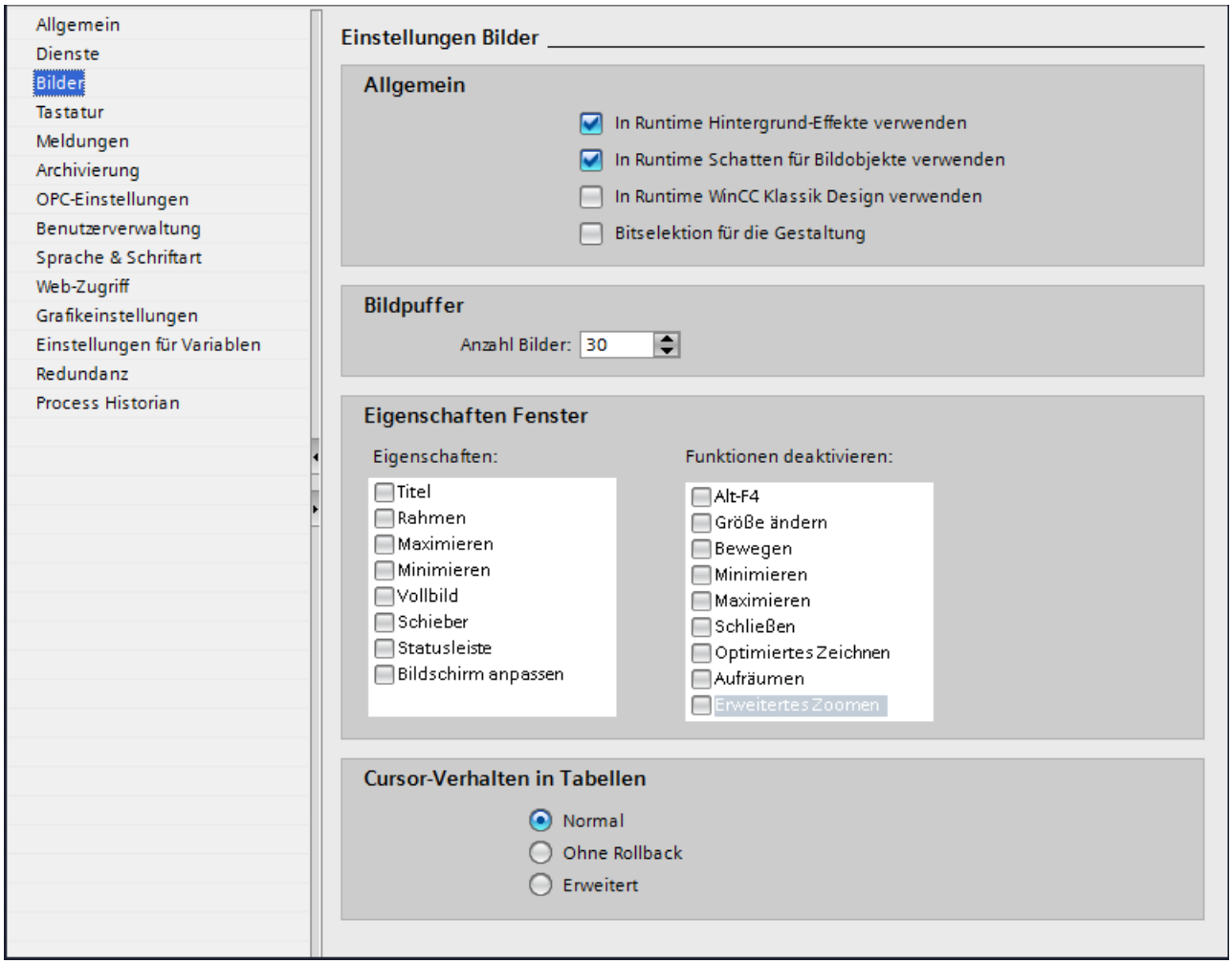

Erweitertes Zoomen bildspezifisch projektieren

Die Funktion "Erweitertes Zoomen" können Sie bildspezifisch ein- bzw- ausschalten. Die Einstellung legen Sie im Inspektorfenster eines Bildes unter "Eigenschaften > Eigenschaften > Verschiedenes" fest.

Siehe auch

Sichtbarkeit eines Objekts dynamisieren (Seite 226)

15.4.1.4 Startsequenz einrichten

Einleitung

Wenn Sie ein Projekt aktivieren, werden zusätzliche Programm-Module geladen, die für die Ausführung von Runtime benötigt werden.

Bei "Startsequenz von WinCC Runtime" legen Sie fest, welche Anwendungen beim Aktivieren eines Projekts gestartet werden. "Grafik in Runtime" wird immer gestartet und ist als Standard-Einstellung aktiviert.

Um eine hohe Performance zu erhalten, starten Sie nur die benötigten Anwendungen in Runtime.

Vorgehen

Um die Anlaufliste einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Editor "Runtime-Einstellungen > Dienste".

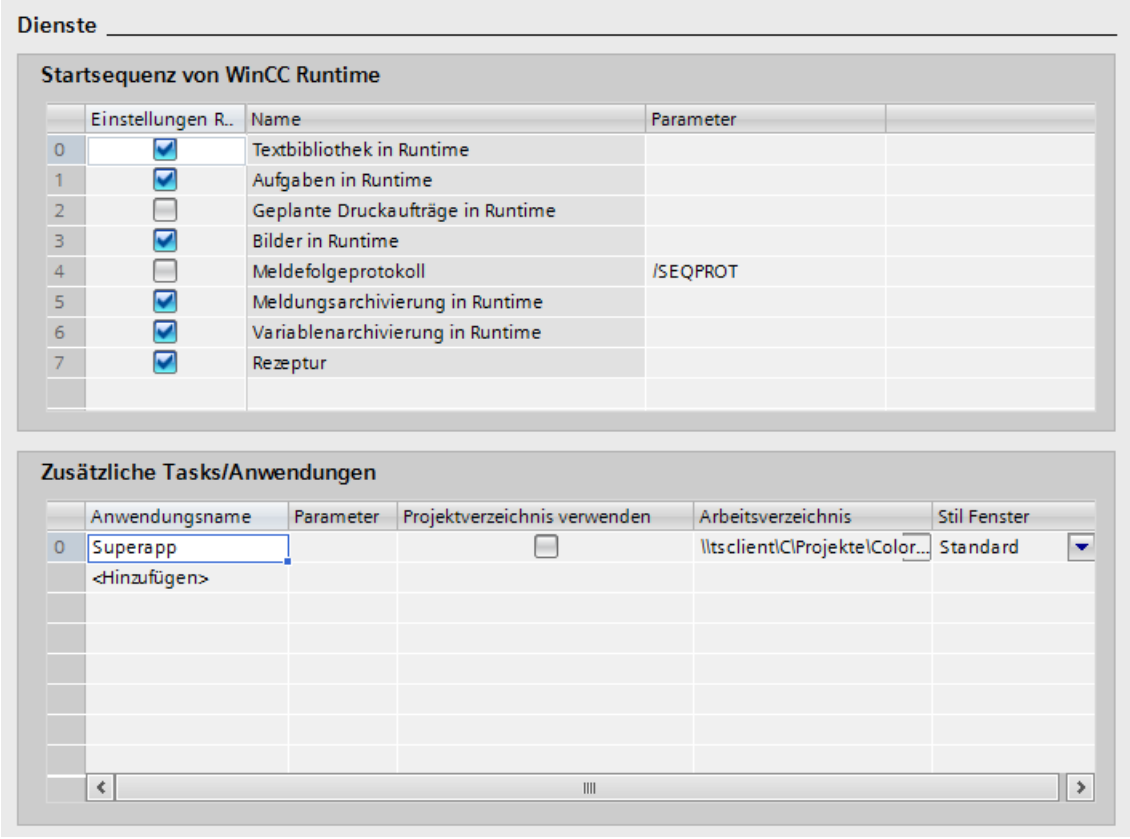

2. Um die Onlinehilfe in Runtime anzuzeigen, aktivieren Sie "Hilfe in Runtime verfügbar machen".
3. Aktivieren Sie unter "Startsequenz von WinCC Runtime" die Anwendungen, die beim Starten von Runtime geladen werden.

4. Wenn Sie zusätzliche Tasks oder Anwendungen beim Starten von Runtime öffnen wollen, klicken Sie in die Tabellenzeile bei "Zusätzliche Tasks/Anwendung".
5. Geben Sie einen Namen für die zusätzliche Task oder Anwendung ein. Legen Sie das Arbeitsverzeichnis fest.
6. Wenn Sie in der Spalte "Projektverzeichnis" das Kontrollkästchen aktivieren, wird der Projektordner als Arbeitsverzeichnis verwendet.
7. Wenn Sie in der Spalte "Projektverzeichnis" das Kontrollkästchen deaktivieren, legen Sie in der Spalte "Arbeitsverzeichnis" über die Schaltfläche "Durchsuchen" den Ordner fest.
8. Legen Sie gegebenenfalls die Parameter fest.
9. Legen Sie in der Spalte "Stil Fenster" fest, ob das Fenster mit der Task oder Anwendung minimiert, maximiert oder in der Standardgröße geöffnet wird.

15.4.1.5 Uhrzeitsynchronisation aktivieren

Einleitung

Um anlagenweit dieselbe Uhrzeit zu haben, synchronisieren Sie die Uhrzeit der unterschiedlichen Komponenten der Anlage mithilfe der Uhrzeitsynchronisation. Die Uhrzeitsynchronisation von WinCC wird als Master-Slave-System betrieben.

Damit alle Komponenten einer Anlage mit einer identischen Uhrzeit arbeiten, muss eine Systemkomponente Zeitgeber für alle anderen Komponenten sein. Die als Zeitgeber fungierende Komponente wird als Uhrzeit-Master bezeichnet. Die zeitempfangenden Komponenten sind die Uhrzeit-Slaves.

Uhrzeit-Master kann ein Computer oder ein externer Zeitgeber, z. B. eine SICLOCK, sein. Dieser Master sendet das Zeitsignal an die anderen Komponenten. Im Netz kann immer nur ein Uhrzeit-Master aktiver Zeitgeber sein.

Um den Ausfall einer Uhrzeitsynchronisation zu verhindern, können zwei Komponenten als Uhrzeit-Master konfiguriert werden. Der redundante Uhrzeit-Master ist so lange Uhrzeit-Slave, wie der aktive Uhrzeit-Master Zeittelegramme sendet.

Die Uhrzeit-Slaves empfangen das vom Uhrzeit-Master gesendete Zeitsignal über den Bus und synchronisieren damit ihre eigene Uhrzeit.

Uhrzeitsynchronisation in Runtime

Bei folgenden Ereignissen wird die Uhrzeitsynchronisation in die Startsequenz des Rechners eingetragen:

- nach der Konfiguration der Uhrzeitsynchronisation

Hinweis

Die Uhrzeitsynchronisation wird wenige Minuten nach dem Aktivieren in Runtime wirksam. Wenn Sie eine exakte Zeitsynchronisation während des Hochlaufes benötigen, synchronisieren Sie die lokalen PC-Uhren bereits vor der Aktivierung des Runtime. Das ist z. B. der Fall, um Leittechnikmeldungen im Zuge des Hochlaufs zeitfolgerichtig sortiert zu erhalten.

Uhrzeitsynchronisation aktivieren

Um die Uhrzeitsynchronisation zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation im Kontextmenü der Runtime "Eigenschaften" und wählen Sie "Uhrzeitsynchronisation > Allgemein".
2. Aktivieren Sie unter "Synchronisation" "Uhrzeitsynchronisation aktivieren".

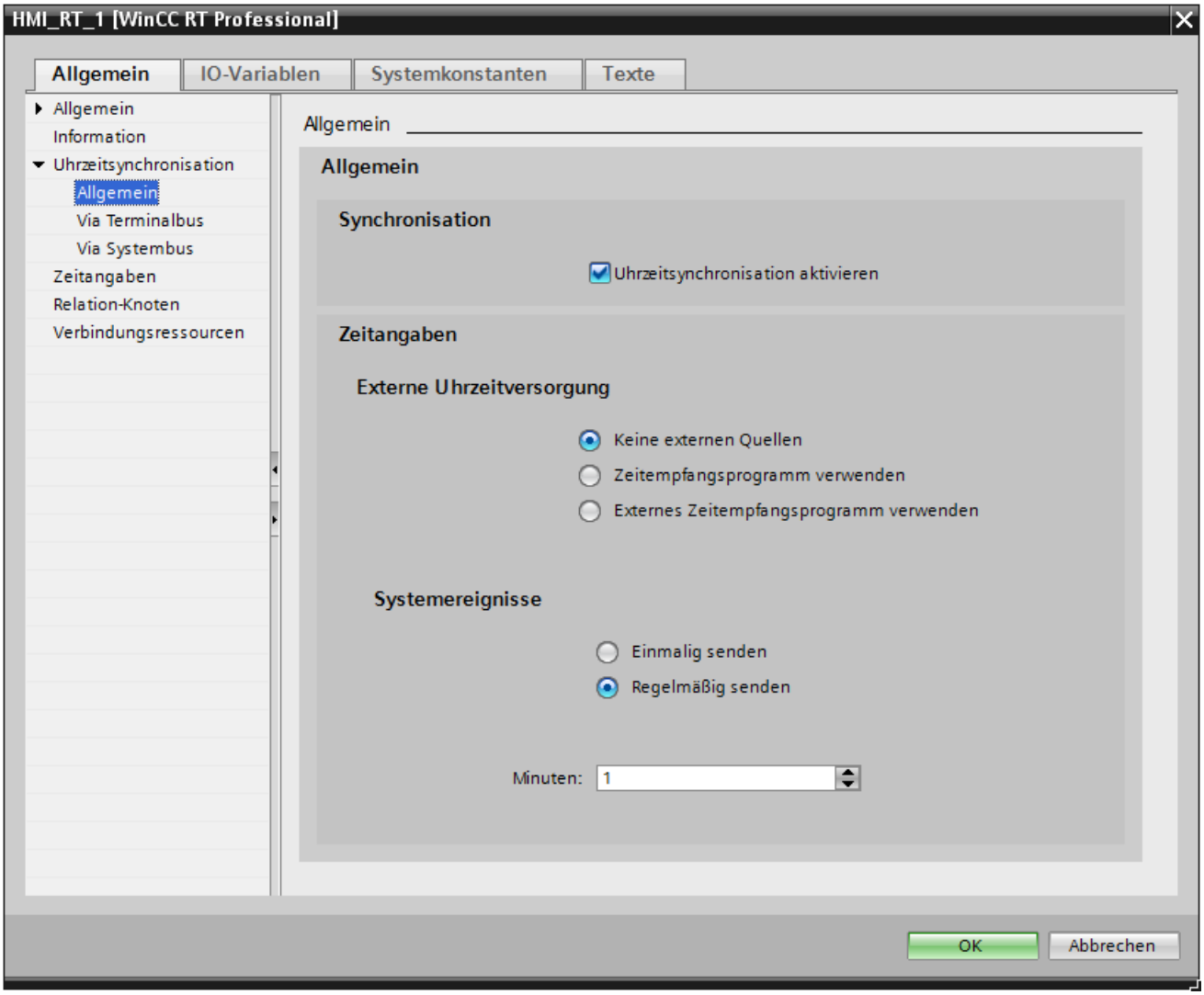

3. Wählen Sie unter "Zeitangaben" eine Uhrzeitquelle, z.B. "Keine externen Quellen".
4. Geben Sie an, wie häufig die Zeittelegramme vom Uhrzeit-Master gesendet werden sollen.

Synchronisation über Terminalbus

Um die Uhrzeit über den Terminalbus zu synchronisieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation im Kontextmenü der Runtime "Eigenschaften" und wählen Sie "Uhrzeitsynchronisation > Via Terminalbus".

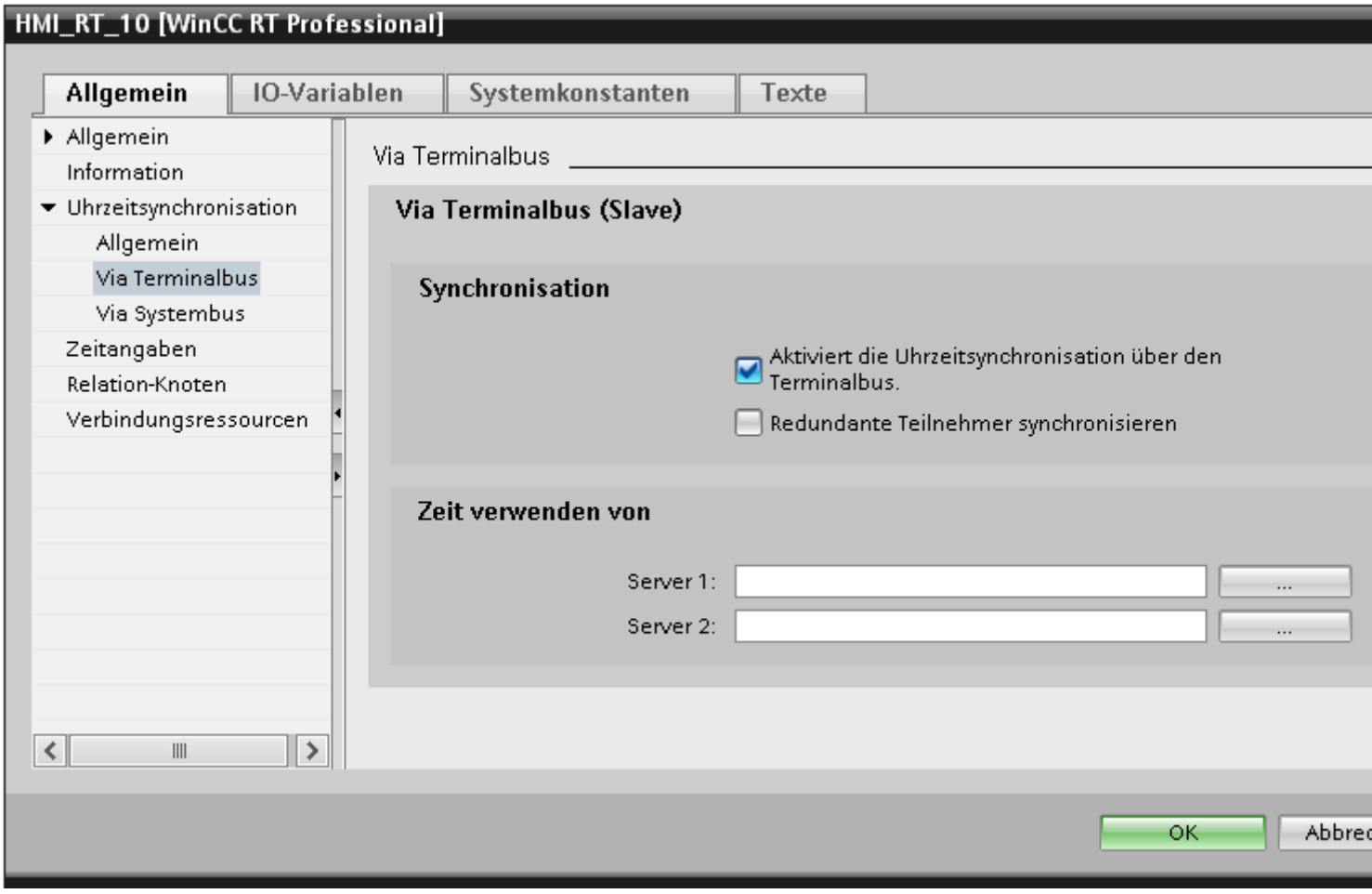

2. Aktivieren Sie unter "Synchronisation" das Optionskästchen "Aktivieren".
3. Wenn Sie auch redundante Komponenten synchronisieren wollen, aktivieren Sie das Optionskästchen "Redundante Teilnehmer synchronisieren".
4. Wählen Sie unter "Server 1" den Server aus, von dem das Bediengerät als Uhrzeit-Slaves die Zeittelegramme erhalten sollen.
5. Klicken Sie auf "OK" oder konfigurieren Sie noch die Synchronisation über den Systembus.

Synchronisation über Systembus

Um die Uhrzeit über den Systembus zu synchronisieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation im Kontextmenü der Runtime "Eigenschaften" und wählen Sie "Uhrzeitsynchronisation > Via Systembus".

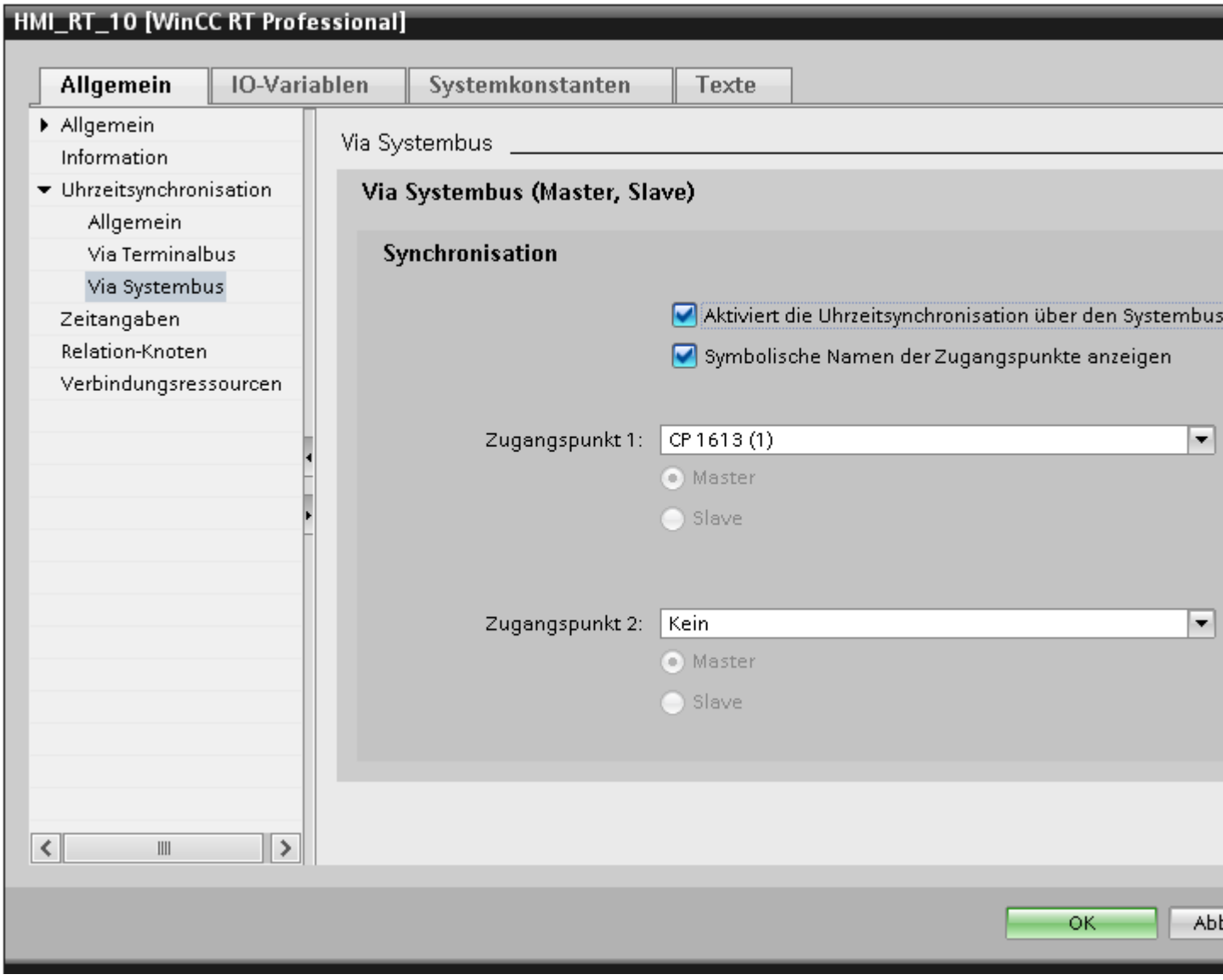

2. Aktivieren Sie unter "Synchronisation" das Optionskästchen "Aktivieren".
3. Aktivieren Sie das Optionskästchen "Symbolische Namen der Zugangspunkte anzeigen".

Hinweis

Wenn Sie von einem anderen Computer die Uhrzeitsynchronisation konfigurieren, sind die installierten Zugangspunkte nicht bekannt. Die symbolischen Namen werden jedoch angezeigt und bei Hochlauf in Runtime den physikalischen Namen der Zugangspunkte zugeordnet.

4. Wählen Sie unter "Zugangspunkt1" einen Zugangspunkt aus und weisen Sie ihm eine Rolle als Master oder als Slave zu.
5. Unter folgenden Bedingungen wählen Sie auch einen zweiten Zugangspunkt aus:
   – Wenn Sie ein redundantes Gerät eingesetzt wollen
   – Wenn Sie zwei Systembusse über eine "Brücke" miteinander synchronisieren
   – Wenn Sie zwei Systembusse gleichzeitig als Master synchronisieren
6. Weisen Sie auch diesem Zugangspunkt eine Rolle als Master oder als Slave zu. Aus der Kombination dieser Rollen ergibt sich das Verhalten wie in der Tabelle unten dargestellt.

Zugangspunkt1	Zugangspunkt2	Verhalten
Slave	Slave	Das Gerät arbeitet immer als Uhrzeit-Slave.
Master	Master	Das Gerät arbeitet als kooperativer Master auf beiden Zugangspunkten.
Master	Slave	Das Gerät dient am Zugangspunkt1 als Master und sendet Zeittelegramme; am Zugangspunkt2 synchronisiert sich das Gerät über die dort empfangenen Zeitsignale.
Slave	Master	Das Gerät dient am Zugangspunkt2 als Master und sendet Zeittelegramme; am Zugangspunkt1 synchronisiert sich das Gerät über die dort empfangenen Zeitsignale.

15.4.1.6 Uhrzeitbasis einstellen

Zeitbasis für Uhrzeit einstellen

Um die Uhrzeitdarstellung am Bediengerät einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation im Kontextmenü der Runtime "Eigenschaften > Zeitangaben".

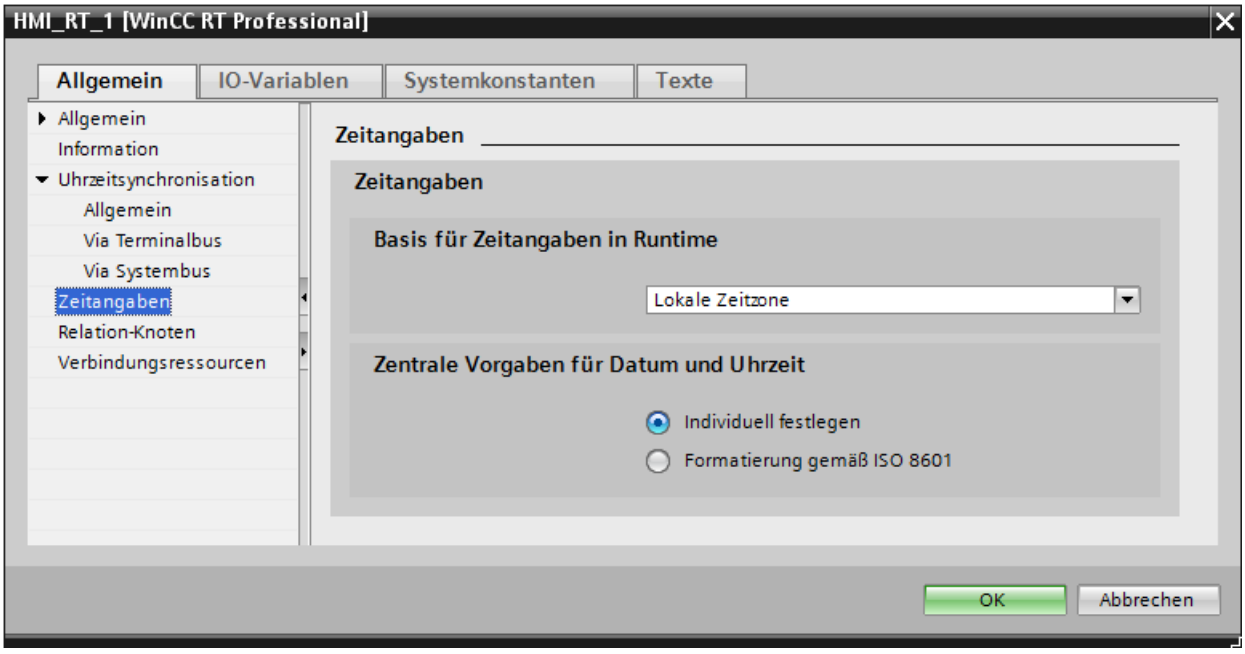

2. Wählen Sie unter "Zeitbasis für Zeitangaben in Runtime" eine der folgenden Möglichkeiten:
   - Zeitzone des Servers (migrierte Projekte)
   - Lokale Zeitzone
   - Koordinierte Weltzeit (UTC)
3. Wählen Sie für das Anzeigeformat für Datum und Uhrzeit eine der folgenden Optionen:
   - "Individuell festlegen": Wenn Sie das Datum im projektierten Format anzeigen wollen.
   - "Formatierung gemäß ISO 8610": Wenn Sie an allen Komponenten das Datum im ISO 8610-Format anzeigen wollen.

15.4.1.7 Einstellung der Persistenz

Einleitung

Über die Persistenz bestimmen Sie, wie lange Projektierungsänderungen beibehalten werden, die am Bediengerät direkt in Runtime projektiert wurden.

Die in Runtime projektierten Änderungen werden im Engineering System gesondert zum Bild gespeichert. Dadurch wird das Bild in seiner ursprünglichen Projektierung beibehalten.

Voreinstellungen der Persistenz

Die Persistenz lässt sich im Engineering System und je nach Projektierung in Runtime einstellen. Es gibt drei Voreinstellungen für Persistenz:

- "Keine Persistenz": Die Online-Konfigurierungen gehen beim nächsten Bildwechsel und beim Deaktivieren/Aktivieren der Projektierung verloren.
- "Persistenz": Diese Voreinstellung lässt dem Benutzer in Runtime die Optionen "verwerfen", "beibehalten" oder "zurücksetzen". Mit der Option "beibehalten" werden die Online-Konfigurierungen beim nächsten Bildwechsel und beim Deaktivieren/Aktivieren des Projekts beibehalten.
- "Persistenz in Runtime": Diese Voreinstellung lässt dem Benutzer in Runtime die Optionen "verwerfen", "beibehalten" oder "zurücksetzen". Mit der Option "beibehalten" werden die Online-Konfigurationen beim nächsten Bildwechsel beibehalten, gehen aber beim Deaktivieren/Aktivieren des Projekts verloren.

Hinweis

Änderung eines Controls

Wenn Sie ein projektiertes Control ändern und auf das Bediengerät laden, werden die Änderungen in Runtime unabhängig von den Einstellungen überschrieben.

Online-Deltaladen

Mit WinCC bearbeiten Sie ein Projekt auf dem Projektierungs-PC, während dasselbe Projekt in Runtime läuft. Wenn Sie das Projekt auf dem Projektierungs-PC ändern, können Sie bei den meisten Projektierungsänderungen die geänderten Anteile mit "Online-Deltaladen" auf das Bediengerät laden. Das Online-Deltaladen benötigt weniger Zeit als das komplette Laden des Projekts.

Beim Online-Deltaladen wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog legen Sie z. B. fest, ob Benutzerverwaltung oder Archive, die zwischenzeitlich am Bediengerät geändert wurden, beim Online-Deltaladen überschrieben oder behalten werden.

Hinweis

Änderungen am Bediengerät

Zwischenzeitliche Änderungen in der Benutzerverwaltung oder in den Archiven gehen beim Online-Deltaladen verloren.

15.4.1.8 Mit Service-Mode arbeiten

Grundlagen zu Service-Mode und Autostart

Übersicht

Der Service-Mode bietet Ihnen die Möglichkeit, WinCC Runtime Professional als Dienst zu betreiben. Als Dienst kann WinCC Runtime Professional auch dann aktiv sein, wenn kein interaktiver Benutzer am Rechner angemeldet ist.

Im Zusammenhang mit der Option "Autostart" kann der Service-Mode beispielsweise bei einem Stromausfall hilfreich sein, weil das System automatisch wieder hochgefahren wird und der Server für die Clients wieder verfügbar ist.

WinCC Runtime Professional ist auch dann auf dem Rechner ablauffähig, wenn kein interaktiver Benutzer am Rechner angemeldet ist.

WinCC Runtime Professional im Service-Mode kann auch mit angemeldetem Benutzer betrieben werden, dann sind interaktive Benutzereingaben möglich.

Hinweis

WinCC nicht ablauffähig bei Eingriffen im System

Änderungen der Prozesse und Dienste von WinCC in der Systemsteuerung und im Windows Taskmanager sind nicht erlaubt. Folgende Änderungen sind betroffen:

- Änderungen der Eigenschaften
- Manuelle Eingriffe:
  - Starten
  - Beenden
  - Anhalten
  - Fortsetzen
  - Neu starten
- Änderung der Priorität

Zwischen den einzelnen Prozessen und Diensten existieren Abhängigkeiten.

Führen Sie keine Änderungen aus.

Siehe auch

Service-Mode konfigurieren (Seite 2360)

Einsatz eines Service-Projekts und Einschränkungen (Seite 2358)

Funktionsweise eines Service-Projekts (Seite 2359)

Konfigurationen für ein Service-Projekt

Übersicht

WinCC Runtime Professional ist in folgenden Konfigurationen auf dem Server im Service-Mode als Service-Projekt ablauffähig:

- WinCC-Server mit Windows Server 2008 R2 SP1 (64-Bit)
  - Clients mit Windows 7 SP1 (32-Bit, 64-Bit)
- WinCC-Server mit Windows Server 2012 R2 (64-Bit)
  - Clients mit Windows 8.1 (32-Bit, 64-Bit)
- WinCC WebNavigator Server
  - WinCC Web-Clients
- DataMonitor-Server
  - DataMonitor-Clients

Einsatz eines Service-Projekts und Einschränkungen

Einsatz eines Service-Projekts

Auf dem Server wird das Projekt im Service-Mode als Service-Projekt betrieben. WinCC Runtime Professional startet als Dienst. Ein Service-Projekt wird automatisch oder manuell gestartet.

Ein Service-Projekt ist für folgende Einsätze verfügbar:

- Betrieb ohne angemeldeten Benutzer
  Ein Service-Projekt ist ablauffähig, ohne dass ein interaktiver Benutzer am Rechner angemeldet ist. Wenn kein interaktiver Benutzer angemeldet ist, ist keine interaktive Bedienung möglich.
- Betrieb mit angemeldetem Benutzer
  In Service-Projekten ist eine interaktive Bedienung in der Regel nicht gewünscht. Ein interaktiver Benutzer kann sich z. B. für Servicezwecke anmelden. In diesem Fall kann der Benutzer die interaktive Bedienung des Service-Projekts aktivieren.
- Automatischer Start
  Beim automatischen Start wird WinCC Runtime Professional beim Einschalten des Servers automatisch gestartet und das eingestellte Projekt aktiviert. Der automatische Start wird durchgeführt, ohne dass ein interaktiver Benutzer angemeldet ist.
- Manueller Start
  Beim manuellen Start muss sich der Benutzer am Server anmelden und danach das Projekt aktivieren. Wenn sich der Benutzer vom Server wieder abmeldet, ist WinCC Runtime Professional weiterhin aktiv.
- Anmelden und Abmelden der Benutzer
  Während das Service-Projekt aktiviert ist, können sich interaktive Benutzer jederzeit am Server anmelden und abmelden.

Einschränkungen

Ein Service-Projekt unterliegt folgenden Einschränkungen:

- VB-Skripte und C-Skripte führen zu Problemen, wenn Sie Interaktionen, z. B. Eingaben, erfordern oder Messageboxen anzeigen.
  Im Service-Mode besteht außerdem kein gemeinsamer Datenbereich für C-Scripting. Daher können beispielsweise keine globalen C-Variablen zwischen benutzerdefinierten C-Funktionen im Aufgabenplaner und benutzerdefinierten Funktionen im Editor "Bilder" ausgetauscht werden.
- Zusätzliche Aufgaben oder Anwendungen können beim Starten von Runtime nicht geladen werden.
- Der OPC-Zugriff ist nicht freigegeben.
- Diagnoseinformationen eines Service-Projekts können nicht am Server angezeigt werden.

Hinweis

Service-Projekt bearbeiten oder migrieren

Um ein Service-Projekt zu bearbeiten oder zu migrieren, müssen Sie auf dem Rechner den Service-Mode-Benutzer entsprechend administrieren. Wenn der Service-Mode-Benutzer nicht vorhanden ist, muss der eingeloggte Windows-Benutzer entsprechend administriert sein, damit das Projekt bearbeitet oder migriert werden kann.

Siehe auch

Grundlagen zu Service-Mode und Autostart (Seite 2357)

Funktionsweise eines Service-Projekts

Grundlagen

Bei einem Service-Projekt wird WinCC Runtime Professional als Dienst gestartet. Je nach Einstellung werden diese Dienste zu folgenden Zeitpunkten gestartet:

- Automatisch nach dem Starten des Betriebssystems.
- Nachdem sich ein Benutzer angemeldet und WinCC Runtime Professional gestartet hat.

WinCC Runtime Professional ist weiterhin aktiv, auch wenn sich der Benutzer wieder abmeldet.

Die Runtime-Daten sind weiterhin zugänglich.

Ein angemeldeter Benutzer kann bei Bedarf die Runtime-Bedienung aktivieren.

Siehe auch

Grundlagen zu Service-Mode und Autostart (Seite 2357)

Service-Mode konfigurieren

Einleitung

Im Dialog "Vorschau laden" legen Sie fest, ob das Projekt als Service-Projekt betrieben wird und ob für das Projekt Autostart eingerichtet wird.

Projekt als Service-Projekt konfigurieren

1. Aktivieren Sie im Dialog "Vorschau laden" die Option "Service-Mode aktivieren".
2. Geben Sie unter "Benutzer" Ihren Benutzernamen ein.
3. Geben Sie unter "Kennwort" Ihr Passwort ein.

Um sich anmelden zu können, müssen Sie in der SIMATIC HMI-Gruppe auf dem Bediengerät als Benutzer angelegt sein.

Hinweis

Im redundanten System müssen Sie sich in Runtime unbedingt mit denselben Benutzerdaten anmelden.

Hinweis

Wenn das Projekt geladen und der Service-Mode aktiviert wurde, ist die Option "Service-Mode aktivieren" bereits aktiviert.

Die Felder "Benutzer" und "Kennwort" sind nicht ausgefüllt.

Autostart einrichten

1. Aktivieren Sie im Dialog "Vorschau laden" die Option "Autostart aktivieren".
   WinCC aktiviert beim Hochfahren des Rechners das gewünschte Projekt.

Hinweis

Wenn das Projekt geladen und der Autostart aktiviert wurde, ist die Option "Autostart aktivieren" bereits aktiviert.

Besonderheiten beim Offline-Laden

Wenn der Projektierungs-PC keine Verbindung zum Bediengerät hat, laden Sie das Projekt offline über einen Datenträger auf das Bediengerät. Dazu kopieren Sie das Runtime-Projekt auf Ihr Bediengerät und starten in Runtime Professional das Projekt mit "WinCC RT Start". Das Runtime-Projekt wird von WinCC erkannt.

Anschließend können Sie den Autostart in Kombination mit "WinCC RT Start" für die Bedienung konfigurieren. Diese Konfiguration funktioniert nur, wenn das Runtime-Projekt nach dem Offline-Laden auf dem Bediengerät gestartet wurde.

Siehe auch

Grundlagen zu Service-Mode und Autostart (Seite 2357)

15.4.1.9 Kennwortschutz in Runtime

Kennwortschutz in Runtime: Grundlagen

Übersicht

Der Verbindungsaufbau kann im TIA Portal durch Kennwörter geschützt werden. Die konfigurierten Kennwörter werden verschlüsselt gespeichert. Beim Transfer ins Zielsystem werden die Kennwörter ebenfalls gegen nicht autorisierte Zugriffe geschützt und können nicht gelesen werden. Auf den Bediengeräten in Runtime werden diese Kennwörter sicher entschlüsselt.

Der Kennwortschutz kann für folgende Verbindungen angelegt werden:

- SIMATIC S7-1200
- SIMATIC S7-1500
- SIMATIC S7-1500 Software Controller (WinAC)
- PC Station 2.0 (Stationmanager)
- OPC-UA-Verbindung (ab V15.0)

Hinweis

Beim Wechsel zu einer älteren Version werden die Kennwörter ungültig.

Hinweis

Wenn in der V14 während der Projektierung der Benutzername lokal auf einem Bediengerät geändert wird, wird das Kennwort ungültig und muss erneut eingegeben werden.

Siehe auch

Kennwort für den Transport konfigurieren (Seite 2361)

Kennwort für den Transport zurücksetzen (Seite 2362)

Kennwort für den Transport konfigurieren

Einleitung

Um den privaten Kennwortschlüssel beim Transport zu schützen, können Sie ein Kennwort für den Transport unter "Runtime-Einstellungen" konfigurieren.

Kennwort anlegen

1. Öffnen Sie im Projektfenster den Editor "Runtime-Einstellungen > Allgemein".
2. Tragen Sie im Bereich "Transportkennwort" im Textfeld "Kennwort eingeben" das Kennwort ein.
3. Bestätigen Sie das Kennwort im Feld "Kennwort bestätigen".

Hinweis

Wenn kein altes Kennwort existiert, ist das Feld "Altes Kennwort" deaktiviert.

Ergebnis

Sie haben das Kennwort für den Transport konfiguriert. Wenn ein Kennwort für den Transport konfiguriert ist und Sie das Projekt auf das Bediengerät übertragen, erscheint in Runtime einmalig ein Dialog mit der Kennwortabfrage.

Wenn das eingegebene Kennwort korrekt ist, wird das Zertifikat im Zertifikatspeicher des Bediengeräts gespeichert.

Wenn Sie das Kennwort ändern, laden Sie das Projekt mit dem geänderten Kennwort auf das Bediengerät und starten Runtime. Es erscheint keine Kennwortabfrage.

Siehe auch

Kennwortschutz in Runtime: Grundlagen (Seite 2361)

Kennwort für den Transport zurücksetzen (Seite 2362)

Kennwort für den Transport zurücksetzen

Einleitung

Wenn Sie beispielsweise das Transportkennwort verlegt haben, können Sie ein neues Sicherheitszertifikat erstellen. Dabei setzen Sie das Transportkennwort zurück.

Kennwort zurücksetzen

1. Klicken Sie auf die Schaltfläche "Kennwort zurücksetzen".
   Das Kennwort wird zurückgesetzt. Ein neues Zertifikat ohne Kennwort wird generiert.

   Hinweis

   Andere Kennwörter wie z. B. das Kennwort von der PLC, werden ungültig. Beim nächsten Übersetzen informiert eine Fehlermeldung Sie darüber, dass diese Kennwörter nicht mehr gültig sind.

2. Geben Sie das neue Kennwort ein und bestätigen Sie es im Feld "Kennwort bestätigen".

   Hinweis

   Solange Sie kein neues Kennwort für den Transport eingeben, ist das Zertifikat nicht kennwortgeschützt.

   Nachdem Sie das Projekt auf das Bediengerät geladen haben, erscheint die Kennwortabfrage.

Siehe auch

Kennwortschutz in Runtime: Grundlagen (Seite 2361)

Kennwort für den Transport konfigurieren (Seite 2361)

15.4.2 Übersicht zum Übersetzen und Laden von Projekten

Übersicht

Bereits während Sie ein Projekt in WinCC projektieren, wird das Projekt laufend im Hintergrund übersetzt. Dadurch werden die Zeiten für das endgültige Übersetzen verkürzt. Wenn Sie das Übersetzen starten, entsteht eine Datei, die am jeweiligen Bediengerät ablauffähig ist.

Wenn beim Übersetzen Fehler auftreten, unterstützt WinCC Sie dabei, die Fehler zu finden und zu beheben.

Nach der Fehlerbehebung laden Sie das übersetzte Projekt auf die Bediengeräte, auf denen Sie das Projekt einsetzen wollen. Wenn der Projektierungs-PC nicht mit dem Bediengerät verbunden ist, speichern Sie das übersetzte Projekt auf einem Datenträger ihrer Wahl. Das übersetzte Projekt übertragen Sie dann von einem mit dem Bediengerät verbundenen PC auf das Bediengerät.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten S7-Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

Hinweis

WinCC-Versionskompatibilität

Um ein Projekt einer vorhergehenden WinCC-Version zu übersetzen, rüsten Sie das Projekt auf Ihre WinCC-Version hoch. Sie können das Projekt dann nicht mehr in der vorhergehenden WinCC –Version öffnen.

Projekt

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Das folgende Bild zeigt den Zusammenhang zwischen WinCC-Projekt und Runtime-Projekt am Beispiel des Vorgangs von "Übersetzen und Laden":

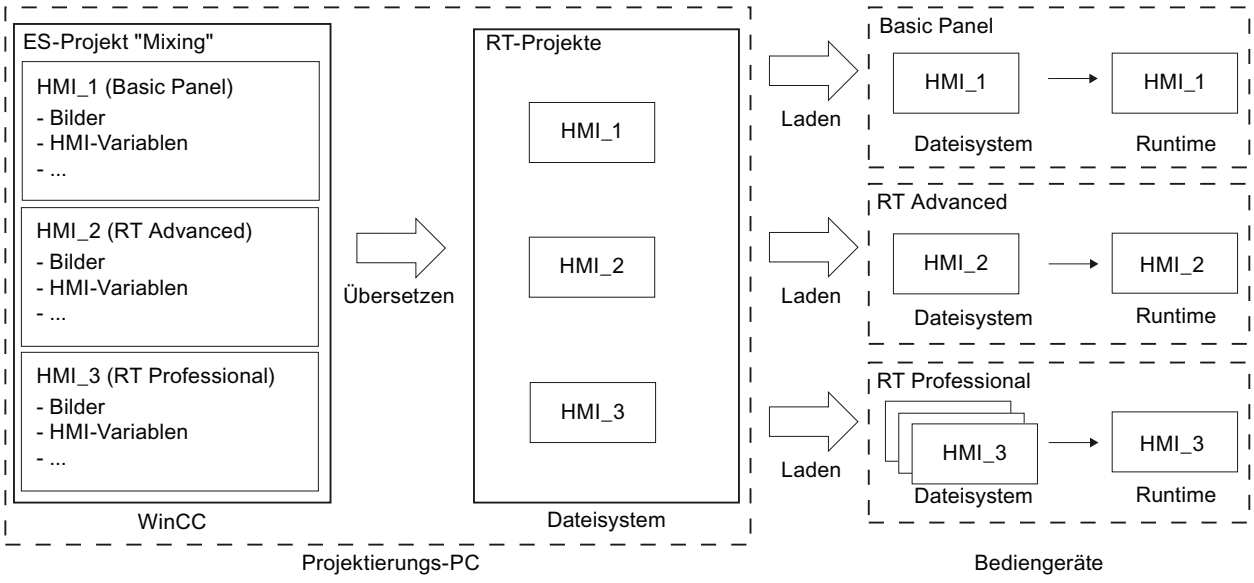

Runtime

Runtime ist die Software zur Prozessvisualisierung. In Runtime führen Sie das Projekt im Prozessbetrieb aus.

Es wird zwischen zwei Arten von Runtime unterschieden:

1. Runtime auf einem Panel
   Bevor Sie ein Runtime-Projekt auf einem Panel ausführen, müssen Sie das Runtime-Projekt vor dem Start auf das Panel übertragen.
2. Runtime auf einem PC
   Wenn Runtime auf dem Projektierungs-PC installiert ist, dann können Sie das Runtime-Projekt direkt auf dem Projektierungs-PC ausführen.
   Wenn Sie das Runtime-Projekt auf einem anderen PC ausführen wollen, müssen Sie das Runtime-Projekt vor dem Start auf den PC übertragen.

Simulation

Mit einer Simulation testen Sie ihre Projektierung. Eine Simulation können Sie ohne eine Anbindung an den laufenden Prozess starten.

In einer Simulation testen Sie beispielsweise projektierte Variablen oder Bildwechsel. Während der Simulation können die projektierten Variablen mithilfe des Variablensimulators manipuliert, aktiviert und deaktiviert werden.

Es wird zwischen zwei Simulationsarten unterschieden:

1. Simulation eines Panels
   Wenn Sie ein Panel in ihrem Projekt angelegt haben, wird das Panel in der Simulation angezeigt. Mithilfe dieser Simulationsart können Sie ihre Projektierung am Bediengerät testen, ohne das Projekt auf das Panel zu übertragen.
2. Simulation einer Runtime
   Bei der Simulation einer Runtime testen Sie das Projekt direkt am Projektierungs-PC.

Siehe auch

"Pack&Go"-Datei erzeugen (Seite 2277)

Runtime Advanced und Panel Runtime starten (Seite 2290)

Dateien über den Editor "HMI Dateien" laden (Seite 2193)

15.4.3 Projekt übersetzen

Einleitung

Bereits während Sie ein Projekt in WinCC projektieren, werden die Änderungen am Projekt laufend im Hintergrund übersetzt. Wenn Sie ein Projekt laden, wird das Projekt automatisch übersetzt. Auf diese Weise wird sichergestellt, dass immer der neueste Stand des Projekts geladen wird.

Während des Übersetzens überprüft WinCC das Projekt auf Konsistenz. Im Inspektorfenster werden alle fehlerhaften Stellen im Projekt aufgelistet. Über den Eintrag im Inspektorfenster springen Sie direkt zur Fehlerursache. Sie überprüfen die festgestellten Fehler und beheben diese.

Umfang des Übersetzens

Sobald Sie mit der Projektierung eines Bediengeräts beginnen, werden die Projektierungsdaten laufend im Hintergrund übersetzt. Wenn Sie ein Projekt manuell übersetzen, werden nur die Änderungen in der Projektierung seit dem letzten Übersetzen im Hintergrund übersetzt.

Um z. B. die Konsistenz der projektierten Daten zu testen, können Sie ein komplettes Übersetzen des Projekts jederzeit manuell starten.

Voraussetzung

- Ein Projekt ist geöffnet.

Vorgehen

Um ein Projekt zu übersetzen, gehen Sie folgendermaßen vor:

1. Wenn Sie mehrere Bediengeräte gleichzeitig übersetzen wollen, wählen Sie alle gewünschten Bediengeräte mit der Mehrfachauswahl in der Projektnavigation aus.
2. Übersetzen Sie das Projekt:
   - Um nur Änderungen im Projekt zu übersetzen, wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software (nur Änderungen)".
   - Um das Projekt komplett zu übersetzen, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (komplett übersetzen)".

Ergebnis

Die Projektierungsdaten aller ausgewählten Bediengeräte werden übersetzt. Wenn beim Übersetzen Fehler auftreten, werden diese im Inspektorfenster angezeigt.

15.4.4 Simulieren von Projekten

15.4.4.1 Grundlagen zur Simulation

Einleitung

Mit dem Simulator testen Sie das Verhalten Ihrer Projektierung am Projektierungs-PC. Auf diese Weise finden Sie frühzeitig und vor dem Produktivbetrieb logische Projektierungsfehler.

Sie können den Simulator auf folgende Arten starten:

- Im Kontextmenü des Bediengeräts oder eines Bildes: "Simulation starten"
- Menübefehl "Online > Simulation > [Starten|Mit Variablen-Simulator|Mit Skript-Debugger]"
- In der Portalansicht unter "Visualisierung > Gerät simulieren"

Voraussetzung

Auf dem Projektierungs-PC ist die Komponente Simulation/Runtime installiert.

Einsatzgebiet

Sie können mit dem Simulator z. B. folgende Funktionen des Bedien- und Beobachtungssystems testen:

- Überprüfung von Grenzwertstufen und Meldeausgaben
- Durchgängigkeit von Alarmen
- Projektierte Alarmsimulationen
- Projektierte Warnungen
- Projektierte Fehlermeldungen
- Kontrolle von Statusanzeigen

15.4.4.2 Simulation von Runtime Professional

Einleitung

Bei der Simulation von Runtime Professional testen Sie das Projekt mit Prozessanbindung direkt am Projektierungs-PC. Darüber hinaus testen Sie die Variablen mithilfe des Variablensimulators.

Funktion

Sowohl Prozesswerte als auch Meldungen werden in einem Simulationsarchiv hinterlegt. Beim Beenden der Simulation werden die erzeugten Archivdaten wieder gelöscht.

Der Ursprungszustand der Runtime-Datenbank vor der Simulation wird wieder hergestellt. Wenn Sie während der laufenden Runtime Projektdaten geändert haben, gehen die Änderungen dabei verloren. Mit dem Variablen-Simulator simulieren Sie die Belegung der Variablen Ihres Projekts mit unterschiedlichen Werten.

Zum Projekt gehörige Archivierungsvorgänge und Backup-Vorgänge laufen auch während der Simulation weiter.

⚠ WARNUNG
Beschädigung der Anlage möglich Wenn Sie während der Simulation mit Steuerungsanbindung Variablenwerte ändern, können unvorhergesehene Anlagenzustände bis zur irreparablen Beschädigung der Anlage eintreten. Beschränken Sie deshalb die manuellen Wertvorgaben auf ein Minimum und wählen Sie die Testwerte sorgfältig aus.

Einschränkungen

Client-Server-Betrieb ist während der Simulation nicht möglich.

15.4.4.3 Projekt simulieren

Voraussetzung

Projekt ist am Projektierungs-PC geöffnet.

Hinweis

Werte von Variablen mit einer gestörten Verbindung zum Prozess können Sie nicht simulieren.

Vorgehen

Um ein Projekt zu simulieren, gehen Sie folgendermaßen vor:

1. Wenn Sie die Funktion des Gesamtprozesses nicht testen wollen, trennen Sie die Verbindung zur Steuerung oder schalten Sie die Steuerung ab.
2. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator". Runtime wird gestartet und der Simulator wird mit allen projektierten Variablen geöffnet.
3. Um eine Variable zum Variablen-Simulator hinzuzufügen, wählen Sie im Menü "Bearbeiten" den Befehl "Neue Variable".
4. Wählen Sie die Variable aus dem Variablenhaushalt von WinCC aus.
5. Um eine Variable mit simulierten Werten zu versorgen, selektieren Sie im Variablen-Simulator auf der Registerkarte "Variablen" die gewünschte Variable.
6. Konfigurieren Sie auf der Registerkarte "Eigenschaften" die Simulationsart:
   - Wählen Sie die Simulationsart aus.
   - Geben Sie Werte für die Simulation ein.
   - Um die Variable in der Simulation zu berücksichtigen, klicken Sie auf "aktiv".

   ⚠ WARNUNG

   Vergewissern Sie sich nochmals, dass der Projektierungs-PC keine veränderten Variablen an eine Steuerung im laufenden Prozess weiter geben kann. Wenn die Steuerung nicht abgeschaltet ist, werden die simulierten Variablenwerte an die Steuerung und damit den laufenden Prozess übergeben.
7. Klicken Sie auf der Registerkarte "Eigenschaften" die Schaltfläche "Start Simulation".
8. Beobachten Sie die Auswirkung der Simulation auf Ihr Projekt in Runtime.
9. Um die Simulation zu stoppen, klicken Sie auf die Schaltfläche "Stop Simulation".
10. Um die Simulationseinstellungen zu speichern, wählen Sie den Menübefehl "Datei > Speichern" und geben Sie einen sprechenden Namen ein.
    Die Datei erhält automatisch die Endung *.sim.

Simulationsdaten verwalten

Wenn Sie die Daten einer früheren Simulation gespeichert haben, können Sie die Datei zu einem späteren Zeitpunkt öffnen und Ihr Projekt erneut simulieren. Die zu simulierenden Variablen und Bereichszeiger müssen im Projekt unverändert geblieben sein, sonst entsprechen die gespeicherten Simulationsdaten nicht mehr Ihrem Projekt.

Um eine Simulationsdatei zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator".
2. Wählen Sie im Variablen-Simulator den Menübefehl "Datei > Öffnen".
3. Wählen Sie die entsprechende Simulationsdatei aus und klicken Sie auf "Öffnen".
   Der Simulator übernimmt die gespeicherten Daten.

Variablen aktivieren und deaktivieren

Damit der Übergang von der Offline-Projektierung zur Online-Projektierung reibungslos möglich ist, können Sie die Simulation jeder Variable einzeln starten und stoppen. Aktivieren Sie dazu auf der Registerkarte "Eigenschaften" das Kontrollkästchen "aktiv".

Wenn eine Variable aktiviert ist, werden die Simulationswerte berechnet und an den Variablen-Simulator übergeben.

Variable löschen

Um eine Variable aus dem Variablen-Simulator zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variablennamen.
2. Wählen Sie den Menübefehl "Bearbeiten > Löschen".

Die Variable wird aus der Tabelle entfernt.

15.4.4.4 Bild simulieren

Einleitung

Wenn Sie nur Änderungen in einem Bild gemacht haben, können Sie dieses Bild temporär als Startbild für die Simulation festlegen. Auf diese Weise können Sie eine Änderung testen, ohne das Startbild zu ändern oder am Bediengerät zu dem Bild zu navigieren.

Voraussetzung

Ein Projekt mit mindestens einem Bild ist erstellt.

Vorgehen

Um ein Bild temporär als Startbild für die Simulation festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Bild, das in der Simulation als Startbild angezeigt werden soll.
2. Wählen Sie im Kontextmenü des Bildes den Befehl "Starte Simulation"

Ergebnis

Die Simulation des Projekts wird gestartet. Im Simulationsfenster wird statt des projektierten Startbildes das in der Projektnavigation selektierte Bild angezeigt.

15.4.4.5 Arbeiten mit dem Variablen-Simulator

Simulationsarten

Der Simulator stellt sechs verschiedene Simulationsarten zur Verfügung. Während der Simulation werden die projektierten Variablen je nach Simulationsart mit realitätsnahen Werten versorgt.

Simulationsart	Beschreibung
Sinus	Verändert den Wert der Variable in Form einer Sinuskurve. Die Darstellung erfolgt als periodische, nicht lineare Funktion. Einstellungen: • Amplitude • Offset (Nullpunkt für Wertebereich) • Schwingungsdauer
Schwingung	Zur Simulation von Sprüngen einer Führungsgröße. Einstellungen: • Überschwingung • Sollwert • Schwingungsdauer • Dämpfung
Zufall	Stellt zufällig erzeugte Werte zur Verfügung. Der Wert der Variable wird über eine Zufallsfunktion verändert. Einstellungen: • Minimalwert • Maximalwert
Inkrement	Erhöht den Wert der Variable kontinuierlich bis zu einem maximal angegebenen Wert. Beginnt nach Erreichen des Maximums wieder beim Minimum. Der Verlauf der Werte entspricht einer positiven Sägezahnkurve. Einstellungen: • Startwert • Endwert

Simulationsart	Beschreibung
Dekrement	Verringert den Wert der Variable kontinuierlich bis zu einem minimal angegebenen Wert. Beginnt nach Erreichen des Minimums wieder beim Maximum. Der Verlauf der Werte entspricht einer negativen Sägezahnkurve. Einstellungen: • Startwert • Endwert
Slider	Setzt einen fest Wert ein. Einstellungen: • Minimum • Maximum

15.4.4.6 Einschränkungen bei der Simulation

Meldungen mit dynamischen Parametern

Wenn Sie Variablen oder Textlisten als externe Variablen für Meldungen verwenden, dann werden die dynamischen Parameter der Meldungen nicht angezeigt.

Für die Simulation von Meldungen im Variablen-Simulator sind nur interne Variablen freigegeben.

Verwenden Sie PLCSim, um dynamische Parameter zu simulieren.

Comfort Panels

Wenn Sie PLCSim mit der Version V14 oder höher für die Simulation mit einem Comfort Panel verwenden möchten, muss die Geräteversion des Comfort Panels 14.0.0.0 oder höher sein.

15.4.5 Laden von Projekten

15.4.5.1 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum ersten Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Wenn sich das Bediengerät z. B. in einem Subnetz befindet, wählen Sie zusätzlich das Subnetz und das erste Gateway aus.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Wenn die Geräteversion des Ziel-Bediengeräts nicht der projektierten Geräteversion entspricht, wird abgefragt, ob Sie die Geräteversion jetzt ändern wollen.
     Solange das Projekt mit der eingestellten Geräteversion des Ziel-Bediengeräts nicht lauffähig ist, können Sie das Laden nicht starten.

ACHTUNG
Das Ändern der Geräteversion löscht alle Daten auf dem Bediengerät Wenn Sie die Geräteversion ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb zuerst folgende Daten: • Benutzerverwaltung • Rezepturen • License Keys Bei den Comfort-Geräten werden die Lizenzen nur beim Zurücksetzen auf Werkseinstellungen gelöscht. Sichern Sie vor dem Zurücksetzen auf Werkseinstellungen die License Keys.

   - Voreinstellungen, die beim Laden wirksam werden. Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional). Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional). Sie können das Projekt nicht laden. Beheben Sie zunächst die Fehler und laden Sie das Projekt erneut.
     Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Hinweis

Unterbrechung des Ladens

Falls das Laden unterbrochen wird, sichert WinCC automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf ein externes Speichermedium, z. B. einen USB-Stick, und laden es vom USB-Stick auf das Bediengerät.

Diese Funktion ist in Verbindung mit einem Bediengeräte-Image verfügbar, das mit der TIA Portal Version V14 oder höher kompatibel ist.

Sie erzeugen die notwendigen Projektdaten in WinCC, indem Sie das Bediengerät projektieren und anschließend den Ordner des Bediengeräts (z. B. "HMI_1 [<*Gerätetyp*>]") per Drag and Drop auf ein externes Speichermedium unter "Card Reader/USB-Speicher" ziehen.

Laden von Projekten ohne Rezepturdatensätze

In einem Projekt verwenden Sie Rezepturen. Sie übertragen das Projekt auf ein Basic Panel jedoch ohne die Rezepturdatensätze.

Wenn Sie die Struktur der Rezeptur im Engineering System geändert haben und auf dem Gerät bereits Rezepturdatensätze vorhanden waren, dann kann es zu Inkonsistenzen kommen.

Überprüfen Sie in diesem Fall die Konsistenz der Datensätze. Auf dem Gerät wird nicht bei allen strukturellen Änderungen ein Hinweis ausgegeben.

Pack&Go

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren Sie das übersetzte Projekt auf einen PC, der mit dem Bediengerät verbunden ist, z. B. über Netzwerk. Von diesem PC aus laden Sie das Projekt auf das Bediengerät .

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- PN/IE

Übertragen von Runtime-Addons in WinCC

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

15.4.5.2 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Verbindungsaufbau

Bitte beachten Sie, dass die im Engineering System projektierte Geräteversion mit der Version der installierten Runtime übereinstimmt, da ansonsten die Kommunikation beim Laden des Projektes nicht oder nur verzögert aufgebaut werden kann.

Wenn die Geräteversion nicht mit der Version der installierten Runtime übereinstimmt, haben Sie folgende Möglichkeiten:

1. Ändern Sie die im Engineering System projektierte Geräteversion.
2. Installieren Sie die zur projektierten Geräteversion passende Version von WinCC Runtime Advanced.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum ersten Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Wenn sich das Bediengerät z. B. in einem Subnetz befindet, wählen Sie zusätzlich das Subnetz und das erste Gateway aus.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Prüfung der Runtime-Version des Ziel-Bediengeräts
     Wenn die Version der auf dem Zielgerät installierten WinCC Runtime Advanced nicht mit der projektierten Geräteversion übereinstimmt, können Sie das Projekt nicht laden.
   - Voreinstellungen, die beim Laden wirksam werden
     Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional)
     Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional)
     Sie können das Projekt nicht laden. Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Hinweis

Unterbrechung des Ladens

Falls das Laden unterbrochen wird, sichert WinCC automatisch, dass keine Daten verloren gehen und dass vorhandene Daten auf dem Bediengerät erst nach vollständiger Übertragung gelöscht werden.

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, kopieren das übersetzte Projekt auf das Bediengerät.

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- PN/IE

Übertragen von Runtime-Addons in WinCC

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

15.4.5.3 Übersicht zum Laden von Projekten

Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

Wenn Sie den Befehl "Laden (komplett)" ausführen, müssen Sie das Überschreiben des Projekts bestätigen.

Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum ersten Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.
   Wenn sich das Bediengerät z. B. in einem Subnetz befindet, wählen Sie zusätzlich das Subnetz und das erste Gateway aus.
   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
   - Die Einzelschritte des Ladens
   - Prüfung der Runtime-Version des Ziel-Bediengeräts
     Wenn die Version der auf dem Zielgerät installierten WinCC Runtime Professional nicht mit der projektierten Geräteversion übereinstimmt, können Sie das Laden nicht starten.
   - Voreinstellungen, die beim Laden wirksam werden. Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
   - Auftretende Warnungen (optional). Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
   - Auftretende Fehler (optional). Sie können ein Projekt trotz Fehler laden. In diesem Fall aktivieren Sie im Dialog "Vorschau Laden" unter "Aktionen" das Optionskästchen "Das Projekt ist nicht fehlerfrei übersetzt. Wollen Sie mit dem Laden fortfahren?" und klicken auf "Laden".
     Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch das Anwenderprogramm.

Laden eines Projekts ohne angeschlossenes Bediengerät

Wenn Sie vom Projektierungs-PC keine direkte Verbindung zum Bediengerät herstellen können, laden Sie das übersetzte Projekt auf einen USB-Stick und kopieren es am Bediengerät über den Windows Explorer in das entsprechende Verzeichnis.

Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- PN/IE

Übertragen von Runtime-Addons in WinCC

Projekte können Runtime-Addons in Form von Controls oder CSPs (Communication Support Packages) enthalten. Diese Runtime-Addons werden automatisch mit dem Projekt übertragen.

Laden von inkonsistenten Projekten

Das Laden von inkonsistenten Projekten auf ein Gerät kann zum Verlust der Online-Delta-Ladefähigkeit des Geräts führen. Daher wird das Laden von inkonsistenten Projekten in Runtime Professional vom System unterbunden. Sie können ein Runtime-Projekt nur simulieren oder auf ein Gerät laden, wenn beim Übersetzen keine Fehler ausgegeben wurden. Die Optionen "Komplettladen", "Online-Deltaladen" sowie das Laden über einen Datenträger stehen bei inkonsistenten Projekten nicht zur Verfügung. Wenn das System Inkonsistenzen feststellt, werden Sie im Dialog "Vorschau laden" durch eine Fehlermeldung darüber informiert, dass das Projekt nicht auf das Gerät geladen werden kann.

Ausnahme bilden Fehler oder Inkonsistenzen in VB-Scripting und C-Scripting. Sie können ein Projekt mit Fehlern in VB-Scripting und C-Scripting auf ein Gerät laden oder simulieren, wenn Sie unter "Runtime-Einstellungen > Allgemein > Optionen für Skripte" das Optionskästchen "Auch mit fehlerhaften Skripten starten" aktivieren.

15.4.5.4 Projekt auf ein Bediengerät laden

Einleitung

Damit ein Projekt auf einem Bediengerät abläuft, laden Sie das Projekt auf das Bediengerät. Beim Laden legen Sie insbesondere fest, ob auf dem Bediengerät vorhandene Daten wie "Benutzerverwaltung" oder "Rezepturdaten" überschrieben werden. Generell kann auf einem Bediengerät immer nur ein Projekt in Runtime sein.

Sie laden ein Projekt vom Engineering System auf ein Bediengerät entweder mit dem Befehl "Laden (komplett)" oder mit dem Befehl "Delta-Laden".

Wenn das Bediengerät PROFINET unterstützt, wird der in der Projektnavigation eingetragene Name des Bediengeräts als Gerätename für die PROFINET-Kommunikation verwendet. Der Name wird beim Laden in das Bediengerät geschrieben. Wenn im Bediengerät bereits ein Gerätename für die PROFINET-Kommunikation eingetragen war, wird dieser überschrieben.

Beachten Sie für das Laden eines Projekts auf ein Bediengerät Folgendes:

- Wenn auf dem Bediengerät kein Projekt geöffnet oder in Runtime ist, können Sie ein Projekt vom Projektierungs-PC auf das Bediengerät laden.
- Wenn der Projektierungs-PC auch das Bediengerät ist, laden Sie das Projekt in Ihr Projektverzeichnis im Dateisystem.

- Wenn auf dem Bediengerät ein Projekt geöffnet ist, können Sie nur genau dieses Projekt vom Projektierungs-PC erneut komplett auf das Bediengerät laden. Um ein anderes Projekt auf das Bediengerät zu laden, schließen Sie auf dem Bediengerät das geöffnete Projekt. Beispiel: Auf dem Bediengerät ist das Projekt "Mixing" geöffnet, aber nicht in Runtime. Wenn Sie das Projekt ändern, laden Sie die Änderungen mit "Software (komplett laden)" auf das Bediengerät. Um z. B. das Projekt "Bottling" auf das Bediengerät zu laden, schließen Sie auf dem Bediengerät das Projekt "Mixing".
- Wenn auf dem Bediengerät ein Projekt in Runtime ist, können Sie einige Änderungen an den Projektierungsdaten laden, ohne Runtime zu beenden (Delta-Laden). Wenn durch eine Änderung die Deltaladefähigkeit verloren geht, werden Sie explizit darauf hingewiesen. Nach dem Verlust der Deltaladefähigkeit können Sie das Projekt nur mit "Laden (komplett)" auf das Bediengerät laden. Dabei wird Runtime deaktiviert und wieder neu gestartet, was zu Stillstandszeiten an der Anlage führen kann.
  Wenn Sie ein anderes Projekt auf das Bediengerät laden wollen, müssen Sie Runtime beenden und das Projekt am Bediengerät schließen.
- Um ein Projekt einer vorhergehenden WinCC-Version zu laden, rüsten Sie vorher das Projekt auf Ihre WinCC-Version hoch.

ACHTUNG
Geräteversion Wenn die Geräteversion des Ziel-Bediengerätes nicht der projektierten Geräteversion entspricht, kann nach dem Laden Runtime nicht gestartet werden. Überprüfen Sie vor dem Übersetzen und Laden Ihres Projektes in Datei, ob die Geräteversion des Ziel-Bediengerätes Ihrer Projektierung entspricht. Ändern Sie bei Bedarf die Geräteversion manuell über die Eigenschaften des Bediengerätes.

- Wenn das gewählte Zielverzeichnis auf dem Bediengerät leer ist, wird nach dem Ausfüllen des Dialogs "Erweitertes Laden" der Befehl "Laden (komplett)" automatisch ausgeführt.
- Wenn ein gleichnamiges Runtime-Projekt im gewählten Zielverzeichnis enthalten ist und Sie den Befehl "Laden (komplett)" ausführen, müssen Sie das Überschreiben des Projekts bestätigen.
- Wenn ein gleichnamiges Runtime-Projekt im gewählten Zielverzeichnis enthalten ist und die Runtime aktiviert ist, können Sie alternativ den Befehl "Delta-Laden" ausführen.
- Wenn der Name des Runtime-Projekts mit dem Projektnamen auf dem Engineering System nicht übereinstimmt, wird das Projekt nicht geladen.
- Das Laden von inkonsistenten Projekten wird in Runtime Professional vom System unterbunden. Sie können ein Runtime-Projekt nur simulieren oder auf ein Gerät laden, wenn beim Übersetzen keine Fehler ausgegeben wurden. Wenn das System Inkonsistenzen feststellt, werden Sie im Dialog "Vorschau laden" durch eine Fehlermeldung darüber informiert, dass das Projekt nicht auf das Gerät geladen werden kann.
  Ausnahme bilden Fehler oder Inkonsistenzen in VB-Scripting und C-Scripting. Sie können ein Projekt mit Fehlern in VB-Scripting und C-Scripting auf ein Gerät laden oder simulieren, wenn Sie unter "Runtime-Einstellungen > Allgemein > Optionen für Skripte" das Optionskästchen "Auch mit fehlerhaften Skripten starten" aktivieren.

Voraussetzung

- Im Projekt ist ein Bediengerät angelegt.
- Das Bediengerät ist mit dem Projektierungs-PC verbunden oder der Projektierungs-PC ist das Bediengerät.
- Auf dem Projektierungs-PC und dem Bediengerät ist dieselbe Kommunikationsart eingestellt, z. B. verschlüsselte Kommunikation. Alternativ verwenden Sie auf einem der Geräte den Migrationsmodus (siehe AUTOHOTSPOT).
- Bei verschlüsselter Kommunikation ist auf beiden Seiten das gleiche Passwort vergeben.

Vorgehen

Um ein Projekt zu laden, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Projekt auf mehrere Bediengeräte gleichzeitig laden wollen, wählen Sie in der Projektnavigation alle gewünschten Bediengeräte mit der Mehrfachauswahl aus.
2. Wählen Sie im Kontextmenü eines Bediengeräts den Befehl "Laden in Gerät > Software".
3. Wenn der Dialog "Erweitertes Laden" geöffnet wird:
   – Wählen Sie den "Zielpfad" für das übersetzte Projekt auf dem Bediengerät.
     Die Windows-Benutzergruppe "SIMATIC HMI" muss in diesem Zielpfad Vollzugriff besitzen.
     Der Vollzugriff wird standardmäßig während der Installation von WinCC Runtime für den Ordner "WinCC Projects" gesetzt. Wo sich der Ordner im Dateisystem befindet, ist abhängig vom Betriebssystem des Bediengeräts.

     Hinweis

     Bei mehreren Runtime-Projekten legen Sie für jedes Projekt ein separates Verzeichnis an. Wenn Sie ein Projekt zum ersten Mal auf das Bediengerät laden, muss das Verzeichnis leer sein.

   – Klicken Sie auf "Laden".

   Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät…" aufrufen.
   Der Dialog "Vorschau laden" wird geöffnet. Gleichzeitig wird das Projekt übersetzt. Das Ergebnis wird im Dialog "Vorschau laden" angezeigt.
4. Überprüfen Sie die angezeigten Voreinstellungen und ändern Sie diese bei Bedarf:
   – Wählen Sie die zu übertragenden Daten.
   – Wählen Sie, ob Runtime auf dem Zielsystem nach dem Laden gestartet werden soll.
5. Klicken Sie auf "Laden".

Ergebnis

Das Projekt wird auf die ausgewählten Bediengeräte geladen. Wenn Fehler oder Warnungen beim Laden auftreten, werden im Inspektorfenster unter "Info > Laden" entsprechende Meldungen ausgegeben.

Nach dem erfolgreichen Laden ist das Projekt auf dem Bediengerät ablauffähig. Wenn Sie im Dialog "Vorschau laden" das Starten von Runtime am Zielsystem aktiviert haben, wird nach dem Laden das Projekt in Runtime gestartet.

Siehe auch

Übersicht zum Online-Deltaladen (Seite 2382)

15.4.5.5 Projekt in ein Dateisystem laden

Einleitung

Wenn der Projektierungs-PC keine Verbindung zum Bediengerät hat, laden Sie das Projekt offline über einen USB-Datenträger oder eine CD auf das Bediengerät. Dazu kopieren Sie das Runtime-Projekt auf Ihr Bediengerät und starten in Runtime Professional das Projekt mit "WinCC RT Start".

Der PC, auf dem Runtime aktiviert ist, wird als Bediengerät bezeichnet.

Beim Laden wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog legen Sie z. B. fest, ob Benutzerverwaltung oder Archive, die zwischenzeitlich am Bediengerät geändert wurden, überschrieben oder behalten werden.

Hinweis

Vorhandene Daten in der Benutzerverwaltung oder in den Archiven werden beim Starten des geladenen Runtime-Projekts überschrieben.

Um Änderungen der Benutzerverwaltung zu erhalten, exportieren Sie die Benutzerverwaltung und importieren die Daten nach dem Starten des kopierten Runtime-Projekts wieder. Die Einstellungen zum Überschreiben legen Sie im Dialog "Vorschau Laden" im Bereich "Überschreiben" fest.

Voraussetzung

Ein Projekt ist geöffnet.

Vorgehen

Um ein Projekt in das Dateisystem zu laden, gehen Sie folgendermaßen vor:

1. Markieren Sie das Projekt in der Projektnavigation.
2. Wählen Sie den Menübefehl "Online > Laden in Dateisystem".
   Ein Dialog wird geöffnet.
3. Wählen Sie den Ablagepfad und klicken Sie auf "OK".
   Das Projekt wird übersetzt.
4. Wenn der Dialog "Vorschau laden" geöffnet wird, wählen Sie die zu übertragendende Daten.

5. Unter "Optionen für die Variablenarchivierung" wählen Sie eine der folgenden Optionen:
   – "Beibehalten": Wenn das Projekt vom Engineering System auf den PC bereits einmal übertragen wurde, werden die Daten auf dem PC nicht durch die Daten vom Engineering System überschrieben.
   – "Zurücksetzen": Die Daten auf dem PC werden durch die Daten vom Engineering System überschrieben.
   – "Übertragen": Die Daten werden vom Engineering System auf den PC übertragen.
6. Klicken Sie auf "Laden".
   Das Runtime-Projekt wird im Laufwerk gespeichert.
   Bei Bedarf übertragen Sie das Runtime-Projekt auf einen USB-Datenträger oder brennen Sie das Projekt auf eine CD.
7. Übertragen Sie das Runtime-Projekt auf Ihr Bediengerät.
8. Starten Sie Runtime und öffnen Sie das geladene Projekt.

15.4.5.6 Online-Deltaladen für Runtime Professional

Übersicht zum Online-Deltaladen

Online-Deltaladen

Mit WinCC können Sie ein Projekt auf dem Projektierungs-PC bearbeiten, während dasselbe Projekt in Runtime läuft. Wenn Sie das Projekt auf dem Projektierungs-PC ändern, können Sie bei den meisten Projektierungsänderungen die geänderten Anteile mit "Online-Deltaladen" auf das Bediengerät laden. Das Online-Deltaladen benötigt weniger Zeit als das komplette Laden des Projekts. Beim Online-Deltaladen wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog legen Sie z. B. fest, ob Benutzerverwaltung oder Archive, die zwischenzeitlich am Bediengerät geändert wurden, beim Online-Deltaladen überschrieben oder behalten werden.

Hinweis

Änderungen in der Benutzerverwaltung oder in den Archiven, die zwischenzeitlich am Bediengerät vorgenommen wurden, gehen beim Online-Deltaladen verloren.

Um Änderungen der Benutzerverwaltung zu erhalten, exportieren Sie die Benutzerverwaltung und importieren sie nach dem Online-Deltaladen wieder.

Hinweis

Geräteversion

Wenn die Geräteversion des Ziel-Bediengerätes nicht der projektierten Geräteversion entspricht, ist das Deltaladen nicht möglich.

Überprüfen Sie vor dem Deltaladen Ihres Projektes, ob die Geräteversion des Ziel-Bediengerätes Ihrer Projektierung entspricht und aktualisieren Sie bei Bedarf die Version Ihres Bediengerätes.

Beachten Sie zum Online-Deltaladen Folgendes:

- Wenn Sie das Online-Deltaladen ohne laufende Runtime anstoßen, wird das Projekt komplett geladen.
- Während einer Simulation ist kein Online-Deltaladen möglich.
  Um die Simulation automatisch vor dem Laden abzubrechen und nach Möglichkeit nach dem Laden wieder zu starten, wählen Sie die entsprechende Option im Dialog "Vorschau laden".

Hinweis

Das Laden von inkonsistenten Projekten auf ein Gerät kann zum Verlust der Online-Delta-Ladefähigkeit des Geräts führen. Daher wird das Laden von inkonsistenten Projekten in Runtime Professional vom System unterbunden.

Komplettladen

Wenn Sie die Projektierung grundlegend ändern, kann dies zu einem Verlust der Online-Deltaladefähigkeit führen. Wenn Sie in diesem Fall das Online-Deltaladen starten, werden Sie darauf hingewiesen, dass Online-Deltaladen nicht möglich ist. Sie müssen dann das Projekt erneut komplett auf das Bediengerät laden. Im Dialog "Laden" müssen Sie diesen Neustart von Runtime am Bediengerät explizit bestätigen. Sie können das Laden an dieser Stelle abbrechen.

Wenn Sie Komplettladen starten, während das am Projektierungs-PC geöffnete Projekt am Bediengerät in Runtime ist, geschieht Folgendes:

- Runtime wird gestoppt
- Das komplette Projekt wird auf das Bediengerät geladen
- Runtime wird erneut gestartet

Beachten Sie, dass das Stoppen von Runtime zu Stillstandszeiten in der Anlage führen kann.

Anwendungsszenarien

In verschiedenen Phasen, z. B. bei Inbetriebnahme, Betrieb oder Wartung, ist es immer wieder erforderlich, die Projektierung anzupassen. Diese Änderungen übernehmen Sie dann online am Bediengerät, ohne dort Runtime zu beenden.

Online-Deltaladen setzen Sie z. B. in folgenden Szenarien ein:

- Kontinuierliche Automatisierungsaufgaben:
  Von einem zentralen Projektierungs-PC aus sollen sämtliche Änderungen online ausgeführt werden. Damit entfällt die Notwendigkeit, die Projektierung direkt vor Ort zu ändern. Sie können auch Runtime-Objekte wie Variablen, Alarme und Archive hinzufügen, ändern und löschen, ohne die Runtime zu beenden.
- Änderungen in geschützter Umgebung testen:
  An dem Projektierungs-PC können die beabsichtigten Änderungen offline ausgeführt werden, bevor diese auf das Bediengerät geladen werden. Der Projekteur kann den Einfluss der Änderungen erst in geschützter Umgebung testen, bevor die Änderungen in den aktiven Betrieb übernommen werden. Mögliche Projektierfehler können so rechtzeitig festgestellt und behoben werden, bevor diese zu Prozessproblemen oder Anlagenstillstand führen.

Siehe auch

Runtime auf dem Projektierungs-PC starten (Seite 2289)

Benutzerverwaltung exportieren (Seite 1412)

Benutzerverwaltung importieren (Seite 1413)

Einsatz und Einschränkungen des Online-Deltaladens

Einleitung

Bei einigen Projektierungen geht die Online-Delta-Ladefähigkeit verloren.

Dateien und Elemente

Die folgende Tabelle zeigt, bei welchen Aktionen die Online-Delta-Ladefähigkeit erhalten bleibt und bei welchen nicht:

Element	Anlegen	Löschen	Bearbeiten	Bemerkungen
Projektname / Eigenschaften des Projekts	-	-	Nein	-
Bediengerät	Ja	Nein	Nein	-
Eigenschaften des Bediengeräts	-	-	Nein	Bearbeiten: Änderungen werden erst dann in Runtime übernommen, wenn Sie das Projekt deaktivieren und Runtime neu starten.
Anwenderzyklus (Projekteigenschaften)	Nein	Nein	Nein	-
Verbindungen (Editor "HMI-Variablen")	Nein	Nein	Nein	-
Parameter einer Verbindung (Editor "HMI-Variablen")	Nein	Nein	Nein	-

Element	Anlegen	Löschen	Bearbeiten	Bemerkungen
Meldungen	-	-	-	siehe folgende Tabelle "Online-Projektierung bei Meldungen"
Menüs und Symbolleisten	Nein	Nein	Nein	-

Online-Projektierung bei Meldungen

Element	Attribut	Anlegen	Löschen	Bearbeiten	Bemerkungen
-	• Meldenummer • Meldegruppe	-	-	Nein	-
-	• Meldeklasse	Ja	Nein	Ja	Wenn Sie die Meldeklasse aber nicht die Quittierpflicht ändern, werden nicht anstehende Meldungen sofort aktualisiert. Wenn Sie die Quittierpflicht einer gerade anstehenden und noch nicht quittierten Meldung ändern, wird die anstehende Meldung vom System quittiert. Wenn Sie eine anstehende Meldung in eine Meldung ohne den Zustand "gegangen" ändern, geht die anstehende Meldung.
Anwenderdefinierte Meldegruppe	-	Ja	Nein	Ja	-
Grenzwerte für Analogmeldungen	-	Nein	Nein	Nein	-
Meldeklassen	-	Ja	Nein	Ja	Bearbeiten: Gilt für die meisten Attribute. Ausnahmen sind einzeln aufgeführt:

Hinweis

Anzeige von Steuerungsmeldungen

Wenn Sie nach dem Übersetzen und Laden unter "Runtime-Einstellungen > Meldungen > Steuerungsmeldungen" die Konfiguration der Option "Automatische Aktualisierung" verändern, geht die Online-Delta-Ladefähigkeit verloren.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Regeln zum Online-Deltaladen (Seite 2390)

15.4.5.7 Runtime-Projekt auf einem Bediengerät auf V13 SP1 aktualisieren

Einleitung

Wenn Sie ein TIA Portal-Projekt von V13 auf V13 SP1 auf Ihrem Engineering System aktualisiert haben, können Sie Ihr Runtime-Projekt auf dem Bediengerät ebenfalls von V13 auf V13 SP1 aktualisieren. Dabei wird das Projekt vom Engineering System vollständig in das Zielverzeichnis Ihres Bediengeräts übertragen. Beim Übertragen ist es möglich, die vorhandenen Runtime-Daten zu übernehmen.

Um ein auf V13 SP1 aktualisiertes Projekt vom Engineering System auf ein Bediengerät zu übertragen, beachten Sie Folgendes:

- Im gewählten Zielverzeichnis muss das zugehörige Runtime-Projekt in der alten Version vorliegen. In diesem Fall müssen Sie das Überschreiben des Runtime-Projektes im Dialog "Vorschau Laden" bestätigen.

Vorgehen

Um ein auf V13 SP1 aktualisiertes Projekt vom Engineering System auf ein Bediengerät zu übertragen und ein bestehendes Runtime-Projekt auf dem Bediengerät auf die Version 13 SP1 zu aktualisieren, gehen Sie folgendermaßen vor:

1. Markieren Sie das Projekt auf dem Engineering System in der Projektnavigation.
2. Wählen Sie den Menübefehl "Online > Laden in Gerät".
   Alternativ können Sie den Befehl "Laden in Gerät > Software (komplett laden)" aus dem Kontextmenü des Bediengeräts wählen.
   Der Dialog "Erweitertes Laden" wird geöffnet.
3. Wählen Sie das Zielverzeichnis des alten Runtime-Projekts unter "Pfad des Zielordners".
4. Klicken Sie auf "OK".
   Das Projekt wird übersetzt.
5. Wenn der Dialog "Vorschau laden" geöffnet wird, wählen Sie die zu übertragendende Daten.
6. Bestätigen Sie das Überschreiben des Zielprojekts.
7. Klicken Sie auf "Laden".
   Das aktualisierte Runtime-Projekt wird auf das Bediengerät übertragen.

15.4.6 Übersetzen und Laden mit Multiuser Engineering

15.4.6.1 Übersetzen und Laden mit Multiuser Engineering (Übersicht)

Einleitung

Beim Einsatz von Multiuser Engineering für die Bearbeitung Ihrer Projekte sollten Sie das Verhalten beim Übersetzen der Runtime-Projekte und das Verhalten beim Laden auf die Bediengeräte berücksichtigen.

Das Übersetzen und Laden auf ein Bediengerät ist sowohl in der Serverprojekt-Ansicht als auch in der lokalen Session möglich.

Weitere Informationen zum Thema "Multiuser Engineering" finden Sie unter "Multiuser Engineering einsetzen".

Grundlagen

Für Bediengeräte und RT Advanced sind im Multiuser Engineering folgende Szenarien möglich:

- Übersetzen in der Serverprojekt-Ansicht
- Übersetzen in der lokalen Session
- Laden aus der Serverprojekt-Ansicht
- Laden aus der lokalen Session

Hinweis

Das Komplettladen aus der Serverprojekt-Ansicht oder aus der lokalen Session unterscheidet sich nicht vom Komplettladen in einem Single-User-Projekt. Beim Komplettladen wird das aktuelle Runtime-Projekt aus der geraden aktiven Ansicht auf ein Bediengerät geladen.

Für RT Professional sind folgende zusätzliche Szenarien möglich:

- Online-Deltaladen aus der Serverprojekt-Ansicht
- Online-Deltaladen aus der lokalen Session

Hinweis

Das Übersetzen und Laden von Projekten in einer lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt mit Ausnahme von Online-Deltaladen.

Alle Regeln, z. B. Aktionen, die die Online-Deltaladefähigkeit aufheben, gelten auch für Online-Deltaladen in Multiuser Engineering.

Weitere Informationen finden Sie unter Einsatz und Einschränkungen des Online-Deltaladens (Seite 2384).

Grundsätzlich können Sie in Multiuser Engineering-Projekten alle Befehle zum Übersetzen und Laden ausführen:

- "Software (komplett übersetzen)"
- "Übersetzen > Software (nur Änderungen)"
- "Software (komplett laden)"
- "Online-Deltaladen" (nur RT Professional)

Der Begriff "Projekt"

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzen Projektierungsdaten eines Bediengeräts

Regeln

Folgende grundsätzliche Regeln gelten für Übersetzen und Laden in Multiuser Engineering:

- Das Runtime-Projekt, das in einer lokalen Session übersetzt wurde, bleibt immer lokal und wird nicht auf den Multiuser-Server hochgeladen. Es kann im Multiuser-Serverprojekt nicht gespeichert werden.
- Nur die Runtime-Projekte, die in der Serverprojekt-Ansicht übersetzt wurden, werden im Multiuser-Serverprojekt gespeichert.

Hinweis

Beachten Sie die Regeln zum Online-Deltaladen.

Weitere Informationen finden Sie unter Online-Deltaladen für Runtime Professional (Seite 2382)

Weitere Informationen zu Multiuser Engineering finden Sie im Siemens YouTube Channel: Multiuser Engineering -ein Team arbeitet gleichzeitig an einem Projekt (https://www.youtube.com/watch?v=6Kq-4YWHJq4).

Siehe auch

Übersetzen in der Serverprojekt-Ansicht (Seite 2388)

Übersetzen in der lokalen Session (Seite 2389)

Regeln zum Online-Deltaladen (Seite 2390)

Online-Deltaladen für Runtime Professional (Seite 2382)

Online-Deltaladen aus der Serverprojekt-Ansicht (Seite 2392)

Online-Deltaladen aus der lokalen Session (Seite 2393)

Einsatz und Einschränkungen des Online-Deltaladens (Seite 2384)

15.4.6.2 Übersetzen in der Serverprojekt-Ansicht

Grundlagen

Das Übersetzen und Laden von Projekten in der Serverprojekt-Ansicht unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Während Sie ein Projekt in der Serverprojekt-Ansicht übersetzen, wird das Multiuser-Serverprojekt blockiert. Andere Benutzer können in dieser Zeit keine Änderungen an diesem Serverprojekt durchführen. Das in der Serverprojekt-Ansicht übersetzte Runtime-Projekt wird zusammen mit dem Engineering Projekt im zentralen Multiuser-Server gespeichert. Durch das Blockieren des Multiuser-Serverprojekts ist gewährleistet, dass die Projektierungsdaten und das Runtime-Projekt synchron bleiben.

Hinweis

Wenn Sie in der Serverprojekt-Ansicht übersetzen und speichern, erhalten andere Benutzer beim "Aktualisieren" ihrer lokalen Session das von Ihnen aktualisierte Runtime-Projekt zusammen mit dem Engineering Projekt. Andere Benutzer müssen dann nach einer Aktualisierung nicht erneut die von Ihnen eingebrachten Änderungen übersetzen.

Beispiel: Übersetzen während des Eincheckens

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Wenn keine Übersetzungsfehler vorliegen, werden über "Änderungen speichern" beide Projekte, das modifizierte Engineering Projekt (mit den geänderten Variablen) und das übersetzte Runtime-Projekt, im zentralen Multiuser-Serverprojekt gespeichert.

Wenn Sie das Übersetzen während des Eincheckens auslassen, enthält das Projekt die Änderungen, die auf dem Server gespeichert wurden.

Der nächste Benutzer, der aus dem Serverprojekt eine lokale Session erstellt oder eine vorhandene lokale Session aktualisiert, muss zusätzlich zu seinen Änderungen noch Ihre zwei Änderungen übersetzen.

Hinweis

Das über mehrere lokale Sessions verteilte Arbeiten an einem gemeinsamen Projekt erhöht die Fehlerwahrscheinlichkeit. Daher ist es empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Übersetzen in der lokalen Session (Seite 2389)

15.4.6.3 Übersetzen in der lokalen Session

Grundlagen

Das Übersetzen und Laden von Projekten in der lokalen Session unterscheidet sich nicht vom Übersetzen und Laden in einem Single-User-Projekt.

Da die lokale Session eine Kopie des Serverprojekts ist, ist der erste Übersetzungsstatus der lokalen Session mit dem des Serverprojekts identisch. Wenn im Serverprojekt unübersetzte Inhalte oder Fehlermeldungen beim Übersetzen vorhanden waren, werden sie in die lokale Session übertragen.

Hinweis

Es ist empfehlenswert, beim Einchecken das Projekt zu übersetzen und alle Fehler, die beim Übersetzen ausgegeben werden, zu beseitigen. So übergeben Sie ein fehlerfreies Projekt an den nächsten Benutzer und vermeiden Fehlerverbreitung.

Aktualisieren in der lokalen Session

Wenn Sie ein Projekt in der lokalen Session aktualisieren, wird die lokale Session komplett - inklusive des Übersetzungsstatus - durch den Inhalt des Serverprojekts ersetzt. Nur die für das Einchecken markieren Änderungen werden in der aktualisierten lokalen Session beibehalten und erzeugen zusätzliche Übersetzungsschritte in der lokalen Session.

Beispiel: Aktualisieren der lokalen Session

Sie führen in einer lokalen Session die Änderung einer Variablen durch. Im dazugehörigen Serverprojekt wurden alle bisherigen Änderungen übersetzt.

Sie aktualisieren die Inhalte der lokalen Session, indem Sie auf die Schaltfläche "Aktualisieren" klicken. Nach dem Aktualisieren bekommt die lokale Session den Übersetzungsstatus des Serverprojekts. Zusätzlich sind Übersetzungsaufgaben vorhanden, die durch die Übernahme der geänderten Variablen begründet sind.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Übersetzen in der Serverprojekt-Ansicht (Seite 2388)

15.4.6.4 Regeln zum Online-Deltaladen

Übersicht

Online-Deltaladen wird nur von WinCC RT Professional unterstützt. Alle Regeln und Einschränkungen beim Online-Deltaladen gelten auch für Online-Deltaladen in Multiuser Engineering.

Das Befolgen dieser Regeln hilft Ihnen, potenzielle Fehler rechtzeitig zu erkennen und zu vermeiden. Weitere Informationen finden Sie unter Einsatz und Einschränkungen des Online-Deltaladens (Seite 2384).

Das folgende Bild zeigt eine Übersicht zum Online-Deltaladen mit Multiuser Engineering:

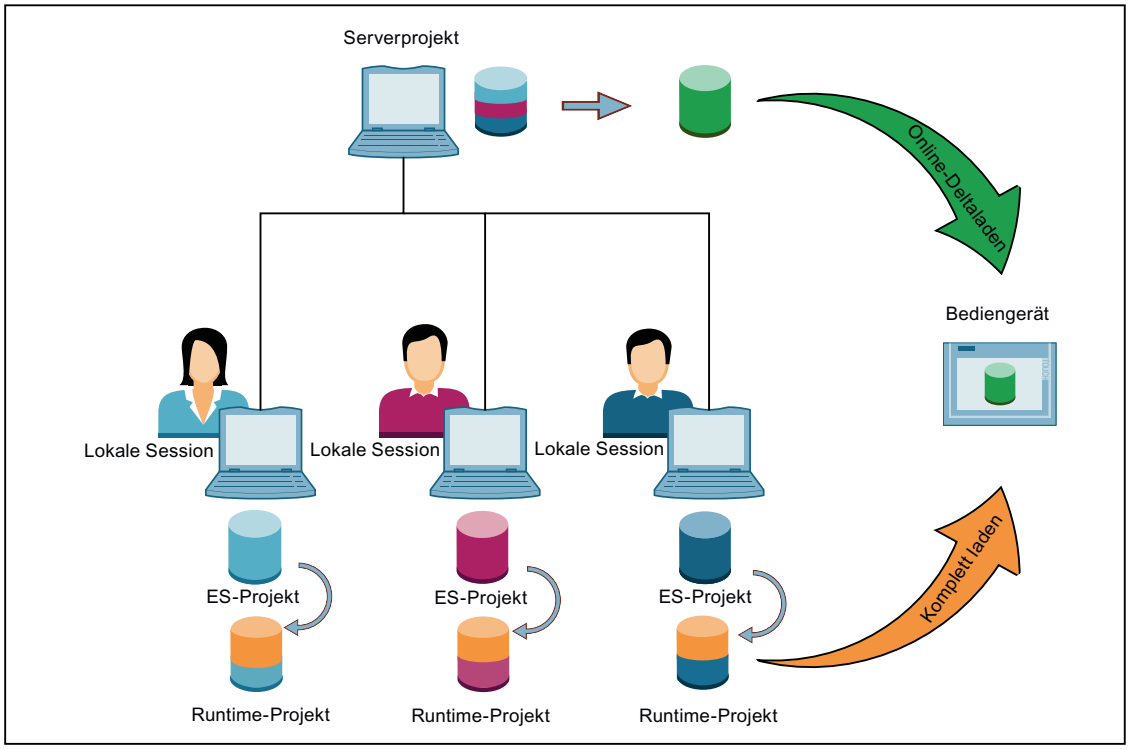

Regeln für Online-Deltaladen aus der Serverprojekt-Ansicht

- Alle Regeln und Einschränkungen beim Online-Deltaladen der Single-User-Projekte gelten auch für Online-Deltaladen in Multiuser Engineering.
- Fortlaufendes Online-Deltaladen ist ausschließlich aus der Serverprojekt-Ansicht ohne Laden aus lokalen Sessions möglich.
- Wenn Sie in der Serverprojekt-Ansicht eine Änderung durchführen, die die Online-Delta-Ladefähigkeit aufhebt, geht die Online-Delta-Ladefähigkeit im Multiuser-Serverprojekt verloren.

Regeln für Online-Deltaladen aus der lokalen Session

- Alle Regeln und Einschränkungen beim Online-Deltaladen der Single-User-Projekte gelten auch für Online-Deltaladen in Multiuser Engineering.
- Die lokale Session muss aus einem Serverprojekt erstellt werden, das über die Online-Delta-Ladefähigkeit verfügt.
- Eine lokale Session selbst verfügt über die Online-Delta-Ladefähigkeit.
- Wenn Sie in einer lokalen Session eine Änderung durchführen, die die Online-Delta-Ladefähigkeit aufhebt, verliert die lokale Session die Online-Delta-Ladefähigkeit.
- Die Online-Delta-Ladefähigkeit in der lokalen Session geht beim Aktualisieren verloren.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Online-Deltaladen für Runtime Professional (Seite 2382)

Einsatz und Einschränkungen des Online-Deltaladens (Seite 2384)

15.4.6.5 Online-Deltaladen aus der Serverprojekt-Ansicht

Grundlagen

Um die Online-Delta-Ladefähigkeit des Serverprojekts aufrechtzuerhalten, dürfen alle Benutzer nur solche Änderungen der Projektierung vornehmen, die die Online-Delta-Ladefähigkeit nicht beeinträchtigen.

Solange alle Benutzer im Engineering Projekt nur solche Änderungen machen, die die Online-Delta-Ladefähigkeit nicht aufheben, können Sie die geänderten Projektierungsanteile in WinCC Runtime Professional aus unterschiedlichen lokalen Sessions über die Serverprojekt-Ansicht auf dasselbe Bediengerät laden.

Hinweis

Beim Komplettladen geht die Online-Delta-Ladefähigkeit des Bediengeräts verloren.

Mehrere Benutzer können parallel in unterschiedlichen lokalen Sessions Projektierungsänderungen vornehmen. Die Schritte "Übersetzen > Software (nur Änderungen)" oder "Online-Deltaladen" müssen jedoch nacheinander in der Serverprojekt-Ansicht ausgeführt werden, weil dafür exklusive Zugriffsrechte erforderlich sind.

Weitere Informationen zum Thema "Multiuser Engineering" finden Sie unter "Multiuser Engineering einsetzen".

Hinweis

Fortlaufendes Online-Deltaladen aus unterschiedlichen lokalen Sessions wird nur über die Serverprojekt-Ansicht unterstützt.

Wenn Sie in ein Bediengerät aus der Serverprojekt-Ansicht laden und die Serverprojekt-Ansicht mit "Änderungen verwerfen" schließen, geht die Online-Delta-Ladefähigkeit verloren. Beim nächsten Laden in dieses Gerät wird der Befehl "Komplett laden" ausgeführt.

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Laden von Projekten (Seite 2371)

Online-Deltaladen für Runtime Professional (Seite 2382)

15.4.6.6 Online-Deltaladen aus der lokalen Session

Grundlagen

Online-Deltaladen auf ein Bediengerät ist nicht nur aus der Serverprojekt-Ansicht sondern auch aus der lokalen Session möglich.

Online-Deltaladen aus einer lokalen Session unterliegt folgenden Einschränkungen:

- Nach dem Online-Deltaladen aus einer lokalen Session kann für dieses Bediengerät in Zukunft nur noch aus dieser lokalen Session Online-Deltaladen ausgeführt werden.
- In der Serverprojekt-Ansicht ist danach nur noch ein Komplettladen des Bediengeräts möglich.
- Jede Aktualisierung der lokalen Session macht ein Komplettladen des Bediengeräts erforderlich.
- Das Runtime-Projekt der lokalen Session wird niemals ins Multiuser-Serverprojekt übertragen und kann dort nicht gespeichert werden.

Hinweis

Online-Deltaladen auf ein Bediengerät aus der lokalen Session ist nicht für alle Projektierungsszenarien geeignet. Daher ist eine genau Prüfung des Projektierungsszenarios im Vorfeld empfehlenswert.

Hinweis

Fortlaufendes Online-Deltaladen wird ausschließlich über Laden aus der Serverprojekt-Ansicht unterstützt.

Beim ersten Laden nach dem Erstellen oder Aktualisieren der lokalen Session erscheint im Dialog "Vorschau laden" die Warnung "Projekte passen nicht zusammen". Führen Sie ein Komplettladen durch.

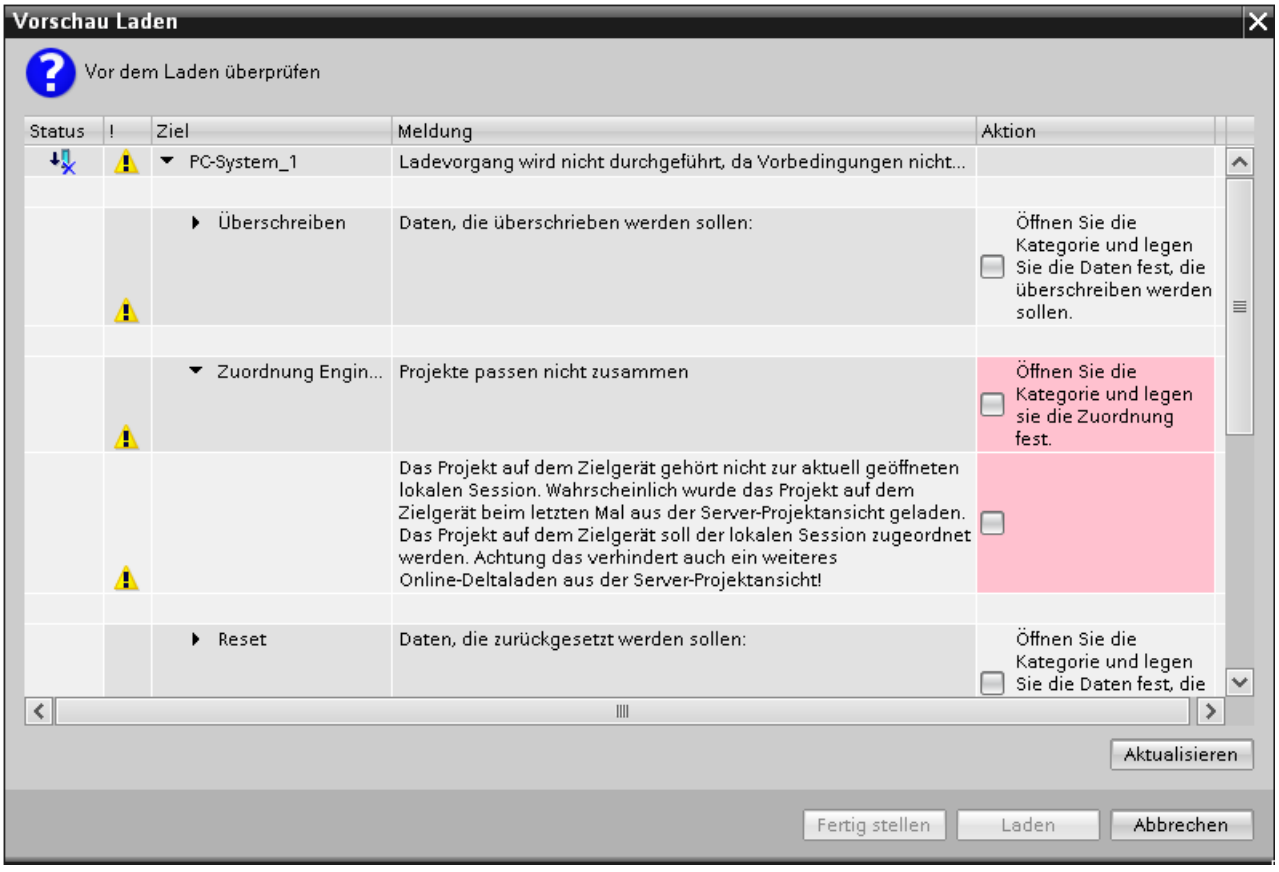

Verlust der Online-Delta-Ladefähigkeit im Serverprojekt vermeiden

Um die Online-Delta-Ladefähigkeit eines Bediengeräts im Serverprojekt aufrechtzuerhalten, dürfen alle Benutzer nur solche Änderungen der Projektierung vornehmen, die die Online-Delta-Ladefähigkeit nicht beeinträchtigen.

Wenn die Online-Delta-Ladefähigkeit eines Bediengeräts in der lokalen Session durch eine Änderung verloren gegangen ist, geht die Online-Delta-Ladefähigkeit dieses Bediengeräts im Serverprojekt nur dann verloren, wenn Sie die betreffende Änderung einchecken. Nicht alle in der lokalen Session gemeldeten Fehler führen beim Einchecken dieser Änderung zum Verlust der Online-Delta-Ladefähigkeit im Serverprojekt. Falls das Einchecken zum Verlust der Online-Delta-Ladefähigkeit im Serverprojekt führen würde, wird das automatische Einchecken angehalten. Eine Fehlermeldung informiert Sie darüber. Anschließend können Sie entscheiden, ob Sie das Einchecken fortsetzen oder abbrechen möchten.

Wenn Sie das Einchecken über "Änderungen verwerfen" abbrechen, bleibt die die Online-Delta-Ladefähigkeit des Bediengeräts erhalten. Ihre Projektierungsänderungen werden nicht ins Serverprojekt übernommen.

Weitere Informationen zum Thema "Objekte ins Serverprojekt einchecken" finden Sie unter "In der lokalen Session arbeiten".

Siehe auch

Übersetzen und Laden mit Multiuser Engineering (Übersicht) (Seite 2386)

Online-Deltaladen für Runtime Professional (Seite 2382)

15.4.7 Runtime starten

15.4.7.1 Runtime auf dem Projektierungs-PC starten

Einleitung

Nachdem Sie Ihr Projekt lokal geladen haben, können Sie am Projektierungs-PC Ihr Projekt in Runtime starten, während Sie gleichzeitig in WinCC projektieren. Das Online-Projektieren unterliegt folgenden Einschränkungen:

- Während auf dem Projektierungs-PC Runtime läuft, wird das Projekt nicht im Hintergrund übersetzt.
- Wenn Sie nach dem Beenden der Runtime das Projekt auf ein Bediengerät laden, wird der zwischenzeitlich geänderte Anteil des Projekts automatisch übersetzt. Alternativ können Sie das Übersetzen auch manuell starten.
- Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben.
- Wenn Runtime auf dem Projektierungs-PC läuft, werden die Änderungen nach einem Übersetzen in die Simulation übernommen. Nach dem Laden werden alle Änderungen in das Runtime-Projekt übernommen.
  - Einige Änderungen werden erst nach einem erneuten Starten der Runtime wirksam.
  - Änderungen im aktuell geöffneten Bild werden nach einem Bildwechsel sichtbar.
  - Bei einigen Änderungen geht die Möglichkeit verloren, Projektierungsänderungen in die laufende Runtime zu übersetzen. Wenn Ihre Änderungen zum Verlust der Delta-Ladefähigkeit führen, werden Sie beim Projektieren darauf hingewiesen, .

Hinweis

Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Control Panel.

Voraussetzung

- Auf dem Projektierungs-PC ist ein Projekt geöffnet.
- Auf dem Projektierungs-PC ist Runtime installiert.

Vorgehen

Um die Runtime am Projektierungs-PC zu starten und um Änderungen in der Projektierung in die laufende Runtime zu übertragen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Bediengerät in der Projektnavigation.
2. Wählen Sie "Online > Laden in Gerät".
3. Wenn der Dialog "Vorschau laden" angezeigt wird, aktivieren Sie das Starten von Runtime nach dem Laden.

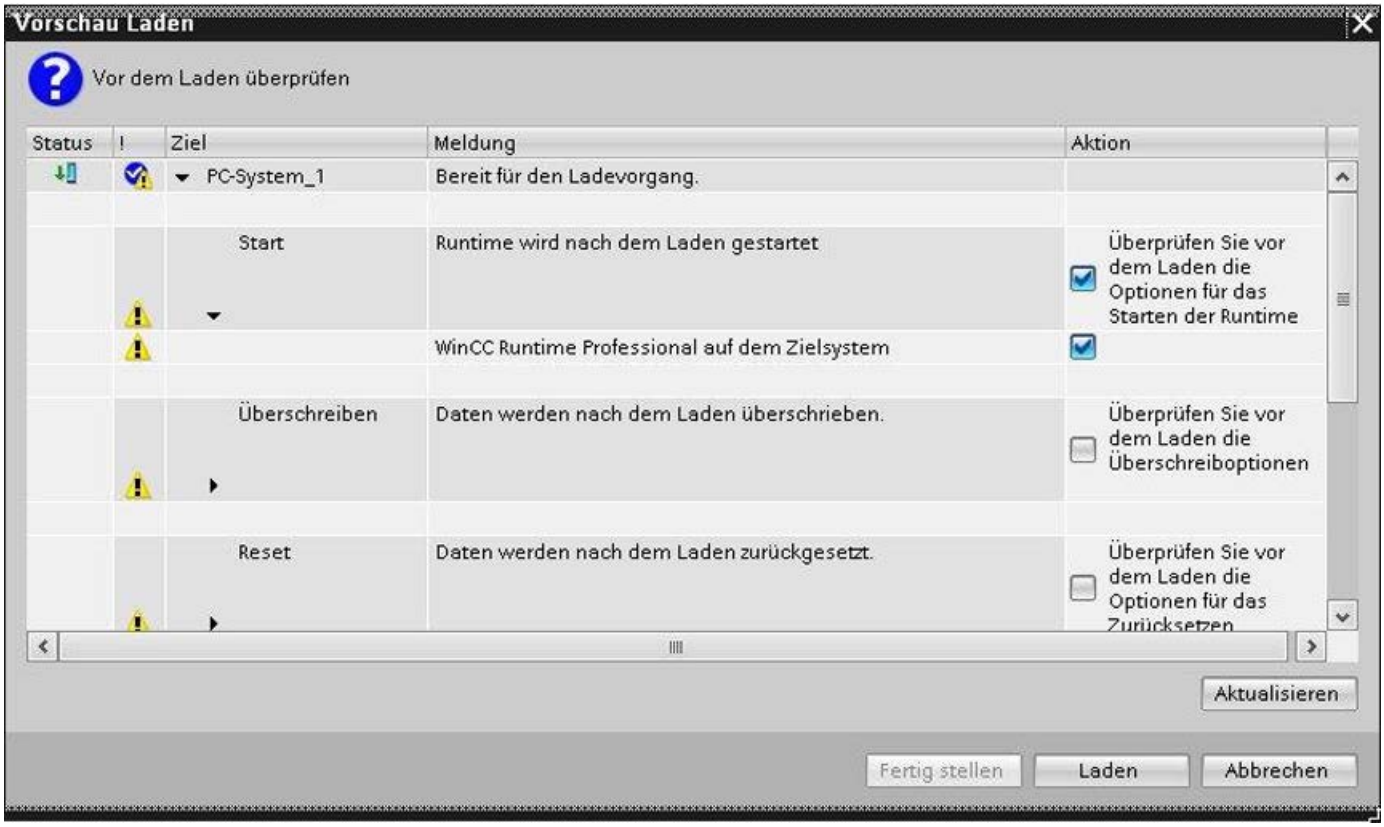

4. Klicken Sie auf "Laden".

Alternativ verwenden Sie zum Starten des Projekts das "WinCC StartTool".

Ergebnis

Auf dem Projektierungs-PC wird Runtime gestartet und das aktualisierte Projekt angezeigt.

Siehe auch

Einsatz und Einschränkungen des Online-Deltaladens (Seite 2384)

15.4.7.2 Runtime Professional starten

Einleitung

Wenn auf einem Projektierungs-PC auch WinCC Runtime installiert ist, starten Sie Runtime beim Laden. Dazu aktivieren Sie im Dialog "Vorschau laden" die entsprechende Option.

Wenn auf einem Bediengerät nur WinCC Runtime installiert ist, laden Sie das Projekt auf das Bediengerät. Um WinCC-Runtime auf dem Bediengerät zu starten, verwenden Sie das "WinCC StartTool" oder aktivieren beim Laden den anschließenden Start von WinCC Runtime.

Voraussetzungen

Auf dem Bediengerät ist "WinCC Runtime" installiert.

Hinweis

Beim Starten der Runtime eines Projekts müssen zusätzlich mindestens 100 MByte freier Speicher auf dem Speichermedium vorhanden sein.

Hinweis

Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Control Panel.

Runtime an einer Engineering Station starten

1. Selektieren Sie in der Projektnavigation von WinCC das gewünschte Projekt.
2. Wählen Sie "Online > Laden in Gerät".

3. Aktivieren Sie den Start von WinCC Runtime nach dem Laden.

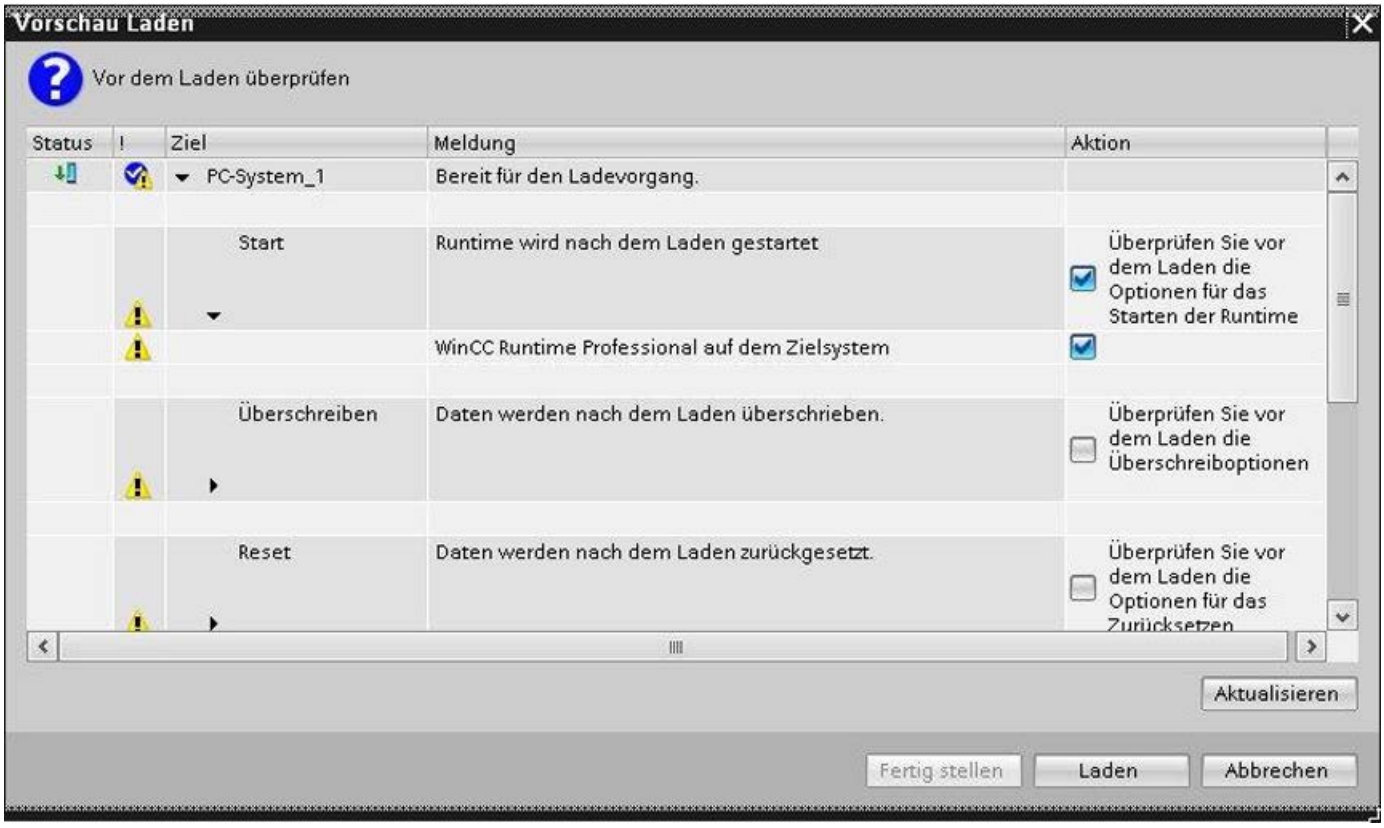

4. Klicken Sie auf "Laden".

Ergebnis

Das Projekt wird übersetzt und geladen. Beim Laden werden alle Runtime-Dienste gestartet, die Sie in den Runtime-Einstellungen des Bediengeräts aktiviert haben.

15.4.8 Grundlagen zur Bedienung in Runtime

15.4.8.1 Übersicht zur Bedienung in Runtime (Professional)

Möglichkeiten der Bedienung

Von der Hardwareausstattung hängt ab, wie Sie das Bediengerät bedienen:

- Touchscreen
  Die in den Bildern dargestellten Bedienobjekte sind berührungssensitiv. Die Bedienung unterscheidet sich grundsätzlich nicht vom Drücken mechanischer Tasten. Sie bedienen Bedienobjekte durch Berühren mit dem Finger. Zum Doppelklicken berühren Sie ein Bedienobjekt zweimal kurz hintereinander.
- Tastatur des Bediengeräts
  Die in den Bildern dargestellten Bedienobjekte werden mit den Tasten des Bediengeräts markiert und bedient.
- Externe Tastatur
- Externe Maus

Bedienungsrückmeldung von Bedienobjekten

Sobald das Bediengerät die Anwahl eines Bedienobjekts erkennt, reagiert es mit einer optischen Rückmeldung. Das Bedienobjekt erhält den Fokus und wird markiert. Die Markierung ist unabhängig von einer Kommunikation mit der Steuerung. Die Markierung ist deshalb kein Indiz dafür, dass die gewünschte Aktion tatsächlich ausgeführt wird.

Der Projekteur kann die Markierung eines Bedienobjekts auch abweichend vom Standard projektieren.

Optische Rückmeldung von Bedienobjekten

Die Art der optischen Bedienungsrückmeldung ist abhängig vom Bedienobjekt:

- Schaltflächen
  Wenn der Projekteur den 3D-Effekt projektiert hat, dann unterscheiden sich die Darstellungen für die beiden Zustände "gedrückt" und "nicht gedrückt":
  - Zustand "gedrückt":

  - Zustand "nicht gedrückt":

  Der Projekteur bestimmt das Erscheinungsbild eines markierten Feldes, z. B. Linienbreite und Farbe für den Fokus.
- Unsichtbare Schaltflächen
  Unsichtbare Schaltflächen werden standardmäßig nach dem Anwählen nicht als gedrückt dargestellt. In diesem Fall folgt keine optische Bedienungsrückmeldung.
  Der Projekteur kann unsichtbare Schaltflächen auch so projektieren, dass deren Umrisse beim Anwählen als Linien sichtbar sind. Die Umrisse bleiben so lange sichtbar, bis Sie ein anderes Bedienobjekt anwählen.
- E/A-Felder
  Wenn Sie ein E/A-Feld markieren, wird der Inhalt des E/A-Feldes farbig hinterlegt. Bei Touch-Bedienung erscheint eine Bildschirmtastatur zur Werteingabe.

15.4.8.2 Bedienung über Touchscreen

Überblick zur Bedienung über Touchscreen

Über den Touch-Screen bedienen Sie das Bediengerät oder das Projekt, das auf Ihrem Bediengerät läuft.

Touch-Screen bedienen

ACHTUNG
Beschädigung des Touch-Screens Berühren Sie den Touch-Screen nicht mit spitzen oder scharfen Gegenständen. Verzichten Sie auf schlagartige Berührungen des Touchscreen mit harten Gegenständen und vermeiden Sie eine dauerhafte Bedienung mit Gesten. Beides kann zu einer erheblichen Reduzierung der Lebensdauer bis hin zum Totalausfall des Touchscreen führen. **Auslösen unbeabsichtigter Aktionen** Das gleichzeitige Berühren mehrere Bedienobjekte kann unbeabsichtigte Aktionen auslösen. Berühren Sie immer nur ein Bedienobjekt am Bildschirm. Bedienobjekte sind berührungssensitive Darstellungen am Bildschirm des Bediengeräts.

Besonderheiten beim Bedienen über Touch-Screen

Die Bedienung über Touch-Screen zeichnet sich durch folgende Besonderheiten aus:

- Aktivieren
  Um ein Bedienelement zu aktivieren, berühren Sie den Touch-Screen mit dem Finger oder einen Touch-Stift. Um einen Doppelklick zu erzeugen, berühren Sie das Bedienelement zweimal kurz hintereinander.
- Werteingabe
  Zahlen und Buchstaben geben Sie am Touch-Screen über eine Bildschirmtastatur ein.
- Achtsame Bedienung
  Wenn Sie gleichzeitig mehrere Bedienelemente berühren, lösen Sie damit unbeabsichtigte Aktionen aus.

Werteingabe über Bildschirmtastatur

Die Bildschirmtastatur wird am Display angezeigt, sobald Sie ein Bildobjekt bedienen, das eine Eingabe erfordert. Abhängig vom Bediengerät und vom projektierten Bedienobjekt werden unterschiedliche Bildschirmtastaturen für numerische oder alphanumerische Werte angezeigt. Beim Beenden der Eingabe wird die Bildschirmtastatur wieder ausgeblendet.

Dateibrowser (Windows 8 oder höher)

Auf einem PC mit Windows 8 oder höher mit einem Touch-Screen kann der Dateibrowser-Dialog nur mit der Maus, Tastatur oder Bildschirmtatstatur ohne Verwendung von Touch-Funktion bedient werden. Auf einem Touch-Screen PC mit Windows 8 oder höher nutzen Sie den Dateibrowser-Dialog von Windows-Betriebssystem mithilfe eines Skripts.

Bildschirmtastatur

Darstellung

Die folgende Abbildung zeigt den prinzipiellen Aufbau einer Bildschirmtastatur auf einem Runtime Professional Bediengerät.

Zahlen

Buchstaben

Multitouch-Funktionen nutzen

Multitouch-Bedienung

Einleitung

RT Professional unterstützt unter Windows 7, Windows 8 und Windows 10 Multitouch-Bedienung. Sie bedienen die Objekte am Multitouch-Screen durch Berühren mit einem oder zwei Fingern. Multitouch-Bedienung wird von TIA Portal ab V13 unterstützt. Versionen V12 oder älter unterstützen nur Singletouch-Bedienung.

Voraussetzung

Für die Nutzung der Multitouch-Funktionen muss der verwendete Monitor multitouchfähig sein.

Grundlagen zu Multitouch-Bedienung

Der multitouchfähige Monitor lässt sich durch Tippen, Ziehen, Auf- und Zuziehen und Wischen bedienen und erkennt exakt die Art der Berührung. Beim Tippen berühren Sie den Monitor kurzzeitig mit einem Finger und aktivieren auf diese Weise z. B. eine Schaltfläche. Beim Ziehen berühren Sie den Monitor mit einem Finger und bewegen den Finger konstant in eine Richtung analog zum Drag&Drop mit der Maus. Beim Auf- und Zuziehen berühren Sie den Monitor mit zwei Fingern und öffnen oder schließen die Finger zueinander. Beim Wischen berühren Sie den Monitor kurz mit einem Finger und bewegen den Finger in eine Richtung.

Mithilfe von Multitouch-Funktionen scrollen Sie in Bildern, vergrößern oder verkleinern Bildausschnitte oder wechseln zwischen den Bildern in einem Bildstapel. Durch horizontales Ziehen verschieben Sie den Inhalt des Bildes nach links bzw. nach rechts. Durch vertikales Ziehen scrollen Sie in der Ansicht nach oben oder nach unten. Während Sie skalieren, wird Ihnen ein Anzeiger als Hinweis für den Skalierungsfaktor angezeigt.

Unterstützte Gesten

Unterstützte Gesten

Folgende Gesten werden unterstützt.

Geste		Funktion
	Tippen	Durch das Tippen wählen Sie z. B. eine Schaltfläche aus.
	Ziehen	Mit einem Finger können Sie z. B. in Bildern horizontal und vertikal scrollen. Durch diagonales Ziehen in Bildern scrollen Sie gleichzeitig horizontal und vertikal.

Geste		Funktion
	Skalieren	Durch Auf- und Zuziehen können Sie die Anzeige des Bildes vergrößern bzw. verkleinern. Abhängig vom projektierten Zoomfaktor lassen sich Objekte beim Skalieren ein- und ausblenden. So lassen Sie sich z. B. ein Objekt nur anzeigen, wenn ein ausreichender Zoomfaktor das erforderliche Objekt auch zeigt.
	Wischen	Durch Wischen von Mitte nach rechts oder nach links können Sie innerhalb eines Bildstapels zwischen den Bildern wechseln.

Hinweis

Bedienung mit drei oder mehr Fingern

Verwenden Sie bei der Bedienung nicht drei oder mehr Finger. Das kann zu Fehlbedienungen führen.

Hinweis

Einstellungen für Ziehen und Skalieren

Um die Funktionen "Ziehen" und "Skalieren" freizuschalten, aktivieren Sie unter "Runtime Settings > Bilder > Eigenschaften Fenster" das Optionskästchen "Schieber" und deaktivieren das Optionskästchen "Erweitertes Zoomen". Zusätzlich aktivieren Sie im Inspektorfenster des jeweiligen Bildes unter "Eigenschaften > Eigenschaften > Verschiedenes" das Optionskästchen "Erweitertes Zoomen".

Siehe auch

Ebenen und Objekte in Runtime verbergen (Seite 270)

Zweihand-Bedienung von Bedienelementen

Zweihand-Bedienung von Bedienelementen

Einleitung

WinCC unterstützt die Zweihand-Bedienung von Bedienlementen mit RT Professional. Damit wird eine sichere Bedienung von solchen Bedienelementen ermöglicht, mit denen kritische Systemeinstellungen geändert werden können, z. B. Steuervariablen mit Maschinengrenzwerten.

Gesperrte und entsperrte Bedienelemente

Für die Zweihand-Bedienung von Bedienelementen definieren Sie bestimmte Bedienelemente als "gesperrte Bedienelemente". Gesperrte Bedienelemente sind in Runtime gewöhnlich nicht bedienbar. Der Bediener kann die gesperrten Bedienelemente nur dann bedienen, wenn er gleichzeitig eine dafür vorgesehene Freigabeschaltfläche drückt.

In Runtime sind die gesperrten Bedienelemente nur dann über die Tab-Reihenfolge erreichbar, wenn gleichzeitig eine Freigabeschaltfläche gedrückt wird.

Gesperrte Bedienelemente und Freigabeschaltflächen

Sie können alle Bedienelemente als gesperrte Bedienelemente konfigurieren.

Wenigstens eine Schaltfläche im Bild müssen Sie als Freigabeschaltfläche konfigurieren. Das kann eine beliebige nicht gesperrte Schaltfläche sein.

Darstellung in Runtime

In Runtime sind die gesperrten Bedienelemente gedimmt dargestellt. Die gesperrten Bedienelemente sind dann normal sichtbar, wenn diese über die Freigabeschaltfläche entsperrt sind.

Simulation von Projekten mit Multitouch-Funktionen

WinCC unterstützt die Simulation der projektierten Multitouch-Funktionen. Voraussetzung ist, dass Sie einen multitouchfähigen Monitor einsetzen.

Bedienelemente sperren und entsperren

In Projekten für Multitouchgeräte können Sie Bedienelemente sperren und entsperren. Gesperrte Bedienelemente sind in Runtime nur dann bedienbar, wenn der Bediener gleichzeitig eine Freigabeschaltfläche drückt.

Sie können ein einzelnes oder gleichzeitig mehrere Bedienelemente sperren und entsperren.

Vorgehensweise

1. Projektieren Sie die Bedienelemente vom Typ EA-Feld, Schaltfläche oder Schieberegler.
2. Selektieren Sie das oder die gewünschten Bedienelemente.
3. Um die Bedienelemente zu sperren, deaktivieren Sie unter "Eigenschaften > Eigenschaften > Sicherheit" die Option "Bedienung zulassen".

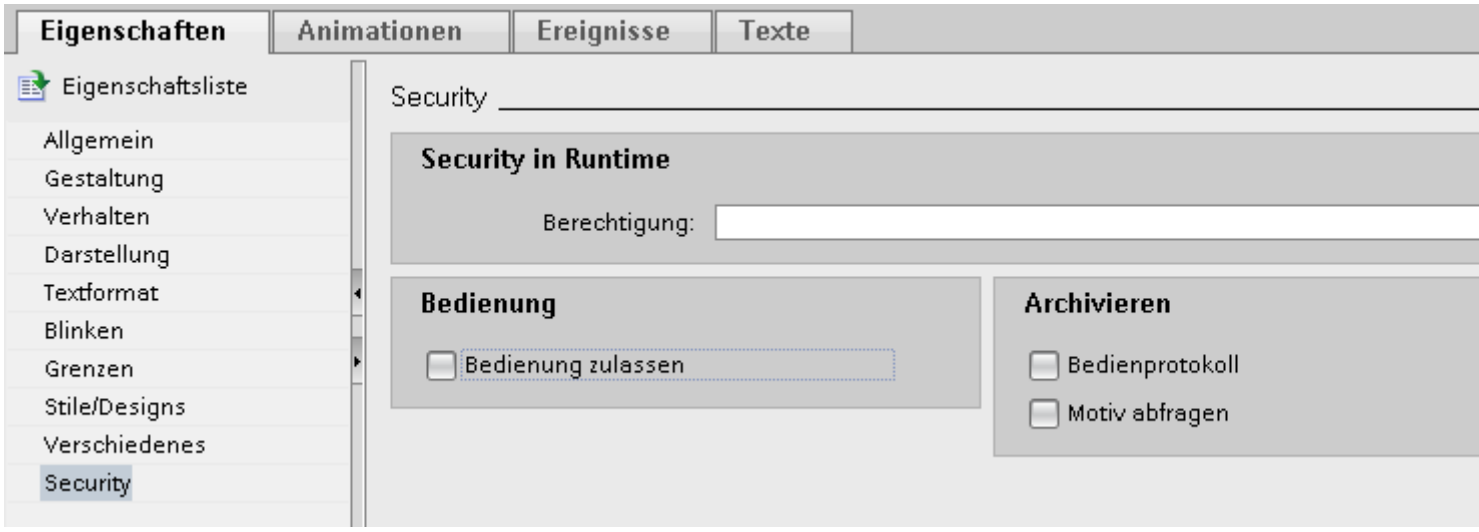

   In Runtime sind die gesperrten Bedienelemente nur dann bedienbar, wenn gleichzeitig eine Freigabeschaltfläche gedrückt wird, die mit entsprechenden Systemfunktionen dynamisiert wurde.

4. Um die Bedienelemente zu entsperren, aktivieren Sie unter "Eigenschaften > Eigenschaften > Sicherheit" die Option "Bedienung zulassen".

Freigabeschaltfläche in Bild oder Vorlage projektieren

Damit der Bediener die gesperrten Bedienelemente auf Multitouchgeräten bedienen kann, muss in demselben Bild mindestens eine Freigabeschaltfläche projektiert sein.

Projektieren Sie eine Freigabeschaltfläche so, dass beim Drücken der Freigabeschaltfläche die Bedienelemente freigegeben und beim Loslassen die Bedienelemente für die Bedienung gesperrt werden.

Wenn mehrere Objekte in einem Bild mit einer Freigabeschaltfläche entsperrt werden müssen, erweitern Sie für jedes Objekt die Funktionslisten der Freigabeschaltfläche um einen weiteren Eintrag.

Voraussetzung

- Eine oder mehrere Schaltflächen sind in einem Bild oder einer Vorlage angelegt.

Vorgehensweise

1. Selektieren Sie die Schaltfläche, die Sie als Freigabeschaltfläche projektieren möchten.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Wählen Sie das Ereignis "Linke Maustaste drücken".
4. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
5. Wählen Sie die Systemfunktion "SetzeEigenschaftDurchKonstante".
6. Wählen Sie aus der Liste "Bildname" das jeweilige Bild aus.
7. Wählen Sie aus der Liste "Bildobjekt" die jeweilige Schaltfläche aus.
8. Wählen Sie aus der Liste "Name der Eigenschaft" die Eigenschaft "Bedienung zulassen".
9. Tragen Sie bei "Wert" den Wert "1" ein.

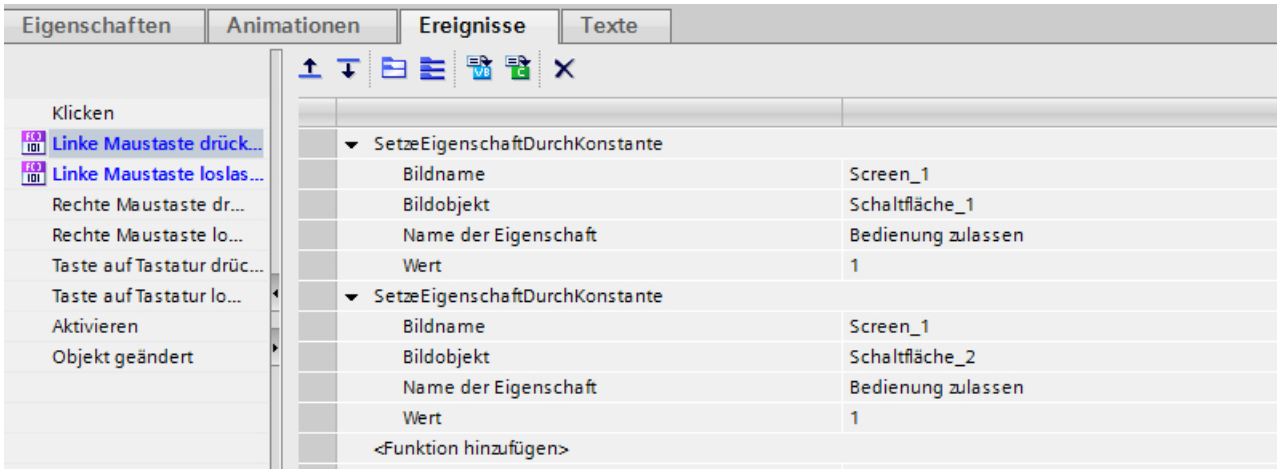

10. Damit die Bedienfreigabe beim Loslassen der Freigabeschaltfläche zurückgenommen wird, wählen Sie das Ereignis "Linke Maustaste loslassen" und tragen bei "Wert" den Wert "0" ein.

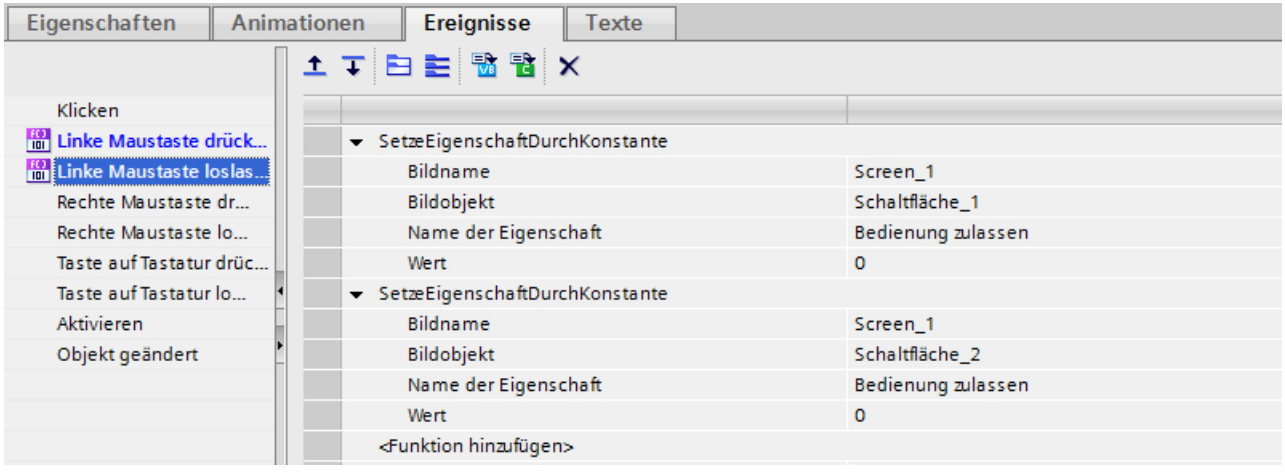

In Runtime sind die gesperrten Bedienelemente nur dann bedienbar, wenn der Bediener gleichzeitig eine der projektierten Freigabeschaltflächen drückt.

15.4.8.3 Bedienung über Tasten

Überblick zur Bedienung über Tasten

Einleitung

Mit dem Tasten des Bediengeräts bedienen Sie das Control Panel/Start Center Ihres Geräts oder das Projekt, das auf Ihrem Gerät läuft. Abhängig vom Gerät stehen Ihnen Steuertasten und Funktionstasten zur Verfügung.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

Steuertasten und Tastenkombinationen

Einleitung

Die folgenden Tabellen zeigen die Steuertasten, mit denen Sie das Projekt bedienen.

Hinweis

Die Verfügbarkeit der Steuertasten ist abhängig vom verwendeten Bediengerät.

Funktionen an Tasten-Bediengeräten lösen Sie entweder über eine Taste oder Tastenkombination aus. Bei Tastenkombinationen halten Sie die erste Taste gedrückt. Danach drücken Sie die zweite Taste.

Im Display navigieren

Taste oder Tastenkombination		Funktion	Beschreibung
TAB		Tabulator	Wählt das nächste Bedienobjekt in der Tab-Reihenfolge aus
SHIFT	TAB	Tabulator	Wählt das vorherige Bedienobjekt in der Tab-Reihenfolge aus
◀ ▶ ▼ ▲		Cursortasten	Wählen das nächste Bedienobjekt, links, rechts, oberhalb oder unterhalb des aktuellen Bildobjekts aus Navigieren im Bedienobjekt

Bedienobjekte bedienen

Taste oder Tastenkombination		Funktion	Beschreibung
ENTER		Eingabetaste	• Bedient Schaltflächen. • Übernimmt und beendet eine Eingabe • Öffnet eine Auswahlliste • Schaltet innerhalb eines Eingabefelds zwischen Zeichenmodus und Normalmodus um Im Zeichenmodus ist ein einzelnes Zeichen markiert. In diesem Modus können Sie mit den Cursortasten im Zeichensatz weiterschalten.
SHIFT	▶ ◀ ▲ ▼	Cursor positionieren	Positionieren den Cursor innerhalb eines Bedienobjekts, z. B. im E/A-Feld
←		Zeichen löschen	Löscht das Zeichen links von der aktuellen Cursor-Position
INS DEL		Zeichen löschen	Löscht das Zeichen rechts von der aktuellen Cursor-Position
ESC		Abbrechen	• Löscht eingegebene Zeichen einer Werteingabe und stellt den ursprünglichen Wert wieder her • Schließt den aktiven Dialog.
FN	HOME	Zum Anfang blättern	Blättert in einer Liste zum Anfang
HOME		Zurückblättern	Blättert in einer Liste um eine Seite zurück
FN	END	Zum Ende blättern	Blättert in einer Liste zum Ende.
END		Weiterblättern	Blättert in einer Liste um eine Seite weiter
ALT	▼	Auswahlliste öffnen	Öffnet eine Auswahlliste
CTRL	ENTER	Wert übernehmen	Übernimmt den ausgewählten Wert in der Auswahlliste, ohne die Liste zu schließen

Tastenkombination eingeben

Taste	Funktion	Zweck
A-Z	Umschalten (Ziffern/Buchstaben)	Schaltet die Belegung der Eingabetasten von Ziffern auf Buchstaben um • Keine LED leuchtet: Die Ziffern-Belegung ist aktiv. Einmaliges Drücken der Taste schaltet auf Buchstaben-Belegung um. • Die rechte oder die linke LED leuchtet: Die linke oder rechte Buchstaben-Belegung ist aktiv. Jedes Drücken der Taste schaltet zwischen der linken Buchstaben-Belegung, der rechten Buchstaben-Belegung und der Ziffern-Belegung um.
SHIFT	Umschalten (Groß-/Kleinschreibung)	Verwendung in Tastenkombinationen, z. B. Umschalten auf Großbuchstaben
FN	Auf zusätzliche Tastenbelegung umschalten	Auf einigen Tasten finden Sie in der linken unteren Ecke blaue Sonderzeichen, z. B. das Prozentzeichen "%". Um diese Zeichen einzugeben, drücken Sie gleichzeitig die links abgebildete Sonderzeichen-Taste.
CTRL	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. Navigation in Kurvendarstellung
ALT	Allgemeine Steuerfunktion	Verwendung in Tastenkombinationen, z. B. bei Bildobjekt "Beobachtungstabelle"

Meldungen quittieren

Taste	Funktion	Zweck
ACK	Quittieren	Quittiert die aktuell angezeigte Meldung oder alle Meldungen einer Meldegruppe Die LED leuchtet, solange unquittierte Meldungen anstehen.

Hilfetext anzeigen

Taste	Funktion	Beschreibung
HELP	Hilfetext anzeigen	Öffnet zum markierten Objekt ein Fenster mit dem projektierten Hilfetext, , z. B. für eine Meldung oder ein E/A-Feld Wenn zum markierten Objekt ein Hilfetext existiert, leuchtet die LED.

Taste oder Tastenkombination

Funktionstasten

Die Belegung der Funktionstasten (F1, F2, F3 …) wird bei der Projektierung festgelegt.

Funktionstasten mit globaler Funktionsbelegung

Eine global belegte Funktionstaste löst unabhängig vom angezeigten Bild immer dieselbe Aktion am Bediengerät oder in der Steuerung aus. Eine solche Aktion ist z. B. das Aktivieren eines Bildes oder das Schließen eines Meldefensters.

Funktionstasten mit lokaler Funktionsbelegung

Eine Funktionstaste mit lokaler Funktionsbelegung ist bildspezifisch und damit nur innerhalb des aktiven Bildes wirksam.

Die Funktion einer lokal belegten Funktionstaste kann von Bild zu Bild unterschiedlich sein.

Innerhalb eines Bildes hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

Funktionstasten bedienen

Hinweis

Funktionstaste nach einem Bildwechsel bedienen

Wenn Sie nach einem Bildwechsel eine Funktionstaste bedienen, kann die zugehörige Funktion im neuen Bild bereits angestoßen werden, bevor das Bild vollständig aufgebaut ist.

Direkttasten

Einleitung

Direkttasten setzen direkt vom Bediengerät aus Bits im Peripheriebereich einer SIMATIC S7.

Direkttasten ermöglichen Tastenbedienungen mit kurzen Reaktionszeiten, die z. B. Voraussetzung für den Tippbetrieb sind.

Hinweis

Die Direkttasten sind auch aktiv, wenn sich das Bediengerät in der Betriebsart "Offline" befindet.

Hinweis

Wenn Sie bei laufendem Projekt eine Funktionstaste mit Direkttastenfunktionalität bedienen, wird die Direkttastenfunktion unabhängig vom aktuellen Bildschirminhalt immer ausgeführt.

Hinweis

Sie verwenden Direkttasten nur bei Kopplung über PROFIBUS DP oder PROFINET IO.

Direkttasten führen zu einer zusätzlichen Grundlast im Bediengerät.

Direkttasten

Folgende Objekte können als Direkttaste projektiert sein:

- Schaltflächen
- Funktionstasten

Abhängig vom Bediengerät und von der Projektierung sind Direkttasten bildspezifisch projektiert.

Weitere Hinweise zu Direkttasten finden Sie in Ihrer Anlagendokumentation.

15.4.8.4 Navigieren im Display (BS)

Einleitung

Sie navigieren auf dem Display Ihres Bediengeräts folgendermaßen:

- Zwischen den projektierten Bildobjekten
- Innerhalb von Bildobjekten
  Wenn Sie ein komplexes Bildobjekt aktivieren, wechselt der Cursorfokus in das Bildobjekt und folgt der dortigen Tab-Reihenfolge.
- In Tabellen von Bildobjekten

Vorgehen

- Um in der festgelegten Tab-Reihenfolge zu navigieren, drücken Sie die Taste <TAB>.
- Um frei zwischen den Bedienelementen zu navigieren, drücken Sie die Cursortasten.

Abhängig von der Projektierung Ihres Bediengeräts, verwenden Sie zum Navigieren auch Funktionstasten oder Tastenkombinationen.

Wenn Sie Ihr Bediengerät über den Touchscreen oder mit der Maus bedienen, navigieren Sie implizit durch das Auslösen einer gewünschten Aktion. Dazu berühren oder klicken Sie das Bedienelement.

Ergebnis

Die Bedienelemente erhalten entsprechend der gewählten Reihenfolge den Cursorfokus. Sie können an dem ausgewählten Bedienelement eine Aktion auslösen.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.4.8.5 Aktion auslösen

Einleitung

Eine Aktion an einem Bedienelement auslösen kann Folgendes bedeuten:

- Ein Befehl wird ausgeführt.
  Beispiel: Um ein Skript zu triggern oder eine vordefinierte Funktion auszuführen, klicken Sie auf eine Schaltfläche.
- Ein Objekt wird aktiviert.
  Beispiel: Um einen Wert einzugeben, aktivieren Sie eine Tabellenzelle mit der Taste <Enter>.

Voraussetzung

- Sie sind zum Bedienelement navigiert, an dem Sie die Aktion auslösen wollen.
- Das Bedienelement hat den Cursorfokus.

Vorgehen

- Drücken Sie die Taste <Enter>.
  Oder
- Berühren Sie das Bedienelement am Touchscreen einfach oder zweimal kurz hintereinander.
  Oder
- Klicken oder doppelklicken Sie das Bedienelement mit der Maus.

Ergebnis

Folgende Ergebnisse sind z. B. möglich:

- Der gewünschte Befehl wird ausgeführt.
- Die Bildschirmtastatur wird geöffnet und/oder der Cursor blinkt im Eingabebereich des Bedienelements.
- Das Element ist ausgewählt und kann bewegt werden.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.4.8.6 Wert eingeben

Einleitung

Je nach Eingabeformat geben Sie numerische oder alphanumerische Werte in ein Eingabefeld ein.

Abhängig von der vorhandenen Hardware geben Sie diese Werte über die Bildschirmtastatur, die Steuertasten des Bediengerätes oder eine externe Tastatur ein.

Voraussetzung

- Das Objekt ist ein Eingabe- oder Tabellenfeld.
- Das Bedienelement ist aktiviert.

Wert eingeben

1. Geben Sie den gewünschten Wert ein.
2. Um den Wert zu bestätigen und das Feld zu verlassen, drücken Sie die Taste <Enter>.
3. Um den Wert zu verwerfen und das Feld zu verlassen, drücken Sie die Taste <Esc>.

Ergebnis

Ein Wert ist eingegeben oder verworfen. Sie navigieren bei Bedarf zum nächsten Bedienelement.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.4.8.7 Bedienelemente bewegen

Einleitung

Sie bewegen in Runtime bei Bedarf die beweglichen Bedienelemente eines Bildobjekts mit der Maus oder über den Touchscreen, z. B. einen Schieberegler oder einen Scrollbalken. Im Folgenden ist die Bedienung mit Tastatur beschrieben.

Voraussetzung

- Ein bewegliches Bedienelement ist aktiviert.

Vorgehen

- Um das Bedienelement zu bewegen, gehen Sie abhängig vom Bedienobjekt folgendermaßen vor:
  - Standard für Touchscreen:: Drücken Sie die Cursortasten.
  - Standard für Tastengeräte:: Drücken Sie <SHIFT> und die Cursortasten.
  - Schalter: Drücken Sie <ENTER>
  - Schieberegler: Drücken Sie <Bild Ab> oder <Bild Auf>

1. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Vorgehen beim Schieberegler

1. Um das Bedienelement zu bewegen, drücken Sie die Tasten " Cursortasten.
2. Um die Bewegung abzuschließen, navigieren Sie zu einem anderen Bildobjekt oder Bedienelement.

Ergebnis

Die Position des beweglichen Bedienelements und die Anzeige im Bildobjekt haben sich verändert.

Weitere Informationen dazu finden Sie in der Betriebsanleitung Ihres Bediengeräts.

15.4.8.8 Hilfetext anzeigen (Professional)

Verwendung

Mit Tooltips stellt Ihnen der Projekteur zusätzliche Informationen und Bedienhinweise zur Verfügung. Der Projekteur kann Tooltips zu Bildern und Bedienobjekten projektieren.

Der Tooltip eines E/A-Feldes kann z. B. Hinweise zum einzugebenden Wert enthalten.

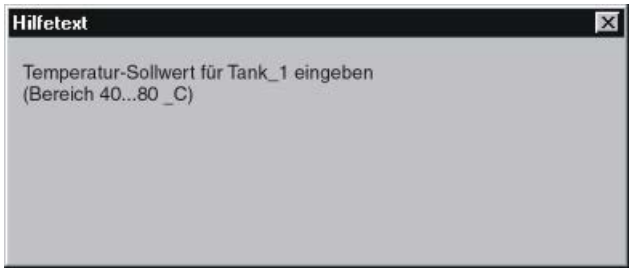

Vorgehen

Abhängig von der Projektierung können Sie Tooltips über ein davor vorgesehenes Bedienobjekt aufrufen.

Nähere Hinweise hierzu finden Sie gegebenenfalls in Ihrer Anlagendokumentation.

15.4.8.9 Runtime-Sprache umstellen

Einleitung

Das Projekt auf dem Bediengerät kann mehrsprachig sein. Zum Wechseln der am Bediengerät eingestellten Sprache während des laufenden Betriebs ist ein entsprechendes Bedienobjekt projektiert.

Nach dem Starten des Projekts ist immer die zuletzt aktive Sprache eingestellt.

Voraussetzung

- Die gewünschte Sprache für das Projekt ist am Bediengerät verfügbar.
- Die Funktion für die Sprachumschaltung ist mit einem Bedienobjekt verknüpft, z. B. mit einer Schaltfläche.

Sprache auswählen

Zwischen den Sprachen können Sie jederzeit umschalten. Nach dem Aufruf der Funktion für die Sprachumschaltung werden die sprachabhängigen Objekte in der neuen Sprache angezeigt.

Um die Sprache in Runtime umzuschalten, haben Sie folgende Möglichkeiten:

- Über ein projektiertes Bedienobjekt schalten Sie in einer Auswahl von Sprachen weiter.
- Über ein projektiertes Bedienobjekt wählen Sie die gewünschte Sprache direkt aus.

15.4.8.10 Runtime-Daten exportieren

Daten exportieren

Um in Runtime Daten in eine CSV-Datei zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Daten exportieren" in der Symbolleiste des Anzeige- und Bedienobjekts.
   Je nach Projektierung öffnet sich ein Dialog.
2. Wenn Sie über die entsprechenden Berechtigungen verfügen, wählen Sie die Datei und das Verzeichnis für den Export aus.

   Hinweis
   Dateiname

   Auch wenn Sie im Dateinamen eine andere Dateinamenserweiterung als "*.csv" angeben, werden die Runtime-Daten in eine CSV-Datei exportiert.

3. Starten Sie den Export.

Hinweis
Export-Dialog

Wenn kein Dialog angezeigt wird, werden die Daten automatisch in die voreingestellte Datei exportiert.

15.5 In die Projektierung springen

15.5.1 In die Projektierung springen

Einleitung

Mit WinCC Runtime Advanced haben Sie die Möglichkeit, einen kontextbezogenen Wechsel aus HMI in die Projektierung im TIA Portal Engineering System zu definieren. Damit kann ein Bediener aus Runtime zu einem bestimmten Projekt oder zu einem Baustein im Engineering System wechseln, um die Daten zu überprüfen oder zu modifizieren. Außerdem kann der Bediener andere Objekte des geöffneten Projekts in den entsprechenden Editoren des TIA Portals öffnen oder bearbeiten. Der Wechsel in die Projektierung ermöglicht schnelle und einfache Diagnose von Störungen.

Je nach Anforderung können Sie einen Lese-/Schreibzugriff oder einen Nur-Lese-Zugriff im Lesemodus projektieren.

Wechsel zum TIA Portal projektieren

Sie projektieren die Systemfunktion ZeigeBlockinTIAPortal, für den Sprung aus einer Meldeanzeige oder einer GRAPH-Übersicht.

Sie wechseln zum Projekt im TIA Portal mithilfe der Systemfunktion, die Sie entweder an einer Schaltfläche für die Meldeanzeige oder an einer GRAPH-Übersicht projektieren.

Systemfunktion

Mithilfe der Systemfunkton ZeigeBlockInTIAPortalVonMeldung projektieren Sie einen Wechsel zu einem Ereignis in Ihrem Programm, welches einen Prozessalarm ausgelöst hat. Damit haben Sie die Möglichkeit, auf dieses Ereignis entsprechend zu reagieren und das Programm auf Störungen, z. B. einen fehlenden Kontakt im Interlock, zu untersuchen.

Diese Systemfunktion projektieren Sie an ein Bildobjekt, z. B. eine Schaltfläche, oder an eine Schaltfläche der Symbolleiste in einer Meldeanzeige. Beim Klicken der Schaltfläche wird der notwendige Kontext vom System vorgemerkt. Nach dem Wechsel ins Engineering System wird automatisch die entsprechende Stelle im Programmcode angezeigt.

Die Beschreibung der Systemfunktion finden Sie unter "Systemfunktionen für Runtime Advanced > ZeigeBlockInTIAPortal"

In ein geschütztes Projekt springen

Sie aktivieren den Projektschutz für Ihr gesamtes Projekt und projektieren Rollen und Benutzer für den Zugriff. Beim Wechsel aus Runtime zu einem geschützten Projekt funktioniert die Vergabe des Zugriffs nach den Funktionsrechten, die Sie für den entsprechenden Benutzer definiert haben. Z. B. wird das im Projektschutz gesetzte Funktionsrecht "Projekt schreibgeschützt öffnen" den Lese- und Schreibzugriff aufheben, den Sie in WinCC projektiert haben.

Geben Sie beim Öffnen eines geschützten Projekts den Benutzernamen und das Kennwort in einem Anmeldedialog ein.

15.5.2 Lesemodus im TIA Portal

Einleitung

Mit WinCC Runtime Adavnced projektieren Sie einen kontextbezogenen Wechsel aus HMI ins TIA Portal. Dabei haben Sie die Möglichkeit, den Einsprung mit zwei verschiedenen Zugiffsarten zu projektieren:

- Lese- und Schreibzugriff
  Nach dem Einsprung ins TIA Portal kann der Bediener alle Projektdaten im Engineering System bearbeiten und hat den Lese- und Schreibzugriff auf die Hardware.
- Nur-Lese-Zugriff
  Nach dem Einsprung ins TIA Portal hat der Bediener nur einen eingeschränkten Zugriff auf die Projektdaten. Der Bediener darf die Daten im Engineering System nicht verändern und hat nur den Online-Lesezugriff auf die Hardware.
  Der Nur-Lese-Zugriff wird mithilfe des Lesemodus realisiert. Im Lesemodus ist die Veränderung der Projektdaten nicht möglich.

Lesemodus

Im Lesemodus darf der Bediener nur diejenigen Projektierungs- und Diagnoseaufgaben durchführen, die keine Veränderung der Projekt- und Online-Daten erfordern. Online-Lesezugriff ist auch im Lesemodus für die weitere Kriterienanalyse im Fall eines Prozessfehlers möglich.

Im Lesemodus kann der Bediener für die Diagnose der Störungen und Analyse des Programms die Navigationsfunktionen des TIA Portals nutzen.

- Projektnavigation im geöffneten Projekt.
  Über die Projektnavigation gelangt der Bediener zu allen im Projekt verfügbaren Objekten und kann diese in den Editoren, die den Lesemodus unterstützen, schreibgeschützt öffnen. Editoren, die den Lesemodus nicht unterstützen, werden nicht geöffnet.
- Funktionen "Gehe zu" Lese-/Schreibzugriff, Definition und Gerät.
  Die Gehe-zu-Funktionen helfen dem Bediener bei der Analyse der Prozesszuständen und Daten.
- Editoren "Querverweise", "Querverweis-Information" und "Programminformationen".
  Die Querverweis-Funktionen bieten dem Bediener einen Überblick über die Verwendung von Objekten innerhalb des Projekts.

Aktionen im Lesemodus

Im Lesemodus kann der Bediener im Engineering System folgende Aktionen durchführen:

- Zwischen der Portalansicht und der Projektansicht umschalten.
- Ein anderes Projekt im TIA Portal per Hauptmenü-Befehl oder per Schaltfläche der Symbolleiste öffnen.
- Auf ein beliebiges verbundenes Gerät zugreifen.
- TIA Portal Engineering System per Hauptmenü-Befehl verlassen.

Editoren im Lesemodus

Folgende Editoren des TIA Portals können im Lesemodus geöffnet werden:

- Programmiereditor zur Anzeige von
  - Programmbausteinen in Programmiersprachen FUP, KOP, GRAPH, AWL, SCL
  - Datenbausteinen und UDTs
  - ProDiag-Funktionsbausteinen
- Variablentabelle
- Programminformationen
- Querverweisliste
- Querverweis-Information
- Geräte & Netze
- HMI-Editoren
- Safety Administration Editor
- Durchsuchen des Projekts

Einschränkungen im Lesemodus

Im Lesemodus kann der Bediener folgende Aktionen im TIA Portal nicht durchführen:

- Projekt hochladen, Projekt herunterladen, Projekt übersetzen
- Simulation starten
- Neue Objekte anlegen
- Projekt bearbeiten
- Variablen im Programmstatus steuern
- Beobachtungstabelle und Forcetabelle anzeigen
- Editor "Online & Diagnose" öffnen
- Editoren von Technologieobjekten öffnen
- Editoren von lokalen Baugruppen öffnen

15.6 In die Projektierung springen

15.6.1 In die Projektierung springen

Einleitung

Mit WinCC Professional haben Sie die Möglichkeit, einen kontextbezogenen Wechsel aus HMI in die Projektierung im TIA Portal Engineering System zu definieren. Damit kann ein Bediener aus Runtime zu einem bestimmten Projekt oder zu einem Baustein im Engineering System wechseln, um die Daten zu überprüfen oder zu modifizieren. Außerdem kann der Bediener andere Objekte des geöffneten Projekts in den entsprechenden Editoren des TIA Portals öffnen oder bearbeiten. Der Wechsel in die Projektierung ermöglicht schnelle und einfache Diagnose von Störungen.

Je nach Anforderung können Sie einen Lese-/Schreibzugriff oder einen Nur-Lese-Zugriff im Lesemodus projektieren.

Wechsel zum TIA Portal projektieren

Für den Wechsel zum Projekt im TIA Portal stehen Ihnen je nach Anwendungsfall zwei Möglichkeiten zur Verfügung:

- Wechsel mithilfe der vier Runtime API-Funktionen
  Sie projektieren ein Bildobjekt, das eine der vier verfügbaren API-Funktionen in einem C-Script ausführt.
- Wechsel mithilfe der Systemfunktion
  Sie projektieren eine Systemfunktion, die es dem Bediener ermöglicht, nach dem Auftreten eines Prozessalarms zur Programmstelle zu wechseln, die den Prozessalarm ausgelöst hat.

Runtime API-Funktionen

Mit den API-Funktionen können Sie von einem Bild in WinCC Runtime direkt zum Programmcode wechseln. Beim Einsprung wird zunächst geprüft, ob das TIA Portal bereits geöffnet ist. Ist dies nicht der Fall, wird das TIA Portal automatisch gestartet. Im TIA Portal wird der entsprechende Editor geöffnet und die Verwendungsstelle, die Zuweisung, der Call oder der Schritt gesucht.

Nutzen Sie eine der vier API-Funktionen, um ein Projekt im TIA Portal zu öffnen:

- OpenTIAPortalProject öffnet das Projekt im TIA Portal.
- OpenTIAPortalIECPLByCall wird für die Sprachen KOP und FUP verwendet und zeigt die Vorverschaltung eines Netzwerkeingangs eines Standardbausteins.
- OpenTIAPortalIECPLByAssignment wird für die Sprachen KOP und FUP verwendet und zeigt die Zuweisung auf einen Operanden und dessen Vorverschaltung.
- OpenTIAPortalS7GraphByBlock wird für die Sprache GRAPH verwendet und zeigt einen Schritt in einer GRAPH-Ablaufkette.

Die Beschreibung der API-Funktionen finden Sie unter "Schnittstellen > Runtime API > Funktionen zur Anzeige von PLC-Code"

Systemfunktion

Mithilfe der Systemfunkton ZeigeBlockInTiaPortalVonMeldung projektieren Sie einen Wechsel zu einem Ereignis in Ihrem Programm, welches einen Prozessalarm ausgelöst hat. Damit haben Sie die Möglichkeit, auf dieses Ereignis entsprechend zu reagieren und das Programm auf Störungen, z. B. einen fehlenden Kontakt im Interlock, zu untersuchen.

Diese Systemfunktion projektieren Sie an ein Bildobjekt, z. B. eine Schaltfläche, oder an eine Schaltfläche der Symbolleiste in einer Meldeanzeige. Beim Klicken der Schaltfläche wird der notwendige Kontext vom System vorgemerkt. Nach dem Wechsel ins Engineering System wird automatisch die entsprechende Stelle im Programmcode angezeigt.

Die Beschreibung der Systemfunktion finden Sie unter "Systemfunktionen für Runtime Professional > ZeigeBlockInTiaPortalVonMeldung" (Seite 1772)

In ein geschütztes Projekt gehen

Sie aktivieren den Projektschutz für Ihr gesamtes Projekt und projektieren Rollen und Benutzer für den Zugriff. Beim Wechsel aus Runtime zu einem geschützten Projekt funktioniert die Vergabe des Zugriffs nach den Funktionsrechten, die Sie für den entsprechenden Benutzer definiert haben. Z. B. wird das im Projektschutz gesetzte Funktionsrecht "Projekt schreibgeschützt öffnen" den Lese- und Schreibzugriff aufheben, den Sie in WinCC projektiert haben.

Geben Sie beim Öffnen eines geschützten Projekts den Benutzernamen und das Kennwort in einem Anmeldedialog ein.

15.6.2 Lesemodus im TIA Portal

Einleitung

Mit WinCC Professional projektieren Sie einen kontextbezogenen Wechsel aus HMI ins TIA Portal. Dabei haben Sie die Möglichkeit, den Einsprung mit zwei verschiedenen Zugiffsarten zu projektieren:

- Lese- und Schreibzugriff
  Nach dem Einsprung ins TIA Portal kann der Bediener alle Projektdaten im Engineering System bearbeiten und hat den Lese- und Schreibzugriff auf die Hardware.
- Nur-Lese-Zugriff
  Nach dem Einsprung ins TIA Portal hat der Bediener nur einen eingeschränkten Zugriff auf die Projektdaten. Der Bediener darf die Daten im Engineering System nicht verändern und hat nur den Online-Lesezugriff auf die Hardware.
  Der Nur-Lese-Zugriff wird mithilfe des Lesemodus realisiert. Im Lesemodus ist die Veränderung der Projektdaten nicht möglich.

Lesemodus

Im Lesemodus darf der Bediener nur diejenigen Projektierungs- und Diagnoseaufgaben durchführen, die keine Veränderung der Projekt- und Online-Daten erfordern. Online-Lesezugriff ist auch im Lesemodus für die weitere Kriterienanalyse im Fall eines Prozessfehlers möglich.

Im Lesemodus kann der Bediener für die Diagnose der Störungen und Analyse des Programms die Navigationsfunktionen des TIA Portals nutzen.

- Projektnavigation im geöffneten Projekt.
  Über die Projektnavigation gelangt der Bediener zu allen im Projekt verfügbaren Objekten und kann diese in den Editoren, die den Lesemodus unterstützen, schreibgeschützt öffnen. Editoren, die den Lesemodus nicht unterstützen, werden nicht geöffnet.
- Funktionen "Gehe zu" Lese-/Schreibzugriff, Definition und Gerät.
  Die Gehe-zu-Funktionen helfen dem Bediener bei der Analyse der Prozesszuständen und Daten.
- Editoren "Querverweise", "Querverweis-Information" und "Programminformationen".
  Die Querverweis-Funktionen bieten dem Bediener einen Überblick über die Verwendung von Objekten innerhalb des Projekts.

Aktionen im Lesemodus

Im Lesemodus kann der Bediener im Engineering System folgende Aktionen durchführen:

- Zwischen der Portalansicht und der Projektansicht umschalten.
- Ein anderes Projekt im TIA Portal per Hauptmenü-Befehl oder per Schaltfläche der Symbolleiste öffnen.
- Auf ein beliebiges verbundenes Gerät zugreifen.
- TIA Portal Engineering System per Hauptmenü-Befehl verlassen.

Editoren im Lesemodus

Folgende Editoren des TIA Portals können im Lesemodus geöffnet werden:

- Programmiereditor zur Anzeige von
  - Programmbausteinen in Programmiersprachen FUP, KOP, GRAPH, AWL, SCL
  - Datenbausteinen und UDTs
  - ProDiag-Funktionsbausteinen
- Variablentabelle
- Programminformationen
- Querverweisliste
- Querverweis-Information
- Geräte & Netze
- HMI-Editoren

- Safety Administration Editor
- Durchsuchen des Projekts

Einschränkungen im Lesemodus

Im Lesemodus kann der Bediener folgende Aktionen im TIA Portal nicht durchführen:

- Projekt hochladen, Projekt herunterladen, Projekt übersetzen
- Simulation starten
- Neue Objekte anlegen
- Projekt bearbeiten
- Variablen im Programmstatus steuern
- Beobachtungstabelle und Forcetabelle anzeigen
- Editor "Online & Diagnose" öffnen
- Editoren von Technologieobjekten öffnen
- Editoren von lokalen Baugruppen öffnen

15.6.3 Lesemodus projektieren

Einleitung

In WinCC Professional haben Sie die Möglichkeit, den kontextbezogenen Wechsel in das TIA Portal im Lesemodus zu projektieren. Lesemodus erlaubt dem Bediener keine Veränderung der Projektdaten.

Lesemodus über eine Runtime API-Funktion projektieren

Wenn Sie den Wechsel ins TIA Portal mithilfe der API-Funktionen in C-Skripten realisieren, projektieren Sie die Verwendung des Lesemodus als einen Parameter der API-Funktion. Mithilfe eines Bits im Bitfeld dwFlags setzen Sie den Nur-Lese-Zugriff oder den Lese-/Schreibzugriff für den Bediener. Je nach Bitwert, wird das TIA Portal im Lesemodus (TRUE) oder nicht im Lesemodus (FALSE) geöffnet.

Die Beschreibung der API-Funktionen finden Sie unter "Schnittstellen > Runtime API > Funktionen zur Anzeige von PLC-Code".

Lesemodus über die Systemfunktion projektieren

Wenn Sie den Einsprung ins TIA Portal mithilfe der Systemfunkton ZeigeBlockInTiaPortalVonMeldung projektieren, setzen Sie den Parameter Lesemodus:

- TRUE (Standard): TIA Portal wird im Lesemodus geöffnet, der Benutzer kann an Projektdaten keine Änderungen vornehmen.
- FALSE: TIA Portal wird mit der Lese- und Schreibberechtigung geöffnet.

Die Beschreibung der Systemfunktion finden Sie unter "Systemfunktionen für Runtime Professional > ZeigeBlockInTiaPortalVonMeldung". (Seite 1772)

Siehe auch

ZeigeBlockInTiaPortalVonMeldung (Seite 1772)

15.6.4 Beispiel: Lesemodus mithilfe der API-Funktion projektieren

Einleitung

Das nachfolgende Beispiel zeigt den Programmcode für den Aufruf des Wechsels ins TIA Portal. Aktivierung des Lesemodus hängt von der Zugriffsberechtigung des Bedieners ab, der sich angemeldet hat.

Voraussetzung

- In der Benutzerverwaltung ist die Gruppe "Supervisors" angelegt
- Der Gruppe "Supervisors" ist die Berechtigung "Go to TIA Portal" mit der Nummer 4 zugewiesen

Beispiel

```
#pragma code("kopapi.dll")
#include "kopapi.h"
#pragma code()
#pragma code("UseAdmin.dll")
#include "pwrt_api.h"
#pragma code()

BOOL result;
CMN_ERROR error;

const DWORD permLevelTiaP = 4;

DWORD dwFlags = KOPAPI_FLAG_TIAPORTAL_OPEN_BRINGINFRONT |

KOPAPI_FLAG_TIAPORTAL_OPEN_READONLY;
char* plc = "PLC_1";
char* contBlock = "OPC1_DB";
char* operand = "#s1";

if(PWRTCheckPermission(permLevelTiaP, 1))
{

dwFlags &= ~KOPAPI_FLAG_TIAPORTAL_OPEN_READONLY;
}

result = OpenTIAPortalIECPLByAssignment(dwFlags,
 GetTagChar("TiaPortalProjectPath"), plc, contBlock, operand, 444
"TIAPErrorTag", &error);
```

Ergebnis

Es wird geprüft, ob der angemeldete Benutzer die Bedienberechtigung besitzt.

Ein Bediener, der der Gruppe "Supervisors" zugeordnet ist, hat die Berechtigung ins TIA Portal zu wechseln. Ein Benutzer, der dieser Gruppe nicht zugeordnet ist, kann zum TIA Portal nicht wechseln.

15.7 Wartung des Bediengeräts

15.7.1 Übersicht zur Wartung des Bediengeräts (Basic Panels)

Aufbau

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts und ihren Bezug zum Engineering System.

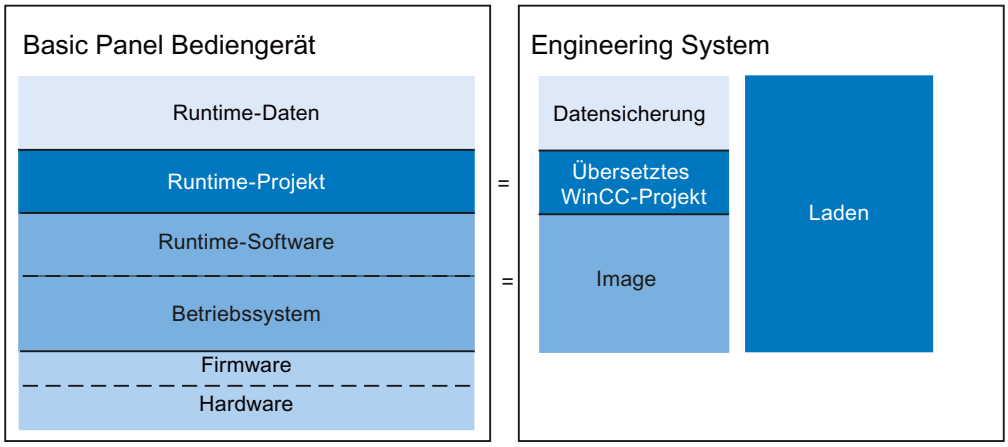

Runtime-Daten

Die Runtime-Daten werden während des laufenden Betriebs der Anlage erzeugt und auf dem Bediengerät gespeichert. Zu ihnen gehören z. B. Rezepturen und Daten zur Benutzerverwaltung. Diese Daten werden beim Laden überschrieben. Sichern Sie bei Bedarf diese Daten vor den Laden eines Runtime-Projekts.

Runtime-Projekt

Das Runtime-Projekt enthält die übersetzten Projektierungsdaten für ein Bediengerät. Sie laden das Runtime-Projekt aus WinCC auf das Bediengerät.

Runtime-Software und Betriebssystem

Runtime-Software und Betriebssystem eines Bediengeräts bilden gemeinsam das Image. Für ein Bediengerät gibt es verschiedene Images. Alle Images eines Bediengeräts sind in WinCC verfügbar. Abhängig von der Projektierung laden Sie bei Bedarf das passende Image gemeinsam mit dem Runtime-Projekt auf das Bediengerät.

Firmware und Hardware

Im Auslieferungszustand des Bediengeräts sind Firmware und Hardware fertig konfiguriert.

15.7.2 Übersicht zur Wartung des Bediengeräts (Panels)

Aufbau

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts und ihren Bezug zum Engineering System.

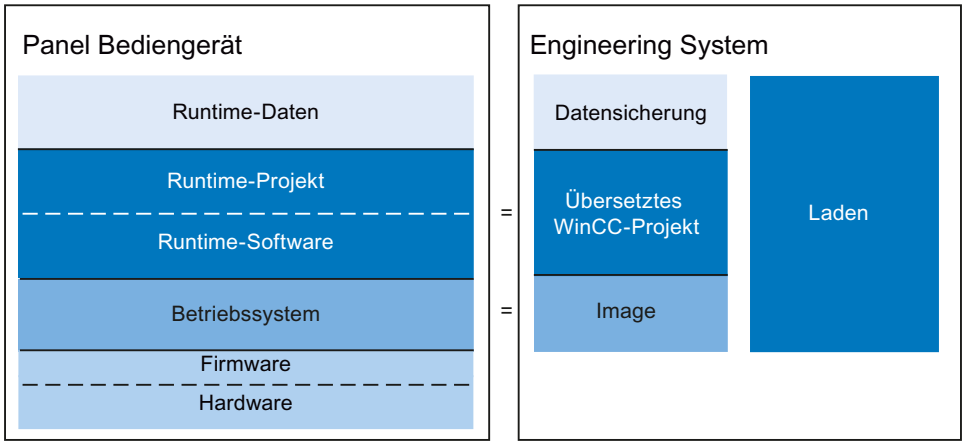

Runtime-Daten

Die Runtime-Daten werden während des laufenden Betriebs der Anlage erzeugt und auf dem Bediengerät gespeichert. Zu ihnen gehören z. B. Rezepturen und Daten zur Benutzerverwaltung. Diese Daten werden beim Laden überschrieben. Sichern Sie bei Bedarf diese Daten vor den Laden eines Runtime-Projekts.

Runtime-Projekt und Runtime-Software

Das Runtime-Projekt enthält die übersetzten Projektierungsdaten für ein Bediengerät. Sie laden das Runtime-Projekt gemeinsam mit der Runtime-Software aus WinCC auf das Bediengerät.

Betriebssystem

Betriebssystem des Bediengeräts wird als Image bezeichnet. Für ein Bediengerät gibt es verschiedene Images. Alle Images eines Bediengeräts sind in WinCC verfügbar. Abhängig von der Projektierung laden Sie bei Bedarf das passende Image gemeinsam mit dem Runtime-Projekt auf das Bediengerät.

Firmware und Hardware

Im Auslieferungszustand des Bediengeräts sind Firmware und Hardware fertig konfiguriert.

15.7.3 Übersicht zur Wartung des Bediengeräts (RT Adv, RT Prof)

Aufbau

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts und ihren Bezug zum Engineering System.

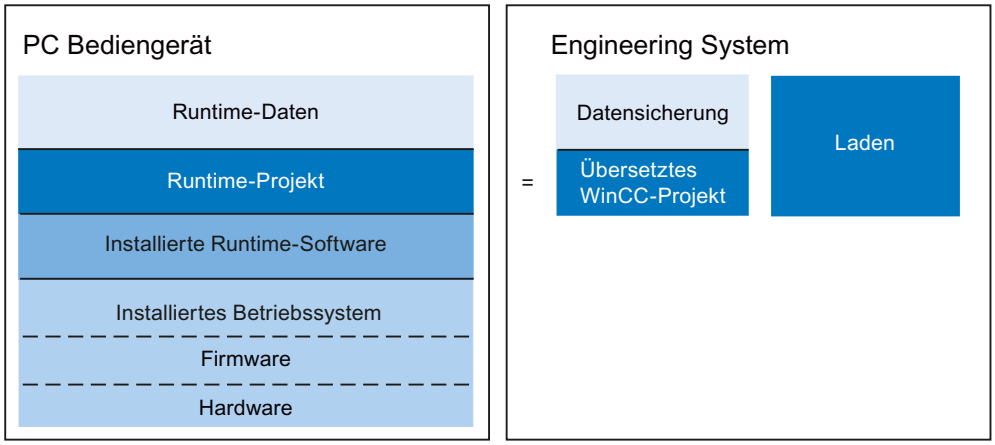

Runtime-Daten

Die Runtime-Daten werden während des laufenden Betriebs der Anlage erzeugt und auf dem Bediengerät gespeichert. Zu ihnen gehören z. B. Rezepturen und Daten zur Benutzerverwaltung. Diese Daten werden beim Laden überschrieben. Sichern Sie bei Bedarf diese Daten vor den Laden eines Runtime-Projekts.

Runtime-Projekt

Das Runtime-Projekt enthält die übersetzten Projektierungsdaten für ein Bediengerät. Sie laden das Runtime-Projekt aus WinCC auf das Bediengerät.

Betriebssystem und Runtime-Software

Betriebssystem und Runtime-Software des Bediengeräts sind auf dem Bediengerät unabhängig von WinCC installiert. Diese Software wird wie bei jedem anderen PC gewartet.

Firmware und Hardware

Im Auslieferungszustand des Bediengeräts sind Firmware und Hardware fertig konfiguriert.

15.7.4 ProSave

Einleitung

Bei der WinCC-Installation wird das Service-Tool "ProSave" standardmäßig mit installiert. Die ProSave-Funktionen rufen Sie in WinCC über das Menü "Online > HMI Bediengeräte Wartung" auf.

Funktionsumfang

ProSave stellt alle Funktionen zur Verfügung, die zum Verwalten von Daten auf dem Bediengerät notwendig sind:

- Datensicherung und Wiederherstellung von gesicherten Daten
- Betriebssystemaktualisierung für Bediengeräte mit Windows CE und kleiner
- Übertragen von License Keys
- Installation und Deinstallation von Treibern und Optionen sowie Auskunft über installierte und installierbare Optionen auf einem Bediengerät
- Kommunikationseinstellungen (werden aus WinCC übernommen)

Siehe auch

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 2426)

15.7.5 Datensicherung des Bediengeräts

Einleitung

Wenn Sie die Daten eines Bediengeräts regelmäßig sichern, verkürzen Sie die Ausfallzeiten z. B. bei einem Gerätetausch auf ein Minimum. Sie übertragen die gesicherten Daten auf das Bediengerät und stellen damit den ursprünglichen Zustand wieder her.

Datensicherung mit WinCC

Wenn ein Bediengerät mit dem Projektierungs-PC verbunden ist, können Sie die Daten des Bediengeräts von dem Projektierungs-PC aus mit WinCC sichern und wiederherstellen.

Umfang der Datensicherung

Welche Daten gesichert und wiederhergestellt werden, ist vom Bediengerättyp abhängig:

- Vollständige Sicherung
  Je nach Bediengerät: Runtime, Firmware, Betriebssystem, Projektierung, Rezepturen, Benutzerverwaltung, Einstelldaten

  Hinweis

  License Keys können nur nur bei folgenden Bediengeräten gespeichert werden:
  - Mobile Panel Wireless
  - Comfort Panel

  Bei folgenden Bediengeräten werden die License Keys nicht gespeichert:
  - alle anderen Windows-CE-Bediengeräte

- Firmware/Projektierung
- Nur Rezepturen
- Nur Rezepturen im CSV-Format
- Nur Benutzerverwaltung

Bei der Datensicherung von einem Bediengerät wird eine komprimierte Datei mit der Endung "*.psb" erstellt.

Grundsätzlich können Sie die Daten auf einem beliebigen Speichermedium sichern. Wenn das Bediengerät in einem Netzwerk integriert ist, können Sie die Daten auch auf einem Server sichern.

Hinweis

Umfang der Datensicherung bei Windows-CE-Bediengeräten

Bei der Datensicherung wird der Inhalt des Flash-Speichers gesichert. Meldearchive und Prozesswertarchive sind grundsätzlich auf dem externen Speichermedium gespeichert. Meldearchive und Prozesswertarchive werden daher nicht gesichert. Sichern Sie Inhalte der Speicherkarte bei Bedarf separat.

Beachten Sie Folgendes bei einer vollständigen Sicherung und Wiederherstellung des Datenbestandes für Windows-CE-Geräte:

- Bei einer vollständigen Sicherung werden alle installierten Optionen gesichert. Grundsätzlich werden alle Daten der Option gesichert, die auch nach "Spannung aus" noch vorhanden sind.
- Bei einer vollständigen Wiederherstellung der Daten werden alle vorher auf dem Gerät befindlichen Daten inklusive License Keys und Betriebssystem unwiderruflich gelöscht.
- Wenn die Wiederherstellung der Daten unterbrochen wurde, führen Sie den Befehl "Auf Werkseinstellungen zurücksetzen" aus. Starten Sie danach die Wiederherstellung der Daten erneut.

Hinweis

Verwenden Sie zum Sichern und Wiederherstellen von Daten Schnittstellen mit hohen Bandbreiten, z. B. USB oder Ethernet.

Hinweis

Bei Windows-CE-Geräten können Sie die Daten auch unabhängig von ProSave und WinCC direkt auf CF-Card oder PC-Card sichern. Weitere Informationen finden Sie in den entsprechenden Betriebsanleitungen.

Siehe auch

Daten des Bediengeräts sichern und wiederherstellen (Seite 2431)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 2426)

15.7.6 Daten des Bediengeräts sichern und wiederherstellen

Hinweis

Verwenden Sie die Wiederherstellung von Projektdaten nur auf Bediengeräten, die mit derselben Projektierungs-Software projektiert wurden.

Voraussetzung

- Das Bediengerät ist mit dem Projektierungs-PC verbunden
- Das Bediengerät ist in der Projektnavigation ausgewählt
- Wenn ein Server zur Datensicherung verwendet wird: Der Projektierungs-PC hat Zugriff auf den Server

Daten des Bediengeräts sichern

Um die Daten des Bediengeräts zu sichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Sichern".
   Der Dialog "Sicherung erstellen" wird geöffnet.
2. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Erstellen".
   Der Dialog "SIMATIC ProSave" wird geöffnet.
3. Wählen Sie unter "Datenart" aus, welche Daten des Bediengeräts gesichert werden.
4. Geben Sie unter "Speichern unter" den Dateinamen der Sicherungsdatei ein.
5. Klicken Sie auf "Start Backup".

Die Datensicherung wird gestartet. Je nach gewählter Verbindung nimmt die Sicherung einige Zeit in Anspruch.

Daten des Bediengeräts wiederherstellen

Um die Daten des Bediengeräts wiederherzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Wiederherstellen". Der Dialog "Sicherung wiederherstellen" wird geöffnet.
2. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Laden". Der Dialog "SIMATIC ProSave" wird geöffnet.
3. Geben Sie unter "Öffnen von …" den Dateinamen der Sicherungsdatei ein. Unter "Datei-Info" werden Informationen zur gewählten Sicherungsdatei angezeigt.
4. Klicken Sie auf "Start Restore".

Die Wiederherstellung wird gestartet. Je nach gewählter Verbindung nimmt dieser Vorgang einige Zeit in Anspruch.

Backup / Restore über Dialog "Backup/Restore" im Start Center des Bediengeräts

Die Funktion über "Backup / Restore" ist für MMC, SD-Speicherkarten sowie USB-Massenspeicher freigegeben. Welche Speichermedien unterstützt werden, ist abhängig vom Bediengerät.

Hinweis

Verbindungen zwischen Basic Panels und Steuerungen

Wenn Sie die Funktion "Backup/Restore" verwenden, sind zeitgleich maximal zwei Verbindungen von Basic Panels zu folgenden Steuerungen möglich:

- SIMATIC S7-1200
- SIMATIC S7-1500

Weiterführende Informationen zu diesem Thema finden Sie im Gerätehandbuch des verwendeten Bediengeräts.

Siehe auch

Datensicherung des Bediengeräts (Seite 2429)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 2426)

15.7.7 Aktualisierung des Betriebssystems

Einleitung

Wenn das Image eines Bediengeräts einen nicht zur Projektierung passenden Versionsstand hat, aktualisieren Sie das Image des Bediengeräts. Die Version des Images entspricht der Geräteversion.

Über die Geräteversion aktualisieren Sie das Betriebssystem und die Runtime-Software des Bediengeräts. Abhängig vom verwendeten Protokoll wird beim Laden des Projekts die Geräteversion auf Nachfrage automatisch aktualisiert.

Danach wird das Laden fortgesetzt. Anderenfalls wird das Laden des Projekts abgebrochen. In diesem Fall starten Sie die Aktualisierung der Geräteversion manuell.

Hinweis

Die Aktualisierung der Geräteversion ist nur möglich bei nicht PC-basierten Bediengeräten.

Aktualisierung der Geräteversion

Um die Geräteversion zu aktualisieren, verbinden Sie das Bediengerät mit dem Projektierungs-PC. Verwenden Sie für diese Verbindung nach Möglichkeit die Schnittstelle mit der höchsten Bandbreite, z. B. Ethernet. Über eine serielle Verbindung kann das Aktualisieren der Geräteversion bis zu einer Stunde dauern.

"Auf Werkseinstellungen zurücksetzen"

Wenn das Betriebssystem auf dem Bediengerät nicht mehr funktionsfähig ist, aktualisieren Sie das Betriebssystem und setzen das Bediengerät auf die Werkseinstellungen zurück.

Hinweis

Auf Werkseinstellungen zurücksetzen über Ethernet für Basic Panels

Um das Bediengerät über Ethernet auf Werkseinstellungen zurückzusetzen, benötigen Sie Folgendes:

- MAC-Adresse des Bediengeräts
- Verfügbare IP-Adresse
- Auf Ethernet TCP/IP eingestellte PG/PC-Schnittstelle des Projektierungs-PC

Sie konfigurieren die PG/PC-Schnittstelle über die Systemsteuerung des Projektierungs-PC. Wählen Sie unter "Zugangspunkt der Applikation" den Eintrag "S7ONLINE (STEP7) -> TCP/IP".

Hinweis

Auf Werkseinstellungen zurücksetzen für Comfort Panels

Um die Comfort Panels auf Werkseinstellungen zurückzusetzen, nutzen Sie bitte ProSave.

Über den Menüeintrag "Online > HMI Bediengeräte Wartung > Zurücksetzen auf Werkseinstellungen" können Sie alle Bediengeräte mit Ausnahme von Comfort Panels auf Werkseinstellungen zurücksetzen.

Siehe auch

Betriebssystem des Bediengeräts aktualisieren (Seite 2434)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 2426)

15.7.8 Betriebssystem des Bediengeräts aktualisieren

Verwenden Sie für diese Verbindung nach Möglichkeit die Schnittstelle mit der höchsten Bandbreite, z. B. Ethernet. Über eine serielle Verbindung kann das Aktualisieren des Betriebssystems bis zu einer Stunde dauern. Wenn Sie das Betriebssystem aktualisieren, aktualisieren Sie damit auch die Runtime-Software auf dem Bediengerät und ändern die Geräteversion.

ACHTUNG
Aktualisieren des Betriebssystems löscht alle Daten auf dem Bediengerät Wenn Sie das Betriebssystem aktualisieren, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb vorher folgende Daten: • Benutzerverwaltung • Rezepturen Zurücksetzen auf Werkseinstellungen löscht darüber hinaus die License Keys. Sichern Sie vor dem Zurücksetzen auf Werkseinstellungen auch die License Keys.

Voraussetzung

- Das Bediengerät ist mit dem Projektierungs-PC verbunden.
- Das Bediengerät ist in der Projektnavigation ausgewählt.

Betriebssystem aktualisieren

Um das Betriebssystem zu aktualisieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Betriebssystem aktualisieren".
   Der Dialog "Betriebssystem aktualisieren" wird geöffnet.
2. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Aktualisieren".
   Der Dialog "SIMATIC ProSave [OS-Update]" wird geöffnet. Der Pfad mit dem Image ist bereits voreingestellt.
3. Wählen Sie bei Bedarf einen anderen Pfad für das Image aus, das Sie auf das Bediengerät übertragen möchten.
4. Klicken Sie auf "Update OS".

Die Aktualisierung wird gestartet. Je nach gewählter Verbindung kann dieser Vorgang einige Zeit in Anspruch nehmen.

Bediengerät auf Werkseinstellungen zurücksetzen

Um das Bediengerät wieder auf Werkseinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Schalten Sie die Stromversorgung für das Bediengerät aus.
2. Verbinden Sie das Bediengerät mit der Engineering-Station.

3. Wählen Sie am Projektierungs-PC in WinCC im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Betriebssystem aktualisieren".
   Der Dialog "Betriebssystem aktualisieren" wird geöffnet.
4. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Aktualisieren".
   Der Dialog "SIMATIC ProSave [OS-Update]" wird geöffnet. Der Pfad mit dem Image ist bereits voreingestellt.
5. Wählen Sie bei Bedarf einen anderen Pfad mit einem anderen Image aus, das Sie auf das Bediengerät übertragen wollen.
6. Aktivieren Sie "Auf Werkseinstellungen zurücksetzen".
7. Klicken Sie auf "Update OS".
8. Um das Zurücksetzen auf Werkseinstellungen zu starten, schalten Sie die Stromversorgung des Bediengeräts wieder ein.
   Dieser Vorgang kann einige Zeit in Anspruch nehmen.

Ergebnis

Das Betriebssystem des Bediengeräts ist funktionsfähig und auf dem aktuellen Stand.

Siehe auch

Aktualisierung des Betriebssystems (Seite 2432)

Übertragung von License Keys (Seite 2437)

Lizenzen verwalten (Seite 2438)

Übersicht zum Laden von Projekten (Seite 2268)

Daten des Bediengeräts sichern und wiederherstellen (Seite 2431)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 2426)

15.7.9 Betriebssystem des Bediengeräts von einem Datenträger aktualisieren

Einleitung

Für Comfort Panels und Mobile Panels können Sie das Betriebssystem mithilfe eines Datenträgers aktualisieren.

Sie finden die Bediengeräte-Image-Dateien z. B. im Installationsverzeichnis von WinCC unter: "\Siemens\Automation\Portal V1x\Data\Hmi\Transfer\<Bediengeräte-Image-Version> \Images".

ACHTUNG
Datenverlust Beim Wiederherstellen werden auf dem Bediengerät vorhandene Daten inklusive Projekt und Bediengeräte-Kennwort gelöscht. Lizenzschlüssel werden erst nach Rückfrage gelöscht. Sichern Sie bei Bedarf Daten vor dem Wiederherstellen.

Voraussetzung

- Die Bediengeräte-Image-Datei befinden sich im Verzeichnis "SIMATIC.HMI\Firmware\" auf Ihrem Datenträger, z. B. eine SIMATIC HMI Memory card oder ein industrietauglicher USB-Stick.
- Der Datenträger mit der jeweiligen Bediengeräte-Image-Datei inklusive Betriebssystem steckt im Bediengerät.

Vorgehensweise

1. Öffnen Sie das "Control Panel" auf Ihrem Bediengerät.
2. Öffnen Sie den Dialog "Service & Commissioning" über das Symbol "Service & Commissioning".
3. Wechseln Sie in das Register "OS Update".
4. Betätigen Sie die Schaltfläche "Next".
   Der Dialog "Update OS image from external memory" wird angezeigt.

   Hinweis

   Wenn sich kein oder ein defektes Speichermedium im Bediengerät befindet, wird die Meldung "0 devices found" angezeigt. Stecken Sie Speichermedium oder tauschen Sie das Speichermedium aus.

5. Betätigen Sie bei Bedarf die Schaltfläche "Refresh".
   Die Gruppe "Accessible devices:" wird aktualisiert. Das Bediengerät prüft das Speichermedium. Im Feld "status information" finden Sie Informationen zu diesem Speichermedium.
6. Wählen Sie in der Gruppe "Accesible devices:" das Speichermedium mit dem erforderlichen Bediengeräte-Image.
7. Betätigen Sie die Schaltfläche "Next". .
   Der Dialog "Update OS image from:\Storage Card USB" wird angezeigt.
8. Wählen Sie in der Gruppe "Firmware files on" die erforderliche Bediengeräte-Image-Datei aus.

9. Wenn Sie Informationen zur gewählten Datei benötigen, betätigen Sie die Schaltfläche "Details".
   Der Dialog "Properties of image file" wird mit folgenden Informationen angezeigt:
   – "Supported": Bediengeräte, die mit dem Bediengeräte-Image kompatibel sind
   – "Image version": Version des Bediengeräte-Image
   – "Image size": Dateigröße der Image-Datei
   – "Creation": Erstellungsdatum der Image-Datei
10. Wenn Sie die gewählte Datei löschen wollen, betätigen Sie die Schaltfläche "Delete".
    Der Dialog "Delete confirmation" wird angezeigt. Wenn Sie die Schaltfläche "OK" betätigen, wird die Datei gelöscht.
11. Wenn Sie die Daten der gewählten Datei wiederherstellen wollen, betätigen Sie Schaltfläche "Update".
    Der Dialog "Update settings" informiert Sie darüber, dass die Einstellungen im Control Panel erhalten bleiben und bietet die Möglichkeit, vorhandene License Keys auf dem Bediengerät zu behalten oder zu löschen.
12. Betätigen Sie die Schaltfläche "Update".
    Der Dialog "Update OS Image" wird angezeigt.
13. Um das Wiederherstellen des Betriebssystems zu starten, betätigen Sie "Yes".
    Der Dialog "Transfer" wird angezeigt. Ein Fortschrittsbalken zeigt den Verlauf des Wiederherstellens an. Danach startet das Bediengerät neu.

Hinweis

Nach dem Wiederherstellen kann eine Neukalibrierung des Touchscreens erforderlich sein.

15.7.10 Übertragung von License Keys

Einleitung

Um einige Optionen von WinCC-Runtime auf einem Bediengerät zu installieren, benötigen Sie für diese Optionen jeweils eine Lizenz. Die benötigten Lizenzen werden als "License Keys" üblicherweise auf einem Datenträger geliefert, z. B. ein USB-Stick. Alternativ können die "License Keys" auch über einen Lizenzserver zur Verfügung gestellt werden.

Um die "License Keys" auf oder von einem Bediengerät zu übertragen, verwenden Sie den "Automation License Manager". Der "Automation License Manager" wird bei der Installation von WinCC automatisch installiert.

ACHTUNG
Sichern von License Keys In folgenden Fällen müssen Sie die "License Keys" sichern, weil die "License Keys" sonst gelöscht werden: • Vor der Aktualisierung des Betriebsystems eines Windows-CE-Bediengeräts • Vor der Wiederherstellung eines kompletten Datenbestandes aus der Sicherung Bei einer Sicherung werden "License Keys", die sich auf einem Bediengerät befinden, abhängig vom Bediengerät gesichert. Weitere Informationen zu diesem Thema finden Sie in der Betriebsanleitung des jeweiligen Bediengeräts.

Siehe auch

Lizenzen verwalten (Seite 2438)

15.7.11 Lizenzen verwalten

Voraussetzung

- Das Bediengerät ist mit dem Projektierungs-PC oder dem PC verbunden, auf dem der "Automation License Manager" installiert ist.
- Wenn Sie den Projektierungs-PC verwenden: Das Bediengerät ist in der Projektnavigation ausgewählt.

Vorgehen

Um die License Keys zu übertragen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den "Automation License Manager". Starten Sie an einem PC ohne WinCC-Installation den "Automation License Manager" über das Windows-Startmenü.
   Der "Automation License Manager" wird gestartet.
2. Wählen Sie im Menü "Bearbeiten > Zielsystem verbinden" den Befehl "Bediengerät verbinden".
   Der Dialog "Zielsystem verbinden" wird geöffnet.
3. Wählen Sie im Bereich "Gerätetyp" entsprechenden Bediengerätetyp aus.
4. Wählen Sie die "Verbindung" aus.

5. Konfigurieren Sie abhängig von der gewählten Verbindung die zugehörigen "Verbindungsparameter".

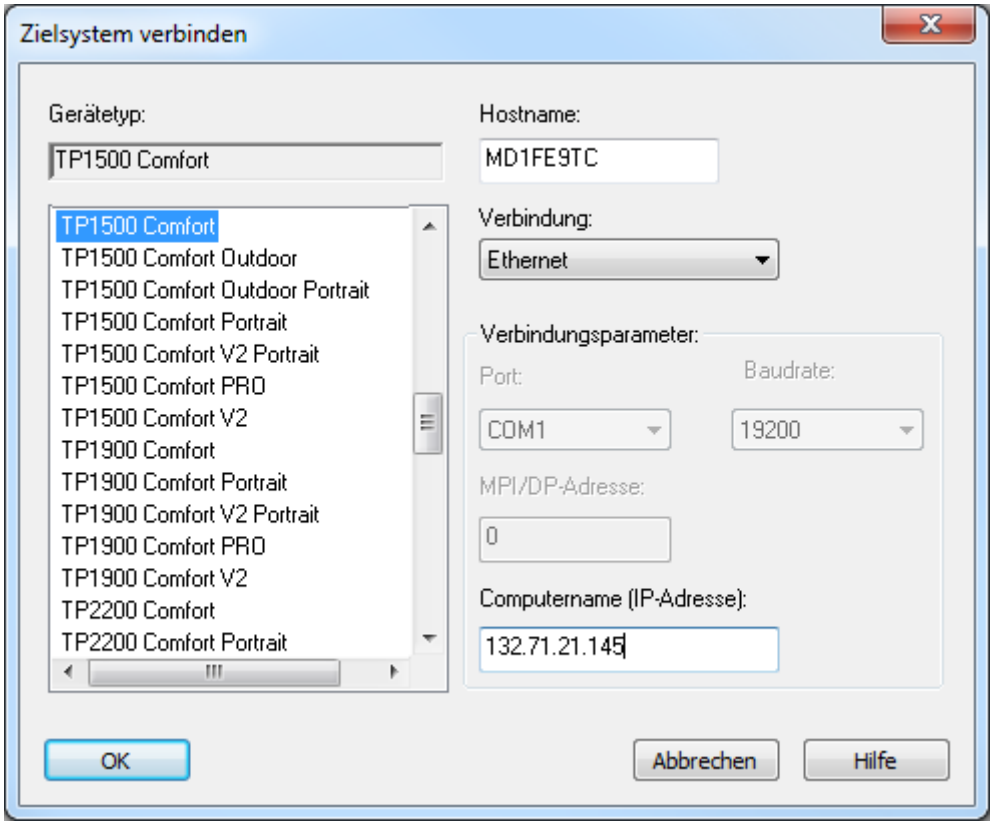

6. Klicken Sie auf "OK".
   Die Verbindung zum Bediengerät wird aufgebaut. Das verbundene Bediengerät wird im linken Bereich des "Automation License Manager" angezeigt.
7. Übertragen Sie die "License Keys" auf das Bediengerät:
   - Selektieren Sie im linken Bereich das Laufwerk, auf dem sich die "License Keys" befinden.
     Die "License Keys" werden im rechten Bereich angezeigt.
   - Selektieren Sie die "License Keys"
   - Verschieben Sie die "License Keys" mit Drag&Drop auf das Bediengerät.

License Keys können Sie vom Bediengerät per Drag&Drop auch wieder entfernen.

Alternative Vorgehensweise

Auf einem PC mit WinCC-Installation können Sie den "Automation License Manager" auch aus WinCC heraus starten: Wählen Sie im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Autorisieren/Lizensieren".

Ergebnis

Die "License Keys" werden auf das Bediengerät übertragen.

Um die "License Keys" vom Bediengerät zu sichern, verschieben Sie die "License Keys" mit Drag&Drop vom Bediengerät auf ein verfügbares Laufwerk.

Siehe auch

Übertragung von License Keys (Seite 2437)

15.7.12 Option installieren und deinstallieren

Einleitung

Auf einem Bediengerät können Sie folgende Optionen installieren:

- Mit WinCC mitgelieferte Zusatzoptionen
- Zusätzlich zu WinCC erworbene Optionen

Welche Optionen Sie installieren können, hängt vom Typ des Bediengeräts ab.

Voraussetzung

- Das Bediengerät ist mit der Engineering-Station oder dem PC mit ProSave verbunden
- Das Bediengerät ist in der Projektnavigation ausgewählt.

Vorgehen

Um eine Option auf dem Bediengerät zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > HMI Bediengeräte Wartung " den Befehl "Optionen".
   Der Dialog "Optionen laden" wird geöffnet.
2. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Laden".
   Der Dialog "SIMATIC ProSave" wird geöffnet.
   Alle verfügbaren Optionen und die bereits installierten Optionen werden angezeigt.
3. Um sich die auf dem Bediengerät installierten Optionen anzeigen zu lassen, klicken Sie auf "Gerätestatus".
4. Um eine Option auf dem Bediengerät zu installieren, selektieren Sie die Option unter "Verfügbare Optionen" und klicken Sie auf ">>".
5. Um eine Option vom Bediengerät zu deinstallieren, selektieren Sie die Option unter "Installierte Optionen" und klicken Sie auf "<<".

Ergebnis

Die ausgewählten Optionen wurden auf dem Bediengerät installiert oder von diesem deinstalliert.

15.7.13 Asiatische Schriftarten auf das Bediengerät laden

Einleitung

Bei Verwendung von asiatischen Sprachen werden nur die asiatischen Zeichen, die in der Projektierung des Bediengerätes verwendet werden, beim Transfer auf das Bediengerät geladen. Wenn Texte wie bei Steuerungsmeldungen oder bei ProDiag direkt von der Steuerung an das Bediengerät gesendet werden, kann dies zu Darstellungsproblemen führen. In diesen Fällen können Sie die komplette asiatische Schriftart auf das Bediengerät laden.

Vorgehen

1. Selektieren Sie in der Projektnavigation das Bediengerät.
2. Wählen Sie im Menü "Online > HMI Bediengeräte Wartung" den Befehl "Optionen". Der Dialog "Optionen laden" wird geöffnet.
3. Wählen Sie den Typ der PG/PC-Schnittstelle und das Zielgerät und klicken Sie auf "Laden". Der Dialog "SIMATIC ProSave" wird geöffnet.
4. Selektieren Sie die jeweilige Schriftart unter "Verfügbare Optionen" und klicken Sie auf ">>". Die Schriftart wird auf das Bediengerät geladen.

Siehe auch

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 989)

Vollständige Meldung von der Steuerung an das Bediengerät senden und automatisch aktualisieren (Seite 987)

Zeichen für HMI-Variablen vom Typ WString/WChar bereitstellen (Seite 574)

15.8 Druckertreiber

15.8.1 Gültigkeit

Gültigkeit

Dieses Dokument beschreibt, welche Erweiterungen mit dem Optionspaket Druckertreiber ermöglicht werden.

Um die erweiterten Funktionalität zu nutzen, installieren Sie das Optionspaket vollständig. Durch die Installation werden neue Druckertreiber für einige Bediengeräte bereitgestellt.

Sie können einzelne Druckertreiber mit dem Tool "SIMATIC ProSave" je nach Bedarf auf Ihrem Bediengerät installieren. Welche Druckertreiber für welche Bediengeräte zur Verfügung stehen, entnehmen Sie dem Kapitel "Unterstützte Bediengeräte".

Installation

Um das "Optionspaket Druckertreiber V1.4" zu installieren, befolgen Sie bitte die Anweisungen im Kapitel "Installation".

Applikationsbeispiel

Ein Applikationsbeispiel zum Thema "Drucken mit HMI Panels" finden Sie im Internet unter

Applikationsbeispiel "Drucken mit SIMATIC Comfort Panels" (https://support.industry.siemens.com/cs/ww/de/view/58205602)

Siehe auch

Applikationsbeispiel "Drucken mit SIMATIC Comfort Panels" (https://support.industry.siemens.com/cs/ww/de/view/58205602)

15.8.2 Unterstützte Bediengeräte

Inhalte des Optionspakets

Das "Optionspaket Druckertreiber V1.4" beinhaltet folgenden Druckertreiber für die angegebenen Bediengeräte.

Der Druckertreiber unterstützt alle Druckoptionen der Bediengeräte.

Bediengerät	Brother P-Touch QL 650 TD [1]	Brightek WH-AB, WH-C1, WH-E19	Postscript
KP400 Comfort KTP400 Comfort	USB	USB	Ja
KP700 Comfort TP700 Comfort	USB	USB	Ja
KP900 Comfort TP900 Comfort	USB	USB	Ja
KP1200 Comfort TP1200 Comfort	USB	USB	Ja
KP1500 Comfort TP1500 Comfort	USB	USB	Ja
TP1900 Comfort	USB	USB	Ja
TP2200 Comfort	USB	USB	Ja
USB: Drucker über die USB Schnittstelle direkt mit dem Bediengerät verbinden [1] Es wird nur Protokolldruck unterstützt.			

Hinweis

Bitte beachten Sie, dass Postscript ein bekanntes und offenes Kommunikationsprotokoll darstellt und Ausdrucke deswegen durch Dritte theoretisch verändert werden können. Es wird empfohlen, den Drucker in besonders sicherheitskritischen Anlagen per USB anzuschließen.

15.8.3 Druckertreiber einrichten

15.8.3.1 Postscript

PostScript V1.4 als Drucker am Bediengerät festlegen

1. Öffnen Sie das Control Panel.
2. Starten Sie die Applikation "Printer".
3. Wählen Sie im Dialog "Printer Properties" unter "Printer Language:" den Eintrag "PostScript V1.4" aus.

Hinweis

Nicht alle Drucker unterstützen PostScript. Vergewissern Sie sich, dass Ihr Drucker diese Druckersprache unterstützt.

Aktivieren Sie die Option "Draft-Mode", um die Druckgeschwindigkeit zu erhöhen.

15.8.3.2 HTML

Einleitung

Nach der Installation des Druckertreibers "HMTL V1.3" ist auf dem Desktop des Bediengeräts die Applikation "HTMLSettings.exe" verfügbar.

Ablageort wählen

1. Öffnen Sie die Applikation "HTMLSettings.exe" auf dem Desktop Ihres Bediengeräts.
2. Öffnen Sie in der Registerkarte "Storage Location".
3. Wählen Sie für den Ablagerort unter "Primary Location" z. B. "USB-Stick" aus.
4. Wählen Sie das Verzeichnis, in dem die Datei nach dem Drucken abgelegt wird.

Dateinamen wählen

1. Öffnen Sie die Applikation "HTMLSettings.exe"
2. Öffnen Sie die Registerkarte "HTML File Names"
3. Wählen Sie die Einstellungen für die Namen der HTML-Datei beim Ausdruck.
4. Bestätigen Sie die vorgenommenen Einstellungen mit "Apply".
5. Um die Standardeinstellungen für die Vergabe der Dateinamen beizubehalten, wählen Sie "Default".

Spalten für den Druck von Protokollen wählen

1. Öffnen Sie die Applikation "HTMLSettings.exe"
2. Öffnen Sie die Registerkarte "Columns Attributes".
3. Wählen Sie ein Attribut in der Kategorie "Available Columns":
4. Mit der Schaltfläche ">>" nehmen Sie das Attribut in die Auswahl mit auf.
5. Bei Bedarf legen Sie für das gewählte Attribut fest, ob es als Image oder Hyperlink in der HTML-Datei angezeigt wird.
6. Um Attribute aus der Kategorie "Selected Columns" abzuwählen, selektieren Sie das Attribut und klicken Sie auf die Schaltfläche "<<".

Stylesheets wählen

1. Öffnen Sie die Applikation "HTMLSettings.exe"
2. Öffnen Sie die Registerkarte "Advanced Options".
3. Wählen Sie unter "Enter Style Sheet File Path" den Ablageort der Stylesheets, die Sie anwenden wollen.
4. Wählen Sie die Datei, die angewendet wird.

5. Wählen Sie das Verzeichnis für Links.
6. Wählen Sie das Verzeichnis für Bilder.

HTML V1.3 als Drucker am Bediengerät festlegen

1. Öffnen Sie das Control Panel.
2. Starten Sie die Applikation "Printer".
3. Wählen Sie im Dialog "Printer Properties" unter "Printer Language:" den Eintrag "HTML V1.3" aus.
4. Um HMTL-Datei schnell zu speichern, aktivieren Sie "Draft Mode".

15.8.3.3 PDF

Einleitung

Nach der Installation des Druckertreibers "PDF V1.3" ist auf dem Desktop des Bediengeräts die Applikation "PDFSettings.exe" verfügbar.

Ablageort wählen

1. Starten Sie die Applikation "PDFSettings.exe" auf dem Desktop ihres Bediengeräts.
2. Öffnen Sie in der Registerkarte "Storage Location".
3. Wählen Sie für den Ablageort unter "Primary Location" z. B. "USB-Stick" aus.
4. Wählen Sie das Verzeichnis, in dem die Datei nach dem Drucken abgelegt wird.

Dateinamen wählen

1. Öffnen Sie die Applikation "PDFSettings.exe"
2. Öffnen Sie die Registerkarte "PDF File Names"
3. Wählen Sie die Einstellungen für die Namen der PDF-Datei beim Ausdruck.
4. Bestätigen Sie die vorgenommenen Einstellungen mit "Apply".
5. Um die Standardeinstellungen für die Vergabe der Dateinamen beizubehalten, wählen Sie "Default".

PDF V1.3 als Drucker am Bediengerät festlegen

1. Öffnen Sie das Control Panel.
2. Starten Sie die Applikation "Printer".

3. Wählen Sie im Dialog "Printer Properties" unter "Printer Language:" den Eintrag "PDF V1.3" aus.
4. Um PDF-Datei schnell zu speichern, aktivieren Sie "Draft Mode".

Regel

Die folgenden Tabelle zeigt die gerätebedingten Unterschiede beim Drucken von PDFs:

Mode	KP700 Comfort TP700 Comfort KP900 Comfort TP900 Comfort KP1200 Comfort TP1200 Comfort	KP1500 Comfort TP1500 Comfort TP1900 Comfort TP2200 Comfort
Normal Mode	10 Sekunden pro Seite	2 Sekunden pro Seite
Draft Mode	2-3 Sekunden pro Seite	1 Sekunde pro Seite

15.8.4 Installation

Voraussetzungen

Auf dem PC ist ProSave installiert.

Vorgehensweise

1. Doppelklicken Sie auf die Datei "PROSAVE-OPT.exe".
2. Wählen Sie im Dialog "WinZip Self-Extractor" wählen "<Setup>" aus.
3. Wählen Sie die Sprache für das Installationsmenü aus.
4. Wählen Sie im Dialog "Optionspaket Druckertreiber Setup" "<Weiter>" aus.
5. Bestätigen Sie die nächste Abfrage mit "<Installieren>".
6. Bestätigen Sie die Meldung "Optionspaket Druckertreiber Setup" mit "<Fertigstellen>".

Hinweis

Um den Druckertreiber auf ein Bediengerät zu übertragen, befolgen Sie bitte die Anweisungen im Kapitel "Transfer der Optionen".

15.8.5 Transfer der Optionen

Voraussetzungen

Das Bediengerät ist an einem PC angeschlossen, auf dem ProSave installiert ist.

Vorgehensweise

1. Starten Sie ProSave auf Ihrem PC.
2. Wählen Sie im Register "Allgemein" den entsprechenden Bediengerätetyp.
3. Wählen Sie die Verbindungsart zwischen Bediengerät und dem PC.
4. Stellen Sie die Parameter für die Verbindung ein.
5. Wählen Sie das Register "Optionen" an.
6. Wählen Sie unter "Verfügbare Optionen" den gewünschten Druckertreiber aus.
7. Schalten Sie das Bediengerät in die Betriebsart "Transfer".
   Wenn Sie für das Bediengerät den automatischen Transfer aktiviert haben, wechselt das Bediengerät beim Starten der Übertragung automatisch in die Betriebsart "Transfer".
8. Starten Sie am PC den Transfer mit der Schaltfläche ">>".
9. Folgen Sie den Anweisungen von ProSave und den Anweisungen, die auf dem Bediengerät erscheinen.
   Während der Installation des Druckertreibers erscheint eine Statusanzeige, die den Fortschritt des Vorgangs anzeigt.

Ergebnis

Wenn die Installation abgeschlossen ist, wird der neue Druckertreiber in ProSave im Bereich "Installierte Optionen" angezeigt. Der neue Druckertreiber kann nun auf dem Bediengerät verwendet werden.

Leistungsmerkmale 16

16.1 Allgemeine technische Daten

16.1.1 Verwendbare Zeichen

Einleitung

Die folgende Tabelle gibt die Beschränkungen an, die bei der Vergabe von Namen und Kennwörter zu beachten sind.

Verwendbare Zeichen

Namen / Kennwörter	Beschränkung
Namen von Computern	Der Name muss mit einem Buchstaben beginnen.
Namen von Objekten	Verwenden Sie nicht die Zeichen ? " / \ * < > %
Namen und Kennwörter in Benutzeranzeige	Verwenden Sie nicht die Zeichen ? " / \ § & $ %
Rezepturdaten	Verwenden Sie nicht die Zeichen , ;
Namen von Meldearchiven	Verwenden Sie in den Ablageorten Datei - RDB, Datei - CSV (ASCII) und Datei - TXT (Unicode) nicht die Zeichen \ / * ? : " < > |
	Verwenden Sie beim Ablageort Datenbank nur die Zeichen a-z A-Z 0-9 _ @ # $ Die Zeichen _ @ # $ dürfen nicht als erstes Zeichen des Namens verwendet werden
Namen von HMI-Variablen	Die Namen von HMI-Variablen dürfen nicht mit dem Zeichen @ beginnen
Namen von Bildern	Verwenden Sie nicht die Sonderzeichen ? " / \ * < > Vermeiden Sie in Bildern für Runtime Professional der Zeichen . :
Variablennamen in Bildbausteinen	Verwenden Sie nicht die Zeichen . @
Eigenschaftsnamen der Bildbausteine	Verwenden Sie in den Namen keine UNICODE Zeichen (z. B. chinesische Schriftzeichen). Der Name muss mit einem Buchstaben beginnen. Der Name darf maximal 255 Zeichen enthalten.
Namen von AuditTrails	Verwenden Sie keine Sonderzeichen
Namen von benutzerdefinierten Anwenderdatentypen als Bibliothekobjekten (RT Professional)	Verwenden Sie nicht die Sonderzeichen : ? " \\ \ * $ % " ' Der Name darf maximal 128 Zeichen enthalten.

Namen / Kennwörter	Beschränkung
Namen von C-Skripten und VB-Skripten als Bibliotheksobjekten	Der Name muss mit einem ASCII-Buchstaben anfangen, gefolgt von einem alphanumerischen ASCII-Zeichen oder Unterstrich. Der Name darf maximal 128 Zeichen enthalten.
Namen von Verbindungen	Verwenden Sie keine Leerzeichen und Sonderzeichen
Namen von VB-Funktionen	Verwenden Sie keine Leerzeichen, Sonderzeichen oder VB-Schlüsselwörter
Namen von Windows-Benutzergruppen und Windows-Benutzer bei Verwendung von SIMATIC Logon	Verwenden Sie nicht die Sonderzeichen / \
Kommunikation / OPC: verwendete Namen	Verwenden Sie weder Leerzeichen noch die Zeichen . : ? " ' \ * %

Hinweis

Geräteabhängigkeit - Anführungszeichen in Projektnamen

WinCC Professional Projekte, deren Namen Anführungszeichen enthalten, werden nicht übersetzt.

Wenn Sie ein Bediengerät für WinCC RT Professional anlegen, verwenden Sie keine Anführungszeichen innerhalb des Projektnamens.

16.1.2 Empfohlene Drucker

Empfohlene Drucker

Die aktuelle Liste der empfohlenen Drucker für die Bediengeräte finden Sie im Internet:

Link zur aktuellen Druckerliste (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=11376409&caller=view)

Hinweis

Alle Bediengeräte außer PC und Panel-PC unterstützen maximal einen Drucker über USB-Anschluss, auch wenn physikalisch mehrere Anschlüsse vorhanden sind.

Siehe auch

Druckerliste (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=11376409&caller=view)

16.1.3 Drucken über Druckserver

Einleitung

Ein Druckserver ermöglicht Ihnen den Zugriff auf Drucker in einem Netzwerk. Druckaufträge werden über den Druckserver an den entsprechenden Drucker weitergeleitet.

Hinweis

Das Drucken über Druckserver ist für folgende Bediengeräte möglich:

- Mobile Panel 177
- Mobile Panel 277, Mobile Panel 277 IWLAN, Mobile Panel 277F IWLAN
- Mobile Panel 277 IWLAN V2, Mobile Panel 277F IWLAN V2, Mobile Panel 277F IWLAN (RFID Tag)
- KTP Mobile Panels (ab Geräteversion V13)
- Comfort Panels
- PC mit WinCC Runtime Advanced

Voraussetzungen

- Der Druckserver nutzt den Modus "RAW".

  Hinweis

  Weitere Informationen zu den Einstellungen entnehmen Sie der Dokumentation des entsprechenden Druckservers.

- Das Gerät ist über Ethernet mit dem Druckserver bzw. Drucker mit integriertem Druckserver verbunden.
- Bei externen Druckservern ist der Druckserver über USB mit einem Drucker verbunden.
- In den "Printer Properties" des Bediengeräts werden die gleiche IP-Adresse und das gleiche Port verwendet, wie am Druckserver.

Vorgehensweise

1. Öffnen Sie das "Control Panel" auf Ihrem Bediengerät.
2. Wählen Sie "Printer". Der Dialog "Printer Properties" wird geöffnet.

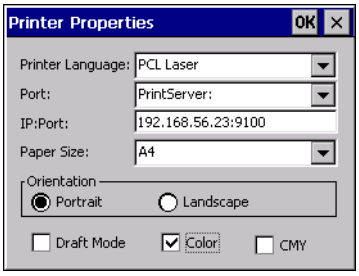

3. Wählen Sie im Auswahlfeld "Printer Language" den Drucker aus.

4. Wählen Sie im Auswahlfeld "Port" den Eintrag "PrintServer".
5. Geben Sie unter "IP:Port" die IP-Adresse und das Port für die Kommunikation mit dem Druckserver ein.

   Hinweis

   Verwenden Sie ":" als Trennzeichen zwischen IP-Adresse und Port z.B. "192.168.56.23:**9100**".

6. Wählen Sie die weiteren Druckeinstellungen.

Hinweis

Sie können auch über das Netzwerk drucken.

Dafür wählen Sie im Auswahlfeld "Port" den Eintrag "Network:".

Im Auswahlfeld "Network:" wählen Sie den Netzwerkpfad, der sich aus dem Computernamen und dem Druckernamen zusammensetzt, z. B. "\\server12\\printer01".

16.1.4 Speicherplatzbedarf von Rezepturen

Einleitung

Die nachfolgende Berechnung des Speicherbedarfs von Rezepturen ist nur für Windows CE-Geräte gültig.

Speicherbedarf berechnen

Der erforderliche Speicherbedarf pro Rezeptur (in kByte) errechnet sich aus den drei Summanden D1 + D2 + D3.

Es gilt:

- D1 = (Anzahl Einträge x 5 + M + 8):1024
  Für M gilt:
  M = Gesamtlänge aller Variablennamen = Summe der Zeichen aller Variablennamen (UTF8-codiert, max. 255 Byte pro Variablenname), die in den Einträgen verwendet werden.
- D2 = [(Anzahl Datensätze x 12) + 4]:1024
- D3 = [Anzahl Datensätze x (Datensatzlänge + N) + 4]:1024
  Für N gilt:

Die Summe der Längen der Datensatznamen des jeweiligen Datensatzes in allen Sprachen (max. 255 Bytes pro Sprache) + Overhead pro Datensatz (1 Byte + Anzahl der Sprachen * 3 Byte).

D1, D2 und D3 sind auf die nächste ganze Zahl aufzurunden.

Speicherbedarf beim Verwenden von Arrays

Der pro Rezeptur erforderliche Speicherbedarf (in kByte) errechnet sich aus den drei Summanden D1 + D2 + D3.

Es gilt:

- D1 = (Anzahl Einträge x 5 + M + 8):1024
  Jedes Element der verwendeten Variablenarrays zählt als ein einzelner Eintrag.
  Für M gilt:
  M = (Länge des Namens der Arrayvariablen + K) x Anzahl der Arrayelemente
  Für K gilt:
  K = 3: 2 bis 9 Elemente im Array
  K = 4: 10 bis 99 Elemente im Array
  K = 5: 100 bis 999 Elemente im Array
  K = 6: 1000 bis 9999 Elemente im Array
  K = 7: 10000 bis 12000 Elemente im Array
- D2 = [(Anzahl Datensätze x 12) + 4] : 1024
- D3 = [Anzahl Datensätze x (Datensatzlänge + N) + 4] : 1024
  Für N gilt:

Die Summe der Längen der Datensatznamen des jeweiligen Datensatzes in allen Sprachen (max. 255 Bytes pro Sprache) + Overhead pro Datensatz (1 Byte + Anzahl der Sprachen * 3 Byte).

D1, D2 und D3 sind auf die nächste ganze Zahl aufzurunden.

Hinweis

Wenn Sie in einer Rezeptur Variablen und Arrays gemeinsam verwenden, dann müssen Sie die Ergebnisse beider Formeln addieren, um den Gesamt-Speicherbedarf zu errechnen.

16.1.5 Speicherplatzbedarf von Rezepturen für Basic Panels

Einleitung

Die nachfolgende Berechnung des Speicherbedarfs von Rezepturen ist für Basic Panels gültig.

Einschränkungen

Der im Bediengerät vorhandene Speicherplatz für Rezepturen beträgt 39 Kilobyte. Dieser Speicherplatz darf nicht überschritten werden. Der Gesamtspeicherplatz für Rezepturen wird folgendermaßen berechnet: Summe aller Rezepturen + Rezeptur mit größtem Speicherbedarf.

Jede einzelne Rezeptur darf eine maximale Speichergröße von 19 Kilobyte nicht überschreiten.

Speicherbedarf berechnen

Der erforderliche Speicherbedarf pro Rezeptur in Kilobyte errechnet sich aus den drei Summanden D1 + D2 + D3.

Es gilt:

- D1 = Anzahl Datensätze x M
  Für M (Größe eines Datensatzes) gilt:
  M = 1 x Anzahl der Elemente eines Bytes + 2 x Anzahl der Elemente von 2 Bytes + 4 x Anzahl der Elemente von 4 Bytes + 8 x Anzahl der Elemente von 8 Bytes + K
  Für K (Größe der String-Elemente) gilt:
  K = Anzahl der String-Elemente x (Größe des Strings + 1) x 2

- D2 - Datensatzgröße
  D2 = 4 + Anzahl der Sprachen x 8 + Anzahl der Sprachen x (4 + 4 x Anzahl der Datensätze + (Länge des Datensatznamens + 1) x 2 x Anzahl der Datensätze) + 8 + 8 x Anzahl der Datensätze
  Oder umformuliert:
  D2 = 12 + 8 x Anzahl der Datensätze + Anzahl der Sprachen x (12 + Anzahl der Datensätze x (4 + (Länge des Datensatznamens +1) x 2))

- D3 Gemeinsam genutzter Speicher
  D3 = 14 + Anzahl der Elemente

Hinweis

Die Berechnung für Arrays und Einzelelemente kann wie oben beschrieben durchgeführt werden.

16.2 Leistungsmerkmale der Geräte

16.2.1 Basic Panel

Basic Panel

Die folgende Tabelle unterstützt Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Leistungsmerkmale für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Variablen

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Variablen im Projekt	250	500	250 (mono) 500 (color)	500	500	500
Anzahl PowerTags	--	--	--	--	--	--
Anzahl Elemente je Array	100	100	100	100	100	100
Anzahl Lokale Variablen	--	--	--	--	--	--
Anzahl Strukturen	--	--	--	--	--	--
Anzahl Strukturelemente	--	--	--	--	--	--

Meldungen

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Meldeklassen	32	32	32	32	32	32
Anzahl Bitmeldungen	200	200	200	200	200	200
Anzahl Analogmeldungen	15	15	15	15	15	15
Länge einer Meldung in Zeichen	80	80	80	80	80	80
Anzahl Prozesswerte je Meldung	8	8	8	8	8	8
Größe des Meldepuffers	256	256	256	256	256	256
Anzahl anstehende Meldeereignisse	64	64	64	64	64	64

Bilder

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Bilder	50	50	50	50	50	50
Anzahl Felder pro Bild	30	30	30	30	30	30
Anzahl Variablen pro Bild	30	30	30	30	30	30
Anzahl Komplexe Objekte pro Bild [1)]	5	5	5	5	5	5
Anzahl Array-Elemente pro Bild [2)]	100	100	100	100	100	100

1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.

2) In Rezepturen enthaltene Array-Elemente zählen dabei mit.

Rezepturen

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Rezepturen	5	5	5	5	5	5
Anzahl Elemente pro Rezeptur [1)]	20	20	20	20	20	20
Nutzdatenlänge in Byte pro Datensatz	--	--	--	--	--	--
Anzahl Datensätze pro Rezeptur	20	20	20	20	20	20
Reservierter Speicher für Datensätze im internen Flash	40 kB	40 kB	40 kB	40 kB	40 kB	40 kB

1) Bei Verwendung von Arrays, zählt jedes Arrayelement als ein Rezepturelement

Archive

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Archive	--	--	--	--	--	--
Anzahl Einträge je Archiv (incl. aller Archivsegmente) [1)]	--	--	--	--	--	--
Anzahl Archivsegmente	--	--	--	--	--	--
zyklischer Trigger für die Archivierung von Variablen	--	--	--	--	--	--
Anzahl archivierbarer Variablen je Archiv	--	--	--	--	--	--

1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

Kurven

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Kurven	25	25	25	25	25	25

Textlisten und Grafiklisten

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Grafiklisten	100	100	100	100	100	100
Anzahl Textlisten	150	150	150	150	150	150
Anzahl der Einträge pro Text- oder Grafikliste	30	30	30	30	30	30
Anzahl Grafikobjekte	500	500	500	500	500	500
Anzahl Textelemente	500	500	500	500	500	500

Skripte

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Skripte	--	--	--	--	--	--

Kommunikation

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Verbindungen	4	4	4	4	4	4
Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP"	--	--	--	--	--	--

Hilfesystem

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Länge eines Hilfetextes in Zeichen	500	500	500	500	500	500

Sprachen

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Runtimesprachen	5	5	5	5	5	5

Aufgabenplaner

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Zeitgetriggerte Aufgaben [1)]	--	--	--	--	--	--

1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

Benutzerverwaltung

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Anzahl Benutzergruppen	50	50	50	50	50	50
Anzahl Berechtigungen	32	32	32	32	32	32
Anzahl Benutzer	50	50	50	50	50	50

Projekt

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Größe der Projektdatei "*.srt"	1024 kB	1024 kB	1024 kB	1024 kB	1024 kB	1024 kB

Siehe auch

S7-1200 Handbuch (http://support.automation.siemens.com/WW/view/en/36932465/0/de)

16.2.2 Basic Panel 2nd Generation

Basic Panel 2nd Generation

Die folgende Tabelle unterstützt Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Leistungsmerkmale für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Variablen

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Variablen im Projekt	800	800	800	800
Anzahl PowerTags	--	--	--	--
Anzahl Elemente je Array	100	100	100	100
Anzahl Lokale Variablen	--	--	--	--
Anzahl Strukturen	--	--	--	--
Anzahl Strukturelemente	--	--	--	--

Meldungen

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Meldeklassen	32	32	32	32
Anzahl Bitmeldungen	1000	1000	1000	1000
Anzahl Analogmeldungen	25	25	25	25
Länge einer Meldung in Zeichen	80	80	80	80
Anzahl Prozesswerte je Meldung	8	8	8	8
Größe des Meldepuffers	256	256	256	256
Anzahl anstehende Meldeereignisse	64	64	64	64

Bilder

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Bilder	250	250	250	250
Anzahl Felder pro Bild	100	100	100	100
Anzahl Variablen pro Bild	100	100	100	100
Anzahl Komplexe Objekte pro Bild [1)]	Diese Angabe ist für Basic Panels 2nd Generation nicht relevant			
Anzahl Rezepturanzeigen pro Bild	10	10	10	10
Anzahl Kurvenanzeigen pro Bild	8	8	8	8
Anzahl Meldeanzeigen pro Bild	20	20	20	20
Anzahl Benutzeranzeigen pro Bild	1	1	1	1
Anzahl Systemdiagnose-Anzeigen pro Bild	5	5	5	5
Anzahl Systemfunktionen pro Bild	150	150	150	150
Anzahl Multiplex-Variablen pro Bild	100	100	100	100

1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.

Rezepturen

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Rezepturen	50	50	50	50
Anzahl Elemente pro Rezeptur [1)]	100	100	100	100
Nutzdatenlänge in kByte pro Datensatz	32	32	32	32
Anzahl Datensätze pro Rezeptur	100	100	100	100
Reservierter Speicher für Datensätze im internen Flash	256 kByte	256 kByte	256 kByte	256 kByte

1) Bei Verwendung von Arrays, zählt jedes Arrayelement als ein Rezepturelement

Archive

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Variablenarchive	1	1	1	1
Anzahl Meldearchive	1	1	1	1
Anzahl Einträge je Archiv (incl. aller Archivsegmente) [1)]	10000	10000	10000	10000
Anzahl Archivsegmente	400	400	400	400
Anzahl archivierbarer Variablen je Archiv	10	10	10	10
Zyklischer Trigger für die Archivierung von Variablen	1 s	1 s	1 s	1 s

1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

Kurven

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Kurven	25	25	25	25

Textlisten und Grafiklisten

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Grafiklisten	100	100	100	100
Anzahl Textlisten	300	300	300	300
Anzahl der Einträge pro Text- oder Grafikliste	100	100	100	100
Anzahl Grafikobjekte	1000	1000	1000	1000
Anzahl Textelemente	2500	2500	2500	2500

Skripte

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Skripte	--	--	--	--

Kommunikation

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Verbindungen	4	4	4	4
Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP"	--	--	--	--

Hilfesystem

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Länge eines Hilfetextes in Zeichen	500	500	500	500

Sprachen

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Runtimesprachen	10	10	10	10

Aufgabenplaner

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Zeitgetriggerte Aufgaben [1)]	--	--	--	--

1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

Benutzerverwaltung

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Anzahl Benutzergruppen	50	50	50	50
Anzahl Berechtigungen	32	32	32	32
Anzahl Benutzer	50	50	50	50

Projekt

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Größe der Projektdatei "*.srt"	10 MB	10 MB	10 MB	10 MB

16.2.3 Mobile Panel

Mobile Panel

Die folgenden Tabellen über die Systemgrenzen unterstützen Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Systemgrenzen für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Variablen

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile TP1000F Mobile
Anzahl Variablen im Projekt	1000	2048	2048	2048	2048	2048	2048	2048
Anzahl PowerTags	--	--	--	--	--	--	--	--
Anzahl Elemente je Array	1000	1000	1000	1000	1000	1000	1000	1000
Anzahl Lokale Variablen	500	1000	1000	1000	1000	1000	1000	1000
Anzahl Strukturen	--	999	999	999	999	999	999	999
Anzahl Strukturelemente	--	400	400	400	400	400	400	400

Meldungen

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F MobileKTP700 F Mobile
Anzahl Meldeklassen	32	32	32	32	32	32	32	32
Anzahl Bitmeldungen	2000	4000	4000	4000	4000	4000	4000	4000
Anzahl Analogmeldungen	50	200	200	200	200	200	200	200
Länge einer Meldung in Zeichen	80	80	80	80	80	80	80	80
Anzahl Prozesswerte je Meldung	8	8	8	8	8	8	8	8

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F MobileKTP700 F Mobile
Größe des Meldepuffers	256	512	512	512	512	1024	1024	1024
Anzahl anstehende Meldeereignisse	64	250	250	250	250	500	500	500

Bilder

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F MobileKTP700 F Mobile
Anzahl Bilder	500	500	500	500	500	500	500	500
Anzahl Felder pro Bild	50	200	200	200	200	50	400	400
Anzahl Variablen pro Bild	50	200	200	200	200	50	400	400
Anzahl Komplexe Objekte pro Bild [1)]	5	10	10	10	10	5	20	20

1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.

Rezepturen

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F MobileKTP700 F Mobile
Anzahl Rezepturen	100	300	300	300	300	300	300	300
Anzahl Elemente pro Rezeptur [1)]	200	1000	1000	1000	1000	1000	1000	1000
Nutzdatenlänge in Byte pro Datensatz	800	4000	4000	4000	4000	262144	262144	262144
Anzahl Datensätze pro Rezeptur	200	500	500	500	500	500	500	500
Reservierter Speicher für Datensätze im internen Flash	32 kB	64 kB	64 kB	64 kB	64 kB	2 MB	2 MB	2 MB

1) Bei Verwendung von Arrays zählt jedes Arrayelement als ein Rezepturelement

Archive

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F MobileKTP700 F MobileKTP700 F Mobile
Anzahl Archive	--	20	20	20	20	50	50	50
Anzahl Einträge je Archiv (incl. aller Archivsegmente) 1)	--	10000	10000	10000	10000	20000	20000	20000
Anzahl Archivsegmente	--	400	400	400	400	400	400	400
zyklischer Trigger für die Archivierung von Variablen	--	1 s	1 s	1 s	1 s	1 s	1 s	1 s
Anzahl archivierbarer Variablen je Archiv	--	2048	2048	2048	2048	2048	2048	2048

1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

Kurven

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Kurven	50	300	300	300	300	300	300	300

Textlisten und Grafiklisten

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Grafiklisten	100	400	400	400	400	500	500	500
Anzahl Textlisten	300	500	500	500	500	500	500	500

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl der Einträge pro Text- oder Grafikliste	30	256	256	256	256	500	500	500
Anzahl Grafikobjekte	1000	1000	1000	1000	1000	4000	4000	4000
Anzahl Textelemente	2500	10000	10000	10000	10000	40000	40000	40000

Skripte

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Skripte	--	50	50	50	50	100	100	100

Kommunikation

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Verbindungen	4	6	6	6	6	8	8	8
Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP"	4	8	8	8	8	8	8	8
Anzahl der maximal verbundenen Sm@rtClients (incl. ein ServiceClient)	2	2	8": 3 10": 2	8": 3 10": 2	8": 3 10": 2	3	3	3

Mobile Wireless

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Zonen	--	--	254	254	--	254	254	254
Anzahl Wirkbereiche	--	--	--	127	--	--	--	--
Anzahl zugeordneter Transponder - Zonen	--	--	255	255	--	--	--	--
Anzahl zugeordneter Transponder - Wirkbereiche	--	--	--	127	--	--	--	--

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Wirkbereiche (RFID)	--	--	--	--	127	--	--	--
Anzahl RFID-Tags, die Wirkbereichen (RFID) in einem Projekt zugeordnet werden können	--	--	--	--	127	--	--	--

Hilfesystem

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Länge eines Hilfetextes in Zeichen	320	320	320	320	320	500	500	500

Sprachen

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Runtimesprachen	5	16	16	16	16	32	32	32

Aufgabenplaner

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Zeitgetriggerte Aufgaben [1)]	10	48	48	48	48	48	48	48

1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

Benutzerverwaltung

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Anzahl Benutzergruppen	50	50	50	50	50	50	50	50
Anzahl Berechtigungen	32	32	32	32	32	32	32	32
Anzahl Benutzer	50	50	50	50	50	50	50	50

Projekt

	Mobile Panel 177	Mobile Panel 277	Mobile Panel 277 IWLAN V2	Mobile Panel 277F IWLAN V2	Mobile Panel 277F IWLAN (RFID Tag)	KTP400F Mobile	KTP700 Mobile / KTP700F Mobile	KTP900 Mobile / KTP900F Mobile
Größe der Projektdatei "*.fwc"	2 MB	6 MB	6 MB	6 MB	6 MB	12 MB	12 MB	12 MB

Siehe auch

S7-1200 Handbuch (http://support.automation.siemens.com/WW/view/en/36932465/0/de)

16.2.4 Comfort Panel

Comfort Panel

Die folgenden Tabellen über die Systemgrenzen unterstützen Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Systemgrenzen für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Hinweis

Grafiken auf Comfort Panels

Das Verwenden von 32-Bit-Grafiken vergrößert den Speicherbedarf der Projektdatei auf Comfort Panels.
Das Verwenden von jpeg-Grafiken verkleinert den Speicherbedarf der Projektdatei auf Comfort Panels.

Variablen

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Variablen im Projekt	1024	2048	4096	4096	4096
Anzahl PowerTags	--	--	--	--	--
Anzahl Elemente je Array	1000	1000	1000	1000	1000
Anzahl Lokale Variablen	500	1000	2000	2000	2000
Anzahl Strukturen	999	999	999	999	999
Anzahl Strukturelemente	400	400	400	400	400

Meldungen

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Meldeklassen	32	32	32	32	32
Anzahl Bitmeldungen	2000	4000	6000	6000	6000
Anzahl Analogmeldungen	50	200	200	200	200
Länge einer Meldung in Zeichen	80	80	80	80	80
Anzahl Prozesswerte je Meldung	8	8	8	8	8
Größe des Meldepuffers	256	1024	1024	1024	1024
Anzahl anstehende Meldeereignisse	64	500	500	500	500

Bilder

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Bilder	500	500	750	750	750
Anzahl Felder pro Bild	50	400	600	600	600
Anzahl Variablen pro Bild	50	400	400	400	400
Anzahl Komplexe Objekte pro Bild [1]	5	20	40	40	40

1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.

Rezepturen

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Rezepturen	100	300	500	500	500
Anzahl Elemente pro Rezeptur [1]	200	1000	2000	2000	2000
Nutzdatenlänge in kByte pro Datensatz	32	256	512	512	512
Anzahl Datensätze pro Rezeptur	200	500	1000	1000	1000
Reservierter Speicher für Datensätze im internen Flash	512 kB	2 MB	4 MB	4 MB	4 MB

1) Bei Verwendung von Arrays zählt jedes Arrayelement als ein Rezepturelement

Archive

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Archive	10	50	50	50	50
Anzahl Einträge je Archiv (incl. aller Archivsegmente) [1]	10000	20000	50000	50000	50000
Anzahl Archivsegmente	400	400	400	400	400
zyklischer Trigger für die Archivierung von Variablen	1 s	1 s	1 s	1 s	1 s
Anzahl archivierbarer Variablen je Archiv	100	2048	2048	2048	2048

1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

Kurven

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Kurven	50	300	400	400	400

Textlisten und Grafiklisten

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Grafiklisten	100	500	500	500	500
Anzahl Textlisten	300	500	500	500	500

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl der Einträge pro Text- oder Grafikliste	30	500	500	500	500
Anzahl Grafikobjekte	1000	4000	4000	4000	4000
Anzahl Textelemente	2500	40000	40000	40000	40000

Skripte

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Skripte	50	100	200	200	200

Kommunikation

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Verbindungen	4	8	8	8	8
Anzahl OPC-Verbindungen inkl. OPC UA	4	8	8	8	8
Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP"	4	8	8	8	8
Anzahl der maximal verbundenen Sm@rtClients (incl. ein ServiceClient)	2	3	3	2	1

Hilfesystem

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Länge eines Hilfetextes in Zeichen	500	500	500	500	500

Sprachen

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Anzahl Runtimesprachen	32	32	32	32	32

Aufgabenplaner

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Zeitgetriggerte Aufgaben [1)]	10	48	48	48	48

1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

Benutzerverwaltung

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
AnzahlBenutzergruppen	50	50	50	50	50
Anzahl Berechtigungen	32	32	32	32	32
Anzahl Benutzer	50	50	50	50	50

Projekt

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200 TP700 Comfort Outdoor TP1500 Comfort Outdoor	TP1500 KP1500	TP1900	TP2200
Größe der Projektdatei "*.fwc"	4 MB	12 MB	24 MB	24 MB	24 MB

Siehe auch

S7-1200 Handbuch (http://support.automation.siemens.com/WW/view/en/36932465/0/de)

16.2.5 WinCC Runtime Advanced

WinCC Runtime Advanced

Die folgenden Tabellen über die Systemgrenzen unterstützen Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Systemgrenzen für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Variablen

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Variablen im Projekt	6144	12288	24576
Anzahl PowerTags	128 - 4096	128 –8192	128-16384
Anzahl Elemente je Array	1600	1600	1600
Anzahl interne Variablen	2048	4096	8192
Anzahl Strukturen	999	999	999
Anzahl Strukturelemente	400	400	400

Meldungen

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Meldeklassen	32	32	32
Anzahl Bitmeldungen	4000	6000	6000
Anzahl Analogmeldungen	500	500	500
Länge einer Meldung in Zeichen	80	80	80
Anzahl Prozesswerte je Meldung	8	8	8
Größe des Meldepuffers	1024	2048	2048
Anzahl anstehende Meldeereignisse	500	500	500

Bilder

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Bilder	500	750	750
Anzahl Felder pro Bild	400	600	600
Anzahl Variablen pro Bild	400	400	400
Anzahl Komplexe Objekte pro Bild [1)]	40	40	40

1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.

Rezepturen

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Rezepturen	999	1000	1000
Anzahl Elemente pro Rezeptur [1)]	2000	2000	2000
Nutzdatenlänge in kByte pro Datensatz	256	512	512
Anzahl Datensätze pro Rezeptur	5000	5000	5000
Reservierter Speicher für Datensätze im internen Flash	--	--	--

1) Bei Verwendung von Arrays zählt jedes Arrayelement als ein Rezepturelement

Archive

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Archive	100	100	100
Anzahl Einträge je Archiv (incl. aller Archivsegmente) [1)]	500000	500000	500000
Anzahl Archivsegmente	400	400	400
zyklischer Trigger für die Archivierung von Variablen	1 s	1 s	1s
Anzahl archivierbarer Variablen je Archiv	6144	12288	24576

1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

Kurven

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Kurven	800	800	800

Textlisten und Grafiklisten

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Grafiklisten	500	500	500
Anzahl Textlisten	500	500	500
Anzahl der Einträge pro Text- oder Grafikliste	3500	3500	3500
Anzahl Grafikobjekte	2000	4000	4000
Anzahl Textelemente	30000	40000	40000

Skripte

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Skripte	200	200	200

Kommunikation

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Verbindungen	8	8	8
Anzahl OPC-Verbindungen inkl. OPC UA	8	8	8
Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP"	16	16	16
Anzahl maximal verbundener Sm@rtClients (incl. ein ServiceClient)	4 [1]	4 [1]	4 [1]

Hilfesystem

	KTP400 KP400	TP700, KP700 TP900, KP900 TP1200, KP1200	TP1500 KP1500	TP1900	TP2200
Länge eines Hilfetextes in Zeichen	320 [1] 500 [2]	320 [1] 500 [2]	320 [1] 500 [2]	320 [1] 500 [2]	320 [1] 500 [2]

1) Für Geräte mit Bediengeräteversion 12 oder kleiner
2) Für Geräte mit Bediengeräteversion größer 12

Sprachen

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Runtimesprachen	32	32	32

Aufgabenplaner

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Zeitgetriggerte Aufgaben [1)]	48	48	48

1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

Benutzerverwaltung

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Anzahl Benutzergruppen	50	50	50
Anzahl Berechtigungen	32	32	32
Anzahl Benutzer	100	100	100

Projekt

	WinCC Runtime Advanced Geräteversion kleiner 13	WinCC Runtime Advanced Geräteversion größer gleich 13	WinCC Runtime AdvancedGeräteversion größer gleich 14
Größe der Projektdatei "*.fwc"	Keine Begrenzung	Keine Begrenzung	Keine Begrenzung

Siehe auch

S7-1200 Handbuch (http://support.automation.siemens.com/WW/view/en/36932465/0/de)

16.2.6 WinCC Runtime Professional

WinCC Runtime Professional

Die folgenden Tabellen über die Systemgrenzen unterstützen Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Systemgrenzen für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

Variablen

	WinCC Runtime Professional
Anzahl Variablen im Projekt	500.000
Anzahl PowerTags	abhängig von der Lizenz
Anzahl Elemente je Array	2.000

Bilder

Grafikdateien, die größer als 100 MByte sind, werden in WinCC Runtime gegebenenfalls nicht angezeigt.

	WinCC Runtime Professional
Objekte pro Bild [1)]	Keine Grenze [2)]
Ebenen pro Bild	32
Bilder pro Projekt	Keine Grenze [2)]
Bildgröße in Pixel	10.000 x 10.000
Schachtelungstiefe von Bildobjekten	20
Anzahl Farben	24 Bit Farbtiefe, 8 Bit Alphakanal für Transparenz

1) Anzahl und Komplexität der Objekte haben Einfluss auf die Performance
2) Durch Systemressourcen begrenzt.

Meldungen

	WinCC Runtime Professional
Projektierbare Meldungen pro Server/Einzelplatz	150.000
Prozessvariablen pro Meldezeile	10

	WinCC Runtime Professional
Anwendertextblöcke pro Meldezeile	10
Meldeklassen	256
Meldeprioritäten	17 (0...16)

Meldungen in Runtime

	Maximal
Meldungen pro Meldearchiv	Keine Grenze [1]
Meldungen pro Kurzzeitarchivliste	1.000
Meldungen pro Langzeitarchivliste	1.000 [2]
Meldungen pro Meldeanzeige	5.000 [3]

1) Durch Systemressourcen begrenzt.
2) Auf Einzelplatz bzw. Server oder auf Client pro Server, wenn "LongTimeArchiveConsistency" auf "nein" gesetzt ist. Auf Einzelplatz, Server, Client, wenn "LongTimeArchiveConsistency" auf "ja" gesetzt ist.
3) Auf Einzelplatz bzw. Server oder auf Client pro Server.

Hinweis

Meldeschwall und Meldungsdauerlast können an einem Einzelplatz bzw. Server gleichzeitig erzeugt werden.

Archive

	WinCC Runtime Professional
Kurvenanzeige pro Bild	25
Kurven pro Kurvenanzeige	80
Werte pro Kurvenanzeige	Vom Tag Logging Server werden maximal 134.217.728 Bytes zu den Kurvenanzeigen übertragen. Wenn ein Prozesswert 20 Bytes enthält, können ca. 6,7 Millionen Werte angezeigt oder in eine ".csv"-Datei exportiert werden.
Tabellenanzeige pro Bild	25
Spalten pro Tabellenanzeige	12
Werte pro Tabellenanzeige	30.000
Archive pro Einzelplatz/Server	100
Archivvariablen pro Einzelplatz/Server [1]	80.000

1) Abhängig vom verwendeten Logging-PowerPack für Archivvariablen. 500 Archivvariablen sind in der Basisversion enthalten.

Hinweis

Bei Kombination der Maximalwerte können hohe Bildanwahlzeiten auftreten.

Rezepturen

	WinCC Runtime Professional
Anzahl Rezepturen	Keine Grenze
Anzahl Rezepturelemente [1]	500
Anzahl Rezepturdatensätze	10.000
Anzahl Sichten	Keine Grenze

1) Maximal 1.000.000 Rezepturelemente ingesamt

Beispiel für ein zulässiges Mengengerüst in Runtime

Die maximale Anzahl von 1.000.000 Rezepturelementen erreichen Sie z. B. mit folgenden Konfigurationen:

- 10.000 Rezepturdatensätze mit jeweils 100 Rezepturelementen
- 2.000 Rezepturdatensätze mit jeweils 500 Rezepturelementen

Annahme: Jedes Rezepturelement ist mit einer Variable verbunden.

Beachten Sie, dass ein solches Mengengerüst auch von der eingesetzten Hardware und durch andere Projektierungen beeinflusst wird.

Protokolle

	WinCC Runtime Professional
Projektierbare Protokolle	Keine Grenze [1]
Protokollzeilen pro Detailseite	66
Variablen pro Protokoll [2]	300

1) Durch Systemressourcen begrenzt.
2) Die Anzahl der Variablen pro Protokoll ist abhängig von der Performance der Prozesskommunikation.

Protokolle in Runtime

	Maximal
Gleichzeitig laufende Meldefolgeprotokolle pro Server/Client	1
Gleichzeitig laufende Meldearchivprotokolle	3

Benutzerverwaltung

	Maximal
Anzahl Benutzer	200

Kommunikation

	S7-1200	S7-1511	S7-1513	S7-1516	S7-1518
Maximale Anzahl an WinCC-Systemen	2	20	30	41	62
ISO-on-TCP Verbindungen	-	64, davon 4 reserviert für ES	96, davon 4 reserviert für ES	128, davon 4 reserviert für ES	192, 4 reserviert für ES
Anzahl maximal verbundener PLCs pro Runtime			64		
Anzahl maximal verbundener PLCs mit Softnet über eine Netzwerkkarte			64		

- Bei der Kommunikation mit S7-1200 Steuerungen ist zu beachten, dass pro Steuerung maximal 200 Variablen projektiert werden sollten.
- Wenn mehrere HMI-Geräte auf eine Steuerung zugreifen, gilt diese Grenze für alle HMI-Geräte gemeinsam.
- Wieviele HMI-Geräte gleichzeitig auf eine SIMATIC S7-1200 zugreifen können, entnehmen Sie aus dem Handbuch der Steuerung.
  Beitrag-ID: 36932465 (https://support.industry.siemens.com/cs/document/36932465/simatic-s7-s7-1200-automatisierungssystem?dti=0&lc=de-WW)

Relation HMI-Verbindung und Verbindungsressource

Eine HMI-Verbindung zu WinCC RT Professional nimmt bis zu drei Verbindungsressourcen in Anspruch.

Hinweis

Unterschiedliche Grenzen bei Verbindungen über PROFIBUS und PROFINET

Beachten Sie, dass bei Verbindungen über PROFIBUS andere Grenzen bezüglich der Relationen zwischen den HMI-Verbindungen und den Verbindungsressourcen als bei PROFINET gelten.

Bei einer Kommunikation über PROFIBUS und SIMATIC NET müssen Sie auch die Grenzen der SIMATIC NET-Komponenten (CPs) berücksichtigen.

Weitere Informationen zu SIMATIC NET-Komponenten, finden Sie im jeweiligen Handbuch zu SIMATIC NET: "PC-Software SIMATIC NET PC".

Konfigurationen - Mengengerüst im Mehrplatz-System

	WinCC Runtime Professional
WinCC-Clients in einem System	32 [1) 2)]
Web Clients in einem System	150 [2) 3)]

1) Wenn der Server auch als Bedienplatz verwendet wird, dann verringert sich die Anzahl für diesen Server auf vier Clients.
2) Mischkonfiguration: 32 Clients + 3 Web-Clients
3) Mischkonfiguration: 50 Web-Clients + 1 WinCC-Client

16.2.7 Orientierungshilfe Performance in Runtime (WinCC Runtime Professional)

WinCC Runtime Professional

Die folgenden Tabellen unterstützen Sie bei der Abschätzung, mit welcher Performance das Projekt in Runtime betrieben werden kann.

Hinweis

Die angegebenen Werte sind stark von der eingesetzten Hardware abhängig.

Bilder in Runtime

Bildwechsel von einem Leerbild zu ...	Zeit in Sekunden
Bild mit Standardobjekten (100 Objekte)	1
Bild mit 2.480 E/A-Feldern (8 interne Variablen)	2
Bild mit 1.000 E/A-Feldern (1.000 interne Variablen)	1
Bild von 10 MByte Größe (Bitmap)	1
Meldeanzeige	2
Tabellenanzeige mit 4 Spalten à jeweils 120 Werten [1)]	1

1) Die angegebenen Werte gelten für Daten aus "Tag Logging Fast".

Meldungen in Runtime

	Maximal
Meldungsdauerlast ohne Verlust (Einzelplatz/Server)	10/sec
Meldeschwall (Einzelplatz/Server)	2.000/10 sec alle 5 min [1)]

1) Wenn der der Abstand zum nächsten Meldeschwall weniger als fünf Minuten beträgt, können Meldungen verloren gehen.

Hinweis

Meldeschwall und Meldungsdauerlast können an einem Einzelplatz bzw. Server gleichzeitig erzeugt werden.

Archive in Runtime

	Maximal
Archivierung in Datenbank für Server/Einzelplatz (Tag Logging Fast)	5.000 Werte/Sekunde [1)]
Archivierung in Datenbank für Server/Einzelplatz (Tag Logging Slow)	1.000 Werte/Sekunde [1) 2)]
Anzahl ausgedruckter Werte in einem Protokoll	Die Anzahl der ausgedruckten Werte richtet sich nach der Anzahl der dargestellten Werte in der Kurvenanzeige.

1) Die angegebenen Werte gelten für die Archivierung ohne Signierung der Daten.
2) Bei Tag Logging Slow müssen Sie bei gleichen Mengengerüsten mit längeren Bildanwahlzeiten rechnen als bei Tag Logging Fast.

Rezepturen in Runtime - Randbedingungen

Eingesetzte Hardware	Eingesetzte Software	Projektierung
Intel® Core™ i3-6100U 2.30 GHz 4 GByte RAM Keine Hardwarekopplung	Windows 7 WinCC V14.0 SP1	Jeweils eine WinCC-Variable pro Feld Jeweils 300.000 Einträge: - 10 Felder mit 30.000 Datensätzen - 500 Felder mit 600 Datensätzen

Rezepturen in Runtime - Ermittelte Werte

	10 Felder	500 Felder
Bildwechsel von einem Bild zu einem Bild mit Rezepturanzeige. Messergebnis ist abhängig vom Füllstand des Controls: Beim ersten Laden oder bei umfangreichen Projektierungsänderungen im Anwenderarchiv dauert die vollständige Anzeige bis zu 15 Sekunden.	1 Sekunde	5 Sekunden
Datensatz lesen: Lesen des Wertes in die entsprechende Variable nach Klicken auf die Schaltfläche in der Rezepturanzeige.	1 - 2 Sekunden [1)]	n Sekunden [2)]

	10 Felder	500 Felder
Datensatz schreiben: Schreiben des Wertes in die entsprechenden Variablen schreiben und Anzeige der Variablen in EA-Feld nach Klicken auf die Schaltfläche in der Rezepturanzeige.	1 - 3 Sekunden [1)]	n Sekunden [2)]
Fokuswechsel vom ersten zum letzten Datensatz.	1 - 2 Sekunden	1 - 2 Sekunden

1) 10 Felder mit insgesamt 10 Variablen.
2) 500 Felder mit insgesamt 500 Variablen.

Protokolle in Runtime

	Maximal
Gleichzeitig laufende Meldefolgeprotokolle pro Server/Client	1
Gleichzeitig laufende Meldearchivprotokolle	3

16.2.8 Orientierungshilfe Kommunikation in Runtime (WinCC Runtime Professional)

WinCC Runtime Professional

Installieren Sie die Software SIMATIC NET von der mitgelieferten DVD. Um WinCC Runtime Professional mit einer S7-1200 / S7-1500 Steuerung zu verwenden, dürfen die spezifischen Systemgrenzen der jeweiligen S7-Steuerung nicht überschritten werden. Im Folgenden werden die Systemgrenzen für Tags erklärt, deren Werte von einer S7-1200 / S7-1500 übermittelt werden. Zur Ermittlung der Werte wird das Tool “WinCC Channel Diagnosis“ verwendet.

Hinweis

Die für Ihre S7-1200 / S7-1500 Steuerung spezifischen Systemgrenzen entnehmen Sie den Parametertabellen. (https://support.industry.siemens.com/cs/ww/de/view/98699910)

Attribute

Attribut	Beschreibung
Plc Attributes (max)	Dieser Wert bezeichnet die maximale Anzahl der Variablen, die zyklisch gelesen werden können. Standardmäßig ist die Kommunikationslast auf einer S7-Steuerung nicht unter 30% (Standardwert: 50%) konfiguriert.
Minimum Plc Attributes (free)	Der Wert “Minimum Plc Attributes (free)“ beruht auf Erfahrungswerten. Der Wert gilt für den eingeschwungenen Zustand ohne Bildwechsel. Die Mindestanzahl „Minimum Plc Attributes (free)" kann temporär niedriger sein. Wenn die Zahl gegen 0 geht, beginnt das zyklische Lesen mit niedrigen Zykluszeiten, womit die Kommunikation zur S7-Steuerung kurzfristig verlangsamt werden kann.
Anzahl WinCC Runtime Professional	Die "Anzahl WinCC Runtime Professional" drückt aus, wieviele WinCC Runtime Professional-Systeme sich maximal zu einer S7-Steuerung verbinden können. In der Anzahl ist berücksichtigt, dass eine WinCC ES mit der S7-Steuerung verbunden ist.

Hinweis

Die "Plc Attributes" können von einem oder mehreren WinCC Runtime Systemen verbraucht werden. Pro S7-Steuerung ist daher die Anzahl gleichzeitig angemeldeter Tags aller verbundenen WinCC Runtime Systeme zu addieren. Wie sich die Anzahl gleichzeitig angemeldeter Tags ergibt, erfahren Sie im Abschnitt "Wesentliche Einflussgrößen für gleichzeitig angemeldete Tags".

Beispiel

Hinweis

Um eine stabile Kommunikation zur S7-Steuerung zu gewährleisten, sollten Sie die "Anzahl gleichzeitig angemeldeter Tags pro S7" nicht überschreiten.

Die Werte für die Spalten "Plc Attributes (max)", "Minimum Plc Attributes (free)" der Parametertabellen können Sie aus dem WinCC Channel Diagnosis Tool unter "CCS7PLUSCHANNEL" und der entsprechenden Verbindung ablesen. Die Anzahl gleichzeitig angemeldeter Tags pro S7-Steuerung wird durch den Zähler "Plc Tag Subscriptions" ermittelt: Pro Zykluszeit wird die Anzahl angemeldeter Tags dargestellt. Im folgenden Beispiel beträgt die Anzahl der angemeldeten Tags 2003 (500 ms: 103 Tags, 1 s: 400 Tags, 2 s: 500 Tags, 5 s: 1000 Tags).

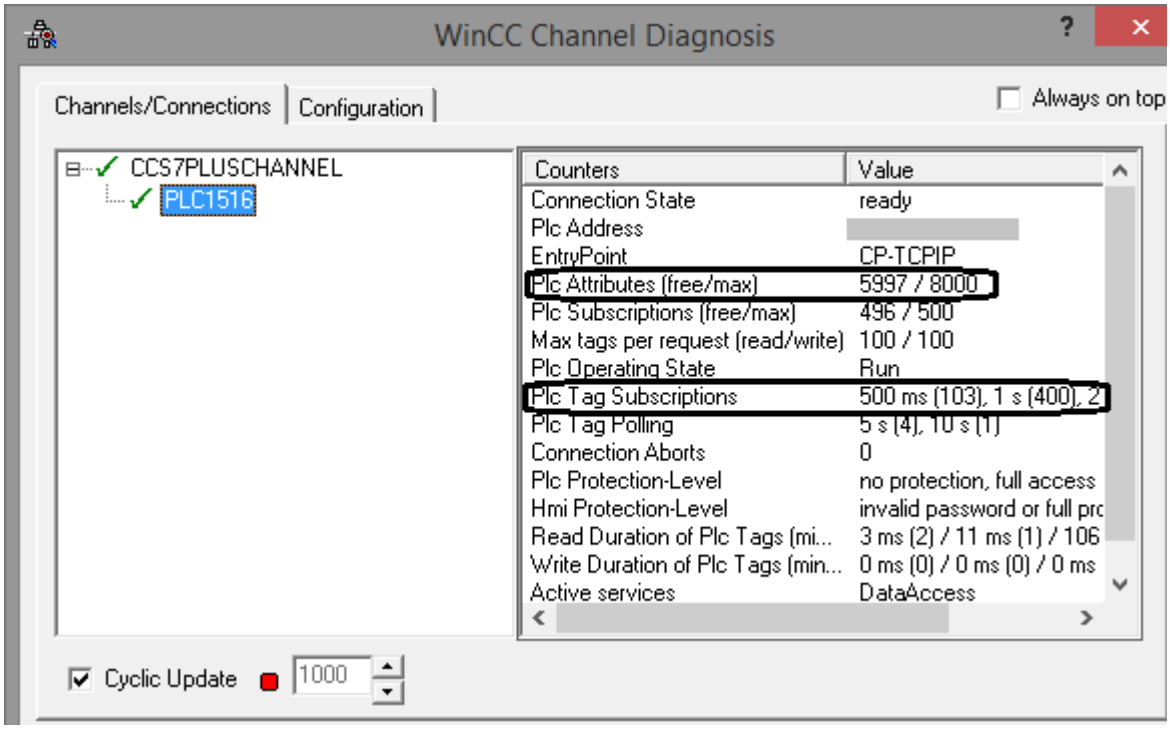

Wesentliche Einflussgrößen für gleichzeitig angemeldete Tags

Die Anzahl der gleichzeitig angemeldeten Tags ergibt sich aus folgender Tabelle. Hierbei sind Zykluszeiten kleiner gleich 5 Sekunden relevant. Generell sind nur die Tags relevant, die von einer S7-Steuerung erfasst werden. Im obigen Beispiel sind dies die Tags der Verbindung "PLC1516".

Funktionalität	Gleichzeitig angemeldete Tags
Archivierung	Summe aller Tags pro Erfassungszyklus.
HMI-Meldungen	Anzahl Tags für Bit-Alarme + Anzahl Tags für Analog-Alarme + Anzahl Begleitwerte aus Tags pro Alarm.
Aufgabenplaner	Summe aller verwendeter Tags pro zyklischem Ereignis. Bei SmartTags sind nur die SmartTags mit Verwendung des Caches relevant. Anzahl verwendeter Trigger-Tags.
Skripte	Betroffen sind zyklische Skripte und zwar dort die Funktionen, die einen Cache verwenden. Hinweise für die Verwendung des Caches sind in der Dokumentation enthalten. Beispiele: • HmiRuntime.Tags().Read 0. • SmartTags bei Verwendung des Caches. Alle betroffenen Tags gehen in die Zählung ein.
Screen	Anzahl direkt verbundener Tags an Objekt-Eigenschaften. Anzahl direkt verwendeter Tags in Controls mit Zyklen (z.B. Gauge Control, Kurvenanzeigen, Tabellenanzeigen). Anzahl verwendeter Tags in zyklischen Animationen, CScripts, VB-Scripts. Anzahl verwendeter Trigger-Tags.

Funktionalität	Gleichzeitig angemeldete Tags
OPC-Server (DA / UA)	Summe aller zyklisch angemeldeter Items.
Control Development	Summe aller zyklisch gelesener Tags.

Siehe auch

S7-1200 / S7-1500 Parametertabellen (https://support.industry.siemens.com/cs/ww/de/view/98699910)

Optionen

17

17.1 WinCC Engineering V15.1 - Optionen

WinCC Optionen

Eine detaillierte Beschreibung zu WinCC Optionen finden Sie im Handbuch "WinCC Engineering V15.1 - Optionen".

Umstieg auf WinCC im TIA Portal 18

18.1 Übersicht

Übersicht zum Kapitel "Umstieg auf WinCC im TIA Portal"

SIMATIC WinCC bietet im TIA Portal eine Reihe funktioneller Änderungen. Einige Funktionen unterscheiden sich von den Funktionen, die Sie aus gewohnten Umgebungen wie WinCC V7 oder WinCC flexible her kennen.

Dieses Dokument gibt einen Überblick über spezielle Funktionen und Vorgehensweisen in SIMATIC WinCC im TIA Portal.

Es sind Funktionen und Vorgehensweisen, die sich grundlegend von den Versionen WinCC V7 und WinCC flexible unterscheiden oder einen anderen Namen tragen.

18.2 WinCC V7

18.2.1 Aktionen

Aktionen in WinCC V7

WinCC ermöglicht den Einsatz von Aktionen zur Dynamisierung der Abläufe in Ihrem WinCC Projekt. Diese Aktionen sind in der Sprache ANSI-C geschrieben. Aktionen werden durch einen Trigger, also durch ein auslösendes Ereignis, zum Ablauf gebracht.

Aktionen können für folgende bildunabhängige Hintergrundtätigkeiten eingesetzt werden:

- der tägliche Ausdruck eines Protokolls
- die Überwachung von Variablen
- die Ausführung von Berechnungen

Wie projektiere ich Aufgaben mit WinCC im TIA Portal?

Aufgaben projektieren Sie im TIA Portal mit dem Aufgabenplaner von WinCC.

Im Aufgabenplaner projektieren Sie Aufgaben, die unabhängig vom Bild im Hintergrund ausgeführt werden. Aktionen werden auf Aufgaben vom Typ Funktionsliste abgebildet.

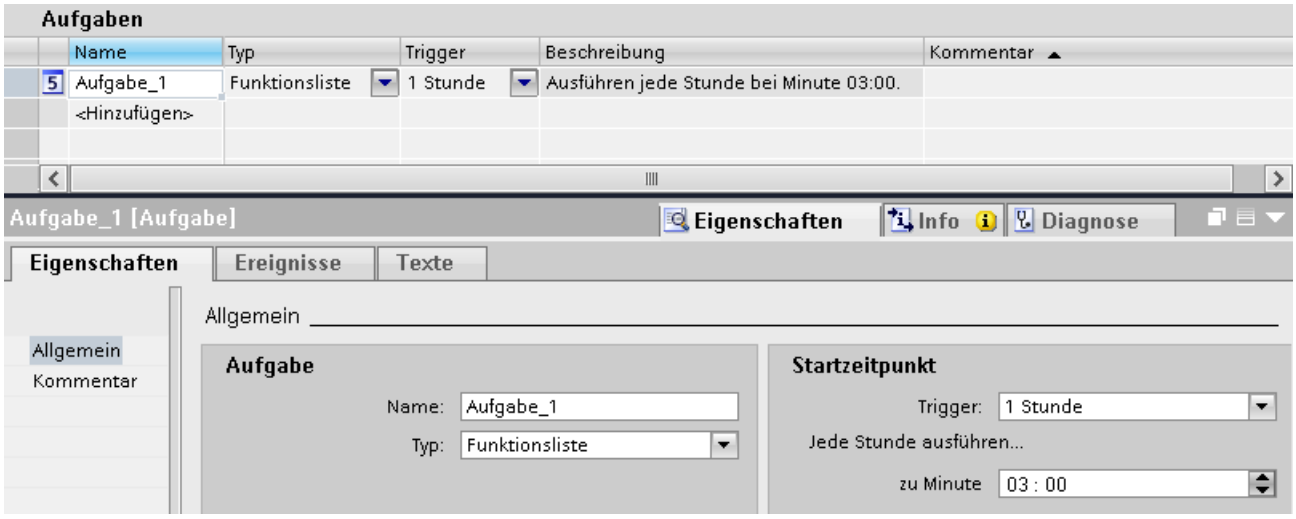

An jede Aufgabe projektieren Sie dabei genau eine Funktionsliste, die Zeile für Zeile angearbeitet wird. Oder Sie projektieren ein C- oder VB-Skript. Zum starten der Aufgabe definieren Sie einen Trigger.

Nähere Hinweise hierzu finden Sie unter:

Arbeitsbereich des Editors Aufgabenplaner (Seite 1872)

Beispiel: Lesen des aktuellen Benutzernamens (Seite 1892)

Siehe auch

Arbeitsbereich des Editors Aufgabenplaner (Seite 1872)

Beispiel: Lesen des aktuellen Benutzernamens (Seite 1892)

18.2.2 Anwender-Objekte

Anwender-Objekte in WinCC V7

Die Objekte der Objektpalette können kombiniert werden, indem aus einer Mehrfachselektion eine Gruppe oder ein Anwender-Objekt gebildet wird. Zusätzlich kann eine Gruppe oder ein kombiniertes Objekt in die Projektbibliothek eingefügt werden.

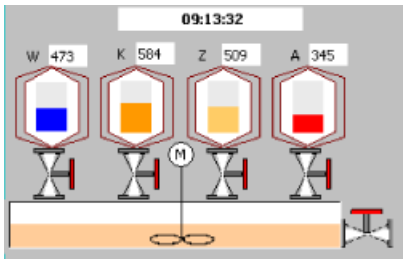

Ein Anwender-Objekt erlaubt, gegenüber der Gruppe, die individuelle Anpassung der Eigenschaften und Ereignisse, die im Fenster "Objekteigenschaften" angezeigt und verändert

werden können. Ein Anwender-Objekt kann wie ein einzelnes Objekt der Objektpalette im Graphics Designer bearbeitet werden.

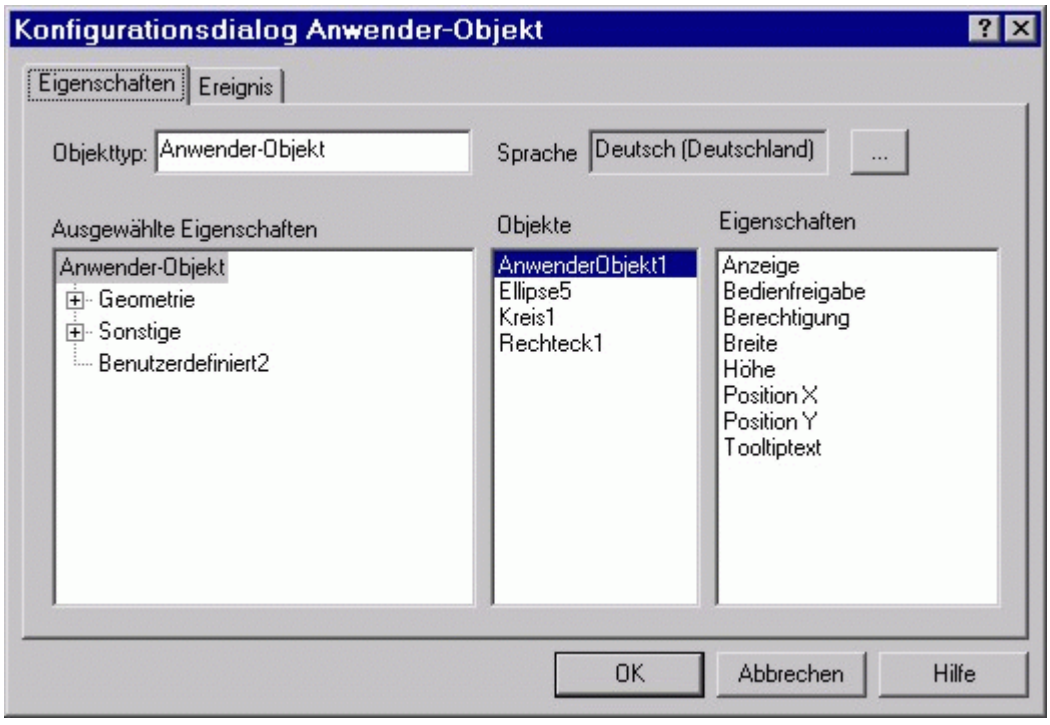

Wie projektiere ich Bildbausteine mit WinCC im TIA Portal?

In WinCC verwenden Sie Bildbausteine, um individuell projektierte Anzeige- und Bedienobjekte zu erstellen. Gegenüber Anwender-Objekten sind Bildbausteine um die zentrale Änderbarkeit erweitert.

Um Bildbausteine zu erstellen und die Eigenschaften zu bearbeiten, bietet WinCC den Editor "Bildbausteine" innerhalb der Projektbibliothek an.

Um die zentrale Änderbarkeit zu unterstützen, beruhen Bildbausteine auf einem Typ-Instanz-Modell. In Typen legen Sie zentral Eigenschaften für einen Bildbaustein an. Die Instanzen stellen lokale Verwendungsstellen der Typen dar.

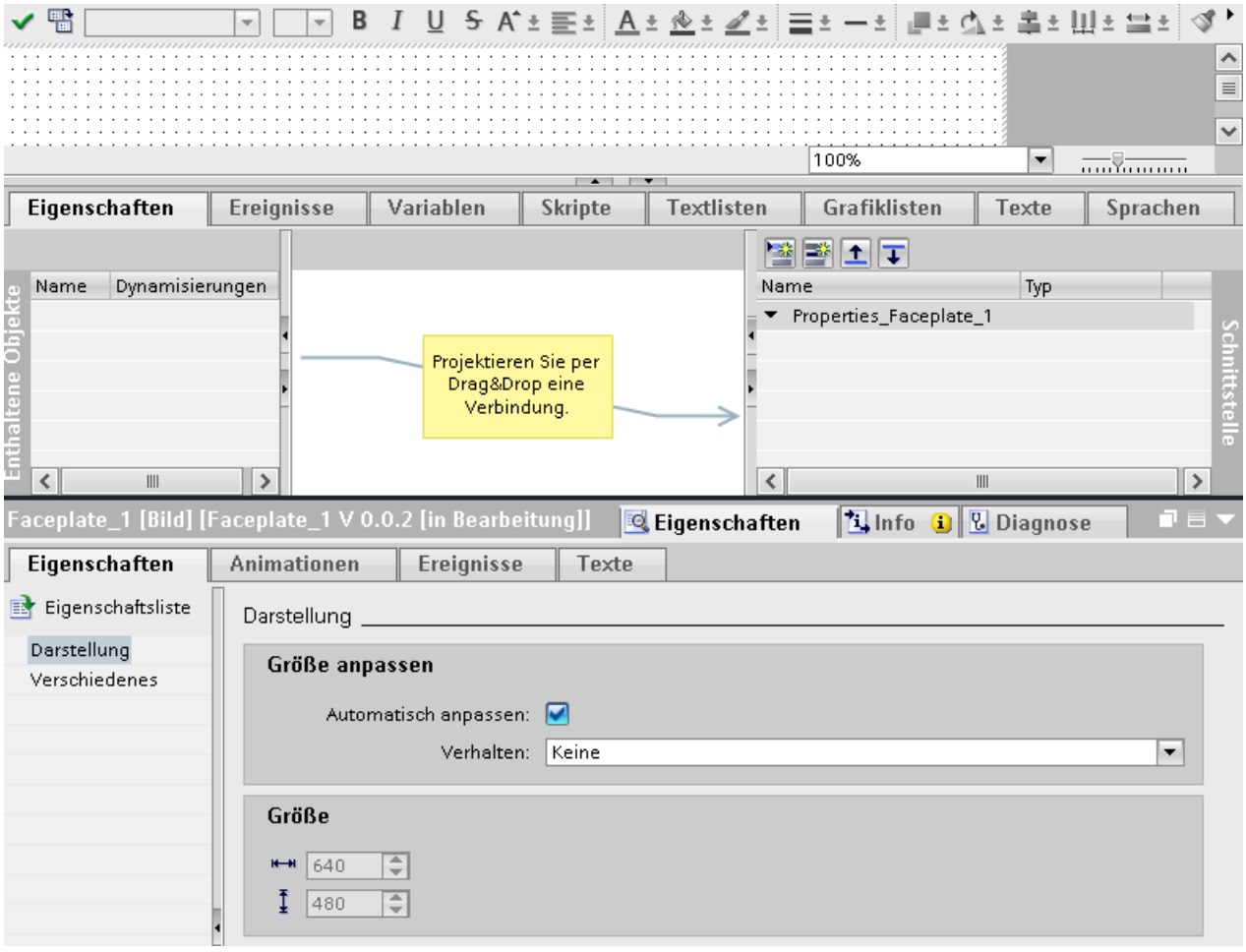

Nähere Hinweise hierzu finden Sie unter:

Grundlage zu Bildbausteinen (Seite 273)

Beispiel: Bildbaustein projektieren (Seite 305)

Siehe auch

Grundlage zu Bildbausteinen (Seite 273)

Beispiel: Bildbaustein projektieren (Seite 305)

18.2.3 Benutzerverwaltung

Benutzerverwaltung in WinCC V7

Der Editor "User Administrator" dient zum Aufbau und Verwaltung einer Benutzerverwaltung.

Im User Administrator können Zugriffsrechte für WinCC-Funktionen und WinCC-Benutzer eingerichtet und verwaltet werden.

Weitere Funktionen wie das zeitlich begrenzte Login oder das Login über Schlüsselschalter oder Chipkarte können im User Administrator einem Benutzer zugeordnet werden.

Den Editor "User Administrator" verwenden Sie beim Engineering und in Runtime.

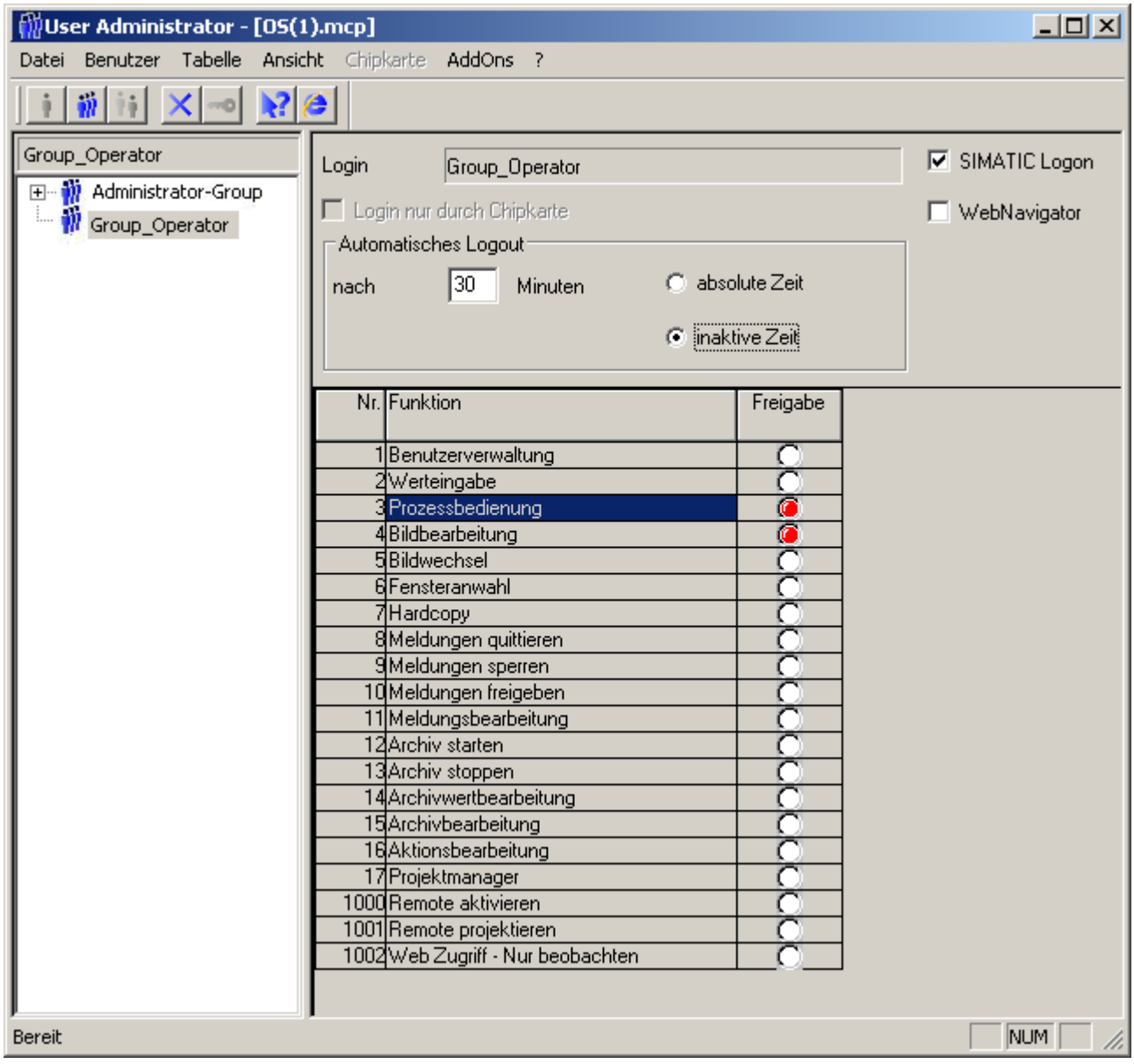

Wie projektiere ich eine Benutzerverwaltung mit WinCC im TIA Portal?

- Benutzeradministration im Engineering System
  Sie administrieren Benutzer, Benutzergruppen und Berechtigungen zentral in der Benutzerverwaltung von WinCC. Sie übertragen Benutzer und Benutzergruppen zusammen mit dem Projekt auf das Bediengerät.
  Benutzer sind werden immer einer Gruppe zugeordnet und besitzen damit die Rechte der Gruppe.

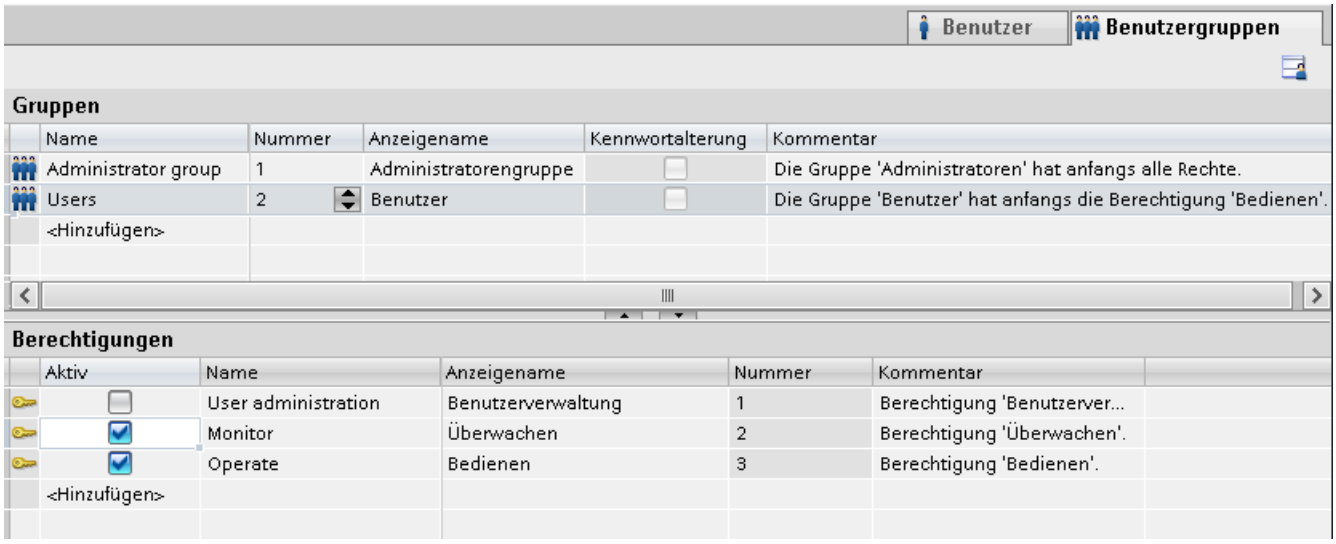

- Benutzeradministration in Runtime
  Um auch in Runtime die Benutzer zu administrieren, verwenden Sie eine Benutzeranzeige in einem Bild. Über die Benutzeranzeige verwalten Sie Benutzer und Kennwörter direkt am Bediengerät.

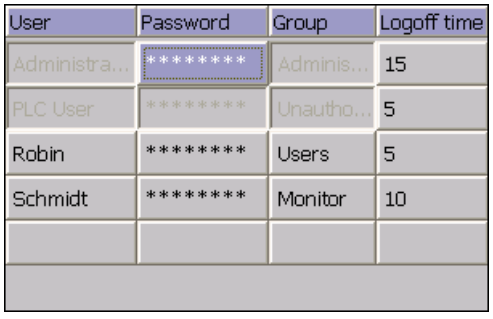

Nähere Hinweise hierzu finden Sie unter:

Aufbau der Benutzerverwaltung (Seite 1368)

Benutzeranzeige (Seite 1402)

Siehe auch

Aufbau der Benutzerverwaltung (Seite 1368)

Benutzeranzeige (Seite 1402)

18.2.4 Direktverbindung

Direktverbindungen in WinCC V7

Die Direktverbindung wird zur Reaktion auf Ereignisse verwendet. Wenn in Runtime dieses Ereignis eintritt, so wird der Wert eines Quellelementes für ein Zielelement übernommen.

Beispiele für das Ereignis "Mausklick":

- Das Füllmuster des Objektes "Rechteck1" an das Füllmuster des Objektes "Kreis1" angepasst.
- Ein Bildwechsel durchgeführt.

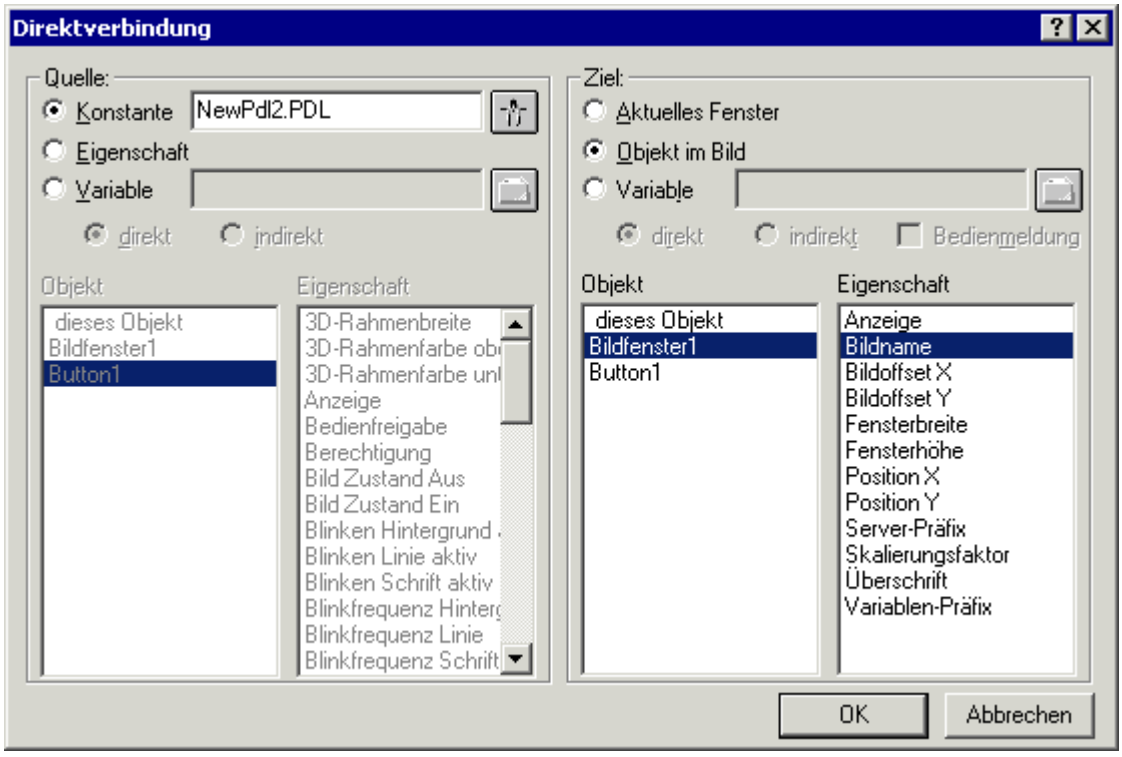

Wie projektiere ich Systemfunktionen mit WinCC im TIA Portal?

Direktverbindungen werden mit WinCC auf Systemfunktionen abgebildet.

Systemfunktionen verwenden Sie in einer Funktionsliste oder in einer benutzerdefinierten Funktion. Mit der Funktionsliste führen Sie beim Eintreten des Ereignisses die projektierten Systemfunktionen aus.

Wenn Sie eine Funktionsliste projektieren, wählen Sie die Systemfunktionen aus einer nach Kategorien geordneten Auswahlliste aus.

Zur Realisierung der Direktverbindung sind spezielle Funktionen integriert, wie z.B. "SetzeEigenschaftDurchEigenschaft".

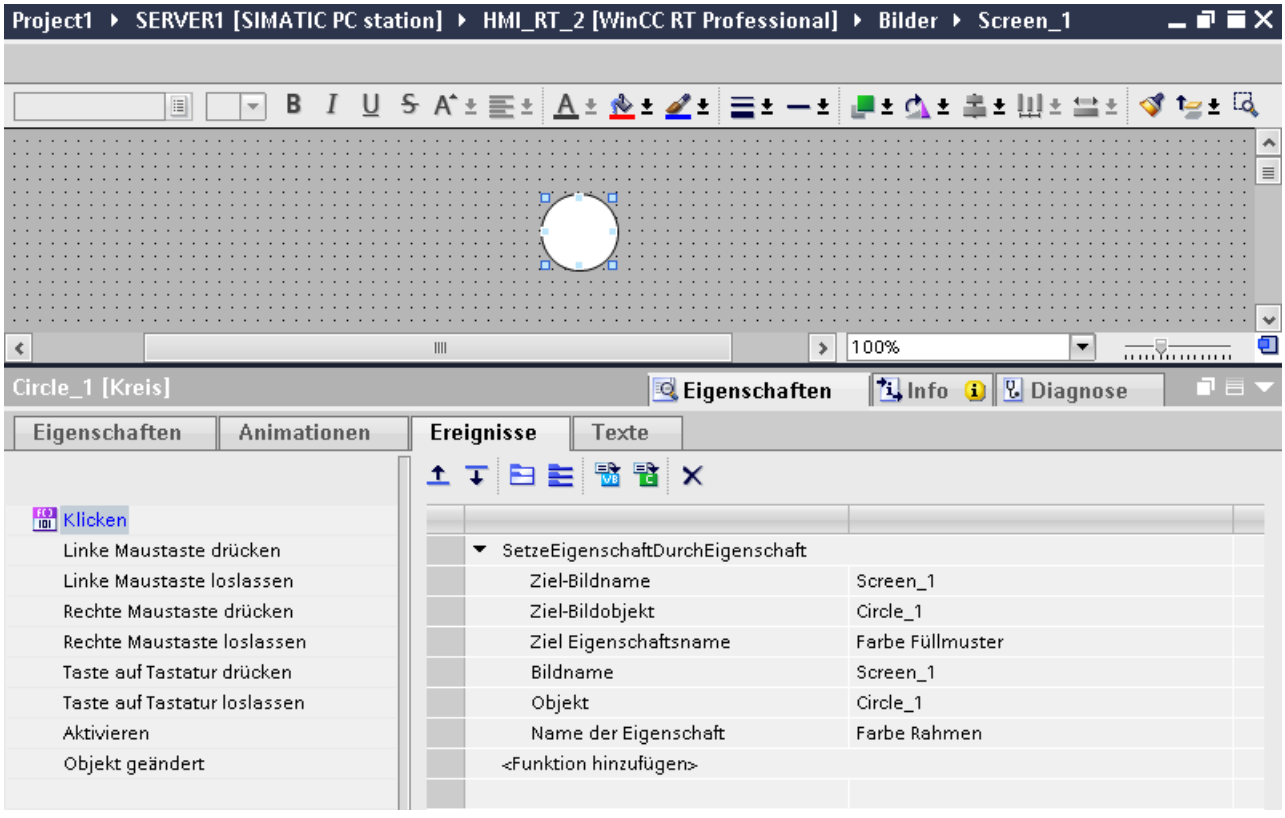

Nähere Hinweise hierzu finden Sie unter:

Systemfunktionen (Seite 1433)

Funktionsliste projektieren (Seite 1440)

Siehe auch

Systemfunktionen (Seite 1433)

Funktionsliste projektieren (Seite 1440)

18.2.5 Dynamic Wizard

Dynamic Wizard in WinCC V7

Mit dem Dynamic Wizard können Sie ein Objekt über C-Aktionen dynamisieren. Bei der Ausführung eines Wizards werden vorgefertigte C-Aktionen, z. B. Bit setzen oder Bit rücksetzen und Triggerereignisse, festgelegt und in den Objekteigenschaften hinterlegt.

Die C-Aktionen können Sie bei Bedarf in den Objekteigenschaften über die Registerkarte Ereignisse verändern.

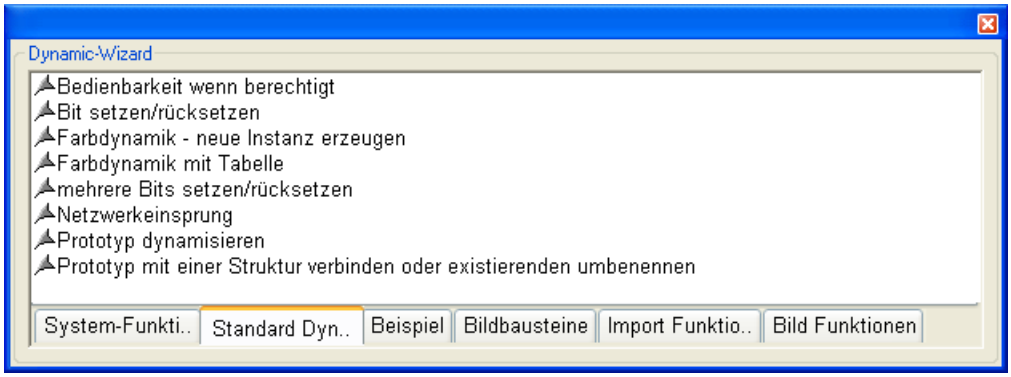

Wie projektiere ich Systemfunktionen mit WinCC im TIA Portal?

Objekte dynamisieren Sie in WinCC u. a. mithilfe von Systemfunktionen.

Systemfunktionen verwenden Sie in einer Funktionsliste oder in einer benutzerdefinierten Funktion. Mit der Funktionsliste führen Sie beim Eintreten des Ereignisses die projektierten Systemfunktionen aus.

Wenn Sie eine Funktionsliste projektieren, wählen Sie die Systemfunktionen aus einer nach Kategorien geordneten Auswahlliste aus.

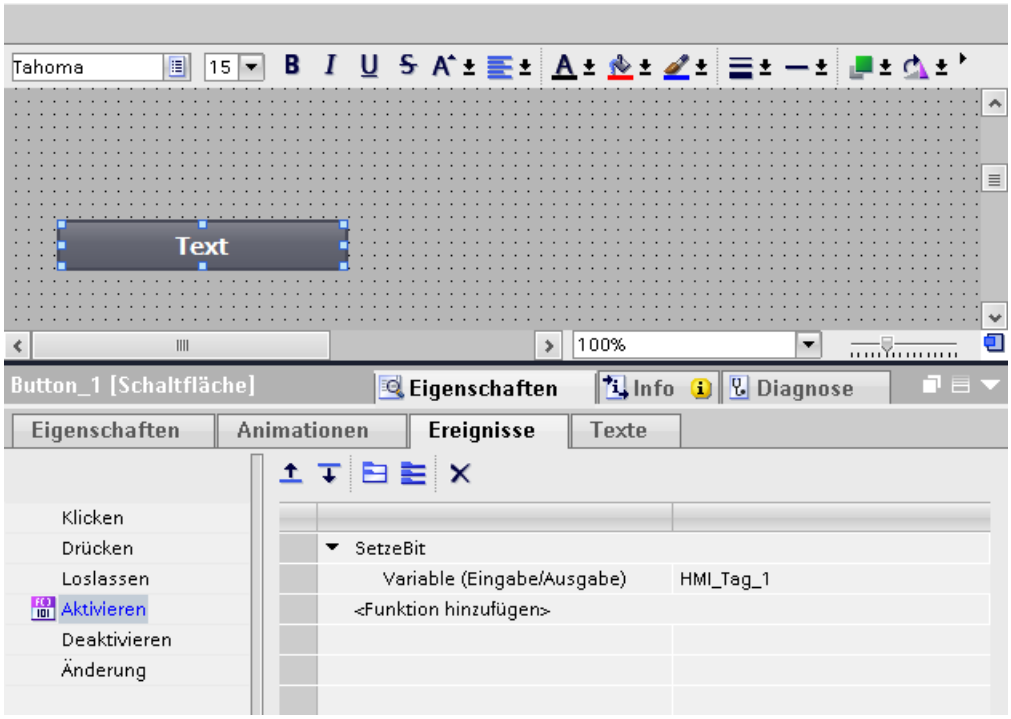

Nähere Hinweise hierzu finden Sie unter:

Systemfunktionen (Seite 1433)

Funktionsliste projektieren (Seite 1440)

Siehe auch

Systemfunktionen (Seite 1433)

Funktionsliste projektieren (Seite 1440)

18.2.6 Dynamik Dialog

Dynamik-Dialog in WinCC V7

Der Dynamik-Dialog wird zur Dynamisierung von Objekteigenschaften verwendet. Im Dynamik-Dialog formulieren Sie einen Ausdruck, indem Sie Variablen, Funktionen und arithmetische Operatoren verwenden.

Der Wert des Ausdrucks, der Status sowie der Quality Code der im Ausdruck verwendeten Variablen werden benutzt, um in Runtime den Wert der Objekteigenschaft zu bilden.

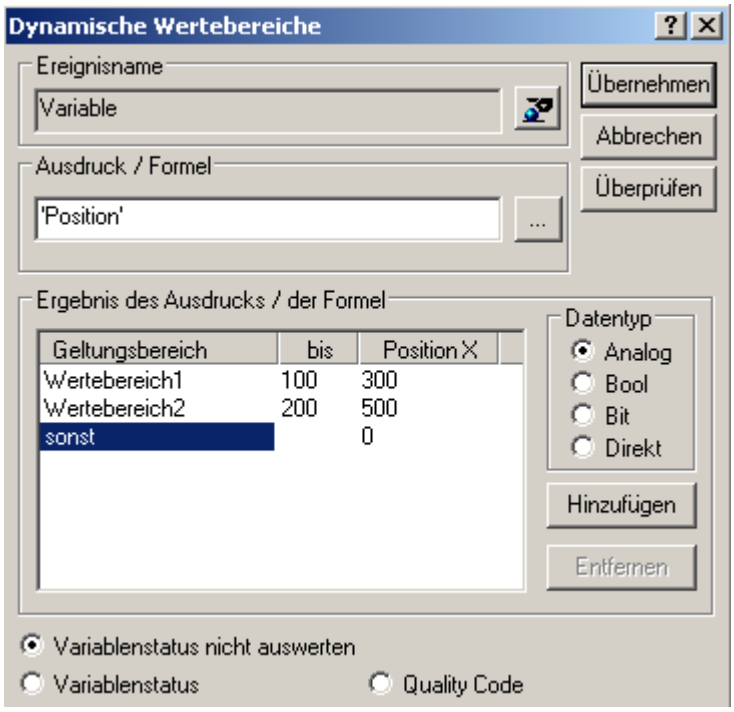

Der Dynamik-Dialog wird z. B. für folgende Zwecke eingesetzt:

- Wertebereiche einer Variablen auf Farben abbilden.
- Einzelne Bits einer Variablen überwachen und den Wert des Bits auf Farben oder Texte abbilden.
- Eine Boole'sche Variable überwachen und den Wert der Variablen auf Farben oder Texte abbilden.
- Den Status einer Variablen überwachen.
- Den Quality Code einer Variablen überwachen.

Wie projektiere ich Animationen mit WinCC im TIA Portal?

Dynamisierungen realisieren Sie z. B. durch Animationen über die Eigenschaftsliste eines Objekts unter "Eigenschaften > Animationen".

Sie verändern z. B. die Hintergrundfarbe eines Objekts dynamisch.

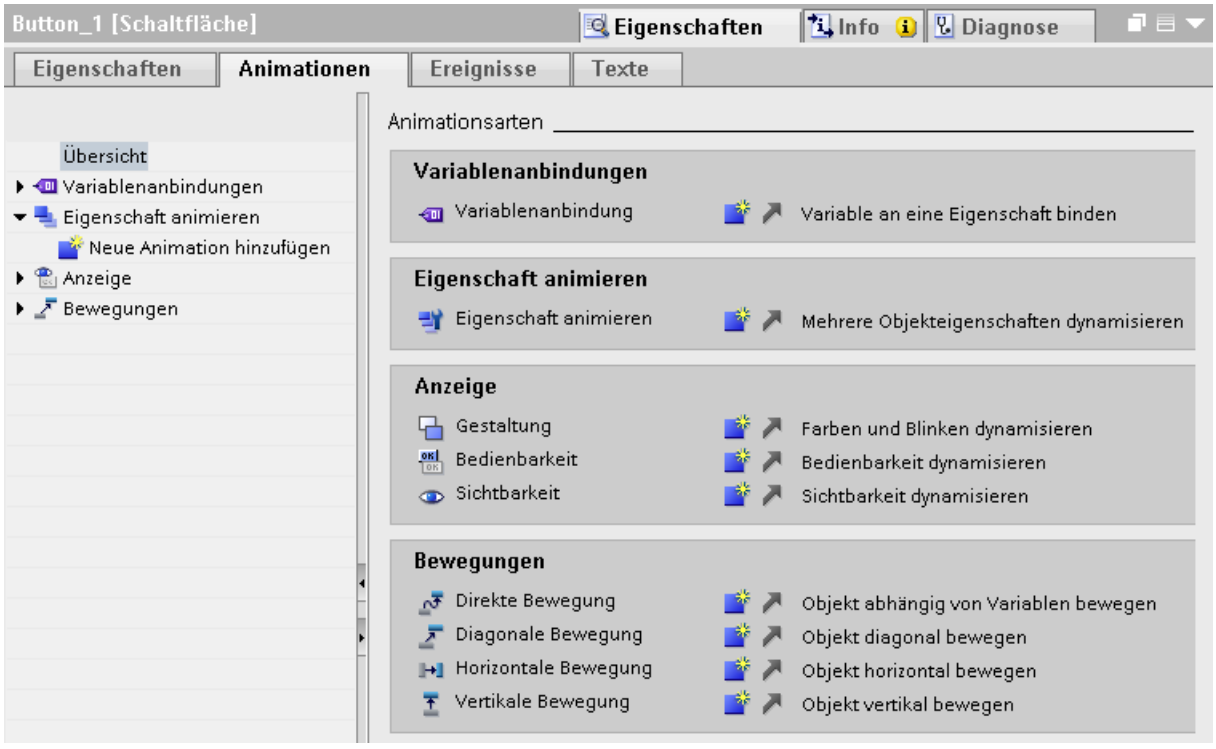

Die Dynamisierungen des Dynamik-Dialogs können Sie insbesondere mithilfe der Funktion "Eigenschaft animieren" projektieren.

Einfache Animationen sind zusätzlich auch über eine Variablenanbindung an Anzeige- und Bedienobjekte möglich.

Nähere Hinweise hierzu finden Sie unter:

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Neue Animation projektieren (Seite 217)

Siehe auch

Grundlagen zum Dynamisieren in der Eigenschaftsliste (Seite 239)

Neue Animation projektieren (Seite 217)

18.2.7 Effizientes Arbeiten

Massendatenverarbeitung in WinCC V7

Das WinCC ConfigurationTool ist ein Excel Add-In. Das Tool ermöglicht die einfache Projektierung von Massendaten, z. B. Variablen.

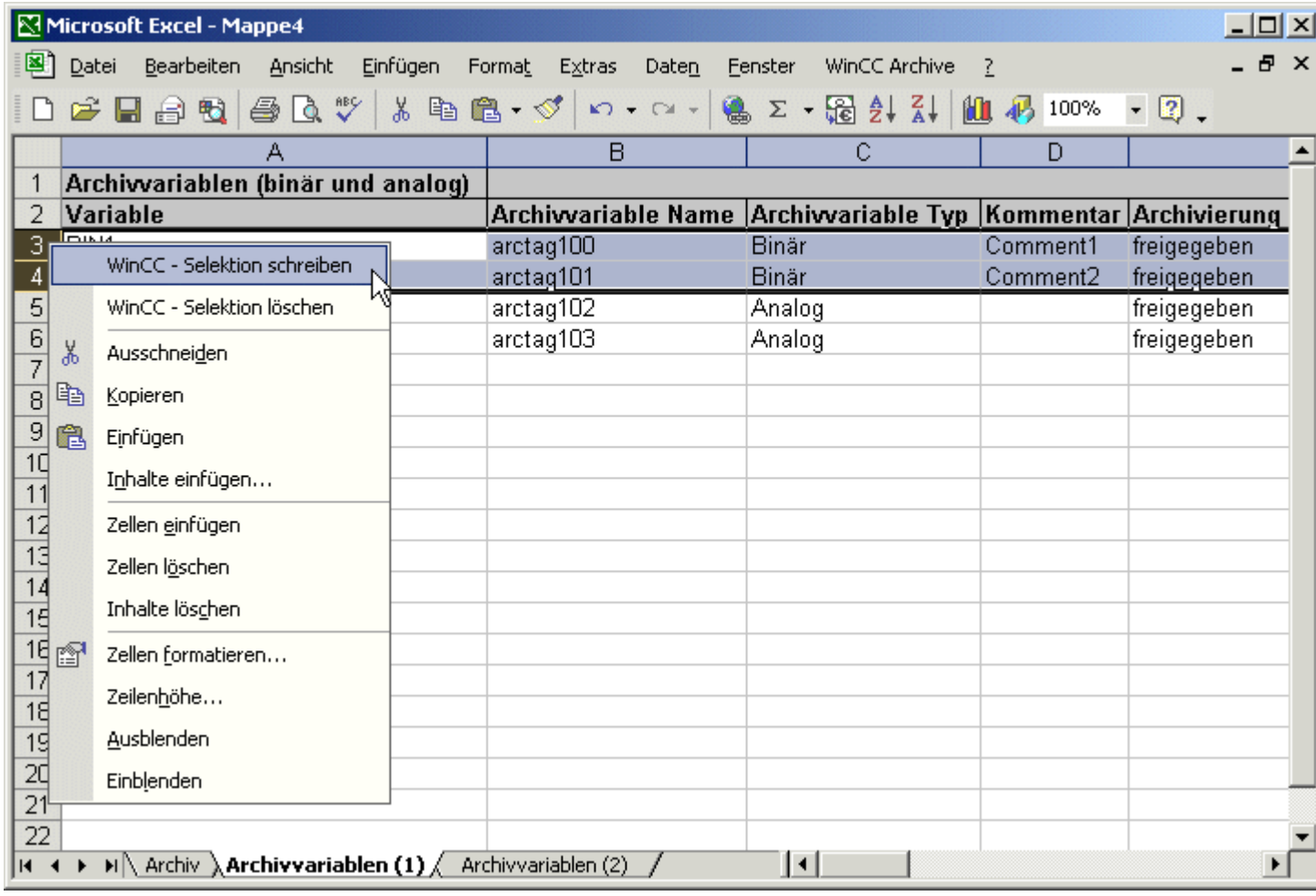

Wie kann ich mit WinCC im TIA Portal effizient arbeiten?

WinCC ermöglicht effizientes Verarbeiten von Prozessdaten in Tabelleneditoren. Dabei bietet WinCC folgende Funktionen an:

- Import und Export von Projektdaten

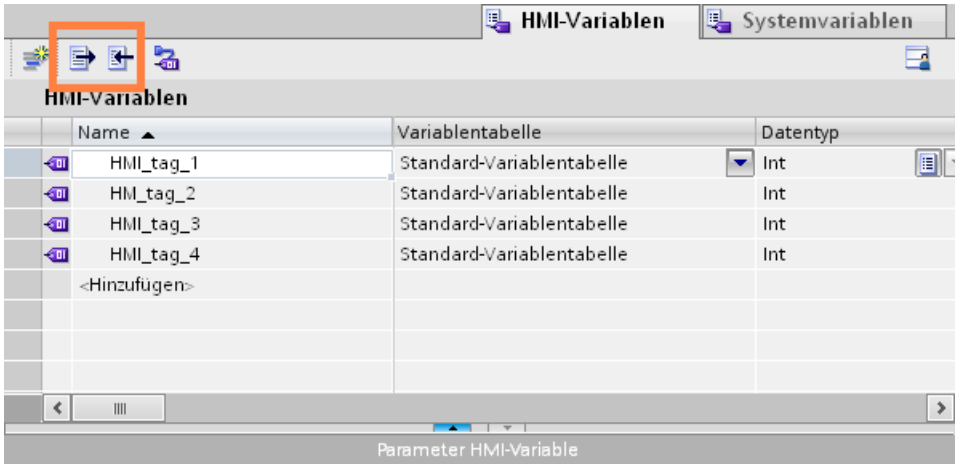

- Multiselektion
- Globales Ändern von einer Eigenschaft

Sie exportieren bzw. importieren folgende Projektdaten:

- Rezepturdatensätze
- Meldungen
- Variablen
- Textlisten
- Projekttexte für Übersetzungen

Durch den Export und Import reduzieren Sie den Arbeitsaufwand.

Anstatt neue Datensätze anzulegen, verwenden Sie Daten, die Sie bereits in früheren Projekten erstellt haben.

Nähere Hinweise hierzu finden Sie unter:

Datenaustausch zwischen Projekten und externen Anwendungen (Seite 2045)

Siehe auch

Datenaustausch zwischen Projekten und externen Anwendungen (Seite 2045)

18.2.8 Strukturen und Strukturvariablen

Strukturvariablen in WinCC V7

Strukturen werden verwendet, um eine größere Anzahl verschiedener Variablen und Variablentypen, die eine logische Einheit bilden, zusammenfassen zu können.

In WinCC V7 erstellen Sie dazu Strukturen, die Sie über ihren Namen bei der Variable ansprechen können.

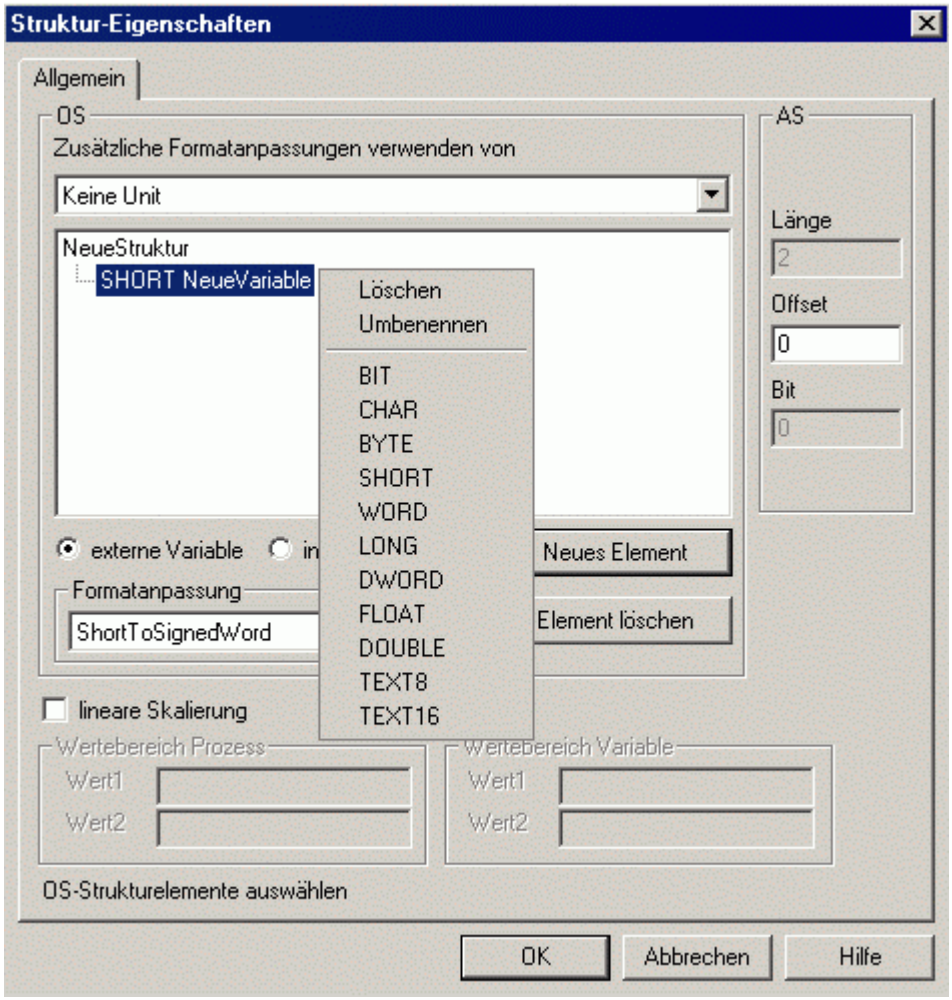

Wie projektiere ich Anwenderdatentypen mit WinCC im TIA Portal?

In WinCC fassen Sie eine Anzahl verschiedener Variablen, die eine logische Einheit bilden, über Anwenderdatentypen zusammen.

Anwenderdatentypen legen Sie als Typ an und verwenden Instanzen dieses Typs im Projekt.

Anwenderdatentypen sind projektzugehörige Daten und sind für alle HMI-Geräte des Projekts verfügbar.

Anwenderdatentypen und Anwenderdatentyp-Elemente legen Sie in der Projektbibliothek an. Einen Anwenderdatentyp müssen Sie freigeben, damit er im Projekt verfügbar ist.

Die projektierten Eigenschaften eines Anwenderdatentyps werden in den Instanzen des Anwenderdatentyps verwendet. Bei Bedarf ändern Sie einzelne Eigenschaften direkt an der Verwendungsstelle, z. B. an einer Variable. Die Änderung einer Eigenschaft an der Variablen wirkt sich nicht auf den erstellten Anwenderdatentyp aus.

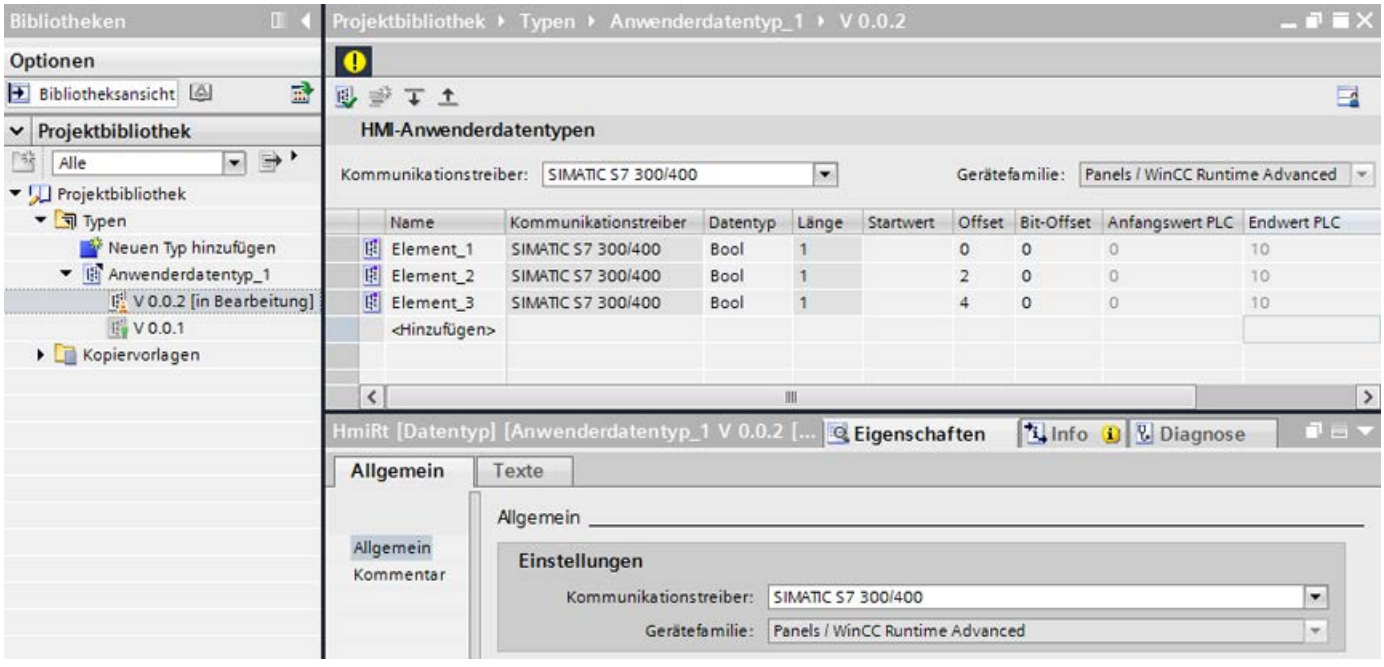

Prüfung der Variablen

Um festzustellen, ob eine Variable mit dem angegebenen Namen existiert, findet an der Verwendungsstelle eine Prüfung statt.

Wenn die Variable nicht existiert, wird die Verwendungsstelle markiert. Der Compiler gibt keine Warnung aus.

Wenn der Variablenname an der Verwendungsstelle mit "." beginnt, findet keine Prüfung im Engineering System statt.

In diesem Fall wird davon ausgegangen, dass für die Runtime Professional der Name aus einem Variablenprefix und dem String an der Verwendungsstelle gebildet wird und daher nicht im Engineering vorhanden ist.

Bei Anwenderdatentypen wird für das Variablenprefix typischerweise der Name der Instanz des Anwenderdatentypen genutzt. An den Verwendungsstellen im Bild nutzen Sie den Namen der jeweiligen Anwenderdatentypen-Elemente mit vorangestelltem Punkt.

Nähere Hinweise hierzu finden Sie unter:

Grundlagen zu Anwenderdatentypen (Seite 607)

Siehe auch

Grundlagen zu Anwenderdatentypen (Seite 607)

18.3 WinCC flexible

18.3.1 Bibliotheken

Bibliotheken in WinCC flexible

Bibliotheken sind Sammlungen von bereits konfigurierten Bildobjekten. Sie erweitern die Menge der verfügbaren Bildobjekte und steigern die Effektivität beim Projektieren, da Bibliotheksobjekte ohne erneute Konfigurierung immer wieder verwendet werden können.

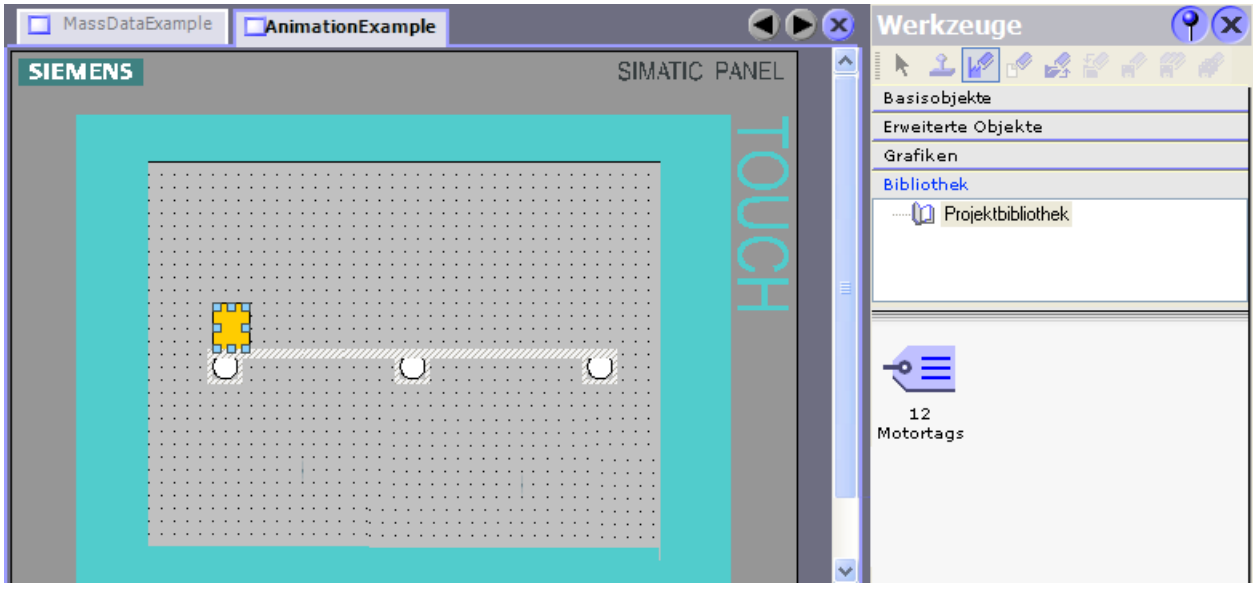

In WinCC flexible können Sie zwei Bibliothekstypen anlegen:

- Projektbibliothek
- Globale Bibliothek

Eine Bibliothek kann alle WinCC flexible Objekte, z. B. Bilder, Variablen, Grafikobjekte oder Meldungen enthalten.

Wie projektiere ich Bibliotheken mit WinCC im TIA Portal?

Mit WinCC projektieren Sie ebenfalls die "Projektbibliothek" und die "Globale Bibliothek".

Im Vergleich zu WinCC flexible können Sie in Bibliotheken keine Systemfunktionen ablegen.

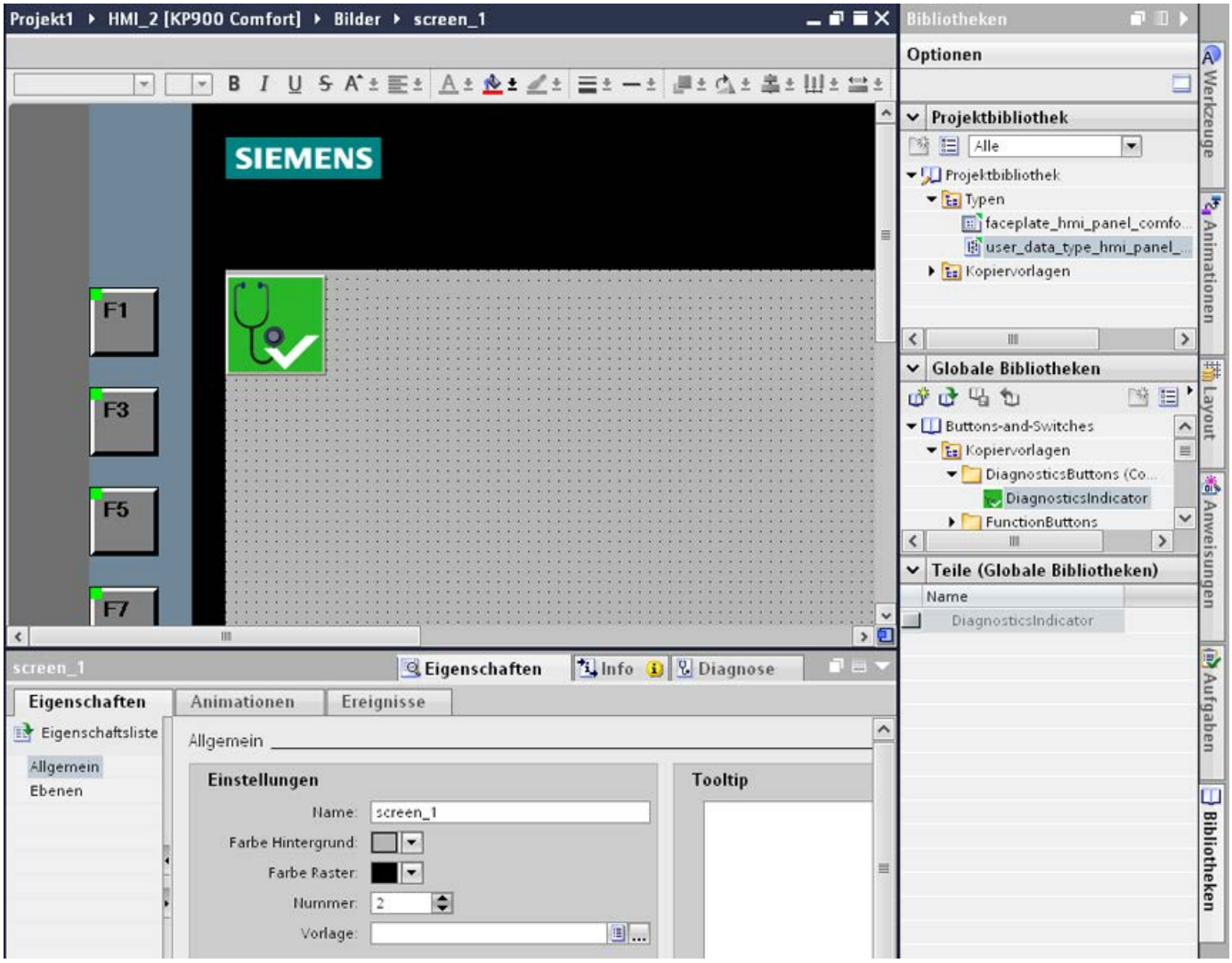

Sowohl die "Projektbibliothek" als auch die "Globale Bibliothek" enthalten die beiden Ordner "Kopiervorlagen" und "Typen". Sie können die Bibliotheksobjekte entweder als Kopiervorlage oder als Typ erzeugen bzw. verwenden.

- Kopiervorlagen
  Um voneinander unabhängige Kopien des Bibliotheksobjekts zu erzeugen, verwenden Sie Kopiervorlagen.
- Typen
  Von Objekten des Ordners "Typen" erstellen Sie Instanzen und verwenden die Instanzen in Ihrem Projekt. Die Instanzen sind an ihren jeweiligen Typ gebunden. Änderungen einer Instanz ändern auch alle anderen Instanzen. Typen werden in der Task Card "Bibliotheken" mit einem grünen Dreieck gekennzeichnet.
- Verwalten der Bibliotheksobjekte
  Sie kopieren und verschieben Bibliotheksobjekte nur innerhalb der gleichen Bibliothek.

Nähere Hinweise hierzu finden Sie unter:

Bibliotheken in WinCC (Seite 2021)

Siehe auch

Bibliotheken in WinCC (Seite 2021)

18.3.2 Bilder und Vorlagen

Bilder und Vorlagen in WinCC flexible

Mit WinCC flexible erstellen Sie Bilder zum Bedienen und Beobachten von Maschinen und Anlagen. Um Bilder zu erstellen, stehen Ihnen vordefinierte Objekte zur Verfügung, mit denen Sie Ihre Anlage nachbilden, Prozessabläufe anzeigen und Prozesswerte vorgeben.

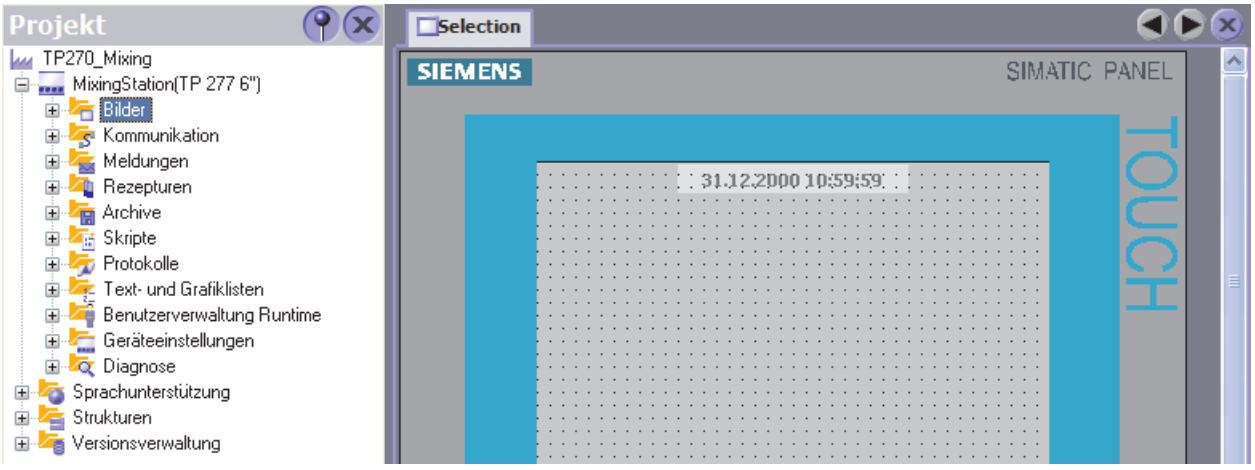

Für jedes Bediengerät existiert im Projekt eine Vorlage, in dem Sie Funktionstasten und Objekte für Ihr Projekt zentral projektieren können.

Jedes Bild, das auf der Vorlage basiert, enthält die in der Vorlage projektierten Funktionstasten und Objekte. Wenn Sie in der Vorlage ein Objekt oder die Belegung einer Funktionstaste ändern, ändert sich das Objekt in allen Bildern, die auf der Vorlage basieren.

Wie projektiere ich Bilder und Vorlagen mit WinCC im TIA Portal?

In WinCC projektieren Sie neben "Bildern" auch "Vorlagen" und ein "Globales Bild".

Innerhalb einer Vorlage legen Sie Funktionen und Objekte fest, die dann für alle Bilder auf der Grundlage dieser Vorlage gelten. Sie können in WinCC mehrere Vorlagen anlegen.

Für alle Bilder eines Bediengeräts legen Sie im "Globalen Bild" Elemente fest, die unabhängig von der verwendeten Vorlage sind. Als globale Objekte stehen die Objekte "Meldefenster", "Meldeindikator" zur Verfügung. Bei Bediengeräten mit Funktionstasten belegen Sie die Funktionstasten innerhalb des Editors "Globales Bild".

Für Comfort Panels haben Sie auch die Möglichkeit ein "System-Diagnosefenster" im Globalen Bild zu projektieren.

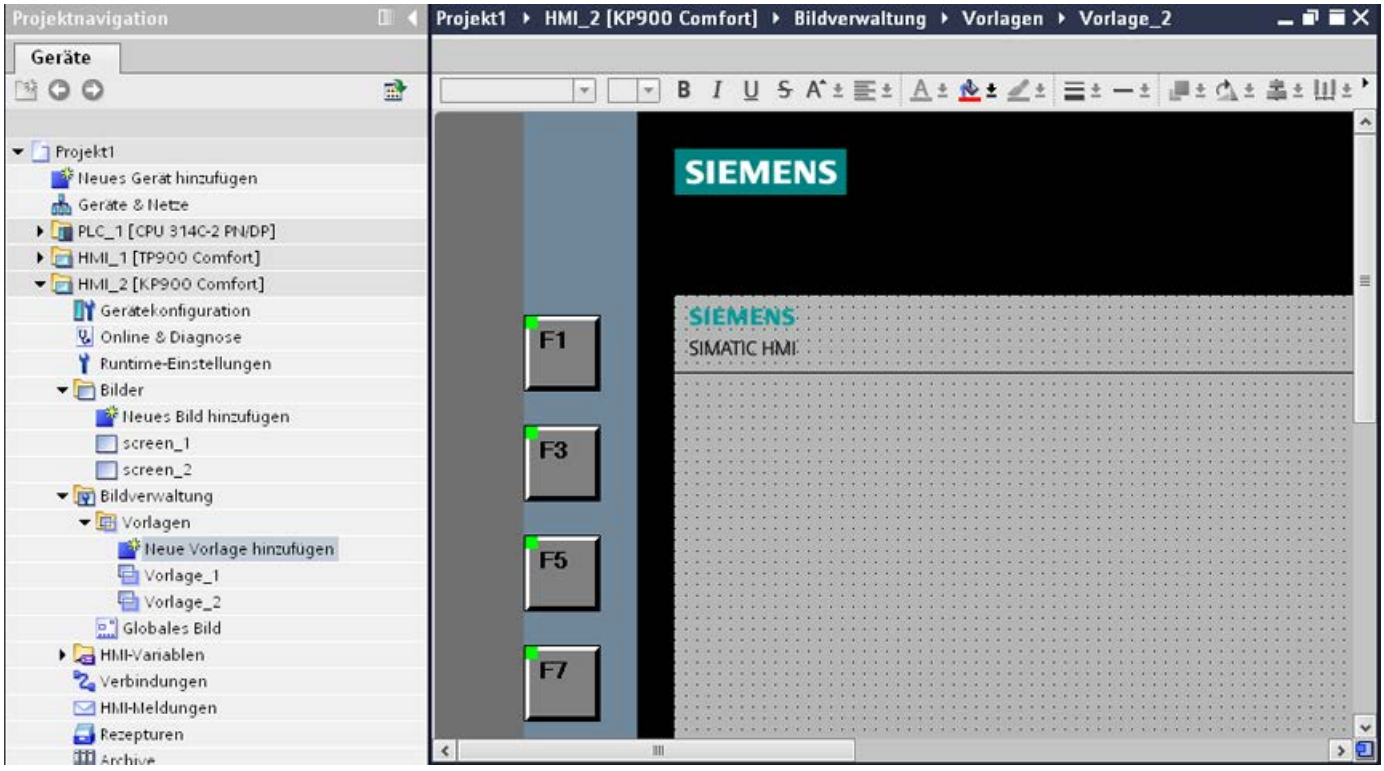

Bei der Anzeige der Bilder in Runtime gilt mit Ausnahme der Controls folgende Reihenfolge:

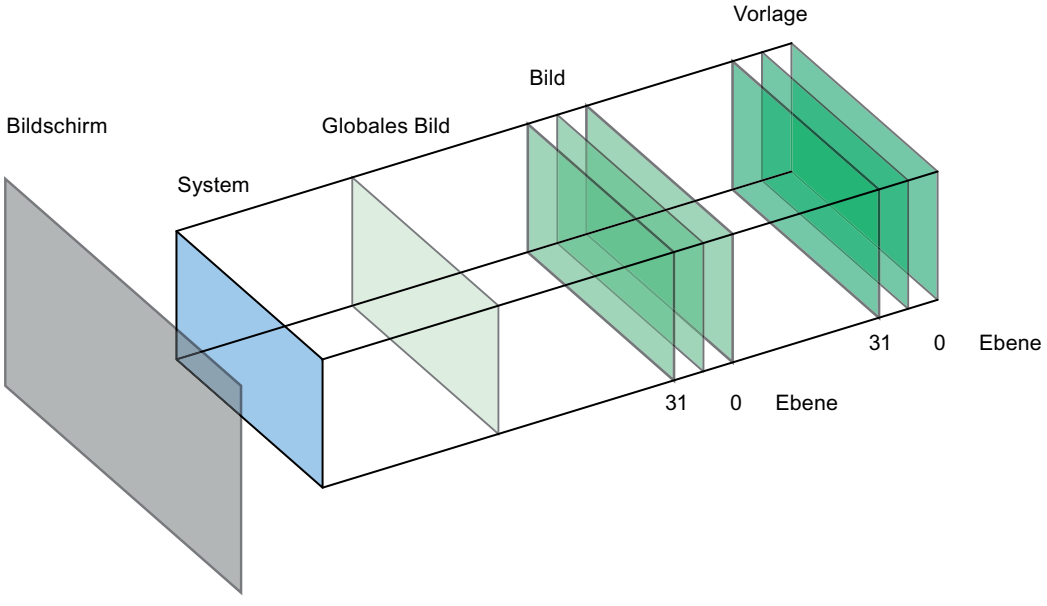

Nähere Hinweise hierzu finden Sie unter:

Grundlagen zu Bildern (Seite 35)

Siehe auch

Grundlagen zu Bildern (Seite 35)

18.3.3 Skripte in Bildbausteinen

Skripte in Bildbausteinen in WinCC flexible

In der Registerkarte "Skript" des Dialogs "Konfiguration Bildbaustein" projektieren Sie ein Skript. Dieses Skript ist nur innerhalb des Bildbausteins verfügbar.

Das Skript verschalten Sie direkt an die Ereignisse der im Bildbaustein enthaltenen Objekte, z. B. an das Ereignis "Klicken" einer Schaltfläche. Wenn Sie den Bildbaustein in einem Bild verwenden, dann wird eine Bildbaustein-Instanz erstellt.

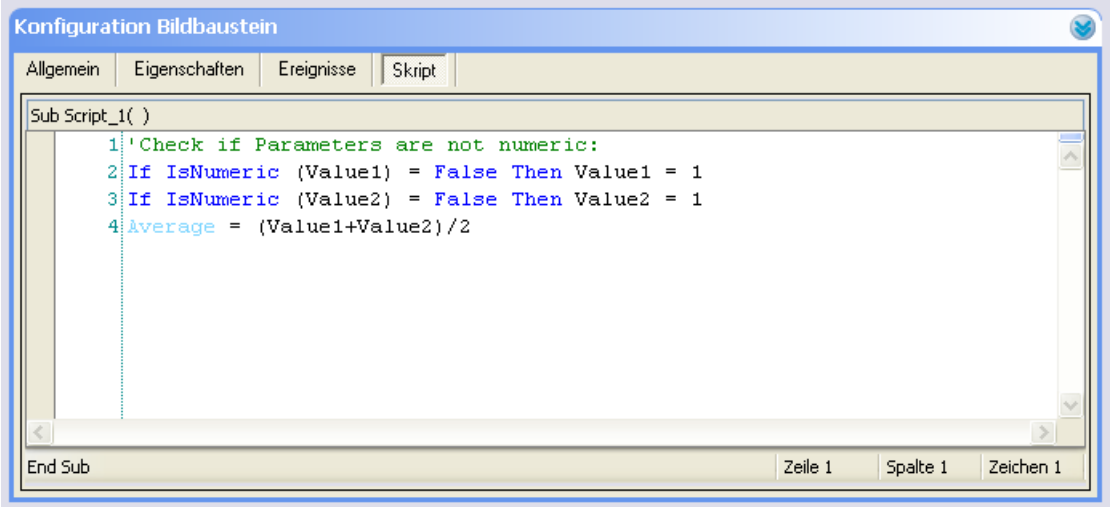

Wie projektiere ich Skripte in Bildbausteinen mit WinCC im TIA Portal?

Im Konfigurationsbereich des Editors "Bildbausteine" erstellen Sie Skripte, die Sie nur innerhalb eines Bildbaustein-Typs verwenden.

Im Gegensatz zu WinCC flexible, können Sie mit WinCC mehrere Skripte für einen Bildbaustein projektieren.

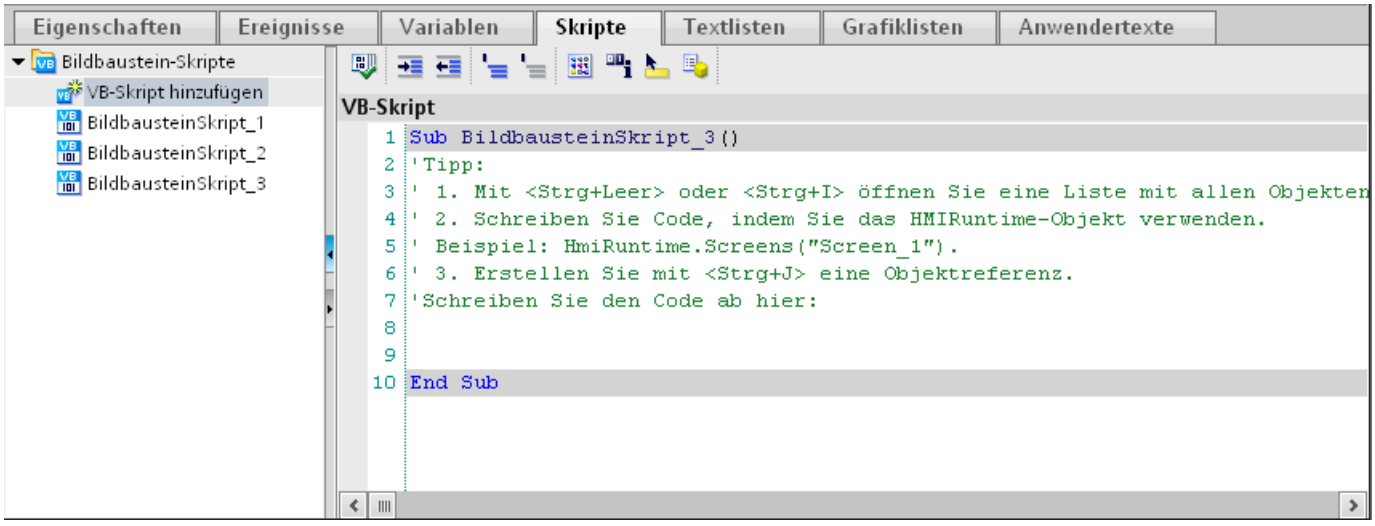

18.3.4 Synchronisierung von Rezepturen

Wie projektiere ich eine Synchronisation von Rezepturen mit WinCC im TIA Portal?

In WinCC projektieren Sie eine Synchronisation von Rezepturen im Editor "Rezepturen".

In WinCC haben sich im Gegensatz zu WinCC flexible einige Begriffe geändert.

Synchronisation bei Panels und RT Advanced

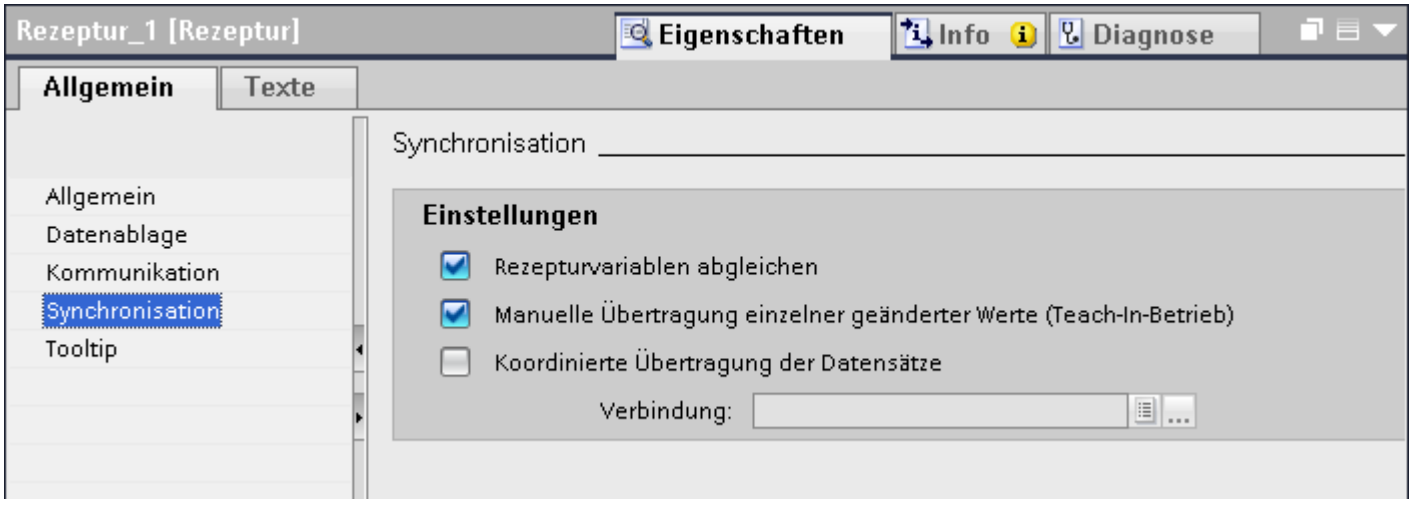

- Um in Runtime Rezepturvariablen, die in EA-Feldern projektiert sind, mit der Rezepturanzeige abzugleichen, aktivieren Sie "Rezepturanzeige und Rezepturvariablen abgleichen".
- Um festzulegen, dass die Rezepturvariablen automatisch beim Editieren in den EA-Feldern in die Steuerung übertragen werden, deaktivieren Sie "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
- Um in Runtime die Übertragung der Rezepturdaten über Bereichszeiger zu überwachen, aktivieren Sie "Koordinierte Übertragung der Datensätze".

Nähere Hinweise hierzu finden Sie unter:

AUTOHOTSPOT

18.3.5 Besonderheiten beim Umstieg

Projektsprachen in WinCC

In WinCC werden nicht alle Projektsprachen unterstützt, die in WinCC flexible einstellbar waren, z.B. Arabisch. Wenn Sie als Ergebnis der Migration ein leeres Projekt erhalten, überprüfen Sie bitte die verwendete Bearbeitungssprache. Die nicht unterstützten Projektsprachen dürfen nicht als Bearbeitungssprache im Quellprojekt eingestellt sein. Gehen Sie wie folgt vor:

1. Öffnen Sie das Projekt mit WinCC flexible.
2. Ändern Sie die Bearbeitungssprache in zum Beispiel Englisch.
3. Speichern Sie das Projekt.
4. Starten Sie die Migration erneut.

Objekte mit Objektreferenzen in der Projektbibliothek

In WinCC flexible konnten zwei Kopiermethoden verwendet werden.

- Beim "Einfachen Kopieren" wird z. B. ein WinCC flexible Bild inklusive eines EA-Felds kopiert. Von einer am EA-Feld projektierten Variable wird nur der Objektname kopiert, da es sich hierbei um eine Referenz handelt.
- Beim "Kopieren" werden ein Bild, ein enthaltenes EA-Feld und die am EA-Feld projektierte Variable mit ihren Eigenschaften kopiert.

Auch beim Ablegen eines Objektes in einer Bibliothek konnten diese beiden Methoden angewendet werden. Durch die Migration werden Projektbibliotheken und darin enthaltene Objekte migriert und sind in WinCC nutzbar.

In WinCC steht jedoch nur noch eine Kopiermethode zur Verfügung. Diese verhält sich bezüglich Variablen wie das "Einfache Kopieren" in WinCC flexible. Bezüglich Grafiken, Grafiklisten und Textlisten verhält sie sich wie das "Kopieren" in WinCC flexible.

Wenn Sie in einer Bibliothek in WinCC flexible Objekte mit Referenzen auf Variablen abgelegt haben, müssen Sie beim Verwenden dieser Objekte in WinCC die referenzierten Objekte neu projektieren.

Änderung der Namen von Meldeklassen

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

WinCC flexible	WinCC
Fehler	Errors
System	System
Warnungen	Warnings

Nach der Migration können die Anzeigenamen der Meldeklassen beliebig verändert werden.

Objekte mit Objektreferenzen in der Projektbibliothek

In WinCC flexible konnten zwei Kopiermethoden verwendet werden.

- Beim "Einfachen Kopieren" wird z. B. ein WinCC flexible Bild inklusive eines EA-Felds kopiert. Von einer am EA-Feld projektierten Variable wird nur der Objektname kopiert, da es sich hierbei um eine Referenz handelt.
- Beim "Kopieren" werden ein Bild, ein enthaltenes EA-Feld und die am EA-Feld projektierte Variable mit ihren Eigenschaften kopiert.

Auch beim Ablegen eines Objektes in einer Bibliothek konnten diese beiden Methoden angewendet werden. Durch die Migration werden Projektbibliotheken und darin enthaltene Objekte migriert und sind in WinCC nutzbar.

In WinCC steht jedoch nur noch eine Kopiermethode zur Verfügung. Diese verhält sich wie das "Einfache Kopieren" in WinCC flexible.

Wenn Sie in einer Bibliothek in WinCC flexible Objekte mit Referenzen auf andere Objekte abgelegt haben, müssen Sie beim Verwenden dieser Objekte in WinCC die referenzierten Objekte neu projektieren.

Index

*

*.bmp, 150
*.emf, 150
*.gif, 150
*.ico, 150
*.jpeg, 150
*.jpg, 150
*.svg, 150
*.tif, 150
*.wmf, 150

.

.Net-Control
    entfernen, 148
    hinzufügen, 147

A

abarbeiten
    Benutzerdefinierte Funktionen in Runtime, 1499
Abbrechen-Taste, 2239, 2320, 2409
Abfragen, 1093, 1099, 1103
    Anwenderarchiv, 1103
    Meldearchiv, 1099
    Prozesswertarchiv, 1093
abgebaut, 1799
Abgleich
    Redundanz, 2339
abgleichen
    Rezepturanzeige und Rezepturbild, 1126
    Rezepturdatensatz, 1162, 1176
    Rezepturvariable, 1131
Ablageort
    Archive mit Prüfsumme, 1076
    Datei - CSV (ASCII), 1075
    Datei - RDB, 1076
    Datei - TXT (Unicode), 1075
    Datenbank, 1076
abmelden,
    am Wirkbereich (RFID) projektieren, 1867
    am Wirkbereich projektieren, 1863
    Wirkbereich, 527
    Wirkbereich (RFID), 529
Abmeldezeit, 2202, 2250
    ändern, 1388
    ändern in Runtime, 1407
Absolute Adressierung
    einer Variable, 545
Abspielen beendet, 1799
Access Point, 1833, 1869
Achse
    gemeinsame, 726
    mehrere, 725
ACK
    Taste, 2240, 2321, 2410
AcknowledgeAlarm, 1690
ActivatePLCCodeView, 1673
ActivatePreviousScreen, 1571
ActivateScreen, 1569, 1744
ActivateScreenByNumber, 1570
ActivateScreenInScreenWindow, 1746
ActivateSystemDiagnosticsView, 1682
ActiveX-Control
    entfernen, 148
    hinzufügen, 147
adressieren
    multiplexen, 585
    Variable indirekt adressieren, 585, 586, 594
Adress-Multiplexen
    mit absoluten Adressen, 576
    mit symbolischen Adressen, 576
Aggregatfunktionen zur Analyse, 1110
AktiviereBild, 1569, 1744
AktiviereBildInAktuellemBildfenster, 1745
AktiviereBildInBildfenster, 1746
AktiviereBildMitNummer, 1570
Aktivieren,
    Autoscroll,
    Meldeblöcke für Meldeanzeige,
    Projektsprache,
    Protokollierung, 910, 911
    S7-Diagnosemeldung, 839
    Schriftgrößenanpassung, 81
AktivierePLCCodeAnzeige, 1673
AktivierePutzbild, 1717
AktiviereVorherigesBild, 1571
aktualisieren
    Bildbaustein-Typ, 298
    Runtime-Projekt auf V13 SP1,
    Variable, 544
AktualisiereVariable, 1572

Aktualisierung
Betriebssystem des Bediengeräts (Windows CE), 2434
Variable, 589
Aktualisierung der Firmware, 2433
Aktualisierung der Geräteversion, 2433
Aktualisierung des Betriebssystems, 2433
Aktualisierungszyklus, 616, 2338
Alarm Control
Notquittierung von Meldungen, 982
AlarmControl
Online-Projektierung, 1240
Runtime-Daten exportieren, 1239
AlarmHitView, 1110, 1113
ALARMVIEW, 1099
Allen Bradley Ethernet IP
Datentypen, 637
alphanumerische Tastenbelegung, 2240, 2321, 2410
ALT
Taste, 2240, 2321, 2410
Analoge Archivvariable erstellen, 683
Analogmeldeverfahren, 790, 791, 792
Analogmeldung, 790, 791, 792
Parameter ausgeben, 848
projektieren, 827, 843, 868
Analogmeldung importieren
Struktur für den Import, 2058, 2063
Analogmeldungen
projektieren, 816, 817
Analysefunktionen, 1110
Meldearchive, 1110, 1113
Prozesswertarchive, 1110
Recordset bei der Analyse von Meldearchiven, 1113
ÄndereKontrast, 1651
ändern
Abmeldezeit in Runtime, 1407
Benutzergruppe in Runtime, 1408
Benutzergruppen Anzeigename, 1390
Farben,
Kennwort, 1388
Name des Benutzers, 1407
Objekteigenschaft, 1504
Objektgröße, 109, 1326
Rezepturdatensatz in Runtime, 1162, 1175, 1176
Änderung, 1800
Anfangswert Gefahrenbereich
Zeigerinstrument, 533
Anfangswert Warnbereich
Zeigerinstrument, 533
Anfangswinkel
Ellipsenbogen, 378
Ellipsensegment, 380
Kreisbogen, 418
Kreissegment, 419
Animation, 214
Diagonale Bewegung, 223
Direkte Bewegung, 223
grüner Pfeil in Übersicht, 218
Horizontale Bewegung, 221
Mehrfachauswahl, 231
Objektgruppe, 229, 230
projektieren, 218, 1148
Übersicht, 218
Variablenanbindung, 227
Vertikale Bewegung, 223
Anlagenzustand, 896, 1069
Anlaufliste, 2338, 2349
einrichten, 2349
Anlaufsprache
festlegen:Anlaufsprache, 655, 933
anlegen, 1185
Array, 600, 605
Benutzer, 1386, 1427
Benutzer in Runtime, 1404
Benutzergruppe, 1383, 1384, 1426
Bild, 42, 43
externe Variable, 550
Globale Bibliothek, 2026
Hauptmenübefehl, 321
hierarchische Meldegruppen,
interne Variable, 552
Konfigurationsdatei, 327
Meldearchiv, 924
Meldegruppe,
Menübefehl, 322
Ordner im Editor HMI Dateien, 2193
Pop-up-Bild, 58
projektweite Meldeklasse,
Protokoll,
Protokoll (Übersicht),
Rezeptur, 1185, 1222
Rezepturabfrage, 1230
Schaltfläche, 324
Symbolleiste, 323
Untermenübefehl, 322
Variablenarchiv, 643
Vorlage, 52
Zyklus, 617, 621
Anmeldedialog
Zugriffsschutz projektieren, 1420

anmelden,
am Wirkbereich (RFID) projektieren, 1867
am Wirkbereich projektieren, 1863
Anmeldung nicht erfolgreich, 1410, 1411
Benutzer, 1408, 1410
dynamisch, 1381
protokollieren, 1421
Wirkbereich, 526
Wirkbereich (RFID), 528
anordnen
Objekt im Bild, 103, 115, 1325
Anweisungen
Funktionsliste, 1446
PLC-Code-Anzeige, 1961, 1974
Anwenderarchiv, 1103, 1109
Archivdaten abfragen, 1103
Daten darstellen, 1109
Rezepturanzeige, 1268
UserArchiveControl, 1266
Anwenderdatentyp, 607
anlegen, 607, 609
Anwenderdatentyp-Element anlegen, 609
bearbeiten, 611, 612
einem Bild zuweisen, 142
freigeben, 612
löschen, 613
Variablen, 607
Anwenderdatentyp-Element
anlegen, 609
Anwenderdatentypen
mit Bildbausteinen verschalten, 284
Anwenderdatentypen (UDTs)
in Bildbausteinen verwenden, 282
Anwendermeldung, 796
Parameter ausgeben, 855
Anwendermeldung triggern
Beispiel, 856
Anwendertextblock
Projektieren, 841
Anwendertextblöcke, 840
Anwendungsbeispiel
Rezepturdaten offline eingeben, 1205
Anwendungsfall, 1089, 1090, 1091, 1092
Remote-Zugriff auf WinCC RT-Datenbank, 1089
Remote-Zugriff auf WinCC-Anwenderarchive, 1092
Remote-Zugriff auf WinCC-Archivdatenbanken, 1090
Zugriff auf lokale WinCC-Anwenderarchive, 1091
Anwenungsbeispiel
Rezeptur mit manuellen Produktionsablauf, 1206
Anzahl Wiedergaben
Media Player, 433
Anzahl Zeilen
Benutzeranzeige, 351
Anzeige, 896
Anzeigebereich festlegen
Meldeanzeige, 444
Anzeigeformat
E/A-Feld, 371, 373
Anzeigeklassen, 797
anzeigen
Archivinhalt,
Backup HMI, 2300
Grenzwertlinien beim Balken, 348, 350
Hilfetext, 2240, 2244, 2321, 2326, 2410, 2415
Anzeigeobjekt
Verfügbarkeit, 344
Verfügbarkeit für Basic Panel, 337
Verfügbarkeit für Comfort Panel, 338
Verfügbarkeit für Mobile Panel, 340
Verfügbarkeit für WinCC Runtime Advanced, 343
Anzeigeunterdrückung
Dauer, 2338
Dauer festlegen, 900
Anzeigeunterdrückungsdauer
festlegen, 900
Anzeigeunterdrückungsmaske, 896, 1069
Anzeigeunterdrückungsvariable, 896, 1069
einer anwenderdefinierten Gruppenmeldung, 1069
Arbeitsbereich
Mobile Wireless, 1843
Mobile Wireless (RFID), 1846
Wirkbereich, 1843
Wirkbereich (RFID), 1846
Zone, 1840, 1842
Arbeitsschritt
zum Erstellen von Bildern, 41, 42
Archiv
Automatische Einträge, 1073
füllstandsabhängig steuern, 645, 935
Größe konfigurieren, 688
Meldearchiv, 1071, 1077
Projektierung, 677
Prüfsumme, 653, 931, 1076
Runtime-Sprache, 655, 933, 2120
Sprachumschaltung, 655, 933, 2120
trennen, 698
Variablenarchiv, 1071, 1077
Verbinden, 697
Verdichtungsarchiv, 1077
verwendbare Zeichen im Variablennamen, 679

Archiv mit füllstandsabhängigem Ausführen von Systemfunktionen, 922
Archiv mit füllstandsabhängigem Systemalarm, 922
Archiv-Backup
Trennen, 698
Verbinden, 697, 946
Archivdaten, 975, 1084, 1099, 1103, 1108
Abfrage, 1099, 1103, 1108
als Reports im Netz, 1106
im Meldefenster ausgeben, 975
über WinCC OLE DB-Provider zugreifen, 1084
Archivdatenbank, 672, 1085
Direktzugriff mit ODBC, 714, 928
Verbinden, 1085
Archive Connector, 1117
Archiveinträge, 1073
ArchiveLogFile, 1603
archivieren
innerhalb/außerhalb von Grenzwerten, 647
Toleranzband, 647
Variablen, 641, 647
ArchiviereProtokolldatei, 1603
ArchiviereVariable, 1605
Archivierung
Archivtyp, 922
Archivvariante, 640
auf Anforderung, 670
bei Wertänderung, 670
segmentiertes Umlaufarchiv, 640
Start, 658
Umlaufarchiv, 640
Variablenwert, 639
Zyklische, 668
Zyklisch-selektive, 669
Archivierungsart, 658
Archivierungsarten, 665
auf Anforderung, 665
bei Änderung, 665
Verdichtung, 665
zyklisch, 665
zyklisch-selektiv, 665
Archivierungsmethode
Füllstandsabhängigkeit, 640
segmentiertes Umlaufarchiv, 640
Umlaufarchiv, 640
Archivierungssprache, 655, 933, 2120, 2201, 2249
Archivierungszyklus, 647, 658, 665
Variable, 616, 619
Archivinhalt
anzeigen, 640, 922
Archivsegmente, 2338
projektieren, 688
Archivtyp, 922
Archivtypen, 695
Archivvariable, 660, 681
analoge, 660
binäre, 660
Erstellung, 681
Projekierung, 681
projektieren, 656, 700
Archivverhalten
bei Systemstart steuern, 644, 934
füllstandsabhängig steuern, 645, 935
mit Systemfunktion steuern, 649
Array, 597, 600, 603, 605
anlegen, 600, 605
indirekt adressieren, 585
Arrayvariable, 597, 603
Asiatische Schriftarten
auf das Bediengerät laden, 2441
asiatische Sprachen
Projektierung, 2122
Schriftgröße, 2122
Textfeldlänge, 2122
asiatische Zeichen
Eingabe am Bediengerät, 2122
Interpretation, 2122
Speicherbedarf, 2122
asiatisches Betriebssystem, 2098
Assistent
Geräte-Assistent, 2011, 2014
Audit Trail
Prüfsumme, 653, 931, 1076
Auf Anforderung, 589
Aufbau Bediengerät, 2187
Aufgabe, 1872
deaktivieren, 1874, 1875
löschen, 1888
Name ändern, 1888
zum automatischen Start eines Druckauftrags anlegen, 1315
zur Laufzeit ändern, 1896
Aufgabenplaner, 1871
Arbeitsbereich, 1872
Aufgabe verwalten, 1888
azyklischer Trigger, 1874, 1883
deaktivierte Aufgabe, 1874
Ereignis-Trigger, 1874, 1875, 1886, 1895
Funktionsliste, 1877
Protokoll drucken, 1896
Timer bei zyklischen und azyklischen Triggern, 1876
Trigger, 1874

Variablentrigger, 1876
zyklischer Trigger, 1875, 1893, 1896, 1898
aufgebaut, 1800
aufheben
Gruppe, 163
Verbindung zum Bildbaustein-Typ, 302
auflösen
Gruppe, 163
Auftragstypen und Fehlercodes
Rohdatenkommunikation bei Rezepturen, 1247
Ausblenden
Abschnitt im Protokoll, 1307
Ausdruck
Dynamisierung, 233
Eigenschaftsanimation, 233
Ausfallszeitraum
Redundanz, 2339
Ausführen, 1801
Ausgabe der Meldung, 436
Ausgabe von Meldungen
projektieren:Ausgabe von Meldungen, 869
Auslagern, 942
Meldearchiv, 942
Signierung Meldearchiv, 942
Auslagern von Prozesswerten, 674
Auslagerung, 674
von Prozesswerten, 674
ausrichten
Objekt bündig, 113, 1330
Auswahl geändert, 1801
auswählen
mehrere Objekte, 103, 111, 1327
Auswahlliste
öffnen, 2239, 2320, 2409
Automatisch
protokollieren, 829
Automatisches Ausblenden, 1069
Automatisches Unterdrücken der Anzeige, 896
Automatisches Update der Steuerungsmeldungen
konfigurieren, 991
Autoscroll
aktivieren, 439
Autostart, 2357, 2360
Autovervollständigung, 1444
azyklischer Trigger, 1883, 1891

B

Backspace-Taste, 2239, 2320, 2409
Backup, 942, 2281, 2436
des Meldearchivs konfigurieren, 942
löschen, 2302
umbennen, 2302
BackupRAMFileSystem, 1587
Balken, 347, 349
Farbverlauf, 347, 349
Grenzwertlinien anzeigen, 348, 350
Skalenteilung festlegen, 350
Balkensegment
definieren, 348, 350
Barcode, 2246, 2334
Barcode scnanner, 2246, 2333
Basic Panel
Anzeige- und Bedienobjekt, 337
Runtime-Start, 2226
Basic Panels
verfügbare Systemfunktionen, 1536, 1541
BearbeiteMeldung, 1686
bearbeiten, (Variablen)
Formatvorlage, 87
Funktionsliste, 1442
Gruppe, 161
Objekt innerhalb einer Gruppe, 166
Ordnerverknüpfung, 151
Rezeptur,
Systemmeldung,
Bearbeitungsmethode, 667
Variablenarchiv, 665
Bedienbarkeit
dynamisieren, 224
Bedienelement
Beobachtungstabelle, 356
einfache Meldeanzeige, 959
f(t)-Kurvenanzeige, 382
f(x)-Kurvenanzeige, 385, 389
gesperrtes, 2315, 2405
HTML-Browser, 404
Kurvenanzeige, 423
Media Player, 432
Meldeanzeige, 436, 446, 449, 453, 956
Meldefenster, 956
PDF-Anzeige, 460
Rezepturanzeige, 477, 485, 1166
sperren/entsperren, 2316, 2406
Tabellenanzeige, 517
Wertetabelle, 524
Bedienelemente
Meldeanzeige, 442
Rezepturanzeige, 480
bedienen
Bedienobjekt, 2239, 2320, 2409
Taste, 2238, 2319, 2408

Bediengerät
auf Werkseinstellungen zurücksetzen, 2434
Aufbau, 2187
Betriebssystem aktualisieren (Windows CE), 2434
Betriebssystem neu laden, 2433
Daten sichern, 2431
Daten wiederherstellen, 2429, 2432
Datensicherung, 2426, 2427, 2428, 2429
Gerätetyp ändern, 2230, 2294
Image, 2426, 2427, 2428
Laden, 2277, 2426, 2427, 2428
Leistungsmerkmale, 2455, 2458, 2462, 2468, 2473, 2478
License Key übertragen, 2438
Software, 2426, 2427, 2428
Stil, 2201, 2249
Systemgrenzen, 2455, 2458, 2462, 2468
tauschen, 2137
WinCC Runtime Professional Performance, 2482
Bediengerätabhängigkeit
Archivierung, 1073
Bediengeräte mit HMI-Verbindungen
kopieren, 2146
Bediengeräteabhängigkeit
in Systenfunktion, 1448
Bediengeräte-Assistent, 2011, 2014
Bediengeräte-Image, 2307
löschen, 2304
Bediengerätesignatur, 2247, 2335
Bediengerätetyp
ändern, 2230, 2294
Bediengeräteversion, 2137, 2187, 2188, (Geräteversion)
Bedienobjekt
bedienen, 2239, 2320, 2409
Rezepturanzeige, 1155
Verfügbarkeit, 344
Verfügbarkeit für Basic Panel, 337
Verfügbarkeit für Comfort Panel, 338
Verfügbarkeit für Mobile Panel, 340
Verfügbarkeit für WinCC Runtime Advanced, 343
Bedienung, (Zweihand)
einfache Meldeanzeige, 959
einfache Rezepturanzeige, 1143
Meldeanzeige, 955
Meldefenster, 955
Meldeindikator, 961
Rezepturanzeige, 1155, 1166
Rückmeldung, 2399
Bedienung mit der Maus
einfache Meldeanzeige, 960
Bedienung mit der Tastatur
einfache Meldeanzeige, 960
Bedienung mit Maus
Meldeanzeige, 957, 969
Meldefenster, 957, 969
Meldeindikator, 962
Rezepturanzeige, 1159, 1171
Bedienung mit Tastatur
Meldeanzeige, 957, 969
Meldefenster, 957, 969
PLC-Code-Anzeige, 465
Rezepturanzeige, 1159, 1171
Bedienungsrückmeldung, 2399
Bei Überschreitung, 1801
Bei Unterschreitung, 1801
Beim Öffnen eines Dialogs, 1802
Beim Schließen eines Dialogs, 1802
Beispiel
Anwendermeldung triggern, 856
Anwendung Meldeklassen, 800, 801
Auf Objekte zugreifen (C), 1531
Auf Objekte zugreifen (VBS), 1513
Benutzerdefinierte Funktion für Temperaturumrechnung, 1505
Betriebsart wechseln und Betriebsart anzeigen, 1509
Farbe von Objekten bestimmen (C), 1531
Meldeparameter einer Anwendermeldung definieren, 857
Skript am Server starten (VBS), 1528
Sprachumschaltung projektieren (C), 1532
Systemmeldung, 796
Beispiel C-Funktion in WinCC
Bildwechsel über Property projektieren, 1533
Diagnoseausgabe über Trace projektieren, 1533
Objekteigenschaften schreiben, 1535
Runtime deaktivieren, 1532
Variablenwerte lesen, 1534
Variablenwerte schreiben, 1534
Beispiel/
Bitmeldung, 794
Systemmeldung, 797
Beispiele
C in WinCC, 1530
Diagnoseausgabe über Trace projektieren (VBS), 1515
Farbe von Objekten bestimmen (VBS), 1514
Meldeanzeige projektieren (VBS), 1525
Objekteigenschaften schreiben (VBS), 1522
Rezepturanzeige projektieren (VBS), 1524
Runtime deaktivieren (VBS), 1515
Sprachumschaltung projektieren (VBS), 1514

Variablenwerte auswerten (VBS), 1527
Variablenwerte lesen (VBS), 1519
Variablenwerte schreiben (VBS), 1517
VBS in WinCC, 1513
Beispielef
Bildwechsel über Property projektieren (VBS), 1515
Belegen
eine Funktionstaste, 248, 250, 252
Benutzedefinierte Funktion
Objekteigenschaft ändern, 1504
Benutzer
ändern, 1407
anlegen, 1386, 1427
anlegen in Runtime, 1404
anmelden, 1408, 1410
Anmeldung protokollieren, 1421
bei Benutzerwechsel aktualisieren, 1894
Benutzergruppe zuordnen, 1387
dynamisch anmelden, 1391
entsperren, 1408
löschen, 1389
löschen in Runtime, 1407
Name ändern, 1388
verwalten, 1387, 1406
Benutzeranzeige, 1402
Anzahl Zeilen, 351, 354
Darstellung, 351
einfache Benutzeranzeige, 354, 448
Komplexe Benutzeranzeige, 1402
projektieren, 1404
Spalten verschiebbar, 355
Spalten verschieben, 351
Spaltenbreite, 352
Verwendung, 350
Benutzerdaten
sichern, 1369
wiederherstellen, 1369
benutzerdefiniert
Menü, 316
Symbolleiste, 316
Benutzerdefinierte C-Funktion, 1431
Gültigkeitsbereich, 1466
umbenennen, 1473
Benutzerdefinierte Funktion, 1431, 1435
abarbeiten in Runtime, 1499
Anzeigeeigenschaften anpassen, 1447
Autovervollständigung, 1444
Code-Vorlagen, 1446
know-how-geschützte Funktion öffnen, 1477
Objekt synchronsieren, 1446
Parameter eingeben, 1444
Parameterübergabe, 1449
projektieren in C, 1469
projektieren mit VBS, 1458
Reihenfolge wird nicht eingehalten, 1502
Rückgabewert in C, 1475
Rückgabewert in VBS, 1462
Syntaxhervorhebung, 1446
Systemfunktion verwenden, 1500
Temperaturumrechnung, 1505
testen, 1453
Variable synchronsieren, 1446
Benutzerdefinierte Meldegruppen, 809
Benutzerdefinierte Meldeklassen, 800, 802
Benutzerdefinierte VB-Funktion, 1431
umbenennen, 1460
Benutzergruppe
ändern in Runtime, 1408
anlegen, 1384, 1426
Anzeigename ändern, 1390
Benutzer zuordnen, 1387
Berechtigung verwalten, 1389
löschen, 1390
Name ändern, 1389
Unauthorized, 1410
verwalten, 1389
zuordnen, 1427
Benutzerrechte
Hierarchieebenen, 2202, 2250
Benutzerverwaltung, 1367, 2202, 2250, 2338
aufbauen, 1422
exportieren, 1412
importieren, 1413
Objekt mit Zugriffsschutz, 1416, 1417
Runtime-Einstellungen, 46, 48, 1374, 1377, 1381, 1391, 1392, 1397, 1398, 1408, 1409
SIMATIC Logon, 1393
zentrale Benutzerverwaltung, 1393
Benutzerwechsel, 1803, 1894
Beobachtungstabelle, 355
Bedienelement, 356
Reihenfolge der Spalten, 357
sichtbare Spalte, 356
StatusSteuernLeseWerte, 357
StatusSteuernSchreibeWerte, 357
Berechtigung
erstellen, 1383, 1423
löschen, 1390
Name ändern, 1390
projektieren, 1414
verwalten, 1389
zuweisen, 1385, 1426

Bereich (... - ...)
Grafikliste, 200, 202
Textliste, 186
Bereichszeiger
Datensatz, 1135
Beschriftung
Rezepturanzeige, 481
Bestandteile
Meldung, 814
Betriebsart wechseln und Betriebsart anzeigen (Beispiel), 1509
Betriebsmodus
GRAPH-Übersicht, 397
Betriebssystem
asiatische Spracheinstellung, 2097
Einstellung auf westlichen, 2097
Bibliothek, 2016
Bibliotheksobjekt kopieren, 2020
Buttons and Switches, 2021
Formatvorlage, 2043
Kopiervorlage, 2020
Objekt hinterlegen, 2038
öffnen, 2028
speichern, 2027
Stil, 2043
System-Diagnoseindikator, 1909, 1927
Typ, 2020
Bibliothek aktualisieren, 2031
Bibliotheksansicht
Übersicht, 2018
Bibliotheksobjekt, 2016, 2017
einfügen, 2039
Bibliotheksverwaltung
filtern, 2025
öffnen, 2024
Übersicht, 2023
Bild
anlegen, 42, 43
Arbeitsschritt zum Erstellen, 41, 42
benutzerdefinierte Symbolleiste festlegen, 329
benutzerdefiniertes Menü festlegen, 329
Bibliothek, 2042
einfügen, 44
Geräteabhängigkeit, 37
kopieren, 44, 2147
löschen, 44
Schriftart, 37
Typ erzeugen, 2042
umbenennen, 44
verschieben, 44
Vorlage verwenden, 56
zoomen, 40
Bildauswahl
einrichten, 2341
Bildbaustein
Beispiel, 305
Bildbaustein-Typ, 273
dynamisieren, 303
Eigenschaften auf Werte des Bildbaustein-Typs zurücksetzen, 273
Eigenschaften dynamisieren, 304
Eigenschaften zurücksetzen, 273
enthaltene Objekte dynamisieren, 311
erzeugen, 305
Instanz des Bildbausteins, 273
Verbindung zum Bildbaustein-Typ aufheben, 302
Verhalten beim Anpassen der Größe, 299, 300
Bildbaustein-Typ, 273
aktualisieren, 298
Anwendertext, 278
bearbeiten, 298
Bildbaustein, 301
dynamisieren, 303
Eigenschaft löschen, 286
Eigenschaft Verbindung löschen, 283
Eigenschaften, 277
Eigenschaften festlegen, 306
Ereignis, 277
erzeugen, 278
freigeben, 298
Grafikliste, 278
Kategorie löschen, 286
Skript, 278
Textliste, 278
Variable, 277
Variable anlegen, 307
Verbindung aufheben, 302
Verbindung löschen, 283
Verhalten beim Anpassen, 299, 300
verwenden, 301
Bilder
Design anwenden, 70
Designs, 66
Bildfenster
Menü projektieren, 330
Symbolleiste projektieren, 330
unabhängig, 2337
Bildobjekt
Bedienbarkeit dynamisieren, 224
Bewegung dynamisieren, 221
BildobjektCursorAbwärts, 1721
BildobjektCursorAufwärts, 1721
BildobjektCursorLinks, 1723
BildobjektCursorRechts, 1722

BildobjektSeiteAbwärts, 1722
BildobjektSeiteAufwärts, 1723
Bildpuffer, 2337
Bildschirmauflösung, 2201, 2249
Bildschirmtastatur, 2202, 2250, 2338
    Basic Panels, 2235
    Basic Panels 2nd Generation, 2235
    RT Advanced, 2311
Bildvorlage, 250
Bildwechsel, 1803
    Bild bei Zoneneintritt anzeigen, 1853, 1855, 1859
Bildzyklus, 619
Binäre Archivvariable erstellen, 682
Bit (0, 1)
    Grafikliste, 203
    Textliste, 188
Bitmeldeverfahren, 790, 791, 792, 793
Bitmeldung, 790, 791, 792, 793, 794
    Parameter ausgeben, 852
    projektieren, 830, 866
    Struktur für den Import, 2067
Bitmeldung importieren
    Struktur für den Import, 2061, 2067
Bitmeldungen
    projektieren, 816, 817
Bitnummer (0 - 31)
    Grafikliste, 205
    Textliste, 190
Bitselektion
    Gestaltungsanalyse, 2201, 2249, 2337
    Text- und Grafiklisten, 2201, 2249
Blinken, 843
    Meldeblock, 843
Box-ID, 1841
Buttons and Switches
    Bibliothek, 2021

C

C
    Beispiele in WinCC, 1530
CalibrateTouchScreen, 1652
C-Funktion
    Beispiel, 1532, 1533, 1534, 1535
ChangeConnection, 1648
ChangeConnectionEIP, 1649
ClearAlarmBuffer, 1686
ClearAlarmBufferProtoolLegacy, 1687
ClearLog, 1607
Client-Umschaltung
    Redundanz, 2339
CloseAllLogs, 1609
Codepage, 1468
    Runtime-Einstellungen, 1469
Code-Vorlagen
    in benutzerdefinierten Funktionen einfügen, 1446
Codieren, 1572
CodierenEx, 1573
Comfort Panel
    Anzeige- und Bedienobjekt, 338
    Bediengeräte-Image löschen, 2304
Comfort Panels
    verfügbare Systemfunktionen, 1547
CommandText, 1108
ConnectionString, 1085
Connectivity Pack, 1117
    Archive Connector, 1117
    MS OLE DB-Provider, 1117
    OPC, 1117
    WinCC OLE DB-Provider, 1117
Control Daten
    Darstellung, 1337
    Verwendung, 1337
Control Hardcopy
    Darstellung, 1338
    Verwendung, 1338
Control Informationen
    Darstellung, 1340
    Verwendung, 1339
Controls
    Kamera-Anzeige, 409
    PDF-Anzeige, 457
ControlSmartServer, 1592
ControlWebServer, 1593
cp_TagStatistic, 1110
csv-Datei,
    Aufbau, 713, 996
    Aufbau für Ausgabe einer Tabelle,
    Aufbau für Ausgabe von Kurven,
    Beispiel, 713, 996
CSV-Provider Kurven
    Aufbau CSV-Datei, 1301
    Darstellung in Protokollen, 1341
    Verwendung in Protokollen, 1340
CSV-Provider Tabelle
    Aufbau CSV-Datei, 1303
    Verwendung im Protokoll, 1341
C-Textliste
    Eintrag erstellen, 209
    Projektierungsschritte, 208
    Verwendung, 208
CTRL
    Taste, 2240, 2321, 2410

Cursormodus
festlegen, 170
Cursorsteuerung, 2338, 2346
Cursor-Taste, 2238, 2319, 2408

D

Darstellung
Balken, 347, 349
Benutzeranzeige, 351, 354
Beobachtungstabelle, 356
Control Daten, 1337
Control Hardcopy, 1338
Control Information, 1340
CSV-Provider Kurven, 1341
Datum/Uhrzeit-Feld, 363
Datum/Uhrzeit-Feld in Protokollen, 1342
Datum-Uhrzeit-Feld in Protokollen, 1343
Druckauftrag/Skriptdiagnose, 365
E/A-Feld, 368
E/A-Feld in Protokollen, 1345
Editierbares Textfeld, 376
Einfache Benutzeranzeige, 353
einfache Rezepturanzeige, 1143
Ellipse, 377
Ellipsenbogen, 378
Ellipsensegment, 379
f(t)-Kurvenanzeige, 381
f(x)-Kurvenanzeige, 384, 387
Grafikanzeige, 391, 1348
Grafisches E/A-Feld, 393, 395
grafisches E/A-Feld in Protokollen, 1349
GRAPH-Übersicht, 397
Handrad, 400
Hardcopy in Protokollen, 1350
HTML-Browser, 403, 407
Kamera-Anzeige, 409
Kreis, 416
Kurvenanzeige, 422
Ladezustand, 424
Leuchtdrucktaster, 426
Linie, 429
Listenfeld, 430
Media Player, 432
Meldeanzeige, 435, 445, 448, 453, 955, 959
Meldeindikator, 961
Meldeprotokoll, 1353
Meldezeile, 447
ODBC-Datenbankfeld, 1355
ODBC-Datenbanktabelle, 1356
Polygon, 470
Polygonzug, 471
Projektname in Protokollen, 1358
Protokollname in Protokollen, 1359
Rechteck, 476
Rezepturanzeige, 477, 480, 484
Rezepturprotokoll, 1361
Schalter, 488
Schaltfläche, 490
Schieberegler, 492
Seitenzahl in Protokollen, 1362, 1363
Sm@rtClient-Anzeige, 498
Symbolbibliothek, 501
Symbolisches E/A-Feld, 503, 505
symbolisches E/A-Feld in Protokollen, 1364
Tabellenanzeige, 516
Textfeld, 519, 1366
Uhr, 521
Verbinder, 523
Wertetabelle, 524
Wirkbereichs Name, 526
Wirkbereichs-Name (RFID), 528
Wirkbereichs-Signal, 529
WLAN-Empfang, 531
Zeigerinstrument, 532
Zonen-Name, 534
Zonen-Signal, 535
Darstellungsmodus
Linealfenster, 754
Statistikbereich-Fenster, 754
Statistik-Fenster, 754
Darstellungsprobleme
asiatische Zeichen, 2441
Steuerungsmeldungen, 2441
DataConnector, 1117
Daten
externe Daten in Protokollen, 1299
Daten wiederherstellen
Bediengerät, 2429, 2432
Datenfach
für Rezepturen, 1135
Datenfluss, 1127
Datenformatunterschiede
zwischen SIMATIC und Microsoft, 1246
Datenpuffer, 672
Datensatz
exportieren, 1180
importieren, 1180
lesen, 1152, 1163, 1178
Rezeptur, 1212
übertragen, 1153, 1164, 1179
Datensicherung
Bediengerät, 2429

Datentyp, 543, 556
Formatanpassung, 556
interne Variable, 548
Datentyp PLC, 632, 635, 636, 637, 638
Datentypen
PLC-Code-Anzeige, 1965, 1978
Datentypen PLC, 631
Datum/Uhrzeit-Feld, 362
Darstellung in Protokollen, 1342
Format, 363
Systemzeit anzeigen, 363
Variable verwenden, 363
Verwendung, 1342
Datum-Uhrzeit-Feld
Darstellung in Protokollen, 1343
Verwendung in Protokollen, 1343
Dauer
Anzeigeunterdrückung, 2338
deaktivieren, 1804
Projektsprache, 2100
Debugger, 1486, 1489
automatisch öffnen, 1489
in WinCC aktivieren, 1489
nach Fehlermeldung öffnen, 1489
starten für Global Script, 1489
starten für Graphics Runtime, 1489
Debugger:Bildwechsel beim Debuggen, 1486
Debugger:Grundlagen, 1486
DecreaseFocusedValue, 1596
DecreaseTag, 1623, 1742
definieren
Balkensegment, 350
Deinstallieren
Option, 2440
DeleteDataRecord, 1705
DeleteDataRecordMemory, 1706
Deltaladen, 2382
Design
ändern, 69
bildspezifische Layoutvorlagen übernehmen, 71
erstellen, 68
Layoutvorgaben auf Basiselemente und Elemente anwenden, 72
Layoutvorgaben eines Designs übernehmen, 70
Layoutvorlagen auf Controls anwenden, 72
löschen, 69
Designs, 2337
Designs für Bilder, 66
Desings
Standarddesign für Bilder, 73
Detailansicht
PLC-Code-Anzeige, 1956
Detailseite
aus Protokoll entfernen, 1307
Protokoll, 1283, 1286
sortieren, 1307
zu Protokoll hinzufügen, 1306
Diagnose, 1485, 1496, 1497, 1498
Meldeklasse, 877
Diagnose:Debugger, 1485
Diagnose:GSC-Diagnose, 1497
Diagnose:GSC-Diagnosefenster in ein Bild einfügen, 1498
Diagnoseinformationen
Service-Projekt, 2359
Diagonale Bewegung
Animation, 223
Dialog "Farbreferenz ändern", 2190
Direkttaste, 1574, 2323, 2412
projektieren, 259
DirekttasteBildnummer, 1577
doppelklicken, 1804
Doppel-T-Stück, 364
drehen
Objekt, 103, 116, 130, 1331
Drehung, 130
T-Stück, 520
Druckauftrag
anlegen für Protokoll, 1312
Aufgabe für automatischen Start erstellen, 1315
Druckauftrag/Skriptdiagnose, 366
Druckauftrag/Skriptdiagnose, 365
Druckauftrag, 366
Funktion, 365
Druckauftragsliste, 1308
DruckeBild, 1638
Drucken
Druckserver, 2451
Netzwerk, 2452
Printserver, 2451
Protokoll, 1307, 1334
Drücken, 1805
DruckeProtokoll, 1639
Druckserver, 2451
DVD 1, 2307
DVD 3, 2307
Dynamischer Zeitbereich, 724
Dynamisches Anmelden, 1381, 1391
dynamisieren
Bedienbarkeit eines Objekts, 224
Bewegung eines Objekts, 221
Bildbaustein, 303, 304
Bildbaustein-Typ, 303
blinken, 219

Direkte Bewegung, 223
Eigenschaftsanimation, 232
Eigenschaftsliste, 213
Farbe eines Objekts, 219
Gestaltung eines Objekts, 219, 1912
grüner Pfeil in Übersicht, 218
Objekt, 210
Objekteigenschaft, 210
Objekteigenschaft mit lokalem Skript, 245, 1484
Quality Code, 233, 240
Variablenanbindung, 227

E

E/A-Feld, 368
Anzeigeformat, 371, 373
Anzeigeformat in Protokollen, 1345
Datenformat, 368, 371, 373
Formatierungscodes, 371, 373
Modus, 368
Verdeckte Eingabe, 369
Verwendung in Protokollen, 1345
EA-Feld
sperren/entsperren, 2316, 2406
Ebene
Objekt einer Ebene zuordnen, 264
Eckpunkte, 470, 472
EditAlarm, 1686
Editierbares Textfeld, 376
Editiersprache, 2096
wählen, 2100
Editor
Designs, 66, 67
Grafiken, 2110
HMI Dateien, 2193
Editor "Lokale Skripte"
Eigenschaften, 1482
Verfügbarkeit, 1482
Editor Menüs & Symbolleisten, 319
Editor Rezepturabfragen, 1229
Editor Wirkbereiche, 1842, 1845
Editor Zonen, 1839
Eigenschaft Bedienmeldung geändert, 1805
Eigenschaft Block geändert, 1805
Eigenschaft Diagramm geändert, 1806
Eigenschaft Element Statuszeile geändert, 1808
Eigenschaft geändert, 1806
Eigenschaft Kurve geändert, 1806
Eigenschaft löschen
Bildbaustein-Typ, 286
Eigenschaft Schaltfläche Symbolleiste geändert, 1808
Eigenschaft Spalte geändert, 1807
Eigenschaft Spalte Meldestatistik geändert, 1807
Eigenschaft Spaltenname Statistikbereich geändert, 1807
Eigenschaft Wertespalte geändert, 1808, 1809
Eigenschaft X-Achse geändert, 1809
Eigenschaft Y-Achse geändert, 1809
Eigenschaft Zeitachse ändern, 1810
Eigenschaft Zeitspalte geändert, 1810
Eigenschaften
Editor "Lokale Skripte", 1482
Fenster, 2337
Eigenschaften für Slide-in-Bild
konfigurieren, 62
Eigenschaften Rechner, 1489
Registerkarte Runtime, 1489
Eigenschaftsanimation
Ausdruck, 233
Quality Code, 233, 240
Typ, 232
Typ Bereich Gleitkommazahl, 232
Typ Bereich integer, 232
Typ Binär, 233
Typ Bit, 233
Typ Bool, 233
Typ Direkt, 233
Eigenschaftsliste
dynamisieren, 213
Einblenden
Abschnitt im Protokoll, 1307
einfache Benutzeranzeige
Anzahl Zeilen,
Darstellung,
Darstellung in Runtime,
projektieren, 354, 448
Verwendung,
einfache Meldeanzeige, 958
Bedienelement, 959
Bedienung, 959
Bedienung mit der Tastatur, 960
Bedienung mit Maus, 960
Verwendung, 958
einfache Rezepturanzeige, 484
Bedienung, 1143
Darstellung, 1143
Einschränkungen, 1148
Menübefehl, 1144
projektieren, 484
Verhalten, 1144
einfaches Meldefenster
Verwendung, 958

Einfügen, 1498
Bibliotheksobjekt,
Bildgröße anpassen, 2147
Code-Vorlage in benutzerdefinierten Funktionen,
externe Grafik,
Farbe, 2142, 2143
Funktionstaste, 2147
Grafik,
Grafik in Grafiksammlung,
Objekt,
Prinzip, 2140, 2150
Rechteck,
Schriftart, 2142, 2143
ungültiges Objekt, 2142, 2143
Einfügen:GSC-Diagnose, 1498
Eingabe abgeschlossen, 1810
Eingabe am Bediengerät
über Funktionstaste, 2241, 2322, 2411
Eingabefeld
Normalmodus, 2239, 2320, 2409
Zeichenmodus, 2239, 2320, 2409
Eingabetaste, 2239, 2320, 2409
eingeben
Parameter in benutzerdefinierten
Funktionen, 1444
einrichten, 2360
Sortierung, 439
Einsatz, 1123
von Rezepturen, 1123
einstellen
Priorität,
Sprache, 2245, 2327, 2415
Totzone,
Verzögerungszeit,
Zeitbasis, 439
Einstellung
Sprachen im Betriebssystem, 2097
Einstellungen für Variablen, 564
Eintrag erstellen
C-Textliste, 209
Einzeiliger Modus
GRAPH-Übersicht, 398
Einzelmeldung
Ausblenden von Meldungen, 919
Filtern, 971
Elemente
in Protokollen, 1296
Elemente eines Anwenderdatentyps
mit Objekten verknüpfen, 144
Ellipse, 377
Radius horizontal, 377
Radius vertikal, 377
Ellipsenbogen, 378
Ellipsensegment, 379
E-mail-Benachrichtigung, 871
EncodeEx, 1573
END-Taste, 2239, 2320, 2409
Endwinkel
Ellipsenbogen, 378
Ellipsensegment, 380
Kreisbogen, 418
Kreissegment, 419
Energiesparmodus, 1868
Engineering-Station
Runtime starten, 2289, 2396
ENTER-Taste, 2239, 2320, 2409
entfernen
Formatierung im Meldetext, 833
Objekt aus der Gruppe, 165
Ordnerverknüpfung, 151
Entsperren
Bedienelement, 2316, 2406
Benutzer,
Eregnis
Schaltfläche Meldeanzeige klicken, 1821
Schaltfläche PLC-Code-Anzeige klicken, 1822
Ereignis
abgebaut, 1799
Abspielen beendet, 1799
aktivieren, 1800
Änderung, 1800
aufgebaut, 1800
Ausführen, 1801
Auswahl geändert, 1801
Bei Überschreitung, 1801
Bei Unterschreitung, 1801
Beim Öffnen eines Dialogs, 1802
Beim Schließen eines Dialogs, 1802
Benutzerwechsel, 1803
Bildwechsel, 1803
deaktivieren, 1804
doppelklicken, 1804
Drücken, 1805
dynamisieren, 215
Eigenschaft Bedienmeldung geändert, 1805
Eigenschaft Block geändert, 1805
Eigenschaft Diagramm geändert, 1806
Eigenschaft Element Statuszeile geändert, 1808
Eigenschaft geändert, 1806
Eigenschaft Kurve geändert, 1806
Eigenschaft Schaltfläche Symbolleiste
geändert, 1808
Eigenschaft Spalte geändert, 1807
Eigenschaft Spalte Meldestatistik geändert, 1807

Eigenschaft Spaltenname Statistikbereich geändert, 1807
Eigenschaft Wertespalte geändert, 1808, 1809
Eigenschaft X-Achse geändert, 1809
Eigenschaft Y-Achse geändert, 1809
Eigenschaft Zeitspalte ändern, 1810
Eigenschaft Zeitspalte geändert, 1810
Eingabe abgeschlossen, 1810
ESC doppelt drücken, 1811
Fehler, 1811
für Funktionslisten, 587
Gegangen, 1812
Gekommen, 1812
Im Meldebereich, 1812
Im Toleranzbereich, 1813
Im Warnbereich, 1813
Klicken, 1813
Klicken bei Blinken, 1814
Kontextmenü, 1814
Lautstärke, 1815
Layout ändern, 1815
Lineal ändern, 1815
Linke Maustaste drücken, 1816
Linke Maustaste loslassen, 1816
Loop-In-Alarm, 1816
loslassen, 1817
Maus bewegen, 1818
Medium geöffnet, 1818
Medium geschlossen, 1818
Meldungspufferüberlauf, 1817
Menü, 325
Objekt geändert, 1819
Pause, 1819
projektieren, 1147, 1158, 1170
Quittieren, 1819
Rand erreicht, 1820
Rechte Maustaste drücken, 1820
Rechte Maustaste loslassen, 1821
Runtime-Stopp, 1821
Schaltfläche Symbolleiste klicken, 1822
Schieber ändern, 1822
SchrittVor, 1823
SchrittZurück, 1823
Sperre geändert, 1823
Status geändert, 1824
Stopp, 1824
Symbolleiste, 325
Symbolleiste umschalten, 1824
Taste auf Tastatur drücken, 1826
Taste auf Tastatur loslassen, 1826
Taste drücken, 1825
Taste loslassen, 1825
Überlauf, 649, 1826
Umschalten, 1827
umschalten AUS, 1827
Umschalten EIN, 1828
Variablen, 587
Verbunden, 1828
Videoformat ändern, 1828
Vollbild, 1829
Vollbild verlassen, 1829
wenig freier Speicherplatz, 1829
Wenig freier Speicherplatz, kritisch, 1830
Wertänderung, 1830
Wiedergabe, 1831
Zeit abgelaufen, 1831
Ereignisgesteuerte Aufgaben
projektieren, 846
projektieren:Ereignisgesteuerte Aufgaben, 832, 850, 851, 854
Ereignisgesteuerte Ausgabe
Protokoll, 1308
Ereignisse, 665
Start, 665
Stopp, 665
Ereignis-Trigger, 1875, 1886, 1895
ErfasseBenutzeraktion, 1577
Erfassungsart
Variable, 588
Erfassungszyklus, 658, 665
festlegen, 592
Variable, 588, 589, 616, 619
ErhöheFokussiertenWert, 1579
ErhöheVariable, 1618, 1730
Ersatzwert, 591
festlegen, 591
Erscheinungsbild
eines Objekts dynamisieren, 219
ersetellen
Bildbaustein-Typ, 278
Ersetzen
Querverweise, 2091
erstellen
Benutzerdefinierte C-Funktion, 1469
benutzerdefinierte VB-Funktion, 1458
Formatvorlage,
Gruppe, 161
Hilfetext,
Protokoll (Übersicht),
Rezepturdatensatz am Bediengerät, 1150, 1151, 1161, 1175
Zyklus, 617, 621
Erstwertmeldung, 806
erweiterte Rezepturanzeige, 1154, 1165

erzeugen
Bildbaustein-Typ, 278
ESC doppelt drücken, 1811
ESC-Taste, 2239, 2320, 2409
Export
Benutzerverwaltung, 1412
Excel-Format, 2046
Meldung, 2054
Projekttexte, 2084, 2107
Rezeptur, 1694, 2046, 2050
Runtime-Daten, 1239
Textliste, 2078
Variable, 2069
Export von Runtime-Daten
Projektieren, 737, 750, 759
ExportDataRecords, 1692
ExportDataRecordsWithChecksum, 1694
ExportiereDatensätze, 1692
ExportiereDatensätzeMitPrüfsumme, 1694
ExportiereImportiereBenutzerverwaltung, 1613, 1728
exportieren
Rezeptur, 1180
Rezepturdatensatz, 1130, 1180
ExportImportUserAdministration, 1613, 1728
externe Grafik
Ordner bearbeiten, 150
Ordner umbenennen, 150
Ordner verknüpfen, 150
Ordnerverknüpfung entfernen, 150
externe Grafikdatei
in Grafiksammlung einfügen, 2112
in Grafiksammlung hinterlegen, 152
verwalten, 104

F

f(t)-Kurvenanzeige, 380, (Siehe Kurvenanzeige)
Bedienung in Runtime, 763
Online-Projektierung, 738, 751, 760
Symbolleiste, 382
Verwendung in Protokollen, 1346
f(x)-Kurvenanzeige, 384, 387, (Siehe Kurvenanzeige)
Online-Projektierung, 738, 751, 760
Regeln zur Datenversorgung, 722
Symbolleiste, 385, 389
Verwendung, 1347
Farbe der einzelnen Bereiche
Zeigerinstrument, 534
Farben
ändern, 2190
Meldeklassen, 444, 451, 2202, 2250
Suchen und Ersetzen, 2189
Farbverlauf
Balken, 347, 349
Fehler, 1811
Fehlermeldung
Laden, 2227, 2291
Fehlertypen, 1486
Fehlertypen:Laufzeitfehler, 1486
Fehlertypen:logische Fehler, 1486
Fehlertypen:Runtimefehler, 1486
Fehlertypen:Syntaxfehler, 1486
Fenster, 1497
Eigenschaften, 2337
Fenster umschalten, 2341
Fenstersteuerung, 2338
festes Seitenverhältnis, 502
festlegen, 655, 933
Cursormodus, 170
Referenzobjekt, 111, 1328
Skaleneinteilung beim Balken, 350
Textformat, 439
Filter projektieren
für feststehende Zeichenkette, 883
Kurzzeitarchiv, 906
Langzeitarchiv, 906
Filtern
der Meldeanzeige,
Meldeliste:filtern, 971
Filtern von Meldungen, 971
Format, 363
Formatanpassung
Datentyp, 556
Formatieren
Meldetext, 832
Formatierungscodes
E/A-Feld, 371, 373
Formatvorlage
bearbeiten, 87
erstellen, 85
kopieren, 88
löschen, 87
neuen Typen erstellen, 2043
verwenden, 89
Freigabeschaltfläche, 2315, 2405
konfigurieren, 2318, 2407
freigeben
Anwenderdatentyp, 611
Füllart, 502
Füllstand, 119

FunctionTrendControl
Online-Projektierung, 1240
Runtime-Daten exportieren, 1239
Funktion, 1431
einer Funktionstaste zuweisen, 253
Funktionsliste, 1446
Funktionen
Variablenwert aktualisieren, 542
Funktionsliste, 587, 1439, 1877
Abarbeitung in Runtime, 1500
asynchrone Abarbeitung, 1500
bearbeiten, 1442
in lokales Skript umwandeln, 1483
lokales Skript löschen, 1484
projektieren, 1440
synchrone Abarbeitung, 1500
Funktionstaste, 248
Funktion zuweisen, 253
global belegen, 250
globale Belegung, 2241, 2322, 2411
Globale Zuordnung verwenden, 51
Globales Bild, 51
Grafik zuweisen, 257
lokal belegen, 51, 252
lokale Belegung, 2241, 2322, 2411
mit Kennwort schützen, 255
zur Bildnavigation verwenden, 261
Funktionstasten, 2202, 2250
Funktionsumfang
ProSave, 2429
Funktionsweise
Service-Projekt, 2359
Fußzeile
Protokoll, 1283, 1286

G

Gegangen, 1812
Gehen, 798
GeheZumAnfang, 1725
GeheZumEnde, 1725
Gekommen, 1812
Gelöschte Variablen
prüfen, 560, 2337
Gerät tauschen
Beschränkung bei Verbindungen, 2126
Prinzip, 2126
geräteabhängige
Schriftgröße, 82
Geräteabhängigkeit
Bild, 37
Geräte-Assistent, 2011, 2014
Geräteversion, 2218, 2434
prüfen, 2188
wechseln, 2187
Gestaltung
Objekt, 119
Gestaltungsanalyse
Bitselektion, 2201, 2249, 2337
GetBrightness, 1655
GetDataRecordFromPLC, 1701
GetDataRecordName, 1703
GetDataRecordTagsFromPLC, 1704
GetGroupNumber, 1615
GetPassword, 1616
GetPLCMode, 1640, 1770
GetUserName, 1615
Globale Belegung
einer Funktionstaste, 248, 250
Globale Bibliothek, 2016
anlegen, 2026
Hochrüsten, 2185
öffnen, 2028
Protokolle anzeigen, 2029
speichern, 2027
Globale C-Variablen, 1466
Definition, 1466
Verwendung, 1466
Globale Zuordnung verwenden
Funktionstaste, 51
Globaler Softkey, 250
Globales Bild, 50
Funktionstaste, 51
Vorlage, 52
GMP, 2202, 2250
GMP-relevante Variable
Zeigerinstrument, 532
GoToEnd, 1725
GoToHome, 1725
GoTo-Objekt, 1917
Grafik
Abstand, 130
anpassen, 391, 394, 1349
aus der Grafiksammlung verwenden, 149
Ausrichtung, 129
Editor, 2110
einer Funktionstaste zuweisen, 257
einfügen, 105, 1320
einfügen in Grafiksammlung, 2110
Grafikanzeige, 391, 1348
Größenanpassung, 129
mit transparentem Hintergrund, 149
Schaltfläche, 490
verwalten, 150

Grafikanzeige, 391, 1348
transparente Farbe, 392
Grafikdatei
in Grafiksammlung hinterlegen, 152, 2112
Grafikliste
Bereich (... - ...), 198, 200, 202
Bereich (0 - 31), 197
Bit (0, 1), 203
Bitnummer (0 - 31), 205
erstellen, 196, 197
Grafisches E/A-Feld, 207
Projektierungsdaten ausgeben, 206
Verwendung, 194
Grafiklisten
Bitselektion, 2201, 2249
Grafiksammlung, 149, 152, 2112, 2201, 2249, 2337
Grafisches E/A-Feld, 393, 395
Grafikliste ausgeben, 207
Typ Bildlaufleiste, 395
Verwendung in Protokollen,
GRAPH-Übersicht, 396, 1937
Betreibsmodus, 397
Einzeiliger Modus, 398
Kompatibilitätsmodus, 398
Programmstatus anzeigen, 399
projektieren, 1952
Grenze
Login-Versuche, 2202, 2250
Grenzwert, 815
Variable, 568
Grenzwerte
anzeigen, 2202
im Tooltipp anzeigen, 2250
Variable, 567
Grenzwerte einer Analogmeldung
projektieren, 850
Größe
mehrerer Objekte ändern, 141, 1329
von Objekten in Protokoll, 1289
Größe anpassen
Bildbaustein, 299, 300
Gruppe
aufheben, 163
auflösen, 163
Bearbeiten, 161
erstellen, 161
Objekt entfernen, 165
Objekte hinzufügen, 164
Gruppenmeldung
Anzeigeunterdrückungsvariable, 1069
gruppieren
Objekt, 161
Gültigkeitsbereich, 1466

H

Handle für Slide-in-Bild
konfigurieren, 63
Handrad, 400
Hardcopy
Darstellung in Protokollen, 1350
HardwareSicherEntfernen, 1579
herunterladen, 2335
Hierarchieebenen
Benutzerrechte, 2202, 2250
Hierarchische Meldegruppen, 809
anlegen, 826
Hilfeindikator, 401
Hilfetext, 815
anzeigen, 958, 2240, 2244, 2321, 2326, 2410, 2415
erstellen, 829, 831, 846, 850, 854
Taste, 2240, 2321, 2410
Hintergrund gestalten, 119
Hintergrundfarbe
dynamisch ändern, 219
hinterlegen
externe Grafik, 152
hinzufügen
.Net-Control, 147
ActiveX-Control, 147
Anwendertextblock zu Meldung, 859
Objekte zur Gruppe, 164
Hitliste, 974
HMI Backup
anzeigen, 2300
löschen, 2302
umbenennen, 2302
HMI Dateien, 2193
HMI Formmatvorlage, 2043
HMI Stil, 2043
HMI Style, 2043
HMI Style sheet, 2043
HMI-Anwenderdatentypen
mit Bildbausteinen verschalten, 284
HMI-Arrayvariablen konfigurieren, 601, 606
HMI-Variablen
PLC-Arrayvariablen verbinden, 600, 605
Hochrüsten
Globale Bibliothek, 2185
Projekt, 2183
Projektversion, 2183
HOME-Taste, 2239, 2320, 2409

Hotkey, 2343
Anmelden, 2343
festlegen, 2343
Hardcopy, 2343
Logoff, 2343
Hotkey festlegen, 491
HTML-Browser, 402, 407
Datei-Explorer, 402, 407
Schaltfläche, 404
Verbindung zu ftp-Server, 402, 407
HTMLBrowserAbbrechen, 1655
HTMLBrowserAbwärtsScrollen, 1656
HTMLBrowserAktualisieren, 1656
HTMLBrowserAufwärtsScrollen, 1657
HTMLBrowserHeranzoomen, 1657
HTMLBrowserHerauszoomen, 1658
HTMLBrowserNachLinksScrollen, 1658
HTMLBrowserNachRechtsScrollen, 1658
HTMLBrowserSeiteAbwärts, 1659
HTMLBrowserSeiteAufwärts, 1659
HTMLBrowserStartseite, 1660
HTMLBrowserVorwärts, 1660
HTMLBrowserZurück, 1661

I

Im Meldebereich, 1812
Im Toleranzbereich, 1813
Im Warnbereich, 1813
Image, 2187, 2217, 2218
Bediengerät, 2426, 2427, 2428
Import
Benutzerverwaltung, 1413
Meldung, 2056, 2057
Projekttexte, 2083, 2109
Rezeptur, 2048, 2051
Struktur der Analogmeldung, 2058, 2063
Struktur der Bitmeldung, 2061
Struktur der Rezepturdaten, 2053
Struktur der Variablendaten, 2076
Struktur für Bitmeldungen, 2067
Textliste, 2080
Variable, 2071, 2072
ImportDataRecords, 1697
ImportDataRecordsWithChecksum, 1698
ImportiereDatenätzeMitPrüfsumme, 1698
ImportiereDatensaetze, 1697
importieren
Rezeptur, 1180
Rezepturdatensatz, 1130, 1180
in ein Bild einfügen, 1498
in Gruppe verschieben
Pop-up-Bild, 59
IncreaseFocusedValue, 1579
IncreaseTag, 1618, 1730
IndexVariablen, 585
indirekt adressieren, 585
Indirekte Adressierung
von Variablen, 593
Informationsbereich
PLC-Code-Anzeige, 1956
initialisieren
Steuerungsdaten, 2196
Installieren
Option, 2440
Integrierte Verbindung, 545
Intel-Format, 1246
Inter-Project_Engineering
IPE, 2196
InverseLinearScaling, 1619, 1731
InvertBit, 1624, 1759
InvertBitInTag, 1625, 1761
InvertiereBit, 1624, 1759
InvertiereBitInVariable, 1625, 1761
InvertiereLineareSkalierung, 1619, 1731
IPE, 2196

K

KalibriereTouchscreen, 1652
Kamera URL, 409
Kamera-Anzeige
Projektieren, 409
Kanaldiagnose-Anzeige, 413
Kategorie löschen
Bildbaustein-Typ, 286
Kennwort, 2202, 2250
ändern, 1388
für Funktionstaste, 255
Hierarchieebene, 1375, 1378
Kennwortalterung, 1376
Kennwortkomplexität, 1376, 1380
Kennwort für den Transport
anlegen, 2207, 2257, 2362
zurücksetzen, 2208, 2258, 2363
Kennwortschutz in Runtime, 2206, 2256, 2361
kennzeichnen
Meldeklasse, 450
Klicken, 1813
Klicken bei Blinken, 1814
Know-how-Schutz
benutzerdefinierte Funktion öffnen, 1477
einrichten, 1477

entfernen, 1478
Passwort ändern, 1479
Kommen, 798
Kommentar, 997
Kommentare
zu Meldungen, 820
Kommentare in Runtime, 2202, 2250
Kommunikation
über Rohdatenvariablen, 1247
Kompatibilität
WinCC-Versionen, 2181, 2182
Kompatibilitätsmodus, 565
Kompatibilitätsprüfung, 2201, 2249
Projektkennung, 2201, 2249
komplexe Benutzeranzeige
projektieren, 354
Komprimierung, 672
Konfiguration
Variable, 560
Konfigurationsdatei
Menü, 327
Symbolleiste, 327
Konfigurieren
Eigenschaften für Slide-in-Bild, 62
Größe Archiv, 688
Größe Variablenarchiv, 688
Handle für Slide-in-Bild, 63
Markierung in der Tabellenanzeige,
mehrere Variablen,
Meldearchiv, 940
Rechteck,
Slide-in-Bild, 62
Slide-in-Bild für Geräte ohne Multitouch-Funktion, 64
Kontextmenü, 1814
Koordinierte Übertragung, 1135
mit Steuerung, 1135
Kopfzeile
Protokoll, 1283, 1286
kopieren
Analogmeldung,
Archivvariable,
Aufgabe,
Bediengeräte mit HMI-Verbindungen, 2146
Benutzer,
benutzerdefinierte Ordner,
Benutzergruppe,
Berechtigung,
Bild, 44
Bildgröße anpassen,
Bildobjekt,
Bildobjekte mit Anwederdatentypen,
Bitmeldung,
Einschränkungen,
Excel-Format,
Farbe,
Formatvorlage, 88
Funktionstaste,
Grafikliste,
Meldegruppe,
Meldeklasse,
Pop-up-Bild, 59
Prinzip,
Protokoll,
Protokollobjekt,
Rezepturdatensatz in Runtime, 1161, 1175
Schriftart,
Strukturelement,
Strukturtyp,
Textliste,
ungültiges Objekt,
Variable, 559
Variablenarchiv,
VB-Skript,
Verbindung,
verknüpfte Objekte,
Vorlage, 54
Zyklus,
Kopiervorlage
Bibliothek, 2020
Kreis, 416
Füllzustand, 119
gestalten, 119
Radius, 417
Kriterienanalyse-Anzeige, 1937
Kurve, 706
Darstellungsformen, 723
gemeinsame Achsen, 726
Kurvenverlauf, 724
mehrere Achsen, 725
projektieren, 733
Schreibrichtung, 724
Kurven
in Protokollen, 1297
Kurvenanzeige, 422, 706, 721
Anzahl Kurven, 720
Auflösung einer Kurvendarstellung, 720
Ausschnitt der ~ in Runtime vergrößern, 770
Bedienung in Runtime, 761, 765
Bereichsanteil, 732
Darstellungsformen, 723
Datenversorgung projektieren, 734, 747
Elemente der Statuszeile, 782

Export von Runtime-Daten projektieren, 737, 750, 759
für Archiv projektieren, 712
für Werte aus der Steuerung projektieren, 704, 709
in archivierten Daten blätern, 781
Kurve im Vordergrund darstellen, 772
Kurve in Runtime vergrößern, 770
Kurve in Runtime verkleinern, 770
Kurve projektieren, 733
Kurvenbereich in Runtime verschieben, 772
Kurvenverlauf, 724
Online-Projektierung, 779
projektieren, 728
Reihenfolge von Kurven ändern, 772
Schaltfläche, 423
Schaltflächen, 761
Schreibrichtung, 724
Statuszeile, 736, 749, 756
Symbolleiste, 736, 749, 756, 761
unterteilen, 731
Wertachse in Runtime zoomen, 771
Wertachse projektieren, 731
X-Achse projektieren, 731
Y-Achse projektieren, 731
Zeitachse in Runtime zoomen, 771
Zeitachse projektieren, 730
KurvenanzeigeBlättereVorwärts, 1661
KurvenanzeigeBlättereZurück, 1662
KurvenanzeigeErweitern, 1662
KurvenanzeigeKomprimieren, 1662
KurvenanzeigeLinealLinks, 1664
KurvenanzeigeLinealRechts, 1663
KurvenanzeigeSetzeLinealmodus, 1664
KurvenanzeigeStartenStoppen, 1665
KurvenanzeigeZurückZumAnfang, 1665
Kurvendarstellung
Auflösung, 720
Kurzzeitarchiv
Filter projektieren, 906
projektieren, 906
Sortierung projektieren, 906
Kurzzeitarchivliste, 974

L

LadeDatensatz, 1700
laden
"Pack&Go"-Datei auf Bediengerät, 2279
Bediengerät,
Dateien in das Projekt, 2193
Dateien in Runtime, 2193
Fehlermeldung,
Projekt, 2220, 2380
Rezeptur,
Rezepturdatensatz in Runtime, 1162, 1175, 1176
über USB,
Laden eines Projekts
mit angeschlossenem Bediengerät, 2217, 2269, 2272, 2372, 2375, 2377
ohne angeschlossenes Bediengerät, 2218, 2270, 2272, 2373, 2375, 2377
vom externen Speichermedium, 2218, 2270, 2373
Laden über S7-Routing, 2218
Laden von inkonsistenten Projekten, 2378
Ladezustand, 424
Bedienung, 425
Langzeitarchiv
Filter projektieren, 906
projektieren, 906
Sortierung projektieren, 906
Langzeitarchivliste, 974
Langzeitarchivserver, 674
Installation (Übersicht), 674
Lautstärke, 1815
Layout ändern, 1815
Leistungsmerkmale
Bediengerät, 2455, 2458, 2462, 2468, 2473, 2478
WinCC Runtime, 2473
WinCC Runtime Professional, 2478
WinCC Runtime Professional Performance, 2482
LeseBenutzername, 1615
LeseDatensatzAusSteuerung, 1701
LeseDatensatzname, 1702
LeseDatensatzvariablenAusSteuerung, 1704
LeseGruppennummer, 1615
LeseHelligkeit, 1655
LeseKennwort, 1616
LesePLCModus, 1640, 1770
Leuchtdiode
quittieren, 2240, 2321, 2410
umschalten, 2240, 2321, 2410
Leuchtdrucktaster, 425
Variable, 426
License Keys, 2437
License Keys
auf ein Bediengerät übertragen, 2438
Lineal ändern, 1815
LineareSkalierung, 1620, 1733
LinearScaling, 1620, 1733
Linie, 429
gestalten, 119

Linienanfang, 430
Linienende, 119, 430
Linienanfang
Linie, 430
Polygonzug, 471
Linienende
Linie, 430
Polygonzug, 471
Linke Maustaste drücken, 1816
Linke Maustaste loslassen, 1816
Linked Server, 1117
Liste der ausgeblendeten Meldungen, 919
Listenfeld, 430
Lizenz
verwalten, 2438
LoadDataRecord, 1700
LoescheArchiv, 1607
Login-Versuche
Grenze, 2202, 2250
Logische Fehler, 1486
Logoff, 1612
Logon, 1613
Lokale Belegung
einer Funktionstaste, 248, 252
Lokale Benutzergruppen
Web-Zugriff, 2338
lokale Variable, 1452
Lokales Skript, 1431
am Ereignis löschen, 1484
am Ereignis projekteren, 1483
an Objekteigenschaft löschen, 245, 1484
Objekteigenschaft dynamisieren, 245, 1484
verwenden, 1480
Loop-In-Alarm, 1816
einstellen, 440
projektiert, 864
triggern, 960
LöscheDatensatz, 1705
LöscheDatensatzSpeicher, 1706
LöscheMeldepuffer, 1686
LöscheMeldepufferProTool, 1687
löschen
Benutzer, 1389
Benutzergruppe, 1390
Berechtigung, 1390
Bild, 44
Formatvorlage, 87
lokales Skript am Ereignis, 1484
lokales Skript an Objekteigenschaft, 245, 1484
Objekt, 106, 1321
Pop-up-Bild, 59
Protokollseite,
Rezepturdatensatz in Runtime, 1162, 1176
Variable, 559
Vorlage, 54
Löschen-Taste, 2239, 2320, 2409
loslassen, 1817

M

Manuelles Ausblenden, 919
Manuelles Unterdrücken der Anzeige, 896
Maus bewegen, 1818
Media Player, 432
Anzahl Wiedergaben, 433
Medium geöffnet, 1818
Medium geschlossen, 1818
Mehrfachauswahl, 111, 1327
Meldanzeige
Filter projektieren, 891
Meldeanzeige, 435, 445, 447, 452, 870, 886, 954, 960, 968, 974
~ für archivierte Meldungen projektieren, 879, 927
Anzeige der S7 Diagnosemeldungen projektieren, 878
Anzeige filtern, 883
Anzeigebereich festlegen, 444
Bedienelement, 436, 446, 449, 956
Bedienelemente, 442
Bedienung, 955, 960
Bedienung mit Maus, 957, 969
Bedienung mit Tastatur, 957, 969
Darstellung, 445, 955, 959
Darstellung projektieren, 876
Datenexport projektieren, 907
Export von Runtime-Daten projektieren, 737, 750, 759
Filtern von Meldungen, 971
Meldeblöcke für ~ aktivieren, 890
Meldeklasse kennzeichnen, 450
Meldeklassen auswählen, 442
Meldeklassen kennzeichnen, 443
Meldetextfenster, 960
Meldezeile, 447
Meldungen filtern, 443
Online-Projektierung, 738, 751, 760
projektieren, 873, 888
Sortierung, 443
Sortierung freigeben, 450
Spalte, 440, 446, 450
Spalten festlegen, 443
Spaltenreihenfolge, 443
Spaltenüberschriften, 444

Statuszeile, 736, 749, 756
Symbol- und Statusleiste projektieren, 890
Symbolleiste, 736, 749, 756
Verwendung, 954
Verwendung in Protokollen, 1351
Zugriffsschutz, 442
MeldeanzeigeAktualisieren, 1688
MeldeanzeigeBearbeiteMeldung, 1688
MeldeanzeigeQuittiereMeldung, 1689
MeldeanzeigeZeigeHilfetext, 1690
Meldearchiv, 921, 1071, 1077, 1099, 1108
anlegen, 921, 924
Archivdaten abfragen, 1099, 1108
Auslagern, 942
Backup, 942
Daten darstellen, 1101
in Runtime, 953, 967
Inhalt anzeigen, 885
konfigurieren, 940
Meldeanzeige projektieren, 885
projektieren, 924, 937
Recordset, 1101
Signierung Backup, 942
verbinden, 947
vom Projekt trennen, 947
Meldearchivierung
Speichermedium, 938
Meldeblock, 843
Projektieren, 841
Meldeereignis
Gehen, 798
Kommen, 798
Quittieren, 798
Meldefenster, 870, 954, 974, 975
Archivdaten ausgeben, 975
Bedienelement, 956
Bedienung, 955
Bedienung mit Maus, 957
Bedienung mit Tastatur, 957, 969
in Runtime, 951, 954, 967
projektieren, 880, 975
Verwendung, 954
Meldefolgeprotokoll, 911, 1300
Verwendung in Protokollen, 1352
Meldegruppe, 808, 815
anlegen, 824, 825
Meldung quittieren, 805
projektieren, 824
Variablen, 811
Zustandsvariable, 886
Meldegruppen aus Meldeklassen, 809
Meldeindikator, 455, 871, 961
Bedienung, 961
Bedienung mit Maus, 962
Darstellung, 961
Ereignisse, 455
in Runtime, 951, 954, 967
Meldeklassen, 455
projektieren, 880
Verwendung, 961
Meldeklasse, 814
Darstellung, 955, 958
Diagnose, 877
in Runtime, 950, 952, 965
kennzeichnen, 450
Meldeklassen, 800, 801
benutzerdefinierte, 800, 802
Farben, 444, 451, 2202, 2250
projektweite, 802
verwenden, 800, 802
vordefinierte, 800, 802
Meldeklassen auswählen
Meldeanzeige, 442
Meldeklassen kennzeichnen
Meldeanzeige, 443
Meldeliste, 971, 974
Meldelisteneinträge nach Stromausfall, 2338
Meldenarchiv
Namenskonventionen, 1075
Meldenummer, 812, 813, 814
Meldeparameter Anwendermeldung
Projektieren, 857
Meldeparameter einer Anwendermeldung definieren
Beispiel, 857
Meldepriorität, 997
Meldeprotokoll
Darstellung in Protokollen, 1353
Verwendung in Protokollen, 1353
Meldepuffer, 927, 2202, 2250
in Runtime, 951, 953, 966
Speichergröße, 927
Meldepufferüberlauf, 927
Meldestatistik, 886, 892
projektieren, 894
Meldesystem, 785, 787, 788
Meldetext, 815
formatieren, 832
Formatierung entfernen, 833
Meldeverfahren, 790, 791, 792
Meldezeile, 441, 447
Meldezustand, 815
Gegangen, 798

Gekommen, 798
Quittiert, 798
Meldung
Anwendertextblock hinzufügen, 859
Anwendertextblöcke, 840
Anzeige unterdrücken, 896
anzeigen, 445, 447
Anzeigeunterdrückungsmaske, 896
ausblenden, 919
ausgeben, 436
bearbeiten, 957
Bestandteile, 814
Darstellung in Runtime, 974
Ereignis, 994, 995
exportieren, 2054
Filtern, 971
Hilfetext aufrufen, 961
importieren, 2056, 2057
in Runtime, 950, 952, 965
in Runtime auswerten, 908
Loop-In-Alarm, 969
Parameterblöcke, 840
Priorität, 997
projektieren, 827, 843, 863, 866
quittieren, 957, 960
Reihenfolge wird nicht eingehalten, 1502
Systemblöcke, 840
Systemfunktion, 993
Meldung importieren
Struktur für den Import, 2067
Meldung mit Einfachquittierung, 806
Meldung ohne Quittierung, 806
Meldung quittieren
Meldegruppe, 805
Meldungen
fortlaufend protokollieren, 1300
in Protokollen, 1298
Kommentare anlegen, 820
Parameter in Analogmeldung ausgeben, 848
Parameter in Anwendermeldung ausgeben, 855
Parameter in Bitmeldung ausgeben, 852
rückladen, 948
Text aus Textliste ausgeben, 835
Wert einer Variable ausgeben, 833
Meldungen filtern
Meldeanzeige, 443
Meldungen in der Ereignisanzeige
Web-Zugriff, 2338
Meldungspufferüberlauf, 1817
Menü
benutzerdefiniert, 316, 329
Ereignis projektieren, 325
Hauptenübefehl anlegen, 321
Menübefehl anlegen, 322
Untermenübefehl anlegen, 322
Menübefehl
einfache Rezepturanzeige, 1144
Menüs
Zeichensatz, 2338
Menüs & Symbolleisten, 2337
Microsoft Script Debugger, 1485
Mitsubishi MC
Datentypen, 638
Mobile Panel
Anzeige- und Bedienobjekt, 340
Mobile Panels
verfügbare Systemfunktionen, 1554
Mobile Wireless, 1843, 1845
Arbeitsbereich, 1843
Zonen-ID / Anschlusspunkt-ID, 1869
Mobile Wireless (RFID)
Arbeitsbereich, 1846
Mobile Wireless V2, 1846
Modicon Modbus TCP/IP
Datentypen, 637
Modus, 490
E/A-Feld, 368
Grafisches E/A-Feld, 394, 395
Symbolisches E/A-Feld, 505
MS OLE DB, 1115
Multiplexen,
Adress-Multiplexen, 576
mit absoluten Adressen, 576
mit symbolischen Adressen, 576
Multirouchgeräte
Ansicht vergrößern und verkleinern, 2313
Multitouch-Bedienung, 2403
Multitouchgeräte
Gesten, 2314, 2403
scrollen, 2312
zoomen, 2313
Zweihand-Bedienung, 2315, 2405
Multiuser Engineering
Regeln zum Übersetzen und Laden, 2224, 2286
Übersetzen in der lokalen Session, 2225, 2288, 2389
Übersetzen in der Serverprojekt-Ansicht, 2224, 2287, 2388
Übersetzen und Laden, 2223, 2285

N

Name ändern
Benutzer, 1388

Benutzergruppe, 1389
Berechtigung, 1390
Namen laden, 2202, 2250
Namenskonventionen
Meldearchiv, 1075
Variablenarchiv, 1075
Neu verbinden
Variable, 566
neuen Typen erstellen
Formatvorlage, 2043
Stil, 2043
Neues Schlüsselpaar
generieren, 2208, 2258, 2363
Neuwertmeldung, 807
nicht-multilinguale Objekte
Runtime-Sprache, 2338
Normalbereich anzeigen, 534
Normierungs-DLL, 660
NotifyUserAction, 1577
Notquittierung, 982
numerische Tastenbelegung, 2240, 2321, 2410
Nummer anzeigen, 485
Rezepturanzeige, 481

O

Oberflächensprache, 2095
auswählen, 2099
Objekt
anordnen, 103, 115, 1325
außerhalb des Bereichs, 116
bearbeiten innerhalb einer Gruppe, 166
blinken, 138, 139
bündig ausrichten, 113, 1330
des gleichen Typs einfügen, 103
Direkte Bewegung dynamisieren, 223
drehen, 103, 116, 130, 1331
dynamisieren, 210
Eigenschaft mit lokalem Skript dynamisieren, 245, 1484
einer Ebene zuordnen, 264
einfügen, 103, 105, 1320
Erscheinungsbild dynamisieren, 219
Füllmuster gestalten, 122
gestalten, 119
gleichmäßig ausrichten, 113, 1331
Größe ändern, 109, 1326
gruppieren, 161
Hintergrundfarbe gestalten, 121
in Bibliothek hinterlegen, 2038
lokales Skript löschen, 245, 1484
löschen, 106, 1321
mehrere auswählen, 103, 111, 1327
mehrfach einfügen, 139, 1322
OLE-Objekt aus Datei erstellen, 104
OLE-Objekt neu erstellen, 104
Position ändern, 107, 1323
Position und Größe mehrerer Objekte ändern, 104, 141, 1329
referenzieren, 1455
spiegeln, 103
Tab-Reihenfolge, 103
Transparenz, 122
Verfügbarkeit, 344
Verfügbarkeit für Basic Panel, 337
Verfügbarkeit für Comfort Panel, 338
Verfügbarkeit für Mobile Panel, 340
Verfügbarkeit für WinCC Runtime Advanced, 343
vervielfachen, 139, 1322
Objekt geändert, 1819
Objekte
CSV-Provider Kurven, 1340
CSV-Provider Tabelle, 1341
Datum-Uhrzeit-Feld, 1343
f(t)-Kurvenanzeige, 1346
Größe in Protokollen, 1292
Meldeanzeige, 1351
Meldefolgeprotokoll, 1352
Meldeprotokoll, 1353
mit dynamischer Länge in Protokollen, 1293
ODBC-Datenbankfeld, 1354
ODBC-Datenbanktabelle, 1356
Pop-up-Bild, 57
Position in Protokollen, 1292
Projektname, 1358
Protokollname, 1359
Rezepturanzeige, 1360
Rezepturprotokoll, 1361
Seitenzahl, 1362, 1363
Slide-in-Bild, 61
Tabellenanzeige, 1365
Objekte verschieben
Tastenkombination, 114
Objekteigenschaft
dynamisieren, 210
Objektgröße
vereinheitlichen, 110, 1326
Objektgruppe
Animation, 229, 230
Objekt entfernen, 165
Objekt innerhalb einer Gruppe bearbeiten, 166
Objektliste
Variable, 551

ODBC-Datenbankfeld
Darstellung in Protokollen, 1355
Verwendung in Protokollen, 1354
ODBC-Datenbanktabelle
Darstellung in Protokollen, 1356
Verwendung in Protokollen, 1356
ÖffneAlleArchive, 1608
ÖffneBildschirmtastatur, 1720
ÖffneDateiBrowser, 1718
ÖffneDialogSystemsteuerung, 1580
ÖffneEingabeaufforderung, 1581
ÖffneInternetExplorer, 1581
öffnen
Auswahlliste, 2239, 2320, 2409
Globale Bibliothek, 2028
Menüs & Symbolleisten, 319
ÖffneSystemsteuerung, 1582
ÖffneTaskManager, 1719
OLE-Objekt
aus Datei erstellen, 104
in Grafiksammlung hinterlegen, 152, 2112
neu erstellen, 104
Online
Rezepturvariable, 1131
Online-Deltaladen, 2382
Einschränkungen, 2384
Projektieren, 2384
Unterstützung WinCC Web Navigator, 2384
Online-Hilfe
Verfügbarkeit in Runtime, 2338
Online-Projektierung, 738, 751, 760, 1240
Kurvenanzeige, 779
OnlineTrendControl, 768
Tabellenanzeige, 775, 779
OnlineTableControl
Online-Projektierung, 1240
Runtime-Daten exportieren, 1239
OnlineTrendControl
Online-Projektierung, 768, 1240
Runtime-Daten exportieren, 1239
OP 177B mono
Symbolbibliothek, 502
OPC, 1117
Meldungen und Ereignisse, 2338
OPC Historical Access, 2338
OPC Unified Architecture Server, 2202, 2250, 2338
OPC-DA
Datentypen, 635
OPC-XML-DA
Datentypen, 636
OPC-Zugriff
Service-Projekt, 2359
OpenAllLogs, 1608
OpenCommandPrompt, 1581
OpenControlPanel, 1582
OpenControlPanelDialog, 1580
OpenFileBrowser, 1718
OpenInternetExplorer, 1581
OpenScreenKeyboard, 1720
OpenTaskManager, 1719
Option
deinstallieren, 2440
installieren, 2440
Optionen der Drittanbieter, 2335
optische Rückmeldung, 2400
optisches Handlesegerät, 2246, 2333
Ordner
anlegen im Editor HMI Dateien, 2193
Ordner verknüpfen
externe Grafik, 150
Ordnerverknüpfung
bearbeiten, 151
entfernen, 151
umbennen, 151
östliche Zeichen
Eingabe am Bediengerät, 2122
Overridefunktion, 530

P

Pack&Go-Datei
auf Bediengerät laden, 2279
erzeugen, 2277
PageDown, 1724
PageUp, 1724
Parameter
in Analogmeldung ausgeben, 848
in Anwendermeldung ausgeben, 855
in Bitmeldung ausgeben, 852
Parameterblock
Projektieren, 841
Parameterblöcke, 840
Parameterübergabe
benutzerdefinierte Funktion, 1449
Pause, 1819
PDFAbwärtsScrollen, 1666
PDFAnpassenAufBreite, 1667
PDFAnpassenAufHöhe, 1667
PDF-Anzeige
Multitouch-Bedienung, 460
Projektieren, 457
Schaltfläche, 460
Symbolleiste, 458
PDFAufwärtsScrollen, 1666

PDFGeheZurErstenSeite, 1668
PDFGeheZurLetztenSeite, 1668
PDFGeheZurNächstenSeite, 1669
PDFGeheZurVorherigenSeite, 1670
PDFGeheZuSeite, 1669
PDFHeranzoomen, 1670
PDFHerauszoomen, 1670
PDFLinksScrollen, 1671
PDFRechtsScrollen, 1671
PDFZoomOriginal, 1672
Permanentbereich, 54
PLC-Anwenderdatentypen
    mit Bildbausteinen verschalten, 284
PLC-Code-Anzeige, 463, 467, 1937
    Anweisungen, 1961, 1974
    Bedienung mit Tastatur, 465
    Datentypen, 1965, 1978
    Detailansicht, 1956
    Informationsbereich, 1956
    projektieren, 465, 1954, 1968
    Symbolleiste, 464, 468, 1959
    Symboltabelle, 1956
    Transition/Interlock-Ansicht, 1957
PLCCodeAnzeigeAktualOderErstwerte, 1677
PLCCodeAnzeigeBildlaufleisteVerschieben
    PLCCodeViewMoveScrollbar, 1674
PLCCodeAnzeigeDetails, 1675
PLCCodeAnzeigeHeranzoomen, 1678
PLCCodeAnzeigeHerauszoomen, 1678
PLCCodeAnzeigeNächstesNetzwerk, 1675
PLCCodeAnzeigeSymbole, 1676
PLCCodeAnzeigeTransitionInterlock, 1677
PLCCodeAnzeigeVorherigesNetzwerk, 1679
PLCCodeAnzeigeZeigeSchrittModus, 1676
PLC-Datentyp in WinCC, 543
PLC-Datentypen, 634
PLC-Präfix, 565
PlcUDTs
    WinCC, 634
PLC-Variable
    neu verbinden, 566
PLC-Variable neu verbinden, 566
PLC-Variable synchronisieren, 566
Polygon, 470
    Darstellung, 470
    Eckpunkte projektieren, 470
    Radien, 470
Polygonzug, 471
    Eckpunkte projektieren, 472
    Linienanfang, 471
    Linienende, 471
    Radien, 472
Pop-up-Bild, 56
    anlegen, 58
    in Gruppe verschieben, 59
    kopieren, 59
    löschen, 59
    Objekte, 57
Position
    mehrerer Objekte verändern, 141, 1329
    von Objekten, 107, 1323
    von Objekten in Protokoll, 1289
Power Management, 1868
PrintReport, 1639
PrintScreen, 1638
Printserver, 2451
Priorität, 997
    einstellen, 846
Process Historian
    Serverauswahl, 2339
    Zwischenspeicherungspfad, 2339
ProDiag-Übersicht, 473
    projektieren, 1946
    Symbole, 1944
    Verwendung, 473
PROFIBUS DP
    Direkttaste, 1577
PROFIsafe
    trennen, 1853
PROFIsafe-Adresse, 1868
PROFIsafeTrennen, 1583
Programmstatus anzeigen
    GRAPH-Übersicht, 399
Projekt, 2368
    aktivieren, 2397
    hochrüsten, 2183
    laden, 2220, 2276, 2380
    laden (Übersicht), 2208, 2258, 2363
    mehrsprachig, 2102
    über Variablen-Simulator simulieren, 2213, 2263
    übersetzen, 2210, 2260, 2365
    übersetzen (Übersicht), 2208, 2258, 2363
    WinCC Status, 2197
Projekt anpassen
    für anderes Bediengerät, 2230, 2294
Projektbibliiothek, 2514
Projektbibliothek, 2016
Projektdaten
    Meldung exportieren, 2054
    Meldung importieren, 2055, 2057
    Rezeptur exportieren, 2046, 2050
    Textliste exportieren, 2078
    Variable exportieren, 2069
    Variable importieren, 2070, 2072

Projektgröße
reduzieren, 2231, 2295
Projektieren, 832, 850, 851, 854, 869
Analogmeldung, 827, 843, 868
Analogmeldungen, 816, 817
Animation,
Anzeige der S7 Diagnosmeldungen, 878
Archivvariable, 656, 700
Auswertung von Meldungen in Runtime,
Benutzerdefinierte C-Funktion,
benutzerdefinierte VB-Funktion,
Berechtigung,
Bitmeldung, 830, 866
Bitmeldungen, 816, 817
Darstellung der Meldeanzeige, 876
Datenexport aus Meldeanzeige,
Datenversorgung der Kurvenanzeige,
Datenversorgung der Tabellenanzeige,
Ereignis,
Ereignisgesteuerte Aufgaben, 829, 846
Filter für Meldeanzeige,
Funktionsliste,
GRAPH-Übersicht, 1952
Grenzwerte einer Analogmeldung, 850
Kamera-Anzeige, 409
Kurve,
Kurvenanzeige,
Kurvenanzeige für Archiv,
Kurvenanzeige für Werte aus der Steuerung,
Kurzzeitarchiv, 906
Langzeitarchiv, 906
lokales Skript an Ereignis,
Meldeanzeige, 873
Meldeanzeige für archivierte Meldungen, 879, 927
Meldearchiv, 937
Meldeblock, 841
Meldegruppe, 824
Meldeparameter Anwendermeldung, 857
Meldestatistik,
Meldung, 843
PLC-Code-Anzeige, 1954, 1968
ProDiag-Übersicht, 1946
Projektweite Meldeklasse, 822
Quittierung Meldung, 913, 915
Rezepturanzeige,
Rezepturbild,
Sortierung einer Tabellenspalte,
Steuerungsmeldungen, 816, 817
Symbol- und Statusleiste der Meldeanzeige,
Systemmeldungen, 816, 817
Tabellenanzeige,
Tabellenelemente,
Trigger Anwendermeldung, 856
Verbindung Archiv-Backup, 946
Wertachse,
Wertetabelle,
Wertspalte,
X-Achse,
Zeitachse,
Zentraler Melder, 905
Zugriffsschutz,
Projektierung
analoge Archivvariable, 683
Archiv, 677
Archivvariable, 681
binäre Archivvariable, 682
Prozessgesteuerte Variable, 681
Variablenarchiv, 677
Verdichtungsarchiv, 679
Verdichtungsvariable, 681
Projektierungs-PC
Runtime starten, 2289, 2396
Projektkennung, 2201, 2249
Projektname
Darstellung in Protokollen, 1358
Verwendung in Protokollen, 1358
Projektsprache, 2095
aktivieren, 2099
deaktivieren, 2100
Projektsprachen
Anwendertexte, 2103
Systemtexte, 2103
Projekttext
exportieren, 2084, 2107
Projekttexte
Einzelne Texte übersetzen, 295, 2104
Exportieren, 296
importieren, 2083, 2109
Referenztext anzeigen, 2106
Übersetzung in Projektsprachen, 2103
Projektversion
hochrüsten, 2183
Projektweite Meldeklasse
anlegen, 822
verwenden, 822
Projektweite Meldeklassen, 802
ProSave, 2429
asiatische Schriftarten, 575
Protokoll, 2202, 2250
Abschnitte ein- und ausblenden, 1307
anlegen, 1294
Aufgabe für automatischen Start eines Druckauftrags anlegen, 1315

Detailseite, 1283, 1286
Detailseite hinzufügen, 1306
Druckauftrag anlegen, 1312
drucken, 1307, 1334
ereignisgesteuerte Ausgabe, 1308
erstellen (Übersicht), 1288, 1291
Fußzeile, 1283, 1286
Größe und Position von Objekten, 1289
Größe von Objekten, 1292
Kopfzeile, 1283, 1286
maximal projektierbare Seitenzahl, 1289, 1294
Objekte mit dynamischer Länge, 1293
Position von Objekten, 1292
Projektierungsschritte für Ausgabe in Runtime Advanced, 1308
Projektierungsschritte für Ausgabe in Runtime Professional, 1310
Rückseite, 1283, 1286
Seite löschen, 1307
Seite sortieren, 1307
Titelseite, 1283, 1286
Titelseite und Rückseite, 1293
Wasserzeichen, 1286
zeitgesteuerte Ausgabe, 1309
Protokolle
CSV-Provider Kurven, 1341
Daten als Kurven ausgeben, 1297
Daten in Tabellenform ausgeben, 1297
Daten von Rezepturen ausgeben, 1298
Datum/Uhrzeit-Feld, 1342
Datum-Uhrzeit-Feld, 1343
E/A-Feld, 1345
Elemente, 1296
externe Daten über ODBC-Schnittstelle ausgeben, 1299
grafisches E/A-Feld, 1349
Hardcopy, 1350
Meldefolgeprotokoll, 1300
Meldeprotokoll, 1353
Meldungen ausgeben, 1298
ODBC-Datenbankfeld, 1355
ODBC-Datenbanktabelle, 1356
Projektname, 1358
Protokollname, 1359
Protokollobjekte, 1299
Rezepturprotokoll, 1361
Schnappschuss von Runtime ausgeben, 1298
Seitenzahl, 1362, 1363
symbolisches E/A-Feld, 1364
Protokollierung
aktivieren, 910, 911
Protokollname
Darstellung in Protokollen, 1359
Verwendung in Protokollen, 1359
Protokollobjekte
in Protokollen, 1299
Prozessdiagnose-Übersicht, 1937
Prozessgesteuerte Variable, 660
Prozesswert
Auslagerung, 674
darstellen, 719
Speicherung in Datenbank, 672
Prozesswertarchiv, 1093, 1108
Abfragen, 1093
Archivdaten abfragen, 1108
Daten darstellen, 1093
Recordset, 1093
Prozesswertarchivierung, 639
Prüfen
Geräteversion, 2188
Prüfsumme, 932
Archiv, 653, 931, 1076
Audit Trail, 653, 931
WinCC aktualisieren, 653, 931
Pufferüberlauf, 2202, 2250

Q

Quality Code
Dynamisierung, 233, 240
Querverweise
Einführung, 2086
Ersetzen, 2091
Nutzen, 2086
Umverdrahten, 2091
Quitterkonzept
Meldung mit Einfachquittierung, 806
Meldung ohne Quittierung, 806
Quittieranforderung, 811
Quittierbit, 1067
rücksetzen, 1067
QuittiereMeldung, 1690
quittieren,
Taste, 2240, 2321, 2410
Quittiergruppe
Text, 2202, 2250
Quittierkonzept, 806
Erstwertmeldung mit Blinken und Einfachquittierung, 806
Meldung mit Einfachquittierung, 806
Meldung mit Zweifachquittierung, 806
Meldung ohne Quittierung, 806

Meldung ohne Zustand Gegangen mit Quittierung, 806
Meldung ohne Zustand Gegangen ohne Quittierung, 806
Neuwertmeldung mit Blinken und Einfachquittierung, 807
Neuwertmeldung mit Blinken und Zweifachquittierung, 807
Quittierung, 804
projektieren, 913, 915
Quittiervariable, 1067

R

Radius, 417, 470, 472
Kreisbogen, 418
Kreissegment, 419
Radius horizontal, 377
Ellipsenbogen, 378
Ellipsensegment, 379
Radius vertikal, 377
Ellipsenbogen, 378
Ellipsensegment, 379
Rahmen gestalten, 119
Rand erreicht, 1820
Rechte Maustaste drücken, 1820
Rechte Maustaste loslassen, 1821
Rechteck
einfügen, 177
einfügen und konfigurieren, 176
konfigurieren, 178
Radius Ecken X, 476
Radius Ecken Y, 476
Recordset, 1093, 1101
Redundante Server und Partner, 2339
Redundanz
Abgleich, 2339
Ausfallszeitraum, 2339
Client-Umschaltung, 2339
Rezepturen, 2339
Reduzieren
Projektgröße, 2231, 2295
referenzieren
Objekt, 1455
Referenzobjekt
festlegen, 111, 1328
Referenzsprache, 2096
wählen, 2100
Reihenfolge
Sprachumschaltung, 2202, 2250, 2338
Reihenfolge der Spalten, 357
Reporting Services
Voraussetzungen, 1106
Reports mit Archivdaten
verfügbar im Netz, 1106
ResetBit, 1627, 1763
ResetBitInTag, 1629, 1764
Rezeptur, 1121, 1123, 1124, 1133, 1210, 1211, 2053
Anzeige in E/A-Feldern, 1233
Anzeige in Runtime, 1214
bearbeiten, 2053
Beispiel, 1212
Datenfluss, 1127
Datensatz, 1124, 1212
Einsatz, 1123, 1210
Einsatzbeispiel: Maschinenparametrierung, 1123
Einsatzbeispiel:Chargenorientierte Produktion, 1123
Element, 1124, 1212
Exportformat, 1694
exportieren, 1180, 2046, 2050
Grundlagen, 1121, 1123, 1210, 1211
importieren, 1180, 2048, 2051
Kommunikation über Rohdatenvariable, 1215
Kommunikation über Steuervariablen, 1215
Kommunikationsart Rohdatenvariable, 1242
Kommunikationsart Variablen, 1242
Konfigurationsmöglichkeit, 1130
Laden, 1222
neu anlegen, 1185, 1222
Rezepturvariable abgleichen, 1162, 1176
Speicherbedarf, 2452, 2453
Tabellenansicht in Runtime, 1232
Verwendung von Textlisten, 1137
Rezeptur bedienen
Datensatz lesen, 1152, 1163, 1178
Datensatz übertragen, 1153, 1164, 1179
Rezeptstruktur ändern, 1181, 1228
Rezepturdatensatz ändern, 1162, 1175, 1176
Rezepturdatensatz erstellen, 1150, 1151, 1161, 1175
Rezepturdatensatz kopieren, 1161, 1175
Rezepturdatensatz laden, 1162, 1175, 1176
Rezepturdatensatz löschen, 1162, 1176
Rezeptur importieren
Struktur für den Import, 2053
Rezepturabfrage
Einschränkungen in Runtime, 1230
neu anlegen, 1230
Rezepturanzeige, 477, 483, 1154, 1165, 1215, 1232
Aktualisierung, 1156, 1168
als Auswahlfeld verwenden, 1158, 1169

Bedienelement, 477, 485, 1166
Bedienelemente, 480
Bedienobjekt, 1155
Bedienung, 1155, 1166
Bedienung mit Maus, 1159, 1171
Bedienung mit Tastatur, 1159, 1171
Beschriftung, 481
Darstellung, 477, 480, 482
Daten bearbeiten, 1262, 1268
dynamisierbare Eigenschaften, 1148
einfache Rezepturanzeige, 484
erweitert, 1154, 1165
Export von Runtime-Daten projektieren, 737, 750, 759
mit Funktionstaste bedienen, 1182
mit Funktionstasten bedienen, 1150, 1159, 1171
Nummer anzeigen, 481, 485
nur eine Rezeptur anzeigen, 1157, 1168
Online-Projektierung, 738, 751, 760
projektierbare Ereignisse, 1147, 1158, 1170
projektieren, 1195, 1198, 1236
Rezepturdatensatz, 1125, 1126, 1183
Schaltflächen, 1259
Spaltenüberschriften, 481
Statuszeile, 736, 749, 756, 1259
Symbolleiste, 483, 736, 749, 756, 1259
UserArchiveControl, 1262
Verhalten bei Bildwechsel, 1160, 1174, 1182
Verwendung, 482, 1154, 1165
Verwendung in Protokollen, 1360
Werte nur anzeigen, 1146
RezepturanzeigeDatensatzNeu, 1707
RezepturanzeigeLeseDatensatzAusSteuerung, 1707
RezepturanzeigeLöscheDatensatz, 1708
RezepturanzeigeMenü, 1708
RezepturanzeigeÖffnen, 1709
RezepturanzeigeSchreibeDatensatzInSteuerung, 1710
RezepturanzeigeSpeichereDatensatz, 1710
RezepturanzeigeSpeichernUnterDatensatz, 1711
RezepturanzeigeSynchronisiereDatensatzMitVariablen, 1711
RezepturanzeigeUmbenennenDatensatz, 1712
RezepturanzeigeZeigeHilfetext, 1712
RezepturanzeigeZurück, 1712
Rezepturbedienung in Runtime
Zugriffsschutz bei Steuervariablen, 1243
Rezepturbild, 1139, 1140
automatisch übertragen, 1140
projektieren, 1200
Rezepturvariable, 1126
Rezepturwert abgleichen, 1139
visuelle Nachbildung der Maschine, 1139
Rezepturdaten
in Protokollen, 1298
Rezepturdatensatz
abgleichen, 1162, 1176
am Bediengerät erstellen, 1150, 1151, 1161, 1175
ändern, 1162, 1175, 1176
Exportformat, 1694
exportieren, 1180
importieren, 1180
importieren und exportieren, 1130
kopieren, 1161, 1175
laden, 1162, 1175, 1176
löschen, 1162, 1176
neu anlegen, 1185, 1222
Übertragungsmöglichkeit, 1127
Verwendung von Textlisten, 1137
Rezepturdatensatz umbenennen, 1176
Rezepturdatensatzname
in Variable schreiben, 1147, 1157, 1169
Rezepturdatensatznummer
in Variable schreiben, 1147, 1157, 1169
Rezepturelement
neu anlegen, 1185, 1222
Rezepturen
Aufbau Auftragskopf der Rohdatenvariable, 1250
Aufbau Quittungskopf der Rohdatenvariable, 1251
Aufbau Telegrammkopf der Rohdatenvariable, 1249
Auftragstypen und Fehlercodes bei Rohdatenkommunikation, 1247
CSV-Datei aus ES exportieren, 1194, 1228
CSV-Datei in ES importieren, 1192, 1226
Kommunikation über Rohdatenvariablen, 1248, 1250
Quittungsdaten der Rohdatenvariable, 1251
Quittungstelegrammaufbau Rohdatenvariable, 1250
Redundanz, 2339
Telegrammaufbau Rohdatenvariable, 1248
Rezepturname
in Variable schreiben, 1147, 1157, 1169
Rezepturnummer
in Variable schreiben, 1147, 1157, 1169
Rezepturprotokoll
Darstellung in Protokollen, 1361
Verwendung in Protokollen, 1361
Rezepturstruktur ändern, 1181, 1228

Rezepturvariable
abgleichen, 1131, 1162, 1176
Teach-In-betrieb, 1132
Rezepturwert, 1139, 1140
abgleichen im Rezepturbild, 1139
automatisch übertragen im Rezepturbild, 1140
Rezepturwerte übertragen
zwischen Bediengerät und Steuerung, 1242
RFID Tag, 1837
Rohdatentyp, 660
Rohdatenvariable, 660, 1252
Aufbau Auftragskopf, 1250
Aufbau Quittungskopf, 1251
Aufbau Telegrammkopf, 1249
für BSEND/BRCV-Funktionen, 1254
Kommunikation zwischen Rezeptur und Steuerung, 1215
Quittungsdaten, 1251
Quittungstelegramm, 1250
Rohdatenvariablen, 1247
Rohr, 487
Rohrbogen, 487
Radius, 488
Winkel, 488
Rückladen
Meldungen, 948
Rückmeldung
optisch, 2400
Rückseite
in Protokollen, 1293
Protokoll, 1283, 1286
RücksetzeBit, 1627, 1763
RücksetzeBitInVariable, 1629, 1764
Rücksetzen
auf Werkseinstellungen, 2433
Quittierbit, 1067
RulerControl, 524
Online-Projektierung, 1240
Runtime-Daten exportieren, 1239
Runtime, 1845, 1847, 2346, 2381, 2382
abarbeiten von benutzerdefinierten Funktionen, 1499
Abarbeitung der Funktionsliste, 1500
archivierte Meldungen anzeigen, 975
auf Panel starten, 2227, 2291
auf PC starten, 2291
Filtern von Meldungen, 971
Mehrplatzsystem, 2397
Notquittierung von Meldungen, 982
Objekteigenschaft ändern, 1504
projektieren in Runtime, (Siehe Online-Projektierung)
redundantes System, 2397
Schaltflächen, 1259
simulieren, 2211
Sortieren von Meldungen, 973
Sprachumschaltung, 1504
Startbild, 46, 47
starten, 2397
Voraussetzungen, 2397
Wirkbereich, 1845
Wirkbereich (RFID), 1847
Runtime Professional
Anlaufliste einrichten, 2349
Cusorsteurung einrichten, 2341
Hotkeys festlegen, 2343
Zoomfunktionen einrichten, 2347
Runtime Scripting, 1431
Benutzerdefinierte Funktionen, 1431
Funktionen, 1431
Lokales Skript, 1431
Systemfunktionen, 1431
Runtime-Bildschirm
in Protokollen, 1298
Runtime-Einstellungen, 2398
Benutzerverwaltung, 46, 48, 1374, 1377, 1381, 1391, 1392, 1397, 1398, 1408, 1409
Codepage, 1469
Dynamisches Anmelden, 1381
Runtime-Projekt
Aktualisieren, 2386
Runtimes
Unterschiede, 2149
Runtime-Sprache, 2096, 2114
Archiv, 655, 933, 2120
auswählen, 2116
nicht-multilinguale Objekte, 2338
Reihenfolge bei Sprachumschaltung, 2117
Schriftart, 2119
Runtime-Sprachen, 2202, 2250, 2338
Runtime-Start
Basic Panel, 2226
Runtime-Stopp, 1821

S

S7 1200 V2
Datentypen, 632, 634
S7 1500
Datentypen, 632
S7 300/400
Datentypen, 631

S7 Protocol Suite
Rohdatenvariable, 1252
Rohdatenvariable für BSEND/BRCV-Funktionen, 1254
S7-Diagnosemeldung
aktivieren, 839
S7-Diagnosemeldungen, 797
S7-Routing
Laden, 2218
SafelyRemoveHardware, 1579
SaveDataRecord, 1716
Schalter, 488
Typ, 489
Schaltfläche, 489
als Freigabeschaltfläche, 2318
als Freigabeschaltfläche projektieren, 2407
anlegen, 324
für Freigabe Bedienelement, 2315, 2405
Grafik, 490
Hotkey festlegen, 491
HTML-Browser, 404
Modus, 490
PDF-Anzeige, 460
projektieren, 1424
Rezepturanzeige, 1259
sperren/entsperren, 2316, 2406
System-Diagnoseindikator einfügen, 1912
Text, 490
Zugriffsschutz projektieren, 1420
Schaltfläche Meldeanzeige klicken, 1821
Schaltfläche PLC-Code-Anzeige klicken, 1822
Schaltfläche Symbolleiste klicken, 1822
SchaltflächeDrücken, 1585
SchaltflächeLoslassen, 1586
Schieber ändern, 1822
Schieberegler, 491
aktuellen Wert anzeigen, 493
Balken anzeigen, 493
maximaler Wert, 492
minimaler Wert, 492
sperren/entsperren, 2316, 2406
SchließeAlleArchive, 1609
Schlüsselpaar für Kennwortschutz, 2206, 2256
Schlüsselschalter
Variable, 494
Verwendung, 493
SchreibeDatensatzInSteuerung, 1713
SchreibeDatensatzvariablenInSteuerung, 1714
Schriftgröße
asiatische Sprachen, 2122
geräteabhängige, 82
Schriftgrößenanpassung
aktivieren, 81
SchrittVor, 1823
SchrittZurück, 1823
schützen
Funktionstaste mit Kennwort, 255
scrollen
Multitouchgeräte, 2312
Segment, 672
Segmente
für Archive festlegen, 688
Segmentiertes Umlaufarchiv, 922
Seite
maximal projektierbare ~nanzahl im Protokoll, 1289
Protokollseite löschen, 1307
Protokollseite sortieren, 1307
zu Protokoll hinzufügen, 1306
SeiteAbwärts, 1724
SeiteAufwärts, 1724
Seitenzahl
Darstellung in Protokollen, 1362, 1363
Verwendung in Protokollen, 1362, 1363
Selektion
Selektionsdialog, 1266
SendeEMail, 1584
SendEMail, 1584
Serverauslastung, 2338
Serverauswahl
Process Historian, 2339
Service-Mode, 2357
Service-Projekt
Diagnoseinformationen, 2359
Einsatz, 2358
Einschränkung, 2359
Konfiguration, 2358
konfigurieren, 2360
OPC-Zugriff, 2359
Skript, 2359
Task, 2359
SetAccessModeViaWeb, 1647
SetAcousticSignal, 1653
SetBit, 1633, 1766
SetBitInTag, 1634, 1767
SetBrightness, 1653
SetConnectionMode, 1646
SetDataRecordTagsToPLC, 1714
SetDataRecordToPLC, 1713
SetDeviceMode, 1642
SetDisplayMode, 1641
SetLanguage, 1645, 1770
SetMessageLogMode, 1639

SetPLCMode, 1643, 1644
SetPropertyByConstant, 1751
SetPropertyByProperty, 1749
SetPropertyByTag, 1753
SetPropertyByTagIndirect, 1755
SetRecipeTags, 1715
SetScreenKeyboardMode, 1642, 1771
SetTagByProperty, 1736
SetTagIndirect, 1737
SetTagIndirectByProperty, 1738, 1740
SetTagIndirectByTagIndirect, 1739
SetTagIndirectWithOperatorInputAlarm, 1741
SetzeAkustischesSignal, 1653
SetzeAnzeigemodus, 1641
SetzeBetriebsartGerät, 1642
SetzeBit, 1633, 1766
SetzeBitInVariable, 1634, 1767
SetzeBitWährendTasteGedrückt, 1637
SetzeEigenschaftAktuellesFenster, 1757
SetzeEigenschaftAktuellesFensterDurchEigenschaft, 1757
SetzeEigenschaftAktuellesFensterDurchVariableIndirekt, 1758
SetzeEigenschaftDurchEigenschaft, 1749
SetzeEigenschaftDurchKonstante, 1751
SetzeEigenschaftDurchVariable, 1753
SetzeEigenschaftDurchVariableIndirekt, 1755
SetzeFarbeHintergrundbeleuchtung, 1654
SetzeHelligkeit, 1653
SetzeModusBildschirmtastatur, 1642, 1771
setzen
    Stil, 126
SetzePLCModus, 1643, 1644
SetzeProtokolliermodusMeldung, 1639
SetzeRezepturvariablen, 1715
SetzeSommerzeit, 1585
SetzeSprache, 1645, 1770
SetzeVariable, 1621, 1734
SetzeVariableIndirekt, 1737
SetzeVariableIndirektDurchEigenschaft, 1738, 1740
SetzeVariableIndirektDurchVariableIndirekt, 1739
SetzeVariableIndirektMitBedienprotokollierung, 1741
SetzeVerbindungsmodus, 1646
SetzeWebZugriff, 1647
SetzteVariableDurchEigenschaft, 1736
SHIFT
    Taste, 2240, 2321, 2410
ShowAlarmWindow, 1685, 1691
ShowLogonDialog, 1729
ShowSoftwareVersion, 1598
ShowSystemAlarm, 1691
SichereRAMDateisystem, 1587
Sicherheitshinweis
    Direkttaste, 2322, 2411
Sichern
    Daten des Bediengeräts, 2429, 2431
Sicht
    Zweck, 1211
sichtbare Spalte, 356
Signalqualität
    des Wirkbereichs berechnen, 530
Signierung Archiv-Backup-Datei, 693, 942
SIMATIC
    Datenformatunterschiede zu Microsoft, 1246
SIMATIC Logon, 1393, 2202, 2250, 2338
    verschlüsselte Kommunikation, 1395
    Zertifikat, 1395
SIMATIC Logon, 1381
Simulation
    Projekt:simulieren, 2368
    Runtime, 2211
Simulationsdaten, 2369
simulieren, 2368
    Projekt über Variablen-Simulator, 2213, 2263
    temporäres Startbild festlegen, 2265, 2370
SimuliereSystemtaste, 1587
SimuliereVariable, 1588
skalieren
    Variable linear skalieren, 572, 573
Skript
    Bibliothek, 2041
    Druckauftrag/Skriptdiagnose, 365
    Service-Projekt, 2359
    Typ erzeugen, 2041
    Variablenwert aktualisieren, 542
Skript anlegen
    Bildbaustein-Typ, 307
Skripte in Runtime, 2337
Slide-in-Bild
    konfigurieren, 62
    konfigurieren für Geräte ohne Multitouch-Funktion, 64
    Objekte, 61
    verwenden, 65
Sm@rtClient-Anzeige, 497, 498
    gemeinsam genutzt, 498
    nur anzeigen, 499
SmartClientAnzeigeAktualisieren, 1679
SmartClientAnzeigeSchreibschutzAus, 1680
SmartClientAnzeigeSchreibschutzEin, 1680
SmartClientAnzeigeTrennen, 1681
SmartClientAnzeigeVerbinden, 1681
SmartClientAnzeigeVerlassen, 1682
SmartServer, 1592

Softkey, 248, 252
global, 250
Software
Bediengerät, 2426, 2427, 2428
Sortieren
Protokollseite, 1307
Sortieren von Meldungen, 973
Sortierung
einrichten, 439
Meldeanzeige, 443
Sortier-Dialog, 1268
Sortierung freigeben
Meldeanzeige, 450
Sortierung projektieren
Kurzzeitarchiv, 906
Langzeitarchiv, 906
Spalenbreite
Benutzeranzeige, 352
Spalte
Meldeanzeige, 440, 446, 450
Spalten festlegen
Meldeanzeige, 443
Spalten verschieben
Benutzeranzeige, 351
Spaltenreihenfolge
Meldeanzeige, 443
Spaltenüberschriften
Meldeanzeige, 444
Rezepturanzeige, 481
Speicherbedarf
Rezeptur, 2452, 2453
SpeichereDatensatz, 1716
Speichergröße
Meldepuffer, 927
Speichermedium
wiederherstellen von externem, 2281, 2436
speichern
globale Bibliothek, 2027
Sperrbit, 1068
Sperre geändert, 1823
Sperren
Bedienelement, 2315, 2316, 2405, 2406
Sperrliste, 974
Sperrvariable, 1068
Meldegruppe, 1068
Spezifisches .Net-Control
entfernen, 148
hinzufügen, 148
spiegeln, 502
Objekt, 103
Spitzenwert anzeigen
Zeigerinstrument, 533
Sprache
Archiv, 655, 933, 2120
asiatische Sprachen, 2098
asiatisches Betriebssystem, 2098
Editiersprache, 2100
Grafik sprachabhängig, 2109
mehrsprachiges Projekt, 2102
Projektsprache aktivieren, 2099
Projektsprache deaktivieren, 2100
Referenzsprache, 2100
sprachabhängige Darstellung, 2097
sprachabhängige Darstellung bei Datum, Uhrzeit, Geldbeträgen und Zahlen, 2097
Sprachunterstützung, 2098
Sprache einstellen, 2245, 2327, 2415
Sprache und Schriftart, 2202, 2250, 2338
Sprachumschaltung, 2115
Archiv, 655, 933, 2120
in Runtime, 1504
Reihenfolge, 2202, 2250, 2338
Runtime-Sprache, 2117
SQL, 1218
SQL Server Import/Export-Wizard, 1105
SQL-Abfragen
auf WinCC-Datenbanken, 1105
SQL-Statements für Filterung von Meldungen, 1000
Standardschrift, 2202, 2250, 2338
Standardschriftart, 2119
Standard-SQL-Abfragen
auf WinCC-Datenbanken, 1105
Standardstil, 2201, 2249
festlegen, 80
Standardvorlage, 2201, 2249
ständig lesen
Variable, 589
Start
Archivierung, 658
Startbild, 46, 47, 313, 2201, 2249, 2337
StarteArchivierung, 1609
starten
Runtime an Engineering-Station, 2289, 2396
Runtime an Projektierungs-PC, 2289, 2396
Runtime auf Panel, 2227, 2291
Runtime auf PC, 2291
StarteNächstesArchiv, 1610
StarteProgramm, 1589
StartLogging, 1609
StartProgram, 1589
Startpunkt
festlegen, 621
Startpunkt der Archivierung, 665
StartSequenceLog, 1610

Startsequenz, 2338
Startwert
Variable, 570
Statischer Zeitbereich, 724
Status gändert, 1824
Statusanzeige
Kanaldiagnose-Anzeige, 414
StatusSteuernLeseWerte, 1591
StatusSteuernSchreibeWerte, 1592
Statuszeile, 438, 736, 749, 756
Kurvenanzeige, 782
Rezepturanzeige, 1259
Tabellenanzeige, 782
stempeln, 139, 1322
SteuereWebServer, 1593
Steuerung
Systemdiagnose aktivieren, 1908, 1920
System-Diagnoseanzeige, 1908, 1920
Variable, 571
Steuerungsdaten
initialisieren, 2196
Steuerungsmeldung, 791, 793, 795
systemdefinierte, 797
Steuerungsmeldungen, 2250, 2338
auf das Bediengerät übertragen, 991
Projektieren, 816, 817
systemdefinierte, 792, 793
Steuervariable
Kommunikation zwischen Rezeptur und Steuerung, 1215
Steuervariablen
Beispiel, 1276
Stil
ändern, 80
bearbeiten, 80
Bediengerät, 2201, 2249
definieren, 77
duplizieren, 78
löschen, 79
neuen Typen erstellen, 2043
setzen, 126
StopLogging, 1611
Stopp, 1824
StoppeArchivierung, 1611
StoppeRuntime, 1593, 1726
StopRuntime, 1593, 1726
Stromausfall
Meldelisteneinträge, 2338
Suchen und Ersetzen
Farben, 2189
Symbol
Meldeklassen, 871
Symbolbibliothek, 501
drehen, 502
festes Seitenverhältnis, 502
Füllart, 502
OP 177B mono, 502
spiegeln, 502
TP 177B mono, 502
Symbole
ProDiag-Übersicht, 1944
Symbolische Adressierung
einer Variable, 546
Symbolisches E/A-Feld, 503, 505
Modus, 503, 505
Textliste, 191, 193, 504, 505
Verwendung in Protokollen,
Symbolleiste, 382, 385, 389, 517, 736, 749, 756
anlegen, 323
benutzerdefiniert, 316, 329
Ereignis projektieren, 325
PDF-Anzeige, 458
PLC-Code-Anzeige, 464, 468, 1959
Reihenfolge, 103, 115, 1325
Rezepturanzeige, 1259
Schaltfläche mit Bildwechsel, 333
Systemfunktion projektieren, 333
Tabellenanzeige, 727
Symbolleiste umschalten, 1824
Symboltabelle
PLC-Code-Anzeige, 1956
Synchronisieren, 541, 543, 564
Objekt in benutzerdefinierten Funktionen,
Variable in benutzerdefinierten Funktionen,
Syntaxhervorhebung
benutzerdefinierte Funktion, 1446
Systemblock, 997
Projektieren, 841
Systemblöcke, 840
Systemdefinierte Steuerungsmeldung, 797
Systemdefinierte Steuerungsmeldungen, 792, 793
Systemdiagnose, 1901, 1904, 1917
Detailansicht, 1901, 1905, 1918
Diagnosepuffer-Ansicht, 1902, 1905, 1918
Einstellung in der Steuerung, 1908, 1920
Geräteansicht, 1901, 1905, 1918
Schaltfläche, 1912
System-Diagnoseanzeige, 1901, 1904, 1917
System-Diagnosefenster, 1904, 1911, 1917
System-Diagnoseaneige
Symbol, 510, 512
Systemdiagnose-Anzeige
Anzeige teilen, 510, 512, 1915, 1923

Darstellung, 508, 510
Einfache Anzeige, 512
System-Diagnoseanzeige, 1904
Detailansicht, 507
Diagnosepuffer-Ansicht, 507
Einstellung in der Steuerung, 1908, 1920
Geräteansicht, 507
projektieren, 1913, 1914, 1915, 1922
SystemDiagnoseanzeigeAktivieren, 1682
SystemDiagnoseanzeigeAktualisierePLCBuffer, 1683
SystemDiagnoseanzeigeDetailansicht, 1683
SystemDiagnoseanzeigeDiagnosepuffer, 1684
SystemDiagnoseanzeigeGerätesicht, 1684
SystemDiagnoseanzeigeZurück, 1685
System-Diagnosefenster, 1904, 1911
Darstellung, 514
projektieren, 1911
System-Diagnoseindikator, 1917
einfügen, 1909, 1927
Schaltfläche als System-Diagnoseindikator, 1912
System-Diagnosefenster, 1909, 1927
Systemdiagnosemeldung, 1909, 1921
System-Diagnosenazeige
System-Diagnoseindikator, 1909, 1927
Systemdruckauftrag
ändern, 1317
löschen, 1318
Systemfunktion, 1431
Bediengeräteabhängigkeit, 1448
DirekttasteBildnummer, 1577
Einsatzmöglichkeit, 1433
HardwareSicherEntfernen, 1579
HTMLBrowserNachLinksScrollen, 1658
in benutzerdefinierten Funktionen, 1433, 1500
in Funktionsliste, 1433
Sprachabhängigkeit in Funktionslisten, 1435
Verwendung, 1434
Systemfunktionen
Abmelden, 1612
AktiviereBild, 1569, 1744
AktiviereBildInAktuellemBildfenster, 1745
AktiviereBildInBildfenster, 1746
AktiviereBildMitNummer, 1570
AktivierePLCCodeAnzeige, 1673
AktivierePutzbild, 1717
AktiviereVorherigesBild, 1571
AktualisiereVariable, 1572
ÄndereKontrast, 1651
Anmelden, 1613
AnzeigenSlideInBild, 1602
ArchiviereProtokolldatei, 1603
ArchiviereVariable, 1605
BearbeiteMeldung, 1686
BildobjektCursorAbwärts, 1721
BildobjektCursorAufwärts, 1721
BildobjektCursorLinks, 1723
BildobjektCursorRechts, 1722
BildobjektSeiteAbwärts, 1722
BildobjektSeiteAufwärts, 1723
Codieren, 1572
CodierenEx, 1573
Direkttaste, 1574
DruckeBild, 1638
DruckeProtokoll, 1639
ErfasseBenutzeraktion, 1577
ErhöheFokussiertenWert, 1579
ErhöheVariable, 1618, 1730
ExportiereDatensätze, 1692
ExportiereDatensätzeMitPrüfsumme, 1694
ExportiereImportiereBenutzerverwaltung, 1613, 1728
GeheZumAnfang, 1725
GeheZumEnde, 1725
HTMLBrowserAbbrechen, 1655
HTMLBrowserAbwärtsScrollen, 1656
HTMLBrowserAktualisieren, 1656
HTMLBrowserAufwärtsScrollen, 1657
HTMLBrowserHeranzoomen, 1657
HTMLBrowserHerauszoomen, 1658
HTMLBrowserNachRechtsScrollen, 1658
HTMLBrowserSeiteNachOben, 1659
HTMLBrowserSeiteNachUnten, 1659
HTMLBrowserStartseite, 1660
HTMLBrowserVorwärts, 1660
HTMLBrowserZurück, 1661
ImportiereDatensaetze, 1697
ImportiereDatensätzeMitPrüfsumme, 1698
InvertiereBit, 1624, 1759
InvertiereBitInVariable, 1625, 1761
InvertiereLineareSkalierung, 1619, 1731
KalibriereTouchscreen, 1652
KurvenanzeigeBlättereVorwärts, 1661
KurvenanzeigeBlättereZurück, 1662
KurvenanzeigeErweitern, 1662
KurvenanzeigeKomprimieren, 1662
KurvenanzeigeLinealLinks, 1664
KurvenanzeigeLinealRechts, 1663
KurvenanzeigeSetzeLinealmodus, 1664
KurvenanzeigeStartenStoppen, 1665
KurvenanzeigeZurückZumAnfang, 1665
LadeDatensatz, 1700
LeseBenutzername, 1615
LeseDatensatzAusSteuerung, 1701

LeseDatensatzname, 1702
LeseDatensatzvariablenAusSteuerung, 1704
LeseGruppennummer, 1615
LeseHelligkeit, 1655
LeseKennwort, 1616
LineareSkalierung, 1620, 1733
LoescheArchiv, 1607
LöscheDatensatz, 1705
LöscheDatensatzSpeicher, 1706
LöscheMeldepuffer, 1686
LöscheMeldepufferProTool, 1687
MeldeanzeigeAktualisieren, 1688
MeldeanzeigeBearbeiteMeldung, 1688
MeldeanzeigeQuittiereMeldung, 1689
MeldeanzeigeZeigeHilfetext, 1690
ÖffneAlleArchive, 1608
ÖffneBildschirmtastatur, 1720
ÖffneDialogSystemsteuerung, 1580
ÖffneEingabeaufforderung, 1581
ÖffneInternetExplorer, 1581
ÖffneSystemsteuerung, 1582
ÖffneTaskManager, 1719
PDFAbwärtsScrollen, 1666
PDFAnpassenAufBreite, 1667
PDFAnpassenAufHöhe, 1667
PDFAufwärtsScrollen, 1666
PDFGeheZurErstenSeite, 1668
PDFGeheZurLetztenSeite, 1668
PDFGeheZurNächstenSeite, 1669
PDFGeheZurVorherigenSeite, 1670
PDFGeheZuSeite, 1669
PDFHeranzoomen, 1670
PDFHerauszoomen, 1670
PDFLinksScrollen, 1671
PDFRechtsScrollen, 1671
PDFZoomOriginal, 1672
PLCCodeAnzeigeAktualOderErstwerte, 1677
PLCCodeAnzeigeBildlaufleisteVerschieben, 1674
PLCCodeAnzeigeDetails, 1675
PLCCodeAnzeigeHeranzoomen, 1678
PLCCodeAnzeigeHerauszoomen, 1678
PLCCodeAnzeigeNächstesNetzwerk, 1675
PLCCodeAnzeigeSymbole, 1676
PLCCodeAnzeigeTransitionInterlock, 1677
PLCCodeAnzeigeVorherigesNetzwerk, 1679
PLCCodeAnzeigeZeigeSchrittModus, 1676
PROFIsafeTrennen, 1583
QuittiereMeldung, 1690
RezepturanzeigeDatensatzNeu, 1707
RezepturanzeigeLeseDatensatzAusSteuerung, 1707
RezepturanzeigeLöscheDatensatz, 1708
RezepturanzeigeMenü, 1708
RezepturanzeigeÖffnen, 1709
RezepturanzeigeSchreibeDatensatzInSteuerung, 1710
RezepturanzeigeSpeichereDatensatz, 1710
RezepturanzeigeSpeichernUnterDatensatz, 1711
RezepturanzeigeSynchronisiereDatensatzMitVariablen, 1711
RezepturanzeigeUmbenennenDatensatz, 1712
RezepturanzeigeZeigeHilfetext, 1712
RezepturanzeigeZurück, 1712
RücksetzeBit, 1627, 1763
RücksetzeBitInVariable, 1629, 1764
SchaltflächeDrücken, 1585
SchaltflächeLoslassen, 1586
SchließeAlleArchive, 1609
SchreibeDatensatzInSteuerung, 1713
SchreibeDatensatzvariablenInSteuerung, 1714
SeiteAbwärts, 1724
SeiteAufwärts, 1724
SendeEMail, 1584
SetzeAkustischesSignal, 1653
SetzeAnzeigemodus, 1641
SetzeBetriebsartGerät, 1642
SetzeBit, 1633, 1766
SetzeBitInVariable, 1634, 1767
SetzeBitWährendTasteGedrückt, 1637
SetzeEigenschaftAktuellesFenster, 1757
SetzeEigenschaftAktuellesFensterDurchEigenschaft, 1757
SetzeEigenschaftAktuellesFensterDurchVariableIndirekt, 1758
SetzeEigenschaftDurchEigenschaft, 1749
SetzeEigenschaftDurchKonstante, 1751
SetzeEigenschaftDurchVariable, 1753
SetzeEigenschaftDurchVariableIndirekt, 1755
SetzeHelligkeit, 1653
SetzeModusBildschirmtastatur, 1642, 1771
SetzeProtokolliermodusMeldung, 1639
SetzeRezepturvariablen, 1715
SetzeSommerzeit, 1585
SetzeSprache, 1645, 1770
SetzeVariable, 1621, 1734
SetzeVariableIndirekt, 1737
SetzeVariableIndirektDurchEigenschaft, 1738, 1740
SetzeVariableIndirektDurchVariableIndirekt, 1739
SetzeVariableMitBedienprotokollierung, 1741
SetzeVerbindungsmodus, 1646
SetzeWebZugriff, 1647
SetzteVariableDurchEigenschaft, 1736
SichereRAMDateisystem, 1587

SimuliereSystemtaste, 1587
SimuliereVariable, 1588
SmartClientAnzeigeAktualisieren, 1679
SmartClientAnzeigeSchreibschutzAus, 1680
SmartClientAnzeigeSchreibschutzEin, 1680
SmartClientAnzeigeTrennen, 1681
SmartClientAnzeigeVerbinden, 1681
SmartClientAnzeigeVerlassen, 1682
SpeichereDatensatz, 1716
StarteArchivierung, 1609
StarteNächstesArchiv, 1610
StarteProgramm, 1589
StatusSteuernLeseWerte, 1591
StatusSteuernSchreibeWerte, 1592
SteuereSmartServer, 1592
SteuereWebServer, 1593
StoppeArchivierung, 1611
StoppeRuntime, 1593, 1726
SystemDiagnoseanzeigeAktivieren, 1682
SystemDiagnoseanzeigeAktualisierePLCBuffer, 1683
SystemDiagnoseanzeigeDetailansicht, 1683
SystemDiagnoseanzeigeDiagnosepuffer, 1684
SystemDiagnoseanzeigeGerätesicht, 1684
SystemDiagnoseanzeigeZurück, 1685
TextAbrufen, 1594, 1727
TrenneVariableVonHandrad, 1595
VerbindeVariableMitHandrad, 1596
VerfolgeBenutzeränderung, 1617
verfügbar auf Comfort Panels, 1547
verfügbar auf Mobile Panels, 1554
verfügbar bei WinCC Runtime, 1562, 1567
VerringereFokussiertenWert, 1596
VerringereVariable, 1623, 1742
WechseleVerbindung, 1648
WechseleVerbindungEIP, 1649
ZeigeAnmeldedialog, 1617, 1729
ZeigeBlockInTIAPortalVonMeldung, 1772
ZeigeHilfetext, 1597
ZeigeMeldefenster, 1691
ZeigePLCCodeAzeigeVonMeldung, 1773
ZeigePopupBild, 1598, 1600
ZeigeSoftwareVersion, 1598
ZeigeSystemDiagnosefenster, 1685
ZeigeSystemmeldung, 1691
Systemgrenzen
Bediengerät, 2455, 2458, 2462, 2468
Systemmeldung, 791, 792, 793, 796, 797
bearbeiten, 836, 859
Bedeutung, 1003, 1004, 1005, 1006, 1007, 1010, 1013, 1014, 1015, 1016, 1017, 1020, 1021, 1022, 1024, 1025, 1026, 1027, 1030, 1031, 1034, 1035, 1041, 1042, 1043, 1045, 1046, 1047, 1048, 1051, 1052, 1053, 1054, 1055, 1056, 1057, 1060, 1062, 1063
Parameter, 1002
Systemmeldungen, 2202, 2250
projektieren, 816, 817
Systemprotokoll
ändern, 1317
löschen, 1318
Systemzeit anzeigen, 363
Sytemfunktion
Reihenfolge wird nicht eingehalten, 1502

T

Tabelle
Tabellenanzeige, (Siehe Tabellenanzeige)
Tabellenelemente projektieren, 744, 758
Tabelle:Rahmen gestalten, 132
Tabellen
in Protokollen, 1297
Tabellenanzeige, 515, 719
Bedienung in Runtime, 761, 765
Datenanbindung in Runtime ändern, 775
Datenquelle konfigurieren, 516
Datenversorgung projektieren, 734, 747
Elemente der Statuszeile, 782
Export von Runtime-Daten projektieren, 737, 750, 759
in archivierten Daten blätern, 781
Konfigurationsdialog in Runtime aufrufen, 775
Markierung konfigurieren, 745
Online-Projektierung, 738, 751, 760, 775, 779
projektieren, 741
Schaltflächen, 761
Spalte in Runtime ausblenden, 776
Spalte in Runtime einblenden, 776
Spalten in Runtime ausblenden, 778
Spalten in Runtime einblenden, 778
Spalten in Runtime verschieben, 778
Statuszeile, 736, 749, 756
Symbolleiste, 517, 727, 736, 749, 756, 761
Tabellenfeld in Runtime bearbeiten, 777
Verwendung in Protokollen, 1365
Wertespalte dynamisieren, 518
Wertspalte projektieren“, 743
Zeitbereich in Runtime ändern, 776
Tabellenspalte
Sortierung projektieren, 746, 759

TAB-Taste, 2238, 2319, 2408
Tabulator-Taste, 2238, 2319, 2408
Tag
R, 1093
RFID Tag, 1834
Transponder, 1834
Task
Service-Projekt, 2359
Task Card, 38
Werkzeuge, 91, 95, 100
Taskumschaltung, 2202, 2250
Tastatur-Befehle des Betriebssystems
aktivieren, 2343
Tastaturbelegung, 2202, 2250, 2338
Taste
Backspace, 2239, 2320, 2409
Cursor, 2238, 2319, 2408
END, 2239, 2320, 2409
ENTER, 2239, 2320, 2409
ESC, 2239, 2320, 2409
Hilfetext, 2240, 2321, 2410
HOME, 2239, 2320, 2409
Löschen, 2239, 2320, 2409
TAB, 2238, 2319, 2408
Weiterblättern, 2239, 2320, 2409
Zurückblättern, 2239, 2320, 2409
Taste auf Tastatur drücken, 1826
Taste auf Tastatur loslassen, 1826
Taste drücken, 1825
Taste loslassen, 1825
Tasten, 2346
Tasten-Bedienung, 2238, 2319, 2408
Tastenkombination
Objekte verschieben, 114
Tastenkombinationen, 2202, 2250, 2338
festlegen, 2344
Tauschen
Bediengerät, 2137
Teach-In-Betrieb, 1132
Rezepturvariable, 1132
Telegrammvariable, 660
Template, 248
TerminatePROFIsafe, 1583
testen,
Benutzerdefinierte Funktion, 1453
Testen:mit Debugger, 1485
Text
Editierbares Textfeld, 377
Quittiergruppe, 2202, 2250
Schaltfläche, 490
Textfeld, 519, 1366
Textformat festlegen, 439
Text erstellen
C-Textliste, 209
TextAbrufen, 1594, 1727
Textfeld, 519, 1365
Größe, 519, 1366
Textfeldlänge
asiatische Sprachen, 2122
Textliste
Bereich (... - ...), 186
Bit (0, 1), 188
Bitnummer (0 - 31), 190
erstellen, 182
exportieren, 2078
importieren, 2080
in Meldung ausgeben, 835
Symbolisches E/A-Feld, 191, 193, 504, 505
Verwendung, 180
Wert/Bereich, 184
Textlisten
Bitselektion, 2201, 2249
Verwendung in Rezepturdatensätzen, 1137
Textuelle Querverweise
anzeigen, 2087
VB-Scripting, 2088
Timer, 1876
azyklischer Trigger, 1876
zyklischer Trigger, 1876, 1889
TIMESTEP, 1093
Titelseite
in Protokollen, 1293
Protokoll, 1283, 1286
Toleranzband
Variablen, 641, 647
Tooltipp
Grenzwerte anzeigen, 2202, 2250
Totzone
einstellen, 829, 846
TP 177B mono
Symbolbibliothek, 502
Transition/Interlock-Ansicht
PLC-Code-Anzeige, 1957
transparente Farbe, 392
Anzeige auf Panels, 392
Transparenz
in Grafik, 149
Transponder, 1834
Trennen
Archiv-Backup, 698
Meldearchiv, vom Projekt, 947
PROFIsafe, 1853
TrenneVariableVonHandrad, 1595
Trennzeichen ersetzen, 565

Trigger, 1874
ändern, 1888
azyklisch, 1874, 1883, 1891
Ereignis-Trigger, 1874, 1875, 1886, 1895
Standardzyklus, 1875
Variablentrigger, 1876, 1892
zyklisch, 1875, 1884, 1893, 1896, 1898
Trigger Anwendermeldung
Projektieren, 856
Triggermodus, 665
Triggervariable, 815
T-Stück, 520
Drehung, 520
Typ
Bibliothek, 2020
Typ erzeugen
Bild, 2042
Skript, 2041
Typen auf neueste Version aktualisieren, 2031

U

Überlauf, 1826
Überprüfung der Signatur
ausschalten, 2248, 2336
Übersetzen
Editor,
Projekt, 2210, 2260, 2365
Übertragen
License Key auf Bediengerät, 2438
Übertragungsprotokoll, 409
Uhr, 521
Anzeige, 521
Breite der Zeiger, 522
Länge des Zeigers, 522
Ziffernblatt anzeigen, 521
umbenennen
benutzerdefinierte C-Funktion, 1473
Benutzerdefinierte VB-Funktion, 1460
Bild, 44
Variable, 559
Vorlage, 54
Umlaufarchiv, 922
Größe wählen, 645, 935
Umlaufpuffer, 672
Grundlagen, 672
umschalten,
Taste, 2240, 2321, 2410
zwischen Runtime-Sprachen, 2115
umschalten AUS, 1827
Umschalten EIN, 1828
Umverdrahten, 2091
unabhängige Bildfenster, 2337
Ungültige Querverweise, 2089
Unterschiede
Runtimes, 2149
unterteilen
Kurvenanzeige, 731
URL, 407
USB
Laden, 2279, 2284
USB-Schnittstelle, 2284
USB-Treiber, 2284
Installation unter Windows 7, 2284
UserArchiveControl, 1256
Daten bearbeiten, 1266
Online-Projektierung, 1240
projektieren, 1263
Runtime-Daten exportieren, 1239

V

Variable, 660
absolut adressieren, 545
Adressierung, 550
Aktualisierung, 589
Anwenderdatentyp, 607
Anwenderdatentyp anlegen, 609
Anwenderdatentyp-Element, 607
archivieren, 641, 647
Archivierungszyklus, 616, 619
Bildbaustein-Typ, 307
Codierung, 625
Datentyp, 552, 556
Ereignis, 587
Erfassungsart, 588
Erfassungszyklus, 588, 589, 616
Erfassungszyklus festlegen, 592
Ersatzwert festlegen, 591
exportieren, 2069
externe, 660
externe Variable, 540, 542
externe Variable anlegen, 550
Grenzwert, 568
Grenzwerte, 567
importieren, 2071, 2072
in Meldung ausgeben, 833
in Runtime, 588
IndexVariablen, 585
indirekt adressieren, 585, 586, 593, 594
interne, 660
interne Variable, 547, 548
interne Variable anlegen, 552
Kommentar, 552

Konfiguration, 560
kopieren, 559
Länge, 552
Leuchtdrucktaster, 426
linear skalieren, 572, 573
lokal, 1452
löschen, 559
mehrere Variable konfigurieren, 561
multiplexen, 585
Name, 552
neu verbinden, 566
Objektliste, 551
Rohdatenvariable, 660
Schlüsselschalter, 494
ständig lesen, 589
Startwert, 570
Steuerung wechseln, 571
symbolisch adressieren, 546
Toleranzband, 641, 647
umbenennen, 559
Variablenarchiv, 641, 647
Variablentyp, 556
Verbindung zur Steuerung, 550
Werte darstellen, 704
Wertebereich, 625
Zugriff im lokalem Skript, 1450
Zugriff in benutzerdefinierte Funktion, 1450
Variable verwenden, 363
Variablen
bearbeiten, 560
Grundlagen, 537
Meldegruppe, 811
Variablenanbindung
Animation, 227
Variablenarchiv, 677, 1071, 1077
anlegen, 643
Archivierungszyklus, 640
Bearbeitungsmethode, 665
Begriffsdefinitionen, 658
Erfassungszyklus, 640
Fast, 695
Größe konfigurieren, 688
Name, 679, 681
Namenskonventionen, 1075
projektieren, 677
Slow, 695
Toleranzband, 647
Variablen, 641, 647
Variablenwert ausgeben, 709
Variablenarchivierung, 639, 659
Anwendung, 639
auf Anforderung, 670
bei Wertänderung, 670
Funktionsweise, 659
Grundlagen, 659
Projektierung, 659
Verwendung, 659
Zyklische, 668
Zyklisch-selektive, 669
Variablenarray, 597, 603
Variablendaten
Struktur für den Import, 2073, 2076, 2081
Variablenimport
Struktur der Variablendaten, 2073, 2081
Variablenliste
indirekt adressieren, 585, 586
Variablenname
verwendbare Namen in Archiven, 679
Variablen-Simulator, 2214, 2265
Variablensynchronisation, 2202, 2250, 2339
Variablentabelle
anwenderdefinierte, 538
für HMI-Geräte, 538
Standard, 538
Variablentrigger, 1876, 1892
Variablentyp, 556
Variablenwert
aktualisieren, 544
ausgeben, 704, 709
Variablenzählung bei Arrays, 598, 604
VBS
Beispiele in WinCC, 1513
VBScript:debuggen, 1485
Verbinden, 697
Archiv-Backup, 697
Meldearchiv, 947
Verbinder, 522
VerbindeVariableMitHandrad, 1596
Verbindung
integrierte Verbindung, 545
Verbunden, 1828
verdeckte Eingabe, 369
Verdichtungsarchiv, 671, 679, 1077
Funktionsweise, 671
projektieren, 679
Verdichtungsvariable, 660
vereinheitlichen
Objektgröße, 110, 1326
VerfolgeBenutzeränderung, 1617
verfügbare Systemfunktionen
Baisc Panels, 1536, 1541
Comfort Panels, 1547
Mobile Panels, 1554
WinCC Runtime, 1562, 1567

Verfügbarkeit
Editor "Lokale Skripte", 1482
Objekt für Basic Panel, 337
Objekt für Comfort Panel, 338
Objekt für Mobile Panel, 340
Objekt für WinCC Runtime Advanced, 343
Verfügbarkeit in Runtime
Online-Hilfe, 2338
Verhalten, 2346
einfache Rezepturanzeige, 1144
Verhalten beim Anpassen
Bildbaustein-Typ, 299, 300
Verknüpfte Objekte
kopieren, 2144, 2145
VerringereFokussiertenWert, 1596
VerringereVariable, 1623, 1742
verschieben
Bild, 44
Vorlage, 54
Versionen in WinCC, 2180, 2182
verteilen
Objekte gleichmäßig, 113, 1331
Vertikale Bewegung
Animation, 223
verwalten
Benutzer, 1387
Benutzer in Runtime, 1406
Benutzergruppe, 1389
Grafik, 150
Lizenz,
verwenden
Bildbaustein-Typ, 301
Formatvorlage, 89
Funktionstaste zur Bildnavigation, 261
Lokales Skript,
projektweite Meldeklasse,
Slide-in-Bild,
Verwendung
Benutzeranzeige, 350
Control Daten, 1337
Control Hardcopy, 1338
Control Informationen, 1339
Datum/Uhrzeit-Feld, 1342
Einfache Benutzeranzeige, 352
einfache Meldeanzeige, 958
einfaches Meldefenster, 958
f(x)-Kurvenanzeige, 1347
Meldeanzeige, 954
Meldefenster, 954
Meldeindikator, 961
ProDiag-Übersicht, 473
Rezepturanzeige, 482, 1154, 1165
Schlüsselschalter, 493
Verzögerungszeit
einstellen, 828, 846
Videoformat ändern, 1828
Vollbild, 1829
Vollbild verlassen, 1829
Vollbildmodus, 2201, 2249
Vordefinierte Meldeklassen, 800, 802
Vordefinierte Stile, 125
Vordergrundfarbe
dynamisch ändern, 219
Vorlage
anlegen, 52
einfügen, 54
Globales Bild, 52
kopieren, 54
löschen, 54
umbenennen, 54
verschieben, 54
verwenden im Bild, 56

W

Wasserzeichen
Protokoll, 1286
WChar, 574
Web-Adresse, 403, 407
Webbrowser, 2328
WebClient
Startbild und Sprache für Benutzer
einstellen, 1400
Startbild und Sprache für Benutzergruppen
einstellen, 1399
WebKit-Engine, 2328
WebNavigator
Startbild und Sprache für Benutzer
einstellen, 1400
Startbild und Sprache für Benutzergruppen
einstellen, 1399
WebNavigator-Lizenz, 2338
WebNavigator-Lizenz für WebUX verwenden, 2338
WebNavigator-Lizenzen
für WebUX verwenden, 1401
WebUX
Anzahl reservierter Lizenzen für Benutzer
konfigurieren, 1400
Startbild und Sprache für Benutzer
einstellen, 1400
Startbild und Sprache für Benutzergruppen
einstellen, 1399
WebUX-Lizenz, 2338

Web-Zugriff
Gestaltung, 2338
Lokale Benutzergruppen, 2338
Meldungen in der Ereignisanzeige, 2338
Serverauslastung, 2338
WechseleVerbindung, 1648
WechseleVerbindungEIP, 1649
Weiterblättern-Taste, 2239, 2320, 2409
wenig freier Speicherplatz, 1829
Wenig freier Speicherplatz, kritisch, 1830
Werkseinstellungen
Rücksetzen auf, 2433
Werkzeuge, 91, 95, 100
zur Gestaltung von Protokollen, 1296
Wert/Bereich
Textliste, 184
Wertachse
projektieren, 731
Wertänderung, 1830
Werte darstellen
als Kurve, 704
Werteanzeige
Export von Runtime-Daten projektieren, 737, 750, 759
Statuszeile, 736, 749, 756
Symbolleiste, 736, 749, 756
Wertebereich
Variablencodierung, 625
Wertespalte dynamisieren
Tabellenanzeige, 518
Wertetabelle, 754
Bedienung in Runtime, 767
Darstellungsmodi, 754
Daten im Statistikbereich-Fenster anzeigen, 779
Koordinaten eines Punkts anzeigen, 773
Online-Projektierung, 738, 751, 760
projektieren, 755
Wertspalte
projektieren, 743
Wetetabelle
Bedienelemente, 524
Wiedergabe, 1831
Wiederherstellen
Daten des Bediengeräts, 2429, 2432
von externem Speichermedium,
WinCC
Runtime starten, 2397
WinCC Archive Connector, 1117
WinCC DataConnector, 1117
WinCC OLE DB, 1115
Grundlagen, 1115
Microsoft, 1115
WinCC, 1115
WinCC OLE DB-Provider, 1084, 1117
Linked Server, 1117
SQL-Server, 1117
Zugriff auf Archivdaten, 1084
WinCC Runtime
verfügbare Systemfunktionen, 1562, 1567
WinCC Runtime Advanced
Anzeige- und Bedienobjekt, 343
WinCC Systemmeldungen, 1044
WinCC UserArchiveControl, (Siehe UserArchiveControl)
WinCC-Versionen
Kompatibilität, 2181, 2182
Wirkbereich, 1833, 1842, 1845, 1847
abmelden, 527
Abmeldung projektieren, 1863
anmelden, 526
Anmeldung projektieren, 1863
Arbeitsbereich, 1843
Funktionsweise, 1834
Overridefunktion, 530
projektieren, 1862
Runtime, 1845
Signalqualität berechnen, 530
Wirkbereich-Objekt projektieren, 1864
Wirkbereich (RFID), 1846
Abmeldung projektieren, 1867
Anmeldung projektieren, 1867
Arbeitsbereich, 1846
projektieren, 1866
Runtime, 1847
Übersicht, 1865
Wirkbereich (RFID)
abmelden, 529
anmelden, 528
Wirkbereichs-Name, 525
Wirkbereichs-Name (RFID), 528
Wirkbereichs-Signal, 529
WLAN-Bereich, 1833
WLAN-Empfang, 531
Darstellung, 531
WString, 574

X

X-Achse
projektieren, 731
xlsx-Datei, 2055, 2057, 2071, 2072, 2078, 2079
Aufbereiten für den Import, 2070, 2072
Variablen importieren, 2070, 2072

Y

Y-Achse
projektieren, 731

Z

Zeichen für WString/WChar
bereitstellen, 575
Zeichensatz
Menüs, 2338
ZeigeAnmeldedialog, 1617, 1729
ZeigeBlockInTIAPortalVonMeldung
ShowBlockInTIAPortalFromAlarm, 1772
ZeigeHilfetext, 1597
ZeigeMeldefenster, 1691
ZeigePLCCodeAnzeigeVonMeldung
ShowPLCCodeViewFromAlarm, 1773
ZeigePopupBild, 1598, 1600
Zeigerinstrument, 532
Farbe der einzelnen Bereiche, 534
GMP-relevante Variable, 532
maximaler Wert, 533
minimaler Wert, 533
Normalbereich anzeigen, 534
Spitzenwert anzeigen, 533
ZeigeSlideInBild, 1602
ZeigeSoftwareVersion, 1598
ZeigeSystemDiagnosefenster, 1685
ZeigeSystemmeldung, 1691
Zeit abgelaufen, 1831
Zeitachse
projektieren, 730
Zeitbasis
einstellen, 439
Zeitbereich, 724
dynamischer, 724
statischer, 724
Zeitgesteuerte Ausgabe
Protokoll, 1309
Zeitstempel, 814
Zentraler Melder, 887
projektieren, 905
Zertifikatprüfung, 2247, 2335
Ziffernblatt anzeigen, 521
Zone, 1834, 1839
Arbeitsbereich, 1840, 1842
Beim Eintritt, 535
Bildwechsel projektieren, 1855, 1859
ID Zone, 1856
Objekt abhängig von der Zone anzeigen, 1856
Qualität berechnen, 536
Zonen-ID, 535
Zonen-ID / Anschlusspunkt-ID, 1869
Zonen-Name, 534
Zonen-Signal, 535
zoomen
Bild, 40
Multitouchgeräte, 2313
Zoomfunktionen, 2338
einrichten, 2347
Zugangspunkt, 1869
Zugriff
Variable, 1450
Zugriff über WinCC OLE DB, 1105
Zugriffsschutz, 1420
Benutzerverwaltung, 1416, 1417
Meldeanzeige, 442
projektieren, 1424
Zugriffsschutz in Runtime
Meldeanzeige, 438
Zurückblättern-Taste, 2239, 2320, 2409
Zurücksetzen
auf Werkseinstellungen, 2433
Bediengerät auf Werkseinstellungen, 2434
Eigenschaften des Bildbausteins,
Zusatztext einer Meldung, 858
Zustandsvariable, 811
Meldegruppe, 886
Zuweisen
einer Funktionstaste eine Funktion, 253
einer Funktionstaste eine Grafik, 257
Objekt einer Ebene,
Zweifachquittierung, 806
Zweihand-Bedienung
Multitouchgeräte, 2315, 2405
Zwischenspeicherungspfad
Process Historian, 2339
Zyklisch
fortlaufend, 589
im Betrieb, 589
zyklischer Trigger, 1875, 1884, 1889, 1893, 1896, 1898
Zyklus, 665
erstellen, 617, 621
Startpunkt festlegen, 621

SIEMENS

Technologiefunktionen einsetzen

Programmier- und Bedienhandbuch

Zählen, Messen und Positionserfassung 1

Nockensteuern (S7-300, S7-400) 2

PID-Regelung 3

Time-based IO 4

Impulsausgabe 5

Motion Control 6

SIMATIC Ident 7

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Zählen, Messen und Positionserfassung....11

1.1 Zählen, Messen und Positionserfassung (S7-1500)....11
1.1.1 Grundlagen zu Zählen, Messen und Positionserfassung....11
1.1.1.1 Übersicht der Module und Eigenschaften....11
1.1.1.2 Grundlagen zu Zählen, Messen und Positionserfassung (TM Count, TM PosInput, Kompakt-CPU)....15
1.1.1.3 Grundlagen zu Zählen (TM Timer DIDQ)....69
1.1.1.4 Grundlagen zu Zählen (Digitalmodule)....74
1.1.2 Technologieobjekt High_Speed_Counter einsetzen....82
1.1.2.1 Konvention....82
1.1.2.2 Technologieobjekt High_Speed_Counter....82
1.1.2.3 Übersicht der Projektierungsschritte....82
1.1.2.4 Technologieobjekt hinzufügen....83
1.1.2.5 High_Speed_Counter konfigurieren....84
1.1.2.6 High_Speed_Counter programmieren....104
1.1.2.7 High_Speed_Counter in Betrieb nehmen....105
1.1.2.8 High_Speed_Counter Diagnose....106
1.1.3 Technologieobjekt SSI_Absolute_Encoder einsetzen....108
1.1.3.1 Technologieobjekt SSI_Absolute_Encoder....108
1.1.3.2 Übersicht der Projektierungsschritte....109
1.1.3.3 Technologieobjekt hinzufügen....109
1.1.3.4 SSI_Absolute_Encoder konfigurieren....111
1.1.3.5 SSI_Absolute_Encoder programmieren....125
1.1.3.6 SSI_Absolute_Encoder in Betrieb nehmen....127
1.1.3.7 SSI_Absolute_Encoder Diagnose....128
1.1.4 Modul einsetzen....129
1.1.4.1 Technologiemodul einsetzen....129
1.1.4.2 Digitalmodul einsetzen....176

1.2 Zählen und Messen (S7-300, S7-400)....182
1.2.1 Zählen und Messen....182
1.2.1.1 Grundlagen zum Zählen und Messen....182
1.2.1.2 FM 350-1 einsetzen....212
1.2.1.3 FM 350-2 einsetzen....266
1.2.1.4 FM 450-1 einsetzen....294
1.2.1.5 ET 200S 1Count24V einsetzen....325
1.2.1.6 ET 200S 1Count5V einsetzen....389

2 Nockensteuern (S7-300, S7-400)....453

2.1 Nockensteuern....453
2.1.1 Grundlagen zum Nockensteuern....453
2.1.1.1 Was ist Nockensteuern?....453
2.1.1.2 Eigenschaften der Nockenarten....453
2.1.1.3 Normalspuren und Spurergebnisse....455
2.1.1.4 Sonderspuren....457
2.1.1.5 Hysterese....458

2.1.1.6 Dynamische Verstellung....459
2.1.1.7 Schnittstellen des Nockensteuerwerks....460
2.1.1.8 Gebersignale....462
2.1.2 Funktionen der FM x52....467
2.1.2.1 Maschinendaten schreiben und aktivieren....467
2.1.2.2 Maschinendaten lesen....469
2.1.2.3 Nockendaten schreiben....470
2.1.2.4 Nockendaten lesen....470
2.1.3 FM 352 einsetzen....471
2.1.3.1 Anschlussbelegung der Frontstecker der FM 352....471
2.1.3.2 Funktionen zum Nockensteuern....474
2.1.3.3 FM 352 konfigurieren....498
2.1.3.4 FM 352 parametrieren....498
2.1.3.5 FM 352 in Betrieb nehmen....533
2.1.3.6 FM 352 aus dem Anwenderprogramm verwenden....533
2.1.3.7 FM 352: Online & Diagnose....534
2.1.4 FM 452 einsetzen....549
2.1.4.1 Anschlussbelegung des Frontsteckers der FM 452....549
2.1.4.2 Funktionen zum Nockensteuern....551
2.1.4.3 FM 452 konfigurieren....576
2.1.4.4 FM 452 parametrieren....576
2.1.4.5 FM 452 in Betrieb nehmen....612
2.1.4.6 FM 452 aus dem Anwenderprogramm verwenden....612
2.1.4.7 FM 452: Online & Diagnose....614

3 PID-Regelung....631

3.1 Grundlagen zum Regeln....631
3.1.1 Regelkreis und Stellglieder....631
3.1.2 Regelstrecken....632
3.1.3 Kennwerte der Regelstrecke....635
3.1.4 Impulsregler....637
3.1.5 Führungs- und Störverhalten....641
3.1.6 Regelverhalten bei unterschiedlichen Rückführstrukturen....642
3.1.7 Wahl der Reglerstruktur bei gegebener Regelstrecke....649
3.1.8 Einstellung der PID-Parameter....650

3.2 Projektieren eines Software-Reglers....651
3.2.1 Übersicht der Software-Regler....651
3.2.2 Schritte für das Projektieren eines Software-Reglers....652
3.2.3 Technologieobjekte hinzufügen....653
3.2.4 Technologieobjekte konfigurieren....654
3.2.5 Anweisung im Anwenderprogramm aufrufen....655
3.2.6 Technologieobjekte in Gerät laden....656
3.2.7 Software-Regler in Betrieb nehmen....657
3.2.8 Optimierte PID-Parameter im Projekt speichern....658
3.2.9 Werte vergleichen....658
3.2.9.1 Vergleichsanzeige und Randbedingungen....658
3.2.9.2 Werte vergleichen....659
3.2.10 Parametersicht....661
3.2.10.1 Einführung in die Parametersicht....661
3.2.10.2 Aufbau der Parametersicht....663
3.2.10.3 Parametersicht öffnen....666
3.2.10.4 Voreinstellung der Parametersicht....666

3.2.10.5 Arbeiten mit der Parametersicht....668
3.2.11 Instanz-DB eines Technologieobjekts anzeigen....680

3.3 PID-Regelung (S7-1200, S7-1500)....681
3.3.1 PID_Compact einsetzen....681
3.3.1.1 Technologieobjekt PID_Compact....681
3.3.1.2 PID_Compact V2....682
3.3.1.3 PID_Compact V1....702
3.3.2 PID_3Step einsetzen....719
3.3.2.1 Technologieobjekt PID_3Step....719
3.3.2.2 PID_3Step V2....720
3.3.2.3 PID_3Step V1....740
3.3.3 PID_Temp einsetzen....758
3.3.3.1 Technologieobjekt PID_Temp....758
3.3.3.2 PID_Temp konfigurieren....759
3.3.3.3 PID_Temp in Betrieb nehmen....782
3.3.3.4 Kaskadenregelung mit PID_Temp....790
3.3.3.5 Mehrzonenregelung mit PID_Temp....797
3.3.3.6 Ablöseregelung mit PID_Temp....799
3.3.3.7 PID_Temp mit PLCSIM simulieren....803

3.4 PID-Regelung (S7-300, S7-400, S7-1500)....804
3.4.1 PID Basisfunktionen einsetzen....804
3.4.1.1 CONT_C....804
3.4.1.2 CONT_S....808
3.4.1.3 TCONT_CP....810
3.4.1.4 TCONT_S....831

3.5 PID-Regelung (S7-300, S7-400)....834
3.5.1 Standard PID Control einsetzen....834
3.5.1.1 PID_CP einsetzen....834
3.5.1.2 PID_ES einsetzen....883
3.5.2 Modular PID Control einsetzen....930
3.5.2.1 Funktionsumfang des Modular PID Control....930
3.5.2.2 Grundsätzliche Schritte zum Aufbau eines Regler-FBs....931
3.5.2.3 Regler-FB programmieren....932
3.5.2.4 Komplexe Reglerstrukturen....950
3.5.2.5 Regler-FB in Betrieb nehmen....962
3.5.3 PID Self-Tuner einsetzen....963
3.5.3.1 Kontinuierliche Regler und Impulsregler optimieren....963
3.5.3.2 Schrittregler optimieren....971
3.5.4 Grundlagen Reglerbaugruppen....978
3.5.4.1 Eingänge....978
3.5.4.2 Regeldifferenz....983
3.5.4.3 Regelalgorithmus....988
3.5.4.4 Reglerausgang....1004
3.5.4.5 Unterstützung Shared Device....1016
3.5.5 FM 355 einsetzen....1017
3.5.5.1 FM 355 konfigurieren....1017
3.5.5.2 FM 355 parametrieren....1017
3.5.5.3 FM 355 programmieren....1033
3.5.5.4 FM 355 in Betrieb nehmen....1033
3.5.5.5 FM 355: Diagnose durch LED-Anzeige....1043
3.5.5.6 Beispielprojekte zur FM 355....1050

3.5.6 FM 355-2 einsetzen....1050
3.5.6.1 FM 355-2 konfigurieren....1050
3.5.6.2 FM 355-2 parametrieren....1051
3.5.6.3 FM 355-2 in Betrieb nehmen....1067
3.5.6.4 FM 355-2: Diagnose durch LED-Anzeige....1092
3.5.7 FM 455 einsetzen....1099
3.5.7.1 FM 455 konfigurieren....1099
3.5.7.2 FM 455 parametrieren....1100
3.5.7.3 FM 455 programmieren....1116
3.5.7.4 FM 455 in Betrieb nehmen....1116
3.5.7.5 FM 455: Diagnose durch LED-Anzeige....1125
3.5.7.6 Beispielprojekte zur FM 455....1135
3.5.8 ET 200S 2PULSE einsetzen....1135
3.5.8.1 Anschlussbelegung des 2PULSE....1135
3.5.8.2 Betriebsarten und Funktionen....1136
3.5.8.3 2PULSE konfigurieren und parametrieren....1180
3.5.8.4 Diagnose des 2PULSE....1189

3.6 Hilfsfunktionen....1191
3.6.1 Polyline....1191
3.6.2 SplitRange....1191
3.6.3 RampFunction....1192

4 Time-based IO....1193

4.1 Hochgenaue Ein-/Ausgabe mit Time-based IO....1193
4.1.1 Grundlagen von Time-based IO....1193
4.1.1.1 Nutzung von Time-based IO....1193
4.1.1.2 Zeitliches Verhalten von Time-based IO....1194
4.1.1.3 Systemumgebung....1196
4.1.1.4 Hardware-Voraussetzungen....1196
4.1.1.5 Projektiersoftware-Voraussetzungen....1197
4.1.1.6 Technische Umsetzung....1198
4.1.2 Konfigurieren und Parametrieren....1201
4.1.2.1 Voraussetzungen....1201
4.1.2.2 Einstellungen für Time-based IO....1202
4.1.2.3 IO-Link konfigurieren und Time-based IO aktivieren....1203
4.1.3 Programmieren....1204
4.1.3.1 Übersicht der Anweisungen....1204
4.1.3.2 Programmierung von Time-based IO....1205
4.1.4 Technologiemodul einsetzen....1209
4.1.4.1 TIO-Modul konfigurieren und parametrieren....1209
4.1.4.2 TIO-Modul Online & Diagnose....1223

5 Impulsausgabe....1225

5.1 Grundlagen zur Impulsausgabe....1225
5.1.1 Übersicht der Module und Eigenschaften....1225
5.1.2 Grundlagen zur Impulsausgabe (TM Pulse 2x24V)....1226
5.1.2.1 Übersicht....1226
5.1.2.2 Betriebsart Impulsausgabe (Einzelimpuls)....1229
5.1.2.3 Betriebsart Impulsweitenmodulation (PWM)....1237
5.1.2.4 Betriebsart Impulskette....1248
5.1.2.5 Betriebsart Ein-/Ausschaltverzögerung....1258
5.1.2.6 Betriebsart Frequenzausgabe....1267

5.1.2.7 Betriebsart Gleichstrommotor....1274
5.1.2.8 Funktion: Schnelle Impulsausgabe....1283
5.1.2.9 Funktion: Sequenzzähler....1284
5.1.2.10 Funktion: Strommessung....1285
5.1.2.11 Funktion: Stromregelung....1287
5.1.2.12 Funktion: Dither PWM-Ausgang....1290
5.1.2.13 Funktion: Taktsynchroner Betrieb....1294
5.1.2.14 Funktion: Direktansteuerung von DQ-Digitalausgängen....1295
5.1.3 Grundlagen zur Impulsausgabe (Kompakt-CPU)....1299
5.1.3.1 Betriebsarten....1299
5.1.3.2 Funktionen....1310
5.1.4 Grundlagen zur Impulsausgabe (TM PTO 4)....1312
5.1.4.1 Übersicht....1312
5.1.4.2 Pulse Train Output (PTO)....1312

5.2 Modul einsetzen....1315
5.2.1 TM Pulse 2x24V einsetzen....1315
5.2.1.1 Technologiemodul konfigurieren und parametrieren....1315
5.2.1.2 Modul Online & Diagnose....1325
5.2.2 TM PTO 4 einsetzen....1326
5.2.2.1 Technologiemodul konfigurieren und parametrieren....1326
5.2.2.2 Modul Online & Diagnose....1332
5.2.3 Kompakt-CPU einsetzen....1332
5.2.3.1 Kompakt-CPU konfigurieren und parametrieren....1332
5.2.3.2 Modul Online & Diagnose....1340

6 Motion Control....1341

6.1 Motion Control (S7-1200, S7-1500)....1341
6.1.1 S7-1200 Motion Control einsetzen....1341
6.1.1.1 Einleitung....1341
6.1.1.2 Grundlagen für das Arbeiten mit S7-1200 Motion Control....1344
6.1.1.3 Leitfaden zum Einsatz von Motion Control....1371
6.1.1.4 Versionen einsetzen....1371
6.1.1.5 Technologieobjekt Positionierachse....1381
6.1.1.6 Technologieobjekt Auftragstabelle....1449
6.1.1.7 Laden in CPU....1469
6.1.1.8 Inbetriebnahme....1470
6.1.1.9 Programmieren....1477
6.1.1.10 Achse - Diagnose....1499
6.1.1.11 Anhang....1504
6.1.2 S7-1500 Motion Control einsetzen....1679
6.1.2.1 Einleitung....1679
6.1.2.2 Technologieobjekte....1688
6.1.2.3 Versionen einsetzen....1826
6.1.2.4 Konfigurieren....1835
6.1.2.5 Programmieren....1988
6.1.2.6 Laden in CPU....2017
6.1.2.7 Inbetriebnahme....2018
6.1.2.8 Diagnose....2033
6.1.2.9 Technologieversion V3.0....2058
6.1.2.10 Anhang....2085
6.1.3 S7-1500T Kinematikfunktionen einsetzen....2224
6.1.3.1 Einleitung....2224

6.1.3.2 Grundlagen....2227
6.1.3.3 Versionsübersicht....2340
6.1.3.4 Konfigurieren....2340
6.1.3.5 Programmieren....2361
6.1.3.6 Inbetriebnahme....2369
6.1.3.7 Diagnose....2382
6.1.3.8 Anhang....2387

6.2 Motion Control (S7-300, S7-400)....2430
6.2.1 Grundlagen zur Wegerfassung....2430
6.2.1.1 Was ist Wegerfassung?....2430
6.2.1.2 Geberwerterfassung....2430
6.2.1.3 Gray-/Dualwandler....2431
6.2.1.4 Geberwert und Normierung....2431
6.2.2 SM 338 einsetzen....2433
6.2.2.1 Anschlussbelegung des Frontsteckers der SM 338....2433
6.2.2.2 SM 338 konfigurieren....2435
6.2.2.3 SM 338 parametrieren....2435
6.2.2.4 SM 338: Online & Diagnose....2441
6.2.3 ET 200S 1SSI einsetzen....2443
6.2.3.1 Anschlussbelegung des 1SSI....2443
6.2.3.2 1SSI konfigurieren und parametrieren....2445
6.2.3.3 Diagnose des 1SSI....2465
6.2.4 Grundlagen zum Positionieren mit Funktionsbaugruppen....2466
6.2.4.1 Was ist Positionieren?....2466
6.2.4.2 Gesteuertes Positionieren mit Eil-/Schleichgang....2467
6.2.4.3 Bereiche und Schaltpunkte der Positionierbaugruppe....2467
6.2.4.4 Gebersignale....2469
6.2.5 FM 351 einsetzen....2473
6.2.5.1 Anschlussbelegung der Frontstecker der FM 351....2473
6.2.5.2 Funktionen der FM 351....2476
6.2.5.3 FM 351 konfigurieren....2513
6.2.5.4 FM 351 parametrieren....2513
6.2.5.5 FM 351 aus dem Anwenderprogramm verwenden....2546
6.2.5.6 FM 351 in Betrieb nehmen....2547
6.2.5.7 FM 351: Online & Diagnose....2547
6.2.6 FM 451 einsetzen....2564
6.2.6.1 Anschlussbelegung der Frontstecker der FM 451....2564
6.2.6.2 Funktionen der FM 451....2568
6.2.6.3 FM 451 konfigurieren....2611
6.2.6.4 FM 451 parametrieren....2611
6.2.6.5 FM 451 aus dem Anwenderprogramm verwenden....2645
6.2.6.6 FM 451 in Betrieb nehmen....2646
6.2.6.7 FM 451: Online & Diagnose....2646
6.2.7 ET 200S 1PosU einsetzen....2663
6.2.7.1 Anschlussbelegung des 1PosU....2663
6.2.7.2 Grundlagen des gesteuerten Positionierens über Eil-/Schleichgang....2665
6.2.7.3 Funktionen des 1PosU....2667
6.2.7.4 1PosU konfigurieren und parametrieren....2708
6.2.7.5 Diagnose des 1PosU....2729
6.2.8 ET 200S 1STEP 5V einsetzen....2730
6.2.8.1 Anschlussbelegung des 1STEP 5V....2730
6.2.8.2 Grundlagen des Positionierens mit Schrittmotor....2732

6.2.8.3 Funktionen des 1STEP 5V....2738
6.2.8.4 1STEP 5V konfigurieren und parametrieren....2759
6.2.8.5 Diagnose des 1STEP 5V....2769
6.2.9 IM 174 einsetzen....2770
6.2.9.1 Übersicht....2770
6.2.9.2 IM174 projektieren und programmieren....2771
6.2.10 Easy Motion Control einsetzen....2786
6.2.10.1 Positionieren mit Easy Motion Control....2786
6.2.10.2 Funktionen mit Easy Motion Control....2794
6.2.10.3 Easy Motion Control konfigurieren....2797
6.2.10.4 Laden in CPU....2820
6.2.10.5 Easy Motion Control programmieren....2822
6.2.10.6 Easy Motion Control in Betrieb nehmen....2836
6.2.10.7 Diagnose....2843

7 SIMATIC Ident....2853

7.1 Ident-Systeme konfigurieren....2853
7.1.1 Technologieobjekt "SIMATIC Ident"....2853
7.1.2 Technologieobjekt hinzufügen....2854
7.1.3 Geräte mithilfe des Technologieobjekts konfigurieren....2855
7.1.3.1 Arbeiten mit dem Konfigurationsdialog....2855
7.1.3.2 Grundparameter konfigurieren....2856
7.1.3.3 Ident-Gerät-Parameter konfigurieren....2856
7.1.3.4 Reader-Parameter konfigurieren....2858
7.1.4 Ident-Anweisung im Anwenderprogramm aufrufen....2861
7.1.5 Projektierung in CPU laden....2862

7.2 Ident-Systeme diagnostizieren....2863
7.2.1 Baustein-Diagnose....2863
7.2.2 Hardware-Diagnose....2864

Index....2867

Zählen, Messen und Positionserfassung 1

1.1 Zählen, Messen und Positionserfassung (S7-1500)

1.1.1 Grundlagen zu Zählen, Messen und Positionserfassung

1.1.1.1 Übersicht der Module und Eigenschaften

Module für die Systeme S7-1500 und ET 200MP

Die folgende Tabelle gibt für die Systeme S7-1500 und ET 200MP eine Übersicht über die Leistungsmerkmale der Module für Zählen, Messen und Positionserfassung.

Eigenschaft	S7-1500 / ET 200MP				
	Technologiemodul			Digitaleingabemodul	Kompakt-CPU
	TM Count 2x24V	TM PosInput 2	TM Timer DIDQ 16x24V	DI 32x24VDC HF DI 16x24VDC HF	CPU 1511C-1 PN CPU 1512C-1 PN
Anzahl Zähler	2	2	4[1]	2	6
Verwendung der Zähler aktivierbar/deaktivierbar	—	—	X	X	X
Maximale Signalfrequenz	200 kHz	1 MHz	50 kHz	1 kHz	100 kHz
Maximale Zählfrequenz bei Inkrementalgeber mit Vierfachauswertung (Seite 66)	800 kHz	4 MHz	200 kHz	—	400 kHz
Maximaler Zählbereich	32 Bit (Seite 20)	32 Bit (Seite 20)	32 Bit (Seite 70)	32 Bit (Seite 75)	32 Bit (Seite 20)
Maximaler Positionswertebereich (Seite 22)	—	32 Bit	—	—	—
Anschluss 24 V-Inkrementalgeber	X (Seite 62)	—	X (Seite 72)	—	X (Seite 62)
Anschluss 24 V-Impulsgeber	X (Seite 62)	—	X (Seite 72)	X (Seite 81)	X (Seite 62)
Anschluss RS422/ TTL-Inkremental- und Impulsgeber (Seite 64)	—	X	—	—	—

Eigenschaft	S7-1500 / ET 200MP				
	Technologiemodul			Digitaleingabemodul	Kompakt-CPU
	TM Count 2x24V	TM PosInput 2	TM Timer DIDQ 16x24V	DI 32x24VDC HF DI 16x24VDC HF	CPU 1511C-1 PN CPU 1512C-1 PN
Anschluss SSI-Absolutwertgeber (Seite 65)	—	X	—	—	—
Positionserfassung für Motion Control (Seite 61)	X	X	—	—	X
5 V-Geberversorgung	—	X	—	—	—
24 V-Geberversorgung	X	X	X	—	X
Anzahl zusätzlicher Digitaleingänge pro Zähler	3	2	—	—	2
Anzahl physikalischer Digitalausgänge pro Zähler	2	2	—	—	1
Anzahl logischer Digitalausgänge pro Zähler	2	2	—	1	2
SW-Tor	X (Seite 25)	X (Seite 25)	—	X (Seite 77)	X (Seite 25)
HW-Tor	X (Seite 26)	X (Seite 26)	—	—	X (Seite 26)
Capture-Funktion (Latch) (Seite 28)	X	X	—	—	X
Synchronisation (Seite 33)	X	X	—	—	X
Vergleichsfunktionen	X (Seite 40)	X (Seite 40)	—	X (Seite 78)	X (Seite 40)
Hysterese (Seite 57)	X	X	—	—	X
Frequenz-, Geschwindigkeits- und Periodendauermessung (Seite 51)	X	X	—	—	X
Unterstützung Taktsynchronität	X (Seite 68)	X (Seite 68)	X (Seite 73)	X (Seite 81)	—
Unterstützung Diagnosealarme für Gebersignale	X (Seite 61)	X (Seite 61)	—	—	X (Seite 61)
Unterstützung Prozessalarme	X (Seite 61)	X (Seite 61)	—	X (Seite 80)	X (Seite 61)
Parametrierbarer Filter für Zählsignale und Digitaleingänge	X	X	—	X	X

[1] Die Anzahl verfügbarer Zähler ist abhängig von der Kanalkonfiguration. Um vier Zähler nutzen zu können, müssen Sie in der Kanalkonfiguration die Verwendung von acht Eingängen wählen. Wenn Sie die Verwendung von drei Eingängen wählen, können Sie einen Zähler nutzen. Andere Kanalkonfigurationen ermöglichen keine Zählernutzung.

Module für das System ET 200SP

Die folgende Tabelle gibt für das System ET 200SP eine Übersicht über die Leistungsmerkmale der Module für Zählen, Messen und Positionserfassung.

Eigenschaft	ET 200SP			
	Technologiemodul			Digitaleingabemodul
	TM Count 1x24V	TM PosInput 1	TM Timer DIDQ 10x24V	DI 8x24VDC HS
Anzahl Zähler	1	1	3[1]	4
Verwendung der Zähler aktivierbar/deaktivierbar	—	—	X	X
Maximale Signalfrequenz	200 kHz	1 MHz	50 kHz	10 kHz
Maximale Zählfrequenz bei Inkrementalgeber mit Vierfachauswertung (Seite 66)	800 kHz	4 MHz	200 kHz	—
Maximaler Zählbereich	32 Bit (Seite 20)	32 Bit (Seite 20)	32 Bit (Seite 70)	32 Bit (Seite 75)
Maximaler Positionswertebereich (Seite 22)	—	32 Bit	—	—
Anschluss 24 V-Inkrementalgeber	X (Seite 62)	—	X (Seite 72)	—
Anschluss 24 V-Impulsgeber	X (Seite 62)	—	X (Seite 72)	X (Seite 81)
Anschluss RS422/TTL-Inkremental- und Impulsgeber (Seite 64)	—	X	—	—
Anschluss SSI-Absolutwertgeber (Seite 65)	—	X	—	—
Positionserfassung für Motion Control (Seite 61)	X	X	—	—
24 V-Geberversorgung	X	X	X	X
Anzahl zusätzlicher Digitaleingänge pro Zähler	3	2	—	1
Anzahl physikalischer Digitalausgänge pro Zähler	2	2	—	—
Anzahl logischer Digitalausgänge pro Zähler	2	2	—	1
SW-Tor	X (Seite 25)	X (Seite 25)	—	X (Seite 77)
HW-Tor	X (Seite 26)	X (Seite 26)	—	X (Seite 77)
Capture-Funktion (Latch) (Seite 28)	X	X	—	—
Synchronisation (Seite 33)	X	X	—	—
Vergleichsfunktionen	X (Seite 40)	X (Seite 40)	—	X (Seite 78)
Hysterese (Seite 57)	X	X	—	—
Frequenz-, Geschwindigkeits- und Periodendauermessung (Seite 51)	X	X	—	—
Unterstützung Taktsynchronität	X (Seite 68)	X (Seite 68)	X (Seite 73)	X (Seite 81)
Unterstützung Diagnosealarme für Gebersignale	X (Seite 61)	X (Seite 61)	—	—

Eigenschaft	ET 200SP			
	Technologiemodul			Digitaleingabemodul
	TM Count 1x24V	TM PosInput 1	TM Timer DIDQ 10x24V	DI 8x24VDC HS
Unterstützung Prozessalarme	X (Seite 61)	X (Seite 61)	—	—
Parametrierbarer Filter für Zählsignale und Digitaleingänge	X	X	—	X

[1] Ein Zähler für Inkrementalgeber (A, B phasenversetzt) und zwei Zähler für Impulsgeber

Module für das System ET 200AL

Die folgende Tabelle gibt für das System ET 200AL eine Übersicht über die Leistungsmerkmale der Module für Zählen, Messen und Positionserfassung.

Eigenschaft	ET 200AL
	Digitalein-/Digitalausgabemodul
	DIQ 16x24VDC/0.5A 8xM12
Anzahl Zähler	4
Verwendung der Zähler aktivierbar/deaktivierbar	X
Maximale Signalfrequenz	2 kHz
Maximaler Zählbereich	32 Bit
Anschluss 24 V-Inkrementalgeber	—
Anschluss 24 V-Impulsgeber	X
24 V-Geberversorgung	X
Anzahl zusätzlicher Digitaleingänge pro Zähler	2
Anzahl physikalischer Digitalausgänge pro Zähler	1
Anzahl logischer Digitalausgänge pro Zähler	1
SW-Tor	X
HW-Tor	X
Capture-Funktion (Latch)	—
Synchronisation	—
Vergleichsfunktionen	X
Frequenz-, Geschwindigkeits- und Periodendauermessung	—
Unterstützung Taktsynchronität	—
Unterstützung Diagnosealarme für Gebersignale	X
Unterstützung Prozessalarme	X
Parametrierbarer Filter Digitaleingänge	X

1.1.1.2 Grundlagen zu Zählen, Messen und Positionserfassung (TM Count, TM PosInput, Kompakt-CPU)

Konvention

Technologiemodul: Die Bezeichnung "Technologiemodul" verwenden wir im vorliegenden Kapitel sowohl für die Technologiemodule TM Count und TM PosInput als auch für den Technologieanteil der Kompakt-CPUs.

Übersicht der Einsatzmöglichkeiten

Einleitung

Sie konfigurieren und parametrieren das Technologiemodul mit der Projektierungssoftware.

Die Steuerung und Kontrolle der Funktionen des Technologiemoduls erfolgt entweder über das Technologieobjekt oder durch das Anwenderprogramm über die Steuer- und Rückmeldeschnittstelle.

Systemumgebung für TM Count und TM PosInput

Die Technologiemodule können in folgenden Systemumgebungen eingesetzt werden:

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Zentraler Betrieb mit einer CPU S7-1500 oder CPU 151xSP	• Automatisierungssystem S7-1500 oder ET 200SP CPU • Technologiemodul	STEP 7 (TIA Portal): Betrieb mit Technologieobjekt "Zählen und Messen" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Technologieobjekt High_Speed_Counter oder SSI_Absolute_Encoder	Für Inkremental- /Impulsgeber: Anweisung High_Speed_Counter zum Technologieobjekt Für SSI-Absolutwertgeber: Anweisung SSI_Absolute_Encoder zum Technologieobjekt
		STEP 7 (TIA Portal): Positionserfassung für Technologieobjekt "Motion Control" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Achs- und Messtaster-Technologieobjekt (Seite 136)	Anweisungen zum Technologieobjekt "Motion Control"
		STEP 7 (TIA Portal): Manueller Betrieb (ohne Technologieobjekt) • Gerätekonfiguration und Parametereinstellung (Seite 130) mit Hardware-Konfiguration	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle des Technologiemoduls in den IO-Daten

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Dezentraler Betrieb mit einer CPU S7-1500	• Automatisierungssystem S7-1500 • Dezentrales Peripheriesystem ET 200 • Technologiemodul	STEP 7 (TIA Portal): Betrieb mit Technologieobjekt "Zählen und Messen" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Technologieobjekt High_Speed_Counter oder SSI_Absolute_Encoder	Für Inkremental- /Impulsgeber: Anweisung High_Speed_Counter zum Technologieobjekt Für SSI-Absolutwertgeber: Anweisung SSI_Absolute_Encoder zum Technologieobjekt
		STEP 7 (TIA Portal): Positionserfassung für Technologieobjekt "Motion Control" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Achs- und Messtaster-Technologieobjekt (Seite 136)	Anweisungen zum Technologieobjekt "Motion Control"
		STEP 7 (TIA Portal): Manueller Betrieb (ohne Technologieobjekt) • Gerätekonfiguration und Parametereinstellung (Seite 130) mit Hardware-Konfiguration	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle (Seite 170) des Technologiemoduls in den IO-Daten
Dezentraler Betrieb mit einer CPU S7-1200	• Automatisierungssystem S7-1200 • Dezentrales Peripheriesystem ET 200 • Technologiemodul	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung (Seite 130) mit Hardware-Konfiguration	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle (Seite 170) des Technologiemoduls in den IO-Daten

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Dezentraler Betrieb mit einer CPU S7-300/400	• Automatisierungssystem S7-300/400 • Dezentrales Peripheriesystem ET 200 • Technologiemodul	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung (Seite 130) mit Hardware-Konfiguration STEP 7: Gerätekonfiguration und Parametereinstellung mit HSP (bis Firmware-Stand V1.2) oder GSD-Datei	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle (Seite 170) des Technologiemoduls in den IO-Daten
Dezentraler Betrieb in einem System anderer Hersteller	• Automatisierungssystem anderer Hersteller • Dezentrales Peripheriesystem ET 200 • Technologiemodul	Projektierungssoftware anderer Hersteller: Gerätekonfiguration und Parametereinstellung mit GSD-Datei	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle (Seite 170) des Technologiemoduls in den IO-Daten

Systemumgebung für Kompakt-CPU

Die Kompakt-CPUs können in folgender Systemumgebung eingesetzt werden:

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Zentraler Betrieb mit einer Kompakt-CPU S7-1500	• Automatisierungssystem S7-1500 • Kompakt-CPU	STEP 7 (TIA Portal): Betrieb mit Technologieobjekt "Zählen und Messen" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Technologieobjekt High_Speed_Counter	Anweisung High_Speed_Counter zum Technologieobjekt
		STEP 7 (TIA Portal): Positionserfassung für Technologieobjekt "Motion Control" • Gerätekonfiguration mit Hardware-Konfiguration (Seite 130) • Parametereinstellung mit Achs- und Messtaster-Technologieobjekt (Seite 136)	Anweisungen zum Technologieobjekt "Motion Control"
		STEP 7 (TIA Portal): Manueller Betrieb (ohne Technologieobjekt) • Gerätekonfiguration und Parametereinstellung (Seite 130) mit Hardware-Konfiguration	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle des Technologiemoduls in den IO-Daten

Möglichkeiten der Parametrierung

In einem S7-1500-System haben Sie zwei alternative Möglichkeiten für die Parametrierung und Steuerung der Funktionen des Technologiemoduls:

- Konfiguration über Technologieobjekt und Steuerung über zugehörige Anweisung
  Der Zugriff auf die Steuer- und Rückmeldeschnittstelle des Technologiemoduls findet durch das Technologieobjekt statt.
- Parametereinstellung über Hardware-Konfiguration
  Der Zugriff auf die Steuer- und Rückmeldeschnittstelle des Technologiemoduls erfolgt durch den direkten Zugriff auf die IO-Daten.

Konfiguration über Technologieobjekt

Für den zentralen und dezentralen Einsatz empfehlen wir die komfortable und grafikunterstützte Konfiguration über ein Technologieobjekt. Die ausführliche Beschreibung dieser Konfiguration finden Sie ab Abschnitt Technologieobjekt High_Speed_Counter oder Technologieobjekt SSI_Absolute_Encoder.

Bei der Gerätekonfiguration des Technologiemoduls legen Sie den "Betrieb mit Technologieobjekt "Zählen und Messen"" fest: siehe Abschnitt Betriebsmodus (Seite 135).

Bei den Grundparametern des Technologieobjekts treffen Sie die Zuordnung zum Technologiemodul und Zählkanal: siehe Abschnitt Grundparameter (Seite 86).

Parametereinstellung über Hardware-Konfiguration

Bei der Gerätekonfiguration des Technologiemoduls legen Sie den "Manueller Betrieb (ohne Technologieobjekt)" fest: siehe Abschnitt Betriebsmodus (Seite 135).

Weitere Unterstützung zur Parametereinstellung über die Hardware-Konfiguration finden Sie in der Kontext-Hilfe zu den Parametern in STEP 7 (TIA Portal). Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie in folgenden Abschnitten:

- Belegung der Steuerschnittstelle (Seite 170)
- Belegung der Rückmeldeschnittstelle (Seite 172)

Erfassung von Zählsignalen

Zählen mit Inkremental- oder Impulsgeber

Zählen ist das Erfassen und Aufsummieren von Ereignissen. Die Zähler der Technologiemodule erfassen Impuls- und Inkrementalsignale und werten diese entsprechend aus. Die Zählrichtung kann durch geeignete Geber- bzw. Impulssignale oder über das Anwenderprogramm vorgegeben werden.

Mit den Digitaleingängen des Technologiemoduls können Sie die Zählvorgänge steuern. Die Digitalausgänge können Sie exakt an definierten Zählwerten unabhängig vom Anwenderprogramm schalten.

Sie können das Verhalten der Zähler mit Hilfe der im Folgenden beschriebenen Funktionalitäten parametrieren.

Zählgrenzen

Die Zählgrenzen definieren den genutzten Wertebereich der Zählwerte. Die Zählgrenzen sind parametrierbar und zur Laufzeit über das Anwenderprogramm änderbar.

Die maximale einstellbare Zählgrenze beträgt 2147483647 (2^{31}–1). Die minimale einstellbare Zählgrenze beträgt –2147483648 (–2^{31}).

Sie können das Verhalten des Zählers an den Zählgrenzen parametrieren:

- Zählvorgänge bei Überschreiten einer Zählgrenze fortsetzen oder beenden (automatischer Torstopp)
- Zählwert bei Überschreiten einer Zählgrenze auf den Startwert oder auf die andere Zählgrenze setzen

Startwert

Sie können einen Startwert innerhalb der Zählgrenzen parametrieren. Der Startwert ist zur Laufzeit über das Anwenderprogramm änderbar.

Das Technologiemodul kann den aktuellen Zählwert je nach Parametrierung bei der Synchronisation, bei der Capture-Funktion, beim Überschreiten einer Zählgrenze oder beim Öffnen des Tors auf den Startwert setzen.

Torsteuerung

Das Öffnen und Schließen des Hardware-Tors (HW-Tor) und Software-Tors (SW-Tor) definiert das Zeitfenster, in dem die Zählsignale erfasst werden.

Die Steuerung des HW-Tors erfolgt extern über einen Digitaleingang des Technologiemoduls. Die Steuerung des SW-Tors erfolgt über das Anwenderprogramm. Das HW-Tor kann durch Parametrierung aktiviert werden. Das SW-Tor (Bit in der Steuerschnittstelle der zyklischen IO-Daten) kann nicht deaktiviert werden.

Capture (Latch)

Sie können die Flanke eines externen Referenzsignals parametrieren, die ein Speichern des aktuellen Zählwerts oder Positionswerts als Capture-Wert auslöst. Folgende externe Signale können die Capture-Funktion auslösen:

- Steigende oder fallende Flanke eines Digitaleingangs
- Beide Flanken eines Digitaleingangs
- Steigende Flanke des Signals N am Gebereingang

Bei Verwendung eines Digitaleingangs können Sie parametrieren, ob im Anschluss an die Capture-Funktion mit dem aktuellen Zählwert oder mit dem Startwert weitergezählt wird. Bei Verwendung der steigenden Flanke des Signals N am Gebereingang wird im Anschluss an die Capture-Funktion mit dem aktuellen Zählwert weitergezählt. Die Verwendung eines Digitaleingangs und die Verwendung des Signals N schliessen sich für die Capture-Funktion nicht gegenseitig aus.

Der Parameter "Häufigkeit der Capture-Funktion" legt fest, ob die Funktion bei jeder parametrierten Flanke oder nur einmalig nach jeder Freigabe ausgeführt wird.

Messtaster

Wenn Sie die Positionserfassung für Motion Control (Seite 61) verwenden, können Sie das Technologieobjekt "Messtaster" dazu verwenden, mit einem Hardware-Digitaleingang eine Messtasterfunktion auszuführen.

Synchronisation

Sie können die Flanke eines externen Referenzsignals parametrieren, die den Zähler mit dem vorgegebenen Startwert lädt. Folgende externe Signale können den Zähler mit dem Startwert laden:

- Steigende oder fallende Flanke eines Digitaleingangs
- Steigende Flanke des Signals N am Gebereingang
- Steigende Flanke des Signals N am Gebereingang in Abhängigkeit des Pegels des zugewiesenen Digitaleingangs

Der Parameter "Häufigkeit der Synchronisation" legt fest, ob die Funktion bei jeder parametrierten Flanke oder nur einmalig nach jeder Freigabe ausgeführt wird.

Hysterese

Sie können für die Vergleichswerte eine Hysterese festlegen, innerhalb der das erneute Schalten eines Digitalausgangs verhindert wird. Ein Geber kann an einer bestimmten Position stehenbleiben und durch geringfügige Bewegungen schwankt der Zählwert um diese Position. Liegt in diesem Schwankungsbereich ein Vergleichswert oder eine Zählgrenze, wird ohne Verwendung einer Hysterese der zugehörige Digitalausgang entsprechend oft ein- und ausgeschaltet. Die Hysterese verhindert diese ungewollten Schaltvorgänge.

Positionserfassung mit SSI-Absolutwertgeber

Beschreibung

Sie können die Technologiemodule TM PosInput mit einem SSI-Absolutwertgeber zur Positionserfassung nutzen. Das Technologiemodul liest den Positionswert über eine synchrone, serielle Schnittstelle vom SSI-Absolutwertgeber und stellt ihn der Steuerung zur Verfügung.

Die Digitalausgänge des Technologiemoduls können Sie exakt an definierten Positionswerten unabhängig vom Anwenderprogramm schalten. Die Positionserfassung mit einem SSI-Absolutwertgeber erfolgt ohne Torsteuerung. Eine Synchronisation ist mit einem SSI-Absolutwertgeber systembedingt nicht möglich.

Gray- und Dual-Code

Es werden Gray- und Dual-codierte SSI-Absolutwertgeber unterstützt.

Capture (Latch)

Sie können eine oder beide Flanken eines Digitaleingangs parametrieren, die ein Speichern des aktuellen Positionswerts als Capture-Wert auslösen.

Der Parameter "Häufigkeit der Capture-Funktion" legt fest, ob die Funktion bei jeder parametrierten Flanke oder nur einmalig nach jeder Freigabe ausgeführt wird.

Messtaster

Wenn Sie die Positionserfassung für Motion Control (Seite 61) verwenden, können Sie das Technologieobjekt "Messtaster" dazu verwenden, mit einem Hardware-Digitaleingang eine Messtasterfunktion auszuführen.

Hysterese

Sie können für die Vergleichswerte eine Hysterese festlegen, innerhalb der das erneute Schalten eines Digitalausgangs verhindert wird. Ein Geber kann an einer bestimmten Position stehenbleiben und durch geringfügige Bewegungen schwankt der Positionswert um diese Position. Liegt in diesem Schwankungsbereich ein Vergleichswert oder eine Grenze, wird ohne Verwendung einer Hysterese der zugehörige Digitalausgang entsprechend oft ein- und ausgeschaltet. Die Hysterese verhindert diese ungewollten Schaltvorgänge.

Bereich für Positionswert

Sie können für den SSI-Absolutwertgeber eine Telegrammlänge von 10 Bit bis 40 Bit festlegen. Die parametrierbaren Bit-Nummern des LSB und des MSB des Positionswerts im Telegramm definieren den Wertebereich. Das Technologiemodul kann einen Positionswert mit einer Länge von maximal 32 Bit einlesen und an die Steuerung übermitteln.

Wenn Sie einen Geber mit einer Positionswertlänge bis 31 Bit verwenden, wird der Positionswert vorzeichenlos als positiver Wert behandelt und kann Werte zwischen 0 und $2^{(MSB-LSB+1)}-1$ annehmen. Wenn Sie einen Geber mit einer Positionswertlänge von 32 Bit verwenden, entspricht das MSB des Positionswerts dem Vorzeichen und der Positionswert kann Werte zwischen –2147483648 und 2147483647 annehmen. Wenn Sie einen 32-Bit-Positionswert für die Vergleichsfunktion verwenden, wird der Positionswert dabei als DINT interpretiert.

Vollständiges SSI-Telegramm

Sie können sich anstelle einer Messgröße die niederwertigsten 32 Bit des unbearbeiteten aktuellen SSI-Telegramms zurückliefern lassen. Dadurch erhalten Sie zusätzlich zum Positionswert geberspezifische Zusatz-Bits, z. B. Fehler-Bits. Wenn das SSI-Telegramm kürzer ist als 32 Bit, werden in der Rückmeldeschnittstelle das vollständige SSI-Telegramm rechtsbündig und die oberen ungenutzten Bits mit "0" zurückgeliefert.

Verhalten an den Zählgrenzen

Überschreiten einer Zählgrenze

Die obere Zählgrenze ist überschritten, wenn der aktuelle Zählwert der oberen Zählgrenze entspricht und ein weiterer Zählimpuls in Vorwärtsrichtung kommt. Die untere Zählgrenze ist überschritten, wenn der aktuelle Zählwert der unteren Zählgrenze entspricht und ein weiterer Zählimpuls in Rückwärtsrichtung kommt.

In der Rückmeldeschnittstelle wird beim Überschreiten das entsprechende Statusbit gesetzt:

Zählgrenze überschritten	Statusbit
Obere Zählgrenze	EVENT_OFLW wird gesetzt
Untere Zählgrenze	EVENT_UFLW wird gesetzt

Sie können die Statusbits mit RES_EVENT zurücksetzen.

Sie können für das Überschreiten einer Zählgrenze parametrieren, ob und mit welchem Zählwert weitergezählt wird.

Hinweis

Die obere Zählgrenze und der Startwert definieren den Wertebereich des Zählers:

Wertebereich des Zählers = (Obere Grenze – Startwert) + 1

Beispiele

Das folgende Bild zeigt ein Beispiel für das Beenden des Zählvorgangs (automatischer Torstopp) nach einem Überlauf und dem Setzen des Zählers auf den Startwert:

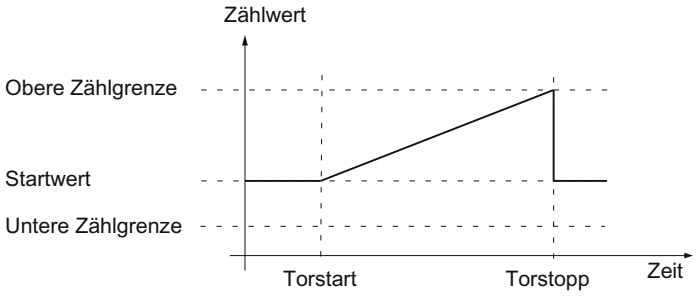

Das folgende Bild zeigt ein Beispiel für das Weiterzählen nach einem Überlauf und dem Setzen des Zählers auf den Startwert:

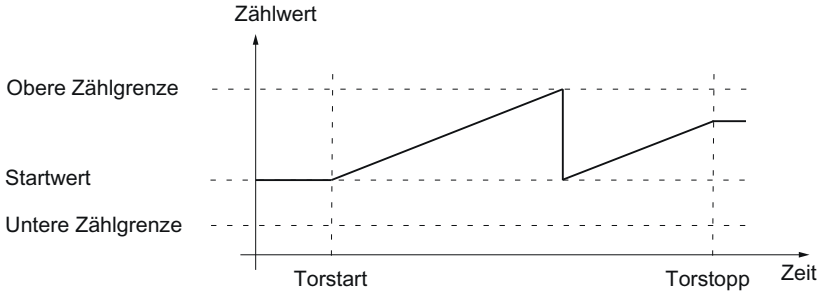

Das folgende Bild zeigt ein Beispiel für das Beenden des Zählvorgangs nach einem Überlauf und dem Setzen des Zählers auf die andere Zählgrenze:

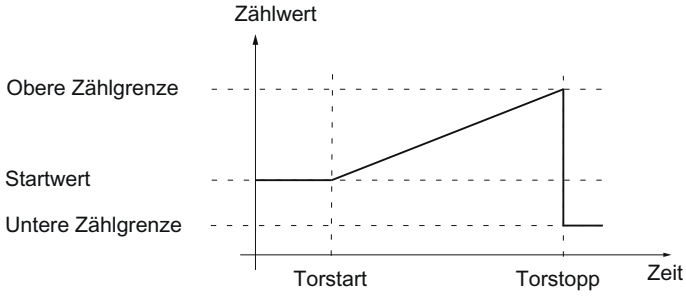

Das folgende Bild zeigt ein Beispiel für das Weiterzählen nach einem Überlauf und dem Setzen des Zählers auf die andere Zählgrenze:

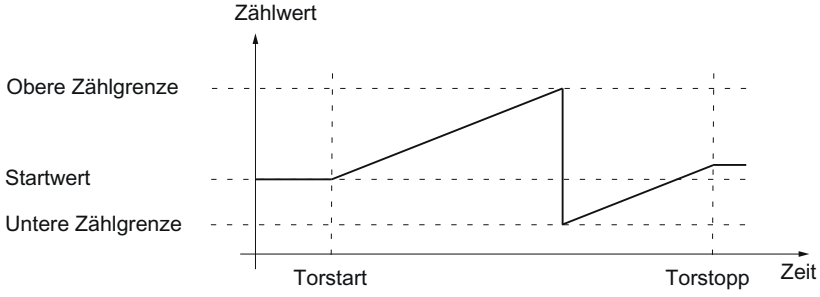

Torsteuerung mit Inkremental- oder Impulsgeber

Viele Anwendungen erfordern, dass der Zählvorgang abhängig von anderen Ereignissen gestartet oder gestoppt werden soll. Dieses Starten und Stoppen des Zählvorgangs geschieht über die Torfunktion.

Die Technologiemodule besitzen pro Kanal zwei Tore, die das resultierende interne Tor bestimmen:

- Software-Tor (SW-Tor)
- Hardware-Tor (HW-Tor)

Software-Tor

Sie öffnen und schließen das SW-Tor des Kanals mit dem Steuerbit (Seite 170) SW_GATE. Der Zustand des SW-Tors wird mit dem Rückmeldebit (Seite 172) STS_SW_GATE angezeigt.

Hardware-Tor

Das HW-Tor ist optional. Sie öffnen und schließen das HW-Tor über Signale an den parametrierten Digitaleingängen des Kanals.

Hinweis

Die parametrierten Eingangsfilter verzögern das Steuersignal des Digitaleingangs.

Der Zustand eines Digitaleingangs DIm wird mit dem jeweiligen Rückmeldebit (Seite 172) STS_DIm angezeigt.

Pegelgesteuertes Öffnen und Schließen des HW-Tors mit einem Digitaleingang

Das folgende Bild zeigt ein Beispiel für das pegelgesteuerte Öffnen und Schließen mit einem Digitaleingang. Der Digitaleingang ist so parametriert, dass er bei High-Pegel aktiv ist:

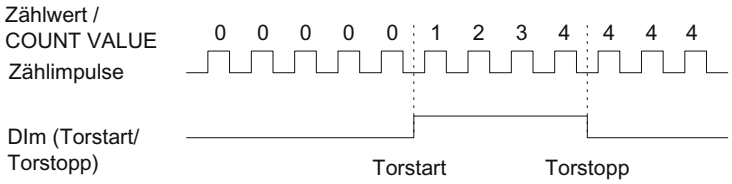

Solange der Digitaleingang aktiv ist, ist das HW-Tor geöffnet und die Zählimpulse werden gezählt. Wenn der Digitaleingang inaktiv wird, wird das HW-Tor geschlossen. Die Zählimpulse werden ignoriert und der Zählwert bleibt konstant.

Flankengesteuertes Öffnen und Schließen des HW-Tors mit zwei Digitaleingängen

Das folgende Bild zeigt ein Beispiel für das Öffnen und Schließen mit zwei Digitaleingängen. Die beiden Digitaleingänge sind so parametriert, dass die steigende Flanke ausgewertet wird:

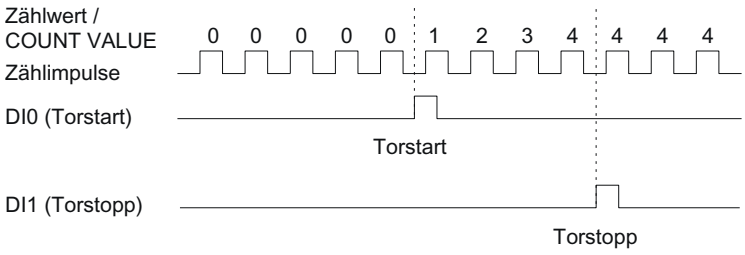

Bei der parametrierten Flanke am Digitaleingang, der für das Öffnen parametriert ist, wird das HW-Tor geöffnet. Bei der parametrierten Flanke am Digitaleingang, der für das Schließen parametriert ist, wird das HW-Tor geschlossen.

Internes Tor

Internes Tor

Das interne Tor ist offen, wenn das SW-Tor offen ist und das HW-Tor offen oder nicht parametriert ist. Der Zustand des internen Tors wird mit dem Rückmeldebit (Seite 172) STS_GATE angezeigt.

Wenn das interne Tor geöffnet ist, wird der Zählvorgang gestartet. Wenn das interne Tor geschlossen ist, werden weitere Zählimpulse ignoriert und der Zählvorgang ist gestoppt.

Wenn Sie einen Zählvorgang nur mit dem HW-Tor kontrollieren wollen, muss das SW-Tor offen sein. Wenn Sie kein HW-Tor parametrieren, gilt das HW-Tor immer als offen. Sie öffnen und schließen das interne Tor dann nur über das SW-Tor.

HW-Tor	SW-Tor	Internes Tor
offen/nicht parametriert	offen	offen
offen/nicht parametriert	geschlossen	geschlossen
geschlossen	offen	geschlossen
geschlossen	geschlossen	geschlossen

Sie legen bei der Parametrierung des Zählerverhaltens fest, ob der Zählvorgang nach Öffnen des internen Tors mit dem Startwert oder dem aktuellen Zählwert beginnt.

Zusätzlich kann das interne Tor bei Überschreiten einer Zählgrenze automatisch geschlossen werden. Zum Fortsetzen des Zählvorgangs ist das Schließen und erneute Öffnen des Software-Tors oder Hardware-Tors notwendig.

Zählerverhalten bei Torstart

Sie haben für das Verhalten des Zählers beim Torstart folgende Parametriermöglichkeiten:

- Zähler auf den Startwert setzen
- Mit dem aktuellen Zählwert fortsetzen

Zähler auf den Startwert setzen

Bei dieser Parametrierung gilt folgendes Zählerverhalten:

Jeder Zählvorgang beginnt nach dem Öffnen des internen Tors mit dem Startwert.

Das folgende Bild zeigt ein Beispiel für das Fortsetzen des Zählvorgangs nach dem Setzen des Zählers auf den Startwert:

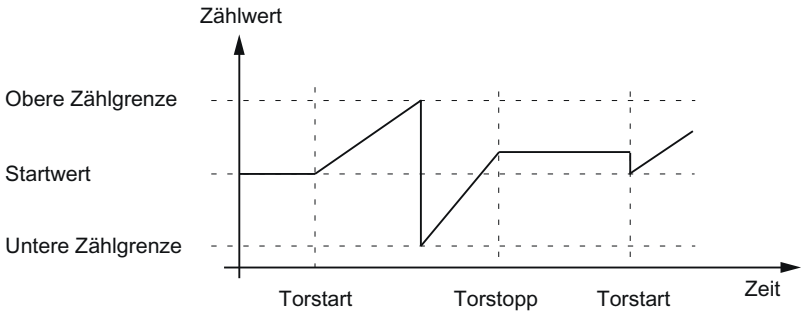

Mit dem aktuellen Zählwert fortsetzen

Bei dieser Parametrierung gilt folgendes Zählerverhalten:

Jeder Zählvorgang beginnt nach dem wiederholten Öffnen des internen Tors mit dem aktuellen Zählwert.

Das folgende Bild zeigt ein Beispiel für das Fortsetzen des Zählvorgangs mit dem aktuellen Zählwert:

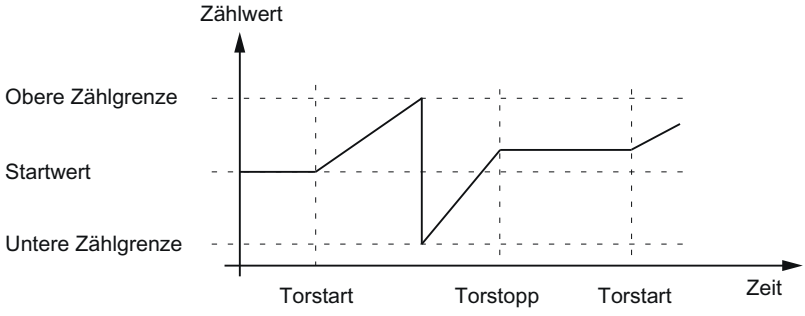

Capture (Latch)

Capture mit Inkremental- oder Impulsgeber

Beschreibung

Mit der Funktion "Capture" speichern Sie durch ein externes Referenzsignal den aktuellen Zählwert. Sie können die Capture-Funktion für folgende Referenzsignale parametrieren:

- Steigende oder fallende Flanke an einem Digitaleingang
- Steigende und fallende Flanke an einem Digitaleingang
- Steigende Flanke des Signals N am Gebereingang

Funktionsweise

Der Capture-Wert ist immer genau der Zählwert zum Zeitpunkt der jeweiligen Flanke (verzögert um die parametrierte Eingangsverzögerung). Die Capture-Funktion wirkt unabhängig vom Zustand des internen Tors. Bei geschlossenem Tor wird der unveränderte Zählwert gespeichert.

Das folgende Bild zeigt ein Beispiel für die Capture-Funktion bei folgender Parametrierung:

- Startwert = 0
- Capture-Ereignis bei steigender Flanke am parametrierten Digitaleingang
- Zähler bei Torstart auf den Startwert setzen
- Nach Capture-Ereignis Zählen fortsetzen

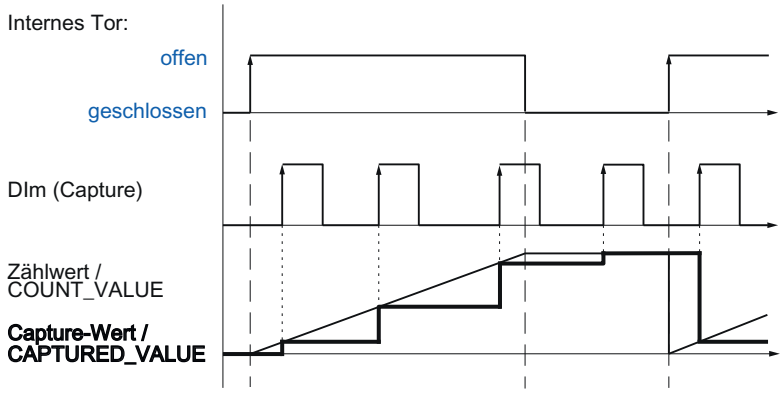

Das folgende Bild zeigt ein weiteres Beispiel für die Capture-Funktion bei folgender Parametrierung:

- Startwert = 0
- Capture-Ereignis bei steigender Flanke am parametrierten Digitaleingang
- Zähler bei Torstart auf den Startwert setzen
- Nach Capture-Ereignis Zählwert auf Startwert setzen und Zählen fortsetzen

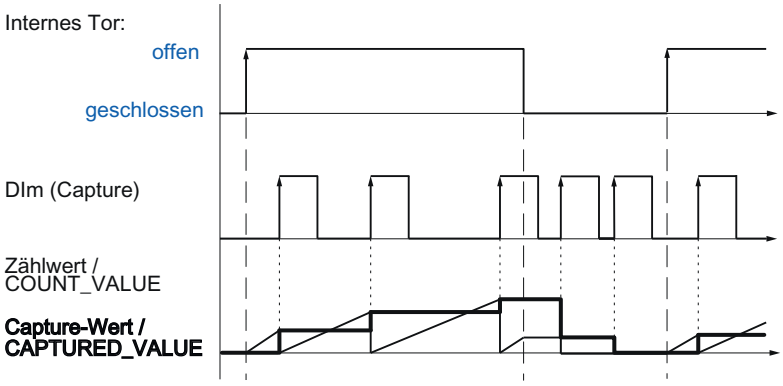

Mit dem Steuerbit (Seite 170) EN_CAPTURE geben Sie die Capture-Funktion frei. Das Rückmeldebit (Seite 172) EVENT_CAP zeigt an, dass ein Zählwert als Capture-Wert in der Rückmeldeschnittstelle gespeichert wurde. Wenn Sie EN_CAPTURE rücksetzen, wird

EVENT_CAP rückgesetzt. Der Zustand eines Digitaleingangs wird mit dem jeweiligen Rückmeldebit (Seite 172) STS_DIm angezeigt.

Das folgende Bild zeigt ein Beispiel für die Bits EN_CAPTURE und EVENT_CAP bei Nutzung der einmaligen Capture-Funktion durch die steigende Flanke an einem Digitaleingang:

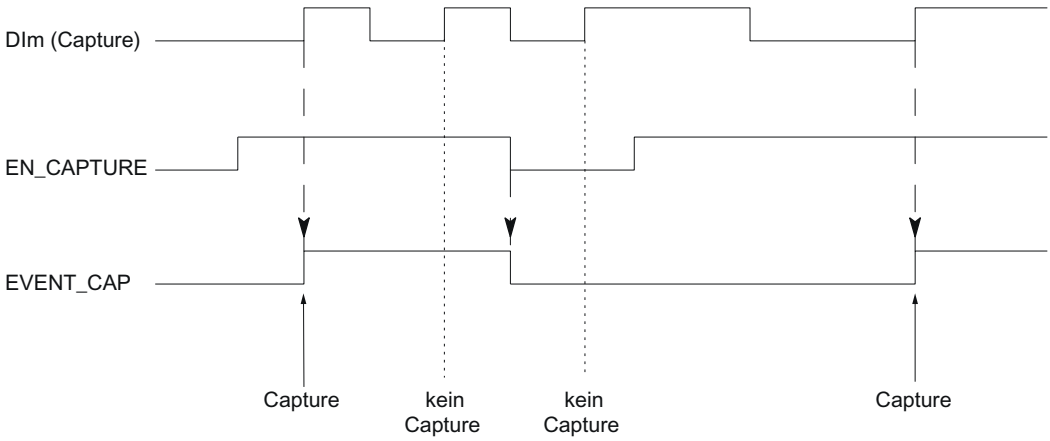

Das folgende Bild zeigt ein Beispiel für die Bits EN_CAPTURE und EVENT_CAP bei Nutzung der periodischen Capture-Funktion durch die steigende Flanke an einem Digitaleingang:

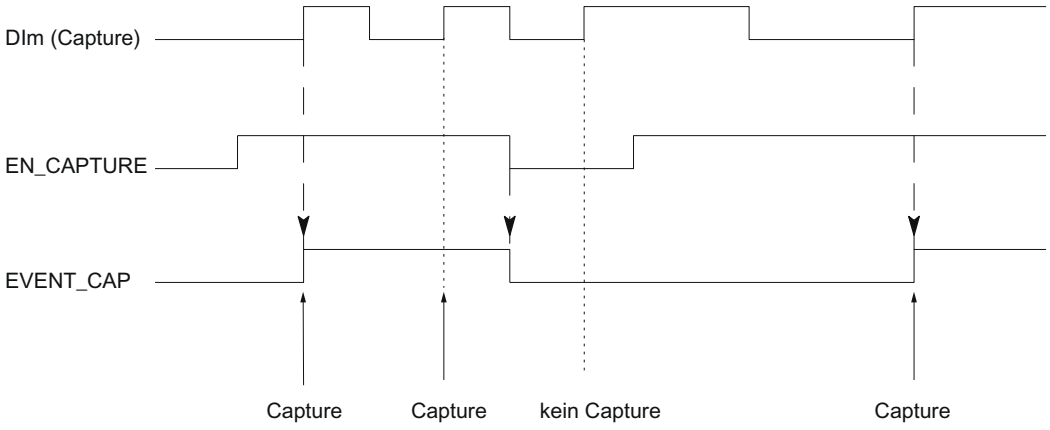

Hinweis

Die parametrierten Eingangsfilter verzögern das Steuersignal des entsprechenden Digitaleingangs.

Die Capture-Funktion hat keinen Einfluss auf das Rückmeldebit STS_CNT und die LEDs UP und DN.

Prozessalarm

Sie können für die Capture-Funktion einen Prozessalarm parametrieren. Wenn die Prozessalarme schneller ausgelöst werden, als sie vom System quittiert werden können, gehen Prozessalarme verloren und der Diagnosealarm "Prozessalarm verloren" wird gemeldet.

Capture mit SSI-Absolutwertgeber

Beschreibung

Mit der Funktion "Capture" speichern Sie durch ein externes Referenzsignal den aktuellen Positionswert. Sie können die Capture-Funktion für folgende Referenzsignale parametrieren:

- Steigende **oder** fallende Flanke an einem Digitaleingang
- Steigende **und** fallende Flanke an einem Digitaleingang

Funktionsweise

Zum Zeitpunkt der jeweiligen Flanke wird der Positionswert des letzten gültigen SSI-Telegramms in den Capture-Wert gespeichert.

Das folgende Bild zeigt ein Beispiel für das Capture-Ereignis durch die steigende Flanke am parametrierten Digitaleingang:

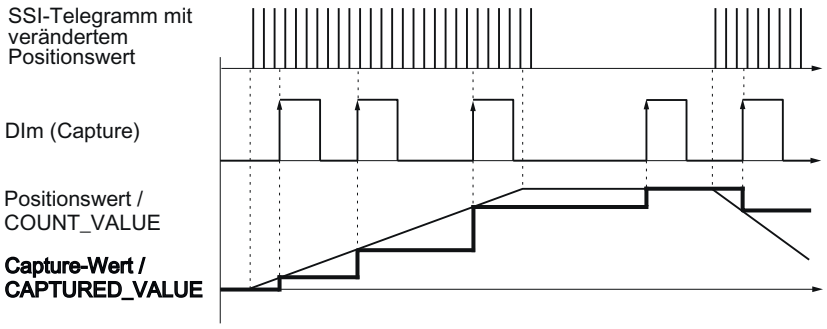

Mit dem Steuerbit (Seite 170) EN_CAPTURE geben Sie die Capture-Funktion frei. Das Rückmeldebit (Seite 172) EVENT_CAP zeigt an, dass ein Positionswert als Capture-Wert in der Rückmeldeschnittstelle gespeichert wurde. Wenn Sie EN_CAPTURE rücksetzen, wird EVENT_CAP rückgesetzt. Der Zustand eines Digitaleingangs wird mit dem jeweiligen Rückmeldebit (Seite 172) STS_DIm angezeigt.

Das folgende Bild zeigt ein Beispiel für die Bits EN_CAPTURE und EVENT_CAP bei Nutzung der einmaligen Capture-Funktion durch die steigende Flanke an einem Digitaleingang:

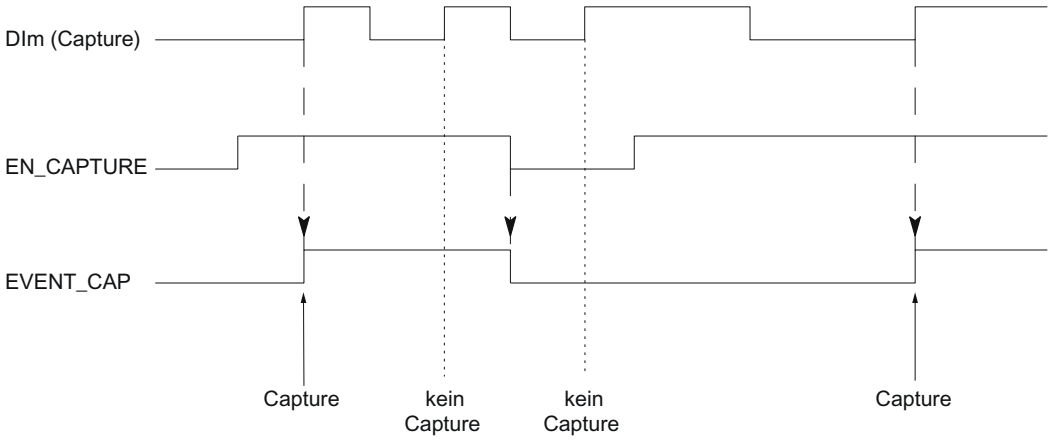

Das folgende Bild zeigt ein Beispiel für die Bits EN_CAPTURE und EVENT_CAP bei Nutzung der periodischen Capture-Funktion durch die steigende Flanke an einem Digitaleingang:

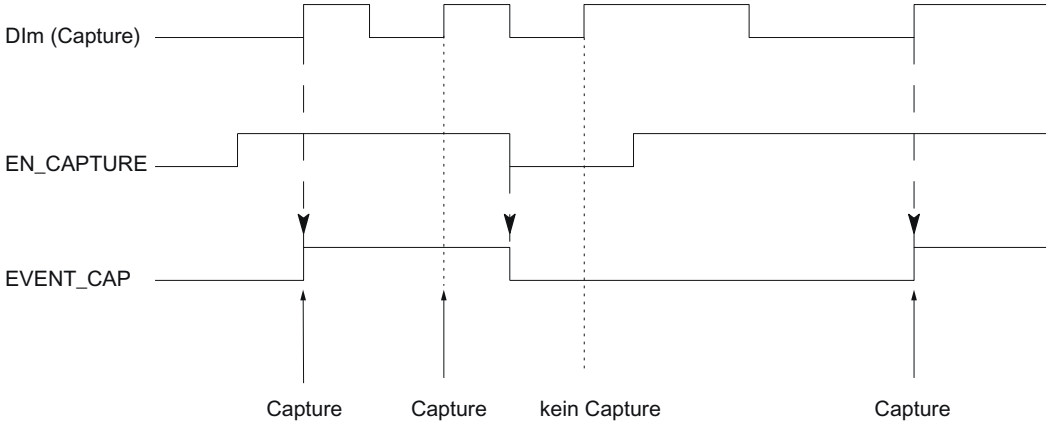

Hinweis

Die parametrierten Eingangsfilter verzögern das Steuersignal des entsprechenden Digitaleingangs.

Prozessalarm

Sie können für die Capture-Funktion einen Prozessalarm parametrieren. Wenn die Prozessalarme schneller ausgelöst werden, als sie vom System quittiert werden können, gehen Prozessalarme verloren und der Diagnosealarm "Prozessalarm verloren" wird gemeldet.

Synchronisation

Beschreibung

Mit der Funktion "Synchronisation" setzen Sie durch ein externes Referenzsignal den Zähler auf den vorgegebenen Startwert. Sie können die Synchronisation für folgende Referenzsignale parametrieren:

- Steigende oder fallende Flanken an einem Digitaleingang
- Steigende Flanke des Signals N am Gebereingang
- Steigende Flanke des Signals N am Gebereingang in Abhängigkeit des Pegels eines Digitaleingangs

Funktionsweise

Die Synchronisation findet immer genau zum Zeitpunkt des Referenzsignals statt. Die Synchronisation wirkt unabhängig vom Zustand des internen Tors.

Mit dem Steuerbit (Seite 170) EN_SYNC_UP geben Sie die Synchronisation für das Zählen in Vorwärtsrichtung frei. Mit dem Steuerbit (Seite 170) EN_SYNC_DN geben Sie die Synchronisation für das Zählen in Rückwärtsrichtung frei. Das Rückmeldebit (Seite 172) EVENT_SYNC zeigt an, dass eine Synchronisation stattgefunden hat. Durch das Rücksetzen von EN_SYNC_UP oder EN_SYNC_DN wird auch EVENT_SYNC rückgesetzt.

Hinweis

Die parametrierten Eingangsfilter verzögern das Steuersignal des entsprechenden Digitaleingangs.

Die Synchronisation hat keinen Einfluss auf das Rückmeldebit (Seite 172) STS_CNT und die LEDs UP und DN.

Einmalige Synchronisation

Das folgende Bild zeigt ein Beispiel für die Bits EN_SYNC_UP, EN_SYNC_DN und EVENT_SYNC bei einmaliger Synchronisation durch eine Flanke an einem Digitaleingang bei Zählimpulsen in Vorwärtsrichtung:

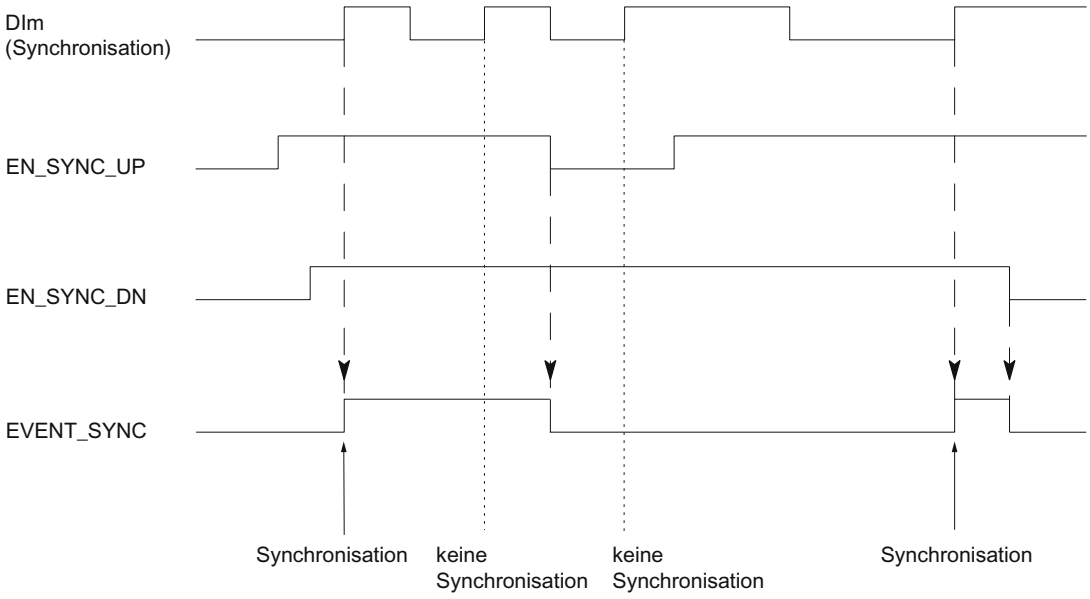

Nach der Freigabe der Synchronisation für das Zählen in Vorwärtsrichtung wird der Zähler bei der ersten steigenden Flanke am parametrierten Digitaleingang synchronisiert. Erst nach Rücksetzen und erneutem Setzen des Steuerbits (Seite 170) EN_SYNC_UP kann der Zähler erneut synchronisiert werden.

Periodische Synchronisation

Das folgende Bild zeigt ein Beispiel für die Bits EN_SYNC_UP, EN_SYNC_DN und EVENT_SYNC bei periodischer Synchronisation durch eine Flanke an einem Digitaleingang bei Zählimpulsen in Vorwärtsrichtung:

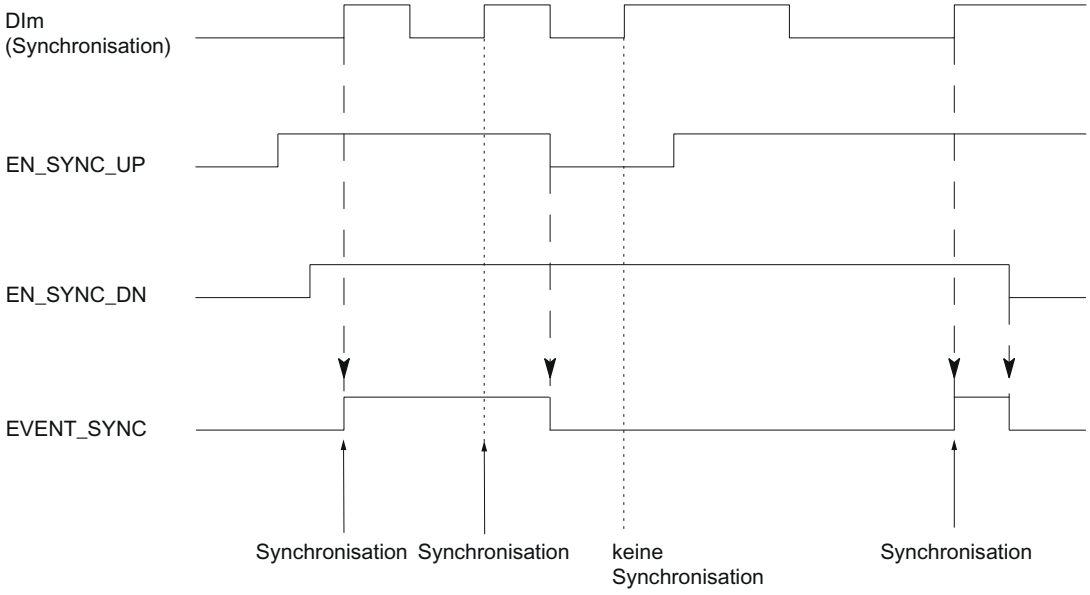

Solange die Synchronisation für das Zählen in Vorwärtsrichtung freigegeben ist, wird der Zähler bei jeder steigenden Flanke am parametrierten Digitaleingang synchronisiert.

Prozessalarm

Sie können für die Synchronisation einen Prozessalarm parametrieren. Wenn die Prozessalarme schneller ausgelöst werden, als sie vom System quittiert werden können, gehen Prozessalarme verloren und der Diagnosealarm "Prozessalarm verloren" wird gemeldet.

Synchronisation durch Digitaleingang

Sie können die Synchronisation durch Flanken an einem Digitaleingang auslösen.

Einmalige Synchronisation

Das folgende Bild zeigt ein Beispiel für einmalige Synchronisation durch eine Flanke an einem Digitaleingang:

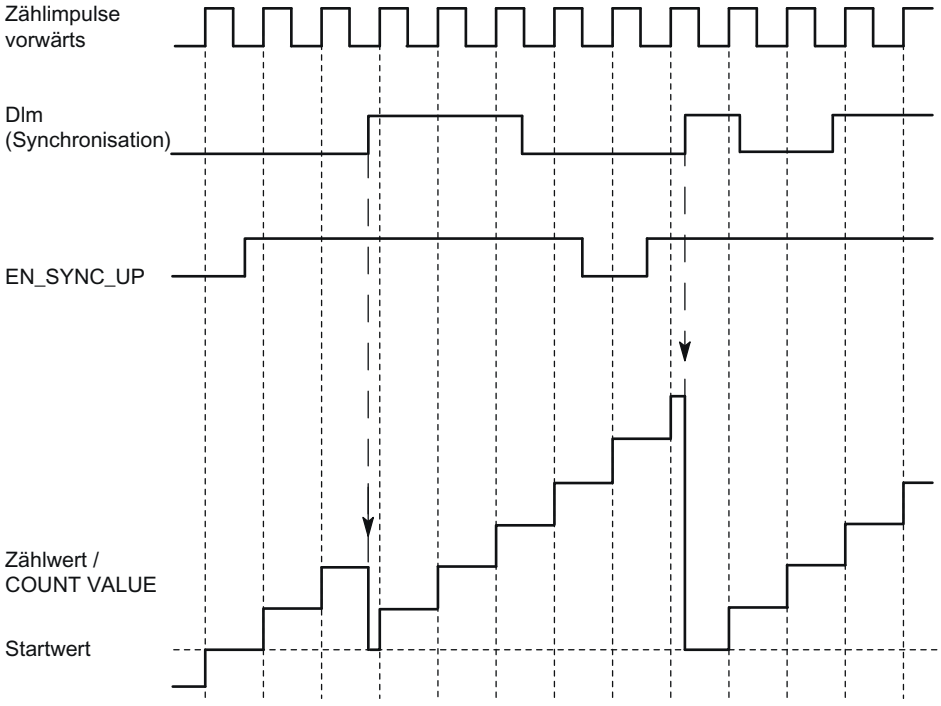

Nach der Freigabe der Synchronisation für das Zählen in Vorwärtsrichtung wird der Zähler bei der ersten steigenden Flanke am parametrierten Digitaleingang synchronisiert. Bis zum Rücksetzen und erneutem Setzen des Steuerbits (Seite 170) EN_SYNC_UP wird eine erneute steigende Flanke am Digitaleingang ignoriert. Anschließend kann der Zähler erneut synchronisiert werden.

Periodische Synchronisation

Das folgende Bild zeigt ein Beispiel für periodische Synchronisation durch eine Flanke an einem Digitaleingang:

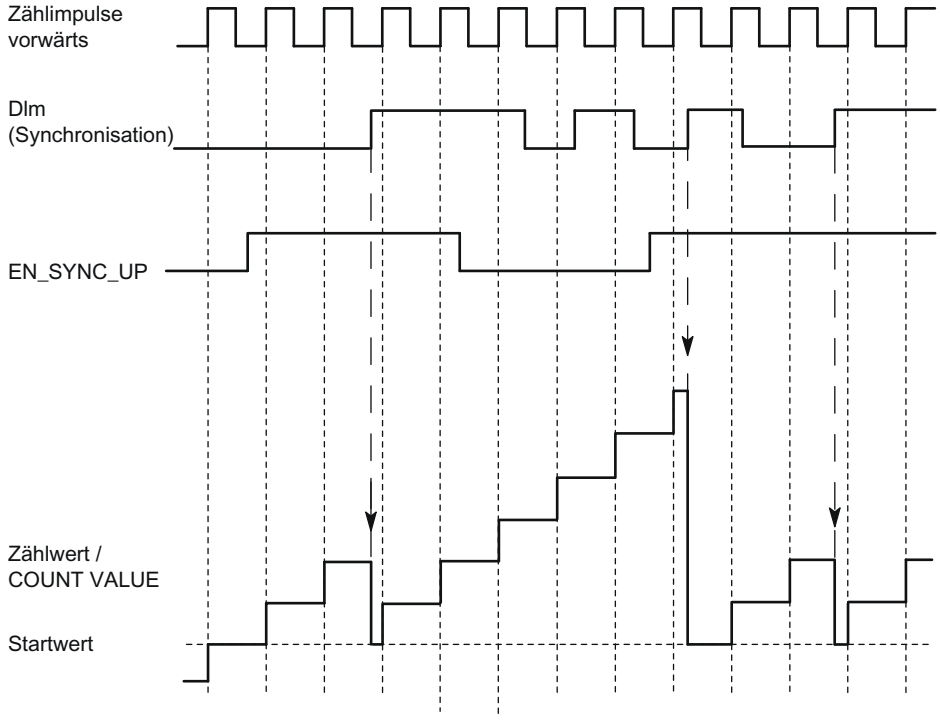

Solange die Synchronisation für das Zählen in Vorwärtsrichtung freigegeben ist, wird der Zähler bei jeder steigenden Flanke am parametrierten Digitaleingang synchronisiert.

Synchronisation bei Signal N

Sie können die Synchronisation bei Signal N am Gebereingang abhängig oder unabhängig vom Zustand eines Digitaleingangs auslösen.

Einmalige Synchronisation

Das folgende Bild zeigt ein Beispiel für einmalige Synchronisation bei Signal N (unabhängig von einem Digitaleingang):

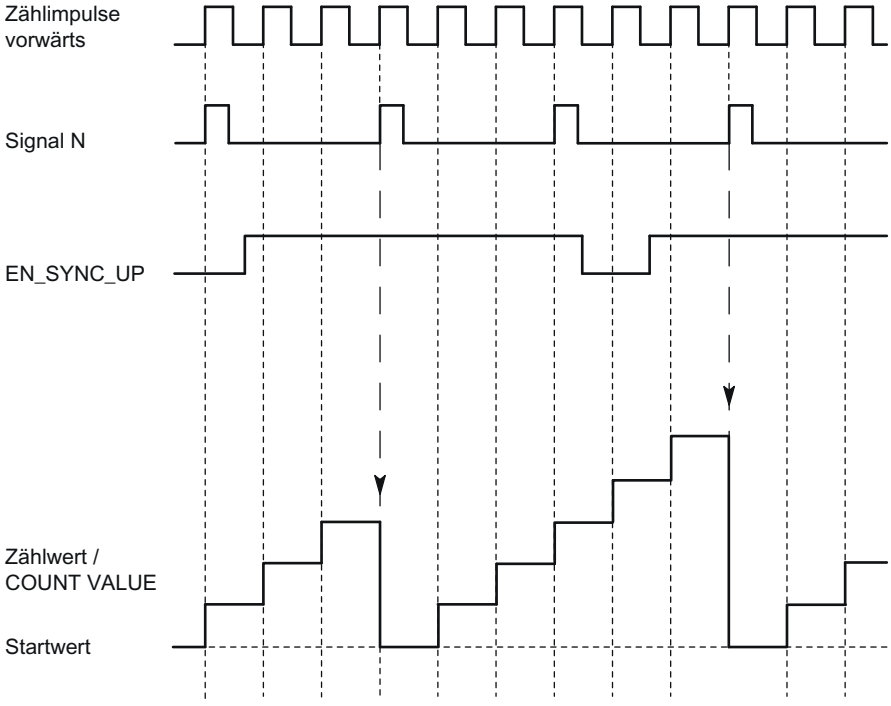

Nach der Freigabe der Synchronisation für das Zählen in Vorwärtsrichtung wird der Zähler beim ersten Signal N synchronisiert. Nach Rücksetzen und erneutem Setzen des Steuerbits (Seite 170) EN_SYNC_UP kann der Zähler erneut synchronisiert werden.

Periodische Synchronisation

Das folgende Bild zeigt ein Beispiel für periodische Synchronisation bei Signal N:

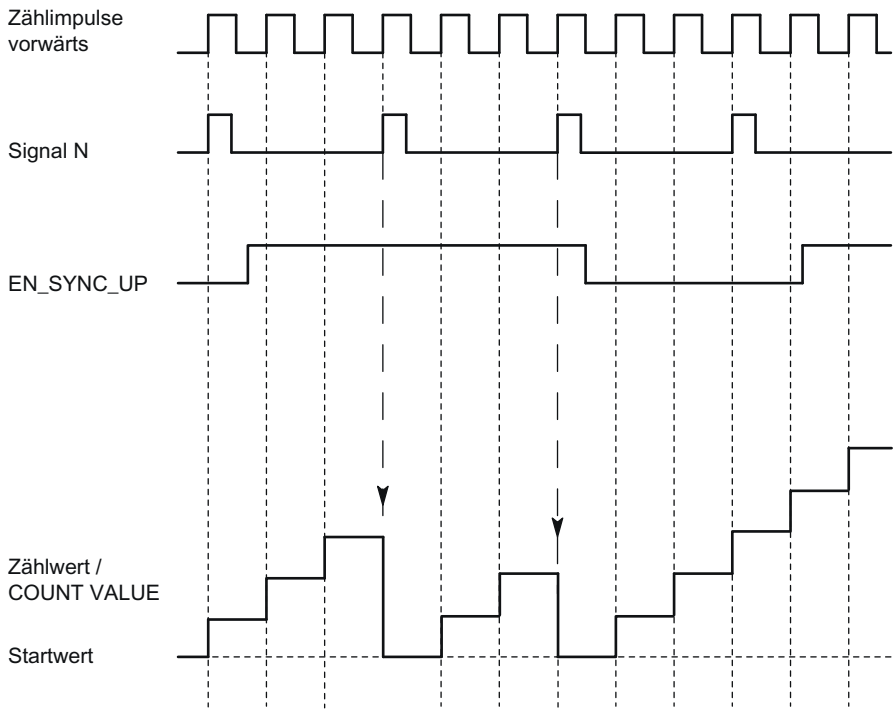

Solange die Synchronisation für das Zählen in Vorwärtsrichtung freigegeben ist, wird der Zähler bei jedem Signal N synchronisiert.

Freigabe durch einen Digitaleingang

Das folgende Bild zeigt ein Beispiel für periodische Synchronisation bei Signal N in Abhängigkeit des Zustands eines Digitaleingangs:

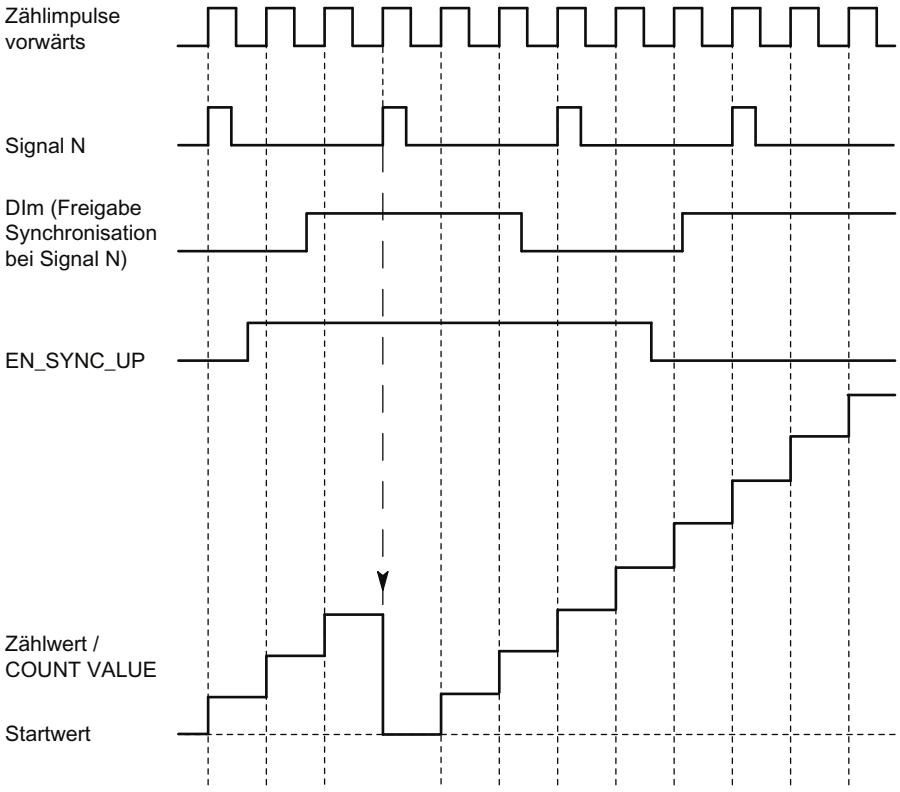

Solange die Synchronisation für das Zählen in Vorwärtsrichtung freigegeben ist und der zugehörige Digitaleingang aktiv ist, wird der Zähler bei jedem Signal N synchronisiert. Wenn eine der beiden Bedingungen nicht erfüllt ist, wird der Zähler bei Signal N nicht synchronisiert.

Vergleichswerte

Vergleichswerte und Ausgänge

Beschreibung

Sie können zwei Vergleichswerte festlegen, welche die beiden Digitalausgänge des Kanals unabhängig vom Anwenderprogramm steuern können:

- Vergleichswert 0 für Digitalausgang DQ0
- Vergleichswert 1 für Digitalausgang DQ1

Abhängig von der Betriebsart und dem verwendeten Geber legen Sie als Vergleichswerte zwei Positions-, Zähl- oder Messwerte fest. Die Vergleichswerte sind parametrierbar und zur Laufzeit über das Anwenderprogramm änderbar.

Hinweis

DQ0 eines Zählers einer Kompakt-CPU

Bei einer Kompakt-CPU ist der jeweilige Digitalausgang DQ0 über die Rückmeldeschnittstelle, aber nicht als physikalischer Ausgang verfügbar.

Schalten der Digitalausgänge aus dem Anwenderprogramm

Mit den Steuerbits (Seite 170) TM_CTRL_DQ0 und TM_CTRL_DQ1 steuern Sie die Verwendung der Digitalausgänge.

Wenn TM_CTRL_DQm auf 0 gesetzt ist, können Sie den jeweiligen Digitalausgang unabhängig von der projektierten technologischen Funktion mit dem Steuerbit SET_DQm aus dem Anwenderprogramm steuern. Wenn TM_CTRL_DQm auf 1 gesetzt ist, ist die technologische Funktion zur Steuerung des jeweiligen Digitalausgangs freigegeben.

Der Zustand eines Digitalausgangs wird mit dem jeweiligen Rückmeldebit STS_DQm gemeldet.

Schalten an Vergleichswerten mit Zählwert als Bezug

Die Vergleichswerte werden mit dem aktuellen Zählwert verglichen. Wenn der Zählwert die parametrierte Vergleichsbedingung erfüllt und die technologische Funktion des zugehörigen Digitalausgangs freigegeben ist, wird der Digitalausgang gesetzt. Wenn Sie für den Digitalausgang DQ1 "Zwischen Vergleichswert 0 und 1" parametrieren, wirken sich beide Vergleichswerte auf DQ1 aus.

Sie können das Schalten eines Digitalausgangs von einer der folgenden Vergleichsereignisse abhängig machen:

Setzen zwischen Vergleichswert und oberer Zählgrenze

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

Vergleichswert <= Zählwert <= obere Zählgrenze

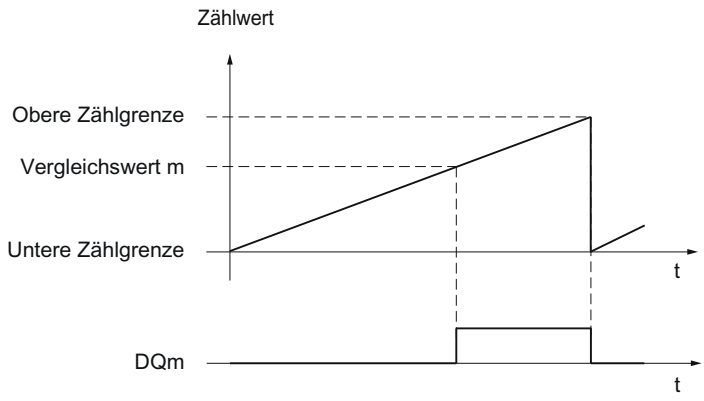

Das Vergleichsereignis ist unabhängig von der Zählrichtung.

Setzen zwischen Vergleichswert und unterer Zählgrenze

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

untere Zählgrenze <= Zählwert <= Vergleichswert

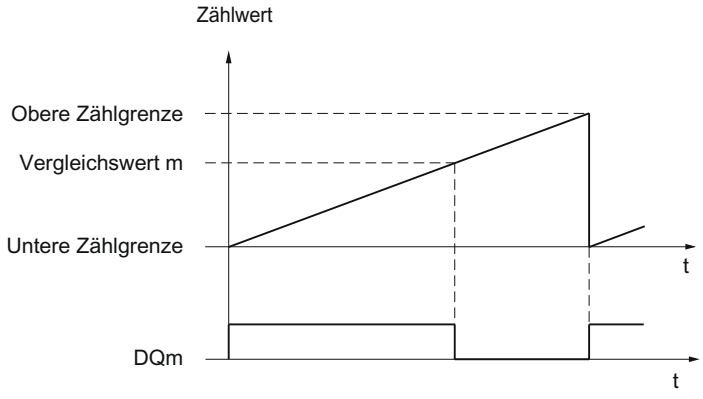

Das Vergleichsereignis ist unabhängig von der Zählrichtung.

Setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das Vergleichsereignis ist für den Digitalausgang DQ1 parametrierbar, wenn für den Digitalausgang DQ0 "Nutzung durch Anwenderprogramm" parametriert ist. Vergleichswert 1 muss größer sein als Vergleichswert 0.

DQ1 wird auf 1 gesetzt, wenn:

Vergleichswert 0 <= Zählwert <= Vergleichswert 1

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis beim Zählen in Vorwärtsrichtung:

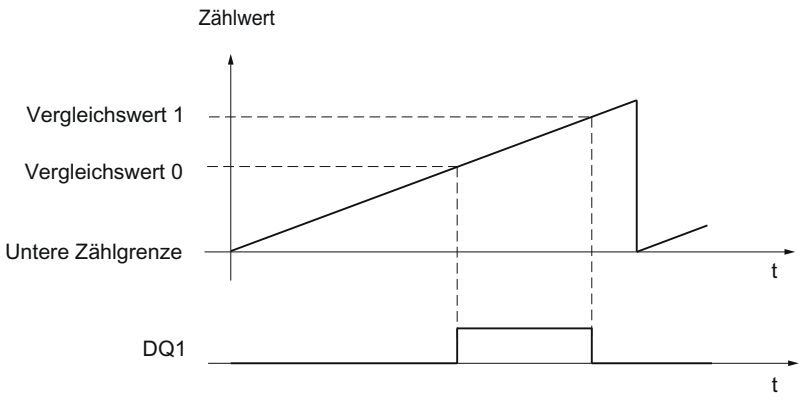

Setzen bei Vergleichswert für eine Impulsdauer

Der jeweilige Digitalausgang wird für eine festgelegte Zeit auf 1 gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Zählwert = Vergleichswert
- Aktuelle Zählrichtung = parametrierte Zählrichtung für das Vergleichsereignis

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis beim Zählen in Vorwärtsrichtung:

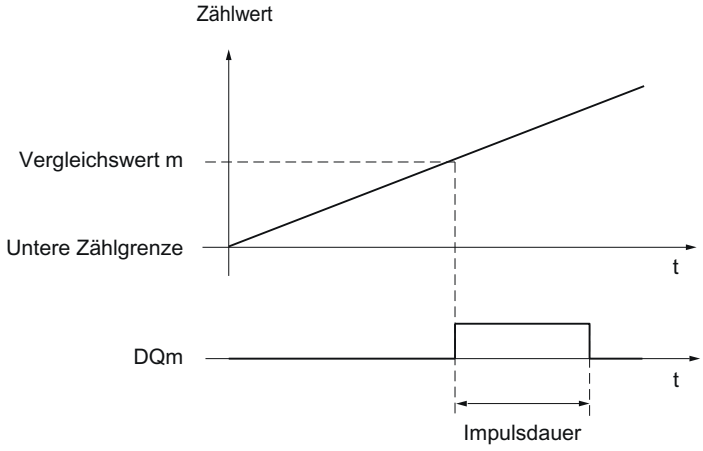

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis beim Zählen in Rückwärtsrichtung:

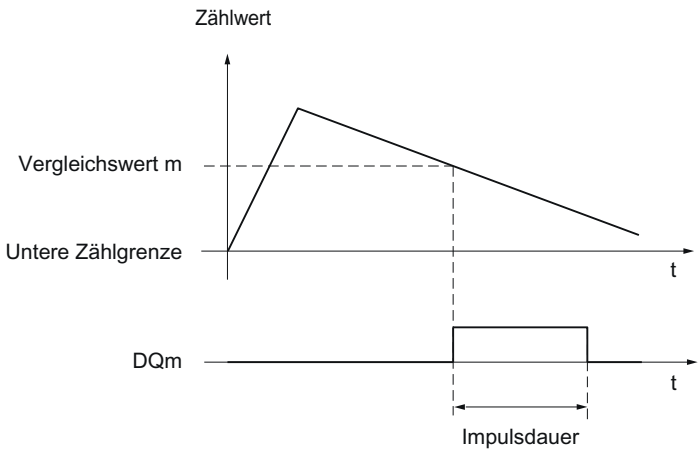

Um das Vergleichsereignis zu wiederholen, muss sich der Zählwert ändern und dem jeweiligen Vergleichswert erneut entsprechen.

Wenn als Impulsdauer "0" festgelegt ist und der Zählwert dem Vergleichswert entspricht, dann wird der jeweilige Digitalausgang bis zum nächsten Zählimpuls auf 1 gesetzt:

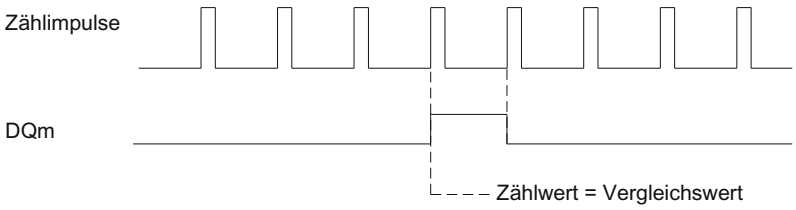

Hinweis

Dieses Vergleichsereignis schaltet den jeweiligen Digitalausgang, wenn der Vergleichswert durch einen Zählimpuls erreicht wird. Beim Setzen des Zählwerts, z. B. durch Synchronisation, schaltet der Digitalausgang nicht.

Setzen durch Anwenderprogramm bis zum Vergleichswert

Sie können durch Setzen des Steuerbit (Seite 170) SET_DQm den jeweiligen Digitalausgang auf 1 setzen (Flanke). Der jeweilige Digitalausgang wird bei einem der folgenden Ereignisse auf 0 gesetzt:

- Übereinstimmen von Zählwert und Vergleichswert in der parametrierten Richtung des Vergleichsereignisses
- Rücksetzen des zugehörigen Steuerbits SET_DQm

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis beim Zählen in Vorwärtsrichtung:

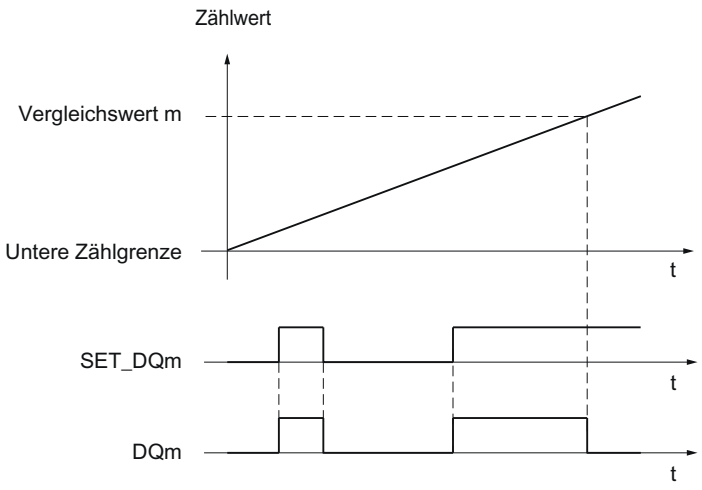

Sie können durch Setzen des Steuerbits SET_DQm auf 0 den Digitalausgang inaktiv schalten, bevor der Zählwert dem Vergleichswert entspricht.

Hinweis

Wenn der Vergleichswert in der parametrierten Zählrichtung erreicht wird, wird das Rückmeldebit EVENT_CMPm unabhängig vom Zustand des Steuerbits SET_DQm gesetzt.

Das Vergleichsereignis schaltet einen Digitalausgang dann, wenn der jeweilige Vergleichswert durch einen Zählimpuls erreicht wird. Beim Setzen des Zählwerts, z. B. durch Synchronisation, schaltet der Digitalausgang nicht.

Schalten an Vergleichswerten mit Positionswert (SSI-Absolutwert) als Bezug

Die Vergleichswerte werden mit dem aktuellen Positionswert verglichen. Wenn der Positionswert die parametrierte Vergleichsbedingung erfüllt und die technologische Funktion des zugehörigen Digitalausgangs freigegeben ist, wird der Digitalausgang gesetzt. Wenn Sie für den Digitalausgang DQ1 "Zwischen Vergleichswert 0 und 1" parametrieren, wirken sich beide Vergleichswerte auf DQ1 aus.

Wenn Sie einen 32-Bit-Positionswert für die Vergleichsfunktion verwenden, wird der Positionswert dabei als DINT interpretiert.

Sie können das Schalten eines Digitalausgangs von einer der folgenden Vergleichsereignisse abhängig machen:

Setzen zwischen Vergleichswert und oberer Grenze

Die obere Grenze entspricht dem maximalen Positionswert.

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

Vergleichswert <= Positionswert <= maximaler Positionswert

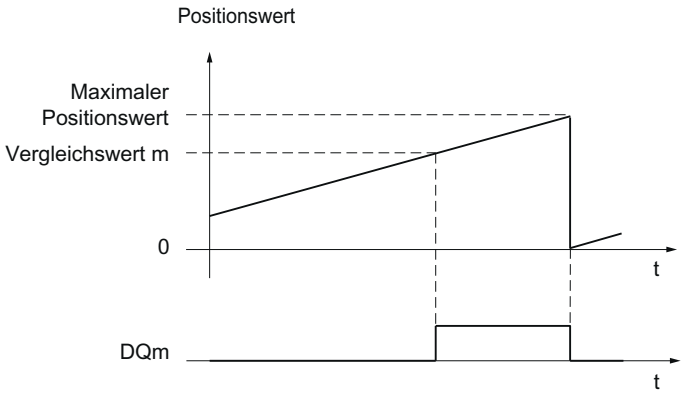

Das Vergleichsereignis ist unabhängig von der Richtung der Positionswertänderung. Der maximale Positionswert ist abhängig von der Auflösung des SSI-Absolutwertgebers.

Setzen zwischen Vergleichswert und unterer Grenze

Die untere Grenze entspricht dem Positionswert "0".

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

0 <= Positionswert <= Vergleichswert

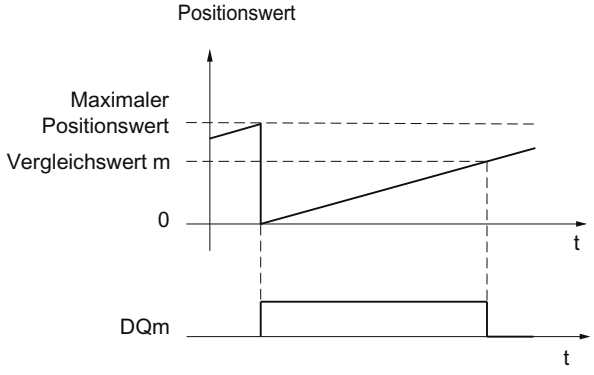

Das Vergleichsereignis ist unabhängig von der Richtung der Positionswertänderung.

Setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das Vergleichsereignis ist für den Digitalausgang DQ1 parametrierbar, wenn für den Digitalausgang DQ0 "Nutzung durch Anwenderprogramm" parametriert ist. Vergleichswert 1 muss größer sein als Vergleichswert 0.

DQ1 wird auf 1 gesetzt, wenn:

Vergleichswert 0 <= Positionswert <= Vergleichswert 1

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis in Vorwärtsrichtung:

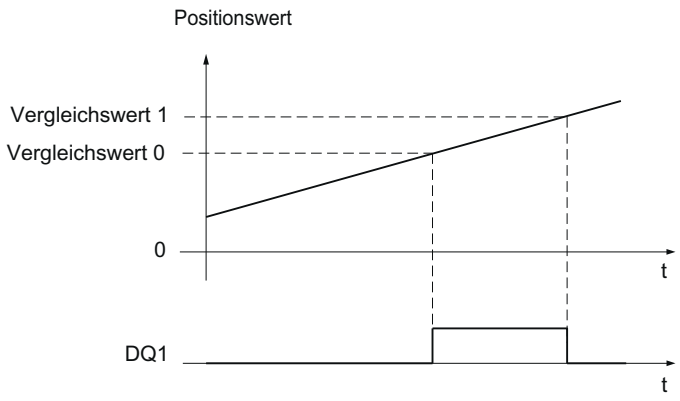

Setzen bei Vergleichswert für eine Impulsdauer

Der jeweilige Digitalausgang wird für eine festgelegte Zeit auf 1 gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Übereinstimmen von Positionswert und Vergleichswert oder Überqueren des Vergleichswerts
- Aktuelle Richtung der Positionswertänderung = parametrierte Richtung für das Vergleichsereignis

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis in Vorwärtsrichtung:

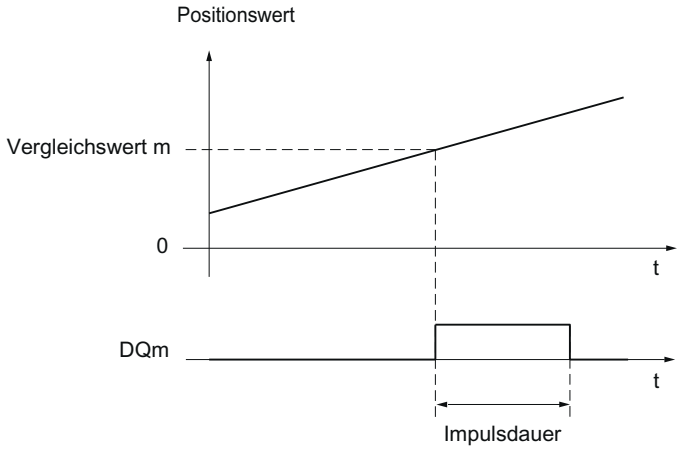

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis in Rückwärtsrichtung:

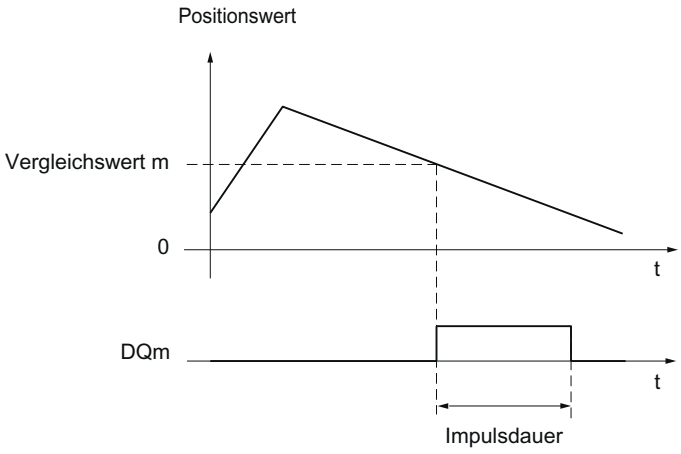

Um das Vergleichsereignis zu wiederholen, muss sich der Positionswert ändern und dem jeweiligen Vergleichswert erneut entsprechen oder ihn überqueren.

Setzen durch Anwenderprogramm bis zum Vergleichswert

Sie können durch Setzen des Steuerbits (Seite 170) SET_DQm den jeweiligen Digitalausgang auf 1 setzen (Flanke). Der jeweilige Digitalausgang wird bei einem der folgenden Ereignisse auf 0 gesetzt:

- Übereinstimmen von Positionswert und Vergleichswert oder Überqueren des Vergleichswerts in der parametrierten Richtung des Vergleichsereignisses
- Rücksetzen des zugehörigen Steuerbits SET_DQm

Das folgende Bild zeigt ein Beispiel für das Vergleichsereignis in Vorwärtsrichtung:

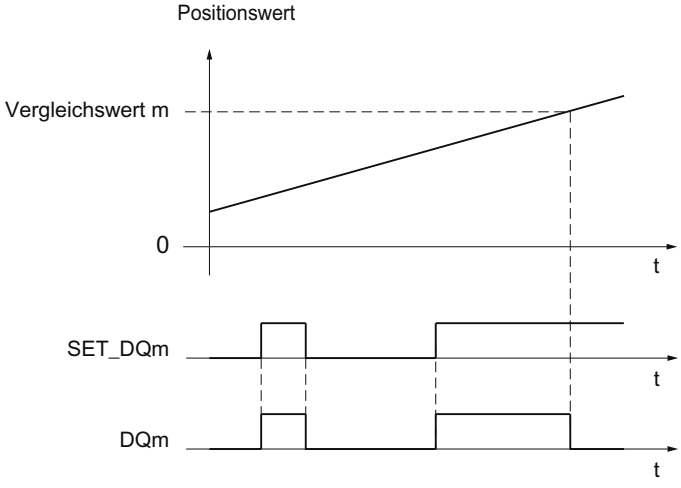

Sie können durch Setzen des Steuerbits SET_DQm auf 0 den Digitalausgang inaktiv schalten, bevor der Positionswert dem Vergleichswert entspricht oder ihn überquert.

Hinweis

Wenn der Vergleichswert in der parametrierten Richtung erreicht oder überquert wird, wird das Rückmeldebit EVENT_CMPm unabhängig vom Zustand des Steuerbits SET_DQm gesetzt.

Schalten an Vergleichswerten mit Messwert als Bezug

Die Vergleichswerte werden mit dem aktuellen Messwert verglichen. Wenn der Messwert die parametrierte Vergleichsbedingung erfüllt und die technologische Funktion des zugehörigen Digitalausgangs freigegeben ist, wird der Digitalausgang gesetzt. Wenn Sie für den Digitalausgang DQ1 "Zwischen Vergleichswert 0 und 1" oder "Nicht zwischen Vergleichswert 0 und 1" parametrieren, wirken sich beide Vergleichswerte auf DQ1 aus.

Sie können das Schalten eines Digitalausgangs von einer der folgenden Vergleichsereignisse abhängig machen:

Setzen oberhalb des Vergleichswerts

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

Messwert >= Vergleichswert

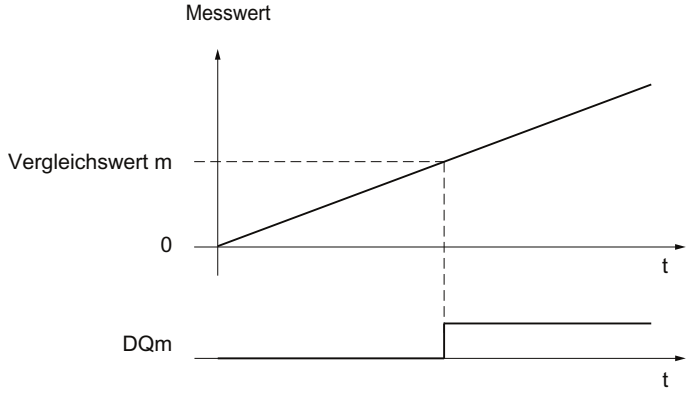

Setzen unterhalb des Vergleichswerts

Der jeweilige Digitalausgang wird auf 1 gesetzt, wenn:

Messwert <= Vergleichswert

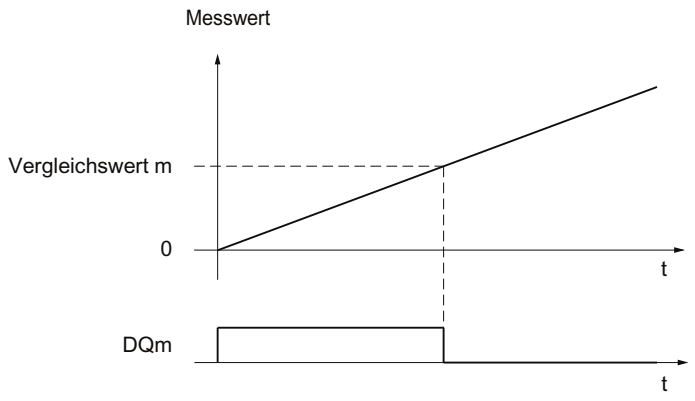

Setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das Vergleichsereignis ist für den Digitalausgang DQ1 parametrierbar, wenn für den Digitalausgang DQ0 "Nutzung durch Anwenderprogramm" parametriert ist. Vergleichswert 1 muss größer sein als Vergleichswert 0.

DQ1 wird auf 1 gesetzt, wenn:

Vergleichswert 0 <= Messwert <= Vergleichswert 1

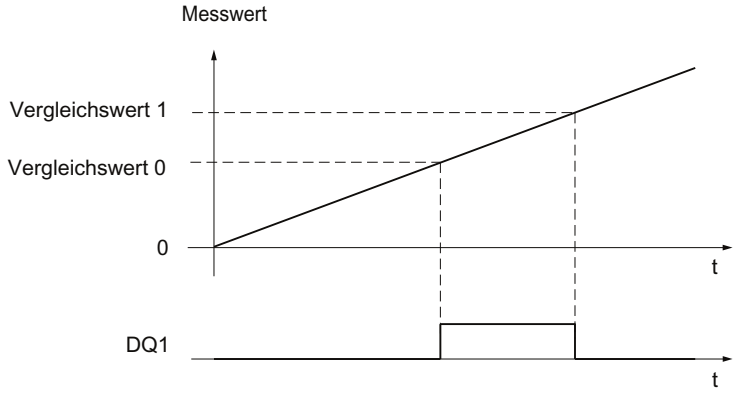

Nicht setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das Vergleichsereignis ist für den Digitalausgang DQ1 parametrierbar, wenn für den Digitalausgang DQ0 "Nutzung durch Anwenderprogramm" parametriert ist. Vergleichswert 1 muss größer sein als Vergleichswert 0.

DQ1 wird auf 1 gesetzt, wenn:

Vergleichswert 1 <= Messwert <= Vergleichswert 0

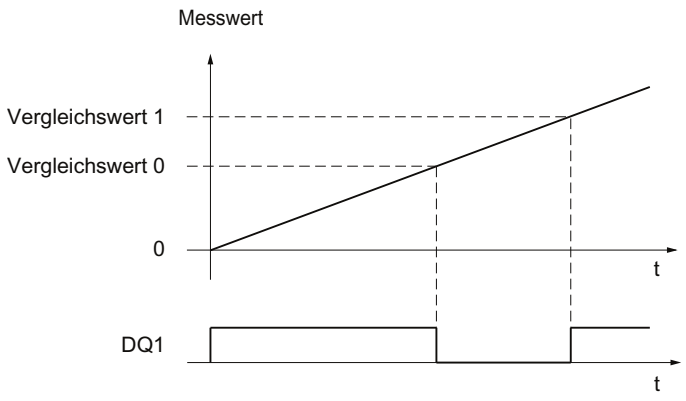

Messwertermittlung

Übersicht der Messfunktionen

Folgende hochgenaue Messfunktionen stehen zur Verfügung (Genauigkeit bis zu 100 ppm):

Messart (Seite 52)	Beschreibung
Frequenzmessung	Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen wird in einem Messintervall die mittlere Frequenz ermittelt und als Gleitkommazahl in der Einheit Hertz zurückgeliefert.
Periodendauermessung	Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen wird in einem Messintervall die mittlere Periodendauer ermittelt und als Gleitkommazahl in der Einheit Sekunden zurückgeliefert.
Geschwindigkeitsmessung	Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen und weiteren Parametern wird in einem Messintervall die mittlere Geschwindigkeit ermittelt und in der parametrierten Einheit zurückgeliefert.

Messwerte und Zählwerte stehen in der Rückmeldeschnittstelle parallel zur Verfügung.

Aktualisierungszeit

Sie können den zeitlichen Abstand, mit dem das Technologiemodul die Messwerte zyklisch aktualisiert, als Aktualisierungszeit parametrieren. Durch größere Aktualisierungszeiten können unruhige Messgrößen geglättet und die Messgenauigkeit erhöht werden.

Torsteuerung bei Inkremental- und Impulsgeber

Das Öffnen und Schließen des internen Tors definiert das Zeitfenster, in dem die Zählimpulse erfasst werden. Die Aktualisierungszeit ist asynchron zum Öffnen des Tors, d.h. die Aktualisierungszeit wird nicht mit dem Öffnen des Tors gestartet. Nach Schließen des internen Tors wird der zuletzt ermittelte Messwert weiter zurückgeliefert.

Messwertermittlung mit Inkremental- oder Impulsgeber

Messbereiche (TM Count und TM PosInput)

Die Messfunktionen haben folgende Messbereichsgrenzen:

Messart	Untere Messbereichsgrenze	Obere Messbereichsgrenze
Frequenzmessung	0,04 Hz	800 kHz* / 4 MHz**
Periodendauermessung	1,25 µs* / 0,25 µs**	25 s
Geschwindigkeitsmessung	Abhängig von der parametrierten Anzahl der "Inkremente pro Einheit" und der "Zeitbasis für Geschwindigkeitsmessung"	

* Gültig für 24 V-Inkrementalgeber und Signalauswertung "vierfach".

** Gültig für RS422-Inkrementalgeber und Signalauswertung "vierfach".

Alle Messwerte werden als vorzeichenbehafteter Wert zurückgeliefert. Das Vorzeichen gibt an, ob der Zählwert im relevanten Zeitintervall gestiegen oder gefallen ist.

Messbereiche (Kompakt-CPU)

Die Messfunktionen haben folgende Messbereichsgrenzen:

Messart	Untere Messbereichsgrenze	Obere Messbereichsgrenze
Frequenzmessung	0,04 Hz	400 kHz*
Periodendauermessung	2,5 µs*	25 s
Geschwindigkeitsmessung	Abhängig von der parametrierten Anzahl der "Inkremente pro Einheit" und der "Zeitbasis für Geschwindigkeitsmessung"	

* Gültig für 24 V-Inkrementalgeber und Signalauswertung "vierfach".

Alle Messwerte werden als vorzeichenbehafteter Wert zurückgeliefert. Das Vorzeichen gibt an, ob der Zählwert im relevanten Zeitintervall gestiegen oder gefallen ist.

Messprinzip

Das Technologiemodul ordnet jedem Zählimpuls einen Zeitwert zu. Das Messintervall ist definiert als der zeitliche Abstand zwischen dem jeweils letzten Zählimpuls vor und während der vorangegangenen Aktualisierungszeit. Zur Berechnung einer Messgröße werden das Messintervall und die Anzahl der Impulse im Messintervall ausgewertet.

Wenn innerhalb einer Aktualisierungszeit kein Zählimpuls auftritt, erfolgt eine dynamische Anpassung des Messintervalls. In diesem Fall wird ein Impuls zum Ende der Aktualisierungszeit angenommen und das Messintervall von dort bis zum letzten aufgetretenen Impuls berechnet. Die Anzahl der Impulse ist dann 1.

Das Rückmeldebit STS_M_INTERVAL meldet, ob im vorangegangenen Messintervall ein Zählimpuls aufgetreten ist. Dadurch lässt sich zwischen einem angenommenen und einem tatsächlichen Zählimpuls unterscheiden.

Die folgenden Bilder zeigen das Messprinzip und die dynamische Anpassung des Messintervalls:

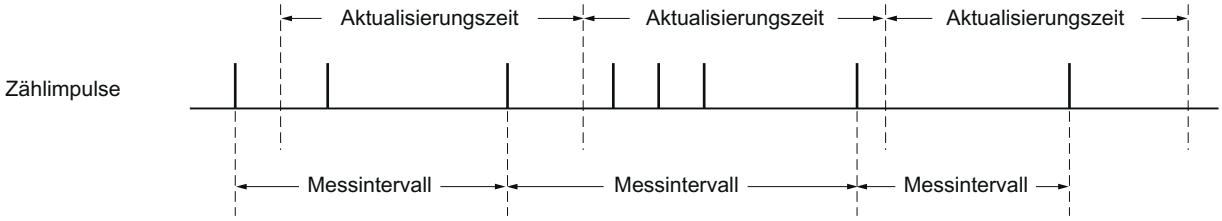

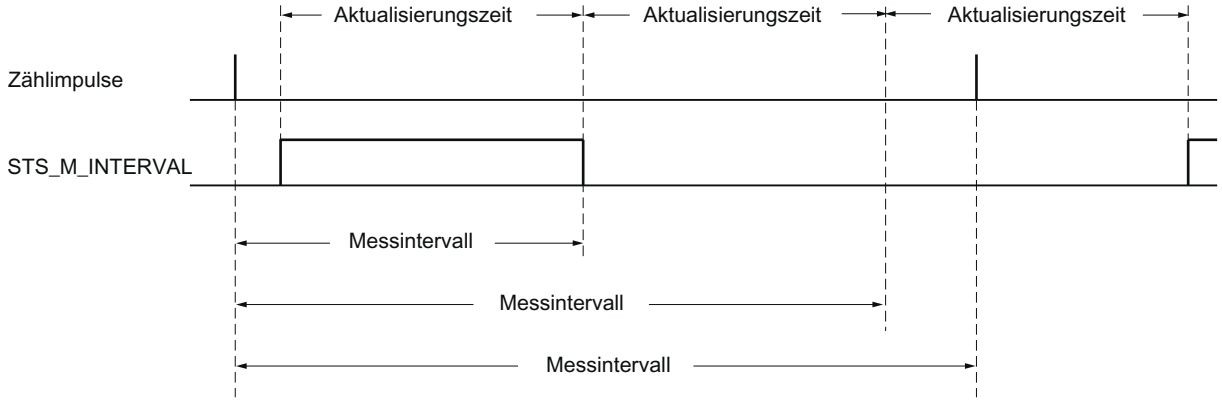

Frequenzmessung

In der Zeit bis zum ersten vorhandenen Messwert wird der Wert "0" zurückgemeldet.

Der Messvorgang beginnt mit dem ersten erfassten Impuls nach dem Öffnen des internen Tors. Der erste Messwert wird frühestens nach dem zweiten Impuls berechnet.

Nach jedem Ablauf der Aktualisierungszeit wird in der Rückmeldeschnittstelle (Seite 172) der Messwert aktualisiert. Wenn das interne Tor geschlossen ist, ist der Messvorgang gestoppt und der Messwert wird nicht mehr aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Messung der Frequenz bei einer Aktualisierungszeit von 1 s:

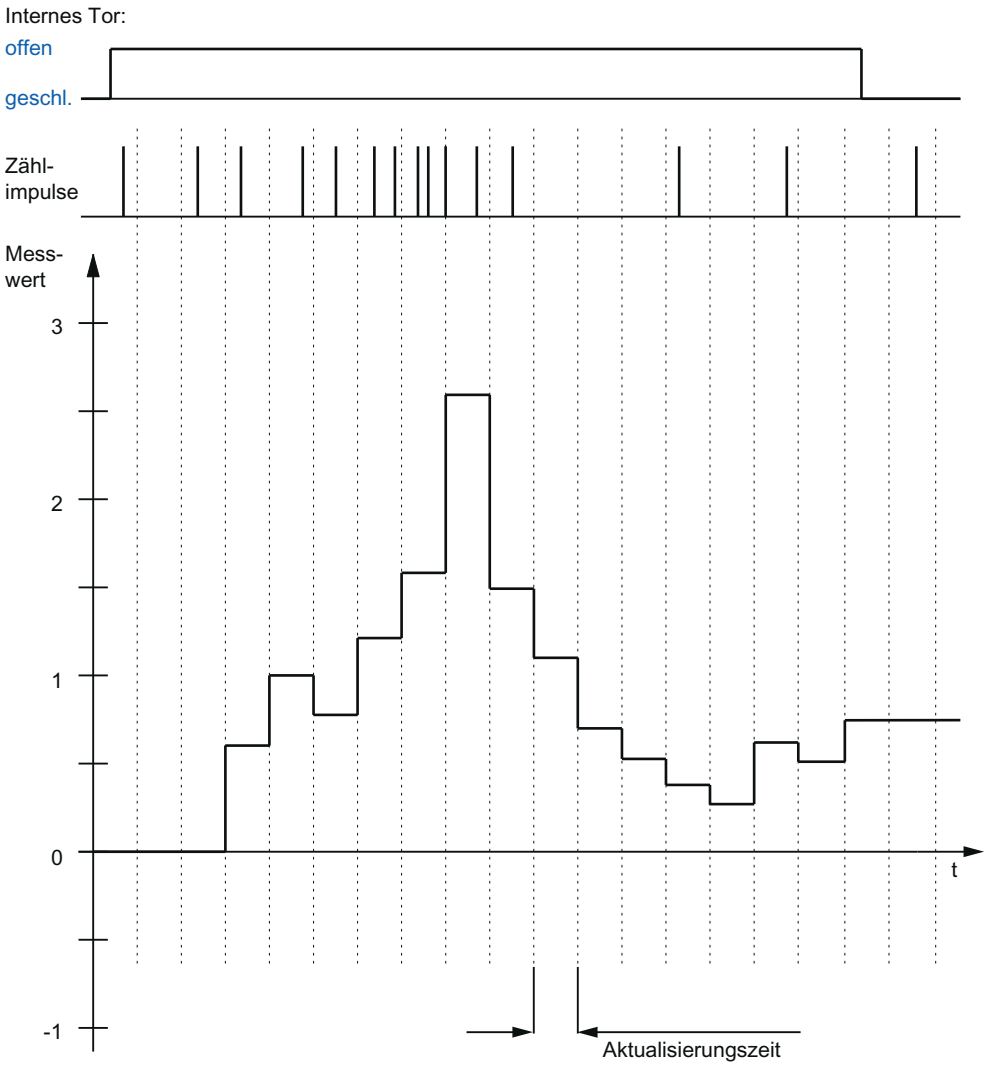

Periodendauermessung

Bei der Periodendauermessung wird als Messwert der Kehrwert der Frequenz geliefert.

In der Zeit bis zum ersten vorhandenen Messwert wird der Wert "25 s" zurückgemeldet.

Geschwindigkeitsmessung

Bei der Geschwindigkeitsmessung wird als Messwert die normierte Frequenz geliefert. Die Normierung parametrieren Sie über die Zeitbasis und die Anzahl der Inkremente, die Ihr Geber pro Einheit liefert.

Beispiel:

Ihr Geber liefert 4000 Inkremente pro Meter. Die Geschwindigkeit soll in Metern pro Minute gemessen werden.

Sie müssen in diesem Fall 4000 Inkremente pro Einheit und eine Zeitbasis von einer Minute parametrieren.

Messwertermittlung mit SSI-Absolutwertgeber

Messbereiche für SSI-Absolutwertgeber

Die Messfunktionen haben folgende Messbereichsgrenzen:

Messart	Untere Messbereichsgrenze	Obere Messbereichsgrenze
Frequenzmessung	0,04 Hz	4 MHz
Periodendauermessung	0,25 µs	25 s
Geschwindigkeitsmessung	Abhängig von der parametrierten Anzahl der "Inkremente pro Einheit" und der "Zeitbasis für Geschwindigkeitsmessung"	

Alle Messwerte werden als vorzeichenbehafteter Wert zurückgeliefert. Das Vorzeichen gibt an, ob der Positionswert im relevanten Zeitintervall gestiegen oder gefallen ist.

Messprinzip

Das Technologiemodul ordnet jedem SSI-Telegramm einen Zeitwert zu. Das Messintervall ist definiert als der zeitliche Abstand zwischen dem jeweils letzten SSI-Telegramm mit einer Positionswertänderung vor und während der vorangegangenen Aktualisierungszeit. Zur Berechnung einer Messgröße werden das Messintervall und die gesamte Positionswertänderung im Messintervall ausgewertet. Die gesamte Positionswertänderung in einem Messintervall entspricht der Anzahl der Geberinkremente im selben Messintervall.

Wenn innerhalb einer Aktualisierungszeit keine Positionswertänderung auftritt, erfolgt eine dynamische Anpassung des Messintervalls. In diesem Fall wird eine Positionswertänderung zum Ende der Aktualisierungszeit angenommen und das Messintervall von dort bis zum letzten SSI-Telegramm mit einer Positionswertänderung berechnet. Die Positionswertänderung ist dann 1.

Das Rückmeldebit STS_M_INTERVAL meldet, ob im vorangegangenen Messintervall eine Positionswertänderung aufgetreten ist. Dadurch lässt sich zwischen einer angenommenen und einer tatsächlichen Positionswertänderung unterscheiden. Wenn das Technologiemodul wegen einer Überschreitung der Messbereichsgrenzen keinen Messwert berechnen kann, wird das Rückmeldebit STS_M_INTERVAL nicht gesetzt.

Frequenzmessung

In der Zeit bis zum ersten vorhandenen Messwert wird der Wert "0,0" zurückgemeldet.

Der Messvorgang beginnt mit der ersten erfassten Positionswertänderung. Der erste Messwert wird frühestens nach der zweiten erfassten Positionswertänderung berechnet.

Nach jedem Ablauf der Aktualisierungszeit wird in der Rückmeldeschnittstelle (Seite 172) der Messwert aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Messung der Frequenz bei einer Aktualisierungszeit von 1 s:

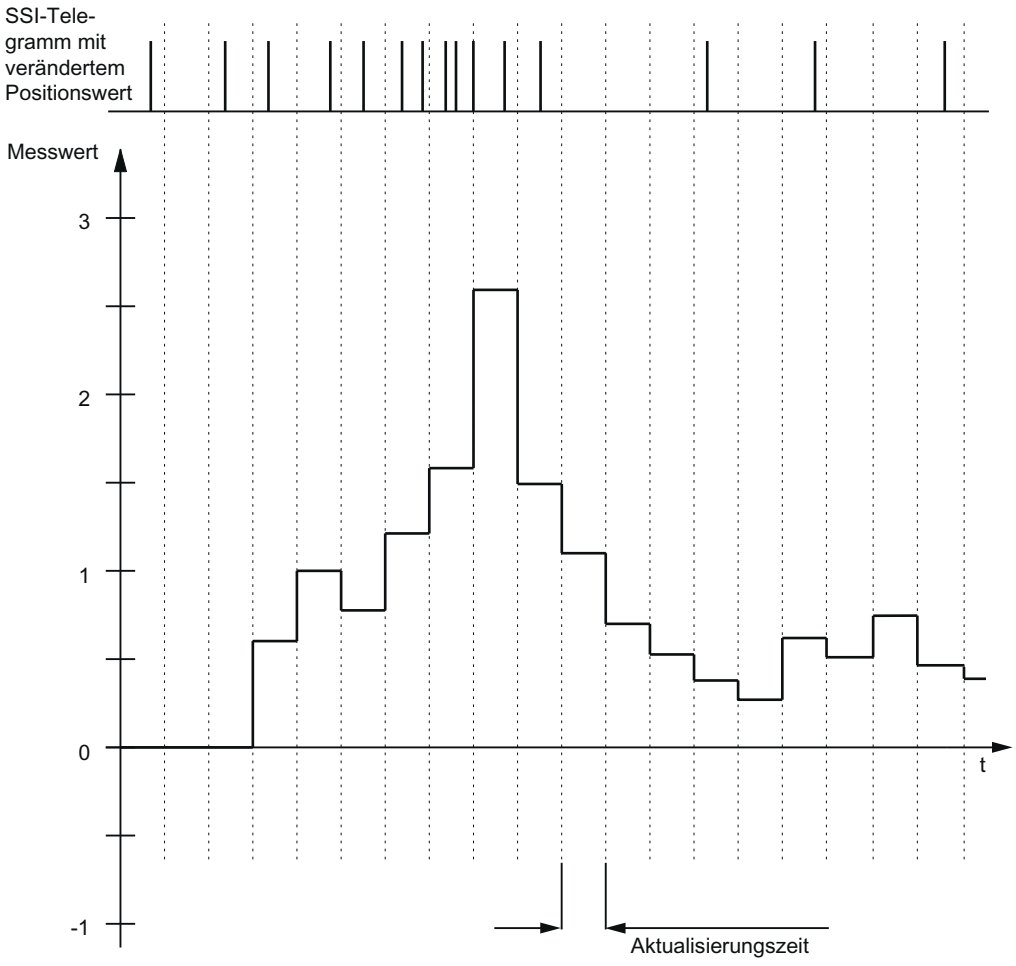

Periodendauermessung

Bei der Periodendauermessung wird als Messwert der Kehrwert der Frequenz geliefert.

In der Zeit bis zum ersten vorhandenen Messwert wird der Wert "25 s" zurückgemeldet.

Geschwindigkeitsmessung

Bei der Geschwindigkeitsmessung wird als Messwert die normierte Frequenz geliefert. Die Normierung parametrieren Sie über die Zeitbasis und die Anzahl der Inkremente, die Ihr Geber pro Einheit liefert.

Beispiel:

Ihr SSI-Absolutwertgeber arbeitet mit einer Auflösung von 12 Bit pro Umdrehung und liefert 4096 Inkremente pro Umdrehung. Die Geschwindigkeit soll in Umdrehungen pro Minute gemessen werden.

Sie müssen in diesem Fall 4096 Inkremente pro Einheit und eine Zeitbasis von einer Minute parametrieren.

Hinweis

Zu hohe Geberdrehzahl kann falsche Drehrichtung liefern

Wenn sich ein SSI-Absolutwertgeber so schnell dreht, dass innerhalb eines Baugruppenzyklus[1] mehr als der halbe Wertebereich zurückgelegt wird, können Geschwindigkeit und Drehrichtung nicht mehr korrekt ermittelt werden. Dadurch kann es zu einer inkorrekten Funktion kommen bei:

- DQ-Funktionen
- Rückmeldebits EVENT_OFLW, EVENT_UFLW, EVENT_ZERO, EVENT_CMP0, EVENT_CMP1 und STS_DIR

[1] Nicht-taktsynchroner Betrieb: 500 µs; taktsynchroner Betrieb: PROFINET-Zykluszeit

Hinweis

Wenn Sie einen SSI-Absolutwertgeber verwenden, dessen Wertebereich nicht einer 2er-Potenz entspricht, kann der berechnete Geschwindigkeitsmesswert im Moment des Überlaufs fehlerhaft sein.

Hysterese

Hysterese mit Inkremental- oder Impulsgeber

Beschreibung

Sie können mit der Hysterese einen Bereich um die Vergleichswerte festlegen, in welchem die Digitalausgänge nicht erneut schalten sollen, bevor der Zählwert diesen Bereich einmal verlassen hat.

Durch geringfügige Bewegungen des Gebers kann der Zählwert um einen bestimmten Wert herum schwanken. Liegt im Schwankungsbereich ein Vergleichswert oder eine Zählgrenze, wird ohne Verwendung einer Hysterese der zugehörige Digitalausgang entsprechend oft ein- und ausgeschaltet. Die Hysterese verhindert diese ungewollten Schaltvorgänge sowie parametrierte Prozessalarme bei Eintritt eines Vergleichsereignisses.

Die Hysterese wird aktiv, wenn der jeweilige Vergleichswert durch einen Zählimpuls erreicht wird. Wenn der Zählwert während einer aktiven Hysterese auf den Startwert gesetzt wird, wird die Hysterese inaktiv.

Der Hysteresebereich endet, unabhängig vom Hysteresewert, an der unteren bzw. oberen Zählgrenze.

Funktionsweise

Das folgende Bild zeigt ein Beispiel für die Hysterese mit folgender Parametrierung:

- Setzen eines Digitalausgangs zwischen Vergleichswert und oberer Zählgrenze
- Vergleichswert = 5
- Hysterese = 0 bzw. 2 (grau hinterlegt)

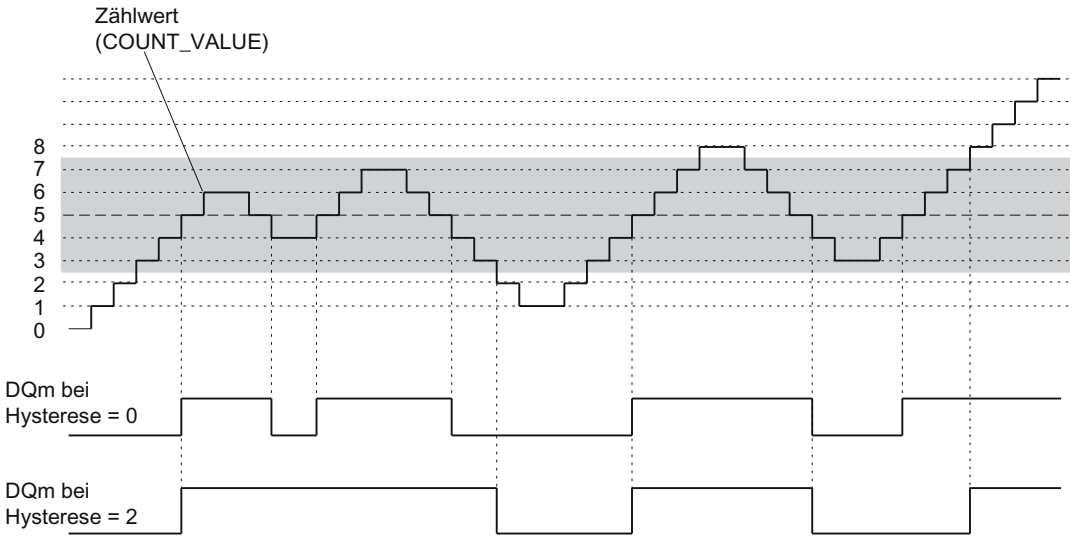

Bei Erreichen des Zählwerts 5 wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert. Bei Erreichen der Zählwerte 2 oder 8 wird die Hysterese inaktiv.

Das folgende Bild zeigt ein Beispiel für die Hysterese mit folgender Parametrierung:

- Setzen bei Vergleichswert für eine Impulsdauer
- Vergleichswert = 5
- Vergleich in beiden Zählrichtungen
- Hysterese = 0 bzw. 2 (grau hinterlegt)

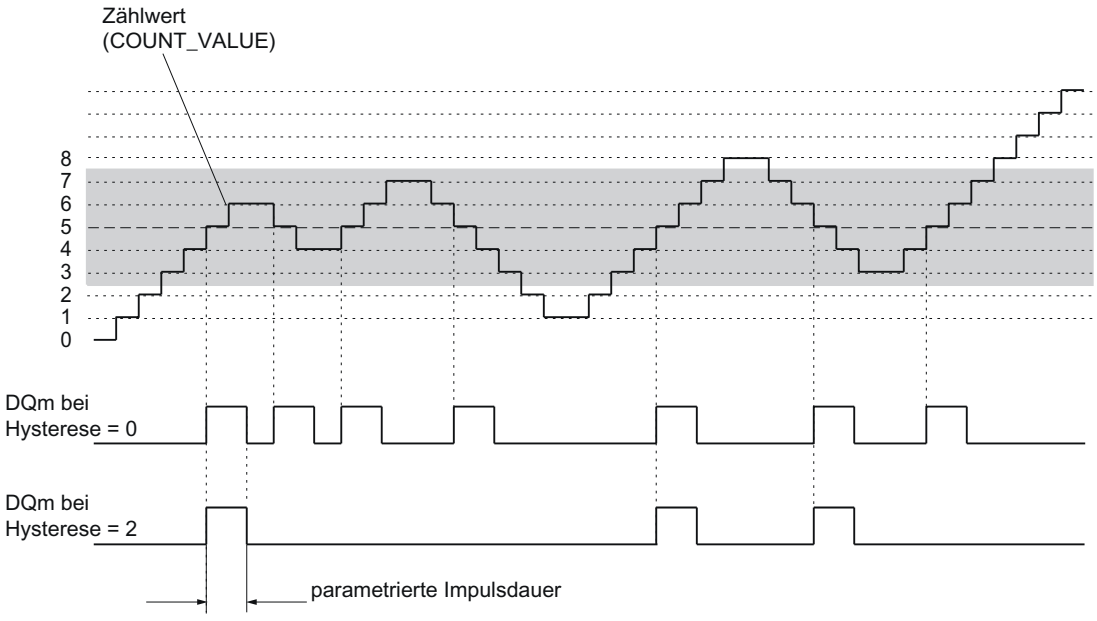

Hysterese mit SSI-Absolutwertgeber

Beschreibung

Sie können mit der Hysterese einen Bereich um die Vergleichswerte festlegen, in welchem die Digitalausgänge nicht erneut schalten sollen, bevor der Positionswert diesen Bereich einmal verlassen hat.

Durch geringfügige Bewegungen des Gebers kann der Positionswert um einen bestimmten Wert herum schwanken. Liegt im Schwankungsbereich ein Vergleichswert, der jeweilige minimale oder maximale Positionswert, wird ohne Verwendung einer Hysterese der zugehörige Digitalausgang entsprechend oft ein- und ausgeschaltet. Die Hysterese verhindert diese ungewollten Schaltvorgänge sowie parametrierte Prozessalarme bei Eintritt eines Vergleichsereignisses.

Der Hysteresebereich endet, unabhängig vom Hysteresewert, beim jeweiligen minimalen bzw. maximalen Positionswert.

Funktionsweise

Das folgende Bild zeigt ein Beispiel für die Hysterese mit folgender Parametrierung:

- Setzen eines Digitalausgangs zwischen Vergleichswert und oberer Grenze
- Vergleichswert = 10
- Hysterese = 0 bzw. 2 (grau hinterlegt)

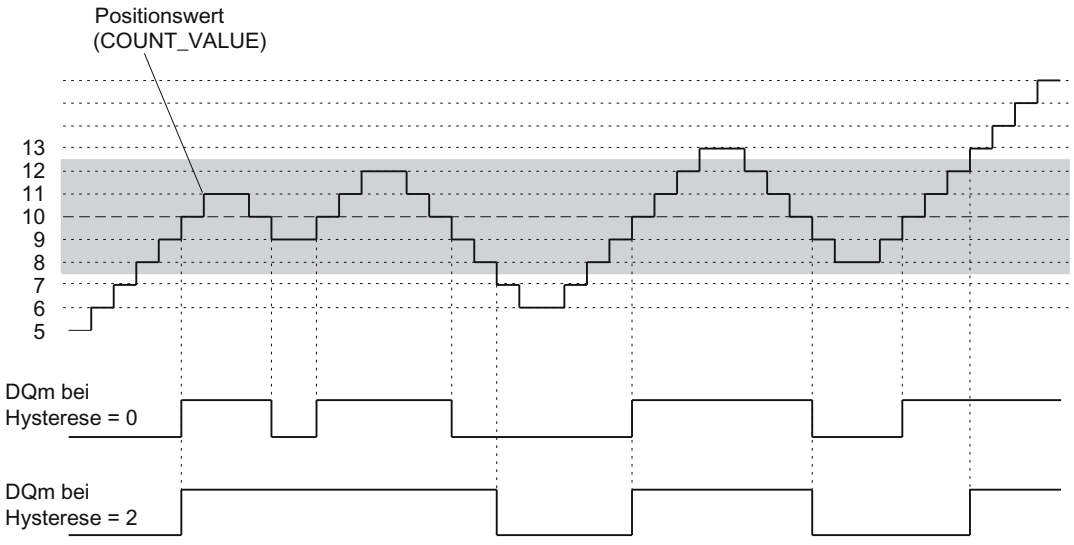

Bei Erreichen des Positionswerts 10 wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert. Bei Erreichen der Positionswerte 7 oder 13 wird die Hysterese inaktiv.

Das folgende Bild zeigt ein Beispiel für die Hysterese mit folgender Parametrierung:

- Setzen bei Vergleichswert für eine Impulsdauer
- Vergleichswert = 10
- Vergleich in beiden Richtungen der Positionswertänderungen
- Hysterese = 0 bzw. 2 (grau hinterlegt)

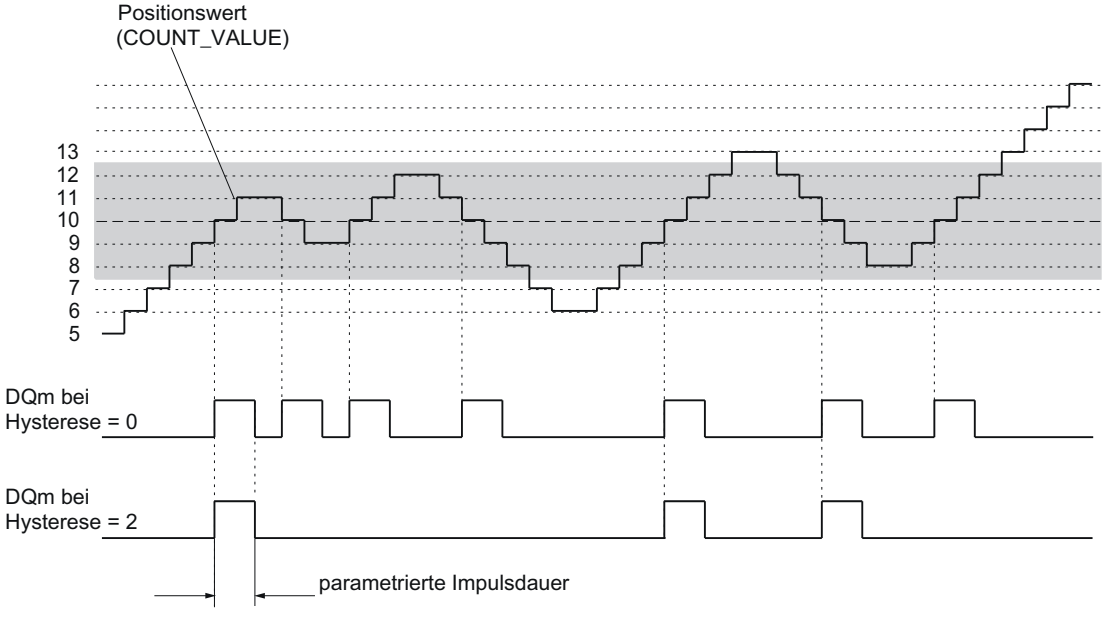

Alarme

Prozessalarm

Das Technologiemodul kann unter anderem bei Eintritt eines Vergleichsereignisses, bei Überlauf, bei Unterlauf, bei Nulldurchgang des Zählers und/oder Wechsel der Zählrichtung (Richtungsumkehr) einen Prozessalarm in der CPU auslösen. Sie können festlegen, welche Ereignisse im Betrieb einen Prozessalarm auslösen sollen.

Diagnosealarm

Das Technologiemodul kann bei Fehlern Diagnosealarme auslösen. Sie geben die Diagnosealarme für bestimmte Fehler in der Gerätekonfiguration frei. Informieren Sie sich im Gerätehandbuch zum Technologiemodul, welche Ereignisse im Betrieb einen Diagnosealarm auslösen können.

Positionserfassung für Motion Control

Beschreibung

Sie können das Technologiemodul z. B. mit einem Inkrementalgeber für die folgenden Achs-Technologieobjekte von S7-1500 Motion Control zur Positionserfassung nutzen:

- TO_PositioningAxis
- TO_SynchronousAxis
- TO_ExternalEncoder

Bei Verwendung eines Inkrementalgebers oder Impulsgebers beruht die Positionserfassung auf der Zählfunktion des Technologiemoduls. Bei einem SSI-Absolutwertgeber wird der Absolutwert über eine synchrone, serielle Schnittstelle eingelesen und entsprechend der Parametrierung aufbereitet und für S7-1500 Motion Control bereitgestellt.

Der Funktionsumfang des Technologiemoduls hat dabei folgende Einschränkungen:

- Keine Zählerverhalten parametrierbar
- Keine Funktionen für Digitaleingänge, außer Messtasterfunktion, verfügbar
- Keine Vergleichsfunktionen für Digitalausgänge verfügbar
- Keine Prozessalarme verfügbar

In der Gerätekonfiguration des Technologiemoduls in STEP 7 (TIA Portal) wählen Sie hierfür den Betriebsmodus "Positionserfassung für Technologieobjekt "Motion Control"" und nutzen im Programm das entsprechende Technologieobjekt. Dadurch werden die Parametriermöglichkeiten auf die notwendigen Parameter reduziert. Bei einem TM Count oder TM PosInput gilt der Betriebsmodus automatisch für alle Kanäle des Technologiemoduls. Bei einer Kompakt-CPU gilt der Betriebsmodus für den jeweiligen Kanal.

In diesem Betriebsmodus können Sie das Technologieobjekt Messtaster (TO_MeasuringInput) dazu verwenden, mit einem Hardware-Digitaleingang eine Messtasterfunktion auszuführen. Wählen Sie dazu im Technologieobjekt Messtaster den

Messtastertyp "Messen über PROFIdrive-Telegramm" und als Nummer des Messtasters den Wert "1".

Informationen über die weitere Projektierung finden Sie in der Hilfe zu den Achs-Technologieobjekten und dem Technologieobjekt Messtaster von S7-1500 Motion Control.

Gebersignale

24 V- und TTL-Zählsignale

Zählsignale von 24 V- und TTL-Inkrementalgebern

Der 24 V-Inkrementalgeber liefert die 24 V-Signale A, B und N an das Technologiemodul. Die Signale A und B sind zueinander um 90° phasenversetzt. Sie können auch Inkrementalgeber ohne Signal N anschließen.

Bei einem 24 V-Inkrementalgeber werden zum Zählen die Signale A und B verwendet. Das Signal N dient bei entsprechender Parametrierung zum Setzen des Zählers auf den Startwert oder zum Speichern des aktuellen Zählwerts in den Capture-Wert.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines 24 V-Inkrementalgebers:

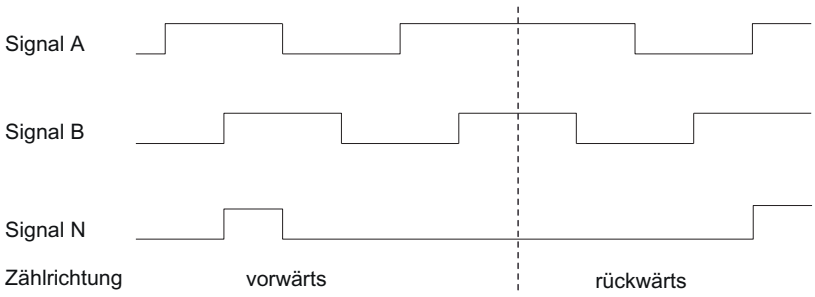

Das Technologiemodul erkennt die Zählrichtung an der Abfolge der Flanken an den Signalen A und B. Sie können eine Invertierung der Zählrichtung festlegen.

Zählsignale von 24 V- und TTL-Impulsgebern ohne/mit Richtungssignal

Der Geber, zum Beispiel ein Initiator (BERO) oder eine Lichtschranke, liefert nur ein Zählsignal, das an den Anschluss A des Zählers angeschlossen wird.

Zusätzlich können Sie ein Signal zur Richtungserkennung an den Anschluss B des Zählers anschließen. Bei einem High-Pegel des Richtungssignals wird rückwärts gezählt. Liefert Ihr Geber kein entsprechendes Signal, können Sie die Zählrichtung mit dem Anwenderprogramm über die Steuerschnittstelle vorgeben.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines 24 V-Impulsgebers mit Richtungssignal und die daraus resultierenden Zählimpulse:

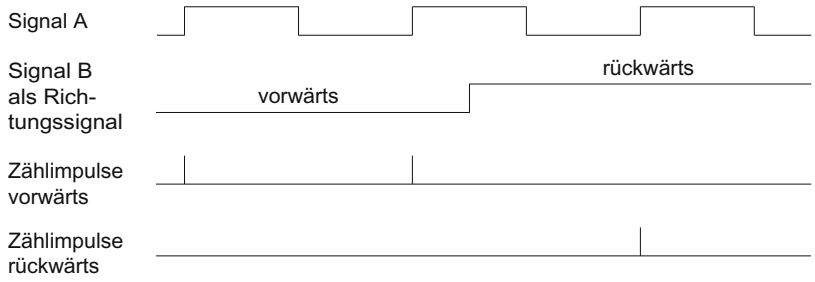

Zählsignale von 24 V- und TTL-Impulsgebern mit Zählsignal vorwärts/rückwärts

Das Zählsignal vorwärts wird an den Anschluss A angeschlossen. Das Zählsignal rückwärts wird an den Anschluss B angeschlossen.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines Impulsgebers mit Zählsignal vorwärts/rückwärts und die daraus resultierenden Zählimpulse:

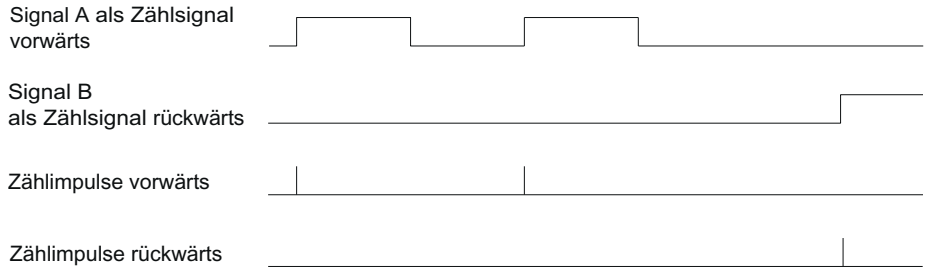

P-Schalter/M-Schalter für 24 V-Zählsignale (TM Count)

An den Zähleingängen können Sie folgende Geber bzw. Sensoren anschließen:

- P-Schalter:
  Die Eingänge A, B und N werden nach 24VDC geschaltet.
- M-Schalter:
  Die Eingänge A, B und N werden nach Masse M geschaltet.
- Gegentakt-schaltend (M- und P-Schalter):
  Die Eingänge A, B und N werden wechselweise nach 24VDC und Masse M geschaltet.

P-Schalter für 24 V-Zählsignale (Kompakt-CPU)

An den Zähleingängen können Sie P-schaltende und Gegentakt-schaltende Geber bzw. Sensoren anschließen.

Überwachung der Gebersignale (TM Count und TM PosInput)

Die Signale von Gegentakt-schaltenden 24 V-Gebern werden vom Technologiemodul auf Drahtbruch überwacht. TTL-Signale werden vom Technologiemodul auf fehlerhafte Versorgungsspannung überwacht.

Wenn Sie in der Gerätekonfiguration den Diagnosealarm freigeben, löst das Technologiemodul bei einem Fehler an den Gebersignalen einen Diagnosealarm aus.

RS422-Zählsignale

Zählsignale von RS422-Inkrementalgebern

Der RS422-Inkrementalgeber liefert folgende Differenzsignale an das Technologiemodul:

- +A und -A
- +B und -B
- +N und -N

Die Signalinformation bei RS422-Signalen ist in deren Differenzspannung jeweils zwischen +A und -A, +B und -B bzw. +N und -N kodiert. Die Signale A und B sind zueinander um 90° phasenversetzt. Sie können auch Inkrementalgeber ohne Signal N anschließen.

Bei einem RS422-Inkrementalgeber werden zum Zählen die Signale A und B benutzt. Das Signal N dient bei entsprechender Parametrierung zum Setzen des Zählers auf den Startwert oder zum Speichern des aktuellen Zählwerts als Capture-Wert.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines RS422-Inkrementalgebers:

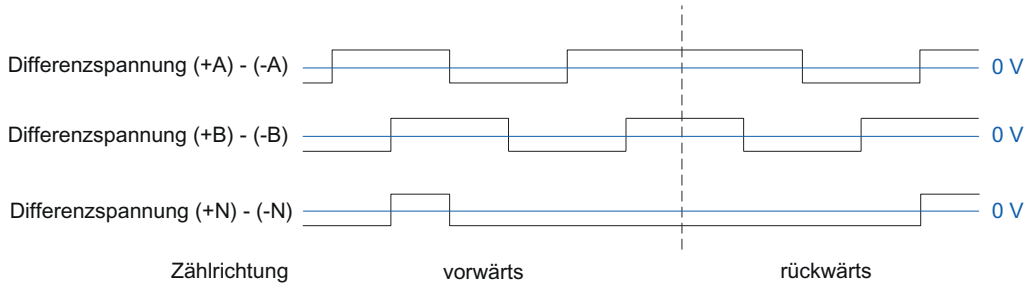

Das Technologiemodul erkennt die Zählrichtung an der Abfolge der Flanken an den Signalen A und B. Sie können eine Invertierung der Zählrichtung festlegen.

Zählsignale von RS422-Impulsgebern ohne/mit Richtungssignal

Der Geber, zum Beispiel eine Lichtschranke, liefert nur ein Zählsignal, das an den Anschluss A angeschlossen wird.

Zusätzlich können Sie ein Signal zur Richtungserkennung an den Anschluss B anschließen. Bei einem High-Pegel des Richtungssignals wird rückwärts gezählt. Liefert Ihr Geber kein entsprechendes Signal, können Sie die Zählrichtung mit dem Anwenderprogramm über die Steuerschnittstelle vorgeben.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines RS422-Impulsgebers mit Richtungssignal und die daraus resultierenden Zählimpulse:

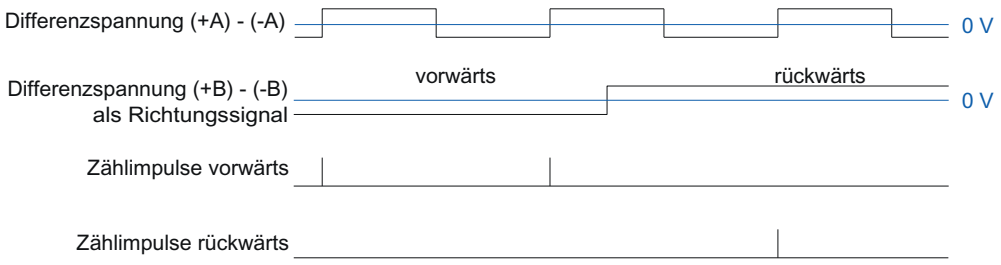

Zählsignale von RS422-Impulsgebern mit Zählsignal vorwärts/rückwärts

Das Zählsignal vorwärts wird an den A-Anschlüssen angeschlossen. Das Zählsignal rückwärts wird an den B-Anschlüssen angeschlossen.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines RS422-Impulsgebers mit Zählsignal vorwärts/rückwärts und die daraus resultierenden Zählimpulse:

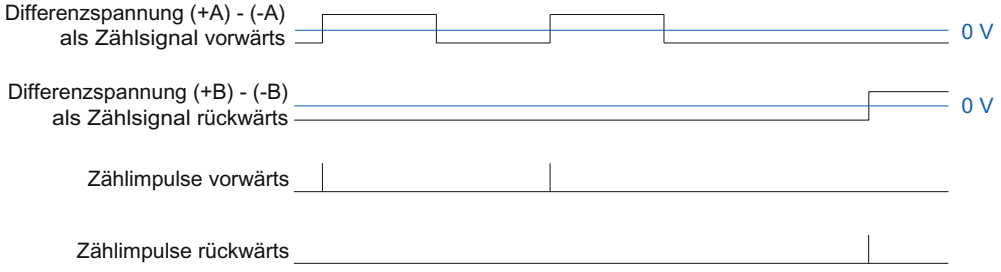

Überwachung der Gebersignale

RS422-Signale werden vom Technologiemodul auf Drahtbruch, Kurzschluss und fehlerhafte Versorgungsspannung überwacht.

Wenn Sie in der Gerätekonfiguration den Diagnosealarm freigeben, löst das Technologiemodul bei einem Fehler an den Gebersignalen einen Diagnosealarm aus.

SSI-Signale

Signale von SSI-Absolutwertgebern

Der SSI-Absolutwertgeber und das Technologiemodul kommunizieren über die SSI-Datensignale +D und -D und die SSI-Taktsignale +C und -C. SSI verwendet den RS422-Signalstandard. Die Signalinformation ist in der jeweiligen Differenzspannung zwischen +C und -C sowie +D und -D kodiert.

Überwachung der Gebersignale und der SSI-Telegramme

Die Signale eines SSI-Absolutwertgebers werden vom Technologiemodul auf Drahtbruch, Kurzschluss und fehlerhafte Versorgungsspannung überwacht. Außerdem überwacht das Technologiemodul SSI-Telegramme auf Fehler.

Wenn Sie in der Gerätekonfiguration die Diagnosealarme freigeben, löst das Technologiemodul bei einem Fehler an den Gebersignalen oder des SSI-Telegramms einen Diagnosealarm aus.

Signalauswertung von Inkrementalsignalen

Übersicht

Der Zähler des Technologiemoduls zählt die Flanken der Gebersignale A und B. Bei Inkrementalgebern mit zueinander phasenversetzten Signalen A und B können Sie wählen zwischen der Einfach- und der Mehrfachauswertung, um die Auflösung zu erhöhen.

Sie können folgende Signalauswertungen parametrieren:

- Einfachauswertung (Seite 66)
- Zweifachauswertung (Seite 67)
- Vierfachauswertung (Seite 67)

Hinweis

Der Phasenversatz zwischen den Flanken der Signale A und B wird ausgewertet. Wenn kein Phasenversatz erkennbar ist, wird ein Geberfehler (Illegaler Übergang der A/B-Signale) über das Rückmelde-Bit ENC_ERROR gemeldet.

Einfachauswertung

Bei der Einfachauswertung werden die steigende und fallende Flanke an Signal A während eines Low-Pegels an Signal B ausgewertet.

Zählimpulse in Vorwärtsrichtung werden bei steigender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt. Zählimpulse in Rückwärtsrichtung werden bei fallender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt.

Das folgende Bild zeigt ein Beispiel für die Einfachauswertung von 24 V- und TTL-Zählsignalen:

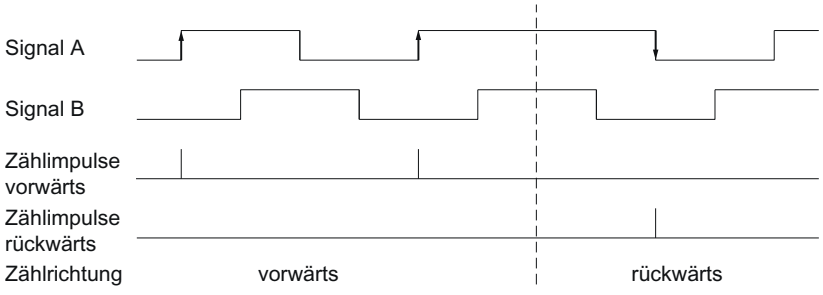

Das folgende Bild zeigt ein Beispiel für die Einfachauswertung von RS422-Zählsignalen:

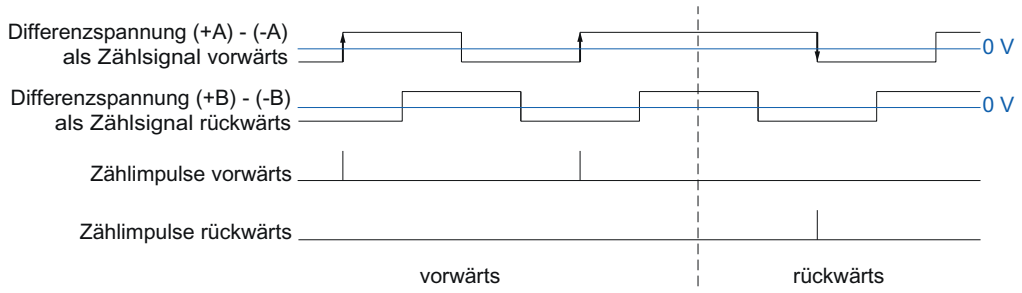

Zweifachauswertung

Bei der Zweifachauswertung werden die steigende und die fallende Flanke an Signal A ausgewertet.

Ob Zählimpulse in Vorwärts- oder Rückwärtsrichtung erzeugt werden, hängt von der Flankenrichtung des Signals A und vom Pegel an Signal B währenddessen ab.

Das folgende Bild zeigt ein Beispiel für die Zweifachauswertung von 24 V- und TTL-Zählsignalen:

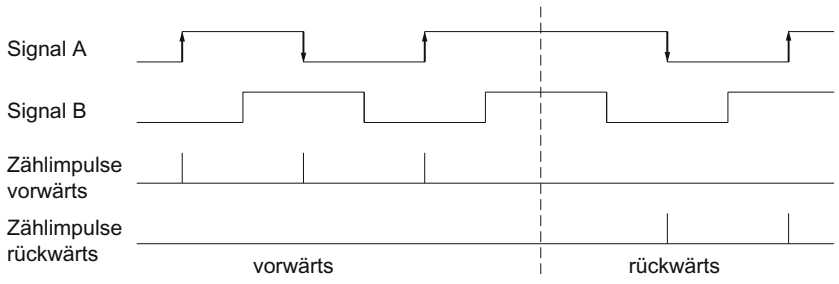

Das folgende Bild zeigt ein Beispiel für die Zweifachauswertung von RS422-Zählsignalen:

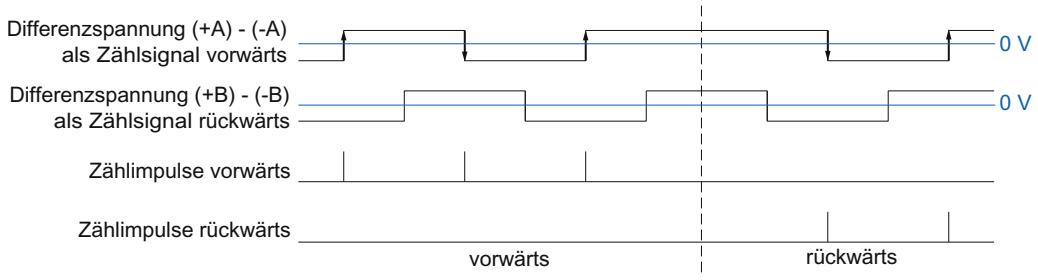

Vierfachauswertung

Bei der Vierfachauswertung werden die steigenden und fallenden Flanken an Signal A und Signal B ausgewertet.

Ob Zählimpulse in Vorwärts- oder Rückwärtsrichtung erzeugt werden, hängt von der Flankenrichtung des einen Signals und dem Pegel des jeweils anderen Signals währenddessen ab.

Das folgende Bild zeigt ein Beispiel für die Vierfachauswertung von 24 V- und TTL-Zählsignalen:

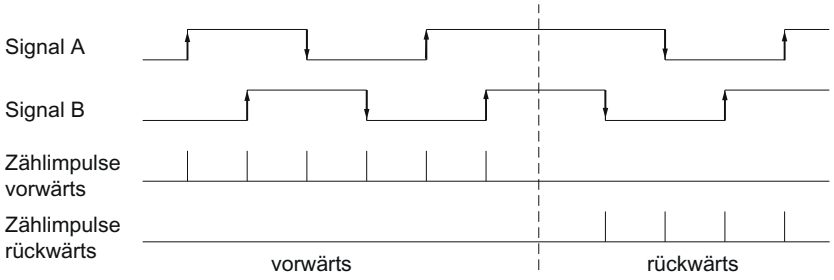

Das folgende Bild zeigt ein Beispiel für die Vierfachauswertung von RS422-Zählsignalen:

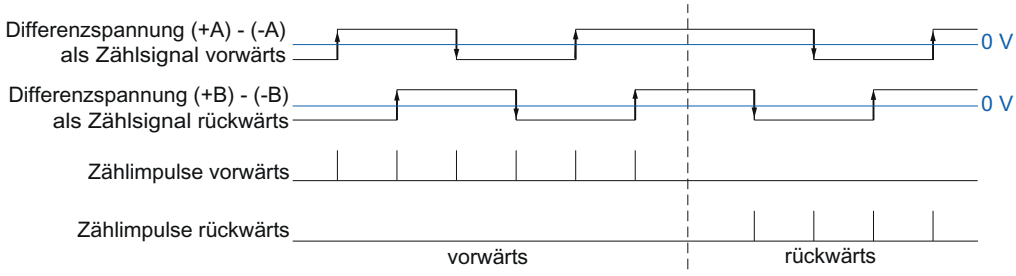

Taktsynchronität (TM Count und TM PosInput)

Das Technologiemodul unterstützt die Systemfunktion "Taktsynchronität". Mit dieser Systemfunktion lassen sich Positions-, Zähl- und Messwerte in einem festen Systemtakt erfassen.

Bei Taktsynchronität werden der Takt des Anwenderprogramms, die Übertragung der Eingangs- und Ausgangsdaten sowie die Bearbeitung im Modul aufeinander synchronisiert. Die Ausgangssignale schalten sofort, wenn die jeweilige Vergleichsbedingung erfüllt ist. Eine Zustandsänderung eines Digitaleingangs bewirkt sofort die vorgesehene Reaktion des Technologiemoduls und die Änderung des Status-Bits des Digitaleingangs in der Rückmeldeschnittstelle.

Bei Betrieb mit einem Technologieobjekt "Zählen und Messen" verwenden Sie einen OB des Typs "Synchronous Cycle" (z.B. OB61). Im zugeordneten OB wird die Anweisung High_Speed_Counter oder SSI_Absolute_Encoder aufgerufen.

Bei Positionserfassung für ein Technologieobjekt "Motion Control" verwenden Sie einen OB des Typs "MC-Servo". Bei Verwendung der Technologieobjekte Nocken und Nockenspur ist Taktsynchronität notwendig. Bei Verwendung des Technologieobjekts Messtaster in Verbindung mit dem Hardware-Digitaleingang DI1 ist keine Taktsynchronität notwendig.

Bei manuellem Betrieb verwenden Sie einen OB des Typs "Synchronous Cycle" (z.B. OB61). Im zugeordneten OB werden die Eingangs- und Ausgangsdaten verarbeitet.

Bearbeitung der Daten

Die Daten, die im aktuellen Buszyklus über die Steuerschnittstelle an das Technologiemodul übergeben wurden, werden wirksam, wenn sie im Rahmen des Technologiemodul-internen Zyklus bearbeitet werden. Zum Zeitpunkt des Einlesens der Eingangsdaten (T_i) werden der Positions- bzw. Zählwert und ggf. der Messwert sowie Status-Bits erfasst und in der Rückmeldeschnittstelle für das Abholen im aktuellen Buszyklus bereitgestellt.

Die Aktualisierungszeit für den Messwert wird in einem geeigneten Verhältnis auf den Systemtakt synchronisiert und falls notwendig, in der Länge angepasst. Wenn Sie "0" parametrieren, wird der Messwert einmal pro Systemtakt aktualisiert.

Taktsynchronparameter

Im taktsynchronen Betrieb können sich die folgenden Parameter auf die Taktsynchronparameter der Sync-Domain auswirken:

- Filterfrequenz
- Telegrammlänge[1]
- Übertragungsgeschwindigkeit[1]
- Monoflopzeit[1]
- Parität[1]

[1] Nur bei Verwendung eines SSI-Absolutwertgebers

Da die Taktsynchronparameter im RUN nicht geprüft werden, können Überläufe auftreten, wenn Sie im RUN einen oder mehrere der genannten Parameter ändern. Sie vermeiden Überläufe, indem Sie bereits in der Offline-Parametereinstellung die Option mit dem größten Zeitbedarf wählen.

Weitere Informationen

Eine ausführliche Beschreibung der Taktsynchronität finden Sie:

- Im Funktionshandbuch Taktsynchronität als Download im Internet (https://support.industry.siemens.com/cs/ww/de/view/109755401).
- Im Funktionshandbuch PROFINET mit STEP 7 als Download im Internet (https://support.industry.siemens.com/cs/de/de/view/49948856).

1.1.1.3 Grundlagen zu Zählen (TM Timer DIDQ)

Übersicht der Einsatzmöglichkeiten

Einleitung

Sie konfigurieren und parametrieren das TM Timer DIDQ mit der Projektierungssoftware.

Die Steuerung und Kontrolle der Funktionen des Moduls erfolgt durch das Anwenderprogramm über die Steuer- und Rückmeldeschnittstelle.

Systemumgebung

Das jeweilige Modul kann mit seinen Zählfunktionen in folgenden Systemumgebungen eingesetzt werden:

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Zentraler Betrieb mit einer CPU S7-1500 oder CPU 151xSP	• Automatisierungssystem S7-1500 oder ET 200SP CPU • TM Timer DIDQ	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration	Direkter Zugriff auf Rückmeldeschnittstelle des Technologiemoduls in den IO-Daten
Dezentraler Betrieb mit einer CPU S7-1500	• Automatisierungssystem S7-1500 • Dezentrales Peripheriesystem ET 200 • TM Timer DIDQ		
Dezentraler Betrieb mit einer CPU S7-300/400	• Automatisierungssystem S7-300/400 • Dezentrales Peripheriesystem ET 200 • TM Timer DIDQ	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration STEP 7: Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration (nur ET 200SP)	

Hinweis

Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum TM Timer DIDQ.

Zählen mit Inkrementalgeber

Sie können einige Kanäle eines TM Timer DIDQ für einfache Zählaufgaben mit einem Inkrementalgeber verwenden. Zählen ist das Erfassen und Aufsummieren von Ereignissen. Die als Zähler parametrierten Kanäle erfassen jeweils die zwei Inkrementalsignale und werten diese entsprechend aus.

Zählrichtung

Das Technologiemodul kann mit einem Inkrementalgeber vorwärts sowie rückwärts zählen. Sie können die Zählrichtung invertieren, um sie an den Prozess anzupassen.

Zählgrenzen

Die Zählgrenzen definieren den genutzten Wertebereich der Zählwerte.

Der minimale Zählwert ist –2147483648 (-2^{31}). Der maximale Zählwert ist 2147483647 ($2^{31}-1$). Der jeweilige Zähler zählt kontinuierlich. Bei einem Überlauf springt der Zähler auf die jeweils andere Zählgrenze und zählt weiter.

Der Zählwert kann nicht vom Anwenderprogramm beeinflusst werden.

Parametrierung

Zur Verwendung eines Zählers für einen Inkrementalgeber werden je zwei Digitaleingänge einer Kanalgruppe zusammengefasst. Dazu wählen Sie in den Kanalparametern für die jeweilige Gruppe die Konfiguration "Inkrementalgeber (A, B phasenversetzt)".

Hinweis

Zähler des TM Timer DIDQ 16x24V

Die Anzahl verfügbarer Zähler des TM Timer DIDQ 16x24V ist abhängig von der Kanalkonfiguration. Um 4 Zähler nutzen zu können, müssen Sie in der Kanalkonfiguration die Verwendung von acht Eingängen wählen. Wenn Sie die Verwendung von drei Eingängen wählen, können Sie 1 Zähler nutzen. Andere Kanalkonfigurationen ermöglichen keine Zählernutzung.

Rückgemeldeter Zählwert

Der aktuelle Zählwert wird in der Rückmeldeschnittstelle im Wert TEC_IN (DIm) angezeigt. DIm entspricht jeweils dem ersten der beiden zusammengefassten Digitaleingänge. Für den zweiten Digitaleingang wird im Wert TEC_IN (DIm+1) "0" zurückgeliefert.

Zählen mit Impulsgeber

Sie können einige Kanäle eines TM Timer DIDQ für einfache Zählaufgaben mit einem Impulsgeber verwenden. Zählen ist das Erfassen und Aufsummieren von Ereignissen. Die als Zähler parametrierten Kanäle erfassen jeweils ein Impulssignal und werten dieses entsprechend aus.

Zählrichtung

Das Technologiemodul kann mit einem Impulsgeber vorwärts zählen.

Zählgrenzen

Die Zählgrenzen definieren den genutzten Wertebereich der Zählwerte.

Der minimale Zählwert ist –2147483648 (-2^{31}). Der maximale Zählwert ist 2147483647 ($2^{31}-1$). Der jeweilige Zähler zählt kontinuierlich. Bei einem Überlauf springt der Zähler auf die jeweils andere Zählgrenze und zählt weiter.

Der Zählwert kann nicht vom Anwenderprogramm beeinflusst werden.

Parametrierung

Zur Verwendung eines Zählers für einen Impulsgeber wählen Sie in den Kanalparametern für die jeweilige Gruppe die Konfiguration "Eingänge einzeln verwenden" oder "Eingang/Ausgang einzeln verwenden". Sie können den ersten Digitaleingang einer Gruppe als Zähler parametrieren.

Hinweis

Zähler des TM Timer DIDQ 16x24V

Die Anzahl verfügbarer Zähler des TM Timer DIDQ 16x24V ist abhängig von der Kanalkonfiguration. Um 4 Zähler nutzen zu können, müssen Sie in der Kanalkonfiguration die Verwendung von acht Eingängen wählen. Wenn Sie die Verwendung von drei Eingängen wählen, können Sie 1 Zähler nutzen. Andere Kanalkonfigurationen ermöglichen keine Zählernutzung.

Rückgemeldeter Zählwert

Der aktuelle Zählwert wird in der Rückmeldeschnittstelle im Wert TEC_IN (DIm) angezeigt. DIm entspricht dem jeweiligen Digitaleingang.

24 V-Zählsignale

Zählsignale von 24 V-Inkrementalgebern

Der 24 V-Inkrementalgeber liefert die 24 V-Signale A und B an das Technologiemodul. Die Signale A und B sind zueinander um 90° phasenversetzt.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines 24 V-Inkrementalgebers:

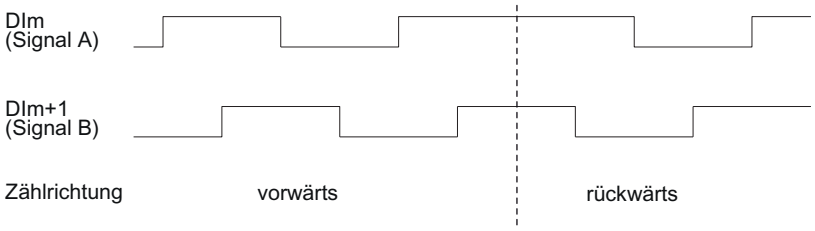

Das Technologiemodul erkennt die Zählrichtung an der Abfolge der Flanken an den Signalen A und B. Sie können eine Invertierung der Zählrichtung festlegen.

Signalauswertung

Die beiden phasenversetzten Signale eines Inkrementalgebers werden vierfach ausgewertet. Bei der Vierfachauswertung werden die steigenden und fallenden Flanken an Signal A und Signal B ausgewertet.

Ob Zählimpulse in Vorwärts- oder Rückwärtsrichtung erzeugt werden, hängt von der Flankenrichtung des einen Signals und dem Pegel des jeweils anderen Signals währenddessen ab.

Das folgende Bild zeigt ein Beispiel für die Vierfachauswertung von 24 V-Zählsignalen:

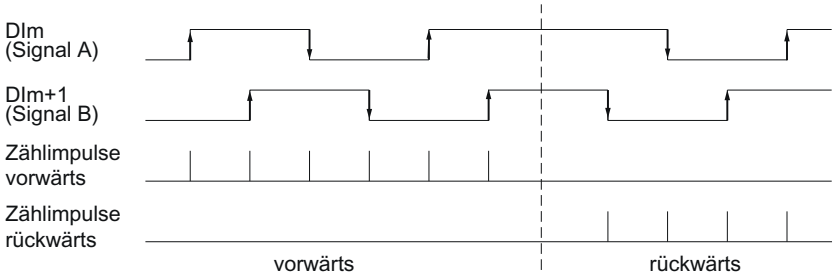

Zählsignale von 24 V-Impulsgebern

Ein Geber, zum Beispiel ein Initiator (BERO) oder eine Lichtschranke, liefert nur ein Zählsignal, das an den Digitaleingang eines Zählers angeschlossen wird.

Sie können die steigenden oder die fallenden Flanken des Signals zählen.

Taktsynchronität

Das TM Timer DIDQ unterstützt die Systemfunktion "Taktsynchronität". Mit dieser Systemfunktion lassen sich Zählwerte in einem festen Systemtakt erfassen.

Bei Taktsynchronität werden der Takt des Anwenderprogramms, die Übertragung der Eingangs- und Ausgangsdaten sowie die Bearbeitung im Modul aufeinander synchronisiert.

Bearbeitung der Daten

Die Daten, die im aktuellen Buszyklus über die Steuerschnittstelle an das Modul übergeben wurden, werden wirksam, wenn sie im Rahmen des modulinternen Zyklus bearbeitet werden. Zum Zeitpunkt T_i werden der Zählwert sowie die Status-Bits erfasst und in der Rückmeldeschnittstelle für das Abholen im aktuellen Buszyklus bereitgestellt.

Weitere Informationen

Eine ausführliche Beschreibung der Taktsynchronität finden Sie:

- Im Funktionshandbuch Taktsynchronität als Download im Internet (https://support.industry.siemens.com/cs/ww/de/view/109755401).
- Im Funktionshandbuch PROFINET mit STEP 7 als Download im Internet (https://support.industry.siemens.com/cs/de/de/view/49948856).

1.1.1.4 Grundlagen zu Zählen (Digitalmodule)

Übersicht der Einsatzmöglichkeiten

Einleitung

Sie konfigurieren und parametrieren das Digitalmodul mit der Projektierungssoftware.

Die Steuerung und Kontrolle der Funktionen des Moduls erfolgt durch das Anwenderprogramm über die Steuer- und Rückmeldeschnittstelle.

Systemumgebung

Das jeweilige Modul kann in folgenden Systemumgebungen eingesetzt werden:

Einsatzmöglichkeiten	Benötigte Komponenten	Projektierungssoftware	Im Anwenderprogramm
Zentraler Betrieb mit einer CPU S7-1500 oder CPU ET 200SP	• Automatisierungssystem S7-1500 oder ET 200SP CPU • Digitalmodul	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration	Direkter Zugriff auf Steuer- und Rückmeldeschnittstelle des Digitalmoduls in den IO-Daten
Dezentraler Betrieb mit einer CPU S7-1500	• Automatisierungssystem S7-1500 • Dezentrales Peripheriesystem ET 200 • Digitalmodul		
Dezentraler Betrieb mit einer CPU S7-300/400	• Automatisierungssystem S7-300/400 • Dezentrales Peripheriesystem ET 200 • Digitalmodul	STEP 7 (TIA Portal): Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration STEP 7: Gerätekonfiguration und Parametereinstellung mit Hardware-Konfiguration (ET 200SP) oder GSD-Datei (ET 200MP)	
Dezentraler Betrieb in einem System anderer Hersteller	• Automatisierungssystem anderer Hersteller • Dezentrales Peripheriesystem ET 200 • Digitalmodul	Projektierungssoftware anderer Hersteller: Gerätekonfiguration und Parametereinstellung mit GSD-Datei	

Hinweis

Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Zählen mit Impulsgeber

Zählen ist das Erfassen und Aufsummieren von Ereignissen. Die Zähler der Module erfassen Impulssignale und werten diese entsprechend aus. Die Zählrichtung kann durch geeignete Geber- bzw. Impulssignale oder über die Parametrierung vorgegeben werden.

Sie können Rückmelde-Bits verwenden, um Digitalausgänge von Digitalausgabemodulen an definierten Zählwerten zu schalten.

Sie können das Verhalten der Zähler mit Hilfe der im Folgenden beschriebenen Funktionalitäten parametrieren.

Zählgrenzen

Die Zählgrenzen definieren den genutzten Wertebereich der Zählwerte. Die Zählgrenzen sind parametrierbar und zur Laufzeit über das Anwenderprogramm änderbar. Informieren Sie sich im Gerätehandbuch zum Modul über die maximal und minimal einstellbaren Zählgrenzen.

Sie können parametrieren, ob die Zählvorgänge bei Überschreiten einer Zählgrenze fortgesetzt oder beendet werden (automatischer Torstopp).

Startwert

Sie können einen Startwert innerhalb der Zählgrenzen parametrieren. Der Startwert ist zur Laufzeit über das Anwenderprogramm änderbar.

Torsteuerung

Das Öffnen und Schließen des Hardware-Tors (HW-Tor) und Software-Tors (SW-Tor) definiert das Zeitfenster, in dem die Zählsignale erfasst werden.

Die Steuerung des HW-Tors erfolgt extern über einen Digitaleingang des Digitalmoduls. Das HW-Tor kann durch Parametrierung aktiviert werden. Die Steuerung des SW-Tors erfolgt über das Anwenderprogramm. Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Verhalten an den Zählgrenzen

Überschreiten einer Zählgrenze

Die obere Zählgrenze ist überschritten, wenn der aktuelle Zählwert der oberen Zählgrenze entspricht und ein weiterer Zählimpuls in Vorwärtsrichtung kommt. Die untere Zählgrenze ist überschritten, wenn der aktuelle Zählwert der unteren Zählgrenze entspricht und ein weiterer Zählimpuls in Rückwärtsrichtung kommt.

Bei Digitalmodulen für ET 200SP und ET 200AL wird in der Rückmeldeschnittstelle beim Überschreiten das entsprechende Ereignis-Bit gesetzt. Sie können ein Ereignis-Bit mit dem jeweiligen Steuerbit zurücksetzen:

Zählgrenze überschritten	Ereignis-Bit	Rücksetz-Bit
Obere Zählgrenze	EVENT_OFLW	RES_EVENT_OFLW
Untere Zählgrenze	EVENT_UFLW	RES_EVENT_UFLW

Hinweis

Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Sie können für das Überschreiten einer Zählgrenze parametrieren, ob auf der anderen Zählgrenze weitergezählt wird.

Hinweis

Die obere Zählgrenze und der Startwert definieren den Wertebereich des Zählers:

Wertebereich des Zählers = (Obere Grenze – Startwert) + 1

Beispiele

Das folgende Bild zeigt ein Beispiel für das Beenden des Zählvorgangs nach einem Überlauf und dem Setzen des Zählers auf die andere Zählgrenze:

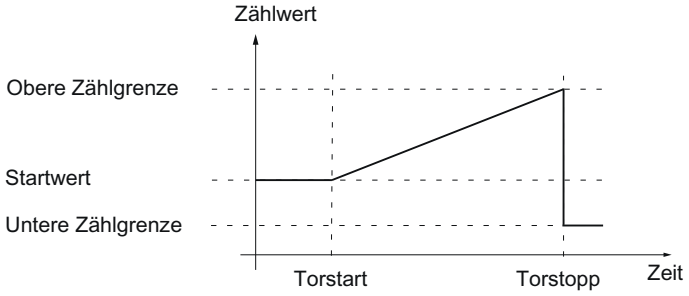

Das folgende Bild zeigt ein Beispiel für das Weiterzählen nach einem Überlauf und dem Setzen des Zählers auf die andere Zählgrenze:

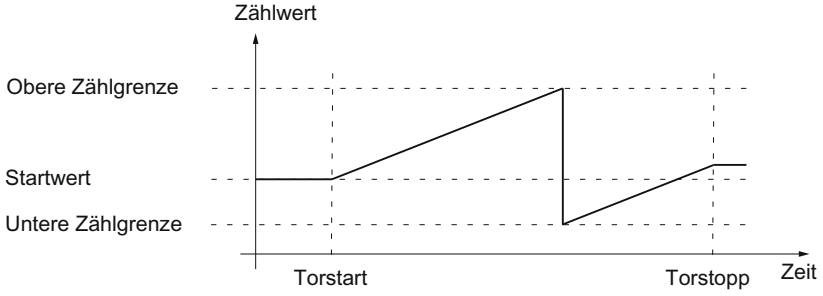

Torsteuerung

Viele Anwendungen erfordern, dass der Zählvorgang abhängig von anderen Ereignissen gestartet oder gestoppt werden soll. Dieses Starten und Stoppen des Zählvorgangs geschieht über die Torfunktion.

Die Digitalmodule besitzen pro Zählkanal bis zu zwei Tore, die das resultierende interne Tor bestimmen:

- Software-Tor (SW-Tor)
- Hardware-Tor (HW-Tor)

Hinweis

Das HW-Tor ist nicht bei allen Digitalmodulen verfügbar.

Software-Tor

Sie öffnen und schließen das SW-Tor des Kanals mit dem Steuerbit SW_GATE.

Hinweis

Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Hardware-Tor

Das HW-Tor ist optional. Sie öffnen und schließen das HW-Tor über Signale am entsprechenden Digitaleingang.

Hinweis

Eine parametrierte Eingangsverzögerung verzögert das Steuersignal des Digitaleingangs.

Der Zustand eines Digitaleingangs DIm wird mit dem jeweiligen Rückmeldebit STS_DIm angezeigt. Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Öffnen und Schließen des HW-Tors

Das folgende Bild zeigt ein Beispiel für das Öffnen und Schließen mit einem Digitaleingang:

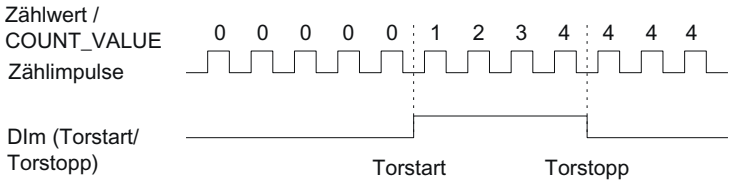

Solange der Digitaleingang gesetzt ist, ist das HW-Tor geöffnet und die Zählimpulse werden gezählt. Wenn der Digitaleingang rückgesetzt wird, wird das HW-Tor geschlossen. Die Zählimpulse werden ignoriert und der Zählwert bleibt konstant.

Internes Tor

Internes Tor

Das interne Tor ist offen, wenn das SW-Tor offen ist und das HW-Tor offen oder nicht parametriert ist. Der Zustand des internen Tors wird mit dem Rückmeldebit STS_GATE angezeigt. Eine Beschreibung der Steuer- und Rückmeldeschnittstelle finden Sie im Gerätehandbuch zum Digitalmodul.

Wenn das interne Tor geöffnet ist, wird der Zählvorgang gestartet. Wenn das interne Tor geschlossen ist, werden weitere Zählimpulse ignoriert und der Zählvorgang ist gestoppt.

Wenn Sie einen Zählvorgang nur mit dem HW-Tor kontrollieren wollen, muss das SW-Tor offen sein. Wenn Sie kein HW-Tor parametrieren, gilt das HW-Tor immer als offen. Sie öffnen und schließen das interne Tor dann nur über das SW-Tor.

HW-Tor	SW-Tor	Internes Tor
offen/nicht parametriert	offen	offen
offen/nicht parametriert	geschlossen	geschlossen
geschlossen	offen	geschlossen
geschlossen	geschlossen	geschlossen

Zusätzlich kann das interne Tor bei Überschreiten einer Zählgrenze automatisch geschlossen werden. Zum Fortsetzen des Zählvorgangs ist das Schließen und erneute Öffnen des Software-Tors oder Hardware-Tors notwendig.

Vergleichswerte

Sie können je nach Modul bis zu zwei Vergleichswerte festlegen, welche ein Rückmelde-Bit des Kanals unabhängig vom Anwenderprogramm steuern.

Bei zwei Vergleichswerten muss Vergleichswert 1 größer sein als Vergleichswert 0. Die Vergleichswerte sind parametrierbar und zur Laufzeit über das Anwenderprogramm änderbar.

Die Vergleichswerte werden mit dem aktuellen Zählwert verglichen. Wenn der Zählwert die parametrierte Vergleichsbedingung erfüllt, wird das jeweilige Rückmelde-Bit STS_DQ gesetzt.

Sie können das jeweilige Rückmelde-Bit verwenden, um einen Digitalausgang eines Digitalausgabemoduls zu schalten. Sie können das Setzen des jeweiligen Rückmelde-Bits STS_DQ von einer der folgenden Vergleichsereignisse abhängig machen. Informieren Sie sich im Gerätehandbuch zum Digitalmodul, welche der Vergleichsereignisse parametriert werden können.

Setzen zwischen Vergleichswert und oberer Zählgrenze

Das jeweilige Rückmelde-Bit STS_DQ wird auf 1 gesetzt, wenn:

Vergleichswert <= Zählwert <= obere Zählgrenze

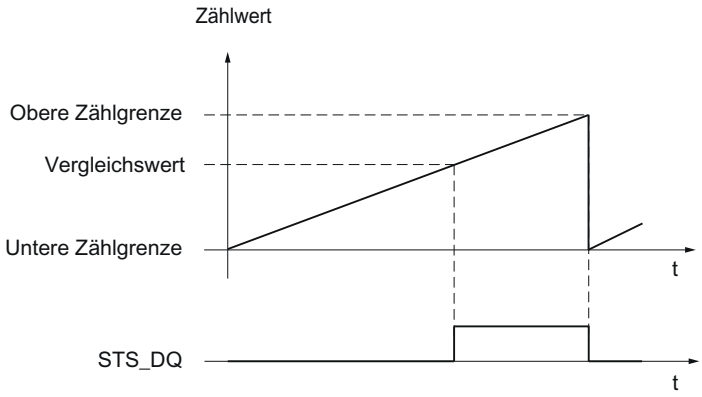

Setzen zwischen Vergleichswert und unterer Zählgrenze

Das jeweilige Rückmelde-Bit STS_DQ wird auf 1 gesetzt, wenn:

untere Zählgrenze <= Zählwert <= Vergleichswert

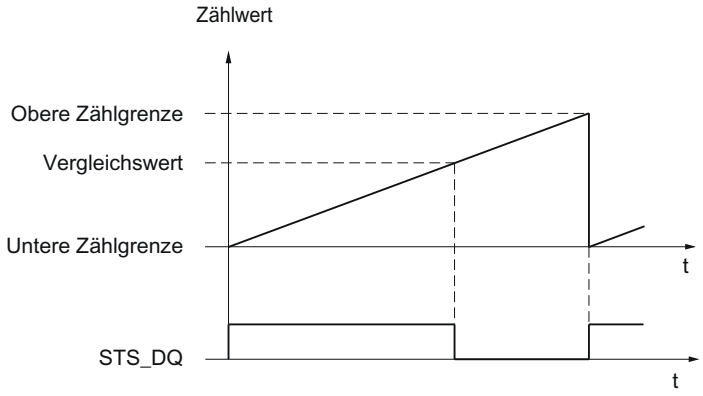

Setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das jeweilige Rückmelde-Bit STS_DQ wird auf 1 gesetzt, wenn:

Vergleichswert 0 <= Zählwert <= Vergleichswert 1

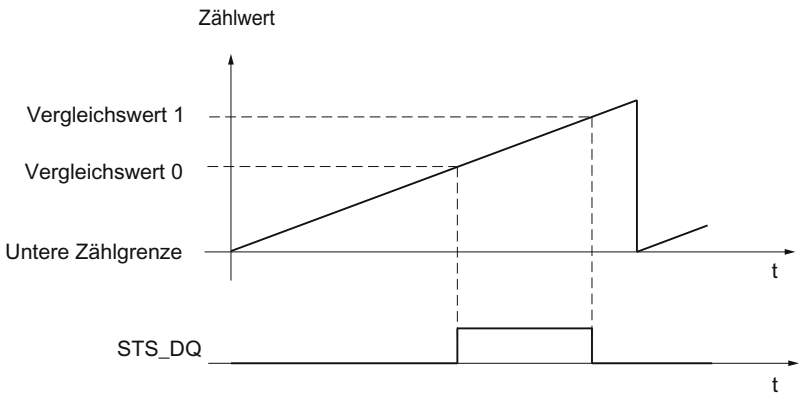

Nicht setzen zwischen Vergleichswert 0 und Vergleichswert 1

Das jeweilige Rückmelde-Bit STS_DQ wird auf 1 gesetzt, wenn:

Vergleichswert 0 <= Zählwert <= Vergleichswert 1

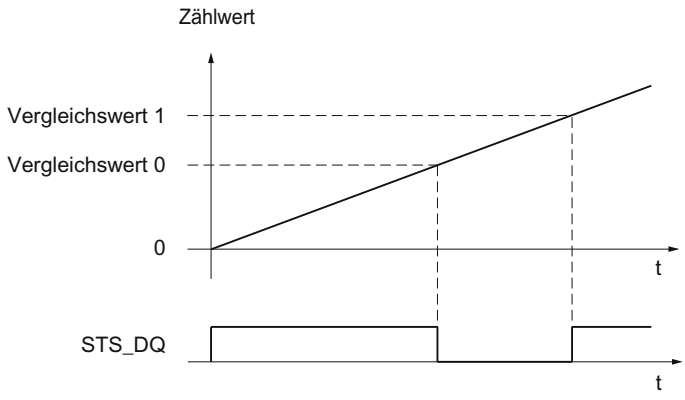

Alarme

Prozessalarm

Die Module können bei bestimmten Ereignissen während des Betriebs einen Prozessalarm in der CPU auslösen. Prozessalarme werden über die Parametrierung freigegeben. Informieren Sie sich im Gerätehandbuch des Moduls, welche Ereignisse im Betrieb einen Prozessalarm auslösen können.

Hinweis

Prozessalarme für das Zählen sind nicht bei allen Modulen vorhanden.

24 V-Zählsignale

Zählsignale von 24 V-Impulsgebern

Ein Geber, zum Beispiel ein Initiator (BERO) oder eine Lichtschranke, liefert ein Zählsignal, das an den Anschluss eines Zählers angeschlossen wird. Bei einigen Modulen können Sie über die Parametrierung der DI-Funktion die Zählrichtung ändern.

Das folgende Bild zeigt ein Beispiel für den zeitlichen Verlauf der Signale eines 24 V-Impulsgebers mit Richtungssignal und die daraus resultierenden Zählimpulse:

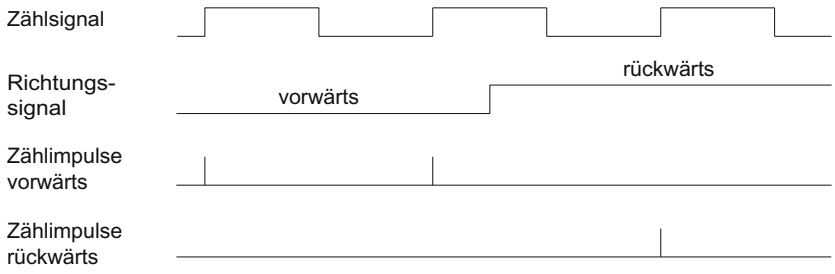

Hinweis

Ein Signal zur Richtungserkennung ist nicht bei allen Digitalmodulen anschließbar.

Taktsynchronität

Das Digitalmodul unterstützt die Systemfunktion "Taktsynchronität". Mit dieser Systemfunktion lassen sich Zählwerte in einem festen Systemtakt erfassen.

Bei Taktsynchronität werden der Takt des Anwenderprogramms, die Übertragung der Eingangs- und Ausgangsdaten sowie die Bearbeitung im Modul aufeinander synchronisiert.

Bearbeitung der Daten

Die Daten, die im aktuellen Buszyklus über die Steuerschnittstelle an das Modul übergeben wurden, werden wirksam, wenn sie im Rahmen des modulinternen Zyklus bearbeitet werden. Zum Zeitpunkt T_i werden der Zählwert sowie die Status-Bits erfasst und in der Rückmeldeschnittstelle für das Abholen im aktuellen Buszyklus bereitgestellt.

Weitere Informationen

Eine ausführliche Beschreibung der Taktsynchronität finden Sie:

- Im Funktionshandbuch Taktsynchronität als Download im Internet (https://support.industry.siemens.com/cs/ww/de/view/109755401).
- Im Funktionshandbuch PROFINET mit STEP 7 als Download im Internet (https://support.industry.siemens.com/cs/de/de/view/49948856).

1.1.2 Technologieobjekt High_Speed_Counter einsetzen

1.1.2.1 Konvention

Technologiemodul: Die Bezeichnung "Technologiemodul" verwenden wir im vorliegenden Kapitel sowohl für die Technologiemodule TM Count und TM PosInput als auch für den Technologieanteil der Kompakt-CPUs.

1.1.2.2 Technologieobjekt High_Speed_Counter

STEP 7 (TIA Portal) unterstützt Sie durch die Funktion "Technologieobjekte" bei der Projektierung, Inbetriebnahme und Diagnose der Zähl- und Messfunktionen für die Technologiemodule:

- In STEP 7 (TIA Portal) konfigurieren Sie das Technologieobjekt High_Speed_Counter mit den Einstellungen für die Zähl- und Messfunktionen.
- Im Anwenderprogramm programmieren Sie die zugehörige Anweisung High_Speed_Counter. Diese Anweisung übernimmt die Versorgung der Steuer- und Rückmeldeschnittstelle des Technologiemoduls.

Das Technologieobjekt High_Speed_Counter entspricht dem Instanz-DB der Anweisung High_Speed_Counter. Im Technologieobjekt wird die Konfiguration der Zähl- und Messfunktionen gespeichert. Das Technologieobjekt liegt im Ordner "PLC > Technologieobjekte".

Das Technologieobjekt High_Speed_Counter kann für die Technologiemodule der Systeme S7-1500 und ET 200SP gleichermaßen verwendet werden.

Betriebsmodus

Um ein Technologiemodul über das Technologieobjekt parametrieren zu können, legen Sie in der Hardware-Konfiguration des Technologiemoduls den Betriebsmodus (Seite 135) "Betrieb mit Technologieobjekt "Zählen und Messen"" fest. Diese Auswahl ist bereits voreingestellt.

1.1.2.3 Übersicht der Projektierungsschritte

Einleitung

Die nachfolgende Übersicht zeigt die grundsätzliche Vorgehensweise, um die Zähl- und Messfunktionen des Technologiemoduls über das Technologieobjekt High_Speed_Counter zu projektieren.

Voraussetzung (TM Count und TM PosInput)

Um das Technologieobjekt High_Speed_Counter einzusetzen, muss in STEP 7 (TIA Portal) ein Projekt mit einer CPU S7-1500 oder einer CPU ET 200SP angelegt sein.

Voraussetzung (Kompakt-CPU)

Um das Technologieobjekt High_Speed_Counter einzusetzen, muss in STEP 7 (TIA Portal) ein Projekt mit einer Kompakt-CPU S7-1500 angelegt sein.

Vorgehen

Gehen Sie in der nachfolgend empfohlenen Reihenfolge vor:

Schritt	Beschreibung
1	Technologiemodul konfigurieren (Seite 130)
2	Technologieobjekt hinzufügen (Seite 83)
3	Technologieobjekt entsprechend Ihrer Applikation konfigurieren (Seite 84)
4	Anweisung im Anwenderprogramm aufrufen (Seite 104)
5	Laden in CPU
6	Technologieobjekt in Betrieb nehmen (Seite 105)
7	Diagnose des Technologieobjekts (Seite 106)

1.1.2.4 Technologieobjekt hinzufügen

Technologieobjekt in der Projektnavigation hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Instanz-DB der Anweisung zu diesem Technologieobjekt erzeugt. In diesem Instanz-DB wird die Konfiguration des Technologieobjekts hinterlegt.

Voraussetzung (TM Count und TM PosInput)

Ein Projekt mit einer CPU S7-1500 ist angelegt.

Voraussetzung (Kompakt-CPU)

Ein Projekt mit einer Kompakt-CPU S7-1500 ist angelegt.

Vorgehen

Um ein Technologieobjekt hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie die Technologie "Zählen und Messen".
5. Wählen Sie das Objekt "High_Speed_Counter".
6. Geben Sie im Eingabefeld "Name" einen individuellen Namen für das Technologieobjekt ein.

7. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt hinterlegen möchten.
8. Bestätigen Sie mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

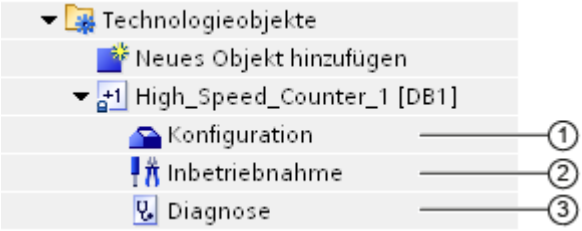

	Objekt	Beschreibung
①	Konfiguration (Seite 84)	Im Konfigurationsdialog: • Zuordnung des Technologiemoduls und des Kanals • Einstellung der Parameter des Technologieobjekts für die Zähl- und Messfunktionen Wenn Sie die Konfiguration des Technologieobjekts ändern, müssen Sie anschließend das Technologieobjekt **und** die Hardware-Konfiguration in die CPU laden.
②	Inbetriebnahme (Seite 105)	Inbetriebnahme und Funktionstest des Technologieobjekts: Parameter der Anweisung High_Speed_Counter simulieren und die Auswirkungen beobachten
③	Diagnose (Seite 106)	Überwachen der Zähl- und Messfunktionen

1.1.2.5 High_Speed_Counter konfigurieren

Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**
  Die Grundparameter enthalten die Auswahl des Technologiemoduls und die Nummer des Zählkanals, für den das Technologieobjekt konfiguriert wird.
- **Erweiterte Parameter**
  Die erweiterten Parameter enthalten die Parameter zur Anpassung der Zähl- und Messfunktionen und zur Einstellung des Verhaltens der Digitaleingänge und Digitalausgänge.

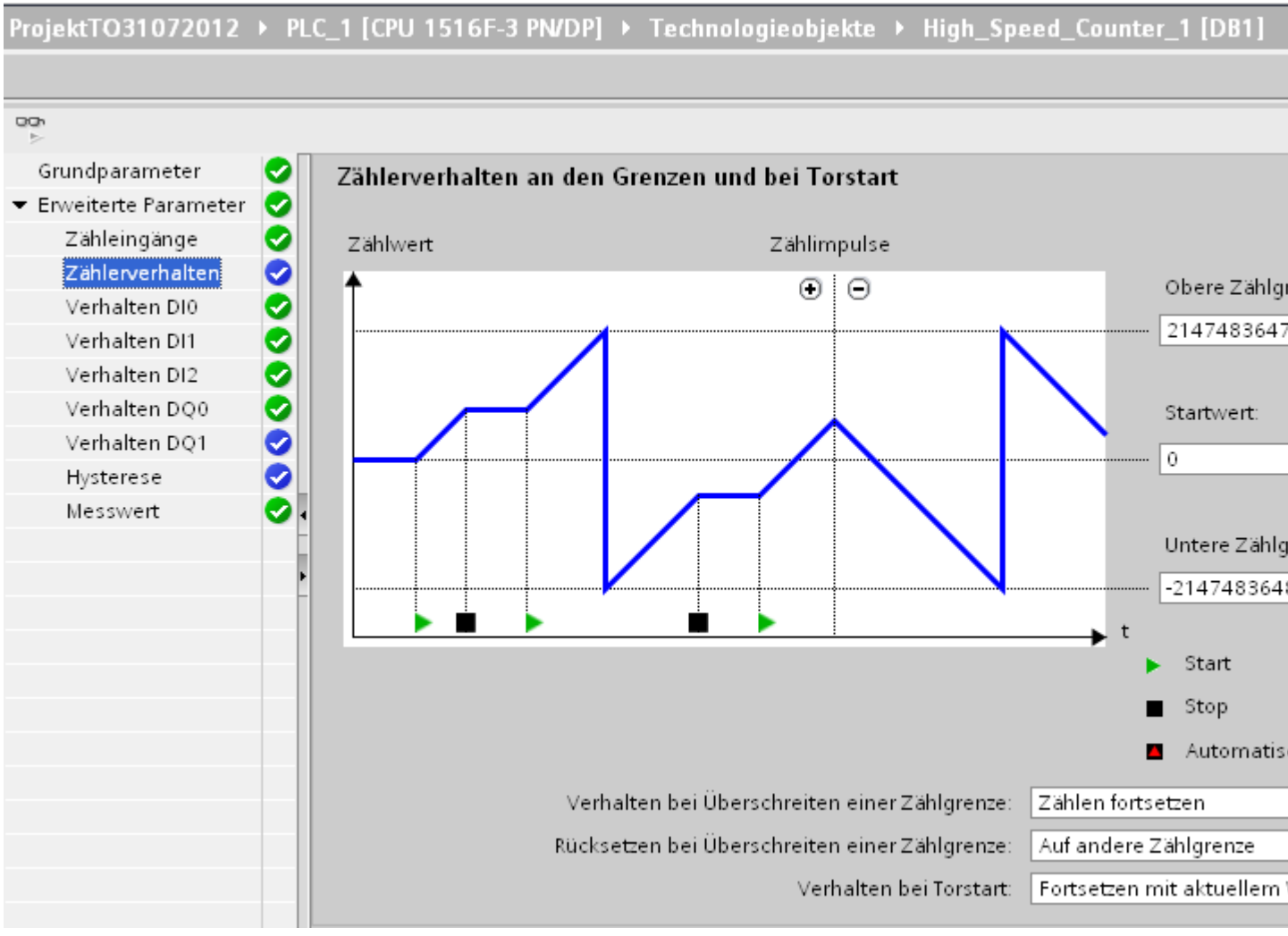

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.

Grundparameter

Unter "Grundparameter" stellen Sie die Verbindung zwischen dem Technologieobjekt High_Speed_Counter und dem Technologiemodul her.

Modul (TM Count und TM PosInput)

Über einen Folgedialog wählen Sie das Technologiemodul aus. Zur Auswahl stehen alle Technologiemodule (zentral oder dezentral), die unter der CPU S7-1500 oder CPU ET 200SP für den Einsatz mit einem Technologieobjekt aus "Zählen und Messen" konfiguriert sind.

Nach Auswahl des Technologiemoduls können Sie durch Klicken auf die Schaltfläche "Gerätekonfiguration" die zum Technologiemodul gehörige Gerätekonfiguration öffnen.

Die bei Verwendung des Technologieobjekts nötige Parametereinstellung des Technologiemoduls erfolgt über "Erweiterte Parameter" des Technologieobjekts.

Modul (Kompakt-CPU)

Über einen Folgedialog wählen Sie einen schnellen Zähler der Kompakt-CPU aus. Zur Auswahl stehen alle schnellen Zähler, die aktiviert und für den Einsatz mit einem Technologieobjekt aus "Zählen und Messen" konfiguriert sind.

Nach Auswahl des schnellen Zählers können Sie durch Klicken auf die Schaltfläche "Gerätekonfiguration" die zur Kompakt-CPU gehörige Gerätekonfiguration öffnen.

Die bei Verwendung des Technologieobjekts nötige Parametereinstellung des schnellen Zählers erfolgt über "Erweiterte Parameter" des Technologieobjekts.

Kanal

Bei einem Technologiemodul mit mehreren Zählkanälen wählen Sie zusätzlich die Nummer des Zählkanals aus, für den das Technologieobjekt High_Speed_Counter gilt.

Hinweis

Ein Kanal kann nur *einem* Technologieobjekt zugeordnet werden. Ein Kanal, der bereits einem Technologieobjekt zugeordnet ist, ist nicht mehr auswählbar.

Abgleich der Parameterwerte

Wenn nach Zuordnung des Kanals zum Technologieobjekt eine Inkonsistenz zwischen den Parameterwerten im Eigenschaftendialog und im Technologieobjekt vorhanden ist, erscheint eine Schaltfläche mit einer diesbezüglichen Rückfrage. Durch Klicken auf die Schaltfläche werden innerhalb von STEP 7 (TIA Portal) die Parameterwerte im Eigenschaftendialog mit den Parameterwerten des Technologieobjekts überschrieben. Die aktuellen Parameterwerte des Technologieobjekts werden im Eigenschaftendialog angezeigt angezeigt (read-only).

Zähleingänge (High_Speed_Counter)

Signalart

Sie können unter folgenden Signalarten (Seite 62) auswählen:

Symbol	Signalart	Bedeutung	Weitere optionsspezifische Parameter
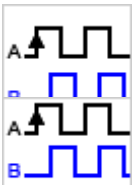	Inkrementalgeber (A, B phasenversetzt)	Ein Inkrementalgeber mit den phasenversetzten Signalen A und B ist angeschlossen.	• Signalauswertung • Richtung invertieren • Filterfrequenz • Sensortyp oder Schnittstellenstandard
  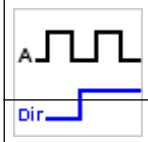	Inkrementalgeber (A, B, N)	Ein Inkrementalgeber mit den zueinander phasenversetzten Signalen A, B sowie einem Nullsignal N ist angeschlossen.	• Signalauswertung • Richtung invertieren • Filterfrequenz • Sensortyp oder Schnittstellenstandard • Verhalten bei Signal N • Häufigkeit der Synchronisation • Häufigkeit der Capture-Funktion
	Impuls (A) und Richtung (B)	Ein Impulsgeber (Signal A) mit Richtungssignal (Signal B) ist angeschlossen.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard

	Signalart	Bedeutung	Weitere optionsspezifische Parameter
	Impuls (A)	Ein Impulsgeber (Signal A) ohne Richtungssignal ist angeschlossen. Sie können die Zählrichtung über die Steuerschnittstelle (Seite 170) vorgeben.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard
	Vorwärts zählen (A), rückwärts zählen (B)	Signale zum Zählen in Vorwärtsrichtung (Signal A) und Rückwärtsrichtung (Signal B) sind angeschlossen.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard

Richtung invertieren

Sie können die Zählrichtung invertieren, um sie an den Prozess anzupassen.

Das Invertieren der Richtung ist bei den folgenden Signalarten parametrierbar und wirksam:

- Inkrementalgeber (A, B phasenversetzt)
- Inkrementalgeber (A, B, N)

Signalauswertung

Mit der Parametrierung der Signalauswertung (Seite 66) legen Sie fest, welche Flanken der Signale gezählt werden.

Sie können unter folgenden Optionen auswählen:

Symbol	Signalauswertung	Bedeutung
	Einfach (Seite 66) (voreingestellt)	Die Flanken des Signals A während eines Low-Pegels des Signals B werden ausgewertet.
	Zweifach (Seite 67)	Jede Flanke des Signals A wird ausgewertet.
	Vierfach (Seite 67)	Jede Flanke der Signale A und B wird ausgewertet.

Der Parameter ist bei den folgenden Signalarten parametrierbar:

- Inkrementalgeber (A, B phasenversetzt)
- Inkrementalgeber (A, B, N)

Filterfrequenz

Mit der Parametrierung der Filterfrequenz unterdrücken Sie Störungen an den Zähleingängen A, B und N.

Die ausgewählte Filterfrequenz bezieht sich auf ein Impuls-Pausen-Verhältnis zwischen ca. 40:60 und ca. 60:40. Dadurch ergibt sich eine bestimmte minimale Impuls-/Pausendauer. Signalwechsel mit einer Dauer kürzer als die minimale Impuls-/Pausendauer werden unterdrückt.

Sie können unter folgenden Filterfrequenzen auswählen:

Filterfrequenz	Minimale Impuls-/Pausendauer
100 Hz	4,0 ms
200 Hz	2,0 ms
500 Hz	800 µs
1 kHz	400 µs
2 kHz	200 µs
5 kHz	80 µs
10 kHz	40 µs
20 kHz	20 µs
50 kHz	8,0 µs
100 kHz (voreingestellt bei Kompakt-CPU)	4,0 µs
200 kHz** (voreingestellt bei TM Count)	2,0 µs
500 kHz*	0,8 µs
1 MHz* (voreingestellt bei TM PosInput)	0,4 µs

* Nur bei TM PosInput verfügbar

** Nur bei TM Count und TM PosInput verfügbar

Sensortyp (TM Count)

Mit der Parametrierung des Sensortyps legen Sie für das TM Count fest, wie die Zähleingänge geschaltet werden.

Sie können unter folgenden Optionen auswählen:

Sensortyp	Bedeutung
P-Schalter (voreingestellt)	Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach 24VDC.
M-Schalter	Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach M.
Gegentakt (M- und P-Schalter)	Der Geber bzw. Sensor schaltet die Eingänge A, B und N wechselweise nach M und 24VDC.

Bei Verwendung von Inkrementalgebern wird typischerweise "Gegentakt" gewählt. Bei Verwendung von 2-Draht-Sensoren, z. B. Lichtschranken oder Näherungsschalter, muss entsprechend der Verdrahtung "P-Schalter" oder "M-Schalter" gewählt werden.

Ob Ihr Inkrementalgeber Gegentakt-schaltend ist, entnehmen Sie dem Datenblatt des Gebers.

Hinweis

Wenn Sie einen Gegentakt-schaltenden Geber verwenden und der Sensortyp "Gegentakt (M- und P-Schalter)" parametriert ist, können Sie die Gebersignale auf Drahtbruch überwachen.

Sensortyp (Kompakt-CPU)

Für eine Kompakt-CPU ist der Sensortyp "P-Schalter" eingestellt und nicht änderbar. Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach 24VDC.

Sie können an der Kompakt-CPU sowohl P-schaltende als auch Gegentakt-schaltende Geber betreiben. Informationen zum Sensortyp entnehmen Sie dem Datenblatt des Gebers.

Schnittstellenstandard (TM PosInput)

Mit diesem Parameter legen Sie für das TM PosInput fest, ob der Geber symmetrische (RS422) oder asymmetrische Signale (TTL) liefert.

Sie können unter folgenden Optionen auswählen:

Schnittstellenstandard	Bedeutung
RS422, symmetrisch (voreingestellt)	Der Geber liefert symmetrische Signale gemäß dem RS422-Standard (Seite 64).
TTL (5 V), asymmetrisch	Der Geber liefert asymmetrische 5 V-Signale gemäß dem TTL-Standard (Seite 62).

Hinweis

Der RS422-Standard bietet eine höhere Störfestigkeit als der TTL-Standard. Wenn Ihr Inkremental- oder Impulsgeber den RS422- **und** den TTL-Standard beherrscht, wird deshalb der RS422-Standard empfohlen.

Verhalten bei Signal N

Mit diesem Parameter legen Sie fest, welches Verhalten bei Signal N ausgelöst wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Keine Reaktion auf Signal N (voreingestellt)	Der Zähler wird durch das Signal N nicht beeinflusst.
Synchronisation bei Signal N (Seite 37)	Der Zähler wird bei Signal N auf den Startwert gesetzt. Wenn Sie für einen Digitaleingang die Funktion "Freigabe Synchronisation bei Signal N" wählen, ist die Synchronisation vom Pegel am Digitaleingang abhängig.
Capture bei Signal N (Seite 28)	Der Zählwert wird bei Signal N in den Capture-Wert gespeichert. Die Verwendung eines Digitaleingangs und die Verwendung des Signals N schliessen sich für die Capture-Funktion nicht gegenseitig aus. Das Technologieobjekt zeigt den Capture-Wert am Ausgangsparameter CapturedValue an.

Hinweis

Sie können das Verhalten bei Signal N nur auswählen, wenn Sie die Signalart "Inkrementalgeber (A, B, N)" gewählt haben.

Hinweis

Wenn Sie " Synchronisation bei Signal N" auswählen, können Sie für einen Digitaleingang (Seite 93) die Funktion "Freigabe Synchronisation bei Signal N" wählen.

Hinweis

Für High_Speed_Counter ab V3.0 gilt:

"Capture bei Signal N" können Sie nur in der Betriebsart "Zählwert als Bezug verwenden" wählen.

Häufigkeit der Synchronisation

Mit diesem Parameter legen Sie die Häufigkeit folgender Ereignisse fest:

- Synchronisation bei Signal N
- Synchronisation als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig (voreingestellt)	Der Zähler wird nur bei dem ersten Signal N bzw. der ersten parametrierten Flanke des Digitaleingangs gesetzt.
Periodisch	Der Zähler wird bei jedem Signal N bzw. jeder parametrierten Flanke des Digitaleingangs gesetzt.

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie für folgende Funktionen die Häufigkeit von Capture-Ereignissen fest:

- Capture bei Signal N
- Capture als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang bzw. erste steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang bzw. jede steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.

Hinweis

Dieser Parameter ist für High_Speed_Counter ab V3.2 verfügbar.

Zählerverhalten

Zählgrenzen und Startwert

Obere Zählgrenze

Mit der Parametrierung der oberen Zählgrenze begrenzen Sie den Zählbereich. Sie können einen Wert bis 2147483647 (2^{31}-1) eingeben. Sie müssen einen Wert eingeben, der über der unteren Zählgrenze liegt.

Die Voreinstellung ist "2147483647".

Untere Zählgrenze

Mit der Parametrierung der unteren Zählgrenze begrenzen Sie den Zählbereich. Sie können einen Wert bis -2147483648 (-2^{31}) eingeben. Sie müssen einen Wert eingeben, der unter der oberen Zählgrenze liegt.

Die Voreinstellung ist "-2147483648".

Startwert

Mit der Parametrierung des Startwerts legen Sie fest, mit welchem Wert das Zählen begonnen und bei definierten Ereignissen weitergezählt wird. Sie müssen einen Wert zwischen den Zählgrenzen oder auf den Zählgrenzen eingeben.

Die Voreinstellung ist "0".

Weitere Informationen

Zusätzliche Informationen finden Sie unter Verhalten an den Zählgrenzen (Seite 23) und Zählerverhalten bei Torstart (Seite 27).

Zählerverhalten an den Grenzen und bei Torstart

Verhalten bei Überschreiten einer Zählgrenze

Sie können folgendes Verhalten bei Überschreiten einer Zählgrenze (Seite 23) parametrieren:

Verhalten	Bedeutung
Zählen stoppen	Nach dem Überschreiten einer Zählgrenze wird der Zählvorgang abgebrochen und das interne Tor geschlossen. Zum erneuten Starten des Zählvorgangs müssen Sie das SW-Tor oder HW-Tor gegebenenfalls schließen und erneut öffnen.
Zählen fortsetzen (voreingestellt)	Der Zählvorgang wird in Abhängigkeit der weiteren Parametrierung entweder mit dem Startwert oder an der jeweils anderen Zählgrenze fortgesetzt.

Rücksetzen bei Überschreiten einer Zählgrenze

Sie können den Zähler bei Überschreiten einer Zählgrenze auf folgende Werte setzen:

Rücksetzen des Werts	Bedeutung
Auf Startwert	Der Zählwert wird auf den Startwert gesetzt.
Auf andere Zählgrenze (voreingestellt)	Der Zählwert wird auf die jeweils andere Zählgrenze gesetzt.

Verhalten bei Torstart

Sie können folgendes Verhalten bei Torstart (Seite 27) parametrieren:

Verhalten	Bedeutung
Setzen auf Startwert	Bei Öffnen des Tors wird der Zählwert auf den Startwert gesetzt.
Fortsetzen mit aktuellem Wert (voreingestellt)	Bei Öffnen des Tors wird mit dem letzten Zählwert weitergezählt.

Verhalten eines DI (High_Speed_Counter)

Funktion des DI einstellen

Mit der Parametrierung eines Digitaleingangs legen Sie fest, welche Funktion der Digitaleingang beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Torstart/-stopp (pegelgesteuert)	Der Pegel am jeweiligen Digitaleingang öffnet und schließt das HW-Tor (Seite 26).	• Eingangsverzögerung • Pegelauswahl
Torstart (flankengesteuert)	Die parametrierte Flanke am jeweiligen Digitaleingang öffnet das HW-Tor (Seite 26).	• Eingangsverzögerung • Flankenauswahl
Torstopp (flankengesteuert)	Die parametrierte Flanke am jeweiligen Digitaleingang schließt das HW-Tor (Seite 26).	• Eingangsverzögerung • Flankenauswahl
Synchronisation (Seite 33)	Die parametrierte Flanke am jeweiligen Digitaleingang setzt den Zähler auf den Startwert. Das Technologieobjekt zeigt am Ausgangsparameter SyncStatus an, ob eine Synchronisation aufgetreten ist.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Synchronisation
Freigabe Synchronisation bei Signal N	Der aktive Pegel am jeweiligen Digitaleingang gibt die Synchronisation des Zählers bei Signal N (Seite 37) frei.	• Eingangsverzögerung • Pegelauswahl

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Capture	Die parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert (Seite 28) als Capture-Wert. Die Verwendung eines Digitaleingangs und die Verwendung des Signals N schliessen sich für die Capture-Funktion nicht gegenseitig aus. Das Technologieobjekt zeigt den Capture-Wert am Ausgangsparameter CapturedValue an.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Capture-Funktion • Verhalten des Zählwerts nach Capture
Digitaleingang ohne Funktion	Dem jeweiligen Digitaleingang ist keine technologische Funktion zugeordnet. Den Signalzustand des Digitaleingangs können Sie über die jeweilige statische Variable des Technologieobjekts lesen: • UserStatusFlags.StatusDI0 • UserStatusFlags.StatusDI1 • UserStatusFlags.StatusDI2	• Eingangsverzögerung

Hinweis

Jede Funktion, außer "Digitaleingang ohne Funktion", kann pro Zähler nur einmal verwendet und bei den jeweils anderen Digitaleingängen nicht mehr gewählt werden.

Hinweis

Für High_Speed_Counter ab V3.0 gilt:

Die Funktion "Capture" können Sie nur in der Betriebsart "Zählwert als Bezug verwenden" wählen.

Eingangsverzögerung (TM Count und TM PosInput)

Mit diesem Parameter unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Hinweis

Sie parametrieren die Eingangsverzögerung unter "Verhalten DI0" für alle Digitaleingänge gemeinsam. Die Eingangsverzögerung wird zusätzlich unter "Verhalten DI1" und bei TM Count auch unter "Verhalten DI2" angezeigt.

Eingangsverzögerung (Kompakt-CPU)

Mit diesem Parameter unterdrücken Sie Störungen an den Digitaleingängen der Signale DIn. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Die Eingangsverzögerung für einen Digitaleingang einer Kompakt-CPU parametrieren Sie im Inspektorfenster der Gerätekonfiguration unter "Eigenschaften > DI 16/DQ 16 > Eingänge > Kanal n".

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms
- 0,4 ms
- 1,6 ms
- 3,2 ms (voreingestellt)
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Pegelauswahl

Mit diesem Parameter legen Sie den Pegel fest, bei dem der Digitaleingang aktiv ist.

Sie können unter folgenden Optionen auswählen:

Pegel	Bedeutung
Aktiv bei High-Pegel (voreingestellt)	Der jeweilige Digitaleingang ist aktiv, wenn er gesetzt ist.
Aktiv bei Low-Pegel	Der jeweilige Digitaleingang ist aktiv, wenn er nicht gesetzt ist.

Der Parameter ist bei den folgenden Funktionen eines Digitaleingangs parametrierbar:

- Torstart/-stopp (pegelgesteuert)
- Freigabe Synchronisation bei Signal N

Flankenauswahl

Mit diesem Parameter legen Sie fest, bei welcher Flanke eines Digitaleingangs die parametrierte Funktion ausgelöst wird.

Sie können, abhängig von der gewählten Funktion, unter folgenden Optionen auswählen:

- Bei steigender Flanke (voreingestellt)
- Bei fallender Flanke
- Bei steigender und fallender Flanke

Der Parameter ist bei den folgenden Funktionen eines Digitaleingangs parametrierbar:

- Torstart (flankengesteuert)
- Torstopp (flankengesteuert)
- Synchronisation
- Capture

Hinweis

"Bei steigender und fallender Flanke" ist nur für die Funktion "Capture" parametrierbar.

Häufigkeit der Synchronisation

Mit diesem Parameter legen Sie die Häufigkeit folgender Ereignisse fest:

- Synchronisation bei Signal N
- Synchronisation als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig (voreingestellt)	Der Zähler wird nur bei dem ersten Signal N bzw. der ersten parametrierten Flanke des Digitaleingangs gesetzt.
Periodisch	Der Zähler wird bei jedem Signal N bzw. jeder parametrierten Flanke des Digitaleingangs gesetzt.

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie für folgende Funktionen die Häufigkeit von Capture-Ereignissen fest:

- Capture bei Signal N
- Capture als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang bzw. erste steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang bzw. jede steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.

Hinweis

Dieser Parameter ist für High_Speed_Counter ab V3.2 verfügbar.

Verhalten des Zählwerts nach Capture

Sie können folgendes Verhalten des Zählers nach einem Capture-Ereignis (Seite 28) parametrieren:

Verhalten	Bedeutung
Zählen fortsetzen (voreingestellt)	Der Zählvorgang wird nach Speichern des aktuellen Zählwerts als Capture-Wert unverändert fortgesetzt.
Setzen auf Startwert und Zählen fortsetzen	Der Zählvorgang wird nach Speichern des aktuellen Zählwerts als Capture-Wert mit dem Startwert fortgesetzt.

Verhalten eines DQ (High_Speed_Counter)

Betriebsart (High_Speed_Counter ab V3.0)

Mit der Betriebsart legen Sie fest, mit welchem Wert die Vergleichsfunktionen arbeiten.

Betriebsart	Bedeutung
Zählwert als Bezug verwenden (voreingestellt)	Die Vergleichsfunktionen und Prozessalarme für Vergleichsereignisse arbeiten mit dem Zählwert. Diese Funktionalität entspricht der Funktionalität des High_Speed_Counter in den Versionen vor V3.0.
Messwert als Bezug verwenden	Die Vergleichsfunktionen und die Prozessalarme für Vergleichsereignisse arbeiten mit dem Messwert.

Hinweis

Sie parametrieren die Betriebsart unter "Verhalten DQ0" für beide Digitalausgänge gemeinsam. Die Betriebsart wird zusätzlich unter "Verhalten DQ1" angezeigt.

Ausgang setzen

Mit der Parametrierung eines Digitalausgangs legen Sie die Bedingung fest, bei welcher der Digitalausgang schaltet.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitalausgangs (Seite 41) in der Betriebsart "Zählwert als Bezug verwenden"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert und oberer Zählgrenze (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn: Vergleichswert <= Zählwert <= obere Zählgrenze	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert und unterer Zählgrenze	Der jeweilige Digitalausgang ist aktiv, wenn: untere Zählgrenze <= Zählwert <= Vergleichswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Zählwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Bei Vergleichswert für eine Impulsdauer	Der jeweilige Digitalausgang ist einmalig aktiv für die parametrierte Zeit und Zählrichtung, wenn der Zählwert dem Vergleichswert entspricht.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Impulsdauer • Hysterese (in Inkrementen)
Nach Setzbefehl aus CPU bis Vergleichswert	Wenn ein Setzbefehl aus der CPU erfolgt, ist der jeweilige Digitalausgang aktiv, bis der Zählwert dem Vergleichswert entspricht.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

DQ0 eines Zählers einer Kompakt-CPU

Bei einer Kompakt-CPU ist der jeweilige Digitalausgang DQ0 über die Rückmeldeschnittstelle, aber nicht als physikalischer Ausgang verfügbar.

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Hinweis

Die Funktionen "Bei Vergleichswert für eine Impulsdauer" und "Nach Setzbefehl aus CPU bis Vergleichswert" schalten den jeweiligen Digitalausgang nur dann, wenn der Vergleichswert durch einen Zählimpuls erreicht wird. Wenn der Zählwert z. B. durch Synchronisation gesetzt wird, schaltet der Digitalausgang nicht.

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messwert als Bezug verwenden"	Bedeutung	Weitere optionsspezifische Parameter
Messwert >= Vergleichswert (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert größer oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Messwert <= Vergleichswert	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert kleiner oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Messwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1
Nicht zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 1 <= Messwert <= Vergleichswert 0	• Vergleichswert 0 • Vergleichswert 1
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle (Seite 40) geschaltet werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" und "Nicht zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Vergleichswert 0 (TM Count und TM PosInput)

Betriebsart "Zählwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Ganzzahl (DINT) eingeben, die größer oder gleich der unteren Zählgrenze ist. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 0 (Kompakt-CPU)

Betriebsart "Zählwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert das Bit STS_DQ0 in der Rückmeldeschnittstelle aufgrund des gewählten Vergleichsereignisses gesetzt wird. Der Digitalausgang DQ0 ist bei einer Kompakt-CPU nicht als physikalischer Ausgang verfügbar.

Sie müssen eine Ganzzahl (DINT) eingeben, die größer oder gleich der unteren Zählgrenze. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert das Bit STS_DQ0 in der Rückmeldeschnittstelle aufgrund des gewählten Vergleichsereignisses gesetzt wird. Der Digitalausgang DQ0 ist bei einer Kompakt-CPU nicht als physikalischer Ausgang verfügbar.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 1

Betriebsart "Zählwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Ganzzahl (DINT) eingeben, die kleiner oder gleich der oberen Zählgrenze ist. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "10".

Betriebsart "Messwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Maximalwert beträgt $7{,}922816 \times 10^{28}$. Die Voreinstellung ist "10.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Zählrichtung

Mit diesem Parameter legen Sie fest, für welche Zählrichtung die gewählte Funktion gilt.

Sie können unter folgenden Optionen auswählen:

Zählrichtung	Bedeutung
In beide Richtungen (voreingestellt)	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden unabhängig von der Zählrichtung statt.
Vorwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Zähler vorwärts zählt.
Rückwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Zähler rückwärts zählt.

Der Parameter ist bei den folgenden Funktionen parametrierbar:

- Bei Vergleichswert für eine Impulsdauer
- Nach Setzbefehl aus CPU bis Vergleichswert

Impulsdauer

Mit der Parametrierung der Impulsdauer für die Funktion "Bei Vergleichswert für eine Impulsdauer" legen Sie fest, wie viele Millisekunden der jeweilige Digitalausgang aktiv ist.

Wenn Sie "0" eingeben und der Zählwert dem jeweiligen Vergleichswert entspricht, ist der Digitalausgang aktiv bis zum nächsten Zählimpuls.

Zulässig ist ein Wert von 0,0 bis 6553,5.

Die Voreinstellung ist "500,0" und entspricht einer Impulsdauer von 0,5 s.

Hysterese (in Inkrementen)

Mit der Parametrierung der Hysterese (Seite 57) legen Sie einen Bereich um die Vergleichswerte fest. Bei den Funktionen "Zwischen Vergleichswert und oberer Zählgrenze" und "Zwischen Vergleichswert und unterer Zählgrenze" gilt die Hysterese auch an den Zählgrenzen. Im Hysteresebereich können die Digitalausgänge nicht erneut schalten, bevor der Zählwert diesen Bereich einmal verlassen hat.

Wählen Sie die Hysterese hinreichend klein. Wenn der Hysteresebereich, ausgehend vom parametrierten Vergleichswert, den gesamten Zählbereich überspannt, kann die korrekte Funktion der Vergleichswerte nicht gewährleistet werden.

Wenn ein Vergleichswert so nahe an einer Zählgrenze liegt, dass der Hysteresebereich über diese Zählgrenze hinausreichen würde, endet der Hysteresebereich an ihr.

Wenn Sie "0" eingeben, ist die Hysterese abgeschaltet. Zulässig ist ein Wert von 0 bis 255. Die Voreinstellung ist "0".

Hinweis

Für High_Speed_Counter ab V3.0 gilt:

Sie parametrieren die Hysterese unter "Verhalten DQ0" für beide Digitalausgänge gemeinsam. Die Hysterese wird zusätzlich unter "Verhalten DQ1" angezeigt.

Die Hysterese ist nur in der Betriebsart "Zählwert als Bezug verwenden" verfügbar.

Messwert spezifizieren (High_Speed_Counter)

Messgröße

Mit diesem Parameter legen Sie fest, welche Messgröße (Seite 52) das Technologiemodul bereitstellen soll. Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.

Sie können unter folgenden Optionen auswählen:

Messgröße	Bedeutung	Weitere optionsspezifische Parameter
Frequenz (voreingestellt)	Die Messgröße zeigt die Anzahl der Inkremente pro Sekunde. Der Wert ist eine Gleitkommazahl (REAL). Die Einheit ist Hz. Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit
Periodendauer	Die Messgröße ist die durchschnittliche Periodendauer zwischen zwei Inkrementen. Der Wert ist eine Ganzzahl (DINT). Die Einheit ist s. Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit
Geschwindigkeit	Die Messgröße ist eine Geschwindigkeit. Beispiele für eine Geschwindigkeitsmessung finden Sie bei der Erläuterung des Parameters "Inkremente pro Einheit". Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit • Zeitbasis für Geschwindigkeitsmessung • Inkremente pro Einheit

Aktualisierungszeit

Mit der Parametrierung der Aktualisierungszeit (Seite 52) in Millisekunden legen Sie das Zeitintervall zwischen zwei Messwertaktualisierungen fest.

Die Aktualisierungszeit und die Signalart beeinflussen die Genauigkeit der Messung. Bei Aktualisierungszeiten von mindestens 100 ms ist der Einfluss der Signalart vernachlässigbar.

Bei Aktualisierungszeiten von weniger als 100 ms erreichen Sie die höchste Messgenauigkeit mit den folgenden Signalarten:

- Inkrementalgeber (A, B phasenversetzt) mit Signalauswertung "Einfach"
- Inkrementalgeber (A, B, N) mit Signalauswertung "Einfach"
- Impuls (A) und Richtung (B)
- Impuls (A)

Bei anderen Signalarten hängt die Messgenauigkeit vom verwendeten Geber und der Leitung ab.

Wenn Sie "0" eingeben, wird der Messwert einmal pro modulinternem Zyklus aktualisiert. Sie können bis zu drei Nachkommastellen eingeben. Zulässig ist ein Wert von 0.0 bis 25000.0. Die Voreinstellung ist "10.0".

Zeitbasis für Geschwindigkeitsmessung

Mit diesem Parameter legen Sie fest, mit welcher Zeitbasis die Geschwindigkeit bereitgestellt werden soll.

Sie können unter folgenden Optionen auswählen:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s

Die Voreinstellung ist "60 s".

Inkremente pro Einheit

Mit diesem Parameter legen Sie für die Geschwindigkeitsmessung die Anzahl der Zählimpulse pro relevanter Einheit fest, die der Inkremental- oder Impulsgeber liefert.

Die Anzahl der Zählimpulse ist abhängig von der parametrierten Signalauswertung. Zulässig ist ein Wert von 1 bis 65535.

Beispiel 1:

Ihr Geber liefert 4000 Zählimpulse beim Verfahren um einen Meter. Die Geschwindigkeit soll in Metern pro Sekunde gemessen werden. Als Signalauswertung ist "zweifach" parametriert.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 8000
- Zeitbasis für Geschwindigkeitsmessung: 1 s

Beispiel 2:

Ihr Geber liefert 4096 Zählimpulse pro Umdrehung. Die Geschwindigkeit soll in Umdrehungen pro Minute gemessen werden. Als Signalauswertung ist "Einfach" parametriert.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 4096
- Zeitbasis für Geschwindigkeitsmessung: 60 s

1.1.2.6 High_Speed_Counter programmieren

Anweisung High_Speed_Counter

High_Speed_Counter

Die Anweisung High_Speed_Counter gehört zum Technologieobjekt High_Speed_Counter und übernimmt die Versorgung der Steuer- und Rückmeldeschnittstelle des Technologiemoduls.

Die Anweisung High_Speed_Counter bildet somit die Softwareschnittstelle zwischen dem Anwenderprogramm und dem Technologiemodul. Sie muss zyklisch aus dem Anwenderprogramm aufgerufen werden, um die Eingangs- und Ausgangsdaten abzugleichen.

Die Anweisung High_Speed_Counter kann für die Technologiemodule der Systeme S7-1500 und ET 200SP gleichermaßen verwendet werden. Die Module können dabei zentral oder dezentral verwendet werden. Die Anweisung gilt jeweils für den Kanal des Technologiemoduls, der dem zugehörigen Technologieobjekt zugeordnet wurde.

Weitere Informationen

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

Anweisung im Anwenderprogramm aufrufen

Die Anweisung High_Speed_Counter kann im Zyklus oder alternativ in einem zeitgesteuerten Programm einmal pro Zähler aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Vorgehen

Um die Anweisung im Anwenderprogramm aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Programmbausteine".

3. Doppelklicken Sie auf den OB für die zyklische Programmbearbeitung.
   Der Baustein wird im Arbeitsbereich geöffnet.
4. Öffnen Sie im Fenster "Anweisungen" die Gruppe "Technologie" und den Ordner "Zählen und Messen".
   Der Ordner enthält die Anweisung.
5. Wählen Sie die Anweisung und ziehen Sie diese per Drag & Drop in Ihren OB.
   Der Dialog "Aufrufoptionen" wird geöffnet.
6. Wählen Sie aus der Liste "Name" ein Technologieobjekt oder geben Sie den Namen für ein neues Technologieobjekt ein.
7. Bestätigen Sie mit "OK".

Ergebnis

Wenn das Technologieobjekt noch nicht besteht, wird es hinzugefügt. Die Anweisung wird in den OB eingefügt. Das Technologieobjekt ist diesem Aufruf der Anweisung zugeordnet.

Hinweis

Wenn Sie auf eine der Schaltflächen "Konfiguration", "Inbetriebnahme" oder "Diagnose" in der Oberfläche der Anweisung klicken, öffnet sich der jeweilige Editor.

1.1.2.7 High_Speed_Counter in Betrieb nehmen

Technologieobjekt in Betrieb nehmen

Der Inbetriebnahme-Editor hilft Ihnen mit einer grafischen Darstellung des Bausteins bei der komfortablen Inbetriebnahme und dem Funktionstest des Technologieobjekts. Sie können bestimmte Parameter der Anweisung High_Speed_Counter im Online-Betrieb der CPU/IM ändern und deren Auswirkungen beobachten.

Voraussetzungen

- Es besteht eine Online-Verbindung zwischen STEP 7 (TIA Portal) und CPU.
- Die CPU ist im Betriebszustand RUN.
- Die zugehörige Anweisung High_Speed_Counter wird zyklisch aus dem Anwenderprogramm aufgerufen.
- Die Parameter des Technologieobjekts werden aus dem Anwenderprogramm nicht überschrieben.

Vorgehen

Um den Inbetriebnahme-Editor eines Technologieobjekts zu öffnen und eine Parameterwertänderung zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt High_Speed_Counter.
3. Doppelklicken Sie auf das Objekt "Inbetriebnahme".
   Die Funktionen zur Inbetriebnahme des Technologieobjekts High_Speed_Counter werden angezeigt.
4. Klicken Sie im Inbetriebnahmedialog oben links auf die Schaltfläche "Alle beobachten".
   Die Parameter (Online-Werte) des Technologieobjekts High_Speed_Counter werden geladen und angezeigt.
5. Wenn der Parameter, den Sie ändern wollen, ein Eingabefeld hat, geben Sie darin den neuen Wert ein.
6. Aktivieren Sie das Optionskästchen des Parameters.
   Der neue Parameterwert wird wirksam und die Auswirkungen der Änderung werden simuliert.

Online-Modus

Im Online-Modus können Sie folgende Parameter modifizieren und damit die Funktionsweise des Technologieobjekts testen:

- Neuer Zählwert (NewCountValue)
- Neue obere Zählgrenze (NewUpperLimit)
- Neue untere Zählgrenze (NewLowerLimit)
- Neuer Vergleichswert 0 (NewReferenceValue0 oder NewReferenceValue0_M)
- Neuer Vergleichswert 1 (NewReferenceValue1 oder NewReferenceValue1_M)
- Neuer Startwert (NewStartValue)
- Zähler starten und stoppen (SwGate)
- Capture freigeben (CaptureEnable)
- Synchronisation freigeben (SyncEnable)
- Bestätigung von gemeldeten Fehlerzuständen (ErrorAck)
- Rücksetzen der Statusflags (EventAck)

1.1.2.8 High_Speed_Counter Diagnose

Zählwerte, Messwerte, DIs und DQs beobachten

Mit den Diagnosefunktionen überwachen Sie die Zähl- und Messfunktionen.

Voraussetzungen

- Es besteht eine Online-Verbindung zwischen STEP 7 (TIA Portal) und CPU.
- Die CPU ist im Betriebszustand RUN.

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt High_Speed_Counter.
3. Doppelklicken Sie auf das Objekt "Diagnose".
4. Klicken Sie auf die Schaltfläche "Alle beobachten".

Anzeige

Folgende Werte werden vom Technologieobjekt aus der Rückmeldeschnittstelle gelesen und angezeigt:

- Ereignisanzeige/Diagnoseinformationen
- Signalzustände der Digitaleingänge und Digitalausgänge
- Zählwert
- Capture-Wert
- Messwert

Weitere Informationen zu den Statusanzeigen finden Sie in der Kontext-Hilfe zu jedem Ereignis in STEP 7 (TIA Portal). Wenn die CPU im STOP ist, wird die Statusanzeige nicht aktualisiert.

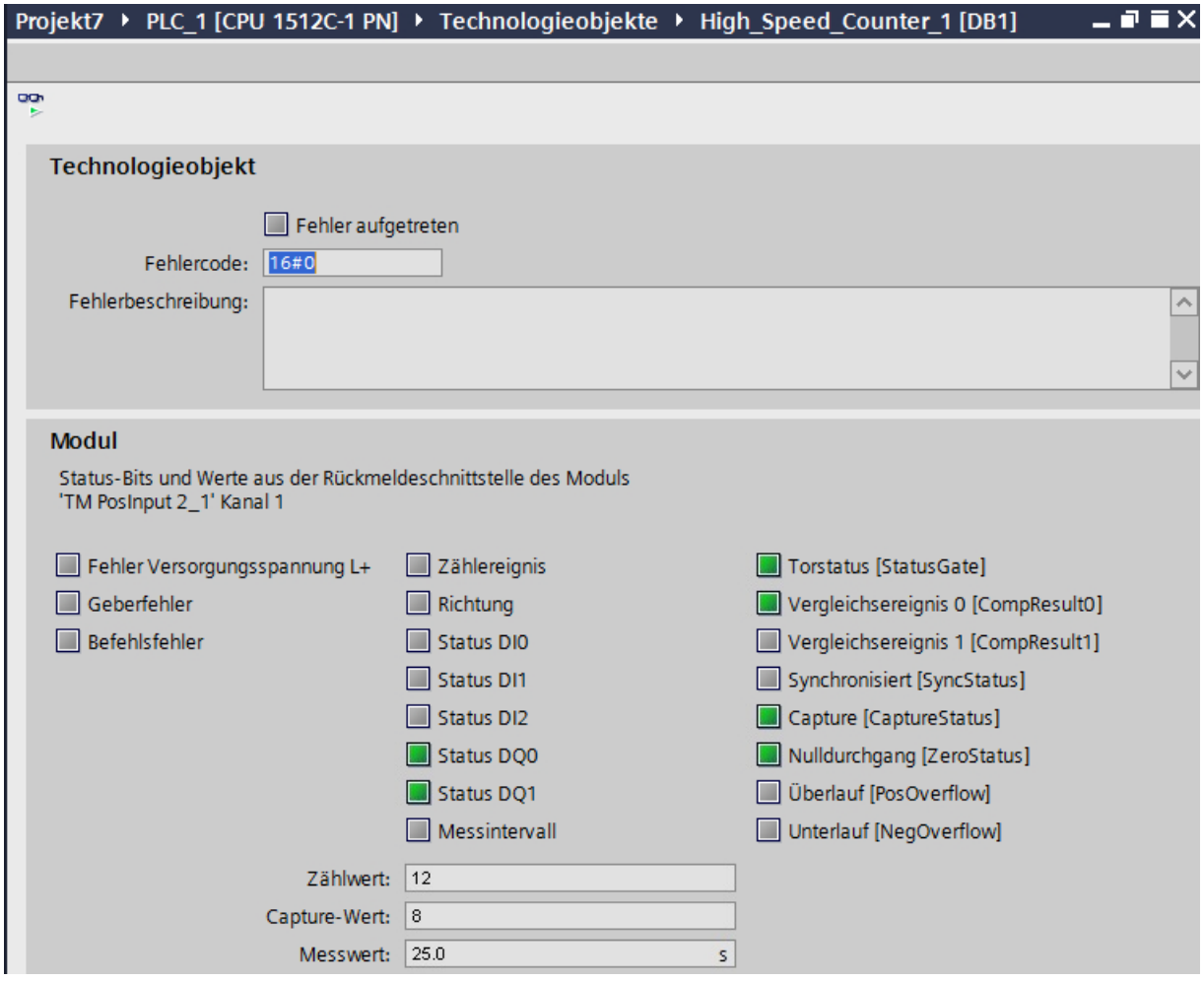

1.1.3 Technologieobjekt SSI_Absolute_Encoder einsetzen

1.1.3.1 Technologieobjekt SSI_Absolute_Encoder

STEP 7 (TIA Portal) unterstützt Sie durch die Funktion "Technologieobjekte" bei der Projektierung, Inbetriebnahme und Diagnose der Positionserfassungs- und Messfunktionen für das Technologiemodul TM PosInput in Kombination mit SSI-Absolutwertgebern:

- In STEP 7 (TIA Portal) konfigurieren Sie das Technologieobjekt SSI_Absolute_Encoder durch Eingabe der Geberparameter.
- Im Anwenderprogramm programmieren Sie die zugehörige Anweisung SSI_Absolute_Encoder. Diese Anweisung übernimmt die Versorgung der Steuer- und Rückmeldeschnittstelle des Technologiemoduls.

Das Technologieobjekt SSI_Absolute_Encoder entspricht dem Instanz-DB der Anweisung SSI_Absolute_Encoder. Im Technologieobjekt wird die Konfiguration der Positionserfassungs- und Messfunktionen gespeichert. Das Technologieobjekt liegt im Ordner "PLC > Technologieobjekte".

Das Technologieobjekt SSI_Absolute_Encoder kann für die TM PosInput der Systeme S7-1500 und ET 200SP gleichermaßen verwendet werden.

Betriebsmodus

Um ein TM PosInput über das Technologieobjekt parametrieren zu können, legen Sie in der Hardware-Konfiguration des TM PosInput den Betriebsmodus (Seite 135) "Betrieb mit Technologieobjekt "Zählen und Messen"" fest. Diese Auswahl ist bereits voreingestellt.

1.1.3.2 Übersicht der Projektierungsschritte

Einleitung

Die nachfolgende Übersicht zeigt die grundsätzliche Vorgehensweise, um die Positionserfassungs- und Messfunktionen des Technologiemoduls über das Technologieobjekt SSI_Absolute_Encoder zu projektieren.

Voraussetzung

Um das Technologieobjekt einzusetzen, muss in STEP 7 (TIA Portal) ein Projekt mit einer CPU S7-1500 oder einer CPU ET 200SP angelegt sein.

Vorgehen

Gehen Sie in der nachfolgend empfohlenen Reihenfolge vor:

Schritt	Beschreibung
1	Technologiemodul konfigurieren (Seite 130)
2	Technologieobjekt hinzufügen (Seite 109)
3	Technologieobjekt entsprechend Ihrer Applikation konfigurieren (Seite 111)
4	Anweisung im Anwenderprogramm aufrufen (Seite 126)
5	Laden in CPU
6	Technologieobjekt in Betrieb nehmen (Seite 127)
7	Diagnose des Technologieobjekts (Seite 128)

1.1.3.3 Technologieobjekt hinzufügen

Technologieobjekt in der Projektnavigation hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Instanz-DB der Anweisung zu diesem Technologieobjekt erzeugt. In diesem Instanz-DB wird die Konfiguration des Technologieobjekts hinterlegt.

Voraussetzung

Ein Projekt mit einer CPU S7-1500 ist angelegt.

Vorgehen

Um ein Technologieobjekt hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie die Technologie "Zählen und Messen".
5. Wählen Sie das Objekt "SSI_Absolute_Encoder".
6. Geben Sie im Eingabefeld "Name" einen individuellen Namen für das Technologieobjekt ein.
7. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt hinterlegen möchten.
8. Bestätigen Sie mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

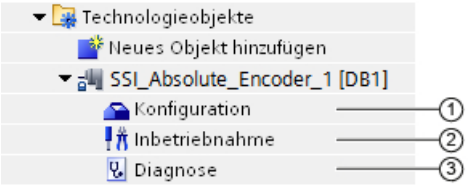

	Objekt	Beschreibung
①	Konfiguration (Seite 111)	Im Konfigurationsdialog: • Zuordnung des Technologiemoduls und des Kanals • Einstellung der Parameter des Technologieobjekts für die Positionserfassungs- und Messfunktionen Wenn Sie die Konfiguration des Technologieobjekts ändern, müssen Sie anschließend das Technologieobjekt **und** die Hardware-Konfiguration in die CPU laden.
②	Inbetriebnahme (Seite 127)	Inbetriebnahme und Funktionstest des Technologieobjekts: Parameter der Anweisung SSI_Absolute_Encoder simulieren und die Auswirkungen beobachten
③	Diagnose (Seite 128)	Überwachen der Positionserfassungs- und Messfunktionen

1.1.3.4 SSI_Absolute_Encoder konfigurieren

Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**
  Die Grundparameter enthalten die Auswahl des Technologiemoduls und die Nummer des Kanals, für den das Technologieobjekt konfiguriert wird.
- **Erweiterte Parameter**
  Die erweiterten Parameter enthalten die Parameter zur Anpassung der Positionserfassungs- und Messfunktionen und zur Einstellung des Verhaltens der Digitaleingänge und Digitalausgänge.

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.

Grundparameter

Unter "Grundparameter" stellen Sie die Verbindung zwischen dem Technologieobjekt und dem Technologiemodul TM PosInput her.

Modul

Über einen Folgedialog wählen Sie das Technologiemodul aus. Zur Auswahl stehen alle Technologiemodule TM PosInput (zentral oder dezentral) unter der CPU S7-1500 oder CPU ET 200SP, die für den Einsatz mit einem Technologieobjekt aus "Zählen und Messen" konfiguriert sind.

Nach Auswahl des Technologiemoduls können Sie durch Klicken auf die Schaltfläche "Gerätekonfiguration" die zum Technologiemodul gehörige Gerätekonfiguration öffnen.

Die bei Verwendung des Technologieobjekts nötige Parametereinstellung des Technologiemoduls erfolgt über "Erweiterte Parameter" des Technologieobjekts.

Kanal

Bei einem Technologiemodul mit mehreren Kanälen wählen Sie die Nummer des Kanals aus, für den das Technologieobjekt gilt.

Hinweis

Ein Kanal kann nur *einem* Technologieobjekt zugeordnet werden. Ein Kanal, der bereits einem Technologieobjekt zugeordnet ist, ist nicht mehr auswählbar.

Abgleich der Parameterwerte

Wenn nach Zuordnung des Kanals zum Technologieobjekt eine Inkonsistenz zwischen den Parameterwerten im Eigenschaftendialog des TM PosInput-Moduls und im Technologieobjekt vorhanden ist, erscheint eine Schaltfläche mit einer diesbezüglichen Rückfrage. Durch Klicken auf die Schaltfläche werden innerhalb von STEP 7 (TIA Portal) die Parameterwerte im Eigenschaftendialog des zugeordneten Moduls mit den Parameterwerten des Technologieobjekts überschrieben. Die aktuellen Parameterwerte des Technologieobjekts werden im Eigenschaftendialog des Moduls angezeigt (read-only).

SSI-Absolutwertgeber

Telegrammlänge

Mit der Parametrierung der Telegrammlänge legen Sie die Bitanzahl eines SSI-Telegramms des verwendeten SSI-Absolutwertgebers (Seite 22) fest. Die Telegrammlänge Ihres SSI-Absolutwertgebers entnehmen Sie dem Datenblatt des Gebers. Sonder-Bits zählen auch zur Telegrammlänge. Ein vorhandenes Paritäts-Bit zählt nicht zur Telegrammlänge.

Zulässig ist eine Telegrammlänge von 10 Bit bis 40 Bit. Die Voreinstellung ist "13 Bit".

Zwei Beispiele des Aufbaus des SSI-Telegramms finden Sie unter Beispiele des Telegrammaufbaus (Seite 123).

Codeart

Mit der Parametrierung der Codeart legen Sie fest, ob der Geber Dual-Code oder Gray-Code liefert.

Sie können unter folgenden Optionen auswählen:

Codeart	Bedeutung
Gray (voreingestellt)	Der vom SSI-Absolutwertgeber in Gray-Code gelieferte Positionswert wird in Dual-Code umgewandelt.
Dual	Der vom SSI-Absolutwertgeber gelieferte Wert wird nicht umgewandelt.

Übertragungsgeschwindigkeit

Mit der Parametrierung der Übertragungsgeschwindigkeit legen Sie die Datenübertragungsrate zwischen Technologiemodul und SSI-Absolutwertgeber fest.

Sie können unter folgenden Optionen auswählen:

- 125 kHz (voreingestellt)
- 250 kHz
- 500 kHz
- 1 MHz
- 1,5 MHz
- 2 MHz

Die maximale Übertragungsgeschwindigkeit ist abhängig von der Leitungslänge und den technischen Daten des SSI-Absolutwertgebers. Weitere Informationen entnehmen Sie dem Gerätehandbuch des TM PosInput und der Beschreibung des Gebers.

Monoflopzeit

Mit der Parametrierung der Monoflopzeit legen Sie die Pausenzeit zwischen zwei SSI-Telegrammen fest.

Die parametrierte Monoflopzeit muss mindestens so groß sein wie die Monoflopzeit des eingesetzten SSI-Absolutwertgebers. Sie finden diesen Wert in den technischen Daten des SSI-Absolutwertgebers.

Sie können unter folgenden Optionen auswählen:

- Automatisch (voreingestellt)
- 16 µs
- 32 µs
- 48 µs
- 64 µs

Hinweis

Wenn Sie die Option "Automatisch" wählen, passt sich die Monoflopzeit an den verwendeten Geber automatisch an.

Im taktsynchronen Betrieb entspricht die Option "Automatisch" einer Monoflopzeit von 64 µs. Wenn die Monoflopzeit des eingesetzten SSI-Absolutwertgebers kleiner ist als 64 µs, können Sie den Wert des Gebers auswählen, um schnellere taktsynchrone Zeiten zu erreichen.

Parität

Mit der Parametrierung der Parität legen Sie fest, ob der SSI-Absolutwertgeber ein Paritäts-Bit überträgt.

Wenn z. B. ein 25 Bit-Geber mit Parität parametriert ist, liest das Technologiemodul 26 Bit ein. Ein Paritätsfehler wird vom Technologieobjekt am Ausgangsparameter ErrorID mit dem Wert 80A2 angezeigt.

Bit-Nummer LSB des Positionswerts

Mit diesem Parameter legen Sie die Bit-Nummer des LSB (Least significant bit) des Positionswerts im Telegramm des SSI-Absolutwertgebers fest. Dadurch begrenzen Sie den Bereich im Telegramm, der den Positionswert liefert.

Der Wert muss kleiner als die Bit-Nummer des MSB des Positionswerts sein. Die Differenz zwischen den Bit-Nummern des MSB und des LSB des Positionswerts muss kleiner sein als 32.

Die Voreinstellung ist "0".

Hinweis

Wenn Sie als Codeart "Gray" gewählt haben, wird nur der Bereich vom LSB bis zum MSB des Positionswerts in Dual-Code umgewandelt.

Bit-Nummer MSB des Positionswerts

Mit diesem Parameter legen Sie die Bit-Nummer des MSB (Most significant bit) des Positionswerts im Telegramm des SSI-Absolutwertgebers fest. Dadurch begrenzen Sie den Bereich im Telegramm, der den Positionswert liefert.

Der Wert muss kleiner sein als die Telegrammlänge und größer als die Bit-Nummer des LSB des Positionswerts. Die Differenz zwischen den Bit-Nummern des MSB und des LSB des Positionswerts muss kleiner sein als 32.

Die Voreinstellung ist "12".

Hinweis

Wenn Sie als Codeart "Gray" gewählt haben, wird nur der Bereich vom LSB bis zum MSB des Positionswerts in Dual-Code umgewandelt.

Richtung invertieren

Mit diesem Parameter invertieren Sie die vom SSI-Absolutwertgeber gelieferten Werte. Dadurch können Sie die erfasste Richtung des Gebers an die Drehrichtung des Motors anpassen.

Hinweis

Dieser Parameter wirkt sich im Telegramm nur auf den Bereich vom LSB bis zum MSB des Positionswerts aus.

SSI-Telegramm

Folgende Parameter können Sie alternativ in der Grafik per Drag & Drop einstellen:

- Telegrammlänge
- Bit-Nummer LSB des Positionswerts
- Bit-Nummer MSB des Positionswerts

Vollständiges SSI-Telegramm

Wenn Sie als Messgröße "Vollständiges SSI-Telegramm" gewählt haben, liefert das Modul die niederwertigsten 32 Bit des unbearbeiteten aktuellen SSI-Telegramms als Messwert zurück. Die Grafik zeigt die jeweilige Bedeutung eines gelieferten Bits. Dabei werden folgende Abkürzungen verwendet:

V	Value: Positionswert als Gray- oder Dual-Code
S	Special: Sonder-Bit
P	Parity: Paritäts-Bit Wenn Sie ein Paritäts-Bit parametriert haben, liefert das Modul die niederwertigsten 31 Bit des SSI-Telegramms *und* das Paritäts-Bit zurück.

Verhalten eines DI (SSI_Absolute_Encoder)

Funktion des DI einstellen

Mit der Parametrierung eines Digitaleingangs legen Sie fest, welche Funktion der Digitaleingang beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Capture (Seite 31)	Die parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Positionswert als Capture-Wert. Das Technologieobjekt zeigt den Capture-Wert am Ausgangsparameter CapturedValue an. Die Funktion kann nur für einen der beiden Digitaleingänge verwendet werden.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Capture-Funktion
Digitaleingang ohne Funktion	Dem jeweiligen Digitaleingang ist keine technologische Funktion zugeordnet. Den Signalzustand des Digitaleingangs können Sie über die jeweilige statische Variable des Technologieobjekts lesen: • UserStatusFlags.StatusDI0 • UserStatusFlags.StatusDI1	• Eingangsverzögerung

Hinweis

Die Funktion "Capture" können Sie nur in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" wählen.

Eingangsverzögerung

Mit der Parametrierung der Eingangsverzögerung unterdrücken Sie Störungen an den Digitaleingängen. Signale mit einer Impulsdauer unterhalb der parametrierten Eingangsverzögerung werden unterdrückt.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Hinweis

Sie parametrieren die Eingangsverzögerung unter "Verhalten DI0" für alle Digitaleingänge gemeinsam. Die Eingangsverzögerung wird zusätzlich unter "Verhalten DI1" angezeigt.

Flankenauswahl

Mit diesem Parameter legen Sie für die Funktion "Capture" fest, bei welcher Flanke eines Digitaleingangs die parametrierte Funktion ausgelöst wird.

Sie können unter folgenden Optionen auswählen:

- Bei steigender Flanke (voreingestellt)
- Bei fallender Flanke
- Bei steigender und fallender Flanke

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie die Häufigkeit von Capture-Ereignissen (Seite 31) fest:

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert als Capture-Wert.

Hinweis

Dieser Parameter ist für SSI_Absolute_Encoder ab V3.0 verfügbar.

Verhalten eines DQ (SSI_Absolute_Encoder)

Betriebsart

Mit der Betriebsart legen Sie fest, mit welchem Bezugswert die Vergleichsfunktionen arbeiten.

Betriebsart	Bedeutung
Positionswert (SSI-Absolutwert) als Bezug verwenden (voreingestellt)	Die Vergleichsfunktionen und die Prozessalarme für Vergleichsereignisse arbeiten mit dem Positionswert.
Messwert als Bezug verwenden	Die Vergleichsfunktionen und die Prozessalarme für Vergleichsereignisse arbeiten mit dem Messwert.

Hinweis

Sie parametrieren die Betriebsart unter "Verhalten DQ0" für beide Digitalausgänge gemeinsam. Die Betriebsart wird zusätzlich unter "Verhalten DQ1" angezeigt.

Ausgang setzen

Mit der Parametrierung eines Digitalausgangs legen Sie die Bedingung fest, bei welcher der Digitalausgang schaltet.

Sie können je nach Betriebsart unter folgenden Optionen auswählen:

Funktion eines Digitalausgangs (Seite 45) in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert und oberer Zählgrenze (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn: Vergleichswert <= Positionswert <= maximaler Positionswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert und unterer Zählgrenze	Der jeweilige Digitalausgang ist aktiv, wenn: Minimaler Positionswert <= Positionswert <= Vergleichswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Positionswert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Bei Vergleichswert für eine Impulsdauer	Der jeweilige Digitalausgang ist einmalig aktiv für die parametrierte Zeit und Richtung der Positionswertänderung, wenn der Positionswert dem Vergleichswert entspricht oder ihn überschritten/unterschritten hat.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Impulsdauer • Hysterese (in Inkrementen)
Nach Setzbefehl aus CPU bis Vergleichswert	Wenn ein Setzbefehl aus der CPU erfolgt, ist der jeweilige Digitalausgang aktiv für die parametrierte Richtung der Positionswertänderung, bis der Positionswert dem Vergleichswert entspricht oder ihn überschritten/unterschritten hat.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messwert als Bezug verwenden"	Bedeutung	Weitere optionsspezifische Parameter
Messwert >= Vergleichswert (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert größer oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Messwert <= Vergleichswert	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert kleiner oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Messwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1
Nicht zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 1 <= Messwert <= Vergleichswert 0	• Vergleichswert 0 • Vergleichswert 1
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" und "Nicht zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Vergleichswert 0

Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 45) legen Sie fest, bei welchem Positionswert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge bis 31 Bit verwenden, müssen Sie eine positive Ganzzahl (DINT) eingeben mit einem Wert zwischen 0 und $2^{(MSB-LSB+1)}-1$. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge von 32 Bit verwenden, müssen Sie eine vorzeichenbehaftete Ganzzahl (DINT) eingeben mit einem Wert zwischen –2147483648 und 2147483647.

Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 1

Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 45) legen Sie fest, bei welchem Positionswert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge bis 31 Bit verwenden, müssen Sie eine positive Ganzzahl (DINT) eingeben mit einem Wert zwischen 0 und $2^{(MSB-LSB+1)}-1$. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge von 32 Bit verwenden, müssen Sie eine vorzeichenbehaftete Ganzzahl (DINT) eingeben mit einem Wert zwischen –2147483648 und 2147483647.

Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "10".

Betriebsart "Messwert als Bezug verwenden"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Maximalwert beträgt 7,922816 x 10^{28}. Die Voreinstellung ist "10.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Zählrichtung

Mit diesem Parameter legen Sie fest, für welche Richtung der Positionswertänderung die gewählte Funktion gilt.

Sie können unter folgenden Optionen auswählen:

Richtung der Positionswertänderung	Bedeutung
In beide Richtungen (voreingestellt)	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden unabhängig davon statt, ob der Positionswert kleiner oder größer wird.
Vorwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Positionswert größer wird.
Rückwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Positionswert kleiner wird.

Der Parameter ist bei den folgenden Funktionen parametrierbar:

- Bei Vergleichswert für eine Impulsdauer
- Nach Setzbefehl aus CPU bis Vergleichswert

Impulsdauer

Mit der Parametrierung der Impulsdauer für die Funktion "Bei Vergleichswert für eine Impulsdauer" legen Sie fest, wie viele Millisekunden der jeweilige Digitalausgang aktiv ist.

Zulässig ist ein Wert von 0,1 bis 6553,5 ms.

Die Voreinstellung ist "500,0" und entspricht einer Impulsdauer von 0,5 s.

Hysterese (in Inkrementen)

Mit der Parametrierung der Hysterese (Seite 59) legen Sie einen Bereich um die Vergleichswerte fest. Bei den Funktionen "Zwischen Vergleichswert und oberer Zählgrenze" und "Zwischen Vergleichswert und unterer Zählgrenze" gilt die Hysterese auch an den Zählgrenzen. Im Hysteresebereich können die Digitalausgänge nicht erneut schalten, bevor der Positionswert diesen Bereich einmal verlassen hat.

Wählen Sie die Hysterese hinreichend klein. Wenn der Hysteresebereich, ausgehend vom parametrierten Vergleichswert, den gesamten Positionswertebereich überspannt, kann die korrekte Funktion der Vergleichswerte nicht gewährleistet werden.

Wenn ein Vergleichswert so nahe an einer Zählgrenze liegt, dass der Hysteresebereich über diese Zählgrenze hinausreichen würde, endet der Hysteresebereich an ihr.

Wenn Sie "0" eingeben, ist die Hysterese abgeschaltet. Zulässig ist ein Wert von 0 bis 255. Die Voreinstellung ist "0".

Hinweis

Sie parametrieren die Hysterese unter "Verhalten DQ0" für beide Digitalausgänge gemeinsam. Die Hysterese wird zusätzlich unter "Verhalten DQ1" angezeigt.

Hinweis

Die Hysterese ist nur in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" verfügbar.

Messwert spezifizieren (SSI_Absolute_Encoder)

Messgröße

Mit diesem Parameter legen Sie fest, ob das Technologiemodul eine bestimmte Messgröße (Seite 55) oder das vollständige SSI-Telegramm bereitstellen soll.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung	Weitere optionsspezifische Parameter
Frequenz (voreingestellt)	Die Messgröße zeigt die Anzahl der Inkremente pro Sekunde, wobei jedes Inkrement einer Positionswertänderung entspricht. Der Wert ist eine Gleitkommazahl (REAL). Die Einheit ist Hz. Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit
Periodendauer	Die Messgröße ist die durchschnittliche Periodendauer zwischen zwei Inkrementen des Positionswerts. Der Wert ist eine Ganzzahl (DINT). Die Einheit ist s. Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit

Option	Bedeutung	Weitere optionsspezifische Parameter
Geschwindigkeit	Die Messgröße ist eine Geschwindigkeit. Beispiele für eine Geschwindigkeitsmessung finden Sie bei der Erläuterung des Parameters "Inkremente pro Einheit". Das Technologieobjekt zeigt den Messwert am Ausgangsparameter MeasuredValue an.	• Aktualisierungszeit • Zeitbasis für Geschwindigkeitsmessung • Inkremente pro Einheit
Vollständiges SSI-Telegramm	Anstelle einer Messgröße werden die ersten 32 Bits des SSI-Telegramms zurückgeliefert (Bit 0 bis Bit 31). Dabei werden auch Sonder-Bits mitgeliefert, die nicht zur Positionsinformation gehören. Eine parametrierte Richtungsinvertierung wird nicht berücksichtigt. Das Technologieobjekt zeigt den Wert am Ausgangsparameter CompleteSSIFrame an. Beispiele finden Sie unter Beispiele des Telegrammaufbaus (Seite 123). Diese Option ist nur in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" verfügbar.	—

Hinweis

Wenn zur Berechnung des Messwerts die Inkrementzahl pro Umdrehung benötigt wird, wird sie automatisch aus der parametrierten Telegrammlänge als 2er-Potenz berechnet, z.B. 8192 Inkremente pro Umdrehung bei einer Telegrammlänge von 13 Bit. Wenn Sie einen SSI-Absolutwertgeber verwenden, dessen Inkrementzahl pro Umdrehung nicht der 2er-Potenz entspricht, kann der berechnete Messwert kurzzeitig fehlerhaft sein.

Aktualisierungszeit

Mit der Parametrierung der Aktualisierungszeit (Seite 55) in Millisekunden legen Sie das Zeitintervall zwischen zwei Messwertaktualisierungen fest. Durch größere Aktualisierungszeiten können unruhige Messgrößen geglättet werden.

Wenn Sie "0" eingeben, wird der Messwert einmal pro modulinternem Zyklus aktualisiert. Sie können bis zu drei Nachkommastellen eingeben. Zulässig ist ein Wert von 0.0 bis 25000.0. Die Voreinstellung ist "10.0".

Zeitbasis für Geschwindigkeitsmessung

Mit diesem Parameter legen Sie fest, mit welcher Zeitbasis die Geschwindigkeit bereitgestellt werden soll.

Sie können unter folgenden Optionen auswählen:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s

Die Voreinstellung ist "60 s".

Inkremente pro Einheit

Mit diesem Parameter legen Sie für die Geschwindigkeitsmessung die Anzahl der Inkremente pro relevanter Einheit fest, die der SSI-Absolutwertgeber liefert.

Zulässig ist ein Wert von 1 bis 65535.

Beispiel 1:

Ihr Geber arbeitet mit einer Auflösung von 12 Bit pro Umdrehung und liefert 4096 Inkremente pro Umdrehung. Die Geschwindigkeit soll in Umdrehungen pro Minute gemessen werden.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 4096
- Zeitbasis für Geschwindigkeitsmessung: 60 s

Beispiel 2:

Ihr Geber liefert 10000 Inkremente beim Verfahren um einen Meter. Die Geschwindigkeit soll in Metern pro Sekunde gemessen werden.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 10000
- Zeitbasis für Geschwindigkeitsmessung: 1 s

Beispiele des Telegrammaufbaus

Beispiel 1

In diesem Beispiel hat der SSI-Absolutwertgeber folgende Spezifikation:

- Der Geber hat eine Auflösung von 13 Bit pro Umdrehung und einen Wertebereich von 12 Bit Umdrehungen. Das SSI-Telegramm hat eine Länge von 25 Bit.
- Das MSB des Positionswerts ist Bit 24.
- Das LSB des Positionswerts ist Bit 0.
- Der Positionswert ist Gray-codiert.
- Ein Paritäts-Bit ist nicht vorhanden.

Das Telegramm ist folgendermaßen aufgebaut:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Bit:	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG
	MSB																								LSB

Positionswert

MG Multiturn-Bit als Gray-Code

SG Singleturn-Bit als Gray-Code

Vollständiges SSI-Telegramm

Wenn Sie "Vollständiges SSI-Telegramm" parametrieren, liefert das Technologiemodul das unbearbeitete SSI-Telegramm in der Rückmeldeschnittstelle rechtsbündig zurück:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG

MG Multiturn-Bit als Gray-Code

SG Singleturn-Bit als Gray-Code

Rückmeldewert Positionswert

Der in Gray-Code gelieferte Positionswert wird vom Technologiemodul in Dual-Code umgewandelt und in der Rückmeldeschnittstelle rechtsbündig zurückgeliefert:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD

MD Multiturn-Bit als Dual-Code

SD Singleturn-Bit als Dual-Code

Beispiel 2

In diesem Beispiel hat der SSI-Absolutwertgeber folgende Spezifikation:

- Der Geber hat eine Auflösung von 17 Bit pro Umdrehung und einen Wertebereich von 11 Bit Umdrehungen. Das SSI-Telegramm hat eine Länge von 34 Bit.
- Das MSB des Positionswerts ist Bit 33.
- Das LSB des Positionswerts ist Bit 6.
- Der Positionswert ist Gray-codiert.
- Das SSI-Telegramm hat sechs Sonder-Bits.
- Ein Paritäts-Bit ist vorhanden. Das Paritäts-Bit zählt nicht zur Telegrammlänge.

Das Telegramm ist folgendermaßen aufgebaut:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
Bit:	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	*	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	S1	S2	P	S3	S4	S5	S6
	MSB																											LSB							

Positionswert

MG Multiturn-Bit als Gray-Code

SG Singleturn-Bit als Gray-Code

Sn Sonder-Bit n

P Paritäts-Bit

Vollständiges SSI-Telegramm

Wenn Sie "Vollständiges SSI-Telegramm" parametrieren, liefert das Technologiemodul die niederwertigsten 32 Bit des SSI-Telegramms als unbearbeitete Bitfolge zurück. Das Technologiemodul liefert das dem LSB folgende Bit als Paritäts-Bit zurück. Deshalb liefert das Technologiemodul in diesem Beispiel nur die niederwertigsten 31 Bit des SSI-Telegramms. Mit dem vollständigen SSI-Telegramm können Sie die zusätzlichen Sonder-Bits in ihrer Applikation auswerten.

Die zurückgelieferte Bitfolge ist folgendermaßen aufgebaut:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	P	S1	S2	S3	S4	S5	S6

MG Multiturn-Bit als Gray-Code

SG Singleturn-Bit als Gray-Code

Sn Sonder-Bit n

P Paritäts-Bit

Positionswert

Der in Gray-Code gelieferte Positionswert wird vom Technologiemodul in Dual-Code umgewandelt und in der Rückmeldeschnittstelle rechtsbündig zurückgeliefert. Die Sonder-bits werden dabei ignoriert. Das Paritäts-Bit wird ausgewertet, aber nicht mit dem Positionswert zurückgeliefert:

Takt:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD

MD Multiturn-Bit als Dual-Code

SD Singleturn-Bit als Dual-Code

1.1.3.5 SSI_Absolute_Encoder programmieren

Anweisung SSI_Absolute_Encoder

SSI_Absolute_Encoder

Die Anweisung SSI_Absolute_Encoder gehört zum Technologieobjekt SSI_Absolute_Encoder und übernimmt die Versorgung der Steuer- und Rückmeldeschnittstelle des Technologiemoduls TM PosInput.

Die Anweisung SSI_Absolute_Encoder bildet somit die Softwareschnittstelle zwischen dem Anwenderprogramm und dem Technologiemodul. Sie muss zyklisch aus dem Anwenderprogramm aufgerufen werden, um die Eingangs- und Ausgangsdaten abzugleichen.

Die Anweisung SSI_Absolute_Encoder kann für die TM PosInput der Systeme S7-1500 und ET 200SP gleichermaßen verwendet werden. Die Module TM PosInput können dabei zentral oder dezentral verwendet werden. Die Anweisung gilt jeweils für den Kanal des Technologiemoduls, der dem zugehörigen Technologieobjekt zugeordnet wurde.

Weitere Informationen

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

Anweisung im Anwenderprogramm aufrufen

Die Anweisung SSI_Absolute_Encoder kann im Zyklus oder alternativ in einem zeitgesteuerten Programm einmal pro Kanal aufgerufen werden. Der Aufruf in einem ereignisgesteuerten Alarmprogramm ist nicht zulässig.

Vorgehen

Um die Anweisung im Anwenderprogramm aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Programmbausteine".
3. Doppelklicken Sie auf den OB für die zyklische Programmbearbeitung.
   Der Baustein wird im Arbeitsbereich geöffnet.
4. Öffnen Sie im Fenster "Anweisungen" die Gruppe "Technologie" und den Ordner "Zählen und Messen".
   Der Ordner enthält die Anweisung.
5. Wählen Sie die Anweisung und ziehen Sie diese per Drag & Drop in Ihren OB.
   Der Dialog "Aufrufoptionen" wird geöffnet.
6. Wählen Sie aus der Liste "Name" ein Technologieobjekt oder geben Sie den Namen für ein neues Technologieobjekt ein.
7. Bestätigen Sie mit "OK".

Ergebnis

Wenn das Technologieobjekt noch nicht besteht, wird es hinzugefügt. Die Anweisung wird in den OB eingefügt. Das Technologieobjekt ist diesem Aufruf der Anweisung zugeordnet.

Hinweis

Wenn Sie auf eine der Schaltflächen "Konfiguration", "Inbetriebnahme" oder "Diagnose" in der Oberfläche der Anweisung klicken, öffnet sich der jeweilige Editor.

1.1.3.6 SSI_Absolute_Encoder in Betrieb nehmen

Technologieobjekt in Betrieb nehmen

Der Inbetriebnahme-Editor hilft Ihnen mit einer grafische Darstellung des Bausteins bei der komfortablen Inbetriebnahme und dem Funktionstest des Technologieobjekts. Sie können bestimmte Parameter der Anweisung SSI_Absolute_Encoder im Online-Betrieb der CPU/IM ändern und deren Auswirkungen beobachten.

Voraussetzungen

- Es besteht eine Online-Verbindung zwischen STEP 7 (TIA Portal) und CPU.
- Die CPU ist im Betriebszustand RUN.
- Die zugehörige Anweisung SSI_Absolute_Encoder wird zyklisch aus dem Anwenderprogramm aufgerufen.
- Die Parameter des Technologieobjekts werden aus dem Anwenderprogramm nicht überschrieben.

Vorgehen

Um den Inbetriebnahme-Editor eines Technologieobjekts zu öffnen und eine Parameterwertänderung zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt SSI_Absolute_Encoder.
3. Doppelklicken Sie auf das Objekt "Inbetriebnahme".
   Die Funktionen zur Inbetriebnahme des Technologieobjekts SSI_Absolute_Encoder werden angezeigt.
4. Klicken Sie im Inbetriebnahmedialog oben links auf die Schaltfläche "Alle beobachten".
   Die Parameter (Online-Werte) des Technologieobjekts SSI_Absolute_Encoder werden geladen und angezeigt.
5. Wenn der Parameter, den Sie ändern wollen, ein Eingabefeld hat, geben Sie darin den neuen Wert ein.
6. Aktivieren Sie das Optionskästchen des Parameters.
   Der neue Parameterwert wird wirksam und die Auswirkungen der Änderung werden simuliert.

Online-Modus

Im Online-Modus können Sie folgende Parameter modifizieren und damit die Funktionsweise des Technologieobjekts testen:

- Neuer Vergleichswert 0 (NewReferenceValue0 oder NewReferenceValue0_M)
- Neuer Vergleichswert 1 (NewReferenceValue1 oder NewReferenceValue1_M)
- Capture freigeben (CaptureEnable)

- Bestätigung von gemeldeten Fehlerzuständen (ErrorAck)
- Rücksetzen der Statusflags (EventAck)

1.1.3.7 SSI_Absolute_Encoder Diagnose

Zählwerte, Messwerte, DIs und DQs beobachten

Mit den Diagnosefunktionen überwachen Sie die Positionserfassungs- und Messfunktionen.

Voraussetzungen

- Es besteht eine Online-Verbindung zwischen STEP 7 (TIA Portal) und CPU.
- Die CPU ist im Betriebszustand RUN.

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt SSI_Absolute_Encoder.
3. Doppelklicken Sie auf das Objekt "Diagnose".
4. Klicken Sie auf die Schaltfläche "Alle beobachten".

Anzeige

Folgende Werte werden vom Technologieobjekt aus der Rückmeldeschnittstelle gelesen und angezeigt:

- Ereignisanzeige/Diagnoseinformationen
- Signalzustände der Digitaleingänge und Digitalausgänge
- Positionswert
- Capture-Wert
- Messwert

Weitere Informationen zu den Statusanzeigen finden Sie in der Kontext-Hilfe zu jedem Ereignis in STEP 7 (TIA Portal). Wenn die CPU im STOP ist, wird die Statusanzeige nicht aktualisiert.

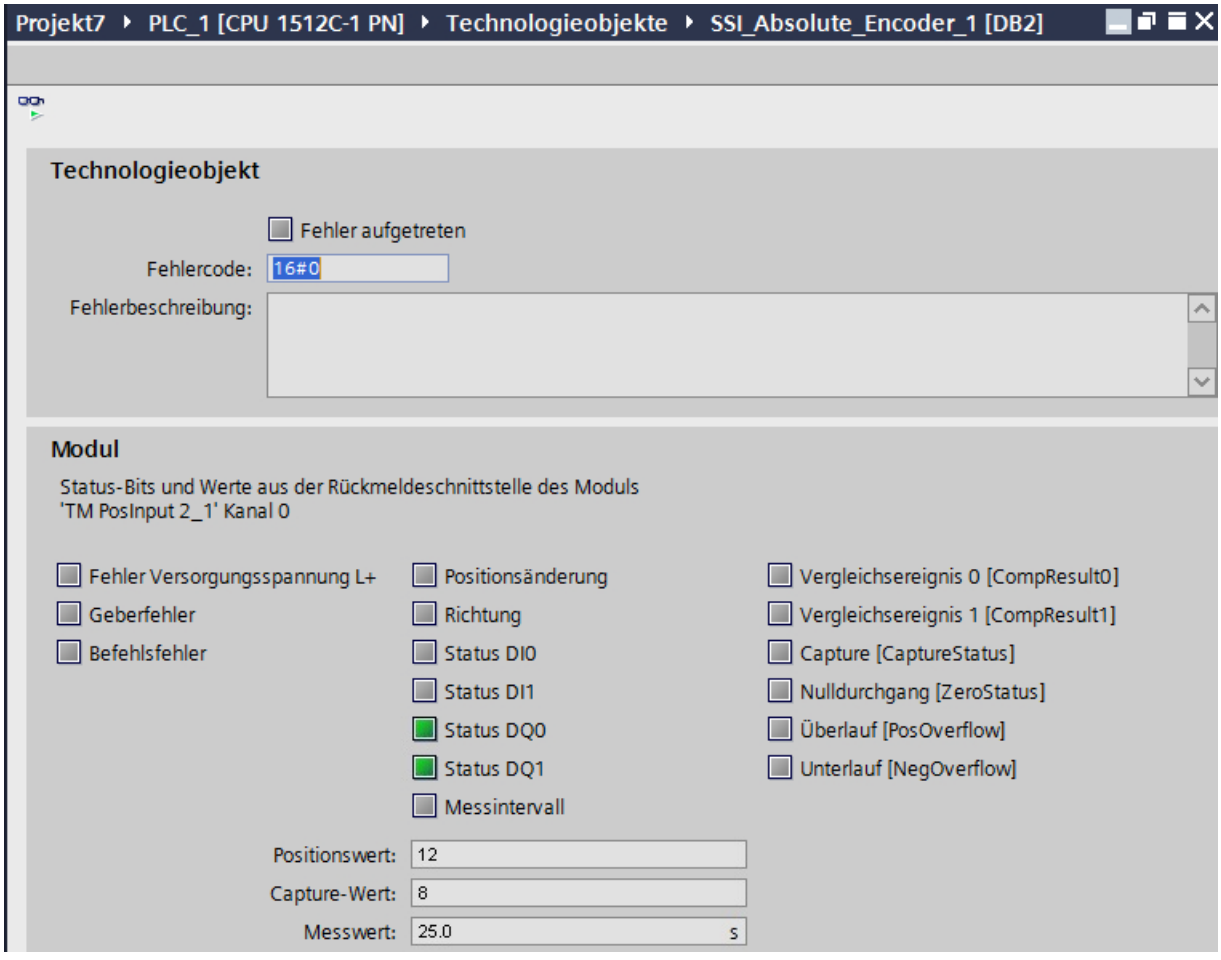

1.1.4 Modul einsetzen

1.1.4.1 Technologiemodul einsetzen

Konvention

Technologiemodul: Die Bezeichnung "Technologiemodul" verwenden wir im vorliegenden Kapitel sowohl für die Technologiemodule TM Count und TM PosInput als auch für den Technologieanteil der Kompakt-CPUs.

Modul konfigurieren

Technologiemodul zur Hardware-Konfiguration hinzufügen (TM Count und TM PosInput)

Voraussetzung

- Das Projekt ist angelegt.
- Die CPU S7-1500 ist angelegt.
- Bei dezentralem Betrieb ist eine Dezentrale Peripherie ET 200 angelegt.

Vorgehen

Um ein Technologiemodul zur Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie das Technologiemodul aus dem Baugruppenkatalog aus:
   "Technologiemodule > Zählen bzw. Wegerfassen > Technologiemodul > Artikelnummer"
4. Ziehen Sie das Technologiemodul per Drag & Drop auf den gewünschten Steckplatz im Baugruppenträger.

Technologiemodul zur Hardware-Konfiguration hinzufügen (Kompakt-CPU)

Voraussetzung

Das Projekt ist angelegt.

Vorgehen

Um eine Kompakt-CPU in der Projektnavigation hinzuzufügen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf "Neues Gerät hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
2. Wählen Sie "Controller".
3. Wählen Sie die Kompakt-CPU aus:
   "SIMATIC S7-1500 > CPU > Kompakt-CPU > Artikelnummer"
4. Bestätigen Sie mit "OK".

Ergebnis

In der Projektnavigation wird die neue Kompakt-CPU mit folgenden Objekten angezeigt. Über Doppelklick gelangen Sie zu dem gewünschten Editor.

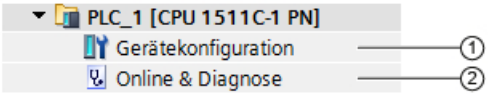

	Objekt	Beschreibung
①	Gerätekonfiguration (Seite 132)	Im Inspektorfenster (pro Kanal): • Aktivierung des Zählers (Seite 142) • Zuordnung Signale zu Ein- und Ausgängen (Seite 142) • Einstellung des Verhaltens bei CPU-STOP (Seite 132) • Freigabe Diagnosealarme (Seite 134) • Einstellung des Betriebsmodus (Seite 135) • Freigabe Prozessalarme (Seite 139) • Einstellung der Moduladressen
②	Online & Diagnose (Seite 169)	• Hardware diagnostizieren • Informationen über Kompakt-CPU erhalten • Firmware-Update durchführen

Hardware-Konfiguration öffnen

Öffnen

Um die Hardware-Konfiguration zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie auf das Modul.

Möglichkeiten der Parametrierung

Zählen, Messen und Positionserfassung mit SSI-Absolutwertgeber

Für die Zähl- und Messfunktionen sowie die Positionserfassung mit einem SSI-Absolutwertgeber haben Sie folgende alternative Möglichkeiten für die Parametrierung und Steuerung des Technologiemoduls:

- Konfiguration eines Technologieobjekts und Steuerung über die zugehörige Anweisung:
  Bei Verwendung eines Inkremental- oder Impulsgeber empfehlen wir die komfortable und grafikunterstützte Konfiguration über das Technologieobjekt High_Speed_Counter. Die ausführliche Beschreibung dieser Konfiguration finden Sie ab Abschnitt Technologieobjekt High_Speed_Counter.
  Bei Verwendung eines SSI-Absolutwertgeber empfehlen wir die komfortable und grafikunterstützte Konfiguration über das Technologieobjekt SSI_Absolute_Encoder. Die ausführliche Beschreibung dieser Konfiguration finden Sie ab Abschnitt Technologieobjekt SSI_Absolute_Encoder.
  Für die Konfiguration eines Technologieobjekts wählen Sie den Betriebsmodus (Seite 135) "Betrieb mit Technologieobjekt "Zählen und Messen"".
- Parametereinstellung über Hardware-Konfiguration und Steuerung über die Steuer- und Rückmeldeschnittstelle des Technologiemoduls:
  Hierfür wählen Sie den Betriebsmodus (Seite 135) "Manueller Betrieb (ohne Technologieobjekt)".
  Eine Beschreibung der Steuer- und Rückmeldeschnittstelle für TM Count und TM PosInput finden Sie in folgenden Abschnitten:
  Belegung der Steuerschnittstelle (Seite 170)
  Belegung der Rückmeldeschnittstelle (Seite 172)
- Parametereinstellung über GSD-Datei und Steuerung über die Steuer- und Rückmeldeschnittstelle des Technologiemoduls:
  Eine Beschreibung der Steuer- und Rückmeldeschnittstelle für TM Count und TM PosInput finden Sie in folgenden Abschnitten:
  Belegung der Steuerschnittstelle (Seite 170)
  Belegung der Rückmeldeschnittstelle (Seite 172)

Positionserfassung für Motion Control

Sie haben alternativ die Möglichkeit, das Technologiemodul zur Positionserfassung für Motion Control zu verwenden.

Hierfür wählen Sie den Betriebsmodus (Seite 135) "Positionserfassung für Technologieobjekt "Motion Control"" und nehmen die Parametrierung des Gebers über die Modulparameter (Seite 136) in der Gerätekonfiguration des Technologiemoduls vor. Die weitere Projektierung dieser Anwendung nehmen Sie über ein Achs-Technologieobjekt von S7-1500 Motion Control vor.

Grundparameter

Verhalten bei CPU-STOP

Verhalten bei CPU-STOP

Das Verhalten des Technologiemoduls bei einem STOP der CPU stellen Sie kanalweise bei den Grundparametern in der Gerätekonfiguration ein.

Option	Bedeutung
Weiterarbeiten	Das Technologiemodul arbeitet voll funktionsfähig weiter. Eingehende Zählimpulse werden verarbeitet bzw. die Ist-Position wird eingelesen. Die Digitalausgänge schalten weiterhin entsprechend der Parametrierung.
Ersatzwert ausgeben	Das Technologiemodul gibt bis zum nächsten STOP-RUN-Übergang der CPU an den Digitalausgängen die parametrierten Ersatzwerte aus. Nach einem STOP-RUN-Übergang wird das Technologiemodul in seinen Anlaufzustand gesetzt: Der Zählwert wird auf den Startwert gesetzt (bei Inkremental- oder Impulsgeber) und die Digitalausgänge schalten entsprechend der Parametrierung.
Letzten Wert halten	Das Technologiemodul gibt bis zum nächsten STOP-RUN-Übergang der CPU an den Digitalausgängen die Werte aus, die zum Zeitpunkt des Übergangs nach STOP gültig waren. Wenn ein Digitalausgang mit der Funktion "Bei Vergleichswert für eine Impulsdauer" bei CPU-STOP gesetzt ist, wird der Digitalausgang nach Ablauf der Impulsdauer rückgesetzt. Nach einem STOP-RUN-Übergang wird das Technologiemodul in seinen Anlaufzustand gesetzt: Der Zählwert wird auf den Startwert gesetzt (bei Inkremental- oder Impulsgeber) und die Digitalausgänge schalten entsprechend der Parametrierung.

Ersatzwert für DQ0 (TM Count und TM PosInput)

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem STOP der CPU am Digitalausgang DQ0 ausgeben soll.

Hinweis

Im Betriebsmodus „Betrieb mit Technologieobjekt "Zählen und Messen"“ legen Sie diesen Parameter über das Technologieobjekt fest.

Ersatzwert für DQ0 (Kompakt-CPU)

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welcher Wert bei einem STOP der Kompakt-CPU in der Rückmeldeschnittstelle für DQ0 ausgegeben wird.

Hinweis

Im Betriebsmodus „Betrieb mit Technologieobjekt "Zählen und Messen"“ legen Sie diesen Parameter über das Technologieobjekt fest.

Ersatzwert für DQ1

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem STOP der CPU am Digitalausgang DQ1 ausgeben soll.

Hinweis

Im Betriebsmodus „Betrieb mit Technologieobjekt "Zählen und Messen"“ legen Sie diesen Parameter über das Technologieobjekt fest.

Diagnosealarme (TM Count und TM PosInput)

Das Technologiemodul kann zusätzliche Diagnosealarme auslösen, wenn Sie die Freigabe der Diagnosealarme bei den Grundparametern aktivieren. Diagnosealarme bearbeiten Sie in einem Alarm-OB.

Detaillierte Informationen zum Fehlerereignis erhalten Sie im Fehler-Organisationsbaustein mit der Anweisung "RALRM" (Alarmzusatzinfo lesen) und im Funktionshandbuch Diagnose (https://support.industry.siemens.com/cs/ww/de/view/59192926) im Kapitel "Systemdiagnose über Anwenderprogramm".

Freigabe Diagnosealarm bei Drahtbruch

Mit diesem Parameter legen Sie für die folgenden Geber fest, ob bei einem Drahtbruch der verwendeten Signale ein Diagnosealarm ausgelöst wird:

- Gegentakt-schaltende 24 V-Geber (Seite 62)
- RS422-Geber (Seite 64) (zusätzlich auf Kurzschluss und fehlerhafte Versorgungsspannung überwacht)
- SSI-Absolutwertgeber (Seite 65) (zusätzlich auf Kurzschluss und fehlerhafte Versorgungsspannung überwacht)

Hinweis

Wenn Sie einen Geber mit einem anderen Sensortyp bzw. Schnittstellenstandard verwenden, kann ein Drahtbruch nicht erkannt werden.

Freigabe weitere Diagnosealarme

Mit diesem Parameter legen Sie fest, ob bei weiteren Fehlern Diagnosealarme ausgelöst werden.

Informieren Sie sich im Gerätehandbuch zum Technologiemodul, welche Fehler im Betrieb einen Diagnosealarm auslösen können.

Diagnosealarme (Kompakt-CPU)

Diagnosealarme freigeben

Eine Kompakt-CPU kann bei bestimmten Fehlern Diagnosealarme auslösen, wenn Sie die Freigabe der Diagnosealarme aktivieren. Diagnosealarme bearbeiten Sie in einem Alarm-OB.

Mit diesem Parameter legen Sie fest, ob die Kompakt-CPU die Diagnosealarme bei Auftreten der jeweiligen Fehler auslöst.

Informieren Sie sich im Gerätehandbuch zur Kompakt-CPU, welche Fehler im Betrieb einen Diagnosealarm auslösen können. In der Voreinstellung sind die Diagnosealarme nicht freigegeben.

Betriebsmodus

Auswahl des Betriebsmodus für den Kanal

Mit dieser Einstellung legen Sie fest, wie die Parametrierung und Steuerung der Zähl- und Messfunktionen des Kanals erfolgen soll.

Betriebsmodus	Beschreibung
Betrieb mit Technologieobjekt "Zählen und Messen"	Die Parametrierung des Kanals erfolgt über ein Technologieobjekt. Den Zugriff auf die Steuer- und Rückmeldeschnittstelle des Technologiemoduls übernimmt die zugehörige Anweisung zum Technologieobjekt im Anwenderprogramm. Die Zuordnung zwischen Technologiemodul/Kanal und Technologieobjekt legen Sie bei den Grundparametern des Technologieobjekts fest.
Positionserfassung für Technologieobjekt "Motion Control"	Das Technologiemodul dient als Positionserfassung für eine übergeordnete Motion Control-Steuerung. Bei TM Count und TM PosInput wirkt dieser Betriebsmodus auf alle Kanäle des Technologiemoduls. Bei einer Kompakt-CPU wirkt die Einstellung des Betriebsmodus auf den jeweiligen Kanal. Die Parametrierung erfolgt über die Gerätekonfiguration des Technologiemoduls. Die Parametrierung der Gebersignale erfolgt dabei über die Modulparameter (Seite 136).
Manueller Betrieb (ohne Technologieobjekt)	Die Parametrierung des Kanals erfolgt über die Hardware-Konfiguration des Technologiemoduls: • Manueller Betrieb (Inkremental- oder Impulsgeber) (Seite 143) • Manueller Betrieb (SSI-Absolutwertgeber) (Seite 159) Sie haben über das Anwenderprogramm direkten Zugriff auf die Steuer- und Rückmeldeschnittstelle des Kanals.
Fast Mode	Das Technologiemodul dient einer sehr schnellen Erfassung des Zähl- oder Positionswerts. Es steht keine Steuerschnittstelle zur Verfügung. Die Parametrierung des Kanals erfolgt über die Parametereinstellung (Hardware-Konfiguration) des Moduls. Sie haben über das Anwenderprogramm direkten Zugriff auf die Rückmeldeschnittstelle des Kanals. Dieser Betriebsmodus wird nicht von jedem Technologiemodul unterstützt.

Hinweis

GSD-Datei

Bei Verwendung einer GSD-Datei bestimmen Sie den Betriebsmodus über die Auswahl des Modulnamens im Hardware-Katalog. Sie können dabei manuellen Betrieb oder Fast Mode wählen.

Auswahl der Betriebsart für den Kanal

Bei "Manueller Betrieb" legen Sie mit diesem Parameter fest, für welche Hauptaufgabe der Kanal des Technologiemoduls eingesetzt wird. Abhängig davon werden die Einstellmöglichkeiten unter "Parameter" (Hardware-Konfiguration) angepasst.

Betriebsart	Beschreibung
Zählen (Seite 20) / Positionserfassung (Seite 22)	Die Hauptaufgabe des Kanals ist Zählen bzw. Positionserfassung. Die Vergleichsfunktionen und Prozessalarme arbeiten mit dem Zählwert bzw. Positionswert. Der Messwert steht parallel zur Verfügung.
Messen (Seite 51)	Die Hauptaufgabe des Kanals ist Messen. Die Vergleichsfunktionen (Seite 49) und die Prozessalarme für Vergleichsereignisse arbeiten mit dem Messwert. Der Zählwert steht parallel zur Verfügung.

Hinweis

GSD-Datei

Bei Verwendung einer GSD-Datei bestimmen Sie die Betriebsart über die Auswahl des Modulnamens im Hardware-Katalog.

Modulparameter (Positionserfassung für Motion Control)

Im Betriebsmodus "Positionserfassung für Technologieobjekt "Motion Control"" stellen Sie unter "Modulparameter" die Parameter für die Gebersignale des Kanals ein. Die Parameter hängen vom verwendeten Geber ab.

Hinweis

Diesen Betriebsmodus können Sie nicht mit einer GSD-Datei verwenden.

Modulparameter für Inkrementalgeber und Impulsgeber

Wenn Sie einen Inkrementalgeber oder Impulsgeber verwenden, stellen Sie folgende Parameter für die Gebersignale des Kanals ein.

- Signalart
- Richtung invertieren
- Signalauswertung
- Filterfrequenz
- Sensortyp (bei TM Count)
- Schnittstellenstandard (bei TM PosInput)
- Signalauswahl für Referenzmarke 0
- Messeingang
- Inkremente pro Umdrehung

- Schritte pro Umdrehung
- Bezugsdrehzahl

Erläuterungen zu den ersten sechs Parametern finden Sie im Abschnitt Zähleingänge (Seite 143).

Modulparameter für SSI-Absolutwertgeber

Wenn Sie ein TM PosInput mit einem SSI-Absolutwertgeber verwenden, stellen Sie folgende Parameter für die Gebersignale des Kanals ein.

- Signalart
- Richtung invertieren
- Telegrammlänge
- Codeart
- Übertragungsgeschwindigkeit
- Monoflopzeit
- Parität
- Bit-Nummer LSB des Positionswerts
- Bit-Nummer MSB des Positionswerts
- Messeingang
- Schritte pro Umdrehung
- Anzahl der Umdrehungen
- Bezugsdrehzahl

Erläuterungen zu den ersten neun Parametern finden Sie im Abschnitt Zähleingänge (Seite 159).

Signalauswahl für Referenzmarke 0

Mit diesem Parameter legen Sie fest, bei welchem externen Referenzsignal eine neue Referenzmarke für die Geberposition gespeichert wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Keine[1]	Es wird kein externes Referenzsignal verwendet.
DI0 (voreingestellt bei TM Count und TM PosInput)	Bei einer steigenden Flanke des Digitaleingangs DI0 wird der aktuelle Zählwert als neue Referenzmarke für die Geberposition gespeichert.
Signal N des Inkrementalgebers (voreingestellt bei Kompakt-CPU[2])	Bei einer steigenden Flanke des Signals N des Inkrementalgebers wird der aktuelle Zählwert als neue Referenzmarke für die Geberposition gespeichert.

[1] Nur bei Kompakt-CPU verfügbar

[2] Ausnahmen: „DI0“ voreingestellt bei HSC 3 und HSC 6 auf der 1511C sowie auf der 1512C im Kompatibilitätsmodus

Messeingang (TM Count und TM PosInput)

Der Parameter gibt an, dass der Hardware-Digitaleingang DI1 als Messtaster-Eingang dient. Der aktuelle Zählwert oder Positionswert wird bei steigender, fallender oder beiden Flanken des DI1 als aktuelle Geberposition gespeichert. Der Wert des Parameters ist nicht änderbar.

Messeingang (Kompakt-CPU)

Mit diesem Parameter legen Sie den Hardware-Eingang fest, der als externer Messeingang zum Speichern der Geberposition verwendet wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Keine (voreingestellt)	Es wird kein externer Messeingang verwendet.
DI1	Der Hardware-Digitaleingang DI1 dient als Messtaster-Eingang. Der aktuelle Zählwert wird bei steigender, fallender oder beiden Flanken des DI1 als aktuelle Geberposition gespeichert.

Inkremente pro Umdrehung

Mit diesem Parameter geben Sie die Strichzahl des Inkremental- oder Impulsgebers an. Die Strichzahl entnehmen Sie dem Datenblatt des Gebers.

Schritte pro Umdrehung (Inkremental- oder Impulsgeber)

Der Parameter gibt die Anzahl der Zählimpulse pro Geberumdrehung an.

Die Anzahl der Zählimpulse ist abhängig von der Inkrementzahl pro Umdrehung und der parametrierten Signalauswertung.

Beispiel:

Ihr Inkremental- oder Impulsgeber liefert 2048 Inkremente pro Umdrehung. Je nach Signalauswertung wird folgender Wert angezeigt:

Signalauswertung	Schritte pro Umdrehung
Einfach	2048
Zweifach	4096
Vierfach	8192

Schritte pro Umdrehung (SSI-Absolutwertgeber)

Mit diesem Parameter legen Sie die Anzahl der Inkremente fest, die der SSI-Absolutwertgeber pro Umdrehung liefert.

Anzahl der Umdrehungen

Der Parameter gibt an, wie viele Umdrehungen der Wertebereich des SSI-Absolutwertgebers umfasst.

Der Wert wird berechnet aus den parametrierten Werten für die Bit-Nummern LSB und MSB des Positionswerts sowie der Schritte pro Umdrehung.

Bezugsdrehzahl

Der Geber überträgt den Istwert der Drehzahl als Prozentwert bezogen auf die Bezugsdrehzahl. Mit diesem Parameter legen Sie die Drehzahl in U/min fest, die dem Wert 100 % entsprechen soll. Die Bezugsdrehzahl muss mit der Einstellung in der Steuerung identisch sein.

Zulässig ist ein Wert von 6,00 bis 210000,00. Die Voreinstellung ist "3000,00".

Weitere Projektierung

Die weitere Projektierung nehmen Sie über ein Achs- und ein Messtaster-Technologieobjekt von S7-1500 Motion Control vor. Bitte Informieren Sie sich in den Funktionshandbüchern S7-1500 Motion Control (https://support.industry.siemens.com/cs/ww/de/view/59381279) und S7-1500T Motion Control (https://support.industry.siemens.com/cs/ww/de/view/109481326) über die weitere Projektierung und Inbetriebnahme für die Positionserfassung und für das Technologieobjekt Messtaster.

Siehe auch

Positionserfassung für Motion Control (Seite 61)

Prozessalarme

Sie können bei den Grundparametern des Technologiemoduls pro Kanal einstellen, welche Ereignisse im Betrieb einen Prozessalarm auslösen sollen.

In einem S7-1500-System geben Sie für jeden freigegebenen Prozessalarm einen passenden Ereignisnamen ein und ordnen jedem Prozessalarm einen entsprechenden Prozessalarm-OB

zu. Wenn ein Prozessalarm ausgelöst wird, wird der jeweilige OB zur Auswertung der Prozessalarmdaten gestartet.

Hinweis

In folgenden Fällen können Sie pro Modul nur *einen* Prozessalarm freigeben:

- Dezentraler Betrieb mit einer CPU S7-300/400
- Verwendung einer GSD-Datei

Ein Prozessalarm wird ausgelöst, wenn die Bedingung für die Änderung des jeweiligen Status- oder Ereignis-Bits in der Rückmeldeschnittstelle erfüllt ist.

Verlorener Prozessalarm

Wenn die Prozessalarme schneller ausgelöst werden, als sie vom System quittiert werden können, gehen Prozessalarme verloren und der Diagnosealarm "Prozessalarm verloren" wird gemeldet.

Aktivierbare Prozessalarme

Prozessalarm	Verfügbar in Betriebsart Zählen bei Verwendung von		Verfügbar in Betriebsart Messen bei Verwendung von		Verfügbar in Betriebsmodus Positionserfassung für Technologieobjekt "Motion Control"	Beschreibung	Event Type-Nummer
	Inkremental- oder Impulsgeber	SSI-Absolutwertgeber	Inkremental- oder Impulsgeber	SSI-Absolutwertgeber			
Neuer Capture-Wert vorhanden	Ja	Ja	Nein	Nein	Nein	Prozessalarm bei Speichern des aktuellen Zählwerts bzw. Positionswerts als Capture-Wert	1000_B
Synchronisation des Zählers durch externes Signal	Ja	Nein	Ja	Nein	Nein	Prozessalarm bei Synchronisation des Zählers durch Signal N oder DI-Flanke	1001_B
Torstart	Ja	Nein	Ja	Nein	Nein	Prozessalarm bei Öffnen des internen Tors	0001_B
Torstopp	Ja	Nein	Ja	Nein	Nein	Prozessalarm bei Schließen des internen Tors	0010_B
Überlauf (obere Zählgrenze überschritten)	Bezogen auf Zählwert	Nein	Bezogen auf Zählwert	Nein	Nein	Prozessalarm, wenn der Zählwert die obere Zählgrenze überschreitet	0011_B
Unterlauf (untere Zählgrenze unterschritten)	Bezogen auf Zählwert	Nein	Bezogen auf Zählwert	Nein	Nein	Prozessalarm, wenn der Zählwert die untere Zählgrenze unterschreitet	0100_B
Richtungsumkehr[1]	Bezogen auf Zählwert	Bezogen auf Positionswert	Bezogen auf Zählwert	Bezogen auf Positionswert	Nein	Prozessalarm bei Richtungsumkehr des Zählwerts bzw. Positionswerts	1010_B

Prozessalarm	Verfügbar in Betriebsart Zählen bei Verwendung von		Verfügbar in Betriebsart Messen bei Verwendung von		Verfügbar in Betriebsmodus Positionserfassung für Technologieobjekt "Motion Control"	Beschreibung	Event Type-Nummer
	Inkremental- oder Impulsgeber	SSI-Absolutwertgeber	Inkremental- oder Impulsgeber	SSI-Absolutwertgeber			
Nulldurchgang	Bezogen auf Zählwert	Bezogen auf Positionswert	Bezogen auf Zählwert	Bezogen auf Positionswert	Nein	Prozessalarm bei Nulldurchgang des Zählwerts bzw. Positionswerts	0111_B
Vergleichsereignis für DQ0 eingetreten	Bezogen auf Zählwert	Bezogen auf Positionswert	Bezogen auf Messwert	Bezogen auf Messwert	Nein	Prozessalarm, wenn ein Vergleichsereignis für DQ0 aufgrund der gewählten Vergleichsbedingung eintritt; Kein Prozessalarm, wenn bei einem Inkremental- oder Impulsgeber die Änderung des Zählwerts nicht durch einen Zählimpuls verursacht wurde	0101_B
Vergleichsereignis für DQ1 eingetreten	Bezogen auf Zählwert	Bezogen auf Positionswert	Bezogen auf Messwert	Bezogen auf Messwert	Nein	Prozessalarm, wenn ein Vergleichsereignis für DQ1 aufgrund der gewählten Vergleichsbedingung eintritt; Kein Prozessalarm, wenn bei einem Inkremental- oder Impulsgeber die Änderung des Zählwerts nicht durch einen Zählimpuls verursacht wurde	0110_B

[1] Das Rückmeldebit STS_DIR ist mit "0" vorbelegt. Wenn die erste Zählwert- oder Positionswertänderung direkt nach Einschalten des Technologiemoduls in Rückwärtsrichtung erfolgt, wird kein Prozessalarm ausgelöst.

Voreinstellung

In der Voreinstellung sind keine Prozessalarme freigegeben.

Zusätzliche Parameter für Kompakt-CPU

Einleitung

Bei Verwendung einer Kompakt-CPU stehen Ihnen für die Signale der schnellen Zähler zusätzlich die nachfolgenden Parameter zur Verfügung.

Kompatibilität 1511C (Schnelle Zähler der Kompakt-CPU 1512C-1 PN)

Frontstecker-Belegung wie CPU 1511C

Mit diesem Parameter legen Sie fest, ob für die schnellen Zähler der CPU 1512C-1 PN die Anschlussbelegung der Frontstecker der CPU 1511C-1 PN verwendet wird:

Option	Bedeutung
Deaktiviert (voreingestellt)	Die CPU 1512C-1 PN verwendet die Anschlussbelegung der integrierten Frontstecker. 1512C-1 PN unterstützt die Nutzung von Anschlüssen beider Frontstecker der digitalen Onboard-Peripherie für die schnellen Zähler. Die Zuweisung von Hardwareeingängen und -ausgängen für die HSC-Kanäle ist im Gerätehandbuch der CPU 1512C-1 PN beschrieben.
Aktiviert	Die CPU 1512C-1 PN verwendet die Anschlussbelegung der Frontstecker der CPU 1511C-1 PN. 1511C-1 PN unterstützt die Nutzung von Anschlüssen des ersten Frontsteckers der digitalen Onboard-Peripherie für die schnellen Zähler. Die Zuweisung von Hardwareeingängen und -ausgängen für die HSC-Kanäle ist im Gerätehandbuch der CPU 1511C-1 PN beschrieben.

Allgemein

Diesen schnellen Zähler aktivieren

Mit diesem Parameter legen Sie fest, ob der jeweilige schnelle Zähler verwendet wird:

Option	Bedeutung
Deaktiviert (voreingestellt)	Der schnelle Zähler wird nicht verwendet. Der Zähler belegt keine Anschlüsse der integrierten Frontstecker und kann keine Alarme auslösen. Schreibvorgänge auf seine Steuerschnittstelle werden ignoriert und seine Rückmeldeschnittstelle liefert ausschließlich Nullwerte.
Aktiviert	Der schnelle Zähler wird verwendet. Die Zuweisung von HSC-Adressen zu Anschlüssen der integrierten Frontstecker ist im Gerätehandbuch der Kompakt-CPU beschrieben.

Hardwareein-/ausgänge

Eingang Taktgeber (A) / Impulseingang (A) / Taktgeber vorwärts (A)

Dieser Parameter gibt für den jeweiligen Zähler an, welcher Eingang für das Gebersignal A verwendet wird. Der Wert ist nicht änderbar.

Eingang Taktgeber (B) / Impulseingang (B) / Taktgeber vorwärts (B)

Wenn Sie für den jeweiligen Zähler einen Geber mit mehreren Signalen verwenden, gibt der Parameter an, welcher Eingang für das Gebersignal B verwendet wird. Der Wert ist nicht änderbar.

Rücksetzeingang (N)

Wenn Sie für den jeweiligen Zähler einen Inkrementalgeber verwenden, gibt der Parameter an, welcher Eingang als Rücksetzeingang (Gebersignal N) verwendet wird. Der Wert ist nicht änderbar.

HSC DI0 / HSC DI1

Mit diesem Parameter legen Sie fest, welcher Digitaleingang der Kompakt-CPU als DIm des Zählers verwendet werden soll.

Hinweis

Die Eingangsverzögerung für einen Digitaleingang parametrieren Sie im Inspektorfenster der Gerätekonfiguration unter "Eigenschaften > DI 16/DQ 16 > Eingänge > Kanal n".

HSC DQ0

Den Zustand des DQ0 lesen Sie über die Rückmeldeschnittstelle. Sie können DQ0 nicht einem physikalischen Digitalausgang der Kompakt-CPU zuweisen.

HSC DQ1

Mit diesem Parameter legen Sie fest, welcher Digitalausgang der Kompakt-CPU als DQ1 verwendet werden soll. Sie können dabei einen Ausgang mit einer Ausgangsverzögerung von 5 µs oder 500 µs auswählen.

Eine Übersicht der Ausgangsverzögerung für alle Digitalausgänge finden Sie im Gerätehandbuch zur Kompakt-CPU.

Manueller Betrieb (Inkremental- oder Impulsgeber)

Zähleingänge

Signalart

Sie können unter folgenden Signalarten (Seite 62) auswählen:

Signalart	Bedeutung	Weitere optionsspezifische Parameter
Inkrementalgeber (A, B phasenversetzt)	Ein Inkrementalgeber mit den phasenversetzten Signalen A und B ist angeschlossen.	• Richtung invertieren • Signalauswertung • Filterfrequenz • Sensortyp oder Schnittstellenstandard
Inkrementalgeber (A, B, N)	Ein Inkrementalgeber mit den zueinander phasenversetzten Signalen A, B sowie einem Nullsignal N ist angeschlossen.	• Richtung invertieren • Signalauswertung • Filterfrequenz • Sensortyp oder Schnittstellenstandard • Verhalten bei Signal N • Häufigkeit der Synchronisation • Häufigkeit der Capture-Funktion
Impuls (A) und Richtung (B)	Ein Impulsgeber (Signal A) mit Richtungssignal (Signal B) ist angeschlossen.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard
Impuls (A)	Ein Impulsgeber (Signal A) ohne Richtungssignal ist angeschlossen. Sie können die Zählrichtung über die Steuerschnittstelle (Seite 170) vorgeben.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard
Vorwärts zählen (A), rückwärts zählen (B)	Signale zum Zählen in Vorwärtsrichtung (Signal A) und Rückwärtsrichtung (Signal B) sind angeschlossen.	• Filterfrequenz • Sensortyp oder Schnittstellenstandard

Richtung invertieren

Sie können die Zählrichtung invertieren, um sie an den Prozess anzupassen.

Das Invertieren der Richtung ist bei den folgenden Signalarten parametrierbar und wirksam:

- Inkrementalgeber (A, B phasenversetzt)
- Inkrementalgeber (A, B, N)

Signalauswertung

Mit der Parametrierung der Signalauswertung (Seite 66) legen Sie fest, welche Flanken der Signale gezählt werden.

Sie können unter folgenden Optionen auswählen:

Signalauswertung	Bedeutung
Einfach (voreingestellt)	Die Flanken des Signals A während eines Low-Pegels des Signals B werden ausgewertet.
Zweifach	Jede Flanke des Signals A wird ausgewertet.
Vierfach	Jede Flanke der Signale A und B wird ausgewertet.

Der Parameter ist bei den folgenden Signalarten parametrierbar:

- Inkrementalgeber (A, B phasenversetzt)
- Inkrementalgeber (A, B, N)

Filterfrequenz

Mit der Parametrierung der Filterfrequenz unterdrücken Sie Störungen an den Zähleingängen A, B und N.

Die ausgewählte Filterfrequenz bezieht sich auf ein Impuls-Pausen-Verhältnis zwischen ca. 40:60 und ca. 60:40. Dadurch ergibt sich eine bestimmte minimale Impuls-/Pausendauer. Signalwechsel mit einer Dauer kürzer als die minimale Impuls-/Pausendauer werden unterdrückt.

Sie können unter folgenden Filterfrequenzen auswählen:

Filterfrequenz	Minimale Impuls-/Pausendauer
100 Hz	4,0 ms
200 Hz	2,0 ms
500 Hz	800 µs
1 kHz	400 µs
2 kHz	200 µs
5 kHz	80 µs
10 kHz	40 µs
20 kHz	20 µs
50 kHz	8,0 µs
100 kHz (voreingestellt bei Kompakt-CPU)	4,0 µs
200 kHz** (voreingestellt bei TM Count)	2,0 µs
500 kHz*	0,8 µs
1 MHz* (voreingestellt bei TM PosInput)	0,4 µs

* Nur bei TM PosInput verfügbar

** Nur bei TM Count und TM PosInput verfügbar

Sensortyp (TM Count)

Mit der Parametrierung des Sensortyps legen Sie für das TM Count fest, wie die Zähleingänge geschaltet werden.

Sie können unter folgenden Optionen auswählen:

Sensortyp	Bedeutung
P-Schalter (voreingestellt)	Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach 24VDC.
M-Schalter	Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach M.
Gegentakt (M- und P-Schalter)	Der Geber bzw. Sensor schaltet die Eingänge A, B und N wechselweise nach M und 24VDC.

Bei Verwendung von Inkrementalgebern wird typischerweise "Gegentakt" gewählt. Bei Verwendung von 2-Draht-Sensoren, z. B. Lichtschranken oder Näherungsschalter, muss entsprechend der Verdrahtung "P-Schalter" oder "M-Schalter" gewählt werden.

Ob Ihr Inkrementalgeber Gegentakt-schaltend ist, entnehmen Sie dem Datenblatt des Gebers.

Hinweis

Wenn Sie einen Gegentakt-schaltenden Geber verwenden und der Sensortyp "Gegentakt (M- und P-Schalter)" parametriert ist, können Sie die Gebersignale auf Drahtbruch überwachen.

Sensortyp (Kompakt-CPU)

Für eine Kompakt-CPU ist der Sensortyp "P-Schalter" eingestellt und nicht änderbar. Der Geber bzw. Sensor schaltet die Eingänge A, B und N nach 24VDC.

Sie können an der Kompakt-CPU sowohl P-schaltende als auch Gegentakt-schaltende Geber betreiben. Informationen zum Sensortyp entnehmen Sie dem Datenblatt des Gebers.

Schnittstellenstandard (TM PosInput)

Mit diesem Parameter legen Sie für das TM PosInput fest, ob der Geber symmetrische (RS422) oder asymmetrische Signale (TTL) liefert.

Sie können unter folgenden Optionen auswählen:

Schnittstellenstandard	Bedeutung
RS422, symmetrisch (voreingestellt)	Der Geber liefert symmetrische Signale gemäß dem RS422-Standard (Seite 64).
TTL (5 V), asymmetrisch	Der Geber liefert asymmetrische 5 V-Signale gemäß dem TTL-Standard (Seite 62).

Hinweis

Der RS422-Standard bietet eine höhere Störfestigkeit als der TTL-Standard. Wenn Ihr Inkremental- oder Impulsgeber den RS422- **und** den TTL-Standard beherrscht, wird deshalb der RS422-Standard empfohlen.

Verhalten bei Signal N

Mit diesem Parameter legen Sie fest, welches Verhalten bei Signal N ausgelöst wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Keine Reaktion auf Signal N (voreingestellt)	Der Zähler wird durch das Signal N nicht beeinflusst.
Synchronisation bei Signal N (Seite 37)	Der Zähler wird bei Signal N auf den Startwert gesetzt. Wenn Sie für einen Digitaleingang die Funktion "Freigabe Synchronisation bei Signal N" wählen, ist die Synchronisation vom Pegel am Digitaleingang abhängig.
Capture bei Signal N (Seite 28)	Der Zählwert wird bei Signal N in den Capture-Wert gespeichert. Die Verwendung eines Digitaleingangs und die Verwendung des Signals N schliessen sich für die Capture-Funktion nicht gegenseitig aus. Das Technologieobjekt zeigt den Capture-Wert am Ausgangsparameter CapturedValue an.

Hinweis

Sie können das Verhalten bei Signal N nur auswählen, wenn Sie die Signalart "Inkrementalgeber (A, B, N)" gewählt haben.

Hinweis

Wenn Sie " Synchronisation bei Signal N" auswählen, können Sie für einen Digitaleingang (Seite 149) die Funktion "Freigabe Synchronisation bei Signal N" wählen.

Hinweis

Für High_Speed_Counter ab V3.0 gilt:

"Capture bei Signal N" können Sie nur in der Betriebsart "Zählwert als Bezug verwenden" wählen.

Häufigkeit der Synchronisation

Mit diesem Parameter legen Sie die Häufigkeit folgender Ereignisse fest:

- Synchronisation bei Signal N
- Synchronisation als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig (voreingestellt)	Der Zähler wird nur bei dem ersten Signal N bzw. der ersten parametrierten Flanke des Digitaleingangs gesetzt.
Periodisch	Der Zähler wird bei jedem Signal N bzw. jeder parametrierten Flanke des Digitaleingangs gesetzt.

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie für folgende Funktionen die Häufigkeit von Capture-Ereignissen fest:

- Capture bei Signal N
- Capture als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang bzw. erste steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang bzw. jede steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.

Zählgrenzen und Startwert

Obere Zählgrenze

Mit der Parametrierung der oberen Zählgrenze begrenzen Sie den Zählbereich. Sie können einen Wert bis 2147483647 (2^{31}-1) eingeben. Sie müssen einen Wert eingeben, der über der unteren Zählgrenze liegt.

Die Voreinstellung ist "2147483647".

Untere Zählgrenze

Mit der Parametrierung der unteren Zählgrenze begrenzen Sie den Zählbereich. Sie können einen Wert bis -2147483648 (-2^{31}) eingeben. Sie müssen einen Wert eingeben, der unter der oberen Zählgrenze liegt.

Die Voreinstellung ist "-2147483648".

Startwert

Mit der Parametrierung des Startwerts legen Sie fest, mit welchem Wert das Zählen begonnen und bei definierten Ereignissen weitergezählt wird. Sie müssen einen Wert zwischen den Zählgrenzen oder auf den Zählgrenzen eingeben.

Die Voreinstellung ist "0".

Weitere Informationen

Zusätzliche Informationen finden Sie unter Verhalten an den Zählgrenzen (Seite 23) und Zählerverhalten bei Torstart (Seite 27).

Zählerverhalten an den Grenzen und bei Torstart

Verhalten bei Überschreiten einer Zählgrenze

Sie können folgendes Verhalten bei Überschreiten einer Zählgrenze (Seite 23) parametrieren:

Verhalten	Bedeutung
Zählen stoppen	Nach dem Überschreiten einer Zählgrenze wird der Zählvorgang abgebrochen und das interne Tor geschlossen. Zum erneuten Starten des Zählvorgangs müssen Sie das SW-Tor oder HW-Tor gegebenenfalls schließen und erneut öffnen.
Zählen fortsetzen (voreingestellt)	Der Zählvorgang wird in Abhängigkeit der weiteren Parametrierung entweder mit dem Startwert oder an der jeweils anderen Zählgrenze fortgesetzt.

Rücksetzen bei Überschreiten einer Zählgrenze

Sie können den Zähler bei Überschreiten einer Zählgrenze auf folgende Werte setzen:

Rücksetzen des Werts	Bedeutung
Auf Startwert	Der Zählwert wird auf den Startwert gesetzt.
Auf andere Zählgrenze (voreingestellt)	Der Zählwert wird auf die jeweils andere Zählgrenze gesetzt.

Verhalten bei Torstart

Sie können folgendes Verhalten bei Torstart (Seite 27) parametrieren:

Verhalten	Bedeutung
Setzen auf Startwert	Bei Öffnen des Tors wird der Zählwert auf den Startwert gesetzt.
Fortsetzen mit aktuellem Wert (voreingestellt)	Bei Öffnen des Tors wird mit dem letzten Zählwert weitergezählt.

Verhalten eines DI

Funktion des DI einstellen

Mit der Parametrierung eines Digitaleingangs legen Sie fest, welche Funktion der Digitaleingang beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Torstart/-stopp (pegelgesteuert)	Der Pegel am jeweiligen Digitaleingang öffnet und schließt das HW-Tor (Seite 26).	• Eingangsverzögerung • Pegelauswahl
Torstart (flankengesteuert)	Die parametrierte Flanke am jeweiligen Digitaleingang öffnet das HW-Tor (Seite 26).	• Eingangsverzögerung • Flankenauswahl
Torstopp (flankengesteuert)	Die parametrierte Flanke am jeweiligen Digitaleingang schließt das HW-Tor (Seite 26).	• Eingangsverzögerung • Flankenauswahl

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Synchronisation (Seite 33)	Die parametrierte Flanke am jeweiligen Digitaleingang setzt den Zähler auf den Startwert. Das Rückmelde-Bit EVENT_SYNC zeigt an, ob eine Synchronisation aufgetreten ist.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Synchronisation
Freigabe Synchronisation bei Signal N	Der aktive Pegel am jeweiligen Digitaleingang gibt die Synchronisation des Zählers bei Signal N (Seite 37) frei.	• Eingangsverzögerung • Pegelauswahl
Capture	Die parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert (Seite 28) als Capture-Wert. Die Verwendung eines Digitaleingangs und die Verwendung des Signals N schliessen sich für die Capture-Funktion nicht gegenseitig aus. Der Wert CAPTURED_VALUE in der Rückmeldeschnittstelle zeigt den Capture-Wert an.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Capture-Funktion • Verhalten des Zählwerts nach Capture
Digitaleingang ohne Funktion	Dem jeweiligen Digitaleingang ist keine technologische Funktion zugeordnet. Den Signalzustand des Digitaleingangs können Sie über das jeweilige Rückmelde-Bit lesen: • STS_DI0 • STS_DI1 • STS_DI2	• Eingangsverzögerung

Hinweis

Jede Funktion, außer "Digitaleingang ohne Funktion", kann pro Zähler nur einmal verwendet und bei den jeweils anderen Digitaleingängen nicht mehr gewählt werden.

Eingangsverzögerung (TM Count und TM PosInput)

Mit diesem Parameter unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Hinweis

Sie parametrieren die Eingangsverzögerung unter "Verhalten DI0" für alle Digitaleingänge gemeinsam. Die Eingangsverzögerung wird zusätzlich unter "Verhalten DI1" und bei TM Count auch unter "Verhalten DI2" angezeigt.

Eingangsverzögerung (Kompakt-CPU)

Mit diesem Parameter unterdrücken Sie Störungen an den Digitaleingängen der Signale DIn. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Die Eingangsverzögerung für einen Digitaleingang einer Kompakt-CPU parametrieren Sie im Inspektorfenster der Gerätekonfiguration unter "Eigenschaften > DI 16/DQ 16 > Eingänge > Kanal n".

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms
- 0,4 ms
- 1,6 ms
- 3,2 ms (voreingestellt)
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Pegelauswahl

Mit diesem Parameter legen Sie den Pegel fest, bei dem der Digitaleingang aktiv ist.

Sie können unter folgenden Optionen auswählen:

Pegel	Bedeutung
Aktiv bei High-Pegel (voreingestellt)	Der jeweilige Digitaleingang ist aktiv, wenn er gesetzt ist.
Aktiv bei Low-Pegel	Der jeweilige Digitaleingang ist aktiv, wenn er nicht gesetzt ist.

Der Parameter ist bei den folgenden Funktionen eines Digitaleingangs parametrierbar:

- Torstart/-stopp (pegelgesteuert)
- Freigabe Synchronisation bei Signal N

Flankenauswahl

Mit diesem Parameter legen Sie fest, bei welcher Flanke eines Digitaleingangs die parametrierte Funktion ausgelöst wird.

Sie können, abhängig von der gewählten Funktion, unter folgenden Optionen auswählen:

- Bei steigender Flanke (voreingestellt)
- Bei fallender Flanke
- Bei steigender und fallender Flanke

Der Parameter ist bei den folgenden Funktionen eines Digitaleingangs parametrierbar:

- Torstart (flankengesteuert)
- Torstopp (flankengesteuert)
- Synchronisation
- Capture

Hinweis

"Bei steigender und fallender Flanke" ist nur für die Funktion "Capture" parametrierbar.

Häufigkeit der Synchronisation

Mit diesem Parameter legen Sie die Häufigkeit folgender Ereignisse fest:

- Synchronisation bei Signal N
- Synchronisation als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig (voreingestellt)	Der Zähler wird nur bei dem ersten Signal N bzw. der ersten parametrierten Flanke des Digitaleingangs gesetzt.
Periodisch	Der Zähler wird bei jedem Signal N bzw. jeder parametrierten Flanke des Digitaleingangs gesetzt.

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie für folgende Funktionen die Häufigkeit von Capture-Ereignissen fest:

- Capture bei Signal N
- Capture als Funktion eines Digitaleingangs

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang bzw. erste steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang bzw. jede steigende Flanke des Signals N speichert den aktuellen Zählwert als Capture-Wert.

Verhalten des Zählwerts nach Capture

Sie können folgendes Verhalten des Zählers nach einem Capture-Ereignis (Seite 28) parametrieren:

Verhalten	Bedeutung
Zählen fortsetzen (voreingestellt)	Der Zählvorgang wird nach Speichern des aktuellen Zählwerts als Capture-Wert unverändert fortgesetzt.
Setzen auf Startwert und Zählen fortsetzen	Der Zählvorgang wird nach Speichern des aktuellen Zählwerts als Capture-Wert mit dem Startwert fortgesetzt.

Verhalten eines DQ

Ausgang setzen

Mit der Parametrierung eines Digitalausgangs legen Sie die Bedingung fest, bei welcher der Digitalausgang schaltet.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitalausgangs (Seite 41) in der Betriebsart "Zählen"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert und oberer Zählgrenze (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn: Vergleichswert <= Zählwert <= obere Zählgrenze	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert und unterer Zählgrenze	Der jeweilige Digitalausgang ist aktiv, wenn: untere Zählgrenze <= Zählwert <= Vergleichswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)

Funktion eines Digitalausgangs (Seite 41) in der Betriebsart "Zählen"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Zählwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Bei Vergleichswert für eine Impulsdauer	Der jeweilige Digitalausgang ist einmalig aktiv für die parametrierte Zeit und Zählrichtung, wenn der Zählwert dem Vergleichswert entspricht.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Impulsdauer • Hysterese (in Inkrementen)
Nach Setzbefehl aus CPU bis Vergleichswert	Wenn ein Setzbefehl aus der CPU erfolgt, ist der jeweilige Digitalausgang aktiv, bis der Zählwert dem Vergleichswert entspricht.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

DQ0 eines Zählers einer Kompakt-CPU

Bei einer Kompakt-CPU ist der jeweilige Digitalausgang DQ0 über die Rückmeldeschnittstelle, aber nicht als physikalischer Ausgang verfügbar.

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Hinweis

Die Funktionen "Bei Vergleichswert für eine Impulsdauer" und "Nach Setzbefehl aus CPU bis Vergleichswert" schalten den jeweiligen Digitalausgang nur dann, wenn der Vergleichswert durch einen Zählimpuls erreicht wird. Wenn der Zählwert z. B. durch Synchronisation gesetzt wird, schaltet der Digitalausgang nicht.

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messen"	Bedeutung	Weitere optionsspezifische Parameter
Messwert >= Vergleichswert (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert größer oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Messwert <= Vergleichswert	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert kleiner oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messen"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Messwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1
Nicht zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 1 <= Messwert <= Vergleichswert 0	• Vergleichswert 0 • Vergleichswert 1
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle (Seite 40) geschaltet werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" und "Nicht zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Vergleichswert 0 (TM Count und TM PosInput)

Betriebsart "Zählen":

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Ganzzahl (DINT) eingeben, die größer oder gleich der unteren Zählgrenze ist. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messen":

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 0 (Kompakt-CPU)

Betriebsart "Zählen":

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert das Bit STS_DQ0 in der Rückmeldeschnittstelle aufgrund des gewählten Vergleichsereignisses gesetzt wird. Der Digitalausgang DQ0 ist bei einer Kompakt-CPU nicht als physikalischer Ausgang verfügbar.

Sie müssen eine Ganzzahl (DINT) eingeben, die größer oder gleich der unteren Zählgrenze. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messen":

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert das Bit STS_DQ0 in der Rückmeldeschnittstelle aufgrund des gewählten

Vergleichsereignisses gesetzt wird. Der Digitalausgang DQ0 ist bei einer Kompakt-CPU nicht als physikalischer Ausgang verfügbar.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 1

Betriebsart "Zählen":

Mit der Parametrierung des Vergleichswerts (Seite 41) legen Sie fest, bei welchem Zählwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Ganzzahl (DINT) eingeben, die kleiner oder gleich der oberen Zählgrenze ist. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "10".

Betriebsart "Messen":

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Maximalwert beträgt $7{,}922816 \times 10^{28}$. Die Voreinstellung ist "10.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Zählrichtung

Mit diesem Parameter legen Sie fest, für welche Zählrichtung die gewählte Funktion gilt.

Sie können unter folgenden Optionen auswählen:

Zählrichtung	Bedeutung
In beide Richtungen (voreingestellt)	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden unabhängig von der Zählrichtung statt.
Vorwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Zähler vorwärts zählt.
Rückwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Zähler rückwärts zählt.

Der Parameter ist bei den folgenden Funktionen parametrierbar:

- Zwischen Vergleichswert 0 und 1 (Betriebsart "Zählen")
- Bei Vergleichswert für eine Impulsdauer
- Nach Setzbefehl aus CPU bis Vergleichswert

Impulsdauer

Mit der Parametrierung der Impulsdauer für die Funktion "Bei Vergleichswert für eine Impulsdauer" legen Sie fest, wie viele Millisekunden der jeweilige Digitalausgang aktiv ist.

Wenn Sie "0" eingeben und der Zählwert dem jeweiligen Vergleichswert entspricht, ist der Digitalausgang aktiv bis zum nächsten Zählimpuls.

Zulässig ist ein Wert von 0,0 bis 6553,5.

Die Voreinstellung ist "500,0" und entspricht einer Impulsdauer von 0,5 s.

Hysterese (in Inkrementen)

Mit der Parametrierung der Hysterese (Seite 57) legen Sie einen Bereich um die Vergleichswerte fest. Bei den Funktionen "Zwischen Vergleichswert und oberer Zählgrenze" und "Zwischen Vergleichswert und unterer Zählgrenze" gilt die Hysterese auch an den Zählgrenzen. Im Hysteresebereich können die Digitalausgänge nicht erneut schalten, bevor der Zählwert diesen Bereich einmal verlassen hat.

Wählen Sie die Hysterese hinreichend klein. Wenn der Hysteresebereich, ausgehend vom parametrierten Vergleichswert, den gesamten Zählbereich überspannt, kann die korrekte Funktion der Vergleichswerte nicht gewährleistet werden.

Wenn ein Vergleichswert so nahe an einer Zählgrenze liegt, dass der Hysteresebereich über diese Zählgrenze hinausreichen würde, endet der Hysteresebereich an ihr.

Wenn Sie "0" eingeben, ist die Hysterese abgeschaltet. Zulässig ist ein Wert von 0 bis 255. Die Voreinstellung ist "0".

Hinweis

Die Hysterese ist nur in der Betriebsart "Zählen" verfügbar.

Messwert spezifizieren

Messgröße

Mit diesem Parameter legen Sie fest, welche Messgröße (Seite 52) das Technologiemodul bereitstellen soll. Der Wert MEASURED_VALUE in der Rückmeldeschnittstelle zeigt den Messwert an.

Sie können unter folgenden Optionen auswählen:

Messgröße	Bedeutung	Weitere optionsspezifische Parameter
Frequenz (voreingestellt)	Die Messgröße zeigt die Anzahl der Inkremente pro Sekunde. Der Wert ist eine Gleitkommazahl (REAL). Die Einheit ist Hz.	• Aktualisierungszeit
Periodendauer	Die Messgröße ist die durchschnittliche Periodendauer zwischen zwei Inkrementen. Der Wert ist eine Ganzzahl (DINT). Die Einheit ist s.	• Aktualisierungszeit
Geschwindigkeit	Die Messgröße ist eine Geschwindigkeit. Beispiele für eine Geschwindigkeitsmessung finden Sie bei der Erläuterung des Parameters "Inkremente pro Einheit".	• Aktualisierungszeit • Zeitbasis für Geschwindigkeitsmessung • Inkremente pro Einheit

Aktualisierungszeit

Mit der Parametrierung der Aktualisierungszeit (Seite 52) in Millisekunden legen Sie das Zeitintervall zwischen zwei Messwertaktualisierungen fest.

Die Aktualisierungszeit und die Signalart (Seite 143) beeinflussen die Genauigkeit der Messung. Bei Aktualisierungszeiten von mindestens 100 ms ist der Einfluss der Signalart vernachlässigbar.

Bei Aktualisierungszeiten von weniger als 100 ms erreichen Sie die höchste Messgenauigkeit mit den folgenden Signalarten:

- Inkrementalgeber (A, B phasenversetzt) mit Signalauswertung "Einfach"
- Inkrementalgeber (A, B, N) mit Signalauswertung "Einfach"
- Impuls (A) und Richtung (B)
- Impuls (A)

Bei anderen Signalarten hängt die Messgenauigkeit vom verwendeten Geber und der Leitung ab.

Wenn Sie "0" eingeben, wird der Messwert einmal pro modulinternem Zyklus aktualisiert. Sie können bis zu drei Nachkommastellen eingeben. Zulässig ist ein Wert von 0.0 bis 25000.0. Die Voreinstellung ist "10.0".

Zeitbasis für Geschwindigkeitsmessung

Mit diesem Parameter legen Sie fest, mit welcher Zeitbasis die Geschwindigkeit bereitgestellt werden soll.

Sie können unter folgenden Optionen auswählen:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s

Die Voreinstellung ist "60 s".

Inkremente pro Einheit

Mit diesem Parameter legen Sie für die Geschwindigkeitsmessung die Anzahl der Zählimpulse pro relevanter Einheit fest, die der Inkremental- oder Impulsgeber liefert.

Die Anzahl der Zählimpulse ist abhängig von der parametrierten Signalauswertung. Zulässig ist ein Wert von 1 bis 65535.

Beispiel 1:

Ihr Geber liefert 4000 Zählimpulse beim Verfahren um einen Meter. Die Geschwindigkeit soll in Metern pro Sekunde gemessen werden. Als Signalauswertung ist "zweifach" parametriert.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 8000
- Zeitbasis für Geschwindigkeitsmessung: 1 s

Beispiel 2:

Ihr Geber liefert 4096 Zählimpulse pro Umdrehung. Die Geschwindigkeit soll in Umdrehungen pro Minute gemessen werden. Als Signalauswertung ist "Einfach" parametriert.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 4096
- Zeitbasis für Geschwindigkeitsmessung: 60 s

Manueller Betrieb (SSI-Absolutwertgeber)

Zähleingänge

Signalart

Wenn ein SSI-Absolutwertgeber mit Datensignal (Signal D) und Taktsignal (Signal C) angeschlossen ist, wählen Sie die Signalart (Seite 65) "Absolutwertgeber (SSI)".

Richtung invertieren

Mit diesem Parameter invertieren Sie die vom SSI-Absolutwertgeber gelieferten Werte. Dadurch können Sie die erfasste Richtung des Gebers an die Drehrichtung des Motors anpassen.

Hinweis

Dieser Parameter wirkt sich im Telegramm nur auf den Bereich vom LSB bis zum MSB des Positionswerts aus.

Telegrammlänge

Mit der Parametrierung der Telegrammlänge legen Sie die Bitanzahl eines SSI-Telegramms des verwendeten SSI-Absolutwertgebers (Seite 22) fest. Die Telegrammlänge Ihres SSI-Absolutwertgebers entnehmen Sie dem Datenblatt des Gebers. Sonder-Bits zählen auch zur Telegrammlänge. Ein vorhandenes Paritäts-Bit zählt nicht zur Telegrammlänge.

Zulässig ist eine Telegrammlänge von 10 Bit bis 40 Bit. Die Voreinstellung ist "13 Bit".

Zwei Beispiele des Aufbaus des SSI-Telegramms finden Sie unter Beispiele des Telegrammaufbaus (Seite 123).

Codeart

Mit der Parametrierung der Codeart legen Sie fest, ob der Geber Dual-Code oder Gray-Code liefert.

Sie können unter folgenden Optionen auswählen:

Codeart	Bedeutung
Gray (voreingestellt)	Der vom SSI-Absolutwertgeber in Gray-Code gelieferte Positionswert wird in Dual-Code umgewandelt.
Dual	Der vom SSI-Absolutwertgeber gelieferte Wert wird nicht umgewandelt.

Übertragungsgeschwindigkeit

Mit der Parametrierung der Übertragungsgeschwindigkeit legen Sie die Datenübertragungsrate zwischen Technologiemodul und SSI-Absolutwertgeber fest.

Sie können unter folgenden Optionen auswählen:

- 125 kHz (voreingestellt)
- 250 kHz
- 500 kHz
- 1 MHz
- 1,5 MHz
- 2 MHz

Die maximale Übertragungsgeschwindigkeit ist abhängig von der Leitungslänge und den technischen Daten des SSI-Absolutwertgebers. Weitere Informationen entnehmen Sie dem Gerätehandbuch des TM PosInput und der Beschreibung des Gebers.

Monoflopzeit

Mit der Parametrierung der Monoflopzeit legen Sie die Pausenzeit zwischen zwei SSI-Telegrammen fest.

Die parametrierte Monoflopzeit muss mindestens so groß sein wie die Monoflopzeit des eingesetzten SSI-Absolutwertgebers. Sie finden diesen Wert in den technischen Daten des SSI-Absolutwertgebers.

Sie können unter folgenden Optionen auswählen:

- Automatisch (voreingestellt)
- 16 µs
- 32 µs
- 48 µs
- 64 µs

Hinweis

Wenn Sie die Option "Automatisch" wählen, passt sich die Monoflopzeit an den verwendeten Geber automatisch an.

Im taktsynchronen Betrieb entspricht die Option "Automatisch" einer Monoflopzeit von 64 µs. Wenn die Monoflopzeit des eingesetzten SSI-Absolutwertgebers kleiner ist als 64 µs, können Sie den Wert des Gebers auswählen, um schnellere taktsynchrone Zeiten zu erreichen.

Parität

Mit der Parametrierung der Parität legen Sie fest, ob der SSI-Absolutwertgeber ein Paritäts-Bit überträgt.

Wenn z. B. ein 25 Bit-Geber mit Parität parametriert ist, liest das Technologiemodul 26 Bit ein. Ein Paritätsfehler wird in der Rückmeldeschnittstelle über das Bit ENC_ERROR gemeldet.

Bit-Nummer LSB des Positionswerts

Mit diesem Parameter legen Sie die Bit-Nummer des LSB (Least significant bit) des Positionswerts im Telegramm des SSI-Absolutwertgebers fest. Dadurch begrenzen Sie den Bereich im Telegramm, der den Positionswert liefert.

Der Wert muss kleiner als die Bit-Nummer des MSB des Positionswerts sein. Die Differenz zwischen den Bit-Nummern des MSB und des LSB des Positionswerts muss kleiner sein als 32.

Die Voreinstellung ist "0".

Hinweis

Wenn Sie als Codeart "Gray" gewählt haben, wird nur der Bereich vom LSB bis zum MSB des Positionswerts in Dual-Code umgewandelt.

Bit-Nummer MSB des Positionswerts

Mit diesem Parameter legen Sie die Bit-Nummer des MSB (Most significant bit) des Positionswerts im Telegramm des SSI-Absolutwertgebers fest. Dadurch begrenzen Sie den Bereich im Telegramm, der den Positionswert liefert.

Der Wert muss kleiner sein als die Telegrammlänge und größer als die Bit-Nummer des LSB des Positionswerts. Die Differenz zwischen den Bit-Nummern des MSB und des LSB des Positionswerts muss kleiner sein als 32.

Die Voreinstellung ist "12".

Hinweis

Wenn Sie als Codeart "Gray" gewählt haben, wird nur der Bereich vom LSB bis zum MSB des Positionswerts in Dual-Code umgewandelt.

Verhalten eines DI

Funktion des DI einstellen

Mit der Parametrierung eines Digitaleingangs legen Sie fest, welche Funktion der Digitaleingang beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Funktion eines Digitaleingangs	Bedeutung	Weitere optionsspezifische Parameter
Capture	Die parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Positionswert als Capture-Wert. Das Rückmelde-Bit CAPTURED_VALUE zeigt den Capture-Wert an. Die Funktion kann nur für einen der beiden Digitaleingänge verwendet werden.	• Eingangsverzögerung • Flankenauswahl • Häufigkeit der Capture-Funktion
Digitaleingang ohne Funktion	Dem jeweiligen Digitaleingang ist keine technologische Funktion zugeordnet. Den Signalzustand des Digitaleingangs können Sie über das jeweilige Rückmelde-Bit lesen: • STS_DI0 • STS_DI1	• Eingangsverzögerung

Hinweis

Die Funktion "Capture" können Sie nur in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" wählen.

Eingangsverzögerung

Mit der Parametrierung der Eingangsverzögerung unterdrücken Sie Störungen an den Digitaleingängen. Signale mit einer Impulsdauer unterhalb der parametrierten Eingangsverzögerung werden unterdrückt.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms

- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Hinweis

Sie parametrieren die Eingangsverzögerung unter "Verhalten DI0" für alle Digitaleingänge gemeinsam. Die Eingangsverzögerung wird zusätzlich unter "Verhalten DI1" angezeigt.

Flankenauswahl

Mit diesem Parameter legen Sie für die Funktion "Capture" fest, bei welcher Flanke eines Digitaleingangs die parametrierte Funktion ausgelöst wird.

Sie können unter folgenden Optionen auswählen:

- Bei steigender Flanke (voreingestellt)
- Bei fallender Flanke
- Bei steigender und fallender Flanke

Häufigkeit der Capture-Funktion

Mit diesem Parameter legen Sie die Häufigkeit von Capture-Ereignissen fest:

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Einmalig	Die erste parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert als Capture-Wert.
Periodisch (voreingestellt)	Jede parametrierte Flanke am jeweiligen Digitaleingang speichert den aktuellen Zählwert als Capture-Wert.

Verhalten eines DQ

Ausgang setzen

Mit der Parametrierung eines Digitalausgangs legen Sie die Bedingung fest, bei welcher der Digitalausgang schaltet.

Sie können je nach Betriebsart unter folgenden Optionen auswählen:

Funktion eines Digitalausgangs (Seite 45) in der Betriebsart "Positionserfassung"	Bedeutung	Weitere optionsspezifische Parameter
Zwischen Vergleichswert und oberer Zählgrenze (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn: Vergleichswert <= Positionswert <= maximaler Positionswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert und unterer Zählgrenze	Der jeweilige Digitalausgang ist aktiv, wenn: Minimaler Positionswert <= Positionswert <= Vergleichswert	• Vergleichswert 0 • Vergleichswert 1 • Hysterese (in Inkrementen)
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Positionswert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Bei Vergleichswert für eine Impulsdauer	Der jeweilige Digitalausgang ist einmalig aktiv für die parametrierte Zeit und Richtung der Positionswertänderung, wenn der Positionswert dem Vergleichswert entspricht oder ihn überschritten/unterschritten hat.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Impulsdauer • Hysterese (in Inkrementen)
Nach Setzbefehl aus CPU bis Vergleichswert	Wenn ein Setzbefehl aus der CPU erfolgt, ist der jeweilige Digitalausgang aktiv für die parametrierte Richtung der Positionswertänderung, bis der Positionswert dem Vergleichswert entspricht oder ihn überschritten/unterschritten hat.	• Vergleichswert 0 • Vergleichswert 1 • Zählrichtung • Hysterese (in Inkrementen)
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messen"	Bedeutung	Weitere optionsspezifische Parameter
Messwert >= Vergleichswert (voreingestellt)	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert größer oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Messwert <= Vergleichswert	Der jeweilige Digitalausgang ist aktiv, wenn der Messwert kleiner oder gleich dem Vergleichswert ist.	• Vergleichswert 0 • Vergleichswert 1
Zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 0 <= Messwert <= Vergleichswert 1	• Vergleichswert 0 • Vergleichswert 1

Funktion eines Digitalausgangs (Seite 49) in der Betriebsart "Messen"	Bedeutung	Weitere optionsspezifische Parameter
Nicht zwischen Vergleichswert 0 und 1	Der Digitalausgang DQ1 ist aktiv, wenn: Vergleichswert 1 <= Messwert <= Vergleichswert 0	• Vergleichswert 0 • Vergleichswert 1
Nutzung durch Anwenderprogramm	Der jeweilige Digitalausgang kann von der CPU über die Steuerschnittstelle geschaltet (Seite 40) werden.	—

Hinweis

Sie können die Funktion "Zwischen Vergleichswert 0 und 1" und "Nicht zwischen Vergleichswert 0 und 1" nur für den Digitalausgang DQ1 wählen und nur, wenn Sie für den Digitalausgang DQ0 die Funktion "Nutzung durch Anwenderprogramm" gewählt haben.

Vergleichswert 0

Betriebsart "Positionserfassung":

Mit der Parametrierung des Vergleichswerts (Seite 45) legen Sie fest, bei welchem Positionswert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge bis 31 Bit verwenden, müssen Sie eine positive Ganzzahl (DINT) eingeben mit einem Wert zwischen 0 und $2^{(MSB-LSB+1)}$-1. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge von 32 Bit verwenden, müssen Sie eine vorzeichenbehaftete Ganzzahl (DINT) eingeben mit einem Wert zwischen –2147483648 und 2147483647.

Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "0".

Betriebsart "Messen":

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ0 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Minimalwert beträgt $-7{,}922816 \times 10^{28}$. Die Voreinstellung ist "0.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Vergleichswert 1

Betriebsart "Positionserfassung"

Mit der Parametrierung des Vergleichswerts (Seite 45) legen Sie fest, bei welchem Positionswert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge bis 31 Bit verwenden, müssen Sie eine positive Ganzzahl (DINT) eingeben mit einem Wert zwischen 0 und $2^{(MSB-LSB+1)}$-1. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge von 32 Bit verwenden, müssen Sie eine vorzeichenbehaftete Ganzzahl (DINT) eingeben mit einem Wert zwischen –2147483648 und 2147483647.

Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Die Voreinstellung ist "10".

Betriebsart "Messen"

Mit der Parametrierung des Vergleichswerts (Seite 49) legen Sie fest, bei welchem Messwert der Digitalausgang DQ1 aufgrund des gewählten Vergleichsereignisses schaltet.

Sie müssen eine Gleitkommazahl (REAL) eingeben. Wenn Sie die DQ-Funktion "Zwischen Vergleichswert 0 und 1" verwenden, muss Vergleichswert 0 kleiner als Vergleichswert 1 sein. Der Maximalwert beträgt 7,922816 x 10^{28}. Die Voreinstellung ist "10.0". Die Einheit des Vergleichswerts ist abhängig von der Messgröße.

Zählrichtung

Mit diesem Parameter legen Sie fest, für welche Richtung der Positionswertänderung die gewählte Funktion gilt.

Sie können unter folgenden Optionen auswählen:

Richtung der Positionswertänderung	Bedeutung
In beide Richtungen (voreingestellt)	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden unabhängig davon statt, ob der Positionswert kleiner oder größer wird.
Vorwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Positionswert größer wird.
Rückwärts	Der Vergleich und das Schalten des jeweiligen Digitalausgangs finden nur statt, wenn der Positionswert kleiner wird.

Der Parameter ist bei den folgenden Funktionen parametrierbar:

- Zwischen Vergleichswert 0 und 1 (Betriebsart "Positionserfassung")
- Bei Vergleichswert für eine Impulsdauer
- Nach Setzbefehl aus CPU bis Vergleichswert

Impulsdauer

Mit der Parametrierung der Impulsdauer für die Funktion "Bei Vergleichswert für eine Impulsdauer" legen Sie fest, wie viele Millisekunden der jeweilige Digitalausgang aktiv ist.

Zulässig ist ein Wert von 0,1 bis 6553,5 ms.

Die Voreinstellung ist "500,0" und entspricht einer Impulsdauer von 0,5 s.

Hysterese (in Inkrementen)

Mit der Parametrierung der Hysterese (Seite 59) legen Sie einen Bereich um die Vergleichswerte fest. Bei den Funktionen "Zwischen Vergleichswert und oberer Zählgrenze" und "Zwischen Vergleichswert und unterer Zählgrenze" gilt die Hysterese auch an den Zählgrenzen. Im Hysteresebereich können die Digitalausgänge nicht erneut schalten, bevor der Positionswert diesen Bereich einmal verlassen hat.

Wählen Sie die Hysterese hinreichend klein. Wenn der Hysteresebereich, ausgehend vom parametrierten Vergleichswert, den gesamten Positionswertebereich überspannt, kann die korrekte Funktion der Vergleichswerte nicht gewährleistet werden.

Wenn ein Vergleichswert so nahe an einer Zählgrenze liegt, dass der Hysteresebereich über diese Zählgrenze hinausreichen würde, endet der Hysteresebereich an ihr.

Wenn Sie "0" eingeben, ist die Hysterese abgeschaltet. Zulässig ist ein Wert von 0 bis 255. Die Voreinstellung ist "0".

Hinweis

Die Hysterese ist nur in der Betriebsart "Positionserfassung" verfügbar.

Messwert spezifizieren

Messgröße

Mit diesem Parameter legen Sie fest, ob das Technologiemodul eine bestimmte Messgröße (Seite 52) oder das vollständige SSI-Telegramm bereitstellen soll.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung	Weitere optionsspezifische Parameter
Frequenz (voreingestellt)	Die Messgröße zeigt die Anzahl der Inkremente pro Sekunde, wobei jedes Inkrement einer Positionswertänderung entspricht. Der Wert ist eine Gleitkommazahl (REAL). Die Einheit ist Hz. Der Wert MEASURED_VALUE in der Rückmeldeschnittstelle zeigt den Messwert an.	• Aktualisierungszeit
Periodendauer	Die Messgröße ist die durchschnittliche Periodendauer zwischen zwei Inkrementen des Positionswerts. Der Wert ist eine Ganzzahl (DINT). Die Einheit ist s. Der Wert MEASURED_VALUE in der Rückmeldeschnittstelle zeigt den Messwert an.	• Aktualisierungszeit
Geschwindigkeit	Die Messgröße ist eine Geschwindigkeit. Beispiele für eine Geschwindigkeitsmessung finden Sie bei der Erläuterung des Parameters "Inkremente pro Einheit". Der Wert MEASURED_VALUE in der Rückmeldeschnittstelle zeigt den Messwert an.	• Aktualisierungszeit • Zeitbasis für Geschwindigkeitsmessung • Inkremente pro Einheit
Vollständiges SSI-Telegramm	Anstelle einer Messgröße werden die ersten 32 Bits des SSI-Telegramms zurückgeliefert (Bit 0 bis Bit 31). Dabei werden auch Sonder-Bits mitgeliefert, die nicht zur Positionsinformation gehören. Eine parametrierte Richtungsinvertierung wird nicht berücksichtigt. Der Wert MEASURED_VALUE in der Rückmeldeschnittstelle zeigt die 32 Bits an. Beispiele finden Sie unter Beispiele des Telegrammaufbaus (Seite 123). Diese Option ist nur in der Betriebsart "Positionswert (SSI-Absolutwert) als Bezug verwenden" verfügbar.	—

Hinweis

Wenn zur Berechnung des Messwerts die Inkrementzahl pro Umdrehung benötigt wird, wird sie automatisch aus der parametrierten Telegrammlänge als 2er-Potenz berechnet, z.B. 8192 Inkremente pro Umdrehung bei einer Telegrammlänge von 13 Bit. Wenn Sie einen SSI-Absolutwertgeber verwenden, dessen Inkrementzahl pro Umdrehung nicht der 2er-Potenz entspricht, kann der berechnete Messwert kurzzeitig fehlerhaft sein.

Aktualisierungszeit

Mit der Parametrierung der Aktualisierungszeit (Seite 52) in Millisekunden legen Sie das Zeitintervall zwischen zwei Messwertaktualisierungen fest. Durch größere Aktualisierungszeiten können unruhige Messgrößen geglättet werden.

Wenn Sie "0" eingeben, wird der Messwert einmal pro modulinternem Zyklus aktualisiert. Sie können bis zu drei Nachkommastellen eingeben. Zulässig ist ein Wert von 0.0 bis 25000.0. Die Voreinstellung ist "10.0".

Zeitbasis für Geschwindigkeitsmessung

Mit diesem Parameter legen Sie fest, mit welcher Zeitbasis die Geschwindigkeit bereitgestellt werden soll.

Sie können unter folgenden Optionen auswählen:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s

Die Voreinstellung ist "60 s".

Inkremente pro Einheit

Mit diesem Parameter legen Sie für die Geschwindigkeitsmessung die Anzahl der Inkremente pro relevanter Einheit fest, die der SSI-Absolutwertgeber liefert.

Zulässig ist ein Wert von 1 bis 65535.

Beispiel 1:

Ihr Geber arbeitet mit einer Auflösung von 12 Bit pro Umdrehung und liefert 4096 Inkremente pro Umdrehung. Die Geschwindigkeit soll in Umdrehungen pro Minute gemessen werden.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 4096
- Zeitbasis für Geschwindigkeitsmessung: 60 s

Beispiel 2:

Ihr Geber liefert 10000 Inkremente beim Verfahren um einen Meter. Die Geschwindigkeit soll in Metern pro Sekunde gemessen werden.

Sie müssen in diesem Fall Folgendes parametrieren:

- Inkremente pro Einheit: 10000
- Zeitbasis für Geschwindigkeitsmessung: 1 s

Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über das Technologiemodul erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen (TM Count und TM PosInput)

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie mit der rechten Maustaste auf das Modul und wählen Sie "Online & Diagnose".
1. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Vorgehen (Kompakt-CPU)

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der Kompakt-CPU.
2. Doppelklicken Sie auf das Objekt "Online & Diagnose".
3. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch des Technologiemoduls.

Hinweis

Positionserfassung für Motion Control

Im Betriebsmodus "Positionserfassung für Technologieobjekt "Motion Control"" ist die Kanaldiagnose für das Technologiemodul nicht verfügbar.

Steuer- und Rückmeldeschnittstelle (TM Count, TM PosInput)

Informationen zur Verwendung der Steuer- und Rückmeldeschnittstelle finden Sie in der Übersicht der Einsatzmöglichkeiten (Seite 15).

Hinweis

Die nachfolgende Beschreibung gilt nicht für die Betriebsmodi "Fast Mode" und "Positionserfassung für Technologieobjekt "Motion Control"". Für Technologiemodule, die den Betriebsmodus "Fast Mode" unterstützen, finden Sie eine Beschreibung der Rückmeldeschnittstelle im jeweiligen Gerätehandbuch.

Belegung der Steuerschnittstelle

Über die Steuerschnittstelle beeinflusst das Anwenderprogramm das Verhalten des Technologiemoduls.

Steuerschnittstelle pro Kanal

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle:

Byte-Offset zur Anfangsadresse Kanal 0/1 ↓	↓	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 … 3	12 … 15	SLOT_0: DINT oder REAL: Ladewert (Bedeutung des Werts wird in LD_SLOT_0 spezifiziert) Wertebereich: –2147483648 bis 2147483647_D bzw. 80000000 bis $7FFFFFFF_H$							
4 … 7	16 … 19	SLOT_1: DINT oder REAL: Ladewert (Bedeutung des Werts wird in LD_SLOT_1 spezifiziert) Wertebereich: –2147483648 bis 2147483647_D bzw. 80000000 bis $7FFFFFFF_H$							
8	20	LD_SLOT_1				LD_SLOT_0			
9	21	EN_ CAPTURE	EN_ SYNC_DN	EN_ SYNC_UP	SET_DQ1	SET_DQ0	TM_ CTRL_DQ1	TM_ CTRL_DQ0	SW_GATE
10	22	SET_DIR	Reserviert					RES_ EVENT	RES_ ERROR
11	23	Reserviert							

Erläuterungen

Steuer-Bit/Wert	Erläuterungen
EN_CAPTURE	Mit diesem Bit geben Sie die Capture-Funktion frei. Ein Rücksetzen des Bits setzt ein gesetztes EVENT_CAP in der Rückmeldeschnittstelle zurück.
EN_SYNC_DN	Mit diesem Bit geben Sie bei Verwendung eines Inkremental- oder Impulsgebers die Synchronisation des Zählers beim Zählen in Rückwärtsrichtung frei. Ein Rücksetzen des Bits setzt ein gesetztes EVENT_SYNC in der Rückmeldeschnittstelle zurück.

Steuer-Bit/Wert	Erläuterungen
EN_SYNC_UP	Mit diesem Bit geben Sie bei Verwendung eines Inkremental- oder Impulsgebers die Synchronisation des Zählers beim Zählen in Vorwärtsrichtung frei. Ein Rücksetzen des Bits setzt ein gesetztes EVENT_SYNC in der Rückmeldeschnittstelle zurück.
LD_SLOT_m	Mit dieser Ladeaufforderung spezifizieren Sie die Bedeutung des Werts in SLOT_m: • 0000 bedeutet: Keine Aktion, Ruhezustand • 0001 bedeutet: Zählwert laden (bei Inkremental- oder Impulsgeber) • 0010 nicht zulässig • 0011 bedeutet: Startwert laden (bei Inkremental- oder Impulsgeber) • 0100 bedeutet: Vergleichswert 0 laden • 0101 bedeutet: Vergleichswert 1 laden • 0110 bedeutet: Untere Zählgrenze laden (bei Inkremental- oder Impulsgeber) • 0111 bedeutet: Obere Zählgrenze laden (bei Inkremental- oder Impulsgeber) • 1000 bis 1111 nicht zulässig Das Technologiemodul führt die jeweilige Aktion aus, sobald sich LD_SLOT_m ändert. Wenn über LD_SLOT_0 und LD_SLOT_1 gleichzeitig Werte geladen werden, wird intern erst der Wert aus SLOT_0 und anschließend der Wert aus SLOT_1 übernommen. Dadurch können unerwartete Zwischenzustände auftreten.
RES_EVENT	Mit diesem Bit stoßen Sie das Rücksetzen der gespeicherten Ereignisse in den Rückmeldebits EVENT_ZERO, EVENT_OFLW, EVENT_UFLW, EVENT_CMP0, EVENT_CMP1 an.
Reserviert	Reserve-Bits müssen auf 0 gesetzt sein.
RES_ERROR	Mit diesem Bit stoßen Sie das Rücksetzen der gespeicherten Fehlerzustände LD_ERROR und ENC_ERROR an.
SET_DIR	Mit diesem Bit geben Sie bei der Signalart "Impuls (A)" die Zählrichtung vor. • 0 bedeutet: vorwärts • 1 bedeutet: rückwärts
SET_DQ0	Mit diesem Bit setzen Sie den Digitalausgang DQ0, wenn TM_CTRL_DQ0 auf 0 gesetzt ist. Bei der Funktion "Nach Setzbefehl aus CPU bis Vergleichswert" wirkt SET_DQ0 unabhängig von TM_CTRL_DQ0, solange der Zählwert nicht dem Vergleichswert entspricht.
SET_DQ1	Mit diesem Bit setzen Sie den Digitalausgang DQ1, wenn TM_CTRL_DQ1 auf 0 gesetzt ist. Bei der Funktion "Nach Setzbefehl aus CPU bis Vergleichswert" wirkt SET_DQ1 unabhängig von TM_CTRL_DQ1, solange der Zählwert nicht dem Vergleichswert entspricht.
SW_GATE	Mit diesem Bit öffnen und schließen Sie bei Verwendung eines Inkremental- oder Impulsgebers das Software-Tor. Das Software-Tor bildet zusammen mit dem Hardware-Tor das interne Tor. Nur wenn das interne Tor offen ist, zählt das Technologiemodul. • 0 bedeutet: Software-Tor geschlossen • 1 bedeutet: Software-Tor offen Die Steuerung des Hardware-Tors erfolgt extern über die Digitaleingänge des Technologiemoduls. Das Hardware-Tor kann durch Parametrierung aktiviert werden. Das Software-Tor kann nicht deaktiviert werden.
TM_CTRL_DQ0	Mit diesem Bit geben Sie die technologische Funktion des Digitalausgangs DQ0 frei. • 0 bedeutet: SET_DQ0 bestimmt Zustand von DQ0 • 1 bedeutet: parametrierte Funktion bestimmt Zustand von DQ0
TM_CTRL_DQ1	Mit diesem Bit geben Sie die technologische Funktion des Digitalausgangs DQ1 frei. • 0 bedeutet: SET_DQ1 bestimmt Zustand von DQ1 • 1 bedeutet: parametrierte Funktion bestimmt Zustand von DQ1

Belegung der Rückmeldeschnittstelle

Über die Rückmeldeschnittstelle empfängt das Anwenderprogramm vom Technologiemodul aktuelle Werte und Statusinformationen.

Rückmeldeschnittstelle pro Kanal

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle:

Byte-Offset zur Anfangsadresse Kanal 0/1 ↓	↓	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 … 3	16 … 19	COUNT_VALUE: DINT: Aktueller Zählwert oder Positionswert							
4 … 7	20 … 23	CAPTURED_VALUE: DINT: Letzter erfasster Capture-Wert							
8 ... 11	24 ... 27	MEASURED_VALUE: REAL: Aktueller Messwert oder DWORD: vollständiges SSI-Telegramm							
12	28	Reserviert					LD_ERROR	ENC_ ERROR	POWER_ ERROR
13	29	Reserviert		STS_SW_ GATE	STS_ READY	LD_STS_ SLOT_1	LD_STS_ SLOT_0	RES_EVEN T_ACK	Reserviert
14	30	STS_DI2[1]	STS_DI1	STS_DI0	STS_DQ1	STS_DQ0	STS_GATE	STS_CNT	STS_DIR
15	31	STS_M_ INTERVAL	EVENT_ CAP	EVENT_ SYNC	EVENT_ CMP1	EVENT_ CMP0	EVENT_ OFLW	EVENT_ UFLW	EVENT_ ZERO

[1] Bei TM PosInput gilt: Reserviert

Hinweis

Gültigkeit des Positionswerts

Der Positionswert eines SSI-Absolutwertgebers ist gültig, wenn STS_READY auf 1 gesetzt und ENC_ERROR auf 0 gesetzt ist. Im Anlauf des Moduls ist STS_READY auf 0 gesetzt.

Erläuterungen

Rückmelde-Bit/Wert	Erläuterungen
CAPTURED_VALUE	Dieser DINT-Wert zeigt den letzten erfassten Capture-Wert an. Folgende externe Signale können die Capture-Funktion auslösen: • Steigende oder fallende Flanke eines Digitaleingangs • Beide Flanken eines Digitaleingangs Der Parameter "Häufigkeit der Capture-Funktion" legt fest, ob die Funktion bei jeder parametrierten Flanke oder nur einmalig nach jeder Freigabe ausgeführt wird.
COUNT_VALUE	Dieser DINT-Wert zeigt den aktuellen Zählwert oder Positionswert an. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge bis 31 Bit verwenden, wird der Positionswert vorzeichenlos als positiver Wert behandelt und kann Werte zwischen 0 und $2^{(MSB-LSB+1)}-1$ annehmen. Wenn Sie einen SSI-Absolutwertgeber mit einer Positionswertlänge von 32 Bit verwenden, entspricht das MSB des Positionswerts dem Vorzeichen und der Positionswert kann Werte zwischen –2147483648 und 2147483647 annehmen. Wenn Sie einen 32-Bit-Positionswert für die Vergleichsfunktion verwenden, wird der Positionswert dabei als DINT interpretiert.
ENC_ERROR	Dieses Bit zeigt an, dass beim jeweiligen Technologiemodul an den Gebersignalen einer der folgenden Fehler aufgetreten ist (speichernd): TM Count: • Drahtbruch digitaler Eingang A, B oder N (bei Gegentakt-schaltendem Geber) • Illegaler Übergang der A/B-Signale (bei Inkrementalgeber) TM PosInput: • Illegaler Übergang der A/B-Signale (bei Inkrementalgeber) • RS422/TTL-Fehler • Fehler am SSI-Geber oder des SSI-Telegramms (bei SSI-Absolutwertgeber) Wenn Sie die Diagnosealarme freigegeben haben, wird bei einem Fehler an den Gebersignalen der jeweilige Diagnosealarm ausgelöst. Informieren Sie sich im Gerätehandbuch zum jeweiligen Technologiemodul, welche Bedeutung die Diagnosealarme haben. Das Bit wird zurückgesetzt, nachdem Sie den Fehler mit RES_ERROR quittiert haben.
EVENT_CAP	Dieses Bit zeigt an, dass ein Capture-Ereignis aufgetreten ist und ein Zählwert in CAPTURED_VALUE gespeichert wurde. Sie setzen den Zustand durch Rücksetzen von EN_CAPTURE zurück.
EVENT_CMP0	Dieses Bit zeigt den gespeicherten Zustand an, dass für den Digitalausgangs DQ0 ein Vergleichsereignis (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten ist. Sie setzen den Zustand durch Quittierung mit RES_EVENT zurück. Wenn in der Betriebsart Zählen der Zählwert auf den Startwert gesetzt wird, wird EVENT_CMP0 nicht gesetzt.
EVENT_CMP1	Dieses Bit zeigt den gespeicherten Zustand an, dass für den Digitalausgangs DQ1 ein Vergleichsereignis (Zustandsänderung) aufgrund der gewählten Vergleichsbedingung aufgetreten ist. Sie setzen den Zustand durch Quittierung mit RES_EVENT zurück. Wenn in der Betriebsart Zählen der Zählwert auf den Startwert gesetzt wird, wird EVENT_CMP1 nicht gesetzt.
EVENT_OFLW	Dieses Bit zeigt den gespeicherten Zustand an, dass der Zählwert einen Überlauf hatte. Sie setzen den Zustand durch Quittierung mit RES_EVENT zurück.
EVENT_SYNC	Dieses Bit zeigt bei Verwendung eines Inkremental- oder Impulsgebers den gespeicherten Zustand an, dass der Zähler durch ein externes Referenzsignal mit dem Startwert geladen wurde (Synchronisation). Sie setzen den Zustand durch das Rücksetzen von EN_SYNC_UP oder EN_SYNC_DN zurück.
EVENT_UFLW	Dieses Bit zeigt den gespeicherten Zustand an, dass der Zählwert einen Unterlauf hatte. Sie setzen den Zustand durch Quittierung mit RES_EVENT zurück.

Rückmelde-Bit/Wert	Erläuterungen
EVENT_ZERO	Dieses Bit zeigt den gespeicherten Zustand an, dass der Zählwert oder der Positionswert einen Nulldurchgang hatte. Sie setzen den Zustand durch Quittierung mit RES_EVENT zurück.
LD_ERROR	Dieses Bit zeigt an, dass beim Laden über die Steuerschnittstelle ein Fehler aufgetreten ist (speichernd). Die Ladewerte wurden nicht angenommen. Bei Verwendung eines Inkremental- oder Impulsgebers ist eine der folgenden Bedingungen nicht erfüllt: • Untere Zählgrenze <= Zählwert <= Obere Zählgrenze • Untere Zählgrenze <= Startwert <= Obere Zählgrenze • Untere Zählgrenze <= Vergleichswert 0/1 <= Obere Zählgrenze Bei Verwendung eines SSI-Absolutwertgebers ist eine der folgenden Bedingungen nicht erfüllt: • 0 <= Positionswert <= maximaler Positionswert • 0 <= Vergleichswert 0/1 <= maximaler Positionswert Das Bit wird zurückgesetzt, nachdem Sie den Fehler mit RES_ERROR quittiert haben.
LD_STS_SLOT_0	Dieses Bit zeigt durch einen Zustandswechsel (Toggeln) an, dass die Ladeaufforderung für Slot 0 (LD_SLOT_0) erkannt und durchgeführt wurde.
LD_STS_SLOT_1	Dieses Bit zeigt durch einen Zustandswechsel (Toggeln) an, dass die Ladeaufforderung für Slot 1 (LD_SLOT_1) erkannt und durchgeführt wurde.
MEASURED_VALUE	Dieser Wert zeigt den aktuellen Messwert mit Datentyp REAL oder das vollständige SSI-Telegramm mit Datentyp DWORD an: • Frequenz: Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen wird in einem Messintervall die mittlere Frequenz ermittelt und als Gleitkommazahl in der Einheit Hertz zurückgeliefert. • Periodendauer: Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen wird in einem Messintervall die mittlere Periodendauer ermittelt und als Gleitkommazahl in der Einheit Sekunden zurückgeliefert. • Geschwindigkeit: Aus dem zeitlichen Verlauf der Zählimpulse oder Positionswertänderungen und weiteren Parametern wird in einem Messintervall die mittlere Geschwindigkeit ermittelt und in der parametrierten Einheit zurückgeliefert. • Vollständiges SSI-Telegramm: Anstelle einer Messgröße werden die niederwertigsten 32 Bit des unbearbeiteten aktuellen SSI-Telegramms zurückliefert. Dadurch erhalten Sie zusätzlich zum Positionswert geberspezifische Zusatz-Bits, z. B. Fehler-Bits. Wenn das SSI-Telegramm kürzer ist als 32 Bit, werden in der Rückmeldeschnittstelle das vollständige SSI-Telegramm rechtsbündig und die oberen ungenutzten Bits mit "0" zurückgeliefert. Die Messwerte werden als vorzeichenbehafteter Wert zurückgeliefert. Das Vorzeichen gibt dabei an, ob der Zählwert oder Positionswert im relevanten Zeitintervall gestiegen oder gefallen ist. Die Aktualisierungszeit ist asynchron zum Öffnen des internen Tors, d.h. die Aktualisierungszeit wird nicht mit dem Öffnen gestartet. Nach dem Schließen des internen Tors wird der zuletzt ermittelte Messwert weiter zurückgeliefert.
POWER_ERROR	Dieses Bit zeigt für ein S7-1500-Technologiemodul an, dass die Versorgungsspannung L+ nicht vorhanden oder zu niedrig ist oder der Frontstecker nicht gesteckt ist. Dieses Bit zeigt für ein ET 200SP-Technologiemodul an, dass die Versorgungsspannung L+ zu niedrig ist. Wenn Sie die Diagnosealarme freigegeben haben, wird bei einem Fehler an der Versorgungsspannung der jeweilige Diagnosealarm ausgelöst. Informieren Sie sich im Gerätehandbuch zum jeweiligen Technologiemodul über die Diagnosealarme und zugehörige Abhilfemaßnahmen. Wenn die Versorgungsspannung L+ wieder in ausreichender Höhe vorhanden ist, wird POWER_ERROR automatisch auf 0 gesetzt.
RES_EVENT_ACK	Dieses Bit zeigt an, dass das Rücksetzen der Ereignis-Bits EVENT_SYNC, EVENT_CMP0, EVENT_CMP1, EVENT_OFLW, EVENT_UFLW, EVENT_ZERO aktiv ist.

Rückmelde-Bit/Wert	Erläuterungen
Reserviert	Reserve-Bits sind auf 0 gesetzt.
STS_CNT	Dieses Bit zeigt an, dass innerhalb der letzten ca. 0,5 s mindestens ein Zählimpuls oder eine Positionswertänderung erfasst wurde.
STS_DI0	Dieses Bit zeigt den Zustand des Digitaleingangs DI0 an.
STS_DI1	Dieses Bit zeigt den Zustand des Digitaleingangs DI1 an.
STS_DI2	Dieses Bit zeigt den Zustand des Digitaleingangs DI2 des TM Count an.
STS_DIR	Dieses Bit zeigt die Zählrichtung des letzten Zählimpulses oder die Richtung der letzten Positionswertänderung an. 0 bedeutet: rückwärts 1 bedeutet: vorwärts
STS_DQ0	Dieses Bit zeigt den Zustand des Digitalausgangs DQ0 an.
STS_DQ1	Dieses Bit zeigt den Zustand des Digitalausgangs DQ1 an.
STS_GATE	Dieses Bit zeigt bei Verwendung eines Inkremental- oder Impulsgebers den Zustand des internen Tors an. 0 bedeutet: Tor geschlossen 1 bedeutet: Tor offen Hinweis für TM PosInput: Damit die Zähllogik inklusive der Torsteuerung korrekt arbeitet, muss der Anlauf des Technologiemoduls mindestens einmal mit angeschlossenem Inkremental- oder Impulsgeber korrekt abschliessen (STS_READY auf 1). Wenn ein angeschlossener Geber während des Anlaufs noch nicht bereit ist, wird die Funktion des Rückmelde-Bits STS_GATE verzögert, bis der Geber für das Technologiemodul verfügbar ist. Wenn das Technologiemodul ohne einen angeschlossenen Geber anläuft, schliesst der Anlauf nicht korrekt ab und STS_READY sowie STS_GATE bleiben auf 0 gesetzt. Sobald dann ein Geber angeschlossen wird, schliesst der Anlauf ab und STS_GATE funktioniert korrekt. Ein Geberfehler nach einem abgeschlossenen Anlauf hat keinen Einfluss auf STS_GATE.
STS_M_INTERVAL	Dieses Bit zeigt an, dass im vorangegangenen Messintervall mindestens ein Zählimpuls oder eine Positionswertänderung erfasst wurde.
STS_READY	Dieses Bit zeigt an, dass das Technologiemodul gültige Nutzdaten liefert. Das Technologiemodul ist angelaufen und parametriert.
STS_SW_GATE	Dieses Bit zeigt den Zustand des SW-Tors an. 0 bedeutet: Tor geschlossen 1 bedeutet: Tor offen

Vollständiges Quittungsprinzip

Speichernde Bits werden nach dem vollständigen Quittungsprinzip quittiert.

Das folgende Bild zeigt ein Beispiel für den Ablauf des vollständigen Quittungsprinzips bei einem Überlauf:

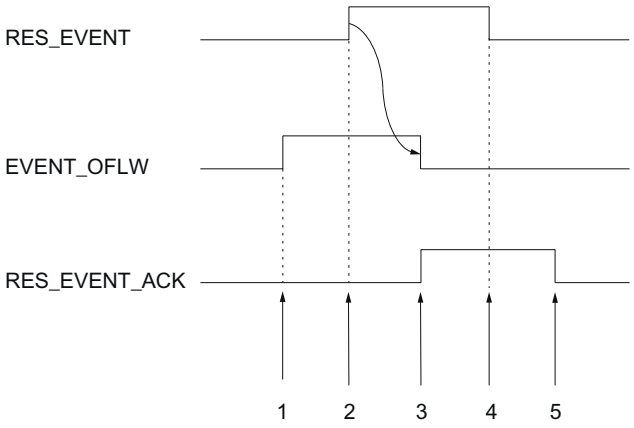

① Das Rückmeldebit EVENT_OFLW wird bei einem Überlauf als speicherndes Ereignis gesetzt.
② Sie setzen das Steuerbit RES_EVENT, um das Rücksetzen von EVENT_OFLW anzustoßen.
③ Das Rückmeldebit RES_EVENT_ACK wird gesetzt, wenn das Rücksetzen von EVENT_OFLW erkannt wurde.
④ Sie setzen das Steuerbit RES_EVENT zurück.
⑤ Das Rückmeldebit RES_EVENT_ACK wird rückgesetzt.

1.1.4.2 Digitalmodul einsetzen

Modul konfigurieren und parametrieren

Modul zur Hardware-Konfiguration hinzufügen

Voraussetzungen

- Das Projekt ist angelegt.
- Die CPU ist angelegt.
- Dezentrale Peripherie ET 200 ist angelegt.

Vorgehen

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie das Digitalmodul aus dem Baugruppenkatalog aus:
   "DI > Digitaleingabemodul > Artikelnummer" oder
   "DIQ > Digitalein-/-ausgabemodul > Artikelnummer"
4. Ziehen Sie das Modul per Drag & Drop auf den gewünschten Steckplatz im Baugruppenträger.

Hardware-Konfiguration öffnen

Öffnen

Um die Hardware-Konfiguration zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie auf das Modul.

Betriebsart Zählen

In der Betriebsart Zählen bzw. in der Zählerkonfiguration können Sie für den jeweiligen Kanal die nachfolgenden Parameter einstellen.

Hinweis

Einige der Parameter und Optionen sind nicht bei allen Digitalmodulen verfügbar. Informieren Sie sich im Gerätehandbuch zum Modul über die zugehörigen Parameter und Optionen.

Kanal aktiviert

Mit diesem Parameter legen Sie fest, ob der jeweilige Kanal aktiviert oder deaktiviert ist.

In der Voreinstellung ist der jeweilige Kanal aktiviert.

Eingangsverzögerung

Mit der Parametrierung der Eingangsverzögerung unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die eingestellte Eingangsverzögerungszeit stabil anstehen.

Eine taktsynchrone Projektierung ist nur möglich, wenn bei mindestens einem Kanal eine Eingangsverzögerung von 0,05 ms parametriert ist. Im taktsynchronen Betrieb wird die Rückmeldeschnittstelle zum Zeitpunkt T_i (Zeit zum Einlesen der Eingangsdaten) aktualisiert.

Sie können unter folgenden Optionen auswählen:

- 0,05 ms
- 0,1 ms
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms (voreingestellt)
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie für die Eingangsverzögerung die Option "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Verhalten bei Überschreiten einer Zählgrenze

Sie können folgendes Verhalten bei Überschreiten der oberen Zählgrenze in Vorwärtsrichtung oder der unteren Zählgrenze in Rückwärtsrichtung (Seite 75) parametrieren:

Verhalten	Bedeutung
Zählen stoppen (voreingestellt bei ET200SP und ET200AL)	Nach Überschreiten einer Zählgrenze wird das interne Tor geschlossen (automatischer Torstopp). Dadurch wird der Zählvorgang angehalten und weitere Zählsignale werden vom Modul ignoriert. Der Zählwert wird auf die andere Zählgrenze gesetzt. Zum erneuten Starten des Zählvorgangs müssen Sie das SW-Tor oder HW-Tor schließen und erneut öffnen.
Zählen fortsetzen (voreingestellt bei S7-1500)	Nach Überschreiten einer Zählgrenze wird der Zählwert auf die andere Zählgrenze gesetzt und der Zählvorgang wird fortgesetzt.

Flankenauswahl

Mit diesem Parameter legen Sie fest, welche Flanken der jeweilige Zähler zählt:

Flankenauswahl	Bedeutung
Bei steigender Flanke (voreingestellt)	Der jeweilige Zähler zählt alle steigenden Flanken am Digitaleingang.
Bei fallender Flanke	Der jeweilige Zähler zählt alle fallenden Flanken am Digitaleingang.
Bei steigender und fallender Flanke	Der jeweilige Zähler zählt alle Flanken am Digitaleingang.

Zählrichtung

Mit diesem Parameter legen Sie die Zählrichtung des jeweiligen Zählers fest.

Sie können unter folgenden Optionen auswählen:

- Vorwärts
- Rückwärts

Ausgang setzen

Mit diesem Parameter legen Sie die Funktion (Seite 78) fest, mit der das Rückmeldebit STS_DQ gesteuert wird. Sie können das Rückmeldebit STS_DQ verwenden, um einen Digitalausgang eines Digitalausgabemoduls zu steuern.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Aus (DQ = 0)	STS_DQ ist unabhängig vom Zählwert und dauerhaft nicht gesetzt.
Aus (DQ = 1)	STS_DQ ist unabhängig vom Zählwert und dauerhaft gesetzt.
Zwischen Vergleichswert 0 und 1	STS_DQ wird gesetzt, wenn der Zählwert zwischen Vergleichswert 0 und Vergleichswert 1 liegt.
Nicht zwischen Vergleichswert 0 und 1	STS_DQ wird gesetzt, wenn der Zählwert außerhalb des Bereichs zwischen Vergleichswert 0 und Vergleichswert 1 liegt.
Zwischen Vergleichswert und oberer Zählgrenze	STS_DQ wird gesetzt, wenn der Zählwert zwischen dem Vergleichswert und der oberen Zählgrenze liegt.
Zwischen Vergleichswert und unterer Zählgrenze	STS_DQ wird gesetzt, wenn der Zählwert zwischen dem Vergleichswert und der unteren Zählgrenze liegt.

Funktion des DI einstellen

Mit diesem Parameter legen Sie fest, welche Funktion der jeweilige Digitaleingang DI_{n+4} beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Digitaleingang ohne Funktion	Dem jeweiligen Digitaleingang DI_{n+4} ist keine Funktion zugeordnet. Der Signalzustand des DI_{n+4} kann von der CPU über die Rückmeldeschnittstelle gelesen werden.
Torstart/-stopp	Das Setzen des jeweiligen Digitaleingangs DI_{n+4} öffnet das HW-Tor (Seite 77) für DI_n. Das Rücksetzen des jeweiligen Digitaleingangs DI_{n+4} schließt das HW-Tor für DI_n.
Zählrichtung	Der jeweilige Digitaleingang DI_{n+4} bestimmt die Zählrichtung des DI_n, um sie an den Prozess anzupassen. Bei nicht gesetztem DI_{n+4} zählt DI_n vorwärts. Bei gesetztem DI_{n+4} zählt DI_n rückwärts.

Hinweis

Wenn "Zählrichtung" gewählt ist und sich im Prozess die Zählrichtung ändert, wird die Zählflanke automatisch angepasst (konträre Flanken).

Obere Zählgrenze

Mit der Parametrierung der oberen Zählgrenze begrenzen Sie den Zählbereich. Der Maximalwert für die obere Zählgrenze ist modulabhängig:

Obere Zählgrenze	DI 8x24VDC HS, DIQ 16x24VDC/0.5A 8xM12	DI 32x24VDC HF, DI 16x24VDC HF
Maximalwert	2147483647 (2^{31}–1)	4294967295 (2^{32}–1)
Voreinstellung	2147483647	4294967295

Sie müssen einen Wert eingeben, der über der unteren Zählgrenze liegt.

Untere Zählgrenze

Mit der Parametrierung der unteren Zählgrenze begrenzen Sie den Zählbereich. Der Minimalwert für die untere Zählgrenze ist modulabhängig:

Untere Zählgrenze	DI 8x24VDC HS, DIQ 16x24VDC/0.5A 8xM12	DI 32x24VDC HF, DI 16x24VDC HF
Minimalwert	–2147483648 (–2^{31})	0 (nicht parametrierbar)
Voreinstellung	0	0

Sie müssen einen Wert eingeben, der unter der oberen Zählgrenze liegt.

Startwert

Mit der Parametrierung des Startwerts legen Sie fest, mit welchem Wert das Zählen begonnen wird. Sie müssen einen Wert zwischen den Zählgrenzen oder auf den Zählgrenzen eingeben.

Die Voreinstellung ist "0".

Vergleichswert

Mit der Parametrierung eines Vergleichswerts (Seite 78) legen Sie fest, mit welchem Zählwert das Rückmeldebit STS_DQ aufgrund der bei "Ausgang setzen" gewählten Vergleichsfunktion gesteuert wird.

Sie müssen einen Wert eingeben, der größer oder gleich der unteren Zählgrenze sowie kleiner oder gleich der oberen Zählgrenze ist.

Die Voreinstellung ist modulabhängig:

Vergleichswert	DI 8x24VDC HS, DIQ 16x24VDC/0.5A 8xM12	DI 32x24VDC HF, DI 16x24VDC HF
Voreinstellung	10	1

Vergleichswert 0

Mit der Parametrierung eines Vergleichswerts (Seite 78) legen Sie fest, mit welchem Zählwert das Rückmeldebit STS_DQ aufgrund der bei "Ausgang setzen" gewählten Vergleichsfunktion gesteuert wird.

Sie müssen einen Wert eingeben, der größer oder gleich der unteren Zählgrenze sowie kleiner als Vergleichswert 1 ist.

Die Voreinstellung ist "0".

Vergleichswert 1

Mit der Parametrierung des zweiten Vergleichswerts (Seite 78) legen Sie fest, mit welchem weiteren Zählwert das Rückmeldebit STS_DQ aufgrund der bei "Ausgang setzen" gewählten Vergleichsfunktion gesteuert wird.

Sie müssen einen Wert eingeben, der größer als Vergleichswert 0 sowie kleiner oder gleich der oberen Zählgrenze ist.

Die Voreinstellung ist "10".

Prozessalarm: Vergleichsereignis für DQ eingetreten

Mit diesem Parameter legen Sie bei S7-1500 und ET200AL fest, ob bei einem Vergleichsereignis aufgrund der bei "Ausgang setzen" gewählten Vergleichsfunktion ein Prozessalarm erzeugt wird.

In der Voreinstellung ist der Prozessalarm nicht freigegeben.

Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über das Modul erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie mit der rechten Maustaste auf das Modul und wählen Sie "Online & Diagnose".
4. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch des Moduls.

1.2 Zählen und Messen (S7-300, S7-400)

1.2.1 Zählen und Messen

1.2.1.1 Grundlagen zum Zählen und Messen

Was ist Zählen und Messen?

Zählen und Messen

Zählen ist das Erfassen und Aufsummieren von Ereignissen. Die Zählerbaugruppen erfassen Gebersignale (Impulse) und werten diese entsprechend aus.

Bei den Messbetriebsarten werden die Impulse zeitgenau erfasst und daraus die betreffenden Messwerte (z. B. die Frequenz) berechnet.

Grundsätzliches zum Zählen

Zählbereiche und Zählgrenzen

In den beiden Zählbereichen "0 bis +32 Bit" und "-31 bis +31 Bit" zählt die Baugruppe innerhalb verschiedener Zählgrenzen. Jeweils an den Zählgrenzen wird ein Überlauf bzw. Unterlauf erkannt.

Im Zählbereich "-31 bis +31 Bit" wird der Zählerstand im 2er Komplement dargestellt.

Zählbereich		Überlauf	Unterlauf
0 bis +32 Bit [1]	0 bis 4 294 967 295 0 bis FFFF FFFFH	Beim Wechsel des Zählerstandes von 4 294 967 295 nach 0	Beim Wechsel des Zählerstandes von 0 nach 4 294 967 295
-31 bis +31 Bit	–2 147 483 648 bis 2 147 483 647 8000 0000H bis 7FFF FFFFH	Beim Wechsel des Zählerstandes von +2 147 483 647 nach –2 147 483 648	Beim Wechsel des Zählerstandes von –2 147 483 648 nach 2 147 483 647
[1] Dieser Zählbereich ist nur mit FM x50-1 verfügbar. Sie dürfen die Werte nur hexadezimal vorgeben und auswerten.			

Ladewert

Der Ladewert ist der Zählerstand, von dem aus die Baugruppe den Zählvorgang beginnt.

Sie können der Baugruppe im Betrieb einen Ladewert LOAD_VAL vorgeben, der den Startwert überschreibt.

Sie können diesen Ladewert **direkt** vorgeben, dann wird er von der Baugruppe direkt als neuer Zählwert übernommen und vorbereitend geladen.

Sie können den Ladewert nur **vorbereitend** laden. Ein vorbereitend geladener Ladewert wird von der Baugruppe bei folgenden Ereignissen als neuer Zählwert übernommen:

- In allen Zählbetriebsarten
  - Starten des Zählvorgangs durch das abbrechende SW-Tor oder HW-Tor (bei unterbrechender Torfunktion wird beim Start des SW-Tors oder HW-Tors vom aktuellen Zählerstand aus weitergezählt, der Ladewert wird nicht übernommen).
  - Synchronisation (über entsprechenden DI)
  - Latch/Retrigger (über entsprechenden DI)
- In den Zählbetriebsarten "Einmalig Zählen" und "Periodisch Zählen" zusätzlich
  - Erreichen der parametrierten oberen Zählgrenze bei Hauptzählrichtung vorwärts.
  - Erreichen der 0 bei Hauptzählrichtung rückwärts.

Torsteuerung

Mit Hilfe von Hardware-Tor (HW-Tor) und Software-Tor (SW-Tor) können Sie die Zählvorgänge der Baugruppe steuern, d. h. starten und stoppen.

Hauptzählrichtung

Mit der Parametrierung einer Hauptzählrichtung (vorwärts oder rückwärts) grenzen Sie durch Parametrieren einer oberen Zählgrenze den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze. Damit ist es z. B. möglich, inkrementierende oder dekrementierende Zähl-Applikationen zu lösen. Die parametrierte Hauptzählrichtung hat keinen Einfluss auf die Richtungsauswertung bei der Erfassung der Zählimpulse.

Startwerte nach der Parametrierung

Wert	Hauptzählrichtung	Startwert
Ladewert	Keine	0
	Vorwärts	0
	Rückwärts	Parametrierte obere Zählgrenze
Zählwert	Keine	0
	Vorwärts	0
	Rückwärts	Parametrierte obere Zählgrenze
Vergleichswert 1 und 2	Keine	0
	Vorwärts	0
	Rückwärts	Parametrierte obere Zählgrenze
Latchwert	Keine	0
	Vorwärts	0
	Rückwärts	Parametrierte obere Zählgrenze

Kommandos bei den Zählbetriebsarten

Sie können im Betrieb den Zählvorgang der Baugruppe mit Kommandos beeinflussen:

Kommando	Beschreibung
Tor öffnen und schließen	Der Zählvorgang wird über das Öffnen eines Tores gestartet und über das Schließen des Tores beendet.
Zähler setzen [1]	Der Zähler wird über verschiedene Signale (DI, Nullmarke) auf den Ladewert gesetzt.
Latch/Retrigger [2]	Abspeichern des Zählerstandes und Laden des Zählers mit dem Ladewert bei positiver Flanke an DI-Start
Latchen [2]	Abspeichern des Zählerstandes bei positiver Flanke an DI-Start
Messen von Zeiten zwischen zwei Flanken [3]	Messen von Zeiten zwischen zwei unmittelbar aufeinanderfolgenden Flanken am Digitaleingang DI-Start
[1] nur mit FM 350-1 und FM 450-1 [2] nur mit FM 350-1, ET 200S 1Count24V und ET 200S 1Count5V [3] nur mit FM 350-1	

Taktsynchroner Betrieb

Im taktsynchronen Betrieb übernimmt die Baugruppe in jedem PROFIBUS / PROFINET-Zyklus Steuerbits und Steuerwerte aus der Steuerschnittstelle und meldet die Reaktion darauf noch im selben Zyklus zurück.

Die Baugruppe übergibt in jedem PROFIBUS / PROFINET-Zyklus den Zählerstand bzw. Latchwert, wie er zum Zeitpunkt T_i gültig war, und die Statusbits, wie sie zum Zeitpunkt T_i gültig waren.

Ein durch Hardware-Eingangssignale beeinflusster Zählerstand wird nur dann im selben Zyklus übergeben, wenn das Eingangssignal vor dem Zeitpunkt T_i aufgetreten ist.

Zählbetriebsarten

Übersicht über die Zählbetriebsarten

Mit der Vorgabe einer Betriebsart wählen Sie, mit welcher Funktionalität Sie die Baugruppe betreiben wollen. Die folgende Tabelle zeigt einen Überblick über die Zählbetriebsarten.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne SW-Tor oder HW-Tor)	Die Baugruppe zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit SW-Tor oder mit HW-Tor)	Die Baugruppe zählt mit dem Öffnen des Tores ab dem Ladewert bis zur Zählgrenze.
Periodisches Zählen (Seite 187) (mit SW-Tor oder mit HW-Tor)	Die Baugruppe zählt mit dem Öffnen des Tores zwischen dem Ladewert und der Zählgrenze.

Endlos Zählen

Beschreibung

In dieser Betriebsart zählt die Baugruppe ab dem Zählerstand endlos:

- Erreicht der Zähler beim Vorwärtszählen die obere Zählgrenze und kommt ein weiterer Zählimpuls, springt er auf die untere Zählgrenze und zählt von dort ohne Impulsverlust weiter.
- Erreicht der Zähler beim Rückwärtszählen die untere Zählgrenze und kommt ein weiterer Zählimpuls, springt er auf die obere Zählgrenze und zählt von dort ohne Impulsverlust weiter.

Für den Zählbereich ±31 Bit gilt:

- Die obere Zählgrenze ist festgelegt auf +2 147 483 647 (2^{31} - 1).
- Die untere Zählgrenze ist festgelegt auf -2 147 483 648 (-2^{31}).

Für den Zählbereich 32 Bit gilt:

- Die obere Zählgrenze ist festgelegt auf +4 294 967 295 (2^{32} - 1).
- Die untere Zählgrenze ist festgelegt auf 0 (Null).

Verhalten an den Zählgrenzen

Hat der Zähler die obere oder die untere Zählgrenze erreicht und kommt ein weiterer Zählimpuls, dann wird der Zähler auf die andere Zählgrenze gesetzt. In der Rückmeldeschnittstelle und ggf. im DB wird ein entsprechendes Statusbit gesetzt:

Zählgrenze erreicht	Statusbit
Obere Zählgrenze	STS_OFLW wird gesetzt
Untere Zählgrenze	STS_UFLW wird gesetzt

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- Ohne Tor (Default)
- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Endlos Zählen mit Torsteuerung.

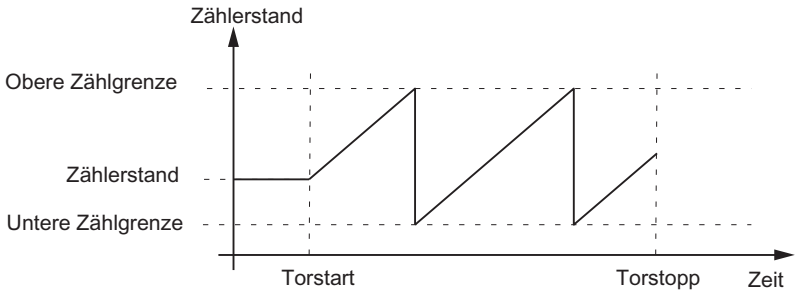

Einmaliges Zählen

Beschreibung

In dieser Betriebsart zählt die Baugruppe je nach parametrierter Hauptzählrichtung einmalig bis zu einer Zählgrenze und stoppt dann den Zählvorgang automatisch.

Verhalten an den Zählgrenzen

Hat der Zähler die obere oder die untere Zählgrenze erreicht und kommt ein weiterer Zählimpuls, dann wird der Zähler

- beim Zählen ohne Hauptzählrichtung (Seite 188) auf die andere Zählgrenze gesetzt,
- beim Zählen mit Hauptzählrichtung (Seite 188) auf den Ladewert gesetzt.

Danach wird das Tor geschlossen und der Zählvorgang beendet, auch wenn der Parameter / das Steuerbit SW_GATE noch gesetzt oder das HW-Tor noch geöffnet ist. In der Rückmeldeschnittstelle / im DB wird das entsprechende Statusbit gesetzt:

Zählgrenze erreicht	Statusbit
Obere Zählgrenze	STS_OFLW wird gesetzt
Untere Zählgrenze	STS_UFLW wird gesetzt

Wenn Sie den Zähler erneut starten wollen, müssen Sie den Parameter / das Steuerbit SW_GATE erneut setzen bzw. das HW-Tor erneut öffnen. Der Zählvorgang wird dann ab dem Ladewert fortgesetzt.

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Einmaliges Zählen mit Ladewert und Torsteuerung.

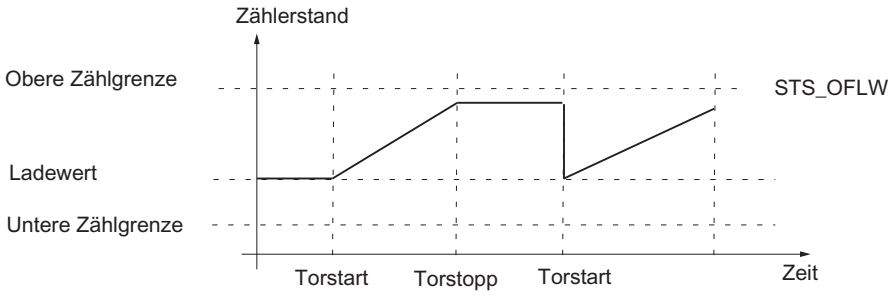

Periodisches Zählen

Beschreibung

In dieser Betriebsart zählt die Baugruppe bei geöffnetem Tor periodisch, ohne an den Zählgrenzen den Zählvorgang zu stoppen.

Verhalten an den Zählgrenzen

Wenn der Zähler

- beim Zählen ohne Hauptzählrichtung (Seite 188) eine der Zählgrenzen erreicht,
- beim Zählen mit Hauptzählrichtung vorwärts (Seite 188) die obere Zählgrenze erreicht,
- beim Zählen mit Hauptzählrichtung rückwärts (Seite 188) die untere Zählgrenze erreicht

und ein weiterer Zählimpuls kommt, dann wird der Zähler auf den Ladewert gesetzt, von dem aus weitergezählt wird. In der Rückmeldeschnittstelle / im DB wird das entsprechende Statusbit gesetzt.

Zählgrenze erreicht	Statusbit
Obere Zählgrenze	STS_OFLW wird gesetzt
Untere Zählgrenze	STS_UFLW wird gesetzt

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Periodisches Zählen mit Ladewert und Torsteuerung.

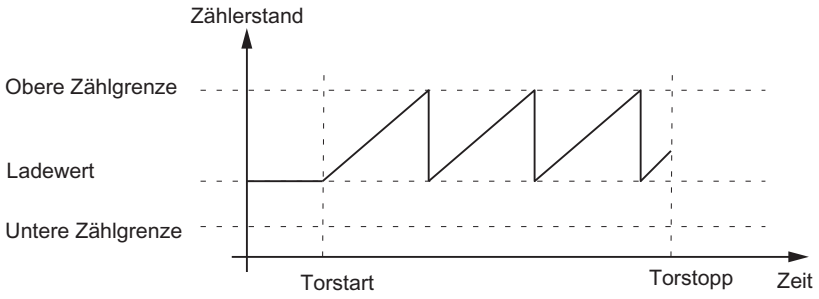

Hauptzählrichtung

Eine Hauptzählrichtung können Sie nur in den Betriebsarten "Einmaliges Zählen", "Periodisches Zählen" und "Dosieren" auswählen.

Sie können folgendes Verhalten parametrieren:

- keine (ohne Hauptzählrichtung)
- vorwärts
- rückwärts

Einmaliges Zählen ohne Hauptzählrichtung

In der Betriebsart "Einmaliges Zählen" ohne Hauptzählrichtung zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis eine der Zählgrenzen überschritten wird.

Beim Überschreiten einer der Zählgrenzen wird

- das Tor geschlossen,
- in der Rückmeldeschnittstelle das Bit STS_OFLW oder STS_UFLW gesetzt,
- der Zähler auf die jeweils andere Zählgrenze gesetzt.

Die Zählgrenzen sind fest auf den maximalen Zählbereich eingestellt.

Beim Zählerstand Null wird das Bit STS_ZERO bzw. STS_ND gesetzt (nicht bei FM 350-2).

Zum erneuten Start des Zählvorgangs müssen Sie das Tor wieder öffnen.

Periodisches Zählen ohne Hauptzählrichtung

In der Betriebsart "Periodisches Zählen" ohne Hauptzählrichtung zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis eine der Zählgrenzen überschritten wird.

Beim Überschreiten einer der Zählgrenzen wird

- in der Rückmeldeschnittstelle das Bit STS_OFLW oder STS_UFLW gesetzt,
- der Zähler auf den Ladewert gesetzt, von dem aus weitergezählt wird.

Die Zählgrenzen sind fest auf den maximalen Zählbereich eingestellt.

Beim Zählerstand Null wird das Bit STS_ZERO bzw. STS_ND gesetzt (nicht bei FM 350-2).

Der Zählvorgang wird bis zum Schließen des Tors fortgesetzt.

Einmaliges Zählen mit Hauptzählrichtung vorwärts

In den Betriebsarten "Einmaliges Zählen" und "Dosieren" mit Hauptzählrichtung vorwärts zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis die obere Zählgrenze überschritten wird.

Beim Überschreiten der oberen Zählgrenze wird

- das Tor geschlossen,
- in der Rückmeldeschnittstelle das Bit STS_OFLW gesetzt,
- der Zähler auf den Ladewert gesetzt.

Die obere Zählgrenze ist parametrierbar, der Ladewert hat seinen Startwert und ist veränderbar.

Zum erneuten Start des Zählvorgangs müssen Sie das Tor wieder öffnen.

Periodisches Zählen mit Hauptzählrichtung vorwärts

In der Betriebsart "Periodisches Zählen" mit Hauptzählrichtung vorwärts zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis die obere Zählgrenze überschritten wird.

Beim Überschreiten der oberen Zählgrenze wird

- in der Rückmeldeschnittstelle das Bit STS_OFLW gesetzt,
- der Zähler auf den Ladewert gesetzt, von dem aus weitergezählt wird.

Die obere Zählgrenze ist parametrierbar, der Ladewert hat seinen Startwert und ist veränderbar.

Der Zählvorgang wird bis zum Schließen des Tors fortgesetzt.

Einmaliges Zählen mit Hauptzählrichtung rückwärts

In den Betriebsarten "Einmaliges Zählen" und "Dosieren" mit Hauptzählrichtung rückwärts zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis die untere Zählgrenze überschritten wird.

Beim Überschreiten der unteren Zählgrenze wird

- das Tor geschlossen,
- in der Rückmeldeschnittstelle das Bit STS_UFLW gesetzt,
- der Zähler auf den Ladewert gesetzt.

Die untere Zählgrenze ist fest auf 0, der Ladewert hat seinen Startwert und ist veränderbar.

Zum erneuten Start des Zählvorgangs müssen Sie das Tor wieder öffnen.

Periodisches Zählen mit Hauptzählrichtung rückwärts

In der Betriebsart "Periodisches Zählen" mit Hauptzählrichtung rückwärts zählt die Baugruppe nach Öffnen des Tors vom Ladewert aus vorwärts oder rückwärts, bis die untere Zählgrenze unterschritten wird.

Beim Unterschreiten der unteren Zählgrenze wird

- in der Rückmeldeschnittstelle das Bit STS_UFLW gesetzt,
- der Zähler auf den Ladewert gesetzt, von dem aus weitergezählt wird.

Die untere Zählgrenze ist fest auf 0, der Ladewert hat seinen Startwert und ist veränderbar.

Der Zählvorgang wird bis zum Schließen des Tors fortgesetzt.

Grundsätzliches zum Messen

Messbereiche

In den Messbetriebsarten gibt es für die Baugruppen die folgenden Messbereiche:

Betriebsart	Messbereich
Frequenzmessung	0,1 Hz [1] bis 650 000 Hz (je nach Baugruppe)
Drehzahlmessung	bis 25 000 /min
Periodendauermessung	bis 120 s

[1] Die untere Grenze von 0,1 Hz ist durch die maximale Aktualisierungszeit von 10 s bedingt. Die Auflösung des Wertes beträgt 0,001 Hz.

Aktualisierungszeit / Integrationszeit

Die Baugruppe aktualisiert die Messwerte zyklisch. Für die Berechnung der Aktualisierungszeit / Integrationszeit geben Sie den Parameter "Messzeit" als Faktor n vor.

Berechnung der Aktualisierungszeit / Integrationszeit:

Randbedingungen		Aktualisierungszeit / Integrationszeit	Wertebereich von n	
			n_{min}	n_{max}
Nicht taktsynchroner Betrieb	T_{DP} beliebig	n × 10 ms	1	1000 [3]
Taktsynchroner Betrieb [2]	T_{DP} < 10 ms	n × T_{DP}	(10 ms/T_{DP} [ms]) + 1 [1]	1000 [3]
	T_{DP} ≥ 10 ms	n × T_{DP}	1	10000 [3] ms/T_{DP} [ms] [1]

[1] Die bei der Division durch T_{DP} entstehenden Nachkommastellen entfallen.

Diese Grenzen dürfen nicht verletzt werden. Bei Verletzung dieser Grenzen erzeugt die Baugruppe einen Parametrierfehler und geht nicht in den taktsynchronen Betrieb.

[2] nicht mit FM 350-2

[3] bei Periodendauermessung: 120000

Nach Ablauf der Aktualisierungszeit / Integrationszeit wird der Messwert in der Rückmeldeschnittstelle aktualisiert und das Bit STS_CMP1 bzw. STS_COMP1 gesetzt.

Messprinzip

Die Baugruppe zählt jede steigende Flanke eines Impulses und ordnet ihr einen Zeitwert in µs zu.

Die dynamische Messzeit ist definiert als Differenz zwischen zwei Zeitwerten.

Bei einer Impulsfolge mit einem oder mehreren Impulsen pro Aktualisierungszeitintervall gilt:

Dynamische Messzeit =	Zeitwert des letzten Impulses im aktuellen Aktualisierungszeitintervall minus Zeitwert des letzten Impulses im vorangegangenen Aktualisierungszeitintervall

Wenn nach Berechnung der dynamischen Messzeit Aktualisierungszeitintervalle ohne Impulse folgen, dann verlängert sich die Messzeit um diese Aktualisierungszeitintervalle. Wenn der Wert "1 Impuls pro dynamische Messzeit" kleiner als der letzte Messwert ist, dann wird dieser Wert als neuer Wert ausgegeben. Das folgende Bild zeigt das Messprinzip.

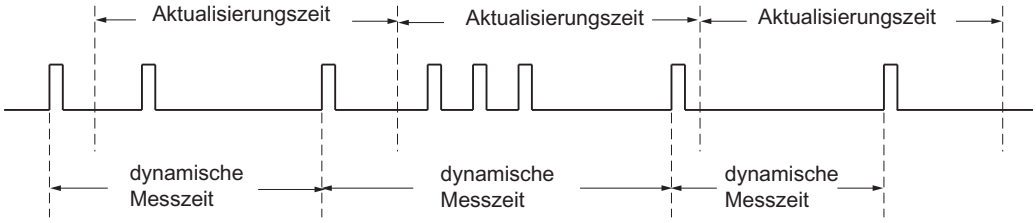

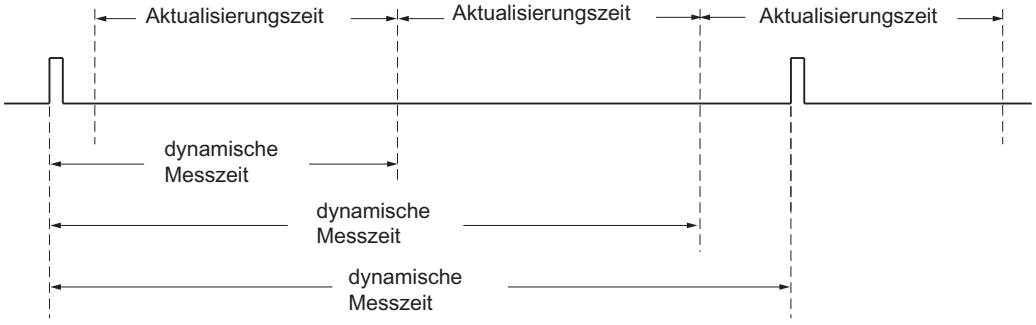

Ablauf der Messung

Die Baugruppe misst kontinuierlich. Sie geben beim Parametrieren eine Aktualisierungszeit vor.

In der Zeit bis zum Ende der ersten abgelaufenen Aktualisierungszeit wird der Wert "-1" zurückgemeldet. Die erste Aktualisierungszeit beginnt mit dem Öffnen des Tores.

Die kontinuierliche Messung beginnt nach dem Öffnen des Tores mit dem ersten Impuls der zu messenden Impulsfolge. Der erste Messwert wird frühestens nach dem zweiten Impuls berechnet.

Nach jedem Ablauf der Aktualisierungszeit wird in der Rückmeldeschnittstelle ein Messwert ausgegeben (Frequenz, Periodendauer oder Drehzahl). Dabei wird das Messende mit den Statusbits STS_CMP1 bzw. STS_COMP1 (bei FM 350-2) gemeldet. Zurückgesetzt wird dieses Bit durch vollständiges Quittungsprinzip mit den Bits RES_ZERO und STS_RES_ZERO bzw. RES_STS und RES_STS_A.

Wenn in einer Aktualisierungszeit eine Drehrichtungsumkehr auftritt, ist der Messwert für diese Messperiode unbestimmt. Durch Auswertung der Rückmeldebits für die Richtung können Sie auf eine eventuelle Prozessunregelmäßigkeit reagieren.

Das folgende Bild zeigt das Prinzip der kontinuierlichen Messung am Beispiel einer Frequenzmessung.

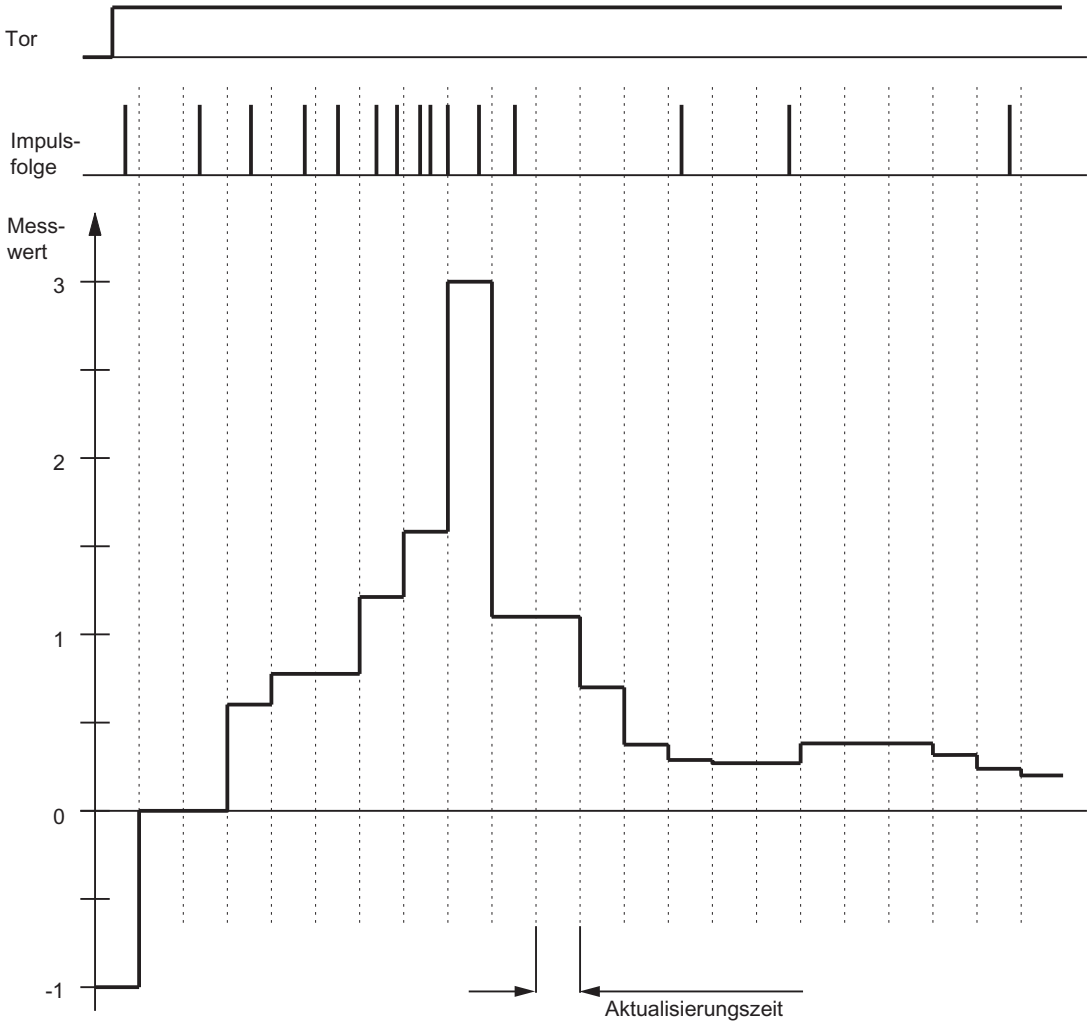

Grenzwertüberwachung

Die Grenzwerte müssen jede der folgenden Bedingungen erfüllen:

- Unterer Grenzwert ≥ untere Messbereichsgrenze
- Oberer Grenzwert ≤ obere Messbereichsgrenze
- Oberer Grenzwert ≥ (unterer Grenzwert + 1)

Nach jeder abgelaufenen Aktualisierungszeit wird der ermittelte Messwert (Frequenz, Drehzahl oder Periodendauer) mit den parametrierten unteren und oberen Grenzwerten verglichen. Der Vergleich liefert folgendes Ergebnis:

Wenn der ermittelte Messwert ...	...dann wird folgendes Bit im Status gesetzt:
größer als der obere Grenzwert ist,	STS_OFLW
kleiner als der untere Grenzwert ist,	STS_UFLW

Zusätzlich kann ein Prozessalarm erzeugt werden.

Das folgende Bild zeigt den Zustand der Statusbits für die Grenzwertüberwachung bei den Messbetriebsarten.

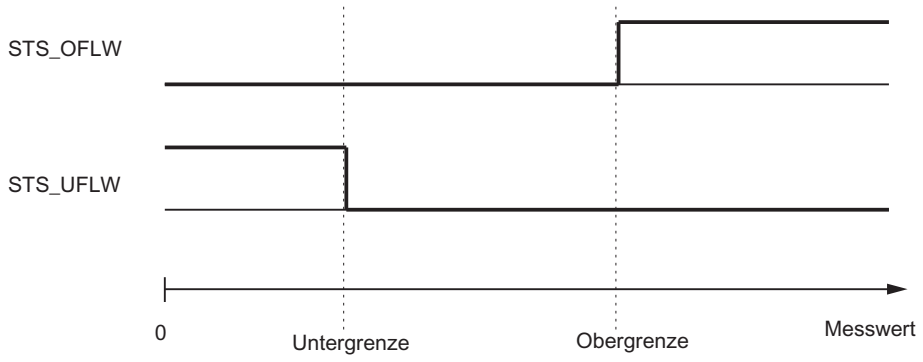

Die Bits STS_OFLW und STS_UFLW müssen Sie mit den Bits RES_ZERO und STS_RES_ZERO bzw. RES_STS durch vollständiges Quittungsprinzip rücksetzen. Befindet sich nach dem Quittieren der Messwert noch oder wieder außerhalb der Grenzen, wird das entsprechende Statusbit erneut gesetzt.

Bei entsprechender Parametrierung können Sie mit Hilfe der Grenzwertüberwachung auch den entsprechenden Ausgang DOn schalten.

Torsteuerung

Mit Hilfe von Hardware-Tor (HW-Tor) und Software-Tor (SW-Tor) können Sie die Messvorgänge der Baugruppe steuern, d. h. starten und stoppen.

Kommandos bei den Messbetriebsarten

Sie können im Betrieb den Messvorgang der Baugruppe mit Kommandos beeinflussen:

Bezeichnung	Beschreibung
Tor öffnen und schließen	Der Messvorgang wird über das Öffnen eines Tores gestartet und über das Schließen des Tores beendet.

Taktsynchroner Betrieb

Im taktsynchronen Betrieb übernimmt die Baugruppe in jedem PROFIBUS / PROFINET-Zyklus zum Zeitpunkt T_o die Steuersignale aus der Steuerschnittstelle. Dadurch werden alle Steuerungen taktsynchron ausgeführt und zum Zeitpunkt T_o wirksam. Die Reaktion auf die Steuerung wird noch im selben PROFIBUS / PROFINET-Zyklus zurückgemeldet.

Die Baugruppe liefert in jedem PROFIBUS / PROFINET-Zyklus zum Zeitpunkt T_i einen Messwert und die Statusbits.

Die Messung beginnt und endet jeweils zum Zeitpunkt T_i.

Hinweis

Da Sie die Aktualisierungszeit beim nicht taktsynchronen Betrieb in ganzzahligen Vielfachen von 10 ms, beim taktsynchronen Betrieb aber in ganzzahligen Vielfachen der Buszykluszeit angeben, müssen Sie bei einem Wechsel zwischen den beiden Betriebsarten immer auch den Parameter Aktualisierungszeit anpassen, wenn Sie die tatsächliche Aktualisierungszeit beibehalten wollen.

Messbetriebsarten

Übersicht über die Messbetriebsarten

Mit der Vorgabe einer Betriebsart wählen Sie, mit welcher Funktionalität Sie die Baugruppe betreiben wollen. Die folgende Tabelle zeigt einen Überblick über die Messbetriebsarten.

Betriebsart	Beschreibung
Frequenzmessung (Seite 195)	Die Baugruppe zählt die Impulse, die in einer dynamischen Messzeit [1] eintreffen und errechnet daraus die Frequenz.
Drehzahlmessung (Seite 196)	Die Baugruppe zählt die Impulse, die in einer dynamischen Messzeit [1] von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.
Periodendauermessung (Seite 198)	Die Baugruppe gibt die dynamische Messzeit [1] als Periodendauer an. Wenn die Periodendauer kleiner als die Aktualisierungszeit ist, dann wird für die Periodendauer ein Mittelwert gebildet.

[1] Dynamische Messzeit bedeutet: Werden vor dem Ablauf der parametrierten Messzeit keine 2 Flanken am Messsignal erkannt, so wird ein geschätzter Messwert mit der dynamischen Messzeit errechnet. Dabei wird angenommen, dass ein Impuls am Ende der Aktualisierungszeit aufgetreten ist. Sie bekommen dadurch einen stetigen Messwert.

Hinweis

Die FM 450-1 unterstützt keine Messbetriebsarten.

Frequenzmessung

Beschreibung

In dieser Betriebsart zählt die Baugruppe die Impulse, die in einer dynamischen Messzeit bzw. einer vorgegebenen Integrationszeit eintreffen. In Verbindung mit den Zeitstempeln der Flanken wird daraus die Frequenz errechnet.

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Frequenzmessung mit Torsteuerung.

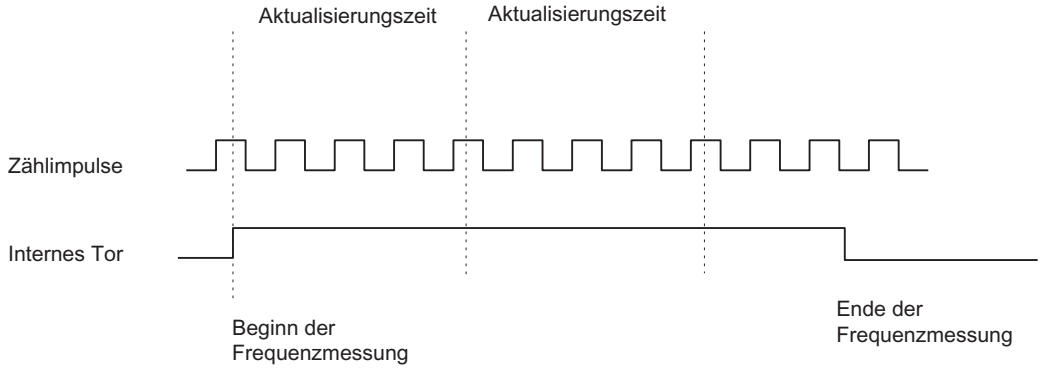

Mögliche Messbereiche mit Fehlerangaben

Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für kontinuierliche Frequenzmessungen.

Frequenz f	absoluter Fehler	
	FM 350-1 ET 200S 1Count5V	ET 200S 1Count24V
0,1 Hz	± 0,001 Hz	
1 Hz	± 0,001 Hz	
10 Hz	± 0,003 Hz	

Frequenz f	absoluter Fehler	
	FM 350-1 ET 200S 1Count5V	ET 200S 1Count24V
100 Hz	± 0,02 Hz	
1 000 Hz	± 0,18 Hz	
10 000 Hz	± 1,8 Hz	
100 000 Hz	± 18 Hz	
500 000 Hz	± 90 Hz	–

Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für Frequenzmessungen mit Integrationszeit.

Integrationszeit	f_{min} ± absoluter Fehler	f_{max} ± absoluter Fehler	
		ET 200S 1Count24V	ET 200S 1Count5V
10 s	0,1 Hz ± 0,001 Hz	100 000 Hz ± 18 Hz	500 000 Hz ± 90 Hz
1 s	1 Hz ± 0,001 Hz	100 000 Hz ± 11 Hz	500 000 Hz ± 55 Hz
0,1 s	10 Hz ± 0,002 Hz	100 000 Hz ± 10 Hz	500 000 Hz ± 52 Hz
0,01 s	100 Hz ± 0,013 Hz	100 000 Hz ± 13 Hz	500 000 Hz ± 63 Hz

Veränderbare Werte während des Betriebs

Während des Betriebs können Sie folgende Werte ändern:

- Unterer Grenzwert
- Oberer Grenzwert
- Aktualisierungszeit (Messzeit) [1)]
- Funktion des Digitalausgangs [1)]

[1)] nicht bei FM 350-2

Ergebnis

Das Ende einer Frequenzmessung wird mit dem Statusbit STS_CMP1 oder STS_COMP1 (bei FM 350-2) gemeldet. Der Wert der ermittelten Frequenz wird in der Einheit 10^{-3} Hz zur Verfügung gestellt. Den gemessenen Frequenzwert können Sie in der Rückmeldeschnittstelle lesen.

Drehzahlmessung

Beschreibung

In dieser Betriebsart zählt die Baugruppe die Impulse, die in einer dynamischen Messzeit bzw. einer vorgegebenen Integrationszeit von einem Drehzahlgeber eintreffen und misst den Zeitabstand zwischen den Flanken. Hieraus berechnet sie mit den Impulsen pro Geberumdrehung die Drehzahl.

Für die Betriebsart "Drehzahlmessung" müssen Sie zusätzlich die Impulse pro Geberumdrehung parametrieren.

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Drehzahlmessung mit Torsteuerung.

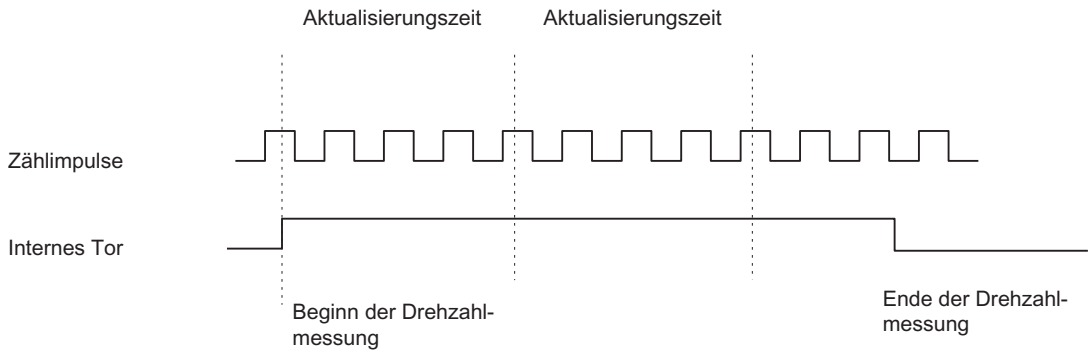

Mögliche Messbereiche mit Fehlerangaben

Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für kontinuierliche Drehzahlmessungen bei Anzahl der Impulse pro Geberumdrehung = 60.

Drehzahl n	absoluter Fehler
1 /min	± 0,04 /min
10 /min	± 0,04 /min
100 /min	± 0,05 /min
1 000 /min	± 0,21 /min
10 000 /min	± 1,82 /min
25 000 /min	± 4,50 /min

Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für Drehzahlmessungen mit Integrationszeit bei Anzahl der Impulse pro Geberumdrehung = 60.

Integrationszeit	n_{min} ± absoluter Fehler	n_{max} ± absoluter Fehler
10 s	1 /min ± 0,03 /min	25000 /min ± 4,5 /min
1 s	1 /min ± 0,03 /min	25000 /min ± 2,8 /min
0,1 s	10 /min ± 0,03 /min	25000 /min ± 2,6 /min
0,01 s	100 /min ± 0,04 /min	25000 /min ± 3,2 /min

Veränderbare Werte während des Betriebs

Während des Betriebs können Sie folgende Werte ändern:

- Unterer Grenzwert
- Oberer Grenzwert

- Aktualisierungszeit (Messzeit) [1)]
- Funktion des Digitalausgangs [1)]

[1)] nicht bei FM 350-2

Ergebnis

Das Ende einer Drehzahlmessung wird mit dem Statusbit STS_CMP1 oder STS_COMP1 (bei FM 350-2) gemeldet. Der Wert der ermittelten Drehzahl wird in der Einheit 10^{-3} /min zur Verfügung gestellt. Die gemessene Drehzahl können Sie in der Rückmeldeschnittstelle lesen.

Periodendauermessung

Beschreibung

In dieser Betriebsart gibt die Baugruppe die dynamische Messzeit als Periodendauer an. Wenn die Periodendauer kleiner als die Aktualisierungszeit ist, dann wird für die Periodendauer ein Mittelwert gebildet.

Das 1Count24V misst in der Betriebsart "Periodendauermessung mit Integrationszeit" die Zeit zwischen zwei positiven Flanken des Zählsignals, indem es die Impulse einer internen quarzgenauen Referenzfrequenz (16 MHz) in einer vorgegebenen Integrationszeit zählt.

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- HW-Tor (Seite 205), pegel- oder flankengesteuert

Das folgende Bild zeigt ein Beispiel für Periodendauermessung mit Torsteuerung.

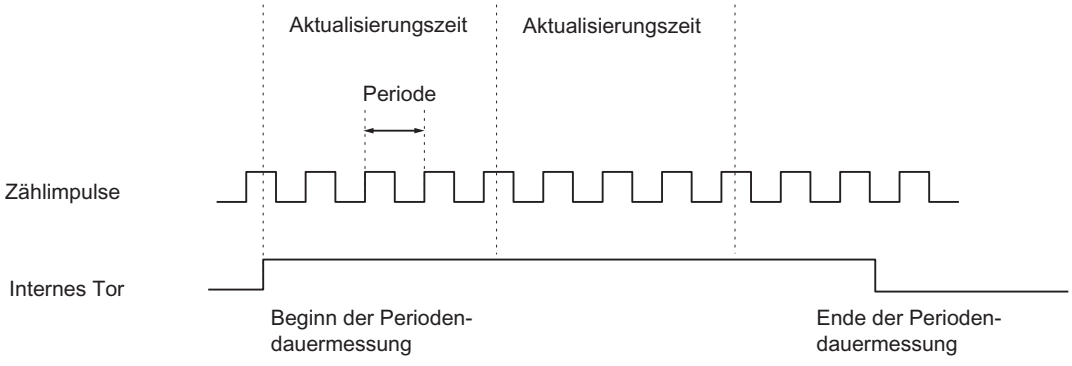

Mögliche Messbereiche mit Fehlerangaben

Kontinuierliche Periodendauermessung:

- Bei kontinuierlichen Periodendauermessungen mit einer **Auflösung von 1 µs** beträgt der absolute Fehler 0,1 ‰ des Messwertes.
- Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für kontinuierliche Periodendauermessungen mit einer **Auflösung von 1/16 µs**.

Periodendauer T_{min} ± absoluter Fehler
1/16 µs × (160 ± 1)
1/16 µs × (1 600 ± 1)
1/16 µs × (16 000 ± 3)
1/16 µs × (160 000 ± 20)
1/16 µs × (1 600 000 ± 160)
1/16 µs × (16 000 000 ± 1 600)
1/16 µs × (160 000 000 ± 16 000)
1/16 µs × (1 600 000 000 ± 160 000)

Periodendauermessungen mit Integrationszeit:

- Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für Periodendauermessungen mit Integrationszeit mit einer **Auflösung von 1 µs**.

Integrationszeit	T_{min} ± absoluter Fehler	T ± absoluter Fehler
100 s	1 µs × (10 ± 0)	1 µs × (100 000 000 ± 10 000)
10 s	1 µs × (10 ± 0)	1 µs × (10 000 000 ± 1 000)
1 s	1 µs × (10 ± 0)	1 µs × (1 000 000 ± 100)
0,1 s	1 µs × (10 ± 0)	1 µs × (100 000 ± 10)
0,01 s	1 µs × (10 ± 0)	1 µs × (10 000 ± 1)

- Die folgende Tabelle zeigt die Messbereiche und absoluten Fehler für Periodendauermessungen mit Integrationszeit mit einer **Auflösung von 1/16 µs**.

Integrationszeit	T_{min} ± absoluter Fehler	T ± absoluter Fehler
100 s	1/16 µs × (160 ± 0)	1/16 µs × (1 600 000 000 ± 160 000)
10 s	1/16 µs × (160 ± 0)	1/16 µs × (160 000 000 ± 16 000)
1 s	1/16 µs × (160 ± 0)	1/16 µs × (16 000 000 ± 1 600)
0,1 s	1/16 µs × (160 ± 0)	1/16 µs × (1 600 000 ± 160)
0,01 s	1/16 µs × (160 ± 0)	1/16 µs × (160 000 ± 16)

Veränderbare Werte während des Betriebs

Während des Betriebs können Sie folgende Werte ändern:

- Unterer Grenzwert
- Oberer Grenzwert

- Aktualisierungszeit (Messzeit) [1)]
- Funktion des Digitalausgangs [1)]

[1)] nicht bei FM 350-2

Ergebnis

Das Ende einer Periodendauermessung wird mit dem Statusbit STS_CMP1 oder STS_COMP1 (bei FM 350-2) gemeldet. Der Wert der ermittelten Periodendauer wird in der Einheit 1 µs oder 1/16 µs zur Verfügung gestellt. Die gemessene Periodendauer können Sie in der Rückmeldeschnittstelle lesen.

Dosieren (nur mit FM 350-2)

Beschreibung

In dieser Betriebsart werden jeweils vier Zählkanäle auf der FM 350-2 zu einem Dosierkanal zusammengefasst. Bei geöffnetem Tor zählt die FM 350-2 in der Hauptzählrichtung einmalig:

- In Hauptzählrichtung vorwärts zwischen 0 und der parametrierten oberen Zählgrenze. Wenn Sie Hauptzählrichtung vorwärts parametrieren, liegt der Startwert auf 0 und Sie geben den Endwert vor.
- In Hauptzählrichtung rückwärts zwischen der parametrierten oberen Zählgrenze und 0. Wenn Sie Hauptzählrichtung rückwärts parametrieren, geben Sie den Startwert vor und der Endwert liegt auf 0.

Torsteuerung auswählen

Sie können in dieser Betriebsart die Torsteuerung auswählen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- SW-Tor (Seite 204)
- SW-Tor (Seite 204) und HW-Tor (Seite 205)
  Sie können zum SW-Tor ein HW-Tor hinzuschalten. Beide Tore wirken gemeinsam wie eine logische UND-Verknüpfung, d. h. die FM 350-2 zählt nur, wenn beide Tore geöffnet sind.

Verhalten an den Zählgrenzen, SW-Tor

Hauptzählrichtung vorwärts: Hat der Zähler den Wert "Obere Zählgrenze – 1" erreicht und kommt dann ein weiterer Zählimpuls, so wird der Zähler auf 0 gesetzt, das interne Tor geschlossen und der Zählvorgang beendet, auch wenn das Bit SW_GATE0, 4 noch gesetzt ist. Das entsprechende Statusbit STS_OFLW0, 4 wird gesetzt. Die obere Zählgrenze selbst wird also nie erreicht.

Hauptzählrichtung rückwärts: Hat der Zähler den Wert "1" erreicht und kommt dann ein weiterer Zählimpuls, so wird der Zähler auf die obere Zählgrenze gesetzt, das interne Tor geschlossen und der Zählvorgang beendet, auch wenn das Bit SW_GATE0, 4 noch gesetzt ist. Das entsprechende Statusbit STS_UFLW0, 4 wird gesetzt. Der Wert "0" wird also nie erreicht.

Wenn Sie den Zähler erneut starten wollen, dann müssen Sie das Bit SW_GATE0, 4 rücksetzen und erneut setzen.

Verhalten an den Zählgrenzen, HW-Tor

Hauptzählrichtung vorwärts: Hat der Zähler den Wert "Obere Zählgrenze – 1" erreicht und kommt dann ein weiterer Zählimpuls, so wird der Zähler auf 0 gesetzt, das interne Tor geschlossen und der Zählvorgang beendet, auch wenn das Bit SW_GATE0, 4 und der Eingang I0, I4 noch gesetzt sind. Die obere Zählgrenze selbst wird also nie erreicht. Das entsprechende Statusbit STS_OFLW0, 4 wird gesetzt.

Hauptzählrichtung rückwärts: Hat der Zähler den Wert "1" erreicht und kommt dann ein weiterer Zählimpuls, so wird der Zähler auf die obere Zählgrenze gesetzt, das interne Tor geschlossen und der Zählvorgang beendet, auch wenn das Bit SW_GATE0, 4 und der Eingang I0, I4 noch gesetzt sind. Der Wert "0" wird also nie erreicht. Das entsprechende Statusbit STS_UFLW0, 4 wird gesetzt.

Wenn Sie den Zähler erneut starten wollen, dann müssen Sie den Eingang I0, I4 rücksetzen und wieder setzen. Einen neuen Zählvorgang können Sie nur mit dem HW-Tor starten.

Vergleichswerte

Sie können für jeden Dosierkanal der FM 350-2 innerhalb des parametrierten Zählbereichs vier Vergleichswerte parametrieren. Weiterhin können Sie parametrieren, ob und unter welchen Bedingungen im Zusammenhang mit einem Vergleichswert ein Digitalausgang gesetzt wird und / oder ein Prozessalarm ausgelöst werden soll. Sie können hierfür folgende Bedingungen einstellen:

- Wenn der aktuelle Zählerstand mit einem Vergleichswert übereinstimmt, dann wird ein Prozessalarm ausgelöst.
- Wenn der aktuelle Zählerstand größer oder gleich einem Vergleichswert ist, dann wird ein Digitalausgang gesetzt.
- Wenn der aktuelle Zählerstand kleiner oder gleich einem Vergleichswert ist, dann wird ein Digitalausgang gesetzt.

Ein Digitalausgang wird nur gesetzt, wenn Sie den entsprechenden Ausgang mit dem Bit CTRL_DQ0…7 freigegeben haben.

Hinweis

Ausgänge werden im RUN der CPU nur dann gesetzt, wenn das Tor geöffnet ist.

Das folgende Bild zeigt ein Beispiel für Dosieren in Hauptzählrichtung rückwärts mit Vergleichswerten.

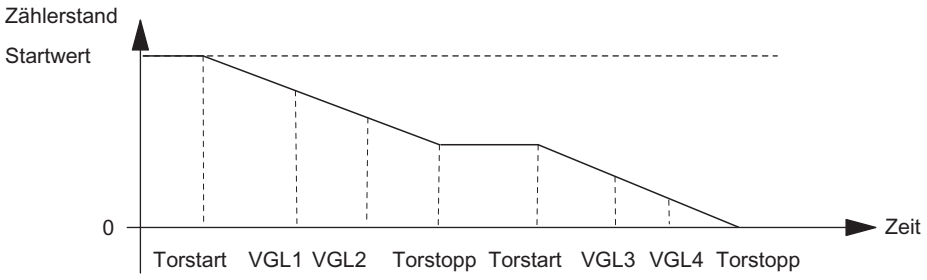

Ladewert vorbereitend

Innerhalb des parametrierten Zählbereichs können Sie im RUN der CPU einen Ladewert vorgeben. Dieser Wert wird vom Zähler bei folgenden Ereignissen als neuer Startwert übernommen:

- Erreichen der oberen Zählgrenze bei Hauptzählrichtung vorwärts.
- Erreichen der 0 bei Hauptzählrichtung rückwärts.
- Abbrechen des Zählvorgangs durch ein SW-Tor oder HW-Tor (beim Unterbrechen des Zählvorgangs wird der Ladewert nicht übernommen).

Der Ladewert ist dann der neue Startwert, ab dem das nächste und alle weiteren Dosieren beginnen. Das parametrierte Ausgangs- und Alarmverhalten bleibt erhalten.

Ladewert direkt

Sie können den aktuellen Zählerstand im RUN der CPU ändern. Der neue Zählerstand wird vom Zähler direkt als aktueller Dosierwert übernommen.

Wertebereich für Ladewerte

Der Wertebereich für Ladewerte ist abhängig von der parametrierten Hauptzählrichtung. Er beträgt:

- 0 bis obere Zählgrenze – 2 in Hauptzählrichtung vorwärts
- Obere Zählgrenze bis 2 in Hauptzählrichtung rückwärts

Torsteuerung

Torfunktionen

Zählen mit Torfunktionen

Viele Anwendungen erfordern, dass der Zählvorgang erst ab einem definierten Zeitpunkt, abhängig von anderen Ereignissen, gestartet oder gestoppt werden soll. Dieses Starten und Stoppen des Zählvorgangs geschieht bei den Zählerbaugruppen über eine Torfunktion. Wenn das Tor geöffnet wird, gelangen Zählimpulse zum Zähler, der Zählvorgang wird gestartet. Wenn das Tor geschlossen wird, gelangen keine Zählimpulse mehr zum Zähler, der Zählvorgang ist gestoppt.

Software-Tor und Hardware-Tor

Die Zählerbaugruppen besitzen zwei Tore:

- Ein Software-Tor (Seite 204) (SW-Tor), das mit dem Eingangsparameter / Steuerbit SW_GATE gesteuert wird.
- Ein Hardware-Tor (Seite 205) (HW-Tor), das über entsprechende Signale an den Digitaleingängen der Baugruppe gesteuert wird.

Beachten Sie auch die Informationen zur Funktion der Tore bei Taktsynchronität (Seite 206).

Internes Tor

Das interne Tor ist die logische UND-Verknüpfung von HW-Tor und SW-Tor. Der Zustand des internen Tores wird mit dem Bit STS_GATE angezeigt.

Falls kein HW-Tor parametriert wurde, ist nur die Einstellung des SW-Tors maßgeblich. Über das interne Tor wird der Zählvorgang aktiviert, unterbrochen, fortgesetzt und abgebrochen. Das interne Tor kann in den Betriebsarten "Einmaliges Zählen" und "Dosieren" auch durch zählerstandsabhängige Ereignisse geschlossen werden.

HW-Tor	SW-Tor	internes Tor	Zählvorgang
offen	offen	offen	aktiv
offen	geschlossen	geschlossen	inaktiv
geschlossen	offen	geschlossen	inaktiv
geschlossen	geschlossen	geschlossen	inaktiv

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Bei abbrechender Wirkung (Seite 206) beginnt der Zählvorgang nach Torstopp und erneutem Torstart wieder von vorne.
- Bei unterbrechender Wirkung (Seite 206) wird der Zählvorgang nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Beispiel

Mit dem Setzen des Torsignals wird das Tor geöffnet und die Zählimpulse werden gezählt. Wenn das Torsignal weggenommen wird, dann wird das Tor geschlossen und die Zählimpulse werden nicht mehr vom Zähler erfasst. Der Zählerstand bleibt konstant.

Das folgende Bild zeigt das Öffnen und Schließen eines Tores und das Zählen der Impulse.

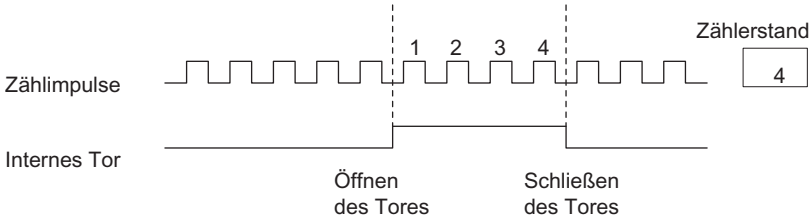

Zählvorgang mit Torstoppfunktion beenden

Sie können bei den Baugruppen FM x50-1 den Zählvorgang unabhängig von den angelegten Signalen oder dem Zustand des SW-Tores mit der Torstoppfunktion beenden. Hierzu setzen Sie den Eingangsparameter GATE_STP.

Wenn Sie diesen Parameter zurücksetzen, dann können Sie das Tor erst wieder durch eine positive Flanke am Digitaleingang "DI-Start" (HW-Tor) oder durch erneutes Setzen des Eingangsparameters SW_GATE (SW-Tor) öffnen.

Software-Tor

Sie öffnen und schließen das SW-Tor mit dem Eingangsparameter / Steuerbit SW_GATE.

Öffnen und Schließen des SW-Tores

Das folgende Bild zeigt das Öffnen und Schließen des SW-Tores. Es darf kein HW-Tor parametriert sein.

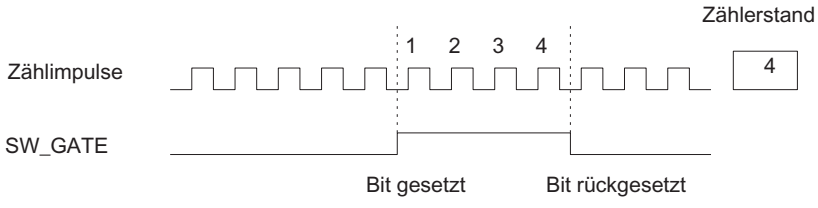

Das SW-Tor wird:

- geöffnet durch Setzen des Eingangsparameters / Steuerbits SW_GATE.
- geschlossen durch Rücksetzen des Eingangsparameters / Steuerbits SW_GATE.

Das geschlossene Tor kann durch erneutes Setzen des Eingangsparameters SW_GATE wieder geöffnet werden.

Hardware-Tor

Sie öffnen und schließen das HW-Tor, indem Sie an den Digitaleingängen die entsprechenden Signale anlegen bzw. wegnehmen.

Gleichzeitig muss auch das SW-Tor geöffnet sein.

Pegelgesteuertes Öffnen und Schließen des HW-Tores

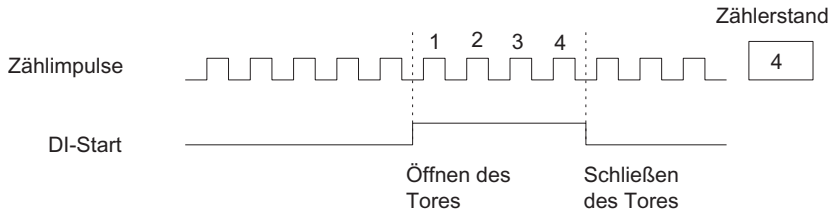

Solange der Digitaleingang gesetzt ist, gelangen die Zählimpulse zum Zähler und werden gezählt. Wenn der Digitaleingang zurückgesetzt wird, dann wird das Tor geschlossen. Die Zählimpulse werden nicht mehr gezählt, der Zähler bleibt stehen.

Wenn das Tor durch Überlauf oder Unterlauf geschlossen wird, dann müssen Sie zuerst den Digitaleingang zurücksetzen und anschließend wieder setzen, um das Tor wieder zu öffnen.

Das pegelgesteuerte HW-Tor wirkt mit der ersten positiven Flanke am Digitaleingang nach der Parametrierung.

ET 200S 1Count24V und 1Count5V:

Das 1Count24V / 1Count5V besitzt ein Hardware-Tor (HW-Tor), das über den Digitaleingang auf dem Modul gesteuert wird. Sie parametrieren das Hardware-Tor als Funktion des Digitaleingangs (Funktion DI "HW-Tor"). Das Tor wird bei einer positiven Flanke am Digitaleingang geöffnet und bei einer negativen Flanke geschlossen.

Flankengesteuertes Öffnen und Schließen des HW-Tores

Flankengesteuertes Öffnen und Schließen des HW-Tores ist nur bei FM x50-1 möglich.

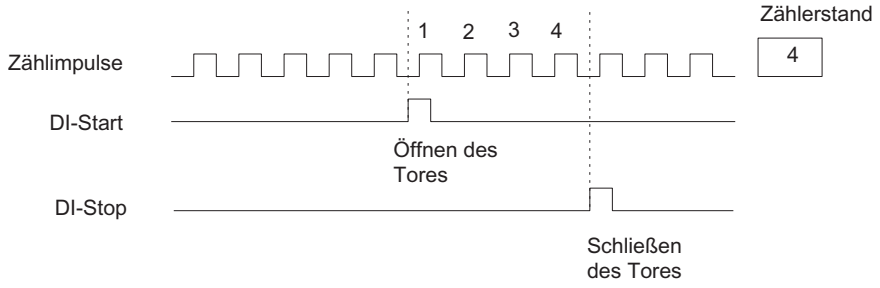

Bei flankengesteuertem Tor erfolgt das Öffnen des HW-Tores durch eine positive Flanke am Digitaleingang "DI-Start". Das Tor wird geschlossen durch eine positive Flanke am Digitaleingang "DI-Stop".

Bei gleichzeitigen positiven Flanken an beiden Eingängen wird ein geöffnetes Tor geschlossen bzw. bleibt ein geschlossenes Tor geschlossen. Wenn der Digitaleingang "DI-Stop" gesetzt ist, dann kann eine positive Flanke am Digitaleingang "DI-Start" das Tor nicht öffnen.

Abbrechende und unterbrechende Torfunktion

Sie können bei der Parametrierung der Torfunktion festlegen, ob das Tor den Zählvorgang abbrechen oder unterbrechen soll.

Abbrechende Torfunktion

Bei abbrechender Torfunktion beginnt der Zählvorgang nach Schließen des Tors und erneutem Torstart wieder mit dem Ladewert (Zeitpunkt ① im nachfolgenden Bild).

Das folgende Bild zeigt ein Beispiel für Endlos Zählen rückwärts mit abbrechender Torfunktion.

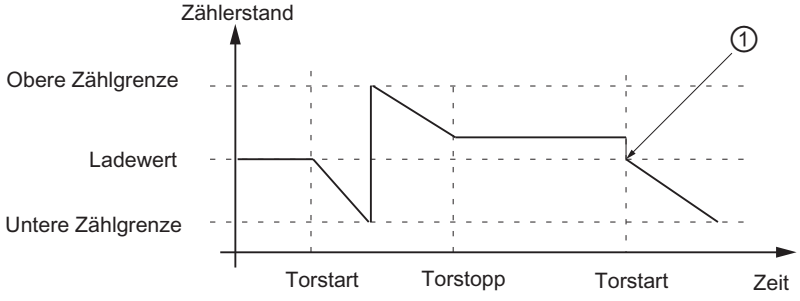

Unterbrechende Torfunktion

Bei unterbrechender Torfunktion wird der Zählvorgang nach Schließen des Tors und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt (Zeitpunkt ① im nachfolgenden Bild).

Das folgende Bild zeigt ein Beispiel für Endlos Zählen vorwärts mit unterbrechender Torfunktion.

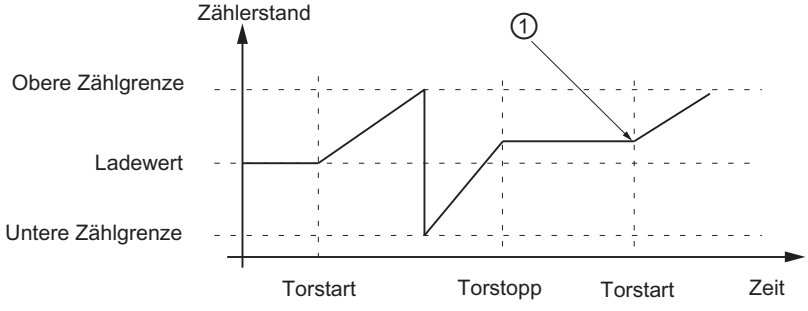

Torsteuerung bei Taktsynchronität

Bei Taktsynchronität funktioniert die Torsteuerung wie folgt:

Torsteuerung mit SW-Tor

Die Steuerung des SW-Tores erfolgt durch Setzen und Rücksetzen des Steuerbits SW_GATE. Der Zählvorgang beginnt und endet dann zum Zeitpunkt T_o des jeweils nächsten PROFIBUS DP-Zyklus nach der Änderung des Steuerbits:

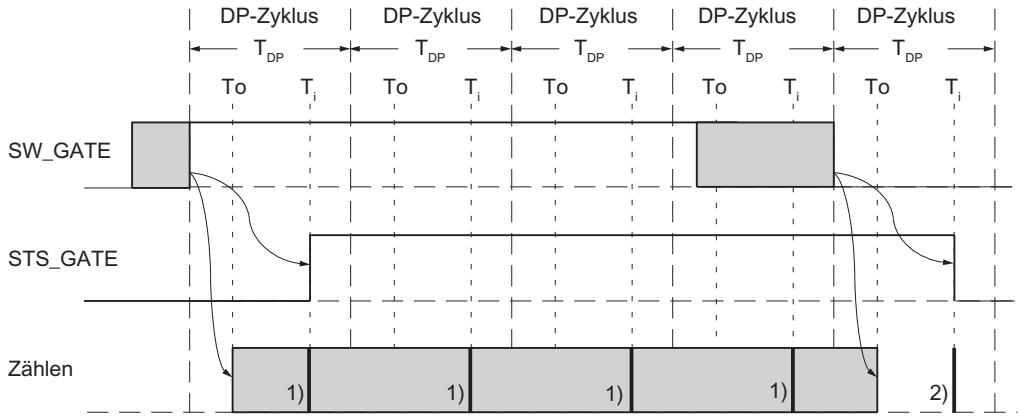

1) Bereitstellen des aktuellen Zählwerts
2) Bereitstellen des Zählwerts, der am Ende des Zählvorgangs gültig war

Torsteuerung mit HW-Tor

Bei der Steuerung mit dem HW-Tor beginnt und endet der Zählvorgang sofort mit dem Öffnen und Schließen des HW-Tores:

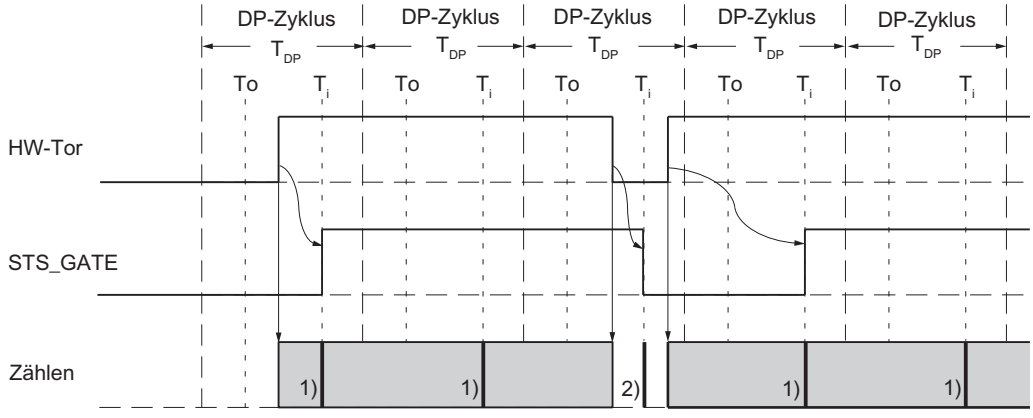

1) Bereitstellen des aktuellen Zählwerts
2) Bereitstellen des Zählwerts, der am Ende des Zählvorgangs gültig war

Gebersignale

Anschließbare Geber

Die Zählsignale, welche die Zählerbaugruppe verarbeiten kann, sind Rechtecksignale, die entweder von Inkrementalgebern oder von Impulsgebern erzeugt werden.

- **Inkrementalgeber** tasten ein Strichgitter ab und erzeugen damit elektrische Rechteckimpulse. Sie unterscheiden sich in der Impulshöhe und in der Anzahl der Signale. Inkrementalgeber liefern zwei um 90° versetzte Spuren.
- **Impulsgeber**, wie zum Beispiel Lichtschranken oder Initiatoren (BEROs), liefern nur ein Rechtecksignal mit einer bestimmten Spannungshöhe.

5 V Differenzsignale

Zählsignale von 5 V-Inkrementalgebern

Der 5 V-Inkrementalgeber liefert folgende Differenzsignale nach RS 422 an die Baugruppe:

- A und /A
- B und /B
- N und /N

Dabei sind die Signale /A, /B und /N die inversen Signale von A, B und N. Die Signale A und B sind um jeweils 90° phasenverschoben.

Bei einem 5 V-Inkrementalgeber werden zum Zählen die Spuren A und B benutzt. Die Spur N dient bei entsprechender Parametrierung zum Setzen des Zählers auf den Ladewert.

Geber, die zu jedem Signal das invertierte Signal liefern, sind symmetrische Geber.

Das folgende Bild zeigt den zeitlichen Verlauf der Gebersignale eines 5 V-Inkrementalgebers.

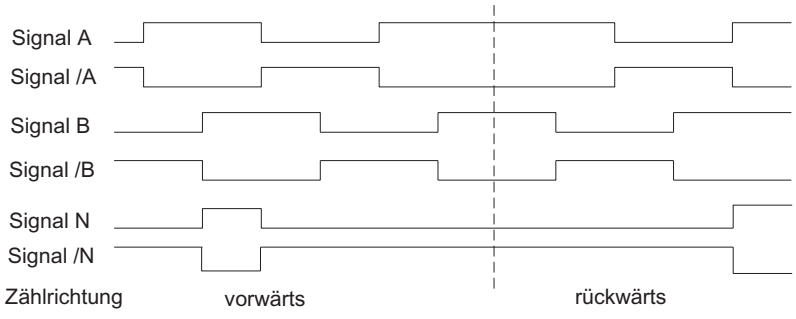

Die Baugruppe erkennt die Zählrichtung an dem Verhältnis der Signale A und B.

Zählrichtung ändern

Sie können mit dem Parameter "Zählrichtung normal" bzw. "Zählrichtung invertiert" die Zählrichtung ändern, ohne die Verdrahtung ändern zu müssen.

Überwachung der Gebersignale

Die Baugruppe überwacht, ob die Leitungen angeschlossen sind und ob ein Drahtbruch oder ein Kurzschluss vorliegt.

Sie können über die Parametrierung festlegen, welche der drei Signalpaare überwacht werden. Unbenutzte Signale müssen daher nicht beschaltet werden, wenn die Diagnose für dieses Signalpaar über die Parametrierung (Überwachung) abgeschaltet wurde.

Wenn alle drei Signale mit Fehler gemeldet werden, dann ist entweder der Geber defekt, die Geberversorgung kurzgeschlossen oder kein Geber angeschlossen.

Wenn die Baugruppe nach ihrer Parametrierung einen Fehler feststellt, dann wird der Fehler in die Diagnosedatensätze DS0 und DS1 eingetragen. Dies kann bei entsprechender Parametrierung zu einem Diagnosealarm führen.

24 V-Signale

Zählsignale von 24 V-Inkrementalgebern

Der 24 V-Inkrementalgeber liefert die 24 V-Signale A*, B* und N* an die Baugruppe. Die Signale A* und B* sind um jeweils 90° phasenverschoben.

Signale mit einer Spannung von 24 V werden mit einem * gekennzeichnet.

Bei einem 24 V-Inkrementalgeber werden zum Zählen die Spuren A* und B* benutzt. Die Spur N* dient bei entsprechender Parametrierung zum Setzen des Zählers auf den Ladewert.

Geber, die keine inversen Signale liefern, sind asymmetrische Geber.

Das folgende Bild zeigt den zeitlichen Verlauf der Gebersignale eines 24 V-Inkrementalgebers.

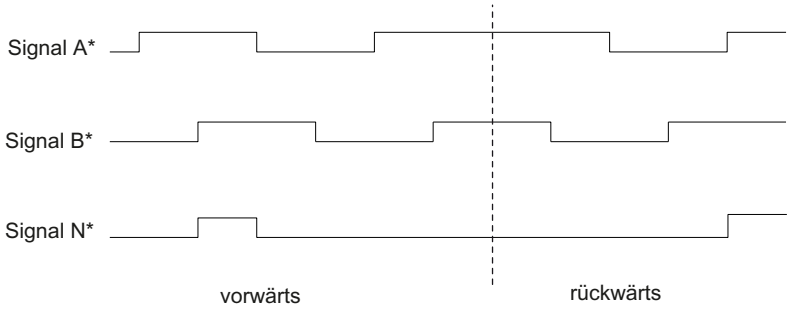

Die Baugruppe erkennt die Zählrichtung an dem Verhältnis der Signale A* und B*.

Bei den Eingängen der 24 V-Gebersignale können Sie parametrieren, ob Sie P-Schalter bzw. Gegentakt oder M-Schalter an den Zähler anschließen. Weitere Informationen hierzu entnehmen Sie der Beschreibung des Gebers.

Zählrichtung ändern

Sie können mit dem Parameter "Zählrichtung normal" bzw. "Zählrichtung invertiert" die Zählrichtung ändern, ohne die Verdrahtung ändern zu müssen.

Zählsignale von 24 V-Impulsgebern ohne/mit Richtungspegel

Der Geber, zum Beispiel ein Initiator (BERO) oder eine Lichtschranke, liefert nur ein Zählsignal, das an den Anschluss A* des Frontsteckers angeschlossen werden muss.

Zusätzlich können Sie ein Signal zur Richtungserkennung an den Anschluss B* des betreffenden Zählers anschließen. Liefert Ihr Geber kein entsprechendes Signal, können Sie auch eine entsprechende Kennung innerhalb des Anwenderprogramms generieren und anschließen oder Sie können ein entsprechendes Prozesssignal verwenden.

Das folgende Bild zeigt den zeitlichen Verlauf der Gebersignale eines 24 V-Impulsgebers mit Richtungspegel und die daraus resultierenden Zählimpulse.

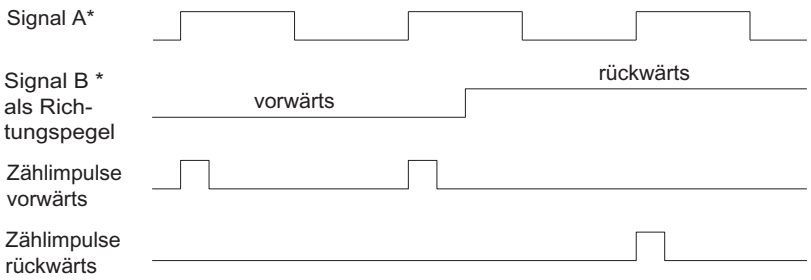

Überwachung der Gebersignale

Bei 24 V-Zählsignalen gibt es keine Überwachung auf Drahtbruch oder Kurzschluss.

Signalauswertung

Übersicht

Der Zähler der Zählerbaugruppe zählt die Flanken der Signale. Im Normalfall wird die Flanke an A (A*) ausgewertet (Einfachauswertung). Um zu einer höheren Auflösung zu gelangen, können Sie bei der Parametrierung zwischen folgenden Alternativen wählen:

- Einfachauswertung (Seite 211)
- Zweifachauswertung (Seite 211)
- Vierfachauswertung (Seite 211)

Eine Mehrfachauswertung ist nur möglich

- bei inkrementellen 5 V-Gebern mit 90° versetzten Signalen A und B
- bei inkrementellen 24 V-Gebern mit 90° versetzten Signalen A* und B*

Voreinstellung

Die Voreinstellung ist "Einfachauswertung".

Einfachauswertung

Einfachauswertung bedeutet, dass nur eine Flanke von A ausgewertet wird.

- Vorwärts-Zählimpulse werden bei steigender Flanke an A und Low-Pegel an B erfasst.
- Rückwärts-Zählimpulse werden bei fallender Flanke von A und Low-Pegel an B erfasst.

Das folgende Bild zeigt die Einfachauswertung der Signale.

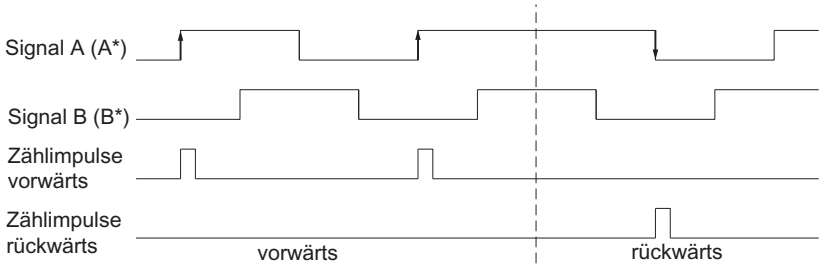

Zweifachauswertung

Zweifachauswertung bedeutet, dass die steigende und fallende Flanke des Signals A ausgewertet werden. Ob Vorwärts- oder Rückwärts-Zählimpulse erzeugt werden, hängt vom Pegel des Signals B ab.

Das folgende Bild zeigt die Zweifachauswertung der Signale.

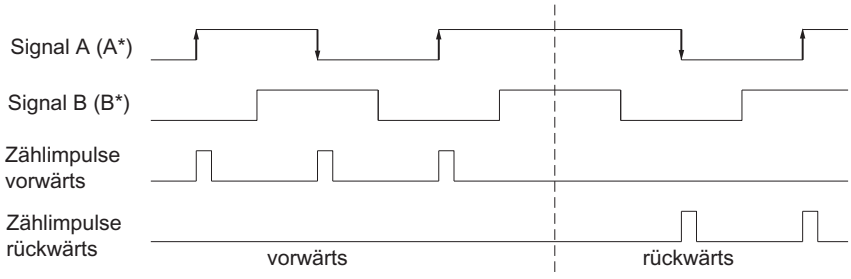

Vierfachauswertung

Vierfachauswertung bedeutet, dass die steigenden und fallenden Flanken von A und B ausgewertet werden. Ob Vorwärts- oder Rückwärts-Zählimpulse erzeugt werden, hängt von den Pegeln der Signale A und B ab.

Das folgende Bild zeigt die Vierfachauswertung der Signale.

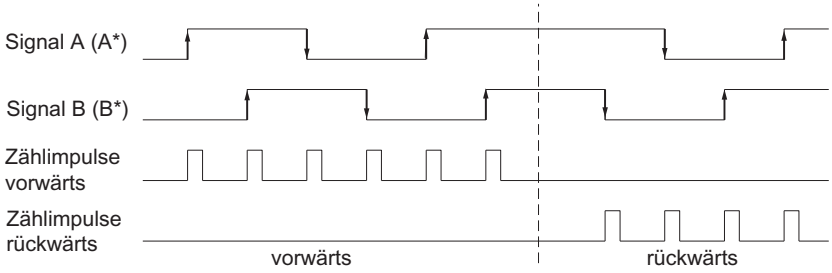

1.2.1.2 FM 350-1 einsetzen

Anschlussbelegung des Frontsteckers der FM 350-1

Frontstecker

Am 20-poligen Frontstecker schließen Sie an:

- Zählsignale
- Digitaleingänge
- Digitalausgänge
- Geberversorgung
- Hilfsspannung und die Lastspannung

Das folgende Bild zeigt die Vorderseite der Baugruppe, den Frontstecker und die Innenseite der Fronttür mit dem Aufdruck der Anschlussbelegung.

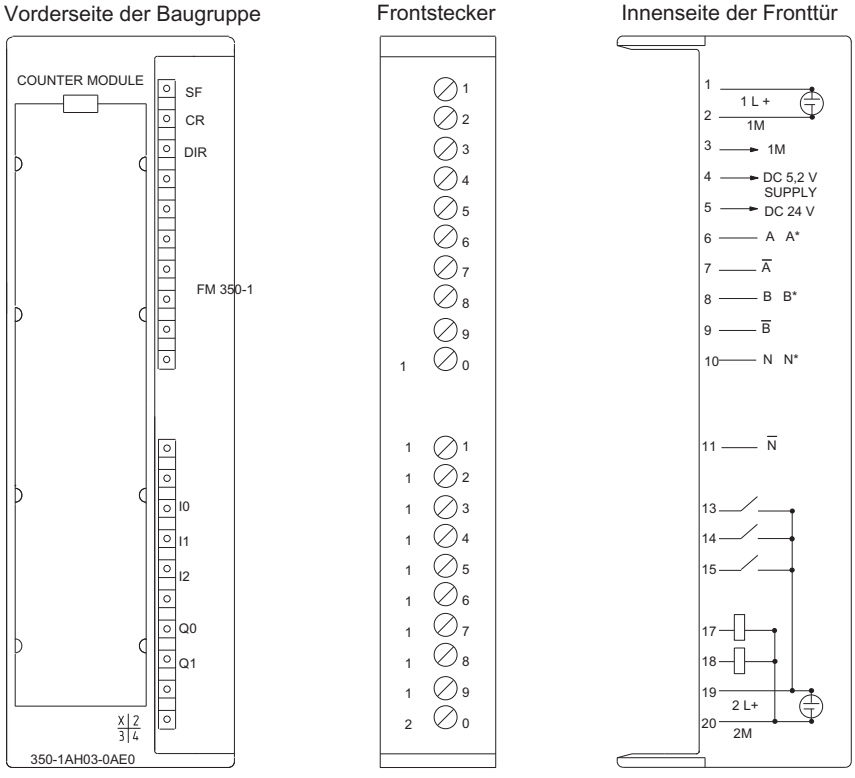

Anschlussbelegung des Frontsteckers

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 350-1.

Anschluss	Name	Eingang / Ausgang	Funktion			
Hilfsspannung						
1	1L+	Eingang	Hilfsspannung 24 V			
2	1M	Eingang	Masse Hilfspannung			
Geberversorgung und Zählsignale			**5 V-Geber RS 422, symmetrisch**	**24 V-Geber asymmetrisch**	**24 V Impulsgeber mit Richtungspegel**	**24 V Initiator**
3	1M	Ausgang	Masse für Geberversorgung			
4	DC5.2V	Ausgang	Geberversorgung 5,2 V			
5	DC24V	Ausgang	Geberversorgung 24 V			
6	A A*	Eingang	Gebersignal A	Gebersignal A*		
7	/A	Eingang	Gebersignal /A	—		
8	B B*	Eingang	Gebersignal B	Gebersignal B*	Richtungssignal	—
9	/B	Eingang	Gebersignal /B	—		
10	N N*	Eingang	Gebersignal N	Gebersignal N*	—	
11	/N	Eingang	Gebersignal /N	—		
12	—	—	—			
Digitaleingänge und Digitalausgänge						
13	I0	Eingang	Digitaleingang DI-Start			
14	I1	Eingang	Digitaleingang DI-Stop			
15	I2	Eingang	Digitaleingang DI-Set			
16	—	—	—			
17	Q0	Ausgang	Digitalausgang DO0			
18	Q1	Ausgang	Digitalausgang DO1			
Lastspannung						
19	2L+	Eingang	Lastspannung 24 V			
20	2M	Eingang	Masse Lastspannung für die Digitaleingänge und Digitalausgänge			

Hinweis

Die Schaltungen für die Zählereingänge (Geberversorgung, Gebersignale) sind potenzialgebunden zur Masse der CPU ausgeführt. Verbinden Sie deshalb die Klemme 2 (1M) mit der Masse der CPU niederohmig. Wenn diese Verbindung fehlt, kann das zu einer Fehlfunktion oder zu einem Defekt der FM 350-1 führen.

Wenn Sie die Geber mit einer externen Spannung versorgen, müssen Sie die Masse dieser externen Spannung ebenfalls mit der Masse der CPU verbinden.

Weitere Informationen zu Aufbau und Verdrahtung finden Sie im Gerätehandbuch (http://support.automation.siemens.com/WW/view/de/1086726) der Zählerbaugruppe FM 350-1.

Hilfsspannung 1L+ / 1M

Für die Spannungsversorgung der 5 V- und 24 V-Geber schließen Sie an den Anschlüssen 1L + und 1M eine Gleichspannung von 24 V an.

Eine integrierte Diode schützt die Baugruppe vor Verpolung der Hilfsspannung.

Die Baugruppe überwacht, ob die Hilfsspannung angeschlossen ist.

Geberversorgung DC5.2V

Die Baugruppe erzeugt aus der Hilfsspannung 1L+ / 1M eine Spannung von 5,2 V bei einem maximalen Strom von 300 mA, die am Anschluss "DC5.2V" für die Spannungsversorgung eines 5 V-Gebers kurzschlussfest zur Verfügung steht. Die Geberversorgung wird auf Kurzschluss kontrolliert.

Geberversorgung DC24V

Für die 24 V-Spannungsversorgung eines Gebers wird die Spannung 1L+ / 1M am Ausgang "DC24V" kurzschlussfest zur Verfügung gestellt. Die Geberversorgung wird auf Kurzschluss kontrolliert.

5 V-Gebersignale A und /A, B und /B, N und /N

Sie können an den Frontstecker Inkrementalgeber mit 5 V-Differenzsignalen nach RS 422 anschließen, d. h. Inkrementalgeber mit den Differenzsignalen A und /A, B und /B, N und /N.

Die Signale A und /A, B und /B, N und /N werden über die entsprechend gekennzeichneten Anschlüsse angeschlossen.

Die Signale N und /N schließen Sie nur dann an, wenn Sie den Zähler auf der Nullmarke des Gebers setzen wollen.

Die Eingänge sind gegenüber dem Bus der S7-300 nicht potenzialgetrennt.

24 V-Gebersignale A*, B* und N*

24 V-Signale werden mit den Buchstaben A*, B* und N* bezeichnet.

Sie können drei verschiedene Gebertypen anschließen:

- Inkrementalgeber mit 24 V-Signalen:
  Die Signale A*, B* und N* werden über die entsprechend gekennzeichneten Anschlüsse angeschlossen.
- Impulsgeber ohne Richtungspegel:
  Das Signal wird am Anschluss A* angeschlossen.
- Impulsgeber mit Richtungspegel:
  Das Zählsignal wird an den Anschluss A* angeschlossen. Der Richtungspegel wird am Anschluss B* angeschlossen.

Die Eingänge sind gegenüber dem Bus der S7-300 nicht potenzialgetrennt.

Eingangsfilter für 24 V-Gebersignale

Um Störungen zu unterdrücken, können Sie für die 24 V-Eingänge A*, B* und N* Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren. Es stehen die folgenden beiden Eingangsfilter zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Maximale Zählfrequenz	200 kHz	20 kHz
Mindestimpulsbreite der Zählsignale	2,5 µs	25 µs

Digitaleingänge DI-Start, DI-Stop und DI-Set

Für die Torsteuerung des Zählers können Sie die Digitaleingänge DI-Start und DI-Stop benutzen. Die Torsteuerung kann sowohl pegel- als auch flankengesteuert erfolgen.

Der Digitaleingang DI-Set dient zum Setzen des Zählers auf den Ladewert.

Die Digitaleingänge werden mit einer Nennspannung von 24 V betrieben.

Die Digitaleingänge sind gegenüber dem Bus der S7-300 und den Zähleingängen potenzialgetrennt.

Eingangsfilter für Digitaleingänge

Um Störungen zu unterdrücken, können Sie für die Digitaleingänge DI-Start, DI-Stop und DI-Set Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren. Es stehen die folgenden beiden Eingangsfilter zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Mindestimpulsbreite der Eingangssignale	2,5 µs	25 µs

Digitalausgänge DO0 und DO1

Zum direkten Auslösen von Steuerungsvorgängen verfügt die FM 350-1 über die beiden Digitalausgänge DO0 und DO1.

Die Digitalausgänge werden über die Lastspannung 2L+ versorgt.

Die Digitalausgänge sind gegenüber dem Bus der S7-300 und den Zähleingängen potenzialgetrennt.

Die Digitalausgänge sind P-Schalter und mit einem Laststrom von 0,5 A belastbar. Sie sind gegen Überlast und Kurzschluss geschützt.

Hinweis

Der direkte Anschluss von Relais und Schützen ist ohne externe Beschaltung möglich.

Das zeitliche Verhalten der Digitalausgänge ist von der Parametrierung abhängig.

Lastspannung 2L+ / 2M

Für die Versorgung der Digitalausgänge DO0 und DO1 muss der Baugruppe über die Klemmen 2L+ und 2M eine Lastspannung von 24 V zugeführt werden.

Eine integrierte Diode schützt die Baugruppe vor Verpolung der Lastspannung.

Die Lastspannung 2L+ / 2M wird von der FM 350-1 nicht überwacht.

FM 350-1 konfigurieren

FM 350-1 parametrieren

Betriebsarten

Betriebsart auswählen

Beschreibung

Mit der Vorgabe einer Betriebsart wählen Sie, mit welcher Funktionalität Sie die Zählerbaugruppe betreiben wollen.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne Tor)	Die FM 350-1 zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit HW- oder SW-Tor)	Die FM 350-1 zählt mit dem Öffnen des Tores ab dem Ladewert bis zur Zählgrenze.
Periodisches Zählen (Seite 187) (mit HW- oder SW-Tor)	Die FM 350-1 zählt mit dem Öffnen des Tores ab dem Ladewert zwischen den Zählgrenzen.
Frequenzmessung (Seite 195) (mit HW- oder SW-Tor)	Die FM 350-1 zählt die Impulse, die in einer dynamischen Messzeit eintreffen.
Drehzahlmessung (Seite 196) (mit HW- oder SW-Tor)	Die FM 350-1 zählt die Impulse, die in einer dynamischen Messzeit von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.
Periodendauermessung (Seite 198) (mit HW- oder SW-Tor)	Die FM 350-1 gibt die dynamische Messzeit als Periodendauer an. Ist die Periodendauer kleiner als die Aktualisierungszeit, dann wird für die Periodendauer ein Mittelwert gebildet.

Die Voreinstellung ist Betriebsart "Endlos Zählen".

Obere Zählgrenze festlegen

Beschreibung

Mit der Parametrierung einer oberen Zählgrenze in Verbindung mit einer Hauptzählrichtung (Seite 222) (vorwärts oder rückwärts) grenzen Sie den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Die obere Zählgrenze kann einen Wert > 1 bis zur oberen Zählbereichsgrenze annehmen.

Messzeit festlegen

Beschreibung

Wenn Sie eine Messbetriebsart ausgewählt haben, dann aktualisiert die Baugruppe die Messwerte zyklisch. Die Aktualisierungszeit ist das Produkt aus Messzeit × 10 ms. Die Messzeit geben Sie als Faktor n vor.

Detaillierte Beschreibungen zu den Wertebereichen für die Messzeit finden Sie unter "Grundsätzliches zum Messen (Seite 190)".

Torsteuerung auswählen

Beschreibung

Mit der Vorgabe einer Torsteuerung wählen Sie, wie der Zählvorgang in der gewählten Betriebsart ablaufen soll.

- Ohne Tor
  Alle Tore sind offen.
- SW-Tor (Seite 204)
  Das SW-Tor wird mit dem Eingangsparameter / Steuerbit SW_GATE geöffnet und geschlossen.
- HW-Tor (Seite 205), pegel- oder flankengesteuert
  Das HW-Tor wird mit entsprechenden Signalen an den Digitaleingängen geöffnet und geschlossen.
- Latchen (Seite 218)
  Mit Flanken am Digitaleingang "DI-Start" werden Zählerstände gespeichert (gelatcht).
- Latch/Retrigger (Seite 219)
  Mit Flanken am Digitaleingang "DI-Start" werden Zählerstände gespeichert (gelatcht). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt.

Die Voreinstellung ist "Ohne Tor".

Latchen

Beschreibung

Wenn Sie bei Torsteuerung "Latchen" auswählen, können Sie mit Flanken am Digitaleingang "DI-Start" Zählerstände speichern (latchen). Dabei wird der Zählerstand nicht verändert.

Voraussetzung

Zum "Latchen" müssen Sie das SW-Tor (Seite 204) benutzen und die Flanke für Latchen auswählen (Seite 222).

Der minimale Abstand der Latchflanken beträgt 1 ms. Ist der zeitliche Abstand zwischen den Flanken kleiner, können Werte verloren gehen.

Funktionsweise

Zählerstand und Latchwert haben ihren Startwert. Der Startwert ist 0 bei einer Betriebsart ohne Hauptzählrichtung und bei Hauptzählrichtung vorwärts. Bei Hauptzählrichtung rückwärts ist der Startwert die parametrierte obere Zählgrenze.

Mit dem Öffnen des SW-Tors wird die Zählfunktion gestartet. Der Zähler beginnt beim Ladewert.

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der Flanke.

Der Zustand des "DI-Start" wird im DB mit dem Statusbit STS_STA angezeigt.

Der Latchwert wird im DB mit LATCH_LOAD angezeigt.

Das folgende Bild zeigt ein Beispiel für "Latchen" mit Ladewert = 0 und positiver Flanke an "DI-Start".

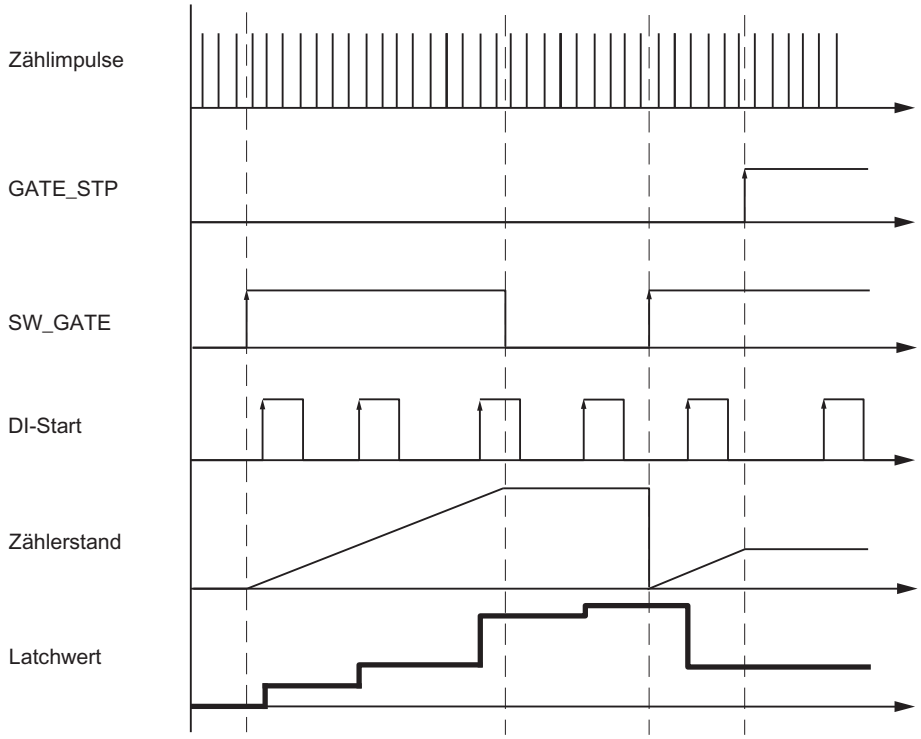

Unterbrechen und Beenden

Wenn Sie das SW-Tor durch Rücksetzen von SW_GATE schließen, wirkt es abbrechend. Mit Flanken an "DI-Start" kann auch bei geschlossenem SW-Tor der unveränderte Zählerstand erneut gespeichert werden.

Wenn Sie das SW-Tor mit GATE_STP der Anweisung CNT_CTL1 schließen, wird der Zählvorgang abgebrochen. Mit "DI-Start" können dann keine Zählerstände mehr gespeichert werden.

Prozessalarm bei "Latchen"

Jedes Speichern von Zählwerten durch "Latchen" kann auch zu einem Prozessalarm führen. Dies kann einen größeren Abstand zwischen den Flanken notwendig machen. Kommen die Alarme schneller, als sie vom System quittiert werden können, gehen Prozessalarme verloren. Dies wird per Diagnosealarm gemeldet.

Latch/Retrigger

Beschreibung

Wenn Sie bei Torsteuerung "Latch/Retrigger" auswählen, können Sie mit Flanken am Digitaleingang "DI-Start" Zählerstände speichern (latchen). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt und zählt vom Ladewert aus weiter (retrigger).

Voraussetzung

Für "Latch/Retrigger" müssen Sie das SW-Tor (Seite 204) benutzen und die Flanke für Latchen auswählen (Seite 222).

Der minimale Abstand der Latchflanken beträgt 1 ms. Ist der zeitliche Abstand zwischen den Flanken kleiner, können Werte verloren gehen.

Funktionsweise

Mit dem Öffnen des SW-Tors wird die Zählfunktion freigegeben.

Zählerstand und Latchwert haben ihren Startwert. Sie bleiben durch das Öffnen des SW-Tors unverändert. Der Startwert ist 0 bei einer Betriebsart ohne Hauptzählrichtung und bei Hauptzählrichtung vorwärts. Bei Hauptzählrichtung rückwärts ist der Startwert die parametrierte obere Zählgrenze.

Erst mit der ersten Flanke am "DI-Start" beginnt der Zählvorgang mit dem Ladewert.

Bei jeder weiteren Flanke am "DI-Start" wird immer wieder der Ladewert geladen.

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der Flanke.

Der Zustand des "DI-Start" wird im DB mit dem Statusbit STS_STA angezeigt.

Der Latchwert wird im DB mit LATCH_LOAD angezeigt.

Das folgende Bild zeigt ein Beispiel für "Latch/Retrigger" mit Ladewert = 0 und positiver Flanke an "DI-Start".

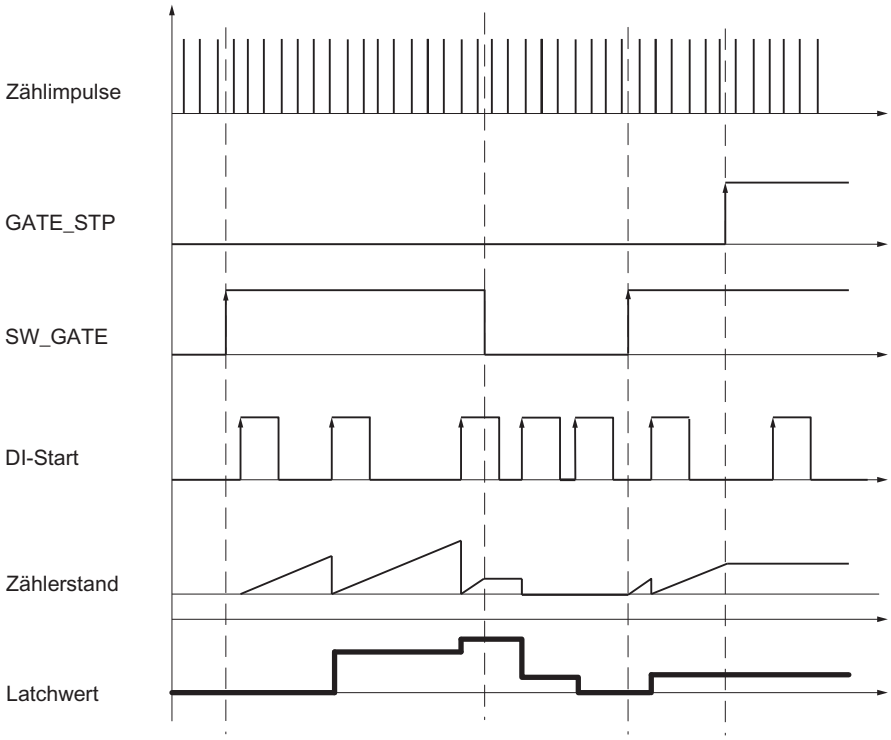

Ein besonderer Anwendungsfall der Torsteuerung "Latch/Retrigger" ist das Messen von Zeiten zwischen zwei Flanken (Seite 232) mit Hilfe der internen Zeitbasis 1 MHz (Seite 225).

Unterbrechen und Beenden

Wenn Sie das SW-Tor durch Rücksetzen von SW_GATE schließen, wirkt es unterbrechend, d. h. mit erneutem Öffnen des SW-Tores wird der Zählvorgang fortgesetzt.
Mit Flanken an "DI-Start" kann auch bei geschlossenem SW-Tor der unveränderte Zählerstand erneut gespeichert werden.

Wenn Sie das SW-Tor mit GATE_STP der Anweisung CNT_CTL1 schließen, wird der Zählvorgang abgebrochen. Mit "DI-Start" können dann keine Zählerstände mehr gespeichert werden.

Prozessalarm bei "Latch/Retrigger"

Jedes Speichern von Zählwerten durch "Latch/Retrigger" kann auch zu einem Prozessalarm führen. Dies kann einen größeren Abstand zwischen den Flanken notwendig machen. Kommen die Alarme schneller, als sie vom System quittiert werden können, gehen Prozessalarme verloren. Dies wird per Diagnosealarm gemeldet.

Zählbereich auswählen

Beschreibung

Auf der Baugruppe befindet sich ein 32 Bit breites Zählregister. Mit der Einstellung Zählbereich wählen Sie, ob die Baugruppe nur im positiven Bereich zählt oder ob das 32. Bit als Vorzeichen-Bit interpretiert wird und somit negative Zahlen dargestellt werden können. Die Wahl des Zählbereichs ist nur möglich, wenn Sie keine Hauptzählrichtung parametrieren.

Wählen Sie den gewünschten Zählbereich aus.

Zusätzliche Informationen finden Sie unter "Zählbereiche und Zählgrenzen (Seite 182)".

Überschreiten der Obergrenze, Unterschreiten der Untergrenze und Nulldurchgang

Bei den Zählgrenzen wird bei Überschreiten bzw. Unterschreiten ein Bit im DB der Anweisung CNT_CTL1 gesetzt.

Im Zählbereich "-31 bis +31 Bit" wird bei einem Nulldurchgang ebenfalls ein Bit im DB gesetzt.

Im Zählbereich "0 bis +32 Bit" wird mit dem Nulldurchgang je nach Zählrichtung zusätzlich noch Überschreiten bzw. Unterschreiten angezeigt.

Ereignis	Statusbit im DB
Überschreiten der Obergrenze	STS_OFLW wird gesetzt.
Unterschreiten der Untergrenze	STS_UFLW wird gesetzt.
Nulldurchgang	STS_ZERO wird gesetzt.

Auslösen von Prozessalarmen

Sie können die Ereignisse Überschreiten der Obergrenze, Unterschreiten der Untergrenze und Nulldurchgang auch per Prozessalarm melden.

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Hauptzählrichtung auswählen

Beschreibung

Mit der Parametrierung einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie durch Parametrieren einer oberen Zählgrenze (Seite 217) den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Eine Hauptzählrichtung können Sie nur in den Betriebsarten "Einmaliges Zählen" und "Periodisches Zählen" auswählen.

Sie können folgendes Verhalten auswählen:

- Keine (ohne Hauptzählrichtung)
- Vorwärts
- Rückwärts

Flanke für Latchen auswählen

Wahl der Flanken

Sie können wählen, mit welcher Flanke am Digitaleingang "DI-Start" Zählerstände gespeichert (gelatcht) werden sollen. Folgendes Verhalten ist parametrierbar:

- Latchen mit positiver Flanke an "DI-Start"
- Latchen mit negativer Flanke an "DI-Start"
- Latchen mit beiden Flanken an "DI-Start"

Auflösung der Periodendauer auswählen

Beschreibung

Der Wert der ermittelten Periodendauer wird in der Einheit 1 µs oder 1/16 µs zur Verfügung gestellt.

Wählen Sie die gewünschte Einheit für die Auflösung aus.

Mögliche Messbereiche mit Fehlerangaben

- Bei einer **Auflösung von 1 µs** beträgt der absolute Fehler 0,1 ‰ des Messwertes.
- Für eine **Auflösung von 1/16 µs** finden Sie Messbereiche und Fehler in der folgenden Tabelle:

Periodendauer T_{min} ± absoluter Fehler
1/16 µs × (160 ± 1)
1/16 µs × (1 600 ± 1)
1/16 µs × (16 000 ± 3)
1/16 µs × (160 000 ± 20)
Ab einer Periodendauer von 100 ms beträgt der absolute Fehler 0,1 ‰ des Messwertes.

Geber

Signalart des Gebers auswählen

Signalart auswählen

1. Wählen Sie die Signalart des eingesetzten Gebers aus.
   Abhängig von Ihrer Auswahl und der Betriebsart sind diejenigen Optionen gesperrt, die für diesen speziellen Geber nicht möglich sind. Eine gesperrte Option wird durch graue Schrift im entsprechenden Auswahlfenster gekennzeichnet.
2. Wählen Sie die Optionen aus, die Sie mit dem ausgewählten Geber nutzen wollen.

Sie können die Signalart von einem der folgenden Geber wählen:

- Geber: 5 V inkrementell (Seite 223) (5 V Differenzsignal)
- Geber: 24 V inkrementell (Seite 224) (24 V Spur A+B phasenversetzt)
- Geber: 24 V Impuls und Richtung (Seite 224)
- Geber: 24 V Initiator (Seite 225)
- Interne Zeitbasis 1 MHz (Seite 225)

Geber: 5 V inkrementell

5 V-Inkrementalgeber

Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B und N, /N einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung (Seite 210)	Einfach	Nur eine Flanke von Signal A wird ausgewertet.
	Zweifach	Beide Flanken von Signal A werden ausgewertet.
	Vierfach	Beide Flanken von Signal A und B werden ausgewertet.
Zählrichtung (Seite 208)	Normal	Zur Anpassung der Zählrichtung an den Prozess
	Invertiert	Zur Anpassung der Zählrichtung an den Prozess
Max. Zählfrequenz	–	Die maximale Zählfrequenz für diesen Geber ist 500 kHz.
Gebereingänge	–	Nicht relevant
Überwachung (Seite 208)	A+B+N	Alle Signalpaare werden auf Kurzschluss und Drahtbruch überwacht.
	A+B	Die Signalpaare A und B werden überwacht.
	A	Nur das Signalpaar A wird überwacht.
	Keine	Kein Signalpaar wird überwacht.

Geber: 24 V inkrementell

24 V-Inkrementalgeber

Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung (Seite 210)	Einfach	Nur eine Flanke von Signal A wird ausgewertet.
	Zweifach	Beide Flanken von Signal A werden ausgewertet.
	Vierfach	Beide Flanken von Signal A und B werden ausgewertet.
Zählrichtung (Seite 209)	Normal	Zur Anpassung der Zählrichtung an den Prozess
	Invertiert	Zur Anpassung der Zählrichtung an den Prozess
Max. Zählfrequenz (Seite 226)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 226)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Geber: 24 V Impuls und Richtung

24 V-Impulsgeber

Wählen Sie die Signalart "24 V Impuls und Richtung", wenn Sie einen 24 V-Impulsgeber mit dem Zählsignal A* und dem Richtungssignal B* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung	–	Nicht relevant
Zählrichtung	–	Nicht relevant
Max. Zählfrequenz (Seite 226)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 226)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Geber: 24 V Initiator

24 V-Initiator

Wählen Sie die Signalart "24 V Initiator", wenn Sie einen 24 V-Impulsgeber mit dem Zählsignal A* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung	–	Nicht relevant
Zählrichtung	–	Nicht relevant
Max. Zählfrequenz (Seite 226)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 226)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Interne Zeitbasis 1 MHz

Interne Zeitbasis 1 MHz

Wählen Sie die Signalart "Interne Zeitbasis 1 MHz", wenn Sie an Stelle eines externen Gebers die interne Zeitbasis von 1 MHz benutzen wollen.

Mit dieser Auswahl können Sie z. B. Zeiten zwischen zwei unmittelbar aufeinanderfolgenden Flanken am Digitaleingang "DI-Start" messen.

Siehe auch

Zeiten zwischen zwei Flanken messen (Seite 232)

Parametrierung der Gebereingänge

Bei der Parametrierung haben Sie bei Signalart (Seite 223) "24 V Impuls und Richtung" gewählt.

Parametrierung der Gebereingänge

Bei der Parametrierung der Gebereingänge wird die Zählrichtung wie folgt festgelegt. Die Tabelle zeigt die Änderung der Zählrichtung in Abhängigkeit von der Parametrierung.

Parametrierung	Anschlussklemme B*	Zählrichtung
M-Schalter	Unbeschaltet	Rückwärts
	Kurzgeschlossen nach Masse	Vorwärts
P-Schalter / Gegentakt	Unbeschaltet	Vorwärts
	24 V angeschlossen	Rückwärts

Eine Richtungsumkehr durch Invertierung des B*-Signals ist bei diesen Zählsignalen nicht möglich.

Hinweis

Bei dieser Art der Auswertung kann der Zählwert bei pendelndem Zählsignal an der Flanke "weglaufen", da alle Signale aufaddiert werden.

Eingangsfilter für 24 V-Zähleingänge

Eingangsfilter für 24 V-Zähleingänge

Um Störungen zu unterdrücken, können Sie für die 24 V-Zähleingänge A*, B* und N* Eingangsfilter mit einer einheitlichen Filterzeit parametrieren. Es gibt folgende Eingangsfilter:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Maximale Zählfrequenz	200 kHz	20 kHz
Mindestimpulsbreite der Zählsignale	2,5 µs	25 µs

Impulse pro Geberumdrehung eintragen

Impulse pro Geberumdrehung

Für die Betriebsart "Drehzahlmessung (Seite 196)" müssen Sie zusätzlich die Impulse pro Geberumdrehung parametrieren.

Den Wert für die Anzahl der Impulse pro Geberumdrehung finden Sie auf dem Typenschild oder in den technischen Daten Ihres Gebers.

Eingänge

Torsignale auswählen

Steuerung des Hardware-Tores

Für die Steuerung des Hardware-Tores können Sie auswählen:

- Pegelgesteuertes HW-Tor (mit Pegel an "DI-Start")
- Flankengesteuertes HW-Tor (mit Flanken an "DI-Start", "DI-Stop")

Weitere Informationen dazu finden Sie unter "Hardware-Tor (Seite 205)".

Mindestimpulsbreite auswählen

Mindestimpulsbreite

Um Störungen zu unterdrücken, können Sie für die Digitaleingänge Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren.

Es stehen zwei Eingangsfilter mit folgenden Daten zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Mindestimpulsbreite der Eingangssignale	2,5 µs	25 µs

Zähler setzen (DI-Set)

Zähler setzen

Übersicht

Wenn Sie den Zählvorgang von einem bestimmten Wert, dem sogenannten Ladewert, starten wollen, müssen Sie parametrieren, mit welchem Signal der Zähler auf den Ladewert gesetzt werden soll. Sie können den Zähler setzen:

- mit dem Eingangsparameter L_DIRECT oder L_PREPAR der Anweisung CNT_CTL1
- mit einem externen Signal entweder über den Eingang "DI-Set" oder über den "DI-Set" in Verbindung mit der Nullmarke des Gebers

Ladewert

Sie können für den Ladewert eine Zahl innerhalb des Zählbereichs vorgeben.

Der Ladewert wird entsprechend dem angewählten Zählbereich interpretiert. Geben Sie zum Beispiel als Ladewert FFFF FFFF H vor, dann wird dieser Wert innerhalb des Zählbereichs "0 bis +32 Bit" als 4 294 967 295 und innerhalb des Zählbereichs "-31 bis +31 Bit" als -1 interpretiert.

Den Ladewert tragen Sie in den DB der Anweisung CNT_CTL1 ein und übertragen ihn mit CNT_CTL1 zur Baugruppe.

Für den Ladewert gelten folgende zulässige Wertebereiche:

Wertebereich für Ladewerte	Hauptzählrichtung		
	keine	vorwärts	rückwärts
Untergrenze	maximale untere Zählgrenze	$-2^{31} + 1$	2
Obergrenze	maximale obere Zählgrenze	parametrierte obere Zählgrenze - 2	$2^{31} - 1$

Zähler setzen mit einem externen Signal

Sie können zwei verschiedene externe Signale wählen, mit denen Sie den Zähler auf den Ladewert setzen:

- Nur "DI-Set" (Seite 229)
- "DI-Set" und Nullmarke (Seite 230) des Gebers

Die Nullmarke des Gebers verwenden Sie dann, wenn Sie an einer bestimmten Stelle Ihres Prozesses den Zähler auf einen bestimmten Zählerstand synchronisieren wollen. Hierdurch erreichen Sie eine größere Präzision des Zählvorgangs.

Das Setzen des Zählers ist unabhängig von der Zählbetriebsart.

Nach erfolgtem Setzen des Zählers mit einem externen Signal wird das Bit STS_SYNC im DB gesetzt.

Hinweis

Die Synchronisierung eines Zählers mit der Nullmarke ist nur bei geöffnetem Tor sinnvoll.

Haben Sie beim Setzen eines Zählers mit einem externen Signal nur eine Zählrichtung freigegeben, dann müssen Sie beachten, dass beim Schließen des Tores die aktuelle Zählrichtung gespeichert (eingefroren) wird. Dadurch ist ein Synchronisieren des Zählers entgegen der freigegebenen Zählrichtung möglich.

Zähler setzen mit "DI-Set"

Zähler setzen mit "DI-Set"

Der Zähler kann über eine steigende Flanke am "DI-Set" mit dem Ladewert geladen werden.

Sie können über die Variablen ENSET_UP und ENSET_DN im DB der Anweisung CNT_CTL1 und durch die Parametrierung das Verhalten der FM 350-1 bei einer steigenden Flanke am "DI-Set" festlegen:

Parameter	Verhalten der FM 350-1
ENSET_UP gesetzt	Der Zähler wird nur beim Vorwärtszählen gesetzt.
ENSET_DN gesetzt	Der Zähler wird nur beim Rückwärtszählen gesetzt.
ENSET_UP und ENSET_DN gesetzt	Der Zähler wird beim Vorwärts- und beim Rückwärtszählen gesetzt.
Parametrierung "Zähler setzen einmalig"	Der Zähler wird nur bei der ersten steigenden Flanke am "DI-Set" gesetzt. Soll der Zähler erneut gesetzt werden, müssen Sie zuerst wieder ENSET_UP oder ENSET_DN setzen. Mit der nächsten steigenden Flanke am "DI-Set" wird dann der Zähler neu gesetzt.
Parametrierung "Zähler setzen mehrmalig"	Der Zähler wird bei jeder steigenden Flanke am "DI-Set" gesetzt, solange ENSET_UP oder/und ENSET_DN gesetzt sind.

Hinweis

Sie müssen auf jeden Fall eine der beiden Variablen ENSET_UP oder/und ENSET_DN setzen, damit der Zähler über den Digitaleingang "DI-Set" gesetzt werden kann.

Einmaliges Setzen mit "DI-Set"

Das folgende Bild zeigt das einmalige Setzen des Zählers mit dem Digitaleingang "DI-Set". Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei der ersten steigenden Flanke am Digitaleingang "DI-Set" wird der Zähler gesetzt, solange auch ENSET_UP gesetzt ist. Wenn Sie den Zähler neu setzen wollen, müssen Sie zuerst ENSET_UP zurücksetzen und anschließend wieder setzen. Dann führt die nächste steigende Flanke am Digitaleingang "DI-Set" zum Setzen des Zählers.

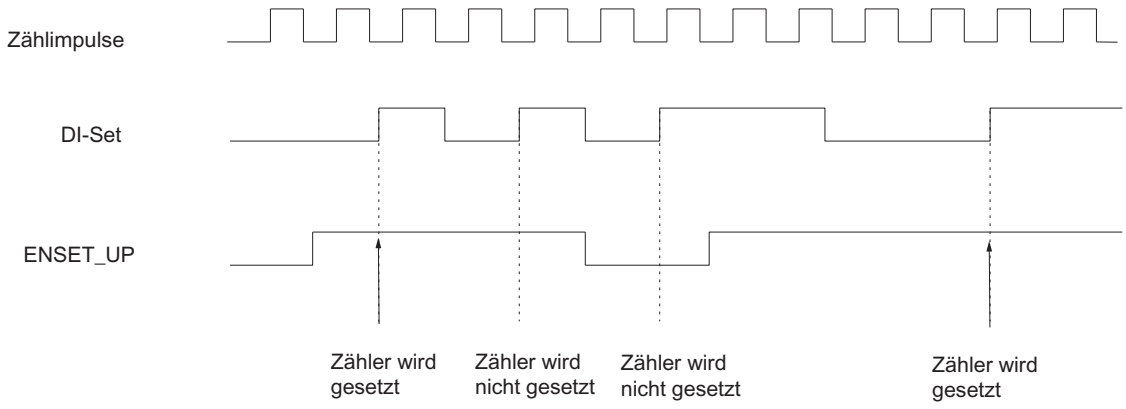

Mehrmaliges Setzen mit "DI-Set"

Das folgende Bild zeigt das mehrmalige Setzen des Zählers mit "DI-Set". Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei jeder steigenden Flanke am "DI-Set" wird der Zähler gesetzt, solange auch ENSET_UP gesetzt ist. Wenn Sie ENSET_UP zurücksetzen, erfolgt kein Setzen des Zählers durch den "DI-Set". Erst wenn Sie ENSET_UP wieder gesetzt haben, führt die nächste steigende Flanke am "DI-Set" zum Setzen des Zählers.

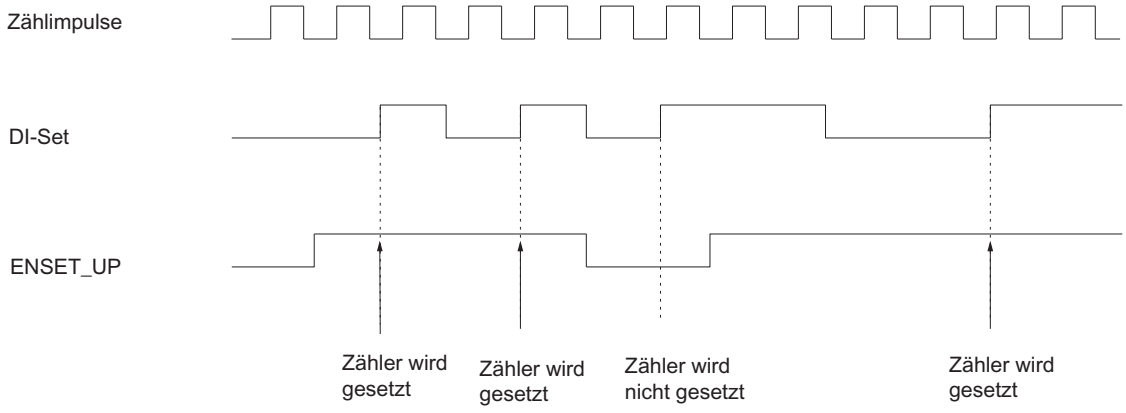

Zähler setzen mit "DI-Set" und Nullmarke

Zähler setzen mit "DI-Set" und Nullmarke

Wenn Sie das Setzen des Zählers mit der Nullmarke des Gebers parametrieren ("Nullmarke für das Setzen auswerten"), dann wird bei der steigenden Flanke der Nullmarke der Zähler auf den Ladewert gesetzt.

Das Setzen wird nur dann durchgeführt, wenn zum Zeitpunkt der steigenden Flanke der Nullmarke auch der Eingang "DI-Set" gesetzt ist.

Sie können über die Variablen ENSET_UP und ENSET_DN im DB der Anweisung CNT_CTL1 und durch die Parametrierung das Verhalten der FM 350-1 bei steigender Flanke der Nullmarke festlegen:

Eingangsparameter	Verhalten der FM 350-1
ENSET_UP gesetzt	Der Zähler wird nur beim Vorwärtszählen gesetzt.
ENSET_DN gesetzt	Der Zähler wird nur beim Rückwärtszählen gesetzt.
ENSET_UP und ENSET_DN gesetzt	Der Zähler wird beim Vorwärts- und beim Rückwärtszählen gesetzt.
Parametrierung "Zähler setzen einmalig"	Der Zähler wird nur bei der ersten steigenden Flanke der Nullmarke gesetzt. Soll der Zähler erneut gesetzt werden, müssen Sie erst wieder ENSET_UP oder ENSET_DN setzen (Flankenauswertung). Mit der nächsten steigenden Flanke der Nullmarke wird dann der Zähler neu gesetzt.
Parametrierung "Zähler setzen mehrmalig"	Der Zähler wird bei jeder steigenden Flanke der Nullmarke gesetzt, solange ENSET_UP oder/und ENSET_DN gesetzt sind.

Hinweis

Sie müssen auf jeden Fall eine der beiden Variablen ENSET_UP oder/und ENSET_DN und den Eingang "DI-Set" setzen, damit der Zähler mit der Nullmarke gesetzt werden kann.

Einmaliges Setzen mit "DI-Set" und Nullmarke

Das folgende Bild zeigt das einmalige Setzen des Zählers mit der Nullmarke. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei der ersten steigenden Flanke der Nullmarke wird der Zähler gesetzt, solange auch ENSET_UP und "DI-Set" gesetzt sind.

Wollen Sie den Zähler erneut setzen, dann müssen Sie ENSET_UP zurücksetzen und dann wieder setzen. Ist "DI-Set" nicht gesetzt, erfolgt das Setzen mit der ersten Nullmarke nach Setzen von "DI-Set". Ist "DI-Set" gesetzt, erfolgt das Setzen mit der nächsten Nullmarke.

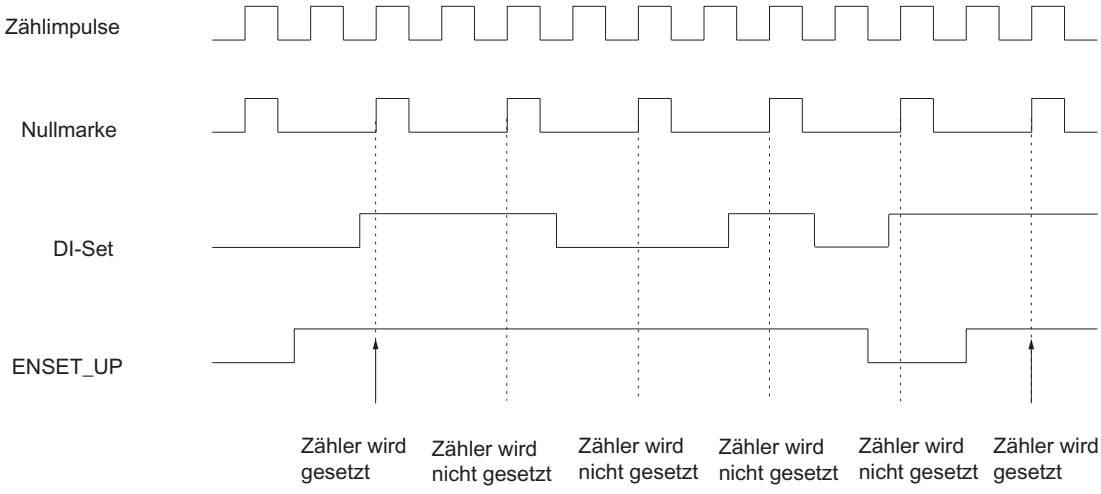

Mehrmaliges Setzen mit "DI-Set" und Nullmarke

Das folgende Bild zeigt das mehrmalige Setzen des Zählers mit der Nullmarke. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei jeder steigenden Flanke der Nullmarke wird der Zähler gesetzt, solange ENSET_UP und "DI-Set" gesetzt sind.

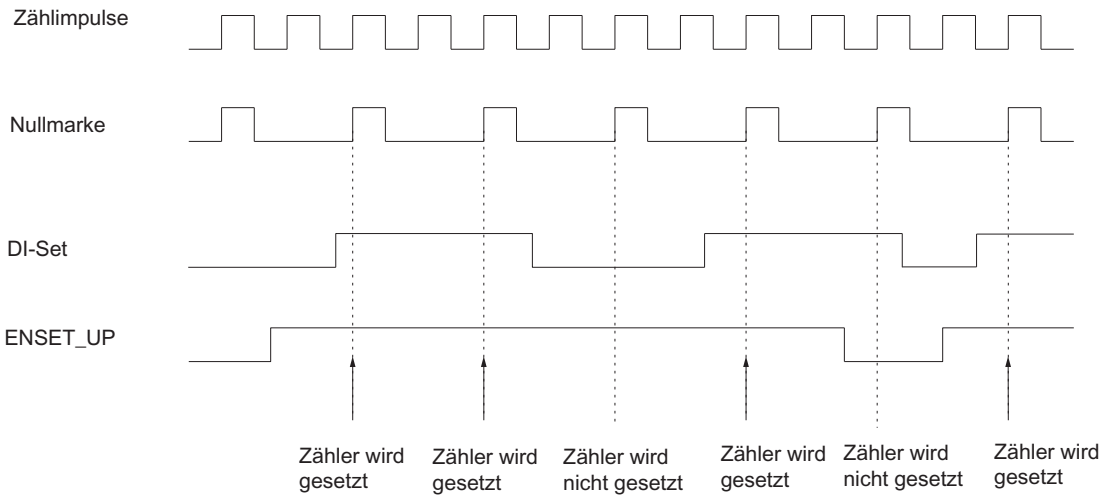

Zeiten zwischen zwei Flanken messen

Einleitung

Sie können die Zeiten zwischen zwei unmittelbar aufeinanderfolgenden Flanken am Digitaleingang "DI-Start" messen.

Voraussetzung

Für die Messung gelten folgende Voraussetzungen:

- Schließen Sie keinen Geber an die FM 350-1 an.
- Parametrieren Sie bei Betriebsarten eine beliebige Zählbetriebsart (Seite 216).
- Parametrieren Sie bei Betriebsarten die Torsteuerung "Latch/Retrigger" (Seite 217).
- Parametrieren Sie bei Geber die Signalart "Interne Zeitbasis 1 MHz" (Seite 223).

Wahl der Flanken

Für die Zeitmessung zwischen zwei unmittelbar aufeinanderfolgenden ...	... parametrieren Sie
steigenden Flanken an "DI-Start"	Latchen mit postiver Flanke
fallenden Flanken an "DI-Start"	Latchen mit negativer Flanke
beliebigen Flanken an "DI-Start"	Latchen mit beiden Flanken

Siehe auch: Flanke für Latchen auswählen (Seite 222)

Funktionsweise

Die FM 350-1 benutzt für die Zeitmessung die interne Zeitbasis 1 MHz. Die Zeitmessung beginnt mit der ersten Flanke an "DI-Start". Mit jeder weiteren Flanke an "DI-Start" wird immer die seit der letzten Flanke vergangene Zeit in µs als Latchwert LATCH_LOAD in der Rückmeldeschnittstelle gespeichert.

Ausgänge

Verhalten der Ausgänge für die Zählbetriebsarten festlegen

Vergleichswerte und Ausgänge

Beschreibung

Sie können auf der FM 350-1 zwei Vergleichswerte (Vergleichswert 1 und 2) ablegen, die den beiden Digitalausgängen zugeordnet sind (Vergleichswert 1: DO0, Vergleichswert 2: DO1). Die Vergleichswerte werden mit dem aktuellen Zählerstand verglichen. Erreicht der Zählerstand einen Vergleichswert, kann der entsprechende Ausgang gesetzt werden.

Bei "Schalten an Vergleichswerten (Seite 239)" sind beide Vergleichswerte dem DO1 zugeordnet.

Vergleichswerte 1 und 2

Die beiden Vergleichswerte tragen Sie im DB der Anweisung CNT_CTL1 ein (CMP_V1, CMP_V2) und übergeben sie an die FM 350-1, indem Sie die Bits T_CMP_V1 bzw. T_CMP_V2 setzen. Der Zählvorgang wird davon nicht beeinflusst.

Die Vergleichswerte müssen innerhalb der Grenzen des jeweiligen Zählbereichs liegen. Der Vergleichswert wird entsprechend dem angewählten Zählbereich interpretiert. Wenn Sie zum Beispiel als Vergleichswert FFFF FFFF H vorgeben, dann wird dieser Wert innerhalb des Zählbereichs "0 bis +32 Bit" als 4 294 967 295 und innerhalb des Zählbereichs "-31 bis +31 Bit" als -1 interpretiert.

Für die Vergleichswerte gelten folgende zulässige Wertebereiche:

Wertebereich für Vergleichswerte	Hauptzählrichtung		
	keine	vorwärts	rückwärts
Untergrenze	maximale untere Zählgrenze	-2^{31}	1
Obergrenze	maximale obere Zählgrenze	parametrierte obere Zählgrenze -1	$2^{31} - 1$

Freigeben der Ausgänge

Bevor die Ausgänge angesteuert werden können, müssen Sie sie zuerst freigeben, indem Sie die entsprechenden Bits im DB setzen. Wenn Sie eines dieser Bits zurücksetzen, wird der zugehörige Ausgang sofort abgeschaltet, auch wenn Sie für ihn eine Impulsdauer parametriert haben.

Für das Freigeben der Ausgänge müssen folgende Bits gesetzt werden:

Ausgang	...wird freigegeben durch Freigabebit
DO0	CTRL_DO0
DO1	CTRL_DO1

Setzen und Rücksetzen der Ausgänge

Wenn Sie für das Verhalten eines Ausgangs "Kein Vergleich" parametrieren, können Sie freigegebene Ausgänge mit den entsprechenden Bits im DB setzen und rücksetzen.

Das Setzen und Rücksetzen der Ausgänge geschieht mit folgenden Bits:

Ausgang	...wird gesetzt mit	...wird rückgesetzt mit
DO0	SET_DO0 = 1	SET_DO0 = 0
DO1	SET_DO1 = 1	SET_DO1 = 0

Verhalten der Ausgänge bei den Zählbetriebsarten

Verhalten der Ausgänge festlegen

Verhalten der Ausgänge DO0 und DO1

Wählen Sie das gewünschte Verhalten des jeweiligen Ausgangs. Aktivieren Sie hierzu das entsprechende Optionsfeld:

- Ersatzwert '1' bei CPU-STOP (Seite 235)
- Kein Vergleich (Seite 236)
- Einschalten bei Zählerstand >= Vergleichswert (Seite 236)
- Einschalten bei Zählerstand <= Vergleichswert (Seite 237)

- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts (Seite 237)
- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts (Seite 238)
- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts (Seite 237)
- Schalten an Vergleichswerten (Seite 239) (nur möglich für Ausgang DO1)

Hinweis

Um die Einstellungen zu aktivieren, müssen Sie im DB mit den Zählerdaten Werte für die Vergleichswerte 1 und 2 eingetragen und mit der Anweisung CNT_CTL1 zur Baugruppe übertragen haben; außerdem müssen Sie die Bits CTRL_DO0 und CTRL_DO1 im DB setzen, um die Ausgänge freizugeben. Siehe auch "Vergleichswerte und Ausgänge (Seite 233)".

Voreinstellung der Ausgänge

In der Voreinstellung sind die Ausgänge abgeschaltet.

Verhalten der Digitalausgänge bei taktsynchronem Betrieb

Auch bei taktsynchronem Betrieb schalten die Ausgänge DO0 und DO1 unmittelbar bei Erfüllung der Vergleichsbedingungen und sind daher unabhängig vom PROFIBUS DP-Zyklus.

Ausnahme:

Wenn Sie für das Verhalten der Digitalausgänge "Kein Vergleich" parametriert haben und die Ausgänge nach Freigabe mit CTRL_DO0 bzw. CTRL_DO1 mit den Steuersignalen SET_DO0 bzw. SET_DO1 ansteuern, dann werden sie zum Zeitpunkt T_o gesetzt und rückgesetzt.

Ersatzwert '1' bei CPU-STOP

Ersatzwert '1' bei CPU-STOP

Wenn Sie in den Grundparametern der Baugruppe bei "Verhalten bei CPU-STOP" die Option "Ersatzwert" gewählt haben und die CPU in STOP geht, gibt die Baugruppe an den Digitalausgängen die hier parametrierten Ersatzwerte aus. Haken Sie z. B. das Optionskästchen für DO0 an, dann wird bei einem STOP der CPU der Ausgang DO0 gesetzt.

Die Ersatzwerte bleiben nach dem STOP-RUN-Übergang der CPU bis zur nächsten Bedienung an den Digitalausgängen gültig.

Wenn Sie den Grundparameter "Verhalten bei CPU-STOP" ändern, werden die Ausgänge rückgesetzt.

⚠ VORSICHT
Es kann zu einem Sachschaden kommen. Parametrierte Ersatzwerte werden bei STOP der CPU immer ausgegeben, auch wenn das Tor geschlossen ist und auch auf Ausgängen, die nicht freigegeben sind. Parametrieren Sie ausschließlich Ersatzwerte, deren Ausgabe nicht zu gefährlichen Anlagenzuständen führen kann.

Kein Vergleich

Kein Vergleich

Der Ausgang bleibt ausgeschaltet und wird durch die Ereignisse Vergleichswert, Nulldurchgang, Überlauf oder Unterlauf nicht beeinflusst. Der Ausgang DOx kann als reiner Digitalausgang verwendet werden. Er ist im freigegebenen Zustand mit dem Bit SET_DOx setzbar und rücksetzbar.

Einschalten bei Zählerstand >= Vergleichswert

Einschalten bei Zählerstand >= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler im Bereich zwischen Vergleichswert und Überlauf befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Überlauf setzt den Ausgang auf 1.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

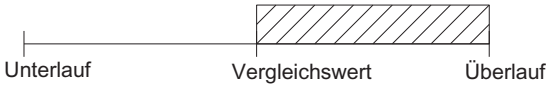

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 240).

Siehe auch

Wirkung der Hysterese (Seite 241)

Einschalten bei Zählerstand <= Vergleichswert

Einschalten bei Zählerstand <= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler zwischen Vergleichswert und Unterlauf befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Unterlauf setzt den Ausgang auf 1.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 240).

Siehe auch

Wirkung der Hysterese (Seite 241)

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts/rückwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler unabhängig von der Zählrichtung den Vergleichswert erreicht.

Voraussetzung:

- Hauptzählrichtung: "Keine"

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

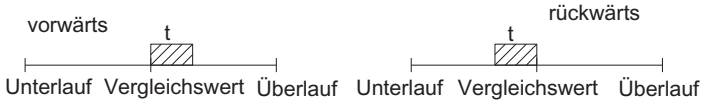

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler beim Vorwärtszählen den Vergleichswert erreicht.

Voraussetzung alternativ:

- Hauptzählrichtung: "Keine"
- Hauptzählrichtung: "Vorwärts"

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

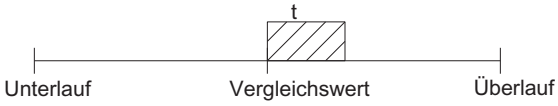

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 240).

Siehe auch

Wirkung der Hysterese (Seite 241)

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer rückwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler beim Rückwärtszählen den Vergleichswert erreicht.

Voraussetzung alternativ:

- Hauptzählrichtung: "Keine"
- Hauptzählrichtung: "Rückwärts"

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 240).

Siehe auch

Wirkung der Hysterese (Seite 241)

Schalten an Vergleichswerten

Schalten an Vergleichswerten

Der Ausgang DO1 schaltet an zwei Vergleichswerten, wenn die folgenden Bedingungen erfüllt sind:

- Sie haben bei DO0 das Verhalten "Kein Vergleich" parametriert.
- Sie haben bei DO1 das Verhalten "Schalten an Vergleichswerten" parametriert.
- Sie haben die beiden Vergleichswerte CMP_V1 und CMP_V2 geladen.
- Sie haben den Ausgang DO1 mit CRTL_DO1 freigeben.

Die nachfolgende Tabelle zeigt Ihnen, wann der DO1 eingeschaltet bzw. ausgeschaltet ist:

Vergleichswerte CMP_V1 und CMP_V2	DO1 ist eingeschaltet bei	DO1 ist ausgeschaltet bei
CMP_V1 < CMP_V2	CMP_V1 ≤ Zählerstand ≤ CMP_V2	Zählerstand < CMP_V1 oder Zählerstand > CMP_V2
CMP_V1 = CMP_V2	CMP_V1 = Zählerstand = CMP_V2	CMP_V1 ≠ Zählerstand ≠ CMP_V2
CMP_V1 > CMP_V2	Zählerstand < CMP_V2 oder Zählerstand > CMP_V1	CMP_V2 ≤ Zählerstand ≤ CMP_V1

Das Vergleichsergebnis wird mit dem Statusbit STS_COMP2 angezeigt.

Quittieren und somit rücksetzen können Sie das Statusbit STS_COMP2 erst, wenn die Vergleichsbedingung nicht mehr erfüllt ist.

Der Zustand des Ausgangs DO1 wird mit dem Statusbit STS_CMP2 angezeigt.

Bei diesem Ausgangsverhalten gibt es keine Hysterese.

Ein Steuern des Ausgangs DO1 durch das Steuerbit SET_DO1 ist bei diesem Ausgangsverhalten nicht möglich.

Die folgenden Bilder zeigen den Zustand des Ausgangs DO1 in Abhängigkeit vom Zählerstand und den Vergleichswerten:

- Beim Start des Zählvorgangs ist CMP_V2 > CMP_V1

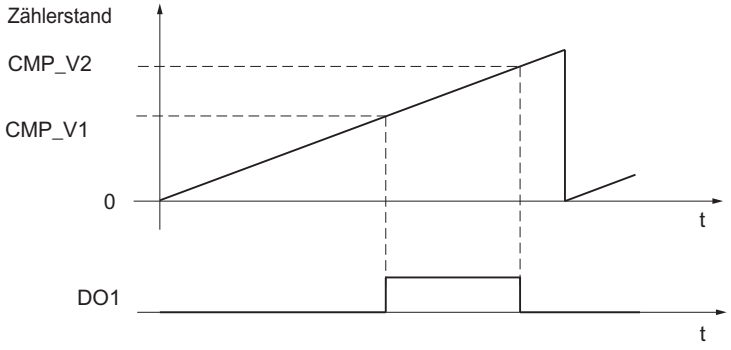

- Beim Start des Zählvorgangs ist CMP_V2 < CMP_V1

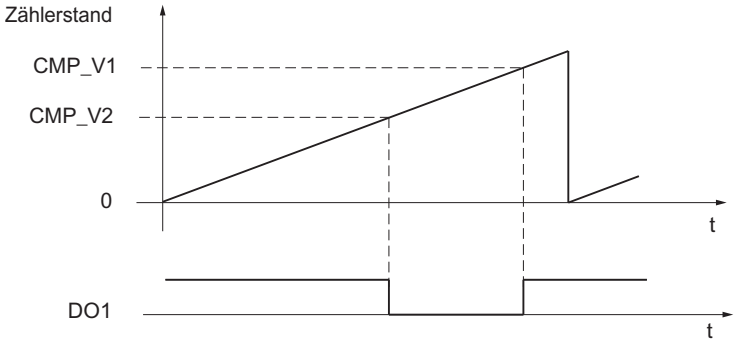

Randbedingungen für das Verhalten der Ausgänge

Randbedingungen für das Verhalten der Digitalausgänge

Wenn Sie das Verhalten der Digitalausgänge parametrieren, müssen Sie folgende Randbedingungen einhalten:

Wenn..	dann...
... Sie einen Ausgang "Einschalten bei Zählerstand >= Vergleichswert oder <= Vergleichswert" parametrieren wollen,	... müssen Sie sicherstellen, dass die Zeit zwischen diesen Ereignissen größer als die minimale Schaltzeit der Ausgänge ist (Schaltzeit: 300 µs); sonst gehen Steuerimpulse an den Ausgängen verloren. Erreicht der Zählerstand den Vergleichswert erneut, während der Ausgang noch eingeschaltet ist, wird kein neuer Impuls ausgelöst. Ein weiterer Impuls kann erst dann ausgelöst werden, wenn der Ausgang nicht mehr eingeschaltet ist.
... Sie einen Ausgang "Einschalten bei Zählerstand >= Vergleichswert" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswertes vorwärts oder rückwärts" freigeben.

Wenn..	dann...
... Sie einen Ausgang " Einschalten bei Zählerstand <= Vergleichswert" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswertes vorwärts oder rückwärts" freigeben.
... Sie einen Ausgang "Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswertes rückwärts" freigeben.
... Sie einen Ausgang "Einschalten bei Erreichen des Vergleichswerts für Impulsdauer rückwärts" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswerts vorwärts" freigeben.

Wirkung der Hysterese

Wirkungsweise der Hysterese bei Parametrierung "Einschalten bei Zählerstand >= Vergleichswert oder <= Vergleichswert"

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Die Parametrierung für dieses Beispiel ist:

- Hauptzählrichtung: "Vorwärts"
- Verhalten des Ausgangs: "Einschalten bei Zählerstand >= Vergleichswert"

Mit dem Erreichen der Vergleichsbedingung (Zählerstand = 5) wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, im Beispiel bei Zählwert 2 bzw. 8, ist die Hysterese nicht mehr aktiv. Der Vergleicher schaltet wieder entsprechend seinen Vergleichsbedingungen, im Beispiel bei Zählwert 5.

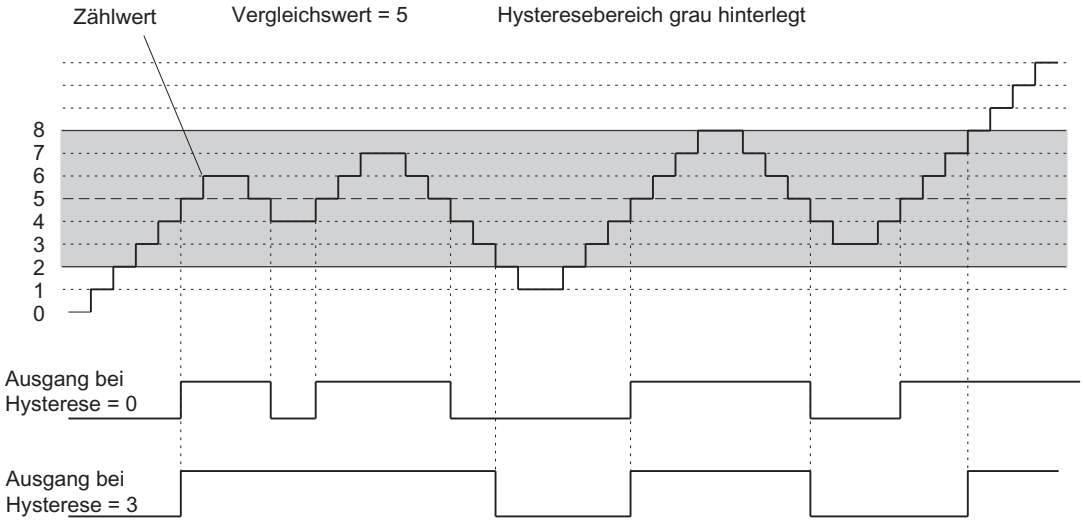

Hinweis

Ist der Zählerstand gleich dem Vergleichswert und die Hysterese aktiv, setzt die FM 350-1 den Ausgang zurück, wenn ein Wechsel der Zählrichtung auf dem Vergleichswert erfolgt (siehe folgendes Bild).

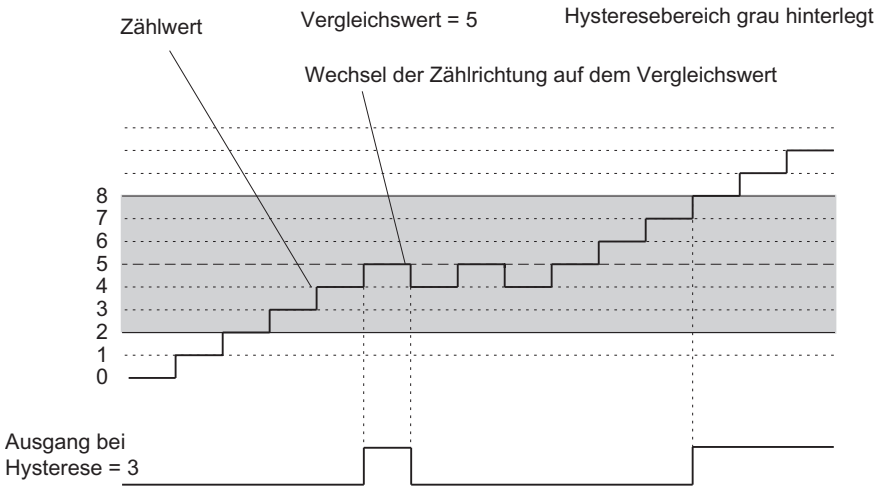

Wirkungsweise der Hysterese bei Parametrierung "Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts oder rückwärts"

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Die Parametrierung für dieses Beispiel ist:

- Hauptzählrichtung: "Keine"
- Verhalten des Ausgangs: "Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts"
- Impulsdauer > 0

Mit dem Erreichen der Vergleichsbedingung (Zählerstand = 5) wird die Hysterese aktiv und am Ausgang ein Impuls der parametrierten Dauer ausgegeben.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv.

Wird die Hysterese aktiv, speichert sich die FM 350-1 die Zählrichtung. Wird der Hysteresebereich entgegen der gespeicherten Richtung verlassen, wird ein Impuls ausgegeben.

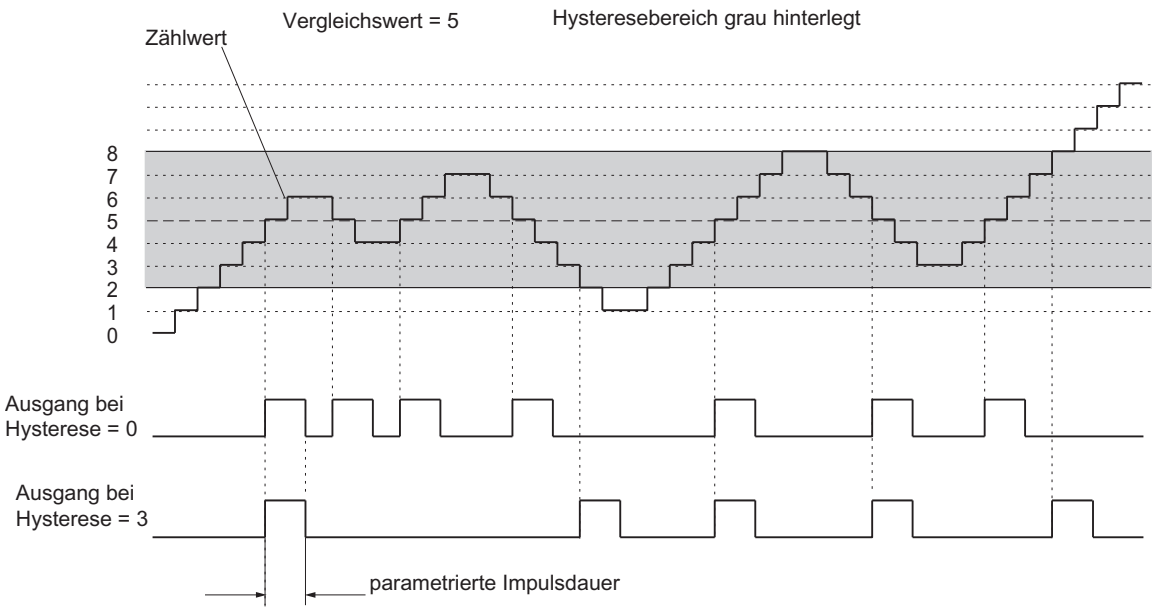

Verhalten der Ausgänge für die Messbetriebsarten festlegen

Grenzwerte und Ausgänge

Unterer und oberer Grenzwert

Sie können für Frequenzmessung, Drehzahlmessung oder Periodendauermessung jeweils einen oberen und einen unteren Grenzwert hinterlegen, bei deren Verletzung der Digitalausgang DO0 aktiviert wird. Diese Grenzwerte können Sie parametrieren und über die Steuerschnittstelle ändern.

Für die unteren und oberen Grenzwerte sind folgende Wertebereiche zulässig:

Messbetriebsart	Unterer Grenzwert (UG)	Oberer Grenzwert
Frequenzmessung (5 V-Geber)	0 bis 499 999 999 × 10^{-3} Hz	UG+1 bis 500 000 000 × 10^{-3} Hz
Frequenzmessung (24 V-Geber)	0 bis 199 999 999 × 10^{-3} Hz	UG+1 bis 200 000 000 × 10^{-3} Hz
Drehzahlmessung	0 bis 24 999 999 × 10^{-3}/min	UG+1 bis 25 000 000 × 10^{-3}/min
Periodendauermessung (Auflösung 1 µs)	0 bis 119 999 999 µs	UG+1 bis 120 000 000 µs
Periodendauermessung (Auflösung 1/16 µs)	0 bis 1 919 999 999 µs	UG+1 bis 1 920 000 000 µs

Freigeben der Ausgänge

Bevor die Ausgänge angesteuert werden können, müssen Sie sie zuerst freigeben, indem Sie die entsprechenden Bits im DB setzen. Wenn Sie eines dieser Bits zurücksetzen, wird der zugehörige Ausgang sofort abgeschaltet.

Ausgang	...wird freigegeben durch Freigabebit
DO0	CTRL_DO0
DO1	CTRL_DO1

Verhalten der Ausgänge bei den Messbetriebsarten

Verhalten der Digitalausgänge

Digitalausgang DO0

Für den Digitalausgang DO0 können Sie eine von 4 möglichen Reaktionen auf das Erreichen der Grenzwerte einstellen. Die verschiedenen Möglichkeiten sehen Sie in der folgenden Tabelle.

Parametrierung des Digitalausgangs DO0	Verhalten des Digitalausgangs DO0	Schaltzeitpunkt	
		Taktsynchroner Betrieb	Nicht taktsynchroner Betrieb
Kein Vergleich	Keine Beeinflussung durch die Grenzwertüberwachung. Ein bereits gesetzter Ausgang DO0 wird durch Umparametrieren auf "Kein Vergleich" zurückgesetzt. Sie können den Ausgang DO0 als Digitalausgang frei verwenden und mit dem Steuersignal SET_DO0 setzen und rücksetzen, wenn Sie ihn mit dem Steuersignal CTRL_DO0 freigegeben haben.	Zum Zeitpunkt T_o	Unmittelbar nach Setzen bzw. Rücksetzen des Ausgangs
Außerhalb der Grenzwerte	DO0 wird in diesen beiden Fällen gesetzt: • Messwert < Unterer Grenzwert • Messwert > Oberer Grenzwert	Am Ende der Aktualisierungszeit zum Zeitpunkt T_i	Am Ende der Aktualisierungszeit
Unterschreiten des unteren Grenzwertes	DO0 wird gesetzt bei • Messwert < Unterer Grenzwert		
Überschreiten des oberen Grenzwertes	DO0 wird gesetzt bei • Messwert > Oberer Grenzwert		

Wählen Sie das gewünschte Verhalten des Ausgangs DO0. Aktivieren Sie hierzu das entsprechende Optionsfeld.

Digitalausgang DO1

Den Ausgang DO1 können Sie im freigegebenen Zustand als Digitalausgang frei verwenden und mit dem Steuersignal SET_DO1 setzen und rücksetzen.

DO1 wird durch keine Grenzwertüberwachung beeinflusst.

Im taktsynchronen Betrieb schaltet DO1 am Ende der Aktualisierungszeit zum Zeitpunkt T_o.

Impulsdauer festlegen

Impulsdauer

Zur Anpassung an die in Ihrem Prozess verwendeten Aktoren (Schütze, Stellglieder etc.) können Sie eine Impulsdauer vorgeben. Die Impulsdauer gibt an, wie lange die Ausgänge DO0 und DO1 bei Erreichen eines Vergleichswerts eingeschaltet sind.

Die Impulsdauer ist bei der Parametrierung "Hauptzählrichtung vorwärts" oder "Hauptzählrichtung rückwärts" nur in Hauptzählrichtung wirksam.

Bei Parametrierung ohne Hauptzählrichtung kann die Impulsdauer in beiden Zählrichtungen wirksam sein.

Die Impulsdauer beginnt mit dem Setzen des Ausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 1 ms.

Sie können für die Impulsdauer einen Wert zwischen 0 und 500 ms vorgeben. Dieser Wert gilt für beide Ausgänge gemeinsam.

Bei Impulsdauer = 0 ms wird der Ausgang bei Erreichen des Vergleichswerts gesetzt und mit dem nächsten Zählimpuls rückgesetzt.

Die Voreinstellung für die Impulsdauer ist 0.

Hinweis

Wenn Sie als Impulsdauer = 0 ms vorgeben, wird der Ausgang so lange eingeschaltet, wie der Zählerstand gleich dem Vergleichswert ist (siehe obere Hälfte im folgenden Bild).

Wegen der Schaltzeiten der Digitalausgänge (maximal 300 µs) können Steuerimpulse an den Ausgängen verloren gehen, wenn der zeitliche Abstand zwischen den Zählimpulsen kleiner ist als die Schaltzeiten der Digitalausgänge (siehe untere Hälfte im folgenden Bild).

Stellen Sie deshalb sicher, dass der zeitliche Abstand zwischen den Zählimpulsen größer ist als die Schaltzeiten der Digitalausgänge.

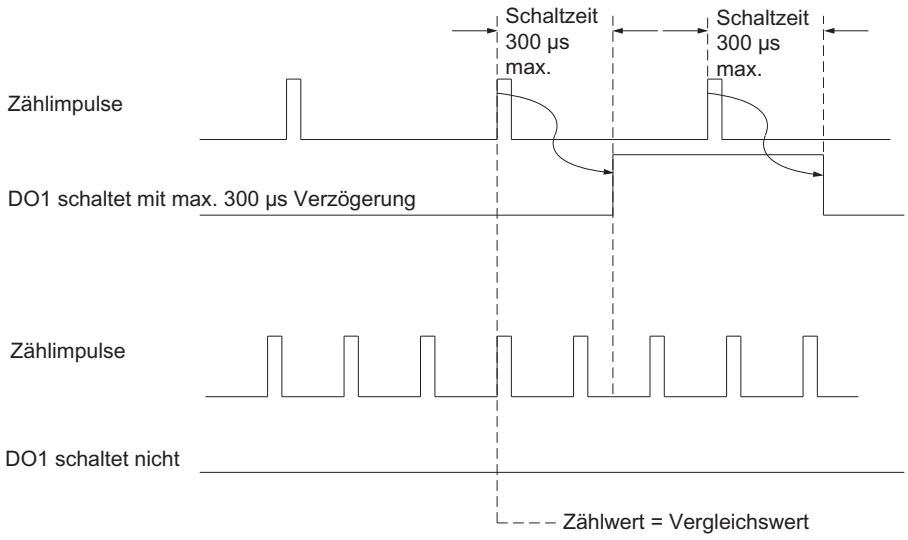

Hysterese festlegen

Prinzipielle Wirkungsweise der Hysterese

Ein Geber kann an einer bestimmten Position stehenbleiben und dann um diese Position "pendeln". Dieser Zustand führt dazu, dass der Zählerstand um einen bestimmten Wert herum schwankt. Liegt nun in diesem Schwankungsbereich zum Beispiel ein Vergleichswert, würde der zugehörige Ausgang im Rhythmus dieser Schwankungen ein- und ausgeschaltet werden. Um dieses Schalten bei kleinen Schwankungen zu verhindern, ist die FM 350-1 mit einer parametrierbaren Hysterese ausgestattet.

Sie können für die Hysterese Werte zwischen 0 und 255 parametrieren.

Abhängig vom parametrierten Wert wirkt die Hysterese wie folgt:

Hysterese	Wirkung
Hysteresewert n = 0, 1	Die Hysterese ist unwirksam (abgeschaltet). Der Ausgang reagiert auf die kleinste Änderung des Zählerstands.
2 ≤ Hysteresewert n ≤ 255	Die Hysterese ist wirksam. Der Ausgang reagiert erst, wenn der Zählerstand sich um n Einheiten vom Vergleichswert entfernt hat.

Die Hysterese wirkt auch bei Überlauf und Unterlauf.

Freigabe Prozessalarme

Auslösen eines Prozessalarms freigeben

Einleitung

Sie können bei der FM 350-1 einstellen, welche Ereignisse im Betrieb einen Prozessalarm auslösen sollen.

Was ist ein Prozessalarm?

Soll auf ein Ereignis unabhängig von dem Zyklus der CPU reagiert werden, kann die FM 350-1 einen Prozessalarm auslösen. Die CPU unterbricht daraufhin das zyklische Programm und bearbeitet den Prozessalarm-OB.

Freigeben des Prozessalarms

Das Freigeben von Prozessalarmen erfolgt in zwei Schritten:

1. Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.
2. Unter "Freigabe Prozessalarme" müssen Sie dann die entsprechenden Ereignisse auswählen:
   - Ereignisse für das Auslösen eines Prozessalarms bei den Zählbetriebsarten (Seite 248)
   - Ereignisse für das Auslösen eines Prozessalarms bei den Messbetriebsarten (Seite 248)

Prozessalarm-OB, OB 40

Tritt ein Prozessalarm auf, wird das Anwenderprogramm unterbrochen, die Daten von der Baugruppe in die Startinformation des OB 40 (Seite 248) übertragen und der OB 40 aufgerufen. Mit dem Verlassen des OB 40 wird der Prozessalarm quittiert.

Ist kein OB 40 programmiert, geht die CPU in STOP. Wird daraufhin wieder in RUN geschaltet, sind die Prozessalarmanforderungen gelöscht.

Verlorener Prozessalarm

Tritt ein Ereignis ein, das einen Prozessalarm auslösen soll, und das vorhergehende gleiche Ereignis wurde noch nicht quittiert, wird kein weiterer Prozessalarm ausgelöst; der Prozessalarm geht verloren. Je nach Parametrierung kann das zu dem Diagnosealarm "Prozessalarm verloren" führen.

Voreinstellung

In der Voreinstellung sind keine Prozessalarme freigegeben.

Ereignisse für das Auslösen eines Prozessalarms bei den Zählbetriebsarten

Welche Ereignisse können einen Prozessalarm bei den Zählbetriebsarten auslösen?

Bei den Zählbetriebsarten können folgende Ereignisse beim Betrieb der FM 350-1 einen Prozessalarm auslösen:

- Öffnen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Schließen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Überschreiten der Obergrenze
- Unterschreiten der Untergrenze
- Nulldurchgang
- Erreichen des Vergleichswertes 1 oder 2 in Vorwärtszählrichtung
- Erreichen des Vergleichswertes 1 oder 2 in Rückwärtszählrichtung
- Zähler setzen (DI-Set)
- Latchen

Aktivieren Sie die Optionskästchen für die betreffenden Ereignisse.

Sie können beliebig viele Ereignisse für die Prozessalarmauslösung auswählen. Für die Prozessalarme bei Erreichen eines Vergleichswertes müssen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 240) beachten.

Ereignisse für das Auslösen eines Prozessalarms bei den Messbetriebsarten

Welche Ereignisse können einen Prozessalarm bei den Messbetriebsarten auslösen?

Bei den Messbetriebsarten können folgende Ereignisse beim Betrieb der FM 350-1 einen Prozessalarm auslösen:

- Öffnen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Schließen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Unterschreiten des unteren Grenzwertes
- Überschreiten des oberen Grenzwertes
- Messende

Aktivieren Sie die Optionskästchen für die betreffenden Ereignisse.

Sie können beliebig viele Ereignisse für die Prozessalarmauslösung auswählen.

Startinformation des OB 40

Startinformation des OB 40

In der Startinformation des OB 40 wird die temporäre Variable OB40_POINT_ADDR beschrieben.

Die Variable OB40_POINT_ADDR (Byte 8 bis 11) umfasst 4 Byte. In Byte 8 und 9 werden die Informationen über das Ereignis eingetragen, das den Prozessalarm ausgelöst hat.

Die Tabelle zeigt, welche Bits bei welchem Alarm gesetzt sind. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung: Alarm bei ...
8	0	Öffnen des Tores
	1	Schließen des Tores
	2	Überschreiten der Obergrenze (bei Zählbetriebsart)
		Messwert außerhalb der Grenzwerte (bei Messbetriebsart)
	3	Unterschreiten der Untergrenze (bei Zählbetriebsart)
		Messende (bei Messbetriebsart)
	4	Erreichen des Vergleichswertes 1 in Vorwärtszählrichtung
	5	Erreichen des Vergleichswertes 1 in Rückwärtszählrichtung
	6	Erreichen des Vergleichswertes 2 in Vorwärtszählrichtung
	7	Erreichen des Vergleichswertes 2 in Rückwärtszählrichtung
9	0	Nulldurchgang
	5	Zähler setzen (DI-Set) (Synchronisation)
	7	Latchen

Steuer- und Rückmeldeschnittstelle der FM 350-1

Übersicht

Wenn Sie die FM 350-1 ohne Anweisungen betreiben wollen, dann steuern und beobachten Sie die FM 350-1 direkt über die Steuer- und Rückmeldeschnittstelle (Nutzdatenschnittstelle).

Die Nutzdatenschnittstelle beginnt bei der Anfangsadresse der Baugruppe und ist 16 Byte lang.

Mit Ladebefehlen können Sie die Rückmeldeschnittstelle lesen.

Mit Transferbefehlen können Sie in die Steuermeldeschnittstelle schreiben.

Ein Mischbetrieb mit Lade-/Transferbefehlen und Programmierung mit Anweisungen ist nicht erlaubt.

Steuerschnittstelle für die Zählbetriebsarten

Steuerschnittstelle für die Zählbetriebsarten

Der Parameter LOAD_VAL (Byte 0 bis 3) hat zweierlei Bedeutung:

- Wenn Sie das Bit L_DIRECT oder L_PREPAR setzen, dann wird LOAD_VAL als Ladewert interpretiert.
- Wenn Sie das Bit C_DOPARA setzen, können Sie im Byte 0 das Verhalten der Ausgänge DO0 und DO1 festlegen, Byte 1 und 2 werden als Hysterese und Impulsdauer interpretiert.

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) der FM 350-1 für die Zählbetriebsarten.

Offset zur Anfangs-adresse	Parameter	Bedeutung				
Byte 0 bis 3	LOAD_VAL	Ladewert; direkt und vorbereitend laden mit Bit L_DIRECT Ladewert; vorbereitend laden mit Bit L_PREPAR				
Byte 0		Verhalten der Ausgänge DO0 und DO1, der Hysterese und der Impulsdauer; festlegen mit Bit C_DOPARA				
		Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Verhalten von Ausgang DO0**
		x	0	0	0	Kein Vergleich
		x	0	0	1	Einschalten bei Zählerstand >= Vergleichswert
		x	0	1	0	Einschalten bei Zählerstand <= Vergleichswert
		x	0	1	1	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts
		x	1	0	0	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts
		x	1	0	1	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts
		Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Verhalten von Ausgang DO1**
		x	0	0	0	Kein Vergleich
		x	0	0	1	Einschalten bei Zählerstand >= Vergleichswert
		x	0	1	0	Einschalten bei Zählerstand <= Vergleichswert
		x	0	1	1	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts
		x	1	0	0	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts
		x	1	0	1	Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts
		x	1	1	0	Schalten an Vergleichswerten
		x = irrelevant				
Byte 1		Hysterese (Wertebereich 0 … 255)				
Byte 2		Impulsdauer (Wertebereich 0 … 250)				
Byte 3		Reserve = 0				
Byte 4 bis 7	CMP_V1	Vergleichswert 1; laden mit Bit T_CMP_V1				
Byte 8 bis 11	CMP_V2	Vergleichswert 2; laden mit Bit T_CMP_V2				
Byte 12	– NEUSTQ – – OT_ERR_A – – –	Bit 7: Reserve = 0 Bit 6: Neustartquittung Bit 5: Reserve = 0 Bit 4: Reserve = 0 Bit 3: Bedienfehlerquittung Bit 2: Reserve = 0 Bit 1: Reserve = 0 Bit 0: Reserve = 0				

Offset zur Anfangs-adresse	Parameter	Bedeutung
Byte 13	–	Bit 7: Reserve = 0
	–	Bit 6: Reserve = 0
	–	Bit 5: Reserve = 0
	–	Bit 4: Reserve = 0
	SW_GATE	Bit 3: Steuerbit SW-Tor
	GATE_STP	Bit 2: Genereller Tor-Stopp
	ENSET_DN	Bit 1: Freigabe Synchronisation rückwärts
	ENSET_UP	Bit 0: Freigabe Synchronisation vorwärts
Byte 14	–	Bit 7: Reserve = 0
	–	Bit 6: Reserve = 0
	–	Bit 5: Reserve = 0
	–	Bit 4: Reserve = 0
	SET_DO1	Bit 3: Steuerbit DO1
	SET_DO0	Bit 2: Steuerbit DO0
	CTRL_DO1	Bit 1: Freigabe DO1
	CTRL_DO0	Bit 0: Freigabe DO0
Byte 15	–	Bit 7: Reserve = 0
	C_DOPARA [1]	Bit 6: Funktion DO0/DO1, Hysterese oder Impulsdauer ändern
	RES_ZERO	Bit 5: Statusbits für Nulldurchgang, Überschreiten, Unterschreiten und Vergleicher rücksetzen
	RES_SYNC	Bit 4: Statusbit für Synchronisation rücksetzen
	T_CMP_V2 [2]	Bit 3: Vergleichswert 2 laden
	T_CMP_V1 [2]	Bit 2: Vergleichswert 1 laden
	L_PREPAR [2]	Bit 1: Zähler vorbereitend laden
	L_DIRECT [2]	Bit 0: Zähler direkt und vorbereitend laden

[1] Dieses Bit dürfen Sie nicht gleichzeitig mit Bit 0, 1, 2, oder 3 von Byte 15 setzen.
[2] Dieses Bit dürfen Sie nicht gleichzeitig mit Bit 6 von Byte 15 setzen.

Erläuterungen zu den Steuerbits für die Zählbetriebsarten

Steuerbit	Erläuterungen
C_DOPARA	Mit diesem Bit ändern Sie die Funktion und das Verhalten von DO0 und DO1, die Hysterese und die Impulsdauer. Die Werte aus Byte 0 bis 2 werden als neue Funktion, Hysterese und Impulsdauer von DO0, DO1 übernommen. Wenn die Werte sich nicht ändern sollen, müssen Sie die Altwerte übergeben.
CTRL_DO0	Freigabe DO0 Mit diesem Bit geben Sie den Ausgang DO0 frei.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
ENSET_DN	Mit diesem Bit geben Sie das Laden des Zählers beim Zählen in Vorwärtsrichtung frei.
ENSET_UP	Mit diesem Bit geben Sie das Laden des Zählers beim Zählen in Rückwärtsrichtung frei.
GATE_STP	Mit diesem Bit schließen Sie das interne Tor.

Steuerbit	Erläuterungen
L_DIRECT	Mit diesem Bit laden Sie den Zähler direkt und vorbereitend.
L_PREPAR	Mit diesem Bit laden Sie den Ladewert vorbereitend.
NEUSTQ	Mit diesem Bit quittieren Sie einen Anlauf der FM 350-1. Die FM 350-1 erkennt nach ihrem Neustart erst dann ein Steuern oder eine Dateneingabe an, wenn dieses Bit gesetzt wurde. Das Bit NEUSTQ wird von der Anweisung CNT_CNTL1 gesetzt, sobald das Rückmeldesignal FM_NEUST gesetzt ist und das Rückmeldesignal FM_NEUSTQ = 0 ist. Es wird von der Anweisung CNT_CNTL1 zurückgesetzt, wenn das Bit FM_NEUST von der FM 350-1 zurückgesetzt und das Bit FM_NEUSTQ von der FM 350-1 gesetzt wurde. Wenn Sie die Anweisung CNT_CNTL1 nicht verwenden, müssen Sie die Neustartkoordinierung über das Anwenderprogramm durchführen.
OT_ERR_A	Mit diesem Bit quittieren Sie einen Bedienfehler. Wenn Sie die detaillierte Bedienfehlerinformation benötigen, müssen Sie diese vor dem Quittieren aus der Rückmeldeschnittstelle lesen. Nach dem Quittieren ist die Fehlermeldung nicht mehr gültig.
RES_SYNC	Mit diesem Bit setzen Sie das Rückmeldebit STS_SYNC zurück und quittieren damit das Laden des Zählers durch den Synchronisationseingang DI-Set.
RES_ZERO	Mit diesem Bit setzen Sie die Rückmeldebits STS_ZERO, STS_OFLW, STS_UFLW, STS_COMP1 und STS_COMP2 zurück.
SET_DO0	Mit diesem Bit schalten Sie den Digitalausgang DO0 ein und aus, wenn Sie für den Ausgang das Verhalten "Kein Vergleich" parametriert haben und wenn das Freigabebit CRTL_DO0 gesetzt ist.
SET_DO1	Mit diesem Bit schalten Sie den Digitalausgang DO1 ein und aus, wenn Sie für den Ausgang das Verhalten "Kein Vergleich" parametriert haben und wenn das Freigabebit CRTL_DO1 gesetzt ist.
SW_GATE	Mit diesem Bit öffnen und schließen Sie das SW-Tor.
T_CMP_V1	Mit diesem Bit laden Sie den Wert aus Byte 4 bis 7 in den Vergleichswert 1.
T_CMP_V2	Mit diesem Bit laden Sie den Wert aus Byte 8 bis 11 in den Vergleichswert 2.

Rückmeldeschnittstelle für die Zählbetriebsarten

Rückmeldeschnittstelle für die Zählbetriebsarten

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) der FM 350-1 für die Zählbetriebsarten.

Offset zur Anfangsadresse	Parameter	Bedeutung
Byte 0 bis 3	LATCH_LOAD	Rücklesbarer Ladewert oder abgespeicherter Zählwert bei Latch-Funktion am Digitaleingang
Byte 4 bis 7	ACT_CNTV	Zählerstand
Byte 8 bis 9	DA_ERR_W	Datenfehler
Byte 10	OT_ERR_B	Bedienfehler

Offset zur Anfangs-adresse	Parameter	Bedeutung
Byte 11	PARA	Bit 7: Parametrierung erfolgt
	FM_NEUST	Bit 6: Neustartanforderung
	FM_NEUSTQ	Bit 5: Neustartquittung erfolgt
	DATA_ERR	Bit 4: Datenfehler
	OT_ERR	Bit 3: Bedienfehler
	DIAG	Bit 2: Diagnoseereignis
	–	Bit 1: Reserve = 0
	–	Bit 0: Reserve = 0
Byte 12	–	Reserve = 0
Byte 13	STS_SW_GATE	Bit 7: Status des SW-Tores
	STS_GATE	Bit 6: Status des Tores
	STS_SYNC	Bit 5: Synchronisation
	STS_UFLW	Bit 4: Unterschreiten der Untergrenze
	STS_OFLW	Bit 3: Überschreiten der Obergrenze
	STS_ZERO	Bit 2: Nulldurchgang
	STS_DIR	Bit 1: Richtungsbit
	STS_RUN	Bit 0: Zähler läuft
Byte 14	STS_COMP2	Bit 7: Remanenter Status des Vergleichers 2
	STS_COMP1	Bit 6: Remanenter Status des Vergleichers 1
	STS_CMP2	Bit 5: Status des Ausgangs DO1
	STS_CMP1	Bit 4: Status des Ausgangs DO0
	STS_STP	Bit 3: Status des Digitaleingangs DI-Stop
	STS_STA	Bit 2: Status des Digitaleingangs DI-Start
	STS_LATCH	Bit 1: Neuer Latchwert bei taktsynchronem Betrieb
	STS_SET	Bit 0: Status des Digitaleingangs DI-Set
Byte 15	–	Bit 7: Reserve = 0
	STS_C_DOPARA	Bit 6: Funktion DO0/DO1, Hysterese oder Impulsdauer ändern
	STS_RES_ZERO	Bit 5: Statusbit Nulldurchgang, Überschreiten, Unterschreiten oder Vergleicher rücksetzen
	STS_RES_SYNC	Bit 4: Statusbit Synchronisation rücksetzen
	STS_T_CMP_V2	Bit 3: Vergleichswert 2 laden
	STS_T_CMP_V1	Bit 2: Vergleichswert 1 laden
	STS_L_PREPAR	Bit 1: Zähler vorbereitend laden
	STS_L_DIRECT	Bit 0: Zähler direkt und vorbereitend laden

Erläuterungen zu den Rückmeldebits für die Zählbetriebsarten

Rückmeldebit	Erläuterungen
DATA_ERR	Dieses Bit zeigt an, dass ein Datenfehler (Parametrierfehler) in der Rückmeldeschnittstelle eingetragen wurde.
DIAG	Das Bit wird gesetzt, wenn der Diagnosedatensatz DS1 aktualisiert wurde, um ein Diagnoseereignis zu melden. Nach dem Lesen des Datensatzes DS1 wird das Bit wieder zurückgesetzt. Wenn kein Diagnosealarm freigegeben ist, kann dieses Bit als Anstoß-Bit für die im OB 1 eingebundene Anweisung DIAG_INF verwendet werden.
FM_NEUST	Die FM 350-1 setzt dieses Bit jedes Mal, wenn sie einen Neustart durchführt oder einen Anlauf des Systems erkennt, egal ob das System automatisch oder manuell anläuft. Mit der nächsten positiven Flanke am Bit NEUSTQ wird das Bit FM_NEUST zurückgesetzt. Danach ist die FM 350-1 steuerbar und Werte können von ihr gelesen und zu ihr übertragen werden.
FM_NEUSTQ	Die FM 350-1 löscht dieses Bit jedes Mal, wenn sie einen Neustart durchführt oder einen Anlauf des Systems erkennt, egal ob das System automatisch oder manuell anläuft. Es wird gesetzt, wenn das Bit FM_NEUST zurückgesetzt wird.
OT_ERR	Dieses Bit wird gesetzt, wenn ein Bedienfehler in der Rückmeldeschnittstelle eingetragen wurde. Es wird zurückgesetzt, wenn das Bit OT_ERR_A gesetzt wird. Solange das Bit OT_ERR gesetzt ist, kann kein weiterer Bedienfehler gemeldet werden.
PARA	Dieses Bit wird gesetzt, wenn die Baugruppe fehlerfrei parametriert wurde. Der Parameterdatensatz ist fehlerfrei auf der Baugruppe. Das Bit wird aber erst gesetzt, wenn das Bit FM_NEUSTQ gesetzt wurde. Ab diesem Zeitpunkt sind die Werte in der Rückmeldeschnittstelle gültig und aktuell.
STS_C_DOPARA	Quittungsbit für das gleichzeitige Ändern des Verhaltens von DO0 und DO1, der Hysterese und der Impulsdauer. Wenn die Werte sich nicht ändern sollen, müssen Sie die Altwerte übergeben.
STS_CMP1	Status des Ausgangs DO0
STS_CMP2	Status des Ausgangs DO1
STS_T_CMP_V1	Quittungsbit für das Laden des Vergleichswerts 1
STS_T_CMP_V2	Quittungsbit für das Laden des Vergleichswerts 2
STS_COMP1	Dieses Bit zeigt den gespeicherten Zustand an, dass der Ausgang DO0 gesetzt war. Dies gilt auch für den Fall, dass der Ausgang DO0 nicht mit CTRL_DO0 freigegeben wurde. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.
STS_COMP2	Dieses Bit zeigt den gespeicherten Zustand an, dass der Ausgang DO1 gesetzt war. Dies gilt auch für den Fall, dass der Ausgang DO1 nicht mit CTRL_DO1 freigegeben wurde. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.
STS_DIR	Dieses Bit zeigt die Zählrichtung des Zählers an: 0 = vorwärts (LED DIR ist aus) 1 = rückwärts (LED DIR leuchtet)
STS_GATE	Dieses Bit zeigt den Zustand des Tors an. 0 = Tor geschlossen 1 = Tor offen
STS_LATCH	Dieses Bit zeigt im taktsynchronen Betrieb an, ob zwischen dem vorletzten Ti und dem letzten Ti mindestens ein neuer Latchwert abgespeichert wurde. Ist das Bit gesetzt, steht in LATCH_LOAD der letzte Latchwert. Wurde kein neuer Latchwert abgespeichert, ist das Bit nicht gesetzt. Im nicht taktsynchronen Betrieb ist das Bit nicht gesetzt.
STS_L_DIRECT	Quittungsbit für das direkte und vorbereitende Laden des Zählers und des Ladewerts.
STS_L_PREPAR	Quittungsbit für das vorbereitende Laden des Ladewertes.
STS_OFLW	Dieses Bit zeigt an, dass ein Überschreiten aufgetreten ist. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.

Rückmeldebit	Erläuterungen
STS_RES_SYNC	Rücksetzen des Rückmeldebit STS_SYNC.
STS_RES_ZERO	Quittungsbit für das Rücksetzen der gespeicherten Zustände in den Rückmeldebits STS_ZERO, STS_OFLW, STS_UFLW, STS_COMP1 und STS_COMP2
STS_RUN	Dieses Bit entspricht dem Bit 2^0 des Zählers. 0 = LED CR ist aus 1 = LED CR leuchtet
STS_SET	Status des Digitaleingangs DI-Set
STS_STA	Status des Digitaleingangs DI-Start
STS_STP	Status des Digitaleingangs DI-Stop
STS_UFLW	Dieses Bit zeigt an, dass ein Unterschreiten aufgetreten ist. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.
STS_SYNC	Dieses Bit zeigt den gespeicherten Zustand an, dass der Zähler durch ein Ereignis am DI-Set (Synchronisation) geladen wurde. Der gespeicherte Zustand wird durch Quittieren mit RES_SYNC zurückgesetzt.
STS_ZERO	Dieses Bit zeigt den gespeicherten Zustand an, dass der Zählerstand einen Nulldurchgang hatte. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.

Steuerschnittstelle für die Messbetriebsarten

Steuerschnittstelle für die Messbetriebsarten

Der Parameter LOAD_VAL (Byte 0 bis 3) hat zweierlei Bedeutung:

- Wenn Sie das Bit L_PREPAR setzen, dann wird LOAD_VAL als unterer Grenzwert interpretiert.
- Wenn Sie das Bit C_DOPARA setzen, dann können Sie im Byte 0 das Verhalten des Ausgangs DO0 festlegen.

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) der FM 350-1 für die Messbetriebsarten.

Offset zur Anfangs-adresse	Parameter	Belegung			
Byte 0 bis 3	LOAD_VAL	Unteren Grenzwert laden mit Bit L_PREPAR			
Byte 0		Verhalten von Ausgang DO0 festlegen mit Bit C_DOPARA			
		Bit 2 bis 7	Bit 1	Bit 0	Verhalten von Ausgang DO0
		irrelevant	0	0	Kein Vergleich
		irrelevant	0	1	Außerhalb der Grenzwerte
		irrelevant	1	0	Unterschreiten des unteren Grenzwertes
		irrelevant	1	1	Überschreiten des oberen Grenzwertes
Byte 1		Reserve = 0			
Byte 2		Reserve = 0			
Byte 3		Reserve = 0			
Byte 4 bis 7	CMP_V1	Oberen Grenzwert; laden mit Bit T_CMP_V1			
Byte 8 bis 9	CMP_V2	Aktualisierungszeit; laden mit Bit T_CMP_V2			
Byte 10 bis 11	–	–			

Offset zur Anfangs-adresse	Parameter	Belegung
Byte 12	–	Bit 7: Reserve = 0
	NEUSTQ	Bit 6: Neustartquittung
	–	Bit 5: Reserve = 0
	–	Bit 4: Reserve = 0
	OT_ERR_A	Bit 3: Bedienfehlerquittung
	–	Bit 2: Reserve = 0
	–	Bit 1: Reserve = 0
	–	Bit 0: Reserve = 0
Byte 13	–	Bit 7: Reserve = 0
	–	Bit 6: Reserve = 0
	–	Bit 5: Reserve = 0
	–	Bit 4: Reserve = 0
	SW_GATE	Bit 3: Steuerbit SW-Tor
	GATE_STP	Bit 2: Genereller Tor-Stopp
	–	Bit 1: Reserve = 0
	–	Bit 0: Reserve = 0
Byte 14	–	Bit 7: Reserve = 0
	–	Bit 6: Reserve = 0
	–	Bit 5: Reserve = 0
	–	Bit 4: Reserve = 0
	SET_DO1	Bit 3: Steuerbit DO1
	SET_DO0	Bit 2: Steuerbit DO0
	CTRL_DO1	Bit 1: Freigabe DO1
	CTRL_DO0	Bit 0: Freigabe DO0
Byte 15	–	Bit 7: Reserve = 0
	C_DOPARA [1]	Bit 6: Funktion DO0 ändern
	RES_ZERO	Bit 5: Statusbits für Überschreiten, Unterschreiten, Messende rücksetzen
	–	Bit 4: Reserve = 0
	T_CMP_V2 [2]	Bit 3: Aktualisierungszeit ändern
	T_CMP_V1 [2]	Bit 2: Oberen Grenzwert laden
	L_PREPAR [2]	Bit 1: Unteren Grenzwert laden
	–	Bit 0: –

[1] Dieses Bit dürfen Sie nicht gleichzeitig mit Bit 1, 2, oder 3 von Byte 15 setzen.
[2] Dieses Bit dürfen Sie nicht gleichzeitig mit Bit 6 von Byte 15 setzen.

Erläuterungen zu den Steuerbits für die Messbetriebsarten

Steuerbit	Erläuterungen
C_DOPARA	Mit diesem Bit ändern Sie die Funktion und das Verhalten von DO0. Die Werte aus Byte 0 werden als neue Funktion von DO0 übernommen. Wenn die Werte sich nicht ändern sollen, müssen Sie die Altwerte übergeben.
CTRL_DO0	Freigabe DO0 Mit diesem Bit geben Sie den Ausgang DO0 frei.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
GATE_STP	Mit diesem Bit schließen Sie das interne Tor.
L_PREPAR	Mit diesem Bit laden Sie den unteren Grenzwert.
NEUSTQ	Mit diesem Bit quittieren Sie einen Anlauf der FM 350-1. Die FM 350-1 erkennt nach ihrem Neustart erst dann ein Steuern oder eine Dateneingabe an, wenn dieses Bit gesetzt wurde. Das Bit NEUSTQ wird von der Anweisung CNT_CNTL1 gesetzt, sobald das Rückmeldesignal FM_NEUST gesetzt ist und das Rückmeldesignal FM_NEUSTQ = 0 ist. Es wird von der Anweisung CNT_CNTL1 zurückgesetzt, wenn das Bit FM_NEUST von der FM 350-1 zurückgesetzt und das Bit FM_NEUSTQ von der FM 350-1 gesetzt wurde. Wenn Sie die Anweisung CNT_CNTL1 nicht verwenden, müssen Sie die Neustartkoordinierung über das Anwenderprogramm durchführen.
OT_ERR_A	Mit diesem Bit quittieren Sie einen Bedienfehler. Wenn Sie die detaillierte Bedienfehlerinformation benötigen, müssen Sie diese vor dem Quittieren aus der Rückmeldeschnittstelle lesen. Nach dem Quittieren ist die Fehlermeldung nicht mehr gültig.
RES_ZERO	Mit diesem Bit setzen Sie die Rückmeldebit STS_OFLW, STS_UFLW und STS_COMP1 zurück.
SET_DO0	Mit diesem Bit schalten Sie den Digitalausgang DO0 ein und aus, wenn Sie für den Ausgang das Verhalten "Kein Vergleich" parametriert haben und wenn das Freigabebit CRTL_DO0 gesetzt ist.
SET_DO1	Mit diesem Bit schalten Sie den Digitalausgang DO1 ein und aus, wenn Sie für den Ausgang das Verhalten "Kein Vergleich" parametriert haben und wenn das Freigabebit CRTL_DO1 gesetzt ist.
SW_GATE	Mit diesem Bit öffnen und schließen Sie das SW-Tor.
T_CMP_V1	Mit diesem Bit laden Sie den oberen Grenzwert.
T_CMP_V2	Mit diesem Bit laden Sie die Aktualisierungszeit.

Rückmeldeschnittstelle für die Messbetriebsarten

Rückmeldeschnittstelle für die Messbetriebsarten

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) der FM 350-1 für die Messbetriebsarten.

Offset zur Anfangsadresse	Parameter	Belegung
Byte 0 bis 3	LATCH_LOAD	Messwert
Byte 4 bis 7	ACT_CNTV	Zählerstand
Byte 8 bis 9	DA_ERR_W	Datenfehler
Byte 10	OT_ERR_B	Bedienfehler

Offset zur Anfangs-adresse	Parameter	Belegung
Byte 11	PARA	Bit 7: Parametrierung erfolgt
	FM_NEUST	Bit 6: Neustartanforderung
	FM_NEUSTQ	Bit 5: Neustartquittung erfolgt
	DATA_ERR	Bit 4: Datenfehler
	OT_ERR	Bit 3: Bedienfehler
	DIAG	Bit 2: Diagnoseereignis
	–	Bit 1: Reserve = 0
	–	Bit 0: Reserve = 0
Byte 12		Reserve = 0
Byte 13	–	Bit 7: Reserve = 0
	STS_GATE	Bit 6: Status des Tores
	–	Bit 5: Reserve = 0
	STS_UFLW	Bit 4: Unterlauf
	STS_OFLW	Bit 3: Überlauf
	STS_COMP1	Bit 2: Messende
	STS_DIR	Bit 1: Richtungsbit
	STS_RUN	Bit 0: Zähler läuft
Byte 14	–	Bit 7: Reserve = 0
	–	Bit 6: Reserve = 0
	STS_CMP2	Bit 5: Status des Ausgangs DO1
	STS_CMP1	Bit 4: Status des Ausgangs DO0
	STS_STP	Bit 3: Status des Digitaleingangs DI-Stop
	STS_STA	Bit 2: Status des Digitaleingangs DI-Start
	–	Bit 1: Reserve = 0
	STS_SET	Bit 0: Status des Digitaleingangs DI-Set
Byte 15	–	Bit 7: Reserve = 0
	STS_C_DOPARA	Bit 6: Funktion DO0 ändern
	STS_RES_ZERO	Bit 5: Statusbit Messende rücksetzen
	–	Bit 4: Reserve = 0
	STS_T_CMP_V2	Bit 3: Aktualisierungszeit ändern
	STS_T_CMP_V1	Bit 2: Oberen Grenzwert laden
	STS_L_PREPAR	Bit 1: Unteren Grenzwert laden
	–	Bit 0: Reserve = 0

Erläuterungen zu den Rückmeldebits für die Messbetriebsarten

Rückmeldebit	Erläuterungen
DATA_ERR	Dieses Bit zeigt an, dass ein Datenfehler in der Rückmeldeschnittstelle eingetragen wurde.
DIAG	Das Bit wird gesetzt, wenn der Diagnosedatensatz DS1 aktualisiert wurde, um ein Diagnoseereignis zu melden. Nach dem Lesen des Datensatzes DS1 wird das Bit wieder zurückgesetzt. Wenn kein Diagnosealarm freigegeben ist, kann dieses Bit als Anstoß-Bit für die im OB 1 eingebundene Anweisung DIAG_INF verwendet werden.

Rückmeldebit	Erläuterungen
FM_NEUST	Die FM 350-1 setzt dieses Bit jedes Mal, wenn sie einen Neustart durchführt oder einen Anlauf des Systems erkennt, egal ob das System automatisch oder manuell anläuft. Mit der nächsten positiven Flanke am Bit NEUSTQ wird das Bit FM_NEUST zurückgesetzt. Danach ist die FM 350-1 steuerbar und Werte können von ihr gelesen und zu ihr übertragen werden.
FM_NEUSTQ	Die FM 350-1 löscht dieses Bit jedes Mal, wenn sie einen Neustart durchführt oder einen Anlauf des Systems erkennt, egal ob das System automatisch oder manuell anläuft. Es wird gesetzt, wenn das Bit FM_NEUST zurückgesetzt wird.
OT_ERR	Dieses Bit wird gesetzt, wenn ein Bedienfehler in der Rückmeldeschnittstelle eingetragen wurde. Es wird zurückgesetzt, wenn das Bit OT_ERR_A gesetzt wird. Solange das Bit OT_ERR gesetzt ist, kann kein weiterer Bedienfehler gemeldet werden.
PARA	Dieses Bit wird gesetzt, wenn die Baugruppe fehlerfrei parametriert wurde. Der Parameterdatensatz ist fehlerfrei auf der Baugruppe. Das Bit wird aber erst gesetzt, wenn das Bit FM_NEUSTQ gesetzt wurde. Ab diesem Zeitpunkt sind die Werte in der Rückmeldeschnittstelle gültig und aktuell.
STS_C_DOPARA	Quittungsbit für das gleichzeitige Ändern des Verhaltens von DO0 und DO1, der Hysterese und der Impulsdauer. Wenn die Werte sich nicht ändern sollen, müssen Sie die Altwerte übergeben.
STS_CMP1	Status des Ausgangs DO0
STS_CMP2	Status des Ausgangs DO1
STS_CMP_T_VAL1	Quittungsbit für das Laden des oberen Grenzwertes
STS_CMP_T_VAL2	Quittungsbit für das Laden der Aktualisierungszeit
STS_DIR	Dieses Bit zeigt die Zählrichtung des Zählers an: 0 = vorwärts (LED DIR ist aus) 1 = rückwärts (LED DIR leuchtet)
STS_GATE	Dieses Bit zeigt den Zustand des Tors an. 0 = Tor geschlossen 1 = Tor offen
STS_L_PREPAR	Quittungsbit für das Laden des unteren Grenzwertes
STS_OFLW	Dieses Bit zeigt den gespeicherten Zustand an, dass ein Messwert größer als der obere Grenzwert war. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.
STS_RES_ZERO	Quittungsbit für das Rücksetzen der gespeicherten Zustände in den Rückmeldebits STS_OFLW, STS_UFLW und STS_COMP1
STS_RUN	Dieses Bit entspricht dem Bit 2^0 des Zählers. 0 = LED CR ist aus 1 = LED CR leuchtet
STS_SET	Status des Digitaleingangs DI-Set
STS_STA	Status des Digitaleingangs DI-Start
STS_STP	Status des Digitaleingangs DI-Stop
STS_UFLW	Dieses Bit zeigt den gespeicherten Zustand an, dass ein Messwert kleiner als der untere Grenzwert war. Der gespeicherte Zustand wird durch Quittieren mit RES_ZERO zurückgesetzt.

FM 350-1 aus dem Anwenderprogramm verwenden

Einbinden der FM 350-1 in ein Anwenderprogramm

Für das Einbinden der FM 350-1 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen CNT_CTRL, CNT_CTL1 und CNT_CTL2 werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 350-1.

Alle zu einem Kanal der Baugruppe gehörenden Daten liegen im DB der Anweisung CNT_CTRL, CNT_CTL1 bzw. CNT_CTL2 (Zähler-DB). Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CNT_CHANTYPE1 in Ihr Projekt ein.

Außerdem gibt es für die FM 350-1 die Anweisung DIAG_INF, mit der Sie Diagnoseinformationen in den Zähler-DB übertragen können.

Die Anweisung CNT_CTL2 wird nur im taktsynchronen Betrieb verwendet.

FM 350-1: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Sammelfehler-LED der FM 350-1

Wo wird der Fehler angezeigt?

Leuchtet die rote Sammelfehler-LED, dann ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten.

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Sammelfehler-LED angezeigt werden.

Fehlerart	Fehlerursache	Behebung
Interne Fehler	Fehler beim EPROM-Test	Baugruppentausch
	Fehler beim RAM-Test	Baugruppentausch
	Zeitüberwachung (Watchdog) hat angesprochen	Baugruppentausch
	Verlorener Prozessalarm	Zeitlichen Abstand zwischen den Alarmursachen vergrößern
	Parametrierung für die Baugruppe fehlt	Parametrierung vorgeben und übertragen

Fehlerart	Fehlerursache	Behebung
Externe Fehler	Kodierstecker falsch gesteckt	Stellung des Kodiersteckers korrigieren
	Hilfsspannung 1L+ / 1M nicht angeschlossen oder Geberversorgung DC24V kurzgeschlossen	Anschluss korrigieren
	Geberversorgung DC5.2V kurzgeschlossen oder überlastet	Anschluss korrigieren
	5 V-Gebersignale fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)	Anschluss korrigieren
	Parametrierung für die Baugruppe passt nicht zur Stellung des Kodiersteckers	Parametrierung korrigieren und übertragen oder Kodierstecker umstecken

Auslösen eines Diagnosealarms

Alle Fehler, bis auf den Fehler beim EPROM-Test, können einen Diagnosealarm auslösen, falls Sie bei der Parametrierung den Diagnosealarm in den Grundparametern freigegeben haben. Aus den Diagnosedatensätzen DS0 und DS1 (Seite 262) erfahren Sie, welcher der Fehler das Leuchten der LED verursacht hat.

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, können Sie einen Diagnosealarm parametrieren, der das zyklische Programm der CPU unterbricht und den Diagnosealarm-OB aufruft.

Welche Ereignisse können einen Diagnosealarm auslösen?

Folgende Ereignisse können einen Diagnosealarm auslösen:

- Externe Hilfsspannung 1L+ / 1M ist fehlerhaft
- Geberversorgung DC5.2V kurzgeschlossen oder überlastet
- Parametrierung der Baugruppe fehlt
- Parametrierung der Baugruppe ist fehlerhaft
- Zeitüberwachung (Watchdog) angesprochen
- RAM defekt
- Prozessalarm verloren
- Signal A fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)
- Signal B fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)

- Signal N fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)
- Kodierstecker falsch gesteckt

Freigeben des Diagnosealarms

Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.

Reaktionen auf einen Diagnosealarm

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht Folgendes:

- Die Diagnoseinformationen werden in die Diagnosedatensätze DS0 und DS1 (Seite 262) auf der Baugruppe abgelegt.
- Die Sammelfehler-LED leuchtet.
  Wenn Sie den Fehler behoben haben, erlischt die Sammelfehler-LED.
- Der Diagnosealarm-OB wird aufgerufen (z. B. OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Der Zählvorgang läuft unverändert weiter, wenn ein Diagnosealarm-OB programmiert ist. Ohne Diagnosealarm-OB geht die CPU in STOP.

Voreinstellung

In der Voreinstellung sind keine Diagnosealarme freigegeben.

Diagnosedatensatz DS0 und DS1

Diagnosedatensatz DS0 und DS1

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Byte, der DS1 umfasst 16 Byte, wobei die ersten vier Byte identisch mit dem DS0 sind.

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 (Seite 263) wird automatisch beim Aufruf des Diagnose-OB in die Startinformation übertragen. Dort werden diese vier Byte ins Lokaldatum (Byte 8 bis 11) des Diagnosealarm-OB abgelegt.

Den Diagnosedatensatz DS1 (Seite 264) (und damit auch den Inhalt des DS0) können Sie mit der Anweisung DIAG_INF von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Belegung des Diagnosedatensatzes DS0

Belegung des Diagnosedatensatzes DS0 und der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Fehler beim RAM-Test • Zeitüberwachung (Watchdog) angesprochen • Verlorener Prozessalarm	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Hilfsspannung 1L+ / 1M nicht angeschlossen oder Geberversorgung DC5.2V kurzgeschlossen • Geberversorgung DC5.2V kurzgeschlossen oder überlastet • 5 V-Signale fehlerhaft • Parametrierung fehlerhaft	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe DS1, Byte 4 (Seite 264)	8:x:03
	4	Externe Hilfsspannung fehlerhaft	Spannung überprüfen	8:x.04
	6	Parametrierung fehlt	Parametrierung durchführen	8:x:06
	7	Parametrierung fehlerhaft	Weitere Aufschlüsselung siehe Datenfehler (Seite 264)	8:x:07
1	0...3	Typklasse	Ist immer mit 8 belegt	
	4	Kanalinformation	Ist immer mit 1 belegt	
2	3	Zeitüberwachung (Watchdog) angesprochen	Baugruppe defekt oder starke Störeinflüsse	8:x:33
3	3	RAM defekt	Baugruppe defekt oder starke Störeinflüsse	8:x:43
	6	Prozessalarm verloren	Projektierung überprüfen. Prozessalarmereignis wurde erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht von der CPU quittiert wurde.	8:x:46

Belegung des Diagnosedatensatzes DS1

Belegung des Diagnosedatensatzes DS1

Der Diagnosedatensatz DS1 besteht aus 16 Byte. Die ersten 4 Byte sind identisch mit dem Diagnosedatensatz DS0. Die folgende Tabelle zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null. Dieser Datensatz wird von der Anweisung DIAG_INF in den DB der Anweisung CNT_CTL1 ab DW54 eingetragen.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
4	0...6	Kanaltyp	Ist immer mit 76H belegt	
	7	Weitere Kanaltypen	Ist immer mit 0 belegt	
5	0...7	Länge der Diagnoseinformation	Ist immer mit 10H belegt	
6	0...7	Kanalanzahl	Ist immer mit 1 belegt	
7	0	Kanalfehlervektor	Bei Kanalfehler mit 1 belegt	
8	0	Signal A fehlerhaft		8:x:B0
	1	Signal B fehlerhaft		8:x:B1
	2	Signal N fehlerhaft		8:x:B2
	4	Geberversorgung 5,2 V fehlerhaft		8:x:B4
	5...7	Reserviert		
9 ... 15		Reserviert		

Datenfehler

Wann treten Datenfehler auf?

Werden an die Baugruppe neue Parameter übertragen, dann prüft die FM 350-1 diese Parameter. Treten bei dieser Prüfung Fehler auf, meldet die Baugruppe diese Datenfehler.

Wo werden Datenfehler angezeigt?

Die Datenfehler werden von der Anweisung CNT_CTL1 mit Fehlernummer in den DB der CNT_CTL1 eingetragen. Über die Variable DA_ERR_W können Sie im Anwenderprogramm auf dieses Datenwort zugreifen.

Welche Datenfehler gibt es?

In der folgenden Tabelle sind die Datenfehler mit Fehlernummer und Bedeutung aufgelistet.

Nr.	Bedeutung
0	Kein Fehler
200	Kodierstecker steckt falsch oder fehlt
201	Die Stellung des Kodiersteckers entspricht nicht dem parametrierten Geber
202	Wert für Diagnose des Signalpaars falsch
203	Wert für Auswertung der Signale falsch
204	Wert für Eingangsfilter der 24 V-Zählsignale falsch

Nr.	Bedeutung
205	Wert für Eingangsfilter der Digitaleingänge falsch
206	Richtungsumkehr nicht erlaubt
207	Verhalten von DO0 falsch parametriert
208	Verhalten von DO1 falsch parametriert
209	Bereichsüberschreitung bei Impulsdauer
211	Falsche Betriebsart angewählt
212	Kein Tor oder beide Tore angegeben
213	Hauptzählrichtung falsch parametriert
214	Bereichsüberschreitung bei oberer Zählgrenze
215	Bei der Parametrierung des Prozessalarms "Erreichen des Vergleichswertes vorwärts bzw. rückwärts" wurde eine unterschiedliche Zählrichtung zur Parametrierung des Verhaltens der Ausgänge "Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts bzw. rückwärts" gewählt. Die Richtungen müssen übereinstimmen.
216	Alarme von der Torsteuerung nur bei Betriebsarten mit Torsteuerung möglich
217	Beim Verhalten der Ausgänge "Einschalten bei Zählerstand >= Vergleichswert" oder "Einschalten bei Zählerstand <= Vergleichswert" ist ein Alarm bei "Erreichen der Vergleichswerte" nicht erlaubt.
218	Alarm bei Nulldurchgang nicht erlaubt
219	Kodierung der Einstellung "Latchen" ist falsch
220	Torsteuerung falsch parametriert
221	Bereichsüberschreitung bei unterem Grenzwert bzw. Ladewert
222	Bereichsüberschreitung bei oberem Grenzwert bzw. Vergleichswert 1
223	Bereichsüberschreitung bei Aktualisierungszeit bzw. Vergleichswert 2
224	Bereichsüberschreitung bei Impulse pro Geberumdrehung

Wie werden Datenfehler quittiert?

Korrigieren Sie die Werte für die Parameter entsprechend den Vorgaben. Übertragen Sie den korrigierten Parametersatz wieder an die FM 350-1. Diese prüft erneut die Parameter und löscht den Datenfehler im DB.

Bedienfehler

Wann treten Bedienfehler auf?

Bedienfehler treten auf, wenn Sie die Baugruppe durch inkorrekte Vorgabe von Steuersignalen falsch bedienen.

Wo werden die Bedienfehler angezeigt?

Die Bedienfehlernummern werden von der Anweisung CNT_CTL1 in den DB eingetragen. Mit dem Setzen des Ausgangsparameters OT_ERR zeigt die Anweisung CNT_CTL1, dass ein Bedienfehler aufgetreten ist. Über die Variable OT_ERR_B können Sie in Ihrem Programm auf dieses Datenbyte zugreifen.

Welche Bedienfehler gibt es?

In der folgenden Tabelle sind die Bedienfehler mit Fehlernummer und Bedeutung aufgelistet.

Nr.	Bedeutung
0	Kein Fehler
1	Betriebsart kann mit SW-Tor nicht gestartet werden
2	Betriebsart kann nicht abgebrochen werden
4	Nur im STOP der CPU erlaubt
5	Nur das Steuerbit für das Parametrieren darf gesetzt werden
6	Nicht erlaubter Auftrag
10	Bereichsüberschreitung bei unterem Grenzwert bzw. Ladewert
11	Bereichsüberschreitung bei oberem Grenzwert bzw. Vergleichswert 1
12	Bereichsüberschreitung bei Aktualisierungszeit bzw. Vergleichswert 2
20	Verhalten von DO0 falsch parametriert
21	Verhalten von DO1 falsch parametriert
22	Bereichsüberschreitung bei Impulsdauer
90	Siehe CNT_CTL2
91	Siehe CNT_CTL2

Wie werden Bedienfehler quittiert?

Quittieren Sie den Fehler mit dem Eingangsparameter OT_ERR_A der Anweisung CNT_CTL1 bzw. CNT_CTL2.

1.2.1.3 FM 350-2 einsetzen

Anschlussbelegung des Frontsteckers der FM 350-2

Frontstecker

Am 40-poligen Frontstecker schließen Sie an:

- Zählsignale
- Digitaleingänge
- Digitalausgänge
- Geberversorgung und Baugruppenspannung

Das folgende Bild zeigt die Vorderseite der Baugruppe, den Frontstecker und die Innenseite der Fronttür mit dem Aufdruck der Anschlussbelegung.

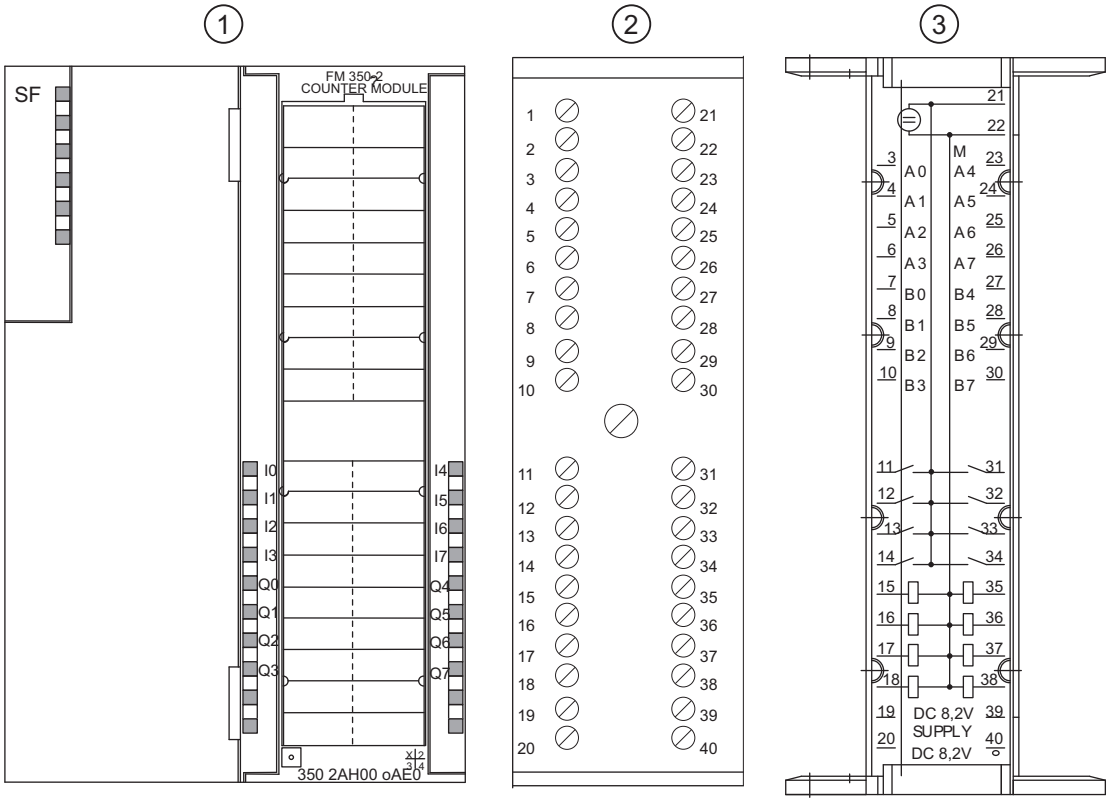

① Vorderseite der Baugruppe
② Frontstecker
③ Innenseite der Fronttür

Belegung des Frontsteckers

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 350-2.

Anschluss	Name	Ein- / Ausgang	Funktion
1	-	-	nicht angeschlossen
2	-	-	nicht angeschlossen
3	A0	Eingang	Kanal 0 Zähleingang NAMUR / BERO
4	A1	Eingang	Kanal 1 Zähleingang NAMUR / BERO
5	A2	Eingang	Kanal 2 Zähleingang NAMUR / BERO
6	A3	Eingang	Kanal 3 Zähleingang NAMUR / BERO
7	B0	Eingang	Kanal 0 Richtungseingang BERO
8	B1	Eingang	Kanal 1 Richtungseingang BERO
9	B2	Eingang	Kanal 2 Richtungseingang BERO
10	B3	Eingang	Kanal 3 Richtungseingang BERO
11	I0	Eingang	Kanal 0 HW-Tor-Eingang BERO
12	I1	Eingang	Kanal 1 HW-Tor-Eingang BERO
13	I2	Eingang	Kanal 2 HW-Tor-Eingang BERO
14	I3	Eingang	Kanal 3 HW-Tor-Eingang BERO

Anschluss	Name	Ein- / Ausgang	Funktion
15	Q0	Ausgang	Kanal 0 Digitalausgang 0,5 A
16	Q1	Ausgang	Kanal 1 Digitalausgang 0,5 A
17	Q2	Ausgang	Kanal 2 Digitalausgang 0,5 A
18	Q3	Ausgang	Kanal 3 Digitalausgang 0,5 A
19	P8V2	Ausgang	Geberversorgung NAMUR 8,2 V
20	P8V2	Ausgang	Geberversorgung NAMUR 8,2 V
21	L+	Eingang	24 V-Baugruppenversorgung
22	M	Eingang	Masse Baugruppenversorgung
23	A4	Eingang	Kanal 4 Zähleingang NAMUR / BERO
24	A5	Eingang	Kanal 5 Zähleingang NAMUR / BERO
25	A6	Eingang	Kanal 6 Zähleingang NAMUR / BERO
26	A7	Eingang	Kanal 7 Zähleingang NAMUR / BERO
27	B4	Eingang	Kanal 4 Richtungseingang BERO
28	B5	Eingang	Kanal 5 Richtungseingang BERO
29	B6	Eingang	Kanal 6 Richtungseingang BERO
30	B7	Eingang	Kanal 7 Richtungseingang BERO
31	I4	Eingang	Kanal 4 HW-Tor-Eingang BERO
32	I5	Eingang	Kanal 5 HW-Tor-Eingang BERO
33	I6	Eingang	Kanal 6 HW-Tor-Eingang BERO
34	I7	Eingang	Kanal 7 HW-Tor-Eingang BERO
35	Q4	Ausgang	Kanal 4 Digitalausgang 0,5 A
36	Q5	Ausgang	Kanal 5 Digitalausgang 0,5 A
37	Q6	Ausgang	Kanal 6 Digitalausgang 0,5 A
38	Q7	Ausgang	Kanal 7 Digitalausgang 0,5 A
39	P8V2	Ausgang	Geberversorgung NAMUR 8,2 V
40	P8V2	Ausgang	Geberversorgung NAMUR 8,2 V

Hinweis

Die Schaltungen für die Zählereingänge (Geberversorgung, Gebersignale) sind potenzialgetrennt zur Masse der CPU ausgeführt.

Sämtliche Eingänge sind untereinander potenzialgebunden und zum Bus der S7-300 potenzialgetrennt.

Weitere Informationen zu Aufbau und Verdrahtung finden Sie im Inbetriebnahme- und Montagehandbuch (http://support.automation.siemens.com/WW/view/de/1105178) der Zählerbaugruppe FM 350-2.

24 V-Spannungsversorgung

Für die Spannungsversorgung der FM 350-2 schließen Sie an den Anschlüssen L+ und M eine Gleichspannung von 24 V an.

Geberversorgung DC 8,2 V

Die Baugruppe erzeugt aus der 24 V-Versorgungsspannung eine Spannung von 8,2 V (max. 200 mA). Diese Spannung steht an den Anschlüssen P8V2 (Stifte 19, 20, 39 und 40) für die Spannungsversorgung der NAMUR-Geber kurzschlussfest zur Verfügung.

Die Geberversorgung wird auf 8,2 V überwacht.

Gebersignale A0 bis A7, B0 bis B7

Sie können vier verschiedene Gebertypen anschließen:

- NAMUR-Geber nach DIN 19234 (mit Diagnose):
  Die Signale werden an den Anschlüssen A0 bis A7 angeschlossen.
- 24 V-Inkrementalgeber:
  Die Signale A0/B0 bis A7/B7 werden über die entsprechend gekennzeichneten Anschlüsse angeschlossen.
- 24 V Impulsgeber mit Richtungspegel:
  Die Zählsignale werden an den Anschlüssen A0 bis A7 angeschlossen. Die Richtungspegel werden an den Anschlüssen B0 bis B7 angeschlossen.
- 24 V Initiatoren:
  Die Signale werden an den Anschlüssen A0 bis A7 angeschlossen.

Hinweis

Die Geberversorgung für die 24 V-Geber müssen Sie über eine externe Spannungsversorgung DC 24 V zuführen.

Digitaleingänge I0 bis I7 (HW-Tore)

Für die Torsteuerung des Zählers können Sie die Digitaleingänge I0 bis I7 benutzen.

Für jeden Zählkanal steht Ihnen ein Digitaleingang zur Verfügung, mit dem Sie den zugehörigen Zähler starten und stoppen können.

Die Digitaleingänge werden mit einer Nennspannung von 24 V betrieben.

Digitalausgänge Q0 bis Q7

Zum direkten Auslösen von Steuerungsvorgängen verfügt die FM 350-2 über die Digitalausgänge Q0 bis Q7.

Je Zähler steht Ihnen ein Digitalausgang zur Verfügung.

Die Digitalausgänge werden über die 24 V-Spannungsversorgung der FM 350-2 versorgt.

Die Digitalausgänge sind P-Schalter und mit einem Laststrom von 0,5 A belastbar. Sie sind gegen Überlast und Kurzschluss geschützt.

Hinweis

Der direkte Anschluss von Relais und Schützen ist ohne externe Beschaltung möglich.

FM 350-2 konfigurieren

FM 350-2 parametrieren

Kanäle spezifizieren

Kanaleinstellungen auswählen

Kanaleinstellung

In diesem Feld stellen Sie ein, welche Funktion die Baugruppenkanäle haben:

- Einzelzähler: In der ausgewählten Gruppe arbeitet jeder Kanal einzeln als Zähler.
- Dosierzähler: Jeweils 4 Zählkanäle werden zu einem Dosierkanal zusammengefasst. Von diesem Kanal sind bis zu 4 Dosiereinheiten ansteuerbar.

Sie haben folgende Auswahlmöglichkeiten:

- Kanal 0..7 als Einzelzähler
- Kanal 0 als Dosierzähler, Kanal 4..7 als Einzelzähler
- Kanal 0..3 als Einzelzähler, Kanal 4 als Dosierzähler
- Kanal 0 als Dosierzähler und Kanal 4 als Dosierzähler

Zählereingang für NAMUR-Geber freigeben

NAMUR-Geber

In diesem Feld können Sie freigeben, an welche Zählereingänge ein NAMUR-Geber angeschlossen wird.

Sie haben folgende Auswahlmöglichkeiten:

- Zählereingang Kanal 0..3 NAMUR
  und / oder
- Zählereingang Kanal 4..7 NAMUR

Siehe auch: "NAMUR-Signale (Seite 270)" und "Hardware-Überwachung für NAMUR-Geber freigeben (Seite 277)"

NAMUR-Signale

NAMUR-Geber

Der Geber liefert ein Zählsignal nach DIN 19234, das an den Anschluss A0 … A7 des Frontsteckers angeschlossen werden muss.

Sie können ein Signal zur Richtungserkennung (24 V) an den Anschluss B0 … B7 des betreffenden Zählers anschließen. Liefert Ihr Geber kein entsprechendes Signal, können Sie auch eine entsprechende Kennung innerhalb der S7 generieren und anschließen oder Sie können ein entsprechendes Prozesssignal verwenden.

Das folgende Bild zeigt den zeitlichen Verlauf der Signale des NAMUR-Gebers mit Richtungspegel.

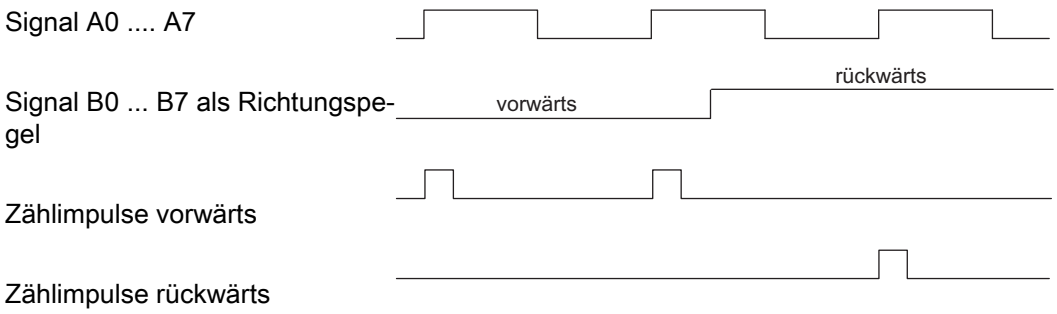

Sie können die Eingänge A0 … A3, A4 … A7 oder alle für den Anschluss von NAMUR-Gebern freigeben.

Wie werden die Signale überwacht?

Wenn Sie die FM 350-2 entsprechend parametriert haben, dann überwacht die Baugruppe, ob an A0 … A7 ein Drahtbruch oder ein Kurzschluss vorliegt. Indirekt wird dadurch auch die 8,2 V-Geberversorgung überwacht. Diese Überwachung ist nur bei NAMUR-Gebern möglich.

Bei Drahtbruch oder Kurzschluss leuchtet die LED Sammelfehler, zusätzlich können Sie einen Diagnosealarm parametrieren.

⚠ VORSICHT

Es kann zu einem Sachschaden kommen.

Wenn Sie an einem Kanal der FM 350-2, der für den Anschluss eines NAMUR-Gebers parametriert wurde, einen anderen Geber betreiben, kann die Baugruppe beschädigt werden.

Schließen Sie an einen Kanal der FM 350-2, der für den Anschluss eines NAMUR-Gebers parametriert wurde, nur einen NAMUR-Geber an.

Belegung der Rückmeldeschnittstelle auswählen

Belegung der Rückmeldeschnittstelle Byte 8 bis 15

In diesem Feld können Sie auswählen, wie Byte 8 bis 15 der Rückmeldeschnittstelle belegt werden.

Für die Variablen USER_STAT_WORD_0 bis USER_STAT_WORD_3 haben Sie folgende Auswahlmöglichkeiten:

- Datentyp
  - WORD
  - DWORD
- Kanal
  - Kanal 0 bis 7
- Wert
  - Zählwert
  - Messwert
- Word
  - Von einem Wert können entweder das Low-Word oder das High-Word oder beide abgelegt werden. Sie können auf diese Werte direkt zugreifen.

Abbildung der Zähl- und Messwerte aller Kanäle in den erweiterten Nutzdaten

Abbildung aller Zähl- bzw. Messwerte im Nutzdatenabbild

Im Nutzdatenabbild können unter bestimmten Voraussetzungen alle Zähl- bzw. Messwerte abgebildet werden (Verwendung der erweiterten Nutzdaten).

Für die Verwendung der erweiterten Nutzdaten gelten die folgenden Randbedingungen:

- Aus dem Baugruppenkatalog die Baugruppe "6ES7350-2AH01-0AE0 E" projektieren.
- Das Optionskästchen "Erweiterte Nutzdatenschnittstelle verwenden" muss aktiviert werden.
- Die erweiterten Nutzdaten können nur dezentral hinter einer IM 153-2 V5.0 (Artikelnummer 6ES7153-2BA01-0XB0 V5.0) oder höher verwendet werden.
- Es stehen nun 40 Byte Eingangsdaten zur Verfügung.
- Der Aktualisierungszyklus der erweiterten Nutzdaten unterscheidet sich vom Aktualisierungszyklus der ersten 16 Byte Eingangsdaten. Je nach PROFIBUS DP-Zykluszeit liegt der Aktualisierungszyklus der erweiterten Nutzdaten bei ca. 100 bis 200 ms.

Der Aufbau der ersten 16 Byte in den Nutzdaten ist wie bisher bei der jeweiligen Betriebsart. Die weiteren 24 Byte definieren sich über die Betriebsart.

Abhängig von der Betriebsart werden in den erweiterten Nutzdaten die Werte (Zähl- oder Messwert) der Kanäle 2 bis 7 abgelegt. Die Vorbelegung dieser Werte ist "0". Solange keine gültigen Werte erfasst wurden, sind die Eingangsdaten mit 0 belegt.

Aufbau der erweiterten Nutzdaten in Abhängigkeit der Betriebsart:

Byte	Betriebsarten Zählen	Betriebsarten Messen	Betriebsart Dosieren (zwei Dosierzähler)
0..15	Wie bisher – Aufbau beeinflussbar über Parametrierung	Wie bisher – Aufbau beeinflussbar über Parametrierung	Wie bisher – Aufbau beeinflussbar über Parametrierung
16..19	Zählwert Kanal 2	Messwert Kanal 2	0
20..23	Zählwert Kanal 3	Messwert Kanal 3	0
24..27	Zählwert Kanal 4	Messwert Kanal 4	Dosierzähler Kanal 4
28..31	Zählwert Kanal 5	Messwert Kanal 5	0
32..35	Zählwert Kanal 6	Messwert Kanal 6	0
36..39	Zählwert Kanal 7	Messwert Kanal 7	0

Kanäle kopieren

Kanäle kopieren

Sie können die Parameter eines Kanals auf einen oder mehrere andere Kanäle kopieren.

1. Wählen Sie den Kanal aus, dessen Parameter kopiert werden sollen (Kanal Quelle).
2. Aktivieren Sie die Kanäle, auf die die Parameter kopiert werden sollen (Kanal Ziel).
3. Drücken Sie die Schaltfläche "Kopieren".

Kanal n Betriebsarten

Betriebsart auswählen

Beschreibung

Mit der Vorgabe einer Betriebsart wählen Sie, mit welcher Funktionalität Sie die Zählerbaugruppe betreiben wollen.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne Tor)	Die Zählerbaugruppe zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit HW- oder SW-Tor)	Die Zählerbaugruppe zählt mit dem Öffnen des Tores von Null bis zur parametrierten oberen Zählgrenze bzw. von der parametrierten oberen Zählgrenze bis Null.
Periodisches Zählen (Seite 187) (mit HW- oder SW-Tor)	Die Zählerbaugruppe zählt mit dem Öffnen des Tores zwischen Null und der parametrierten oberen Zählgrenze.
Frequenzmessung (Seite 195) (mit HW- oder SW-Tor)	Die Zählerbaugruppe erfasst die Impulse, die in einem parametrierten Zeitfenster von n × 10 ms eintreffen, und bestimmt daraus die aktuelle Frequenz.

Betriebsart	Beschreibung
Drehzahlmessung (Seite 196) (mit HW- oder SW-Tor)	Die Zählerbaugruppe erfasst die Impulse, die in einem parametrierten Zeitfenster von n × 10 ms eintreffen, und bestimmt aus der parametrierten Anzahl von Impulsen pro Geberumdrehung die Drehzahl.
Periodendauermessung (Seite 198) (mit HW- oder SW-Tor)	Die Zählerbaugruppe erfasst die Impulse, die in einem parametrierten Zeitfenster von n × 10 ms eintreffen, und bestimmt daraus die Periodendauer.
Dosieren (Seite 200) (mit HW- oder SW-Tor)	In dieser Betriebsart werden jeweils vier Zählkanäle zu einem Dosierkanal zusammengefasst. Funktionell ist diese Betriebsart identisch zur Betriebsart "Einmalig Zählen".

Die Voreinstellung ist Betriebsart "Endlos Zählen".

Obere Zählgrenze festlegen

Beschreibung

Mit der Parametrierung einer oberen Zählgrenze in Verbindung mit einer Hauptzählrichtung (vorwärts oder rückwärts) grenzen Sie den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Die obere Zählgrenze kann einen Wert > 1 bis zur oberen Zählbereichsgrenze annehmen.

Messzeit festlegen

Beschreibung

Wenn Sie eine Messbetriebsart ausgewählt haben, dann aktualisiert die Baugruppe die Messwerte zyklisch. Die Integrationszeit ist das Produkt aus Messzeit × 10 ms. Die Messzeit geben Sie als Faktor n vor.

Detaillierte Beschreibungen zu den Wertebereichen für die Messzeit finden Sie unter "Grundsätzliches zum Messen (Seite 190)".

HW-Tor verwenden

HW-Tor verwenden

Sie können wählen, ob der Kanal mit oder ohne Hardware-Tor zählen soll.

Aktivieren Sie für Zählen mit Hardware-Tor das Optionskästchen "HW-Tor verwenden".

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Hauptzählrichtung auswählen

Beschreibung

Mit der Parametrierung einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie durch Parametrieren einer oberen Zählgrenze den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Eine Hauptzählrichtung können Sie nur in den Betriebsarten "Einmaliges Zählen", "Periodisches Zählen" und "Dosieren" auswählen.

Sie können folgendes Verhalten auswählen:

- Vorwärts
- Rückwärts

Kanal n Geber

Signalart des Gebers auswählen

Signalart auswählen

1. Wählen Sie die Signalart des eingesetzten Gebers aus.
   Abhängig von Ihrer Auswahl und der Betriebsart sind diejenigen Optionen gesperrt, die für diesen speziellen Geber nicht möglich sind. Eine gesperrte Option wird durch graue Schrift im entsprechenden Auswahlfenster gekennzeichnet.
2. Wählen Sie die Optionen aus, die Sie mit dem ausgewählten Geber nutzen wollen.

Sie können die Signalart von einem der folgenden Geber wählen:

- Geber: 24 V inkrementell (Seite 276) (24 V Spur A+B phasenversetzt)
- Geber: 24 V Impuls und Richtung (Seite 276)

Geber: 24 V inkrementell

24 V-Inkrementalgeber

Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A* und B* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung (Seite 210)	Einfach	Nur eine Flanke von Signal A wird ausgewertet.
	Zweifach	Beide Flanken von Signal A werden ausgewertet.
	Vierfach	Beide Flanken von Signal A und B werden ausgewertet.
Zählrichtung (Seite 276)	Normal	Zur Anpassung der Zählrichtung an den Prozess
	Invertiert	Zur Anpassung der Zählrichtung an den Prozess

Geber: 24 V Impuls und Richtung

24 V-Impulsgeber

Wählen Sie die Signalart "24 V Impuls und Richtung", wenn Sie einen 24 V-Impulsgeber mit dem Zählsignal A* und dem Richtungssignal B* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung	–	Nicht relevant
Zählrichtung (Seite 276)	Invertiert	Zur Anpassung der Zählrichtung an den Prozess
	Normal	Zur Anpassung der Zählrichtung an den Prozess

Zählrichtung auswählen

Zählrichtung ändern

Sie können mit dem Parameter "Zählrichtung normal" bzw. "Zählrichtung invertiert" die Zählrichtung ändern, ohne die Verdrahtung ändern zu müssen.

Durch die Parametrierung "Hauptzählrichtung rückwärts (Seite 188)" wird die Zählrichtung **nicht** automatisch umgedreht.

Impulse pro Geberumdrehung eintragen

Impulse pro Geberumdrehung

Für die Betriebsart "Drehzahlmessung (Seite 196)" müssen Sie zusätzlich die Impulse pro Geberumdrehung parametrieren.

Den Wert für die Anzahl der Impulse pro Geberumdrehung finden Sie auf dem Typenschild oder in den technischen Daten Ihres Gebers.

Hysterese festlegen

Prinzipielle Wirkungsweise der Hysterese

Ein Geber kann an einer bestimmten Position stehenbleiben und dann um diese Position "pendeln". Dieser Zustand führt dazu, dass der Zählerstand um einen bestimmten Wert herum schwankt. Liegt nun in diesem Schwankungsbereich zum Beispiel ein Vergleichswert, würde der zugehörige Ausgang im Rhythmus dieser Schwankungen ein- und ausgeschaltet werden. Um dieses Schalten bei kleinen Schwankungen zu verhindern, ist die FM 350-2 mit einer parametrierbaren Hysterese ausgestattet.

Sie können für die Hysterese Werte zwischen 0 und 255 parametrieren.

Abhängig vom parametrierten Wert wirkt die Hysterese wie folgt:

Hysterese	Wirkung
Hysteresewert n = 0, 1	Die Hysterese ist unwirksam (abgeschaltet). Der Ausgang reagiert auf die kleinste Änderung des Zählerstands.
2 ≤ Hysteresewert n ≤ 255	Die Hysterese ist wirksam. Der Ausgang reagiert erst, wenn der Zählerstand sich um n Einheiten vom Vergleichswert entfernt hat.

Die Hysterese wirkt auch bei Überlauf und Unterlauf.

Hardware-Überwachung für NAMUR-Geber freigeben

Wie werden die Signale von NAMUR-Gebern überwacht?

Die FM 350-2 kann überwachen, ob an A0 … A7 ein Drahtbruch oder ein Kurzschluss vorliegt. Indirekt wird dadurch auch die 8,2 V-Geberversorgung überwacht.

Aktivieren Sie das Optionskästchen "Hardware-Überwachung für NAMUR-Geber", wenn die FM 350-2 die Gebersignale überwachen soll.

Kanal n Ausgänge

Vergleichswerte und Ausgänge

Beschreibung

Sie können auf der FM 350-2 acht Vergleichswerte ablegen, die den acht Digitalausgängen zugeordnet sind (Vergleichswert 0: Q0, Vergleichswert 1: Q1, usw.). Abhängig von Zählerstand und Vergleichswert kann der entsprechende Ausgang gesetzt werden.

Bei der Betriebsart "Dosieren (Seite 200)" können Sie für jeden Dosierkanal der FM 350-2 vier Vergleichswerte parametrieren.

Vergleichswerte

Die Vergleichswerte tragen Sie bei der Parametrierung ein.

Im RUN der CPU können Sie die Vergleichswerte im DB der Anweisung CNT2_WR / CNT2WRPN eintragen (CMP_VAL0…7) und an die FM 350-2 übergeben, indem Sie die Aufträge 30...37 bzw. 60...62 verwenden. Der Zählvorgang wird davon nicht beeinflusst.

Die Vergleichswerte müssen innerhalb des Zählbereichs der jeweiligen Betriebsart liegen, die Zählbereichsgrenzen sind nicht zulässig.

Wenn Sie bei einer Betriebsart mit Hauptzählrichtung dennoch den Vergleichswert auf den Start- oder Endwert legen, zeigen die Digitalausgänge folgendes Verhalten:

Parametriertes Verhalten der Digitalausgänge	Reaktion des zugeordneten Digitalausgangs
Einschalten bei Zählerstand >= Vergleichswert	Der Ausgang ist unabhängig vom Zählerstand immer eingeschaltet
Einschalten bei Zählerstand <= Vergleichswert	Der Ausgang ist unabhängig vom Zählerstand immer ausgeschaltet

Freigeben der Ausgänge

Bevor die Ausgänge angesteuert werden können, müssen Sie sie zuerst freigeben, indem Sie die entsprechenden Bits im Zähler-DB setzen. Wenn Sie eines dieser Bits zurücksetzen, wird der zugehörige Ausgang sofort abgeschaltet. Die Bits werden zwischen DB und Baugruppe mit der Anweisung CNT2_CTR übertragen.

Ausgang	...wird freigegeben durch
Q0…7	CTRL_DQ0…7

Abschalten der Ausgänge

Die Ausgänge werden unabhängig von der Parametrierung durch folgende Ereignisse abgeschaltet:

- Zeitüberwachung (Watchdog) der Baugruppe angesprochen (interner Fehler)
- Wegnahme der Freigabebits (CTRL_DQ0...7 für Q0...7 im DB)

Steuern der Ausgänge

Unabhängig vom Schalten der Ausgänge über die Vergleicher können Sie die Ausgänge nach der Freigabe mit CTRL_DQ0…7 mit den Steuersignalen SET_DQ0…7 setzen und rücksetzen.

Für das Verhältnis von Steuern und Schalten gilt: Das Steuern hat Vorrang vor dem Schalten durch die Vergleicher. Dies bedeutet:

- Ist für den Ausgang kein Schalten durch den Vergleicher vorgesehen, so kann der Ausgang wie ein Digitalausgang verwendet werden.
- Haben Sie für die Ausgänge eine Vergleichsfunktion gewählt, können Sie die Ausgänge mit SET_DQ0…7 weiterhin steuern. Damit können Sie die Wirkung der Vergleichsfunktionen über Ihr Steuerungsprogramm simulieren:
  - Mit der positiven Flanke von SET_DQ0…7 wird der Ausgang gesetzt.
  - Eine negative Flanke von SET_DQ0…7 setzt den Ausgang wieder zurück.

  Beachten Sie, dass die Vergleicher weiterhin aktiv sind und den Ausgang bei Änderung des Vergleichsergebnisses setzen bzw. rücksetzen können.

  Hinweis

  Ein durch SET_DQ0…7 gesetzter Ausgang wird durch den Vergleicher nicht zurückgesetzt.

Verhalten der Ausgänge bei den Zählbetriebsarten

Verhalten der Ausgänge festlegen

Verhalten der Ausgänge DQ0 bis DQ7

Wählen Sie das gewünschte Verhalten des jeweiligen Ausgangs. Aktivieren Sie hierzu das entsprechende Optionsfeld:

- Kein Vergleich (Seite 280)
- Einschalten bei Zählerstand >= Vergleichswert (Seite 280)
- Einschalten bei Zählerstand <= Vergleichswert (Seite 280)

Hinweis

Um die Einstellungen zu aktivieren, müssen Sie im DB mit den Zählerdaten Werte für die Vergleichswerte eingetragen und mit der Anweisung CNT2_WR / CNT2WRPN zur Baugruppe übertragen haben; außerdem müssen Sie die Bits CTRL_DQ0…7 im DB setzen, um die Ausgänge freizugeben. Siehe auch "Vergleichswerte und Ausgänge (Seite 277)".

Hinweis

In den Betriebsarten "Frequenzmessung", "Drehzahlmessung" und "Periodendauermessung" können Sie die Ausgänge nicht parametrieren.

Voreinstellung der Ausgänge

In der Voreinstellung sind die Ausgänge abgeschaltet.

Kein Vergleich

Kein Vergleich

Der Ausgang bleibt ausgeschaltet und wird durch die Ereignisse Vergleichswert, Nulldurchgang, Sprung von Endwert auf Startwert (Überlauf) oder Sprung von Startwert auf Endwert (Unterlauf) nicht beeinflusst.

Einschalten bei Zählerstand >= Vergleichswert

Einschalten bei Zählerstand >= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler im Bereich zwischen Vergleichswert und Endwert befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Endwert setzt den Ausgang auf 1.

Die schraffierte Fläche In der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

Randbedingung:

Wenn Sie einen Ausgang "Einschalten bei Zählerstand >= Vergleichswert" parametrieren wollen, dann gilt folgende Randbedingung: Die Zeit zwischen dem Erreichen des Vergleichswertes und des Start- bzw. Endwertes muss größer sein als die minimale Schaltzeit der Ausgänge (Schaltzeit: 300 µs). Sonst gehen die Steuerimpulse an den Ausgängen verloren.

Siehe auch

Wirkung der Hysterese (Seite 281)

Einschalten bei Zählerstand <= Vergleichswert

Einschalten bei Zählerstand <= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler im Bereich zwischen Vergleichswert und Startwert befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Startwert setzt den Ausgang auf 1.

Die schraffierte Fläche In der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

Randbedingung:

Wenn Sie einen Ausgang "Einschalten bei Zählerstand <= Vergleichswert" parametrieren wollen, dann gilt folgende Randbedingung: Die Zeit zwischen dem Erreichen des Vergleichswertes und des Start- bzw. Endwertes muss größer sein als die minimale Schaltzeit der Ausgänge (Schaltzeit: 300 µs). Sonst gehen die Steuerimpulse an den Ausgängen verloren.

Siehe auch

Wirkung der Hysterese (Seite 281)

Wirkung der Hysterese

Wirkungsweise der Hysterese bei Parametrierung "Einschalten bei Zählerstand >= Vergleichswert oder <= Vergleichswert"

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Die Parametrierung für dieses Beispiel ist:

- Hauptzählrichtung: "Vorwärts"
- Verhalten des Ausgangs: "Einschalten bei Zählerstand >= Vergleichswert"

Mit dem Erreichen der Vergleichsbedingung (Zählerstand = 5) wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, im Beispiel bei Zählwert 2 bzw. 8, ist die Hysterese nicht mehr aktiv. Der Vergleicher schaltet wieder entsprechend seinen Vergleichsbedingungen, im Beispiel bei Zählwert 5.

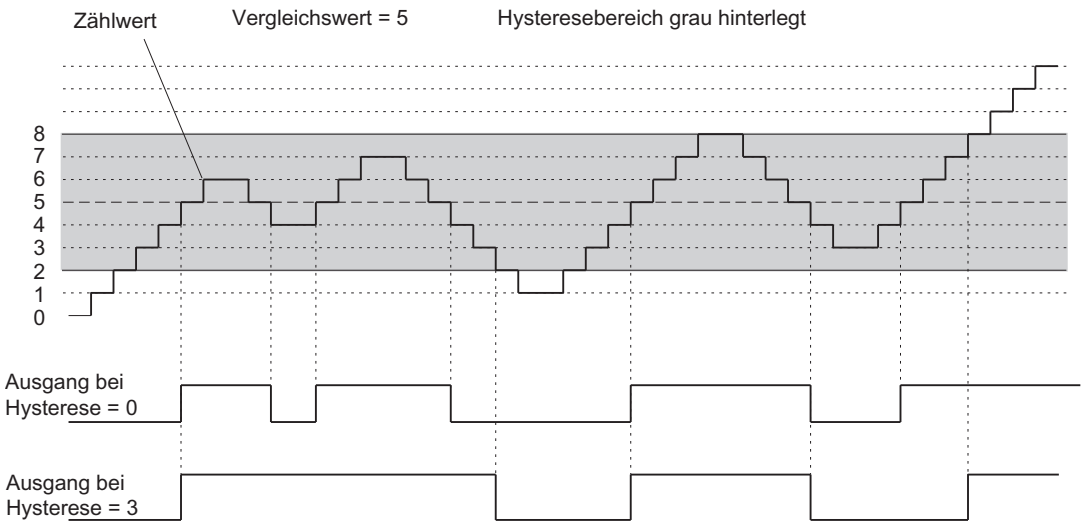

Hinweis

Ist der Zählerstand gleich dem Vergleichswert und die Hysterese aktiv, setzt die FM 350-2 den Ausgang zurück, wenn ein Wechsel der Zählrichtung auf dem Vergleichswert erfolgt (siehe folgendes Bild).

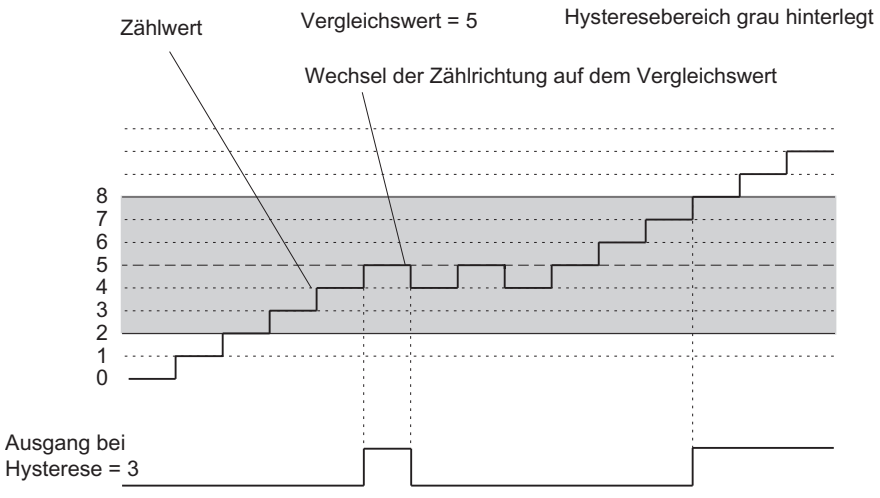

Ersatzwert '1' bei CPU-STOP freigeben

Ersatzwert '1' bei CPU-STOP

Wenn Sie in den Grundparametern der Baugruppe bei "Verhalten bei CPU-STOP" die Option "Ersatzwert" gewählt haben und die CPU in STOP geht, gibt die Baugruppe an den Digitalausgängen die parametrierten Ersatzwerte aus. Aktivieren Sie z. B. das Optionskästchen für DQ0, dann wird bei einem STOP der CPU der Ausgang Q0 gesetzt.

In der Betriebsart "Dosieren" können Sie bis zu vier Ersatzwerte einstellen, in den anderen Zählbetriebsarten können Sie einen Ersatzwert einstellen.

Die Ersatzwerte bleiben nach dem STOP-RUN-Übergang der CPU bis zur nächsten Bedienung an den Digitalausgängen gültig.

Wenn Sie den Grundparameter "Verhalten bei CPU-STOP" ändern, werden die Ausgänge rückgesetzt.

⚠ VORSICHT
Es kann zu einem Sachschaden kommen. Parametrierte Ersatzwerte werden bei STOP der CPU immer ausgegeben, auch wenn das Tor geschlossen ist und auch auf Ausgängen, die nicht freigegeben sind. Parametrieren Sie ausschließlich Ersatzwerte, deren Ausgabe nicht zu gefährlichen Anlagenzuständen führen kann.

Kanal n Freigabe Prozessalarme

Auslösen eines Prozessalarms freigeben

Einleitung

Sie können bei der FM 350-2 einstellen, welche Ereignisse im Betrieb einen Prozessalarm auslösen sollen.

Was ist ein Prozessalarm?

Soll auf ein Ereignis unabhängig von dem Zyklus der CPU reagiert werden, kann die FM 350-2 einen Prozessalarm auslösen. Die CPU unterbricht daraufhin das zyklische Programm und bearbeitet den Prozessalarm-OB.

Freigeben des Prozessalarms

Das Freigeben von Prozessalarmen erfolgt in zwei Schritten:

1. Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.
2. Unter "Freigabe Prozessalarme" müssen Sie dann die entsprechenden Ereignisse auswählen:
   - Ereignisse für das Auslösen eines Prozessalarms bei den Zählbetriebsarten (Seite 284)
   - Ereignisse für das Auslösen eines Prozessalarms bei den Messbetriebsarten (Seite 285)

Prozessalarm-OB, z. B. OB 40

Tritt ein Prozessalarm auf, wird das Anwenderprogramm unterbrochen, die Daten von der Baugruppe in die Startinformation des OB 40 (Seite 285) übertragen und der OB 40 aufgerufen. Mit dem Verlassen des OB 40 wird der Prozessalarm quittiert.

Ist kein Prozessalarm-OB programmiert, so geht die CPU in STOP. Wird daraufhin wieder in RUN geschaltet, sind die Prozessalarmanforderungen gelöscht.

Verlorener Prozessalarm

Tritt ein Ereignis ein, das einen Prozessalarm auslösen soll, und das vorhergehende gleiche Ereignis wurde noch nicht quittiert, so wird kein weiterer Prozessalarm ausgelöst; der Prozessalarm geht verloren. Je nach Parametrierung kann das zu dem Diagnosealarm "Prozessalarm verloren" führen.

Liegt zwischen zwei Ereignissen, die einen Prozessalarm auslösen sollen, eine Zeit von < 2 ms, so geht der zweite Prozessalarm verloren, ohne dass ein Diagnosealarm ausgelöst werden kann.

Voreinstellung

In der Voreinstellung sind keine Prozessalarme freigegeben.

Ereignisse für das Auslösen eines Prozessalarms bei den Zählbetriebsarten

Welche Ereignisse können einen Prozessalarm bei den Zählbetriebsarten auslösen?

Bei den Zählbetriebsarten können folgende Ereignisse beim Betrieb der FM 350-2 einen Prozessalarm auslösen:

- Öffnen des HW-Tores
- Schließen des HW-Tores

- Überschreitung/Unterschreitung
- Erreichen des Vergleichswertes
  Tragen Sie den zugehörigen Vergleichswert in das Eingabefeld ein. Die Vergleichswerte müssen innerhalb des Zählbereichs der jeweiligen Betriebsart liegen, die Zählbereichsgrenzen sind nicht zulässig.

Aktivieren Sie die Optionskästchen für die betreffenden Ereignisse.

Sie können, abhängig von der parametrierten Betriebsart, mehrere Ereignisse für die Prozessalarmauslösung auswählen. Für die Prozessalarme bei Erreichen eines Vergleichswertes müssen Sie die Randbedingungen beachten (für "Einschalten bei Zählerstand >= Vergleichswert (Seite 280)" bzw. "Einschalten bei Zählerstand <= Vergleichswert (Seite 280)").

Ereignisse für das Auslösen eines Prozessalarms bei den Messbetriebsarten

Welche Ereignisse können einen Prozessalarm bei den Messbetriebsarten auslösen?

Bei den Messbetriebsarten können folgende Ereignisse beim Betrieb der FM 350-2 einen Prozessalarm auslösen:

- Öffnen des HW-Tores
- Schließen des HW-Tores
- Messende
- Unterschreiten des unteren Grenzwertes
  Tragen Sie den zugehörigen unteren Grenzwert in das Eingabefeld ein. Informationen zu den Wertebereichen für die unteren Grenzwerte finden Sie unter "Grundsätzliches zum Messen (Seite 190)".
- Überschreiten des oberen Grenzwertes
  Tragen Sie den zugehörigen oberen Grenzwert in das Eingabefeld ein. Informationen zu den Wertebereichen für die oberen Grenzwerte finden Sie unter "Grundsätzliches zum Messen (Seite 190)".

Aktivieren Sie die Optionskästchen für die betreffenden Ereignisse.

Sie können, abhängig von der parametrierten Betriebsart, mehrere Ereignisse für die Prozessalarmauslösung auswählen.

Startinformation des OB 40

Startinformation des OB 40

In der Startinformation des OB 40 wird die temporäre Variable OB40_POINT_ADDR beschrieben.

Die Variable OB40_POINT_ADDR (Byte 8 bis 11) umfasst 4 Byte. In diesen Bytes werden die Informationen über das Ereignis eingetragen, das den Prozessalarm ausgelöst hat.

Die folgenden Tabellen zeigen, welche Bits bei welchem Alarm gesetzt sind. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Betriebsart: Endlos Zählen, Einmaliges Zählen, Periodisches Zählen

Kanal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8					Vergleichswert erreicht	Überschreitung / Unterschreitung	Schließen HW-Tor	Öffnen HW-Tor
1		Vergleichswert erreicht	Überschreitung / Unterschreitung	Schließen HW-Tor	Öffnen HW-Tor				
2 bis 7	9 bis 11	siehe Byte 8							

Betriebsart: Frequenzmessung, Drehzahlmessung, Periodendauermessung

Kanal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8					Oberer Grenzwert überschritten / Unterer Grenzwert unterschritten	Messende	Schließen HW-Tor	Öffnen HW-Tor
1		Oberer Grenzwert überschritten / Unterer Grenzwert unterschritten	Messende	Schließen HW-Tor	Öffnen HW-Tor				
2 bis 7	9 bis 11	siehe Byte 8							

Betriebsart: Dosieren

Kanal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8	–	Überschreitung / Unterschreitung	Vergleichswert 4 erreicht	Vergleichswert 3 erreicht	Vergleichswert 2 erreicht	Vergleichswert 1 erreicht	Schließen HW-Tor	Öffnen HW-Tor
4	10	siehe Byte 8							

FM 350-2 in Betrieb nehmen

FM 350-2 aus dem Anwenderprogramm verwenden

Einbinden der FM 350-2 in ein Anwenderprogramm

Für das Einbinden der FM 350-2 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen CNT2WRPN, CNT2RDPN und CNT2_CTR werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 350-2.

Alle zu einem Kanal der Baugruppe gehörenden Daten liegen im DB der Anweisung CNT2_CTR, CNT2WRPN bzw. CNT2RDPN (Zähler-DB). Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CNT2_CHANTYPE in Ihr Projekt ein.

Außerdem gibt es für die FM 350-2 die Anweisung DIAG_RD, mit der Sie Diagnoseinformationen in den Zähler-DB übertragen können.

FM 350-2: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Sammelfehler-LED der FM 350-2

Wo wird der Fehler angezeigt?

Leuchtet die rote Sammelfehler-LED, so ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten oder die Parametrierung ist fehlerhaft.

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Sammelfehler-LED angezeigt werden.

Fehlerart	Fehlerursache	Behebung
Interne Fehler	Zeitüberwachung (Watchdog) hat angesprochen	Baugruppentausch
	Verlorener Prozessalarm	wird durch Bearbeitung des Prozessalarms quittiert
Externe Fehler	Parametrierung für die Baugruppe fehlt	Parametrierung vorgeben und übertragen
	Parametrierung ist fehlerhaft	
Externe Kanalfehler	Geberversorgung kurzgeschlossen oder überlastet	Anschluss korrigieren
	NAMUR-Gebersignale fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)	Anschluss korrigieren

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, können Sie einen Diagnosealarm parametrieren, der das zyklische Programm der CPU unterbricht und den Diagnosealarm-OB, OB 82, aufruft.

Welche Ereignisse können einen Diagnosealarm auslösen?

Folgende Ereignisse können einen Diagnosealarm auslösen:

- Kanalfehler bei parametriertem Kanal
- Parametrierung der Baugruppe fehlt
- Parametrierung der Baugruppe ist fehlerhaft
- Zeitüberwachung (Watchdog) angesprochen
- Prozessalarm verloren
- Signalleitungsüberwachung NAMUR-Geber meldet Fehler
- Geberversorgung 8,2 V NAMUR-Geber fehlerhaft

Freigeben des Diagnosealarms

Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.

Reaktionen auf einen Diagnosealarm

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht Folgendes:

- Die Diagnoseinformationen werden in die Diagnosedatensätze DS0 und DS1 (Seite 289) auf der Baugruppe abgelegt.
- Die Sammelfehler-LED leuchtet.
  Wenn Sie den Fehler behoben haben, erlischt die Sammelfehler-LED.
- Der Diagnosealarm-OB wird aufgerufen (OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Der Zählvorgang läuft unverändert weiter, wenn ein OB 82 programmiert ist. Ohne OB 82 geht die CPU in STOP.

Voreinstellung

In der Voreinstellung sind keine Diagnosealarme freigegeben.

Diagnosedatensatz DS0 und DS1

Diagnosedatensatz DS0 und DS1

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Byte, der DS1 umfasst 16 Byte, wobei die ersten vier Byte identisch mit dem DS0 sind.

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 (Seite 289) wird automatisch beim Aufruf des Diagnose-OB in die Startinformation übertragen. Dort werden diese vier Byte ins Lokaldatum (Byte 8 bis 11) des OB 82 abgelegt.

Den Diagnosedatensatz DS1 (Seite 290) (und damit auch den Inhalt des DS0) können Sie mit der Anweisung DIAG_RD von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Belegung des Diagnosedatensatzes DS0

Belegung des Diagnosedatensatzes DS0 und der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe DS 1, Byte 4 (Seite 290)	8:x:03
	6	Parametrierung fehlt	Parametrierung durchführen	8:x:06
	7	Parametrierung fehlerhaft	Weitere Aufschlüsselung siehe Datenfehler (Seite 290)	8:x:07
1	0 ... 3	Typklasse	Ist immer mit 8 belegt	
	4	Kanalinformation	Ist immer mit 1 belegt	
2	3	Zeitüberwachung (Watchdog) angesprochen	Baugruppe defekt oder starke Störeinflüsse	8:x:33
3	6	Prozessalarm verloren	Projektierung überprüfen. Prozessalarmereignis wurde erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht von der CPU quittiert wurde.	8:x:46

Belegung des Diagnosedatensatzes DS1

Belegung des Diagnosedatensatzes DS1

Der Diagnosedatensatz DS1 besteht aus 16 Byte. Die ersten 4 Byte sind identisch mit dem Diagnosedatensatz DS0. Nachfolgende Tabelle zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null. Dieser Datensatz wird von der Anweisung DIAG_RD in den DB der Anweisung CNT2_CTR ab DW212 eingetragen.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
4	0 ... 6	Kanaltyp	Ist immer mit 76H belegt	
	7	Weitere Kanaltypen	Ist immer mit 0 belegt	
5	0 ... 7	Länge der Diangnoseinformation	Ist immer mit 8 belegt	
6	0 ... 7	Kanalanzahl	Ist immer mit 8 belegt	
7	0	Kanalfehlervektor	Kanalbit	
8	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 0)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
9	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 1)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
10	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 2)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
11	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 3)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
12	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 4)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
13	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 5)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
14	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 6)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96
15	4	Geberversorgung 8,2 V fehlerhaft	NAMUR-Geber	8:x:94
(Kanal 7)	6	Signalleitung NAMUR-Geber	Kurzschluss/Drahtbruch	8:x:96

Datenfehler

Wo treten Datenfehler auf?

Werden an die Baugruppe vom PG oder mit der Anweisung CNT2WRPN Aufträge gegeben, dann werden diese geprüft. Treten bei dieser Prüfung Fehler auf, meldet die Baugruppe diese Datenfehler.

Fehlerhafte Aufträge werden von der Baugruppe nicht übernommen.

Wo werden Datenfehler angezeigt?

Die Datenfehler werden in der Parametrieroberfläche angezeigt.

Wenn bei der Überprüfung von Aufträgen ein Fehler festgestellt wird, dann wird im Zähler-DB das Bit CHECKBACK_SIGNALS, DATA_ERR = 1 gesetzt.

Wie werden Datenfehler quittiert?

Korrigieren Sie die Aufträge entsprechend den Vorgaben. Übertragen Sie die korrigierten Aufträge wieder an die FM 350-2.

Diagnosepuffer der FM 350-2

Einträge im Diagnosepuffer der FM 350-2 werden in der Parametrieroberfläche angezeigt.

Fehlercodes im Diagnosepuffer

Fehlercodes

Nachfolgend sind die Fehlernummern mit den zugehörigen Fehlerbeschreibungen aufgelistet. Im Diagnosepuffer wird ein Fehler mit der Fehler-ID in folgender Form eingetragen:

16#Fx83 yyzz

Dabei bedeutet:

- yy = Kennung für die Fehlerklasse, z. B. 04 für einen Kanalparametrierfehler
- zz = Fehlernummer, z. B. 18 für den Fehler "unzulässige Hauptzählrichtung"

Kanalparametrierfehler	Fehler-nummer	Beschreibung
04	01	Auftrag nicht erlaubt, da Kanal für Dosieren belegt
04	18	unzulässige Hauptzählrichtung
04	19	Zählbereich fehlerhaft
04	1A	Vergleichswert liegt nicht im erreichbaren Zählbereich
04	1B	unzulässiges Ausgangsverhalten
04	1C	Prozessalarmfreigabe nicht möglich
04	1D	unzulässige Parametrierung der Hardware-Überwachung
04	1E	unzulässige Parametrierung des HW-Tores
04	1F	Signalauswahl nicht möglich
04	20	Signalauswahl fehlerhaft
04	21	Hysteresewert zu groß
04	22	Ersatzwert unzulässig
04	23	Reaktion beim Schließen eines Tores unzulässig
04	24	Reservierter Parameter ungleich 0
04	25	Reservierter Parameter ungleich 0
04	26	Reservierter Parameter ungleich 0
04	27	Geberauswahl fehlerhaft
04	28	Diagnose des Zähleingangs nur bei NAMUR-Geber möglich

Kanalparametrierfehler	Fehler-nummer	Beschreibung
04	29	Ladewert liegt nicht im zulässigen Zählbereich
04	2B	Der Zählbereich kann nicht vergrößert werden
04	2C	Vergleichswert 1 liegt nicht im Zählbereich
04	2D	Vergleichswert 2 liegt nicht im Zählbereich
04	2E	Vergleichswert 3 liegt nicht im Zählbereich
04	2F	Vergleichswert 4 liegt nicht im Zählbereich
04	30	Unterer Grenzwert für Frequenz oder Drehzahl unzulässig
04	31	Oberer Grenzwert für Frequenz oder Drehzahl unzulässig
04	32	Unterer Grenzwert >= Oberer Grenzwert
04	33	Integrationszeit unzulässig
04	34	Impulse pro Geberumdrehung unzulässig
04	35	Unterer Grenzwert für Periodendauer unzulässig
04	36	Oberer Grenzwert für Periodendauer unzulässig

Baugruppenparamet-rierfehler	Fehler-nummer	Beschreibung
05	0A	unzulässige Alarmgenerierung
05	0B	unzulässige Alarmauswahl
05	0C	unzulässige Reaktion bei aktivem output disable
05	0D	Reservierter Parameter ungleich 0
05	18	Zeitstempel unzulässig
05	19	unzulässige Gebersignalauswahl
05	1A	unzulässige Kanalbetriebsart
05	1B	unzulässige Aufteilung von CHECKBACK_SIG-NALS.USER_STAT im Datenbaustein

Datenfehler	Fehler-nummer	Beschreibung
06	18	unzulässige Hauptzählrichtung
06	19	Zählbereich fehlerhaft
06	1A	Vergleichswert liegt nicht im erreichbaren Zählbereich
06	1B	unzulässiges Ausgangsverhalten
06	1C	Freigabe Prozessalarme nicht möglich
06	1D	unzulässige Parametrierung der Hardware-Überwachung
06	1E	unzulässige Parametrierung des HW-Tores
06	1F	Signalauswahl nicht möglich
06	20	Signalauswahl fehlerhaft
06	21	Hysteresewert zu groß
06	22	Ersatzwert unzulässig
06	23	Reaktion beim Schließen eines Tores unzulässig
06	24	Reservierter Parameter ungleich 0

Datenfehler	Fehler-nummer	Beschreibung
06	25	Reservierter Parameter ungleich 0
06	26	Reservierter Parameter ungleich 0
06	27	Geberauswahl fehlerhaft
06	28	Diagnose des Zähleingangs nur bei NAMUR-Geber möglich
06	29	Ladewert liegt nicht im zulässigen Zählbereich
06	2B	Der Zählbereich kann nicht vergrößert werden
06	2C	Vergleichswert 1 liegt nicht im Zählbereich
06	2D	Vergleichswert 2 liegt nicht im Zählbereich
06	2E	Vergleichswert 3 liegt nicht im Zählbereich
06	2F	Vergleichswert 4 liegt nicht im Zählbereich
06	30	Unterer Grenzwert für Frequenz oder Drehzahl unzulässig
06	31	Oberer Grenzwert für Frequenz oder Drehzahl unzulässig
06	32	Unterer Grenzwert >= Oberer Grenzwert
06	33	Integrationszeit unzulässig
06	34	Impulse pro Geberumdrehung unzulässig
06	35	Unterer Grenzwert für Periodendauer unzulässig
06	36	Oberer Grenzwert für Periodendauer unzulässig

Meldung	Fehler-nummer	Beschreibung
0F	01	Ende des Baugruppenanlaufs
0F	02	keine Parameter auf der Baugruppe
0F	03	Baugruppe ist default parametriert
0F	04	Baugruppe ist parametriert
0F	05	PG-Betrieb ist eingestellt
0F	06	PG-Betrieb ist abgewählt
0F	07	PG-Betrieb ist ausgefallen
0F	08	PG-Betrieb ist eingestellt, kein Auftrag von der CPU zulässig
0F	09	fehlerhafte Auftragssequenz

Diagnosefehler	Fehler-nummer	Beschreibung
80	04	Externe Hilfsspannung fehlt
80	05	Frontstecker fehlt
80	06	Baugruppe nicht parametriert
80	07	Falsche Parameter in Baugruppe
80	30	Anwendermodul falsch/fehlt
80	31	Kommunikationsstörung
80	32	Betriebszustand STOP
80	33	Zeitüberwachung angesprochen
80	34	BG-interne Versorgungsspannung ausgefallen

Diagnosefehler	Fehler-nummer	Beschreibung
80	35	Batterie 1 leer
80	36	Gesamte Pufferung ausgefallen
80	41	Prozessorausfall
80	42	EPROM-Fehler
80	43	RAM-Fehler
80	44	ADU/DAU - Fehler
80	45	Sicherungsfehler
80	46	Prozessalarm ging verloren
80	90	Gebersignal A am Kanal fehlerhaft
80	91	Gebersignal B am Kanal fehlerhaft
80	92	Gebersignal N am Kanal fehlerhaft
80	93	fehlerhafter Wert zwischen den Kanälen
80	94	Geberversorgung 5,2 V oder NAMUR am Kanal fehlerhaft
80	95	Geberversorgung 24 V fehlerhaft
80	96	NAMUR-Geber Kurzschluss/Drahtbruch am Kanal

1.2.1.4 FM 450-1 einsetzen

Anschlussbelegung des Frontsteckers der FM 450-1

Frontstecker

Am 48-poligen Frontstecker schließen Sie an:

- Zählsignale
- Digitaleingänge
- Digitalausgänge
- Geberversorgung
- Hilfsspannung und Lastspannung

Das folgende Bild zeigt die Vorderseite des Frontsteckers, den Streifen mit dem Aufdruck der Anschlussbelegung und die Beschriftungsstreifen.

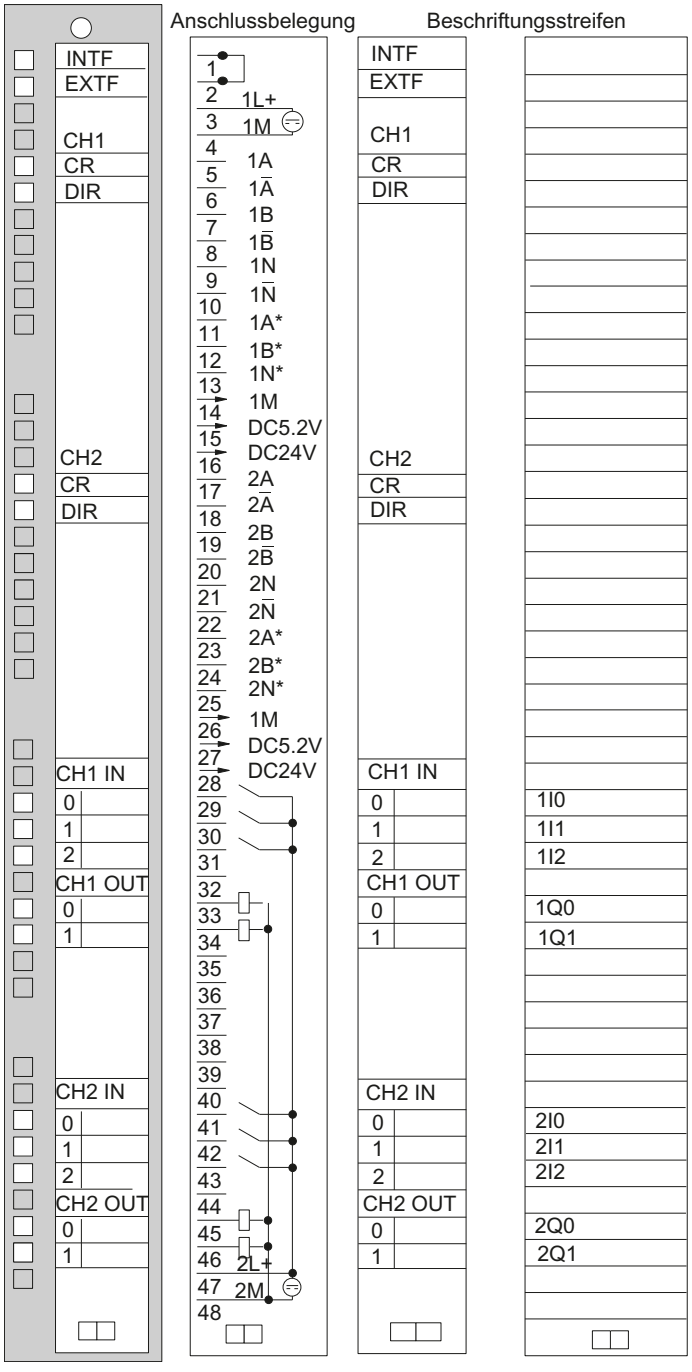
Anschlussbelegung
Beschriftungsstreifen
INTF
EXTF
CH1
CR
DIR
CH2
CR
DIR
CH1 IN
0
1
2
CH1 OUT
0
1
CH2 IN
0
1
2
CH2 OUT
0
1
1
2 1L+
3 1M
4 1A
5 1Ā
6 1B
7 1B̄
8 1N
9 1N̄
10 1A*
11 1B*
12 1N*
13 1M
14 DC5.2V
15 DC24V
16 2A
17 2Ā
18 2B
19 2B̄
20 2N
21 2N̄
22 2A*
23 2B*
24 2N*
25 1M
26 DC5.2V
27 DC24V
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46 2L+
47 2M
48
INTF
EXTF
CH1
CR
DIR
CH2
CR
DIR
CH1 IN
0
1
2
CH1 OUT
0
1
CH2 IN
0
1
2
CH2 OUT
0
1
1I0
1I1
1I2
1Q0
1Q1
2I0
2I1
2I2
2Q0
2Q1

Anschlussbelegung des Frontsteckers

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 450-1.

Anschluss	Name	Eingang / Ausgang	Funktion			
			5 V-Geber RS 422, symmetrisch	24 V-Geber asymmetrisch	24 V Impulsgeber mit Richtungspegel	24 V Initiator
1			-			
2			-			
3	1L+	Eingang	Hilfsspannung 24 V für Geberversorgung			
4	1M	Eingang	Masse Hilfsspannung für Geberversorgung			
Zähler 1						
5	1 A	Eingang	Gebersignal A	-		
6	1 /A	Eingang	Gebersignal /A	-		
7	1 B	Eingang	Gebersignal B	-		
8	1 /B	Eingang	Gebersignal /B	-		
9	1 N	Eingang	Gebersignal N	-		
10	1 /N	Eingang	Gebersignal /N	-		
11	1 A*	Eingang	-	Gebersignal A*		
12	1 B*	Eingang	-	Gebersignal B*	Richtungssignal	-
13	1 N*	Eingang	-	Gebersignal N*	-	
14	1M	Ausgang	Masse Geberversorgung			
15	DC5.2V	Ausgang	Geberversorgung 5,2 V	-		
16	DC24V	Ausgang	-	Geberversorgung 24 V		
Zähler 2						
17	2 A	Eingang	Gebersignal A	-		
18	2 /A	Eingang	Gebersignal /A	-		
19	2 B	Eingang	Gebersignal B	-		
20	2 /B	Eingang	Gebersignal /B	-		
21	2 N	Eingang	Gebersignal N	-		
22	2 /N	Eingang	Gebersignal /N	-		
23	2 A*	Eingang	-	Gebersignal A*		
24	2 B*	Eingang	-	Gebersignal B*, Richtungssignal		
25	2 N*	Eingang	-	Gebersignal N*		
26	1M	Ausgang	Masse Geberversorgung			
27	DC5.2V	Ausgang	Geberversorgung 5,2 V	-		
28	DC24V	Ausgang	-	Geberversorgung 24 V		
Zähler 1						
29	1I0	Eingang	Digitaleingang 1I0			
30	1I1	Eingang	Digitaleingang 1I1			
31	1I2	Eingang	Digitaleingang 1I2 (Zähler setzen)			
32			-			
33	1Q0	Ausgang	Digitalausgang 1Q0			

Anschluss	Name	Eingang / Ausgang	Funktion			
			5 V-Geber RS 422, symmetrisch	24 V-Geber asym-metrisch	24 V Impulsgeber mit Richtungspegel	24 V Initiator
34	1Q1	Ausgang	Digitalausgang 1Q1			
35			-			
36			-			
37			-			
38			-			
39			-			
40			-			
Zähler 2						
41	2I0	Eingang	Digitaleingang 2I0			
42	2I1	Eingang	Digitaleingang 2I1			
43	2I2	Eingang	Digitaleingang 2I2 (Zähler setzen)			
44			-			
45	2Q0	Ausgang	Digitalausgang 2Q0			
46	2Q1	Ausgang	Digitalausgang 2Q1			
47	2L+	Eingang	Lastspannung 24 V für die Digitaleingänge und -ausgänge			
48	2M	Eingang	Masse Lastspannung für die Digitaleingänge und -ausgänge			

Hinweis

Die Schaltungen für die Zählereingänge (Geberversorgung, Gebersignale) sind potenzialgebunden zur Masse der CPU ausgeführt. Verbinden Sie deshalb die Klemme 4 (1M) mit der Masse der CPU niederohmig.

Wenn Sie die Geber mit einer externen Spannung versorgen, müssen Sie die Masse dieser externen Spannung ebenfalls mit der Masse der CPU verbinden.

Weitere Informationen zu Aufbau und Verdrahtung finden Sie im Gerätehandbuch (http://support.automation.siemens.com/WW/view/de/1118412) der Zählerbaugruppe FM 450-1.

Hilfsspannung 1L+, 1M

Für die Spannungsversorgung der 5 V- und 24 V-Geber schließen Sie an den Anschlüssen 1L + und 1M eine Gleichspannung von 24 V an.

Eine integrierte Diode schützt die Baugruppe vor Verpolung der Hilfsspannung.

Die Baugruppe überwacht, ob die Hilfsspannung angeschlossen ist.

Geberversorgung DC5.2V

Die Baugruppe erzeugt aus der Hilfsspannung 1L+ / 1M eine Spannung von 5,2 V bei einem maximalen Strom von 300 mA pro Zählkanal, die am jeweiligen Anschluss "DC5.2V" für die Spannungsversorgung eines 5 V-Gebers kurzschlussfest zur Verfügung steht.

Geberversorgung DC24V

Für die 24 V-Spannungsversorgung eines Gebers wird die Spannung 1L+ / 1M am Ausgang "DC24V" kurzschlussfest zur Verfügung gestellt. Die Geberversorgung wird auf Kurzschluss kontrolliert.

5 V-Gebersignale A und /A, B und /B, N und /N

Sie können an jeden Zähler Inkrementalgeber mit 5 V-Differenzsignalen nach RS 422 anschließen, d. h. Inkrementalgeber mit den Differenzsignalen A und /A, B und /B, N und /N.

Die Signale A und /A, B und /B, N und /N werden über die entsprechend gekennzeichneten Anschlüsse angeschlossen.

Die Signale N und /N schließen Sie nur dann an, wenn Sie den Zähler auf der Nullmarke des Gebers setzen wollen.

Die Eingänge sind gegenüber dem Bus der S7-400 nicht potenzialgetrennt.

24 V-Gebersignale A*, B* und N*

24 V-Signale werden mit den Buchstaben A*, B* und N* bezeichnet.

Sie können an jeden Zähler drei verschiedene Gebertypen anschließen:

- Inkrementalgeber mit 24 V-Signalen:
  Die Signale A*, B* und N* werden über die entsprechend gekennzeichneten Anschlüsse angeschlossen.
- Impulsgeber ohne Richtungspegel:
  Das Signal wird am Anschluss A* angeschlossen.
- Impulsgeber mit Richtungspegel:
  Das Zählsignal wird an den Anschluss A* angeschlossen. Der Richtungspegel wird am Anschluss B* angeschlossen.

Die Eingänge sind gegenüber dem Bus der S7-400 nicht potenzialgetrennt.

Eingangsfilter für 24 V-Gebersignale

Um Störungen zu unterdrücken, können Sie für die 24 V-Eingänge A*, B* und N* Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren. Es stehen für jeden Zähler die folgenden beiden Eingangsfilter zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Maximale Zählfrequenz	200 kHz	20 kHz
Mindestimpulsbreite der Zählsignale	2,5 µs	25 µs

Digitaleingänge

Die Digitaleingänge 1I0 und 1I1 können Sie für die Torsteuerung des Zählers 1 benutzen.
Die Digitaleingänge 2I0 und 2I1 können Sie für die Torsteuerung des Zählers 2 benutzen.
Die Tore können sowohl pegelgesteuert als auch flankengesteuert betrieben werden.

Der Digitaleingang 1I2 dient zum Setzen des Zählers 1 auf den Ladewert.
Der Digitaleingang 2I2 dient zum Setzen des Zählers 2 auf den Ladewert.

Die Digitaleingänge werden mit einer Nennspannung von 24 V betrieben.

Die Digitaleingänge sind gegenüber dem Bus der S7-400 und den Zähleingängen potenzialgetrennt.

Eingangsfilter für Digitaleingänge

Um Störungen zu unterdrücken, können Sie für die Digitaleingänge 1I0, 1I1 und 1I2 bzw. 2I0, 2I1 und 2I2 Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren. Es stehen die folgenden beiden Eingangsfilter zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Mindestimpulsbreite der Eingangssignale	2,5 µs	25 µs

Digitalausgänge

Zum direkten Auslösen von Steuerungsvorgängen verfügt die FM 450-1 über die Digitalausgänge 1Q0 und 1Q1 (für Zähler 1) bzw. 2Q0 und 2Q1 (für Zähler 2).

Die Digitalausgänge werden über die Lastspannung 2L+ versorgt.

Die Digitalausgänge sind gegenüber dem Bus der S7-400 und den Zähleingängen potenzialgetrennt.

Die Digitalausgänge sind P-Schalter und mit einem Laststrom von 0,5 A belastbar. Sie sind gegen Überlast und Kurzschluss geschützt.

Hinweis

Der direkte Anschluss von Relais und Schützen ist ohne externe Beschaltung möglich.

Das zeitliche Verhalten der Digitalausgänge ist von der Parametrierung abhängig.

Lastspannung 2L+ / 2M

Für die Versorgung der Digitalausgänge 1Q0 und 1Q1 bzw. 2Q0 und 2Q1 müssen Sie an den Klemmen 2L+ und 2M der Baugruppe eine Lastspannung von 24 V anschließen.

Eine integrierte Diode schützt die Baugruppe vor Verpolung der Lastspannung.

Die Lastspannung 2L+ / 2M wird von der FM 450-1 nicht überwacht.

FM 450-1 konfigurieren

FM 450-1 parametrieren

Kanal spezifizieren

Kanal kopieren

Sie können die Parameter eines Kanals auf einen anderen Kanal kopieren.

1. Wählen Sie den Kanal aus, dessen Parameter kopiert werden sollen (Kanal Quelle).
2. Aktivieren Sie den Kanal, auf den die Parameter kopiert werden sollen (Kanal Ziel).
3. Drücken Sie die Schaltfläche "Kopieren".

Kanal n Betriebsarten

Betriebsart auswählen

Beschreibung

Mit der Vorgabe einer Betriebsart wählen Sie, mit welcher Funktionalität Sie die Zählerbaugruppe betreiben wollen.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne Tor)	Die FM 450-1 zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit HW- oder SW-Tor)	Die FM 450-1 zählt mit dem Öffnen des Tores ab dem Ladewert bis zur Zählgrenze.
Periodisches Zählen (Seite 187) (mit HW- oder SW-Tor)	Die FM 450-1 zählt mit dem Öffnen des Tores ab dem Ladewert zwischen den Zählgrenzen.

Die Voreinstellung ist Betriebsart "Endlos Zählen".

Torsteuerung auswählen

Beschreibung

Mit der Vorgabe einer Torsteuerung wählen Sie, wie der Zählvorgang in der gewählten Betriebsart ablaufen soll.

- Ohne Tor
  Alle Tore sind offen.
- SW-Tor (Seite 204)
  Das SW-Tor wird mit dem Eingangsparameter / Steuerbit SW_GATE geöffnet und geschlossen.
- HW-Tor (Seite 205) , pegel- oder flankengesteuert
  Das HW-Tor wird mit entsprechenden Signalen an den Digitaleingängen geöffnet und geschlossen.

Die Voreinstellung ist "Ohne Tor".

Zählbereich auswählen

Beschreibung

Auf der Baugruppe befindet sich ein 32 Bit breites Zählregister. Mit der Einstellung Zählbereich wählen Sie, ob die Baugruppe nur im positiven Bereich zählt oder ob das 32. Bit als Vorzeichen-Bit interpretiert wird und somit negative Zahlen dargestellt werden können.

Wählen Sie den gewünschten Zählbereich aus.

Zusätzliche Informationen finden Sie unter "Zählbereiche und Zählgrenzen (Seite 182)".

Überschreiten der Obergrenze, Unterschreiten der Untergrenze und Nulldurchgang

Bei den Zählgrenzen wird bei Überschreiten bzw. Unterschreiten ein Bit im DB der Anweisung CNT_CTRL gesetzt.

Im Zählbereich "-31 bis +31 Bit" wird bei einem Nulldurchgang ebenfalls ein Bit im DB gesetzt.

Im Zählbereich "0 bis +32 Bit" wird mit dem Nulldurchgang je nach Zählrichtung zusätzlich noch Überschreiten bzw. Unterschreiten angezeigt.

Ereignis	Statusbit im DB
Überschreiten der Obergrenze	STS_OFLW wird gesetzt.
Unterschreiten der Untergrenze	STS_UFLW wird gesetzt.
Nulldurchgang	STS_ZERO wird gesetzt.

Auslösen von Prozessalarmen

Sie können die Ereignisse Überschreiten der Obergrenze, Unterschreiten der Untergrenze und Nulldurchgang auch per Prozessalarm melden.

Kanal n Geber

Signalart des Gebers auswählen

Signalart auswählen

1. Wählen Sie die Signalart des eingesetzten Gebers aus.
   Abhängig von Ihrer Auswahl und der Betriebsart sind diejenigen Optionen gesperrt, die für diesen speziellen Geber nicht möglich sind. Eine gesperrte Option wird durch graue Schrift im entsprechenden Auswahlfenster gekennzeichnet.
2. Wählen Sie die Optionen aus, die Sie mit dem ausgewählten Geber nutzen wollen.

Sie können die Signalart von einem der folgenden Geber wählen:

- Geber: 5 V inkrementell (Seite 302) (5 V Differenzsignal)
- Geber: 24 V inkrementell (Seite 303) (24 V Spur A+B phasenversetzt)
- Geber: 24 V Impuls und Richtung (Seite 303)
- Geber: 24 V Initiator (Seite 304)

Geber: 5 V inkrementell

5 V-Inkrementalgeber

Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B, und N, /N einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung (Seite 210)	Einfach	Nur eine Flanke von Signal A wird ausgewertet.
	Zweifach	Beide Flanken von Signal A werden ausgewertet.
	Vierfach	Beide Flanken von Signal A und B werden ausgewertet.
Zählrichtung (Seite 208)	Normal	Zur Anpassung der Zählrichtung an den Prozess
	Invertiert	Zur Anpassung der Zählrichtung an den Prozess
Max. Zählfrequenz	–	Die maximale Zählfrequenz für diesen Geber ist 500 kHz.
Gebereingänge	–	Nicht relevant
Überwachung (Seite 208)	A+B+N	Alle Signalpaare werden auf Kurzschluss und Drahtbruch überwacht.
	A+B	Die Signalpaare A und B werden überwacht.
	A	Nur das Signalpaar A wird überwacht.
	Keine	Kein Signalpaar wird überwacht.

Geber: 24 V inkrementell

24 V-Inkrementalgeber

Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung (Seite 210)	Einfach	Nur eine Flanke von Signal A wird ausgewertet.
	Zweifach	Beide Flanken von Signal A werden ausgewertet.
	Vierfach	Beide Flanken von Signal A und B werden ausgewertet.
Zählrichtung (Seite 209)	Normal	Zur Anpassung der Zählrichtung an den Prozess
	Invertiert	Zur Anpassung der Zählrichtung an den Prozess
Max. Zählfrequenz (Seite 305)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 304)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Geber: 24 V Impuls und Richtung

24 V-Impulsgeber

Wählen Sie die Signalart "24 V Impuls und Richtung", wenn Sie einen 24 V-Impulsgeber mit dem Zählsignal A* und dem Richtungssignal B* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung	–	Nicht relevant
Zählrichtung	–	Nicht relevant
Max. Zählfrequenz (Seite 305)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 304)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Geber: 24 V Initiator

24 V-Initiator

Wählen Sie die Signalart "24 V Initiator", wenn Sie einen 24 V-Impulsgeber mit dem Zählsignal A* einsetzen.

Für diesen Geber können Sie unter folgenden Optionen auswählen:

Parameter	Option	Bedeutung
Signalauswertung	–	Nicht relevant
Zählrichtung	–	Nicht relevant
Max. Zählfrequenz (Seite 305)	200 kHz	An den Eingängen wirkt ein Filter mit 1 µs Verzögerung.
	20 kHz	An den Eingängen wirkt ein Filter mit 15 µs Verzögerung.
Gebereingänge (Seite 304)	M-Schalter	Der Geber schaltet nach 0 V (Masse).
	P-Schalter / Gegentakt	Der Geber schaltet nach 24 V.
Überwachung	–	Nicht relevant

Parametrierung der Gebereingänge

Parametrierung der Gebereingänge

Die Zählrichtung ist über die Parametrierung der Gebereingänge festgelegt. Die Tabelle zeigt die Änderung der Zählrichtung in Abhängigkeit von der Parametrierung.

Parametrierung	Anschlussklemme B*	Zählrichtung
M-Schalter	Unbeschaltet	Rückwärts
	Kurzgeschlossen nach Masse	Vorwärts
P-Schalter / Gegentakt	Unbeschaltet	Vorwärts
	24 V angeschlossen	Rückwärts

Bei der Parametrierung wählen Sie bei Geberauswahl "24 V Impuls und Richtung".

Eine Richtungsumkehr durch Invertierung des B*-Signals ist bei diesen Zählsignalen nicht möglich.

Hinweis

Bei dieser Art der Auswertung kann der Zählwert bei pendelndem Zählsignal an der Flanke "weglaufen", da alle Signale aufaddiert werden.

Eingangsfilter für 24 V-Zähleingänge

Eingangsfilter für 24 V-Zähleingänge

Um Störungen zu unterdrücken, können Sie für die 24 V-Zähleingänge A*, B* und N* und für die Digitaleingänge Eingangsfilter mit einer einheitlichen Filterzeit parametrieren. Es gibt folgende Eingangsfilter:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Maximale Zählfrequenz	200 kHz	20 kHz
Mindestimpulsbreite der Zählsignale	2,5 µs	25 µs

Kanal n Eingänge

Torsignale auswählen

Steuerung des Hardware-Tores

Für die Steuerung des Hardware-Tores können Sie auswählen:

- Pegelgesteuertes HW-Tor (mit Pegel am Eingang 1I0 bzw. 2I0)
- Flankengesteuertes HW-Tor (zum Öffnen mit Flanke am Eingang 1I0 bzw. 2I0; zum Schließen mit Flanke am Eingang 1I1 bzw. 2I1)

Weitere Informationen dazu finden Sie unter "Hardware-Tor (Seite 205)".

Mindestimpulsbreite auswählen

Mindestimpulsbreite

Um Störungen zu unterdrücken, können Sie für die 24 V-Eingänge A*, B* und N* und für die Digitaleingänge Eingangsfilter (RC-Glieder) mit einer einheitlichen Filterzeit parametrieren.

Es stehen zwei Eingangsfilter mit folgenden Daten zur Verfügung:

Merkmale	Eingangsfilter 1 (voreingestellt)	Eingangsfilter 2
Typische Eingangsverzögerung	1 µs	15 µs
Maximale Zählfrequenz	200 kHz	20 kHz
Mindestimpulsbreite der Zählsignale	2,5 µs	25 µs

Zähler setzen

Zähler setzen

Übersicht

Wenn Sie den Zählvorgang von einem bestimmten Wert, dem sogenannten Ladewert, starten wollen, müssen Sie parametrieren, mit welchem Signal der Zähler auf den Ladewert gesetzt werden soll. Sie können den Zähler folgendermaßen setzen:

- Mit dem Eingangsparameter L_DIRECT der Anweisung CNT_CTRL
- Mit einem externen Signal entweder über den Eingang 1I2 (2I2) oder über den Eingang in Verbindung mit der Nullmarke des Gebers

Ladewert

Sie können für den Ladewert eine Zahl innerhalb des Zählbereichs vorgeben.

Der Ladewert wird entsprechend dem angewählten Zählbereich interpretiert. Geben Sie zum Beispiel als Ladewert FFFF FFFF H vor, dann wird dieser Wert innerhalb des Zählbereichs "0 bis +32 Bit" als 4 294 967 295 und innerhalb des Zählbereichs "-31 bis +31 Bit" als -1 interpretiert.

Den Ladewert tragen Sie in den DB der Anweisung CNT_CTRL ein.

Für den Ladewert gelten folgende zulässige Wertebereiche:

Wertebereich für Ladewerte	
Untergrenze	maximale untere Zählgrenze
Obergrenze	maximale obere Zählgrenze

Zähler setzen mit einem externen Signal

Sie können zwei verschiedene externe Signale wählen, mit denen Sie den Zähler auf den Ladewert setzen:

- Nur Eingang 1I2 (2I2) (Seite 307)
- Eingang 1I2 (2I2) und Nullmarke (Seite 309) des Gebers

Die Nullmarke des Gebers verwenden Sie dann, wenn Sie an einer bestimmten Stelle Ihres Prozesses den Zähler auf einen bestimmten Zählerstand synchronisieren wollen. Hierdurch erreichen Sie eine größere Präzision des Zählvorgangs.

Das Setzen des Zählers ist unabhängig von der Zählbetriebsart.

Nach erfolgtem Setzen des Zählers mit einem externen Signal wird das Bit STS_SYNC im DB gesetzt.

Hinweis

Die Synchronisierung eines Zählers mit der Nullmarke ist nur bei geöffnetem Tor sinnvoll.

Haben Sie beim Setzen eines Zählers mit einem externen Signal nur eine Zählrichtung freigegeben, dann müssen Sie beachten, dass beim Schließen des Tores die aktuelle Zählrichtung gespeichert (eingefroren) wird. Dadurch ist ein Synchronisieren des Zählers entgegen der freigegebenen Zählrichtung möglich.

Zähler setzen mit Eingang 1I2 (2I2)

Zähler setzen mit Eingang 1I2 (2I2)

Der Zähler kann über eine steigende Flanke am Eingang 1I2 (2I2) (DI-Set) mit dem Ladewert geladen werden.

Sie können über die Variablen ENSET_UP und ENSET_DN im DB der Anweisung CNT_CTRL und durch die Parametrierung das Verhalten der FM 450-1 bei einer steigenden Flanke am Eingang 1I2 (2I2) festlegen:

Parameter	Verhalten der FM 450-1
ENSET_UP gesetzt	Der Zähler wird nur beim Vorwärtszählen gesetzt.
ENSET_DN gesetzt	Der Zähler wird nur beim Rückwärtszählen gesetzt.
ENSET_UP und ENSET_DN gesetzt	Der Zähler wird beim Vorwärts- und beim Rückwärtszählen gesetzt.
Parametrierung "Zähler setzen einmalig"	Der Zähler wird nur bei der ersten steigenden Flanke am Eingang 1I2 (2I2) gesetzt. Soll der Zähler erneut gesetzt werden, müssen Sie zuerst wieder ENSET_UP oder ENSET_DN setzen. Mit der nächsten steigenden Flanke am Eingang 1I2 (2I2) wird dann der Zähler neu gesetzt.
Parametrierung "Zähler setzen mehrmalig"	Der Zähler wird bei jeder steigenden Flanke am Eingang 1I2 (2I2) gesetzt, solange ENSET_UP oder/und ENSET_DN gesetzt sind.

Hinweis

Sie müssen auf jeden Fall eine der beiden Variablen ENSET_UP oder/und ENSET_DN setzen, damit der betreffende Zähler über den Eingang 1I2 (2I2) gesetzt werden kann.

Einmaliges Setzen mit dem Eingang 1I2 (2I2)

Das folgende Bild zeigt das einmalige Setzen des Zählers 1 mit dem Eingang 1I2. Für Zähler 2 gelten mit Eingang 2I2 analoge Verhältnisse. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei der ersten steigenden Flanke am Eingang 1I2 wird der Zähler 1 gesetzt, solange auch ENSET_UP gesetzt ist. Wenn Sie den Zähler neu setzen wollen, müssen Sie zuerst ENSET_UP zurücksetzen und anschließend wieder setzen. Dann führt die nächste steigende Flanke am Eingang 1I2 zum Setzen des Zählers.

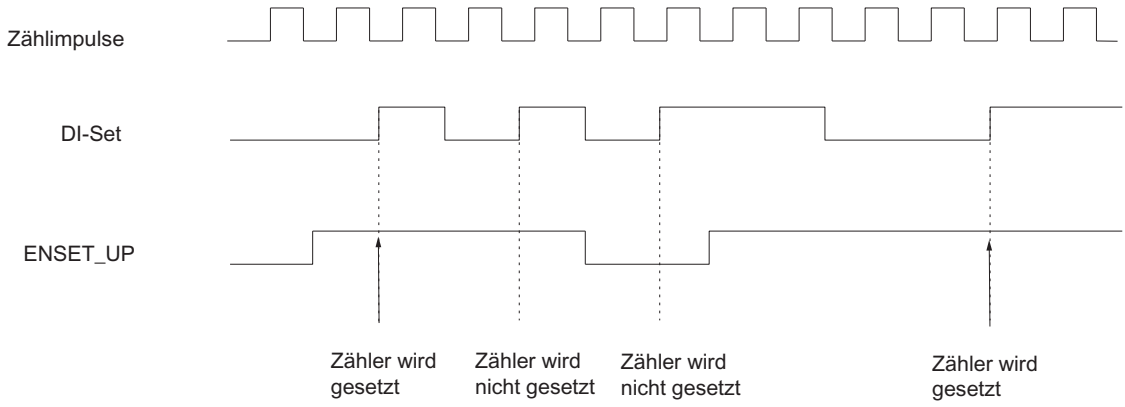

Mehrmaliges Setzen mit dem Eingang 1I2 (2I2)

Das folgende Bild zeigt das mehrmalige Setzen des Zählers 1 mit dem Eingang 1I2. Für Zähler 2 gelten mit Eingang 2I2 analoge Verhältnisse. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei jeder steigenden Flanke am Eingang 1I2 wird der Zähler 1 gesetzt, solange auch ENSET_UP gesetzt ist. Wenn Sie ENSET_UP zurücksetzen, erfolgt kein Setzen des Zählers mit einer steigenden Flanke am Eingang 1I2. Erst wenn Sie ENSET_UP wieder gesetzt haben, führt die nächste steigende Flanke am Eingang 1I2 zum Setzen des Zählers.

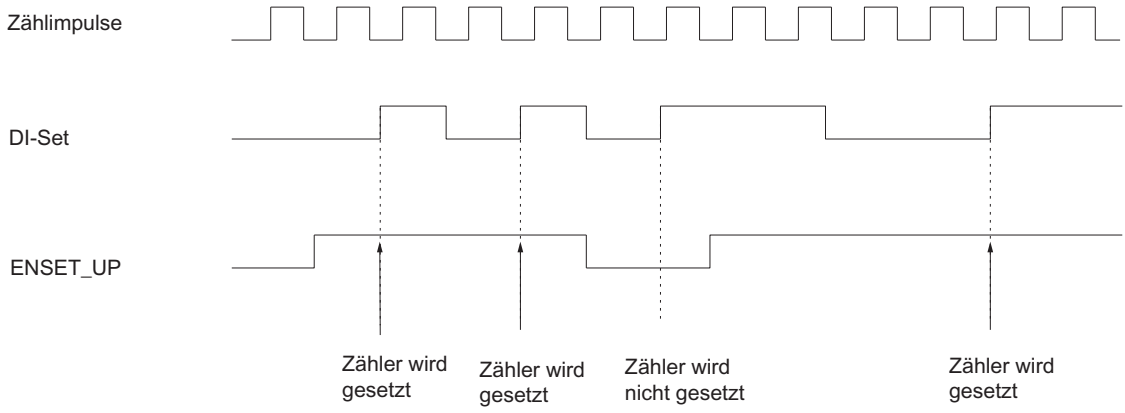

Zähler setzen mit Eingang 1I2 (2I2) und Nullmarke

Zähler setzen mit Eingang 1I2 (2I2) und Nullmarke

Wenn Sie das Setzen des Zählers mit der Nullmarke des Gebers parametrieren ("Nullmarke für das Setzen auswerten"), dann wird bei der steigenden Flanke der Nullmarke der Zähler auf den Ladewert gesetzt.

Das Setzen wird nur dann durchgeführt, wenn zum Zeitpunkt der steigenden Flanke der Nullmarke auch der Eingang 1I2 (2I2) des Zählers gesetzt ist.

Sie können über die Variablen ENSET_UP und ENSET_DN im DB der Anweisung CNT_CTRL und durch die Parametrierung das Verhalten der FM 450-1 bei steigender Flanke der Nullmarke festlegen:

Eingangsparameter	Verhalten der FM 450-1
ENSET_UP gesetzt	Der Zähler wird nur beim Vorwärtszählen gesetzt.
ENSET_DN gesetzt	Der Zähler wird nur beim Rückwärtszählen gesetzt.
ENSET_UP und ENSET_DN gesetzt	Der Zähler wird beim Vorwärts- und beim Rückwärtszählen gesetzt.
Parametrierung "Zähler setzen einmalig"	Der Zähler wird nur bei der ersten steigenden Flanke der Nullmarke gesetzt. Soll der Zähler erneut gesetzt werden, müssen Sie erst wieder ENSET_UP oder ENSET_DN setzen (Flankenauswertung). Mit der nächsten steigenden Flanke der Nullmarke wird dann der Zähler neu gesetzt.
Parametrierung "Zähler setzen mehrmalig"	Der Zähler wird bei jeder steigenden Flanke der Nullmarke gesetzt, solange ENSET_UP oder/und ENSET_DN gesetzt sind.

Hinweis

Sie müssen auf jeden Fall eine der beiden Variablen ENSET_UP oder/und ENSET_DN und den Eingang 1I2 (2I2) setzen, damit der Zähler mit der Nullmarke gesetzt werden kann.

Einmaliges Setzen mit Eingang 1I2 (2I2) und Nullmarke

Das folgende Bild zeigt das einmalige Setzen des Zählers 1 mit der Nullmarke. Für Zähler 2 gelten mit Eingang 2I2 analoge Verhältnisse. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei der ersten steigenden Flanke der Nullmarke wird der Zähler 1 gesetzt, solange auch ENSET_UP und der Eingang 1I2 gesetzt sind.

Wenn Sie den Zähler 1 neu setzen wollen, dann müssen Sie zuerst ENSET_UP zurücksetzen und anschließend wieder setzen. Ist der Eingang 1I2 nicht gesetzt, erfolgt das Setzen des Zählers mit der ersten Nullmarke nach Setzen des Eingangs 1I2. Ist der Eingang 1I2 gesetzt, erfolgt das Setzen mit der nächsten Nullmarke.

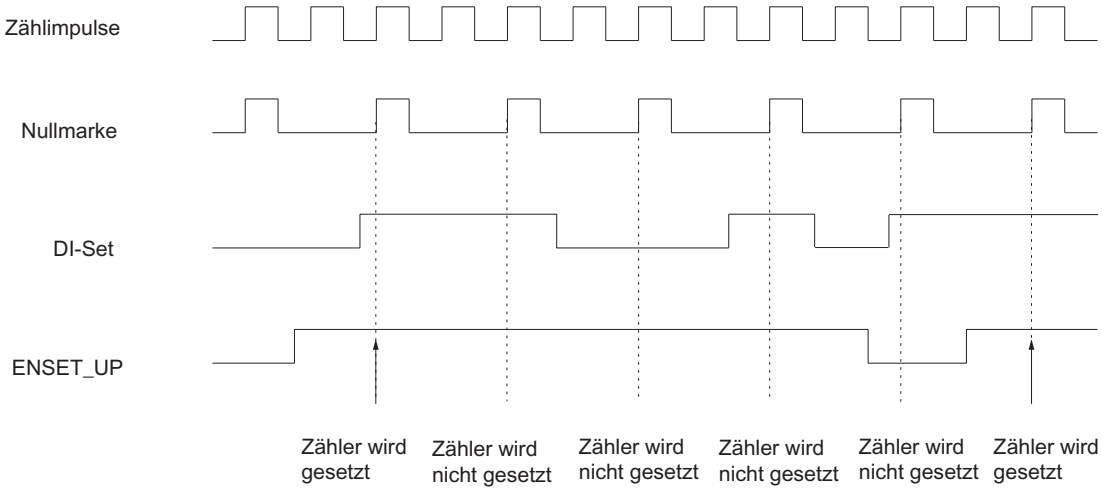

Mehrmaliges Setzen mit Eingang 1I2 (2I2) und Nullmarke

Das folgende Bild zeigt das mehrmalige Setzen des Zählers 1 mit der Nullmarke. Für Zähler 2 gelten mit Eingang 2I2 analoge Verhältnisse. Dargestellt ist der Fall, dass nur ENSET_UP gesetzt wird, d. h. der Zähler wird beim Vorwärtszählen gesetzt.

Bei jeder steigenden Flanke der Nullmarke wird der Zähler gesetzt, solange ENSET_UP und der Eingang 1I2 gesetzt sind.

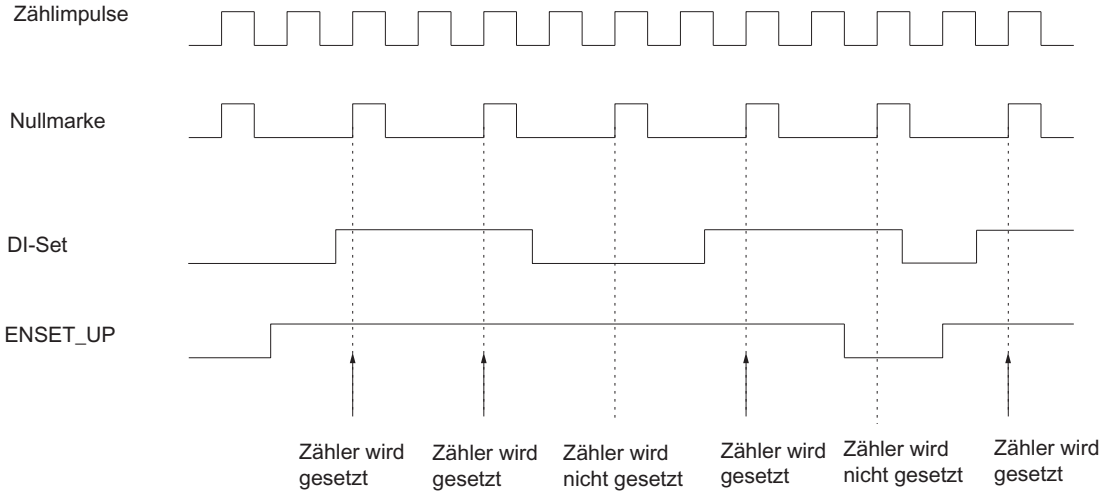

Kanal n Ausgänge

Impulsdauer festlegen

Impulsdauer

Zur Anpassung an die in Ihrem Prozess verwendeten Aktoren (Schütze, Stellglieder etc.) können Sie eine Impulsdauer vorgeben. Die Impulsdauer gibt an, wie lange die Ausgänge Q0 und Q1 bei Erreichen eines Vergleichswerts eingeschaltet sind.

Mit der Impulsdauer geben Sie vor, wie lange der Ausgang mindestens gesetzt werden soll. Diese Einstellung ist nur wirksam, wenn Sie das Verhalten des Ausgangs entsprechend vorwählen. Die Impulsdauer ist nicht wirksam, wenn der Ausgang zwischen Vergleichswert und Überlauf bzw. Unterlauf eingeschaltet werden soll.

Die Impulsdauer beginnt mit dem Setzen des Ausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 1 ms.

Sie können für die Impulsdauer einen Wert zwischen 0 und 500 ms vorgeben. Dieser Wert gilt für beide Ausgänge gemeinsam.

Bei Impulsdauer = 0 ms wird der Ausgang bei Erreichen des Vergleichswerts gesetzt und mit dem nächsten Zählimpuls rückgesetzt.

Die Voreinstellung für die Impulsdauer ist 0.

Hinweis

Wenn Sie als Impulsdauer = 0 ms vorgeben, wird der Ausgang so lange eingeschaltet, wie der Zählerstand gleich dem Vergleichswert ist (siehe oben im folgenden Bild).

Wegen der Schaltzeiten der Digitalausgänge (maximal 300 µs) können Steuerimpulse an den Ausgängen verloren gehen, wenn der zeitliche Abstand zwischen den Zählimpulsen kleiner ist als die Schaltzeiten der Digitalausgänge (siehe unten im folgenden Bild).

Stellen Sie deshalb sicher, dass der zeitliche Abstand zwischen den Zählimpulsen größer ist als die Schaltzeiten der Digitalausgänge.

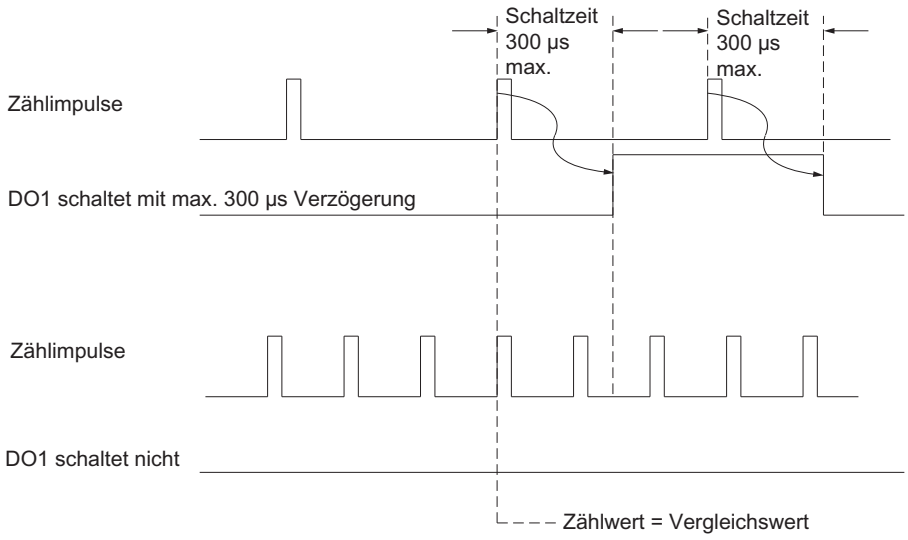

Verhalten der Ausgänge festlegen

Vergleichswerte und Ausgänge

Beschreibung

Sie können auf der FM 450-1 für die Zähler zwei Vergleichswerte (Vergleichswert 1 und 2) ablegen, die den jeweiligen beiden Digitalausgängen zugeordnet sind (Vergleichswert 1: 1Q0 bzw. 2Q0, Vergleichswert 2: 1Q1 bzw. 2Q1). Die Vergleichswerte werden mit dem aktuellen Zählerstand verglichen. Erreicht der Zählerstand einen Vergleichswert, kann der entsprechende Ausgang gesetzt werden.

Vergleichswerte 1 und 2

Die beiden Vergleichswerte tragen Sie im DB der Anweisung CNT_CTRL ein (CMP_V1, CMP_V2) und übergeben sie an die FM 450-1, indem Sie die Bits T_CMP_V1 bzw. T_CMP_V2 setzen. Der Zählvorgang wird davon nicht beeinflusst.

Die Vergleichswerte müssen innerhalb der Grenzen des jeweiligen Zählbereichs liegen. Der Vergleichswert wird entsprechend dem angewählten Zählbereich interpretiert. Wenn Sie zum Beispiel als Vergleichswert FFFF FFFF H vorgeben, dann wird dieser Wert innerhalb des Zählbereichs "0 bis +32 Bit" als 4 294 967 295 und innerhalb des Zählbereichs "-31 bis +31 Bit" als -1 interpretiert.

Für die Vergleichswerte gelten folgende zulässige Wertebereiche:

Wertebereich für Vergleichswerte	
Untergrenze	maximale untere Zählgrenze
Obergrenze	maximale obere Zählgrenze

Freigeben der Ausgänge

Bevor die Ausgänge angesteuert werden können, müssen Sie sie zuerst freigeben, indem Sie die entsprechenden Bits im DB setzen. Wenn Sie eines dieser Bits zurücksetzen, wird der zugehörige Ausgang sofort abgeschaltet, auch wenn Sie für ihn eine Impulsdauer parametriert haben.

Für das Freigeben der Ausgänge müssen folgende Bits gesetzt werden:

Ausgang	...wird freigegeben durch Freigabebit
Q0	CTRL_DQ0
Q1	CTRL_DQ1

Verhalten der Ausgänge bei den Zählbetriebsarten

Verhalten der Ausgänge festlegen

Verhalten der Ausgänge Q0 und Q1

Wählen Sie das gewünschte Verhalten des jeweiligen Ausgangs. Aktivieren Sie hierzu das entsprechende Optionsfeld:

- Kein Vergleich (Seite 313)
- Einschalten bei Zählerstand >= Vergleichswert (Seite 314)
- Einschalten bei Zählerstand <= Vergleichswert (Seite 314)
- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts (Seite 315)
- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts (Seite 315)
- Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts (Seite 314)

Hinweis

Um die Einstellungen zu aktivieren, müssen Sie im DB mit den Zählerdaten Werte für die Vergleichswerte 1 und 2 eingetragen und mit der Anweisung CNT_CTRL zur Baugruppe übertragen haben; außerdem müssen Sie die Bits CTRL_DQ0 und CTRL_DQ1 im DB setzen, um die Ausgänge freizugeben. Siehe auch "Vergleichswerte und Ausgänge (Seite 312)".

Kein Vergleich

Kein Vergleich

Der Ausgang bleibt ausgeschaltet und wird durch die Ereignisse Vergleichswert, Nulldurchgang, Überlauf oder Unterlauf nicht beeinflusst. Der Ausgang Qx kann als reiner Digitalausgang verwendet werden.

Einschalten bei Zählerstand >= Vergleichswert

Einschalten bei Zählerstand >= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler im Bereich zwischen Vergleichswert und Überlauf befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Überlauf setzt den Ausgang auf 1.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

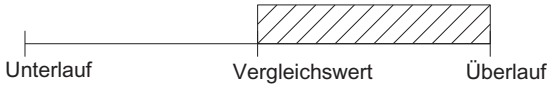

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 316).

Einschalten bei Zählerstand <= Vergleichswert

Einschalten bei Zählerstand <= Vergleichswert

Der Ausgang wird auf 1 gesetzt, wenn sich der Zähler zwischen Vergleichswert und Unterlauf befindet.

Das Setzen des Zählers auf einen Wert zwischen Vergleichswert und Unterlauf setzt den Ausgang auf 1.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist.

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 316).

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts/rückwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts/rückwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler unabhängig von der Zählrichtung den Vergleichswert erreicht.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

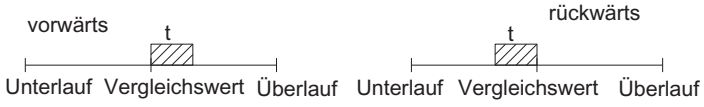

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler beim Vorwärtszählen den Vergleichswert erreicht.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

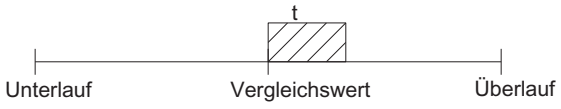

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 316).

Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts

Einschalten bei Erreichen des Vergleichswerts für Impulsdauer rückwärts

Der Ausgang wird für die Zeit der Impulsdauer auf 1 gesetzt, wenn der Zähler beim Rückwärtszählen den Vergleichswert erreicht.

Die schraffierte Fläche in der folgenden Grafik bedeutet, dass der Ausgang in diesem Bereich gesetzt ist (t = Impulsdauer).

Berücksichtigen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 316).

Randbedingungen für das Verhalten der Ausgänge

Randbedingungen für das Verhalten der Digitalausgänge

Wenn Sie das Verhalten der Digitalausgänge parametrieren, müssen Sie folgende Randbedingungen einhalten:

Wenn..	dann...
... Sie einen Ausgang "Einschalten bei Zählerstand >= Vergleichswert oder <= Vergleichswert" parametrieren wollen,	... müssen Sie sicherstellen, dass die Zeit zwischen diesen Ereignissen größer als die minimale Schaltzeit der Ausgänge ist (Schaltzeit: 300 µs); sonst gehen Steuerimpulse an den Ausgängen verloren. Erreicht der Zählerstand den Vergleichswert erneut, während der Ausgang noch eingeschaltet ist, wird kein neuer Impuls ausgelöst. Ein weiterer Impuls kann erst dann ausgelöst werden, wenn der Ausgang nicht mehr eingeschaltet ist.
... Sie einen Ausgang "Einschalten bei Erreichen des Vergleichswertes für Impulsdauer vorwärts" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswerts rückwärts" freigeben.
... Sie einen Ausgang "Einschalten bei Erreichen des Vergleichswertes für Impulsdauer rückwärts" parametrieren wollen,	... dürfen Sie keinen Prozessalarm bei "Erreichen des zugehörigen Vergleichswerts vorwärts" freigeben.

Kanal n Freigabe Prozessalarme

Auslösen eines Prozessalarms freigeben

Einleitung

Sie können für die Zähler der FM 450-1 einstellen, welche Ereignisse im Betrieb einen Prozessalarm auslösen sollen.

Was ist ein Prozessalarm?

Soll auf ein Ereignis unabhängig von dem Zyklus der CPU reagiert werden, kann jeder Zähler der FM 450-1 einen Prozessalarm auslösen. Die CPU unterbricht daraufhin das zyklische Programm und bearbeitet den Prozessalarm-OB.

Freigeben des Prozessalarms

Das Freigeben von Prozessalarmen erfolgt in zwei Schritten:

1. Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.
2. Unter "Freigabe Prozessalarme" müssen Sie dann die entsprechenden Ereignisse (Seite 317) auswählen.

Prozessalarm-OB, OB 4x

Tritt ein Prozessalarm auf, wird das Anwenderprogramm unterbrochen, die Daten von der Baugruppe in die Startinformation des OB 4x (Seite 318) übertragen und der OB 4x aufgerufen. Mit dem Verlassen des OB 4x wird der Prozessalarm quittiert.

Ist kein OB 4x programmiert, geht die CPU in STOP. Wird daraufhin wieder in RUN geschaltet, sind die Prozessalarmanforderungen gelöscht.

Verlorener Prozessalarm

Tritt ein Ereignis ein, das einen Prozessalarm auslösen soll, und das vorhergehende gleiche Ereignis wurde noch nicht quittiert, wird kein weiterer Prozessalarm ausgelöst; der Prozessalarm geht verloren. Je nach Parametrierung kann das zu dem Diagnosealarm "Prozessalarm verloren" führen.

Voreinstellung

In der Voreinstellung sind keine Prozessalarme freigegeben.

Ereignisse für das Auslösen eines Prozessalarms

Welche Ereignisse können einen Prozessalarm auslösen?

Folgende Ereignisse können beim Betrieb der FM 450-1 einen Prozessalarm auslösen (unabhängig voneinander für jeden der beiden Zähler):

- Öffnen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Schließen des Tores (bei Betriebsarten mit HW- oder SW-Tor)
- Überschreiten der Obergrenze
- Unterschreiten der Untergrenze
- Nulldurchgang
- Erreichen des Vergleichswertes 1 oder 2 in Vorwärtszählrichtung
- Erreichen des Vergleichswertes 1 oder 2 in Rückwärtszählrichtung
- Zähler setzen (DI-Set)

Aktivieren Sie die Optionskästchen für die betreffenden Ereignisse.

Sie können beliebig viele Ereignisse für die Prozessalarmauslösung auswählen. Für die Prozessalarme bei Erreichen eines Vergleichswertes müssen Sie die Randbedingungen für das Verhalten der Ausgänge (Seite 316) beachten.

Startinformation des OB 4x

Startinformation des OB 4x

In der Startinformation des OB 4x wird die temporäre Variable OB4x_POINT_ADDR beschrieben.

Die Variable OB4x_POINT_ADDR (Byte 8 bis 11) umfasst 4 Byte. In diesen 4 Byte werden die Informationen über das Ereignis eingetragen, das den Prozessalarm ausgelöst hat.

Die Tabelle zeigt, welche Bits bei welchem Alarm gesetzt sind. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung: Alarm bei ...
		Zähler 1
8	0	Öffnen des Tores
	1	Schließen des Tores
	2	Überschreiten der Obergrenze
	3	Unterschreiten der Untergrenze
	4	Erreichen des Vergleichswertes 1 in Vorwärtszählrichtung
	5	Erreichen des Vergleichswertes 1 in Rückwärtszählrichtung
	6	Erreichen des Vergleichswertes 2 in Vorwärtszählrichtung
	7	Erreichen des Vergleichswertes 2 in Rückwärtszählrichtung
9	0	Nulldurchgang
	5	Zähler setzen (DI-Set) (Synchronisation)
		Zähler 2
10	0	Öffnen des Tores
	1	Schließen des Tores
	2	Überschreiten der Obergrenze
	3	Unterschreiten der Untergrenze
	4	Erreichen des Vergleichswertes 1 in Vorwärtszählrichtung
	5	Erreichen des Vergleichswertes 1 in Rückwärtszählrichtung
	6	Erreichen des Vergleichswertes 2 in Vorwärtszählrichtung
	7	Erreichen des Vergleichswertes 2 in Rückwärtszählrichtung
11	0	Nulldurchgang
	5	Zähler setzen (DI-Set) (Synchronisation)

FM 450-1 aus dem Anwenderprogramm verwenden

Einbinden der FM 450-1 in ein Anwenderprogramm

Für das Einbinden der FM 450-1 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisung CNT_CTRL wird im Anwenderprogramm der CPU aufgerufen. Sie realisiert die Kommunikation zwischen der CPU und der FM 450-1.

Alle zu einem Kanal der Baugruppe gehörenden Daten liegen im DB der Anweisung CNT_CTRL (Zähler-DB). Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CNT_CHANTYPE1 in Ihr Projekt ein.

Außerdem gibt es für die FM 450-1 die Anweisung DIAG_INF, mit der Sie Diagnoseinformationen in den Zähler-DB übertragen können.

FM 450-1: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Diagnose-LEDs der FM 450-1

Wo wird der Fehler angezeigt?

Fehler werden durch die beiden roten Diagnose-LEDs angezeigt:

- Die Diagnose-LED INTF zeigt interne Fehler auf der Baugruppe an.
- Die Diagnose-LED EXTF zeigt externe Fehler an den Leitungsanschlüssen an.

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Diagnose-LEDs angezeigt werden.

Fehlerart / LED	Fehlerursache	Behebung
Interne Fehler Diagnose-LED INTF	Fehler beim EPROM-Test	Baugruppentausch
	Fehler beim RAM-Test	Baugruppentausch
	Zeitüberwachung (Watchdog) hat angesprochen	Baugruppentausch
	Verlorener Prozessalarm	Wird durch Bearbeitung des Prozessalarms quittiert.
Externe Fehler Diagnose-LED EXTF	Hilfsspannung 1L+ / 1M nicht angeschlossen oder Geberversorgung DC24V kurzgeschlossen	Anschluss korrigieren
	Geberversorgung DC5.2V kurzgeschlossen oder überlastet	Anschluss korrigieren
	5 V-Gebersignale fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)	Anschluss korrigieren

Auslösen eines Diagnosealarms

Alle Fehler, bis auf den Fehler beim EPROM-Test, können einen Diagnosealarm auslösen, falls Sie bei der Parametrierung den Diagnosealarm in den Grundparametern freigegeben haben. Aus den Diagnosedatensätzen DS0 und DS1 (Seite 321) erfahren Sie, welcher Fehler das Leuchten der LED verursacht hat.

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, können Sie einen Diagnosealarm parametrieren, der das zyklische Programm der CPU unterbricht und den Diagnosealarm-OB, OB 82, aufruft.

Welche Ereignisse können einen Diagnosealarm auslösen?

Folgende Ereignisse können einen Diagnosealarm auslösen:

- Externe Hilfsspannung 1L+/1M ist fehlerhaft
- Geberversorgung DC5,2V kurzgeschlossen oder überlastet
- Parametrierung der Baugruppe fehlt
- Parametrierung der Baugruppe ist fehlerhaft
- Zeitüberwachung (Watchdog) angesprochen
- RAM defekt
- Prozessalarm verloren
- Signal A fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)
- Signal B fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)
- Signal N fehlerhaft (Drahtbruch, Kurzschluss, fehlende Leitung)

Freigeben des Diagnosealarms

Bei der Konfiguration der Hardware geben Sie in den Grundparametern die Alarme für die Baugruppe frei und wählen, ob die Baugruppe Diagnose- und/oder Prozessalarme auslösen soll.

Reaktionen auf einen Diagnosealarm

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht Folgendes:

- Die Diagnoseinformationen werden in die Diagnosedatensätze DS0 und DS1 (Seite 321) auf der Baugruppe abgelegt.
- Die Sammelfehler-LED leuchtet.
  Wenn Sie den Fehler behoben haben, erlischt die Sammelfehler-LED.
- Der Diagnosealarm-OB wird aufgerufen (OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Der Zählvorgang läuft unverändert weiter, wenn ein OB 82 programmiert ist. Ohne OB 82 geht die CPU in STOP.

Voreinstellung

In der Voreinstellung sind keine Diagnosealarme freigegeben.

Diagnosedatensatz DS0 und DS1

Diagnosedatensatz DS0 und DS1

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Byte, der DS1 umfasst 16 Byte, wobei die ersten vier Byte identisch mit dem DS0 sind.

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 (Seite 322) wird automatisch beim Aufruf des Diagnose-OB in die Startinformation übertragen. Dort werden diese vier Byte ins Lokaldatum (Byte 8 bis 11) des OB 82 abgelegt.

Den Diagnosedatensatz DS1 (Seite 323) (und damit auch den Inhalt des DS0) können Sie mit der Anweisung DIAG_INF von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Belegung des Diagnosedatensatzes DS0

Belegung des Diagnosedatensatzes DS0 und der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Fehler beim RAM-Test • Zeitüberwachung (Watchdog) angesprochen • Verlorener Prozessalarm	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Hilfsspannung 1L+/1M nicht angeschlossen oder Geberversorgung DC5,2V kurzgeschlossen • Geberversorgung DC5,2V kurzgeschlossen oder überlastet • 5 V-Signale fehlerhaft • Parametrierung fehlerhaft	8:x:02
	3	Fehler in einem Kanal	Ist bei internen und externen Fehlern immer mit 1 belegt.	8:x:03
	4	Externe Hilfsspannung fehlerhaft	Spannung überprüfen	8:x.04
	7	Parametrierung fehlerhaft	Weitere Aufschlüsselung siehe Datenfehler (Seite 323)	8:x:07
1	0...3	Typklasse	Ist immer mit 8 belegt	
	4	Kanalinformation	Ist immer mit 1 belegt	
2	3	Zeitüberwachung (Watchdog) angesprochen	Baugruppe defekt oder starke Störeinflüsse	8:x:33
3	3	RAM defekt	Baugruppe defekt oder starke Störeinflüsse	8:x:43
	6	Prozessalarm verloren	Projektierung überprüfen. Prozessalarmereignis wurde erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht von der CPU quittiert wurde.	8:x:46

Belegung des Diagnosedatensatzes DS1

Belegung des Diagnosedatensatzes DS1

Der Diagnosedatensatz DS1 besteht aus 12 Byte. Die ersten 4 Byte sind identisch mit dem Diagnosedatensatz DS0. Die folgende Tabelle zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null. Dieser Datensatz wird von der Anweisung DIAG_INF in den DB der Anweisung CNT_CTRL ab DW54 eingetragen.

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
4	0...6	Kanaltyp	Ist immer mit 76H belegt	
	7	Weitere Kanaltypen	Ist immer mit 0 belegt	
5	0...7	Länge der Diagnoseinformation	Ist immer mit CH belegt	
6	0...7	Kanalanzahl	Ist immer mit 2 belegt	
7	0…1	Kanalfehlervektor	Bit 0= 1: Fehler auf Kanal 1 Bit 1= 1: Fehler auf Kanal 2	
8	0	Signal A fehlerhaft Kanal 1		8:x:B0
	1	Signal B fehlerhaft Kanal 1		8:x:B1
	2	Signal N fehlerhaft Kanal 1		8:x:B2
	4	Geberversorgung 5,2 V fehlerhaft Kanal 1		8:x:B4
	5...7	Reserviert		
9		Reserviert		
10	0	Signal A fehlerhaft Kanal 2		8:x:B0
	1	Signal B fehlerhaft Kanal 2		8:x:B1
	2	Signal N fehlerhaft Kanal 2		8:x:B2
	4	Geberversorgung 5,2 V fehlerhaft Kanal 2		8:x:B4
	5...7	Reserviert		
11		Reserviert		

Datenfehler

Wann treten Datenfehler auf?

Werden an die Baugruppe neue Parameter übertragen, dann prüft die FM 450-1 diese Parameter. Treten bei dieser Prüfung Fehler auf, meldet die Baugruppe diese Datenfehler.

Wo werden Datenfehler angezeigt?

Die Datenfehler werden von der Anweisung CNT_CTRL mit Fehlernummer in den DB der CNT_CTRL eingetragen. Über die Variable DA_ERR_W können Sie im Anwenderprogramm auf dieses Datenwort zugreifen.

Welche Datenfehler gibt es?

In der folgenden Tabelle sind die Datenfehler mit Fehlernummer und Bedeutung aufgelistet.

Nr.	Bedeutung
0	Kein Fehler
201	Wert für Empfängerauswahl falsch
202	Wert für Diagnose des Signalpaars falsch
203	Wert für Auswertung der Signale falsch
204	Wert für Eingangsfilter der 24 V-Zählsignale falsch
205	Wert für Eingangsfilter der Digitaleingänge falsch
206	Richtungsumkehr nicht erlaubt
207	Verhalten von 1Q0 (2Q0) falsch parametriert
208	Verhalten von 1Q1 (2Q1) falsch parametriert
209	Bereichsüberschreitung bei Impulsdauer
211	Falsche Betriebsart angewählt
212	Kein Tor oder beide Tore angegeben
215	Bei der Parametrierung des Prozessalarms "Erreichen des Vergleichswertes vorwärts bzw. rückwärts" wurde eine unterschiedliche Zählrichtung zur Parametrierung des Verhaltens der Ausgänge "Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts bzw. rückwärts" gewählt. Die Richtungen müssen übereinstimmen.
216	Alarme von der Torsteuerung nur bei Betriebsarten mit Torsteuerung möglich
217	Beim Verhalten der Ausgänge "Einschalten bei Zählerstand >= Vergleichswert" oder "Einschalten bei Zählerstand <= Vergleichswert" ist ein Alarm bei "Erreichen der Vergleichswerte" nicht erlaubt.

Wie werden Datenfehler quittiert?

Korrigieren Sie die Werte für die Parameter entsprechend den Vorgaben. Übertragen Sie den korrigierten Parametersatz wieder an die FM 450-1. Diese prüft erneut die Parameter und löscht den Datenfehler im DB.

Bedienfehler

Wann treten Bedienfehler auf?

Bedienfehler treten auf, wenn Sie die Baugruppe durch inkorrekte Vorgabe von Steuersignalen falsch bedienen.

Wo werden die Bedienfehler angezeigt?

Die Bedienfehlernummern werden von der Anweisung CNT_CTRL in den DB eingetragen. Mit dem Setzen des Ausgangsparameters OT_ERR zeigt die Anweisung CNT_CTRL, dass ein Bedienfehler aufgetreten ist. Über die Variable OT_ERR_B können Sie in Ihrem Programm auf dieses Datenwort zugreifen.

Welche Bedienfehler gibt es?

In der folgenden Tabelle sind die Bedienfehler mit Fehlernummer und Bedeutung aufgelistet.

Nr.	Bedeutung
0	Kein Fehler
1	Betriebsart kann mit SW-Tor nicht gestartet werden
2	Betriebsart kann nicht abgebrochen werden
4	Nur im STOP der CPU erlaubt

Wie werden Bedienfehler quittiert?

Quittieren Sie den Fehler mit dem Eingangsparameter OT_ERR_A der Anweisung CNT_CTRL.

1.2.1.5 ET 200S 1Count24V einsetzen

Anschlussbelegung des 1Count24V

Verdrahtungsregeln

Die Leitungen (Klemmen 1 und 5 sowie Klemmen 2 und 8) müssen geschirmt sein. Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung des 1Count24V

Die folgende Tabelle zeigt die Anschlussbelegung für das 1Count24V.

Ansicht	Anschlussbelegung	Bemerkungen
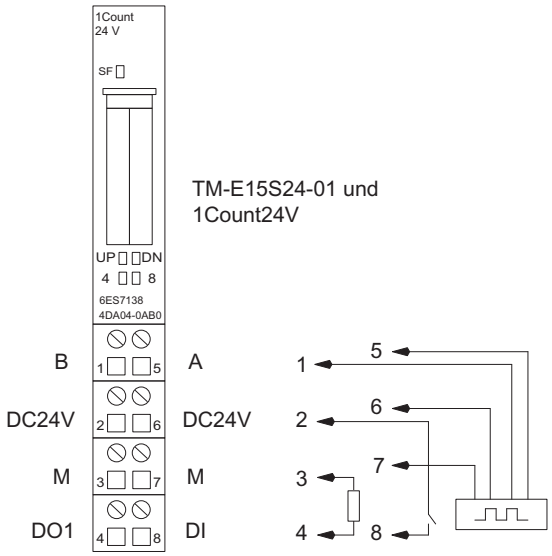		B: Richtungseingang oder Spur B A: Impulseingang oder Spur A DC24V: Geberversorgung M: Masse DI: Digitaleingang DO1: Digitalausgang

Anschluss des Impulsgebers

Die folgende Tabelle zeigt die Anschlüsse eines Impulsgebers am Terminalmodul und die möglichen Zählrichtungen.

Geberart	Anschluss	Zählrichtung
Impulsgeber ohne Richtungspegel	Zählimpulse 24 V an Klemme 5 (A)	vorwärts
Impulsgeber mit Richtungspegel	Zählimpulse 24 V an Klemme 5 (A) und Richtung 24 V an Klemme 1 (B)	vorwärts, rückwärts
Impulsgeber mit 2 um 90° phasenversetzten Spuren	Spur A Klemme 5 (A) und Spur B Klemme 1 (B)	vorwärts, rückwärts

1Count24V konfigurieren und parametrieren

Betrieb

Betriebsmodus auswählen

Betriebsmodus

Mit der Vorgabe eines Betriebsmodus wählen Sie, mit welcher Funktionalität Sie das 1Count24V betreiben wollen.

Betriebsmodus	Beschreibung
Zählen (Seite 327)	Vorauswahl für die Zählbetriebsarten
Messen (Seite 351)	Vorauswahl für die Messbetriebsarten
Wegerfassen (Seite 370)	Das 1Count24V zählt in diesem Betriebsmodus ab dem Ladewert endlos: • Erreicht das 1Count24V beim Vorwärtszählen die obere Zählgrenze und kommt ein weiterer Zählimpuls, springt es auf die untere Zählgrenze und zählt von dort ohne Impulsverlust weiter. • Erreicht das 1Count24V beim Rückwärtszählen die untere Zählgrenze und kommt ein weiterer Zählimpuls, springt es auf die obere Zählgrenze und zählt ohne Impulsverlust weiter.
Fast Mode (Seite 382) (ab Firmware-Version V2.0 unterstützt)	Das 1Count24V zählt in diesem Betriebsmodus ab dem Startwert endlos: • Erreicht das 1Count24V beim Vorwärtszählen den mit 25 Bit maximal darstellbaren Wert (alle Bits des Zählers gesetzt) und kommt ein weiterer Zählimpuls, springt der Zählwert auf "0" und zählt von dort ohne Impulsverlust weiter. • Erreicht das 1Count24V beim Rückwärtszählen den Wert "0" und kommt ein weiterer Zählimpuls, springt der Zählwert auf den mit 25 Bit maximal darstellbaren Wert (alle Bits des Zählers gesetzt) und zählt ohne Impulsverlust weiter. Dieser Betriebsmodus eignet sich zum Wegerfassen in besonders kurzen taktsynchronen Zyklen.

Parameter für Zählen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count24V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count24V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO 1 abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.
DO 1 Ersatzwert schalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der parametrierte Ersatzwert für den Digitalausgang wird durchgeschaltet. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen. Ist für den Ausgang "Impuls bei Erreichen des Vergleichswertes" parametriert, steht nur für die Impulsdauer der Ersatzwert = 1 an.	Die geänderten Parameter werden übernommen und wirksam.
DO 1 letzten Wert halten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Zustand des Digitalausgangs wird beibehalten. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.

[1] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count24V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count24V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 347) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count24V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung

Zähl- und Richtungsauswertung auswählen

Signalauswertung A*, B*

Die Signalauswertung über A*, B* ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Impuls und Richtung (Seite 330)
  Bei 24 V-Impulsgebern mit Richtungspegel muss gewährleistet sein, dass zwischen Richtungssignal B* und Zählsignal A* eine Zeitspanne von mindestens 5 µs / 50 µs liegt, je nach parametriertem Eingangsfilter.
  Das folgende Bild zeigt die Zeitspanne zwischen Richtungssignal und Zählsignal.

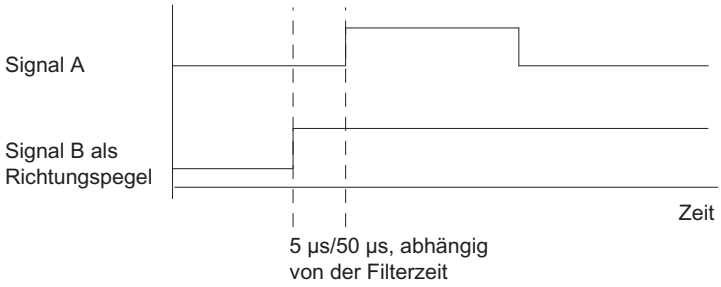

- Drehgeber (Seite 210)
  Wenn Sie einen 24 V-Drehgeber mit zwei um 90 Grad phasenversetzten Spuren am Zähl- und Richtungseingang anschließen, können Sie in allen Zählbetriebsarten eine Einfachauswertung (Seite 211) parametrieren. Alternativ können Sie Zweifach- (Seite 211) oder Vierfachauswertung (Seite 211) parametrieren.
  Bei allen Auswertungsarten können Sie die Zählrichtung (Seite 331) am Richtungseingang B* invertieren.
  An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie mit dem 1Count24V beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Hinweis

Die Angabe 100 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A* bzw. B* bezogen. Bei Zweifachauswertung ergibt sich somit maximal 200 kHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 400 kHz.

Auswertung "Impuls und Richtung"

Impuls und Richtung

Bei dieser Auswertungsart wird als Richtungsvorgabe der Pegel am Richtungseingang B* benutzt.

Ein unbeschalteter Eingang entspricht der Zählrichtung "vorwärts", wenn Sie beim Parameter Signalauswertung "Impuls und Richtung" gewählt haben.

Das folgende Bild zeigt die Signale eines 24 V-Impulsgebers mit Richtungspegel.

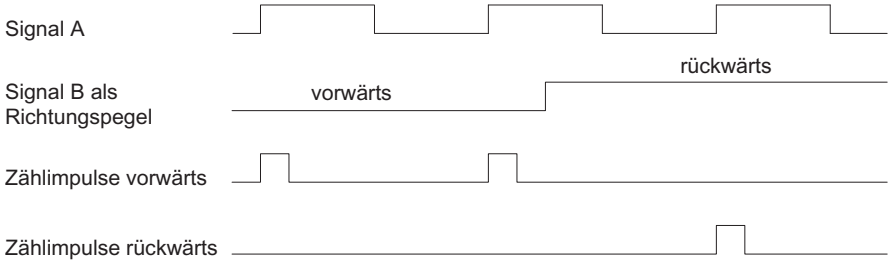

Filter für Eingangssignale auswählen

Eingangsfilter

Zur Filterung der Signale am Zähleingang A*, Richtungseingang B* und Digitaleingang DI können Sie entsprechend der Mindestimpulsdauer bzw. der maximalen Signalfrequenz einen Filter einschalten.

Sie können für jeden Eingang jeweils eine Mindestimpulsbreite von 2,5 µs bzw. 25 µs wählen.

Gebereingänge auswählen

Gebereingänge

An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Verhalten der Ausgänge bei den Zählbetriebsarten

Vergleichswerte und Ausgänge

Einleitung

Sie können auf dem 1Count24V zwei Vergleichswerte ablegen, die den Digitalausgängen zugeordnet sind. Abhängig von Zählerstand und Vergleichswerten können die Ausgänge aktiviert werden.

Digitalausgänge

Das 1Count24V hat einen "echten" Digitalausgang und einen "virtuellen" Digitalausgang, der nur als Statusbit in der Rückmeldeschnittstelle existiert.

Beide Ausgänge sind parametrierbar (Parameter "Funktion DO1", "Funktion DO2").

Sie können die Funktion und das Verhalten der Digitalausgänge im Betrieb ändern. Die neue Funktion wird sofort wirksam.

Steuern der Ausgänge gleichzeitig zu den Vergleichern

Haben Sie für die Ausgänge eine Vergleichsfunktion gewählt, können Sie die Ausgänge mit SET_DO1 oder SET_DO2 weiterhin steuern. Damit haben Sie die Möglichkeit, die Wirkung der Vergleichsfunktionen über Ihr Steuerungsprogramm zu simulieren:

- Mit der positiven Flanke von SET_DO1 oder SET_DO2 wird der Ausgang gesetzt.
  Ist die Funktion "Impuls bei Erreichen des Vergleichswertes" gewählt, wird aber nur ein Impuls der vorgegebenen Dauer ausgegeben. Bei Impulsdauer = 0 haben SET_DO1 und SET_DO2 keine Wirkung.
  Beim Ausgangsverhalten "Schalten an Vergleichswerten" ist das Steuerbit SET_DO1 nicht erlaubt.
- Eine negative Flanke von SET_DO1 oder SET_DO2 setzt den Ausgang wieder zurück.

Beachten Sie, dass die Vergleicher weiterhin aktiv sind und den Ausgang bei Änderung des Vergleichsergebnisses setzen bzw. rücksetzen können.

Hinweis

Ein durch SET_DO1 oder SET_DO2 gesetzter Ausgang wird durch den Vergleicher nicht zurückgesetzt.

Vergleichswerte laden

Die Vergleichswerte übergeben Sie an das 1Count24V. Der Zählvorgang wird davon nicht beeinflusst.

Gültiger Wertebereich für die zwei Vergleichswerte

Hauptzählrichtung: keine	Hauptzählrichtung: vorwärts	Hauptzählrichtung: rückwärts
Untere Zählgrenze bis Obere Zählgrenze	–2147483648 bis Obere Zählgrenze –1	1 bis 2147483647

Funktion und Verhalten der Digitalausgänge ändern

Über die Steuerschnittstelle (Seite 347) können Sie Funktionen und Verhalten der Ausgänge im Betrieb ändern. Das 1Count24V löscht dabei die Ausgänge und übernimmt die Werte folgendermaßen:

- **Funktion der Digitalausgänge DO1 und DO2:** Wenn Sie die Funktion so ändern, dass die Vergleichsbedingung erfüllt ist, wird der Ausgang erst nach dem nächsten Zählimpuls gesetzt. Ist jedoch die Hysterese (Seite 337) aktiv, führt das 1Count24V keine Änderung am Ausgang durch.
- **Hysterese:** Eine aktive Hysterese (Seite 337) bleibt nach der Änderung aktiv. Der neue Hysteresebereich wird beim nächsten Erreichen des Vergleichswertes übernommen.
- **Impulsdauer:** Die neue Impulsdauer (Seite 340) wird mit dem nächsten Impuls wirksam.

Verhalten der Ausgänge auswählen

Verhalten der Ausgänge DO1 und DO2

Wählen Sie in der Klappliste das gewünschte Verhalten des jeweiligen Ausgangs:

- Ausgang (Seite 334)
- Einschalten bei Zählerstand ≥ Vergleichswert (Seite 334)
- Einschalten bei Zählerstand ≤ Vergleichswert (Seite 334)
- Impuls bei Erreichen des Vergleichswertes (Seite 334)
- Schalten an Vergleichswerten (Seite 335) (nur DO1)

Funktion und Verhalten des Digitalausgangs DO1 einstellen oder ändern

Wenn Sie das Verhalten des DO1 einstellen oder ändern, müssen Sie alle parametrierten Abhängigkeiten berücksichtigen, sonst erzeugen Sie einen Parametrierfehler oder einen Fehler beim Laden.

Randbedingungen:

Wenn Sie für DO1 "Schalten an Vergleichswerten" parametrieren, dann müssen Sie

- die Hysterese = 0 setzen und
- zusätzlich die "Funktion DO2" als "Ausgang" parametrieren.

Ausgang

Ausgang

Mit den Steuerbits (Seite 347) SET_DO1 und SET_DO2 schalten Sie die Ausgänge ein und aus.

Dazu müssen die Steuerbits (Seite 347) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Mit den Statusbits STS_DO1 und STS_DO2 in der Rückmeldeschnittstelle (Seite 349) können Sie den Status der Ausgänge abfragen.

Die Statusbits (Seite 349) STS_CMP1 und STS_CMP2 zeigen, dass der jeweilige Ausgang eingeschaltet ist oder eingeschaltet war. Diese Statusbits behalten so lange ihren Zustand, bis sie quittiert werden. Wenn der Ausgang noch geschaltet ist, wird das entsprechende Bit sofort wieder gesetzt. Diese Statusbits werden auch gesetzt, wenn bei nicht freigegebenem DO1 oder DO2 die Steuerbits SET_DO1 oder SET_DO2 betätigt werden.

Taktsynchroner Betrieb

Im taktsynchronen Betrieb schaltet der Ausgang DO1 zum Zeitpunkt T_o. Der Zustand des virtuellen Ausgangs DO2 wird zum Zeitpunkt T_i gemeldet.

Einschalten bei Zählerstand ≤ Vergleichswert bzw. Zählerstand ≥ Vergleichswert

Zählerstand ≤ Vergleichswert und Zählerstand ≥ Vergleichswert

Wenn die Vergleichsbedingungen erfüllt sind, schaltet der jeweilige Vergleicher den Ausgang ein. Der Zustand des Ausgangs (Seite 349) wird mit STS_DO1 und STS_DO2 angezeigt.

Dazu müssen die Steuerbits (Seite 347) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Das Vergleichsergebnis wird mit den Statusbits (Seite 349) STS_CMP1 bzw. STS_CMP2 angezeigt. Quittieren und somit rücksetzen können Sie diese Bits erst, wenn die Vergleichsbedingungen nicht mehr erfüllt sind.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schaltet der Ausgang DO1 unmittelbar bei Erfüllung der Vergleichsbedingung und ist daher unabhängig vom Buszyklus. Der Zustand des virtuellen Ausgangs DO2 wird zum Zeitpunkt T_i gemeldet.

Impuls bei Erreichen des Vergleichswertes

Impuls bei Erreichen des Vergleichswertes

Erreicht der Zählerstand den Vergleichswert, dann schaltet der Vergleicher den jeweiligen Digitalausgang für die parametrierte Impulsdauer ein.

Dazu muss das Steuerbit (Seite 347) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Die Statusbits (Seite 349) STS_DO1 und STS_DO2 haben immer den Zustand des jeweiligen Digitalausgangs.

Das Vergleichsergebnis wird mit dem Statusbit (Seite 349) STS_CMP1 bzw. STS_CMP2 angezeigt und ist durch Quittieren erst rücksetzbar, wenn die Impulsdauer abgelaufen ist.

Ist eine Hauptzählrichtung parametriert, dann schaltet der Vergleicher nur beim Erreichen des Vergleichswerts in Hauptzählrichtung.

Ist keine Hauptzählrichtung parametriert, dann schaltet der Vergleicher beim Erreichen des Vergleichswerts aus beiden Richtungen.

War der Digitalausgang durch das Steuerbit (Seite 347) SET_DO1 bzw. SET_DO2 gesetzt, wird er nach Ablauf der Impulsdauer zurückgesetzt.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schaltet der Ausgang DO1 unmittelbar bei Erfüllung der Vergleichsbedingung und ist daher unabhängig vom Buszyklus. Der Zustand des virtuellen Ausgangs DO2 wird zum Zeitpunkt T_i gemeldet.

Schalten an Vergleichswerten

Schalten an Vergleichswerten

Der Vergleicher schaltet den Ausgang, wenn die folgenden Bedingungen erfüllt sind:

- Die beiden Vergleichswerte müssen mit der Ladefunktion (Seite 347) CMP_VAL1 und CMP_VAL2 geladen sein und
- nach dem Laden der Vergleichswerte müssen Sie den Ausgang DO1 (Seite 347) mit CRTL_DO1 freigeben.

Die folgende Tabelle zeigt, wann der DO1 eingeschaltet bzw. ausgeschaltet ist:

	DO1 ist eingeschaltet bei	DO1 ist ausgeschaltet bei
V2 < V1 (siehe Bild unten)	V2 ≤ Zählerstand ≤ V1	V2 > Zählerstand oder Zählerstand > V1
V2 = V1	V2 = Zählerstand = V1	V2 ≠ Zählerstand ≠ V1
V2 > V1 (siehe Bild unten)	V1 > Zählerstand oder Zählerstand > V2	V1 ≤ Zählerstand ≤ V2

Die folgenden Bilder zeigen den Zustand des Ausgangs DO1 in Abhängigkeit vom Zählerstand und den Vergleichswerten:

- Beim Start des Zählvorgangs ist V2 < V1

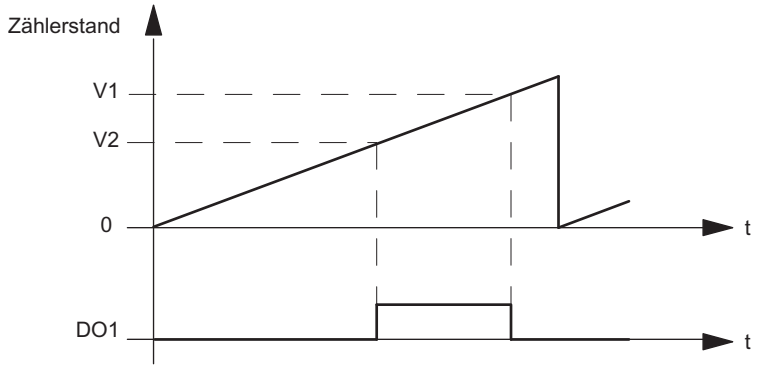

- Beim Start des Zählvorgangs ist V2 > V1

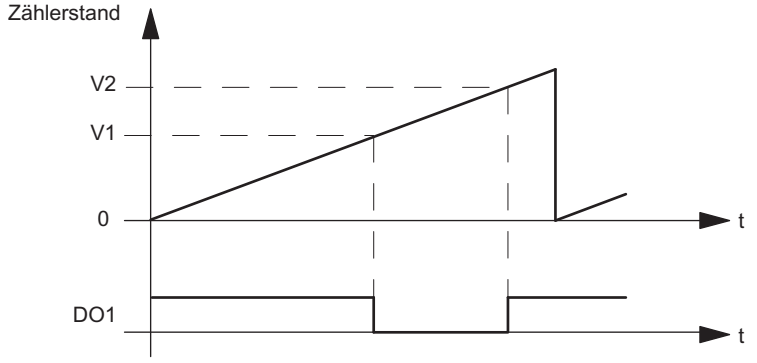

Das Vergleichsergebnis wird mit dem Statusbit (Seite 349) STS_CMP1 angezeigt. Quittieren und somit rücksetzen können Sie dieses Bit erst, wenn die Vergleichsbedingung nicht mehr erfüllt ist.

Bei diesem Ausgangsverhalten gibt es keine Hysterese.

Ein Steuern des Ausgangs DO1 durch das Steuerbit (Seite 347) SET_DO1 ist bei diesem Ausgangsverhalten nicht möglich.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schaltet der Ausgang DO1 unmittelbar bei Erfüllung der Vergleichsbedingung und ist daher unabhängig vom Buszyklus.

Ersatzwert DO1 auswählen

Ersatzwert DO1

Wenn Sie unter "Verhalten bei CPU/Master-STOP (Seite 328)" den Parameter "DO1 Ersatzwert schalten" ausgewählt haben, dann wählen Sie hier den Ersatzwert aus, der am Digitalausgang DO1 ausgegeben werden soll.

Diagnose DO1 aktivieren

Diagnose DO1

Die "Diagnose DO1" (Leitungsbruch, Kurzschluss) ist nur bei Impulslängen > 90 ms am Digitalausgang DO1 möglich.

Aktivieren Sie das Optionskästchen für die Diagnose am Ausgang DO1.

Hysterese festlegen

Hysterese

Ein Geber kann an einer bestimmten Position stehen bleiben und dann um diese Position "pendeln". Dieser Zustand führt dazu, dass der Zählerstand um einen bestimmten Wert herum schwankt. Liegt nun in diesem Schwankungsbereich zum Beispiel ein Vergleichswert, würde der zugehörige Ausgang im Rhythmus dieser Schwankungen ein- und ausgeschaltet werden. Um dieses Schalten bei kleinen Schwankungen zu verhindern, ist das 1Count24V mit einer parametrierbaren Hysterese ausgestattet. Sie können einen Bereich zwischen 0 und 255 (0 bedeutet: Hysterese abgeschaltet) parametrieren.

Die Hysterese wirkt auch bei Überlauf und Unterlauf.

Wirkungsweise bei Zählerstand ≤ Vergleichswert und Zählerstand ≥ Vergleichswert

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Hauptzählrichtung vorwärts" und Ausgang "Einschalten bei Zählerstand ≥ Vergleichswert" parametriert.

Mit dem Erreichen der Vergleichsbedingung wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv. Der Vergleicher schaltet wieder entsprechend seiner Vergleichsbedingungen.

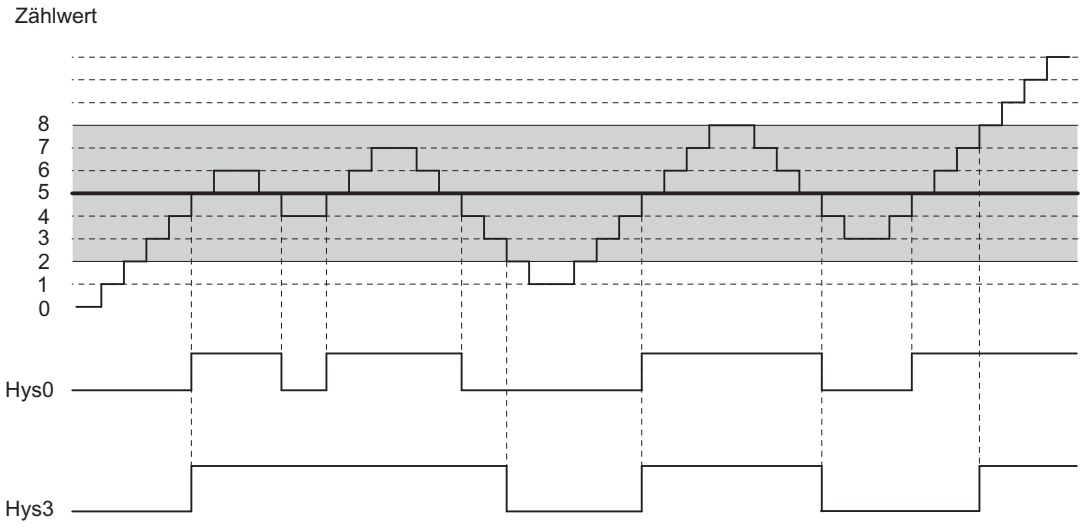

Hinweis

Bei einem Wechsel der Zählrichtung auf dem Vergleichswert bei aktiver Hysterese wird der Ausgang zurückgesetzt.

Wirkungsweise bei Vergleichswert erreicht und Impulsdauer = 0

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Impuls bei Erreichen des Vergleichswertes", "keine Hauptzählrichtung" und "Impulsdauer = 0" parametriert.

Mit dem Erreichen der Vergleichsbedingungen wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv. Der Vergleicher löscht das Vergleichsergebnis.

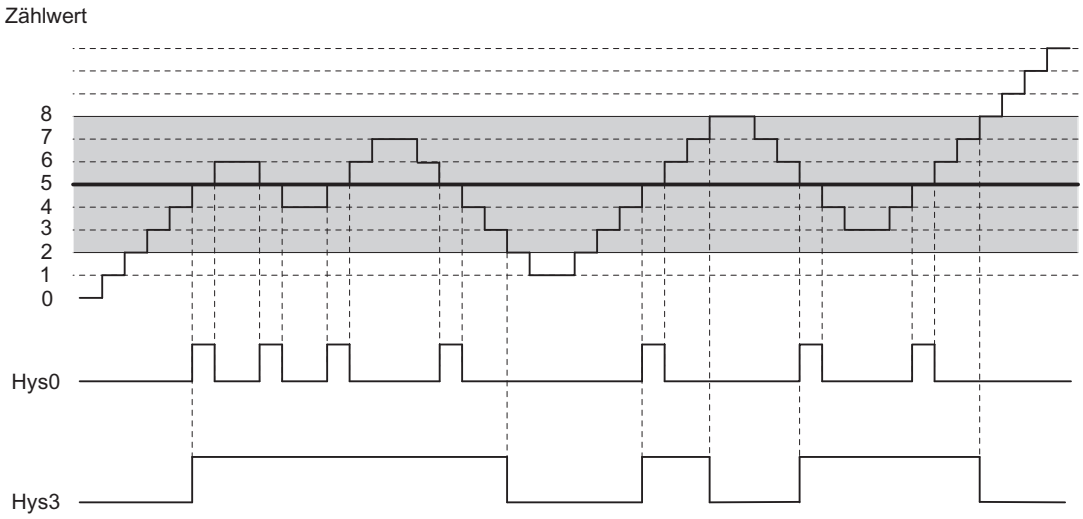

Wirkungsweise bei Vergleichswert erreicht, Impulsdauer ausgeben

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Impuls bei Erreichen des Vergleichswertes", "keine Hauptzählrichtung" und "Impulsdauer > 0" parametriert.

Mit dem Erreichen der Vergleichsbedingungen wird die Hysterese aktiv und ein Impuls der parametrierten Dauer ausgegeben.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv.

Wird die Hysterese aktiv, speichert das 1Count24V die Zählrichtung.

Wird der Hysteresebereich entgegen der gespeicherten Richtung verlassen, wird ein Impuls ausgegeben.

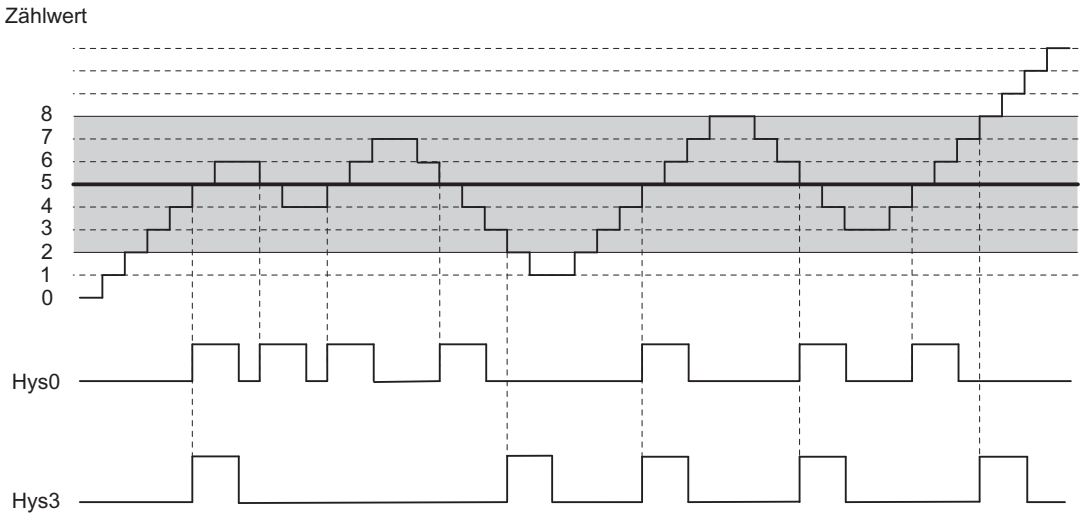

Impulsdauer festlegen

Impulsdauer

Die Impulsdauer beginnt mit dem Setzen des jeweiligen Digitalausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 2 ms.

Zur Anpassung an die verwendeten Aktoren kann die Impulsdauer vorgegeben werden. Die Impulsdauer gibt an, wie lange der Ausgang gesetzt werden soll. Die Impulsdauer kann in Schritten zu 2 ms zwischen 0 und 510 ms vorgewählt werden.

Wenn die Impulsdauer = 0 ist, wird der Ausgang so lange gesetzt, bis die Vergleichsbedingung nicht mehr erfüllt ist. Beachten Sie, dass die Zählimpulszeiten größer sein müssen als die minimalen Schaltzeiten des Digitalausgangs.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schaltet der Ausgang DO1 unmittelbar bei Erfüllung der Vergleichsbedingung und ist daher unabhängig vom Buszyklus. Der Zustand des virtuellen Ausgangs DO2 wird zum Zeitpunkt T_i gemeldet.

Zählbetriebsart auswählen

Beschreibung

Mit der Vorgabe einer Zählbetriebsart wählen Sie, mit welcher Funktionalität Sie das 1Count24V betreiben wollen.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne Tor)	Das 1Count24V zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit HW- oder SW-Tor)	Das 1Count24V zählt mit dem Öffnen des Tores ab dem Ladewert bis zur Zählgrenze.
Periodisches Zählen (Seite 187) (mit HW- oder SW-Tor)	Das 1Count24V zählt mit dem Öffnen des Tores ab dem Ladewert zwischen den Zählgrenzen.

Die Voreinstellung ist Betriebsart "Endlos Zählen".

Hauptzählrichtung auswählen

Beschreibung

Mit der Parametrierung einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie durch Parametrieren einer oberen Zählgrenze (Seite 347) den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Eine Hauptzählrichtung können Sie nur in den Betriebsarten "Einmaliges Zählen" und "Periodisches Zählen" auswählen.

Sie können folgendes Verhalten auswählen:

- Keine (ohne Hauptzählrichtung)
- Vorwärts
- Rückwärts

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Latch/Retrigger bei positiver Flanke (Seite 342)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt.
- Synchronisation bei positiver Flanke (Seite 344)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.
- Latchen bei positiver Flanke (Seite 345)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht).

Die Voreinstellung ist "Eingang".

Latch/Retrigger bei positiver Flanke

Latch/Retrigger

Mit dieser Funktion speichern Sie bei einer positiven Flanke am Digitaleingang den aktuellen internen Zählerstand des 1Count24V und retriggern den Zählvorgang. Dies bedeutet, dass der aktuelle interne Zählerstand zum Zeitpunkt der positiven Flanke abgespeichert (Latchwert) und dann das 1Count24V wieder mit dem Ladewert geladen wird und von dort weiterzählt.

Das folgende Bild zeigt Latch/Retrigger mit dem Ladewert = 0.

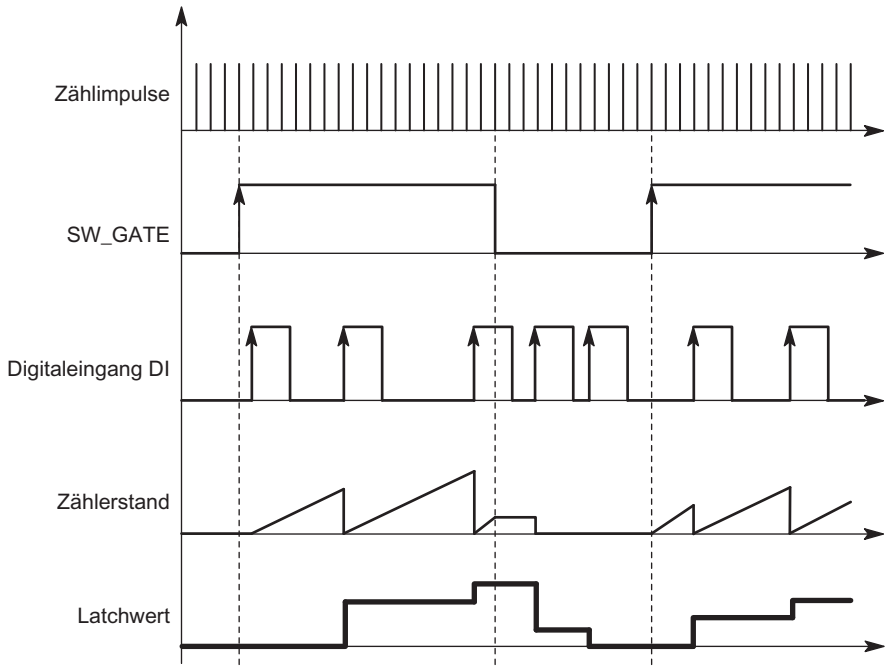

Zur Ausführung der Funktion muss die Zählbetriebsart mit dem SW-Tor freigegeben sein. Gestartet wird sie mit der ersten positiven Flanke am Digitaleingang.

In der Rückmeldeschnittstelle (Seite 349) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Status des Latch- und Retrigger-Signals.

Der Latchwert wird mit seinem RESET-Zustand vorbelegt:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Er wird durch das Öffnen des SW-Tors nicht verändert.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Wenn Sie das SW-Tor schließen, wirkt es nur unterbrechend; d. h. wenn Sie das SW-Tor erneut öffnen, wird der Zählvorgang fortgesetzt. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Auch im taktsynchronen Betrieb wird der Zählvorgang mit jeder Flanke am Digitaleingang gelatcht und getriggert. In der Rückmeldeschnittstelle wird der Zählerstand angezeigt, der zum Zeitpunkt der letzten Flanke vor T_i gültig war.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count24V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 349) gelesen werden.

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die steigende Flanke eines Referenzsignals am Eingang zum Setzen des 1Count24V auf den Ladewert.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

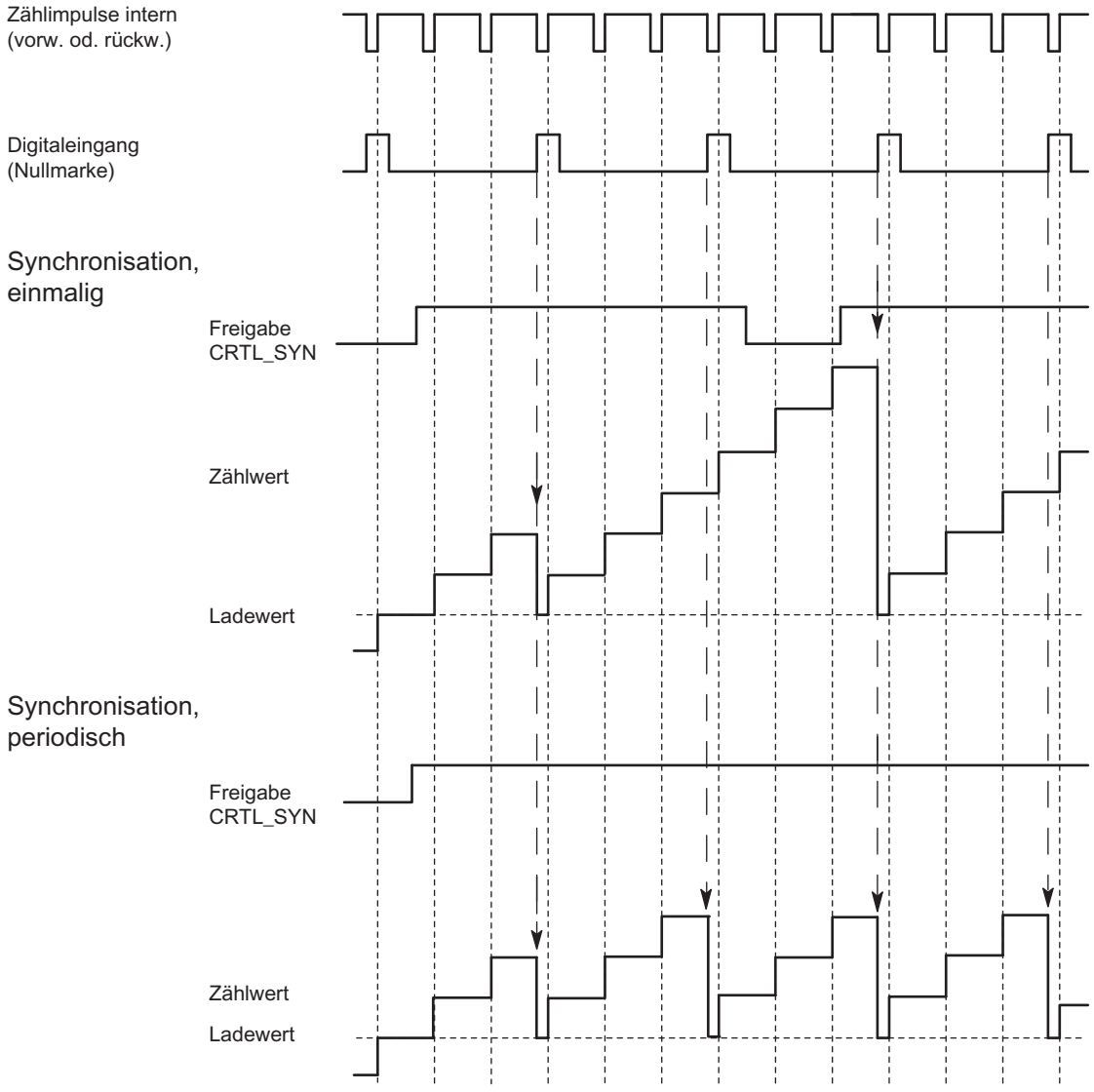

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 347) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.

- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count24V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count24V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 349) STS_SYN gesetzt. Es muss durch das Steuerbit (Seite 347) RES_STS zurückgesetzt werden.
- Als Referenzsignal kann das Signal eines prellfreien Schalters oder die Nullmarke eines Drehgebers dienen.
- Das Rückmeldebit (Seite 349) STS_DI zeigt den Pegel des Referenzsignals an.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 349) STS_SYN an, dass die steigende Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Latchen bei positiver Flanke

Latchen

Zählerstand und Latchwert werden mit ihren RESET-Zuständen vorbelegt:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Mit dem Öffnen des SW-Tors wird die Zählfunktion gestartet. Das 1Count24V beginnt beim Ladewert:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Das folgende Bild zeigt Latchen mit Ladewert = 0.

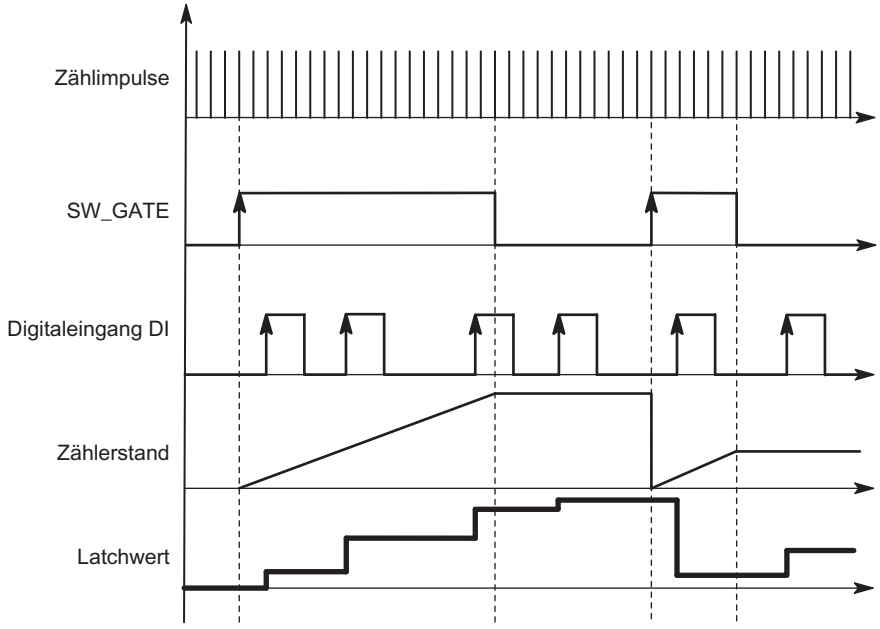

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der positiven Flanke am Digitaleingang DI.

In der Rückmeldeschnittstelle (Seite 349) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Pegel des Latch-Signals.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Im taktsynchronen Betrieb wird in der Rückmeldeschnittstelle der Zählerstand angezeigt, der zum Zeitpunkt der letzten positiven Flanke vor T_i gelatcht wurde.

Wenn Sie das SW-Tor schließen, wirkt es wie parametriert, abbrechend oder unterbrechend. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count24V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 349) gelesen werden.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 349) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Synchronisation auswählen

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" am Parameter "Funktion DI (Seite 342)" gewählt haben, dann können Sie hier zwischen einmaliger und periodischer Synchronisation wählen:

- Einmalig: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count24V mit dem Ladewert.
- Periodisch: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count24V mit dem Ladewert.

Obere Zählgrenze festlegen

Beschreibung

Mit der Parametrierung einer oberen Zählgrenze in Verbindung mit einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Die obere Zählgrenze kann einen Wert > 1 bis zur oberen Zählbereichsgrenze annehmen.

Steuer- und Rückmeldeschnittstelle für die Zählbetriebsarten

Steuerschnittstelle für die Zählbetriebsarten

Hinweis

Für das 1Count24V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count24V für die Zählbetriebsarten.

Adresse		Belegung			
Byte 0 bis 3		**Ladewert direkt, vorbereitend, Vergleichswert 1 oder 2**			
	Byte 0	**Verhalten von DO1, DO2 des 1Count24V**			
		Bit 2	Bit 1	Bit 0	Funktion DO1
		0	0	0	Ausgang
		0	0	1	Einschalten bei Zählerstand ≥ Vergleichswert
		0	1	0	Einschalten bei Zählerstand ≤ Vergleichswert
		0	1	1	Impuls beim Erreichen des Vergleichswertes
		1	0	0	Schalten an Vergleichswerten
		1	0	1	gesperrt
		1	1	0	gesperrt
		1	1	1	gesperrt
			Bit 5	Bit 4	Funktion DO2
			0	0	Ausgang
			0	1	Einschalten bei Zählerstand ≥ Vergleichswert
			1	0	Einschalten bei Zählerstand ≤ Vergleichswert
			1	1	Impuls beim Erreichen des Vergleichswertes
		Bit 3, 6 und 7: reserviert = 0			
	Byte 1 bis 3	Byte 1	Hysterese DO1, DO2 (Wertebereich 0 … 255)		
		Byte 2	Impulsdauer DO1, DO2 (Wertebereich 0 … 255)		
		Byte 3	reserviert = 0		
					Bezeichnung
Byte 4		Bit 7	Fehlerquittung		EXTF_ACK
		Bit 6	Freigabe DO2		CTRL_DO2
		Bit 5	Steuerbit DO2		SET_DO2
		Bit 4	Freigabe DO1		CTRL_DO1
		Bit 3	Steuerbit DO1		SET_DO1
		Bit 2	Anstoß Rücksetzen Statusbits		RES_STS
		Bit 1	Freigabe Synchronisation		CTRL_SYN
		Bit 0	Steuerbit SW-Tor		SW_GATE
Byte 5		Bit 7	reserviert = 0		
		Bit 6	reserviert = 0		
		Bit 5	reserviert = 0		
		Bit 4	Funktion und Verhalten von DO1, DO2 ändern		C_DOPARAM
		Bit 3	Vergleichswert 2 laden		CMP_VAL2
		Bit 2	Vergleichswert 1 laden		CMP_VAL1
		Bit 1	Zähler vorbereitend laden		LOAD_PREPARE
		Bit 0	Zähler direkt laden		LOAD_VAL
Byte 6 bis 7		reserviert = 0 [1]			

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
C_DOPARAM	Funktion und Verhalten von DO1, DO2 ändern Die Werte aus Byte 0 bis 2 werden als neue Funktion, Hysterese und Impulsdauer von DO1, DO2 übernommen. Dabei ist folgender Fehler möglich: Die Bedingungen für das Verhalten "Schalten an Vergleichswerten" sind nicht erfüllt.
CMP_VAL1	Vergleichswert 1 laden Der Wert aus Byte 0 bis 3 wird mit dem Steuerbit "Vergleichswert 1 laden" (CMP_VAL1) in den Vergleichswert 1 übertragen.
CMP_VAL2	Vergleichswert 2 laden Der Wert aus Byte 0 bis 3 wird mit dem Steuerbit "Vergleichswert 2 laden" (CMP_VAL2) in den Vergleichswert 2 übertragen.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
CTRL_DO2	Freigabe DO2 Mit diesem Bit geben Sie den Ausgang DO2 frei.
CTRL_SYN	Sie geben damit die Synchronisation frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Zähler vorbereitend laden Der Wert aus Byte 0 bis 3 wird als Ladewert übernommen.
LOAD_VAL	Der Wert aus Byte 0 bis 3 wird als neuer Zählwert direkt geladen
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SET_DO1	Steuerbit DO1 Digitalausgang DO1 ein- und ausschalten, wenn CRTL_DO1 gesetzt ist.
SET_DO2	Steuerbit DO2 Digitalausgang DO2 ein- und ausschalten, wenn CRTL_DO2 gesetzt ist.
SW_GATE	Steuerbit SW-Tor Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für die Zählbetriebsarten

Hinweis

Für das 1Count24V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count24V für die Zählbetriebsarten.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Zählwert oder abgespeicherter Zählwert bei Latch-Funktion am Digitaleingang		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO1
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	Status DO2	STS_DO2
	Bit 3	Status DO1	STS_DO1
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE
Byte 6	Bit 7	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung	STS_ND
	Bit 6	Untere Zählgrenze	STS_UFLW
	Bit 5	Obere Zählgrenze	STS_OFLW
	Bit 4	Status Vergleicher 2	STS_CMP2
	Bit 3	Status Vergleicher 1	STS_CMP1
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	Status Synchronisation	STS_SYN
Byte 7	reserviert = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 347) EXTF_ACK quittiert werden Diagnosemeldung, falls parametriert.
ERR_DO1	Kurzschluss / Drahtbruch / Übertemperatur durch Überlast am Ausgang DO1 Das Fehlerbit muss durch das Steuerbit (Seite 347) EXTF_ACK quittiert werden Diagnosemeldung, falls parametriert.

Bit	Erläuterung
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 347) LOAD_VAL, LOAD_PREPARE, CMP_VAL1, CMP_VAL2 und C_DOPARAM dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_CMP1	Status Vergleicher 1 Das Statusbit STS_CMP1 zeigt an, dass der Ausgang eingeschaltet ist oder eingeschaltet war. Es muss mit dem Steuerbit (Seite 347) RES_STS quittiert werden. Wird das Statusbit quittiert, wenn der Ausgang noch eingeschaltet ist, wird das Bit sofort wieder gesetzt. Dieses Bit wird auch gesetzt, wenn bei nicht freigegeben DO1 das Steuerbit SET_DO1 betätigt wird.
STS_CMP2	Status Vergleicher 2 Das Statusbit STS_CMP2 zeigt an, dass der Ausgang eingeschaltet ist oder eingeschaltet war. Es muss mit dem Steuerbit (Seite 347) RES_STS quittiert werden. Wird das Statusbit quittiert, wenn der Ausgang noch eingeschaltet ist, wird das Bit sofort wieder gesetzt. Dieses Bit wird auch gesetzt, wenn bei nicht freigegeben DO2 das Steuerbit SET_DO2 betätigt wird.
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_DO1	Status DO1 Das Statusbit STS_DO1 zeigt den Zustand des Digitalausgangs DO1 an.
STS_DO2	Status DO2 Das Statusbit STS_DO2 zeigt den Zustand des virtuellen Digitalausgangs DO2 an.
STS_GATE	Status internes Tor: Es wird gezählt.
STS_LOAD	Ladefunktion läuft
STS_ND	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung. Das Bit muss durch das Steuerbit (Seite 347) RES_STS zurückgesetzt werden.
STS_OFLW STS_UFLW	Obere Zählgrenze überschritten Untere Zählgrenze unterschritten Beide Bits müssen zurückgesetzt werden.
STS_SYN	Status Synchronisation Nach erfolgreicher Synchronisation ist das STS_SYN-Bit gesetzt. Es muss durch das Steuerbit (Seite 347) RES_STS zurückgesetzt werden.

Parameter für Messen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count24V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count24V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO 1 abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.
DO 1 Ersatzwert schalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der parametrierte Ersatzwert für den Digitalausgang wird durchgeschaltet. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen. Ist für den Ausgang "Impuls bei Erreichen des Vergleichswertes" parametriert, steht nur für die Impulsdauer der Ersatzwert = 1 an.	Die geänderten Parameter werden übernommen und wirksam.
DO 1 letzten Wert halten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Zustand des Digitalausgangs wird beibehalten. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.

[1] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count24V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count24V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 366) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count24V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung

Zähl- und Richtungsauswertung auswählen

Signalauswertung A*, B*

Die Signalauswertung über A*, B* ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Impuls und Richtung (Seite 354)
  Bei 24 V-Impulsgebern mit Richtungspegel muss gewährleistet sein, dass zwischen Richtungssignal B* und Zählsignal A* eine Zeitspanne von mindestens 5 µs / 50 µs liegt, je nach parametriertem Eingangsfilter.
  Das folgende Bild zeigt die Zeitspanne zwischen Richtungssignal und Zählsignal.

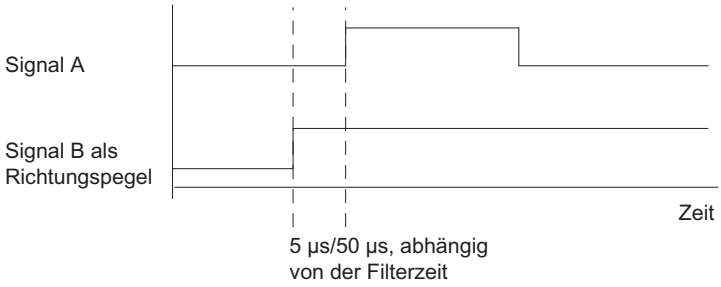

- Drehgeber
  Wenn Sie einen 24 V-Drehgeber mit zwei um 90 Grad phasenversetzten Spuren am Zähl- und Richtungseingang anschließen, können Sie in allen Messbetriebsarten eine Einfachauswertung (Seite 211) parametrieren.
  Am Richtungseingang B* können Sie die Zählrichtung (Seite 360) invertieren.
  An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie mit dem 1Count24V beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Hinweis

Die Angabe 100 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A* bzw. B* bezogen.

Auswertung "Impuls und Richtung"

Impuls und Richtung

Bei dieser Auswertungsart wird als Richtungsvorgabe der Pegel am Richtungseingang B* benutzt.

Ein unbeschalteter Eingang entspricht der Zählrichtung "vorwärts", wenn Sie beim Parameter Signalauswertung "Impuls und Richtung" gewählt haben.

Das folgende Bild zeigt die Signale eines 24 V-Impulsgebers mit Richtungspegel.

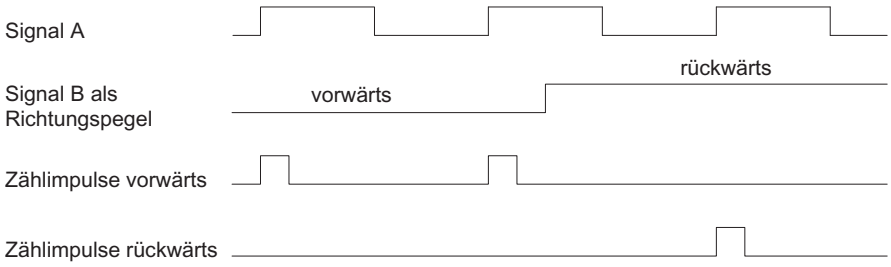

Messverfahren

Übersicht

Messverfahren

Sie können beim Parameter "Messverfahren" zwischen folgenden Messverfahren wählen:

- mit Integrationszeit (Seite 355)
- kontinuierlich (Seite 356)

Messung mit Integrationszeit

Ablauf der Messung mit Integrationszeit

Die Messung wird während der von Ihnen parametrierten Integrationszeit durchgeführt. Wenn die Integrationszeit abgelaufen ist, wird der Messwert aktualisiert.

Dabei wird das Ende einer Messung mit dem Statusbit (Seite 368) STS_CMP1 gemeldet. Zurückgesetzt wird dieses Bit durch das Steuerbit (Seite 366) RES_STS in der Steuerschnittstelle.

Lagen in der parametrierten Integrationszeit nicht mindestens zwei steigende Flanken, so wird als Messwert 0 zurückgemeldet.

Bis zum Ende der ersten Integrationszeit wird der Wert -1 zurückgemeldet.

Sie können die Integrationszeit für die nächste Messung im Betrieb ändern.

Drehrichtungsumkehr

Wenn während einer Integrationszeit eine Drehrichtungsumkehr auftritt, dann ist der Messwert für diese Messperiode unbestimmt. Wenn Sie die Rückmeldebits (Seite 368) STS_C_UP und STS_C_DN (Richtungsauswertung) auswerten, können Sie auf eine eventuelle Prozessunregelmäßigkeit reagieren.

Siehe auch

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb (Seite 359)

Kontinuierliche Messung

Messprinzip

Das 1Count24V zählt jede positive Flanke eines Impulses und ordnet ihr einen Zeitwert in µs zu.

Die Aktualisierungszeit gibt an, in welchem Zeitabstand der Messwert vom Modul in der Rückmeldeschnittstelle aktualisiert wird.

Bei einer Impulsfolge mit einem oder mehreren Impulsen pro Aktualisierungszeit gilt:

Dynamische Messzeit =	Zeitwert des letzten Impulses im aktuellen Aktualisierungszeitintervall minus Zeitwert des letzten Impulses im vorangegangenen Aktualisierungszeitintervall

Nach dem Ablauf der Aktualisierungszeit wird ein neuer Messwert mit der dynamischen Messzeit errechnet und ausgegeben.

Enthält die aktuelle Aktualisierungszeit keinen Impuls, so ergibt sich folgende dynamische Messzeit:

Dynamische Messzeit =	Zeitwert der aktuellen abgelaufenen Aktualisierungszeit minus dem Zeitwert des letzten Impulses

Nach dem Ablauf einer Aktualisierungszeit wird ein geschätzter Messwert mit der dynamischen Messzeit errechnet unter der Annahme, dass ein Impuls am Ende der Aktualisierungszeit aufgetreten ist.

Ist der geschätzte Messwert "1 Impuls pro dynamische Messzeit" bei der Frequenz– und Drehzahlmessung kleiner als der letzte Messwert, dann wird dieser geschätzte Messwert als neuer Messwert ausgegeben. Bei der Periodendauermessung wird die dynamische Messzeit als geschätzte Periodendauer ausgegeben, wenn die dynamische Messzeit größer als die letzte gemessene Periodendauer ist.

Das folgende Bild zeigt das Messprinzip.

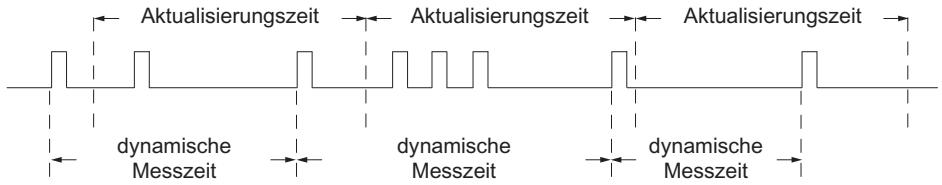

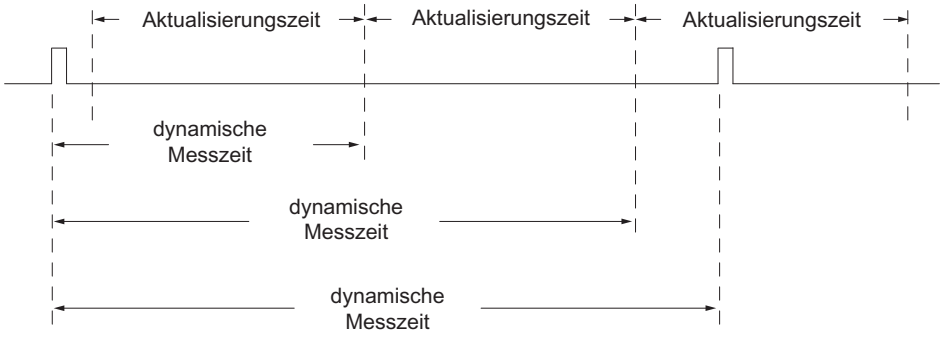

Das 1Count24V misst kontinuierlich. Sie geben beim Parametrieren eine Aktualisierungszeit vor.

In der Zeit bis zum Ende der ersten abgelaufenen Aktualisierungszeit wird der Wert "–1" zurückgemeldet.

Die kontinuierliche Messung beginnt nach dem Öffnen des Tores mit dem ersten Impuls der zu messenden Impulsfolge. Der erste Messwert kann frühestens nach dem zweiten Impuls berechnet werden.

Nach jedem Ablauf der Aktualisierungszeit wird in der Rückmeldeschnittstelle ein Messwert ausgegeben (Frequenz, Periodendauer oder Drehzahl). Dabei wird das Ende einer Messung mit dem Statusbit (Seite 368) STS_CMP1 gemeldet. Zurückgesetzt wird dieses Bit durch vollständiges Quittungsprinzip mit den Bits RES_STS und RES_STS_A.

Wenn in einer Aktualisierungszeit eine Drehrichtungsumkehr auftritt, dann ist der Messwert für diese Messperiode unbestimmt. Durch Auswertung der Rückmeldebits (Seite 368) STS_C_DN und STS_C_UP (Richtungsauswertung) können Sie auf eine eventuelle Prozessunregelmäßigkeit reagieren.

Das folgende Bild zeigt das Prinzip der kontinuierlichen Messung am Beispiel einer Frequenzmessung.

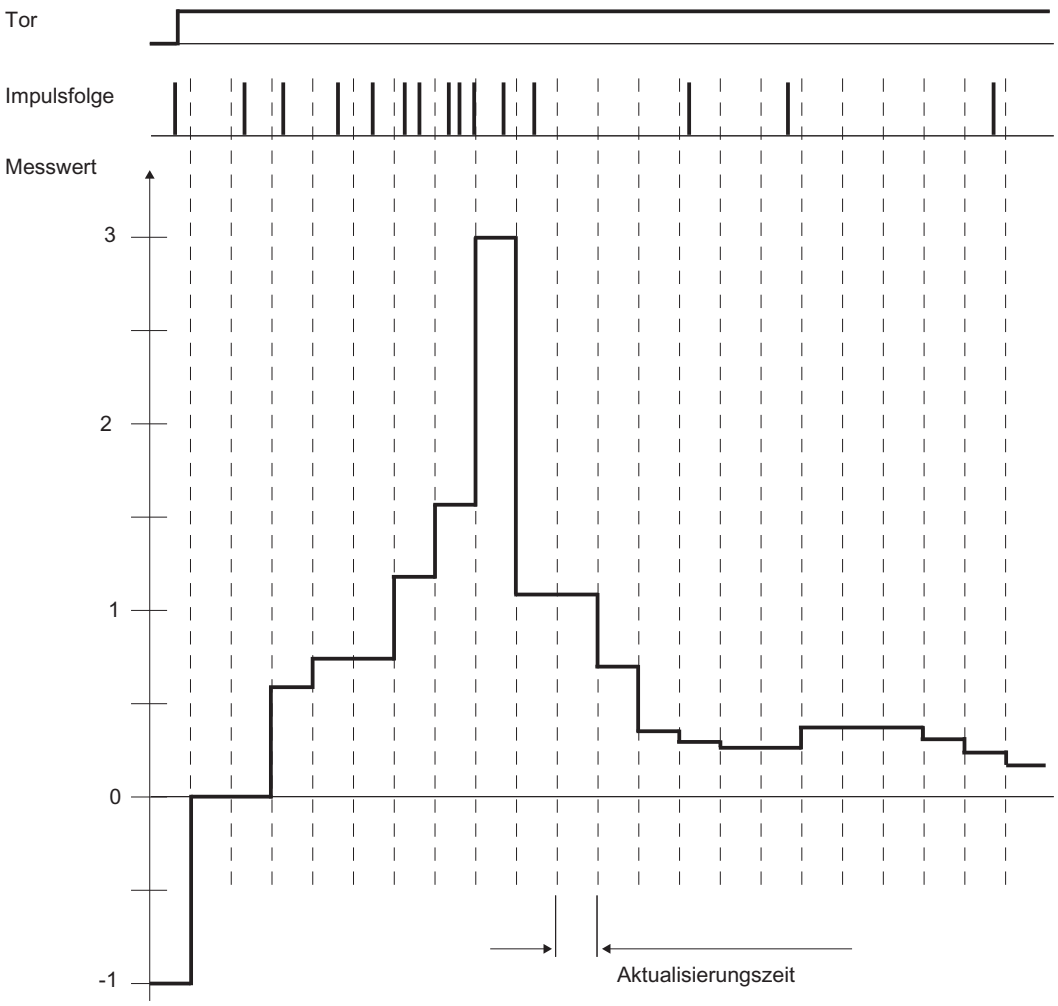

Torsteuerung

Zur Steuerung des 1Count24V müssen Sie Torfunktionen einsetzen.

Taktsynchroner Betrieb

Im taktsynchronen Betrieb übernimmt das 1Count24V in jedem Buszyklus Steuerbits und Steuerwerte aus der Steuerschnittstelle und meldet die Reaktion darauf noch im selben Zyklus zurück.

Das 1Count24V übergibt in jedem Zyklus einen Messwert und die Statusbits, wie sie zum Zeitpunkt T_i gültig waren.

Die Messung beginnt und endet jeweils zum Zeitpunkt T_i.

Siehe auch

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb (Seite 359)

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb

Ist die Integrationszeit / Aktualisierungszeit mehrere T_{DP}-Zyklen lang, dann können Sie im Anwenderprogramm den neuen Messwert am Statusbit (Seite 368) STS_CMP1 (Messung beendet) der Rückmeldeschnittstelle erkennen. Dies ermöglicht eine Überwachung des Messvorgangs oder eine Synchronisation mit dem Messvorgang. Allerdings dauert das Quittieren dieser Meldung 4 T_{DP}-Zyklen. In diesem Fall beträgt die minimale Integrationszeit/ Aktualisierungszeit (4 × T_{DP}).

Wenn die Applikation einen Jitter der Integrationszeit / Aktualisierungszeit von einem T_{DP} und einen über mehrere Zyklen konstanten Messwert tolerieren kann, brauchen Sie das Statusbit STS_CMP1 nicht kontinuierlich auszuwerten. Dann sind auch Integrationszeiten / Aktualisierungszeiten von (1 × T_{DP}) bis (3 × T_{DP}) möglich.

Durch einen Verlust der Taktsynchronität im letzten T_{DP}-Zyklus der Integrationszeit verlängert sich die Integrationszeit / Aktualisierungszeit um einen T_{DP}-Zyklus. Der Messwert wird dadurch nicht verfälscht.

Hinweis

Sie dürfen die Wertebereichsgrenzen für die Integrationszeit / Aktualisierungszeit nicht überschreiten (siehe Tabellen zu jeder einzelnen Messbetriebsart).

Eine Verletzung der Wertebereichsgrenzen führt zu einem Parametrierfehler und das 1Count24V geht nicht in den taktsynchronen Betrieb.

Hinweis

Beim Ändern der Konfiguration vom nicht taktsynchronen in den taktsynchronen Betrieb und umgekehrt müssen Sie immer den Parameter Integrationszeit / Aktualisierungszeit anpassen, wenn Sie die Länge der Integrationszeit / Aktualisierungszeit beibehalten wollen.

Filter für Eingangssignale auswählen

Eingangsfilter

Zur Filterung der Signale am Zähleingang A*, Richtungseingang B* und Digitaleingang DI können Sie entsprechend der Mindestimpulsdauer bzw. der maximalen Signalfrequenz einen Filter einschalten.

Sie können für jeden Eingang jeweils eine Mindestimpulsbreite von 2,5 µs bzw. 25 µs wählen.

Gebereingänge auswählen

Gebereingänge

An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Verhalten des Ausgangs bei den Messbetriebsarten

Verhalten des Ausgangs DO1 auswählen

Verhalten des Ausgangs bei Messbetriebsarten

Der Digitalausgang des 1Count24V ist parametrierbar.

Sie können für Frequenzmessung, Drehzahlmessung oder Periodendauermessung jeweils einen oberen und einen unteren Grenzwert hinterlegen, bei deren Verletzung der Digitalausgang DO1 aktiviert wird. Diese Grenzwerte sind parametrierbar und mit der Ladefunktion änderbar.

Sie können die Funktion und das Verhalten des Digitalausgangs im Betrieb ändern. Die neue Funktion wird sofort wirksam, im taktsynchronen Betrieb immer zum Zeitpunkt T_i.

Sie können in der Klappliste zwischen folgenden Funktionen auswählen:

- Ausgang (Seite 361)
- Außerhalb der Grenzwerte (Seite 361) (Grenzwertüberwachung)
- Unterschreiten des unteren Grenzwertes (Seite 361) (Grenzwertüberwachung)
- Überschreiten des oberen Grenzwertes (Seite 361) (Grenzwertüberwachung)

Ausgang

Ausgang

Wenn Sie den Ausgang ein- oder ausschalten wollen, müssen Sie ihn mit dem Steuerbit (Seite 366) CTRL_DO1 freigeben.

Mit dem Steuerbit (Seite 366) SET_DO1 schalten Sie den Ausgang ein und aus.

Mit dem Statusbit (Seite 368) STS_DO1 in der Rückmeldeschnittstelle können Sie den Status des Ausgangs abfragen.

Im taktsynchronen Betrieb schaltet der Ausgang zum Zeitpunkt T_o.

Grenzwertüberwachung

Grenzwertüberwachung

Das folgende Bild zeigt das Prinzip der Grenzwertüberwachung.

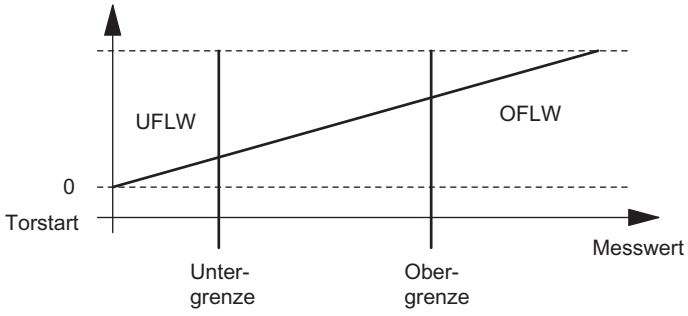

Nach abgelaufener Integrationszeit wird der ermittelte Messwert (Frequenz, Drehzahl oder Periodendauer) mit den parametrierten Grenzwerten verglichen.

Unterschreitet der aktuelle Messwert den parametrierten unteren Grenzwert (Messwert < unterer Grenzwert), wird das Bit STS_UFLW = 1 in der Rückmeldeschnittstelle (Seite 368) gesetzt.

Überschreitet der aktuelle Messwert den parametrierten oberen Grenzwert (Messwert > oberer Grenzwert), wird das Bit STS_OFLW = 1 in der Rückmeldeschnittstelle (Seite 368) gesetzt.

Diese Bits müssen Sie mit dem Steuerbit (Seite 366) RES_STS quittieren.

Befindet sich nach dem Quittieren der Messwert noch oder wieder außerhalb der Grenzwerte, wird das entsprechende Statusbit erneut gesetzt.

Wenn Sie den unteren Grenzwert = 0 setzen, schalten Sie damit die dynamische Überwachung auf "Unterschreiten des unteren Grenzwertes" aus.

Die folgende Tabelle zeigt, wie der freigegebene Digitalausgang DO1 je nach Parametrierung durch die Grenzwertüberwachung gesetzt werden kann:

Parameter "Funktion DO1"	DO1 wird gesetzt bei ...
Außerhalb der Grenzwerte	Messwert < unterer Grenzwert ODER Messwert > oberer Grenzwert
Unterschreiten des unteren Grenzwertes	Messwert < unterer Grenzwert
Überschreiten des oberen Grenzwertes	Messwert > oberer Grenzwert

Im taktsynchronen Betrieb schaltet der Ausgang nach Ende der Messung zum Zeitpunkt T_i.

Ersatzwert DO1 auswählen

Ersatzwert DO1

Wenn Sie unter "Verhalten bei CPU/Master-STOP (Seite 352)" den Parameter "DO1 Ersatzwert schalten" ausgewählt haben, dann wählen Sie hier den Ersatzwert aus, der am Digitalausgang DO1 ausgegeben werden soll.

Diagnose DO1 aktivieren

Diagnose DO1

Die "Diagnose DO1" (Leitungsbruch, Kurzschluss) ist nur bei Impulslängen > 90 ms am Digitalausgang DO1 möglich.

Aktivieren Sie das Optionskästchen für die Diagnose am Ausgang DO1.

Messbetriebsart auswählen

Beschreibung

Mit der Vorgabe einer Messbetriebsart wählen Sie, mit welcher Funktionalität Sie das 1Count24V betreiben wollen.

	Beschreibung	
Betriebsart	Messung mit Integrationszeit	Kontinuierliche Messung
Frequenzmessung (Seite 195) (mit HW- oder SW-Tor)	Das 1Count24V zählt die Impulse, die in einer vorgegebenen Integrationszeit eintreffen.	Das 1Count24V zählt die Impulse, die in einer dynamischen Messzeit eintreffen.
Drehzahlmessung (Seite 196) (mit HW- oder SW-Tor)	Das 1Count24V zählt die Impulse, die in einer vorgegebenen Integationszeit von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.	Das 1Count24V zählt die Impulse, die in einer dynamischen Messzeit von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.
Periodendauermessung (Seite 198) (mit HW- oder SW-Tor)	Das 1Count24V misst die Zeit zwischen zwei positiven Flanken des Zählsignals, indem es die Impulse einer internen quarzgenauen Referenzfrequenz (16 MHz) in einer vorgegebenen Integrationszeit zählt.	Das 1Count24V gibt die dynamische Messzeit als Periodendauer an. Ist die Periodendauer kleiner als die Aktualisierungszeit, dann wird für die Periodendauer ein Mittelwert gebildet.

Die Voreinstellung ist Betriebsart "Frequenzmessung".

Auflösung der Periodendauer auswählen

Beschreibung

Der Wert der ermittelten Periodendauer wird in der Einheit 1 µs oder 1/16 µs zur Verfügung gestellt.

Wählen Sie die gewünschte Einheit für die Auflösung aus.

Mögliche Messbereiche mit Fehlerangaben

- Periodendauermessung mit Integrationszeit
  - Messbereiche und Fehler bei einer Auflösung von 1 µs:

Integrationszeit	T_{min} ± absoluter Fehler	T_{min} ± absoluter Fehler
100 s	1 µs* (10 ± 0)	1 µs* (100 000 000 ± 10 000)
10 s	1 µs* (10 ± 0)	1 µs* (10 000 000 ± 1 000)
1 s	1 µs* (10 ± 0)	1 µs* (1 000 000 ± 100)
0,1 s	1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
0,01 s	1 µs* (10 ± 0)	1 µs* (10 000 ± 1)

  - Messbereiche und Fehler bei einer Auflösung von 1/16 µs:

Integrationszeit	T_{min} ± absoluter Fehler	T_{min} ± absoluter Fehler
100 s	1/16 µs* (160 ± 0)	1/16 µs* (1 600 000 000 ± 160 000)
10 s	1/16 µs* (160 ± 0)	1/16 µs* (160 000 000 ± 16 000)
1 s	1/16 µs* (160 ± 0)	1/16 µs* (16 000 000 ± 1 600)
0,1 s	1/16 µs* (160 ± 0)	1/16 µs* (1 600 000 ± 160)
0,01 s	1/16 µs* (160 ± 0)	1/16 µs* (160 000 ± 16)

- Kontinuierliche Periodendauermessung:
  - Messbereiche und Fehler bei einer Auflösung von 1 µs:

Periodendauer T_{min} ± absoluter Fehler	Periodendauer T_{min} ± absoluter Fehler
1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
1 µs* (100 ± 0)	1 µs* (1 000 000 ± 100)
1 µs* (1 000 ± 0)	1 µs* (10 000 000 ± 1 002)
1 µs* (10 000 ± 1)	1 µs* (100 000 000 ± 10 020)

  - Messbereiche und Fehler bei einer Auflösung von 1/16 µs:

Periodendauer T_{min} ± absoluter Fehler	Periodendauer T_{min} ± absoluter Fehler
1/16 µs* (160 ± 1)	1/16 µs* (1 600 000 ± 160)
1/16 µs* (1 600 ± 1)	1/16 µs* (16 000 000 ± 1 600)
1/16 µs* (16 000 ± 3)	1/16 µs* (160 000 000 ± 16 000)
1/16 µs* (160 000 ± 20)	1/16 µs* (1 600 000 000 ± 160 000)

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.

Die Voreinstellung ist "Eingang".

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 368) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Unteren und oberen Grenzwert festlegen

Unterer und oberer Grenzwert

Sie können für Frequenzmessung, Drehzahlmessung oder Periodendauermessung jeweils einen oberen und einen unteren Grenzwert hinterlegen, bei deren Verletzung der Digitalausgang DO1 aktiviert (Seite 360) wird. Diese Grenzwerte können Sie parametrieren und über die Ladefunktion ändern.

Für die unteren und oberen Grenzwerte sind folgende Wertebereiche zulässig:

Messbetriebsart	Unterer Grenzwert (UG)	Oberer Grenzwert
Frequenzmessung	0 bis 99 999 999 × 10^{-3} Hz	UG+1 bis 100 000 000 × 10^{-3} Hz
Drehzahlmessung	0 bis 24 999 999 × 10^{-3}/min	UG+1 bis 25 000 000 × 10^{-3}/min
Periodendauermessung (Auflösung 1 µs)	0 bis 119 999 999 µs	UG+1 bis 120 000 000 µs
Periodendauermessung (Auflösung 1/16 µs)	0 bis 1 919 999 999 × 1/16 µs	UG+1 bis 1 920 000 000 × 1/16 µs

Messzeit festlegen

Beschreibung

Wenn Sie eine Messbetriebsart ausgewählt haben, dann aktualisiert die Baugruppe die Messwerte zyklisch. Die Aktualisierungszeit / Integrationszeit ist das Produkt aus Messzeit × 10 ms. Die Messzeit geben Sie als Faktor n vor.

Detaillierte Beschreibungen zu den Wertebereichen für die Messzeit finden Sie unter "Grundsätzliches zum Messen (Seite 190)".

Impulse pro Geberumdrehung eintragen

Impulse pro Geberumdrehung

Für die Betriebsart "Drehzahlmessung (Seite 196)" müssen Sie zusätzlich die Impulse pro Geberumdrehung parametrieren.

Den Wert für die Anzahl der Impulse pro Geberumdrehung finden Sie auf dem Typenschild oder in den technischen Daten Ihres Gebers.

Steuer- und Rückmeldeschnittstelle für die Messbetriebsarten

Steuerschnittstelle für die Messbetriebsarten

Hinweis

Für das 1Count24V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count24V für die Messbetriebsarten.

Adresse	Belegung				
Byte 0 bis 3	**Unterer oder oberer Grenzwert**				
	Funktion von DO1				
	Byte 0	Bit 1	Bit 0	Funktion DO1	
		0	0	Ausgang	
		0	1	Außerhalb der Grenzwerte	
		1	0	Unterschreiten des unteren Grenzwertes	
		1	1	Überschreiten des oberen Grenzwertes	
	Byte 1 bis 3	reserviert = 0			
	Integrationszeit				
	Byte 0, 1	Integrationszeit [n × 10 ms] (Wertebereich 1 … 1000 / 12000)			
	Byte 2, 3	reserviert = 0			
					Bezeichnung
Byte 4	Bit 7	Fehlerquittung			EXTF_ACK
	Bit 6	reserviert = 0			
	Bit 5	reserviert = 0			
	Bit 4	Freigabe DO1			CTRL_DO1
	Bit 3	Steuerbit DO1			SET_DO1
	Bit 2	Anstoß Rücksetzen Statusbits			RES_STS
	Bit 1	reserviert = 0			
	Bit 0	Steuerbit SW-Tor			SW_GATE
Byte 5	Bit 7	reserviert = 0			
	Bit 6	reserviert = 0			
	Bit 5	reserviert = 0			
	Bit 4	Funktion von DO1 ändern			C_DOPARAM
	Bit 3	reserviert = 0			
	Bit 2	Integrationszeit ändern			C_INTTIME
	Bit 1	Oberen Grenzwert laden			LOAD_PREPARE
	Bit 0	Unteren Grenzwert laden			LOAD_VAL
Byte 6 bis 7	reserviert = 0 [1]				

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
C_DOPARAM	Funktion von DO1 ändern Der Wert aus Byte 0 wird als neue Funktion von DO1 übernommen.
C_INTTIME	Integrationszeit ändern Der Wert aus Byte 0 und 1 wird als neue Integrationszeit für die nächste Messung übernommen.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Oberen Grenzwert laden Der Wert aus Byte 0 bis 3 wird als neuer oberer Grenzwert übernommen.
LOAD_VAL	Unteren Grenzwert laden Der Wert aus Byte 0 bis 3 wird als neuer unterer Grenzwert übernommen.
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SET_DO1	Steuerbit DO1 Digitalausgang DO1 ein- und ausschalten, wenn CRTL_DO1 gesetzt ist.
SW_GATE	Steuerbit SW-Tor Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für die Messbetriebsarten

Hinweis

Für das 1Count24V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count24V für die Messbetriebsarten.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Messwert		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO1
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	Status DO1	STS_DO1
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE
Byte 6	Bit 7	reserviert = 0	
	Bit 6	Unterer Grenzwert Messbereich	STS_UFLW
	Bit 5	Oberer Grenzwert Messbereich	STS_OFLW
	Bit 4	reserviert = 0	
	Bit 3	Messung beendet	STS_CMP1
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	reserviert = 0	
Byte 7	Reserve = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 366) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_DO1	Kurzschluss / Drahtbruch / Übertemperatur am Ausgang DO1 Das Fehlerbit muss durch das Steuerbit (Seite 366) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.

Bit	Erläuterung
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 366) LOAD_VAL, LOAD_PREPARE, C_DOPARAM und C_INTTIME dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_CMP1	Messung beendet Nach jedem abgelaufenen Zeitintervall (Aktualisierungszeit / Integrationszeit) wird der Messwert aktualisiert. **Messung mit Integrationszeit:** Dabei wird das Ende einer Messung (nach Ablauf des Zeitintervalls) mit dem Statusbit STS_CMP1 gemeldet. **Kontinuierliche Messung:** Am Ende der Aktualisierungszeit wird das Ende der Messung mit dem Statusbit STS_CMP1 gemeldet, wenn ein gemessener Wert ausgegeben wird. Bei Ausgabe eines geschätzten Messwertes bleibt das Bit "0". Zurückgesetzt wird dieses Bit durch das Steuerbit (Seite 366) RES_STS in der Steuerschnittstelle.
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_DO1	Status DO1
STS_GATE	Status internes Tor: Es wird gemessen.
STS_LOAD	Ladefunktion läuft
STS_OFLW STS_UFLW	Oberer Grenzwert überschritten Unterer Grenzwert unterschritten Beide Bits müssen zurückgesetzt werden.

Parameter für Wegerfassen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count24V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count24V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO 1 abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1)]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.

1) Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count24V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count24V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 379) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count24V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung

Zähl- und Richtungsauswertung auswählen

Signalauswertung A*, B*

Die Signalauswertung über A*, B* ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Impuls und Richtung (Seite 372)
  Bei 24 V-Impulsgebern mit Richtungspegel muss gewährleistet sein, dass zwischen Richtungssignal B* und Zählsignal A* eine Zeitspanne von mindestens 5 µs / 50 µs liegt, je nach parametriertem Eingangsfilter.
  Das folgende Bild zeigt die Zeitspanne zwischen Richtungssignal und Zählsignal.

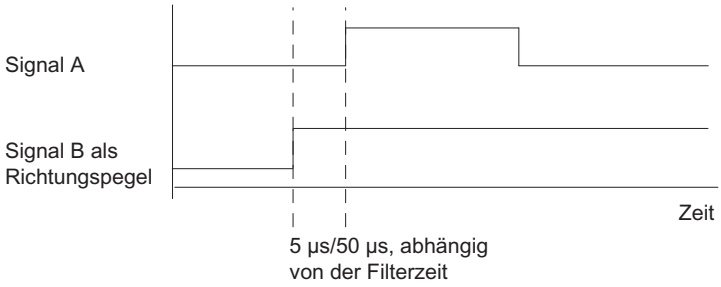

- Drehgeber (Seite 210)
  Wenn Sie einen 24 V-Drehgeber mit zwei um 90 Grad phasenversetzten Spuren am Zähl- und Richtungseingang anschließen, können Sie in allen Zählbetriebsarten eine Einfachauswertung (Seite 211) parametrieren. Alternativ können Sie Zweifach- (Seite 211) oder Vierfachauswertung (Seite 211) parametrieren.
  Bei allen Auswertungsarten können Sie die Zählrichtung (Seite 373) am Richtungseingang B* invertieren.
  An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie mit dem 1Count24V beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Hinweis

Die Angabe 100 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A* bzw. B* bezogen. Bei Zweifachauswertung ergibt sich somit maximal 200 kHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 400 kHz.

Auswertung "Impuls und Richtung"

Impuls und Richtung

Bei dieser Auswertungsart wird als Richtungsvorgabe der Pegel am Richtungseingang B* benutzt.

Ein unbeschalteter Eingang entspricht der Zählrichtung "vorwärts", wenn Sie beim Parameter Signalauswertung "Impuls und Richtung" gewählt haben.

Das folgende Bild zeigt die Signale eines 24 V-Impulsgebers mit Richtungspegel.

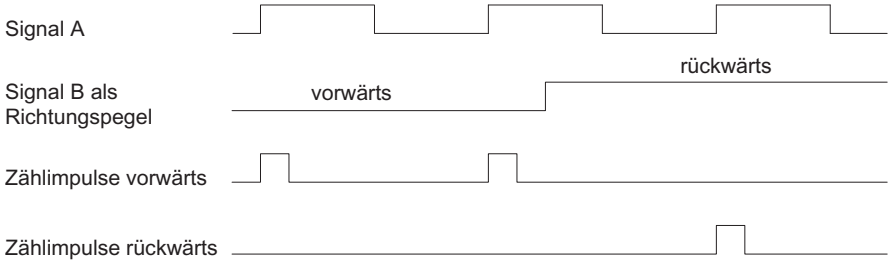

Filter für Eingangssignale auswählen

Eingangsfilter

Zur Filterung der Signale am Zähleingang A*, Richtungseingang B* und Digitaleingang DI können Sie entsprechend der Mindestimpulsdauer bzw. der maximalen Signalfrequenz einen Filter einschalten.

Sie können für jeden Eingang jeweils eine Mindestimpulsbreite von 2,5 µs bzw. 25 µs wählen.

Gebereingänge auswählen

Gebereingänge

An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Latch/Retrigger bei positiver Flanke (Seite 374)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt.
- Synchronisation bei positiver Flanke (Seite 376)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.
- Latchen bei positiver Flanke (Seite 377)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht).

Die Voreinstellung ist "Eingang".

Latch/Retrigger bei positiver Flanke

Latch/Retrigger

Mit dieser Funktion speichern Sie bei einer positiven Flanke am Digitaleingang den aktuellen internen Zählerstand des 1Count24V und retriggern den Zählvorgang. Dies bedeutet, dass der aktuelle interne Zählerstand zum Zeitpunkt der positiven Flanke abgespeichert (Latchwert) und dann das 1Count24V wieder mit dem Ladewert geladen wird und von dort weiterzählt.

Das folgende Bild zeigt Latch/Retrigger mit dem Ladewert = 0.

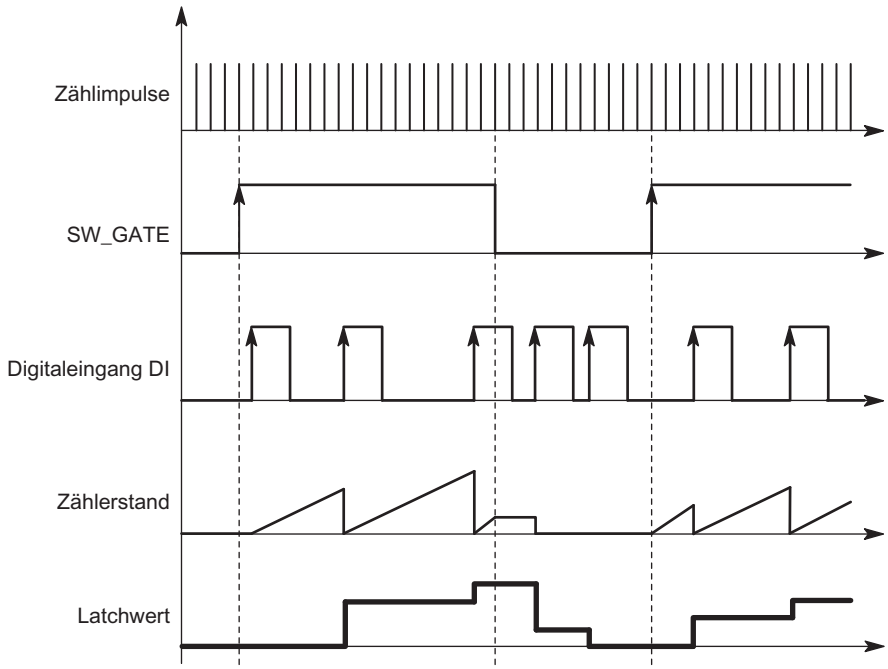

Zur Ausführung der Funktion muss die Zählbetriebsart mit dem SW-Tor freigegeben sein. Gestartet wird sie mit der ersten positiven Flanke am Digitaleingang.

In der Rückmeldeschnittstelle (Seite 380) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Status des Latch- und Retrigger-Signals.

Der Latchwert wird mit seinem RESET-Zustand ("0") vorbelegt. Er wird durch das Öffnen des SW-Tors nicht verändert.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Wenn Sie das SW-Tor schließen, wirkt es nur unterbrechend; d. h. wenn Sie das SW-Tor erneut öffnen, wird der Zählvorgang fortgesetzt. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Auch im taktsynchronen Betrieb wird der Zählvorgang mit jeder Flanke am Digitaleingang gelatcht und getriggert. In der Rückmeldeschnittstelle wird der Zählerstand angezeigt, der zum Zeitpunkt der letzten Flanke vor T_i gültig war.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count24V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 380) gelesen werden.

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die steigende Flanke eines Referenzsignals am Eingang zum Setzen des 1Count24V auf den Ladewert.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

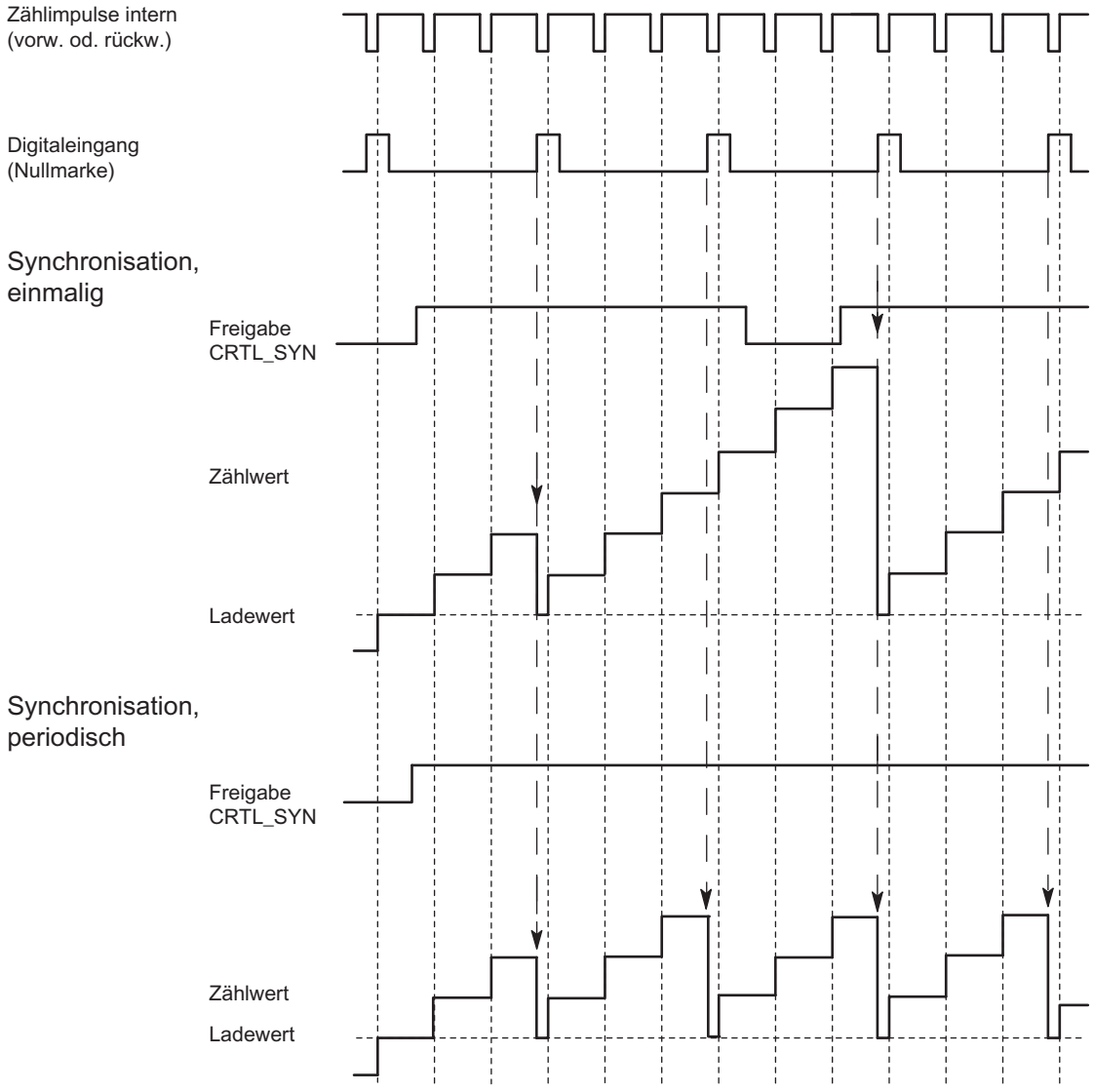

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 379) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.

- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count24V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count24V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 380) STS_SYN gesetzt. Es muss durch das Steuerbit (Seite 379) RES_STS zurückgesetzt werden.
- Als Referenzsignal kann das Signal eines prellfreien Schalters oder die Nullmarke eines Drehgebers dienen.
- Das Rückmeldebit (Seite 380) STS_DI zeigt den Pegel des Referenzsignals an.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 380) STS_SYN an, dass die steigende Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Latchen bei positiver Flanke

Latchen

Zählerstand und Latchwert werden mit ihren RESET-Zuständen ("0") vorbelegt:

Mit dem Öffnen des SW-Tors wird die Zählfunktion gestartet. Das 1Count24V beginnt beim Ladewert ("0").

Das folgende Bild zeigt Latchen mit Ladewert = 0.

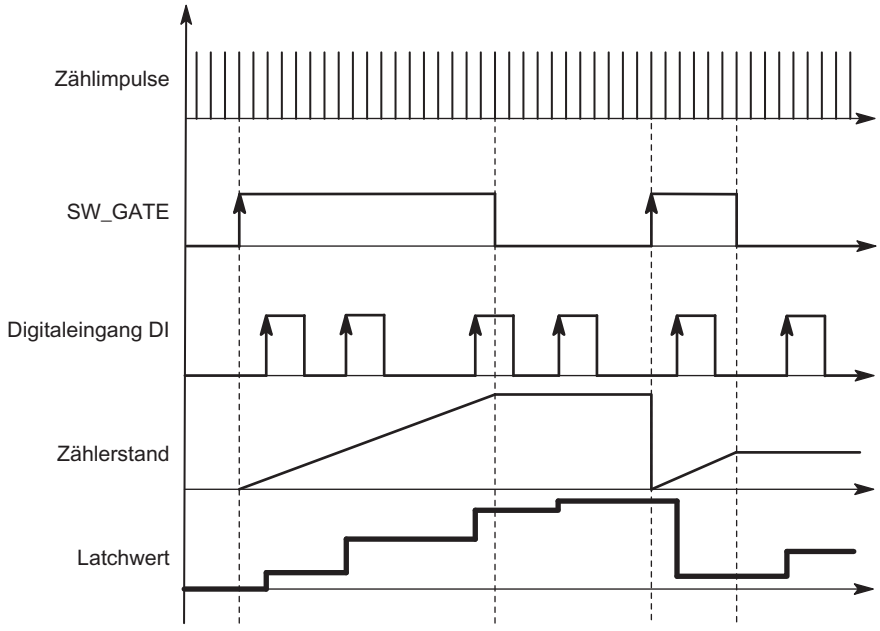

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der positiven Flanke am Digitaleingang DI.

In der Rückmeldeschnittstelle (Seite 380) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Pegel des Latch-Signals.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Im taktsynchronen Betrieb wird in der Rückmeldeschnittstelle der Zählerstand angezeigt, der zum Zeitpunkt der letzten positiven Flanke vor T_i gelatcht wurde.

Wenn Sie das SW-Tor schließen, wirkt es wie parametriert, abbrechend oder unterbrechend. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count24V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 380) gelesen werden.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 380) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Synchronisation auswählen

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" am Parameter "Funktion DI (Seite 374)" gewählt haben, dann können Sie hier zwischen einmaliger und periodischer Synchronisation wählen:

- Einmalig: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count24V mit dem Ladewert.
- Periodisch: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count24V mit dem Ladewert.

Steuer- und Rückmeldeschnittstelle für Wegerfassen

Steuerschnittstelle für Wegerfassen

Hinweis

Für das 1Count24V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count24V für Wegerfassen.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Ladewert direkt, vorbereitend		
Byte 4	Bit 7	Fehlerquittung	EXTF_ACK
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Anstoß Rücksetzen Statusbits	RES_STS
	Bit 1	Freigabe Synchronisation	CTRL_SYN
	Bit 0	Steuerbit SW-Tor	SW_GATE
Byte 5	Bit 7	reserviert = 0	
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	Zähler vorbereitend laden	LOAD_PREPARE
	Bit 0	Zähler direkt laden	LOAD_VAL
Byte 6 bis 7	reserviert = 0 [1]		

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
CTRL_SYN	Sie geben damit die Synchronisation frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Zähler vorbereitend laden Der Wert aus Byte 0 bis 3 wird als Ladewert übernommen.
LOAD_VAL	Der Wert aus Byte 0 bis 3 wird als neuer Zählwert direkt geladen.
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SW_GATE	Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für Wegerfassen

Hinweis

Für das 1Count24V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count24V für Wegerfassen.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Zählwert oder abgespeicherter Zählwert bei Latch-Funktion am Digitaleingang		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	reserviert = 0	
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD

Adresse	Belegung		Bezeichnung
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE
Byte 6	Bit 7	Nulldurchgang	STS_ND
	Bit 6	Untere Zählgrenze	STS_UFLW
	Bit 5	Obere Zählgrenze	STS_OFLW
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	Status Synchronisation	STS_SYN
Byte 7	reserviert = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 379) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 379) LOAD_VAL und LOAD_PREPARE dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_GATE	Status internes Tor: Es wird gezählt.
STS_LOAD	Ladefunktion läuft
STS_ND	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung. Das Bit muss durch das Steuerbit (Seite 379) RES_STS zurückgesetzt werden.

Bit	Erläuterung
STS_OFLW STS_UFLW	Obere Zählgrenze überschritten Untere Zählgrenze unterschritten Beide Bits müssen zurückgesetzt werden.
STS_SYN	Status Synchronisation: Nach erfolgreicher Synchronisation ist das STS_SYN-Bit gesetzt. Es muss durch das Steuerbit (Seite 379) RES_STS zurückgesetzt werden.

Parameter für Fast Mode

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count24V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count24V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO 1 abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.

[1] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count24V.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count24V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung

Zähl- und Richtungsauswertung auswählen

Signalauswertung A*, B*

Die Signalauswertung über A*, B* ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Impuls und Richtung (Seite 383)
  Bei 24 V-Impulsgebern mit Richtungspegel muss gewährleistet sein, dass zwischen Richtungssignal B* und Zählsignal A* eine Zeitspanne von mindestens 5 µs / 50 µs liegt, je nach parametriertem Eingangsfilter.
  Das folgende Bild zeigt die Zeitspanne zwischen Richtungssignal und Zählsignal.

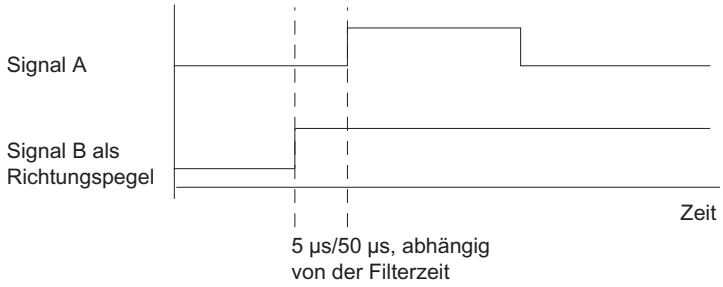

- Drehgeber (Seite 210)
  Wenn Sie einen 24 V-Drehgeber mit zwei um 90 Grad phasenversetzten Spuren am Zähl- und Richtungseingang anschließen, können Sie in allen Zählbetriebsarten eine Einfachauswertung (Seite 211) parametrieren. Alternativ können Sie Zweifach- (Seite 211) oder Vierfachauswertung (Seite 211) parametrieren.
  Bei allen Auswertungsarten können Sie die Zählrichtung (Seite 384) am Richtungseingang B* invertieren.
  An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie mit dem 1Count24V beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Hinweis

Die Angabe 100 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A* bzw. B* bezogen. Bei Zweifachauswertung ergibt sich somit maximal 200 kHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 400 kHz.

Auswertung "Impuls und Richtung"

Impuls und Richtung

Bei dieser Auswertungsart wird als Richtungsvorgabe der Pegel am Richtungseingang B* benutzt.

Ein unbeschalteter Eingang entspricht der Zählrichtung "vorwärts", wenn Sie beim Parameter Signalauswertung "Impuls und Richtung" gewählt haben.

Das folgende Bild zeigt die Signale eines 24 V-Impulsgebers mit Richtungspegel.

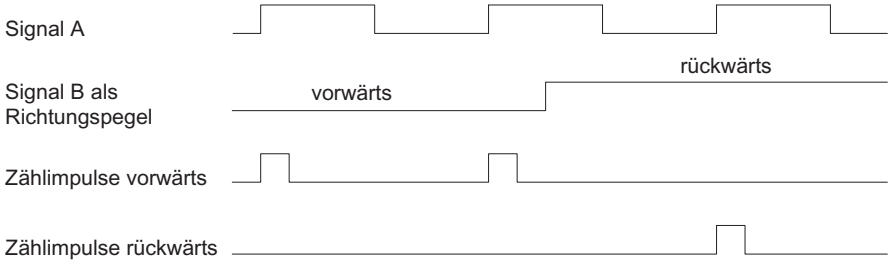

Filter für Eingangssignale auswählen

Eingangsfilter

Zur Filterung der Signale am Zähleingang A*, Richtungseingang B* und Digitaleingang DI können Sie entsprechend der Mindestimpulsdauer bzw. der maximalen Signalfrequenz einen Filter einschalten.

Sie können für jeden Eingang jeweils eine Mindestimpulsbreite von 2,5 µs bzw. 25 µs wählen.

Gebereingänge auswählen

Gebereingänge

An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden (P-Schalter/Gegentakt oder M-Schalter).

Hinweis

Wenn Sie beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205) , pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Synchronisation bei positiver Flanke (Seite 385)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.

Die Voreinstellung ist "Eingang".

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die positive Flanke eines Referenzsignals am Eingang zum Setzen des 1Count24V auf den Startwert (= Ladewert).

Das folgende Bild zeigt die Zählerzustände bei Synchronisation mit positiver Flanke.

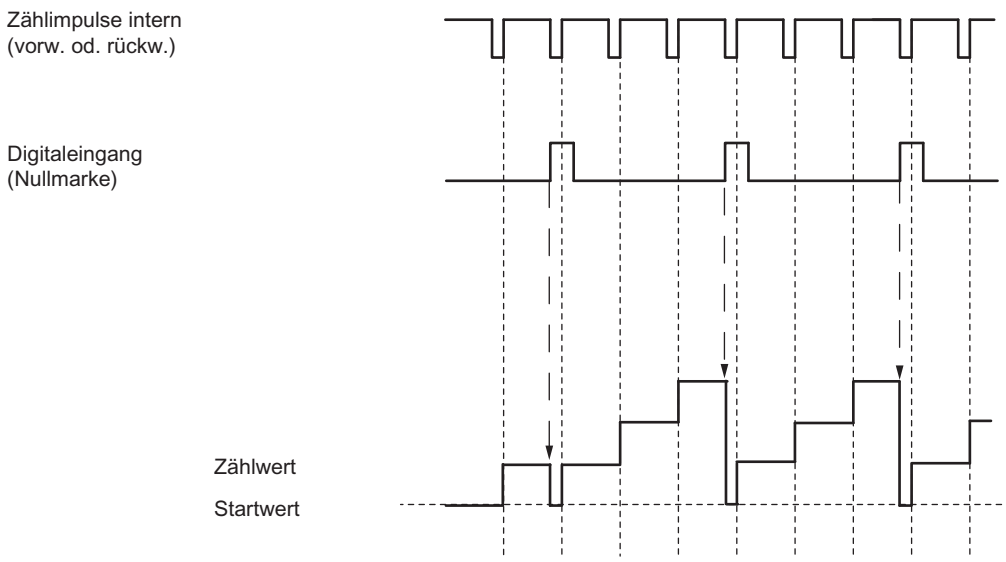

Es gibt folgende Bedingungen:

- Der Betriebsmodus "Fast Mode" muss aktiv (HW-Tor) sein.
  - Bei aktivierter Synchronisation lädt die erste und jede weitere Flanke das 1Count24V mit dem Startwert (= Ladewert).
- Als Referenzsignal kann das Signal eines prellfreien Schalters oder die Nullmarke eines Drehgebers dienen.
- Das Rückmeldebit STS_DI zeigt den Pegel des Referenzsignals an.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 387) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Ladewert festlegen

Ladewert

Sie können dem 1Count24V einen Ladewert vorgeben.

Dieser Ladewert wird direkt als Startwert übernommen.

Zulässiger Wertebereich: -16777216 bis +16777215

Rückmeldeschnittstelle für Fast Mode

Hinweis

Für das 1Count24V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count24V für Fast Mode.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Bit 31	Lebenszeichen	LZ
	Bit 30	Taktsynchronen Betrieb aufgenommen	STS_TIC
	Bit 29	Parametrierfehler	ERR_PARA
	Bit 28	Sammelfehler	EXTF
	Bit 27	Status DI	STS_DI
	Bit 26	Status Richtung vorwärts / rückwärts	STS_DIR
	Bit 25	Status internes Tor	STS_GATE
	Bit 0 bis 24	Zählwert	

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
EXTF	Sammelfehler Mögliche Ursache: • Kurzschluss Geberversorgung EXTF wird zurückgesetzt, wenn die Fehlerursachen beseitigt sind.
LZ	Das Lebenszeichen wird bei jeder Aktualisierung der Rückmeldeschnittstelle getoggelt, d. h., der zuletzt gesendete Wert wird invertiert.
STS_DI	Das Bit zeigt den Status des Digitaleingangs DI an.
STS_DIR	Status Richtung: • bei Geberwertänderung von größeren zu kleineren Geberpositionen (inkl. Nulldurchgang) → "1" • bei Geberwertänderung von kleineren zu größeren Geberpositionen (inkl. Nulldurchgang) → "0"
STS_GATE	Status internes Tor: Es wird gezählt.
STS_TIC	Der taktsynchrone Betrieb (falls parametriert) wurde aufgenommen.

Diagnose des 1Count24V

Diagnose durch LED-Anzeige

LED-Anzeige am 1Count24V

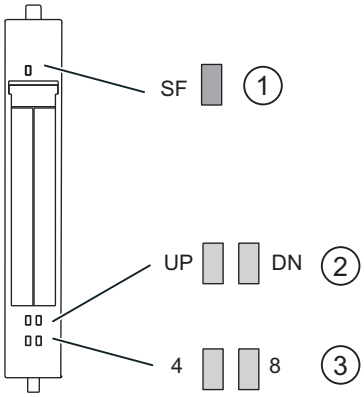

① Sammelfehler (rot)
② Statusanzeige Zählrichtung (grün)
③ Statusanzeige für Digitaleingang / Digitalausgang (grün)

Status- und Fehleranzeigen durch LEDs am 1Count24V

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 1Count24V.

Ereignis (LED)					Ursache	Maßnahme
SF	UP	DN	4	8		
ein					Keine Parametrierung oder falsches Modul gesteckt. Diagnosemeldung liegt vor.	Überprüfen Sie die Parametrierung. Werten Sie die Diagnose aus.
	ein				Status des niederwertigen Bits des Zählers, falls der Zähler vorwärts zählt.	
		ein			Status des invertierten niederwertigen Bits des Zählers, falls der Zähler rückwärts zählt.	
			ein		DO (direkte Ansteuerung, Vergleicherausgang) aktiviert.	
				ein	DI (HW-Tor, Synchronisation, Latch) aktiviert.	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

1Count24V Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 1Count24V.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Geberversorgung oder des Aktors.	Überprüfen Sie die Verdrahtung zum Geber. Korrektur der Prozessverdrahtung.
5_D	00101: Übertemperatur	Digitalausgang ist überlastet.	Korrektur der Prozessverdrahtung.
6_D	00110: Leitungsbruch	Leitung zum Aktor unterbrochen.	Korrektur der Prozessverdrahtung.
9_D	01001: Fehler	Interner Modulfehler ist aufgetreten	Austausch des Moduls.
		Lastspannung vom Powermodul zu niedrig.	Korrektur der Prozessverdrahtung. Überprüfen Sie die Lastspannung.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.

1.2.1.6 ET 200S 1Count5V einsetzen

Anschlussbelegung des 1Count5V

Verdrahtungsregeln

Die Leitungen (Klemmen 1 bis 8 sowie Klemmen 15 und 16) müssen geschirmt sein. Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung des 1Count5V

Die folgende Tabelle zeigt die Anschlussbelegung für das 1Count5V.

Ansicht	Anschlussbelegung	Bemerkungen
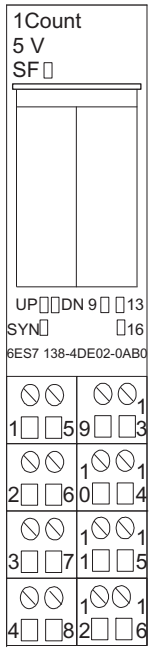	TM-E30S44-01 und 1Count5V 9 = DO1; 13 = DO2 10 = M; 14 = M 11 = DC24V; 15 = DC24V 16 = DI	A, /A: Spur A B, /B: Spur B N, /N: Spur N 24V: Lastspannung DC24V: Geberversorgung M: Masse DI: Digitaleingang DO1: Digitalausgang DO2: Digitalausgang

1Count5V konfigurieren und parametrieren

Betrieb

Betriebsmodus auswählen

Betriebsmodus

Mit der Vorgabe eines Betriebsmodus wählen Sie, mit welcher Funktionalität Sie das 1Count5V betreiben wollen.

Betriebsmodus	Beschreibung
Zählen (Seite 391)	Vorauswahl für die Zählbetriebsarten
Messen (Seite 415)	Vorauswahl für die Messbetriebsarten

Betriebsmodus	Beschreibung
Wegerfassen (Seite 431)	Das 1Count5V zählt in diesem Betriebsmodus ab dem Ladewert endlos: • Erreicht das 1Count5V beim Vorwärtszählen die obere Zählgrenze und kommt ein weiterer Zählimpuls, springt es auf die untere Zählgrenze und zählt von dort ohne Impulsverlust weiter. • Erreicht das 1Count5V beim Rückwärtszählen die untere Zählgrenze und kommt ein weiterer Zählimpuls, springt es auf die obere Zählgrenze und zählt ohne Impulsverlust weiter.
Fast Mode (Seite 444) (ab Firmware-Version V2.0 unterstützt)	Das 1Count5V zählt in diesem Betriebsmodus ab dem Startwert endlos: • Erreicht das 1Count5V beim Vorwärtszählen den mit 25 Bit maximal darstellbaren Wert (alle Bits des Zählers gesetzt) und kommt ein weiterer Zählimpuls, springt der Zählwert auf "0" und zählt von dort ohne Impulsverlust weiter. • Erreicht das 1Count5V beim Rückwärtszählen den Wert "0" und kommt ein weiterer Zählimpuls, springt der Zählwert auf den mit 25 Bit maximal darstellbaren Wert (alle Bits des Zählers gesetzt) und zählt ohne Impulsverlust weiter. Dieser Betriebsmodus eignet sich zum Wegerfassen in besonders kurzen taktsynchronen Zyklen.

Parameter für Zählen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count5V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count5V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.
DO Ersatzwert schalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der parametrierte Ersatzwert für den Digitalausgang wird durchgeschaltet. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen. Ist für den Ausgang "Impuls bei Erreichen des Vergleichswertes" parametriert, steht nur für die Impulsdauer der Ersatzwert = 1 an.	Die geänderten Parameter werden übernommen und wirksam.
DO letzten Wert halten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Zustand des Digitalausgangs wird beibehalten. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.

[1] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count5V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count5V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 411) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count5V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung auswählen

Signalauswertung A, B

Die Signalauswertung über A, B ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Drehgeber (Seite 210)
  Das 1Count5V kann die Flanken der Signale zählen. Im Normalfall werden nur die Flanken an A ausgewertet (Einfachauswertung). Um zu einer höheren Auflösung zu gelangen, können Sie bei der Parametrierung (Parameter "Signalauswertung") wählen, ob die Signale einfach (Seite 211), zweifach (Seite 211) oder vierfach (Seite 211) ausgewertet werden sollen.
  Mehrfachauswertung ist nur bei asymmetrischen Inkrementalgebern mit um 90 Grad versetzten Signalen A und B möglich.

  Hinweis

  Die Angabe 500 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A bzw. B bezogen. Bei Zweifachauswertung ergibt sich somit max. 1 MHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 2 MHz.

Diagnose der Signalpaare aktivieren

Überwachung der Signalpaare

Die Signalpaare A und /A, B und /B, N und /N können überwacht werden.

Aktivieren Sie dazu das entsprechende Optionskästchen.

Hinweis

Die "Diagnose N, /N" können Sie erst aktivieren, wenn Sie zuvor die "Diagnose A, /A und B, /B" aktiviert haben.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Verhalten der Ausgänge bei den Zählbetriebsarten

Vergleichswerte und Ausgänge

Einleitung

Sie können auf dem 1Count5V zwei Vergleichswerte ablegen, die den Digitalausgängen zugeordnet sind. Abhängig von Zählerstand und Vergleichswerten können die Ausgänge aktiviert werden.

Digitalausgänge

Das 1Count5V hat zwei Digitalausgänge.

Beide Ausgänge sind parametrierbar (Parameter "Funktion DO1" und "Funktion DO2").

Sie können die Funktion und das Verhalten der Digitalausgänge im Betrieb ändern.

Steuern der Ausgänge gleichzeitig zu den Vergleichern

Haben Sie für die Ausgänge eine Vergleichsfunktion gewählt, können Sie die Ausgänge mit SET_DO1 oder SET_DO2 weiterhin steuern. Damit haben Sie die Möglichkeit, die Wirkung der Vergleichsfunktionen über Ihr Steuerungsprogramm zu simulieren:

- Mit der positiven Flanke von SET_DO1 oder SET_DO2 wird der Ausgang gesetzt.
  Ist die Funktion "Impuls bei Erreichen des Vergleichswertes" gewählt, wird aber nur ein Impuls der vorgegebenen Dauer ausgegeben. Bei Impulsdauer = 0 kann der Ausgang mit SET_DO1 oder SET_DO2 gesetzt werden, solange sich der Zählwert auf dem Vergleichswert befindet bzw. die Hysterese aktiv ist.
  Beim Ausgangsverhalten "Schalten an Vergleichswerten" ist das Steuerbit SET_DO1 nicht erlaubt.
- Eine negative Flanke von SET_DO1 oder SET_DO2 setzt den Ausgang wieder zurück.

Beachten Sie, dass die Vergleicher weiterhin aktiv sind und den Ausgang bei Änderung des Vergleichsergebnisses setzen bzw. rücksetzen können.

Hinweis

Ein durch SET_DO1 oder SET_DO2 gesetzter Ausgang wird am Vergleichswert (durch den Vergleicher) nicht zurückgesetzt.

Vergleichswerte laden

Die Vergleichswerte übergeben Sie an das 1Count5V. Der Zählvorgang wird davon nicht beeinflusst.

Gültiger Wertebereich für die zwei Vergleichswerte

Hauptzählrichtung: keine	Hauptzählrichtung: vorwärts	Hauptzählrichtung: rückwärts
Untere Zählgrenze bis Obere Zählgrenze	-2147483648 bis Obere Zählgrenze -1	1 bis 2147483647

Funktion und Verhalten der Digitalausgänge ändern

Über die Steuerschnittstelle (Seite 411) können Sie Funktionen und Verhalten der Ausgänge im Betrieb ändern. Das 1Count5V löscht dabei die Ausgänge und übernimmt die Werte folgendermaßen:

- **Funktion der Digitalausgänge DO1 und DO2:** Wenn Sie die Funktion so ändern, dass die Vergleichsbedingung erfüllt ist, wird der Ausgang erst nach dem nächsten Zählimpuls gesetzt. Ist jedoch die Hysterese (Seite 399) aktiv, führt das 1Count5V keine Änderung am Ausgang durch.
- **Hysterese:** Eine aktive Hysterese (Seite 399) bleibt nach der Änderung aktiv. Der neue Hysteresebereich wird beim nächsten Erreichen des Vergleichswertes übernommen.
- **Impulsdauer:** Die neue Impulsdauer (Seite 402) wird mit dem nächsten Impuls wirksam.

Verhalten der Ausgänge auswählen

Verhalten der Ausgänge DO1 und DO2

Wählen Sie in der Klappliste das gewünschte Verhalten des jeweiligen Ausgangs:

- Ausgang (Seite 396)
- Einschalten bei Zählerstand ≥ Vergleichswert (Seite 396)
- Einschalten bei Zählerstand ≤ Vergleichswert (Seite 396)
- Impuls bei Erreichen des Vergleichswertes (Seite 397)
- Schalten an Vergleichswerten (Seite 397) (nur DO1)

Funktion und Verhalten des Digitalausgangs DO1 einstellen oder ändern

Wenn Sie das Verhalten des DO1 einstellen oder ändern, müssen Sie alle parametrierten Abhängigkeiten berücksichtigen, sonst erzeugen Sie einen Parametrierfehler oder einen Fehler beim Laden.

Randbedingungen:

Wenn Sie für DO1 "Schalten an Vergleichswerten" parametrieren, dann müssen Sie

- die Hysterese = 0 setzen und
- zusätzlich die "Funktion DO2" als "Ausgang" parametrieren.

Ausgang

Ausgang

Mit den Steuerbits (Seite 411) SET_DO1 und SET_DO2 schalten Sie die Ausgänge ein und aus.

Dazu müssen die Steuerbits (Seite 411) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Mit den Statusbits STS_DO1 und STS_DO2 in der Rückmeldeschnittstelle (Seite 413) können Sie den Status der Ausgänge abfragen.

Die Statusbits (Seite 413) STS_CMP1 und STS_CMP2 zeigen, dass der jeweilige Ausgang eingeschaltet ist oder eingeschaltet war. Diese Statusbits behalten so lange ihren Zustand, bis sie quittiert werden. Wenn der Ausgang noch geschaltet ist, wird das entsprechende Bit sofort wieder gesetzt. Diese Statusbits werden auch gesetzt, wenn bei nicht freigegebenem DO1 oder DO2 die Steuerbits SET_DO1 oder SET_DO2 betätigt werden.

Taktsynchroner Betrieb

Im taktsynchronen Betrieb schalten die Ausgänge DO1 und DO2 zum Zeitpunkt T_o.

Einschalten bei Zählerstand ≤ Vergleichswert bzw. Zählerstand ≥ Vergleichswert

Zählerstand ≤ Vergleichswert und Zählerstand ≥ Vergleichswert

Wenn die Vergleichsbedingungen erfüllt sind, schaltet der jeweilige Vergleicher den Ausgang ein. Der Zustand des Ausgangs (Seite 413) wird mit STS_DO1 und STS_DO2 angezeigt.

Dazu müssen die Steuerbits (Seite 411) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Das Vergleichsergebnis wird mit den Statusbits (Seite 413) STS_CMP1 bzw. STS_CMP2 angezeigt. Quittieren und somit rücksetzen können Sie diese Bits erst, wenn die Vergleichsbedingungen nicht mehr erfüllt sind.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schalten die Ausgänge DO1 und DO2 unmittelbar bei Erfüllung der Vergleichsbedingung und sind daher unabhängig vom Buszyklus.

Impuls bei Erreichen des Vergleichswertes

Impuls bei Erreichen des Vergleichswertes

Erreicht der Zählerstand den Vergleichswert, dann schaltet der Vergleicher den jeweiligen Digitalausgang für die parametrierte Impulsdauer ein.

Dazu muss das Steuerbit (Seite 411) CTRL_DO1 bzw. CTRL_DO2 gesetzt sein.

Die Statusbits (Seite 413) STS_DO1 und STS_DO2 haben immer den Zustand des jeweiligen Digitalausgangs.

Das Vergleichsergebnis wird mit dem Statusbit (Seite 413) STS_CMP1 bzw. STS_CMP2 angezeigt und ist durch Quittieren erst rücksetzbar, wenn die Impulsdauer abgelaufen ist.

Ist eine Hauptzählrichtung parametriert, dann schaltet der Vergleicher nur beim Erreichen des Vergleichswerts in Hauptzählrichtung.

Ist keine Hauptzählrichtung parametriert, dann schaltet der Vergleicher beim Erreichen des Vergleichswerts aus beiden Richtungen.

War der Digitalausgang durch das Steuerbit (Seite 411) SET_DO1 bzw. SET_DO2 gesetzt, wird er nach Ablauf der Impulsdauer zurückgesetzt.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schalten die Ausgänge DO1 und DO2 unmittelbar bei Erfüllung der Vergleichsbedingung und sind daher unabhängig vom Buszyklus.

Schalten an Vergleichswerten

Schalten an Vergleichswerten

Der Vergleicher schaltet den Ausgang, wenn die folgenden Bedingungen erfüllt sind:

- Die beiden Vergleichswerte müssen mit der Ladefunktion (Seite 411) CMP_VAL1 und CMP_VAL2 geladen sein und
- nach dem Laden der Vergleichswerte müssen Sie den Ausgang DO1 (Seite 411) mit CRTL_DO1 freigeben.

Die folgende Tabelle zeigt, wann der DO1 eingeschaltet bzw. ausgeschaltet ist:

	DO1 ist eingeschaltet bei	DO1 ist ausgeschaltet bei
V2 < V1 (siehe Bild unten)	V2 ≤ Zählerstand ≤ V1	V2 > Zählerstand oder Zählerstand > V1
V2 = V1	V2 = Zählerstand = V1	V2 ≠ Zählerstand ≠ V1
V2 > V1 (siehe Bild unten)	V1 > Zählerstand oder Zählerstand > V2	V1 ≤ Zählerstand ≤ V2

Die folgenden Bilder zeigen den Zustand des Ausgangs DO1 in Abhängigkeit vom Zählerstand und den Vergleichswerten:

- Beim Start des Zählvorgangs ist V2 < V1

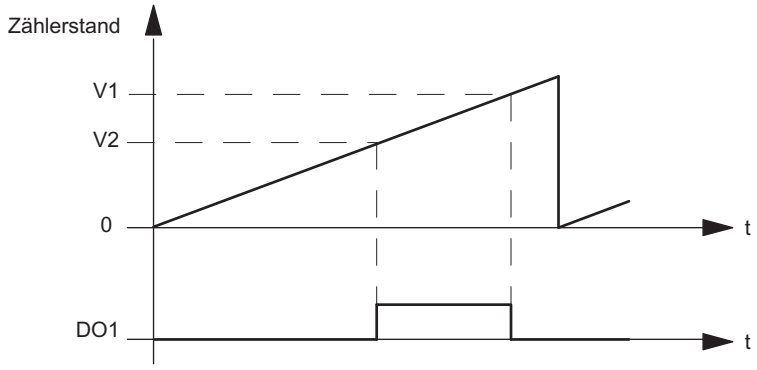

- Beim Start des Zählvorgangs ist V2 > V1

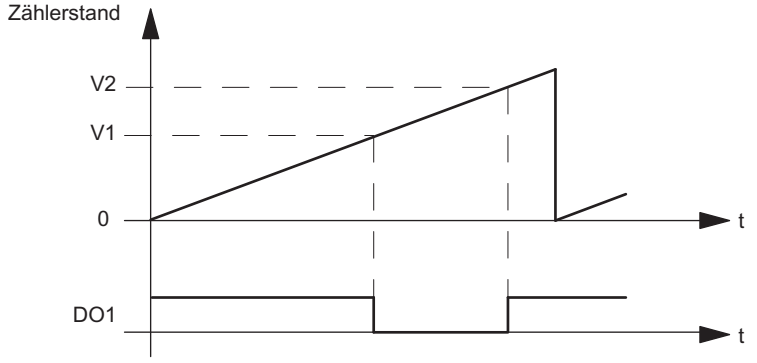

Das Vergleichsergebnis wird mit dem Statusbit (Seite 413) STS_CMP1 angezeigt. Quittieren und somit rücksetzen können Sie dieses Bit erst, wenn die Vergleichsbedingung nicht mehr erfüllt ist.

Bei diesem Ausgangsverhalten gibt es keine Hysterese.

Ein Steuern des Ausgangs DO1 durch das Steuerbit (Seite 411) SET_DO1 ist bei diesem Ausgangsverhalten nicht möglich.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schaltet der Ausgang DO1 unmittelbar bei Erfüllung der Vergleichsbedingung und ist daher unabhängig vom Buszyklus.

Ersatzwert DOx auswählen

Ersatzwert DO1 und Ersatzwert DO2

Wenn Sie unter "Verhalten bei CPU/Master-STOP (Seite 392)" den Parameter "DO Ersatzwert schalten" ausgewählt haben, dann wählen Sie jeweils den Ersatzwert aus, der am Digitalausgang DO1 bzw. DO2 ausgegeben werden soll.

Diagnose DOx aktivieren

Diagnose DO1 und Diagnose DO2

"Diagnose DO1" bzw. "Diagnose DO2" (Leitungsbruch, Kurzschluss) ist nur bei Impulslängen > 90 ms am Digitalausgang DO1 bzw. DO2 möglich.

Aktivieren Sie das jeweilige Optionskästchen für die Diagnose am Ausgang DO1 bzw. DO2.

Hysterese festlegen

Hysterese

Ein Geber kann an einer bestimmten Position stehen bleiben und dann um diese Position "pendeln". Dieser Zustand führt dazu, dass der Zählerstand um einen bestimmten Wert herum schwankt. Liegt nun in diesem Schwankungsbereich zum Beispiel ein Vergleichswert, würde der zugehörige Ausgang im Rhythmus dieser Schwankungen ein- und ausgeschaltet werden. Um dieses Schalten bei kleinen Schwankungen zu verhindern, ist das 1Count5V mit einer parametrierbaren Hysterese ausgestattet. Sie können einen Bereich zwischen 0 und 255 (0 bedeutet: Hysterese abgeschaltet) parametrieren.

Die Hysterese wirkt auch bei Überlauf und Unterlauf.

Wirkungsweise bei Zählerstand ≤ Vergleichswert und Zählerstand ≥ Vergleichswert

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Hauptzählrichtung vorwärts" und Ausgang "Einschalten bei Zählerstand ≥ Vergleichswert" parametriert.

Mit dem Erreichen der Vergleichsbedingung wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv. Der Vergleicher schaltet wieder entsprechend seiner Vergleichsbedingungen.

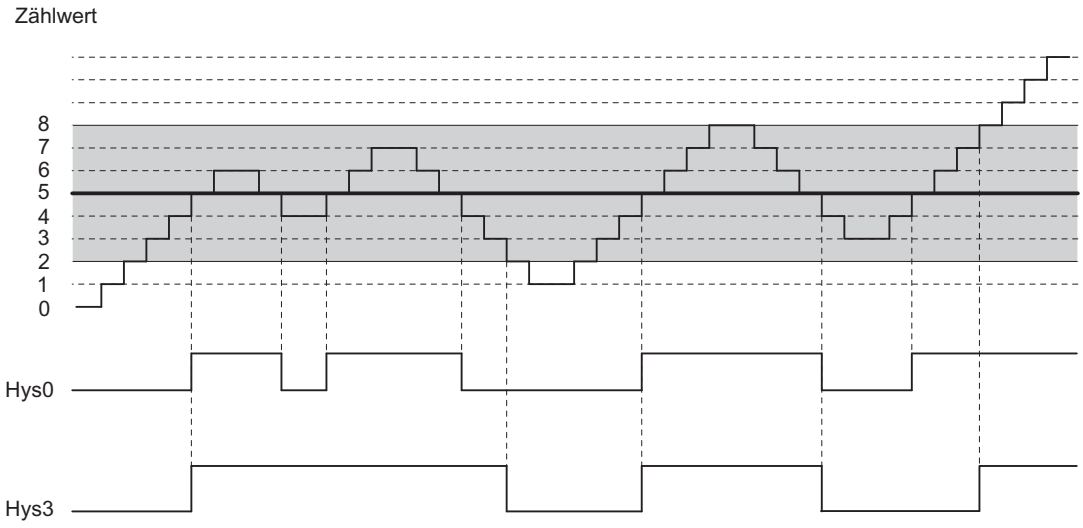

Hinweis

Bei einem Wechsel der Zählrichtung auf dem Vergleichswert bei aktiver Hysterese wird der Ausgang zurückgesetzt.

Wirkungsweise bei Vergleichswert erreicht und Impulsdauer = 0

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Impuls bei Erreichen des Vergleichswertes", "keine Hauptzählrichtung" und "Impulsdauer = 0" parametriert.

Mit dem Erreichen der Vergleichsbedingungen wird die Hysterese aktiv. Bei aktiver Hysterese bleibt das Vergleichsergebnis unverändert.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv. Der Vergleicher löscht das Vergleichsergebnis.

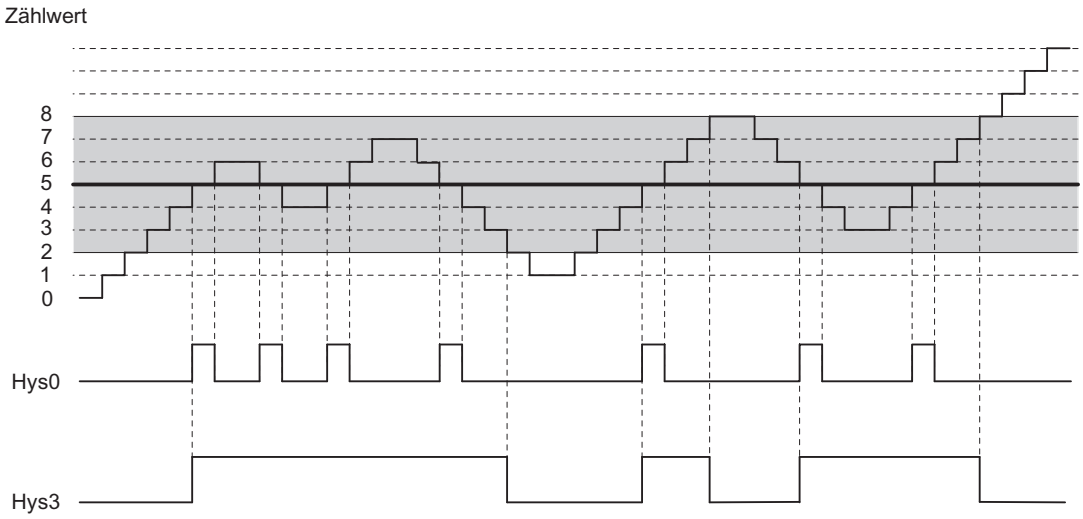

Wirkungsweise bei Vergleichswert erreicht, Impulsdauer ausgeben

Das folgende Bild zeigt ein Beispiel für die Wirkung der Hysterese. Im Bild ist das unterschiedliche Verhalten eines Ausgangs bei einer parametrierten Hysterese von 0 (= abgeschaltet) und bei einer Hysterese von 3 dargestellt. Im Beispiel ist der Vergleichswert = 5.

Der Zähler ist mit den Einstellungen "Impuls bei Erreichen des Vergleichswertes", "keine Hauptzählrichtung" und "Impulsdauer > 0" parametriert.

Mit dem Erreichen der Vergleichsbedingungen wird die Hysterese aktiv und ein Impuls der parametrierten Dauer ausgegeben.

Verlässt der Zählwert den Hysteresebereich, ist die Hysterese nicht mehr aktiv.

Wird die Hysterese aktiv, speichert das 1Count5V die Zählrichtung.

Wird der Hysteresebereich entgegen der gespeicherten Richtung verlassen, wird ein Impuls ausgegeben.

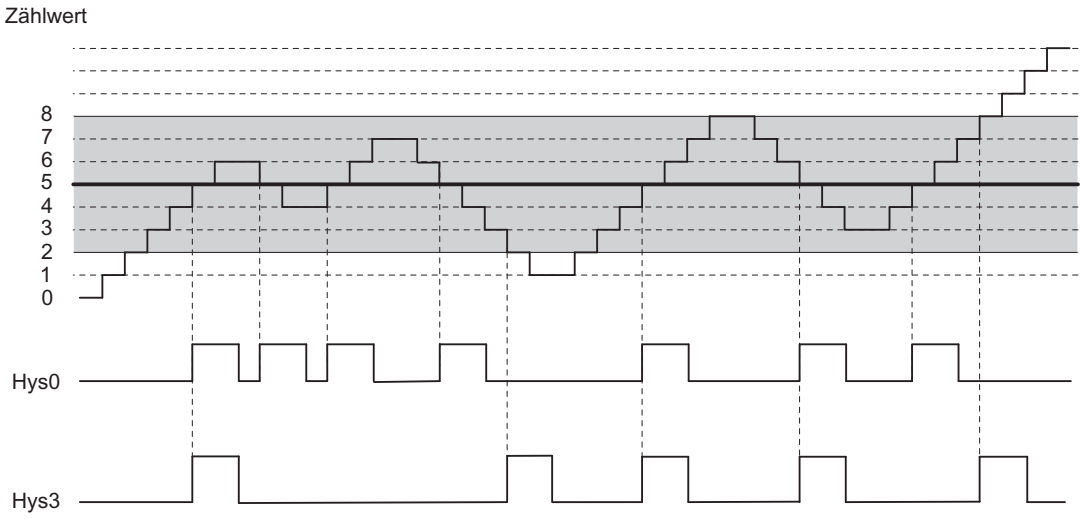

Impulsdauer festlegen

Impulsdauer

Die Impulsdauer beginnt mit dem Setzen des jeweiligen Digitalausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 2 ms.

Zur Anpassung an die verwendeten Aktoren kann die Impulsdauer vorgegeben werden. Die Impulsdauer gibt an, wie lange der Ausgang gesetzt werden soll. Die Impulsdauer kann in Schritten zu 2 ms zwischen 0 und 510 ms vorgewählt werden.

Wenn die Impulsdauer = 0 ist, wird der Ausgang so lange gesetzt, bis die Vergleichsbedingung nicht mehr erfüllt ist. Beachten Sie, dass die Zählimpulszeiten größer sein müssen als die minimalen Schaltzeiten des Digitalausgangs.

Taktsynchroner Betrieb

Auch im taktsynchronen Betrieb schalten die Ausgänge DO1 und DO2 unmittelbar bei Erfüllung der Vergleichsbedingung und sind daher unabhängig vom Buszyklus.

Zählbetriebsart auswählen

Beschreibung

Mit der Vorgabe einer Zählbetriebsart wählen Sie, mit welcher Funktionalität Sie das 1Count5V betreiben wollen.

Betriebsart	Beschreibung
Endlos Zählen (Seite 185) (mit oder ohne Tor)	Das 1Count5V zählt ab dem aktuellen Zählerstand endlos.
Einmaliges Zählen (Seite 186) (mit HW- oder SW-Tor)	Das 1Count5V zählt mit dem Öffnen des Tores ab dem Ladewert bis zur Zählgrenze.
Periodisches Zählen (Seite 187) (mit HW- oder SW-Tor)	Das 1Count5V zählt mit dem Öffnen des Tores ab dem Ladewert zwischen den Zählgrenzen.

Die Voreinstellung ist Betriebsart "Endlos Zählen".

Hauptzählrichtung auswählen

Beschreibung

Mit der Parametrierung einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie durch Parametrieren einer oberen Zählgrenze (Seite 410) den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Eine Hauptzählrichtung können Sie nur in den Betriebsarten "Einmaliges Zählen" und "Periodisches Zählen" auswählen.

Sie können folgendes Verhalten auswählen:

- Keine (ohne Hauptzählrichtung)
- Vorwärts
- Rückwärts

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Latch/Retrigger bei positiver Flanke (Seite 404)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt.
- Synchronisation bei positiver Flanke (Seite 406)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.
- Latchen bei positiver Flanke (Seite 407)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht).
- HW-Freigabe für Synchronisation (Seite 408)
  Der Digitaleingang dient als HW-Freigabe. Bei aktiver HW-Freigabe wird mit der Nullmarke des Gebers der Zähler auf den Ladewert gesetzt.

Die Voreinstellung ist "Eingang".

Latch/Retrigger bei positiver Flanke

Latch/Retrigger

Mit dieser Funktion speichern Sie bei einer positiven Flanke am Digitaleingang den aktuellen internen Zählerstand des 1Count5V und retriggern den Zählvorgang. Dies bedeutet, dass der aktuelle interne Zählerstand zum Zeitpunkt der positiven Flanke abgespeichert (Latchwert) und dann das 1Count5V wieder mit dem Ladewert geladen wird und von dort weiterzählt.

Das folgende Bild zeigt Latch/Retrigger mit dem Ladewert = 0.

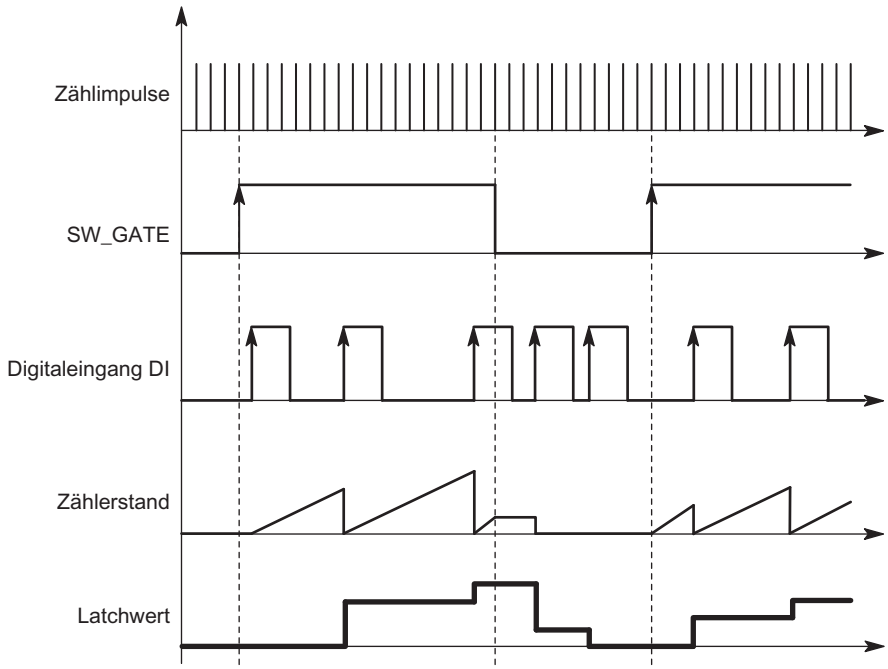

Zur Ausführung der Funktion muss die Zählbetriebsart mit dem SW-Tor freigegeben sein. Gestartet wird sie mit der ersten positiven Flanke am Digitaleingang.

In der Rückmeldeschnittstelle (Seite 413) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Status des Latch- und Retrigger-Signals.

Der Latchwert wird mit seinem RESET-Zustand vorbelegt:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Er wird durch das Öffnen des SW-Tors nicht verändert.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Wenn Sie das SW-Tor schließen, wirkt es nur unterbrechend; d. h. wenn Sie das SW-Tor erneut öffnen, wird der Zählvorgang fortgesetzt. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Auch im taktsynchronen Betrieb wird der Zählvorgang mit jeder Flanke am Digitaleingang gelatcht und getriggert. In der Rückmeldeschnittstelle wird der Zählerstand angezeigt, der zum Zeitpunkt der letzten Flanke vor T_i gültig war.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count5V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 413) gelesen werden.

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die steigende Flanke eines Referenzsignals am Eingang zum Setzen des 1Count5V auf den Ladewert.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

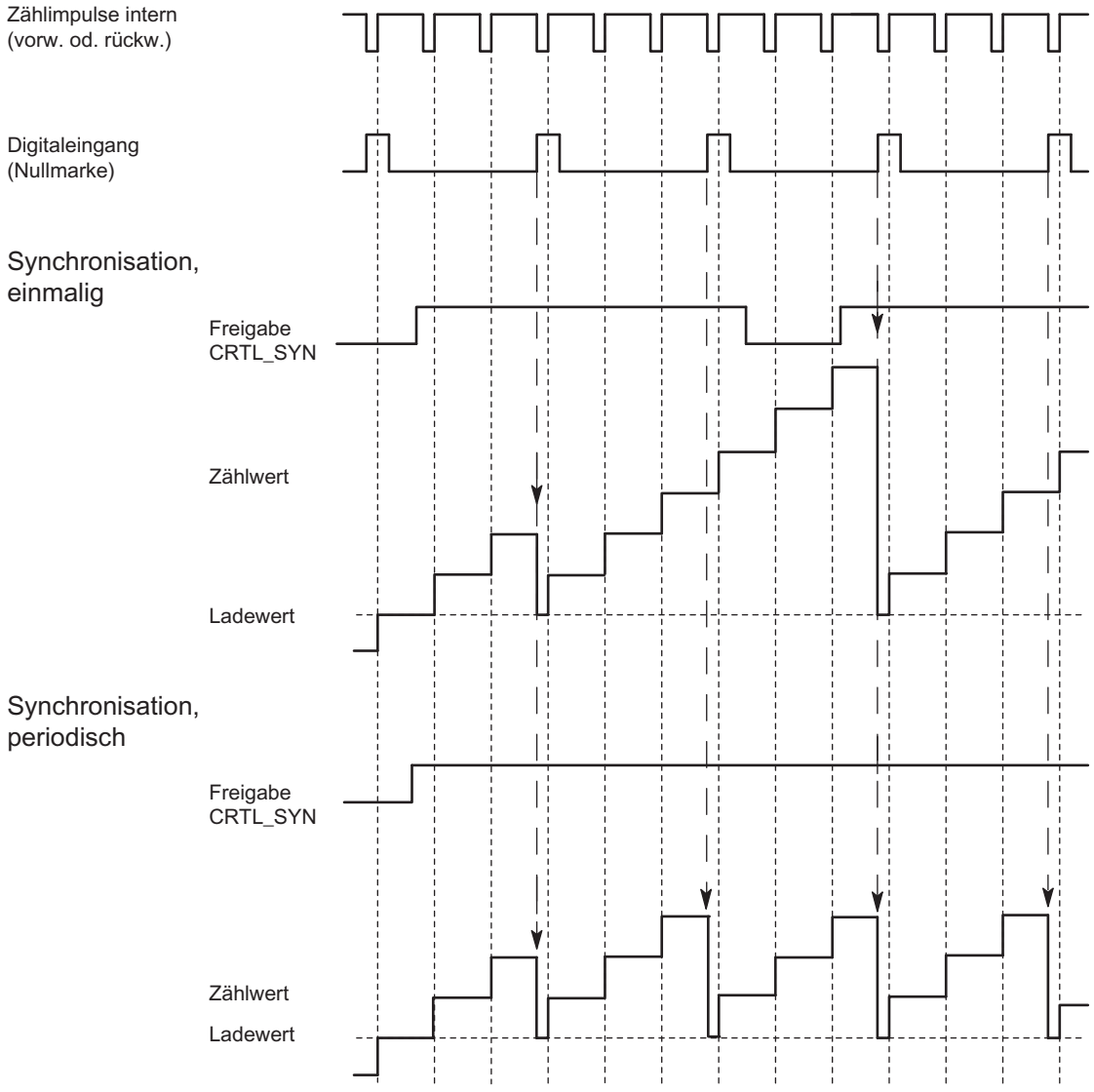

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 411) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.

- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count5V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count5V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 413) STS_SYN gesetzt und die LED SYN leuchtet. Durch das Steuerbit (Seite 411) RES_STS wird das Rückmeldebit zurückgesetzt und die LED ausgeschaltet.
- Als Referenzsignal kann das Signal eines prellfreien Schalters dienen.
- Das Rückmeldebit (Seite 413) STS_DI zeigt den Pegel des Referenzsignals an.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 413) STS_SYN an, dass die steigende Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Latchen bei positiver Flanke

Latchen

Zählerstand und Latchwert werden mit ihren RESET-Zuständen vorbelegt:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Mit dem Öffnen des SW-Tors wird die Zählfunktion gestartet. Das 1Count5V beginnt beim Ladewert:

- "0" bei Hauptzählrichtung "Keine" und "Vorwärts"
- "Parametrierte obere Zählgrenze" bei Hauptzählrichtung "Rückwärts"

Das folgende Bild zeigt Latchen mit Ladewert = 0.

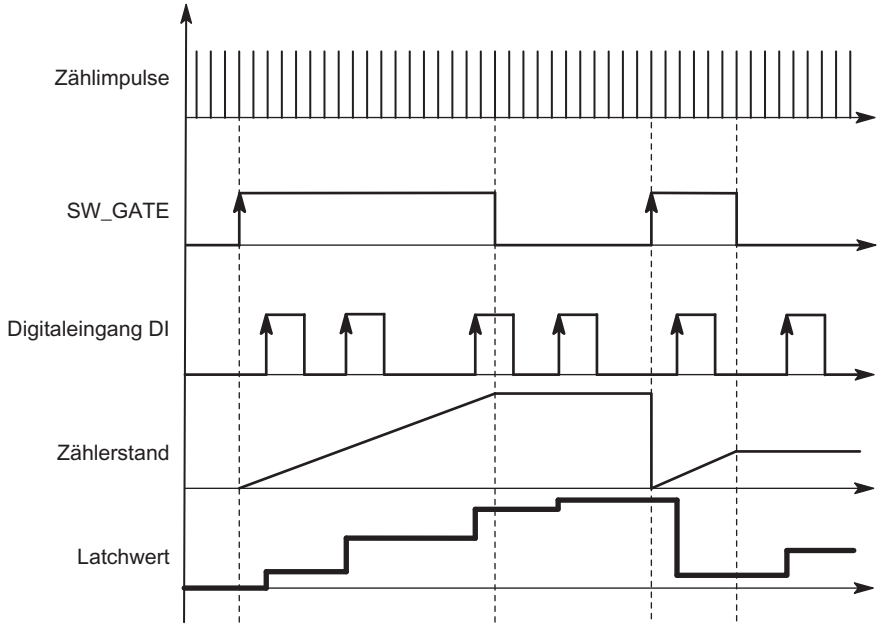

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der positiven Flanke am Digitaleingang DI.

In der Rückmeldeschnittstelle (Seite 413) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Pegel des Latch-Signals.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Im taktsynchronen Betrieb wird in der Rückmeldeschnittstelle der Zählerstand angezeigt, der zum Zeitpunkt der letzten positiven Flanke vor T_i gelatcht wurde.

Wenn Sie das SW-Tor schließen, wirkt es wie parametriert, abbrechend oder unterbrechend. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count5V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 413) gelesen werden.

HW-Freigabe für Synchronisation

Synchronisation mit DI und Nullmarke

Wenn Sie "HW-Freigabe für Synchronisation" parametriert haben, dient der DI als HW-Freigabe. Wenn die HW-Freigabe aktiv ist, wird das 1Count5V durch die Nullmarke des Gebers mit dem Ladewert geladen.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation (Seite 410)") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

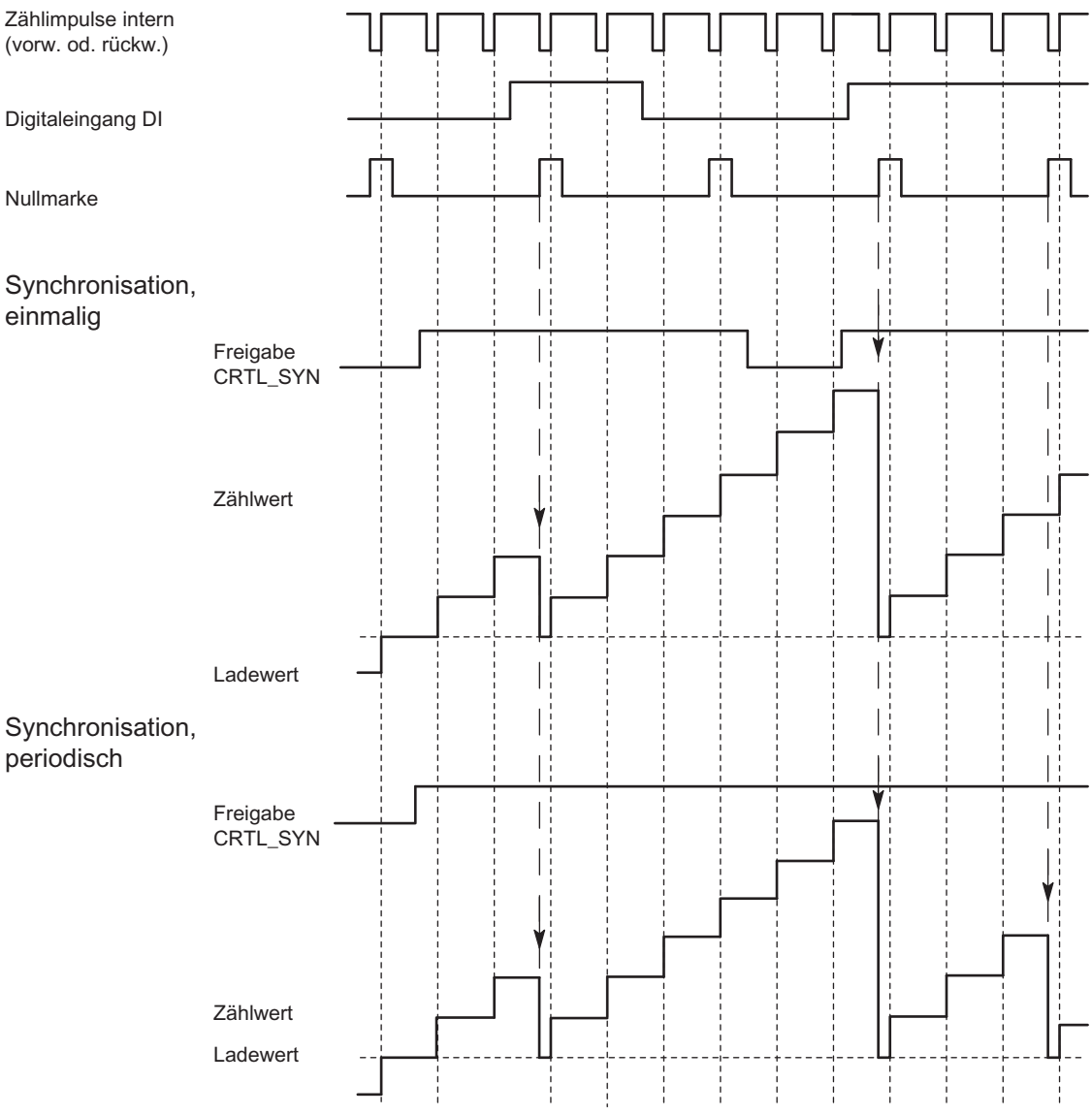

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 411) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.
- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist und die HW-Freigabe vorliegt, lädt die erste Nullmarke das 1Count5V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist und die HW-Freigabe vorliegt, lädt die erste und jede weitere Nullmarke das 1Count5V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 413) STS_SYN gesetzt und die LED SYN leuchtet. Durch das Steuerbit (Seite 411) RES_STS wird das Rückmeldebit zurückgesetzt und die LED ausgeschaltet.
- Als Referenzsignal zur HW-Freigabe kann das Signal eines prellfreien Schalters dienen.
- Das Rückmeldebit (Seite 413) STS_DI zeigt den Pegel des Referenzsignals.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 413) STS_SYN an, dass die positive Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 413) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Synchronisation auswählen

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" am Parameter "Funktion DI (Seite 404)" gewählt haben, dann können Sie hier zwischen einmaliger und periodischer Synchronisation wählen:

- Einmalig: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count5V mit dem Ladewert.
- Periodisch: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count5V mit dem Ladewert.

Obere Zählgrenze festlegen

Beschreibung

Mit der Parametrierung einer oberen Zählgrenze in Verbindung mit einer Hauptzählrichtung (Seite 188) (vorwärts oder rückwärts) grenzen Sie den maximalen Zählbereich auf einen kleineren Zählbereich ein. Der parametrierte Zählbereich liegt dann zwischen 0 und der parametrierten oberen Zählgrenze.

Die obere Zählgrenze kann einen Wert > 1 bis zur oberen Zählbereichsgrenze annehmen.

Steuer- und Rückmeldeschnittstelle für die Zählbetriebsarten

Steuerschnittstelle für die Zählbetriebsarten

Hinweis

Für das 1Count5V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count5V für die Zählbetriebsarten.

Adresse		Belegung			
Byte 0 bis 3		**Ladewert direkt, vorbereitend, Vergleichswert 1 oder 2**			
	Byte 0	**Verhalten von DO1, DO2 des 1Count5V**			
		Bit 2	Bit 1	Bit 0	Funktion DO1
		0	0	0	Ausgang
		0	0	1	Einschalten bei Zählerstand ≥ Vergleichswert
		0	1	0	Einschalten bei Zählerstand ≤ Vergleichswert
		0	1	1	Impuls beim Erreichen des Vergleichswertes
		1	0	0	Schalten an Vergleichswerten
		1	0	1	gesperrt
		1	1	0	gesperrt
		1	1	1	gesperrt
			Bit 5	Bit 4	Funktion DO2
			0	0	Ausgang
			0	1	Einschalten bei Zählerstand ≥ Vergleichswert
			1	0	Einschalten bei Zählerstand ≤ Vergleichswert
			1	1	Impuls beim Erreichen des Vergleichswertes
		Bit 3, 6 und 7: Reserve = 0			
	Byte 1 bis 3	Byte 1	Hysterese DO1, DO2 (Wertebereich 0 … 255)		
		Byte 2	Impulsdauer DO1, DO2 (Wertebereich 0 … 255)		
		Byte 3	reserviert = 0		
					Bezeichnung

Adresse	Belegung		
Byte 4	Bit 7	Fehlerquittung	EXTF_ACK
	Bit 6	Freigabe DO2	CTRL_DO2
	Bit 5	Steuerbit DO2	SET_DO2
	Bit 4	Freigabe DO1	CTRL_DO1
	Bit 3	Steuerbit DO1	SET_DO1
	Bit 2	Anstoß Rücksetzen Statusbits	RES_STS
	Bit 1	Freigabe Synchronisation	CTRL_SYN
	Bit 0	Steuerbit SW-Tor	SW_GATE
Byte 5	Bit 7	reserviert = 0	
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	Funktion und Verhalten von DO1, DO2 ändern	C_DOPARAM
	Bit 3	Vergleichswert 2 laden	CMP_VAL2
	Bit 2	Vergleichswert 1 laden	CMP_VAL1
	Bit 1	Zähler vorbereitend laden	LOAD_PREPARE
	Bit 0	Zähler direkt laden	LOAD_VAL
Byte 6 bis 7	reserviert = 0 [1]		

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
C_DOPARAM	Funktion und Verhalten von DO1, DO2 ändern Die Werte aus Byte 0 bis 2 werden als neue Funktion, Hysterese und Impulsdauer von DO1, DO2 übernommen. Dabei ist folgender Fehler möglich: Die Bedingungen für das Verhalten "Schalten an Vergleichswerten" sind nicht erfüllt.
CMP_VAL1	Vergleichswert 1 laden Der Wert aus Byte 0 bis 3 wird mit dem Steuerbit "Vergleichswert 1 laden" (CMP_VAL1) in den Vergleichswert 1 übertragen.
CMP_VAL2	Vergleichswert 2 laden Der Wert aus Byte 0 bis 3 wird mit dem Steuerbit "Vergleichswert 2 laden" (CMP_VAL2) in den Vergleichswert 2 übertragen.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
CTRL_DO2	Freigabe DO2 Mit diesem Bit geben Sie den Ausgang DO2 frei.
CTRL_SYN	Sie geben damit die Synchronisation frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Zähler vorbereitend laden Der Wert aus Byte 0 bis 3 wird als Ladewert übernommen.
LOAD_VAL	Der Wert aus Byte 0 bis 3 wird als neuer Zählwert direkt geladen

Bit	Erläuterung
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SET_DO1	Steuerbit DO1 Digitalausgang DO1 ein- und ausschalten, wenn CRTL_DO1 gesetzt ist.
SET_DO2	Steuerbit DO2 Digitalausgang DO2 ein- und ausschalten, wenn CRTL_DO2 gesetzt ist.
SW_GATE	Steuerbit SW-Tor Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für die Zählbetriebsarten

Hinweis

Für das 1Count5V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count5V für die Zählbetriebsarten.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Zählwert oder abgespeicherter Zählwert bei Latch-Funktion am Digitaleingang		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO1
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO2
	Bit 3	Kurzschluss / Drahtbruch / Gebersignal	ERR_ENCODER
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	Status DO2	STS_DO2
	Bit 3	Status DO1	STS_DO1
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE

Adresse	Belegung		Bezeichnung
Byte 6	Bit 7	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung	STS_ND
	Bit 6	Untere Zählgrenze	STS_UFLW
	Bit 5	Obere Zählgrenze	STS_OFLW
	Bit 4	Status Vergleicher 2	STS_CMP2
	Bit 3	Status Vergleicher 1	STS_CMP1
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	Status Synchronisation	STS_SYN
Byte 7	reserviert = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 411) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_DO1	Kurzschluss / Drahtbruch / Übertemperatur durch Überlast am Ausgang DO1 Das Fehlerbit muss durch das Steuerbit (Seite 411) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_DO2	Kurzschluss / Drahtbruch / Übertemperatur am Ausgang DO2 Das Fehlerbit muss durch das Steuerbit (Seite 411) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_ENCODER	Kurzschluss / Drahtbruch 5 V-Gebersignal Das Fehlerbit muss durch das Steuerbit (Seite 411) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 411) LOAD_VAL, LOAD_PREPARE, CMP_VAL1, CMP_VAL2 und C_DOPARAM dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_CMP1	Status Vergleicher 1 Das Statusbit STS_CMP1 zeigt an, dass der Ausgang eingeschaltet ist oder eingeschaltet war. Es muss mit dem Steuerbit (Seite 411) RES_STS quittiert werden. Wird das Statusbit quittiert, wenn der Ausgang noch eingeschaltet ist, wird das Bit sofort wieder gesetzt. Dieses Bit wird auch gesetzt, wenn bei nicht freigegeben DO1 das Steuerbit SET_DO1 betätigt wird.
STS_CMP2	Status Vergleicher 2 Das Statusbit STS_CMP2 zeigt an, dass der Ausgang eingeschaltet ist oder eingeschaltet war. Es muss mit dem Steuerbit (Seite 411) RES_STS quittiert werden. Wird das Statusbit quittiert, wenn der Ausgang noch eingeschaltet ist, wird das Bit sofort wieder gesetzt. Dieses Bit wird auch gesetzt, wenn bei nicht freigegeben DO2 das Steuerbit SET_DO2 betätigt wird.

Bit	Erläuterung
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_DO1	Status DO1 Das Statusbit STS_DO1 zeigt den Zustand des Digitalausgangs DO1 an.
STS_DO2	Status DO2 Das Statusbit STS_DO2 zeigt den Zustand des Digitalausgangs DO2 an.
STS_GATE	Status internes Tor: Es wird gezählt.
STS_LOAD	Ladefunktion läuft
STS_ND	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung. Das Bit muss durch das Steuerbit (Seite 411) RES_STS zurückgesetzt werden.
STS_OFLW STS_UFLW	Obere Zählgrenze überschritten Untere Zählgrenze unterschritten Beide Bits müssen zurückgesetzt werden.
STS_SYN	Status Synchronisation Nach erfolgreicher Synchronisation ist das STS_SYN-Bit gesetzt. Es muss durch das Steuerbit (Seite 411) RES_STS zurückgesetzt werden.

Parameter für Messen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count5V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count5V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.
DO Ersatzwert schalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der parametrierte Ersatzwert für den Digitalausgang wird durchgeschaltet. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen. Ist für den Ausgang "Impuls bei Erreichen des Vergleichswertes" parametriert, steht nur für die Impulsdauer der Ersatzwert = 1 an.	Die geänderten Parameter werden übernommen und wirksam.
DO letzten Wert halten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Zustand des Digitalausgangs wird beibehalten. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.

[1] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count5V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count5V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 427) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count5V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Messverfahren

Übersicht

Messverfahren

Sie können beim Parameter "Messverfahren" zwischen folgenden Messverfahren wählen:

- mit Integrationszeit (Seite 417)
- kontinuierlich (Seite 418)

Messung mit Integrationszeit

Ablauf der Messung mit Integrationszeit

Die Messung wird während der von Ihnen parametrierten Integrationszeit durchgeführt. Wenn die Integrationszeit abgelaufen ist, wird der Messwert aktualisiert.

Dabei wird das Ende einer Messung mit dem Statusbit (Seite 429) STS_CMP1 gemeldet. Zurückgesetzt wird dieses Bit durch das Steuerbit (Seite 427) RES_STS in der Steuerschnittstelle.

Lagen in der parametrierten Integrationszeit nicht mindestens zwei steigende Flanken, so wird als Messwert 0 zurückgemeldet.

Bis zum Ende der ersten Integrationszeit wird der Wert -1 zurückgemeldet.

Sie können die Integrationszeit für die nächste Messung im Betrieb ändern.

Drehrichtungsumkehr

Wenn während einer Integrationszeit eine Drehrichtungsumkehr auftritt, dann ist der Messwert für diese Messperiode unbestimmt. Wenn Sie die Rückmeldebits (Seite 429) STS_C_UP und STS_C_DN (Richtungsauswertung) auswerten, können Sie auf eine eventuelle Prozessunregelmäßigkeit reagieren.

Siehe auch

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb (Seite 421)

Kontinuierliche Messung

Messprinzip

Das 1Count5V zählt jede positive Flanke eines Impulses und ordnet ihr einen Zeitwert in µs zu.

Die Aktualisierungszeit gibt an, in welchem Zeitabstand der Messwert vom Modul in der Rückmeldeschnittstelle aktualisiert wird.

Bei einer Impulsfolge mit einem oder mehreren Impulsen pro Aktualisierungszeit gilt:

Dynamische Messzeit =	Zeitwert des letzten Impulses im aktuellen Aktualisierungszeitintervall minus Zeitwert des letzten Impulses im vorangegangenen Aktualisierungszeitintervall

Nach dem Ablauf der Aktualisierungszeit wird ein neuer Messwert mit der dynamischen Messzeit errechnet und ausgegeben.

Enthält die aktuelle Aktualisierungszeit keinen Impuls, so ergibt sich folgende dynamische Messzeit:

Dynamische Messzeit =	Zeitwert der aktuellen abgelaufenen Aktualisierungszeit minus dem Zeitwert des letzten Impulses

Nach dem Ablauf einer Aktualisierungszeit wird ein geschätzter Messwert mit der dynamischen Messzeit errechnet unter der Annahme, dass ein Impuls am Ende der Aktualisierungszeit aufgetreten ist.

Ist der geschätzte Messwert "1 Impuls pro dynamische Messzeit" bei der Frequenz– und Drehzahlmessung kleiner als der letzte Messwert, dann wird dieser geschätzte Messwert als neuer Messwert ausgegeben. Bei der Periodendauermessung wird die dynamische Messzeit als geschätzte Periodendauer ausgegeben, wenn die dynamische Messzeit größer als die letzte gemessene Periodendauer ist.

Das folgende Bild zeigt das Messprinzip.

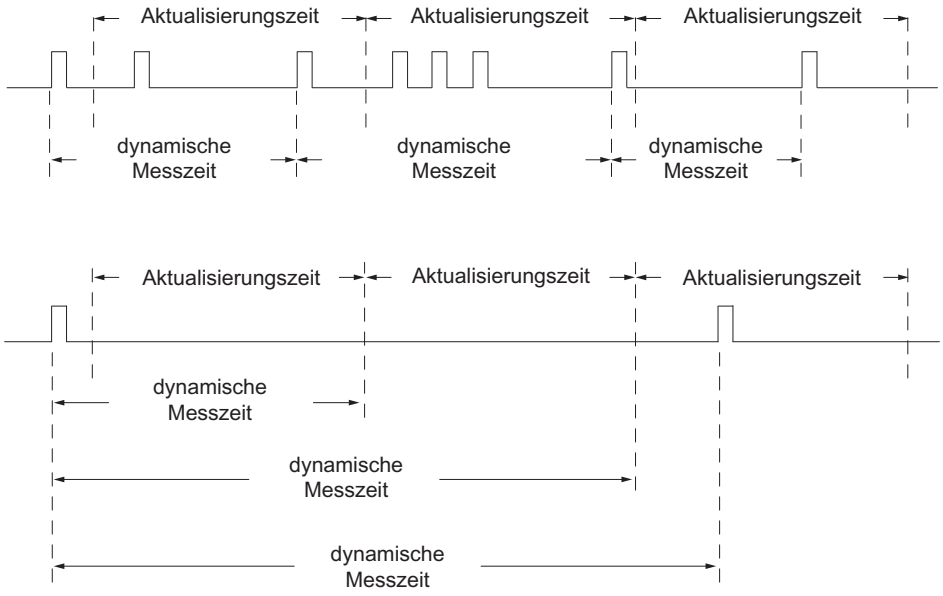

Das 1Count5V misst kontinuierlich. Sie geben beim Parametrieren eine Aktualisierungszeit vor.

In der Zeit bis zum Ende der ersten abgelaufenen Aktualisierungszeit wird der Wert "–1" zurückgemeldet.

Die kontinuierliche Messung beginnt nach dem Öffnen des Tores mit dem ersten Impuls der zu messenden Impulsfolge. Der erste Messwert kann frühestens nach dem zweiten Impuls berechnet werden.

Nach jedem Ablauf der Aktualisierungszeit wird in der Rückmeldeschnittstelle ein Messwert ausgegeben (Frequenz, Periodendauer oder Drehzahl). Dabei wird das Ende einer Messung mit dem Statusbit (Seite 429) STS_CMP1 gemeldet. Zurückgesetzt wird dieses Bit durch vollständiges Quittungsprinzip mit den Bits RES_STS und RES_STS_A.

Wenn in einer Aktualisierungszeit eine Drehrichtungsumkehr auftritt, dann ist der Messwert für diese Messperiode unbestimmt. Durch Auswertung der Rückmeldebits (Seite 429) STS_C_DN und STS_C_UP (Richtungsauswertung) können Sie auf eine eventuelle Prozessunregelmäßigkeit reagieren.

Das folgende Bild zeigt das Prinzip der kontinuierlichen Messung am Beispiel einer Frequenzmessung.

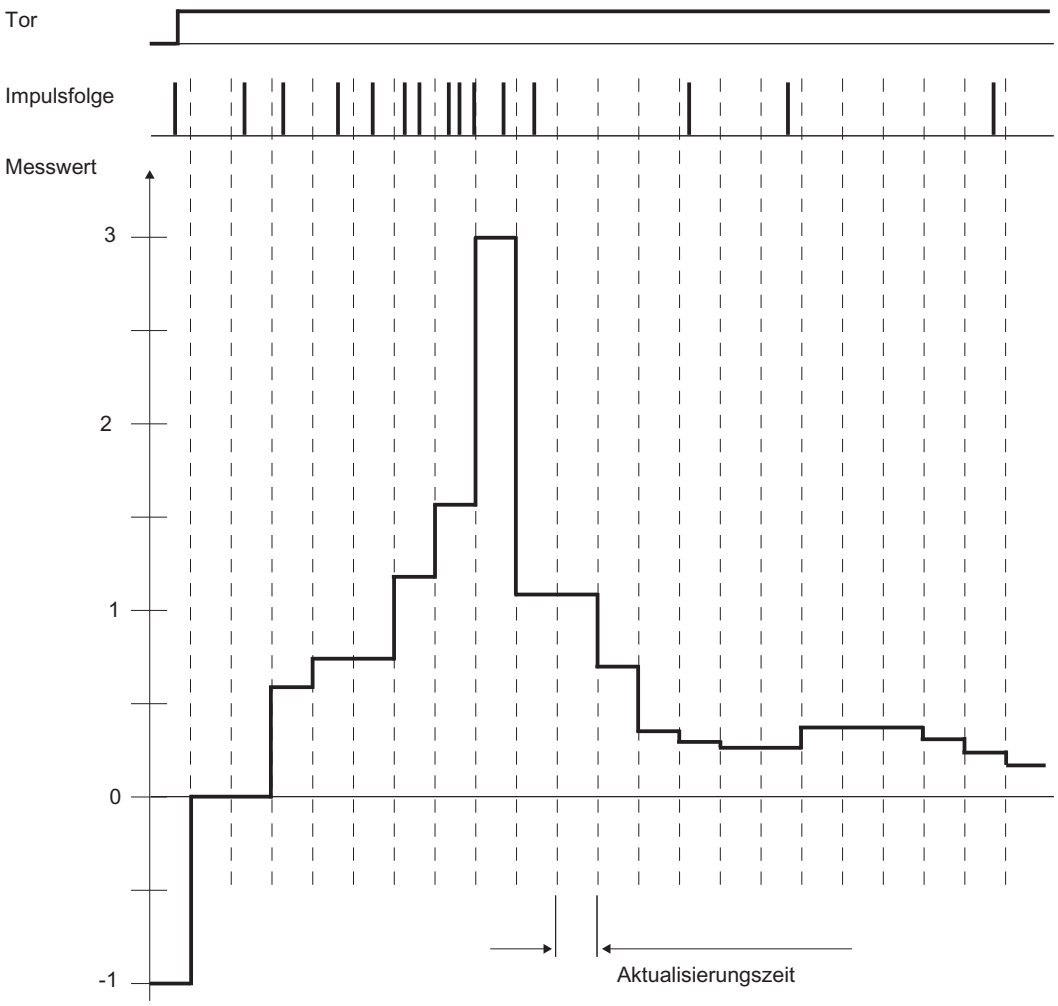

Torsteuerung

Zur Steuerung des 1Count5V müssen Sie Torfunktionen einsetzen.

Taktsynchroner Betrieb

Im taktsynchronen Betrieb übernimmt das 1Count5V in jedem Buszyklus Steuerbits und Steuerwerte aus der Steuerschnittstelle und meldet die Reaktion darauf noch im selben Zyklus zurück.

Das 1Count5V übergibt in jedem Zyklus einen Messwert und die Statusbits, wie sie zum Zeitpunkt T_i gültig waren.

Die Messung beginnt und endet jeweils zum Zeitpunkt T_i.

Siehe auch

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb (Seite 421)

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb

Integrationszeit und Aktualisierungszeit bei taktsynchronem Betrieb

Ist die Integrationszeit / Aktualisierungszeit mehrere T_{DP}-Zyklen lang, dann können Sie im Anwenderprogramm den neuen Messwert am Statusbit (Seite 429) STS_CMP1 (Messung beendet) der Rückmeldeschnittstelle erkennen. Dies ermöglicht eine Überwachung des Messvorgangs oder eine Synchronisation mit dem Messvorgang. Allerdings dauert das Quittieren dieser Meldung 4 T_{DP}-Zyklen. In diesem Fall beträgt die minimale Integrationszeit/ Aktualisierungszeit (4 × T_{DP}).

Wenn die Applikation einen Jitter der Integrationszeit / Aktualisierungszeit von einem T_{DP} und einen über mehrere Zyklen konstanten Messwert tolerieren kann, brauchen Sie das Statusbit STS_CMP1 nicht kontinuierlich auszuwerten. Dann sind auch Integrationszeiten / Aktualisierungszeiten von (1 × T_{DP}) bis (3 × T_{DP}) möglich.

Durch einen Verlust der Taktsynchronität im letzten T_{DP}-Zyklus der Integrationszeit verlängert sich die Integrationszeit / Aktualisierungszeit um einen T_{DP}-Zyklus. Der Messwert wird dadurch nicht verfälscht.

Hinweis

Sie dürfen die Wertebereichsgrenzen für die Integrationszeit / Aktualisierungszeit nicht überschreiten (siehe Tabellen zu jeder einzelnen Messbetriebsart).

Eine Verletzung der Wertebereichsgrenzen führt zu einem Parametrierfehler und das 1Count5V geht nicht in den taktsynchronen Betrieb.

Hinweis

Beim Ändern der Konfiguration vom nicht taktsynchronen in den taktsynchronen Betrieb und umgekehrt müssen Sie immer den Parameter Integrationszeit / Aktualisierungszeit anpassen, wenn Sie die Länge der Integrationszeit / Aktualisierungszeit beibehalten wollen.

Diagnose der Signalpaare aktivieren

Überwachung der Signalpaare

Die Signalpaare A und /A, B und /B können überwacht werden.

Aktivieren Sie dazu das Optionskästchen.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Verhalten der Ausgänge bei den Messbetriebsarten

Verhalten des Ausgangs DO1 auswählen

Verhalten des Ausgangs bei Messbetriebsarten

Der Digitalausgang des 1Count5V ist parametrierbar.

Sie können für Frequenzmessung, Drehzahlmessung oder Periodendauermessung jeweils einen oberen und einen unteren Grenzwert hinterlegen, bei deren Verletzung der Digitalausgang DO1 aktiviert wird. Diese Grenzwerte sind parametrierbar und mit der Ladefunktion änderbar.

Sie können die Funktion und das Verhalten des Digitalausgangs im Betrieb ändern. Die neue Funktion wird sofort wirksam, im taktsynchronen Betrieb immer zum Zeitpunkt T_i.

Sie können in der Klappliste zwischen folgenden Funktionen auswählen:

- Ausgang (Seite 422)
- Außerhalb der Grenzwerte (Seite 423) (Grenzwertüberwachung)
- Unterschreiten des unteren Grenzwertes (Seite 423) (Grenzwertüberwachung)
- Überschreiten des oberen Grenzwertes (Seite 423) (Grenzwertüberwachung)

Ausgang

Ausgang

Wenn Sie die Ausgänge ein- oder ausschalten wollen, müssen Sie sie mit den Steuerbits (Seite 427) CTRL_DO1 bzw. CTRL_DO2 freigeben.

Mit den Steuerbits (Seite 427) SET_DO1 bzw. SET_DO2 schalten Sie die Ausgänge ein und aus.

Mit den Statusbits (Seite 429) STS_DO1 bzw. STS_DO2 in der Rückmeldeschnittstelle können Sie den Status der Ausgänge abfragen.

Im taktsynchronen Betrieb schalten die Ausgänge zum Zeitpunkt T_o.

Grenzwertüberwachung

Grenzwertüberwachung

Das folgende Bild zeigt das Prinzip der Grenzwertüberwachung.

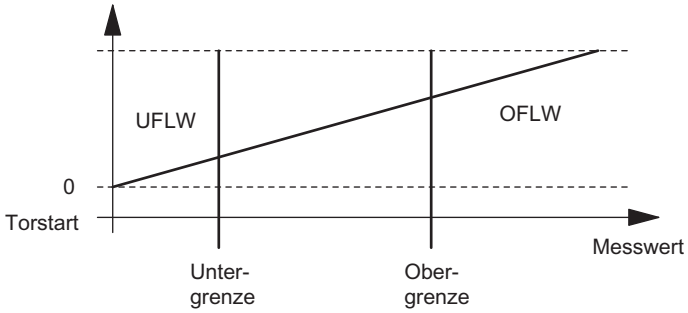

Nach abgelaufener Integrationszeit wird der ermittelte Messwert (Frequenz, Drehzahl oder Periodendauer) mit den parametrierten Grenzwerten verglichen.

Unterschreitet der aktuelle Messwert den parametrierten unteren Grenzwert (Messwert < unterer Grenzwert), wird das Bit STS_UFLW = 1 in der Rückmeldeschnittstelle (Seite 429) gesetzt.

Überschreitet der aktuelle Messwert den parametrierten oberen Grenzwert (Messwert > oberer Grenzwert), wird das Bit STS_OFLW = 1 in der Rückmeldeschnittstelle (Seite 429) gesetzt.

Diese Bits müssen Sie mit dem Steuerbit (Seite 427) RES_STS quittieren.

Befindet sich nach dem Quittieren der Messwert noch oder wieder außerhalb der Grenzwerte, wird das entsprechende Statusbit erneut gesetzt.

Wenn Sie den unteren Grenzwert = 0 setzen, schalten Sie damit die dynamische Überwachung auf "Unterschreiten des unteren Grenzwertes" aus.

Die folgende Tabelle zeigt, wie der freigegebene Digitalausgang DO1 je nach Parametrierung durch die Grenzwertüberwachung gesetzt werden kann:

Parameter "Funktion DO1"	DO1 wird gesetzt bei ...
Außerhalb der Grenzwerte	Messwert < unterer Grenzwert ODER Messwert > oberer Grenzwert
Unterschreiten des unteren Grenzwertes	Messwert < unterer Grenzwert
Überschreiten des oberen Grenzwertes	Messwert > oberer Grenzwert

Im taktsynchronen Betrieb schaltet der Ausgang nach Ende der Messung zum Zeitpunkt T_i.

Ersatzwert DOx auswählen

Ersatzwert DO1 und Ersatzwert DO2

Wenn Sie unter "Verhalten bei CPU/Master-STOP (Seite 416)" den Parameter "DO Ersatzwert schalten" ausgewählt haben, dann wählen Sie jeweils den Ersatzwert aus, der am Digitalausgang DO1 bzw. DO2 ausgegeben werden soll.

Diagnose DOx aktivieren

Diagnose DO1 und Diagnose DO2

"Diagnose DO1" bzw. "Diagnose DO2" (Leitungsbruch, Kurzschluss) ist nur bei Impulslängen > 90 ms am Digitalausgang DO1 bzw. DO2 möglich.

Aktivieren Sie das jeweilige Optionskästchen für die Diagnose am Ausgang DO1 bzw. DO2.

Messbetriebsart auswählen

Beschreibung

Mit der Vorgabe einer Messbetriebsart wählen Sie, mit welcher Funktionalität Sie das 1Count5V betreiben wollen.

	Beschreibung	
Betriebsart	**Messung mit Integrationszeit**	**Kontinuierliche Messung**
Frequenzmessung (Seite 195) (mit HW- oder SW-Tor)	Das 1Count5V zählt die Impulse, die in einer vorgegebenen Integrationszeit eintreffen.	Das 1Count5V zählt die Impulse, die in einer dynamischen Messzeit eintreffen.
Drehzahlmessung (Seite 196) (mit HW- oder SW-Tor)	Das 1Count5V zählt die Impulse, die in einer vorgegebenen Integationszeit von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.	Das 1Count5V zählt die Impulse, die in einer dynamischen Messzeit von einem Drehzahlgeber eintreffen, und berechnet hieraus mit den Impulsen pro Geberumdrehung die Drehzahl.
Periodendauermessung (Seite 198) (mit HW- oder SW-Tor)	Das 1Count5V misst die Zeit zwischen zwei positiven Flanken des Zählsignals, indem es die Impulse einer internen quarzgenauen Referenzfrequenz (16 MHz) in einer vorgegebenen Integrationszeit zählt.	Das 1Count5V gibt die dynamische Messzeit als Periodendauer an. Ist die Periodendauer kleiner als die Aktualisierungszeit, dann wird für die Periodendauer ein Mittelwert gebildet.

Die Voreinstellung ist Betriebsart "Frequenzmessung".

Auflösung der Periodendauer auswählen

Beschreibung

Der Wert der ermittelten Periodendauer wird in der Einheit 1 µs oder 1/16 µs zur Verfügung gestellt.

Wählen Sie die gewünschte Einheit für die Auflösung aus.

Mögliche Messbereiche mit Fehlerangaben

- Periodendauermessung mit Integrationszeit
  - Messbereiche und Fehler bei einer Auflösung von 1 µs:

Integrationszeit	T_{min} ± absoluter Fehler	T_{min} ± absoluter Fehler
100 s	1 µs* (10 ± 0)	1 µs* (100 000 000 ± 10 000)
10 s	1 µs* (10 ± 0)	1 µs* (10 000 000 ± 1 000)
1 s	1 µs* (10 ± 0)	1 µs* (1 000 000 ± 100)
0,1 s	1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
0,01 s	1 µs* (10 ± 0)	1 µs* (10 000 ± 1)

  - Messbereiche und Fehler bei einer Auflösung von 1/16 µs:

Integrationszeit	T_{min} ± absoluter Fehler	T_{min} ± absoluter Fehler
100 s	1/16 µs* (160 ± 0)	1/16 µs* (1 600 000 000 ± 160 000)
10 s	1/16 µs* (160 ± 0)	1/16 µs* (160 000 000 ± 16 000)
1 s	1/16 µs* (160 ± 0)	1/16 µs* (16 000 000 ± 1 600)
0,1 s	1/16 µs* (160 ± 0)	1/16 µs* (1 600 000 ± 160)
0,01 s	1/16 µs* (160 ± 0)	1/16 µs* (160 000 ± 16)

- Kontinuierliche Periodendauermessung:
  - Messbereiche und Fehler bei einer Auflösung von 1 µs:

Periodendauer T_{min} ± absoluter Fehler	Periodendauer T_{min} ± absoluter Fehler
1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
1 µs* (100 ± 0)	1 µs* (1 000 000 ± 100)
1 µs* (1 000 ± 0)	1 µs* (10 000 000 ± 1 002)
1 µs* (10 000 ± 1)	1 µs* (100 000 000 ± 10 020)

  - Messbereiche und Fehler bei einer Auflösung von 1/16 µs:

Periodendauer T_{min} ± absoluter Fehler	Periodendauer T_{min} ± absoluter Fehler
1/16 µs* (160 ± 1)	1/16 µs* (1 600 000 ± 160)
1/16 µs* (1 600 ± 1)	1/16 µs* (16 000 000 ± 1 600)
1/16 µs* (16 000 ± 3)	1/16 µs* (160 000 000 ± 16 000)
1/16 µs* (160 000 ± 20)	1/16 µs* (1 600 000 000 ± 160 000)

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.

Die Voreinstellung ist "Eingang".

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 429) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Unteren und oberen Grenzwert festlegen

Unterer und oberer Grenzwert

Sie können für Frequenzmessung, Drehzahlmessung oder Periodendauermessung jeweils einen oberen und einen unteren Grenzwert hinterlegen, bei deren Verletzung der Digitalausgang DO1 aktiviert (Seite 422) wird. Diese Grenzwerte können Sie parametrieren und über die Ladefunktion ändern.

Für die unteren und oberen Grenzwerte sind folgende Wertebereiche zulässig:

Messbetriebsart	Unterer Grenzwert (UG)	Oberer Grenzwert
Frequenzmessung	0 bis 99 999 999 × 10^{-3} Hz	UG+1 bis 100 000 000 × 10^{-3} Hz
Drehzahlmessung	0 bis 24 999 999 × 10^{-3}/min	UG+1 bis 25 000 000 × 10^{-3}/min
Periodendauermessung (Auflösung 1 µs)	0 bis 119 999 999 µs	UG+1 bis 120 000 000 µs
Periodendauermessung (Auflösung 1/16 µs)	0 bis 1 919 999 999 × 1/16 µs	UG+1 bis 1 920 000 000 × 1/16 µs

Messzeit festlegen

Beschreibung

Wenn Sie eine Messbetriebsart ausgewählt haben, dann aktualisiert die Baugruppe die Messwerte zyklisch. Die Aktualisierungszeit / Integrationszeit ist das Produkt aus Messzeit × 10 ms. Die Messzeit geben Sie als Faktor n vor.

Detaillierte Beschreibungen zu den Wertebereichen für die Messzeit finden Sie unter "Grundsätzliches zum Messen (Seite 190)".

Impulse pro Geberumdrehung eintragen

Impulse pro Geberumdrehung

Für die Betriebsart "Drehzahlmessung (Seite 196)" müssen Sie zusätzlich die Impulse pro Geberumdrehung parametrieren.

Den Wert für die Anzahl der Impulse pro Geberumdrehung finden Sie auf dem Typenschild oder in den technischen Daten Ihres Gebers.

Steuer- und Rückmeldeschnittstelle für die Messbetriebsarten

Steuerschnittstelle für die Messbetriebsarten

Hinweis

Für das 1Count5V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count5V für die Messbetriebsarten.

Adresse	Belegung				
Byte 0 bis 3	**Unterer oder oberer Grenzwert**				
	Funktion von DO1				
	Byte 0	Bit 1	Bit 0	Funktion DO1	
		0	0	Ausgang	
		0	1	Außerhalb der Grenzen	
		1	0	Unterschreiten des unteren Grenzwertes	
		1	1	Überschreiten des oberen Grenzwertes	
	Byte 1 bis 3	reserviert = 0			
	Integrationszeit / Aktualisierungszeit				
	Byte 0, 1	Integrationszeit [n × 10 ms] (Wertebereich 1 … 1000 / 12000)			
	Byte 2, 3	reserviert = 0			
					Bezeichnung
Byte 4	Bit 7	Fehlerquittung			EXTF_ACK
	Bit 6	Freigabe DO2			CTRL_DO2
	Bit 5	Steuerbit DO2			SET_DO2
	Bit 4	Freigabe DO1			CTRL_DO1
	Bit 3	Steuerbit DO1			SET_DO1
	Bit 2	Anstoß Rücksetzen Statusbits			RES_STS
	Bit 1	reserviert = 0			
	Bit 0	Steuerbit SW-Tor			SW_GATE
Byte 5	Bit 7	reserviert = 0			
	Bit 6	reserviert = 0			
	Bit 5	reserviert = 0			
	Bit 4	Funktion von DO1 ändern			C_DOPARAM
	Bit 3	reserviert = 0			
	Bit 2	Integrationszeit ändern			C_INTTIME
	Bit 1	Oberen Grenzwert laden			LOAD_PREPARE
	Bit 0	Unteren Grenzwert laden			LOAD_VAL
Byte 6 bis 7	reserviert = 0 [1]				

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
C_DOPARAM	Funktion von DO1 ändern Der Wert aus Byte 0 wird als neue Funktion von DO1 übernommen.
C_INTTIME	Integrationszeit ändern Der Wert aus Byte 0 und 1 wird als neue Integrationszeit für die nächste Messung übernommen.
CTRL_DO1	Freigabe DO1 Mit diesem Bit geben Sie den Ausgang DO1 frei.
CTRL_DO2	Freigabe DO2 Mit diesem Bit geben Sie den Ausgang DO2 frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Oberen Grenzwert laden Der Wert aus Byte 0 bis 3 wird als neuer oberer Grenzwert übernommen.
LOAD_VAL	Unteren Grenzwert laden Der Wert aus Byte 0 bis 3 wird als neuer Unterer Grenzwert übernommen.
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SET_DO1	Steuerbit DO1 Digitalausgang DO1 ein- und ausschalten, wenn CRTL_DO1 gesetzt ist.
SET_DO2	Steuerbit DO2 Digitalausgang DO2 ein- und ausschalten, wenn CRTL_DO2 gesetzt ist.
SW_GATE	Steuerbit SW-Tor Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für die Messbetriebsarten

Hinweis

Für das 1Count5V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count5V für die Messbetriebsarten.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Messwert		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO1
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	Kurzschluss / Drahtbruch / Übertemperatur	ERR_DO2
	Bit 3	Kurzschluss / Drahtbruch / Gebersignal	ERR_ENCODER
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	Status DO2	STS_DO2
	Bit 3	Status DO1	STS_DO1
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE
Byte 6	Bit 70	reserviert = 0	
	Bit 6	Unterer Grenzwert Messbereich	STS_UFLW
	Bit 5	Oberer Grenzwert Messbereich	STS_OFLW
	Bit 4	reserviert = 0	
	Bit 3	Messung beendet	STS_CMP1
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	reserviert = 0	
Byte 7	reserviert = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 427) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_DO1	Kurzschluss / Drahtbruch / Übertemperatur am Ausgang DO1 Das Fehlerbit muss durch das Steuerbit (Seite 427) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.

Bit	Erläuterung
ERR_DO2	Kurzschluss / Drahtbruch / Übertemperatur am Ausgang DO2 Das Fehlerbit muss durch das Steuerbit (Seite 427) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_ENCODER	Kurzschluss / Drahtbruch Gebersignal Das Fehlerbit muss durch das Steuerbit (Seite 427) EXTF_ACK quittiert werden Diagnosemeldung, falls parametriert.
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 427) LOAD_VAL, LOAD_PREPARE, C_DOPARAM und C_INTTIME dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_CMP1	Messung beendet Nach jedem abgelaufenen Zeitintervall (Aktualisierungszeit / Integrationszeit) wird der Messwert aktualisiert. **Messung mit Integrationszeit:** Dabei wird das Ende einer Messung (nach Ablauf des Zeitintervalls) mit dem Statusbit STS_CMP1 gemeldet. **Kontinuierliche Messung:** Am Ende der Aktualisierungszeit wird das Ende der Messung mit dem Statusbit STS_CMP1 gemeldet, wenn ein gemessener Wert ausgegeben wird. Bei Ausgabe eines geschätzten Messwertes bleibt das Bit "0". Zurückgesetzt wird dieses Bit durch das Steuerbit (Seite 427) RES_STS in der Steuerschnittstelle.
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_DO1	Status DO1
STS_DO2	Status DO2
STS_GATE	Status internes Tor: Es wird gemessen.
STS_LOAD	Ladefunktion läuft
STS_OFLW STS_UFLW	Oberer Grenzwert überschritten Unterer Grenzwert unterschritten Beide Bits müssen zurückgesetzt werden.

Parameter für Wegerfassen

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count5V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count5V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1)]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.

1) Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count5V.

Verlassen des parametrierten Zustands

Unter welchen Bedingungen verlässt das 1Count5V den parametrierten Zustand?

Die CPU bzw. der Master müssen im RUN sein, und Sie müssen eine Änderung an der Steuerschnittstelle (Seite 440) vornehmen.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count5V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung auswählen

Signalauswertung A, B

Die Signalauswertung über A, B ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Drehgeber (Seite 210)
  Das 1Count5V kann die Flanken der Signale zählen. Im Normalfall werden nur die Flanken an A ausgewertet (Einfachauswertung). Um zu einer höheren Auflösung zu gelangen, können Sie bei der Parametrierung (Parameter "Signalauswertung") wählen, ob die Signale einfach (Seite 211), zweifach (Seite 211) oder vierfach (Seite 211) ausgewertet werden sollen.
  Mehrfachauswertung ist nur bei asymmetrischen Inkrementalgebern mit um 90 Grad versetzten Signalen A und B möglich.

  Hinweis

  Die Angabe 500 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A bzw. B bezogen. Bei Zweifachauswertung ergibt sich somit max. 1 MHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 2 MHz.

Diagnose der Signalpaare aktivieren

Überwachung der Signalpaare

Die Signalpaare A und /A, B und /B, N und /N können überwacht werden.

Aktivieren Sie dazu das entsprechende Optionskästchen.

Hinweis

Die "Diagnose N, /N" können Sie erst aktivieren, wenn Sie zuvor die "Diagnose A, /A und B, /B" aktiviert haben.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Latch/Retrigger bei positiver Flanke (Seite 434)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht). Nach jedem Speichern wird der Zähler auf den Ladewert gesetzt.
- Synchronisation bei positiver Flanke (Seite 436)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.
- Latchen bei positiver Flanke (Seite 437)
  Mit positiven Flanken am Digitaleingang werden Zählerstände gespeichert (gelatcht).
- HW-Freigabe für Synchronisation (Seite 438)
  Der Digitaleingang dient als HW-Freigabe. Bei aktiver HW-Freigabe wird mit der Nullmarke des Gebers der Zähler auf den Ladewert gesetzt.

Die Voreinstellung ist "Eingang".

Latch/Retrigger bei positiver Flanke

Latch/Retrigger

Mit dieser Funktion speichern Sie bei einer positiven Flanke am Digitaleingang den aktuellen internen Zählerstand des 1Count5V und retriggern den Zählvorgang. Dies bedeutet, dass der aktuelle interne Zählerstand zum Zeitpunkt der positiven Flanke abgespeichert (Latchwert) und dann das 1Count5V wieder mit dem Ladewert geladen wird und von dort weiterzählt.

Das folgende Bild zeigt Latch/Retrigger mit dem Ladewert = 0.

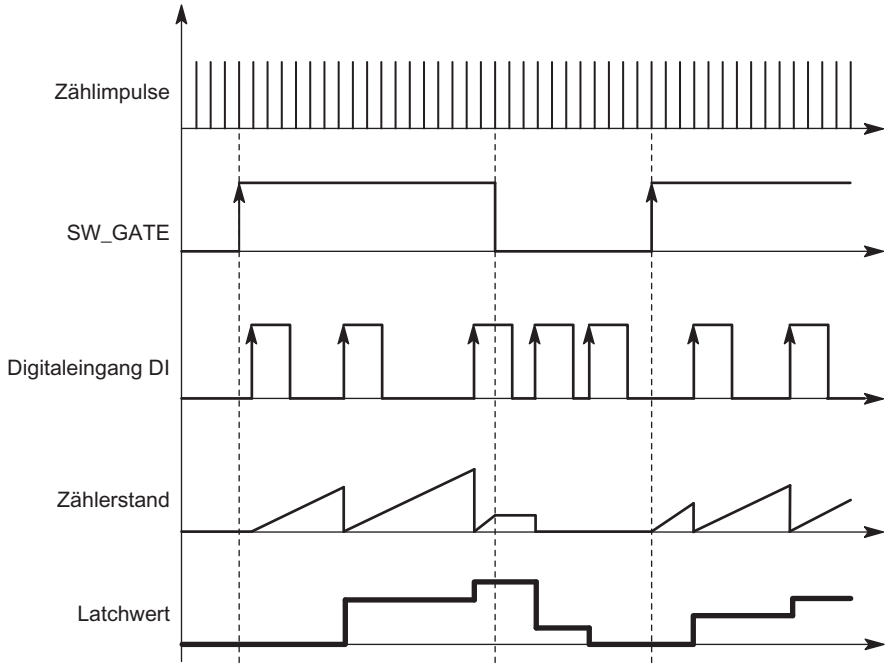

Zur Ausführung der Funktion muss die Zählbetriebsart mit dem SW-Tor freigegeben sein. Gestartet wird sie mit der ersten positiven Flanke am Digitaleingang.

In der Rückmeldeschnittstelle (Seite 442) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Status des Latch- und Retrigger-Signals.

Der Latchwert wird mit seinem RESET-Zustand ("0") vorbelegt. Er wird durch das Öffnen des SW-Tors nicht verändert.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Wenn Sie das SW-Tor schließen, wirkt es nur unterbrechend; d. h. wenn Sie das SW-Tor erneut öffnen, wird der Zählvorgang fortgesetzt. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Auch im taktsynchronen Betrieb wird der Zählvorgang mit jeder Flanke am Digitaleingang gelatcht und getriggert. In der Rückmeldeschnittstelle wird der Zählerstand angezeigt, der zum Zeitpunkt der letzten Flanke vor T_i gültig war.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count5V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 442) gelesen werden.

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die steigende Flanke eines Referenzsignals am Eingang zum Setzen des 1Count5V auf den Ladewert.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

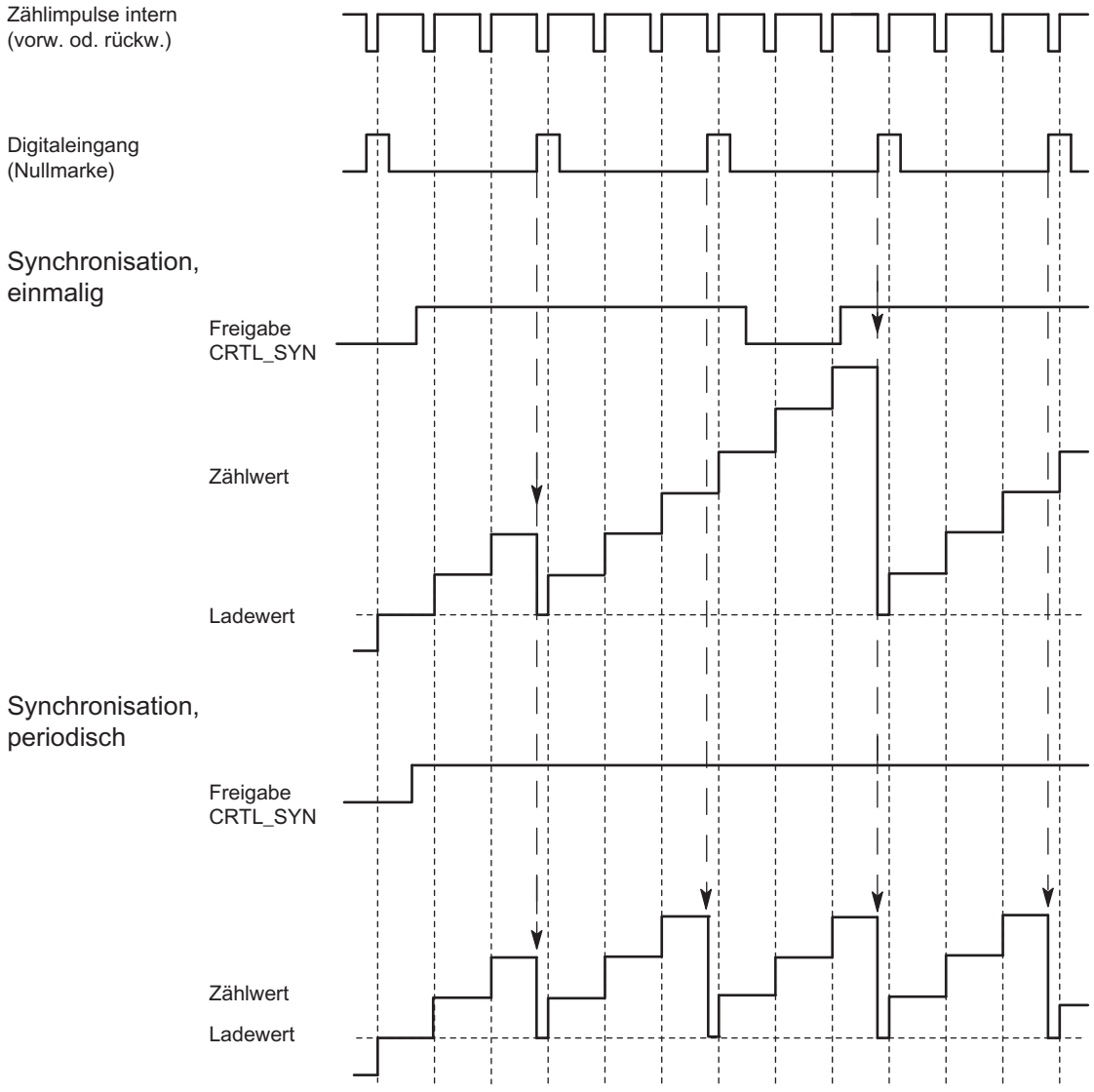

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 440) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.

- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count5V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count5V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 442) STS_SYN gesetzt und die LED SYN leuchtet. Durch das Steuerbit (Seite 440) RES_STS wird das Rückmeldebit zurückgesetzt und die LED ausgeschaltet.
- Als Referenzsignal kann das Signal eines prellfreien Schalters dienen.
- Das Rückmeldebit (Seite 442) STS_DI zeigt den Pegel des Referenzsignals an.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 442) STS_SYN an, dass die steigende Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Latchen bei positiver Flanke

Latchen

Zählerstand und Latchwert werden mit ihren RESET-Zuständen ("0") vorbelegt.

Mit dem Öffnen des SW-Tors wird die Zählfunktion gestartet. Das 1Count5V beginnt beim Ladewert ("0").

Das folgende Bild zeigt Latchen mit Ladewert = 0.

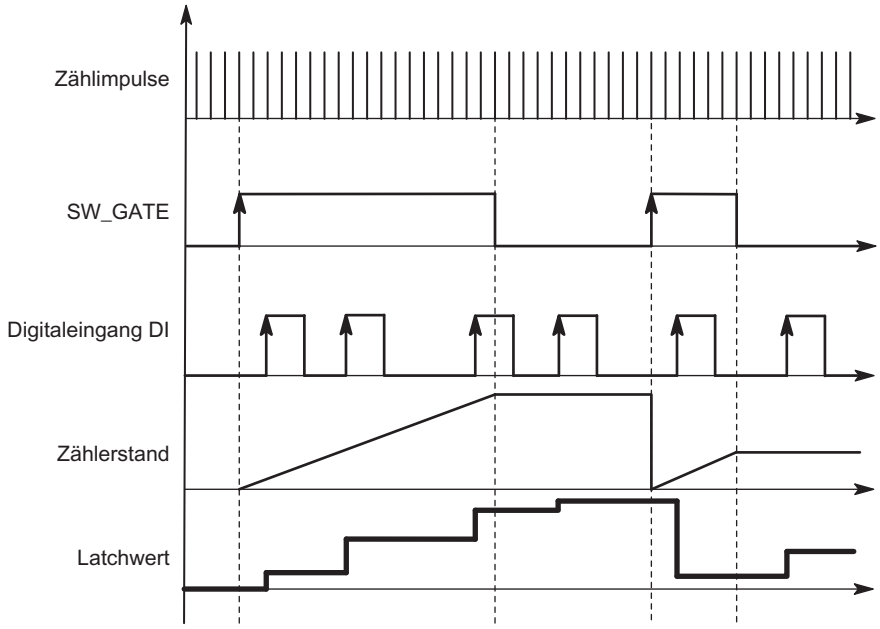

Der Latchwert ist immer genau der Zählerstand zum Zeitpunkt der positiven Flanke am Digitaleingang DI.

In der Rückmeldeschnittstelle (Seite 442) wird statt des aktuellen Zählerstands der gespeicherte Zählerstand angezeigt. Das STS_DI-Bit zeigt den Pegel des Latch-Signals.

Ein direktes Laden des Zählers führt nicht zum Verändern des angezeigten abgespeicherten Zählerstands.

Im taktsynchronen Betrieb wird in der Rückmeldeschnittstelle der Zählerstand angezeigt, der zum Zeitpunkt der letzten positiven Flanke vor T_i gelatcht wurde.

Wenn Sie das SW-Tor schließen, wirkt es wie parametriert, abbrechend oder unterbrechend. Der Digitaleingang DI bleibt auch bei geschlossenem SW-Tor aktiv.

Erweiterte Rückmeldeschnittstelle

Steckt das 1Count5V hinter einer IM 151, die das Lesen und Beschreiben von breiteren Nutzdatenschnittstellen unterstützt, so kann der laufende Zählwert aus den Bytes 8 bis 11 der Rückmeldeschnittstelle (Seite 442) gelesen werden.

HW-Freigabe für Synchronisation

Synchronisation mit DI und Nullmarke

Wenn Sie "HW-Freigabe für Synchronisation" parametriert haben, dient der DI als HW-Freigabe. Wenn die HW-Freigabe aktiv ist, wird das 1Count5V durch die Nullmarke des Gebers mit dem Ladewert geladen.

Sie können zwischen einmaliger und periodischer Synchronisation (Parameter "Synchronisation (Seite 440)") wählen.

Das folgende Bild zeigt die Zählerzustände bei einmaliger und periodischer Synchronisation.

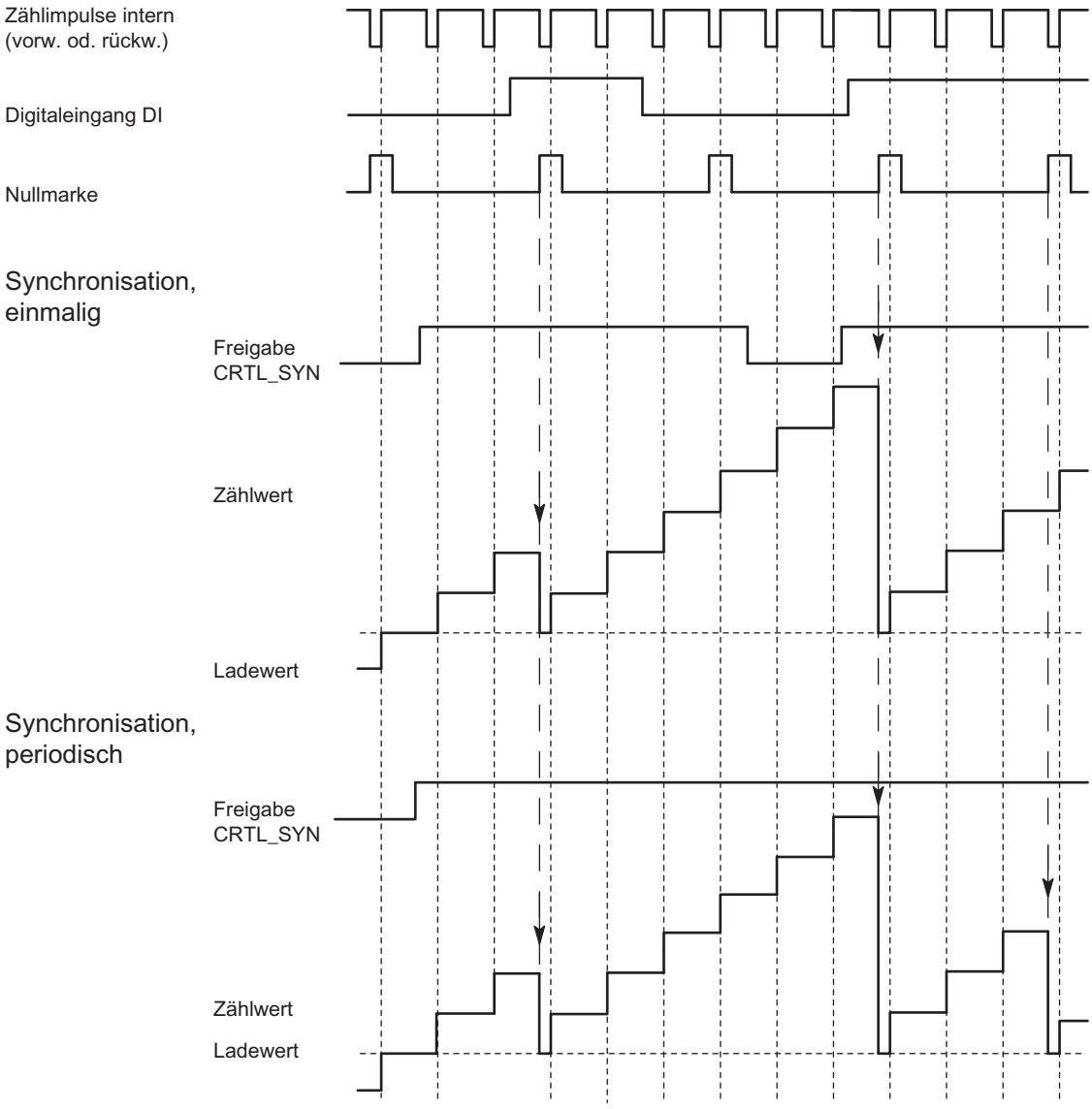

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit (Seite 440) "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.
- Einmalige Synchronisation: Wenn das Freigabebit gesetzt ist und die HW-Freigabe vorliegt, lädt die erste Nullmarke das 1Count5V mit dem Ladewert.
- Periodische Synchronisation: Wenn das Freigabebit gesetzt ist und die HW-Freigabe vorliegt, lädt die erste und jede weitere Nullmarke das 1Count5V mit dem Ladewert.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 442) STS_SYN gesetzt und die LED SYN leuchtet. Durch das Steuerbit (Seite 440) RES_STS wird das Rückmeldebit zurückgesetzt und die LED ausgeschaltet.
- Als Referenzsignal zur HW-Freigabe kann das Signal eines prellfreien Schalters dienen.
- Das Rückmeldebit (Seite 442) STS_DI zeigt den Pegel des Referenzsignals.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit (Seite 442) STS_SYN an, dass die positive Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 442) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Synchronisation auswählen

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" am Parameter "Funktion DI (Seite 434)" gewählt haben, dann können Sie hier zwischen einmaliger und periodischer Synchronisation wählen:

- Einmalig: Wenn das Freigabebit gesetzt ist, lädt die erste Flanke das 1Count5V mit dem Ladewert.
- Periodisch: Wenn das Freigabebit gesetzt ist, lädt die erste und jede weitere Flanke das 1Count5V mit dem Ladewert.

Steuer- und Rückmeldeschnittstelle für Wegerfassen

Steuerschnittstelle für Wegerfassen

Hinweis

Für das 1Count5V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1Count5V für Wegerfassen.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Ladewert direkt, vorbereitend		
Byte 4	Bit 7	Fehlerquittung	EXTF_ACK
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Anstoß Rücksetzen Statusbits	RES_STS
	Bit 1	Freigabe Synchronisation	CTRL_SYN
	Bit 0	Steuerbit SW-Tor	SW_GATE
Byte 5	Bit 7	reserviert = 0	
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	Zähler vorbereitend laden	LOAD_PREPARE
	Bit 0	Zähler direkt laden	LOAD_VAL
Byte 6 bis 7	reserviert = 0 [1]		

[1] Fehlt bei erweiterter Nutzdatenschnittstelle

Erläuterungen zu den Steuerbits

Bit	Erläuterung
CTRL_SYN	Sie geben damit die Synchronisation frei.
EXTF_ACK	Fehlerquittung Die Fehlerbits müssen mit dem Steuerbit EXTF_ACK nach der Beseitigung der Ursache quittiert werden.
LOAD_PREPARE	Zähler vorbereitend laden Der Wert aus Byte 0 bis 3 wird als Ladewert übernommen.
LOAD_VAL	Der Wert aus Byte 0 bis 3 wird als neuer Zählwert direkt geladen.
RES_STS	Anstoß Rücksetzen Statusbits Das Rücksetzen der Statusbits erfolgt durch den Quittungsablauf zwischen RES_STS-Bit und RES_STS_A-Bit.
SW_GATE	Steuerbit SW-Tor Das SW-Tor wird über die Steuerschnittstelle mit dem Bit SW_GATE geöffnet / geschlossen.

Rückmeldeschnittstelle für Wegerfassen

Hinweis

Für das 1Count5V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7
- Byte 8 … 11 (erweiterte Nutzdatenschnittstelle)

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count5V für Wegerfassen.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Zählwert oder abgespeicherter Zählwert bei Latch-Funktion am Digitaleingang		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	reserviert = 0	
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	Rücksetzen der Statusbits läuft	RES_STS_A
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 5	Bit 7	Status Richtung rückwärts	STS_C_DN
	Bit 6	Status Richtung vorwärts	STS_C_UP
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	Status DI	STS_DI
	Bit 0	Status internes Tor	STS_GATE
Byte 6	Bit 7	Nulldurchgang	STS_ND
	Bit 6	Untere Zählgrenze	STS_UFLW
	Bit 5	Obere Zählgrenze	STS_OFLW
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	reserviert = 0	
	Bit 0	Status Synchronisation	STS_SYN
Byte 7	reserviert = 0		
Byte 8 bis 11	Zählwert [1]		

[1] Erweiterte Nutzdatenschnittstelle

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_24V	Kurzschluss Geberversorgung Das Fehlerbit muss durch das Steuerbit (Seite 440) EXTF_ACK quittiert werden. Diagnosemeldung, falls parametriert.
ERR_LOAD	Fehler bei Ladefunktion Die Steuerbits (Seite 440) LOAD_VAL und LOAD_PREPARE dürfen während der Übergabe nicht gleichzeitig gesetzt sein. Dies führt, wie das Laden eines falschen Werts (dieser wird nicht übernommen), zum Setzen des Statusbits ERR_LOAD.
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
RES_STS_A	Rücksetzen der Statusbits läuft
STS_C_DN	Status Richtung rückwärts
STS_C_UP	Status Richtung vorwärts
STS_DI	Status DI Der Zustand des DI wird bei jeder Betriebsart in der Rückmeldeschnittstelle mit dem STS_DI-Bit angezeigt.
STS_GATE	Status internes Tor: Es wird gezählt.
STS_LOAD	Ladefunktion läuft
STS_ND	Nulldurchgang im Zählbereich bei Zählen ohne Hauptzählrichtung. Das Bit muss durch das Steuerbit (Seite 440) RES_STS zurückgesetzt werden.
STS_OFLW STS_UFLW	Obere Zählgrenze überschritten Untere Zählgrenze unterschritten Beide Bits müssen zurückgesetzt werden.
STS_SYN	Status Synchronisation Nach erfolgreicher Synchronisation ist das STS_SYN-Bit gesetzt. Es muss durch das Steuerbit (Seite 440) RES_STS zurückgesetzt werden.

Parameter für Fast Mode

Verhalten bei CPU/Master-STOP auswählen

Einstellung des Verhaltens bei CPU/Master-STOP

Sie können das Verhalten des 1Count5V bei Ausfall der überlagerten Steuerung parametrieren.

Parameter	Zustand des 1Count5V bei CPU/Master-STOP	Was passiert bei Neuparametrierung?
DO abschalten	Die laufende Betriebsart wird abgebrochen. Das Tor wird geschlossen und der Digitalausgang gesperrt. Vergleichswert 1 und 2 und Ladewert werden zurückgesetzt. Oberer und unterer Grenzwert, Funktion und Verhalten der Digitalausgänge und die Integrationszeit werden aus den Parametern übernommen.	Die geänderten Parameter werden übernommen und wirksam.
Betriebsart weiterarbeiten [1)]	Die laufende Betriebsart arbeitet weiter. Tor und Digitalausgang behalten ihren Zustand.	Das Tor wird geschlossen. Die laufende Betriebsart wird abgebrochen. Der Digitalausgang wird gesperrt. Die geänderten Parameter werden übernommen und wirksam.

[1)] Soll die Betriebsart beim Wechsel von CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU/der Master die Ausgänge nicht löschen.
Mögliche Abhilfe: Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit SW_GATE und übertragen Sie die Werte zum 1Count5V.

Automatische Neuparametrierung

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU/DP-Master erfolgt bei:

- NETZ-EIN der CPU/DP-Master
- NETZ-EIN der IM 151-1 / IM 151-1 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU/DP-Master
- beim Stecken des 1Count5V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Signalauswertung auswählen

Signalauswertung A, B

Die Signalauswertung über A, B ermöglicht Ihnen ein richtungsabhängiges Zählen. Je nach Parametrierung sind verschiedene Auswertungsarten möglich:

- Drehgeber (Seite 210)
  Das 1Count5V kann die Flanken der Signale zählen. Im Normalfall werden nur die Flanken an A ausgewertet (Einfachauswertung). Um zu einer höheren Auflösung zu gelangen, können Sie bei der Parametrierung (Parameter "Signalauswertung") wählen, ob die Signale einfach (Seite 211), zweifach (Seite 211) oder vierfach (Seite 211) ausgewertet werden sollen.
  Mehrfachauswertung ist nur bei asymmetrischen Inkrementalgebern mit um 90 Grad versetzten Signalen A und B möglich.

  Hinweis

  Die Angabe 500 kHz bei der Zählfrequenz ist auf die maximale Frequenz der Signale A bzw. B bezogen. Bei Zweifachauswertung ergibt sich somit max. 1 MHz bei den Zählimpulsen, bzw. bei Vierfachauswertung maximal 2 MHz.

Diagnose der Signalpaare aktivieren

Überwachung der Signalpaare

Die Signalpaare A und /A, B und /B, N und /N können überwacht werden.

Aktivieren Sie dazu das entsprechende Optionskästchen.

Hinweis

Die "Diagnose N, /N" können Sie erst aktivieren, wenn Sie zuvor die "Diagnose A, /A und B, /B" aktiviert haben.

Zählrichtung auswählen

Zählrichtung

Zur Anpassung der Zählrichtung an den Prozess können Sie wählen:

- Zählrichtung normal
- Zählrichtung invertiert

Torfunktion auswählen

Beschreibung

Sie können bei der Parametrierung von Hard- und Software-Tor festlegen, ob das interne Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Zählvorgang abbrechen (Seite 206): Der Zählvorgang beginnt nach Torstopp und erneutem Torstart wieder von vorne.
- Zählvorgang unterbrechen (Seite 206): Der Zählvorgang wird nach Torstopp und erneutem Torstart beim letzten aktuellen Zählwert fortgesetzt.

Funktion DI

Funktion DI auswählen

Beschreibung

Mit "Funktion DI" wählen Sie die Funktion für den Digitaleingang aus:

- Eingang
  Der Digitaleingang funktioniert als "normaler" Eingang.
- HW-Tor (Seite 205), pegelgesteuert
  Das HW-Tor wird mit entsprechenden Signalen am Digitaleingang geöffnet und geschlossen.
- Synchronisation bei positiver Flanke (Seite 446)
  Mit positiven Flanken am Digitaleingang wird der Zähler auf den Ladewert gesetzt.
- HW-Freigabe für Synchronisation (Seite 447)
  Der Digitaleingang dient als HW-Freigabe. Bei aktiver HW-Freigabe wird mit der Nullmarke des Gebers der Zähler auf den Ladewert gesetzt.

Die Voreinstellung ist "Eingang".

Synchronisation bei positiver Flanke

Synchronisation

Wenn Sie "Synchronisation bei positiver Flanke" parametriert haben, dient die positive Flanke eines Referenzsignals am Eingang zum Setzen des 1Count5V auf den Startwert (= Ladewert).

Das folgende Bild zeigt die Zählerzustände bei Synchronisation mit positiver Flanke.

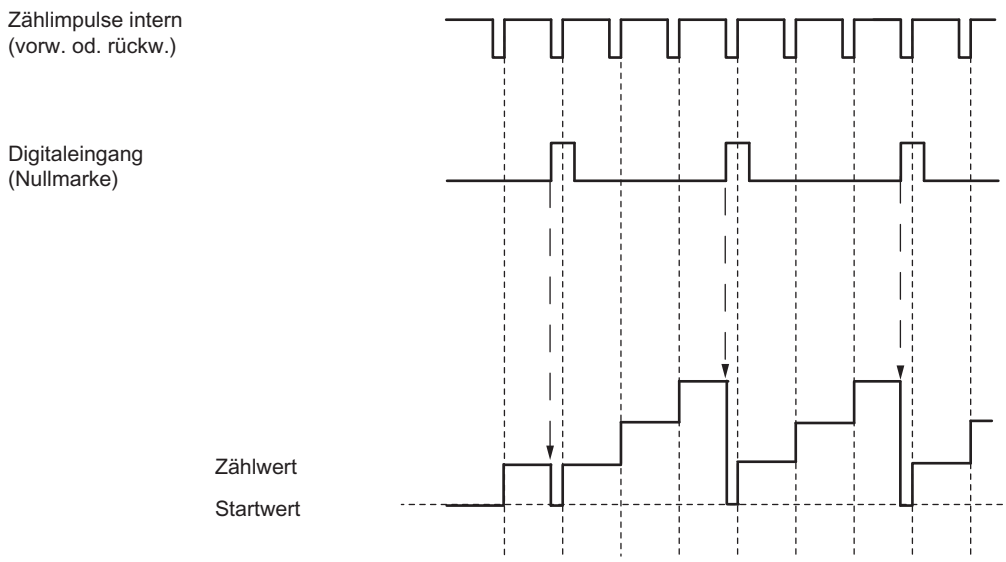

Es gibt folgende Bedingungen:

- Der Betriebsmodus "Fast Mode" muss aktiv (HW-Tor) sein.
  - Bei aktivierter Synchronisation lädt die erste und jede weitere Flanke das 1Count5V mit dem Startwert (= Ladewert).
- Als Referenzsignal kann das Signal eines prellfreien Schalters oder die Nullmarke eines Drehgebers dienen.
- Das Rückmeldebit STS_DI zeigt den Pegel des Referenzsignals an.

HW-Freigabe für Synchronisation

Synchronisation mit DI und Nullmarke

Wenn Sie "HW-Freigabe für Synchronisation" parametriert haben, dient der DI als HW-Freigabe. Wenn die HW-Freigabe aktiv ist, wird das 1Count5V durch die Nullmarke des Gebers mit dem Startwert (= Ladewert) geladen.

Das folgende Bild zeigt die Zählerzustände bei HW-Freigabe am DI und Synchronisation mit positiver Flanke.

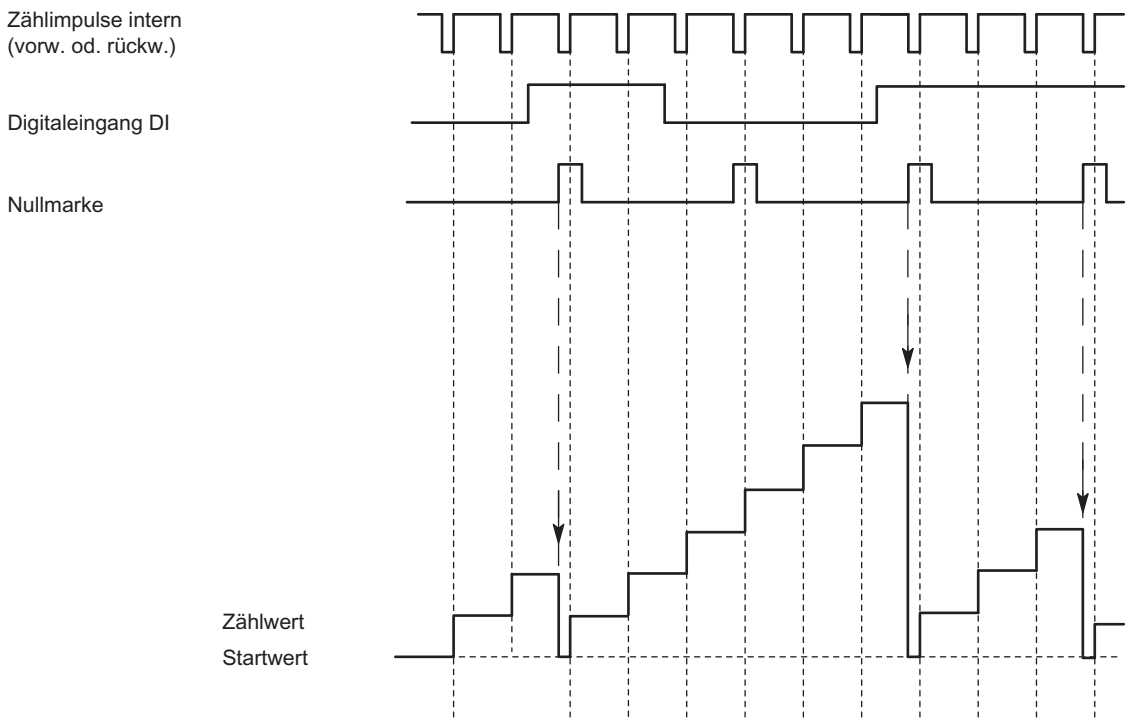

Es gibt folgende Bedingungen:

- Die Zählbetriebsart muss mit dem SW-Tor gestartet sein.
- Das Steuerbit "Freigabe Synchronisation" (CTRL_SYN) muss gesetzt sein.
- Nach erfolgreicher Synchronisation ist das Rückmeldebit (Seite 449) STS_SYN gesetzt und die LED SYN leuchtet. Durch das Steuerbit RES_STS wird das Rückmeldebit zurückgesetzt und die LED ausgeschaltet.
- Als Referenzsignal zur HW-Freigabe kann das Signal eines prellfreien Schalters dienen.
- Das Rückmeldebit (Seite 449) STS_DI zeigt den Pegel des Referenzsignals.

Im taktsynchronen Betrieb zeigt das gesetzte Rückmeldebit STS_SYN an, dass die positive Flanke am Digitaleingang zwischen dem Zeitpunkt T_i des aktuellen Zyklus und dem Zeitpunkt T_i des vergangenen Zyklus lag.

Eingangssignal HW-Tor auswählen

Eingangssignal HW-Tor

Der Pegel des Digitaleingangs ist bei der Parametrierung "Funktion DI = HW-Tor" mit dem Parameter "Eingangssignal HW-Tor" invertierbar.

Das Rückmeldebit (Seite 449) STS_DI zeigt Ihnen den Pegel des Digitaleingangs an.

Ladewert festlegen

Ladewert

Sie können dem 1Count5V einen Ladewert vorgeben.

Dieser Ladewert wird direkt als Startwert übernommen.

Zulässiger Wertebereich: -16777216 bis +16777215

Rückmeldeschnittstelle für Fast Mode

Hinweis

Für das 1Count5V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1Count5V für Fast Mode.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Bit 31	Lebenszeichen	LZ
	Bit 30	Taktsynchronen Betrieb aufgenommen	STS_TIC
	Bit 29	Parametrierfehler	ERR_PARA
	Bit 28	Sammelfehler • Kurzschluss Geberversorgung • Kurzschluss / Drahtbruch Gebersignal	EXTF
	Bit 27	Status DI	STS_DI
	Bit 26	Status Richtung vorwärts / rückwärts	STS_DIR
	Bit 25	Status internes Tor	STS_GATE
	Bit 0 bis 24	Zählwert	

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_PARA	Bei der Parametrierung des Moduls sind Parameter fehlerhaft.
EXTF	Sammelfehler Mögliche Ursachen: • Kurzschluss Geberversorgung • Kurzschluss oder Drahtbruch Gebersignal EXTF wird zurückgesetzt, wenn die Fehlerursachen beseitigt sind.

Bit	Erläuterung
LZ	Das Lebenszeichen wird bei jeder Aktualisierung der Rückmeldeschnittstelle getoggelt, d. h., der zuletzt gesendete Wert wird invertiert.
STS_DI	Das Bit zeigt den Status des Digitaleingangs DI an.
STS_DIR	Status Richtung: • bei Geberwertänderung von größeren zu kleineren Geberpositionen (inkl. Nulldurchgang) → "1" • bei Geberwertänderung von kleineren zu größeren Geberpositionen (inkl. Nulldurchgang) → "0"
STS_GATE	Status internes Tor: Es wird gezählt.
STS_TIC	Der taktsynchrone Betrieb (falls parametriert) wurde aufgenommen.

Diagnose des 1Count5V

Diagnose durch LED-Anzeige

LED-Anzeige am 1Count5V

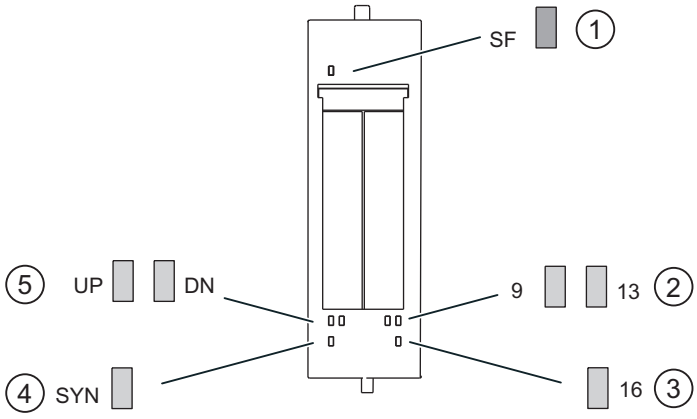

① Sammelfehler (rot)
② Statusanzeige für Digitalausgang (grün)
③ Statusanzeige für Digitaleingang (grün)
④ Statusanzeige für Synchronisation (grün)
⑤ Statusanzeige Zählrichtung (grün)

Status- und Fehleranzeigen durch LEDs am 1Count5V

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 1Count5V.

Ereignis (LED)							Ursache	Maßnahme
SF	UP	DN	SYN	9	13	16		
ein							Keine Parametrierung. Diagnosemeldung liegt vor.	Überprüfen Sie die Parametrierung. Werten Sie die Diagnose aus.
	ein						Status des niederwertigen Bits des Zählers, falls der Zähler vorwärts zählt.	
		ein					Status des invertierten niederwertigen Bits des Zählers, falls der Zähler rückwärts zählt.	
			ein				Synchronisation ist ausgeführt (nur in den Zählbetriebsarten; Abbild vom Rückmeldebit STS_SYN).	
				ein			DO 1 ist aktiviert	
					ein		DO 2 ist aktiviert.	
						ein	DI ist aktiviert.	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

1Count5V Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 1Count5V.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Geberversorgung oder des Aktors.	Überprüfen Sie die Verdrahtung zum Geber. Korrektur der Prozessverdrahtung.
5_D	00101: Übertemperatur	Digitalausgang ist überlastet.	Korrektur der Prozessverdrahtung.
6_D	00110: Leitungsbruch	Leitung zum Aktor unterbrochen.	Korrektur der Prozessverdrahtung.
9_D	01001: Fehler	Interner Modulfehler ist aufgetreten.	Austausch des Moduls.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.
26_D	11010: externer Fehler	Leitungsbruch / Kurzschluss der 5 V-Gebersignale: A, /A, B, /B, N, /N,	Korrektur der Parametrierung

Nockensteuern (S7-300, S7-400) 2

2.1 Nockensteuern

2.1.1 Grundlagen zum Nockensteuern

2.1.1.1 Was ist Nockensteuern?

Nockensteuern

Nockensteuerwerke werden eingesetzt, um positions- oder zeitabhängige Funktionen anzusteuern. Sie sind den mechanischen Nockenschaltwerken vor allem durch ihre hohe Flexibilität weit überlegen, z. B. können während des laufenden Betriebs Änderungen per Software vorgenommen werden.

Ein Nocken erzeugt positionsabhängige Schaltsignale abhängig von Positionswerten der Achse oder eines externen Gebers. Unterschieden werden Wegnocken und Zeitnocken.

Ein Nocken ist definiert über:

- Anfangsposition und Endposition (Wegnocken (Seite 453))
- Anfangsposition und Einschaltdauer (Zeitnocken (Seite 453))

2.1.1.2 Eigenschaften der Nockenarten

Nockenarten

Sie können jeden Nocken als Wegnocken oder als Zeitnocken parametrieren.

Die folgende Tabelle stellt die Eigenschaften der beiden Nockenarten gegenüber.

	Wegnocken	Zeitnocken
Darstellung	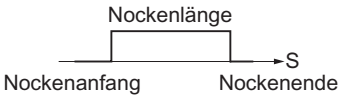	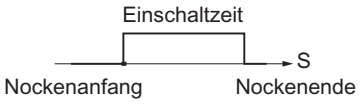
Parametrierung	Parametriert wird: • Nockenanfang • Nockenende • Wirkrichtung • Vorhaltezeit	Parametriert wird: • Nockenanfang • Einschaltzeit • Wirkrichtung • Vorhaltezeit

	Wegnocken	Zeitnocken
Wirkrichtung	Zwei Wirkrichtungen sind möglich: • positiv: Der Nocken wird am Nockenanfang eingeschaltet, wenn die Achse in Richtung steigender Istwerte läuft. • negativ: Der Nocken wird am Nockenende eingeschaltet, wenn die Achse in Richtung fallender Istwerte läuft. Sie können auch beide Wirkrichtungen gleichzeitig einschalten.	Zwei Wirkrichtungen sind möglich: • positiv: Der Nocken wird am Nockenanfang eingeschaltet, wenn die Achse in Richtung steigender Istwerte läuft. • negativ: Der Nocken wird am Nockenanfang eingeschaltet, wenn die Achse in Richtung fallender Istwerte läuft. Sie können auch beide Wirkrichtungen gleichzeitig einschalten.
Einschalten	Der Nocken schaltet ein: • am Nockenanfang, wenn die Bewegungsrichtung der Achse positiv ist und positive Wirkrichtung eingestellt ist. • am Nockenende, wenn die Bewegungsrichtung der Achse negativ ist und negative Wirkrichtung eingestellt ist. • wenn der Istwert im Bereich des Nockens liegt.	Der Nocken schaltet ein: • am Nockenanfang, wenn die Bewegungsrichtung der Achse positiv ist und positive Wirkrichtung eingestellt ist bzw. wenn die Bewegungsrichtung der Achse negativ ist und negative Wirkrichtung eingestellt ist Nach dem Einschalten läuft die volle Nockeneinschaltzeit ab. Auch dann, wenn nach dem Einschalten des Nockens die Bewegungsrichtung der Achse verändert wird. Wenn während der Einschaltzeit eines Nockens dessen Nockenanfang erneut überfahren wird, wird der Nocken **nicht** nachgetriggert.
Ausschalten	Der Nocken schaltet aus, wenn • der parametrierte Weg überfahren wurde, • die Wirkrichtung entgegen der Bewegungsrichtung der Achse erkannt ist und keine Hysterese parametriert ist, • der Istwert nicht mehr im Bereich des Nockens liegt.	Der Nocken schaltet aus, wenn die parametrierte Zeit abgelaufen ist.
Weglänge	Die Weglänge des Nockens ist durch Nockenanfang und Nockenende definiert. Nockenanfang und Nockenende gehören zum aktiven Teil des Nockens.	Die Weglänge des Nockens ist abhängig von der Geschwindigkeit, mit der die Achse während der Einschaltzeit des Nockens verfährt.
Einschaltdauer	Die Einschaltdauer des Nockens ist abhängig von der Geschwindigkeit, mit der die Achse die Weglänge des Nockens überfährt.	Die Einschaltdauer des Nockens ist mit der Einschaltzeit parametriert.

Richtungserkennung

Die Bewegungsrichtung der Achse wird folgendermaßen bestimmt:

- Mit jedem Impuls beim Inkrementalgeber.
- Mit jedem fehlerfreien Telegramm beim SSI-Geber.

Inverser Nocken

Bei "Nockenflanken ändern" kann kurzzeitig ein inverser Nocken entstehen, wenn der neue Nockenanfang größer ist als das alte Nockenende. Die folgende Tabelle zeigt die Wirkung eines inversen Nockens bei einer Linear- und einer Rundachse.

Inverser Nocken bei einer Linearachse	Inverser Nocken bei einer Rundachse
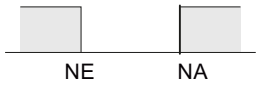	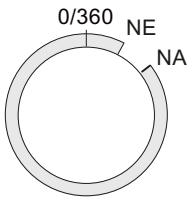
Nockenanfang ist größer ist als das Nockenende	Nockenanfang ist positiver als das Nockenende
Bei beiden Achstypen muss zwischen Nockenanfang und Nockenende ein Mindestabstand von 4 Impulsen eingehalten werden.	

2.1.1.3 Normalspuren und Spurergebnisse

Nockenspuren

Mit den 32 Spuren können Sie maximal 32 unterschiedliche Schaltvorgänge steuern. Die Spuren können Sie mit Hilfe der Rückmeldesignale auswerten.

Den ersten 13 Spuren (Spur 0 bis 12) ist je ein Digitalausgang (Q0 bis Q12) der FM 352 zugeordnet, der z. B. ein angeschlossenes Schütz direkt steuern kann.
Bei der FM 452 ist den ersten 16 Spuren (Spur 0 bis 15) je ein Digitalausgang (Q0 bis Q15) zugeordnet.

Spurergebnis

Sie haben maximal 128 Nocken zur Verfügung, die Sie beliebigen Spuren zuordnen können.

Jeder Spur können mehrere Nocken zugeordnet werden. Das Spurergebnis ist die logische ODER-Verknüpfung aller Nockenwerte dieser Spur.

Beispiel für ein Spurergebnis

Sie legen bei der Parametrierung folgende Nocken für Spur 3 fest:

Nocken	Nockenanfang	Nockenende
1	101 µm	106 µm
2	100 µm	104 µm

Das folgende Bild zeigt das daraus resultierende Spurergebnis.

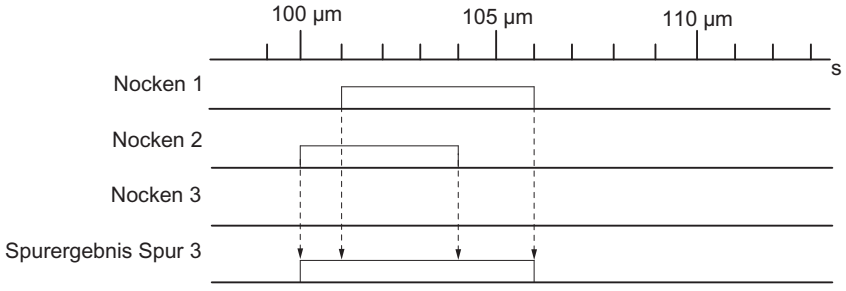

Spurfreigabe

Damit die Spurergebnisse der Spuren 0 bis 12 als Spursignal auf die Digitalausgänge Q0 bis Q12 der FM 352 gelegt werden können, müssen Sie die verwendeten Spuren freischalten. Bei der FM 452 stehen die Spuren 0 bis 15 zur Verfügung.

Externe Freigabe einer Spur

Sie können eine externe Freigabe der Spur 3 in den Maschinendaten parametrieren. Das Spursignal 3 wird dann noch mit dem Digitaleingang I3 UND-verknüpft, bevor es den Digitalausgang Q3 der FM 352 schalten kann.

Der Digitalausgang Q3 wird also dann geschaltet, wenn die nachfolgenden Bedingungen erfüllt sind:

- Die zugehörige Spur ist freigeschaltet.
- Mindestens ein Nocken ist auf dieser Spur aktiv (Spurergebnis = 1).
- Durch ein externes Ereignis wurde der zugehörige Digitaleingang I3 gesetzt.

Bei der FM 452 stehen die Spuren 3 bis 10 für eine externe Freigabe zur Verfügung.

Setzen der Spursignale

Die Spursignale 0 bis 12 (entsprechend den Digitalausgängen Q0 bis Q12) können über das Nockensteuerwerk oder die CPU gesetzt werden. Bei der FM 452 können die Spursignale 0 bis 15 gesetzt werden.

2.1.1.4 Sonderspuren

Definition

Sie haben die Möglichkeit, die Spuren 0 ... 2 als Sonderspuren zu parametrieren:

- Spur 0 oder 1: Zählnockenspur
- Spur 2: Bremsnockenspur
  Damit die Spur eingeschaltet werden kann, wird der Eingang I0 ausgewertet.

Voraussetzung

Für die Arbeit mit den Sonderspuren gelten folgende Voraussetzungen:

- Auf der Spur sind Nocken parametriert.
- Die Nockenbearbeitung ist eingeschaltet.
- Die entsprechende Spur ist freigeschaltet.
- Die Spur ist als Sonderspur parametriert.

Zählnockenspur

Eine Zählnockenspur zählt die Zustandswechsel der Spurergebnisse auf dieser Spur.

Sie müssen einen oberen Zählwert für die Zählnockenspur angeben und die Zählfunktion starten.

Jede steigende Flanke des Spurergebnisses dekrementiert den Zählerwert der entsprechenden Spur um 1.

Solange der Zählwert der Zählnockenspur nicht 0 ist, bleibt das Spurkennbit 0.

Erreicht der Zählwert den Wert 0, wird das Spurkennbit gesetzt und bei entsprechender Parametrierung das Spursignal (Seite 460) gesetzt.

Mit der nächsten fallenden Flanke des Spurergebnisses (alle Nocken auf dieser Spur sind aus), wird das Spurkennbit wieder ausgeschaltet und der Zähler wieder auf den vorgegebenen Zählwert gesetzt.

Das folgende Bild zeigt das Schalten einer Zählnockenspur.

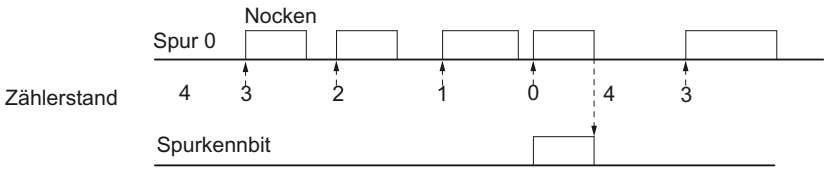

In den Maschinendaten parametrierter oberer Zählwert = 4

Bremsnockenspur

Für den Einsatz der Spur 2 als Bremsnockenspur muss der Digitaleingang I0 verschaltet sein.

Mit der steigenden Flanke des Signals an I0 schaltet das Spurkennbit sofort ein.

Das Spurkennbit wird wieder ausgeschaltet, wenn

- an I0 kein "1"-Signal mehr anliegt **und** danach
- die fallende Flanke des Spurergebnisses der Spur 2 erkannt wird.

Das folgende Bild zeigt das Schalten einer Bremsnockenspur.

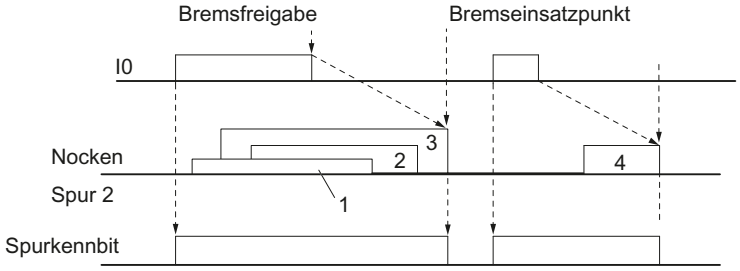

1 ... 4 kennzeichnen 4 Nocken, die auf die Bremsnockenspur wirken

Im Beispiel wird das Spurkennbit durch die fallenden Flanken der Nocken 3 bzw. 4 ausgeschaltet.

2.1.1.5 Hysterese

Definition

Eine mechanische Unruhe an der Achse kann Lageistwertänderungen hervorrufen. Schwankt der Lageistwert dabei um eine Flanke eines Nockens oder innerhalb eines aktiven Nockens mit nur einer Wirkrichtung, dann wird dieser Nocken permanent ein- und ausgeschaltet. Durch die Hysterese wird dieses Schalten unterbunden.

Eine eingestellte Hysterese ist istwertabhängig und gilt bei allen Nocken. Sie wird aktiv, sobald ein Richtungswechsel erkannt wurde. Eine Hysterese wirkt auch, wenn kein Nocken an der aktuellen Achsposition gesetzt ist.

Regeln für den Hysteresebereich

Für den Hysteresebereich gelten die folgenden Regeln:

- Die Hysterese wird immer bei einem Richtungswechsel aktiv geschaltet.
- Innerhalb der Hysterese bleibt die Anzeige des Istwerts konstant.
- Innerhalb der Hysterese wird die Richtung nicht neu bestimmt.
- Innerhalb der Hysterese wird ein Wegnocken weder ein- noch ausgeschaltet.
- Innerhalb der Hysterese wird ein Zeitnocken nicht eingeschaltet, ein eingeschalteter Zeitnocken wird nach Ablauf der parametrierten Einschaltzeit ausgeschaltet (nicht erst am Ende der Hysterese).

- Nach Verlassen des Hysteresebereichs bestimmt die FM x52:
  - den Lageistwert,
  - die aktuelle Bewegungsrichtung der Achse,
  - die aktuellen Zustände aller Nocken.
- Der Hysteresebereich ist für alle Nocken gültig.

Richtungsumkehr auf einem Nocken mit Hysterese

Die folgende Tabelle zeigt beispielhaft das Verhalten bei einer Richtungsumkehr auf einem Nocken. Sie müssen dabei das Verhalten eines Wegnockens und das eines Zeitnockens unterscheiden. Die Wirkrichtung der Nocken ist **positiv**.

Wegnocken	Zeitnocken
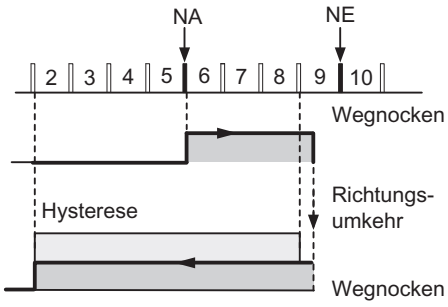	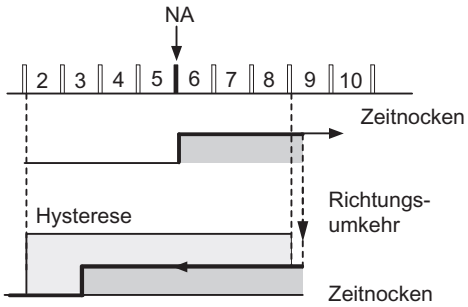
Die Hysterese wird nach Erkennen des Richtungswechsels aktiv. Der Nocken wird ausgeschaltet, sobald die Hysterese verlassen wird.	Der Nocken bleibt immer für die Dauer der parametrierten Einschaltzeit eingeschaltet.
Nocken Hysterese	

2.1.1.6 Dynamische Verstellung

Aufgabe

Die dynamische Verstellung dient zur Kompensation von Verzögerungszeiten der angeschlossenen Schaltelemente.

Vorhaltezeit

Die Verzögerungszeit können Sie als Vorhaltezeit parametrieren, die Sie für jeden einzelnen Nocken angeben. Sie können pro Nocken eine Vorhaltezeit vergeben. Die Vorhaltezeit gilt für Nockenanfang und Nockenende.

Vorhalteweg

Der Vorhalteweg eines Nockens wird in Abhängigkeit von der aktuellen Geschwindigkeit und der Vorhaltezeit laufend neu berechnet. Um diesen Weg wird der gesamte Nocken in Richtung des Istwerts verschoben. Der parametrierte Bereich wird als "statischer Bereich", der aufgrund der Vorhaltezeit berechnete Bereich als "dynamischer Bereich" bezeichnet.

Vorhalteweg = Vorhaltezeit × aktuelle Geschwindigkeit der Achse

Die Berechnung der Vorhaltewege aller Nocken wird innerhalb 1/4 der längsten parametrierten Vorhaltezeit in der FM x52 durchgeführt.

Wenn Sie für einen Nocken eine sehr große Vorhaltezeit parametrieren, reduzieren Sie die Dynamik der Nockenbearbeitung.

2.1.1.7 Schnittstellen des Nockensteuerwerks

Übersicht

Das folgende Bild zeigt Ihnen in einer prinzipiellen Darstellung am Beispiel der FM 352 die wesentlichen Schnittstellen des Nockensteuerwerks, um den Zusammenhang von Daten, Eingängen und Ausgängen zu verdeutlichen. Differenzen zur FM 452 entnehmen Sie der Tabelle unten.

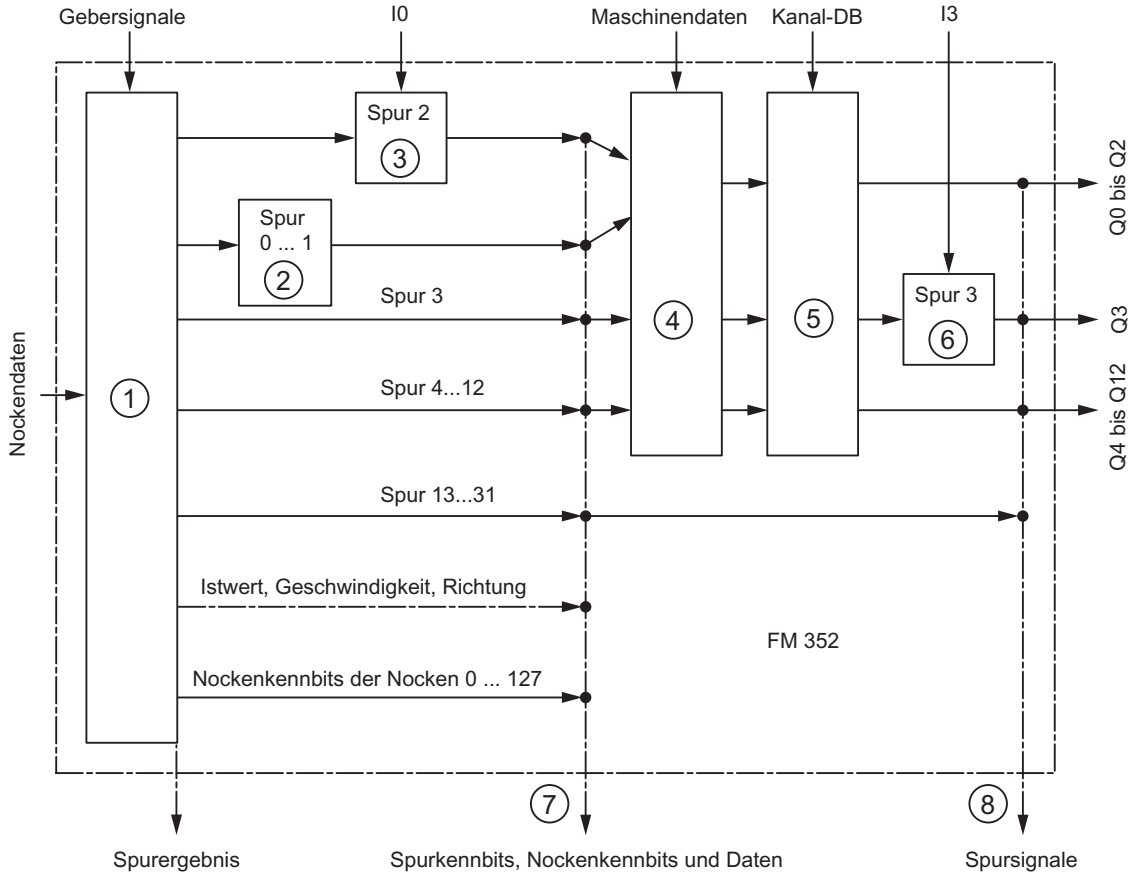

Nr.	Beschreibung
①	In der Nockenbearbeitung der FM x52 werden die Nockenkennbits aus den Schaltbedingungen und dem aktuellen Istwert errechnet. Außerdem werden die Spurergebnisse aufgrund der Zuordnung der Nocken zu den Spuren ermittelt.
②	Wenn Sie die Spur 0 oder 1 als Zählnockenspur parametriert haben, wird das Spurergebnis des Nockensteuerwerks (Punkt 1) mit dem Zählergebnis zum Spurkennbit verknüpft. Ansonsten ist das Spurkennbit gleich dem Spurergebnis.
③	Wenn Sie die Spur 2 als Bremsnockenspur parametriert haben, wird das Spurergebnis des Nockensteuerwerks (Punkt 1) mit dem Eingang I0 zum Spurkennbit verknüpft. Ansonsten ist das Spurkennbit gleich dem Spurergebnis.
④	Über Maschinendaten können Sie steuern, ob die bisher ermittelten Spurkennbits der Spuren 0 bis 12 (bei FM 352) bzw. 0 bis 15 (bei FM 452) des Nockensteuerwerks an den Kanal-DB weitergereicht werden oder unabhängig davon direkt über die Spurfreigabe (TRACK_EN) gesetzt werden.
⑤	Die Spursignale der Spuren 0 bis 12 (bzw. 0 bis 15) geben Sie über TRACK_EN, die Zählfunktion über CNTC0_EN / CNTC1_EN frei.
⑥	FM 352: Das Spursignal der Spur 3 kann mit dem Digitaleingang I3 UND-verknüpft werden, wenn Sie diese Möglichkeit in den Maschinendaten (EN_IN_I3) parametriert haben.
	FM 452: Die Spursignale der Spuren 3 bis 10 können mit den Digitaleingängen I3 bis I10 UND-verknüpft werden, wenn Sie diese Möglichkeit in den Maschinendaten (EN_IN_I3 bis EN_IN_I10) parametriert haben.
⑦	Alle Spur- und Nockenkennbits können Sie an dieser Stelle (also vor der Verknüpfung mit den Maschinen- und Kanaldaten) mit dem Auftrag ACTPOS_EN bzw. CAMOUT_EN auslesen. Für die Spuren 3 bis 31 ist das Spurkennbit gleich dem Spurergebnis (Punkt 1).
⑧	FM 352: Nach der Verknüpfung mit den Maschinendaten (z. B. Achsart, Software-Endschalter, Weg pro Geberumdrehung, …) und Kanaldaten stehen die Spursignale der Spuren 0 bis 12 in den Rückmeldesignalen zur Verfügung. Die Spursignale der Spuren 13 bis 31 sind identisch mit den Spurkennbits von Punkt 7. Die Spursignale der Spuren 0 bis 12 stehen auch an den Digitalausgängen Q0 bis Q12 zur Verfügung.
	FM 452: Nach der Verknüpfung mit den Maschinen- und Kanaldaten stehen die Spursignale der Spuren 0 bis 15 in den Rückmeldesignalen zur Verfügung. Die Spursignale der Spuren 16 bis 31 sind identisch mit den Spurkennbits von Punkt 7. Die Spursignale der Spuren 0 bis 15 stehen auch an den Digitalausgängen Q0 bis Q15 zur Verfügung.

2.1.1.8 Gebersignale

Anschließbare Geber

Anschließbare Inkrementalgeber

Es werden Inkrementalgeber (Seite 463) mit zwei um 90° versetzten Impulsen mit oder ohne Nullmarke unterstützt:

- Geber mit asymmetrischen Ausgangssignalen 24 V-Pegel
  - Grenzfrequenz = 50 kHz
  - Max. 100 m Leitungslänge geschirmt
- Geber mit symmetrischen Ausgangssignalen mit 5 V-Differenzschnittstelle nach RS 422
  - Grenzfrequenz = 1 MHz
  - Bei 5 V-Spannungsversorgung: max. 32 m Leitungslänge geschirmt
  - Bei 24 V-Spannungsversorgung: max. 100 m Leitungslänge geschirmt

Hinweis

Wenn der 5 V-Geber kein Nullmarkensignal ausgibt und Sie die Drahtbruchüberwachung aktiviert haben, dann müssen Sie die Nullmarkeneingänge N und /N extern verschalten, damit diese Eingänge einen unterschiedlichen Pegel aufweisen (z. B. N auf 5 V, /N auf Masse).

Anschließbare Initiatoren

Folgende Initiatoren (Seite 465) können Sie an die FM x52 anschließen:

- Initiatoren mit 24 V-Pegel (Näherungsschalter)
- Grenzfrequenz = 50 kHz
- maximal 100 m Leitungslänge

Anschließbare Absolutwertgeber

Es werden Absolutwertgeber (Seite 465) mit serieller Schnittstelle unterstützt. Die Übertragung der Weginformation erfolgt synchron nach dem SSI-Protokoll (**s**ynchron-**s**erielles-**I**nterface). Die FM x52 unterstützt nur GRAY-Code. Durch die Anordnung der Datenbits im Telegrammrahmen ergeben sich die Datenformate "Tannenbaum", "halber Tannenbaum" und "rechtsbündig".

Absolutwertgeber werden unterschieden in

- Singleturn-Geber
  Singleturn-Geber bilden den gesamten Messbereich auf eine Umdrehung des Gebers ab.
- Multiturn-Geber
  Multiturn-Geber bilden den gesamten Messbereich auf mehreren Umdrehungen des Gebers ab.

Geberart	Telegrammlänge / -art
Singleturn-Geber	13 Bit halber Tannenbaum
Singleturn-Geber	13 Bit rechtsbündig
Singleturn-Geber	25 Bit rechtsbündig
Multiturn-Geber	25 Bit Tannenbaum
Multiturn-Geber	25 Bit rechtsbündig
Mithören	Tannenbaum
Mithören	rechtsbündig
Sondereinstellung: Multiturn-Geber als Singleturn-Geber	 25 Bit halber Tannenbaum

Die Baudrate zur Datenübertragung ist abhängig von der Leitungslänge:

- Max. 125 kHz bei 320 m Leitungslänge geschirmt
- Max. 250 kHz bei 160 m Leitungslänge geschirmt
- Max. 500 kHz bei 60 m Leitungslänge geschirmt
- Max. 1 MHz bei 20 m Leitungslänge geschirmt

Inkrementalgeber

Signalformen

Das folgende Bild zeigt die Signalformen von Gebern mit asymmetrischen und symmetrischen Ausgangssignalen.

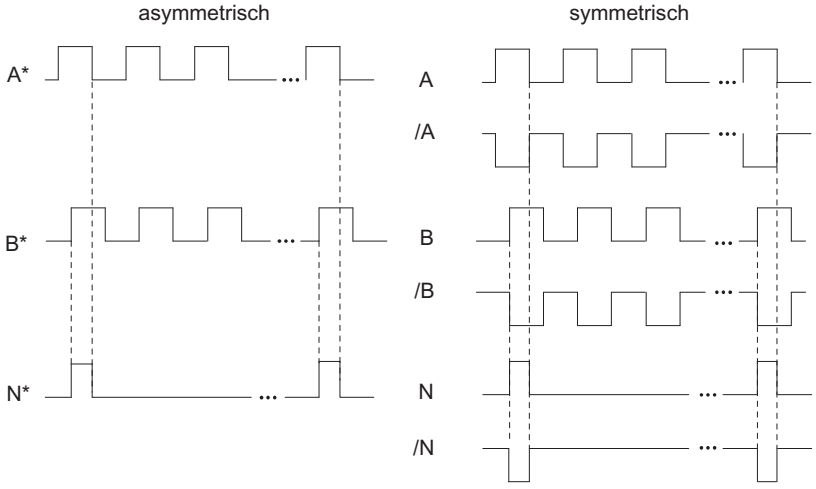

Signalauswertung

Inkremente

Ein Inkrement kennzeichnet eine Signalperiode der beiden Signale A und B eines Gebers. Dieser Wert wird in den Technischen Daten eines Gebers und/oder auf dessen Typenschild angegeben.

Das folgende Bild zeigt das Verhältnis von Inkrementen und Impulsen.

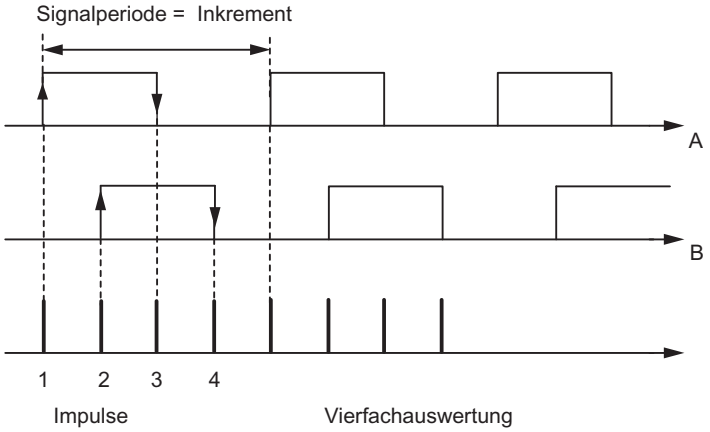

Impulse

Die FM x52 wertet alle 4 Flanken der Signale A und B in jedem Inkrement aus (Vierfachauswertung).

1 Inkrement (Gebervorgabe) = 4 Impulse (FM-Auswertung)

Reaktionszeiten

Die FM x52 hat für angeschlossene Inkrementalgeber folgende Reaktionszeiten:

- Minimale Reaktionszeit = Nockenzykluszeit + Schaltzeit der angeschlossenen Schaltelemente
- Maximale Reaktionszeit = 2 × Nockenzykluszeit + Schaltzeit der angeschlossenen Schaltelemente

Beispiel Reaktionszeiten

Ein Beispiel für die minimale und maximale Reaktionszeit bei einem Mengengerüst von 16 Nocken:

- Nockenzykluszeit: ca. 20 µs
- Schaltzeit der Hardware: ca. 150 µs

Minimale Reaktionszeit = 20 µs + 150 µs = 170 µs

Maximale Reaktionszeit = 2 × 20 µs + 150 µs = 190 µs

Hinweis

Die Reaktionszeit können Sie durch entsprechendes Parametrieren der Nocken bzw. mit dynamischer Verstellung kompensieren.

Unschärfe

Die Unschärfe ist die Differenz aus maximaler und minimaler Reaktionszeit. Bei Inkrementalgebern beträgt sie

Unschärfe = Nockenzykluszeit

Hinweis

Wenn die Schaltzeit der Hardware auf der FM x52 und die Schaltzeit der angeschlossenen Schaltelemente vernachlässigt werden können, dann ist ein sicheres Schalten des Nockens gewährleistet, wenn der Nocken länger ist als der Weg, der innerhalb der Nockenzykluszeit verfahren wird.

Initiatoren

Definition

Initiatoren sind einfache Schalter ohne Richtungsinformation, die Impulse abgeben. Die Richtung geben Sie mit dem Maschinendatum "Richtung Initiator" vor.

⚠ VORSICHT

Es kann zu Sachschäden kommen.

Eine falsche Richtungsvorgabe kann zu schwerwiegenden Fehlern in der Anlage führen (z. B. durch fehlerhafte Ansteuerung von Aggregaten).

Kontrollieren Sie die Richtungsvorgabe bei der Inbetriebnahme und bei jedem Tausch eines Initiators.

Signalauswertung

Bei einem Initiator wird die steigende Flanke des Signals A* gezählt.

Absolutwertgeber (SSI)

Impulsauswertung Absolutwertgeber

1 Inkrement (Gebervorgabe) = 1 Impuls (FM-Auswertung)

Mithören

"Mithören" bedeutet: Ein Absolutwertgeber wird parallel an zwei Baugruppen (z. B. FM x51 und FM x52) betrieben. Die Positionierbaugruppe FM x51 ist Master und taktet den Absolutwertgeber, das Elektronische Nockensteuerwerk FM x52 ist Slave und hört die Signale des SSI-Telegramms mit.

Stellen Sie "Inkremente/Geberumdrehung" und "Anzahl Umdrehungen" auf die Mastereinstellung ein. Die "Baudrate" ist nicht relevant. Bei "Telegrammlänge" wählen Sie je nach Gebertyp "Mithören" bzw. "Mithören rechtsbündig".

Reaktionszeiten

Die FM x52 hat für Absolutwertgeber folgende Reaktionszeiten:

- Minimale Reaktionszeit = Telegrammlaufzeit + Nockenzykluszeit + Schaltzeit der angeschlossenen Schaltelemente
- Maximale Reaktionszeit = 2 × Telegrammlaufzeit + Monoflopzeit + 2 × Nockenzykluszeit + Schaltzeit der angeschlossenen Schaltelemente
- Bei programmierbaren Absolutwertgebern:
  Maximale Reaktionszeit = Telegrammlaufzeit + Monoflopzeit + 2 × Nockenzykluszeit + Schaltzeit der angeschlossenen Schaltelemente +1/max. Schrittfolgefrequenz

Monoflopzeit

Für die Monoflopzeit gelten folgende Grenzwerte:

- minimale Monoflopzeit: > 15 µs
- maximale Monoflopzeit: < 64 µs

Geber mit Werten außerhalb der hier genannten Grenzen sind nicht zulässig.

Telegrammlaufzeiten

Die Telegrammlaufzeiten sind abhängig von der Baudrate:

Baudrate	Telegrammlaufzeit bei 13 Bit	Telegrammlaufzeit bei 25 Bit
0,125 MHz	112 µs	208 µs
0,250 MHz	56 µs	104 µs
0,500 MHz	28 µs	52 µs
1,000 MHz	14 µs	26 µs

Beispiel Reaktionszeiten

Dieses Beispiel zeigt Ihnen, wie Sie die minimale und maximale Reaktionszeit berechnen. Im Beispiel wird kein programmierbarer Geber verwendet.

- Nockenzykluszeit: ca. 20 µs bei max. 16 Nocken
- Schaltzeit der Hardware: ca. 150 µs
- Telegrammlaufzeit: 26 µs bei 1 MHz Baudrate (25 Bit-Telegrammrahmen)
- Monoflopzeit: 20 µs (abhängig vom Geber: typisch 20 bis 40 µs)

Minimale Reaktionszeit = 26 µs + 20 µs + 150 µs = 196 µs

Maximale Reaktionszeit = 2 × 26 µs + 20 µs + 2 × 20 µs + 150 µs = 262 µs

Hinweis

Die Reaktionszeit können Sie durch entsprechendes Parametrieren der Nocken bzw. mit dynamischer Verstellung kompensieren.

Unschärfe

Die Unschärfe ist die Differenz aus maximaler und minimaler Reaktionszeit.

Bei einem Absolutwertgeber beträgt sie:

- Unschärfe = Nockenzykluszeit + Telegrammlaufzeit + Monoflopzeit

Bei programmierbaren Absolutwertgebern beträgt sie:

- Unschärfe = Nockenzykluszeit + Telegrammlaufzeit + Monoflopzeit + 1/max. Schrittfolgefrequenz

Hinweis

Wenn die Schaltzeit der Hardware auf der FM x52 und die Schaltzeit der angeschlossenen Schaltelemente vernachlässigt werden können, dann ist ein sicheres Schalten des Nockens gewährleistet, wenn der Nocken länger ist als der Weg, der innerhalb der Nockenzykluszeit verfahren wird.

2.1.2 Funktionen der FM x52

2.1.2.1 Maschinendaten schreiben und aktivieren

Maschinendaten schreiben und aktivieren

Mit den Maschinendaten passen Sie die FM x52 an die Achse und den Geber an.

Über das Anwenderprogramm können Sie die Parameter während des Betriebs ändern.

Alle Parameter sind im Parameter-DB abgelegt:

- Maschinendaten liegen im Parameter-DB auf den Adressen 3.1 bis 104.0.

Die Nummer des Parameter-DB müssen Sie in den jeweils zugehörigen Kanal-DB eintragen.

Die Parameter können Sie mit dem DB-Editor eingeben oder auch komfortabel in den Dialogen "Maßsystem", "Achse", "Geber" und "Spuren" eintragen und mit der Funktion "Export" in den Parameter-DB schreiben.

Die Parameter in einem bereits vorhandenen Parameter-DB können Sie mit der Funktion "Import" in die Dialoge importieren.

Erstparametrierung

Falls die Baugruppe noch keine Maschinendaten enthält (Rückmeldesignal PARA = 0), gehen Sie bei einer Erstparametrierung ohne die Parametrierdialoge folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie das folgende Anstoßbit im Kanal-DB:
   – Maschinendaten schreiben (MDWR_EN = 1)
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.

Maschinendaten ändern

Um vorhandene Maschinendaten (Rückmeldesignal PARA = 1) per Anwenderprogramm zu ändern, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Setzen Sie die Anstoßbits im Kanal-DB:
   – Maschinendaten schreiben (MDWR_EN = 1)
   – Maschinendaten aktivieren (MD_EN = 1)
3. Rufen Sie im zyklischen Anwenderprogramm die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.
4. Überprüfen Sie, ob die bisherigen Nockendaten mit den geänderten Maschinendaten verträglich sind.
5. Schreiben Sie in jedem Fall die Nockendaten der parametrierten Nocken - geändert oder ungeändert - neu (CAM1WR_EN … CAM8WR_EN).

Wenn Sie die Anstoßbits gleichzeitig setzen, sorgt die Anweisung CAM_CTRL bzw. CAM_CTRL_452 dafür, dass die Aufträge in der richtigen Reihenfolge abgearbeitet werden.

Ansonsten ändern Sie die Maschinendaten immer in folgender Reihenfolge:

1. Maschinendaten schreiben
2. Maschinendaten aktivieren

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren

Hinweis

Falls synchronisationsrelevante Parameter verändert wurden, wird beim Aktivieren der Maschinendaten die Synchronisation gelöscht. Außerdem werden die Funktionen zurückgesetzt und alle Maschinen- und Nockendaten auf der Baugruppe gelöscht.

Synchronisationsrelevante Parameter sind:

- Achsart
- Ende der Rundachse
- Signalart
- Weg pro Geberumdrehung
- Inkremente pro Geberumdrehung
- Anzahl Umdrehungen
- Referenzpunktkoordinate
- Absolutwertgeberjustage
- Art des Referenzpunktnachtriggerns
- Richtungsanpassung
- Mengengerüst
- Software-Endschalter Anfang und Ende

2.1.2.2 Maschinendaten lesen

Maschinendaten lesen

Um die aktuellen Maschinendaten von der Baugruppe zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie das folgende Anstoßbit im Kanal-DB:
   – Maschinendaten lesen (MDRD_EN = 1)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.

Damit werden die aktuellen Maschinendaten im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.1	MDRD_EN	BOOL	FALSE	1 = Maschinendaten lesen

2.1.2.3 Nockendaten schreiben

Nockendaten schreiben

Mit den Nockendaten geben Sie Art und Wirkungsweise der Nocken sowie deren Zuordnung zu den Spuren vor.

Nockendaten liegen im Parameter-DB ab Adresse 108.0. Sie sind in Paketen zu je 16 Nocken zusammengefasst.

Nockendaten sind nach dem Schreiben sofort aktiv.

Um Nockendaten ohne die Parametrierdialoge zu schreiben, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie die Anstoßbits im Kanal-DB (CAM1WR_EN … CAM8WR_EN)
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.3	CAM1WR_EN	BOOL	FALSE	1 = Nockendaten 1 schreiben (Nocken 0 bis 15)
35.4	CAM2WR_EN	BOOL	FALSE	1 = Nockendaten 2 schreiben (Nocken 16 bis 31)
35.5	CAM3WR_EN	BOOL	FALSE	1 = Nockendaten 3 schreiben (Nocken 32 bis 47)
35.6	CAM4WR_EN	BOOL	FALSE	1 = Nockendaten 4 schreiben (Nocken 48 bis 63)
35.7	CAM5WR_EN	BOOL	FALSE	1 = Nockendaten 5 schreiben (Nocken 64 bis 79)
36.0	CAM6WR_EN	BOOL	FALSE	1 = Nockendaten 6 schreiben (Nocken 80 bis 95)
36.1	CAM7WR_EN	BOOL	FALSE	1 = Nockendaten 7 schreiben (Nocken 96 bis 111)
36.2	CAM8WR_EN	BOOL	FALSE	1 = Nockendaten 8 schreiben (Nocken 112 bis 127)

2.1.2.4 Nockendaten lesen

Nockendaten lesen

Um die aktuellen Nockendaten von der Baugruppe zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie das folgende Anstoßbit im Kanal-DB:
   – Nockendaten lesen (CAM1RD_EN … CAM8RD_EN)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.

Damit werden die aktuellen Nockendaten im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.2	CAM1RD_EN	BOOL	FALSE	1 = Nockendaten 1 lesen (Nocken 0 bis 15)
37.3	CAM2RD_EN	BOOL	FALSE	1 = Nockendaten 2 lesen (Nocken 16 bis 31)
37.4	CAM3RD_EN	BOOL	FALSE	1 = Nockendaten 3 lesen (Nocken 32 bis 47)
37.5	CAM4RD_EN	BOOL	FALSE	1 = Nockendaten 4 lesen (Nocken 48 bis 63)
37.6	CAM5RD_EN	BOOL	FALSE	1 = Nockendaten 5 lesen (Nocken 64 bis 79)
37.7	CAM6RD_EN	BOOL	FALSE	1 = Nockendaten 6 lesen (Nocken 80 bis 95)
38.0	CAM7RD_EN	BOOL	FALSE	1 = Nockendaten 7 lesen (Nocken 96 bis 111)
38.1	CAM8RD_EN	BOOL	FALSE	1 = Nockendaten 8 lesen (Nocken 112 bis 127)

2.1.3 FM 352 einsetzen

2.1.3.1 Anschlussbelegung der Frontstecker der FM 352

Beschreibung der Geberschnittstelle

Lage der D-SUB-Buchse

Das folgende Bild zeigt die Einbaulage und die Bezeichnung der Buchse auf der Baugruppe. An die D-SUB-Buchse können Sie einen Initiator, Inkrementalgeber oder Absolutwertgeber (SSI) anschließen.

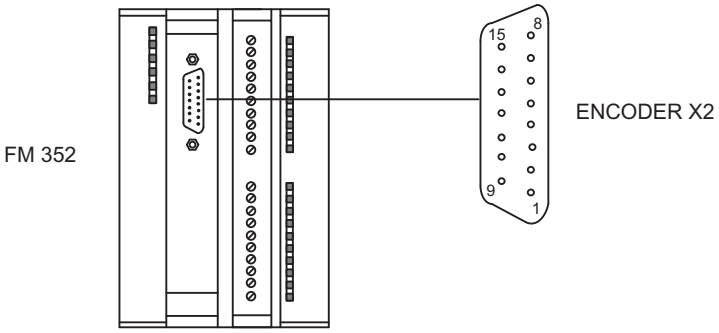

Belegung der Geberschnittstelle (X2)

Pin	Name	Initiator	Inkrementalgeber	Absolutwertgeber
1	A*	Gebersignal A (24 V)		---
2	CLS	---	---	SSI-Schiebetakt
3	/CLS	---	---	SSI-Schiebetakt invers
4	B*	---	Gebersignal B (24 V)	---
5	DC 24 V	Geberversorgung 24 V		

Pin	Name	Initiator	Inkrementalgeber	Absolutwertgeber
6	DC 5.2 V	---	Geberversorgung 5,2 V	
7	M	Masse		
8	N*	---	Nullmarkensignal (24 V)	---
9	RE	---	P/M-schaltend [2]	---
10	N	---	Nullmarkensignal (5 V)	---
11	/N	---	Nullmarkensignal invers (5 V)	---
12	/B / /CLI [1]	---	Gebersignal B invers (5 V)	SSI-Schiebetakt invers
13	B / CLI [1]	---	Gebersignal B (5 V)	SSI-Schiebetakt
14	/A / /DAT	---	Gebersignal A invers (5 V)	SSI-Daten invers
15	A / DAT	---	Gebersignal A (5 V)	SSI-Daten

[1] Im Mithörbetrieb

[2] **Hinweis:** Wenn Sie einen Inkrementalgeber eines Fremdherstellers in Gegentaktschaltung (P/M-schaltend) anschließen, müssen Sie Folgendes beachten:

- P-schaltend: Verbinden Sie RE (9) mit Masse (7).
- M-schaltend: Verbinden Sie RE (9) mit +24 V (5).

Belegung des Frontsteckers

Frontstecker

Am 20-poligen Frontstecker schließen Sie an:

- die digitalen Ausgänge und Schaltelemente (Eingänge)
- die Versorgungsspannung der Geber und der Digitalausgänge

Anschlussbelegung des Frontsteckers (X1)

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 352.

Anschluss	Name	Funktion
1	L+	DC 24 V Geberversorgung und Digitalausgänge
2	M	Masse Geberversorgung und Digitalausgänge
3	I 0	Bremsfreigabe
4	I 1	Längenmessung / Kantenerfassung / Fliegendes Istwert setzen
5	I 2	Referenzpunktschalter
6	I 3	Freigabe Spursignal 3
7	Q 0	Digitalausgang 0
8	Q 1	Digitalausgang 1
9	Q 2	Digitalausgang 2
10	Q 3	Digitalausgang 3
11	Q 4	Digitalausgang 4
12	Q 5	Digitalausgang 5
13	Q 6	Digitalausgang 6

Anschluss	Name	Funktion
14	Q 7	Digitalausgang 7
15	Q 8	Digitalausgang 8
16	Q 9	Digitalausgang 9
17	Q 10	Digitalausgang 10
18	Q 11	Digitalausgang 11
19	Q 12	Digitalausgang 12
20	–	–

Weitere Informationen zu Aufbau und Verdrahtung finden Sie in der Betriebsanleitung (http://support.automation.siemens.com/WW/view/de/2103044) des Elektronischen Nockensteuerwerks FM 352.

Hilfsspannung für Geber und Digitalausgänge (L+, M)

Die DC 24 V-Hilfsspannung der Geber und Digitalausgänge wird überwacht:

- auf Drahtbruch der 24 V-Zuleitung
- auf Spannungsausfall.

An der Geberschnittstelle (D-SUB-Buchse X2) werden DC 24 V und DC 5,2 V für die unterschiedlichen Geberarten zur Verfügung gestellt.

Laststromversorgungen

Die DC-Laststromversorgung muss folgenden Anforderungen genügen:

Als Laststromversorgung darf nur vom Netz sicher getrennte Kleinspannung DC ≤ 60 V verwendet werden. Die sichere Trennung kann realisiert sein nach den Anforderungen u. a. in

- VDE 0100 Teil 410 / HD 384-4-41 / IEC 364-4-41
  (als Funktionskleinspannung mit sicherer Trennung) bzw.
- VDE 0805 / EN 60950 / IEC 950
  (als Sicherheitskleinspannung SELV) bzw.
- VDE 0106 Teil 101.

4 Digitaleingänge (I 0 bis I 3)

An die 4 Digitaleingänge können Sie prellfreie Schalter (24 V P-schaltend) oder berührungslose Sensoren (2- oder 3-Draht Näherungsschalter) anschließen.

Die Digitaleingänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgebunden zur Baugruppenmasse.

13 Digitalausgänge (Q 0 bis Q 12)

Über 13 Digitalausgänge wird der jeweilige Zustand (ein / aus) der Spuren 0 bis 12 ausgegeben. Die Digitalausgänge sind potentialgebunden zur Baugruppenmasse.

Folgende Belastungen sind möglich:

- Arbeitsspannung 24 V
- Strombelastung 0,5 A / kurzschlussfest

Der Zustand jedes Ausgangs ist an der zugehörigen LED ablesbar.

2.1.3.2 Funktionen zum Nockensteuern

Einfluss von Funktionen auf das Schaltverhalten von Zeitnocken

Istwertveränderungen

Ein Zeitnocken kann durch folgende istwertverändernde Funktionen übersprungen werden:

- Istwert setzen
- Fliegendes Istwert setzen
- Nullpunktverschiebung
- Referenzpunkt nachtriggern

Schalten eines Zeitnockens

Wenn Sie durch eine der oben genannten Funktionen den Nockenanfang eines Zeitnockens überspringen, schaltet dieser Nocken ein, sofern die tatsächliche Bewegungsrichtung der Achse mit der für den Nocken parametrierten Wirkrichtung übereinstimmt. Die parametrierte Einschaltzeit läuft ab.

Hinweis

Im Stillstand ist die Bewegungsrichtung abhängig von den Istwertschwankungen.

Wenn Sie auch im Stillstand eine definierte Bewegungsrichtung berücksichtigen wollen, müssen Sie eine Hysterese parametrieren, die größer ist als die Schwankungen des Istwerts im Stillstand.

Ist die Achse nicht in Bewegung, so bleibt die zuletzt ermittelte Bewegungsrichtung erhalten.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Istwertverändernde Funktionen können bei einer Rundachse zu einem nicht gewünschten Einschalten von Zeitnocken führen.

Schalten Sie auf jeden Fall bei einer Rundachse die Zeitnocken "ungültig", wenn Sie durch die oben genannten Funktionen den Istwert beeinflussen wollen.

Istwert setzen / Fliegendes Istwert setzen / Istwert setzen rückgängig

Definition

Mit den Funktionen "Istwert setzen" und "Fliegendes Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Das Koordinatensystem verschiebt sich dadurch um den Wert: IST_{neu} - $IST_{aktuell}$

Dabei ist:

- IST_{neu} der Vorgabewert
- $IST_{aktuell}$ der Istwert zum Zeitpunkt der Ausführung

Neue Koordinaten ermitteln

Alle von Ihnen vorgegebenen Positionen im verschobenen Koordinatensystem berechnen Sie mit der folgenden Formel:

$Koordinate_{neu} = Koordinate_{alt} + (IST_{neu} - IST_{aktuell})$

Voraussetzungen

- Die Achse muss synchronisiert sein.
- Bei "Fliegendes Istwert setzen": Der Digitaleingang I1 muss verschaltet sein.

Ablauf der Funktion

1. Tragen Sie die Koordinate für den Istwert bzw. für den Fliegenden Istwert in den Kanal-DB ein.
   - Linearachse:
     Den angegebenen Istwert müssen Sie so wählen, dass die Software-Endschalter nach dem Aufruf der Funktion noch innerhalb des zulässigen Verfahrbereichs liegen.
     Der Betrag der Verschiebung, der sich aus **(IST_{neu} - $IST_{aktuell}$)** ergibt, muss kleiner oder gleich dem Betrag des zulässigen Verfahrbereichs sein (maximal 100 m bzw. 1000 m).
   - Rundachse:
     Für den angegebenen Istwert muss gelten:
     0 ≤ Istwert < Ende der Rundachse
2. Setzen Sie die entsprechenden Anstoßbits im Kanal-DB.
3. Rufen Sie die Anweisung CAM_CTRL auf.
   "Istwert setzen" wird sofort ausgeführt.
   "Fliegendes Istwert setzen" wird mit der nächsten steigenden Flanke am Digitaleingang I1 ausgeführt. Das Bit FVAL_DONE wird gesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.4	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.5	FVAL_EN	BOOL	FALSE	1 = Fliegendes Istwert setzen
90.0	AVAL	DINT	L#0	Koordinate für Istwert
94.0	FVAL	DINT	L#0	Koordinate für Fliegenden Istwert
25.5	FVAL_DONE	BOOL	FALSE	1 = Fliegendes Istwert setzen ausgeführt

Auswirkungen der Funktion

Am Beispiel "Istwert setzen" auf 400 mm (bei Position 200 mm) erkennen Sie, wie diese Funktion das Koordinatensystem verschiebt. Es ergeben sich folgende Auswirkungen:

- Die Lage des Arbeitsbereichs wird physikalisch **nicht** verschoben.
- Den einzelnen Punkten (wie z. B. den Software-Endschaltern) werden neue Koordinatenwerte zugeordnet.
- Die Nocken behalten ihre Koordinatenwerte und liegen somit an einer anderen physikalischen Position.
- Bei synchronisierter Achse und freigegebener Nockenbearbeitung kann der Lageistwert durch diese Funktion Nockenflanken oder komplette Nocken überspringen.
- Der Zustandswechsel des Nockens, der normalerweise einen Alarm auslösen würde, könnte verloren gehen.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 474).

Die folgende Tabelle zeigt die Verschiebung des Koordinatensystems durch "Istwert setzen" / "Fliegendes Istwert setzen".

Istwert setzen	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
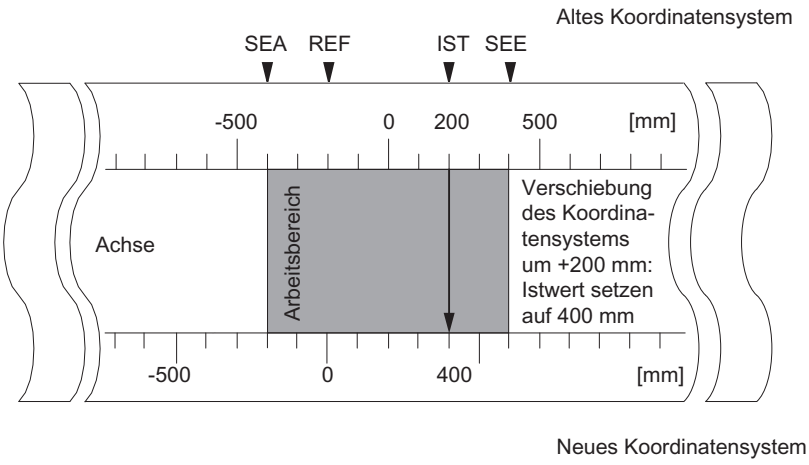	-400	-200	200	400
	-200	0	400	600

Rücknahme der Funktion

Mit der Funktion "Istwert setzen rückgängig" setzen Sie die Koordinatenverschiebung, die durch "Istwert setzen" bzw. "Fliegendes Istwert setzen" hergestellt wurde, wieder zurück.

Ein einmal angestoßenes "Fliegendes Istwert setzen" kann vor der Ausführung durch eine steigende Flanke am Eingang I1 nicht mehr gelöscht werden. Es kann aber durch ein neues "Fliegendes Istwert setzen" überschrieben werden.

Bei einem Baugruppenanlauf werden diese Funktionen zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.2	AVALREM_EN	BOOL	FALSE	1: Istwert setzen zurücknehmen

Mögliche Fehlerquellen

"Fliegendes Istwert setzen" und "Referenzpunkt nachtriggern" dürfen nicht gleichzeitig ausgeführt werden.

Bei der Funktion "Fliegendes Istwert setzen" kann ein Betriebsfehler gemeldet werden, wenn durch diese Funktion ein Software-Endschalter bei der steigenden Flanke am I1 den zulässigen Verfahrbereich verlassen würde. Dieser Betriebsfehler wird mit einem Diagnosealarm gemeldet und im Diagnosepuffer eingetragen.

Nullpunktverschiebung ausführen

Definition

Mit der Funktion "Nullpunktverschiebung" verschieben Sie den Nullpunkt im Koordinatensystem um den eingegebenen Wert. Das Vorzeichen bestimmt die Richtung der Verschiebung.

Neue Koordinate ermitteln

Alle Werte im verschobenen Koordinatensystem berechnen Sie mit der folgenden Formel:

$$\text{Koordinate}_{neu} = \text{Koordinate}_{alt} - (\text{NPV}_{neu} - \text{NPV}_{alt})$$

NPV_{alt} kennzeichnet dabei eine eventuell bestehende alte Nullpunktverschiebung. Wenn vor dem Aufruf keine Nullpunktverschiebung aktiv war, setzen Sie für NPV_{alt} den Wert 0 ein.

Sie können mit dieser Formel ermitteln, welche Koordinatenwerte z. B. die Software-Endschalter annehmen.

Ablauf der Funktion

1. Tragen Sie den Wert für die Nullpunktverschiebung in den Kanal-DB ein.
   - Linearachse:
     Die Nullpunktverschiebung muss so gewählt werden, dass die Software-Endschalter nach dem Aufruf der Funktion noch innerhalb des zulässigen Verfahrbereichs liegen.
   - Rundachse:
     Für die Nullpunktverschiebung muss gelten:
     Betrag der Nullpunktverschiebung ≤ Ende der Rundachse.
2. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.6	ZOFF_EN	BOOL	FALSE	1 = Nullpunktverschiebung setzen
86.0	ZOFF	DINT	L#0	Nullpunktverschiebung

Auswirkungen bei einer Linearachse

Am Beispiel einer Nullpunktverschiebung von -200 mm erkennen Sie, dass diese Funktion das Koordinatensystem in positive Richtung verschiebt. Es ergeben sich folgende Auswirkungen:

- Der Arbeitsbereich wird physikalisch **nicht** verschoben.
- Den einzelnen Punkten (wie z. B. den Software-Endschaltern) werden neue Koordinatenwerte zugeordnet.
- Die Nocken behalten ihre Koordinatenwerte und liegen somit an einer anderen physikalischen Position.
- Bei synchronisierter Achse und freigegebener Nockenbearbeitung kann der Lageistwert durch diese Funktion Nockenflanken oder komplette Nocken überspringen.
- Der Zustandswechsel des Nockens, der normalerweise einen Alarm auslösen würde, kann verloren gehen.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 474).

Die folgende Tabelle zeigt die Verschiebung des Koordinatensystems durch Nullpunktverschiebung.

Nullpunktverschiebung	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
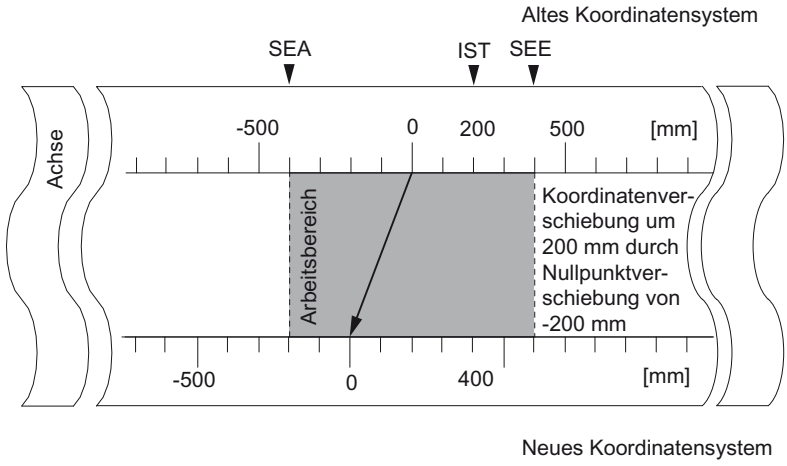	-400	-200	200	400
	-200	0	400	600

Auswirkungen bei einer Rundachse

Am Beispiel einer Nullpunktverschiebung um -45° erkennen Sie, wie diese Funktion das Koordinatensystem dreht.

Die folgende Tabelle zeigt die Drehung des Koordinatensystems durch eine Nullpunktverschiebung.

Werkzeug bei IST = 340°	Werkzeug bei IST = 25°
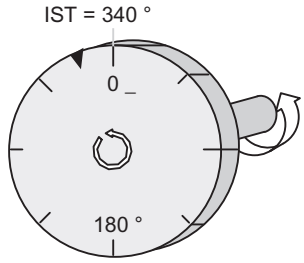	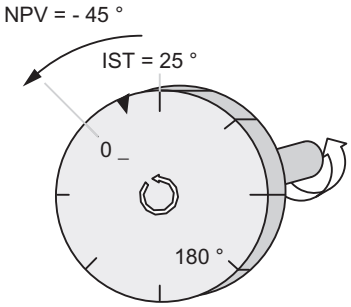
Der Nullpunkt dreht sich um -45 Grad. Alle Punkte der Achse erhalten neue Koordinatenwerte.	

Unter Berücksichtigung einer **NPV_{alt} = 0** ergibt sich ein neuer Wert von 385°

Da am Ende der Rundachse bei einer positiven Drehrichtung der Istwert wieder bei 0 beginnt, errechnet sich der tatsächliche Istwert von 25° aus:

$Koordinate_{neu}$ = $Koordinate_{alt}$ - (NPV_{neu} - NPV_{alt}) - Ende der Rundachse

Der Wert **Ende der Rundachse** muss nur dann subtrahiert werden, wenn

Koordinate$_{alt}$ - (**NPV**$_{neu}$ - **NPV**$_{alt}$) größer als das Ende der Rundachse wird.

Wegfall der Synchronisation

Wenn durch einen Fehler die Synchronisation verloren geht oder durch "Referenzpunkt nachtriggern" neu gesetzt wird, **bleibt** eine Nullpunktverschiebung aktiv.

Funktion zurücknehmen

Mit der Vorgabe einer Nullpunktverschiebung von 0 setzen Sie eine bestehende Nullpunktverschiebung wieder zurück.

Bezugspunkt setzen

Definition

Mit der Funktion "Bezugspunkt setzen" synchronisieren Sie die Achse. Die Funktion verschiebt den Arbeitsbereich. Alle Verschiebungen, die durch eine Nullpunktverschiebung oder durch "Istwert setzen" erzeugt wurden, bleiben erhalten.

Die Funktion projiziert den Arbeitsbereich auf die Achse. Durch unterschiedliche Wertvorgaben kann deshalb der Arbeitsbereich an unterschiedlichen Positionen im physikalischen Bereich der Achse liegen.

Voraussetzung

Die Nockenbearbeitung muss abgeschaltet sein.

Ablauf der Funktion

1. Tragen Sie den Wert für die Bezugspunktkoordinate in den Kanal-DB ein.
   - Linearachse:
     Die Bezugspunktkoordinate darf nicht außerhalb der Software-Endschalter liegen. Das gilt auch für die Bezugspunktkoordinate in einem verschobenen Koordinatensystem.
   - Rundachse:
     Für die Bezugspunktkoordinate muss gelten:
     0 ≤ Bezugspunktkoordinate < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.3	REFPT_EN	BOOL	FALSE	1 = Bezugspunktkoordinate setzen
98.0	REFPT	DINT	L#0	Bezugspunktkoordinate
25.0	SYNC	BOOL	FALSE	1 = Achse synchronisiert

Auswirkungen der Funktion

Am Beispiel "Bezugspunkt setzen" auf 300 mm erkennen Sie, wie diese Funktion den Arbeitsbereich auf der Achse verschiebt.

Daraus ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Bezugspunktkoordinate gesetzt.
- Der Arbeitsbereich wird physikalisch auf der Achse verschoben.
- Die einzelnen Punkte behalten ihren ursprünglichen Koordinaten, liegen aber auf neuen physikalischen Positionen.
- Das Bit SYNC in den Rückmeldesignalen ist gesetzt.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Bezugspunkt setzen".

Bezugspunkt setzen	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
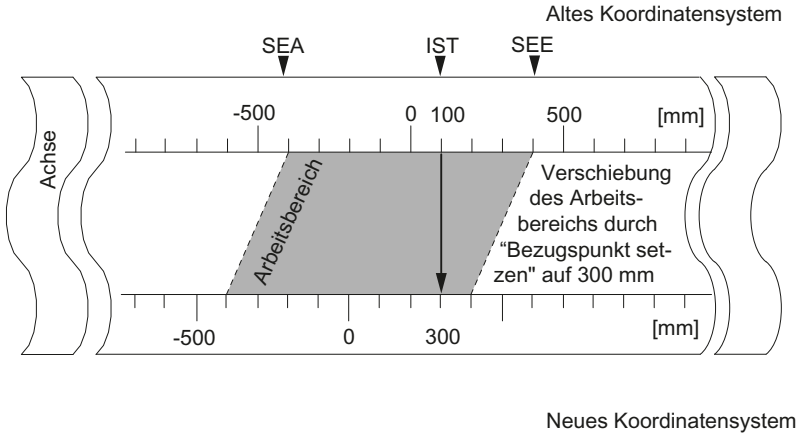	-400	-200	100	400
	-400	-200	300	400

Besonderheiten Absolutwertgeber

Diese Funktion ist für eine Absolutwertgeberjustage (Seite 504) notwendig.

Nockenflanken ändern

Definition

Mit der Funktion "Nockenflanken ändern" können Sie im laufenden Betrieb den Nockenanfang und - bei einem Wegnocken - auch das Nockenende eines einzelnen Nockens ändern.

Voraussetzung

Der Nocken, den Sie ändern wollen, muss gültig sein.

Ablauf der Funktion

1. Tragen Sie die Nockennummer in den Kanal-DB ein.
2. Bei einem Wegnocken:
   Tragen Sie Nockenanfang und Nockenende in den Kanal-DB ein.
   Bei einem Zeitnocken:
   Tragen Sie den Wert für den Nockenanfang in den Kanal-DB ein.
3. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.7	CH01CAM_EN	BOOL	FALSE	1 = Funktion für Nockenflanken (1 Nocken) schreiben
102.0	CAM_NO	INT	0	Nockennummer
104.0	CAM_START	DINT	L#0	Nockenanfang
108.0	CAM_END	DINT	L#0	Nockenende

Auswirkungen der Funktion

Die FM x52 verschiebt zunächst die Einschaltflanke und dann die Ausschaltflanke des Nockens. Dieser Ablauf ist unabhängig von der Richtung, in die der Nocken geschoben wird.

Sonderfall:

Durch den oben erklärten Ablauf kann kurzzeitig ein inverser Nocken (Seite 453) entstehen, wenn der neue Nockenanfang größer ist als das alte Nockenende.

Das folgende Bild zeigt die Veränderung der Nockenflanken in Einzelschritten.

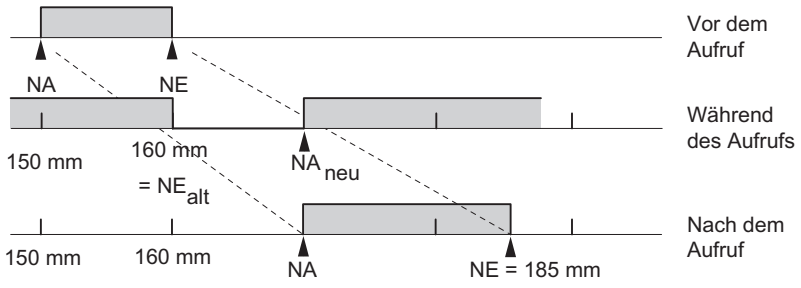

Hinweis

Wenn für diesen Nocken ein Prozessalarm freigeschaltet wurde, kann die FM x52 bei Erkennen des inversen Nockens je nach Parametrierung ein oder zwei Prozessalarme auslösen.

Durch das Ändern der Ein- und/oder Ausschaltflanke kann es zum Überspringen einer Nockenflanke oder des ganzen Nockens kommen.

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 474).

Zustandswechsel des Nockens, die normalerweise einen Prozessalarm auslösen würden, können verloren gehen.

Geänderte Werte auslesen

Die geänderten Werte können Sie mit einem der Aufträge CAM1RD_EN bis CAM8RD_EN auslesen.

Rücknahme der Funktion

Die geänderten Werte gehen beim Neustart der Baugruppe verloren.

"Schnelles Nockenändern" ausführen

Definition

Mit der Funktion "Schnelles Nockenändern" können Sie im laufenden Betrieb Daten von bis zu 16 beliebigen Nocken gleichzeitig ändern.

Voraussetzung

Die Nocken, die Sie ändern wollen, müssen gültig sein.

Ablauf der Funktion

1. Tragen Sie die Anzahl der zu ändernden Nocken in den Kanal-DB ein.
2. Tragen Sie die Nummer des ersten zu ändernden Nockens in den Kanal-DB ein.

3. Setzen Sie die Anstoßbits für die gewünschte Änderungen.
4. Tragen Sie die neuen Werte in den Kanal-DB ein.
5. Wiederholen Sie die Schritte 2 bis 4 für jeden zu ändernden Nocken.
6. Setzen Sie das entsprechende Anstoßbit im Kanal-DB

Verwendete Daten im Kanal-DB

Adresse absolut	Name	Typ	Anfangswert	Kommentar
37.0	CH16CAM_EN	BOOL	FALSE	1 = Funktion für schnelles Nockenändern (16 Nocken) schreiben
176.0	C_QTY	BYTE	B#16#0	Anzahl der zu ändernden Nocken
177.0	DIS_CHECK	BOOL	FALSE	1 = Datenprüfung ausschalten

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.0	CAM_NO	BYTE	B#16#0	Nummer des zu ändernden Nockens
+1.0	C_EFFDIR	BOOL	FALSE	1 = Änderung der Nockenwirkrichtung
+1.1	C_CBEGIN	BOOL	FALSE	1 = Änderung des Nockenanfangs auf den Wert CBEGIN
+1.2	C_CEND	BOOL	FALSE	1 = Änderung des Nockenendes / der Einschaltzeit auf den Wert CEND
+1.3	C_LTIME	BOOL	FALSE	1 = Änderung der Vorhaltezeit auf den Wert LTIME
+1.4	CAM_OFF	BOOL	FALSE	1 = Ausschalten des Nockens während der Nockenänderung
+1.5	EFFDIR_P	BOOL	FALSE	1 = Wirkrichtung positiv (plus)
+1.6	EFFDIR_M	BOOL	FALSE	1 = Wirkrichtung negativ (minus)
+2.0	CBEGIN	DINT	L#0	Neuer Nockenanfang
+6.0	CEND	DINT	L#0	Neues Nockenende / neue Einschaltzeit
+10.0	LTIME	INT	L#0	Neue Vorhaltezeit

Nocken abschalten beim Ändern

Ein konsistentes Ändern des Nockenanfangs und des Nockenendes ist nur dann möglich, wenn Sie den Nocken zum Zeitpunkt des Änderns abgeschaltet haben (CAM_OFF).

Datenprüfung durch die Baugruppe

Mit dem Parameter DIS_CHECK (Kanal-DB) stellen Sie ein, ob die Datenprüfung der übertragenen Daten durch die FM x52 ausgeschaltet werden soll. Wenn Sie die Datenprüfung ausschalten, müssen Sie selbst sicherstellen, dass nur zulässige Werte übertragen werden. Wenn Sie unzulässige Werte ohne Prüfung eingeben, kann dies zu einem unvorhergesehenen Verhalten der Baugruppe führen.

- FALSE: Die Baugruppe überprüft alle Daten, die übertragen werden
- TRUE: Die Datenprüfung bzgl. der Nockenparameter ist abgeschaltet. Dadurch werden die zu ändernden Daten schneller auf der FM x52 eingearbeitet.
  Unabhängig von dieser Funktion wird immer geprüft, ob
  - die Achse parametriert ist,
  - die Anzahl der zu ändernden Nocken (C_QTY) zulässig ist,
  - der Nocken (Nockennummer), der geändert werden soll, gültig ist.

Erst wenn alle Daten geprüft und fehlerfrei sind, werden sie auf der Baugruppe aktiviert.

Im Fehlerfall werden alle Daten verworfen.

Auswirkungen der Funktion

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 474).

Geänderte Werte auslesen

Die geänderten Werte können mit einem der Aufträge CAM1RD_EN bis CAM8RD_EN ausgelesen werden.

Rücknahme der Funktion

Die geänderten Werte gehen beim Neustart der Baugruppe verloren.

"Längenmessung" und "Kantenerfassung" ausführen

Definition

Mit den Funktionen "Längenmessung" und "Kantenerfassung" können Sie die Länge eines Werkstücks ermitteln.

Die Längenmessung und Kantenerfassung sind und bleiben so lange aktiv, bis Sie sie wieder ausschalten oder die jeweils andere Messmethode wählen. Falls Sie beide Messmethoden gleichzeitig wählen, wird die Anweisung CAM_CTRL die Längenmessung einschalten.

Voraussetzung

Am Eingang I1 muss ein **prellfreier** Schalter angeschlossen sein.

Ablauf der Funktionen

Je nach der Art der Messung aktualisiert die FM 352 die Daten auf der Baugruppe zu einem unterschiedlichen Zeitpunkt. Jede Aktualisierung meldet die FM 352 in einem Parameter an der Rückmeldeschnittstelle.

Längenmessung

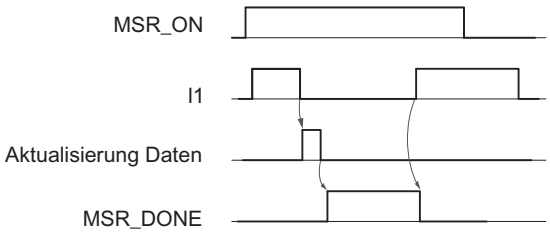

1. Setzen Sie den Funktionsschalter für "Längenmessung".
2. Die steigende Flanke des Eingangs I1 startet die Längenmessung.
3. Die fallende Flanke des Eingangs I1 beendet eine laufende Messung. Die FM 352 aktualisiert die Daten Startwert, Endwert und Länge.
4. Mit dem gesetzten Parameter MSR_DONE meldet die FM 352 das Aktualisieren der Daten. Der Parameter zeigt an, dass die Messung beendet ist. Die Messergebnisse können ausgelesen werden.
5. Der Start einer weiteren Messung mit der steigenden Flanke von I1 setzt den Parameter MSR_DONE zurück.

Wenn während einer Längenmessung die Funktion ausgeschaltet wird, dann aktualisiert die FM 352 die Daten nicht. Der Parameter MSR_DONE bleibt zurückgesetzt.

Kantenerfassung

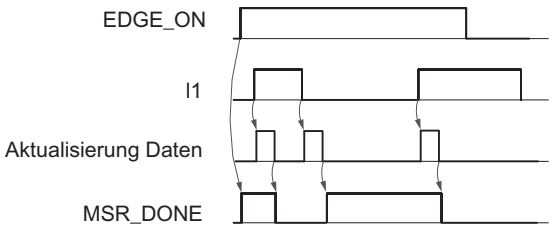

1. Tragen Sie ggf. einen Wert für den minimalen Kantenabstand im Parameter-DB ein. Schreiben und aktivieren Sie die Maschinendaten.
2. Setzen Sie den Funktionsschalter für "Kantenerfassung". Der Parameter MSR_DONE wird gesetzt.
3. Die steigende Flanke des Eingangs I1 startet die Kantenerfassung. Die Messergebnisse werden aktualisiert und können ausgelesen werden, der Startwert der Messung wird eingetragen; Endwert und Länge werden -1.

4. Nach der Aktualisierung meldet die FM 352 die Veränderung durch das Rücksetzen des Parameters MSR_DONE.
5. Die fallende Flanke des Eingangs I1 beendet eine laufende Messung. Die FM 352 aktualisiert die Daten für Endwert der Messung und Länge.
6. Nach der Aktualisierung meldet die FM 352 die Veränderung durch das Setzen des Parameters MSR_DONE. Die Messergebnisse können ausgelesen werden.
7. Der Start einer weiteren Messung mit der steigenden Flanke von I1 setzt den Parameter MSR_DONE zurück.

Wenn während einer Kantenerfassung die Funktion ausgeschaltet wird, dann aktualisiert die FM 352 die Daten nicht. Der Parameter MSR_DONE bleibt zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
25.1	MSR_DONE	BOOL	FALSE	1 = Längenmessung ist abgeschlossen
34.0	EDGE_ON	BOOL	FALSE	1 = Kantenerfassung ein
34.2	MSR_ON	BOOL	FALSE	1 = Längenmessung ein
38.2	MSRRD_EN	BOOL	FALSE	1 = Messwerte lesen
112.0	BEG_VAL	DINT	L#0	Startwert
116.0	END_VAL	DINT	L#0	Endwert
120.0	LEN_VAL	DINT	L#0	Länge

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	**Minimaler Kantenabstand bei Kantenerfassung** Zulässiger Wertebereich: • 0 ... 1 000 000 000 µm

Mit dem minimalen Kantenabstand definieren Sie einen Bereich nach dem Erkennen des Messbeginns bei einer Kantenerfassung. Wenn das Ende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen.

Der Messbeginn wird erst nach Verfahren des Wegstücks "minimaler Kantenabstand" gemeldet.

Randbedingungen für eine Längenmessung

- Der Abstand zwischen Ausschaltflanke und Einschaltflanke am Eingang I1 muss so groß sein, dass Ihr Programm in der CPU das Ergebnis der Messung korrekt auswerten kann, bevor eine neue Messung beginnt.
- Der minimale Abstand zwischen der steigenden und der fallenden Flanke am Eingang I1 wie auch zwischen der fallenden und der nächsten steigenden Flanke am Eingang I1 muss größer als 2 ms sein.

Fehlerhafte Messung

Im Fall einer fehlerhaften Längenmessung/Kantenerfassung liefert die FM 352 den Wert -1 für die Länge zurück.

Eine "Längenmessung" oder "Kantenerfassung" darf maximal 126 Nullübergänge in eine Richtung durchlaufen. Ein Nullübergang ist der Rundachsenübergang vom Wert Ende der Rundachse nach 0 und umgekehrt. Sobald mehr als 126 Nullübergänge in eine Richtung von der FM 352 erkannt werden, wird eine fehlerhafte "Längenmessung" bzw. "Kantenerfassung" gemeldet, auch wenn anschließend wieder Nullübergänge in der anderen Richtung erkannt wurden.

Eine Messung ist auch fehlerhaft, wenn

- bei einer Rundachse die gemessene Länge größer als 2^{31} ist,
- gleichzeitig die Einschalt- und die Ausschaltflanke von der FM 352 erkannt wird (z. B. durch Schalterprellen).

Dieses Ereignis kann als Prozessalarm gemeldet werden.

Verschiebung des Koordinatensystems während einer Längenmessung

Koordinatenverschiebungen beeinflussen die gemessene Länge unter folgenden Voraussetzungen:

- Sie setzen einen Inkrementalgeber oder einen Initiator ein oder Sie betreiben die FM 352 im Simulationsbetrieb.
- Sie führen "Bezugspunkt setzen" oder "Referenzpunkt nachtriggern" während einer laufenden Längenmessung aus.

Beispiel

Sie können die oben genannte Beeinflussung der gemessenen Länge wie folgt ausnutzen:

Sie haben ein System, bei dem während einer Längenmessung immer ein Schlupf auftritt.

Durch "Referenzpunkt nachtriggern" können Sie diesen Schlupf korrigieren, so dass dann korrekte Längenmesswerte ausgegeben werden.

Referenzpunkt nachtriggern

Definition

Mit der Funktion "Referenzpunkt nachtriggern" können Sie die Achse aufgrund eines wiederkehrenden externen Ereignisses synchronisieren.

Die Funktion bleibt solange aktiv, bis Sie sie wieder ausschalten.

Voraussetzung

- Sie verwenden einen Inkrementalgeber oder Initiator.
- Das externe Ereignis kann eine Nullmarke eines Inkrementalgebers oder ein Referenzpunktschalter am Eingang I2 sein.

Ablauf der Funktion

1. Tragen Sie den Wert der Referenzpunktkoordinate in den Parameter-DB ein.
2. Tragen Sie die Art des "Referenzpunkt nachtriggerns" in den Parameter-DB ein. Hierbei haben Sie folgende Möglichkeiten:
   - Nur die Nullmarke des Gebers wird ausgewertet (RETR_TYPE = 7).
   - Nur der Referenzpunktschalter wird ausgewertet (RETR_TYPE = 6).
   - Die Nullmarke wird ausgewertet
     in positiver Richtung: Die erste steigende Flanke der Nullmarke nach Verlassen des Referenzpunktschalters in positiver Richtung wird ausgewertet (RETR_TYPE = 0).
     in negativer Richtung: Die erste fallende Flanke der Nullmarke nach Verlassen des Referenzpunktschalters in negativer Richtung wird ausgewertet (RETR_TYPE = 1).
3. Schreiben und aktivieren Sie die Maschinendaten.
4. Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.3	REFTR_ON	BOOL	FALSE	1 = Referenzpunkt nachtriggern
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
52.0	RETR_TYPE	DINT	L#0	Art des "Referenzpunkt nachtriggern"

Auswirkungen der Funktion

- Die FM x52 wertet die Nullmarke und den Referenzpunktschalter in Abhängigkeit der Bewegungsrichtung der Achse aus.
  - Bei positiver Bewegungsrichtung werden die steigenden Flanken ausgewertet.
  - Bei negativer Bewegungsrichtung werden die fallenden Flanken ausgewertet.
- Die Istposition wird auf den Wert der Referenzpunktkoordinate gesetzt.
- Der Arbeitsbereich wird physikalisch auf der Achse verschoben.
- Die einzelnen Punkte behalten ihren ursprünglichen Wert, liegen aber auf neuen physikalischen Positionen.

- Zustandswechsel des Nockens, die normalerweise einen Alarm auslösen würden, können verloren gehen.
- Das Bit SYNC in den Rückmeldesignalen wird gesetzt.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 474).

Beispiel

Für das Beispiel gilt:

- Von Referenzpunktschalter und Nullmarke werden die steigenden Flanken ausgewertet (Positive Bewegungsrichtung der Achse).
- Die Referenzpunktkoordinate hat den Wert 300 mm.
- Zum Zeitpunkt der Ausführung ist keine Nullpunktverschiebung aktiv.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Referenzpunkt nachtriggern".

Referenzpunkt nachtriggern	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
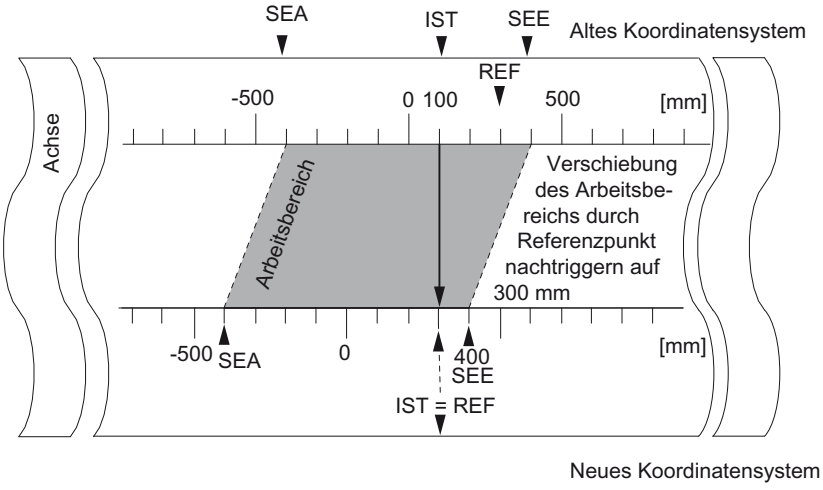	-400	300	100	400
	-400	300	300	400

Berücksichtigung einer Nullpunktverschiebung

Wenn eine Nullpunktverschiebung aktiv ist, wird diese bei der Funktion "Referenzpunkt nachtriggern" mit berücksichtigt. Das bedeutet, dass die Referenzpunktkoordinate, die gesetzt wird, sich nach folgender Formel berechnet:

$Ref = Ref_{MD}$ - Nullpunktverschiebung

Ref_{MD} ist der Wert, der in den Maschinendaten als Referenzpunktkoordinate abgelegt ist.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Referenzpunkt nachtriggern" mit Nullpunktverschiebung.

Referenzpunkt nachtriggern	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
Altes Koordinatensystem Achse SEA -500 IST 0 SEE REF 500 [mm] 0 NPV = -100 Arbeitsbereich Verschiebung des Arbeitsbereichs durch Referenzpunkt nachtriggern -500 SEA 0 500 [mm] SEE = REF = REF_{MD} - NPV Neues Koordinatensystem	-500	300	0	300
	-400	400	100	400
	-400	400	400	400

Software-Endschalter abschalten

Definition

Die Funktion "Software-Endschalter abschalten" schaltet die Überwachung der Software-Endschalter einer Linearachse aus.

Die Funktion bleibt solange aktiv, bis Sie sie wieder ausschalten. Dann werden die ursprünglich parametrierten Software-Endschalter wieder aktiv.

Ablauf der Funktion

Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.4	SSW_OFF	BOOL	FALSE	1 = Software-Endschalter abgeschaltet

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L#-1000000000	Software-Endschalter Anfang
68.0	SSW_END	DINT	L#1000000000	Software-Endschalter Ende

Auswirkungen der Funktion

- Simulation
  - Wird im Simulationsbetrieb ein Software-Endschalter überfahren, dann wird der Simulationsbetrieb angehalten.
  - Schalten Sie jetzt die Überwachung der Software-Endschalter aus, wird der Simulationsbetrieb fortgesetzt. Die Achse bewegt sich in die vorgegebene Richtung.
- Nullpunktverschiebung bei ausgeschalteter Überwachung
  Wird eine Nullpunktverschiebung vorgegeben, bei der die Software-Endschalter noch innerhalb der Verfahrbereichsgrenzen liegen, kann trotzdem der Istwert außerhalb des zulässigen Zahlenbereichs sein.
- Nocken, die außerhalb der parametrierten Software-Endschalter liegen, können geschaltet werden.

⚠ VORSICHT
Es kann zu Sachschäden kommen. Wen Sie mit den Software-Endschaltern den Verfahrbereich einschränken und damit Ihre Anlage schützen, kann es beim Abschalten der Software-Endschalter zu Sachschäden kommen. Achten Sie deshalb bei der Planung der Anlage darauf, dass der Antrieb im gesamten physikalischen Verfahrbereich fahren kann.

"Simulation" ausführen

Definition

Die Funktion "Simulation" bietet die Möglichkeit, das Nockensteuerwerk ohne angeschlossene Geber zu aktivieren.

Ablauf der Funktion

1. Tragen Sie die Simulationsgeschwindigkeit in den Parameter-DB ein.
2. Schreiben und aktivieren Sie die Maschinendaten.
3. Wählen Sie im Kanal-DB als Simulationsrichtung entweder die positive oder die negative Richtung aus.
4. Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.2	DIR_M	BOOL	FALSE	1 = Simulation in negativer Richtung
15.3	DIR_P	BOOL	FALSE	1 = Simulation in positiver Richtung
34.1	SIM_ON	BOOL	FALSE	1 = Simulation ein

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	SIM_SPD	DINT	L#0	Simulationsgeschwindigkeit

Auswirkungen beim Einschalten der Simulation

- Die Gebersignale werden nicht mehr ausgewertet.
- Alle Überwachungen bezüglich des Gebereingangs werden abgeschaltet.
- Eventuell gemeldete Fehler bezüglich des Gebers werden zurückgesetzt.
- Die FM x52 simuliert die Bewegung einer Achse mit der konstanten Simulationsgeschwindigkeit.
- Die Nockenbearbeitung wird mit dem Einschalten der Simulation abgeschaltet. Sie können sie dann jedoch wieder einschalten. Die Synchronisation bleibt dabei erhalten.
- Der Lageistwert verändert sich ab dem aktuellen Istwert in Abhängigkeit von der Simulationsgeschwindigkeit und der Simulationsrichtung.

Auswirkungen beim Ausschalten der Simulation

- Die Nockenbearbeitung wird abgeschaltet.
- Bei einem Inkrementalgeber oder Initiator wird die Synchronisation gelöscht. Als Istwert wird dann der Wert der Referenzpunktkoordinate eingestellt.
- Bei einem Absolutgeber wird der Lageistwert gemeldet, der dem aktuellen Geberstand entspricht. Die Gebersignale werden wieder so ausgewertet, wie sie in den Maschinendaten parametriert sind.

Grenzwerte

Die maximale und minimale Grenze der Simulationsgeschwindigkeit ist abhängig von der Auflösung (Seite 509).

Geschwindigkeit

Zwischen der vorgegebenen und der tatsächlichen Geschwindigkeit (Seite 509) , mit der die Baugruppe arbeitet, kann es Unterschiede geben.

"Zählwerte der Zählnockenspuren" auslesen

Definition

Mit der Funktion "Zählwerte der Zählnockenspuren" lesen Sie die aktuellen Zählwerte aus.

Ablauf der Funktion

1. Legen Sie die Zählnockenspuren und die oberen Zählwerte in den Maschinendaten fest.
2. Schreiben und aktivieren Sie die Maschinendaten.
3. Geben Sie die Zählfunktion frei.
4. Der Zählwert wird auf den oberen Zählwert gesetzt.
5. Bei jeder steigenden Flanke des Spurergebnisses wird der Zählwert um 1 verringert.
6. Setzen Sie das Anstoßbit im Kanal-DB zum Lesen der Zählwerte.
7. Die Zählwerte für beide Spuren werden in den Kanal-DB geschrieben. Für eine Spur, die nicht als Zählnockenspur parametriert wurde, wird 0 ausgegeben.
8. Erreicht der Zählwert den Wert 0, wird das Spurkennbit der Zählnockenspur auf 1 gesetzt.
9. Mit der nächsten fallenden Flanke des Spurergebnisses wird das Spurkennbit wieder 0 und der Zähler auf den oberen Zählwert gesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.5	CNTC0_EN	BOOL	FALSE	1 = Zählfunktion der Zählnockenspur 0 freigeben
15.6	CNTC1_EN	BOOL	FALSE	1 = Zählfunktion der Zählnockenspur 1 freigeben
38.3	CNTTRC_EN	BOOL	FALSE	1 = Zählwerte der Zählnockenspuren lesen
124.0	CNT_TRC0	INT	0	Aktueller Zählwert für Zählnockenspur 0
126.0	CNT_TRC1	INT	0	Aktueller Zählwert für Zählnockenspur 1

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
99.0	SPEC_TRC0	BOOL	FALSE	1 = Spur 0 ist Zählnockenspur
99.1	SPEC_TRC1	BOOL	FALSE	1 = Spur 1 ist Zählnockenspur
100.0	CNT_LIM0	DINT	L#2	Oberer Zählwert für Zählnockenspur 0
104.0	CNT_LIM1	DINT	L#2	Oberer Zählwert für Zählnockenspur 1

"Positions- und Spurdaten" auslesen

Definition

Mit der Funktion "Positions- und Spurdaten" lesen Sie die zu diesem Zeitpunkt aktuelle Position, die Geschwindigkeit und die Spurkennbits. Die Spurkennbits werden erfasst, bevor sie mit Maschinen- und Kanaldaten verknüpft werden.

Der in der FM x52 implementierte Rechenalgorithmus verrechnet Geschwindigkeitsänderungen, die größer sind als 1 Impuls pro 4 ms. Die angezeigte Geschwindigkeit ist also mit der entsprechenden Ungenauigkeit behaftet und ist insbesondere für Regelungszwecke nicht geeignet. Die interne Geschwindigkeit, die zum Dynamisieren der Nocken verwendet wird, ist genauer.

Ablauf der Funktion

1. Setzen Sie das Anstoßbit im Kanal-DB.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
38.4	ACTPOS_EN	BOOL	FALSE	1 = Positions- und Spurdaten lesen
128.0	ACTPOS	DINT	L#0	Aktuelle Position
132.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
136.0	TRACK_ID	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31

"Geberdaten" auslesen

Definition

Mit der Funktion "Geberdaten" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.

Voraussetzungen

Sie können den Wert für die Absolutwertgeberjustage (Seite 504) auslesen, nachdem Sie die Funktion "Bezugspunkt setzen (Seite 480)" ausgeführt haben.

Ablauf der Funktion

1. Setzen Sie das Anstoßbit im Kanal-DB.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
38.5	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
140.0	ENCVAL	DINT	L#0	Geberwert / Zählerstand (interne Darstellung)
144.0	ZEROVAL	DINT	L#0	Zählerstand bei der letzten Nullmarke (interne Darstellung)
148.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage

"Nocken- und Spurdaten" auslesen

Definition

Mit der Funktion "Nocken- und Spurdaten" lesen Sie die zu diesem Zeitpunkt aktuellen Nocken- und Spurkennbits sowie die Position. Die Spurkennbits werden erfasst, bevor sie mit Maschinen- und Kanaldaten verknüpft werden.

Ablauf der Funktion

1. Tragen Sie als Typkennung eine 1 in den Parameter FM_TYPE des Kanal-DB ein. Damit können Sie 24 Byte Nocken- und Spurdaten lesen.
   Wenn Sie für die Typkennung eine 0 eintragen, werden nur die Nockenkennbits (16 Byte) gelesen.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 bis V4.0 1 = FM 352 ab V5.0 und FM 452
38.6	CAMOUT_EN	BOOL	FALSE	1 = Nocken- und Spurdaten lesen
152.0	CAM_00_31	DWORD	DW#16#0	Nockenkennbits für Nocken 0 bis 31
156.0	CAM_32_63	DWORD	DW#16#0	Nockenkennbits für Nocken 32 bis 63
160.0	CAM_64_95	DWORD	DW#16#0	Nockenkennbits für Nocken 64 bis 95
164.0	CAM_96_127	DWORD	DW#16#0	Nockenkennbits für Nocken 96 bis 127
168.0	TRACK_ID1	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31
172.0	ACTPOS1	DINT	L#0	Aktuelle Position

"Steuersignale für das Nockensteuerwerk" setzen

Definition

Mit der Funktion "Steuersignale für das Nockensteuerwerk" geben Sie die Nockenbearbeitung sowie die Spuren frei.

Ablauf der Funktion

1. Setzen Sie die gewünschten Bits im Kanal-DB.
2. Die Daten werden bei jedem Aufruf der Anweisung CAM_CTRL zur Baugruppe übertragen.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.4	CAM_EN	BOOL	FALSE	1 = Nockenbearbeitung freigeben
16.0	TRACK_EN	WORD	W#16#0	Nockenspuren 0 bis 12 freischalten Bit 0 = Spur 0

Auswirkungen

Die Nockenbearbeitung wird je nach Freigabe gestartet oder angehalten.

Die Spurkennbits der freigegebenen Spuren werden an die Spursignale und die Digitalausgänge weitergegeben.

"Rückmeldesignale für das Nockensteuerwerk" abfragen

Definition

Mit der Funktion "Rückmeldesignale für das Nockensteuerwerk" werden Sie über den aktuellen Zustand des Nockensteuerwerks und der Spursignale informiert. Konsistenz zwischen der gemeldeten Position und den Spursignalen ist nicht gewährleistet.

Ablauf der Funktion

Die Daten werden bei jedem Aufruf der Anweisung CAM_CTRL im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.4	CAM_ACT	BOOL	FALSE	1 = Nockenbearbeitung läuft
26.0	ACT_POS	DINT	L#0	Aktuelle Position der Achse
30.0	TRACK_OUT	DWORD	DW#16#0	Aktuelle Spursignale der Spuren 0 bis 31 Bit 0 = Spur 0

"Rückmeldesignale für die Diagnose" abfragen

Definition

Mit der Funktion "Rückmeldesignale für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Ablauf der Funktion

1. Wenn die Baugruppe einen neuen Eintrag in den Diagnosepuffer macht, setzt sie in der Rückmeldeschnittstelle das Bit DIAG. Jedes Auftreten eines Fehlers (Seite 540) bewirkt einen Eintrag in den Diagnosepuffer.
2. Wenn die Baugruppe einen Schreibauftrag mit fehlerhaften Daten erkennt, setzt sie in der Rückmeldeschnittstelle das Bit DATA_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen.
3. Die Anweisung CAM_CTRL legt die Rückmeldeschnittstelle im Kanal-DB ab.
4. Nachdem der Diagnosepuffer von der Anweisung CAM_DIAG gelesen wurde, setzt die Baugruppe das Bit DIAG wieder auf 0.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler

2.1.3.3 FM 352 konfigurieren

2.1.3.4 FM 352 parametrieren

Maßsystem

Maßsystem auswählen

Maßsystem auswählen

Sie können für die Eingabe und Ausgabe der Daten unter den folgenden Maßsystemen wählen:

- Millimeter (Voreinstellung)
- Inch
- Grad
- Impulse

Hinweis

Wenn Sie das Maßsystem im Dialog "Maßsystem" verändern, dann werden die Werte in das neue System umgerechnet. Dabei können Rundungsfehler auftreten.

Wenn Sie das Maßsystem über das Maschinendatum ändern, werden die Werte nicht automatisch umgerechnet.

Wenn das Maßsystem von oder nach "Impulse" wechselt, wird die Nockenbearbeitung abgeschaltet und die Achse ist nicht mehr synchronisiert.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
8.0	UNITS	DINT	L#1	**Maßsystem** Zulässige Werte: • 1 = 10^{-3} mm • 2 = 10^{-4} inch • 3 = 10^{-4} grad • 4 = 10^{-2} grad • 5 = Impulse • 6 = 10^{-3} grad

Standard-Maßsystem in der Online-Hilfe

Wir verwenden für die Angabe der Grenzwerte das Maßsystem mm. Für die Bestimmung der Grenzwerte in den anderen Maßsystemen nehmen Sie deshalb folgende Umrechnung vor:

Für die Umrechnung von		berechnen Sie
mm → inch		Grenzwert (inch) = Grenzwert (mm) × 0,1 [1)]
mm → grad	10^{-4} (4 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 0,1
	10^{-3} (3 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 1
	10^{-2} (2 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 10
mm → Impulse		Grenzwert (Impulse) = Grenzwert (mm) × 1000

1) Die Anzahl der Nachkommastellen hat beim Maximalwert Auswirkungen auf die Anzahl der Stellen vor dem Komma. Im Maßsystem Inch werden 4 Nachkommastellen verwendet, somit können maximal 100 000,0000 inch eingegeben werden. Das Maßsystem Millimeter verwendet 3 Nachkommastellen, hier können maximal 1 000 000,000 mm eingegeben werden.

Achse

Achsart festlegen

Achsart auswählen

Achsarten

Das Nockensteuerwerk unterstützt folgende Achsarten:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

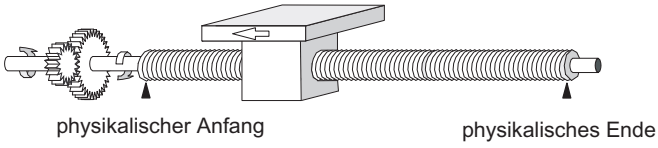

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

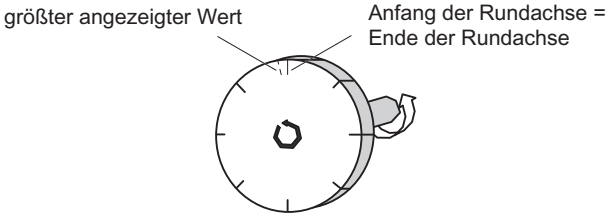

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	AXIS_TYPE	DINT	L#0	**Achsart** Zulässige Werte: • 0 = Linearachse • 1 = Rundachse

Software-Endschalter festlegen

Software-Endschalter

Software-Endschalter gibt es nur für die Linearachse.

Der Bereich, der durch die Software-Endschalter begrenzt wird, ist der Arbeitsbereich. Die Grenzen des Arbeitsbereichs können von der FM x52 überwacht werden.

Der Software-Endschalter Anfang (SEA) muss immer kleiner sein als der Software-Endschalter Ende (SEE).

Die Software-Endschalter sind aktiv, wenn die FM x52 synchronisiert ist.

- Bei Verwendung eines Inkrementalgebers oder Initiators:
  Nach jedem Anlauf der FM x52 ist die Achse zunächst nicht synchronisiert. Erst nach einer Synchronisation werden die parametrierten Software-Endschalter überwacht.
- Bei Verwendung eines Absolutwertgebers (SSI):
  Die FM x52 ist synchronisiert, nachdem sie ein vollständiges fehlerfreies Telegramm empfangen hat. Ab diesem Zeitpunkt werden die Software-Endschalter überwacht. Der von Ihnen verwendete Absolutwertgeber muss mindestens den Arbeitsbereich inklusive der Software-Endschalter abdecken.

Zusammenhang zwischen Arbeitsbereich, Geberbereich und Verfahrbereich

- Der Arbeitsbereich ist der Bereich, den Sie für Ihre Aufgabe durch die Software-Endschalter bestimmen.
- Der Geberbereich ist der vom Geber eindeutig abgedeckte Bereich. Der Geberbereich wird bei einer Linearachse von der Baugruppe symmetrisch über den Arbeitsbereich gelegt. D. h., die Baugruppe verschiebt den Geberbereich so, dass die Abstände zwischen den Software-Endschaltern und den Enden des Geberbereichs jeweils gleich sind (siehe Bild unten).
- Der Verfahrbereich ist der Wertebereich, den die FM x52 verarbeiten kann. Der Verfahrbereich ist abhängig von der Auflösung.

Es gilt: Verfahrbereich ≥ Geberbereich ≥ Arbeitsbereich

Das folgende Bild zeigt den Zusammenhang zwischen Software-Endschaltern, Arbeitsbereich, Geberbereich und Verfahrbereich.

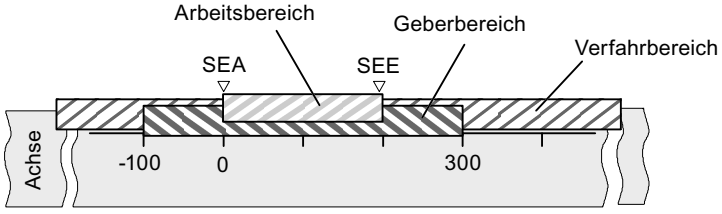

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L# -100 000 000	**Software-Endschalter Anfang**
68.0	SSW_END	DINT	L# 100 000 000	**Software-Endschalter Ende** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm

Ende der Rundachse festlegen

Ende der Rundachse

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Der größte Wert, der bei einer Rundachse angezeigt wird, ist:

Ende der Rundachse [µm] - Auflösung [µm/Impuls] × 1 [Impuls]

Beispiel: Ende der Rundachse 1000 mm

Die Anzeige springt:

- bei positiver Drehrichtung von 999 mm auf 0 mm.
- bei negativer Drehrichtung von 0 mm auf 999 mm.

Rundachse mit Absolutwertgebern

Bei einer Rundachse mit einem Absolutwertgeber muss sich der Rundachsenbereich (0 bis Ende der Rundachse) genau mit der Gesamtschrittzahl des Absolutgebers decken:

$$\text{Rundachsenende [µm]} = \text{Anzahl Umdrehungen (Geber)} * \frac{\text{Schritte (Geber) [Imp]}}{\text{Umdrehung}} * \text{AUFL} \left[\frac{\text{µm}}{\text{Imp}}\right]$$

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
16.0	ENDROTAX	DINT	L#100000	**Ende der Rundachse** Zulässiger Wertebereich: • 1 µm bis +1 000 000 000 µm

Synchronisation festlegen

Art für "Referenzpunkt nachtriggern" auswählen

Referenzpunkt nachtriggern

Mit der Art für "Referenzpunkt nachtriggern (Seite 488)" bestimmen Sie die Bedingungen für die Synchronisation der Achse bei Betrieb mit einem Inkrementalgeber oder einem Initiator.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
52.0	RETR_TYPE	DINT	L#0	**Art für "Referenzpunkt nachtriggern"** Zulässige Werte: • 0 = Referenzpunktschalter und Nullmarke Richtung + • 1 = Referenzpunktschalter und Nullmarke Richtung - • 6 = Nur Referenzpunktschalter • 7 = Nur Nullmarke

Referenzpunktkoordinate festlegen

Referenzpunktkoordinate

- **Inkrementalgeber und Initiator:**
  Durch den Funktionsschalter "Referenzpunkt nachtriggern" und ein Synchronisationsereignis, festgelegt durch die Art für "Referenzpunkt nachtriggern" (Seite 502), wird diesem Ereignis die Referenzpunktkoordinate zugeordnet.
- **Absolutwertgeber (SSI)**
  Eine parametrierte Achse mit einem Absolutwertgeber ist nach der Übertragung des ersten fehlerfreien SSI-Telegramms immer synchronisiert.
  Lesen Sie hierzu auch die Beschreibung der Absolutwertgeberjustage (Seite 504) , die das Zusammenspiel der Absolutwertgeberjustage mit den anderen Daten beschreibt.

Der Wert der Referenzpunktkoordinate muss innerhalb des Arbeitsbereichs liegen:

- **Linearachse**
  Einschließlich der Software-Endschalter.
- **Rundachse**
  Größer oder gleich 0 und kleiner als der Wert "Ende der Rundachse" (0 ≤ Referenzpunktkoordinate < Ende der Rundachse).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis +1 000 000 000 µm

Absolutwertgeberjustage festlegen

Absolutwertgeberjustage ermitteln

Definition

Absolutwertgeberjustage ist nur für einen Absolutwertgeber (SSI) erforderlich.

Mit der Absolutwertgeberjustage und der Referenzpunktkoordinate wird der Wertebereich des Gebers eindeutig auf das Koordinatensystem der Achse abgebildet. Es wird der Geberwert ermittelt, welcher der Referenzpunktkoordinate auf der Achse entspricht. Der Wert muss kleiner sein als die Gesamtschrittzahl des Absolutwertgebers.

Weitere Informationen finden Sie unter "Vorgehen für eine Absolutwertgeberjustage (Seite 504)" sowie in einem Beispiel (Seite 505).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
48.0	ENC_ADJ	DINT	L#0	**Absolutwertgeberjustage** Zulässiger Wertebereich: • 0 bis (2^{25} - 1)

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
98.0	REFPT	DINT	L#0	**Bezugspunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis +1 000 000 000 µm

Vorgehen für eine Absolutwertgeberjustage

Richtige Absolutwertgeberjustage ermitteln

Nach dem ersten Parametrieren müssen Sie weitere Schritte ausführen, damit eine korrekte Beziehung zwischen Geber und Koordinatensystem hergestellt wird. Der Ablauf ist unter Verwendung der Parametrierdialoge dargestellt.

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Ende" sein.
2. Rufen Sie die Einstellung "Bezugspunkt setzen (Seite 480)" mit der Koordinate des unter 1. definierten Punktes auf.
   Die FM x52 ermittelt nun für die im Kanal-DB eingetragene Bezugspunktkoordinate (REFPT im Kanal-DB) einen Geberwert, die Absolutwertgeberjustage. Diesen Wert können Sie mit der Einstellung "Geberdaten (Seite 495)" auslesen.

3. Tragen Sie den ausgelesenen Wert in das Feld "Absolutwertgeberjustage" ein.
4. Speichern Sie Ihre Parametrierung mit der Exportfunktion in den entsprechenden Parameter-DB.
5. Laden Sie die Daten in HWCN zur CPU.
6. Damit die Daten aktiv werden, führen Sie einen Neustart der CPU durch.

Hinweis

Diesen Abgleich führen Sie bei der Inbetriebnahme einmal durch. Die FM x52 ist nach einer Parametrierung im Anlauf synchronisiert, sobald ein vollständiges, fehlerfreies Telegramm des Gebers nach dem Anlauf empfangen wurde.

Alternative: Mechanische Justage eines Gebers

Eine korrekte Beziehung zwischen dem Koordinatensystem und dem Geber erreichen Sie auch folgendermaßen:

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Anfang" sein.
2. Tragen Sie diesen Koordinatenwert in den Maschinendaten als "Referenzpunktkoordinate" ein.
3. Lesen Sie den an dieser Position angezeigten Geberwert mit der Einstellung "Geberdaten (Seite 495)" aus.
4. Tragen Sie diesen Wert als "Absolutwertgeberjustage" in die Maschinendaten ein.

Nach der Parametrierung wird dann immer ein korrekter Istwert angezeigt.

Anstelle der Schritte 3 und 4 können Sie auch den Geber über "Reset", falls vorhanden, auf Null setzen. Dann tragen Sie den Wert "0" als Absolutwertgeberjustage in die Maschinendaten ein.

Beispiel für eine Absolutwertgeberjustage

Beispiel einer Absolutwertgeberjustage

Für das Beispiel gelten folgende Annahmen:

- Referenzpunktkoordinate = -125 mm
- Arbeitsbereich von SSW_STRT = -1000 mm bis SSW_END = 1000 mm
- Absolutwertgeberjustage = 0
- Geberbereich = 2048 Inkremente mit einer Auflösung von 1 mm/Impuls
- Der verwendete Absolutwertgeber lässt sich nicht exakt mechanisch justieren und besitzt auch nicht die Möglichkeit, den Geberwert gezielt zu setzen.

Das folgende Bild zeigt ein Beispiel einer Absolutwertgeberjustage.

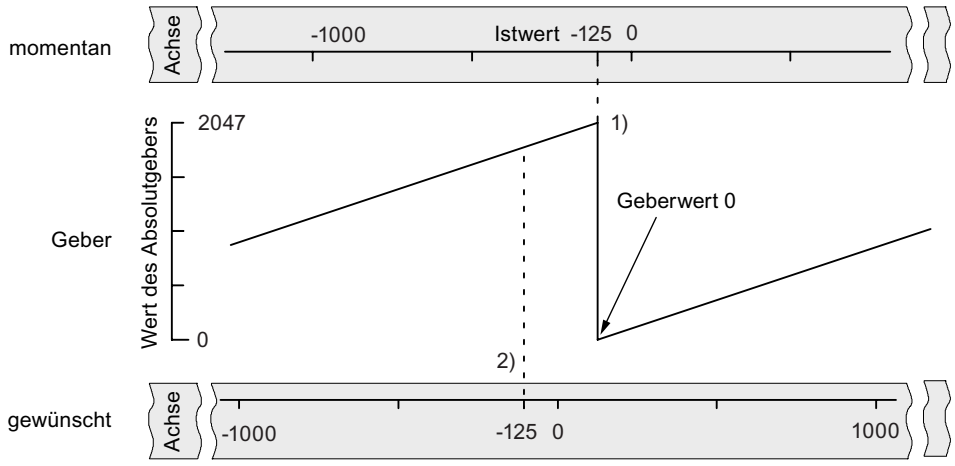

1) Zuordnung des Koordinatensystems zu den Geberwerten mit der eingestellten Absolutwertgeberjustage. Der Geberwert 0 entspricht dem Istwert -125.

2) Gewünschte Zuordnung des Koordinatensystems zum Geber. An dieser Position soll die Koordinate -125 sein.

Ergebnis nach "Bezugspunkt setzen"

Nach "Bezugspunkt setzen" sieht die Beziehung wie folgt aus:

Der Referenzpunktkoordinate auf der Achse (-125) wird der aus der Absolutwertgeberjustage ermittelte Geberwert (1798) zugeordnet. Das folgende Bild zeigt diese Zuordnung.

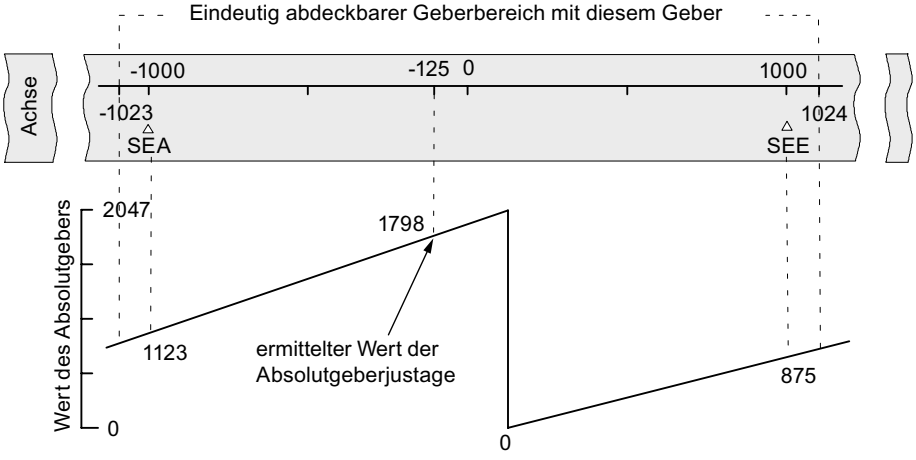

Der Geber liefert 2048 eindeutige Werte. Der Arbeitsbereich wird durch die Software-Endschalter festgelegt. Aufgrund der gewählten Auflösung von 1 mm pro Impuls kann der Geber aber einen größeren Arbeitsbereich abdecken als durch die Software-Endschalter vorgesehen ist.

Bei der eingestellten Auflösung ist der Arbeitsbereich bereits mit 2001 Werten abgedeckt. Es bleiben im Beispiel deshalb 47 Impulse "übrig", die sich symmetrisch um den Arbeitsbereich legen.

Hysterese festlegen

Hysterese

Der Wertebereich der Hysterese (Seite 458) ist abhängig von der Auflösung (Seite 513).

Der maximale Eingabewert ist wie folgt:

- Bei Linearachsen: Maximaler Eingabewert < ¼ des Arbeitsbereichs
- Bei Rundachsen: Maximaler Eingabewert < ¼ des Rundachsenbereichs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
80.0	HYS	DINT	L#0	**Hysterese** Zulässiger Wertebereich: • 0 … 65 535 [Imp] × Auflösung [µm/Imp]

Wegnocken bei Hysterese

Ein Wegnocken schaltet ein, wenn folgende Bedingungen erfüllt sind:

- Der erkannte Istwert liegt innerhalb des Wegnockens.
- Die Hysterese ist nicht aktiv.

Je nach der Position, an welcher der Richtungswechsel stattfindet, kann die Lage eines Schaltpunkts variieren.

Das folgende Bild zeigt das Schaltverhalten eines Wegnockens bei Hysterese.

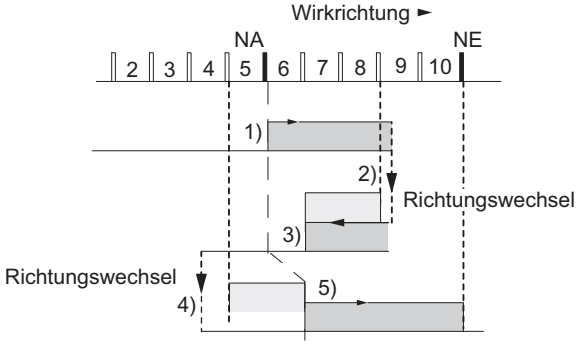

	Nocken
	Hysterese
1)	Nocken schaltet ein, da der Nockenanfang in positiver Wirkrichtung überfahren wird.
2)	Nocken bleibt trotz Richtungswechsel ein, wegen Hysterese.
3)	Nocken schaltet aus bei Ende der Hysterese.
4)	Nocken bleibt aus nach dem Richtungswechsel.
5)	Nach Ende der Hysterese schaltet Nocken wieder ein.

Hinweis

Wegnocken, die kürzer sind als die Hysterese, können bei einem Richtungswechsel von der Hysterese nicht verkürzt werden.

Zeitnocken bei Hysterese

Ein Zeitnocken schaltet ein, wenn folgende Bedingungen erfüllt sind:

- Der Nockenanfang wird in Wirkrichtung überfahren.
- Die Hysterese ist nicht aktiv.

Hinweis

Ist der Bereich zwischen Umkehrpunkt und Nockenanfang des Zeitnockens kleiner als die Hysterese, wird der Zeitnocken von der Hysterese verdeckt.

Das folgende Bild zeigt das Schaltverhalten eines Zeitnockens bei Hysterese. Dargestellt ist ein Zeitnocken, der **nicht** wieder einschaltet.

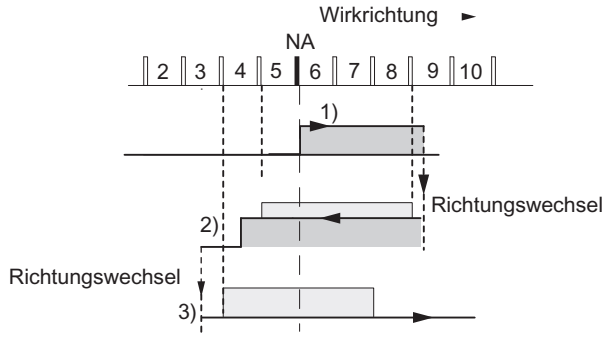

	Nocken
	Hysterese
1)	Nocken schaltet ein, da der Nockenanfang in positiver Wirkrichtung überfahren wird.
2)	Nocken bleibt nach dem Richtungswechsel bis zum Ablauf der Einschaltzeit eingeschaltet, unabhängig von der Hysterese.
3)	Nockenanfang wird von der Hysterese überdeckt, Nocken schaltet nicht.

Simulationsgeschwindigkeit festlegen

Simulationsgeschwindigkeit

Dieses Maschinendatum bestimmt die Simulationsgeschwindigkeit für die "Simulation". Beachten Sie dazu auch die Informationen unter "Einfluss von Einstellungen auf das Schaltverhalten von Zeitnocken (Seite 474)".

Die tatsächliche Simulationsgeschwindigkeit V_{sim} kann von der eingegebenen Simulationsgeschwindigkeit $V_{sim, V}$ abweichen und wird nach der folgenden Formel berechnet:

$$V_{Sim} = \frac{6 * 10exp7 * AUFL}{\text{Ganzzahl}\left(\frac{6 * 10exp7 * AUFL}{V_{Sim, V}}\right)}$$

In dieser Formel bedeutet:

- V_{sim}: Simulationsgeschwindigkeit, die die FM 352 einstellt. Einheit: $^{\mu m}/_{min}$
- $V_{sim, V}$: Simulationsgeschwindigkeit, die Sie in den Maschinendaten vorgeben. Einheit: $^{\mu m}/_{min}$
- AUFL: Auflösung, die sich aus den Geberdaten ermittelt. Einheit: $^{\mu m}/_{Impuls}$
- Ganzzahl ( ): von diesem Ausdruck wird für die weitere Berechnung nur der Vorkommawert weiter berücksichtigt. Dieser Ausdruck muss bei allen Berechnungen im Bereich von 2 … 65536 liegen.

Aufgrund der Zusammenhänge (siehe obige Formel) verändert sich die tatsächliche Simulationsgeschwindigkeit sprunghaft.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	SIM_SPD	DINT	L#0	**Simulationsgeschwindigkeit** Zulässiger Wertebereich: • 0 = Stillstand • 5×10^8 = Größtmögliche Einstellung der Baugruppe Innerhalb dieses Bereichs ist die Simulationsgeschwindigkeit abhängig von der Auflösung (Seite 513): 1000 × Auflösung ≤ Simulationsgeschwindigkeit ≤ 3×10^7 × Auflösung

Minimalen Kantenabstand festlegen

Minimaler Kantenabstand

Mit diesem Maschinendatum definieren Sie einen Bereich nach dem Erkennen des Messbeginns bei einer Kantenerfassung. Wenn das Messende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen.

Der Messbeginn wird erst nach Verfahren des Wegstücks "minimaler Kantenabstand" gemeldet.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	**Minimaler Kantenabstand** Zulässiger Wertebereich: • 0 ... 1 000 000 000 µm

Geber

Signalart auswählen

Signalart

Der Geber liefert Weginformation zur Baugruppe, die diese auswertet und mit der Auflösung in einen Istwert umrechnet.

Sie können die Signalart von einem der folgenden Geber (Seite 462) wählen:

- 5 V Differenzsignal (Seite 463)
  Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B und N, /N einsetzen.
- 24 V Spur A+B phasenversetzt (Seite 463)
  Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.
- SSI absolut (Seite 465)
  Wählen Sie die Signalart "SSI absolut", wenn Sie einen Absolutwertgeber mit serieller Schnittstelle einsetzen.
- 24 V Initiator (Seite 465)
  Wählen Sie die Signalart "24 V Initiator", wenn Sie einen 24 V-Initiator mit dem Signal A* einsetzen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
20.0	ENC_TYPE	DINT	L#1	**Signalart** Zulässige Werte: • 1 = 5 V inkrementell • 2 = 24 V inkrementell • 3 = SSI 13 Bit Telegrammlänge • 4 = SSI 25 Bit Telegrammlänge • 5 = Mithören • 6 = 24 V Initiator vorwärts (positiv) • 7 = 24 V Initiator rückwärts (negativ) • 8 = SSI 13 Bit (rechtsbündig) • 9 = SSI 25 Bit (rechtsbündig) • 10 = Mithören (rechtsbündig)

Auflösung festlegen

Weg pro Geberumdrehung festlegen

Weg pro Geberumdrehung

Mit dem Maschinendatum "Weg pro Geberumdrehung" geben Sie der FM x52 bekannt, welchen Weg das Antriebssystem je Geberumdrehung zurücklegt.

Der Wert "Weg pro Geberumdrehung" ist abhängig vom Aufbau der Achse und von der Art, wie der Geber angebracht ist. Sie müssen alle Übertragungsglieder wie Kupplungen oder Getriebe berücksichtigen. Das folgende Bild zeigt beispielhaft zwei unterschiedliche Geberanordnungen.

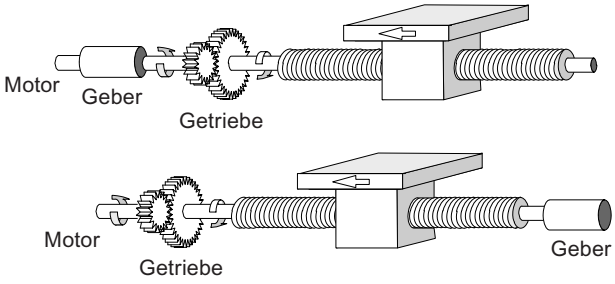

Der Zusammenhang zwischen den Maschinendaten "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" ist unter Auflösung (Seite 513) beschrieben.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
24.0	DISP_REV	DINT	L#80000	**Weg pro Geberumdrehung** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm

Inkremente pro Geberumdrehung festlegen

Inkremente pro Geberumdrehung

Das Maschinendatum "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt. Aus diesem Wert und dem Maschinendatum "Weg pro Geberumdrehung" ermittelt die FM x52 die Auflösung.

Inkrementalgeber

Als Eingabe (Seite 463) ist jeder Wert aus dem Wertebereich möglich. Die Baugruppe wertet die Inkremente 4fach aus.

Initiator

Es ist jeder Wert aus dem Wertebereich möglich.

Absolutwertgeber

Die Grenzen sind bei den einzelnen Geberarten verschieden:

Geberart	Telegrammlänge / -art	Wertebereich	als Linearachse verwendbar
Singleturn-Geber	13 Bit halber Tannenbaum	64 ... 8192 in 2er Potenzen	
Singleturn-Geber	13 Bit rechtsbündig	64 ... 8192 alle Werte	X
Singleturn-Geber	25 Bit rechtsbündig	64 ... 2^{25} alle Werte	X
Multiturn-Geber	25 Bit Tannenbaum	64 ... 8192 in 2er Potenzen	
Multiturn-Geber	25 Bit rechtsbündig	64 ... 2^{24} alle Werte	
Mithören	Tannenbaum	64 ... 8192 in 2er Potenzen	
Mithören	rechtsbündig	64 ... 2^{25} alle Werte	X
Sondereinstellung: Multiturn-Geber als Singleturn-Geber	25 Bit halber Tannenbaum	64 ... 8192 in 2er Potenzen	

Hinweis

Die Anzahl der Impulse eines Gebers ermittelt sich aus den beiden Daten "Inkremente pro Geberumdrehung" mal "Anzahl Umdrehungen" (siehe unter "Auflösung (Seite 513)").

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
32.0	INC_REV	DINT	L#500	**Inkremente pro Geberumdrehung** Zulässiger Wertebereich: • 1 bis 2^{25} Hinweis: Im Maßsystem "Impulse" ist dieser Eintrag ohne Bedeutung.

Auflösung

Auflösung

Die Auflösung ist ein Maß für die Genauigkeit der Nockenbearbeitung. Sie bestimmt auch den möglichen maximalen Verfahrbereich.

Die Auflösung (AUFL) berechnen Sie wie in der folgenden Tabelle gezeigt:

	Inkrementalgeber	Absolutwertgeber / Initiator
Eingangswerte	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – Impulsauswertung: 4fach – 1 Inkrement = 4 Impulse	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung. – 1 Inkrement = 1 Impuls
Berechnung	$\text{AUFL} = \dfrac{\frac{\text{Weg}}{\text{Geberumdrehung}}}{\frac{\text{Impulse}}{\text{Geberumdrehung}}}$	

Hinweis

Im Maßsystem "Impulse" hat die Auflösung immer den Wert 1.

Alle Positionsangaben werden auf das ganzzahlige Vielfache der Auflösung gerundet. Somit können sich die eingegebenen und die verwendeten Werte unterscheiden.

Wertebereich der Auflösung

Den Wertebereich für die Auflösung müssen Sie auf das zugrundeliegende Maßsystem umrechnen. Durch die Wahl der beiden Werte "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" muss die Auflösung in diesen Bereich gelegt werden.

Das gewählte Maßsystem bestimmt den Wertebereich der Auflösung:

Maßsystem	Angaben in ...	Wertebereich der Auflösung
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ mm .... 1000×10^{-3} mm/Impuls
inch	10^{-4} inch	$0{,}1 \times 10^{-4}$ inch .... 1000×10^{-4} inch/Impuls

Maßsystem	Angaben in ...	Wertebereich der Auflösung
grad	10^{-4} grad	$0{,}1 \times 10^{-4}$ grad .... 1000×10^{-4} grad/Impuls
	10^{-3} grad	$0{,}1 \times 10^{-3}$ grad .... 1000×10^{-3} grad/Impuls
	10^{-2} grad	$0{,}1 \times 10^{-2}$ grad .... 1000×10^{-2} grad/Impuls
Impulse	1 Impuls	1

Beispiel

- Ein Inkrementalgeber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 5000
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 4 Impulse

  Damit ergibt sich die Auflösung (4fach-Auswertung):

$$\text{Auflösung} = \frac{1000\ \text{mm}}{5000\ \text{Inkremente}} = 0{,}2000\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{Impulse}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impuls}}$$

- Ein SSI-Geber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 4096
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 1 Impuls

  Damit ergibt sich die Auflösung:

$$\text{Auflösung} = \frac{1000\ \text{mm}}{4096\ \text{Inkremente}} = 0{,}2441\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2441\ \frac{\text{mm}}{\text{Impuls}}$$

Abhängigkeit des Verfahrbereichs von der Auflösung

Der Verfahrbereich wird begrenzt durch die Zahlendarstellung in der FM x52. Diese Zahlendarstellung variiert in Abhängigkeit der Auflösung. Achten Sie deshalb darauf, dass die Vorgaben immer innerhalb der zulässigen Grenzen liegen.

Die folgende Tabelle zeigt den maximalen Verfahrbereich:

Auflösung (AUFL) liegt im Bereich	Maximaler Verfahrbereich
0,1 µm/Impuls ≦ AUFL < 1 µm/Impuls	-10^8 µm bis 10^8 µm (-100 m bis +100 m)
1 µm/Impuls ≦AUFL ≦ 1000 µm/Impuls	-10^9 µm bis 10^9 µm (-1000 m bis +1000 m)

Abhängigkeit der Geschwindigkeit von der Auflösung

Die Geschwindigkeit, die angezeigt wird, kann sich in Abhängigkeit von der Auflösung in den folgenden Grenzen bewegen (die Angaben beziehen sich auf das Maßsystem mm):

- von 1 µm/min bis 90 m/min bei einer Auflösung < 1 µm/Impuls
- von 1 µm/min bis 900 m/min bei einer Auflösung ≥ 1 µm/Impuls

Die Geschwindigkeit wird von der Baugruppe alle 4 ms ermittelt und geglättet.

Sie hat mindestens eine Ungenauigkeit von einem Impuls/4 ms und ist für Regelungszwecke nicht geeignet.

Zählrichtung auswählen

Zählrichtung

Mit dem Maschinendatum "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an. Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

- normal = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen aufsteigenden Lageistwerten
- invertiert = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen absteigenden Lageistwerten

Hinweis:
Eine Vorhaltezeit in Verbindung mit einem Absolutwertgeber (SSI) und invertiert parametrierter Zählrichtung ist nicht zulässig.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
59.0	CNT_DIR	BOOL	FALSE	**Zählrichtung** Zulässige Werte: • 0 = normal • 1 = invertiert

Überwachung auswählen

Überwachung

Drahtbruch

Mit dem Aktivieren der Überwachung überwacht die FM x52 bei einem Inkrementalgeber die Signale A, /A, B, /B, N und /N. Die Überwachung erkennt:

- Drahtbruch
- Kurzschluss der einzelnen Leitungen
  Bei Inkrementalgebern ohne Nullmarken müssen Sie alternativ
  - die Drahtbruch-Überwachung ausschalten
  - die Signale N und /N extern verschalten (Seite 462)
- Flankenabstand der Zählimpulse
- Ausfall der Geberversorgung

Telegrammfehler

Die Überwachung auf Telegrammfehler bei Absolutwertgebern (SSI) ist nicht abschaltbar. Sie überwacht:

- das Telegramm auf Start- und Stoppbitfehler
- die Monoflopzeit des angeschlossen Gebers

Fehlimpulse (Inkrementalgeber)

Ein Inkrementalgeber muss zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Inkremente liefern.

Die FM x52 überprüft, ob die Nullmarke eines Inkrementalgebers zum richtigen Geberstand auftritt.

Für Geber ohne Nullmarken müssen Sie die Überwachung der Fehlimpulse abschalten.

Ebenso müssen Sie die Drahtbruchüberwachung abschalten oder die Nullmarkeneingänge N und /N extern verschalten.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Überwachungen
63.0	MON_WIRE	BOOL	TRUE	1 = Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1 = Telegrammfehler (muss immer 1 sein)
63.2	MON_PULSE	BOOL	TRUE	1 = Fehlimpulse

Daten für Absolutwertgeber (SSI) auswählen

Anzahl Geberumdrehungen

Anzahl Geberumdrehungen

Das Maschinendatum "Anzahl Geberumdrehungen" gibt es nur für Absolutwertgeber. Damit geben Sie die Anzahl der Umdrehungen an, die mit diesem Geber möglich sind. Weitere Informationen zu Absolutwertgebern finden Sie unter "Absolutwertgeber (SSI) (Seite 465)".

- **Singleturn-Geber**
  Es ist nur der Wert 1 möglich.
- **Multiturn-Geber**
  - Multiturn-Geber / Mithören (Tannenbaum): 2 ... 4096 in 2er Potenzen
  - Multiturn-Geber / Mithören (rechtsbündig): 1 ... 4096 alle Werte
- **Linearmaßstab**
  Sie können auch einen Linearmaßstab anschließen. Tragen Sie dazu den Wert 1 ein.

Die **Gesamtschrittzahl des Gebers** ist kein Maschinendatum. Sie errechnet sich wie folgt:

Gesamtschrittzahl = Inkremente pro Geberumdrehung × Anzahl Umdrehungen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.0	NO_REV	DINT	L#1024	**Anzahl Geberumdrehungen** Zulässiger Wertebereich: • 1 (Singleturn-Geber) • 1 bis 4096 (Multiturn-Geber)

Baudrate

Baudrate

Mit dem Maschinendatum "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur FM x52.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
40.0	BAUDRATE	DINT	L#0	**Baudrate** Zulässige Werte: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Telegrammlänge

Telegrammlänge

Mit der "Telegrammlänge" bestimmen Sie den Taktrahmen, den die FM x52 für Absolutwertgeber ausgibt.

Wenn Sie "Mithören" wählen, schalten Sie den Takt der FM x52 aus. Die FM x52 kann dann auch andere SSI-Telegramme mit 13 oder 25 Bit-Telegrammrahmen mithören. Die Geschwindigkeit der Übertragung richtet sich nach der Taktgeschwindigkeit der Masterbaugruppe.

Verwendete Daten im Parameter-DB

Die hier ausgewählte Telegrammlänge ist zusammen mit der Signalart unter dem Datum ENC_TYPE (Seite 510) im Parameter-DB abgelegt.

Richtung Initiator

Richtung Initiator

Mit der "Richtung Initiator" bestimmen Sie die Richtung für einen angeschlossenen Initiator:

- Positiv: entspricht Vorwärtsrichtung (aufsteigende Zählimpulse)
- Negativ: entspricht Rückwärtsrichtung (absteigende Zählimpulse)

Spuren

Ansteuerart der Spurausgänge auswählen

Ansteuerung der Spurausgänge

Mit dem Maschinendatum "Ansteuerung der Spurausgänge" legen Sie die Ansteuerart der Spurausgänge der Spuren 0 bis 12 fest. Die Ansteuerung ist möglich durch:

- Nockensteuerwerk: Die Spursignale werden von der Nockenbearbeitung der FM 352 ein- und ausgeschaltet.
- CPU: Die Spursignale geben direkt die zugehörigen Werte der Spurfreigaben im Kanal-DB wieder.

Damit sind die Spurausgänge von Ihrem Programm aus gezielt ansteuerbar.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
90.0	TRACK_OUT	WORD	W#16#0	**Ansteuerung der Spurausgänge** Zulässige Werte: • 0 = Nockensteuerwerk • 1 = CPU Bitnummer = Spurnummer Die Bits 13 bis 15 müssen 0 sein.

Sonderspuren auswählen

Sonderspuren

Sie können die Spuren 0, 1 und 2 als Sonderspuren parametrieren:

- Spur 0 und Spur 1 als Zählnockenspur (Seite 457)
- Spur 2 als Bremsnockenspur (Seite 457)

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Sonderspuren
99.0	SPEC_TRC0	BOOL	FALSE	1 = Spur 0 ist Zählnockenspur
99.1	SPEC_TRC1	BOOL	FALSE	1 = Spur 1 ist Zählnockenspur
99.2	SPEC_TRC2	BOOL	FALSE	1 = Spur 2 ist Bremsnockenspur

Oberen Zählwert für Zählnockenspur festlegen

Oberer Zählwert für Zählnockenspur

Mit diesem Maschinendatum legen Sie den oberen Zählwert für die parametrierte Zählnockenspur fest.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CNT_LIM0	DINT	L#2	**Oberer Zählwert für Zählnockenspur** (Spur 0)
104.0	CNT_LIM1	DINT	L#2	**Oberer Zählwert für Zählnockenspur** (Spur 1) Zulässiger Wertebereich: • 2 … 65535

Freigabeeingang auswählen

Freigabeeingang

Das Spursignal Q3 wird eingeschaltet, wenn folgende Bedingungen alle erfüllt sind:

- Die Spur ist freigeschaltet.
- Der entsprechende externe Freigabeeingang I3 ist gesetzt.
- Das Spurergebnis der Spur ist 1.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
95.0	EN_IN_I3	BOOL	FALSE	**Freigabeeingang** 1 = Spursignal Spur 3 ist mit Freigabeeingang I3 UND-verknüpft Die Bits 95.1 bis 95.7 müssen 0 sein.

Nocken

Mengengerüst

Mengengerüst

Das Mengengerüst bestimmt die Anzahl der maximal parametrierbaren Nocken und damit die Nockenzykluszeit:

Mengengerüst	Nockenzykluszeit
16 Nocken	20,48 µs
32 Nocken	40,96 µs
64 Nocken	81,92 µs
128 Nocken	163,84 µs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	C_QTY	DINT	UDT3: L#0 UDT4: L#1 UDT5: L#2 UDT6: L#3	Mengengerüst 0 = max. 16 Nocken 1 = max. 32 Nocken 2 = max. 64 Nocken 3 = max. 128 Nocken

Nockendaten

Nockendaten

Nockendaten

Nockendaten beschreiben die Eigenschaften eines Nockens, die Zuordnung jedes Nockens zu einer Spur und das Schaltverhalten des Nockens.

Die nachfolgenden Nockendaten werden für jeden Nocken einzeln eingestellt:

- Gültigkeit (Seite 522)
- Spurnummer (Seite 522)
- Nockenart (Seite 523)
- Nockenanfang und Nockenende für Wegnocken (Seite 523)
- Nockenanfang und Einschaltzeit für Zeitnocken (Seite 524)
- Vorhaltezeit (Seite 525)
- Wirkrichtung (Seite 527)
- Prozessalarm (Seite 529)

Allgemein gilt für die Nockenbearbeitung:

- Nur die "gültig" gesetzten Nocken werden von der Baugruppe interpretiert und verarbeitet.
- Die Nocken 0 ... 7 sind prozessalarmfähig.
- Die Anzahl der parametrierbaren Nocken ist abhängig vom Mengengerüst.

Gültigkeit eines Nockens

Nocken "gültig"

Sie müssen alle Nocken "gültig" setzen, die vom Nockensteuerwerk bearbeitet werden sollen.

Hinweis

Setzen Sie nicht benutzte Nocken auf jeden Fall auf "ungültig" (CAMVALID = FALSE).

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.0	CAMVALID	BOOL	FALSE	1 = Nocken "gültig"

Spurnummer für einen Nocken festlegen

Spurnummer

Mit der Wahl der Spurnummer legen Sie für jeden Nocken die Spur fest, auf die dieser Nocken wirken soll.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+1.0	TRACK_NO	BYTE	B#16#0	**Spurnummer** Zulässiger Wertebereich: • 0 bis 31

Nockenart auswählen

Nockenart

Wählen Sie die gewünschte Nockenart aus:

- Wegnocken
- Zeitnocken

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.3	CAM_TYPE	BOOL	FALSE	**Nockenart** Zulässige Werte: • 0 = Wegnocken • 1 = Zeitnocken

Nockenanfang und Nockenende für Wegnocken festlegen

Nockenanfang (NA) / Nockenende (NE) bei Wegnocken

Nockenanfang und Nockenende legen bei einem Wegnocken die Nockenlänge fest.

Mindestlänge eines Wegnockens

Das folgende Bild zeigt die kürzesten Nocken bei positiver Bewegungsrichtung der Achse.

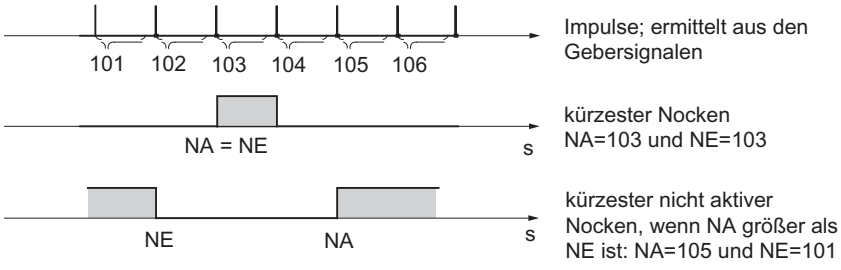

Generell gilt:

- Der nichtaktive Teil eines Nockens muss immer zwischen Nockenende (NE) und Nockenanfang (NA) einen Mindestabstand von 4 Impulsen einhalten.
- Bei NE = NA schaltet der Nocken für einen Impuls ein.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar für Wegnocken
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Nockenanfang (NA)** **Nockenende (NE)** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm

Nockenanfang und Einschaltzeit für Zeitnocken festlegen

Nockenanfang (NA) / Nockenende (NE) ≙ Einschaltzeit bei Zeitnocken

Bei einem Zeitnocken müssen Sie einen Nockenanfang und anstatt des Nockenendes eine Einschaltzeit vorgeben. Die Einschaltzeit geben Sie in 100 µs-Schritten vor. Die Zeit läuft ab dem Einschalten des Nockens.

Für die Vorgabe der Zeiten gelten folgende Bedingungen:

- 0 µs: Ein Nocken mit der Einschaltzeit 0 µs wird nie eingeschaltet
- 0 µs < t ≤ 400 µs: Die FM x52 stellt eine minimale Nockenzeit von ca. 330 µs ein.
- t > 400 µs: Die FM x52 berechnet die tatsächliche Einschaltzeit t_{tats} aus der vorgegebenen Einschaltzeit t_{Vor} nach folgender Formel:

$$t_{tats} = \text{Ganzzahl} \left(\frac{t_{Vor}}{\text{Nockenzykluszeit}} \right) * \text{Nockenzykluszeit}$$

Der maximale Fehler ist immer kleiner als die Nockenzykluszeit.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar für Zeitnocken
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Nockenanfang (NA)** **Nockenende (NE) ≙ Einschaltzeit** Zulässiger Wertebereich für Nockenanfang: • -1 000 000 000 µm bis 1 000 000 000 µm Zulässiger Wertebereich für Einschaltzeit: • (0 bis 13421) × 100 µs bei maximal 16 Nocken • (0 bis 26843) × 100 µs bei maximal 32 Nocken • (0 bis 53686) × 100 µs bei maximal 64 Nocken • (0 bis 65535) × 100 µs bei maximal 128 Nocken

Vorhaltezeit festlegen

Vorhaltezeit

Verzögerungszeiten der angeschlossenen Schaltelemente können Sie durch die Vorgabe einer Vorhaltezeit kompensieren. Die Vorhaltezeit geben Sie in 100 µs-Schritten vor. Sie können pro Nocken eine Vorhaltezeit vergeben. Die Vorhaltezeit gilt für Nockenanfang und Nockenende.

Vorhalteweg

Der Vorhalteweg eines Nockens wird in Abhängigkeit von der aktuellen Geschwindigkeit und der Vorhaltezeit laufend neu berechnet. Um diesen Weg wird der gesamte Nocken in Richtung des Istwerts verschoben. Der parametrierte Bereich wird als "statischer Bereich", der aufgrund der Vorhaltezeit berechnete Bereich als "dynamischer Bereich" bezeichnet.

Vorhalteweg = Vorhaltezeit × aktuelle Geschwindigkeit

Die Berechnung der Vorhaltewege aller Nocken wird innerhalb ¼ der längsten parametrierten Vorhaltezeit in der FM x52 durchgeführt. Wenn Sie für einen Nocken eine sehr große Vorhaltezeit parametrieren, wird die dynamische Verstellung weniger oft berechnet.

Tatsächliche Vorhaltezeit

Die tatsächliche Vorhaltezeit ermitteln Sie folgendermaßen:

1. Ermitteln Sie die Nockenzykluszeit: Dies ist die Zeit, in der die FM x52 alle Nocken komplett abgearbeitet hat. Sie ist abhängig vom parametrierten Mengengerüst.
2. Berechnen Sie die tatsächliche Vorhaltezeit mit folgender Formel:

$$\text{Vorhaltezeit}_{tats} = \text{Ganzzahl}\left(\frac{\text{Vorhaltezeit}_{V}}{\text{Nockenzykluszeit} * 4}\right) * \text{Nockenzykluszeit} * 4$$

   Dabei bedeuten:

   – Vorhaltezeit $_{tats}$ ist die Vorhaltezeit, die von der FM x52 eingestellt wird
   – Vorhaltezeit $_{V}$ ist die Vorhaltezeit, die Sie vorgegeben haben.
   – Ganzzahl ( ) bedeutet, dass von der Berechnung der Klammer nur der Vorkommawert berücksichtigt wird.

Der maximale Fehler der Vorhaltezeit $_{tats}$ ist immer kleiner als Nockenzykluszeit × 4.

Beispiel:

Folgende Werte sind vorgegeben:

- Mengengerüst: maximal 32 Nocken
- Nockenzykluszeit: 40,96 µs
- Vorhaltezeit $_{V}$ = 1000 µs

Ergebnis: Sie erhalten eine tatsächliche Vorhaltezeit von 983 µs.

Hinweis

Die tatsächliche Vorhaltezeit ist immer kleiner als die parametrierte Vorhaltezeit. Sie kann 0 werden, obwohl die parametrierte Vorhaltezeit ≥ 100 µs ist.

Der Vorhalteweg muss bei einer Rundachse kleiner als der Rundachsenbereich und der nicht aktive Teil des Nockens sein. Das muss für alle Geschwindigkeiten sichergestellt werden.

Hinweis

Eine Vorhaltezeit in Verbindung mit einem Absolutgeber (SSI) und invertiert parametrierter Zählrichtung ist nicht zulässig.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+ 10.0	LTIME	INT	0	**Vorhaltezeit** Zulässiger Wertebereich: • (0 bis 53686) × 100 µs bei max. 16 Nocken • (0 bis 65535) × 100 µs bei max. 32, 64 oder 128 Nocken

Dynamische Verstellung auf einem Nocken

Für den Bereich des Nockens unterscheiden Sie zwei Fälle:

1. Der statische und der dynamische Bereich des Nockens überlappen sich.
2. Der statische und der dynamische Bereich des Nockens überlappen sich nicht.

Die folgende Tabelle zeigt für die beiden Fälle die Ermittlung der dynamischen Verstellung.

Dynamische Verstellung	Beschreibung
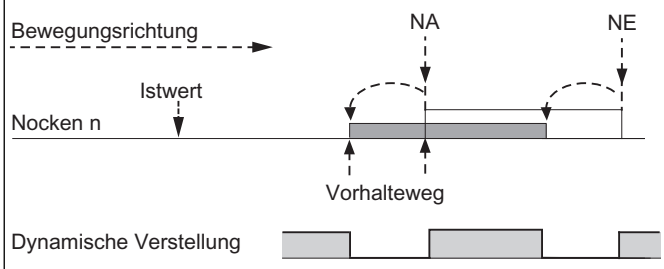	Wenn der dynamische Bereich den statischen Bereich des Nockens überlappt, gilt Folgendes: • Wenn der dynamische Bereich des Nockens erreicht ist, schaltet der Nocken ein. Gleichzeitig wird die Ermittlung einer neuen dynamischen Verstellung gesperrt. • Erreicht der Istwert den statischen Bereich des Nockens, wird die Ermittlung einer neuen dynamischen Verstellung wieder freigegeben, eine Geschwindigkeitsänderung wirkt sich auf das Nockenende aus. • Schaltet der Nocken am Ende des dynamischen Bereichs aus, ist die dynamische Verstellung bis zum Ende des statischen Bereichs des Nockens wieder gesperrt.
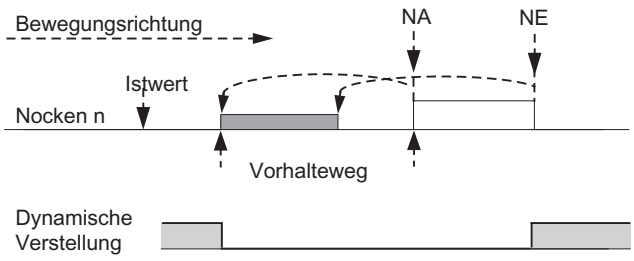	Wenn der dynamische Bereich den statischen Bereich des Nockens nicht überlappt, gilt Folgendes: • Wenn der dynamische Bereich des Nockens erreicht ist, schaltet der Nocken ein. Gleichzeitig wird die Ermittlung einer neuen dynamischen Verstellung gesperrt. • Am Ende des statischen Bereichs des Nockens wird die dynamische Verstellung wieder freigegeben.
Dynamischer Bereich Statischer Bereich Eine **neue** dynamische Verstellung ist möglich	

Hinweis

Durch eine Änderung der Drehrichtung wird die Ermittlung der dynamischen Verstellung wieder freigegeben.

Wirkrichtung auswählen

Wirkrichtung

Zwei Wirkrichtungen sind möglich:

- Positiv: Der Nocken wird am Nockenanfang eingeschaltet, wenn die Achse in Richtung steigender Istwerte läuft.
- Negativ: Der Nocken wird am Nockenende eingeschaltet, wenn die Achse in Richtung fallender Istwerte läuft.

Sie können auch beide Wirkrichtungen gleichzeitig einstellen.

Mit der Auswahl "Keine" schalten Sie den Nocken wirkungslos.

Schaltverhalten von Nocken in Abhängigkeit von der Wirkrichtung

Die folgende Tabelle zeigt das Schaltverhalten von Weg- und Zeitnocken in Abhängigkeit von der Wirkrichtung. Mit Ausnahme von Beispiel 5 wird immer die positive Wirkrichtung angenommen.

Nr.	Beschreibung	Wegnocken	Zeitnocken
1	Ein Nocken wird in Wirkrichtung überfahren	NA NE	NA Einschaltzeit
2	Ein Nocken wird entgegen der Wirkrichtung überfahren	NA NE	NA Einschaltzeit
3	Ein Nocken wird in Wirkrichtung angefahren; während der Nocken geschaltet ist, wechselt die Bewegungsrichtung der Achse	NA NE	NA Einschaltzeit x t1 t2 Einschaltzeit x = t1 + t2
4	Ein Nocken wird entgegen der Wirkrichtung angefahren; auf dem Nocken wechselt die Bewegungsrichtung der Achse in Wirkrichtung	NA NE	Nocken wird nicht geschaltet
5	Ein Nocken wird in beliebiger Richtung angefahren und in beliebiger Richtung verlassen; die Wirkrichtung ist auf **beide Richtungen** eingestellt	NA NE	NA Einschaltzeit x t1 t2 t3 Einschaltzeit x = t1 + t2 + t3

parametrierter Nocken
geschalteter Nocken

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.1	EFFDIR_P	BOOL	TRUE	1 = Wirkrichtung positiv (plus)
+0.2	EFFDIR_M	BOOL	TRUE	1 = Wirkrichtung negativ (minus)

Prozessalarm für Nocken auswählen

Prozessalarm beim Ein- / Ausschalten von Nocken

Sie können für die Nocken 0 bis 7 einstellen, ob beim Einschalten, Ausschalten oder Ein- und Ausschalten des Nockens ein Prozessalarm generiert werden soll.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.4	PI_SW_ON	BOOL	FALSE	1 = Prozessalarm beim Einschalten
+0.5	PI_SW_OFF	BOOL	FALSE	1 = Prozessalarm beim Ausschalten

Freigabe Prozessalarme

Prozessalarme freigeben

Prozessalarme freigeben

Mit diesem Maschinendatum geben Sie Prozessalarme frei.

Sie können einstellen, wann Prozessalarme generiert werden sollen:

- Nocken ein/aus
  Sie können in den Nockendaten (Seite 529) einstellen, ob Prozessalarme beim Ein- und/ oder Ausschalten der Nocken 0 bis 7 generiert werden sollen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
3.2	PI_CAM	BOOL	FALSE	1 = Prozessalarm freigeben: Nocken ein/aus

Auswertung eines Prozessalarms

Wenn ein Prozessalarm von der FM 352 ausgelöst wird, dann steht in der Variablen OB4x_POINT_ADDR des Prozessalarm-OB die Alarminformation zur Verfügung.

Die folgende Tabelle zeigt den Inhalt des Doppelworts OB4x_POINT_ADDR.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	Nocken	0	0
2	Nocken 7 ein	Nocken 7 aus	Nocken 6 ein	Nocken 6 aus	Nocken 5 ein	Nocken 5 aus	Nocken 4 ein	Nocken 4 aus
3	Nocken 3 ein	Nocken 3 aus	Nocken 2 ein	Nocken 2 aus	Nocken 1 ein	Nocken 1 aus	Nocken 0 ein	Nocken 0 aus

Die Alarmursache können Sie aus Byte 1 entnehmen.

Nocken: Werten Sie Byte 2 und 3 entsprechend der Tabelle aus.

Verlorene Prozessalarme

Ist die Bearbeitung eines Prozessalarms im Prozessalarm-OB noch nicht abgeschlossen, merkt sich die Baugruppe alle folgenden Prozessalarmereignisse. Tritt ein Ereignis ein weiteres Mal auf, bevor der Prozessalarm ausgelöst werden konnte, löst die Baugruppe den Diagnosealarm "Prozessalarm verloren" aus.

Steuer- und Rückmeldeschnittstelle der FM 352

Anwendung

In speziellen Anwendungen oder in einer Alarmebene ist ein besonders schneller Zugriff auf Rückmelde- und Steuersignale erforderlich. Diese Daten erreichen Sie direkt über die Ein- und Ausgangsbereiche der Baugruppe.

Zur Anlaufkoordinierung nach jedem Anlauf der Baugruppe, z. B. nach Baugruppe stecken oder nach einem STOP-RUN-Übergang der CPU, müssen Sie die Anweisung CAM_CTRL solange aufrufen, bis durch RETVAL = 0 das Ende des Anlaufs angezeigt wird.

Hinweis

Wenn Sie direkt auf Daten der FM 352 zugreifen, dürfen Sie nur die hier beschriebenen nicht internen Daten in der angegebenen Weise verwenden. Ansonsten kann der Zugriff Ihres Anwenderprogramms auf die Baugruppe gestört werden.

Rückmeldesignale lesen durch Direktzugrifff

Die Byte-Adressen sind relativ zur Ausgangsadresse der Baugruppe angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

In AWL greifen Sie mit den Befehlen PEB (1 Byte lesen) und PED (4 Byte lesen) auf die Daten zu.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	PARA	intern	intern	DATA_ERR	intern	DIAG	intern	intern

Adresse	Bitnummer							
Byte 1	0	0	0	CAM_ACT	0	0	0	0
Byte 2	intern							
Byte 3	0	0	FVAL_DONE	HYS	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								
Byte 8	TRACK_OUT							
Byte 9								
Byte 10								
Byte 11								

Steuersignale schreiben durch Direktzugriff

Die Byte-Adressen sind relativ zur Eingangsadresse der Baugruppe angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

In AWL greifen Sie auf die Daten mit den Befehlen PAB (1 Byte schreiben) und PAW (2 Byte schreiben) zu.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	intern							
Byte 1	0	CNTC1_EN	CNTC0_EN	CAM_EN	DIR_P	DIR_M	0	0
Byte 2	TRACK_EN							
Byte 3								

Beispiel: Lageistwert ACT_POS

Die Anfangsadresse der Baugruppe ist 512.

AWL

```
L PED 516      //Aktuellen Lageistwert (ACT_POS) mit Direktzugriff lesen:
               //Anfangsadresse der Baugruppe + 4
```

Parameterübertragungswege

Unter Parameter werden nachfolgend Maschinen- und Nockendaten verstanden.

Das folgende Bild zeigt die Parameterübertragungswege zur FM 352.

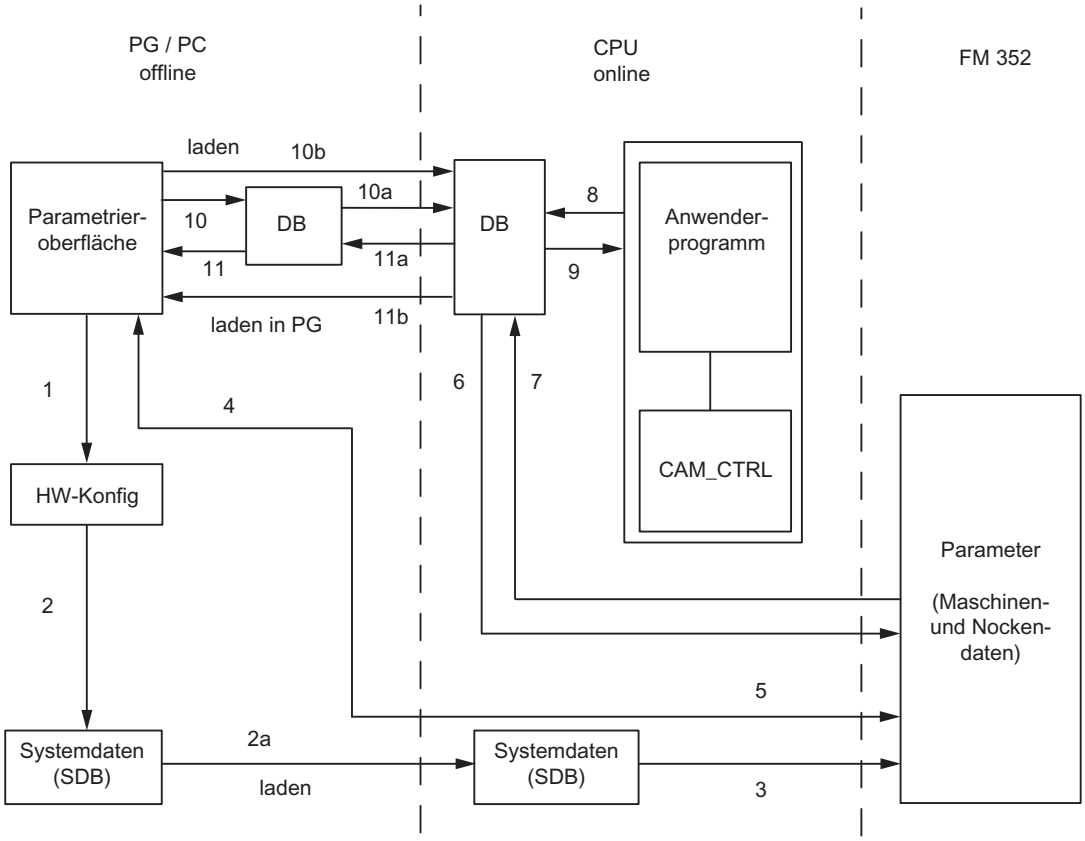

1 Parameter in der Parametrieroberfläche speichern.

2 HW-Konfiguration speichern, übersetzen und zur CPU laden.

3 Die CPU schreibt die Parameter bei Systemparametrierungen zur Baugruppe.

4 Parameter der Baugruppe ins PG laden mit dem Befehl "Zielsystem laden in PG".

5 Parameter aus der Parametrieroberfläche in die Baugruppe laden mit dem Befehl "Zielsystem laden".

6 Parameter über Aufträge des Anwenderprogramms in die Baugruppe schreiben.

7 Parameter über Aufträge des Anwenderprogramms aus der Baugruppe lesen.

8 Parameter aus dem Anwenderprogramm in den Online-DB ablegen.

9 Parameter aus dem Online-DB in das Anwenderprogramm einlesen.

10 Parameter aus der Parametrieroberfläche in den DB exportieren (Offline- oder Online-DB); ein Offline-DB muss anschließend noch in die CPU geladen werden.

11 Parameter aus einem Online- oder Offline-DB in die Parametrieroberfläche importieren.

Einige Anwendungsfälle für die Übertragung von Parametern

Anwendungsfall	Schritte
Sie bearbeiten die Parameter mit der Parametrieroberfläche. Die Baugruppe soll anschließend automatisch beim Anlauf parametriert werden.	Führen Sie die Schritte 1, 2, 3 aus.
Sie ändern Parameter bei der Inbetriebnahme im Testbetrieb in der Parametrieroberfläche.	Führen Sie die Schritte 4, 5 aus

Anwendungsfall	Schritte
Die bei der Inbetriebnahme veränderten Parameter sollen anschließend beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 1, 2, 3 aus.
Sie erstellen die Parameter mit der Parametrieroberfläche. Die Baugruppe soll beim Anlauf nur vom Anwenderprogramm über Datenbausteine parametriert werden.	Führen Sie die Schritte 10, 6 aus.
Sie möchten komfortabel Vorratsdaten für Rezepturen erstellen.	Führen Sie den Schritt 10 aus.
Sie erstellen die Parameter mit der Parametrieroberfläche. Diese sollen dem Anwenderprogramm für temporäre Änderungen zur Verfügung stehen.	Führen Sie die Schritte 1, 2, 3 für die automatische Parametrierung aus.
	Führen Sie die Schritte 10, 7 für den Zugriff durch das Anwenderprogramm aus.
Sie ändern vorhandene Parameter (ausschließlich) über das Anwenderprogramm.	Führen Sie die Schritte 7, 9, 8, 6 aus.
Sie möchten die über das Anwenderprogramm geänderten Daten mit der Parametrieroberfläche ansehen.	Führen Sie den Schritt 11 aus.
Die vom Anwenderprogramm veränderten Parameter sollen auch beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 6, 11, 1, 2, 3 aus.

2.1.3.5 FM 352 in Betrieb nehmen

2.1.3.6 FM 352 aus dem Anwenderprogramm verwenden

Einbinden der FM 352 in ein Anwenderprogramm

Für das Einbinden der FM 352 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen CAM_INIT und CAM_CTRL werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 352. Der Kanal-DB wird automatisch als Instanz-DB zur Anweisung CAM_CTRL angelegt, wenn Sie die Anweisung CAM_CTRL in das Projekt einbinden.

Alle zu einem Kanal der Baugruppe gehörenden Parameterdaten (Maschinendaten, Nockendaten) liegen im Parameter-DB für die Baugruppe. Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CAM_PxxxTYPE in Ihr Projekt ein. Abhängig vom Mengengerüst benötigen Sie folgende DB-Typen:

- Bis zu 16 Nocken: CAM_P016TYPE
- Bis zu 32 Nocken: CAM_P032TYPE
- Bis zu 64 Nocken: CAM_P064TYPE
- Bis zu 128 Nocken: CAM_P128TYPE

Außerdem gibt es für die FM 352 die Anweisung CAM_DIAG, mit der Sie Diagnoseinformationen in den Diagnose-DB übertragen können. Der Diagnose-DB wird

automatisch als Instanz-DB zur Anweisung CAM_DIAG angelegt, wenn Sie die Anweisung CAM_DIAG in das Projekt einbinden.

Inbetriebnahme der Baugruppe

Auch ohne Anwenderprogramm ist ein erster Test der Baugruppe mit dem Dialog "Inbetriebnahme" möglich. Beachten Sie dabei insbesondere die nachfolgende Warnung.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Zur Vermeidung von Schäden an Personen und Gegenständen beachten Sie folgende Punkte:

- Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners.
  Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Software-Ausfalls die Anlage sicher ausgeschaltet werden kann.
- Installieren Sie Hardware-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
- Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.
- Durch paralleles Steuern und Beobachten der FM 352 von Ihrem Programm und aus dem Dialog "Inbetriebnahme" kann es zu Konflikten kommen, deren Auswirkungen nicht eindeutig sind. Schalten Sie deshalb immer die CPU in den STOP-Zustand, wenn Sie mit dem Dialog "Inbetriebnahme" arbeiten, oder deaktivieren Sie Ihr Programm.

2.1.3.7 FM 352: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Sammelfehler-LED der FM 352

Wo wird der Fehler angezeigt?

Leuchtet die rote Sammelfehler-LED, dann ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten. Die LED leuchtet auch bei kurzfristig auftretenden Fehlern mindestens 3 s lang.

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Sammelfehler-LED angezeigt werden.

LED	Bedeutung	Erläuterungen
SF	Sammelfehler für interne und externe Fehler	Diese LED zeigt folgende Fehlerzustände der FM 352 an: • Prozessalarm verloren • Watchdog abgelaufen • FM 352 nicht projektiert • FM 352 fehlerhaft parametriert • externe Hilfsspannung 24 V fehlt • Frontstecker fehlt • Drahtbruch Geber • Betriebsfehler • Telegrammfehler Absolutwertgeber • Fehlimpuls Inkrementalgeber oder Nullmarke fehlt Zur Beseitigung des Fehlers siehe Fehlerklassen (Seite 540).

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Alarmbearbeitung

Die FM 352 kann Prozessalarme und Diagnosealarme auslösen. Diese Alarme bearbeiten Sie in einem Alarm-OB. Wenn ein Alarm ausgelöst wird, ohne dass der zugehörige OB geladen ist, geht die CPU in STOP.

Die Bearbeitung der Diagnosealarme geben Sie folgendermaßen frei:

1. Wählen Sie die Baugruppe in HWCN aus
2. Geben Sie über "Bearbeiten > Objekteigenschaften > Grundparameter" den Diagnosealarm frei.
3. Speichern und übersetzen Sie die Konfiguration.
4. Laden Sie die Konfiguration in die CPU.

Übersicht über die Diagnosealarme

Folgende Ereignisse und Fehler lösen einen Diagnosealarm aus:

- Betriebsfehler
- Fehlerhafte Maschinendaten
- Fehlerhafte Nockendaten
- Diagnosefehler

Sie finden diese Fehler unter "Fehlerklassen (Seite 540)" ausführlich erläutert.

Reaktion der Baugruppe bei einem Fehler mit Diagnosealarm

- Die Nockenbearbeitung wird abgeschaltet.
- Die Synchronisation wird bei folgenden Diagnosealarmen gelöscht:
  - Frontstecker fehlt, externe Hilfsspannung fehlt
  - ein Nullmarkenfehler wurde erkannt, Leitungsfehler (5 V-Gebersignale)
  - der Verfahrbereich wurde verlassen (wird durch einen Betriebsfehler gemeldet)
  - Istwert setzen nicht ausführbar (wird durch einen Betriebsfehler gemeldet)
- Steuersignale werden, bis auf eine Ausnahme, nicht mehr bearbeitet.
  Ausnahme:
  Bei Überfahren der Software-Endschalter ist ein Richtungswechsel im Simulationsbetrieb noch möglich.
- Funktionsschalter und Aufträge werden weiter bearbeitet.

Baugruppe erkennt einen Fehler ("kommend")

Ein Diagnosealarm ist "kommend", wenn mindestens ein Fehler ansteht. Wurden nicht alle Fehler beseitigt, werden die restlichen anstehenden Fehler nochmals "kommend" gemeldet.

Ablauf:

1. Die FM 352 erkennt einen oder mehrere Fehler und löst einen Diagnosealarm aus. Die Fehler-LED leuchtet. Der Fehler wird in den Diagnosepuffer eingetragen.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Sie können die Startinfo des OB 82 auswerten.
4. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
5. Weitere Informationen erhalten Sie, wenn Sie die Anweisung CAM_DIAG aufrufen.

Baugruppe erkennt den Übergang in den fehlerfreien Zustand ("gehend")

Ein Diagnosealarm ist nur dann "gehend", wenn der letzte Fehler auf der Baugruppe behoben wurde.

Ablauf:

1. Die FM 352 erkennt, dass alle Fehler behoben wurden und löst einen Diagnosealarm aus. Die Fehler-LED leuchtet nicht mehr. Der Diagnosepuffer wird nicht verändert.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
4. Werten Sie das Bit OB82_MDL_DEFECT aus.
   Wenn dieses Bit "0" ist, sind keine Fehler auf der Baugruppe vorhanden. Ihre Auswertung kann hier enden.

Diagnosealarme in Abhängigkeit des CPU-Status

- Im STOP-Zustand der CPU sind die Diagnosealarme von der FM 352 gesperrt
- Werden im STOP-Zustand der CPU nicht alle der anstehenden Fehler behoben, meldet die FM 352 die noch nicht beseitigten Fehler noch einmal nach dem Übergang in den RUN-Zustand als "kommend".
- Werden alle anstehenden Fehler im STOP-Zustand der CPU behoben, dann wird der fehlerfreie Zustand der FM 352 nach dem Übergang in den RUN-Zustand **nicht** mit einem Diagnosealarm gemeldet.

Fehlerauswertung im Anwenderprogramm

Fehlerreaktion im Anwenderprogramm

Im Anwenderprogramm können Sie gezielt auf Fehler reagieren. Dafür stehen Ihnen folgende Mittel zur Verfügung:

- Die Rückgabewerte RETVAL der eingebundenen Anweisungen als Sammelanzeige für Fehler, die während des Ablaufs der Anweisungen aufgetreten sind.
- Die Fehlerbits _ERR der Aufträge als Sammelanzeige für Fehler, die während der Bearbeitung eines Auftrags aufgetreten sind.
- Das Fehlerbit DATA_ERR als Sammelanzeige für einen Fehler, den die FM 352 bei einem Schreibauftrag erkannt hat.
- Die Fehlerkennung in JOB_ERR für die Fehlerursache bei der Kommunikation zwischen Anweisung und FM 352.
- Die Anweisung CAM_DIAG zum Auslesen des Diagnosepuffers der FM 352. Hier können Sie die Fehlerursachen für Aufträge und asynchrone Ereignisse (Betriebsfehler, Diagnosefehler) erfahren.
- Diagnosealarme zur schnellen Reaktion auf Ereignisse.

Diagnosepuffer der Baugruppe

Diagnoseereignisse

Der Diagnosepuffer der Baugruppe enthält maximal 4 Diagnoseereignisse und ist als Ringpuffer organisiert.

Ein Diagnoseereignis wird in den Puffer geschrieben, wenn eine (Fehler-) Meldung "kommend" erkannt wird. Dies kann eine Meldung, ein synchroner Fehler (Datenfehler) oder auch ein asynchroner Fehler (Betriebsfehler und Diagnosefehler) sein. Aus einer Fehlerursache können auch mehrere Einträge als Folgefehler entstehen. Gehende Meldungen erzeugen keine Einträge im Diagnosepuffer.

Für jedes Diagnoseereignis werden angegeben:

- Status (immer kommend)
- Interner Fehler

- Externer Fehler
- Fehlerklasse
- Fehlernummer
- Kanalnummer
- Nockennummer (bei Nockendatenfehler)

Wenn ein Diagnoseereignis in den Diagnosepuffer geschrieben wird, wird das Rückmeldesignal DIAG = 1 gesetzt.

Der Diagnosepuffer kann als Ganzes mit der Anweisung CAM_DIAG in einen Datenbaustein (Diagnose-DB) übertragen werden oder über den Fehlerauswertungsdialog der Projektiersoftware angezeigt werden. Wird der Diagnosepuffer gelesen, setzt die Baugruppe das Rückmeldesignal DIAG = 0.

Hinweis

Wird der Diagnosepuffer gleichzeitig von der Anweisung CAM_DIAG und dem Fehlerauswertungsdialog gelesen, kann es sein, dass ein neu eingetroffenes Diagnoseereignis vom Programm nicht erkannt wird.

Daten und Aufbau des Diagnose-DB

Daten und Aufbau des Diagnose-DB

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Die folgende Tabelle zeigt den Aufbau des Diagnose-DB.

Adresse	Name	Typ	Anfangswert	Kommentar
0.0	MOD_ADDR (Eintragen!)	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Kommunikationsfehler
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten letzter Eintrag

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 4: Datenfehler 5: Maschinendatenfehler 7: Nockendatenfehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer 0 … 255
+6.0	CH_NO	INT	0	Kanalnummer (immer 1)
+8.0	CAMNO	INT	0	Nockennummer 0 bis 127 bei Fehlerklasse = Nockendatenfehler

Liste der JOB_ERR-Meldungen

JOB_ERR-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80A0	32928	-32608	Negative Quittung beim Lesen von Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2 [1)]	32930	-32606	Protokollfehler bei Layer 2 (Datentransfer innerhalb PROFINET / PROFIBUS DP unterbrochen, z. B. wegen Drahtbruch, fehlendem Abschlussstecker, Parametrierfehler etc.)
80A3 [1)]	32931	-32605	Protokollfehler bei User-Interface/User (Datentransfer innerhalb PROFINET / PROFIBUS DP unterbrochen, z. B. wegen Drahtbruch, fehlendem Abschlussstecker, Parametrierfehler etc.)
80A4 [1)]	32932	-32604	Kommunikation am K-Bus gestört
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0 [1)]	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1 [1)]	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2 [1)]	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3 [1)]	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4 [1)]	32964	-32572	Kommunikationsfehler
80C5 [1)]	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6 [1)]	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)

[1] Die Fehler 80A2 … 80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein. Meldungen der Form 7xxx zeigen temporäre Betriebszustände der Kommunikation an.

Fehlerklassen

Betriebsfehler

Fehlerklasse 1: Betriebsfehler

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt.

Die folgende Tabelle zeigt die möglichen Betriebsfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
2	**Software-Endschalter Ende überfahren**		ja
3	**Verfahrbereichsanfang überfahren**		ja
4	**Verfahrbereichsende überfahren**		ja
13	**Fliegendes Istwertsetzen nicht ausführbar**		ja
	Ursache	Die Software-Endschalter liegen nach dem fliegenden Istwertsetzen außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen ist größer als ±100 m bzw. ±1000 m.	
	Wirkung	Achse nicht synchronisiert.	

Datenfehler

Fehlerklasse 4: Datenfehler

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt.

Die folgende Tabelle zeigt die möglichen Datenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
10	**Fehlerhafte Nullpunktverschiebung**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten-Aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen (fehlerfreien) Maschinendaten auf der Baugruppe vorhanden.	
21	**Fliegendes Istwertsetzen nicht zulässig**		nein
	Ursache	Es wurde versucht, "fliegendes Istwertsetzen" bei eingeschaltetem "Referenzpunkt nachtriggern" auszuführen.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0. Es wurde versucht, "Längenmessung" und "Kantenerfassung" gleichzeitig anzuwählen.	
28	**Referenzpunkt nachtriggern nicht zulässig**		nein
	Ursache	Es wurde versucht, "Referenzpunkt nachtriggern" bei anstehendem "fliegendes Istwertsetzen" auszuführen. Es wurde versucht, "Referenzpunkt nachtriggern" bei einem SSI-Geber auszuführen.	
29	**Unerlaubtes bitcodiertes Kommando**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
30	**Falsche Vorhaltezeit**		nein
31	**Falsche Nockennummer**		nein
	Ursache	Der Nocken ist nicht gültig. Die Nockennummer ist nicht im Bereich 0 … 127.	

Nr.	Bedeutung		Diagnosealarm
32	**Falscher Nockenanfang**		nein
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Nockenanfang ist < 0 oder größer als das Ende der Rundachse.	
33	**Falsches Nockenende / falsche Einschaltzeit**		nein
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m) . Rundachse: Das Nockenende ist < 0 oder größer als das Ende der Rundachse. Der Nocken ist nicht mindestens für die Dauer eines Impulses eingeschaltet. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereiches liegen.	
35	**Fehlerhafte Istwertvorgabe beim Istwertsetzen / fliegendes Istwertsetzen**		nein
	Ursache	Die Istwertvorgabe liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw ±1000 m. Die Software-Endschalter würden nach der Ausführung der Einstellung außerhalb des Verfahrbereichs (-100 m … +100m bzw. -1000 m … +1000 m) liegen. Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen wäre größer als ±100 m bzw. ±1000 m.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind keine Maschinendaten vorhanden. Auf der Achse sind keine Maschinendaten aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Eine der Einstellungen "Istwertsetzen" oder "fliegendes Istwertsetzen" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	
109	**Nockenbearbeitung läuft.**		nein
110	**Falsche Anzahl zu ändernder Nocken**		nein

Maschinendatenfehler

Fehlerklasse 5: Maschinendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Die folgende Tabelle zeigt die möglichen Maschinendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	

Nr.	Bedeutung		Diagnosealarm
6	**Falscher minimaler Kantenabstand**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben.	
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben.	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 10.	
11	**Falscher Weg pro Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Anzahl Inkremente/Geberumdrehung** (siehe "Inkremente pro Geberumdrehung (Seite 512)")		ja
14	**Falsche Anzahl Umdrehungen** (siehe "Anzahl Geberumdrehungen (Seite 517)")		ja
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereiches von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m je nach Auflösung. Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Weggeber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art für "Referenzpunkt nachtriggern"**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 6 und 7 angegeben.	
19	**Falsche Richtungsanpassung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung auf Telegrammfehler im Parameter-DB auf "FALSE" gesetzt.	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw -1000 m (je nach Auflösung).	
22	**Falscher Software-Endschalter Ende**		ja

Nr.	Bedeutung		Diagnosealarm
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
144	**Falsches Mengengerüst**		ja
	Ursache	Sie haben für das Mengengerüst einen Wert ungleich 0 bis 3 angegeben.	
145	**Falsche Hysterese**		ja
	Ursache	Die Hysterese liegt außerhalb des Bereichs 0 … 65535 × Auflösung. Die Hysterese ist größer als ¼ × Arbeitsbereich bzw. ¼ × Rundachsenbereich.	
146	**Falsche Simulationsgeschwindigkeit**		ja
	Ursache	Die Simulationsgeschwindigkeit liegt außerhalb des Bereichs 1000 × AUFL bis 3×10^7 × AUFL oder ist größer als 5×10^8 µm/min. Die Simulationsgeschwindigkeit ist intern nicht einstellbar.	
147	**Falsche Spur**		ja
	Ursache	Die Ansteuerung einer Spur außerhalb 0 … 15 (Bit 0 … 15) wurde gewählt.	
148	**Falsche Anwahl der Freigabeeingänge**		ja
	Ursache	Sie wollten eine Spur außerhalb 3 … 10 (Bit 0 … 7) durch ein externes Signal freischalten.	
149	**Falsche Sonderspuranwahl**		ja
	Ursache	Sie wollten eine Spur außerhalb 0, 1 und 2 (Bit 0, 1 und 2) als Sonderspur definieren.	
150	**Falscher oberer Zählwert Spur 0**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
151	**Falscher oberer Zählwert Spur 1**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben einen Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Weggeber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Weggeber und Rundachse: Der Weggeber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Weggeber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Nockendatenfehler

Fehlerklasse 7: Nockendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Die folgende Tabelle zeigt die möglichen Nockendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Prozessalarm unzulässig**		ja
	Ursache	Sie wollen einen Prozessalarm bei einem Nocken mit einer Nockennummer > 7 angeben.	
2	**Falsche Spurnummer**		ja
	Ursache	Die Spurnummer liegt außerhalb des Bereichs 0 bis 31.	
3	**Falscher Nockenanfang**		ja
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Nockenanfang ist < 0 oder größer als das Ende der Rundachse.	
4	**Falsches Nockenende**		ja
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Der Nocken ist nicht mindestens 1 Impuls lang. Rundachse: Das Nockenende ist < 0 oder größer als das Ende der Rundachse. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
5	**Falsche Einschaltzeit**		ja
	Ursache	Die Einschaltzeit ist < 0 µs. Der Maximalwert (Seite 524) ist abhängig vom Mengengerüst.	
6	**Falsche Vorhaltezeit** [1]		ja
	Ursache	Die Vorhaltezeit ist < 0 µs. Der Maximalwert (Seite 525) ist abhängig vom Mengengerüst.	
50	**Zuviele Nockensätze**		ja
	Ursache	Sie wollten mehr Nockensätze eingeben als bei diesem Mengengerüst möglich sind.	
51	**Achse in Betrieb**		ja
	Ursache	Sie wollten bei eingeschaltetem Nockensteuerwerk Nockensätze eingeben.	
52	**Achse nicht parametriert**		ja
	Ursache	Sie wollen Nockendaten eingeben, obwohl noch keine Maschinendaten aktiv sind.	

[1] Die Fehlermeldung kann auch dann auftreten, wenn Sie in Verbindung mit einem Absolutwertgeber (SSI) als Zählrichtung "invertiert" parametriert haben.

Meldungen

Fehlerklasse 15: Meldungen

Die folgende Tabelle zeigt die möglichen Meldungen und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	

Diagnosefehler

Fehlerklasse 128: Diagnosefehler

Die folgende Tabelle zeigt die möglichen Diagnosefehler mit ihren Ursachen und Wirkungen sowie Abhilfemöglichkeiten.

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen, Frontstecker fehlt	
	Wirkung	Siehe "Auslösen von Diagnosealarmen (Seite 535)": • die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern • Die FM 352 ist nicht parametriert (Rückmeldesignal PARA = 0).	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (Wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
51	**Watchdog abgelaufen**		ja
	Ursache	• starke Störeinflüsse auf die FM 352 • Fehler in der FM 352	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind • Tauschen der FM 352	

Nr.	Bedeutung		Diagnosealarm
52	**Baugruppeninterne Versorgung ausgefallen**		ja
	Ursache	Fehler in der FM 352	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit	
	Behebung	Tauschen der FM 352	
70	**Prozessalarm verloren**		ja
	Ursache	Ein Prozessalarmereignis wurde von der FM 352 erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht vom Anwenderprogramm / der CPU verarbeitet wurde	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• OB 40 in Anwenderprogramm einbinden • Bus-Anschluss der Baugruppe überprüfen • Prozessalarm deaktivieren • Passen Sie Ihre Hardware und Software Ihren Prozessanforderungen an (z. B. schnellere CPU, Anwenderprogramm optimieren)	
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden • Technische Daten der Baugruppe einhalten	

Nr.	Bedeutung		Diagnosealarm
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen FM 352 und dem Absolutwertgeber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • der letzte korrekte Istwert bleibt bis zum Ende der nächsten korrekten SSI-Übertragung unverändert	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und FM 352	
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden	

2.1.4 FM 452 einsetzen

2.1.4.1 Anschlussbelegung des Frontsteckers der FM 452

Frontstecker

Am 48-poligen Frontstecker schließen Sie an:

- den Geber
- die digitalen Ausgänge und Eingänge
- die Versorgungsspannungen der Geber und der Digitalausgänge

Anschlussbelegung des Frontsteckers (X1)

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 452.

Anschluss	Name	Initiator	Inkrementalgeber	Absolutwertgeber
1		–		
2		–		
3	1L+	Hilfsspannung DC 24 V		
4	A / DAT	–	Gebersignal A (5 V)	SSI-Daten
5	/A / /DAT	–	Gebersignal A invers (5 V)	SSI-Daten invers
6	B / CLI [1]	–	Gebersignal B (5 V)	SSI-Schiebetakt Eingang [1]
7	/B / /CLI [1]	–	Gebersignal B invers (5 V)	SSI-Schiebetakt Eingang invers [1]
8	N	–	Nullmarkensignal (5 V)	–
9	/N	–	Nullmarkensignal invers (5 V)	–
10	CLS [2]	–	–	SSI-Schiebetakt Ausgang
11	/CLS [2]	–	–	SSI-Schiebetakt Ausgang invers
12	A*	Gebersignal A (24 V)		–
13	B*	–	Gebersignal B (24 V)	–
14	N*	–	Nullmarkensignal (24 V)	–
15	Q0	Digitalausgang 0		
16	Q1	Digitalausgang 1		
17	Q2	Digitalausgang 2		
18	Q3	Digitalausgang 3		
19	Q4	Digitalausgang 4		
20	Q5	Digitalausgang 5		
21	Q6	Digitalausgang 6		
22	Q7	Digitalausgang 7		
23	DC 5.2 V	–	Geberversorgung (5,2 V)	
24	DC 24 V	Geberversorgung (24 V)		
25	M [3]	Masse Geber		

Anschluss	Name	Initiator	Inkrementalgeber	Absolutwertgeber
26	2L+	Hilfsspannung DC 24 V		
27	RE	–	P/M-schaltend [4]	–
28	Q8	Digitalausgang 8		
29	Q9	Digitalausgang 9		
30	Q10	Digitalausgang 10		
31	Q11	Digitalausgang 11		
32	Q12	Digitalausgang 12		
33	Q13	Digitalausgang 13		
34	Q14	Digitalausgang 14		
35	Q15	Digitalausgang 15		
36	3L+	Hilfsspannung DC 24 V		
37	I0	Freigabe Bremsnockenspur		
38	I1	Längenmessung / Kantenerfassung / Fliegendes Istwert setzen		
39	I2	Referenzpunktschalter		
40	I3	Freigabe Spursignal 3		
41	I4	Freigabe Spursignal 4		
42	I5	Freigabe Spursignal 5		
43	I6	Freigabe Spursignal 6		
44	I7	Freigabe Spursignal 7		
45	I8	Freigabe Spursignal 8		
46	I9	Freigabe Spursignal 9		
47	I10	Freigabe Spursignal 10		
48	M [3]	Masse für Hilfsspannungen		

[1] Im Mithörbetrieb

[2] Im Masterbetrieb

[3] Masseanschlüsse sind auf der Baugruppe verbunden

[4] **Hinweis:** Wenn Sie einen Inkrementalgeber eines Fremdherstellers in Gegentaktschaltung (P/M-schaltend) anschließen, müssen Sie Folgendes beachten:

- P-schaltend: Verbinden Sie RE (27) mit Masse (25).
- M-schaltend: Verbinden Sie RE (27) mit +24 V (24).

Weitere Informationen zu Aufbau und Verdrahtung finden Sie in der Betriebsanleitung (http://support.automation.siemens.com/WW/view/de/1123332) des Elektronischen Nockensteuerwerks FM 452.

Hilfsspannung für Geber und Digitalausgänge (1L+, 2L+, 3L+)

Die DC 24 V-Hilfsspannung der Geber und Digitalausgänge wird überwacht:

- auf Drahtbruch der 24 V-Zuleitung
- auf Spannungsausfall.

Am Frontstecker (X1) werden DC 24 V (Klemme 24) und DC 5,2 V (Klemme 23) für die unterschiedlichen Geberarten zur Verfügung gestellt.

Laststromversorgungen

Die DC-Laststromversorgung muss folgenden Anforderungen genügen:

Als Laststromversorgung darf nur vom Netz sicher getrennte Kleinspannung DC ≤ 60 V verwendet werden. Die sichere Trennung kann realisiert sein nach den Anforderungen u. a. in

- VDE 0100 Teil 410 / HD 384-4-41 / IEC 364-4-41
  (als Funktionskleinspannung mit sicherer Trennung) bzw.
- VDE 0805 / EN 60950 / IEC 950
  (als Sicherheitskleinspannung SELV) bzw.
- VDE 0106 Teil 101.

11 Digitaleingänge (I 0 bis I 10)

Über 11 Digitaleingänge können Sie prellfreie Schalter (24 V P-schaltend) oder berührungslose Sensoren (2- oder 3-Draht Näherungsschalter) anschließen.

Die Digitaleingänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgebunden zur Baugruppenmasse.

Der Zustand jedes Eingangs ist an der zugehörigen LED ablesbar.

16 Digitalausgänge (Q 0 bis Q 15)

Über 16 Digitalausgänge wird der jeweilige Zustand (ein / aus) der Spuren 0 bis 15 ausgegeben. Die Digitalausgänge sind potentialgebunden zur Baugruppenmasse.

Folgende Belastungen sind möglich:

- Arbeitsspannung 24 V
- Strombelastung 0,5 A / kurzschlussfest

Der Zustand jedes Ausgangs ist an der zugehörigen LED ablesbar.

2.1.4.2 Funktionen zum Nockensteuern

Einfluss von Funktionen auf das Schaltverhalten von Zeitnocken

Istwertveränderungen

Ein Zeitnocken kann durch folgende istwertverändernde Funktionen übersprungen werden:

- Istwert setzen
- Fliegendes Istwert setzen
- Nullpunktverschiebung
- Referenzpunkt nachtriggern

Schalten eines Zeitnockens

Wenn Sie durch eine der oben genannten Funktionen den Nockenanfang eines Zeitnockens überspringen, schaltet dieser Nocken ein, sofern die tatsächliche Bewegungsrichtung der Achse mit der für den Nocken parametrierten Wirkrichtung übereinstimmt. Die parametrierte Einschaltzeit läuft ab.

Hinweis

Im Stillstand ist die Bewegungsrichtung abhängig von den Istwertschwankungen.

Wenn Sie auch im Stillstand eine definierte Bewegungsrichtung berücksichtigen wollen, müssen Sie eine Hysterese parametrieren, die größer ist als die Schwankungen des Istwerts im Stillstand.

Ist die Achse nicht in Bewegung, so bleibt die zuletzt ermittelte Bewegungsrichtung erhalten.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Istwertverändernde Funktionen können bei einer Rundachse zu einem nicht gewünschten Einschalten von Zeitnocken führen.

Schalten Sie auf jeden Fall bei einer Rundachse die Zeitnocken "ungültig", wenn Sie durch die oben genannten Funktionen den Istwert beeinflussen wollen.

Istwert setzen / Fliegendes Istwert setzen / Istwert setzen rückgängig

Definition

Mit den Funktionen "Istwert setzen" und "Fliegendes Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Das Koordinatensystem verschiebt sich dadurch um den Wert: IST_{neu} - $IST_{aktuell}$

Dabei ist:

- IST_{neu} der Vorgabewert
- $IST_{aktuell}$ der Istwert zum Zeitpunkt der Ausführung

Neue Koordinaten ermitteln

Alle von Ihnen vorgegebenen Positionen im verschobenen Koordinatensystem berechnen Sie mit der folgenden Formel:

$$Koordinate_{neu} = Koordinate_{alt} + (IST_{neu} - IST_{aktuell})$$

Voraussetzungen

- Die Achse muss synchronisiert sein.
- Bei "Fliegendes Istwert setzen": Der Digitaleingang I1 muss verschaltet sein.

Ablauf der Funktion

1. Tragen Sie die Koordinate für den Istwert bzw. für den Fliegenden Istwert in den Kanal-DB ein.
   - Linearachse:
     Den angegebenen Istwert müssen Sie so wählen, dass die Software-Endschalter nach dem Aufruf der Funktion noch innerhalb des zulässigen Verfahrbereichs liegen.
     Der Betrag der Verschiebung, der sich aus **(IST_{neu} - $IST_{aktuell}$)** ergibt, muss kleiner oder gleich dem Betrag des zulässigen Verfahrbereichs sein (maximal 100 m bzw. 1000 m).
   - Rundachse:
     Für den angegebenen Istwert muss gelten:
     0 ≤ Istwert < Ende der Rundachse
2. Setzen Sie die entsprechenden Anstoßbits im Kanal-DB.
3. Rufen Sie die Anweisung CAM_CTRL bzw. CAM_CTRL_452 auf.
   "Istwert setzen" wird sofort ausgeführt.
   "Fliegendes Istwert setzen" wird mit der nächsten steigenden Flanke am Digitaleingang I1 ausgeführt. Das Bit FVAL_DONE wird gesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.4	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.5	FVAL_EN	BOOL	FALSE	1 = Fliegendes Istwert setzen
90.0	AVAL	DINT	L#0	Koordinate für Istwert
94.0	FVAL	DINT	L#0	Koordinate für Fliegenden Istwert
25.5	FVAL_DONE	BOOL	FALSE	1 = Fliegendes Istwert setzen ausgeführt

Auswirkungen der Funktion

Am Beispiel "Istwert setzen" auf 400 mm (bei Position 200 mm) erkennen Sie, wie diese Funktion das Koordinatensystem verschiebt. Es ergeben sich folgende Auswirkungen:

- Die Lage des Arbeitsbereichs wird physikalisch **nicht** verschoben.
- Den einzelnen Punkten (wie z. B. den Software-Endschaltern) werden neue Koordinatenwerte zugeordnet.
- Die Nocken behalten ihre Koordinatenwerte und liegen somit an einer anderen physikalischen Position.
- Bei synchronisierter Achse und freigegebener Nockenbearbeitung kann der Lageistwert durch diese Funktion Nockenflanken oder komplette Nocken überspringen.
- Der Zustandswechsel des Nockens, der normalerweise einen Alarm auslösen würde, könnte verloren gehen.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 551).

Die folgende Tabelle zeigt die Verschiebung des Koordinatensystems durch "Istwert setzen" / "Fliegendes Istwert setzen".

Istwert setzen	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
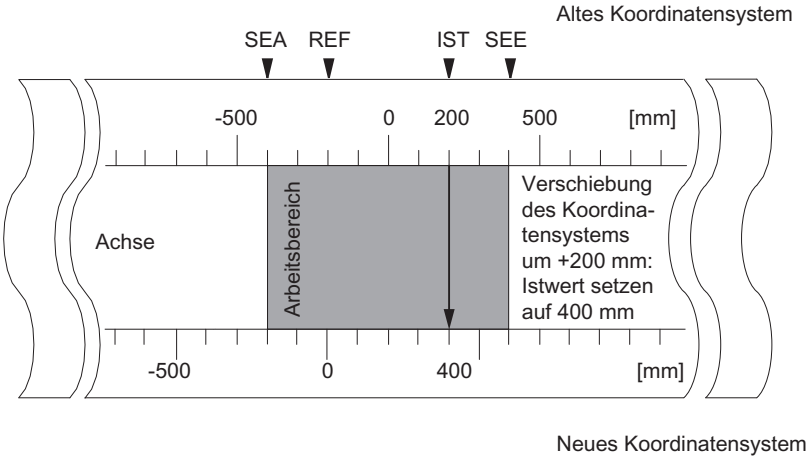	-400	-200	200	400
	-200	0	400	600

Rücknahme der Funktion

Mit der Funktion "Istwert setzen rückgängig" setzen Sie die Koordinatenverschiebung, die durch "Istwert setzen" bzw. "Fliegendes Istwert setzen" hergestellt wurde, wieder zurück.

Ein einmal angestoßenes "Fliegendes Istwert setzen" kann vor der Ausführung durch eine steigende Flanke am Eingang I1 nicht mehr gelöscht werden. Es kann aber durch ein neues "Fliegendes Istwert setzen" überschrieben werden.

Bei einem Baugruppenanlauf werden diese Funktionen zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.2	AVALREM_EN	BOOL	FALSE	1: Istwert setzen zurücknehmen

Mögliche Fehlerquellen

"Fliegendes Istwert setzen" und "Referenzpunkt nachtriggern" dürfen nicht gleichzeitig ausgeführt werden.

Bei der Funktion "Fliegendes Istwert setzen" kann ein Betriebsfehler gemeldet werden, wenn durch diese Funktion ein Software-Endschalter bei der steigenden Flanke am I1 den zulässigen Verfahrbereich verlassen würde. Dieser Betriebsfehler wird mit einem Diagnosealarm gemeldet und im Diagnosepuffer eingetragen.

Nullpunktverschiebung ausführen

Definition

Mit der Funktion "Nullpunktverschiebung" verschieben Sie den Nullpunkt im Koordinatensystem um den eingegebenen Wert. Das Vorzeichen bestimmt die Richtung der Verschiebung.

Neue Koordinate ermitteln

Alle Werte im verschobenen Koordinatensystem berechnen Sie mit der folgenden Formel:

$\text{Koordinate}_{neu} = \text{Koordinate}_{alt} - (\text{NPV}_{neu} - \text{NPV}_{alt})$

NPV_{alt} kennzeichnet dabei eine eventuell bestehende alte Nullpunktverschiebung. Wenn vor dem Aufruf keine Nullpunktverschiebung aktiv war, setzen Sie für NPV_{alt} den Wert 0 ein.

Sie können mit dieser Formel ermitteln, welche Koordinatenwerte z. B. die Software-Endschalter annehmen.

Ablauf der Funktion

1. Tragen Sie den Wert für die Nullpunktverschiebung in den Kanal-DB ein.
   - Linearachse:
     Die Nullpunktverschiebung muss so gewählt werden, dass die Software-Endschalter nach dem Aufruf der Funktion noch innerhalb des zulässigen Verfahrbereichs liegen.
   - Rundachse:
     Für die Nullpunktverschiebung muss gelten:
     Betrag der Nullpunktverschiebung ≤ Ende der Rundachse.
2. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.6	ZOFF_EN	BOOL	FALSE	1 = Nullpunktverschiebung setzen
86.0	ZOFF	DINT	L#0	Nullpunktverschiebung

Auswirkungen bei einer Linearachse

Am Beispiel einer Nullpunktverschiebung von -200 mm erkennen Sie, dass diese Funktion das Koordinatensystem in positive Richtung verschiebt. Es ergeben sich folgende Auswirkungen:

- Der Arbeitsbereich wird physikalisch **nicht** verschoben.
- Den einzelnen Punkten (wie z. B. den Software-Endschaltern) werden neue Koordinatenwerte zugeordnet.
- Die Nocken behalten ihre Koordinatenwerte und liegen somit an einer anderen physikalischen Position.

- Bei synchronisierter Achse und freigegebener Nockenbearbeitung kann der Lageistwert durch diese Funktion Nockenflanken oder komplette Nocken überspringen.
- Der Zustandswechsel des Nockens, der normalerweise einen Alarm auslösen würde, kann verloren gehen.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 551).

Die folgende Tabelle zeigt die Verschiebung des Koordinatensystems durch Nullpunktverschiebung.

Nullpunktverschiebung	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
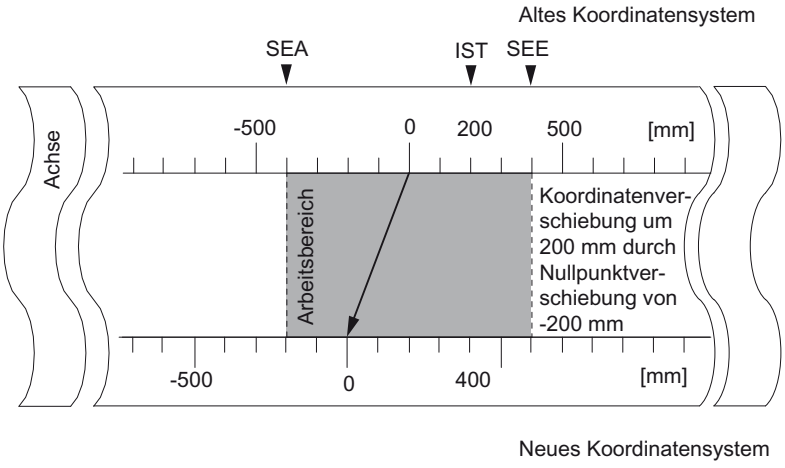	-400	-200	200	400
	-200	0	400	600

Auswirkungen bei einer Rundachse

Am Beispiel einer Nullpunktverschiebung um -45° erkennen Sie, wie diese Funktion das Koordinatensystem dreht.

Die folgende Tabelle zeigt die Drehung des Koordinatensystems durch eine Nullpunktverschiebung.

Werkzeug bei IST = 340°	Werkzeug bei IST = 25°
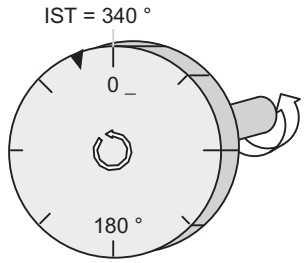	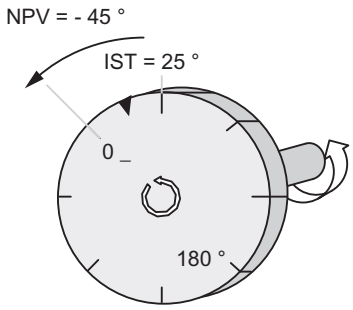
Der Nullpunkt dreht sich um -45 Grad. Alle Punkte der Achse erhalten neue Koordinatenwerte.	

Unter Berücksichtigung einer **NPV_{alt} = 0** ergibt sich ein neuer Wert von 385°

Da am Ende der Rundachse bei einer positiven Drehrichtung der Istwert wieder bei 0 beginnt, errechnet sich der tatsächliche Istwert von 25° aus:

$Koordinate_{neu}$ = $Koordinate_{alt}$ - (NPV_{neu} - NPV_{alt}) - Ende der Rundachse

Der Wert **Ende der Rundachse** muss nur dann subtrahiert werden, wenn

$Koordinate_{alt}$ - **(NPV_{neu} - NPV_{alt})** größer als das Ende der Rundachse wird.

Wegfall der Synchronisation

Wenn durch einen Fehler die Synchronisation verloren geht oder durch "Referenzpunkt nachtriggern" neu gesetzt wird, **bleibt** eine Nullpunktverschiebung aktiv.

Funktion zurücknehmen

Mit der Vorgabe einer Nullpunktverschiebung von 0 setzen Sie eine bestehende Nullpunktverschiebung wieder zurück.

Bezugspunkt setzen

Definition

Mit der Funktion "Bezugspunkt setzen" synchronisieren Sie die Achse. Die Funktion verschiebt den Arbeitsbereich. Alle Verschiebungen, die durch eine Nullpunktverschiebung oder durch "Istwert setzen" erzeugt wurden, bleiben erhalten.

Die Funktion projiziert den Arbeitsbereich auf die Achse. Durch unterschiedliche Wertvorgaben kann deshalb der Arbeitsbereich an unterschiedlichen Positionen im physikalischen Bereich der Achse liegen.

Voraussetzung

Die Nockenbearbeitung muss abgeschaltet sein.

Ablauf der Funktion

1. Tragen Sie den Wert für die Bezugspunktkoordinate in den Kanal-DB ein.
   - Linearachse:
     Die Bezugspunktkoordinate darf nicht außerhalb der Software-Endschalter liegen. Das gilt auch für die Bezugspunktkoordinate in einem verschobenen Koordinatensystem.
   - Rundachse:
     Für die Bezugspunktkoordinate muss gelten:
     0 ≤ Bezugspunktkoordinate < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.3	REFPT_EN	BOOL	FALSE	1 = Bezugspunktkoordinate setzen
98.0	REFPT	DINT	L#0	Bezugspunktkoordinate
25.0	SYNC	BOOL	FALSE	1 = Achse synchronisiert

Auswirkungen der Funktion

Am Beispiel "Bezugspunkt setzen" auf 300 mm erkennen Sie, wie diese Funktion den Arbeitsbereich auf der Achse verschiebt.

Daraus ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Bezugspunktkoordinate gesetzt.
- Der Arbeitsbereich wird physikalisch auf der Achse verschoben.
- Die einzelnen Punkte behalten ihren ursprünglichen Koordinaten, liegen aber auf neuen physikalischen Positionen.
- Das Bit SYNC in den Rückmeldesignalen ist gesetzt.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Bezugspunkt setzen".

Bezugspunkt setzen	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
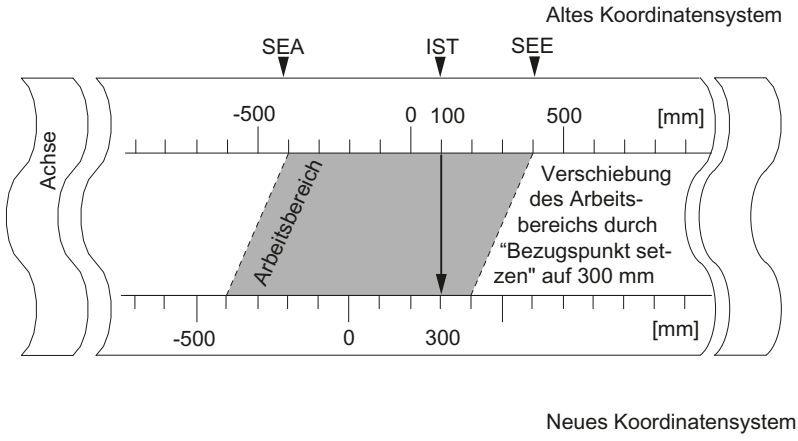	-400	-200	100	400
	-400	-200	300	400

Besonderheiten Absolutwertgeber

Diese Funktion ist für eine Absolutwertgeberjustage (Seite 582) notwendig.

Nockenflanken ändern

Definition

Mit der Funktion "Nockenflanken ändern" können Sie im laufenden Betrieb den Nockenanfang und - bei einem Wegnocken - auch das Nockenende eines einzelnen Nockens ändern.

Voraussetzung

Der Nocken, den Sie ändern wollen, muss gültig sein.

Ablauf der Funktion

1. Tragen Sie die Nockennummer in den Kanal-DB ein.
2. Bei einem Wegnocken:
   Tragen Sie Nockenanfang und Nockenende in den Kanal-DB ein.
   Bei einem Zeitnocken:
   Tragen Sie den Wert für den Nockenanfang in den Kanal-DB ein.
3. Setzen Sie das entsprechende Anstoßbit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.7	CH01CAM_EN	BOOL	FALSE	1 = Funktion für Nockenflanken (1 Nocken) schreiben
102.0	CAM_NO	INT	0	Nockennummer
104.0	CAM_START	DINT	L#0	Nockenanfang
108.0	CAM_END	DINT	L#0	Nockenende

Auswirkungen der Funktion

Die FM x52 verschiebt zunächst die Einschaltflanke und dann die Ausschaltflanke des Nockens. Dieser Ablauf ist unabhängig von der Richtung, in die der Nocken geschoben wird.

Sonderfall:

Durch den oben erklärten Ablauf kann kurzzeitig ein inverser Nocken (Seite 453) entstehen, wenn der neue Nockenanfang größer ist als das alte Nockenende.

Das folgende Bild zeigt die Veränderung der Nockenflanken in Einzelschritten.

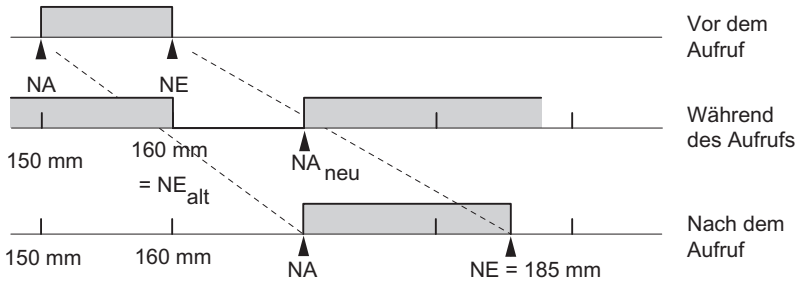

Hinweis

Wenn für diesen Nocken ein Prozessalarm freigeschaltet wurde, kann die FM x52 bei Erkennen des inversen Nockens je nach Parametrierung ein oder zwei Prozessalarme auslösen.

Durch das Ändern der Ein- und/oder Ausschaltflanke kann es zum Überspringen einer Nockenflanke oder des ganzen Nockens kommen.

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 551).

Zustandswechsel des Nockens, die normalerweise einen Prozessalarm auslösen würden, können verloren gehen.

Geänderte Werte auslesen

Die geänderten Werte können Sie mit einem der Aufträge CAM1RD_EN bis CAM8RD_EN auslesen.

Rücknahme der Funktion

Die geänderten Werte gehen beim Neustart der Baugruppe verloren.

"Schnelles Nockenändern" ausführen

Definition

Mit der Funktion "Schnelles Nockenändern" können Sie im laufenden Betrieb Daten von bis zu 16 beliebigen Nocken gleichzeitig ändern.

Voraussetzung

Die Nocken, die Sie ändern wollen, müssen gültig sein.

Ablauf der Funktion

1. Tragen Sie die Anzahl der zu ändernden Nocken in den Kanal-DB ein.
2. Tragen Sie die Nummer des ersten zu ändernden Nockens in den Kanal-DB ein.
3. Setzen Sie die Anstoßbits für die gewünschte Änderungen.
4. Tragen Sie die neuen Werte in den Kanal-DB ein.
5. Wiederholen Sie die Schritte 2 bis 4 für jeden zu ändernden Nocken.
6. Setzen Sie das entsprechende Anstoßbit im Kanal-DB

Verwendete Daten im Kanal-DB

Adresse absolut	Name	Typ	Anfangswert	Kommentar
37.0	CH16CAM_EN	BOOL	FALSE	1 = Funktion für schnelles Nockenändern (16 Nocken) schreiben
176.0	C_QTY	BYTE	B#16#0	Anzahl der zu ändernden Nocken
177.0	DIS_CHECK	BOOL	FALSE	1 = Datenprüfung ausschalten

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.0	CAM_NO	BYTE	B#16#0	Nummer des zu ändernden Nockens
+1.0	C_EFFDIR	BOOL	FALSE	1 = Änderung der Nockenwirkrichtung
+1.1	C_CBEGIN	BOOL	FALSE	1 = Änderung des Nockenanfangs auf den Wert CBEGIN
+1.2	C_CEND	BOOL	FALSE	1 = Änderung des Nockenendes / der Einschaltzeit auf den Wert CEND
+1.3	C_LTIME	BOOL	FALSE	1 = Änderung der Vorhaltezeit auf den Wert LTIME
+1.4	CAM_OFF	BOOL	FALSE	1 = Ausschalten des Nockens während der Nockenänderung

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+1.5	EFFDIR_P	BOOL	FALSE	1 = Wirkrichtung positiv (plus)
+1.6	EFFDIR_M	BOOL	FALSE	1 = Wirkrichtung negativ (minus)
+2.0	CBEGIN	DINT	L#0	Neuer Nockenanfang
+6.0	CEND	DINT	L#0	Neues Nockenende / neue Einschaltzeit
+10.0	LTIME	INT	L#0	Neue Vorhaltezeit

Nocken abschalten beim Ändern

Ein konsistentes Ändern des Nockenanfangs und des Nockenendes ist nur dann möglich, wenn Sie den Nocken zum Zeitpunkt des Änderns abgeschaltet haben (CAM_OFF).

Datenprüfung durch die Baugruppe

Mit dem Parameter DIS_CHECK (Kanal-DB) stellen Sie ein, ob die Datenprüfung der übertragenen Daten durch die FM x52 ausgeschaltet werden soll. Wenn Sie die Datenprüfung ausschalten, müssen Sie selbst sicherstellen, dass nur zulässige Werte übertragen werden. Wenn Sie unzulässige Werte ohne Prüfung eingeben, kann dies zu einem unvorhergesehenen Verhalten der Baugruppe führen.

- FALSE: Die Baugruppe überprüft alle Daten, die übertragen werden
- TRUE: Die Datenprüfung bzgl. der Nockenparameter ist abgeschaltet. Dadurch werden die zu ändernden Daten schneller auf der FM x52 eingearbeitet.
  Unabhängig von dieser Funktion wird immer geprüft, ob
  - die Achse parametriert ist,
  - die Anzahl der zu ändernden Nocken (C_QTY) zulässig ist,
  - der Nocken (Nockennummer), der geändert werden soll, gültig ist.

Erst wenn alle Daten geprüft und fehlerfrei sind, werden sie auf der Baugruppe aktiviert.

Im Fehlerfall werden alle Daten verworfen.

Auswirkungen der Funktion

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 551).

Geänderte Werte auslesen

Die geänderten Werte können mit einem der Aufträge CAM1RD_EN bis CAM8RD_EN ausgelesen werden.

Rücknahme der Funktion

Die geänderten Werte gehen beim Neustart der Baugruppe verloren.

"Längenmessung" und "Kantenerfassung" ausführen

Definition

Mit den Funktionen "Längenmessung" und "Kantenerfassung" können Sie die Länge eines Werkstücks ermitteln.

Die Längenmessung und Kantenerfassung sind und bleiben so lange aktiv, bis Sie sie wieder ausschalten oder die jeweils andere Messmethode wählen. Falls Sie beide Messmethoden gleichzeitig wählen, wird die Anweisung CAM_CTRL bzw. CAM_CTRL_452 die Längenmessung einschalten.

Voraussetzung

Am Eingang I1 muss ein **prellfreier** Schalter angeschlossen sein.

Ablauf der Funktionen

Je nach der Art der Messung aktualisiert die FM 452 die Daten auf der Baugruppe zu einem unterschiedlichen Zeitpunkt. Jede Aktualisierung meldet die FM 452 in einem Parameter an der Rückmeldeschnittstelle.

Längenmessung

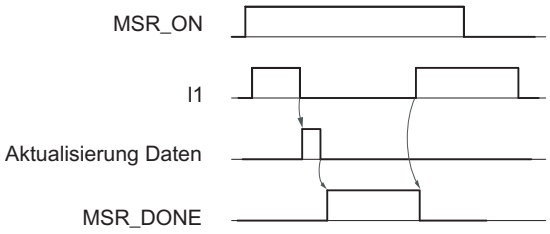

1. Setzen Sie den Funktionsschalter für "Längenmessung".
2. Die steigende Flanke des Eingangs I1 startet die Längenmessung.
3. Die fallende Flanke des Eingangs I1 beendet eine laufende Messung. Die FM 452 aktualisiert die Daten Startwert, Endwert und Länge.
4. Mit dem gesetzten Parameter MSR_DONE meldet die FM 452 das Aktualisieren der Daten. Der Parameter zeigt an, dass die Messung beendet ist. Die Messergebnisse können ausgelesen werden.
5. Der Start einer weiteren Messung mit der steigenden Flanke von I1 setzt den Parameter MSR_DONE zurück.

Wenn während einer Längenmessung die Funktion ausgeschaltet wird, dann aktualisiert die FM 452 die Daten nicht. Der Parameter MSR_DONE bleibt zurückgesetzt.

Kantenerfassung

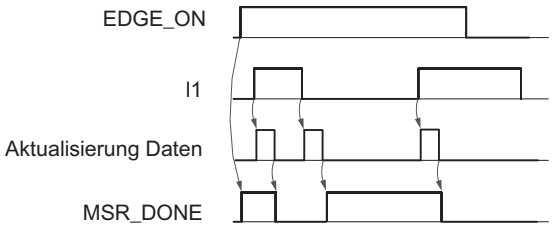

1. Tragen Sie ggf. einen Wert für den minimalen Kantenabstand im Parameter-DB ein. Schreiben und aktivieren Sie die Maschinendaten.
2. Setzen Sie den Funktionsschalter für "Kantenerfassung". Der Parameter MSR_DONE wird gesetzt.
3. Die steigende Flanke des Eingangs I1 startet die Kantenerfassung. Die Messergebnisse werden aktualisiert und können ausgelesen werden, der Startwert der Messung wird eingetragen; Endwert und Länge werden -1.
4. Nach der Aktualisierung meldet die FM 452 die Veränderung durch das Rücksetzen des Parameters MSR_DONE.
5. Die fallende Flanke des Eingangs I1 beendet eine laufende Messung. Die FM 452 aktualisiert die Daten für Endwert der Messung und Länge.
6. Nach der Aktualisierung meldet die FM 452 die Veränderung durch das Setzen des Parameters MSR_DONE. Die Messergebnisse können ausgelesen werden.
7. Der Start einer weiteren Messung mit der steigenden Flanke von I1 setzt den Parameter MSR_DONE zurück.

Wenn während einer Kantenerfassung die Funktion ausgeschaltet wird, dann aktualisiert die FM 452 die Daten nicht. Der Parameter MSR_DONE bleibt zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
25.1	MSR_DONE	BOOL	FALSE	1 = Längenmessung ist abgeschlossen
34.0	EDGE_ON	BOOL	FALSE	1 = Kantenerfassung ein
34.2	MSR_ON	BOOL	FALSE	1 = Längenmessung ein
38.2	MSRRD_EN	BOOL	FALSE	1 = Messwerte lesen
112.0	BEG_VAL	DINT	L#0	Startwert
116.0	END_VAL	DINT	L#0	Endwert
120.0	LEN_VAL	DINT	L#0	Länge

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	**Minimaler Kantenabstand bei Kantenerfassung** Zulässiger Wertebereich: • 0 ... 1 000 000 000 µm

Mit dem minimalen Kantenabstand definieren Sie einen Bereich nach dem Erkennen des Messbeginns bei einer Kantenerfassung. Wenn das Ende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen.

Der Messbeginn wird erst nach Verfahren des Wegstücks "minimaler Kantenabstand" gemeldet.

Randbedingungen für eine Längenmessung

- Der Abstand zwischen Ausschaltflanke und Einschaltflanke am Eingang I1 muss so groß sein, dass Ihr Programm in der CPU das Ergebnis der Messung korrekt auswerten kann, bevor eine neue Messung beginnt.
- Der minimale Abstand zwischen der steigenden und der fallenden Flanke am Eingang I1 wie auch zwischen der fallenden und der nächsten steigenden Flanke am Eingang I1 muss größer als 2 ms sein.

Prozessalarme

Bei der FM 452 können Beginn und Ende einer Messung über Prozessalarme gemeldet werden.

Fehlerhafte Messung

Im Fall einer fehlerhaften Längenmessung/Kantenerfassung liefert die FM 452 den Wert -1 für die Länge zurück.

Eine "Längenmessung" oder "Kantenerfassung" darf maximal 126 Nullübergänge in eine Richtung durchlaufen. Ein Nullübergang ist der Rundachsenübergang vom Wert Ende der Rundachse nach 0 und umgekehrt. Sobald mehr als 126 Nullübergänge in eine Richtung von der FM 452 erkannt werden, wird eine fehlerhafte "Längenmessung" bzw. "Kantenerfassung" gemeldet, auch wenn anschließend wieder Nullübergänge in der anderen Richtung erkannt wurden.

Eine Messung ist auch fehlerhaft, wenn

- bei einer Rundachse die gemessene Länge größer als 2^{31} ist,
- gleichzeitig die Einschalt- und die Ausschaltflanke von der FM 452 erkannt wird (z. B. durch Schalterprellen).

Dieses Ereignis kann als Prozessalarm gemeldet werden.

Verschiebung des Koordinatensystems während einer Längenmessung

Koordinatenverschiebungen beeinflussen die gemessene Länge unter folgenden Voraussetzungen:

- Sie setzen einen Inkrementalgeber oder einen Initiator ein oder Sie betreiben die FM 452 im Simulationsbetrieb.
- Sie führen "Bezugspunkt setzen" oder "Referenzpunkt nachtriggern" während einer laufenden Längenmessung aus.

Beispiel

Sie können die oben genannte Beeinflussung der gemessenen Länge wie folgt ausnutzen:

Sie haben ein System, bei dem während einer Längenmessung immer ein Schlupf auftritt.

Durch "Referenzpunkt nachtriggern" können Sie diesen Schlupf korrigieren, so dass dann korrekte Längenmesswerte ausgegeben werden.

Referenzpunkt nachtriggern

Definition

Mit der Funktion "Referenzpunkt nachtriggern" können Sie die Achse aufgrund eines wiederkehrenden externen Ereignisses synchronisieren.

Die Funktion bleibt solange aktiv, bis Sie sie wieder ausschalten.

Voraussetzung

- Sie verwenden einen Inkrementalgeber oder Initiator.
- Das externe Ereignis kann eine Nullmarke eines Inkrementalgebers oder ein Referenzpunktschalter am Eingang I2 sein.

Ablauf der Funktion

1. Tragen Sie den Wert der Referenzpunktkoordinate in den Parameter-DB ein.
2. Tragen Sie die Art des "Referenzpunkt nachtriggerns" in den Parameter-DB ein.
   Hierbei haben Sie folgende Möglichkeiten:
   – Nur die Nullmarke des Gebers wird ausgewertet (RETR_TYPE = 7).
   – Nur der Referenzpunktschalter wird ausgewertet (RETR_TYPE = 6).
   – Die Nullmarke wird ausgewertet
     in positiver Richtung: Die erste steigende Flanke der Nullmarke nach Verlassen des Referenzpunktschalters in positiver Richtung wird ausgewertet (RETR_TYPE = 0).
     in negativer Richtung: Die erste fallende Flanke der Nullmarke nach Verlassen des Referenzpunktschalters in negativer Richtung wird ausgewertet (RETR_TYPE = 1).
3. Schreiben und aktivieren Sie die Maschinendaten.
4. Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.3	REFTR_ON	BOOL	FALSE	1 = Referenzpunkt nachtriggern
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
52.0	RETR_TYPE	DINT	L#0	Art des "Referenzpunkt nachtriggern"

Auswirkungen der Funktion

- Die FM x52 wertet die Nullmarke und den Referenzpunktschalter in Abhängigkeit der Bewegungsrichtung der Achse aus.
  - Bei positiver Bewegungsrichtung werden die steigenden Flanken ausgewertet.
  - Bei negativer Bewegungsrichtung werden die fallenden Flanken ausgewertet.
- Die Istposition wird auf den Wert der Referenzpunktkoordinate gesetzt.
- Der Arbeitsbereich wird physikalisch auf der Achse verschoben.
- Die einzelnen Punkte behalten ihren ursprünglichen Wert, liegen aber auf neuen physikalischen Positionen.
- Zustandswechsel des Nockens, die normalerweise einen Alarm auslösen würden, können verloren gehen.
- Das Bit SYNC in den Rückmeldesignalen wird gesetzt.

Hinweis

Beachten Sie das Schaltverhalten für Zeitnocken (Seite 551).

Beispiel

Für das Beispiel gilt:

- Von Referenzpunktschalter und Nullmarke werden die steigenden Flanken ausgewertet (Positive Bewegungsrichtung der Achse).
- Die Referenzpunktkoordinate hat den Wert 300 mm.
- Zum Zeitpunkt der Ausführung ist keine Nullpunktverschiebung aktiv.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Referenzpunkt nachtriggern".

Referenzpunkt nachtriggern	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
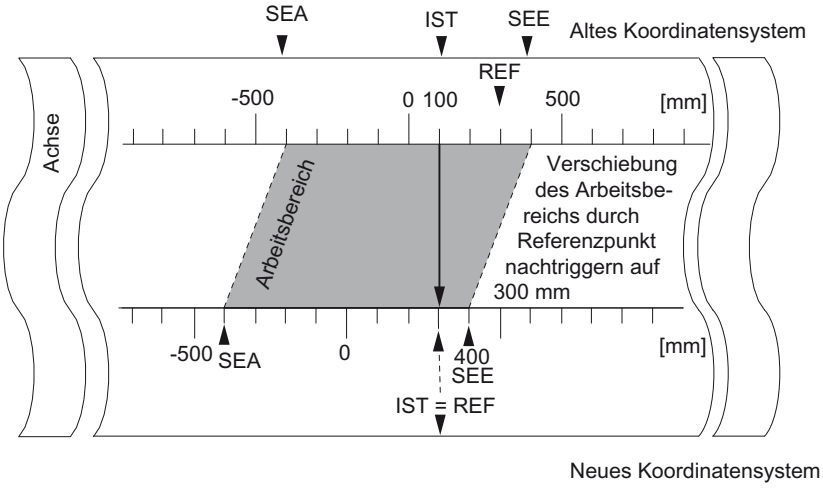	-400	300	100	400
	-400	300	300	400

Berücksichtigung einer Nullpunktverschiebung

Wenn eine Nullpunktverschiebung aktiv ist, wird diese bei der Funktion "Referenzpunkt nachtriggern" mit berücksichtigt. Das bedeutet, dass die Referenzpunktkoordinate, die gesetzt wird, sich nach folgender Formel berechnet:

Ref = Ref_{MD} - Nullpunktverschiebung

Ref_{MD} ist der Wert, der in den Maschinendaten als Referenzpunktkoordinate abgelegt ist.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Referenzpunkt nachtriggern" mit Nullpunktverschiebung.

Referenzpunkt nachtriggern	SEA [mm]	REF [mm]	IST [mm]	SEE [mm]
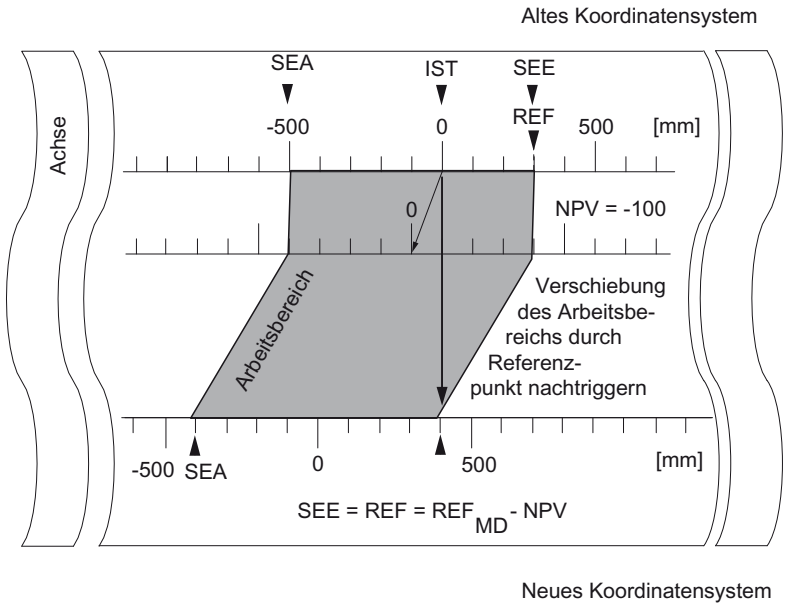	-500	300	0	300
	-400	400	100	400
	-400	400	400	400

Software-Endschalter abschalten

Definition

Die Funktion "Software-Endschalter abschalten" schaltet die Überwachung der Software-Endschalter einer Linearachse aus.

Die Funktion bleibt solange aktiv, bis Sie sie wieder ausschalten. Dann werden die ursprünglich parametrierten Software-Endschalter wieder aktiv.

Ablauf der Funktion

Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.4	SSW_OFF	BOOL	FALSE	1 = Software-Endschalter abgeschaltet

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L#-1000000000	Software-Endschalter Anfang
68.0	SSW_END	DINT	L#1000000000	Software-Endschalter Ende

Auswirkungen der Funktion

- Simulation
  - Wird im Simulationsbetrieb ein Software-Endschalter überfahren, dann wird der Simulationsbetrieb angehalten.
  - Schalten Sie jetzt die Überwachung der Software-Endschalter aus, wird der Simulationsbetrieb fortgesetzt. Die Achse bewegt sich in die vorgegebene Richtung.
- Nullpunktverschiebung bei ausgeschalteter Überwachung
  Wird eine Nullpunktverschiebung vorgegeben, bei der die Software-Endschalter noch innerhalb der Verfahrbereichsgrenzen liegen, kann trotzdem der Istwert außerhalb des zulässigen Zahlenbereichs sein.
- Nocken, die außerhalb der parametrierten Software-Endschalter liegen, können geschaltet werden.

> ⚠ **VORSICHT**
>
> Es kann zu Sachschäden kommen.
>
> Wen Sie mit den Software-Endschaltern den Verfahrbereich einschränken und damit Ihre Anlage schützen, kann es beim Abschalten der Software-Endschalter zu Sachschäden kommen.
>
> Achten Sie deshalb bei der Planung der Anlage darauf, dass der Antrieb im gesamten physikalischen Verfahrbereich fahren kann.

"Simulation" ausführen

Definition

Die Funktion "Simulation" bietet die Möglichkeit, das Nockensteuerwerk ohne angeschlossene Geber zu aktivieren.

Ablauf der Funktion

1. Tragen Sie die Simulationsgeschwindigkeit in den Parameter-DB ein.
2. Schreiben und aktivieren Sie die Maschinendaten.
3. Wählen Sie im Kanal-DB als Simulationsrichtung entweder die positive oder die negative Richtung aus.
4. Setzen Sie den Funktionsschalter im Kanal-DB.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.2	DIR_M	BOOL	FALSE	1 = Simulation in negativer Richtung
15.3	DIR_P	BOOL	FALSE	1 = Simulation in positiver Richtung
34.1	SIM_ON	BOOL	FALSE	1 = Simulation ein

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	SIM_SPD	DINT	L#0	Simulationsgeschwindigkeit

Auswirkungen beim Einschalten der Simulation

- Die Gebersignale werden nicht mehr ausgewertet.
- Alle Überwachungen bezüglich des Gebereingangs werden abgeschaltet.
- Eventuell gemeldete Fehler bezüglich des Gebers werden zurückgesetzt.
- Die FM x52 simuliert die Bewegung einer Achse mit der konstanten Simulationsgeschwindigkeit.
- Die Nockenbearbeitung wird mit dem Einschalten der Simulation abgeschaltet. Sie können sie dann jedoch wieder einschalten. Die Synchronisation bleibt dabei erhalten.
- Der Lageistwert verändert sich ab dem aktuellen Istwert in Abhängigkeit von der Simulationsgeschwindigkeit und der Simulationsrichtung.

Auswirkungen beim Ausschalten der Simulation

- Die Nockenbearbeitung wird abgeschaltet.
- Bei einem Inkrementalgeber oder Initiator wird die Synchronisation gelöscht. Als Istwert wird dann der Wert der Referenzpunktkoordinate eingestellt.
- Bei einem Absolutgeber wird der Lageistwert gemeldet, der dem aktuellen Geberstand entspricht. Die Gebersignale werden wieder so ausgewertet, wie sie in den Maschinendaten parametriert sind.

Grenzwerte

Die maximale und minimale Grenze der Simulationsgeschwindigkeit ist abhängig von der Auflösung (Seite 587).

Geschwindigkeit

Zwischen der vorgegebenen und der tatsächlichen Geschwindigkeit (Seite 587) , mit der die Baugruppe arbeitet, kann es Unterschiede geben.

"Zählwerte der Zählnockenspuren" auslesen

Definition

Mit der Funktion "Zählwerte der Zählnockenspuren" lesen Sie die aktuellen Zählwerte aus.

Ablauf der Funktion

1. Legen Sie die Zählnockenspuren und die oberen Zählwerte in den Maschinendaten fest.
2. Schreiben und aktivieren Sie die Maschinendaten.
3. Geben Sie die Zählfunktion frei.
4. Der Zählwert wird auf den oberen Zählwert gesetzt.
5. Bei jeder steigenden Flanke des Spurergebnisses wird der Zählwert um 1 verringert.
6. Setzen Sie das Anstoßbit im Kanal-DB zum Lesen der Zählwerte.
7. Die Zählwerte für beide Spuren werden in den Kanal-DB geschrieben. Für eine Spur, die nicht als Zählnockenspur parametriert wurde, wird 0 ausgegeben.
8. Erreicht der Zählwert den Wert 0, wird das Spurkennbit der Zählnockenspur auf 1 gesetzt.
9. Mit der nächsten fallenden Flanke des Spurergebnisses wird das Spurkennbit wieder 0 und der Zähler auf den oberen Zählwert gesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.5	CNTC0_EN	BOOL	FALSE	1 = Zählfunktion der Zählnockenspur 0 freigeben
15.6	CNTC1_EN	BOOL	FALSE	1 = Zählfunktion der Zählnockenspur 1 freigeben
38.3	CNTTRC_EN	BOOL	FALSE	1 = Zählwerte der Zählnockenspuren lesen
124.0	CNT_TRC0	INT	0	Aktueller Zählwert für Zählnockenspur 0
126.0	CNT_TRC1	INT	0	Aktueller Zählwert für Zählnockenspur 1

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
99.0	SPEC_TRC0	BOOL	FALSE	1 = Spur 0 ist Zählnockenspur
99.1	SPEC_TRC1	BOOL	FALSE	1 = Spur 1 ist Zählnockenspur
100.0	CNT_LIM0	DINT	L#2	Oberer Zählwert für Zählnockenspur 0
104.0	CNT_LIM1	DINT	L#2	Oberer Zählwert für Zählnockenspur 1

"Positions- und Spurdaten" auslesen

Definition

Mit der Funktion "Positions- und Spurdaten" lesen Sie die zu diesem Zeitpunkt aktuelle Position, die Geschwindigkeit und die Spurkennbits. Die Spurkennbits werden erfasst, bevor sie mit Maschinen- und Kanaldaten verknüpft werden.

Der in der FM x52 implementierte Rechenalgorithmus verrechnet Geschwindigkeitsänderungen, die größer sind als 1 Impuls pro 4 ms. Die angezeigte Geschwindigkeit ist also mit der entsprechenden Ungenauigkeit behaftet und ist insbesondere für Regelungszwecke nicht geeignet. Die interne Geschwindigkeit, die zum Dynamisieren der Nocken verwendet wird, ist genauer.

Ablauf der Funktion

1. Setzen Sie das Anstoßbit im Kanal-DB.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
38.4	ACTPOS_EN	BOOL	FALSE	1 = Positions- und Spurdaten lesen
128.0	ACTPOS	DINT	L#0	Aktuelle Position
132.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
136.0	TRACK_ID	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31

"Geberdaten" auslesen

Definition

Mit der Funktion "Geberdaten" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.

Voraussetzungen

Sie können den Wert für die Absolutwertgeberjustage (Seite 582) auslesen, nachdem Sie die Funktion "Bezugspunkt setzen (Seite 557)" ausgeführt haben.

Ablauf der Funktion

1. Setzen Sie das Anstoßbit im Kanal-DB.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
38.5	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
140.0	ENCVAL	DINT	L#0	Geberwert / Zählerstand (interne Darstellung)
144.0	ZEROVAL	DINT	L#0	Zählerstand bei der letzten Nullmarke (interne Darstellung)
148.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage

"Nocken- und Spurdaten" auslesen

Definition

Mit der Funktion "Nocken- und Spurdaten" lesen Sie die zu diesem Zeitpunkt aktuellen Nocken- und Spurkennbits sowie die Position. Die Spurkennbits werden erfasst, bevor sie mit Maschinen- und Kanaldaten verknüpft werden.

Ablauf der Funktion

1. Tragen Sie als Typkennung eine 1 in den Parameter FM_TYPE des Kanal-DB ein. Damit können Sie 24 Byte Nocken- und Spurdaten lesen.
   Wenn Sie für die Typkennung eine 0 eintragen, werden nur die Nockenkennbits (16 Byte) gelesen.
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 bis V4.0 1 = FM 352 ab V5.0 und FM 452
38.6	CAMOUT_EN	BOOL	FALSE	1 = Nocken- und Spurdaten lesen
152.0	CAM_00_31	DWORD	DW#16#0	Nockenkennbits für Nocken 0 bis 31
156.0	CAM_32_63	DWORD	DW#16#0	Nockenkennbits für Nocken 32 bis 63
160.0	CAM_64_95	DWORD	DW#16#0	Nockenkennbits für Nocken 64 bis 95
164.0	CAM_96_127	DWORD	DW#16#0	Nockenkennbits für Nocken 96 bis 127
168.0	TRACK_ID1	DWORD	DW#16#0	Spurkennbits der Spuren 0 bis 31
172.0	ACTPOS1	DINT	L#0	Aktuelle Position

"Steuersignale für das Nockensteuerwerk" setzen

Definition

Mit der Funktion "Steuersignale für das Nockensteuerwerk" geben Sie die Nockenbearbeitung sowie die Spuren frei.

Ablauf der Funktion

1. Setzen Sie die gewünschten Bits im Kanal-DB.
2. Die Daten werden bei jedem Aufruf der Anweisung CAM_CTRL bzw. CAM_CTRL_452 zur Baugruppe übertragen.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.4	CAM_EN	BOOL	FALSE	1 = Nockenbearbeitung freigeben
16.0	TRACK_EN	WORD	W#16#0	Nockenspuren 0 bis 12 freischalten Bit 0 = Spur 0

Auswirkungen

Die Nockenbearbeitung wird je nach Freigabe gestartet oder angehalten.

Die Spurkennbits der freigegebenen Spuren werden an die Spursignale und die Digitalausgänge weitergegeben.

"Rückmeldesignale für das Nockensteuerwerk" abfragen

Definition

Mit der Funktion "Rückmeldesignale für das Nockensteuerwerk" werden Sie über den aktuellen Zustand des Nockensteuerwerks und der Spursignale informiert. Konsistenz zwischen der gemeldeten Position und den Spursignalen ist nicht gewährleistet.

Ablauf der Funktion

Die Daten werden bei jedem Aufruf der Anweisung CAM_CTRL bzw. CAM_CTRL_452 im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.4	CAM_ACT	BOOL	FALSE	1 = Nockenbearbeitung läuft
26.0	ACT_POS	DINT	L#0	Aktuelle Position der Achse
30.0	TRACK_OUT	DWORD	DW#16#0	Aktuelle Spursignale der Spuren 0 bis 31 Bit 0 = Spur 0

"Rückmeldesignale für die Diagnose" abfragen

Definition

Mit der Einstellung "Rückmeldesignale für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Ablauf der Einstellung

1. Wenn die Baugruppe einen neuen Eintrag in den Diagnosepuffer macht, setzt sie in der Rückmeldeschnittstelle das Bit DIAG. Jedes Auftreten eines Fehlers (Seite 620) bewirkt einen Eintrag in den Diagnosepuffer.
2. Wenn die Baugruppe einen Schreibauftrag mit fehlerhaften Daten erkennt, setzt sie in der Rückmeldeschnittstelle das Bit DATA_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen.
3. Die Anweisung CAM_CTRL bzw. CAM_CTRL_452 legt die Rückmeldeschnittstelle im Kanal-DB ab.
4. Nachdem der Diagnosepuffer von der Anweisung CAM_DIAG bzw. CAM_DIAG_452 gelesen wurde, setzt die Baugruppe das Bit DIAG wieder auf 0.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler

2.1.4.3 FM 452 konfigurieren

2.1.4.4 FM 452 parametrieren

Maßsystem

Maßsystem auswählen

Maßsystem auswählen

Sie können für die Eingabe und Ausgabe der Daten unter den folgenden Maßsystemen wählen:

- Millimeter (Voreinstellung)
- Inch
- Grad
- Impulse

Hinweis

Wenn Sie das Maßsystem im Dialog "Maßsystem" verändern, dann werden die Werte in das neue System umgerechnet. Dabei können Rundungsfehler auftreten.

Wenn Sie das Maßsystem über das Maschinendatum ändern, werden die Werte nicht automatisch umgerechnet.

Wenn das Maßsystem von oder nach "Impulse" wechselt, wird die Nockenbearbeitung abgeschaltet und die Achse ist nicht mehr synchronisiert.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
8.0	UNITS	DINT	L#1	**Maßsystem** Zulässige Werte: • 1 = 10^{-3} mm • 2 = 10^{-4} inch • 3 = 10^{-4} grad • 4 = 10^{-2} grad • 5 = Impulse • 6 = 10^{-3} grad

Standard-Maßsystem in der Online-Hilfe

Wir verwenden für die Angabe der Grenzwerte das Maßsystem mm. Für die Bestimmung der Grenzwerte in den anderen Maßsystemen nehmen Sie deshalb folgende Umrechnung vor:

Für die Umrechnung von		berechnen Sie
mm → inch		Grenzwert (inch) = Grenzwert (mm) × 0,1 [1)]
mm → grad	10^{-4} (4 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 0,1
	10^{-3} (3 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 1
	10^{-2} (2 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 10
mm → Impulse		Grenzwert (Impulse) = Grenzwert (mm) × 1000

1) Die Anzahl der Nachkommastellen hat beim Maximalwert Auswirkungen auf die Anzahl der Stellen vor dem Komma. Im Maßsystem Inch werden 4 Nachkommastellen verwendet, somit können maximal 100 000,0000 inch eingegeben werden. Das Maßsystem Millimeter verwendet 3 Nachkommastellen, hier können maximal 1 000 000,000 mm eingegeben werden.

Achse

Achsart festlegen

Achsart auswählen

Achsarten

Das Nockensteuerwerk unterstützt folgende Achsarten:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

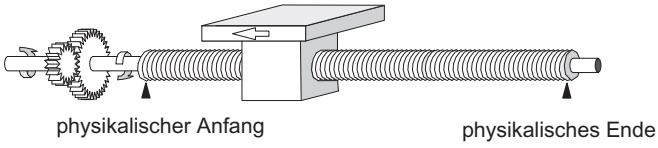

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

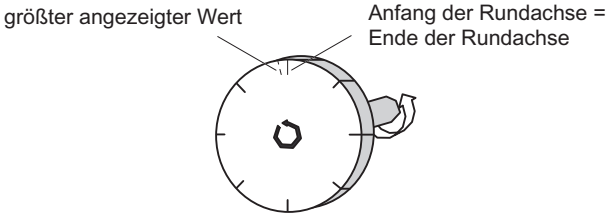

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	AXIS_TYPE	DINT	L#0	**Achsart** Zulässige Werte: • 0 = Linearachse • 1 = Rundachse

Software-Endschalter festlegen

Software-Endschalter

Software-Endschalter gibt es nur für die Linearachse.

Der Bereich, der durch die Software-Endschalter begrenzt wird, ist der Arbeitsbereich. Die Grenzen des Arbeitsbereichs können von der FM x52 überwacht werden.

Der Software-Endschalter Anfang (SEA) muss immer kleiner sein als der Software-Endschalter Ende (SEE).

Die Software-Endschalter sind aktiv, wenn die FM x52 synchronisiert ist.

- Bei Verwendung eines Inkrementalgebers oder Initiators:
  Nach jedem Anlauf der FM x52 ist die Achse zunächst nicht synchronisiert. Erst nach einer Synchronisation werden die parametrierten Software-Endschalter überwacht.
- Bei Verwendung eines Absolutwertgebers (SSI):
  Die FM x52 ist synchronisiert, nachdem sie ein vollständiges fehlerfreies Telegramm empfangen hat. Ab diesem Zeitpunkt werden die Software-Endschalter überwacht. Der von Ihnen verwendete Absolutwertgeber muss mindestens den Arbeitsbereich inklusive der Software-Endschalter abdecken.

Zusammenhang zwischen Arbeitsbereich, Geberbereich und Verfahrbereich

- Der Arbeitsbereich ist der Bereich, den Sie für Ihre Aufgabe durch die Software-Endschalter bestimmen.
- Der Geberbereich ist der vom Geber eindeutig abgedeckte Bereich. Der Geberbereich wird bei einer Linearachse von der Baugruppe symmetrisch über den Arbeitsbereich gelegt. D. h., die Baugruppe verschiebt den Geberbereich so, dass die Abstände zwischen den Software-Endschaltern und den Enden des Geberbereichs jeweils gleich sind (siehe Bild unten).
- Der Verfahrbereich ist der Wertebereich, den die FM x52 verarbeiten kann. Der Verfahrbereich ist abhängig von der Auflösung.

Es gilt: Verfahrbereich ≥ Geberbereich ≥ Arbeitsbereich

Das folgende Bild zeigt den Zusammenhang zwischen Software-Endschaltern, Arbeitsbereich, Geberbereich und Verfahrbereich.

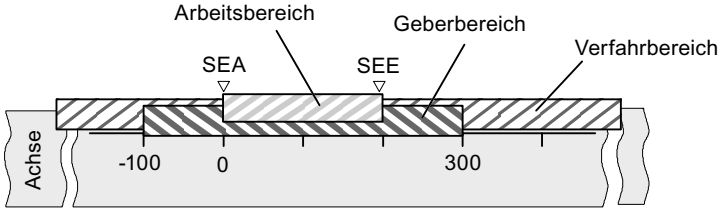

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L# -100 000 000	**Software-Endschalter Anfang**
68.0	SSW_END	DINT	L# 100 000 000	**Software-Endschalter Ende** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm

Ende der Rundachse festlegen

Ende der Rundachse

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Der größte Wert, der bei einer Rundachse angezeigt wird, ist:

Ende der Rundachse [µm] - Auflösung [µm/Impuls] × 1 [Impuls]

Beispiel: Ende der Rundachse 1000 mm

Die Anzeige springt:

- bei positiver Drehrichtung von 999 mm auf 0 mm.
- bei negativer Drehrichtung von 0 mm auf 999 mm.

Rundachse mit Absolutwertgebern

Bei einer Rundachse mit einem Absolutwertgeber muss sich der Rundachsenbereich (0 bis Ende der Rundachse) genau mit der Gesamtschrittzahl des Absolutgebers decken:

$$\text{Rundachsenende [µm]} = \text{Anzahl Umdrehungen (Geber)} * \frac{\text{Schritte (Geber) [Imp]}}{\text{Umdrehung}} * \text{AUFL} \left[\frac{\text{µm}}{\text{Imp}}\right]$$

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
16.0	ENDROTAX	DINT	L#100000	**Ende der Rundachse** Zulässiger Wertebereich: • 1 µm bis +1 000 000 000 µm

Synchronisation festlegen

Art für "Referenzpunkt nachtriggern" auswählen

Referenzpunkt nachtriggern

Mit der Art für "Referenzpunkt nachtriggern (Seite 566)" bestimmen Sie die Bedingungen für die Synchronisation der Achse bei Betrieb mit einem Inkrementalgeber oder einem Initiator.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
52.0	RETR_TYPE	DINT	L#0	**Art für "Referenzpunkt nachtriggern"** Zulässige Werte: • 0 = Referenzpunktschalter und Nullmarke Richtung + • 1 = Referenzpunktschalter und Nullmarke Richtung - • 6 = Nur Referenzpunktschalter • 7 = Nur Nullmarke

Referenzpunktkoordinate festlegen

Referenzpunktkoordinate

- **Inkrementalgeber und Initiator:**
  Durch den Funktionsschalter "Referenzpunkt nachtriggern" und ein Synchronisationsereignis, festgelegt durch die Art für "Referenzpunkt nachtriggern" (Seite 580), wird diesem Ereignis die Referenzpunktkoordinate zugeordnet.
- **Absolutwertgeber (SSI)**
  Eine parametrierte Achse mit einem Absolutwertgeber ist nach der Übertragung des ersten fehlerfreien SSI-Telegramms immer synchronisiert.
  Lesen Sie hierzu auch die Beschreibung der Absolutwertgeberjustage (Seite 582) , die das Zusammenspiel der Absolutwertgeberjustage mit den anderen Daten beschreibt.

Der Wert der Referenzpunktkoordinate muss innerhalb des Arbeitsbereichs liegen:

- **Linearachse**
  Einschließlich der Software-Endschalter.
- **Rundachse**
  Größer oder gleich 0 und kleiner als der Wert "Ende der Rundachse" (0 ≤ Referenzpunktkoordinate < Ende der Rundachse).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis +1 000 000 000 µm

Absolutwertgeberjustage festlegen

Absolutwertgeberjustage ermitteln

Definition

Absolutwertgeberjustage ist nur für einen Absolutwertgeber (SSI) erforderlich.

Mit der Absolutwertgeberjustage und der Referenzpunktkoordinate wird der Wertebereich des Gebers eindeutig auf das Koordinatensystem der Achse abgebildet. Es wird der Geberwert ermittelt, welcher der Referenzpunktkoordinate auf der Achse entspricht. Der Wert muss kleiner sein als die Gesamtschrittzahl des Absolutwertgebers.

Weitere Informationen finden Sie unter "Vorgehen für eine Absolutwertgeberjustage (Seite 582)" sowie in einem Beispiel (Seite 583).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
48.0	ENC_ADJ	DINT	L#0	**Absolutwertgeberjustage** Zulässiger Wertebereich: • 0 bis (2^{25} - 1)

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
98.0	REFPT	DINT	L#0	**Bezugspunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis +1 000 000 000 µm

Vorgehen für eine Absolutwertgeberjustage

Richtige Absolutwertgeberjustage ermitteln

Nach dem ersten Parametrieren müssen Sie weitere Schritte ausführen, damit eine korrekte Beziehung zwischen Geber und Koordinatensystem hergestellt wird. Der Ablauf ist unter Verwendung der Parametrierdialoge dargestellt.

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Ende" sein.
2. Rufen Sie die Einstellung "Bezugspunkt setzen (Seite 557)" mit der Koordinate des unter 1. definierten Punktes auf.
   Die FM x52 ermittelt nun für die im Kanal-DB eingetragene Bezugspunktkoordinate (REFPT im Kanal-DB) einen Geberwert, die Absolutwertgeberjustage. Diesen Wert können Sie mit der Einstellung "Geberdaten (Seite 573)" auslesen.

3. Tragen Sie den ausgelesenen Wert in das Feld "Absolutwertgeberjustage" ein.
4. Speichern Sie Ihre Parametrierung mit der Exportfunktion in den entsprechenden Parameter-DB.
5. Laden Sie die Daten in HWCN zur CPU.
6. Damit die Daten aktiv werden, führen Sie einen Neustart der CPU durch.

Hinweis

Diesen Abgleich führen Sie bei der Inbetriebnahme einmal durch. Die FM x52 ist nach einer Parametrierung im Anlauf synchronisiert, sobald ein vollständiges, fehlerfreies Telegramm des Gebers nach dem Anlauf empfangen wurde.

Alternative: Mechanische Justage eines Gebers

Eine korrekte Beziehung zwischen dem Koordinatensystem und dem Geber erreichen Sie auch folgendermaßen:

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Anfang" sein.
2. Tragen Sie diesen Koordinatenwert in den Maschinendaten als "Referenzpunktkoordinate" ein.
3. Lesen Sie den an dieser Position angezeigten Geberwert mit der Einstellung "Geberdaten (Seite 573)" aus.
4. Tragen Sie diesen Wert als "Absolutwertgeberjustage" in die Maschinendaten ein.

Nach der Parametrierung wird dann immer ein korrekter Istwert angezeigt.

Anstelle der Schritte 3 und 4 können Sie auch den Geber über "Reset", falls vorhanden, auf Null setzen. Dann tragen Sie den Wert "0" als Absolutwertgeberjustage in die Maschinendaten ein.

Beispiel für eine Absolutwertgeberjustage

Beispiel einer Absolutwertgeberjustage

Für das Beispiel gelten folgende Annahmen:

- Referenzpunktkoordinate = -125 mm
- Arbeitsbereich von SSW_STRT = -1000 mm bis SSW_END = 1000 mm
- Absolutwertgeberjustage = 0
- Geberbereich = 2048 Inkremente mit einer Auflösung von 1 mm/Impuls
- Der verwendete Absolutwertgeber lässt sich nicht exakt mechanisch justieren und besitzt auch nicht die Möglichkeit, den Geberwert gezielt zu setzen.

Das folgende Bild zeigt ein Beispiel einer Absolutwertgeberjustage.

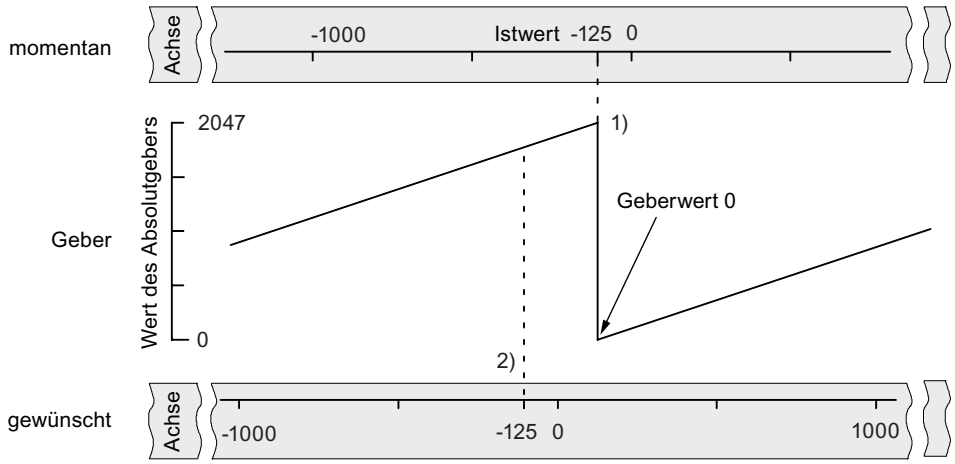

1) Zuordnung des Koordinatensystems zu den Geberwerten mit der eingestellten Absolutwertgeberjustage. Der Geberwert 0 entspricht dem Istwert -125.

2) Gewünschte Zuordnung des Koordinatensystems zum Geber. An dieser Position soll die Koordinate -125 sein.

Ergebnis nach "Bezugspunkt setzen"

Nach "Bezugspunkt setzen" sieht die Beziehung wie folgt aus:

Der Referenzpunktkoordinate auf der Achse (-125) wird der aus der Absolutwertgeberjustage ermittelte Geberwert (1798) zugeordnet. Das folgende Bild zeigt diese Zuordnung.

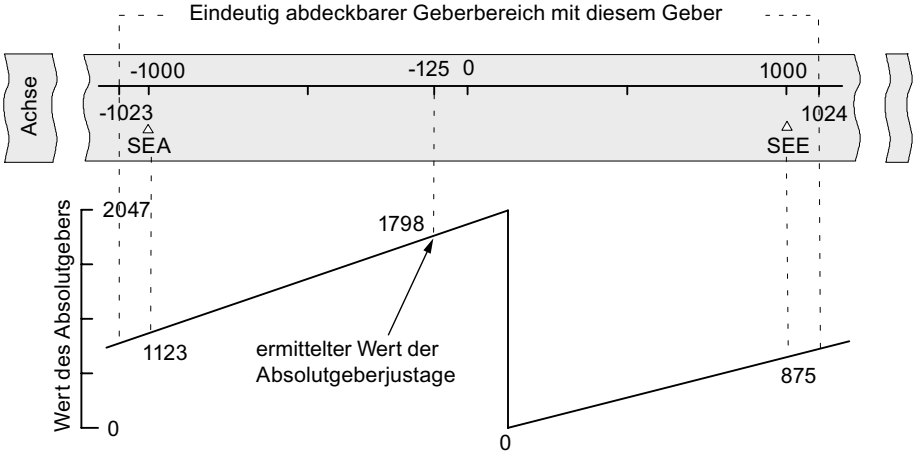

Der Geber liefert 2048 eindeutige Werte. Der Arbeitsbereich wird durch die Software-Endschalter festgelegt. Aufgrund der gewählten Auflösung von 1 mm pro Impuls kann der Geber aber einen größeren Arbeitsbereich abdecken als durch die Software-Endschalter vorgesehen ist.

Bei der eingestellten Auflösung ist der Arbeitsbereich bereits mit 2001 Werten abgedeckt. Es bleiben im Beispiel deshalb 47 Impulse "übrig", die sich symmetrisch um den Arbeitsbereich legen.

Hysterese festlegen

Hysterese

Der Wertebereich der Hysterese (Seite 458) ist abhängig von der Auflösung (Seite 591).

Der maximale Eingabewert ist wie folgt:

- Bei Linearachsen: Maximaler Eingabewert < ¼ des Arbeitsbereichs
- Bei Rundachsen: Maximaler Eingabewert < ¼ des Rundachsenbereichs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
80.0	HYS	DINT	L#0	**Hysterese** Zulässiger Wertebereich: • 0 … 65 535 [Imp] × Auflösung [µm/Imp]

Wegnocken bei Hysterese

Ein Wegnocken schaltet ein, wenn folgende Bedingungen erfüllt sind:

- Der erkannte Istwert liegt innerhalb des Wegnockens.
- Die Hysterese ist nicht aktiv.

Je nach der Position, an welcher der Richtungswechsel stattfindet, kann die Lage eines Schaltpunkts variieren.

Das folgende Bild zeigt das Schaltverhalten eines Wegnockens bei Hysterese.

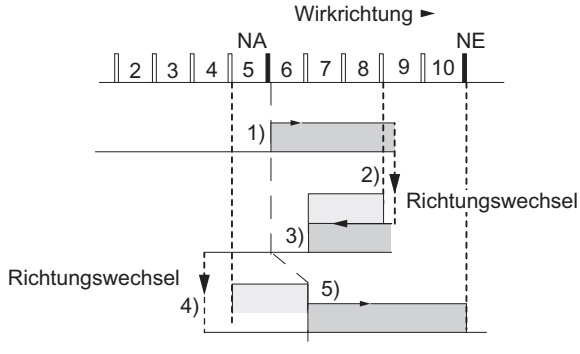

	Nocken
	Hysterese
1)	Nocken schaltet ein, da der Nockenanfang in positiver Wirkrichtung überfahren wird.
2)	Nocken bleibt trotz Richtungswechsel ein, wegen Hysterese.
3)	Nocken schaltet aus bei Ende der Hysterese.
4)	Nocken bleibt aus nach dem Richtungswechsel.
5)	Nach Ende der Hysterese schaltet Nocken wieder ein.

Hinweis

Wegnocken, die kürzer sind als die Hysterese, können bei einem Richtungswechsel von der Hysterese nicht verkürzt werden.

Zeitnocken bei Hysterese

Ein Zeitnocken schaltet ein, wenn folgende Bedingungen erfüllt sind:

- Der Nockenanfang wird in Wirkrichtung überfahren.
- Die Hysterese ist nicht aktiv.

Hinweis

Ist der Bereich zwischen Umkehrpunkt und Nockenanfang des Zeitnockens kleiner als die Hysterese, wird der Zeitnocken von der Hysterese verdeckt.

Das folgende Bild zeigt das Schaltverhalten eines Zeitnockens bei Hysterese. Dargestellt ist ein Zeitnocken, der **nicht** wieder einschaltet.

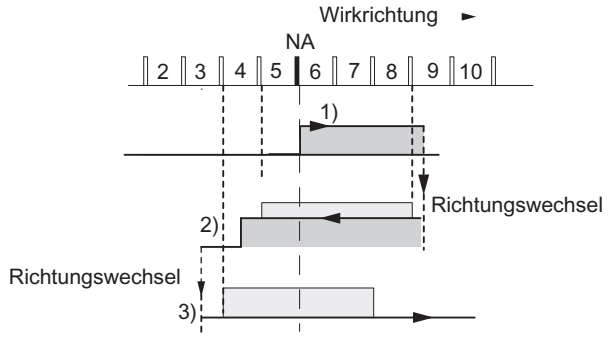

	Nocken
	Hysterese
1)	Nocken schaltet ein, da der Nockenanfang in positiver Wirkrichtung überfahren wird.
2)	Nocken bleibt nach dem Richtungswechsel bis zum Ablauf der Einschaltzeit eingeschaltet, unabhängig von der Hysterese.
3)	Nockenanfang wird von der Hysterese überdeckt, Nocken schaltet nicht.

Simulationsgeschwindigkeit festlegen

Simulationsgeschwindigkeit

Dieses Maschinendatum bestimmt die Simulationsgeschwindigkeit für die "Simulation". Beachten Sie dazu auch die Informationen unter "Einfluss von Einstellungen auf das Schaltverhalten von Zeitnocken (Seite 551)".

Die tatsächliche Simulationsgeschwindigkeit V_{sim} kann von der eingegebenen Simulationsgeschwindigkeit $V_{sim, V}$ abweichen und wird nach der folgenden Formel berechnet:

$$V_{Sim} = \frac{6 * 10exp7 * AUFL}{Ganzzahl\left(\frac{6 * 10exp7 * AUFL}{V_{Sim, V}}\right)}$$

In dieser Formel bedeutet:

- V_{sim}: Simulationsgeschwindigkeit, die die FM 352 einstellt. Einheit: $^{\mu m}/_{min}$
- $V_{sim, V}$: Simulationsgeschwindigkeit, die Sie in den Maschinendaten vorgeben. Einheit: $^{\mu m}/_{min}$
- AUFL: Auflösung, die sich aus den Geberdaten ermittelt. Einheit: $^{\mu m}/_{Impuls}$
- Ganzzahl ( ): von diesem Ausdruck wird für die weitere Berechnung nur der Vorkommawert weiter berücksichtigt. Dieser Ausdruck muss bei allen Berechnungen im Bereich von 2 … 65536 liegen.

Aufgrund der Zusammenhänge (siehe obige Formel) verändert sich die tatsächliche Simulationsgeschwindigkeit sprunghaft.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	SIM_SPD	DINT	L#0	**Simulationsgeschwindigkeit** Zulässiger Wertebereich: • 0 = Stillstand • 5×10^8 = Größtmögliche Einstellung der Baugruppe Innerhalb dieses Bereichs ist die Simulationsgeschwindigkeit abhängig von der Auflösung (Seite 591): 1000 × Auflösung ≤ Simulationsgeschwindigkeit ≤ 3×10^7 × Auflösung

Minimalen Kantenabstand festlegen

Minimaler Kantenabstand

Mit diesem Maschinendatum definieren Sie einen Bereich nach dem Erkennen des Messbeginns bei einer Kantenerfassung. Wenn das Messende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen.

Der Messbeginn wird erst nach Verfahren des Wegstücks "minimaler Kantenabstand" gemeldet.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	**Minimaler Kantenabstand** Zulässiger Wertebereich: • 0 ... 1 000 000 000 µm

Geber

Signalart auswählen

Signalart

Der Geber liefert Weginformation zur Baugruppe, die diese auswertet und mit der Auflösung in einen Istwert umrechnet.

Sie können die Signalart von einem der folgenden Geber (Seite 462) wählen:

- 5 V Differenzsignal (Seite 463)
  Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B und N, /N einsetzen.
- 24 V Spur A+B phasenversetzt (Seite 463)
  Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.
- SSI absolut (Seite 465)
  Wählen Sie die Signalart "SSI absolut", wenn Sie einen Absolutwertgeber mit serieller Schnittstelle einsetzen.
- 24 V Initiator (Seite 465)
  Wählen Sie die Signalart "24 V Initiator", wenn Sie einen 24 V-Initiator mit dem Signal A* einsetzen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
20.0	ENC_TYPE	DINT	L#1	**Signalart** Zulässige Werte: • 1 = 5 V inkrementell • 2 = 24 V inkrementell • 3 = SSI 13 Bit Telegrammlänge • 4 = SSI 25 Bit Telegrammlänge • 5 = Mithören • 6 = 24 V Initiator vorwärts (positiv) • 7 = 24 V Initiator rückwärts (negativ) • 8 = SSI 13 Bit (rechtsbündig) • 9 = SSI 25 Bit (rechtsbündig) • 10 = Mithören (rechtsbündig)

Auflösung festlegen

Weg pro Geberumdrehung festlegen

Weg pro Geberumdrehung

Mit dem Maschinendatum "Weg pro Geberumdrehung" geben Sie der FM x52 bekannt, welchen Weg das Antriebssystem je Geberumdrehung zurücklegt.

Der Wert "Weg pro Geberumdrehung" ist abhängig vom Aufbau der Achse und von der Art, wie der Geber angebracht ist. Sie müssen alle Übertragungsglieder wie Kupplungen oder Getriebe berücksichtigen. Das folgende Bild zeigt beispielhaft zwei unterschiedliche Geberanordnungen.

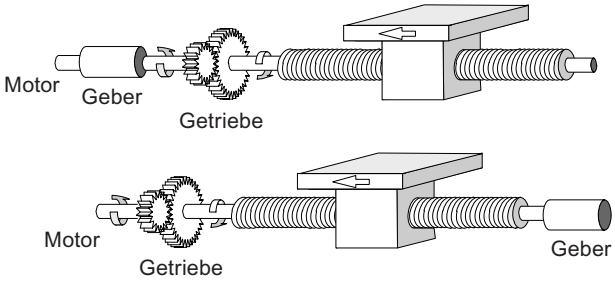

Der Zusammenhang zwischen den Maschinendaten "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" ist unter Auflösung (Seite 591) beschrieben.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
24.0	DISP_REV	DINT	L#80000	**Weg pro Geberumdrehung** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm

Inkremente pro Geberumdrehung festlegen

Inkremente pro Geberumdrehung

Das Maschinendatum "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt. Aus diesem Wert und dem Maschinendatum "Weg pro Geberumdrehung" ermittelt die FM x52 die Auflösung.

Inkrementalgeber

Als Eingabe (Seite 463) ist jeder Wert aus dem Wertebereich möglich. Die Baugruppe wertet die Inkremente 4fach aus.

Initiator

Es ist jeder Wert aus dem Wertebereich möglich.

Absolutwertgeber

Die Grenzen sind bei den einzelnen Geberarten verschieden:

Geberart	Telegrammlänge / -art	Wertebereich	als Linearachse verwendbar
Singleturn-Geber	13 Bit halber Tannenbaum	64 ... 8192 in 2er Potenzen	
Singleturn-Geber	13 Bit rechtsbündig	64 ... 8192 alle Werte	X
Singleturn-Geber	25 Bit rechtsbündig	64 ... 2^{25} alle Werte	X
Multiturn-Geber	25 Bit Tannenbaum	64 ... 8192 in 2er Potenzen	
Multiturn-Geber	25 Bit rechtsbündig	64 ... 2^{24} alle Werte	
Mithören	Tannenbaum	64 ... 8192 in 2er Potenzen	
Mithören	rechtsbündig	64 ... 2^{25} alle Werte	X
Sondereinstellung: Multiturn-Geber als Singleturn-Geber	25 Bit halber Tannenbaum	64 ... 8192 in 2er Potenzen	

Hinweis

Die Anzahl der Impulse eines Gebers ermittelt sich aus den beiden Daten "Inkremente pro Geberumdrehung" mal "Anzahl Umdrehungen" (siehe unter "Auflösung (Seite 591)").

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
32.0	INC_REV	DINT	L#500	**Inkremente pro Geberumdrehung** Zulässiger Wertebereich: • 1 bis 2^{25} Hinweis: Im Maßsystem "Impulse" ist dieser Eintrag ohne Bedeutung.

Auflösung

Auflösung

Die Auflösung ist ein Maß für die Genauigkeit der Nockenbearbeitung. Sie bestimmt auch den möglichen maximalen Verfahrbereich.

Die Auflösung (AUFL) berechnen Sie wie in der folgenden Tabelle gezeigt:

	Inkrementalgeber	Absolutwertgeber / Initiator
Eingangswerte	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – Impulsauswertung: 4fach – 1 Inkrement = 4 Impulse	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung. – 1 Inkrement = 1 Impuls
Berechnung	$\text{AUFL} = \dfrac{\frac{\text{Weg}}{\text{Geberumdrehung}}}{\frac{\text{Impulse}}{\text{Geberumdrehung}}}$	

Hinweis

Im Maßsystem "Impulse" hat die Auflösung immer den Wert 1.

Alle Positionsangaben werden auf das ganzzahlige Vielfache der Auflösung gerundet. Somit können sich die eingegebenen und die verwendeten Werte unterscheiden.

Wertebereich der Auflösung

Den Wertebereich für die Auflösung müssen Sie auf das zugrundeliegende Maßsystem umrechnen. Durch die Wahl der beiden Werte "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" muss die Auflösung in diesen Bereich gelegt werden.

Das gewählte Maßsystem bestimmt den Wertebereich der Auflösung:

Maßsystem	Angaben in ...	Wertebereich der Auflösung
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ mm .... 1000×10^{-3} mm/Impuls
inch	10^{-4} inch	$0{,}1 \times 10^{-4}$ inch .... 1000×10^{-4} inch/Impuls

Maßsystem	Angaben in ...	Wertebereich der Auflösung
grad	10^{-4} grad	$0{,}1 \times 10^{-4}$ grad .... 1000×10^{-4} grad/Impuls
	10^{-3} grad	$0{,}1 \times 10^{-3}$ grad .... 1000×10^{-3} grad/Impuls
	10^{-2} grad	$0{,}1 \times 10^{-2}$ grad .... 1000×10^{-2} grad/Impuls
Impulse	1 Impuls	1

Beispiel

- Ein Inkrementalgeber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 5000
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 4 Impulse

  Damit ergibt sich die Auflösung (4fach-Auswertung):

$$\text{Auflösung} = \frac{1000\ \text{mm}}{5000\ \text{Inkremente}} = 0{,}2000\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{Impulse}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impuls}}$$

- Ein SSI-Geber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 4096
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 1 Impuls

  Damit ergibt sich die Auflösung:

$$\text{Auflösung} = \frac{1000\ \text{mm}}{4096\ \text{Inkremente}} = 0{,}2441\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2441\ \frac{\text{mm}}{\text{Impuls}}$$

Abhängigkeit des Verfahrbereichs von der Auflösung

Der Verfahrbereich wird begrenzt durch die Zahlendarstellung in der FM x52. Diese Zahlendarstellung variiert in Abhängigkeit der Auflösung. Achten Sie deshalb darauf, dass die Vorgaben immer innerhalb der zulässigen Grenzen liegen.

Die folgende Tabelle zeigt den maximalen Verfahrbereich:

Auflösung (AUFL) liegt im Bereich	Maximaler Verfahrbereich
0,1 µm/Impuls ≦ AUFL < 1 µm/Impuls	-10^8 µm bis 10^8 µm (-100 m bis +100 m)
1 µm/Impuls ≦AUFL ≦ 1000 µm/Impuls	-10^9 µm bis 10^9 µm (-1000 m bis +1000 m)

Abhängigkeit der Geschwindigkeit von der Auflösung

Die Geschwindigkeit, die angezeigt wird, kann sich in Abhängigkeit von der Auflösung in den folgenden Grenzen bewegen (die Angaben beziehen sich auf das Maßsystem mm):

- von 1 µm/min bis 90 m/min bei einer Auflösung < 1 µm/Impuls
- von 1 µm/min bis 900 m/min bei einer Auflösung ≥ 1 µm/Impuls

Die Geschwindigkeit wird von der Baugruppe alle 4 ms ermittelt und geglättet.

Sie hat mindestens eine Ungenauigkeit von einem Impuls/4 ms und ist für Regelungszwecke nicht geeignet.

Zählrichtung auswählen

Zählrichtung

Mit dem Maschinendatum "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an. Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

- normal = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen aufsteigenden Lageistwerten
- invertiert = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen absteigenden Lageistwerten

Hinweis:
Eine Vorhaltezeit in Verbindung mit einem Absolutwertgeber (SSI) und invertiert parametrierter Zählrichtung ist nicht zulässig.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
59.0	CNT_DIR	BOOL	FALSE	**Zählrichtung** Zulässige Werte: • 0 = normal • 1 = invertiert

Überwachung auswählen

Überwachung

Drahtbruch

Mit dem Aktivieren der Überwachung überwacht die FM x52 bei einem Inkrementalgeber die Signale A, /A, B, /B, N und /N. Die Überwachung erkennt:

- Drahtbruch
- Kurzschluss der einzelnen Leitungen
  Bei Inkrementalgebern ohne Nullmarken müssen Sie alternativ
  - die Drahtbruch-Überwachung ausschalten
  - die Signale N und /N extern verschalten (Seite 462)
- Flankenabstand der Zählimpulse
- Ausfall der Geberversorgung

Telegrammfehler

Die Überwachung auf Telegrammfehler bei Absolutwertgebern (SSI) ist nicht abschaltbar. Sie überwacht:

- das Telegramm auf Start- und Stoppbitfehler
- die Monoflopzeit des angeschlossen Gebers

Fehlimpulse (Inkrementalgeber)

Ein Inkrementalgeber muss zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Inkremente liefern.

Die FM x52 überprüft, ob die Nullmarke eines Inkrementalgebers zum richtigen Geberstand auftritt.

Für Geber ohne Nullmarken müssen Sie die Überwachung der Fehlimpulse abschalten.

Ebenso müssen Sie die Drahtbruchüberwachung abschalten oder die Nullmarkeneingänge N und /N extern verschalten.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Überwachungen
63.0	MON_WIRE	BOOL	TRUE	1 = Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1 = Telegrammfehler (muss immer 1 sein)
63.2	MON_PULSE	BOOL	TRUE	1 = Fehlimpulse

Daten für Absolutwertgeber (SSI) auswählen

Anzahl Geberumdrehungen

Anzahl Geberumdrehungen

Das Maschinendatum "Anzahl Geberumdrehungen" gibt es nur für Absolutwertgeber. Damit geben Sie die Anzahl der Umdrehungen an, die mit diesem Geber möglich sind. Weitere Informationen zu Absolutwertgebern finden Sie unter "Absolutwertgeber (SSI) (Seite 465)".

- **Singleturn-Geber**
  Es ist nur der Wert 1 möglich.
- **Multiturn-Geber**
  - Multiturn-Geber / Mithören (Tannenbaum): 2 ... 4096 in 2er Potenzen
  - Multiturn-Geber / Mithören (rechtsbündig): 1 ... 4096 alle Werte
- **Linearmaßstab**
  Sie können auch einen Linearmaßstab anschließen. Tragen Sie dazu den Wert 1 ein.

Die **Gesamtschrittzahl des Gebers** ist kein Maschinendatum. Sie errechnet sich wie folgt:

Gesamtschrittzahl = Inkremente pro Geberumdrehung × Anzahl Umdrehungen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.0	NO_REV	DINT	L#1024	**Anzahl Geberumdrehungen** Zulässiger Wertebereich: • 1 (Singleturn-Geber) • 1 bis 4096 (Multiturn-Geber)

Baudrate

Baudrate

Mit dem Maschinendatum "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur FM x52.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
40.0	BAUDRATE	DINT	L#0	**Baudrate** Zulässige Werte: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Telegrammlänge

Telegrammlänge

Mit der "Telegrammlänge" bestimmen Sie den Taktrahmen, den die FM x52 für Absolutwertgeber ausgibt.

Wenn Sie "Mithören" wählen, schalten Sie den Takt der FM x52 aus. Die FM x52 kann dann auch andere SSI-Telegramme mit 13 oder 25 Bit-Telegrammrahmen mithören. Die Geschwindigkeit der Übertragung richtet sich nach der Taktgeschwindigkeit der Masterbaugruppe.

Verwendete Daten im Parameter-DB

Die hier ausgewählte Telegrammlänge ist zusammen mit der Signalart unter dem Datum ENC_TYPE (Seite 588) im Parameter-DB abgelegt.

Richtung Initiator

Richtung Initiator

Mit der "Richtung Initiator" bestimmen Sie die Richtung für einen angeschlossenen Initiator:

- Positiv: entspricht Vorwärtsrichtung (aufsteigende Zählimpulse)
- Negativ: entspricht Rückwärtsrichtung (absteigende Zählimpulse)

Spuren

Ansteuerart der Spurausgänge auswählen

Ansteuerung der Spurausgänge

Mit dem Maschinendatum "Ansteuerung der Spurausgänge" legen Sie die Ansteuerart der Spurausgänge der Spuren 0 bis 15 fest. Die Ansteuerung ist möglich durch:

- Nockensteuerwerk: Die Spursignale werden von der Nockenbearbeitung der FM 452 ein- und ausgeschaltet.
- CPU: Die Spursignale geben direkt die zugehörigen Werte der Spurfreigaben im Kanal-DB wieder.

Damit sind die Spurausgänge von Ihrem Programm aus gezielt ansteuerbar.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
90.0	TRACK_OUT	WORD	W#16#0	**Ansteuerung der Spurausgänge** Zulässige Werte: • 0 = Nockensteuerwerk • 1 = CPU Bitnummer = Spurnummer

Sonderspuren auswählen

Sonderspuren

Sie können die Spuren 0, 1 und 2 als Sonderspuren parametrieren:

- Spur 0 und Spur 1 als Zählnockenspur (Seite 457)
- Spur 2 als Bremsnockenspur (Seite 457)

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Sonderspuren
99.0	SPEC_TRC0	BOOL	FALSE	1 = Spur 0 ist Zählnockenspur
99.1	SPEC_TRC1	BOOL	FALSE	1 = Spur 1 ist Zählnockenspur
99.2	SPEC_TRC2	BOOL	FALSE	1 = Spur 2 ist Bremsnockenspur

Oberen Zählwert für Zählnockenspur festlegen

Oberer Zählwert für Zählnockenspur

Mit diesem Maschinendatum legen Sie den oberen Zählwert für die parametrierte Zählnockenspur fest.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CNT_LIM0	DINT	L#2	**Oberer Zählwert für Zählnockenspur** (Spur 0)
104.0	CNT_LIM1	DINT	L#2	**Oberer Zählwert für Zählnockenspur** (Spur 1) Zulässiger Wertebereich: • 2 … 65535

Freigabeeingang auswählen

Freigabeeingang

Das Spursignal Q3 bis Q10 wird eingeschaltet, wenn folgende Bedingungen alle erfüllt sind:

- Die Spur ist freigeschaltet.
- Der entsprechende externe Freigabeeingang I3 bis I10 ist gesetzt.
- Das Spurergebnis der Spur ist 1.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Freigabeeingang
95.0	EN_IN_I3	BOOL	FALSE	1 = Spursignal Spur 3 ist mit Freigabeeingang I3 UND-verknüpft
…	…	…	…	…
95.7	EN_IN_I10	BOOL	FALSE	1 = Spursignal Spur 10 ist mit Freigabeeingang I10 UND-verknüpft

Nocken

Mengengerüst

Mengengerüst

Das Mengengerüst bestimmt die Anzahl der maximal parametrierbaren Nocken und damit die Nockenzykluszeit:

Mengengerüst	Nockenzykluszeit
16 Nocken	20,48 µs
32 Nocken	40,96 µs
64 Nocken	81,92 µs
128 Nocken	163,84 µs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	C_QTY	DINT	UDT3: L#0 UDT4: L#1 UDT5: L#2 UDT6: L#3	Mengengerüst 0 = max. 16 Nocken 1 = max. 32 Nocken 2 = max. 64 Nocken 3 = max. 128 Nocken

Nockendaten

Nockendaten

Nockendaten

Nockendaten beschreiben die Eigenschaften eines Nockens, die Zuordnung jedes Nockens zu einer Spur und das Schaltverhalten des Nockens.

Die nachfolgenden Nockendaten werden für jeden Nocken einzeln eingestellt:

- Gültigkeit (Seite 600)
- Spurnummer (Seite 600)
- Nockenart (Seite 601)
- Nockenanfang und Nockenende für Wegnocken (Seite 601)
- Nockenanfang und Einschaltzeit für Zeitnocken (Seite 602)
- Vorhaltezeit (Seite 603)
- Wirkrichtung (Seite 605)
- Prozessalarm (Seite 607)

Allgemein gilt für die Nockenbearbeitung:

- Nur die "gültig" gesetzten Nocken werden von der Baugruppe interpretiert und verarbeitet.
- Die Nocken 0 ... 7 sind prozessalarmfähig.
- Die Anzahl der parametrierbaren Nocken ist abhängig vom Mengengerüst.

Gültigkeit eines Nockens

Nocken "gültig"

Sie müssen alle Nocken "gültig" setzen, die vom Nockensteuerwerk bearbeitet werden sollen.

Hinweis

Setzen Sie nicht benutzte Nocken auf jeden Fall auf "ungültig" (CAMVALID = FALSE).

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.0	CAMVALID	BOOL	FALSE	1 = Nocken "gültig"

Spurnummer für einen Nocken festlegen

Spurnummer

Mit der Wahl der Spurnummer legen Sie für jeden Nocken die Spur fest, auf die dieser Nocken wirken soll.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+1.0	TRACK_NO	BYTE	B#16#0	**Spurnummer** Zulässiger Wertebereich: • 0 bis 31

Nockenart auswählen

Nockenart

Wählen Sie die gewünschte Nockenart aus:

- Wegnocken
- Zeitnocken

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.3	CAM_TYPE	BOOL	FALSE	**Nockenart** Zulässige Werte: • 0 = Wegnocken • 1 = Zeitnocken

Nockenanfang und Nockenende für Wegnocken festlegen

Nockenanfang (NA) / Nockenende (NE) bei Wegnocken

Nockenanfang und Nockenende legen bei einem Wegnocken die Nockenlänge fest.

Mindestlänge eines Wegnockens

Das folgende Bild zeigt die kürzesten Nocken bei positiver Bewegungsrichtung der Achse.

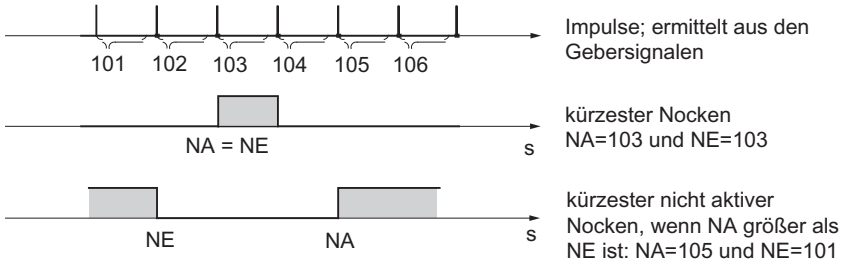

Generell gilt:

- Der nichtaktive Teil eines Nockens muss immer zwischen Nockenende (NE) und Nockenanfang (NA) einen Mindestabstand von 4 Impulsen einhalten.
- Bei NE = NA schaltet der Nocken für einen Impuls ein.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar für Wegnocken
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Nockenanfang (NA)** **Nockenende (NE)** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm

Nockenanfang und Einschaltzeit für Zeitnocken festlegen

Nockenanfang (NA) / Nockenende (NE) ≙ Einschaltzeit bei Zeitnocken

Bei einem Zeitnocken müssen Sie einen Nockenanfang und anstatt des Nockenendes eine Einschaltzeit vorgeben. Die Einschaltzeit geben Sie in 100 µs-Schritten vor. Die Zeit läuft ab dem Einschalten des Nockens.

Für die Vorgabe der Zeiten gelten folgende Bedingungen:

- 0 µs: Ein Nocken mit der Einschaltzeit 0 µs wird nie eingeschaltet
- 0 µs < t ≤ 400 µs: Die FM x52 stellt eine minimale Nockenzeit von ca. 330 µs ein.
- t > 400 µs: Die FM x52 berechnet die tatsächliche Einschaltzeit t_{tats} aus der vorgegebenen Einschaltzeit t_{Vor} nach folgender Formel:

$$t_{tats} = \text{Ganzzahl} \left(\frac{t_{Vor}}{\text{Nockenzykluszeit}} \right) * \text{Nockenzykluszeit}$$

Der maximale Fehler ist immer kleiner als die Nockenzykluszeit.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar für Zeitnocken
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Nockenanfang (NA)** **Nockenende (NE) ≙ Einschaltzeit** Zulässiger Wertebereich für Nockenanfang: • -1 000 000 000 µm bis 1 000 000 000 µm Zulässiger Wertebereich für Einschaltzeit: • (0 bis 13421) × 100 µs bei maximal 16 Nocken • (0 bis 26843) × 100 µs bei maximal 32 Nocken • (0 bis 53686) × 100 µs bei maximal 64 Nocken • (0 bis 65535) × 100 µs bei maximal 128 Nocken

Vorhaltezeit festlegen

Vorhaltezeit

Verzögerungszeiten der angeschlossenen Schaltelemente können Sie durch die Vorgabe einer Vorhaltezeit kompensieren. Die Vorhaltezeit geben Sie in 100 µs-Schritten vor. Sie können pro Nocken eine Vorhaltezeit vergeben. Die Vorhaltezeit gilt für Nockenanfang und Nockenende.

Vorhalteweg

Der Vorhalteweg eines Nockens wird in Abhängigkeit von der aktuellen Geschwindigkeit und der Vorhaltezeit laufend neu berechnet. Um diesen Weg wird der gesamte Nocken in Richtung des Istwerts verschoben. Der parametrierte Bereich wird als "statischer Bereich", der aufgrund der Vorhaltezeit berechnete Bereich als "dynamischer Bereich" bezeichnet.

Vorhalteweg = Vorhaltezeit × aktuelle Geschwindigkeit

Die Berechnung der Vorhaltewege aller Nocken wird innerhalb ¼ der längsten parametrierten Vorhaltezeit in der FM x52 durchgeführt. Wenn Sie für einen Nocken eine sehr große Vorhaltezeit parametrieren, wird die dynamische Verstellung weniger oft berechnet.

Tatsächliche Vorhaltezeit

Die tatsächliche Vorhaltezeit ermitteln Sie folgendermaßen:

1. Ermitteln Sie die Nockenzykluszeit: Dies ist die Zeit, in der die FM x52 alle Nocken komplett abgearbeitet hat. Sie ist abhängig vom parametrierten Mengengerüst.
2. Berechnen Sie die tatsächliche Vorhaltezeit mit folgender Formel:

$$\text{Vorhaltezeit}_{tats} = \text{Ganzzahl}\left(\frac{\text{Vorhaltezeit}_{V}}{\text{Nockenzykluszeit} * 4}\right) * \text{Nockenzykluszeit} * 4$$

   Dabei bedeuten:

   – Vorhaltezeit $_{tats}$ ist die Vorhaltezeit, die von der FM x52 eingestellt wird
   – Vorhaltezeit $_{V}$ ist die Vorhaltezeit, die Sie vorgegeben haben.
   – Ganzzahl ( ) bedeutet, dass von der Berechnung der Klammer nur der Vorkommawert berücksichtigt wird.

Der maximale Fehler der Vorhaltezeit $_{tats}$ ist immer kleiner als Nockenzykluszeit × 4.

Beispiel:

Folgende Werte sind vorgegeben:

- Mengengerüst: maximal 32 Nocken
- Nockenzykluszeit: 40,96 µs
- Vorhaltezeit $_{V}$ = 1000 µs

Ergebnis: Sie erhalten eine tatsächliche Vorhaltezeit von 983 µs.

Hinweis

Die tatsächliche Vorhaltezeit ist immer kleiner als die parametrierte Vorhaltezeit. Sie kann 0 werden, obwohl die parametrierte Vorhaltezeit ≥ 100 µs ist.

Der Vorhalteweg muss bei einer Rundachse kleiner als der Rundachsenbereich und der nicht aktive Teil des Nockens sein. Das muss für alle Geschwindigkeiten sichergestellt werden.

Hinweis

Eine Vorhaltezeit in Verbindung mit einem Absolutgeber (SSI) und invertiert parametrierter Zählrichtung ist nicht zulässig.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+ 10.0	LTIME	INT	0	**Vorhaltezeit** Zulässiger Wertebereich: • (0 bis 53686) × 100 µs bei max. 16 Nocken • (0 bis 65535) × 100 µs bei max. 32, 64 oder 128 Nocken

Dynamische Verstellung auf einem Nocken

Für den Bereich des Nockens unterscheiden Sie zwei Fälle:

1. Der statische und der dynamische Bereich des Nockens überlappen sich.
2. Der statische und der dynamische Bereich des Nockens überlappen sich nicht.

Die folgende Tabelle zeigt für die beiden Fälle die Ermittlung der dynamischen Verstellung.

Dynamische Verstellung	Beschreibung
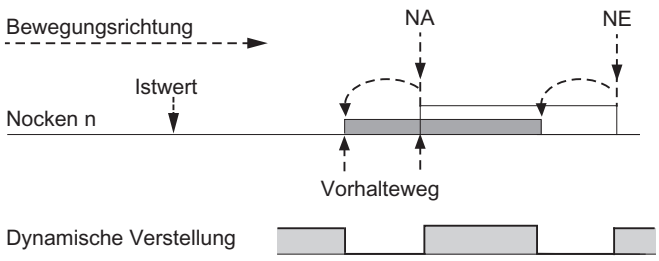	Wenn der dynamische Bereich den statischen Bereich des Nockens überlappt, gilt Folgendes: • Wenn der dynamische Bereich des Nockens erreicht ist, schaltet der Nocken ein. Gleichzeitig wird die Ermittlung einer neuen dynamischen Verstellung gesperrt. • Erreicht der Istwert den statischen Bereich des Nockens, wird die Ermittlung einer neuen dynamischen Verstellung wieder freigegeben, eine Geschwindigkeitsänderung wirkt sich auf das Nockenende aus. • Schaltet der Nocken am Ende des dynamischen Bereichs aus, ist die dynamische Verstellung bis zum Ende des statischen Bereichs des Nockens wieder gesperrt.
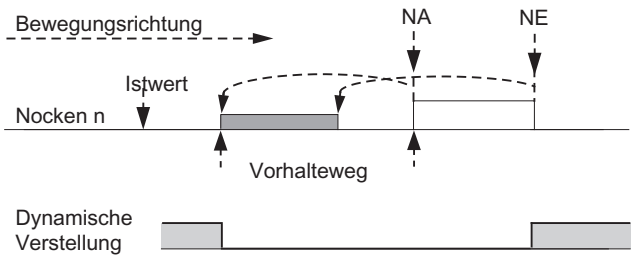	Wenn der dynamische Bereich den statischen Bereich des Nockens nicht überlappt, gilt Folgendes: • Wenn der dynamische Bereich des Nockens erreicht ist, schaltet der Nocken ein. Gleichzeitig wird die Ermittlung einer neuen dynamischen Verstellung gesperrt. • Am Ende des statischen Bereichs des Nockens wird die dynamische Verstellung wieder freigegeben.
Dynamischer Bereich Statischer Bereich Eine **neue** dynamische Verstellung ist möglich	

Hinweis

Durch eine Änderung der Drehrichtung wird die Ermittlung der dynamischen Verstellung wieder freigegeben.

Wirkrichtung auswählen

Wirkrichtung

Zwei Wirkrichtungen sind möglich:

- Positiv: Der Nocken wird am Nockenanfang eingeschaltet, wenn die Achse in Richtung steigender Istwerte läuft.
- Negativ: Der Nocken wird am Nockenende eingeschaltet, wenn die Achse in Richtung fallender Istwerte läuft.

Sie können auch beide Wirkrichtungen gleichzeitig einstellen.

Mit der Auswahl "Keine" schalten Sie den Nocken wirkungslos.

Schaltverhalten von Nocken in Abhängigkeit von der Wirkrichtung

Die folgende Tabelle zeigt das Schaltverhalten von Weg- und Zeitnocken in Abhängigkeit von der Wirkrichtung. Mit Ausnahme von Beispiel 5 wird immer die positive Wirkrichtung angenommen.

Nr.	Beschreibung	Wegnocken	Zeitnocken
1	Ein Nocken wird in Wirkrichtung überfahren	NA NE	NA Einschaltzeit
2	Ein Nocken wird entgegen der Wirkrichtung überfahren	NA NE	NA Einschaltzeit
3	Ein Nocken wird in Wirkrichtung angefahren; während der Nocken geschaltet ist, wechselt die Bewegungsrichtung der Achse	NA NE	NA Einschaltzeit x t1 t2 Einschaltzeit x = t1 + t2
4	Ein Nocken wird entgegen der Wirkrichtung angefahren; auf dem Nocken wechselt die Bewegungsrichtung der Achse in Wirkrichtung	NA NE	Nocken wird nicht geschaltet
5	Ein Nocken wird in beliebiger Richtung angefahren und in beliebiger Richtung verlassen; die Wirkrichtung ist auf **beide Richtungen** eingestellt	NA NE	NA Einschaltzeit x t1 t2 t3 Einschaltzeit x = t1 + t2 + t3

parametrierter Nocken
geschalteter Nocken

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.1	EFFDIR_P	BOOL	TRUE	1 = Wirkrichtung positiv (plus)
+0.2	EFFDIR_M	BOOL	TRUE	1 = Wirkrichtung negativ (minus)

Prozessalarm für Nocken auswählen

Prozessalarm beim Ein- / Ausschalten von Nocken

Sie können für die Nocken 0 bis 7 einstellen, ob beim Einschalten, Ausschalten oder Ein- und Ausschalten des Nockens ein Prozessalarm generiert werden soll.

Verwendete Daten im Parameter-DB

Adresse relativ	Name	Typ	Anfangswert	Kommentar
+0.4	PI_SW_ON	BOOL	FALSE	1 = Prozessalarm beim Einschalten
+0.5	PI_SW_OFF	BOOL	FALSE	1 = Prozessalarm beim Ausschalten

Freigabe Prozessalarme

Prozessalarme freigeben

Prozessalarme freigeben

In den Maschinendaten geben Sie die Prozessalarme frei.

Sie können einstellen, wann Prozessalarme generiert werden sollen:

- Messbeginn
  In der Einstellung "Kantenerfassung (Seite 563)" kann eine steigende Flanke des Digitaleingangs I1 einen Prozessalarm auslösen.
- Messende
  Sowohl in der Einstellung "Kantenerfassung (Seite 563)" als auch in der Einstellung "Längenmessung (Seite 563)" kann eine fallende Flanke des Digitaleingangs I1 einen Prozessalarm auslösen.
- Nocken ein/aus
  Sie können in den Nockendaten (Seite 607) einstellen, ob Prozessalarme beim Ein– und/ oder Ausschalten der Nocken 0 bis 7 generiert werden sollen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
3.1	PI_MEND	BOOL	FALSE	1 = Prozessalarm freigeben: Messende
3.2	PI_CAM	BOOL	FALSE	1 = Prozessalarm freigeben: Nocken ein/aus
3.5	PI_MSTRT	BOOL	FALSE	1 = Prozessalarm freigeben: Messbeginn

Auswertung eines Prozessalarms

Wenn ein Prozessalarm von der FM 452 ausgelöst wird, dann steht in der Variablen OB4x_POINT_ADDR des Prozessalarm-OB die Alarminformation zur Verfügung.

Die folgende Tabelle zeigt den Inhalt des Doppelworts OB4x_POINT_ADDR.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	0
1	0	0	Mess-be-ginn	0	0	Nocken	Mess-en-de	0
2	Nocken 7 ein	Nocken 7 aus	Nocken 6 ein	Nocken 6 aus	Nocken 5 ein	Nocken 5 aus	Nocken 4 ein	Nocken 4 aus
3	Nocken 3 ein	Nocken 3 aus	Nocken 2 ein	Nocken 2 aus	Nocken 1 ein	Nocken 1 aus	Nocken 0 ein	Nocken 0 aus

Die Alarmursache können Sie aus Byte 1 entnehmen:

- Nocken:
  Werten Sie Byte 2 und 3 entsprechend der Tabelle aus.
- Messbeginn / Messende:
  Über die Anweisung CAM_MSRM_452 können Sie die aktuellen Messwerte aus der Baugruppe lesen.

Verlorene Prozessalarme

Ist die Bearbeitung eines Prozessalarms im Prozessalarm-OB noch nicht abgeschlossen, merkt sich die Baugruppe alle folgenden Prozessalarmereignisse. Tritt ein Ereignis ein weiteres Mal auf, bevor der Prozessalarm ausgelöst werden konnte, löst die Baugruppe den Diagnosealarm "Prozessalarm verloren" aus.

Steuer- und Rückmeldeschnittstelle der FM 452

Anwendung

In speziellen Anwendungen oder in einer Alarmebene ist ein besonders schneller Zugriff auf Rückmelde- und Steuersignale erforderlich. Diese Daten erreichen Sie direkt über die Ein- und Ausgangsbereiche der Baugruppe.

Zur Anlaufkoordinierung nach jedem Anlauf der Baugruppe, z. B. nach Baugruppe stecken oder nach einem STOP-RUN-Übergang der CPU, müssen Sie die Anweisung CAM_CTRL solange aufrufen, bis durch RETVAL = 0 das Ende des Anlaufs angezeigt wird.

Hinweis

Wenn Sie direkt auf Daten der FM 452 zugreifen, dürfen Sie nur die hier beschriebenen nicht internen Daten in der angegebenen Weise verwenden. Ansonsten kann der Zugriff Ihres Anwenderprogramms auf die Baugruppe gestört werden.

Rückmeldesignale lesen durch Direktzugrifff

Die Byte-Adressen sind relativ zur Ausgangsadresse der Baugruppe angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

In AWL greifen Sie mit den Befehlen PEB (1 Byte lesen) und PED (4 Byte lesen) auf die Daten zu.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	PARA	intern	intern	DATA_ERR	intern	DIAG	intern	intern
Byte 1	0	0	0	CAM_ACT	0	0	0	0
Byte 2	intern							
Byte 3	0	0	FVAL_DONE	HYS	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								
Byte 8	TRACK_OUT							
Byte 9								
Byte 10								
Byte 11								

Steuersignale schreiben durch Direktzugriff

Die Byte-Adressen sind relativ zur Eingangsadresse der Baugruppe angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

In AWL greifen Sie auf die Daten mit den Befehlen PAB (1 Byte schreiben) und PAW (2 Byte schreiben) zu.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	intern							
Byte 1	0	CNTC1_EN	CNTC0_EN	CAM_EN	DIR_P	DIR_M	0	0
Byte 2	TRACK_EN							
Byte 3								

Beispiel: Lageistwert ACT_POS

Die Anfangsadresse der Baugruppe ist 512.

AWL

```
L PED 516      //Aktuellen Lageistwert (ACT_POS) mit Direktzugriff lesen:
               //Anfangsadresse der Baugruppe + 4
```

Parameterübertragungswege

Unter Parameter werden nachfolgend Maschinen- und Nockendaten verstanden.

Das folgende Bild zeigt die Parameterübertragungswege zur FM 452.

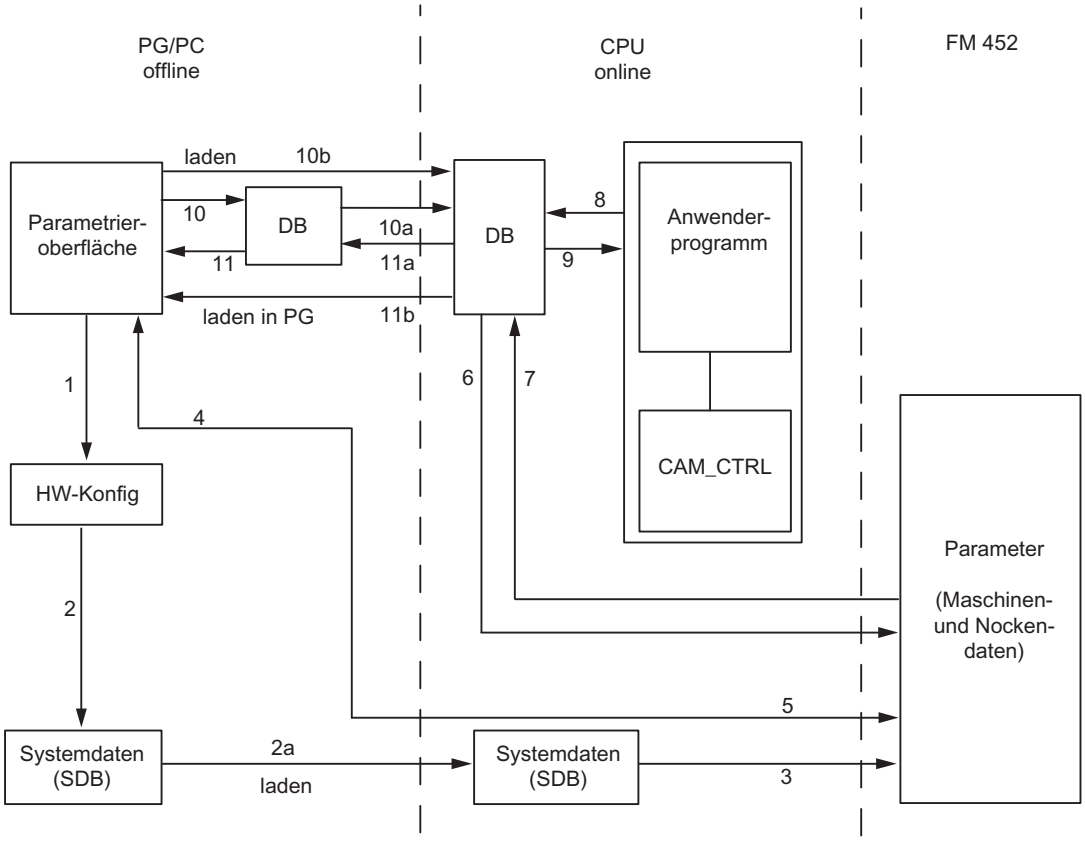

1 Parameter in der Parametrieroberfläche speichern.

2 HW-Konfiguration speichern, übersetzen und zur CPU laden.

3 Die CPU schreibt die Parameter bei Systemparametrierungen zur Baugruppe.

4 Parameter der Baugruppe ins PG laden mit dem Befehl "Zielsystem laden in PG".

5 Parameter aus der Parametrieroberfläche in die Baugruppe laden mit dem Befehl "Zielsystem laden".

6 Parameter über Aufträge des Anwenderprogramms in die Baugruppe schreiben.

7 Parameter über Aufträge des Anwenderprogramms aus der Baugruppe lesen.

8 Parameter aus dem Anwenderprogramm in den Online-DB ablegen.

9 Parameter aus dem Online-DB in das Anwenderprogramm einlesen.

10 Parameter aus der Parametrieroberfläche in den DB exportieren (Offline- oder Online-DB); ein Offline-DB muss anschließend noch in die CPU geladen werden.

11 Parameter aus einem Online- oder Offline-DB in die Parametrieroberfläche importieren.

Einige Anwendungsfälle für die Übertragung von Parametern

Anwendungsfall	Schritte
Sie bearbeiten die Parameter mit der Parametrieroberfläche. Die Baugruppe soll anschließend automatisch beim Anlauf parametriert werden.	Führen Sie die Schritte 1, 2, 3 aus.
Sie ändern Parameter bei der Inbetriebnahme im Testbetrieb in der Parametrieroberfläche.	Führen Sie die Schritte 4, 5 aus

Anwendungsfall	Schritte
Die bei der Inbetriebnahme veränderten Parameter sollen anschließend beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 1, 2, 3 aus.
Sie erstellen die Parameter mit der Parametrieroberfläche. Die Baugruppe soll beim Anlauf nur vom Anwenderprogramm über Datenbausteine parametriert werden.	Führen Sie die Schritte 10, 6 aus.
Sie möchten komfortabel Vorratsdaten für Rezepturen erstellen.	Führen Sie den Schritt 10 aus.
Sie erstellen die Parameter mit der Parametrieroberfläche. Diese sollen dem Anwenderprogramm für temporäre Änderungen zur Verfügung stehen.	Führen Sie die Schritte 1, 2, 3 für die automatische Parametrierung aus.
	Führen Sie die Schritte 10, 7 für den Zugriff durch das Anwenderprogramm aus.
Sie ändern vorhandene Parameter (ausschließlich) über das Anwenderprogramm.	Führen Sie die Schritte 7, 9, 8, 6 aus.
Sie möchten die über das Anwenderprogramm geänderten Daten mit der Parametrieroberfläche ansehen.	Führen Sie den Schritt 11 aus.
Die vom Anwenderprogramm veränderten Parameter sollen auch beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 6, 11, 1, 2, 3 aus.

2.1.4.5 FM 452 in Betrieb nehmen

2.1.4.6 FM 452 aus dem Anwenderprogramm verwenden

Einbinden der FM 452 in ein Anwenderprogramm

Für das Einbinden der FM 452 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen CAM_INIT und CAM_CTRL bzw. CAM_CTRL_452 werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 452.

Wenn Sie die Anweisung CAM_CTRL in das Projekt einbinden, dann wird der Kanal-DB automatisch als Instanz-DB zur Anweisung CAM_CTRL angelegt.
Wenn Sie die Anweisung CAM_CTRL_452 in das Projekt einbinden, dann müssen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CAM_CHANTYPE als Kanal-DB in Ihr Projekt einfügen.

Alle zu einem Kanal der Baugruppe gehörenden Parameterdaten (Maschinendaten, Nockendaten) liegen im Parameter-DB für die Baugruppe. Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CAM_PxxxTYPE in Ihr Projekt ein. Abhängig vom Mengengerüst benötigen Sie folgende DB-Typen:

- Bis zu 16 Nocken: CAM_P016TYPE
- Bis zu 32 Nocken: CAM_P032TYPE

- Bis zu 64 Nocken: CAM_P064TYPE
- Bis zu 128 Nocken: CAM_P128TYPE

Die Anweisung CAM_MSRM_452 verwenden Sie für die sofortige Auswertung der Messdaten von Längenmessung oder Kantenerfassung einer FM 452 im Prozessalarm-OB.

Außerdem gibt es für die FM 452 die Anweisung CAM_DIAG bzw. CAM_DIAG_452, mit der Sie Diagnoseinformationen in den Diagnose-DB übertragen können.

Wenn Sie die Anweisung CAM_DIAG in das Projekt einbinden, dann wird der Diagnose-DB automatisch als Instanz-DB zur Anweisung CAM_DIAG angelegt.
Wenn Sie die Anweisung CAM_DIAG_452 in das Projekt einbinden, dann müssen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ CAM_DIAGTYPE als Diagnose-DB in Ihr Projekt einfügen.

Inbetriebnahme der Baugruppe

Auch ohne Anwenderprogramm ist ein erster Test der Baugruppe mit dem Dialog "Inbetriebnahme" möglich. Beachten Sie dabei insbesondere die nachfolgende Warnung.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Zur Vermeidung von Schäden an Personen und Gegenständen beachten Sie folgende Punkte:

- Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners.
  Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Software-Ausfalls die Anlage sicher ausgeschaltet werden kann.
- Installieren Sie Hardware-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
- Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.
- Durch paralleles Steuern und Beobachten der FM 452 von Ihrem Programm und aus dem Dialog "Inbetriebnahme" kann es zu Konflikten kommen, deren Auswirkungen nicht eindeutig sind. Schalten Sie deshalb immer die CPU in den STOP-Zustand, wenn Sie mit dem Dialog "Inbetriebnahme" arbeiten, oder deaktivieren Sie Ihr Programm.

2.1.4.7 FM 452: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Fehler-LEDs der FM 452

Wo wird der Fehler angezeigt?

Die roten Fehler-LED auf der Baugruppe zeigen verschiedene Fehlerzustände an. Die LED leuchten auch bei kurzfristig auftretenden Fehlern mindestens 3 s lang.

Das folgende Bild zeigt die Lage der Fehler-LED an der FM 452.

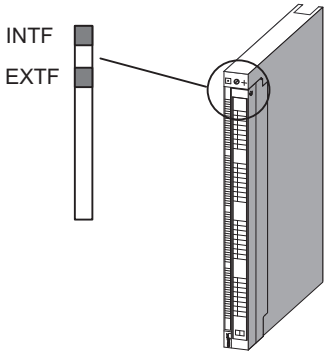

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Fehler-LED angezeigt werden.

LED	Bedeutung	Erläuterungen
INTF	Sammelfehler für interne Fehler	Diese LED zeigt folgende Fehlerzustände der FM 452 an: • Prozessalarm verloren • Watchdog abgelaufen • FM 452 nicht projektiert • FM 452 fehlerhaft parametriert Zur Beseitigung des Fehlers siehe Fehlerklassen (Seite 620).
EXTF	Sammelfehler für externe Fehler	Diese LED zeigt folgende Fehlerzustände an: • externe Hilfsspannung 24 V fehlt • Frontstecker fehlt • Drahtbruch Geber • Betriebsfehler • Telegrammfehler Absolutwertgeber • Fehlimpuls Inkrementalgeber oder Nullmarke fehlt
INTF + EXTF	Baugruppe defekt	Alle Ausgänge werden abgeschaltet. Die Baugruppe muss ausgetauscht werden.

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Alarmbearbeitung

Die FM 452 kann Prozessalarme und Diagnosealarme auslösen. Diese Alarme bearbeiten Sie in einem Alarm-OB. Wenn ein Alarm ausgelöst wird, ohne dass der zugehörige OB geladen ist, geht die CPU in STOP.

Die Bearbeitung der Diagnosealarme geben Sie folgendermaßen frei:

1. Wählen Sie die Baugruppe in HWCN aus
2. Geben Sie über "Bearbeiten > Objekteigenschaften > Grundparameter" den Diagnosealarm frei.
3. Speichern und übersetzen Sie die Konfiguration.
4. Laden Sie die Konfiguration in die CPU.

Übersicht über die Diagnosealarme

Folgende Ereignisse und Fehler lösen einen Diagnosealarm aus:

- Betriebsfehler
- Fehlerhafte Maschinendaten

- Fehlerhafte Nockendaten
- Diagnosefehler

Sie finden diese Fehler unter "Fehlerklassen (Seite 620)" ausführlich erläutert.

Reaktion der Baugruppe bei einem Fehler mit Diagnosealarm

- Die Nockenbearbeitung wird abgeschaltet.
- Die Synchronisation wird bei folgenden Diagnosealarmen gelöscht:
  - Frontstecker fehlt, externe Hilfsspannung fehlt
  - ein Nullmarkenfehler wurde erkannt, Leitungsfehler (5 V-Gebersignale)
  - der Verfahrbereich wurde verlassen (wird durch einen Betriebsfehler gemeldet)
  - Istwert setzen nicht ausführbar (wird durch einen Betriebsfehler gemeldet)
- Steuersignale werden, bis auf eine Ausnahme, nicht mehr bearbeitet.
  Ausnahme:
  Bei Überfahren der Software-Endschalter ist ein Richtungswechsel im Simulationsbetrieb noch möglich.
- Funktionsschalter und Aufträge werden weiter bearbeitet.

Baugruppe erkennt einen Fehler ("kommend")

Ein Diagnosealarm ist "kommend", wenn mindestens ein Fehler ansteht. Wurden nicht alle Fehler beseitigt, werden die restlichen anstehenden Fehler nochmals "kommend" gemeldet.

Ablauf:

1. Die FM 452 erkennt einen oder mehrere Fehler und löst einen Diagnosealarm aus. Die Fehler-LED leuchtet. Der Fehler wird in den Diagnosepuffer eingetragen.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Sie können die Startinfo des OB 82 auswerten.
4. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
5. Weitere Informationen erhalten Sie, wenn Sie die Anweisung CAM_DIAG bzw. CAM_DIAG_452 aufrufen.

Baugruppe erkennt den Übergang in den fehlerfreien Zustand ("gehend")

Ein Diagnosealarm ist nur dann "gehend", wenn der letzte Fehler auf der Baugruppe behoben wurde.

Ablauf:

1. Die FM 452 erkennt, dass alle Fehler behoben wurden und löst einen Diagnosealarm aus. Die Fehler-LED leuchtet nicht mehr. Der Diagnosepuffer wird nicht verändert.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.

3. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
4. Werten Sie das Bit OB82_MDL_DEFECT aus.
   Wenn dieses Bit "0" ist, sind keine Fehler auf der Baugruppe vorhanden. Ihre Auswertung kann hier enden.

Diagnosealarme in Abhängigkeit des CPU-Status

- Im STOP-Zustand der CPU sind die Diagnosealarme von der FM 452 gesperrt
- Werden im STOP-Zustand der CPU nicht alle der anstehenden Fehler behoben, meldet die FM 452 die noch nicht beseitigten Fehler noch einmal nach dem Übergang in den RUN-Zustand als "kommend".
- Werden alle anstehenden Fehler im STOP-Zustand der CPU behoben, dann wird der fehlerfreie Zustand der FM 452 nach dem Übergang in den RUN-Zustand **nicht** mit einem Diagnosealarm gemeldet.

Fehlerauswertung im Anwenderprogramm

Fehlerreaktion im Anwenderprogramm

Im Anwenderprogramm können Sie gezielt auf Fehler reagieren. Dafür stehen Ihnen folgende Mittel zur Verfügung:

- Die Rückgabewerte RET_VAL der eingebundenen Anweisungen als Sammelanzeige für Fehler, die während des Ablaufs der Anweisungen aufgetreten sind.
- Die Fehlerbits _ERR der Aufträge als Sammelanzeige für Fehler, die während der Bearbeitung eines Auftrags aufgetreten sind.
- Das Fehlerbit DATA_ERR als Sammelanzeige für einen Fehler, den die FM 452 bei einem Schreibauftrag erkannt hat.
- Die Fehlerkennung in JOB_ERR für die Fehlerursache bei der Kommunikation zwischen Anweisung und FM 452.
- Die Anweisung CAM_DIAG bzw. CAM_DIAG_452 zum Auslesen des Diagnosepuffers der FM 452. Hier können Sie die Fehlerursachen für Aufträge und asynchrone Ereignisse (Betriebsfehler, Diagnosefehler) erfahren.
- Diagnosealarme zur schnellen Reaktion auf Ereignisse.

Diagnosepuffer der Baugruppe

Diagnoseereignisse

Der Diagnosepuffer der Baugruppe enthält maximal 4 Diagnoseereignisse und ist als Ringpuffer organisiert.

Ein Diagnoseereignis wird in den Puffer geschrieben, wenn eine (Fehler-) Meldung "kommend" erkannt wird. Dies kann eine Meldung, ein synchroner Fehler (Datenfehler) oder auch ein asynchroner Fehler (Betriebsfehler und Diagnosefehler) sein. Aus einer Fehlerursache können

auch mehrere Einträge als Folgefehler entstehen. Gehende Meldungen erzeugen keine Einträge im Diagnosepuffer.

Für jedes Diagnoseereignis werden angegeben:

- Status (immer kommend)
- Interner Fehler
- Externer Fehler
- Fehlerklasse
- Fehlernummer
- Kanalnummer
- Nockennummer (bei Nockendatenfehler)

Wenn ein Diagnoseereignis in den Diagnosepuffer geschrieben wird, wird das Rückmeldesignal DIAG = 1 gesetzt.

Der Diagnosepuffer kann als Ganzes mit der Anweisung CAM_DIAG bzw. CAM_DIAG_452 in einen Datenbaustein (Diagnose-DB) übertragen werden oder über den Fehlerauswertungsdialog der Projektiersoftware angezeigt werden. Wird der Diagnosepuffer gelesen, setzt die Baugruppe das Rückmeldesignal DIAG = 0.

Hinweis

Wird der Diagnosepuffer gleichzeitig von der Anweisung CAM_DIAG bzw. CAM_DIAG_452 und dem Fehlerauswertungsdialog gelesen, kann es sein, dass ein neu eingetroffenes Diagnoseereignis vom Programm nicht erkannt wird.

Daten und Aufbau des Diagnose-DB

Daten und Aufbau des Diagnose-DB

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Die folgende Tabelle zeigt den Aufbau des Diagnose-DB.

Adresse	Name	Typ	Anfangswert	Kommentar
0.0	MOD_ADDR (Eintragen!)	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Kommunikationsfehler
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag

Adresse	Name	Typ	Anfangswert	Kommentar
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten letzter Eintrag

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 4: Datenfehler 5: Maschinendatenfehler 7: Nockendatenfehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer 0 … 255
+6.0	CH_NO	INT	0	Kanalnummer (immer 1)
+8.0	CAMNO	INT	0	Nockennummer 0 bis 127 bei Fehlerklasse = Nockendatenfehler

Liste der JOB_ERR-Meldungen

JOB_ERR-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80A0	32928	-32608	Negative Quittung beim Lesen von Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2 [1)]	32930	-32606	Protokollfehler bei Layer 2 (Datentransfer innerhalb PROFINET / PROFIBUS DP unterbrochen, z. B. wegen Drahtbruch, fehlendem Abschlussstecker, Parametrierfehler etc.)
80A3 [1)]	32931	-32605	Protokollfehler bei User-Interface/User (Datentransfer innerhalb PROFINET / PROFIBUS DP unterbrochen, z. B. wegen Drahtbruch, fehlendem Abschlussstecker, Parametrierfehler etc.)
80A4 [1)]	32932	-32604	Kommunikation am K-Bus gestört
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0 [1)]	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1 [1)]	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80C2 [1]	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3 [1]	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4 [1]	32964	-32572	Kommunikationsfehler
80C5 [1]	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6 [1]	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)

[1] Die Fehler 80A2 … 80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein. Meldungen der Form 7xxx zeigen temporäre Betriebszustände der Kommunikation an.

Fehlerklassen

Betriebsfehler

Fehlerklasse 1: Betriebsfehler

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt.

Die folgende Tabelle zeigt die möglichen Betriebsfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
2	**Software-Endschalter Ende überfahren**		ja
3	**Verfahrbereichsanfang überfahren**		ja
4	**Verfahrbereichsende überfahren**		ja
13	**Fliegendes Istwertsetzen nicht ausführbar**		ja
	Ursache	Die Software-Endschalter liegen nach dem fliegenden Istwertsetzen außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen ist größer als ±100 m bzw. ±1000 m.	
	Wirkung	Achse nicht synchronisiert.	

Datenfehler

Fehlerklasse 4: Datenfehler

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt.

Die folgende Tabelle zeigt die möglichen Datenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
10	**Fehlerhafte Nullpunktverschiebung**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten-Aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen (fehlerfreien) Maschinendaten auf der Baugruppe vorhanden.	
21	**Fliegendes Istwertsetzen nicht zulässig**		nein
	Ursache	Es wurde versucht, "fliegendes Istwertsetzen" bei eingeschaltetem "Referenzpunkt nachtriggern" auszuführen.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0. Es wurde versucht, "Längenmessung" und "Kantenerfassung" gleichzeitig anzuwählen.	
28	**Referenzpunkt nachtriggern nicht zulässig**		nein
	Ursache	Es wurde versucht, "Referenzpunkt nachtriggern" bei anstehendem "fliegendes Istwertsetzen" auszuführen. Es wurde versucht, "Referenzpunkt nachtriggern" bei einem SSI-Geber auszuführen.	
29	**Unerlaubtes bitcodiertes Kommando**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
30	**Falsche Vorhaltezeit**		nein
31	**Falsche Nockennummer**		nein
	Ursache	Der Nocken ist nicht gültig. Die Nockennummer ist nicht im Bereich 0 … 127.	

Nr.	Bedeutung		Diagnosealarm
32	**Falscher Nockenanfang**		nein
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Nockenanfang ist < 0 oder größer als das Ende der Rundachse.	
33	**Falsches Nockenende / falsche Einschaltzeit**		nein
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m) . Rundachse: Das Nockenende ist < 0 oder größer als das Ende der Rundachse. Der Nocken ist nicht mindestens für die Dauer eines Impulses eingeschaltet. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereiches liegen.	
35	**Fehlerhafte Istwertvorgabe beim Istwertsetzen / fliegendes Istwertsetzen**		nein
	Ursache	Die Istwertvorgabe liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw ±1000 m. Die Software-Endschalter würden nach der Ausführung der Einstellung außerhalb des Verfahrbereichs (-100 m … +100m bzw. -1000 m … +1000 m) liegen. Die sich ergebende Verschiebung aus Istwertsetzen / fliegendes Istwertsetzen wäre größer als ±100 m bzw. ±1000 m.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind keine Maschinendaten vorhanden. Auf der Achse sind keine Maschinendaten aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Eine der Einstellungen "Istwertsetzen" oder "fliegendes Istwertsetzen" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	
109	**Nockenbearbeitung läuft.**		nein
110	**Falsche Anzahl zu ändernder Nocken**		nein

Maschinendatenfehler

Fehlerklasse 5: Maschinendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Die folgende Tabelle zeigt die möglichen Maschinendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	

Nr.	Bedeutung		Diagnosealarm
6	**Falscher minimaler Kantenabstand**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben.	
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben.	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 10.	
11	**Falscher Weg pro Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Anzahl Inkremente/Geberumdrehung** (siehe "Inkremente pro Geberumdrehung (Seite 590)")		ja
14	**Falsche Anzahl Umdrehungen** (siehe "Anzahl Geberumdrehungen (Seite 595)")		ja
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereiches von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m je nach Auflösung. Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Weggeber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art für "Referenzpunkt nachtriggern"**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 6 und 7 angegeben.	
19	**Falsche Richtungsanpassung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung auf Telegrammfehler im Parameter-DB auf "FALSE" gesetzt.	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw -1000 m (je nach Auflösung).	
22	**Falscher Software-Endschalter Ende**		ja

Nr.	Bedeutung		Diagnosealarm
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
144	**Falsches Mengengerüst**		ja
	Ursache	Sie haben für das Mengengerüst einen Wert ungleich 0 bis 3 angegeben.	
145	**Falsche Hysterese**		ja
	Ursache	Die Hysterese liegt außerhalb des Bereichs 0 … 65535 × Auflösung. Die Hysterese ist größer als ¼ × Arbeitsbereich bzw. ¼ × Rundachsenbereich.	
146	**Falsche Simulationsgeschwindigkeit**		ja
	Ursache	Die Simulationsgeschwindigkeit liegt außerhalb des Bereichs 1000 × AUFL bis 3×10^7 × AUFL oder ist größer als 5×10^8 µm/min. Die Simulationsgeschwindigkeit ist intern nicht einstellbar.	
147	**Falsche Spur**		ja
	Ursache	Die Ansteuerung einer Spur außerhalb 0 … 15 (Bit 0 … 15) wurde gewählt.	
148	**Falsche Anwahl der Freigabeeingänge**		ja
	Ursache	Sie wollten eine Spur außerhalb 3 … 10 (Bit 0 … 7) durch ein externes Signal freischalten.	
149	**Falsche Sonderspuranwahl**		ja
	Ursache	Sie wollten eine Spur außerhalb 0, 1 und 2 (Bit 0, 1 und 2) als Sonderspur definieren.	
150	**Falscher oberer Zählwert Spur 0**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
151	**Falscher oberer Zählwert Spur 1**		ja
	Ursache	Sie haben einen Zählwert < 2 oder > 65535 als oberen Zählwert angegeben.	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben einen Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Weggeber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Weggeber und Rundachse: Der Weggeber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Weggeber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Nockendatenfehler

Fehlerklasse 7: Nockendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst.

Die folgende Tabelle zeigt die möglichen Nockendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Prozessalarm unzulässig**		ja
	Ursache	Sie wollen einen Prozessalarm bei einem Nocken mit einer Nockennummer > 7 angeben.	
2	**Falsche Spurnummer**		ja
	Ursache	Die Spurnummer liegt außerhalb des Bereichs 0 bis 31.	
3	**Falscher Nockenanfang**		ja
	Ursache	Der Nockenanfang liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Rundachse: Der Nockenanfang ist < 0 oder größer als das Ende der Rundachse.	
4	**Falsches Nockenende**		ja
	Ursache	Das Nockenende liegt außerhalb des Verfahrbereichs (-100 m … +100 m bzw. -1000 m … +1000 m). Der Nocken ist nicht mindestens 1 Impuls lang. Rundachse: Das Nockenende ist < 0 oder größer als das Ende der Rundachse. Bei einem inversen Nocken liegen zwischen Nockenanfang und Nockenende nicht mindestens 4 Impulse.	
5	**Falsche Einschaltzeit**		ja
	Ursache	Die Einschaltzeit ist < 0 µs. Der Maximalwert (Seite 602) ist abhängig vom Mengengerüst.	
6	**Falsche Vorhaltezeit** [1]		ja
	Ursache	Die Vorhaltezeit ist < 0 µs. Der Maximalwert (Seite 603) ist abhängig vom Mengengerüst.	
50	**Zuviele Nockensätze**		ja
	Ursache	Sie wollten mehr Nockensätze eingeben als bei diesem Mengengerüst möglich sind.	
51	**Achse in Betrieb**		ja
	Ursache	Sie wollten bei eingeschaltetem Nockensteuerwerk Nockensätze eingeben.	
52	**Achse nicht parametriert**		ja
	Ursache	Sie wollen Nockendaten eingeben, obwohl noch keine Maschinendaten aktiv sind.	

[1] Die Fehlermeldung kann auch dann auftreten, wenn Sie in Verbindung mit einem Absolutwertgeber (SSI) als Zählrichtung "invertiert" parametriert haben.

Meldungen

Fehlerklasse 15: Meldungen

Die folgende Tabelle zeigt die möglichen Meldungen und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	

Diagnosefehler

Fehlerklasse 128: Diagnosefehler

Die folgende Tabelle zeigt die möglichen Diagnosefehler mit ihren Ursachen und Wirkungen sowie Abhilfemöglichkeiten.

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen, Frontstecker fehlt	
	Wirkung	Siehe "Auslösen von Diagnosealarmen (Seite 615)": • die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern • Die FM 452 ist nicht parametriert (Rückmeldesignal PARA = 0).	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (Wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
5	**Frontstecker fehlt**		ja
	Ursache	Frontstecker ist nicht gesteckt	
	Wirkung	• Externe Hilfsspannung 24 V fehlt • Baugruppe ist nicht betriebsbereit	
	Behebung	Frontstecker aufstecken	

Nr.	Bedeutung		Diagnosealarm
51	**Watchdog abgelaufen**		ja
	Ursache	• starke Störeinflüsse auf die FM 452 • Fehler in der FM 452	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind • Tauschen der FM 452	
52	**Baugruppeninterne Versorgung ausgefallen**		ja
	Ursache	Fehler in der FM 452	
	Wirkung	• Baugruppe wird rückgesetzt • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit	
	Behebung	Tauschen der FM 452	
70	**Prozessalarm verloren**		ja
	Ursache	Ein Prozessalarmereignis wurde von der FM 452 erkannt und kann nicht gemeldet werden, da das gleiche Ereignis noch nicht vom Anwenderprogramm / der CPU verarbeitet wurde	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• OB 40 in Anwenderprogramm einbinden • Bus-Anschluss der Baugruppe überprüfen • Prozessalarm deaktivieren • Passen Sie Ihre Hardware und Software Ihren Prozessanforderungen an (z. B. schnellere CPU, Anwenderprogramm optimieren)	

Nr.	Bedeutung		Diagnosealarm
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation bei Inkrementalgebern	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden • Technische Daten der Baugruppe einhalten	
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen FM 452 und dem Absolutwertgeber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • der letzte korrekte Istwert bleibt bis zum Ende der nächsten korrekten SSI-Übertragung unverändert	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und FM 452	

Nr.	Bedeutung		Diagnosealarm
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• die Nockenbearbeitung wird ausgeschaltet • Abschalten der Spurausgänge • Löschen der Synchronisation	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden	

PID-Regelung

3

3.1 Grundlagen zum Regeln

3.1.1 Regelkreis und Stellglieder

Regelkreis

Ein einfaches Beispiel für einen Regelkreis ist die Regelung der Raumtemperatur durch eine Heizung. Die Raumtemperatur wird mit einem Sensor gemessen und einem Regler zugeführt. Dieser vergleicht die aktuelle Raumtemperatur mit einem Sollwert und berechnet einen Ausgangswert (Stellwert) für die Ansteuerung der Heizung.

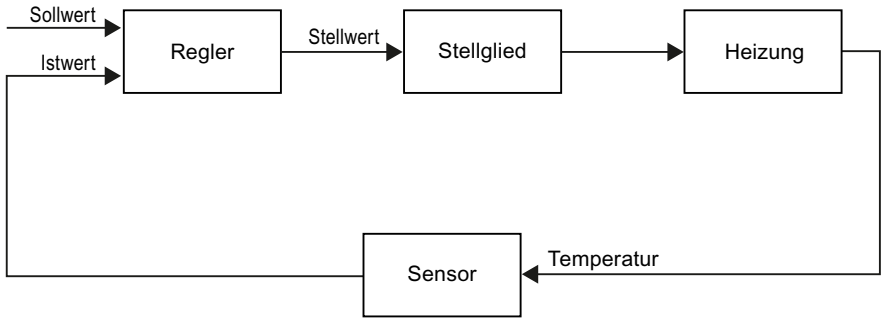

Ein korrekt eingestellter PID-Regler erreicht den Sollwert möglichst schnell und hält ihn dann konstant. Nach einer Änderung des Ausgangswerts ändert sich der Istwert häufig erst zeitlich verzögert. Dieses Verhalten soll durch den Regler ausgeglichen werden.

Stellglieder

Das Stellglied ist Bestandteil des Regelkreises und wird vom Regler beeinflusst. Dadurch wird ein Masse- oder Energiefluss verändert.

Die folgende Tabelle gibt einen Überblick über den Einsatz der Stellglieder

Einsatz für	Stellglied
flüssigen und gasförmigen Massefluss	Ventil, Klappe, Schieber
festen Massefluss, z. B. Schüttgüter	Abzugschieber, Förderband, Vibratorrinne
elektrischen Energiefluss	Schaltkontakt, Schaltschütz, Relais, Thyristor
	Stellwiderstand, Stelltransformator, Transistor

Stellglieder werden folgendermaßen unterschieden:

- Proportionale Stellglieder mit stetigem Stellsignal
  Proportional zum Ausgangswert werden Öffnungsgrade, Drehwinkel oder Positionen eingenommen. Der Ausgangswert wirkt innerhalb des Stellbereiches in analoger Weise auf den Prozess ein.
  Zu den Stellgliedern dieser Gruppe gehören federbelastete pneumatische Antriebe, aber auch motorische Antriebe mit Stellungsrückmeldung, bei denen ein Stellungsregelkreis gebildet wird.
  Der Ausgangswert wird von einem kontinuierlichen Regler erzeugt, z. B. PID_Compact.
- Proportionale Stellglieder mit pulsweitenmoduliertem Signal
  Bei diesen Stellgliedern wird im Takt der Abtastzeit ein Impuls ausgegeben, dessen Länge proportional zum Ausgangswert ist. Das Stellglied - z. B. ein Heizwiderstand oder ein Kühlaggregat - wird taktsynchron je nach Ausgangswert unterschiedlich lange eingeschaltet.
  Das Stellsignal kann entweder unipolar die Zustände "Ein" oder "Aus" annehmen oder bipolar z. B. die Werte "Auf/Zu", "Vorwärts/Rückwärts", "Beschleunigen/Bremsen" usw. repräsentieren.
  Der Ausgangswert wird von einem Zweipunktregler erzeugt, z. B. PID_Compact mit Pulsweitenmodulation.
- Integral wirkende Stellglieder mit Dreipunktstellsignal
  Häufig werden Stellglieder durch Motoren betätigt, bei denen die Einschaltdauer proportional zum Verstellweg des Drosselorgans ist. Dazu zählen z. B. Ventile, Klappen und Schieber. Trotz unterschiedlichster Bauformen ist diesen Stellgliedern gemeinsam, dass sie der Wirkung eines I-Anteils am Streckeneingang entsprechen.
  Der Ausgangswert wird von einem Schrittregler erzeugt, z. B. PID_3Step.

3.1.2 Regelstrecken

Die Eigenschaften einer Regelstrecke sind durch verfahrenstechnische und maschinentechnische Gegebenheiten festgelegt und kaum zu beeinflussen. Ein gutes Regelergebnis kann nur erreicht werden durch Auswahl eines für diese Regelstrecke geeigneten Reglertyps und dessen Anpassung an das Zeitverhalten der Regelstrecke. Genaue Kenntnisse des Typs und der Kenndaten der Regelstrecke sind deshalb für die Konfiguration von P-, I- und D-Anteil des Reglers unerlässlich.

Regelstreckentypen

Regelstrecken werden klassifiziert anhand Ihres Zeitverhaltens auf die sprunghafte Veränderung des Ausgangswerts.

Folgende Regelstrecken werden unterschieden:

- Regelstrecken mit Ausgleich
  - P-Regelstrecken
  - PT1-Regelstrecken
  - PT2-Regelstrecken
- Regelstrecken ohne Ausgleich
- Regelstrecken mit und ohne Totzeit

Regelstrecken mit Ausgleich

P-Regelstrecken

Bei Proportionalstrecken folgt der Istwert nahezu unverzögert dem Ausgangswert. Das Verhältnis von Istwert zu Ausgangswert wird angegeben durch den Proportionalbeiwert Gain der Regelstrecke.

Beispiele:

- Ventilschieber in einem Rohrsystem
- Spannungsteiler
- Untersetzung in Hydrauliksystemen

PT1-Regelstrecken

Bei einer PT1-Regelstrecke ändert sich der Istwert zunächst proportional zur Änderung des Ausgangswerts. Mit der Zeit ändert sich der Istwert langsamer - also verzögert - bis ein Endwert erreicht ist.

Beispiele:

- Feder-Dämpfer-System
- Ladung von RC-Gliedern
- Wasserbehälter, der mit Dampf aufgeheizt wird.

Häufig gelten gleiche Zeitkonstanten für Heiz- und Kühlvorgänge oder Lade- und Entladekurven. Unterscheiden sich die die Zeitkonstanten, wird die Regelung deutlich schwieriger.

PT2-Regelstrecken

Bei einer PT2-Regelstrecke ändert sich der Istwert bei einem Ausgangswertsprung zunächst nicht, steigt dann mit zunehmender Steigung an und strebt danach mit abnehmender Steigung dem Sollwert zu. Die Regelstrecke zeigt ein proportionales Übertragungsverhalten mit Verzögerung 2. Ordnung.

Beispiele:

- Druckregelung
- Durchflussregelung
- Temperaturregelung

Regelstrecken ohne Ausgleich

Regelstrecken ohne Ausgleich zeigen integrales Verhalten. Der Istwert strebt einen unendlich großen Wert an.

Beispiel:

- Zufluss einer Flüssigkeit in einen Behälter

Regelstrecken mit Totzeit

Eine Totzeit ist immer die Lauf- oder Transportzeit, bis eine Änderung am Systemeingang am Systemausgang gemessen werden kann.

Bei Regelstrecken mit Totzeit ändert sich der Istwert um die Totzeit verspätet.

Beispiel:

Förderband

3.1.3 Kennwerte der Regelstrecke

Ermitteln des Zeitverhaltens aus der Sprungantwort

Das Zeitverhalten der Regelstrecke kann durch den zeitlichen Verlauf des Istwerts x nach sprunghafter Änderung des Ausgangswerts y ermittelt werden. Die meisten Regelstrecken sind Regelstrecken mit Ausgleich.

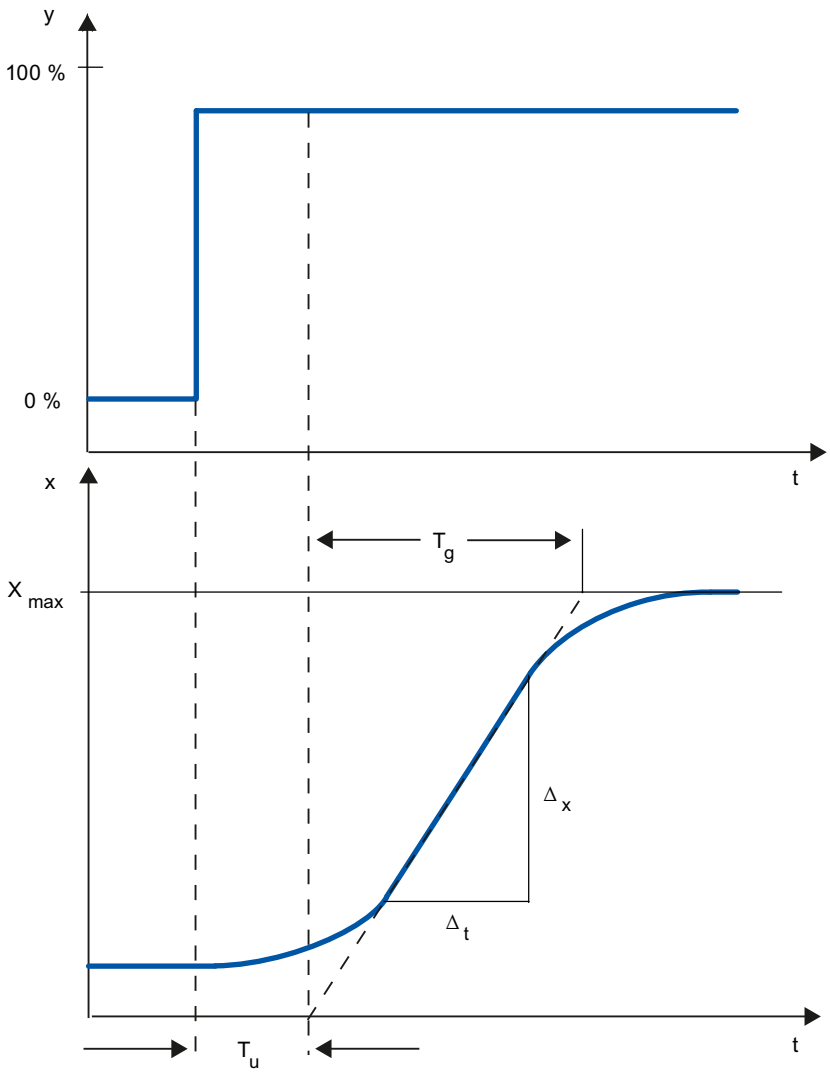

Das Zeitverhalten lässt sich näherungsweise durch die Größen Verzugszeit T_u, Ausgleichszeit T_g und Maximalwert X_{max} festlegen. Die Größen werden durch Anlegen von Tangenten an den Maximalwert und an den Wendepunkt der Sprungantwort ermittelt. Die Aufnahme der Übergangsfunktion bis zum Maximalwert ist in vielen Fällen nicht möglich, weil der Istwert bestimmte Werte nicht überschreiten darf. Dann wird die Anstiegsgeschwindigkeit v_{max} zur Identifikation der Regelstrecke verwendet ($v_{max} = \Delta_x/\Delta_t$).

Die Regelbarkeit der Regelstrecke lässt sich abschätzen aus dem Verhältnis T_u/T_g bzw. $T_u \times v_{max}/X_{max}$ Es gilt:

Streckentyp	T_u / T_g	Regelbarkeit der Regelstrecke
I	< 0,1	gut regelbar
II	0,1 bis 0,3	noch regelbar
III	> 0,3	schwer regelbar

Einfluss der Totzeit auf die Regelbarkeit einer Regelstrecke

Eine Regelstrecke mit Totzeit und Ausgleich zeigt folgendes Verhalten auf einen Sprung des Ausgangswerts.

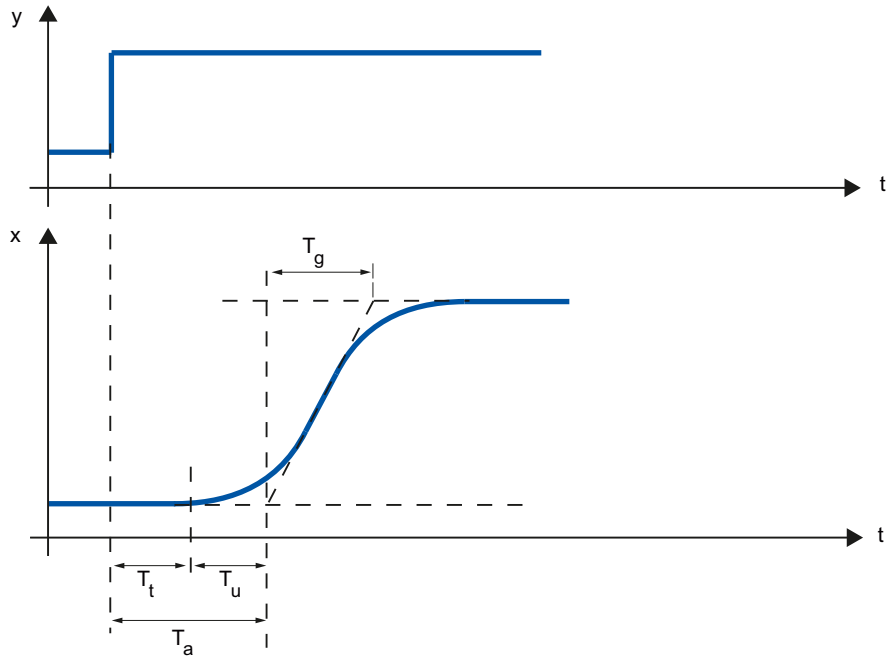

T_t	Totzeit
T_u	Verzugszeit
T_g	Ausgleichszeit
y	Ausgangswert
x	Istwert

Die Regelbarkeit einer Regelstrecke mit Totzeit und Ausgleich wird durch das Verhältnis von T_t zu T_g bestimmt. T_t muss klein sein gegenüber T_g. Es muss gelten:

$T_t/T_g \leq 1$

Reaktionsgeschwindigkeit von Regelstrecken

Nach folgenden Werten lassen sich Regelstrecken beurteilen:

T_u < 0,5 min, T_g < 5 min = schnelle Regelstrecke

T_u > 0,5 min, T_g > 5 min = langsame Regelstrecke

Kennwerte einiger Regelstrecken

Physikalische Größe	Regelstrecke	Verzugszeit T_u	Ausgleichszeit T_g	Anstiegsgeschwindigkeit v_{max}
Temperatur	kleiner, elektrisch beheizter Ofen	0,5 bis 1 min	5 bis 15 min	bis 60 K/min
	großer, elektrisch beheizter Glühofen	1 bis 5 min	10 bis 20 min	bis 20 K/min
	großer, gasbeheizter Glühofen	0,2 bis 5 min	3 bis 60 min	1 bis 30 K/min
	Destillationskolonne	1 bis 7 min	40 bis 60 min	0,1 bis 0,5 °C/s
	Autoklaven (2,5 m^3)	0,5 bis 0,7 min	10 bis 20 min	keine Angabe
	Hochdruck-Autoklaven	12 bis 15 min	200 bis 300 min	keine Angabe
	Dampfüberhitzer	30 s bis 2,5 min	1 bis 4 min	2 °C/s
	Spritzgießmaschinen	0,5 bis 3 min	3 bis 30 min	5 bis 20 K/min
	Extruder	1 bis 6 min	5 bis 60 min	
	Verpackungsmaschinen	0,5 bis 4 min	3 bis 40 min	2 bis 35 K/min
	Raumbeheizung	1 bis 5 min	10 bis 60 min	1 °C/min
Durchfluss	Rohrleitung mit Gas	0 bis 5 s	0,2 bis 10 s	nicht relevant
	Rohrleitung mit Flüssigkeit	keine	keine	
Druck	Gasrohrleitung	keine	0,1 s	nicht relevant
	Trommelkessel mit Gas– oder Ölfeuerung	keine	150 s	nicht relevant
	Trommelkessel mit Schlägermühlen	1 bis 2 min	2 bis 5 min	nicht relevant
Behälterstand (Niveau)	Trommelkessel	0,6 bis 1 min	keine Angabe	0,1 bis 0,3 cm/s
Drehzahl	Kleiner elektrischer Antrieb	keine	0,2 bis 10 s	nicht relevant
	Großer elektrischer Antrieb	keine	5 bis 40 s	nicht relevant
	Dampfturbine	keine	keine Angabe	50 min^{-1}
Elektrische Spannung	Kleine Generatoren	keine	1 bis 5 s	nicht relevant
	Große Generatoren	keine	5 bis 10 s	nicht relevant

3.1.4 Impulsregler

Zweipunktregler ohne Rückführung

Zweipunktregler haben als Schaltfunktion den Zustand "EIN" und "AUS". Dies entspricht 100 % bzw. 0 % Leistung. Durch dieses Verhalten tritt eine Dauerschwingung des Istwerts x um den Sollwert w auf.

Amplitude und Periodendauer der Schwingung wachsen mit dem Verhältnis von Verzugszeit T_u zur Ausgleichszeit T_g der Regelstrecke. Diese Regler werden hauptsächlich für einfache Temperaturregelungen (z. B. für elektrisch direktbeheizte Öfen) oder als Grenzwertmelder eingesetzt.

Die folgende Grafik zeigt die Kennlinie eines Zweipunktreglers

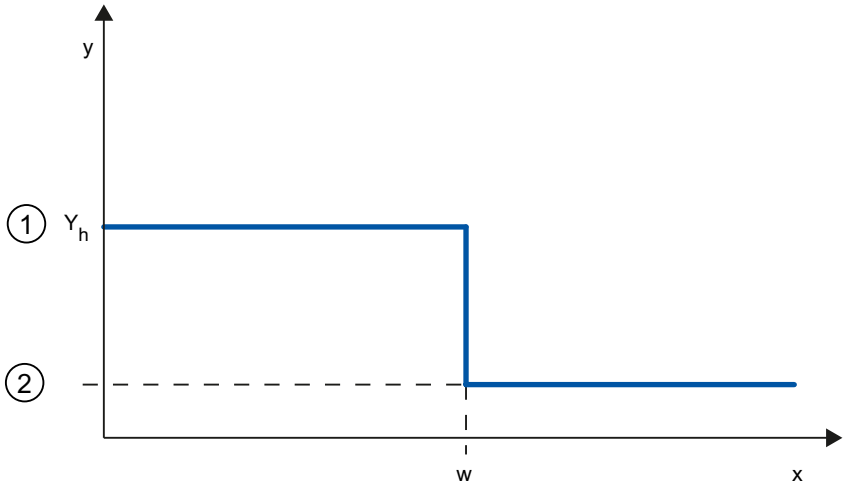

①	EIN
②	AUS
Y_h	Stellbereich
w	Sollwert

Die folgende Grafik zeigt die Regelfunktion eines Zweipunktreglers

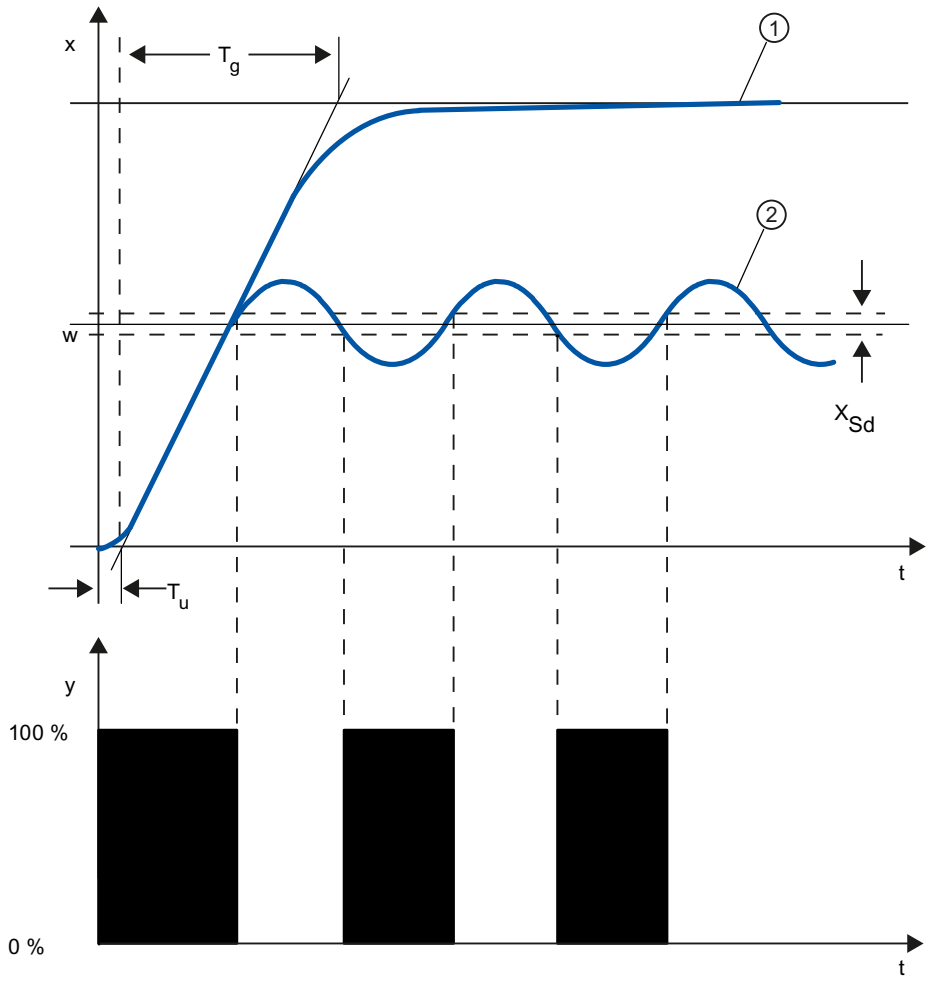

①	Übergangsfunktion ohne Regler
②	Übergangsfunktion mit Zweipunktregler
T_u	Verzugszeit
T_g	Ausgleichszeit
X_{Sd}	Schaltdifferenz

Zweipunktregler mit Rückführung

Das Verhalten von Zweipunktreglern bei Regelstrecken mit größeren Verzugszeiten, z. B. Öfen, bei denen der Nutzraum von der Heizung getrennt ist, lässt sich durch elektronische Rückführungen verbessern.

Durch die Rückführung wird die Schaltfrequenz des Reglers erhöht, wodurch sich die Amplitude des Istwerts verkleinert. Außerdem können die Regelergebnisse im dynamischen Betrieb wesentlich verbessert werden. Die Grenze für die Schaltfrequenz ist durch die Ausgangsstufe gesetzt. Sie sollte bei mechanischen Stellgliedern wie Relais und Schützen 1 bis 5 Schaltungen pro Minute nicht überschreiten. Bei binären Spannungs- und Stromausgängen mit nachgeschalteten Thyristor- oder Triacstellern können hohe Schaltfrequenzen gewählt werden, die weit über der Grenzfrequenz der Regelstrecke liegen.

Da die Schaltimpulse am Ausgang der Regelstrecke nicht mehr feststellbar sind, erhält man vergleichbare Ergebnisse wie mit kontinuierlichen Reglern.

Der Ausgangswert wird durch Pulsweitenmodulation des Ausgangswerts eines kontinuierlichen Reglers erzeugt

Zweipunktregler mit Rückführung werden zur Temperaturregelung in Öfen, an Verarbeitungsmaschinen der Kunststoff-, Textil-, Papier-, Gummi- und Lebensmittelindustrie sowie für Heiz- und Kühlgeräte eingesetzt.

Dreipunktregler

Dreipunktregler werden für Heizen/Kühlen eingesetzt. Diese Regler haben als Ausgang zwei Schaltpunkte. Durch elektronische Rückführstrukturen werden die Regelergebnisse optimiert. Einsatzgebiete für derartige Regler sind Wärme-, Kälte-, Klimakammern und Werkzeugbeheizungen für kunststoffverarbeitende Maschinen.

Die folgende Grafik zeigt die Kennlinie eines Dreipunktreglers

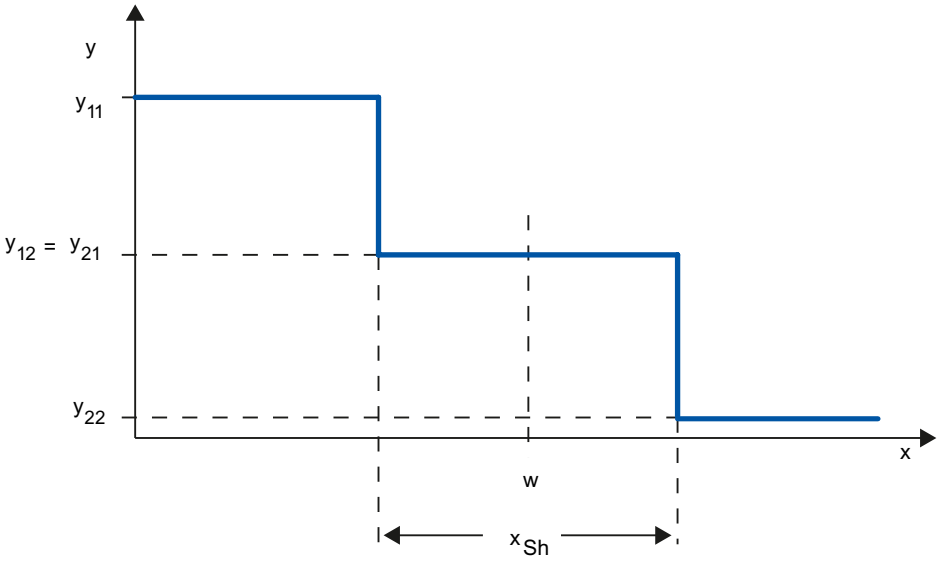

y	Ausgangswert, z. B. y11 = 100 % Heizung y12 = 0 % Heizung y21 = 0 % Kühlung y22 = 100 % Kühlung
x	Physikalische Größe des Istwerts, z. B. Temperatur in °C
w	Sollwert
x_{Sh}	Abstand zwischen Schaltpunkt 1 und Schaltpunkt 2

3.1.5 Führungs- und Störverhalten

Führungsverhalten

Der Istwert soll einer Änderung des Sollwerts möglichst schnell folgen. Das Führungsverhalten ist umso besser, je kürzer die Zeit, in der der neue Sollwert erreicht wird und je geringer die Schwankung des Istwerts ist.

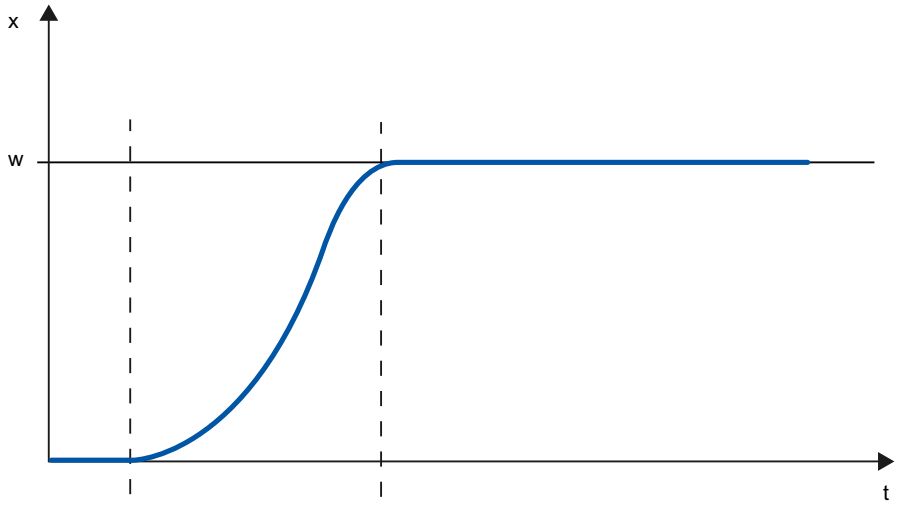

x Istwert
w Sollwert

Störverhalten

Der Sollwert wird durch Störgrößen beeinflusst. Der Regler muss die dadurch auftretende Regeldifferenz möglichst schnell beseitigen. Das Störverhalten ist umso besser, je kürzer die Zeit, in der der Sollwert wieder erreicht wird und je geringer die Schwankung des Istwerts.

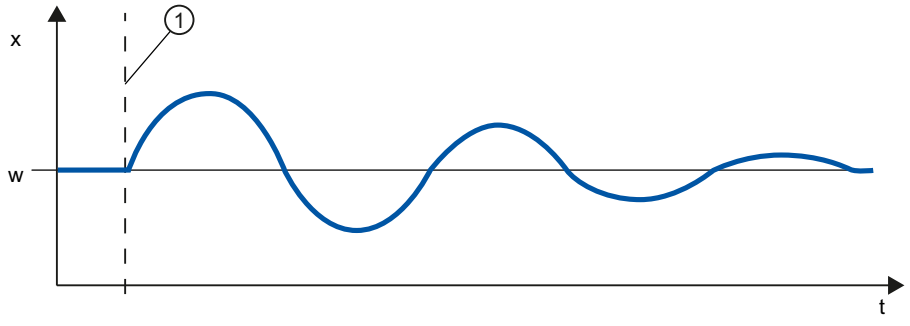

x Istwert
w Sollwert
① Einwirken einer Störgröße

Eine Störgröße wird durch Regler mit I–Anteil ausgeglichen. Die Qualität der Regelung wird durch eine konstant wirkende Störgröße nicht verringert, da die Regeldifferenz relativ konstant ist. Eine dynamische Störgröße beeinflusst die Qualität der Regelung stärker, da die

Regeldifferenz schwankt. Erst über den sich langsam aufbauenden I-Anteil wird die Regeldifferenz wieder abgebaut.

Eine messbare Störgröße kann in den Regelkreis einfließen. Die Reaktion des Reglers kann dadurch erheblich beschleunigt werden.

3.1.6 Regelverhalten bei unterschiedlichen Rückführstrukturen

Regler-Verhaltensweisen

Je genauer der Regler an das Zeitverhalten der Regelstrecke angepasst ist, desto genauer stellt er den Sollwert ein und reagiert optimal auf Störgrößen.

Die Rückführschaltung kann Proportional-Verhalten (P), Proportional-Differential-Verhalten (PD), Proportional-Integral-Verhalten (PI) oder Proportional-Integral-Differential-Verhalten (PID) haben.

Wenn eine Sprungfunktion auf die Regeldifferenz gegeben wird, entstehen je nach Regler unterschiedliche Sprungantworten.

Sprungantwort eines P-Reglers

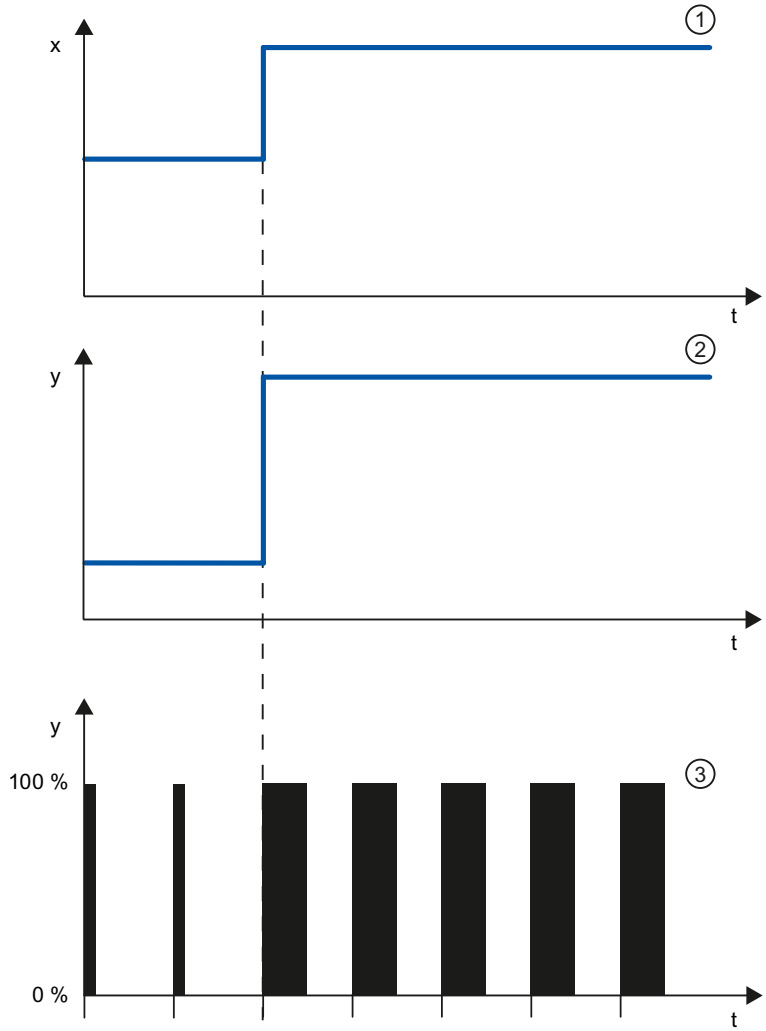

① Regeldifferenz
② Ausgangswert eines kontinuierlichen Reglers
③ Ausgangswert eines Impulsreglers

Formel für P-Regler

Ausgangswert und Regeldifferenz sind direkt proportional, d. h.:

Ausgangswert = Proportionalbeiwert × Regeldifferenz

y = GAIN × x

Sprungantwort eines PD-Reglers

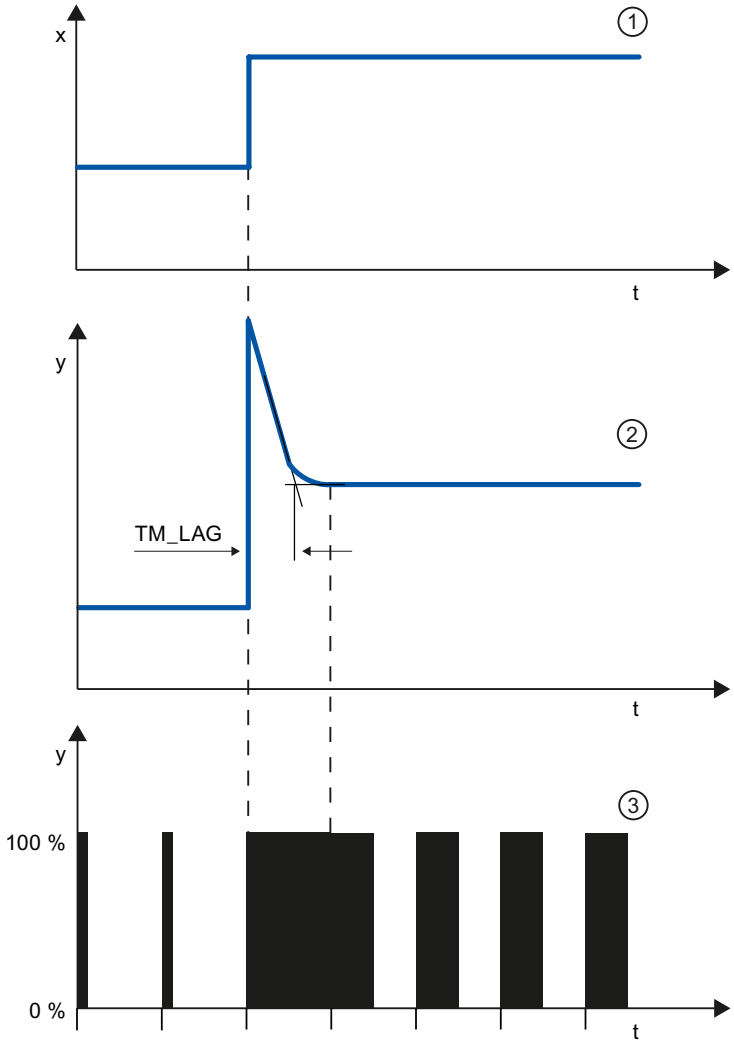

①	Regeldifferenz
②	Ausgangswert eines kontinuierlichen Reglers
③	Ausgangswert eines Impulsreglers
TM_LAG	Verzögerung des D-Anteils

Formel für PD-Regler

Für die Sprungantwort des PD-Reglers im Zeitbereich gilt:

$$y = GAIN \cdot X_W \cdot \left(1 + \frac{TD}{TM_LAG} \cdot e^{\frac{-t}{TM_LAG}} \right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

Der D-Anteil erzeugt einen Ausgangswert in Abhängigkeit von der Geschwindigkeit, mit der sich der Istwert ändert. Ein reiner D-Anteil ist nicht zur Regelung geeignet, da nur eine

Änderung des Istwerts eine Änderung des Ausgangswerts bewirkt. Wenn der Istwert konstant bleibt, ändert sich auch der Ausgangswert nicht mehr.

In Verbindung mit einem P-Anteil verbessert der D-Anteil das Störverhalten. Störungen werden nicht vollkommen ausgeregelt. Vorteilhaft ist das gute dynamische Verhalten. Beim Anfahren und bei der Sollwertänderung wird ein gut gedämpfter, schwingungsfreier Übergang erreicht.

Ein Regler mit D-Anteil ist nicht angebracht, wenn eine Regelstrecke pulsierende Messgrößen hat, z. B. bei Druck- oder Durchflussregelungen.

Sprungantwort eines PI-Reglers

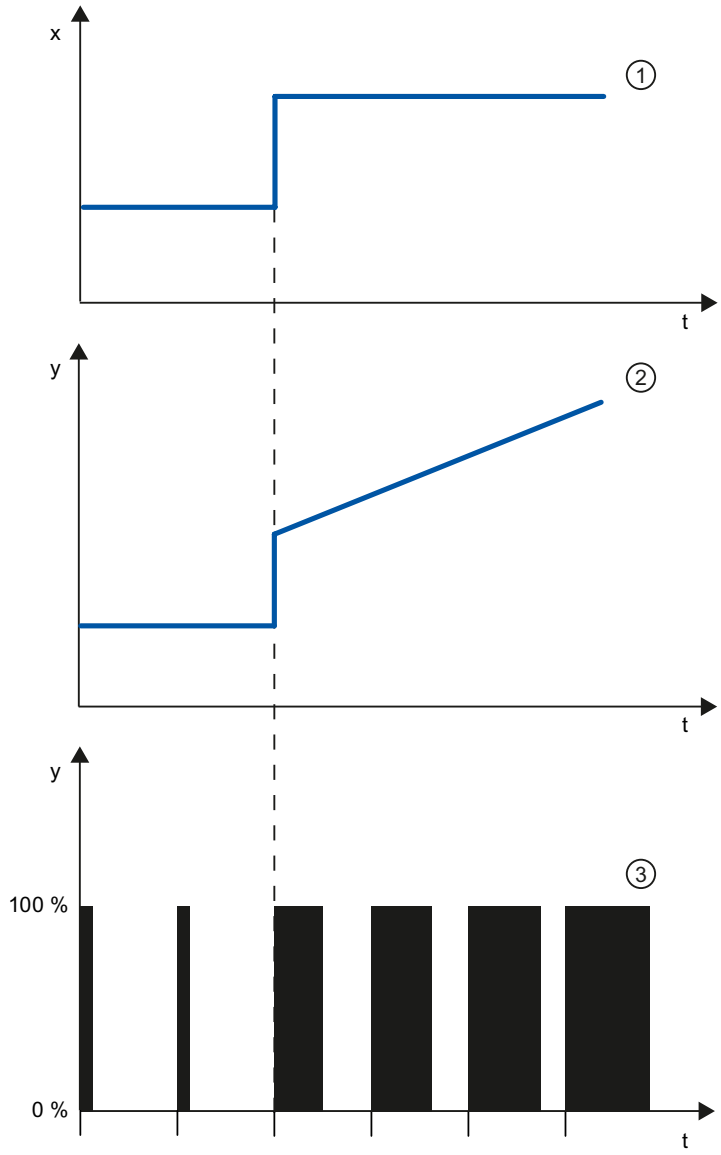

① Regeldifferenz
② Ausgangswert eines kontinuierlichen Reglers
③ Ausgangswert eines Impulsreglers

Ein I-Anteil im Regler summiert die Regeldifferenz über die Zeit. Dadurch wird der Regler so lange nachstellen, bis die Regeldifferenz aufgehoben ist. Bei einem reinen P-Regler tritt eine dauerhafte Regeldifferenz auf. Diese kann durch einen I-Anteil im Regler behoben werden.

In der Praxis ist, je nach Anforderung an das Regelverhalten, eine Kombination aus P-Anteil, I-Anteil und D-Anteil ideal. Das Zeitverhalten der einzelnen Anteile lässt sich durch die Reglerparameter Proportionalbeiwert GAIN, Nachstellzeit TI (I-Anteil) und Vorhaltezeit TD (D-Anteil) beschreiben.

Formel für PI-Regler

Für die Sprungantwort des PI-Reglers im Zeitbereich gilt:

$$y = GAIN \cdot X_W \cdot \left(1 + \frac{1}{TI \cdot t}\right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

Sprungantwort eines PID-Reglers

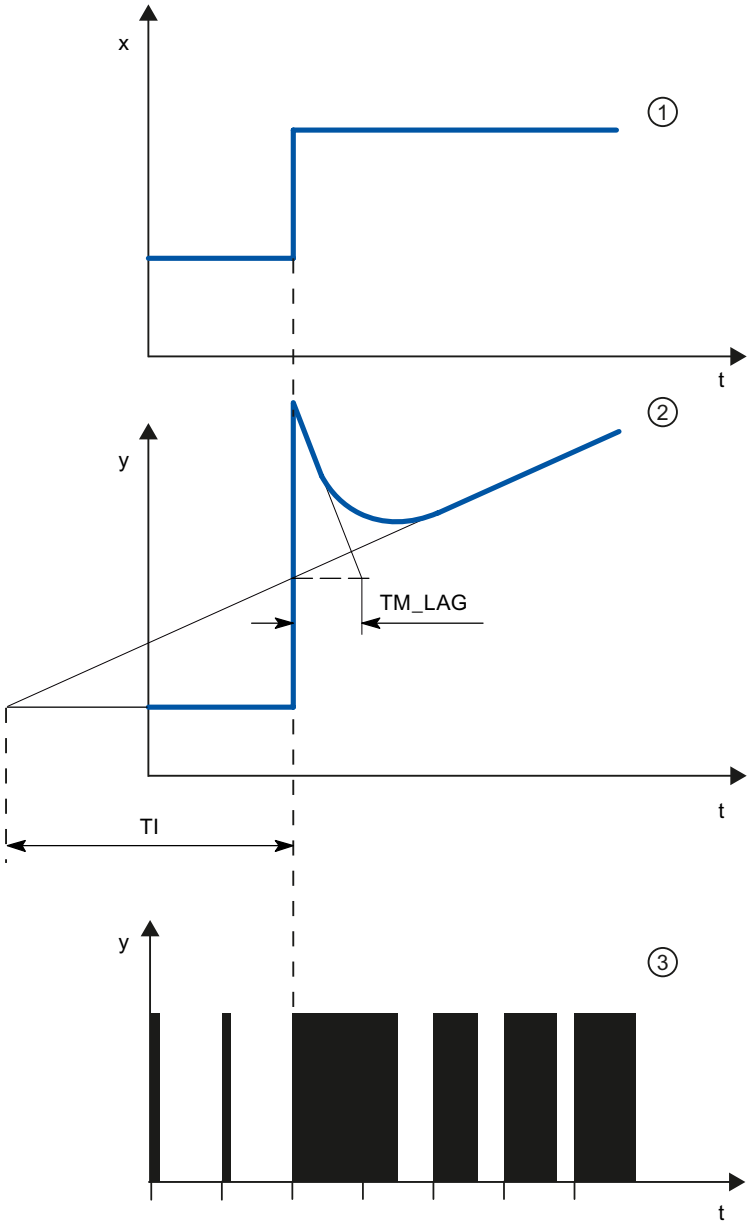

①	Regeldifferenz
②	Ausgangswert eines kontinuierlichen Reglers
③	Ausgangswert eines Impulsreglers
TM_LAG	Verzögerung des D-Anteils
T_i	Integrationszeit

Formel für PID-Regler

Für die Sprungantwort des PID-Reglers im Zeitbereich gilt:

$$y = GAIN \cdot X_w \cdot \left(1 + \frac{1}{TI \cdot t} + \frac{TD}{TM_LAG} \cdot e^{\frac{-t}{TM_LAG}} \right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

Verhalten einer Regelstrecke bei verschiedenen Reglerstrukturen

Die meisten in der Verfahrenstechnik vorkommenden Regelungen lassen sich mit einem Regler mit PI-Verhalten beherrschen. Bei langsamen Regelstrecken mit großer Verzugszeit, z. B. Temperaturregelungen, lässt sich das Regelergebnis durch einen Regler mit PID-Verhalten verbessern.

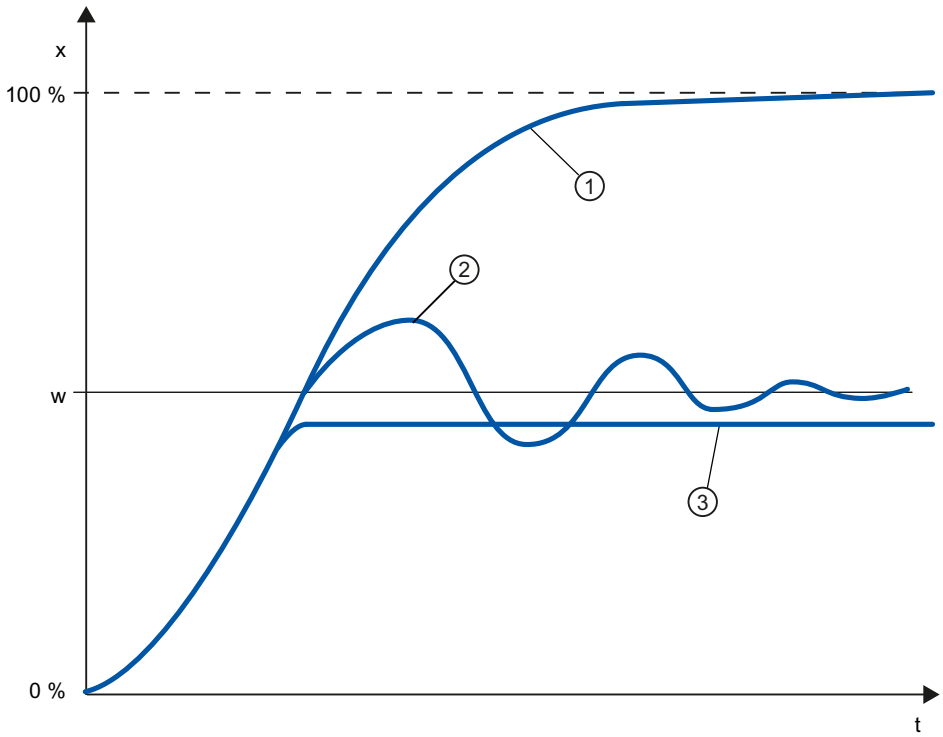

①	Kein Regler
②	PID-Regler
③	PD-Regler
w	Sollwert
x	Istwert

Regler mit PI- und PID-Verhalten haben den Vorteil, dass nach dem Einschwingen der Istwert keine Abweichung zum Sollwert zeigt. Der Istwert schwingt beim Anfahren über den Sollwert.

3.1.7 Wahl der Reglerstruktur bei gegebener Regelstrecke

Auswahl der geeigneten Reglerstrukturen

Sie erhalten nur dann ein optimales Regelergebnis, wenn Sie eine Reglerstruktur wählen, die zur Regelstrecke passt und sich innerhalb bestimmter Grenzen an die Regelstrecke anpassen lässt.

Die folgende Tabelle gibt einen Überblick, welche Reglerstruktur für welche Regelstrecke geeignet ist.

Regelstrecke		Reglerstruktur			
		P	PD	PI	PID
	nur mit Totzeit	ungeeignet	ungeeignet	geeignet	ungeeignet
	PT1 mit Totzeit	ungeeignet	ungeeignet	gut geeignet	gut geeignet
	PT2 mit Totzeit	ungeeignet	bedingt geeignet	gut geeignet	gut geeignet
	höhere Ordnung	ungeeignet	ungeeignet	bedingt geeignet	gut geeignet
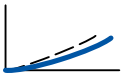	ohne Ausgleich	gut geeignet	gut geeignet	gut geeignet	gut geeignet

Die folgende Tabelle gibt einen Überblick, welche Reglerstruktur für welche physikalische Größe geeignet ist.

Physikalische Größe	Reglerstruktur			
	P	PD	PI	PID
	bleibende Regeldifferenz		keine bleibende Regeldifferenz	
Temperatur	für geringe Ansprüche und bei P-Strecken mit T_u/Tg < 0,1	gut geeignet	für gehobene Ansprüche die am besten geeigneten Reglerstrukturen (von besonders angepassten Spezialreglern abgesehen)	
Druck	geeignet, wenn keine nennenswerte Verzugszeit	ungeeignet	für gehobene Ansprüche die am besten geeigneten Reglerstrukturen (von besonders angepassten Spezialreglern abgesehen)	
Durchfluss	ungeeignet, weil erforderlicher GAIN-Bereich meist zu groß	ungeeignet	brauchbar, aber I-Regler allein oft besser	kaum erforderlich

3.1.8 Einstellung der PID-Parameter

Faustformel für die Parametereinstellung

Reglerstruktur	Einstellung
P	GAIN ≈ v_{max} × T_u [°C]
PI	GAIN ≈ 1,2 × v_{max} × T_u [°C] TI ≈ 4 × T_u [min]
PD	GAIN ≈ 0,83 × v_{max} × T_u [°C] TD ≈ 0,25 × v_{max} × T_u [min] TM_LAG ≈ 0,5 × TD[min]
PID	GAIN ≈ 0,83 × v_{max} × T_u [°C] TI ≈ 2 × T_u [min] TD ≈ 0,4 × T_u [min] TM_LAG ≈ 0,5 × TD[min]
PD/PID	GAIN ≈ 0,4 × v_{max} × T_u [°C] TI ≈ 2 × T_u [min] TD ≈ 0,4 × T_u [min] TM_LAG ≈ 0,5 × TD[min]

Statt $v_{max} = \Delta_x / \Delta_t$ kann X_{max} / T_g eingesetzt werden.

Bei Reglern mit PID-Struktur ist die Einstellung von Nachstellzeit und Vorhaltezeit meist miteinander gekoppelt.

Das Verhältnis TI / TD liegt zwischen 4 und 5 und ist für die meisten Regelstrecken optimal.

Das Nichteinhalten der Vorhaltezeit TD ist bei PD-Reglern unkritisch.

Bei PI- oder PID-Reglern treten Regelschwingungen auf, wenn die Nachstellzeit TI um mehr als die Hälfte zu klein gewählt wurde.

Eine zu große Nachstellzeit verlangsamt das Ausregeln von Störungen. Man kann nicht erwarten, dass die Regelkreise nach den ersten Parametereinstellungen "optimal" arbeiten. Erfahrungsgemäß ist ein Nachjustieren immer dann erforderlich, wenn eine "schwer regelbare" Strecke mit $T_u / T_g > 0,3$ vorliegt.

3.2 Projektieren eines Software-Reglers

3.2.1 Übersicht der Software-Regler

Für die Projektierung eines Software-Reglers benötigen Sie eine Anweisung mit dem Regelalgorithmus und ein Technologieobjekt. DasTechnologieobjekt für einen Software-Regler entspricht dem Instanz-DB der Anweisung. Im Technologieobjekt wird die Konfiguration des Reglers gespeichert. Im Unterschied zu den Instanz-DBs anderer Anweisungen, werden Technologieobjekte nicht bei den Programmresourcen ablegt, sondern unter CPU > Technologieobjekte.

Technologieobjekte und Anweisungen

CPU	Bibliothek	Anweisung	Technologieobjekt	Beschreibung
S7-1200	Compact PID	PID_Compact V1.x	PID_Compact V1.x	Universeller PID-Regler mit integrierter Optimierung
S7-1200		PID_3Step V1.x	PID_3Step V1.x	PID-Regler für Ventile mit integrierter Optimierung
S7-1500 S7-1200 V4.x		PID_Compact V2.x	PID_Compact V2.x	Universeller PID-Regler mit integrierter Optimierung
S7-1500 S7-1200 V4.x		PID_3Step V2.x	PID_3Step V2.x	PID-Regler für Ventile mit integrierter Optimierung
S7-1500 ≥ V1.7 S7-1200 ≥ V4.1		PID_Temp V1.x	PID_Temp V1.x	Universeller PID-Temperaturregler mit integrierter Optimierung
S7-1500/300/400	PID Basisfunktionen	CONT_C	CONT_C	Kontinuierlicher Regler
S7-1500/300/400		CONT_S	CONT_S	Schrittregler für integrierende Stellglieder
S7-1500/300/400		PULSEGEN	-	Pulsgenerator für proportional wirkende Stellglieder
S7-1500/300/400		TCONT_CP	TCONT_CP	Kontinuierlicher Temperaturregler mit Pulsgenerator
S7-1500/300/400		TCONT_S	TCONT_S	Temperaturregler für integrierende Stellglieder
S7-300/400	PID Self-Tuner	TUN_EC	TUN_EC	Optimierung eines kontinuierlichen Reglers
S7-300/400		TUN_ES	TUN_ES	Optimierung eines Schrittreglers
S7-300/400	Standard PID Control (Optionspaket PID Professional)	PID_CP	PID_CP	Kontinuierlicher Regler mit Pulsegenerator
S7-300/400		PID_ES	PID_ES	Schrittregler für integrierende Stellglieder
S7-300/400		LP_SCHED	-	Regleraufrufe verteilen

CPU	Bibliothek	Anweisung	Technologieobjekt	Beschreibung
S7-300/400	Modular PID Control (Optionspaket PID Professional)	A_DEAD_B	-	Störsignale aus Regeldifferenz filtern
S7-300/400		CRP_IN	-	Analoges Eingangssignal skalieren
S7-300/400		CRP_OUT	-	Analoges Ausgangssignal skalieren
S7-300/400		DEAD_T	-	Eingangssignal verzögert ausgeben
S7-300/400		DEADBAND	-	Kleine Schwankungen des Istwerts unterdrücken
S7-300/400		DIF	-	Eingangssignal über die Zeit differenzieren
S7-300/400		ERR_MON		Regeldifferenz überwachen
S7-300/400		INTEG	-	Eingangssignal über die Zeit integrieren
S7-300/400		LAG1ST	-	Verzögerungsglied 1. Ordnung
S7-300/400		LAG2ND	-	Verzögerungsglied 2. Ordnung
S7-300/400		LIMALARM	-	Grenzwerte melden
S7-300/400		LIMITER	-	Stellwert begrenzen
S7-300/400		LMNGEN_C	-	Stellwert für kontinuierlichen Regler ermitteln
S7-300/400		LMNGEN_S	-	Stellwert für Schrittregler ermitteln
S7-300/400		NONLIN	-	Gebersignal linearisieren
S7-300/400		NORM	-	Istwert physikalisch normieren
S7-300/400		OVERRIDE	-	Stellwerte von 2 PID-Reglern auf 1 Stellglied schalten
S7-300/400		PARA_CTL	-	Parametersätze umschalten
S7-300/400		PID	-	PID Algorithmus
S7-300/400		PUSLEGEN_M	-	Pulse für proportional wirkende Stellglieder generieren
S7-300/400		RMP_SOAK	-	Sollwerte nach Zeitplan vorgeben
S7-300/400		ROC_LIM	-	Änderungsgeschwindigkeit begrenzen
S7-300/400		SCALE_M	-	Istwert skalieren
S7-300/400		SP_GEN	-	Sollwert manuell vorgeben
S7-300/400		SPLT_RAN	-	Stellwertbereiche aufteilen
S7-300/400		SWITCH	-	Analogwerte schalten
S7-300/400		LP_SCHED_M	-	Regleraufrufe verteilen

3.2.2 Schritte für das Projektieren eines Software-Reglers

Alle SW-Regler projektieren Sie nach demselben Schema:

Schritt	Beschreibung
1	Technologieobjekt hinzufügen (Seite 653)
2	Technologieobjekt konfigurieren (Seite 654)
3	Anweisung im Anwenderprogramm aufrufen (Seite 655)
4	Technologieobjekt in Gerät laden (Seite 656)
5	Software-Regler in Betrieb nehmen (Seite 657)
6	Optimierte PID-Parameter im Projekt speichern (Seite 658)

Schritt	Beschreibung
7	Werte vergleichen (Seite 659)
8	Instanzen eines Technologieobjektes anzeigen (Seite 680)

3.2.3 Technologieobjekte hinzufügen

Technologieobjekt im Projektnavigator hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Instanz-DB der Anweisung zu diesem Technologieobjekt erzeugt. In diesem Instanz-DB wird die Konfiguration des Technologieobjekts hinterlegt.

Voraussetzung

Ein Projekt mit einer CPU ist angelegt.

Vorgehen

Um ein Technologieobjekt hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "PID-Regler".
   Alle verfügbaren PID-Regler für diese CPU werden angezeigt.
5. Wählen Sie die Anweisung für das Technologieobjekt, z. B. PID_Compact.
6. Geben Sie im Eingabefeld "Name" einen individuellen Namen für das Technologieobjekt ein.
7. Wählen Sie die Option "manuell", falls Sie die vorgeschlagene Datenbausteinnummer des Instanz-DB ändern möchten.
8. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt hinterlegen möchten.
9. Bestätigen Sie mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt. Das Technologieobjekt wird verwendet, wenn die Anweisung zu diesem Technologieobjekt in einem Weckalarm-OB aufgerufen wird.

Hinweis

Sie können im unteren Bereich des Dialogfelds das Optionskästchen "Neu hinzufügen und öffnen" aktivieren. Dadurch wird die Konfiguration des Technologieobjekts nach dem Hinzufügen geöffnet.

3.2.4 Technologieobjekte konfigurieren

Die Eigenschaften eines Technologieobjekts auf einer S7-1200 CPU können Sie auf zwei Arten konfigurieren.

- Im Inspektorfenster des Programmiereditors
- Im Konfigurationseditor

Die Eigenschaften eines Technologieobjekts auf einer S7-300/400 CPU können Sie nur im Konfigurationseditor konfigurieren.

Inspektorfenster des Programmiereditors

Im Inspektorfenster des Programmiereditors können Sie nur die Parameter konfigurieren, die für den Betrieb notwendig sind.

Auch im Online-Modus werden die Offline-Werte der Parameter angezeigt. Sie können die Online-Werte nur in Fenster Inbetriebnahme ändern.

Um das Inspektorfenster eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Doppelklicken Sie den Baustein (Weckalarm-OB), in welchem Sie die Anweisung des SW-Reglers aufrufen.
   Der Baustein wird im Arbeitsbereich geöffnet.
3. Klicken Sie auf die Anweisung des SW-Reglers.
4. Wählen Sie im Inspektorfenster nacheinander die Register "Eigenschaften" und "Konfiguration".

Konfigurationsfenster

Für jedes Technologieobjekt gibt es ein spezifisches Konfigurationsfenster, in dem Sie alle Eigenschaften konfigurieren.

Um das Konfigurationsfenster eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Konfiguration".

Symbole

Symbole in der Bereichsnavigation des Konfigurations- und Inspektorfensters zeigen weitere Details zur Vollständigkeit der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich Voreinstellungswerte. Mit diesen Voreinstellungswerten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein Voreinstellungswert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste enthält keinen oder einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.

Die Eigenschaften eines Technologieobjekts sind detailliert im Kapitel für das Technologieobjekt beschrieben.

3.2.5 Anweisung im Anwenderprogramm aufrufen

Die Anweisung des Software-Reglers muss in einem Weckalarm-OB aufgerufen werden. Die Abtastzeit des Software-Reglers wird durch den zeitlichen Abstand der Aufrufe im Weckalarm-OB bestimmt.

Voraussetzung

Der Weckalarm-OB ist angelegt und die Zykluszeit des Weckalarm-OBs ist korrekt konfiguriert.

Vorgehen

Um die Anweisung im Anwenderprogramm aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Programmbausteine".
3. Doppelklicken Sie den Weckalarm-OB.
   Der Baustein wird im Arbeitsbereich geöffnet.

4. Öffnen Sie im Fenster "Anweisungen" die Gruppe "Technologie" und den Ordner "PID Control".
   Der Ordner enthält alle Anweisungen für Software-Regler, die auf der CPU projektiert werden können.
5. Wählen Sie eine Anweisung und ziehen Sie diese per Drag & Drop in Ihren Weckalarm-OB. Der Dialog "Aufrufoptionen" wird geöffnet.
6. Wählen Sie aus der Liste "Name" ein Technologieobjekt oder geben Sie den Namen für ein neues Technologieobjekt ein.

Ergebnis

Wenn das Technologieobjekt noch nicht besteht, wird es hinzugefügt. Die Anweisung wird in den Weckalarm-OB eingefügt. Das Technologieobjekt ist diesem Aufruf der Anweisung zugeordnet.

3.2.6 Technologieobjekte in Gerät laden

Eine neue oder geänderte Konfiguration des Technologieobjekts muss für den Onlinebetrieb in die CPU geladen werden. Beim Laden remanenter Daten gelten folgende Besonderheiten:

- **Software (nur Änderungen)**
  - S7-1200, S7-1500:
    Remanente Daten bleiben erhalten.
  - S7-300/400:
    Remanente Daten werden sofort aktualisiert. Die CPU geht nicht in Stop.
- **PLC-Programm in Gerät laden und zurücksetzen**
  - S7-1200, S7-1500:
    Remanente Daten werden beim nächsten Übergang von Stop in RUN aktualisiert. Das PLC-Programm kann nur vollständig geladen werden.
  - S7-300/400:
    Remanente Daten werden beim nächsten Übergang von Stop in RUN aktualisiert.

Remanente Daten auf eine CPU S7-1200 oder S7-1500 laden

Hinweis

Das Laden und zurücksetzen des PLC-Programms bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie das PLC-Programm laden und zurücksetzen!

Um remanente Daten zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eintrag der CPU in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "PLC-Programm in Gerät laden und zurücksetzen".
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden".
   - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen.
   Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
4. Klicken Sie auf "Laden".
   Das vollständige PLC-Programm wird geladen und der Dialog "Ergebnisse laden" geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
5. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

Ergebnis

Das komplette PLC-Programm wird in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

3.2.7 Software-Regler in Betrieb nehmen

Vorgehen

Um den Arbeitsbereich "Inbetriebnahme" eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Inbetriebnahme".

Die Funktionen zur Inbetriebnahme sind für jeden Regler spezifisch und dort beschrieben.

3.2.8 Optimierte PID-Parameter im Projekt speichern

Der Software-Regler wird in der CPU optimiert. Dadurch stimmen die Werte im Instanz-DB auf der CPU nicht mehr mit denen im Projekt überein.

Um die PID-Parameter im Projekt mit den optimierten PID-Parametern in der CPU zu aktualisieren, gehen Sie folgendermaßen vor:

Voraussetzung

- Eine Onlineverbindung zur CPU ist aufgebaut und die CPU befindet sich im Betriebzustand "RUN".
- Die Funktionen des Inbetriebnahmefensters sind mit der Schaltfläche "Start" freigegeben.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Öffnen Sie ein Technologieobjekt.
4. Doppelklicken Sie auf "Inbetriebnahme".
5. Klicken Sie auf das Symbol "PID-Parameter laden".
6. Speichern Sie das Projekt.

Ergebnis

Die aktuell wirkenden PID-Parameter werden in den Projektdaten gesichert. Beim erneuten Laden der Projektdaten in die CPU werden die optimierten Parameter verwendet.

3.2.9 Werte vergleichen

3.2.9.1 Vergleichsanzeige und Randbedingungen

Die Funktion "Werte vergleichen" bietet folgende Möglichkeiten:

- Vergleich der konfigurierten Startwerte des Projekts mit den Startwerten in der CPU und den Aktualwerten
- Direktes Bearbeiten der Aktualwerte und der Startwerte des Projekts
- Sofortiges Erkennen und Anzeigen von Eingabefehlern mit Hilfestellungen zur Korrektur
- Sicherung der Aktualwerte im Projekt
- Übertragen der Startwerte des Projekts in die CPU als Aktualwerte

Symbole und Bedienelemente

Folgende Symbole und Bedienelemente stehen zur Verfügung:

Symbol	Funktion
	Startwert in CPU ist gleich dem konfigurierten Startwert im Projekt
	Startwert in CPU ist ungleich dem konfigurierten Startwert im Projekt
	Der Vergleich Startwert in CPU und konfiguriertem Startwert im Projekt kann nicht durchgeführt werden
	Mindestens einer der beiden Vergleichswerte ist technologisch oder syntaktisch falsch.
	Aktualwerte werden in das Offline-Projekt übertragen
	Aktualisierte Startwerte im Projekt werden in die CPU übertragen (Einstellwerte initialisieren)
	Dialog "Werte vergleichen" wird geöffnet

Randbedingungen

Die Funktion "Werte vergleichen" ist uneingeschränkt verfügbar für S7-1200 und S7-1500.

Für S7-300 und S7-400 gilt folgende Einschränkung:

Im Beobachtungsmodus kann eine S7-300/S7-400 die Startwerte nicht in dIe CPU übertragen. Diese Werte können mit "Werte vergleichen" nicht online angezeigt werden.

Die Aktualwerte des Technologieobjekts werden angezeigt und sind direkt veränderbar.

3.2.9.2 Werte vergleichen

Im Folgenden wird das Vorgehen am Beispiel der "PID-Parameter" gezeigt.

Voraussetzungen

- Ein Projekt mit einem Software-Regler ist konfiguriert.
- Das Projekt ist in die CPU geladen.
- Im Projektnavigator ist der Konfigurationsdialog geöffnet.

Vorgehen

1. Öffnen Sie in der Projektnavigation den gewünschten Software-Regler.
2. Doppelklicken Sie auf das Objekt "Konfiguration".
3. Navigieren Sie im Konfigurationsfenster zum Dialog "PID-Parameter".
4. Klicken Sie auf das Symbol , um den Beobachtungsmodus zu aktivieren.
   Die Symbole und Bedienelemente (Seite 658) der Funktion "Werte vergleichen" werden hinter den Parametern angezeigt.

5. Klicken Sie in das Eingabefeld des gewünschten Parameters und ändern Sie die Parameterwerte manuell durch direkte Eingabe.
   - Wenn der Hintergrund des Eingabefelds grau ist, sind die Werte nur lesbar, nicht änderbar.
   - Um im Dialog "PID-Parameter" die Werte zu ändern, aktivieren Sie vorher die manuelle Eingabe durch Anklicken des Optionskästchens "Manuelle Eingabe aktivieren".
6. Klicken Sie auf das Symbol , um den Dialog der Startwerte zu öffnen.
   Dieser Dialog zeigt zwei Werte des Parameters an:
   - Startwert in CPU: Im oberen Teil wird der Startwert in der CPU angezeigt.
   - Startwert im Projekt: Im unteren Teil wird der konfigurierte Startwert im Projekt angezeigt.
7. Tragen Sie den gewünschten Wert in das Eingabefeld für das Projekt ein.

Fehlererkennung

Die Eingabe von fehlerhaften Werten wird erkannt. In diesem Fall werden Hilfestellungen zur Korrektur angeboten.

Wenn Sie einen syntaktisch falschen Wert eingeben, wird unter dem Parameter ein Rollout mit der entsprechenden Fehlermeldung geöffnet. Der falsche Wert wird nicht übernommen.

Wenn Sie einen technologisch falschen Wert eingeben, wird ein Dialog geöffnet, in dem der Fehler gemeldet und eine Korrekturinformation angezeigt wird:

- Durch Klicken auf "Nein" können Sie die Korrektur annehmen und Ihre Eingabe korrigieren.
- Durch Klicken auf "OK" übernehmen Sie den falschen Wert.

ACHTUNG
Fehlfunktion des Regler Technologisch falsche Werte können zu Fehlfunktionen des Reglers führen.

Aktualwerte sichern

Durch Klicken auf das Symbol übertragen Sie die Aktualwerte des Reglers in die Startwerte Ihres konfigurierten Projekts.

Projektwerte in die CPU übertragen

Durch Klicken auf das Symbol übertragen Sie die konfigurierten Werte Ihres Projekts in die CPU.

> **⚠ VORSICHT**
>
> **Personen- und Sachschäden verhindern!**
>
> Das Laden und Zurücksetzen des Anwenderprogramms bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!
>
> Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie das Anwenderprogramm laden und zurücksetzen!

3.2.10 Parametersicht

3.2.10.1 Einführung in die Parametersicht

Die Parametersicht bietet Ihnen eine Gesamtübersicht über alle relevanten Parameter eines Technologieobjektes. Sie erhalten einen Überblick über die Parametereinstellungen und können diese komfortabel im Offline- und Online-Betrieb ändern.

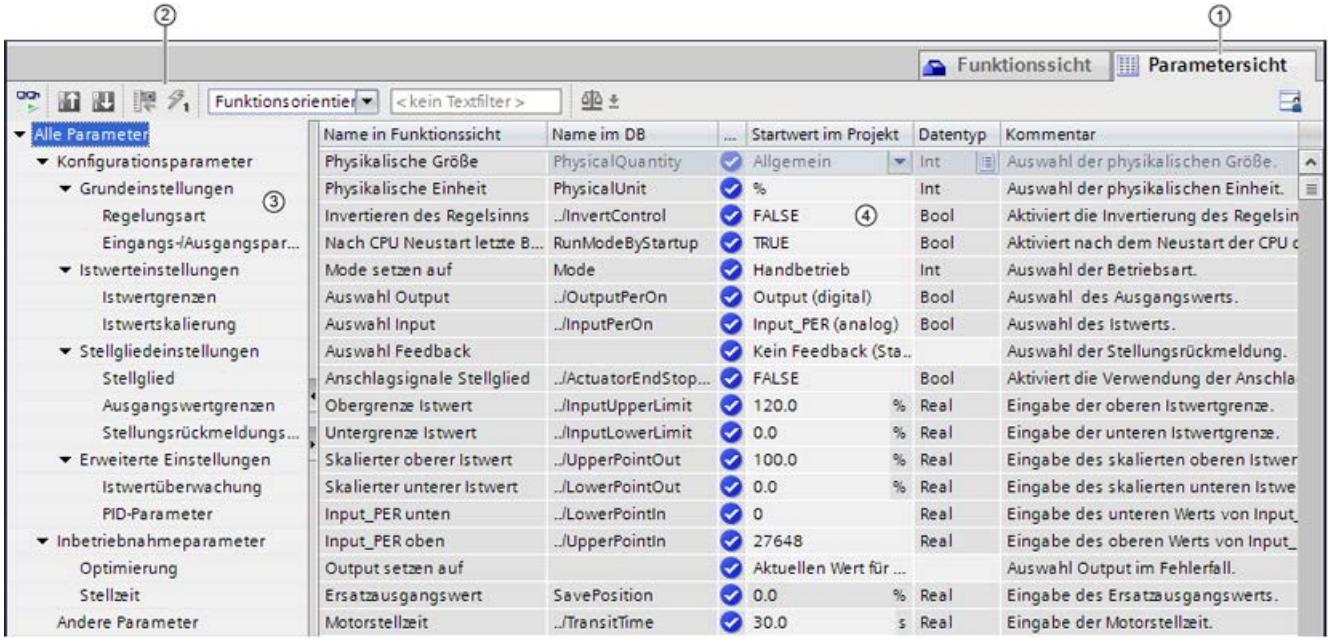

① Register "Parametersicht"

② Funktionsleiste (Seite 663)

③ Navigation (Seite 664)

④ Parametertabelle (Seite 664)

Funktionsumfang

Um die Parameter der Technologieobjekte zu analysieren und gezielt beobachten und steuern zu können, stehen folgende Funktionen zu Verfügung.

Anzeigefunktionen:

- Anzeige der Parameterwerte im Offline- und Online-Betrieb
- Anzeige von Statusinformationen der Parameter
- Anzeige von Werteabweichungen und Möglichkeit der direkten Korrektur
- Anzeige von Konfigurationsfehlern
- Anzeige von Wertänderungen in Folge von Parameterabhängigkeiten
- Anzeige aller Speicherwerte eines Parameters: Startwert in CPU, Startwert im Projekt, Beobachtungswert
- Anzeige des Parametervergleichs der Speicherwerte eines Parameters

Bedienfunktionen:

- Navigation, um schnell zwischen den Parametern und Parameterstrukturen zu wechseln.
- Text-Filter, um bestimmte Parameter schneller zu finden.
- Sortierfunktion, um die Reihenfolge von Parameter und Parametergruppen dem Bedarf anzupassen.
- Speicherfunktion, um strukturelle Einstellungen der Parametersicht zu sichern.
- Parameterwerte online beobachten und steuern.
- Anzeigeformat des Werts umstellen.
- Momentaufnahme von Parameterwerten der CPU speichern, um kurzzeitige Situationen abzubilden und darauf zu reagieren.
- Momentaufnahme von Parameterwerten als Startwerte übernehmen.
- Geänderte Startwerte in die CPU laden.
- Vergleichsfunktionen, um Parameterwerte miteinander zu vergleichen.

Gültigkeit

Die hier beschriebene Parametersicht steht für folgende Technologieobjekte zur Verfügung:

- PID_Compact
- PID_3Step
- PID_Temp
- CONT_C (nur S7-1500)
- CONT_S (nur S7-1500)
- TCONT_CP (nur S7-1500)
- TCONT_S (nur S7-1500)
- TO_Axis_PTO (S7-1200 Motion Control)

- TO_Positioning_Axis (S7-1200 Motion Control)
- TO_CommandTable_PTO (S7-1200 Motion Control)
- TO_CommandTable (S7-1200 Motion Control)

3.2.10.2 Aufbau der Parametersicht

Funktionsleiste

In der Funktionsleiste der Parametersicht sind folgende Funktionen anwählbar:

Symbol	Funktion	Erläuterung
	Alle beobachten	Startet das Beobachten der sichtbaren Parameter in der aktiven Parametersicht (Online-Betrieb).
	Momentaufnahme der Beobachtungswerte erstellen und Einstellwerte dieser Momentaufnahme als Startwerte übernehmen	Übernimmt die aktuellen Beobachtungswerte in die Spalte “Momentaufnahme“ und aktualisiert die Startwerte im Projekt. Nur im Online-Betrieb bei PID_Compact, PID_3Step und PID_Temp.
	Einstellwerte initialisieren	Überträgt die im Projekt aktualisierten Startwerte in die CPU. Nur im Online-Betrieb bei PID_Compact, PID_3Step und PID_Temp.
	Momentaufnahme der Beobachtungswerte erstellen	Übernimmt die aktuellen Beobachtungswerte in die Spalte “Momentaufnahme“. Nur im Online-Betrieb.
	Alle ausgewählten Parameter sofort und einmalig steuern	Dieser Befehl wird einmalig und schnellstmöglich ausgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm. Nur im Online-Betrieb.
Funktionsorientier	Navigationsstruktur auswählen	Wechselt zwischen der Funktionsorientierten Navigation und Datenorientierten Navigation.
< no text filter >	Textfilter...	Nach Eingabe einer Zeichenkette: Anzeige aller Parameter, die die eingegebene Zeichenkette in einer der aktuell sichtbaren Spalten enthalten.
	Vergleichswerte auswählen	Auswahl, welche Parameterwerte im Online-Betrieb miteinander verglichen werden sollen (Startwert im Projekt, Startwert in CPU, Momentaufnahme) Nur im Online-Betrieb.
	Anordnung merken	Speichert Ihre vorgenommenen Anzeigeeinstellungen der Parametersicht (z.B. ausgewählte Navigationsstruktur, aktivierte Tabellenspalten, usw.)

Navigation

Innerhalb des Registers "Parametersicht" sind folgende Navigationsstrukturen alternativ anwählbar:

Navigation		Erläuterung
Funktionsorientierte Navigation	Alle Parameter Konfigurationsparameter Inbetriebnahmeparameter Andere Parameter	In der Funktionsorientierten Navigation basiert die Struktur der Parameter auf der Struktur im Konfigurationsdialog (Register "Funktionssicht"), Inbetriebnahmedialog und Diagnosedialog. Die letzte Gruppe "Andere Parameter" enthält alle übrigen Parameter des Technologieobjektes.
Datenorientierte Navigation	Alle Parameter Input Output InOut Static Andere Parameter	In der Datenorientierten Navigation basiert die Struktur der Parameter auf der Struktur im Instanz-DB/Technologie-DB. Die letzte Gruppe "Andere Parameter" enthält diejenigen Parameter, die nicht im Instanz-DB/Technologie-DB enthalten sind.

Über die Klappliste "Navigationsstruktur auswählen" können Sie die Navigationsstruktur umstellen.

Parametertabelle

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten der Parametertabelle. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

- Spalte "Offline" = X: Spalte ist im Offline-Betrieb sichtbar.
- Spalte "Online"= X: Spalte ist im Online-Betrieb sichtbar (Online-Verbindung zur CPU).

Spalte	Erläuterung	Offline	Online
Name in Funktionssicht	Name des Parameters in der Funktionssicht. Das Anzeigefeld ist leer bei Parametern, die nicht über das Technologieobjekt konfiguriert werden.	X	X
Vollständiger Name im DB	Vollständiger Pfad des Parameters im Instanz-DB/Technologie-DB. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie-DB enthalten sind.	X	X
Name im DB	Name des Parameters im Instanz-DB/Technologie-DB. Wenn der Parameter Teil einer Struktur oder UDT ist, dann ist der Präfix ". ./" hinzugefügt. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/technologie-DB enthalten sind.	X	X
Status der Konfiguration	Anzeige der Vollständigkeit der Konfiguration durch Statussymbole. siehe Status der Konfiguration (offline) (Seite 672)	X	
Vergleichsergebnis	Ergebnis der Funktion "Werte vergleichen". Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist.		X
Startwert im Projekt	Konfigurierter Startwert im Projekt. Fehleranzeige bei syntaktisch oder technologisch falsch eingegebenen Werten.	X	X

Spalte	Erläuterung	Offline	Online
Defaultwert	Wert, mit dem der Parameter vorbelegt ist. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie-DB enthalten sind.	X	X
Momentaufnahme	Momentaufnahme der aktuellen Werte in der CPU (Beobachtungswerte). Fehleranzeige bei technologisch falschen Werten.	X	X
Startwert in CPU	Startwert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei technologisch falschen Werten.		X
Beobachtungswert	Aktueller Wert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei technologisch falschen Werten.		X
Steuerwert	Wert, mit dem der Beobachtungswert geändert werden soll. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei syntaktisch oder technologisch falsch eingegebenen Werten.		X
Auswahl Steuerwert	Auswahl der Steuerwerte, die mit Hilfe der Schaltfläche "Alle ausgewählten Parameter sofort und einmalig steuern" übertragen werden sollen. Diese Spalte wird zusammen mit der Spalte "Steuerwert" eingeblendet.		X
Minimalwert	Technologisch kleinster Wert des Parameters. Ist der Minimalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: Von den Startwerten im Projekt. • Online: Von den Beobachtungswerten.	X	X
Maximalwert	Technologisch höchster Wert des Parameters. Ist der Maximalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: Von den Startwerten im Projekt. • Online: Von den Beobachtungswerten.	X	X
Einstellwert	Kennzeichnet den Parameter als Einstellwert. Diese Parameter können online initialisiert werden.	X	X
Datentyp	Datentyp des Paramters. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie enthalten sind.	X	X
Remanenz	Kennzeichnet den Wert als remanent. Die Werte remanenter Parameter bleiben auch nach Ausschalten der Versorgungsspannung erhalten.	X	X
Erreichbar aus HMI	Zeigt an, ob HMI zur Laufzeit auf diesen Parameter zugreifen kann.	X	X
Sichtbar in HMI	Zeigt an, ob der Parameter in der Auswahlliste von HMI per Voreinstellung sichtbar ist.	X	X
Kommentar	Kurzbeschreibung des Parameters.	X	X

Siehe auch

Werte vergleichen (Seite 658)

3.2.10.3 Parametersicht öffnen

Voraussetzung

Das Technologieobjekt ist im Projektnavigator hinzufügt, d.h. der zugehörige Instanz-DB/ Technologie-DB der Anweisung ist erzeugt.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Konfiguration".
4. Wählen Sie in der rechten oberen Ecke das Register "Parametersicht".

Ergebnis

Die Parametersicht wird geöffnet. In der Parametertabelle wird jeder angezeigte Parameter durch eine Tabellenzeile repräsentiert.

Die anzeigbaren Parametereigenschaften (Tabellenspalten) sind abhängig davon, ob die Parametersicht im Offline- oder Online-Betrieb arbeitet.

Zusätzlich können Sie gezielt einzelne Tabellenspalten ein- und ausblenden.

Siehe auch

Voreinstellung der Parametersicht (Seite 666)

3.2.10.4 Voreinstellung der Parametersicht

Voreinstellungen

Für ein effektives Arbeiten mit der Parametersicht können Sie die Parameterdarstellung anpassen und die vorgenommen Einstellungen speichern.

Es sind folgende Anpassungen möglich und speicherbar:

- Spalten ein- und ausblenden
- Spaltenbreite ändern
- Reihenfolge der Spalten ändern
- Navigation umschalten
- Parametergruppe in der Navigation wählen
- Vergleichswerte auswählen

Spalten ein- und ausblenden

Um Spalten in der Parametertabelle ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

oder

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle.
2. Wählen Sie im Kontextmenü den Befehl "Alle Spalten anzeigen", wenn alle Spalten des Offline- oder Online-Betriebes angezeigt werden sollen.

Einige Spalten können nur im Online-Betrieb eingeblendet werden: siehe Parametertabelle (Seite 664).

Spaltenbreite ändern

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form eines Kreuzes annimmt.
2. Doppelklicken Sie auf diese Stelle.

oder

1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Parametertabelle.
2. Klicken Sie auf
   - "Spaltenbreite optimieren" oder
   - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

Reihenfolge der Spalten ändern

Die Spalten der Parametertabelle sind beliebig platzierbar.

Um die Reihenfolge der Spalten zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Spaltenkopf und ziehen Sie ihn per Drag&Drop an die gewünschte Stelle.
   Wenn Sie die Maustaste loslassen, wird die Spalte an der neuen Position verankert.

Navigation umschalten

Um die Anzeigestruktur der Parameter umzuschalten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste “Navigationsstruktur auswählen“ die gewünschte Navigation aus:
   - Datenorientierte Navigation
   - Funktionsorientierte Navigation

Siehe auch Navigation (Seite 664).

Parametergruppe in der Navigation wählen

Innerhalb der gewählten Navigation können Sie wählen zwischen der Anzeige “Alle Parameter“ oder der Anzeige einer gewünschten unterlagerten Parametergruppe.

1. Klicken Sie in der Navigation auf die gewünschte Parametergruppe.
   In der Parametertabelle werden nur die Parameter der Parametergruppe angezeigt.

Vergleichswerte auswählen (online)

Um die Vergleichswerte für die Funktion “Werte vergleichen“ einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste “Vergleichswerte auswählen“ die gewünschten Vergleichswerte aus:
   - Startwert im Projekt / Startwert in CPU
   - Startwert im Projekt / Momentaufnahme
   - Startwert in CPU / Momentaufnahme

Defaultmäßig ist die Option “Startwert im Projekt / Startwert in CPU“ eingestellt.

Voreinstellung der Parametersicht speichern

Um obige Anpassungen der Parametersicht zu speichern, gehen Sie folgendermaßen vor:

1. Passen Sie die Parametersicht nach Ihren Bedürfnissen an.
2. Klicken Sie auf die Schaltfläche “Anordnung merken“ rechts oben in der Parametersicht.

3.2.10.5 Arbeiten mit der Parametersicht

Übersicht

Die folgende Tabelle gibt einen Überblick über die nachfolgend beschriebenen Funktionen der Parametersicht im Online- und Offline-Betrieb.

- Spalte "Offline" = X: Diese Funktion ist im Offline-Betrieb möglich.
- Spalte "Online" = X: Diese Funktion ist im Online-Betrieb möglich.

Funktion/Aktion	Offline	Online
Parametertabelle filtern (Seite 669)	X	X
Parametertabelle sortieren (Seite 670)	X	X
Parameterdaten in andere Editoren übernehmen (Seite 670)	X	X
Fehler anzeigen (Seite 671)	X	X
Startwerte im Projekt bearbeiten (Seite 671)	X	X
Status der Konfiguration (offline) (Seite 672)	X	
Werte in der Parametersicht online beobachten (Seite 673)		X
Momentaufnahme der Beobachtungswerte erstellen (Seite 675)		X
Werte steuern (Seite 676)		X
Werte vergleichen (Seite 677)		X
Werte aus dem Online-Programm als Startwerte übernehmen (Seite 678)		X
Einstellwerte im Online-Programm initialisieren (Seite 679)		X

Parametertabelle filtern

Die Parameter in der Parametertabelle können Sie auf folgende Arten filtern:

- Mit dem Textfilter
- Mit den Untergruppen der Navigation

Beide Filtermethoden sind gleichzeitig anwendbar.

Mit dem Textfilter

Gefiltert werden kann nach Texten, die in der Parametertabelle sichtbar sind. D.h. es kann nur nach Texten in angezeigten Parameterzeilen und eingeblendeten Spalten gefiltert werden.

1. Geben Sie im Eingabefeld “Textfilter...“ die gewünschte Zeichenkette ein, nach der gefiltert werden soll.
   Die Parametertabelle zeigt nur noch diejenigen Parameter an, in denen die Zeichenkette enthalten ist.

Die Textfilterung wird zurückgesetzt:

- Bei Anwahl einer anderen Parametergruppe in der Navigation.
- Bei Wechsel zwischen Datenorientierter und Funktionsorientierter Navigation.

Mit den Untergruppen der Navigation

1. Klicken Sie in der Navigation auf die gewünschte Parametergruppe, z.B. "Static".
   In der Parametertabelle werden nur noch die Static-Parameter angezeigt. Sie können bei einigen Gruppen der Navigation weitere Untergruppen anwählen.
2. Klicken Sie in der Navigation auf “Alle Parameter“, wenn alle Parameter wieder angezeigt werden sollen.

Parametertabelle sortieren

Die Werte der Parameter sind zeilenweise angeordnet. Die Parametertabelle ist nach jeder angezeigten Spalte sortierbar.

- In Spalten mit numerischen Werten wird nach der Höhe des numerischen Wertes sortiert.
- In Text-Spalten wird alphabetisch sortiert.

Spaltenweise sortieren

1. Positionieren Sie den Mauszeiger in der Kopfzelle der gewünschten Spalte.
   Der Hintergrund dieser Zelle wird blau markiert.
2. Klicken Sie auf den Spaltenkopf.

Ergebnis

Die ganze Parametertabelle wird nach der markierten Spalte sortiert. Im Spaltenkopf erscheint ein Dreieck mit Spitze nach oben.

Durch wiederholtes Klicken auf den Spaltenkopf wird die Sortierung wie folgt geändert:

- Symbol "▲": Parametertabelle wird aufsteigend sortiert.
- Symbol "▼": Parametertabelle wird absteigend sortiert.
- Kein Symbol: Die Sortierung wird wieder aufgehoben. Die Parametertabelle nimmt die defaultmäßige Anzeige an.

Bei der Sortierung wird der Präfix "../" in der Spalte "Name im DB" ignoriert.

Parameterdaten in andere Editoren übernehmen

Nach Anwahl einer ganzen Parameterzeile der Parametertabelle können Sie mit

- Drag&Drop
- <Strg+C>/<Strg+V>
- Kopieren/Einfügen per Kontextmenü

Parameter in folgende Editoren des TIA-Portals übernehmen:

- In den Programmeditor
- In die Beobachtungstabelle
- In die Signaltabelle für Trace

Der Parameter wird mit vollständigen Namen eingefügt: vgl. Angabe in Spalte "Vollständiger Name im DB".

Fehler anzeigen

Fehleranzeige

Parametrierfehler, die zu Übersetzungsfehlern führen (z.B. Grenzwertüberschreitung), werden in der Parametersicht angezeigt.

Bei jeder Eingabe eines Wertes in der Parametersicht wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt.

Fehlerhafte Werte werden angezeigt durch:

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)

und/oder

- Tabellenfeld mit rotem Hintergrund
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der erforderlichen Syntax (Format)

Übersetzungsfehler

Von der Fehlermeldung des Compilers kann direkt die Parametersicht (Funktionsorientierte Navigation) mit dem fehlerverursachenden Parameter geöffnet werden, bei denjenigen Parametern, die nicht im Konfigurationsdialog angezeigt werden.

Startwerte im Projekt bearbeiten

Mit der Parametersicht können Sie im Offline-Betrieb und Online-Betrieb die Startwerte im Projekt bearbeiten:

- Wertänderungen nehmen Sie in der Spalte “Startwert im Projekt“ der Parametertabelle vor.
- In der Spalte “Status der Konfiguration“ der Parametertabelle wird der Fortschritt der Konfiguration durch die bekannten Statussymbole aus dem Konfigurationsdialog des Technologieobjektes angezeigt.

Randbedingungen

- Wenn vom Parameter, dessen Startwert geändert wurde, andere Parameter abhängen, dann wird der Startwert der abhängigen Parameter ebenfalls angepasst.
- Wenn ein Parameter eines Technologieobjektes nicht editierbar ist, dann ist er auch in der Parametersicht nicht editierbar. Die Editierbarkeit eines Parameters kann auch von den Werten anderer Parameter abhängen.

Neue Startwerte definieren

Um Startwerte für Parameter in der Parametersicht festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Parametersicht des Technologieobjektes.
2. Tragen Sie in die Spalte "Startwert im Projekt" die gewünschten Startwerte ein. Der Wert muss dem Datentyp des Parameters entsprechen und darf den Wertebereich des Parameters nicht überschreiten.
   In den Spalten “Maximalwert“ und “Minimalwert“ sind die Grenzwerte des Wertebereichs sichtbar.

In der Spalte "Status der Konfiguration" wird der “Fortschritt“ der Konfiguration durch farbliche Symbole angezeigt.

Siehe auch Status der Konfiguration (offline) (Seite 672)

Nach der Anpassung der Startwerte und dem Laden des Technologieobjekts in die CPU, nehmen die Parameter beim Anlauf den definierten Wert ein, sofern sie nicht als remanent deklariert sind (Spalte “Remanenz“).

Fehleranzeige

Bei Eingabe eines Startwertes wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt:

Fehlerhafte Startwerte werden angezeigt durch

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)

und/oder

- Roten Hintergrund im Feld “Startwert im Projekt“
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der notwendigen Syntax (Format)

Fehlerhafte Startwerte korrigieren

1. Korrigieren Sie fehlerhafte Startwerte mit Hilfe der Informationen aus der Roll-out-Fehlermeldung.
   Rotes Fehlersymbol, roter Feldhintergrund und Roll-out-Fehlermeldung werden nicht mehr angezeigt.

Nur mit fehlerfreien Startwerten kann das Projekt erfolgreich übersetzt werden.

Status der Konfiguration (offline)

Der Status der Konfiguration wird durch Symbole angezeigt:

- In Spalte “Status der Konfiguration“ in der Parametertabelle
- In der Navigationsstruktur der Funktionsorientierten Navigation bzw. Datenorientierten Navigation

Symbol in Spalte “Status der Konfiguration“

Symbol	Bedeutung
	Der Startwert des Parameters entspricht dem Defaultwert und ist gültig. Es wurde noch kein Startwert vom Anwender definiert.
	Der Startwert des Parameters enthält einen vom Anwender definierten oder automatisch angepassten Wert. Der Startwert unterscheidet sich vom Defaultwert. Der Startwert ist fehlerfrei und gültig.
	Der Startwert des Parameters ist ungültig (syntaktischer oder technologischer Fehler). Das Eingabefeld ist rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.
	Nur bei S7-1200 Motion Control: Der Startwert des Parameters ist gültig, enthält jedoch Warnhinweise. Das Eingabefeld ist gelb hinterlegt.
	Der Parameter ist in der aktuellen Konfiguration nicht relevant.

Symbol in der Navigation

Die Symbole in der Navigation zeigen den “Fortschritt“ der Konfiguration in gleicher Weise an wie im Konfigurationsdialog des Technologieobjektes.

Siehe auch

Technologieobjekte konfigurieren (Seite 654)

Werte in der Parametersicht online beobachten

Sie können die Werte, die die Parameter des Technologieobjektes aktuell in der CPU einnehmen (Beobachtungswerte), direkt in der Parametersicht beobachten.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Der Parametersicht des Technologieobjekt ist geöffnet.

Vorgehen

1. Starten Sie die Beobachtung, indem Sie auf das Symbol klicken.
   Sobald die Parametersicht online ist, werden folgende Spalten zusätzlich angezeigt:
   - Vergleichsergebnis
   - Startwert in CPU
   - Beobachtungswert
   - Steuerwert
   - Auswahl Steuerwert

   Die Spalte "Beobachtungswert" zeigt die aktuellen Parameterwerte auf der CPU an.
   Bedeutung der weiteren Spalten: siehe Parametertabelle (Seite 664)
2. Sie beenden die Beobachtung, indem Sie erneut auf das Symbol klicken.

Anzeige

Alle ausschließlich online verfügbaren Spalten haben einen orangen Hintergrund:

- Werte in den hellorangen Zellen sind änderbar.
- Werte in Zellen mit dunklorangem Hintergrund sind nicht änderbar.

Anzeigeformat des Werts umstellen

Das Anzeigeformat des Werts ist über das Kontextmenü einer Tabellenzeile in der Parametersicht des Technologieobjekts auswählbar.

Das Anzeigeformat der folgenden Werte kann sowohl im Online- als auch im Offline-Betrieb umgestellt werden:

- Startwert im Projekt
- Startwert in CPU
- Maximalwert
- Minimalwert
- Momentaufnahme
- Beobachtungswert
- Defaultwert
- Steuerwert

Das eingestellte Anzeigeformat gilt für alle Werte der Tabellenzeile.

Folgende Anzeigeformate des Werts sind änderbar:

- Voreinstellung
- Hex
- Oktal
- Bin

- Dec (+/-)
- DEC

Abhängig von dem in der Parametersicht ausgewählten Parameter sind nur die unterstützten Anzeigeformate auswählbar.

Voraussetzungen

- Die Parametersicht des Technologieobjekts ist geöffnet.

Vorgehen

Um das Anzeigeformat des Werts umzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Tabellenzeilen, in der Sie das Anzeigeformat umstellen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigeformat".
3. Wählen Sie das gewünschte Anzeigeformat aus.

Hinweis

Um das Anzeigeformat eines bestimmten Datentyps in mehreren Tabellenzeilen zu ändern, sortieren Sie die Parametersicht nach diesem Datentyp. Anschließend selektieren Sie die erste und letzte Tabellenzeile mit diesem Datentyp bei gedrückter <Shift>-Taste und ändern das Anzeigeformat für die selektierten Tabellenzeilen.

Momentaufnahme der Beobachtungswerte erstellen

Sie können die aktuellen Werte des Technologieobjektes auf der CPU (Beobachtungswerte) sichern und in der Parametersicht anzeigen.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um die aktuellen Parameterwerte anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Parametersicht auf das Symbol “Momentaufnahme der Beobachtungswerte erstellen".

Ergebnis

Die aktuellen Beobachtungswerte werden einmalig in die Spalte "Momentaufnahme" der Parametertabelle übernommen.

Sie können die so "eingefrorenen" Werte analysieren, während die Beobachtungswerte in der Spalte "Beobachtungswerte" weiterhin aktualisiert werden.

Werte steuern

Mit der Parametersicht können Sie Werte des Technologieobjektes in der CPU steuern.

Sie können dem Parameter einmalig Werte zuweisen (Steuerwert) und diese sofort steuern. Beim Ausführen wird der Steuerauftrag schnellstmöglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR
Gefahr beim Steuern: Ein Verändern der Parameterwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.
- Der Parameter kann gesteuert werden (zugehöriges Feld in der Spalte "Steuerwert" hat hellorangen Hintergrund).

Vorgehen

Um Parameter sofort zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Steuerwerte in die Spalte “Steuerwerte“ der Parametertabelle ein.
2. Prüfen Sie, ob das Optionskästchen für das Steuern in der Spalte "Auswahl Steuerwert" aktiviert ist.
   Die Steuerwerte und zugehörige Optionskästchen abhängiger Parameter werden automatisch mit angepasst.
3. Klicken Sie auf das Symbol “Alle ausgewählten Parameter sofort und einmalig steuern".

Die ausgewählten Parameter werden einmalig und sofort mit den vorgegebenen Werten gesteuert und können in der Spalte "Bebachtungswerte" beobachtet werden. Die Optionskästchen für das Steuern in der Spalte "Auswahl Steuerwert" werden nach Durchführung des Steuerauftrags automatisch deaktiviert.

Fehleranzeige

Bei Eingabe eines Steuerwertes wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt:

Fehlerhafte Steuerwerte werden angezeigt durch

- Roten Hintergrund im Feld “Steuerwert“

und

- Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der notwendigen Syntax (Format)

Fehlerhafte Steuerwerte

- Technologisch falsche Steuerwerte können übertragen werden.
- Syntaktisch falsche Steuerwerte können **nicht** übertragen werden.

Werte vergleichen

Über Vergleichfunktionen können Sie die folgenden Speicherwerte eines Parameters vergleichen:

- Startwert im Projekt
- Startwert in CPU
- Momentaufnahme

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um die Startwerte auf den verschiedenen Zielsystemen zu vergleichen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Vergleichswerte auswählen".
   Eine Auswahlliste mit den Vergleichsoptionen wird geöffnet:
   - Startwert im Projekt - Startwert in CPU (Defaulteinstellung)
   - Startwert im Projekt - Momentaufnahme
   - Startwert in CPU - Momentaufnahme
2. Wählen Sie die gewünschte Vergleichsoption.
   Die gewählte Vergleichsoption wird wie folgt ausgeführt:
   - In den Kopfzellen der beiden zum Vergleich ausgewählten Spalten erscheint ein Waage-Zeichen.
   - In der Spalte "Vergleichsergebnis" wird das Ergebnis des Vergleichs der gewählten Spalten durch Symbole angezeigt.

Symbol in Spalte "Vergleichergebnis"

Symbol	Bedeutung
	Die Vergleichswerte sind gleich und fehlerfrei.
	Die Vergleichswerte sind ungleich und fehlerfrei.
	Mindestens einer der beiden Vergleichswerte ist technologisch oder syntaktisch falsch.
	Der Vergleich kann nicht durchgeführt werden. Mindestens einer der beiden Vergleichswerte ist nicht verfügbar (z.B. Momentaufnahme).
	Der Wert ist nicht sinnvoll vergleichbar, da er in einer der Konfigurationen nicht relevant ist.

Symbol in der Navigation

Die Symbole werden in gleicher Weise in der Navigation angezeigt, wenn mindestens für einen der Parameter unterhalb der angezeigten Navigationsstruktur das Vergleichsergebnis zutrifft.

Werte aus dem Online-Programm als Startwerte übernehmen

Um in einem Schritt optimierte Werte aus der CPU als Startwerte in das Projekt zu übernehmen, erzeugen Sie eine Momentaufnahme der Beobachtungswerte. Als "Einstellwert" markierte Werte der Momentaufnahme werden anschließend als Startwerte in das Projekt übernommen.

Voraussetzungen

- Das Technologieobjekt ist vom Typ "PID_Compact" oder "PID_3Step".
- Eine Online-Verbindung besteht.

- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um optimierte Werte aus der CPU zu übernehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Momentaufnahme der Beobachtungswerte erstellen und Einstellwerte dieser Momentaufnahme als Startwerte übernehmen".

Ergebnis

Die aktuellen Beobachtungswerte werden in die Spalte "Momentaufnahme" übernommen und deren Einstellwerte als neue Startwerte in die Spalte "Startwert im Projekt" kopiert.

Hinweis

Werte einzelner Parameter übernehmen

Sie können auch die Werte einzelner Parameter, die nicht als Einstellwert markiert sind, aus der Spalte "Momentaufnahme" in die Spalte "Startwerte im Projekt" übernehmen. Kopieren Sie dazu die Werte mit Hilfe der Befehle "Kopieren" und "Einfügen" aus dem Kontextmenü und fügen Sie sie in die Spalte "Startwert im Projekt" ein.

Einstellwerte im Online-Programm initialisieren

Sie können alle Parameter, die in der Parametersicht als "Einstellwert" markiert sind, in einem Schritt in der CPU mit neuen Werten initialisieren. Dabei werden die Startwerte aus dem Projekt in die CPU geladen. Die CPU bleibt im Betriebszustand "RUN".

Um Datenverlust auf der CPU bei Kaltstart oder Neustart (Warmstart) zu vermeiden, müssen Sie zusätzlich noch das Technologieobjekt in die CPU laden.

⚠ GEFAHR

Gefahr beim Ändern von Parameterwerten

Ein Verändern der Parameterwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Einstellwerte neu initialisieren.

Voraussetzungen

- Das Technologieobjekt ist vom Typ "PID_Compact" oder "PID_3Step".
- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.
- Die Parameter, die als "Einstellwert" markiert sind, verfügen über einen technologisch und syntaktisch fehlerfreien "Startwert im Projekt".

Vorgehen

Um alle Einstellwerte zu initialisieren, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Werte in die Spalte "Startwert im Projekt" ein.
   Achten Sie darauf, dass die Startwerte technologisch und syntaktisch fehlerfrei sind.
2. Klicken Sie auf das Symbol "Einstellwerte initialisieren".

Ergebnis

Die Einstellwerte in der CPU werden mit den Startwerten aus dem Projekt initialisiert.

3.2.11 Instanz-DB eines Technologieobjekts anzeigen

Zu jedem Technologieobjekt wird ein Instanz-DB angelegt, in dem die Parameter und statischen Variablen gespeichert sind.

Vorgehen

Um den Instanz-DB eines Technologieobjekts anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Markieren Sie ein Technologieobjet.
4. Wählen Sie im Kontextmenü den Befehl "Öffnen im Editor".

3.3 PID-Regelung (S7-1200, S7-1500)

3.3.1 PID_Compact einsetzen

3.3.1.1 Technologieobjekt PID_Compact

Das Technologieobjekt PID_Compact stellt einen kontinuierlichen PID-Regler mit integrierter Optimierung zur Verfügung. Alternativ können Sie einen Impulsregler konfigurieren. Es sind Hand- und Automatikbetrieb möglich.

PID-Compact erfasst innerhalb eines Regelkreises fortlaufend den gemessenen Istwert und vergleicht diesen mit dem gewünschten Sollwert. Aus der sich ergebenden Regeldifferenz errechnet die Anweisung PID_Compact einen Ausgangswert, durch den der Istwert möglichst schnell und stabil an den Sollwert angeglichen wird. Der Ausgangswert setzt sich beim PID-Regler aus drei Anteilen zusammen:

- **P**-Anteil
  Der P-Anteil des Ausgangswerts steigt proportional zur Regeldifferenz.
- **I**-Anteil
  Der I-Anteil des Ausgangswerts steigt solange bis die Regeldifferenz ausgeglichen ist.
- **D**-Anteil
  Der D-Anteil steigt mit wachsender Änderungsgeschwindigkeit der Regeldifferenz. Der Istwert wird möglichst schnell an den Sollwert angeglichen. Nimmt die Änderungsgeschwindigkeit der Regeldifferenz wieder ab, so verringert sich der D-Anteil wieder.

Die Anweisung PID_Compact berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der Erstoptimierung. Die Parameter können über eine Nachoptimierung weiter optimiert werden. Sie brauchen die Parameter nicht manuell ermitteln.

Weitere Informationen

- Übersicht der Software-Regler (Seite 651)
- Technologieobjekte hinzufügen (Seite 653)
- Technologieobjekte konfigurieren (Seite 654)
- PID_Compact V2 konfigurieren (Seite 682)
- PID_Compact V1 konfigurieren (Seite 702)

FAQ

Weitere Informationen hierzu finden Sie im folgenden FAQ im Siemens Industry Online Support:

- Beitrags-ID 79047707 (https://support.industry.siemens.com/cs/ww/de/view/79047707)

3.3.1.2 PID_Compact V2

PID_Compact V2 konfigurieren

Grundeinstellungen V2

Einleitung V2

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID_Compact:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)

Sollwert, Istwert und Ausgangswert

Sollwert, Istwert und Ausgangswert können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB
  Es wird der Wert verwendet der im Instanz-DB gespeichert ist.
  Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.
  An der Anweisung darf kein Wert stehen.
  Änderung über HMI möglich.
- Anweisung
  Es wird der Wert verwendet, der an der Anweisung verschaltet ist.
  Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.
  Änderung über HMI nicht möglich.

Regelungsart V2

Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Sollwert, Istwert und Störgröße. Sollwert, Istwert und Störgröße werden in dieser Einheit angezeigt.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID_Compact arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktiveren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

Anlaufverhalten

1. Um nach Neustart der CPU in die Betriebsart "Inaktiv" zu wechseln, deaktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
   Um nach Neustart der CPU in die Betriebsart zu wechseln, die an Mode gespeichert ist, aktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
2. Wählen Sie in der Klappliste "Mode setzen auf" die Betriebsart aus, die nach einem vollständigen Laden in Gerät aktiviert werden soll.
   Nach einem vollständigen Laden in Gerät startet PID_Compact in der gewählten Betriebsart. Bei jedem weiteren Neustart, startet PID_Compact in der Betriebsart, die zuletzt an Mode gespeichert war.

Beispiel

Sie haben das Optionskästchen "Nach CPU Neustart Mode aktivieren" aktiviert und in der Liste "Mode setzen auf" den Eintrag "Erstoptimierung" gewählt. Nach einem vollständigen Laden in Gerät startet PID_Compact in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung noch aktiv ist, startet PID_Compact nach dem Neustart der CPU wieder in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung erfolgreich beendet wurde und der Automatikbetrieb aktiv ist, startet PID_Compact nach dem Neustart der CPU im "Automatikbetrieb".

Sollwert V2

Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.
   Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

Istwert V2

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID_Compact den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

Ausgangswert V2

PID_Compact stellt drei Ausgangswerte zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output_PER
  Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output
  Der Ausgangswert muss über das Anwenderprogramm aufbereitet werden, z.B. weil das Stellglied nichtlineares Verhalten zeigt.
- Output_PWM
  Das Stellglied wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.

Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output_PER (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output".
2. Wählen Sie "Instanz-DB".
   Der berechnete Ausgangswert wird im Instanz-DB gespeichert.
3. Verwenden Sie für die Aufbereitung des Ausgangswerts den Ausgangsparameter Output.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Digital- oder Analogausgang der CPU zum Stellglied.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output_PWM".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Digitalausgangs ein.

Istwerteinstellungen V2

Istwert skalieren V2

Wenn Sie in der Grundeinstellung die Verwendung von Input_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input_PER anhand eines unteren und oberen Wertepaars skaliert.

Vorgehen

Um den Istwert zu skalieren, gehen Sie folgendermaßen vor:

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID_Compact.
2. Verschalten Sie in den Grundeinstellungen Input_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

Istwertgrenzen V2

Als Grenzwerte müssen Sie für Ihre Regelstrecke eine sinnvolle absolute Ober- und Untergrenze des Istwerts festlegen. Sobald diese Grenzen über- oder unterschritten werden, tritt ein Fehler auf (ErrorBits = 0001h). Die Optimierung wird abgebrochen, wenn die Istwertgrenzen übrschritten werden. Wie PID_Compact im Fehlerfall im Automatikbetrieb reagiert, konfigurieren Sie bei den Ausgangswerteinstellungen.

Erweiterte Einstellungen V2

Istwertüberwachung V2

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID_Compact eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID_Compact verhält sich folgendermaßen:

Istwert	InputWarning_H	InputWarning_L	ErrorBits	Betriebsart
> 98 °C	TRUE	FALSE	0001h	Inaktiv oder Ersatzausgangs-wert mit Fehler-überwachung
≤ 98 °C und > 90 °C	TRUE	FALSE	0000h	Automatikbetrieb
≤ 90 °C und ≥ 10 °C	FALSE	FALSE	0000h	Automatikbetrieb
< 10°C und ≥ 0 °C	FALSE	TRUE	0000h	Automatikbetrieb
< 0 °C	FALSE	TRUE	0001h	Inaktiv oder Ersatzausgangs-wert mit Fehler-überwachung

Wie PID_Compact beim Über- bzw. Unterschreiten der Ober- oder Untergrenze Istwert reagiert, konfigurieren Sie bei den Ausgangswerteinstellungen.

PWM-Begrenzungen V2

Über eine Pulsweitenmodulation wird der Wert am Ausgangsparameter Output in eine Impulsfolge transformiert, die am Ausgangsparameter Output_PWM ausgegeben wird. Output wird in der Abtastzeit PID-Algorithmus berechnet, Output_PWM wird in der Abtastzeit PID_Compact ausgegeben.

Die Abtastzeit PID-Algorithmus wird während der Erst- oder Nachoptimierung ermittelt. Wenn Sie die PID-Parameter manuell einstellen, müssen Sie dort auch die Abtastzeit PID-Algorithmus konfigurieren. Die Abtastzeit PID_Compact entspricht der Zykluszeit des aufrufenden OBs.

Die Impulsdauer ist proportional zum Wert an Output und beträgt stets ein ganzzahliges Vielfaches der Abtastzeit PID_Compact.

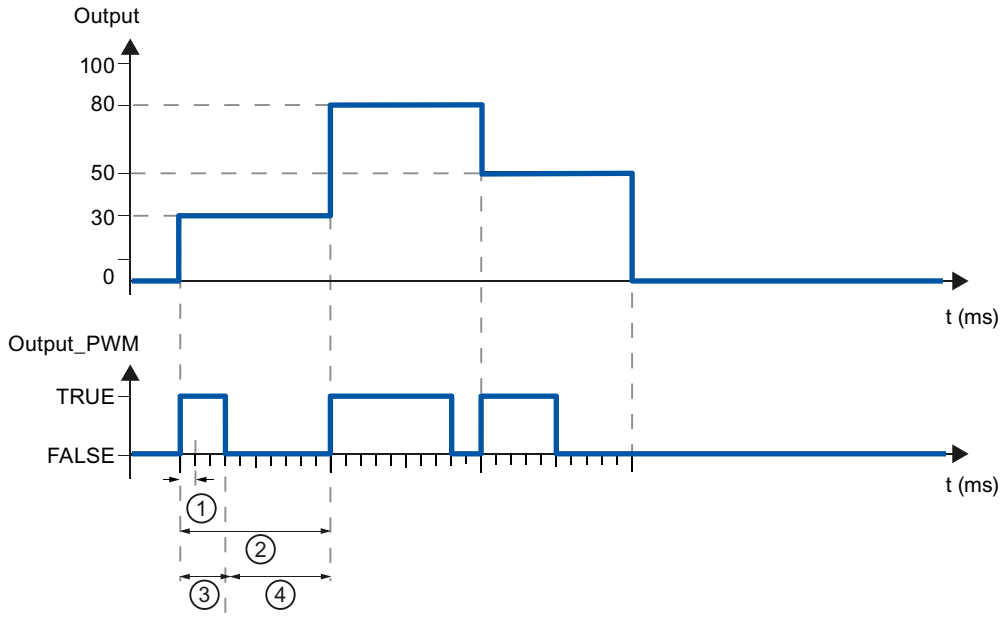

① Abtastzeit PID_Compact
② Abtastzeit PID-Algorithmus
③ Impulsdauer
④ Pausendauer

Die "Minimale Einschaltzeit" und die "Minimale Ausschaltzeit" werden auf ein ganzzahliges Vielfaches der Abtastzeit PID_Compact gerundet.

Ein Impuls oder eine Pause wird nie kürzer als die minimale Ein- oder Ausschaltzeiten. Die Ungenauigkeiten, die sich dadurch ergeben, werden aufsummiert und im nächsten Zyklus ausgeglichen.

Beispiel

Abtastzeit PID_Compact = 100 ms

Abtastzeit PID-Algorithmus = 1000 ms

Minimale Einschaltzeit = 200 ms

Output beträgt konstant 15 %. Der kleinste Impuls, den PID_Compact ausgeben kann, entspricht 20%. Im ersten Zyklus wird kein Impuls ausgegeben. Im zweiten Zyklus wird der nicht ausgegebene Impuls des ersten Zyklus zum Impuls des zweiten Zyklus addiert.

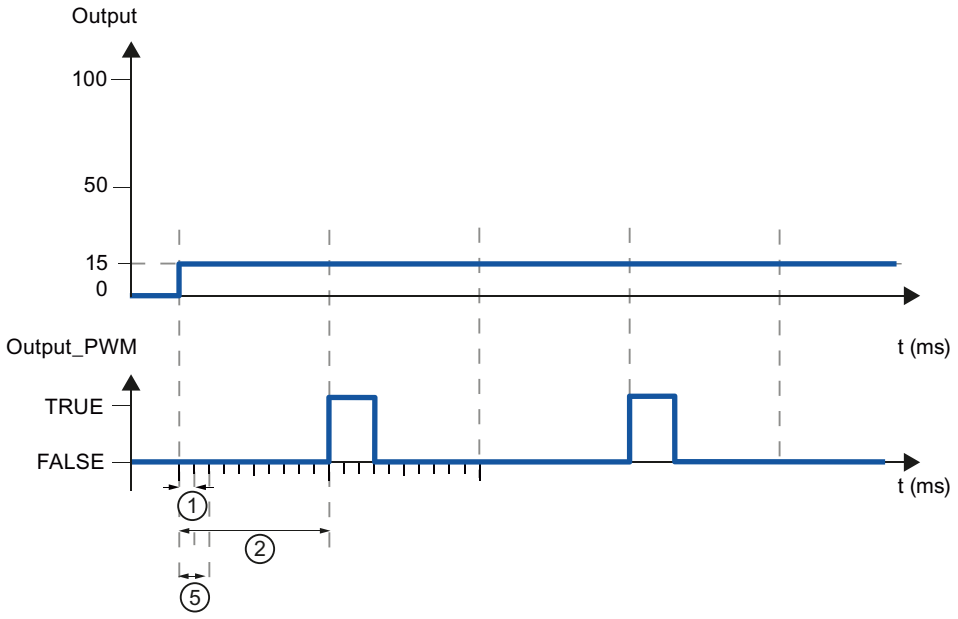

① Abtastzeit PID_Compact
② Abtastzeit PID-Algorithmus
⑤ Minimale Einschaltzeit

Um die Schalthäufigkeit zu verringern und das Stellglied zu schonen, verlängern Sie die minimalen Ein- und Ausschaltzeiten.

Wenn Sie "Output" oder "Output_PER" verwenden, müssen Sie die minimale Einschaltzeit und die minimale Ausschaltzeit auf den Wert 0.0 konfigurieren.

Hinweis

Die minimalen Ein- und Ausschaltzeiten wirken nur auf den Ausgangsparameter Output_PWM und werden nicht für evtl. in der CPU integrierte Impulsgeneratoren verwendet.

Ausgangswert V2

Ausgangswertgrenzen

Konfigurieren Sie im Konfigurationsfenster "Ausgangswertgrenzen" die absoluten Grenzen Ihres Ausgangswerts in Prozent. Absolute Ausgangswertgrenzen werden weder im Handbetrieb noch im Automatikbetrieb über- oder unterschritten. Wird im Handbetrieb ein Ausgangswert außerhalb der Grenzen vorgegeben, so wird der effektive Wert in der CPU auf die konfigurierten Grenzen begrenzt.

Die Ausgangswertgrenzen müssen zum Regelsinn passen.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

Output	-100.0 bis 100.0 %
Output_PER	-100.0 bis 100.0 %
Output_PWM	0.0 bis 100.0 %

Verhalten im Fehlerfall

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn Sie im Fehlerfall "Aktueller Wert für die Fehlerdauer" oder "Ersatzausgangswert für die Fehlerdauer" ausgeben, bleibt PID_Compact im Automatikbetrieb. Dadurch können die Istwertgrenzen überschritten und Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

PID_Compact ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

Im Fehlerfall gibt PID_Compact einen konfigurierbaren Ausgangswert aus:

- Null (Inaktiv)
  PID_Compact gibt bei allen Fehlern 0.0 als Ausgangswert aus und schaltet in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate reaktiviert.
- Aktuellen Wert für die Fehlerdauer
  Wenn im **Automatikbetrieb** folgende Fehler aufgetreten sind, kehrt PID_Compact in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.
  Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_Compact im Automatikbetrieb:
  - 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen.
  - 0800h: Abtastzeitfehler
  - 40000h: Ungültiger Wert am Parameter Disturbance.

  Wenn im **Automatikbetrieb** einer oder mehrere der folgenden Fehler auftreten, wechselt PID_Compact in die Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt den letzten gültigen Ausgangswert aus:
  - 0002h: Ungültiger Wert am Parameter Input_PER.
  - 0200h: Ungültiger Wert am Parameter Input.
  - 0400h: Berechnung des Ausgangswerts fehlgeschlagen.
  - 1000h: Ungültiger Wert am Parameter Setpoint.

  Wenn im **Handbetrieb** ein Fehler auftritt, verwendet PID_Compact weiterhin den Handwert als Ausgangswert. Wenn der Handwert ungültig ist, wird der Ersatzausgangswert verwendet. Wenn Handwert und Ersatzausgangswert ungültig sind, wird die Untergrenze Ausgangswert verwendet.
  Wenn während einer **Erst- oder Nachoptimierung** der folgende Fehler auftritt, bleibt PID_Compact in der aktiven Betriebsart:
  - 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt.

  Bei allen anderen Fehlern bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.
  Sobald keine Fehler nicht mehr anstehen, kehrt PID_Compact in den Automatikbetrieb zurück.
- Ersatzausgangswert für die Fehlerdauer
  PID_Compact gibt den Ersatzausgangswert aus.
  Wenn der folgende Fehler auftritt, bleibt PID_Compact in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt die Untergrenze Ausgangswert aus:
  - 20000h: Ungültiger Wert an der Variablen SubstituteOutput.

  Bei allen anderen Fehler verhält sich PID_Compact wie bei "Aktuellen Wert für die Fehlerdauer" beschrieben.

PID-Parameter V2

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Hinweis

Die aktuell wirksamen PID-Parameter befinden sich bei PID_Compact V1 in der Struktur sRet und bei PID_Compact V2 in der Struktur Retain.CtrlParams.

Um ein Fehlverhalten des PID-Reglers zu vermeiden, ändern Sie online die aktuell wirksamen PID-Parameter nur in der Betriebsart "inaktiv".

Wenn Sie die PID-Parameter in den Betriebsarten "Automatikbetrieb" oder "Handbetrieb" online ändern möchten, dann ändern Sie die PID-Parameter folgendermaßen:

- PID_Compact V1: Ändern Sie die PID-Parameter in der Struktur sBackUp und übernehmen Sie diese Änderungen per sPid_Cmpt.b_LoadBackUp = TRUE in die Struktur sRet.
- PID_Compact V2: Ändern Sie die PID-Parameter in der Struktur CtrlParamsBackUp und übernehmen Sie diese Änderungen per LoadBackUp = TRUE in die Struktur Retain.CtrlParams.

Online Änderungen an den PID-Parametern in der Betriebsart "Automatikbetrieb" können zu Sprüngen am Ausgangswert führen.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

y	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient Differenzierverzug (Differenzierverzug T1 = a ×T_D)
T_D	Differenzierzeit
c	Gewichtung des D-Anteils

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:

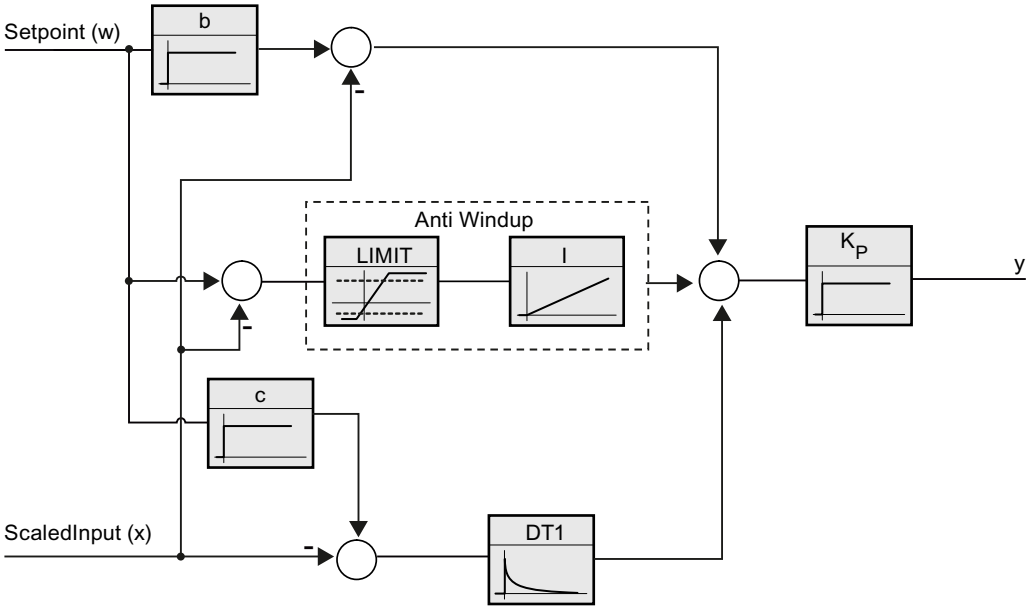

Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID_Compact vollständig laden.

Technologieobjekte in Gerät laden (Seite 656)

Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID_Compact arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0. Wenn die Integrationszeit in der Betriebsart "Automatikbetrieb" von einem anderen Wert auf 0.0 online geändert wird, dann wird der bisherige I-Anteil gelöscht und es erfolgt ein Sprung des Ausgangswerts.

Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

Koeffizient Differenzierverzug

Die Wirkung des D-Anteils wird durch den Koeffizent Differenzierverzug verzögert.

Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Abtastzeit PID-Algorithmus entspricht der Periodendauer der Pulsweitenmodulation. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

Regel für Optimierung

Wählen Sie in der Klappliste "Reglerstruktur" aus, ob PI oder PID-Parameter berechnet werden.

- **PID**
  Berechnet während der Erst- und Nachoptimierung PID-Parameter
- **PI**
  Berechnet während der Erst- und Nachoptimierung PI-Parameter
- **Benutzerdefiniert**
  Wenn Sie über ein Anwenderprogramm unterschiedliche Reglerstrukturen für Erst- und Nachoptimierung eingestellt haben, wird "Benutzerdefiniert" in der Klappliste angezeigt.

Siehe auch

Wahl der Reglerstruktur bei gegebener Regelstrecke (Seite 649)

PID_Compact V2 in Betrieb nehmen

Erstoptimierung V2

Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Voraussetzung

- Die Anweisung "PID_Compact" wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- PID_Compact befindet sich in der Betriebsart "Handbetrieb", "Inaktiv" oder "Automatikbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz zwischen Obergrenze Istwert und Untergrenze Istwert.
- Der Abstand zwischen Sollwert und Istwert ist > 50% des Sollwerts.

Vorgehen

Um die "Erstoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Compact > Inbetriebnahme".
2. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Die Erstoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, verhält sich PID_Compact wie unter Verhalten im Fehlerfall konfiguriert.

Nachoptimierung V2

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Voraussetzung

- Die Anweisung PID_Compact wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE

- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID_Compact befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

Ablauf abhängig von Startsituation

Die Nachoptimierung können Sie aus den Betriebsarten "Inaktiv", "Automatikbetrieb" oder "Handbetrieb" starten. Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb
  Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.
  PID_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb
  Wenn die Voraussetzungen für eine Erstoptimierung erfüllt sind, wird eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, verhält sich PID_Compact wie unter Verhalten im Fehlerfall konfiguriert.
  Wenn sich der Istwert für eine Erstoptimierung bereits zu nah am Sollwert befindet, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.

Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Der Ablauf der Nachoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wenn während der Nachoptimierung keine Fehler auftraten, wurden die PID-Parameter optimiert. PID_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der "Nachoptimierung" Fehler auftraten, verhält sich PID_Compact wie unter Verhalten im Fehlerfall konfiguriert.

Betriebsart "Handbetrieb" V2

Im Folgenden wird beschrieben, wie Sie die Betriebsart "Handbetrieb" im Inbetriebnahmefenster des Technologieobjekts "PID_Compact" nutzen können. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.

Voraussetzung

- Die Anweisung "PID_Compact" wird in einem Weckalarm-OB aufgerufen.
- Es wurde eine Onlineverbindung zur CPU aufgebaut und die CPU befindet sich im Betriebzustand "RUN".

Vorgehen

Verwenden Sie im Inbetriebnahmefenster "Handbetrieb", wenn Sie die Regelstrecke durch Vorgabe eines Handwerts testen möchten. Um einen Handwert vorzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
2. Aktivieren Sie im Bereich "Online-Zustand des Reglers" das Kontrollkästchen "Handbetrieb".
   PID_Compact arbeitet im Handbetrieb. Der zuletzt aktuelle Ausgangswert bleibt aktiv.
3. Tragen Sie im Feld "Output" den Handwert in der Einheit % ein.
4. Klicken Sie auf das Symbol .

Ergebnis

Der Handwert wird in die CPU geschrieben und ist sofort wirksam.

Deaktivieren Sie das Kontrollkästchen "Handbetrieb", wenn der Ausgangswert wieder durch den PID-Regler vorgegeben werden soll. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

Ablöseregelung mit PID_Compact V2

Ablöseregelung

Bei einer Ablöseregelung teilen sich zwei oder mehr Regler ein gemeinsames Stellglied. Zu jedem Zeitpunkt hat nur ein Regler Zugriff auf das Stellglied und wirkt auf den Prozess.

Eine Logik entscheidet welcher Regler Zugriff auf das Stellglied erhält. Häufig erfolgt diese Entscheidung anhand eines Vergleichs der Ausgangswerte aller Regler z. B. erhält bei einer Maximalauswahl der Regler mit dem größten Ausgangswert den Zugriff auf das Stellglied.

Die Auswahl anhand des Ausgangswerts macht es notwendig, dass alle Regler im Automatikbetrieb arbeiten. Die Regler, die nicht auf das Stellglied wirken, werden nachgeführt. Dies ist notwendig um Windup-Effekte und deren negative Auswirkungen auf das Regelverhalten und die Umschaltung zwischen den Reglern zu vermeiden.

PID_Compact unterstützt ab Version 2.3 Ablöseregelungen indem der Regler ein einfaches Verfahren zum Nachführen der nicht aktiven Regler bietet:

- Über die Variablen OverwriteInitialOutputValue und PIDCtrl.PIDInit können Sie den I-Anteil des Reglers im Automatikbetrieb so vorbelegen lassen, als ob der PID-Algorithmus im letzten Zyklus für den Ausgangswert Output = OverwriteInititalOutputValue berechnet hätte.
- OverwriteInitialOutputValue wird dazu mit dem Ausgangswert des Reglers verschaltet, der gerade Zugriff auf das Stellglied hat.
- Durch Setzen des Bit PIDCtrl.PIDInit stoßen Sie die Vorbelegung des I-Anteils an, sowie den Neustart des Reglertakts und der PWM-Periode.
- Die anschließende Berechnung des Ausgangswerts im aktuellen Zyklus erfolgt anhand des vorbelegten (und für alle Regler angeglichenen) I-Anteils sowie des P- und I-Anteils aus der aktuellen Regeldifferenz.
- Der D-Anteil ist während des Aufrufs mit PIDCtrl.PIDInit = TRUE nicht aktiv und trägt damit nicht zum Ausgangswert bei.

Dieses Verfahren stellt sicher, dass die Berechnung des aktuellen Ausgangswerts und damit die Entscheidung welcher Regler Zugriff auf das Stellglied erhält, nur anhand des aktuellen Prozesszustands und der PI-Parameter erfolgt. Windup-Effekte bei den nicht aktiven Reglern und damit falsche Entscheidungen der Umschaltlogik werden somit verhindert.

Voraussetzungen

- PIDCtrl.PIDInit ist nur wirksam, wenn der I-Anteil aktiviert ist (Variable Retain.CtrlParams.Ti > 0.0).
- Sie müssen PIDCtrl.PIDInit und OverwriteInitialOutputValue in Ihrem Anwenderprogramm selbst belegen (siehe Beispiel unten). PID_Compact nimmt keine automatische Änderung an diesen Variablen vor.
- PIDCtrl.PIDInit ist nur wirksam, wenn sich PID_Compact im Automatikbetrieb (Parameter State = 3) befindet.
- Falls möglich, wählen Sie die Abtastzeit des PID-Algorithmus (Variable Retain.CtrlParams.Cycle) so, dass sie für alle Regler identisch ist und rufen Sie alle Regler im gleichen Weckalarm-OB auf. Auf diese Weise stellen Sie sicher, dass die Umschaltung nicht innerhalb eines Reglertakts bzw. einer PWM-Periode erfolgt.

Hinweis

Ständige Adaption der Ausgangswertgrenzen

Statt dem hier beschriebenen aktiven Nachführen der Regler ohne Zugriff auf das Stellglied wird dies bei anderen Reglersystemen alternativ über eine ständige Adaption der Ausgangswertgrenzen realisiert.

Dies ist mit PID_Compact nicht möglich, da eine Änderung der Ausgangswertgrenzen im Automatikbetrieb nicht unterstützt wird.

Beispiel: Regelung einer Gas-Pipeline

PID_Compact wird für die Regelung einer Gas-Pipeline eingesetzt.

Hauptziel ist, den Durchfluss Input1 zu regeln. Hierfür wird der Regler PID_Compact_1 eingesetzt. Zusätzlich soll mit dem Begrenzungsregler PID_Compact_2 der Druck Input2 (gemessen in Strömungsrichtung vor dem Ventil) unterhalb einer Obergrenze gehalten werden.

Durchfluss und Druck werden über ein einziges Magnetventil beeinflusst. Der Ausgangswert der Regler entspricht der Ventilöffnung: Steigt der Ausgangswert, wird das Ventil geöffnet. Dadurch steigt der Durchfluss (normaler Regelsinn) während der Druck fällt (invertierter Regelsinn).

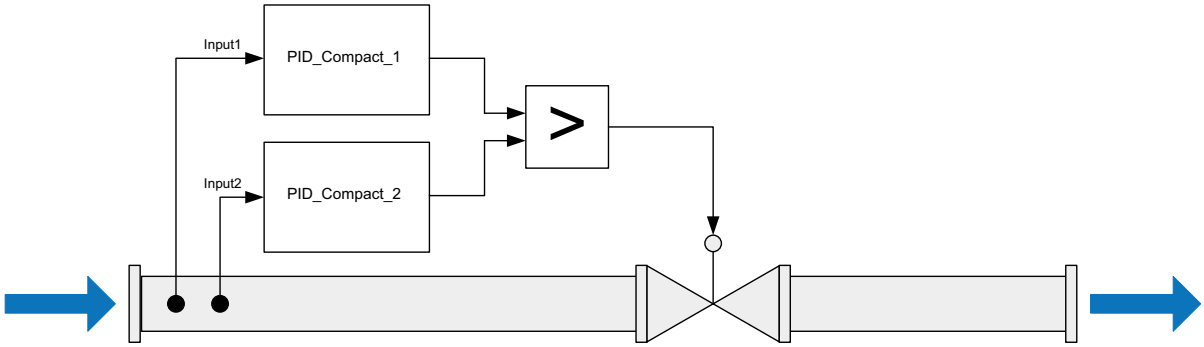

Das Ventil wird über den Ausgangswert von PID_Compact im Peripherieformat (Parameter Output_PER) angesteuert, indem die Programmvariable ActuatorInput beschrieben wird.

Die Vorgabe des Sollwerts für den Durchfluss erfolgt an Parameter PID_Compact_1.Setpoint.

Die Druckobergrenze wird als Sollwert am Parameter PID_Compact_2.Setpoint vorgegeben.

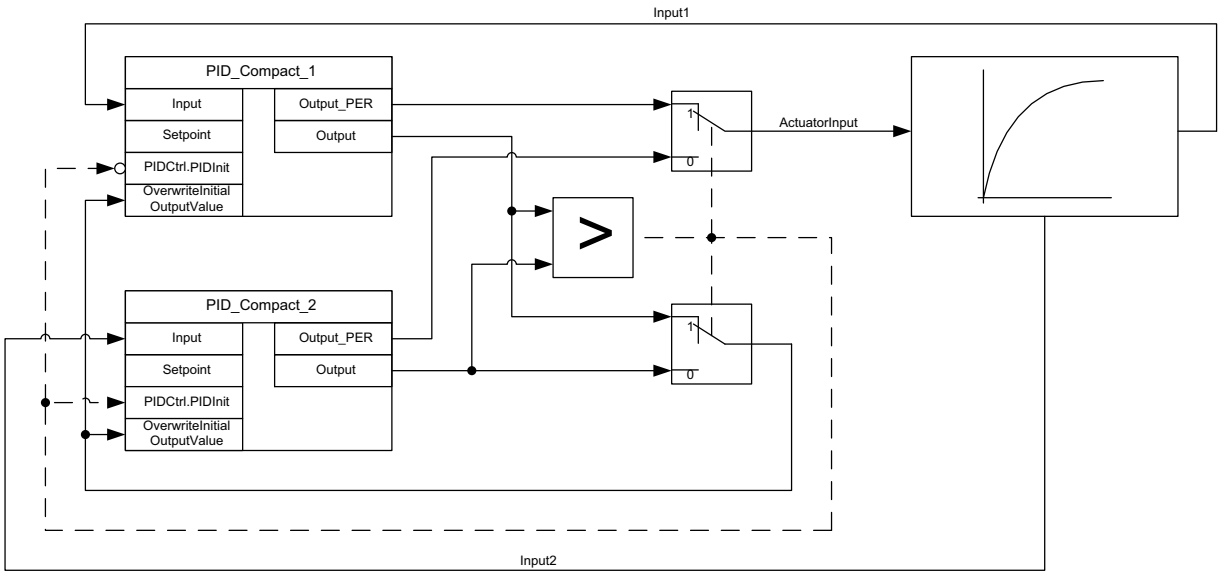

Beiden Reglern steht nur das eine Ventil als gemeinsames Stellglied zur Verfügung. Die Logik zur Entscheidung welcher Regler Zugriff auf das Stellglied erhält wird hier über eine Maximalauswahl des Ausgangswerts (im Real-Format, Parameter Output) realisiert. Da der Ausgangswert der Öffnung des Ventils entspricht, erhält somit der Regler die Kontrolle, der die größere Ventilöffnung fordert.

Hinweis

Invertierung des Regelsinns aktivieren

Da beim Druckregler PID_Compact_2 mit einer Erhöhung des Ausgangswerts (Ventilöffnung) eine Verringerung des Istwerts (Druck) erreicht werden soll, muss die Invertierung des Regelsinns aktiviert werden: PID_Compact_2.Config.InvertControl = TRUE.

Im Normalbetrieb der Anlage entspricht der Istwert des Durchflusses dem Sollwert. Der Durchflussregler PID_Compact_1 hat sich auf einen stationären Ausgangswert PID_Compact_1.Output eingeschwungenen. Der Istwert des Drucks liegt im Normalbetrieb deutlich unter der Obergrenze, die als Sollwert für PID_Compact_2 vorgegeben ist. Deshalb möchte der Druckregler das Ventil weiter schließen, um den Druck zu erhöhen, d. h. er wird einen Ausgangswert PID_Compact_2.Output berechnen der kleiner ist, als der des Durchflussreglers PID_Compact_1.Output. Die Maximalauswahl der Umschaltlogik lässt deshalb den Durchflussregler PID_Compact_1 weiter auf das Stellglied zugreifen. Zusätzlich wird über die Zuweisungen PID_Compact_2.OverwriteInitialOutputValue = PID_Compact_1.Output und PID_Compact_2.PIDCtrl.PIDInit = TRUE sichergestellt, dass PID_Compact_2 nachgeführt wird.

Nähert sich nun der Druck der Obergrenze an oder überschreitet diese, z. B. auf Grund einer Störung, wird der Druckregler PID_Compact_2 einen höheren Ausgangswert berechnen, um das Ventil weiter zu öffnen und so den Druck zu reduzieren. Ist PID_Compact_2.Output größer als PID_Compact_1.Output erhält der Druckregler PID_Compact_2 über die Maximalauswahl Zugriff auf das Stellglied und öffnet dieses. Über die Zuweisungen PID_Compact_1.OverwriteInitialOutputValue = PID_Compact_2.Output und PID_Compact_1.PIDCtrl.PIDInit = TRUE wird sichergestellt, dass PID_Compact_1 nachgeführt wird.

Der Druck wird reduziert, während der Durchfluss ansteigt und nicht mehr am Sollwert gehalten werden kann.

Steht die Störung nicht mehr an, wird der Druck weiter fallen und die Öffnung des Ventils wird durch den Druckregler verringert. Berechnet der Durchflussregler eine größere Öffnung als Ausgangswert, kehrt die Anlage in den Normalbetrieb zurück, sodass der Durchflussregler PID_Compact_1 wieder Zugriff auf das Stellglied erhält.

Dieses Beispiel kann mit folgendem SCL-Programmcode realisiert werden:

```
"PID_Compact_1"(Input := "Input1");
"PID_Compact_2"(Input := "Input2");
IF "PID_Compact_1".Output >= "PID_Compact_2".Output THEN
  "ActuatorInput" := "PID_Compact_1".Output_PER;
  "PID_Compact_1".PIDCtrl.PIDInit := FALSE;
  "PID_Compact_2".PIDCtrl.PIDInit := TRUE;
  "PID_Compact_2".OverwriteInitialOutputValue :=
  "PID_Compact_1".Output;
ELSE
  "ActuatorInput" := "PID_Compact_2".Output_PER;
  "PID_Compact_1".PIDCtrl.PIDInit := TRUE;
  "PID_Compact_2".PIDCtrl.PIDInit := FALSE;
  "PID_Compact_1".OverwriteInitialOutputValue :=
  "PID_Compact_2".Output;
END_IF;
```

PID_Compact V2 mit PLCSIM simulieren

Hinweis

Simulation mit PLCSIM

Die Simulation von PID_Compact V2.x mit PLCSIM für CPU S7-1200 wird nicht unterstützt.

PID_Compact V2.x kann nur für CPU S7-1500 mit PLCSIM simuliert werden.

Bei der Simulation mit PLCSIM ist das zeitliche Verhalten der simulierten PLC nicht exakt identisch zu einer "echten" PLC. Der tatsächliche Zeittakt eines Weckalarm-OB kann bei einer simulierten PLC größere Schwankungen aufweisen als bei "echten" PLCs.

In der Standardkonfiguration ermittelt PID_Compact die Zeit zwischen den Aufrufen automatisch und überwacht diese auf Schwankungen.

Bei der Simulation von PID_Compact mit PLCSIM kann deshalb ein Abtastzeitfehler (ErrorBits = DW#16#00000800) erkannt werden.

Dies führt zum Abbruch von laufenden Optimierungen.

Die Reaktion im Automatikbetrieb ist abhängig vom Wert der Variable ActivateRecoverMode.

Um dies zu verhindern, sollten Sie PID_Compact bei Simulation mit PLCSIM wie folgt konfigurieren:

- CycleTime.EnEstimation = FALSE
- CycleTime.EnMonitoring = FALSE
- CycleTime.Value: Weisen Sie dieser Variablen den Zeittakt des aufrufenden Weckalarm-OB in der Einheit Sekunden zu.

3.3.1.3 PID_Compact V1

PID_Compact V1 konfigurieren

Grundeinstellungen V1

Einleitung V1

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID_Compact:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)

Sollwert, Istwert und Ausgangswert

Sollwert, Istwert und Ausgangswert können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB
  Es wird der Wert verwendet der im Instanz-DB gespeichert ist.
  Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.
  An der Anweisung darf kein Wert stehen.
  Änderung über HMI möglich.
- Anweisung
  Es wird der Wert verwendet, der an der Anweisung verschaltet ist.
  Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.
  Änderung über HMI nicht möglich.

Regelungsart V1

Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Soll- und Istwert. Soll- und Istwert werden in dieser Einheit angezeigt.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID_Compact arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktiveren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

Anlaufverhalten nach Reset

Um nach Neustart der CPU sofort in die zuletzt aktive Betriebsart zu wechseln, aktivieren Sie das Optionskästchen "Nach CPU Neustart letzte Betriebsart aktivieren".

Wenn das Optionskästchen deaktiviert ist, bleibt PID_Compact in der Betriebsart "Inaktiv".

Sollwert V1

Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.
   Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

Istwert V1

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID_Compact den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

Ausgangswert V1

PID_Compact stellt drei Ausgangswerte zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output_PER
  Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output
  Der Ausgangswert muss über das Anwenderprogramm aufbereitet werden, z.B. weil das Stellglied nichtlineares Verhalten zeigt.
- Output_PWM
  Das Stellglied wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.

Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output_PER (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output".
2. Wählen Sie "Instanz-DB".
   Der berechnete Ausgangswert wird im Instanz-DB gespeichert.
3. Verwenden Sie für die Aufbereitung des Ausgangswerts den Ausgangsparameter Output.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Digital- oder Analogausgang der CPU zum Stellglied.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output_PWM".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Digitalausgangs ein.

Istwerteinstellung V1

Konfigurieren Sie im Konfigurationsfenster "Istwerteinstellungen" die Normierung Ihres Istwerts und legen Sie die absoluten Istwertgrenzen fest.

Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input_PER anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID_Compact.
2. Verschalten Sie in den Grundeinstellungen Input_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

Istwert überwachen

Legen Sie die absolute Ober- und Untergrenze des Istwerts fest. Sobald im Betrieb diese Grenzen über oder unterschritten werden, so schaltet die Regelung ab und der Ausgangswert wird auf 0% gesetzt. Als Grenzwerte müssen Sie für Ihre Regelstrecke sinnvolle Werte eingeben. Während der Optimierung sind sinnvolle Grenzwerte wichtig, um optimale PID-Parameter zu erhalten.

Die Vorbelegung der "Obergrenze Istwert" beträgt 120 %. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird keine Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_Compact wechselt in die Betriebsart "Inaktiv".

⚠ WARNUNG
Wenn Sie als Istwertgrenzen sehr hohe Werte (z. B. $-3{,}4*10^{38}$...$+3{,}4*10^{38}$) einstellen, wird die Überwachung des Istwerts deaktiviert. Dann kann es durch einen Fehler zu Schäden an Ihrer Anlage kommen.

Siehe auch

Istwertüberwachung V1 (Seite 707)

PWM-Begrenzungen V1 (Seite 707)

Ausgangswertgrenzen V1 (Seite 709)

PID-Parameter V1 (Seite 710)

Erweiterte Einstellungen V1

Istwertüberwachung V1

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID_Compact eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID_Compact verhält sich folgendermaßen:

Istwert	InputWarning_H	InputWarning_L	Betriebsart
> 98 °C	TRUE	FALSE	Inaktiv
≤ 98 °C und > 90 °C	TRUE	FALSE	Automatikbetrieb
≤ 90 °C und ≥ 10 °C	FALSE	FALSE	Automatikbetrieb
< 10°C und ≥ 0 °C	FALSE	TRUE	Automatikbetrieb
< 0 °C	FALSE	TRUE	Inaktiv

Siehe auch

Istwerteinstellung V1 (Seite 705)

PWM-Begrenzungen V1 (Seite 707)

Ausgangswertgrenzen V1 (Seite 709)

PID-Parameter V1 (Seite 710)

PWM-Begrenzungen V1

Über eine Pulsweitenmodulation wird der Wert am Ausgangsparameter Output in eine Impulsfolge transformiert, die am Ausgangsparameter Output_PWM ausgegeben wird. Output wird in der Abtastzeit PID-Algorithmus berechnet, Output_PWM wird in der Abtastzeit PID_Compact ausgegeben.

Die Abtastzeit PID-Algorithmus wird während der Erst- oder Nachoptimierung ermittelt. Wenn Sie die PID-Parameter manuell einstellen, müssen Sie dort auch die Abtastzeit PID-Algorithmus konfigurieren. Die Abtastzeit PID_Compact entspricht der Zykluszeit des aufrufenden OBs.

Die Impulsdauer ist proportional zum Wert an Output und beträgt stets ein ganzzahliges Vielfaches der Abtastzeit PID_Compact.

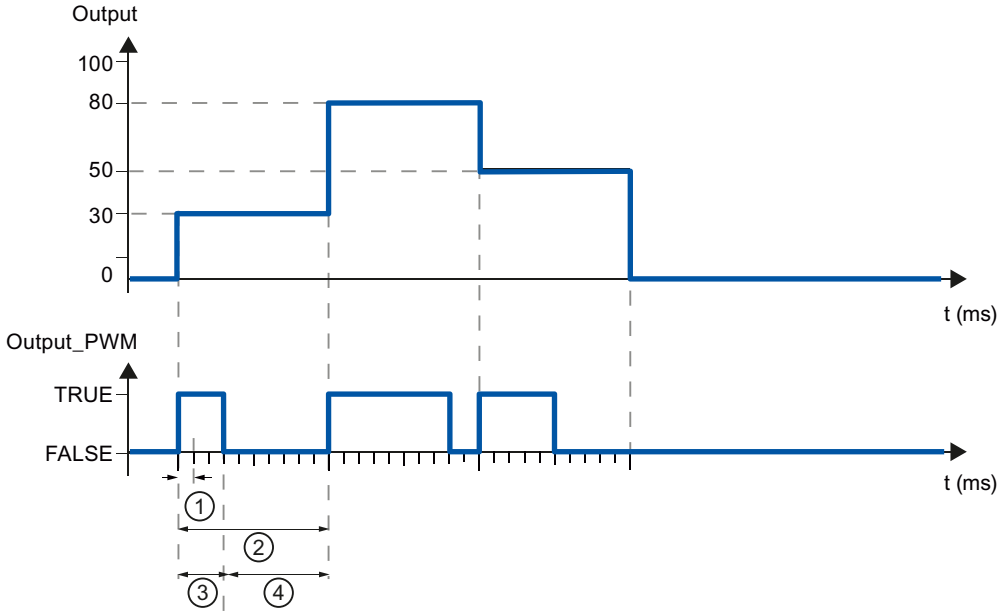

① Abtastzeit PID_Compact
② Abtastzeit PID-Algorithmus
③ Impulsdauer
④ Pausendauer

Die "Minimale Einschaltzeit" und die "Minimale Ausschaltzeit" werden auf ein ganzzahliges Vielfaches der Abtastzeit PID_Compact gerundet.

Ein Impuls oder eine Pause wird nie kürzer als die minimale Ein- oder Ausschaltzeiten. Die Ungenauigkeiten, die sich dadurch ergeben, werden aufsummiert und im nächsten Zyklus ausgeglichen.

Beispiel

Abtastzeit PID_Compact = 100 ms

Abtastzeit PID-Algorithmus = 1000 ms

Minimale Einschaltzeit = 200 ms

Output beträgt konstant 15 %. Der kleinste Impuls, den PID_Compact ausgeben kann, entspricht 20%. Im ersten Zyklus wird kein Impuls ausgegeben. Im zweiten Zyklus wird der nicht ausgegebene Impuls des ersten Zyklus zum Impuls des zweiten Zyklus addiert.

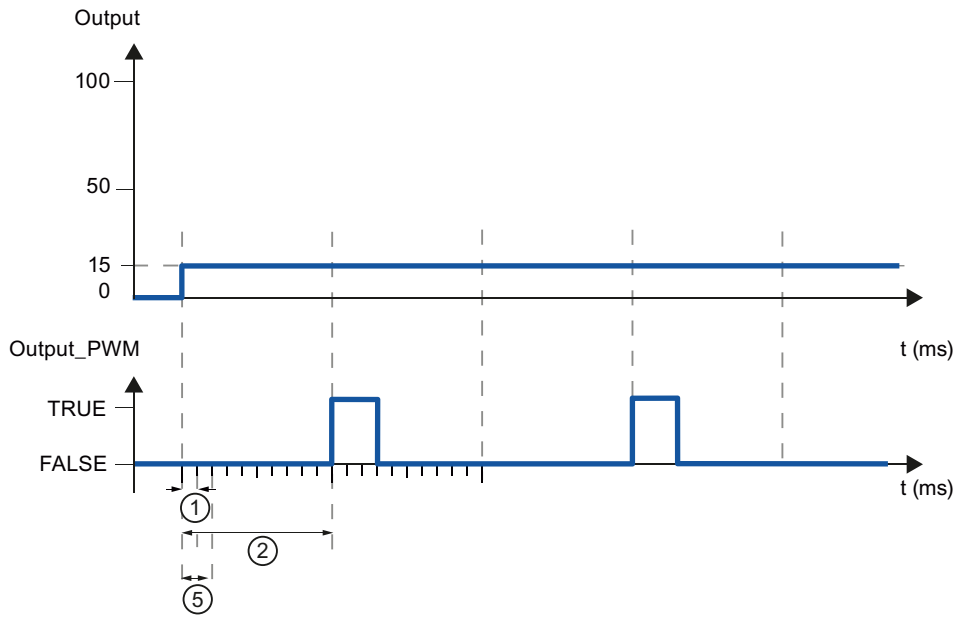

① Abtastzeit PID_Compact
② Abtastzeit PID-Algorithmus
⑤ Minimale Einschaltzeit

Um die Schalthäufigkeit zu verringern und das Stellglied zu schonen, verlängern Sie die minimalen Ein- und Ausschaltzeiten.

Wenn Sie "Output" oder "Output_PER" verwenden, müssen Sie die minimale Einschaltzeit und die minimale Ausschaltzeit auf den Wert 0.0 konfigurieren.

Hinweis

Die minimalen Ein- und Ausschaltzeiten wirken nur auf den Ausgangsparameter Output_PWM und werden nicht für evtl. in der CPU integrierte Impulsgeneratoren verwendet.

Siehe auch

Istwerteinstellung V1 (Seite 705)

Istwertüberwachung V1 (Seite 707)

Ausgangswertgrenzen V1 (Seite 709)

PID-Parameter V1 (Seite 710)

Ausgangswertgrenzen V1

Konfigurieren Sie im Konfigurationsfenster "Ausgangswertgrenzen" die absoluten Grenzen Ihres Ausgangswerts in Prozent. Absolute Ausgangswertgrenzen werden weder im Handbetrieb noch im Automatikbetrieb über- oder unterschritten. Wird im Handbetrieb ein Ausgangswert außerhalb der Grenzen vorgegeben, so wird der effektive Wert in der CPU auf die konfigurierten Grenzen begrenzt.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

Output	-100.0 bis 100.0
Output_PER	-100.0 bis 100.0
Output_PWM	0.0 bis 100.0

Im Fehlerfall setzt PID_Compact den Ausgangswert auf 0.0. Daher muss 0.0 immer innerhalb der Ausgangswertgrenzen liegen. Um eine Untergrenze Ausgangswert größer 0.0 zu erhalten, müssen Sie im Anwenderprogramm zu Output und Output_PER einen Offset addieren.

Siehe auch

Istwerteinstellung V1 (Seite 705)

Istwertüberwachung V1 (Seite 707)

PWM-Begrenzungen V1 (Seite 707)

PID-Parameter V1 (Seite 710)

PID-Parameter V1

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Hinweis

Die aktuell wirksamen PID-Parameter befinden sich bei PID_Compact V1 in der Struktur sRet und bei PID_Compact V2 in der Struktur Retain.CtrlParams.

Um ein Fehlverhalten des PID-Reglers zu vermeiden, ändern Sie online die aktuell wirksamen PID-Parameter nur in der Betriebsart "inaktiv".

Wenn Sie die PID-Parameter in den Betriebsarten "Automatikbetrieb" oder "Handbetrieb" online ändern möchten, dann ändern Sie die PID-Parameter folgendermaßen:

- PID_Compact V1: Ändern Sie die PID-Parameter in der Struktur sBackUp und übernehmen Sie diese Änderungen per sPid_Cmpt.b_LoadBackUp = TRUE in die Struktur sRet.
- PID_Compact V2: Ändern Sie die PID-Parameter in der Struktur CtrlParamsBackUp und übernehmen Sie diese Änderungen per LoadBackUp = TRUE in die Struktur Retain.CtrlParams.

Online Änderungen an den PID-Parametern in der Betriebsart "Automatikbetrieb" können zu Sprüngen am Ausgangswert führen.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

y	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient Differenzierverzug (Differenzierverzug T1 = a ×T_D)
T_D	Differenzierzeit
c	Gewichtung des D-Anteils

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:

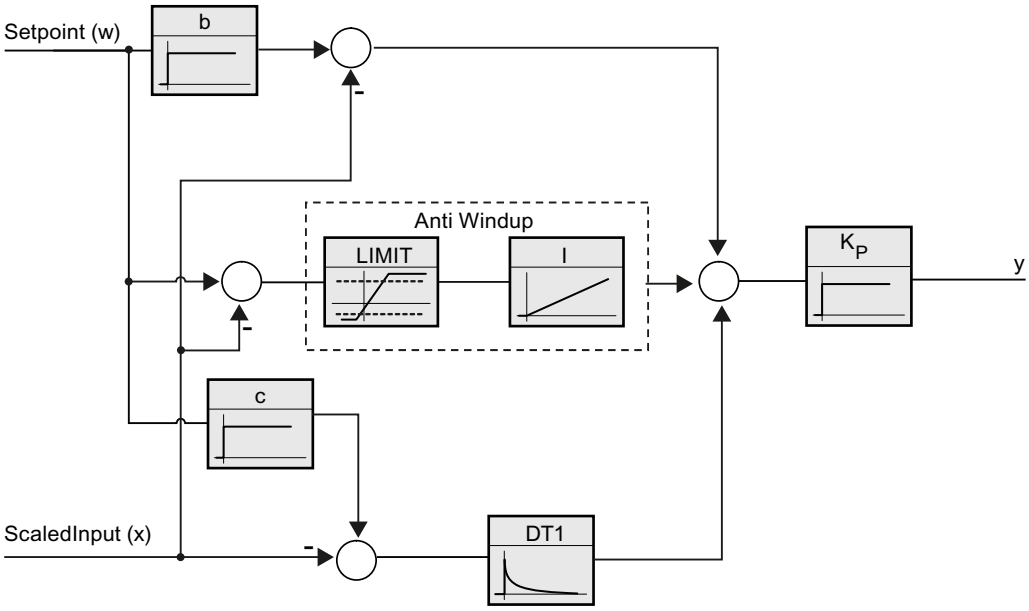

Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID_Compact vollständig laden.

Technologieobjekte in Gerät laden (Seite 656)

Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID_Compact arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0. Wenn die Integrationszeit in der Betriebsart "Automatikbetrieb" von einem anderen Wert auf 0.0 online geändert wird, dann wird der bisherige I-Anteil gelöscht und es erfolgt ein Sprung des Ausgangswerts.

Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

Koeffizient Differenzierverzug

Die Wirkung des D-Anteils wird durch den Koeffizent Differenzierverzug verzögert.

Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Abtastzeit PID-Algorithmus entspricht der Periodendauer der Pulsweitenmodulation. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

Regel für Optimierung

Wählen Sie in der Klappliste "Reglerstruktur" aus, ob PI oder PID-Parameter berechnet werden.

- **PID**
  Berechnet während der Erst- und Nachoptimierung PID-Parameter
- **PI**
  Berechnet während der Erst- und Nachoptimierung PI-Parameter
- **Benutzerdefiniert**
  Wenn Sie über ein Anwenderprogramm unterschiedliche Reglerstrukturen für Erst- und Nachoptimierung eingestellt haben, wird "Benutzerdefiniert" in der Klappliste angezeigt.

Siehe auch

Wahl der Reglerstruktur bei gegebener Regelstrecke (Seite 649)

PID_Compact V1 in Betrieb nehmen

Inbetriebnahme V1

Das Inbetriebnahmefenster hilft Ihnen bei der Inbetriebnahme des PID-Reglers. In der Kurvenanzeige können Sie die Werte für Sollwert, Istwert und Ausgangswert entlang der Zeitachse beobachten. Folgende Funktionen werden im Inbetriebnahmefenster unterstützt:

- Erstoptimierung des Reglers
- Nachoptimierung des Reglers
  Verwenden Sie die Nachoptimierung, wenn Sie eine Feinjustage der PID-Parameter wünschen.
- Beobachtung der laufenden Regelung im Kurvenfenster
- Test der Regelstrecke durch Vorgabe eines manuellen Ausgangswerts

Für alle Funktionen muss eine Onlineverbindung zur CPU aufgebaut sein.

Grundsätzliche Bedienung

- Wählen Sie in der Klappliste "Aktualisierungszeit" die gewünschte Aktualisierungszeit aus. Alle Werte im Inbetriebnahmefenster werden in der ausgewählten Aktualisierungszeit aktualisiert.
- Klicken Sie auf das Symbol "Start" in der Gruppe Messung, wenn Sie die Inbetriebnahmefunktionen nutzen möchten.
  Die Aufzeichnung der Werte wird gestartet. In der Kurvenanzeige werden die aktuellen Werte für Sollwert, Istwert und Ausgangswert eingetragen. Die Bedienung des Inbetriebnahmefensters wird freigegeben.
- Klicken Sie auf das Symbol "Stop", wenn Sie die Inbetriebnahmefunktionen beenden möchten.
  Die in der Kurvenanzeige aufgezeichneten Werte können weiterhin analysiert werden.

Mit Schließen des Inbetriebnahmefensters wird die Aufzeichnung in der Kurvenanzeige beendet und die aufgezeichneten Werte werden gelöscht.

Siehe auch

Erstoptimierung V1 (Seite 714)

Nachoptimierung V1 (Seite 715)

Betriebsart "Handbetrieb" V1 (Seite 717)

Erstoptimierung V1

Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Voraussetzung

- Die Anweisung "PID_Compact" wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- PID_Compact befindet sich in der Betriebsart "Inaktiv" oder "Handbetrieb".
- Der Sollwert darf während der Optimierung nicht geändert werden. Sonst wird PID_Compact deaktiviert.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz Obergrenze Istwert und Untergrenze Istwert.
- Der Abstand zwischen Sollwert und Istwert ist > 50% des Sollwerts.

Vorgehen

Um die "Erstoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Compact > Inbetriebnahme".
2. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Die Erstoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, wechselt PID_Compact in die Betriebsart "Inaktiv".

Siehe auch

Inbetriebnahme V1 (Seite 713)

Nachoptimierung V1 (Seite 715)

Betriebsart "Handbetrieb" V1 (Seite 717)

Nachoptimierung V1

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung.

PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Voraussetzung

- Die Anweisung PID_Compact wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- Der Sollwert darf während der Optimierung nicht geändert werden.
- PID_Compact befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

Ablauf abhängig von Startsituation

Die Nachoptimierung können Sie aus den Betriebsarten "Inaktiv", "Automatikbetrieb" oder "Handbetrieb" starten. Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb
  Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.
  PID_Comact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb
  Wenn die Voraussetzungen für eine Erstoptimierung erfüllt sind, wird eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstoptimierung nicht möglich, wechselt PID_Compact in die Betriebsart "Inaktiv".
  Wenn sich der Istwert für eine Erstoptimierung bereits zu nah am Sollwert befindet, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.

Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Der Ablauf der Nachoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Nachoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der "Nachoptimierung" Fehler auftraten, wechselt PID_Compact in die Betriebsart "Inaktiv".

Siehe auch

Inbetriebnahme V1 (Seite 713)

Erstoptimierung V1 (Seite 714)

Betriebsart "Handbetrieb" V1 (Seite 717)

Betriebsart "Handbetrieb" V1

Im Folgenden wird beschrieben, wie Sie die Betriebsart "Handbetrieb" im Inbetriebnahmefenster des Technologieobjekts "PID Compact" nutzen können.

Voraussetzung

- Die Anweisung "PID_Compact" wird in einem Weckalarm-OB aufgerufen.
- Es wurde eine Onlineverbindung zur CPU aufgebaut und die CPU befindet sich im Betriebzustand "RUN".
- Die Funktionen des Inbetriebnahmefensters sind über das Symbol "Messung Ein" freigegeben.

Vorgehen

Verwenden Sie im Inbetriebnahmefenster "Handbetrieb", wenn Sie die Regelstrecke durch Vorgabe eines Handwerts testen möchten. Um einen Handwert vorzugeben, gehen Sie folgendermaßen vor:

1. Aktivieren Sie im Bereich "Online-Zustand des Relgers" das Kontrollkästchen "Handbetrieb".
   PID_Compact arbeitet im Handbetrieb. Der zuletzt aktuelle Ausgangswert bleibt aktiv.
2. Tragen Sie im Feld "Output" den Handwert in der Einheit % ein.
3. Klicken Sie auf das Steuersymbol .

Ergebnis

Der Handwert wird in die CPU geschrieben und ist sofort wirksam.

Hinweis

PID_Compact überwacht den Istwert weiterhin. Wenn die Istwertgrenzen überschritten werden, wird PID_Compact deaktiviert.

Deaktivieren Sie das Kontrollkästchen "Handbetrieb", wenn der Ausgangswert wieder durch den PID-Regler vorgegeben werden soll. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

Siehe auch

Inbetriebnahme V1 (Seite 713)

Erstoptimierung V1 (Seite 714)

Nachoptimierung V1 (Seite 715)

PID_Compact V1 mit PLCSIM simulieren

Hinweis

Simulation mit PLCSIM

Bei der Simulation mit PLCSIM ist das zeitliche Verhalten der simulierten PLC nicht exakt identisch zu einer "echten" PLC. Der tatsächliche Zeittakt eines Weckalarm-OB kann bei einer simulierten PLC größere Schwankungen aufweisen als bei "echten" PLCs.

In der Standardkonfiguration ermittelt PID_Compact die Zeit zwischen den Aufrufen automatisch und überwacht diese auf Schwankungen.

Bei der Simulation von PID_Compact mit PLCSIM kann deshalb ein Abtastzeitfehler (ErrorBits = DW#16#00000800) erkannt werden.

PID_Compact wechselt in diesem Fall in die Betriebsart Inaktiv (State = 0)..

Um dies zu verhindern, sollten Sie PID_Compact bei Simulation mit PLCSIM wie folgt konfigurieren:

- sb_EnCyclEstimation = FALSE
- sb_EnCyclMonitoring = FALSE
- sPid_Calc.r_Cycle: Weisen Sie dieser Variablen den Zeittakt des aufrufenden Weckalarm-OB in der Einheit Sekunden zu.

3.3.2 PID_3Step einsetzen

3.3.2.1 Technologieobjekt PID_3Step

Das Technologieobjekt PID_3Step stellt einen PID-Regler mit Optimierung für Ventile oder Stellglieder mit integrierendem Verhalten zur Verfügung.

Sie können folgende Regler konfigurieren:

- Dreipunktschrittregler mit Stellungsrückmeldung
- Dreipunktschrittregler ohne Stellungsrückmeldung
- Ventilregler mit analogem Ausgangswert

PID_3Step erfasst innerhalb eines Regelkreises fortlaufend den gemessenen Istwert und vergleicht diesen mit dem Sollwert. Aus der sich ergebenden Regeldifferenz errechnet

PID_3Step einen Ausgangswert, durch den der Istwert den Sollwert möglichst schnell und stabil erreicht. Der Ausgangswert setzt sich beim PID-Regler aus drei Anteilen zusammen:

- **P**-Anteil
  Der P-Anteil des Ausgangswerts steigt proportional zur Regeldifferenz.
- **I**-Anteil
  Der I-Anteil des Ausgangswerts steigt solange bis die Regeldifferenz ausgeglichen ist.
- **D**-Anteil
  Der D-Anteil steigt mit wachsender Änderungsgeschwindigkeit der Regeldifferenz. Der Istwert wird möglichst schnell an den Sollwert angeglichen. Nimmt die Änderungsgeschwindigkeit der Regeldifferenz wieder ab, so verringert sich der D-Anteil wieder.

Die Anweisung PID_3Step berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der Erstoptimierung. Die Parameter können über eine Nachoptimierung weiter optimiert werden. Sie brauchen die Parameter nicht manuell ermitteln.

Weitere Informationen

- Übersicht der Software-Regler (Seite 651)
- Technologieobjekte hinzufügen (Seite 653)
- Technologieobjekte konfigurieren (Seite 654)
- PID_3Step V2 konfigurieren (Seite 720)
- PID_3Step V1 konfigurieren (Seite 740)

Prinzip

Weitere Informationen hierzu finden Sie im folgenden FAQ im Siemens Industry Online Support:

- Beitrags-ID 68011827 (https://support.industry.siemens.com/cs/ww/de/view/68011827)

3.3.2.2 PID_3Step V2

PID_3Step V2 konfigurieren

Grundeinstellungen V2

Einleitung V2

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID_3Step:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset

- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)
- Stellungsrückmeldung (nur im Inspektorfenster)

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB
  Es wird der Wert verwendet der im Instanz-DB gespeichert ist.
  Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.
  An der Anweisung darf kein Wert stehen.
  Änderung über HMI möglich.
- Anweisung
  Es wird der Wert verwendet, der an der Anweisung verschaltet ist.
  Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.
  Änderung über HMI nicht möglich.

Regelungsart V2

Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Sollwert, Istwert und Störgröße. Sollwert, Istwert und Störgröße werden in dieser Einheit angezeigt.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID_3Step arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktiveren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

Anlaufverhalten

1. Um nach Neustart der CPU in die Betriebsart "Inaktiv" zu wechseln, deaktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
   Um nach Neustart der CPU in die Betriebsart zu wechseln, die an Mode gespeichert ist, aktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
2. Wählen Sie in der Klappliste "Mode setzen auf" die Betriebsart aus, die nach einem vollständigen Laden in Gerät aktiviert werden soll.
   Nach einem vollständigen Laden in Gerät startet PID_3Step in der gewählten Betriebsart. Bei jedem weiteren Neustart, startet PID_3Step in der Betriebsart, die zuletzt an Mode gespeichert war.

Beispiel

Sie haben das Optionskästchen "Nach CPU Neustart Mode aktivieren" aktiviert und in der Liste "Mode setzen auf" den Eintrag "Erstoptimierung" gewählt. Nach einem vollständigen Laden in Gerät startet PID_3Step in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung noch aktiv ist, startet PID_3Step nach dem Neustart der CPU wieder in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung erfolgreich beendet wurde und der Automatikbetrieb aktiv ist, startet PID_3Step nach dem Neustart der CPU im "Automatikbetrieb".

Sollwert V2

Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.
   Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

Istwert V2

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID_3Step den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

Stellungsrückmeldung V2

Die Konfiguration der Stellungsrückmeldung ist abhängig vom eingesetzten Stellglied.

- Stellglied ohne Stellungsrückmeldung
- Stellglied mit digitalen Anschlagssignalen
- Stellglied mit analoger Stellungsrückmeldung
- Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Stellglied ohne Stellungsrückmeldung

Um PID_3Step für ein Stellglied ohne Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".

Stellglied mit digitalen Anschlagssignalen

Um PID_3Step für ein Stellglied mit Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".
2. Aktivieren Sie das Optionskästchen "Anschlagsignale Stellglied".
3. Wählen Sie für Actuator_H und Actuator_L als Quelle "Anweisung".
4. Geben Sie für Actuator_H und Actuator_L die Adressen der Digitaleingänge ein.

Stellglied mit analoger Stellungsrückmeldung

Um PID_3Step für ein Stellglied mit analoger Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback_PER".
   - Bei Feedback_PER verwenden Sie den Wert des Analogeingangs direkt. Die Skalierung von Feedback_PER konfigurieren Sie in den Stellgliedeinstellungen.
   - Bei Feedback bereiten Sie den Wert des Analogeingangs über Ihr Anwenderprogramm auf.
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.

Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Um PID_3Step für ein Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.
4. Aktivieren Sie das Optionskästchen "Anschlagsignale Stellglied".
5. Wählen Sie für Actuator_H und Actuator_L als Quelle "Anweisung".
6. Geben Sie für Actuator_H und Actuator_L die Adressen der Digitaleingänge ein.

Ausgangswert V2

PID_3Step stellt einen analogen Ausgangswert (Output_PER) und digitale Ausgangswerte (Output_UP, Output_DN) zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output_PER
  Das Stellglied verfügt über eine relevante Motorstellzeit und wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10 V oder 4...20 mA. Der Wert an Output_PER entspricht der Zielstellung des Ventils z. B. Output_PER = 13824, wenn das Ventil zu 50 % geöffnet werden soll.
  PID_3Step berücksichtigt z. B. für Selbstoptimierung und Anti-Windup-Verhalten, dass der analoge Ausgangswert wegen der Motorstellzeit erst verzögert auf den Prozess wirkt. Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.
- Output_UP, Output_DN
  Das Stellglied verfügt über eine relevante Motorstellzeit und wird über zwei Digitalausgänge gesteuert.
  Output_UP verfährt das Ventil in Richtung geöffneter Zustand.
  Output_DN verfährt das Ventil in Richtung geschlossener Zustand.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der Selbstoptimierung und dem Anti-Windup-Verhalten für eine korrekte Arbeitsweise notwendig. Konfigurieren Sie deshalb die Motorstellzeit unter "Stellgliedeinstellungen" mit dem Wert, den der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen.

Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (digital)".
2. Wählen Sie für Output_UP und Output_DN "Anweisung".
3. Geben Sie die Adressen der Digitalausgänge ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den zum Stellglied passenden Eintrag.
2. Wählen Sie "Anweisung".
3. Geben Sie den Namen der Variablen an, die Sie für die Aufbereitung des Ausgangswerts verwenden.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Analog- oder Digitalausgang der CPU zum Stellglied.

Istwerteinstellungen V2

Istwert skalieren V2

Wenn Sie in der Grundeinstellung die Verwendung von Input_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input_PER anhand eines unteren und oberen Wertepaars skaliert.

Vorgehen

Um den Istwert zu skalieren, gehen Sie folgendermaßen vor:

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID_3Step.
2. Verschalten Sie in den Grundeinstellungen Input_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

Istwertgrenzen V2

Als Grenzwerte müssen Sie für Ihre Regelstrecke eine sinnvolle absolute Ober- und Untergrenze des Istwerts festlegen. Sobald diese Grenzen über- oder unterschritten werden, tritt ein Fehler auf (ErrorBits = 0001h). Die Optimierung wird abgebrochen, wenn die Istwertgrenzen übrschritten werden. Wie PID_3Step im Fehlerfall im Automatikbetrieb reagiert, legen Sie bei den Stellgliedeinstellungen fest.

Stellgliedeinstellungen V2

Stellglied V2

Stellglied spezifische Zeiten

Um das Stellglied vor Beschädigung zu schützen, konfigurieren Sie die Motorstellzeit, die minimale Einschaltzeit und die minimale Ausschaltzeit. Die Daten finden Sie im Datenblatt des Stellglieds.

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Die Motorstellzeit kann während der Inbetriebnahme ermittelt werden.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der Selbstoptimierung und beim Anti-Windup-Verhalten für eine korrekte Arbeitsweise erforderlich.

Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.

Die Motorstellzeit ist remanent. Wenn Sie die Motorstellzeit manuell ändern, müssen Sie PID_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 656)

Wenn Sie Output_UP und Output_DN verwenden, verringern Sie die Schalthäufigkeit mit der minimalen Ein- und Ausschaltzeit.

Im Automatikbetrieb werden die berechneten Ein- oder Ausschaltzeiten summiert und erst wirksam, wenn die Summe größer oder gleich der minimalen Ein- oder Ausschaltzeit ist.

Im Handbetrieb wird das Stellglied durch Manual_UP = TRUE oder Manual_DN = TRUE mindestens um die minimale Ein- oder Ausschaltzeit angesteuert.

Falls Sie in den Grundeinstellungen den analogen Ausgangswert Output_PER gewählt haben, werden die minimale Einschaltzeit und die minimale Ausschaltzeit nicht ausgewertet und sind nicht änderbar.

Verhalten im Fehlerfall

PID_3Step ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn Sie im Fehlerfall "Aktueller Wert für die Fehlerdauer" oder "Ersatzausgangswert für die Fehlerdauer " ausgeben, bleibt PID_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

Im Fehlerfall gibt PID_3Step einen konfigurierbaren Ausgangswert aus:

- Aktueller Wert
  PID_3Step ist ausgeschaltet und verändert die Position des Stellglieds nicht mehr.
- Aktueller Wert für die Fehlerdauer
  Die Reglerfunktionen von PID_3Step sind ausgeschaltet und die Position des Stellglieds wird nicht mehr verändert.
  Wenn im Automatikbetrieb folgende Fehler aufgetreten sind, kehrt PID_3Step in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.
  - 0002h: Ungültiger Wert am Parameter Input_PER.
  - 0200h: Ungültiger Wert am Parameter Input.
  - 0400h: Berechnung des Ausgangswerts fehlgeschlagen.
  - 1000h: Ungültiger Wert am Parameter Setpoint.
  - 2000h: Ungültiger Wert am Parameter Feedback_PER.
  - 4000h: Ungültiger Wert am Parameter Feedback.
  - 8000h: Fehler bei der digitalen Stellungsrückmeldung.
  - 20000h: Ungültiger Wert an der Variablen SavePosition.

  Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_3Step im Automatikbetrieb:
  - 0001h: Der Parameter Input ist außerhalb der Istwertgrenzen.
  - 0800h: Abtastzeitfehler
  - 40000h: Ungültiger Wert am Parameter Disturbance.

  Wenn im Handbetrieb ein Fehler auftritt, bleibt PID_3Step im Handbetrieb.
  Wenn während der Optimierung oder Stellzeitmessung ein Fehler auftritt, wechselt PID_3Step in die Betriebsart, aus der die Optimierung oder Stellzeitmessung gestartet wurde. Nur bei folgendem Fehler wird die Optimierung nicht abgebrochen:
  - 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt.
- Ersatzausgangswert
  PID_3Step fährt das Stellglied auf den Ersatzausgangswert und schaltet sich aus.
- Ersatzausgangswert für die Fehlerdauer
  PID_3Step fährt das Stellglied auf den Ersatzausgangswert. Nach dem Erreichen des Ersatzausgangswerts verhält sich PID_3Step wie bei "Aktueller Wert für die Fehlerdauer" beschrieben.

Den Ersatzausgangswert tragen Sie in "%" ein.

Bei Stellgliedern ohne analoge Stellungsrückmeldung können nur die Ersatzausgangswerte 0 % und 100 % exakt angefahren werden. Ein Ersatzausgangswert ungleich 0 % oder 100 % wird über eine intern simulierte Stellungsrückmeldung angefahren. Mit diesem Verfahren kann der Ersatzausgangswert aber nie exakt angefahren werden.

Bei Stellgliedern mit analoger Stellungsrückmeldung können alle Ersatzausgangswerte exakt angefahren werden.

Stellungsrückmeldung skalieren V2

Stellungsrückmeldung skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Feedback_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in % umrechnen. Im Anzeigefeld "Feedback" wird die aktuelle Konfiguration angezeigt.

Feedback_PER wird anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Unterer Anschlag" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Oberer Anschlag" und "Oben" das obere Wertepaar ein.

"Unterer Anschlag" muss kleiner sein als "Oberer Anschlag"; "Unten" muss kleiner sein als "Oben".

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- Kein Feedback, Feedback, Feedback_PER
- Output (analog), Output (digital)

Output	Feedback	Unterer Anschlag	Oberer Anschlag
Output (digital)	Kein Feedback	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
Output (digital)	Feedback	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (digital)	Feedback_PER	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (analog)	Kein Feedback	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
Output (analog)	Feedback	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (analog)	Feedback_PER	-100.0 % oder 0.0 %	0.0 % oder +100.0 %

Ausgangswertgrenzen V2

Ausgangswert begrenzen

Während der Stellzeitmessung und bei Mode = 10 können die Ausgangswertgrenzen über- oder unterschritten werden. In allen anderen Betriebsarten wird der Ausgangswert auf diese Werte begrenzt.

Geben Sie in den Eingabefeldern "Obergrenze Ausgangswert" und "Untergrenze Ausgangswert" die absoluten Ausgangswertgrenzen ein. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

Wenn kein Feedback vorhanden ist und Output (digital) eingestellt ist, können Sie den Ausgangswert nicht begrenzen. Output_UP und Output_DN werden dann bei Actuator_H = TRUE oder Actuator_L = TRUE zurück gesetzt. Wenn keine Anschlagsignale vorhanden sind, werden Output_UP und Output_DN nach einer Verfahrzeit von 150% der Motorstellzeit zurück gesetzt.

Der Default-Wert 150% kann über die Variable Config.VirtualActuatorLimit angepasst werden. Ab PID_3Step Version 2.3 kann die Überwachung und Begrenzung der Verfahrzeit mit Config.VirtualActuatorLimit = 0.0 deaktiviert werden.

Erweiterte Einstellungen V2

Istwertüberwachung V2

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID_3Step eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID_3Step verhält sich folgendermaßen:

Istwert	InputWarning_H	InputWarning_L	ErrorBits	Betriebsart
> 98 °C	TRUE	FALSE	0001h	Wie konfiguriert
≤ 98 °C und > 90 °C	TRUE	FALSE	0000h	Automatikbetrieb
≤ 90 °C und ≥ 10 °C	FALSE	FALSE	0000h	Automatikbetrieb
< 10°C und ≥ 0 °C	FALSE	TRUE	0000h	Automatikbetrieb
< 0 °C	FALSE	TRUE	0001h	Wie konfiguriert

Wie PID_3Step beim Über- oder Unterschreiten der Obergrenze bzw. Untergrenze Istwert reagiert, konfigurieren Sie bei den Stellgliedeinstellungen.

PID-Parameter V2

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Hinweis

Die aktuell wirksamen PID-Parameter befinden sich in der Struktur Retain.CtrlParams.

Um ein Fehlverhalten des PID-Reglers zu vermeiden, ändern Sie online die aktuell wirksamen PID-Parameter nur in der Betriebsart "inaktiv".

Wenn Sie die PID-Parameter in den Betriebsarten "Automatikbetrieb" oder "Handbetrieb" online ändern möchten, dann ändern Sie die PID-Parameter in der Struktur CtrlParamsBackUp und übernehmen diese Änderungen in die Struktur Retain.CtrlParams folgendermaßen:

- PID_3Step V1: Übernehmen Sie die Änderungen per Config.LoadBackUp = TRUE
- PID_3Step V2: Übernehmen Sie die Änderungen per LoadBackUp = TRUE

Online Änderungen an den PID-Parametern in der Betriebsart "Automatikbetrieb" können zu Sprüngen am Ausgangswert führen.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Δy	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)
T_D	Differenzierzeit
c	Gewichtung des D-Anteils

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:

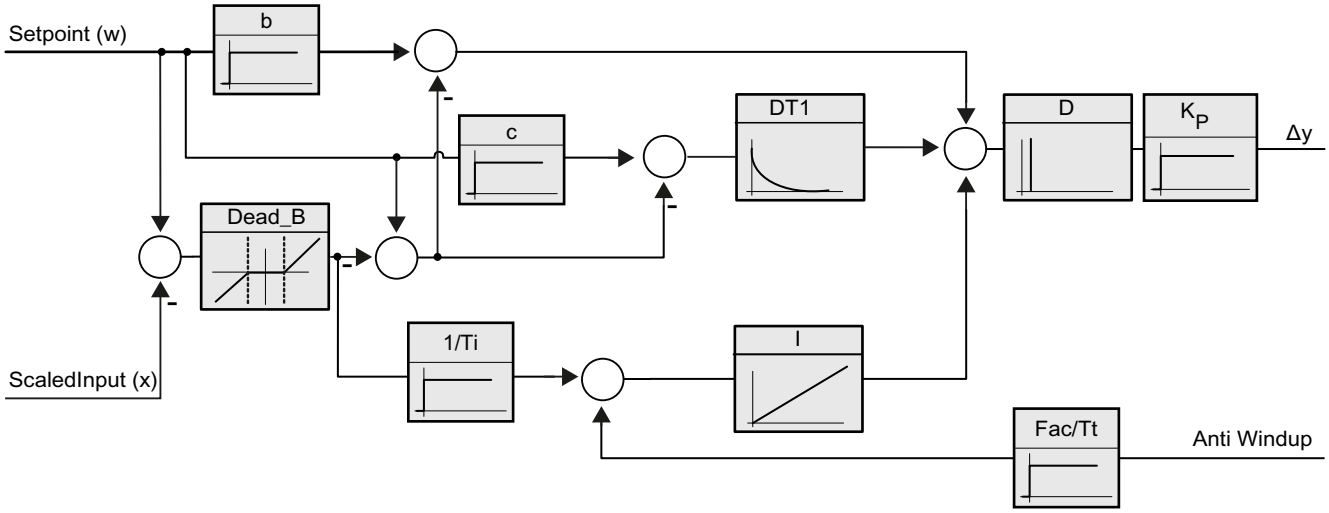

Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 656)

Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID_3Step arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0. Wenn die Integrationszeit in der Betriebsart "Automatikbetrieb" von einem anderen Wert auf 0.0 online geändert wird, dann wird der bisherige I-Anteil gelöscht und es erfolgt ein Sprung des Ausgangswerts.

Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

Koeffizient Differenzierverzug

Die Wirkung des D-Anteils wird durch den Koeffizent Differenzierverzug verzögert.

Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Abtastzeit PID_3Step gerundet. Alle anderen Funktionen von PID_3Step werden bei jedem Aufruf durchgeführt.

Totzonenbreite

Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Falls für die Gewichtung des P-Anteils oder die Gewichtung des D-Anteils Werte ungleich 1.0 konfiguriert sind, wirken sich Sollwertänderungen auch innerhalb der Totzone auf den Ausgangswert aus.
Istwertänderungen innerhalb der Totzone wirken sich unabhängig von den Gewichtungen nicht auf den Ausgangswert aus.

Siehe auch

Wahl der Reglerstruktur bei gegebener Regelstrecke (Seite 649)

PID_3Step V2 in Betrieb nehmen

Erstoptimierung V2

Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Erstoptimierung eingefroren.

Voraussetzung

- Die Anweisung PID_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- PID_3Step befindet sich in der Betriebsart "Inaktiv", "Handbetrieb" oder "Automatikbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

Vorgehen

Um die Erstoptimierung durchzuführen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Inbetriebnahme".
2. Wählen Sie im Arbeitsbereich "Optimierung" in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Die Erstoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, verhält sich PID_3Step wie unter Verhalten im Fehlerfall konfiguriert.

Nachoptimierung V2

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Nachoptimierung eingefroren.

Voraussetzung

- Die Anweisung PID_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID_3Step befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

Ablauf abhängig von Startsituation

Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb
  Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.
  PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb
  Zuerst wird immer eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.

Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Der Ablauf der Nachoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wenn während der Nachoptimierung keine Fehler auftraten, wurden die PID-Parameter optimiert. PID_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der Nachoptimierung Fehler auftraten, verhält sich PID_3Step wie unter Verhalten im Fehlerfall konfiguriert.

Mit manuellen PID-Parametern in Betrieb nehmen V2

Voraussetzung

- Die Anweisung PID_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- PID_3Step befindet sich in der Betriebsart "Inaktiv".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

Vorgehen

Um PID_3Step mit manuellen PID-Parametern in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Konfiguration".
2. Klicken Sie im Konfigurationsfenster auf "Erweiterte Einstellungen > PID-Parameter".
3. Aktivieren Sie das Optionskästchen "Manuelle Eingabe aktivieren".
4. Geben Sie die PID-Parameter ein.
5. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Inbetriebnahme".
6. Stellen Sie eine Online-Verbindung zur CPU her.
7. Laden Sie die PID-Parameter auf die CPU.
8. Klicken Sie auf das Symbol "Start PID_3Step".

Ergebnis

PID_3Step wechselt in den Automatikbetrieb und regelt mit den aktuellen PID-Parametern.

Siehe auch

PID-Parameter V2 (Seite 731)

Motorstellzeit messen V2

Einleitung

PID_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

Voraussetzung

- In den Grundeinstellungen ist Feedback oder Feedback_PER ausgewählt und das Signal ist verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

1. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
2. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Die Differenz zwischen "Zielstellung" und "Stellungsrückmeldung" muss mindestens 50 % des zulässigen Ausgangswertbereichs sein.
3. Klicken Sie auf das Symbol "Start".

Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

Motorstellzeit = (Obergrenze Ausgangswert – Untergrenze Ausgangswert) × Messzeit / BETRAG(Zielstellung – Startstellung).

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart, aus der die Stellzeitmessung gestartet wurde. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = FALSE ist, wechselt PID_3Step in die Betriebsart "Inaktiv".

Hinweis

Um die gemessene Motorstellzeit in das Projekt zu übernehmen, klicken Sie auf das Symbol "Gemessene Stellzeit laden".

Stellglieder mit Anschlagsignalen

Um die Stellzeit von Stellgliedern mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

Voraussetzung

- In den Grundeinstellungen ist das Optionskästchen "Anschlagsignale" aktiviert und Actuator_H und Actuator_L sind verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
2. Wählen Sie die Richtung, in die das Stellglied bewegt werden soll.
   - Öffnen - Schließen - Öffnen
     Das Stellglied wird zuerst bis zum oberen Anschlag gefahren, dann zum unteren Anschlag und erneut zum oberen Anschlag.
   - Schließen - Öffnen - Schließen
     Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag.
3. Klicken Sie auf das Symbol "Start".

Ergebnis

Das Stellglied wird in der gewählten Richtung bewegt. Die Zeitmessung startet, wenn das Stellglied den ersten Anschlag erreicht hat und endet, wenn das Stellglied diesen Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart, aus der die Stellzeitmessung gestartet wurde. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = FALSE ist, wechselt PID_3Step in die Betriebsart "Inaktiv".

Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung mit der Schaltfläche Stop abbrechen, wechselt PID_3Step in die Betriebsart "Inaktiv".

PID_3Step V2 mit PLCSIM simulieren

Hinweis

Simulation mit PLCSIM

Die Simulation von PID_3Step V2.x mit PLCSIM für CPU S7-1200 wird nicht unterstützt.

PID_3Step V2.x kann nur für CPU S7-1500 mit PLCSIM simuliert werden.

Bei der Simulation mit PLCSIM ist das zeitliche Verhalten der simulierten PLC nicht exakt identisch zu einer "echten" PLC. Der tatsächliche Zeittakt eines Weckalarm-OB kann bei einer simulierten PLC größere Schwankungen aufweisen als bei "echten" PLCs.

In der Standardkonfiguration ermittelt PID_3Step die Zeit zwischen den Aufrufen automatisch und überwacht diese auf Schwankungen.

Bei der Simulation von PID_3Step mit PLCSIM kann deshalb ein Abtastzeitfehler (ErrorBits = DW#16#00000800) erkannt werden.

Dies führt zum Abbruch von laufenden Optimierungen.

Die Reaktion im Automatikbetrieb ist abhängig vom Wert der Variable ActivateRecoverMode.

Um dies zu verhindern, sollten Sie PID_3Step bei Simulation mit PLCSIM wie folgt konfigurieren:

- CycleTime.EnEstimation = FALSE
- CycleTime.EnMonitoring = FALSE
- CycleTime.Value: Weisen Sie dieser Variablen den Zeittakt des aufrufenden Weckalarm-OB in der Einheit Sekunden zu.

3.3.2.3 PID_3Step V1

PID_3Step V1 konfigurieren

Grundeinstellungen V1

Einleitung V1

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID_3Step:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)
- Stellungsrückmeldung (nur im Inspektorfenster)

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB
  Es wird der Wert verwendet der im Instanz-DB gespeichert ist.
  Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.
  An der Anweisung darf kein Wert stehen.
  Änderung über HMI möglich.
- Anweisung
  Es wird der Wert verwendet, der an der Anweisung verschaltet ist.
  Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.
  Änderung über HMI nicht möglich.

Regelungsart V1

Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Soll- und Istwert. Soll- und Istwert werden in dieser Einheit angezeigt.

Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID_3Step arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktiveren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

Anlaufverhalten nach Reset

Um nach Neustart der CPU sofort in die zuletzt aktive Betriebsart zu wechseln, aktivieren Sie das Optionskästchen "Nach CPU Neustart letzte Betriebsart aktivieren".

Wenn das Optionskästchen deaktiviert ist, bleibt der PID_3Step in der Betriebsart "Inaktiv".

Sollwert V1

Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.
   Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

Istwert V1

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID_3Step den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

Stellungsrückmeldung V1

Die Konfiguration der Stellungsrückmeldung ist abhängig vom eingesetzten Stellglied.

- Stellglied ohne Stellungsrückmeldung
- Stellglied mit digitalen Anschlagssignalen
- Stellglied mit analoger Stellungsrückmeldung
- Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Stellglied ohne Stellungsrückmeldung

Um PID_3Step für ein Stellglied ohne Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".

Stellglied mit digitalen Anschlagssignalen

Um PID_3Step für ein Stellglied mit Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".
2. Aktivieren Sie das Optionskästchen "Anschlagsignale Stellglied".
3. Wählen Sie für Actuator_H und Actuator_L als Quelle "Anweisung".
4. Geben Sie für Actuator_H und Actuator_L die Adressen der Digitaleingänge ein.

Stellglied mit analoger Stellungsrückmeldung

Um PID_3Step für ein Stellglied mit analoger Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback_PER".
   - Bei Feedback_PER verwenden Sie den Wert des Analogeingangs direkt. Die Skalierung von Feedback_PER konfigurieren Sie in den Stellgliedeinstellungen.
   - Bei Feedback bereiten Sie den Wert des Analogeingangs über Ihr Anwenderprogramm auf.
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.

Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Um PID_3Step für ein Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.
4. Aktivieren Sie das Optionskästchen "Anschlagsignale Stellglied".
5. Wählen Sie für Actuator_H und Actuator_L als Quelle "Anweisung".
6. Geben Sie für Actuator_H und Actuator_L die Adressen der Digitaleingänge ein.

Ausgangswert V1

PID_3Step stellt einen analogen Ausgangswert (Output_PER) und digitale Ausgangswerte (Output_UP, Output_DN) zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output_PER
  Das Stellglied verfügt über eine relevante Motorstellzeit und wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10 V oder 4...20 mA. Der Wert an Output_PER entspricht der Zielstellung des Ventils z. B. Output_PER = 13824, wenn das Ventil zu 50 % geöffnet werden soll.
  PID_3Step berücksichtigt z. B. für Selbstoptimierung und Anti-Windup-Verhalten, dass der analoge Ausgangswert wegen der Motorstellzeit erst verzögert auf den Prozess wirkt. Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.
- Output_UP, Output_DN
  Das Stellglied verfügt über eine relevante Motorstellzeit und wird über zwei Digitalausgänge gesteuert.
  Output_UP verfährt das Ventil in Richtung geöffneter Zustand.
  Output_DN verfährt das Ventil in Richtung geschlossener Zustand.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der Selbstoptimierung und dem Anti-Windup-Verhalten für eine korrekte Arbeitsweise notwendig. Konfigurieren Sie deshalb die Motorstellzeit unter "Stellgliedeinstellungen" mit dem Wert, den der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen.

Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (digital)".
2. Wählen Sie für Output_UP und Output_DN "Anweisung".
3. Geben Sie die Adressen der Digitalausgänge ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den zum Stellglied passenden Eintrag.
2. Wählen Sie "Anweisung".
3. Geben Sie den Namen der Variablen an, die Sie für die Aufbereitung des Ausgangswerts verwenden.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Analog- oder Digitalausgang der CPU zum Stellglied.

Istwerteinstellung V1

Konfigurieren Sie im Konfigurationsfenster "Istwerteinstellungen" die Normierung Ihres Istwerts und legen Sie die absoluten Istwertgrenzen fest.

Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input_PER anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID_3Step.
2. Verschalten Sie in den Grundeinstellungen Input_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung". Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

Istwert überwachen

Legen Sie die absolute Ober- und Untergrenze des Istwerts fest. Als Grenzwerte müssen Sie für Ihre Regelstrecke sinnvolle Werte eingeben. Während der Optimierung sind sinnvolle Grenzwerte wichtig, um optimale PID-Parameter zu erhalten. Die Vorbelegung der "Obergrenze Istwert" beträgt 120 %. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird mit dieser Einstellung kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert.

ACHTUNG
Ihre Anlage kann beschädigt werden. Wenn Sie als Istwertgrenzen sehr hohe Werte (z. B. $-3{,}4*10^{38}$...$+3{,}4*10^{38}$) einstellen, wird die Überwachung des Istwerts deaktiviert. Dann kann es durch einen Fehler zu Schäden an Ihrer Anlage kommen. Konfigurieren Sie für Ihre Regelstrecke sinnvolle Istwertgrenzen.

Stellgliedeinstellung V1

Stellglied spezifische Zeiten

Um das Stellglied vor Beschädigung zu schützen, konfigurieren Sie die Motorstellzeit, die minimale Einschaltzeit und die minimale Ausschaltzeit. Die Daten finden Sie im Datenblatt des Stellglieds.

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal 110 % der Motorstellzeit in eine Richtung bewegt. Die Motorstellzeit können Sie während der Inbetriebnahme messen.

Die Motorstellzeit wird sowohl bei der Berechnung des analogen Ausgangswerts als auch bei der Berechnung der digitalen Ausgangswerte berücksichtigt. Sie ist vor allem bei der Selbstoptimierung und beim Anti-Windup-Verhalten für eine korrekte Arbeitsweise erforderlich.

Ist in Ihrem Prozess keine relevante Motorstellzeit wirksam (z. B. bei Magnetventilen), sodass der Ausgangswert direkt und in vollem Umfang auf den Prozess wirkt, so verwenden Sie stattdessen PID_Compact.

Wenn Sie Output_UP und Output_DN verwenden, verringern Sie die Schalthäufigkeit mit der minimalen Ein- und Ausschaltzeit.

Im Automatikbetrieb werden die berechneten Ein- oder Ausschaltzeiten summiert und erst wirksam, wenn die Summe größer oder gleich der minimalen Ein- oder Ausschaltzeit ist.

Im Handbetrieb wird durch eine steigende Flanke an Manual_UP oder Manual_DN das Stellglied mindestens um die minimale Ein- oder Ausschaltzeit angesteuert.

Falls Sie in den Grundeinstellungen den analogen Ausgangswert Output_PER gewählt haben, werden die minimale Einschaltzeit und die minimale Ausschaltzeit nicht ausgewertet und sind nicht änderbar.

Verhalten im Fehlerfall

PID_3Step ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

Im Fehlerfall gibt PID_3Step einen konfigurierbaren Ausgangswert aus:

- Aktueller Wert
  PID_3Step ist ausgeschaltet und verändert die Position des Stellglieds nicht mehr.
- Aktueller Wert für die Fehlerdauer
  Die Reglerfunktionen von PID_3Step sind ausgeschaltet und die Position des Stellglieds wird nicht mehr verändert.
  Wenn im Automatikbetrieb folgende Fehler aufgetreten sind, kehrt PID_3Step in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.
  - 0002h: Ungültiger Wert am Parameter Input_PER.
  - 0200h: Ungültiger Wert am Parameter Input.
  - 0800h: Abtastzeitfehler
  - 1000h: Ungültiger Wert am Parameter Setpoint.
  - 2000h: Ungültiger Wert am Parameter Feedback_PER.
  - 4000h: Ungültiger Wert am Parameter Feedback.
  - 8000h: Fehler bei der digitalen Stellungsrückmeldung.

  Wenn im Handbetrieb einer dieser Fehler auftritt, bleibt PID_3Step im Handbetrieb.
  Wenn während der Optimierung oder Stellzeitmessung ein Fehler auftritt, wird PID_3Step ausgeschaltet.
- Ersatzausgangswert
  PID_3Step fährt das Stellglied auf den Ersatzausgangswert und schaltet sich aus.
- Ersatzausgangswert für die Fehlerdauer
  PID_3Step fährt das Stellglied auf den Ersatzausgangswert. Nach dem Erreichen des Ersatzausgangswerts verhält sich PID_3Step wie bei "Aktueller Wert für die Fehlerdauer" beschrieben.

Den Ersatzausgangswert tragen Sie in "%" ein.

Bei Stellgliedern ohne analoge Stellungsrückmeldung können nur die Ersatzausgangswerte 0 % und 100 % exakt angefahren werden. Damit der obere oder untere Anschlag erreicht wird, wird das Stellglied mit 110% der Motorstellzeit in eine Richtung gefahren. Die Anschlagsignale werden vorrangig berücksichtig. Ein Ersatzausgangswert ungleich 0 % oder 100 % wird über eine intern simulierte Stellungsrückmeldung angefahren. Mit diesem Verfahren kann der Ersatzausgangswert aber nie exakt angefahren werden.

Bei Stellgliedern mit analoger Stellungsrückmeldung können alle Ersatzausgangswerte exakt angefahren werden.

Stellungsrückmeldung skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Feedback_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in % umrechnen. Im Anzeigefeld "Feedback" wird die aktuelle Konfiguration angezeigt.

Feedback_PER wird anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Unterer Anschlag" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Oberer Anschlag" und "Oben" das obere Wertepaar ein.

"Unterer Anschlag" muss kleiner sein als "Oberer Anschlag"; "Unten" muss kleiner sein als "Oben".

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- Kein Feedback, Feedback, Feedback_PER
- Output (analog), Output (digital)

Output	Feedback	Unterer Anschlag	Oberer Anschlag
Output (digital)	Kein Feedback	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
Output (digital)	Feedback	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (digital)	Feedback_PER	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (analog)	Kein Feedback	nicht einstellbar (0.0 %)	nicht einstellbar (100.0 %)
Output (analog)	Feedback	-100.0 % oder 0.0 %	0.0 % oder +100.0 %
Output (analog)	Feedback_PER	-100.0 % oder 0.0 %	0.0 % oder +100.0 %

Ausgangswert begrenzen

Nur während der Stellzeitmessung können die Ausgangswertgrenzen über- oder unterschritten werden. In allen anderen Betriebsarten wird der Ausgangswert auf diese Werte begrenzt.

Geben Sie in den Eingabefeldern "Obergrenze Ausgangswert" und "Untergrenze Ausgangswert" die absoluten Ausgangswertgrenzen ein. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

Wenn kein Feedback vorhanden ist und Output (digital) eingestellt ist, können Sie den Ausgangswert nicht begrenzen. Die Digitalausgänge entweder bei Actuator_H = TRUE oder Actuator_L = TRUE oder nach einer Verfahrzeit von 110% der Motorstellzeit zurück gesetzt.

Erweiterte Einstellungen V1

Istwertüberwachung V1

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID_3Step eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID_3Step verhält sich folgendermaßen:

Istwert	InputWarning_H	InputWarning_L	Betriebsart
> 98 °C	TRUE	FALSE	Inaktiv
≤ 98 °C und > 90 °C	TRUE	FALSE	Automatikbetrieb
≤ 90 °C und ≥ 10 °C	FALSE	FALSE	Automatikbetrieb
< 10°C und ≥ 0 °C	FALSE	TRUE	Automatikbetrieb
< 0 °C	FALSE	TRUE	Inaktiv

PID-Parameter V1

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Hinweis

Die aktuell wirksamen PID-Parameter befinden sich in der Struktur Retain.CtrlParams.

Um ein Fehlverhalten des PID-Reglers zu vermeiden, ändern Sie online die aktuell wirksamen PID-Parameter nur in der Betriebsart "inaktiv".

Wenn Sie die PID-Parameter in den Betriebsarten "Automatikbetrieb" oder "Handbetrieb" online ändern möchten, dann ändern Sie die PID-Parameter in der Struktur CtrlParamsBackUp und übernehmen diese Änderungen in die Struktur Retain.CtrlParams folgendermaßen:

- PID_3Step V1: Übernehmen Sie die Änderungen per Config.LoadBackUp = TRUE
- PID_3Step V2: Übernehmen Sie die Änderungen per LoadBackUp = TRUE

Online Änderungen an den PID-Parametern in der Betriebsart "Automatikbetrieb" können zu Sprüngen am Ausgangswert führen.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Δy	Ausgangswert des PID-Algorithmus
K_p	Proportionalverstärkung
s	Laplace-Operator
b	Gewichtung des P-Anteils
w	Sollwert
x	Istwert
T_I	Integrationszeit
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)
T_D	Differenzierzeit
c	Gewichtung des D-Anteils

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:

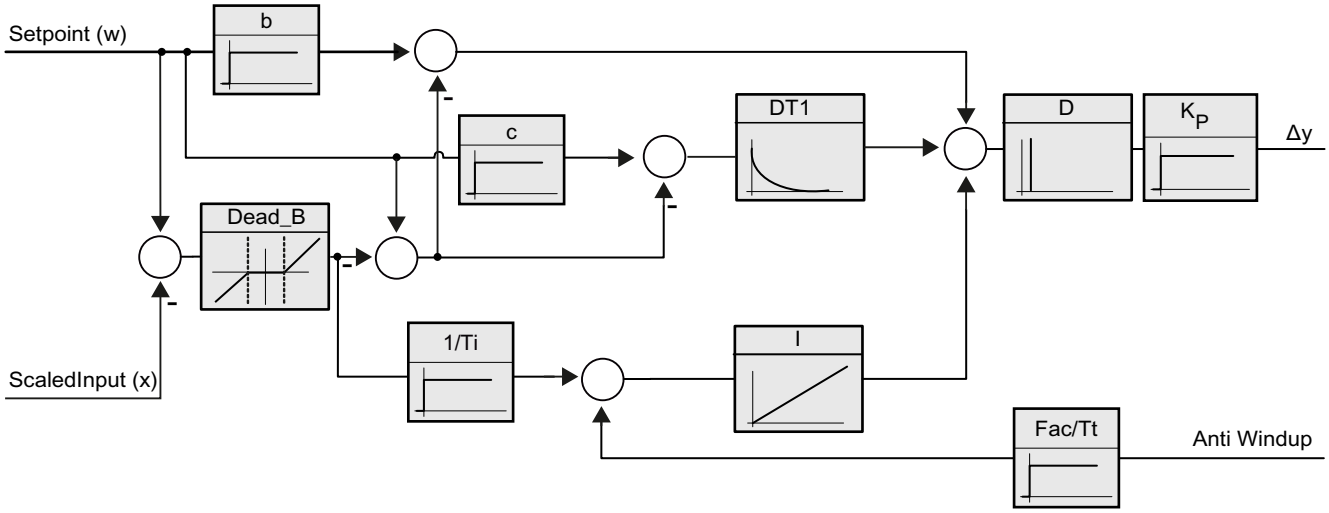

Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 656)

Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID_3Step arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0. Wenn die Integrationszeit in der Betriebsart "Automatikbetrieb" von einem anderen Wert auf 0.0 online geändert wird, dann wird der bisherige I-Anteil gelöscht und es erfolgt ein Sprung des Ausgangswerts.

Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

Koeffizient Differenzierverzug

Die Wirkung des D-Anteils wird durch den Koeffizent Differenzierverzug verzögert.

Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Abtastzeit PID_3Step gerundet. Alle anderen Funktionen von PID_3Step werden bei jedem Aufruf durchgeführt.

Totzonenbreite

Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Falls für die Gewichtung des P-Anteils oder die Gewichtung des D-Anteils Werte ungleich 1.0 konfiguriert sind, wirken sich Sollwertänderungen auch innerhalb der Totzone auf den Ausgangswert aus.
Istwertänderungen innerhalb der Totzone wirken sich unabhängig von den Gewichtungen nicht auf den Ausgangswert aus.

Siehe auch

Wahl der Reglerstruktur bei gegebener Regelstrecke (Seite 649)

PID_3Step V1 in Betrieb nehmen

Inbetriebnahme V1

Im Arbeitsbereich "Optimierung" können Sie Sollwert, Istwert und Ausgangswert in Abhängigkeit von der Zeit beobachten. Folgende Funktionen zur Inbetriebnahme werden im Kurvenschreiber unterstützt:

- Erstoptimierung des Reglers
- Nachoptimierung des Reglers
- Beobachtung der laufenden Regelung im Kurvenfenster

Für alle Funktionen muss eine Onlineverbindung zur CPU aufgebaut sein.

Grundsätzliche Bedienung

- Wählen Sie in der Klappliste "Aktualisierungszeit" die gewünschte Aktualisierungszeit aus. Alle Werte im Arbeitsbereich Optimierung werden in der ausgewählten Aktualisierungszeit aktualisiert.
- Klicken Sie auf das Symbol "Start" in der Gruppe Messung, wenn Sie die Inbetriebnahmefunktionen nutzen möchten.
  Die Aufzeichnung der Werte wird gestartet. In der Kurvenanzeige werden die aktuellen Werte für Sollwert, Istwert und Ausgangswert eingetragen. Die Bedienung des Inbetriebnahmefensters wird freigegeben.
- Klicken Sie auf das Symbol "Stop", wenn Sie die Inbetriebnahmefunktionen beenden möchten.
  Die in der Kurvenanzeige aufgezeichneten Werte können weiterhin analysiert werden.
- Mit Schließen des Inbetriebnahmefensters wird die Aufzeichnung in der Kurvenanzeige beendet und die aufgezeichneten Werte werden gelöscht.

Erstoptimierung V1

Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Erstoptimierung eingefroren.

Voraussetzung

- Die Anweisung PID_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE

- PID_3Step befindet sich in der Betriebsart "Inaktiv" oder "Handbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

Vorgehen

Um die Erstoptimierung durchzuführen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Inbetriebnahme".
2. Wählen Sie im Arbeitsbereich "Optimierung" in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Die Erstoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie auf das Symbol "Stop" wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, wechselt PID_3Step in die Betriebsart "Inaktiv".

Nachoptimierung V1

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung.

PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Nachoptimierung eingefroren.

Voraussetzung

- Die Anweisung PID_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID_3Step befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

Ablauf abhängig von Startsituation

Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb
  Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.
  PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb
  Zuerst wird immer eine Erstoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.

Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
   - Eine Online-Verbindung wird aufgebaut.
   - Die Aufzeichnung der Werte wird gestartet.
   - Der Ablauf der Nachoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Nachoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der Nachoptimierung Fehler auftraten, wechselt PID_3Step in die Betriebsart "Inaktiv".

Mit manuellen PID-Parametern in Betrieb nehmen V1

Vorgehen

Um PID_3Step mit manuellen PID-Parametern in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Konfiguration".
2. Klicken Sie im Konfigurationsfenster auf "Erweiterte Einstellungen > PID-Parameter".
3. Aktivieren Sie das Optionskästchen "Manuelle Eingabe aktivieren".
4. Geben Sie die PID-Parameter ein.
5. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_3Step > Inbetriebnahme".
6. Stellen Sie eine Online-Verbindung zur CPU her.
7. Laden Sie die PID-Parameter auf die CPU.
8. Klicken Sie auf das Symbol "Regler aktivieren".

Ergebnis

PID_3Step wechselt in den Automatikbetrieb und regelt mit den aktuellen PID-Parametern.

Motorstellzeit messen V1

Einleitung

PID_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Dass Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

Voraussetzung

- In den Grundeinstellungen ist Feedback oder Feedback_PER ausgewählt und das Signal ist verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

1. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
2. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Die Differenz zwischen "Zielstellung" und "Stellungsrückmeldung" muss mindestens 50 % des zulässigen Ausgangswertbereichs sein.
3. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

Motorstellzeit = (Obergrenze Ausgangswert – Untergrenze Ausgangswert) × Messzeit / BETRAG(Zielstellung – Startstellung).

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wechselt PID_3Step in die Betriebsart "Inaktiv".

Hinweis

Um die gemessene Motorstellzeit in das Projekt zu übernehmen, klicken Sie auf das Symbol "Gemessene Stellzeit laden".

Stellglieder mit Anschlagsignalen

Um die Stellzeit von Stellgliedern mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

Voraussetzung

- In den Grundeinstellungen ist das Optionskästchen "Anschlagsignale" aktiviert und Actuator_H und Actuator_L sind verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
2. Wählen Sie die Richtung, in die das Stellglied bewegt werden soll.
   - Öffnen - Schließen - Öffnen
     Das Stellglied wird zuerst bis zum oberen Anschlag gefahren, dann zum unteren Anschlag und erneut zum oberen Anschlag.
   - Schließen - Öffnen - Schließen
     Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag.
3. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird in der gewählten Richtung bewegt. Die Zeitmessung startet, wenn das Stellglied den ersten Anschlag erreicht hat und endet, wenn das Stellglied diesen Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wechselt PID_3Step in die Betriebsart "Inaktiv".

Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung abbrechen, wechselt PID_3Step sofort in die Betriebsart "Inaktiv". Das Stellglied wird nicht mehr bewegt. Sie können PID-3Step im Kurvenschreiber wieder aktivieren.

PID_3Step V1 mit PLCSIM simulieren

Hinweis

Simulation mit PLCSIM

Bei der Simulation mit PLCSIM ist das zeitliche Verhalten der simulierten PLC nicht exakt identisch zu einer "echten" PLC. Der tatsächliche Zeittakt eines Weckalarm-OB kann bei einer simulierten PLC größere Schwankungen aufweisen als bei "echten" PLCs.

In der Standardkonfiguration ermittelt PID_3Step die Zeit zwischen den Aufrufen automatisch und überwacht diese auf Schwankungen.

Bei der Simulation von PID_3Step mit PLCSIM kann deshalb ein Abtastzeitfehler (ErrorBits = DW#16#00000800) erkannt werden.

Dies führt zum Abbruch von laufenden Optimierungen.

Die Reaktion im Automatikbetrieb ist abhängig vom Wert der Variable ActivateRecoverMode.

Um dies zu verhindern, sollten Sie PID_3Step bei Simulation mit PLCSIM wie folgt konfigurieren:

- CycleTime.EnEstimation = FALSE
- CycleTime.EnMonitoring = FALSE
- CycleTime.Value: Weisen Sie dieser Variablen den Zeittakt des aufrufenden Weckalarm-OB in der Einheit Sekunden zu.

3.3.3 PID_Temp einsetzen

3.3.3.1 Technologieobjekt PID_Temp

Das Technologieobjekt PID_Temp stellt einen kontinuierlichen PID-Regler mit integrierter Optimierung zur Verfügung. PID_Temp ist speziell für Temperaturregelung ausgelegt und eignet sich für Heizen- oder Heizen/Kühlen-Anwendungen. Dafür stehen zwei Ausgänge zur Verfügung, je einer für Heizen und einer für Kühlen. Zusätzlich kann PID_Temp auch für andere Regelungsaufgaben verwendet werden. PID_Temp ist kaskadierbar und kann in Hand- oder Automatikbetrieb eingesetzt werden.

PID_Temp erfasst innerhalb eines Regelkreises fortlaufend den gemessenen Istwert und vergleicht diesen mit dem eingestellten Sollwert. Aus den sich ergebenden Regeldifferenzen errechnet die Anweisung PID_Temp den Ausgangswert für Heizen und/oder Kühlen, durch d

ie der Istwert an den Sollwert angeglichen wird. Die Ausgangswerte setzen sich beim PID-Regler aus drei Anteilen zusammen:

- P-Anteil
  Der P-Anteil des Ausgangswerts steigt proportional zur Regeldifferenz.
- I-Anteil
  Der I-Anteil des Ausgangswerts steigt, solange bis die Regeldifferenz ausgeglichen ist.
- D-Anteil
  Der D-Anteil steigt mit wachsender Änderungsgeschwindigkeit der Regeldifferenz. Der Istwert wird möglichst schnell an den Sollwert angeglichen. Nimmt die Änderungsgeschwindigkeit der Regeldifferenz wieder ab, so verringert sich der D-Anteil wieder.

Die Anweisung PID_Temp berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der "Erstoptimierung". Die Parameter können über eine "Nachoptimierung" weiter optimiert werden. Sie brauchen die Parameter nicht manuell ermitteln.

Für Heizen/Kühlen-Anwendungen kann entweder ein fester Kühlfaktor oder zwei PID-Parametersätze verwendet werden.

Weitere Informationen

- Übersicht der Software-Regler (Seite 651)
- Technologieobjekte hinzufügen (Seite 653)
- Technologieobjekte konfigurieren (Seite 654)
- PID_Temp konfigurieren (Seite 759)

3.3.3.2 PID_Temp konfigurieren

Grundeinstellungen

Einleitung

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID_Temp:

- Physikalische Größe
- Anlaufverhalten nach Reset
- Quelle und Eingabe des Sollwerts (nur im Inspektorfenster)
- Auswahl des Istwerts
- Quelle und Eingabe des Istwerts (nur im Inspektorfenster)
- Auswahl des Ausgangswerts Heizen
- Quelle und Eingabe des Ausgangswerts Heizen (nur im Inspektorfenster)
- Aktivierung und Auswahl des Ausgangswerts Kühlen

- Quelle und Eingabe des Ausgangswerts Kühlen (nur im Inspektorfenster)
- PID_Temp als Master oder Slave einer Kaskade aktivieren
- Anzahl der Slaves
- Auswahl des Master (nur im Inspektorfenster)

Sollwert, Istwert, Ausgangswert Heizen und Ausgangswert Kühlen

Für Sollwert, Istwert, Ausgangswert Heizen und Ausgangswert Kühlen können Sie im Inspektorfenster des Programmiereditors die Quelle auswählen und Werte bzw. Variablen eingeben.

Für jeden Wert wählen Sie die Quelle:

- Instanz-DB:
  Es wird der Wert verwendet, der im Instanz-DB gespeichert ist. Der Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden. An der Anweisung darf kein Wert stehen. Änderung über HMI möglich.
- Anweisung:
  Es wird der Wert verwendet, der an der Anweisung verschaltet ist. Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben. Änderung über HMI nicht möglich.

Regelungsart

Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Soll- und Istwert. Soll- und Istwert werden in dieser Einheit angezeigt.

Anlaufverhalten

1. Um nach Neustart der CPU in die Betriebsart "Inaktiv" zu wechseln, deaktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
   Um nach Neustart der CPU in die Betriebsart zu wechseln, die an Mode gespeichert ist, aktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
2. Wählen Sie in der Klappliste "Mode setzen auf" die Betriebsart aus, die nach einem vollständigen Laden in Gerät aktiviert werden soll.
   Nach einem vollständigen "Laden in Gerät" startet PID_Temp in der gewählten Betriebsart. Bei jedem weiteren Neustart, startet PID_Temp in der Betriebsart, die zuletzt an Mode gespeichert war.
   Bei der Wahl von Erstoptimierung oder Nachoptimierung müssen Sie zusätzlich die Variablen Heat.EnableTuning und Cool.EnableTuning setzen bzw. rücksetzen, um zwischen Optimierung für Heizen und Optimierung für Kühlen zu wählen.

Beispiel:

Sie haben das Optionskästchen "Nach CPU Neustart Mode aktivieren" aktiviert und in der Liste "Mode setzen auf" den Eintrag "Erstoptimierung" gewählt. Nach einem vollständigen "Laden in Gerät" startet PID_Temp in der Betriebsart "Erstoptimierung". Falls die Erstoptimierung noch aktiv ist, startet PID_Temp nach dem Neustart der CPU wieder in der

Betriebsart "Erstoptimierung" (Heizen/Kühlen abhängig von den Variablen Heat.EnableTuning und Cool.EnableCooling). Falls die Erstoptimierung erfolgreich beendet wurde und der Automatikbetrieb aktiv ist, startet PID_Temp nach dem Neustart der CPU im "Automatikbetrieb".

Sollwert

Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie wie folgt vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80°C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie wie folgt vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.
   Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert zeitgesteuert zu ändern.

Istwert

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID_Temp den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

Ausgangswert Heizen und Kühlen

Die Anweisung PID_Temp stellt einen PID-Regler mit integrierter Optimierung für Temperaturprozesse zur Verfügung. PID_Temp eignet sich für Heizen- oder Heizen/Kühlen-Anwendungen.

PID_Temp stellt folgende Ausgangswerte zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- OutputHeat
  Ausgangswert Heizen (Gleitpunktformat): Der Ausgangswert für Heizen muss über das Anwenderprogramm aufbereitet werden, z. B. weil das Stellglied nichtlineares Verhalten zeigt.
- OutputHeat_PER
  Analoger Ausgangswert Heizen: Das Stellglied für Heizen wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10 V, 4...20 mA.
- OutputHeat_PWM
  Pulsweitenmodulierter Ausgangswert Heizen: Das Stellglied für Heizen wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.
- OutputCool
  Ausgangswert Kühlen (Gleitpunktformat): Der Ausgangswert für Kühlen muss über das Anwenderprogramm aufbereitet werden, z. B. weil das Stellglied nichtlineares Verhalten zeigt.
- OutputCool_PER
  Analoger Ausgangswert Kühlen: Das Stellglied für Kühlen wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10 V, 4...20 mA.
- OutputCool_PWM
  Pulsweitenmodulierter Ausgangswert Kühlen: Das Stellglied für Kühlen wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.

Der Kühlausgang ist nur verfügbar, falls er über das Optionskästchen "Kühlung aktivieren" aktiviert wurde.

- Ist das Optionskästchen deaktiviert, wird der Ausgangswert des PID-Algorithmus (PidOutputSum) skaliert und an den Ausgängen für Heizen ausgegeben.
- Ist das Optionskästchen aktiviert, werden positive Ausgangswerte des PID-Algorithmus (PidOutputSum) skaliert und an den Ausgängen für Heizen ausgegeben. Negative Ausgangswerte des PID-Algorithmus werden skaliert und an den Ausgängen für Kühlen ausgegeben. In den Ausgangseinstellungen kann zwischen zwei Methoden zur Ausgangswertberechnung gewählt werden.

Hinweis

Beachten Sie folgendes:

- Die Ausgänge OutputHeat_PWM, OutputHeat_PER, OutputCool_PWM, OutputCool_PER werden nur berechnet, falls Sie diese in der Klappliste entsprechend auswählen.
- Der Ausgang OutputHeat wird immer berechnet.
- Der Ausgang OutputCool wird berechnet, falls das Optionskästchen für die Kühlung aktiviert ist.
- Das Optionskästchen "Kühlung aktivieren" ist nur verfügbar, falls der Regler nicht als Master in einer Kaskade konfiguriert ist.

Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "OutputHeat" bzw. "OutputCool" den Eintrag "OutputHeat_PER" bzw. "OutputCool_PER".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den pulsweitenmodulierten Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "OutputHeat" bzw. "OutputCool" den Eintrag "OutputHeat_PWM" bzw. "OutputCool_PWM".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Digitalausgangs ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "OutputHeat" bzw. "OutpuCool" den Eintrag "OutputHeat" bzw. "OutputCool".
2. Wählen Sie "Anweisung".
3. Geben Sie den Namen der Variablen an, die Sie für die Aufbereitung des Ausgangswerts verwenden.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Analog- oder Digitalausgang der CPU zum Stellglied.

Kaskade

Erhält eine PID_Temp-Instanz ihren Sollwert von einem übergeordneten Master-Regler und gibt ihren Ausgangswert selbst an einen untergeordneten Slave-Regler weiter, ist diese PID_Temp-Instanz sowohl Master-Regler als auch Slave-Regler zur gleichen Zeit. Für eine solche PID_Temp-Instanz sind dann beide unten aufgeführte Konfigurationen vorzunehmen. Dies ist zum Beispiel für die mittlere PID_Temp-Instanz bei einer Kaskadenregelung mit drei verketteten Messgrößen und drei PID_Temp-Instanzen der Fall.

Regler als Master in einer Kaskade konfigurieren

Ein Master-Regler gibt mit seinem Ausgang den Sollwert eines Slave-Reglers vor.

Um PID_Temp als Master in einer Kaskade zu verwenden, müssen Sie in den Grundeinstellungen die Kühlung deaktivieren. Um diese PID_Temp Instanz als Master-Regler in einer Kaskade zu konfigurieren, aktivieren Sie das Optionskästchen "Regler ist Master". Die Auswahl des Ausgangswerts für Heizen wird dabei automatisch auf OutputHeat gesetzt.

OutputHeat_PWM und OutputHeat_PER können bei einem Master in einer Kaskade nicht verwendet werden.

Geben Sie anschließend die Anzahl direkt untergeordneter Slave-Regler an, die ihren Sollwert von diesem Master-Regler erhalten.

Wird bei der Zuweisung des Parameter OutputHeat des Master auf den Parameter Setpoint des Slave keine eigene Skalierungsfunktion eingesetzt, kann es notwendig sein, die Ausgangswertgrenzen und die Ausgangsskalierung des Master an den Sollwert-/ Istwertbereich des Slave anzupassen. Dies können Sie in den Ausgangseinstellungen des Master im Bereich "OutputHeat / OutputCool" vornehmen.

Regler als Slave in einer Kaskade konfigurieren

Ein Slave-Regler erhält seinen Sollwert (Parameter Setpoint) vom Ausgang seines Master-Reglers (Parameter OutputHeat).

Um diese PID_Temp Instanz als Slave-Regler in einer Kaskade zu konfigurieren, aktivieren Sie in den Grundeinstellungen das Optionskästchen "Regler ist Slave".

Wählen Sie anschließend im Inspektorfenster des Programmiereditors die PID_Temp-Instanz aus, die als Master-Regler für diesen Slave-Regler verwendet werden soll. Durch die Auswahl werden die Parameter Master und Setpoint des Slave-Reglers mit dem gewählten Master-Regler verschaltet (bisherige Verschaltungen an diesen Parametern werden überschrieben). Diese Verschaltung ermöglicht den Informationsaustausch und die Sollwertvorgabe zwischen Master und Slave. Bei Bedarf können Sie die Verschaltung am Parameter Setpoint des Slave-Reglers nachträglich ändern z .B. um ein zusätzliches Filter einzufügen. Die Verschaltung an Parameter Master darf nachträglich nicht verändert werden.

Am ausgewählten Master-Regler muss das Optionskästchen "Regler ist Master" aktiviert und die Anzahl der Slaves korrekt konfiguriert sein. Der Master-Regler muss vor dem Slave-Regler im gleichen Weckalarm-OB aufgerufen werden.

Weitere Informationen

Weitere Informationen zur Programmerstellung, Konfiguration und Inbetriebnahme beim Einsatz von PID_Temp in Kaskadenregelungen finden Sie unter Kaskadenregelung mit PID_Temp (Seite 790).

Istwerteinstellungen

Istwertgrenzen

Als Grenzwerte müssen Sie für Ihre Regelstrecke eine sinnvolle absolute Ober- und Untergrenze des Istwerts festlegen. Sobald diese Grenzen über- oder unterschritten werden, tritt ein Fehler auf (ErrorBits = 0001h). Die Optimierung wird abgebrochen, wenn die Istwertgrenzen überschritten werden. Die Reaktion von PID_Temp im Fehlerfall im Automatikbetrieb legen Sie bei den Ausgangseinstellungen fest.

Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input_PER anhand eines unteren und oberen Wertepaars skaliert.

Vorgehen

Um den Istwert zu skalieren, gehen Sie folgendermaßen vor:

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID_Temp.
2. Verschalten Sie in den Grundeinstellungen Input_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

Ausgangseinstellungen

Grundeinstellungen Ausgang

Methode für Heizen und Kühlen

Falls die Kühlung in den Grundeinstellungen aktiviert ist, stehen für die Berechnung des PID Ausgangswerts zwei Methoden zur Verfügung:

- PID-Parameterumschaltung (Config.AdvancedCooling = TRUE):
  Die Ausgangswertberechnung für Kühlen erfolgt über einen eigenen PID-Parametersatz. Der PID-Algorithmus entscheidet anhand des berechneten Ausgangswerts und der Regeldifferenz, ob die PID-Parameter für Heizen oder Kühlen verwendet werden. Diese Methode ist geeignet, falls Heiz- und Kühlstellglied unterschiedliches Zeitverhalten und unterschiedliche Verstärkungen aufweisen.
  Nur bei Wahl dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen zur Verfügung.
- Kühlfaktor (Config.AdvancedCooling = FALSE):
  Die Ausgangswertberechnung für Kühlen erfolgt mit den PID-Parametern für Heizen unter Berücksichtigung des konfigurierbaren Kühlfaktors Config.CoolFactor. Diese Methode ist geeignet, falls Heiz- und Kühlstellglied ähnliches Zeitverhalten aber unterschiedliche Verstärkungen aufweisen. Bei Wahl dieser Methode stehen Erstoptimierung und Nachoptimierung für Kühlen sowie der PID-Parametersatz für Kühlen nicht zur Verfügung. Es können nur die Optimierungen für Heizen ausgeführt werden.

Kühlfaktor

Falls als Methode für Heizen/Kühlen der Kühlfaktor gewählt ist, wird dieser als Faktor in der Berechnung des Ausgangswerts für Kühlen berücksichtigt. Auf diese Weise können unterschiedliche Verstärkungen von Heiz- und Kühlstellglied berücksichtigt werden.

Der Kühlfaktor wird nicht automatisch eingestellt oder während der Optimierung angepasst. Sie müssen den Kühlfaktor manuell mit dem Verhältnis "Verstärkung Heizstellglied / Verstärkung Kühlstellglied" korrekt konfigurieren.

Beispiel: Kühlfaktor = 2.0 bedeutet, dass die Verstärkung des Heizstellglieds doppelt so stark wie die Verstärkung des Kühlstellglieds ist.

Der Kühlfaktor ist nur wirksam und veränderbar, falls "Kühlfaktor" als Methode für Heizen / Kühlen gewählt ist.

Verhalten im Fehlerfall

ACHTUNG
Ihre Anlage kann beschädigt werden. Falls Sie im Fehlerfall "Aktueller Wert für die Fehlerdauer" oder "Ersatzausgangswert für die Fehlerdauer" ausgeben, bleibt PID_Temp im Automatikbetrieb bzw. im Handbetrieb. Dadurch können die Istwertgrenzen überschritten und Ihre Anlage beschädigt werden. Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

PID_Temp ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt.

Falls im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache.

Im Fehlerfall gibt PID_Temp einen konfigurierbaren Ausgangswert aus:

- Null (Inaktiv)
  PID_Temp schaltet bei allen Fehlern in die Betriebsart "Inaktiv" und gibt folgendes aus:
  - 0.0 als PID Ausgangswert (PidOutputSum)
  - 0.0 als Ausgangswert für Heizen (OutputHeat) und Ausgangswert für Kühlen (OutputCool)
  - 0 als analoger Ausgangswert für Heizen (OutputHeat_PER) und analoger Ausgangswert für Kühlen (OutputCool_PER)
  - FALSE als PWM Ausgangswert für Heizen (OutputHeat_PWM) und PWM Ausgangswert für Kühlen (OutputCool_PWM)

  Dies ist unabhängig von den konfigurierten Ausgangswertgrenzen und der -skalierung. Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate reaktiviert.
- Aktuellen Wert für die Fehlerdauer
  Die Fehlerreaktion ist abhängig vom auftretenden Fehler und der Betriebsart.
  Falls im Automatikbetrieb einer oder mehrere der folgenden Fehler auftreten, bleibt PID_Temp im Automatikbetrieb:
  - 0000001h: Der Parameter Input ist außerhalb der Istwertgrenzen.
  - 0000800h: Abtastzeitfehler
  - 0040000h: Ungültiger Wert am Parameter Disturbance.
  - 8000000h: Fehler während der Berechnung der PID Parameter.

  Falls im Automatikbetrieb einer oder mehrere der folgenden Fehler auftreten, wechselt PID_Temp in die Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt den letzten gültigen PID Ausgangswert (PidOutputSum) aus:
  - 0000002h: Ungültiger Wert am Parameter Input_PER.
  - 0000200h: Ungültiger Wert am Parameter Input.
  - 0000400h: Berechnung des Ausgangswerts fehlgeschlagen.
  - 0001000h: Ungültiger Wert am Parameter Setpoint oder SubstituteSetpoint.

  Die aus dem PID Ausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung.
  Sobald die Fehler nicht mehr anstehen, wechselt PID_Temp wieder in den Automatikbetrieb.
  Falls im Handbetrieb ein Fehler auftritt, bleibt PID_Temp im Handbetrieb und verwendet weiterhin den Handwert als PID Ausgangswert.
  Falls der Handwert ungültig ist, wird der konfigurierte Ersatzausgangswert verwendet.
  Falls Handwert und Ersatzausgangswert ungültig sind, wird die Untergrenze PID Ausgangswert für Heizen (Config.Output.Heat.PidLowerLimit) verwendet.
  Falls während einer Erst- oder Nachoptimierung folgender Fehler auftritt, bleibt PID_Temp in der aktiven Betriebsart:
  - 0000020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt.

Bei allen anderen Fehlern bricht PID_Temp die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.

- Ersatzausgangswert für die Fehlerdauer
  PID_Temp verhält sich wie bei "Aktuellen Wert für die Fehlerdauer" beschrieben, gibt aber in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" den konfigurierten Ersatzausgangswert (SubstituteOutput) als PID Ausgangswert (PidOutputSum) aus.
  Die aus dem PID Ausgangswert resultierenden Werte an den Ausgängen für Heizen und Kühlen ergeben sich aus der konfigurierten Ausgangsskalierung.
  Für Regler mit aktiviertem Kühlausgang (Config.ActivateCooling = TRUE) geben Sie,
  - um den Wert an den Ausgängen für Heizen auszugeben, einen positiven Ersatzausgangswert ein.
  - um den Wert an den Ausgängen für Kühlen auszugeben, einen negativen Ersatzausgangswert ein.

  Falls folgender Fehler auftritt, bleibt PID_Temp in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt die Untergrenze PID Ausgangswert für Heizen (Config.Output.Heat.PidLowerLimit) aus:
  - 0020000h: Ungültiger Wert an der Variablen SubstituteOutput.

Ausgangswertgrenzen und -skalierung

Der PID Ausgangswert (PidOutputSum) wird je nach Betriebsart durch den PID-Algorithmus automatisch berechnet oder durch den Handwert (ManualValue) oder den konfigurierten Ersatzausgangswert (SubstituteOutput) vorgegeben.

Der PID Ausgangswert wird abhängig von der Konfiguration begrenzt:

- Ist die Kühlung in den Grundeinstellungen deaktiviert (Config.ActivateCooling = FALSE), wird der Wert auf die Obergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidUpperLimit) und die Untergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidLowerLimit) begrenzt.
  Beide Grenzwerte können Sie im Abschnitt "OutputHeat / OutputCool" an der horizontalen Achse der Skalierungskennlinie konfigurieren. In den Abschnitten "OutputHeat_PWM / OutputCool_PWM" und "OutputHeat_PER / OutputCool_PER" werden diese angezeigt, können aber nicht verändert werden.
- Ist die Kühlung in den Grundeinstellungen aktiviert (Config.ActivateCooling = TRUE), wird der Wert auf die Obergrenze PID Ausgangswert (Config.Output.Heat.PidUpperLimit) und die Untergrenze PID Ausgangswert (Kühlen) (Config.Output.Cool.PidLowerLimit) begrenzt.
  Beide Grenzwerte können Sie im Abschnitt „OutputHeat / OutputCool“ an der horizontalen Achse der Skalierungskennlinie konfigurieren. In den Abschnitten "OutputHeat_PWM / OutputCool_PWM" und "OutputHeat_PER / OutputCool_PER" werden diese angezeigt, können aber nicht verändert werden.
  Die Untergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidLowerLimit) und die Obergrenze PID Ausgangswert (Kühlen) (Config.Output.Cool.PidUpperLimit) können nicht verändert werden und müssen mit dem Wert 0.0 belegt sein.

Der PID Ausgangswert wird skaliert und an den Ausgängen für Heizen und Kühlen ausgegeben. Die Skalierung kann für jeden Ausgang getrennt vorgegeben werden und wird

über je 2 Wertepaare, bestehend aus einem Grenzwert des PID Ausgangswerts und einem Skalierungswert, festgelegt:

Ausgang	Wertepaar	Parameter
OutputHeat	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer Ausgangswert (Heizen) Config.Output.Heat.UpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer Ausgangswert (Heizen) Config.Output.Heat.LowerScaling
OutputHeat_PWM	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer PWM Ausgangswert (Heizen) Config.Output.Heat.PwmUpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer PWM Ausgangswert (Heizen) Config.Output.Heat.PwmLowerScaling
OutputHeat_PER	Wertepaar 1	Obergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidUpperLimit, Skalierter oberer analoger Ausgangswert (Heizen) Config.Output.Heat.PerUpperScaling
	Wertepaar 2	Untergrenze PID Ausgangswert (Heizen) Config.Output.Heat.PidLowerLimit, Skalierter unterer analoger Ausgangswert (Heizen) Config.Output.Heat.PerLowerScaling
OutputCool	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer Ausgangswert (Kühlen) Config.Output.Cool.UpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer Ausgangswert (Kühlen) Config.Output.Cool.LowerScaling
OutputCool_PWM	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer PWM Ausgangswert (Kühlen) Config.Output.Cool.PwmUpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer PWM Ausgangswert (Kühlen) Config.Output.Cool.PwmLowerScaling

Ausgang	Wertepaar	Parameter
OutputCool_PER	Wertepaar 1	Untergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidLowerLimit, Skalierter oberer analoger Ausgangswert (Kühlen) Config.Output.Cool.PerUpperScaling
	Wertepaar 2	Obergrenze PID Ausgangswert (Kühlen) Config.Output.Cool.PidUpperLimit, Skalierter unterer analoger Ausgangswert (Kühlen) Config.Output.Cool.PerLowerScaling

Die Untergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidLowerLimit) muss den Wert 0.0 haben, falls die Kühlung aktiviert ist (Config.ActivateCooling = TRUE).

Die Obergrenze PID Ausgangswert (Kühlen) (Config.Output.Cool.PidUpperLimit) muss immer den Wert 0.0 haben.

Beispiel:

Ausgangsskalierung bei Verwendung von Ausgang OutputHeat (Kühlung deaktiviert; die Untergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidLowerLimit) darf ungleich 0.0 sein):

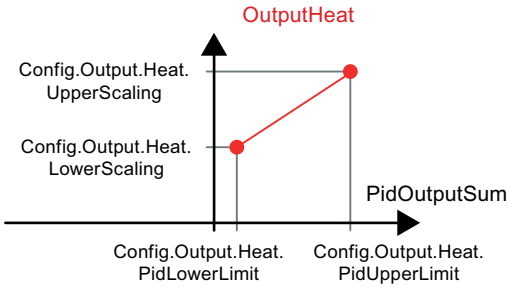

Beispiel:

Ausgangsskalierung bei Verwendung von Ausgang OutputHeat_PWM und OutputCool_PER (Kühlung aktiviert; die Untergrenze PID Ausgangswert (Heizen) (Config.Output.Heat.PidLowerLimit) muss 0.0 sein):

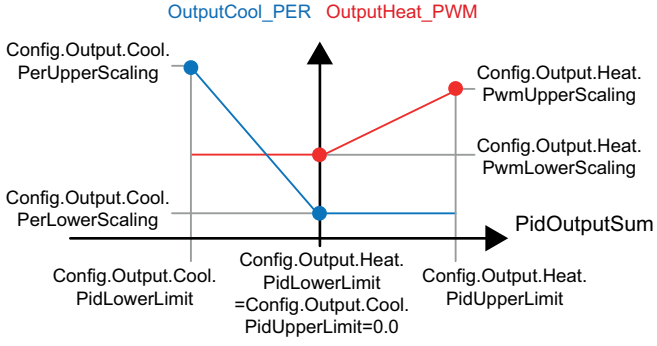

Mit Ausnahme der Betriebsart "Inaktiv", liegt der Wert an einem Ausgang immer zwischen seinem skalierten oberen Ausgangswert und skalierten unteren Ausgangswert z. B. für

OutputHeat immer zwischen dem skalierten oberen Ausgangswert (Heizen) (Config.Output.Heat.UpperScaling) und dem skalierten unteren Ausgangswert (Heizen) (Config.Output.Heat.LowerScaling).

Falls Sie den Wert an dem zugehörigen Ausgang begrenzen wollen, müssen Sie daher auch diese Skalierungswerte anpassen.

Die Skalierungswerte eines Ausgangs können Sie an den vertikalen Achsen der Skalierungskennlinie konfigurieren. Jeder Ausgang verfügt über zwei eigene Skalierungswerte. Für OutputHeat_PWM, OutputCool_PWM, OutputHeat_PER und OutputCool_PER sind diese nur änderbar, falls der entsprechende Ausgang in den Grundeinstellungen ausgewählt ist. Bei allen Ausgängen für Kühlen muss zusätzlich die Kühlung in den Grundeinstellungen aktiviert sein.

Die Kurvenanzeige im Inbetriebnahmedialog zeichnet nur die Werte von OutputHeat und OutputCool auf, unabhängig vom gewählten Ausgang in den Grundeinstellungen. Passen Sie daher bei Bedarf auch die Skalierungswerte für OutputHeat bzw. OutputCool an, falls Sie OutputHeat_PWM oder OutputHeat_PER bzw. OutputCool_PWM oder OutputCool_PER verwenden und die Kurvenanzeige im Inbetriebnahmedialog nutzen wollen.

Erweiterte Einstellungen

Istwertüberwachung

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über- oder unterschritten, so wird an der Anweisung PID_Temp eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID_Temp verhält sich folgendermaßen:

Istwert	InputWarning_H	InputWarning_L	ErrorBits
> 98 °C	TRUE	FALSE	0001h
≤ 98 °C und > 90 °C	TRUE	FALSE	0000h
≤ 90 °C und ≥ 10 °C	FALSE	FALSE	0000h
< 10°C und ≥ 0 °C	FALSE	TRUE	0000h
< 0 °C	FALSE	TRUE	0001h

Wie PID_Temp beim Über- oder Unterschreiten der Obergrenze bzw. Untergrenze Istwert reagiert, konfigurieren Sie bei den Ausgangseinstellungen.

PWM-Begrenzungen

Der PID Ausgangswert PidOutputSum wird skaliert und über eine Pulsweitenmodulation in eine Impulsfolge transformiert, die am Ausgangsparameter OutputHeat_PWM bzw. OutputCool_PWM ausgegeben wird. Die "Abtastzeit PID-Algorithmus" ist der Abstand zwischen zwei Berechnungen des PID Ausgangswerts. Die Abtastzeit wird als Periodendauer der Pulsweitenmodulation verwendet.

Während des Heizens, wird der PID Ausgangswert immer in der "Abtastzeit PID-Algorithmus für Heizen" berechnet.

Die Berechnung des PID Ausgangswerts während des Kühlens hängt von der in "Grundeinstellungen Ausgang" gewählten Art der Kühlung ab:

- Wird der Kühlfaktor verwendet, gilt die "Abtastzeit PID-Algorithmus für Heizen".
- Wird PID-Parameterumschaltung verwendet, gilt die "Abtastzeit PID-Algorithmus für Kühlen".

OutputHeat_PWM und OutputCool_PWM werden in der Abtastzeit PID_Temp ausgegeben (entspricht der Zykluszeit des aufrufenden OB).

Die Abtastzeit PID-Algorithmus für Heizen bzw. Kühlen wird während der Erst- oder Nachoptimierung ermittelt. Falls Sie die PID-Parameter manuell einstellen, müssen Sie dort auch die Abtastzeit PID-Algorithmus für Heizen bzw. Kühlen konfigurieren. Die Abtastzeit PID_Temp entspricht der Zykluszeit des aufrufenden OB.

Die Impulsdauer ist proportional zum PID Ausgangswert·und beträgt stets ein ganzzahliges Vielfaches der Abtastzeit PID_Temp.

Beispiel für OutputHeat_PWM

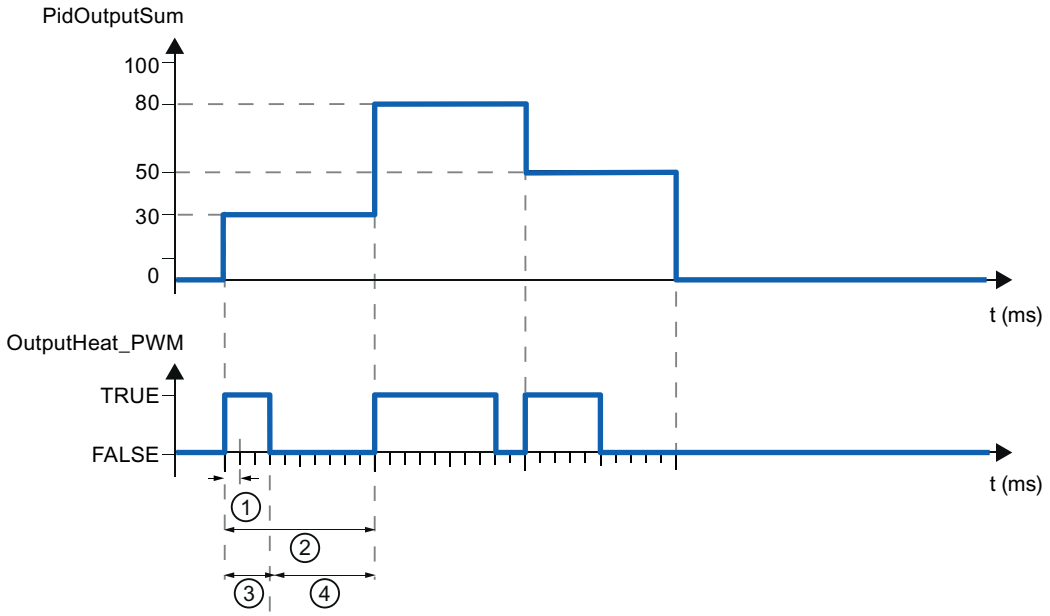

① Abtastzeit PID_Temp
② Abtastzeit PID-Algorithmus für Heizen
③ Impulsdauer
④ Pausendauer

Die "Minimale Einschaltzeit" und die "Minimale Ausschaltzeit" können, gerundet auf ein ganzzahliges Vielfaches der Abtastzeit PID_Temp, für Heizen und Kühlen getrennt eingestellt werden.

Ein Impuls oder eine Pause wird nie kürzer als die minimale Ein- oder Ausschaltzeiten. Die Ungenauigkeiten, die sich dadurch ergeben, werden aufsummiert und im nächsten Zyklus ausgeglichen.

Beispiel für OutputHeat_PWM

Abtastzeit PID_Temp = 100 ms

Abtastzeit PID-Algorithmus = 1000 ms

Minimale Einschaltzeit = 200 ms

Der PID Ausgangswert PidOutputSum beträgt konstant 15 %. Der kleinste Impuls, den PID_Temp ausgeben kann, entspricht 20 %. Im ersten Zyklus wird kein Impuls ausgegeben. Im zweiten Zyklus wird der nicht ausgegebene Impuls des ersten Zyklus zum Impuls des zweiten Zyklus addiert.

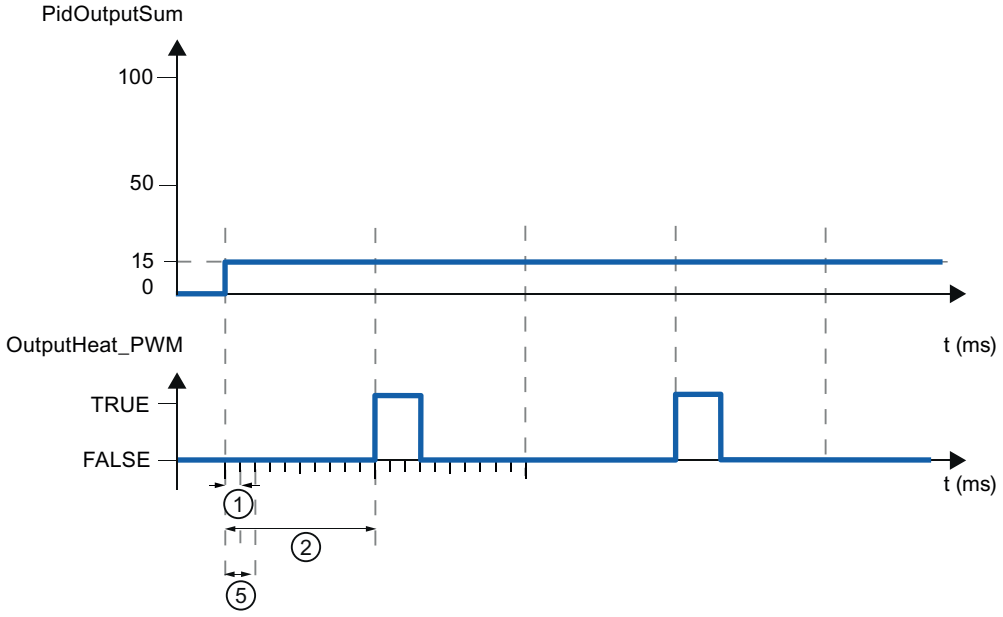

① Abtastzeit PID_Temp

② Abtastzeit PID-Algorithmus für Heizen

⑤ Minimale Einschaltzeit

Um die Schalthäufigkeit zu verringern und das Stellglied zu schonen, verlängern Sie die minimalen Ein- und Ausschaltzeiten.

Falls Sie in den Grundeinstellungen als Ausgang OutputHeat bzw. OutputCool oder OutputHeat_PER bzw. OutputCool_PER gewählt haben, werden die minimale Einschaltzeit und die minimale Ausschaltzeit nicht ausgewertet und sind nicht änderbar.

Ist bei Verwendung von OutputHeat_PWM bzw. OutputCool_PWM die "Abtastzeit PID-Algorithmus" (Retain.CtrlParams.Heat.Cycle bzw. Retain.CtrlParams.Cool.Cycle) und damit die Periodendauer der Pulsweitenmodulation sehr groß, können Sie, um die Glattheit des Istwerts zu verbessern, an den Parametern Config.Output.Heat.PwmPeriode bzw.

Config.Output.Cool.PwmPeriode eine abweichende kürzere Periodendauer vorgeben (siehe auch AUTOHOTSPOT).

Hinweis

Die minimalen Ein- und Ausschaltzeiten wirken nur auf die Ausgangsparameter OutputHeat_PWM bzw. OutputCool_PWM und werden nicht für evtl. in der CPU integrierte Impulsgeneratoren verwendet.

PID-Parameter

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt.

Falls in den Grundeinstellungen die Kühlung aktiviert und in den Ausgangseinstellungen PID-Parameterumschaltung als Methode für Heizen/Kühlen gewählt ist, stehen zwei Parametersätze zur Verfügung: einer für Heizen und einer für Kühlen.

In diesem Fall entscheidet der PID-Algorithmus anhand des berechneten Ausgangswerts und der Regeldifferenz, ob die PID-Parameter für Heizen oder Kühlen verwendet werden.

Falls die Kühlung deaktiviert oder Kühlfaktor als Methode für Heizen/Kühlen gewählt ist, wird immer der Parametersatz für Heizen verwendet.

Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst, mit Ausnahme der Totzonenbreite, die manuell konfiguriert werden muss.

Hinweis

Die aktuell wirksamen PID-Parameter befinden sich in der Struktur Retain.CtrlParams.

Um ein Fehlverhalten des PID-Reglers zu vermeiden, ändern Sie online die aktuell wirksamen PID-Parameter nur in der Betriebsart "inaktiv".

Wenn Sie die PID-Parameter in den Betriebsarten "Automatikbetrieb" oder "Handbetrieb" online ändern möchten, dann ändern Sie die PID-Parameter in der Struktur CtrlParamsBackUp und übernehmen diese Änderungen per LoadBackUp = TRUE in die Struktur Retain.CtrlParams.

Online Änderungen an den PID-Parametern in der Betriebsart „Automatikbetrieb“ können zu Sprüngen am Ausgangswert führen.

PID_Temp ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils.

Der PID-Algorithmus arbeitet nach folgender Formel (Regelzone und Totzone deaktiviert):

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Symbol	Beschreibung	Zugehörige Parameter der Anweisung PID_Temp
y	Ausgangswert des PID-Algorithmus	-
K_p	Proportionalverstärkung	Retain.CtrlParams.Heat.Gain Retain.CtrlParams.Cool.Gain CoolFactor

Symbol	Beschreibung	Zugehörige Parameter der Anweisung PID_Temp
s	Laplace-Operator	-
b	Gewichtung des P-Anteils	Retain.CtrlParams.Heat.PWeighting Retain.CtrlParams.Cool.PWeighting
w	Sollwert	CurrentSetpoint
x	Istwert	ScaledInput
T_I	Integrationszeit	Retain.CtrlParams.Heat.Ti Retain.CtrlParams.Cool.Ti
T_D	Differenzierzeit	Retain.CtrlParams.Heat.Td Retain.CtrlParams.Cool.Td
a	Koeffizient für den Differenzierverzug (Differenzierverzug T1 = a × T_D)	Retain.CtrlParams.Heat.TdFiltRatio Retain.CtrlParams.Cool.TdFiltRatio
c	Gewichtung des D-Anteils	Retain.CtrlParams.Heat.DWeighting Retain.CtrlParams.Cool.DWeighting
DeadZone	Totzonenbreite	Retain.CtrlParams.Heat.DeadZone Retain.CtrlParams.Cool.DeadZone
ControlZone	Regelzonenbreite	Retain.CtrlParams.Heat.ControlZone Retain.CtrlParams.Cool.ControlZone

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:

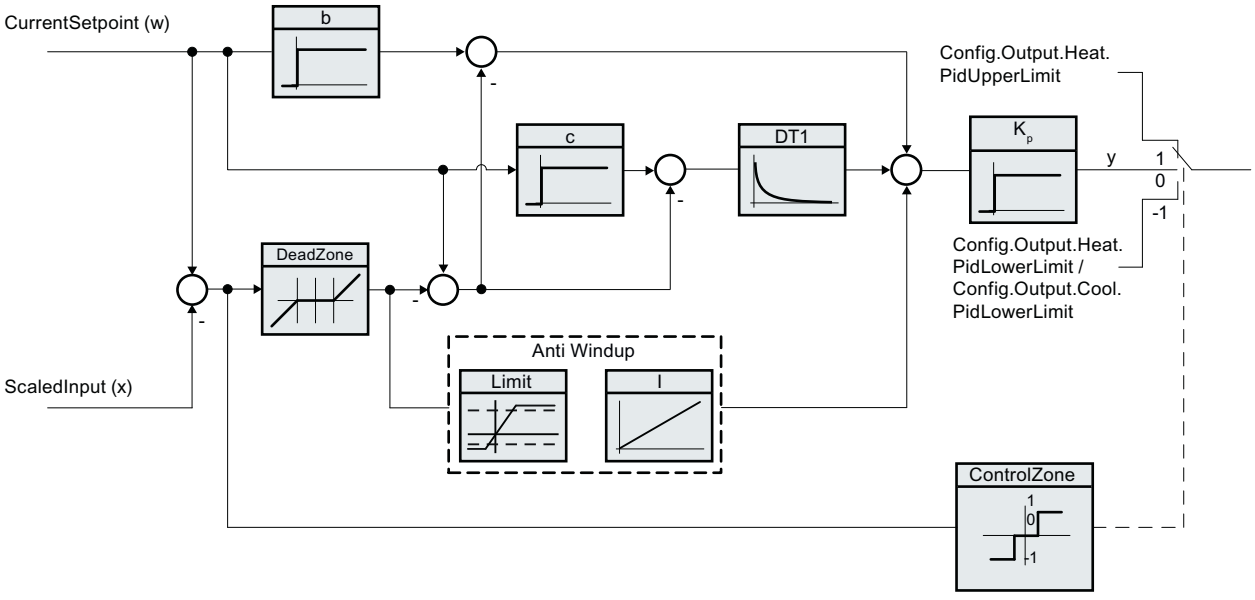

Alle PID-Parameter sind remanent. Falls Sie die PID-Parameter manuell eingeben, müssen Sie PID_Temp vollständig laden (Technologieobjekte in Gerät laden (Seite 656)).

Blockschaltbild PID_Temp

Das folgende Blockdiagramm zeigt, wie der PID-Algorithmus in PID_Temp integriert ist.

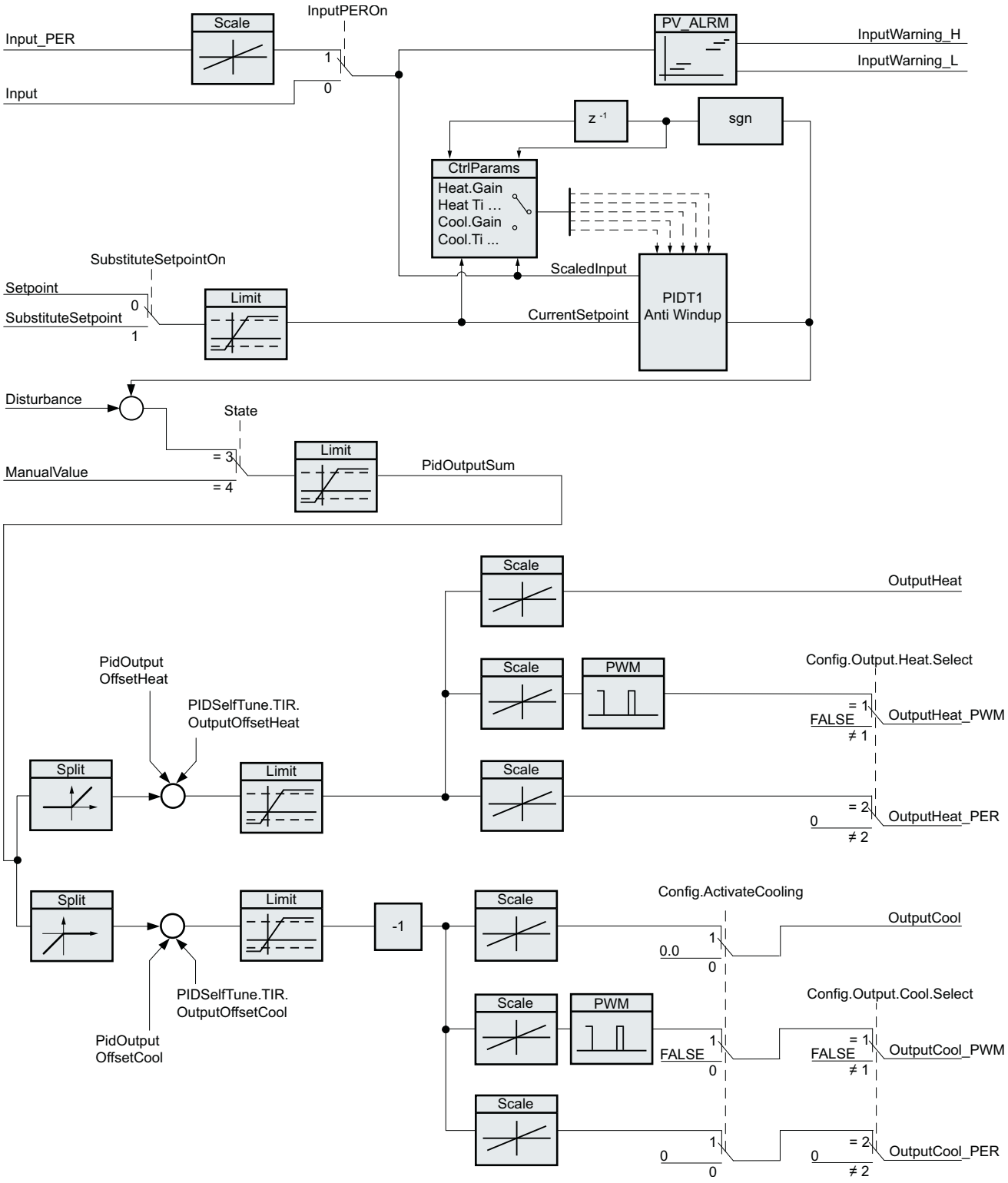

Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID_Temp arbeitet nicht mit einer negativen Proportionalverstärkung und unterstützt nur normalen Regelsinn d. h. mit einer Erhöhung des PID-Ausgangswerts (PidOutputSum) soll eine Erhöhung des Istwerts erreicht werden.

Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0. Wenn die Integrationszeit in der Betriebsart "Automatikbetrieb" von einem anderen Wert auf 0.0 online geändert wird, dann wird der bisherige I-Anteil gelöscht und es erfolgt ein Sprung des Ausgangswerts.

Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

Koeffizient Differenzierverzug

Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierverzug verzögert.

Differenzierverzug = Differenzierzeit × Koeffizient Differenzierverzug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit einer dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit "PID-Algorithmus" ist der Abstand zwischen zwei Berechnungen des PID Ausgangswerts. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Abtastzeit PID_Temp (Zykluszeit des Weckalarm-OB) gerundet. Alle anderen Funktionen von PID_Temp werden bei jedem Aufruf durchgeführt.

Falls Sie OutputHeat_PWM bzw. OutputCool_PWM verwenden, wird die Abtastzeit des PID-Algorithmus als Periodendauer der Pulsweitenmodulation verwendet. Die Genauigkeit des Ausgangssignals wird bestimmt durch das Verhältnis von Abtastzeit des PID-Algorithmus zu Zykluszeit des OB. Die Zykluszeit sollte höchstens ein Zehntel der Abtastzeit des PID-Algorithmus betragen.

Welche Abtastzeit des PID-Algorithmus als Periodendauer der Pulsweitenmodulation bei OutputCool_PWM verwendet wird, hängt von der in "Grundeinstellungen Ausgang" gewählten Methode für Heizen/Kühlen ab:

- Wird der Kühlfaktor verwendet, gilt auch für OutputCool_PWM die "Abtastzeit PID-Algorithmus für Heizen".
- Wird PID-Parameterumschaltung verwendet, gilt die "Abtastzeit PID-Algorithmus für Kühlen" als Periodendauer für OutputCool_PWM.

Ist bei Verwendung von OutputHeat_PWM bzw. OutputCool_PWM die Abtastzeit des PID-Algorithmus und damit die Periodendauer der Pulsweitenmodulation sehr groß, können Sie, um die Glattheit des Istwerts zu verbessern, an den Parametern Config.Output.Heat.PwmPeriode bzw. Config.Output.Cool.PwmPeriode eine abweichende kürzere Periodendauer vorgeben.

Totzonenbreite

Falls der Istwert verrauscht ist, wirkt sich der Rauschanteil auch am Ausgangswert aus. Bei hoher Reglerverstärkung und eingeschaltetem D-Anteil kann der Ausgangswert stark schwanken. Liegt der Istwert innerhalb der Totzone um den Sollwert, wird die Regeldifferenz unterdrückt, sodass der PID-Algorithmus nicht reagiert und unnötige Schwankungen des Ausgangswerts reduziert werden.

Die Totzonenbreite für Heizen bzw. Kühlen wird nicht automatisch während der Optimierung eingestellt. Sie müssen die Totzonenbreite manuell korrekt konfigurieren. Das Abschalten der Totzone erfolgt mit Totzonenbreite = 0.0.

Bei eingeschalteter Totzone kann sich eine dauerhafte Regeldifferenz (Abweichung zwischen Sollwert und Istwert) einstellen. Dies kann sich negativ auf die Durchführung einer Nachoptimierung auswirken.

Ist die Kühlung in den Grundeinstellungen aktiviert und die PID-Parameterumschaltung in den Ausgangseinstellungen als Methode für Heizen/Kühlen gewählt, liegt die Totzone zwischen "Sollwert - Totzonenbreite (Heizen)" und "Sollwert + Totzonenbreite (Kühlen)".

Ist die Kühlung in den Grundeinstellungen deaktiviert oder wird der Kühlfaktor verwendet, so liegt die Totzone symmetrisch zwischen "Sollwert - Totzonenbreite (Heizen)" und "Sollwert + Totzonenbreite (Heizen)".

Falls für die Gewichtung des P-Anteils oder die Gewichtung des D-Anteils Werte ungleich 1.0 konfiguriert sind, wirken sich Sollwertänderungen auch innerhalb der Totzone auf den

Ausgangswert aus.
Istwertänderungen innerhalb der Totzone wirken sich unabhängig von den Gewichtungen nicht auf den Ausgangswert aus.

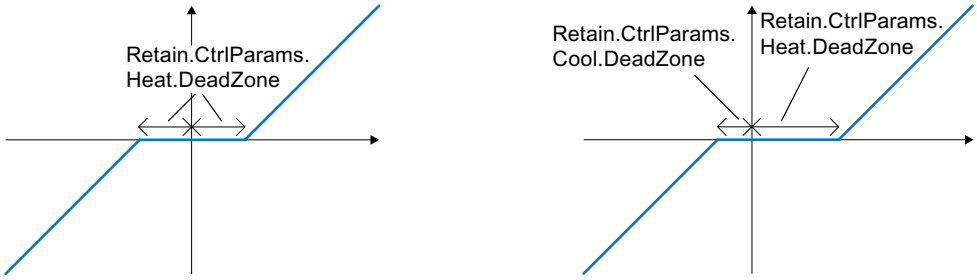

Totzone mit deaktivierter Kühlung oder Kühlfaktor (links) bzw. aktivierter Kühlung und PID-Parameterumschaltung (rechts). Die x / horizontale Achse zeigt die Regeldifferenz = Sollwert - Istwert. Die y / vertikale Achse zeigt das Ausgangssignal der Totzone, das an den PID-Algorithmus weitergegeben wird.

Regelzonenbreite

Verlässt der Istwert die Regelzone um den Sollwert, wird der minimale oder maximale Ausgangswert ausgegeben. Dadurch erreicht der Istwert den Sollwert schneller.

Liegt der Istwert innerhalb der Regelzone um den Sollwert, wird der Ausgangswert vom PID-Algorithmus berechnet.

Die Regelzonenbreite für Heizen bzw. Kühlen wird nur während der Erstoptimierung automatisch eingestellt, falls als Reglerstruktur für Heizen bzw. Kühlen "PID (Temperatur)" gewählt ist.

Das Abschalten der Regelzone erfolgt mit Regelzonenbreite = 3.402822e+38.

Ist die Kühlung in den Grundeinstellungen deaktiviert oder wird der Kühlfaktor verwendet, liegt die Regelzone symmetrisch zwischen "Sollwert - Regelzonenbreite (Heizen)" und "Sollwert + Regelzonenbreite (Heizen)".

Ist die Kühlung in den Grundeinstellungen aktiviert und die PID-Parameterumschaltung in den Ausgangseinstellungen als Methode für Heizen/Kühlen gewählt, liegt die Regelzone zwischen "Sollwert - Regelzonenbreite (Heizen)" und "Sollwert + Regelzonenbreite (Kühlen)".

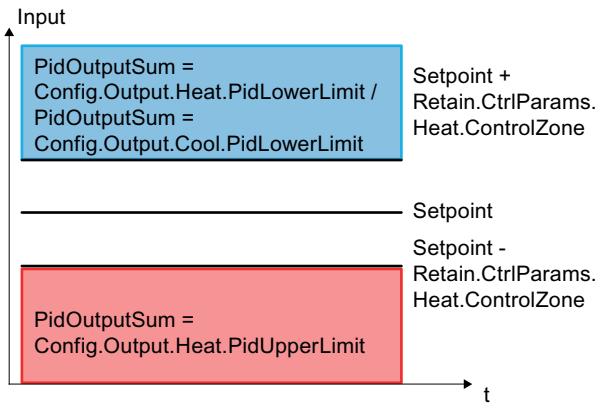

Regelzone mit deaktivierter Kühlung oder Kühlfaktor.

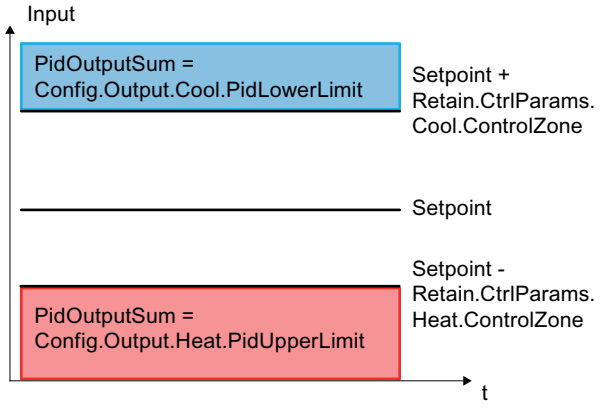

Regelzone mit aktivierter Kühlung und PID-Parameterumschaltung.

Regel für Optimierung

Wählen Sie in der Klappliste "Reglerstruktur" aus, ob PI oder PID-Parameter berechnet werden. Sie können die Regeln für Optimierung für Heizen und Optimierung für Kühlen getrennt vorgeben.

- PID (Temperatur)
  Berechnet während der Erst- und Nachoptimierung PID-Parameter.
  Die Erstoptimierung ist dabei für Temperaturprozesse ausgelegt und führt zu einem langsameren und eher asymptotischen Regelverhalten mit geringerem Überschwinger als mit der Option "PID". Die Nachoptimierung ist identisch zur Option "PID".
  Nur bei Wahl dieser Option wird die Regelzonenbreite während der Erstoptimierung automatisch ermittelt.
- PID
  Berechnet während der Erst- und Nachoptimierung PID-Parameter.
- PI
  Berechnet während der Erst- und Nachoptimierung PI-Parameter.
- Benutzerdefiniert
  Falls Sie über ein Anwenderprogramm oder die Parametersicht unterschiedliche Reglerstrukturen für Erst- und Nachoptimierung eingestellt haben, wird "Benutzerdefiniert" in der Klappliste angezeigt.

3.3.3.3 PID_Temp in Betrieb nehmen

Inbetriebnahme

Das Inbetriebnahmefenster hilft Ihnen bei der Inbetriebnahme des PID-Reglers. In der Kurvenanzeige können Sie die Werte für Sollwert, Istwert und die Ausgangswerte für Heizen und Kühlen entlang der Zeitachse beobachten. Folgende Funktionen werden im Inbetriebnahmefenster unterstützt:

- Erstoptimierung des Reglers
- Nachoptimierung des Reglers
  Verwenden Sie die Nachoptimierung, wenn Sie eine Feinjustage der PID-Parameter wünschen.
- Beobachtung der laufenden Regelung im Kurvenfenster
- Test der Regelstrecke durch Vorgabe eines manuellen PID Ausgangswerts und eines Ersatzsollwerts
- Sichern der Aktualwerte der PID-Parameter ins Offline-Projekt.

Für alle Funktionen muss eine Onlineverbindung zur CPU aufgebaut sein.

Über die Schaltflächen "Alle beobachten" oder "Start" der Kurvenanzeige wird die Onlineverbindung zur CPU, sofern nicht bereits vorhanden, aufgebaut und die Bedienung des Inbetriebnahmefensters freigegeben.

Bedienung der Kurvenanzeige

- Wählen Sie in der Klappliste "Abtastzeit" die gewünschte Abtastzeit aus.
  Alle Werte in der Kurvenanzeige werden in der ausgewählten Abtastzeit aktualisiert.
- Klicken Sie auf das Symbol "Start" in der Gruppe Messung, falls Sie die Kurvenanzeige nutzen möchten.
  Die Aufzeichnung der Werte wird gestartet. In der Kurvenanzeige werden die aktuellen Werte für Sollwert, Istwert und Ausgangswerte für Heizen und Kühlen eingetragen.
- Klicken Sie auf das Symbol "Stop", falls Sie die Kurvenanzeige beenden möchten.
  Die in der Kurvenanzeige aufgezeichneten Werte können weiterhin analysiert werden.

Mit Schließen des Inbetriebnahmefensters wird die Aufzeichnung in der Kurvenanzeige beendet und die aufgezeichneten Werte werden gelöscht.

Erstoptimierung

Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

PID_Temp bietet abhängig von der Konfiguration verschiedene Erstoptimierungsarten:

- Erstoptimierung Heizen
  Es wird ein Sprung am Ausgangswert Heizen ausgegeben, die PID-Parameter für Heizen werden berechnet und anschließend wird im Automatikbetrieb auf den Sollwert geregelt.
- Erstoptimierung Heizen und Kühlen
  Es wird ein Sprung am Ausgangswert Heizen ausgegeben.
  Sobald sich der Istwert in der Nähe des Sollwerts befindet, wird ein Sprung am Ausgangswert Kühlen ausgegeben.
  Die PID-Parameter für Heizen (Struktur Retain.CtrlParams.Heat) und Kühlen (Struktur Retain.CtrlParams.Cool) werden berechnet und anschließend wird im Automatikbetrieb auf den Sollwert geregelt.
- Erstoptimierung Kühlen
  Es wird ein Sprung am Ausgangswert Kühlen ausgegeben.
  Die PID-Parameter für Kühlen werden berechnet und anschließend wird im Automatikbetrieb auf den Sollwert geregelt.

Falls Sie die PID-Parameter für Heizen und Kühlen optimieren wollen, können Sie bei Durchführung einer "Erstoptimierung Heizen" und anschließender "Erstoptimierung Kühlen" ein besseres Regelverhalten erwarten als bei Durchführung einer "Erstoptimierung Heizen und Kühlen". Die Durchführung der Erstoptimierung in zwei Schritten erfordert jedoch mehr Zeit.

Allgemeine Voraussetzungen

- Die Anweisung PID_Temp wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- PID_Temp befindet sich in der Betriebsart "Inaktiv", "Handbetrieb" oder "Automatikbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration Istwertüberwachung (Seite 772)).

Voraussetzungen für Erstoptimierung Heizen

- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz zwischen Obergrenze Istwert und Untergrenze Istwert.
- Der Abstand zwischen Sollwert und Istwert ist größer als 50 % des Sollwerts.
- Der Sollwert ist größer als der Istwert.

Voraussetzungen für Erstoptimierung Heizen und Kühlen

- Der Kühlausgang in den "Grundeinstellungen" ist aktiviert (Config.ActivateCooling = TRUE).
- Die PID-Parameterumschaltung in den "Grundeinstellungen Ausgangswert" ist aktiviert (Config.AdvancedCooling = TRUE).
- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz zwischen Obergrenze Istwert und Untergrenze Istwert.

- Der Abstand zwischen Sollwert und Istwert ist größer als 50 % des Sollwerts.
- Der Sollwert ist größer als der Istwert.

Voraussetzungen für Erstoptimierung Kühlen

- Der Kühlausgang in den "Grundeinstellungen" ist aktiviert (Config.ActivateCooling = TRUE).
- Die PID-Parameterumschaltung in den "Grundeinstellungen Ausgangswert" ist aktiviert (Config.AdvancedCooling = TRUE).
- Eine "Erstoptimierung Heizen" oder "Erstoptimierung Heizen und Kühlen" wurde erfolgreich durchgeführt (PIDSelfTune.SUT.ProcParHeatOk = TRUE). Für alle Optimierungen sollte der gleiche Sollwert verwendet werden.
- Die Differenz zwischen Sollwert und Istwert ist kleiner als 5 % der Differenz zwischen Obergrenze Istwert und Untergrenze Istwert.

Vorgehen

Um eine Erstoptimierung durchzuführen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Temp > Inbetriebnahme".
2. Aktivieren Sie die Schaltfläche "Alle beobachten" oder Starten Sie die Kurvenanzeige. Eine Online-Verbindung wird aufgebaut.
3. Wählen Sie in der Klappliste "Optimierungsart" den gewünschten Erstoptimierungseintrag aus.
4. Klicken Sie auf das Symbol "Start".
   - Die Erstoptimierung wird gestartet.
   - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

   Hinweis

   Klicken Sie auf das Symbol "Stop", wenn der Fortschrittsbalken (Variable "Progress") sich lange Zeit nicht verändert und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID_Temp wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Falls eine Erstoptimierung nicht möglich ist, verhält sich PID_Temp wie unter Verhalten im Fehlerfall konfiguriert.

Nachoptimierung

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden die PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

PID_Temp versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

PID_Temp bietet abhängig von der Konfiguration verschiedene Nachoptimierungsarten:

- Nachoptimierung Heizen:
  PID_Temp erzeugt mit periodischen Änderungen am Ausgangswert Heizen eine Schwingung des Istwerts und berechnet die PID-Parameter für Heizen.
- Nachoptimierung Kühlen:
  PID_Temp erzeugt mit periodischen Änderungen am Ausgangswert Kühlen eine Schwingung des Istwerts und berechnet die PID-Parameter für Kühlen.

Vorübergehender Optimierungsoffset für Heizkühlregler

Wird PID_Temp als Heizkühlregler eingesetzt (Config.ActivateCooling = TRUE), muss der PID Ausgangswert (PidOutputSum) am Sollwert folgende Voraussetzung erfüllen, damit eine Istwertschwingung erzeugt und die Nachoptimierung erfolgreich durchgeführt werden kann:

- Positiver PID Ausgangswert für Nachoptimierung Heizen
- Negativer PID Ausgangswert für Nachoptimierung Kühlen

Ist diese Voraussetzung nicht erfüllt, können Sie einen vorübergehenden Offset für die Nachoptimierung vorgeben, der am entgegengesetzt wirkenden Ausgang ausgegeben wird.

- Offset für Kühlausgang (PIDSelfTune.TIR.OutputOffsetCool) bei Nachoptimierung Heizen. Geben Sie vor dem Start der Optimierung einen negativen Optimierungsoffset Kühlen vor, der kleiner ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand.
- Offset für Heizausgang (PIDSelfTune.TIR.OutputOffsetHeat) bei Nachoptimierung Kühlen Geben Sie vor dem Start der Optimierung einen positiven Optimierungsoffset Heizen vor, der größer ist als der PID Ausgangswert (PidOutputSum) am Sollwert im stationären Zustand.

Der vorgegebene Offset wird dann vom PID-Algorithmus ausgeglichen, sodass der Istwert am Sollwert verbleibt. Durch die Höhe des Offsets kann somit der PID Ausgangswert entsprechend angepasst werden, damit er oben genannte Voraussetzung erfüllt.

Um größere Überschwinger des Istwerts bei Vorgabe des Offsets zu vermeiden, kann dieser auch in mehreren Schritten erhöht werden.

Verlässt PID_Temp die Betriebsart Nachoptimierung, wird der Optimierungsoffset zurückgesetzt.

Beispiel: Vorgabe eines Offset für Nachoptimierung Kühlen

- Ohne Offset
  - Sollwert (Setpoint) = Istwert (ScaledInput) = 80 °C
  - PID Ausgangswert (PidOutputSum) = 30.0
  - Ausgangswert Heizen (OutputHeat) = 30.0
  - Ausgangswert Kühlen (OutputCool) = 0.0
    Eine Schwingung des Istwerts um den Sollwert kann mit dem Kühlausgang alleine nicht erzeugt werden. Die Nachoptimierung würde hier fehlschlagen.
- Mit Offset für Heizausgang (PIDSelfTune.TIR.OutputOffsetHeat) = 80.0
  - Sollwert (Setpoint) = Istwert (ScaledInput) = 80 °C
  - PID Ausgangswert (PidOutputSum) = -50.0
  - Ausgangswert Heizen (OutputHeat) = 80.0
  - Ausgangswert Kühlen (OutputCool) = -50.0
    Durch die Vorgabe eines Offset für den Heizausgang kann der Kühlausgang nun eine Schwingung des Istwerts um den Sollwert erzeugen. Die Nachoptimierung kann somit erfolgreich durchgeführt werden.

Allgemeine Voraussetzungen

- Die Anweisung PID_Temp wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
  Bei eingeschalteter Totzone kann sich eine dauerhafte Regeldifferenz (Abweichung zwischen Sollwert und Istwert) einstellen. Dies kann sich negativ auf die Durchführung der Nachoptimierung auswirken.
- Es werden keine Störungen erwartet.
- PID_Temp befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

Voraussetzungen für Nachoptimierung Heizen

- Heat.EnableTuning = TRUE
- Cool.EnableTuning = FALSE
- Falls PID_Temp als Heizkühlregler konfiguriert ist (Config.ActivateCooling = TRUE), muss am Arbeitspunkt, an dem die Optimierung durchgeführt werden soll, der Heizausgang aktiv sein.
  PidOutputSum > 0.0 (siehe Optimierungsoffset)

Voraussetzungen für Nachoptimierung Kühlen

- Heat.EnableTuning = FALSE
- Cool.EnableTuning = TRUE
- Der Kühlausgang ist aktiviert (Config.ActivateCooling = TRUE).
- Die PID-Parameterumschaltung ist aktiviert (Config.AdvancedCooling = TRUE).
- Der Kühlausgang muss am Arbeitspunkt, an dem die Optimierung durchgeführt werden soll, aktiv sein.
  PidOutputSum < 0.0 (siehe Optimierungsoffset)

Ablauf abhängig von Startsituation

Die Nachoptimierung können Sie aus den Betriebsarten "Inaktiv", "Automatikbetrieb" oder "Handbetrieb" starten.

Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb mit PIDSelfTune.TIR.RunIn = FALSE (Vorbelegung)
  Falls Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.
  PID_Temp regelt so lange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv, Handbetrieb oder Automatikbetrieb mit PIDSelfTune.TIR.RunIn = TRUE
  Es wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert (Zwei-Punkt-Regelung) zu erreichen:
  - mit minimalem oder maximalem Ausgangswert Heizen bei Nachoptimierung Heizen.
  - mit minimalem oder maximalem Ausgangswert Kühlen bei Nachoptimierung Kühlen.

  Dies kann ein erhöhtes Überschwingen verursachen. Ist der Sollwert erreicht startet die Nachoptimierung.
  Falls der Sollwert nicht erreicht werden kann, bricht PID_Temp die Optimierung nicht automatisch ab.

Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Temp > Inbetriebnahme".
2. Aktivieren Sie die Schaltfläche "Alle beobachten" oder Starten Sie die Kurvenanzeige.
   Eine Online-Verbindung wird aufgebaut.
3. Wählen Sie in der Klappliste "Optimierungsart" den gewünschten Nachoptimierungseintrag aus.

4. Geben Sie bei Bedarf (siehe Optimierungsoffset) einen Optimierungsoffset vor und warten Sie, bis der stationäre Zustand wieder erreicht ist.
5. Klicken Sie auf das Symbol "Start".
   – Der Ablauf der Nachoptimierung wird gestartet.
   – Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt.
     Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", falls der Fortschrittsbalken (Variable "Progress") sich lange Zeit nicht verändert und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

Insbesondere in folgenden Phasen erfolgt kein automatischer Abbruch der Optimierung, falls der Sollwert nicht erreicht werden kann.

- "Versuchen Sollwert mit Zwei-Punkt-Regelung Heizen zu erreichen."
- "Versuchen Sollwert mit Zwei-Punkt-Regelung Kühlen zu erreichen."

Ergebnis

Wenn während der Nachoptimierung keine Fehler auftraten, wurden die PID-Parameter optimiert. PID_Temp wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der Nachoptimierung Fehler auftraten, verhält sich PID_Temp wie unter Verhalten im Fehlerfall konfiguriert.

Betriebsart "Handbetrieb"

Im Folgenden wird beschrieben, wie Sie die Betriebsart "Handbetrieb" im Inbetriebnahmefenster des Technologieobjekts "PID_Temp" nutzen können.

Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.

Voraussetzung

- Die Anweisung "PID_Temp" wird in einem Weckalarm-OB aufgerufen.
- Eine Onlineverbindung zur CPU ist aufgebaut.
- Die CPU befindet sich im Betriebszustand "RUN".

Vorgehen

Falls Sie die Regelstrecke durch Vorgabe eines Handwerts testen möchten, verwenden Sie im Inbetriebnahmefenster "Handbetrieb".

Um einen Handwert vorzugeben, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Temp > Inbetriebnahme".
2. Aktivieren sie die Schaltfläche "Alle beobachten" oder starten Sie die Kurvenanzeige. Eine Online-Verbindung wird aufgebaut.
3. Aktivieren Sie im Bereich "Online-Zustand des Reglers" das Kontrollkästchen "Handbetrieb".
   PID_Temp arbeitet im Handbetrieb. Der zuletzt aktuelle Ausgangswert bleibt aktiv.
4. Tragen Sie im editierbaren Feld den Handwert in der Einheit % ein.
   Ist die Kühlung in den Grundeinstellungen aktiviert, geben Sie den Handwert wie folgt ein:
   – um den Wert an den Ausgängen für Heizen auszugeben, geben Sie einen positiven Handwert ein.
   – um den Wert an den Ausgängen für Kühlen auszugeben, geben Sie einen negativen Handwert ein.
5. Klicken Sie auf das Symbol .

Ergebnis

Der Handwert wird in die CPU geschrieben und ist sofort wirksam.

Deaktivieren Sie das Kontrollkästchen "Handbetrieb", wenn der Ausgangswert wieder durch den PID-Regler vorgegeben werden soll.

Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

Ersatzsollwert

Im Folgenden wird beschrieben, wie Sie den Ersatzsollwert im Inbetriebnahmefenster des Technologieobjekts "PID_Temp" nutzen können.

Voraussetzung

- Die Anweisung "PID_Temp" wird in einem Weckalarm-OB aufgerufen.
- Eine Onlineverbindung zur CPU ist aufgebaut.
- Die CPU befindet sich im Betriebszustand "RUN".

Vorgehen

Falls Sie vorübergehend einen anderen Wert als den an Parameter "Setpoint" als Sollwert verwenden möchten (z. B. um einen Slave in einer Kaskade zu optimieren), verwenden Sie im Inbetriebnahmefenster den Ersatzsollwert.

Um einen Ersatzsollwert vorzugeben, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID_Temp > Inbetriebnahme".
2. Aktivieren sie die Schaltfläche "Alle beobachten" oder starten Sie die Kurvenanzeige. Eine Online-Verbindung wird aufgebaut.

3. Aktivieren Sie im Bereich "Online-Zustand des Reglers" das Kontrollkästchen "Subst.Setpoint".
   Der Ersatzsollwert (Variable SubstituteSetpoint) wird mit dem zuletzt aktuellen Sollwert initialisiert und nun verwendet.
4. Tragen Sie im editierbaren Feld den Ersatzsollwert ein.
5. Klicken Sie auf das Symbol .

Ergebnis

Der Ersatzsollwert wird in die CPU geschrieben und ist sofort wirksam.

Deaktivieren Sie das Kontrollkästchen "Subst.Setpoint", wenn als Sollwert wieder der Wert am Parameter "Setpoint" verwendet werden soll.

Der Wechsel erfolgt nicht stoßfrei.

Kaskadeninbetriebnahme

Informationen zur Kaskadeninbetriebnahme mit PID_Temp finden Sie unter Inbetriebnahme (Seite 794).

3.3.3.4 Kaskadenregelung mit PID_Temp

Einleitung

Bei der Kaskadenregelung werden mehrere Regelkreise ineinander geschachtelt. Slaves erhalten dabei ihren Sollwert (Setpoint) aus dem Ausgangswert (OutputHeat) des jeweils überlagerten Master.

Voraussetzung für den Aufbau einer Kaskadenregelung ist, dass sich die Regelstrecke in Teilstrecken mit jeweils einer eigenen Messgröße unterteilen lässt.

Die Sollwertvorgabe für die Regelgröße wird am äußersten Master vorgenommen.

Der Ausgangswert des innersten Slave wird auf das Stellglied gegeben und wirkt damit auf die Regelstrecke.

Folgende wesentliche Vorteile ergeben sich durch den Einsatz einer Kaskadenregelung im Vergleich zum einschleifigen Regelkreis:

- Durch die zusätzlichen unterlagerten Regelkreise werden Störungen, die dort auftreten, schnell ausgeregelt. Ihr Einfluss auf die Regelgröße wird dadurch stark reduziert. Das Störverhalten wird somit verbessert.
- Die unterlagerten Regelkreise wirken linearisierend. Die negativen Auswirkungen solcher Nichtlinearitäten auf die Regelgröße werden dadurch abgemildert.

PID_Temp bietet folgende Funktionalität speziell für den Einsatz in Kaskadenregelungen:

- Vorgabe eines Ersatzsollwertes
- Austausch von Statusinformationen zwischen Master und Slave (z. B. aktuelle Betriebsart)
- Verschiedene Anti-Wind-Up Modi (Reaktion des Master auf Begrenzung seiner Slaves)

Beispiel

Das folgende Blockschaltbild zeigt eine Kaskadenregelung mit PID_Temp am vereinfachten Beispiel einer Schokoladenschmelze:

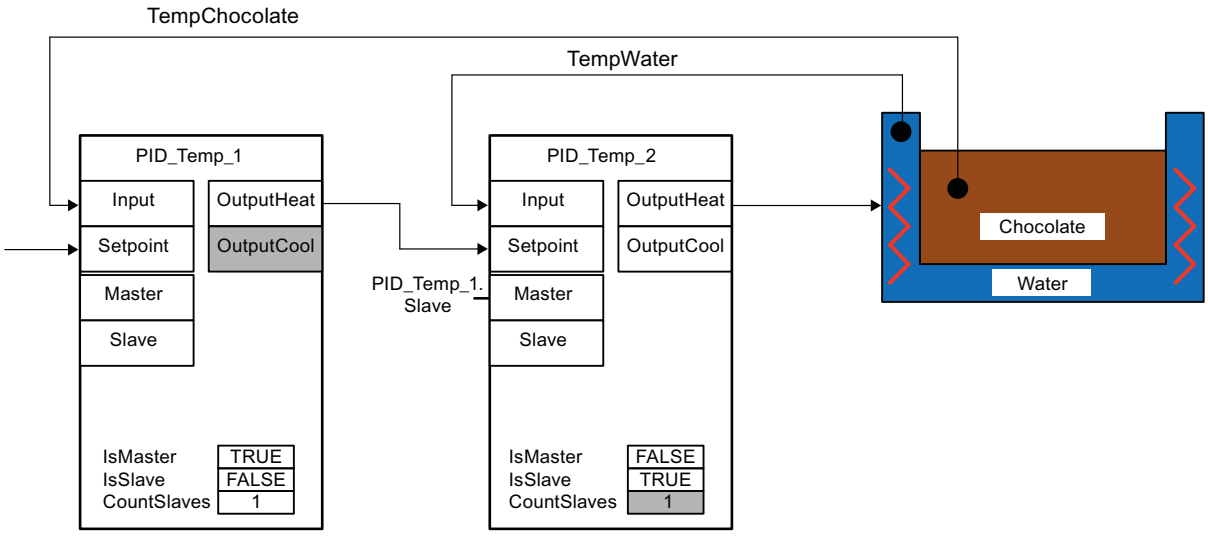

Der Master PID_Temp_1 vergleicht den Istwert der Schokoladentemperatur (TempChocolate) mit der Sollwertvorgabe durch den Anwender an seinem Parameter Setpoint. Sein Ausgangswert OutputHeat bildet den Sollwert des Slave PID_Temp_2.

PID_Temp_2 versucht den Istwert der Wasserbadtemperatur (TempWater) auf diesen Sollwert zu regeln. Der Ausgangswert von PID_Temp_2 wird direkt auf das Stellglied der Regelstrecke (Heizung des Wasserbades) gegeben und beeinflusst so die Wasserbadtemperatur. Die Wasserbadtemperatur wirkt wiederum auf die Schokoladentemperatur.

FAQ

Weitere Informationen hierzu finden Sie im folgenden FAQ im Siemens Industry Online Support:

- Beitrags-ID 103526819 (https://support.industry.siemens.com/cs/ww/de/view/103526819)

Siehe auch

Programmerstellung (Seite 792)

Programmerstellung

Beachten Sie folgende Punkte bei der Programmerstellung:

- Anzahl der PID_Temp-Instanzen
  Es müssen so viele verschiedene PID_Temp-Instanzen in einem Weckalarm-OB aufgerufen werden, wie verkettete Messgrößen im Prozess vorhanden sind.
  Im Beispiel sind zwei verkettete Messgrößen vorhanden: TempChocolate und TempWater. Daher sind zwei PID_Temp-Instanzen notwendig.
- Aufrufreihenfolge
  Ein Master muss vor seinen Slaves im gleichen Weckalarm-OB aufgerufen werden.
  Der äußerste Master, an dem der Anwendersollwert vorgegeben wird, wird zuerst aufgerufen.
  Der Slave, der seinen Sollwert vom äußersten Master vorgegeben bekommt, wird als nächstes aufgerufen usw.
  Der innerste Slave, der mit seinem Ausgangswert auf das Stellglied des Prozesses wirkt, wird als letztes aufgerufen.
  Im Beispiel wird PID_Temp_1 vor PID_Temp_2 aufgerufen.
- Verschaltung der Messgrößen
  Der äußerste Master wird mit der äußersten Messgröße verschaltet, die auf den Anwendersollwert geregelt werden soll.
  Der innerste Slave wird mit der innersten Messgröße verschaltet, die direkt vom Stellglied beeinflusst wird.
  Die Verschaltung der Messgrößen mit PID_Temp erfolgt mit den Parametern Input oder Input_PER.
  Im Beispiel wird die äußere Messgröße TempChocolate mit PID_Temp_1 und die innere Messgröße TempWater mit PID_Temp_2 verschaltet.
- Verschaltung des Ausgangswerts des Master auf den Sollwert des Slave
  Der Ausgangswert (OutputHeat) eines Master muss auf den Sollwert (Setpoint) seines Slave zugewiesen werden.
  Diese Verschaltung können Sie manuell im Programmiereditor vornehmen oder automatisch im Inspektorfenster des Slave in den Grundeinstellungen über die Auswahl des Master vornehmen lassen.
  Bei Bedarf können Sie eigene Filter- oder Skalierungsfunktionen einfügen z .B. um den Ausgangswertbereich des Master an den Sollwert-/Istwertbereich des Slave anzupassen.
  Im Beispiel wird OutputHeat von PID_Temp_1 auf Setpoint von PID_Temp_2 zugewiesen.
- Verschaltung der Schnittstelle zum Informationsaustausch zwischen Master und Slave
  Der Parameter "Slave" eines Master muss bei all seinen direkt untergeordneten Slaves (die ihren Sollwert von diesem Master erhalten) auf deren Parameter "Master" zugewiesen werden. Um die Verschaltung eines Master mit mehreren Slaves und die Anzeige der Verschaltung im Inspektorfenster des Slave in den Grundeinstellungen zu ermöglichen, sollte die Zuweisung über die Schnittstelle des Slave vorgenommen werden.
  Diese Verschaltung können Sie manuell im Programmiereditor vornehmen oder automatisch im Inspektorfenster des Slave in den Grundeinstellungen über die Auswahl des Master vornehmen lassen.
  Nur falls diese Verschaltung vorgenommen ist, können die Anti-Wind-Up-Funktionalität und die Auswertung der Slave-Betriebsarten beim Master korrekt arbeiten.
  Im Beispiel wird Parameter "Slave" von PID_Temp_1 auf Parameter "Master" von PID_Temp_2 zugewiesen.

Programmcode des Beispiels in der Sprache SCL (ohne Zuweisung des Ausgangswerts des Slave auf das Stellglied):

```
"PID_Temp_1"(Input:="TempChocolate");
"PID_Temp_2"(Input:="TempWater", Master := "PID_Temp_1".Slave,
Setpoint := "PID_Temp_1".OutputHeat);
```

Konfiguration

Die Konfiguration können Sie über Ihr Anwenderprogramm, den Konfigurationseditor oder das Inspektorfenster des PID_Temp Aufrufes durchführen.

Achten Sie beim Einsatz von PID_Temp in einer Kaskadenregelung auf die korrekte Konfiguration der im Folgenden erläuterten Einstellungen.

Erhält eine PID_Temp-Instanz ihren Sollwert von einem übergeordneten Master und gibt diese ihren Ausgangswert an einen untergeordneten Slave weiter, ist diese PID_Temp-Instanz gleichzeitig sowohl Master als auch Slave. Für eine solche PID_Temp-Instanz sind beide unten aufgeführte Konfigurationen vorzunehmen. Dies ist zum Beispiel für die mittlere PID_Temp-Instanz bei einer Kaskadenregelung mit drei verketteten Messgrößen und drei PID_Temp-Instanzen der Fall.

Konfiguration eines Master

Einstellung im Konfigurationseditor oder Inspektorfenster	DB-Parameter	Erläuterung
Grundeinstellungen → Kaskade: Optionskästchen "Regler ist Master" aktivieren	Config.Cascade.IsMaster = TRUE	Aktiviert diesen Regler als Master in einer Kaskade
Grundeinstellungen → Kaskade: Anzahl der Slaves	Config.Cascade.CountSlaves	Anzahl der direkt untergeordneten Slaves, die ihren Sollwert von diesem Master erhalten
Grundeinstellungen → Eingangs-/Ausgangsparameter: Auswahl des Ausgangswerts (Heizen) = OutputHeat	Config.Output.Heat.Select = 0	Der Master verwendet nur den Ausgangsparameter OutputHeat. OutputHeat_PWM und OutputHeat_PER sind deaktiviert.
Grundeinstellungen → Eingangs-/Ausgangsparameter: Optionskästchen "Kühlung aktivieren" deaktivieren	Config.ActivateCooling = FALSE	Bei einem Master muss die Kühlung deaktiviert sein.

Einstellung im Konfigurationseditor oder Inspektorfenster	DB-Parameter	Erläuterung
Ausgangseinstellungen → Ausgangswertgrenzen und –skalierung → OutputHeat / OutputCool: Untergrenze PID Ausgangswert (Heizen), Obergrenze PID Ausgangswert (Heizen), skalierter unterer Ausgangswert (Heizen), skalierter oberer Ausgangswert (Heizen)	Config.Output.Heat.PidLowerLimit, Config.Output.Heat.PidUpperLimit, Config.Output.Heat.LowerScaling, Config.Output.Heat.UpperScaling	Wird bei der Zuweisung von OutputHeat des Master auf Setpoint des Slave keine eigene Skalierungsfunktion eingesetzt, kann es notwendig sein, die Ausgangswertgrenzen und –skalierung des Master an den Sollwert-/Istwertbereich des Slave anzupassen.
Diese Variable ist nicht im Inspektorfenster oder in der Funktionssicht des Konfigurationseditors verfügbar. Sie können sie über die Parametersicht des Konfigurationseditors verändern.	Config.Cascade.AntiWindUpMode	Der Anti-Wind-Up-Modus bestimmt, wie der I-Anteil dieses Master behandelt wird, falls direkt untergeordnete Slaves ihre Ausgangswertgrenzen erreichen. Möglich sind: • AntiWindUpMode = 0: Die AntiWindUp Funktionalität ist deaktiviert. Der Master reagiert nicht auf Begrenzung seiner Slaves. • AntiWindUpMode = 1 (default): Der I-Anteil des Master wird im Verhältnis "Slaves in Begrenzung / Anzahl der Slaves" reduziert. Dadurch werden die Auswirkungen der Begrenzung auf das Regelverhalten verringert. • AntiWindUpMode = 2: Der I-Anteil des Master wird gehalten, sobald sich ein Slave in Begrenzung befindet.

Konfiguration eines Slave

Einstellung im Konfigurationseditor oder Inspektorfenster	DB-Parameter	Erläuterung
Grundeinstellungen → Kaskade: Optionskästchen "Regler ist Slave" aktivieren	Config.Cascade.IsSlave = TRUE	Aktiviert diesen Regler als Slave in einer Kaskade

Inbetriebnahme

Nach dem Übersetzen und Laden des Programms können Sie die Inbetriebnahme der Kaskadenregelung starten.

Beginnen Sie bei der Inbetriebnahme (Durchführung einer Optimierung oder Wechsel in den Automatikbetrieb mit vorhandenen PID-Parametern) mit dem innersten Slave und fahren Sie nach außen fort, bis der äußerste Master erreicht ist.

Im obigen Beispiel startet die Inbetriebnahme mit PID_Temp_2 und wird anschließend mit PID_Temp_1 fortgeführt.

Optimierung des Slave

Die Optimierung von PID_Temp erfordert einen konstanten Sollwert. Aktivieren Sie daher für die Optimierung eines Slave dessen Ersatzsollwert (Variablen SubstituteSetpoint und SubstituteSetpointOn) oder versetzen Sie den zugehörigen Master in den Handbetrieb mit entsprechendem Handwert. Auf diese Weise ist sichergestellt, dass der Sollwert des Slave während der Optimierung konstant bleibt.

Optimierung des Master

Damit ein Master den Prozess beeinflussen oder eine Optimierung durchführen kann, müssen sich alle nachfolgenden Slaves im Automatikbetrieb befinden und den Ersatzsollwert deaktiviert haben. Über die Schnittstelle zum Informationsaustausch zwischen Master und Slave (Parameter Master und Parameter Slave) wertet ein Master diese Bedingungen aus und zeigt den aktuellen Zustand an den Variablen AllSlaveAutomaticState und NoSlaveSubstituteSetpoint an. Im Inbetriebnahmeeditor werden entsprechende Statusmeldungen ausgegeben.

Statusmeldung im Inbetriebnahmeeditor des Master	DB-Parameter des Master	Behebung
Ein oder mehrere Slaves sind nicht im Automatikbetrieb.	AllSlaveAutomaticState = FALSE, NoSlaveSubstituteSetpoint = TRUE	Führen Sie zuerst die Inbetriebnahme aller nachfolgenden Slaves durch. Stellen Sie sicher, dass folgende Bedingungen erfüllt sind, bevor Sie eine Optimierung durchführen oder Handbetrieb oder Automatikbetrieb des Master aktivieren: • Alle nachfolgenden Slaves befinden sich im Automatikbetrieb (State = 3). • Alle nachfolgenden Slaves haben den Ersatzsollwert deaktiviert (SubstituteSetpointOn = FALSE).
Ein oder mehrere Slaves haben den Ersatzsollwert aktiviert.	AllSlaveAutomaticState = TRUE, NoSlaveSubstituteSetpoint = FALSE	
Ein oder mehrere Slaves sind nicht im Automatikbetrieb und haben den Ersatzsollwert aktiviert.	AllSlaveAutomaticState = FALSE, NoSlaveSubstituteSetpoint = FALSE	

Wird für einen Master eine Erstoptimierung oder Nachoptimierung gestartet, bricht PID_Temp in folgenden Fällen die Optimierung ab und zeigt einen Fehler mit ErrorBits = DW#16#0200000 an:

- ein oder mehrere Slaves sind nicht im Automatikbetrieb (AllSlaveAutomaticState = FALSE)
- ein oder mehrere Slaves haben den Ersatzsollwert aktiviert (NoSlaveSubstituteSetpoint = FALSE).

Der anschließende Betriebsartenwechsel ist abhängig von ActivateRecoverMode.

Ersatzsollwert

Für die Vorgabe eines Sollwerts bietet PID_Temp, zusätzlich zum Parameter Setpoint, einen Ersatzsollwert an der Variablen SubstituteSetpoint. Dieser kann mit SubstituteSetpointOn = TRUE oder durch Aktivieren des entsprechenden Optionskästchens im Inbetriebnahmeeditor aktiviert werden.

Mit dem Ersatzsollwert können Sie den Sollwert vorübergehend direkt am Slave vorgeben, z. B. für die Inbetriebnahme oder Optimierung.

Dabei muss die, für den normalen Betrieb der Kaskadenregelung erforderliche, Verschaltung des Ausgangswerts des Master auf den Sollwert des Slave im Programm nicht verändert werden

Damit ein Master den Prozess beeinflussen oder eine Optimierung durchführen kann, müssen alle nachfolgenden Slave den Ersatzsollwert deaktiviert haben.

Den aktuell wirksamen Sollwert, wie er vom PID-Algorithmus zur Berechnung verwendet wird, können Sie an der Variablen CurrentSetpoint beobachten.

Betriebsarten und Fehlerreaktion

Master oder Slave einer PID_Temp-Instanz verändern die Betriebsart dieser PID_Temp-Instanz nicht.

Falls bei einem seiner Slaves ein Fehler auftritt, bleibt der Master in seiner aktuellen Betriebsart.

Falls bei seinem Master ein Fehler auftritt, bleibt der Slave in seiner aktuellen Betriebsart. Jedoch hängt der weitere Betrieb des Slave dann vom Fehler und von der konfigurierten Fehlerreaktion des Master ab, da der Ausgangswert des Master als Sollwert des Slave verwendet wird:

- Ist beim Master ActivateRecoverMode = TRUE konfiguriert und der Fehler verhindert die Berechnung von OutputHeat nicht, wirkt sich der Fehler nicht auf den Slave aus.
- Ist beim Master ActivateRecoverMode = TRUE konfiguriert und der Fehler verhindert die Berechnung von OutputHeat, gibt der Master abhängig von SetSubstituteOutput den letzten gültigen Ausgangswert oder den konfigurierten Ersatzausgangswert SubstituteOutput aus. Dieser wird dann vom Slave als Sollwert verwendet. PID_Temp ist so voreingestellt, dass in diesem Fall der Ersatzausgangswert 0.0 ausgegeben wird (ActivateRecoverMode = TRUE, SetSubstituteOutput = TRUE, SubstituteOutput = 0.0). Konfigurieren Sie für Ihre Anwendung einen passenden Ersatzausgangswert oder aktivieren Sie die Verwendung des letzten gültigen PID Ausgangswerts (SetSubstituteOutput = FALSE).
- Ist beim Master ActivateRecoverMode = FALSE konfiguriert, wechselt der Master im Fehlerfall in die Betriebsart "Inaktiv" und gibt OutputHeat = 0.0 aus. Der Slave verwendet dann 0.0 als Sollwert.

Die Fehlerreaktion finden Sie in den Ausgangseinstellungen im Konfigurationseditor.

3.3.3.5 Mehrzonenregelung mit PID_Temp

Einleitung

Bei einer Mehrzonenregelung werden mehrere Teilbereiche, sogenannte Zonen, einer Anlage gleichzeitig auf unterschiedliche Temperaturen geregelt. Kennzeichnend für die Mehrzonenregelung ist die gegenseitige Beeinflussung der Temperaturzonen durch thermische Kopplung d. h. der Istwert einer Zone kann den Istwert einer anderen Zone durch Wärmekopplung beeinflussen. Wie stark sich diese Beeinflussung auswirkt, ist abhängig vom Aufbau der Anlage und den gewählten Arbeitspunkten der Zonen.

Beispiel: Extrusionsanlage, wie sie unter anderem in der Kunststoffverarbeitung eingesetzt wird.

Das Stoffgemisch, das den Extruder durchläuft, muss für eine optimale Verarbeitung auf unterschiedliche Temperaturen geregelt werden. So können an der Befüllung des Extruders andere Temperaturen gefordert sein als an seiner Auslassdüse. Die einzelnen Temperaturzonen beeinflussen sich dabei durch Wärmekopplung gegenseitig.

Beim Einsatz von PID_Temp in Mehrzonenregelungen wird jede Temperaturzone von einer eigenen PID_Temp-Instanz geregelt.

Beachten Sie die folgenden Erläuterungen, falls Sie PID_Temp in einer Mehrzonenregelung einsetzen.

Getrennte Erstoptimierung für Heizen und Kühlen

Die Erstinbetriebnahme einer Anlage beginnt in der Regel mit der Durchführung einer Erstoptimierung, um eine erste Einstellung der PID-Parameter vorzunehmen und zum Arbeitspunkt zu regeln. Die Erstoptimierung für Mehrzonenregelungen wird dabei häufig gleichzeitig für alle Zonen vorgenommen.

PID_Temp bietet für Regler mit aktivierter Kühlung und PID-Parameterumschaltung als Methode für Heizen/Kühlen (Config.ActivateCooling = TRUE, Config.AdvancedCooling = TRUE) die Möglichkeit die Erstoptimierung für Heizen und Kühlen in einem Schritt durchzuführen (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = TRUE).

Es wird jedoch empfohlen diese Optimierung bei gleichzeitiger Erstoptimierung mehrerer PID_Temp-Instanzen in einer Mehrzonenregelung nicht zu verwenden. Führen Sie stattdessen die Erstoptimierung für Heizen (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = FALSE) und die Erstoptimierung für Kühlen (Mode = 1, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE) getrennt aus.

Die Erstoptimierung für Kühlen sollte erst gestartet werden, wenn alle Zonen die Erstoptimierung für Heizen abgeschlossen und ihren Arbeitspunkt erreicht haben.

Auf diese Weise werden gegenseitige Beeinflussungen durch thermische Kopplungen zwischen den Zonen während der Optimierung vermindert.

Anpassung der Verzugszeit

Wird PID_Temp in einer Mehrzonenregelung mit starken thermischen Kopplungen zwischen den Zonen eingesetzt, sollten Sie sicherstellen, dass die Anpassung der Verzugszeit für die Erstoptimierung mit PIDSelfTune.SUT.AdaptDelayTime = 0 deaktiviert ist. Sonst kann die Ermittlung der Verzugszeit verfälscht werden, wenn während der Anpassung der Verzugszeit (Heizung ist in dieser Phase deaktiviert) die Abkühlung dieser Zone durch Wärmezufluss von anderen Zonen verhindert wird.

Vorübergehende Abschaltung der Kühlung

PID_Temp bietet die Möglichkeit für Regler mit aktivierter Kühlung (Config.ActivateCooling = TRUE) die Kühlung im Automatikbetrieb vorübergehend zu deaktivieren, indem DisableCooling = TRUE gesetzt wird.

Auf diese Weise können Sie während der Inbetriebnahme verhindern, dass dieser Regler im Automatikbetrieb kühlt, während die Regler anderer Zonen die Optimierung der Heizung noch nicht abgeschlossen haben. Auf Grund der thermischen Kopplung zwischen den Zonen kann sonst die Optimierung negativ beeinflusst werden.

Vorgehen

Bei der Inbetriebnahme von Mehrzonenregelungen mit relevanten thermischen Kopplungen können Sie wie folgt vorgehen:

1. Setzen Sie für alle Regler mit aktivierter Kühlung DisableCooling = TRUE.
2. Setzen Sie für alle Regler PIDSelfTune.SUT.AdaptDelayTime = 0.
3. Geben Sie die gewünschten Sollwerte vor (Parameter Setpoint) und starten Sie für alle Regler gleichzeitig die Erstoptimierung für Heizen (Mode = 1, Heat.EnableTuning = TRUE, Cool.EnableTuning = FALSE).
4. Warten Sie bis alle Regler die Erstoptimierung Heizen abgeschlossen haben.
5. Setzen Sie für alle Regler mit aktivierter Kühlung DisableCooling = FALSE.
6. Warten Sie bis die Istwerte aller Zonen stetig und nahe dem jeweilgen Sollwert sind. Kann für eine Zone der Sollwert dauerhaft nicht erreicht werden, ist das Heiz- bzw. Kühlstellglied zu schwach ausgelegt.
7. Starten Sie für alle Regler mit aktivierter Kühlung die Erstoptimierung für Kühlen (Mode = 1, Heat.EnableTuning = FALSE, Cool.EnableTuning = TRUE).

Hinweis

Grenzwertüberschreitung des Istwerts

Wird die Kühlung im Automatikbetrieb mit DisableCooling = TRUE deaktiviert, kann dies dazu führen, dass der Istwert den Sollwert und die Istwertgrenzen überschreitet solange DisableCooling = TRUE. Beobachten Sie die Istwerte und und greifen Sie ggf. ein, falls Sie DisableCooling nutzen.

Hinweis

Mehrzonenregelungen

Bei Mehrzonenregelungen können die thermischen Kopplungen zwischen den Zonen zu verstärkten Überschwingern, dauerhaftem oder vorübergehendem Überschreiten von Grenzwerten und dauerhaften oder vorübergehenden Regelabweichungen während der Inbetriebnahme und im laufenden Betrieb führen. Beobachten Sie die Istwerte und seien Sie bereit einzugreifen. Je nach Anlage kann es notwendig sein, von der oben beschriebenen Vorgehensweise abzuweichen.

Synchronisierung mehrerer Nachoptimierungen

Wird die Nachoptimierung aus dem Automatikbetrieb heraus mit PIDSelfTune.TIR.RunIn = FALSE gestartet, versucht PID_Temp den Sollwert mit PID-Regelung und den aktuellen PID-Parametern zu erreichen. Erst wenn der Sollwert erreicht ist, startet die eigentliche Optimierung. Die Zeit, die zum Erreichen des Sollwerts notwendig ist, kann für die einzelnen Zonen einer Mehrzonenregelung unterschiedlich sein.

Falls Sie die Nachoptimierung für mehrere Zonen gleichzeitig durchführen wollen, bietet PID_Temp die Möglichkeit diese zu synchronisieren, indem nach Erreichen des Sollwerts mit den weiteren Optimierungsschritten gewartet wird.

Vorgehen

Auf diese Weise können Sie sicherstellen, dass alle Regler ihren Sollwert erreicht haben, wenn die eigentlichen Optimierungsschritte starten. Dadurch werden gegenseitige Beeinflussungen durch thermische Kopplungen zwischen den Zonen während der Optimierung vermindert.

Gehen Sie bei Reglern, für deren Zonen Sie die Nachoptimierung gleichzeitig durchführen wollen, wie folgt vor:

1. Setzen Sie PIDSelfTune.TIR.WaitForControlIn = TRUE für alle Regler.
   Diese Regler müssen sich im Automatikbetrieb mit PIDSelfTune.TIR.RunIn = FALSE befinden.
2. Geben Sie die gewünschten Sollwerte vor (Parameter Setpoint) und starten Sie die Nachoptimierung für alle Regler.
3. Warten Sie bis PIDSelfTune.TIR.ControlInReady = TRUE bei allen Reglern gesetzt ist.
4. Setzen Sie PIDSelfTune.TIR.FinishControlIn = TRUE für alle Regler.

Alle Regler starten somit gleichzeitig die eigentliche Optimierung.

3.3.3.6 Ablöseregelung mit PID_Temp

Ablöseregelung

Bei einer Ablöseregelung teilen sich zwei oder mehr Regler ein gemeinsames Stellglied. Zu jedem Zeitpunkt hat nur ein Regler Zugriff auf das Stellglied und wirkt auf den Prozess.

Eine Logik entscheidet welcher Regler Zugriff auf das Stellglied erhält. Häufig erfolgt diese Entscheidung anhand eines Vergleichs der Ausgangswerte aller Regler z. B. erhält bei einer Maximalauswahl der Regler mit dem größten Ausgangswert den Zugriff auf das Stellglied.

Die Auswahl anhand des Ausgangswerts macht es notwendig, dass alle Regler im Automatikbetrieb arbeiten. Die Regler, die nicht auf das Stellglied wirken, werden nachgeführt. Dies ist notwendig um Windup-Effekte und deren negative Auswirkungen auf das Regelverhalten und die Umschaltung zwischen den Reglern zu vermeiden.

PID_Temp unterstützt ab Version 1.1 Ablöseregelungen indem der Regler ein einfaches Verfahren zum Nachführen der nicht aktiven Regler bietet: Über die Variablen OverwriteInitialOutputValue und PIDCtrl.PIDInit können Sie den I-Anteil des Reglers im Automatikbetrieb so vorbelegen lassen, als ob der PID-Algorithmus im letzten Zyklus für den PID Ausgangswert PidOutputSum = OverwriteInititalOutputValue berechnet hätte. OverwriteInitialOutputValue wird dazu mit dem PID Ausgangswert des Reglers verschaltet, der gerade Zugriff auf das Stellglied hat. Durch Setzen des Bits PIDCtrl.PIDInit stoßen Sie die Vorbelegung des I-Anteils an, sowie den Neustart des Reglertakts und der PWM-Periode. Die anschließende Berechnung des PID Ausgangswerts im aktuellen Zyklus erfolgt anhand des vorbelegten (und für alle Regler angeglichenen) I-Anteils sowie des P- und I-Anteils aus der aktuellen Regeldifferenz. Der D-Anteil ist während des Aufrufs mit PIDCtrl.PIDInit = TRUE nicht aktiv und trägt damit nicht zum Ausgangswert bei.

Dieses Verfahren stellt sicher, dass die Berechnung des aktuellen PID Ausgangswerts und damit die Entscheidung welcher Regler Zugriff auf das Stellglied erhält, nur anhand des aktuellen Prozesszustands und der PI-Parameter erfolgt. Windup-Effekte bei den nicht aktiven Reglern und damit falsche Entscheidungen der Umschaltlogik werden somit verhindert.

Voraussetzung

- PIDCtrl.PIDInit ist nur wirksam, wenn der I-Anteil aktiviert ist (Variablen Retain.CtrlParams.Heat.Ti und Retain.CtrlParams.Cool.Ti > 0.0).
- Sie müssen PIDCtrl.PIDInit und OverwriteInitialOutputValue in Ihrem Anwenderprogramm selbst belegen (siehe Beispiel unten). PID_Temp nimmt keine automatische Änderung an diesen Variablen vor.
- PIDCtrl.PIDInit ist nur wirksam, wenn sich PID_Temp im Automatikbetrieb (Parameter State = 3) befindet.
- Falls möglich, wählen Sie die Abtastzeit PID-Algorithmus (Variablen Retain.CtrlParams.Heat.Cycle und Retain.CtrlParams.Cool.Cycle) so, dass sie für alle Regler identisch ist und rufen Sie alle Regler im gleichen Weckalarm-OB auf. Auf diese Weise stellen Sie sicher, dass die Umschaltung nicht innerhalb eines Reglertakts bzw. einer PWM-Periode erfolgt.

Hinweis

Ständige Adaption der Ausgangswertgrenzen

Statt dem hier beschriebenen aktiven Nachführen der Regler ohne Zugriff auf das Stellglied wird dies bei anderen Reglersystemen alternativ über eine ständige Adaption der Ausgangswertgrenzen realisiert.

Dies ist mit PID_Temp nicht möglich, da eine Änderung der Ausgangswertgrenzen im Automatikbetrieb nicht unterstützt wird.

Beispiel: Regelung eines großen Kessels

PID_Temp wird für die Regelung eines großen Kessels eingesetzt.

Hauptziel ist, die Temperatur Input1 zu regeln. Hierfür wird der Regler PID_Temp_1 eingesetzt. Zusätzlich soll mit dem Begrenzungsregler PID_Temp_2 die Temperatur Input2 an einer zusätzlichen Messstelle unterhalb einer Obergrenze gehalten werden.

Beide Temperaturen werden über eine einzige Heizung beeinflusst. Der Ausgangswert der Regler entspricht der Heizleistung.

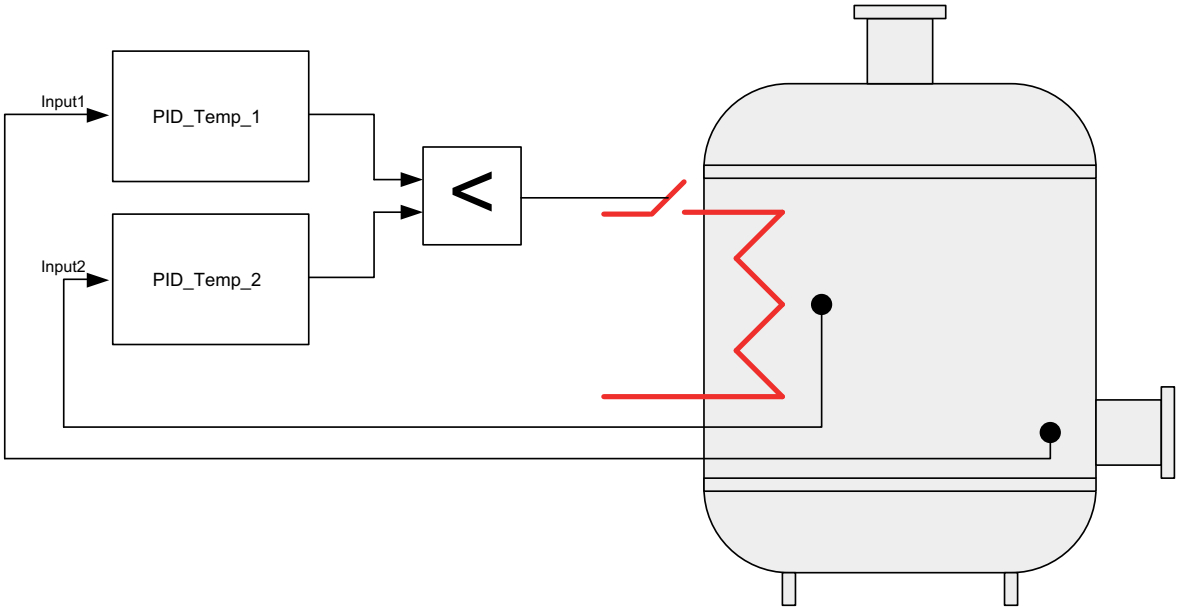

Die Heizung wird über den pulsweitenmodulierten Ausgangswert von PID_Temp (Parameter OutputHeat_PWM) angesteuert, indem die Programmvariable ActuatorInput beschrieben wird. Die Vorgabe des Sollwerts für die Temperatur Input1 erfolgt an Parameter PID_Temp_1.Setpoint. Die Temperaturobergrenze für die zusätzliche Messstelle wird als Sollwert am Parameter PID_Temp_2.Setpoint vorgegeben.

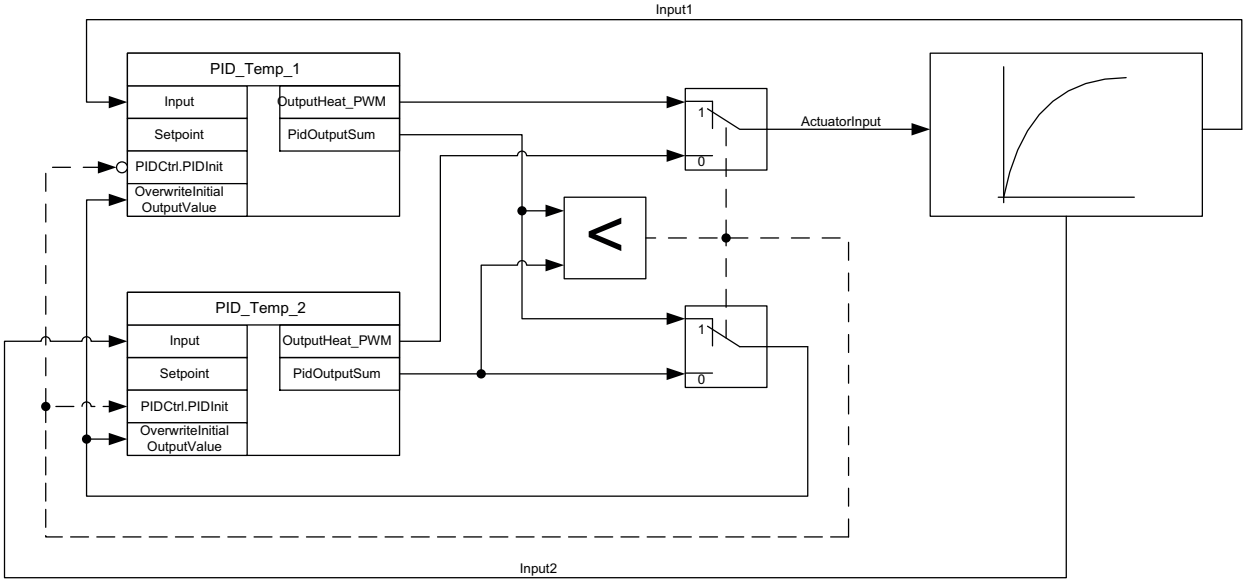

Beiden Reglern steht nur eine Heizung als gemeinsames Stellglied zur Verfügung. Die Logik zur Entscheidung, welcher Regler Zugriff auf das Stellglied erhält, wird hier über eine Minimalauswahl des PID Ausgangswerts (im Real-Format, Parameter PidOutputSum) realisiert. Da der PID Ausgangswert der Heizleistung entspricht, erhält somit der Regler die Kontrolle, der die kleinere Heizleistung fordert.

Im Normalbetrieb der Anlage entspricht der Istwert der Hauptregelgröße dem Sollwert. Der Hauptregler PID_Temp_1 hat sich auf einen stationären PID Ausgangswert PID_Temp_1.PidOutputSum eingeschwungenen. Der Istwert des Begrenzungsreglers Input2 liegt im Normalbetrieb deutlich unter der Obergrenze, die als Sollwert für PID_Temp_2 vorgegeben ist. Deshalb möchte der Begrenzungsregler die Heizleistung steigern, um seinen Istwert zu erhöhen, d. h. er wird einen PID Ausgangswert PID_Temp_2.PidOutputSum berechnen der größer ist als der des Hauptreglers PID_Temp_1.PidOutputSum. Die Minimalauswahl der Umschaltlogik lässt deshalb den Hauptregler PID_Temp_1 weiter auf das Stellglied zugreifen. Zusätzlich wird über die Zuweisungen PID_Temp_2.OverwriteInitialOutputValue = PID_Temp_1.PidOutputSum und PID_Temp_2.PIDCtrl.PIDInit = TRUE sichergestellt, dass PID_Temp_2 nachgeführt wird.

Nähert sich nun Input2 der Obergrenze an oder überschreitet diese, z. B. auf Grund einer Störung, wird der Begrenzungsregler PID_Temp_2 einen kleineren PID Ausgangswert berechnen, um die Heizleistung zu drosseln und so Input2 zu reduzieren. Ist PID_Temp_2.PidOutputSum kleiner als PID_Temp_1.PidOutputSum, erhält der Begrenzungsregler PID_Temp_2 über die Minimalauswahl Zugriff auf das Stellglied und reduziert die Heizleistung. Über die Zuweisungen PID_Temp_1.OverwriteInitialOutputValue = PID_Temp_2.PidOutputSum und PID_Temp_1.PIDCtrl.PIDInit = TRUE wird sichergestellt, dass PID_Temp_1 nachgeführt wird.

Die Temperatur an der zusätzlichen Messstelle Input2 fällt. Die Temperatur der Hauptregelgröße Input1 fällt ebenfalls und kann nicht mehr am Sollwert gehalten werden.

Steht die Störung nicht mehr an, wird Input2 weiter fallen und die Heizleistung vom Begrenzungsregler wird weiter erhöht. Sobald der Hauptregler eine kleinere Heizleistung als Ausgangswert berechnet, kehrt die Anlage in den Normalbetrieb zurück, sodass der

Hauptregler PID_Temp_1 wieder Zugriff auf das Stellglied erhält. Dieses Beispiel kann mit folgendem SCL-Programmcode realisiert werden:

```
"PID_Temp_1"(Input := "Input1");
"PID_Temp_2"(Input := "Input2");
IF "PID_Temp_1".PidOutputSum <= "PID_Temp_2".PidOutputSum THEN
  "ActuatorInput" := "PID_Temp_1".OutputHeat_PWM;
  "PID_Temp_1".PIDCtrl.PIDInit := FALSE;
  "PID_Temp_2".PIDCtrl.PIDInit := TRUE;
  "PID_Temp_2".OverwriteInitialOutputValue := "PID_Temp_1".PidOutputSum;
ELSE
  "ActuatorInput" := "PID_Temp_2".OutputHeat_PWM;
  "PID_Temp_1".PIDCtrl.PIDInit := TRUE;
  "PID_Temp_2".PIDCtrl.PIDInit := FALSE;
  "PID_Temp_1".OverwriteInitialOutputValue := "PID_Temp_2".PidOutputSum;
END_IF;
```

3.3.3.7 PID_Temp mit PLCSIM simulieren

Hinweis

Simulation mit PLCSIM

Die Simulation von PID_Temp mit PLCSIM für CPU S7-1200 wird nicht unterstützt.

PID_Temp kann nur für CPU S7-1500 mit PLCSIM simuliert werden.

Bei der Simulation mit PLCSIM ist das zeitliche Verhalten der simulierten PLC nicht exakt identisch zu einer "echten" PLC. Der tatsächliche Zeittakt eines Weckalarm-OB kann bei einer simulierten PLC größere Schwankungen aufweisen als bei "echten" PLCs.

In der Standardkonfiguration ermittelt PID_Temp die Zeit zwischen den Aufrufen automatisch und überwacht diese auf Schwankungen.

Bei der Simulation von PID_Temp mit PLCSIM kann deshalb ein Abtastzeitfehler (ErrorBits = DW#16#00000800) erkannt werden.

Dies führt zum Abbruch von laufenden Optimierungen.

Die Reaktion im Automatikbetrieb ist abhängig vom Wert der Variable ActivateRecoverMode.

Um dies zu verhindern, sollten Sie PID_Temp bei Simulation mit PLCSIM wie folgt konfigurieren:

- CycleTime.EnEstimation = FALSE
- CycleTime.EnMonitoring = FALSE
- CycleTime.Value: Weisen Sie dieser Variablen den Zeittakt des aufrufenden Weckalarm-OB in der Einheit Sekunden zu.

3.4 PID-Regelung (S7-300, S7-400, S7-1500)

3.4.1 PID Basisfunktionen einsetzen

3.4.1.1 CONT_C

Technologieobjekt CONT_C

Das Technologieobjekt CONT_C stellt einen kontinuierlichen PID-Regler für Automatik- und Handbetrieb zur Verfügung. Es entspricht dem Instanz-Datenbaustein der Anweisung CONT_C. Mit der Anweisung PULSEGEN können Sie einen Impulsregler konfigurieren.

Der Proportional-, Integral (INT) und Differentialanteil (DIF) sind parallel geschaltet und einzeln zu- und abschaltbar. Damit lassen sich P-, I, PI-, PD- und PID-Regler parametrieren.

S7-1500
Alle Parameter und Variablen des Technologieobjekts sind remanent und können beim Laden in Gerät nur geändert werden, wenn Sie CONT_C vollständig laden.

Siehe auch

Übersicht der Software-Regler (Seite 651)

Technologieobjekte hinzufügen (Seite 653)

Technologieobjekte konfigurieren (Seite 654)

Technologieobjekte in Gerät laden (Seite 656)

Regeldifferenz CONT_C konfigurieren

Istwert Peripherie verwenden

Um den Istwert im Peripherieformat am Eingangsparameter PV_PER zu verwenden gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Kontrollkästchen "Peripherie einschalten".
2. Wenn der Istwert als physikalische Größe vorliegt, erfassen Sie Faktor und Offset für die Normierung auf Prozent.
   Der Istwert wird dann nach folgender Formel bestimmt:
   PV = PV_PER × PV_FAC + PV_OFF

Internen Istwert verwenden

Um den Istwert im Gleitpunktformat am Eingangsparameter PV_IN zu verwenden gehen Sie folgendermaßen vor:

1. Deaktivieren Sie das Kontrollkästchen " Peripherie einschalten ".

Regeldifferenz

Sie stellen unter folgender Voraussetzung eine Totzonenbreite ein:

- Das Istwert-Signal ist verrauscht.
- Die Reglerverstärkung ist hoch.
- Der D-Anteil ist aktiviert.

Der Rauschanteil des Istwerts verursacht in diesem Fall starke Schwankungen des Stellwerts. Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Regelalgorithmus CONT_C konfigurieren

Allgemein

Um festzulegen, welche Anteile des Regelalgorithmus aktiviert sind, gehen Sie folgendermaßen vor:

1. Wählen Sie aus der Liste "Reglerstruktur" einen Eintrag.
   Sie können nur die für die gewählte Reglerstruktur benötigten Parameter eingeben.

P-Anteil

1. Wenn die Reglerstruktur einen P-Anteil enthält, geben Sie die "Proportionalverstärkung" ein.

I-Anteil

1. Wenn die Reglerstruktur einen I-Anteil enthält, geben Sie die Integrationszeit ein.
2. Um den I-Anteil mit einem Initialisierungswert zu belegen, aktivieren Sie das Optionskästchen "I-Anteil initialisieren und geben Sie den Initalisierungswert ein.
3. Um den I-Anteil permanent auf diesem Initalisierungswert einzustellen, aktivieren Sie das Optionskästchen "I-Anteil anhalten".

D-Anteil

1. Wenn die Reglerstruktur einen D-Anteil enthält, geben Sie die Differenzierzeit, die Gewichtung D-Anteil und die Verzögerungszeit ein.

Stellwert CONT_C konfigurieren

Allgemein

Sie können CONT_C im Hand- oder im Automatikbetrieb einsetzen.

1. Um einen Handstellwert vorzugeben, aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
   Sie können am Eingangsparameter MAN einen Handstellwert vorgeben.

Stellwertgrenzen

Der Stellwert wird nach oben und untern begrenzt, damit er nur gültige Werte annehmen kann. Sie können die Begrenzung nicht abschalten. Das Überschreiten der Grenzen wird durch die Ausgangsparameter QLMN_HLM und QLMN_LLM angezeigt.

1. Geben Sie einen Wert für die obere und untere Stellwertgrenze ein.
   Wenn der Stellwert eine physikalische Größe ist, müssen die Einheiten für obere und untere Stellwertgrenze übereinstimmen.

Normierung

Der Stellwert kann für die Ausgabe als Gleitpunkt- und Peripheriewert über einen Faktor und ein Offset nach folgender Formel normiert werden.

Normierter Stellwert = Stellwert × Faktor + Offset

Vorbelegt sind ein Faktor von 1.0 und ein Offset von 0.0.

1. Geben Sie einen Wert für Faktor und Offset ein.

Impulsregler programmieren

Mit dem kontinuierlichen Regler CONT_C und dem Impulsformer PULSEGEN kann ein Festwertregler mit schaltendem Ausgang für proportionale Stellglieder realisiert werden. Das folgende Bild zeigt den prinzipiellen Signalverlauf des Regelkreises.

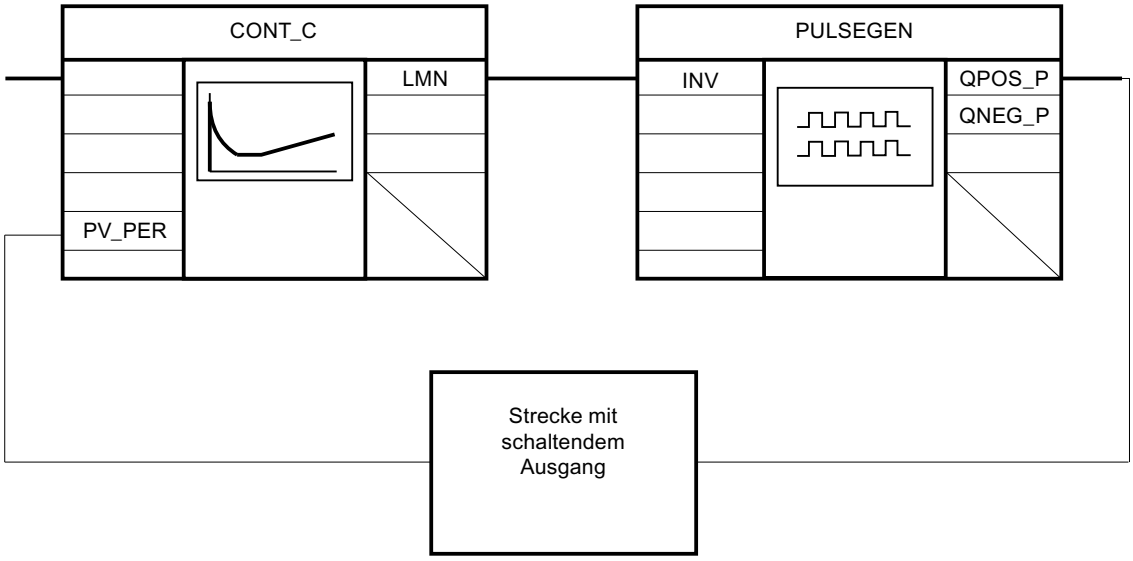

Der kontinuierliche Regler CONT_C bildet den Stellwert LMN, der vom Impulsformer PULSEGEN in Puls-Pausesignale QPOS_P bzw. QNEG_P gewandelt wird.

CONT_C in Betrieb nehmen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Geben Sie in den Feldern "P", "I", "D" und "Verzögerungszeit" neue PID-Parameter ein.
3. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".
4. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
5. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
6. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.

7. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
8. Wiederholen Sie die Schritte 2 bis 6, bis Sie mit dem Regelergebnis zufrieden sind.

3.4.1.2 CONT_S

Technologieobjekt CONT_S

Das Technologieobjekt CONT_S stellt einen Schrittregler für Stellglieder mit integrierendem Verhalten zur Verfügung und dient zum Regeln von technischen Temperatur-Prozessen mit binären Stellwertausgangssignalen. Das Technologieobjekt entspricht dem Instanz-Datenbaustein der Anweisung CONT_S. Die Arbeitsweise basiert auf dem PI-Regelalgorithmus des Abtastreglers. Der Schrittregler arbeitet ohne Stellungsrückmeldung. Es sind Hand- und Automatikbetrieb möglich.

S7-1500
Alle Parameter und Variablen des Technologieobjekts sind remanent und können beim Laden in Gerät nur geändert werden, wenn Sie CONT_S vollständig laden.

Siehe auch

Übersicht der Software-Regler (Seite 651)

Technologieobjekte hinzufügen (Seite 653)

Technologieobjekte konfigurieren (Seite 654)

Technologieobjekte in Gerät laden (Seite 656)

Regeldifferenz CONT_S konfigurieren

Istwert Peripherie verwenden

Um den Istwert im Peripherieformat am Eingangsparameter PV_PER zu verwenden gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Kontrollkästchen "Peripherie einschalten".
2. Wenn der Istwert als physikalische Größe vorliegt, erfassen Sie Faktor und Offset für die Normierung auf Prozent.
   Der Istwert wird dann nach folgender Formel bestimmt:
   PV = PV_PER × PV_FAC + PV_OFF

Internen Istwert verwenden

Um den Istwert im Gleitpunktformat am Eingangsparameter PV_IN zu verwenden gehen Sie folgendermaßen vor:

1. Deaktivieren Sie das Kontrollkästchen "Peripherie einschalten".

Regeldifferenz

Sie stellen unter folgender Voraussetzung eine Totzonenbreite ein:

- Das Istwert-Signal ist verrauscht.
- Die Reglerverstärkung ist hoch.
- Der D-Anteil ist aktiviert.

Der Rauschanteil des Istwerts verursacht in diesem Fall starke Schwankungen des Stellwerts. Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Regelalgorithmus CONT_S konfigurieren

PI-Algorithmus

1. Geben Sie die "Proportionalverstärkung" für den P-Anteil ein.
2. Geben Sie die Integrationszeit für das Zeitverhalten des I-Anteils ein.
   Bei einer Integrationszeit von 0.0 ist der I-Anteil ausgeschaltet.

Stellwert CONT_S konfigurieren

Allgemein

Sie können CONT_S im Hand- oder im Automatikbetrieb einsetzen.

1. Um einen Handstellwert vorzugeben, aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
   An den Eingangsparametern LMNUP und LMNDN geben Sie einen Handstellwert vor.

Pulsegenerator

1. Geben Sie die Mindestimpulsdauer und die Mindestpausendauer ein.
   Die Werte müssen größer oder gleich der Zykluszeit am Eingangsparameter CYCLE sein. Dadurch wird die Schalthäufigkeit reduziert.
2. Geben Sie die Motorstellzeit ein.
   Der Wert muss größer oder gleich der Zykluszeit am Eingangsparameter CYCLE sein.

CONT_S in Betrieb nehmen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Geben Sie in den Feldern "P" und "I" einen neuen Proportionalbeiwert und eine neue Integrationszeit ein.
3. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".
4. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
5. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
6. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen Parametern und regelt auf den neuen Sollwert.
7. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
8. Wiederholen Sie die Schritte 2 bis 6, bis Sie mit dem Regelergebnis zufrieden sind.

3.4.1.3 TCONT_CP

Technologieobjekt TCONT_CP

Das Technologieobjekt TCONT_CP stellt einen kontinuierlichen Temperaturregler mit Pulsgenerator zur Verfügung. Es entspricht dem Instanz-Datenbaustein der Anweisung TCONT_CP. Die Arbeitsweise basiert auf dem PID-Regelalgorithmus des Abtastreglers. Es sind Hand- und Automatikbetrieb möglich.

Die Anweisung TCONT_CP berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der "Erstoptimierung". Die Parameter können über eine "Nachoptimierung" weiter optimiert werden. Alternativ können Sie die PID-Parameter manuell ermitteln.

S7-1500
Alle Parameter und Variablen des Technologieobjekts sind remanent und können beim Laden in Gerät nur geändert werden, wenn Sie TCONT_CP vollständig laden.

Siehe auch

Übersicht der Software-Regler (Seite 651)

Technologieobjekte hinzufügen (Seite 653)

Technologieobjekte konfigurieren (Seite 654)

Technologieobjekte in Gerät laden (Seite 656)

TCONT_CP konfigurieren

Regeldifferenz

Istwert Peripherie verwenden

Um den Eingangsparameter PV_PER zu verwenden gehen Sie folgendermaßen vor:

1. Wählen Sie aus der Liste "Quelle" den Eintrag "Peripherie".
2. Wählen Sie den "Sensortyp".
   Je nach Sensortyp wird der Istwert nach unterschiedlichen Formeln normiert.
   - Standard
     Thermoelemente; PT100/NI100
     PV = 0.1 × PV_PER × PV_FAC + PV_OFFS
   - Klima;
     PT100/NI100
     PV = 0.01 × PV_PER × PV_FAC + PV_OFFS
   - Strom/Spannung
     PV = 100/27648 × PV_PER × PV_FAC + PV_OFFS
3. Erfassen Sie Faktor und Offset für die Normierung des Istwerts Peripherie.

Internen Istwert verwenden

Um den Eingangsparameter PV_IN zu verwenden gehen Sie folgendermaßen vor:

1. Wählen Sie aus der Liste "Quelle" den Eintrag "Intern".

Regeldifferenz

Sie stellen unter folgender Voraussetzung eine Totzonenbreite ein:

- Das Istwert-Signal ist verrauscht.
- Die Reglerverstärkung ist hoch.
- Der D-Anteil ist aktiviert.

Der Rauschanteil des Istwerts verursacht in diesem Fall starke Schwankungen des Stellwerts. Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Regelalgorithmus

Allgemein

1. Geben Sie die "Abtastzeit PID-Algorithmus" ein.
   Die Regler-Abtastzeit sollte 10 % der ermittelten Integrationszeit des Reglers (TI) nicht übersteigen.
2. Wenn die Reglerstruktur einen P-Anteil enthält, geben Sie die "Proportionalverstärkung" ein.
   Eine negative Proportionalverstärkung invertiert den Regelsinn.

P-Anteil

Bei Änderungen des Sollwerts kann es durch den P-Anteil zu Überschwingern kommen. Über die Gewichtung P-Anteil können Sie wählen, wie stark der P-Anteil bei Sollwertänderungen wirken soll. Die Abschwächung des P-Anteils wird durch eine Kompensation am I-Anteil erreicht.

1. Um den P-Anteil bei Sollwertänderungen abzuschwächen, geben Sie die "Gewichtung P-Anteil" ein.
   - 1.0: P-Anteil bei Sollwertänderung voll wirksam
   - 0.0: P-Anteil bei Sollwertänderung nicht wirksam

I-Anteil

Bei Begrenzung des Stellwerts wird der I-Anteil angehalten. Bei einer Regeldifferenz, die den I-Anteil in Richtung innerer Stellbereich bewegt, wird der I-Anteil wieder freigegeben.

1. Wenn die Reglerstruktur einen I-Anteil enthält, geben Sie die "Integrationszeit" ein.
   Bei einer Integrationszeit von 0.0 ist der I-Anteil ausgeschaltet.
2. Um den I-Anteil mit einem Initialisierungswert zu belegen, aktivieren Sie das Optionskästchen "I-Anteil initialisieren" und geben Sie den "Initalisierungswert" ein.
   Bei Neustart oder COM_RST = TRUE wird der I-Anteil auf diesen Wert gesetzt.

D-Anteil

1. Wenn die Reglerstruktur einen D-Anteil enthält, geben Sie die Differenzierzeit (TD) und den Koeffizienten DT1 (D_F) ein.
   Bei zugeschaltetem D-Anteil sollte folgende Gleichung eingehalten werden:
   TD = 0.5 × CYCLE× D_F.
   Daraus wird die Verzögerungszeit berechnet nach der Formel:
   Verzögerungszeit = TD/D_F

PD-Regler mit Arbeitspunkt parametrieren

1. Geben Sie die Integrationszeit 0.0 ein.
2. Aktivieren Sie das Optionskästchen "I-Anteil initialisieren".
3. Geben Sie als Initialisierungswert den Arbeitspunkt ein.

P-Regler mit Arbeitspunkt parametrieren

1. Parametrieren Sie einen PD-Regler mit Arbeitspunkt.
2. Geben Sie die Differenzierzeit 0.0 ein.
   Der D-Anteil wird deaktiviert.

Regelzone

Die Regelzone begrenzt den Wertebereich der Regeldifferenz. Wenn die Regeldifferenz außerhalb dieses Wertebereichs liegt, werden die Stellwertgrenzen verwendet.

Beim Eintritt in die Regelzone führt der zugeschaltete D-Anteil zu einem sehr schnellen Reduzieren der Stellgröße. Daher ist die Regelzone nur bei eingeschaltetem D-Anteil sinnvoll. Ohne Regelzone würde im Wesentlichen nur der sich reduzierende P-Anteil die Stellgröße reduzieren. Die Regelzone führt zu einem schnelleren Einschwingen ohne Über-/Unterschwingen, wenn die ausgegebene minimale oder maximale Stellgröße weit von der für den neuen Arbeitspunkt stationär notwendigen Stellgröße entfernt ist.

1. Aktivieren Sie in der Gruppe "Regelzone" das Optionskästchen "Aktivieren".
2. Geben Sie im Eingabefeld "Breite" einen Wert ein, den der Istwert nach oben und unten vom Sollwert abweichen darf.

Stellwert kontinuierlicher Regler

Stellwertgrenzen

Der Stellwert wird nach oben und untern begrenzt, damit er nur gültige Werte annehmen kann. Sie können die Begrenzung nicht abschalten. Das Überschreiten der Grenzen wird durch die Ausgangsparameter QLMN_HLM und QLMN_LLM angezeigt.

1. Geben Sie einen Wert für die obere und untere Stellwertgrenze ein.

Normierung

Der Stellwert kann für die Ausgabe als Gleitpunkt- und Peripheriewert über einen Faktor und ein Offset nach folgender Formel normiert werden.

Normierter Stellwert = Stellwert × Faktor + Offset

Vorbelegt sind ein Faktor von 1.0 und ein Offset von 0.0.

1. Geben Sie einen Wert für Faktor und Offset ein.

Pulsegenerator

Für einen kontinuierlichen Regler muss der Pulsgenerator ausgeschaltet sein.

1. Deaktivieren Sie in der Gruppe "Pulsgenerator" das Optionskästchen "Aktivieren".

Stellwert Impulsregler

Pulsegenerator

Der analoge Stellwert (LmnN) kann durch Pulsweitenmodulation am Ausgangsparameter QPULSE als eine Impulsfolge ausgegeben werden.

Um den Pulsgenerator zu verwenden, gehen Sie folgendermaßen vor:

1. Aktivieren Sie in der Gruppe "Pulsgenerator" das Optionskästchen "Aktivieren".
2. Geben Sie die "Abtastzeit Pulsgenerator", die "Mindestimpuls-/Pausendauer" und die "Periodendauer" ein.

Die folgenden Grafiken verdeutlichen den Zusammenhang zwischen der "Abtastzeit Pulsgenerator" (CYCLE_P), der "Mindestimpuls-/Pausendauer" (P_B_TM) und der "Periodendauer" (PER_TM):

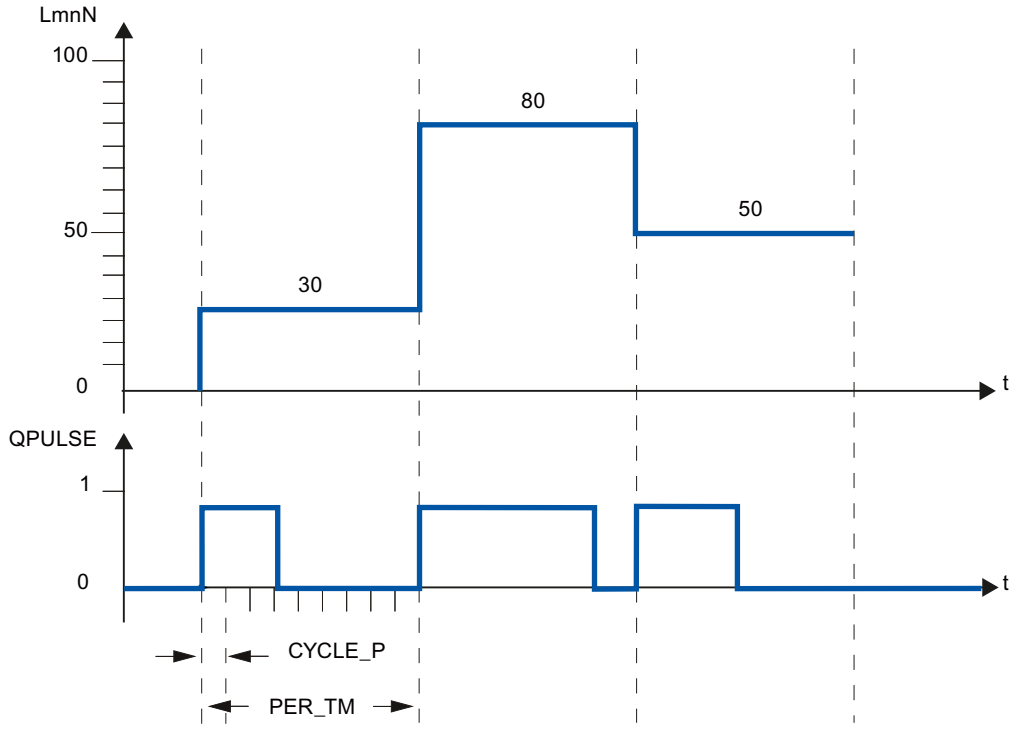

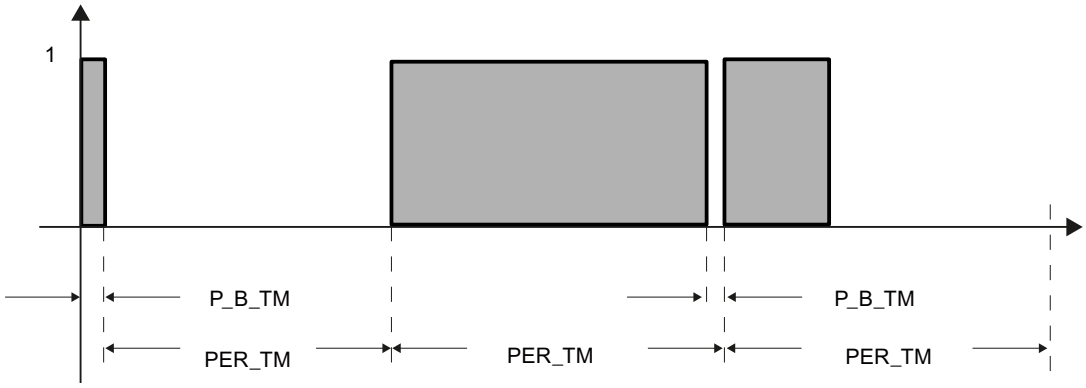

Abtastzeit Pulsgenerator

Die Abtastzeit Pulsgenerator muss mit dem Zeittakt des aufrufenden Weckalarm-OB übereinstimmen. Die Dauer des erzeugten Impulses beträgt stets ein ganzzahliges Vielfaches dieses Wertes. Für eine hinreichend genaue Stellwertauflösung sollte folgende Beziehung gelten:
CYCLE_P ≤ PER_TM/50

Mindestimpuls-/Pausendauer

Durch die Mindestimpuls-/Pausendauer werden kurze Ein- oder Ausschaltzeiten am Stellglied vermeiden. Ein Impuls kleiner als P_B_TM wird unterdrückt.

Empfohlen sind Werte P_B_TM ≤ 0.1 × PER_TM.

Periodendauer

Die Periodendauer sollte 20 % der ermittelten Integrationszeit des Reglers (TI) nicht übersteigen:
PER_TM ≤ TI/5

Beispiel für die Wirkung der Parameter CYCLE_P, CYCLE und PER_TM:

Periodendauer PER_TM = 10 s

Abtastzeit PID-Algorithmus CYCLE = 1 s

Abtastzeit Pulsgenerator CYCLE_P = 100 ms.

Alle Sekunde wird ein neuer Stellwert berechnet, alle 100 ms erfolgt der Vergleich des Stellwertes mit der bislang ausgegebenen Impulslänge bzw. Pausenlänge.

- Wenn ein Impuls ausgegeben wird, gibt es 2 Möglichkeiten:
  - Der berechnete Stellwert ist größer ist als die bisherige Impulslänge/PER_TM. Dann wird der Impuls verlängert.
  - Der berechnete Stellwert ist kleiner oder gleich der bisherigen Impulslänge/PER_TM. Dann wird kein Impulssignal mehr ausgegeben.
- Wenn kein Impuls ausgegeben wird, gibt es auch 2 Möglichkeiten:
  - Der Wert (100 % - berechneter Stellwert) ist größer als die bisherige Pausenlänge/ PER_TM. Dann wird die Pause verlängert.
  - Der Wert (100 % - berechneter Stellwert) ist kleiner oder gleich der bisherigen Pausenlänge/ PER_TM. Dann wird ein Impulssignal ausgegeben.

TCONT_CP in Betrieb nehmen

Optimierung TCONT_CP

Einsatzmöglichkeiten

Die Regleroptimierung ist für reine Heiz- oder reine Kühlprozesse vom Streckentyp I anwendbar. Aber auch für Strecken höherer Ordnung Streckentyp II oder III können Sie den Baustein einsetzen.

Die PI/-PID-Parameter werden automatisch ermittelt und eingestellt. Der Reglerentwurf ist auf optimales Störverhalten ausgelegt. Die daraus resultierenden "scharfen" Parameter führen bei Sollwertsprüngen zu Überschwingern von 10 % bis 40 % der Sprunghöhe.

Phasen der Regleroptimierung

Bei der Regleroptimierung werden einzelne Phasen durchlaufen, die Sie am Parameter PHASE ablesen können.

PHASE = 0

Es läuft keine Optimierung. TCONT_CP arbeitet im Automatik- oder Handbetrieb.

Während PHASE = 0 sorgen Sie dafür, dass die Regelstrecke die Vorrausetzungen für eine Optimierung erfüllt.

Am Ende der Optimierung wechselt TCONT_CP wieder in PHASE = 0.

PHASE = 1

TCONT_CP befindet sich in Optimierbereitschaft. PHASE = 1 darf nur gestartet werden, wenn die Vorrausetzungen für eine Optimierung erfüllt sind.

Während PHASE = 1 werden folgende Werte ermittelt:

- Istwert-Rauschen NOISE_PV
- Anfangssteigung PVDT0
- Mittelwert des Stellwerts
- Abtastzeit PID-Algorithmus CYCLE
- Abtastzeit Pulsgenerator CYCLE_P

PHASE = 2

In Phase 2 wird bei konstantem Stellwert der Wendepunkt des Istwerts gesucht. Das Verfahren verhindert, dass durch Rauschen von PV der Wendepunkt zu früh erkannt wird:

Beim Impulsregler wird PV über N Impulszyklen gemittelt und dann dem Reglerteil zur Verfügung gestellt. Im Reglerteil findet eine weitere Mittelung von PV statt: Zu Beginn ist diese Mittelung inaktiv, d. h. es wird nur immer über 1 Zyklus gemittelt. Solange das Rauschen ein bestimmtes Maß überschreitet, wird die Anzahl der Zyklen verdoppelt.

Die Periodendauer und die Amplitude des Rauschens werden ermittelt. Erst wenn der Gradient während der geschätzten Periodendauer immer kleiner als die maximale Steigung ist, wird die Wendepunktsuche abgebrochen und Phase 2 verlassen. TU und T_P_INF werden jedoch am tatsächlichen Wendepunkt berechnet.

Die Optimierung wird jedoch erst dann beendet, wenn auch die folgenden beiden Bedingungen erfüllt sind:

1. Der Istwert ist weiter als 2*NOISE_PV vom Wendepunkt entfernt.
2. Der Istwert hat den Wendepunkt um 20 % überschritten.

Hinweis

Bei Anregung über Sollwertsprung wird die Optimierung spätestens beendet, wenn der Istwert 75 % des Sollwertsprunges (SP_INT-PV0) durchlaufen hat (siehe unten).

PHASE = 3, 4, 5

Die Phasen 3, 4 und 5 dauern jeweils 1 Zyklus.

In Phase 3 werden die vor der Optimierung gültigen PI-/PID-Parameter gespeichert und die Prozess-Parameter berechnet.

In Phase 4 werden die neuen PI-/PID-Parameter berechnet.

In Phase 5 wird der neue Stellwert berechnet und auf die Regelstrecke gegeben.

PHASE = 7

Der Streckentyp wird in Phase 7 überprüft, da TCONT_CP nach der Optimierung immer in den Automatikbetrieb wechselt. Der Automatikbetrieb startet mit LMN = LMN0 + 0.75*TUN_DLMN als Stellwert. Die Überprüfung des Streckentyps findet **im Automatikbetrieb** mit den gerade neu berechneten Reglerparametern statt und endet spätestens 0,35*TA (Ausgleichszeit) nach dem Wendepunkt. Weicht die Prozessordnung stärker vom geschätzten Wert ab, werden die Reglerparameter neu berechnet und STATUS_D um 1 hochgezählt, ansonsten bleiben die Reglerparameter unverändert.

Dann ist der Optimierbetrieb beendet und TCONT_CP befindet sich wieder in PHASE = 0. Am Parameter STATUS_H erkennen Sie, ob die Optimierung erfolgreich beendet wurde.

Vorzeitiger Abbruch der Optimierung

In Phase 1, 2 oder 3 können Sie durch Rücksetzen von TUN_ON = FALSE die Optimierung abbrechen ohne dass neue Parameter berechnet werden. Der Regler startet im Automatikbetrieb mit LMN = LMN0 + TUN_DLMN. War der Regler vor der Optimierung im Handbetrieb, wird der alte Handstellwert ausgegeben.

Wird in Phase 4, 5 oder 7 mit TUN_ON = FALSE die Optimierung abgebrochen, bleiben die bis dahin ermittelten Regelparameter erhalten.

Voraussetzungen für eine Optimierung

Einschwingverhalten

Der Prozess muss ein stabiles, verzugsbehaftetes, asymptotisches Einschwingverhalten zeigen.

Nach einem Sprung der Stellgröße muss der Istwert in einen stationären Zustand übergehen. Ausgeschlossen sind damit Prozesse, die schon ohne Regelung ein oszillatorisches Verhalten zeigen, sowie Regelstrecken ohne Ausgleich (Integrator in der Regelstrecke).

⚠ WARNUNG
Es kann Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten. Während einer Optimierung ist der Parameter MAN_ON nicht wirksam. Hierdurch können Stellwert oder Istwert ungewünschte –auch extreme- Werte annehmen. Der Stellwert wird durch die Optimierung vorgegeben. Um die Optimierung abzubrechen müssen Sie zuerst TUN_ON = FALSE setzen. Damit ist auch MAN_ON wieder wirksam.

Sicherstellen eines quasi-stationären Anfangszustandes (Phase 0)

Bei niederfrequenten Schwingungen des Istwerts, z. B. auf Grund falscher Reglerparameter, ist der Regler vor dem Start der Optimierung auf Handbetrieb zu stellen und das Abklingen der Schwingung abzuwarten. Alternativ könnte auch auf einen "sanft" eingestellten PI-Regler (kleine Kreisverstärkung, große Integrationszeit) umgeschaltet werden.

Nun müssen Sie warten, bis der stationäre Zustand erreicht ist, d. h. bis Istwert und Stellwert eingeschwungen sind. Zulässig ist auch ein asymptotisches Einschwingen bzw. langsames Driften des Istwertes (quasi-stationärer Zustand, siehe nachfolgendes Bild). Die Stellgröße muss konstant sein bzw. um einen konstanten Mittelwert schwanken.

Hinweis

Vermeiden Sie, die Stellgröße kurz vor Start der Optimierung zu verändern. Eine Änderung der Stellgröße kann auch unbeabsichtigt durch das Herstellen der Versuchsbedingungen erfolgen (z. B. Schließen einer Ofentür)! Sollte dies doch geschehen sein, müssen Sie mindestens warten, bis der Istwert wieder asymptotisch in einen stationären Zustand einschwingt. Bessere Reglerparameter erzielen Sie jedoch, wenn Sie warten bis der Einschwingvorgang vollständig abgeklungen ist.

Im folgenden Bild ist das Einschwingen in den stationären Zustand dargestellt:

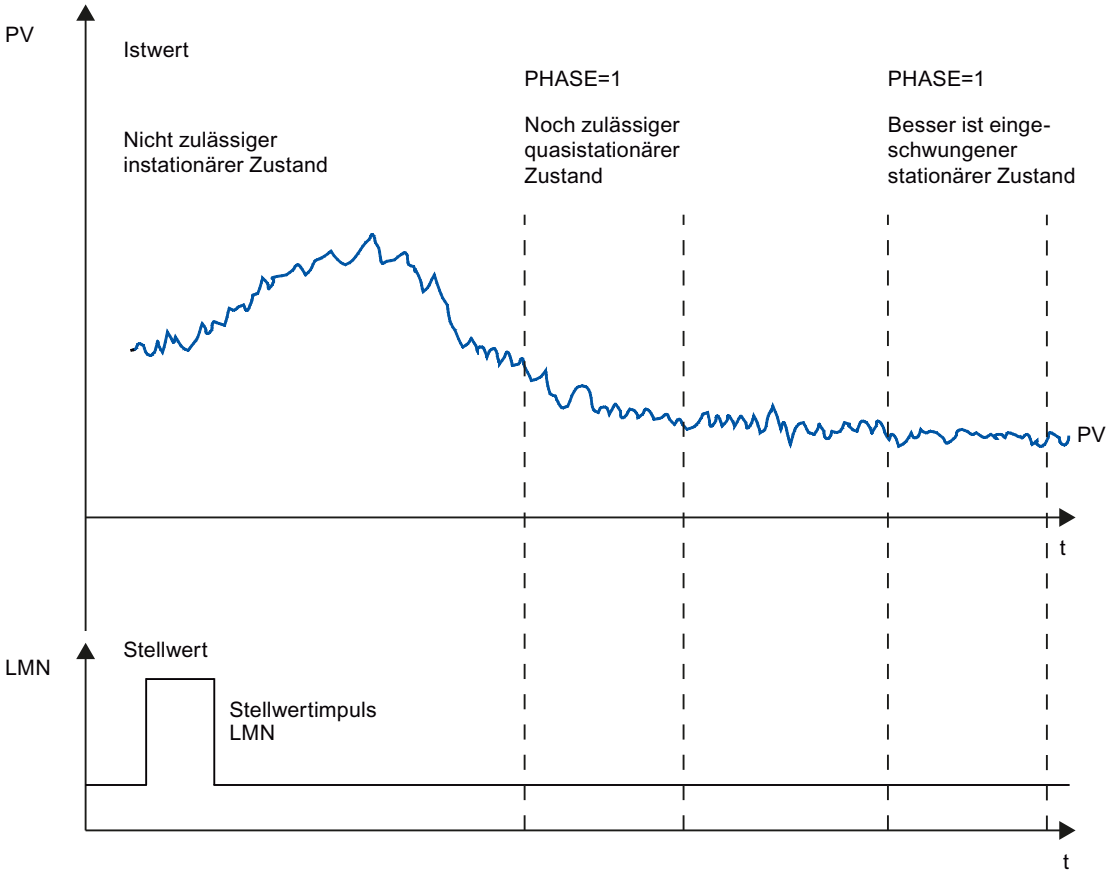

Linearität und Arbeitsbereich

Der Prozess muss ein lineares Verhalten über dem Arbeitsbereich zeigen. Ein nichtlineares Verhalten tritt z. B. bei einem Wechsel eines Aggregatszustandes auf. Die Optimierung muss in einem linearen Teil des Arbeitsbereiches stattfinden.

Das heißt, dass sowohl für die Optimierung als auch für den normalen Regelungsbetrieb nichtlineare Effekte innerhalb dieses Arbeitsbereichs vernachlässigbar klein sein müssen. Allerdings ist es möglich, beim Wechsel des Arbeitspunktes den Prozess neu zu optimieren, wenn die Optimierung wieder in einer kleinen Umgebung des neuen Arbeitspunktes durchgeführt wird und während der Optimierung die Nichtlinearität nicht durchfahren wird.

Wenn bestimmte statische Nichtlinearitäten (z. B. Ventilkennlinien) bekannt sind, ist es in jedem Fall sinnvoll, sie vorab mit einem Polygonzug zu kompensieren, um das Prozessverhalten zu linearisieren.

Störeinflüsse bei Temperaturprozessen

Störeinflüsse wie die Wärmeübertragung an benachbarten Zonen dürfen den Gesamttemperaturprozess nicht zu stark beeinflussen. Z. B. müssen bei der Optimierung von Zonen eines Extruders alle Zonen gleichzeitig aufgeheizt werden.

Möglichkeiten der Optimierung

Es gibt die folgenden Möglichkeiten zur Optimierung:

- Erstoptimierung
- Nachoptimierung
- Manuelle Nachoptimierung im Regelbetrieb

Erstoptimierung

Während dieser Optimierung wird der Arbeitspunkt aus dem kalten Zustand durch einen Sollwertsprung angefahren.

Mit TUN_ON = TRUE stellen Sie die Optimierbereitschaft her. Der Regler wechselt von PHASE = 0 nach PHASE = 1.

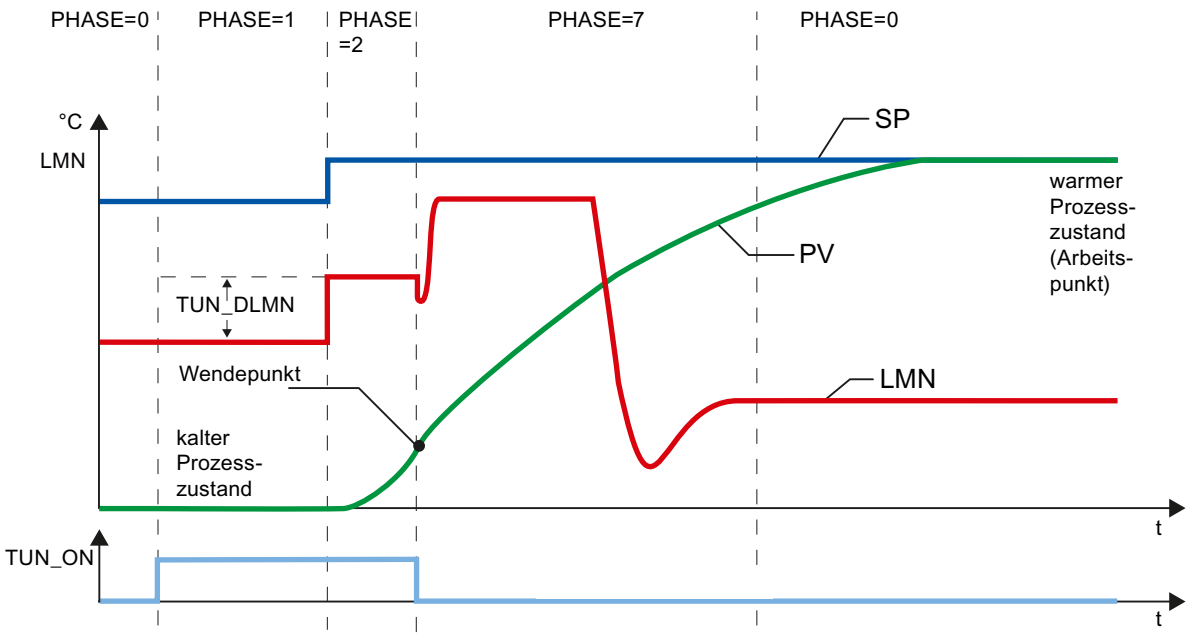

Die Optimierungs-Stellgröße (LMN0 + TUN_DLMN) wird durch Sollwertänderung aufgeschaltet (Übergang Phase 1 -> 2). Der Sollwert wird erst bei Erreichen des Wendepunktes wirksam (erst dann wird in den Automatikbetrieb geschaltet).

Sie legen das Delta der Stellanregung (TUN_DLMN) entsprechend der zulässigen Istwertänderung in Eigenverantwortung fest. Das Vorzeichen von TUN_DLMN muss entsprechend der beabsichtigten Istwertänderung sein (Regelsinn berücksichtigen).

Der Sollwertsprung und TUN_DLMN müssen aufeinander abgestimmt sein. Ein zu großes TUN_DLMN birgt die Gefahr, dass der Wendepunkt nicht innerhalb von 75 % des Sollwertsprunges gefunden wird.

TUN_DLMN muss aber so groß sein, dass der Istwert mindestens 22 % des Sollwertsprunges erreicht. Sonst bleibt das Verfahren im Optimierbetrieb (Phase 2).

Abhilfe: Verringern des Sollwerts während der Wendepunktsuche.

Hinweis

Bei stark verzögerten Prozessen ist es ratsam, den Zielsollwert bei einer Optimierung etwas tiefer zu legen als den gewünschten Arbeitspunkt und die Statusbits sowie PV genau zu beobachten (Gefahr des Überschwingens).

Optimierung nur im linearen Bereich:

Bestimmte Regelstrecken (z. B. Zink- oder Magnesium-Schmelztiegel) durchlaufen kurz vor dem Arbeitspunkt einen nichtlinearen Bereich (Änderung des Aggregatzustands).

Durch geschickte Wahl des Sollwertsprunges kann die Optimierung auf den linearen Bereich begrenzt werden. Hat der Istwert 75 % des Sollwertsprungs (SP_INT-PV0) durchlaufen, wird die Optimierung beendet.

Parallel dazu sollte TUN_DLMN so weit reduziert werden, damit der Wendepunkt mit Sicherheit vor Erreichen von 75 % des Sollwertsprunges gefunden wird.

Nachoptimierung

Während dieser Optimierung wird der Prozess bei konstantem Sollwert durch einen Stellwertsprung angeregt.

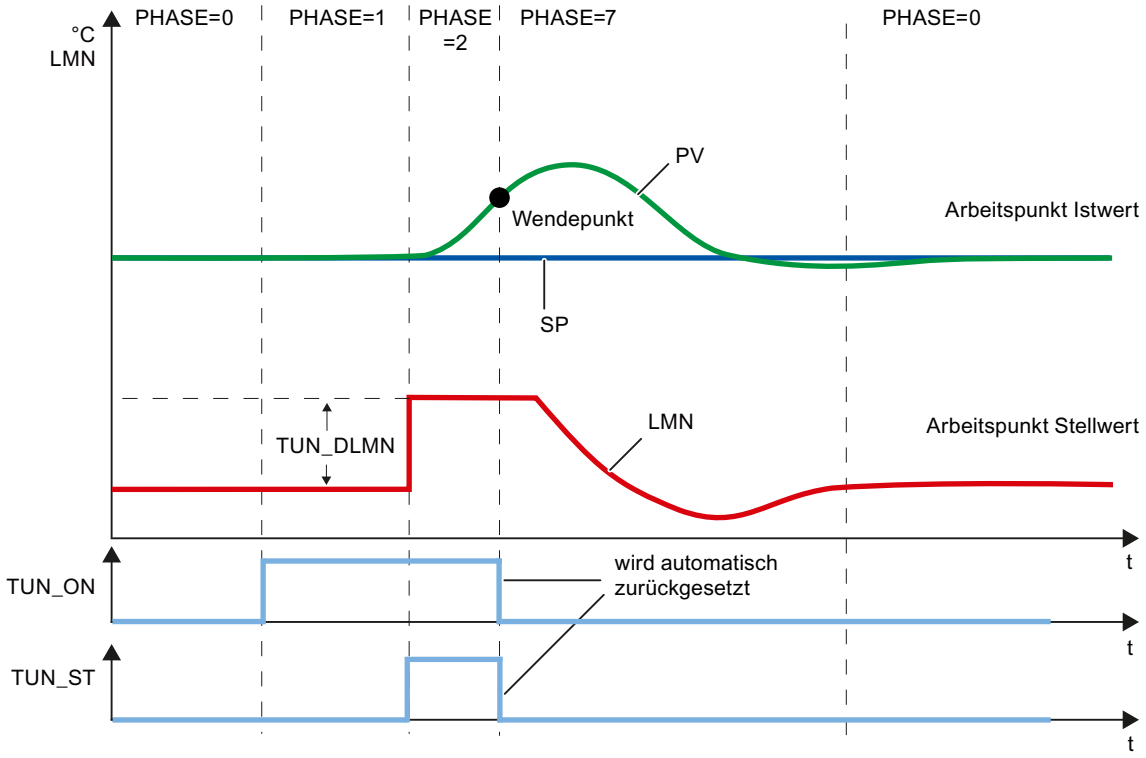

Die Optimierungs-Stellgröße (LMN0 + TUN_DLMN) wird durch Setzen des Start-Bits TUN_ST aufgeschaltet (Übergang Phase 1 -> 2). Wenn Sie den Sollwert verändern, wird der neue Sollwert erst bei Erreichen des Wendepunktes wirksam (erst dann wird in den Automatikbetrieb geschaltet).

Sie legen das Delta der Stellanregung (TUN_DLMN) entsprechend der zulässigen Istwertänderung in Eigenverantwortung fest. Das Vorzeichen von TUN_DLMN muss entsprechend der beabsichtigten Istwertänderung sein (Regelsinn berücksichtigen).

ACHTUNG
Bei Anregung über TUN_ST gibt es kein Sicherheitsabschalten bei 75 %. Die Optimierung wird bei Erreichen des Wendepunktes beendet. Bei rauschbehafteten Strecken kann jedoch der Wendepunkt deutlich überschritten werden.

Manuelle Nachoptimierung im Regelbetrieb

Um ein überschwingfreies Sollwertverhalten zu erreichen, können Sie folgende Maßnahmen ergreifen:

- Regelzone anpassen
- Führungsverhalten optimieren

- Regelparameter dämpfen
- Regelparameter ändern

Ergebnis der Optimierung

Die linke Ziffer von STATUS_H zeigt den Optimierungsstatus

STATUS_H	Ergebnis
0	Default bzw. keine oder noch keine neuen Reglerparameter gefunden.
10000	Geeignete Regel-Parameter gefunden
2xxxx	Regel-Parameter über Schätzwerte gefunden; Überprüfen Sie das Regelverhalten bzw. schauen Sie nach der Diagnosemeldung STATUS_H und wiederholen Sie die Regleroptimierung.
3xxxx	Bedien-Fehler aufgetreten; Schauen Sie nach der Diagnosemeldung STATUS_H und wiederholen Sie die Regleroptimierung.

In Phase 1 wurden schon die Abtastzeiten CYCLE und CYCLE_P geprüft.

Die folgenden Regel-Parameter werden an TCONT_CP aktualisiert:

- P (Proportionalbeiwert GAIN)
- I (Integrationszeit TI)
- D (Differenzierzeit TD)
- Gewichtung des P-Anteils PFAC_SP
- Koeffizient DT1 (D_F)
- Regelzone ein/aus CONZ_ON
- Regelzonenbreite CON_ZONE

Die Regelzone wird nur bei passendem Streckentyp (Streckentyp I und II) und PID-Regler aktiviert (CONZ_ON = TRUE).

Abhängig von PID_ON wird entweder mit PI- oder PID-Regler geregelt. Die alten Reglerparameter werden gesichert und können über UNDO_PAR wieder aktiviert werden. Ferner werden ein PI- und ein PID-Parametersatz in den Strukturen PI_CON und PID_CON gespeichert. Über LOAD_PID und ein entsprechendes Setzen von PID_ON kann auch später noch zwischen den optimierten PI- oder PID-Parametern gewechselt werden.

Parallele Optimierung von Reglerkanälen

Nachbar-Zonen (starke Wärmekopplung)

Regeln zwei oder mehrere Regler die Temperatur z. B. auf einer Platte (d. h. zwei Heizungen und zwei gemessene Istwerte mit starker Wärmekopplung), gehen Sie folgendermaßen vor:

1. Verodern Sie die beiden Ausgänge QTUN_RUN.
2. Verschalten Sie die beiden TUN_KEEP Eingänge jeweils mit dem Ausgang des Oder-Gliedes.
3. Starten Sie beide Regler durch gleichzeitige Vorgabe eines Sollwertsprunges oder gleichzeitiges Setzen von TUN_ST.

In folgendem Bild ist die parallele Optimierung von Reglerkanälen dargestellt:

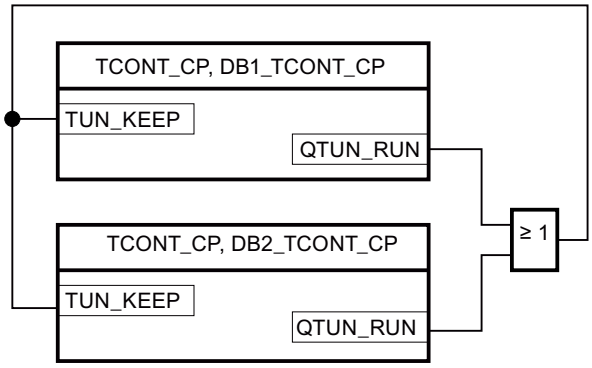

Vorteil:

Beide Regler geben solange LMN0 + TUN_DLMN aus, bis beide Regler Phase 2 verlassen haben. Damit wird vermieden, dass der Regler, der die Optimierung früher beendet, durch die Änderung seiner Stellgröße das Optimierungsergebnis des anderen Reglers verfälscht.

ACHTUNG
Das Erreichen von 75 % des Sollwertsprunges führt zu einem Verlassen der Phase 2 und damit zu einem Zurücksetzen des Ausgangs QTUN_RUN. Der Automatikbetrieb wird jedoch erst begonnen, wenn auch TUN_KEEP 0 wird.

Nachbar-Zonen (schwache Wärmekopplung)

Generell gilt, dass man so optimieren sollte, wie man später regelt. Werden im Produktionsbetrieb die Zonen gemeinsam parallel verfahren, so dass die Temperaturunterschiede zwischen den Zonen gleich bleiben, sollte man auch bei der Optimierung das Temperaturniveau der Nachbarzonen entsprechend mit anheben.

Die Temperaturunterschiede zu Beginn des Versuchs spielen keine Rolle, da sie durch eine entsprechende Anfangsheizung ausgeglichen werden (→ Anfangssteigung = 0).

Fehlerbilder und Abhilfemaßnahmen

Abfangen von Bedienfehlern

Bedienfehler	STATUS und Maßnahme	Kommentar
Gleichzeitiges Setzen von TUN_ON und Sollwertsprung bzw. TUN_ST	Übergang in Phase 1, jedoch kein Optimierungsstart. • SP_INT = SP_{alt} bzw. • TUN_ST = FALSE	Die Sollwertänderung wird zurückgenommen. Dadurch wird verhindert, dass der Regler auf den neuen Sollwert einregelt und unnötig den stationären Arbeitspunkt verlässt.
Effektives TUN_DLMN < 5 % (Ende von Phase 1)	STATUS_H = 30002 • Übergang in Phase 0 • TUN_ON = FALSE • SP = SP_{alt}	Abbruch der Optimierung. Die Sollwertänderung wird zurückgenommen. Dadurch wird verhindert, dass der Regler auf den neuen Sollwert einregelt und unnötig den stationären Arbeitspunkt verlässt.

Wendepunkt nicht erreicht (nur bei Anregung durch Sollwertsprung)

Die Optimierung wird spätestens beendet, wenn der Istwert 75 % des Sollwertsprunges (SP-INT-PV0) durchlaufen hat. Dies wird durch "Wendepunkt nicht erreicht" in STATUS_H (2xx2x) signalisiert.

Dabei gilt immer der momentan eingestellte Sollwert. Durch ein Reduzieren des Sollwertes kann also im Nachhinein ein früheres Beenden der Optimierung herbeigeführt werden.

Bei typischen Temperaturstrecken reicht der Optimierungsabbruch bei 75 % des Sollwertsprunges in der Regel aus, um ein Überschwingen zu verhindern. Vor allem bei stärker verzögerten Strecken (TU/TA > 0.1, Streckentyp III) ist jedoch Vorsicht geboten. Bei zu starker Stellanregung im Vergleich zum Sollwertsprung kann der Istwert stark überschwingen (bis zu Faktor 3).

Ist bei Strecken höherer Ordnung nach Erreichen von 75 % des Sollwertsprunges der Wendepunkt noch in weiter Ferne, gibt es ein deutliches Überschwingen. Außerdem sind die Reglerparameter zu scharf. Schwächen Sie dann die Reglerparameter ab oder wiederholen Sie den Versuch.

Im folgenden Bild ist das Überschwingen des Istwertes bei zu starker Anregung (Streckentyp III) dargestellt:

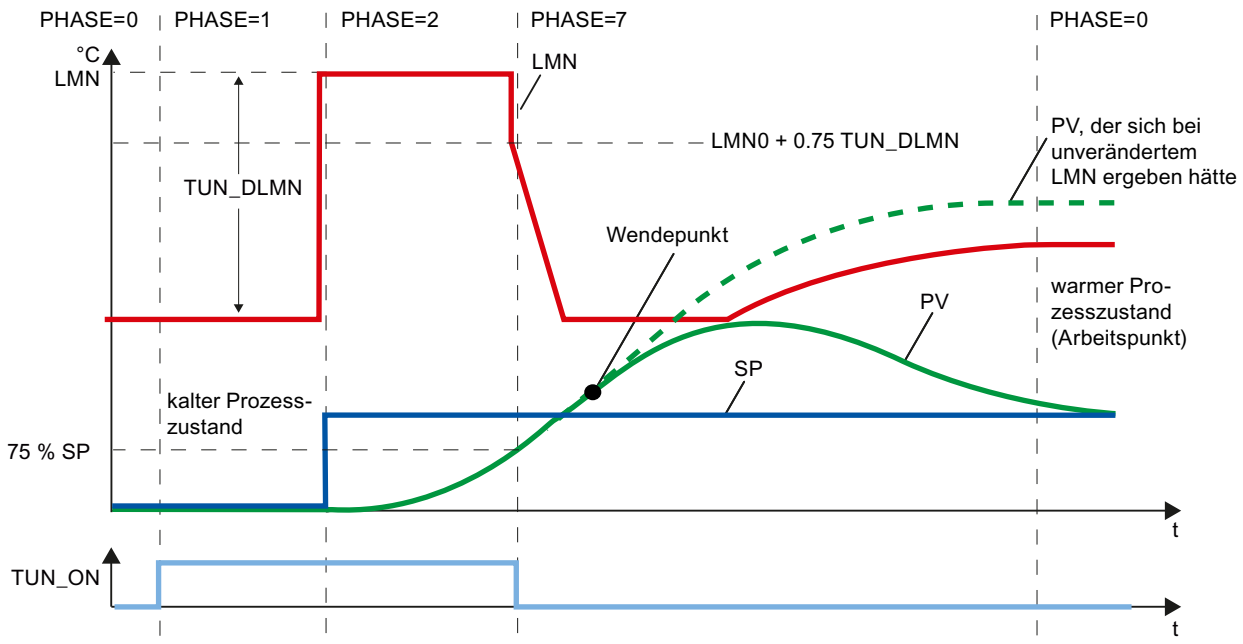

Bei typischen Temperaturstrecken ist ein Abbruch kurz vor Erreichen des Wendepunkts unkritisch bezüglich der Reglerparameter.

Falls Sie den Versuch wiederholen, reduzieren Sie TUN_DLMN oder erhöhen Sie den Sollwertsprung.

Prinzip: Der Optimierungs-Stellwert muss zum Sollwertsprung passen.

Schätzfehler bei Verzugszeit oder Ordnung

Die Verzugszeit (STATUS_H = 2x1xx oder 2x3xx) oder die Ordnung (STATUS_H = 21xxx oder 22xxx) konnten nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann.

Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Hinweis

Der Sonderfall einer reinen PT1-Strecke wird auch durch STATUS_H = 2x1xx (TU <= 3*CYCLE) signalisiert. Ein Wiederholen des Versuchs ist dann nicht notwendig. Schwächen Sie die Reglerparameter ab, falls die Regelung schwingt.

Qualität der Messsignale (Mess-Rauschen, niederfrequente Störungen)

Das Optimierungsergebnis kann durch Messrauschen oder durch niederfrequente Störungen beeinträchtigt worden sein. Beachten Sie hierbei Folgendes:

- Wählen Sie bei Messrauschen die Abtastfrequenz eher höher als niedriger. Dabei sollte innerhalb einer Rauschperiode der Istwert mindestens zwei Mal abgetastet werden. Bei Impulsbetrieb ist dabei die integrierte Mittelwertfilterung hilfreich. Sie setzt allerdings voraus, dass der Istwert PV im schnellen Impulszyklus der Anweisung übergeben wird. Das Rauschmaß sollte 5 % der Nutzsignaländerung nicht übersteigen.
- Hochfrequente Störungen können nicht mehr durch TCONT_CP weggefiltert werden. Diese sollten bereits im Messwertaufnehmer gefiltert worden sein, um den sogenannten Aliasing-Effekt zu verhindern.
  In folgendem Bild ist der Aliasing-Effekt bei zu großer Abtastzeit dargestellt:

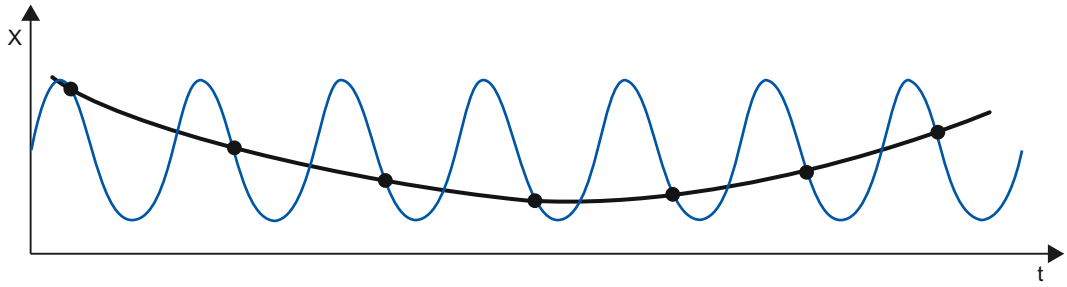

- Bei niederfrequenten Störungen ist es relativ leicht, eine genügend hohe Abtastrate sicherzustellen. Andererseits muss dann der TCONT_CP durch ein großes Intervall der Mittelwertfilterung sich ein gleichmäßiges Messsignal erzeugen. Eine Mittelwertfilterung muss sich dabei über mindestens zwei Rauschperioden erstrecken. Intern im Baustein entstehen somit schnell größere Abtastzeiten, so dass die Genauigkeit der Optimierung beeinträchtigt wird. Eine genügend große Genauigkeit ist bei mindestens 40 Rauschperioden bis zum Wendepunkt sichergestellt.
  Mögliche Maßnahme bei Wiederholung des Versuchs:
  Erhöhen von TUN_DLMN.

Überschwingen

In folgenden Situationen kann Überschwingen auftreten:

Situation	Ursache	Abhilfe
Optimierungs-ende	• Anregung durch zu starke Stellwertänderung im Vergleich zum Sollwertsprung (siehe oben). • PI-Regler durch PID_ON = FALSE aktiviert.	• Sollwertsprung erhöhen oder Stellwertsprung verringern • Wenn der Prozess einen PID-Regler zulässt, starten Sie die Optimierung mit PID_ON = TRUE.
Optimierung in Phase 7	Es wurden zunächst sanftere Reglerparameter ermittelt (Streckentyp III), die zu einem Überschwingen in Phase 7 führen können.	-
Regelbetrieb	PI-Regler und mit PFAC_SP = 1.0 für Streckentyp I.	Wenn der Prozess einen PID-Regler zulässt, starten Sie die Optimierung mit PID_ON = TRUE.

Erstoptimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um bei der ersten Inbetriebnahme die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Erstoptimierung".
   TCONT_CP ist in Optimierbereitschaft.
3. Geben Sie im Feld "Stellwertsprung" ein, um welchen Wert der Stellwert erhöht werden soll.
4. Geben Sie im Feld "Sollwert" einen Sollwert ein. Erst durch einen anderen Sollwert wird der Stellwertsprung wirksam.
5. Klicken Sie auf das Symbol ▶ "Optimierung starten".
   Die Erstoptimierung startet. Der Status der Optimierung wird angezeigt.

Nachoptimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um am Arbeitspunkt die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Nachoptimierung".
   TCONT_CP ist in Optimierbereitschaft.
3. Geben Sie im Feld "Stellwertsprung" ein, um welchen Wert der Stellwert erhöht werden soll.
4. Klicken Sie auf das Symbol ▶ "Optimierung starten".
   Die Nachoptimierung startet. Der Status der Optimierung wird angezeigt.

Abbruch der Erst- oder Nachoptimierung

Um eine Erst- oder Nachoptimierung abzubrechen, klicken Sie auf das Symbol ■ "Optimierung stoppen".

Wenn die PID-Parameter noch nicht berechnet und gespeichert wurden, startet TCONT_CP im Automatikbetrieb mit LMN = LMN0 + TUN_DLMN. War der Regler vor der Optimierung im Handbetrieb, wird der alte Handstellwert ausgegeben.

Wenn die berechneten PID-Parameter bereits gespeichert sind, startet TCONT_CP im Automatikbetrieb und arbeitet mit den bis dahin ermittelten PID-Parametern.

Manuelle Nachoptimierung im Regelbetrieb

Um ein überschwingfreies Sollwertverhalten zu erreichen, können Sie nachfolgend beschriebene Maßnahmen ergreifen:

Regelzone anpassen

Bei der Optimierung wird eine Regelzone CON_ZONE von TCONT_CP ermittelt und bei passendem Streckentyp (Streckentyp I und II) und PID-Regler aktiviert (CONZ_ON = TRUE). Sie können im Regelbetrieb die Regelzone ändern oder ganz abschalten (mit CONZ_ON = FALSE).

Hinweis

Aktivieren Sie die Regelzone bei Strecken höherer Ordnung (Streckentyp III), bringt dies in der Regel keinen Vorteil, da die Regelzone dann größer ist als der mit 100 % Stellgröße erzielbare Regelbereich. Auch ein Aktivieren der Regelzone für PI-Regler bringt keinen Vorteil.

Stellen Sie vor dem händischen Einschalten der Regelzone sicher, dass die Regelzonenbreite nicht zu klein eingestellt ist. Bei zu klein eingestellter Regelzonenbreite treten Schwingungen im Verlauf der Stellgröße und des Istwertverlaufs auf.

Führungsverhalten mit PFAC_SP kontinuierlich abschwächen

Das Führungsverhalten können Sie mit dem Parameter PFAC_SP abschwächen. Dieser Parameter legt fest, zu welchem Anteil der P-Anteil bei Sollwertsprüngen wirksam wird.

PFAC_SP wird unabhängig vom Streckentyp durch die Optimierung auf 0.8 voreingestellt, Sie können den Wert anschließend verändern. Um das Überschwingen bei Sollwertsprüngen (bei sonst korrekten Reglerparametern) auf etwa 2 % zu begrenzen, sind folgende Werte für PFAC_SP ausreichend:

	Streckentyp I	Streckentyp II	Streckentyp III
	Typische Temperaturstrecke	Übergangsbereich	Temperaturstrecke höherer Ordnung
PI	0.8	0.82	0.8
PID	0.6	0.75	0.96

Passen Sie den voreingestellten Faktor (0.8) insbesondere in folgenden Fällen an:

- Streckentyp I mit PID (0.8 -> 0.6): Sollwertsprünge innerhalb der Regelzone führen mit PFAC_SP = 0.8 noch zu ca. 18 % Überschwingen.
- Streckentyp III mit PID (0.8 -> 0.96): Sollwertsprünge mit PFAC_SP = 0.8 werden zu stark gedämpft. Man verschenkt deutlich Einregelzeit.

Regelparameter dämpfen

Treten im geschlossenen Regelkreis Schwingungen auf oder sind Überschwinger nach Sollwertsprüngen vorhanden, können Sie die Reglerverstärkung GAIN zurückdrehen (z. B. auf 80 % des ursprünglichen Werts) und die Integrationszeit TI vergrößern (z. B. auf 150 % des ursprünglichen Werts). Wird der analoge Stellwert des kontinuierlichen Reglers mit einem Impulsformer in binäre Stellsignale gewandelt, können durch Quantisierungseffekte kleine Dauerschwingungen auftreten. Sie können diese durch Vergrößern der Reglertotzone DEADB_W eliminieren.

Regelparameter ändern

Wenn Sie Ihre Reglerparameter ändern wollen, gehen Sie folgendermaßen vor:

1. Speichern Sie die aktuellen Parameter mit SAVE_PAR.
2. Ändern Sie die Parameter.
3. Testen Sie das Regelverhalten.

Falls die neuen Parameter schlechter sein sollten als die alten, stellen Sie mit UNDO_PAR die alten Parameter wieder ein.

Manuelle Nachoptimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Manuell".
3. Geben Sie die neuen PID-Parameter ein.
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".
5. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
6. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .

7. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.
8. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
9. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

3.4.1.4 TCONT_S

Technologieobjekt TCONT_S

Das Technologieobjekt TCONT_S stellt einen Schrittregler für Stellglieder mit integrierendem Verhalten zur Verfügung und dient zum Regeln von technischen Temperatur-Prozessen mit binären Stellwertausgangssignalen. Das Technologieobjekt entspricht dem Instanz-Datenbaustein der Anweisung TCONT_S. Die Arbeitsweise basiert auf dem PI-Regelalgorithmus des Abtastreglers. Der Schrittregler arbeitet ohne Stellungsrückmeldung. Es sind Hand- und Automatikbetrieb möglich.

S7-1500
Alle Parameter und Variablen des Technologieobjekts sind remanent und können beim Laden in Gerät nur geändert werden, wenn Sie TCONT_S vollständig laden.

Siehe auch

Übersicht der Software-Regler (Seite 651)

Technologieobjekte hinzufügen (Seite 653)

Technologieobjekte konfigurieren (Seite 654)

Technologieobjekte in Gerät laden (Seite 656)

Regeldifferenz TCONT_S konfigurieren

Istwert Peripherie verwenden

Um den Eingangsparameter PV_PER zu verwenden gehen Sie folgendermaßen vor:

1. Wählen Sie aus der Liste "Quelle" den Eintrag "Peripherie".
2. Wählen Sie den "Sensortyp".
   Je nach Sensortyp wird der Istwert nach unterschiedlichen Formeln normiert.
   - Standard
     Thermoelemente; PT100/NI100
     PV = 0.1 × PV_PER × PV_FAC + PV_OFFS
   - Klima;
     PT100/NI100
     PV = 0.01 × PV_PER × PV_FAC + PV_OFFS
   - Strom/Spannung
     PV = 100/27648 × PV_PER × PV_FAC + PV_OFFS
3. Erfassen Sie Faktor und Offset für die Normierung des Istwerts Peripherie.

Internen Istwert verwenden

Um den Eingangsparameter PV_IN zu verwenden gehen Sie folgendermaßen vor:

1. Wählen Sie aus der Liste "Quelle" den Eintrag "Intern".

Regeldifferenz

Sie stellen unter folgender Voraussetzung eine Totzonenbreite ein:

- Das Istwert-Signal ist verrauscht.
- Die Reglerverstärkung ist hoch.
- Der D-Anteil ist aktiviert.

Der Rauschanteil des Istwerts verursacht in diesem Fall starke Schwankungen des Stellwerts. Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

Regelalgorithmus TCONT_S konfigurieren

Allgemein

1. Geben Sie die "Abtastzeit PID-Algorithmus" ein.
   Die Regler-Abtastzeit sollte 10 % der ermittelten Integrationszeit des Reglers (TI) nicht übersteigen.
2. Wenn die Reglerstruktur einen P-Anteil enthält, geben Sie die "Proportionalverstärkung" ein.
   Eine negative Proportionalverstärkung invertiert den Regelsinn.

P-Anteil

Bei Änderungen des Sollwerts kann es durch den P-Anteil zu Überschwingern kommen. Über die Gewichtung P-Anteil können Sie wählen, wie stark der P-Anteil bei Sollwertänderungen wirken soll. Die Abschwächung des P-Anteils wird durch eine Kompensation am I-Anteil erreicht.

1. Um den P-Anteil bei Sollwertänderungen abzuschwächen, geben Sie die "Gewichtung P-Anteil" ein.
   - 1.0: P-Anteil bei Sollwertänderung voll wirksam
   - 0.0: P-Anteil bei Sollwertänderung nicht wirksam

I-Anteil

1. Wenn die Reglerstruktur einen I-Anteil enthält, geben Sie die "Integrationszeit" ein.
   Bei einer Integrationszeit von 0.0 ist der I-Anteil ausgeschaltet.

Stellwert TCONT_S konfigurieren

Pulsegenerator

1. Geben Sie die Mindestimpulsdauer und die Mindestpausendauer ein.
   Die Werte müssen größer oder gleich der Zykluszeit am Eingangsparameter CYCLE sein. Dadurch wird die Schalthäufigkeit reduziert.
2. Geben Sie die Motorstellzeit ein.
   Der Wert muss größer oder gleich der Zykluszeit am Eingangsparameter CYCLE sein.

TCONT_S in Betrieb nehmen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Geben Sie in den Feldern "P", "I" und Gewichtung P-Anteil die neuen PI-Parameter ein.
3. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".
4. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
5. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
6. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen Parametern und regelt auf den neuen Sollwert.
7. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
8. Wiederholen Sie die Schritte 2 bis 6, bis Sie mit dem Regelergebnis zufrieden sind.

3.5 PID-Regelung (S7-300, S7-400)

3.5.1 Standard PID Control einsetzen

3.5.1.1 PID_CP einsetzen

Technologieobjekt PID_CP

Das Technologieobjekt PID_CP stellt einen kontinuierlichen PID-Regler für Automatik und Handbetrieb zur Verfügung. Es entspricht dem Instanz-Datenbaustein der Anweisung PID_CP. Sie können die folgenden Regler konfigurieren:

- Festwertregelung mit kontinuierlichem P–, PI–, PD–, PID–Regler
- Festwertregelung mit Störgrößenaufschaltung
- Kaskadenregelung
- Zweischleifige Verhältnisregelung
- Mischungsregelung

Sie konfigurieren die folgenden Funktionen des Technologieobjekts PID_CP:

- Normierung des Istwertsignals
- Unterdrückung von höherfrequenten Schwingungen im Istwertzweig durch Glättung (Verzögerung) des Istwertsignals
- Linearisierung von quadratischen Funktionen des Istwerts (Durchflussregelung mit Differenzdruck–Gebern)
- Aufruf von "eigenen Funktionen" im Istwertzweig
- Überwachung des Istwerts auf Überschreitung von Alarm- und Warngrenzen
- Überwachung der Änderungsgeschwindigkeit des Istwerts
- Verstellung des Sollwertes durch Zeitplangeber
- Normierung eines externen Sollwertsignals bei Verhältnisregelungen
- Aufruf von "eigenen Funktionen" im Sollwertzweig
- Begrenzung der Änderungsgeschwindigkeit des Sollwerts
- Begrenzung der Absolutwerte des Sollwerts
- Filterung des Regeldifferenzsignals durch eine Totzone
- Überwachung der Regeldifferenz auf Überschreitung von Alarm- und Warngrenzen
- Einzeln zuschaltbaren P-, I- und D-Anteil
- Wahlweise P– und D–Wirkung in der Rückführung des Reglers.
- Aufschalten einer Störgröße
- Steuerung des Stellwerts im Handbetrieb von einem PG oder HMI aus)

- Aufruf von "eigenen Funktionen" im Stellwertzweig
- Begrenzung der Änderungsgeschwindigkeit des Stellwerts
- Begrenzung der Absolutwerte des Stellwerts
- Normierung des Stellwerts
- Pulsweitenmodulation für Impulsregler

Regeldifferenzbildung

Istwert

Normierung (PV_NORM)

Über das Optionskästen "Peripherie einschalten" legen Sie fest, welche analoge Eingangsgröße auf die physikalische Einheit des Istwerts normiert wird.

- Aktiviert: Eingangsgröße ist der Istwert Peripherie (PV_PER)
- Deaktiviert: Eingangsgröße ist der interne Istwert (PV_IN)

Zur eindeutigen Festlegung der Normierungsgeraden definieren Sie die folgenden vier Parameter:

- Eingang oben (NM_PIHR)
- Eingang unten (NM_PILR)
- Ausgang oben (NM_PVHR)
- Ausgang unten (NM_PVLR)

Je nach Sensortyp werden die Parameter vorbelegt.

Der Istwert wird nach folgender Formel aus dem jeweiligen Eingangswert PV_PER bzw. PV_IN berechnet:

MP4 = (PV_PER - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

bzw.

MP4 = (PV_IN - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

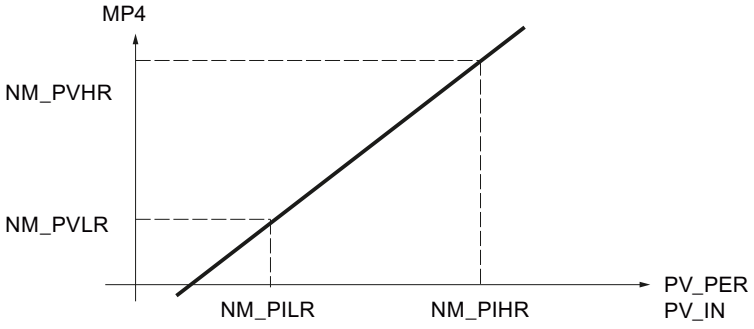

Der Istwert wird in der statischen Variablen MP4 ausgegeben.

Es werden keine Werte begrenzt, und eine Prüfung der Parameter findet nicht statt. Falls Sie für Eingang oben und Eingang unten denselben Wert eingeben, kann nach den obigen Formeln eine Division durch Null auftreten. Dies wird in der Anweisung nicht abgefangen!

Filterfunktionen

Regelgröße glätten (LAG1ST)

Zur Glättung von Prozesssignalen mit überlagerten schnellen Schwingungen (Noise) wird ein Verzögerungsglied erster Ordnung im Istwertzweig eingesetzt. Diese Funktion dämpft die analoge Regelgröße je nach eingestellter Zeitkonstante (PV_TMLAG) mehr oder weniger stark. Störsignale werden so wirksam unterdrückt. Insgesamt wird dadurch jedoch auch die Zeitkonstante des Gesamtregelkreises erhöht, d. h. die Regelung wird träger.

Die Dämpfungswirkung wird durch einen Verzögerungsalgorithmus 1. Ordnung erzielt.

Die Sprungantwort im Zeit–Bereich ist:

$outv(t) = MP4(0)\ (1 - e^{-t/PV_TMLAG})$

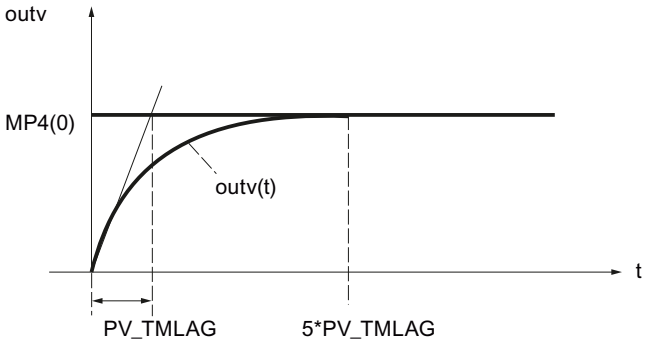

MP4(0)	die Höhe des Istwertsprungs am Eingang
outv(t)	die Ausgangsgröße

PV_TMLAG	die Verzögerungszeitkonstante
t	Zeit

Glättung des Istwerts

Wenn die Glättung des Istwerts (LAG1STON = FALSE) nicht aktiviert ist, dann ist der Peripherieeingang (PV_PER) bzw. der interne Eingang (PV_IN) verzögerungsfrei auf den Istwertzweig durchgeschaltet.

Verzögerungszeit

Der Eingangsparameter PV_TMLAG legt die Verzögerungszeit für den Istwert fest. Der gesamte Wertebereich ist zulässig.

Für PV_TMLAG ≤ 0.5 * CYCLE ist keine Verzögerung mehr wirksam.

Eine Abtastzeit (CYCLE) von weniger als einem Fünftel der Verzögerungszeit ist erforderlich, um ein dem analogen Verhalten annähernd entsprechendes Verzögerungsverhalten zu erzielen.

Quadratwurzel bilden (SQRT)

Die Funktion "Bildung der Quadratwurzel" (SQRT) dient der Linearisierung der Regelgröße. Liegt der von einem Geber gelieferte Istwert als physikalische Größe vor, die in einem quadratischen Zusammenhang mit der gemessenen Prozessgröße steht, dann muss der Verlauf der Regelgröße vor der Weiterverarbeitung im Regler linearisiert werden.

Das Messsignal muss immer dann durch Radizieren linearisiert werden, wenn Durchflussmessungen mithilfe von Messblenden oder Venturirohren usw. vorgenommen werden. Der gemessene Differenzdruck (Wirkdruck) ist dann proportional zum Quadrat des Durchflusses.

Über das Eingangssignal SQRT_ON = TRUE wird die Wurzelfunktion im Istwertzweig aktiviert. Der Algorithmus der Wurzelfunktion hat folgende Form:

outv = SQRT (MP5) × (SQRT_HR – SQRT_LR) / 10.0 + SQRT_LR

Dabei ist outv der normierte Ausgangswert der Wurzelfunktion (physikalischer Durchfluss).

Diese Formel sieht vor, dass der Eingangswert der Wurzel auf einen Zahlenbereich von 0 bis 100 normiert ist. In der Normierung des Istwerts müssen der "Messbereich oben" und "Messbereich unten" auf 100.0 und 0.0 parametriert werden. Die Wurzel aus diesem Wert ergibt einen Messbereich von 0 bis 10. Über die obere Grenze (SQRT_HR) und die untere Grenze des physikalischen Messbereichs (SQRT_LR) wird dieser Zahlenbereich normiert.

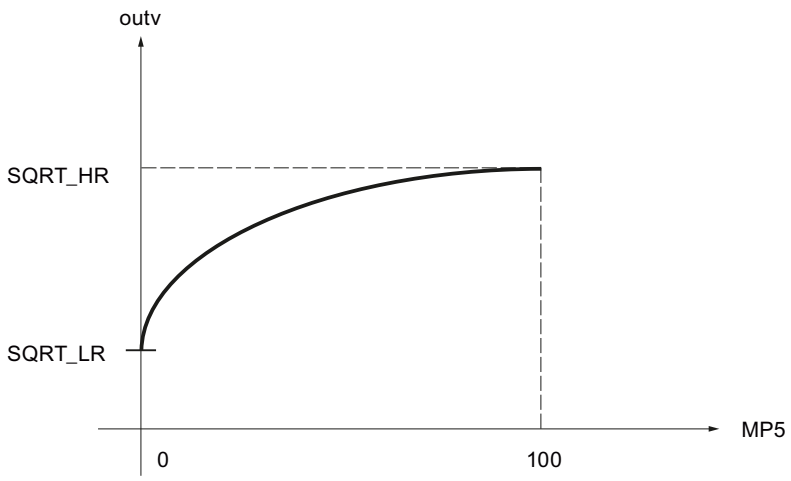

Beispiel zur Normierung

Der Eingangswert PV_IN des Reglers sei der Differenzdruck in mbar:

Messbereichs–Anfang NM_PILR	Messbereichs–Ende NM_PIHR	Wertebeispiel für PV_IN
20.0 mbar	200.0 mbar	150.0 mbar

Mit Hilfe der Normierfunktion PV_NORM erfolgt die Berechnung des normierten Differenzdrucks, wobei NM_PVHR = 100.0 und NM_PVLR = 0.0 gewählt wird:

MP4 = (PV_IN - NM_PILR) * (NM_PVHR - NM_PVLR) /
(NM_PIHR - NM_PILR) + NM_PVLR
= (PV_IN - 20.0 mbar) * (100.0 - 0.0) /
(200.0 mbar - 20.0 mbar) + 0.0
= (PV_IN - 20.0 mbar) * 100 / 180.0 mbar

Anfangswert von MP4	Endwert von MP4	Wertebeispiel für MP4
0.0	100.0	72.222

In diesem Beispiel wird keine Glättung verwendet und daher gilt: MP5 = MP4.

Für die Wurzel aus dem normierten Differenzdruck MP5 ergibt sich:

Anfangswert nach der Wurzel	Endwert nach der Wurzel	Wertebeispiel für SQRT(MP5)
0.0	10.0	8.498

Für den normierten Ausgangswert outv der Wurzelfunktion (physikalischer Durchfluss) folgt mit SQRT_HR = 20000.0 m^3/h und SQRT_LR = 0.0 m3/h:

outv = SQRT(MP5) * (SQRT_HR - SQRT_LR) / 10.0 + SQRT_LR
= SQRT(MP5) * (20000.0 m^3/h - 0.0 m3/h) / 10.0 + 0.0 m3/h
= 2000.0 m^3/h * SQRT(MP5)

Messbereichs–Anfang outv	Messbereichs–Ende outv	Wertebeispiel für outv
0.0 m^3/h	20000.0 m^3/h	16996.732 m^3/h

FC-Aufruf im Istwertzweig (PVFC)

Durch Einfügen einer anwenderspezifischen Funktion in den Istwertzweig kann die Regelgröße vor der Weiterverarbeitung im Regler einer Signalbehandlung, z. B. einer Signalverzögerung oder Linearisierung, unterzogen werden.

Funktion aufrufen

Wenn Sie eine anwenderspezifische Funktion in den Istwertzweig einfügen möchten, aktivieren Sie "FC Aufruf aktivieren" (PVFC_ON). Im Eingabefeld "FC Nummer" (PVFC_NBR) wird die Nummer der verwendeten Funktion eingegeben. Der Regler führt einen Aufruf der Funktion durch. Vorhandene Ein-/Ausgangsparameter der Anwender-Funktion werden dabei nicht versorgt. Die Datenübertragung müssen Sie selber in ihrer Funktion programmieren.

Den Ausgangswert der Funktion können Sie an der statischen Variable MP6 beobachten.

Hinweis

Eine Prüfung, ob eine Funktion vorhanden ist, wird nicht durchgeführt. Ist die Funktion nicht vorhanden, dann geht die CPU mit einem internen Systemfehler in STOP.

Alarm

Istwert auf Grenzwerte überwachen (PV_ALARM)

Werden bestimmte Werte der Prozessgrößen (z. B. Drehzahl, Druck, Füllstand, Temperatur, ...) über– oder unterschritten, können bei Regelungen unzulässige Prozesszustände oder Anlagenzustände auftreten. In diesen Fällen wird die Überwachung des Istwertes auf Grenzwerte (PV_ALARM) eingesetzt, um den Istwert auf Über–/ Unterschreitung des zulässigen Betriebsbereiches zu überwachen. Die Grenzüberschreitungen werden erfasst und gemeldet, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung des Istwerts auf Grenzwerte kann nicht ausgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Überwachung des Istwertes auf Grenzwerte (PV_ALARM) überwacht die Regelgröße PV(t) auf vier vorgebbare Grenzen in zwei Toleranzbändern. Werden die Grenzen erreicht oder überschritten, meldet die Funktion an der ersten Grenze "Warnung" und an der zweiten Grenze "Alarm".

Grenzwerte

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für "Warnung" und "Alarm" eingestellt. Überschreitet bzw. unterschreitet der Istwert (PV) diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits für den oberen Alarm (QPVH_ALM), die obere Warnung (QPVH_WRN), die untere Warnung (QPVL_WRN) und den unteren Alarm (QPVL_ALM) gesetzt.

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

QPVH_ALM	QPVH_WRN	QPVL_WRN	QPVL_ALM	wenn:	und:
TRUE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_ALM PV ≥ PVH_ALM - PV_HYS
FALSE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_WRN PV ≥ PVH_WRN - PV_HYS
FALSE	FALSE	TRUE	FALSE	PV ↘ PV ↗	PV ≤ PVL_WRN PV ≤ PVL_WRN + PV_HYS
FALSE	FALSE	TRUE	TRUE	PV ↘ PV ↗	PV ≤ PVL_ALM PV ≤ PVL_ALM + PV_HYS

Für ein sinnvolles Arbeiten des Bausteins muss gelten:

Oberer Alarm < Obere Warnung < Untere Warnung < Unterer Alarm

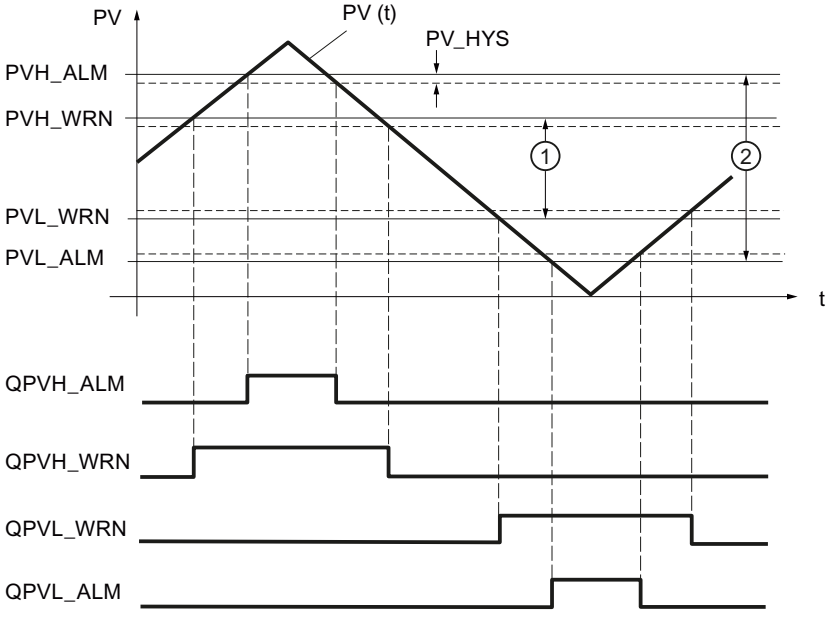

①	1. Toleranzband
②	2. Toleranzband

Hysterese

Um bei geringfügigen Änderungen der Eingangsgröße oder bei Rundungsfehlern ein "Flackern" der Meldebits zu verhindern, wird eine Hysterese (Rückschaltdifferenz) PV_HYS eingestellt. Die Regelgröße muss die Hysterese überwinden, bevor die Meldebits rückgesetzt werden.

Änderungsgeschwindigkeit des Istwertes überwachen (ROCALARM)

Wird die Änderung der Prozessgröße (z. B. Drehzahl, Druck, Füllstand, Temperatur, ...) zu groß, können bei Regelungen unzulässige Prozesszustände oder unzulässige Anlagenzustände auftreten. In diesen Fällen wird die Überwachung der Änderungsgeschwindigkeit des Istwertes (ROCALARM) eingesetzt, um den Istwert auf Überschreiten bzw. Unterschreiten einer zulässigen Änderungsgeschwindigkeit zu überwachen. Die Grenzüberschreitungen werden erfasst und gemeldet, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung der Änderungsgeschwindigkeit des Istwertes (ROCALARM) kann nicht abgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Funktion ROCALARM überwacht die Regelgröße PV(t) auf je nach Vorzeichen unterschiedlich vorgebbare Grenzen für die zulässige Änderungsgeschwindigkeit.

Die Zahlenwerte der Grenzsteigungen werden an den Eingangsparametern für "Maximaler Anstieg" und "Maximaler Abstieg" im positiven und negativen Bereich der Regelgröße eingestellt. Die Steigungen beziehen sich auf einen Anstieg bzw. Abfall in Prozent pro Sekunde. Überschreitet die Änderungsgeschwindigkeit der Regelgröße diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits QPVURLMP ... QPVDRLMN gesetzt.

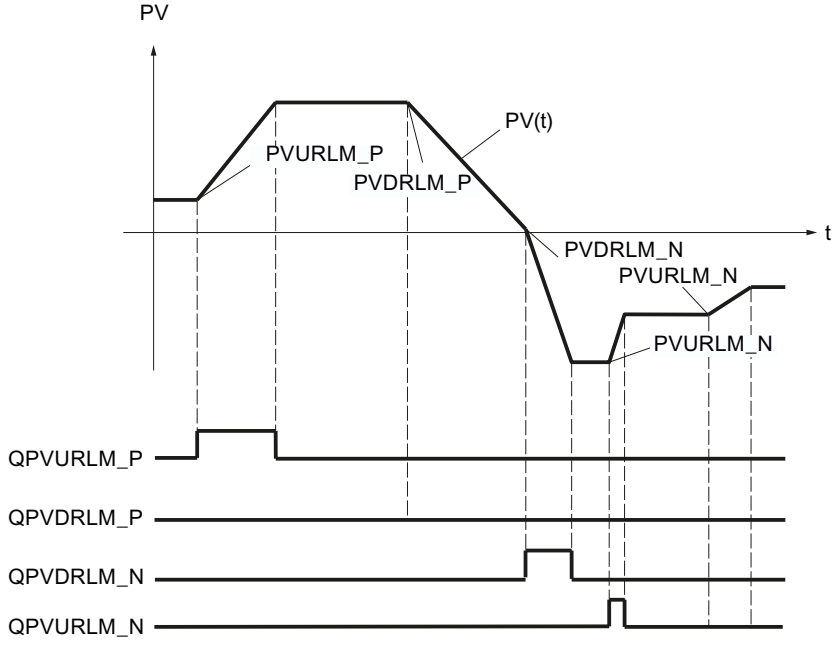

Die Kennzeichnung der Rampen–Parameter erfolgt nach folgendem Schema:

Parameter	PV–Änderung
Maximaler Anstieg im positiven Bereich (PVURLM_P)	PV > 0 und |PV| steigend
Maximaler Abstieg im positiven Bereich (PVDRLM_P)	PV > 0 und |PV| fallend
Maximaler Anstieg im negativen Bereich (PVURLM_N)	PV < 0 und |PV| steigend
Maximaler Abstieg im negativen Bereich (PVDRLM_N)	PV < 0 und |PV| fallend

Sollwert

Sollwert festlegen

Wählen Sie eine Quelle für den Sollwert aus. Folgende Optionen stehen zur Auswahl:

- Interner Sollwert
  Der Eingangsparameter SP_INT wird als Sollwert verwendet
- Sollwertgenerator
  Der Eingangsparameter SP_INT wird als Sollwert verwendet und kann über die statischen Variablen SPUP und SPDN vergrößert und verkleinert werden.
- Zeitplangeber
  Der Eingangsparameter SP_INT wird als Sollwert verwendet und kann nach einem vorgegebenen Zeitplan verändert werden.
- Externer Sollwert
  Bei Mischungs- und Verhältnisreglern sowie Kaskadenreglern wird der Eingangsparameter SP_EXT als Sollwert verwendet. Der externe Sollwert kann normiert werden.

Am Ausgangsparameter SP wird der wirksame Sollwert angezeigt.

Sollwert–Generator (SP_GEN)

Der interne Sollwert SP_INT kann über die Schalter SPUP und SPDN schrittweise vergrößert oder verkleinert werden. Der modifizierte Wert ist an MP1 zu beobachten. Die konfigurierten Begrenzungen des Sollwerts werden berücksichtigt.

Wirkung der Schalter SPUP und SPDN

Für eine feinstufige Änderung sollte der Regler eine Abtastzeit von maximal 100 ms haben.

Die Änderungsgeschwindigkeit der Ausgangsgröße hängt von der Betätigungsdauer der Schalter SPUP bzw. SPDN und von den eingestellten Begrenzungen nach folgender Beziehung ab:

- während der ersten 3 s nach Setzen von SPUP oder SPDN gilt:
  doutv/dt = (SP_HLM - SP_LLM) / 100 s
- danach
  doutv/dt = (SP_HLM - SP_LLM) / 10 s

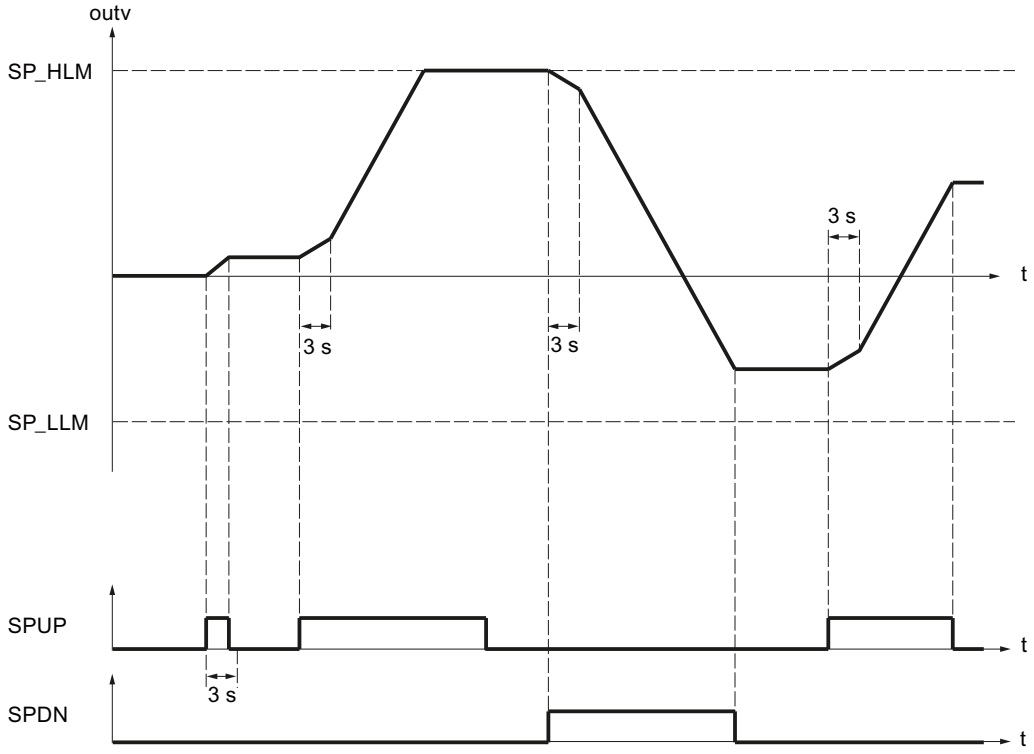

Bei einer Abtastzeit von 100 ms und einem Sollwertbereich von -100,0 bis 100,0 ändert sich z. B. der Sollwert in den ersten drei Sekunden um 0,2 pro Zyklus. Bleibt SPUP länger als 3 Sekunden TRUE, steigt die Änderungsgeschwindigkeit auf den zehnfachen Wert, hier also auf 2 pro Zyklus.

Anlauf– und Betriebsweise des Sollwertgenerators

- Bei Neustart wird der Ausgang outv auf 0.0 zurückgesetzt.
- Schalten Sie den Sollwertgenerator ein (SPGEN_ON = TRUE), dann wird am Ausgang outv zunächst der Signalwert SPFC_IN ausgegeben. Das Umschalten zum Sollwertgenerator aus einer anderen Betriebsart ist deshalb immer stoßfrei. Solange die Schalter SPUP und SPDN (Auf–/Ab–Tasten) nicht aktiviert werden, bleibt SPFC_IN am Ausgang stehen.

Parameter der Funktion SP_GEN

Der Ausgangsparameter outv ist ein impliziter Parameter. Er ist am Konfigurationswerkzeug über den Messpunkt MP1 zu beobachten.

Parameter	Bedeutung	Zulässiger Wertebereich
SPFC_IN	Sollwert FC Eingang	technischer Wertebereich
SP_INT	interner Sollwert	technischer Wertebereich

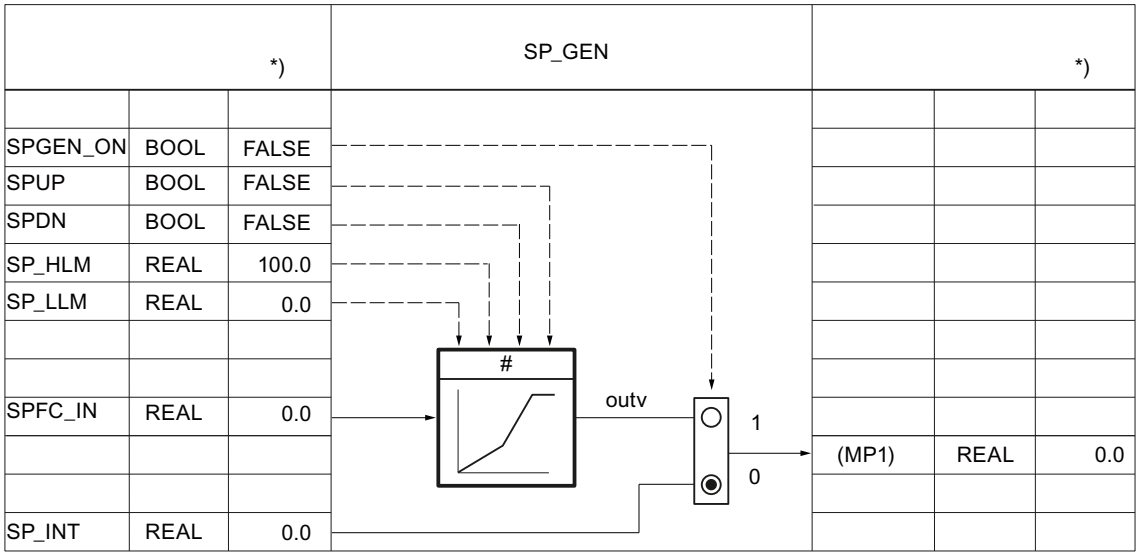

Bild 3-1 Funktionsschema und Parameter des Sollwert–Generators

Zeitplangeber (RMP_SOAK)

Falls der Sollwert SP_INT selbsttätig zeitabhängig geändert werden soll, z. B. bei der Regelung von Prozessen nach einem zeitgesteuerten Temperaturprogramm, so ist dies durch Projektieren eines Zeitplans und Aktivieren des Zeitplangebers möglich. Die Sollwertänderung wird aus Geradenabschnitten mit max. 256 Stützpunkten gebildet.

Hinweis

Die Bausteinfunktion überprüft nicht, ob ein globaler DB mit der Nummer ’DB_NBR’ vorhanden ist und ob der Parameter 'Anzahl der Stützpunkte' NBR_PTS zur DB-Länge passt. Bei inkorrekter Parametrierung geht die CPU mit einem internen Systemfehler in **STOP**.

Arbeitsweise des Zeitplangebers

Der Zeitplangeber liefert eine nach einem definierten Zeitplan ablaufende Funktion der Ausgangsgröße outv. Der Ablauf des Zeitplans wird durch Festlegen einer Reihe von Stützpunkten in einem globalen Datenbaustein DB_RMPSK mit den Zeitwerten PI[i].TMV und den zugehörigen Ausgangswerten PI[i].OUTV definiert.

- PID-CP
  AUTOHOTSPOT ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.
- PID-ES
  AUTOHOTSPOT ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Das folgende Bild zeigt einen Zeitplan mit 6 Stützpunkten.

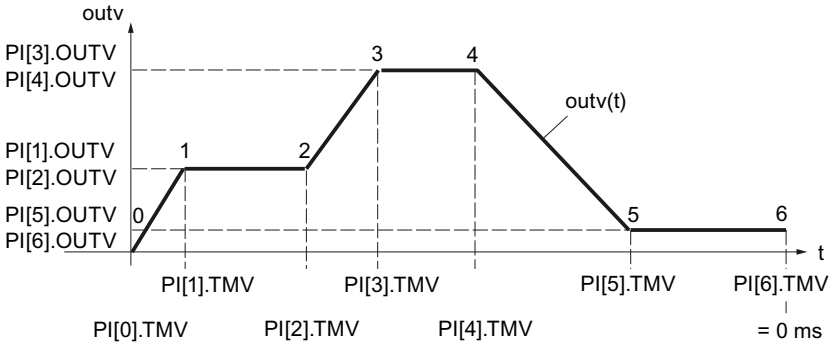

Für den letzten Stützpunkt n ist der Zeitwert PI[n].TMV = 0 ms. Die Bearbeitungszeit eines Zeitplans wird ausgehend vom Startwert bis auf Null heruntergerechnet.

PI[i].TMV gibt die Zeitabstände von Stützpunkt zu Stützpunkt an. Zwischen den Stützpunkten wird nach folgender Funktion interpoliert:

$$\text{outv}(t) = \text{PI}[n + 1].\ \text{OUTV} - \frac{\text{RS_TM}}{\text{PI}[n].\text{TMV}}\ (\text{PI}[n + 1].\text{OUTV} - \text{PI}[n].\text{OUTV})$$

$0 \leq n \leq (\text{NBR_PTS} - 1)$

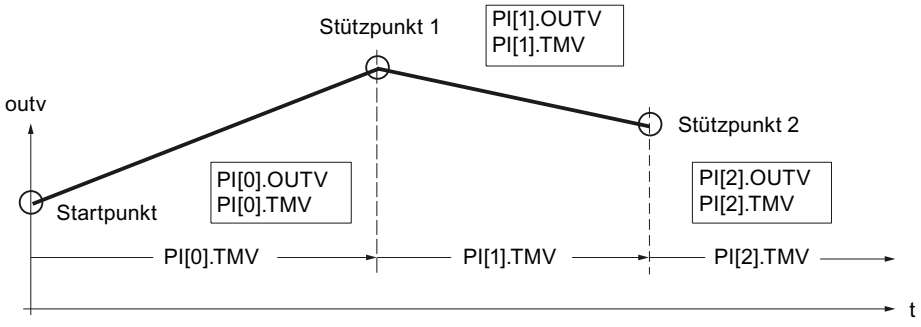

Hinweis

Hinweis

Beim Interpolieren der Fahrkurve zwischen den Stützpunkten kann es zum zeitweisen Verharren des Ausgangswerts kommen, wenn die Abtastzeit CYCLE sehr klein gegenüber der Zeit zwischen den Stützpunkten PI[n].TMV ist. Der Zeitplangeber kann aufgrund der CPU-Rechengenauigkeit nicht beliebig flache Fahrkurven linearisieren. Ist die Fahrkurve zu flach, verharrt der Ausgangswert eine Zeit lang am jeweiligen Stützpunkt, um nach einer gewissen Zeit mit der minimalen Steilheit zum nächsten Stützpunkt zu integrieren.

Abhilfe: Verkürzen Sie die Zeit zwischen den Stützpunkten, indem Sie weitere Stützpunkte einfügen. Sie nähern damit die ausgegebene Fahrkurve der gewünschten flachen Fahrkurve trapezförmig an.

Zeitplangeber einschalten

Mit dem Wechsel RMPSK_ON von FALSE nach TRUE wird der Zeitplangeber eingeschaltet. Nach Erreichen des letzten Stützpunktes ist der Zeitplan beendet. Bei erneutem Start der Funktion durch den Bediener muss RMPSK_ON zunächst auf 'FALSE' und dann wieder auf 'TRUE' gesetzt werden.

Bei **Neustar**t wird der Ausgang outv auf 0.0 zurückgesetzt und die Gesamtzeit bzw. Gesamtrestzeit ermittelt. Beim Übergang in den Normalbetrieb wird sofort die Fahrkurve vom Startpunkt aus nach eingestellter Betriebsart bearbeitet. Wird dies nicht gewünscht, muss der Parameter RMPSK_ON im Neustart–OB auf FALSE gesetzt werden.

Funktionen des Zeitplangebers

Die Anweisung verfügt über folgende Funktionen:

- Zeitplangeber einschalten für einmaligen Durchlauf
- Ausgang mit festem Wert vorbelegen
- Zyklische Wiederholung einschalten
- Zeitplangeber anhalten
- Zeitplangeber fortsetzen
- Gesamtbearbeitungszeit und Gesamtrestzeit aktualisieren

Zur Einstellung einer gewünschten Betriebsart gilt für die Wertigkeit der Steuereingänge folgende Tabelle:

Betriebsart	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Ausgangssignal OUTV
Zeitplangeber ein	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) Endwert wird nach Bearbeitungsende gehalten.
Ausgang vorgeben	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV

Betriebsart	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Ausgangssignal OUTV
Zyklische Wiederholung ein	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) nach Ende: automatischer Start
Zeitplangeber anhalten	TRUE	FALSE	TRUE	FALSE	**)	**)	aktueller Wert von OUTV(t) wird gehalten *)
Zeitplangeber fortsetzen	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (alt) *)
			FALSE				Es wird mit neu vorgegebenen Werten weitergefahren.
Gesamtzeit und Gesamtrestzeit aktualisieren	**)	**)	**)	**)	**)	FALSE	beeinflusst OUTV nicht
						TRUE	beeinflusst OUTV nicht

*) Bis zum nächsten Stützpunkt hat die Fahrkurve nicht die vom Anwender parametrierte Form!

**) Die jeweils eingestellte Betriebsart wird unabhängig von der Wertigkeit der Steuersignale in den mit **) gekennzeichneten Feldern ausgeführt

Ausgang des Zeitplangebers vorgeben

Wenn Sie diese Funktion aktivieren (DFRMP_ON = TRUE), dann wird der Ausgangswert des Zeitplangebers auf den Signalwert SP_INT bzw. den Ausgangswert SP_GEN gesetzt.

Hinweis

Die Vorbelegung mit dem internen Sollwert wirkt sich nur dann aus, wenn der Zeitplangeber eingeschaltet ist (RMPSK_ON = TRUE).

Nach dem Umschalten DFRMP_ON = FALSE wird outv linear vom eingestellten Sollwert (z. B. SP_INT) zum Ausgangswert der aktuellen Stützpunktnummer PI[NBR_ATMS].OUTV gefahren.

Die interne Zeitbearbeitung läuft auch bei durchgeschaltetem festem Sollwert (RMPSK_ON = TRUE und DFRMP_ON = TRUE) weiter.

Beim Start der Fahrkurve RMPSK_ON = TRUE wird so lange der feste Sollwert SP_INT ausgegeben, bis DFRMP_ON nach der Dauer von T* von TRUE nach FALSE wechselt.

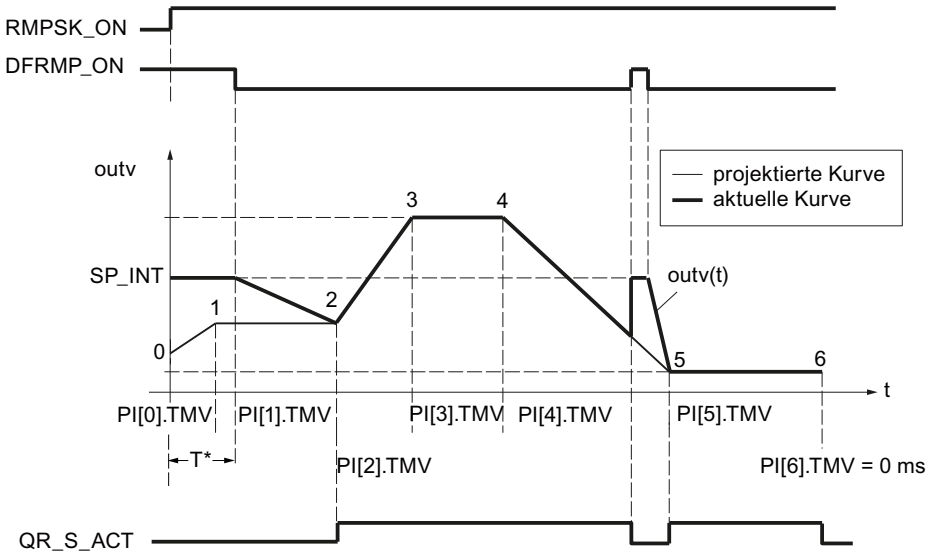

Zu diesem Zeitpunkt ist die Zeit PI[0].TMV und ein Teil der Zeit PI[1].TMV abgelaufen. OUTV wird von SP_INT nach PI[2].OUTV gefahren, d.h. zum Sollwert am Stützpunkt 2.

Erst ab dem Stützpunkt 2 wird die projektierte Fahrkurve ausgegeben. Die Übereinstimmung von projektierter und aktueller Fahrkurve wird angezeigt, indem QR_S_ACT = TRUE wird. Bei einer steigenden Flanke an DFRMP_ON während der Fahrkurvenbearbeitung springt der ausgegebene Wert OUTV ohne Verzögerung auf SP_INT.

Zyklische Wiederholung einschalten

Ist die Betriebsart 'Zyklische Wiederholung' (CYC_ON = TRUE) eingeschaltet, dann kehrt der Zeitplangeber nach Ausgabe des letzten Stützpunktwertes automatisch zum Startpunkt zurück und beginnt einen neuen Durchlauf.

Zwischen dem letzten Stützpunkt und dem Startpunkt wird nicht interpoliert. Für stoßfreien Übergang muss gelten: PI[NBR_PTS].OUTV = PI[0].OUTV.

Zeitplangeber anhalten

Mit RMP_HOLD = TRUE wird der Wert der Ausgangsgröße (einschließlich der Zeitbearbeitung) eingefroren. Bei Rücksetzen RMP_HOLD = FALSE wird an der Unterbrechungsstelle PI[x].TMV fortgefahren.

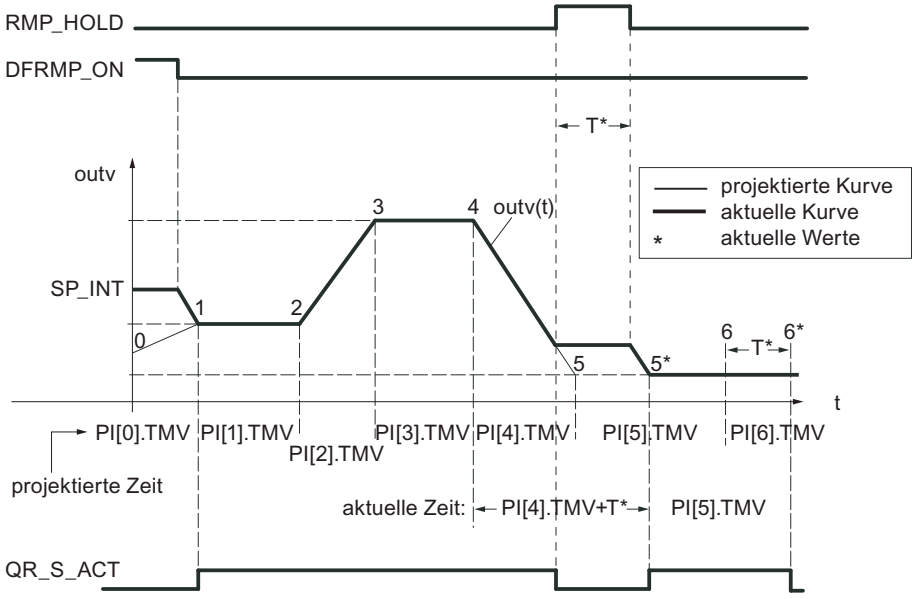

Die Bearbeitungszeit der Fahrkurve verlängert sich um die Zeit T*. Die Fahrkurve hat vom Startpunkt bis zur steigenden Flanke an RMP_HOLD und vom Stützpunkt 5* bis zum Stützpunkt 6* den projektierten Verlauf, d.h. das Ausgangssignal QR_S_ACT hat den Wert TRUE.

Ist der Schalter CONT_ON gesetzt, fährt der angehaltene Zeitplangeber an der vorgegebenen Stelle TM_CONT mit der Fahrkurvenbearbeitung fort.

Zeitplangeber fortsetzen

Mit CONT_ON wird eine angehaltene Fahrkurve an eine definierten Stelle wieder aufgenommen. Wenn der Zeitplangeber nicht angehalten ist, hat CONT_ON keine Auswirkungen.

Ist der Steuereingang für das Fortsetzen CONT_ON = TRUE gesetzt, dann wird der angehaltene Zeitplangeber am Stützpunkt TM_SNBR fortgesetzt. TM_CONT bestimmt die Restzeit, die der Zeitplangeber benötigt, bis der Stützpunkt TM_SNBR erreicht wird.

Das folgende Bild zeigt den Einfluss von RMP_HOLD und CONT_ON bei folgender Konfituration:

Fortsetzen am Stützpunkt 5 (TM_SNBR = 5) und Fortsetzen in T* s: TM_CONT = T*

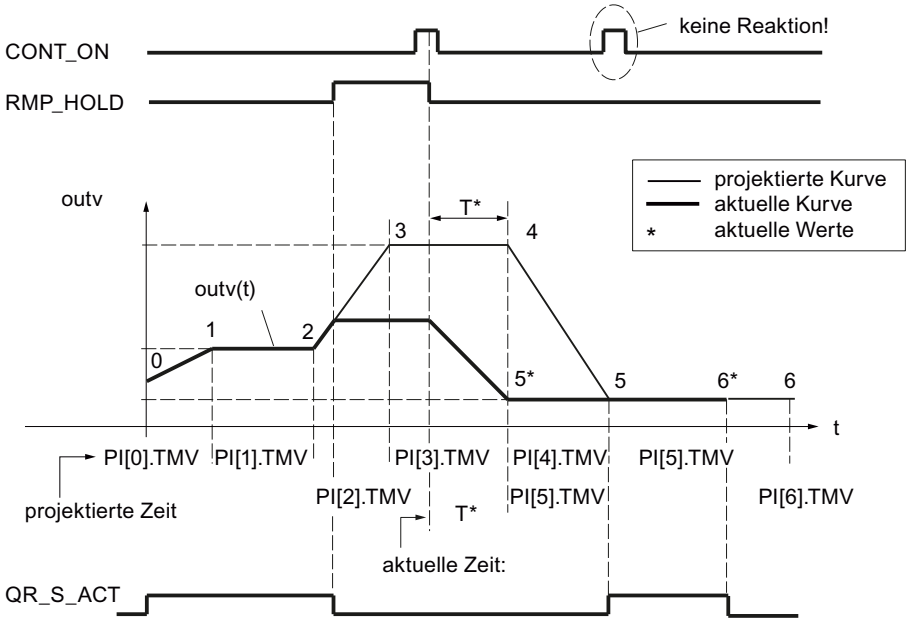

Die Stützpunkte 3 und 4 entfallen für diesen en Bearbeitungszyklus des Zeitplangebers. Nach Signalwechsel RMP_HOLD von TRUE nach FALSE wird der projektierte Kurvenverlauf erst ab Stützpunkt 5 wieder erreicht.

Der Ausgang QR_S_ACT wird nur dann gesetzt, wenn der Zeitplangeber die vom Anwender parametrierte Fahrkurve abarbeitet.

Gesamtzeit und Gesamtrestzeit aktualisieren

In jedem Zyklus werden die aktuelle Stützpunktnummer NBR_ATMS, die aktuelle Restzeit bis zum Erreichen des Stützpunktes RS_TM, die Gesamtzeit T_TM und die Gesamtrestzeit bis zum Erreichen des Fahrkurvenendes RT_TM aktualisiert.

Bei Online–Änderungen von PI[n].TMV ändert sich die Gesamtzeit und die Gesamtrestzeit der Fahrkurve. Da die Berechnung von T_TM und RT_TM bei vielen Stützpunkten die Bearbeitungszeit des Funktionsbausteins stark vergrößert, wird sie nur nach Neustart oder bei TUPDT_ON = TRUE durchgeführt. Die Zeitabschnitte PI[0 ... NBR_PTS].TMV zwischen den einzelnen Stützpunkten werden aufsummiert und am Ausgang Gesamtzeit T_TM und Gesamtrestzeit RT_TM angezeigt.

Bitte beachten Sie, dass die Ermittlung der Gesamtzeiten relativ viel CPU–Laufzeit erfordert!

Vorgehen

Um den Zeitplangeber zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Liste "Sollwertquelle" den Eintrag "Zeitplangeber".
2. Legen Sie einen AUTOHOTSPOT (PID-CP) bzw. AUTOHOTSPOT (PID-ES) an.
3. Definieren Sie die benötigen Parameter für den Zeitplan
4. Geben Sie die Nummer des Datenbausteins DB_RMPSK im Eingabefeld "DB-Nummer" ein.

5. Aktivieren oder deaktivieren Sie die weiteren Funktionen des Zeitplangebers.
6. Wenn Sie die Funktion Zeitplangeber fortsetzen aktiviert haben, geben Sie ein, an welchem Stützpunkt und wann der Zeitplangeber fortgesetzt werden soll.

Normierung

Normierung eines externen Sollwertes

Liegt der externe Sollwert nicht in der physikalischen Einheit des Istwerts vor (z. B. in % bei einer Reglerkaskade), dann müssen dieser Wert und sein Einstellbereich auf die physikalische Einheit des Istwertes normiert werden. Dies erfolgt über die Funktion "Normierung im Sollwertzweig".

Die Funktion SP_NORM normiert eine analoge Eingangsgröße. Der analoge externe Sollwert wird mit Hilfe der Normierungsgeraden in die Ausgangsgröße outv überführt. Der Ausgangswert OUTV ist am Konfigurationswerkzeug über den Messpunkt MP2 zugänglich.

Der Ausgangswert der Funktion ist wirksam, wenn der Steuereingang SPEXT_ON = TRUE ist. Zur eindeutigen Festlegung der Normierungsgeraden definieren Sie die folgenden vier Parameter:

- Eingang oben: SP_EXT: NM_SPEHR
- Eingang unten: SP_EXT: NM_SPELR
- Ausgang oben: NM_PVHR (Diesen Wert geben Sie in der Normierfunktion des Istwerts vor.)
- Ausgang unten: NM_PVLR (Diesen Wert geben Sie in der Normierfunktion des Istwerts vor.)

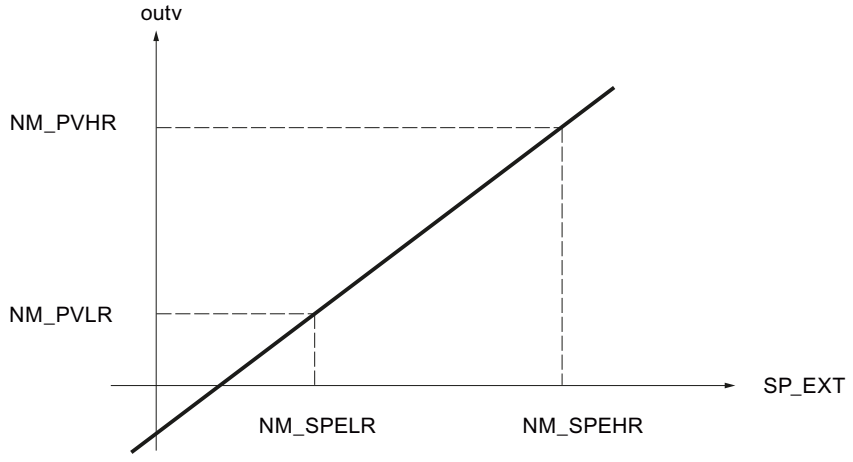

Der Ausgangswert outv wird gemäß folgender Formel aus dem jeweiligen Eingangswert SP_EXT berechnet:

outv = (SP_EXT – NM_SPELR) x (NM_PVHR – NM_PVLR) / (NM_SPEHR – NM_SPELR) + NM_PVLR

Für den Sonderfall einer eingeschalteten Wurzelfunktion im Istwertzweig gelten als Ober– und Untergrenze des Ausgangswerts outv die Normierungswerte der Wurzelfunktion (SQRT_HR und SQRT_LR).

Der Ausgangswert outv der Normierungsfunktion wird in diesem Fall gemäß folgender Formel aus dem jeweiligen Eingangswert SP_EXT berechnet:

outv = (SP_EXT – NM_SPELR) x (SQRT_HR – SQRT_LR) / (NM_SPEHR – NM_SPELR) + SQRT_LR

Funktionsintern werden keine Werte begrenzt, und eine Prüfung der Parameter findet nicht statt. Falls Sie für NM_SPEHR und NM_SPELR denselben Wert eingeben, kann nach den obigen Formeln eine Division durch Null auftreten. Dies wird in der Anweisung nicht abgefangen!

Mischungs-/Verhältnisfaktor

Bei einer Verhältnisregelung werden die Sollwerte der Regelkreise so aneinander angepasst, dass das eingestellte Verhältnis dem tatsächlichen Verhältnis der Istwerte entspricht.

Bei einer Mischungsregelung wird der gewünschte Mengenanteil der einzelnen Komponenten eingestellt. Die Summe der Mischungsfaktoren FAC muss dabei 1 sein.

Vorgehen

Um die Normierung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Geben in die Eingabefelder "Eingang oben" und "Eingang unten" die Werte für die Normierung ein.
2. Geben Sie einen Mischungs-/Verhältnis-Faktor (FAC) ein.

Begrenzungen

FC-Aufruf im Sollwertzweig

Durch Einfügen einer anwenderspezifischen Funktion in den Sollwertzweig ist es möglich, eine extern vorgegebene Führungsgröße vor Aufschaltung auf den Regler einer Signalbehandlung (z. B. einer Signalverzögerung oder Linearisierung) zu unterziehen.

Aktiviert man den FC-Aufruf im Sollwertzweig, wird eine anwenderspezifische Funktion aufgerufen. Der Regler führt einen Aufruf der FC durch. Vorhandene Ein–/ Ausgangsparameter der Anwender–FC werden dabei nicht versorgt. Die Datenübertragung müssen Sie deshalb mit S7–AWL in der Anwender–FC programmieren. Den aufbereiteten Sollwert können Sie an MP3 beobachten.

Der Wert von SPFC_ON bestimmt dann, ob in den Sollwertkanal an dieser Stelle eine frei programmierte Funktion in Form einer Standard–FC (z. B. eine Kennlinie) eingefügt wird oder ob der Sollwert ohne eine derartige Beeinflussung weiter verarbeitet wird.

Hinweis

Die Anweisung überprüft nicht, ob eine Funktion vorhanden ist. Ist die Funktion nicht vorhanden, dann geht die CPU mit einem internen Systemfehler in STOP.

Vorgehen

Um eine Funktion im Sollwertzweig aufzurufen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "FC-Aufruf aktivieren".
2. Geben Sie im Eingabefeld "FC" die Nummer der aufzurufenden Funktion ein.

Begrenzung der Änderungsgeschwindigkeit des Sollwertes

Rampenfunktionen im Sollwertzweig werden dann verwendet, wenn der Prozess keine sprungförmigen Änderungen des Stellsignals verträgt, denn eine sprungförmige Sollwertänderung hat im Allgemeinen auch einen Stellwertsprung des Reglers zur Folge. Diese abrupten Stellwertänderungen sind z. B. dann zu vermeiden, wenn zwischen geregeltem Motor und anzutreibender Last ein Getriebe zwischengeschaltet ist und ein zu schnelles Ansteigen der Motordrehzahl zu einer Überlastung des Getriebes führen würde.

Die Funktion SP_ROC begrenzt die Änderungsgeschwindigkeit der im Regler verarbeiteten Sollwerte sowohl getrennt für den An– und Abstieg als auch getrennt für den positiven und den negativen Bereich der Führungsgröße.

An den vier Eingängen SPURLM_P, SPDRLM_P, SPURLM_N und SPDRLM_N werden die jeweiligen Begrenzungen für die Steigungen der Rampenfunktion im positiven und negativen Bereich der Führungsgröße eingegeben. Die Steigungen beziehen sich auf einen Anstieg bzw. Abfall pro Sekunde. Schnellere Sollwertänderungen werden auf diese Grenzgeschwindigkeiten verzögert.

Wird z. B. SPURLM_P auf 10.0 [techn. Wertebereich/s] parametriert, so werden in jedem Abtastzyklus die folgenden Werte zum "Altwert" von outv addiert, und zwar solange inv > outv ist:

Abtastzeit	1 s	$outv_{alt}$ + 10
	100 ms	$outv_{alt}$ + 1
	10 ms	$outv_{alt}$ + 0.1

Die Funktionsweise der Signalbehandlung zeigt das folgende Bild anhand eines Beispiels. Aus Sprungfunktionen am Eingang inv(t) werden Rampenfunktionen am Ausgang outv(t).

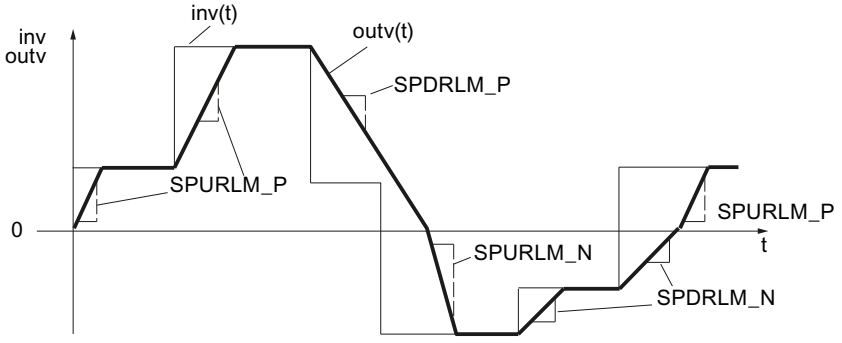

Bild 3-2 Begrenzung der Änderungsgeschwindigkeit der Führungsgröße SP(t)

Das Erreichen der Steigungsbegrenzungen wird nicht gemeldet!

Vorgehen

Um die Änderungsgeschwindigkeit zu begrenzen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "Änderungsgeschwindigkeit begrenzen".
2. Geben Sie sowohl für den positiven als auch für den negativen Bereich folgende Werte fest:
   - Sollwertanstieg
   - Sollwertabstieg

Begrenzung des Absolutwertes der Führungsgröße

Der Einstellbereich der Führungsgröße bestimmt den Bereich, in dem die Regelgröße schwanken kann, d. h. den Bereich, in dem sich der Prozess im Rahmen zulässiger Zustandswerte bewegt.

Zur Vermeidung von kritischen oder unzulässigen Prozesszuständen wird deshalb im Sollwertzweig von Standard PID Control der Einstellbereich der Führungsgröße nach oben und unten begrenzt (SP_LIMIT).

Die Funktion SP_LIMIT begrenzt den Sollwert SP so lange auf den vorgebbaren unteren und oberen Wert SP_LLM und SP_HLM, wie die Eingangsgröße INV außerhalb dieser Begrenzungen liegt. Da die Funktion nicht abgeschaltet werden kann, muss bei der Konfiguration immer die Angabe einer Unter– und Obergrenze berücksichtigt werden.

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für die untere und obere Begrenzung eingestellt. Bei Überschreitungen durch die Eingangsgröße inv(t) werden die zugehörigen Anzeigen über die Meldeausgänge ausgegeben.

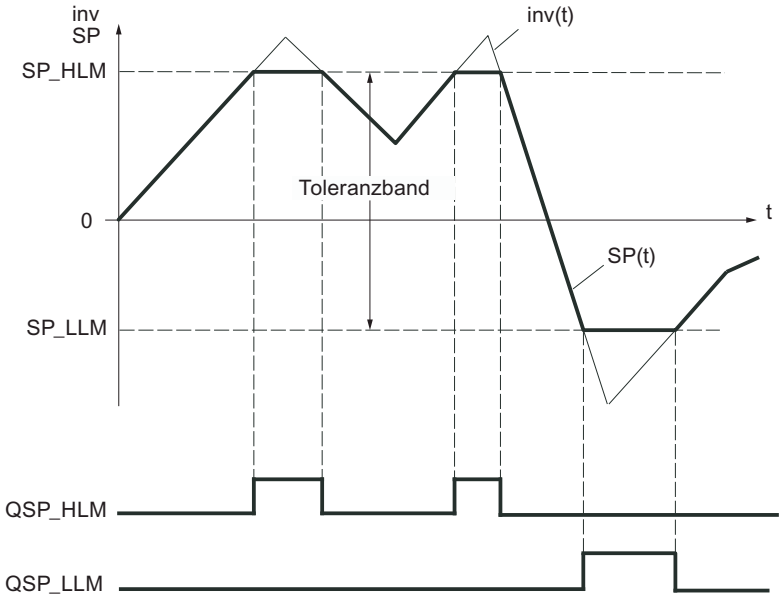

Bild 3-3 Absolutwertbegrenzung der Führungsgröße SP(t)

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

SP =PVH_ALM	QSP_HLM	QSP_LLM	wenn:
SP_HLM	TRUE	FALSE	inv ≥ SP_HLM
SP_LLM	FALSE	TRUE	inv ≤ SP_LLM
inv	FALSE	FALSE	SP_HLM < inv < SP_LLM

Vorgehen

Um den Absolutwert der Führungsgröße zu begrenzen, gehen Sie folgendermaßen vor:

1. Geben Sie im Eingabefeld "Obergrenze" die obere absolute Sollwertgrenze (SP_HLM) ein.
2. Geben Sie im Eingabefeld "Untergrenze" die untere absolute Sollwertgrenze (SP_LLM) ein.

Regeldifferenz

Totzone

Bei Überlagerung der Regel- oder der Führungsgröße mit einem höherfrequenten Störsignal wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam. Dies kann z. B. bei hoher Verstärkung des Reglers und eingeschalteter Differenzierung starke Schwankungen der Stellgröße verursachen. Wegen der vermehrten Schalthäufigkeit (Schrittregler) hat das einen erhöhten Verschleiß des Stellgliedes zur Folge.

Die Funktion vermindert im eingeschwungenen Zustand des Reglers den Rauschanteil im Signal der Regeldifferenz und vermindert dadurch ein unerwünschtes Oszillieren des Reglerausgangs.

Die Totzone kann abgeschaltet werden.

Signalfilterung durch Totzonenfunktion

Die Funktion DEADBAND unterdrückt in einem einstellbaren Bereich kleine Schwankungen der Eingangsgröße um einen festgelegten Nullpunkt. Außerhalb dieser Schwankungsbreite steigt bzw. fällt die Regeldifferenz ER proportional zur Eingangsgröße. Die Breite der unwirksamen Zone (Totzone) kann mit dem Parameter DEADB_W festgelegt werden. Die Totzonenbreite darf nur positive Werte annehmen.

Befindet sich die Eingangsgröße innerhalb der Totzone, wird am Ausgang der Wert 0 (Regeldifferenz = 0) ausgegeben. Erst wenn die Eingangsgröße diesen Unempfindlichkeitsbereich verlässt, steigt bzw. fällt der Ausgang um die gleichen Werte wie die Eingangsgröße inv. Daraus resultiert eine Verfälschung des übertragenen Signals auch außerhalb der Totzone. Das wird jedoch zur Vermeidung von Sprüngen an den Grenzen der Totzone in Kauf genommen (siehe nachfolgendes Bild). Die Verfälschung entspricht dem Wert DEADB_W und ist deshalb leicht kontrollierbar.

Die Funktion DEADBAND arbeitet gemäß folgenden Funktionen:

(ER) = inv + DEADB_W für inv < - DEADB_W

(ER) = 0 für -DEADB_W ≤ inv ≤ + DEADB_W

(ER) = inv - DEADB_W für inv > + DEADB_W

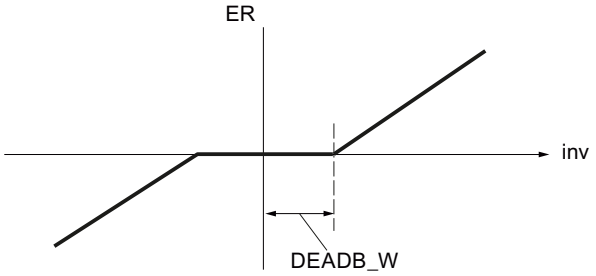

Auswirkungen der Signalfilterung können Sie am Ausgangsparameter ER beobachten.

Alarm

Zu große Abweichungen der Regelgröße vom eingestellten Sollwert können unerwünschte Prozesszustände hervorrufen. In diesen Fällen wird die Funktion ER_ALARM eingesetzt, um die Regeldifferenz auf Über–/Unterschreitung des zulässigen Betriebsbereiches zu überwachen. ER_ALARM erfasst und meldet evtl. auftretende Grenzüberschreitungen, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung der Regeldifferenz auf Grenzwerte kann nicht ausgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Funktion ER_ALARM überwacht die Größe der Regeldifferenz ER(t) auf vier vorgebbare Grenzen in zwei Toleranzbändern. Werden die Grenzen erreicht oder überschritten, meldet die Funktion zunächst "Warnung" und an der zweiten Grenze "Alarm".

Grenzwerte

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für "Warnung" und "Alarm" eingestellt. Überschreitet bzw. unterschreitet die Regeldifferenz (ER) diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits QERN_ALM ... QERP_ALM gesetzt.

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	wenn:	und:
TRUE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_ALM ER ≥ ERP_ALM - ER_HYS
FALSE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_WRN ER ≥ ERP_WRN - ER_HYS

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	wenn:	und:
FALSE	FALSE	TRUE	FALSE	ER ↘ ER ↗	ER ≤ ERP_WRN ER ≤ ERP_WRN + ER_HYS
FALSE	FALSE	TRUE	TRUE	ER ↘ ER ↗	ER ≤ ERP_ALM ER ≤ ERP_ALM + ER_HYS

Für ein sinnvolles Arbeiten des Bausteins muss gelten:

Oberer Alarm < Obere Warnung < Untere Warnung < Unterer Alarm

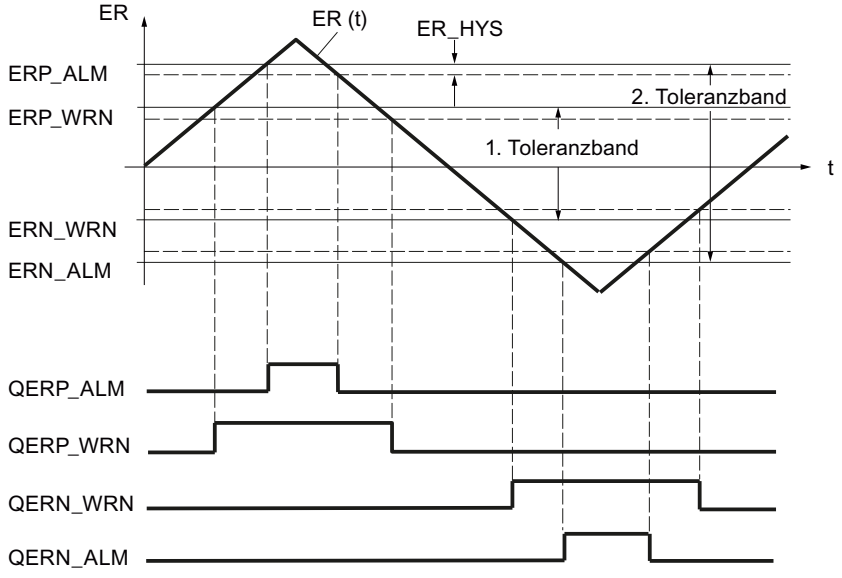

Hysterese

Um bei geringfügigen Änderungen der Eingangsgröße oder bei Rundungsfehlern ein "Flackern" der Meldebits zu vermeiden, wird eine Hysterese (Rückschaltdifferenz) ER_HYS eingestellt. Diese Hysterese muss die Regeldifferenz überwinden, bevor die Meldebits rückgesetzt werden.

Regler

Regelalgorithmus

Im Zyklus der projektierten Abtastzeit wird die Stellgröße des kontinuierlich arbeitenden Reglers aus der Regeldifferenz im PID–Stellungsalgorithmus errechnet. Der Regler ist in reiner Parallelstruktur ausgeführt. Der Proportional–, Integral– oder Differentialanteil kann jeweils einzeln abgeschaltet werden.

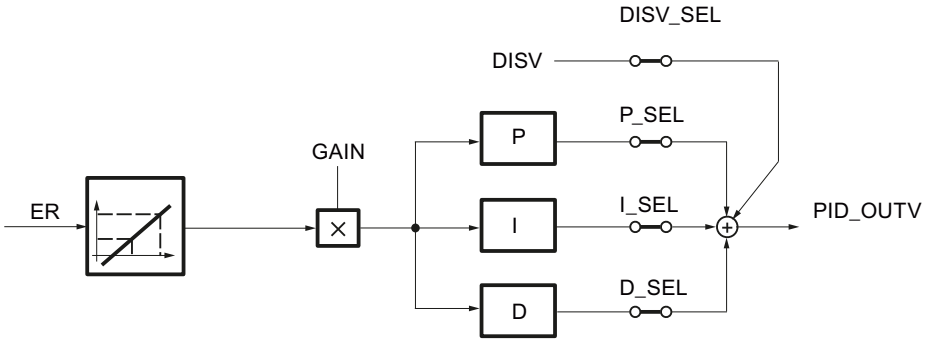

Störgrößenaufschaltung

Dem Ausgangssignal des Reglers PID_OUTV kann zusätzlich eine Störgröße DISV aufgeschaltet werden. Die Zu– bzw. Abschaltung erfolgt im PID–Fenster des Konfigurationswerkzeugs über den Strukturschalter DISV_SEL bzw. durch "Störgröße ein".

P- und D-Anteil in der Rückführung

In der Parallelstruktur erhält jeder Anteil des Regelalgorithmus die Regeldifferenz als Eingangssignal. Die Stellgröße wird über den P–Anteil den D–Anteil durch Sollwertsprünge unmittelbar beeinflusst. Eine andere Struktur des Reglers jedoch, bei der die Bildung des P–Anteils des D–Anteils in die Rückführung verlegt wird, garantiert bei Sprungänderungen der Führungsgröße einen stoßfreien Verlauf der Stellgröße. In dieser Struktur verarbeitet der I–Anteil die Regeldifferenz als Eingangssignal, auf den P–Anteil und den D–Anteil wird nur die negative Regelgröße (Faktor = -1) aufgeschaltet.

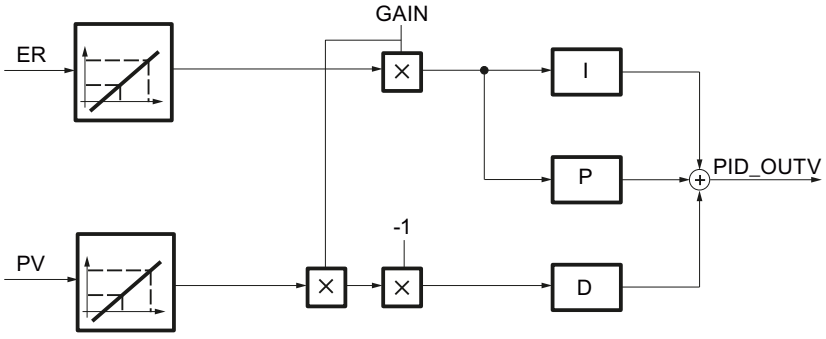

Normierung der Eingangsgrößen ER und PV

Die Eingangsgrößen ER und PV des PID–Reglers werden vor der Reglerbearbeitung normiert auf den Bereich 0 bis 100 gemäß folgenden Formeln:

- bei ausgeschalteter Wurzelfunktion (SQRT_ON = FALSE):
  - $ER_{normiert}$ = ER * 100.0 / (NM_PVHR – NM_PVLR)
  - $PV_{normiert}$ = (PV - NM_PVLR) * 100.0 / (NM_PVHR – NM_PVLR)
- bei eingeschalteter Wurzelfunktion (SQRT_ON = TRUE):
  - $ER_{normiert}$ = ER * 100.0 / (SQRT_HR – SQRT_LR)
  - $PV_{normiert}$ = (PV - SQRT_LR) * 100.0 / (SQRT_HR – SQRT_LR)

Diese Normierung wird durchgeführt, damit die Proportionalverstärkung GAIN des PID–Reglers dimensionslos eingegeben werden kann. Bei einer Änderung der Obergrenze und Untergrenze des physikalischen Messbereichs (z. B. von bar auf mbar) muss somit der Verstärkungsfaktor nicht geändert werden.

Die normierten Eingangsgrößen $ER_{normiert}$ und $PV_{normiert}$ sind nicht beobachtbar.

Siehe auch

Reglerstrukturen (Seite 859)

P-Anteil (Seite 863)

D-Anteil (Seite 866)

I-Anteil (Seite 863)

Reglerstrukturen

Einsatz und Parametrierung des PID–Reglers

Die meisten der in der Verfahrensindustrie und Prozessindustrie vorkommenden Regelstrecken lassen sich mit der PI–Reglerfunktion/PID–Reglerfunktion von Standard PID Control regeln. Nur in Spezialfällen sind zusätzliche Methoden und Maßnahmen zur Beherrschung der betreffenden Regelung erforderlich.

Ein großes praktisches Problem ist jedoch nach wie vor die Parametrierung des PI–Reglers/ PID–Reglers, d. h. die "richtigen" Einstellwerte für die Reglerparameter zu finden. Die Qualität dieser Parametrierung ist von ausschlaggebender Bedeutung für die aufgabengemäße Wirkung der PID–Regelung und erfordert entweder große praktische Erfahrung, Spezialkenntnisse oder einen hohen Zeitaufwand.

P-Regelung

Beim P–Regler sind der I–Anteil und der D–Anteil abgeschaltet. (I_SEL und D_SEL = FALSE). Das bedeutet, dass bei Regeldifferenz ER = 0 auch das Ausgangssignal OUTV = 0 ist. Soll ein Arbeitspunkt ≠ 0, d. h. ein Zahlenwert für das Ausgangssignal bei Regeldifferenz Null eingestellt werden, so ist der I–Zweig zu aktivieren (siehe nachfolgendes Bild).

Im I–Anteil lässt sich über entsprechende Einstellung des Initialisierungswertes I_ITLVAL ein Arbeitspunkt ≠ 0 für den P–Regler vorgeben. Setzen Sie dazu die Schalter 'I_ITL_ON' und 'I_SEL' = TRUE.

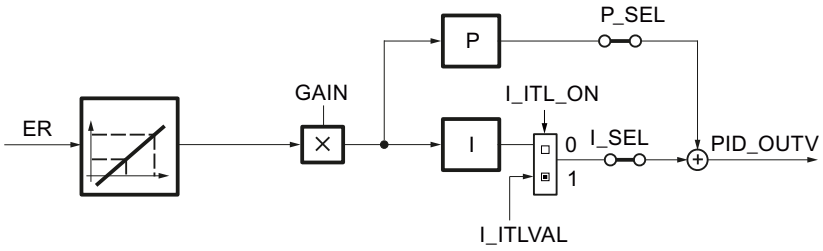

Die Sprungantwort des P–Reglers im Zeitbereich ist:
PID_OUTV (t) = I_ITLVAL + GAIN * ER $_{normiert}$ (t)

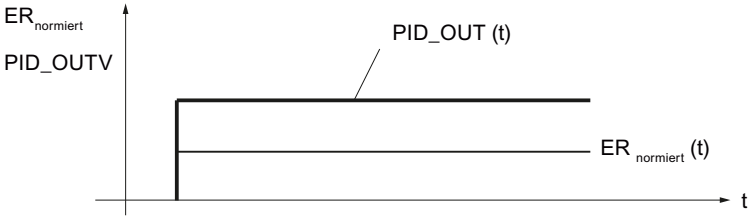

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
I_ITLVAL	der Arbeitspunkt des P–Reglers
GAIN	die Reglerverstärkung
ER $_{normiert}$ (t)	die normierte Regeldifferenz

PI-Regelung

Beim PI–Regler ist der D–Anteil abgeschaltet. (D_SEL = FALSE). Ein PI–Regler verstellt über den I–Anteil die Ausgangsgröße PID_OUTV so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0)\left(1 + \frac{1}{TI} * t\right)$$

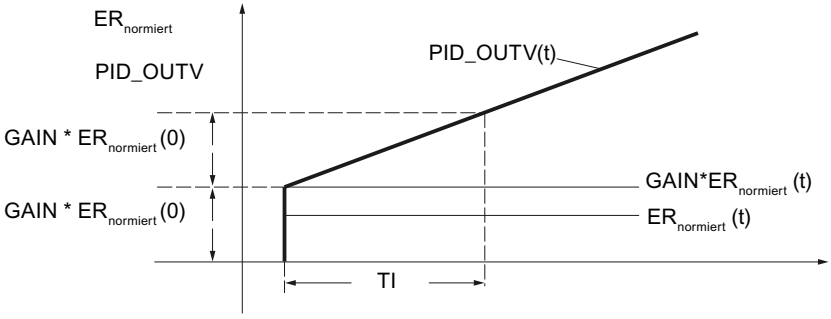

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TI	Integrationszeitkonstante

PD-Regelung

Beim PD–Regler ist der I–Anteil abgeschaltet (I–SEL = FALSE). Das bedeutet, dass bei Regeldifferenz ER = 0 auch das Ausgangssignal OUTV = 0 ist. Soll ein Arbeitspunkt ≠ 0, d. h. ein Zahlenwert für das Ausgangssignal bei Regeldifferenz Null eingestellt werden, so muss der I–Zweig aktiviert werden (siehe auch Blockschaltbild der P-Regelung.

Im I–Anteil lässt sich über entsprechende Einstellung des Initialisierungswertes I_ITLVAL ein Arbeitspunkt ≠ 0 für den PD–Regler vorgeben. Dazu sind die Schalter 'I_ITL_ON' und 'I_SEL' auf TRUE zu setzen.

Der PD–Regler bildet die Eingangsgröße ER(t) proportional auf das Ausgangssignal ab und addiert den durch Differentiation von ER(t) gebildeten D–Anteil dazu, der nach der Trapezregel (Padé–Näherung) doppeltgenau berechnet wird. Das Zeitverhalten wird durch die Differentiationszeitkonstante (Vorhaltzeit) TD bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeit: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert. Meist genügt ein kleiner Wert für TM_LAG, um den gewünschten Erfolg zu erzielen. Wird TM_LAG ≤ CYCLE/2 parametriert, dann ist die Verzögerung abgeschaltet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0)\left(1 + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$

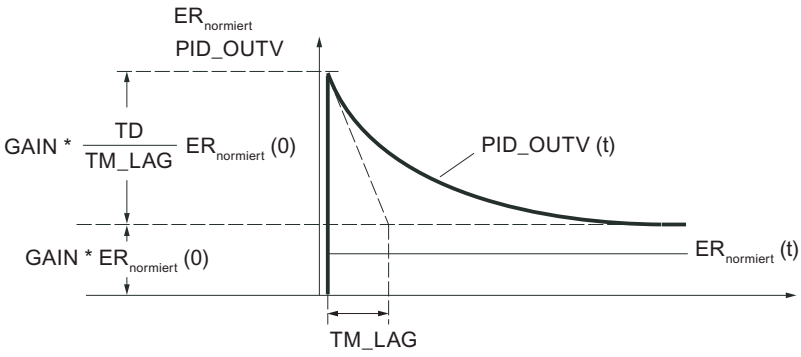

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TD	Differentiationszeit
TM_LAG	Verzögerungszeit

PID-Regelung

Beim PID–Regler sind der P–Anteil, I–Anteil und D–Anteil eingeschaltet (P–SEL = TRUE, I–SEL = TRUE, D_SEL = TRUE). Ein PID–Regler verstellt die Ausgangsgröße PID_OUTV über den I–Anteil so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet. Werden Stellwertbegrenzungen überschritten, behält der I–Anteil den an der Grenze erreichten Wert bei (Anti Reset Wind–up).

Der PID–Regler bildet die normierte Eingangsgröße $ER_{normiert}(t)$ proportional auf das Ausgangssignal ab und addiert die durch Differentiation und Integration von $ER_{normiert}(t)$ gebildeten Anteile dazu, die nach der Trapezregel (Padé–Näherung) doppeltgenau berechnet werden. Das Zeitverhalten wird durch die Differentiationszeit (Vorhaltzeit) TD und die Integrationszeit (Nachstellzeit) TI bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeitkonstante: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert. Meist genügt ein kleiner Wert für TM_LAG, um den gewünschten Erfolg zu erzielen. Wird TM_LAG ≤ CYCLE/2 parametriert, dann ist die Verzögerung abgeschaltet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0) \left(1 + \frac{1}{TI} * t + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}} \right)$$

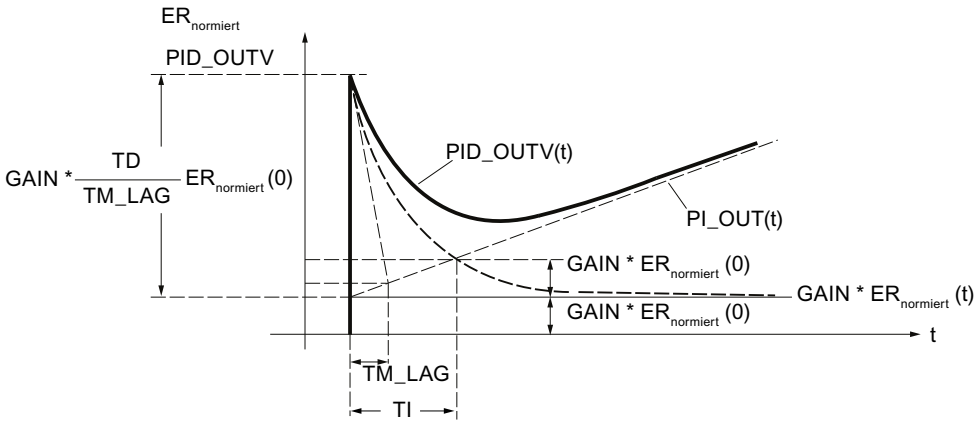

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TI	Integrationszeitkonstante
TD	Differentiationszeit
TM_LAG	Verzögerungszeit

Hinweis

Bei einer Änderung von TD sollten Sie auch TM_LAG entsprechend anpassen.

Empfehlung: 5 ≤ (TM / TM_LAG) ≤ 10

Siehe auch

Regelalgorithmus (Seite 858)

P-Anteil (Seite 863)

D-Anteil (Seite 866)

I-Anteil (Seite 863)

P-Anteil

Proportionalverstärkung

Das Vorzeichen der Proportionalverstärkung bestimmt die Wirkungsrichtung des kontinuierlichen Reglers.

- Positive Proportionalverstärkung
  steigende Regeldifferenz → steigender Stellwert
- Negative Proportionalverstärkung
  steigende Regeldifferenz → fallender Stellwert

P-Anteil in Rückführung legen

Ist der P-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den P-Anteil. Die Stellgröße macht daher – wie auch bei einem Sollwertsprung – bei einer Hand-Automatik-Umschaltung keinen Sprung durch den P-Anteil; die Umschaltung ist stoßfrei.

Siehe auch

Regelalgorithmus (Seite 858)

I-Anteil

Funktion des I-Anteils

Der I-Anteil sorgt dafür, dass durch Nachstellung des Arbeitspunktes die Regeldifferenz für beliebige Werte der Stellgröße zu 0 werden kann.

Der I-Anteil erzeugt ein Ausgangssignal, dessen Änderungsgeschwindigkeit sich proportional zur Änderung des Absolutwertes der Eingangsgröße verhält. Das Zeitverhalten wird durch die Integrationszeit TI bestimmt.

Die Sprungantwort auf einen Eingangsprung inv_0 ist:

$$OUTV(t) = \frac{1}{TI} \int inv(t)\, dt$$

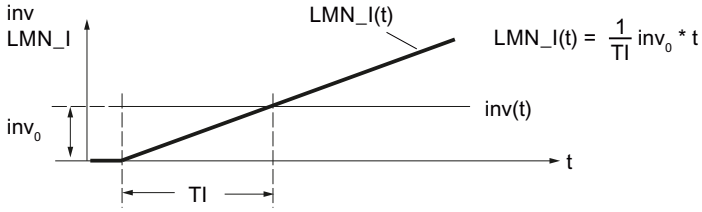

LMN_I(t)	die Ausgangsgröße des I-Anteils
inv_0	die Sprunghöhe am Eingang des I-Anteils
TI	Integrationszeit

Der I-Anteil ist an der statischen Variable LMN_I zu beobachten.

Im ausgeschalteten Zustand (I_SEL = FALSE) wird der I-Anteil, d. h. der interne Speicher und die statische Variable LMN_I auf Null gesetzt.

Der Ausgang und der Speicher des I-Anteils wird durch die absoluten Stellwertgrenzen LMN_HLM und LMN_LLM begrenzt (Anti Reset Wind–up).

Bei Beeinflussung des Stellsignals im Handbetrieb, d. h. wenn MAN_ON, LMNOP_ON oder CAS_ON = TRUE ist, wird der interne Speicherwert des I-Anteils dem Wert LMNFC_IN - LMN_P - DISV nachgeführt.

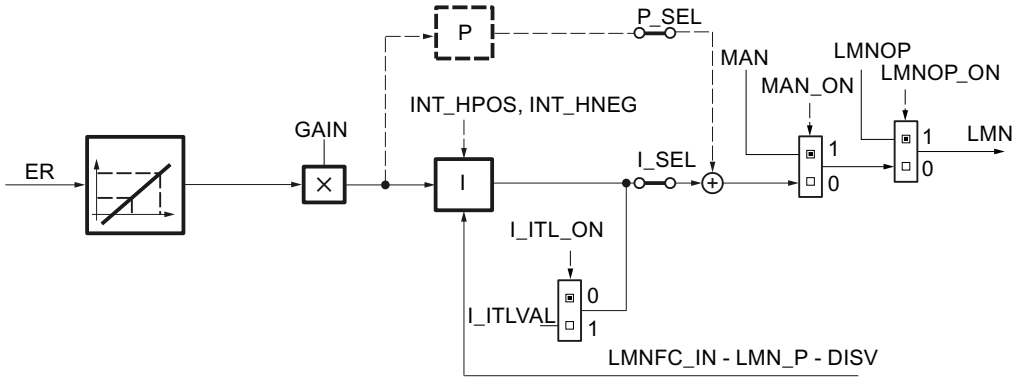

Stoßfreie Umschaltung von Hand nach Automatik

Im Handbetrieb werden der I- und D-Anteil des Reglers so nachgeführt, dass der Regler bei der Umschaltung auf Automatikbetrieb mit einer sinnvollen Stellgröße beginnt.

- Stoßfreie Umschaltung aktiviert (SMOO_CHG = TRUE)
  Der I–Anteil wird im Handbetrieb so gesetzt, dass die Stellgröße bei der Hand–Automatik–Umschaltung zunächst unverändert bleibt.
  Der D-Anteil wird im Handbetrieb auf Null gesetzt. Die Umschaltung auf Automatik erfolgt ohne Stellgrößensprung.
  Eine anstehende Regelabweichung wird langsam ausgeregelt.
- Stoßfreie Umschaltung deaktiviert (SMOO_CHG = FALSE)
  Die Stellgröße macht bei der Hand–Automatik–Umschaltung ausgehend von der aktuellen Stellung einen Sprung, dessen Höhe durch den P-Anteil bestimmt wird. Die Sprunghöhe des P-Anteils entspricht der Stellgrößenänderung bei einem Sollwertsprung vom aktuellen Istwert auf den aktuellen Sollwert.
  Der D-Anteil wird im Handbetrieb auf einen Wert gesetzt, der der anstehenden Regeldifferenz entspricht. Die Umschaltung auf Automatik erfolgt mit Stellgrößensprung, die Regeldifferenz wird schneller ausgeregelt.
  Ist jedoch der P-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den P–Anteil. Die Stellgröße macht daher bei einer Hand-Automatik-Umschaltung keinen Sprung durch den P-Anteil; die Umschaltung ist stoßfrei. Das gleiche gilt auch für den D–Anteil, falls dieser
  in die Rückführung gelegt wurde (DFDB_SEL = TRUE).
  Die anstehende Regeldifferenz wird schneller ausgeregelt. Dies ist z. B. bei Temperaturregelstrecken wünschenswert.

Zulässige Bereiche für die Integrationszeit TI und CYCLE

Wegen der begrenzten Genauigkeit der in der CPU errechneten REAL–Zahlen kann beim Integrieren folgender Effekt auftreten: wurde die Abtastzeit CYCLE der Regelung im Vergleich zur Integrationszeit TI zu klein gewählt und ist der Eingangswert inv des I-Anteils klein gegenüber seinem Ausgangswert OUTV, dann spricht der I-Anteil nicht an und bleibt auf seinem momentanen Ausgangswert stehen.

Dieser Effekt lässt sich vermeiden, wenn bei der Konfiguration folgende Dimensionierungsregel beachtet wird:

$CYCLE > 10^{-4} \times TI$

Damit reagiert der I-Anteil noch auf Änderungen der Eingangswerte, die im Bereich von millionstel Promille der aktuellen Ausgangsgröße liegen:

$inv > 10^{-10} \times OUTV$

Damit das Übertragungsverhalten des Integrier–Algorithmus dem analogen Verhalten entspricht, sollte die Abtastzeit kleiner als 20 % der eingestellten Integrationszeit sein bzw. TI mindestens den fünffachen Wert der gewählten Abtastzeit haben:

$CYCLE < 0{,}2 \times TI$

Der Algorithmus lässt Werte für die Abtastzeit bis $CYCLE \leq 0{,}5 * TI$ zu.

I–Anteil initialisieren

Wird die Initialisierung des I-Anteils aktiviert, dann wird bei Neustart der CPU oder COM_RST = TRUE der Initialisierungswert auf die statische Variable LMN_I durchgeschaltet. Beim Übergang in den Normalbetrieb durch Zurücksetzen der Initialisierung beginnt der I-Anteil vom Initialisierungswert aus mit der Integration seiner Eingangsgröße.

I-Anteil anhalten

Über die statischen Variablen INT_HPOS und INT_HNEG kann der I-Anteil in positive bzw. negative Richtung blockiert werden. Dies kann sinnvoll sein bei Reglerkaskaden. Gerät z. B. die Stellgröße des Folgereglers in die obere Begrenzung, kann somit eine weitere Vergrößerung der Stellgröße des Führungsreglers durch dessen I-Anteil verhindert werden. Dieses Verhalten programmieren Sie durch die folgenden Anweisungen, z. B. in AWL:

```
U    "Folgeregler".QLMN_HLM
=    "Fuehrungsregler".INT_HPOS
U    "Folgeregler".QLMN_ LLM
=    "Fuehrungsregler".INT_HNEG
```

Siehe auch

Regelalgorithmus (Seite 858)

Reglerstrukturen (Seite 859)

D-Anteil

Funktion des D-Anteils

Der D-Anteil erzeugt ein Ausgangssignal, dessen Größe sich proportional zur Änderungsgeschwindigkeit der Eingangsgröße ändert. Das Zeitverhalten wird durch die Differenzierzeit TD und die Verzögerungszeit TM_LAG bestimmt.

Die Sprungantwort auf einen Eingangssprung inv_0 ist:

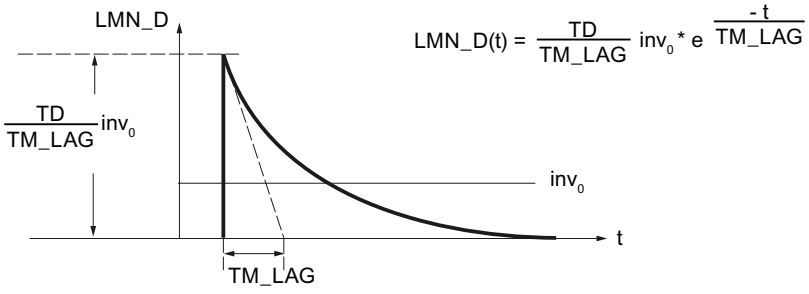

LMN_D(t)	die Ausgangsgröße des D-Anteils
inv_0	die Sprunghöhe am Eingang des D-Anteils
TD	Differenzierzeit
TM_LAG	Verzögerungszeit

Der D-Anteil ist an der statischen Variable LMN_D zu beobachten.

Im ausgeschalteten Zustand (D_SEL = FALSE) wird der D-Anteil, d. h. der interne Speicher und die statische Variable LMN_D auf Null gesetzt.

D-Anteil in Rückführung legen

Ist der D-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den D-Anteil. Die Stellgröße macht daher – wie auch bei einem Sollwertsprung – bei einer Hand-Automatik-Umschaltung keinen Sprung durch den D-Anteil; die Umschaltung ist stoßfrei.

Zulässige Werte für Differenzier- und Verzögerungszeit

Damit der D-Anteil korrekt berechnet werden kann, müssen für die Differenzier- und Verzögerungszeit folgende Bedingungen eingehalten werden:

- TD ≥ CYCLE und
- TM_LAG ≥ 0,5 × CYCLE

Wenn TD < CYCLE, dann arbeitet der D-Anteil so, als würde TD den Wert CYCLE haben.

Wenn TM_LAG < 0,5 × CYCLE, dann arbeitet der D-Anteil ohne Verzögerung. Ein Eingangssprung wird dann mit dem Faktor TD/CYCLE multipliziert und dieser Wert als "Nadelimpuls" auf den Ausgang gegeben. Das heißt, im folgenden Bearbeitungszyklus wird LMN_D wieder auf Null zurückgenommen.

Koeffizient DT1

Der Koeffizient DT1 wird folgendermaßen berechnet:

Koeffizient DT1 = TD/TM_LAG

Verzögerungszeit

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeitkonstante: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert.

Meist genügt ein kleiner Wert als Verzögerungszeit, um den gewünschten Erfolg zu erzielen. Wird die Verzögerungszeit kleiner als die Hälfte der Reglerabtastzeit parametriert, dann ist die Verzögerung abgeschaltet.

Vorgehen

Um den D-Anteil zu konfigurieren, gehen Sie folgendermaßen vor:

1. Um den D-Anteil in Rückführung zu legen, aktivieren Sie das entsprechende Optionskästchen.
2. Geben Sie die Differenzierzeit in Sekunden ein.
3. Geben Sie die Verzögerungszeit in Sekunden ein.

Siehe auch

Regelalgorithmus (Seite 858)

Stellwert

Handbetrieb

Neben der Betriebsart "Automatik" mit auf den Ausgang durchgeschaltetem Ausgang des PID–Algorithmus (PID_OUTV) gibt es die zwei Betriebsarten, bei denen das Stellsignal manuell beeinflusst werden kann: "Handbetrieb ohne Schalterbedienung" und "Handbetrieb über Mehr–/Weniger–Schalter" (MAN_GEN).

Über den Parameter MAN kann eine externe Stellgrößenbeeinflussung aufgeschaltet oder vom Anwenderprogramm aus vorgegeben werden. Der Eingangswert MAN wird auf die Stellgrößenbegrenzungen LMN_HLM (obere) und LMN_LLM (untere) begrenzt.

Die Struktur der Handwert–Erzeugung und Handwert–Aufschaltung geht aus nachfolgendem Bild hervor. Beim Einschalten von MAN_GEN aus einer anderen Betriebsart wird der momentan wirksame Stellwert am Ausgang von MAN_GEN übernommen. Die Umschaltung zum Handwertgenerator ist damit immer stoßfrei.

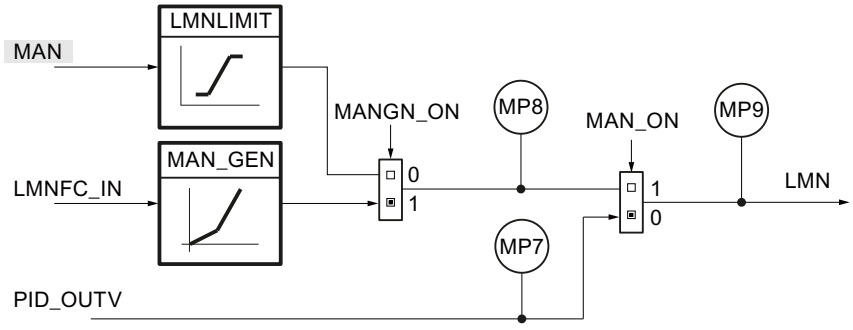

Automatikbetrieb aktivieren

Ist der Handbetrieb ausgeschaltet (MAN_ON = FALSE), wird der Stellwert des PID–Algorithmus auf den Ausgang durchgeschaltet. Im Handbetrieb (MAN_ON = TRUE) wird der I–Anteil des Reglers so nachgeführt, dass der Regler bei der Umschaltung auf Automatikbetrieb mit einer sinnvollen Stellgröße beginnt. Der Ausgang des PID–Algorithmus ist am Messpunkt MP7 hinterlegt.

Im Automatikbetrieb wird der Handwert MAN der Stellgröße (abzüglich D–Anteil) nachgeführt. Bei der Umschaltung auf Handbetrieb bleibt die Stellgröße daher auf dem zuletzt berechneten Wert stehen. Sie kann per Bedienung verändert werden.

1. Deaktivieren Sie das Optionskästchen "Handbetrieb aktivieren".

Handbetrieb mit Handwert aktivieren

In dieser Betriebsart (MANGN_ON = FALSE und MAN_ON = TRUE) wird der Handstellwert als Absolutwert am Eingang MAN eingegeben. Der Handstellwert wird am Messpunkt MP8 angezeigt.

1. Aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
2. Wählen Sie die Option "Handwert".

Handwertgenerator aktivieren

Der Handwert MAN kann über die Schalter MANUP und MANDN schrittweise vergrößert oder verkleinert werden. Der modifizierte Wert ist an MP8 zu beobachten. Die konfigurierten Begrenzungen des Stellwerts werden berücksichtigt.

1. Aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
2. Wählen Sie die Option "Handwertgenerator".

Wirkung der Schalter MANUP und MANDN

Für eine feinstufige Änderung sollte der Regler eine Abtastzeit von maximal 100 ms haben.

Die Änderungsgeschwindigkeit des Stellwerts hängt von der Betätigungsdauer der Schalter MANUP bzw. MANDN und von den eingestellten Begrenzungen nach folgender Beziehung ab:

- während der ersten 3 s nach Setzen von MANUP oder MANDN gilt:
  doutv/dt = (LMN_HLM - LMN_LLM) / 100 s
- danach
  doutv/dt = (LMN_HLM - LMN_LLM) / 10 s

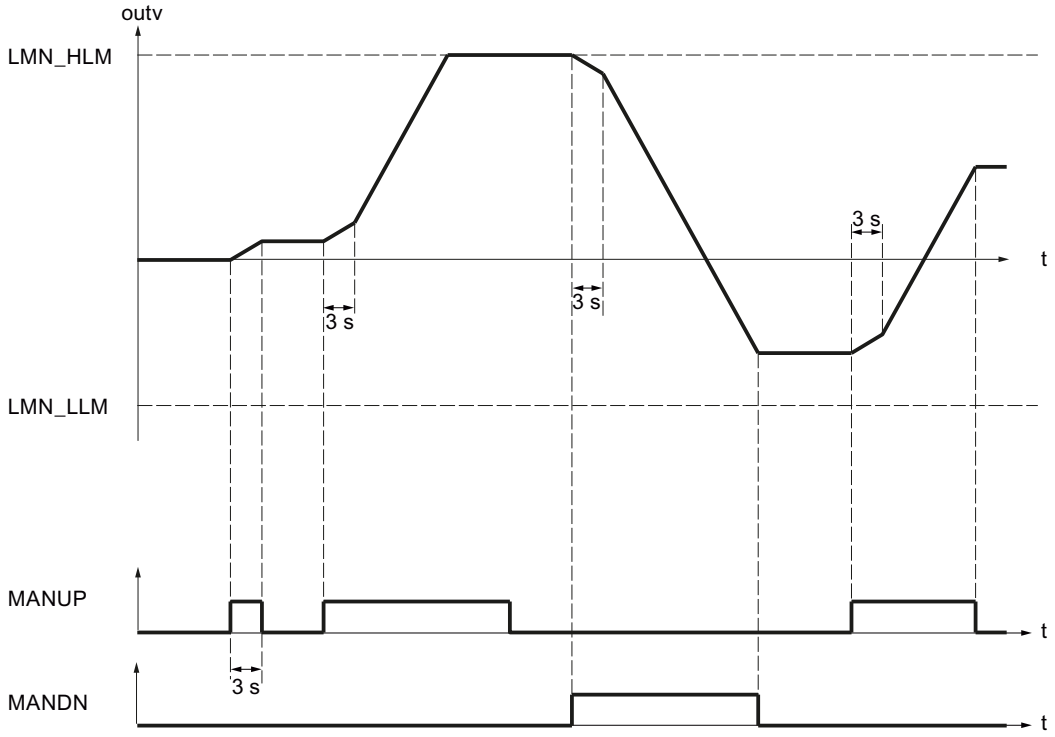

Bei einer Abtastzeit von 100 ms und einem Stellwertbereich von -100,0 bis 100,0 ändert sich z. B. der Stellwert in den ersten drei Sekunden um 0,2 pro Zyklus. Bleibt MANUP länger als 3 Sekunden TRUE, steigt die Änderungsgeschwindigkeit auf den zehnfachen Wert, hier also auf 2 pro Zyklus.

Der Handwertgenerator wird stoßfrei aktiviert.

Betriebsarten

Folgende Tabelle zeigt die möglichen Betriebsarten des kontinuierlichen Reglers mit den erforderlichen Werten der Schalter.

Betriebsart	MAN_ON	MANGN_ON
Automatikbetrieb	FALSE	nicht relevant
Handbetrieb mit Handwert	TRUE	FALSE
Handbetrieb mit Handwertgenerator	TRUE	TRUE

Begrenzungen

Eine Funktion aufrufen

Wenn Sie eine anwenderspezifische Funktion in den Stellgrößenzweig einfügen, können Sie die im Regler gebildete Stellgröße PID_OUTV vor Aufschaltung auf den Ausgang des Reglers einer Signalbehandlung (z. B. einer Signalverzögerung) unterziehen.

Der Regler führt einen Aufruf der Funktion durch. Vorhandene Ein-/Ausgangsparameter der Anwender-Funktion werden dabei nicht versorgt. Die Datenübertragung müssen Sie selber in ihrer Funktion programmieren.

Hinweis

Es wird keine Prüfung durchgeführt, ob eine Funktion vorhanden ist. Ist die Funktion nicht vorhanden, dann geht die CPU mit einem internen Systemfehler in STOP.

1. Aktivieren Sie das Optionskästchen "FC-Aufruf aktivieren".
2. Geben Sie die Nummer der Funktion in das Eingabefeld "FC" ein.

Änderungsgeschwindigkeit begrenzen

Rampenfunktionen im Stellgrößenzweig werden verwendet, wenn der Prozess keine sprungförmigen Änderungen des Streckeneingangssignals verträgt. Abrupte Änderungen des Stellwertes sind z. B. zu vermeiden, wenn zwischen geregeltem Motor und anzutreibender Last ein Getriebe zwischengeschaltet ist und ein zu schnelles Ansteigen der Motordrehzahl zu einer Überlastung des Getriebes führt.

Die Änderungsgeschwindigkeit der Stellwerte am Ausgang des Reglers wird getrennt für den An– und Abstieg der Stellgröße begrenzt. Für den gesamten Wertebereich sind, vom Nullpunkt aus gesehen, zwei Rampen mit steigenden oder fallenden Werten parametrierbar. Die Funktion ist eingeschaltet, wenn LMNRC_ON = TRUE.

An den beiden statischen Variablen LMN_URLM und LMN_DRLM werden die jeweiligen Begrenzungen für die Steigungen der Rampenfunktion im positiven und negativen Bereich der Stellgröße eingegeben. Die Steigungen beziehen sich auf einen Anstieg bzw. Abfall in Prozent pro Sekunde. Schnellere Stellwertänderungen werden auf diese Grenzgeschwindigkeiten verzögert.

Wird z. B. 'LMN_URLM' auf 10.0 [%/s] parametriert, so werden in jedem Abtastzyklus die folgenden Werte zum "Altwert" von outv addiert, und zwar solange |inv| > |outv| ist:

Abtastzeit 1 s → $outv_{alt}$ + 10 %
100 ms → $outv_{alt}$ + 1 %
10 ms → $outv_{alt}$ + 0,1 %

Das folgende Bild zeigt die Funktionsweise der Signalbehandlung. Aus Sprungfunktionen am Eingang inv (t) werden Rampenfunktionen am Ausgang outv (t).

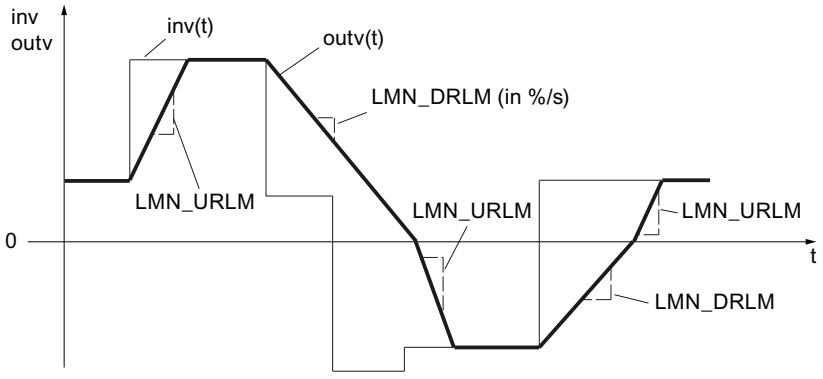

Das Erreichen der Steigungsbegrenzungen wird nicht gemeldet!

Die Kennzeichnung der Rampen–Parameter erfolgt nach folgendem Schema:

Parameter	Rampe
LMN_URLM	|outv| steigend
LMN_DRLM	|outv| fallend

1. Aktivieren Sie das Optionskästchen "Änderungsgeschwindigkeit begrenzen".
2. Geben Sie in das Eingabefeld "Maximaler Anstieg" den maximalen Anstieg des Stellwertes ein.
   Auf diesen Wert werden schnellere Stellwertänderungen verzögert.
3. Geben Sie in das Eingabefeld "Maximaler Abfall" den maximalen Abfall des Stellwertes ein.
   Auf diesen Wert werden schnellere Stellwertänderungen verzögert.

Absolutwert der Stellgröße begrenzen

Der Arbeitsbereich, d. h. der Bereich, in dem sich das Stellglied im Rahmen zulässiger Stellwerte bewegen kann, wird vom Einstellbereich des Stellwertes bestimmt. Da sich, abhängig von der Art des Stellgliedes, die Grenzen für erlaubte Stellwerte meist nicht mit der 0 %– oder der 100 %– Grenze des Stellbereiches decken, können Sie die Ober- und Untergrenze des Stellwerts konfigurieren.

Um im jeweiligen Prozess unzulässige Zustände zu vermeiden, wird im Stellwertzweig der Regelung der Einstellbereich für die Stellgröße nach oben und unten begrenzt.

Die Stellgröße LMN(t) wird auf die vorgebbare Obergrenze (LMN_HLM) und Untergrenze (LMN_LLM) begrenzt. Die Eingangsgröße inv muss jedoch außerhalb dieser Begrenzungen liegen. Da die Funktion nicht abgeschaltet werden kann, muss bei der Konfiguration immer die Angabe einer Unter– und Obergrenze berücksichtigt werden.

Bei Überschreitungen durch die Eingangsgröße inv(t) werden die zugehörigen Anzeigen über die Meldeausgänge ausgegeben.

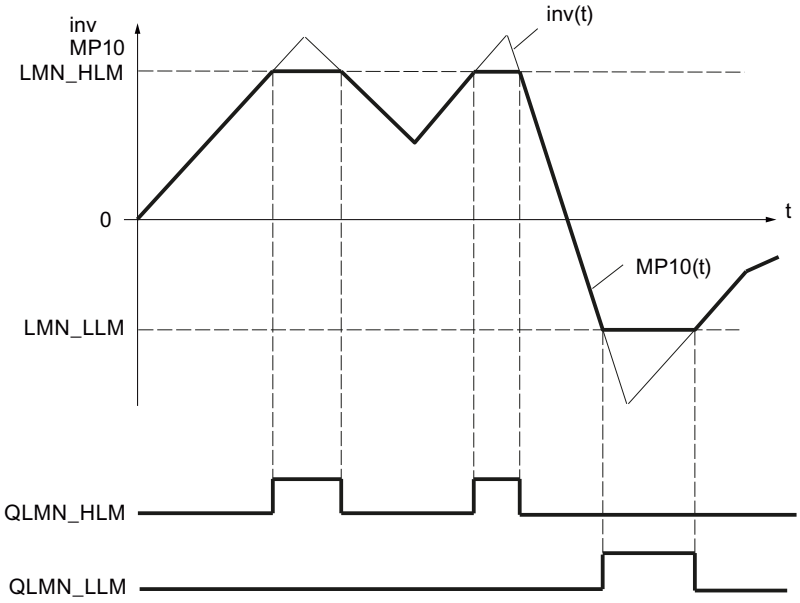

Anlauf und Betriebsweise

- Bei Neustart werden alle Meldeausgänge nullgesetzt.
- Die Begrenzung arbeitet gemäß folgenden Beziehungen:

LMN =	QLMN_HLM	QLMN_LLM	wenn:
LMN_HLM	TRUE	FALSE	inv ≥ LMN_HLM
LMN_LLM	FALSE	TRUE	inv ≤ LMN_LLM
inv	FALSE	FALSE	LMN_HLM < inv < LMN_LLM

Am Ausgang d. h. am Parameter LMN bzw. am Messpunkt MP10 wird der wirksame Stellwert der Regelung angezeigt.

Vorgehen

Um den Absolutwert der Stellgröße zu begrenzen, gehen Sie folgendermaßen vor:

1. Geben Sie in das Eingabefeld "Obergrenze" den Wert für den höchsten zulässigen Wert ein.
2. Geben Sie in das Eingabefeld "Untergrenze" den Wert für den niedrigsten zulässigen Wert ein.

Normierung

Wird am Streckeneingang eine physikalische Größe als Stellgröße gefordert, dann müssen die Gleitpunktwerte im Bereich 0 bis 100 % auf den physikalischen Bereich (z. B. 150 bis 3000 U/min) der Stellgröße denormiert werden.

Zur eindeutigen Festlegung der Normierungsgeraden sind zwei Parameter zu definieren:

- der Faktor für die Steigung (LMN_FAC)
- der Offset der Normierungsgeraden im Nullpunkt (LMN_OFF)

interne Prozentwerte (im REAL–Format) ⇒ externe physikalische Werte

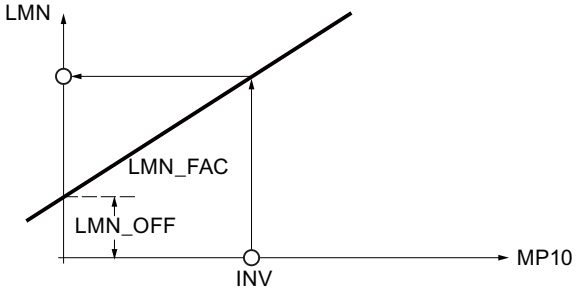

Der Normierungswert wird gemäß folgender Funktion aus dem jeweiligen Eingangswert MP10 berechnet:

LMN = MP10 * LMN_FAC + LMN_OFF

Beispiel

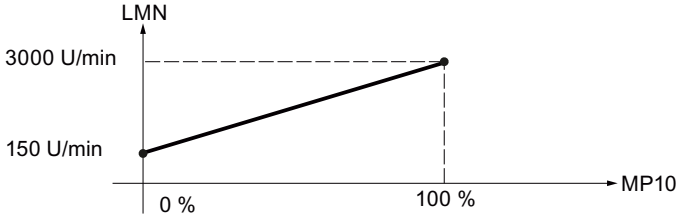

$$\text{LMNFAC} = \frac{(3000 - 150)\text{U/min}}{(100 - 0)\%} = 28{,}5\ \frac{\text{U/min}}{\%}$$

LMNOFF = 150U/min

Funktionsintern werden keine Werte begrenzt, eine Prüfung der Parameter findet nicht statt.

Normierung durchführen

1. Geben Sie in das Eingabefeld "Faktor" die Steigung der Normierungsgeraden ein.
2. Geben Sie in das Eingabefeld "Offset" den Offset der Normierungsgeraden zum Nullpunkt ein.

Pulsgenerator

Anwendung

Der Pulsgenerator transformiert die Eingangsgröße "Stellwert des PID–Reglers am Messpunkt MP10" durch Pulsweitenmodulation in eine Impulsfolge mit konstanter Periodendauer. Die Dauer eines Impulses pro Periodendauer ist proportional der Eingangsgröße.

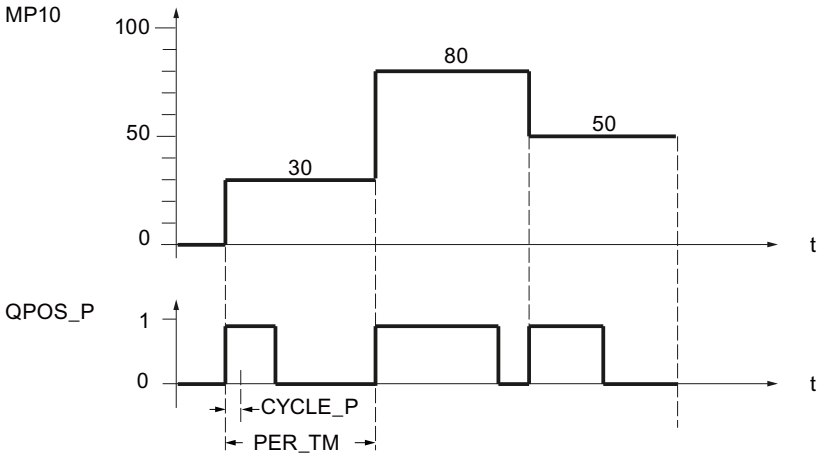

Eine Eingangsgröße von 30 % und 10 Aufrufe des Pulsgenerators pro Periodendauer bedeuten also:

- "Eins" am Ausgang QPOS für die ersten drei Aufrufe (30 % von 10 Aufrufen),
- "Null" am Ausgang QPOS für sieben weitere Aufrufe (70 % von 10 Aufrufen).

Betriebsarten des Reglers mit Impulsausgang

Je nach Parametrierung des Pulsgenerators können PID–Regler mit Dreipunktverhalten oder mit bipolarem bzw. unipolarem Zweipunktausgang konfiguriert werden. Nachstehende Tabelle zeigt die Einstellung der Schalterkombinationen für die möglichen Betriebsarten.

Betriebsart	Schalter		
	MAN_ON	STEP3_ON	ST2BI_ON
Dreipunktregelung	FALSE	TRUE	beliebig
Zweipunktregelung mit bipolarem Stellbereich (–100 % bis 100 %)	FALSE	FALSE	TRUE
Zweipunktregelung mit unipolarem Stellbereich (0 % bis 100 %)	FALSE	FALSE	FALSE
Handbetrieb	TRUE	beliebig	beliebig

Dreipunktregelung

In der Betriebsart "Dreipunktregelung" können drei Zustände des Stellsignals erzeugt werden, z. B. je nach Stellglied und Prozess: Mehr – Aus – Weniger, Vorwärts – Stop – Rückwärts, Heizen – Aus – Kühlen usw. Nach Anforderung des zu regelnden Prozesses werden die Zustandswerte der binären Ausgangssignale QPOS_P und QNEG_P den jeweiligen Betriebszuständen des Stellgliedes zugeordnet. Die Tabelle zeigt zwei Beispiele.

	heizen vorwärts	aus stop	kühlen rückwärts
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

Eine Dimensionierung der Mindestimpulsdauer bzw. Mindestpausendauer P_B_TM_P kann sehr kurze Ein– oder Ausschaltzeiten verhindern, die die Lebensdauer von Schaltgliedern und Stelleinrichtungen beeinträchtigen. Dabei wird der proportionalen Ausgangskennlinie, über die die Impulsdauer berechnet wird, ein Ansprechwert für die Impulsausgabe überlagert.

Hinweis

Kleine Absolutwerte der Eingangsgröße "Stellwert des PID-Reglers am Messpunkt MP10", die eine Impulsdauer kleiner als P_B_TM_P erzeugen würden, werden unterdrückt. Für große Eingangswerte, die eine Impulsdauer größer als PER_TM_P – P_B_TM_P erzeugen würden, wird die Impulsdauer auf 100 % bzw. –100 % gesetzt.

Es werden Einstellwerte P_B_TM_P ≤ 0,1 * PER_TM_P empfohlen.

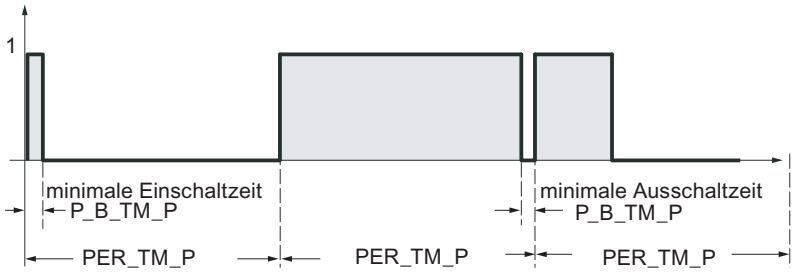

Die Dauer der positiven oder negativen Impulse errechnet sich aus der Eingangsgröße "Stellwert des PID–Reglers am Messpunkt MP10" (in %) mal Periodendauer:

$$\text{Impulsdauer} = \frac{\text{MP10}}{100} * \text{PER_TM_P[s]}$$

Durch die Unterdrückung von Mindestimpulsdauer bzw. Mindestpausendauer erhält die Umformkennlinie Knickpunkte im Anfangsbereich und Endbereich.

Die zuvor getroffenen Aussagen gelten ebenso für P_B_TM_N und PER_TM_N.

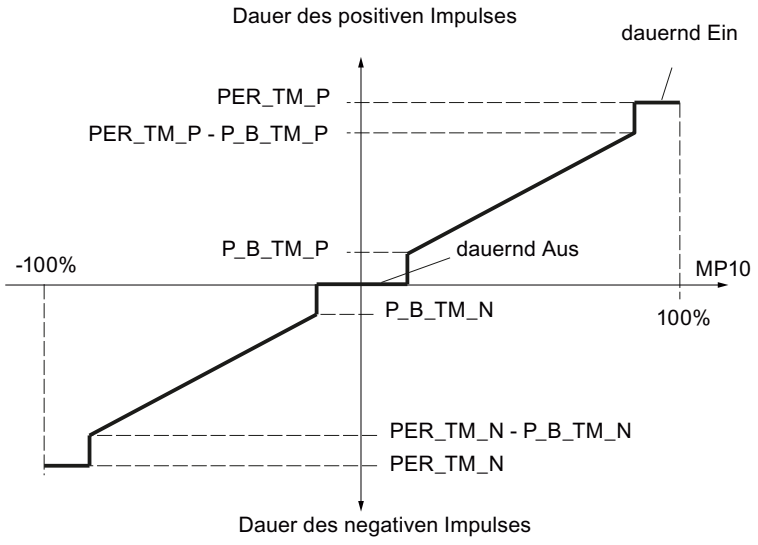

Dreipunktregelung unsymmetrisch

Über den Verhältnisfaktor RATIOFAC kann das Verhältnis der Dauer von positiven zu negativen Impulsen verändert werden. Bei einem thermischen Prozess lassen sich damit z. B. unterschiedliche Streckenzeitkonstanten für Heizen und Kühlen berücksichtigen.

Soll bei gleichem Absolutwert "Stellwert des PID–Reglers am Messpunkt MP10" die Impulsdauer am negativen Impulsausgang gegenüber dem positiven Impuls verkürzt werden, so ist ein Verhältnisfaktor kleiner 1 einzustellen:

pos. Impuls > neg. Impuls: RATIOFAC < 1

Impulsdauer negativ: $\frac{MP10}{100} * PER_TM_N * RATIOFAC$

Impulsdauer positiv: $\frac{MP10}{100} * PER_TM_N$

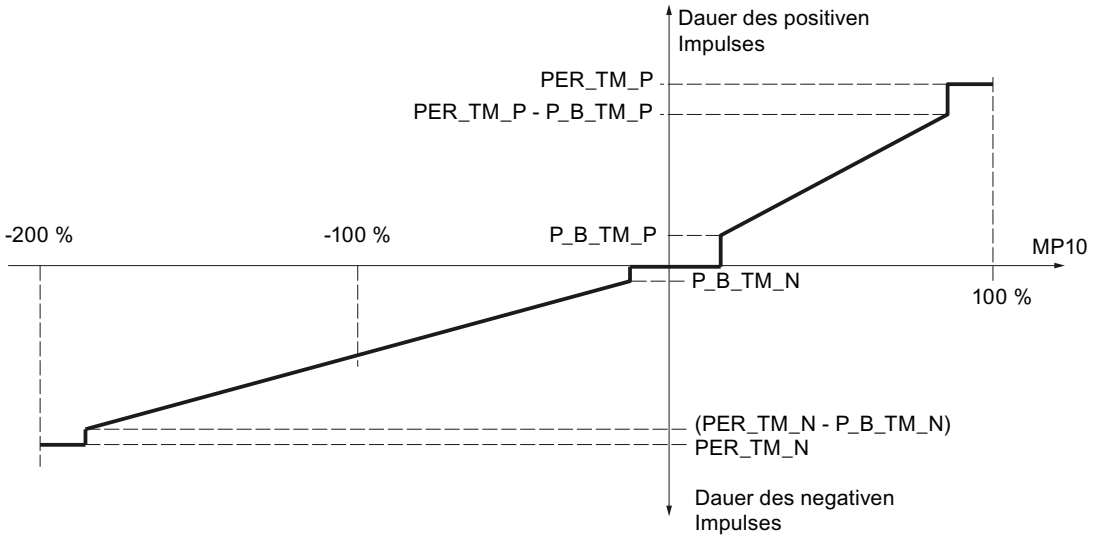

Bild 3-4 Unsymmetrische Kennlinie des Dreipunktreglers (Verhältnisfaktor = 0.5)

Soll umgekehrt bei gleichem Absolutwert |MP10| die Impulsdauer am positiven Impulsausgang gegenüber dem negativen Impuls verkürzt werden, so ist ein Verhältnisfaktor größer 1 einzustellen:

pos. Impuls < neg. Impuls: RATIOFAC > 1

Impulsdauer negativ: $\frac{MP10}{100} * PER_TM_N$

Impulsdauer positiv: $\frac{MP10 * PER_TM_P}{100 * RATIOFAC}$

Rechnerisch bedeutet dies, dass bei RATIOFAC < 1 der Ansprechwert für negative Impulse mit dem Verhältnisfaktor multipliziert und bei RATIOFAC > 1 der Ansprechwert für positive Impulse durch den Verhältnisfaktor dividiert wird.

Hinweis

Bei unsymmetrischer Dreipunktregelung RATIOFAC ≠ 1 müssen Sie die Stellwertgrenzen nach folgenden Formeln anpassen:

RATIOFAC < 1:

LMN_HLM = 100

LMN_LLM = –100 * (1 / RATIOFAC)

RATIOFAC > 1:

LMN_HLM = 100 * RATIOFAC

LMN_LLM = –100

Beispiele:

RATIOFAC	=	1	RATIOFAC	=	0.5	RATIOFAC	=	2.0
LMN_HLM	=	100	LMN_HLM	=	100	LMN_HLM	=	200
LMN_LLM	=	–100	LMN_LLM	=	–200	LMN_LLM	=	–100

Zweipunktregelung

Bei der Zweipunktregelung wird nur der positive Impulsausgang QPOS_P mit dem betreffenden Ein/Aus–Stellglied verbunden. Je nach genutztem Stellbereich, MP10 = -100.0 % bis +100.0 % oder MP10 = 0.0 % bis 100.0 %, hat der Zweipunktregler einen bipolaren oder einen unipolaren Stellbereich.

Bei unipolarer Betriebsart darf MP10 nur Werte zwischen 0.0 % und 100.0 % annehmen.

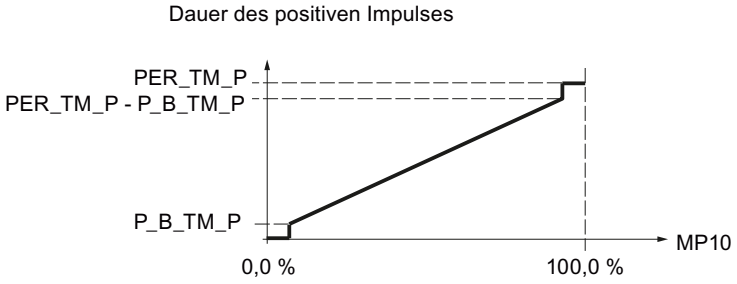

Bei bipolarer Betriebsart darf MP10 Werte zwischen -100.0 % und 100.0 % annehmen.

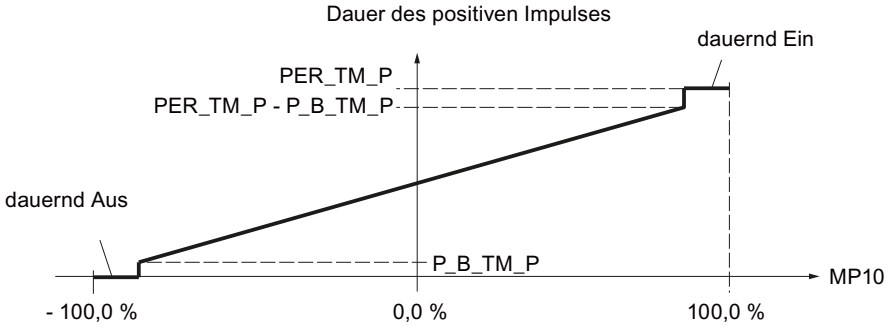

An QNEG_P steht das negierte Ausgangssignal zur Verfügung für den Fall, dass die Verschaltung des Zweipunktreglers im Regelkreis ein logisch invertiertes Binärsignal für die Stellimpulse erfordert.

	Ein	Aus
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Reglerabtastzeit CYCLE und Impulsrasterbreite CYCLE_P

Wenn Sie die Impulsformerstufe eingeschaltet haben (PULSE_ON = TRUE), dann müssen Sie am Eingang CYCLE_P den Zeittakt des aufrufenden Weckalarm–OB vorgeben. Die Dauer des erzeugten Impulses beträgt stets ein ganzzahliges Vielfaches dieses Wertes.

Am Eingang CYCLE geben Sie die Abtastzeit für die übrigen Regelungsfunktionen des PID_CP vor. PID_CP ermittelt die Zeituntersetzung und bearbeitet die Regelungsfunktionen mit der Abtastzeit CYCLE.

CYCLE muss ein ganzzahliges Vielfaches von CYCLE_P ist. Falls Sie diese Bedingung nicht einhalten, rundet PID_CP die Abtastzeit für die Regelungsfunktionen auf ein ganzzahliges Vielfaches von CYCLE_P auf. Die zeitabhängigen Funktionen (z. B. Glättung, Integration, Differentiation) arbeiten dann nicht ganz korrekt.

CYCLE kann kleiner gewählt werden als die Periodendauer PER_TM_P bzw. PER_TM_N. Dies ist dann sinnvoll, wenn einerseits eine möglichst große Periodendauer gewünscht wird, um die Stellglieder zu schonen, und andererseits die Abtastzeit aufgrund einer schnellen Regelstrecke klein sein muss.

Als sinnvoller Wert für die Abtastzeit CYCLE gilt wie beim kontinuierlichen Regler ohne Impulsformerstufe, dass CYCLE nicht kleiner werden darf als ca. 10% der dominierenden Streckenzeitkonstanten des geregelten Systems.

Beispiel für die Wirkung der Parameter CYCLE_P, CYCLE und PER_TM_P bzw. PER_TM_N:

- PER_TM_P = 10 s, CYCLE = 1 s, CYCLE_P = 100 ms.

Jede Sekunde wird ein neuer Stellwert berechnet, alle 100 ms erfolgt der Vergleich des Stellwertes mit der bislang ausgegebenen Impulslänge bzw. Pausenlänge.

Wenn ein Impuls ausgegeben wird und der berechnete Stellwert größer ist als die bisherige Impulslänge / PER_TM_P, wird der Impuls verlängert. Andernfalls wird kein Impulssignal mehr ausgegeben. Wenn kein Impuls ausgegeben wird und (100% - berechneter Stellwert) größer ist als die bisherige Pausenlänge / PER_TM_P, wird die Pause verlängert. Andernfalls wird ein Impulssignal ausgegeben.

Aufgrund eines besonderen Verfahrens der Pulsformung bewirkt eine Vergrößerung bzw. Verkleinerung der Stellgröße während der Periode eine Verlängerung bzw. Verkürzung des ausgegebenen Impulses. Ist in diesem Fall (CYCLE < PER_TM_P) die Periodendauer so groß parametriert, dass sie zum Schwingen des Istwertes führen würde, dann wird die wirksame Periodendauer von PID_CP selbständig auf einen sinnvollen Wert reduziert.

Genauigkeit der Pulsformung

Die Periodendauer PER_TM ist nicht identisch mit dem Bearbeitungszyklus des Pulsgenerators. Eine Periodendauer setzt sich aus mehreren Bearbeitungszyklen des Pulsgenerators zusammen. Die Anzahl der Aufrufe des Pulsgenerators pro Periodendauer sind ein Maß für die Genauigkeit der Impulsweite.

Je kleiner die Abtastzeit CYCLE_P gegenüber der Periodendauer PER_TM_P (bzw. PER_TM_N) ist, desto genauer ist die Pulsweitenmodulation. Für eine hinreichend genaue Regelung sollte folgende Bedingung erfüllt sein:

$$\text{CYCLE_P} \leq \frac{\text{PER_TM}}{20 \ldots 50}$$

Realisierung von sehr kurzen Abtastzeiten

Bei einer schnellen Regelstrecke sind sehr kleine Abtastzeiten (z. B. 10 ms) notwendig. Aufgrund der Programmlaufzeit ist es in diesem Fall nicht sinnvoll, die Regelungsteile im selben Weckalarm–OB zu bearbeiten wie die Berechnung der Impulsausgabe. Die Bearbeitung der Regelungsfunktionen verlagern Sie dann entweder in den OB 1 oder in einen langsameren Weckalarm–OB. Die Bearbeitung der Regelungsfunktion im OB 1 ist nur dann sinnvoll, wenn die Zykluszeit des OB 1 deutlich kleiner ist als die Abtastzeit CYCLE des Reglers.

Über den Parameter SELECT bestimmen Sie, welcher Programmteil bearbeitet werden soll. Die folgende Tabelle verschafft Ihnen einen Überblick über die Parametrierung des Eingangsparameters SELECT:

SELECT	Verwendete Funktionalität des Bausteins	Zugrundeliegende Methode
0	Regelungsteil und Impulsausgabe	Regelungsteil und Impulsausgabe in ein und demselben Baustein
1	Aufruf im OB 1 (Regelungsteil)	Regelungsteil im OB 1, Impulsausgabe in schnellem Weckalarm–OB
2	Aufruf in Weckalarm–OB (Impulsausgabe)	
3	Aufruf in langsamem Weckalarm–OB (Regelungsteil)	Regelungsteil in langsamem Weckalarm–OB, Impulsausgabe in schnellem Weckalarm–OB
2	Aufruf in schnellem Weckalarm–OB (Impulsausgabe)	

Im Folgenden werden die in der obigen Tabelle angedeuteten Methoden zur Realisierung sehr kurzer Impulsrasterbreiten näher erklärt.

- Regelungsfunktion im OB 1, Impulsausgabe in Weckalarm–OB
  Beim Aufruf mit SELECT = 2 erfolgt die Berechnung der Impulsausgabe und die Prüfung, ob seit der letzten Bearbeitung des Regelungsteils die an CYCLE parametrierte Abtastzeit schon abgelaufen ist. Falls diese Abtastzeit abgelaufen ist, beschreibt PID_CP die Variable QC_ACT im Instanz–DB mit dem Wert TRUE.
  Beim Aufruf mit SELECT = 1 erfolgt die Bewertung der Variablen QC_ACT im Instanz–DB: Wenn QC_ACT den Wert FALSE hat, wird der Baustein sofort wieder beendet; er hat also nur eine sehr kleine Laufzeit benötigt. Wenn QC_ACT den Wert TRUE hat, wird der Regelungsteil einmalig durchlaufen, und anschließend QC_ACT wieder zurück gesetzt. Diese Vorgehensweise bewirkt, dass die Abtastzeit für die Regelungsfunktion nicht exakt eingehalten werden kann. Sie schwankt um die Laufzeit des OB 1 (einschließlich aller Unterbrechungen). Dieses Verfahren ist daher nur geeignet, wenn die Laufzeit des OB 1 klein ist gegenüber der Abtastzeit CYCLE.
- Regelungsfunktion in langsamem, Impulsausgabe in schnellem Weckalarm–OB
  Beim Aufruf mit SELECT = 2 erfolgt grundsätzlich die Berechnung der Impulsausgabe
  Beim Aufruf mit SELECT = 3 erfolgt stets die Bearbeitung des Regelungsteils.

Hinweis

Damit der Aufruf der Anweisung PID_CP mit seinen vielen Formaloperanden nicht zweimal ausprogrammiert werden muss, empfiehlt es sich, diesen einmalig in einer FC zu programmieren, die als Eingangsparameter ebenfalls einen Parameter SELECT hat. Diesen Eingangsparameter verschalten sie auf PID_CP.SELECT. Im OB 1 bzw. im Weckalarm-OB rufen Sie dann nur diese FC auf.

Diese Vorgehensweise ist änderungsfreundlich und spart Programmspeicher.

PID_CP in Betrieb nehmen

Manuelle Optimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- MAN_ON = FALSE

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Geben Sie in den Feldern "P", "I", "D" und "Verzögerungszeit" neue PID-Parameter ein.
3. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".

4. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
5. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
   Wenn die Änderungsgeschwindigkeit des Sollwerts begrenzt ist, wird stoßfrei auf den neuen Sollwert umgeschaltet.
6. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.
7. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
8. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

Inbetriebnahme mit dem PID Self Tuner

Die Anweisung PID_CP können Sie mit dem PID Self-Tuner in Betrieb nehmen und die optimalen PID-Parameter ermitteln.

Vorgehen

Um einen kontinuierlichen Regler in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Rufen Sie die Anweisung PID_CP in einem Weckalarm-OB auf und verschalten Sie die Ein- und Ausgangsparameter.
2. Rufen Sie die Anweisung TUN_EC im selben Weckalarm-OB auf.
3. Verschalten Sie die Anweisung TUN_EC folgendermaßen mit PID_CP:
   - TUN_EC.PV mit PID_CP.PV
   - TUN_EC.LMN mit PID_CP.LMN
   - TUN_EC.SP_OUT mit PID_CP.SP_IN oder PID_CP.SP_EXT
   - TUN_EC.GAIN mit PID_CP.GAIN
   - TUN_EC.TI mit PID_CP.TI
   - TUN_EC.TD mit PID_CP.TD
   - TUN_EC.TM_LAG mit PID_CP.TM_LAG
   - Wenn Sie einen Dreipunktregler einsetzen, TUN_EC.RATIOFAC mit PID_CP.RATIOFAC
   - TUN_EC.QMAN_ON mit PID_CP.MAN_ON
4. Klicken Sie auf das Symbol an der Anweisung TUN_EC.
   Der Inbetriebnahmeeditor für TUN_EC wird geöffnet. Die Bedienung des Inbetriebnahmeeditors ist beim PID Self-Tuner beschrieben.

3.5.1.2 PID_ES einsetzen

Technologieobjekt PID_ES

Das Technologieobjekt PID_ES stellt einen Schrittregler für Automatik und Handbetrieb zur Verfügung. Es entspricht dem Instanz-Datenbaustein der Anweisung PID_ES. Sie können die folgenden Regler konfigurieren:

- Festwertregelung mit P–, PI–, PD–, PID–Schrittegler
- Schrittregler mit und ohne Stellungsrückmeldung
- Festwertregelung mit Störgrößenaufschaltung
- Kaskadenregelung mit Schrittregler im Folgekreis
- Zweischleifige Verhältnisregelung
- Mischungsregelung

Sie konfigurieren die folgenden Funktionen des Technologieobjekts PID_ES:

- Normierung des Istwertsignals
- Unterdrückung von höherfrequenten Schwingungen im Istwertzweig durch Glättung (Verzögerung) des Istwertsignals
- Linearisierung von quadratischen Funktionen des Istwerts (Durchflussregelung mit Differenzdruck–Gebern)
- Aufruf von "eigenen Funktionen" im Istwertzweig
- Überwachung des Istwerts auf Überschreitung von Alarm- und Warngrenzen
- Überwachung der Änderungsgeschwindigkeit des Istwerts
- Verstellung des Sollwertes durch Zeitplangeber
- Normierung eines externen Sollwertsignals bei Verhältnisregelungen
- Aufruf von "eigenen Funktionen" im Sollwertzweig
- Begrenzung der Änderungsgeschwindigkeit des Sollwerts
- Begrenzung der Absolutwerte des Sollwerts
- Filterung des Regeldifferenzsignals durch eine Totzone
- Überwachung der Regeldifferenz auf Überschreitung von Alarm- und Warngrenzen
- Einzeln zuschaltbaren P-, I- und D-Anteil
- Wahlweise P– und D–Wirkung in der Rückführung des Reglers.
- Aufschalten einer Störgröße
- Aufschalten der Stellungsrückmeldung
- Steuerung des Stellwerts im Handbetrieb von einem PG oder HMI aus)
- Begrenzung der Absolutwerte des Stellwerts
- Berücksichtigung stellgliedspezifischer Zeiten beim Erzeugen der Impulse

Regeldifferenzbildung

Istwert

Normierung (PV_NORM)

Über das Optionskästen "Peripherie einschalten" legen Sie fest, welche analoge Eingangsgröße auf die physikalische Einheit des Istwerts normiert wird.

- Aktiviert: Eingangsgröße ist der Istwert Peripherie (PV_PER)
- Deaktiviert: Eingangsgröße ist der interne Istwert (PV_IN)

Zur eindeutigen Festlegung der Normierungsgeraden definieren Sie die folgenden vier Parameter:

- Eingang oben (NM_PIHR)
- Eingang unten (NM_PILR)
- Ausgang oben (NM_PVHR)
- Ausgang unten (NM_PVLR)

Je nach Sensortyp werden die Parameter vorbelegt.

Der Istwert wird nach folgender Formel aus dem jeweiligen Eingangswert PV_PER bzw. PV_IN berechnet:

MP4 = (PV_PER - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

bzw.

MP4 = (PV_IN - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

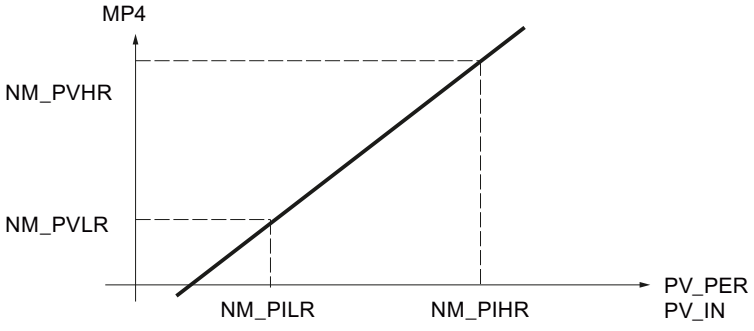

Der Istwert wird in der statischen Variablen MP4 ausgegeben.

Es werden keine Werte begrenzt, und eine Prüfung der Parameter findet nicht statt. Falls Sie für Eingang oben und Eingang unten denselben Wert eingeben, kann nach den obigen Formeln eine Division durch Null auftreten. Dies wird in der Anweisung nicht abgefangen!

Filterfunktionen

Regelgröße glätten (LAG1ST)

Zur Glättung von Prozesssignalen mit überlagerten schnellen Schwingungen (Noise) wird ein Verzögerungsglied erster Ordnung im Istwertzweig eingesetzt. Diese Funktion dämpft die analoge Regelgröße je nach eingestellter Zeitkonstante (PV_TMLAG) mehr oder weniger stark. Störsignale werden so wirksam unterdrückt. Insgesamt wird dadurch jedoch auch die Zeitkonstante des Gesamtregelkreises erhöht, d. h. die Regelung wird träger.

Die Dämpfungswirkung wird durch einen Verzögerungsalgorithmus 1. Ordnung erzielt.

Die Sprungantwort im Zeit–Bereich ist:

$outv(t) = MP4(0)\ (1 - e^{-t/PV_TMLAG})$

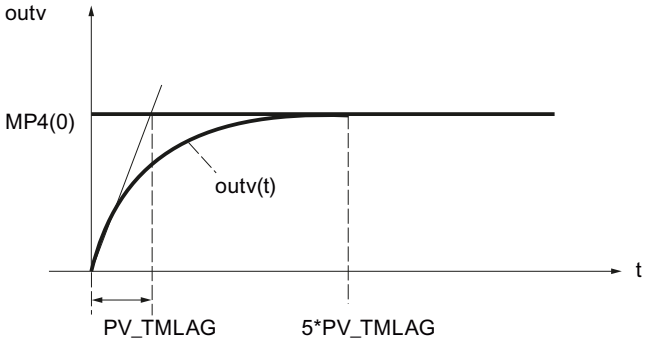

MP4(0)	die Höhe des Istwertsprungs am Eingang
outv(t)	die Ausgangsgröße
PV_TMLAG	die Verzögerungszeitkonstante
t	Zeit

Glättung des Istwerts

Wenn die Glättung des Istwerts (LAG1STON = FALSE) nicht aktiviert ist, dann ist der Peripherieeingang (PV_PER) bzw. der interne Eingang (PV_IN) verzögerungsfrei auf den Istwertzweig durchgeschaltet.

Verzögerungszeit

Der Eingangsparameter PV_TMLAG legt die Verzögerungszeit für den Istwert fest. Der gesamte Wertebereich ist zulässig.

Für PV_TMLAG ≤ 0.5 * CYCLE ist keine Verzögerung mehr wirksam.

Eine Abtastzeit (CYCLE) von weniger als einem Fünftel der Verzögerungszeit ist erforderlich, um ein dem analogen Verhalten annähernd entsprechendes Verzögerungsverhalten zu erzielen.

Quadratwurzel bilden (SQRT)

Die Funktion "Bildung der Quadratwurzel" (SQRT) dient der Linearisierung der Regelgröße. Liegt der von einem Geber gelieferte Istwert als physikalische Größe vor, die in einem quadratischen Zusammenhang mit der gemessenen Prozessgröße steht, dann muss der Verlauf der Regelgröße vor der Weiterverarbeitung im Regler linearisiert werden.

Das Messsignal muss immer dann durch Radizieren linearisiert werden, wenn Durchflussmessungen mithilfe von Messblenden oder Venturirohren usw. vorgenommen werden. Der gemessene Differenzdruck (Wirkdruck) ist dann proportional zum Quadrat des Durchflusses.

Über das Eingangssignal SQRT_ON = TRUE wird die Wurzelfunktion im Istwertzweig aktiviert. Der Algorithmus der Wurzelfunktion hat folgende Form:

outv = SQRT (MP5) × (SQRT_HR – SQRT_LR) / 10.0 + SQRT_LR

Dabei ist outv der normierte Ausgangswert der Wurzelfunktion (physikalischer Durchfluss).

Diese Formel sieht vor, dass der Eingangswert der Wurzel auf einen Zahlenbereich von 0 bis 100 normiert ist. In der Normierung des Istwerts müssen der "Messbereich oben" und "Messbereich unten" auf 100.0 und 0.0 parametriert werden. Die Wurzel aus diesem Wert ergibt einen Messbereich von 0 bis 10. Über die obere Grenze (SQRT_HR) und die untere Grenze des physikalischen Messbereichs (SQRT_LR) wird dieser Zahlenbereich normiert.

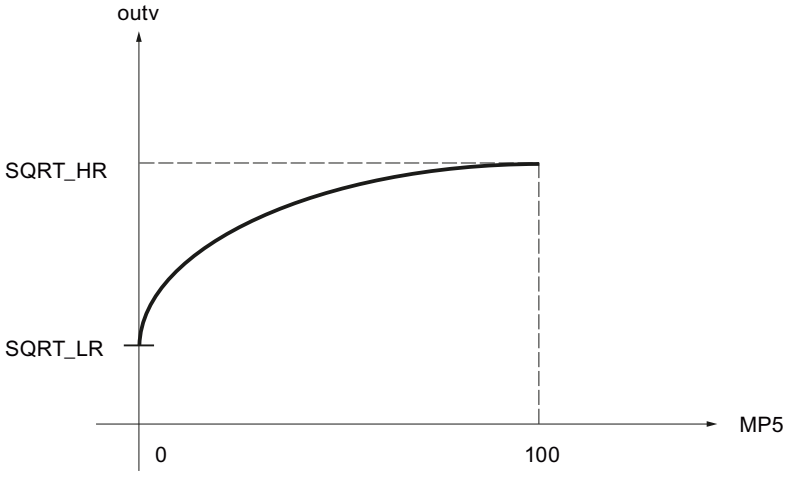

Beispiel zur Normierung

Der Eingangswert PV_IN des Reglers sei der Differenzdruck in mbar:

Messbereichs–Anfang NM_PILR	Messbereichs–Ende NM_PIHR	Wertebeispiel für PV_IN
20.0 mbar	200.0 mbar	150.0 mbar

Mit Hilfe der Normierfunktion PV_NORM erfolgt die Berechnung des normierten Differenzdrucks, wobei NM_PVHR = 100.0 und NM_PVLR = 0.0 gewählt wird:

MP4 = (PV_IN - NM_PILR) * (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR
= (PV_IN - 20.0 mbar) * (100.0 - 0.0) / (200.0 mbar - 20.0 mbar) + 0.0
= (PV_IN - 20.0 mbar) * 100 / 180.0 mbar

Anfangswert von MP4	Endwert von MP4	Wertebeispiel für MP4
0.0	100.0	72.222

In diesem Beispiel wird keine Glättung verwendet und daher gilt: MP5 = MP4.

Für die Wurzel aus dem normierten Differenzdruck MP5 ergibt sich:

Anfangswert nach der Wurzel	Endwert nach der Wurzel	Wertebeispiel für SQRT(MP5)
0.0	10.0	8.498

Für den normierten Ausgangswert outv der Wurzelfunktion (physikalischer Durchfluss) folgt mit SQRT_HR = 20000.0 m^3/h und SQRT_LR = 0.0 m3/h:

outv = SQRT(MP5) * (SQRT_HR - SQRT_LR) / 10.0 + SQRT_LR
= SQRT(MP5) * (20000.0 m^3/h - 0.0 m3/h) / 10.0 + 0.0 m3/h
= 2000.0 m^3/h * SQRT(MP5)

Messbereichs–Anfang outv	Messbereichs–Ende outv	Wertebeispiel für outv
0.0 m^3/h	20000.0 m^3/h	16996.732 m^3/h

FC-Aufruf im Istwertzweig (PVFC)

Durch Einfügen einer anwenderspezifischen Funktion in den Istwertzweig kann die Regelgröße vor der Weiterverarbeitung im Regler einer Signalbehandlung, z. B. einer Signalverzögerung oder Linearisierung, unterzogen werden.

Funktion aufrufen

Wenn Sie eine anwenderspezifische Funktion in den Istwertzweig einfügen möchten, aktivieren Sie "FC Aufruf aktivieren" (PVFC_ON). Im Eingabefeld "FC Nummer" (PVFC_NBR) wird die Nummer der verwendeten Funktion eingegeben. Der Regler führt einen Aufruf der Funktion durch. Vorhandene Ein-/Ausgangsparameter der Anwender-Funktion werden dabei nicht versorgt. Die Datenübertragung müssen Sie selber in ihrer Funktion programmieren.

Den Ausgangswert der Funktion können Sie an der statischen Variable MP6 beobachten.

Hinweis

Eine Prüfung, ob eine Funktion vorhanden ist, wird nicht durchgeführt. Ist die Funktion nicht vorhanden, dann geht die CPU mit einem internen Systemfehler in STOP.

Alarm

Istwert auf Grenzwerte überwachen (PV_ALARM)

Werden bestimmte Werte der Prozessgrößen (z. B. Drehzahl, Druck, Füllstand, Temperatur, ...) über– oder unterschritten, können bei Regelungen unzulässige Prozesszustände oder Anlagenzustände auftreten. In diesen Fällen wird die Überwachung des Istwertes auf Grenzwerte (PV_ALARM) eingesetzt, um den Istwert auf Über–/ Unterschreitung des zulässigen Betriebsbereiches zu überwachen. Die Grenzüberschreitungen werden erfasst und gemeldet, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung des Istwerts auf Grenzwerte kann nicht ausgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Überwachung des Istwertes auf Grenzwerte (PV_ALARM) überwacht die Regelgröße PV(t) auf vier vorgebbare Grenzen in zwei Toleranzbändern. Werden die Grenzen erreicht oder überschritten, meldet die Funktion an der ersten Grenze "Warnung" und an der zweiten Grenze "Alarm".

Grenzwerte

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für "Warnung" und "Alarm" eingestellt. Überschreitet bzw. unterschreitet der Istwert (PV) diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits für den oberen Alarm (QPVH_ALM), die obere Warnung (QPVH_WRN), die untere Warnung (QPVL_WRN) und den unteren Alarm (QPVL_ALM) gesetzt.

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

QPVH_ALM	QPVH_WRN	QPVL_WRN	QPVL_ALM	wenn:	und:
TRUE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_ALM PV ≥ PVH_ALM - PV_HYS
FALSE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_WRN PV ≥ PVH_WRN - PV_HYS

QPVH_ALM	QPVH_WRN	QPVL_WRN	QPVL_ALM	wenn:	und:
FALSE	FALSE	TRUE	FALSE	PV ↘ PV ↗	PV ≤ PVL_WRN PV ≤ PVL_WRN + PV_HYS
FALSE	FALSE	TRUE	TRUE	PV ↘ PV ↗	PV ≤ PVL_ALM PV ≤ PVL_ALM + PV_HYS

Für ein sinnvolles Arbeiten des Bausteins muss gelten:

Oberer Alarm < Obere Warnung < Untere Warnung < Unterer Alarm

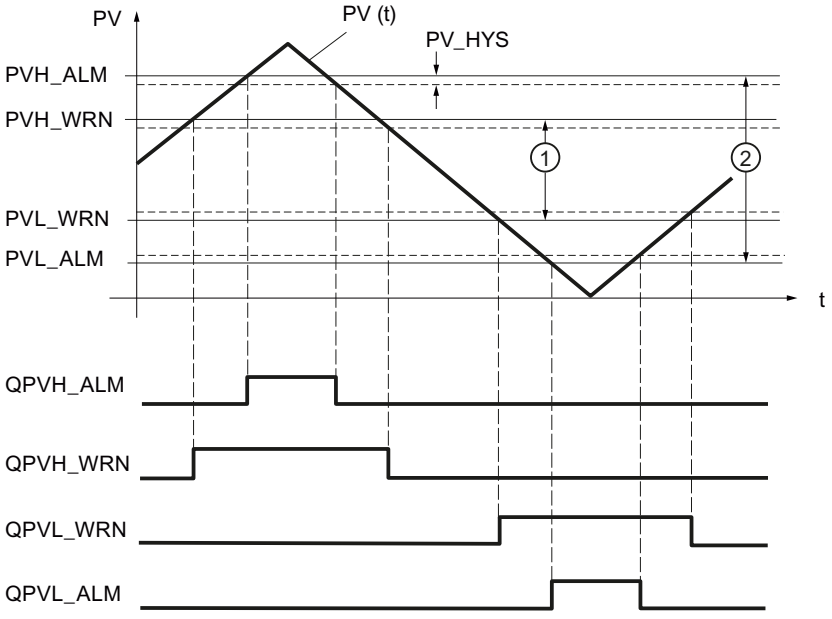

①	1. Toleranzband
②	2. Toleranzband

Hysterese

Um bei geringfügigen Änderungen der Eingangsgröße oder bei Rundungsfehlern ein "Flackern" der Meldebits zu verhindern, wird eine Hysterese (Rückschaltdifferenz) PV_HYS eingestellt. Die Regelgröße muss die Hysterese überwinden, bevor die Meldebits rückgesetzt werden.

Änderungsgeschwindigkeit des Istwertes überwachen (ROCALARM)

Wird die Änderung der Prozessgröße (z. B. Drehzahl, Druck, Füllstand, Temperatur, ...) zu groß, können bei Regelungen unzulässige Prozesszustände oder unzulässige Anlagenzustände auftreten. In diesen Fällen wird die Überwachung der Änderungsgeschwindigkeit des Istwertes (ROCALARM) eingesetzt, um den Istwert auf Überschreiten bzw. Unterschreiten einer zulässigen Änderungsgeschwindigkeit zu überwachen. Die Grenzüberschreitungen werden erfasst und gemeldet, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung der Änderungsgeschwindigkeit des Istwertes (ROCALARM) kann nicht abgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Funktion ROCALARM überwacht die Regelgröße PV(t) auf je nach Vorzeichen unterschiedlich vorgebbare Grenzen für die zulässige Änderungsgeschwindigkeit.

Die Zahlenwerte der Grenzsteigungen werden an den Eingangsparametern für "Maximaler Anstieg" und "Maximaler Abstieg" im positiven und negativen Bereich der Regelgröße eingestellt. Die Steigungen beziehen sich auf einen Anstieg bzw. Abfall in Prozent pro Sekunde. Überschreitet die Änderungsgeschwindigkeit der Regelgröße diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits QPVURLMP ... QPVDRLMN gesetzt.

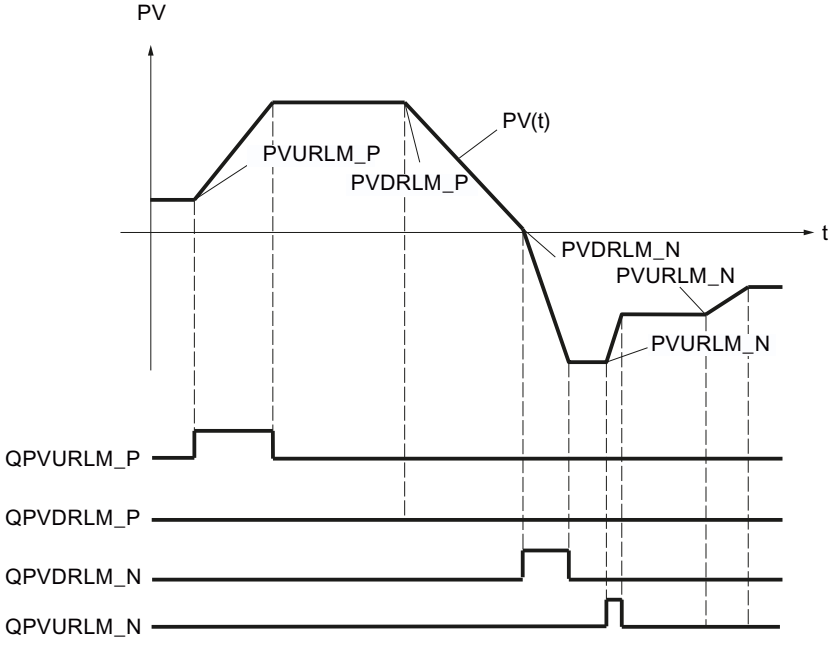

Die Kennzeichnung der Rampen–Parameter erfolgt nach folgendem Schema:

Parameter	PV–Änderung
Maximaler Anstieg im positiven Bereich (PVURLM_P)	PV > 0 und |PV| steigend
Maximaler Abstieg im positiven Bereich (PVDRLM_P)	PV > 0 und |PV| fallend

Parameter	PV–Änderung
Maximaler Anstieg im negativen Bereich (PVURLM_N)	PV < 0 und |PV| steigend
Maximaler Abstieg im negativen Bereich (PVDRLM_N)	PV < 0 und |PV| fallend

Sollwert

Sollwert festlegen

Wählen Sie eine Quelle für den Sollwert aus. Folgende Optionen stehen zur Auswahl:

- Interner Sollwert
  Der Eingangsparameter SP_INT wird als Sollwert verwendet
- Sollwertgenerator
  Der Eingangsparameter SP_INT wird als Sollwert verwendet und kann über die statischen Variablen SPUP und SPDN vergrößert und verkleinert werden.
- Zeitplangeber
  Der Eingangsparameter SP_INT wird als Sollwert verwendet und kann nach einem vorgegebenen Zeitplan verändert werden.
- Externer Sollwert
  Bei Mischungs- und Verhältnisreglern sowie Kaskadenreglern wird der Eingangsparameter SP_EXT als Sollwert verwendet. Der externe Sollwert kann normiert werden.

Am Ausgangsparameter SP wird der wirksame Sollwert angezeigt.

Siehe auch

Zeitplangeber (RMP_SOAK) (Seite 893)

Sollwert–Generator (SP_GEN)

Der interne Sollwert SP_INT kann über die Schalter SPUP und SPDN schrittweise vergrößert oder verkleinert werden. Der modifizierte Wert ist an MP1 zu beobachten. Die konfigurierten Begrenzungen des Sollwerts werden berücksichtigt.

Wirkung der Schalter SPUP und SPDN

Für eine feinstufige Änderung sollte der Regler eine Abtastzeit von maximal 100 ms haben.

Die Änderungsgeschwindigkeit der Ausgangsgröße hängt von der Betätigungsdauer der Schalter SPUP bzw. SPDN und von den eingestellten Begrenzungen nach folgender Beziehung ab:

- während der ersten 3 s nach Setzen von SPUP oder SPDN gilt:
  doutv/dt = (SP_HLM - SP_LLM) / 100 s
- danach
  doutv/dt = (SP_HLM - SP_LLM) / 10 s

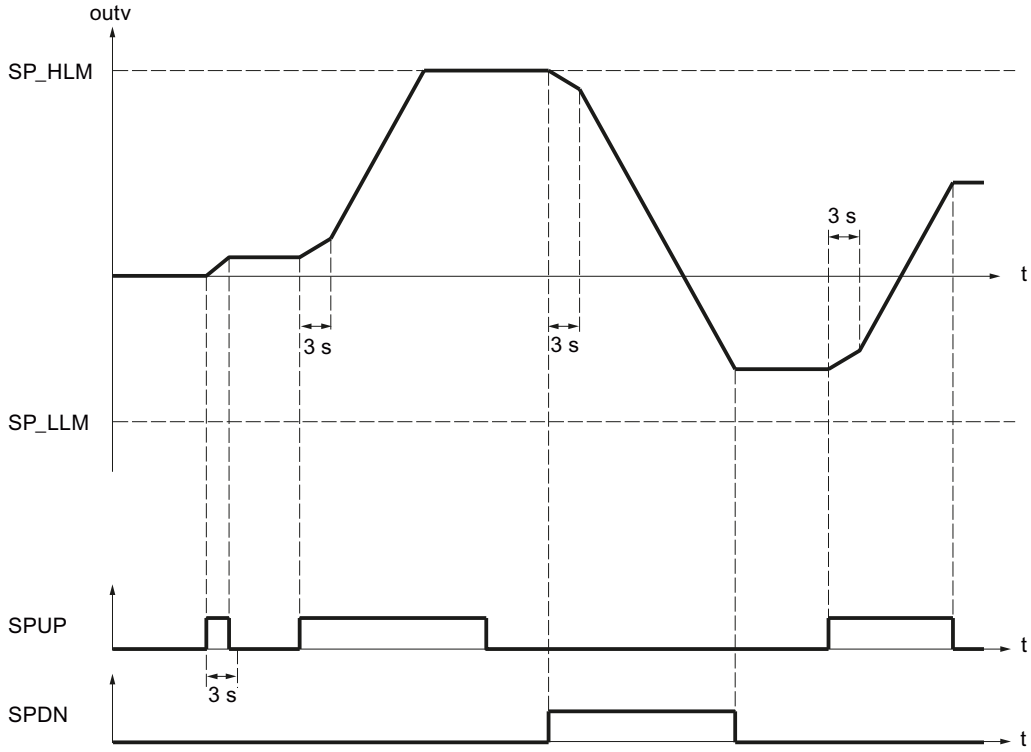

Bei einer Abtastzeit von 100 ms und einem Sollwertbereich von -100,0 bis 100,0 ändert sich z. B. der Sollwert in den ersten drei Sekunden um 0,2 pro Zyklus. Bleibt SPUP länger als 3 Sekunden TRUE, steigt die Änderungsgeschwindigkeit auf den zehnfachen Wert, hier also auf 2 pro Zyklus.

Anlauf– und Betriebsweise des Sollwertgenerators

- Bei Neustart wird der Ausgang outv auf 0.0 zurückgesetzt.
- Schalten Sie den Sollwertgenerator ein (SPGEN_ON = TRUE), dann wird am Ausgang outv zunächst der Signalwert SPFC_IN ausgegeben. Das Umschalten zum Sollwertgenerator aus einer anderen Betriebsart ist deshalb immer stoßfrei. Solange die Schalter SPUP und SPDN (Auf–/Ab–Tasten) nicht aktiviert werden, bleibt SPFC_IN am Ausgang stehen.

Parameter der Funktion SP_GEN

Der Ausgangsparameter outv ist ein impliziter Parameter. Er ist am Konfigurationswerkzeug über den Messpunkt MP1 zu beobachten.

Parameter	Bedeutung	Zulässiger Wertebereich
SPFC_IN	Sollwert FC Eingang	technischer Wertebereich
SP_INT	interner Sollwert	technischer Wertebereich

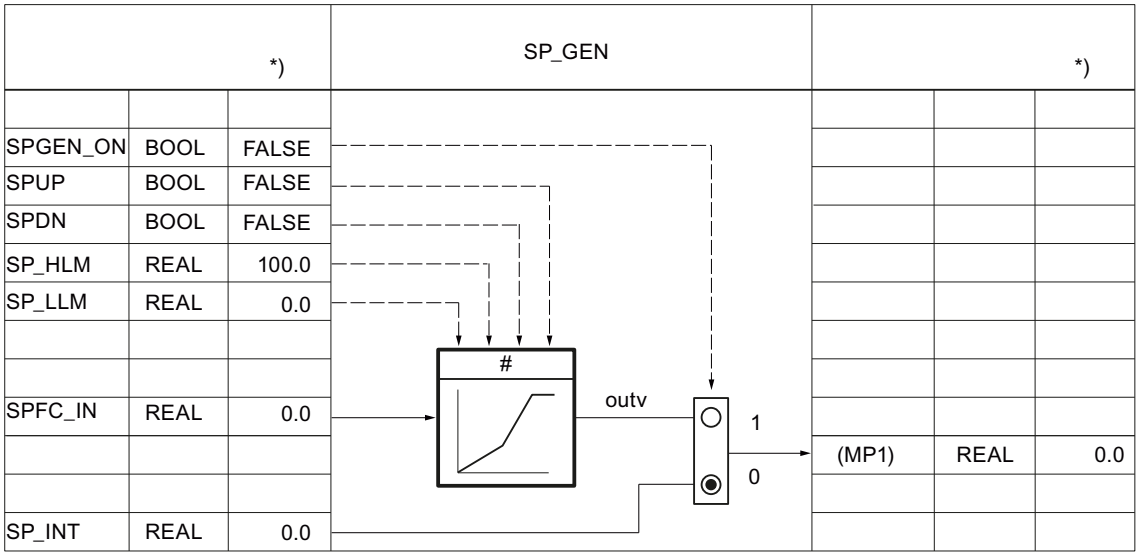

Bild 3-5 Funktionsschema und Parameter des Sollwert–Generators

Zeitplangeber (RMP_SOAK)

Falls der Sollwert SP_INT selbsttätig zeitabhängig geändert werden soll, z. B. bei der Regelung von Prozessen nach einem zeitgesteuerten Temperaturprogramm, so ist dies durch Projektieren eines Zeitplans und Aktivieren des Zeitplangebers möglich. Die Sollwertänderung wird aus Geradenabschnitten mit max. 256 Stützpunkten gebildet.

Hinweis

Die Bausteinfunktion überprüft nicht, ob ein globaler DB mit der Nummer ’DB_NBR’ vorhanden ist und ob der Parameter 'Anzahl der Stützpunkte' NBR_PTS zur DB-Länge passt. Bei inkorrekter Parametrierung geht die CPU mit einem internen Systemfehler in **STOP**.

Arbeitsweise des Zeitplangebers

Der Zeitplangeber liefert eine nach einem definierten Zeitplan ablaufende Funktion der Ausgangsgröße outv. Der Ablauf des Zeitplans wird durch Festlegen einer Reihe von Stützpunkten in einem globalen Datenbaustein DB_RMPSK mit den Zeitwerten PI[i].TMV und den zugehörigen Ausgangswerten PI[i].OUTV definiert.

- PID-CP
  AUTOHOTSPOT ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.
- PID-ES
  AUTOHOTSPOT ist nicht in der Bibliothek enthalten. Sie müssen diesen globalen Datenbaustein selbst erstellen.

Das folgende Bild zeigt einen Zeitplan mit 6 Stützpunkten.

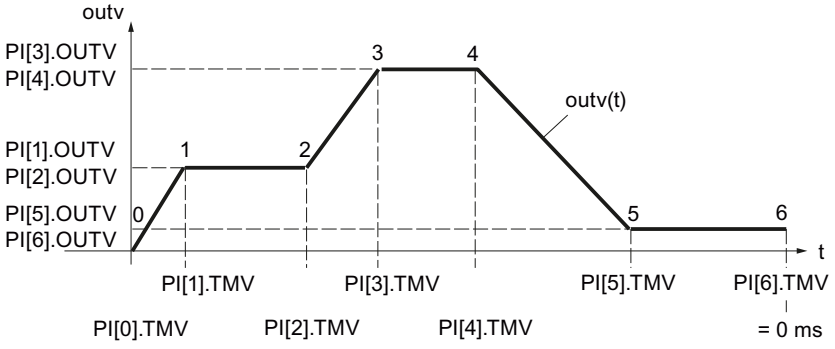

Für den letzten Stützpunkt n ist der Zeitwert PI[n].TMV = 0 ms. Die Bearbeitungszeit eines Zeitplans wird ausgehend vom Startwert bis auf Null heruntergerechnet.

PI[i].TMV gibt die Zeitabstände von Stützpunkt zu Stützpunkt an. Zwischen den Stützpunkten wird nach folgender Funktion interpoliert:

$$outv(t) = PI[n + 1].\ OUTV - \frac{RS_TM}{PI[n].TMV}\ (PI[n + 1].OUTV - PI[n].OUTV)$$

$0 \leq n \leq (NBR_PTS - 1)$

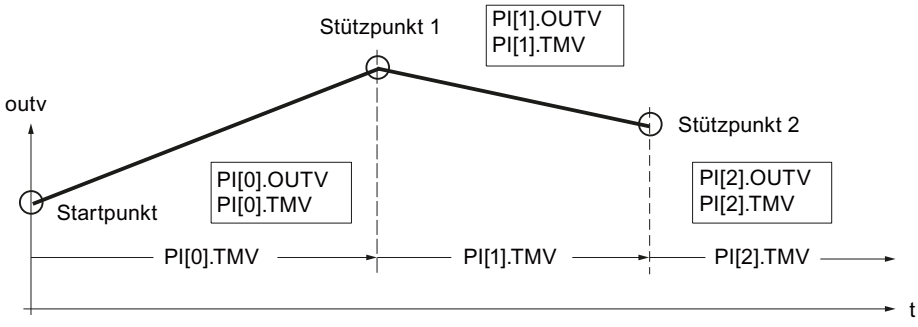

Hinweis

Hinweis

Beim Interpolieren der Fahrkurve zwischen den Stützpunkten kann es zum zeitweisen Verharren des Ausgangswerts kommen, wenn die Abtastzeit CYCLE sehr klein gegenüber der Zeit zwischen den Stützpunkten PI[n].TMV ist. Der Zeitplangeber kann aufgrund der CPU-Rechengenauigkeit nicht beliebig flache Fahrkurven linearisieren. Ist die Fahrkurve zu flach, verharrt der Ausgangswert eine Zeit lang am jeweiligen Stützpunkt, um nach einer gewissen Zeit mit der minimalen Steilheit zum nächsten Stützpunkt zu integrieren.

Abhilfe: Verkürzen Sie die Zeit zwischen den Stützpunkten, indem Sie weitere Stützpunkte einfügen. Sie nähern damit die ausgegebene Fahrkurve der gewünschten flachen Fahrkurve trapezförmig an.

Zeitplangeber einschalten

Mit dem Wechsel RMPSK_ON von FALSE nach TRUE wird der Zeitplangeber eingeschaltet. Nach Erreichen des letzten Stützpunktes ist der Zeitplan beendet. Bei erneutem Start der Funktion durch den Bediener muss RMPSK_ON zunächst auf 'FALSE' und dann wieder auf 'TRUE' gesetzt werden.

Bei **Neustart** wird der Ausgang outv auf 0.0 zurückgesetzt und die Gesamtzeit bzw. Gesamtrestzeit ermittelt. Beim Übergang in den Normalbetrieb wird sofort die Fahrkurve vom Startpunkt aus nach eingestellter Betriebsart bearbeitet. Wird dies nicht gewünscht, muss der Parameter RMPSK_ON im Neustart–OB auf FALSE gesetzt werden.

Funktionen des Zeitplangebers

Die Anweisung verfügt über folgende Funktionen:

- Zeitplangeber einschalten für einmaligen Durchlauf
- Ausgang mit festem Wert vorbelegen
- Zyklische Wiederholung einschalten
- Zeitplangeber anhalten
- Zeitplangeber fortsetzen
- Gesamtbearbeitungszeit und Gesamtrestzeit aktualisieren

Zur Einstellung einer gewünschten Betriebsart gilt für die Wertigkeit der Steuereingänge folgende Tabelle:

Betriebsart	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Ausgangssignal OUTV
Zeitplangeber ein	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) Endwert wird nach Bearbeitungsende gehalten.
Ausgang vorgeben	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV

Betriebsart	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Ausgangssignal OUTV
Zyklische Wiederholung ein	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) nach Ende: automatischer Start
Zeitplangeber anhalten	TRUE	FALSE	TRUE	FALSE	**)	**)	aktueller Wert von OUTV(t) wird gehalten *)
Zeitplangeber fortsetzen	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (alt) *)
			FALSE				Es wird mit neu vorgegebenen Werten weitergefahren.
Gesamtzeit und Gesamtrestzeit aktualisieren	**)	**)	**)	**)	**)	FALSE	beeinflusst OUTV nicht
						TRUE	beeinflusst OUTV nicht

*) Bis zum nächsten Stützpunkt hat die Fahrkurve nicht die vom Anwender parametrierte Form!

**) Die jeweils eingestellte Betriebsart wird unabhängig von der Wertigkeit der Steuersignale in den mit **) gekennzeichneten Feldern ausgeführt

Ausgang des Zeitplangebers vorgeben

Wenn Sie diese Funktion aktivieren (DFRMP_ON = TRUE), dann wird der Ausgangswert des Zeitplangebers auf den Signalwert SP_INT bzw. den Ausgangswert SP_GEN gesetzt.

Hinweis

Die Vorbelegung mit dem internen Sollwert wirkt sich nur dann aus, wenn der Zeitplangeber eingeschaltet ist (RMPSK_ON = TRUE).

Nach dem Umschalten DFRMP_ON = FALSE wird outv linear vom eingestellten Sollwert (z. B. SP_INT) zum Ausgangswert der aktuellen Stützpunktnummer PI[NBR_ATMS].OUTV gefahren.

Die interne Zeitbearbeitung läuft auch bei durchgeschaltetem festem Sollwert (RMPSK_ON = TRUE und DFRMP_ON = TRUE) weiter.

Beim Start der Fahrkurve RMPSK_ON = TRUE wird so lange der feste Sollwert SP_INT ausgegeben, bis DFRMP_ON nach der Dauer von T* von TRUE nach FALSE wechselt.

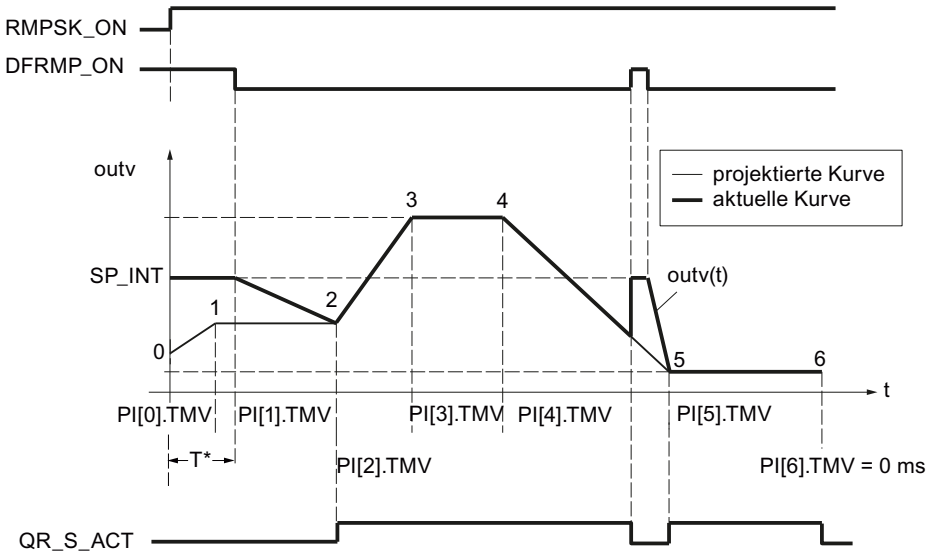

Zu diesem Zeitpunkt ist die Zeit PI[0].TMV und ein Teil der Zeit PI[1].TMV abgelaufen. OUTV wird von SP_INT nach PI[2].OUTV gefahren, d.h. zum Sollwert am Stützpunkt 2.

Erst ab dem Stützpunkt 2 wird die projektierte Fahrkurve ausgegeben. Die Übereinstimmung von projektierter und aktueller Fahrkurve wird angezeigt, indem QR_S_ACT = TRUE wird. Bei einer steigenden Flanke an DFRMP_ON während der Fahrkurvenbearbeitung springt der ausgegebene Wert OUTV ohne Verzögerung auf SP_INT.

Zyklische Wiederholung einschalten

Ist die Betriebsart 'Zyklische Wiederholung' (CYC_ON = TRUE) eingeschaltet, dann kehrt der Zeitplangeber nach Ausgabe des letzten Stützpunktwertes automatisch zum Startpunkt zurück und beginnt einen neuen Durchlauf.

Zwischen dem letzten Stützpunkt und dem Startpunkt wird nicht interpoliert. Für stoßfreien Übergang muss gelten: PI[NBR_PTS].OUTV = PI[0].OUTV.

Zeitplangeber anhalten

Mit RMP_HOLD = TRUE wird der Wert der Ausgangsgröße (einschließlich der Zeitbearbeitung) eingefroren. Bei Rücksetzen RMP_HOLD = FALSE wird an der Unterbrechungsstelle PI[x].TMV fortgefahren.

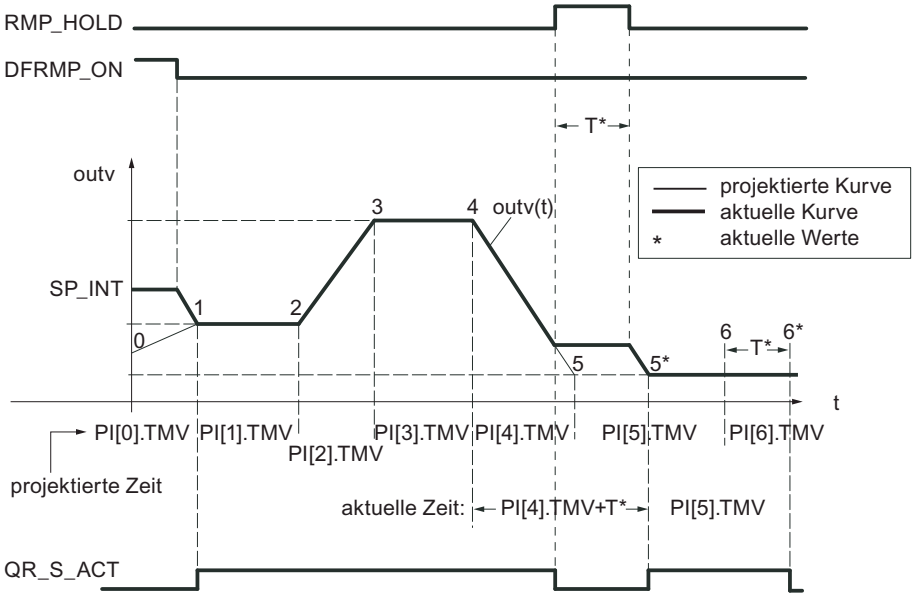

Die Bearbeitungszeit der Fahrkurve verlängert sich um die Zeit T*. Die Fahrkurve hat vom Startpunkt bis zur steigenden Flanke an RMP_HOLD und vom Stützpunkt 5* bis zum Stützpunkt 6* den projektierten Verlauf, d.h. das Ausgangssignal QR_S_ACT hat den Wert TRUE.

Ist der Schalter CONT_ON gesetzt, fährt der angehaltene Zeitplangeber an der vorgegebenen Stelle TM_CONT mit der Fahrkurvenbearbeitung fort.

Zeitplangeber fortsetzen

Mit CONT_ON wird eine angehaltene Fahrkurve an eine definierten Stelle wieder aufgenommen. Wenn der Zeitplangeber nicht angehalten ist, hat CONT_ON keine Auswirkungen.

Ist der Steuereingang für das Fortsetzen CONT_ON = TRUE gesetzt, dann wird der angehaltene Zeitplangeber am Stützpunkt TM_SNBR fortgesetzt. TM_CONT bestimmt die Restzeit, die der Zeitplangeber benötigt, bis der Stützpunkt TM_SNBR erreicht wird.

Das folgende Bild zeigt den Einfluss von RMP_HOLD und CONT_ON bei folgender Konfituration:

Fortsetzen am Stützpunkt 5 (TM_SNBR = 5) und Fortsetzen in T* s: TM_CONT = T*

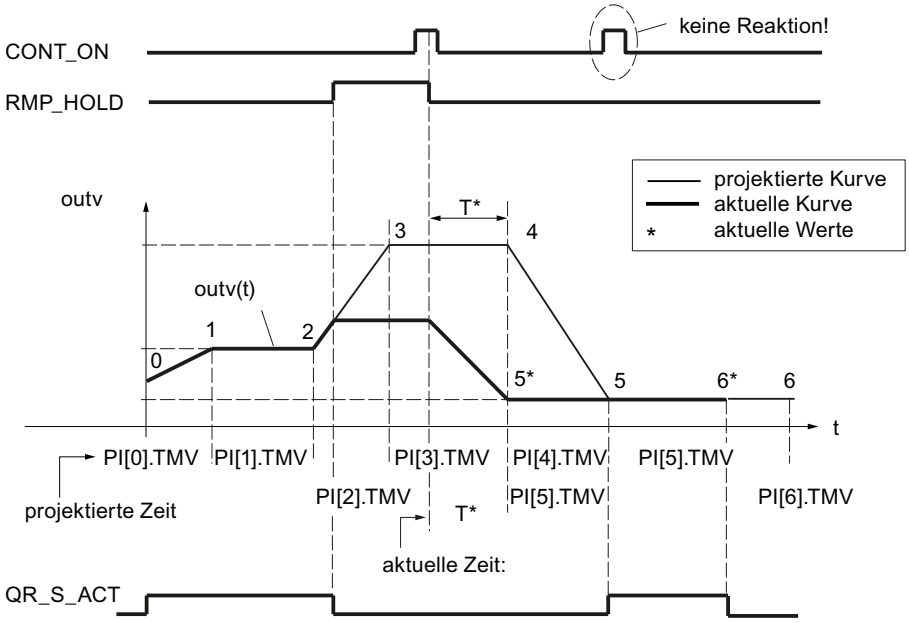

Die Stützpunkte 3 und 4 entfallen für diesen en Bearbeitungszyklus des Zeitplangebers. Nach Signalwechsel RMP_HOLD von TRUE nach FALSE wird der projektierte Kurvenverlauf erst ab Stützpunkt 5 wieder erreicht.

Der Ausgang QR_S_ACT wird nur dann gesetzt, wenn der Zeitplangeber die vom Anwender parametrierte Fahrkurve abarbeitet.

Gesamtzeit und Gesamtrestzeit aktualisieren

In jedem Zyklus werden die aktuelle Stützpunktnummer NBR_ATMS, die aktuelle Restzeit bis zum Erreichen des Stützpunktes RS_TM, die Gesamtzeit T_TM und die Gesamtrestzeit bis zum Erreichen des Fahrkurvenendes RT_TM aktualisiert.

Bei Online–Änderungen von PI[n].TMV ändert sich die Gesamtzeit und die Gesamtrestzeit der Fahrkurve. Da die Berechnung von T_TM und RT_TM bei vielen Stützpunkten die Bearbeitungszeit des Funktionsbausteins stark vergrößert, wird sie nur nach Neustart oder bei TUPDT_ON = TRUE durchgeführt. Die Zeitabschnitte PI[0 ... NBR_PTS].TMV zwischen den einzelnen Stützpunkten werden aufsummiert und am Ausgang Gesamtzeit T_TM und Gesamtrestzeit RT_TM angezeigt.

Bitte beachten Sie, dass die Ermittlung der Gesamtzeiten relativ viel CPU–Laufzeit erfordert!

Vorgehen

Um den Zeitplangeber zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Liste "Sollwertquelle" den Eintrag "Zeitplangeber".
2. Legen Sie einen AUTOHOTSPOT (PID-CP) bzw. AUTOHOTSPOT (PID-ES) an.
3. Definieren Sie die benötigen Parameter für den Zeitplan
4. Geben Sie die Nummer des Datenbausteins DB_RMPSK im Eingabefeld "DB-Nummer" ein.

5. Aktivieren oder deaktivieren Sie die weiteren Funktionen des Zeitplangebers.
6. Wenn Sie die Funktion Zeitplangeber fortsetzen aktiviert haben, geben Sie ein, an welchem Stützpunkt und wann der Zeitplangeber fortgesetzt werden soll.

Normierung

Normierung eines externen Sollwertes

Liegt der externe Sollwert nicht in der physikalischen Einheit des Istwerts vor (z. B. in % bei einer Reglerkaskade), dann müssen dieser Wert und sein Einstellbereich auf die physikalische Einheit des Istwertes normiert werden. Dies erfolgt über die Funktion "Normierung im Sollwertzweig".

Die Funktion SP_NORM normiert eine analoge Eingangsgröße. Der analoge externe Sollwert wird mit Hilfe der Normierungsgeraden in die Ausgangsgröße outv überführt. Der Ausgangswert OUTV ist am Konfigurationswerkzeug über den Messpunkt MP2 zugänglich.

Der Ausgangswert der Funktion ist wirksam, wenn der Steuereingang SPEXT_ON = TRUE ist. Zur eindeutigen Festlegung der Normierungsgeraden definieren Sie die folgenden vier Parameter:

- Eingang oben: SP_EXT: NM_SPEHR
- Eingang unten: SP_EXT: NM_SPELR
- Ausgang oben: NM_PVHR (Diesen Wert geben Sie in der Normierfunktion des Istwerts vor.)
- Ausgang unten: NM_PVLR (Diesen Wert geben Sie in der Normierfunktion des Istwerts vor.)

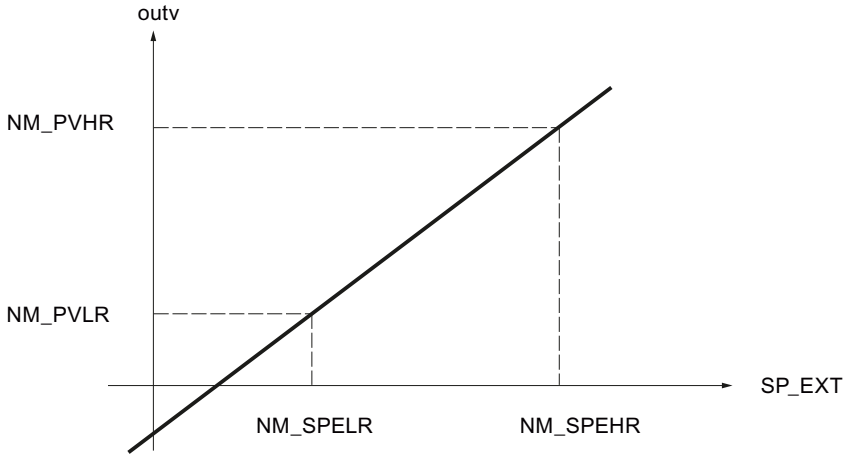

Der Ausgangswert outv wird gemäß folgender Formel aus dem jeweiligen Eingangswert SP_EXT berechnet:

outv = (SP_EXT – NM_SPELR) x (NM_PVHR – NM_PVLR) / (NM_SPEHR – NM_SPELR) + NM_PVLR

Für den Sonderfall einer eingeschalteten Wurzelfunktion im Istwertzweig gelten als Ober– und Untergrenze des Ausgangswerts outv die Normierungswerte der Wurzelfunktion (SQRT_HR und SQRT_LR).

Der Ausgangswert outv der Normierungsfunktion wird in diesem Fall gemäß folgender Formel aus dem jeweiligen Eingangswert SP_EXT berechnet:

outv = (SP_EXT – NM_SPELR) x (SQRT_HR – SQRT_LR) / (NM_SPEHR – NM_SPELR) + SQRT_LR

Funktionsintern werden keine Werte begrenzt, und eine Prüfung der Parameter findet nicht statt. Falls Sie für NM_SPEHR und NM_SPELR denselben Wert eingeben, kann nach den obigen Formeln eine Division durch Null auftreten. Dies wird in der Anweisung nicht abgefangen!

Mischungs-/Verhältnisfaktor

Bei einer Verhältnisregelung werden die Sollwerte der Regelkreise so aneinander angepasst, dass das eingestellte Verhältnis dem tatsächlichen Verhältnis der Istwerte entspricht.

Bei einer Mischungsregelung wird der gewünschte Mengenanteil der einzelnen Komponenten eingestellt. Die Summe der Mischungsfaktoren FAC muss dabei 1 sein.

Vorgehen

Um die Normierung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Geben in die Eingabefelder "Eingang oben" und "Eingang unten" die Werte für die Normierung ein.
2. Geben Sie einen Mischungs-/Verhältnis-Faktor (FAC) ein.

Begrenzungen

FC-Aufruf im Sollwertzweig

Durch Einfügen einer anwenderspezifischen Funktion in den Sollwertzweig ist es möglich, eine extern vorgegebene Führungsgröße vor Aufschaltung auf den Regler einer Signalbehandlung (z. B. einer Signalverzögerung oder Linearisierung) zu unterziehen.

Aktiviert man den FC-Aufruf im Sollwertzweig, wird eine anwenderspezifische Funktion aufgerufen. Der Regler führt einen Aufruf der FC durch. Vorhandene Ein–/ Ausgangsparameter der Anwender–FC werden dabei nicht versorgt. Die Datenübertragung müssen Sie deshalb mit S7–AWL in der Anwender–FC programmieren. Den aufbereiteten Sollwert können Sie an MP3 beobachten.

Der Wert von SPFC_ON bestimmt dann, ob in den Sollwertkanal an dieser Stelle eine frei programmierte Funktion in Form einer Standard–FC (z. B. eine Kennlinie) eingefügt wird oder ob der Sollwert ohne eine derartige Beeinflussung weiter verarbeitet wird.

Hinweis

Die Anweisung überprüft nicht, ob eine Funktion vorhanden ist. Ist die Funktion nicht vorhanden, dann geht die CPU mit einem internen Systemfehler in STOP.

Vorgehen

Um eine Funktion im Sollwertzweig aufzurufen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "FC-Aufruf aktivieren".
2. Geben Sie im Eingabefeld "FC" die Nummer der aufzurufenden Funktion ein.

Begrenzung der Änderungsgeschwindigkeit des Sollwertes

Rampenfunktionen im Sollwertzweig werden dann verwendet, wenn der Prozess keine sprungförmigen Änderungen des Stellsignals verträgt, denn eine sprungförmige Sollwertänderung hat im Allgemeinen auch einen Stellwertsprung des Reglers zur Folge. Diese abrupten Stellwertänderungen sind z. B. dann zu vermeiden, wenn zwischen geregeltem Motor und anzutreibender Last ein Getriebe zwischengeschaltet ist und ein zu schnelles Ansteigen der Motordrehzahl zu einer Überlastung des Getriebes führen würde.

Die Funktion SP_ROC begrenzt die Änderungsgeschwindigkeit der im Regler verarbeiteten Sollwerte sowohl getrennt für den An– und Abstieg als auch getrennt für den positiven und den negativen Bereich der Führungsgröße.

An den vier Eingängen SPURLM_P, SPDRLM_P, SPURLM_N und SPDRLM_N werden die jeweiligen Begrenzungen für die Steigungen der Rampenfunktion im positiven und negativen Bereich der Führungsgröße eingegeben. Die Steigungen beziehen sich auf einen Anstieg bzw. Abfall pro Sekunde. Schnellere Sollwertänderungen werden auf diese Grenzgeschwindigkeiten verzögert.

Wird z. B. SPURLM_P auf 10.0 [techn. Wertebereich/s] parametriert, so werden in jedem Abtastzyklus die folgenden Werte zum "Altwert" von outv addiert, und zwar solange inv > outv ist:

Abtastzeit	1 s	$outv_{alt}$ + 10
	100 ms	$outv_{alt}$ + 1
	10 ms	$outv_{alt}$ + 0.1

Die Funktionsweise der Signalbehandlung zeigt das folgende Bild anhand eines Beispiels. Aus Sprungfunktionen am Eingang inv(t) werden Rampenfunktionen am Ausgang outv(t).

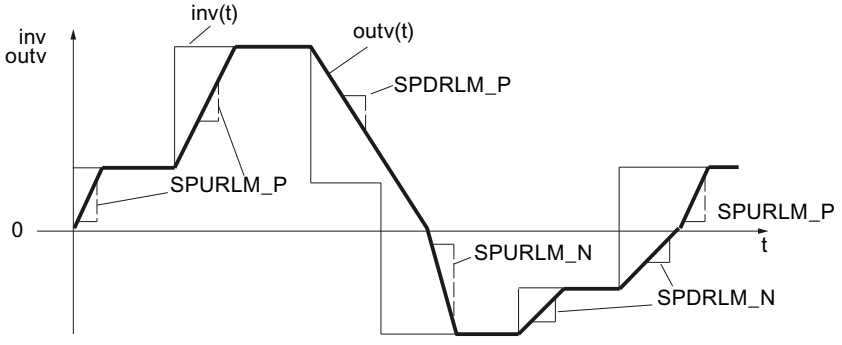

Bild 3-6 Begrenzung der Änderungsgeschwindigkeit der Führungsgröße SP(t)

Das Erreichen der Steigungsbegrenzungen wird nicht gemeldet!

Vorgehen

Um die Änderungsgeschwindigkeit zu begrenzen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "Änderungsgeschwindigkeit begrenzen".
2. Geben Sie sowohl für den positiven als auch für den negativen Bereich folgende Werte fest:
   - Sollwertanstieg
   - Sollwertabstieg

Begrenzung des Absolutwertes der Führungsgröße

Der Einstellbereich der Führungsgröße bestimmt den Bereich, in dem die Regelgröße schwanken kann, d. h. den Bereich, in dem sich der Prozess im Rahmen zulässiger Zustandswerte bewegt.

Zur Vermeidung von kritischen oder unzulässigen Prozesszuständen wird deshalb im Sollwertzweig von Standard PID Control der Einstellbereich der Führungsgröße nach oben und unten begrenzt (SP_LIMIT).

Die Funktion SP_LIMIT begrenzt den Sollwert SP so lange auf den vorgebbaren unteren und oberen Wert SP_LLM und SP_HLM, wie die Eingangsgröße INV außerhalb dieser Begrenzungen liegt. Da die Funktion nicht abgeschaltet werden kann, muss bei der Konfiguration immer die Angabe einer Unter– und Obergrenze berücksichtigt werden.

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für die untere und obere Begrenzung eingestellt. Bei Überschreitungen durch die Eingangsgröße inv(t) werden die zugehörigen Anzeigen über die Meldeausgänge ausgegeben.

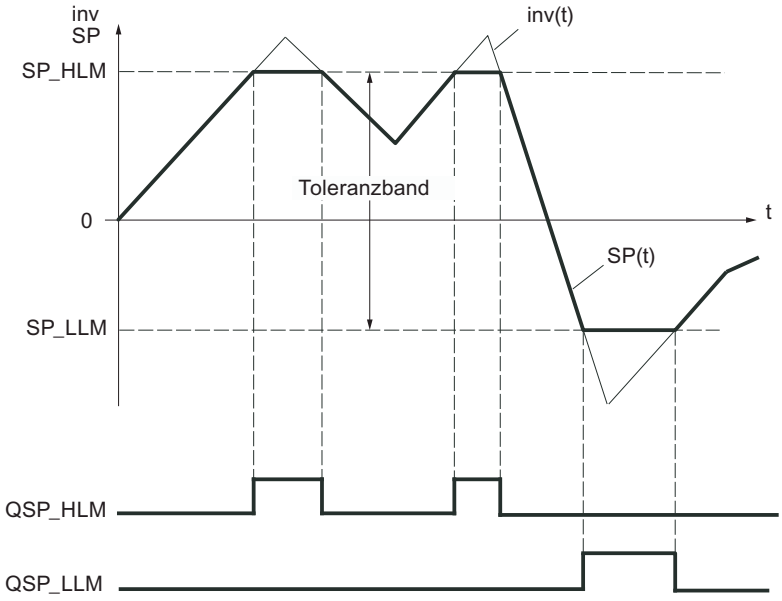

Bild 3-7 Absolutwertbegrenzung der Führungsgröße SP(t)

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

SP =PVH_ALM	QSP_HLM	QSP_LLM	wenn:
SP_HLM	TRUE	FALSE	inv ≥ SP_HLM
SP_LLM	FALSE	TRUE	inv ≤ SP_LLM
inv	FALSE	FALSE	SP_HLM < inv < SP_LLM

Vorgehen

Um den Absolutwert der Führungsgröße zu begrenzen, gehen Sie folgendermaßen vor:

1. Geben Sie im Eingabefeld "Obergrenze" die obere absolute Sollwertgrenze (SP_HLM) ein.
2. Geben Sie im Eingabefeld "Untergrenze" die untere absolute Sollwertgrenze (SP_LLM) ein.

Regeldifferenz

Totzone

Bei Überlagerung der Regel- oder der Führungsgröße mit einem höherfrequenten Störsignal wird bei optimal eingestelltem Regler der Rauschanteil auch am Reglerausgang wirksam. Dies kann z. B. bei hoher Verstärkung des Reglers und eingeschalteter Differenzierung starke Schwankungen der Stellgröße verursachen. Wegen der vermehrten Schalthäufigkeit (Schrittregler) hat das einen erhöhten Verschleiß des Stellgliedes zur Folge.

Die Funktion vermindert im eingeschwungenen Zustand des Reglers den Rauschanteil im Signal der Regeldifferenz und vermindert dadurch ein unerwünschtes Oszillieren des Reglerausgangs.

Die Totzone kann abgeschaltet werden.

Signalfilterung durch Totzonenfunktion

Die Funktion DEADBAND unterdrückt in einem einstellbaren Bereich kleine Schwankungen der Eingangsgröße um einen festgelegten Nullpunkt. Außerhalb dieser Schwankungsbreite steigt bzw. fällt die Regeldifferenz ER proportional zur Eingangsgröße. Die Breite der unwirksamen Zone (Totzone) kann mit dem Parameter DEADB_W festgelegt werden. Die Totzonenbreite darf nur positive Werte annehmen.

Befindet sich die Eingangsgröße innerhalb der Totzone, wird am Ausgang der Wert 0 (Regeldifferenz = 0) ausgegeben. Erst wenn die Eingangsgröße diesen Unempfindlichkeitsbereich verlässt, steigt bzw. fällt der Ausgang um die gleichen Werte wie die Eingangsgröße inv. Daraus resultiert eine Verfälschung des übertragenen Signals auch außerhalb der Totzone. Das wird jedoch zur Vermeidung von Sprüngen an den Grenzen der Totzone in Kauf genommen (siehe nachfolgendes Bild). Die Verfälschung entspricht dem Wert DEADB_W und ist deshalb leicht kontrollierbar.

Die Funktion DEADBAND arbeitet gemäß folgenden Funktionen:

(ER) = inv + DEADB_W für inv < - DEADB_W

(ER) = 0 für -DEADB_W ≤ inv ≤ + DEADB_W

(ER) = inv - DEADB_W für inv > + DEADB_W

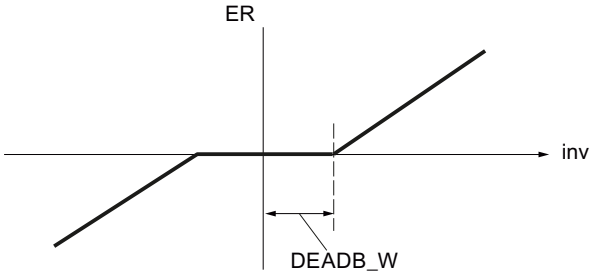

Auswirkungen der Signalfilterung können Sie am Ausgangsparameter ER beobachten.

Alarm

Zu große Abweichungen der Regelgröße vom eingestellten Sollwert können unerwünschte Prozesszustände hervorrufen. In diesen Fällen wird die Funktion ER_ALARM eingesetzt, um die Regeldifferenz auf Über–/Unterschreitung des zulässigen Betriebsbereiches zu überwachen. ER_ALARM erfasst und meldet evtl. auftretende Grenzüberschreitungen, damit eine sinnvolle Reaktion ausgelöst werden kann.

Hinweis

Die Überwachung der Regeldifferenz auf Grenzwerte kann nicht ausgeschaltet werden. Die Überschreitung der konfigurierten Grenzwerte wird immer gemeldet.

Die Funktion ER_ALARM überwacht die Größe der Regeldifferenz ER(t) auf vier vorgebbare Grenzen in zwei Toleranzbändern. Werden die Grenzen erreicht oder überschritten, meldet die Funktion zunächst "Warnung" und an der zweiten Grenze "Alarm".

Grenzwerte

Die Zahlenwerte der Grenzen werden an den Eingangsparametern für "Warnung" und "Alarm" eingestellt. Überschreitet bzw. unterschreitet die Regeldifferenz (ER) diese Grenzen, dann werden die zugehörigen Ausgangsmeldebits QERN_ALM ... QERP_ALM gesetzt.

Bei Neustart werden die Meldebits zurück gesetzt.

Die Meldebits werden folgendermaßen gesetzt:

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	wenn:	und:
TRUE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_ALM ER ≥ ERP_ALM - ER_HYS
FALSE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_WRN ER ≥ ERP_WRN - ER_HYS

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	wenn:	und:
FALSE	FALSE	TRUE	FALSE	ER ↘ ER ↗	ER ≤ ERP_WRN ER ≤ ERP_WRN + ER_HYS
FALSE	FALSE	TRUE	TRUE	ER ↘ ER ↗	ER ≤ ERP_ALM ER ≤ ERP_ALM + ER_HYS

Für ein sinnvolles Arbeiten des Bausteins muss gelten:

Oberer Alarm < Obere Warnung < Untere Warnung < Unterer Alarm

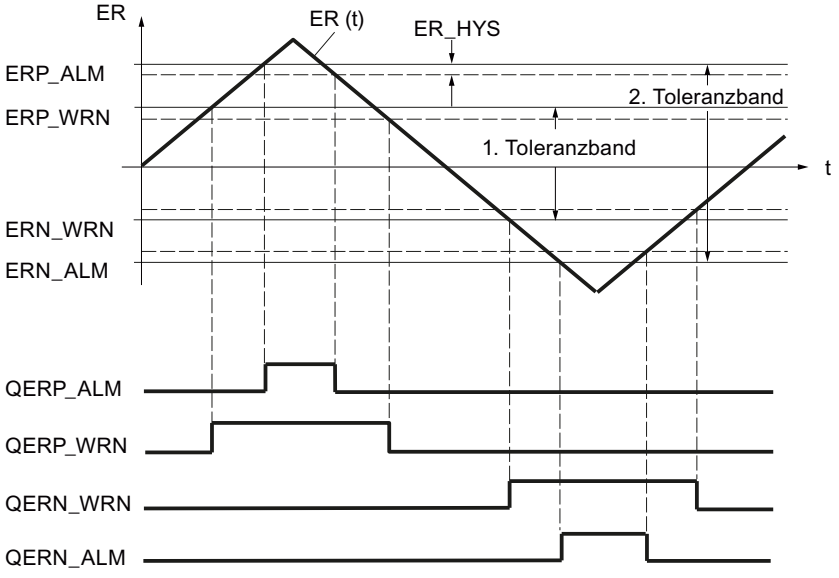

Hysterese

Um bei geringfügigen Änderungen der Eingangsgröße oder bei Rundungsfehlern ein "Flackern" der Meldebits zu vermeiden, wird eine Hysterese (Rückschaltdifferenz) ER_HYS eingestellt. Diese Hysterese muss die Regeldifferenz überwinden, bevor die Meldebits rückgesetzt werden.

Regler

Regelalgorithmus

Im Zyklus der projektierten Abtastzeit wird die Stellgröße des kontinuierlich arbeitenden Reglers aus der Regeldifferenz im PID–Stellungsalgorithmus errechnet. Der Regler ist in reiner Parallelstruktur ausgeführt. Der Proportional–, Integral– oder Differentialanteil kann jeweils einzeln abgeschaltet werden.

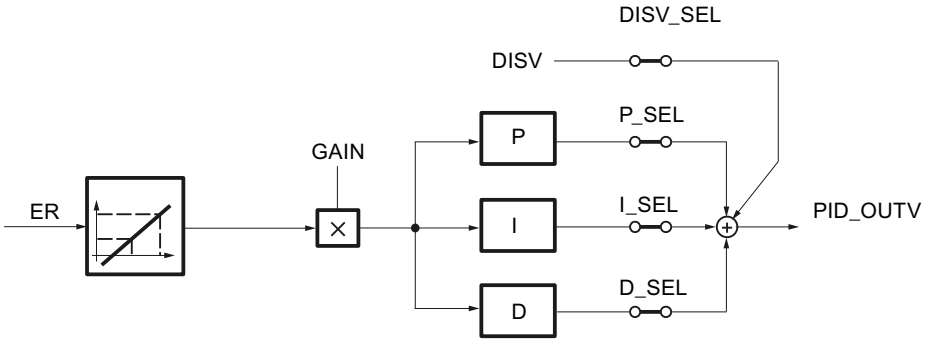

Störgrößenaufschaltung

Dem Ausgangssignal des Reglers PID_OUTV kann zusätzlich eine Störgröße DISV aufgeschaltet werden. Die Zu– bzw. Abschaltung erfolgt im PID–Fenster des Konfigurationswerkzeugs über den Strukturschalter DISV_SEL bzw. durch "Störgröße ein".

P- und D-Anteil in der Rückführung

In der Parallelstruktur erhält jeder Anteil des Regelalgorithmus die Regeldifferenz als Eingangssignal. Die Stellgröße wird über den P–Anteil den D–Anteil durch Sollwertsprünge unmittelbar beeinflusst. Eine andere Struktur des Reglers jedoch, bei der die Bildung des P–Anteils des D–Anteils in die Rückführung verlegt wird, garantiert bei Sprungänderungen der Führungsgröße einen stoßfreien Verlauf der Stellgröße. In dieser Struktur verarbeitet der I–Anteil die Regeldifferenz als Eingangssignal, auf den P–Anteil und den D–Anteil wird nur die negative Regelgröße (Faktor = -1) aufgeschaltet.

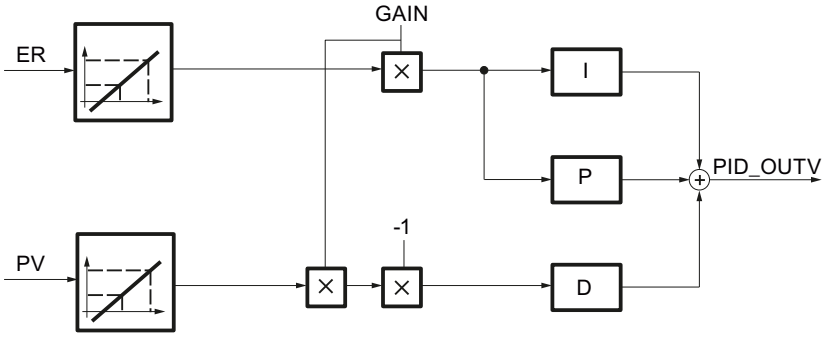

Normierung der Eingangsgrößen ER und PV

Die Eingangsgrößen ER und PV des PID–Reglers werden vor der Reglerbearbeitung normiert auf den Bereich 0 bis 100 gemäß folgenden Formeln:

- bei ausgeschalteter Wurzelfunktion (SQRT_ON = FALSE):
  - $ER_{normiert}$ = ER * 100.0 / (NM_PVHR – NM_PVLR)
  - $PV_{normiert}$ = (PV - NM_PVLR) * 100.0 / (NM_PVHR – NM_PVLR)
- bei eingeschalteter Wurzelfunktion (SQRT_ON = TRUE):
  - $ER_{normiert}$ = ER * 100.0 / (SQRT_HR – SQRT_LR)
  - $PV_{normiert}$ = (PV - SQRT_LR) * 100.0 / (SQRT_HR – SQRT_LR)

Diese Normierung wird durchgeführt, damit die Proportionalverstärkung GAIN des PID–Reglers dimensionslos eingegeben werden kann. Bei einer Änderung der Obergrenze und Untergrenze des physikalischen Messbereichs (z. B. von bar auf mbar) muss somit der Verstärkungsfaktor nicht geändert werden.

Die normierten Eingangsgrößen $ER_{normiert}$ und $PV_{normiert}$ sind nicht beobachtbar.

Siehe auch

Reglerstrukturen (Seite 908)

P-Anteil (Seite 912)

D-Anteil (Seite 915)

I-Anteil (Seite 912)

Reglerstrukturen

Einsatz und Parametrierung des PID–Reglers

Die meisten der in der Verfahrensindustrie und Prozessindustrie vorkommenden Regelstrecken lassen sich mit der PI–Reglerfunktion/PID–Reglerfunktion von Standard PID Control regeln. Nur in Spezialfällen sind zusätzliche Methoden und Maßnahmen zur Beherrschung der betreffenden Regelung erforderlich.

Ein großes praktisches Problem ist jedoch nach wie vor die Parametrierung des PI–Reglers/ PID–Reglers, d. h. die "richtigen" Einstellwerte für die Reglerparameter zu finden. Die Qualität dieser Parametrierung ist von ausschlaggebender Bedeutung für die aufgabengemäße Wirkung der PID–Regelung und erfordert entweder große praktische Erfahrung, Spezialkenntnisse oder einen hohen Zeitaufwand.

P-Regelung

Beim P–Regler sind der I–Anteil und der D–Anteil abgeschaltet. (I_SEL und D_SEL = FALSE). Das bedeutet, dass bei Regeldifferenz ER = 0 auch das Ausgangssignal OUTV = 0 ist. Soll ein Arbeitspunkt ≠ 0, d. h. ein Zahlenwert für das Ausgangssignal bei Regeldifferenz Null eingestellt werden, so ist der I–Zweig zu aktivieren (siehe nachfolgendes Bild).

Im I–Anteil lässt sich über entsprechende Einstellung des Initialisierungswertes I_ITLVAL ein Arbeitspunkt ≠ 0 für den P–Regler vorgeben. Setzen Sie dazu die Schalter 'I_ITL_ON' und 'I_SEL' = TRUE.

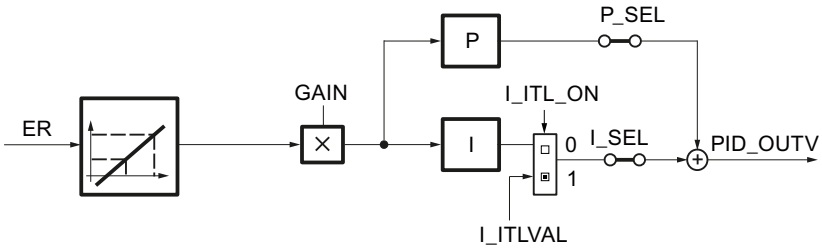

Die Sprungantwort des P–Reglers im Zeitbereich ist:
PID_OUTV (t) = I_ITLVAL + GAIN * ER $_{normiert}$ (t)

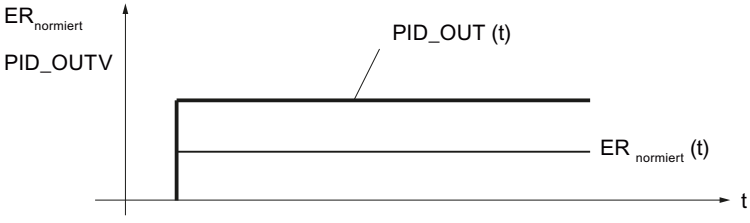

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
I_ITLVAL	der Arbeitspunkt des P–Reglers
GAIN	die Reglerverstärkung
ER $_{normiert}$ (t)	die normierte Regeldifferenz

PI-Regelung

Beim PI–Regler ist der D–Anteil abgeschaltet. (D_SEL = FALSE). Ein PI–Regler verstellt über den I–Anteil die Ausgangsgröße PID_OUTV so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0)\left(1 + \frac{1}{TI} * t\right)$$

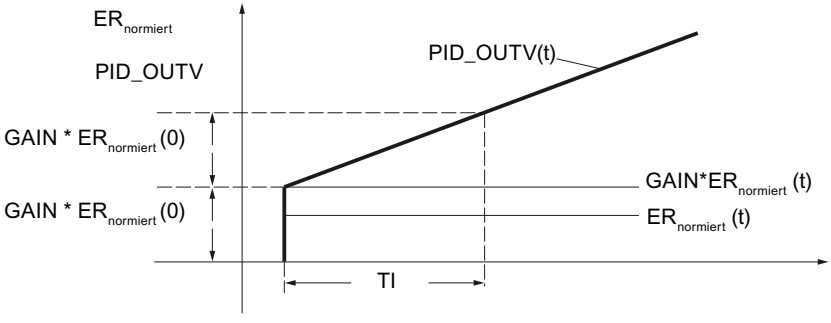

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TI	Integrationszeitkonstante

PD-Regelung

Beim PD–Regler ist der I–Anteil abgeschaltet (I–SEL = FALSE). Das bedeutet, dass bei Regeldifferenz ER = 0 auch das Ausgangssignal OUTV = 0 ist. Soll ein Arbeitspunkt ≠ 0, d. h. ein Zahlenwert für das Ausgangssignal bei Regeldifferenz Null eingestellt werden, so muss der I–Zweig aktiviert werden (siehe auch Blockschaltbild der P-Regelung.

Im I–Anteil lässt sich über entsprechende Einstellung des Initialisierungswertes I_ITLVAL ein Arbeitspunkt ≠ 0 für den PD–Regler vorgeben. Dazu sind die Schalter 'I_ITL_ON' und 'I_SEL' auf TRUE zu setzen.

Der PD–Regler bildet die Eingangsgröße ER(t) proportional auf das Ausgangssignal ab und addiert den durch Differentiation von ER(t) gebildeten D–Anteil dazu, der nach der Trapezregel (Padé–Näherung) doppeltgenau berechnet wird. Das Zeitverhalten wird durch die Differentiationszeitkonstante (Vorhaltzeit) TD bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeit: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert. Meist genügt ein kleiner Wert für TM_LAG, um den gewünschten Erfolg zu erzielen. Wird TM_LAG ≤ CYCLE/2 parametriert, dann ist die Verzögerung abgeschaltet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0)\left(1 + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$

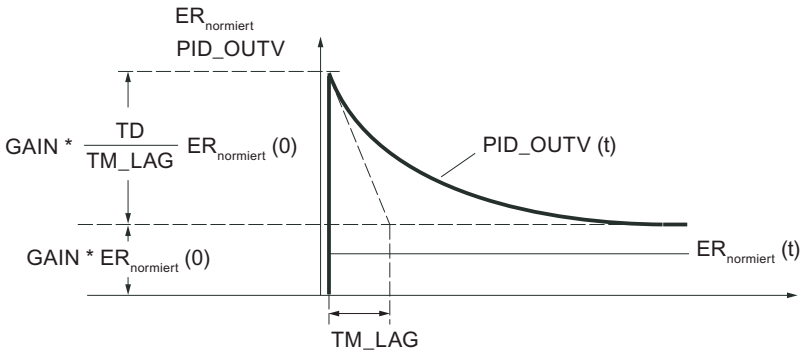

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TD	Differentiationszeit
TM_LAG	Verzögerungszeit

PID-Regelung

Beim PID–Regler sind der P–Anteil, I–Anteil und D–Anteil eingeschaltet (P–SEL = TRUE, I–SEL = TRUE, D_SEL = TRUE). Ein PID–Regler verstellt die Ausgangsgröße PID_OUTV über den I–Anteil so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet. Werden Stellwertbegrenzungen überschritten, behält der I–Anteil den an der Grenze erreichten Wert bei (Anti Reset Wind–up).

Der PID–Regler bildet die normierte Eingangsgröße $ER_{normiert}(t)$ proportional auf das Ausgangssignal ab und addiert die durch Differentiation und Integration von $ER_{normiert}(t)$ gebildeten Anteile dazu, die nach der Trapezregel (Padé–Näherung) doppeltgenau berechnet werden. Das Zeitverhalten wird durch die Differentiationszeit (Vorhaltzeit) TD und die Integrationszeit (Nachstellzeit) TI bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeitkonstante: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert. Meist genügt ein kleiner Wert für TM_LAG, um den gewünschten Erfolg zu erzielen. Wird TM_LAG ≤ CYCLE/2 parametriert, dann ist die Verzögerung abgeschaltet.

Die Sprungantwort im Zeitbereich ist:

$$PID_OUTV(t) = GAIN * ER_{normiert}(0) \left(1 + \frac{1}{TI} * t + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}} \right)$$

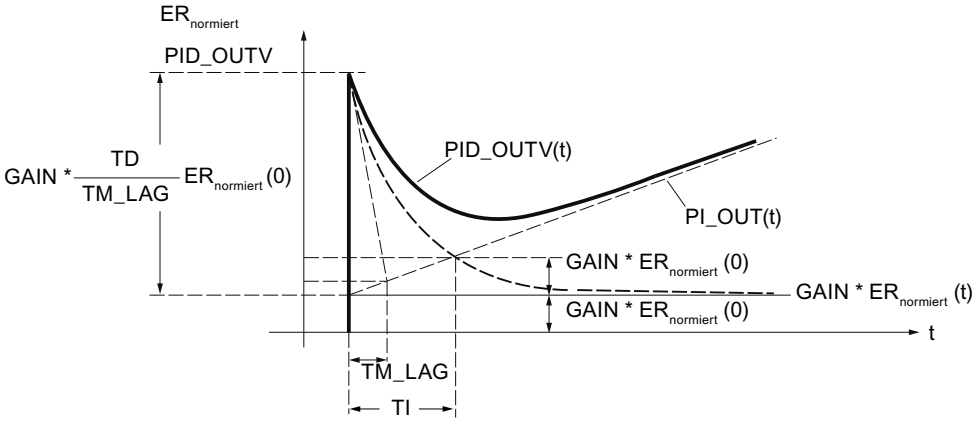

PID_OUTV(t)	die Stellgröße bei Automatikbetrieb des Reglers
GAIN	die Reglerverstärkung
$ER_{normiert}(0)$	die Sprunghöhe der normierten Regeldifferenz
TI	Integrationszeitkonstante
TD	Differentiationszeit
TM_LAG	Verzögerungszeit

Hinweis

Bei einer Änderung von TD sollten Sie auch TM_LAG entsprechend anpassen.

Empfehlung: 5 ≤ (TM / TM_LAG) ≤ 10

Siehe auch

Regelalgorithmus (Seite 907)

P-Anteil (Seite 912)

D-Anteil (Seite 915)

P-Anteil

Proportionalverstärkung

Das Vorzeichen der Proportionalverstärkung bestimmt die Wirkungsrichtung des kontinuierlichen Reglers.

- Positive Proportionalverstärkung
  steigende Regeldifferenz → steigender Stellwert
- Negative Proportionalverstärkung
  steigende Regeldifferenz → fallender Stellwert

P-Anteil in Rückführung legen

Ist der P-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den P-Anteil. Die Stellgröße macht daher – wie auch bei einem Sollwertsprung – bei einer Hand-Automatik-Umschaltung keinen Sprung durch den P-Anteil; die Umschaltung ist stoßfrei.

Siehe auch

Regelalgorithmus (Seite 907)

I-Anteil

Funktion des I-Anteils

Der I-Anteil sorgt dafür, dass durch Nachstellung des Arbeitspunktes die Regeldifferenz für beliebige Werte der Stellgröße zu 0 werden kann.

Der I-Anteil erzeugt ein Ausgangssignal, dessen Änderungsgeschwindigkeit sich proportional zur Änderung des Absolutwertes der Eingangsgröße verhält. Das Zeitverhalten wird durch die Integrationszeit TI bestimmt.

Die Übertragungsfunktion im Zeitbereich ist:

$$OUTV(t) = \frac{1}{TI} \int inv(t)\, dt$$

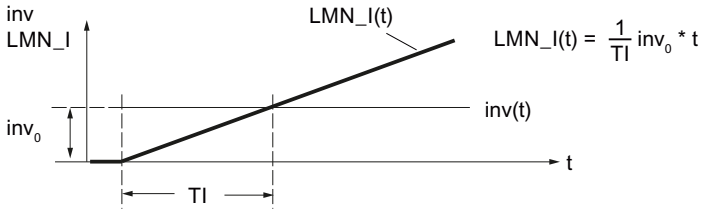

LMN_I(t)	die Ausgangsgröße des I-Anteils
inv_0	die Sprunghöhe am Eingang des I-Anteils
TI	Integrationszeit

Der I-Anteil ist an der statischen Variable LMN_I zu beobachten.

Parameter	Bedeutung	Zulässiger Wertebereich
TI	Integrationszeit	≥ 5 * CYCLE
I_ITLVAL	Initialisierungswert für I–Anteil	-100.0 bis +100.0 [%]

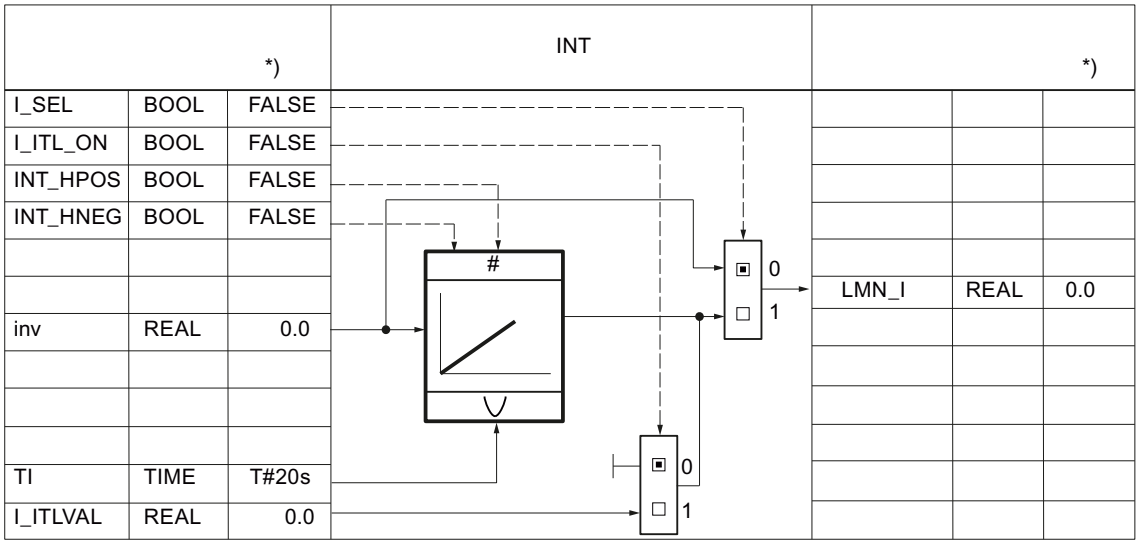

*) Vorbelegung bei Neuerstellung des Technologieobjekts

Im ausgeschalteten Zustand (I_SEL = FALSE) wird der I-Anteil, d. h. der interne Speicher und die statische Variable LMN_I auf Null gesetzt.

Der Ausgang und der Speicher des I-Anteils wird durch die absoluten Stellwertgrenzen LMN_HLM und LMN_LLM begrenzt (Anti Reset Wind–up).

Bei Beeinflussung des Stellsignals im Handbetrieb, d. h. wenn MAN_ON, LMNOP_ON oder CAS_ON = TRUE ist, wird der interne Speicherwert des I-Anteils dem Wert LMN nachgeführt.

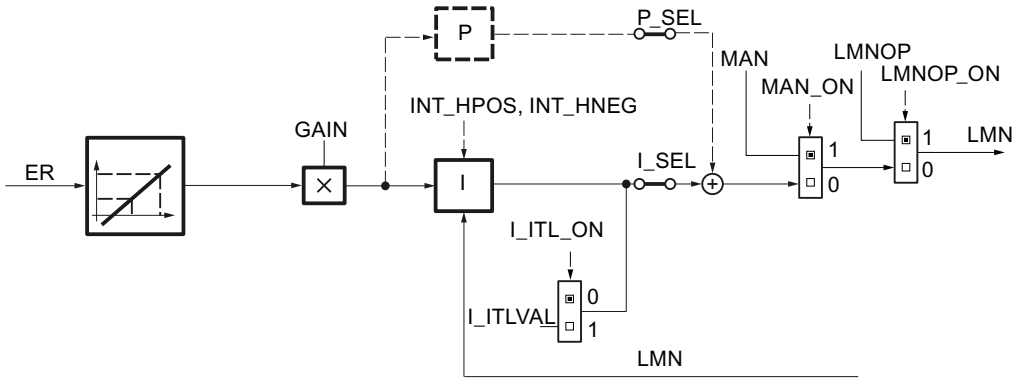

Umschaltung von Handbetrieb nach Automatikbetrieb

Der I–Anteil wird im Handbetrieb so gesetzt, dass die Stellgröße bei der Hand–Automatik–Umschaltung ausgehend von der aktuellen Stellung einen Sprung macht, dessen Höhe durch den P-Anteil bestimmt wird. Die Sprunghöhe des P-Anteils entspricht der Stellgrößenänderung bei einem Sollwertsprung vom aktuellen Istwert auf den aktuellen Sollwert.

Ist jedoch der P-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den P–Anteil. Die Stellgröße macht daher bei einer Hand-Automatik-Umschaltung keinen Sprung durch den P-Anteil; die Umschaltung ist stoßfrei.

Zulässige Bereiche für die Integrationszeit TI und CYCLE

Wegen der begrenzten Genauigkeit der in der CPU errechneten REAL–Zahlen kann beim Integrieren folgender Effekt auftreten: wurde die Abtastzeit CYCLE der Regelung im Vergleich zur Integrationszeit TI zu klein gewählt und ist der Eingangswert inv des I-Anteils klein gegenüber seinem Ausgangswert OUTV, dann spricht der I-Anteil nicht an und bleibt auf seinem momentanen Ausgangswert stehen.

Dieser Effekt lässt sich vermeiden, wenn bei der Konfiguration folgende Dimensionierungsregel beachtet wird:

$CYCLE > 10^{-4} \times TI$

Damit reagiert der I-Anteil noch auf Änderungen der Eingangswerte, die im Bereich von millionstel Promille der aktuellen Ausgangsgröße liegen:

$inv > 10^{-10} \times OUTV$

Damit das Übertragungsverhalten des Integrier–Algorithmus dem analogen Verhalten entspricht, sollte die Abtastzeit kleiner als 20 % der eingestellten Integrationszeit sein bzw. TI mindestens den fünffachen Wert der gewählten Abtastzeit haben:

$CYCLE < 0{,}2 \times TI$

Der Algorithmus lässt Werte für die Abtastzeit bis $CYCLE \leq 0{,}5 * TI$ zu.

I–Anteil initialisieren

Wird die Initialisierung des I-Anteils aktiviert, dann wird bei Neustart der CPU oder COM_RST = TRUE der Initialisierungswert auf die statische Variable LMN_I durchgeschaltet. Beim Übergang in den Normalbetrieb durch Zurücksetzen der Initialisierung beginnt der I-Anteil vom Initialisierungswert aus mit der Integration seiner Eingangsgröße.

I-Anteil anhalten

Über die statischen Variablen INT_HPOS und INT_HNEG kann der I-Anteil in positive bzw. negative Richtung blockiert werden. Dies kann sinnvoll sein bei Reglerkaskaden. Gerät z. B. die Stellgröße des Folgereglers in die obere Begrenzung, kann somit eine weitere Vergrößerung der Stellgröße des Führungsreglers durch dessen I-Anteil verhindert werden. Dieses Verhalten programmieren Sie durch die folgenden Anweisungen, z. B. in AWL:

```
U    "Folgeregler".QLMN_HLM
=    "Fuehrungsregler".INT_HPOS
U    "Folgeregler".QLMN_ LLM
=    "Fuehrungsregler".INT_HNEG
```

Siehe auch

Regelalgorithmus (Seite 907)

D-Anteil

Funktion des D-Anteils

Der D-Anteil erzeugt ein Ausgangssignal, dessen Größe sich proportional zur Änderungsgeschwindigkeit der Eingangsgröße ändert. Das Zeitverhalten wird durch die Differenzierzeit TD und die Verzögerungszeit TM_LAG bestimmt.

Die Sprungantwort auf einen Eingangssprung inv_0 ist:

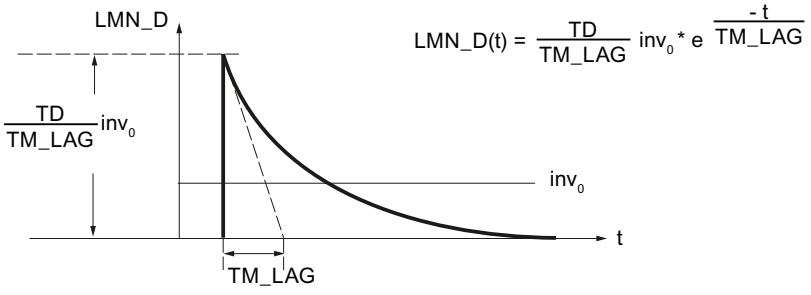

LMN_D(t)	die Ausgangsgröße des D-Anteils
inv_0	die Sprunghöhe am Eingang des D-Anteils
TD	Differenzierzeit
TM_LAG	Verzögerungszeit

Der D-Anteil ist an der statischen Variable LMN_D zu beobachten.

Im ausgeschalteten Zustand (D_SEL = FALSE) wird der D-Anteil, d. h. der interne Speicher und die statische Variable LMN_D auf Null gesetzt.

D-Anteil in Rückführung legen

Ist der D-Anteil in die Rückführung gelegt, dann wirkt nur der Istwert auf den D-Anteil. Die Stellgröße macht daher – wie auch bei einem Sollwertsprung – bei einer Hand-Automatik-Umschaltung keinen Sprung durch den D-Anteil; die Umschaltung ist stoßfrei.

Zulässige Werte für Differenzier- und Verzögerungszeit

Damit der D-Anteil korrekt berechnet werden kann, müssen für die Differenzier- und Verzögerungszeit folgende Bedingungen eingehalten werden:

- TD ≥ CYCLE und
- TM_LAG ≥ 0,5 × CYCLE

Wenn TD < CYCLE, dann arbeitet der D-Anteil so, als würde TD den Wert CYCLE haben.

Wenn TM_LAG < 0,5 × CYCLE, dann arbeitet der D-Anteil ohne Verzögerung. Ein Eingangssprung wird dann mit dem Faktor TD/CYCLE multipliziert und dieser Wert als "Nadelimpuls" auf den Ausgang gegeben. Das heißt, im folgenden Bearbeitungszyklus wird LMN_D wieder auf Null zurückgenommen.

Koeffizient DT1

Der Koeffizient DT1 wird folgendermaßen berechnet:

Koeffizient DT1 = TD/TM_LAG

Verzögerungszeit

Zur Signalglättung und Unterdrückung von Störsignalen ist eine Verzögerung 1. Ordnung (einstellbare Zeitkonstante: TM_LAG) in den Algorithmus zur Bildung des D–Anteils integriert.

Meist genügt ein kleiner Wert als Verzögerungszeit, um den gewünschten Erfolg zu erzielen. Wird die Verzögerungszeit kleiner als die Hälfte der Reglerabtastzeit parametriert, dann ist die Verzögerung abgeschaltet.

Vorgehen

Um den D-Anteil zu konfigurieren, gehen Sie folgendermaßen vor:

1. Um den D-Anteil in Rückführung zu legen, aktivieren Sie das entsprechende Optionskästchen.
2. Geben Sie die Differenzierzeit in Sekunden ein.
3. Geben Sie die Verzögerungszeit in Sekunden ein.

Siehe auch

Regelalgorithmus (Seite 907)

Stellwert

Schrittregler mit Stellungsrückmeldung

Funktionsschema des Schrittreglers mit Stellungsrückmeldung im Regelkreis

Die Arbeitsweise des Schrittreglers mit Stellungsrückmeldung basiert auf dem PID–Regelalgorithmus und wird um die Funktionsglieder zur Erzeugung der binärenAusgangssignale aus dem analogen Stellsignal des Reglers ergänzt

Das Dreipunktglied formt dabei Abweichungen zwischen Stellgröße undStellungsrückmeldung je nach Vorzeichen in positive oder negative Impulse desAusgangssignals um, die z. B. auf einen motorischen Ventilantrieb gegeben werden. Praktischhandelt es sich dabei um eine Kaskadenregelung mit einem unterlagerten Stellungsregelkreis.

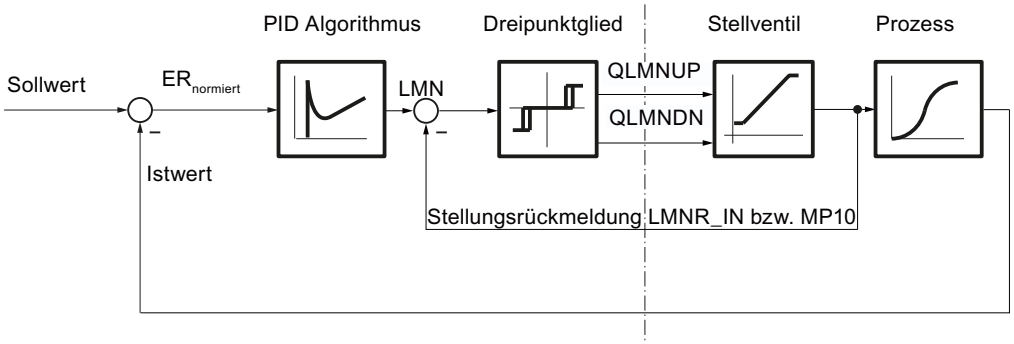

Stellungsrückmeldung auswählen

Für die Aufschaltung der Stellungsrückmeldung auf den Vergleicher im Stellwertzweig des Schrittreglers stehen, je nach Format des zu verarbeitenden Wertes, Eingänge mit entsprechender Signalverarbeitung zur Verfügung. Über LMNR_PER lassen sich Signale im Format der SIMATIC–Peripherie und über LMNR_IN Signale im Gleitpunktformat aufschalten.

Am Messpunkt MP10 wird der zugehörige interne Wert im Prozentbereich hinterlegt.

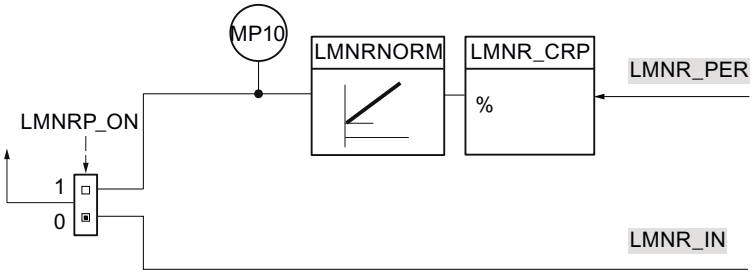

Wird die Stellungsrückmeldung von einer Analogeingabebaugruppe geliefert, dann muss der an LMNR_PER anliegende Zahlenwert der Stellungsrückmeldung in einen auf Prozent normierten Gleitpunktwert umgerechnet werden.

Eine Prüfung auf positiven/negativen Überlauf, Erreichen des Über–/ Untersteuerungsbereiches und auf Drahtbruch wird dabei nicht durchgeführt.

Folgende Tabelle gibt einen Überblick über Bereiche und Zahlenwerte vor und nach der Bearbeitung.

Peripheriewert LMNR_PER	Ausgangswert in %
32767	118,515
27648	100,000
1	0,003617
0	0,000
- 1	- 0,003617
- 27648	- 100,000
- 32768	- 118,519

Vorgehen

Hinweis

Die Stellungsrückmeldung darf im Online-Betrieb nicht aktiviert oder deaktiviert werden. D. H. LMNR_ON darf online nicht verändert werden.

Um die Stellungsrückmeldung zu aktivieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "Aktivieren" im Bereich "Stellungsrückmeldung auswählen".
2. Wählen Sie die Option "Stellungsrückmeldung Peripherie", um Signale im Format der SIMATIC-Peripherie zu aufzuschalten.
   Wählen Sie die Option "Stellungsrückmeldung", um Signale im Gleitpunktformat aufzuschalten.

Stellungssrückmeldung normieren

Liegt die Stellungsrückmeldung als physikalische Größe vor (z. B. 240 bis 800 mm oder im Winkel von 0° bis 60°), dann muss die Stellungsrückmeldung im Gleitpunktformat (%) auf den für die Weiterverarbeitung erforderlichen internen Gleitpunktwert in die physikalische Größe normiert werden.

Zur eindeutigen Festlegung der Normierungsgeraden sind folgende Parameter zu definieren:

- der Faktor (für die Steigung): LMNR_FAC
- der Offset der Normierungsgeraden im Nullpunkt: LMNR_OFF

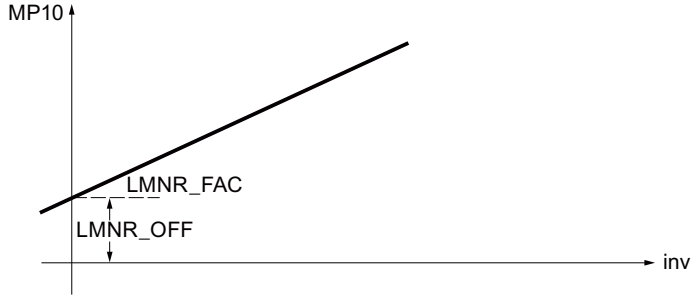

Die normierte Stellungsrückmeldung wird nach folgender Formel aus inv (LMNR_PER) berechnet und an MP10 ausgegeben:

MP10 = inv * LMNR_FAC + LMNR_OFF

Die Normierung ist wirksam, wenn die Stellungsrückmeldung Peripherie eingeschaltet ist (LMNRP_ON = TRUE). Funktionsintern werden keine Werte begrenzt. Eine Prüfung der Parameter findet nicht statt.

Vorgehen

Um das Signal der Stellungsrückmeldung zu normieren, gehen Sie folgendermaßen vor:

1. Geben Sie in das Eingabefeld "Faktor" die Steigung der Normierungsgeraden ein.
2. Geben Sie in das Eingabefeld "Offset" den Offset der Normierungsgeraden zum Nullpunkt ein.

Schrittregler ohne Stellungsrückmeldung

Struktur und Funktion des Schrittreglers

Der Schrittregler ohne Stellungsrückmeldung setzt sich aus zwei Teilen zusammen: dem mit kontinuierlichen Signalen arbeitenden PD–Reglerteil und einem zweiten Teil, in dem die binären Stellsignale aus der Differenz von PD–Anteil und Rückführung erzeugt werden.

Der I–Anteil des Schrittreglers ohne Stellungsrückmeldung wird in der Rückführung berechnet. Das mit dem Reglerausgang des PD–Reglers verglichene Rückführsignal wird hierbei durch Integrieren von folgenden zwei Signalanteilen gebildet:

- Signalkomponente für die simulierte Stellungsrückmeldung
  ± 100/MTR_TM
- Signalkomponente für den I-Anteil
  $ER_{normiert}$*GAIN/TI

Das Rückführsignal ist somit die Differenz zwischen simulierter Stellungsrückmeldung und I-Anteil.

Das Dreipunktglied formt Abweichungen zwischen Stellgröße und Rückführsignal je nach Vorzeichen in positive oder negative Impulse des Ausgangssignals um, die z. B. auf einen motorischen Ventilantrieb gegeben werden.

Im eingeschwungenen Zustand werden der Ausgang des I-Anteils und der PD–Anteil zu Null. Da der Eingang des Dreipunktgliedes auch zu Null wird, bleiben die binären Stellsignale QLMNUP und QLMNDN auf FALSE stehen.

Da die Integration in der Rückführung erfolgt, wird der I-Anteil im PID-Algorithmus nicht berechnet. Der I-Anteil im PID-Algorithmus kann jedoch mit I_ITL_ON und I_ITLVAL auf einen festen Wert gesetzt werden.

Beim Schrittregler ohne Stellungsrückmeldung wird das Einfrieren des I-Anteils mit INT_HPOS und INT_HNEG nicht unterstützt.

Der Handbetrieb über den Parameter MAN entfällt hier wegen der fehlenden Information über die Lage des Stellgliedes.

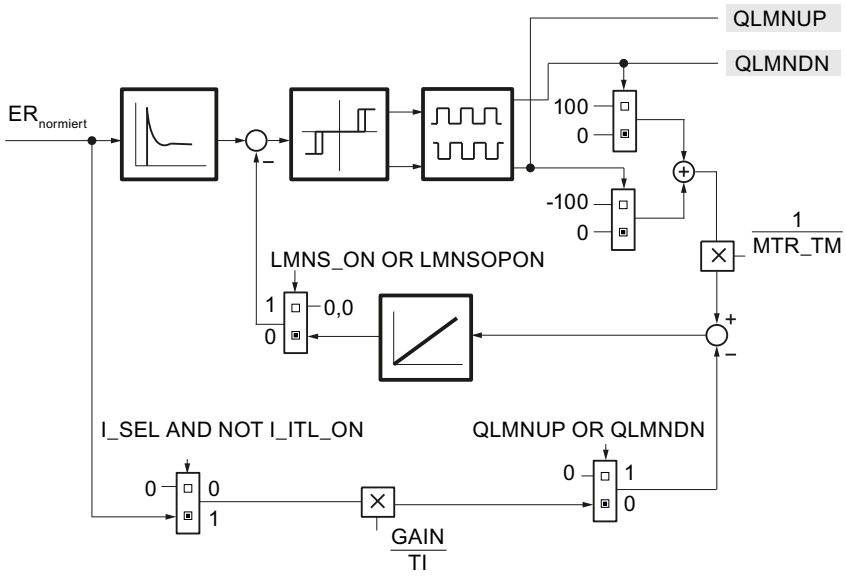

Ein Ansprechen der Anschlagsignale des Stellgliedes (LMNR_HS/LMNR_LS) verursacht eine Sperrung des Stellwerts LMNUP bzw. LMNDN.

Wenn das Stellglied keine Anschlagsignale liefert, dann müssen die Eingangsparameter LMNR_HS und LMNR_LS = FALSE gesetzt werden.

Hinweis

Wenn keine Anschlagsignale vorhanden sind, kann der Regler nicht erkennen, ob ein mechanischer Anschlag erreicht wurde. Es besteht die Möglichkeit, dass der Regler z. B. Signale zum Öffnen des Ventils ausgibt, obwohl es sich bereits am oberen Anschlag befindet.

Handbetrieb

Wegen der fehlenden Information über die Stellung des Stellgliedes entfällt beim Schrittregler ohne Stellungsrückmeldung der Handbetrieb über den Parameter MAN bzw. den Handstellwertgeber MAN_GEN. Es ist nur der Handbetrieb durch direkte Schaltung der Stellsignale (LMNS_ON = TRUE) möglich.

Simulation der Stellungsrückmeldung

Steht keine Stellungsrückmeldung als messbare Größe zur Verfügung, so kann diese auch simuliert werden (LMNRS_ON = TRUE).

Die Nachbildung der Stellungsrückmeldung geschieht durch einen Integrator mit der Motorstellzeit MTR_TM als Integrationszeitkonstante. Im Zustand LMNRS_ON = FALSE wird am Ausgang des Integrators LMNR_ SIM der Startwert des Parameters LMNRSVAL ausgegeben. Nach dem Umschalten auf LMNRS_ON = TRUE läuft die Simulation mit diesem Startwert an.

Ist LMNR_HS = TRUE, wird die Integration nach oben angehalten, ist LMNR_LS = TRUE, wird sie nach unten angehalten. Ein Abgleich der simulierten Stellungsrückmeldung an den Endlagen erfolgt nicht.

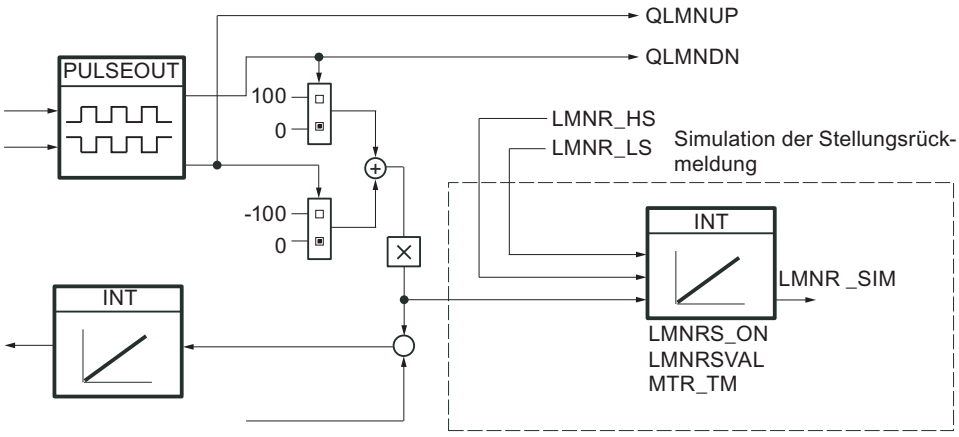

Hinweis

Die Stellungsrückmeldung wird lediglich simuliert. Sie muss nicht mit der wirklichen Position des Stellgliedes übereinstimmen. Ist eine reale Stellungsrückmeldung vorhanden, dann sollte diese in jedem Falle genutzt werden.

Handbetrieb

Handbetrieb

Die Struktur der Handwerterzeugung und Handwertaufschaltung geht aus nachfolgendem Bild hervor.

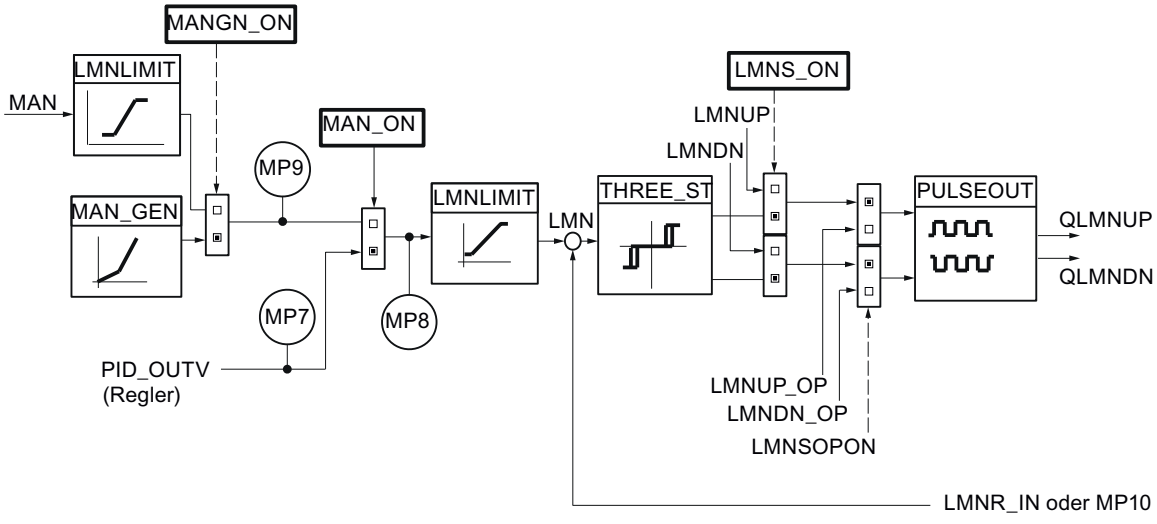

Beim Schrittregler gibt es drei Betriebsarten, in denen der Stellwert manuell beeinflusst werden kann:

- Handwert über den Eingangsparameter MAN
  Über den Parameter MAN (-100.0 % bis100.0 %) kann eine externe Stellgrößenbeeinflussung als Absolutwert aufgeschaltet oder vom Anwenderprogramm aus vorgegeben werden.
  Nur möglich, wenn die Stellungsrückmeldung aktiviert ist.
- Handbetrieb über Mehr–/Weniger–Schalter des Handwertgenerators (MANUP, MANDN),
  Wird MAN_GEN aus einer anderen Betriebsart eingeschaltet, dann wird der Stellwert MP9 übernommen. Die Umschaltung zum Handwertgenerator ist damit immer stoßfrei. Der Handstellwert kann innerhalb der Grenzen LMN_HLM und LMN_LLM vergrößert oder verkleinert werden.
  Nur möglich, wenn die Stellungsrückmeldung aktiviert ist.
- Handbetrieb durch direkte Schaltung der Stellsignale
  Der Handbetrieb durch direkte Schaltung der Stellsignale LMNUP und LMNDN hat immer Vorrang.
  Bei Rückschaltung LMNS_ON = FALSE wird eine beliebige eingeschaltete Betriebsart "stoßfrei" übernommen.

Automatikbetrieb

Ist der Handbetrieb ausgeschaltet (MAN_ON = FALSE), wird der Stellwert des PID–Algorithmus auf das Dreipunktglied geschaltet. Die Umschaltung von Handbetrieb auf Automatikbetrieb bewirkt einen Sprung des Stellwertes LMN. Dies wirkt sich jedoch nicht nachteilig aus, da der Prozess über das integrierend wirkende Stellglied rampenförmig angesteuert wird. Der Ausgang des PID–Algorithmus ist am Messpunkt MP7 hinterlegt.

Handwertnachführung im Automatikbetrieb

Wenn die Stellungsrückmeldung Peripherie aktiviert ist (LMNRP_ON = TRUE), wird der Eingangsparameter MAN der Stellung des Stellglieds (MP10) nachgeführt.

Wenn die Stellungsrückmeldung aktiviert ist (LMNRP_ON = FALSE), wird der Eingangsparameter MAN der Stellung des Stellglieds (LMNR_IN) nachgeführt.

Bei der Umschaltung auf Handbetrieb bleibt die Stellgröße daher auf dem Wert stehen, der der Stellung des Stellglieds entspricht. Sie kann nun per Bedienung verändert werden.

Handbetrieb mit Handwert aktivieren

In dieser Betriebsart (MANGN_ON = FALSE und MAN_ON = TRUE) wird der Handstellwert als Absolutwert am Eingang MAN eingegeben. Der Handstellwert wird am Messpunkt MP8 angezeigt.

1. Aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
2. Wählen Sie die Option "Handwert".

Handwertgenerator aktivieren

Der Handwert MAN kann über die Schalter MANUP und MANDN schrittweise vergrößert oder verkleinert werden. Der modifizierte Wert ist an MP8 zu beobachten. Die konfigurierten Begrenzungen des Stellwerts werden berücksichtigt.

1. Aktivieren Sie das Optionskästchen "Handbetrieb aktivieren".
2. Wählen Sie die Option "Handwertgenerator".

Wirkung der Schalter MANUP und MANDN

Für eine feinstufige Änderung sollte der Regler eine Abtastzeit von maximal 100 ms haben.

Die Änderungsgeschwindigkeit des Stellwerts hängt von der Betätigungsdauer der Schalter MANUP bzw. MANDN und von den eingestellten Begrenzungen nach folgender Beziehung ab:

- während der ersten 3 s nach Setzen von MANUP oder MANDN gilt:
  doutv/dt = (LMN_HLM - LMN_LLM) / 100 s
- danach
  doutv/dt = (LMN_HLM - LMN_LLM) / 10 s

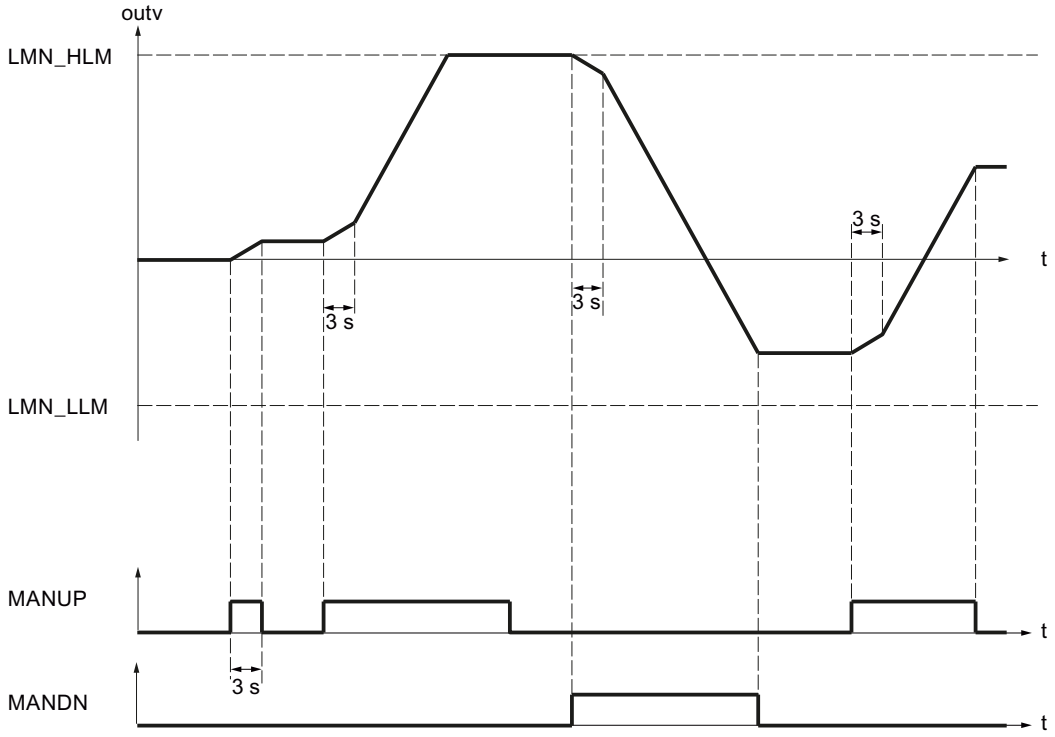

Bei einer Abtastzeit von 100 ms und einem Stellwertbereich von -100,0 bis 100,0 ändert sich z. B. der Stellwert in den ersten drei Sekunden um 0,2 pro Zyklus. Bleibt MANUP länger als 3 Sekunden TRUE, steigt die Änderungsgeschwindigkeit auf den zehnfachen Wert, hier also auf 2 pro Zyklus.

Der Handwertgenerator wird stoßfrei aktiviert.

Schaltung der Betriebsarten

Folgende Tabelle zeigt die möglichen Betriebsarten des Schrittreglers mit den erforderlichen Werten der strukturbestimmenden Schalter.

Tabelle 3-1 Betriebsarten des Schrittreglers

	MAN_ON	MNGN_ON	LMNS_ON
Automatikbetrieb	FALSE	beliebig	FALSE
Handbetrieb ohne Schalterbedienung	TRUE	FALSE	FALSE
Handbetrieb über Handwertgenerator	TRUE	TRUE	FALSE
Handbetrieb über Stellsignale	beliebig	beliebig	TRUE

Begrenzungen

Beim Schrittregler mit Stellungsrückmeldung können Sie den Stellwert begrenzen.

Absolutwert der Stellgröße begrenzen

Der Arbeitsbereich, d. h. der Bereich, in dem sich das Stellglied im Rahmen zulässiger Stellwerte bewegen kann, wird vom Einstellbereich des Stellwertes bestimmt. Da sich, abhängig von der Art des Stellgliedes, die Grenzen für erlaubte Stellwerte meist nicht mit der 0 %– oder der 100 %– Grenze des Stellbereiches decken, können Sie die Ober- und Untergrenze des Stellwerts konfigurieren.

Um im jeweiligen Prozess unzulässige Zustände zu vermeiden, wird im Stellwertzweig der Regelung der Einstellbereich für die Stellgröße nach oben und unten begrenzt.

Die Funktion LMNLIMIT begrenzt die Stellgröße LMN(t) auf den unteren und oberen Wert LMN_HLM und LMN_LLM. Diese Werte können vorgegeben werden. Die Eingangsgröße inv muss allerdings außerhalb dieser Begrenzungen liegen. Da die Funktion nicht abgeschaltet werden kann, muss die Angabe einer Untergrenze und Obergrenze bei der Konfiguration immer berücksichtigt werden.

Die Zahlenwerte der Grenzen (in Prozent) werden an den Eingangsparametern für die untere und obere Begrenzung eingestellt. Bei Überschreitungen durch die Eingangsgröße inv(t) werden die zugehörigen Anzeigen über die Meldeausgänge ausgegeben.

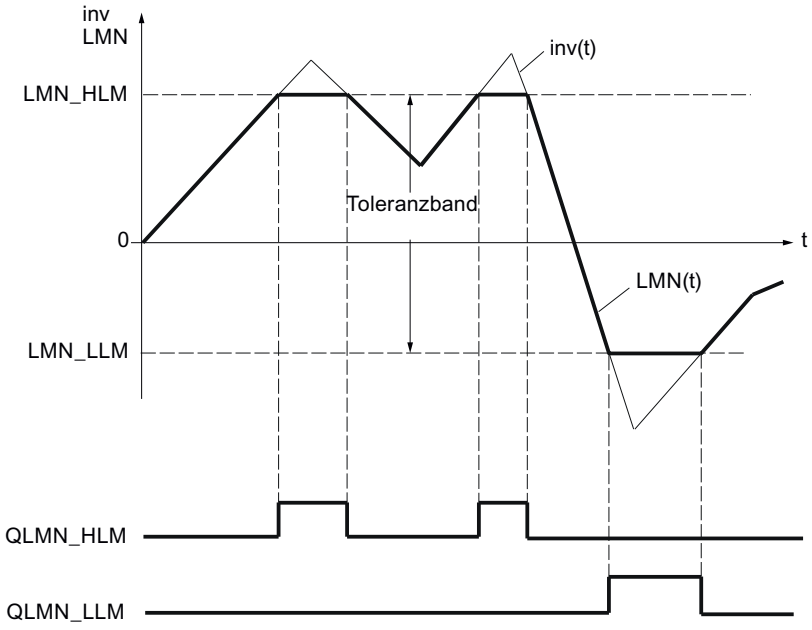

Anlauf und Betriebsweise

- Bei Neustart werden alle Meldeausgänge nullgesetzt.
- Die Begrenzung arbeitet gemäß folgenden Beziehungen:

LMN =	QLMN_HLM	QLMN_LLM	wenn:
LMN_HLM	TRUE	FALSE	inv ≥ LMN_HLM
LMN_LLM	FALSE	TRUE	inv ≤ LMN_LLM
inv	FALSE	FALSE	LMN_HLM < inv < LMN_LLM

Am Ausgang d. h. am Parameter LMN bzw. am Messpunkt MP10 wird der wirksame Stellwert der Regelung angezeigt.

Vorgehen

Um den Absolutwert der Stellgröße zu begrenzen, gehen Sie folgendermaßen vor:

1. Geben Sie in das Eingabefeld "Obergrenze" den Wert für den höchsten zulässigen Wert ein.
2. Geben Sie in das Eingabefeld "Untergrenze" den Wert für den niedrigsten zulässigen Wert ein.

Pulsgenerator

Der Pulsgenerator sorgt dafür, dass eine Minimale Ein- und Ausschaltzeit eingehalten wird, um die Stellglieder zu schonen. Ein Ansprechen der Anschlagsignale des Stellgliedes (LMNR_HS/LMNR_LS) verursacht eine Sperrung des Stellwerts LMNUP bzw. LMNDN. Das gilt für den Automatik- und den Handbetrieb.

Sind beide Signalschalter zur Stellwertsignalbedienung gesetzt (LMNUP = LMNDN = TRUE bzw. LMNUP_OP = LMNDN_OP = TRUE), wird an den Ausgängen QLMNUP und QLMNDN immer FALSE ausgegeben.

Der direkte Wechsel von "Stellwertsignal auf" (QLMNUP = TRUE, QLMNDN = FALSE) nach "Stellwertsignal zu" (QLMNUP = FALSE, QLMNDN = TRUE) ist nicht möglich. Der Pulsgenerator fügt einen Zyklus mit QLMNUP = QLMNDN = FALSE ein.

Eingangssignal INV beim Schrittregler mit Stellungsrückmeldung

Die Differenz zwischen Stellwert LMN und Stellungsrückmeldung LMNR wird auf das Dreipunktglied mit Hysterese THREE_ST geschaltet.

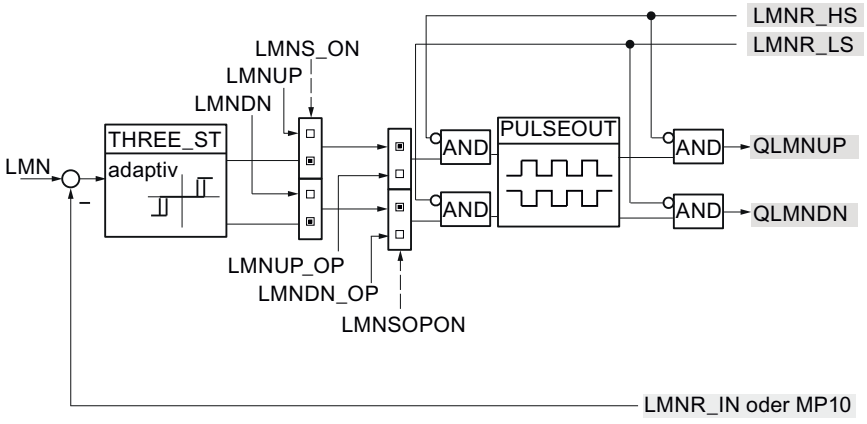

Eingangssignal INV beim Schrittregler ohne Stellungsrückmeldung

Die Differenz zwischen dem PD–Anteil des Reglers und der Rückführgröße (MP11) wird auf das Dreipunktglied mit Hysterese THREE_ST geschaltet.

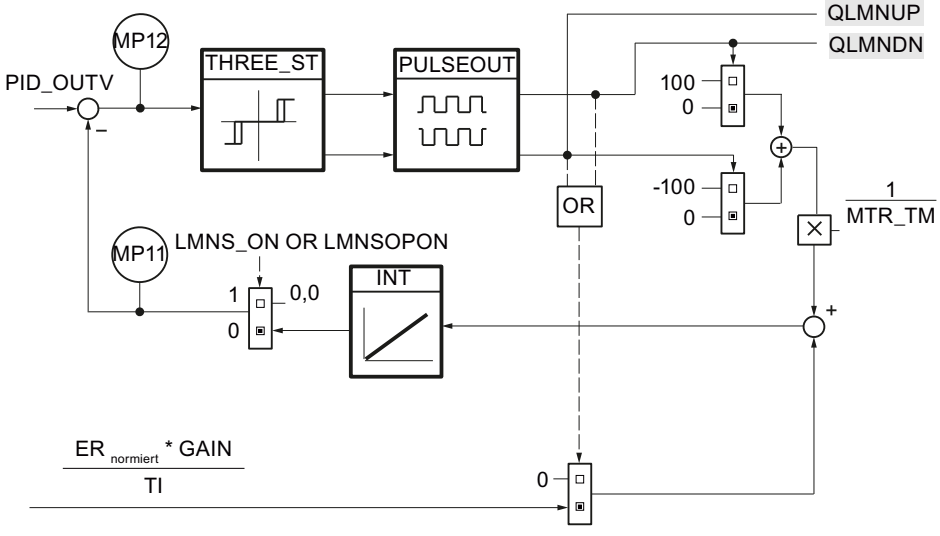

Einfluss der Motorstellzeit auf das Dreipunktglied

Das Dreipunktglied erzeugt zwei Binärsignale, die je nach Größe und Vorzeichen des Differenzwertes am Eingang TRUE oder FALSE sind.

Das Dreipunktglied THREE_ST reagiert auf das Eingangssignal INV entsprechend folgender Beziehungen und nimmt dabei jeweils eine der drei möglichen Kombinationen seiner Ausgangssignale UP/DOWN an:

UP	DOWN	Eingangskombination
TRUE	FALSE	|INV| ≥ ThrOn oder INV > ThrOff und UP_{alt} = TRUE
FALSE	TRUE	|INV| ≤ -ThrOn oder INV < -ThrOff und $DOWN_{alt}$ = TRUE
FALSE	FALSE	|INV| ≤ ThrOff

ThrOn = Einschaltschwelle, ThrOff = Ausschaltschwelle

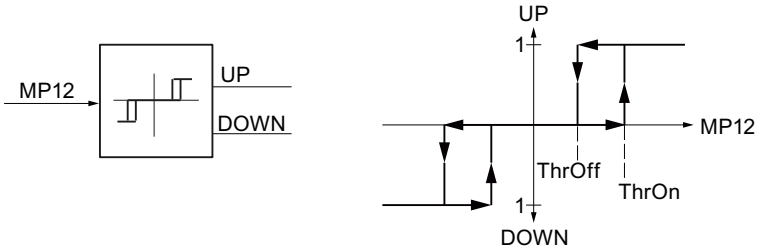

Die Abschaltschwelle ThrOff muss größer sein als die Änderung der Stellungsrückmeldung oder die Positionsänderung nach einer Impulsdauer. Dieser Wert ist abhängig von der Stellzeit des Motors MTR_TM und wird wie folgt berechnet:

$$\text{ThrOff} = 0{,}5 * \frac{110}{\text{MTR_TM}} * \text{CYCLE}$$

MTR_TM muss ein ganzzahliges Vielfaches von CYCLE sein.

Hinweis

Eine zu groß eingestellte Motorstellzeit (10% über der realen Stellzeit) bewirkt ein ständiges Ein- und Ausschalten der Stellwertsignale QLMNUP und QLMNDN.

Adaption der Einschaltschwelle ThrOn

Um bei der Ausregelung größerer Regeldifferenzen die Schalthäufigkeit zu reduzieren, wird die Einschaltschwelle ThrOn während des Betriebes automatisch angepasst, während ThrOff konstant bleibt. ThrOn ist begrenzt auf:

$$\text{Min ThrOn} = \frac{100}{\text{MTR_TM}} * \text{MAX (PULSE_TM, CYCLE)}$$

$$\text{Max ThrOn} = 10$$

Die Adaption der Einschaltschwelle ist bei reinen P–, D– oder PD–Reglern abgeschaltet. Hierbei ist:

ThrOn = Min ThrOn

Minimale Einschaltzeit und minimale Ausschaltzeit

Der Pulsgenerator sorgt dafür, dass beim Ein- und Ausschalten der Ausgangsimpulse jeweils die minimale Ein- und Ausschaltzeit eingehalten wird.

Zur Schonung des Stellgliedes können deshalb eine minimale Einschaltzeit PULSE_TM und eine minimale Ausschaltzeit BREAK_TM parametriert werden. Die Dauer der Ausgangsimpulse QLMNUP bzw. QLMNDN ist immer größer als die minimale Einschaltzeit und die Pause zwischen zwei Impulsen immer größer als die minimale Ausschaltzeit. Das nachfolgende Bild zeigt die Arbeitsweise am Beispiel des UP–Signals.

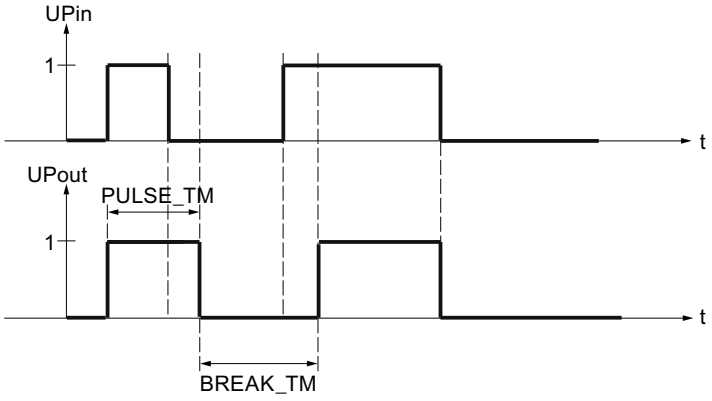

Vorgehen

Um den Pulsgenerator zu konfigurieren, gehen Sie folgendermaßen vor:

1. Geben Sie die Stellzeit des Motors in das Eingabefeld "Motorstellzeit" ein.
2. Geben Sie "Minimale Einschaltzeit" ein.
3. Geben Sie die "Minimale Ausschaltzeit" ein.

PID_ES in Betrieb nehmen

Manuelle Optimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- MAN_ON = FALSE

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Geben Sie in den Feldern "P", "I", "D" und "Verzögerungszeit" neue PID-Parameter ein.
3. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an CPU senden".
4. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
5. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
   Wenn die Änderungsgeschwindigkeit des Sollwerts begrenzt ist, wird stoßfrei auf den neuen Sollwert umgeschaltet.
6. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.
7. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
8. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

Inbetriebnahme mit dem PID Self Tuner

Die Anweisung PID_ES können Sie mit dem PID Self-Tuner in Betrieb nehmen und die optimalen PID-Parameter ermitteln.

Vorgehen

Um einen Schrittregler in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Rufen Sie die Anweisung PID_ES in einem Weckalarm-OB auf und verschalten Sie die Ein- und Ausgangsparameter.
2. Rufen Sie die Anweisung TUN_ES im selben Weckalarm-OB auf.

3. Verschalten Sie die Anweisung TUN_ES folgendermaßen mit PID_ES:
   - TUN_ES.PV mit PID_ES.PV
   - Wenn Sie eine Stellungsrückmeldung einsetzen TUN_ES.LMNR mit PID_ES.LMNR_IN
   - TUN_ES.C_LMNUP und TUN_ES.C_LMNDN mit PID_ES.QLMN_UP und PID_ES.QLMNDN
   - TUN_ES.LMNR_HS mit PID_ES.LMNR_HS
   - TUN_ES.LMNR_ON mit PID_ES.LMNR_ON
   - TUN_ES.LMNS_ON mit PID_ES.LMNS_ON
   - TUN_ES.SP_OUT mit PID_ES.SP_IN oder PID_ES.SP_EXT
   - TUN_ES.GAIN mit PID_ES.GAIN
   - TUN_ES.TI mit PID_ES.TI
   - TUN_ES.TD mit PID_ES.TD
   - TUN_ES.TM_LAG mit PID_ES.TM_LAG
   - TUN_ES.MTR_TM mit PID_ES.MTR_TM
   - Wenn Sie eine Totzone einsetzen, TUN_ES.DEADB_W mit PID_ES.DEADB_W
   - TUN_ES.QLMNUP und TUN_ES.QLMNDN mit PID_ES.LMNUP und PID_ES.LMNDN
4. Klicken Sie auf das Symbol an der Anweisung TUN_ES.
   Der Inbetriebnahmeeditor für TUN_ES wird geöffnet. Die Bedienung des Inbetriebnahmeeditors ist beim PID Self-Tuner beschrieben.

3.5.2 Modular PID Control einsetzen

3.5.2.1 Funktionsumfang des Modular PID Control

Bei vielen Regelungsaufgaben steht nicht allein der klassische PID–Regler als Prozess beeinflussendes Element im Vordergrund, sondern es werden auch hohe Anforderungen an die Signalverarbeitung gestellt.

Modular PID Control enthält Anweisungen mit Funktionen zur Aufbereitung des Ist- und Sollwerts, zur Berechnung und Nachbearbeitung des Stellwerts.

Folgende Regler sind möglich:

- Kontinuierliche Regler
- Schrittregler für integrierende Stellglieder
- Impulsregler

Durch Verschalten der einzelnen Anweisungen realisieren Sie Regler für unterschiedliche Anwendungen:

- Festwertregler
- Kaskadenregler

- Verhältnisregler
- Mischungsregler
- Splitrange–Regler
- Override–Regler
- Regler mit Störgrößenverarbeitung
- Mehrgrößenregler

Es können sowohl träge Regelstrecken (Temperatur, Füllstände usw.) als auch sehr schnelle Regelstrecken (Durchflüsse, Druck usw.) geregelt werden.

Mit der Anweisung LP_SCHED_M können Sie viele Regelkreise äquidistant aber mit unterschiedlichen Abtastzeiten aufrufen. Die Abtastzeit des Regelkreises variieren Sie je nach Trägheit der jeweiligen Regelstrecke.

So wird eine gleichmäßige Auslastung der CPU gewährleistet.

Zur Inbetriebnahme und zur Optimierung der PID-Parameter verwenden Sie den PID Self-Tuner.

3.5.2.2 Grundsätzliche Schritte zum Aufbau eines Regler-FBs

Grundsätzliches Vorgehen zum Programmieren eines Reglers

Sie programmieren einen Regler z. B. mit AWL als Funktionsbaustein (Regler-FB). Ihr Regler-FB muss mindestens folgende Parameter besitzen:

- Eingangsparameter COM_RST
- Eingangsparameter CYCLE
- Eingangsparameter für Sollwert
- Eingangsparameter für Istwert
- Ausgangsparameter für den Stellwert
- Durchgangsparameter für den Handstellwert (Kontinuierlicher Regler)

Zusätzlich können Sie je nach Anforderung an die Regelung folgende Parameter anlegen

- Eingangsparameter für Stellungsrückmeldung (Schrittregler)
- Eingangsparameter für eine Störgröße
- Ausgangsparameter für den aufbereiteten Istwert
- Ausgangsparameter für den aufbereiteten Sollwert
- Ausgangsparameter für das Überschreiten von Grenzen

Wie Sie die Ein- und Ausgänge detailliert gestalten, hängt von Ihren spezifischen Gegebenheiten ab, z. B. ob der Istwert im Peripherie- oder Gleitpunktformat vorliegt, welche Stellungsrückmeldung es gibt usw.

Im Regler-FB rufen Sie die Anweisungen von Modular PID Control auf und verschalten die Ein- und Ausgangsparameter. Den so erstellten Regler–FB rufen Sie z.B. im OB 100 oder in einem Weckalarm-OB auf.

Um mit Modular PID Control einen Regler zu programmieren, sind die folgenden Schritte notwendig:

Schritt	Beschreibung
1.	Sollwert aufbereiten (Seite 932)
2.	Istwert aufbereiten (Seite 934)
3.	Regeldifferenz bilden und aufbereiten (Seite 936)
4.	Störgröße aufschalten (Seite 938) (optional)
5.	Stellwert berechnen • Kontinuierlichen Regler erstellen (Seite 940) • Impulsregler erstellen (Seite 942) • Schrittregler erstellen (Seite 945)
6.	Stellungsrückmeldung aufbereiten (Seite 947)

Siehe auch

Sollwert aufbereiten (Seite 932)
Istwert aufbereiten (Seite 934)
Regeldifferenz bilden und aufbereiten (Seite 936)
Störgröße aufschalten (Seite 938)
Kontinuierlichen Regler erstellen (Seite 940)
Impulsregler erstellen (Seite 942)
Schrittregler erstellen (Seite 945)
Stellungsrückmeldung aufbereiten (Seite 947)

3.5.2.3 Regler-FB programmieren

Sollwert aufbereiten

Anweisungen für die Aufbereitung des Sollwerts

Folgende Anweisungen können Sie einsetzen.

- SP_GEN: Sollwert manuell vorgeben
- ROC_LIM: Änderungsgeschwindigkeit begrenzen
- LAG1ST: Sollwert glätten
- NONLIN: Sollwert linearisieren
- LIMITER: Sollwert begrenzen
- RMP_SOAK: Sollwert nach Zeitplan vorgeben

- SCALE_M: Skalieren des Sollwerts
- NORM: Normieren des Sollwerts

Beispiel für die Aufbereitung des Sollwerts

Ein vorgegebener Sollwert soll von Hand vergrößert oder verkleinert werden können. Um sprunghafte Änderungen des Sollwerts zu vermeiden, wird die Änderungsgeschwindigkeit begrenzt.

Um den Sollwert wie im Beispiel beschrieben aufzubereiten, gehen Sie folgendermaßen vor:

1. Deklarieren Sie am Regler-FB, hier PIDCTR, einen Eingangsparameter SP_IN vom Datentyp "Real".
2. Verschalten Sie SP_IN mit einer Variablen, die den gewünschten Sollwert enthält.
3. Rufen Sie im Regler-FB die Anweisung SP_GEN als Multiinstanz auf und verschalten Sie SP_IN mit SP_GEN.DF_OUTV.
4. Parametrieren Sie in der Struktur SP_GEN_Instance der Multiinstanz Ober- und Untergrenze des Sollwerts.
5. Rufen Sie im Regler-FB die Anweisung ROC_LIM als Multiinstanz auf und verschalten Sie SP_GEN.OUTV über eine temporäre Variable mit ROC_LIM.INV.
6. Parametrieren Sie in der Struktur ROC_LIM_Instance der Multiinstanz die maximalen Änderungsgeschwindigkeiten.
7. Um die Regeldifferenz zu bilden, subtrahieren Sie von ROC_LIM.OUTV den aufbereiteten Istwert oder verschalten Sie ROC_LIM.OUTV mit ERR_MON.SP.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR aufgerufen und verschaltet werden.

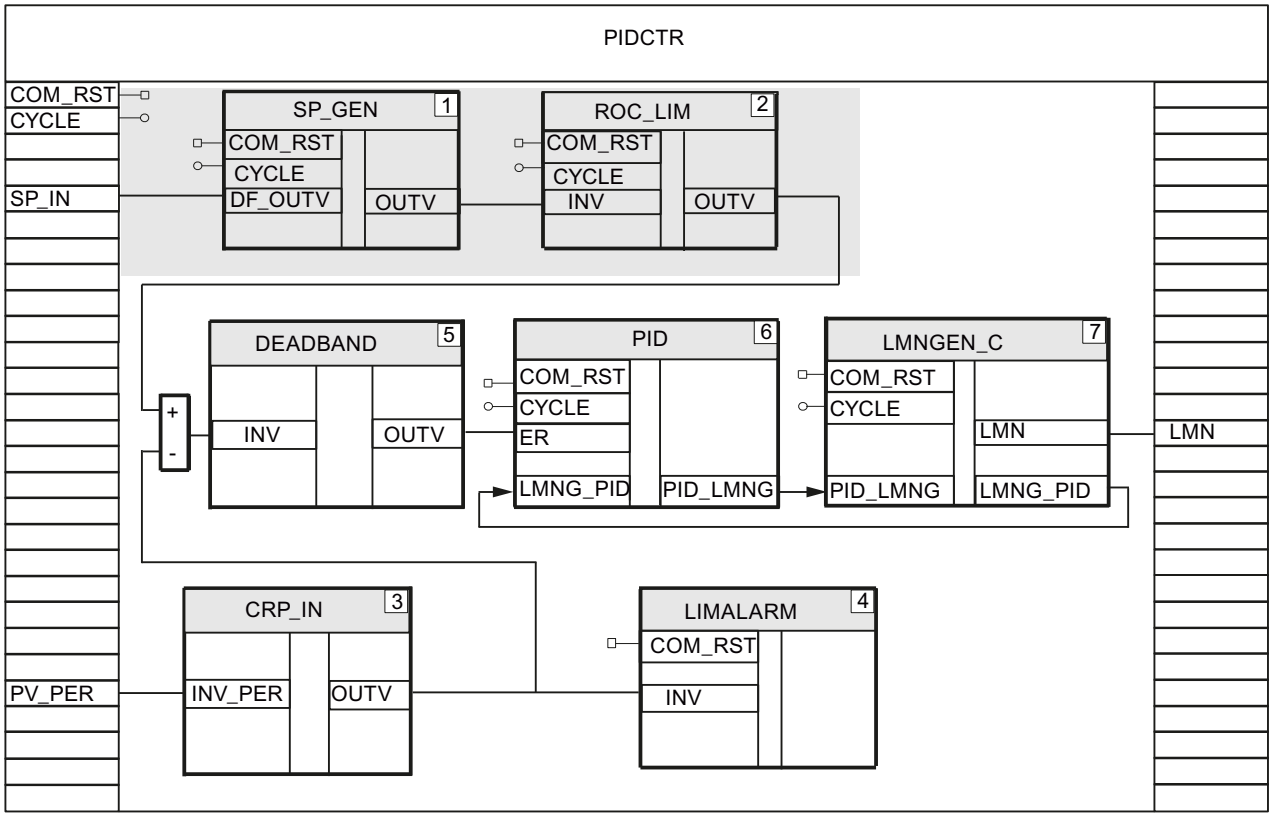

Siehe auch

Istwert aufbereiten (Seite 934)

Istwert aufbereiten

Anweisungen für die Aufbereitung des Istwerts

Folgende Anweisungen können Sie einsetzen.

- CRP_IN: Peripheriewert skalieren
- LAG1ST: Istwert glätten
- LIMITER: Istwert begrenzen
- LIMALARM: Istwert überwachen
- NONLIN: Istwert linearisieren
- SCALE_M: Skalieren des Istwerts
- NORM: Normieren des Istwerts

Beispiel für eine Aufbereitung des Istwerts

Der Istwert liegt im Perpherieformat vor und soll auf die Einhaltung der Istwertgrenzen überwacht werden.

Um den Istwert wie im Beispiel beschrieben aufzubereiten, gehen Sie folgendermaßen vor:

1. Deklarieren Sie am Regler-FB, hier PIDCTR, einen Eingangsparameter PV_PER vom Datentyp "Word".
2. Verschalten Sie PV_PER mit dem Peripherie-Eingang, an dem der Istwert gemessen wird.
3. Rufen Sie im Regler-FB die Anweisung CRP_IN als Multiinstanz auf, z. B. mit dem Namen CRP_IN_Instance_PV.
4. Verschalten Sie PV_PER mit CRP_IN.INV_PER.
5. Parametrieren Sie in der Struktur CRP_IN_Instance_PV der Multiinstanz Faktor und Offset für die Skalierung.
6. Rufen Sie im Regler-FB die Anweisung LIMALARM als Multiinstanz auf und verschalten Sie CRP_IN.OUTV über eine temporäre Variable mit LIMALARM.INV.
7. Parametrieren Sie in der Struktur LIMALARM_Instance der Multiinstanz die Istwertgrenzen.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR aufgerufen und verschaltet werden.

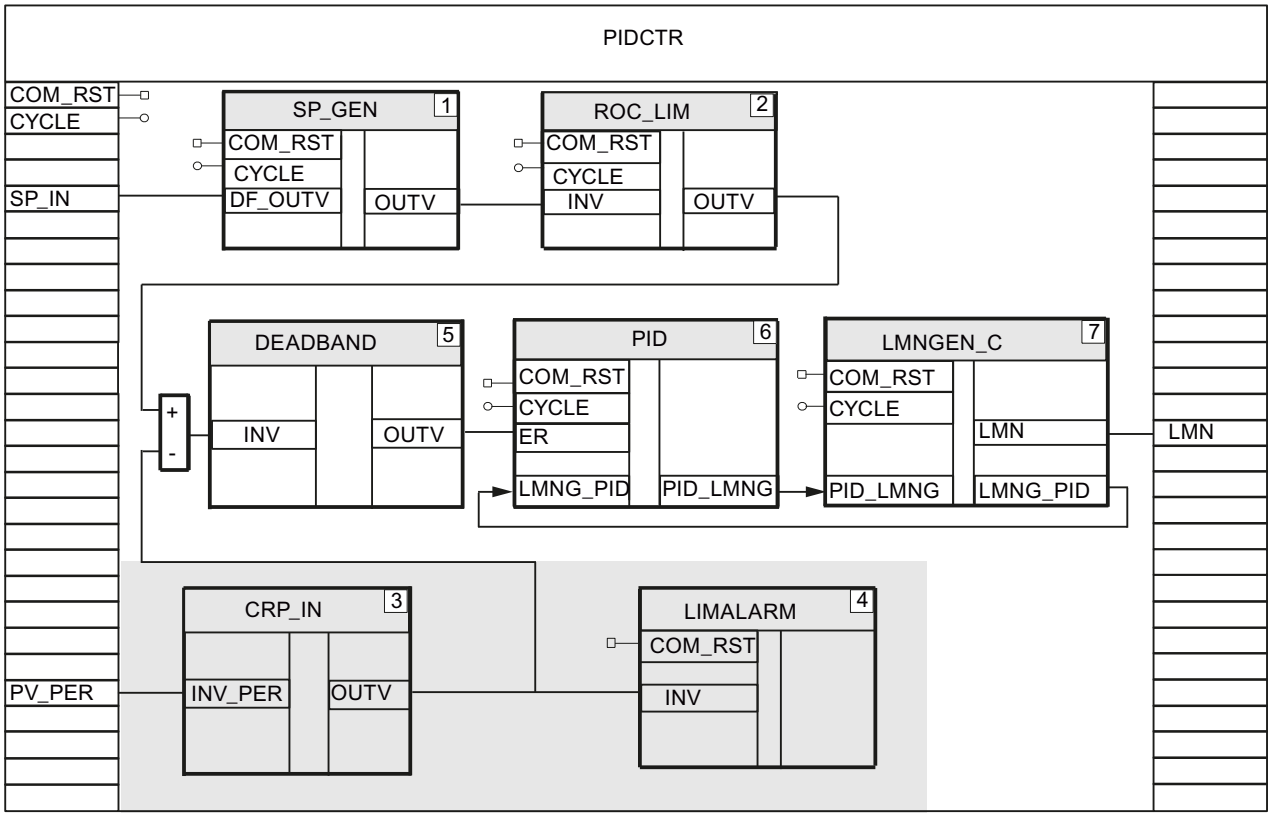

Siehe auch

Sollwert aufbereiten (Seite 932)

Regeldifferenz bilden und aufbereiten (Seite 936)

Regeldifferenz bilden und aufbereiten

Regeldifferenz ermitteln und begrenzen

Die Regeldifferenz können Sie auf zwei Arten ermitteln. Entweder programmieren Sie die Subtraktion des Istwerts vom Sollwert, oder Sie verwenden die Anweisung ERR_MON. In der Anweisung ERR_MON ist bereits eine Überwachung der Regeldifferenz enthalten.

Anweisungen für die Aufbereitung der Regeldifferenz

Folgende Anweisungen können Sie einsetzen.

- A_DEAD_B: Regeldifferenz über adaptive Totzone leiten
- DEADBAND: Regeldifferenz über Totzone leiten

Beispiel für die Subtraktion

Die Regeldifferenz wird durch Subtraktion gebildet und über eine Totzone geleitet.

Um die Regeldifferenz wie im Beispiel beschrieben zu ermittelten, gehen Sie folgendermaßen vor:

1. Rufen Sie die Anweisung DEADBAND als Multiinstanz auf und verschalten Sie DEADBAND.INV mit der Differenz von Sollwert und Istwert, z. B. ROC_LIM.OUTV - CRP_IN.OUTV.
2. Parametrieren Sie in der Struktur DEADBAND_Instance der Multiinstanz die Totzonenbreite und einen Totzonenversatz.
   An DEADBAND.OUTV wird die aufbereitete Regeldifferenz ausgegeben.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR aufgerufen und verschaltet werden.

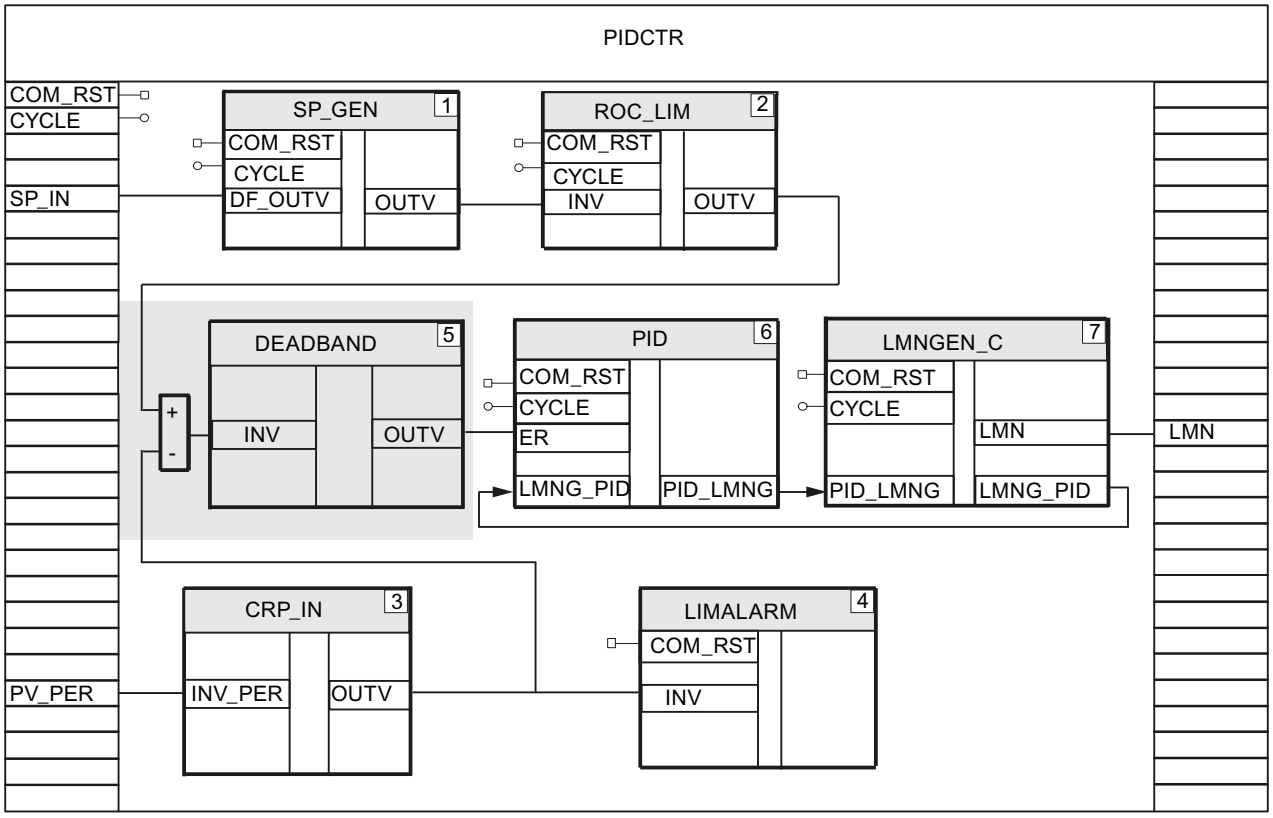

Beispiel für den Einsatz von ERR_MON

Die Regeldifferenz wird mit der Anweisung ERR_MON berechnet und überwacht.

Um die Regeldifferenz wie im Beispiel beschrieben zu ermittelten, gehen Sie folgendermaßen vor:

1. Rufen Sie die Anweisung ERR_MON als Multiinstanz auf.
2. Verschalten Sie über temporäre Variablen den aufbereiteten Sollwert mit ERR_MON.SP, z. B. ROC_LIM.OUTV, und den aufbereiteten Istwert mit ERR_MON.PV, z. B. CRP_IN.OUTV.
   An ERR_MON.ER wird die Regeldifferenz ausgegeben.
3. Parametrieren Sie in der Struktur ERR_MON_Instance der Multiinstanz die Überwachung der Regeldifferenz.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR_ER aufgerufen und verschaltet werden.

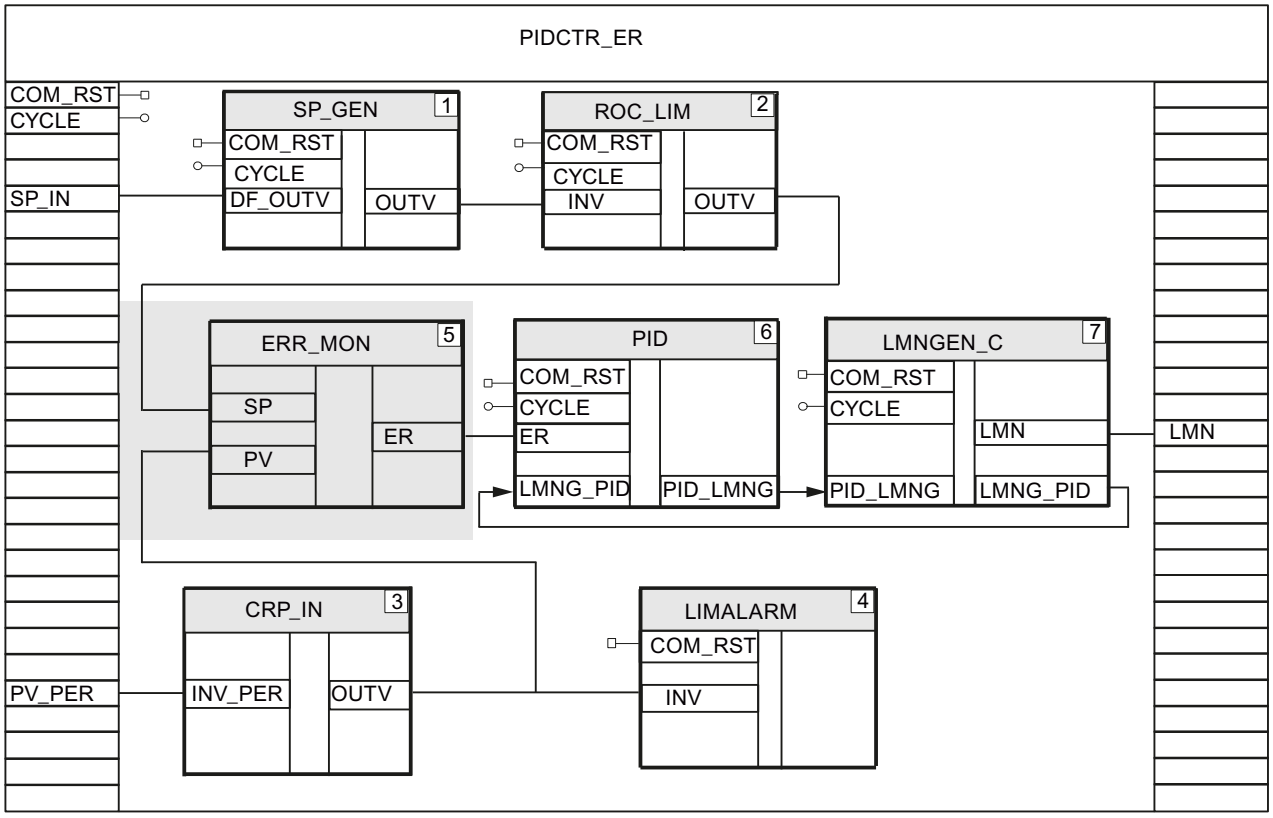

Siehe auch

Istwert aufbereiten (Seite 934)

Störgröße aufschalten (Seite 938)

Störgröße aufschalten

Anweisungen für die Aufbereitung der Störgröße

Folgende Anweisungen können Sie einsetzen.

- CRP_IN: Peripheriewert skalieren
- DIF: Auf schnelle Änderungen reagieren
- LAG1ST: Störungen filtern
- NON_LIN: Störgröße linearisieren
- SCALE_M: Störgröße skalieren
- NORM: Störgröße normieren

Beispiel für eine Aufbereitung der Störgröße

Die Störgröße liegt im Peripherieformat vor, das Signal soll geglättet werden und über eine Kennlinie linearisiert werden.

Um die Störgröße wie im Beispiel beschrieben aufzubereiten, gehen Sie folgendermaßen vor:

1. Deklarieren Sie am Regler-FB einen Eingangsparameter DISV_PER vom Datentyp "Word".
2. Verschalten Sie DISV_PER mit dem Peripherie-Eingang, an dem die Störgröße gemessen wird.
3. Rufen Sie im Regler-FB die Anweisung CRP_IN als Multiinstanz auf, z. B. mit dem Namen CRP_IN_Instance_DISV.
4. Verschalten Sie DISV_PER mit CRP_IN.INV_PER.
5. Parametrieren Sie in der Struktur CRP_IN_Instance_DISV der Multiinstanz Faktor und Offset für die Skalierung.
6. Rufen Sie im Regler-FB die Anweisung LAG1ST als Multiinstanz auf und verschalten Sie CRP_IN.OUTV über eine temporäre Variable mit LAG1ST.INV.
7. Parametrieren Sie in der Struktur LAG1ST_Instance der Multiinstanz die Verzögerungszeit.
8. Rufen Sie im Regler-FB die Anweisung NONLIN als Multiinstanz auf und verschalten Sie LAG1ST.OUTV über eine temporäre Variable mit NONLIN.INV.
9. Legen Sie einen AUTOHOTSPOT an, in dem Sie die Stützpunkte der Kennlinie speichern.
10. Parametrieren Sie in der Struktur NONLIN_Instance der Multiinstanz die Nummer des globalen Datenbausteins mit der Kennlinie.
11. An NONLIN.OUTV wird die aufbereitete Störgröße ausgegeben.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB CTR_C_FF aufgerufen und verschaltet werden.

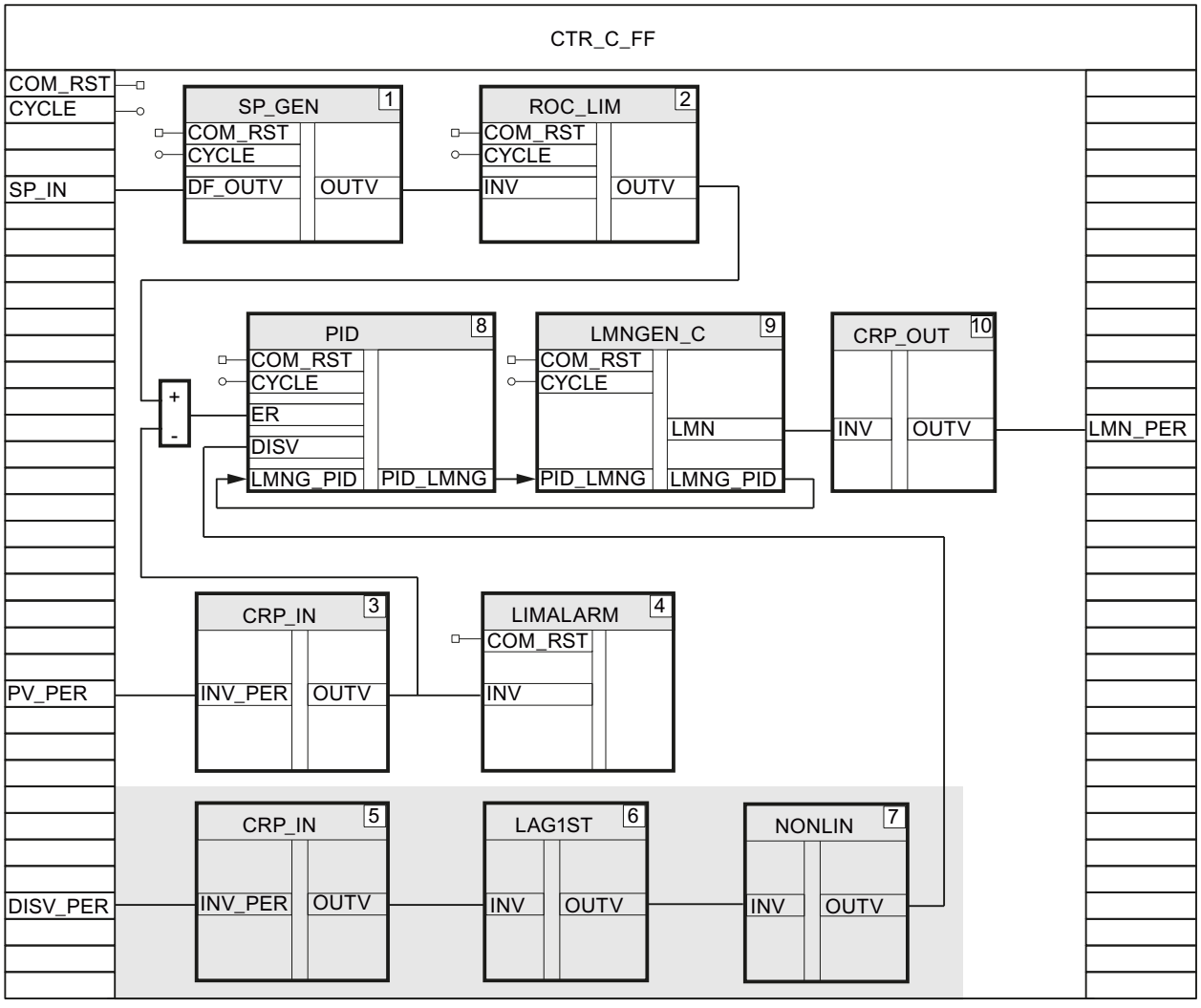

Siehe auch

Regeldifferenz bilden und aufbereiten (Seite 936)

Kontinuierlichen Regler erstellen (Seite 940)

Impulsregler erstellen (Seite 942)

Schrittregler erstellen (Seite 945)

Kontinuierlichen Regler erstellen

Für einen kontinuierlichen Regler mit analogem Stellwert benötigen Sie die Anweisungen PID, LMNGEN_C und CRP_OUT.

Vorgehen

Um einen kontinuierlichen Regler mit analogem Stellwert zu programmieren, gehen Sie folgendermaßen vor:

1. Deklarieren Sie am Regler-FB einen Durchgangsparameter MAN vom Datentyp "Real". MAN benötigen Sie für die Inbetriebnahme des Reglers mit den PID Self-Tuner.
2. Rufen Sie im Regler-FB die Anweisungen PID, LMNGEN_C und CRP_OUT als Multiinstanzen auf.
3. Deklarieren Sie am Regler-FB einen Durchgangsparameter MAN vom Datentyp "Real" und verschalten Sie MAN mit LMNGEN_C.MAN.
4. Parametrieren Sie im Regler-DB in der Struktur PID_Instance das Verhalten des Regelalgorithmus, z. B. ob der D-Anteil verwendet wird.
5. Parametrieren Sie in der Struktur LMNGEN_C_Instance der Multiinstanz z. B. die Grenzen für den Stellwert und für die Änderungsgeschwindigkeit des Stellwerts.
6. Parametrieren Sie in der Struktur CRP_OUT_Instance der Multiinstanz den Faktor und den Offset für die Umwandlung des Gleitpunktwerts in das Peripherieformat.
7. Verschalten Sie über eine temporäre Variable PID.ER mit der Regeldifferenz, z. B. ROC_LIM.OUTV - CRP_IN.OUTV oder ERR_MON.ER.
8. Wenn Sie eine Störgröße aufschalten, verschalten Sie PID.DISV über eine temporäre Variable mit dem aufbereiteten Signal der Störgröße, z. B. NONLIN.OUTV.
9. Verschalten Sie über die Anweisung BLKMOV den Ausgangsparameter PID.PID_LMNG mit dem Eingangsparameter LMNGEN_C.PID_LMNG und den Ausgangsparameter LMNGEN_C.LMNG_PID mit dem Eingangsparameter PID.LMNG_PID.
10. Verschalten Sie LMNGEN_C.LMN über eine temporäre Variable mit CRP_OUT.INV.
11. Deklarieren Sie am Regler-FB einen Ausgangsparameter LMN_PER vom Datentyp "Word" und verschalten Sie CRP_OUT.OUTV mit LMN_PER.
12. Verschalten Sie LMN_PER mit dem Analogausgang der CPU, der das Stellglied steuert.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB CTR_C_FF aufgerufen und verschaltet werden.

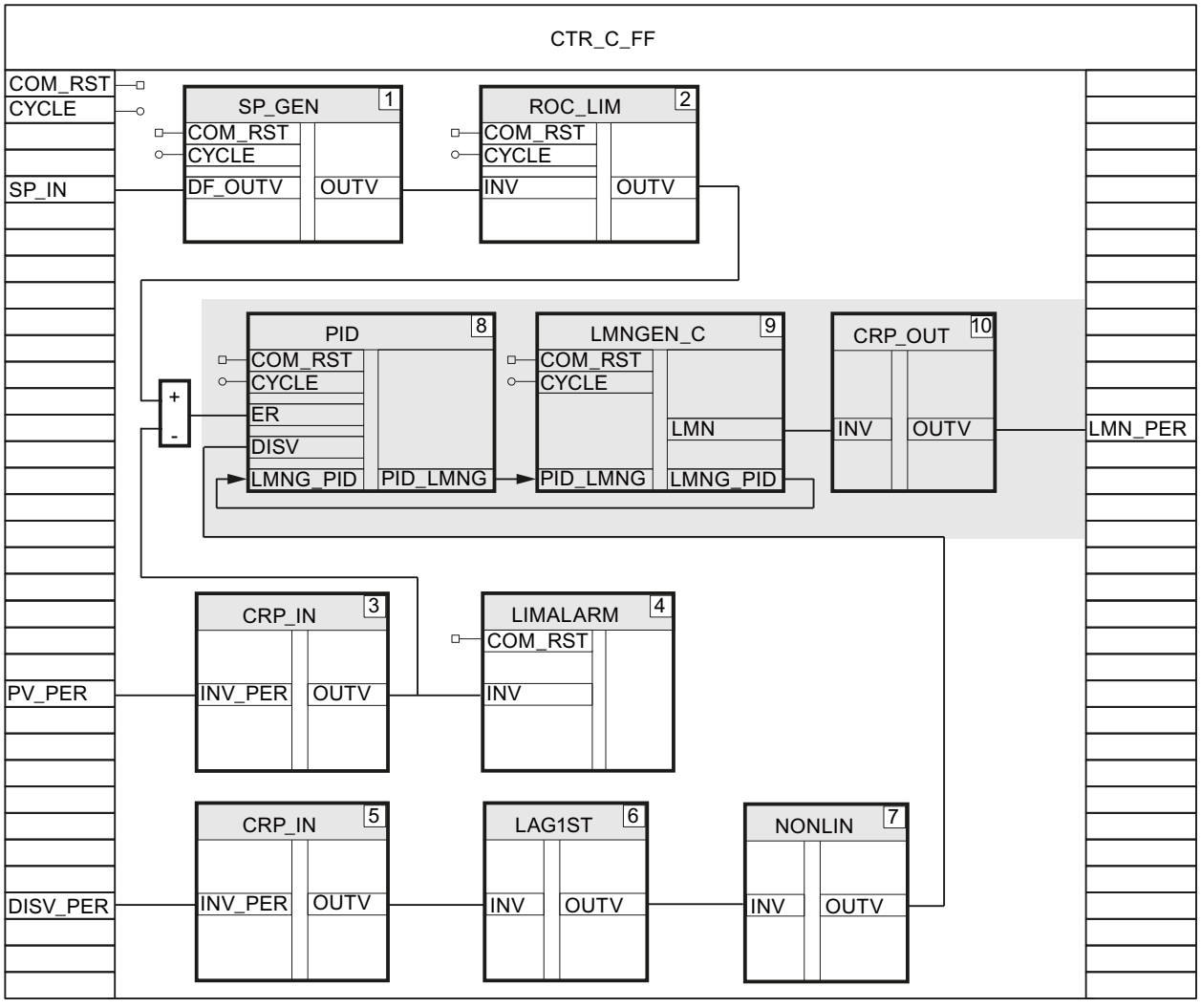

Siehe auch

Störgröße aufschalten (Seite 938)

Impulsregler erstellen

Für einen Impulsregler erzeugen Sie mit den Anweisungen PID und LMNGEN_C einen kontinuierlichen Stellwert im Gleitpunktformat. Aus diesem kontinuierlichen Stellwert generieren Sie mit der Anweisung PULSEGEN_M durch Pulsweitenmodulation eine Impulsfolge.

Kontinuierlichen Stellwert im Gleitpunktformat erzeugen

1. Deklarieren Sie am Regler-FB, hier PIDCTR, einen Ausgangsparameter LMN vom Datentyp "Real".
2. Rufen Sie im Regler-FB die Anweisungen PID und LMNGEN_C als Multiinstanz auf.

3. Verschalten und parametrieren Sie PID und LMNGEN_C wie beim kontinuierlichen Regler beschrieben.
4. Verschalten Sie LMNGEN_C.LMN mit dem Ausgangsparameter LMN des Regler-FBs.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR aufgerufen und verschaltet werden.

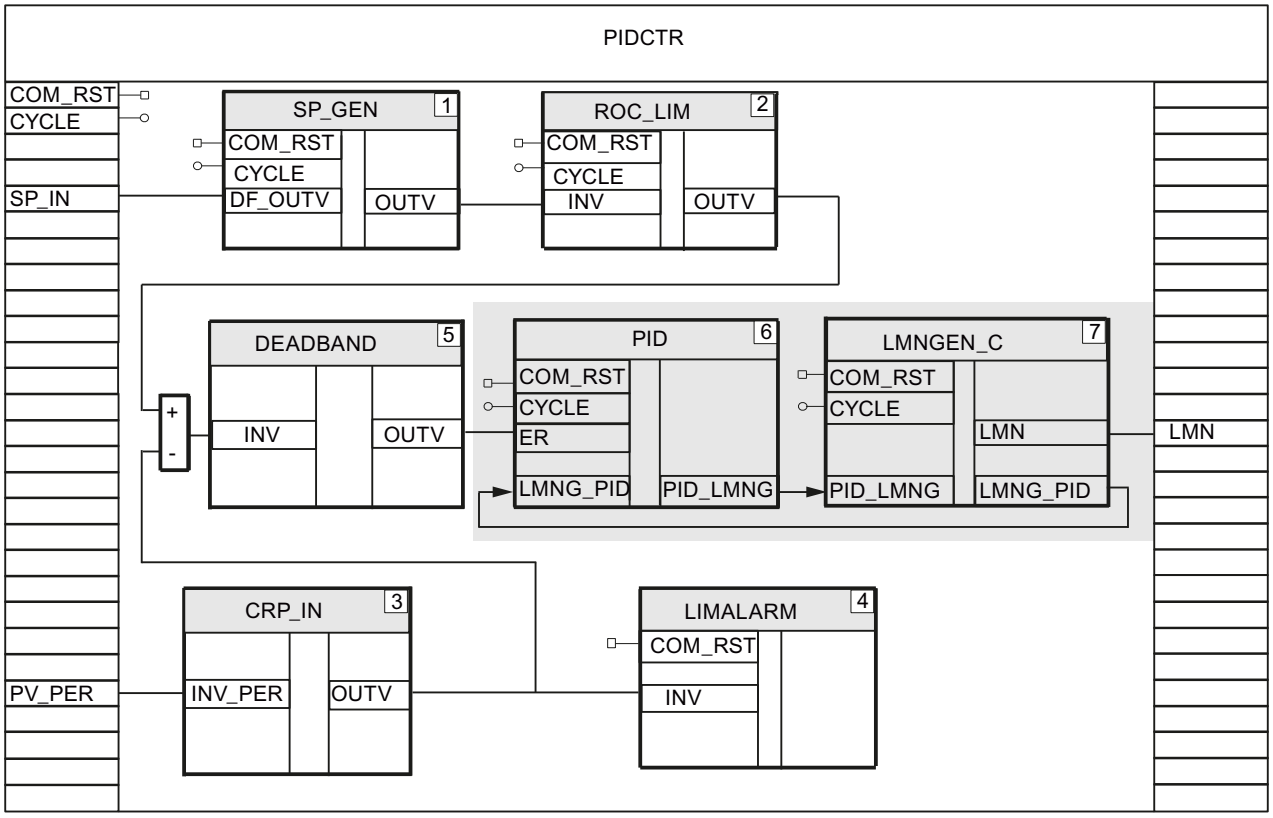

Impulsfolge generieren

Um aus dem kontinuierlichen Stellwert eine Impulsfolge zu generieren, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine Funktion, z. B. FC50, mit den Eingangsparametern COM_RST und CYCLE.
2. Legen Sie einen AUTOHOTSPOT an, in dem Sie die Zykluszeiten für den Aufruf des Regler-FBs und des Pulsgenerators PULSEGEN_M parametrieren.
   Das Verhältnis von Zykluszeit für den Aufruf des Regler-FBs und PULSEGEN_M sollte ≥ 20 sein.
3. Rufen Sie in der Funktion die Anweisung LP_SCHED_M auf.
4. Verschalten Sie LP_SCHED_M.DB_NBR mit dem globalen Datenbaustein DB_LOOP.
5. Rufen Sie ihren Regler-FB auf, z. B. PIDCTR, wenn DB_LOOP.LOOP_DAT[1].ENABLE = TRUE ist.

6. Verschalten Sie DB_LOOP.LOOP_DAT[1].COM_RST und DB_LOOP.LOOP_DAT[1].CYCLE mit den Eingangsparametern COM_RST und CYCLE des Regler-FBs.
7. Setzen Sie DB_LOOP.LOOP_DAT[1].ENABLE = FALSE.
8. Rufen Sie die Anweisung PULSEGEN_M auf, wenn DB_LOOP.LOOP_DAT[2].ENABLE = TRUE ist.
9. Verschalten Sie DB_LOOP.LOOP_DAT[2].COM_RST und DB_LOOP.LOOP_DAT[2].CYCLE mit PULSEGEN_M.COM_RST und PULSEGEN_M.CYCLE.
10. Verschalten Sie PULSEGEN_M.INV mit dem Ausgangsparameter LMN ihres Regler-FBs.
11. Verschalten Sie PULSEGEN_M.PER_TM mit dem Eingangsparameter CYCLE ihres Regler-FBs.
12. Setzen Sie DB_LOOP.LOOP_DAT[2].ENABLE = FALSE.

13. Verschalten Sie PULSEGEN_M.QPOS_P und ggf. PULSEGEN_M.QNEG_P mit den digitalen Ausgängen der CPU, die das Stellglied steuern.
14. Rufen Sie die Funktion im OB 100 und im Weckalarm-OB auf.

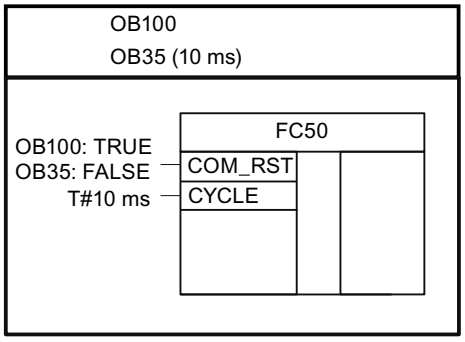

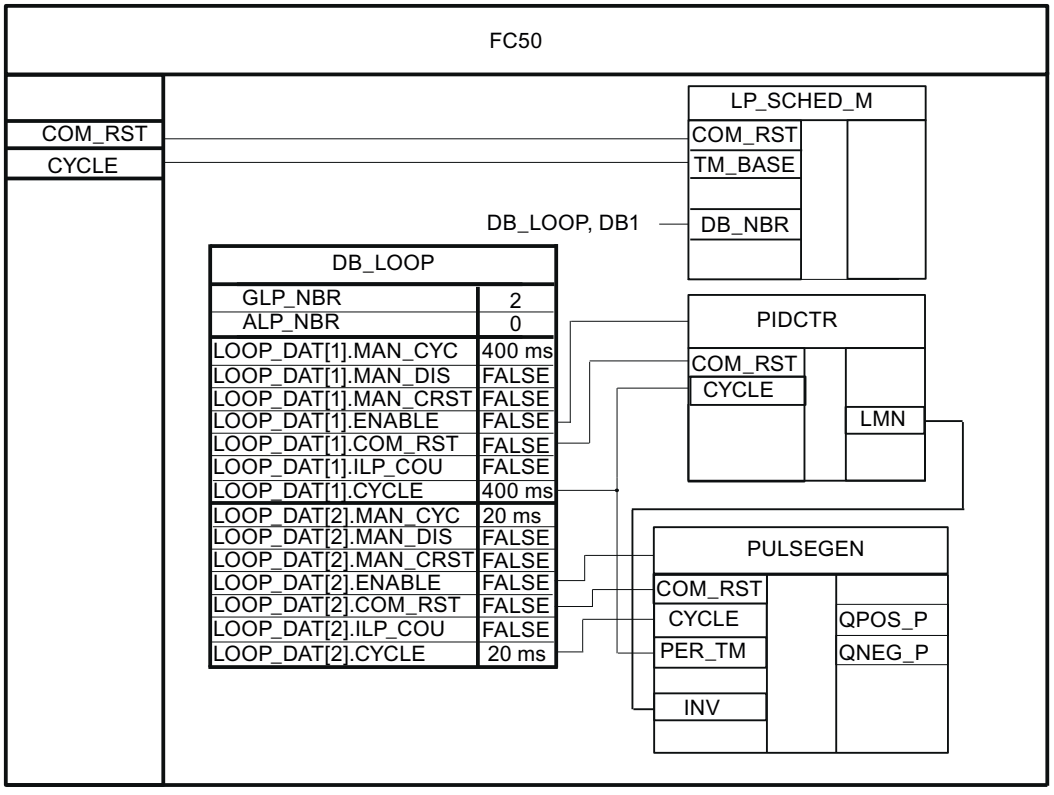

Siehe auch

Störgröße aufschalten (Seite 938)

Schrittregler erstellen

Für einen Schrittregler benötigen Sie die Anweisungen PID und LMNGEN_S.

Vorgehen

Um einen Schrittregler zu programmieren, gehen Sie folgendermaßen vor:

1. Rufen Sie im Regler-FB die Anweisungen PID und LMNGEN_S als Multiinstanz auf.
2. Parametrieren Sie in der Struktur PID_Instance der Multiinstanz das Verhalten des Regelalgorithmus, z. B. ob der D-Anteil verwendet wird.
3. Parametrieren Sie in der Struktur LMNGEN_S_Instance der Multiinstanz z. B., dass keine Stellungsrückmeldung verwendet werden soll, LMNR_ON = FALSE.
4. Verschalten Sie über eine temporäre Variable PID.ER mit der Regeldifferenz, z. B. DEADBAND.OUTV oder ERR_MON.ER.
5. Wenn Sie eine Störgröße aufschalten, verschalten Sie PID.DISV über eine temporäre Variable mit dem aufbereiteten Signal der Störgröße, z. B. CRP_IN.OUTV.
6. Verschalten Sie über die Anweisung BLKMOV den Ausgangsparameter PID.PID_LMNG mit dem Eingangsparameter LMNGEN_S.PID_LMNG und den Ausgangsparameter LMNGEN_S.LMNG_PID mit dem Eingangsparameter PID.LMNG_PID.
7. Deklarieren Sie am Regler-FB die Ausgangsparameter QLMNUP und QLMNDN vom Datentyp "Bool".
8. Verschalten Sie LNMGEN_S.QLMNUP und LMNGEN_S. QLMNDN mit diesen Ausgangsparametern.
9. Verschalten Sie die Ausgangsparameter des Regler-FBs mit den digitalen Ausgängen der CPU, die das Stellglied steuern.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR_S aufgerufen und verschaltet werden.

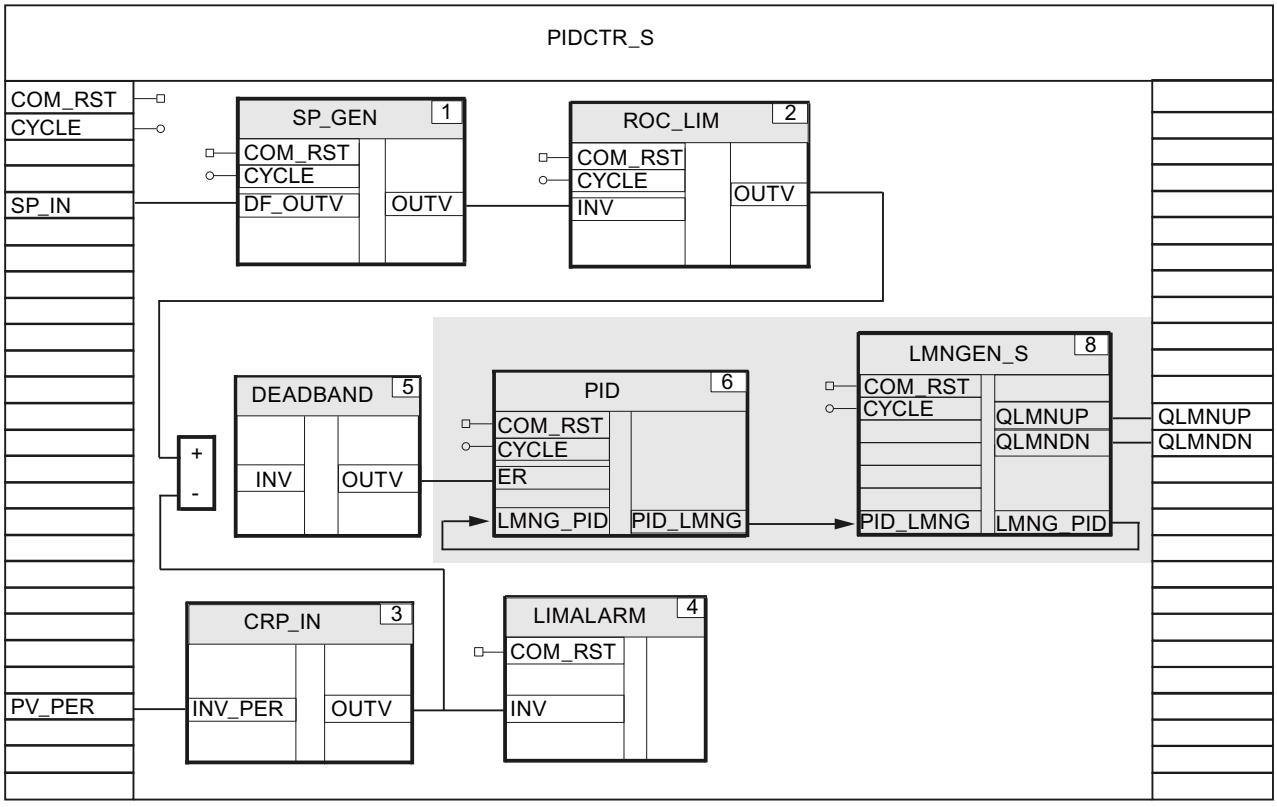

Siehe auch

Störgröße aufschalten (Seite 938)

Stellungsrückmeldung aufbereiten (Seite 947)

Stellungsrückmeldung aufbereiten

Die Konfiguration der Stellungsrückmeldung ist abhängig vom eingesetzten Stellglied.

- Stellglied mit analoger Stellungsrückmeldung
- Stellglied mit digitalen Anschlagssignalen

Voraussetzung

Sie haben bereits einen Schrittregler programmiert.

Schrittregler mit analoger Stellungsrückmeldung erstellen

Für die Aufbereitung der analogen Stellungsrückmeldung sind folgende Anweisungen sinnvoll.

- CRP_IN: Peripheriewert skalieren
- NONLIN: Stellungsrückmeldung linearisieren

- SCALE_M: Stellungsrückmeldung skalieren
- NORM: Stellungsrückmeldung normieren

Beispiel für eine analoge Stellungsrückmeldung

Ein Stellglied liefert eine Stellungsrückmeldung im Peripherieformat, die bei der Berechnung des Stellwerts berücksichtigt werden soll.

1. Deklarieren Sie am Regler-FB einen Eingangsparameter LMNR_PER vom Datentyp "Word".
2. Verschalten Sie LMNR_PER mit dem Peripherie-Eingang.
3. Rufen Sie im Regler-FB die Anweisung CRP_IN als Multiinstanz auf, z. B. mit dem Namen CRP_IN_Instance_LMNR.
4. Verschalten Sie LMNR_PER mit CRP_IN.INV_PER.
5. Parametrieren Sie in der Struktur CRP_IN_Instance_LMNR der Multiinstanz Faktor und Offset für die Skalierung.
6. Verschalten Sie CRP_IN.OUTV über eine temporäre Variable mit LMNGEN_S.LMNR_IN.
7. Aktivieren Sie in der Struktur LMNGEN_S_Instance der Multiinstanz die Stellungsrückmeldung mit LMNR_ON = TRUE.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR_SAF aufgerufen und verschaltet werden.

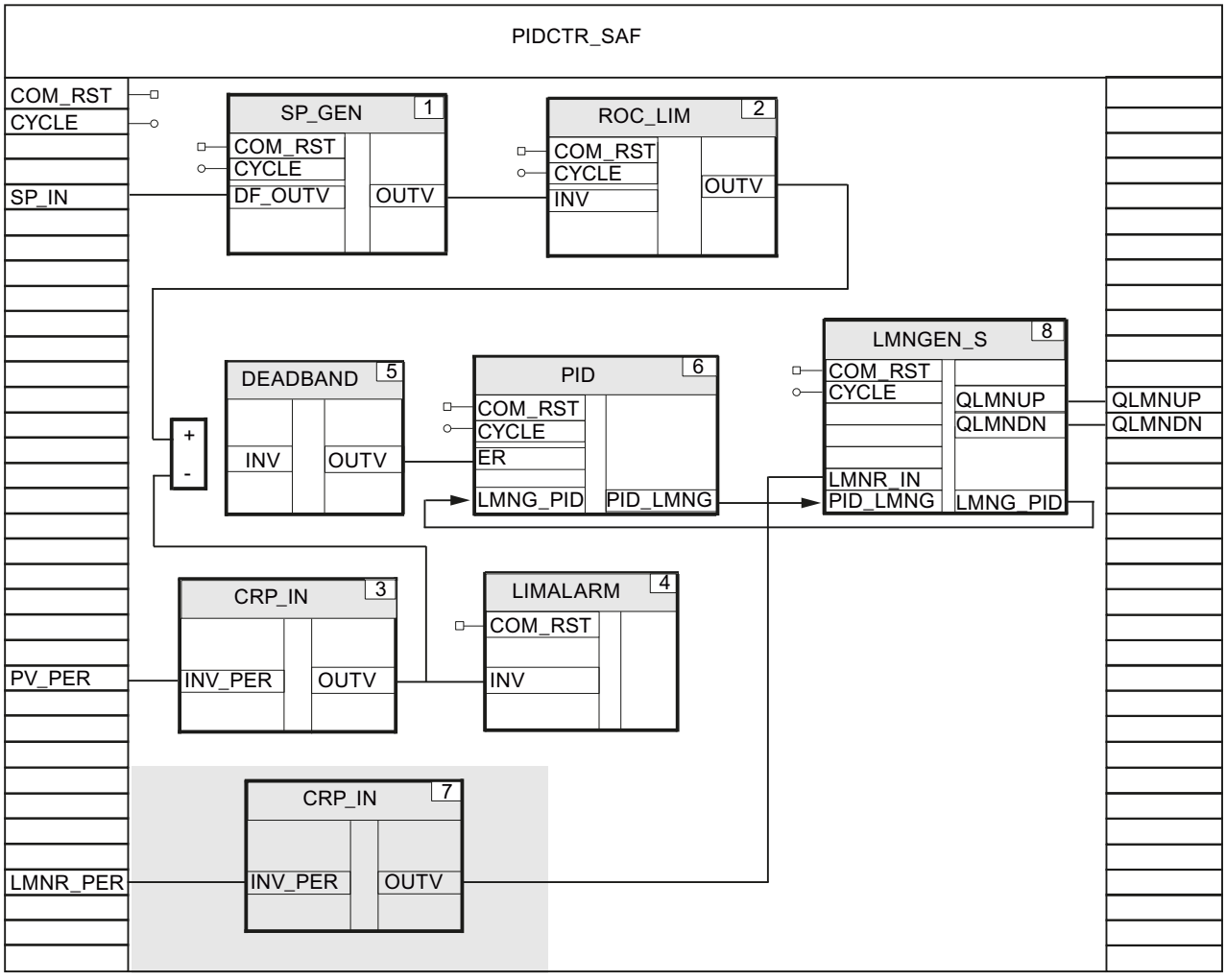

Schrittregler mit digitaler Stellungsrückmeldung erstellen

1. Deklarieren Sie am Regler-FB die Eingangsparameter LMNR_HS und LMNR_LS vom Datentyp "Bool".
2. Verschalten Sie die Eingangsparameter LMNR_HS und LMNR_LS mit LMNGEN_S.LMNR_HS und LMNGEN_S.LMNR_LS.
3. Aktivieren Sie in der Struktur LMNGEN_S_Instance die Stellungsrückmeldung mit LMNR_ON = TRUE.

Das folgende Bild zeigt, wie die Anweisungen des Modular PID Control im Regler-FB PIDCTR_SDF aufgerufen und verschaltet werden.

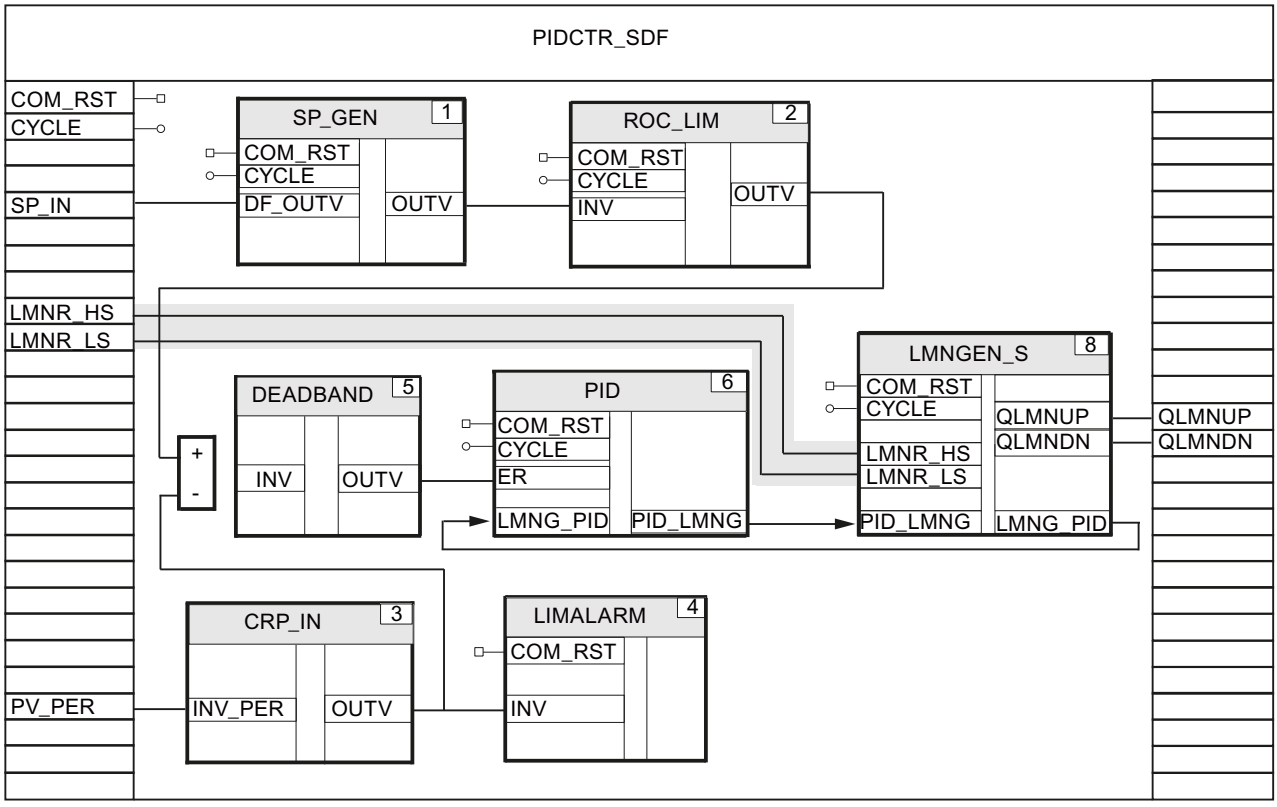

Siehe auch

Schrittregler erstellen (Seite 945)

3.5.2.4 Komplexe Reglerstrukturen

Einschleifiger Verhältnisregler

Das folgende Bild zeigt einen typischen Regelkreis für einen einschleifigen Verhältnisregler.

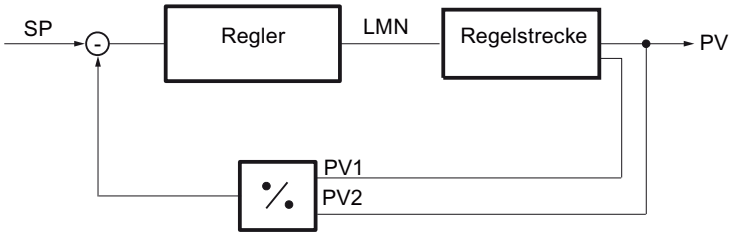

Das Verhältnis von zwei Istwerten wird auf einen eingestellten Sollwert mit einem PID-Algorithmus geregelt.

Vorgehen

Um einen einschleifigen Verhältnisregler zu erstellen, gehen Sie folgendermaßen vor:

1. Programmieren Sie in Ihrem Regler-FB die Aufbereitung der Istwerte wie unten dargestellt.
2. Um eine Division durch Null zu vermeiden, müssen Sie den Wertebereich für den Istwert PV_PER2 mit der Anweisung LIMITER auf Werte > Null begrenzen.

Das folgende Bild zeigt einen Regler-FB, hier RATIOCTR, für einen einschleifigen Verhältnisregler mit kontinuierlichem Stellwert.

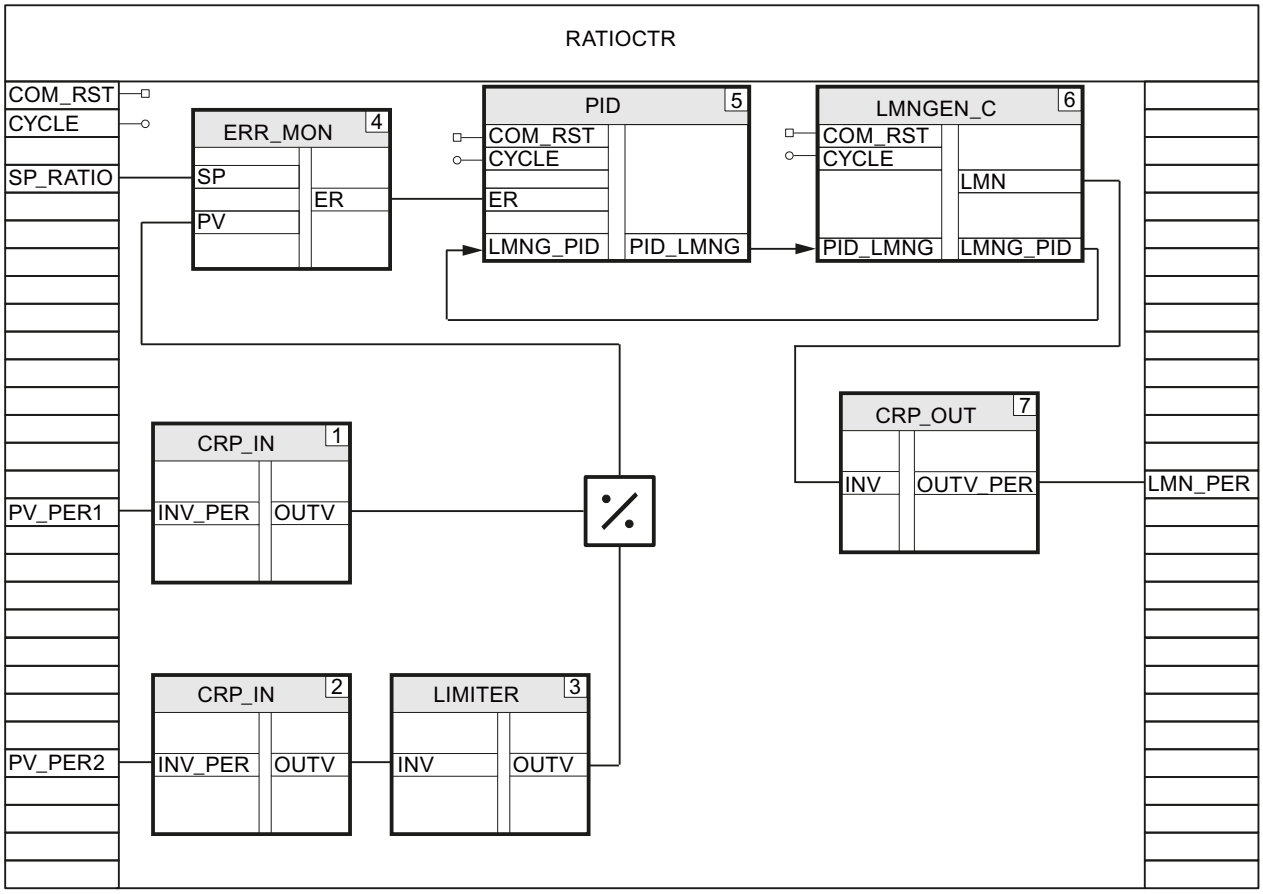

Mehrschleifiger Verhältnisregler

Das folgende Bild zeigt einen typischen Regelkreis für einen mehrschleifigen Verhältnisregler.

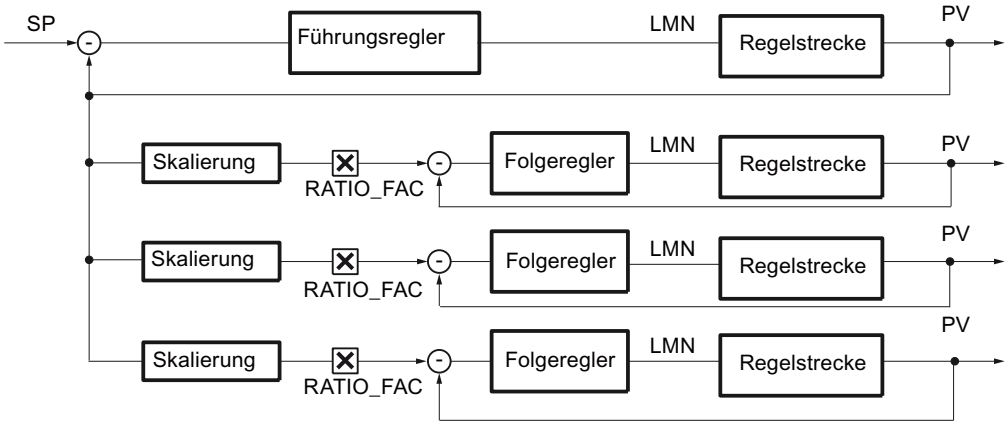

Regelkreis programmieren

Um dieses Beispiel mit Modular PID Control zu programmieren, gehen Sie folgendermaßen vor:

1. Programmieren Sie einen Regler-FB für den Führungsregler, z. B. PIDCTR_PV.
2. Programmieren Sie einen oder mehrere Regler-FBs für die Folgeregler, z. B. RB_CTR_S.
3. Programmieren Sie eine Funktion, in der Sie die vier Regler aufrufen und miteinander verschalten.

Führungsregler programmieren

Um den Führungsregler zu programmieren, gehen Sie folgendermaßen vor:

1. Programmieren Sie den Führungsregler nach ihren Anforderungen wie im Kapitel Regler-FB programmieren (Seite 932) beschrieben.
2. Deklarieren Sie am Regler-FB, hier PIDCTR_PV, einen Ausgangsparameter PV vom Datentyp "Real", an dem Sie den aufbereiteten Istwert des Führungsreglers ausgeben.

Das folgende Bild zeigt für den Führungsregler die Verschaltung am Beispiel des Regler-FBs PIDCTR_PV.

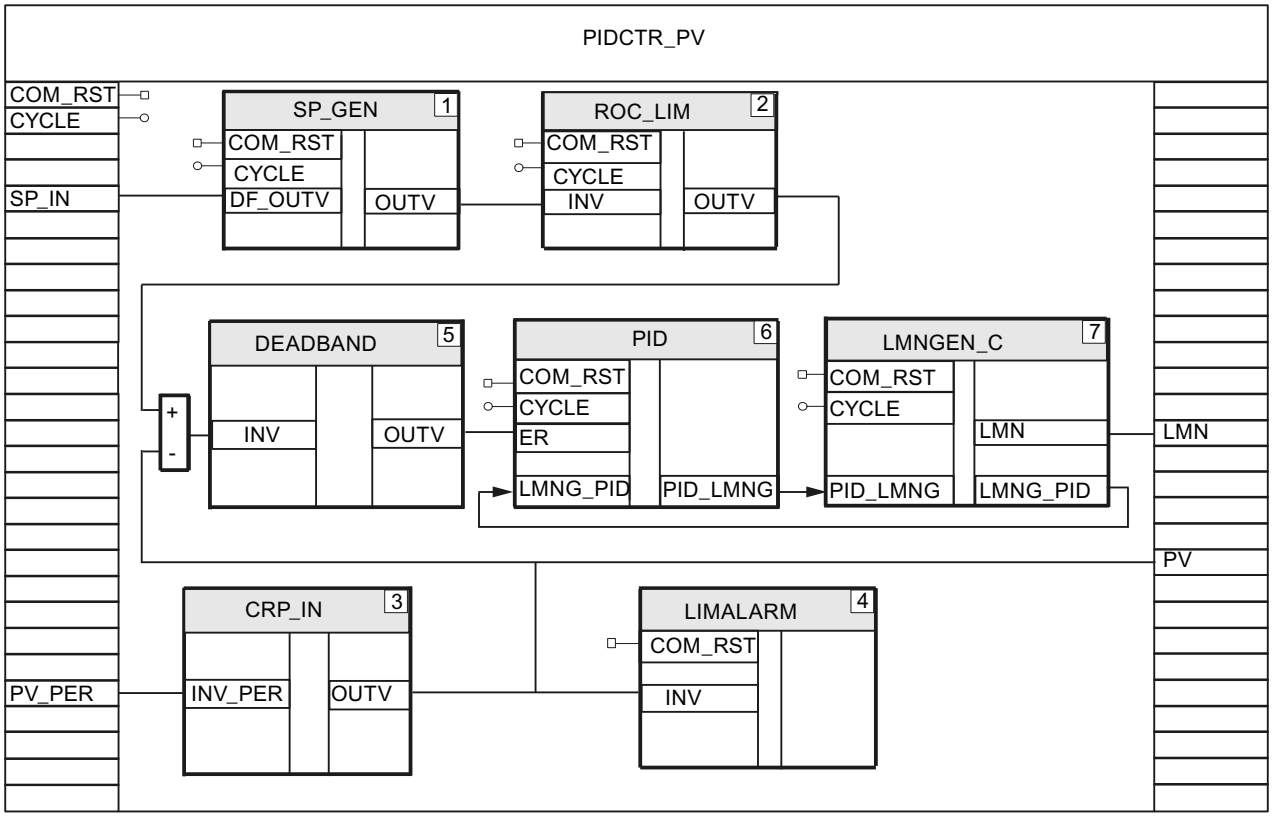

Folgeregler programmieren

Um die Folgeregler zu programmieren, gehen Sie folgendermaßen vor:

1. Programmieren Sie die Folgeregler nach ihren Anforderungen wie im Kapitel Regler-FB programmieren (Seite 932) beschrieben.
2. Deklarieren Sie am Regler-FB, hier RB_CTR_S, einen Eingangsparameter RATIO_FAC vom Datentyp "Int". Mit diesem Wert wird der aufbereitete Sollwert, z. B. SCALE.OUTV des Folgereglers multipliziert.

Das folgende Bild zeigt die Verschaltung für den Folgeregler am Beispiel des Regler-FBs RB_CTR_S.

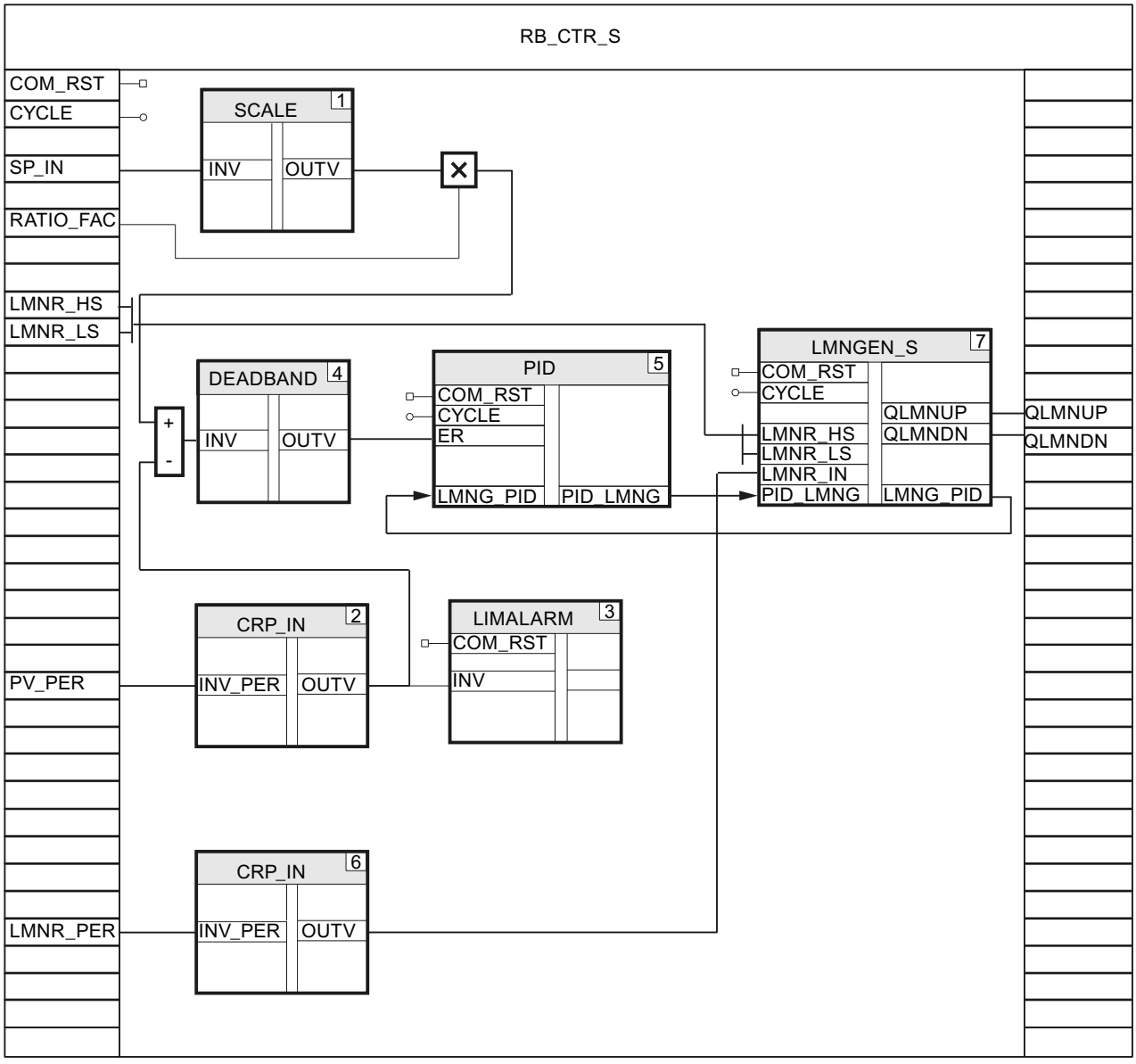

Regler aufrufen und miteinander verschalten

Um die Führungs- und Folgeregler aufzurufen und miteinander zu verschalten, gehen Sie folgendermaßen vor:

1. Erstellen Sie eine Funktion, z. B. FC50, mit den Eingangsparametern COM_RST und CYCLE.
2. Rufen Sie in der Funktion die Anweisung LP_SCHED_M auf.
3. Legen Sie einen AUTOHOTSPOT an, in dem Sie die Anzahl der Regler und die Zykluszeiten für den Aufruf des Führungsreglers und der Folgeregler parametrieren. Die Zykluszeiten der Folgeregler müssen kleiner oder gleich der Zykluszeit des Führungsreglers sein.

4. Rufen Sie den Regler-FB für den Führungsregler auf, hier PIDCTR_PV, wenn DB_LOOP.LOOP_DAT[1].ENABLE = TRUE ist.
5. Verschalten Sie DB_LOOP.LOOP_DAT[1].COM_RST und DB_LOOP.LOOP_DAT[1].CYCLE mit den Eingangsparametern COM_RST und CYCLE des Führungsreglers.
6. Setzen Sie DB_LOOP.LOOP_DAT[1].ENABLE = FALSE.
7. Rufen Sie den Regler-FB für den ersten Folgeregler auf, hier RB_CTR_S, wenn DB_LOOP.LOOP_DAT[2].ENABLE = TRUE ist.
8. Verschalten Sie DB_LOOP.LOOP_DAT[2].COM_RST und DB_LOOP.LOOP_DAT[2].CYCLE mit COM_RST und CYCLE des Folgereglers.
9. Verschalten Sie den Ausgangsparameter PV des Führungsreglers mit dem Eingangsparameter SP_IN des Folgereglers.
10. Setzen Sie DB_LOOP.LOOP_DAT[2].ENABLE = FALSE.
11. Wiederholen Sie die letzten vier Schritte für die beiden weiteren Folgeregler. Erhöhen Sie für jeden Regler den Zähler des Arrays.
12. Verschalten Sie die Stellwerte mit den Ausgängen der CPU.
13. Rufen Sie die Funktion im OB 100 und im Weckalarm-OB auf.

Das folgende Bild zeigt die Aufrufe und Verschaltung der Regler am Beispiel der Funktion FC50.

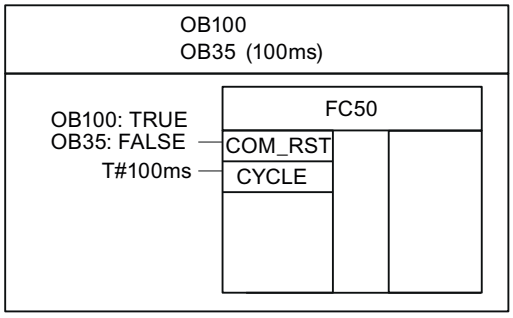

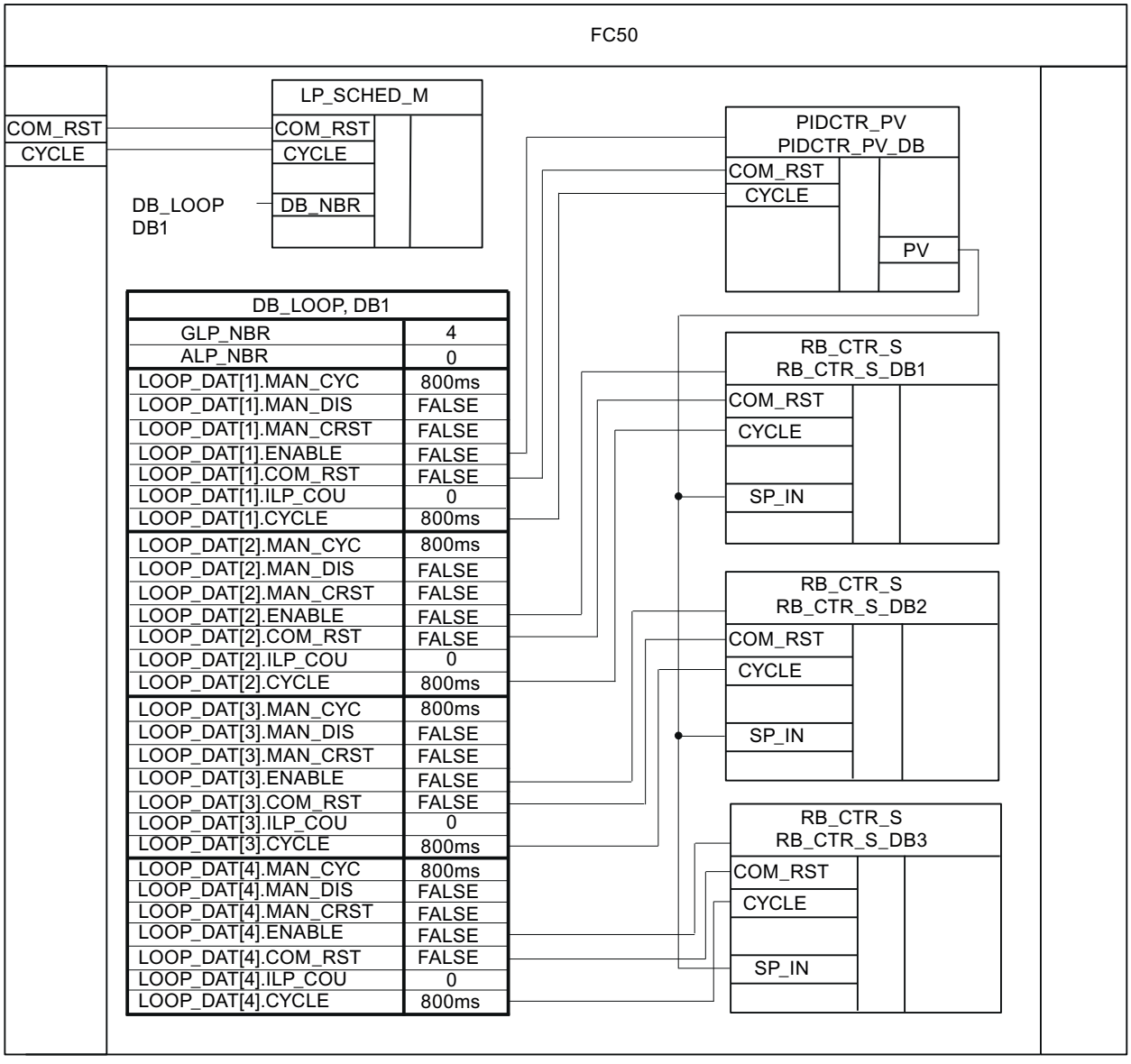

DB_LOOP, DB1	
GLP_NBR	4
ALP_NBR	0
LOOP_DAT[1].MAN_CYC	800ms
LOOP_DAT[1].MAN_DIS	FALSE
LOOP_DAT[1].MAN_CRST	FALSE
LOOP_DAT[1].ENABLE	FALSE
LOOP_DAT[1].COM_RST	FALSE
LOOP_DAT[1].ILP_COU	0
LOOP_DAT[1].CYCLE	800ms
LOOP_DAT[2].MAN_CYC	800ms
LOOP_DAT[2].MAN_DIS	FALSE
LOOP_DAT[2].MAN_CRST	FALSE
LOOP_DAT[2].ENABLE	FALSE
LOOP_DAT[2].COM_RST	FALSE
LOOP_DAT[2].ILP_COU	0
LOOP_DAT[2].CYCLE	800ms
LOOP_DAT[3].MAN_CYC	800ms
LOOP_DAT[3].MAN_DIS	FALSE
LOOP_DAT[3].MAN_CRST	FALSE
LOOP_DAT[3].ENABLE	FALSE
LOOP_DAT[3].COM_RST	FALSE
LOOP_DAT[3].ILP_COU	0
LOOP_DAT[3].CYCLE	800ms
LOOP_DAT[4].MAN_CYC	800ms
LOOP_DAT[4].MAN_DIS	FALSE
LOOP_DAT[4].MAN_CRST	FALSE
LOOP_DAT[4].ENABLE	FALSE
LOOP_DAT[4].COM_RST	FALSE
LOOP_DAT[4].ILP_COU	0
LOOP_DAT[4].CYCLE	800ms

Kaskadenregler

Das folgende Bild zeigt einen typischen Regelkreis für einen Kaskadenregler.

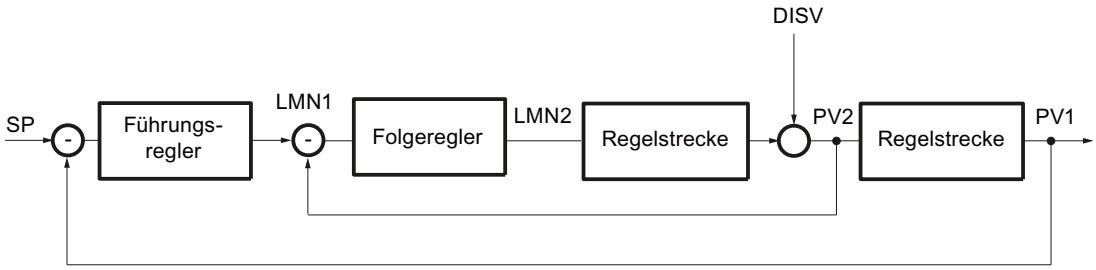

Regelkreis programmieren

Um dieses Beispiel mit Modular PID Control zu programmieren, gehen Sie folgendermaßen vor:

1. Programmieren Sie einen Regler-FB für den Führungsregler, z. B. PIDCTR.
2. Programmieren Sie einen Regler-FB für den Folgeregler, z. B. PIDCTR_SAF.
3. Programmieren Sie eine Funktion, in der Sie die beiden Regler aufrufen und miteinander verschalten.
4. Rufen Sie in der Funktion die Anweisung LP_SCHED_M auf.
5. Legen Sie einen AUTOHOTSPOT an, in dem Sie die Anzahl der Regler und die Zykluszeiten für den Aufruf des Führungsreglers und der Folgeregler parametrieren. Die Zykluszeit des Folgereglers muss kleiner oder gleich der Zykluszeit des Führungsreglers sein.
6. Rufen Sie den Regler-FB für den Führungsregler auf, wenn DB_LOOP.LOOP_DAT[1].ENABLE = TRUE ist.
7. Verschalten Sie DB_LOOP.LOOP_DAT[1].COM_RST und DB_LOOP.LOOP_DAT[1].CYCLE mit den Eingangsparametern COM_RST und CYCLE des Führungsreglers.
8. Setzen Sie DB_LOOP.LOOP_DAT[1].ENABLE = FALSE.
9. Rufen Sie den Regler-FB für den Folgeregler auf, wenn DB_LOOP.LOOP_DAT[2].ENABLE = TRUE ist.
10. Verschalten Sie DB_LOOP.LOOP_DAT[2].COM_RST und DB_LOOP.LOOP_DAT[2].CYCLE mit COM_RST und CYCLE des Folgereglers.
11. Verschalten Sie den Ausgangsparameter LMN des Führungsreglers mit dem Eingangsparameter SP_IN des Folgereglers.
12. Setzen Sie DB_LOOP.LOOP_DAT[2].ENABLE = FALSE.
13. Verschalten Sie den Stellwert des Folgereglers mit dem analogen Ausgang der CPU, der das Stellglied steuert.
14. Rufen Sie die Funktion im OB 100 und im Weckalarm-OB auf.

Das folgende Bild zeigt die Aufrufe und Verschaltung der Regler am Beispiel der Funktion FC50.

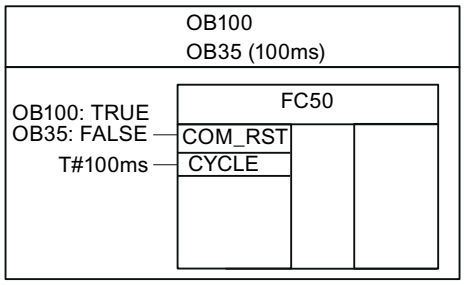

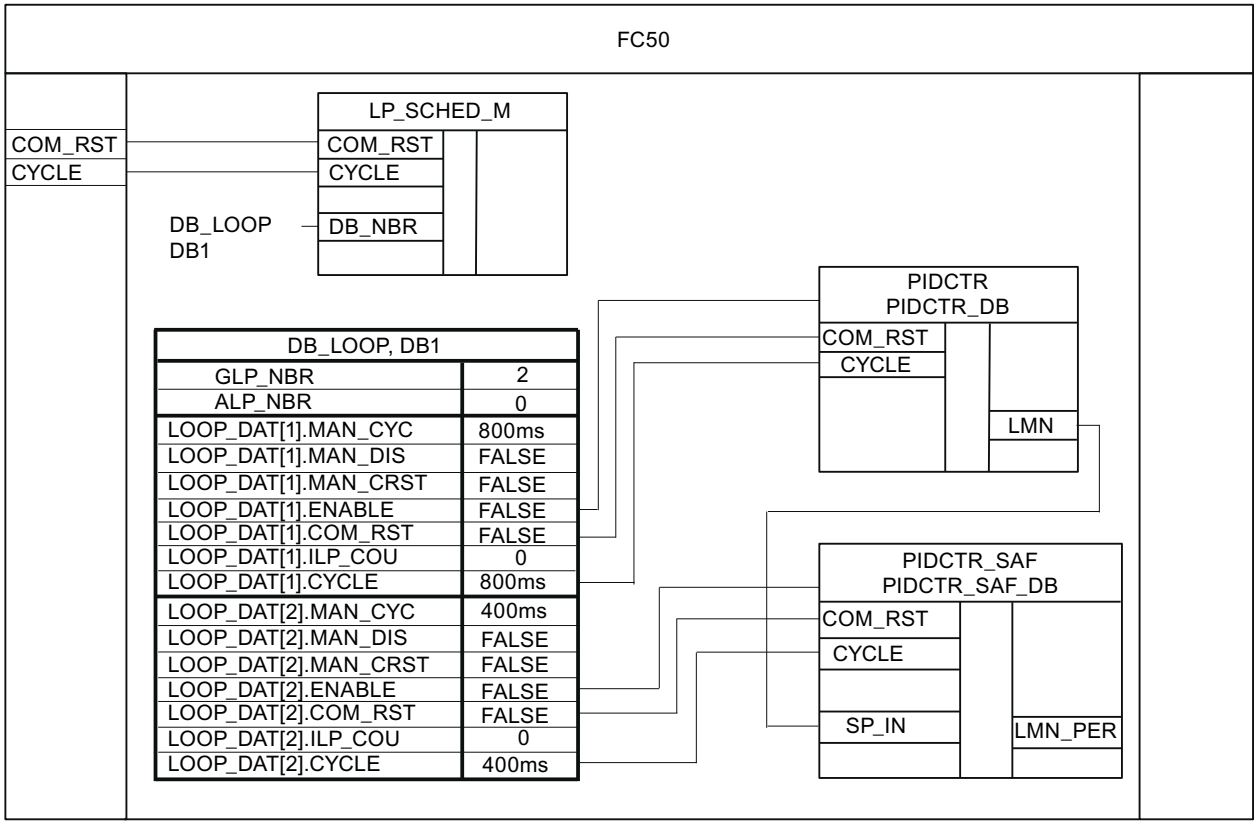

DB_LOOP, DB1	
GLP_NBR	2
ALP_NBR	0
LOOP_DAT[1].MAN_CYC	800ms
LOOP_DAT[1].MAN_DIS	FALSE
LOOP_DAT[1].MAN_CRST	FALSE
LOOP_DAT[1].ENABLE	FALSE
LOOP_DAT[1].COM_RST	FALSE
LOOP_DAT[1].ILP_COU	0
LOOP_DAT[1].CYCLE	800ms
LOOP_DAT[2].MAN_CYC	400ms
LOOP_DAT[2].MAN_DIS	FALSE
LOOP_DAT[2].MAN_CRST	FALSE
LOOP_DAT[2].ENABLE	FALSE
LOOP_DAT[2].COM_RST	FALSE
LOOP_DAT[2].ILP_COU	0
LOOP_DAT[2].CYCLE	400ms

Mehrgrößenregler

Das folgende Bild zeigt einen typischen Regelkreis für einen Mehrgrößenregler. Zwei PID–Regler mit kontinuierlichem Ausgang werden auf die Regelstrecke geschaltet.

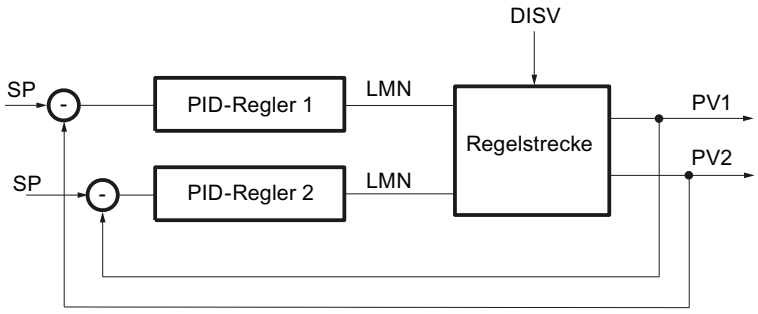

Regelkreis programmieren

Um dieses Beispiel mit Modular PID Control zu programmieren, gehen Sie folgendermaßen vor:

1. Deklarieren Sie am Regler-FB, hier MUL_CTR, folgende Eingangsparameter:
   - COM_RST vom Datentyp "Bool"
   - CYCLE vom Datentyp "Time"
   - SP_IN1 und SP_IN2vom Datentyp "Real"
   - PV_PER1 und PV_PER2 vom Datentyp "Word"
2. Deklarieren Sie am Regler-FB die Ausgangsparameter LMN_PER1 und LMN_PER2 vom Datentyp "Word".
3. Programmieren Sie für beide PID-Regler die Aufbereitung des Soll- und Istwerts und die Berechnung der Stellgröße.
4. Um die Regeldifferenz des ersten PID-Reglers zu berechnen, subtrahieren Sie den aufbereiteten Istwert PV_PER1 vom Produkt aus Sollwert SP_IN1 und dem Stellwert des zweiten PID-Reglers.
5. Um die Regeldifferenz des zweiten PID-Reglers zu berechnen, subtrahieren Sie den aufbereiteten Istwert PV_PER2 vom Produkt aus Sollwert SP_IN2 und dem Stellwert des ersten PID-Reglers.
6. Verschalten Sie die Stellwerte LMN_PER1 und LMN_PER2 mit zwei analogen Ausgängen, die die Stellglieder steuern.
7. Rufen Sie den Regler-FB im OB 100 und im Weckalarm-OB auf.

Das folgende Bild zeigt den Aufruf am Beispiel des Regler-FBs MUL_CTR.

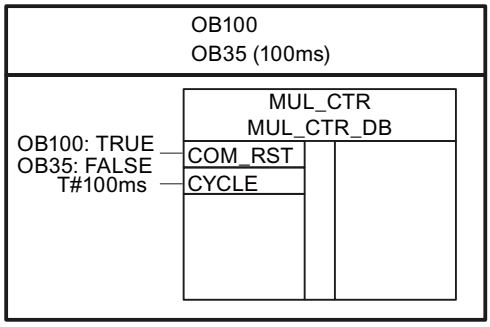

Das folgende Bild zeigt die Verschaltung der Regler am Beispiel des Regler-FBs MUL_CTR.

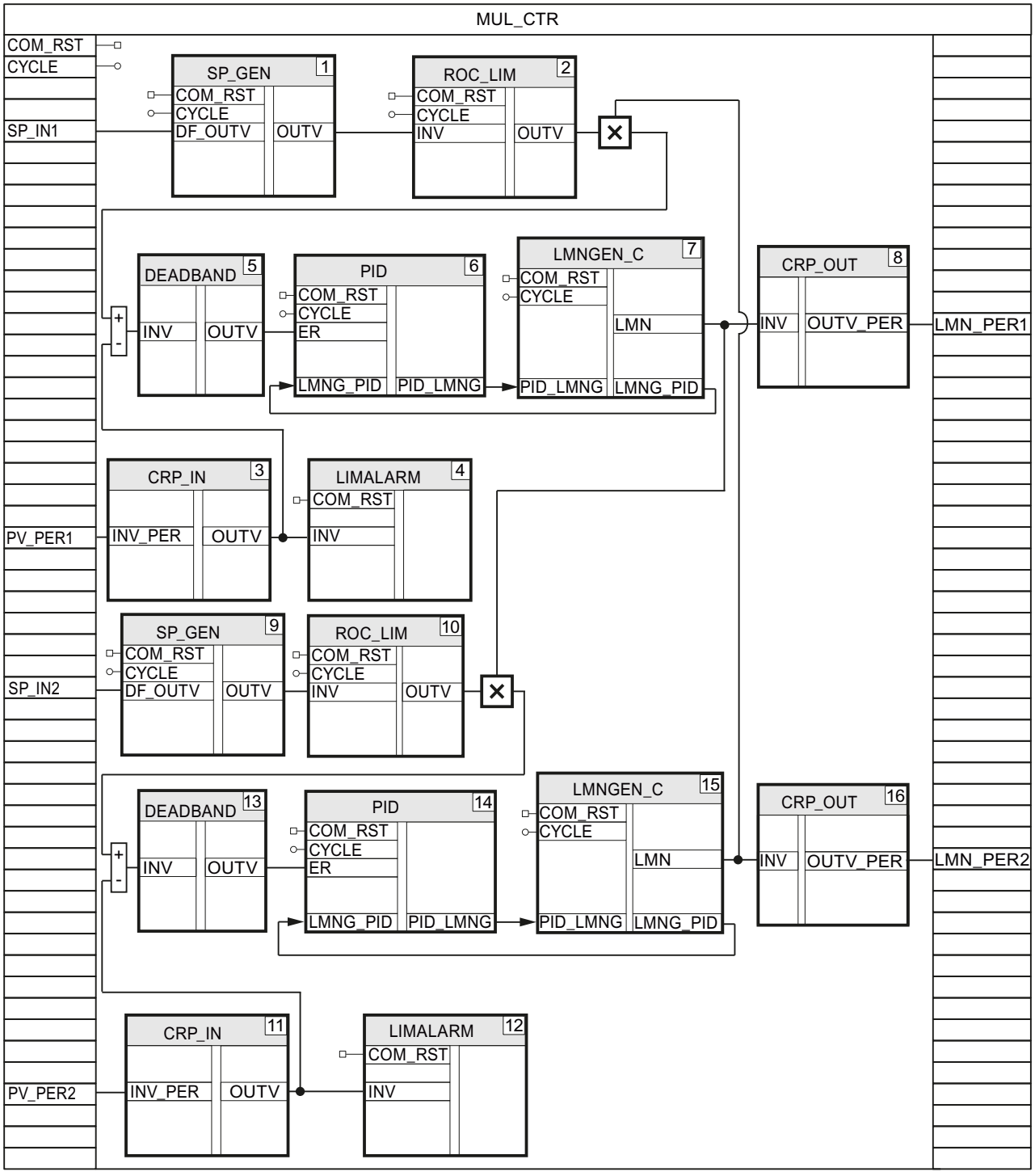
MUL_CTR
COM_RST
CYCLE
SP_IN1
SP_GEN 1
COM_RST
CYCLE
DF_OUTV
OUTV
ROC_LIM 2
COM_RST
CYCLE
INV
OUTV
DEADBAND 5
INV
OUTV
PID 6
COM_RST
CYCLE
ER
LMNG_PID
PID_LMNG
LMNGEN_C 7
COM_RST
CYCLE
LMN
PID_LMNG
LMNG_PID
CRP_OUT 8
INV
OUTV_PER
LMN_PER1
CRP_IN 3
PV_PER1
INV_PER
OUTV
LIMALARM 4
COM_RST
INV
SP_GEN 9
COM_RST
CYCLE
SP_IN2
DF_OUTV
OUTV
ROC_LIM 10
COM_RST
CYCLE
INV
OUTV
DEADBAND 13
INV
OUTV
PID 14
COM_RST
CYCLE
ER
LMNG_PID
PID_LMNG
LMNGEN_C 15
COM_RST
CYCLE
LMN
PID_LMNG
LMNG_PID
CRP_OUT 16
INV
OUTV_PER
LMN_PER2
CRP_IN 11
PV_PER2
INV_PER
OUTV
LIMALARM 12
COM_RST
INV

3.5.2.5 Regler-FB in Betrieb nehmen

Kontinuierlichen Regler in Betrieb nehmen

Ihren Regler-FB können Sie mit dem PID Self-Tuner in Betrieb nehmen und die optimalen PID-Parameter ermitteln.

Vorgehen

Um einen kontinuierlichen Regler oder einen Impulsregler in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Rufen Sie den Regler-FB in einem Weckalarm-OB auf und verschalten Sie die Ein- und Ausgangsparameter.
2. Rufen Sie die Anweisung TUN_EC im selben Weckalarm-OB auf.
3. Verschalten Sie die Anweisung TUN_EC folgendermaßen mit ihrem Regler-FB:
   - TUN_EC.PV mit dem aufbereiteten Istwert, z. B. der statischen Variablen CRP_IN_Instance_PV.OUTV oder dem Ausgangsparameter PV
   - TUN_EC.LMN mit dem Stellwert im Real-Format, z. B. die statischer Variable LMNGEN_C_Instance.LMN
   - TUN_EC.MAN mit dem Durchgangsparameter für den Handwert, z. B. MAN.
   - TUN_EC.SP_OUT mit dem Eingangsparameter für den Sollwert, z. B. SP_IN
   - TUN_EC.GAIN mit der Proportionalverstärkung, z. B. PID_Instance.GAIN
   - TUN_EC.TI mit der Integrationszeit, z. B. PID_Instance.TI
   - TUN_EC.TD mit der Differenzierzeit, z. B. PID_Instance.TD
   - TUN_EC.TM_LAG mit der Verzögerungszeit, z. B. PID_Instance.TM_LAG
4. Klicken Sie auf das Symbol an der Anweisung TUN_EC.
   Der Inbetriebnahmeeditor für TUN_EC wird geöffnet. Die Bedienung des Inbetriebnahmeeditors ist beim PID Self-Tuner beschrieben.

Schrittregler in Betrieb nehmen

Ihren Regler-FB können Sie mit dem PID Self-Tuner in Betrieb nehmen und die optimalen PID-Parameter ermitteln.

Vorgehen

Um einen Schrittregler in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Rufen Sie den Regler-FB in einem Weckalarm-OB auf und verschalten Sie die Ein- und Ausgangsparameter.
2. Rufen Sie die Anweisung TUN_ES im selben Weckalarm-OB auf.

3. Verschalten Sie die Anweisung TUN_ES folgendermaßen mit ihrem Regler-FB:
   - TUN_ES.PV mit dem aufbereiteten Istwert, z. B. der statischen Variablen CRP_IN_Instance _PV.OUTV oder dem Ausgangsparameter PV
   - TUN_ES.LMNR_IN mit der aufbereiteten Stellungsrückmeldung, z. B. LMNGEN_S_instance.LMNR_IN
   - TUN_ES.C_LMNUP und TUN_ES.C_LMNDN mit den Stellwertsignalen, z. B. den Ausgangsparametern QLMN_UP und QLMNDN
   - TUN_ES.LMNR_HS mit dem Signal für den oberen Anschlag, z. B. LMNGEN_S_Instance.LMNR_HS
   - TUN_ES.LMNR_ON mit dem Schalter für die Stellungsrückmeldung, z. B. LMNGEN_S_Instance.LMNR_ON
   - TUN_ES.SP_OUT mit dem Eingangsparameter für den Sollwert, z. B. SP_IN
   - TUN_ES.GAIN mit der Proportionalverstärkung, z. B. PID_Instance.GAIN
   - TUN_ES.TI mit der Integrationszeit, z. B. PID_Instance.TI
   - TUN_ES.TD mit der Differenzierzeit, z. B. PID_Instance.TD
   - TUN_ES.TM_LAG mit der Verzögerungszeit, z. B. PID_Instance.TM_LAG
   - TUN_ES.MTR_TM mit der Motorstellzeit, z. B. LMNGEN_S_Instance.MTR_TM
   - Wenn Sie eine Totzone einsetzen, TUN_ES.DEADB_W mit der Totzonenbreite, z. B. DEADBAND_Instance.OUTV
   - TUN_ES.QLMNUP und TUN_ES.QLMNDN mit den Stellwertsignalen LMNGEN_S_Instance.LMNUP und LMNGEN_S_Instance.LMNDN
4. Klicken Sie auf das Symbol an der Anweisung TUN_ES.
   Der Inbetriebnahmeeditor für TUN_ES wird geöffnet. Die Bedienung des Inbetriebnahmeeditors ist beim PID Self-Tuner beschrieben.

3.5.3 PID Self-Tuner einsetzen

3.5.3.1 Kontinuierliche Regler und Impulsregler optimieren

TUN_EC mit Regler verschalten

Die Anweisung TUN_EC dient zur Erstellung eines kontinuierlichen Reglers oder eines Impulsreglers. Sie verschalten TUN_EC je nach Regler mit einer der folgenden Anweisungen:

Regler	Anweisung
FM 355 C, FM 355 S Impulsregler	PID_FM
FM 455 C, FM 455 S Impulsregler	PID_FM
PID Basisfunktionen	CONT_C mit oder ohne PULSEGEN

Regler	Anweisung
Standard PID Control (Optionspaket PID Professional)	PID_CP
Modular PID Control (Optionspaket PID Professional)	PID LMNGEN_C

Verschalten Sie die Ein- und Ausgangsparameter der Anweisung TUN_EC folgendermaßen mit der Anweisung des Reglers.

TUN_EC	PID_FM	CONT_C	PID_CP
PV	PV	PV	PV
LMN	LMN	LMN	LMN
NORM_FAC	--	--	(NM_PVHR-NM_PVLR)*0.01
WRITE_DIS	Oderverknüpfung aus LOAD_OP und LOAD_PAR	--	--
MAN_OUT	LMN_RE	MAN	MAN
SP_OUT	SP_RE	SP_INT	SP_INT
GAIN	GAIN	GAIN	GAIN
TI	TI	TI	TI
TD	TD	TD	TD
TM_LAG	TM_LAG	TM_LAG	TM_LAG
RATIOFAC	--	--	RATIOFAC
PHASE	--	--	PHASE
QMAN_ON	LMN_REON	MAN_ON	MAN_ON
QI_SEL	--	I_SEL	I_SEL
QD_SEL	--	D_SEL	D_SEL
QWRITE	LOAD_PAR und LOAD_OP	--	--

Möglichkeiten zur Optimierung

Erstoptimierung

Die Optimierung der PID-Parameter erfolgt während des Aufheizvorgangs. Dabei wird eine Prozessidentifikation mit anschließendem Reglerentwurf durchgeführt.

Sie können eine Optimierung mit ADAPT1ST anfordern.

Wenn Sie ADAPT1ST = TRUE setzen, dann wird ADAPT_ON = TRUE gesetzt und zwingend eine Erstoptimierung durchgeführt. Bei einer Erstoptimierung geben Sie den Stellwertsprung am Parameter LHLM_TUN an.

Der Parameter ADAPT1ST ist nur erforderlich, wenn durch gravierende Fehler bei einer Adaption so unsinnige Prozessparameter ermittelt worden sind, dass der automatisch berechnete Stellgrößensprung für eine erneute Adaption ungeeignet ist.

Bei ADAPT1ST = TRUE wird die Initialisierungsroutine der Anweisung TUN_EC durchlaufen. Dabei werden folgende Vorbelegungen gemacht:

- Es wird auf PI-Regler geschaltet. Hierzu werden die Parameter von PI_CON an den Regler übertragen. Haben die Parameter PI_CON.GAIN und PI_CON.TI beide den Wert 0.0, dann werden Sie mit PI_CON.GAIN = 1.5 und PI_CON.TI = 3600.0 s vorbelegt.
- PROCESS.GAIN = 999.0
- RATIOFAC = 1.0
- STATUS_H = 0; STATUS_D = 0; STATUS_C = 0

Hinweis

Wenn nicht explizit gewünscht, rufen Sie bei Neustart (Warmstart) der CPU TUN_EC **nicht** wie bei anderen Anweisungen üblich in seiner Initialisierungsroutine auf, da sonst alle Prozessparameter verloren gehen.

Mit ADAPT1ST = TRUE und einem anschließenden Sollwertsprung ≥ MIN_STEP in positiver Richtung wird die Optimierung gestartet.

Der Sollwertsprung bei der Erstoptimierung geht in der Regel vom Sollwert des kalten Prozesses (Umgebungstemperatur) bis in die Nähe des Arbeitspunktes. Es ist aber auch ein beliebiger Sollwert als Anfangszustand möglich, jedoch muss sichergestellt werden, dass sich der Prozess in einem stationären Zustand befindet. Während der Optimierung dürfen Sie keine weiteren Sollwertsprünge vorgeben.

Kurvenverläufe der Erstoptimierung

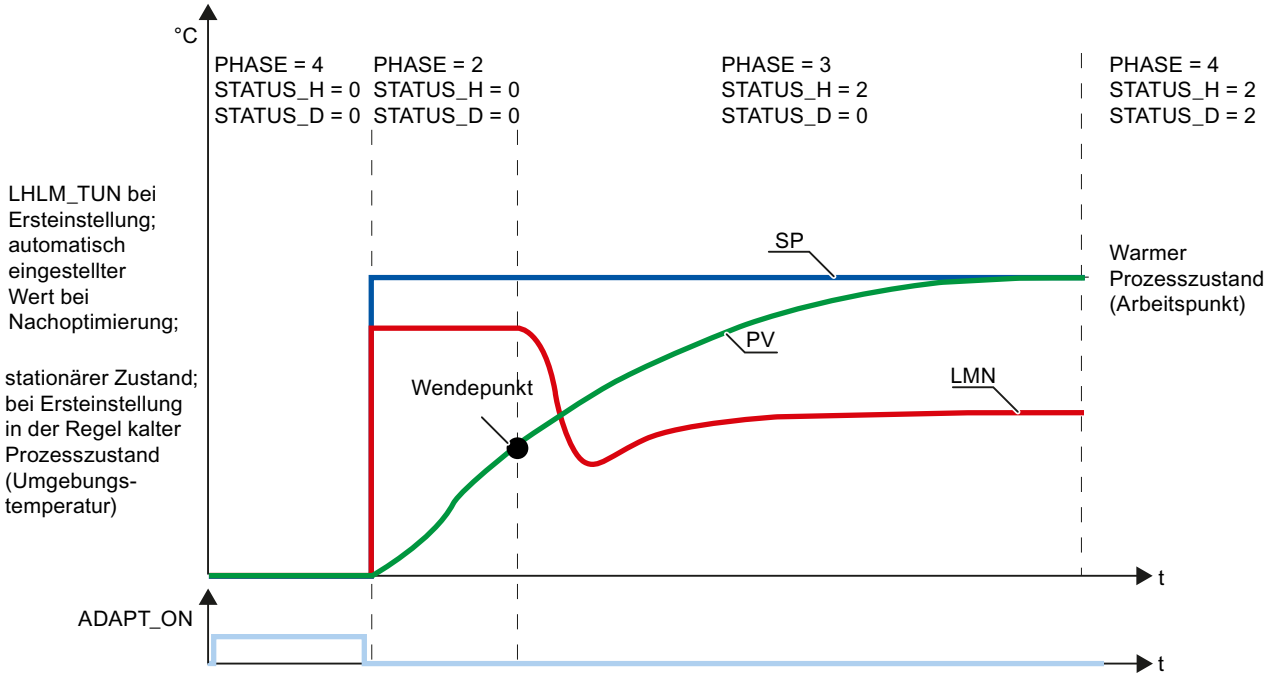

Nachoptimierung

Eine Nachoptimierung kann nur nach einer korrekt durchlaufenen Erstoptimierung erfolgen. Mit ADAPT_ON = TRUE entscheidet TUN_EC, ob er eine Erstoptimierung oder eine Nachoptimierung vornimmt. Wenn bereits Informationen über den Prozess vorliegen, berechnet TUN_EC automatisch einen Stellwertsprung und führt eine Nachoptimierung durch. Wenn keine Informationen über den Prozess vorliegen, wird eine Erstoptimierung durchgeführt.

Optimierung starten

Sie können eine Optimierung sowohl aus dem Automatikbetrieb heraus als auch aus dem Handbetrieb heraus starten:

- **Starten der Optimierung aus dem Automatikbetrieb**
  Setzen Sie im Automatikbetrieb ADAPT_ON oder ADAPT1ST = TRUE und geben Sie einen Sollwertsprung ≥ MIN_STEP vor.
- **Starten der Optimierung aus dem Handbetrieb**
  Setzen Sie im Handbetrieb ADAPT_ON oder ADAPT1ST = TRUE und geben Sie einen Sollwertsprung ≥ MIN_STEP vor. Erst wenn Sie anschließend MAN_ON = FALSE zurücksetzen, wird der Sollwertsprung wirksam und die Optimierung beginnt.

Kühloptimierung

Die Kühloptimierung kann nur nach einer korrekt durchlaufenen Erstoptimierung erfolgen. Sie wird in der PHASE 4 gestartet. Mit dem Setzen von COOLID_ON = TRUE wird der Regler in Handbetrieb gesetzt und LLLM_TUN ausgegeben: MAN_OUT = LLLM_TUN. TUN_EC wechselt in PHASE = 3, und COOLID_ON wird sofort wieder zurückgesetzt. Sobald ein Wendepunkt im Istwertverlauf gefunden wird, wechselt TUN_EC wieder in den Regelbetrieb (PHASE = 4). TUN_EC ermittelt einen Verhältnisfaktor RATIOFAC, dessen Wert an den Pulsgenerator des Reglers weitergeleitet wird.

Kurvenverläufe der Kühloptimierung

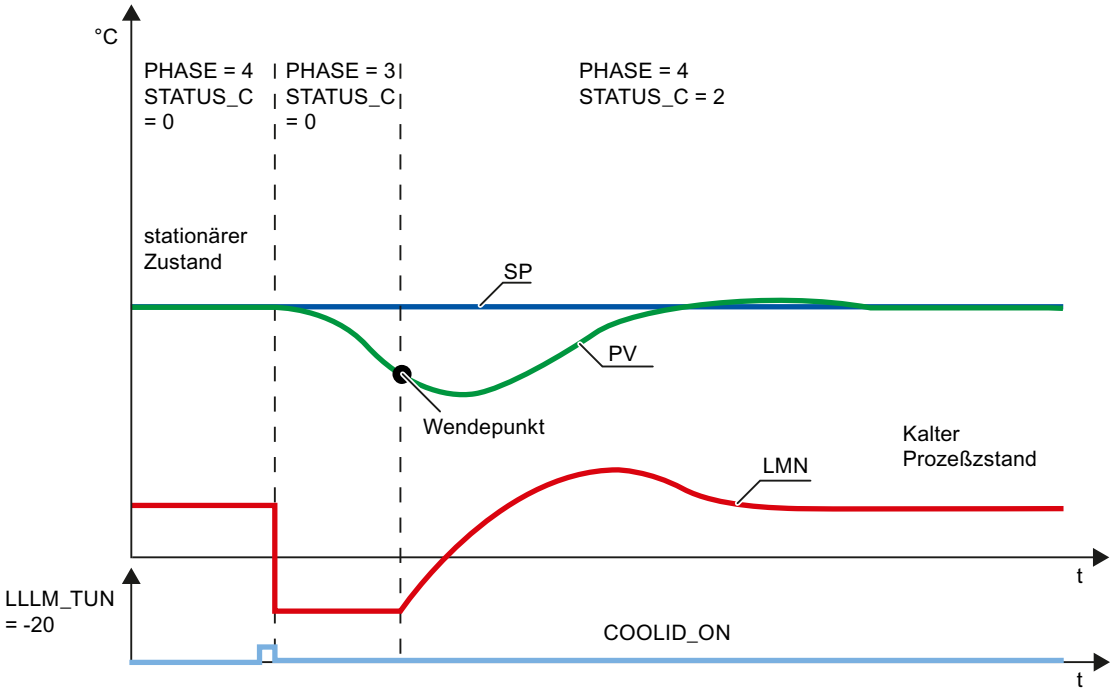

Hinweis

- Das Startbit COOLID_ON kann gleichzeitig mit ADAPT1ST oder ADAPT_ON oder während einer laufenden Optimierung der Reglerparameter gesetzt werden. Dadurch wird nach Beendigung der Optimierung der Reglerparameter für das Heizen sofort die Kühloptimierung gestartet.
- Bei einer Anlage mit mehreren thermisch gekoppelten Temperaturzonen (Bsp. Kunststoffmaschinen) sollte die Kühloptimierung erst dann gestartet werden, wenn alle (!) Zonen die Erstoptimierung beendet haben, da sonst die Ergebnisse der Erstoptimierung verfälscht werden.

Optimierung abbrechen

Um die Optimierung vor dem Sollwertsprung abzubrechen, müssen Sie ADAPT_ON = FALSE setzen. Um die Optimierung nach dem Sollwertsprung (PHASE 2 oder 3) oder die Kühloptimierung abzubrechen, müssen Sie auf Handbetrieb (MAN_ON = TRUE) umschalten.

Durch das Setzen von ADAPT_ON, ADAPT1ST oder COOLID_ON während einer aktiven Optimierung (PHASE 2 oder 3) wird die Optimierung nicht abgebrochen sondern bis zu Ende durchgeführt. Da ADAPT_ON, ADAPT1ST oder COOLID_ON nicht zurückgesetzt werden, wird nach Erfüllung der Startbedingungen erneut eine Optimierung gestartet.

Phasen der Optimierung

Bei der Regleroptimierung werden einzelne Phasen durchlaufen, die Sie am Parameter PHASE ablesen können.

PHASE = 0

Bei Erstellung eines Instanz-DB zu TUN_EC ist der Parameter PHASE mit Null und MAN_ON mit TRUE vorbelegt. Daher wird nach dem ersten Zyklus in den Handbetrieb (PHASE = 7) gewechselt. Sie können direkt vom Handbetrieb aus die Erstoptimierung starten (ADAPT_ON = TRUE) oder auf Automatikbetrieb umschalten. Beim erstmaligen Umschalten in den Automatikbetrieb MAN_ON = FALSE werden die Parameter von PI_CON übernommen. Die Vorbelegungswerte für GAIN und TI lauten: GAIN = 1.5, TI = 1h.

PHASE = 2

Sobald Sie einen Sollwertsprung ≥ MIN_STEP in positiver Richtung zum Arbeitspunkt des warmen Prozesszustandes vorgeben, wird bei Erstoptimierung MAN_OUT mit LHLM_TUN und bei Nachoptimierung MAN_OUT mit einem von TUN_EC berechneten Wert belegt. Zusätzlich wird QMAN_ON = TRUE gesetzt. Beide Werte werden an den PID-Regler übertragen. Der PID-Regler wird damit im Handbetrieb gesteuert. MIN_STEP sollte größer als 10 % des Arbeitsbereichs von Soll- und Istwert sein.

PHASE = 3

Wenn der Wendepunkt der Sprungantwort erkannt wird (STATUS_H = 2) oder der Istwert 70 % der Sprunghöhe des Sollwertes erreicht hat (STATUS_H = 3), wird ein vorsichtig eingestellter PI-Regler entworfen. Die Reglerverstärkung GAIN wird auf den Bereich 0.2 bis 20.0 begrenzt. Der Regler arbeitet ab sofort als PI-Regler und versucht die Strecke in einen stationären Zustand einzuregeln. TUN_EC legt die neu ermittelten PI-Reglerparameter in GAIN, TI und PI_CON ab. Die alten Werte des PI-Reglers werden in PI_CON_OLD gesichert. Dauert es sehr lange, bis der stationäre Zustand erreicht ist (kriechendes Einschwingverhalten bei Temperaturprozessen), so können Sie schon beim annähernd stationären Zustand durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten anstoßen. Auch zu einem späteren Zeitpunkt in PHASE = 3 oder PHASE = 4 können Sie durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten neu anstoßen. Oft wird der Reglerentwurf dadurch noch etwas verbessert.

Sollte ein Überschwinger auftreten oder wird kein Wendepunkt gefunden, so kann das bei der Erstoptimierung an einer zu groß gewählten Stellwertsprunghöhe LHLM_TUN liegen und muss nicht zwangsläufig zu einer schlechten Reglereinstellung führen. Bei der nächsten Erstoptimierung sollten Sie LHLM_TUN ca. 20 % kleiner wählen.

Wenn TUN_EC den stationären Zustand erkannt hat oder die Zeit 30 × TI (TI: Integrationszeit des unter PHASE = 3 eingestellten PI-Reglers) seit dem Sollwertsprung verstrichen ist, wird ein verbesserter Reglerentwurf vorgenommen und auf PHASE = 4 umgeschaltet. TUN_ES legt die neu ermittelten PI/PID-Reglerparameter in GAIN, TI, TD und PI_CON bzw. PID_CON ab. Die alten Werte des PID-Reglers werden in PID_CON_OLD gesichert. Wenn möglich, wird ein PID-Regler und ein PI-Regler entworfen. Falls PID_ON = TRUE, wird der PID-, anderenfalls der PI-Regler aktiviert. GAIN wird so begrenzt, dass die Kreisverstärkung des offenen Kreises (Produkt aus GAIN und Prozessverstärkung) unter 80.0 liegt.

PHASE = 4

In dieser Phase regelt der Regler mit seinen optimierten Parametern. Aus diesem Zustand heraus können Sie eine Kühloptimierung, eine Nachoptimierung oder eine Erstoptimierung starten.

Sie können jederzeit zwischen dem PI- und dem PID-Regler stoßfrei umschalten.

Erstoptimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_EC ist mit einem kontinuierlichen Regler oder einem Impulsregler verschaltet.
- MAN_ON = FALSE
- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um bei der ersten Inbetriebnahme die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Erstoptimierung".
   TUN_EC ist in Optimierbereitschaft.
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie im Feld "Stellwert" einen Stellwert ein, der bis zum Übergang in PHASE = 3 verwendet wird.
5. Geben Sie im Feld "Sollwert" einen Sollwert ein. Der Sollwertsprung muss ≥ MIN_STEP sein. Erst durch einen anderen Sollwert wird der neue Stellwert wirksam.
6. Klicken Sie auf das Symbol "Optimierung starten".
   Sollwert und Stellwert werden an den Regler übertragen. Die Erstoptimierung startet. Der Status der Optimierung wird angezeigt.

Nachoptimierung durchführen

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_EC ist mit einem kontinuierlichen Regler oder einem Impulsregler verschaltet.
- MAN_ON = FALSE
- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um am Arbeitspunkt die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Nachoptimierung".
   TUN_EC ist in Optimierbereitschaft.
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie im Feld "Sollwert" einen Sollwert ein. Der Sollwertsprung muss ≥ MIN_STEP sein.
5. Klicken Sie auf das Symbol "Optimierung starten".
   Der neue Sollwert wird an den Regler gesendet. TUN-EC berechnet einen Stellwertsprung. Die Nachoptimierung startet. Der Status der Optimierung wird angezeigt.

Kühloptimierung durchführen

Während der Kühloptimierung wird der Verhältnisfaktor für Regelungen mit zwei entgegengesetzt wirkenden Stellgliedern (z. B. Heizen/Kühlen) ermittelt.

Voraussetzungen

- Die Erstoptimierung wurde korrekt durchgeführt.
- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_EC ist mit einem kontinuierlichen Regler oder einem Impulsregler verschaltet.
- MAN_ON = FALSE
- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um den Verhältnisfaktor zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Kühloptimierung".
   TUN_EC ist in Optimierbereitschaft.
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie im Feld "Stellwert" einen Stellwert ein, der solange verwendet wird bis der Wendepunkt gefunden wird.
5. Klicken Sie auf das Symbol "Optimierung starten".
   Die Kühloptimierung startet. Der Status der Optimierung wird angezeigt.

Manuelle Optimierung durchführen

Wenn nach einer Ersteinstellung oder Nachoptimierung im geschlossenen Regelkreis Schwingungen auftreten oder Überschwinger nach Sollwertsprüngen vorhanden sind, können Sie die PID-Parameter manuell verändern. Sie können z. B. die Reglerverstärkung GAIN auf 80 % des ursprünglichen Werts zurückstellen und die Integrationszeit TI auf 150 % des ursprünglich eingestellten Werts vergrößern. Bei einem Impulsregler können kleine Dauerschwingungen auftreten, die Sie durch Vergrößern der Totzone DEADB_W eliminieren können.

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Manuell".
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie in den Feldern "P", "I", und "D" neue PID-Parameter ein.
5. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an Regler senden".
6. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
7. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
8. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.
9. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
10. Wiederholen Sie die Schritte 3 bis 9, bis Sie mit dem Regelergebnis zufrieden sind.

3.5.3.2 Schrittregler optimieren

TUN_ES mit Regler verschalten

Die Anweisung TUN_ES dient zur Optimierung eines Schrittreglers. Sie verschalten TUN_ES je nach Regler mit einer der folgenden Anweisungen:

Regler	Anweisung
FM 355 S Schrittregler	PID_FM
FM 455 S Schrittregler	PID_FM
PID Basisfunktionen	CONT_S TCONT_S

Regler	Anweisung
Standard PID Control (Optionspaket)	PID_ES
Modular PID Control (Optionspaket)	PID LMNGEN_S

Verschalten Sie die Ein- und Ausgangsparameter der Anweisung TUN_ES folgendermaßen mit der Anweisung des Reglers.

TUN_ES	PID_FM	CONT_S	TCONT_S	PID_ES
PV	PV	PV	PV	PV
LMNR	LMN_A	--	--	LMNR_IN
NORM_FAC	--	--	--	(NM_PVHR-NM_PVLR)*0.01
C_LMNUP	QLMNUP	QLMNUP	QLMNUP	QLMNUP
C_LMNDN	QLMNDN	QLMNDN	QLMNDN	QLMNDN
LMNR_HS	QLMNR_HS	LMNR_HS	LMNR_HS	LMNR_HS
LMNR_ON	QLMNR_ON	--	--	LMNR_ON
WRITE_DIS	Oderverknüpfung aus LOAD_OP und LO-AD_PAR	--	--	--
MAN_OUT	LMN_RE	--	--	MAN
SP_OUT	SP_RE	SP_INT	SP_INT	SP_INT oder SP_EXT
GAIN	GAIN	GAIN	GAIN	GAIN
TI	TI	TI	TI	TI
TD	TD	--	--	TD
TM_LAG	TM_LAG	--	--	TM_LAG
MTR_TM	MTR_TM	MTR_TM	MTR_TM	MTR_TM
DEADB_W	DEADB_W	DEADB_W	DEADB_W	DEADB_W
PHASE	--	--	--	PHASE
QMAN_ON	LMN_REON	--	--	MAN_ON
QLMNS_ON	LMNSOPON	LMNS_ON	LMNS_ON	LMNS_ON
QLMNUP	LMNUP_OP	LMNUP	LMNUP	LMNUP
QLMNDN	LMNDN_OP	LMNDN	LMNDN	LMNDN
QI_SEL	--	--	--	I_SEL
QD_SEL	--	--	--	D_SEL
QWRITE	LOAD_PAR und LOAD_OP	--	--	--

Möglichkeiten der Optimierung

Erstoptimierung

Die Optimierung der PID-Parameter erfolgt während des Aufheizvorgangs. Dabei wird eine Prozessidentifikation mit anschließendem Reglerentwurf durchgeführt.

Sie können eine Erstoptimierung mit ADAPT1ST anfordern.

Wenn Sie ADAPT1ST = TRUE setzen, dann wird ADAPT_ON = TRUE und das Ventil geschlossen (PHASE = 1).

Bei ADAPT1ST = TRUE wird die Initialisierungsroutine der Anweisung TUN_ES durchlaufen. Dabei werden folgende Vorbelegungen gemacht:

- Es wird auf PI-Regler geschaltet. Hierzu werden die Parameter von PI_CON an den Regler übertragen. Haben die Parameter PI_CON.GAIN und PI_CON.TI beide den Wert 0.0, dann werden Sie mit PI_CON.GAIN = 1.5 und PI_CON.TI = 3600.0 s vorbelegt.
- PROCESS.GAIN = 999.0
- STATUS_H = 0; STATUS_D = 0

Hinweis

Wenn nicht explizit gewünscht, rufen Sie bei Neustart (Warmstart) der CPU TUN_ES **nicht** wie bei anderen Anweisungen üblich in seiner Initialisierungsroutine auf, da sonst alle Prozessparameter verloren gehen.

Nach einem Sollwertsprung ≥ MIN_STEP wird die Motorstellzeit gemessen und anschließend eine Erstoptimierung durchgeführt. Bei einer Erstoptimierung verwendet TUN_ES für die Prozessanregung den Parameter LHLM_TUN zum Stellwertsprung. Bei Schrittreglern mit Stellungsrückmeldung wird der Wert LHLM_TUN direkt angefahren. Bei Schrittreglern ohne Stellungsrückmeldung wird das Stellventil ganz geöffnet und so lange wieder zugefahren, bis der durch (100 - LHLM_TUN) vorgegebene Anteil der Motorstellzeit verstrichen ist.

Erstoptimierung mit Messung der Motorstellzeit beim Schrittregler ohne Stellungsrückmeldung

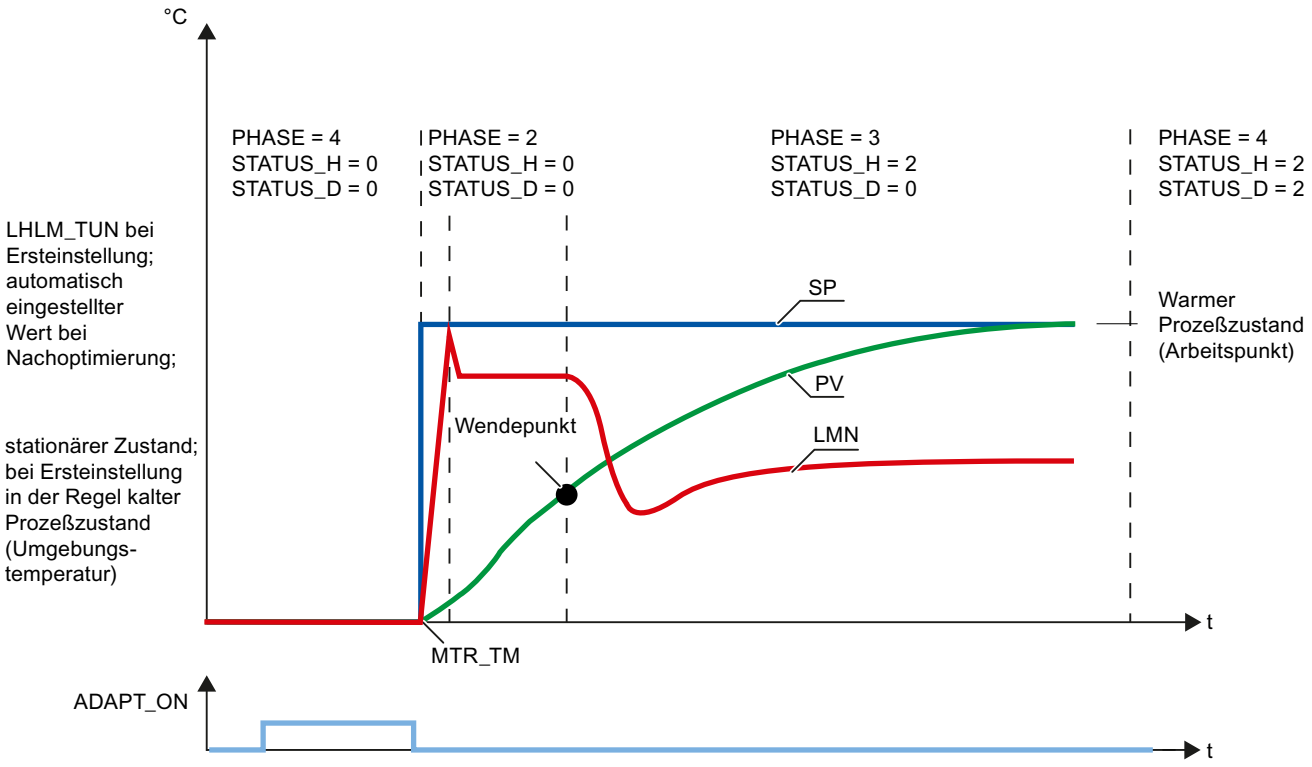

Nachoptimierung

Nur bei Schrittreglern mit Stellungsrückmeldung können Sie mit ADAPT_ON die Optimierung starten. TUN_ES entscheidet an Hand alter Prozesswerte, ob eine Erstoptimierung ohne Messung der Motorstellzeit oder eine Nachoptimierung durchgeführt wird. Mit ADAPT_ON = TRUE und einem anschließenden Sollwertsprung ≥ MIN_STEP in positiver Richtung wird die Optimierung gestartet. TUN_ES geht dann sofort in PHASE = 2 und führt einen Stellwertsprung auf LHLM_TUN bei Erstoptimierung oder auf einen automatisch berechneten Stellwert bei Nachoptimierung durch.

Der Sollwertsprung bei der Erstoptimierung geht in der Regel vom Sollwert des kalten Prozesses (Umgebungstemperatur) bis in die Nähe des Arbeitspunktes. Es ist aber auch ein beliebiger Sollwert als Anfangszustand möglich, jedoch muss es sich um einen stationären Zustand handeln. Während der Optimierung dürfen Sie keine weiteren Sollwertsprünge vorgeben.

Optimierung starten

Sie können eine Optimierung sowohl aus dem Automatikbetrieb heraus als auch aus dem Handbetrieb heraus starten:

- Starten der Optimierung aus dem Automatikbetrieb
  Setzen Sie im Automatikbetrieb ADAPT_ON oder ADAPT1ST = TRUE und geben Sie einen Sollwertsprung ≥ MIN_STEP vor.
- Starten der Optimierung aus dem Handbetrieb
  Setzen Sie im Handbetrieb ADAPT_ON oder ADAPT1ST = TRUE und geben Sie einen Sollwertsprung ≥ MIN_STEP vor. Erst wenn Sie anschließend MAN_ON = FALSE zurücksetzen, wird der Sollwertsprung wirksam und die Optimierung beginnt.

Optimierung abbrechen

Um die Optimierung vor dem Sollwertsprung abzubrechen, müssen Sie ADAPT_ON = FALSE setzen. Um die Optimierung nach dem Sollwertsprung (PHASE 2 oder 3) abzubrechen, müssen Sie auf Handbetrieb umschalten (MAN_ON = TRUE oder LMNS_ON = TRUE).

Durch das Setzen von ADAPT_ON oder ADAPT1ST während einer aktiven Optimierung (PHASE 2 oder 3) wird die Optimierung nicht abgebrochen sondern bis zu Ende durchgeführt. Da ADAPT_ON oder ADAPT1ST nicht zurückgesetzt werden, wird nach Erfüllung der Startbedingungen erneut eine Optimierung gestartet.

Phasen der Optimierung

Bei der Regleroptimierung werden einzelne Phasen durchlaufen, die Sie am Parameter PHASE ablesen können.

PHASE = 0

Bei Erstellung eines Instanz-DB zu TUN_ES ist der Parameter PHASE mit Null und LMNS_ON mit TRUE vorbelegt. Daher wird nach dem ersten Zyklus in den Handbetrieb (PHASE = 7) gewechselt. Sie können direkt vom Handbetrieb aus die Erstoptimierung starten (ADAPT_ON = TRUE) oder auf Automatikbetrieb umschalten. Beim erstmaligen Umschalten in den Automatikbetrieb MAN_ON = FALSE werden die Parameter von PI_CON übernommen. Die Vorbelegungswerte für GAIN und TI lauten: GAIN = 1.5, TI = 1h.

PHASE = 1

Nach dem Starten der Optimierung über ADAPT1ST = TRUE fährt TUN_ES das Stellventil zu (Stellwert = 0 bzw. LMNS_ON = LMNDN = TRUE) und wartet auf einen positiven Sollwertsprung ≥ MIN_STEP.

PHASE = 2

Sobald Sie einen Sollwertsprung ≥ MIN_STEP in positiver Richtung zum Arbeitspunkt des warmen Prozesszustandes vorgeben, wird bei Erstoptimierung MAN_OUT mit LHLM_TUN und bei Nachoptimierung MAN_OUT mit einem von TUN_ES berechneten Wert belegt. MIN_STEP sollte größer als 10 % des Arbeitsbereichs von Soll- und Istwert sein.

Bei der Schrittregelung mit Stellungsrückmeldung (LMNR_ON=TRUE) erhält QMAN_ON den Wert TRUE. Der Regler steuert das Ventil auf den Wert von MAN_OUT, und TUN_ES berechnet die Motorstellzeit MTR_TM.

Bei der Schrittregelung ohne Stellungsrückmeldung werden QLMNS_ON und QLMNUP auf TRUE gesetzt, und das Ventil wird bis zum oberen Anschlag gefahren. Mit Erreichen des oberen Anschlags (LMNR_HS =TRUE) berechnet TUN_ES die Motorstellzeit und gibt sie an den Regler weiter. Anschließend wird das Ventil bis auf den Wert von MAN_OUT geschlossen (QLMNDN = TRUE).

PHASE = 3

Wenn der Wendepunkt der Sprungantwort erkannt wird (STATUS_H = 2) oder der Istwert 70 % der Sprunghöhe des Sollwertes erreicht hat (STATUS_H = 3), wird ein vorsichtig eingestellter PI-Regler entworfen. Die Reglerverstärkung GAIN wird auf den Bereich 0.2 bis 20.0 begrenzt. Der Regler arbeitet ab sofort als PI-Regler und versucht die Strecke in einen stationären Zustand einzuregeln. TUN_ES legt die neu ermittelten PI-Reglerparameter in GAIN, TI und PI_CON ab. Die alten Werte des PI-Reglers werden in PI_CON_OLD gesichert. Dauert es sehr lange, bis der stationäre Zustand erreicht ist (kriechendes Einschwingverhalten bei Temperaturprozessen), so können Sie schon beim annähernd stationären Zustand durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten anstoßen. Auch zu einem späteren Zeitpunkt in PHASE = 3 oder PHASE = 4 können Sie durch Setzen von STEADY = TRUE den Reglerentwurf mit den aktuellen Daten neu anstoßen. Oft wird der Reglerentwurf dadurch noch etwas verbessert.

Sollte ein Überschwinger auftreten oder wird kein Wendepunkt gefunden, so kann das bei der Erstoptimierung an einer zu groß gewählten Stellwertsprunghöhe LHLM_TUN liegen und muss nicht zwangsläufig zu einer schlechten Reglereinstellung führen. Bei der nächsten Erstoptimierung sollten Sie LHLM_TUN ca. 20 % kleiner wählen.

Wenn TUN_ES den stationären Zustand erkannt hat oder die Zeit 30 × TI (TI: Integrationszeit des unter PHASE = 3 eingestellten PI-Reglers) seit dem Sollwertsprung verstrichen ist, wird ein verbesserter Reglerentwurf vorgenommen und auf PHASE = 4 umgeschaltet. TUN_ES legt die neu ermittelten PI/PID-Reglerparameter in GAIN, TI, TD und PI_CON bzw. PID_CON ab. Die alten Werte des PID-Reglers werden in PID_CON_OLD gesichert. Wenn möglich, wird ein PID-Regler und ein PI-Regler entworfen. Falls PID_ON = TRUE, wird der PID-, anderenfalls der PI-Regler aktiviert. Bei schwierigen Prozessen entwirft TUN_ES unabhängig von PID_ON immer einen PI-Regler. GAIN wird so begrenzt, dass die Kreisverstärkung des offenen Kreises (Produkt aus GAIN und Prozessverstärkung) unter 40.0 liegt.

PHASE = 4

In dieser Phase regelt der Regler mit seinen optimierten Parametern. Aus diesem Zustand heraus können Sie eine Nachoptimierung oder eine Erstoptimierung starten.

Sie können jederzeit zwischen dem PI- und dem PID-Regler stoßfrei umschalten.

Erstoptimierung durchführen

Die Erstoptimierung der Reglerparameter erfolgt während des Aufheizvorgangs. Dabei wird eine Prozessidentifikation mit anschließendem Reglerentwurf durchgeführt.

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_ES ist mit einem Schrittregler verschaltet.
- MAN_ON = FALSE
- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um bei der ersten Inbetriebnahme die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Erstoptimierung".
   TUN_ES ist in Optimierbereitschaft.
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie im Feld "Stellwert" einen Stellwert ein, der bis zum Übergang in PHASE = 3 verwendet wird.
5. Geben Sie einen neuen Sollwert ein. Der Sollwertsprung muss ≥ MIN_STEP sein.
6. Klicken Sie auf das Symbol "Optimierung starten".
   Die Erstoptimierung startet. Der Status der Optimierung wird angezeigt.

Nachoptimierung durchführen

Bei Schrittreglern mit Stellungsrückmeldung können Sie eine Nachoptimierung starten. Beim Wechsel des Arbeitspunkts oder bei Änderungen des Prozessverhaltens kann der Regler online während eines positiven Sollwertsprungs nachoptimiert werden.

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_ES ist mit einem Schrittregler verschaltet.
- MAN_ON = FALSE
- LMNR_ON = TRUE an der Anweisung TUN_ES und am Regler.
- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um bei der ersten Inbetriebnahme die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Nachoptimierung".
   TUN_ES ist in Optimierbereitschaft.
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie einen neuen Sollwert ein. Der Sollwertsprung muss ≥ MIN_STEP sein.
5. Klicken Sie auf das Symbol "Optimierung starten".
   Die Nachoptimierung startet. Der Stellwert wird von TUN_ES ermittelt. Der Status der Optimierung wird angezeigt.

Manuelle Optimierung durchführen

Wenn nach einer Ersteinstellung oder Nachoptimierung im geschlossenen Regelkreis Schwingungen auftreten oder Überschwinger nach Sollwertsprüngen vorhanden sind, können Sie die PID-Parameter manuel verändern. Sie können z. B. die Reglerverstärkung GAIN auf 80 % des ursprünglichen Werts zurückstellen und die Integrationszeit TI auf 150 % des ursprünglich eingestellten Werts vergrößern.

Voraussetzungen

- Die Anweisung und das Technologieobjekt sind auf die CPU geladen.
- TUN_ES ist mit einem Schrittregler verschaltet.
- MAN_ON = FALSE

- Der Regler befindet sich im Automatikbetrieb.
- Der Prozess befindet sich in einem stationären Zustand.

Vorgehen

Um manuell die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
2. Wählen Sie in der Liste "Modus" den Eintrag "Manuell".
3. Wählen Sie eine Reglerstruktur.
4. Geben Sie in den Feldern "P", "I", und "D" neue PID-Parameter ein.
5. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Parameter an Regler senden".
6. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
7. Geben Sie einen neuen Sollwert ein und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
8. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb".
   Der Regler arbeitet mit den neuen PID-Parametern und regelt auf den neuen Sollwert.
9. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
10. Wiederholen Sie die Schritte 3 bis 9, bis Sie mit dem Regelergebnis zufrieden sind.

3.5.4 Grundlagen Reglerbaugruppen

3.5.4.1 Eingänge

Einführung

An den Analogeingängen können unterschiedliche Sensortypen angeschlossen werden. Die Eingangssignale der Sensoren werden anschließend den Erfordernissen entsprechend aufbereitet.

Mithilfe der Digitaleingänge kann die Baugruppe in verschiedene Betriebsarten geschaltet werden.

K-Regler und S-Regler haben bei den Analog- und Digitaleingängen die gleiche Struktur.

Anzahl der Eingänge

Eingang	FM 355	FM 355-2	FM 455
Analogeingänge mit Analogwertaufbereitung	4	4	16
Vergleichsstelleneingang zur Kompensation von Thermoelementen	1	1	1
Digitaleingänge	8	8	16

Analogeingänge

Funktionsblöcke eines Analogeingangs

Das folgende Bild zeigt das Schema für die Analogwertaufbereitung:

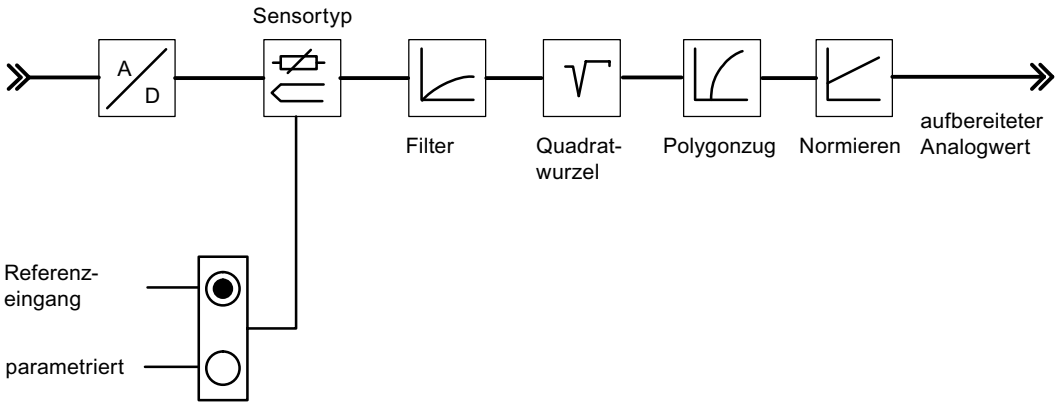

Anpassung an Sensoren

Die Analogeingänge können durch Parametrierung an verschiedene Sensoren angepasst werden. Folgende Einstellungen sind möglich:

- Analogeingang wird nicht bearbeitet (z. B. unbenutzter Eingang)
  Bei dieser Einstellung werden alle PID-Parameter auf Null gesetzt, der Reglerkanal wird nicht bearbeitet und es wird kein Stellwert berechnet.
- Stromsensor 0 bis 20 mA
- Stromsensor 4 bis 20 mA
- Spannungssensor 0 bis 10 V
- Pt 100, –200 ... 850 °C
- Pt 100, –200 ... 556 °C (doppelte Auflösung)
- Pt 100, –200 ... 130 °C (vierfache Auflösung)
- Thermoelemente Typ B, J, K, R und S (Analogeingang auf ±80 mV eingestellt)
- Freies Thermoelement (Analogeingang auf ±80 mV eingestellt)

Die Analogeingänge parametrieren Sie in der Maske "Analogeingang".

Umschaltung Celsius/Fahrenheit

Temperaturen können sowohl in °C als auch in °F gemessen werden.

Anpassung an Netzfrequenz

Zur Unterdrückung von Störungen bei der Messung analoger Signale kann die Eingangssignalbearbeitung an die Netzfrequenz angepasst werden. Folgende Einstellungen sind möglich:

- 50 Hz Betrieb
- 60 Hz Betrieb

Vergleichsstelle

Wenn Sie an einem Analogeingang ein Thermoelement als Sensor eingestellt haben, können Sie am Vergleichsstelleneingang der Baugruppe einen Pt 100 zur Kompensation der Vergleichsstellentemperatur bei Thermoelementen anschließen. Alternativ dazu kann eine feste Vergleichsstellentemperatur parametriert werden.

Bei der FM 355 können Sie alternativ eine interne Kompensation für die Thermoelemente J, K und E parametrieren. Ein interner Temperatursensor misst die Vergleichsstellentemperatur direkt in der Baugruppe.

Bei Nutzung des Vergleichsstelleneingangs verlängert sich die Abtastzeit jedes Reglers um die Wandlungszeit für den Vergleichsstelleneingang.

Analogwertaufbereitung

Die Analogwertaufbereitung bietet verschiedene parametrierbare Möglichkeiten, die Eingangssignale aufzubereiten. Nachfolgende Tabelle gibt einen Überblick über diese Parameter und die einstellbaren Werte.

Parameter	Einstellbare Werte	Anmerkungen
Auflösung	• 12 Bit • 14 Bit	Wandlungszeit 20 ms (50 Hz) Wandlungszeit $16^2/_3$ ms (60 Hz) Wandlungszeit 100 ms
Filter	• Ein/Aus • Zeitkonstante in s	Filter 1. Ordnung, dessen Zeitverhalten durch die Zeitkonstante festgelegt wird
Quadratwurzel	• Ein/Aus	Zur Linearisierung von Gebersignalen, bei denen der Istwert als physikalische Größe vorliegt, die in quadratischem Zusammenhang mit der gemessenen Prozessgröße steht

Parameter	Einstellbare Werte	Anmerkungen
Polygonzug	• Ein/Aus • 13 Stützpunkte wählbar in – mA bei Stromeingang – mV bei Spannungseingang	Zur Linearisierung von Geberkennlinien
Normierung	• Oberer Wert • Unterer Wert	Zur Wandlung des Eingangssignals in eine andere physikalische Einheit durch lineare Interpolation zwischen Anfangswert (unten) und Endwert (oben)

Hinweis

Normierung/Polygonzug: Die Wandlung der Einheit mA bzw. mV in eine physikalische Einheit erfolgt entweder über den Polygonzug oder – falls dieser nicht eingeschaltet ist – über die Normierung. Der Polygonzug wird zur Linearisierung eines freien Thermoelements oder für eine beliebige Linearisierung eingesetzt.

Parameter kopieren

Die FMs bieten am analogen Eingang die Möglichkeit, Parameter eines Eingangs auf einen oder mehrere andere Eingänge zu übertragen. Dabei werden alle eingestellten Parameter des Ausgangskanals auf die Zielkanäle übertragen.

Digitaleingänge

Betriebsarten

Die Digitaleingänge dienen dem Umschalten der Betriebsarten einzelner Reglerkanäle.

Der Wirksinn der Digitaleingänge ist parametrierbar. Für jeden einzelnen Digitaleingang sind folgende Einstellungen möglich:

- High-aktiv
- Low-aktiv bzw. offen

Folgende Betriebsarten können Sie wählen:

- Umschaltung auf Stellwertvorgabe durch den FB PID_FM
- Umschaltung auf Nachführbetrieb (Stellwertvorgabe über einen Analogeingang)
- Umschaltung auf Sicherheitsstellwert

Beim Schrittregler können Sie zudem über Digitaleingänge folgende Signale vorgeben:

- Rückmeldung: Stelleinrichtung am oberen Anschlag
- Rückmeldung: Stelleinrichtung am unteren Anschlag

Paralleloptimierung

Die Funktion "Paralleloptimierung" steht ausschließlich für die FM 355-2 zur Verfügung.

Nachbar-Zonen (starke Wärmekopplung)

Regeln zwei oder mehrere Regler einer Baugruppe die Temperatur z. B. auf einer Platte (z.B. zwei Heizungen und zwei gemessene Istwerte mit starker Wärmekopplung), haben Sie die Möglichkeit, eine parallele Optimierung für diese Regler durchzuführen. Gehen Sie dazu folgendermaßen vor:

1. Definieren Sie die zusammengehörigen Kanäle in Zone A oder Zone B.
2. Starten Sie die Optimierung in einem der Kanäle. Die Optimierungen der zugehörigen Kanäle werden dann automatisch gestartet.

Hinweis

Bei Anregung über Sollwertsprung müssen jedoch Sie als Anwender für ein gleichzeitiges Setzen der Sollwertsprünge sorgen. Über den Assistenten können Sie die Parameter PID_ON und TUN_DLMN/TUN_CLMN nur für einen Kanal vorgeben. Die Parameter für die übrigen Kanäle müssen Sie über die entsprechenden Instanz-DBs des FB FMT_PID vor Start der Optimierung vorgeben.

Abbruch der Optimierung

Tritt ein Fehler in einem der Kanäle auf (Phase 0 oder Ende von Phase 1), wird die Optimierung aller Kanäle abgebrochen.

Die Optimierung kann vom Anwender durch Setzen von TUN_ON=FALSE in irgendeinem der beteiligten Kanäle zurückgesetzt werden.

Geht ein Kanal in den Sicherheitsbetrieb, wird auch die Optimierung der beteiligten Kanäle abgeschaltet.

Beim Assistenten werden nur die Statusinformationen des ausgewählten Kanals angezeigt. Fehlerursachen der übrigen Kanäle finden Sie in den jeweiligen Instanz-DBs.

Vorteil

Die beteiligten Regler geben solange ihre jeweilige Optimierungsstellgröße aus, bis alle Regler Phase 2 verlassen haben. Damit wird vermieden, dass der Regler, der die Optimierung früher beendet, durch die Änderung seiner Stellgröße das Optimierungsergebnis der übrigen Regler verfälscht.

ACHTUNG
Das Erreichen von 75% des Sollwertsprunges bei einem der beteiligten Kanäle führt nicht zu einem Beenden der Optimierung (Gefahr des Überschwingens). Der Automatikbetrieb wird erst begonnen, wenn alle beteiligten Regler ihre Optimierung beendet haben.

Nachbar-Zonen (schwache Wärmekopplung)

Generell gilt, dass man so optimieren sollte, wie man später regelt. Werden im Produktionsbetrieb die Zonen gemeinsam parallel verfahren, so dass die Temperaturunterschiede zwischen den Zonen gleich bleiben, sollte man auch bei der Optimierung das Temperaturniveau der Nachbarzonen entsprechend mit anheben.

Die Temperaturunterschiede zu Beginn des Versuchs spielen keine Rolle, da sie durch eine entsprechende Anfangsheizung ausgeglichen werden (Anfangssteigung = 0).

3.5.4.2 Regeldifferenz

Kanäle kopieren

Die FMs bieten beim Regler die Möglichkeit, einen Reglerkanal auf einen oder mehrere andere Kanäle zu übertragen. Dabei werden alle Einstellungen des Ausgangskanals auf die Zielkanäle übertragen.

Reglerarten

Bei allen Reglerbaugruppen wird die Regeldifferenz nach der gleiche Struktur gebildet.

1. Das Signal für den Istwert wird zum wirksamen Istwert aufbereitet.
2. Das Signal für den Sollwert wird zum wirksamen Sollwert aufbereitet.
3. Der wirksame Sollwert wird vom wirksamen Istwert abgezogen.
4. Diese Regeldifferenz wird dem Regelalgorithmus zugeführt.

Für Soll- und Istwert können Sie die Signalquellen wählen und dadurch die Reglerbaugruppe einsetzen als:

- Festwert- und Kaskadenregler
- Dreikomponentenregler
- Verhältnis- und Mischungsregler

Festwert- und Kaskadenregler

Beim Folgeregler einer Kaskadenregelung wird als Sollwert der Stellwert eines Führungsreglers ausgewählt.

Dreikomponentenregler

Der Dreikomponentenregler verfügt über drei Eingänge für den Istwert. Istwert A, Istwert B und Istwert C werden addiert und ergeben die Gesamtmenge PV bei einer Mischungsregelung.

- Mischungsregler
  Ein Mischungsregler ist immer ein Folgeregler. Der zugehörige Führungsregler ist ein Dreikomponentenregler.
  Die Stellgröße des Führungsreglers wird über den Eingang Istwert D verschaltet. Der Mischungsfaktor wird über den Sollwerteingang des Reglers vorgegeben.

Verhältnisregler

Ein Verhältnisregler ist immer ein Folgeregler. Der zugehörige Führungsregler ist ein Festwertregler.

Als Istwert D wird der Istwert des Führungsreglers ausgewählt. Der Verhältnisfaktor wird über den Sollwerteingang vorgegeben. Wenn als Verhältnisfaktor ein Reglerausgang angewählt ist, dann wird der Sollwert mit Hilfe der unteren und oberen Schranke von "0 .. 100%" auf den Wertebereich "untere Schranke ... obere Schranke" gewandelt (normiert).

Mischungsregler

Ein Mischungsregler ist immer ein Folgeregler. Der zugehörige Führungsregler ist ein Dreikomponentenregler.

Als Istwert D wird die Stellgröße des Führungsreglers gewählt. Der Mischungsfaktor wird über den Sollwerteingang des Reglers vorgegeben.

Die Stellgröße des Gesamtmengenreglers wird im Wertebereich 0% bis 100% vorgegeben. Vom Folgeregler wird diese Größe am Istwerteingang D in den Wertebereich des Istwertes A umgerechnet. Wertebereich des Istwertes A sind die Normierungswerte "oben" und "unten" des ausgewählten Analogeingangs.

Wenn die Stellgröße eines Folgereglers in die Begrenzung gerät oder wenn die Sollwertsteigung eines Folgereglers durch die Rampenfunktion im Sollwertzweig begrenzt ist, dann wird der I-Anteil des Führungsreglers richtungsabhängig blockiert (Anti-Reset-Windup), bis die Ursache für die Begrenzung im Folgeregler beseitigt ist.

Signalauswahl für Sollwert, Istwert, D-Eingang und Störgröße

Signalauswahl

Für den Sollwert, den Istwert, den Wert des D-Eingangs (Differenziereingang) und die Störgröße jedes Reglerkanals können Sie eine Auswahl unter verschiedenen Signalquellen treffen. Nachfolgende Tabelle gibt einen Überblick über die Möglichkeiten der Signalauswahl.

Tabelle 3-2 Signalauswahl für Sollwert, Istwert, D-Eingang und Störgröße

Betroffene Werte	Auswählbare Signalquelle
Sollwert	• Ein vom Anwenderprogramm durch den Funktionsbaustein vorgegebener Wert • Der aufbereitete Analogwert eines Analogeingangs • Der Stellwert (LMN, LMN_A oder LMN_B) eines anderen Reglerkanals (bei Kaskadierung von Reglern)
Istwerte A, B und C	• Null • Der aufbereitete Analogwert eines Analogeingangs (die Istwerte B und C können zusätzlich durch Faktoren bewertet werden)
Istwert D	• Null • Der aufbereitete Analogwert eines Analogeingangs • Stellwert eines anderen Reglerkanals

Betroffene Werte	Auswählbare Signalquelle
Wert für D-Eingang (nur relevant bei PD- oder PID-Regler)	• Die Regeldifferenz nach der Totzone des eigenen Reglerkanals • Der aufbereitete Analogwert eines Analogeingangs • Der negierte wirksame Istwert des eigenen Reglerkanals
Störgröße	• Null • Der aufbereitete Analogwert eines Analogeingangs

Aufbereitung Sollwert

Die Aufbereitung des Sollwerts zum wirksamen Sollwert können Sie durch folgende Funktionen beeinflussen:

Quelle für den Sollwert

Der Sollwert kann vorgegeben werden von:

- Einer Anweisung
- Einem Analogeingang
- Einem aufbereiten Stellwert eines anderen Reglerkanals der Baugruppe
  Diese Quelle für den Sollwert wird gewählt bei einem Mischungsregler oder Folgeregler in einer Kaskadenregelung.
- Einem aufbereiten Stellwert A eines anderen Reglerkanals der Baugruppe (nur FM 355-2)
- Einem aufbereiten Stellwert B eines anderen Reglerkanals der Baugruppe (nur FM 355-2)

Sicherheitssollwert

Bei CPU-Ausfall oder CPU-Stop kann die Baugruppe auf einen Sicherheitssollwert regeln.

Dazu müssen Sie einen Sicherheitssollwert eingeben und als Reaktion bei CPU-Stop "Sollwert = Sicherheitssollwert" einstellen.

Reaktion bei CPU-Ausfall

Die Reglerbaugruppe kann bei CPU-Ausfall oder CPU-Stop auf folgende Sollwerte regeln:

- Sollwert = letzter gültiger Sollwert
  Ist als Sollwertauswahl "von Anweisung" eingestellt, dann bleibt der Sollwert nach dem CPU-Ausfall oder CPU-Stop konstant auf dem zuletzt vorgegebenen Wert.
  Wenn der Sollwert von dem Stellwert eines Reglers der FM oder von einem Analogeingang vorgegeben wird, dann ändert sich der Sollwert entsprechend dem angewählten Wert.
- Sollwert = Sicherheitssollwert
  Wenn Sie einen Sicherheitssollwert eingegeben haben, regelt die Baugruppe auf diesen Sollwert.

Reaktion bei Anlauf der Baugruppe (nur FM 355 und FM 455)

Die Reglerbaugruppe kann bei Anlauf der Baugruppe auf folgende Sollwerte regeln:

- Sollwert = letzter gültiger Sollwert
- Sollwert = Sicherheitssollwert

Rampe

Durch Wahl einer Hochlaufzeit vom physikalischen Anfangs- auf den Endwert können Sie die Änderungsgeschwindigkeit des Sollwertes begrenzen.

Grenzen des Sollwerts

Wenn der Sollwert vom Funktionsbaustein vorgegeben wird oder ein aufbereiteter Analogwert eines Analogeingangs ist, wird der Sollwert begrenzt auf eine vorgebbare untere und obere Grenze.

Wenn beim Verhältnisregler als Sollwert der aufbereitete Stellwert eines anderen Reglerkanals gewählt ist, dann wirkt dieser Wert als Faktor für die Multiplikation des Istwerts D. Der Sollwert, der am Eingang in % vorliegt, wird in diesem Fall mit Hilfe der unteren und oberen Grenze normiert.

Wenn beim Festwert- oder Kaskadenregler als Sollwert der Stellwert eines anderen Reglerkanals benutzt wird, dann wird dieser mit Hilfe der Normierungskonstanten des angewählten Istwertkanals auf einen physikalischen Wert normiert.

Multiplizieren

Beim Reglertyp Verhältnisregler wird der Istwert A als Regelgröße und der Istwert D als Verhältnisgröße verwendet. Der Sollwerteingang dient als Verhältnisfaktor. Er wird durch Multiplikation mit dem Istwert D und Addition eines einstellbaren Offsets zum wirksamen Sollwert aufbereitet. Wird der Istwert D abgeschaltet, dann wird zum Sollwert nur der Offset addiert.

Aufbereitung Istwert

Die Aufbereitung des Istwerts zum wirksamen Istwert ist abhängig von der Reglerstruktur.

Wirksamer Istwert beim Festwert- oder Kaskadenregler

Der wirksame Istwert ist identisch mit dem Istwert A. Quelle für den Istwert A kann ein Analogeingang sein. Die Istwert B, Istwert C und Istwert D werden nicht berücksichtigt.

Wirksamer Istwert beim Dreikomponentenregler

Der wirksame Istwert wird berechnet, indem die Istwerte A, B und C sowie ein einstellbarer Offset addiert werden. Die Istwerte B und C können zusätzlich durch Faktoren bewertet werden. Istwert D wird nicht berücksichtigt.

Wirksamer Istwert beim Verhältnisregler

Beim Verhältnisregler wählen Sie folgende Istwerte

- Istwert A: Das aufbereitete Signal eines Analogeingangs.
- Istwert D: Der Istwert des Führungsreglers.

Der Verhältnisfaktor wird als Sollwert vorgegeben.

Wenn als Verhältnisfaktor FAC ein Reglerausgang angewählt ist, dann wird der Sollwert mit Hilfe der unteren und oberen Schranke von "0 ... 100%" auf den Wertebereich "untere Schranke ... obere Schranke" gewandelt (normiert).

Wirksamer Istwert beim Mischungsregler

Beim Mischungsregler wählen Sie folgende Istwerte

- Istwerte A, B und C: Aufbereitete Signale der zugehöriger Analogeingänge.
- Istwert D: Der Istwert des Führungsreglers.

Die Verhältnisfaktoren der einzelnen Regelstrecken werden als Sollwert vorgegeben.

Die Stellgröße LMN des Gesamtmengenreglers wird im Wertebereich "0 ... 100%" vorgegeben. Vom Folgeregler wird diese Größe am Eingang Istwert D in den Wertebereich "untere Schranke ... obere Schranke" des Istwertes A gewandelt (normiert).

Die Stellgröße des Führungsreglers wird über den Eingang "Istwert D" des Folgereglers verschaltet. Der Verhältnisfaktor FAC wird über den Sollwerteingang des Reglers vorgegeben.

Alarm

Grenzwertüberwachung

In der Reglerbaugruppe ist eine Grenzwertüberwachung realisiert. Diese erlaubt,

- entweder die Regeldifferenz
- oder den wirksamen Istwert

auf eine obere und untere Warngrenze und auf eine obere und untere Alarmgrenze zu überwachen. Außerdem können Sie für diese Grenzen eine Hysterese einstellen.

Das folgende Bild zeigt die Hysterese für Warn- und Alarmgrenze:

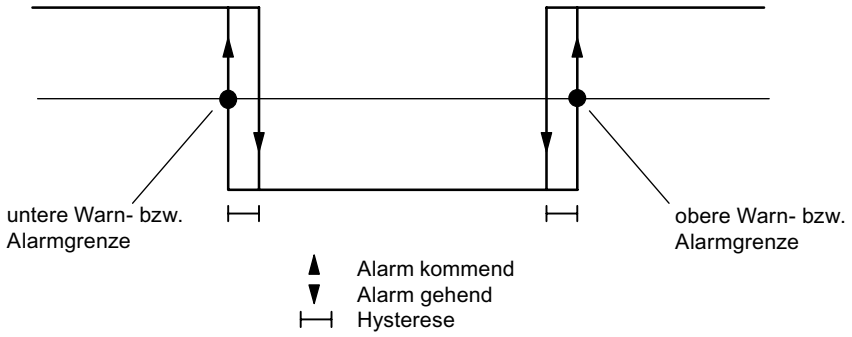

3.5.4.3 Regelalgorithmus

Regelalgorithmus FM 355-2

Komponenten des Regelalgorithmus

Kontinuierlicher Regler (FM 355-2 C) und Impulsregler (FM 355-2 S) haben die gleiche Struktur des Regelalgorithmus. Beim Schrittregler (FM 355-2 S) können die Schaltflächen Kühlen und Regelzone nicht angewählt werden.

Das folgende Bild zeigt das Blockschaltbild des Regelalgorithmus für kontinuierliche Regler und Impulsregler:

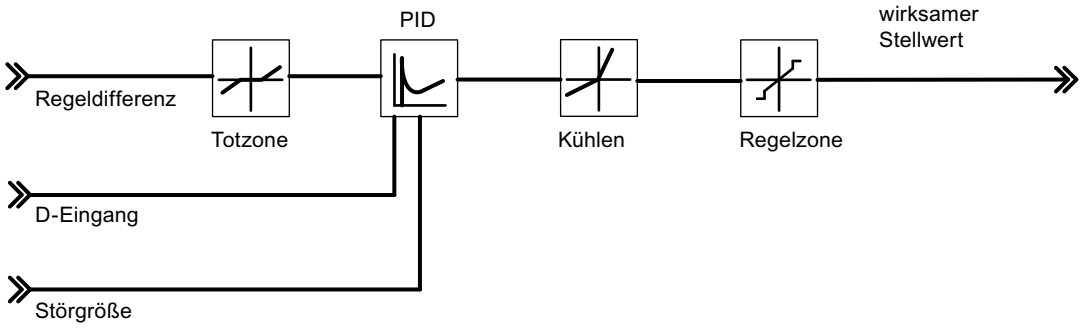

Regelalgorithmus: PID in Parallelstruktur

Im Zyklus der projektierten Abtastzeit wird die Stellgröße des Reglers aus der Regeldifferenz im PID-Stellungsalgorithmus errechnet. Der Regler ist in reiner Parallelstruktur ausgeführt. Der Proportional-, Integral- oder Differentialanteil kann jeweils einzeln abgeschaltet werden. Beim Integral- und Differentialanteil geschieht dies, indem die jeweiligen Parameter TI bzw. TD zu Null gesetzt werden.

Das folgende Bild zeigt den Regelalgorithmus der FM 355-2 (Parallelstruktur):

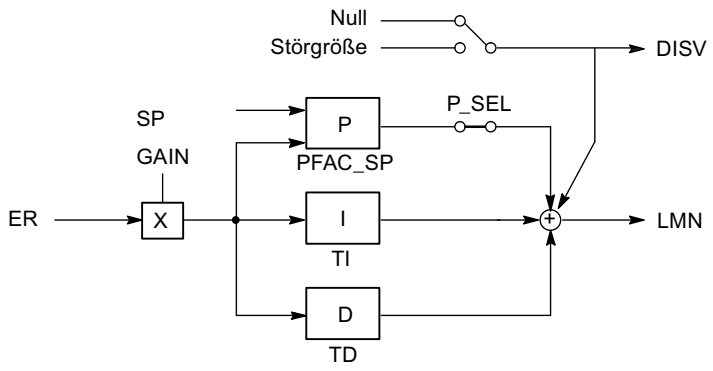

Regelalgorithmus FM 355 und FM 455

Komponenten des Regelalgorithmus

Beim Regelalgorithmus können Sie zwischen folgenden Betriebsarten wählen:

- Temperaturregler (selbsteinstellender Fuzzy-Regler)
- PID-Regler

K-Regler und S-Regler haben die gleiche Struktur des Regelalgorithmus. Das folgende Bild zeigt das Blockschaltbild des Regelalgorithmus:

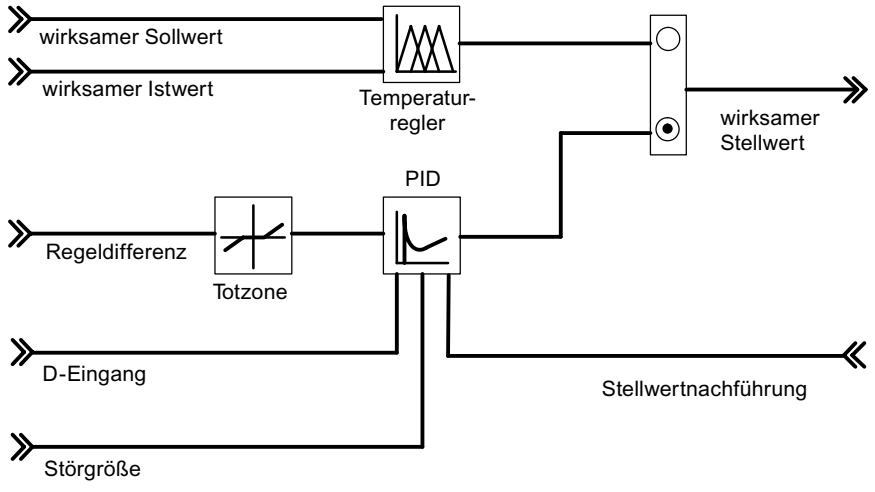

Regelalgorithmus und Reglerstruktur

Im Zyklus der projektierten Abtastzeit wird die Stellgröße des kontinuierlich arbeitenden Reglers aus der Regeldifferenz im PID-Stellungsalgorithmus errechnet. Der Regler ist in reiner Parallelstruktur ausgeführt (siehe Bild unten). Der Proportional-, Integral- oder Differentialanteil kann jeweils einzeln abgeschaltet werden. Beim Integral- und Differentialanteil geschieht dies, indem die jeweiligen Parameter TI bzw. TD auf Null gesetzt werden.

Das folgende Bild zeigt den Regelalgorithmus der FM 355 (Parallelstruktur):

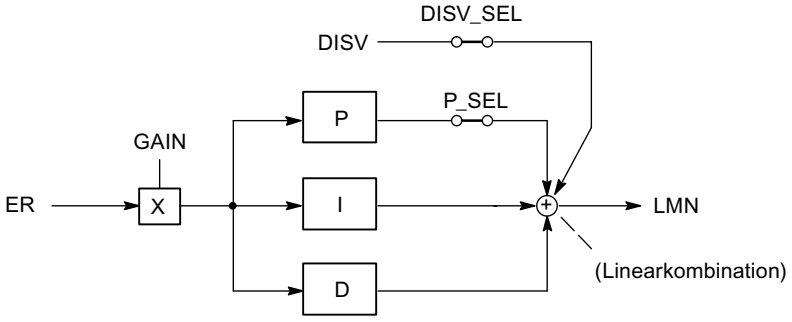

Betriebsart wählen

Auswählen des Betriebsmodus

Wurde als Reglertyp der Eintrag "PID-Regler" gewählt, können Sie aus den verschiedenen Betriebsmodi wählen:

- P, zum Realisieren eines P-Reglers
- PI, zum Realisieren eines PI-Reglers
- PD, zum Realisieren eines PD-Reglers
- PID, zum Realisieren eines PID-Reglers
- I, zum Realisieren eines I-Reglers
- ID, zum Realisieren eines ID-Reglers
- D, zum Realisieren eines D-Reglers

Stoßfreie Umschaltung Hand-Automatik

Haben Sie "Stoßfreie Umschaltung Hand-Automatik" gewählt (nicht bei Schrittregler), wird der Integrator im Handbetrieb so nachgeführt, dass die Stellgröße bei der Hand-Automatik-Umschaltung keinen Sprung durch P- und D-Anteil macht. Eine anstehende Regeldifferenz wird nur langsam über den I-Anteil ausgeregelt. Wenn keine stoßfreie Hand-Automatik-Umschaltung ausgewählt wird, dann macht die Stellgröße ausgehend vom aktuellen Handwert bei der Hand-Automatik-Umschaltung einen Sprung, der der aktuellen Regeldifferenz entspricht. Eine anstehende Regeldifferenz wird so schnell ausgeregelt.

Hinweis

Beim Schrittregler erfolgt das Umschalten in den Automatikbetrieb immer stoßbehaftet. Die anstehende Regeldifferenz führt über GAIN zu einer sprungförmigen Änderung der internen Stellgröße. Durch das integral wirkende Stellglied wird jedoch nur eine rampenförmige Ansteuerung des Prozesses bewirkt.

Störgrößenaufschaltung

Dem Ausgangssignal des Reglers kann zusätzlich eine Störgröße DISV aufgeschaltet werden. Die Zu- bzw. Abschaltung erfolgt im Fenster "Regeldifferenz" des Konfigurationswerkzeugs über den Schalter "Signalauswahl Störgröße Regler".

Totzone

Zweck der Totzone

Dem PID-Regler vorgeschaltet ist eine Totzone. Die Totzone unterdrückt im eingeschwungenen Reglerzustand den Rauschanteil im Signal der Regeldifferenz, der durch Überlagerung der Regel- oder Führungsgröße mit einem höherfrequenten Störsignal entstehen kann. Dadurch verhindert sie unerwünschtes Oszillieren des Reglerausgangs.

Totzonenbreite

Die Totzonenbreite ist einstellbar. Befindet sich die Regeldifferenz innerhalb der eingestellten Totzonenbreite, wird am Ausgang der Totzone der Wert 0 (Regeldifferenz = 0) ausgegeben. Erst wenn die Eingangsgröße den Empfindlichkeitsbereich verlässt, ändert sich der Ausgang um die gleichen Werte wie die Eingangsgröße.

Daraus resultiert eine Verfälschung des übertragenen Signals auch außerhalb der Totzone. Dies wird jedoch zur Vermeidung von Sprüngen an den Grenzen der Totzone in Kauf genommen. Die Verfälschung entspricht dem Wert der Totzonenbreite und ist deshalb leicht kontrollierbar.

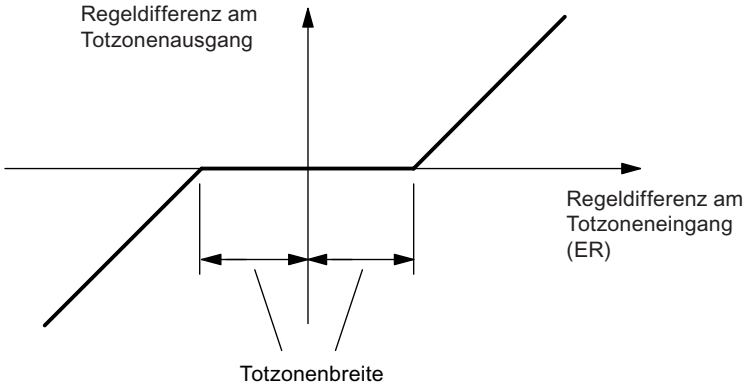

Bild 3-8 Totzone

P-Regelung

P-Regler

Das folgende Bild zeigt die Sprungantwort eines P-Reglers:

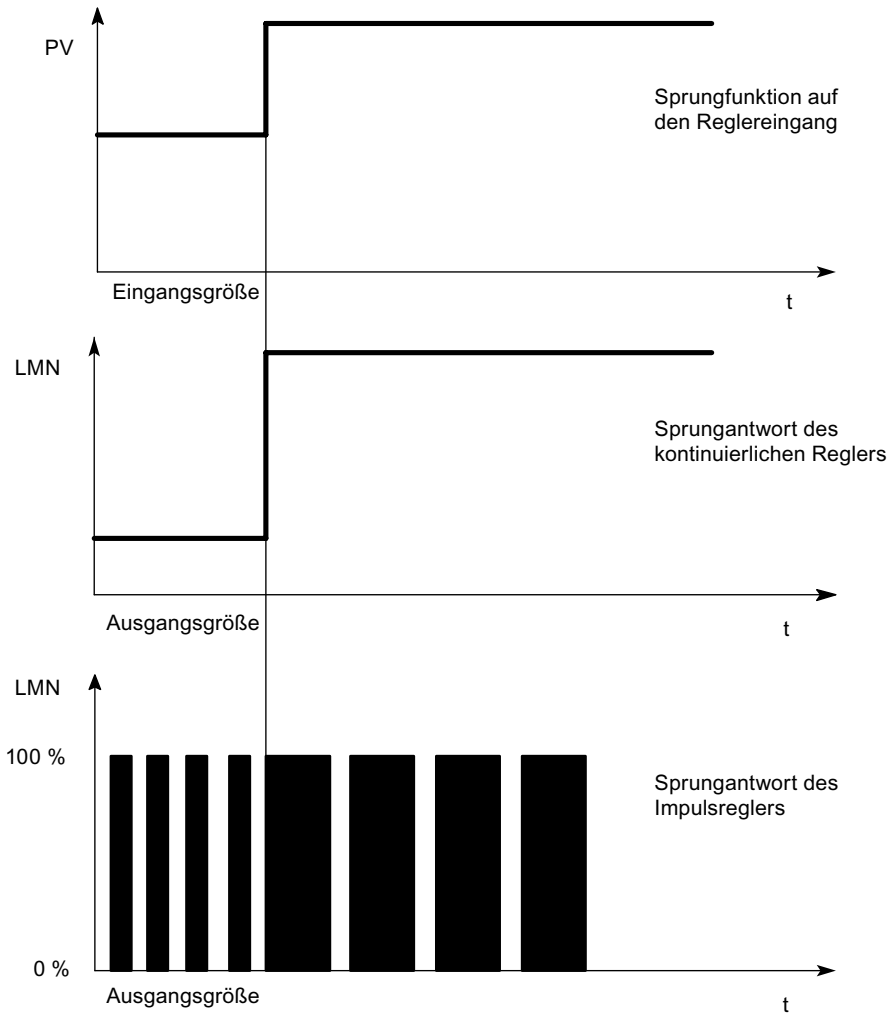

Formel für P-Regler

Ausgangsgröße und Eingangsgröße sind direkt proportional, d. h.:

Ausgangsgrößenänderung = Proportionalbeiwert × Eingangsgrößenänderung

LMN = GAIN x ER

P-Regelung

Beim P-Regler sind der I- und der D-Anteil abgeschaltet. Das bedeutet, dass bei Regeldifferenz ER = 0 auch die Stellgröße = 0 ist. Soll ein Arbeitspunkt ≠0, also ein Zahlenwert für die Stellgröße bei Regeldifferenz 0 eingestellt werden, ist dies über den Arbeitspunkt möglich:

- Arbeitspunkt automatisch: Der Arbeitspunkt wird bei der Hand-Automatik-Umschaltung vom Regler auf die aktuelle (Hand-)Stellgröße gesetzt.
- Arbeitspunkt nicht automatisch: Sie können den Arbeitspunkt parametrieren.
- Beispiel: Arbeitspunkt AP = 5% ergibt eine Stellgröße von 5% bei Regeldifferenz ER = 0.

Das folgende Bild zeigt einen P-Regler mit Arbeitspunkteinstellung über I-Glied:

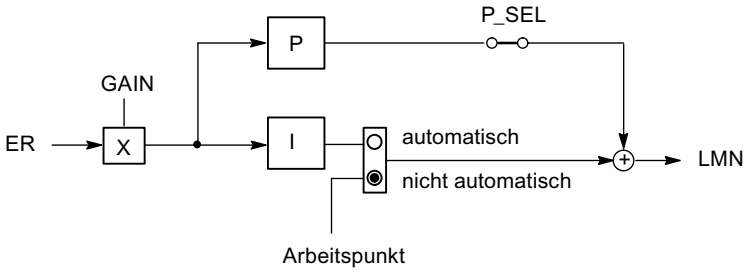

Das folgende Bild zeigt die Sprungantwort des P-Reglers:

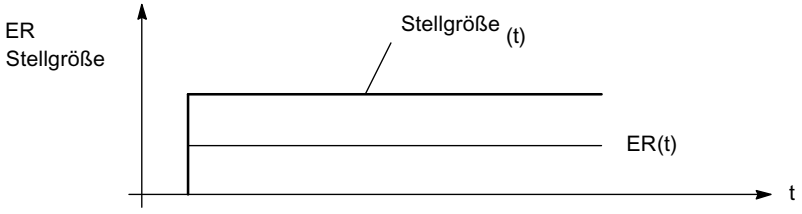

PI-Regelung

PI-Regler

Das folgende Bild zeigt die Sprungantwort eines PI-Reglers:

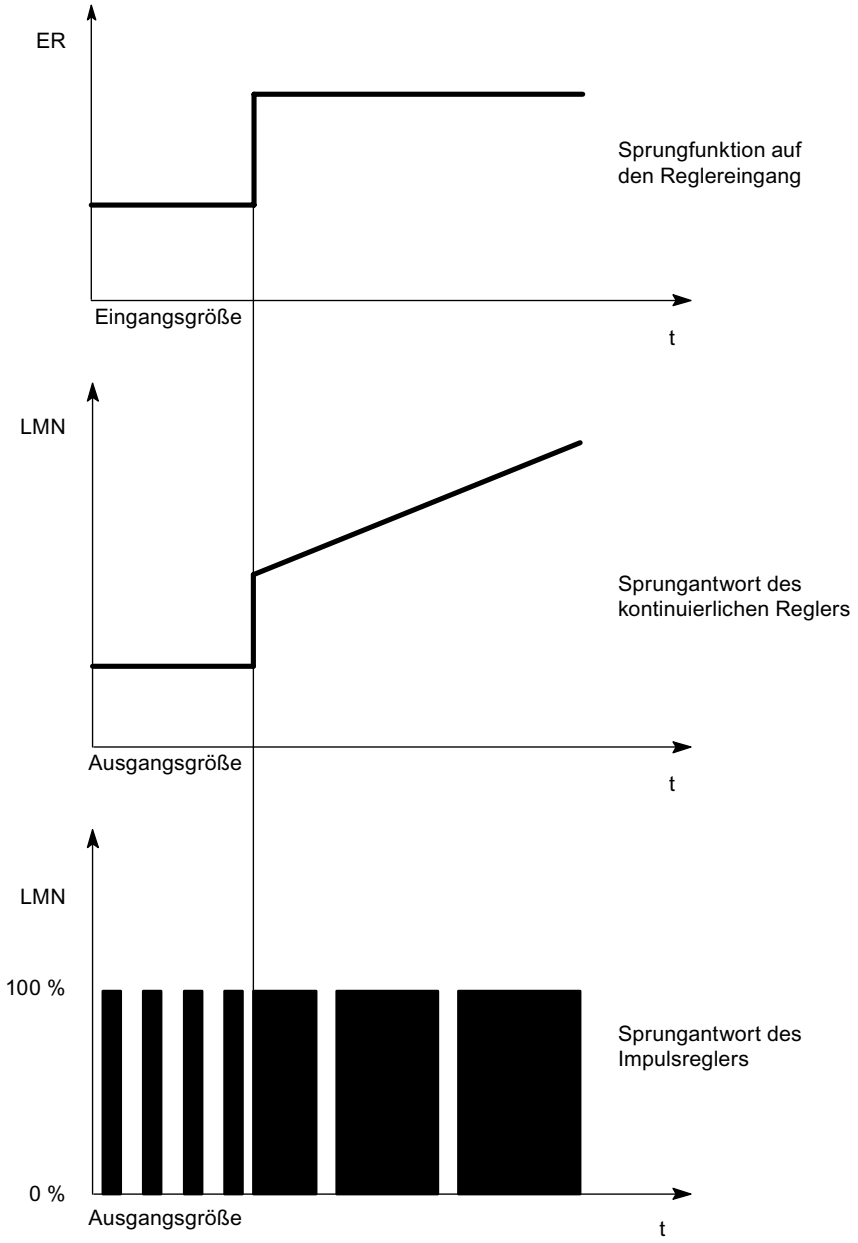

I-Regelglieder haben als Ausgangsgröße das Integral der Eingangsgröße, d. h. der Regler summiert die Abweichung vom Sollwert über die Zeit. Das bedeutet, dass der Regler so lange nachstellt, bis die Abweichung vom Sollwert aufgehoben ist. In der Praxis ist, je nach Anforderung an das Regelverhalten, eine Kombination aus den verschiedenen Zeitgliedern ideal. Das Zeitverhalten der einzelnen Glieder lässt sich durch die Reglerparameter

Proportionalbereich GAIN, Nachstellzeit TI (I-Anteil) und Vorhaltezeit TD (D-Anteil) beschreiben.

Formel für PI-Regler

Für die Sprungantwort des PI-Reglers im Zeitbereich gilt:

$$LMN = GAIN \times ER \times \left(1 + \frac{t}{TI}\right)$$

t = Zeitdauer seit dem Sprung der Eingangsgröße

PI-Regelung

Beim PI-Regler ist der D-Anteil abgeschaltet (TD=0.0). Ein PI-Regler verstellt über den I-Anteil die Ausgangsgröße so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet. Werden die Stellwertbegrenzungen überschritten, behält der I-Anteil den an der Grenze erreichten Wert bei (Anti-Reset-Windup).

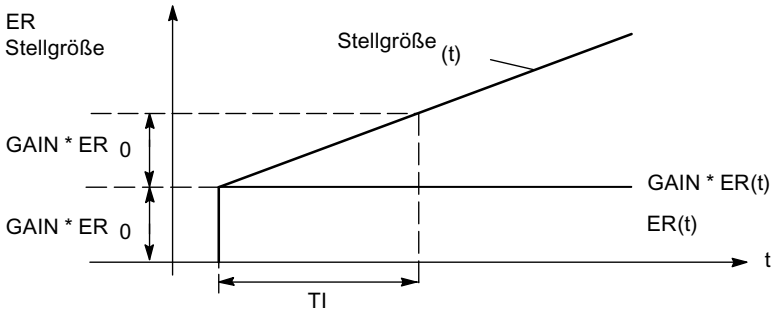

PD-Regelung

PD-Regler

Das folgende Bild zeigt die Sprungantwort eines PD-Reglers:

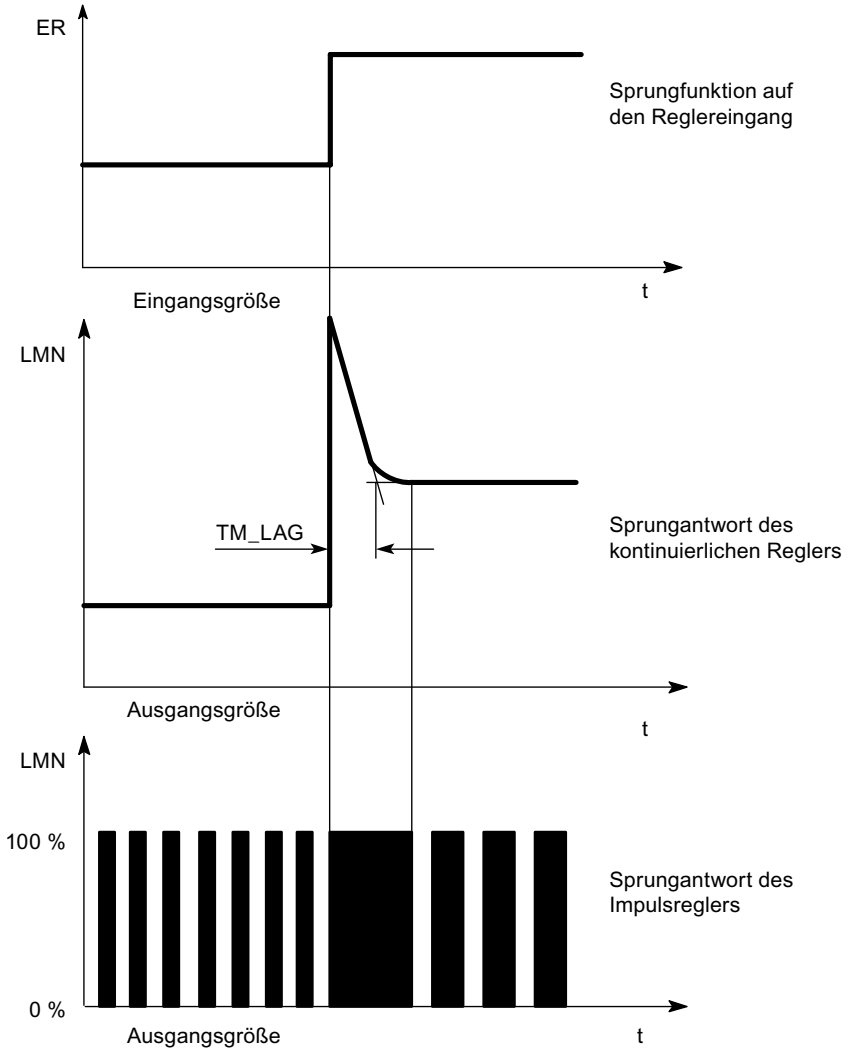

D-Regelglieder sind allein zur Regelung ungeeignet, denn sie geben keinen Stellbefehl mehr ab, wenn die Eingangsgröße sich wieder auf einen statischen Wert eingestellt hat.

In Verbindung mit P-Regelgliedern nutzt man den D-Anteil, um in Abhängigkeit von der Änderungsgeschwindigkeit der Regelgröße einen entsprechenden Stellimpuls zu erzeugen. Wirkt auf die Regelstrecke eine Störgröße z, stellt sich der PD-Regler aufgrund des geänderten Stellgrades auf eine andere Regelabweichung ein. Störungen werden nicht vollkommen ausgeregelt. Vorteilhaft ist das gute dynamische Verhalten. Beim Anfahren und bei der Führungsgrößenänderung wird ein gut gedämpfter, schwingungsfreier Übergang erreicht. Ein Regler mit D-Anteil ist jedoch nicht angebracht, wenn eine Regelstrecke pulsierende Messgrößen hat, z. B. bei Druck- oder Durchflussregelungen.

Formel für PD-Regler

Für die Sprungantwort des PD-Reglers im Zeitbereich gilt:

$$LMN = GAIN \times ER \times \left(1 + \frac{TD}{TM_LAG} \times e^{\frac{-t}{TM_LAG}}\right)$$

t = Zeitdauer seit dem Sprung der Eingangsgröße

PD-Regelung

Beim PD-Regler ist der I-Anteil abgeschaltet (TI=0.0). Das bedeutet, dass bei Regeldifferenz ER = 0 auch das Ausgangssignal = 0 ist. Soll ein Arbeitspunkt ≠0, also ein Zahlenwert für die Stellgröße bei Regeldifferenz 0 eingestellt werden, ist dies über den Arbeitspunkt möglich:

- Arbeitspunkt automatisch: Der Arbeitspunkt wird bei der Hand-Automatik-Umschaltung vom Regler auf die aktuelle (Hand-)Stellgröße gesetzt.
- Arbeitspunkt nicht automatisch: Sie können den Arbeitspunkt parametrieren.

Der PD-Regler bildet die Eingangsgröße ER(t) proportional auf das Ausgangssignal ab und addiert den durch Differentiation von ER(t) gebildeten D-Anteil dazu, der nach der Trapezregel (Padé-Näherung) doppeltgenau berechnet wird. Das Zeitverhalten (Stärke des D-Anteils bzw. der Stellfläche) wird durch die Differenzierzeit TD (Vorhaltzeit) bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen wird der D-Anteil über eine Verzögerung erster Ordnung (einstellbare Zeitkonstante: TM_LAG) realisiert. Meist genügt ein kleiner Wert für TM_LAG, um den gewünschten Erfolg zu erzielen.

Je größer Sie den Differenzierfaktor D_F wählen,

- umso kleiner wird die wirksame Zeitkonstante TD/D_F der Verzögerung und
- umso größer ist die maximale Stellgröße zu Beginn und
- umso besser das Regelverhalten und
- umso größer aber auch die Rauschempfindlichkeit.

D_F wird auf den Wertebereich zwischen 5.0 und 10.0 begrenzt.

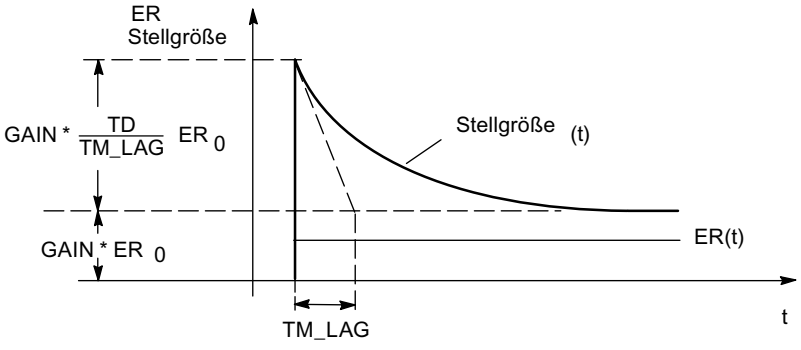

PID-Regelung

PID-Regler

Das folgende Bild zeigt die Sprungantwort eines PID-Reglers:

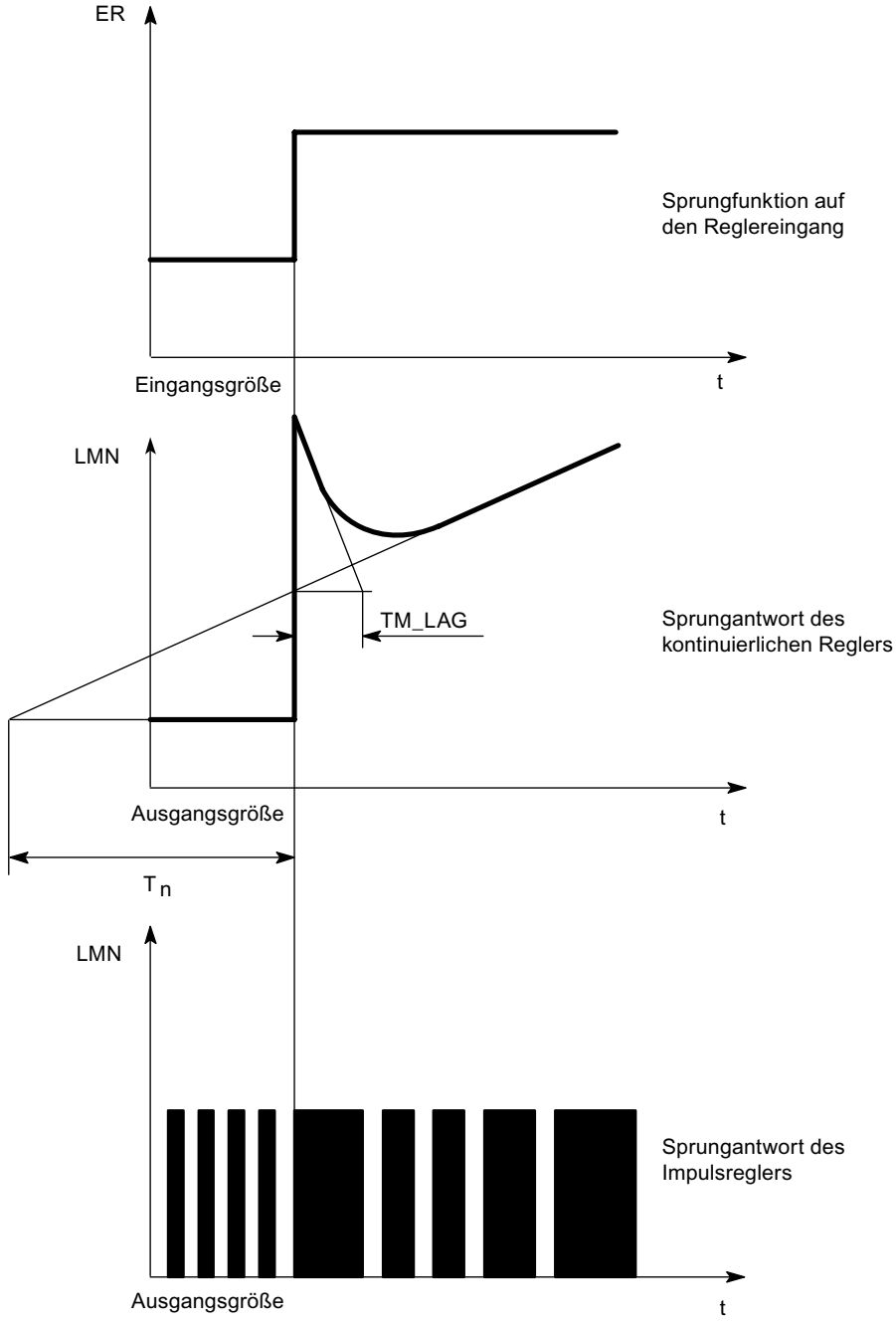

Die meisten in der Verfahrenstechnik vorkommenden Regelungen lassen sich mit einem Regler mit PI-Verhalten beherrschen. Bei langsamen Regelstrecken mit großer Verzugszeit,

z. B. Temperaturregelungen, lässt sich das Regelergebnis durch einen Regler mit PID-Verhalten verbessern.

Das folgende Bild zeigt die Sprungantwort bei unterschiedlichem Regelverhalten:

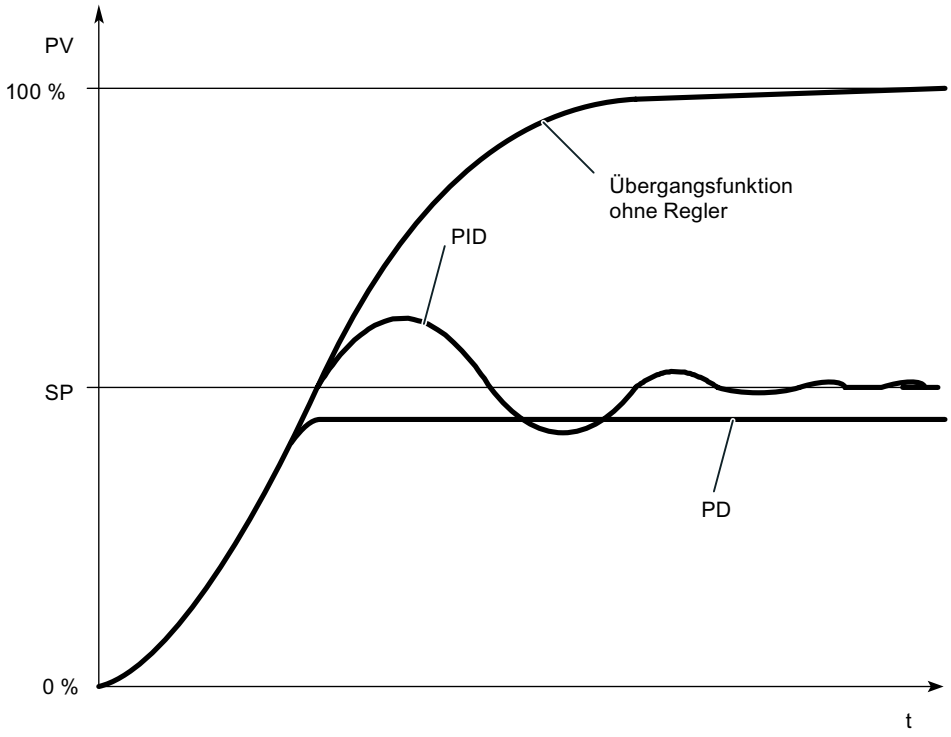

Regler mit PI- und PID-Verhalten haben den Vorteil, dass nach dem Einschwingen die Regelgröße keine Abweichung zum Sollwert zeigt. Die Regelgröße schwingt beim Anfahren über den Sollwert.

Formel für PID-Regler

Für die Sprungantwort des PI-Reglers im Zeitbereich gilt:

$$LMN = GAIN \times ER \times \left(1 + \frac{t}{TI} + \frac{TD}{TM_LAG} \times e^{\frac{-t}{TM_LAG}}\right)$$

t = Zeitdauer seit dem Sprung der Eingangsgröße

PID-Regelung

Beim PID-Regler sind der P-, I- und D-Anteil eingeschaltet. Ein PID-Regler verstellt die Ausgangsgröße über den I-Anteil so lange, bis die Regeldifferenz ER = 0 geworden ist. Das gilt jedoch nur, wenn die Ausgangsgröße dabei die Grenzen des Stellbereiches nicht überschreitet. Werden Stellwertbegrenzungen überschritten, behält der I-Anteil den an der Grenze erreichten Wert bei (Anti-Reset-Windup).

Der PID-Regler bildet die Eingangsgröße ER(t) proportional auf das Ausgangssignal ab und addiert die durch Differentiation und Integration von ER(t) gebildeten Anteile dazu. Das Zeitverhalten wird durch die Differenzierzeit TD (Vorhaltzeit) und die Integrationszeit TI (Nachstellzeit) bestimmt.

Zur Signalglättung und Unterdrückung von Störsignalen wird der D-Anteil über eine Verzögerung erster Ordnung realisiert.

Je größer Sie den Differenzierfaktor D_F wählen,

- umso kleiner wird die wirksame Zeitkonstante TD/D_F der Verzögerung und
- umso größer ist die maximale Stellgröße zu Beginn und
- umso besser das Regelverhalten und
- umso größer aber auch die Rauschempfindlichkeit.

D_F wird auf den Wertebereich zwischen 5.0 und 10.0 begrenzt.

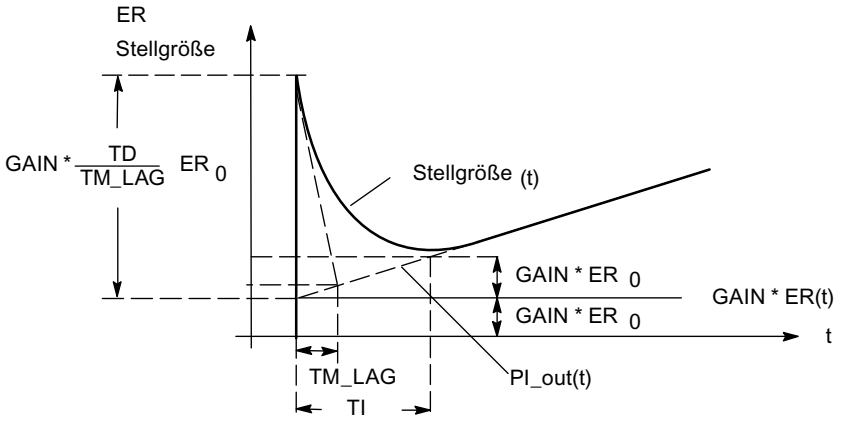

I-Regelung

I-Regler

Um eine reine I-Regelung zu realisieren, können Sie den P-Anteil abschalten. Dies ist auch möglich über den Parameter P_SEL des Funktionsbausteins PID_FM.

P-Anteil

Abschwächung des P-Anteils bei Sollwertveränderungen

Um Überschwingen beim Istwert oder zu starke Ausschläge bei der Stellgröße zu vermeiden, können Sie den P-Anteil über den Parameter Proportionalfaktor bei Sollwertänderungen (PFAC_SP) abschwächen. Über PFAC_SP können Sie zwischen 0.0 und 1.0 kontinuierlich wählen, wie stark der P-Anteil bei Sollwertänderungen wirken soll:

- PFAC_SP=1.0: P-Anteil bei Sollwertänderung voll wirksam
- PFAC_SP=0.0: P-Anteil bei Sollwertänderung nicht wirksam

In der FM355-2 gibt es zwei Möglichkeiten die Geschwindigkeit einer Sollwertänderung zu begrenzen:

- Rampe einschalten (> 0.0s)
- Faktor für Sollwertänderung < 1.0

Verwenden sie nur eine der beiden Begrenzungen. Wenn beide Begrenzungen gleichzeitig aktiviert werden, dann tritt bei einem Sollwertsprung eine Stellgrößenänderung invers zur Sollwertänderung auf (Sprungantwort).

Besonderheiten beim Schrittregler

Ein Wert von PFAC_SP < 1.0 kann das Überschwingen reduzieren, falls die Motorstellzeit MTR_TM klein gegenüber der Ausgleichszeit TA ist und das Verhältnis TU/TA < 0.2 ist. Erreicht MTR_TM 20 % von TA, ist nur noch eine geringe Verbesserung zu erzielen.

Umkehrung der Reglerwirkung

Die Umkehrung, d. h. die Umstellung des Reglers von der Zuordnung

- steigende Regeldifferenz = steigende Stellgröße
  auf
- steigende Regeldifferenz = fallende Stellgröße

erreichen Sie durch Einstellung eines negativen Proportionalbeiwertes (GAIN). Das Vorzeichen dieses Parameterwertes bestimmt die Wirkungsrichtung des Reglers.

D-Anteil

D-Anteil in der Rückführung

Um bei Sollwertänderungen impulsförmige Ausschläge im D-Anteil der Stellgröße zu verhindern, können Sie den D-Anteil in die Rückführung legen.

In dieser Struktur wird auf den D-Anteil nur die negative Regelgröße (Faktor = -1) aufgeschaltet. Beim D-Anteil erfolgt die Umschaltung in die Rückführung im Fenster "Regeldifferenz" über den Schalter "D-Eingang Regler", indem als Eingangssignal der negierte wirksame Istwert ausgewählt wird.

Hinweis

Falls Sie den D-Anteil in die Rückführung legen, sollten Sie auch PFAC_SP reduzieren, sonst erhöht sich das Überschwingen im Istwert.

Das folgende Bild zeigt den Regelalgorithmus mit D-Anteil im Rückführzweig:

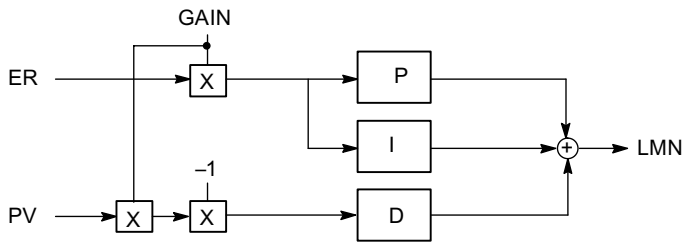

P-/D-Anteil

P-/D-Anteil in der Rückführung

In der Parallelstruktur erhält jeder Anteil des Regelalgorithmus die Regeldifferenz als Eingangssignal. In dieser Struktur wirken Sollwertsprünge direkt auf den Regler. Die Stellgröße wird über den P- und den D-Anteil durch Sollwertsprünge unmittelbar beeinflusst.

Eine andere Struktur des Reglers jedoch, bei der die Bildung des P- und des D-Anteils in die Rückführung verlegt wird, garantiert bei Sprungänderungen der Führungsgröße einen stoßfreien Verlauf der Stellgröße (siehe folgendes Bild).

In dieser Struktur verarbeitet der I-Anteil die Regeldifferenz als Eingangssignal, auf den P- und den D-Anteil wird nur die **negative** Regelgröße (Faktor = - 1) aufgeschaltet. Beim D-Anteil erfolgt die Umschaltung in die Rückführung im Fenster "Regeldifferenz" über den Schalter "D-Eingang Regler", indem als Eingangssignal der negierte wirksame Istwert ausgewählt wird.

Das folgende Bild zeigt den Regelalgorithmus mit P- und D-Anteil im Rückführzweig:

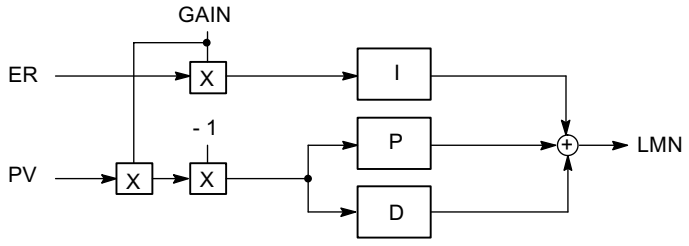

Fuzzy Temperatur-Reglerbaugruppe

Temperaturregler

Der Temperaturregler ist ein selbsteinstellender Fuzzy-Regler, der nach einer Identifikation der Regelstrecke mit selbst ermittelten Regelparametern arbeitet.

Beim Temperaturregler sind folgende Einstellungen möglich:

- Kühlregler
- Heizregler
- Aggressivität
  Mit dem Parameter Aggressivität können Sie die Geschwindigkeit des Einschwingverhaltens beeinflussen.

Mögliche Werte für die Aggressivität	
-1 ≤ Aggressivität < 0	langsameres Einschwingverhalten als über Identifikation ermittelt
Aggressivität = 0	Einschwingverhalten wie über Identifikation ermittelt
0 < Aggressivität ≤ 1	schnelleres Einschwingverhalten als über Identifikation ermittelt

3.5.4.4 Reglerausgang

Unterschiede der Regler

Im Wesentlichen lassen sich die Regler FM 355, FM 355-2 und FM 455 in zwei Typen unterteilen: in Schrittregler und kontinuierliche Regler. Schrittregler sind mit einem ergänzenden S gekennzeichnet, also FM 355 S, FM 355-2 S und FM 455 S, während kontinuierliche Regler mit einem C gekennzeichnet sind, also FM 355 C, FM 355-2 C und FM 455 C. Die sonstige Arbeitsweise der Reglerbaugruppen unterscheidet sich nicht, eine FM 355 S und eine FM 355 C besitzen identische Algorithmen. Die Unterschiede liegen im Reglerausgang.

Schrittregler

Der kontinuierliche Schrittregler besitzt digitale Ausgänge. Somit kann eine Steuerklappe nur über die Signale "offen" oder "geschlossen" angesprochen werden. Soll eine Steuerklappe z. B. auf 50% der Kapazität geregelt werden, kann dies nur durch gleich langes Einstellen von "offen" und "geschlossen" realisiert werden.

Nähere Hinweise zu den Schrittreglern hierzu finden Sie in den folgenden Stellen:

- Digitalausgänge des Reglers (Seite 1005)
- Schrittregler ohne Stellungsrückmeldung (Seite 1007)
- Schrittregler mit Stellungsrückmeldung (Seite 1007)
- Impulsregler (Seite 1007)

Kontinuierlicher Regler

Der kontinuierliche Regler besitzt analoge Ausgänge. Eine Steuerklappe kann also nicht nur "offen" oder "geschlossen" sein, sondern auch zu bestimmten Anteilen geöffnet werden. Soll also eine Steuerklappe z. B. auf 50% der Kapazität geregelt werden, gibt die Reglerbaugruppe den Wert "50%" aus und die Klappe wird zur Hälfte geöffnet.

Nähere Hinweise zu den kontinuierlichen Reglern hierzu finden Sie in den folgenden Stellen:

- Analogausgänge des Reglers (Seite 1005)
- Kontinuierlicher Regler (Seite 1006)

Analogausgänge des Reglers

Analogausgänge

Für jeden Analogausgang können Sie die folgenden Funktionen parametrieren:

- Quelle für das Signal
- Signaltyp

Auswahl der Quelle für das Signal an den Analogausgängen

Mit der Auswahl der Quelle für das Signal können Sie festlegen, welcher Signalwert am jeweiligen Analogausgang ausgegeben wird.

Folgende Quellen für das Signal sind parametrierbar:

- der Wert Null
- der aufbereitete Analogwert eines der Analogeingänge
- der Stellwert A eines der Reglerkanäle
- der Stellwert B eines der Reglerkanäle

Die FM 355 und die FM 355-2 haben jeweils vier Analogeingänge und vier Reglerkanäle.

Die FM 455 hat 16 Analogeingänge und 16 Reglerkanäle.

Signaltyp an den Analogausgängen

Für jeden Analogausgang können Sie den Signaltyp bestimmen.

Folgende Signaltypen sind parametrierbar:

- Stromausgang 0 ... 20 mA
- Stromausgang 4 ... 20 mA
- Spannungsausgang 0 ... 10 V
- Spannungsausgang -10 ... 10 V

Digitalausgänge des Reglers

Digitalausgänge des Reglers

Die Digitalausgänge des Reglers dienen zur Ansteuerung von integrierenden oder nichtintegrierenden Stellgliedern.

Für jeden Digitalausgang können Sie die folgenden Funktionen parametrieren:

- Ausgangstyp
- Signalauswahl
- Signalverarbeitung

Ausgangstyp des Signals am Digitalausgang

Für jeden Digitalausgang können Sie den Signaltyp bestimmen.

Folgende Signaltypen stehen zur Verfügung:

- Schrittregler ohne Stellungsrückmeldungseingang
- Schrittregler mit Stellungsrückmeldungseingang
- Impulsregler

Signalauswahl

In Abhängigkeit des Signaltyps am Ausgang können für die Signalauswahl verschiedene Werte parametriert werden:

- Externe Stellgröße (Seite 1008)
- Nachführung (Seite 1008)
- Stellungsrückmeldungseingang (Seite 1009)
- Sicherheitsstellwert (Seite 1010)

Signalverarbeitung

In Abhängigkeit des Signaltyps am Ausgang können für die Signalverarbeitung verschiedene Werte parametriert werden:

- Grenzwert Ausgangssignal (Seite 1011)
- Split Range (Seite 1011)
- Impulsformer (Seite 1014)

Kontinuierlicher Regler

Signaltyp an den Digitalausgängen

Die nachfolgenden Parameter können nur an einer analogen Baugruppe eingestellt werden.

Einstellbare Parameter

- Externe Stellgröße (Seite 1008) schalten
- Nachführung (Seite 1008) schalten
- Sicherheitsstellwert (Seite 1010) einstellen
- Stellwert begrenzen (Seite 1011)
- Zwei Stellglieder ansteuern (Seite 1011)
- Ausgangssignal und Signaltyp wählen (Seite 1014)

Schrittregler ohne Stellungsrückmeldung

Signaltyp an den Digitalausgängen

Die nachfolgenden Parameter können nur an einer digitalen Baugruppe eingestellt werden.

Einstellbare Parameter

- Externe Stellgröße (Seite 1008) schalten
- Sicherheitsstellwert (Seite 1010) einstellen
- Impulsformer (Seite 1014)

Schrittregler mit Stellungsrückmeldung

Signaltyp an den Digitalausgängen

Die nachfolgenden Parameter können nur an einer digitalen Baugruppe eingestellt werden.

Einstellbare Parameter

- Externe Stellgröße (Seite 1008) schalten
- Nachführung (Seite 1008) schalten
- Sicherheitsstellwert (Seite 1010) einstellen
- Stellwert begrenzen (Seite 1011)
- Stellungsrückmeldung (Seite 1009) einstellen
- Impulsformer (Seite 1014)

Impulsregler

Signaltyp an den Digitalausgängen

Die nachfolgenden Parameter können nur an einer digitalen Baugruppe eingestellt werden.

Einstellbare Parameter

- Externe Stellgröße (Seite 1008) schalten
- Nachführung (Seite 1008) schalten
- Sicherheitsstellwert (Seite 1010) einstellen
- Stellwert begrenzen (Seite 1011)
- Analogsignal vorbereiten auf die Umwandlung in ein Digitalsignal (Seite 1011)
- Analogsignal in Digitalsignal umwandeln (Seite 979)

Funktionen des Reglerausgangs

Externe Stellgröße

Übersicht

Beim Reglerausgang sind verschiedene Verschaltungsmöglichkeiten für den Stellwert, den Nachführeingang und den Sicherheitsstellwert realisiert (Stellwertumschaltung).

Um zu verhindern, dass der Stellwert unzulässige Werte für den Prozess annehmen kann, ist eine Begrenzung vorgesehen.

Schalten externer Stellwert

Die Umschaltung zwischen externem Stellwert und wirksamem Stellwert aus dem Regler erfolgt alternativ durch

- einen Binärwert aus dem Funktionsbaustein
- ein Signal, das sich durch Veroderung eines Binärwerts aus dem Funktionsbaustein und einem Digitaleingang ergibt

Nachführung

Übersicht

Die Stellwertnachführung verhindert bei der Umschaltung von Hand auf Automatik einen Stellwertsprung.

Bei der Hand-Automatik-Umschaltung bleibt der Stellwert unverändert. Die Stellwertnachführung ist nicht aktiv, wenn ein reiner P-Regler mit festem Arbeitspunkt realisiert ist ("automatisch" ist in der Maske PID-Regler nicht angekreuzt).

Nachführeingang

Es gibt folgende alternative Einstellungen:

- Der Nachführeingang hat den Wert Null
- Der Nachführeingang ist der aufbereitete Analogwert eines Analogeingangs

Schalten Nachführen

Die Umschaltung zwischen Stellwert und Nachführeingang erfolgt alternativ durch

- einen Binärwert aus dem Funktionsbaustein
- ein Signal, das sich durch Veroderung eines Binärwerts aus dem Funktionsbaustein und einem Digitaleingang ergibt

Stellungsrückmeldung

Übersicht

Ein Stellungsrückmeldungseingang ist nur bei S-Reglern verfügbar.

Stellungsrückmeldungseingang

Es gibt folgende alternative Einstellungen:

- Der Stellungsrückmeldungseingang hat den Wert Null
- Der Stellungsrückmeldungseingang ist der aufbereitete Analogwert eines Analogeingangs

Reglerausgang des Schrittreglers

Das folgende Bild zeigt das Schema für den Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung):

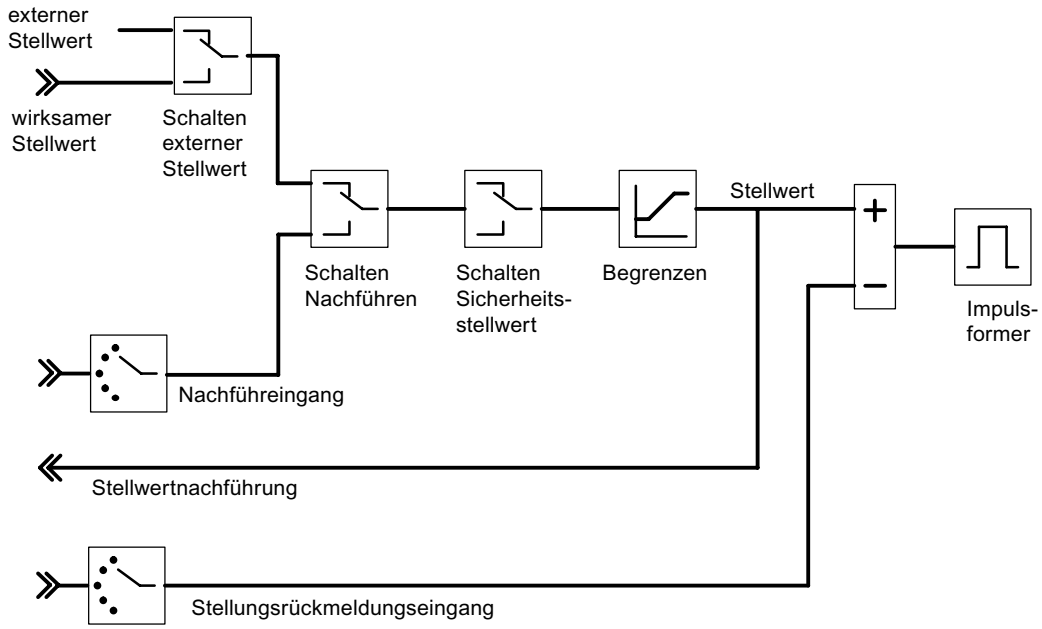

Das folgende Bild zeigt das Schema für den Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung):

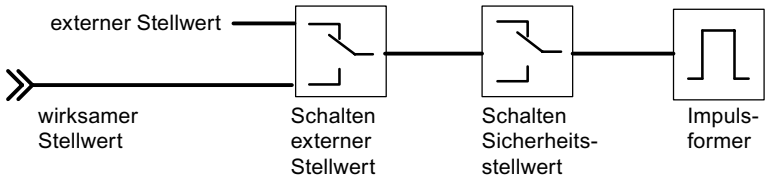

Beim Schrittregler ohne analoge Stellungsrückmeldung wirken der externe Stellwert und der Sicherheitsstellwert wie folgt:

Wenn ein Wert zwischen 40.0% und 60.0% vorgegeben wird, dann wird kein Binärausgang gesetzt, die Stelleinrichtung bleibt unverändert.

Wenn ein Wert > 60.0% vorgegeben wird, dann wird "Stellsignal hoch" so lange ausgegeben, bis die Rückmeldung "Stelleinrichtung am oberen Anschlag" anspricht.

Wenn ein Wert < 40.0% vorgegeben wird, dann wird "Stellsignal tief" so lange ausgegeben, bis die Rückmeldung "Stelleinrichtung am unteren Anschlag" anspricht.

Sicherheitsstellwert

Übersicht

Beim Reglerausgang sind verschiedene Verschaltungsmöglichkeiten für den Stellwert, den Nachführeingang und den Sicherheitsstellwert realisiert (Stellwertumschaltung).

Schalten Sicherheitsstellwert

- Festlegung des Sicherheitsstellwerts
- Alternative Reaktion des Funktionsmoduls bei Anlauf:
  - Das Funktionsmodul geht in Regelbetrieb
  - Als Stellwert wird der Sicherheitsstellwert ausgegeben
- Die Umschaltung auf Sicherheitsstellwert erfolgt alternativ durch
  - einen Binärwert aus dem Funktionsbaustein
  - ein Signal, das sich durch Veroderung eines Binärwerts aus dem Funktionsbaustein und einem Digitaleingang ergibt
- Reaktion bei Messumformerstörung des Istwerts A:
  - Bei Einstellung "Regelbetrieb" bleibt die Betriebsart des Reglers unverändert
  - Bei Einstellung "Stellwert = Sicherheitsstellwert" wird auf den Sicherheitsstellwert umgeschaltet
- Reaktion bei Messumformerstörung eines Analogeingangs:
  - Bei Einstellung "Regelbetrieb" bleibt die Betriebsart des Reglers unverändert
  - Bei Einstellung "Stellwert = Sicherheitsstellwert" wird auf den Sicherheitsstellwert umgeschaltet

Stellwertbegrenzung

Übersicht

Um zu verhindern, dass der Stellwert unzulässige Werte für den Prozess annehmen kann, ist eine Begrenzung vorgesehen.

Begrenzung des Stellwerts

- Obere und untere Begrenzung (nicht abschaltbar)

Split Range

Übersicht

Die Splitrangefunktion erzeugt aus dem Stellwert als Eingangssignal zwei unterschiedlich normierbare Ausgangssignale Stellwert A und Stellwert B. Damit lassen sich z. B. mit einem Stellwert zwei Ventile ansteuern.

Bildung der Splitrangestellwerte

- Ein/Aus (nur K-Regler)
- Anfangs- und Endwert Eingangssignal
- Anfangs- und Endwert Ausgangssignal

Splitrange

Mit Hilfe der Splitrangefunktion kann man mit einer Stellgröße zwei Stellventile ansteuern. Die Splitrangefunktion erzeugt aus dem Stellwert LMN als Eingangssignal die beiden Ausgangssignale Stellwert A und Stellwert B.

Das folgende Bild zeigt die Wirkung der Parameter für den Ausgang Stellwert A:

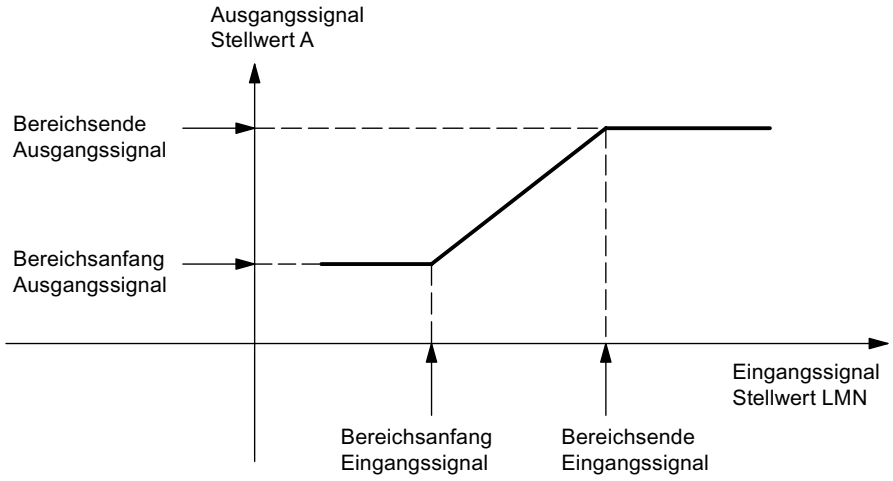

Das folgende Bild zeigt die Wirkung der Parameter für den Ausgang Stellwert B:

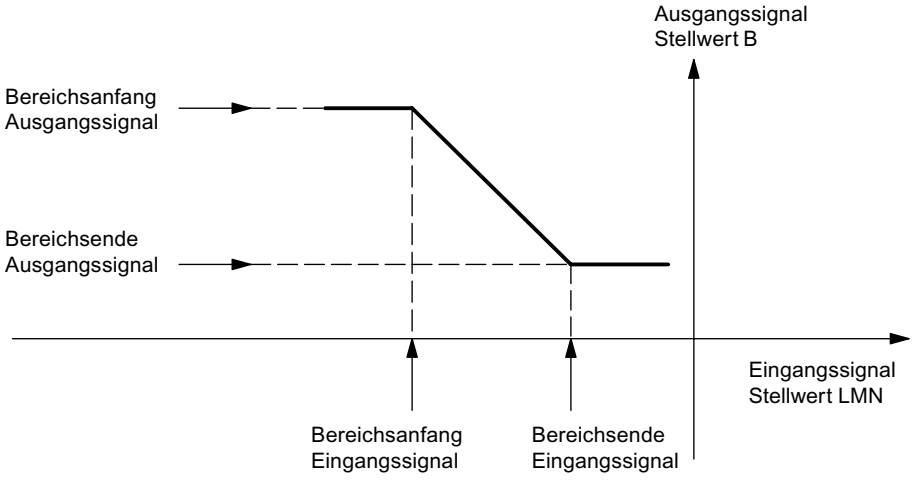

Der Bereichsanfang des Eingangssignals muss kleiner sein als das Bereichsende des Eingangssignals.

Splitrange/Impulsformer

Die Splitrangefunktion ist die Vorbereitung des Analogsignals zur Umsetzung in ein Binärsignal.

Bei einem **Zweipunktregler** (z. B. einem Heizregler) ist nur der Stellwert A relevant. Die Wandlung des Stellwertes in den Stellwert A wird in der nachstehenden Abbildung dargestellt. Die Umsetzung in ein binäres Ausgangssignal erfolgt so, dass am zugeordneten Digitalausgang das Verhältnis von Impulslänge zu Periodendauer dem Stellwert A entspricht.

Zum Beispiel ergibt ein Stellwert A von 40% bei einer Periodendauer von 60 Sekunden eine Impulslänge von 24 Sekunden und eine Pausenlänge von 36 Sekunden.

Das folgende Bild zeigt die Splitrangefunktion an einem Zweipunktregler:

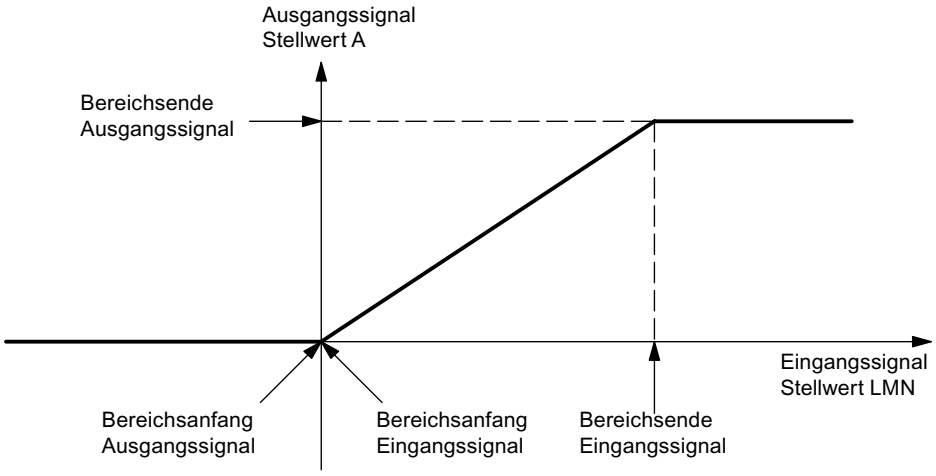

Bei einem **Dreipunktregler** (z. B. einem Heiz- und Kühlregler) gelten für den Stellwert A die oben gemachten Aussagen. Das zweite Signal zur Ansteuerung der Kühlung wird über den Stellwert B gebildet. Die Wandlung des Stellwertes in die Stellwerte A und B wird im Bild unten dargestellt. Die Umsetzung in ein binäres Ausgangssignal erfolgt so, dass an den zugeordneten Digitalausgängen das Verhältnis von Impulslänge zu Periodendauer den Stellwerten A bzw. B entspricht.

Das folgende Bild zeigt die Splitrangefunktion an einem Dreipunktregler:

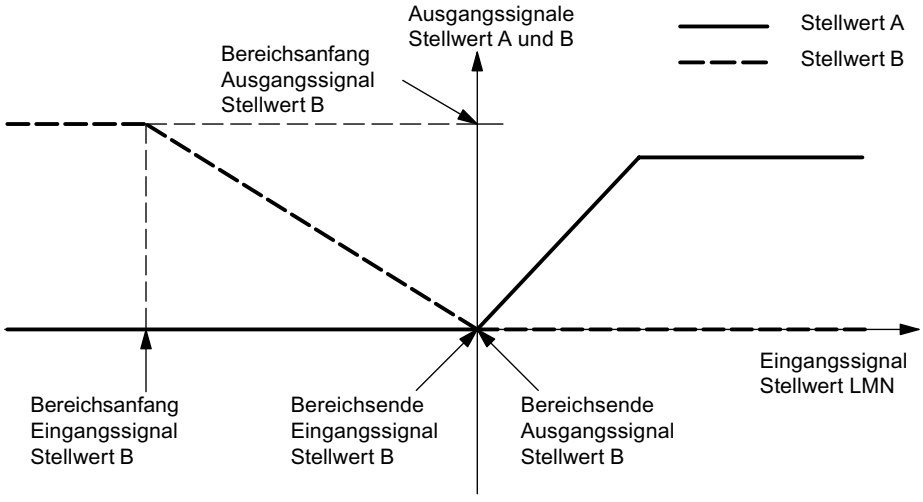

Impulsformer des Schrittreglers

Übersicht

Der Impulsformer steht nur für Schrittregler (FM 355 S, FM 355-2 S und FM 455) zur Verfügung. Folgende Einstellmöglichkeiten existieren.

Impulsformer (nur S-Regler)

- Motorstellzeit
- Mindestimpulsdauer
- Mindestpausendauer

Ausgangssignal und Signaltyp wählen

Analogausgang

Am Analogausgang können Sie für jeden Kanal wählen, welches Signal ausgegeben werden soll. Meist ist dies der Stellwert A eines Reglers. Sie können jedoch auch den Stellwert B eines Reglers oder auch einen Analogeingangswert wählen. Letzteres können Sie für die Linearisierung eines Analogwertes nutzen. So lässt sich z. B. das von einem Thermoelement gelieferte Signal linearisieren und auf 0 bis 10 V umsetzen.

Reglerausgang des Impulsreglers

Das folgende Bild zeigt den Reglerausgang des S-Reglers (Betriebsart Impulsregler):

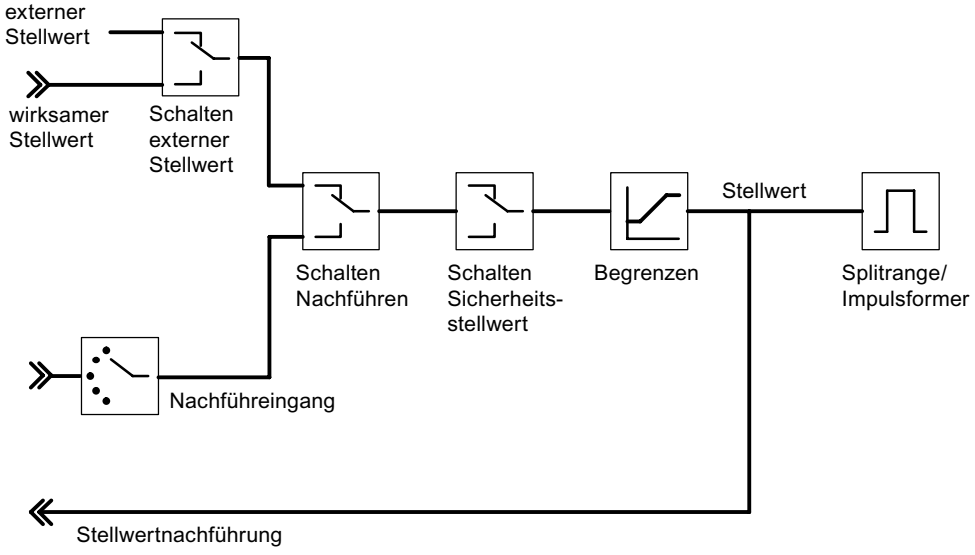

Reglerausgang des Schrittreglers

Das folgende Bild zeigt den Reglerausgang des S-Reglers (Betriebsart Schrittregler mit Stellungsrückmeldung):

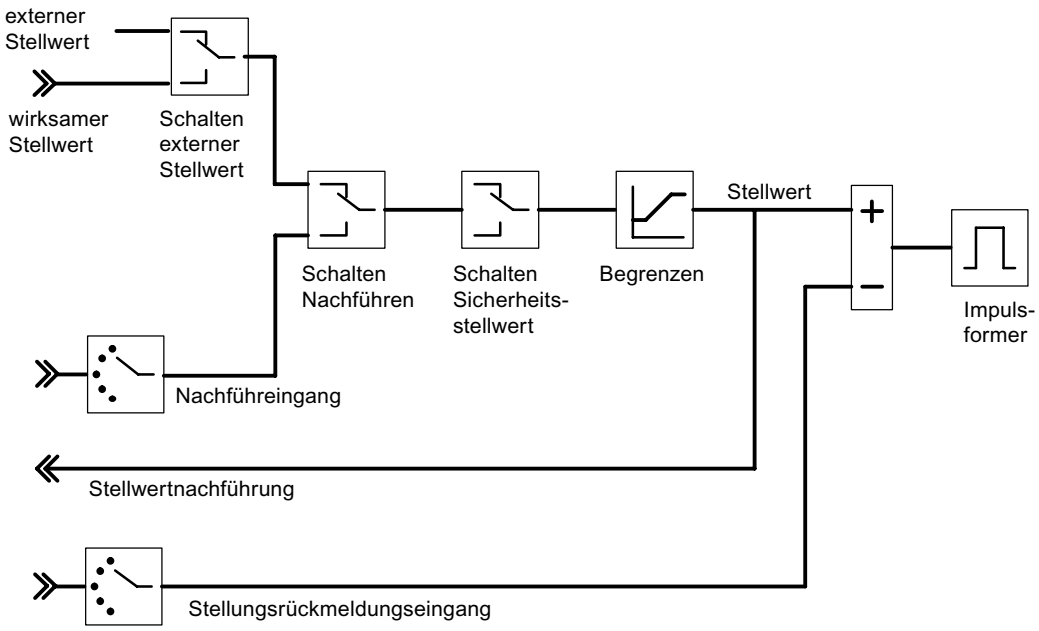

Das folgende Bild zeigt den Reglerausgang des S-Reglers (Betriebsart Schrittregler ohne Stellungsrückmeldung):

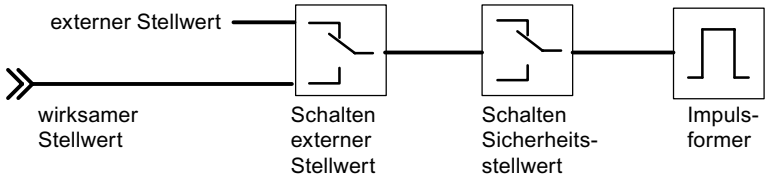

Beim Schrittregler ohne analoge Stellungsrückmeldung wirken der externe Stellwert und der Sicherheitsstellwert wie folgt:

Wenn ein Wert zwischen 40.0% und 60.0% vorgegeben wird, dann wird kein Binärausgang gesetzt, die Stelleinrichtung bleibt unverändert.

Wenn ein Wert > 60.0% vorgegeben wird, dann wird "Stellsignal hoch" so lange ausgegeben, bis die Rückmeldung "Stelleinrichtung am oberen Anschlag" anspricht.

Wenn ein Wert < 40.0% vorgegeben wird, dann wird "Stellsignal tief" so lange ausgegeben, bis die Rückmeldung "Stelleinrichtung am unteren Anschlag" anspricht.

3.5.4.5 Unterstützung Shared Device

Die Reglerbaugruppen FM 355 und FM 355-2 unterstützen die Funktionalität Shared Device bei Verwendung eines ET 200M Interfacemoduls.

Allgemeine Informationen zu Shared Device finden Sie unter Shared Device konfigurieren.

Nutzung von Shared Device mit den Reglerbaugruppen

Zur Nutzung von Shared Device mit den Reglerbaugruppen müssen zusätzlich folgende Organisationsbausteine in Ihrem Programm enthalten sein:

- **Ziehen/Stecken-OB (OB 83):**
  Bei der Projektierung als Shared Device ist der OB 83 in die Bausteinliste einzubinden, da mit dem Übergang von STOP nach RUN so genannte "Return of Submodule"-Alarme generiert werden. Durch den OB 83 werden diese Alarme abgefangen, ansonsten würde die CPU in STOP gehen mit dem Ereignis "STOP durch Ziehen/Stecken".
- **Peripheriezugriffsfehler-OB (OB 122):**
  Solange sich CPU A in STOP und CPU B in RUN befindet, treten in der CPU B "Peripherie-Zugriffsfehler, schreibend" auf, falls im Programm Anweisungen aus den Bibliotheken "FM 355 PID Control" oder "FM 355-2 Temp Control" aufgerufen oder sonstige Schreibzugriffe auf Moduladressen ausgeführt werden. Durch den OB 122 werden diese Fehler abgefangen, ansonsten würde die CPU B mit dem Ereignis "STOP durch Peripheriezugriffsfehler" in STOP gehen.
- **Programmablauffehler-OB (OB 85):**
  Der OB 85 muss nur enthalten sein, falls für die Einstellung "OB85-Aufruf bei Peripheriezugriffsfehler" nicht die Option "Kein OB85-Aufruf" in den Eigenschaften der CPU unter "Zyklus" konfiguriert ist. Bei allen anderen Optionen wird für die Peripheriezugriffsfehler, die oben bei OB 122 beschrieben sind, auch ein OB 85 Aufruf angefordert. Ist der OB 85 in diesem Fall nicht vorhanden, geht die CPU in STOP mit dem Ereignis "STOP durch Programmablauffehler".

Beide CPUs müssen sich in "RUN" befinden, damit Schreibzugriffe auf Moduladressen und die Anweisungen aus den Bibliotheken "FM355 PID Control" und "FM355-2 Temp Control" korrekt ausgeführt werden können.

Solange sich eine CPU in STOP und die andere in RUN befindet, arbeiten alle Reglerbaugruppen dieses Interfacemoduls im Back-up-Betrieb. Dies ist unabhängig vom "Zugriff", der in den "Shared Device"-Einstellungen des Interfacemoduls vorgenommen wurde. Erst wenn sich alle CPUs in RUN befinden, arbeiten die Reglerbaugruppen im Normalbetrieb.

3.5.5 FM 355 einsetzen

3.5.5.1 FM 355 konfigurieren

Hardware konfigurieren

Vorgehensweise

Konfigurieren setzt voraus, dass Sie ein Projekt angelegt haben, in dem Sie die Parametrierung speichern können. Nachfolgend sind nur die wichtigsten Schritte erläutert.

1. Starten Sie das TIA-Portal und öffnen Sie die Gerätekonfiguration.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie eine Reglerbaugruppe aus dem Baugruppenkatalog aus.
4. Ziehen Sie die Reglerbaugruppe in die entsprechende Stelle auf dem Baugruppenträger.
5. Notieren Sie aus der Konfigurationstabelle die Eingangsadresse der Baugruppe, z. B. 272. Der Wert, den Sie auslesen, wird im Dezimalformat angezeigt.

3.5.5.2 FM 355 parametrieren

Baugruppenparameter

Vorgehensweise

Beim Parametrieren stellen Sie die Baugruppenparameter ein.

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung". Wählen Sie im Inspektorfenster die Registerkarte "Eigenschaften".
   Ergebnis: Die Eigenschaften werden im Inspektorfenster angezeigt.
2. Parametrieren Sie die Grundparameter der Baugruppe und die übrigen Parameter.
3. Parametrieren Sie in der Projektieransicht die weiteren Bausteinparameter.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".
5. Übersetzen Sie das Projekt mit "Bearbeiten > Übersetzen".
6. Laden Sie die Parametrierdaten im STOP-Zustand zur CPU mit "Online > Laden in Gerät..."
   Ergebnis: Die Daten befinden sich im Speicher der CPU und werden von dort direkt zur Baugruppe übertragen.
7. Führen Sie einen CPU-Anlauf durch.

Analogeingangsparameter

Auswertung des Analogeingangs

Vorgehensweise

Zum Auswählen eines Sensortyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Auswertung des Analogeingangs".
3. Legen Sie die folgenden Parameter fest:
   - Auflösung Analogeingang
   - Sensortyp Analogeingang
     Für verschiedene Sensortypen wird die Option "Drahtbruchüberwachung" freigeschaltet. Je nach Anforderung kann die Option aktiviert werden.
   - Vergleichsstellentemperatur
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Filterfunktionen

Vorgehensweise

Zum Festlegen der Filterfunktionen gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Filterfunktionen".
3. Legen Sie die folgenden Parameter fest:
   - Zum Filtern nach einer Zeitkonstante aktivieren Sie die Option "Signalfilter".
     Ist die Option "Signalfilter" ausgewählt, kann die Filterzeitkonstante gesetzt werden.
   - Filterzeitkonstante
     Kann nur gesetzt werden, wenn die Option "Signalfilter" aktiviert ist.
   - Zum Linearisieren des Gebersignals aktivieren Sie die Option "Quadratwurzelfilter".
   - Zum Linearisieren der Gebersignalkennlinie aktivieren Sie die Option "Polygonfilter"
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalfilter

Filter 1. Ordnung, dessen Zeitverhalten durch die Zeitkonstante festgelegt wird.

Quadratwurzelfilter

Dient zur Linearisierung von Gebersignalen, bei denen der Istwert als physikalische Größe vorliegt, die in quadratischem Zusammenhang mit der gemessenen Prozessgröße steht.

Polygonfilter

Dient zur Linearisierung von Geberkennlinien.

Die 13 wählbaren Stützpunkte sind wählbar in:

- mA bei Stromeingang
- mV bei Spannungseingang

Normierung

Vorgehensweise

Zum Wandeln des Eingangssignals in eine physikalische Einheit gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Normierung".
3. Legen Sie die folgenden Parameter fest:
   - Oberer Wert
   - Unterer Wert
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Normierung

Zur Wandlung des Eingangssignals in eine andere physikalische Einheit durch lineare Interpolation zwischen Anfangswert (unter Wert) und Endwert (oberer Wert).

Regeldifferenz

Festwert- oder Kaskadenregler realisieren

Festwert- oder Kaskadenregler

Bei den Regelungsstrukturen Festwert- oder Kaskadenregler ist der wirksame Istwert mit dem Istwert A identisch.

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Festwert- oder Kaskadenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Aufbereiten des Sollwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Setzen Sie in den verschiedenen Bereichen die entsprechenden Parameter:
   - Sollwert Quelle
     Wählen Sie die Quelle des Sollwerts.
   - Sicherheitssollwert
     Stellen Sie hier den Sicherheitssollwert ein. Legen Sie ebenfalls die Reaktion auf einen CPU-Ausfall und das Anlaufverhalten fest.
   - Sollwert Rampe
     Legen Sie die Hochlaufzeit fest.
   - Sollwertgrenze
     Wählen Sie hier die obere und die untere Sollwertgrenze aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A wählen

Zum Auswählen der Signalquelle für den Istwert A gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz ".
3. Wählen Sie im Bereich "Stellwert" die entsprechende Quelle. Zur Verfügung stehen folgende Auswahlmöglichkeiten:
   - Null
   - Aufbereiteter Analogeingang
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Dreikomponentenregler realisieren

Dreikomponentenregler

Bei der Regelungsstruktur Dreikomponentenregler wird der wirksame Istwert durch Summieren der drei Istwerte A, B und C und durch Addition eines einstellbaren Offsets gebildet. Die Istwerte B und C können zusätzlich durch Faktoren bewertet werden.

Reglertyp wählen

Zum Realisieren eines Dreikomponentenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Dreikomponentenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.
6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A, B und C wählen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".

3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Faktoren für Istwert B und C sowie Offset erfassen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Legen Sie im Bereich "Summieren" für den Istwert B und C den erforderlichen Gewichtungsfaktor fest.
4. Legen Sie den notwendigen "Offset" fest, der zu den Istwerten addiert werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Verhältnis- oder Mischungsregler realisieren

Reglertyp wählen

Zum Realisieren eines Verhältnis- oder Mischungsreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Verhältnis- oder Mischungsregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.

6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A und D wählen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge, für den Istwert D können Sie auch den Stellwert eines der Regler auswählen.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Multiplikator erfassen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Multiplizieren" den gewünschten Offset.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Beim Verhältnisregler wird der Istwert A als Regelgröße verwendet, der Istwert D als Verhältnisgröße. Der Sollwerteingang dient als Verhältnisfaktor. Er wird durch Multiplikation mit dem Istwert D und Addition eines einstellbaren Offsets zum wirksamen Sollwert aufbereitet. Wird der Istwert D abgeschaltet, dann wird zum Sollwert nur der Offset addiert.

Alarmgenerierung einstellen

Vorgehensweise

Zum Einstellen der Alarmgenerierung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".

3. Wählen Sie im Bereich "Alarmgenerierung" die erforderlichen Optionen aus:
   - Wählen Sie bei "Grenzwertmelder für" aus, für welches Signal der Alarm gesetzt werden soll.
   - Unter "Alarm obere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der oberen Grenze ausgelöst wird.
   - Unter "Warnung obere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die obere Grenze ausgelöst wird.
   - Unter "Warnung untere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die untere Grenze ausgelöst wird.
   - Unter "Alarm untere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der unteren Grenze ausgelöst wird.
   - Zum Vermeiden von Flackern der Überwachungsanzeigen kann unter "Hysterese" eine Hysterese parametriert werden.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Regelalgorithmus

P-Regelung realisieren

Vorgehen

Zum Realisieren einer P-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "P" aus.
   Alle Optionen, die für eine P-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PI-Regelung realisieren

Vorgehen

Zum Realisieren einer PI-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.

4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PI" aus.
   Alle Optionen, die für eine PI-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

I-Regelung realisieren

Vorgehen

Zum Realisieren einer I-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "I" aus.
   Alle Optionen, die für eine I-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PID-Regelung realisieren

Vorgehen

Zum Realisieren einer PID-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PID" aus.
   Alle Optionen, die für eine PID-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PD-Regelung realisieren

Vorgehen

Zum Realisieren einer PD-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.

4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PD" aus.
   Alle Optionen, die für eine PD-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Fuzzy Temperaturregler realisieren

Vorgehen

Zum Realisieren einer Fuzzy-Temperaturregelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "Fuzzy Temperatur-Reglerbaugruppe" aus.
   Alle Optionen, die für eine Fuzzy Temperatur-Reglerbaugruppe parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Rauschanteil der Regeldifferenz unterdrücken

Vorgehen

Zum Unterdrücken des Rauschanteils der Regeldifferenz gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Legen Sie im Bereich "Allgemein" die Einstellung "Totzonenbreite" fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Störgröße aufschalten

Vorgehen

Zum Aufschalten einer Störgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Allgemein" aus der Liste "Störgrößenquelle" die Quelle für das Störgröße aus.
   Wählen Sie den Eintrag "Null" aus, um keine Störgröße aufzuschalten.

4. Geben Sie unter "Störgrößenfaktor" einen Wert ein, mit dem Sie die Störgröße gewichten wollen.
   Ein Wert < 0 schwächt die Störgröße ab, ein Wert > 0 verstärkt die Störgröße. Beim Wert 1 bleibt die Störgröße unverändert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

P-Anteil in Rückführung legen

Vorgehen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Wählen Sie im Bereich "P-Anteil" die Option "In Rückführung" aus.
   Ist die Option aktiviert, ist der P-Anteil in Rückführung gelegt.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

D-Anteil bei Sollwertänderungen abschwächen

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "D-Anteil" mit der Option "D-Anteil" die Quelle für den D-Anteil aus.
4. Legen Sie im Bereich "Differenzierfaktor" den Faktor für den D-Anteil fest.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerwirkung umkehren

Prinzip

Die Umkehrung des Reglers von der Zuordnung "steigende Regeldifferenz = steigende Stellgröße" auf "steigende Regeldifferenz = fallende Stellgröße" erreichen Sie durch das Einstellen eines negativen Proportionalbeiwerts (GAIN).

Vorgehen

Zum Umkehren der Reglerwirkung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".

3. Legen Sie im Bereich "Allgemein" einen Proportionalbeiwert mit umgekehrtem Vorzeichen fest.
   Ist also ein positiver Proportionalbeiwert angegeben und die Reglerwirkung soll umgekehrt werden, legen Sie einen negativen Proportionalbeiwert fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerausgang

Kontinuierlichen Regler realisieren

Ein kontinuierlicher Regler kann nur mit einer analogen Baugruppe realisiert werden.

Vorgehen

1. Externen Stellwert schalten (Seite 1029)
2. Sicherheitsstellwert einstellen (Seite 1030)
3. Nachführung schalten (Seite 1030)
4. Stellwert begrenzen (Seite 1031)
5. Zwei Stellglieder ansteuern - Splitrange (Seite 1031)
6. Ausgangssignal und Signaltyp wählen (Seite 1033)

Schrittregler ohne Stellungsrückmeldung

Ein Schrittregler ohne Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler ohne Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1029)
3. Sicherheitsstellwert einstellen (Seite 1030)
4. Impulsformer Schrittregler (Seite 1032)

Schrittregler mit Stellungsrückmeldung realisieren

Ein Schrittregler mit Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler mit Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1029)
3. Nachführung schalten (Seite 1030)

4. Sicherheitsstellwert einstellen (Seite 1030)
5. Stellwert begrenzen (Seite 1031)
6. Stellungsrückmeldung einstellen (Seite 1030)
7. Impulsformer Schrittregler (Seite 1032)

Impulsregler realisieren

Ein Impulsregler kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Impulsregler wählen
2. Externen Stellwert schalten (Seite 1029)
3. Nachführung schalten (Seite 1030)
4. Sicherheitsstellwert einstellen (Seite 1030)
5. Stellwert begrenzen (Seite 1031)
6. Analogsignal vorbereiten auf die Umwandlung in Digitalsignal
7. Analogsignal in Digitalsignal wandeln (Seite 1032)

Funktionen des Reglerausgangs parametrieren

Externen Stellwert schalten

Vorgehen

Zum Schalten der externen Stellgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Externe Stellgröße" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   – Aktivierung vom Funktionsbaustein.
   – Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Nachführung schalten

Vorgehen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Nachführung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Wählen Sie im Bereich "Nachführung" die Quelle des Signals für die Nachführung aus. Dabei können Sie wählen aus den folgenden Optionen:
   - Null, dabei wird keine Quelle verwendet.
   - Einer der Analogeingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellungsrückmeldung einstellen

Vorgehen

Zum Einstellen der Stellungsrückmeldung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".
3. Wählen Sie im Bereich "Stellungsrückmeldungseingang" die Option "Quelle" aus, um das Signal für die Stellungsrückmeldung auszuwählen.
   - Null: keine Stellungsrückmeldung.
   - Einer der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sicherheitsstellwert einstellen

Vorgehen

Zum Einstellen des Sicherheitsstellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Sicherheitsstellwert" den Wert fest, der als Sicherheitsstellwert verwendet werden soll.

4. Wählen Sie unter "Aktivierung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   – Aktivierung vom Funktionsbaustein.
   – Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
5. Wählen Sie unter "Reaktion bei Anlauf der Baugruppe" die Option aus, die beim Anlaufen der Baugruppe durchgeführt werden soll. Dabei können Sie aus wählen aus den folgenden Optionen:
   – Regelbetrieb: versetzt die Baugruppe beim Anlaufen sofort in den Regelbetrieb.
   – Sicherheitsstellwert: zunächst wird auf den Sicherheitsstellwert geregelt.
6. Wählen Sie unter "Reaktion bei Messumformerstörung des Istwert A" die Option aus, die bei einer Messumformerstörung des Istwert A durchgeführt werden soll.
7. Wählen Sie unter "Reaktion auf Messumformerstörung eines Eingangs" die Option aus, die bei einer Messumformerstörung eines Eingangs durchgeführt werden soll.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellwert begrenzen

Vorgehen

Zum Begrenzen des Stellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Grenzwert Ausgangssignal" die die folgenden Werte fest:
   – Oberer Wert: stellt den oberen Grenzwert für das Ausgangssignal dar.
   – Unterer Wert: stellt den unteren Grenzwert für das Ausgangssignal dar.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Zwei Stellglieder ansteuern - Splitrange

Vorgehen

Zum Ansteuern von zwei Stellgliedern gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Split Range" die Option "Aktivieren" aus, um zwei Stellglieder anzusteuern.

4. Legen Sie die folgenden Werte sowohl für die Ein- und Ausgangssignale A als auch für die Ein- und Ausgangssignale B fest:
   - Unteres Eingangssignal.
   - Oberes Eingangssignal.
   - Unteres Ausgangssignal.
   - Oberes Ausgangssignal.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Impulsformer Schrittregler

Vorgehen

Zum Parametrieren des Impulsformers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".
3. Legen Sie im Bereich "Impulsformer" folgende Parameter fest:
   - Motorstellzeit.
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Wählen Sie für den oberen und den unteren Wert die passende Option aus, woher der Wert übernommen werden soll:
   - Vom Funktionsbaustein.
   - Vom Funktionsbaustein oder von einem der Digitaleingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Analogsignal in Digitalsignal wandeln

Vorgehen

Zum Umwandeln eines Analogsignals in ein Digitalsignal gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Kanal [Nr.]".
3. Legen Sie im Bereich "Impulsformer" die folgenden Parameter fest:
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Ausgangssignal und Signaltyp wählen

Vorgehen

Zum Auswählen des Ausgangssignals und des Signaltyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogausgang > Ausgang [Nr.]".
3. Wählen Sie unter der Option "Quelle" aus, welches Signal als Ausgangssignal verwendet werden soll. Zur Auswahl stehen:
   - Null: keine Quelle für das Ausgangssignal.
   - Einer der Analogeingänge.
   - Einer der Stellwerte für Regler A bzw. B.
4. Wählen Sie unter der Option "Signaltyp" den Typ aus, in dem das Ausgangssignal ausgegeben werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

3.5.5.3 FM 355 programmieren

3.5.5.4 FM 355 in Betrieb nehmen

Instanz-DBs anlegen und versorgen

Vorgehen

Um die Voraussetzungen für die Inbetriebnahme zu erfüllen, gehen Sie folgendermaßen vor:

1. Erstellen Sie ein Anwenderprogramm, das die Anweisung PID_FM zyklisch in einem Weckalarm-OB aufruft.
2. Setzen Sie im Anwenderprogramm READ_VAR zyklisch TRUE.
3. Erzeugen Sie für jeden Reglerkanal einen Einzel-Instanz-DB der Anweisung PID_FM.
4. Tragen Sie bei jedem Instanz-DB die Baugruppenadresse in den Parameter PID_FM.MOD_ADDR ein.
   Als Baugruppenadresse der FM 355 verwenden Sie die Anfangsadresse, die Sie beim Konfigurieren Ihrer Hardware festgelegt haben.
5. Tragen Sie in den Parameter PID_FM.CHANNEL die Nummer des Reglerkanals (1 ... 4) ein.

6. Wenn Sie die Anweisung PID_PAR verwenden, tragen Sie in den Parameter PID_PAR.CHANNEL die Nummer des Reglerkanals ein.
7. Kontrollieren Sie, dass der "Sensortyp Analogeingang" dieses Kanals aktiviert ist. Wenn der Analogeingang nicht bearbeitet wird, werden alle PID-Parameter auf Null gesetzt und es wird kein Stellwert berechnet.

Hinweis

Multiinstanz-DBs

Wenn Sie in Ihrem Projekt Multiinstanz-DBs verwenden, müssen Sie die FM über eine Beobachtungstabelle in Betrieb nehmen.

Regler in Betrieb nehmen

Einführung

Sie können die PID-Parameter folgendermaßen optimieren:

- PID-/PI-Regler(Schrittregler, Kontinuierlicher Regler, Impulsregler): Manuell
- Fuzzy-Temperaturregler: Automatisch
- Schrittregler: Automatisch mit der Anweisung TUN_ES
- Kontinuierliche Regler: Automatisch mit der Anweisung TUN_EC
- Impulsregler: Automatisch mit der Anweisung TUN_EC

Vorgehen

Zum Optimieren eines Reglers gehen Sie wie folgt vor:

1. Bringen Sie die CPU in den Zustand RUN.
2. Bei Verwendung eines Schrittreglers messen Sie die Motorstellzeit.
3. Optimieren Sie die PID-Parameter mit einer der oben genannten Möglichkeiten.
4. Beobachten Sie den Regelkreis mit dem Kurvenschreiber.

Siehe auch

Motorstellzeit messen (Seite 1041)

PID-Parameter manuell optimieren

Voraussetzung

Sie haben bei der Parametrierung des Regelalgorithmus "PID-Regler" gewählt.

Vorgehen

Um einen PID-/PI-Regler manuell einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Geben Sie für den P-Anteil, den I-Anteil, den D-Anteil und die Verzögerung die entsprechenden Werte an.
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "PID-Parameter an FM senden".
5. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
6. Geben Sie einen neuen Sollwert ein.
7. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb" und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
   Der Regler arbeitet mit den neuen Parametern und regelt auf den neuen Sollwert.
8. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
9. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

Fuzzy-Streckenidentifikation

Anforderungen an den Prozess beim Fuzzy-Temperaturregler

Voraussetzung

Um eine optimale Regelung mit Hilfe des Fuzzy-Temperaturreglers zu erreichen, sollte der Prozess folgenden Anforderungen genügen:

- Bei Badheizungen muss die zu heizende Flüssigkeit gut durchgemischt werden.
- Für Bad-im-Bad-Regelungen müssen beide Flüssigkeiten durchgerührt werden. Gleichzeitig ist für gute Wärmeübergänge zwischen allen wärmeübertragenden Medien zu sorgen. Bei schlecht wärmeübertragenden Materialien sollten große Übertragungsflächen einen guten Wärmetransport ermöglichen.
- Bei Raumlufttemperaturregelungen muss für eine gute Durchmischung (z. B. mit Ventilatoren) gesorgt werden.
- Die Verstärkung der Regelstrecke darf den Faktor 3 nicht überschreiten.
- Die Verzugszeit darf höchstens 3 % der Ausgleichszeit sein.
- Die zu regelnde Temperatur sollte sich innerhalb der Abtastzeit des Reglers bei maximaler Stellgrößenausgabe um höchstens 1 ‰ der vorgegebenen Maximaltemperatur ändern.

Klassifikation der Regelstrecken

Eine Regelstrecke bzw. ein zu regelnder Prozess zeichnet sich durch Parameter aus wie z. B. die Heizleistung, die Heizungsmasse oder die Wärmekapazität des zu heizenden Mediums. Bezüglich des Fuzzy-Reglers unterscheidet man eine "kritische" und eine "nicht kritische" Temperaturregelstrecke wie folgt: Die Regelung wird um so kritischer,

- je größer die Heizleistung,
- je größer die Wärmekapazität der Heizung,
- je kleiner die Wärmekapazität des zu heizenden Mediums,
- je größer der Wärmeübergangswiderstand,
- je kleiner die Wärme übertragende Fläche ist.

Nach Aufschaltung eines Stellwertsprungs auf die Regelstrecke reagiert diese mit einer Sprungantwort. Anhand dieser Sprungantwort kann man die Klassifikation der Regelstrecke ebenfalls vornehmen: Die Regelung wird um so kritischer, je größer das Verhältnis t_u / t_g ist und je größer die Verstärkung der Regelstrecke ist. Mit t_u / t_g < 1/10 hat man eine unkritische Regelstrecke.

Siehe auch

Kennwerte der Regelstrecke (Seite 635)

Ablauf der Fuzzy-Streckenidentifikation

Ablauf der Parameteroptimierung beim Fuzzy-Temperaturregler

Die Parameteroptimierung basiert auf einem selbstoptimierenden Fuzzy-Regler.

Die Identifikation der Regelstrecke beginnt mit einer Beobachtungsphase, während der keine Heizleistung ausgegeben wird. Die Dauer der Beobachtungsphase ist wie folgt:

- Beobachtungsphase kontinuierlicher Regler: ca. 1 min
- Beobachtungsphase Schrittregler: ca. 1 min + 1/2 x Stellgliedlaufzeit

Diese Zeit wird genutzt, um Temperaturtrends im Heizmedium feststellen zu können. Danach wird maximale Heizleistung von 100% ausgegeben; diese ist sichtbar am Ausgangsparameter LMN im Instanz-DB der Anweisung PID_FM.

Der Bereich der ersten 4 % Temperaturerhöhung vom Sollwertbereich wird für die Identifikation genutzt, wobei jeweils bei 1 % und 4 % Temperaturerhöhung anhand der verstrichenen Zeit Informationen über das Prozessverhalten gewonnen werden.

Die Identifikation ist abgeschlossen, wenn die Heizleistung geringer als 100% wird. Dieses Verhalten kann genutzt werden, um z. B. die Identifikation über das Anwenderprogramm auszuschalten. Nach der Identifikationsphase arbeitet der Regler mit den ermittelten Parametern weiter.

Wird der Sollwertsprung nicht ausgeregelt und bleibt die Heizleistung dauerhaft auf Null, dann ist die Identifikation erfolglos beendet worden, d. h. der Regler kann die angeschlossene Regelstrecke nicht regeln.

Das Reglerverhalten bei verschiedenen Regelstrecken

Bei einer "unkritischen" Regelstrecke hat man keine Probleme, weder bei der Identifikation noch bei der Regelung.

Die Identifikation einer "zu kritischen" Regelstrecke wird abgebrochen. Die Regelung einer identifizierten "kritischen" Regelstrecke erfolgt sehr "vorsichtig" bzw. langsam.

Fuzzy-Streckenidentifikation durchführen.

Voraussetzung

Sie haben bei der Parametrierung des Regelalgorithmus "Fuzzy-Regler" gewählt.

Nach Möglichkeit müssen Sie dafür sorgen, dass sich die Regelstrecke vor der Identifikation beruhigt hat (kein Aufheiz- oder Abkühlvorgang) oder langsam monoton ändert. Das Kriterium dafür kann sein, dass die Temperaturänderung während einer Minute Zeit mit einer Geraden anzunähern ist. Bei schnelleren Prozessen ist diese Anforderung besonders aktuell.

Da zu Beginn der Identifikation vom Regler für ca. eine Minute die Stellgröße Null ausgegeben wird, muss die zu regelnde Temperatur nahe der Umgebungstemperatur sein.

Fuzzy-Streckenidentifikation starten

Um einen Fuzzy-Regler zu optimieren, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Geben Sie in der Gruppe "Optimierung" einen neuen Sollwert (SP_OP) ein.
   Der neue Sollwert muss folgende Bedingungen erfüllen:
   – Sollwert-Sprung > 5 %
   – Neuer Sollwert > Istwert + Obergrenze Sollwert x 0,12
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Fuzzy-Streckenidentifikation".

Ergebnis

Der neue Sollwert (SP_OP), SP_OP_ON = TRUE und FUZID_ON = TRUE werden an die FM gesendet. Die Fuzzy-Streckenidentifikation startet. Der Parameter IDSTATUS der Anweisung CH_DIAG liefert Informationen über den Zustand der Identifikation. Im Anzeigefeld Status sehen Sie, ob eine Fuzzy-Streckenidentifikation aktiviert ist oder läuft.

Fuzzy-Streckenidentifkation deaktivieren

Solange das Bit FUZID_ON = TRUE ist, wird mit jedem weiteren ausreichend großen Sollwertsprung die nächste Identifikation gestartet.

Um den Optimierungsmodus auszuschalten, gehen Sie folgendermaßen vor:

1. Fügen Sie FUZID_ON in einer Beobachtungstabelle.
2. Setzen Sie FUZID_ON = FALSE.

Hinweis

Ausschalten des Optimierungsmodus FUZID_ON = FALSE vor Ende der Identifikation hält die Identifikation nicht an. Die einmal gestartete Identifikation läuft unter allen Umständen weiter – außer bei negativem Sollwertsprung.

Fuzzy-Streckenidentifikation wiederholen

Bei jedem weiteren Sollwertsprung von ≥12 % wird eine erneute Identifikation durchgeführt, es sei denn, die Identifikation wird mit FUZID_ON = FALSE wieder ausgeschaltet.

Es besteht auch die Möglichkeit, die Identifikation erneut zu starten durch Verkleinern und anschließendes Vergrößern des Sollwertes um 12> %.

Fuzzy-Streckenidentifikation abbrechen

Die Identifikation kann in folgenden Fällen abgebrochen werden:

- vom Regler bei der Identifikation einer "kritischen" Regelstrecke. Nach dem Abbruch befindet sich der Regler im Fehler-Zustand. Dieser Zustand wird dadurch angezeigt, dass die Stellgröße vom Regler auf Dauer zurückgesetzt wird. Dies ändert sich auch nicht beim Ausschalten der Optimierung. Das Starten einer neuen Identifikation löscht den Fehler-Zustand.
  Informationen über den Zustand der Identifikation erhalten Sie durch den Parameter IDSTATUS des FB CH_DIAG. Mehr dazu am Ende dieses Kapitels unter "Reglerzustandsinformationen".
- vom Bediener durch die Erzeugung eines negativen Sollwertsprungs, wobei der Sollwert die Bedingung 2 (siehe oben, "Wie starten Sie die Identifikation") unterschreiten muss.

Um die Optimierung abzubrechen, gehen Sie folgendermaßen vor:

1. Geben Sie einen neuen Sollwert ein, der unter dem aktuellen Istwert liegt und die gleichen Bedingungen erfüllt, wie zum Start der Fuzzy-Streckenidentifikation.
2. Klicken Sie in der Gruppe "Optimierung" auf das Symbol ▶ "Fuzzy-Streckenidentifikation".

Optimierte Parameter in Gerät laden

Einführung

Damit die Parameter auf der FM nicht beim nächsten Anlauf der CPU mit den Systemdaten überschrieben werden, können Sie folgendermaßen vorgehen:

- Sie laden nach jedem Anlauf der FM mit LOAD_PAR=TRUE die optimierten PID-Parameter aus dem Instanz-DB des PID_FM in die FM.
- Sie laden die optimierten PID-Parameter in den SDB.

Hinweis

Die PI-, PID- und SAVE-Parametersätze können nicht im SDB gesichert werden.

Vorgehen

Um die Parameter in den SDB zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die PID-Parameter werden in der Parametereinstellung der FM gespeichert.
2. Stoppen Sie die CPU.
3. Klicken Sie auf das Symbol "Laden in Gerät".

Mit dem Kurvenschreiber arbeiten

Beobachten

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".
3. Wählen Sie den Instanz-DB aus, den Sie beobachten wollen. Nach dem Auswählen wird der Kanal angezeigt, dem der Instanz-DB zugeordnet ist.
4. Legen Sie die Aktualisierungszeit fest, mit der der Kurvenschreiber protokollieren soll.
5. Klicken Sie auf "Messung ein", um die Messung zu starten.
6. Klicken Sie auf "Messung aus", um die Messung zu beenden.

Vorgehen

Zum Anpassen des Maßstabs gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".

3. Zum Ändern der Zeitachse:
   - Klicken Sie auf die Pfeil-Schaltfläche neben dem Datum, um einen Kalender zu öffnen. Wählen Sie aus dem Kalender ein Datum aus. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für das Start- als auch für das Enddatum.
   - Klicken Sie mit dem Mauszeiger in die Zeitskala. Der Mauszeiger wird als Pfeil links/rechts dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach links oder rechts.
     Ist die Anfangs- oder Endzeit gesperrt, ändert sich die Skalierung der Zeit.
     Ist weder Anfangs- noch Endzeit gesperrt, ändert sich der angegebene Zeitraum, die Skalierung bleibt erhalten.
4. Zum Ändern der Werteachse:
   - Klicken Sie auf das Eingabefeld für den oberen bzw. den unteren dargestellten Wert. Geben Sie dort Ihren gewünschten Wert ein. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für den oberen als auch für den unteren Wert.
   - Klicken Sie mit dem Mauszeiger in die Werteskala. Der Mauszeiger wird als Pfeil oben/unten dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach oben oder unten.
     Ist der obere oder untere Wert gesperrt, ändert sich die Skalierung der Werteachse.
     Ist weder der obere noch der untere Wert gesperrt, ändert sich der angegebene Wertebereich, die Skalierung bleibt erhalten.

Parameter über HMI verändern

Vorgehen

Um Reglerparameter der Anweisung PID_FM durch Bedienung über HMI ändern wollen, gehen Sie folgendermaßen vor:

1. Schreiben Sie die zu ändernden Parameter vom HMI aus in einen Hilfs-DB auf der CPU (siehe ①).
2. Setzen Sie PID_FM.COM_RST auf TRUE.
   Die Anweisung führt einen Initialsierungslauf durch. Die Parameter sind zwischen FM und CPU abgeglichen (siehe ②).
3. Übertragen Sie die zu ändernden Parameter aus dem Hilfs-DB in den Instanz-DB der Anweisung PID_FM (siehe ③).
4. Setzen Sie PID_FM.LOAD_PAR auf TRUE.
   Die geänderten Parameter werden zur FM übertragen(siehe ④).

Die Ablage der Parameter in einem Hilfs-DB ist notwendig, weil PID_FM nach dem Anlauf der CPU mit COM_RST = TRUE diejenigen Parameter aus der Baugruppe liest, die die CPU zuvor aus den Systemdaten zur FM übertragen hat.

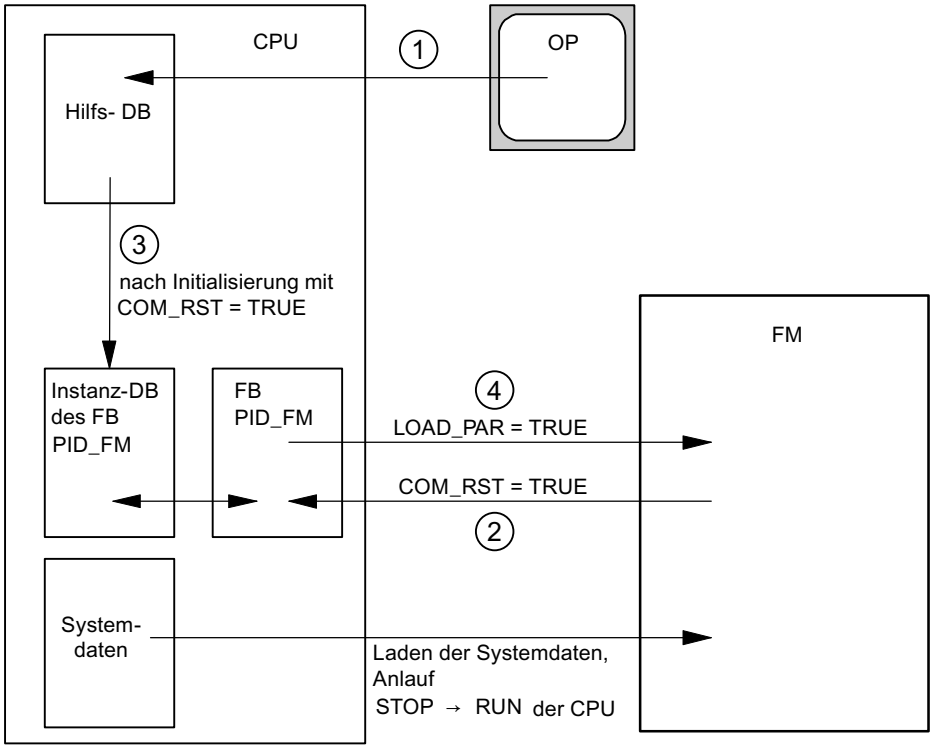

Motorstellzeit messen

Einführung

Die Motorstellzeit wird so genau wie möglich benötigt, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

Voraussetzung

- Die parametrierte Motorstellzeit muss mindestens halb so groß sein wie die reale Stellzeit. Das Stellglied wird maximal die doppelte Stellzeit in eine Richtung gefahren.
- Bei Stellgliedern mit analoger Stellungsrückmeldung muss die Stellungsrückmeldung verschaltet sein.
- Bei Stellgliedern mit Anschlagsignalen müssen die Anschlagsignale verschaltet sein.

Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
3. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

Motorstellzeit = (Obergrenze Ausgangswert – Untergrenze Ausgangswert) × Messzeit / BETRAG(Zielstellung – Startstellung).

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE). Das Stellglied wird nicht mehr bewegt.

Stellglieder mit Anschlagsignalen

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
3. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag. Die Zeitmessung startet, wenn das Stellglied den unteren Anschlag erreicht hat und endet, wenn das Stellglied den unteren Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE). Das Stellglied wird nicht mehr bewegt.

Gemessene Stellzeit laden

1. Um die gemessene Motorstellzeit lokal im Projekt zu speichern, klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die gemessene Stellzeit wird in der Parametereinstellung der FM gespeichert.
2. Um die Änderungen beim nächsten CPU-Start in Kraft treten zu lassen, klicken Sie auf das Symbol "Laden in Gerät".

Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung abbrechen, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE) und der Handbetrieb mit Stellwertvorgabe ausgeschaltet (LMNOP_ON = FALSE). Das Stellglied wird nicht mehr bewegt.

3.5.5.5 FM 355: Diagnose durch LED-Anzeige

Fehleranzeige durch die Sammelfehler-LED

Einleitung

Leuchtet die rote Sammelfehler-LED, so ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten.

Wenn die gelbe LED blinkt, dann ist die Firmware gelöscht worden. Dieser Zustand kann nur auftreten bei einer defekten Hardware oder wenn der Ladevorgang der Firmware abgebrochen wird.

Welche Fehler werden angezeigt?

Folgende Fehler werden durch das Leuchten der Sammelfehler-LED angezeigt:

Fehlerart	Diagnosemeldung	Mögliche Fehlerursache	Behebung
Interne Fehler	Baugruppe defekt	Hardwarefehler	Baugruppe tauschen
	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	Baugruppe tauschen
	EEPROM-Inhalt ungültig	Ausfall der Versorgungsspannung beim Parametrieren	Baugruppe neu parametrieren
Externe Fehler	Falsche Parameter in Baugruppe	An die Baugruppe wurden falsche Parameter übertragen	Baugruppe neu parametrieren
	Fehler an Analogeingängen bzw. Analogausgängen	Analogeingang Hardwarefehler	Baugruppe tauschen
		Analogeingang Drahtbruch	Drahtbruch beseitigen
		Analogeingang Messbereichsunterschreitung	Messsignal überprüfen
		Analogeingang Messbereichsüberschreitung	Messsignal überprüfen
		Analogausgang Drahtbruch	Drahtbruch beseitigen
		Analogausgang Kurzschluss	Kurzschluss beseitigen
	Externe Hilfsspannung fehlt	24-V-Versorgung ausgefallen	24-V-Versorgung wiederherstellen

Diagnosealarm bei Fehlern

Alle Fehler können einen Diagnosealarm auslösen, falls Sie den Diagnosealarm in der entsprechenden Parametriermaske freigegeben haben. Aus den Diagnosedatensätzen DS0 und DS1 erfahren Sie, welche Fehler das Leuchten der LED verursacht haben. Die Belegung der Diagnosedatensätze DS0 und DS1 ist im nächsten Abschnitt beschrieben.

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, so können Sie einen Diagnosealarm parametrieren, der das zyklische Programm der CPU unterbricht und den Diagnosealarm-OB, OB 82, aufruft.

Welche Ereignisse können einen Diagnosealarm auslösen?

Die Liste zeigt Ihnen, welche Ereignisse einen Diagnosealarm auslösen können:

- Parametrierung der Baugruppe fehlt oder ist fehlerhaft
- Baugruppe defekt
- Leitungsbruch bei Analogeingaben (nur 4 bis 20 mA)
- Überlauf und Unterlauf bei Analogeingaben
- Bürdenbruch und Kurzschluss bei Analogausgaben

Defaulteinstellung

Der Diagnosealarm ist defaultmäßig gesperrt.

Freigeben des Diagnosealarms

In der Maske "Grundparameter" sperren Sie den Diagnosealarm für die Baugruppe oder geben ihn frei.

Reaktionen auf ein alarmauslösendes Ereignis

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht folgendes:

- Die Diagnoseinformationen werden in die DiagnosedatensätzeDS0 und DS1 auf der Baugruppe abgelegt.
- Die Sammelfehler-LED leuchtet.
- Der Diagnosealarm-OB wird aufgerufen (OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Falls kein Hardwaredefekt vorliegt, regelt die Baugruppe weiter.

Ist kein OB 82 programmiert, geht die CPU in STOP.

Diagnosedatensatz DS0 und DS1

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Bytes, der DS1 umfasst 16 Bytes, wobei die ersten vier Bytes identisch mit dem DS0 sind.

Diagnosedatensätze DS0 und DS1

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 wird automatisch beim Aufruf des Diagnose-OBs in die Startinformation übertragen. Dort werden diese vier Bytes ins Lokaldatum (Byte 8-11) des OB 82 abgelegt.

Den Diagnosedatensatz DS1 (und damit auch den Inhalt des DS0) können Sie mit der Anweisung "CH_DIAG" von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Die Anweisung "CH_DIAG" muss im selben OB aufgerufen werden wie der FB "PID_FM". Dies erreichen Sie durch folgende Maßnahme: Setzen Sie, während der OB 82 bearbeitet wird, ein Bit. Fragen Sie dieses Bit in dem OB ab, in dem auch der FB "PID_FM" aufgerufen wird, und rufen Sie bei gesetztem Bit die Anweisung "CH_DIAG" auf.

Wie erscheint der Diagnosetext im Diagnosepuffer?

Wenn Sie die Diagnosemeldung in den Diagnosepuffer eintragen wollen, müssen Sie die Anweisung "WR_REC" "Anwenderspezifische Meldung in Diagnosepuffer eintragen" in dem Anwenderprogramm aufrufen. Am Eingangsparameter EVENTN wird die Ereignisnummer der jeweiligen Diagnosemeldung angegeben. Der Alarm wird mit x=1 als kommend und mit x=0 als gehend in den Diagnosepuffer eingetragen. Im Diagnosepuffer erscheint neben der Zeit des Eintrags auch der entsprechende Diagnosetext, der in der Spalte "Bedeutung" angegeben ist.

Belegung des Diagnosedatensatzes DS0 in der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Tabelle 3-3 Belegung des Diagnosedatensatzes DS0

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Zeitüberwachung angesprochen • EEPROM-Inhalt ungültig; Baugruppe läuft an ohne zu regeln und wartet auf erneute Parametrierung durch die CPU • EPROM-Fehler • ADU/DAU-Fehler • Analogeingang Hardwarefehler	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Externe Hilfsspannung fehlt • Parametrierung fehlerhaft • Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA) • Analogeingang Messbereichsunterschreitung • Analogeingang Messbereichsüberschreitung • Analogausgang Drahtbruch • Analogausgang Kurzschluss	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe DS 1, ab Byte 7	8:x:03
	4	Externe Hilfsspannung fehlt	24-V-Versorgung der FM 355 ausgefallen	8:x:04
	6	EEPROM-Inhalt ungültig	Ausfall der Versorgungsspannung während eines Schreibvorgangs ins EEPROM. Die Baugruppe läuft mit Defaultparametern an.	8:x:03
	7	Parametrierung fehlerhaft	Die Baugruppe kann einen Parameter nicht verwerten. Grund: Parameter unbekannt oder unzulässige Kombination von Parametern. Siehe Menü **Zielsystem > Parametrierfehleranzeige**	8:x:07
1	0 ... 3	Baugruppenklasse	Ist immer mit 8 belegt	–
	4	Kanalspezifische Diagnose	Wird gesetzt, wenn die Baugruppe zusätzliche Kanalinformation liefern kann und ein Kanalfehler vorhanden ist (siehe DS 1 Byte 7 bis 12)	–
2	3	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	8:x:33

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
3	2	EPROM-Fehler	Baugruppe defekt	8:x:42
	4	ADU/DAU-Fehler	Baugruppe defekt	8:x:44

Belegung des Diagnosedatensatzes DS1

Der Diagnosedatensatz DS1 besteht aus 16 Bytes. Die ersten 4 Bytes sind identisch mit dem Diagnosedatensatz DS0. Die folgende Tabelle zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Tabelle 3-4 Belegung der Bytes 4 bis 12 des Diagnosedatensatzes DS1

Byte	Bit	Bedeutung	Anmerkung		Ereignisnr.
4	0 ... 7	Kanaltyp	Ist immer mit 75H belegt		–
5	0 ... 7	Länge der Diagnoseinformation	Ist immer mit 8 belegt		–
6	0 ... 7	Kanalanzahl	Ist immer mit 5 belegt (4 Regler + 1 Referenzkanal)		–
7	0 ... 7	Kanalfehlervektor	Jedem Kanal ist ein Bit zugeordnet		–
8	0	Analogeingang Hardwarefehler	Kanalspezifische Diagnose Kanal 1		8:x:B0
	1	unbenutzt			8:x:B1
	2	Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA)			8:x:B2
	3	unbenutzt			8:x:B3
	4	Analogeingang Messbereichsunterschreitung			8:x:B4
	5	Analogeingang Messbereichsüberschreitung			8:x:B5
	6	Analogausgang Drahtbruch		Nur bei Stromausgang des K-Reglers	8:x:B6
	7	Analogausgang Kurzschluss		Nur bei Spannungsausgang des K-Reglers	8:x:B7
9	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 2		siehe oben
10	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 3		siehe oben
11	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 4		siehe oben
12	0 ... 5	siehe Byte 8	Diagnose für Referenzkanal		siehe oben

Weitere Informationen

Um weitere Informationen zum Parameter "RET_VALU" zu erhalten, siehe Liste der RET_VALU-Meldungen (Seite 1049) und im Bereich "Erweiterte Anweisungen" im Hilfesystem.

Messumformerstörung

Störungen bei Messumformern

Folgende Messumformerstörungen können von der Reglerbaugruppe erkannt werden:

- Messbereichsunterschreitung
- Messbereichsüberschreitung
- Drahtbruch (nicht bei allen Messbereichen)

Beim Auftreten einer dieser Störungen werden im Diagnosedatensatz DS0 das Sammelfehlerbit "Fehler extern" und im Diagnosedatensatz DS1 kanalspezifische Fehlerbits gesetzt (siehe die Tabellen im vorhergehenden Abschnitt. Mit Verschwinden dieser Störungen werden die entsprechenden Bits wieder zurückgesetzt.

Nachfolgende Tabelle zeigt, bei welchen Grenzen in den einzelnen Messbereichen die Fehlerbits gesetzt bzw. rückgesetzt werden:

Messbereich	Fehlerbit Messbereichsunterschreitung bei ...	Fehlerbit Messbereichsüberschreitung bei ...	Fehlerbit Drahtbruchanzeige
	DS1: Byte 10 bis 26, Bit 4	DS1: Byte 10 bis 26, Bit 5	DS1: Byte 10 bis 26, Bit 2
0 bis 20 mA	< –3,5 mA	> 23,5 mA	–
4 bis 20 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA	> 22,8 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA
0 bis 10 V	< –1,175 V	> 11,75 V	–
Pt 100 (–200 bis 850 °C) (–328 bis 1562 °F)	< 30,82 mV	> 650,46 mV	–
Pt 100 (–200 bis 556 °C) (–328 bis 1032 °F)	< 30,82 mV	> 499,06 mV	–
Pt 100 (–200 bis 130 °C) (–328 bis 264 °F)	< 30,82 mV	> 254,12 mV	–
Thermoelement Typ B	< 0 mV	> 13,81 mV	–
Thermoelement Typ J	< –8,1 mV	> 69,54 mV	–
Thermoelement Typ K	< –6,45 mV	> 54,88 mV	–
Thermoelement Typ R	< –0,23 mV	> 21,11 mV	–
Thermoelement Typ S	< –0,24 mV	> 18,7 mV	–
Freies Thermoelement	< unterer Eingangswert des Polygonzugs	> oberer Eingangswert des Polygonzugs	–

Liste der RET_VALU-Meldungen

RET_VALU-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
7000	28672	-32624	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.
7001	28673	-32624	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.
7002	28674	-32624	Zwischenaufruf (REQ irrelevant). Datenübertragung bereits aktiv; BUSY hat den Wert 1.
8090	32912	-32624	Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.
80A0	32928	-32608	Negative Quittung beim Lesen von Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2	32930	-32606	Protokollfehler bei Layer 2
80A3	32931	-32605	Protokollfehler bei User-Interface / User
80A4	32932	-32604	Kommunikation am K-Bus gestört
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4	32964	-32572	Kommunikationsfehler
80C5	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)
80ff	33023	-32513	Falsche Indexangabe beim Baustein FMT_PAR
Die Fehler 80A2..80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein. Meldungen der Form 7xxx zeigen temporäre Betriebszustände der Kommunikation an.			

3.5.5.6 Beispielprojekte zur FM 355

3.5.6 FM 355-2 einsetzen

3.5.6.1 FM 355-2 konfigurieren

Hardware konfigurieren

Vorgehensweise

Konfigurieren setzt voraus, dass Sie ein Projekt angelegt haben, in dem Sie die Parametrierung speichern können. Nachfolgend sind nur die wichtigsten Schritte erläutert.

1. Starten Sie das TIA-Portal und öffnen Sie die Gerätekonfiguration.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie eine Reglerbaugruppe aus dem Baugruppenkatalog aus.
4. Ziehen Sie die Reglerbaugruppe in die entsprechende Stelle auf dem Baugruppenträger.
5. Notieren Sie aus der Konfigurationstabelle die Eingangsadresse der Baugruppe, z. B. 272. Der Wert, den Sie auslesen, wird im Dezimalformat angezeigt.

3.5.6.2 FM 355-2 parametrieren

Baugruppenparameter

Vorgehensweise

Beim Parametrieren stellen Sie die Baugruppenparameter ein.

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung". Wählen Sie im Inspektorfenster die Registerkarte "Eigenschaften".
   Ergebnis: Die Eigenschaften werden im Inspektorfenster angezeigt.
2. Parametrieren Sie die Grundparameter der Baugruppe und die übrigen Parameter.
3. Parametrieren Sie in der Projektieransicht die weiteren Bausteinparameter.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".
5. Übersetzen Sie das Projekt mit "Bearbeiten > Übersetzen".
6. Laden Sie die Parametrierdaten im STOP-Zustand zur CPU mit "Online > Laden in Gerät..."
   Ergebnis: Die Daten befinden sich im Speicher der CPU und werden von dort direkt zur Baugruppe übertragen.
7. Führen Sie einen CPU-Anlauf durch.

Analogeingangsparameter

Auswertung des Analogeingangs

Vorgehensweise

Zum Auswählen eines Sensortyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Auswertung des Analogeingangs".
3. Legen Sie die folgenden Parameter fest:
   - Sensortyp Analogeingang
     Für verschiedene Sensortypen wird die Option "Drahtbruchüberwachung" freigeschaltet. Je nach Anforderung kann die Option aktiviert werden.
   - Vergleichsstellentemperatur
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Filterfunktionen

Vorgehensweise

Zum Festlegen der Filterfunktionen gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Filterfunktionen".
3. Legen Sie die folgenden Parameter fest:
   - Zum Filtern nach einer Zeitkonstante aktivieren Sie die Option "Signalfilter".
     Ist die Option "Signalfilter" ausgewählt, kann die Filterzeitkonstante gesetzt werden.
   - Filterzeitkonstante
     Kann nur gesetzt werden, wenn die Option "Signalfilter" aktiviert ist.
   - Zum Linearisieren des Gebersignals aktivieren Sie die Option "Quadratwurzelfilter".
   - Zum Linearisieren der Gebersignalkennlinie aktivieren Sie die Option "Polygonfilter"
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalfilter

Filter 1. Ordnung, dessen Zeitverhalten durch die Zeitkonstante festgelegt wird.

Quadratwurzelfilter

Dient zur Linearisierung von Gebersignalen, bei denen der Istwert als physikalische Größe vorliegt, die in quadratischem Zusammenhang mit der gemessenen Prozessgröße steht.

Polygonfilter

Dient zur Linearisierung von Geberkennlinien.

Die 13 wählbaren Stützpunkte sind wählbar in:

- mA bei Stromeingang
- mV bei Spannungseingang

Normierung

Vorgehensweise

Zum Wandeln des Eingangssignals in eine physikalische Einheit gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Normierung".

3. Legen Sie die folgenden Parameter fest:
   - Oberer Wert
   - Unterer Wert
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Normierung

Zur Wandlung des Eingangssignals in eine andere physikalische Einheit durch lineare Interpolation zwischen Anfangswert (unter Wert) und Endwert (oberer Wert).

Regeldifferenz

Festwert- oder Kaskadenregler realisieren

Festwert- oder Kaskadenregler

Bei den Regelungsstrukturen Festwert- oder Kaskadenregler ist der wirksame Istwert mit dem Istwert A identisch.

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Festwert- oder Kaskadenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Aufbereiten des Sollwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".

3. Setzen Sie in den verschiedenen Bereichen die entsprechenden Parameter:
   - Sollwert Quelle
     Wählen Sie die Quelle des Sollwerts.
   - Sicherheitssollwert
     Stellen Sie hier den Sicherheitssollwert ein. Legen Sie ebenfalls die Reaktion auf einen CPU-Ausfall und das Anlaufverhalten fest.
   - Sollwert Rampe
     Legen Sie die Hochlaufzeit fest.
   - Sollwertgrenze
     Wählen Sie hier die obere und die untere Sollwertgrenze aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A wählen

Zum Auswählen der Signalquelle für den Istwert A gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz ".
3. Wählen Sie im Bereich "Stellwert" die entsprechende Quelle. Zur Verfügung stehen folgende Auswahlmöglichkeiten:
   - Null
   - Aufbereiteter Analogeingang
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Dreikomponentenregler realisieren

Dreikomponentenregler

Bei der Regelungsstruktur Dreikomponentenregler wird der wirksame Istwert durch Summieren der drei Istwerte A, B und C und durch Addition eines einstellbaren Offsets gebildet. Die Istwerte B und C können zusätzlich durch Faktoren bewertet werden.

Reglertyp wählen

Zum Realisieren eines Dreikomponentenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Dreikomponentenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.
6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A, B und C wählen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Faktoren für Istwert B und C sowie Offset erfassen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Legen Sie im Bereich "Summieren" für den Istwert B und C den erforderlichen Gewichtungsfaktor fest.
4. Legen Sie den notwendigen "Offset" fest, der zu den Istwerten addiert werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Verhältnis- oder Mischungsregler realisieren

Reglertyp wählen

Zum Realisieren eines Verhältnis- oder Mischungsreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Verhältnis- oder Mischungsregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.
6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A und D wählen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge, für den Istwert D können Sie auch den Stellwert eines der Regler auswählen.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Multiplikator erfassen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Multiplizieren" den gewünschten Offset.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Beim Verhältnisregler wird der Istwert A als Regelgröße verwendet, der Istwert D als Verhältnisgröße. Der Sollwerteingang dient als Verhältnisfaktor. Er wird durch Multiplikation mit dem Istwert D und Addition eines einstellbaren Offsets zum wirksamen Sollwert aufbereitet. Wird der Istwert D abgeschaltet, dann wird zum Sollwert nur der Offset addiert.

Alarmgenerierung einstellen

Vorgehensweise

Zum Einstellen der Alarmgenerierung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Alarmgenerierung" die erforderlichen Optionen aus:
   - Wählen Sie bei "Grenzwertmelder für" aus, für welches Signal der Alarm gesetzt werden soll.
   - Unter "Alarm obere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der oberen Grenze ausgelöst wird.
   - Unter "Warnung obere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die obere Grenze ausgelöst wird.
   - Unter "Warnung untere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die untere Grenze ausgelöst wird.
   - Unter "Alarm untere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der unteren Grenze ausgelöst wird.
   - Zum Vermeiden von Flackern der Überwachungsanzeigen kann unter "Hysterese" eine Hysterese parametriert werden.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Regelalgorithmus

P-Regelung realisieren

Vorgehen

Zum Realisieren einer P-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "P" aus.
   Alle Optionen, die für eine P-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PI-Regelung realisieren

Vorgehen

Zum Realisieren einer PI-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PI" aus.
   Alle Optionen, die für eine PI-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

I-Regelung realisieren

Vorgehen

Zum Realisieren einer I-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "I" aus.
   Alle Optionen, die für eine I-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PID-Regelung realisieren

Vorgehen

Zum Realisieren einer PID-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PID" aus.
   Alle Optionen, die für eine PID-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PD-Regelung realisieren

Vorgehen

Zum Realisieren einer PD-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PD" aus.
   Alle Optionen, die für eine PD-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Fuzzy Temperaturregler realisieren

Vorgehen

Zum Realisieren einer P-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "Fuzzy Temperatur-Reglerbaugruppe" aus.
   Alle Optionen, die für eine Fuzzy Temperatur-Reglerbaugruppe parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Rauschanteil der Regeldifferenz unterdrücken

Vorgehen

Zum Unterdrücken des Rauschanteils der Regeldifferenz gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Legen Sie im Bereich "Allgemein" die Einstellung "Totzonenbreite" fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Störgröße aufschalten

Vorgehen

Zum Aufschalten einer Störgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Allgemein" aus der Liste "Störgrößenquelle" die Quelle für das Störgröße aus.
   Wählen Sie den Eintrag "Null" aus, um keine Störgröße aufzuschalten.
4. Geben Sie unter "Störgrößenfaktor" einen Wert ein, mit dem Sie die Störgröße gewichten wollen.
   Ein Wert < 0 schwächt die Störgröße ab, ein Wert > 0 verstärkt die Störgröße. Beim Wert 1 bleibt die Störgröße unverändert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

P-Anteil in Rückführung legen

Vorgehen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Wählen Sie im Bereich "P-Anteil" die Option "In Rückführung" aus.
   Ist die Option aktiviert, ist der P-Anteil in Rückführung gelegt.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

D-Anteil bei Sollwertänderungen abschwächen

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "D-Anteil" mit der Option "D-Anteil" die Quelle für den D-Anteil aus.
4. Legen Sie im Bereich "Differenzierfaktor" den Faktor für den D-Anteil fest.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerwirkung umkehren

Prinzip

Die Umkehrung des Reglers von der Zuordnung "steigende Regeldifferenz = steigende Stellgröße" auf "steigende Regeldifferenz = fallende Stellgröße" erreichen Sie durch das Einstellen eines negativen Proportionalbeiwerts (GAIN).

Vorgehen

Zum Umkehren der Reglerwirkung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Legen Sie im Bereich "Allgemein" einen Proportionalbeiwert mit umgekehrtem Vorzeichen fest.
   Ist also ein positiver Proportionalbeiwert angegeben und die Reglerwirkung soll umgekehrt werden, legen Sie einen negativen Proportionalbeiwert fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerausgang

Kontinuierlichen Regler realisieren

Ein kontinuierlicher Regler kann nur mit einer analogen Baugruppe realisiert werden.

Vorgehen

1. Externen Stellwert schalten (Seite 1063)
2. Sicherheitsstellwert einstellen (Seite 1064)
3. Nachführung schalten (Seite 1063)
4. Stellwert begrenzen (Seite 1065)

5. Zwei Stellglieder ansteuern - Splitrange (Seite 1065)
6. Ausgangssignal und Signaltyp wählen (Seite 1066)

Schrittregler ohne Stellungsrückmeldung

Ein Schrittregler ohne Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler ohne Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1063)
3. Sicherheitsstellwert einstellen (Seite 1064)
4. Impulsformer Schrittregler (Seite 1065)

Schrittregler mit Stellungsrückmeldung realisieren

Ein Schrittregler mit Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler mit Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1063)
3. Nachführung schalten (Seite 1063)
4. Sicherheitsstellwert einstellen (Seite 1064)
5. Stellwert begrenzen (Seite 1065)
6. Stellungsrückmeldung einstellen (Seite 1064)
7. Impulsformer Schrittregler (Seite 1065)

Impulsregler realisieren

Ein Impulsregler kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Impulsregler wählen
2. Externen Stellwert schalten (Seite 1063)
3. Nachführung schalten (Seite 1063)
4. Sicherheitsstellwert einstellen (Seite 1064)
5. Stellwert begrenzen (Seite 1065)

6. Analogsignal vorbereiten auf die Umwandlung in Digitalsignal
7. Analogsignal in Digitalsignal wandeln (Seite 1066)

Funktionen des Reglerausgangs parametrieren

Externen Stellwert schalten

Vorgehen

Zum Schalten der externen Stellgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Externe Stellgröße" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Nachführung schalten

Vorgehen

Zum Schalten der Nachführung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Nachführung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Wählen Sie im Bereich "Nachführung" die Quelle des Signals für die Nachführung aus. Dabei können Sie wählen aus den folgenden Optionen:
   - Null, dabei wird keine Quelle verwendet.
   - Einer der Analogeingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellungsrückmeldung einstellen

Vorgehen

Zum Einstellen der Stellungsrückmeldung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".
3. Wählen Sie im Bereich "Stellungsrückmeldungseingang" die Option "Quelle" aus, um das Signal für die Stellungsrückmeldung auszuwählen.
   – Null: keine Stellungsrückmeldung.
   – Einer der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sicherheitsstellwert einstellen

Vorgehen

Zum Einstellen des Sicherheitsstellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Sicherheitsstellwert" den Wert fest, der als Sicherheitsstellwert verwendet werden soll.
4. Wählen Sie unter "Aktivierung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   – Aktivierung vom Funktionsbaustein.
   – Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
5. Wählen Sie unter "Reaktion bei Anlauf der Baugruppe" die Option aus, die beim Anlaufen der Baugruppe durchgeführt werden soll. Dabei können Sie aus wählen aus den folgenden Optionen:
   – Regelbetrieb: versetzt die Baugruppe beim Anlaufen sofort in den Regelbetrieb.
   – Sicherheitsstellwert: zunächst wird auf den Sicherheitsstellwert geregelt.
6. Wählen Sie unter "Reaktion bei Messumformerstörung des Istwert A" die Option aus, die bei einer Messumformerstörung des Istwert A durchgeführt werden soll.
7. Wählen Sie unter "Reaktion auf Messumformerstörung eines Eingangs" die Option aus, die bei einer Messumformerstörung eines Eingangs durchgeführt werden soll.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellwert begrenzen

Vorgehen

Zum Begrenzen des Stellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Grenzwert Ausgangssignal" die die folgenden Werte fest:
   – Oberer Wert: stellt den oberen Grenzwert für das Ausgangssignal dar.
   – Unterer Wert: stellt den unteren Grenzwert für das Ausgangssignal dar.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Zwei Stellglieder ansteuern - Splitrange

Vorgehen

Zum Ansteuern von zwei Stellgliedern gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Split Range" die Option "Aktivieren" aus, um zwei Stellglieder anzusteuern.
4. Legen Sie die folgenden Werte sowohl für die Ein- und Ausgangssignale A als auch für die Ein- und Ausgangssignale B fest:
   – Unteres Eingangssignal.
   – Oberes Eingangssignal.
   – Unteres Ausgangssignal.
   – Oberes Ausgangssignal.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Impulsformer Schrittregler

Vorgehen

Zum Parametrieren des Impulsformers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".

3. Legen Sie im Bereich "Impulsformer" folgende Parameter fest:
   - Motorstellzeit.
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Wählen Sie für den oberen und den unteren Wert die passende Option aus, woher der Wert übernommen werden soll:
   - Vom Funktionsbaustein.
   - Vom Funktionsbaustein oder von einem der Digitaleingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Analogsignal in Digitalsignal wandeln

Vorgehen

Zum Umwandeln eines Analogsignals in ein Digitalsignal gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Kanal [Nr.]".
3. Legen Sie im Bereich "Impulsformer" die folgenden Parameter fest:
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Ausgangssignal und Signaltyp wählen

Vorgehen

Zum Auswählen des Ausgangssignals und des Signaltyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogausgang > Ausgang [Nr.]".
3. Wählen Sie unter der Option "Quelle" aus, welches Signal als Ausgangssignal verwendet werden soll. Zur Auswahl stehen:
   - Null: keine Quelle für das Ausgangssignal.
   - Einer der Analogeingänge.
   - Einer der Stellwerte für Regler A bzw. B.
4. Wählen Sie unter der Option "Signaltyp" den Typ aus, in dem das Ausgangssignal ausgegeben werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

3.5.6.3 FM 355-2 in Betrieb nehmen

Instanz-DB anlegen und versorgen

Instanz-DB anlegen und versorgen

1. Erstellen Sie ein Anwenderprogramm, das die Anweisung FMT_PID zyklisch aufruft.
2. Setzen Sie im Anwenderprogramm READ_OUT zyklisch TRUE.
3. Erzeugen Sie für jeden Reglerkanal einen Instanz-DB der Anweisung FMT_PID.
4. Tragen Sie bei jedem Instanz-DB die Baugruppenadresse in den Parameter MOD_ADDR ein.
   Als Baugruppenadresse der FM 355-2 verwenden Sie die Anfangsadresse, die Sie beim Konfigurieren Ihrer Hardware festgelegt haben.
5. Tragen Sie in den Parameter FMT_PID.CHANNEL die Kanalnummer des Reglerkanals (0, 1, 2 oder 3) ein.

1. Wenn Sie die Anweisungen FMT_PAR und FMT_TUN verwenden, tragen Sie in die Parameter FMT_PAR.CHANNEL und FMT_TUN.CHANNEL die Kanalnummer des Reglerkanals ein.
2. Kontrollieren Sie, dass der "Sensortyp Analogeingang" dieses Kanals aktiviert ist.
   Wenn der Analogeingang nicht bearbeitet wird, werden alle PID-Parameter auf Null gesetzt und es wird kein Stellwert berechnet.

Hinweis

Multiinstanz-DBs

Wenn Sie in Ihrem Projekt Multiinstanz-DBs verwenden, müssen Sie die FM über eine Beobachtungstabelle in Betrieb nehmen.

Regler in Betrieb nehmen

Einführung

Sie können die PID-Parameter folgendermaßen optimieren:

- Erstoptimierung
- Nachoptimierung Heizen
- Nachoptimierung Kühlen
- Manuelle Nachoptimierung

Vorgehen

Zum Optimieren eines Reglers gehen Sie wie folgt vor:

1. Bringen Sie die CPU in den Zustand RUN.
2. Bei Verwendung eines S-Reglers messen Sie die Motorstellzeit.

3. Führen Sie eine Regleroptimierung durch.
4. Beobachten Sie den Regelkreis mit dem Kurvenschreiber.

Möglichkeiten der Regleroptimierung

Erstoptimierung

Während dieser Optimierung wird der Arbeitspunkt aus dem kalten Zustand durch einen Sollwertsprung angefahren.

Mit TUN_ON = TRUE stellen Sie die Optimierbereitschaft her. Der Regler wechselt von PHASE = 0 nach PHASE = 1.

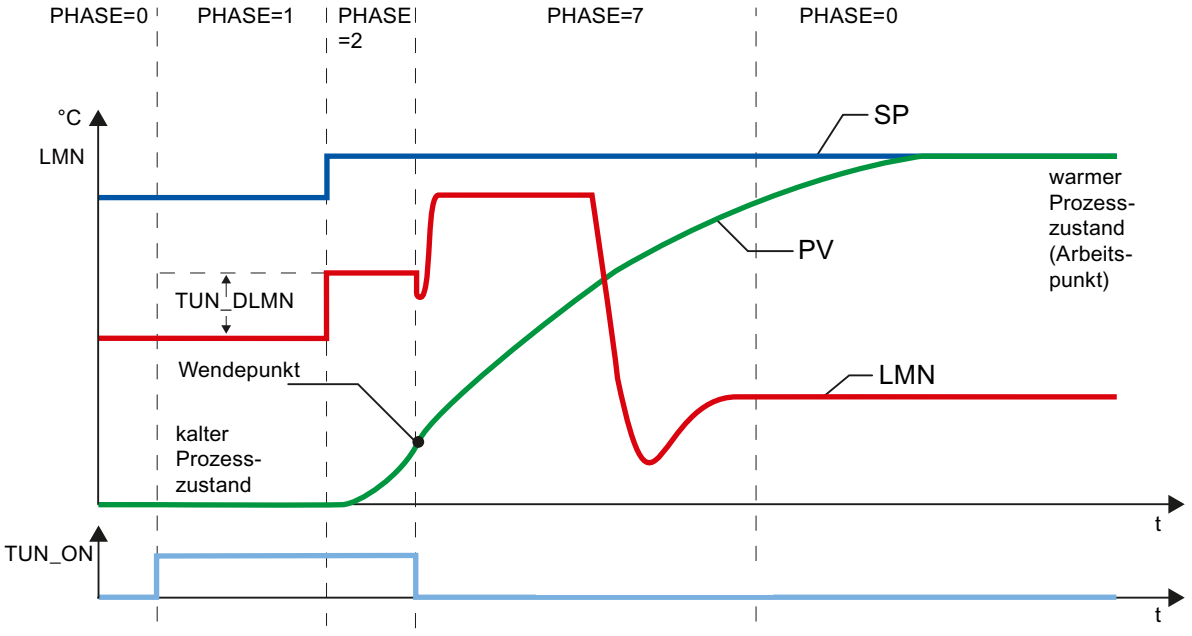

Die Optimierungs-Stellgröße (LMN0 + TUN_DLMN) wird durch Sollwertänderung aufgeschaltet (Übergang PHASE 1 -> 2). Der Sollwert wird erst bei Erreichen des Wendepunktes wirksam (erst dann wird in den Automatikbetrieb geschaltet).

Sie legen das Delta der Stellanregung (TUN_DLMN) entsprechend der zulässigen Istwertänderung in Eigenverantwortung fest. Das Vorzeichen von TUN_DLMN muss entsprechend der beabsichtigten Istwertänderung sein (Regelsinn berücksichtigen).

Der Sollwertsprung und TUN_DLMN müssen aufeinander abgestimmt sein. Ein zu großes TUN_DLMN birgt die Gefahr, dass der Wendepunkt nicht innerhalb von 75 % des Sollwertsprunges gefunden wird.

TUN_DLMN muss aber so groß sein, dass der Istwert mindestens 22 % des Sollwertsprunges erreicht. Sonst bleibt das Verfahren im Optimierbetrieb (Phase 2).

Abhilfe: Verringern des Sollwerts während der Wendepunktsuche.

Hinweis

Bei stark verzögerten Prozessen ist es ratsam, den Zielsollwert bei einer Optimierung etwas tiefer zu legen als den gewünschten Arbeitspunkt und die Statusbits sowie PV genau zu beobachten (Gefahr des Überschwingens).

Optimierung nur im linearen Bereich:

Bestimmte Regelstrecken (z. B. Zink- oder Magnesium-Schmelztiegel) durchlaufen kurz vor dem Arbeitspunkt einen nichtlinearen Bereich (Änderung des Aggregatzustands).

Durch geschickte Wahl des Sollwertsprunges kann die Optimierung auf den linearen Bereich begrenzt werden. Hat der Istwert 75 % des Sollwertsprungs (SP_INT-PV0) durchlaufen, wird die Optimierung beendet.

Parallel dazu sollte TUN_DLMN so weit reduziert werden, damit der Wendepunkt mit Sicherheit vor Erreichen von 75 % des Sollwertsprunges gefunden wird.

Nachoptimierung Heizen

Während dieser Optimierung wird der Prozess bei konstantem Sollwert durch einen Stellwertsprung angeregt.

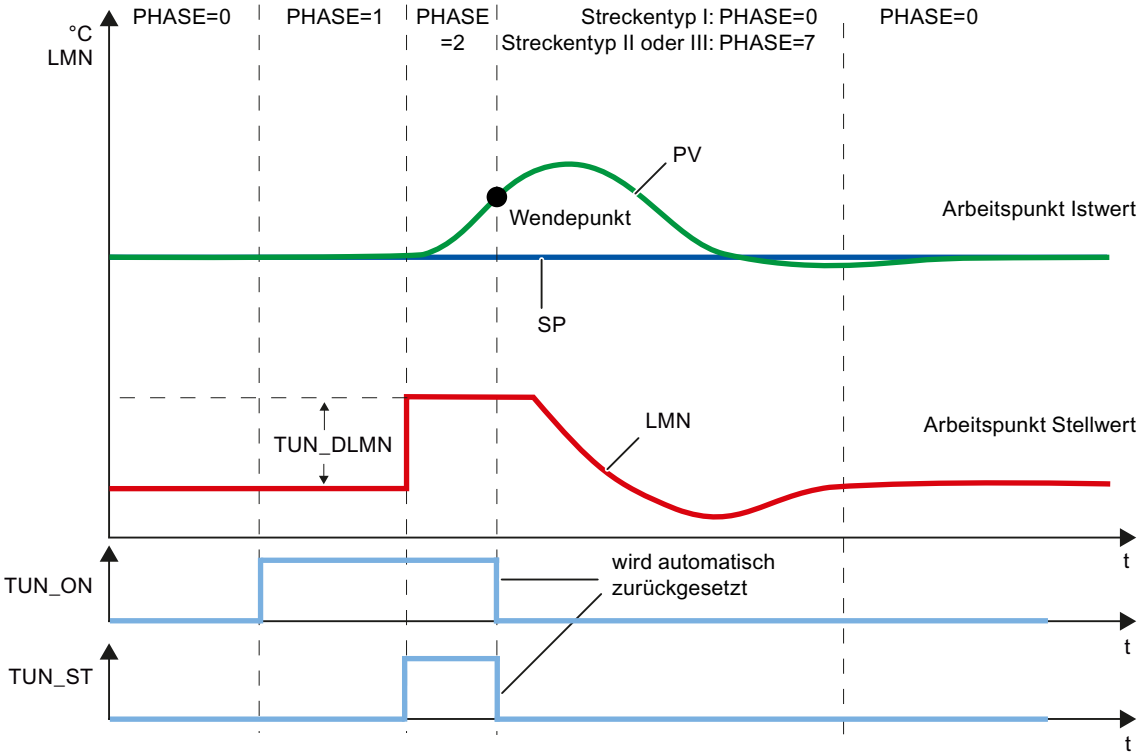

Die Optimierungs-Stellgröße (LMN0 + TUN_DLMN) wird durch Setzen des Start-Bits TUN_ST aufgeschaltet (Übergang Phase 1 -> 2). Wenn Sie den Sollwert verändern, wird der neue Sollwert erst bei Erreichen des Wendepunktes wirksam (erst dann wird in den Automatikbetrieb geschaltet).

Sie legen das Delta der Stellanregung (TUN_DLMN) entsprechend der zulässigen Istwertänderung in Eigenverantwortung fest. Das Vorzeichen von TUN_DLMN muss entsprechend der beabsichtigten Istwertänderung sein (Regelsinn berücksichtigen).

ACHTUNG
Bei Anregung über TUN_ST gibt es kein Sicherheitsabschalten bei 75 %. Die Optimierung wird bei Erreichen des Wendepunktes beendet. Bei rauschbehafteten Strecken kann jedoch der Wendepunkt deutlich überschritten werden.

Nachoptimierung Kühlen

Während dieser Optimierung wird der Prozess bei konstantem Sollwert durch einen Stellwertsprung angeregt.

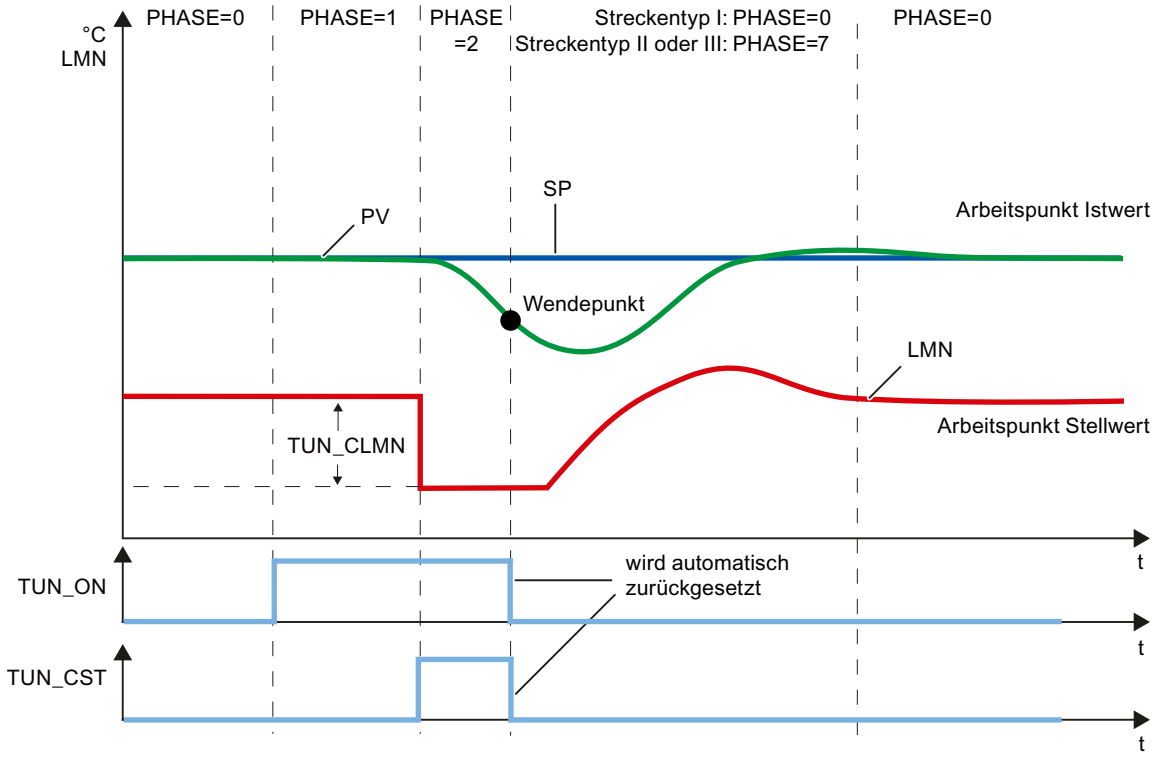

Die Optimierungs-Stellgröße (LMN0 + TUN_CLMN) wird durch Setzen des Start-Bits TUN_CST aufgeschaltet (Übergang Phase 1 -> 2). Dabei wird die vorher ermittelte Stellgröße LMN0 konstant gehalten und TUN_CLMN direkt in der Splitrange-Funktion aufgeschaltet und somit nur die Kühlleistung verstellt.TUN_CST wird sofort wieder zurückgesetzt.

Wenn Sie während der Phase 2 den Sollwert verändern, wird der neue Sollwert erst bei Erreichen des Wendepunkts wirksam (erst dann wird in den Automatikbetrieb geschaltet).

Sie legen das Delta der Stellanregung (TUN_CLMN) entsprechend der zulässigen Istwertänderung in Eigenverantwortung fest. Das Vorzeichen von TUN_CLMN muss entsprechend der beabsichtigten Istwertänderung sein (Regelsinn berücksichtigen). Beachten Sie, dass ein negatives TUN_CLMN eine Erhöhung der Kühlleistung bewirkt. Sie können aber auch mit TUN_CLMN > 0.0 einen Stellwertsprung durch Reduzierung der Kühlleistung aufschalten.

ACHTUNG
Bei Anregung über TUN_CST gibt es kein Sicherheitsabschalten bei 75 %. Die Optimierung wird bei Erreichen des Wendepunktes beendet. Bei rauschbehafteten Strecken kann jedoch der Wendepunkt deutlich überschritten werden.

Splitrange-Funktion

Bei der Optimierung wird die Steigung der Splitrange-Funktion dem Prozess zugerechnet. Sollten Sie die Steigung der Splitrange-Funktion nach der Optimierung ändern, müssen Sie auch den Reglerparameter GAIN bzw. RATIOFAC entsprechend anpassen.

Manuelle Nachoptimierung im Regelbetrieb

Um ein überschwingfreies Sollwertverhalten zu erreichen, können Sie folgende Maßnahmen ergreifen:

- Regelzone anpassen
- Führungsverhalten optimieren
- Regelparameter dämpfen
- Regelparameter ändern

Automatische Optimierung

Anforderungen an den Prozess

Einschwingverhalten

Der Prozess muss ein stabiles, verzugsbehaftetes, asymptotisches Einschwingverhalten zeigen.

Nach einem Sprung der Stellgröße muss der Istwert in einen stationären Zustand übergehen. Ausgeschlossen sind damit Prozesse, die schon ohne Regelung ein oszillatorisches Verhalten zeigen, sowie Regelstrecken ohne Ausgleich (Integrator in der Regelstrecke).

Sicherstellen eines quasi-stationären Anfangszustandes (Phase 0)

Bei niederfrequenten Schwingungen des Istwerts, z. B. auf Grund falscher Reglerparameter, ist der Regler vor dem Start der Optimierung auf Handbetrieb zu stellen und das Abklingen der Schwingung abzuwarten. Alternativ können Sie auch auf einen PI-Regler mit kleiner Proportionalverstärkung und großer Integrationszeit umschalten.

Nun müssen Sie warten, bis der stationäre Zustand erreicht ist, d. h. bis Istwert und Stellwert eingeschwungen sind. Zulässig ist auch ein asymptotisches Einschwingen bzw. langsames Driften des Istwertes (quasi-stationärer Zustand, siehe nachfolgendes Bild). Die Stellgröße muss konstant sein bzw. um einen konstanten Mittelwert schwanken.

Hinweis

Vermeiden Sie, die Stellgröße kurz vor Start der Optimierung zu verändern. Eine Änderung der Stellgröße kann auch unbeabsichtigt durch das Herstellen der Versuchsbedingungen erfolgen (z. B. Schließen einer Ofentür)! Sollte dies doch geschehen sein, müssen Sie mindestens warten, bis der Istwert wieder asymptotisch in einen stationären Zustand einschwingt. Bessere Reglerparameter erzielen Sie jedoch, wenn Sie warten bis der Einschwingvorgang vollständig abgeklungen ist.

Im folgenden Bild ist das Einschwingen in den stationären Zustand dargestellt:

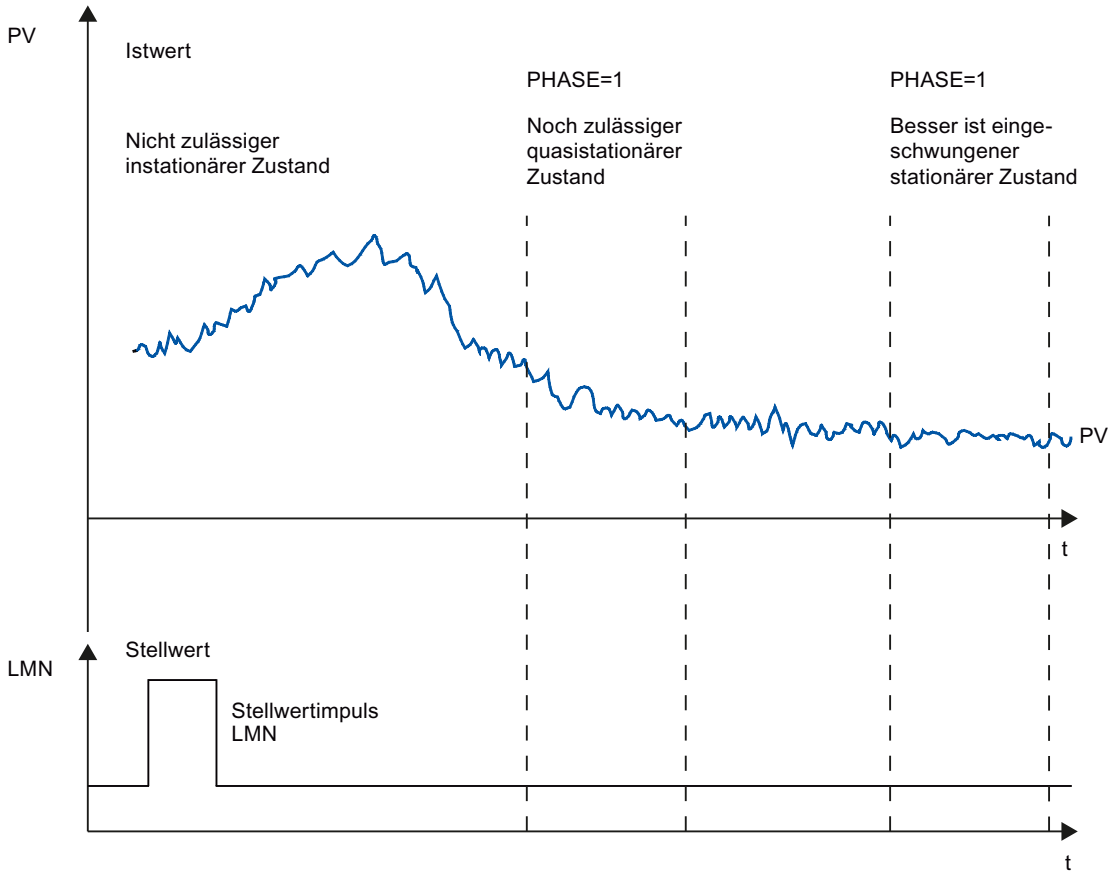

Linearität und Arbeitsbereich

Der Prozess muss ein lineares Verhalten über dem Arbeitsbereich zeigen. Ein nichtlineares Verhalten tritt z. B. bei einem Wechsel eines Aggregatszustandes auf. Die Optimierung muss in einem linearen Teil des Arbeitsbereiches stattfinden.

Das heißt, dass sowohl für die Optimierung als auch für den normalen Regelungsbetrieb nichtlineare Effekte innerhalb dieses Arbeitsbereichs vernachlässigbar klein sein müssen. Allerdings ist es möglich, beim Wechsel des Arbeitspunkts den Prozess neu zu optimieren, wenn die Optimierung wieder in einer kleinen Umgebung des neuen Arbeitspunkts durchgeführt wird und während der Optimierung die Nichtlinearität nicht durchfahren wird.

Wenn bestimmte statische Nichtlinearitäten (z. B. Ventilkennlinien) bekannt sind, ist es in jedem Fall sinnvoll, sie vorab mit einem Polygonzug zu kompensieren, um das Prozessverhalten zu linearisieren.

Störeinflüsse bei Temperaturprozessen

Störeinflüsse wie die Wärmeübertragung an benachbarten Zonen dürfen den Gesamttemperaturprozess nicht zu stark beeinflussen. Z. B. müssen bei der Optimierung von Zonen eines Extruders alle Zonen gleichzeitig aufgeheizt werden.

Ablauf der Regleroptimierung

PI/PID-Reglerparameter

Mit der Regleroptimierung in der FM 355-2 werden die PI/PID-Reglerparameter automatisch eingestellt. Es können sowohl reine Heizprozesse bzw. reine Kühlprozesse wie auch Prozesse mit zwei entgegengesetzt wirkenden Stellgliedern (z. B. Heiz- und Kühlstellglied) optimiert werden.

Phasen der Regleroptimierung

Bei der Regleroptimierung werden einzelne Phasen durchlaufen, die Sie am Parameter PHASE ablesen können.

PHASE = 0

Es läuft keine Optimierung. Die FM 355-2 arbeitet im Automatik- oder Handbetrieb.

Während PHASE = 0 sorgen Sie dafür, dass die Regelstrecke die Vorrausetzungen für eine Optimierung erfüllt.

Am Ende der Optimierung wechselt die FM 355-2 wieder in PHASE = 0.

PHASE = 1

Die FM 355-2 befindet sich in Optimierbereitschaft. PHASE = 1 darf nur gestartet werden, wenn die Vorrausetzungen für eine Optimierung erfüllt sind.

Während PHASE = 1 werden folgende Werte ermittelt:

- Istwert-Rauschen NOISE_PV
- Anfangssteigung PVDT0
- Mittelwert des Stellwerts

PHASE = 2

In Phase 2 wird bei konstantem Stellwert der Wendepunkt des Istwerts gesucht. Das Verfahren verhindert, dass durch Rauschen von PV der Wendepunkt zu früh erkannt wird:

Zu Beginn ist diese Mittelung inaktiv, d. h. es wird nur immer über einen Zyklus gemittelt. Solange das Rauschen ein bestimmtes Maß überschreitet, wird die Anzahl der Zyklen verdoppelt.

Die Periodendauer und die Amplitude des Rauschens werden ermittelt. Erst wenn der Gradient während der geschätzten Periodendauer immer kleiner als die maximale Steigung ist, wird die Wendepunktsuche abgebrochen und Phase 2 verlassen. TU und T_P_INF werden jedoch am tatsächlichen Wendepunkt berechnet.

Die Optimierung wird jedoch erst dann beendet, wenn auch die folgenden beiden Bedingungen erfüllt sind:

1. Der Istwert ist weiter als 2*NOISE_PV vom Wendepunkt entfernt.
2. Der Istwert hat den Wendepunkt um 20 % überschritten.

Hinweis

Bei Anregung über Sollwertsprung wird die Optimierung spätestens beendet, wenn der Istwert 75 % des Sollwertsprunges (SP_INT-PV0) durchlaufen hat (siehe unten).

PHASE = 3, 4, 5, 6

Die Phasen 3, 4, 5 und 6 dauern jeweils 1 Zyklus.

In Phase 3 werden die vor der Optimierung gültigen PI-/PID-Parameter gespeichert und die Prozess-Parameter berechnet.

In Phase 4 werden die neuen PI-/PID-Parameter berechnet.

In Phase 5 und 6 wird der neue Stellwert berechnet und auf die Regelstrecke gegeben.

Wenn eine Regelstrecke vom Typ I identifiziert wurde, ist der Optimierbetrieb beendet und die Optimierung befindet sich wieder in Phase 0. Der Regler startet nun mit LMN = LMN0 + 0.75*TUN_DLMN immer im Automatikbetrieb (auch wenn Sie vor dem Start der Optimierung im Handbetrieb geregelt haben).

Wenn eine Regelstrecke vom Typ II oder III identifiziert wurde, wechselt die Optimierung in Phase 7.

PHASE = 7

Bei typischen Temperaturstrecken (Streckentyp I) besteht die Gefahr, dass durch Rauschen der Wendepunkt zu früh erkannt wird. Dann wird durch die geringere Wendepunktzeit T_P_INF evtl. ein Streckentyp II oder III ermittelt.

In Phase 7 wird daher überprüft, ob der Streckentyp richtig ist. Diese Überprüfung findet im Automatikbetrieb mit den gerade neu berechneten Reglerparametern statt und endet spätestens 6*TA (Ausgleichszeit) nach dem Wendepunkt. Wird Streckentyp I erkannt, werden die Reglerparameter neu berechnet (STATUS_D = 122), ansonsten bleiben die Reglerparameter unverändert.

Die Überprüfung des Streckentyps wird abgebrochen, wenn der Istwert bei Optimierung am Arbeitspunkt den Istwert vor der Optimierung PV0 wieder erreicht.

Wird Phase 7 durch TUN_ON = FALSE abgebrochen, bleiben die bereits ermittelten Reglerparameter erhalten!

Hinweis

Es ist grundsätzlich möglich, die Heizoptimierung während des Kühlens zu starten. Die Ordnungsüberprüfung in Phase 7 erfolgt dann allerdings nicht.

Der umgekehrte Fall (Starten der Kühloptimierung während des Heizens) ist unkritisch, da eine Kühloptimierung Phase 7 grundsätzlich nicht durchläuft!

Vorzeitiger Abbruch der Optimierung

In Phase 1, 2 oder 3 können Sie durch Rücksetzen von TUN_ON = FALSE die Optimierung abbrechen ohne dass neue Parameter berechnet werden. Der Regler startet im Automatikbetrieb mit LMN = LMN0. War der Regler vor der Optimierung im Handbetrieb, wird der alte Handstellwert ausgegeben.

Wird in Phase 4, 5, 6 oder 7 mit TUN_ON = FALSE die Optimierung abgebrochen, bleiben die bis dahin ermittelten Regelparameter erhalten.

Ergebnis der Optimierung

Optimierungsstatus

Die linke Ziffer von STATUS_H und STATUS_C zeigt den Optimierungsstatus an.

STATUS_H STATUS_C	Ergebnis
0	Default bzw. keine oder noch keine neuen Reglerparameter gefunden.
10000	Geeignete Reglerparameter gefunden
2xxxx	Regel-Parameter über Schätzwerte gefunden; Überprüfen Sie das Regelverhalten bzw. schauen Sie nach der Diagnosemeldung STATUS_H/C und wiederholen Sie die Regleroptimierung.
3xxxx	Bedien-Fehler aufgetreten; Schauen Sie nach der Diagnosemeldung STATUS_H/C und wiederholen Sie die Regleroptimierung.

Die folgenden Regel-Parameter werden nach Erkennen des Wendepunkts in Phase 6 an der FM 355-2 und am Online-Instanz DB der Anweisung FMT_PID aktualisiert:

- Faktor zur Abschwächung der P-Anteils PFAC_SP = 0.8
- Reglerverstärkung GAIN
- Integrationszeit TI (begrenzt auf ≥0.5 s)
- Differenzierzeit TD (begrenzt auf ≥1.0 s)
- Differenzierfaktor D_F = 5.0
- Regelzone ein/aus CONZ_ON=TRUE/FALSE

- Regelzonenbreite CON_ZONE=250/GAIN
- P_SEL = TRUE (auch wenn vorher reiner I-Regler)

Ist RATIOFAC<>0.0, wird CON_ZONE mit 1.5 multipliziert.

Ist das von der Optimierung berechnete TD<1.0 s, wird nur ein PI-Regler ermittelt und PID_ON sowie die PID-Parameter zu 0 gesetzt.

Die Regelzone wird nur bei passendem Streckentyp (Streckentyp I und II) und PID-Regler aktiviert (CONZ_ON = TRUE).

Abhängig von PID_ON wird entweder mit PI- oder PID-Regler geregelt. Die alten Reglerparameter werden gesichert und können über UNDO_PAR wieder aktiviert werden. Ferner wird ein PI- und ein PID-Parametersatz gespeichert. Über LOAD_PID und ein entsprechendes Setzen von PID_ON kann auch später noch zwischen den optimierten PI- oder PID-Parametern gewechselt werden.

Eine eventuell vorher bestehende Strukturzerlegung (D-Anteil in Rückführung) wird beibehalten.

Hinweis

Überprüfen Sie sofort nach Ende der Regleroptimierung die korrekte Funktion der Reglerparameter.

Splitrange-Funktion

Bei der Optimierung wird die Steigung der Splitrange-Funktion dem Prozess zugerechnet. Sollten Sie die Steigung der Splitrange-Funktion nach der Optimierung ändern, müssen Sie auch den Reglerparameter GAIN entsprechend anpassen.

Fehlerbilder und Abhilfemaßnahmen

Abfangen von Bedienfehlern

Bei allen in dieser Tabelle aufgeführten Fehlern wird die Optimierung abgebrochen.

Fehler	STATUS und Maßnahmen	Kommentar
Gleichzeitiges Setzen von TUN_ON und TUN_ST bzw. TUN_CST	STATUS_H/C = 30001 • Übergang in Phase 0 • TUN_ON = FALSE	Kann beim Assistenten nicht auftreten, da in der ersten Maske TUN_ST und TUN_CST zu FALSE gesetzt werden.
Vorzeichen der Sollwertänderung passt nicht zur Istwertänderung.	Beenden der Optimierung am Wendepunkt Sicherheitsabschalten bei 75% wirkungslos	

Fehler	STATUS und Maßnahmen	Kommentar
|TUN_DLMN| oder |Effektives TUN_DLMN| < 5 % (Ende von Phase 1) Sonderfall STATUS_H=30004: Effektive Stellwertdifferenz begrenzt durch Splitrange-Grenzen und nicht durch Stellwertgrenzen	STATUS_H/C = 30002/30004 • Übergang in Phase 99 (Phase 0 bei Paralleloptimierung) • Ausgabe LMN = LMN0	Entweder haben Sie für TUN_DLMN einen zu kleinen Wert parametriert oder die Stellgröße war vor Optimierungsstart nahe an einer Stellgrenze. Falls STATUS_H=30004: Bedenken Sie z.B., dass bei LMN_A < 5% eine Heizoptimierung mit negativem TUN_DLMN nicht möglich ist (Grund: die Kühlleistung darf nicht verstellt werden). Dann muss verhindert werden, dass der Regler auf den neuen Sollwert einregelt und unnötig den stationären Arbeitspunkt verlässt (nicht möglich bei Paralleloptimierung). Sie sollten nun so vorgehen: • Sollwert zurücknehmen • TUN_ON=FALSE • TUN_DLMN korrigieren • Optimierung erneut starten
Nur bei Schrittregler ohne analoge Stellungsrückmeldung: |Effektives TUN_DLMN| < 5 % (Beginn von Phase 2)	STATUS_H = 30002 • Übergang in Phase 0 • TUN_ON = FALSE	Fehler tritt auf, wenn zwar |TUN_DLMN| >= 5 %, aber die Stellgröße vor Optimierungsstart nahe an einer Stellgrenze war.
Falls TUN_CST: |TUN_CLMN| oder |Effektives TUN_CLMN| < 5 % (Ende von Phase 1) Sonderfall STATUS_H=30004: Effektive Stellwertdifferenz begrenzt durch Splitrange-Grenzen und nicht durch Stellwertgrenzen.	STATUS_C = 30002/30004 • Übergang in Phase 0 • TUN_ON = FALSE	Entweder haben Sie für TUN_CLMN einen zu kleinen Wert parametriert oder die Stellgröße war vor Optimierungsstart nahe an einer Stellgrenze.
Falls TUN_CST: Sonderfall von "|Effektives TUN_CLMN| < 5 % (Ende von Phase 1)": TUN_CLMN <= -5%, aber LMN_LLM > -5%.	STATUS_C = 30003 • Übergang in Phase 0 • TUN_ON = FALSE	Die untere Grenze LMN_LLM ist zu hoch (z.B. 0%) und verhindert daher die Ausgabe einer Kühlleistung.
TUN_CST aber keine Heizoptimierung zuvor	STATUS_C = 30008 • Übergang in Phase 0 • TUN_ON = FALSE	
Sicherheitsbetrieb	STATUS_C = 30009 • Übergang in Phase 0 • TUN_ON = FALSE	Geht die FM 355-2 bei PHASE > 2 in den Sicherheitsbetrieb, so wird dies durch STATUS_C=30009 angezeigt.

Wendepunkt nicht erreicht (nur bei Anregung durch Sollwertsprung)

Die Optimierung wird spätestens beendet, wenn der Istwert 75 % des Sollwertsprunges (SP-INT-PV0) durchlaufen hat. Dies wird durch "Wendepunkt nicht erreicht" in STATUS_H/ STATUS_C (2xx2x) signalisiert.

Dabei gilt immer der momentan eingestellte Sollwert. Durch ein Reduzieren des Sollwertes kann also im Nachhinein ein früheres Beenden der Optimierung herbeigeführt werden.

Bei typischen Temperaturstrecken reicht der Optimierungsabbruch bei 75 % des Sollwertsprunges in der Regel aus, um ein Überschwingen zu verhindern. Vor allem bei stärker verzögerten Strecken (TU/TA > 0.1, Streckentyp III) ist jedoch Vorsicht geboten. Bei zu starker Stellanregung im Vergleich zum Sollwertsprung kann der Istwert stark überschwingen (bis zu Faktor 3).

Ist bei Strecken höherer Ordnung nach Erreichen von 75 % des Sollwertsprunges der Wendepunkt noch in weiter Ferne, gibt es ein deutliches Überschwingen. Außerdem sind die Reglerparameter zu scharf. Schwächen Sie dann die Reglerparameter ab oder wiederholen Sie den Versuch.

Im folgenden Bild ist das Überschwingen des Istwertes bei zu starker Anregung (Streckentyp III) dargestellt:

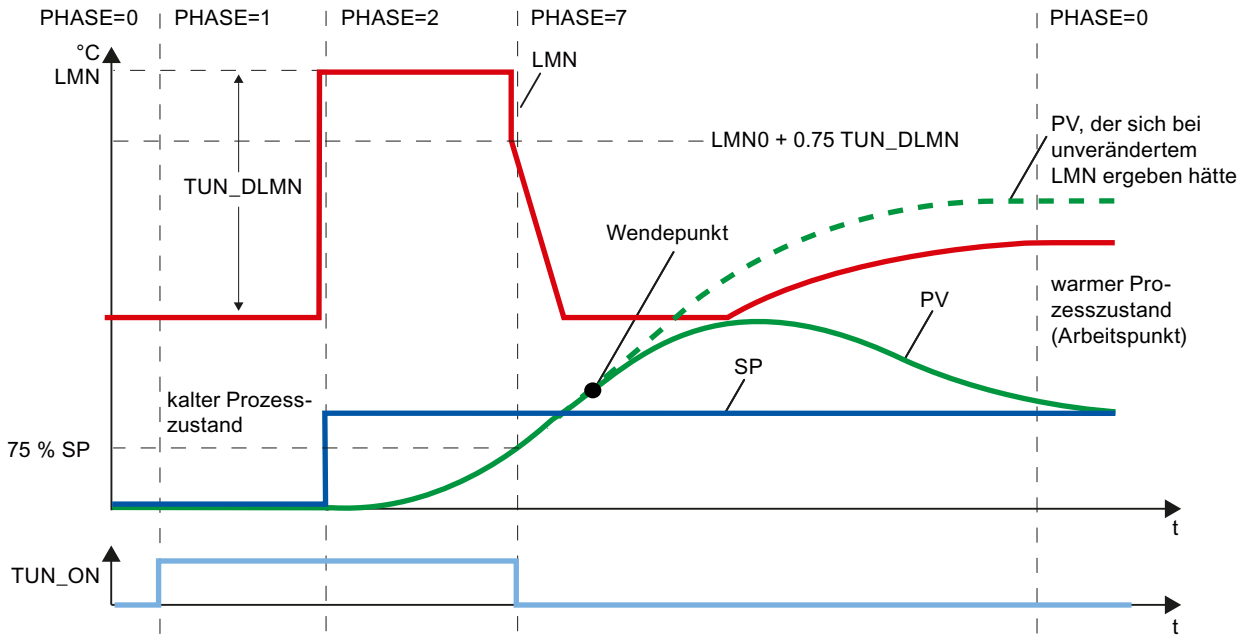

Bei typischen Temperaturstrecken ist ein Abbruch kurz vor Erreichen des Wendepunkts unkritisch bezüglich der Reglerparameter.

Falls Sie den Versuch wiederholen, reduzieren Sie TUN_DLMN oder TUN_CLMN oder erhöhen Sie den Sollwertsprung.

Prinzip: Der Optimierungs-Stellwert muss zum Sollwertsprung passen.

Schätzfehler bei Verzugszeit oder Ordnung

Die Verzugszeit (STATUS_H/STATUS_C = 2x1xx, 2x2xx oder 2x3xx) oder die Ordnung (STATUS_H/STATUS_C = 21xxx oder 22xxx) konnten nicht korrekt erfasst werden. Es wird mit einem Schätzwert weitergearbeitet, der zu nicht optimalen Reglerparametern führen kann.

Wiederholen Sie die Optimierung und achten Sie darauf, dass keine Störungen am Istwert auftreten.

Hinweis

Der Sonderfall einer reinen PT1-Strecke wird auch durch STATUS_H/STATUS_C = 2x2xx (TU < CYCLE) signalisiert. Ein Wiederholen des Versuchs ist dann nicht notwendig. Schwächen Sie die Reglerparameter ab, falls die Regelung schwingt.

Qualität der Messsignale (Mess-Rauschen, niederfrequente Störungen)

Das Optimierungsergebnis kann durch Messrauschen oder durch niederfrequente Störungen beeinträchtigt worden sein. Beachten Sie hierbei Folgendes:

- Innerhalb einer Rauschperiode sollte der Istwert mindestens zwei Mal abgetastet werden. Das Rauschmaß sollte 5% der Nutzsignaländerung nicht übersteigen.
- Hochfrequente Störungen können nicht mehr durch einen Softwarebaustein weggefiltert werden. Diese sollten bereits im Messwertaufnehmer gefiltert worden sein, um den sogenannten Aliasing-Effekt zu verhindern.
  In folgendem Bild ist der Aliasing-Effekt bei zu großer Abtastzeit dargestellt:

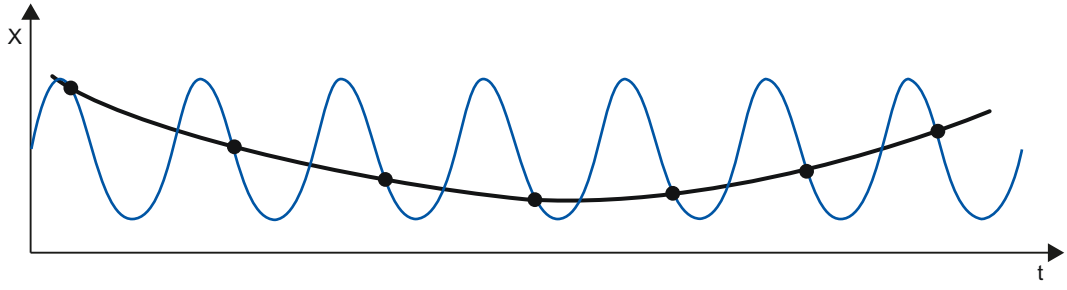

- Bei niederfrequenten Störungen kann man davon ausgehen, dass die Abtastzeit der FM 355-2 genügend klein ist. Andererseits muss dann die FM 355-2 durch ein großes Intervall der Mittelwertfilterung sich ein gleichmäßiges Messsignal erzeugen. Eine Mittelwertfilterung muss sich dabei über mindestens zwei Rauschperioden erstrecken. Intern im Baustein entstehen somit schnell größere Abtastzeiten, so dass die Genauigkeit der Optimierung beeinträchtigt wird. Eine genügend große Genauigkeit ist bei mindestens 40 Rauschperioden bis zum Wendepunkt sichergestellt. Mögliche Maßnahme bei Wiederholung des Versuchs: erhöhen von TUN_DLMN/TUN_CLMN.

Überschwingen

In folgenden Situationen kann Überschwingen auftreten:

Situation	Ursache	Abhilfe
Optimierungs-ende	• Anregung durch zu starke Stellwertänderung im Vergleich zum Sollwertsprung (siehe oben). • PI-Regler durch PID_ON = FALSE aktiviert.	• Sollwertsprung erhöhen oder Stellwertsprung verringern • Wenn der Prozess einen PID-Regler zulässt, starten Sie die Optimierung mit PID_ON = TRUE.
Optimierung in Phase 7	Es wurden zunächst sanftere Reglerparameter ermittelt (Streckentyp III), die zu einem Überschwingen in Phase 7 führen können.	-
Regelbetrieb	PI-Regler und mit PFAC_SP = 1.0 für Streckentyp I.	Wenn der Prozess einen PID-Regler zulässt, starten Sie die Optimierung mit PID_ON = TRUE.

Optimierung mit Schrittregler

Einleitung

Generell gelten die Erläuterungen zur Regleroptimierung.

Besonderheiten für Schrittregler

- Die Schrittregler auf der FM 355-2 arbeiten ohne Regelzone.
- Phase 7 wird nicht durchlaufen.
- Keine Kühloptimierung.

Hinweis

Die Motorstellzeit wird nicht durch die Optimierung ermittelt und muss vor Start der Optimierung gemessen oder vorgegeben werden.

Reglerentwurf

Die Motorstellzeit MTR_TM sollte möglichst klein gegenüber der Wendepunktzeit T_P_INF und der Verzugszeit TU sein.

Bei größeren Motorstellzeiten werden automatisch sanftere Regler entworfen.

Je größer die Stellanregung TUN_DLMN, desto größer ist der Einfluss der Motorstellzeit auf den Reglerentwurf.

PI- oder PID-Parameter

PI-Parameter werden im Vergleich zum Reglerentwurf für kontinuierlichen Regler und Impulsregler um 25% abgeschwächt. PID-Parameter werden auch ermittelt (jedoch ohne 25% Sicherheitszuschlag). Die PID-Parameter sollten nur verwendet werden, wenn die Motorstellzeit nicht zu groß ist im Vergleich zu den Streckenparametern und die Stellgliedbelastung durch den D-Anteil im Rahmen bleibt.

Schrittregler ohne analoge Stellungsrückmeldung

Bereits zu Beginn der Phase 1 wird die Ausgabe von Auf/Zu-Befehlen gestoppt (Quasi-Hand).

Der Mittelwert der Stellgröße in Phase 1 (LMN0) wird nicht berechnet.

Zu Beginn von Phase 2 wird während der Zeit MTR_TM * TUN_DLMN / 100 ein Auf-Befehl (bzw. bei negativem TUN_DLMN ein Zu-Befehl) ausgegeben.

Spricht während des Pulsens in Phase 2 ein Anschlagsignal an, wird das effektive TUN_DLMN berechnet zu: 100 * Zeit / MTR_TM.

Ist der Betrag des effektiven TUN_DLMN < 5 %, wird die Fehlermeldung STATUS_H=30002 ausgegeben und die Optimierung abgebrochen. Dieser Fehler kann nur auftreten, wenn das Stellglied wider Erwarten nahe an einer Grenze war.

Ende oder Abbruch der Optimierung

Der Regler startet mit LMN = LMN0 + TUN_DLMN.

Erstoptimierung durchführen

Voraussetzung

- Die Anforderungen an den Prozess sind erfüllt.
- Für den Kanal ist ein Instanz-DB angelegt und versorgt.
- Als Quelle für den Sollwert ist "Anweisung" parametriert.
- Die FM 355-2 befindet sich nicht im Sicherheitsbetrieb (SAFE_ON = FALSE). Eine laufende Optimierung wird abgebrochen, wenn der Sicherheitsbetrieb eingeschaltet wird.

Vorgehen

⚠ WARNUNG

Es kann Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten.

Während einer Optimierung ist der Parameter LMN_REON nicht wirksam. Auch Nachführschaltungen, die z. B. von Alarmgrenzen abgeleitet sind, greifen nicht. Hierdurch können Stell- oder Istwert ungewünschte - auch extreme - Werte annehmen.

Der Stellwert ist durch die Optimierung vorgegeben. Um die Optimierung abzubrechen, müssen Sie TUN_ON = FALSE setzen. Damit wird LMN_REON wieder wirksam.

Um bei der ersten Inbetriebnahme die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Wählen Sie in der Liste "Modus" den Eintrag "Erstoptimierung".
   Die FM 355-2 ist in Optimierbereitschaft (PHASE = 1).
4. Warten Sie, bis der stationäre Zustand erreicht ist.
5. Geben Sie im Feld "Stellwertsprung" ein, um welchen Wert der Stellwert erhöht werden soll.
6. Geben Sie im Feld "Sollwert" einen Sollwert ein. Erst durch einen anderen Sollwert wird der Stellwertsprung wirksam.
7. Klicken Sie auf das Symbol "Optimierung starten".
   Die Erstoptimierung startet. Der Status der Optimierung wird angezeigt.

Hinweis

In Phase 1 kann eine Optimierung auch durch sehr kleine Sollwertänderungen ausgelöst werden (z. B. Messrauschen eines Analogeingangs). Die Optimierung wird dann sehr schnell beendet (falsche Reglerparameter; Gefahr der Instabilität).

Nachoptimierung Heizen durchführen

Voraussetzung

- Die Anforderungen an den Prozess sind erfüllt.
- Für den Kanal ist ein Instanz-DB angelegt und versorgt.
- Als Quelle für den Sollwert ist "Anweisung" parametriert.
- Die FM 355-2 befindet sich nicht im Sicherheitsbetrieb (SAFE_ON = FALSE). Eine laufende Optimierung wird abgebrochen, wenn der Sicherheitsbetrieb eingeschaltet wird.

Vorgehen

Um am Arbeitspunkt die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Wählen Sie in der Liste "Modus" den Eintrag "Nachoptimierung Heizen".
   Die FM 355-2 ist in Optimierbereitschaft (PHASE = 1).

4. Geben Sie im Feld "Stellwertsprung" ein, um welchen Wert der Stellwert erhöht werden soll.
5. Klicken Sie auf das Symbol "Optimierung starten".
   Die Nachoptimierung startet. Der Status der Optimierung wird angezeigt.

Nachoptimierung kühlen durchführen

Einführung

Für Regelungen, die mit zwei entgegengesetzt wirkenden Stellgliedern arbeiten (Heiz- und Kühlstellglied), ermittelt die FM355-2 bei einem Stellwertsprung mit dem Kühlstellglied das Verhältnis der Prozessverstärkungen RATIOFAC (Heizverstärkung/Kühlverstärkung). Auch die Regelzonenbreite CON_ZONE wird neu berechnet. Die übrigen Reglerparameter bleiben unverändert.

Voraussetzung

- Als Quelle für den Sollwert ist "Anweisung" parametriert.
- Die FM 355-2 befindet sich nicht im Sicherheitsbetrieb (SAFE_ON = FALSE). Eine laufende Optimierung wird abgebrochen, wenn der Sicherheitsbetrieb eingeschaltet wird.
- Eine Nachoptimierung Heizen für den Kanal wurde erfolgreich abgeschlossen. Nach einem Spannungsausfall der FM 355-2 müssen Sie die Heizoptimierung wiederholen.
- Bei einer Anlage mit mehreren thermisch gekoppelten Temperaturzonen (z.B. Kunststoffmaschinen) ist für alle Zonen die Nachoptimierung Heizen erfolgreich abgeschlossen und der eingeschwungene Zustand erreicht.
- Stellgröße A muss für Heizen und Stellgröße B für Kühlen verwendet werden.
- RATIOFAC wirkt bei LMN<0.0, also muss die Splitrange-Funktion entsprechend definiert sein: A für LMN>=0 .0 und B für LMN<0.0.
- Für einen Start der Nachoptimierung Kühlen während des Heizens ist, müssen Heiz- und Kühlsignale gleichzeitig am Prozess wirksam werden können.

Vorgehen

Um am Arbeitspunkt die optimalen PID-Parameter zu ermitteln, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Wählen Sie in der Liste "Modus" den Eintrag "Nachoptimierung Kühlen".
   Die FM 355-2 ist in Optimierbereitschaft (PHASE = 1).

4. Geben Sie im Feld "Stellwertsprung" ein, um welchen Wert der Stellwert erhöht oder verringert werden soll.
5. Klicken Sie auf das Symbol "Optimierung starten".
   Die Nachoptimierung startet. Der Status der Optimierung wird angezeigt.

Ergebnis

Sobald ein Wendepunkt im Istwertverlauf gefunden wird, wechselt die FM 355-2 wieder in den Regelbetrieb (PHASE 0).

Im Gegensatz zur Heizoptimierung werden die alten Reglerparameter nicht gesichert (das alte RATIOFAC war bereits bei der Heizoptimierung gesichert worden) und Phase 4 wird nicht durchlaufen.

STATUS_D bleibt unverändert bleibt und bezieht sich immer noch auf die Heizoptimierung. Auch die PI- und PID-Parametersätze der letzten Heizoptimierung bleiben erhalten, so dass Sie diese mit LOAD_PID auch nach einer Kühloptimierung noch laden können.

Paralleloptimierung von Reglerkanälen

Nachbar-Zonen mit starker Wärmekopplung

Regeln zwei oder mehrere Regler einer Baugruppe die Temperatur z. B. auf einer Platte (z.B. zwei Heizungen und zwei gemessene Istwerte mit starker Wärmekopplung), haben Sie die Möglichkeit, eine parallele Optimierung für diese Regler durchzuführen.

Vorgehen

Um mehrere Kanäle gleichzeitig zu optimieren, gehen Sie folgendermaßen vor:

1. Parametrieren Sie die zusammengehörigen Kanäle in Zone A oder Zone B.
2. Starten Sie die Optimierung in einem der Kanäle. Die Optimierungen der zugehörigen Kanäle werden dann automatisch gestartet.

Hinweis

Bei Anregung über Sollwertsprung müssen Sie für ein gleichzeitiges Setzen der Sollwertsprünge sorgen. Sie können die Parameter PID_ON und TUN_DLMN/TUN_CLMN nur für einen Kanal vorgeben. Die Parameter für die übrigen Kanäle müssen Sie über die entsprechenden Instanz-DBs der Anweisung FMT_PID vor Start der Optimierung vorgeben.

Abbruch der Optimierung

Tritt ein Fehler in einem der Kanäle auf (Phase 0 oder Ende von Phase 1), wird die Optimierung aller Kanäle abgebrochen.

Die Optimierung kann vom Anwender durch Setzen von TUN_ON=FALSE in irgendeinem der beteiligten Kanäle zurückgesetzt werden.

Geht ein Kanal in den Sicherheitsbetrieb, wird auch die Optimierung der beteiligten Kanäle abgeschaltet.

Nur die Statusinformationen des ausgewählten Kanals werden angezeigt. Fehlerursachen der übrigen Kanäle finden Sie in den jeweiligen Instanz-DBs.

Vorteil

Die beteiligten Regler geben solange ihre jeweilige Optimierungsstellgröße aus, bis alle Regler Phase 2 verlassen haben. Damit wird vermieden, dass der Regler, der die Optimierung früher beendet, durch die Änderung seiner Stellgröße das Optimierungsergebnis der übrigen Regler verfälscht.

⚠ VORSICHT
Das Erreichen von 75% des Sollwertsprunges bei einem der beteiligten Kanäle führt nicht zu einem Beenden der Optimierung (Gefahr des Überschwingens). Der Automatikbetrieb wird erst begonnen, wenn alle beteiligten Regler ihre Optimierung beendet haben.

Nachbar-Zonen mit schwacher Wärmekopplung

Generell gilt, dass man so optimieren sollte, wie man später regelt. Werden im Produktionsbetrieb die Zonen gemeinsam parallel verfahren, so dass die Temperaturunterschiede zwischen den Zonen gleich bleiben, sollte man auch bei der Optimierung das Temperaturniveau der Nachbarzonen entsprechend mit anheben.

Die Temperaturunterschiede zu Beginn des Versuchs spielen keine Rolle, da sie durch eine entsprechende Anfangsheizung ausgeglichen werden (→Anfangssteigung = 0).

Manuelle Nachoptimierung im Regelbetrieb

Einführung

Um ein überschwingfreies Führungsverhalten zu erreichen, können Sie nachfolgend beschriebene Maßnahmen ergreifen:

Regelzone anpassen

Bei der Optimierung wird eine Regelzonenbreite CON_ZONE von der FM 355-2 ermittelt und bei passendem Streckentyp (Streckentyp I und II) und PID-Regler aktiviert: CONZ_ON = TRUE. Sie können im Regelbetrieb die Regelzone ändern oder ganz abschalten (mit CONZ_ON = FALSE).

Keine Regelzone für Streckentyp III, PI-Regler, Schrittregler

Aktivieren Sie die Regelzone bei Strecken höherer Ordnung (Streckentyp III), bringt dies in der Regel keinen Vorteil, da die Regelzone dann größer ist als der mit 100 % Stellgröße erzielbare Regelbereich. Auch ein Aktivieren der Regelzone für PI-Regler bringt keinen Vorteil.

Hinweis

Stellen Sie vor dem manuellen Einschalten der Regelzone sicher, dass die Regelzonenbreite nicht zu klein eingestellt ist. Bei zu klein eingestellter Regelzonenbreite treten nämlich Schwingungen im Verlauf der Stellgröße und des Istwerts auf.

Führungsverhalten mit PFAC_SP kontinuierlich abschwächen

Die Regelzone ist die dynamisch beste Möglichkeit, ein überschwingfreies Führungsverhalten zu erzielen. Nutzen Sie diese wo immer möglich, also bei PID-Reglern für Streckentyp I und II.

Bei PI-Reglern, Streckentyp III oder Sollwertsprüngen innerhalb der Regelzone können Sie das Führungsverhalten mit dem Parameter PFAC_SP abschwächen. Dieser Parameter legt fest, zu welchem Anteil der P-Anteil bei Sollwertsprüngen wirksam wird.

PFAC_SP wird unabhängig vom Streckentyp durch die Optimierung auf 0.8 voreingestellt; Sie können den Wert anschließend verändern. Um das Überschwingen bei Sollwertsprüngen (bei sonst korrekten Reglerparametern) auf etwa 2% zu begrenzen, sind folgende Werte für PFAC_SP ausreichend:

	Streckentyp I	Streckentyp II	Streckentyp III
	Typische Temperaturstrecke	Übergangsbereich	Temperaturstrecke höherer Ordnung
PI	0.8	0.82	0.8
PID	0.6	0.75	0.96

Passen Sie den voreingestellten Faktor (0.8) insbesondere in folgenden Fällen an:

- Streckentyp I mit PID (0.8 →0.6): Sollwertsprünge innerhalb der Regelzone führen mit PFAC_SP = 0.8 noch zu ca. 18% Überschwingen.
- Streckentyp III mit PID (0.8 →0.96): Sollwertsprünge mit PFAC_SP = 0.8 werden zu stark gedämpft. Man verschenkt deutlich Einregelzeit.

Regelparameter dämpfen

Treten im geschlossenen Regelkreis Schwingungen auf oder sind Überschwinger nach Sollwertsprüngen vorhanden, können Sie die Reglerverstärkung GAIN zurückdrehen (z. B. auf 80% des ursprünglichen Werts) und die Integrationszeit TI vergrößern (z. B. auf 150% des ursprünglichen Werts). Wird die analoge Stellgröße des kontinuierlichen Reglers mit einem Impulsformer in binäre Stellsignale gewandelt, können durch Quantisierungseffekte kleine Dauerschwingungen auftreten. Sie können diese durch Vergrößern der Totzone DEADB_W eliminieren.

PID-Parameter manuell optimieren

Voraussetzung

- Die Anforderungen an den Prozess sind erfüllt.
- Für den Kanal ist ein Instanz-DB angelegt und versorgt.
- Als Quelle für den Sollwert ist "Anweisung" parametriert.

Vorgehen

Um den Regler manuell einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Messung ein".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Geben Sie für folgende Parameter die entsprechenden Werte an:
   - P-Anteil
   - Gewichtung P-Anteil bei Sollwertsprüngen
   - I-Anteil
   - D-Anteil
   - Koeffizient DT1
   - Verhältnis Heizen / Kühlen
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "PID-Parameter an FM senden".
5. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
6. Geben Sie einen neuen Sollwert ein.
7. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb" und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
   Der Regler arbeitet mit den neuen Parametern und regelt auf den neuen Sollwert.
8. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
9. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

Hinweis

Mit FMT_PID.OP.UNDO_PAR = TRUE können Sie die alten Reglerparameter wieder einstellen.

Optimierte Parameter in Gerät laden

Einführung

Damit die Parameter auf der FM nicht beim nächsten Anlauf der CPU mit den Systemdaten überschrieben werden, können Sie folgendermaßen vorgehen:

- Sie laden nach jedem Anlauf der FM 355-2 mit LOAD_PAR=TRUE die optimierten PID-Parameter aus dem Instanz-DB des FMT_PID in die FM.
- Sie laden die optimierten PID-Parameter in den SDB.

Hinweis

Die PI-, PID- und SAVE-Parametersätze können nicht im SDB gesichert werden.

Vorgehen

Um die Parameter in den SDB zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die PID-Parameter werden in der Parametereinstellung der FM gespeichert.
2. Stoppen Sie die CPU.
3. Klicken Sie auf das Symbol "Laden in Gerät".

Reglerparameter speichern und zurückladen

Gesicherter Parametersatz

Auf der FM 355-2 steht Ihnen neben dem wirksamen Parametersatz noch ein gesicherter Parametersatz zur Verfügung. Das Speichern bzw. Zurückladen erfolgt mittels SAVE_PAR bzw. UNDO_PAR in der Struktur OP der Anweisung FMT_PID.

So können Sie z.B. vor einer manuellen Änderung mittels SAVE_PAR = TRUE die aktuellen Parameter sichern. Mittels UNDO_PAR = TRUE können die zuletzt gesicherten Reglerparameter wieder für den Regler aktiviert werden. Nach Ende der Aktion wird SAVE_PAR bzw. UNDO_PAR vom FMT_PID zurückgesetzt.

Wenn Sie wissen wollen, welche Werte sich im aktuellen Parametersatz bzw. welche sich im gesicherten Parametersatz befinden, können Sie diese über FMT_PID.READ_PAR = TRUE oder FMT_TUN.READ_OUT = TRUE auslesen:

Aktuelle Parameter in der Struktur PAR des FMT_PID:	Gesicherte Parameter in den Ausgangsparametern des FMT_TUN:
PFAC_SP	SAV_PFAC
GAIN	SAV_GAIN
TI	SAV_TI
TD	SAV_TD
D_F	SAV_D_F
CON_ZONE	SAV_CONZ

Aktuelle Parameter in der Struktur PAR des FMT_PID:	Gesicherte Parameter in den Ausgangsparametern des FMT_TUN:
RATIOFAC	SAV_RATI
CONZ_ON	SAV_CZON
P_SEL	SAV_PSEL

Hinweis

Am Ende einer Regleroptimierung werden die gesicherten Parameter durch die vor der Optimierung gültigen Werte überschrieben (Ausnahme: bei der Kühloptimierung entfällt diese Zwischensicherung).

Wechsel zwischen PI- und PID-Parametern

Nach einer Optimierung werden sowohl PI- wie auch PID-Parameter auf der FM355-2 gespeichert. Das Laden dieser Parametersätze erfolgt durch Setzen von FMT_PID.OP.LOAD_PID. Ist PID_ON in der Struktur PAR = TRUE, wird der PID-Parametersatz in die wirksamen Reglerparameter kopiert, sonst der PI-Parametersatz. Nach Ende der Aktion wird LOAD_PID vom FMT_PID zurückgesetzt. Der Wert von PID_ON bleibt natürlich erhalten, da es sich nicht um einen Bedienparameter handelt.

Wenn Sie wissen wollen, welche Werte sich im PI- bzw. PID-Parametersatz befinden, können Sie diese über FMT_TUN. READ_OUT = TRUE auslesen:

	PI-Parametersatz	PID-Parametersatz
Reglerverstärkung	PI_GAIN	PID_GAIN
Integrationszeit	PI_TI	PID_TI
Differenzierzeit	0.0	PID_TD

Hinweis

Beachten Sie Folgendes:

- Die Reglerparameter werden mit UNDO_PAR oder LOAD_PID nur dann zurückgeschrieben, wenn die Reglerverstärkung (SAV_GAIN, PID_GAIN oder PI_GAIN) ungleich Null ist:
- Damit ist der Fall berücksichtigt, dass noch keine Optimierung durchgeführt wurde bzw. PID-Parameter fehlen. War z.B. PID_ON = TRUE und PID_GAIN = 0.0, werden PID_ON auf FALSE gesetzt und die PI-Parameter kopiert.
- D_F, PFAC_SP werden durch die Optimierung voreingestellt. Sie können anschließend vom Anwender modifiziert werden. LOAD_PID verändert diese Parameter nicht.
- Die Regelzone wird bei LOAD_PID immer neu berechnet (CON_ZONE = 250/GAIN bzw. 375/GAIN falls RATIOFAC <> 0.0), auch wenn CONZ_ON = FALSE.

Stoßfreie Umschaltung bei LOAD_PID und UNDO_PAR

Die Umschaltung erfolgt so, dass die Summe aus P- und I-Anteil gleich bleiben. Beim Umschalten von PID nach PI bewirkt dies einen Stellgrößensprung, weil der D-Anteil wegfällt. Das Umschalten von PI nach PID erfolgt zunächst stoßfrei. Nach einigen Zyklen kommt es jedoch durch den D-Anteil zu einer schnellen Änderung.

Motorstellzeitmessung durchführen

Einführung

Die Motorstellzeit wird so genau wie möglich benötigt, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

Voraussetzung

- Die parametrierte Motorstellzeit muss mindestens halb so groß sein wie die reale Stellzeit. Das Stellglied wird maximal die doppelte Stellzeit in eine Richtung gefahren.
- Bei Stellgliedern mit analoger Stellungsrückmeldung muss die Stellungsrückmeldung verschaltet sein.
- Bei Stellgliedern mit Anschlagsignalen müssen die Anschlagsignale verschaltet sein.

Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
3. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

Motorstellzeit = (Obergrenze Ausgangswert – Untergrenze Ausgangswert) × Messzeit / BETRAG(Zielstellung – Startstellung).

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der

Handbetrieb der Stellsignale eingeschaltet (OP.LMNS_ON = TRUE). Das Stellglied wird nicht mehr bewegt.

Stellglieder mit Anschlagsignalen

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
3. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag. Die Zeitmessung startet, wenn das Stellglied den unteren Anschlag erreicht hat und endet, wenn das Stellglied den unteren Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der Handbetrieb der Stellsignale eingeschaltet (OP.LMNS_ON = TRUE). Das Stellglied wird nicht mehr bewegt.

Gemessene Stellzeit laden

1. Um die gemessene Motorstellzeit lokal im Projekt zu speichern, klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die gemessene Stellzeit wird in der Parametereinstellung der FM gespeichert.
2. Um die Änderungen beim nächsten CPU-Start in Kraft treten zu lassen, klicken Sie auf das Symbol "Laden in Gerät".

Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung abbrechen, wird der Handbetrieb der Stellsignale eingeschaltet (OP.LMNS_ON = TRUE) und der Handbetrieb mit Stellwertvorgabe ausgeschaltet (OP.LMN_REON = FALSE). Das Stellglied wird nicht mehr bewegt.

Mit dem Kurvenschreiber arbeiten

Beobachten

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".
3. Wählen Sie den Instanz-DB aus, den Sie beobachten wollen. Nach dem Auswählen wird der Kanal angezeigt, dem der Instanz-DB zugeordnet ist.
4. Legen Sie die Aktualisierungszeit fest, mit der der Kurvenschreiber protokollieren soll.

5. Klicken Sie auf "Messung ein", um die Messung zu starten.
6. Klicken Sie auf "Messung aus", um die Messung zu beenden.

Vorgehen

Zum Anpassen des Maßstabs gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".
3. Zum Ändern der Zeitachse:
   - Klicken Sie auf die Pfeil-Schaltfläche neben dem Datum, um einen Kalender zu öffnen. Wählen Sie aus dem Kalender ein Datum aus. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für das Start- als auch für das Enddatum.
   - Klicken Sie mit dem Mauszeiger in die Zeitskala. Der Mauszeiger wird als Pfeil links/rechts dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach links oder rechts.
     Ist die Anfangs- oder Endzeit gesperrt, ändert sich die Skalierung der Zeit.
     Ist weder Anfangs- noch Endzeit gesperrt, ändert sich der angegebene Zeitraum, die Skalierung bleibt erhalten.
4. Zum Ändern der Werteachse:
   - Klicken Sie auf das Eingabefeld für den oberen bzw. den unteren dargestellten Wert. Geben Sie dort Ihren gewünschten Wert ein. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für den oberen als auch für den unteren Wert.
   - Klicken Sie mit dem Mauszeiger in die Werteskala. Der Mauszeiger wird als Pfeil oben/unten dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach oben oder unten.
     Ist der obere oder untere Wert gesperrt, ändert sich die Skalierung der Werteachse.
     Ist weder der obere noch der untere Wert gesperrt, ändert sich der angegebene Wertebereich, die Skalierung bleibt erhalten.

3.5.6.4 FM 355-2: Diagnose durch LED-Anzeige

Fehleranzeige durch die Sammelfehler-LED

Einleitung

Leuchtet die rote Sammelfehler-LED, so ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten.

Wenn die gelbe LED blinkt, dann ist die Firmware gelöscht worden. Dieser Zustand kann nur auftreten bei einer defekten Hardware oder wenn der Ladevorgang der Firmware abgebrochen wird.

Welche Fehler werden angezeigt?

Folgende Fehler werden durch das Leuchten der Sammelfehler-LED angezeigt:

Fehlerart	Diagnosemeldung	Mögliche Fehlerursache	Behebung
Interne Fehler	Baugruppe defekt	Hardwarefehler	Baugruppe tauschen
	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	Baugruppe tauschen
	EEPROM-Inhalt ungültig	Ausfall der Versorgungsspannung beim Parametrieren	Baugruppe neu parametrieren
Externe Fehler	Falsche Parameter in Baugruppe	An die Baugruppe wurden falsche Parameter übertragen	Baugruppe neu parametrieren
	Fehler an Analogeingängen bzw. Analogausgängen	Analogeingang Hardwarefehler	Baugruppe tauschen
		Analogeingang Drahtbruch	Drahtbruch beseitigen
		Analogeingang Messbereichsunterschreitung	Messsignal überprüfen
		Analogeingang Messbereichsüberschreitung	Messsignal überprüfen
		Analogausgang Drahtbruch	Drahtbruch beseitigen
		Analogausgang Kurzschluss	Kurzschluss beseitigen
	Externe Hilfsspannung fehlt	24-V-Versorgung ausgefallen	24-V-Versorgung wiederherstellen

Diagnosealarm bei Fehlern

Alle Fehler können einen Diagnosealarm auslösen, falls Sie den Diagnosealarm in der entsprechenden Parametriermaske freigegeben haben. Aus den Diagnosedatensätzen DS0 und DS1 erfahren Sie, welche Fehler das Leuchten der LED verursacht haben. Die Belegung der Diagnosedatensätze DS0 und DS1 ist im nächsten Abschnitt beschrieben.

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, so können Sie einen Diagnosealarm parametrieren. Das zyklische Progamm der CPU wird unterbrochen und der Diagnosealarm-OB, OB 82 wird aufgerufen.

Welche Ereignisse können einen Diagnosealarm auslösen?

Die Liste zeigt Ihnen, welche Ereignisse einen Diagnosealarm auslösen können:

- Parametrierung der Baugruppe fehlt oder ist fehlerhaft
- Baugruppe defekt
- Leitungsbruch bei Analogeingaben (nur 4 bis 20 mA)
- Überlauf und Unterlauf bei Analogeingaben
- Bürdenbruch und Kurzschluss bei Analogausgaben

Freigeben des Diagnosealarms

Im Register Grundparameter sperren Sie den Diagnosealarm für die Baugruppe oder geben ihn frei.

Der Diagnosealarm ist defaultmäßig gesperrt.

Reaktionen auf ein alarmauslösendes Ereignis

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht folgendes:

- Die Diagnoseinformationen werden in die Diagnosedatensätze DS0 und DS1 auf der Baugruppe abgelegt.
- Die Sammelfehler-LED leuchtet.
- Der Diagnosealarm-OB wird aufgerufen (OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Falls kein Hardwaredefekt vorliegt, regelt die Baugruppe weiter.

Ist kein OB 82 programmiert, geht die CPU in STOP.

Diagnosedatensatz DS0 und DS1

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Bytes, der DS1 umfasst 16 Bytes, wobei die ersten vier Bytes identisch mit dem DS0 sind.

Diagnosedatensätze DS0 und DS1

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 wird automatisch beim Aufruf des Diagnose-OBs in die Startinformation übertragen. Dort werden diese vier Bytes ins Lokaldatum (Byte 8-11) des OB 82 abgelegt.

Den Diagnosedatensatz DS1 (und damit auch den Inhalt des DS0) können Sie mit dem FB 55 "FMT_DS1" von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Der FB 55 "FMT_DS1" muss im selben OB aufgerufen werden wie ein eventuell vorhandener FB FMT_PID (z.B. OB 35). Dies erreichen Sie durch folgende Maßnahme: Setzen Sie, während der OB 82 bearbeitet wird, das Bit READ_DS1. Der FMT_DS1 im OB 35 liest dann den Diagnosedatensatz DS1.

Wie erscheint der Diagnosetext im Diagnosepuffer?

Wenn Sie die Diagnosemeldung in den Diagnosepuffer eintragen wollen, müssen Sie die Anweisung "WR_REC" "Anwenderspezifische Meldung in Diagnosepuffer eintragen" in dem Anwenderprogramm aufrufen. Am Eingangsparameter EVENTN wird die Ereignisnummer der jeweiligen Diagnosemeldung angegeben. Der Alarm wird mit x=1 als kommend und mit x=0 als gehend in den Diagnosepuffer eingetragen. Im Diagnosepuffer erscheint neben der Zeit des Eintrags auch der entsprechende Diagnosetext, der in der Spalte "Bedeutung" angegeben ist.

Belegung des Diagnosedatensatzes DS0 und der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Tabelle 3-5 Belegung des Diagnosedatensatzes DS0

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Zeitüberwachung angesprochen • EPROM-Fehler • ADU/DAU-Fehler • Analogeingang Hardwarefehler	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Externe Hilfsspannung fehlt • Parametrierung fehlerhaft • Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA) • Analogeingang Messbereichsunterschreitung • Analogeingang Messbereichsüberschreitung • Analogausgang Bürdenbruch • Analogausgang Kurzschluss	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe DS 1, ab Byte 7	8:x:03
	4	Externe Hilfsspannung fehlt	24-V-Versorgung der FM 355-2 ausgefallen	8:x:04
	6	unbenutzt	-	-
	7	Parametrierung fehlerhaft	Die Baugruppe kann einen Parameter nicht verwerten. Grund: Parameter unbekannt oder unzulässige Kombination von Parametern. Siehe Menü Zielsystem > Parametrierfehleranzeige.	8:x:07

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
1	0 ... 3	Baugruppenklasse	Ist immer mit 8 belegt.	-
	4	Kanalspezifische Diagnose	Wird gesetzt, wenn die Baugruppe zusätzliche Kanalinformation liefern kann und ein Kanalfehler vorhanden ist (siehe DS 1 Byte 7 bis 12)	-
2	3	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	8:x:33
3	2	EPROM-Fehler	Baugruppe defekt	8:x:42
	4	ADU/DAU-Fehler	Baugruppe defekt	8:x:44

Diagnosedatensatz DS1

Der Diagnosedatensatz DS1 besteht aus 16 Bytes. Die ersten 4 Bytes sind identisch mit dem Diagnosedatensatz DS0. Die Tabelle unten zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Diagnosedatensatz DS 1 der FM 355-2

Tabelle 3-6 Belegung der Bytes 4 bis 12 des Diagnosedatensatzes DS1 der FM 355-2

Byte	Bit	Bedeutung	Anmerkung		Ereignisnr.
4	0 ... 7	Kanaltyp	Ist immer mit 75H belegt.		-
5	0 ... 7	Länge der Diagnoseinformation	Ist immer mit 8 belegt.		-
6	0 ... 7	Kanalanzahl	Ist immer mit 5 belegt (4 Regler + 1 Referenzkanal)		-
7	0 ... 4	Kanalfehlervektor	Jedem Kanal ist ein Bit zugeordnet (0...3; 4 für Referenzkanal).		-
8	0	Analogeingang Hardwarefehler	Kanalspezifische Diagnose Kanal 0		8:x:B0
	1	unbenutzt			8:x:B1
	2	Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA)			8:x:B2
	3	unbenutzt			8:x:B3
	4	Analogeingang Messbereichsunterschreitung			8:x:B4
	5	Analogeingang Messbereichsüberschreitung			8:x:B5
	6	Analogausgang Drahtbruch		Nur bei Stromausgang des K-Reglers	8:x:B6
	7	Analogausgang Kurzschluss		Nur bei Spannungsaus-gang des K-Keglers	8:x:B7
9	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 1		siehe oben
10	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 2		siehe oben

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
11	0 ... 7	siehe Byte 8	Kanalspezifische Diagnose Kanal 3	siehe oben
12	0 ... 5	siehe Byte 8	Diagnose für Referenzkanal	siehe oben

Weitere Informationen

Um weitere Informationen zum Parameter "RET_VALU" zu erhalten, siehe Liste der RET_VALU-Meldungen (Seite 1098) und im Bereich "Erweiterte Anweisungen" im Hilfesystem.

Messumformerstörung

Störungen bei Messumformern

Folgende Messumformerstörungen können von der FM 355-2 erkannt werden:

- Messbereichsunterschreitung
- Messbereichsüberschreitung
- Drahtbruch (nicht bei allen Messbereichen)

Beim Auftreten einer dieser Störungen werden im Diagnosedatensatz DS0 das Sammelfehlerbit "Fehler extern" und im Diagnosedatensatz DS1 kanalspezifische Fehlerbits gesetzt (siehe Tabellen oben). Mit Verschwinden dieser Störungen werden die entsprechenden Bits wieder zurückgesetzt.

Nachfolgende Tabelle zeigt, bei welchen Grenzen in den einzelnen Messbereichen die Fehlerbits gesetzt bzw. rückgesetzt werden:

Messbereich	Fehlerbit Messbereichsunter-schreitung bei ...	Fehlerbit Messbereichsüber-schreitung bei ...	Fehlerbit Drahtbruchanzeige
	DS1: Byte 10 bis 26, Bit 4	DS1: Byte 10 bis 26, Bit 5	DS1: Byte 10 bis 26, Bit 2
0 bis 20 mA	< - 3,5 mA	> 23,5 mA	-
4 bis 20 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA	> 22,8 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA
0 bis 10 V	< - 1,175 V	> 11,75 V	-
Pt 100 (-200 bis 850 °C) (-328 bis 1562 °F)	< 30,82 mV	> 650,46 mV	-
Pt 100 (-200 bis 556 °C) (-328 bis 1032 °F)	< 30,82 mV	> 499,06 mV	-
Pt 100 (-200 bis 130 °C) (-328 bis 264 °F)	< 30,82 mV	> 254,12 mV	-
Thermoelement Typ B	< 0 mV	> 13,81 mV	-
Thermoelement Typ E (-270.0 bis 1000.03 °C)	< -9,84 mV	> 76,36 mV	-
Thermoelement Typ J	< - 8,1 mV	> 69,54 mV	-

Messbereich	Fehlerbit Messbereichsunterschreitung bei ...	Fehlerbit Messbereichsüberschreitung bei ...	Fehlerbit Drahtbruchanzeige
	DS1: Byte 10 bis 26, Bit 4	DS1: Byte 10 bis 26, Bit 5	DS1: Byte 10 bis 26, Bit 2
Thermoelement Typ K	< - 6,45 mV	> 54,88 mV	-
Thermoelement Typ R	< - 0,23 mV	> 21,11 mV	-
Thermoelement Typ S	< - 0,24 mV	> 18,7 mV	-
Freies Thermoelement	< unterer Eingangswert des Polygonzugs	> oberer Eingangswert des Polygonzugs	-

Liste der RET_VALU-Meldungen

RET_VALU-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
7000	28672	-32624	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.
7001	28673	-32624	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.
7002	28674	-32624	Zwischenaufruf (REQ irrelevant). Datenübertragung bereits aktiv; BUSY hat den Wert1.
8090	32912	-32624	Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.
80A0	32928	-32608	Negative Quittung beim Lesen von Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2	32930	-32606	DP-Protokollfehler bei Layer 2
80A3	32931	-32605	DP-Protokollfehler bei User-Interface / User
80A4	32932	-32604	Kommunikation am K-Bus gestört
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4	32964	-32572	Kommunikationsfehler
80C5	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DBs zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DBs zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)
80ff	33023	-32513	Falsche Indexangabe beim Baustein FMT_PAR
Die Fehler 80A2..80A4 sowie 80Cx sind temporär, d.h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein. Meldungen der Form 7xxx zeigen temporäre Betriebszustände der Kommunikation an.			

3.5.7 FM 455 einsetzen

3.5.7.1 FM 455 konfigurieren

Hardware konfigurieren

Vorgehensweise

Konfigurieren setzt voraus, dass Sie ein Projekt angelegt haben, in dem Sie die Parametrierung speichern können. Nachfolgend sind nur die wichtigsten Schritte erläutert.

1. Starten Sie das TIA-Portal und öffnen Sie die Gerätekonfiguration.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie eine Reglerbaugruppe aus dem Baugruppenkatalog aus.
4. Ziehen Sie die Reglerbaugruppe in die entsprechende Stelle auf dem Baugruppenträger.
5. Notieren Sie aus der Konfigurationstabelle die Eingangsadresse der Baugruppe, z. B. 512. Der Wert, den Sie auslesen, wird im Dezimalformat angezeigt.

3.5.7.2 FM 455 parametrieren

Baugruppenparameter

Vorgehensweise

Beim Parametrieren stellen Sie die Baugruppenparameter ein.

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung". Wählen Sie im Inspektorfenster die Registerkarte "Eigenschaften".
   Ergebnis: Die Eigenschaften werden im Inspektorfenster angezeigt.
2. Parametrieren Sie die Grundparameter der Baugruppe und die übrigen Parameter.
3. Parametrieren Sie in der Projektieransicht die weiteren Bausteinparameter.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".
5. Übersetzen Sie das Projekt mit "Bearbeiten > Übersetzen".
6. Laden Sie die Parametrierdaten im STOP-Zustand zur CPU mit "Online > Laden in Gerät..."
   Ergebnis: Die Daten befinden sich im Speicher der CPU und werden von dort direkt zur Baugruppe übertragen.
7. Führen Sie einen CPU-Anlauf durch.

Analogeingangsparameter

Auswertung des Analogeingangs

Vorgehensweise

Zum Auswählen eines Sensortyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Auswertung des Analogeingangs".
3. Legen Sie die folgenden Parameter fest:
   - Auflösung Analogeingang
   - Sensortyp Analogeingang
     Für verschiedene Sensortypen wird die Option "Drahtbruchüberwachung" freigeschaltet. Je nach Anforderung kann die Option aktiviert werden.
   - Vergleichsstellentemperatur
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Filterfunktionen

Vorgehensweise

Zum Festlegen der Filterfunktionen gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Filterfunktionen".
3. Legen Sie die folgenden Parameter fest:
   - Zum Filtern nach einer Zeitkonstante aktivieren Sie die Option "Signalfilter".
     Ist die Option "Signalfilter" ausgewählt, kann die Filterzeitkonstante gesetzt werden.
   - Filterzeitkonstante
     Kann nur gesetzt werden, wenn die Option "Signalfilter" aktiviert ist.
   - Zum Linearisieren des Gebersignals aktivieren Sie die Option "Quadratwurzelfilter".
   - Zum Linearisieren der Gebersignalkennlinie aktivieren Sie die Option "Polygonfilter"
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalfilter

Filter 1. Ordnung, dessen Zeitverhalten durch die Zeitkonstante festgelegt wird.

Quadratwurzelfilter

Dient zur Linearisierung von Gebersignalen, bei denen der Istwert als physikalische Größe vorliegt, die in quadratischem Zusammenhang mit der gemessenen Prozessgröße steht.

Polygonfilter

Dient zur Linearisierung von Geberkennlinien.

Die 13 wählbaren Stützpunkte sind wählbar in:

- mA bei Stromeingang
- mV bei Spannungseingang

Normierung

Vorgehensweise

Zum Wandeln des Eingangssignals in eine physikalische Einheit gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogeingang > Eingang [Nr.] > Normierung".

3. Legen Sie die folgenden Parameter fest:
   - Oberer Wert
   - Unterer Wert
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Normierung

Zur Wandlung des Eingangssignals in eine andere physikalische Einheit durch lineare Interpolation zwischen Anfangswert (unter Wert) und Endwert (oberer Wert).

Regeldifferenz

Festwert- oder Kaskadenregler realisieren

Festwert- oder Kaskadenregler

Bei den Regelungsstrukturen Festwert- oder Kaskadenregler ist der wirksame Istwert mit dem Istwert A identisch.

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Festwert- oder Kaskadenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Aufbereiten des Sollwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".

3. Setzen Sie in den verschiedenen Bereichen die entsprechenden Parameter:
   - Sollwert Quelle
     Wählen Sie die Quelle des Sollwerts.
   - Sicherheitssollwert
     Stellen Sie hier den Sicherheitssollwert ein. Legen Sie ebenfalls die Reaktion auf einen CPU-Ausfall und das Anlaufverhalten fest.
   - Sollwert Rampe
     Legen Sie die Hochlaufzeit fest.
   - Sollwertgrenze
     Wählen Sie hier die obere und die untere Sollwertgrenze aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A wählen

Zum Auswählen der Signalquelle für den Istwert A gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz ".
3. Wählen Sie im Bereich "Stellwert" die entsprechende Quelle. Zur Verfügung stehen folgende Auswahlmöglichkeiten:
   - Null
   - Aufbereiteter Analogeingang
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Dreikomponentenregler realisieren

Dreikomponentenregler

Bei der Regelungsstruktur Dreikomponentenregler wird der wirksame Istwert durch Summieren der drei Istwerte A, B und C und durch Addition eines einstellbaren Offsets gebildet. Die Istwerte B und C können zusätzlich durch Faktoren bewertet werden.

Reglertyp wählen

Zum Realisieren eines Dreikomponentenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Dreikomponentenregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.
6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A, B und C wählen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Faktoren für Istwert B und C sowie Offset erfassen

Zum Festlegen der Signalquellen für die Istwerte A, B und C gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Legen Sie im Bereich "Summieren" für den Istwert B und C den erforderlichen Gewichtungsfaktor fest.
4. Legen Sie den notwendigen "Offset" fest, der zu den Istwerten addiert werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Verhältnis- oder Mischungsregler realisieren

Reglertyp wählen

Zum Realisieren eines Verhältnis- oder Mischungsreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.]".
3. Wählen Sie im Bereich "Reglertyp" die Option "Verhältnis- oder Mischungsregler" aus.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sollwert aufbereiten

Zum Einstellen, wie der Sollwert aufbereitet werden soll, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Sollwert Quelle", aus welcher Quelle der Sollwert bezogen werden soll.
4. Geben Sie im Bereich "Sicherheitssollwert" einen Wert ein, der bei einem CPU-Fehler und beim Anlaufen als Sicherheitswert dienen soll.
5. Wählen Sie das Verhalten der Reglerbaugruppe aus, das als "Reaktion auf CPU-Fehler" und beim "Anlaufverhalten" gelten soll. Dabei besteht die Auswahl aus:
   - Sollwert = letzter gültiger Sollwert: beim Eintreten des Ereignisses wird dem Sollwert der zuletzt gültige Sollwert zugewiesen. Der zuletzt gültige Sollwert bleibt also bestehen.
   - Sollwert = Sicherheitssollwert: beim Eintreten des Ereignisses wird dem Sollwert der Sicherheitssollwert zugewiesen. So kann ein abgesicherter Notbetrieb realisiert werden.
6. Legen Sie im Bereich "Sollwert Rampe" die Anstiegszeit fest, die bei einer Änderung des Sollwerts eingehalten werden soll, um ein Überschwingen zu reduzieren.
7. Legen Sie im Bereich "Sollwertgrenze" sowohl eine obere als auch eine untere Grenze für den Sollwert fest.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Signalquelle für Istwert A und D wählen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Stellwert" für jeden Istwert die zugehörige Signalquelle aus. Dabei können Sie wählen, ob Sie kein Signal ("Null") als Quelle verwenden oder einen der aufbereiteten Analogeingänge, für den Istwert D können Sie auch den Stellwert eines der Regler auswählen.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Multiplikator erfassen

Zum Festlegen der Signalquellen für die Istwerte A und D gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Multiplizieren" den gewünschten Offset.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Beim Verhältnisregler wird der Istwert A als Regelgröße verwendet, der Istwert D als Verhältnisgröße. Der Sollwerteingang dient als Verhältnisfaktor. Er wird durch Multiplikation mit dem Istwert D und Addition eines einstellbaren Offsets zum wirksamen Sollwert aufbereitet. Wird der Istwert D abgeschaltet, dann wird zum Sollwert nur der Offset addiert.

Alarmgenerierung einstellen

Vorgehensweise

Zum Einstellen der Alarmgenerierung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regeldifferenz".
3. Wählen Sie im Bereich "Alarmgenerierung" die erforderlichen Optionen aus:
   - Wählen Sie bei "Grenzwertmelder für" aus, für welches Signal der Alarm gesetzt werden soll.
   - Unter "Alarm obere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der oberen Grenze ausgelöst wird.
   - Unter "Warnung obere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die obere Grenze ausgelöst wird.
   - Unter "Warnung untere Grenze" legen Sie fest, ab welchem Wert eine Warnung für das Nähern an die untere Grenze ausgelöst wird.
   - Unter "Alarm untere Grenze" legen Sie fest, ab welchem Wert der Alarm für das Erreichen der unteren Grenze ausgelöst wird.
   - Zum Vermeiden von Flackern der Überwachungsanzeigen kann unter "Hysterese" eine Hysterese parametriert werden.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Regelalgorithmus

P-Regelung realisieren

Vorgehen

Zum Realisieren einer P-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "P" aus.
   Alle Optionen, die für eine P-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PI-Regelung realisieren

Vorgehen

Zum Realisieren einer PI-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PI" aus.
   Alle Optionen, die für eine PI-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

I-Regelung realisieren

Vorgehen

Zum Realisieren einer I-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "I" aus.
   Alle Optionen, die für eine I-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PID-Regelung realisieren

Vorgehen

Zum Realisieren einer PID-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PID" aus.
   Alle Optionen, die für eine PID-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

PD-Regelung realisieren

Vorgehen

Zum Realisieren einer PD-Regelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "PID-Regler" aus.
4. Wählen Sie im Bereich "Allgemein" unter "Betriebsart" die Option "PD" aus.
   Alle Optionen, die für eine PD-Regelung parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Fuzzy Temperaturregler realisieren

Vorgehen

Zum Realisieren einer Fuzzy-Temperaturregelung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Typ" die Option "Fuzzy Temperatur-Reglerbaugruppe" aus.
   Alle Optionen, die für eine Fuzzy Temperatur-Reglerbaugruppe parametrierbar sind, werden aktiviert, alle übrigen Optionen werden deaktiviert.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Rauschanteil der Regeldifferenz unterdrücken

Vorgehen

Zum Unterdrücken des Rauschanteils der Regeldifferenz gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Legen Sie im Bereich "Allgemein" die Einstellung "Totzonenbreite" fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Störgröße aufschalten

Vorgehen

Zum Aufschalten einer Störgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "Allgemein" aus der Liste "Störgrößenquelle" die Quelle für das Störgröße aus.
   Wählen Sie den Eintrag "Null" aus, um keine Störgröße aufzuschalten.
4. Geben Sie unter "Störgrößenfaktor" einen Wert ein, mit dem Sie die Störgröße gewichten wollen.
   Ein Wert < 0 schwächt die Störgröße ab, ein Wert > 0 verstärkt die Störgröße. Beim Wert 1 bleibt die Störgröße unverändert.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

P-Anteil in Rückführung legen

Vorgehen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regelalgorithmus".
3. Wählen Sie im Bereich "P-Anteil" die Option "In Rückführung" aus.
   Ist die Option aktiviert, ist der P-Anteil in Rückführung gelegt.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

D-Anteil bei Sollwertänderungen abschwächen

Reglertyp wählen

Zum Realisieren eines Festwert- oder Kaskadenreglers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Wählen Sie im Bereich "D-Anteil" mit der Option "D-Anteil" die Quelle für den D-Anteil aus.
4. Legen Sie im Bereich "Differenzierfaktor" den Faktor für den D-Anteil fest.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerwirkung umkehren

Prinzip

Die Umkehrung des Reglers von der Zuordnung "steigende Regeldifferenz = steigende Stellgröße" auf "steigende Regeldifferenz = fallende Stellgröße" erreichen Sie durch das Einstellen eines negativen Proportionalbeiwerts (GAIN).

Vorgehen

Zum Umkehren der Reglerwirkung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Regleralgorithmus".
3. Legen Sie im Bereich "Allgemein" einen Proportionalbeiwert mit umgekehrtem Vorzeichen fest.
   Ist also ein positiver Proportionalbeiwert angegeben und die Reglerwirkung soll umgekehrt werden, legen Sie einen negativen Proportionalbeiwert fest.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Reglerausgang

Kontinuierlichen Regler realisieren

Ein kontinuierlicher Regler kann nur mit einer analogen Baugruppe realisiert werden.

Vorgehen

1. Externen Stellwert schalten (Seite 1112)
2. Sicherheitsstellwert einstellen (Seite 1113)
3. Nachführung schalten (Seite 1112)
4. Stellwert begrenzen (Seite 1114)

5. Zwei Stellglieder ansteuern - Splitrange (Seite 1114)
6. Ausgangssignal und Signaltyp wählen (Seite 1115)

Schrittregler ohne Stellungsrückmeldung

Ein Schrittregler ohne Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler ohne Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1112)
3. Sicherheitsstellwert einstellen (Seite 1113)
4. Impulsformer Schrittregler (Seite 1114)

Schrittregler mit Stellungsrückmeldung realisieren

Ein Schrittregler mit Stellungsrückmeldung kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Schrittregler mit Stellungsrückmeldung wählen
2. Externen Stellwert schalten (Seite 1112)
3. Nachführung schalten (Seite 1112)
4. Sicherheitsstellwert einstellen (Seite 1113)
5. Stellwert begrenzen (Seite 1114)
6. Stellungsrückmeldung einstellen (Seite 1113)
7. Impulsformer Schrittregler (Seite 1114)

Impulsregler realisieren

Ein Impulsregler kann nur mit einer digitalen Baugruppe realisiert werden.

Vorgehen

1. Reglerausgang Impulsregler wählen
2. Externen Stellwert schalten (Seite 1112)
3. Nachführung schalten (Seite 1112)
4. Sicherheitsstellwert einstellen (Seite 1113)
5. Stellwert begrenzen (Seite 1114)

6. Analogsignal vorbereiten auf die Umwandlung in Digitalsignal
7. Analogsignal in Digitalsignal wandeln (Seite 1115)

Funktionen des Reglerausgangs parametrieren

Externen Stellwert schalten

Vorgehen

Zum Schalten der externen Stellgröße gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Externe Stellgröße" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Nachführung schalten

Vorgehen

Zum Schalten der Nachführung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Nachführung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
4. Wählen Sie im Bereich "Nachführung" die Quelle des Signals für die Nachführung aus. Dabei können Sie wählen aus den folgenden Optionen:
   - Null, dabei wird keine Quelle verwendet.
   - Einer der Analogeingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellungsrückmeldung einstellen

Vorgehen

Zum Einstellen der Stellungsrückmeldung gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".
3. Wählen Sie im Bereich "Stellungsrückmeldungseingang" die Option "Quelle" aus, um das Signal für die Stellungsrückmeldung auszuwählen.
   - Null: keine Stellungsrückmeldung.
   - Einer der aufbereiteten Analogeingänge.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Sicherheitsstellwert einstellen

Vorgehen

Zum Einstellen des Sicherheitsstellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Sicherheitsstellwert" den Wert fest, der als Sicherheitsstellwert verwendet werden soll.
4. Wählen Sie unter "Aktivierung" die Option aus, wodurch die externe Stellgröße aktiviert werden soll. Dabei können Sie wählen aus folgenden Optionen:
   - Aktivierung vom Funktionsbaustein.
   - Aktivierung vom Funktionsbaustein oder einem der Digitaleingänge.
5. Wählen Sie unter "Reaktion bei Anlauf der Baugruppe" die Option aus, die beim Anlaufen der Baugruppe durchgeführt werden soll. Dabei können Sie aus wählen aus den folgenden Optionen:
   - Regelbetrieb: versetzt die Baugruppe beim Anlaufen sofort in den Regelbetrieb.
   - Sicherheitsstellwert: zunächst wird auf den Sicherheitsstellwert geregelt.
6. Wählen Sie unter "Reaktion bei Messumformerstörung des Istwert A" die Option aus, die bei einer Messumformerstörung des Istwert A durchgeführt werden soll.
7. Wählen Sie unter "Reaktion auf Messumformerstörung eines Eingangs" die Option aus, die bei einer Messumformerstörung eines Eingangs durchgeführt werden soll.
8. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Stellwert begrenzen

Vorgehen

Zum Begrenzen des Stellwerts gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Legen Sie im Bereich "Grenzwert Ausgangssignal" die die folgenden Werte fest:
   – Oberer Wert: stellt den oberen Grenzwert für das Ausgangssignal dar.
   – Unterer Wert: stellt den unteren Grenzwert für das Ausgangssignal dar.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Zwei Stellglieder ansteuern - Splitrange

Vorgehen

Zum Ansteuern von zwei Stellgliedern gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Regler > Kanal [Nr.] > Reglerausgang".
3. Wählen Sie im Bereich "Split Range" die Option "Aktivieren" aus, um zwei Stellglieder anzusteuern.
4. Legen Sie die folgenden Werte sowohl für die Ein- und Ausgangssignale A als auch für die Ein- und Ausgangssignale B fest:
   – Unteres Eingangssignal.
   – Oberes Eingangssignal.
   – Unteres Ausgangssignal.
   – Oberes Ausgangssignal.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Impulsformer Schrittregler

Vorgehen

Zum Parametrieren des Impulsformers gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Ausgang [Nr.]".

3. Legen Sie im Bereich "Impulsformer" folgende Parameter fest:
   - Motorstellzeit.
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Wählen Sie für den oberen und den unteren Wert die passende Option aus, woher der Wert übernommen werden soll:
   - Vom Funktionsbaustein.
   - Vom Funktionsbaustein oder von einem der Digitaleingänge.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Analogsignal in Digitalsignal wandeln

Vorgehen

Zum Umwandeln eines Analogsignals in ein Digitalsignal gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Digitalausgang > Kanal [Nr.]".
3. Legen Sie im Bereich "Impulsformer" die folgenden Parameter fest:
   - Mindestimpulsdauer.
   - Mindestpausendauer.
4. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

Ausgangssignal und Signaltyp wählen

Vorgehen

Zum Auswählen des Ausgangssignals und des Signaltyps gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Parametereinstellung".
2. Navigieren Sie in der Projektieransicht zu "Analogausgang > Ausgang [Nr.]".
3. Wählen Sie unter der Option "Quelle" aus, welches Signal als Ausgangssignal verwendet werden soll. Zur Auswahl stehen:
   - Null: keine Quelle für das Ausgangssignal.
   - Einer der Analogeingänge.
   - Einer der Stellwerte für Regler A bzw. B.
4. Wählen Sie unter der Option "Signaltyp" den Typ aus, in dem das Ausgangssignal ausgegeben werden soll.
5. Speichern Sie die eingegebenen Parameter mit "Projekt > Speichern".

3.5.7.3 FM 455 programmieren

3.5.7.4 FM 455 in Betrieb nehmen

Instanz-DBs anlegen und versorgen

Vorgehen

Um die Voraussetzungen für die Inbetriebnahme zu erfüllen, gehen Sie folgendermaßen vor:

1. Erstellen Sie ein Anwenderprogramm, das die Anweisung PID_FM_455 zyklisch in einem Weckalarm-OB aufruft.
2. Setzen Sie im Anwenderprogramm READ_VAR zyklisch TRUE.
3. Erzeugen Sie für jeden Reglerkanal einen Einzel-Instanz-DB der Anweisung PID_FM_455.
4. Tragen Sie bei jedem Instanz-DB die Baugruppenadresse in den Parameter PID_FM_455.MOD_ADDR ein.
   Als Baugruppenadresse der FM 455 verwenden Sie die Anfangsadresse, die Sie beim Konfigurieren Ihrer Hardware festgelegt haben.
5. Tragen Sie in den Parameter PID_FM_455.CHANNEL die Nummer des Reglerkanals (1 ... 16) ein.
6. Wenn Sie die Anweisung PID_PAR_455 verwenden, tragen Sie in den Parameter PID_PAR_455.CHANNEL die Nummer des Reglerkanals ein.
7. Kontrollieren Sie, dass der "Sensortyp Analogeingang" dieses Kanals aktiviert ist.
   Wenn der Analogeingang nicht bearbeitet wird, werden alle PID-Parameter auf Null gesetzt und es wird kein Stellwert berechnet.

Hinweis

Multiinstanz-DBs

Wenn Sie in Ihrem Projekt Multiinstanz-DBs verwenden, müssen Sie die FM über eine Beobachtungstabelle in Betrieb nehmen.

Regler in Betrieb nehmen

Einführung

Sie können die PID-Parameter folgendermaßen optimieren:

- PID-/PI-Regler(Schrittregler, Kontinuierlicher Regler, Impulsregler): Manuell
- Fuzzy-Temperaturregler: Automatisch
- Schrittregler: Automatisch mit der Anweisung TUN_ES
- Kontinuierliche Regler: Automatisch mit der Anweisung TUN_EC
- Impulsregler: Automatisch mit der Anweisung TUN_EC

Vorgehen

Zum Optimieren eines Reglers gehen Sie wie folgt vor:

1. Bringen Sie die CPU in den Zustand RUN.
2. Bei Verwendung eines Schrittreglers messen Sie die Motorstellzeit.
3. Optimieren Sie die PID-Parameter mit einer der oben genannten Möglichkeiten.
4. Beobachten Sie den Regelkreis mit dem Kurvenschreiber.

Siehe auch

Motorstellzeit messen (Seite 1124)

PID-Parameter manuell optimieren

Voraussetzung

Sie haben bei der Parametrierung des Regelalgorithmus "PID-Regler" gewählt.

Vorgehen

Um einen PID-/PI-Regler manuell einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Geben Sie für den P-Anteil, den I-Anteil, den D-Anteil und die Verzögerung die entsprechenden Werte an.
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "PID-Parameter an FM senden".
5. Aktivieren Sie in der Gruppe "Aktuelle Werte" das Optionskästchen "Sollwert vorgeben".
6. Geben Sie einen neuen Sollwert ein.
7. Deaktivieren Sie ggf. das Optionskästchen "Handbetrieb" und klicken Sie in der Gruppe "Aktuelle Werte" auf das Symbol .
   Der Regler arbeitet mit den neuen Parametern und regelt auf den neuen Sollwert.
8. Kontrollieren Sie anhand der Kurvenverläufe die Qualität der PID-Parameter.
9. Wiederholen Sie die Schritte 3 bis 8, bis Sie mit dem Regelergebnis zufrieden sind.

Fuzzy-Streckenidentifikation

Anforderungen an den Prozess beim Fuzzy-Temperaturregler

Voraussetzung

Um eine optimale Regelung mit Hilfe des Fuzzy-Temperaturreglers zu erreichen, sollte der Prozess folgenden Anforderungen genügen:

- Bei Badheizungen muss die zu heizende Flüssigkeit gut durchgemischt werden.
- Für Bad-im-Bad-Regelungen müssen beide Flüssigkeiten durchgerührt werden. Gleichzeitig ist für gute Wärmeübergänge zwischen allen wärmeübertragenden Medien zu sorgen. Bei schlecht wärmeübertragenden Materialien sollten große Übertragungsflächen einen guten Wärmetransport ermöglichen.
- Bei Raumlufttemperaturregelungen muss für eine gute Durchmischung (z. B. mit Ventilatoren) gesorgt werden.
- Die Verstärkung der Regelstrecke darf den Faktor 3 nicht überschreiten.
- Die Verzugszeit darf höchstens 3 % der Ausgleichszeit sein.
- Die zu regelnde Temperatur sollte sich innerhalb der Abtastzeit des Reglers bei maximaler Stellgrößenausgabe um höchstens 1 ‰ der vorgegebenen Maximaltemperatur ändern.

Klassifikation der Regelstrecken

Eine Regelstrecke bzw. ein zu regelnder Prozess zeichnet sich durch Parameter aus wie z. B. die Heizleistung, die Heizungsmasse oder die Wärmekapazität des zu heizenden Mediums. Bezüglich des Fuzzy-Reglers unterscheidet man eine "kritische" und eine "nicht kritische" Temperaturregelstrecke wie folgt: Die Regelung wird um so kritischer,

- je größer die Heizleistung,
- je größer die Wärmekapazität der Heizung,
- je kleiner die Wärmekapazität des zu heizenden Mediums,
- je größer der Wärmeübergangswiderstand,
- je kleiner die Wärme übertragende Fläche ist.

Nach Aufschaltung eines Stellwertsprungs auf die Regelstrecke reagiert diese mit einer Sprungantwort. Anhand dieser Sprungantwort kann man die Klassifikation der Regelstrecke ebenfalls vornehmen: Die Regelung wird um so kritischer, je größer das Verhältnis t_u / t_g ist und je größer die Verstärkung der Regelstrecke ist. Mit t_u / t_g < 1/10 hat man eine unkritische Regelstrecke.

Siehe auch

Kennwerte der Regelstrecke (Seite 635)

Ablauf der Fuzzy-Streckenidentifikation

Ablauf der Parameteroptimierung beim Fuzzy-Temperaturregler

Die Parameteroptimierung basiert auf einem selbstoptimierenden Fuzzy-Regler.

Die Identifikation der Regelstrecke beginnt mit einer Beobachtungsphase, während der keine Heizleistung ausgegeben wird. Die Dauer der Beobachtungsphase ist wie folgt:

- Beobachtungsphase kontinuierlicher Regler: ca. 1 min
- Beobachtungsphase Schrittregler: ca. 1 min + 1/2 x Stellgliedlaufzeit

Diese Zeit wird genutzt, um Temperaturtrends im Heizmedium feststellen zu können. Danach wird maximale Heizleistung von 100% ausgegeben; diese ist sichtbar am Ausgangsparameter LMN im Instanz-DB der Anweisung PID_FM.

Der Bereich der ersten 4 % Temperaturerhöhung vom Sollwertbereich wird für die Identifikation genutzt, wobei jeweils bei 1 % und 4 % Temperaturerhöhung anhand der verstrichenen Zeit Informationen über das Prozessverhalten gewonnen werden.

Die Identifikation ist abgeschlossen, wenn die Heizleistung geringer als 100% wird. Dieses Verhalten kann genutzt werden, um z. B. die Identifikation über das Anwenderprogramm auszuschalten. Nach der Identifikationsphase arbeitet der Regler mit den ermittelten Parametern weiter.

Wird der Sollwertsprung nicht ausgeregelt und bleibt die Heizleistung dauerhaft auf Null, dann ist die Identifikation erfolglos beendet worden, d. h. der Regler kann die angeschlossene Regelstrecke nicht regeln.

Das Reglerverhalten bei verschiedenen Regelstrecken

Bei einer "unkritischen" Regelstrecke hat man keine Probleme, weder bei der Identifikation noch bei der Regelung.

Die Identifikation einer "zu kritischen" Regelstrecke wird abgebrochen. Die Regelung einer identifizierten "kritischen" Regelstrecke erfolgt sehr "vorsichtig" bzw. langsam.

Fuzzy-Streckenidentifikation durchführen.

Voraussetzung

Sie haben bei der Parametrierung des Regelalgorithmus "Fuzzy-Regler" gewählt.

Nach Möglichkeit müssen Sie dafür sorgen, dass sich die Regelstrecke vor der Identifikation beruhigt hat (kein Aufheiz- oder Abkühlvorgang) oder langsam monoton ändert. Das Kriterium dafür kann sein, dass die Temperaturänderung während einer Minute Zeit mit einer Geraden anzunähern ist. Bei schnelleren Prozessen ist diese Anforderung besonders aktuell.

Da zu Beginn der Identifikation vom Regler für ca. eine Minute die Stellgröße Null ausgegeben wird, muss die zu regelnde Temperatur nahe der Umgebungstemperatur sein.

Fuzzy-Streckenidentifikation starten

Um einen Fuzzy-Regler zu optimieren, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB des Kanals, den Sie optimieren wollen.
2. Klicken Sie auf das Symbol "Start".
   Wenn noch keine Online-Verbindung besteht, wird diese aufgebaut. Die aktuellen Werte für Sollwert, Istwert und Stellwert werden aufgezeichnet.
3. Geben Sie in der Gruppe "Optimierung" einen neuen Sollwert (SP_OP) ein.
   Der neue Sollwert muss folgende Bedingungen erfüllen:
   – Sollwert-Sprung > 5 %
   – Neuer Sollwert > Istwert + Obergrenze Sollwert x 0,12
4. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Fuzzy-Streckenidentifikation".

Ergebnis

Der neue Sollwert (SP_OP), SP_OP_ON = TRUE und FUZID_ON = TRUE werden an die FM gesendet. Die Fuzzy-Streckenidentifikation startet. Der Parameter IDSTATUS der Anweisung CH_DIAG liefert Informationen über den Zustand der Identifikation. Im Anzeigefeld Status sehen Sie, ob eine Fuzzy-Streckenidentifikation aktiviert ist oder läuft.

Fuzzy-Streckenidentifkation deaktivieren

Solange das Bit FUZID_ON = TRUE ist, wird mit jedem weiteren ausreichend großen Sollwertsprung die nächste Identifikation gestartet.

Um den Optimierungsmodus auszuschalten, gehen Sie folgendermaßen vor:

1. Fügen Sie FUZID_ON in einer Beobachtungstabelle.
2. Setzen Sie FUZID_ON = FALSE.

Hinweis

Ausschalten des Optimierungsmodus FUZID_ON = FALSE vor Ende der Identifikation hält die Identifikation nicht an. Die einmal gestartete Identifikation läuft unter allen Umständen weiter – außer bei negativem Sollwertsprung.

Fuzzy-Streckenidentifikation wiederholen

Bei jedem weiteren Sollwertsprung von ≥12 % wird eine erneute Identifikation durchgeführt, es sei denn, die Identifikation wird mit FUZID_ON = FALSE wieder ausgeschaltet.

Es besteht auch die Möglichkeit, die Identifikation erneut zu starten durch Verkleinern und anschließendes Vergrößern des Sollwertes um 12> %.

Fuzzy-Streckenidentifikation abbrechen

Die Identifikation kann in folgenden Fällen abgebrochen werden:

- vom Regler bei der Identifikation einer "kritischen" Regelstrecke. Nach dem Abbruch befindet sich der Regler im Fehler-Zustand. Dieser Zustand wird dadurch angezeigt, dass die Stellgröße vom Regler auf Dauer zurückgesetzt wird. Dies ändert sich auch nicht beim Ausschalten der Optimierung. Das Starten einer neuen Identifikation löscht den Fehler-Zustand.
  Informationen über den Zustand der Identifikation erhalten Sie durch den Parameter IDSTATUS des FB CH_DIAG. Mehr dazu am Ende dieses Kapitels unter "Reglerzustandsinformationen".
- vom Bediener durch die Erzeugung eines negativen Sollwertsprungs, wobei der Sollwert die Bedingung 2 (siehe oben, "Wie starten Sie die Identifikation") unterschreiten muss.

Um die Optimierung abzubrechen, gehen Sie folgendermaßen vor:

1. Geben Sie einen neuen Sollwert ein, der unter dem aktuellen Istwert liegt und die gleichen Bedingungen erfüllt, wie zum Start der Fuzzy-Streckenidentifikation.
2. Klicken Sie in der Gruppe "Optimierung" auf das Symbol "Fuzzy-Streckenidentifikation".

Optimierte Parameter in Gerät laden

Einführung

Damit die Parameter auf der FM nicht beim nächsten Anlauf der CPU mit den Systemdaten überschrieben werden, können Sie folgendermaßen vorgehen:

- Sie laden nach jedem Anlauf der FM mit LOAD_PAR=TRUE die optimierten PID-Parameter aus dem Instanz-DB des PID_FM in die FM.
- Sie laden die optimierten PID-Parameter in den SDB.

Hinweis

Die PI-, PID- und SAVE-Parametersätze können nicht im SDB gesichert werden.

Vorgehen

Um die Parameter in den SDB zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die PID-Parameter werden in der Parametereinstellung der FM gespeichert.
2. Stoppen Sie die CPU.
3. Klicken Sie auf das Symbol "Laden in Gerät".

Mit dem Kurvenschreiber arbeiten

Beobachten

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".
3. Wählen Sie den Instanz-DB aus, den Sie beobachten wollen. Nach dem Auswählen wird der Kanal angezeigt, dem der Instanz-DB zugeordnet ist.
4. Legen Sie die Aktualisierungszeit fest, mit der der Kurvenschreiber protokollieren soll.
5. Klicken Sie auf "Messung ein", um die Messung zu starten.
6. Klicken Sie auf "Messung aus", um die Messung zu beenden.

Vorgehen

Zum Anpassen des Maßstabs gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Geräteansicht unter der Baugruppe die "Inbetriebnahme".
2. Navigieren Sie in der Projektieransicht zu "Kurvenschreiber".
3. Zum Ändern der Zeitachse:
   - Klicken Sie auf die Pfeil-Schaltfläche neben dem Datum, um einen Kalender zu öffnen. Wählen Sie aus dem Kalender ein Datum aus. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für das Start- als auch für das Enddatum.
   - Klicken Sie mit dem Mauszeiger in die Zeitskala. Der Mauszeiger wird als Pfeil links/rechts dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach links oder rechts.
     Ist die Anfangs- oder Endzeit gesperrt, ändert sich die Skalierung der Zeit.
     Ist weder Anfangs- noch Endzeit gesperrt, ändert sich der angegebene Zeitraum, die Skalierung bleibt erhalten.
4. Zum Ändern der Werteachse:
   - Klicken Sie auf das Eingabefeld für den oberen bzw. den unteren dargestellten Wert. Geben Sie dort Ihren gewünschten Wert ein. Zum Sperren der Auswahl klicken Sie auf das blaue Schloss. Diese Möglichkeit existiert sowohl für den oberen als auch für den unteren Wert.
   - Klicken Sie mit dem Mauszeiger in die Werteskala. Der Mauszeiger wird als Pfeil oben/unten dargestellt. Halten Sie den Mauszeiger gedrückt und ziehen ihn nach oben oder unten.
     Ist der obere oder untere Wert gesperrt, ändert sich die Skalierung der Werteachse.
     Ist weder der obere noch der untere Wert gesperrt, ändert sich der angegebene Wertebereich, die Skalierung bleibt erhalten.

Parameter über HMI verändern

Vorgehen

Um Reglerparameter der Anweisung PID_FM durch Bedienung über HMI ändern wollen, gehen Sie folgendermaßen vor:

1. Schreiben Sie die zu ändernden Parameter vom HMI aus in einen Hilfs-DB auf der CPU (siehe ①).
2. Setzen Sie PID_FM.COM_RST auf TRUE.
   Die Anweisung führt einen Initialsierungslauf durch. Die Parameter sind zwischen FM und CPU abgeglichen (siehe ②).
3. Übertragen Sie die zu ändernden Parameter aus dem Hilfs-DB in den Instanz-DB der Anweisung PID_FM (siehe ③).
4. Setzen Sie PID_FM.LOAD_PAR auf TRUE.
   Die geänderten Parameter werden zur FM übertragen(siehe ④).

Die Ablage der Parameter in einem Hilfs-DB ist notwendig, weil PID_FM nach dem Anlauf der CPU mit COM_RST = TRUE diejenigen Parameter aus der Baugruppe liest, die die CPU zuvor aus den Systemdaten zur FM übertragen hat.

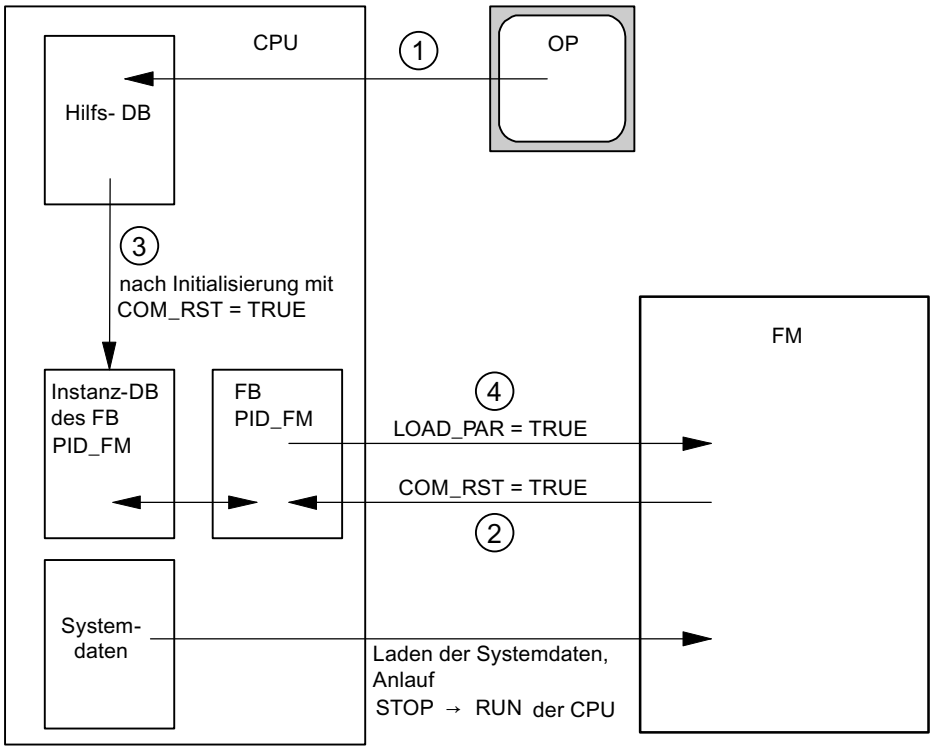

Motorstellzeit messen

Einführung

Die Motorstellzeit wird so genau wie möglich benötigt, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

Voraussetzung

- Die parametrierte Motorstellzeit muss mindestens halb so groß sein wie die reale Stellzeit. Das Stellglied wird maximal die doppelte Stellzeit in eine Richtung gefahren.
- Bei Stellgliedern mit analoger Stellungsrückmeldung muss die Stellungsrückmeldung verschaltet sein.
- Bei Stellgliedern mit Anschlagsignalen müssen die Anschlagsignale verschaltet sein.

Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
3. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

Motorstellzeit = (Obergrenze Ausgangswert – Untergrenze Ausgangswert) × Messzeit / BETRAG(Zielstellung – Startstellung).

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE). Das Stellglied wird nicht mehr bewegt.

Stellglieder mit Anschlagsignalen

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Instanz-DB, für den die Motorstellzeit gemessen werden soll.
2. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
3. Klicken Sie auf das Symbol "Stellzeitmessung starten".

Ergebnis

Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag. Die Zeitmessung startet, wenn das Stellglied den unteren Anschlag erreicht hat und endet, wenn das Stellglied den unteren Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE). Das Stellglied wird nicht mehr bewegt.

Gemessene Stellzeit laden

1. Um die gemessene Motorstellzeit lokal im Projekt zu speichern, klicken Sie auf das Symbol "Übertrage Online-Datein in ein Offline-Projekt".
   Die gemessene Stellzeit wird in der Parametereinstellung der FM gespeichert.
2. Um die Änderungen beim nächsten CPU-Start in Kraft treten zu lassen, klicken Sie auf das Symbol "Laden in Gerät".

Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung abbrechen, wird der Handbetrieb der Stellsignale eingeschaltet (LMNSOPON = TRUE) und der Handbetrieb mit Stellwertvorgabe ausgeschaltet (LMNOP_ON = FALSE). Das Stellglied wird nicht mehr bewegt.

3.5.7.5 FM 455: Diagnose durch LED-Anzeige

Fehleranzeige durch die Fehler-LED

Wo werden Fehler angezeigt?

Die rote LED INTF zeigt interne Fehler auf der Baugruppe an.

Die rote LED EXTF zeigt externe Fehler an, z. B. Fehler an den Anschlüssen.

Wenn die gelbe LED blinkt, dann ist die Firmware gelöscht worden. Dieser Zustand kann nur auftreten bei einer defekten Hardware oder wenn der Ladevorgang der Firmware abgebrochen wird.

Welche Fehler werden angezeigt?

Folgende Fehler werden durch das Leuchten der Fehler-LEDs angezeigt:

Fehlerart	Diagnosemeldung	Mögliche Fehlerursache	Behebung
Interne Fehler	Baugruppe defekt	Hardwarefehler	Baugruppe tauschen
	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	Baugruppe tauschen
	EEPROM-Inhalt ungültig	Ausfall der Versorgungsspannung beim Parametrieren	Baugruppe neu parametrieren
Externe Fehler	Falsche Parameter in Baugruppe	An die Baugruppe wurden falsche Parameter übertragen	Baugruppe neu parametrieren
	Fehler an Analogeingängen bzw. Analogausgängen	Analogeingang Hardwarefehler	Baugruppe tauschen
		Analogeingang Drahtbruch	Drahtbruch beseitigen
		Analogeingang Messbereichsunterschreitung	Messsignal überprüfen
		Analogeingang Messbereichsüberschreitung	Messsignal überprüfen
		Analogausgang Drahtbruch	Drahtbruch beseitigen
		Analogausgang Kurzschluss	Kurzschluss beseitigen
	Externe Hilfsspannung fehlt	Linker Frontstecker nicht gesteckt 24-V-Versorgung ausgefallen	Linken Frontstecker stecken 24-V-Versorgung wiederherstellen

Diagnosealarm bei Fehlern

Alle Fehler können einen Diagnosealarm auslösen, falls Sie den Diagnosealarm in der entsprechenden Parametriermaske freigegeben haben. Aus den Diagnosedatensätzen DS0 und DS1 erfahren Sie, welche Fehler das Leuchten der LED verursacht haben. Die Belegung der Diagnosedatensätze DS0 und DS1 ist im Kapitel "Auslösen von Diagnosealarmen (Seite 1126)" beschrieben.

Auslösen von Diagnosealarmen

Was ist ein Diagnosealarm?

Soll das Anwenderprogramm auf einen internen oder externen Fehler reagieren, so können Sie einen Diagnosealarm parametrieren, der das zyklische Progamm der CPU unterbricht und den Diagnosealarm-OB, OB 82, aufruft.

Welche Ereignisse können einen Diagnosealarm auslösen?

Die Liste zeigt Ihnen, welche Ereignisse einen Diagnosealarm auslösen können:

- Parametrierung der Baugruppe fehlt oder ist fehlerhaft
- Baugruppe defekt
- Leitungsbruch bei Analogeingaben (4 bis 20 mA)
- Überlauf und Unterlauf bei Analogeingaben

- Bürdenbruch und Kurzschluss bei Analogausgaben
- Drahtbruch bei Thermoelementen und Pt 100

Defaulteinstellung

Der Diagnosealarm ist defaultmäßig gesperrt.

Freigeben des Diagnosealarms

In der Maske Grundparameter sperren Sie den Diagnosealarm für die Baugruppe oder geben ihn frei.

Reaktionen auf ein alarmauslösendes Ereignis

Tritt ein Ereignis ein, das einen Diagnosealarm auslösen kann, geschieht Folgendes:

- Die Diagnoseinformationen werden in die Diagnosedatensätze DS0 und DS1 auf der Baugruppe abgelegt.
- Eine oder beide Fehler-LED leuchten.
- Der Diagnosealarm-OB wird aufgerufen (OB 82).
- Der Diagnosedatensatz DS0 wird in die Startinformation des Diagnosealarm-OB eingetragen.
- Falls kein Hardwaredefekt vorliegt, regelt die Baugruppe weiter.

Ist kein OB 82 programmiert, geht die CPU in STOP.

Diagnosedatensatz DS0 und DS1

Einleitung

Die Information, welches Ereignis einen Diagnosealarm ausgelöst hat, wird in den Diagnosedatensätzen DS0 und DS1 abgelegt. Der Diagnosedatensatz DS0 umfasst vier Bytes, der DS1 umfasst 27 Bytes, wobei die ersten vier Bytes identisch mit dem DS0 sind.

Datensatz von der Baugruppe lesen

Der Diagnosedatensatz DS0 wird automatisch beim Aufruf des Diagnose-OBs in die Startinformation übertragen. Dort werden diese vier Bytes ins Lokaldatum (Byte 8-11) des OB 82 abgelegt.

Den Diagnosedatensatz DS1 (und damit auch den Inhalt des DS0) können Sie mit der Anweisung "CH_DIAG_455" von der Baugruppe auslesen. Dies ist nur dann sinnvoll, wenn im DS0 ein Fehler in einem Kanal gemeldet wird.

Wie erscheint der Diagnosetext im Diagnosepuffer?

Wenn Sie die Diagnosemeldung in den Diagnosepuffer eintragen wollen, müssen Sie die Anweisung "WR_REC" "Anwenderspezifische Meldung in Diagnosepuffer eintragen" in dem Anwenderprogramm aufrufen. Am Eingangsparameter EVENTN wird die Ereignisnummer der jeweiligen Diagnosemeldung angegeben. Der Alarm wird mit x=1 als kommend und mit x=0 als gehend in den Diagnosepuffer eingetragen. Im Diagnosepuffer erscheint neben der Zeit des Eintrags auch der entsprechende Diagnosetext, der in der Spalte "Bedeutung" angegeben ist.

Belegung des Diagnosedatensatzes DS0 in der Startinformation

Die folgende Tabelle zeigt die Belegung des Diagnosedatensatzes DS0 in der Startinformation. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Tabelle 3-7 Belegung des Diagnosedatensatzes DS0

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
0	0	BG in Störung	Wird bei jedem Diagnoseereignis gesetzt	8:x:00
	1	Interner Fehler	Wird bei allen internen Fehlern gesetzt: • Zeitüberwachung angesprochen • EEPROM-Inhalt ungültig; Baugruppe läuft an ohne zu regeln und wartet auf erneute Parametrierung durch die CPU • EPROM-Fehler • ADU/DAU-Fehler • Analogeingang Hardwarefehler	8:x:01
	2	Externer Fehler	Wird bei allen externen Fehlern gesetzt: • Externe Hilfsspannung fehlt • Parametrierung fehlerhaft • Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA) • Analogeingang Messbereichsunterschreitung • Analogeingang Messbereichsüberschreitung • Analogausgang Drahtbruch • Analogausgang Kurzschluss	8:x:02
	3	Fehler in einem Kanal	Weitere Aufschlüsselung siehe DS 1, ab Byte 7	8:x:03
	4	Externe Hilfsspannung fehlt	24-V-Versorgung der FM 455 ausgefallen	8:x:04
	6	EEPROM-Inhalt ungültig	Ausfall der Versorgungsspannung während eines Schreibvorgangs ins EEPROM. Die Baugruppe läuft mit Defaultparametern an.	8:x:03
	7	Parametrierung fehlerhaft	Die Baugruppe kann einen Parameter nicht verwerten. Grund: Parameter unbekannt oder unzulässige Kombination von Parametern. Siehe Menü **Zielsystem > Parametrierfehleranzeige**	8:x:07
1	0 ... 3	Baugruppenklasse	Ist immer mit 8 belegt	–
	4	Kanalspezifische Diagnose	Wird gesetzt, wenn die Baugruppe zusätzliche Kanalinformation liefern kann und ein Kanalfehler vorhanden ist (siehe DS 1 Byte 7 bis 12)	–

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
2	0	Anwendermodul falsch/fehlt	Die Stellung (A, B, C, D) des Messbereichsmoduls passt nicht zur Parametrierung der FM 455	8:x:30
	3	Zeitüberwachung angesprochen (watch dog)	Hardwarefehler	8:x:33
3	2	EPROM-Fehler	Baugruppe defekt	8:x:42
	4	ADU/DAU-Fehler	Baugruppe defekt	8:x:44

Belegung des Diagnosedatensatzes DS1 der FM 455

Der Diagnosedatensatz DS1 besteht aus 27 Bytes. Die ersten 4 Bytes sind identisch mit dem Diagnosedatensatz DS0. Die folgende Tabelle zeigt die Belegung der restlichen Bytes. Alle nicht aufgeführten Bits sind nicht von Bedeutung und null.

Tabelle 3-8 Belegung der Bytes 4 bis 12 des Diagnosedatensatzes DS1 der FM 455

Byte	Bit	Bedeutung	Anmerkung	Ereignisnr.
4	0 ... 7	Kanaltyp	Ist immer mit 75H belegt	–
5	0 ... 7	Länge der Diagnoseinformation	Ist immer mit 8 belegt	–
6	0 ... 7	Kanalanzahl	Ist immer mit 17 belegt (16 Regler + 1 Referenzkanal)	–
7	0 … 7	Kanalfehlervektor	Den Kanälen 1 bis 8 ist je ein Bit zugeordnet	-
8	0 … 7	Kanalfehlervektor	Den Kanälen 9 bis 16 ist je ein Bit zugeordnet	-
9	0	Fehler am Referenzkanal		-

Byte	Bit	Bedeutung	Anmerkung		Ereignisnr.
10	0	Analogeingang Hardwarefehler	Kanalspezifische Diagnose Kanal 1	–	8:x:B0
	1	Parametrierfehler		Die Stellung (A, B, C, D,) des Messbereichsmoduls passt nicht zur Parametrierung der FM 455	8:x:B1
	2	Analogeingang Drahtbruch (nur Bereich 4 bis 20 mA)			8:x:B2
	3	unbenutzt			8:x:B3
	4	Analogeingang Messbereichsunterschreitung[1]			8:x:B4
	5	Analogeingang Messbereichsüberschreitung[1]			8:x:B5
	6	Analogausgang Drahtbruch		Nur bei Stromausgang des K-Reglers	8:x:B6
	7	Analogausgang Kurzschluss		Nur bei Spannungsausgang des K-Reglers	8:x:B7
11	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 2		siehe oben
12	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 3		siehe oben
13	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 4		siehe oben
14	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 5		siehe oben
15	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 6		siehe oben
16	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 7		siehe oben
17	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 8		siehe oben
18	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 9		siehe oben
19	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 10		siehe oben
20	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 11		siehe oben
21	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 12		siehe oben
22	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 13		siehe oben

Byte	Bit	Bedeutung	Anmerkung	Ereig-nisnr.
23	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 14	siehe oben
24	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 15	siehe oben
25	0 ... 7	siehe Byte 10	Kanalspezifische Diagnose Kanal 16	siehe oben
26	0 ... 7	siehe Byte 10	Diagnose für Referenzkanal	siehe oben
[1] siehe Kapitel "Messumformerstörung (Seite 1132)"				

Was Sie besonders beachten müssen

Die FM 455 wird ausschließlich über den linken Frontstecker mit Spannung versorgt.

Die CPU erkennt deshalb in folgenden Fällen auf "Baugruppe gezogen/nicht ansprechbar":

- bei nicht gestecktem linken Frontstecker der FM 455
- bei fehlender 24-V-Versorgungsspannung am linken Frontstecker

Hinweis

Falls im Diagnosepuffer der CPU der Eintrag "Baugruppe gezogen/nicht ansprechbar" eingetragen ist, überprüfen Sie, ob der linke Frontstecker gesteckt ist und ob die 24-V-Versorgungsspannung der FM 455 vorhanden ist.

Weitere Informationen

Um weitere Informationen zum Parameter "RET_VALU" zu erhalten, siehe Liste der RET_VALU-Meldungen (Seite 1133) und im Bereich "Erweiterte Anweisungen" im Hilfesystem.

Messumformerstörung

Messumformerstörung

Folgende Messumformerstörungen können von der FM 455 erkannt werden:

- Messbereichsunterschreitung
- Messbereichsüberschreitung
- Drahtbruch (nicht bei allen Messbereichen)

Beim Auftreten einer dieser Störungen werden im Diagnosedatensatz DS0 das Sammelfehlerbit "Fehler extern" und im Diagnosedatensatz DS1 kanalspezifische Fehlerbits gesetzt (siehe Tabellen im vorhergehenden Abschnitt). Mit Verschwinden dieser Störungen werden die entsprechenden Bits wieder zurückgesetzt.

Nachfolgende Tabelle zeigt, bei welchen Grenzen in den einzelnen Messbereichen die Fehlerbits gesetzt bzw. rückgesetzt werden:

Messbereich	Fehlerbit Messbereichs-unterschreitung bei ...	Fehlerbit Messbereichs-überschreitung bei ...	Fehlerbit Drahtbruchanzeige
	DS1: Byte 10 bis 26, Bit 4	DS1: Byte 10 bis 26, Bit 5	DS1: Byte 10 bis 26, Bit 2
0 bis 20 mA	< - 3,5 mA	> 23,5 mA	–
4 bis 20 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA	> 22,8 mA	Fehlerbit = 1 bei < 3,6 mA Fehlerbit = 0 bei > 3,8 mA
0 bis 10 V	< - 1,175 V	> 11,75 V	–
Pt 100 (-200 bis 850 °C) (-328 bis 1562 °F)	< 30,82 mV	> 650,46 mV	ja, parametrierbar, Anzeige: 650,46 mV
Pt 100 (-200 bis 556 °C) (-328 bis 1032 °F)	< 30,82 mV	> 499,06 mV	ja, parametrierbar, Anzeige: 499,06 mV
Pt 100 (-200 bis 130 °C) (-328 bis 264 °F)	< 30,82 mV	> 254,12 mV	ja, parametrierbar, Anzeige: 254,12 mV
Thermoelement Typ B	< 0 mV	> 13,81 mV	ja, parametrierbar, der letzte Wert wird gehalten
Thermoelement Typ J	< - 8,1 mV	> 69,54 mV	ja, parametrierbar, der letzte Wert wird gehalten
Thermoelement Typ K	< - 6,45 mV	> 54,88 mV	ja, parametrierbar, der letzte Wert wird gehalten
Thermoelement Typ R	< - 0,23 mV	> 21,11 mV	ja, parametrierbar, der letzte Wert wird gehalten
Thermoelement Typ S	< - 0,24 mV	> 18,7 mV	ja, parametrierbar, der letzte Wert wird gehalten
Freies Thermoelement	< unterer Eingangswert des Polygonzugs	> oberer Eingangswert des Polygonzugs	ja, parametrierbar, der letzte Wert wird gehalten

Liste der RET_VALU-Meldungen

RET_VALU-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
7000	28672	-32624	Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.
7001	28673	-32624	Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
7002	28674	-32624	Zwischenaufruf (REQ irrelevant). Datenübertragung bereits aktiv; BUSY hat den Wert 1.
8090	32912	-32624	Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.
80A0	32928	-32608	Negative Quittung beim Lesen von Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2	32930	-32606	Protokollfehler bei Layer 2
80A3	32931	-32605	Protokollfehler bei User-Interface / User
80A4	32932	-32604	Kommunikation am K-Bus gestört
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4	32964	-32572	Kommunikationsfehler
80C5	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)
80ff	33023	-32513	Falsche Indexangabe beim Baustein FMT_PAR
Die Fehler 80A2..80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein. Meldungen der Form 7xxx zeigen temporäre Betriebszustände der Kommunikation an.			

3.5.7.6 Beispielprojekte zur FM 455

3.5.8 ET 200S 2PULSE einsetzen

3.5.8.1 Anschlussbelegung des 2PULSE

Verdrahtungsregeln

Die Leitungen (Klemmen 1 und 2 sowie Klemmen 5 und 6) müssen geschirmt sein. Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung des 2PULSE

Die folgende Tabelle zeigt die Anschlussbelegung für das 2PULSE.

Ansicht	Anschlussbelegung	Bedeutung
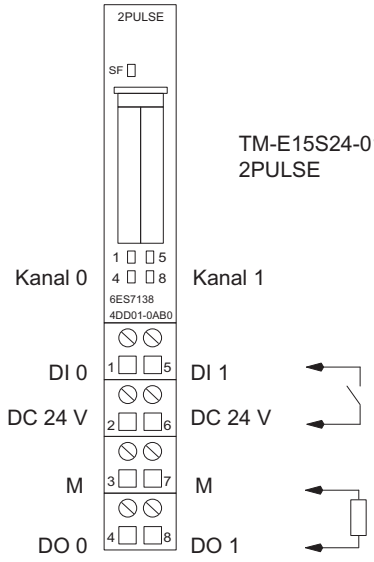	TM-E15S24-01 und 2PULSE	Kanal 0: Klemme 1...4 Kanal 1: Klemme 5...8 DC24V: Geberversorgung M: Masse DI: Eingangssignal DO: Ausgangssignal (max. 2 A pro Kanal)

3.5.8.2 Betriebsarten und Funktionen

Übersicht

Prinzip

Das 2PULSE ist 2kanalig. Sie können für jeden Kanal eine eigene Betriebsart auswählen. Die Betriebsart legen Sie beim Parametrieren fest. Mit Ihrem Steuerungsprogramm ist die parametrierte Betriebsart nicht mehr änderbar.

Sie können für jeden Kanal zwischen verschiedenen Betriebsarten wählen:

- Impulsausgabe (Seite 1137)
- Pulsweitenmodulation (Seite 1143)
- Impulskette (Seite 1153)
- Ein-/Ausschaltverzögerung (Seite 1159)
- Frequenzausgabe (Seite 1167) (ab 6ES7138-4DD01-0AB0)

Zusätzlich zu der eingestellten Betriebsart kann das 2PULSE folgende Funktionen ausführen:

- Strommessung (Seite 1173) in den Betriebsarten "Pulsweitenmodulation" und "Impulskette" (ab 6ES7138-4DD01-0AB0)
- Direktes Steuern des Digitalausgangs DO (Seite 1174) über Ihr Steuerungsprogramm; für jeden Kanal getrennt steuerbar.
- Fehlererkennung / Diagnose (Seite 1175); das 2PULSE erkennt für jeden Kanal getrennt die Fehler.
- Parallelschaltung beider Kanäle (Seite 1177) , um einen höheren Ausgangsstrom zu erreichen (ab 6ES7138-4DD01-0AB0).
- Verhalten bei CPU/Master-STOP (Seite 1179); das 2PULSE erkennt den CPU/Master-STOP gemeinsam für beide Kanäle und reagiert entsprechend Ihrer Parametrierung.

Das folgende Bild zeigt die Arbeitsweise des 2PULSE.

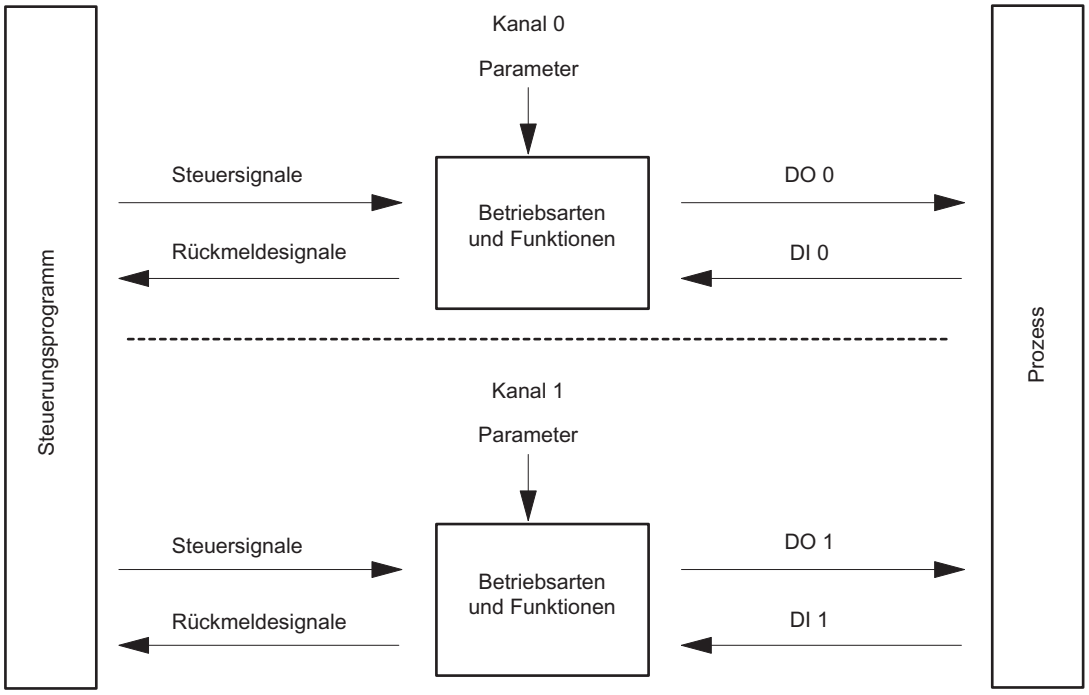

Schnittstellen zum Steuerungsprogramm und zum Prozess

Zur Ausführung der Betriebsarten und Funktionen hat das 2PULSE als Schnittstelle zum Prozess pro Kanal je einen Digitaleingang und einen Digitalausgang (DI 0, DO 0 für Kanal 0 und DI 1, DO 1 für Kanal 1).

Die Betriebsarten und Funktionen steuern und beobachten Sie mit Ihrem Steuerungsprogramm durch Steuersignale (Seite 1184) und Rückmeldesignale (Seite 1187).

Den einzelnen Betriebsarten sind Parameter zugeordnet.

In den Beschreibungen zu den Betriebsarten und Funktionen finden Sie:

- die jeweils relevanten Parameter und
- die Steuer- und Rückmeldesignale.

Die Beschreibung der Betriebsarten und Funktionen gilt jeweils für beide Kanäle, deshalb ist der einzelne Kanal in der Beschreibung nicht mehr gesondert gekennzeichnet.

Betriebsart "Impulsausgabe"

Betriebsart "Impulsausgabe"

Impulsausgabe

Für die von Ihnen vorgegebene Impulsdauer gibt das 2PULSE nach Ablauf der eingestellten Einschaltverzögerung (Seite 1182) einen Impuls am Digitalausgang DO aus (Ausgabesequenz).

Das folgende Bild zeigt das Prinzipschaltbild für die Betriebsart "Impulsausgabe".

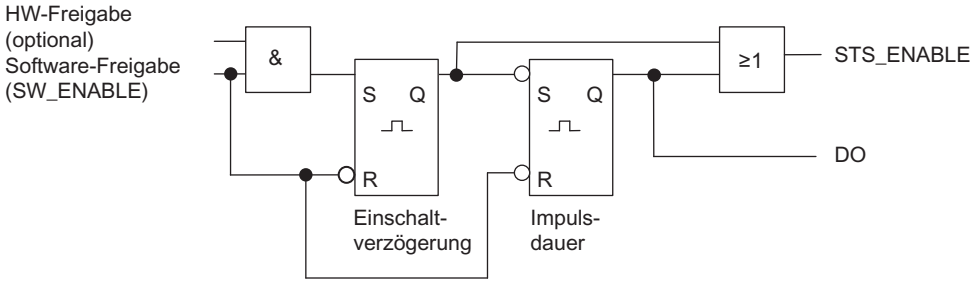

Starten der Ausgabesequenz

Die Freigabe für die Ausgabesequenz müssen Sie immer per Software-Freigabe (Seite 1184) (SW_ENABLE 0 → 1; MANUAL_DO = 0) in Ihrem Steuerungsprogramm erteilen.

Das Rückmeldebit (Seite 1187) ACK_SW_ENABLE zeigt die am 2PULSE anstehende Software-Freigabe an.

Zusätzlich können Sie mit dem Parameter "Funktion DI (Seite 1182)" den Digitaleingang DI des 2PULSE als Hardware-Freigabe (HW-Freigabe) einstellen.

Wenn Sie mit Software-Freigabe und Hardware-Freigabe gleichzeitig arbeiten, startet die Ausgabesequenz bei erteilter Software-Freigabe mit der ersten positiven Flanke der Hardware-Freigabe. Weitere positive Flanken der Hardware-Freigabe während der laufenden Ausgabesequenz werden vom 2PULSE ignoriert. Bei erteilter Software-Freigabe reicht zum Start der nächsten Ausgabesequenz eine positive Flanke der Hardware-Freigabe.

Durch Erteilen der Freigabe (positive Flanke) wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Nach Ablauf der Einschaltverzögerung wird der Impuls mit der vorgegebenen Impulsdauer ausgegeben. Die Ausgabesequenz endet (Seite 1172) mit dem Ende des Impulses; STS_ENABLE wird gelöscht.

Wenn Sie die Impulsdauer im Betrieb unzulässig ändern, zeigt Ihnen das Signal (Seite 1187) ERR_PULS einen Impulsausgabefehler an. Danach müssen Sie die Ausgabesequenz neu starten.

Mit dem nächsten Start der Ausgabesequenz löscht das 2PULSE das Rückmeldebit ERR_PULS.

Impulsschema

Das folgende Bild zeigt eine Ausgabesequenz in der Betriebsart "Impulsausgabe".

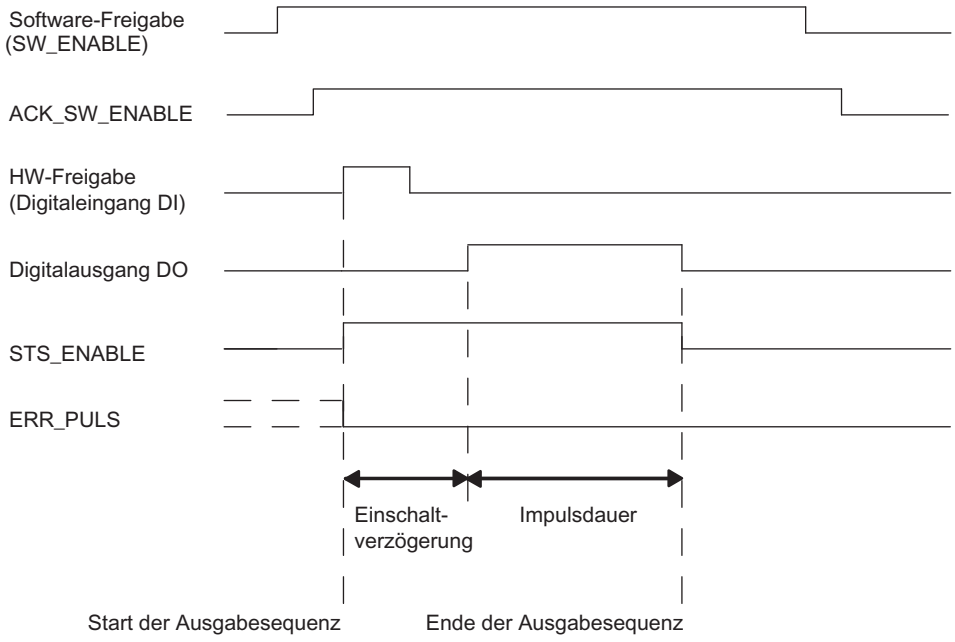

Abbrechen der Ausgabesequenz

Durch Löschen der Software-Freigabe (SW_ENABLE = 0) während der Einschaltverzögerung oder der Impulsdauer wird die Ausgabesequenz abgebrochen und STS_ENABLE und der Digitalausgang DO werden gelöscht.

Danach müssen Sie die Ausgabesequenz neu starten.

Wahrheitstabelle

Die folgende Tabelle zeigt, auf welche Signale das 2PULSE wie reagiert.

Software-Freigabe SW_ENABLE	Hardware-Freigabe (Digitaleingang DI)	Digitalausgang DO	STS_ENABLE	Ausgabesequenz
1	0 → 1	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0 → 1	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0	beliebiger Zustand	0	0	abbrechen
1	beliebiger Zustand	alter Zustand bleibt		-
0 → 1	0	0	0	-

Zeiten einstellen mittels Zeitbasis

Sie wählen über die parametrierbare Zeitbasis (Seite 1182) die Auflösung und den Wertebereich der Impulsdauer und der Einschaltverzögerung.

Zeitbasis = 0,1 ms:	Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
Zeitbasis = 1 ms:	Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Impulsdauer einstellen und ändern

Die Impulsdauer (Seite 1183) geben Sie direkt als Zahlenwert zwischen 0 und 65535 in Ihrem Steuerungsprogramm vor:

Impulsdauer = Zeitbasis × vorgegebener Zahlenwert

Ändern Sie die Impulsdauer während einer laufenden Ausgabesequenz, wird die schon ausgegebene Zeit von der neuen Impulsdauer subtrahiert und der Impuls weiter ausgegeben.

Impulsdauer verkürzen

Haben Sie die Impulsdauer auf eine Zeit verkürzt, die kleiner ist als die schon ausgegebene Zeit, wird die Ausgabesequenz abgebrochen, STS_ENABLE und der Digitalausgang DO werden gelöscht und das Statusbit ERR_PULS gesetzt. Mit der nächsten Ausgabesequenz wird das Statusbit ERR_PULS gelöscht.

Einschaltverzögerung einstellen und ändern bei Nutzung der kurzen Steuerschnittstelle

Die Einschaltverzögerung (Seite 1182) legen Sie als Zahlenwert zwischen 0 und 65535 in den Parametern fest:

parametrierte Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Mit dem Faktor für die Einschaltverzögerung können Sie die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung

Wenn Sie den Faktor der Einschaltverzögerung während der Ausgabesequenz ändern, wird die neue Einschaltverzögerung bei der nächsten Ausgabesequenz wirksam.

Taktsynchroner Betrieb (ab 6ES7138-4DD01-0AB0)

In der Betriebsart "Impulsausgabe" hat der taktsynchrone Betrieb keinen Einfluss auf die Funktionalität.

Parameter der Betriebsart "Impulsausgabe"

Parameter der Betriebsart "Impulsausgabe"

Die folgende Tabelle zeigt die Parameter der Betriebsart "Impulsausgabe".

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart (Seite 1181)	Stellen Sie die Betriebsart "Impulsausgabe" ein.	• Impulsausgabe • Pulsweitenmodulation • Impulskette • Ein-/Ausschaltverzögerung • Frequenzausgabe	Pulsweitenmodulation
Zeitbasis (Seite 1182)	Sie wählen mit der Zeitbasis die Auflösung und den Wertebereich der Impulsdauer und der Einschaltverzögerung.	• 0,1 ms • 1 ms	0,1 ms
Funktion DI (Seite 1182)	Den Digitaleingang DI können Sie als Eingang verwenden oder als Hardware-Freigabe.	• Eingang • HW-Freigabe	Eingang
Einschaltverzögerung (Seite 1182) [1]	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe des Impulses abläuft. Die Einschaltverzögerung können Sie in Ihrem Steuerungsprogramm mit dem "Faktor Einschaltverzögerung" ändern.	0 bis 65535	0
[1] Nur bei Nutzung der kurzen Steuerschnittstelle (Seite 1184) Bei Nutzung der langen Steuerschnittstelle geben Sie diesen Wert direkt in der Steuerschnittstelle vor.			

Steuer- und Rückmeldesignale der Betriebsart "Impulsausgabe"

Steuer- und Rückmeldesignale der Betriebsart "Impulsausgabe"

Die folgende Tabelle zeigt die Steuer- und Rückmeldesignale der Betriebsart "Impulsausgabe".

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Steuersignale bei Nutzung der kurzen Steuerschnittstelle (8 Byte)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 2: Bit 0	Byte 6: Bit 0
Impulsdauer	Zeit, die der Digitalausgang DO nach Ablauf der Einschaltverzögerung gesetzt ist.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, wird vom 2PULSE kein Impuls ausgegeben.	Word 0	Word 4

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Faktor Einschaltverzögerung	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 x parametrierte Einschaltverzögerung	0 bis 255 Bei Einschaltverzögerung < 0,2 ms oder bei Faktor = 0 ist die wirksame Einschaltverzögerung = 0. Bei Einschaltverzögerung > 65,535 s wird die Einschaltverzögerung auf 65,535 s begrenzt.	Byte 3	Byte 7
Steuersignale bei Nutzung der langen Steuerschnittstelle (12 Byte) (ab 6ES7138-4DD01-0AB0)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 4: Bit 0	Byte 10: Bit 0
Impulsdauer	Zeit, die der Digitalausgang DO nach Ablauf der Einschaltverzögerung gesetzt ist.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, wird vom 2PULSE kein Impuls ausgegeben.	Word 0	Word 6
Einschaltverzögerung	Sie können die Einschaltverzögerung vor dem Start der Ausgabesequenz ändern.	0 bis 65535 Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert	Word 2	Word 8
Rückmeldesignale				
STS_ENABLE	Zeigt eine laufende Ausgabesequenz an.	0 = Impulsausgabe gesperrt 1 = Impulsausgabe läuft	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Zeigt den Zustand von SW_ENABLE an.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Zeigt einen Impulsausgabefehler an.	0 = kein Impulsausgabefehler 1 = Impulsausgabefehler	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (ab 6ES7138-4DD01-0AB0)	Wird nach Abschluss einer Ausgabesequenz inkrementiert.	Ohne HW-Freigabe: 0 … 1 Mit HW-Freigabe: 0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reserviert	–	0	Word 2	Word 6

Ein- und Ausgangssignale der Betriebsart "Impulsausgabe"

Ein- und Ausgangssignale der Betriebsart "Impulsausgabe"

Die folgende Tabelle zeigt die Ein- und Ausgangssignale der Betriebsart "Impulsausgabe".

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Eingangssignal				
HW-Freigabe	Die HW-Freigabe können Sie mit dem Parameter "Funktion DI" wählen. Das Signal des Digitaleingangs DI wird dann vom 2PULSE zum Start der Ausgabesequenz ausgewertet.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe erteilt 0 → 1 = Start der Ausgabesequenz; abhängig von der Software-Freigabe (SW_ENABLE)	1	5
Ausgangssignal				
Impuls am Digitalausgang DO	Für die vorgegebene Impulsdauer wird am Digitalausgang DO ein Impuls ausgegeben.	0 = kein Impuls 1 = Impuls	4	8

Betriebsart "Pulsweitenmodulation" (PWM)

Betriebsart "Pulsweitenmodulation" (PWM)

Pulsweitenmodulation

Sie geben dem 2PULSE einen Ausgabewert vor. Das 2PULSE erzeugt daraus eine kontinuierliche Impulsfolge. Der Ausgabewert bestimmt das Puls/Pause-Verhältnis in einer Periode (Pulsweitenmodulation). Die Periodendauer ist einstellbar.

Die Impulsfolge wird nach Ablauf der parametrierten Einschaltverzögerung (Seite 1182) am Digitalausgang DO des 2PULSE ausgegeben (Ausgabesequenz).

Das folgende Bild zeigt das Prinzipschaltbild für die Betriebsart "Pulsweitenmodulation".

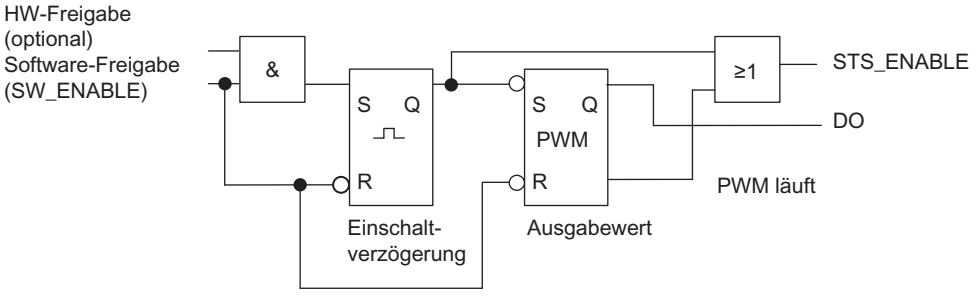

Starten der Ausgabesequenz

Die Freigabe für die Ausgabesequenz müssen Sie immer per Software-Freigabe (Seite 1184) (SW_ENABLE 0 → 1; MANUAL_DO = 0) in Ihrem Steuerungsprogramm erteilen.

Das Rückmeldebit (Seite 1187) ACK_SW_ENABLE zeigt die am 2PULSE anstehende Software-Freigabe an.

Zusätzlich können Sie mit dem Parameter "Funktion DI (Seite 1182)" den Digitaleingang DI des 2PULSE als HW-Freigabe einstellen. Im taktsynchronen Betrieb (ab 6ES7138-4DD01-0AB0) ist die Projektierung einer HW-Freigabe nicht wirksam.

Wenn Sie mit Software-Freigabe und Hardware-Freigabe gleichzeitig arbeiten, startet die Ausgabesequenz bei erteilter Software-Freigabe mit der ersten positiven Flanke der Hardware-Freigabe. Weitere positive Flanken der Hardware-Freigabe während der laufenden Ausgabesequenz werden vom 2PULSE ignoriert.

Durch Erteilen der Freigabe (positive Flanke) wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Nach Ablauf der Einschaltverzögerung wird die Impulsfolge ausgegeben. Die Ausgabesequenz läuft endlos, so lange SW_ENABLE gesetzt ist.

Impulsschema

Das folgende Bild zeigt eine Ausgabesequenz in der Betriebsart "Pulsweitenmodulation".

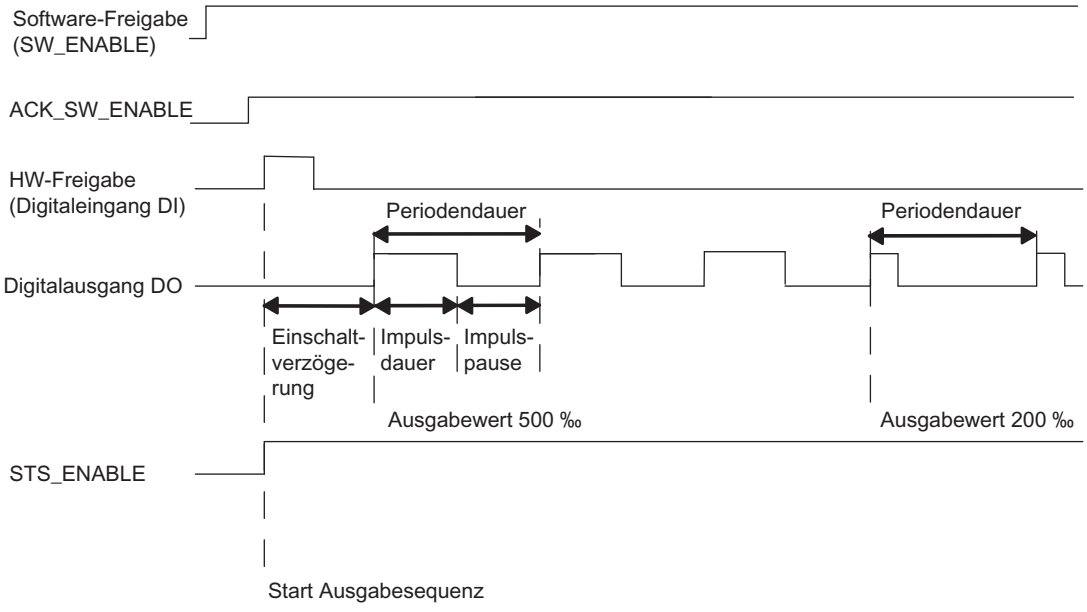

Abbrechen der Ausgabesequenz

Durch Löschen der Software-Freigabe (SW_ENABLE = 0) während der Einschaltverzögerung oder der Impulsausgabe wird die Ausgabesequenz abgebrochen und STS_ENABLE und der Digitalausgang DO werden gelöscht.

Danach müssen Sie die Ausgabesequenz erneut starten.

Wahrheitstabelle

Die folgende Tabelle zeigt, auf welche Signale das 2PULSE wie reagiert.

Software-Freigabe SW_ENABLE	Hardware-Freigabe (Digitaleingang DI)	Digitalausgang DO	STS_ENABLE	Ausgabesequenz
1	0 → 1	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0 → 1	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0	beliebiger Zustand	0	0	abbrechen
1	beliebiger Zustand	alter Zustand bleibt		-
0 → 1	0	0	0	-

Mindestimpulsdauer und Mindestimpulspause

Der proportionalen Ausgangskennlinie werden die Mindestimpulsdauer und die Mindestimpulspause überlagert.

Mindestimpulsdauer und Mindestimpulspause parametrieren Sie mit dem Parameter "Mindestimpulsdauer (Seite 1183)", sie haben immer denselben Wert.

Eine vom 2PULSE berechnete Impulsdauer, die kleiner ist als die Mindestimpulsdauer, wird unterdrückt.

Eine vom 2PULSE berechnete Impulsdauer, die größer ist als Periodendauer minus Mindestimpulspause, wird auf 1000 ‰ gesetzt.

Das folgende Bild zeigt die Modulation der Impulsdauer.

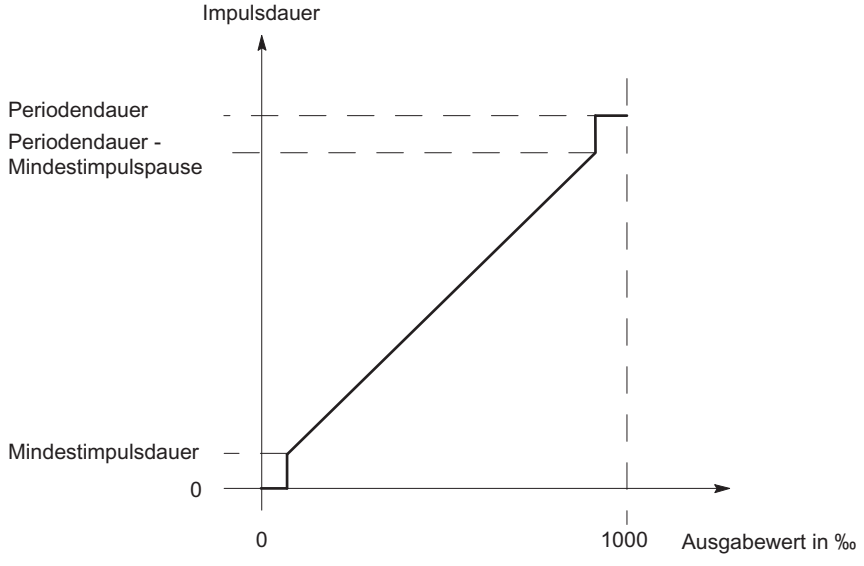

Zeiten einstellen mittels Zeitbasis

Sie wählen über die parametrierbare Zeitbasis (Seite 1182) die Auflösung und den Wertebereich der Periodendauer, der Mindestimpulsdauer und der Einschaltverzögerung.

Zeitbasis = 0,1 ms:	Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
Zeitbasis = 1 ms:	Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Ausgabewert einstellen und ändern

Sie wählen mit dem Parameter "Ausgabeformat (Seite 1182)" den Wertebereich des Ausgabewertes.

- Ausgabeformat "Promille": Wertebereich zwischen 0 und 1000
  Mit dem vorgegebenen Ausgabewert berechnet das 2PULSE die Impulsdauer:
  Impulsdauer = (Ausgabewert / 1000) × Periodendauer
- Ausgabeformat "S7-Analogausgabe": Wertebereich zwischen 0 und 27648
  Mit dem vorgegebenen Ausgabewert berechnet das 2PULSE die Impulsdauer:
  Impulsdauer = (Ausgabewert / 27648) × Periodendauer

Den Ausgabewert geben Sie direkt mit Ihrem Steuerungsprogramm vor. Der neue Ausgabewert wird mit der nächsten steigenden Flanke der Ausgabe übernommen.

Einstellung der Periodendauer

Abhängig von der Projektierung wird die Periodendauer als Parameterwert oder in der Steuerschnittstelle vorgegeben. Die folgende Tabelle zeigt, wann welcher Wert verwendet wird.

Für den taktsynchronen Betrieb (ab 6ES7138-4DD01-0AB0) müssen Sie die Periodendauer bereits bei der Parametrierung festlegen. Im Gegensatz zum nicht taktsynchronen Betrieb können Sie hier die Periodendauer im späteren Betrieb nicht mehr ändern.

	Projektierung mit …	
	kurzer Steuerschnittstelle	langer Steuerschnittstelle
Taktsynchroner Betrieb	Periodendauer = Parameterwert "Periodendauer" Der Wert aus der Steuerschnittstelle wird ignoriert.	
Nicht taktsynchroner Betrieb	Periodendauer = Parameterwert "Periodendauer" × "Faktor Periodendauer" aus Steuerschnittstelle	Periodendauer = Sollwert "Periodendauer" aus Steuerschnittstelle Der Parameterwert "Periodendauer" wird ignoriert.

Periodendauer beim Parametrieren mit kurzer Steuerschnittstelle

Die Periodendauer (Seite 1183) legen Sie abhängig von der Zeitbasis als Zahlenwert in den Parametern fest:

- bei Zeitbasis 1 ms: von 1 bis 65535
- bei Zeitbasis 0,1 ms: von 2 bis 65535

parametrierte Periodendauer = Zeitbasis × vorgegebener Zahlenwert

Mit dem Faktor für die Periodendauer können Sie die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Periodendauer = Faktor × 0,1 × parametrierte Periodendauer

Die neuen Sollwerte für die Periodendauer werden mit der nächsten steigenden Flanke der Ausgabe übernommen.

Hinweis

Bei Änderung der Periodendauer im laufenden Betrieb kann es zu einer kurzzeitigen geringen Verlängerung der Periodendauer kommen (ca. 10 µs).

Hinweis

Eine Änderung der Periodendauer über die Steuerschnittstelle ist im taktsynchronen Betrieb (ab 6ES7138-4DD01-0AB0) nicht wirksam.

Mindestimpulsdauer und Mindestimpulspause einstellen

Die Mindestimpulsdauer und die Mindestimpulspause legen Sie als Zahlenwert zwischen 0 und 65535 mit dem Parameter "Mindestimpulsdauer (Seite 1183)" fest:

parametrierte Mindestimpulsdauer/Mindestimpulspause = Zeitbasis × vorgegebener Zahlenwert

Einschaltverzögerung einstellen

Die Einschaltverzögerung (Seite 1182) legen Sie als Zahlenwert zwischen 0 und 65535 in den Parametern fest.

parametrierte Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Bei Verwendung der langen Steuerschnittstelle können Sie mit dem Faktor für die Einschaltverzögerung die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung

Strommessung (ab 6ES7138-4DD01-0AB0)

Ein wichtiger Anwendungsfall der Strommessung in der Betriebsart "Pulsweitenmodulation" ist die Stromregelung eines Proportionalventils. Damit können zum Beispiel thermische Auswirkungen am Ventil kompensiert werden.

Weitere Informationen finden Sie unter "Funktion: Strommessung (ab 6ES7138-4DD01-0AB0) (Seite 1173)".

Taktsynchroner Betrieb in der Betriebsart "Pulsweitenmodulation"

Taktsynchroner Betrieb

Hinweis

Taktsynchroner Betrieb ist mit dem 2PULSE ab der Artikelnummer 6ES7138-4DD01-0AB0 möglich.

Der taktsynchrone Betrieb hat in der Betriebsart "Pulsweitenmodulation" folgende Besonderheiten:

- Die Ausgabe beginnt immer ohne Einschaltverzögerung, eine parametrierte Einschaltverzögerung wird ignoriert.
- Die Ausgabe beginnt immer nach Erteilen der Software-Freigabe, eine parametrierte Hardware-Freigabe wird ignoriert.
- Eine Änderung der Periodendauer im laufenden Betrieb ist nicht möglich, entsprechende Vorgaben über die Steuerschnittstelle werden ignoriert.

Im taktsynchronen Betrieb ist die Ausgabesequenz mit dem Zeitpunkt T_o synchronisiert. Die Periodendauer wird auf den DP-Zyklus abgestimmt. Ihre vorgegebene Periodendauer kann dabei gegebenenfalls nicht genau realisiert werden.

Der parametrierte Wert der Periodendauer wird vom 2PULSE nach einem Berechnungsalgorithmus an den DP-Zyklus angepasst. Die Berechnung erfolgt so, dass die Differenz zwischen vorgegebener und berechneter Periodendauer minimiert ist. Im ungünstigsten Fall beträgt die Abweichung die Hälfte des DP-Zyklus. Die folgende Tabelle zeigt Beispiele dazu.

DP-Zykluszeit T_{DP}[1)]	vorgegebene Periodendauer T_{Soll}	berechnete Periodendauer T_{Ist}[1)]
10 ms	5 ms	5 ms
10 ms	2 ms	2 ms
10 ms	6 ms	5 ms
10 ms	16 ms	20 ms

1) Die berechnete Periodendauer und die DP-Zykluszeit stehen immer in einem ganzzahligen Verhältnis.

Nachfolgend ist der Zusammenhang zwischen Periodendauer und DP-Zyklus grafisch dargestellt. Das Puls/Pause-Verhältnis ist in den Beispielen jeweils mit 50 % dargestellt.

Weiterführende Informationen finden Sie im Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

- Periodendauer ist gleich dem DP-Zyklus

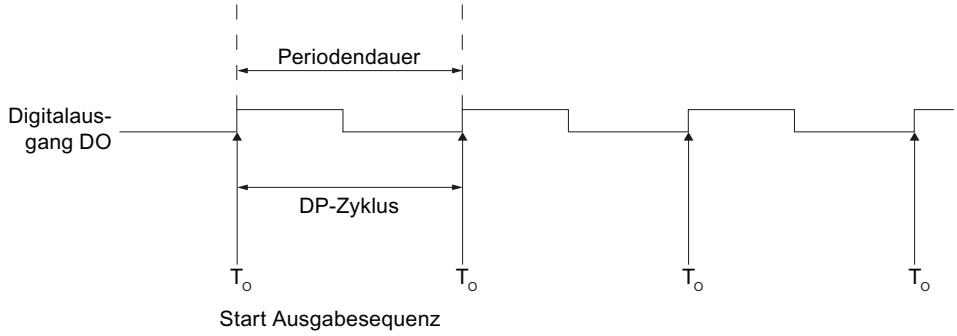

- Periodendauer ist kleiner als der DP-Zyklus

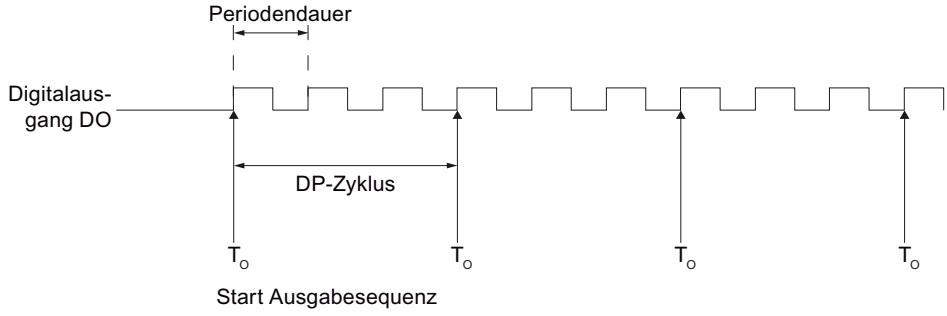

- Periodendauer ist größer als der DP-Zyklus

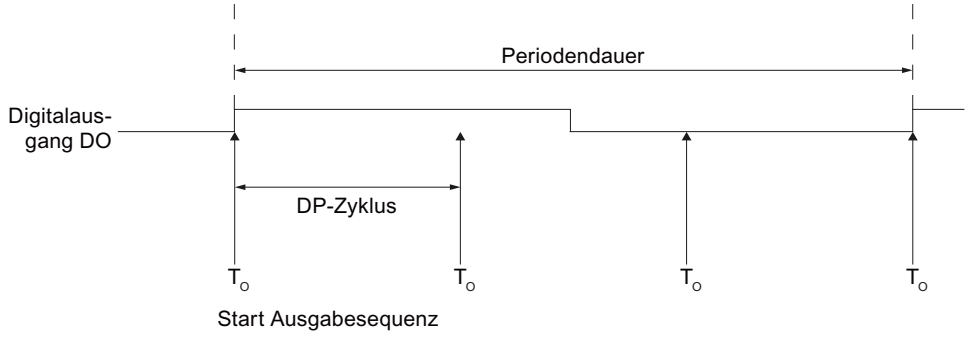

Parameter der Betriebsart "Pulsweitenmodulation"

Parameter der Betriebsart "Pulsweitenmodulation"

Die folgende Tabelle zeigt die Parameter der Betriebsart "Pulsweitenmodulation".

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart (Seite 1181)	Stellen Sie die Betriebsart "Pulsweitenmodulation" ein.	• Impulsausgabe • Pulsweitenmodulation • Impulskette • Ein-/Ausschaltverzögerung • Frequenzausgabe	Pulsweitenmodulation
Zeitbasis (Seite 1182)	Sie wählen mit der Zeitbasis die Auflösung und den Wertebereich der Periodendauer, der Mindestimpulsdauer und der Einschaltverzögerung.	• 0,1 ms • 1 ms	0,1 ms
Ausgabeformat (Seite 1182)	Je nach benötigter Auflösung des Ausgabewertes wählen Sie zwischen den Ausgabeformaten Promille oder S7-Analogausgabe.	• Promille • S7-Analogausgabe	Promille
Funktion DI (Seite 1182)	Den Digitaleingang DI können Sie als Eingang verwenden oder als HW-Freigabe.	• Eingang • HW-Freigabe	Eingang
Einschaltverzögerung (Seite 1182)	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulsfolge abläuft.	0 bis 65535	0
Mindestimpulsdauer (Seite 1183)	Mindestimpulsdauer und Mindestimpulspause: Hier tragen Sie die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds ein.	0 bis 65535	0
Periodendauer (Seite 1183) [1]	Sollwert der Periodendauer Beachten Sie bei der Vorgabe der Periodendauer die eingestellte Mindestimpulsdauer und die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds. Bei Nutzung der kurzen Steuerschnittstelle können Sie im nicht taktsynchronen Betrieb die Periodendauer in Ihrem Steuerungsprogramm mit dem "Faktor Periodendauer" ändern.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535	20000 → 2 s

[1] Beachten Sie die Informationen zur "Einstellung der Periodendauer (Seite 1143)" und zum "Taktsynchronen Betrieb (Seite 1148)".

Steuer- und Rückmeldesignale der Betriebsart "Pulsweitenmodulation"

Steuer- und Rückmeldesignale der Betriebsart "Pulsweitenmodulation"

Die folgende Tabelle zeigt die Steuer- und Rückmeldesignale der Betriebsart "Pulsweitenmodulation".

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Steuersignale bei Nutzung der kurzen Steuerschnittstelle (8 Byte)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 2: Bit 0	Byte 6: Bit 0
Ausgabewert	Wert, der am Digitalausgang DO pulsweitenmoduliert ausgegeben wird.	je nach Ausgabeformat: • Promille: 0 ... 1000 • S7-Analogausgabe: 0 ... 27648 Geben Sie einen Ausgabewert > 1000 bzw. 27648 vor, begrenzt das 2PULSE diesen auf 1000 bzw. 27648.	Word 0	Word 4
Faktor Periodendauer	Sie können die parametrierte Periodendauer ändern: Periodendauer = Faktor × 0,1 × parametrierte Periodendauer	Faktor: 0 bis 255 Periodendauer: 2 × Mindestimpulsdauer bis 65,635 s. Ergibt sich eine Periodendauer < 2 × Mindestimpulsdauer oder < 200 µs oder ist der Faktor = 0, dann wird die Periodendauer auf 2 × Mindestimpulsdauer, mindestens aber 200 µs begrenzt. Ergibt sich eine Periodendauer > 65,535 s, wird diese auf 65,535 s begrenzt.	Byte 3	Byte 7
Steuersignale bei Nutzung der langen Steuerschnittstelle (12 Byte) (ab 6ES7138-4DD01-0AB0)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 4: Bit 0	Byte 10: Bit 0
Ausgabewert	Wert, der am Digitalausgang DO pulsweitenmoduliert ausgegeben wird.	je nach Ausgabeformat: • Promille: 0 ... 1000 • S7-Analogausgabe: 0 ... 27648 Geben Sie einen Ausgabewert > 1000 bzw. 27648 vor, begrenzt das 2PULSE diesen auf 1000 bzw. 27648.	Word 0	Word 6

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Periodendauer [1]	Sollwert der Periodendauer im nicht taktsynchronen Betrieb	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535	Word 2	Word 8
Faktor Einschaltverzögerung	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung	0 bis 255 Bei Einschaltverzögerung < 0,2 ms oder bei Faktor = 0 ist die wirksame Einschaltverzögerung = 0. Bei Einschaltverzögerung > 65,535 s wird die Einschaltverzögerung auf 65,535 s begrenzt.	Byte 5	Byte 11
Rückmeldesignale				
STS_ENABLE	Zeigt eine laufende Ausgabesequenz an.	0 = Impulsausgabe gesperrt 1 = Impulsausgabe läuft	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Zeigt den Zustand von SW_ENABLE an.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 0: Bit 3	Byte 4: Bit 3
SEQ_CNT (ab 6ES7138-4DD01-0AB0)	Keine Funktion in dieser Betriebsart.	–	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Strommesswert (ab 6ES7138-4DD01-0AB0)	Strom am DO als SIMATIC S7-Analogwert.	0 … 27648 (Übersteuerbereich bis 32767)	Word 2	Word 6
[1] Beachten Sie die Informationen zur "Einstellung der Periodendauer (Seite 1143)".				

Ein- und Ausgangssignale der Betriebsart "Pulsweitenmodulation"

Ein- und Ausgangssignale der Betriebsart "Pulsweitenmodulation"

Die folgende Tabelle zeigt die Ein- und Ausgangssignale der Betriebsart "Pulsweitenmodulation".

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Eingangssignal				
HW-Freigabe	Die HW-Freigabe können Sie mit dem Parameter "Funktion DI" wählen. Das Signal des Digitaleingangs DI wird dann vom 2PULSE zum Start der Ausgabesequenz ausgewertet.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe erteilt 0 → 1 = Start der Ausgabesequenz; abhängig von der Software-Freigabe	1	5

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Ausgangssignal				
Impulsfolge am Digitalausgang DO	Die Impulsfolge wird am Digitalausgang DO ausgegeben.	0 = kein Impuls 1 = Impuls	4	8

Betriebsart "Impulskette"

Betriebsart "Impulskette"

Definition

Die von Ihnen vorgegebene Impulsanzahl gibt das 2PULSE nach Ablauf der eingestellten Einschaltverzögerung (Seite 1182) am Digitalausgang DO als Impulskette aus (Ausgabesequenz). Die Perioden- und Impulsdauer der Impulse ist einstellbar.

Das folgende Bild zeigt das Prinzipschaltbild für die Betriebsart "Impulskette".

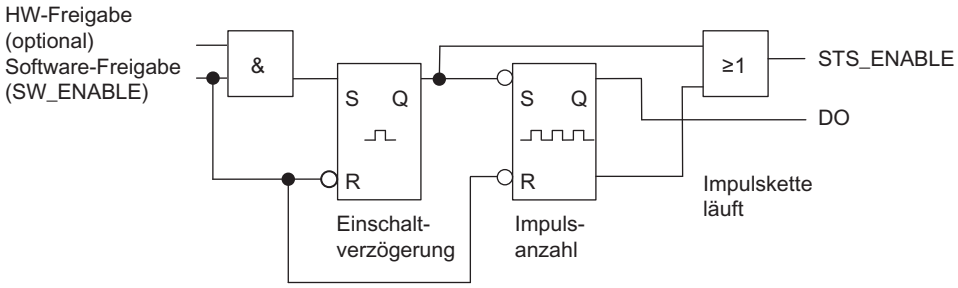

Starten der Ausgabesequenz

Die Freigabe für die Ausgabesequenz müssen Sie immer per Software-Freigabe (Seite 1184) (SW_ENABLE 0 → 1; MANUAL_DO = 0) in Ihrem Steuerungsprogramm erteilen. Das Rückmeldebit (Seite 1187) ACK_SW_ENABLE zeigt die am 2PULSE anstehende Software-Freigabe an.

Zusätzlich können Sie mit dem Parameter "Funktion DI (Seite 1182)" den Digitaleingang DI des 2PULSE als HW-Freigabe einstellen.

Wenn Sie mit Software-Freigabe und Hardware-Freigabe gleichzeitig arbeiten, startet die Ausgabesequenz bei erteilter Software-Freigabe mit der ersten positiven Flanke der Hardware-Freigabe. Weitere positive Flanken der Hardware-Freigabe während der laufenden Ausgabesequenz werden vom 2PULSE ignoriert. Bei erteilter Software-Freigabe reicht zum Start der nächsten Ausgabesequenz eine positive Flanke der Hardware-Freigabe.

Durch Erteilen der Freigabe (positive Flanke) wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Nach Ablauf der Einschaltverzögerung wird die Impulskette mit der vorgegebenen Impulsanzahl ausgegeben. Die Ausgabesequenz endet (Seite 1172), sobald der letzte Impuls ausgegeben ist; STS_ENABLE wird gelöscht.

Wenn Sie die Impulsanzahl im Betrieb unzulässig ändern, zeigt Ihnen das Signal (Seite 1187) ERR_PULS einen Impulsausgabefehler an.

Mit der nächsten Ausgabesequenz löscht das 2PULSE das Rückmeldebit ERR_PULS.

Impulsschema

Das folgende Bild zeigt eine Ausgabesequenz in der Betriebsart "Impulskette".

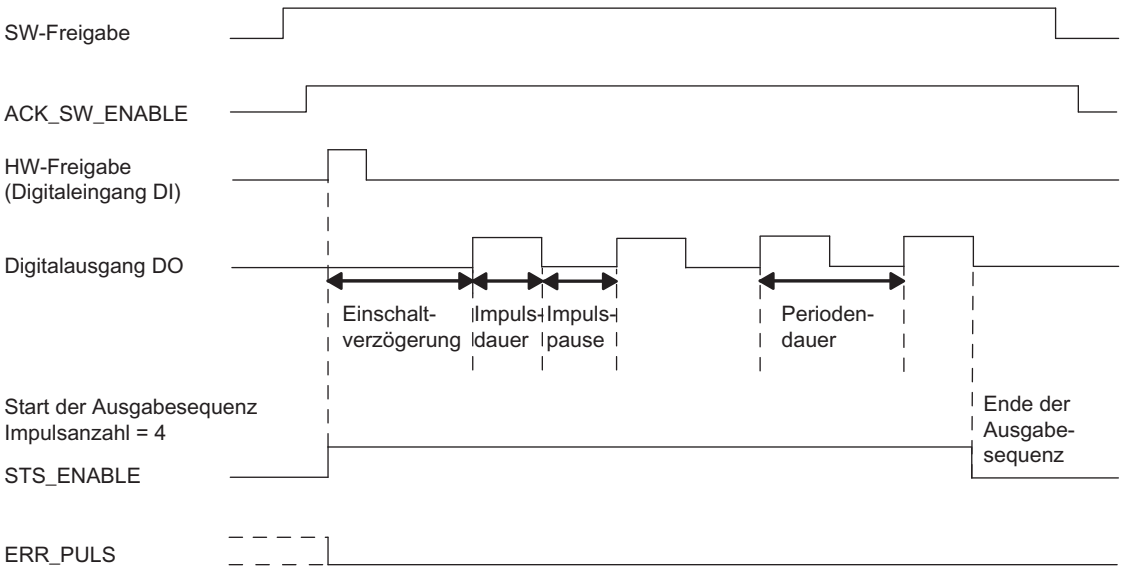

Abbrechen der Ausgabesequenz

Durch Löschen der Software-Freigabe während der Einschaltverzögerung oder der Impulskette wird die Ausgabesequenz abgebrochen und STS_ENABLE und der Digitalausgang DO werden gelöscht.

Danach müssen Sie die Ausgabesequenz neu starten.

Wahrheitstabelle

Die folgende Tabelle zeigt, auf welche Signale das 2PULSE wie reagiert.

Software-Freigabe SW_ENABLE	Hardware-Freigabe (Digitaleingang DI)	Digitalausgang DO	STS_ENABLE	Ausgabesequenz
1	0 → 1	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0 → 1	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0	beliebiger Zustand	0	0	abbrechen
1	beliebiger Zustand	alter Zustand bleibt		-
0 → 1	0	0	0	-

Zeiten einstellen mittels Zeitbasis

Sie wählen über die parametrierbare Zeitbasis (Seite 1182) die Auflösung und den Wertebereich der Periodendauer, Impulsdauer und der Einschaltverzögerung.

Zeitbasis = 0,1 ms:	Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
Zeitbasis = 1 ms:	Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Impulsanzahl einstellen und ändern

Die Impulsanzahl (Seite 1184) geben Sie direkt als Zahlenwert zwischen 0 und 65535 mit Ihrem Steuerungsprogramm vor.

Wenn Sie die Impulsanzahl nach Ablauf der Einschaltverzögerung ändern, dann wird der neue Wert sofort wirksam:

- haben Sie die Impulsanzahl vergrößert, wird die neue, größere Impulsanzahl ausgegeben
- haben Sie die Impulsanzahl verkleinert und ist die kleinere Impulsanzahl bereits ausgegeben, wird die Ausgabesequenz abgebrochen, STS_ENABLE und der Digitalausgang DO werden gelöscht und ERR_PULS gesetzt. Mit der nächsten Ausgabesequenz wird ERR_PULS gelöscht.

Periodendauer einstellen und ändern bei Nutzung der kurzen Steuerschnittstelle

Die Periodendauer (Seite 1183) legen Sie als Zahlenwert zwischen 2 und 65535 in den Parametern fest:

parametrierte Periodendauer = Zeitbasis × vorgegebener Zahlenwert

Mit dem Faktor für die Periodendauer können Sie die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Periodendauer = Faktor × 0,1 × parametrierte Periodendauer

Ändern Sie den Faktor während der Ausgabesequenz, wirkt die neue Periodendauer mit dem Start der nächsten Ausgabesequenz.

Impulsdauer einstellen

Die Impulsdauer (Seite 1183) legen Sie als Zahlenwert zwischen 1 und 65535 mit dem Parameter "Impulsdauer" fest:

parametrierte Impulsdauer = Zeitbasis × vorgegebener Zahlenwert

Bei Nutzung der langen Steuerschnittstelle können Sie das Verhältnis von Impulsdauer zu Impulspause einstellen:

Puls/Pause-Verhältnis = (Sollwert / 255) × Periodendauer

Einschaltverzögerung einstellen

Die Einschaltverzögerung (Seite 1182) legen Sie als Zahlenwert zwischen 0 und 65535 in den Parametern fest:

parametrierte Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Taktsynchroner Betrieb (ab 6ES7138-4DD01-0AB0)

In der Betriebsart "Impulskette" hat der taktsynchrone Betrieb keinen Einfluss auf die Funktionalität.

Strommessung (ab 6ES7138-4DD01-0AB0)

Weitere Informationen finden Sie unter "Funktion: Strommessung (ab 6ES7138-4DD01-0AB0) (Seite 1173)".

Parameter der Betriebsart "Impulskette"

Parameter der Betriebsart "Impulskette"

Die folgende Tabelle zeigt die Parameter der Betriebsart "Impulskette".

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart (Seite 1181)	Stellen Sie die Betriebsart "Impulskette" ein.	• Impulsausgabe • Pulsweitenmodulation • Impulskette • Ein-/Ausschaltverzögerung • Frequenzausgabe	Pulsweitenmodulation
Zeitbasis (Seite 1182)	Sie wählen mit der Zeitbasis die Auflösung und den Wertebereich der Periodendauer, Impulsdauer und Einschaltverzögerung.	• 0,1 ms • 1 ms	0,1 ms
Funktion DI (Seite 1182)	Den Digitaleingang DI können Sie als Eingang verwenden oder als HW-Freigabe.	• Eingang • HW-Freigabe	Eingang
Einschaltverzögerung (Seite 1182)	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulskette abläuft.	0 bis 65535	0
Impulsdauer (Seite 1183) [1)]	Impulsdauer: Hier tragen Sie die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds ein.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, setzt das 2PULSE die Impulsdauer auf 0,2 ms bzw. 1 ms.	10000 → 1 s

Parameter	Bedeutung	Wertebereich	Voreinstellung
Periodendauer (Seite 1183) [1]	Die Periodendauer sollte immer ein Vielfaches der Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds sein. Sie bestimmen die Periodendauer entsprechend der erforderlichen Wiederholrate der Impulse. Die Periodendauer können Sie in Ihrem Steuerungsprogramm mit dem "Faktor Periodendauer" ändern.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535	20000 → 2 s
[1] Nur bei Nutzung der kurzen Steuerschnittstelle (Seite 1184) Bei Nutzung der langen Steuerschnittstelle geben Sie diesen Wert direkt in der Steuerschnittstelle vor.			

Steuer- und Rückmeldesignale der Betriebsart "Impulskette"

Steuer- und Rückmeldesignale der Betriebsart "Impulskette"

Die folgende Tabelle zeigt die Steuer- und Rückmeldesignale der Betriebsart "Impulskette".

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Steuersignale bei Nutzung der kurzen Steuerschnittstelle (8 Byte)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der HW-Freigabe	Byte 2: Bit 0	Byte 6: Bit 0
Impulsanzahl	Anzahl Impulse, die am Digitalausgang DO nach Ablauf der Einschaltverzögerung ausgegeben werden.	0 bis 65535 bei Impulsanzahl 0 gibt das 2PULSE keinen Impuls aus. Die Ausgabesequenz wird mit ERR_PULS = 1 beendet.	Word 0	Word 4
Faktor Periodendauer	Sie können die parametrierte Periodendauer vor dem Start der Ausgabesequenz ändern: Periodendauer = Faktor × 0,1 × parametrierte Periodendauer	Faktor: 0 bis 255 Periodendauer: > Impulsdauer bis 65,535 s Ergibt sich eine Periodendauer > 65,535 s, wird diese auf 65,535 s gesetzt. Ergibt sich eine Periodendauer ≤ Impulsdauer, wird diese auf Impulsdauer + 0,2 ms (bzw. Impulsdauer + 1 ms) gesetzt.	Byte 3	Byte 7
Steuersignale bei Nutzung der langen Steuerschnittstelle (12 Byte) (ab 6ES7138-4DD01-0AB0)				

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der HW-Freigabe	Byte 4: Bit 0	Byte 10: Bit 0
Impulsanzahl	Anzahl Impulse, die am Digitalausgang DO nach Ablauf der Einschaltverzögerung ausgegeben werden.	0 bis 65535 bei Impulsanzahl 0 gibt das 2PULSE keinen Impuls aus. Die Ausgabesequenz wird mit ERR_PULS = 1 beendet.	Word 0	Word 6
Periodendauer	Sollwert der Periodendauer	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535	Word 2	Word 8
Puls/Pause-Verhältnis	Sie können das Verhältnis von Impulsdauer zu Impulspause einstellen	0 bis 255 Z. B. entspricht der Wert 128 einem Verhältnis von 50 : 50.	Byte 5	Byte 11
Rückmeldesignale				
STS_ENABLE	Zeigt eine laufende Ausgabesequenz.	0 = Impulsausgabe gesperrt 1 = Impulsausgabe läuft	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Zeigt den Zustand von SW_ENABLE an.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Zeigt einen Impulsausgabefehler an.	0 = kein Impulsausgabefehler 1 = Impulsausgabefehler	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (ab 6ES7138-4DD01-0AB0)	Wird nach Abschluss einer Ausgabesequenz inkrementiert.	Ohne HW-Freigabe: 0 … 1 Mit HW-Freigabe: 0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Strommesswert (ab 6ES7138-4DD01-0AB0)	Strom am DO als SIMATIC S7-Analogwert.	0 … 27648 (Übersteuerbereich bis 32767)	Word 2	Word 6

Ein- und Ausgangssignale der Betriebsart "Impulskette"

Ein- und Ausgangssignale der Betriebsart "Impulskette"

Die folgende Tabelle zeigt die Ein- und Ausgangssignale der Betriebsart "Impulskette".

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Eingangssignal				
HW-Freigabe	Die HW-Freigabe können Sie mit dem Parameter "Funktion DI" wählen. Das Signal des Digitaleingangs DI wird dann vom 2PULSE zum Start ausgewertet.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe erteilt 0 → 1 = Start der Ausgabesequenz; abhängig von der Software-Freigabe (SW_ENABLE)	1	5
Ausgangssignal				
Impulskette am Digitalausgang DO	Die vorgegebene Impulsanzahl wird am Digitalausgang DO ausgegeben.	0 = kein Impuls 1 = Impuls	4	8

Betriebsart "Ein-/Ausschaltverzögerung"

Betriebsart "Ein-/Ausschaltverzögerung"

Definition

Das am Digitaleingang DI anstehende Signal wird vom 2PULSE ein- und ausschaltverzögert am Digitalausgang DO ausgegeben.

Das folgende Bild zeigt das Prinzipschaltbild für die Betriebsart "Ein-/Ausschaltverzögerung".

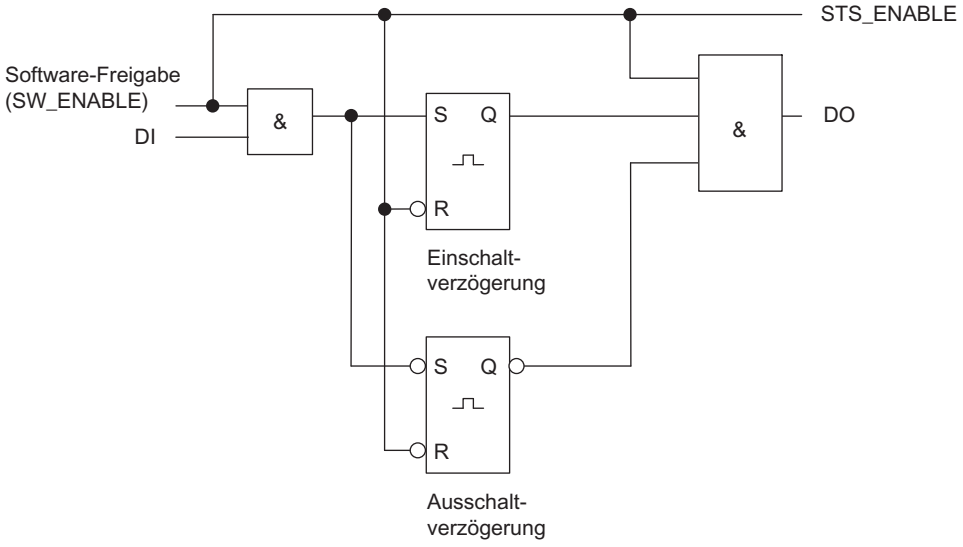

Freigabe der Ausgabesequenz

Die Freigabe für die Ausgabesequenz müssen Sie immer per Software-Freigabe (Seite 1184) (SW_ENABLE 0 → 1; MANUAL_DO = 0) in Ihrem Steuerungsprogramm erteilen, STS_ENABLE ist dadurch gesetzt. Das Rückmeldebit (Seite 1187) ACK_SW_ENABLE zeigt die am 2PULSE anstehende Software-Freigabe an.

Mit der positiven Flanke am Digitaleingang DI (0 → 1) wird die Einschaltverzögerung gestartet und nach Ablauf der Einschaltverzögerung der Digitalausgang DO (Seite 1172) gesetzt.

Mit der negativen Flanke am Digitaleingang DI (1 → 0) wird die Ausschaltverzögerung gestartet und nach Ablauf der Ausschaltverzögerung der Digitalausgang DO (Seite 1172) gelöscht.

Erkennt das 2PULSE eine zu kurze Impulsdauer oder Impulspause, wird das mit dem Impulsausgabefehler (Seite 1187) ERR_PULS angezeigt.

Mit der nächsten Flanke am Digitaleingang DI löscht das 2PULSE das Rückmeldebit ERR_PULS.

Impulsschema

SW_ENABLE wird gesetzt, während DI = 0 ist:

Das folgende Bild zeigt eine Ausgabesequenz in der Betriebsart "Ein-/Ausschaltverzögerung", wenn beim Start DI = 0 ist.

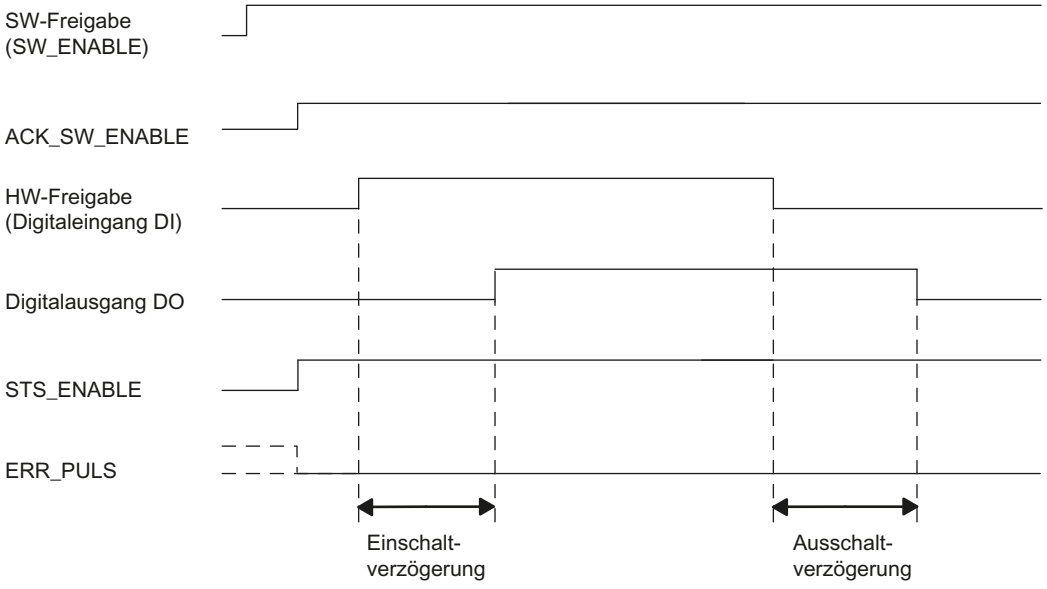

SW_ENABLE wird gesetzt, während DI = 1 ist:

Wenn SW_ENABLE gesetzt wird, während DI = 1 ist, dann wird die erste Flanke an DI (fallende Flanke) ignoriert. Das folgende Bild zeigt die entsprechende Ausgabesequenz.

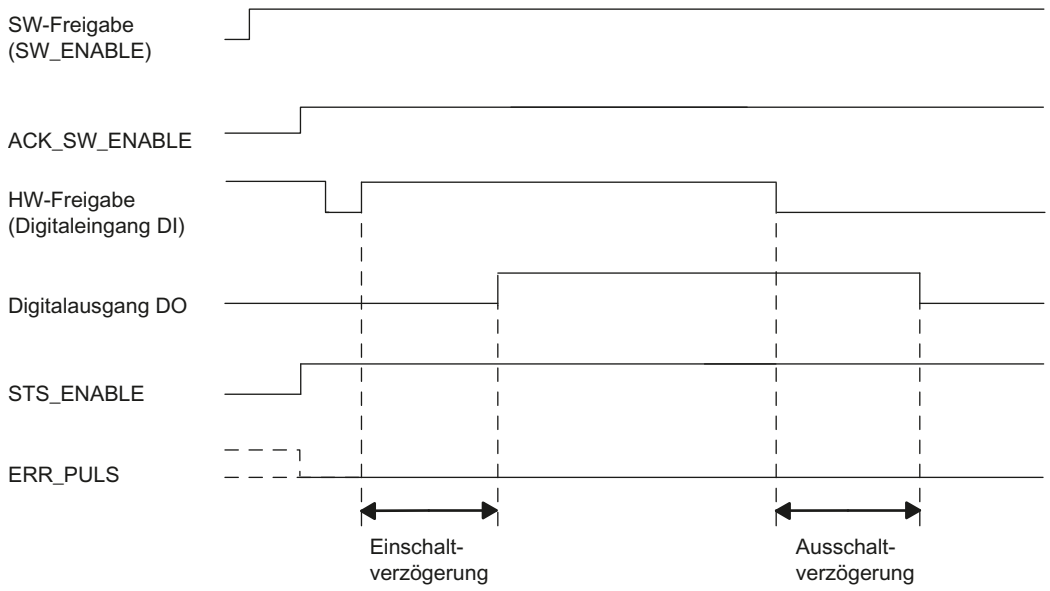

Abbrechen der Ausgabesequenz

Durch Löschen der Software-Freigabe (SW_ENABLE = 0) während der Ausgabesequenz wird diese abgebrochen und STS_ENABLE und der Digitalausgang werden gelöscht.

Wahrheitstabelle

Die folgende Tabelle zeigt, auf welche Signale das 2PULSE wie reagiert.

Software-Freigabe SW_ENABLE	Digitaleingang DI	Digitalausgang DO	STS_ENABLE	Ausgabesequenz
1	0 → 1	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	1	starten
1	1 → 0	1, wenn Ausschaltverzögerung > 0 0, wenn Ausschaltverzögerung = 0	1	starten
0	beliebiger Zustand	0	0	abbrechen
1	beliebiger Zustand	alter Zustand bleibt	1	-
0 → 1	0	0	1	-

Mindestimpulsdauer/-pause des Digitalausgangs DO

Die Mindestimpulsdauer/-pause des Digitalausgangs DO beträgt 100 µs.

Beachten Sie dies bei der Einstellung der Ein- und Ausschaltverzögerung und der Impulsdauer/-pause des Digitaleingangs DI, sonst ist die Reaktion am Digitalausgang DO nicht definiert.

Impulsdauer des Digitaleingangs DI zu kurz

Das 2PULSE erkennt bei der negativen Flanke am Digitaleingang DI einen zu kurzen Impuls, wenn:

Impulsdauer + Ausschaltverzögerung ≤ Einschaltverzögerung.

Reaktion des 2PULSE auf die zu kurze Impulsdauer:

- ERR_PULS wird gesetzt
- die laufende Einschaltverzögerung wird gelöscht
- die Ausschaltverzögerung wird nicht gestartet
- der Pegel am Digitalausgang DO bleibt 0

ERR_PULS wird mit der nächsten positiven Flanke am Digitaleingang DI gelöscht.

Das folgende Bild zeigt das Verhalten, wenn die Impulsdauer zu kurz ist.

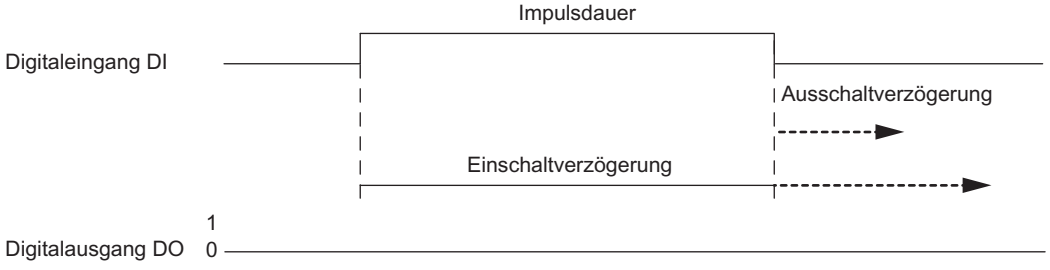

Impulspause des Digitaleingangs DI zu kurz

Das 2PULSE erkennt bei der positiven Flanke am Digitaleingang DI eine zu kurze Impulspause, wenn:

Impulspause + Einschaltverzögerung ≤ Ausschaltverzögerung.

Reaktion des 2PULSE auf die zu kurze Impulspause:

- ERR_PULS wird gesetzt
- die laufende Ausschaltverzögerung wird gelöscht
- die Einschaltverzögerung wird nicht gestartet
- der Pegel am Digitalausgang DO bleibt 1

ERR_PULS wird mit der nächsten negativen Flanke am Digitaleingang DI gelöscht.

Das folgende Bild zeigt das Verhalten, wenn die Impulspause zu kurz ist.

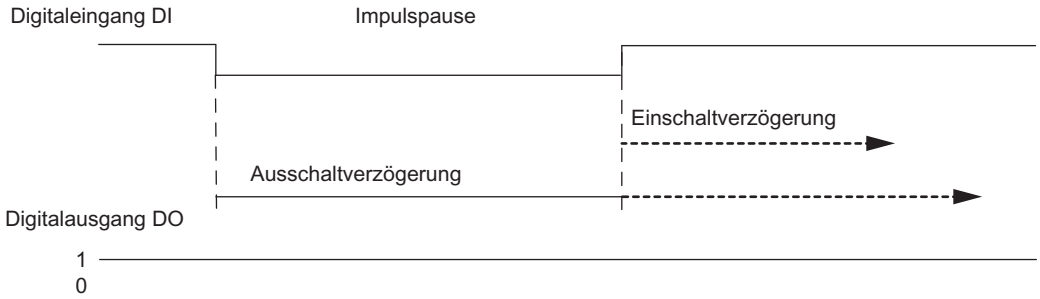

Nachtriggern der laufenden Einschaltverzögerung

Das 2PULSE startet bei der positiven Flanke am Digitaleingang DI die Einschaltverzögerung neu, wenn:

Einschaltverzögerung > Impulsdauer + Impulspause.

Die laufende Ausschaltverzögerung wird dabei gelöscht.

Der Digitalausgang DO wird erst gesetzt, wenn am Digitaleingang DI der Signalpegel 1 länger als die Einschaltverzögerung ansteht. Dadurch können Sie schnelle Impulsfolgen filtern.

Das folgende Bild zeigt das Nachtriggern der laufenden Einschaltverzögerung.

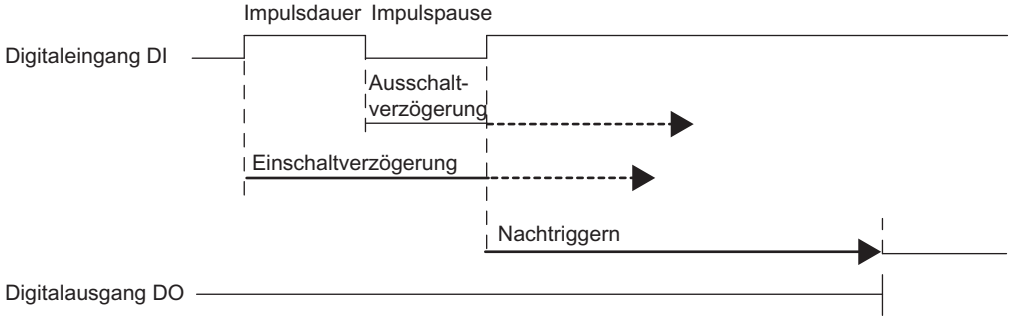

Nachtriggern der laufenden Ausschaltverzögerung

Das 2PULSE startet bei der negativen Flanke am Digitaleingang DI die Ausschaltverzögerung neu, wenn:

Ausschaltverzögerung > Impulsdauer + Impulspause.

Die laufende Einschaltverzögerung wird dabei gelöscht.

Der Digitalausgang DO wird erst gelöscht, wenn am Digitaleingang DI der Signalpegel 0 länger als die Ausschaltverzögerung ansteht.

Das folgende Bild zeigt das Nachtriggern der laufenden Ausschaltverzögerung.

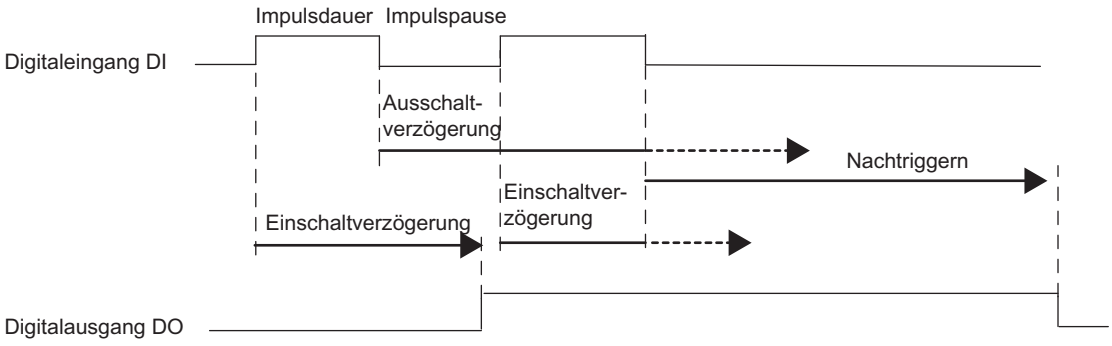

Zeiten einstellen mittels Zeitbasis

Sie wählen über die parametrierbare Zeitbasis (Seite 1182) die Auflösung und den Wertebereich der Einschaltverzögerung und der Ausschaltverzögerung.

Zeitbasis = 0,1 ms:	Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
Zeitbasis = 1 ms:	Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Einschaltverzögerung einstellen und ändern bei Nutzung der kurzen Steuerschnittstelle

Die Einschaltverzögerung (Seite 1182) legen Sie als Zahlenwert zwischen 0 und 65535 in den Parametern fest:

parametrierte Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Mit dem Faktor für die Einschaltverzögerung können Sie die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung

Wenn Sie den Faktor der Einschaltverzögerung ändern, wird die neue Einschaltverzögerung mit der nächsten positiven Flanke am Digitaleingang DI wirksam.

Ausschaltverzögerung einstellen und ändern

Die Ausschaltverzögerung geben Sie direkt als Zahlenwert zwischen 0 und 65535 in Ihrem Steuerungsprogramm (Seite 1184) vor:

Ausschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Wenn Sie die Ausschaltverzögerung ändern, wird die neue Ausschaltverzögerung mit der nächsten negativen Flanke am Digitaleingang DI wirksam.

Taktsynchroner Betrieb (ab 6ES7138-4DD01-0AB0)

In der Betriebsart "Ein-/Ausschaltverzögerung" hat der taktsynchrone Betrieb keinen Einfluss auf die Funktionalität.

Parameter der Betriebsart "Ein-/Ausschaltverzögerung"

Parameter der Betriebsart "Ein-/Ausschaltverzögerung"

Die folgende Tabelle zeigt die Parameter der Betriebsart "Ein-/Ausschaltverzögerung".

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart (Seite 1181)	Stellen Sie die Betriebsart "Ein-/Ausschaltverzögerung" ein.	• Impulsausgabe • Pulsweitenmodulation • Impulskette • Ein-/Ausschaltverzögerung • Frequenzausgabe	Pulsweitenmodulation
Zeitbasis (Seite 1182)	Sie wählen mit der Zeitbasis die Auflösung und den Wertebereich der Ein- und Ausschaltverzögerung.	• 0,1 ms • 1 ms	0,1 ms
Einschaltverzögerung (Seite 1182) [1]	Zeit zwischen positiver Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO. Die Einschaltverzögerung können Sie in Ihrem Steuerungsprogramm mit dem "Faktor Einschaltverzögerung" ändern.	0 bis 65535	0
[1] Nur bei Nutzung der kurzen Steuerschnittstelle (Seite 1184) Bei Nutzung der langen Steuerschnittstelle geben Sie diesen Wert direkt in der Steuerschnittstelle vor.			

Steuer- und Rückmeldesignale der Betriebsart "Ein-/Ausschaltverzögerung"

Steuer- und Rückmeldesignale der Betriebsart "Ein-/Ausschaltverzögerung"

Die folgende Tabelle zeigt die Steuer- und Rückmeldesignale der Betriebsart "Ein-/Ausschaltverzögerung".

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Steuersignale bei Nutzung der kurzen Steuerschnittstelle (8 Byte)				
Software-Freigabe (SW_ENABLE)	Die Software-Freigabe müssen Sie immer in Ihrem Steuerungsprogramm erteilen. Löschen Sie die Software-Freigabe, wird die laufende Ausgabesequenz abgebrochen.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 2: Bit 0	Byte 6: Bit 0
Ausschaltverzögerung	Zeit zwischen negativer Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, wirkt die Ausschaltverzögerung nicht.	Word 0	Word 4

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Faktor Einschaltverzögerung	Sie können die parametrierte Einschaltverzögerung ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung	Faktor: 0 bis 255 Einschaltverzögerung: 0,2 ms bis 65,535 s Wenn sich eine Einschaltverzögerung < 0,2 ms ergibt oder der Faktor = 0 ist, dann ist die wirksame Einschaltverzögerung = 0. Wenn sich eine Einschaltverzögerung > 65,535 s ergibt, dann wird diese auf 65,535 s begrenzt.	Byte 3	Byte 7
Steuersignale bei Nutzung der langen Steuerschnittstelle (12 Byte) (ab 6ES7138-4DD01-0AB0)				
Software-Freigabe (SW_ENABLE)	Die Software-Freigabe müssen Sie immer in Ihrem Steuerungsprogramm erteilen. Löschen Sie die Software-Freigabe, wird die laufende Ausgabesequenz abgebrochen.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 4: Bit 0	Byte 10: Bit 0
Ausschaltverzögerung	Zeit zwischen negativer Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, wirkt die Ausschaltverzögerung nicht.	Word 0	Word 6
Einschaltverzögerung	Zeit zwischen positiver Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO.	bei Zeitbasis 0,1 ms: 2 bis 65535 bei Zeitbasis 1 ms: 1 bis 65535 Wenn Sie den Wertebereich unterschreiten, wirkt die Einschaltverzögerung nicht.	Word 2	Word 8
Rückmeldesignale				
STS_ENABLE	Zeigt den Zustand der Software-Freigabe (SW_ENABLE) an.	0 = Software-Freigabe gesperrt 1 = Software-Freigabe erteilt	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Zeigt den Zustand von SW_ENABLE an.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Zeigt bei zu kurzer Impulsdauer oder Impulspause einen Impulsausgabefehler an.	0 = kein Impulsausgabefehler 1 = Impulsausgabefehler	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (ab 6ES7138-4DD01-0AB0)	Wird mit jeder Flanke (positiv und negativ) am DO inkrementiert.	0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reserviert	–	0	Word 2	Word 6

Ein- und Ausgangssignale der Betriebsart "Ein-/Ausschaltverzögerung"

Ein- und Ausgangssignale der Betriebsart "Ein-/Ausschaltverzögerung"

Die folgende Tabelle zeigt die Ein- und Ausgangssignale der Betriebsart "Ein-/Ausschaltverzögerung".

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Eingangssignal				
Digitaleingang DI	Das Signal des Digitaleingangs DI wird vom 2PULSE ein- und ausschaltverzögert am Digitalausgang DO ausgegeben.	0 = kein Impuls 1 = Impuls	1	5
Ausgangssignal				
Impuls am Digitalausgang DO	Das Signal des Digitaleingangs DI wird vom 2PULSE ein- und ausschaltverzögert am Digitalausgang DO ausgegeben.	0 = kein Signal 1 = Signal	4	8

Betriebsart "Frequenzausgabe" (ab 6ES7138-4DD01-0AB0)

Betriebsart "Frequenzausgabe"

Definition

Am Digitalausgang des 2PULSE wird ein Rechtecksignal mit einer vorgegebenen Frequenz ausgegeben.

Die Ausgabesequenz wird nach Ablauf der parametrierten Einschaltverzögerung (Seite 1182) am Digitalausgang DO des 2PULSE gestartet.

Das folgende Bild zeigt das Prinzipschaltbild für die Betriebsart "Frequenzausgabe".

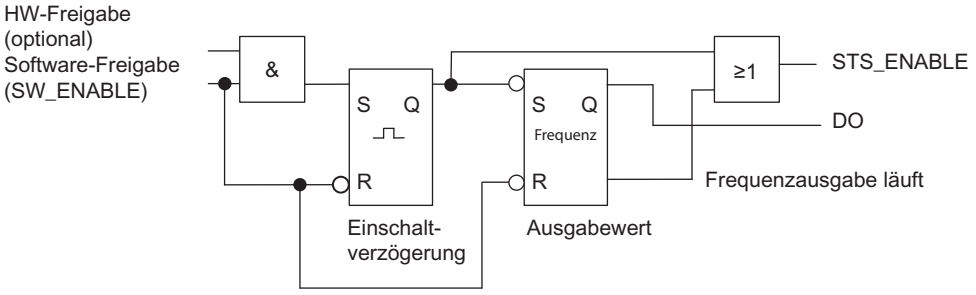

Starten der Ausgabesequenz

Die Freigabe für die Ausgabesequenz müssen Sie immer per Software-Freigabe (Seite 1184) (SW_ENABLE 0 → 1; MANUAL_DO = 0) in Ihrem Steuerungsprogramm erteilen.

Das Rückmeldebit (Seite 1187) ACK_SW_ENABLE zeigt die am 2PULSE anstehende Software-Freigabe an.

Zusätzlich können Sie mit dem Parameter "Funktion DI (Seite 1182)" den Digitaleingang DI des 2PULSE als HW-Freigabe einstellen.

Wenn Sie mit Software-Freigabe und Hardware-Freigabe gleichzeitig arbeiten, startet die Ausgabesequenz bei erteilter Software-Freigabe mit der ersten positiven Flanke der Hardware-Freigabe. Weitere positive Flanken der Hardware-Freigabe während der laufenden Ausgabesequenz werden vom 2PULSE ignoriert.

Durch Erteilen der Freigabe (positive Flanke) wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Nach Ablauf der Einschaltverzögerung wird die Frequenzausgabe gestartet. Die Ausgabesequenz läuft endlos, so lange SW_ENABLE gesetzt ist.

Impulsschema

Das folgende Bild zeigt eine Ausgabesequenz in der Betriebsart "Frequenzausgabe".

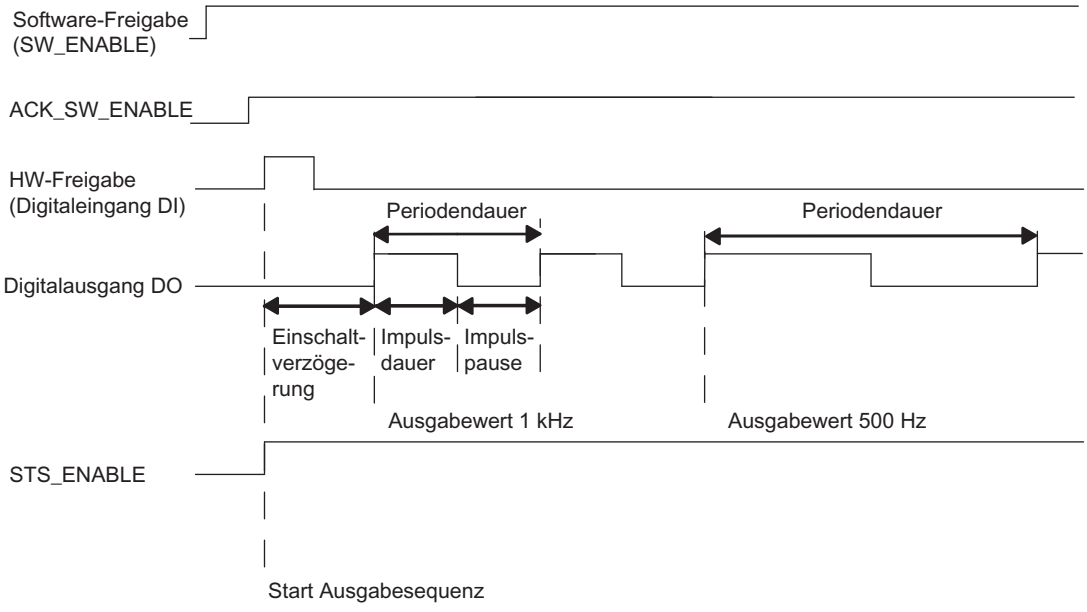

Abbrechen der Ausgabesequenz

Durch Löschen der Software-Freigabe (SW_ENABLE = 0) während der Einschaltverzögerung oder der Frequenzausgabe wird die Ausgabesequenz abgebrochen und STS_ENABLE und der Digitalausgang DO werden gelöscht.

Danach müssen Sie die Ausgabesequenz erneut starten.

Wahrheitstabelle

Die folgende Tabelle zeigt, auf welche Signale das 2PULSE wie reagiert.

Software-Freigabe SW_ENABLE	Hardware-Freigabe (Digitaleingang DI)	Digitalausgang DO	STS_ENABLE	Ausgabesequenz
1	0 → 1	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0 → 1	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	starten
0	beliebiger Zustand	0	0	abbrechen
1	beliebiger Zustand	alter Zustand bleibt		-
0 → 1	0	0	0	-

Zeiten einstellen mittels Zeitbasis

Sie wählen über die parametrierbare Zeitbasis (Seite 1182) die Auflösung und den Wertebereich der Einschaltverzögerung.

Zeitbasis = 0,1 ms:	Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
Zeitbasis = 1 ms:	Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Ausgabewert einstellen und ändern

Sie wählen mit dem Parameter "Ausgabeformat (Seite 1182)" den Wertebereich des Ausgabewertes.

- Liegt Ihr Ausgabewert zwischen 1 Hz und 5000 Hz, wählen Sie das Ausgabeformat "1 Hz".
- Ist Ihr Ausgabewert ein SIMATIC S7-Analogwert (zwischen 0 und 27648), wählen Sie das Ausgabeformat "S7-Analogausgabe". Mit "S7-Analogausgabe" erreichen Sie eine höhere Genauigkeit bei der Sollwertvorgabe.
  Das 2PULSE berechnet die Frequenz nach folgender Formel:
  Frequenz = (Sollwert / 27648) × 5 kHz

Den Ausgabewert geben Sie direkt mit Ihrem Steuerungsprogramm vor. Der neue Frequenzwert wird mit der nächsten steigenden Flanke der Ausgabe ausgegeben.

Genauigkeit der Ausgabefrequenz

Die parametrierte Ausgabefrequenz wird mit folgenden Genauigkeiten am Digitalausgang DO ausgegeben:

- 0,01 Hz bei Ausgabefrequenzen von 0,1 Hz bis 800 Hz
- 0,1 Hz bei Ausgabefrequenzen > 800 Hz bis 2000 Hz
- 0,5 Hz bei Ausgabefrequenzen > 2000 Hz bis 5000 Hz

Einschaltverzögerung einstellen bei Nutzung der kurzen Steuerschnittstelle

Die Einschaltverzögerung (Seite 1182) legen Sie als Zahlenwert zwischen 0 und 65535 in den Parametern fest.

parametrierte Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert

Bei Verwendung der kurzen Steuerschnittstelle können Sie mit dem Faktor für die Einschaltverzögerung die parametrierte Zeit in Ihrem Steuerungsprogramm anpassen. Den Faktor stellen Sie zwischen 0 und 255 ein, er wirkt mit der Gewichtung 0,1:

Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung

Taktsynchroner Betrieb

In der Betriebsart "Frequenzausgabe" hat der taktsynchrone Betrieb keinen Einfluss auf die Funktionalität.

Parameter der Betriebsart "Frequenzausgabe"

Parameter der Betriebsart "Frequenzausgabe"

Die folgende Tabelle zeigt die Parameter der Betriebsart "Frequenzausgabe".

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart (Seite 1181)	Stellen Sie die Betriebsart "Frequenzausgabe" ein.	• Impulsausgabe • Pulsweitenmodulation • Impulskette • Ein-/Ausschaltverzögerung • Frequenzausgabe	Pulsweitenmodulation
Zeitbasis (Seite 1182)	Sie wählen mit der Zeitbasis die Auflösung und den Wertebereich der Einschaltverzögerung.	• 0,1 ms • 1 ms	0,1 ms
Ausgabeformat (Seite 1182)	Je nach benötigter Auflösung der Ausgabefrequenz wählen Sie zwischen 1 Hz oder S7-Analogausgabe.	• 1 Hz • S7-Analogausgabe	1 Hz
Funktion DI (Seite 1182)	Den Digitaleingang DI können Sie als Eingang verwenden oder als HW-Freigabe.	• Eingang • HW-Freigabe	Eingang
Einschaltverzögerung (Seite 1182) [1]	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulsfolge abläuft. Die Einschaltverzögerung können Sie in Ihrem Steuerungsprogramm mit dem "Faktor Einschaltverzögerung" ändern.	0 bis 65535	0

[1] Nur bei Nutzung der kurzen Steuerschnittstelle (Seite 1184)
Bei Nutzung der langen Steuerschnittstelle geben Sie diesen Wert direkt in der Steuerschnittstelle vor.

Steuer- und Rückmeldesignale der Betriebsart "Frequenzausgabe"

Steuer- und Rückmeldesignale der Betriebsart "Frequenzausgabe"

Die folgende Tabelle zeigt die Steuer- und Rückmeldesignale der Betriebsart "Frequenzausgabe".

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
Steuersignale bei Nutzung der kurzen Steuerschnittstelle (8 Byte)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 2: Bit 0	Byte 6: Bit 0
Ausgabewert	Wert, der am Digitalausgang DO als Frequenz ausgegeben wird.	je nach Ausgabeformat: • 1 Hz: 0 ... 5000 • S7-Analogausgabe: 0 ... 27648 Geben Sie einen Ausgabewert > 5000 bzw. 27648 vor, begrenzt das 2PULSE diesen auf 5000 bzw. 27648.	Word 0	Word 4
Faktor Einschaltverzögerung	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung	0 bis 255 Bei Einschaltverzögerung < 0,2 ms oder bei Faktor = 0 ist die wirksame Einschaltverzögerung = 0. Bei Einschaltverzögerung > 65,535 s wird die Einschaltverzögerung auf 65,535 s begrenzt.	Byte 3	Byte 7
Steuersignale bei Nutzung der langen Steuerschnittstelle (12 Byte)				
Software-Freigabe (SW_ENABLE)	Starten und Abbrechen der Ausgabesequenz.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt 0 → 1 = Start der Ausgabesequenz; ggf. abhängig von der Hardware-Freigabe	Byte 4: Bit 0	Byte 10: Bit 0
Ausgabewert	Wert, der am Digitalausgang DO als Frequenz ausgegeben wird.	je nach Ausgabeformat: • 1 Hz: 0 ... 5000 • S7-Analogausgabe: 0 ... 27648 Geben Sie einen Ausgabewert > 5000 bzw. 27648 vor, begrenzt das 2PULSE diesen auf 5000 bzw. 27648.	Word 0	Word 6
Einschaltverzögerung	Sie können die Einschaltverzögerung vor dem Start der Ausgabesequenz ändern.	0 bis 65535 Einschaltverzögerung = Zeitbasis × vorgegebener Zahlenwert	Word 2	Word 8
Rückmeldesignale				
STS_ENABLE	Zeigt eine laufende Ausgabesequenz an.	0 = Impulsausgabe gesperrt 1 = Impulsausgabe läuft	Byte 0: Bit 0	Byte 4: Bit 0

Steuer- und Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Zeigt den Zustand von SW_ENABLE an.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 0: Bit 3	Byte 4: Bit 3
SEQ_CNT	Keine Funktion in dieser Betriebsart.	–	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reserviert	–	0	Word 2	Word 6

Ein- und Ausgangssignale der Betriebsart "Frequenzausgabe"

Ein- und Ausgangssignale der Betriebsart "Frequenzausgabe"

Die folgende Tabelle zeigt die Ein- und Ausgangssignale der Betriebsart "Frequenzausgabe".

Ein- und Ausgangssignal	Bedeutung	Wertebereich	Klemme Kanal 0	Klemme Kanal 1
Eingangssignal				
HW-Freigabe	Die HW-Freigabe können Sie mit dem Parameter "Funktion DI" wählen. Das Signal des Digitaleingangs DI wird dann vom 2PULSE zum Start der Ausgabesequenz ausgewertet.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe erteilt 0 → 1 = Start der Ausgabesequenz; abhängig von der Software-Freigabe	1	5
Ausgangssignal				
Frequenz am Digitalausgang DO	Die Frequenz wird am Digitalausgang DO ausgegeben.	0 = kein Impuls 1 = Impuls	4	8

Sequenzzähler (ab 6ES7138-4DD01-0AB0)

Das 2PULSE hat für jeden Kanal einen Sequenzzähler, mit dem die abgeschlossenen Ausgabesequenzen gezählt werden.
Den Abschluss einer Ausgabesequenz können Sie mit dem Sequenzzähler SEC_CNT in der Rückmeldeschnittstelle überwachen.

Der Zähler hat eine Breite von 4 Bit. Nach einem Überlauf springt er wieder auf 0 zurück.

Funktion des Sequenzzählers

In den einzelnen Betriebsarten hat der Sequenzzähler folgende Funktion:

- Impulsausgabe (Seite 1137) und Impulskette (Seite 1153)
  - Ohne Hardware-Freigabe wird der Zähler nach Abschluss der Ausgabesequenz auf 1 gesetzt.
  - Mit Hardware-Freigabe wird der Zähler nach jeder abgeschlossenen Ausgabesequenz inkrementiert.
- Ein-/Ausschaltverzögerung (Seite 1159)
  Der Zähler wird mit jeder Flanke (positiv und negativ) am DO inkrementiert.
- Pulsweitenmodulation und Frequenzausgabe
  Der Zähler hat keine Funktion.

Mit Löschen der Software-Freigabe (SW_ENABLE = 0) wird der Zähler auf 0 zurückgesetzt.

Verwendungsmöglichkeiten

Der Sequenzzähler kann insbesondere verwendet werden für:

- das Erfassen (Zählen) von sehr kurzen Impulsen
- das Zählen von Ausgabesequenzen in Verbindung mit der Hardware-Freigabe

Funktion: Strommessung (ab 6ES7138-4DD01-0AB0)

Die Strommessung ist in einer Vielzahl von Anwendungen zu Regel- und Diagnosezwecken einsetzbar. Den Strommesswert können Sie in der Rückmeldeschnittstelle auslesen.

Funktionsweise

Eine Strommessung findet nur während der Impulsausgabe in den Betriebsarten "Pulsweitenmodulation (Seite 1143)" und "Impulskette (Seite 1153)" statt. Ohne Impulsausgabe (bei STS_ENABLE = 0) wird 0 als Messwert geliefert. Bei STS_ENABLE = 1 ist der Messwert gültig.

In den Betriebsarten "Impulsausgabe", "Ein-/Ausschaltverzögerung" und "Frequenzausgabe" wird als Strommesswert immer 0 geliefert.

Der zurückgelieferte Strommesswert ist ein Mittelwert aus Messwerten, die während der Zeitdauer einer Periode aufgenommen werden. Bei Periodendauern < 5 ms beträgt der zeitliche Abstand zwischen zwei einzelnen Messpunkten höchstens 40 µs. Bei längeren Periodendauern werden immer 128 Messpunkte pro Periode aufgenommen, dadurch entstehen entsprechend längere zeitliche Abstände zwischen den Messpunkten.

Im taktsynchronen Betrieb wird immer zum Zeitpunkt T_i ein neuer Strommesswert in der Rückmeldeschnittstelle bereitgestellt. Im nicht taktsynchronen Betrieb wird ein neuer Strommesswert etwa alle 500 µs bereitgestellt.

Hinweis

Für die korrekte Funktion der Strommessung dürfen Sie keine Freilaufdiode an der Last anbringen.

Messbereich und Messwert

Der Messbereich ist fest eingestellt auf einen Nennstrom von 2 A.

Der Strommesswert wird in der Rückmeldeschnittstelle (Seite 1187) als SIMATIC S7-Analogwert geliefert.

- Ein Strom von 2 A entspricht dem SIMATIC S7-Analogwert 27648.
- Bei Parallelschaltung entspricht ein Strom von 4 A dem SIMATIC S7-Analogwert 27648.

Strommessungen sind bis zu einem SIMATIC S7-Analogwert 32767 möglich. Das entspricht einem Strom von 2,4 A. Ströme über 2 A dürfen nur kurzzeitig auftreten.

Genauigkeit

Die Genauigkeit der Strommessung beträgt 3 % vom Messbereich.

Hinweis

Diese Genauigkeit bei der Strommessung erreichen Sie nur, wenn Sie während der Messung die Periodendauer nicht verändern.

Funktion: Direktes Steuern des Digitalausgangs DO

Definition

Zum Testen Ihres angeschlossenen Aktors können Sie den Digitalausgang DO des 2PULSE direkt steuern. Dazu müssen Sie die Funktion über Ihr Steuerungsprogramm (Seite 1184) mit gesetztem Steuerbit MANUAL_DO und gelöschtem Steuerbit SW_ENABLE anwählen. Die Funktion

- unterstützt Sie beim Inbetriebnehmen einer Anlage
- ermöglicht Ihnen die Nutzung eines nicht verwendeten Kanals des 2PULSE als DO

Wenn Sie die Funktion angewählt haben, werden vom 2PULSE die Rückmeldebits (Seite 1187) STS_ENABLE und ERR_PULS gelöscht und eine laufende Ausgabesequenz abgebrochen. Die Funktion hat Vorrang gegenüber der aktuell ausgewählten Betriebsart.

Den Zustand des Digitalausgangs DO geben Sie mit dem Steuerbit (Seite 1184) SET_DO vor.

Durch Löschen des Steuerbits MANUAL_DO wählen Sie die Funktion "Direktes Steuern des Digitalausgangs DO" ab. Damit wird der Digitalausgang DO gelöscht. Danach können Sie mit der steigenden Flanke von SW_ENABLE eine Ausgabesequenz neu starten.

Steuer- und Rückmeldesignale / Ausgangssignal

Signale	Bedeutung	Wertebereich	Bei Nutzung der kurzen Steuerschnittstelle (8 Byte)		Bei Nutzung der langen Steuerschnittstelle (12 Byte)	
			Adresse Kanal 0	Adresse Kanal 1	Adresse Kanal 0	Adresse Kanal 1
Steuersignale						
SW_ENABLE	Zur Anwahl der Funktion muss das Steuerbit gelöscht sein.	0 = SW_ENABLE gelöscht 1 = SW_ENABLE gesetzt	Byte 2: Bit 0	Byte 6: Bit 0	Byte 4: Bit 0	Byte 10: Bit 0
MANUAL_DO	Mit dem Steuerbit wählen Sie die Funktion an und ab.	0 = Direktes Steuern des DO nicht angewählt. 1 = Direktes Steuern des DO angewählt.	Byte 2: Bit 1	Byte 6: Bit 1	Byte 4: Bit 1	Byte 10: Bit 1
SET_DO	Mit dem Steuerbit geben Sie den Zustand des Digitalausgangs DO vor.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 2: Bit 2	Byte 6: Bit 2	Byte 4: Bit 2	Byte 10: Bit 2
Rückmeldesignale						
STS_ENABLE	Ist nach Anwahl der Funktion gelöscht.	0 = Impulsausgabe gesperrt 1 = Impulsausgabe läuft	Byte 0: Bit 0	Byte 4: Bit 0	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an. Beachten Sie dabei die Aktualisierungsrate.	0 = Signal 0 am Digitalausgang DO 1 = Signal 1 am Digitalausgang DO	Byte 0: Bit 1	Byte 4: Bit 1	Byte 0: Bit 1	Byte 4: Bit 1
STS_ DI	Zeigt den Signalpegel am Digitaleingang DI an.	0 = Signal 0 am Digitaleingang DI 1 = Signal 1 am Digitaleingang DI	Byte 0: Bit 2	Byte 4: Bit 2	Byte 0: Bit 2	Byte 4: Bit 2
Ausgangssignal						
Digitalausgang DO	Der mit dem Steuerbit SET_DO vorgegebene Zustand wird am Digitalausgang DO ausgegeben.	0 = kein Signal 1 = Signal	4	8	4	8

Funktion: Fehlererkennung/Diagnose

Parametrierfehler ERR_PARA

Kann das 2PULSE die Parameter nicht als eigene identifizieren, erzeugt es einen Parametrierfehler (Seite 1187). Beide Kanäle sind damit nicht parametriert.

Der von Ihnen projektierte Steckplatz des 2PULSE muss mit dem Aufbau übereinstimmen.

Impulsausgabefehler ERR_PULS

Das 2PULSE erkennt kanalspezifisch in den Betriebsarten "Impulsausgabe (Seite 1137)", "Ein-/Ausschaltverzögerung (Seite 1159)" und "Impulskette (Seite 1153)" einen Impulsausgabefehler.

Ursachen und Reaktionen finden Sie in der jeweiligen Betriebsartenbeschreibung und unter "Rückmeldeschnittstelle des 2PULSE (Seite 1187)".

Der erkannte Impulsausgabefehler wird für den betroffenen Kanal mit dem Rückmeldebit (Seite 1187) ERR_PULS angezeigt.

Kurzschluss Geberversorgung ERR_24V

Das 2PULSE erkennt einen Kurzschluss der Geberversorgung, die es an den Klemmen 2 und 6 zur Verfügung stellt.

Der erkannte Kurzschluss wird für beide Kanäle jeweils mit dem Rückmeldebit (Seite 1187) ERR_24V angezeigt.

Ein Einbruch der 24 V-Versorgungsspannung kann ebenfalls zu einer Kurzschlussmeldung führen.

Kurzschluss Digitalausgang ERR_DO

Das 2PULSE erkennt einen Kurzschluss am Digitalausgang des Kanals.

Der erkannte Kurzschluss wird für den betroffenen Kanal mit dem Rückmeldebit (Seite 1187) ERR_DO angezeigt.

Hinweis

Bei einem Kurzschluss ist der Digitalausgang in der Lage, kurzzeitig einen deutlich höheren Strom als den Nennwert zu liefern.

Diagnosemeldung

Bei "Kurzschluss Geberversorgung" oder "Kurzschluss Digitalausgang" generiert das 2PULSE eine Diagnosemeldung zur angeschlossenen CPU/Master. Dazu müssen Sie den Parameter "Sammeldiagnose (Seite 1180)" freigeben.

Hinweis

Über Diagnosemeldungen erhalten Sie die Information, dass in einem oder beiden Kanälen des 2PULSE ein Fehler aufgetreten ist. Sie können die Diagnose mit STEP 7 auswerten.

Der Diagnosedatensatz DS1 enthält nur die Informationen zu Kanal 0. Ein Diagnosealarm wird aber auch bei Fehlern zu Kanal 1 ausgelöst. Detaillierte Informationen zu den aufgetretenen Fehlern aus beiden Kanälen liefern Ihnen die Rückmeldesignale. Werten Sie in neuen Anwenderprogrammen die Rückmeldeschnittstelle (Seite 1187) aus (siehe unten).

Parameter

Parameter	Bedeutung	Wertebereich	Voreinstellung
Sammeldiagnose (Seite 1180)	Das 2PULSE generiert bei freigegebener Sammeldiagnose eine Diagnosemeldung zur CPU/Master.	sperren/freigeben	sperren
Diagnose DO (Seite 1181)	Das 2PULSE erkennt einen Kurzschluss an den Digitalausgängen bei "Diagnose DO" = ein.	aus/ein	ein

Rückmeldesignale

Rückmeldesignale	Bedeutung	Wertebereich	Adresse Kanal 0	Adresse Kanal 1
ERR_PARA	Zeigt einen Parametrierfehler an.	0 = kein Parametrierfehler 1 = Parametrierfehler	Byte 0: Bit 5	Byte 4: Bit 5
ERR_PULS	Zeigt einen Impulsausgabefehler an.	0 = kein Impulsausgabefehler 1 = Impulsausgabefehler	Byte 0: Bit 4	Byte 4: Bit 4
ERR_24V	Zeigt einen Kurzschluss der Geberversorgung an.	0 = kein Kurzschluss Geberversorgung 1 = Kurzschluss Geberversorgung	Byte 0: Bit 7	Byte 4: Bit 7
ERR_DO	Zeigt einen Kurzschluss des Digitalausgangs DO an.	0 = kein Kurzschluss Digitalausgang 1 = Kurzschluss Digitalausgang	Byte 0: Bit 6	Byte 4: Bit 6

Parallelschaltung beider Kanäle (ab 6ES7138-4DD01-0AB0)

Definition

Um einen höheren Ausgangsstrom zu erreichen, können Sie die beiden Kanäle des 2PULSE parallel schalten. Sie erhalten dann einen Nennstrom von 4 A. Das ermöglicht es Ihnen, auch Aktoren mit höherem Strombedarf direkt am 2PULSE zu betreiben.

Parametrierung und Bedienung

Die Parallelschaltung der beiden Kanäle aktivieren Sie bei der Parametrierung mit dem Optionskästchen "Parallelschaltung Kanal 0 und 1 (Seite 1181)".

Nach dieser Aktivierung arbeitet das 2PULSE wie ein einkanaliges Modul. Alle Parametrierungen und Bedienungen über die Steuerschnittstelle (Seite 1184) müssen Sie mit Kanal 0 ausführen. Kanal 1 ist nicht parametrierbar und bedienbar.

Rückmeldesignale (Seite 1187) werden ebenso nur für Kanal 0 geliefert.

Strommessung

Bei der Strommessung werden die Messwerte aus beiden Kanälen nochmals gemittelt. Damit ergibt sich ein Messbereich bis 4 A.

Der Strommesswert wird in der Rückmeldeschnittstelle als SIMATIC S7-Analogwert geliefert. Ein Strom von 4 A entspricht dabei dem SIMATIC S7-Analogwert 27648.

Die Genauigkeit der Strommessung beträgt bei Parallelschaltung ebenfalls 3 % vom Messbereich.

Anschlussbelegung für die Parallelschaltung

Für die Parallelschaltung beider Kanäle des 2PULSE müssen Sie am Terminalmodul folgende Klemmen miteinander verbinden:

- Klemme 3 und Klemme 7
- Klemme 4 und Klemme 8

Die Last (z. B. einen Aktor) schließen Sie dann direkt an die Klemmen 3 und 4 bzw. an die Klemmen 7 und 8 an. Das folgende Bild zeigt ein Anschlussschema.

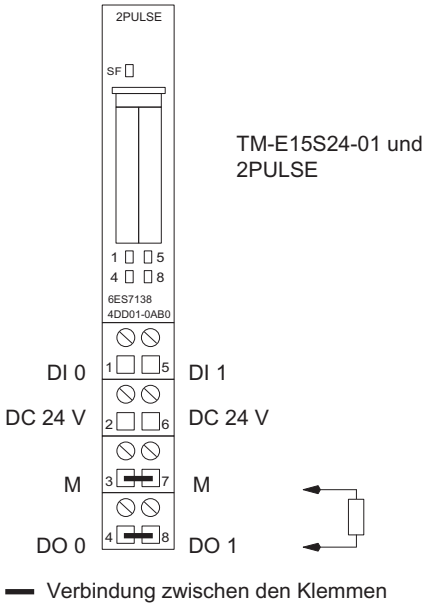

Verhalten bei CPU/Master-STOP

Definition

Sie parametrieren das Verhalten (Seite 1180) des 2PULSE bei Ausfall der überlagerten Steuerung für beide Kanäle gemeinsam.

Verhalten bei CPU/Master-STOP	kanalspezifische Reaktion und Zustand des 2PULSE
DO abschalten	den Digitalausgang DO löschen, STS_ENABLE löschen und die laufende Ausgabesequenz abbrechen
Betriebsart weiterarbeiten	der Digitalausgang DO bleibt unverändert, STS_ENABLE bleibt unverändert und die laufende Ausgabesequenz arbeitet weiter
DO Ersatzwert schalten	den kanalspezifischen, parametrierten Ersatzwert des Digitalausgangs DO ausgeben, STS_ENABLE löschen und die laufende Ausgabesequenz abbrechen
DO letzten Wert halten	der Digitalausgang DO bleibt unverändert, STS_ENABLE löschen und die laufende Ausgabesequenz abbrechen

Anlauf

Zum Start einer neuen Ausgabesequenz nach CPU/Master-STOP und gesetztem ACK_SW_ENABLE löschen Sie zuerst SW_ENABLE. Wiederholen Sie dieses Löschen so lange, bis ACK_SW_ENABLE ebenfalls gelöscht ist.

Soll die Betriebsart beim Wechsel vom CPU/Master-STOP nach RUN (Anlauf) weiterarbeiten, darf die CPU / der Master die Ausgänge nicht löschen. **Mögliche Abhilfe:** Setzen Sie in dem Teil des Anwenderprogramms, der beim Anlauf bearbeitet wird, das Steuerbit "Software-Freigabe" (SW_ENABLE = 1).

Geänderte Parametrierung

Der vom 2PULSE eingenommene Zustand bei CPU/Master-STOP bleibt auch bei einer Parametrierung bzw. Konfiguration der ET 200S-Station erhalten. Dies erfolgt z. B. bei NETZ-EIN der CPU / Master oder der IM 151 oder bei Wiederkehr der DP-Übertragung.

Aber im Zustand "Betriebsart weiterarbeiten" und nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU / Master bricht das 2PULSE die Weiterarbeit ab. Dies hat zur Folge, dass das 2PULSE

- den Digitalausgang DO löscht
- STS_ENABLE löscht und
- die laufende Ausgabesequenz abbricht.

3.5.8.3 2PULSE konfigurieren und parametrieren

Parameter

Betriebsmodus auswählen

Betriebsmodus

Wählen Sie den Betriebsmodus des 2PULSE nach Ihren Anforderungen aus:

- 2 Pulse: 8 Byte Ausgangsdaten – kompatibel zur Vorgängerbaugruppe
  Die Steuerschnittstelle belegt weniger Adressen im Prozessabbild.
- 2 Pulse erw. Steuerschnittstelle: 12 Byte Ausgangsdaten – verbesserte Steuerungsmöglichkeiten
  Sie können die Sollwerte komfortabler eingeben.

Sammeldiagnose aktivieren

Sammeldiagnose

Die Sammeldiagnose aktivieren Sie im entsprechenden Optionskästchen.

Das 2PULSE generiert bei freigegebener Sammeldiagnose eine Diagnosemeldung zur CPU bzw. zum Master.

Verhalten bei CPU/Master-STOP auswählen

Verhalten bei CPU/Master-STOP

Wählen Sie aus der Klappliste das gewünschte Verhalten bei CPU/Master-STOP aus:

- DO abschalten
- Betriebsart weiterarbeiten
- DO Ersatzwert schalten
- DO letzten Wert halten

Weitere Informationen zu Reaktionen und Zustand des 2PULSE finden Sie unter "Verhalten bei CPU/Master-STOP (Seite 1179)".

Parallelschaltung Kanal 0 und 1 aktivieren

Parallelschaltung Kanal 0 und 1 (ab 6ES7138-4DD01-0AB0)

Wenn Sie an den Digitalausgängen einen höheren Nennstrom als 2 A benötigen, dann können Sie die beiden Kanäle des 2PULSE parallel schalten. Dazu müssen Sie das Optionskästchen "Parallelschaltung Kanal 0 und 1" aktivieren.

Weitere Informationen finden Sie unter "Parallelschaltung beider Kanäle (ab 6ES7138-4DD01-0AB0) (Seite 1177)".

Kanal n

Diagnose DO aktivieren

Diagnose DO

Wenn Sie das Optionskästchen "Diagnose DO" aktivieren, dann erkennt das 2PULSE einen Kurzschluss an den Digitalausgängen.

Ersatzwert DO auswählen

Ersatzwert DO

Wählen Sie aus der Klappliste den gewünschten Ersatzwert für den Digitalausgang aus:

- 0
- 1

Der Parameter "Ersatzwert DO" darf nur dann einen Wert ungleich 0 haben, wenn der Parameter "Verhalten bei CPU/Master-STOP (Seite 1180)" auf "DO Ersatzwert schalten" gestellt ist.

Betriebart auswählen

Betriebsart

Wählen Sie aus der Klappliste die gewünschte Betriebsart aus:

- Impulsausgabe (Seite 1137)
- Pulsweitenmodulation (PWM) (Seite 1143)
- Impulskette (Seite 1153)
- Ein-/Ausschaltverzögerung (Seite 1159)
- Frequenzausgabe (Seite 1167) (ab 6ES7138-4DD01-0AB0)

Ausgabeformat auswählen

Ausgabeformat

Das Ausgabeformat ist relevant für die Betriebsarten "Pulsweitenmodulation (PWM)" und "Frequenzausgabe" (ab 6ES7138-4DD01-0AB0).

- In der Betriebsart "Pulsweitenmodulation (PWM) (Seite 1143)" können Sie aus der Klappliste als Ausgabeformat wählen:
  - Promille
  - S7-Analogausgabe
- In der Betriebsart "Frequenzausgabe (Seite 1167)" können Sie aus der Klappliste als Ausgabeformat wählen:
  - 1 Hz
  - S7-Analogausgabe

Zeitbasis auswählen

Zeitbasis

Wählen Sie über die Zeitbasis die Auflösung und den Wertebereich aller vorgegebenen Zeiten aus:

- 0,1 ms
  Sie können Zeiten von 0,2 ms bis 6,5535 s mit einer Auflösung von 0,1 ms einstellen.
- 1 ms
  Sie können Zeiten von 1 ms bis 65,535 s mit einer Auflösung von 1 ms einstellen.

Funktion DI auswählen

Funktion DI

Den Digitaleingang DI können Sie als Eingang verwenden oder als Hardware-Freigabe.

Wählen Sie aus der Klappliste die gewünschte Funktion für den Digitaleingang aus:

- Eingang
- HW-Freigabe

Einschaltverzögerung festlegen

Einschaltverzögerung

Die Einschaltverzögerung ist die Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulse abläuft.

Tragen Sie die gewünschte Einschaltverzögerung in das Eingabefeld ein.

Zulässiger Wertebereich: 0 … 65535

Mindestimpulsdauer festlegen

Mindestimpulsdauer

Die Mindestimpulsdauer ist relevant für die Betriebsarten "Pulsweitenmodulation (PWM)" (Seite 1143) und "Frequenzausgabe (Seite 1167)" (ab 6ES7138-4DD01-0AB0).

Mindestimpulsdauer und Mindestimpulspause parametrieren Sie mit dem Parameter "Mindestimpulsdauer", sie haben immer denselben Wert.

Tragen Sie hier z. B. die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds ein.

Zulässiger Wertebereich: 0 … 65535

Impulsdauer festlegen

Impulsdauer

Die Impulsdauer ist relevant für die Betriebsart "Impulskette (Seite 1153)".

Tragen Sie hier z. B. die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds ein.

Zulässiger Wertebereich:

- bei Zeitbasis 1 ms: von 1 … 65535
- bei Zeitbasis 0,1 ms: von 2 … 65535

Periodendauer festlegen

Periodendauer

Die Periodendauer ist relevant für die Betriebsarten "Pulsweitenmodulation (PWM) (Seite 1143)" und "Impulskette (Seite 1153)".

Beachten Sie bei der Vorgabe der Periodendauer die eingestellte Mindestimpulsdauer / Impulsdauer und die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds. Die Periodendauer sollte immer ein Vielfaches der Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds sein.

Die Periodendauer legen Sie abhängig von der Zeitbasis als Zahlenwert fest:

- bei Zeitbasis 1 ms: von 1 … 65535
- bei Zeitbasis 0,1 ms: von 2 … 65535

parametrierte Periodendauer = Zeitbasis × vorgegebener Zahlenwert

Steuer- und Rückmeldeschnittstelle des 2PULSE

Steuerschnittstelle des 2PULSE

Hinweis

Für das 2PULSE sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7

Belegung der kurzen Steuerschnittstelle (8 Byte)

Die folgende Tabelle zeigt die Belegung der kurzen Steuerschnittstelle (Ausgänge) des 2PULSE.

Adresse		Belegung
Kanal 0	Kanal 1	
Word 0	Word 4	je nach Betriebsart • Impulsausgabe: Impulsdauer • Pulsweitenmodulation: Ausgabewert • Impulskette: Impulsanzahl • Ein-/Ausschaltverzögerung: Ausschaltverzögerung • Frequenzausgabe: Ausgabefrequenz (ab 6ES7138-4DD01-0AB0)
Byte 2	Byte 6	Bit 7: Reserve = 0 Bit 6: Reserve = 0 Bit 5: Reserve = 0 Bit 4: Reserve = 0 Bit 3: Reserve = 0 Bit 2: SET_DO Bit 1: MANUAL_DO Bit 0: SW_ENABLE
Byte 3	Byte 7	je nach Betriebsart • Impulsausgabe: Faktor Einschaltverzögerung • Pulsweitenmodulation: Faktor Periodendauer • Impulskette: Faktor Periodendauer • Ein-/Ausschaltverzögerung: Faktor Einschaltverzögerung • Frequenzausgabe: Faktor Einschaltverzögerung (ab 6ES7138-4DD01-0AB0)

Belegung der langen Steuerschnittstelle (12 Byte) (ab 6ES7138-4DD01-0AB0)

Die folgende Tabelle zeigt die Belegung der langen Steuerschnittstelle (Ausgänge) des 2PULSE.

Adresse		Belegung
Kanal 0	Kanal 1	
Word 0	Word 6	je nach Betriebsart • Impulsausgabe: Impulsdauer • Pulsweitenmodulation: Ausgabewert • Impulskette: Impulsanzahl • Ein-/Ausschaltverzögerung: Ausschaltverzögerung • Frequenzausgabe: Ausgabefrequenz
Word 2	Word 8	je nach Betriebsart • Impulsausgabe: Einschaltverzögerung • Pulsweitenmodulation: Periodendauer • Impulskette: Periodendauer • Ein-/Ausschaltverzögerung: Einschaltverzögerung • Frequenzausgabe: Einschaltverzögerung
Byte 4	Byte 10	Bit 7: Reserve = 0 Bit 6: Reserve = 0 Bit 5: Reserve = 0 Bit 4: Reserve = 0 Bit 3: Reserve = 0 Bit 2: SET_DO Bit 1: MANUAL_DO Bit 0: SW_ENABLE
Byte 5	Byte 11	je nach Betriebsart • Impulsausgabe: nicht genutzt • Pulsweitenmodulation: Faktor Einschaltverzögerung • Impulskette: Puls/Pause-Verhältnis • Ein-/Ausschaltverzögerung: nicht genutzt • Frequenzausgabe: nicht genutzt

Erläuterungen zu den Steuersignalen

Steuersignal	Erläuterung
Betriebsart "Impulsausgabe":	
• Impulsdauer [1) 2)]	Zeit, die der Digitalausgang DO nach Ablauf der Einschaltverzögerung gesetzt ist.
• Faktor Einschaltverzögerung [1)]	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung

Steuersignal	Erläuterung
• Einschaltverzögerung [2]	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulsfolge abläuft.
Betriebsart "Pulsweitenmodulation":	
• Ausgabewert [1] [2]	Wert, der am Digitalausgang DO nach Ablauf der Einschaltverzögerung pulsweitenmoduliert ausgegeben wird.
• Faktor Periodendauer [1]	Sie können die parametrierte Periodendauer ändern: Periodendauer = Faktor × 0,1 × parametrierte Periodendauer
• Periodendauer [2]	Beachten Sie bei der Vorgabe der Periodendauer die eingestellte Mindestimpulsdauer und die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds.
• Faktor Einschaltverzögerung [2]	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung
Betriebsart "Impulskette":	
• Impulsanzahl [1] [2]	Anzahl Impulse, die am Digitalausgang DO nach Ablauf der Einschaltverzögerung ausgegeben werden.
• Faktor Periodendauer [1]	Sie können die parametrierte Periodendauer vor dem Start der Ausgabesequenz ändern: Periodendauer = Faktor × 0,1 × parametrierte Periodendauer
• Periodendauer [2]	Beachten Sie bei der Vorgabe der Periodendauer die eingestellte Mindestimpulsdauer und die Ansprechzeit Ihres am Digitalausgang DO angeschlossenen Stellglieds.
• Puls/Pause-Verhältnis [2]	Sie können das Verhältnis von Impulsdauer zu Impulspause einstellen: Puls/Pause-Verhältnis = (Sollwert / 255) × Periodendauer
Betriebsart "Ein-/Ausschaltverzögerung":	
• Ausschaltverzögerung [1] [2]	Zeit zwischen negativer Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO.
• Faktor Einschaltverzögerung [1]	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung
• Einschaltverzögerung [2]	Zeit zwischen positiver Flanke des Digitaleingangs DI und deren Ausgabe am Digitalausgang DO.
Betriebsart "Frequenzausgabe" (ab 6ES7138-4DD01-0AB0)**:**	
• Ausgabefrequenz [1] [2]	Frequenz, die am Digitalausgang DO nach Ablauf der Einschaltverzögerung ausgegeben wird.
• Faktor Einschaltverzögerung [1]	Sie können die parametrierte Einschaltverzögerung vor dem Start der Ausgabesequenz ändern: Einschaltverzögerung = Faktor × 0,1 × parametrierte Einschaltverzögerung
• Einschaltverzögerung [2]	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Frequenz abläuft.
Direktes Steuern des Digitalausgangs:	
• MANUAL_DO	Mit dem Steuerbit wählen Sie die Funktion "Direktes Steuern des Digitalausgangs DO" an und ab.
• SET_DO	Mit dem Steuerbit geben Sie den Zustand des Digitalausgangs DO vor.

Steuersignal	Erläuterung
Software-Freigabe	
• SW_ENABLE	Die Software-Freigabe müssen Sie immer in Ihrem Steuerungsprogramm erteilen. Wenn Sie keine HW-Freigabe verwenden, dann wird die Ausgabesequenz durch die positive Flanke der Software-Freigabe gestartet. Wenn Sie die Software-Freigabe löschen, dann wird die laufende Ausgabesequenz abgebrochen.

1) bei Nutzung der kurzen Steuerschnittstelle

2) bei Nutzung der langen Steuerschnittstelle

Rückmeldeschnittstelle des 2PULSE

Hinweis

Für das 2PULSE sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 2PULSE.

Adresse		Belegung
Kanal 0	Kanal 1	
Byte 0	Byte 4	Bit 7: ERR_24V Bit 6: ERR_DO Bit 5: ERR_PARA Bit 4: ERR_PULS Bit 3: ACK_SW_ENABLE Bit 2: STS_DI Bit 1: STS_DO Bit 0: STS_ENABLE
Byte 1	Byte 5	Bit 0 … Bit 3: Sequenzzähler SEQ_CNT (ab 6ES7138-4DD01-0AB0) Bit 4 … Bit 7: Reserve = 0
Word 2	Word 6	Strommesswert (ab 6ES7138-4DD01-0AB0)

Hinweis

Abhängig vom verwendeten Interfacemodul IM 151 und der Projektierung wird bei Ausfall des Moduls als Rückmeldewert (Ersatzwert) 16#7FFF7FFF oder 16#00000000 geliefert.

Erläuterungen zu den Rückmeldesignalen

Rückmeldesignal	Erläuterung
ACK_SW_ENABLE	Zeigt den am 2PULSE anstehenden Zustand der Software-Freigabe an.
ERR_24V	Zeigt einen Kurzschluss bzw. Einbruch der Geberversorgung an.
ERR_DO	Zeigt einen Kurzschluss des Digitalausgangs an.
ERR_PARA	Zeigt einen Parametrierfehler an.
ERR_PULS	Betriebsart "Impulsausgabe": Zeigt einen Impulsausgabefehler an. Wenn Sie nach Ablauf der Einschaltverzögerung die Impulsdauer so verkürzen, dass die Zeit kleiner ist als die schon ausgegebene Zeit, wird dies vom 2PULSE erkannt. Mit dem nächsten Start der Ausgabesequenz löscht das 2PULSE das Rückmeldebit ERR_PULS. Betriebsart "Impulskette": Zeigt einen Impulsausgabefehler an. Wenn Sie nach Ablauf der Einschaltverzögerung die Impulsanzahl verkleinern und ist die kleinere Impulsanzahl bereits ausgegeben, wird dies vom 2PULSE erkannt. Mit dem nächsten Start der Ausgabesequenz löscht das 2PULSE das Rückmeldebit ERR_PULS. Betriebsart "Ein-/Ausschaltverzögerung": Zeigt einen Impulsausgabefehler bei zu kurzer Impulsdauer oder Impulspause an. Mit der nächsten positiven Flanke der Software-Freigabe bzw. mit der nächsten Flanke am Digitaleingang DI löscht das 2PULSE das Rückmeldebit ERR_PULS.
STS_ DI	Zeigt den Signalpegel am Digitaleingang DI an.
STS_DO	Zeigt den Signalpegel am Digitalausgang DO an.
STS_ENABLE	Betriebsart "Impulsausgabe": Ist mit dem Start der Ausgabesequenz bis zum Ablauf der Impulsdauer gesetzt. Wenn Sie die Software-Freigabe (SW_ENABLE) löschen oder das 2PULSE einen Impulsausgabefehler ERR_PULS erkennt, wird STS_ENABLE gelöscht. Betriebsart "Pulsweitenmodulation" (PWM): Ist mit dem Start der Ausgabesequenz gesetzt. Wenn Sie die Software-Freigabe (SW_ENABLE) löschen, wird STS_ENABLE gelöscht. Betriebsart "Impulskette": Ist mit dem Start der Ausgabesequenz bis einschließlich der Ausgabe des letzten Impulses gesetzt. Wenn Sie die Software-Freigabe (SW_ENABLE) löschen oder das 2PULSE einen Impulsausgabefehler ERR_PULS erkennt, wird STS_ENABLE gelöscht. Betriebsart "Ein-/Ausschaltverzögerung": Zeigt den vom 2PULSE erkannten Zustand der Software-Freigabe (SW_ENABLE) an. Betriebsart "Frequenzausgabe"(ab 6ES7138-4DD01-0AB0): Ist mit dem Start der Ausgabesequenz gesetzt. Wenn Sie die Software-Freigabe (SW_ENABLE) löschen, wird STS_ENABLE gelöscht.
SEQ_CNT (ab 6ES7138-4DD01-0AB0)	Zählt die abgeschlossenen Ausgabesequenzen in den Betriebsarten "Impulsausgabe", "Impulskette" und "Ein-/Ausschaltverzögerung". Keine Funktion in den Betriebsarten "Pulsweitenmodulation" und "Frequenzausgabe".
Strommesswert (ab 6ES7138-4DD01-0AB0)	Strom am DO als SIMATIC S7-Analogwert

3.5.8.4 Diagnose des 2PULSE

Diagnose durch LED-Anzeige

LED-Anzeige am 2PULSE

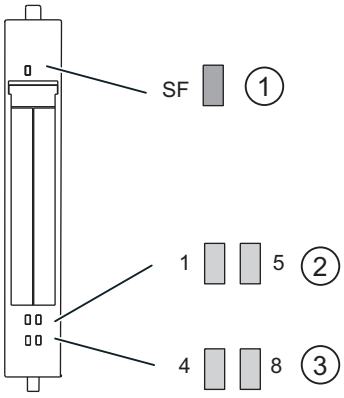

① Sammelfehler (rot)
② Statusanzeige für Digitaleingang (grün)
③ Statusanzeige für Digitalausgang (grün)

Status- und Fehleranzeigen durch LEDs am 2PULSE

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 2PULSE.

Ereignis (LED)					Ursache	Maßnahme
SF	1	5	4	8		
ein					Keine Parametrierung. Diagnosemeldung liegt vor.	Überprüfen Sie die Fehlerbits in der Rückmeldeschnittstelle. Werten Sie die Diagnose aus.
	ein				Eingang DI0 am Kanal 0 aktiviert.	
		ein			Eingang DI1 am Kanal 1 aktiviert.	
			ein		Ausgang DO0 am Kanal 0 aktiviert.	
				ein	Ausgang DO1 am Kanal 1 aktiviert.	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

2PULSE Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 2PULSE.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Geberversorgung oder des Aktors. Ein Einbruch der 24 V-Versorgungsspannung kann ebenfalls zu dieser Fehlermeldung führen.	Überprüfen Sie die Verdrahtung zu den Schaltern und den Aktoren. Korrektur der Prozessverdrahtung.
9_D	01001: Fehler	Interner Modulfehler ist aufgetreten.	Austausch des Moduls.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.

Eine einfache Fehlerauswertung ist auch über die entsprechenden Bits in der Rückmeldeschnittstelle (Seite 1187) möglich.

3.6 Hilfsfunktionen

3.6.1 Polyline

Polyline

Die Anweisung Polyline stellt eine Funktion mit der Kennlinie des Polygonzugs zur Verfügung, mit dessen Stützpunkten z. B. das Verhalten von nicht linearen Sensoren linearisiert werden kann.

Die Anweisung Polyline ist mit S7-1500 CPU ab Firmware 2.0 und S7-1200 CPU ab Firmware 4.2 einsetzbar.

Weitere Informationen

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

3.6.2 SplitRange

SplitRange

Die Anweisung SplitRange teilt den Ausgangswertebereich des PID-Reglers in mehrere Unterbereiche auf. Diese Unterbereiche ermöglichen die Regelung eines Prozesses, den mehrere Aktoren beeinflussen.

Die Anweisung SplitRange ist mit S7-1500 CPU ab Firmware 2.0 und S7-1200 CPU ab Firmware 4.2 einsetzbar.

Weitere Informationen

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

3.6.3 RampFunction

RampFunction

Die Anweisung RampFunction begrenzt die Änderungsgeschwindigkeit eines Signals. Ein Signalsprung am Eingang wird als Rampenfunktion des Ausgangswerts ausgegeben, um z. B. ein sanfteres Führungsverhalten zu erreichen, ohne Beeinflussung des Störverhaltens.

Die Anweisung RampFunction ist mit S7-1500 CPU ab Firmware 2.0 und S7-1200 CPU ab Firmware 4.2 einsetzbar.

Weitere Informationen

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

AUTOHOTSPOT

Time-based IO 4

4.1 Hochgenaue Ein-/Ausgabe mit Time-based IO

4.1.1 Grundlagen von Time-based IO

4.1.1.1 Nutzung von Time-based IO

Viele Vorgänge in einer Anlage erfordern eine zeitlich möglichst genaue Reproduzierbarkeit. Die Reproduzierbarkeit lässt sich durch kürzere CPU-Zykluszeiten in Grenzen optimieren. Die Verwendung hochgenauer Ein-/Ausgabe mit Time-based IO ermöglicht eine genauere Reproduzierbarkeit, als diese durch Optimierung der CPU-Zykluszeit möglich ist. Ein weiterer Vorteil von Time-based IO liegt in der Unabhängigkeit vom minimalen Zeitraster des Applikationszyklus.

Im Folgenden finden Sie eine Auswahl typischer Anwendungsbereiche, die von dieser Technologie profitieren können.

Definierte Reaktionszeiten

Durch Addition einer vorgegebenen Zeit auf den Zeitstempel einer Flanke am Digitaleingang kann ein Ausgangssignal ausgegeben werden, das mit einer präzisen, vom Applikationszyklus unabhängigen Verzögerungszeit reagiert.

Längenmessung

Durch die Differenz zweier Eingangs-Zeitstempel und einer dazugehörigen Transportgeschwindigkeit kann die Länge eines vorbeifahrenden Produktes ermittelt werden.

Nockensteuerwerk

Aus einer synchron erfassten Positionsinformation (z. B. von einer Zählerbaugruppe oder einer Achse) kann der weitere Bewegungsverlauf extrapoliert werden. Darauf aufbauend wird der Zeitpunkt einer Schaltposition (Nockenposition) errechnet und an das TIO-Modul (Ausgabe) übermittelt. Dadurch tritt das Schaltereignis genau an der gewünschten Position ein.

Dosieren

Durch die Vorgabe einer Ein- und einer Ausschaltflanke auf einen zeitgesteuerten Digitalausgang kann ein Ventil für eine bestimmte Zeit geöffnet und die entsprechende Menge an Flüssigkeit dosiert werden.

4.1.1.2 Zeitliches Verhalten von Time-based IO

Zeitliches Verhalten Standard-Technologie

Das zeitliche Verhalten der Ein-/Ausgänge ist bei Standard-Technologie von folgenden Faktoren abhängig:

- CPU-Programm (Programmstruktur)
- Buszykluszeiten (Feldbus, Rückwandbus)
- Zykluszeit der E/A-Module
- interne Zykluszeit der Sensoren/Aktoren

Eine deterministische Aussage darüber,

- wann ein Eingangsereignis (z. B. Sensorsignal) stattgefunden hat
- wann auf ein Eingangsereignis das Ausgangsereignis wirksam wird (z. B. Ausgang schaltet)

wird durch die oben genannten Zeitfaktoren ungenau.

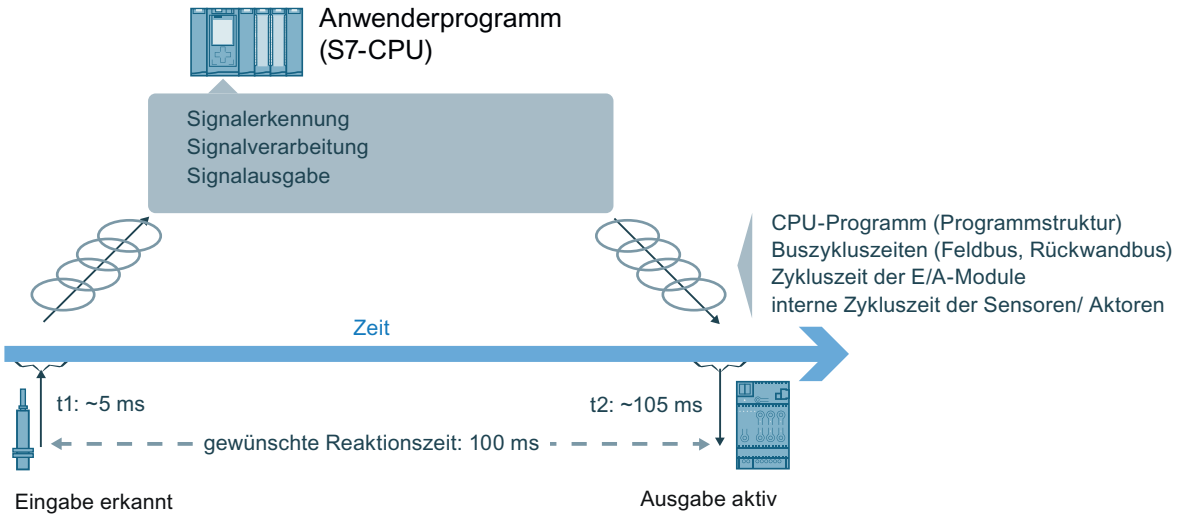

Eigenschaften von Time-based IO

Zeitgesteuerte E/A-Funktionalität

Time-based IO steht für eine zeitbasierte Verarbeitung von E/A-Signalen. Dabei werden alle Eingangssignale auf eine Zeit (TIO_Time) bezogen. Die Eingangssignale erhalten den Zeitstempel t1. Nach der Signalverarbeitung kann das Ausgangsereignis ebenfalls mit der TIO_Time verknüpft und zum gewünschten Zeitpunkt ausgegeben werden. Die Ausgabe erfolgt zum Zeitpunkt t2.

Im Folgenden ein Beispiel für Time-based IO:

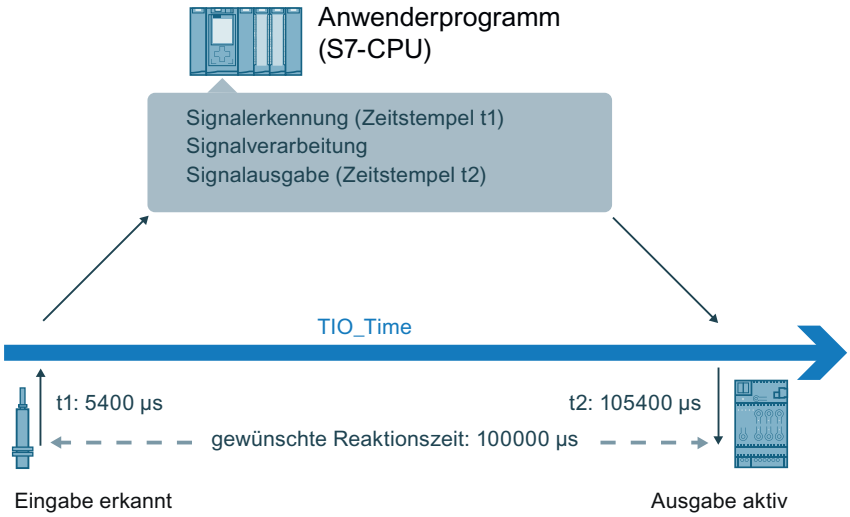

Unabhängigkeit und Fokus von Time-based IO

Basis für Time-based IO ist die gemeinsame Zeitbasis (TIO_Time) aller beteiligten Komponenten. Durch die Nutzung der TIO_Time ist bei Time-based IO die Genauigkeit der Ausgabe unabhängig von:

- CPU-Programm (Programmstruktur)
- Buszykluszeiten (Feldbus, Rückwandbus)
- Zykluszeit der E/A-Module
- interne Zykluszeit der IO-Link Sensoren/Aktoren

Der Fokus von Time-based IO liegt nicht auf der E/A-Reaktionszeit sondern auf der Vorhersagbarkeit (Determinismus) von E/A-Signalen. Mit Time-based IO ist es möglich, auf ein Eingangssignal in einer definierten Zeit mit einem Ausgangssignal zu reagieren. Beim Einsatz von Time-based IO ist die systembedingte Mindestreaktionszeit zu beachten.

Time-based IO steht für:

- hochgenau ausgeführte E/A-Funktionalitäten
- zeitgestempelte E/A-Prozesse

Genauigkeiten

Für die Leistungsfähigkeit von Time-based IO ist Genauigkeit entscheidend.

Die Genauigkeit ist eine Eigenschaft von

- TIO-Modul
- IO-Link Sensor/Aktor

und gibt die Abweichung an, mit der die zu erzielende Reaktion erreicht wird. Bei Time-based IO liegt die Genauigkeit und Reproduzierbarkeit der Reaktion im Mikrosekundenbereich.

Reaktionszeit

Neben einer sehr hohen Genauigkeit ergibt sich abhängig von der Konfiguration eine minimale Reaktionszeit auf ein Eingangsereignis.

Die Reaktionszeit ist der zeitliche Abstand vom Eingangsereignis bis zum gewünschten Ausgangsereignis.

Für die minimale Reaktionszeit gilt:

mind. 3 * Applikationszyklus (T_{APP})

4.1.1.3 Systemumgebung

Einleitung

Im Folgenden sind die möglichen Systemkonfigurationen mit PROFINET für die Nutzung von Time-based IO aufgezeigt.

Einsatz mit TIA Portal

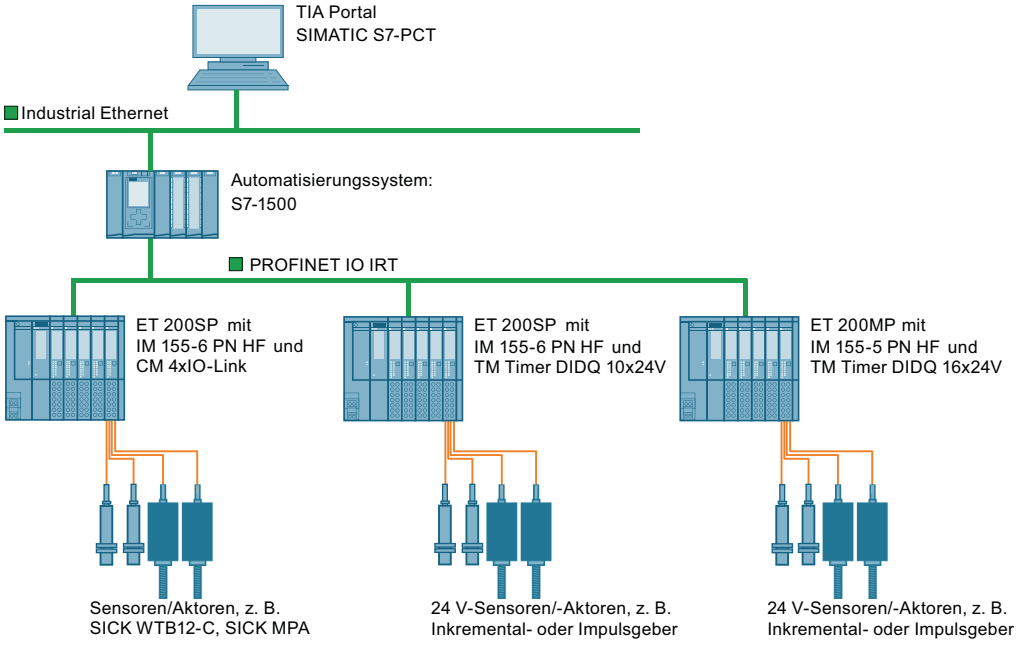

4.1.1.4 Hardware-Voraussetzungen

Einleitung

Im Folgenden sind die Eigenschaften der Hardware-Komponenten benannt, welche für Time-based IO notwendig sind. Zudem sind beispielhaft konkrete Module aufgeführt.

Voraussetzungen

Komponente	Eigenschaften	Beispiele
CPU	• Taktsynchronität/PROFINET IO IRT: ermöglicht definierte Reaktionszeiten und hochpräzises Anlagenverhalten.	• CPU 1511-1 PN
ET 200 Interface-modul	• unterstützt Taktsynchronität	• ET 200SP mit IM 155-6 PN HF (ab Firmware V2.1) • ET 200MP mit IM 155-5 PN HF
E/A-Modul	• TIO-Modul	• ET 200SP: CM 4xIO-Link (ab Firmware V2.0) TM Timer DIDQ 10x24V • ET 200MP: TM Timer DIDQ 16x24V
IO-Link Device für CM 4xIO-Link	• unterstützt IO-Link Zeitstempelfunktion	• SICK WTB12-C • halstrup-walcher PSE, PSS, PSW

4.1.1.5 Projektiersoftware-Voraussetzungen

Einleitung

Im Folgenden sind die Software-Versionen aufgelistet, welche die Funktion "Time-based IO" unterstützen.

Voraussetzungen

Projektiersoftware	Voraussetzungen	Unterstützte Hardware-Komponen-ten	Weitere Informationen
STEP 7 (TIA Portal) ab V13	• PROFINET IO IRT • Für CM 4xIO-Link: IO-Link Technologie	• Automatisierungssystem S7-1500 • Dezentrale Peripheriesysteme ET 200SP und ET 200MP • CM 4xIO-Link • TM Timer DIDQ 16x24V (ab V13 Update 3) • TM Timer DIDQ 10x24V (ab V13 Update 3)	Online-Hilfe von TIA Portal
S7-PCT ab V3.2	—	• CM 4xIO-Link	Online-Hilfe von S7-PCT

TIO-Anweisungen

Die TIO-Anweisungen sind spezifische Funktionsbausteine für die Nutzung von Time-based IO. Folgende TIO-Anweisungen stehen zur Verfügung:

- TIO_SYNC
- TIO_IOLink_IN
- TIO_IOLink_OUT
- TIO_DI
- TIO_DQ

Ab V13 Update 3 stehen die TIO-Anweisungen im TIA Portal integriert zur Verfügung.

4.1.1.6 Technische Umsetzung

Einleitung

Im Folgenden machen Sie sich mit den Kernaspekten von Time-based IO vertraut und erfahren, mit welchen SIMATIC-Funktionen die beschriebenen Aspekte realisiert werden.

Synchronisierung der beteiligten Module am PROFINET (gemeinsame Zeitbasis)

Die Technologie Time-based IO verwendet Taktsynchronität für alle beteiligten Teilnehmer am PROFINET. Ermöglicht wird die Taktsynchronisierung durch PROFINET IRT (Isochronous Real-Time) und dem taktsynchronen Betrieb der Teilnehmer am PROFINET.

Weitere Informationen zu Taktsynchronität finden Sie in den Handbüchern PROFINET mit STEP 7 (http://support.automation.siemens.com/WW/view/de/49948856) und Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Über das taktsynchrone PROFINET ist die Möglichkeit gegeben, mehrere TIO-Module auf eine gemeinsame Zeitbasis zu synchronisieren. Grundlage der gemeinsamen Zeitbasis für die TIO-Module ist die relative Zeit TIO_Time.

TIO_Time

Die TIO_Time ist die zentrale Zeitbasis, auf die sich alle Zeitstempel beziehen.

Die TIO_Time hat folgende Eigenschaften:

- Gemeinsame Zeitbasis für alle TIO-Module, welche über die TIO-Anweisung TIO_SYNC synchronisiert werden.
- Mit jedem CPU-Anlauf beginnt die Zeit neu zu zählen.
- Die TIO_Time ist vom Datentyp LTime (z. B. LT#14s830ms652us315ns).
- Alle gültigen Zeitstempel sind auf die TIO_Time bezogen:
  - Eingangszeitstempel der TIO-Module werden in den TIO-Anweisungen TIO_IOLink_IN und TIO_DI auf die TIO_Time umgerechnet.
  - Ausgangszeitstempel werden in den TIO-Anweisungen TIO_IOLink_OUT und TIO_DQ auf Ausgangszeitstempel der TIO-Module umgerechnet.

TIO-Anweisungen im OB "Synchronous Cycle"

Die TIO-Anweisungen müssen in einem OB des Typs "Synchronous Cycle" aufgerufen werden.

Weitere Informationen finden Sie im Kapitel Programmieren.

Modi für die Aktualisierung des Prozessabbildes

Im taktsynchronen Betrieb können Sie die Reihenfolge der Aktualisierung des Teilprozessabbilds der Eingangs- und Ausgangsdaten beeinflussen. Dabei können Sie folgende Modelle der Programmbearbeitung wählen:

- EVA-Modell (Applikationszyklus-Faktor = 1)
- AEV-Modell (Applikationszyklus-Faktor >= 1)

Die Abkürzungen E, V, A stehen für folgende Prozesse: E = Eingabe, V= Verarbeitung, A = Ausgabe.

Weitere Informationen zum Applikationszyklus-Faktor finden Sie im Handbuch PROFINET mit STEP 7 (http://support.automation.siemens.com/WW/view/de/49948856).

EVA-Modell (Applikationszyklus-Faktor = 1)

Das Anwenderprogramm wird nach der Verzögerungszeit gestartet. Aktualisieren Sie zunächst im Anwenderprogramm durch Aufruf der Systemanweisung SYNC_PI das entsprechende Teilprozessabbild der Eingänge. Nachfolgend beginnt die Verarbeitung (z. B. Berechnung der Zeitstempel). Zum Ende des Anwenderprogramms wird das entsprechende Teilprozessabbild der Ausgänge in der CPU durch SYNC_PO aktualisiert.

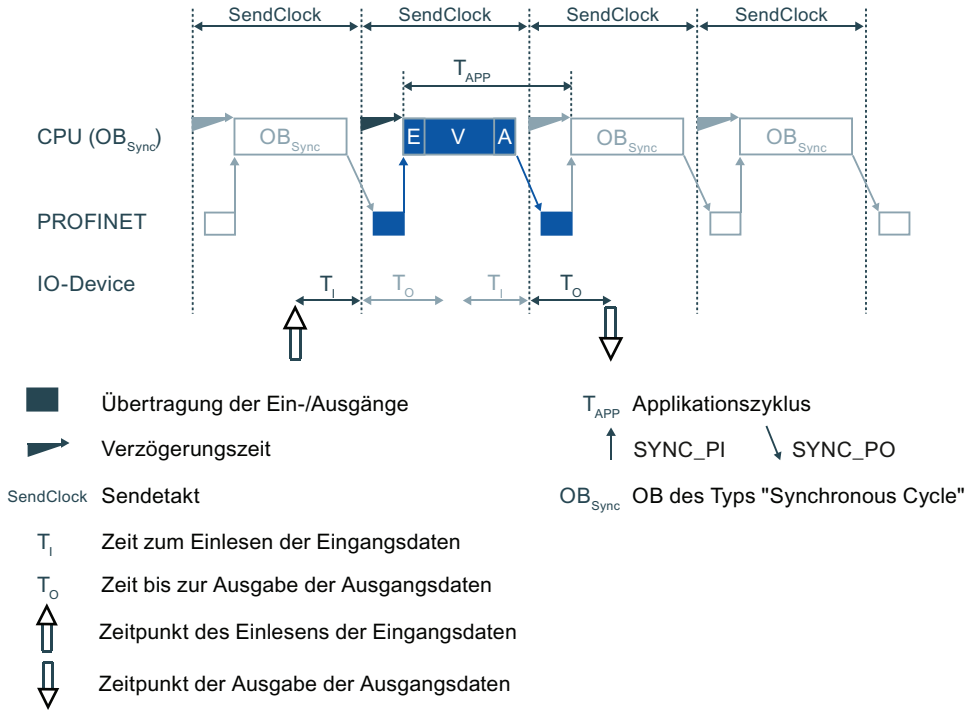

Eigenschaften des EVA-Modells:

- Ermöglicht kurze Reaktionszeiten
- Der Applikationszyklus darf nicht größer sein als der Sendetakt.
  Dadurch steht weniger Zeit für die Applikation zur Verfügung als beim AEV-Modell.

AEV-Modell (Applikationszyklus-Faktor >= 1)

Das Anwenderprogramm wird nach der Verzögerungszeit gestartet. Im Modus PIP_Mode 0 übernimmt die Anweisung TIO_SYNC die Aktualisierung des Prozessabbilds. In den anderen Modi aktualisieren Sie zunächst im Anwenderprogramm durch Aufruf der Systemanweisung SYNC_PO das entsprechende Teilprozessabbild der Ausgänge. Damit werden die Ausgangsdaten, die im vorherigen Netzwerkzyklus errechnet wurden, während des nächsten Netzwerkzyklus (T_O) aktiv. Nachfolgend wird das entsprechende Teilprozessabbild der Eingänge in der CPU durch SYNC_PI aktualisiert. Nach der Datenübertragung beginnt die Verarbeitung (z. B. Berechnung der Zeitstempel).

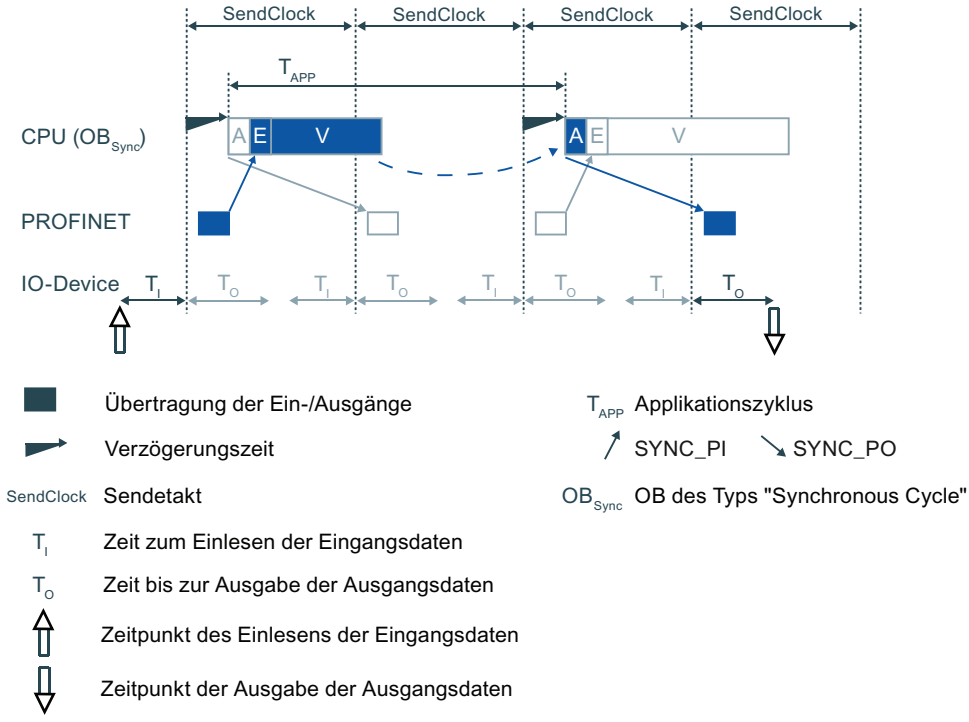

Eigenschaften des AEV-Modells:

- Die Reaktionszeit ist höher als beim EVA-Modell.
- Es ist ein Applikationszyklus größer als der Sendetakt erlaubt.
  Dadurch steht mehr Zeit für die Applikation zur Verfügung als beim EVA-Modell.

Einfluss der Genauigkeit

Zur Abschätzung der Genauigkeit benötigen Sie die Genauigkeiten von

- TIO-Modul
- Sensor/Aktor

Informieren Sie sich in den Datenblättern zum entsprechenden Modul nach dem Wert für die Genauigkeit.

Es sind die einzelnen Jitter der TIO-Module und der Sensoren/Aktoren zu addieren. Andere Einflussfaktoren sind vernachlässigbar.

4.1.2 Konfigurieren und Parametrieren

4.1.2.1 Voraussetzungen

Einleitung

Zur Verwendung von Time-based IO benötigen Sie zusätzliche Software-Komponenten. Zudem müssen Sie die Standard-Konfiguration für Ihr Projekt angelegt haben.

Im Folgenden lernen Sie die Standard-Konfiguration für Time-based IO kennen.

Voraussetzungen

Im TIA Portal:

- Das Projekt ist angelegt.
- Die CPU ist angelegt und parametriert.
- Dezentrale Peripherie ET 200 und Module sind angelegt und parametriert.
- Die Verbindung über PROFINET ist angelegt und parametriert.
- Ein OB des Typs "Synchronous Cycle" ist angelegt.

Zusätzlich für IO-Link:

- S7-PCT ist auf dem PG/PC installiert.
- IODD für IO-Link Device (mit Zeitstempelfunktion) sind installiert.

Siehe auch

Systemumgebung (Seite 1196)

Projektiersoftware-Voraussetzungen (Seite 1197)

4.1.2.2 Einstellungen für Time-based IO

Im Folgenden erhalten Sie eine Übersicht, welche Einstellungen für welche Komponenten für Time-based IO vorgenommen werden müssen.

Tabelle 4-1 Übersicht der Einstellungen für Time-based IO

Komponente	Wo einstellbar (*)	Einzustellende Eigenschaften	Weitere Informationen
PROFINET-Subnetz	Eigenschaften zum PROFINET-Subnetz > Sync-Domain	Sync-Domain erstellen bzw. Eigenschaften der Sync-Domain bearbeiten • z. B. Sendetakt einstellen (z. B. 2 ms)	• Funktionshandbuch PROFINET mit STEP 7 (http://support.automation.siemens.com/WW/view/de/49948856) • Online-Hilfe im TIA Portal Infosystem
		Teilnehmer der Sync-Domain festlegen: • CPU als Sync-Master festlegen. • ET 200 Interfacemodul als Sync-Slave festlegen mit RT-Klasse "IRT"	
ET 200-Station (mit TIO-Modul)	Eigenschaften der PROFINET-Schnittstelle > Taktsynchronisation	Zuordnung der TIO-Module zum taktsynchronen PROFINET-Subnetz	• Funktionshandbuch PROFINET mit STEP 7 (http://support.automation.siemens.com/WW/view/de/49948856) • Gerätehandbuch Technologiemodul TM Timer DIDQ 16x24V (http://support.automation.siemens.com/WW/view/de/95153313) • Gerätehandbuch Technologiemodul TM Timer DIDQ 10x24V (http://support.automation.siemens.com/WW/view/de/95153951) • Online-Hilfe im TIA Portal Infosystem
	Eigenschaften des TIO-Moduls > Taktsynchronität	Taktsynchronität aktivieren	
TIO-Modul	Eigenschaften des TIO-Moduls > E/A-Adressen	einen OB des Typs "Synchronous Cycle" zuordnen bzw. erstellen	
	Eigenschaften des TIO-Moduls > E/A-Adressen	Zuordnung der E/A-Adressen zum Teilprozessabbild (z. B. TPA1)	
	Eigenschaften des TM Timer DIDQ > Grundparameter/Kanalparameter	Verwendung der Timer-DI und Timer-DQ parametrieren	
OB "Synchronous Cycle"	Eigenschaften des OB "Synchronous Cycle" > Taktsynchronität	Gegebenenfalls Applikationszyklus anpassen	
IO-Link Device (Sensor/Aktor)	SIMATIC S7-PCT	Port für IO-Link Device für Time-based IO aktivieren	Siehe Kapitel IO-Link konfigurieren und Time-based IO aktivieren (Seite 1203)

* Beschreibt nicht den genauen Ort, sondern den Themenbereich in der Projektiersoftware.

Einstellung für Time-based IO

Sofern Sie keine besonderen Anforderungen an die Reaktionszeit haben, eignet sich als Einstieg folgende Einstellung:

- Sendetakt: 2 ms
- Applikationszyklus: 4 ms
- Zuordnung zum Teilprozessabbild: TPA1
- PIP_Mode: 0 (AEV-Modell)

FAQ

Weitere Informationen hierzu finden Sie in den folgenden FAQs im Siemens Industry Online Support:

- Beitrags-ID 109738186
- Beitrags-ID 109736374

4.1.2.3 IO-Link konfigurieren und Time-based IO aktivieren

Einleitung

Für Time-based IO müssen die Ports des IO-Link Master auf die gewünschte Betriebsart eingestellt und an den Ports passende Sensoren/Aktoren angeschlossen werden.

Voraussetzungen

- S7-PCT ist auf dem PG/PC installiert.
- IODD für IO-Link Device (mit Zeitstempelfunktion) sind installiert.
  Weitere Informationen zur Installation finden Sie in der Online-Hilfe zu S7-PCT.
- Das Projekt im TIA Portal ist in der Netzsicht geöffnet.
- Der Hardware-Katalog ist geöffnet.

Vorgehen

1. Klicken Sie mit der rechten Maustaste im TIA Portal auf den IO-Link Master und wählen Sie "Starte Device Tool".
   Das Port Configuration Tool (S7-PCT) wird geoffnet.
2. Wählen Sie für die Ports des IO-Link Master in S7-PCT die angeschlossenen IO-Link Device (Sensoren/Aktoren mit Zeitstempelfunktion) aus dem Gerätekatalog.
   Ziehen Sie dazu per Drag & Drop das IO-Link Device auf den passenden Port des IO-Link Master.

3. Passen Sie die Einstellungen der Ports an ihre jeweilige Aufgabe in der Anwendung an. Mögliche Einstellungen für die Ports:
   – IO-Link Time based IN: Die Eingangssignale des Sensors werden mit einem Zeitstempel versehen.
   – IO-Link Time based OUT: Die Signalausgabe erfolgt zu dem vorgegebenen Zeitpunkt.
4. Speichern Sie die Einstellungen ab.
5. Laden Sie die Port-Konfiguration in den IO-Link Master.
6. Schließen Sie S7-PCT.

Ergebnis

Der IO-Link Master ist mit den angeschlossenen Sensor- und Aktorinformationen konfiguriert.

4.1.3 Programmieren

4.1.3.1 Übersicht der Anweisungen

Einleitung

Time-based IO wird mit speziellen Anweisungen (TIO-Anweisungen) eingesetzt. Die TIO-Anweisung TIO_SYNC übernimmt dabei die Synchronisierung aller beteiligten TIO-Module und erzeugt eine eindeutige Zeitbasis (TIO_Time), auf die alle Aktionen referenziert werden.

Abhängig vom Typ des TIO-Moduls übernehmen weitere Anweisungen das Einlesen von Prozesseingangssignalen mit zugehörigen Zeitstempeln und/oder die zeitgesteuerte Ausgabe von Prozessausgangssignalen.

Hinweis

Die TIO-Anweisungen nutzen die Zeitstempelfunktionen der TIO-Module. Für die weiteren Funktionen des jeweiligen TIO-Moduls ist keine Nutzung von Anweisungen notwendig.

TIO-Anweisungen

Anweisung	Einsetzbar mit TIO-Modul	Kurzbeschreibung
TIO_SYNC	• CM 4xIO-Link • TM Timer DIDQ 16x24V • TM Timer DIDQ 10x24V	Synchronisiert die TIO-Module und gibt die Zeitbasis für Time-based IO vor
TIO_IOLink_IN	• CM 4xIO-Link	Liest die zeitgestempelten Eingangssignale vom IO-Link Device (Sensor)
TIO_DI	• TM Timer DIDQ 16x24V • TM Timer DIDQ 10x24V	Erfasst die Flanken am Digitaleingang (Timer-DI) und liefert die zugehörigen Zeitstempel

Anweisung	Einsetzbar mit TIO-Modul	Kurzbeschreibung
TIO_IOLink_OUT	• CM 4xIO-Link	Gibt die Ausgangssignale zeitgesteuert am IO-Link Device aus
TIO_DQ	• TM Timer DIDQ 16x24V • TM Timer DIDQ 10x24V	Gibt zeitgesteuert Flanken am Digitalausgang (Timer-DQ) aus

4.1.3.2 Programmierung von Time-based IO

Einleitung

Zur Nutzung von Time-based IO müssen die TIO-Anweisungen in einem OB des Typs "Synchronous Cycle" aufgerufen werden. Die Applikation kann auch in einem anderen OB ablaufen. Dadurch können Sie die Laufzeit des OB "Synchronous Cycle" verkürzen.

Entsprechend der gewünschten Aufgabe benötigen Sie folgende TIO-Anweisungen:

TIO-Modul	TIO-Anweisungen	
CM 4xIO-Link	• Pro Sensor einen TIO_IOLink_IN zum Einlesen • Pro Aktor einen TIO_IOLink_OUT zur Ausgabe	Einen TIO_SYNC (für bis zu acht TIO-Module)
TM Timer DIDQ	• Pro Digitaleingang einen TIO_DI zum Einlesen • Pro Digitalausgang einen TIO_DQ zur Ausgabe	

Im Folgenden ist die Programmierung der CPU für Time-based IO beschrieben.

Voraussetzungen

Hardware-Konfiguration im TIA Portal:

- Die TIO-Module sind einem taktsynchronen Netz zugeordnet.
- Die TIO-Module sind einem gemeinsamen Teilprozessabbild zugeordnet.
- Das Teilprozessabbild ist einem OB des Typs "Synchronous Cycle" zugeordnet.

Weitere Informationen zur Konfiguration von Time-based IO finden Sie im Kapitel Einstellungen für Time-based IO (Seite 1202).

Vorgehen

1. Legen Sie im OB "Synchronous Cycle" eine TIO-Anweisung TIO_SYNC an.
2. Verschalten Sie an der TIO-Anweisung TIO_SYNC über die Parameter HWID_1 bis HWID_8 alle zu synchronisierenden TIO-Module.
   Die HWID finden Sie in HWCN unter den Moduleigenschaften.
3. Parametrieren Sie an der TIO-Anweisung TIO_SYNC den Sendetakt.
   Den Sendetakt finden Sie bei den Eigenschaften der ET 200-Station.

4. Stellen Sie am TIO_SYNC am Eingangsparameter PIP_Mode den Modus der Datenaktualisierung ein.
   Die Beschreibung der Modi finden Sie im Kapitel Technische Umsetzung (Seite 1198).
   Die TIO-Anweisung TIO_SYNC ist fertig parametriert.
5. Achten Sie bei den folgenden Anweisungen auf die Aufrufreihenfolge abhängig vom gewählten Wert für PIP_Mode:

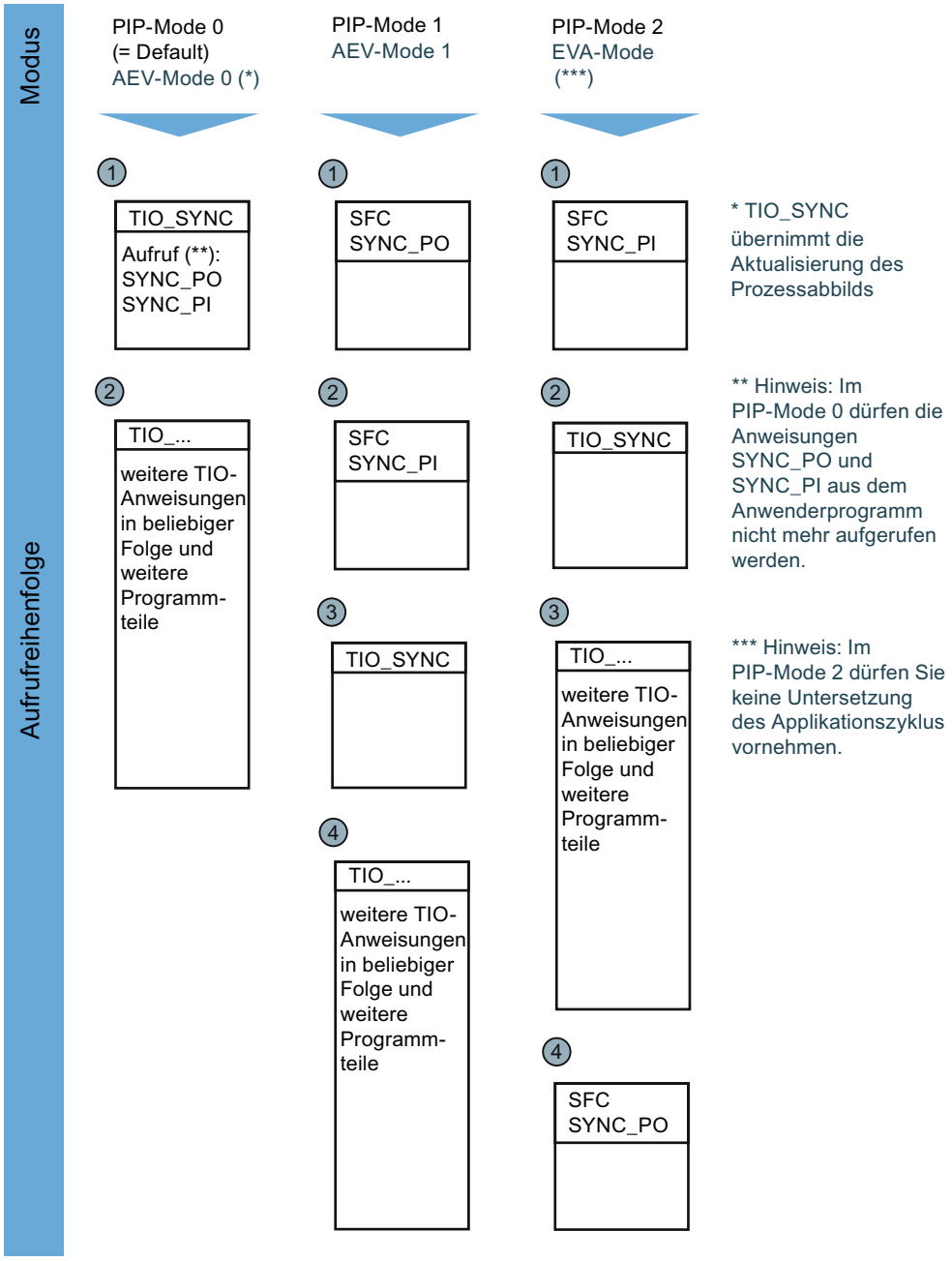

6. Fügen Sie für Ihre Applikation die notwendigen TIO-Anweisungen für Dateneinlesen/-ausgabe in den OB "Synchronous Cycle" ein.
7. Verschalten Sie an den TIO-Anweisungen für Einlesen und Ausgabe jeweils den Eingang TIO_SYNC_Data mit dem gleichnamigen Ausgang am TIO_SYNC.

8. Parametrieren Sie die Eingangsparameter TO (siehe Eigenschaften der ET 200-Station), HWID (siehe Moduleigenschaften in HWCN) und Port/Channel für die TIO-Anweisungen für Dateneinlesen/-ausgabe.
   Die Funktionalität Time-based IO ist fertig programmiert.
9. Verschalten Sie die Funktionalität Time-based IO mit Ihrer Applikation.
10. Übersetzen und laden Sie das gesamte Projekt in die CPU.

Ergebnis

Sie haben die Programmierung zur Nutzung von Time-based IO abgeschlossen.

Beispiel

Im Folgenden sehen Sie ein Anwendungsbeispiel mit einem TIO_IOLink_IN und einem TIO_IOLink_OUT.

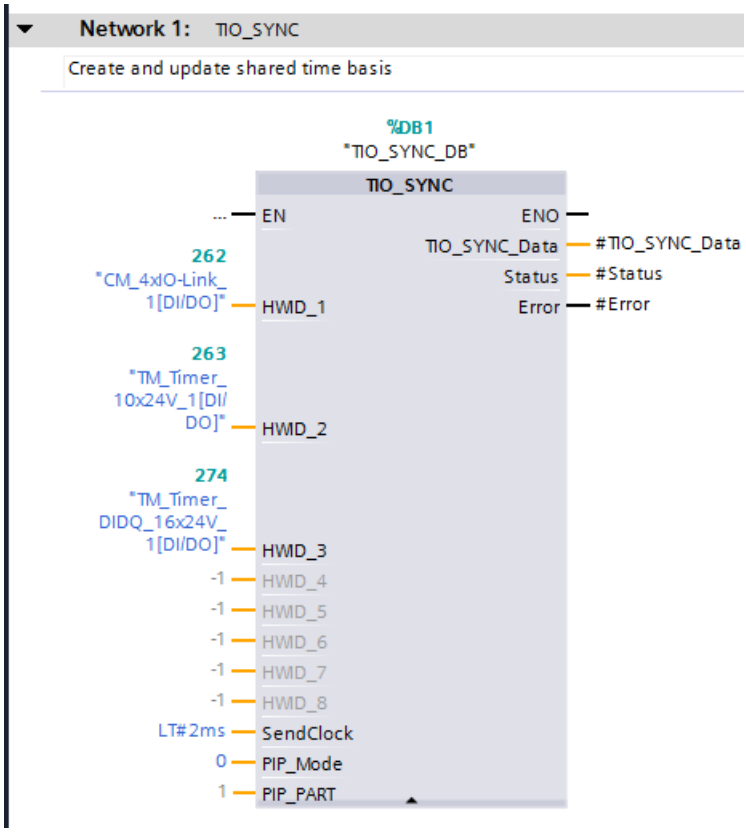

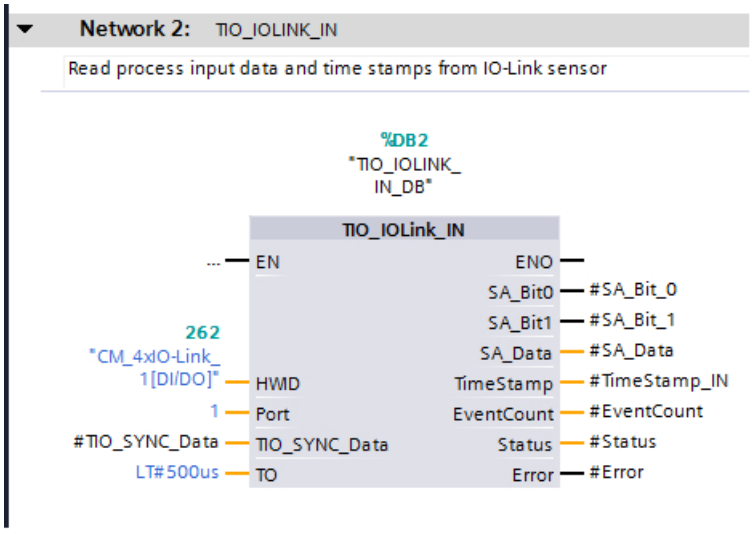
Network 2: TIO_IOLINK_IN
Read process input data and time stamps from IO-Link sensor
%DB2
"TIO_IOLINK_
IN_DB"
TIO_IOLink_IN
... EN
ENO
SA_Bit0 #SA_Bit_0
SA_Bit1 #SA_Bit_1
SA_Data #SA_Data
262
"CM_4xIO-Link_
1[DI/DO]" HWID
TimeStamp #TimeStamp_IN
1 Port
EventCount #EventCount
#TIO_SYNC_Data TIO_SYNC_Data
Status #Status
LT#500us TO
Error #Error

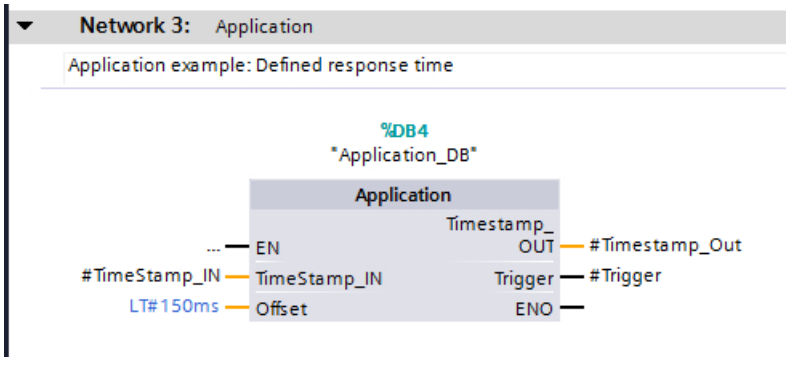
Network 3: Application
Application example: Defined response time
%DB4
"Application_DB"
Application
Timestamp_
... EN
OUT #Timestamp_Out
#TimeStamp_IN TimeStamp_IN
Trigger #Trigger
LT#150ms Offset
ENO

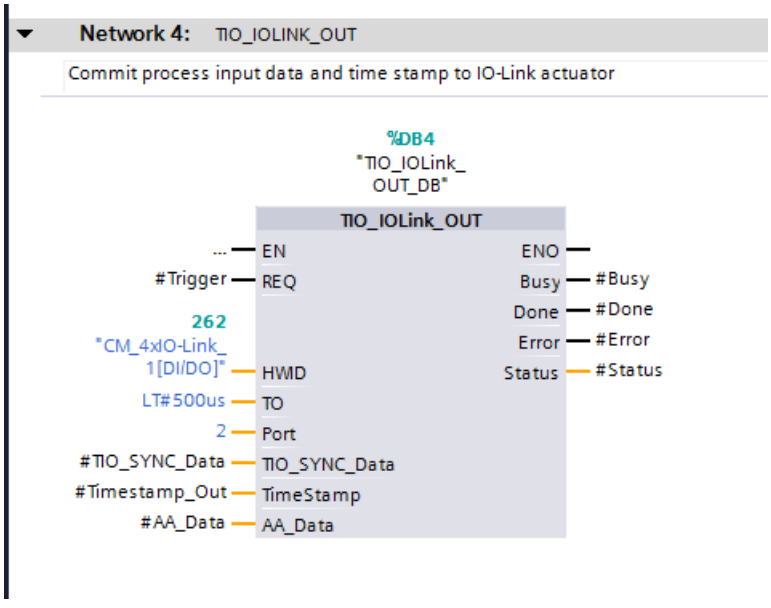
Network 4: TIO_IOLINK_OUT
Commit process input data and time stamp to IO-Link actuator
%DB4
"TIO_IOLink_
OUT_DB"
TIO_IOLink_OUT
... EN
ENO
#Trigger REQ
Busy #Busy
Done #Done
262
Error #Error
"CM_4xIO-Link_
1[DI/DO]" HWID
Status #Status
LT#500us TO
2 Port
#TIO_SYNC_Data TIO_SYNC_Data
#Timestamp_Out TimeStamp
#AA_Data AA_Data

FAQ

Weitere Informationen hierzu finden Sie in den folgenden FAQs im Siemens Industry Online Support:

- Beitrags-ID 109738186
- Beitrags-ID 109736374

4.1.4 Technologiemodul einsetzen

4.1.4.1 TIO-Modul konfigurieren und parametrieren

Technologiemodul zur Hardware-Konfiguration hinzufügen

Voraussetzungen

- Das Projekt ist angelegt.
- Die CPU ist angelegt.
- Dezentrale Peripherie ET 200 ist angelegt.

Vorgehen

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie das Technologiemodul aus dem Baugruppenkatalog aus:
   "Technologiemodule > Time-based IO > Technologiemodul > Artikelnummer"
4. Ziehen Sie das Technologiemodul per Drag & Drop auf den gewünschten Steckplatz im Baugruppenträger.

Ergebnis

In der Projektnavigation wird unter "Lokale Baugruppen" bzw. "Dezentrale Peripherie" das neue Technologiemodul angezeigt.

Hardware-Konfiguration (HWCN) öffnen

Öffnen über Projektnavigation

Gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Lokale Baugruppen" bzw. "Dezentrale Peripherie".
2. Doppelklicken Sie in der Projektnavigation auf das Technologiemodul.

Öffnen über Gerätesicht

Gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie auf das Technologiemodul.

Grundparameter

Kanalkonfiguration (TM Timer DIDQ 16x24V)

Mit der Kanalkonfiguration legen Sie für TM Timer DIDQ 16x24V fest, welche Kombination aus Digitaleingängen und -ausgängen Sie verwenden möchten.

Sie können unter folgenden Konfigurationen auswählen:

- 0 Eingänge, 16 Ausgänge (voreingestellt)
- 3 Eingänge, 13 Ausgänge
- 4 Eingänge, 12 Ausgänge
- 8 Eingänge, 8 Ausgänge

Hinweis

Ungenutzte Ausgänge stehen als Geberversorgung zur Verfügung.

Ungenutzte Eingänge stehen als Digitaleingänge ohne Technologiefunktion zur Verfügung. Die Eingangsverzögerung eines Eingangs legen Sie gemeinsam mit den Parametern des entsprechenden Ausgangskanals fest.

PWM-Periode

Mit diesem Parameter legen Sie fest, welche Periodendauer die Signale an Digitalausgängen mit der Betriebsart "PWM" haben.

Sie können unter folgenden Periodendauern auswählen:

- 10 ms (voreingestellt)
- 5 ms
- 2 ms
- 1 ms
- 0,5 ms
- 0,2 ms

Hinweis

Das Impuls-Periode-Verhältnis geben Sie in der Steuerschnittstelle vor. Das Impuls-Periode-Verhältnis ist ein Prozentwert und wird mit einer Genauigkeit von ca. 3 % ausgewertet.

Verhalten bei CPU-STOP

Mit diesem Parameter legen Sie das Verhalten des Technologiemoduls bei einem STOP der CPU fest.

Sie können unter folgenden Optionen auswählen:

Verhalten bei CPU-STOP	Bedeutung
Ersatzwert ausgeben (voreingestellt)	Das Technologiemodul gibt bis zum nächsten STOP-RUN-Übergang der CPU an den Digitalausgängen die parametrierten Ersatzwerte aus. Nach einem STOP-RUN-Übergang wird das Technologiemodul in seinen Anlaufzustand gesetzt: Die Zählwerte werden, falls Sie Zähler verwenden, auf 0 gesetzt und die Digitalausgänge schalten entsprechend der Parametrierung und der Sollwerte.
Letzten Wert halten	Das Technologiemodul gibt bis zum nächsten STOP-RUN-Übergang der CPU an den Digitalausgängen die Werte aus, die zum Zeitpunkt des Übergangs nach STOP gültig waren. Bei einer parametrierten Pulsweitenmodulation wird bis zum nächsten STOP-RUN-Übergang die letzte gültige Periodendauer mit dem letzten gültigen Impuls-Pause-Verhältnis ausgegeben. Nach einem STOP-RUN-Übergang wird das Technologiemodul in seinen Anlaufzustand gesetzt: Die Zählwerte werden, falls Sie Zähler verwenden, auf 0 gesetzt und die Digitalausgänge schalten entsprechend der Parametrierung und der Sollwerte.

Diagnosealarme freigeben

Das Technologiemodul löst bei bestimmten Fehlern Diagnosealarme aus. Es kann zusätzliche Diagnosealarme auslösen, wenn Sie die Freigabe der Diagnosealarme aktivieren. Diagnosealarme bearbeiten Sie in einem Alarm-OB.

Mit diesem Parameter legen Sie fest, ob das Technologiemodul die zusätzlichen Diagnosealarme bei Auftreten der jeweiligen Fehler auslöst.

Informieren Sie sich im Gerätehandbuch zum Technologiemodul, welche Fehler im Betrieb einen Diagnosealarm auslösen können. In der Voreinstellung sind die Diagnosealarme nicht freigegeben.

Kanalparameter

Übersicht

Die Digitaleingänge und -ausgänge werden in den Kanalparametern in Gruppen mit gruppenspezifischen Parametern unterteilt.

Jede Gruppe ist eine der folgenden Gruppenarten:

- DQ-Gruppe (Seite 1212)
- DQ/DI-Gruppe (Seite 1214)
- DI-Gruppe (Seite 1219)

Die verfügbaren Gruppenarten für ein TM Timer DIDQ 16x24V sind abhängig von der Kanalkonfiguration in den Grundparametern.

DQ-Gruppe

Sie können für die Digitalausgänge einer DQ-Gruppe jeweils folgende Parameter einstellen:

Betriebsart

Mit diesem Parameter legen Sie die technologische Funktion des jeweiligen Digitalausgangs fest.

Sie können unter folgenden Optionen auswählen:

Betriebsart	Bedeutung
Timer-DQ (voreingestellt)	Das Technologiemodul gibt am jeweiligen Digitalausgang zu definierten Zeitpunkten Flanken aus. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Nockenanwendung.
Oversampling	Der jeweilige Digitalausgang gibt pro Applikationszyklus (z. B. OB61) nacheinander 32 Zustände mit gleichmäßigem zeitlichen Abstand aus. Die 32 Zustände werden über die Steuerschnittstelle vorgegeben. Diese Einstellung erfordert einen taktsynchronen Betrieb.
PWM	Das Technologiemodul schaltet den jeweiligen Digitalausgang mit der in den Grundparametern vorgegebenen Periodendauer und dem Impuls-Periode-Verhältnis aus der Steuerschnittstelle. Das Impuls-Periode-Verhältnis ist ein Prozentwert und wird mit einer Genauigkeit von ca. 3 % ausgewertet.

Ersatzwert

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem STOP der CPU am jeweiligen Digitalausgang ausgeben soll.

Die Voreinstellung ist "0".

High-Speed-Ausgang (0.1 A)

Mit diesem Parameter legen Sie fest, ob der jeweilige Digitalausgang als schneller Gegentakt- oder als P-Schalter arbeitet.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Aktiviert (voreingestellt)	Der Digitalausgang arbeitet als schneller Gegentakt-Schalter und mit einem Nennlaststrom von 0,1 A belastbar. Ein Gegentakt-Schalter wird wechselweise nach DC 24 V und Masse geschaltet. Dadurch sind sehr schnelle Flanken möglich.
Deaktiviert	Der Digitalausgang arbeitet als 24 V-P-Schalter mit Bezug zu M und ist mit einem Nennlaststrom von 0,5 A belastbar.

DI/DQ invertieren (TM Timer DIDQ 16x24V)

Sie können die 24 V-Signale am Digitalausgang und am entsprechenden Digitaleingang gemeinsam invertieren, um sie an den Prozess anzupassen.

In der Voreinstellung werden die beiden Signale nicht invertiert.

DQ invertieren

Sie können das 24 V-Signal am Digitalausgang invertieren, um es an den Prozess anzupassen.

In der Voreinstellung wird das Signal nicht invertiert.

Eingangsverzögerung

Mit der Parametrierung der Eingangsverzögerung unterdrücken Sie eingekoppelte Störungen am Digitaleingang, der als Eingang ohne Technologiefunktion zur Verfügung steht. Signale mit einer Impulsdauer unterhalb der parametrierten Eingangsverzögerung werden unterdrückt.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
  (Eingangsverzögerung von 4 µs, Mindestimpulsbreite von 3 µs)
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden. Zur Erhöhung der Genauigkeit der Zeitstempelfunktion wird die Verwendung geschirmter Leitungen auch für größere Eingangsverzögerungen empfohlen.

Hinweis

Die Eingangsverzögerung unterliegt einer Toleranz, die Sie im Gerätehandbuch zum Technologiemodul in den technischen Daten finden.

DQ/DI-Gruppe

Eine DQ/DI-Gruppe besteht aus einem Digitalausgang und einem bzw. zwei Digitaleingängen. Sie können für die Gruppe folgende Parameter einstellen:

Konfiguration DQ/DI-Gruppe

Mit diesem Parameter legen Sie fest, ob der DI als Freigabe-Eingang für den DQ verwendet wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung	Weitere optionsspezifische Parameter
Timer-DQ mit Freigabe-Eingang (voreingestellt)	Das Technologiemodul gibt am DQ zeitgesteuert Flanken aus. Der DQ wird durch den DI freigegeben. Die Betriebsart des DQ ist "Timer-DQ" und nicht änderbar. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Nockenanwendung.	Ersatzwert High-Speed-Ausgang (0.1 A) DI/DQ invertieren (TM Timer DIDQ 16x24V) DQ invertieren (TM Timer DIDQ 10x24V) Eingangsverzögerung HW-Freigabe
Eingang/Ausgang einzeln verwenden	Der DQ und der DI werden unabhängig voneinander parametriert und verwendet.	Betriebsart (DQ) Ersatzwert High-Speed-Ausgang (0.1 A) DI/DQ invertieren (TM Timer DIDQ 16x24V) DQ invertieren (TM Timer DIDQ 10x24V) Betriebsart (DI) Eingangsverzögerung DI invertieren

Konfiguration DQ/DI-Gruppe (DQ2/DI2/DI3 des TM Timer DIDQ 10x24V)

Mit diesem Parameter legen Sie fest, wie DQ2, DI2 und DI3 des TM Timer DIDQ 10x24V verwendet werden.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung	Weitere optionsspezifische Parameter
Inkrementalgeber (A, B phasenversetzt) (voreingestellt)	DQ2 wird einzeln verwendet. An DI2 und DI3 ist ein Inkrementalgeber mit den phasenversetzten 24 V-Signalen A und B angeschlossen. Der Zähler (DI2) wertet die beiden Signale immer vierfach aus.	Betriebsart (DQ) Ersatzwert High-Speed-Ausgang (0.1 A) DQ invertieren Zählrichtung invertieren
Timer-DI2 mit Freigabe-Eingang DI3	Zu definierten Flanken an DI2 werden Zeitstempel erfasst. DI2 wird durch DI3 freigegeben. Die Betriebsart des DI2 ist "Timer-DI" und nicht änderbar. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Messtasteranwendung.	Betriebsart (DQ) Ersatzwert High-Speed-Ausgang (0.1 A) DQ invertieren Eingangsverzögerung DI invertieren HW-Freigabe
Timer-DQ2 mit Freigabe-Eingang DI2	Das Technologiemodul gibt an DQ2 zeitgesteuert Flanken aus. DQ2 wird durch DI2 freigegeben. Die Betriebsart des DQ2 ist "Timer-DQ" und nicht änderbar. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Nockenanwendung.	Ersatzwert High-Speed-Ausgang (0.1 A) DQ invertieren Eingangsverzögerung HW-Freigabe Betriebsart (DI) DI invertieren
Eingang/Ausgang einzeln verwenden	Der DQ und der DI werden unabhängig voneinander parametriert und verwendet.	Betriebsart (DQ) Ersatzwert High-Speed-Ausgang (0.1 A) DQ invertieren Betriebsart (DI) Eingangsverzögerung DI invertieren

Betriebsart (DQ)

Mit diesem Parameter legen Sie die technologische Funktion des Digitalausgangs fest.

Sie können unter folgenden Optionen auswählen:

Betriebsart	Bedeutung
Timer-DQ (voreingestellt)	Das Technologiemodul gibt am DQ zu definierten Zeitpunkten Flanken aus. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Nockenanwendung.
Oversampling	Das Technologiemodul gibt am DQ pro Applikationszyklus (z. B. OB61) 32 Zustände nacheinander mit gleichmäßigem zeitlichen Abstand aus. Die 32 Zustände werden über die Steuerschnittstelle vorgegeben. Diese Einstellung erfordert einen taktsynchronen Betrieb.
PWM	Das Technologiemodul schaltet den DQ mit der in den Grundparametern vorgegebenen Periodendauer und dem Impuls-Periode-Verhältnis aus der Steuerschnittstelle. Das Impuls-Periode-Verhältnis ist ein Prozentwert und wird mit einer Genauigkeit von ca. 3 % ausgewertet.

Ersatzwert

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem STOP der CPU am DQ ausgeben soll.

Die Voreinstellung ist "0".

High-Speed-Ausgang (0.1 A)

Mit diesem Parameter legen Sie fest, ob der DQ als schneller Gegentakt- oder als P-Schalter arbeitet.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung
Aktiviert (voreingestellt)	Der DQ arbeitet als schneller Gegentakt-Schalter und ist mit einem Nennlaststrom von 0,1 A belastbar. Ein Gegentakt-Schalter wird wechselweise nach DC 24 V und Masse geschaltet. Dadurch sind sehr schnelle Flanken möglich.
Deaktiviert	Der DQ arbeitet als 24 V-P-Schalter mit Bezug zu M und ist mit einem Nennlaststrom von 0,5 A belastbar.

DI/DQ invertieren (TM Timer DIDQ 16x24V)

Sie können die 24 V-Signale des DQ und des entsprechenden DI gemeinsam invertieren, um sie an den Prozess anzupassen.

In der Voreinstellung werden die Signale nicht invertiert.

DQ invertieren (TM Timer DIDQ 10x24V)

Sie können das 24 V-Signal des DQ invertieren, um es an den Prozess anzupassen.

In der Voreinstellung wird das Signal nicht invertiert.

Eingangsverzögerung

Mit diesem Parameter unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
  (Eingangsverzögerung von 4 µs, Mindestimpulsbreite von 3 µs)
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms

Die Eingangsverzögerung hat folgenden Einfluss auf die am Digitaleingang eingestellte Funktion zur Signalauswertung:

Funktion	Einfluss der Eingangsverzögerung
Timer-DI	Der erfasste Zeitstempel ist um die Eingangsverzögerung verschoben.
Inkrementalgeber (A, B phasenversetzt)	Zurückgeliefert wird der Zählwert, der gültig war zum Zeitpunkt T_i abzüglich der Eingangsverzögerung.
Zähler	
Oversampling	Die erfassten Zustände sind gemeinsam um die Eingangsverzögerung verschoben.

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden. Zur Erhöhung der Genauigkeit der Zeitstempelfunktion wird die Verwendung geschirmter Leitungen auch für größere Eingangsverzögerungen empfohlen.

Hinweis

Die Eingangsverzögerung unterliegt einer Toleranz, die Sie im Gerätehandbuch zum Technologiemodul in den technischen Daten finden.

HW-Freigabe

Mit diesem Parameter legen Sie fest, ob der Timer-DQ über den Pegel oder die Flanken am DI freigegeben wird.

Sie können unter folgenden Optionen auswählen:

Freigabe	Bedeutung	Weitere optionsspezifische Parameter
Pegelgesteuert (voreingestellt)	Der DQ wird über den Pegel des DI freigegeben.	Pegelauswahl für HW-Freigabe
Flankengesteuert	Diese Einstellung können Sie bei Verwendung des Steuerungssystems SIMOTION nutzen.	—

Das folgende Bild zeigt ein Beispiel für die Ausgabe von steigenden und fallenden Flanken an DQ0 bei Freigabe des DI0 durch den High-Pegel des DI1:

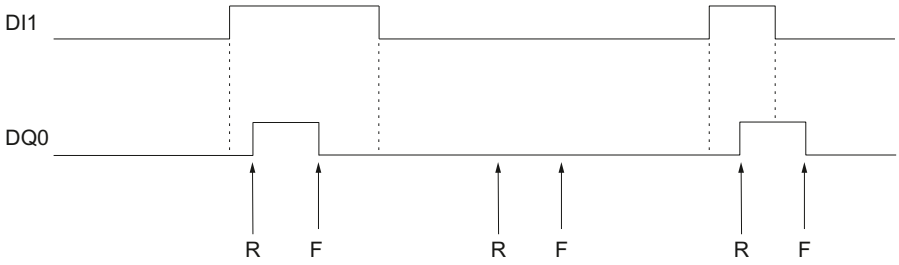

R Vorgegebener Zeitpunkt einer steigenden DQ0-Flanke

F Vorgegebener Zeitpunkt einer fallenden DQ0-Flanke

Pegelauswahl für HW-Freigabe

Mit diesem Parameter legen Sie für eine pegelgesteuerte HW-Freigabe den Pegel fest, bei dem der DQ freigegeben ist.

Sie können unter folgenden Optionen auswählen:

Pegel	Bedeutung
Aktiv bei High-Pegel (voreingestellt)	Der DQ ist freigegeben, wenn der DI gesetzt ist.
Aktiv bei Low-Pegel	Der DQ ist freigegeben, wenn der DI nicht gesetzt ist.

Betriebsart (DI)

Mit diesem Parameter legen Sie die technologische Funktion des DIm+1 fest.

Sie können unter folgenden Optionen auswählen:

Betriebsart	Bedeutung
Zähler	Am DI ist ein Impulsgeber oder Sensor angeschlossen. Der Zähler zählt die steigenden oder die fallenden Flanken am DI. Informieren Sie sich im Gerätehandbuch zum Technologiemodul, welche Digitaleingänge Sie als Zähler verwenden können.
Timer-DI (voreingestellt)	Zu definierten Flanken am DI werden Zeitstempel zugehörigen Zeitstempel eingelesen. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Messtasteranwendung.
Oversampling	Das Technologiemodul liest den Zustand des DI pro Applikationszyklus (z. B. OB61) zu 32 Zeitpunkten mit gleichmäßigem zeitlichen Abstand. Die 32 Zustände werden gemeinsam zurückgeliefert. Diese Einstellung erfordert einen taktsynchronen Betrieb.

DI invertieren

Sie können das 24 V-Signal des DI invertieren, um es an den Prozess anzupassen.

In der Voreinstellung wird das Signal nicht invertiert.

DI-Gruppe

Eine DI-Gruppe besteht aus zwei benachbarten Digitaleingängen DIm und DIm+1. Sie können für die Gruppe folgende Parameter einstellen:

Konfiguration DI-Gruppe

Mit diesem Parameter legen Sie fest, wie die Gruppe verwendet wird.

Sie können unter folgenden Optionen auswählen:

Option	Bedeutung	Weitere optionsspezifische Parameter
Inkrementalgeber (A, B phasenversetzt) (voreingestellt)	An den beiden Digitaleingängen ist ein Inkrementalgeber mit den phasenversetzten 24 V-Signalen A und B angeschlossen. Der Zähler der DI-Gruppe wertet die beiden Signale immer vierfach aus.	Zählrichtung invertieren
Timer-DI mit Freigabe-Eingang	Zu definierten Flanken an DIm werden Zeitstempel erfasst. DIm wird durch DIm+1 freigegeben. Die Betriebsart des DIm ist "Timer-DI" und nicht änderbar. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Messtasteranwendung.	Eingangsverzögerung Invertieren HW-Freigabe
Eingänge einzeln verwenden	Die beiden Digitaleingänge werden unabhängig voneinander parametriert und verwendet.	Betriebsart Eingangsverzögerung

Zählrichtung invertieren

Sie können die Zählrichtung invertieren, um sie an den Prozess anzupassen. Das Technologiemodul invertiert dabei das Signal B des Inkrementalgebers an DIm+1. Der jeweilige Zähler interpretiert die Flanken beider Signale dann als Rückwärtsimpulse.

Das folgende Bild zeigt ein Beispiel für Zählimpulse in Vorwärts- und Rückwärtsrichtung:

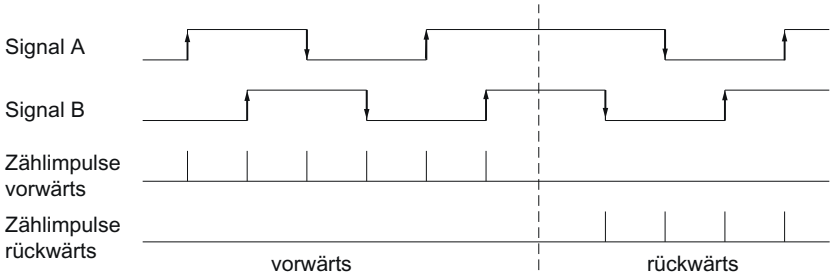

In der Voreinstellung wird die Zählrichtung nicht invertiert.

Eingangsverzögerung

Mit diesem Parameter unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
  (Eingangsverzögerung von 4 µs, Mindestimpulsbreite von 3 µs)
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms

Die Eingangsverzögerung hat folgenden Einfluss auf die am Digitaleingang eingestellte Funktion zur Signalauswertung:

Funktion	Einfluss der Eingangsverzögerung
Timer-DI	Der erfasste Zeitstempel ist um die Eingangsverzögerung verschoben.
Inkrementalgeber (A, B phasenversetzt)	Zurückgeliefert wird der Zählwert, der gültig war zum Zeitpunkt T_i abzüglich der Eingangsverzögerung.
Zähler	
Oversampling	Die erfassten Zustände sind gemeinsam um die Eingangsverzögerung verschoben.

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden. Zur Erhöhung der Genauigkeit der Zeitstempelfunktion wird die Verwendung geschirmter Leitungen auch für größere Eingangsverzögerungen empfohlen.

Hinweis

Die Eingangsverzögerung unterliegt einer Toleranz, die Sie im Gerätehandbuch zum Technologiemodul in den technischen Daten finden.

Invertieren

Sie können das 24 V-Signal des DIm invertieren, um es an den Prozess anzupassen.

In der Voreinstellung wird das Signal nicht invertiert.

HW-Freigabe

Mit diesem Parameter legen Sie fest, ob der Timer-DI über den Pegel oder die Flanken an DIm+1 freigegeben wird.

Sie können unter folgenden Optionen auswählen:

Freigabe	Bedeutung	Weitere optionsspezifische Parameter
Pegelgesteuert (voreingestellt)	DIm wird über den Pegel des DIm+1 freigegeben.	Pegelauswahl für HW-Freigabe
Flankengesteuert	Diese Einstellung können Sie bei Verwendung des Steuerungssystems SIMOTION nutzen.	—

Das folgende Bild zeigt ein Beispiel für die Erfassung von Zeitstempeln zu steigenden und fallenden Flanken bei Freigabe des DI0 durch den High-Pegel des DI1:

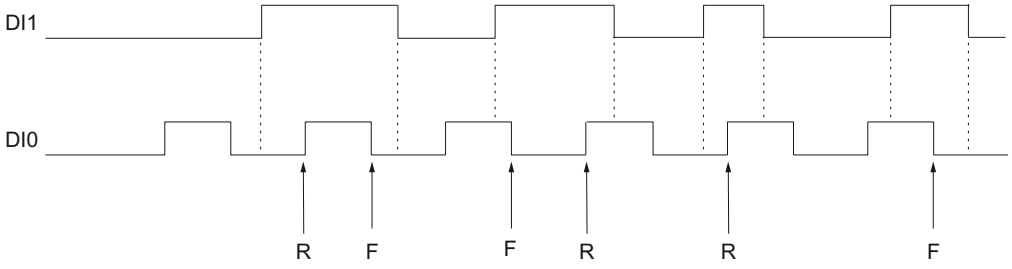

R Erfassung des zugehörigen Zeitstempels zu steigender DI0-Flanke

F Erfassung des zugehörigen Zeitstempels zu fallender DI0-Flanke

Pegelauswahl für HW-Freigabe

Mit diesem Parameter legen Sie für eine pegelgesteuerte HW-Freigabe den Pegel fest, bei dem DIm freigegeben ist.

Sie können unter folgenden Optionen auswählen:

Pegel	Bedeutung
Aktiv bei High-Pegel (voreingestellt)	DIm ist freigegeben, wenn der DIm+1 gesetzt ist.
Aktiv bei Low-Pegel	DIm ist freigegeben, wenn der DIm+1 nicht gesetzt ist.

Betriebsart

Mit diesem Parameter legen Sie die technologische Funktion des jeweiligen Digitaleingangs fest.

Sie können unter folgenden Optionen auswählen:

Betriebsart	Bedeutung	Weitere optionsspezifische Parameter
Zähler	An DIm ist ein Impulsgeber oder Sensor angeschlossen. Der Zähler zählt die steigenden oder die fallenden Flanken an DIm. Informieren Sie sich im Gerätehandbuch zum Technologie-modul, welche Digitaleingänge Sie als Zähler verwenden können.	Signalauswertung für Zähler
Timer-DI (voreingestellt)	Zu definierten Flanken am jeweiligen Digitaleingang werden die zugehörigen Zeitstempel eingelesen. Diese Einstellung nutzt Time-based IO und erfordert einen taktsynchronen Betrieb. Bei Verwendung des Steuerungssystems SIMOTION nutzen Sie diese Einstellung für eine Messtasteranwendung.	Invertieren
Oversampling	Das Technologiemodul erfasst den Zustand des jeweiligen Digitaleingangs pro Applikationszyklus (z. B. OB61) zu 32 Zeitpunkten mit gleichmäßigem zeitlichen Abstand. Die 32 Zustände werden gemeinsam zurückgeliefert. Diese Einstellung erfordert einen taktsynchronen Betrieb.	Invertieren

Signalauswertung für Zähler

Mit diesem Parameter legen Sie fest, ob der Zähler die steigenden oder die fallenden Flanken an DIm zählt.

In der Voreinstellung werden die steigenden Flanken gezählt.

4.1.4.2 TIO-Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über das Technologiemodul erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie mit der rechten Maustaste auf das Technologiemodul und wählen Sie "Online & Diagnose".
4. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch des Technologiemoduls.

Impulsausgabe 5

5.1 Grundlagen zur Impulsausgabe

5.1.1 Übersicht der Module und Eigenschaften

Module

Die folgende Tabelle gibt für das System S7-1500 eine Übersicht über die Leistungsmerkmale der Module für Impulsausgabe.

Eigenschaft	S7-1500		
	Technologiemodul	Kompakt-CPU	Technologiemodul
	TM Timer DIDQ 16x24V	CPU 1511C-1 PN, CPU 1512C-1 PN	TM PTO 4
Anzahl Impulsgeneratoren	16	4	4
Betriebsart Oversampling	X	—	—
Betriebsart Pulsweitenmodulation PWM	X	X (Seite 1299)	—
Betriebsart Impulsausgabe	—[1]	—	—
Betriebsart Impulskette	—[1]	—	—
Betriebsart Ein-/Ausschaltverzögerung	—	—	—
Betriebsart Frequenzausgabe	—	X (Seite 1299)	—
Betriebsart PWM mit DC-Motor	—	—	—
Betriebsart Pulse Train Output (PTO)	—	X (Seite 1299)	X
Sequenzzähler	—	—	—
High-Speed-Ausgang möglich	X	X (Seite 1310)	—
Dithering möglich	—	—	—
Strommessung/-regelung möglich	—	—	—
Unterstützung Motion Control	—	X	X
HW-Freigabe	—	—	X
Parallelbetrieb von 2 Kanälen	—	—	—
Steuerung Digitalausgänge über Anwenderprogramm	X	X (Seite 1311)	—
Unterstützung Taktsynchronität am PROFINET	X	—	X
Unterstützung Diagnosealarme	X	X	X

[1] Eine vergleichbare Funktion können Sie mit der Timer-Funktion realisieren.

Die folgende Tabelle gibt für das System ET 200SP eine Übersicht über die Leistungsmerkmale der Module für Impulsausgabe.

Eigenschaft	ET 200SP		
	Technologiemodul		Digitalausgabemodul
	TM Timer DIDQ 10x24V	TM Pulse 2x24V	DQ 4x24VDC/ 2A HS
Anzahl Impulsgeneratoren	6	2	4
Betriebsart Oversampling	X	—	X
Betriebsart Pulsweitenmodulation PWM	X	X	X
Betriebsart Impulsausgabe	—[1]	X	—
Betriebsart Impulskette	—[1]	X	—
Betriebsart Ein-/Ausschaltverzögerung	—	X	—
Betriebsart Frequenzausgabe	—	X	—
Betriebsart PWM mit DC-Motor	—	X	—
Betriebsart Pulse Train Output (PTO)	—	—	—
Sequenzzähler	—	X	—
High-Speed-Ausgang möglich	X	X	—
Dithering möglich	—	X	—
Strommessung/-regelung möglich	—	X	—
Unterstützung Motion Control	—	—	—
HW-Freigabe	—	X	—
Parallelbetrieb von 2 Kanälen	—	X	—
Steuerung Digitalausgänge über Anwenderprogramm	X	X	X
Unterstützung Taktsynchronität am PROFINET	X	X	X
Unterstützung Diagnosealarme	X	X	X

[1] Eine vergleichbare Funktion können Sie mit der Timer-Funktion realisieren.

5.1.2 Grundlagen zur Impulsausgabe (TM Pulse 2x24V)

5.1.2.1 Übersicht

Betriebsarten und Funktionen

Das TM Pulse 2x24V besitzt zwei Kanäle. Sie konfigurieren die Betriebsarten mit STEP 7 (TIA-Portal) oder STEP 7.

Sie können für jeden Kanal eine der folgenden sechs Betriebsarten auswählen:

- Impulsausgabe
- Pulsweitenmodulation PWM
- Impulskette
- Ein-/Ausschaltverzögerung
- Frequenzausgabe
- PWM mit DC-Motor

Neben der Betriebsart besitzt das Technologiemodul die folgenden Funktionen:

Funktion	Erläuterung
High-Speed-Ausgang	Sie können eine Mindestimpulsdauer von 1,5 µs bei einem Strom von 100 mA erzeugen. Ohne diese Funktion beträgt die Mindestimpulsdauer 10 µs bei einem Strom von maximal 2 A (zwei Kanäle) oder 4 A (ein Kanal).
Sequenzzähler	Der Sequenzzähler zählt abgeschlossene Ausgabesequenzen und gibt ein Rückmeldesignal aus.
Strommessung	Für die Betriebsarten PWM und Impulskette
Stromregelung	Für den Ausgangsstrom in der Betriebsart PWM
Dithering	Überlagerung eines Dither-Signals am PWM-Ausgang zur Verbesserung einer Proportionalventilsteuerung
Steuerung über Anwenderprogramm	Sie können die Digitalausgänge direkt über das Anwenderprogramm kanalweise steuern.
Parallelschaltung der Kanäle	Eine Parallelschaltung beider Kanäle ermöglicht einen logischen Kanal, der einen Ausgangsstrom von 4 A antreiben kann.
Verhalten bei CPU-STOP	Sie können die Reaktion auf einen CPU-STOP konfigurieren.
Diagnose	Das Technologiemodul ermöglicht die Moduldiagnose und Kanalfehlererkennung.

Das folgende Bild zeigt den Zweikanalbetrieb.

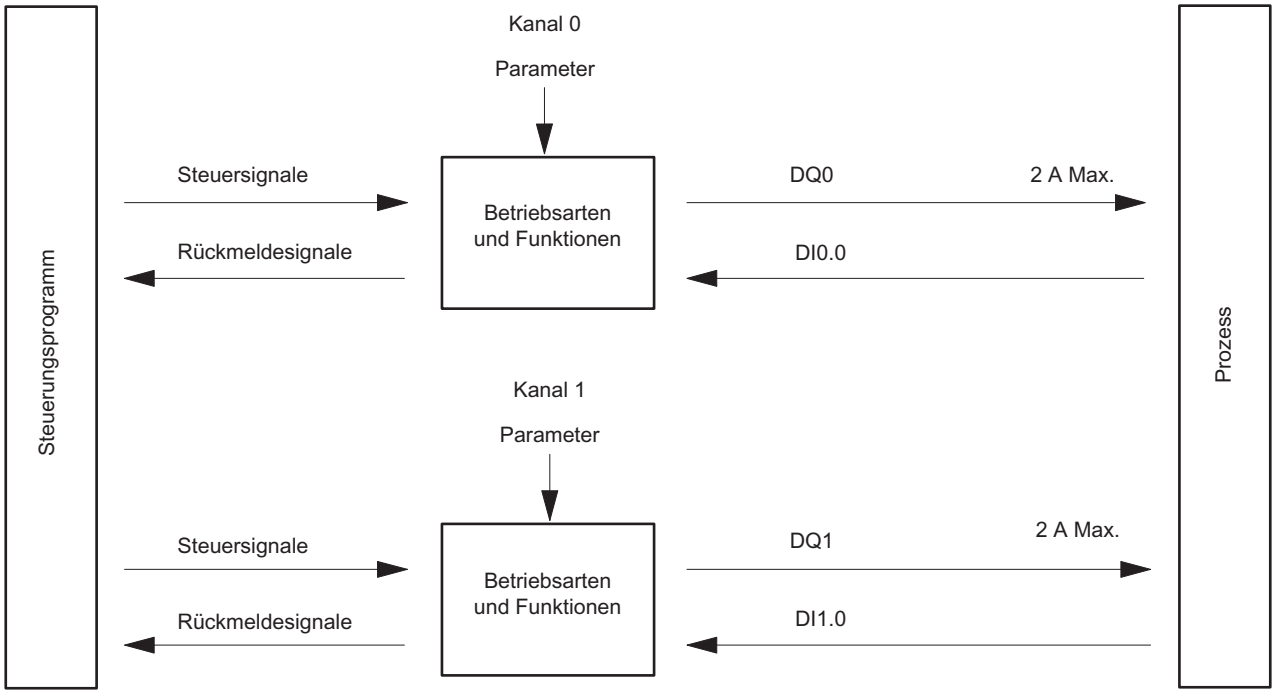

Bild 5-1 Zweikanalbetrieb

Das folgende Bild zeigt den Einkanalbetrieb.

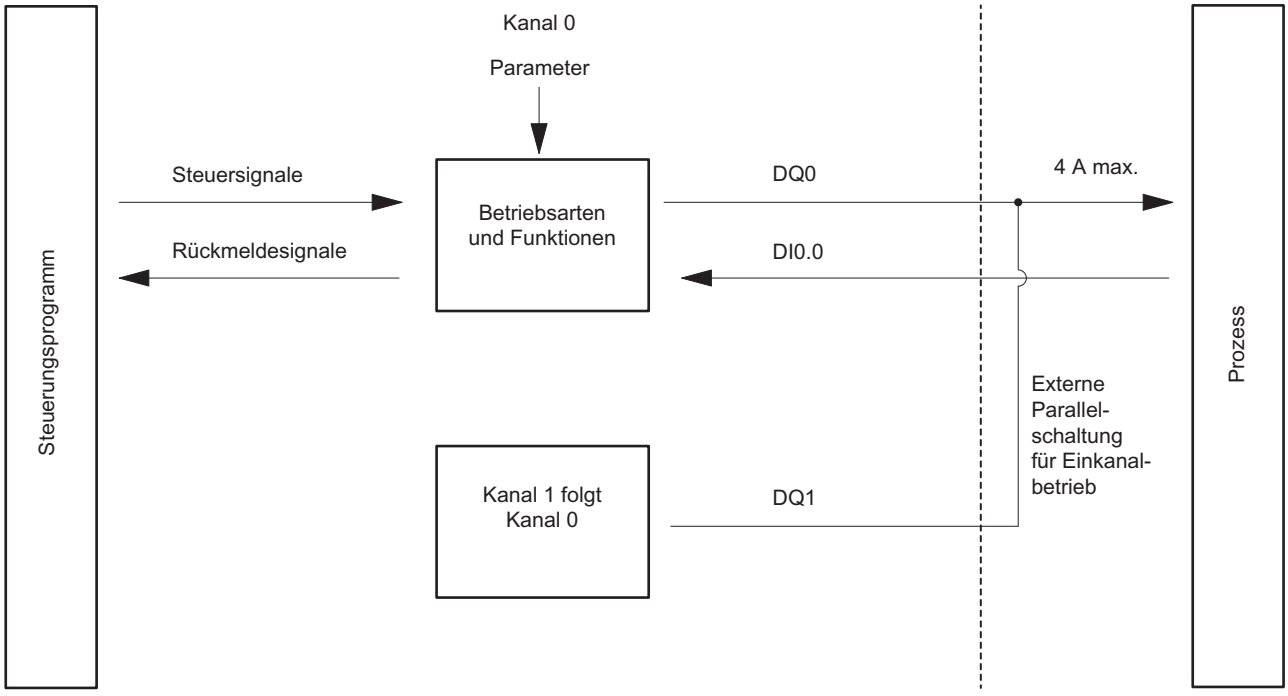

Bild 5-2 Einkanalbetrieb

Schnittstellen zum Steuerungsprogramm und der gesteuerte Prozess

Das TM Pulse 2x24V besitzt die folgenden I/O BaseUnit-Pinanschlüsse zum gesteuerten Prozess:

Channel 0:

- DI0.0 (Digitaleingang 0)
- DQ0.A und DQ0.B (Digitalausgang 0)
  - Für jeden Kanalausgang stehen zwei Anschlüsse (DQ0.A und DQ0.B) zur Verfügung. Die Lastverdrahtung hängt von der Betriebsartenzuweisung eines Kanals ab.

Channel 1:

- DI1.0 (Digitaleingang 1)
- DQ1.A und DQ1.B (Digitalausgang 1)

Eine Tabelle mit Konfigurationsparametern finden Sie im Abschnitt "Parameterzuweisung und Struktur der Parameterdatensätze".

Die Betriebsarten und Funktionen können mithilfe von Steuer- und Rückmeldesignalen im Steuerungsprogramm geändert und überwacht werden. Diese Parameter finden Sie im Abschnitt "Steuer- und Rückmeldeschnittstelle".

In den nachfolgenden Kapiteln finden Sie:

- Beschreibung von Operationen
- Die relevanten Parameter
- Die relevanten Steuer- und Rückmeldesignale

Die "Beschreibung von Operationen" bei Betriebsarten und Funktionen gilt für beide Kanäle. Die Begriffe DIn.0, DQn.A und DQn.B werden in Beschreibungen verwendet, die auf beide Kanäle zutreffen.

5.1.2.2 Betriebsart Impulsausgabe (Einzelimpuls)

Definition

Sobald die zugewiesene Einschaltverzögerung abgelaufen ist, gibt TM Pulse 2x24V einen Impuls am Digitalausgang DQn.A (Ausgabesequenz) für die von Ihnen festgelegte Impulsdauer aus.

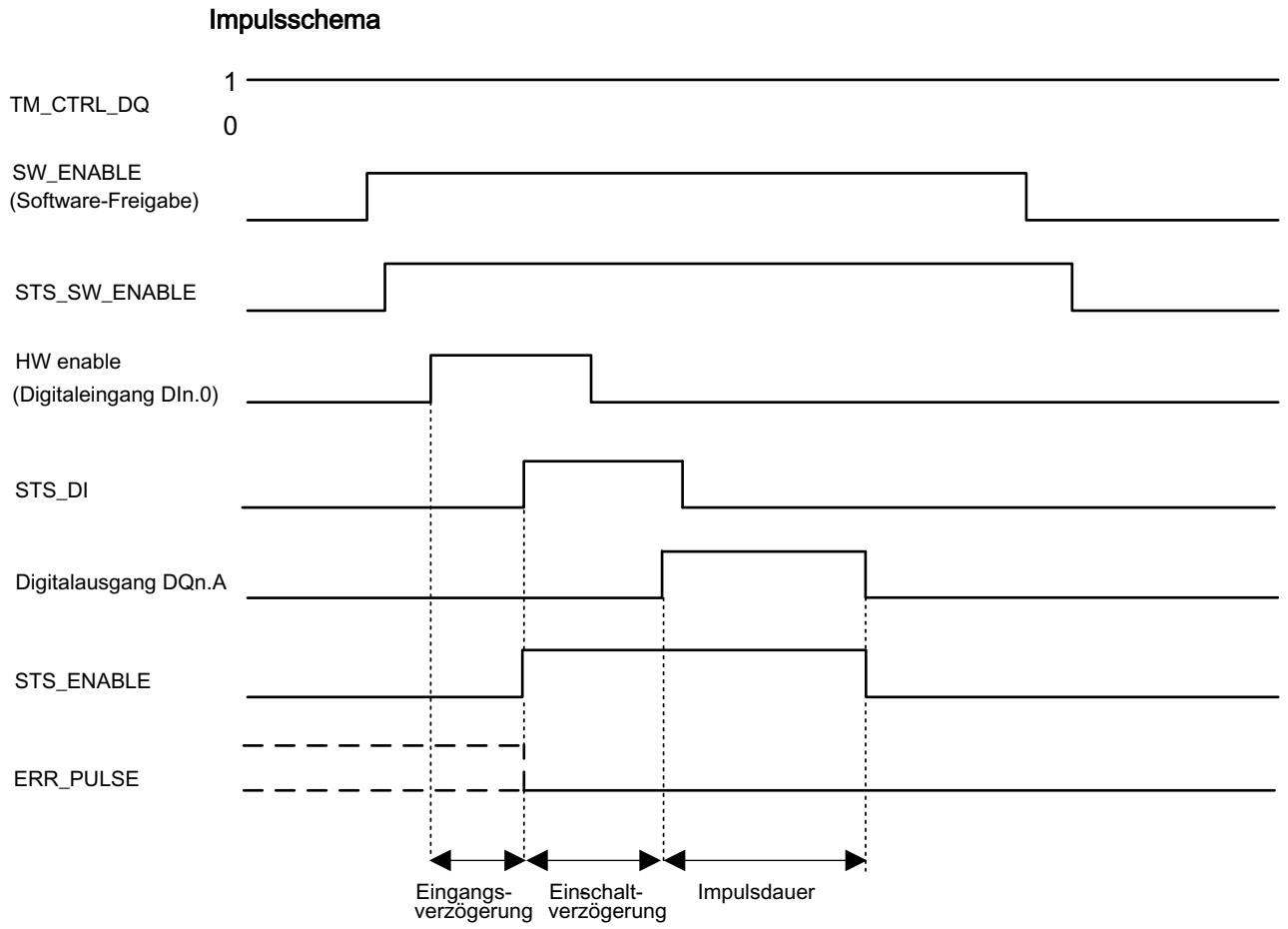

Bild 5-3 Die Ausgabesequenz der Betriebsart Impulsausgabe nutzt ein optionales HW-Freigabesignal zum Starten der Ausgabesequenz

Das obige Impulsschema hat den auf "HW-Freigabe" eingestellten Parameter "Funktion DI". Bei der anderen Option wird "Funktion DI" auf "Eingang" gesetzt. Wenn der Parameter "Funktion DI" auf "Eingang" gesetzt ist, beginnt die Einschaltverzögerung mit der steigenden Flanke von SW_ENABLE.

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe ausgeben (SW_ENABLE 0 → 1.)

Das Rückmeldebit STS_SW_ENABLE gibt an, dass eine Software-Freigabe im TM Pulse 2x24V anhängig ist.

Außerdem können Sie den Digitaleingang DIn.0 eines TM Pulse 2x24V-Kanals als Hardware-Freigabe (HW freigeben) über den Parameter "DI-Funktion" einstellen. Die Eingangsverzögerung (Störfilter) der Hardware-Freigabe kann über den Parameter "Eingangsverzögerung" eingestellt werden.

Wenn Sie die Hardware-Freigabe verwenden möchten, muss diese mit der Software-Freigabe kombiniert werden. Bei aktivierter Software-Freigabe startet die Ausgabesequenz bei der ersten steigenden Flanke der Hardware-Freigabe. Weitere steigende Flanken der Hardware-Freigabe werden während der aktuellen Ausgabesequenz ignoriert. Wenn die HW-Freigabe

in den "High"-Zustand (steigende Flanke) geht und für die Dauer der Eingangsverzögerung "High" ("Ein") bleibt, wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Sobald die Einschaltverzögerung abgelaufen ist, wird der Impuls mit der zugewiesenen Impulsdauer ausgegeben. Die Ausgabesequenz wird mit dem Impulsende beendet und STS_ENABLE wird gelöscht.

Wenn Sie die Impulsdauer auf eine bereits abgelaufene Dauer reduzieren, dann gibt das ERR_PULSE-Signal einen Impulsausgabefehler aus und die Ausgabesequenz wird angehalten. Um die Impulsausgabe fortzusetzen, müssen Sie die Ausgabesequenz nach Auftreten eines ERR_PULSE-Fehlers neu starten. Mit dem nächsten Start der Ausgabesequenz löscht das TM Pulse 2x24V das Rückmeldebit ERR_PULSE.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ des Technologiemoduls

- Wenn TM_CTRL_DQ = 1, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen an den DQn.A-Ausgängen.
- Wenn TM_CTRL_DQ = 0, dann hat das CPU die Steuergewalt und das Programm kann die DQn.A/DQn.B-Ausgänge direkt mit den Steuerbits SET_DQA/SET_DQBeinstellen.

Abbrechen der Ausgabesequenz

Bei Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang DQn.A werden umgehend auf 0 gesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Wahrheitstabelle

Software-Freigabe SW_ENABLE	Funktionsparameter DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.A (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	HW_ENABLE	0 → 1 und bleibt 1 während der Eingangsverzögerung. Nur aktiv bei der ersten steigenden Flanke, weitere steigenden Flanken werden ignoriert, kein Start.	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0 → 1	Eingang	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0	HW_ENABLE oder Eingang	Beliebiger Zustand	0	0	Abbrechen

Software-Freigabe SW_ENABLE	Funktionsparameter DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.A (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	HW_ENABLE oder Eingang	Beliebiger Zustand	0, wenn Einschaltverzögerung nicht abgelaufen ist oder wenn die Impulsdauer abgelaufen ist. 1, wenn die Einschaltverzögerung abgelaufen ist und die Impulsdauer nicht abgelaufen ist.		-
0 → 1	HW_ENABLE	0	0	0	-

Einstellen und Ändern der Impulsdauer

Das Steuerungsprogramm kann die Impulsdauer direkt über das Steuerschnittstellenfeld OUTPUT_VALUE als ein DWord-Zahlenwert in Mikrosekunden einstellen:

- Schnelle Impulsausgänge aktiviert, von 2 µs bis 85.000.000 µs
- Schnelle Impulsausgänge deaktiviert, von 10 µs bis 85.000.000 µs

Wenn Sie die Impulsdauer bei laufender Ausgabesequenz ändern, wird die bereits abgelaufene Impulsdauer von der neuen Impulsdauer abgezogen und die Impulsausgabe wird fortgesetzt.

Verringern der Impulsdauer

Wenn Sie die Impulsdauer auf eine Dauer verringert haben, die kleiner ist als die bereits abgelaufene Impulsdauer, dann wird die Ausgabesequenz abgebrochen. STS_ENABLE und der Digitalausgang DQn.A werden gelöscht und das Statusbit ERR_PULSE wird eingestellt. Mit der nächsten Ausgabesequenz wird das Statusbit ERR_PULSE gelöscht.

Einstellen und Ändern der Einschaltverzögerung

- Permanente Aktualisierung
  Die Einschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss 1 sein (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs im Steuerschnittstellenfeld SLOT ein.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs in den Konfigurationsparametern ein. Alternativ können Sie eine einzelne Aktualisierung über die Steuerschnittstelle ausführen. MODE_SLOT muss 0 sein (einzelne Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs im Steuerschnittstellenfeld SLOT ein.

Wenn Sie den Wert der Einschaltverzögerung während der Ausgabesequenz ändern, dann wird die neue Einschaltverzögerung mit der nächsten Ausgabesequenz aktiviert.

Weitere Informationen zur Verwendung des SLOT-Wertes finden Sie unter Umgang mit dem SLOT-Parameter.

Verwenden des Sequenzzählers

Der Sequenzzähler zählt abgeschlossene Impulsausgabesequenzen und stellt die Zählerwertvariable SEQ_CNT in der Rückmeldeschnittstelle bereit. Der Sequenzzähler kann kurze Ausgabesequenzen zählen, die zu schnell ablaufen und daher nicht vom Programm überwacht werden können. Nähere Informationen finden Sie unter Sequenzzähler.

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter Funktion: Taktsynchroner Betrieb.

Der taktsynchrone Betrieb hat keinen Einfluss auf die Funktionalität der Betriebsart Impulsausgabe.

Wenn Sie die Ausgabesequenz mit T_o synchronisieren möchten, dann stellen Sie den Parameter "Funktion DI"auf "Eingang" und die Impulsausgabesequenz startet bei T_o.

Impulsausgabeparameter

Impulsausgabeparameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	0 = Einstellen der Betriebsart Impulsausgabe	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = DC-Motor	1
Schnelle Impulsausgabe [1]	Der Ausgang unterstützt im aktivierten Zustand höhere Frequenzen bei kleineren Lasten.	0 = Deaktiviert 1 = Aktiviert	Deaktiviert
Funktion DI	Sie können den Digitaleingang DIn.0 als Eingang oder als Hardware-Freigabe verwenden. Das Signal am DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = Eingang 1 = HW-Freigabe	Eingang
Eingangsverzögerung	DIn.0: Ein Digitaleingang muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Einschaltverzögerung	Die Dauer vom Start der Ausgabesequenz bis zur Impulsausgabe. Sie können die Einschaltverzögerung im Steuerungsprogramm mithilfe des SLOT-Parameters ändern.	0 µs bis 85.000.000 µs	0 µs

[1] Nur wenn das Modul als "2-kanalig (2 A)" konfiguriert ist.

Steuer- und Rückmeldesignale für die Betriebsart Impulsausgabe

Steuerschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung				
Channel 0	Channel 1 [1]						
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (DWord)	Impulsdauer: Die Dauer, während der der Digitalausgang DQn.A eingestellt bleibt, nachdem die Einschaltverzögerungszeit abgelaufen ist. Wenn Sie die Unter- oder Obergrenze des Bereichs missachten, dann wird ERR_OUT_VAL in der Rückmeldeschnittstelle ausgegeben und der zuletzt gültige Wert wird verwendet.				
			Schnelle Impulsausgänge deaktiviert:	**Schnelle Impulsausgänge aktiviert:**			
			10 µs bis 85.000.000 µs	2 µs bis 85.000.000 µs			
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Die Einschaltverzögerung kann vor dem Start der Ausgabesequenz geändert werden. Siehe MODE_SLOT.				
			0 µs bis 85.000.000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation des Wertes SLOT: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	1	0	Einschaltverzögerung in µs
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Betriebsart zur Verwendung des Feldes SLOT.			
			0	Betriebsart für einzelne Aktualisierung			
			1	Betriebsart für permanente Aktualisierung			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software-Freigabe: Starten/Aktivieren und Beenden/Deaktivieren der Ausgabesequenz.			
			0	Ausgabe deaktiviert/beendet			
			0 → 1	Startet Ausgabesequenz bei steigender Flanke wenn "Funktion DI" = "Eingang"			
			1	Aktivieren der Ausgabesequenz, wenn der Start abhängt von HW-Freigabe mit "Funktion DI" = "HW-Freigabe"			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Einstellen der DQn.A-Ausgabequelle: Auswählen des CPU-Programms oder der Ausgabesequenz des Moduls.			
			0	DQn.A und DQn.B werden von dem CPU (Logik Ihres Programm) mithilfe der Steuerbits SET_DQA und SET_DQB gesteuert.			
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert. DQn.B ist immer 0.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ gelöscht wird.			
			0	0 bei DQn.A			
			1	1 bei DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ und SET_DQA gelöscht werden.			
			0	0 bei DQn.B			
			1	1 bei DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB, und ERR_24V).			
			0	Rücksetzen von Fehlern ist nicht aktiv			
			1	Rücksetzen von Fehlern ist aktiv			

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Channel 0	Channel 1[1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf
			1	PWR wurde erkannt, weist jedoch Unterspannung auf
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am Ausgang 24 V DC an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der Betriebsart für einzelne Aktualisierung SLOT).
			0	Kein Lastfehler anstehend.
			1	Lastfehler anstehend: Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Zeigt einen Impulsausgabefehler an.
			0	Kein Impulsausgabefehler
			1	Impulsausgabefehler
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.B an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung beider DQs mithilfe von SET_DQA, SET_DQB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B oder die versuchte Einstellung von DQn.A und DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Zeigt an, dass ein ungültiger Wert in OUTPUT_VALUE erkannt wurde.
			0	OUTPUT_VALUE ist gültig.
			1	OUTPUT_VALUE ist ungültig. Dieses Bit wird automatisch zurückgesetzt, wenn ein gültiger Wert vom Modul ausgelesen wird.
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass ein ungültiger Wert in SLOT erkannt wurde (nur in der Betriebsart für permanente Aktualisierung SLOT).
			0	SLOT-Wert ist gültig.
			0 → 1	SLOT-Wert ist ungültig. Dieses Bit wird automatisch zurückgesetzt, wenn ein gültiger Wert vom Modul ausgelesen wird.
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Channel 0	Channel 1[1]			
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE eingestellt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine laufende Ausgabesequenz an.
			0	Ausgabesequenz läuft nicht
			1	Ausgabesequenz läuft
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 am Digitalausgang DQn.A
			1	1 am Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 am Digitalausgang DQn.B
			1	1 am Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	0 am Digitaleingang DIn.0
			1	1 am Digitaleingang DIn.0
Byte 3: Bit 0 … 3	Byte 11: Bit 0 … 3	SEQ_CNT	Der Sequenzzähler wird nach Abschluss einer Ausgabesequenz erhöht	
			Bei SW_ENABLE: 0 bis 1	
			Bei HW-Freigabe: 0 bis 15	
Word 3	Word 7	Reserviert	Als 0 lesen	

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen bytes und bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für die Betriebsart Impulsausgabe

Eingangs- und Ausgangssignale	Bedeutung	Wertebereich	Channel 0 BaseUnit Anzahl Pins	Channel 1 BaseUnit Anzahl Pins
Eingangssignal				
HW-Freigabe	Sie können die HW-Freigabe mit dem Parameter "Funktion DI" auswählen und die Eingangsverzögerung mit dem Parameter "Eingangsverzögerung“ auswählen. Das Signal am DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe aktiviert 0 → 1 = Start der Ausgabesequenz nach der Eingangsverzögerung in Abhängigkeit von der Software-Freigabe (SW_ENABLE)	3	4
Ausgangssignal				
Impuls am Digitalausgang DQn.A	Ein Impuls wird am Digitalausgang DQn.A für die eingestellte Impulsdauer ausgegeben.	0 = kein Impuls 1 = Impuls	9	10

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.3 Betriebsart Impulsweitenmodulation (PWM)

Definition

Sie steuern die Einschaltdauer der Impulsweite über das Feld OUTPUT_VALUE der Steuerschnittstelle. Das TM Pulse 2x24V erzeugt anhand dieses Wertes kontinuierliche Impulse. Das Steuerschnittstellenfeld OUTPUT_VALUE bestimmt die Einschaltdauer (Impulsdauer/Periodendauer) für PWM. Die Periodendauer ist anpassbar.

Nach Ablauf der zugewiesenen Einschaltverzögerung beginnen die DQn.A-Ausgangsimpulse (Ausgabesequenz).

In der Betriebsart PWM können Sie die Stromregelung mithilfe der PID-Schleifenfunktion aktivieren, um den Ausgangslaststrom zu regeln. Bei aktivierter Stromregelung übernimmt das TM Pulse 2x24V die Regelung der Einschaltdauer und das Steuerschnittstellenfeld OUTPUT_VALUE dient zur Zuweisung des Zielstroms als Wert für das Verhältnis von Zielstrom zu Referenzstrom.

Impulsschema

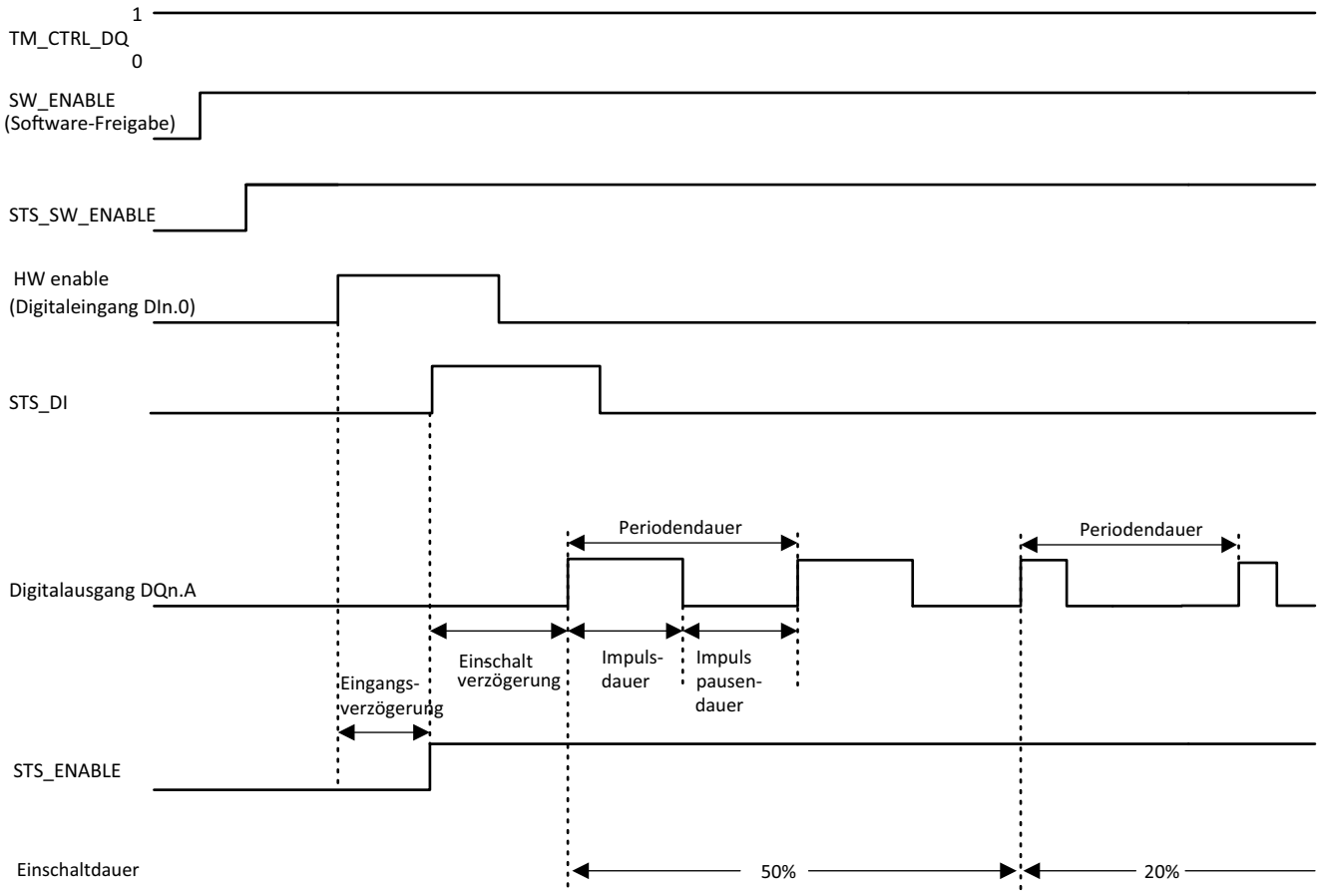

Bild 5-4 PWM-Ausgabesequenz

Im obigen Impulsschema ist der Parameter „Funktion DI" auf „HW_ENABLE“ eingestellt. Bei der anderen Option wird "Funktion DI" auf "Eingang" gesetzt. Wenn der Parameter "Funktion

DI" auf "Eingang" gesetzt ist, beginnt die Einschaltverzögerung mit der steigenden Flanke von SW_ENABLE.

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe ausgeben (SW_ENABLE 0 → 1).

Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am TM Pulse 2x24V ansteht.

Außerdem können Sie den Digitaleingang DIn.0 des TM Pulse 2x24V als HW-Freigabe über den Parameter "Funktion DI" zuweisen. Die Eingangsverzögerung (Störfilter) der Hardware-Freigabe kann über den Parameter "Eingangsverzögerung" eingestellt werden.

Wenn Sie die Hardware-Freigabe verwenden möchten, muss diese mit der Software-Freigabe kombiniert werden. Bei aktivierter Software-Freigabe startet die Ausgabesequenz bei der ersten steigenden Flanke der Hardware-Freigabe. Weitere steigende Flanken der Hardware-Freigabe werden während der aktuellen Ausgabesequenz vom TM Pulse 2x24V ignoriert. Die Hardware-Freigabeoption wird im taktsynchronen Betrieb nicht unterstützt.

Wenn die Freigabe aktiviert ist (steigende Flanke) und für die Dauer der Eingangsverzögerung "High" ("Ein") bleibt, wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Mit Ablauf der Einschaltverzögerung wird die PWM-Impulskette ausgegeben. Die Ausgabesequenz läuft kontinuierlich, solange SW_ENABLE gesetzt ist.

Hinweis

Ausgabesteuersignal TM_CTRL des Technologiemoduls

- Wenn TM_CTRL_DQ = 1 ist, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen an den DQn.A-Ausgängen.
- Wenn TM_CTRL_DQ = 0 ist, übernimmt die CPU die Steuerung und das Programm kann DQn.A/DQn.B-Ausgänge direkt über die Steuerbits SET_DQA/SET_DQB einstellen.

Abbrechen der Ausgabesequenz

Durch Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang DQn.A werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Wahrheitstabelle

Software-Freigabe SW_ENABLE	Funktionsparameter DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.A (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	HW_ENABLE	0 → 1 und bleibt 1 während der Eingangsverzögerung. Nur aktiv bei der ersten steigenden Flanke, weitere steigenden Flanken werden ignoriert, es erfolgt kein Start.	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0 → 1	Eingang	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0	HW_ENABLE oder Eingang	Beliebiger Zustand	0	0	Abbrechen
1	HW_ENABLE oder Eingang	Beliebiger Zustand	0, wenn die Einschaltverzögerung nicht abgelaufen ist oder der Zeitpunkt zwischen den Impulsdauern liegt 1, wenn die Einschaltverzögerung abgelaufen und die Impulsdauer noch anhält		-
0 → 1	HW_ENABLE	0	0	0	-

Mindestimpulsdauer und MIndestimpulspause

Die Mindestimpulsdauer und die MIndestimpulspause überlagern die Merkmale des Proportionalausgangs.

Die Mindestimpulsdauer und die MIndestimpulspause werden mithilfe des Parameters "Mindestimpulsdauer" zugewiesen und haben immer denselben Wert.

- Eine vom TM Pulse 2x24V ermittelte Impulsdauer, die kürzer als die Mindestimpulsdauer ist, wird unterdrückt.
- Eine vom TM Pulse 2x24V ermittelte Impulsdauer, die länger als die Periodendauer minus der Mindestimpulspause ist, wird auf den Wert der Periodendauer (Einschaltdauer 1.000 ‰) gesetzt.

Hinweis

PWM-Stromregelung und Mindestimpulsdauer

Wenn die Stromregelung vom Stromsteuerbit im Parametrierdatensatz aktiviert wird, dann werden die parametrierte Mindestimpulsdauer und die MIndestimpulspause ignoriert.

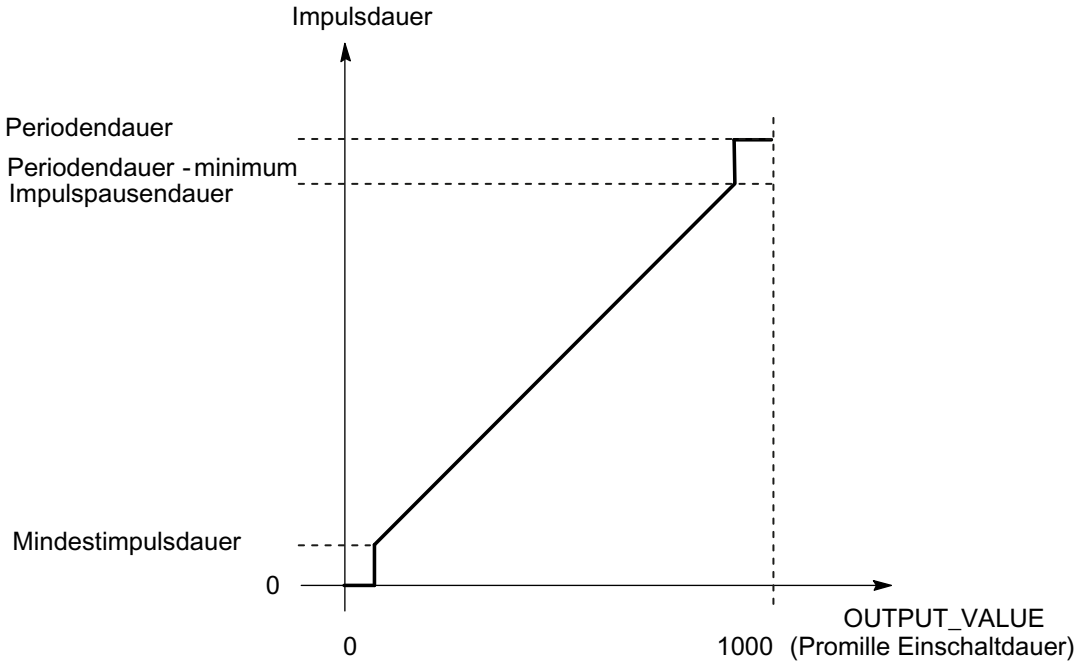

Bild 5-5 Impulsdauermodulation

Hinweis

Wenn die Dithering-Option ohne Stromregelung aktiviert ist, werden die Mindestimpulsdauer und die MIndestimpulspause vom Modul verwendet. In diesem Fall wird die Überlagerung durch das Dithering reduziert, sodass die effektive Impulsdauer in den zulässigen Bereich passt.

Einstellen und Ändern der Impulseinschaltdauer

OUTPUT_VALUE weist die Einschaltdauer für die aktuelle Periodendauer zu. Sie wählen den Bereich des Feldes OUTPUT_VALUE der Steuerschnittstelle mit dem Parameter "Ausgabeformat" aus.

- Ausgabeformat "Pro 100 (%)": Wertebereich zwischen 0 und 100
  Impulsdauer = (OUTPUT_VALUE/100) x Periodendauer.
- Ausgabeformat "Pro 1.000 (‰)": Wertebereich zwischen 0 und 1.000
  Impulsdauer = (OUTPUT_VALUE/1.000) x Periodendauer.
- Ausgabeformat "Pro 10.000 (‰)": Wertebereich zwischen 0 und 10.000
  Impulsdauer = (OUTPUT_VALUE/10.000) x Periodendauer.
- Ausgabeformat "S7-Analogausgabe": Wertebereich zwischen 0 und 27.648
  Impulsdauer = (OUTPUT_VALUE/27.648) x Periodendauer.

Sie weisen OUTPUT_VALUE direkt über das Steuerungsprogramm zu. Ein neuer OUTPUT_VALUE wird bei der nächsten steigenden Flanke am Ausgang übernommen.

Bei aktivierter Stromregelung übernimmt das TM Pulse 2x24V-Modul die Steuerung der Einschaltdauer und das Steuerschnittstellenfeld OUTPUT_VALUE dient zur Zuweisung des

Zielstroms als Wert für das Verhältnis von Zielstrom zu Referenzstrom. Nähere Informationen finden Sie unter der Funktion Stromregelung.

Einstellen und Ändern der Periodendauer

- Permanente Aktualisierung
  Die Periodendauer wird über die Steuerschnittstelle permanent gesteuert. Das MODE_SLOT-Bit muss gesetzt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben.
  Stellen Sie den Periodenwert im Feld SLOT ein. Die Einheit ist immer die Mikrosekunde.
  - Schnelle Impulsausgänge **aktiviert**: zwischen 10 µs und 85.000.000 µs im Feld SLOT.
  - Schnelle Impulsausgänge **deaktiviert**: zwischen 100 µs und 85.000.000 µs im Feld SLOT.
- Einzelne Aktualisierung
  Stellen Sie die Periodendauer in den Konfigurationsparametern ein. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden ("0" bedeutet einzelne Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben. Stellen Sie den Wert der Periodendauer im Feld SLOT ein. Die Einheit ist immer die Mikrosekunde.
  - Schnelle Impulsausgänge **aktiviert**: zwischen 10 µs und 85.000.000 µs in den Parametern.
  - Schnelle Impulsausgänge **deaktiviert**: zwischen 100 µs und 85.000.000 µs in den Parametern.

Die neue Periodendauer wird bei der nächsten steigenden Flanke des Ausgangs übernommen.

Weitere Informationen zur Handhabung des Parameters SLOT finden Sie unter "Handhabung des SLOT-Parameters (Steuerschnittstelle)".

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter "Funktion: Taktsynchroner Betrieb".

Im taktsynchronen Betrieb wird die Ausgabesequenz mit dem Moment T_O synchronisiert. Die Periodendauer wird auf den Anwendungszyklus abgestimmt (d. h. den synchronen Zyklus als Vielfaches des PROFINET-Zyklus). Die zugewiesene Periodendauer wird möglicherweise nicht genau implementiert. Der konfigurierte Wert der Periodendauer wird vom TM Pulse 2x24V an den Anwendungszyklus mithilfe eines Berechnungsalgorithmus angepasst. Die Berechnung dient dazu, die Differenz zwischen der zugewiesenen und der berechneten Periodendauer zu minimieren. Im ungünstigsten Fall beträgt die Abweichung einen halben Anwendungszyklus. Die folgende Tabelle zeigt Beispiele dazu.

Anwendungszyklus T_{CAC} [1]	Zugewiesene Periodendauer $T_{Sollwert}$	$T_{CAC} \setminus T_{Istwert}$	Berechnete tatsächliche Periodendauer T_{ist} [1]	
10 ms (10000 µs)	5000 µs	2:1	5000 µs	
10 ms (10000 µs)	2000 µs	5:1	2000 µs	
10 ms (10000 µs)	3000 µs	3:1	3333 µs	Nächster möglicher Wert wird als Istwert verwendet
10 ms (10000 µs)	1800 µs	6:1	1666 µs	Nächster möglicher Wert wird als Istwert verwendet

Anwendungszyklus T_{CAC} [1]	Zugewiesene Periodendauer $T_{Sollwert}$	$T_{CAC} \setminus T_{Istwert}$	Berechnete tatsächliche Periodendauer T_{Ist} [1]	
10 ms (10000 µs)	6000 µs	2:1	5000 µs	Nächster möglicher Wert wird als Istwert verwendet
10 ms (10000 µs)	12000 µs	1:1	10000 µs	Nächster möglicher Wert wird als Istwert verwendet
10 ms (10000 µs)	16000 µs	1:2	20000 µs	Nächster möglicher Wert wird als Istwert verwendet
10 ms (10000 µs)	26000 µs	1:3	30000 µs	Nächster möglicher Wert wird als Istwert verwendet

[1] Die ermittelte tatsächliche Periodendauer und die Einschaltdauer der Anwendung stehen immer in einem ganzzahligen Verhältnis (1:1, 1:2, 1:3, ..., 2:1, 3:1, ...) zueinander, das auf den nächstmöglichen Wert abgerundet wird.

Der Takt zwischen dem Digitalausgang und dem Anwendungszyklus wird unten dargestellt. Das Ein/Aus-Verhältnis der DQ-Einschaltdauer wird in den Beispielen als 50 % dargestellt.

- Beispiel 1: Die Periodendauer 10000 µs entspricht der Anwendungszykluszeit 10 ms (10000 µs).

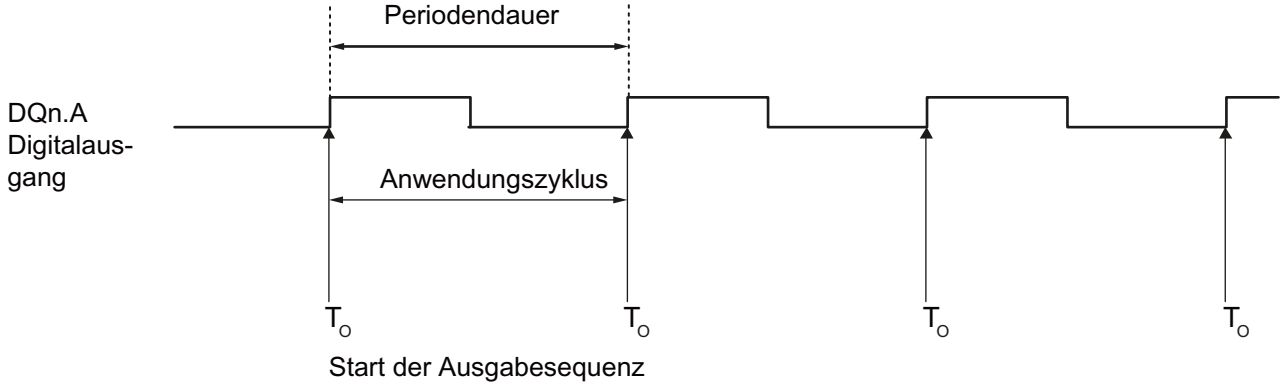

- Beispiel 2: Die Periodendauer 3333 µs ist kleiner als die Anwendungszykluszeit 10 ms (10000 µs).

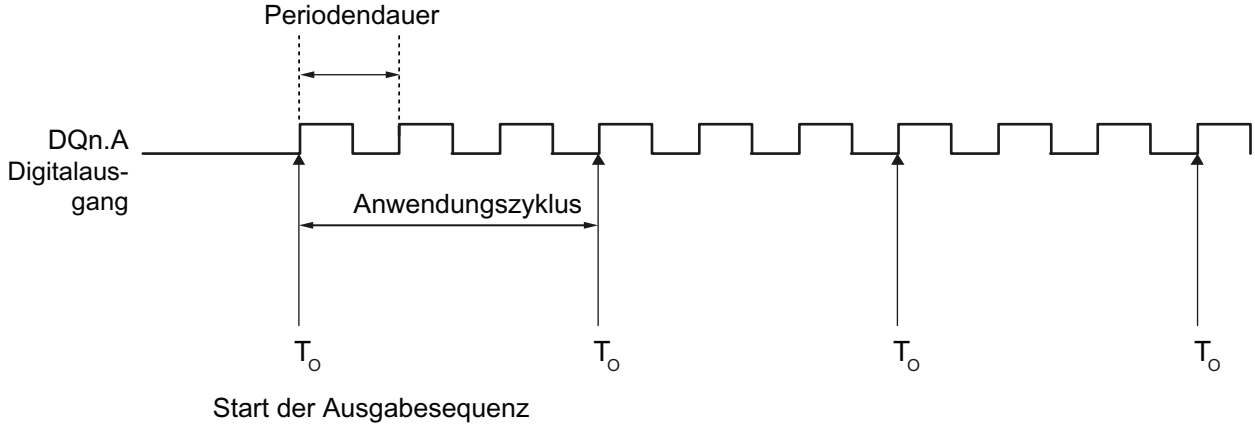

- Beispiel 3: Die Periodendauer 30000 µs ist größer als die Anwendungszykluszeit 10 ms (10000 µs).

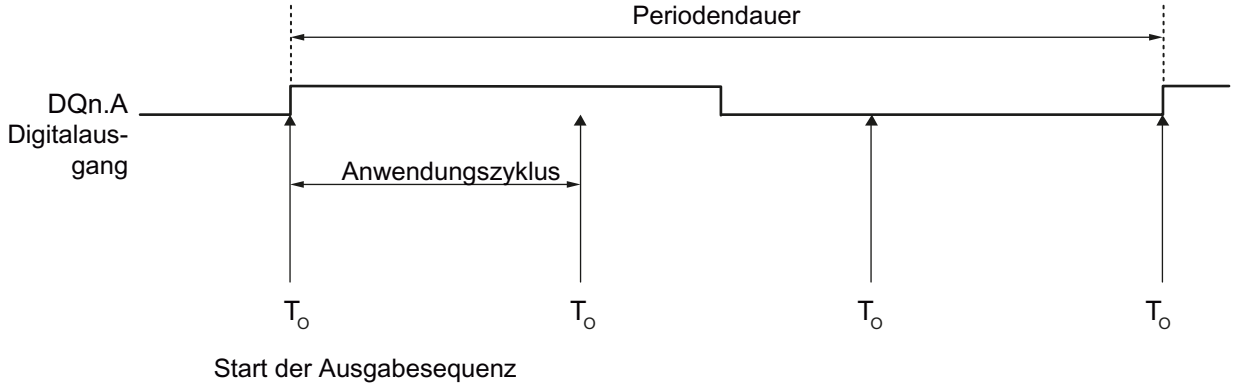

Hinweis

Taktsynchroner PWM-Betrieb

Die Einschaltverzögerung wird vom Modul nicht verwendet (d. h. immer als null betrachtet) und der Parameter "Funktion DI" ist immer "Eingang". Zum Starten der Ausgabesequenz wird nur die Software-Freigabe SW_ENABLE 0 → 1 verwendet.

Einstellen der Mindestimpulsdauer und der Mindestimpulspause

Sie weisen die Mindestimpulsdauer und die MIndestimpulspause als DWord-Zahlenwert zwischen 0 und 85.000.000 µs mithilfe der Kanalparameterkonfiguration "Mindestimpulsdauer" zu.

Die Einheit ist immer die Mikrosekunde. Dieser Wert kann nur über den Konfigurationsparameter-Datensatz geändert werden.

Einstellen und Ändern der Einschaltverzögerung

- Permanente Aktualisierung
  Die Einschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss eingestellt werden (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung im Feld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Die Einheit ist immer die Mikrosekunde.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltverzögerung in den Konfigurationsparametern auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Die Einheit ist immer die Mikrosekunde. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung im Feld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein.

Wenn Sie den Wert der Einschaltverzögerung während der Ausgabesequenz ändern, wird die neue Einschaltverzögerung mit der nächsten Ausgabesequenz aktiviert. Weitere Informationen zur Verwendung des Parameters SLOT finden Sie unter Handhabung des SLOT-Parameters.

Strommessung

Die Strommessung ist in der Betriebsart PWM verfügbar. Das Steuerungsprogramm kann die Strommessung zu Steuer- und Diagnosezwecken verwenden.

Weitere Informationen finden Sie im Abschnitt Funktion: Strommessung.

Stromregelung

Die Strommessung ist in der Betriebsart PWM verfügbar. Das Steuerungsprogramm kann die Strommessung zu Steuer- und Diagnosezwecken verwenden.

Weitere Informationen finden Sie im Abschnitt Funktion: Stromregelung.

Parameter der Betriebsart PWM

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	1 = Einstellen der Betriebsart PWM	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = DC-Motor	1
Schnelle Impulsausgabe [1]	Diese Ausgabe unterstützt höhere Frequenzen.	0 = deaktiviert 1 = aktiviert	Deaktiviert
Funktion DI	Sie können den Digitaleingang DIn.0 als Eingang oder als Hardware-Freigabe verwenden. Das Signal am DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = Eingang 1 = HW-Freigabe	Eingang
Ausgabeformat	Definiert das Format des Verhältniswertes (Einschaltdauer).	0 = S7-Analogformat 1 = Pro 100 (%) 2 = Pro 1.000 3 = Pro 10.000	Pro 100 (%)
Eingangsverzögerung	Der Digitaleingang DIn.0 muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Mindestimpulsdauer	Mindestimpulsdauer und MIndestimpulspause.	0 µs bis 85.000.000 µs	0 µs

Parameter	Bedeutung	Wertebereich	Voreinstellung
Periode	Periodendauer des Ausgangsimpulszyklus in µs. Sie können die Periodendauer im Steuerungsprogramm im Feld SLOT der Steuerschnittstelle ändern.	• Schnelle Impulsausgänge deaktiviert 100 µs bis 85.000.000 µs • Schnelle Impulsausgänge aktiviert 10 µs bis 85.000.000 µs	2.000.000 µs
Einschaltverzögerung	Die Dauer vom Start der Ausgabesequenz bis zur Impulsausgabe. Sie können die Einschaltverzögerung im Steuerungsprogramm im Feld SLOT der Steuerschnittstelle ändern.	0 µs bis 85.000.000 µs	0 µs

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Steuer- und Rückmeldesignale für Betriebsart PWM

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung				
Channel 0	Channel 1 [1]						
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (DWord)	Der OUTPUT_VALUE bestimmt die Einschaltdauer (Verhältnis Impulsdauer/Periodendauer) innerhalb einer Periode (Impulsweitenmodulation). Die Periodendauer ist anpassbar. Der neue Ausgabewert wird mit der nächsten steigenden Flanke der Ausgabe übernommen. Bei aktivierter Stromregelung übernimmt das TM Pulse 2x24V-Modul die Steuerung der Einschaltdauer und das Steuerschnittstellenfeld OUTPUT_VALUE dient zur Zuweisung des Zielstroms als Wert für das Verhältnis von Zielstrom zu Referenzstrom. Nähere Informationen finden Sie unter der Funktion Stromregelung. Datentyp UDInt: Es werden nur die 2 niederwertigsten Bytes verwendet Für Kanal 0: Bytes 2 und 3 Für Kanal 1: Bytes 14 und 15				
			Ausgabeformat	"Pro 100": Wertebereich ist 0 bis 100			
				"Pro 1000": Wertebereich ist 0 bis 1.000			
				"Pro 10000": Wertebereich ist 0 bis 10.000			
				"S7-Analogausgabe": Wertebereich ist 0 bis 27.648			
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Ihr Programm kann die in der folgenden Tabelle Byte 8 LD_SLOT aufgeführten Parameter vor dem Start der Ausgabesequenz mithilfe der Parameter SLOT und MODE_SLOT ändern.				
			0 µs bis 85.000.000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation des Wertes SLOT: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	Bit 2	Bit 1	Bit 0	Parameter
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	0	1	Periode in µs
			0	0	1	0	Einschaltverzögerung in µs
			0	1	0	1	Dither-Rampe in ms
			0	1	1	0	Dither-Amplitude in Promille
			0	1	1	1	Ditherperiode in µs

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung	
Channel 0	Channel 1 [1]			
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Betriebsart zur Verwendung des Feldes SLOT.
			0	Betriebsart für einzelne Aktualisierung
			1	Betriebsart für permanente Aktualisierung
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software-Freigabe: Starten und Beenden der Ausgabesequenz.
			0	Ausgabe abgebrochen
			0 → 1	Startet die Ausgabesequenz bei steigender Flanke, wenn "Funktion DI" = "Eingang".
			1	Aktivieren der Ausgabesequenz, wenn Start abhängig ist von HW-Freigabe mit "Funktion DI" = "HW-Freigabe".
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Einstellen der DQn.A-Ausgabequelle: Auswählen des CPU-Programms oder der Ausgabesequenz des Moduls.
			0	DQn.A und DQn.B werden vom CPU-Programm (in Ihrem Programm) mithilfe der Steuerbits SET_DQA und SET_DQB gesteuert.
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert. DQn.B ist immer 0 bei TM_CTRL_DQ = 1.
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ = 0.
			0	0 bei DQn.A
			1	1 bei DQn.A
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Steuert den Wert des Digitalausgangs, wenn DQn.B, wenn TM_CTRL_DQ = 0 und wenn SET_DQA gelöscht wird.
			0	0 bei DQn.B
			1	1 bei DQn.B
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB, und ERR_24V).
			0	Rücksetzen von Fehlern ist nicht aktiv
			1	Rücksetzen von Fehlern ist aktiv

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen bytes und bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Channel 0	Channel 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf
			1	PWR wurde erkannt, weist jedoch Unterspannung auf
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am Ausgang 24 V DC an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Channel 0	Channel 1 [1]			
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der Betriebsart für einzelne Aktualisierung SLOT).
			0	Kein Lastfehler anstehend
			1	Lastfehler anstehend: Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung von DQn.A und DQn.B mithilfe von SET_DQA, SET_DB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B, oder die versuchte Einstellung von DQn.A und DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass ein ungültiger Wert in SLOT erkannt wurde (nur in der Betriebsart für permanente AktualisierungSLOT).
			0	SLOT-Wert ist gültig
			0 → 1	SLOT-Wert ist ungültig
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE eingestellt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine laufende Ausgabesequenz an.
			0	Ausgabesequenz läuft nicht
			1	Ausgabesequenz läuft
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 am Digitalausgang DQn.A
			1	1 am Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 am Digitalausgang DQn.B
			1	1 am Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	0 am Digitaleingang DIn.0
			1	1 am Digitaleingang DIn.0
Byte 3: Bit 0 bis 3	Byte 11: Bit 0 bis 3	SEQ_CNT	Sequenzzähler = 0. Der Sequenzzähler wird in der Betriebsart Pulsweitenmodulation nicht verwendet.	

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung
Channel 0	Channel 1 [1]		
Word 2	Word 6	MEASURED_CURRENT	Positiver Wert S7-Analogformat (0 bis 32.767): • 2-Ampere-Kanalausgang: Messbereichsendwert 27.648; d. h. 2.000 mA • 4-Ampere-Kanalausgang: Messbereichsendwert 27.648; d. h. 4.000 mA • 32.767 bedeutet, dass keine gültige Strommessung verfügbar ist; z. B. während der ersten PWM-Periode.
Word 3	Word 7	Reserviert	Als 0 lesen

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen bytes und bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für Betriebsart PWM

Eingangs- und Ausgangssignal	Bedeutung	Wertebereich	Channel 0 BaseUnit Anzahl Pins	Channel 1 BaseUnit Anzahl Pins
Eingangssignal				
HW-Freigabe Hinweis: HW-Freigabe wird nicht unterstützt für PWM im taktsynchronen Betrieb	Sie können die HW-Freigabe mit dem Parameter "Funktion DI" auswählen und die Eingangsverzögerung mit dem Parameter "Eingangsverzögerung" auswählen. Das Signal am Digitaleingang DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe aktiviert 0 → 1 = Start der Ausgabesequenz nach der Eingangsverzögerung in Abhängigkeit von der Software-Freigabe (SW_ENABLE)	3	4
Ausgangssignal				
Impuls am Digitalausgang DQn.A	Ein Impuls wird am Digitalausgang DQn.A für die eingestellte Einschaltdauer und Periodendauer ausgegeben.	0 = kein Impuls 1 = Impuls	9	10

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.4 Betriebsart Impulskette

Definition

Nach Ablauf der zugewiesenen Einschaltverzögerung gibt das TM Pulse 2x24V die Anzahl der Impulse aus, die Sie als Impulskette (Ausgabesequenz) zugewiesen haben. Periodendauer und Impulsdauer sind anpassbar.

Impulsschema

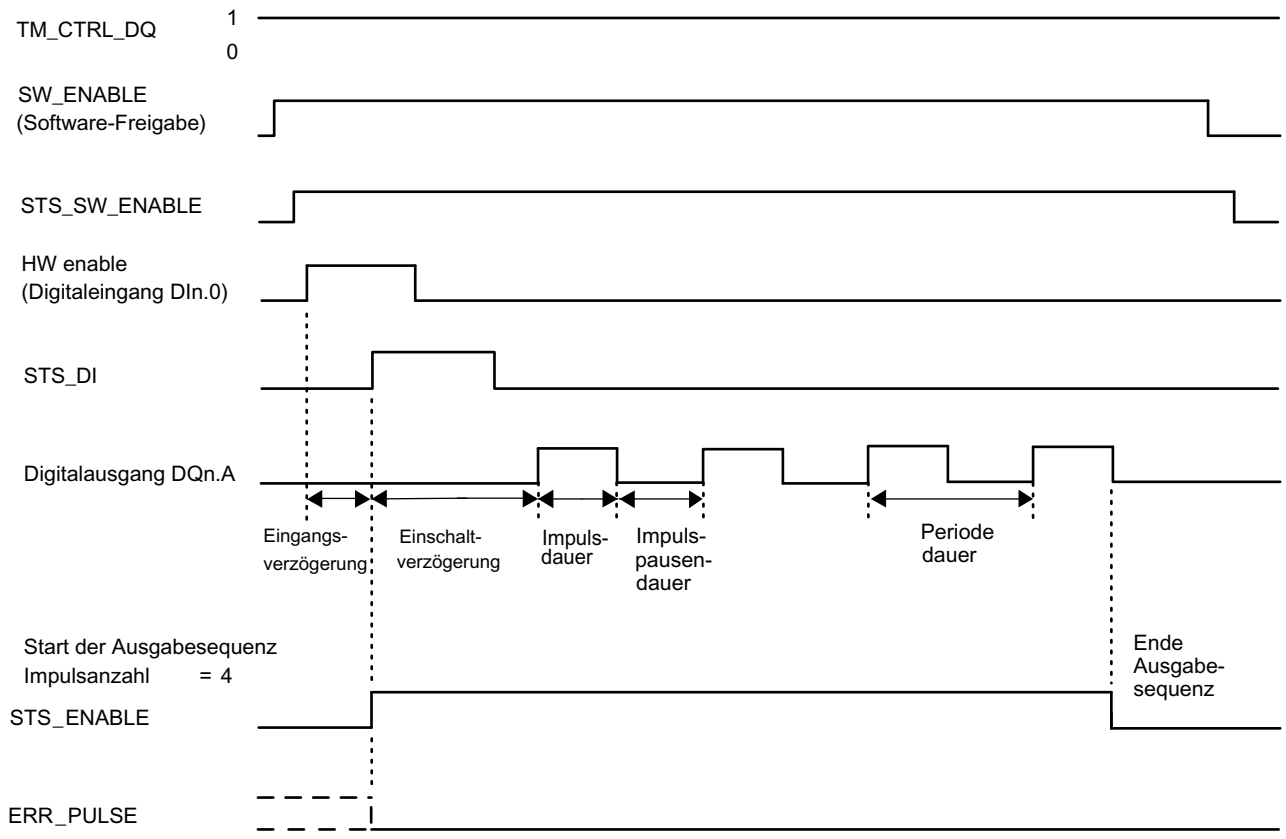

Bild 5-6 Ausgabesequenz der Impulskette

Im obigen Impulsschema ist der Parameter "Funktion DI“ auf "HW-Freigabe" gesetzt. Bei der anderen Option ist "Funktion DI" auf "Eingang" gesetzt. Wenn der Parameter "Funktion DI" auf "Eingang" gesetzt ist, beginnt die Einschaltverzögerung mit der steigenden Flanke von SW_ENABLE.

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe veranlassen (SW_ENABLE 0 → 1.)

Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am TM Pulse 2x24V ansteht.

Daneben können Sie auch den Digitaleingang DIn.0 des TM Pulse 2x24V als HW-Freigabe über den Parameter "Funktion DI" einstellen. Die Eingangsverzögerung der Hardware-Freigabe kann über den Parameter "Eingangsverzögerung" eingestellt werden.

Wenn Sie die Hardware-Freigabe verwenden möchten, muss diese mit der Software-Freigabe kombiniert werden. Bei aktivierter Software-Freigabe mit SW_ENABLE startet die Ausgabesequenz bei der ersten steigenden Flanke der Hardware-Freigabe. Weitere steigende Flanken der Hardware-Freigabe werden während der aktuellen Ausgabesequenz vom TM Pulse 2x24V ignoriert. Sobald die Software-Freigabe aktiviert wurde, startet eine steigende Flanke der Hardware-Freigabe (erkannt nach der Eingangsverzögerung) die nächste Ausgabesequenz.

Wenn die Freigabe aktiviert ist (steigende Flanke) und für die Dauer der Eingangsverzögerung "High" ("Ein") bleibt (Rauschfilter), wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Bei Ablauf der Einschaltverzögerung wird die Impulskette mit der zugewiesenen Impulsdauer ausgegeben. Die Ausgabesequenz wird mit dem Impulsende beendet und STS_ENABLE wird gelöscht.

Wenn Sie die Impulsanzahl während der Ausgabesequenz auf einen Wert ändern, der in der Sequenz bereits erreicht wurde, zeigt das Bit ERR_PULSE einen Impulsausgabefehler an.

Das Rückmeldebit ERR_PULSE wird von TM Pulse 2x24V mit dem nächsten Start der Ausgabesequenz gelöscht.

Hinweis

Steuersignal des Technologiemodulausgangs TM_CTRL_DQ

- Wenn TM_CTRL_DQ = 1, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen am Ausgang DQn.A.
- Wenn TM_CTRL_DQ = 0, übernimmt die CPU die Steuerung und das Programm kann die Ausgänge DQn.A/DQn.B direkt über die Steuerbits SET_DQA/SET_DQB einstellen.

Abbrechen der Ausgabesequenz

Durch Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) während der Einschaltverzögerung oder Impulskette wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang DQn.A werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Wahrheitstabelle

Software-Freigabe SW_ENABLE	Parameter Funktion DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.1 (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	HW_ENABLE	0 → 1 und bleibt 1 während der Eingangsverzögerung. Nur aktiv bei der ersten steigenden Flanke, weitere steigenden Flanken werden ignoriert, es erfolgt kein Start.	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0 → 1	Eingang	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0	HW_ENABLE oder Eingang	Beliebiger Zustand	0	0	Abbrechen
1	HW_ENABLE oder Eingang	Beliebiger Zustand	0, wenn die Einschaltverzögerung nicht abgelaufen ist oder der Zeitpunkt zwischen den Impulsdauern liegt oder die Ausgabesequenz abgeschlossen ist 1, wenn die Einschaltverzögerung abgelaufen und die Impulsdauer noch anhält		-
0 → 1	HW_ENABLE	0	0	0	-

Einstellen und Ändern der Impulsanzahl

Das Steuerungsprogramm kann die Impulsanzahl direkt über den Parameter OUTPUT_VALUE der Steuerschnittstelle einstellen.

Stellen Sie die Impulsanzahl direkt als einen DWord-Zahlenwert zwischen 1 und 4.294.967.295 (2^{32}-1) ein.

Wenn Sie die Impulsanzahl ändern, ist der neue Wert sofort wirksam, unabhängig davon, ob die Einschaltverzögerung möglicherweise abgelaufen ist.

- Falls die neue Impulsanzahl in der aktuellen Ausgabesequenz noch nicht erreicht wurde, wird die neue Impulsanzahl in der aktuellen Ausgabesequenz verwendet.
- Wenn Sie die Impulsanzahl auf eine Anzahl verringert haben, die kleiner als die bereits ausgegebene aktuelle Impulsanzahl ist, wird die Ausgabesequenz abgebrochen, STS_ENABLE und der Digitalausgang DQn.A werden gelöscht und ERR_PULSE wird gesetzt. Mit dem nächsten Start einer Ausgabesequenz wird ERR_PULSE gelöscht.
- Wenn die Impulsanzahl null ist, wird die Ausgabesequenz abgebrochen, STS_ENABLE und der Digitalausgang DQn.A werden gelöscht und ERR_PULSE wird gesetzt. Mit dem nächsten Start einer Ausgabesequenz wird ERR_PULSE gelöscht.

Einstellen und Ändern der Periodendauer

- Permanente Aktualisierung
  Die Periodendauer kann permanent über die Steuerschnittstelle gesteuert werden. Das MODE_SLOT-Bit muss gesetzt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 1 (für Periodendauer) haben. Stellen Sie die Periodendauer als DWord-Zahlenwert in Mikrosekunden im Parameterfeld SLOT ein.
  - Schnelle Impulsausgänge **aktiviert** (siehe Parameter): zwischen 10 µs und 85.000.000 µs.
  - Schnelle Impulsausgänge **deaktiviert** (siehe Parameter): zwischen 100 µs und 85.000.000 µs.
- Einzelne Aktualisierung
  Stellen Sie die Periodendauer in den Konfigurationsparametern ein. Alternativ können Sie eine einzelne Aktualisierung über die Steuerschnittstelle vornehmen. MODE_SLOT muss 0 sein (einzelne Aktualisierung); LD_SLOT muss den Wert 1 (für Periodendauer) haben. Stellen Sie die Periodendauer als DWord-Zahlenwert in Mikrosekunden im Parameterfeld SLOT ein.
  - Schnelle Impulsausgänge **aktiviert** (siehe Parameter): zwischen 10 µs und 85.000.000 µs.
  - Schnelle Impulsausgänge **deaktiviert** (siehe Parameter): zwischen 100 µs und 85.000.000 µs.

Einstellen und Ändern der Einschaltverzögerung

- Permanente Aktualisierung
  Die Einschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss gesetzt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 2 haben (für Einschaltverzögerung). Stellen Sie die Einschaltverzögerung im Parameterfeld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltverzögerung in den Konfigurationsparametern auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden ("0" bedeutet einzelne Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung im Feld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein.
- Wenn Sie den Wert der Einschaltverzögerung während der Ausgabesequenz ändern, wird die neue Einschaltverzögerung mit der nächsten Ausgabesequenz aktiviert.

Weitere Informationen zur Verwendung des SLOT-Parameters finden Sie unter Handhabung des SLOT-Parameters.

Einstellen und Ändern der Einschaltdauer

Der Bereich für den Parameter Einschaltdauer wird über den Parameter "Ausgabeformat" ausgewählt. Das TM Pulse 2x24V verwendet diesen zugewiesenen Einschaltdauerwert zur Berechnung der Impulsdauer. Wenn der von Ihnen zugewiesene Zahlenwert die Obergrenze überschreitet, wird eine Einschaltdauer von 100 % der Periodendauer verwendet. Diese Aktion verursacht keinen Fehler.

- Ausgabeformat "Pro 100 (%)": Wertebereich 0 bis 100
  - Impulsdauer = (Einschaltdauer/100) x Periodendauer.
- Ausgabeformat "Pro 1000": Wertebereich 0 bis 1.000
  - Impulsdauer = (Einschaltdauer/1.000) x Periodendauer.
- Ausgabeformat "Pro 10000": Wertebereich 0 bis 10.000
  - Impulsdauer = (Einschaltdauer/10.000) x Periodendauer.
- Ausgabeformat "S7-Analogausgabe": Wertebereich 0 bis 27.648
  - Impulsdauer = (Einschaltdauer/27.648) x Periodendauer.

Aktualisieren der Einschaltdauer

- Permanente Aktualisierung
  Die Einschaltdauer kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss gesetzt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 4 haben (für Einschaltdauer). Legen Sie einen Wert für die Einschaltdauer im Steuerschnittstellenfeld SLOT fest. Die Einheit hängt vom zugewiesenen Ausgabeformat ab.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltdauer in den Konfigurationsparametern ein. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden ("0" bedeutet einzelne Aktualisierung); LD_SLOT muss den Wert 4 (für Einschaltdauer) haben. Legen Sie einen Wert für die Einschaltdauer im Steuerschnittstellenfeld SLOT fest.
- Die Einheit hängt vom zugewiesenen Ausgabeformat ab. Wenn Sie den Wert der Einschaltdauer während der Ausgabesequenz ändern, wird die neue Einschaltdauer mit der nächsten Ausgabesequenz aktiviert.

Weitere Informationen zur Verwendung des Parameters SLOT finden Sie unter Handhabung des SLOT-Parameters.

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter Funktion: Taktsynchroner Betrieb.

Der taktsynchrone Betrieb hat keinen Einfluss auf die Funktionsweise der Betriebsart Impulskette.

Wenn Sie die Ausgabesequenz mit T_o synchronisieren möchten, setzen Sie den Parameter Funktion DI auf „Eingang“. In diesem Fall beginnt die Ausgabesequenz der Impulskette bei T_o.

Strommessung

Die Strommessung ist in der Betriebsart Impulskette verfügbar. Das Steuerungsprogramm kann die Strommessung zu Steuer- und Diagnosezwecken verwenden.

Weitere Informationen finden Sie im Abschnitt Funktion: Strommessung.

Parameter für die Betriebsart Impulskette

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	2 = Einstellen der Betriebsart "Impulskette".	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = Gleichstrommotor	1
Schnelle Impulsausgabe [1]	Der Ausgang unterstützt höhere Frequenzen (siehe Impulsdauerbereich).	0 = Deaktiviert 1 = Aktiviert	Deaktiviert

Parameter	Bedeutung	Wertebereich	Voreinstellung
Funktion DI	Sie können den Digitaleingang DIn.0 als Eingang oder als Hardware-Freigabe verwenden. Das Signal am DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	• Eingang • HW-Freigabe	Eingang
Ausgabeformat	Definiert das Format für den Verhältniswert der Einschaltdauer.	0 = S7-Analogformat 1 = Pro 100 (%) 2 = Pro 1.000 3 = Pro 10.000	Pro 100 (%)
Eingangsverzögerung	Der Digitaleingang DIn.0 muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Periode	Periodendauer der Ausgabesequenz: Sie können die Periodendauer im Steuerungsprogramm über das Feld SLOT der Steuerschnittstelle ändern.	• Schnelle Impulsausgabe deaktiviert 100 µs bis 85.000.000 µs • Schnelle Impulsausgabe aktiviert 10 µs bis 85.000.000 µs	2.000.000 µs
Einschaltverzögerung	Zeit, die ab dem Start der Ausgabesequenz bis zur Ausgabe der Impulskette verstreicht. Sie können die Einschaltverzögerung im Steuerungsprogramm mithilfe des SLOT-Parameters ändern.	0 µs bis 85.000.000 µs	0 µs
Einschaltdauer	Die Impulsdauer ergibt sich aus Einschaltdauer und Periodendauer. Das Format der Einschaltdauer wird vom Parameter "Ausgabeformat" bestimmt. Sie können die Einschaltdauer im Steuerungsprogramm über das Feld SLOT der Steuerschnittstelle ändern.	Ausgabeformat: Minimalwert bis Maximalwert • Pro 100: 0 bis 100 • Pro 1000: 0 bis 1.000 • Pro 10000: 0 bis 10.000 • S7-Analogausgabe: 0 bis 27.648	50 %

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Steuer- und Rückmeldesignale für die Betriebsart Impulskette

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung				
Kanal 0	Kanal 1 [1]						
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (DWord)	Weisen Sie die Impulsanzahl dem OUTPUT_VALUE als DWord-Zahlenwert zwischen 0 und 4.294.967.295 (2^{32}-1) zu.				
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Einschaltverzögerung, Periodendauer und Einschaltdauer können vor dem Start der Ausgabesequenz geändert werden. Siehe Handhabung des SLOT-Parameters.				
			0 µs bis 85.000.000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation des SLOT-Wertes: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	Bit 2	Bit 1	Bit 0	
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	0	1	Periodendauer in Mikrosekunden
			0	0	1	0	Einschaltverzögerung in Mikrosekunden
			0	1	0	0	Einschaltdauer in dem vom Parameter "Ausgabeformat" definierten Format
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	Bit 4	Betriebsart zur Verwendung des Feldes SLOT.			
			0	Betriebsart für einzelne Aktualisierung			
			1	Betriebsart für permanente Aktualisierung			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	Bit 0	Software-Freigabe: Starten/Aktivieren und Beenden/Deaktivieren der Ausgabesequenz, wenn "Funktion DI" = "Eingang".			
			0	Ausgabe deaktiviert/beendet			
			0 → 1	Startet die Ausgabesequenz bei steigender Flanke, wenn "Funktion DI" = "Eingang".			
			1	Aktivieren der Ausgabesequenz, wenn Start abhängig ist von HW-Freigabe mit "Funktion DI" = „HW-Freigabe".			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	Bit 1	Einstellen der DQn.A-Ausgabequelle: Auswählen des CPU-Programms oder der Ausgabesequenz des Moduls.			
			0	DQn.A und DQn.B werden von der CPU (im Programm) mithilfe der Steuerbits SET_DQA und SET_DQB gesteuert			
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert. DQn.B ist immer 0			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	Bit 3	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ gelöscht wird.			
			0	0 an DQn.A			
			1	1 an DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	Bit 4	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ und SET_DQA gelöscht werden.			
			0	0 an DQn.B			
			1	1 an DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	Bit 0	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB und ERR_24V).			
			0	Rücksetzen von Fehlern ist nicht aktiv			
			1	Rücksetzen von Fehlern ist aktiv			

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf
			1	PWR wurde erkannt, weist jedoch Unterspannung auf
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am 24 V DC-Ausgang an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der Betriebsart für einzelne Aktualisierung SLOT).
			0	Kein Lastfehler anstehend
			1	Lastfehler anstehend: Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Zeigt einen Impulsausgabefehler an.
			0	Kein Impulsausgabefehler
			1	Impulsausgabefehler
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung beider DQs mithilfe von SET_DQA, SET_DQB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B oder die versuchte Einstellung von DQn.A und DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass ein ungültiger Wert in SLOT erkannt wurde (nur in der SLOT-Betriebsart für permanente Aktualisierung).
			0	SLOT-Wert ist gültig
			1	SLOT-Wert ist ungültig
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE gesetzt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine aktive Ausgabesequenz an.
			0	Ausgabesequenz läuft nicht
			1	Ausgabesequenz läuft

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 am Digitalausgang DQn.A
			1	1 am Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 am Digitalausgang DQn.B
			1	1 am Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	Signal 0 am Digitaleingang DIn.0
			1	Signal 1 am Digitaleingang DIn.0
Byte 3: Bit 0 bis 3	Byte 11: Bit 0 bis 3	SEQ_CNT	Sequenzzähler: wird nach Abschluss einer Ausgabesequenz erhöht.	
			Bei SW_ENABLE: 0 bis 1	
			Bei HW_ENABLE: 0 bis 15	
Word 2	Word 6	MEASURED_CURRENT	Positiver Wert S7-Analogformat (0 bis 32.767): • 2-Ampere-Ausgang: Messbereichsendwert 27.648; d. h. 2.000 mA • 4-Ampere-Ausgang: Messbereichsendwert 27.648; d. h. 4.000 mA • 32.767 bedeutet, dass keine gültige Strommessung verfügbar ist; z. B. während der ersten Periode	
Word 3	Word 7	Reserviert	Als 0 lesen	

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für Betriebsart Impulskette

Eingangs- und Ausgangssignal	Bedeutung	Wertebereich	Kanal 0 BaseUnit Pinnummer	Kanal 1 BaseUnit Pinnummer
Eingangssignal				
HW-Freigabe	Sie können die HW-Freigabe mit dem Parameter "Funktion DI" auswählen und die Eingangsverzögerung mit dem Parameter "Eingangsverzögerung" auswählen. Das Signal des Digitaleingangs DIn. 0 wird vom TM Pulse 2x24V nach der Filterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe erteilt 0 → 1 = Start der Ausgabesequenz; abhängig von der Software-Freigabe (SW_ENABLE)	3	4
Ausgangssignal				
Impulskette am Digitalausgang DQn.A	Impulse werden am Digitalausgang DQn.A für die eingestellte Impulsdauer ausgegeben.	0 = kein Impuls 1 = Impuls	9	10

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.5 Betriebsart Ein-/Ausschaltverzögerung

Definition

Das am Digitaleingang DIn.0 des TM Pulse 2x24V anstehende Signal wird mit einer zugewiesenen Ein-/Ausschaltverzögerung am Digitalausgang DQn.A ausgegeben.

Impulsschema

SW_ENABLE wird gesetzt, während Digitaleingang DIn.0 = 0:

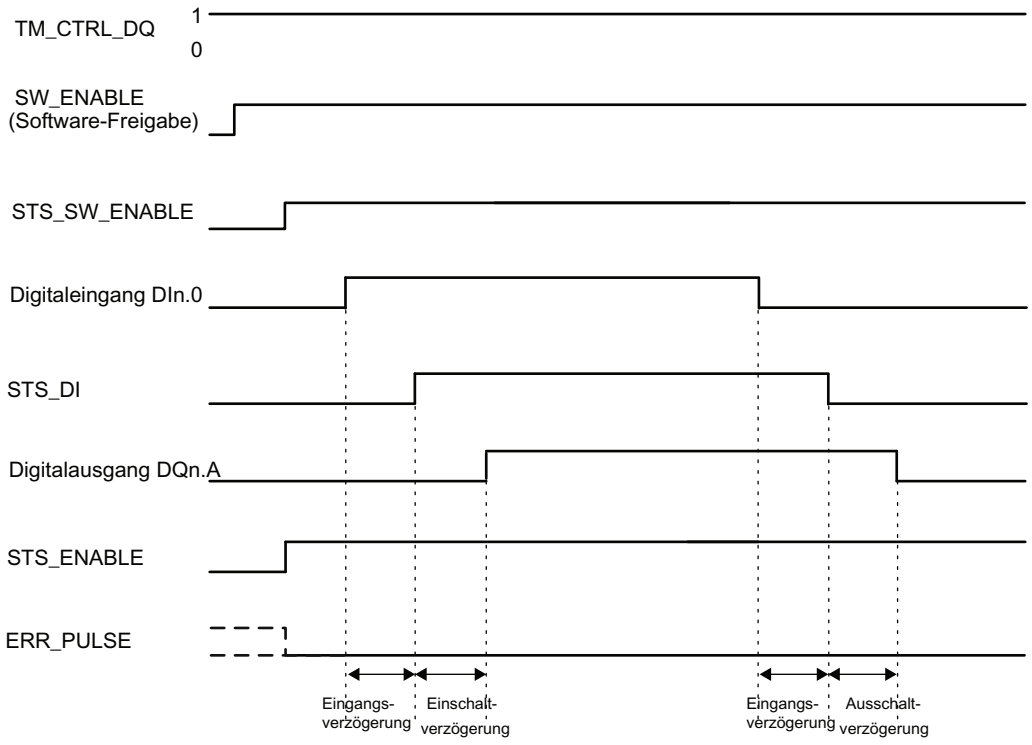

Bild 5-7 Ein-/Ausschaltverzögerung Ausgabesequenz (am Start Digitaleingang DIn.0 = 0)

SW_ENABLE wird gesetzt, während Digitaleingang DIn.0 = 1:

Wenn SW_ENABLE gesetzt wird, während Digitaleingang DIn.0 = 1, wird die erste Flanke eines Digitaleingangs (fallende Flanke) ignoriert.

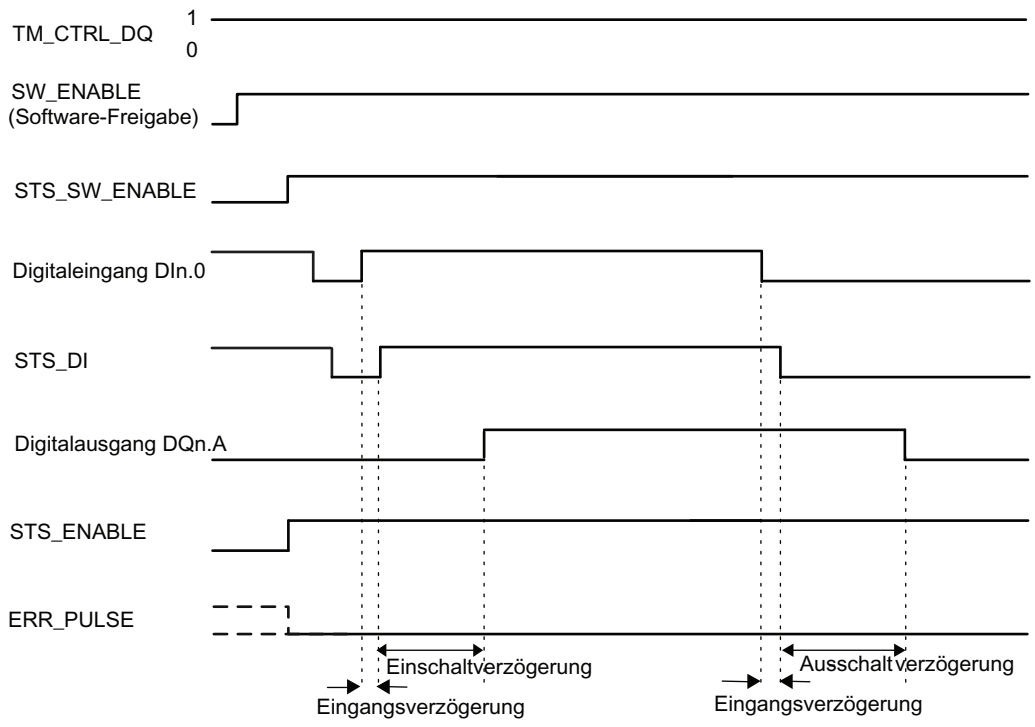

Bild 5-8 Ein-/Ausschaltverzögerung Ausgabesequenz (am Start Digitaleingang DIn.0 = 1)

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe veranlassen (SW_ENABLE 0 → 1.)

Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am TM Pulse 2x24V ansteht.

- Wenn der Digitaleingang DIn.0 in den "High"-Zustand (steigende Flanke) geht und für die Dauer der Eingangsverzögerung auf "High" bleibt, wird die Ausschaltverzögerung gestartet. Nachdem die Einschaltverzögerung abgelaufen ist, wird der Digitalausgang DQn.A gesetzt.
- Wenn der Digitaleingang DIn.0 für die Dauer der Eingangsverzögerung in den "Low"-Zustand ("Aus") geht, wird die Ausschaltverzögerung gestartet. Nachdem die Einschaltverzögerung abgelaufen ist, wird der Digitalausgang DQn.A gelöscht.
- Wenn das TM Pulse 2x24V am Digitaleingang DIn.0 eine Impulsdauer oder Impulspause erkennt, die kürzer als die Eingabeverzögerungszeit ist, wird die Eingabe ignoriert und der Digitalausgang DQn.A bleibt unverändert.
- Wenn das TM Pulse 2x24V am Digitaleingang DIn.0 eine Impulsdauer oder Impulspause erkennt, die länger als die Eingabeverzögerungszeit ist, aber zu kurz, um die parametrierte Impulsdauer oder Impulspause zu erzeugen, wird das ERR_PULSE -Bit gesetzt und der Digitalausgang DQn.A bleibt unverändert.
- Mit der nächsten Flanke des Digitaleingangs DIn.0 löscht das TM Pulse 2x24V das Rückmeldebit ERR_PULSE.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ des Technologiemoduls

- Wenn TM_CTRL_DQ = 1, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen am Ausgang DQn.A.
- Wenn TM_CTRL_DQ = 0, übernimmt die CPU die Steuerung und das Programm kann die Ausgänge DQn.A/DQn.B über die Steuerbits SET_DQA/SET_DQB direkt einstellen.

Abbrechen der Ausgabesequenz

Durch Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) während der Einschaltverzögerung oder Impulskette wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang DQn.A werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Wahrheitstabelle

Software-Freigabe SW_ENABLE	Digitaleingang DIn.0	Digitalausgang DQn.A (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	0 → 1 und bleibt 1 während der Eingangsverzögerung.	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	1	Starten
1	1 → 0	1, wenn Einschaltverzögerung > 0 0, wenn Einschaltverzögerung = 0	1	Starten
0	Beliebiger Zustand	0	0	Abbrechen
1	Beliebiger Zustand	0, wenn die Einschaltverzögerung nicht abgelaufen ist oder die Ausschaltverzögerung abgelaufen ist 1, wenn die Einschaltverzögerung abgelaufen ist und die Ausschaltverzögerung nicht abgelaufen ist	1	-
0 → 1	0	0	1	-

Mindestimpulsdauer/MIndestimpulspause des Digitalausgangs DQn.A

Die Mindestimpulsdauer/MIndestimpulspause des Digitalausgangs DQn.A beträgt 1,5 µs (bei aktiver schneller Impulsausgabe) und 10 µs (bei inaktiver schneller Impulsausgabe). Bitte beachten Sie, dass niedrigere Werte möglich, aufgrund der Hardwareabhängigkeit jedoch nicht garantiert sind.

Bitte berücksichtigen Sie diesen Hinweis beim Einstellen der Ein-/Ausschaltverzögerung und der Impulsdauer/Impulspause für den Digitaleingang DIn.0. Andernfalls kann die Antwort am Digitalausgang DQn.A nicht garantiert werden.

Die Impulsdauer des Digitaleingangs DIn.0 ist zu kurz

Fall 1: Die DIn.0-Impulsdauer < Einschaltverzögerung: Der DIn.0-Impuls wird gefiltert und ignoriert (es tritt kein Fehler auf).

Fall 2: Das TM Pulse 2x24V erkennt eine zu kurze DIn.0-Impulsdauer, wenn: DIn.0-Impulsdauer + Ausschaltverzögerung ≤ Einschaltverzögerung (Fehler wird gesetzt)

Fall 3: Das TM Pulse 2x24V erkennt eine zu kurze DIn.0-Impulsdauer, wenn: DIn.0-Impulsdauer + Ausschaltverzögerung + minimale DQ-Impulsdauer < Einschaltverzögerung (Fehler wird gesetzt)

Antwort des TM Pulse 2x24V auf eine zu kurze Impulsdauer:

- ERR_PULSE wird gesetzt.
- Die aktuelle Einschaltverzögerung wird gelöscht.
- Die Ausschaltverzögerung wird nicht gestartet.
- Der Signalpegel am Digitalausgang DQn.A bleibt 0.

ERR_PULSE wird mit der nächsten steigenden Flanke am Digitaleingang DIn.0 gelöscht.

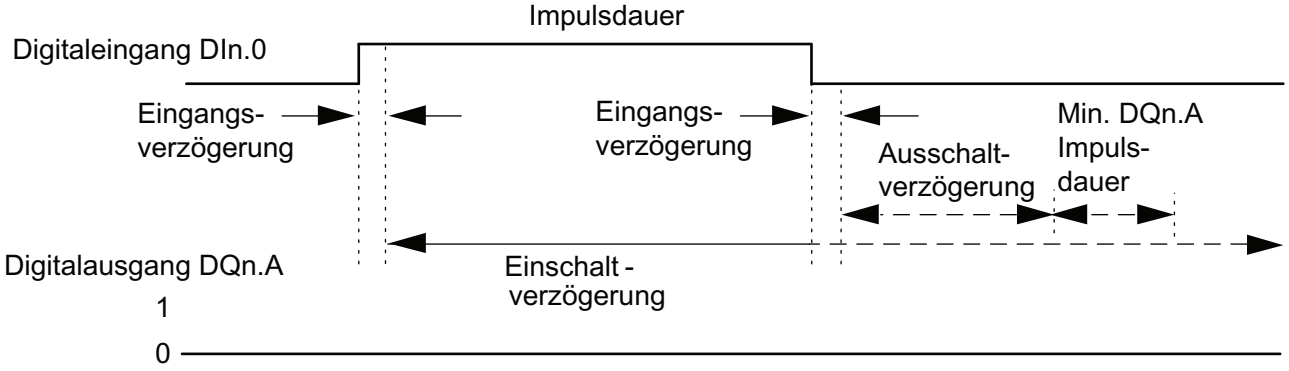

Bild 5-9 Die DIn.0-Impulsdauer ist zu kurz

Die Impulspause des Digitaleingangs DIn.0 ist zu kurz

Das TM Pulse 2x24V erkennt eine zu kurze Impulspause an der steigenden Flanke am Digitaleingang DIn.0, wenn:

Impulspause + Einschaltverzögerung ≤ Ausschaltverzögerung.

- Fall 1: Impulspause < Einschaltverzögerung: Die Impulspause am Eingang DIn.0 wird ausgefiltert und ignoriert (es tritt kein Fehler auf).
- Fall 2: Impulspause + Einschaltverzögerung ≤ Ausschaltverzögerung (Fehler wird gesetzt).
- Fall 3: Impulspause + Einschaltverzögerung < Ausschaltverzögerung + MIndestimpulspause (Fehler wird gesetzt).

Antwort des TM Pulse 2x24V auf eine zu kurze Impulspause:

- ERR_PULSE wird gesetzt.
- Die aktuelle Ausschaltverzögerung wird gelöscht.
- Die Einschaltverzögerung wird nicht gestartet.
- Der Signalpegel am Digitalausgang DQn.A bleibt 1.

ERR_PULSE wird mit der nächsten fallenden Flanke am Digitaleingang DIn.0 gelöscht.

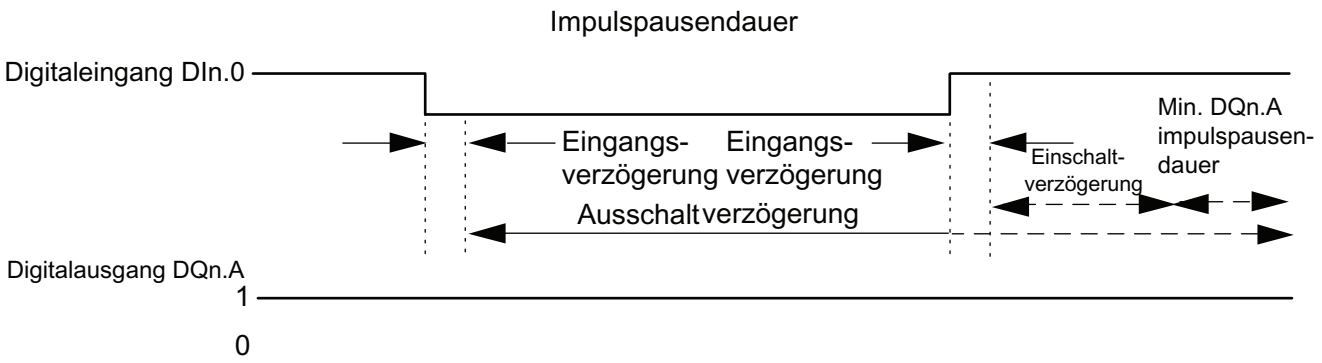

Bild 5-10 Die DIn.0-Impulspause ist zu kurz

Die aktuelle Einschaltverzögerung wird erneut ausgelöst

Das TM Pulse 2x24V startet eine neue Einschaltverzögerung an der steigenden Flanke am Digitaleingang DIn.0, wenn:

Einschaltverzögerung > Impulsdauer + Impulspause

Hierdurch wird die aktuelle Ausschaltverzögerung gelöscht.

Der Digitalausgang DQn.A wird nur gesetzt, wenn der Signalpegel 1 am Digitaleingang DIn.0 länger ansteht als die Einschaltverzögerung. Dadurch können Sie schnelle Impulse filtern.

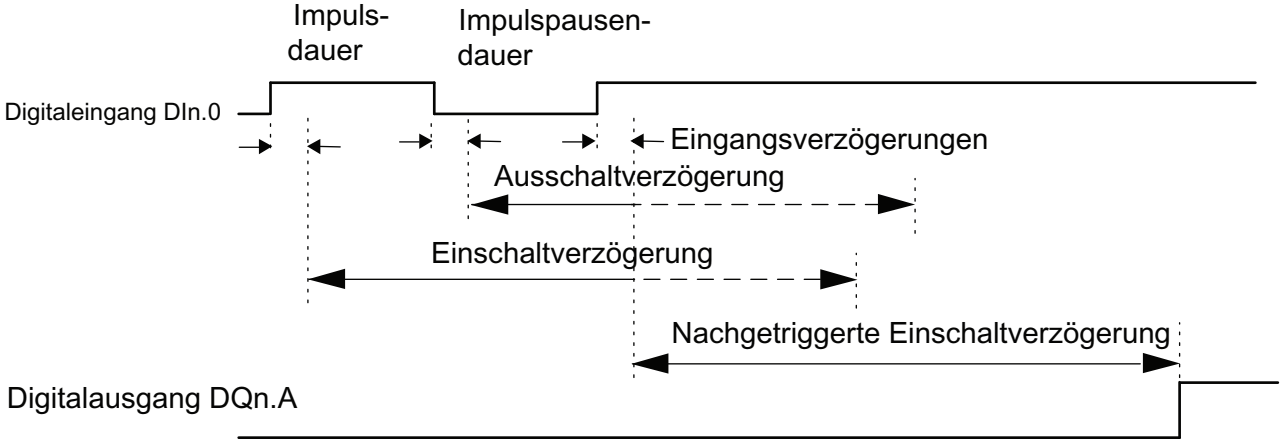

Bild 5-11 DIn.0 löst die aktuelle Einschaltverzögerung erneut aus

Neuauslösen der aktuellen Ausschaltverzögerung

Das TM Pulse 2x24V startet eine neue Ausschaltverzögerung an der fallenden Flanke am Digitaleingang DIn.0, wenn:

Ausschaltverzögerung > Impulsdauer + Impulspause

Der Digitalausgang DQn.A wird nur gelöscht, wenn der Signalpegel 0 am Digitaleingang Dn. 0I länger ansteht als die Ausschaltverzögerung.

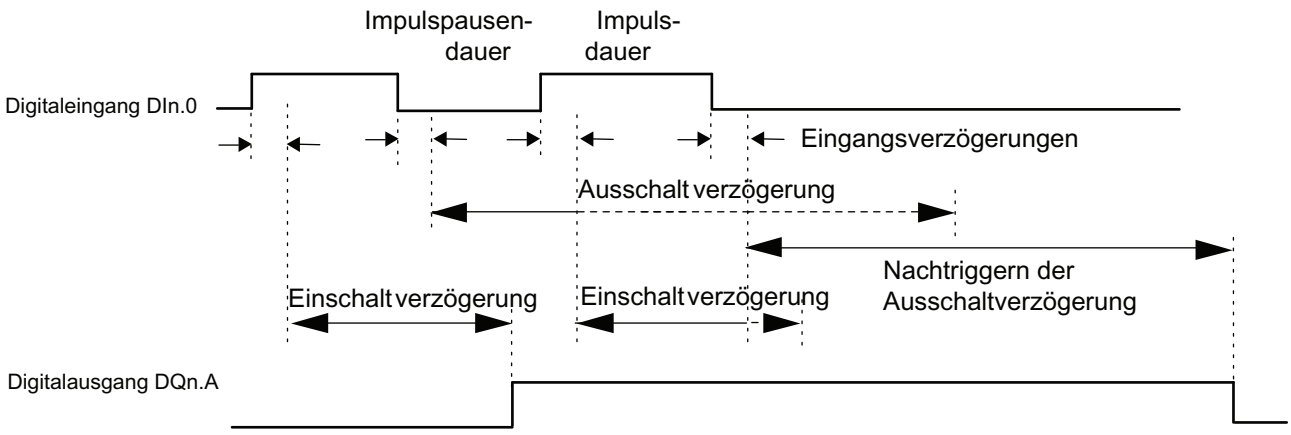

Bild 5-12 Nachtriggern der aktuellen Ausschaltverzögerung

Einstellen und Ändern von OUTPUT_VALUE (Einschaltverzögerung)

- Sie stellen den OUTPUT_VALUE der Einschaltverzögerung direkt ein, indem Sie mit dem Steuerungsprogramm einen Wert zuweisen.
- Die Einheit ist immer die Mikrosekunde. Möglicher Bereich: zwischen 0 µs und 85.000.000 µs. Wird vom Programm ein ungültiger Wert zugewiesen, gibt das Modul in der Rückmeldeschnittstelle den Fehler ERR_OUT_VAL aus und verwendet weiter den letzten gültigen Wert.
- Der neue Einschaltverzögerungswert wird mit der nächsten steigenden Flanke am Digitaleingang DIn.0 aktiviert.

Einstellen und Ändern der Ausschaltverzögerung

- Permanente Aktualisierung
  Die Ausschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss gesetzt werden (permanente Aktualisierung); LD_SLOT muss den Wert 3 (für Ausschaltverzögerung) haben. Stellen Sie im Feld SLOT die Ausschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Die Einheit ist immer die Mikrosekunde.
- Einzelne Aktualisierung
  Stellen Sie die Ausschaltverzögerung in den Konfigurationsparametern auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Alternativ nehmen Sie eine einzelne Aktualisierung über die Steuerschnittstelle vor. MODE_SLOT muss gelöscht werden (einzelne Aktualisierung); LD_SLOT muss den Wert 3 (für Ausschaltverzögerung) haben. Stellen Sie im Feld SLOT die Ausschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs ein.

Wenn Sie den Ausschaltverzögerungswert während der Ausgabesequenz ändern, wird die neue Ausschaltverzögerung mit der nächsten fallenden Flanke am Digitaleingang DIn.0 aktiviert.
Weitere Informationen zur Verwendung des SLOT-Parameters finden Sie unter Handhabung des SLOT-Parameters.

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter Funktion: Taktsynchroner Betrieb.

Der taktsynchrone Betrieb hat keinen Einfluss auf die Funktionsweise der Betriebsart Ein-/Ausschaltverzögerung.

Parameter für die Betriebsart Ein-/Ausschaltverzögerung

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	3 = Einstellen der Betriebsart "Ein-/Ausschaltverzögerung"	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = Gleichstrommotor	1
Schnelle Impulsausgabe [1]	Diese Ausgabe unterstützt höhere Frequenzen.	0 = deaktiviert 1 = aktiviert	Deaktiviert
Funktion DI	Der Digitaleingang DIn.0 kann nur als Eingang verwendet werden.	0 = Eingang 1 = reserviert (nicht verwendbar)	Eingang
Eingangsverzögerung	Der Digitaleingang DI muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Ausschaltverzögerung	Die Zeitspanne von der fallenden Flanke des Digitaleingangs DIn.0 bis zur fallenden Flanke von DQn.A. Die Ausschaltverzögerung kann im Steuerungsprogramm mithilfe des SLOT-Parameters geändert werden.	0 µs bis 85.000.000 µs	0 µs

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Steuer- und Rückmeldesignale für die Betriebsart Ein-/Ausschaltverzögerung

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung
Kanal 0	Kanal 1 [1]		
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (DWord)	Einschaltverzögerung
			0 µs bis 85.000.000 µs
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Die Ausschaltverzögerung kann jederzeit geändert werden, wird jedoch erst mit der nächsten fallenden Flanke des Digitaleingangs DIn.0 wirksam.
			0 µs bis 85.000.000 µs

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung				
Kanal 0	Kanal 1 [1]						
Byte 8	Byte 20	LD_SLOT	Interpretation des SLOT-Wertes: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	Bit 2	Bit 1	Bit 0	
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	1	1	Ausschaltverzögerung in Mikrosekunden
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	Bit 4	Betriebsart zur Verwendung des Feldes SLOT.			
			0	Betriebsart für einzelne Aktualisierung			
			1	Betriebsart für permanente Aktualisierung			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	Bit 0	Software-Freigabe: Starten und Beenden der Ausgabesequenz.			
			0	Ausgabe abgebrochen			
			0 → 1	Startet Ausgabesequenz bei steigender Flanke			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	Bit 1	Einstellen der DQn.A-Ausgabequelle: Bewirkt die Auswahl des CPU-Programms oder der Ausgabesequenz des Moduls.			
			0	DQn.A und DQn.B werden von der CPU (im Programm) mithilfe der Steuerbits SET_DQA und SET_DQB gesteuert.			
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert. DQn.B ist immer 0 bei TM_CTRL = 1.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	Bit 3	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ gelöscht wird.			
			0	0 an DQn.A			
			1	1 an DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	Bit 4	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ und SET_DQA gelöscht werden.			
			0	0 an DQn.B			
			1	1 an DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	Bit 0	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB und ERR_24V).			
			0	Rücksetzen von Fehlern ist nicht aktiv			
			1	Rücksetzen von Fehlern ist aktiv			

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	Bit 0	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf.
			1	PWR wurde erkannt, weist jedoch Unterspannung auf.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am 24 V DC-Ausgang an. Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der SLOT-Betriebsart für einzelne Aktualisierung).
			0	Kein Lastfehler anstehend.
			1	Lastfehler anstehend: Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Zeigt einen Impulsausgabefehler an.
			0	Kein Impulsausgabefehler
			1	Impulsausgabefehler
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung von DQn.A und DQn.B mithilfe von SET_DQA, SET_DB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B oder die versuchte Einstellung von DQn.A und DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Zeigt an, dass in OUTPUT_VALUE ein ungültiger Wert erkannt wurde.
			0	OUTPUT_VALUE ist gültig
			1	OUTPUT_VALUE ist ungültig
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass in SLOT ein ungültiger Wert erkannt wurde (nur in der SLOT-Betriebsart für permanente Aktualisierung).
			0	SLOT-Wert ist gültig
			0 → 1	SLOT-Wert ist ungültig
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE gesetzt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine laufende Ausgabesequenz an.
			0	Die Impulsausgabe ist nicht aktiv
			1	Die Impulsausgabe ist aktiv

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 am Digitalausgang DQn.A
			1	1 am Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 am Digitalausgang DQn.B
			1	1 am Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	0 am Digitaleingang DIn.0
			1	1 am Digitaleingang DIn.0
Byte 3: Bit 0 bis 3	Byte 11: Bit 0 bis 3	SEQ_CNT	Zählen von steigenden und fallenden Flanken am Ausgang DQn.A	
Word 3	Word 7	Reserviert	Als 0 lesen	

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für Betriebsart Ein-/Ausschaltverzögerung

Eingangs- und Ausgangssignal	Bedeutung	Wertebereich	Kanal 0 BaseUnit Pinnummer	Kanal 1 BaseUnit Pinnummer
Eingangssignal				
Digitaleingang DIn.0	Das Signal am Digitaleingang DIn.0 wird mit einer Ein-/Ausschaltverzögerung am Digitalausgang DQn.A ausgegeben.	0 = kein Impuls 1 = Impuls	3	4
Ausgangssignal				
Impuls am Digitalausgang DQn.A	Das Signal am Digitaleingang DIn.0 wird mit einer Ein-/Ausschaltverzögerung am Digitalausgang DQn.A ausgegeben.	0 = kein Signal 1 = Signal	9	10

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.6 Betriebsart Frequenzausgabe

Definition

In dieser Betriebsart können Sie einen Frequenzwert mit hohen Frequenzen präziser zuweisen als über die PWM-Perioden- und Periodendauer.

Ein Rechtecksignal mit einer zugewiesenen Frequenz und einer konstanten Einschaltdauer von 50 % wird am Digitalausgang des TM Pulse 2x24V erzeugt.

Die Ausgabesequenz wird nach Ablauf der konfigurierten Einschaltverzögerung am Digitalausgang DQn.A gestartet.

Impulsschema

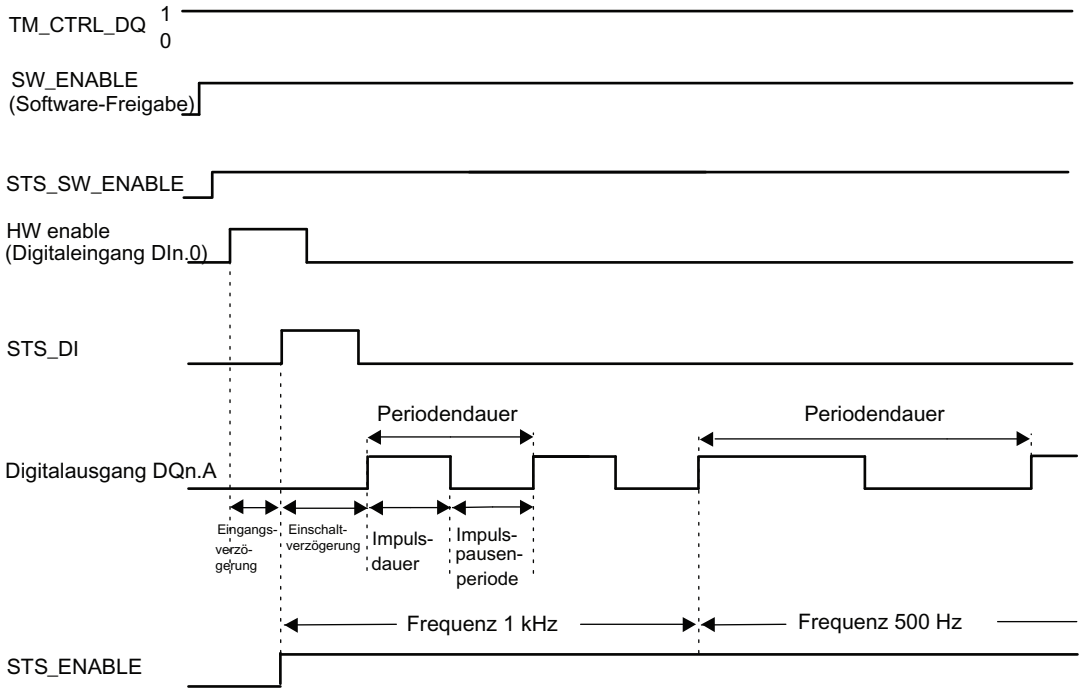

Bild 5-13 Frequenzausgabe - Ausgabesequenz

Im obigen Impulsschema ist der Parameter "Funktion DI“ auf "HW-Freigabe" gesetzt. Bei der anderen Option ist "Funktion DI" auf "Eingang" gesetzt. Wenn der Parameter "Funktion DI" auf "Eingang" gesetzt ist, beginnt die Einschaltverzögerung mit der steigenden Flanke von SW_ENABLE.

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe veranlassen (SW_ENABLE 0 → 1.)

Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am TM Pulse 2x24V ansteht.

Daneben können Sie auch den Digitaleingang DIn.0 des TM Pulse 2x24V als HW-Freigabe über den Parameter "Funktion DI" einstellen.

Wenn Sie die Hardware-Freigabe verwenden möchten, muss diese mit der Software-Freigabe kombiniert werden. Bei aktivierter Software-Freigabe mit SW_ENABLE startet die Ausgabesequenz bei der ersten steigenden Flanke der Hardware-Freigabe. Weitere steigende Flanken der Hardware-Freigabe werden während der aktuellen Ausgabesequenz vom TM Pulse 2x24V ignoriert.

Wenn die Freigabe aktiviert ist (steigende Flanke) und für die Dauer der Eingangsverzögerung "High" ("Ein") bleibt (Rauschfilter), wird die Einschaltverzögerung gestartet und STS_ENABLE

gesetzt. Die Frequenzsequenz wird mit Ablauf der Einschaltverzögerung ausgegeben. Die Ausgabesequenz läuft kontinuierlich, solange SW_ENABLE gesetzt ist.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ des Technologiemoduls

- Wenn TM_CTRL_DQ = 1, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen am Ausgang DQn.A.
- Wenn TM_CTRL_DQ = 0 ist, übernimmt die CPU die Steuerung und das Programm kann die Ausgänge DQn.A/DQn.B über die Steuerbits SET_DQA/SET_DQB direkt einstellen.

Abbrechen der Ausgabesequenz

Durch Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) während der Einschaltverzögerung oder der Frequenzausgabe wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang DQn.A werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Wahrheitstabelle

Software-Freigabe SW_ENABLE	Parameter Funktion DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.A (wenn TM_CTRL_DQ = 1)	STS_ENABLE	Ausgabesequenz
1	HW_ENABLE	0 → 1 und bleibt 1 während der Eingangsverzögerung. Nur aktiv bei der ersten steigenden Flanke, weitere steigenden Flanken werden ignoriert, es erfolgt kein Start.	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0 → 1	Eingang	nicht verwendet	0, wenn Einschaltverzögerung > 0 1, wenn Einschaltverzögerung = 0	0 → 1	Starten
0	HW_ENABLE oder Eingang	Beliebiger Zustand	0	0	Abbrechen
1	HW_ENABLE oder Eingang	Beliebiger Zustand	0, wenn die Einschaltverzögerung nicht abgelaufen ist oder der Zeitpunkt zwischen den Impulsdauern liegt 1, wenn die Einschaltverzögerung abgelaufen und die Impulsdauer noch anhält		-
0 → 1	HW_ENABLE	0	0	0	-

Einstellen und Ändern des Ausgabewertes (Frequenz)

Sie stellen den OUTPUT_VALUE direkt mit dem Steuerungsprogramm in der Steuerschnittstelle ein. Der Wert wird im Real-Format angegeben und die Einheit ist immer "Hz". Der mögliche Bereich hängt vom Parameter "Schnelle Impulsausgabe" wie folgt ab:

- Schnelle Impulsausgabe deaktiviert
  - Frequenz (OUTPUT_VALUE): 0,02 Hz bis 10.000 Hz
- Schnelle Impulsausgabe aktiviert
  - Frequenz (OUTPUT_VALUE): 0,02 Hz bis 100.000 Hz

Der neue Frequenzwert wird mit der nächsten steigenden Flanke der Ausgabe ausgegeben.

Genauigkeit der Ausgabefrequenz

Die konfigurierte Ausgabefrequenz wird mit einer Genauigkeit von +/- 100 ppm am Digitalausgang DQn.A ausgegeben.

Einstellen und Ändern der Einschaltverzögerung

- Permanente Aktualisierung
  Die Einschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss eingestellt werden (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung im Feld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein. Die Einheit ist immer die Mikrosekunde.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs in den Konfigurationsparametern ein, die Einheit ist immer Mikrosekunden. Alternativ nehmen Sie eine einzelne Aktualisierung über die Steuerschnittstelle vor. MODE_SLOT muss gelöscht werden (einzelne Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung im Feld SLOT auf einen Wert zwischen 0 µs und 85.000.000 µs ein.

Wenn Sie den Wert der Einschaltverzögerung während der Ausgabesequenz ändern, wird die neue Einschaltverzögerung mit der nächsten Ausgabesequenz aktiviert.

Weitere Informationen zur Verwendung des Parameters SLOT finden Sie unter Handhabung des SLOT-Parameters.

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter "Funktion: Taktsynchroner Betrieb".

Der taktsynchrone Betrieb hat keinen Einfluss auf die Funktionalität der Betriebsart Frequenzausgabe.

Wenn Sie die Ausgabesequenz mit T_o synchronisieren möchten, stellen Sie den Parameter "Funktion DI" auf "Eingang" und die Frequenzausgabesequenz startet bei T_o.

Parameter der Betriebsart Frequenzausgabe

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	4 = Einstellen der Betriebsart Frequenzausgabe.	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = Gleichstrommotor	1
Schnelle Impulsausgabe [1]	Diese Ausgabe unterstützt höhere Frequenzen.	0 = deaktiviert 1 = aktiviert	Deaktiviert
Funktion DI	Sie können den Digitaleingang DIn.0 als Eingang oder als Hardware-Freigabe verwenden.	0 = Eingang 1 = HW-Freigabe	Eingang
Eingangsverzögerung	Der Digitaleingang DIn.0 muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Einschaltverzögerung	Die Dauer vom Start der Ausgabesequenz bis zur Impulsausgabe. Sie können die Einschaltverzögerung im Steuerungsprogramm mithilfe des SLOT-Parameters ändern.	0 µs bis 85.000.000 µs	0 µs

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist.

Steuer- und Rückmeldesignale für die Betriebsart Frequenzausgabe

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (Real)	Frequenz des Digitalausgangs DQn.A als Real-Zahl. Die Einheit ist immer Hz. Wenn Sie die Unter- oder Obergrenze des Bereichs nicht einhalten, verwendet TM Pulse 2x24V den letzten gültigen Wert und der Fehler ERR_OUT_VAL wird aktiviert.	
			Schnelle Impulsausgabe deaktiviert	Schnelle Impulsausgabe aktiviert
			0,02 Hz bis 10.000 Hz	0,02 Hz bis 100.000 Hz
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Die Einschaltverzögerung kann vor dem Start der Ausgabesequenz geändert werden. Siehe MODE_SLOT.	
			0 µs bis 85.000.000 µs	

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung				
Kanal 0	Kanal 1 [1]						
Byte 8	Byte 20	LD_SLOT	Interpretation des SLOT-Wertes: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	1	0	Einschaltverzögerung in Mikrosekunden
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Betriebsart zur Verwendung des Feldes SLOT.			
			0	Betriebsart für einzelne Aktualisierung			
			1	Betriebsart für permanente Aktualisierung			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software-Freigabe: Starten/Aktivieren und Beenden/Deaktivieren der Ausgabesequenz.			
			0	Ausgabe deaktiviert/beendet			
			0 → 1	Startet die Ausgabesequenz bei steigender Flanke, wenn "Funktion DI" = "Eingang".			
			1	Aktivieren der Ausgabesequenz, wenn Start abhängig ist von HW-Freigabe mit "Funktion DI" = „HW-Freigabe".			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Einstellen der DQn.A-Ausgabequelle: Bewirkt die Auswahl des CPU-Programms oder der Ausgabesequenz des Moduls.			
			0	DQn.A und DQn.B werden von der CPU (im Programm) mithilfe der Steuerbits SET_DQA und SET_DQB gesteuert.			
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert. DQn.B ist immer 0.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ gelöscht wird.			
			0	0 an DQn.A			
			1	1 an DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ und SET_DQA gelöscht werden.			
			0	0 an DQn.B			
			1	1 an DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB und ERR_24V).			
			0	Rücksetzen von Fehlern ist nicht aktiv			
			1	Rücksetzen von Fehlern ist aktiv			

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1[1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf
			1	PWR wurde erkannt, weist jedoch Unterspannung auf

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1[1]			
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am 24 V DC-Ausgang an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der SLOT-Betriebsart für einzelne Aktualisierung).
			0	Kein Lastfehler anstehend
			1	Lastfehler anstehend; um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung von DQn.A und DQn.B mithilfe von SET_DQA, SET_DB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B oder die versuchte Einstellung von DQn.A und DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Zeigt an, dass in OUTPUT_VALUE ein ungültiger Wert erkannt wurde.
			0	OUTPUT_VALUE ist gültig
			1	OUTPUT_VALUE ist ungültig
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass in SLOT ein ungültiger Wert erkannt wurde (nur in der SLOT-Betriebsart für permanente Aktualisierung).
			0	SLOT-Wert ist gültig
			0 → 1	SLOT-Wert ist ungültig
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE gesetzt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine aktive Ausgabesequenz an.
			0	Die Impulsausgabe ist nicht aktiv
			1	Die Impulsausgabe ist aktiv
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 am Digitalausgang DQn.A
			1	1 am Digitalausgang DQn.A

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1[1]			
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 an Digitalausgang DQn.B
			1	1 an Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	0 an Digitaleingang DIn.0
			1	1 an Digitaleingang DIn.0
Byte 3: Bit 0 bis 3	Byte 11: Bit 0 bis 3	SEQ_CNT	Sequenzzähler = 0. Der Sequenzzähler wird in der Betriebsart Frequenzausgabe nicht verwendet.	
Word 3	Word 7	Reserviert	Als 0 lesen	

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für Betriebsart Frequenzausgabe

Eingangs- und Ausgangssignal	Bedeutung	Wertebereich	Kanal 0 BaseUnit Pinnummer	Kanal 1 BaseUnit Pinnummer
Eingangssignal				
HW-Freigabe	Sie können die HW-Freigabe mit dem Parameter "Funktion DI" auswählen und die Eingangsverzögerung mit dem Parameter "Eingangsverzögerung" auswählen. Das Signal des Digitaleingangs DIn.0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	0 = HW-Freigabe gelöscht 1 = HW-Freigabe aktiviert 0 → 1 = Start der Ausgabesequenz nach der Eingangsverzögerung, abhängig von der Software-Freigabe (SW_ENABLE)	3	4
Ausgangssignal				
Impuls am Digitalausgang DQn.A	Ein Impuls wird am Digitalausgang DQn.A für die zugewiesene Frequenz ausgegeben.	0 = kein Impuls 1 = Impuls	9	10

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.7 Betriebsart Gleichstrommotor

Definition

Jeder Kanal besitzt einen A- und einen B-Ausgang für den Anschluss an die Gleichstrommotorlast. Über den bipolaren Ausgangsschalter und die Impulsweitenmodulation können die Drehrichtung und die Einschaltdauer der Ausgangsspannung zugewiesen werden.

Sie können eine Zweikanalschaltung für maximal 2 A pro Kanal oder eine Einkanal-Parallelschaltung mit maximal 4 A verwenden.

Zweikanaliger Motorlastanschluss:

- Kanal 0: Motor angeschlossen zwischen DQ0.A und DQ0.B
- Kanal 1: Motor angeschlossen zwischen DQ1.A und DQ1.B

Einkanaliger Motorlastanschluss

- 4-A-Kanal 0: Motor angeschlossen zwischen DQ0.A und DQ0.B.
  DQ0.A und DQ1.A sind miteinander verbunden; DQ0.B und DQ1.B sind miteinander verbunden.

Weitere Informationen zum Anschluss finden Sie im Kapitel Anschließen dieses Handbuchs.

Motordrehung

Drehrichtung

- Vorwärts während der Hoch-Phase des PWM-Signals: DQn.A ist 1 und DQn.B ist 0.
- Rückwärts während der Hoch-Phase des PWM-Signals: DQn.A ist 0 und DQn.B ist 1.
- Vorwärts/rückwärts während der Niedrig-Phase des PWM-Signals: DQn.A ist 0 und DQn.B ist 0.

Motorstopp

Der Motor kann mithilfe eines an den Digitaleingang DIn.0 angeschlossenen externen Signals oder mithilfe des Signals SW_ENABLE in der Steuerschnittstelle angehalten werden. Konfigurieren Sie die "Funktion DI" entsprechend.

Einstellung der Funktion DI:

- Die "HW-Freigabe" startet den Motor an der steigenden Flanke des DI und stoppt den Motor an der fallenden Flanke des DI.
- "Externer Stopp" startet den Motor an der steigenden Flanke von SW_ENABLE und stoppt den Motor an der steigenden Flanke am DI oder der abfallenden Flanke von SW_ENABLE.
- „Eingang" startet den Motor an der steigenden Flanke von SW_ENABLE und stoppt den Motor an der fallenden Flanke von SW_ENABLE (DI hat keinen Einfluss auf die Motorsteuerung).

Sie steuern die Einschaltdauer des Ausgangsimpulses mit dem Feld OUTPUT_VALUE der Steuerschnittstelle. Das TM Pulse 2x24V erzeugt anhand dieses Wertes kontinuierliche Impulse. Der OUTPUT_VALUE bestimmt die Einschaltdauer (Impulsdauer/Periodendauer) innerhalb einer Periode für die Impulsweitenmodulation. Die Periodendauer ist anpassbar.

Nach Ablauf der zugewiesenen Einschaltverzögerung beginnen die DQn.A- und DQn.B-Ausgangsimpulse (Ausgabesequenz).

Schnelle Impulsausgabe, Strommessung und Stromregelung sind in der Betriebsart Gleichstrommotor nicht verfügbar.

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mithilfe der Software-Freigabe ausgeben (SW_ENABLE 0 → 1.)

Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am TM Pulse 2x24V ansteht.

Außerdem können Sie den Digitaleingang DIn.0 des TM Pulse 2x24V als HW-Freigabe über den Parameter "Funktion DI" zuweisen. Die Eingangsverzögerung (Störfilter) der Hardware-Freigabe kann über den Parameter "Eingangsverzögerung" eingestellt werden.

Wenn Sie die Hardware-Freigabe verwenden möchten, muss diese mit der Software-Freigabe kombiniert werden. Bei aktivierter Software-Freigabe mit SW_ENABLE startet die Ausgabesequenz bei der ersten steigenden Flanke der Hardware-Freigabe. Weitere steigende Flanken der Hardware-Freigabe werden während der aktuellen Ausgabesequenz vom TM Pulse 2x24V ignoriert. Die Hardware-Freigabeoption wird im taktsynchronen Betrieb nicht unterstützt.

Wenn die Freigabe aktiviert ist (steigende Flanke) und für die Dauer der Eingangsverzögerung "High" ("Ein") bleibt, wird die Einschaltverzögerung gestartet und STS_ENABLE gesetzt. Die PWM-Impulskette wird mit Ablauf der Einschaltverzögerung ausgegeben. Die Ausgabesequenz läuft kontinuierlich weiter, solange SW_ENABLE gesetzt ist.

Abbrechen der Ausgabesequenz

- Verwenden des SW_ENABLE-Signals: Durch Deaktivierung der Software-Freigabe (SW_ENABLE = 1→0) wird die aktuelle Ausgabesequenz abgebrochen und die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und die Digitalausgänge DQn.A und DQn.B werden sofort auf 1 (Motorstopp) gesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

- Verwenden des Digitaleingangs DIn.0:
  - Wenn "Funktion DI" als "HW_ENABLE" parametriert ist: Eine fallende Flanke am DIn.0 stoppt die Ausgabesequenz mit demselben Verhalten wie bei Verwendung von SW_ENABLE.
  - Wenn "Funktion DI" als „externer Stopp" parametriert ist: Eine steigende Flanke am DIn. 0 stoppt die Sequenz mit derselben Verhaltensweise wie bei Verwendung von SW_ENABLE.

Wahrheitstabelle

<table>
<tr><th>Software-Freigabe SW_ENABLE</th><th>Parameter Funktion DI</th><th>Hardware-Freigabe (Digitaleingang DIn.0)</th><th colspan="2">Digitalausgang DQn.A und DQn.B</th><th>STS_ENABLE</th><th>Ausgabesequenz</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">HW_ENABLE</td><td rowspan="3">0 → 1 und bleibt 1 während der Eingangsverzögerung.
Nur aktiv bei der ersten steigenden Flanke, weitere steigenden Flanken werden ignoriert, kein Start.</td><td colspan="2">0, wenn Einschaltverzögerung > 0:
DQn.A: 0
DQn.B: 0</td><td rowspan="3">0 → 1</td><td rowspan="3">Starten</td></tr>
<tr><td colspan="2">Wenn Einschaltverzögerung = 0:</td></tr>
<tr><td>Vorwärts:
DQn.A: 1
DQn.B: 0</td><td>Rückwärts:
DQn.A: 0
DQn.B: 1</td></tr>
</table>

Software-Freigabe SW_ENABLE	Parameter Funktion DI	Hardware-Freigabe (Digitaleingang DIn.0)	Digitalausgang DQn.A und DQn.B		STS_ENABLE	Ausgabesequenz
0 → 1	Eingang oder externer Stopp	Beliebiger Zustand	Wenn Einschaltverzögerung > 0 DQn.A: 0 DQn.B: 0		0 → 1	Starten
			Wenn Einschaltverzögerung = 0			
			Vorwärts: DQn.A: 1 DQn.B: 0	Rückwärts: DQn.A: 0 DQn.B: 1		
0	HW_ENABLE oder Eingang oder externer Stopp	Beliebiger Zustand	DQn.A: Tri-Zustand DQn.B: Tri-Zustand		0	abbrechen
1	Eingang	Beliebiger Zustand	Wenn die Einschaltverzögerung nicht abgelaufen ist oder sich in der Inter-Impulsdauer befindet: DQn.A: 0 DQn.B: 0			-
			Wenn die Einschaltverzögerung abgelaufen und sich in der Impulsdauer befindet:			
			Vorwärts: DQn.A: 1 DQn.B: 0	Rückwärts: DQn.A: 0 DQn.B: 1		
1	HW_ENABLE	0	DQn.A: Tri-Zustand DQn.B: Tri-Zustand		1 → 0	Stopp
1	Externer Stopp	0 → 1	DQn.A: Tri-Zustand DQn.B: Tri-Zustand		1 → 0	Stopp
0 → 1	HW_ENABLE	0	DQn.A: Tri-Zustand DQn.B: Tri-Zustand		0	-

Einstellen und Ändern der Impulseinschaltdauer und Drehrichtung

OUTPUT_VALUE weist die Einschaltdauer und die Richtung für die aktuelle Periodendauer zu.

OUTPUT_VALUE wird als S7-Analogwert angegeben, das Vorzeichen gibt die Richtung der Motordrehung an ("+" bedeutet vorwärts, "-" bedeutet rückwärts). Der mögliche Bereich liegt zwischen -27.648 und +27.648.

Es werden nur die beiden niederwertigsten Bytes von OUTPUT_VALUE verwendet, die beiden anderen Bytes werden ignoriert. Ein Wert der beiden niederwertigsten Bytes größer als +27.648 wird als Wert +27.648 (100 % vorwärts) interpretiert. Ein Wert der beiden niederwertigsten Bytes kleiner als +27.648 wird als Wert -27.648 (100 % rückwärts) interpretiert.

Vor Änderung der Drehrichtung wird empfohlen, OUTPUT_VALUE ausreichend lang auf 0 zu stellen, damit der Motor zunächst gestoppt wird. Bitte berücksichtigen Sie, dass das TM Pulse 2x24V-Modul die zugewiesene Einschaltdauer zum Schutz des Motors nicht übersteuert. Wenn ein Hochlauf oder Rücklauf motorseitig erforderlich ist, muss diese Rampe im

Automatisierungsprogramm implementiert und in das Modul durch entsprechende Ansteuerung des Wertes des OUTPUT_VALUE-Feldes übertragen werden.

- Ausgabeformat "S7-Analogausgabe": Wertebereich zwischen -27.648 und +27.648. Das Vorzeichen bestimmt die Drehrichtung des Motors.
- Impulsdauer = (OUTPUT_VALUE/27.648) x Periodendauer.

Sie weisen OUTPUT_VALUE direkt über das Steuerungsprogramm zu. Ein neuer OUTPUT_VALUE wird bei der nächsten steigenden Flanke am Ausgang übernommen.

Einstellen und Ändern der Periodendauer

- Permanente Aktualisierung
  Die Periodendauer wird über die Steuerschnittstelle permanent gesteuert. Das MODE_SLOT-Bit muss eingestellt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben.
  Stellen Sie den Periodendauerwert im Feld SLOT zwischen 100 µs und 85.000.000 µs ein.
- Einzelne Aktualisierung
  Stellen Sie die Periodendauer in den Konfigurationsparametern ein. Alternativ nehmen Sie eine einzelne Aktualisierung über die Steuerschnittstelle vor. MODE_SLOT muss gelöscht werden ("0" bedeutet einzelne Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben.
  Stellen Sie den Periodendauerwert im Feld SLOT zwischen 100 µs und 85.000.000 µs ein.

Eine neue Periodendauer wird bei der nächsten steigenden Flanke am Ausgang übernommen.

Weitere Informationen zur Handhabung des Parameters SLOT finden Sie unter "Handhabung des SLOT-Parameters (Steuerschnittstelle)".

Taktsynchroner Betrieb

Allgemeine Informationen finden Sie unter "Funktion: Taktsynchroner Betrieb".

Im taktsynchronen Betrieb wird die Ausgabesequenz mit dem Moment T_O synchronisiert. Die Periodendauer wird auf den Anwendungszyklus abgestimmt (d. h. den synchronen Zyklus als Vielfaches des PROFINET-Zyklus). Das Verhalten in der Betriebsart Gleichstrommotor ist mit dem Verhalten in der Betriebsart PWM identisch. Lesen Sie hierzu bitte das entsprechende Kapitel in diesem Handbuch. Nachstehend werden ausschließlich Unterschiede aufgeführt:

- Während der "Impulsdauerphase" (Hoch-Phase) lautet der Zustand der Ausgänge wie folgt:
  - In Vorwärtsrichtung: DQn.A ist 1 und DQn.B ist 0
  - In Rückwärtsrichtung: DQn.A ist 0 und DQn.B ist 1
- Während der „Impulspausephase" (Niedrig-Phase) lautet der Zustand der Ausgänge wie folgt:
  - DQn.A ist 0 und DQn.B ist 0
- Der Parameter "Funktion DI" kann im taktsynchronen Betrieb als "externer Stopp" parametriert werden, um die steigenden Flanken am DIn.0 zum Stoppen des Motors zu verwenden. Bitte berücksichtigen Sie, dass bei Parametrierung von "Funktion DI" als "HW_ENABLE" im taktsynchronen Betrieb, was nicht unterstützt wird, diese Parametrierung vom TM Pulse 2x24V-Modul als "externer Stopp" interpretiert wird.

Einstellen und Ändern der Einschaltverzögerung

- Permanente Aktualisierung
  Die Einschaltverzögerung kann über die Steuerschnittstelle permanent gesteuert werden. Das MODE_SLOT-Bit muss eingestellt werden (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs im Feld SLOT ein. Die Einheit ist immer die Mikrosekunde.
- Einzelne Aktualisierung
  Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs in den Konfigurationsparametern ein. Die Einheit ist immer die Mikrosekunde. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden (permanente Aktualisierung); LD_SLOT muss den Wert 2 (für Einschaltverzögerung) haben. Stellen Sie die Einschaltverzögerung auf einen Wert zwischen 0 µs und 85.000.000 µs im Feld SLOT ein.

Wenn Sie den Wert der Einschaltverzögerung während der Ausgabesequenz ändern, wird die neue Einschaltverzögerung mit der nächsten Ausgabesequenz aktiviert. Weitere Informationen zur Verwendung des Parameters SLOT finden Sie unter Handhabung des SLOT-Parameters.

Parameter für die Betriebsart Gleichstrommotor

Parameter	Bedeutung	Wertebereich	Voreinstellung
Betriebsart	5 = Einstellen der Betriebsart Gleichstrommotor	0 = Impulsausgabe 1 = Impulsweitenmodulation 2 = Impulskette 3 = Ein-/Ausschaltverzögerung 4 = Frequenzausgabe 5 = Gleichstrommotor	1
Funktion DI	Sie können den Digitaleingang DIn.0 als Eingang für HW_ENABLE oder für das Signal „externer Stopp“ verwenden. Das Signal am DIn. 0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung interpretiert. Wenn HW_ENABLE im nicht-taktsynchronen Betrieb verwendet wird, startet eine steigende Flanke an DIn.0 die Ausgabesequenz (wenn SW_ENABLE gesetzt ist) und eine abfallende Flanke an DIn.0 stoppt die Ausgabesequenz. Wenn HW_ENABLE im taktsynchronen Betrieb verwendet wird, wird es vom Modul als externer Stopp interpretiert. Wenn der externe Stopp verwendet wird, startet die Ausgabesequenz mit der steigenden Flanke von SW_ENABLE und stoppt mit der steigenden Flanke von DIn.0.	0 = Eingang 1 = HW-Freigabe 2 = externer Stopp	Eingang
Ausgabeformat	Definiert das Format des Verhältniswertes (Einschaltdauer).	0 = S7-Analogformat	S7-Analogformat

Parameter	Bedeutung	Wertebereich	Voreinstellung
Eingangsverzögerung	Der Digitaleingang DIn.0 muss über die Dauer der Verzögerung stabil bleiben (Unterdrückung von Signalrauschen).	0 = Aus (4 µs) 1 = 0,05 ms 2 = 0,1 ms 3 = 0,4 ms 4 = 0,8 ms 5 = 1,6 ms 6 = 3,2 ms 7 = 12,8 ms 8 = 20 ms	0,1 ms
Periode	Periodendauer des Ausgangsimpulszyklus in µs. Sie können die Periodendauer im Steuerungsprogramm im Feld SLOT der Steuerschnittstelle ändern.	100 µs bis 85.000.000 µs	1000 µs
Einschaltverzögerung	Die Dauer vom Start der Ausgabesequenz bis zur Impulsausgabe. Sie können die Einschaltverzögerung im Steuerungsprogramm im Feld SLOT der Steuerschnittstelle ändern.	0 µs bis 85.000.000 µs	0 µs

Steuer- und Rückmeldesignale für Betriebsart Gleichstrommotor

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung				
Kanal 0	Kanal 1 [1]						
Bytes 0 bis 3	Bytes 12 bis 15	OUTPUT_VALUE (DWord)	Der OUTPUT_VALUE bestimmt die Einschaltdauer (Verhältnis Impulsdauer/Periodendauer) innerhalb einer Periode (PWM). Die Periodendauer ist anpassbar. Der neue Ausgabewert wird mit der nächsten steigenden Flanke der Ausgabe übernommen. Das Vorzeichen OUTPUT_VALUE kennzeichnet die Drehrichtung (positiv für vorwärts und negativ für rückwärts). Ausgabeformat „S7-Analogausgabe“: Wertebereich -27.648 bis +27.648 DInt-Datentyp: Es werden nur die 2 niederwertigsten Bytes verwendet Für Kanal 0: Bytes 2 und 3 Für Kanal 1: Bytes 14 und 15				
Bytes 4 bis 7	Bytes 16 bis 19	SLOT (DWord)	Ihr Programm kann die Einschaltverzögerung und Periodendauer vor dem Start der Ausgabesequenz mithilfe der Parameter SLOT und MODE_SLOT ändern.				
			0 µs bis 85.000.000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation des SLOT-Wertes: Alle übrigen nachfolgend nicht aufgeführten Werte sind ungültig und verursachen den Fehler ERR_LD (in der Betriebsart für einzelne Aktualisierung) oder ERR_SLOT_VAL (in der Betriebsart für permanente Aktualisierung).				
			Bit 3	Bit 2	Bit 1	Bit 0	
			0	0	0	0	Leerlauf; der Wert wird nicht verarbeitet
			0	0	0	1	Periode in µs
			0	0	1	0	Einschaltverzögerung in µs
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	Bit 4	Betriebsart zur Verwendung des Feldes SLOT.			
			0	Betriebsart für einzelne Aktualisierung			
			1	Betriebsart für permanente Aktualisierung			

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software-Freigabe: Starten und Beenden der Ausgabesequenz.
			0	Ausgabe abgebrochen
			0 → 1	Startet Ausgabesequenz bei steigender Flanke wenn "Funktion DI" = "Eingang" oder "Funktion DI" = "externer Stopp".
			1	Aktivieren der Ausgabesequenz, wenn Start abhängig ist von HW-Freigabe mit "Funktion DI" = „HW-Freigabe".
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Die Ausgänge werden vom Modul in der Betriebsart Gleichstrommotor immer gesteuert. Dieses Bit wird ignoriert.
			X	Gleichgültig: Keine Auswirkung auf die Ausgänge.
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ = 0.
			X	Gleichgültig: Keine Auswirkung auf die Ausgänge.
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ = 0.
			X	Gleichgültig: Keine Auswirkung auf die Ausgänge.
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Rücksetzen anstehender Fehler (ERR_LD, ERR_DQA, ERR_DQB und ERR_24V).
			0	Rücksetzen von Fehlern ist nicht aktiv
			1	Rücksetzen von Fehlern ist aktiv

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und müssen 0 sein.

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Zeigt Unterspannung in der Spannungsversorgung an. Das Bit ist nicht gesetzt, wenn keine Spannung vorhanden ist.
			0	PWR weist keine Unterspannung auf
			1	PWR wurde erkannt, weist jedoch Unterspannung auf
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Zeigt einen Kurzschluss oder eine Überlast am 24 V DC-Ausgang an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an 24 V DC
			1	Kurzschluss an 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Zeigt einen Fehler beim Laden eines Wertes mit dem Feld SLOT an (nur in der Betriebsart für einzelne Aktualisierung SLOT).
			0	Kein Lastfehler anstehend
			1	Lastfehler anstehend: Um diesen Fehler zurückzusetzen und SLOT erneut verwenden zu können, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1 [1]			
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Zeigt an, dass ein ungültiger Wert in SLOT erkannt wurde (nur in der Betriebsart für permanente Aktualisierung SLOT).
			0	SLOT-Wert ist gültig
			0 → 1	SLOT-Wert ist ungültig
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Umschaltquittierbit für jede Aktion des SLOT in der SLOT-Betriebsart für einzelne Aktualisierung. Jede Umschaltung dieses Bits bedeutet eine erfolgreiche LD_SLOT-Aktion.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Zeigt an, dass das Modul bereit und parametriert ist.
			0	Modul ist nicht parametriert
			1	Modul ist parametriert
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Zeigt den Zustand von SW_ENABLE (Steuerschnittstelle) an.
			0	SW_ENABLE gelöscht
			1	SW_ENABLE gesetzt
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE	**Bit 0**	Zeigt eine laufende Ausgabesequenz an.
			0	Ausgabesequenz läuft nicht
			1	Ausgabesequenz läuft
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	0 an Digitalausgang DQn.A
			1	1 an Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	0 an Digitalausgang DQn.B
			1	1 an Digitalausgang DQn.B
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Zeigt den Signalpegel am Digitaleingang DIn.0 an.
			0	0 an Digitaleingang DIn.0
			1	1 an Digitaleingang DIn.0
Byte 3: Bit 0 bis 3	Byte 11: Bit 0 bis 3	SEQ_CNT	Sequenzzähler = 0. Der Sequenzzähler wird in der Betriebsart PWM nicht verwendet.	
Word 3	Word 7	Reserviert	Als 0 lesen	

[1] Nur wenn das Modul als "2-kanalig (2 A)" und nicht als "1-kanalig (4 A)" konfiguriert ist

Hinweis: Alle in der vorstehenden Tabelle nicht beschriebenen Bytes und Bits sind reserviert und werden als 0 gelesen.

Eingangs- und Ausgangssignale für Betriebsart Gleichstrommotor

Eingangs- und Ausgangssignal	Bedeutung	Wertebereich	Kanal 0 BaseUnit Pinnummer	Kanal 1 BaseUnit Pinnummer
Eingangssignal				
HW-Freigabe/externer Stopp Hinweis: Die HW-Freigabe wird für PWM im taktsynchronen Betrieb nicht unterstützt	Sie können die Interpretation des Digitaleingangssignals an der Ausgabesequenz mit dem Parameter "Funktion DI" auswählen und die Eingangsverzögerung mit dem Parameter "Eingangsverzögerung" auswählen. Das Signal am Digitaleingang DIn. 0 wird vom TM Pulse 2x24V nach der Rauschfilterung durch die Eingangsverzögerung als Start der Ausgabesequenz interpretiert.	Funktion DI = "HW_ENABLE": 0 = HW-Freigabe gelöscht 1 = HW-Freigabe ausgegeben 0→1 = Start der Ausgabesequenz nach der Eingangsverzögerung, abhängig von der Software-Freigabe (SW_ENABLE) 1→0 = Stopp der Ausgabesequenz nach der Eingangsverzögerung, abhängig von der Software-Freigabe (SW_ENABLE). Funktion DI = externer Stopp: 0→1 = Stopp der Ausgabesequenz nach der Eingangsverzögerung, abhängig von der Software-Freigabe (SW_ENABLE)	3	4
Ausgangssignal				
Impuls an den Digitalausgängen DQn.A und DQn.B	Ein Impuls wird an den Digitalausgängen DQn.A und DQn.B für die zugewiesene Einschaltdauer und Periodendauer ausgegeben.	**Keine Impuls- oder Impulspauseperiode:** DQn.A ist 0 V und DQn.B ist 0 V **Impulsdauer in Vorwärtsrichtung:** DQn.A ist 24 V und DQn.B ist 0 V **Impulsdauer in Rückwärtsrichtung:** DQn.A ist 0 V und DQn.B ist 24 V	9 (DQ0.A) 11 (DQ0.B)	10 (DQ1.A) 12 (DQ1.B)

Siehe auch Anschlussbelegung und Last-/Geberverdrahtung

5.1.2.8 Funktion: Schnelle Impulsausgabe

Die Betriebsart Schnelle Impulsausgabe verbessert den Signaltakt der DQ-Digitalausgänge. An den Schaltflanken treten weniger Verzögerung, Schwankungen, Jitter sowie kürzere Anstiegs-/Fallzeiten auf.

Die Betriebsart Schnelle Impulsausgabe eignet sich dazu, Impulssignale in einem präziseren Takt zu erzeugen, bietet jedoch einen geringeren maximalen Laststrom.

Die schnelle Impulsausgabe ist nur im Zweikanalbetrieb und nicht im Einkanalbetrieb verfügbar.

Sie können die STEP 7 (TIA Portal)- oder STEP 7-Hardwarekonfiguration zum Auswählen der „Schnellen Impulsausgabe (0,1 A)" für jeden Kanal separat verwenden. Außerdem können Sie die Parametrierung zur Laufzeit mithilfe des Programms über den Datensatz 128 ändern.

STEP 7 (TIA Portal) und STEP 7 unterstützen Sie bei der Parametrierung, indem sie die Tastatureingabe ungültiger Parameter verhindern und den Wertebereich der eingegebenen Werte prüfen. Je nachdem, welche Parameter zuvor ausgewählt wurden, sind bestimmte Optionen deaktiviert. Wenn Sie beispielsweise den Einkanalbetrieb (Parallelschaltung beider Ausgangskanäle) auswählen, sind die Parameteroptionen für Kanal 1 und die schnelle Impulsausgabe deaktiviert.

Die schnelle Impulsausgabe ist für folgende Betriebsarten verfügbar:

- Impulsausgabe
- PWM

- Impulskette
- Ein-/Ausschaltverzögerung
- Frequenzausgabe

Die schnelle Impulsausgabe ist in der Betriebsart Gleichstrommotor nicht verfügbar.

Option Schnelle Impulsausgabe

Impulstakt	Minimum		Maximum	
	Schnelle Impulsausgabe deaktiviert	Schnelle Impulsausgabe aktiviert	Schnelle Impuls-ausgabe deaktiviert	Schnelle Impulsausga-be aktiviert
Impulsdauer	10 µs	1,5 µs	85.000.000 µs	
Periodendauer	100 µs	10 µs		
Einschaltverzöge-rung	0 µs			
Ausschaltverzöge-rung				
Frequenz	0,02 Hz		10 kHz	100 kHz

Schnelle Impulsausgabe Laststrom

Parallele Betriebsart (Einkanalbetrieb)	Laststrom bei Schneller Impulsausgabe	
	Schnelle Impulsausgabe deaktiviert	Schnelle Impulsausgabe aktiviert
Deaktiviert	2 A (zwei Kanäle)	100 mA (zwei Kanäle)
Aktiviert	4 A (ein Kanal)	Nicht zulässig

5.1.2.9 Funktion: Sequenzzähler

Das TM Pulse 2x24V besitzt pro Kanal einen Sequenzzähler, der abgeschlossene Ausgabesequenzen zählt. Es werden erfolgreich abgeschlossene und nicht erfolgreich abgeschlossene Ausgabesequenzen gezählt.
Sie können den Abschluss einer Ausgabesequenz mit der Variablen SEQ_CNT des Sequenzzählers in der Rückmeldeschnittstelle überwachen.

Der Zähler hat eine Breite von 4 Bits. Nach einem Überlauf springt der Zähler zurück auf 0.

Funktion des Sequenzzählers

In den einzelnen Betriebsarten hat der Sequenzzähler folgende Funktion:

- Impulsausgabe und Impulskette
  - Die Funktion DI-Konfiguration wird auf "Eingang" gesetzt: Der Zähler wird nach Abschluss der Ausgabesequenz auf 1 gesetzt (Bereich 0 bis 1).
  - Die Funktion DI-Konfiguration wird auf "HW-Freigabe" gesetzt: Der Zähler wird nach jeder abgeschlossenen Ausgabesequenz erhöht (Bereich 0 bis 15).
- Ein-/Ausschaltverzögerung
  - Der Zähler wird mit jeder Flanke (steigend und fallend) am DQ-Ausgang (Bereich 0 bis 15) erhöht.
- PWM, Frequenzausgabe und Gleichstrommotor
  - Der Zähler hat keine Funktion.

Kommt es zur Software-Freigabe (SW_ENABLE = 0), wird der Zähler auf 0 zurückgesetzt.

Anwendungsmöglichkeiten

Der Sequenzzähler ist für Folgendes einsetzbar:

- Erkennen (Zählen) sehr kurzer Impulssequenzen
- Zählen von mit der Hardware-Freigabe gesteuerten Ausgabesequenzen

5.1.2.10 Funktion: Strommessung

Funktionsprinzip

Ihre Programmlogik kann Laststrommessungen mit einer Steuerschleife dazu nutzen, die an eine induktive oder ohmsche Last übertragene Energie proportional zu steuern. Die Strommessungen werden im Rückmeldeschnittstellen MEASURED_CURRENT-Wert im SIMATIC S7-Analogwerteformat bereitgestellt.

Die Strommessung ist möglich:

- In den Betriebsarten PWM und Impulskette und wenn die Option Schnelle Impulsausgabe inaktiv ist
  - Wenn es keine Impulsausgabe (bei STS_ENABLE = 0) gibt, wird 7FFFH als Messwert ausgegeben.
  - Der Messwert ist gültig bei (STS_ENABLE = 1) nach der ersten Periodendauer. Der zurückgegebene Strommesswert ist ein Mittelwert der Messwerte, die über die Dauer mindestens einer Periode erfasst wurden.
  - Bei aktivem Dithering wird der Mittelwert über die Dauer der gesamten Dither-Periode erfasst.

Die Strommessung ist nicht möglich und gibt den MEASURED_CURRENT-Wert 0 zurück:

- Bei allen übrigen Betriebsarten oder wenn die Option Schnelle Impulsausgabe aktiv ist.

Hinweis

Damit die Strommessung ordnungsgemäß funktioniert, schließen Sie keine Freilaufdiode (Überspannungsschutz-/Suppressordiode) an die Ausgangslast an.

Messbereich und Messwert

- **Die Kanalkonfiguration ist 2-kanalig (2 A)** (die parallele Kanalschaltung ist deaktiviert).
  - Ein gemessenes Strommittel von 2 A entspricht dem SIMATIC S7-Analogwert 27.648 ($6C00_H$).
  - Strommessungen sind bis zu einem SIMATIC S7-Analogwert von 32.511 ($7EFF_H$) möglich, was einem Strom von 2,37 A entspricht.
  - Der Strom darf nur kurzzeitig 2 A überschreiten.
- **Die Kanalkonfiguration ist 1-kanalig (4 A)** (die parallele Kanalschaltung ist aktiviert).
  - Ein gemessenes Strommittel von 4 A entspricht dem SIMATIC S7-Analogwert 27.648.
  - Strommessungen sind bis zu einem SIMATIC S7-Analogwert von 32.511 möglich, was einem Strom von 4,74 A entspricht.
  - Der Strom darf nur kurzzeitig 4 A überschreiten.

Diagnosemeldung beim Erreichen der Ausgangsstromgrenze

Wenn die Diagnose aktiviert wird und sich das Modul in der Betriebsart PWM oder PTO befindet, dann wird ein Überstrom-Diagnosefehler gemeldet, wenn das Modul einen Strom über 2,37 A (4,74 A im Einkanalmodus) misst.

Genauigkeit der Strommessung

Die Genauigkeit bei der Strommessung beträgt ±2 % des Messbereichs-Endwerts (27.648)

- ±40 mA in der Zweikanal-Betriebsart
- ±80 mA in der Einkanal-Betriebsart

Die Genauigkeit der Strommessung sinkt oberhalb einer Impulsfrequenz von 3 kHz auf eine Genauigkeit von -2% bis +5% ab, wenn ohmsche Lasten bei einer Frequenz von 10 kHz angesteuert werden.

Hinweis

Eine Genauigkeit von 2% bei der Strommessung ist nur dann möglich, wenn die Periodendauer während des Messvorgangs nicht geändert wird.

5.1.2.11 Funktion: Stromregelung

Stromregelung (nur Betriebsart PWM)

In der Betriebsart PWM kann das TM Pulse 2x24V-Modul einen PID-(Proportional-Integral-Differential)-Algorithmus zur Regelung des Ausgangsstroms verwenden. Der Sollwert des Stroms wird vom Programm festgelegt. Das Modul steuert die Einschaltdauer der PWM-Ausgabe entsprechend dem Sollwert, wobei die Reaktion auf den zugeordneten PID-Parametern beruht.

Der Sollwert und der Messwert des Laststroms werden vom PID-Regler miteinander verglichen. Wenn die resultierende Differenz (der Fehler) außerhalb einer symmetrischen Totzone liegt, wird eine Reaktion des Reglers durch Berücksichtigung des jeweils ausgewählten Anteils des PID-Reglers (proportional, integral und/oder differenziell) berechnet. Der so manipulierte Wert wird durch die definierte Ober-/Untergrenze begrenzt und die Ausgangsimpulsdauer (Einschaltdauer) wird in Abhängigkeit von der PWM-Periodendauer modifiziert.

Weisen Sie den Referenzstromwert dem Strom zu, der in der DQn.A-Ausgangslast des Kanals gemessen wird, wenn sich der Ausgang kontinuierlich im "High"-Zustand (Ein) befindet.

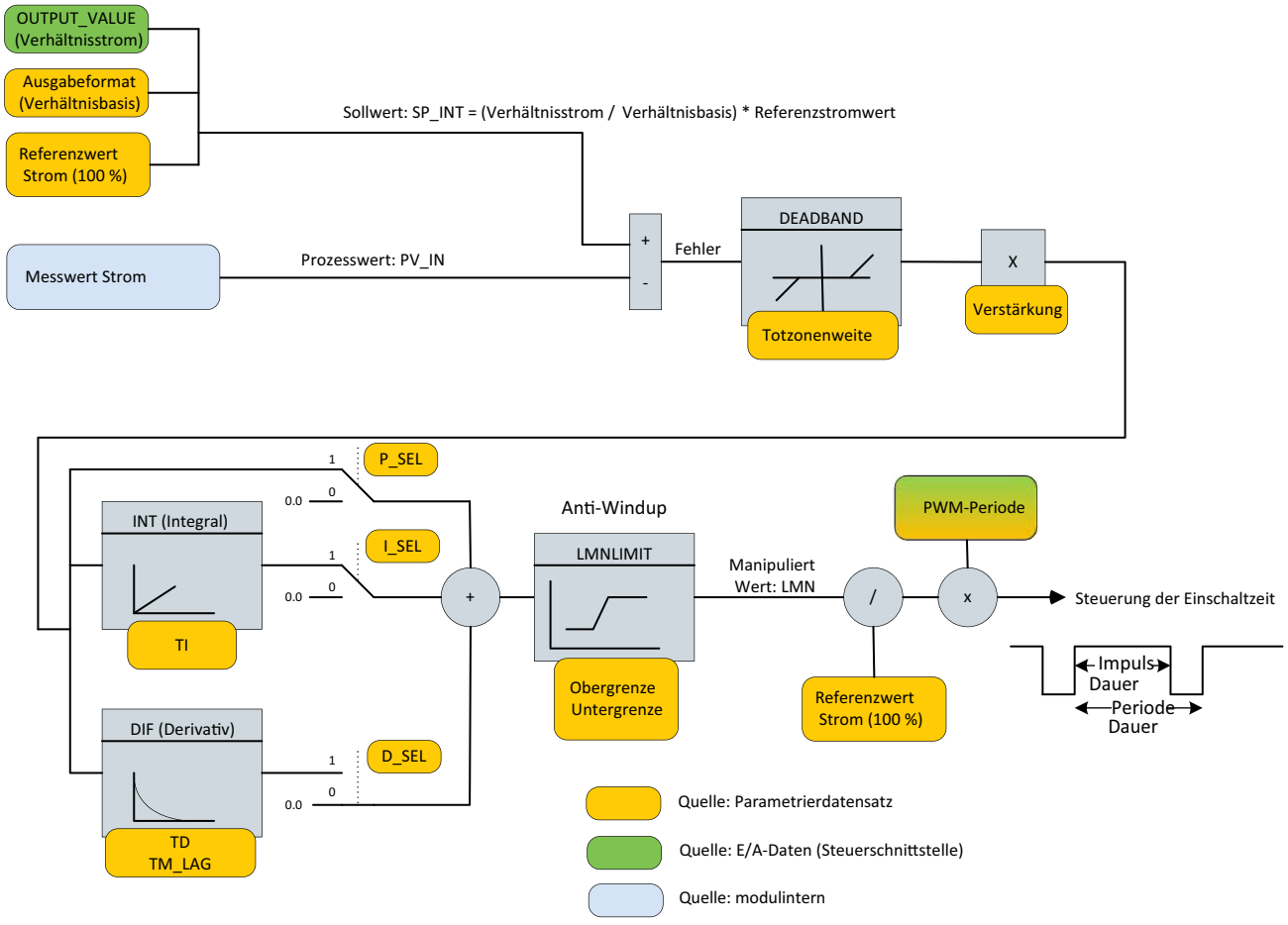

Bild 5-14 PID-Funktion zur Stromregelung

Ausführliche Informationen zu dieser PID-Regler-Methode finden Sie unter der CONT_C-Anweisung in der Online-Hilfe des TIA Portal.

Einstellen der Stromregelung

Um die Stromregelungsfunktion zu verwenden, muss das Modul korrekt parametriert und der Sollwert (Strom-Zielwert) muss vom Anwenderprogramm gesteuert werden. Außerdem kann das Programm die Grenze erreichte Merker lesen.

PID-Parameter

Die PID-Parameter können in der TIA Portal-Gerätekonfiguration oder vom Parametrierungsdatensatz (Datensatz 128) des Programms im Modul eingestellt werden. Nähere Informationen finden Sie in der Beschreibung des Parametrierungsdatensatzes.

- Stromregelung: "1" aktiviert den PID-Regler. Einschränkungen: Die Stromregelung kann nur bei ausgewählter Betriebsart PWM und deaktiviertem schnellen Impulsausgang aktiviert werden.
  Die folgenden Parameter haben keine Auswirkungen, wenn die Stromregelung deaktiviert ist.
- P_Sel, I_Sel, D_Sel: Definiert die Berechnung der PID-Regelung. Der proportionale, integrale und differenzielle Anteil der Regelung kann in der Gerätekonfiguration aktiviert ("1") oder deaktiviert ("0") werden.
- Referenzstromwert (mA): Der Referenzwert wird zur Definition des maximalen Sollwertes und der oberen und unteren Grenze der Stromregelung verwendet. Der maximale Strom kann üblicherweise in der Betriebsart PWM bei deaktivierter Stromregelung und einer Einschaltdauer von 100% gemessen werden. Der gemessene Wert kann als Referenz für die Stromregelung herangezogen werden. Der Höchstwert beträgt 4000 mA im Einkanalbetrieb (parallele Kanalschaltung aktiviert) und 2000 mA pro Kanal im Zweikanalbetrieb (Parallelschaltung deaktiviert).
- Totzonenbreite (µA): Eine Totzone wird auf die Ausgangsstromabweichung vom Sollwertstrom angewandt. Die "Totzonenbreite" entspricht der halben Größe der Totzone. Die Totzone ist symmetrisch: Wenn beispielsweise die Totzonenbreite = 1500 µA, dann liegt die Totzone zwischen -1500 µA und +1500 µA.
- Obere Grenze (S7-Analogwert): Der manipulierte Wert ist immer auf eine obere und eine untere Grenze beschränkt. Der Parameter "Obere Grenze" weist die obere Grenze im S7-Analogformat relativ zum Referenzstromwert zu. Ein Wert größer oder gleich 27.648 bedeutet 100% des Referenzstromwertes. Der Wert der oberen Grenze muss größer sein als der der unteren Grenze.
  Beispiel für eine obere Grenze:
  - Der Referenzstrom ist auf 1000 mA gesetzt, was dem mit einer Einschaltdauer von 100% gemessenen Laststrom entspricht.
  - Wenn die obere Grenze 20000 ist, dann beträgt der Maximalstrom, den der Regler ansteuert, 20000/27648 * 1000 mA = 723,4 mA, was einer Einschaltdauer von 72,34% entspricht.
  - Wenn der Regler einen Ansteuerungswert größer als 72,34% ermittelt, dann steuert er mit 72,34% an und überschreitet diesen Wert zu keinem Zeitpunkt.

- **Untere Grenze (S7-Analogwert):** Der manipulierte Wert ist immer auf eine obere und eine untere Grenze beschränkt. Der Parameter "Untere Grenze" weist die untere Grenze im S7-Analogformat relativ zum Referenzstromwert zu. Ein Wert größer als oder gleich 27.648 ist nicht zulässig. Der Wert der unteren Grenze muss kleiner sein als der der oberen Grenze.
  Beispiel für eine untere Grenze:
  - Der Referenzstrom ist auf 1000 mA gesetzt, was dem mit einer Einschaltdauer von 100% gemessenen Laststrom entspricht.
  - Wenn die untere Grenze 100 ist, dann beträgt der Minimalstrom, den der Regler ansteuert, 100/27648 * 1000 mA = 3,6 mA, was einer Einschaltdauer von 0,36% entspricht.
  - Wenn der Regler einen Ansteuerungswert kleiner als 0,36% ermittelt, dann steuert er mit 0,36% an und fällt zu keinem Zeitpunkt unter diesen Wert.
- **Verstärkung:** Der Parameter der Proportionalverstärkung weist den Verstärkungsfaktor für den P-Anteil des PID-Algorithmus zu.
- **TI (s)**: Der Parameter der Integrationszeit bestimmt den Zeitintervall des Integralanteils. Wenn TI kleiner als die Zyklusdauer des Reglers ist, dann wird TI intern auf die Zyklusdauer des Reglers gesetzt.
- **TD (s)**: Der Parameter der Vorhaltezeit bestimmt den Zeitintervall des Vorhalts. Wenn TD kleiner als die Zyklusdauer des Reglers ist, dann wird TI intern auf die Zyklusdauer des Reglers gesetzt.
- **TM_LAG (s)**: Zeitverzögerung des Vorhalts. Der Algorithmus des Vorhalts enthält die Verzögerung TM_LAG. Wenn TM_LAG kleiner als die halbe Zyklusdauer des Reglers ist, dann wird TI intern auf die halbe Zyklusdauer des Reglers gesetzt.

Sollwertregler

Das Programm steuert den Zielwert des Ausgangsstroms durch Einstellen des Feldes OUTPUT_VALUE der Steuerschnittstelle. Das Ausgabeformat wird im Parametrierungsdatensatz ausgewählt (siehe Parameter "Ausgabeformat"). Wenn beispielsweise das Ausgabeformat "pro 100" ausgewählt wird, bedeutet das Schreiben des Wertes 60 in OUTPUT_VALUE, dass der Zielstromwert 60% des Referenzstromwertes beträgt (siehe Parameter "Referenzstromwert").

Grenze erreichte Merker

Die Grenze erreichte Merker sind in der Rückmeldeschnittstelle verfügbar.

QLMN_HLM: "1" bedeutet, dass der manipulierte Wert die obere Grenze erreicht hat.

QLMN_LLM: "1" bedeutet, dass der manipulierte Wert die untere Grenze erreicht hat.

Ausführliche Informationen zu den PID-Regler-Parametern finden Sie unter der CONT_C-Anweisung in der Online-Hilfe des TIA Portals.

Regler-Zyklusdauer

Die interne Regler-Zyklusdauer hängt vom konfigurierten Automatisierungssystem und der PWM-Periode ab. Der Regler kann den Strom nicht schneller regeln als die parametrierte PWM-Periode, da der gemessene Strom über eine gesamte PWM-Periode gemittelt wird.

Wenn die Dithering-Funktion parallel zur Stromregelung aktiv ist, dann nutzt der PID-Regler die Dither-Periodendauer als interne Regler-Zyklusdauer.

Rücksetzen des PID-Stromreglers

Die internen Daten des Stromreglers werden in den folgenden Fällen zurückgesetzt:

- SW_ENABLE (siehe Steuerschnittstelle) ist im Low-Zustand (Aus).
- Ein neuer Parametrierungsdatensatz wird an das Modul gesendet.

5.1.2.12 Funktion: Dither PWM-Ausgang

Dither-Übersicht

Die Dither-Funktion verursacht eine Schwingung in einem Proportionalventil, wenn die gewünschte Ventilposition mit Strom aus dem PWM-Ausgang geregelt wird. Die Schwingung wird dadurch induziert, dass die Dither-Stromschwankung den Zielstrom in einer PWM-Ausgangslast (Ventilspule) überlagert.

Die Dither-Schwingung verbessert die Genauigkeit und Linearität von Proportionalregelventilen. Sie können die Dither-Funktion aktivieren und konfigurieren, um durch Haftreibung und Hysterese hervorgerufene Probleme bei der Ventilsteuerung zu minimieren.

Dither-Parameter

Die Dither-Parameter können in der TIA Portal-Gerätekonfiguration oder in dem Parametrierungsdatensatz (Datensatz 128) eingestellt werden, der vom Programm an das Modul gesendet wird. Nähere Informationen finden Sie in der Beschreibung des Parametrierungsdatensatzes. Einige Dither-Parameter können zu Kalibrierungszwecken im laufenden Betrieb mithilfe des SLOT-Mechanismus in der Steuerschnittstelle geändert werden. Es wird empfohlen, die PWM-Ausgabesequenz nach einer derartigen Parameteränderung neu zu starten.

- **Dither**: "1" aktiviert die Dither-Funktion. Neben der Aktivierung von "Dither" in der PWM-Konfiguration des Moduls müssen Sie das DITHER-Bit in der Steuerschnittstelle zum Starten des Dither-Signals einstellen.
- **Dither-Amplitude (‰)**: Zuordnung des Amplitudenverhältnisses des überlagerten Dither-Signals als Promille-Verhältnis. Der zulässige Bereich liegt bei einer Einschaltdauer von 0 bis 500 ‰. Wird ein größerer Wert zugewiesen, werden vom Modul 500 ‰ verwendet. Beispiel: Wird die Dither-Amplitude auf 100 ‰ und die Einschaltdauer in der Betriebsart PWM auf 50 % gesetzt, schwankt die effektive Einschaltdauer des Signals periodisch zwischen 40 % und 60 %.
  Die Dither-Amplitude wird vom Modul dynamisch angepasst (gesenkt), wenn die ermittelte effektive Einschaltdauer größer als 100 % oder kleiner als 0% ist, damit das Dither-Signal symmetrisch bleibt.
  Wird die Dither-Amplitude beispielsweise auf 100 ‰ und die Einschaltdauer in der Betriebsart PWM auf 95% gesetzt, dann schwankt die effektive Einschaltdauer des Signals periodisch zwischen 90 % und 100 %. Die Dither-Amplitude wird immer so korrigiert, dass sie symmetrisch bleibt. Die Dither-Amplitude nimmt erneut den zugeordneten Wert an, sobald das Ausgangssignal einen ausreichenden Spielraum der Einschaltdauer eröffnet, der eine symmetrische Schwankung der überlagerten Dither-Einschaltdauer ermöglicht.

- **Dither-Periode (µs)**: Zuweisen der Periodendauer in Mikrosekunden für das überlagernde Dither-Signal. Der zulässige Bereich liegt zwischen (4 x PWM-Periode) und 100.000 µs. Außerdem muss die Dither-Periode größer als 2 ms sein. Wird ein Wert kleiner als 2 ms oder kleiner als (4 x PWM-Periode) parametriert, tritt ein Fehler auf (siehe Parametervalidierung für ERR_SLOT_VAL und ERR_LD). Wird eine Dither-Periode größer als 100 ms zugewiesen, verwendet das Modul den Wert 100 ms. Die vom Modul verwendete Dither-Periode darf nur ein gerades Vielfaches der PWM-Periode sein. Das Modul verwendet den möglichen Wert, der der zugeordneten Dither-Periode am nächsten kommt.
- **Dither-Rampen**: Der Dither-Rampen-Parameter ist ein aus zwei verschiedenen Datenwörtern bestehendes Doppelwort. Das Lowword steht für die Dither-Hochlaufzeit, das Highword für die Dither-Rücklaufzeit.
  - **Dither-Hochlaufzeit (ms):** Zuweisen der Zeit in Millisekunden für den Anstieg der Einschaltdauer von 0 % auf 100 %. Der zulässige Bereich ist 0 ms bis 30.000 ms. Wird ein höherer Wert zugewiesen, verwendet das Modul 30.000.
    Die effektive Hochlaufzeit hängt von der Dither-Amplitude und dem Nennwert der Einschaltdauer ab.
    Beispiel: Wenn der Nennwert der Einschaltdauer 50 % und die Dither-Amplitude 100 ‰ beträgt, dann schwankt die effektive Einschaltdauer zwischen 40 % und 60 %. Die Hochlaufzeit im Bereich zwischen 50 % und 60 % beträgt dann 100 ‰ x Dither-Hochlaufzeit.
  - **Dither-Rücklaufzeit (ms):** Zuweisen der Zeit in Millisekunden für den Abfall der Einschaltdauer von 100 % auf 0 %. Der zulässige Bereich ist 0 ms bis 30.000 ms. Wird ein höherer Wert zugewiesen, verwendet das Modul 30.000.
    Die effektive Rücklaufzeit hängt von der Dither-Amplitude und dem Nennwert der Einschaltdauer ab.
    Beispiel: Wenn der Nennwert der Einschaltdauer 50 % und die Dither-Amplitude 100 ‰ beträgt, dann schwankt die effektive Einschaltdauer zwischen 40 % und 60 %. Die Rücklaufzeit im Bereich zwischen 50 % und 40 % beträgt dann 100 ‰ x Dither-Rücklaufzeit.

Hinweis

Ändern der Parameter während der Dither-Ausgabe

Wird die PWM-Periode im laufenden Betrieb mithilfe des SLOT-Mechanismus geändert, sodass die Dither-Periode kleiner als (4 x PWM-Periode) wird, dann wird der Dither-Strom deaktiviert und das Rückmeldebit STS_DITHER = 0, bis eine gültige Kombination aus Dither- und PWM-Periode zugeordnet wird.

Wenn Sie die Parameterwerte während der Dither-Ausgabe mithilfe des SLOT-Mechanismus ändern, dann sollten Sie die Ausgabesequenz neu starten.

Hinweis

PWM mit Dithering im taktsynchronen Betrieb

Setzen Sie die Dither-Periodendauer auf ein gerades ganzzahliges Vielfaches des Anwendungszyklus, um ein bestmögliches Ergebnis zu erzielen.

Beispiel: Wenn der Anwendungszyklus 1 ms beträgt, dann beträgt die optimale Mindest-Dither-Periode 2 ms. Die nächsten optimalen Werte sind 4 ms, 6 ms, ...

Dither-Start und -Stopp

Der Hochlauf des Dither-Stroms beginnt, sobald das DITHER-Bit in der Steuerschnittstelle gesetzt ist, die aktuelle Dither-Periode beendet wurde und eine neue Dither-Periode startet. Die Rückmeldeschnittstelle stellt das Quittierbit STS_DITHER in der Rückmeldeschnittstelle bereit, das in den High-Zustand (Ein) übergeht, wenn die Hochlaufphase gestartet wird (durch Setzen des DITHER-Bits bei laufender Ausgabesequenz), und in den Low-Zustand (Aus) übergeht, sobald die Rücklaufphase abgelaufen ist oder die Ausgabesequenz gestoppt wurde. Der Rücklauf beginnt, wenn das DITHER-Bit in der Steuerschnittstelle gelöscht ist, die aktuelle Dither-Periode beendet wurde und eine neue Dither-Periode startet.

Beispiel 1: Kein Dither (Dither-Bit = 0)

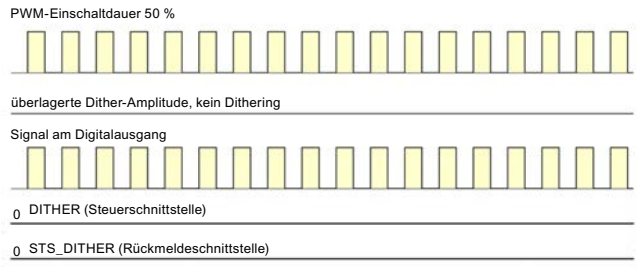

Beispiel 2: Dither ohne Rampe (Dither-Periode = 6 x PWM-Periode)

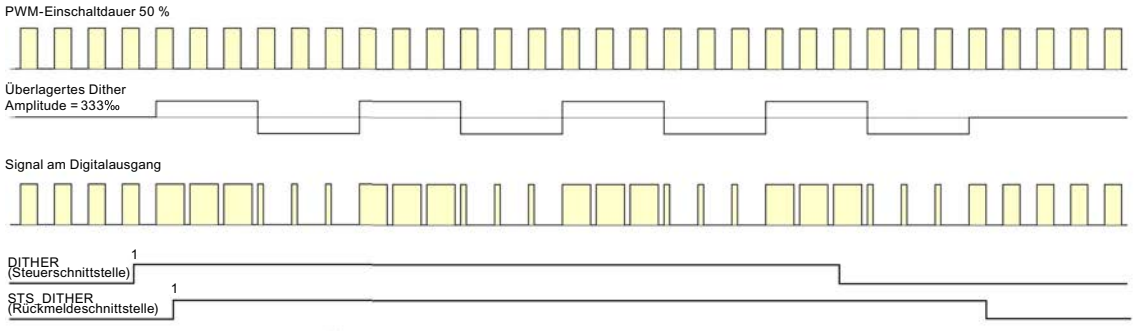

Beispiel 3: Dither mit Hochlauf

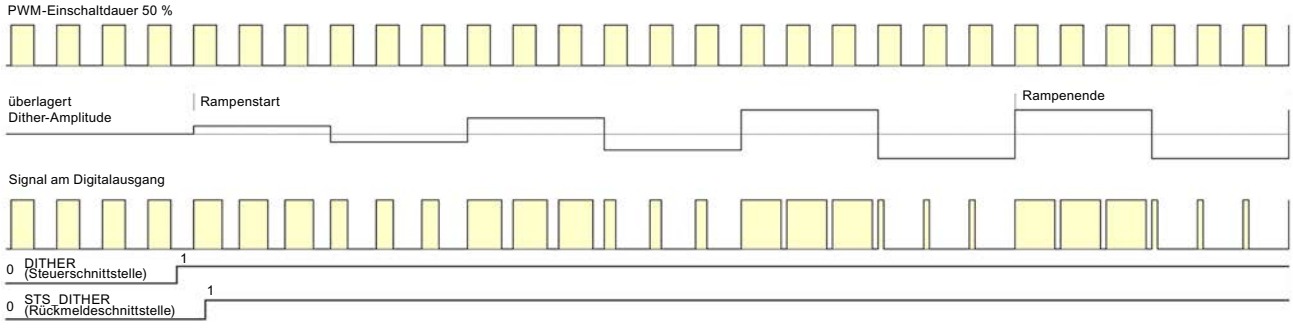

Beispiel 4: Dither mit Rücklauf

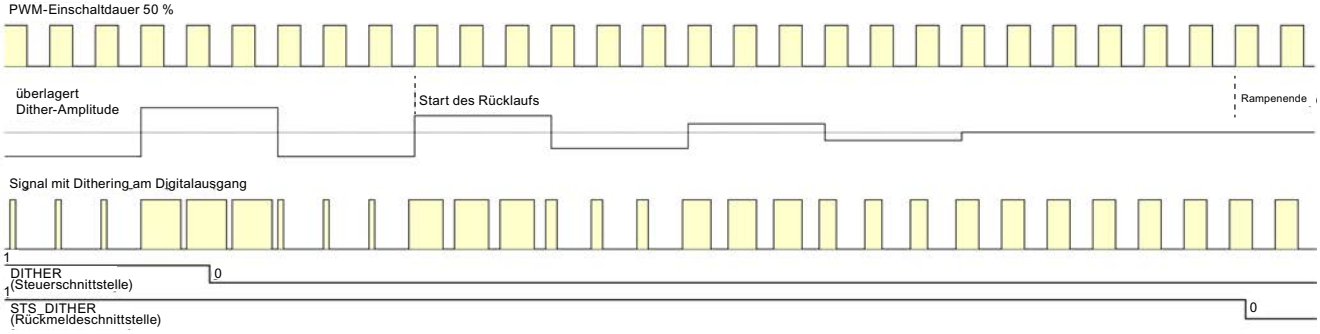

5.1.2.13 Funktion: Taktsynchroner Betrieb

Hinweis

Grundlegende Informationen zum taktsynchronen Betrieb finden Sie im Handbuch SIMATIC PROFINET mit STEP 7.

Voraussetzungen

Für den taktsynchronen Betrieb des TM Pulse 2x24V ist Folgendes erforderlich:

- CPU, die Taktsynchronität unterstützt
- Ein IM (Interfacemodul), das die Taktsynchronität unterstützt
- Engineeringsoftware wie TIA Portal oder STEP 7 zur Parametrierung der Taktsynchronität

Reaktion des TM Pulse 2x24V

In Abhängigkeit von der Systemparametrierung arbeitet das TM Pulse 2x24V entweder nicht-taktsynchron oder taktsynchron.

Im taktsynchronen Betrieb

- Die Ausgabesequenzen werden in dem Moment T_O gestartet, in dem nur die Software-Freigabe genutzt wird.
- Die Datenkommunikation zwischen der PROFINET-Steuerung und dem TM Pulse 2x24V erfolgt taktsynchron zum Zyklus.
- Alle 12 Byte der Steuerschnittstelle eines Kanals sind konsistent (24 Byte bei beiden Kanälen).
- Alle 8 Byte der Rückmeldeschnittstelle eines Kanals sind konsistent (16 Byte bei beiden Kanälen).
- In der Betriebsart PWM wird die Periodendauer mit der Zykluszeit der Anwendung (PROFINET) synchronisiert. Nähere Informationen finden Sie im Kapitel "PWM".

5.1.2.14 Funktion: Direktansteuerung von DQ-Digitalausgängen

Definition

Sie können die DQ-Digitalausgänge des TM Pulse 2x24V direkt über das Steuerungsprogramm einstellen. Wählen Sie die Funktion für die DQ-Direktansteuerung, indem Sie das Ausgangssteuerbit des Technologiemoduls (TM_CTRL_DQ = 0),) in der Steuerschnittstelle löschen.

Die Direktansteuerung eines Digitalausgangs kann bei der Inbetriebnahme eines Steuerungssystems für die Automatisierung hilfreich sein.

Wenn Sie die Direktansteuerung eines DQ während einer Impulsausgabesequenz auswählen, läuft die Sequenz im Hintergrund weiter, sodass die Ausgabesequenz fortgesetzt wird, sobald das Modul die Steuerung wieder übernimmt (durch Einstellen von TM_CTRL_DQ = 1).

Sie weisen den Zustand eines Digitalausgangs DQn.A und DQn.B mit den Steuerbits SET_DQA und SET_DQB zu.

Sie können die Direktsteuerfunktion nicht verwenden, um die beiden Ausgänge DQn.A und DQn.B im selben Kanal in den High-Zustand (Ein) zu versetzen. Kommt es zu einem derartigen Versuch, dann wird der Fehler ERR_DQB in der Rückmeldeschnittstelle gesetzt und nur der Ausgang DQn.A wird auf High gesetzt.

Wenn Sie TM_CTRL_DQ = 1 setzen, wählen Sie die Direktansteuerung der Digitalausgänge ab. Wenn die Ausgabesequenz noch läuft (STS_ENABLE noch aktiv), dann übernimmt das TM Pulse 2x24V-Modul erneut die Ansteuerung der Ausgänge DQn.A und DQn.B.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ des Technologiemoduls

- Wenn TM_CTRL_DQ = 1, übernimmt das TM Pulse 2x24V-Modul die Steuerung und erzeugt Impulssequenzen an den Ausgängen DQn.A und DQn.B. DQn.B ist immer 0, außer in der Betriebsart Gleichstrommotor.
- Wenn TM_CTRL_DQ = 0, dann übernimmt die CPU die Steuerung und das Programm kann die DQ-Ausgänge direkt mit den Steuerbits SET_DQA/SET_DQB einstellen. In der Betriebsart Gleichstrommotor können die Ausgänge DQn.A und DQn.B nicht manuell angesteuert werden; TM_CTRL_DQ, SET_DQA und SET_DQB haben keine Auswirkung.

Impulsschema

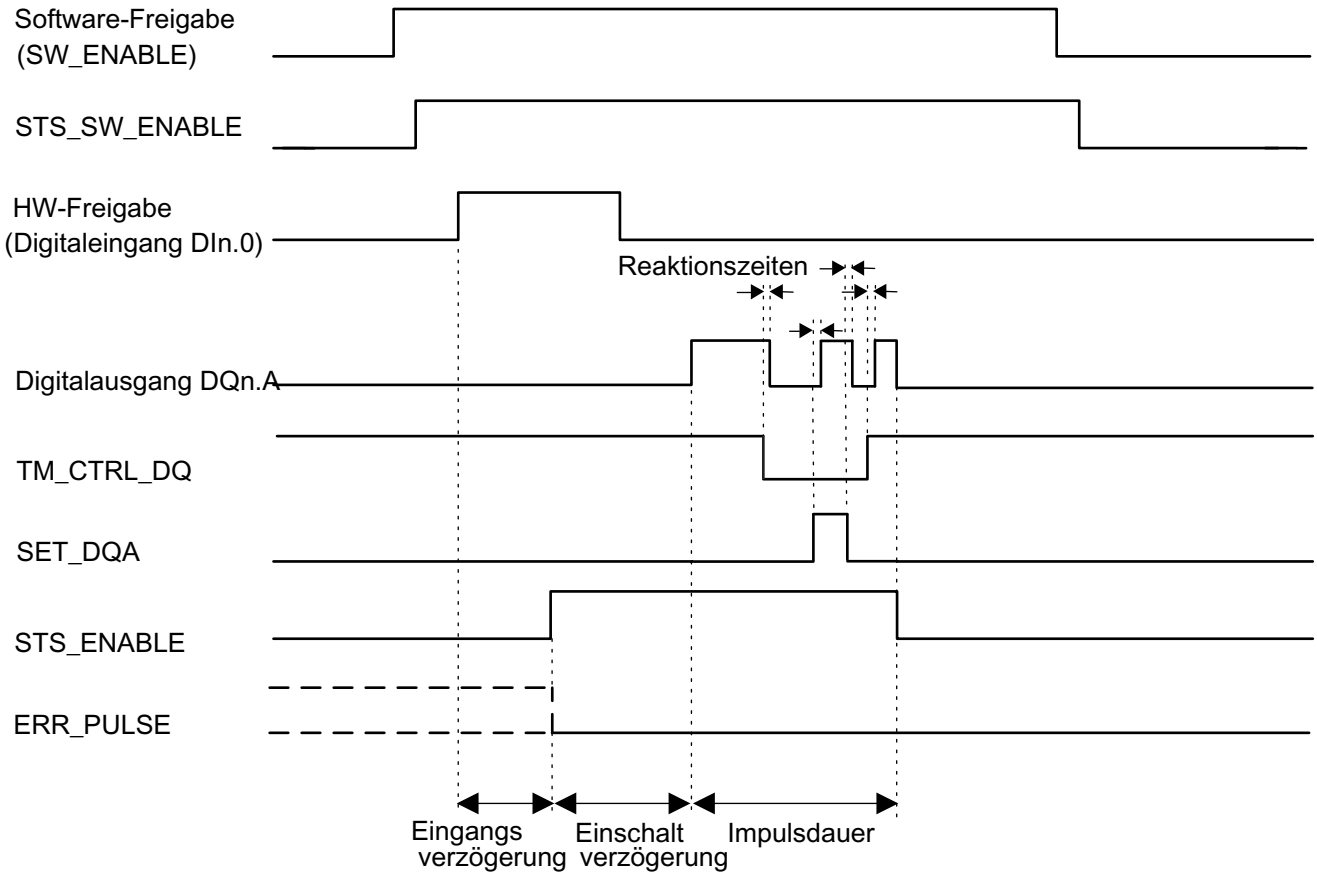

Bild 5-15 Direktansteuerung des DQ-Takts in der Betriebsart Impulsausgabe

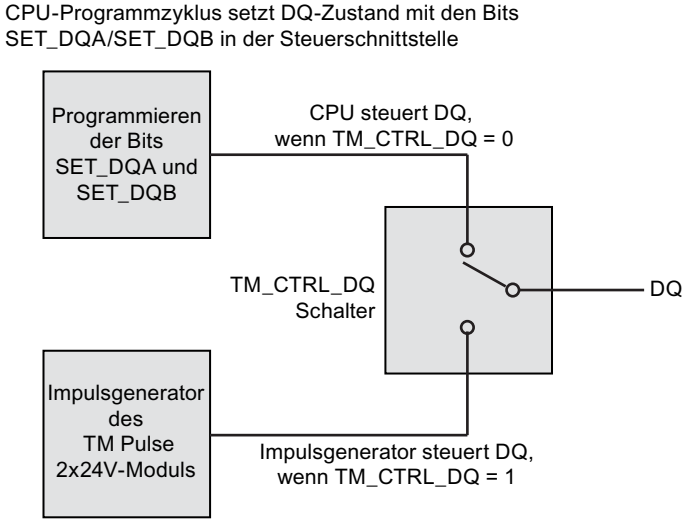

TM Pulse 2x24V-Modul setzt die Ausführung der Ausgabesequenz fort, wenn CPU-Programm DQ steuert.

Bild 5-16 Ausgangsschalter TM_CTRL_DQ

Steuer- und Rückmeldesignale

Steuerschnittstelle Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software-Freigabe: Starten und Beenden der Ausgabesequenz.
			0	Ausgabe abgebrochen.
			0 → 1 1	Startet Ausgabesequenz bei steigender Flanke; möglicherweise von der Hardware-Freigabe abhängig.
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Einstellen der DQ-Ausgabequelle: Auswählen des SPS-Programms oder der Ausgabesequenz des Moduls.
			0	DQn.A und DQn.B werden vom SPS-Programm mithilfe von SET_DQA und SET_DQB angesteuert.
			1	DQn.A wird von der Impulsausgabesequenz des Moduls gesteuert.
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Steuert den Wert des Digitalausgangs DQn.A, wenn TM_CTRL_DQ gelöscht wird.
			0	0 V an DQn.A
			1	24 V an DQn.A
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Steuert den Wert des Digitalausgangs DQn.B, wenn TM_CTRL_DQ und SET_DQA gelöscht werden.
			0	0 V an DQn.B
			1	24 V an DQn.B

Rückmeldeschnittstelle: Offset zur Anfangsadresse		Parameter	Bedeutung	
Kanal 0	Kanal 1			
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Zeigt einen Kurzschluss am Ausgang DQn.A an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.A
			1	Kurzschluss an DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Zeigt einen Kurzschluss am Ausgang DQn.B oder die versuchte manuelle Einstellung von DQn.A und DQn.B mithilfe von SET_DQA, SET_DB und TM_CTRL_DQ an. Um diesen Fehler zurückzusetzen, müssen Sie RES_ERROR (Steuerschnittstelle) setzen.
			0	Kein Kurzschluss an DQn.B
			1	Kurzschluss an DQn.B oder die versuchte Einstellung von DQn.A und DQn.B
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Zeigt den Signalpegel am Digitalausgang DQn.A an.
			0	Signal 0 am Digitalausgang DQn.A
			1	Signal 1 am Digitalausgang DQn.A
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Zeigt den Signalpegel am Digitalausgang DQn.B an.
			0	Signal 0 am Digitalausgang DQn.B
			1	Signal 1 am Digitalausgang DQn.B

Zustand der DQ-Bits				
TM_CTRL_DQ	SET_DQA	SET_DQB	Reaktion an DQn.A	Reaktion an DQn.B
0	0	0	0	0
0	1	0	1	0

Zustand der DQ-Bits				
TM_CTRL_DQ	SET_DQA	SET_DQB	Reaktion an DQn.A	Reaktion an DQn.B
0	1	1	1	0 (ERR_DQB ist gesetzt)
0	0	1	0	1
1	Gleichgültig	Gleichgültig	Zustand durch Impulsverarbeitung gesteuert	Zustand durch Impulsverarbeitung gesteuert

Hinweis

Einstellung von DQn.A und DQn.B auf den nicht zulässigen Zustand 1

DQn.A und DQn.B dürfen nicht gleichzeitig auf 1 gesetzt werden. Wenn SET_DQA und SET_DQB auf High (Ein) und TM_CTRL_DQ auf Low (Aus) gesetzt sind, geht nur Ausgang DQn.A in den High-Zustand und das Fehlerbit ERR_DQB wird in der Rückmeldeschnittstelle gesetzt. Dieser Fehler muss in der Steuerschnittstelle mithilfe des RES_ERROR-Bits quittiert werden.

5.1.3 Grundlagen zur Impulsausgabe (Kompakt-CPU)

5.1.3.1 Betriebsarten

Übersicht

In diesem Kapitel sind die neuen Funktionen der CPU mit der Firmware-Version V2.0 aufgelistet.

Betriebsart	Anwendungen	Kundennutzen
Pulsweitenmodulation (PWM) (Seite 1300)	Die Betriebsart PWM setzen Sie dann ein, wenn ein Ausgabemodul bei geringer Verlustleistung (Erwärmung, Baugröße) möglichst große Leistungen steuern soll. Sie verwenden die Pulsweitenmodulation z. B. zur Steuerung: • der Temperatur in einem Heizwiderstand • der Kraft einer Spule in einem Proportionalventil und damit der Position eines Ventils von geschlossen bis vollständig geöffnet • der Drehzahl eines Motors vom Stillstand bis zur vollen Drehzahl	Bei der Pulsweitenmodulation wird ein Signal mit definierter Periodendauer und variabler Einschaltdauer am Digitalausgang ausgegeben. Die Einschaltdauer ist das Verhältnis von Impulsdauer zu Periodendauer. In der Betriebsart PWM können Sie neben der Einschaltdauer auch die Periodendauer steuern. Mit Pulsweitenmodulation variieren Sie den Mittelwert der Ausgangsspannung. Je nach angeschlossener Last können Sie damit den Laststrom oder die Leistung steuern.
Frequenzausgabe (Seite 1305)	Sie können Applikationen mit Frequenzen bis zu 100 kHz realisieren und damit in Bereichen arbeiten, die durch eine CPU mit einem einfachen Digitalausgang nicht erreicht werden.	Sie können Frequenzen sehr genau erzeugen. Der Empfänger kann bei nicht optimalen Übertragungsverhältnissen die Information genau rekonstruieren. In der Betriebsart Frequenzausgabe weisen Sie einen Frequenzwert mit hohen Frequenzen präziser zu als über die Periodendauer (PWM).
Pulse Train Output (PTO) (Seite 1307)	Pulse Train Output ist eine weit verbreitete Schnittstelle zur Antriebssteuerung. Sie wird in vielen Positionieranwendungen eingesetzt, wie z. B. für Verstell- oder Zuführachsen.	PTO wird auch als Puls-/Richtungs-Schnittstelle bezeichnet. Die Puls-/Richtungs-Schnittstelle besteht aus 2 Signalen. Dabei steht die Frequenz des Pulsausgangs für die Geschwindigkeit und die Anzahl der ausgegebenen Impulse für die zu fahrende Wegstrecke. Der Richtungsausgang legt die Verfahrrichtung fest. Die Position wird so inkrementgenau vorgegeben. Die Steuerung der Ausgänge erfolgt mit S7-1500 Motion Control über Technologieobjekte. PTO ist eine einfache und universelle Schnittstelle zwischen Steuerung und Antrieb. Sie wird deshalb weltweit von vielen Schritt- und Servo-Antrieben unterstützt.

Betriebsart: Pulsweitenmodulation (PWM)

Eigenschaften

Die Betriebsart Pulsweitenmodulation (PWM) der Kompakt-CPU hat folgende technische Eigenschaften:

	Minimum			Maximum		
	Standard-Ausgang	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert	Standard-Ausgang	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert
Impulsdauer	100 µs mit Last > 0,1A [1] 200 µs mit Last ≥ 2mA [1]	20 µs mit Last > 0,1 A [1] 40 µs mit Last ≥ 2mA [1]	2 µs [1]	10 000 000 µs (10 s)		
Periodendauer	10 ms [2]	100 µs [2]	10 µs			

[1] ein niedrigerer Wert ist theoretisch möglich, jedoch kann je nach angeschlossener Last die Ausgangsspannung nicht mehr als vollständiger Rechteckimpuls ausgegeben werden

[2] lastabhängig

Funktionsweise

Bei der Pulsweitenmodulation wird ein Signal mit definierter Periodendauer und variabler Einschaltdauer am Digitalausgang ausgegeben. Die Einschaltdauer ist das Verhältnis von Impulsdauer zu Periodendauer. In der Betriebsart PWM können Sie neben der Einschaltdauer auch die Periodendauer steuern.

Mit Pulsweitenmodulation variieren Sie den Mittelwert der Ausgangsspannung. Je nach angeschlossener Last können Sie damit den Laststrom oder die Leistung steuern.

Sie können die Impulsdauer als Hundertstel der Periodendauer (0 bis 100), als Tausendstel (0 bis 1 000), als Zehntausendstel (0 bis 10 000) oder im S7-Analogformat angeben.

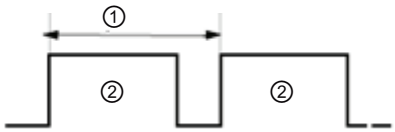

① Periodendauer
② Impulsdauer

Die Impulsdauer kann zwischen 0 (kein Impuls, immer Aus) und Vollausschlag (kein Impuls, Periodendauer immer Ein) liegen.

Der PWM-Ausgang kann z. B. zur Steuerung der Drehzahl eines Motors vom Stillstand bis zur vollen Drehzahl dienen oder Sie können ihn einsetzen, um die Position eines Ventils von geschlossen bis vollständig geöffnet zu steuern.

Zur Steuerung schneller Impulsausgänge (High-Speed-Ausgang) stehen vier Impulsgeneratoren jeweils für PWM und Impulsfolge (PTO) zur Verfügung. PTO wird von den Technologieobjekten Positionierachse, Drehzahlachse oder Gleichlaufachse genutzt. Sie

können jeden Impulsgenerator entweder PWM oder PTO zuordnen, jedoch nicht beiden gleichzeitig.

Sie konfigurieren die Betriebsart Pulsweitenmodulation (PWM) in STEP 7 (TIA Portal).

Die Betriebsart Pulsweitenmodulation besitzt die folgenden Funktionen:

- Wenn die Option "High-Speed-Ausgang (0,1 A)" aktiviert ist, können Sie eine Mindestimpulsdauer von 2 µs bei einem Strom von 100 mA erzeugen. Wenn die Option "High-Speed-Ausgang (0,1 A)" nicht aktiviert ist, können Sie eine Mindestimpulsdauer von 20 µs bei einer Last > 0,1 A und eine Mindestimpulsdauer von 40 µs bei einer Last von ≥ 2mA und einen Strom von maximal 0,5 A erzeugen. Bei Verwendung eines Standardausgangs können Sie eine Mindestimpulsdauer von 100 µs bei einer Last von > 0,1 A erzeugen und eine Mindestimpulsdauer von 200 µs bei einer Last von ≥ 2mA.
- Sie können den Impulsausgang (DQA) des Kanals manuell über die Steuer- und Rückmeldeschnittstelle steuern.
- Sie können die Reaktion auf CPU-STOP konfigurieren. Bei Wechsel nach CPU-STOP wird der Impulsausgang (DQA) in den konfigurierten Zustand versetzt.

Steuerung

Für die Betriebsart Pulsweitenmodulation (PWM) greift das Anwenderprogramm direkt auf die Steuer- und Rückmeldeschnittstelle des Kanals zu.

Ein Umparametrieren über die Anweisungen WRREC/RDREC und den Parametrierdatensatz 128 wird unterstützt. Weitere Informationen hierzu finden Sie im Kapitel Parameterdatensätze (PWM) des jeweiligen Gerätehandbuchs.

Sie steuern die Einschaltdauer der Impulsweite über das Feld OUTPUT_VALUE der Steuerschnittstelle. Die Pulsweitenmodulation erzeugt anhand dieses Wertes kontinuierliche Impulse. Die Periodendauer ist anpassbar.

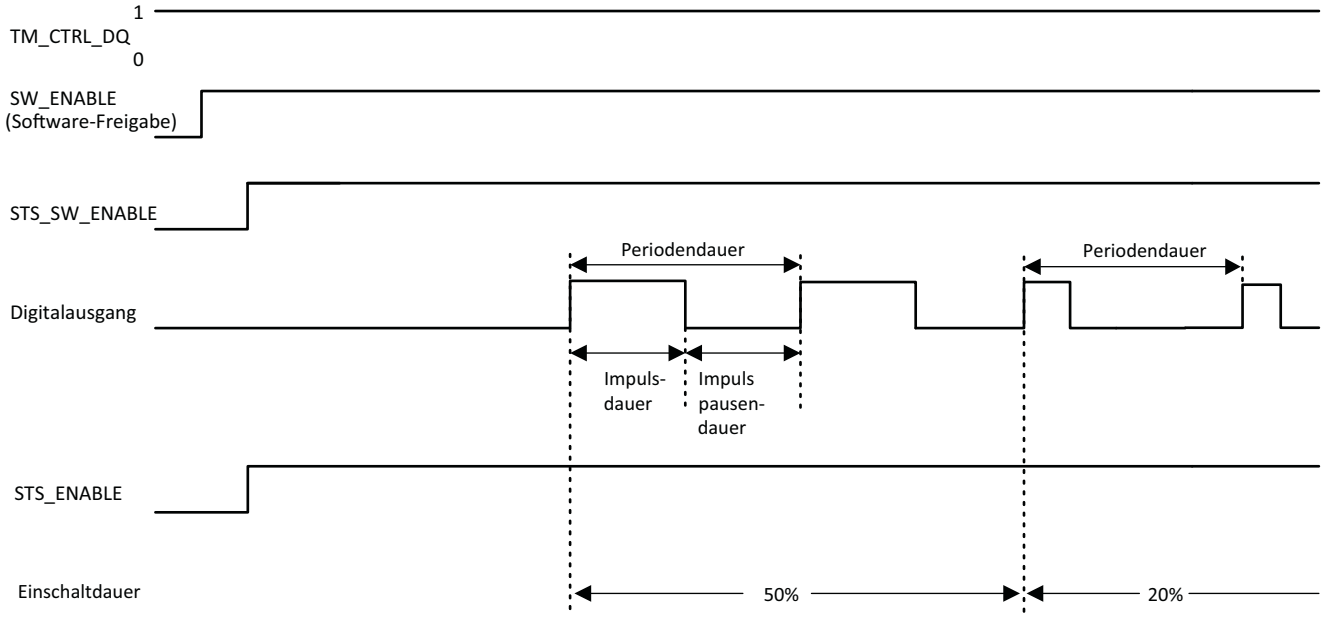

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mit Hilfe der Software-Freigabe ausgeben (SW_ENABLE 0 → 1). Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe an die PWM ansteht.

Wenn die Software-Freigabe aktiviert ist (steigende Flanke), wird STS_ENABLE gesetzt. Die Ausgabesequenz läuft kontinuierlich, solange SW_ENABLE gesetzt ist.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ

- Wenn TM_CTRL_DQ = 1, übernimmt die Technologiefunktion die Steuerung und erzeugt Impulssequenzen am Ausgang PWM DQA.
- Wenn TM_CTRL_DQ = 0 ist, übernimmt das Anwenderprogramm die Steuerung und der Anwender kann den Ausgang PWM DQA über das Steuerbit SET_DQA direkt einstellen.

Abbrechen der Ausgabesequenz

Eine Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) bricht die aktuelle Ausgabesequenz ab. Die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang PWM DQA werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Mindestimpulsdauer und Mindestimpulspause

Die Mindestimpulsdauer und die Mindestimpulspause weisen Sie mit Hilfe des Parameters "Mindestimpulsdauer" zu.

- Eine von der Technologiefunktion oder des PWM-Kanals ermittelte Impulsdauer, die kürzer als die Mindestimpulsdauer ist, wird unterdrückt.
- Eine von der Technologiefunktion oder des PWM-Kanals ermittelte Impulsdauer, die länger als die Periodendauer minus der Mindestimpulspause ist, wird auf den Wert der Periodendauer gesetzt (Ausgang dauerhaft eingeschaltet).

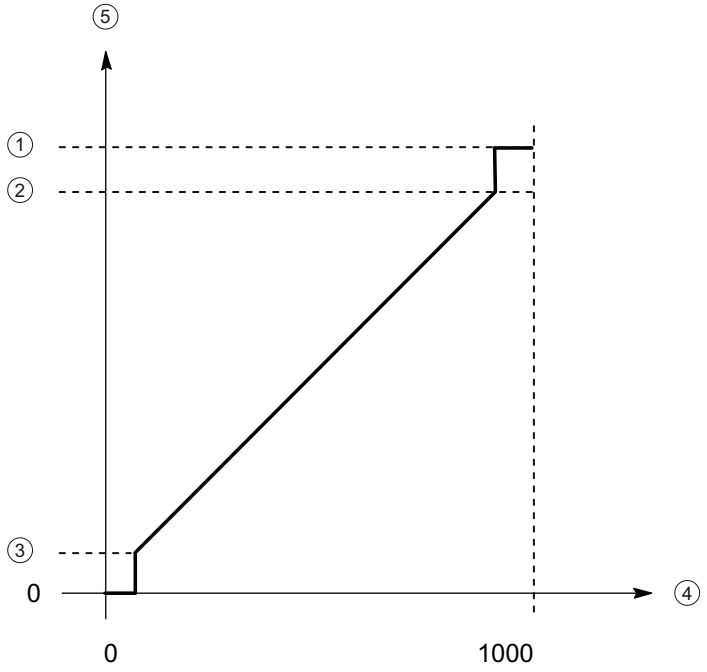

① Periodendauer
② Periodendauer minus Mindestimpulspause
③ Mindestimpulsdauer
④ OUTPUT_VALUE (Promille Einschaltdauer)
⑤ Impulsdauer

Einstellen und Ändern der Impulseinschaltdauer

OUTPUT_VALUE weist die Einschaltdauer für die aktuelle Periodendauer zu. Sie wählen den Bereich des Feldes OUTPUT_VALUE der Steuerschnittstelle mit dem Parameter "Ausgabeformat" aus.

- Ausgabeformat 1/100: Wertebereich zwischen 0 und 100
  Impulsdauer = (OUTPUT_VALUE/100) x Periodendauer.
- Ausgabeformat 1/1000: Wertebereich zwischen 0 und 1 000
  Impulsdauer = (OUTPUT_VALUE/1 000) x Periodendauer.

- Ausgabeformat 1/10000: Wertebereich zwischen 0 und 10 000
  Impulsdauer = (OUTPUT_VALUE/10 000) x Periodendauer.
- Ausgabeformat "S7-Analogausgabe": Wertebereich zwischen 0 und 27 648
  Impulsdauer = (OUTPUT_VALUE/27 648) x Periodendauer.

Sie weisen OUTPUT_VALUE direkt über das Steuerungsprogramm zu. Ein neuer OUTPUT_VALUE wird bei der nächsten steigenden Flanke am Ausgang übernommen.

Einstellen und Ändern der Periodendauer

- Permanente Aktualisierung
  Die Periodendauer wird über die Steuerschnittstelle permanent gesteuert. Das MODE_SLOT-Bit muss gesetzt werden ("1" bedeutet permanente Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben. Stellen Sie den Periodenwert im Feld SLOT ein. Die Einheit ist immer eine Mikrosekunde.
  - High-Speed-Ausgang aktiviert: zwischen 10 µs und 10 000 000 µs (10 s) im Feld SLOT
  - High-Speed-Ausgang deaktiviert: zwischen 100 µs und 10 000 000 µs (10 s) im Feld SLOT
  - Standard-Ausgang (100 Hz-Ausgang): zwischen 10 000 µs (10 ms) und 10 000 000 µs (10 s) im Feld SLOT
- Einzelne Aktualisierung
  Stellen Sie die Periodendauer in den Konfigurationsparametern ein. Alternativ führen Sie eine einzelne Aktualisierung über die Steuerschnittstelle aus. MODE_SLOT muss gelöscht werden ("0" bedeutet einzelne Aktualisierung); LD_SLOT muss den Wert 1 ("1" bedeutet Periodendauer) haben. Stellen Sie den Wert der Periodendauer im Feld SLOT ein. Die Einheit ist immer eine Mikrosekunde.
  - High-Speed-Ausgang aktiviert: zwischen 10 µs und 10 000 000 µs (10 s) in den Parametern
  - High-Speed-Ausgang deaktiviert: zwischen 100 µs und 10 000 000 µs (10 s) in den Parametern
  - Standard-Ausgang (100 Hz-Ausgang): zwischen 10 000 µs (10 ms) und 10 000 000 µs (10 s) in den Parametern

Die neue Periodendauer wird bei der nächsten steigenden Flanke des Ausgangs übernommen.

Einstellen der Mindestimpulsdauer und der Mindestimpulspause

Sie weisen die Mindestimpulsdauer und die Mindestimpulspause als DWord-Zahlenwert zwischen 0 und 10 000 000 µs (10 s) mit Hilfe der Kanalparameterkonfiguration "Mindestimpulsdauer" zu.

Ausgangssignale für Betriebsart Pulsweitenmodulation (PWM)

Ausgangssignal	Bedeutung	Wertebereich
Kontinuierlicher Impulsstrom am Digitalausgang PWM DQA	Ein Impuls wird am Digitalausgang PWM DQA für die eingestellte Einschaltdauer und Periodendauer ausgegeben.	kontinuierlicher Impulsstrom

Betriebsart: Frequenzausgabe

In dieser Betriebsart können Sie einen Frequenzwert mit hohen Frequenzen präziser zuweisen als über die Periodendauer im PWM-Betrieb.

Ein Rechtecksignal mit einer zugewiesenen Frequenz und einer konstanten Einschaltdauer von 50 % wird am Digitalausgang erzeugt.

Die Betriebsart Frequenzausgabe besitzt die folgenden Funktionen:

- Wenn die Option "High-Speed-Ausgang (0,1 A)" aktiviert ist, können Sie eine Mindestimpulsdauer von 2 µs bei einem Strom von 100 mA erzeugen. Wenn die Option "High-Speed-Ausgang (0,1 A)" nicht aktiviert ist, können Sie eine Mindestimpulsdauer von 20 µs bei einer Last von > 0,1 A und eine Mindestimpulsdauer von 40 µs bei einer Last von ≥ 2mA und einen Strom von maximal 0,5 A erzeugen.
  Wenn Sie einen Standard-Ausgang verwenden, können Sie eine Mindestimpulsdauer von 100 µs bei einer Last von > 0,1 A und eine Mindestimpulsdauer von 200 µs bei einer Last von ≥ 2mA und einen Strom von maximal 0,5 A erzeugen.

	Minimum			Maximum		
	Standard-Ausgang	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert	Standard-Ausgang	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert
Frequenz	0,1 Hz			100 Hz [1]	10 kHz [1]	100 kHz

[1] lastabhängig

- Sie können den Impulsausgang (DQA) des Kanals manuell über die Steuer- und Rückmeldeschnittstelle steuern.
- Sie können die Reaktion auf CPU-STOP konfigurieren. Bei Wechsel nach CPU-STOP wird der Impulsausgang (DQA) in den konfigurierten Zustand versetzt.

Steuerung

Für die Betriebsart Frequenzausgabe greift das Anwenderprogramm direkt auf die Steuer- und Rückmeldeschnittstelle des Kanals zu.

Ein Umparametrieren über die Anweisungen WRREC/RDREC und den Parametrierdatensatz 128 wird unterstützt. Weitere Informationen hierzu finden Sie im Gerätehandbuch zur jeweiligen Kompakt-CPU im Kapitel "Parameterdatensätze (PWM)".

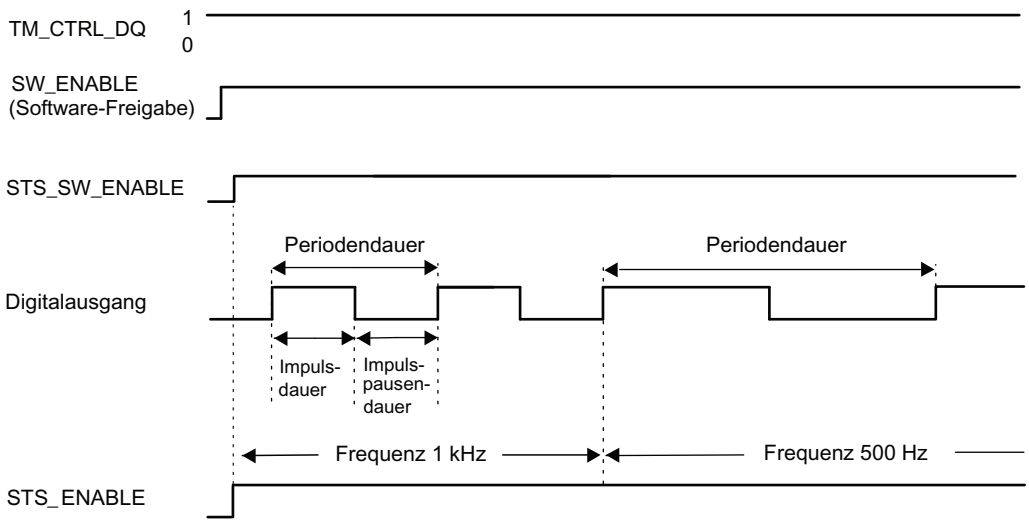

Starten der Ausgabesequenz

Das Steuerungsprogramm muss die Freigabe für die Ausgabesequenz mit Hilfe der Software-Freigabe veranlassen (SW_ENABLE 0 → 1.). Das Rückmeldebit STS_SW_ENABLE zeigt an, dass die Software-Freigabe am Impulsgenerator ansteht.

Wenn die Software-Freigabe aktiviert ist (steigende Flanke), wird STS_ENABLE gesetzt. Die Ausgabesequenz läuft kontinuierlich, solange SW_ENABLE gesetzt ist.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ

- Wenn TM_CTRL_DQ = 1, übernimmt die Technologiefunktion die Steuerung und erzeugt Impulssequenzen am Ausgang PWM DQA.
- Wenn TM_CTRL_DQ = 0 ist, übernimmt das Anwenderprogramm die Steuerung und der Anwender kann den Ausgang PWM DQA über das Steuerbit SET_DQA direkt einstellen.

Abbrechen der Ausgabesequenz

Eine Deaktivierung der Software-Freigabe (SW_ENABLE = 1 → 0) während der Frequenzausgabe bricht die aktuelle Ausgabesequenz ab. Die letzte Periodendauer wird nicht abgeschlossen. STS_ENABLE und der Digitalausgang PWM DQA werden sofort auf 0 zurückgesetzt.

Eine erneute Impulsausgabe ist erst nach einem Neustart der Ausgabesequenz möglich.

Einstellen und Ändern des Ausgabewerts (Frequenz)

Sie stellen die Frequenz mit dem OUTPUT_VALUE direkt mit dem Steuerungsprogramm in der Steuerschnittstelle ein. Der Wert wird im Real-Format angegeben und die Einheit ist immer "Hz". Der mögliche Bereich hängt vom Parameter "High-Speed-Ausgang (0,1 A)" wie folgt ab:

- Schnelle Impulsausgabe deaktiviert
  - Frequenz (OUTPUT_VALUE): 0,1 Hz bis 10 000 Hz
- Schnelle Impulsausgabe aktiviert
  - Frequenz (OUTPUT_VALUE): 0,1 Hz bis 100 000 Hz
- Standard-Ausgang (100 Hz-Ausgang)
  - Frequenz (OUTPUT_VALUE): 0,1 Hz bis 100 Hz

Die neue Frequenz wird zu Beginn der nächsten Periode übernommen. Die neue Frequenz hat keine Auswirkung auf die fallende Flanke oder das Impuls-Periode-Verhältnis (Einschaltdauer). Allerdings kann die Übernahme in Abhängigkeit von der zuvor eingestellten Frequenz bis zu 10 s betragen.

Genauigkeit der Ausgabefrequenz

Die konfigurierte Ausgabefrequenz wird mit einer frequenzabhängigen Genauigkeit am Digitalausgang PWM DQA ausgegeben. Eine Übersicht über die Genauigkeit in Abhängigkeit von der verwendeten Frequenz finden Sie im Gerätehandbuch zur jeweiligen Kompakt-CPU im Kapitel "Verschaltungsübersicht der Ausgänge".

Ausgangssignale für Betriebsart Frequenzausgabe

Ausgangssignal	Bedeutung	Wertebereich
Kontinuierlicher Impulsstrom am Digitalausgang PWM DQA	Ein Impuls wird am Digitalausgang PWM DQA für die zugewiesene Frequenz ausgegeben.	kontinuierlicher Impulsstrom

Betriebsart: PTO

Steuerung

Für die vier Betriebsarten der Impulsgeneratoren (PTO) erfolgt die Steuerung der Impulsausgabekanäle mittels Motion Control über die Technologieobjekte TO_SpeedAxis, TO_PositioningAxis und TO_SynchronousAxis. Die Steuer- und Rückmeldeschnittstelle der Kanäle ist bei diesen Betriebsarten eine partielle Umsetzung der PROFIdrive-Schnittstelle mit dem Standardtelegramm 3.

Eine ausführliche Beschreibung des Einsatzes von Motion Control und dessen Projektierung finden Sie in der Onlinehilfe zu Motion Control (Seite 1835) sowie im Funktionshandbuch S7-1500 Motion Control (http://support.automation.siemens.com/WW/view/de/59381279).

Signalarten

Die Betriebsart PTO unterteilt sich in die folgenden vier Signalarten:

- PTO (Impuls (A) und Richtung (B)):
  Ein Ausgang (A) steuert die Impulse und ein Ausgang (B) steuert die Richtung. B ist 'High' (aktiv), wenn Impulse in negativer Richtung erzeugt werden. B ist 'Low' (inaktiv), wenn Impulse in positiver Richtung erzeugt werden.

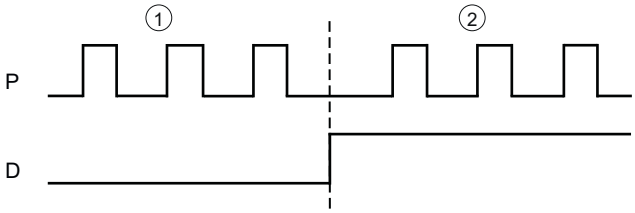

① Positive Drehrichtung
② Negative Drehrichtung

- PTO (Vorwärtszählen (A) und rückwärtszählen (B)):
  Ein Ausgang (A) gibt Impulse für positive Richtungen und ein anderer Ausgang (B) Impulse für negative Richtungen aus.

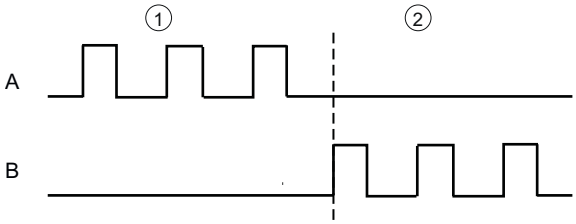

① Positive Drehrichtung
② Negative Drehrichtung

- PTO (A, B phasenversetzt):
  Beide Ausgänge geben Impulse mit der angegebenen Geschwindigkeit aus, jedoch um 90 Grad phasenversetzt. Es handelt sich um eine Einfach-Impulsausgabe, bei der ein Impuls die Zeitdauer zwischen zwei Übergängen von Signal A während eines Low-Pegels von Signal B aufweist.
  Eine positive Drehrichtung wird bei steigender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt. Eine negative Drehrichtung wird bei fallender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt.

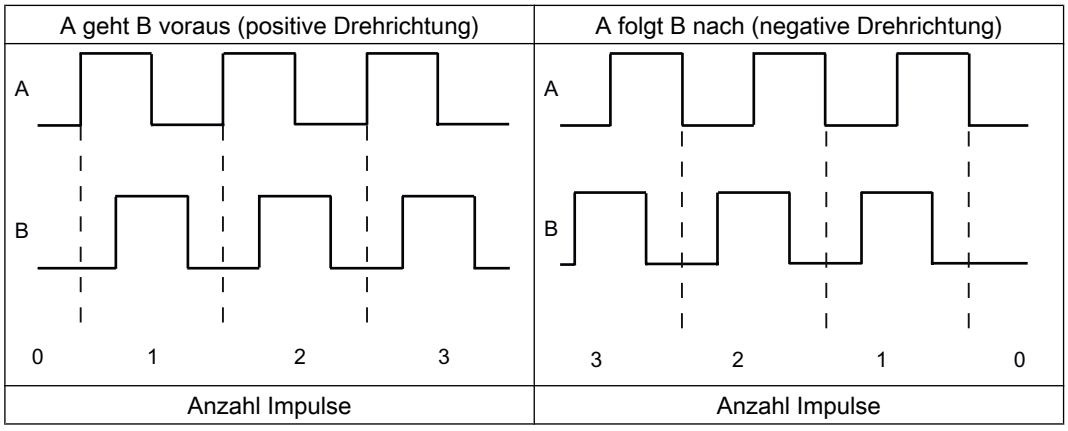

- PTO (A, B phasenversetzt, vierfach):
  Beide Ausgänge geben Impulse mit der angegebenen Geschwindigkeit aus, jedoch um 90 Grad phasenversetzt. Bei dieser Signalart handelt es sich um eine Vierfach-Impulsausgabe, bei der jeder Flankenübergang einem Inkrement entspricht. Eine vollständige Periode des Signals A enthält also vier Inkremente. Auf diese Weise lässt sich z. B. mit zwei Ausgängen mit jeweils 100 kHz Signalfrequenz ein Steuersignal ausgeben, das 400 000 Inkremente pro Sekunde liefert.
  Ob Zählimpulse in positiver oder negativer Drehrichtung erzeugt werden, hängt von der Flankenrichtung des einen Signals und dem Pegel des jeweils anderen Signals währenddessen ab.

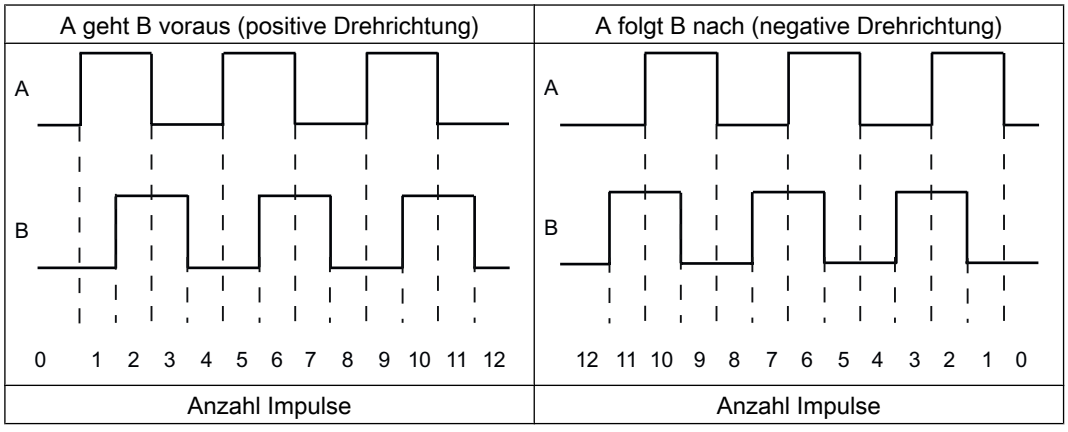

Verhalten des PTO-Kanals bei CPU-STOP

Auf einen Wechsel nach CPU-STOP reagiert der PTO-Kanal mit Wegnahme der Antriebsfreigabe (sofern ein Antriebsfreigabe-Ausgang konfiguriert ist) und mit Ausgabe des Geschwindigkeitssollwerts 0 an den für die Signalspuren A und B konfigurierten Hardware-Ausgängen. Das CPU-STOP-Verhalten der PTO-Kanäle ist nicht konfigurierbar.

Hinweis

Verhalten bei CPU-STOP

Bei CPU-STOP können die für die PTO-Ausgänge A und B zugewiesenen Hardware-Ausgänge in den Signalzustand 'High' (1) schalten und/oder dort verbleiben. Ein Schalten/Verbleiben der beiden Hardware-Ausgänge nach/in Signalpegel 'Low' (0) ist nicht garantiert.

5.1.3.2 Funktionen

Funktion: High-Speed-Ausgang

Die Funktion "High-Speed-Ausgang (0,1 A)" verbessert den Signaltakt der Digitalausgänge (DQ0 bis DQ7). An den Schaltflanken treten weniger Verzögerung, Schwankungen, Jitter sowie kürzere Anstiegs-/Fallzeiten auf.

Die Funktion "High-Speed-Ausgang (0,1 A)" eignet sich dazu, Impulssignale in einem präziseren Takt zu erzeugen, bietet jedoch einen geringeren maximalen Laststrom.

Für die Betriebsarten PWM und Frequenzausgabe wählen Sie in STEP 7 (TIA Portal) den High-Speed-Ausgang des Kanals aus. Außerdem können Sie die Parametrierung zur Laufzeit mit Hilfe des Programms über den Datensatz ändern.

Die schnelle Impulsausgabe (High-Speed-Ausgang) ist für folgende Betriebsarten verfügbar:

- PWM
- Frequenzausgabe
- PTO (die Impulsausgänge für die Betriebsart PTO sind immer "High-Speed-Ausgang (0,1 A)")

High-Speed-Ausgang

	Minimum		Maximum	
	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert	High-Speed-Ausgang deaktiviert	High-Speed-Ausgang aktiviert
Impulsdauer	20 µs mit Last > 0,1 A [1] 40 µs mit Last ≥ 2 mA [1]	2 µs [1]	10 000 000 µs (10 s)	
Periodendauer	100 µs [2]	10 µs		
Frequenz	0,1 Hz		10 kHz [2]	100 kHz

[1] ein niedrigerer Wert ist theoretisch möglich, jedoch kann je nach angeschlossener Last die Ausgangsspannung nicht mehr als vollständiger Rechteckimpuls ausgegeben werden

[2] lastabhängig

Funktion: Direktansteuerung des Impulsausgangs (DQA)

Direktansteuerung des Impulsausgangs (DQA)

In den Betriebsarten "Pulsweitenmodulation PWM" und "Frequenzausgabe" können Sie den Impulsausgangs (DQA) eines Impulsgenerators direkt über das Steuerungsprogramm einstellen. Wählen Sie die Funktion für die DQ-Direktansteuerung, indem Sie das Ausgangssteuerbit des PWM-Kanals in der Steuerschnittstelle rücksetzen (TM_CTRL_DQ = 0).

Die Direktansteuerung des Impulsausgangs (DQA) kann bei der Inbetriebnahme eines Steuerungssystems für die Automatisierung hilfreich sein.

Wenn Sie die Direktansteuerung des Impulsausgangs (DQA) während einer Impulsausgabesequenz auswählen, läuft die Sequenz im Hintergrund weiter, so dass die Ausgabesequenz fortgesetzt wird, sobald der Kanal die Steuerung wieder übernimmt (durch Einstellen von TM_CTRL_DQ = 1).

Sie weisen den Zustand des Impulsausgangs (DQA) mit dem Steuerbit SET_DQA zu.

Wenn Sie TM_CTRL_DQ = 1 setzen, wählen Sie die Direktansteuerung des Impulsausgangs (DQA) ab und der Kanal übernimmt die Verarbeitung. Wenn die Ausgabesequenz noch läuft (STS_ENABLE noch aktiv), dann übernimmt der PWM-Kanal erneut die Ansteuerung des Ausgangs. Wenn TM_CTRL_DQ = 1 ist und STS_ENABLE nicht aktiv ist, übernimmt ebenfalls der Kanal des Moduls die Verarbeitung, aber gibt am Ausgang dann "0" aus.

Hinweis

Ausgangssteuersignal TM_CTRL_DQ des PWM-Kanals

- Wenn TM_CTRL_DQ = 1, übernimmt die Technologiefunktion die Steuerung und erzeugt Impulssequenzen am Ausgang PWM DQA.
- Wenn TM_CTRL_DQ = 0, dann übernimmt das Anwenderprogramm die Steuerung und der Anwender kann den PWM DQA direkt mit dem Steuerbit SET_DQA einstellen.

5.1.4 Grundlagen zur Impulsausgabe (TM PTO 4)

5.1.4.1 Übersicht

Das Technologiemodul bietet folgende Funktionen:

Betriebsart	Anwendungen	Kundennutzen
Pulse Train Output (PTO) (Seite 1312)	Pulse Train Output ist eine weit verbreitete Schnittstelle zur Antriebssteuerung. Sie wird in vielen Positionieranwendungen eingesetzt, wie z. B. für Verstell- oder Zuführachsen.	PTO wird auch als Puls-/Richtungs-Schnittstelle bezeichnet. Die Puls-/Richtungs-Schnittstelle besteht aus 2 Signalen. Dabei steht die Frequenz des Pulsausgangs für die Geschwindigkeit und die Anzahl der ausgegebenen Impulse für die zu fahrende Wegstrecke. Der Richtungsausgang legt die Verfahrrichtung fest. Die Position wird so inkrementgenau vorgegeben. Die Steuerung der Ausgänge erfolgt mit S7-1500 Motion Control über Technologieobjekte. PTO ist eine einfache und universelle Schnittstelle zwischen Steuerung und Antrieb. Sie wird deshalb weltweit von vielen Schritt- und Servo-Antrieben unterstützt.

5.1.4.2 Pulse Train Output (PTO)

Steuerung

Die Steuerung der Impulsausgabekanäle ist vor allem mit S7-1500 Motion Control über die Technologieobjekte TO_SpeedAxis, TO_PositioningAxis und TO_SynchronousAxis vorgesehen. Die Steuer- und Rückmeldeschnittstelle der Kanäle ist eine Umsetzung der PROFIdrive-Schnittstelle mit dem Standardtelegramm 3.

Eine ausführliche Beschreibung der Projektierung des Technologiemoduls mit den Achs-Technologieobjekten finden Sie im Funktionshandbuch S7-1500T Motion Control im Kapitel Konfigurieren als Download im Internet (https://support.industry.siemens.com/cs/ww/de/view/109481326).

Signalarten

Das Technologiemodul unterstützt die folgenden vier Signalarten:

- Impuls (P) und Richtung (D):
  Ein Ausgang (P) steuert die Impulse und ein Ausgang (D) steuert die Richtung. D ist 'High' (aktiv), wenn Impulse in negativer Richtung erzeugt werden. D ist 'Low' (inaktiv), wenn Impulse in positiver Richtung erzeugt werden.

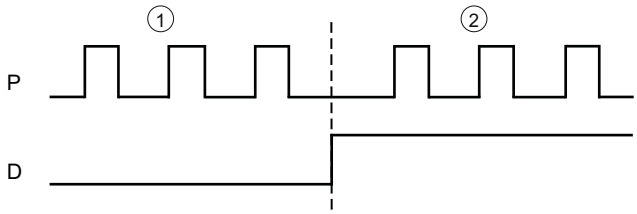

① Positive Drehrichtung
② Negative Drehrichtung

- Vorwärts zählen (A), rückwärts zählen (B):
  Ein Ausgang (A) gibt Impulse für positive Richtungen und ein anderer Ausgang (B) Impulse für negative Richtungen aus.

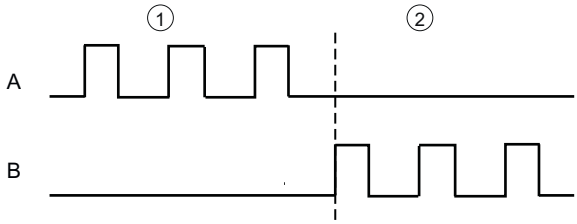

① Positive Drehrichtung
② Negative Drehrichtung

- Inkrementalgeber (A, B phasenversetzt):
  Beide Ausgänge geben Impulse mit der angegebenen Geschwindigkeit aus, jedoch um 90 Grad phasenversetzt. Es handelt sich um eine Einfach-Impulsausgabe, bei der ein Impuls die Zeitdauer zwischen zwei Übergängen von Signal A während eines Low-Pegels von Signal B aufweist.
  Eine positive Drehrichtung wird bei steigender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt. Eine negative Drehrichtung wird bei fallender Flanke an Signal A während eines Low-Pegels an Signal B erzeugt.

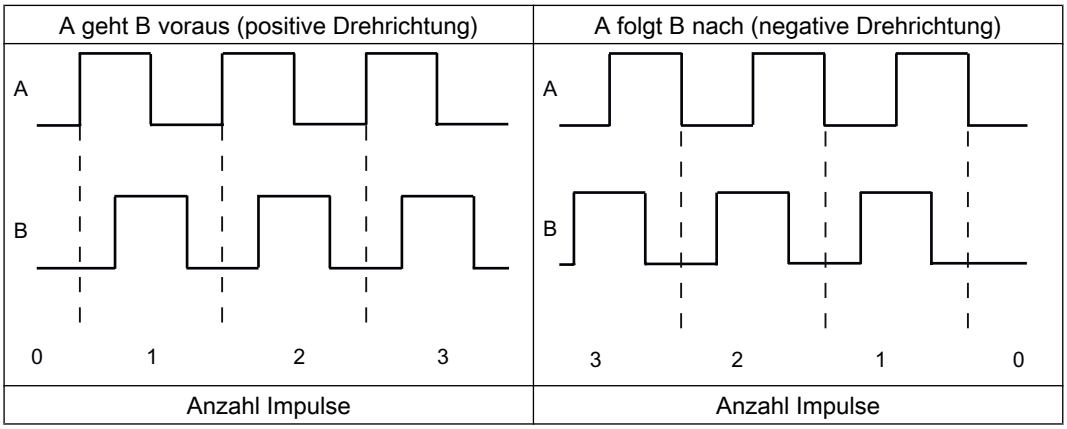

- Inkrementalgeber (A, B phasenversetzt, vierfach):
  Beide Ausgänge geben Impulse mit der angegebenen Geschwindigkeit aus, jedoch um 90 Grad phasenversetzt. Bei dieser Signalart handelt es sich um eine Vierfach-Impulsausgabe, bei der jeder Flankenübergang einem Inkrement entspricht. Eine vollständige Periode des Signals A enthält also vier Inkremente. Auf diese Weise lässt sich z. B. mit zwei Ausgängen mit jeweils 100 kHz Signalfrequenz ein Steuersignal ausgeben, das 400 000 Inkremente pro Sekunde liefert.
  Ob Zählimpulse in positiver oder negativer Drehrichtung erzeugt werden, hängt von der Flankenrichtung des einen Signals und dem Pegel des jeweils anderen Signals währenddessen ab.

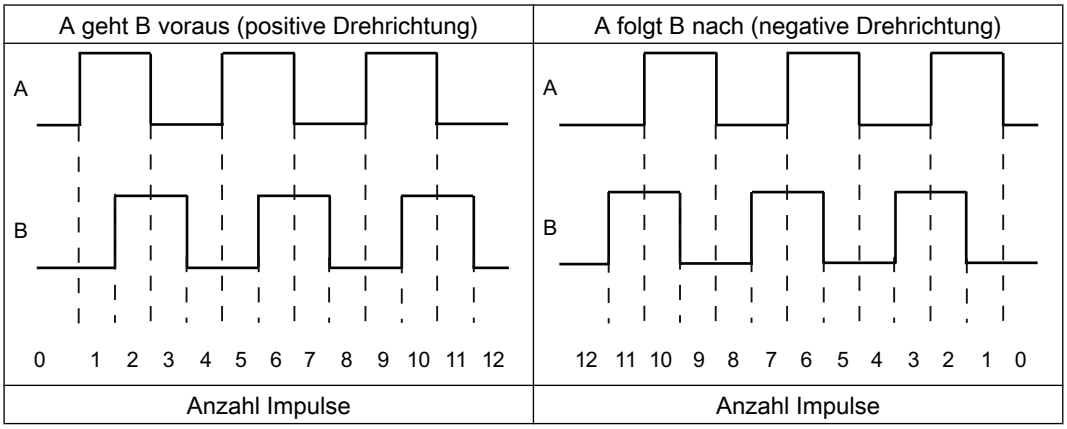

5.2 Modul einsetzen

5.2.1 TM Pulse 2x24V einsetzen

5.2.1.1 Technologiemodul konfigurieren und parametrieren

Technologiemodul zur Hardware-Konfiguration hinzufügen

Voraussetzungen

- Das Projekt ist angelegt.
- Die CPU ist angelegt.
- Dezentrale Peripherie ET 200 ist angelegt.

Vorgehen

Um das Technologiemodul zur Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie das Technologiemodul aus dem Baugruppenkatalog aus:
   "Technologiemodule > Pulsausgabe > TM Pulse 2x24V > Artikelnummer"
4. Ziehen Sie das Technologiemodul per Drag & Drop auf den gewünschten Steckplatz im Baugruppenträger.

Parametereinstellung (Hardware-Konfiguration) öffnen

Öffnen über Gerätesicht

Gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie auf das Modul.

Kanalkonfiguration

Kanalkonfiguration

Mit diesem Parameter legen Sie fest, ob die beiden Kanäle parallel geschaltet werden sollen.

Kanalkonfiguration	Bedeutung
2 Kanäle (2 A) (voreingestellt)	Die beiden Kanäle arbeiten separat und haben je einem maximalen Ausgangsstrom von 2 A.
1 Kanal (4 A)	Die beiden Kanäle sind parallelgeschaltet und arbeiten als ein logischer Kanal mit einem maximalen Ausgangsstrom von 4 A. Alle Parametrierungen und Vorgänge, die die Steuerschnittstelle nutzen, müssen Adressen mit Kanal 0 verwenden. Rückmeldesignale werden ausschließlich für Kanal 0 bereitgestellt. Die Steuer- und Rückmeldeschnittstelle beschränkt sich auf nur einen Kanal. Kanal 1 kann nicht konfiguriert oder betrieben werden. Die Strommessung addiert die gemessenen Werte beider Kanäle.

Hinweis

High-Speed-Ausgang

Den Parameter "High-Speed-Ausgang (0.1 A)" können Sie nur bei der Kanalkonfiguration "2 Kanäle (2 A)" wählen.

Betriebsart

Betriebsart

Mit diesem Parameter legen Sie fest, für welche Funktion der Kanal des Technologiemoduls eingesetzt wird. Abhängig davon werden die Einstellmöglichkeiten unter "Parameter" angepasst.

Betriebsart	Bedeutung	Betriebsartabhängige Parameter (Seite 1320)
Impulsausgabe	Der Kanal gibt einen Einzelimpuls mit der parametrierten Einschaltverzögerung aus. Die Impulsdauer geben Sie in der Steuerschnittstelle vor.	• High-Speed-Ausgang (0.1 A) • Funktion DI • Eingangsverzögerung • Ausgabeformat • Einschaltverzögerung
Pulsweitenmodulation PWM (voreingestellt)	Der Kanal gibt ein pulsweitenmoduliertes Signal mit der parametrierten Mindestimpulsdauer und der parametrierten Periodendauer aus. Die Einschaltdauer bzw. den Stromsollwert geben Sie in der Steuerschnittstelle vor.	• High-Speed-Ausgang (0.1 A) • Funktion DI[1] • Eingangsverzögerung • Ausgabeformat • Mindestimpulsdauer • Periodendauer • Einschaltverzögerung[1] • Dithering • Steigung Dither-Einschaltrampe • Steigung Dither-Ausschaltrampe • Dither-Amplitude • Dither-Periodendauer • Stromgeregelt • P-Anteil einschalten • I-Anteil einschalten • D-Anteil einschalten • Strombezugswert • Totzonenbreite • Stellwert obere Begrenzung • Stellwert untere Begrenzung • Proportionalbeiwert (Verstärkung) • Integrationszeit • Differenzierzeit • Verzögerungszeit des D-Anteils

Betriebsart	Bedeutung	Betriebsartabhängige Parameter (Seite 1320)
Impulskette	Der Kanal gibt eine Anzahl von Impulsen mit vorgegebener Periodendauer, Einschaltdauer und Einschaltverzögerung aus. Die Impulsanzahl geben Sie in der Steuerschnittstelle vor.	• High-Speed-Ausgang (0.1 A) • Funktion DI • Eingangsverzögerung • Ausgabeformat • Periodendauer • Einschaltverzögerung • Einschaltdauer
Ein-/Ausschaltverzögerung	Der Kanal gibt Impulse mit vorgegebener Schaltverzögerung bezogen auf DI0 aus. Die Einschaltverzögerung geben Sie in der Steuerschnittstelle vor.	• High-Speed-Ausgang (0.1 A) • Eingangsverzögerung • Ausgabeformat • Ausschaltverzögerung
Frequenzausgabe	Der Kanal gibt ein Signal mit einer festen Einschaltdauer von 50 % aus. Die Frequenz geben Sie in der Steuerschnittstelle vor.	• High-Speed-Ausgang (0.1 A) • Funktion DI • Eingangsverzögerung • Ausgabeformat • Einschaltverzögerung
PWM mit DC-Motor	Der Kanal gibt ein pulsweitenmoduliertes Signal aus zur Ansteuerung eines DC-Motors in beiden Drehrichtungen. Die Einschaltdauer geben Sie in der Steuerschnittstelle vor.	• Funktion DI[1] • Eingangsverzögerung • Ausgabeformat • Periodendauer • Einschaltverzögerung[1]

[1] Nicht im taktsynchronen Betrieb parametrierbar

Verhalten bei CPU-STOP

Verhalten bei CPU-STOP

Mit diesem Parameter legen Sie die Reaktion des Kanals auf den Ausfall einer überlagerten Steuerung fest.

Verhalten bei CPU-STOP	Bedeutung
Ersatzwert ausgeben (voreingestellt)	Der Kanal gibt bis zum nächsten STOP-RUN-Übergang der CPU an den Digitalausgängen die parametrierten Ersatzwerte aus. Das Rückmeldebit STS_ENABLE wird auf 0 gesetzt. Nach einem STOP-RUN-Übergang wird das Modul in seinen Anlaufzustand gesetzt. Bevor Sie eine neue Impulsausgabe starten können, müssen Sie das Steuerbit SW_ENABLE auf 0 setzen.
Weiterarbeiten	Der Kanal arbeitet weiter. Die Digitalausgänge schalten weiterhin entsprechend der Parametrierung.

Ersatzwert für DQA

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem CPU-STOP am Digitalausgang DQA ausgeben soll.

Die Voreinstellung ist "0".

Ersatzwert für DQB

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert das Technologiemodul bei einem CPU-STOP am Digitalausgang DQB ausgeben soll.

Die Voreinstellung ist "0".

Hinweis

Sie können nur für einen der beiden Digitalausgänge den Ersatzwert 1 auswählen.

Diagnose

Das Technologiemodul kann zusätzliche Diagnosealarme auslösen, wenn Sie die Freigabe der Diagnosealarme bei den Grundparametern aktivieren. Diagnosealarme bearbeiten Sie in einem Alarm-OB.

Sammeldiagnose

Mit diesem Parameter legen Sie fest, ob bei den folgenden Fehlern ein Diagnosealarm ausgelöst wird:

- Versorgungsspannungsfehler (Rückmeldebit ERR_PWR)
- Kurzschluss oder Überlast der Geberversorgung (Rückmeldebit ERR_24V)
- Übertemperatur

In der Voreinstellung ist der Parameter deaktiviert.

Diagnose DQA

Mit diesem Parameter legen Sie fest, ob bei einem Fehler am Digitalausgang DQA der Diagnosealarm "Fehler an Digitalausgängen" ausgelöst wird. Unabhängig davon wird bei einem Fehler das Rückmeldebit ERR_DQA gesetzt.

In der Voreinstellung ist der Parameter deaktiviert.

Diagnose DQB

Mit diesem Parameter legen Sie fest, ob bei einem Fehler am Digitalausgang DQB der Diagnosealarm "Fehler an Digitalausgängen" ausgelöst wird. Unabhängig davon wird bei einem Fehler das Rückmeldebit ERR_DQB gesetzt.

In der Voreinstellung ist der Parameter deaktiviert.

Parameter

High-Speed-Ausgang (0.1 A)

Mit diesem Parameter legen Sie fest, ob der jeweilige Digitalausgang als schneller Gegentakt-Schalter arbeitet:

Option	Bedeutung
Aktiviert	Die Digitalausgänge arbeiten als schnelle Gegentakt-Schalter und sind mit einem Nennlaststrom von je 0,1 A belastbar. Ein Gegentakt-Schalter wird wechselweise nach DC 24 V und Masse geschaltet. Dadurch sind sehr schnelle Flanken möglich.
Deaktiviert (voreingestellt)	Die Digitalausgänge arbeiten als Gegentakt-Schalter und sind mit einem Nennlaststrom von je 2 A belastbar.

Hinweis

Kanalkonfiguration

Sie können diesen Parameter nur bei der Kanalkonfiguration "2 Kanäle (2 A)" wählen.

Hinweis

Sie können diesen Parameter nicht aktivieren in den Betriebsarten "PWM mit DC-Motor" und "Pulsweitenmodulation PWM" mit Stromregelung.

Funktion DI

Mit der Parametrierung eines Digitaleingangs legen Sie fest, welche Funktion der Digitaleingang beim Schalten auslöst.

Sie können unter folgenden Optionen auswählen:

Funktion	Bedeutung
Eingang (voreingestellt)	Dem jeweiligen Digitaleingang ist keine Funktion zugeordnet. Der Signalzustand des Digitaleingangs kann von der CPU über die Rückmeldeschnittstelle gelesen werden.
HW-Freigabe	Der jeweilige Digitaleingang dient als Freigabe-Eingang für die Ausgabesequenz.
Endschalter	Der jeweilige Digitaleingang dient als externer Endschalter für einen DC-Motor in der Betriebsart "PWM mit DC-Motor".

Eingangsverzögerung

Mit der Parametrierung der Eingangsverzögerung unterdrücken Sie Störungen an den Digitaleingängen. Signale mit einer Impulsdauer unterhalb der parametrierten Eingangsverzögerung werden unterdrückt.

Sie können unter folgenden Eingangsverzögerungen auswählen:

- Keine
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Ausgabeformat

Mit diesem Parameter legen Sie das Format bzw. den Wertebereich fest, mit dem die Einschaltdauer oder der Stromsollwert ausgewertet wird:

Ausgabeformat	Wertebereich
S7-Analogausgabe	Betriebsart "PWM mit DC-Motor": –27.648 ... 27.648 Andere Betriebsart: 0 ... 27648 ("27648" entspricht 100 % des Bezugswerts)
1/100 (voreingestellt)	0 ... 100
1/1000	0 ... 1000
1/10000	0 ... 10000

Hinweis

In der Betriebsart "Frequenzausgabe" ist als Ausgabeformat "1 Hz" ausgewählt und nicht änderbar.

In der Betriebsart "PWM mit DC-Motor" ist als Ausgabeformat "S7-Analogausgabe" ausgewählt und nicht änderbar.

Mindestimpulsdauer

Mit diesem Parameter legen Sie die minimal zulässige Impulsdauer für das PWM-Signal fest.

Impulse und Pausen, die den Wert unterschreiten, werden unterdrückt.

Die Voreinstellung ist "0 µs".

Hinweis

Die Mindestimpulsdauer ist nicht wirksam, falls Sie "Stromgeregelt" aktiviert haben.

Periodendauer

Mit diesem Parameter legen Sie die Periodendauer des Ausgabesignals fest.

Die Voreinstellung ist "2000000 µs". Das entspricht 2 s.

Tatsächliche Periodendauer

Die tatsächliche Periodendauer wird nur im taktsynchronen Betrieb angezeigt.

Der angezeigte Periodendauerwert ist die eigentliche mit der Taktsynchronität kompatible Periodendauer, die aus dem Wert berechnet wird, den Sie für die Periodendauer eingegeben haben.

Einschaltverzögerung

Mit diesem Parameter legen Sie die Verzögerung fest, mit welcher der Start der Ausgabesequenz die Ausgabe des Impulses bewirkt.

Die Voreinstellung ist "0 µs".

Hinweis

Taktsynchroner Betrieb

Im taktsynchronen Betrieb können Sie die Einschaltverzögerung nicht in den Betriebsarten "Pulsweitenmodulation PWM" und "PWM mit DC-Motor" einstellen.

Ausschaltverzögerung

Mit diesem Parameter legen Sie die Verzögerung fest, mit welcher eine fallende Flanke des Digitaleingangs DI0 auf den Digitalausgang DQA wirkt.

Die Voreinstellung ist "0 µs".

Einschaltdauer

Mit diesem Parameter legen Sie das Impuls-Periode-Verhältnis (auch Tastverhältnis oder Tastgrad genannt) des Ausgabesignals im gewählten Ausgabeformat fest.

Die Voreinstellung ist "50 %".

Dithering

Mit diesem Parameter legen Sie fest, ob das PWM-Signal mit einem Dither-Signal überlagert wird.

In der Voreinstellung ist kein Dithering aktiviert.

Steigung Dither-Einschaltrampe

Mit diesem Parameter legen Sie die Steigung der Einschaltrampe der Dither-Amplitude für einen theoretischen Anstieg von 0 % auf 100 % fest.

Die Steigung beeinflusst die effektive Rampenzeit:

(Dither-Amplitude) x (Steigung Dither-Einschaltrampe) = Effektive Rampenzeit

Beispiel für Berechnung der effektiven Rampenzeit:

10 % x 2500 ms/100 % = 250 ms

Die Voreinstellung ist "0".

Steigung Dither-Ausschaltrampe

Mit diesem Parameter legen Sie die Steigung der Ausschaltrampe der Dither-Amplitude für ein theoretisches Sinken von 100 % auf 0 % fest.

Die Steigung beeinflusst die effektive Rampenzeit:

(Dither-Amplitude) x (Steigung Dither-Ausschaltrampe) = Effektive Rampenzeit

Beispiel für Berechnung der effektiven Rampenzeit:

10 % x 2500 ms/100 % = 250 ms

Die Voreinstellung ist "0".

Dither-Amplitude

Mit diesem Parameter legen Sie die Amplitude des Dither-Signals bezogen auf die Periodendauer des Ausgabesignals fest.

Die Voreinstellung ist "5,0 %".

Dither-Periodendauer

Mit diesem Parameter legen Sie die Periodendauer des Dither-Signals fest. Der Parameter ist in einem feststehenden Raster in Abhängigkeit der Periodendauer des PWM-Signals einstellbar.

Die Voreinstellung ist "50000 µs".

Dither-Frequenz

Dieser Wert wird berechnet aus der parametrierten Periodendauer des Dither-Signals.

Stromgeregelt

Mit diesem Parameter aktivieren Sie den PID-Algorithmus für die Regelung des Ausgangsstroms.

In der Voreinstellung ist der Parameter deaktiviert.

Hinweis

High-Speed-Ausgang

Sie können die Stromregelung mit den zugehörigen Parametern nur in der Betriebsart "Pulsweitenmodulation PWM" mit deaktiviertem "High-Speed-Ausgang (0.1 A)" verwenden.

P-Anteil einschalten

Mit diesem Parameter aktivieren den Proportionalanteil des PID-Algorithmus.

In der Voreinstellung ist der Parameter aktiviert.

I-Anteil einschalten

Mit diesem Parameter aktivieren den Integralanteil des PID-Algorithmus.

In der Voreinstellung ist der Parameter aktiviert.

D-Anteil einschalten

Mit diesem Parameter aktivieren den Differentialanteil des PID-Algorithmus.

In der Voreinstellung ist der Parameter deaktiviert.

Strombezugswert

Mit diesem Parameter legen Sie den maximalen Strom als Bezugswert für die Stellwerte und den Stromsollwert fest.

Die Voreinstellung ist "0 mA".

Totzonenbreite

Mit diesem Parameter legen Sie die Abweichung des Ausgangsstroms vom Stromsollwert fest, innerhalb welcher nicht nachgeregelt wird. Die Abweichung bezieht sich auf ober- sowie unterhalb des Stromsollwerts.

Die Voreinstellung ist "0 µA".

Stellwert obere Begrenzung

Mit diesem Parameter legen Sie die obere Regelungsgrenze relativ zum Strombezugswert fest. "27648" entspricht dabei 100% des Strombezugswerts. Der Wert muss größer sein als der "Stellwert untere Begrenzung".

Die Voreinstellung ist "27648".

Stellwert untere Begrenzung

Mit diesem Parameter legen Sie die untere Regelungsgrenze relativ zum Strombezugswert fest. "0" entspricht dabei 0% des Strombezugswerts. Der Wert muss kleiner sein als der "Stellwert obere Begrenzung".

Die Voreinstellung ist "0".

Proportionalbeiwert (Verstärkung)

Mit diesem Parameter legen Sie den Verstärkungsfaktor für den P-Anteil des PID-Algorithmus fest.

Die Voreinstellung ist "2,0000".

Integrationszeit

Mit diesem Parameter legen Sie die Zeitdauer fest, die der I-Anteil des PID-Algorithmus verwendet.

Die Voreinstellung ist "2,0000".

Differenzierzeit

Dieser Wert gibt die Zeitdauer an, die der D-Anteil des PID-Algorithmus verwendet.

Verzögerungszeit des D-Anteils

Dieser Wert gibt die Verzögerungszeit des D-Anteils des PID-Algorithmus an.

5.2.1.2 Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über das Modul erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.

3. Klicken Sie mit der rechten Maustaste auf das Modul und wählen Sie "Online & Diagnose".
4. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch des Moduls.

5.2.2 TM PTO 4 einsetzen

5.2.2.1 Technologiemodul konfigurieren und parametrieren

Technologiemodul zur Hardware-Konfiguration hinzufügen

Voraussetzungen

- Das Projekt ist angelegt.
- Die CPU S7-1500 ist angelegt.

Vorgehen

Um das Technologiemodul zur Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie einen Baugruppenträger aus.
3. Wählen Sie das Technologiemodul aus dem Baugruppenkatalog aus: "Technologiemodule > Pulsausgabe > TM PTO 4 > Artikelnummer"
4. Ziehen Sie das Technologiemodul per Drag & Drop auf den gewünschten Steckplatz im Baugruppenträger.

Parametereinstellung (Hardware-Konfiguration) öffnen

Öffnen über Gerätesicht

Gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie auf das Modul.

Kanalkonfiguration

Kanalkonfiguration

Mit diesem Parameter legen Sie die Anzahl der verwendeten Kanäle fest. Die Kanäle werden in aufsteigender Reihenfolge belegt.

Sie können unter folgenden Optionen auswählen:

- 4 Kanäle (voreingestellt)
- 3 Kanäle
- 2 Kanäle
- 1 Kanal

Betriebsart

Signalart

Mit diesem Parameter legen Sie die Art der PTO-Impulsausgabe (Seite 1312) fest.

Option	Bedeutung
Impuls (P) und Richtung (D) (voreingestellt)	Der Kanal gibt ein Impulssignal und ein Richtungssignal aus.
Vorwärts zählen (A), rückwärts zählen (B)	Der Kanal gibt ein Vorwärtssignal und ein Rückwärtssignal aus.
Inkrementalgeber (A, B phasenversetzt)	Der Kanal gibt zwei phasenversetzte Signale mit Einfachauswertung aus.
Inkrementalgeber (A, B phasenversetzt, vierfach)	Der Kanal gibt zwei phasenversetzte Signale mit Vierfachauswertung aus.

Signalschnittstelle

Mit diesem Parameter legen Sie fest, welche Schnittstelle für die Impulsausgabe verwendetet wird.

Option	Bedeutung
24 V, asymmetrisch	Der Kanal gibt 24 V-Signale auf den Klemmen DQm.0 und DQm.1 aus.
RS422, symmetrisch / TTL (5 V), asymmetrisch (voreingestellt)	Der Kanal gibt entweder RS422-Signale auf den Klemmen P/A und D/B und den jeweils invertierten Klemmen aus oder 5 V-TTL-Signale auf den Klemmen P/A und D/B aus.

Impulspause bei Richtungsumkehr

Mit diesem Parameter legen Sie die Dauer fest, die mindestens zwischen einem Richtungswechsel und dem ersten ausgegebenen Impuls in der neuen Richtung liegt.

Sie können unter folgenden Optionen auswählen:

- 0 ms (voreingestellt)
- 1 ms
- 4 ms
- 10 ms

Diagnosealarme

Diagnosealarme freigeben

Mit diesem Parameter legen Sie fest, ob bei einer fehlenden Versorgungsspannung oder einem Fehler an den Digitalausgängen ein Diagnosealarm ausgelöst wird. Der erkannte Fehler wird für den jeweiligen Kanal mit den Rückmeldebits Fault_Present und Sensor_Error angezeigt.

Hinweis

Die Hardware-Ausgänge der Kanäle 0 und 1 sowie der Kanäle 2 und 3 haben je eine gemeinsame Diagnose. Dadurch wird bei einem Fehler an einem Ausgang die Impulsausgabe je beider Kanäle gestoppt. Ein Kurzschluss an einem Ausgang kann zusätzlich auch den Diagnosealarm "Versorgungsspannung fehlt" auslösen.

Diagnosealarme bearbeiten Sie in einem Alarm-OB.

In der Voreinstellung ist der Parameter deaktiviert.

Achsparameter

Hinweis

Technologieobjekt für Motion Control

Achten Sie darauf, dass der jeweilige Parameterwert mit dem Wert in der Konfiguration des verwendeten Technologieobjekts übereinstimmt.

Inkremente pro Umdrehung

Mit diesem Parameter legen Sie die Anzahl der Schritte fest (auch Mikroschritte), die einer Umdrehung des Antriebs entspricht.

Der Wertebereich beträgt 1 bis 1000000. Der Wert beeinflusst den Wertebereich der maximalen Drehzahl und der Bezugsdrehzahl. Die Voreinstellung ist "200".

Bezugsdrehzahl

Mit diesem Parameter legen Sie die Drehzahl fest, mit welcher der Antrieb bei einem Drehzahlsollwert von 100 % dreht. Der Wertebereich beträgt 1,0 bis 20000,0 U/min. Der zulässige Wertebereich für den Drehzahlsollwert beträgt –200 % bis 200 % der Bezugsdrehzahl.

Die Bezugsdrehzahl beeinflusst den Wertebereich der maximalen Drehzahl und der Inkremente pro Umdrehung. Die Voreinstellung ist "3000.0".

Maximale Drehzahl

Mit diesem Parameter legen Sie die für die Anwendung maximal zulässige Drehzahl fest.

Die folgende Tabelle zeigt die Berechnung des Wertebereichs der maximalen Drehzahl:

Signal-schnittstelle	Wertebereich der maximalen Drehzahl			
	Untere Grenze		Obere Grenze (Es gilt der kleinere der beiden Werte)	
	Signalauswertung "Einfach"	Signalauswertung "Vierfach"	Signalauswertung "Einfach"	Signalauswertung "Vierfach"
24 V, asymmetrisch	0,1 Hz * 60 / (Inkremente pro Umdrehung)	0,1 Hz * 60 * 4 / (Inkremente pro Umdrehung)	• 2 * Bezugsdrehzahl • 200000 Hz * 60 / (Inkremente pro Umdrehung)	• 2 * Bezugsdrehzahl • 200000 Hz * 60 * 4 / (Inkremente pro Umdrehung)
TTL (5 V), asymmetrisch				
RS422, symmetrisch			• 2 * Bezugsdrehzahl • 1000000 Hz * 60 / (Inkremente pro Umdrehung)	• 2 * Bezugsdrehzahl • 1000000 Hz * 60 * 4 / (Inkremente pro Umdrehung)

Die Voreinstellung ist "3000.0".

Bits im inkr. Istwert (G1_XIST1)

Dieser Parameter gibt die Anzahl der Bits an für die Kodierung der Feinauflösung innerhalb des inkrementellen Positions-Istwerts von G1_XIST1.

Der Wert ist für dieses Modul immer "0".

Zeit Schnellhalt (OFF3)

Mit diesem Parameter legen Sie die Zeitspanne fest, um einen Schnellhalt von Maximaldrehzahl bis zum Stillstand durchzuführen.

Der Wertebereich beträgt 1 bis 65535 ms. Die Voreinstellung ist "1000 ms".

Zeit Rampenstopp (OFF1)

Mit diesem Parameter legen Sie die Zeitspanne fest, um einen Halt von Maximaldrehzahl bis zum Stillstand durchzuführen.

Der Wertebereich beträgt 1 bis 65535 ms. Die Voreinstellung ist "5000 ms".

Hardwareein-/ausgänge

Antriebsfreigabe verwenden

Mit diesem Parameter legen Sie fest, ob ein Hardware-Ausgang für das Freigabesignal für den Antrieb verwendet wird.

In der Voreinstellung ist der Parameter deaktiviert.

Antriebsfreigabe-Ausgang

Mit diesem Parameter legen Sie fest, welcher Hardware-Ausgangs für die Freigabe des Antriebs verwendet wird.

Sie können unter folgenden Optionen auswählen:

- DQ0 (voreingestellt)
- DIQ2

DI0 als Referenzschalter verwenden

Mit diesem Parameter legen Sie fest, ob der Hardware-Ausgang DI0 für das Referenzschalter-Signal für den Antrieb verwendet wird. Über das Referenzschalter-Signal können Sie die Referenzmarke mit der aktuellen Position der Antriebsachse synchronisieren.

In der Voreinstellung ist der Parameter deaktiviert.

Flankenauswahl Referenzschalter

Mit diesem Parameter legen Sie die Flanke an DI0 fest, mit der die Referenzmarke erfasst wird.

Sie können unter folgenden Optionen auswählen:

- Bei steigender Flanke (voreingestellt)
- Bei fallender Flanke

Hinweis

Der Parameter ist nur wirksam, falls Sie einen Referenzschalter-Eingang verwenden.

Dieser Parameter ist nur auswählbar und wirksam, wenn "Antriebsfreigabe verwenden" aktiviert ist. Die Voreinstellung ist "Keine".

DI1 als Messtaster verwenden

Mit diesem Parameter legen Sie fest, ob der Hardware-Ausgang DI1 für das Messtaster-Signal für den Antrieb verwendet wird. Über das Messtaster-Signal können Sie die aktuelle Position der Antriebsachse speichern.

In der Voreinstellung ist der Parameter deaktiviert.

"Antrieb bereit" verwenden

Mit diesem Parameter legen Sie fest, ob an einem Hardware-Digitaleingang das Bereit-Signal des Antriebs verwendet wird.

In der Voreinstellung ist der Parameter deaktiviert.

Eingang "Antrieb bereit"

Mit diesem Parameter legen Sie fest, an welchem Hardware-Digitaleingang das Bereit-Signal des Antriebs angeschlossen und eingelesen wird.

Sie können unter folgenden Optionen auswählen:

- DI0 (voreingestellt)
- DI1
- DIQ2

Hinweis

Dieser Parameter ist nur auswählbar und wirksam, wenn ""Antrieb bereit" verwenden" aktiviert ist.

Eingangsverzögerung

Mit diesem Parameter unterdrücken Sie Signalstörungen an den Digitaleingängen. Änderungen am Signal werden erst erfasst, wenn sie länger als die parametrierte Eingangsverzögerungszeit stabil anstehen.

Sie können unter folgenden Optionen auswählen:

- Keine
  (Eingangsverzögerung von 4 µs, Mindestimpulsbreite von 3 µs)
- 0,05 ms
- 0,1 ms (voreingestellt)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Hinweis

Wenn Sie die Option "Keine" oder "0,05 ms" wählen, müssen Sie geschirmte Leitungen für den Anschluss der Digitaleingänge verwenden.

Lebenszeichen-Fehler

Tolerierte Anzahl Lebenszeichen-Fehler

Mit diesem Parameter legen Sie fest, wie viele Fehler des Master Sign-Of-Life vom Modul toleriert werden. Ein Überschreiten der Anzahl löst eine Fehlermeldung über das Rückmeldebit Sensor_Error aus.

Der Wertebereich beträgt 1 bis 65535. Der Wert 65535 bedeutet, dass keine Überwachung auf Lebenszeichen-Fehler stattfindet. Die Voreinstellung ist "1".

5.2.2.2 Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über das Modul erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Gerätekonfiguration der CPU bzw. IM.
2. Wählen Sie die Gerätesicht.
3. Klicken Sie mit der rechten Maustaste auf das Modul und wählen Sie "Online & Diagnose".
4. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch des Moduls.

5.2.3 Kompakt-CPU einsetzen

5.2.3.1 Kompakt-CPU konfigurieren und parametrieren

Kompakt-CPU zur Hardware-Konfiguration hinzufügen

Voraussetzung

Das Projekt ist angelegt.

Vorgehen

Um eine Kompakt-CPU zur Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf "Neues Gerät hinzufügen".
   Der Dialog "Neues Gerät hinzufügen" wird geöffnet.
2. Wählen Sie "Controller".
3. Wählen Sie die Kompakt-CPU aus:
   "SIMATIC S7-1500 > CPU > Name der Kompakt-CPU > Artikelnummer"
4. Bestätigen Sie mit "OK".

Ergebnis

In der Projektnavigation wird die neue Kompakt-CPU mit folgenden Objekten angezeigt. Über Doppelklick gelangen Sie zu dem gewünschten Editor.

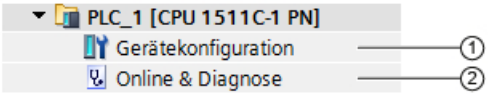

	Objekt	Beschreibung
①	Gerätekonfiguration	Im Inspektorfenster (pro PTO/PWM-Kanal): • Einstellung der Betriebsart (Seite 1334) • Einstellung des Verhaltens bei CPU-STOP (Seite 1335) • Freigabe Diagnosealarme (Seite 1336) • Einstellung Parameter für PWM oder Frequenzausgabe (Seite 1338) • Einstellung Achsparameter für PTO-Betriebsarten (Seite 1338) • Zuordnung Signale zu Ein- und Ausgängen (Seite 1336) • Einstellung der Moduladressen
②	Online & Diagnose (Seite 1340)	• Hardware diagnostizieren • Informationen über Kompakt-CPU erhalten • Firmware-Update durchführen

Kompatibilität 1511C (Impulsgeneratoren der Kompakt-CPU 1512C-1 PN)

Frontstecker-Belegung wie CPU 1511C

Mit diesem Parameter legen Sie fest, ob für die Impulsgeneratoren der CPU 1512C-1 PN die Anschlussbelegung der Frontstecker der CPU 1511C-1 PN verwendet wird:

Option	Bedeutung
Deaktiviert (voreingestellt)	Die CPU 1512C-1 PN verwendet die Anschlussbelegung der integrierten Frontstecker. 1512C-1 PN unterstützt die Nutzung von Anschlüssen beider Frontstecker der digitalen Onboard-Peripherie für die Impulsgeneratoren. Die Zuweisung von Hardwareeingängen und -ausgängen für die PTO/PWM-Kanäle ist im Gerätehandbuch der CPU 1512C-1 PN beschrieben.
Aktiviert	Die CPU 1512C-1 PN verwendet die Anschlussbelegung der Frontstecker der CPU 1511C-1 PN. 1511C-1 PN unterstützt die Nutzung von Anschlüssen des ersten Frontsteckers der digitalen Onboard-Peripherie für die Impulsgeneratoren. Die Zuweisung von Hardwareeingängen und -ausgängen für die PTO/PWM-Kanäle ist im Gerätehandbuch der CPU 1511C-1 PN beschrieben.

Betriebsart

Betriebsart

Mit diesem Parameter legen Sie fest, für welche Funktion der Impulsgenerator der Kompakt-CPU eingesetzt wird. Die betriebsartabhängigen Einstellmöglichkeiten finden Sie unter:

Verhalten bei CPU-STOP (Seite 1335)

Diagnosealarme (Seite 1336)

Hardwareein-/ausgänge (PWM, Frequenzausgabe) (Seite 1336) bzw. Hardwareein-/ausgänge (PTO) (Seite 1336)

Parameter (PWM, Frequenzausgabe) (Seite 1338) bzw. Achsparameter (PTO) (Seite 1338)

Betriebsart	Bedeutung	Betriebsartabhängige Parameter
Deaktiviert (voreingestellt)	Der Impulsgenerator ist nicht aktiviert. Der Kanal gibt keine Impulse aus.	—
Pulsweitenmodulation PWM	Der Kanal gibt ein pulsweitenmoduliertes Signal (Seite 1300) mit der parametrierten Mindestimpulsdauer und der parametrierten Periodendauer aus. Die Einschaltdauer und die Periodendauer geben Sie in der Steuerschnittstelle vor.	• Verhalten bei CPU-STOP • Ersatzwert für Impulsausgang (DQA) • Fehlende Versorgungsspannung L+ • Impulsausgang (DQA) • High-Speed-Ausgang (0.1 A) • Ausgabeformat • Mindestimpulsdauer • Periodendauer

Betriebsart	Bedeutung	Betriebsartabhängige Parameter
Frequenzausgabe	Der Kanal gibt ein Signal mit einer festen Einschaltdauer (Seite 1305) von 50 % aus. Die Frequenz geben Sie in der Steuerschnittstelle vor.	• Verhalten bei CPU-STOP • Ersatzwert für Impulsausgang (DQA) • Fehlende Versorgungsspannung L+ • Impulsausgang (DQA) • High-Speed-Ausgang (0.1 A) • Ausgabeformat
PTO (Impuls (A) und Richtung (B))	Der Kanal gibt ein Impulssignal und ein Richtungssignal (Seite 1307) für eine Motion Control-Anwendung aus.	• Fehlende Versorgungsspannung L+ • Inkremente pro Umdrehung • Bezugsdrehzahl • Maximale Drehzahl • Bits im inkr. Istwert (G1_XIST1) • Zeit Schnellhalt • Referenzschalter-Eingang • Flankenauswahl Referenzschalter • Messeingang • Eingang "Antrieb bereit" • Impulsausgang (A) / Taktgeber vorwärts (A) / Ausgang Taktgeber (A) • Richtungsausgang (B) / Taktgeber rückwärts (B) / Ausgang Taktgeber (B) • Antriebsfreigabe-Ausgang
PTO (Vorwärts zählen (A), rückwärts zählen (B))	Der Kanal gibt ein Vorwärtssignal und ein Rückwärtssignal (Seite 1307) für eine Motion Control-Anwendung aus.	
PTO (A, B phasenversetzt)	Der Kanal gibt zwei phasenversetzte Signale mit Einfachauswertung (Seite 1307) für eine Motion Control-Anwendung aus.	
PTO (A, B phasenversetzt, vierfach)	Der Kanal gibt zwei phasenversetzte Signale mit Vierfachauswertung (Seite 1307) für eine Motion Control-Anwendung aus.	

Verhalten bei CPU-STOP

Verhalten bei CPU-STOP

Mit diesem Parameter legen Sie die Reaktion des Kanals auf den Ausfall der Steuerung fest.

Verhalten bei CPU-STOP	Bedeutung
Ersatzwert ausgeben (voreingestellt)	Der Kanal gibt bis zum nächsten STOP-RUN-Übergang der CPU an seinem Impulsausgang (DQA) den parametrierten Ersatzwert aus. Das Rückmeldebit STS_ENABLE wird auf 0 gesetzt. Nach einem STOP-RUN-Übergang wird der Kanal in seinen Anlaufzustand gesetzt. Bevor Sie eine neue Impulsausgabe starten können, müssen Sie das Steuerbit SW_ENABLE auf 0 setzen.
Weiterarbeiten	Der Kanal arbeitet weiter. Der Impulsausgang (DQA) schaltet weiterhin entsprechend der Parametrierung und dem letzten Sollwert.

Ersatzwert für Impulsausgang (DQA)

Mit diesem Parameter legen Sie für das Verhalten "Ersatzwert ausgeben" fest, welchen Wert der Kanal bei einem CPU-STOP am Digitalausgang DQA ausgeben soll.

Die Voreinstellung ist "0".

Diagnosealarme

Fehlende Versorgungsspannung L+

Mit diesem Parameter legen Sie fest, ob bei einer fehlenden Versorgungsspannung ein Diagnosealarm ausgelöst wird. Diagnosealarme bearbeiten Sie in einem Alarm-OB.

In der Voreinstellung ist der Parameter deaktiviert.

Hardwareein-/ausgänge (PWM, Frequenzausgabe)

Impulsausgang (DQA)

Mit diesem Parameter legen Sie fest, welcher Hardware-Digitalausgang für die Impulsausgabe verwendet wird.

High-Speed-Ausgang (0.1 A)

Mit diesem Parameter legen Sie fest, ob der jeweilige Hardware-Digitalausgang als schneller Gegentakt- oder als P-Schalter arbeitet:

Option	Bedeutung
Aktiviert	Der Digitalausgang arbeitet als schneller Gegentakt-Schalter mit bis zu 100 kHz und ist mit einem Nennlaststrom von 0,1 A belastbar. Ein Gegentakt-Schalter wird wechselweise nach DC 24 V und Masse geschaltet. Dadurch sind sehr schnelle Flanken möglich.
Deaktiviert (voreingestellt)	Der Digitalausgang arbeitet als 24 V-P-Schalter mit Bezug zu M (bis zu 10 kHz) und ist mit einem Nennlaststrom von 0,5 A belastbar.

Hinweis

Der Parameter ist nur wirksam, falls der als Impulsausgang gewählte Hardware-Ausgang den 100 kHz-Betrieb unterstützt. Für Hardware-Ausgänge, die nur Frequenzen bis 100 Hz unterstützen, ist der Parameter nicht verfügbar.

Hardwareein-/ausgänge (PTO)

Referenzschalter-Eingang

Mit diesem Parameter legen Sie fest, welcher Hardware-Digitaleingang als Referenzschalter-Eingang verwendet wird.

Die Voreinstellung ist "Keine".

Flankenauswahl Referenzschalter

Mit diesem Parameter legen Sie die Flanke am Referenzschalter-Eingang fest, mit der die Referenzmarke erfasst wird.

Die Voreinstellung ist "Steigende Flanke".

Hinweis

Der Parameter ist nur wirksam, falls Sie einen Referenzschalter-Eingang verwenden.

Messeingang

Mit diesem Parameter legen Sie fest, welcher Hardware-Digitaleingang als Messeingang verwendet wird.

Die Voreinstellung ist "Keine".

Eingang "Antrieb bereit"

Mit diesem Parameter legen Sie fest, an welchem Hardware-Digitaleingang das Bereit-Signal des Antriebs angeschlossen und eingelesen wird.

Die Voreinstellung ist "Keine".

Impulsausgang (A) / Taktgeber vorwärts (A) / Ausgang Taktgeber (A)

Dieser Parameter gibt an, welcher Hardware-Ausgang für das PTO-Signal A verwendet wird. Der Wert ist nicht änderbar.

Richtungsausgang (B) / Taktgeber rückwärts (B) / Ausgang Taktgeber (B)

Dieser Parameter gibt an, welcher Hardware-Ausgang für das PTO-Signal B verwendet wird. Sie können den Wert nur in der Betriebsart "PTO (Impuls (A) und Richtung (B))" ändern.

Antriebsfreigabe-Ausgang

Mit diesem Parameter legen Sie fest, welcher Hardware-Ausgangs für die Freigabe des Antriebs verwendet wird.

Die Voreinstellung ist "Keine".

Parameter (PWM, Frequenzausgabe)

Ausgabeformat

Betriebsart "Pulsweitenmodulation PWM":

Mit diesem Parameter legen Sie das Format bzw. den Wertebereich fest, mit dem die Einschaltdauer ausgewertet wird:

Ausgabeformat	Wertebereich
S7-Analogausgabe	0 ... 27648 ("27648" entspricht 100 % des Bezugswerts)
1/100 (voreingestellt)	0 ... 100
1/1000	0 ... 1000
1/10000	0 ... 10000

Betriebsart "Frequenzausgabe":

Als Ausgabeformat ist "1 Hz" ausgewählt und nicht änderbar.

Mindestimpulsdauer

Mit diesem Parameter legen Sie die minimal zulässige Impulsdauer für das PWM-Signal fest.

Impulse und Pausen, die den Wert unterschreiten, werden unterdrückt.

Die Voreinstellung ist "0 µs".

Periodendauer

Mit diesem Parameter legen Sie die Periodendauer des Ausgabesignals fest.

Die Voreinstellung ist "2000000 µs". Das entspricht 2 s.

Achsparameter (PTO)

Inkremente pro Umdrehung

Mit diesem Parameter legen Sie die Anzahl der Schritte fest (auch Mikroschritte), die einer Umdrehung des Antriebs entspricht.

Die Voreinstellung ist "200".

Hinweis

Technologieobjekt für Motion Control

Achten Sie darauf, dass der Parameterwert mit dem Wert in der Konfiguration des verwendeten Technologieobjekts übereinstimmt.

Bezugsdrehzahl

Mit diesem Parameter legen Sie die Drehzahl fest, auf welche sich der Drehzahlsollwert bezieht.

Der zulässige Wertebereich für den Drehzahlsollwert beträgt –200 % bis 200 % der Bezugsdrehzahl.

Die Voreinstellung ist "3000.0".

Hinweis

Technologieobjekt für Motion Control

Achten Sie darauf, dass der Parameterwert mit dem Wert in der Konfiguration des verwendeten Technologieobjekts übereinstimmt.

Maximale Drehzahl

Mit diesem Parameter legen Sie die für die Anwendung maximal zulässige Drehzahl fest.

Die Voreinstellung ist "3000.0".

Hinweis

Technologieobjekt für Motion Control

Achten Sie darauf, dass der Parameterwert mit dem Wert in der Konfiguration des verwendeten Technologieobjekts übereinstimmt.

Bits im inkr. Istwert (G1_XIST1)

Dieser Parameter gibt die Anzahl der Bits an für die Kodierung der Feinauflösung innerhalb des inkrementellen Istwerts von G1_XIST1.

Hinweis

Technologieobjekt für Motion Control

Achten Sie darauf, dass der Parameterwert mit dem Wert in der Konfiguration des verwendeten Technologieobjekts übereinstimmt.

Zeit Schnellhalt

Mit diesem Parameter legen Sie die Zeitspanne fest, um einen Schnellhalt von Maximaldrehzahl bis zum Stillstand durchzuführen.

Die Voreinstellung ist "1000 ms".

5.2.3.2 Modul Online & Diagnose

Diagnose anzeigen und auswerten

Über die Online- und Diagnosesicht können Sie die Hardware diagnostizieren. Außerdem können Sie

- Informationen über die Kompakt-CPU erhalten (z. B. Firmware-Version und Seriennummer)
- Bei Bedarf ein Firmware-Update durchführen

Vorgehen

Um den Anzeige-Editor für die Diagnosefunktionen zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der Kompakt-CPU.
2. Doppelklicken Sie auf das Objekt "Online & Diagnose".
3. Klicken Sie in der Diagnosenavigation auf die gewünschte Anzeige.

Weitere Informationen

Weitere Informationen zu den Diagnosemeldungen und möglichen Abhilfemaßnahmen finden Sie im Gerätehandbuch der Kompakt-CPU.

Motion Control

6

6.1 Motion Control (S7-1200, S7-1500)

6.1.1 S7-1200 Motion Control einsetzen

6.1.1.1 Einleitung

Motionfunktionalität der CPU S7-1200

Das TIA-Portal unterstützt Sie zusammen mit der Motion Control-Funktionalität der CPU S7-1200 bei der Steuerung von Schrittmotoren und Servomotoren:

- Im TIA-Portal konfigurieren Sie die Technologieobjekte Positionierachse und Auftragstabelle. Mit Hilfe dieser Technologieobjekte steuert die CPU S7-1200 die Ausgänge zur Ansteuerung der Antriebe.
- Im Anwenderprogramm steuern Sie mit Hilfe von Motion Control-Anweisungen die Achse und initiieren hiermit Bewegungsaufträge Ihres Antriebs.

Siehe auch

Hardwarekomponenten für Motion Control (Seite 1342)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Verwendung des Technolgieobjekts Auftragstabelle (Seite 1449)

Werkzeuge des Technologieobjekts Auftragstabelle (Seite 1450)

Hardwarekomponenten für Motion Control

Die nachfolgende Darstellung zeigt den prinzipiellen Hardwareaufbau für eine Motion Control-Anwendung mit der CPU S7-1200.

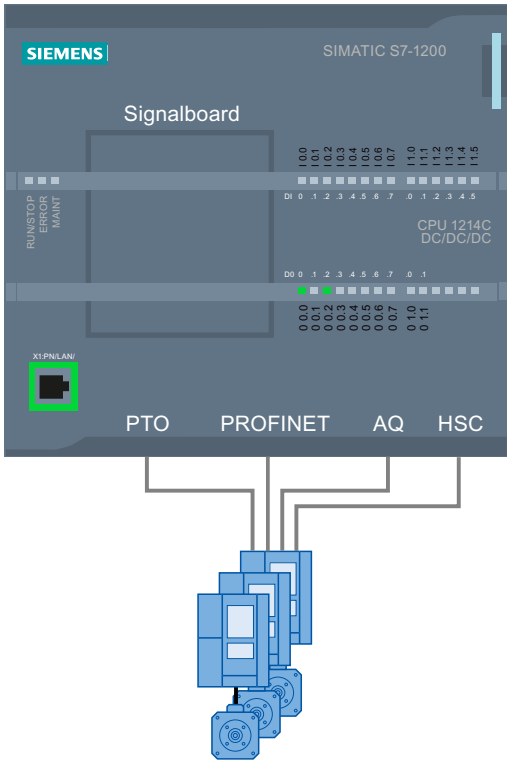

Antriebe und Geber

CPU S7-1200

Die CPU S7-1200 vereint die Funktionalität einer speicherprogrammierbaren Steuerung mit der Motion Control-Funktionalität für den Betrieb von Antrieben. Die Motion Control-Funktionalität übernimmt die Steuerung und Überwachung der Antriebe.

Signalboard

Mit den Signalboards erweitern Sie die CPU um weitere Ein- und Ausgänge.

Die digitalen Ausgänge können Sie bei Bedarf als Impulsgeneratorausgänge zur Ansteuerung von Antrieben verwenden. Bei CPUs mit Relais-Ausgängen kann das Impulssignal nicht auf den on-board Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen. Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, müssen Sie ein Signalboard mit digitalen Ausgängen verwenden.

Die analogen Ausgänge können Sie bei Bedarf zur Ansteuerung von analog angebundenen Antrieben verwenden.

PROFINET

Über die PROFINET-Schnittstelle stellen Sie die Onlineverbindung zwischen der CPU S7-1200 und dem Programmiergerät her. Neben den Onlinefunktionen der CPU stehen Ihnen für Motion Control zusätzliche Inbetriebnahme- und Diagnosefunktionen zur Verfügung.

Weiterhin ünterstützt PROFINET das PROFIdrive-Profil zur Anbindung von PROFIdrive-fähigen Antrieben und Gebern.

Antriebe und Geber

Antriebe sorgen für die Bewegung der Achse. Geber liefern die Istposition für die Lageregelung der Achse.

Folgende Tabelle zeigt die Anschlussmöglichkeiten für Antriebe und Geber:

Antriebsanbindung	Achse gesteuert / geregelt	Geberanschluss
PTO (Pulse Train Output) (Schrittmotoren und Servomotoren mit Impulsschnittstelle)	positionsgesteuert	-
Analogausgang (AQ)	lagegeregelt	• Geber am schnellen Zähler (HSC) • Geber am Technologiemodul (TM) • Geber am PROFINET
PROFINET	lagegeregelt	• Geber am Antrieb • Geber am schnellen Zähler (HSC) • Geber am Technologiemodul (TM) • Geber am PROFINET

Bezeichnung	Artikelnummer
CPU 1211C DC/DC/DC	6ES7211-1AE40-0XB0
CPU 1211C AC/DC/RLY	6ES7211-1BE40-0XB0
CPU 1211C DC/DC/RLY	6ES7211-1HE40-0XB0
CPU 1212C DC/DC/DC	6ES7212-1AE40-0XB0
CPU 1212C AC/DC/RLY	6ES7212-1BE40-0XB0
CPU 1212C DC/DC/RLY	6ES7212-1HE40-0XB0
CPU 1214C DC/DC/DC	6ES7214-1AG40-0XB0
CPU 1214C AC/DC/RLY	6ES7214-1BG40-0XB0
CPU 1214C DC/DC/RLY	6ES7214-1HG40-0XB0
CPU 1214FC DC/DC/DC	6ES7214-1AF40-0XB0
CPU 1214FC DC/DC/RLY	6ES7214-1HF40-0XB0
CPU 1215C DC/DC/DC	6ES7215-1AG40-0XB0
CPU 1215C AC/DC/RLY	6ES7215-1BG40-0XB0
CPU 1215C DC/DC/RLY	6ES7215-1HG40-0XB0
CPU 1215FC DC/DC/DC	6ES7215-1AF40-0XB0
CPU 1215FC DC/DC/RLY	6ES7215-1HF40-0XB0
CPU 1217C DC/DC/DC	6ES7217-1AG40-0XB0

Bezeichnung	Artikelnummer
Signalboard DI4 x DC 24 V (200 kHz)	6ES7221-3BD30-0XB0
Signalboard DI4 x DC 5 V (200 kHz)	6ES7 221-3AD30-0XB0
Signalboard DQ4 x DC 24 V (200 kHz)	6ES7222-1BD30-0XB0
Signalboard DQ4 x DC 5 V (200 kHz)	6ES7222-1AD30-0XB0
Signalboard DI2/DQ2 x DC 24 V (20 kHz)	6ES7223-0BD30-0XB0
Signalboard DI2/DQ2 x DC 24 V (200 kHz)	6ES7223-3BD30-0XB0
Signalboard DI2/DQ2 x DC 5 V (200 kHz)	6ES7223-3AD30-0XB0
Signalboard AQ1 x 12 bit (±10 V, 0 bis 20 mA)	6ES7 232-4HA30-0XB0

Bestellinformationen CPU-Firmware V4.1

Die nachfolgend aufgelisteten Bestellinformationen gelten für die aktuell installierte Lieferstufe (ohne evt. installierte Hardware Support Packages) des TIA-Portals.

Neuere Hardware-Komponenten können Sie über ein Hardware Support Package (HSP) installieren. Die Hardware-Komponente steht danach im Hardware-Katalog zur Verfügung.

Siehe auch

Motionfunktionalität der CPU S7-1200 (Seite 1341)

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

6.1.1.2 Grundlagen für das Arbeiten mit S7-1200 Motion Control

Schrittmotor am PTO

Für Motion Control relevante Ausgänge der CPU

Die Anzahl der nutzbaren Antriebe ist abhängig von der Anzahl der PTOs (Pulse Train Outputs) und der zur Verfügung stehenden Anzahl an Impulsgeneratorausgängen.

Die folgenden Aufstellungen geben Auskunft über die entsprechenden Abhängigkeiten:

Maximale Anzahl der PTOs

Mit der Technologieversion V4 stehen pro CPU 4 PTOs zur Verfügung. Damit können maximal 4 Antriebe angesteuert werden.

Signalart des PTO

Je nach gewählter Signalart des PTO werden 1-2 Impulsgeneratorausgänge pro PTO (Antrieb) benötigt:

Signalart	Anzahl Impulsgeneratorausgänge
Impuls A und Richtung B (Richtungsausgang deaktiviert) [1]	1
Impuls A und Richtung B [1]	2
Vorwärts takten A und rückwärts takten B	2
A/B phasenversetzt	2
A/B phasenversetzt - vierfach	2

[1] Der Richtungsausgang muss Onboard oder auf einem Signalboard sein.

Nutzbare Impulsgeneratorausgänge und Grenzfrequenzen

Die Relaisvarianten der CPUs können nur auf die Impulsgeneratorausgänge eines Signalboards zugreifen.

Je nach CPU können die Impulsgeneratorausgänge Q0.0 bis Q1.1 mit folgenden Grenzfrequenzen verwendet werden:

CPU	Q0.0	Q0.1	Q0.2	Q0.3	Q0.4	Q0.5	Q0.6	Q0.7	Q1.0	Q1.1
1211 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	-	-	-	-	-	-
1212 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	-	-	-	-
1214(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz
1215(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz	20 kHz
1217 (DC/DC/DC)	1 MHz	1 MHz	1 MHz	1 MHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz

Je nach Signalboard können die Impulsgeneratorausgänge Qx.0 bis Qx.3 mit folgenden Grenzfrequenzen verwendet werden:

Signalboard	Qx.0	Qx.1	Qx.2	Qx.3	-	-	-	-	-	-
DI2/DQ2 x DC24V 20kHz	20 kHz	20 kHz	-	-	-	-	-	-	-	-
DI2/DQ2 x DC24V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
DQ4 x DC24V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-
DI2/DQ2 x DC5V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
DQ4 x DC5V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-

Die untere Grenzfrequenz ist jeweils 1Hz.

Die Impulsgeneratorausgänge können den PTOs frei zugeordnet werden.

Hinweis

Werden entsprechend der Signalart Impulsgeneratorausgänge mit unterschiedlicher Grenzfrequenz verwendet, so wird die jeweils niedrigere Grenzfrequenz verwendet.

Eine Ausnahme bildet die Signalart "Impuls A und Richtung B"; mit dieser Signalart wird immer die Grenzfrequenz des Impulsgeneratorausgangs verwendet.

Hinweis

Zugriff auf Impulsgeneratorausgänge über das Prozessabbild

Wurde der PTO (Pulse Train Output) aktiviert und einer Achse zugeordnet, so übernimmt die Firmware die Kontrolle über die zugehörigen Impulsgenerator- und Richtungsausgänge.

Mit der Übernahme der Kontrolle wird auch die Verbindung zwischen dem Prozessabbild und dem Peripherie-Ausgang getrennt. Der Anwender hat zwar die Möglichkeit das Prozessabbild von Impulsgenerator- und Richtungsausgängen über Anwenderprogramm oder Beobachtungstabelle zu beschreiben, dieses wird jedoch nicht auf den Peripherie-Ausgang übertragen werden. Dementsprechend ist es auch nicht möglich den Peripherie-Ausgang über Anwenderprogramm oder Beobachtungstabelle zu beobachten. Die gelesenen Informationen spiegeln den Wert des Prozessabbilds wieder, welche nicht mit dem realen Zustand des Peripherie-Ausgangs übereinstimmen.

Bei allen anderen, nicht fest durch die CPU-Firmware verwendeten CPU-Ausgängen, kann der Status des Peripherie-Ausgangs wie gewohnt über das Prozessabbild gesteuert bzw. überwacht werden.

Ausgänge für Antriebssignale

Für Motion Control können Sie optional eine Antriebsschnittstelle für "Antriebsfreigabe" und "Antrieb bereit" parametrieren.

Bei Verwendung der Antriebsschnittstelle kann der digitale Ausgang für die Antriebsfreigabe und der digitale Eingang für "Antrieb bereit" frei gewählt werden.

Grenzwerte Beschleunigung / Verzögerung

Für die Beschleunigung und Verzögerung gelten folgende Grenzwerte:

Beschleunigung / Verzögerung	Wert
Minimale Beschleunigung / Verzögerung	5,0E-3 Pulse/s^2
Maximale Beschleunigung / Verzögerung	9,5E+9 Pulse/s^2

Grenzwerte Ruck

Für den Ruck gelten folgende Grenzwerte:

Ruck	Wert
Minimaler Ruck	4,0E-3 Pulse/s^3
Maximaler Ruck	1,0E+10 Pulse/s^3

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Technologieversion V1...3) (Seite 1581)

Prinzip der Impulsschnittstelle (Seite 1347)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Hardware- und Software-Endschalter (Seite 1367)

Ruckbegrenzung (Seite 1368)

Referenzieren (Seite 1369)

Hardwarekomponenten für Motion Control (Seite 1342)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Prinzip der Impulsschnittstelle

In Abhängigkeit zu den Einstellungen des Schrittmotors hat jeder Impuls das Verfahren des Schrittmotors um einen definierten Winkel zur Folge. Ist der Schrittmotor auf beispielsweise 1000 Impulse pro Umdrehung eingestellt, verfährt der Schrittmotor pro Impuls um 0,36°.

Die Geschwindigkeit des Schrittmotors wird über die Anzahl der Impulse pro Zeiteinheit bestimmt.

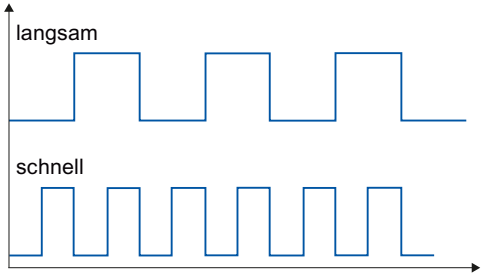

Die hier getroffenen Aussagen gelten auch für Servomotoren mit Impulsschnittstelle.

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Hardware- und Software-Endschalter (Seite 1367)

Ruckbegrenzung (Seite 1368)

Referenzieren (Seite 1369)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Zusammenhang zwischen der Signalart und der Fahrtrichtung

Die CPU gibt die Geschwindigkeit und die Fahrtrichtung über zwei Ausgänge aus.

Die Zusammenhänge zwischen Konfiguration und Fahrtrichtung unterscheiden sich je nach gewählter Signalart. In der Konfiguration der Achse unter "Grundparameter > Allgemein" können Sie folgende Signalarten konfigurieren:

- "PTO – Impuls A und Richtung B"
- "PTO – vorwärts takten A und rückwärts takten B" (ab V4)
- "PTO – A/B phasenversetzt" (ab V4)
- "PTO – A/B phasenversetzt, vierfach" (ab V4)

Den Richtungssinn konfigurieren Sie in der Konfiguration der Achse unter "Erweiterte Parameter > Mechanik". Wenn Sie die Option "Richtungssinn invertieren" aktivieren, wird die nachfolgend beschriebene Richtungslogik für die jeweilige Signalart invertiert.

PTO – Impuls A und Richtung B

Bei dieser Signalart werden die Impulse des Impulsausgangs und der Pegel des Richtungsausgangs ausgewertet.

Die Impulse werden über den Impulsausgang der CPU ausgegeben. Der Richtungsausgang der CPU gibt die Drehrichtung des Antriebs vor:

- 5 V/24 V am Richtungsausgang ⇒ positive Drehrichtung
- 0 V am Richtungsausgang ⇒ negative Drehrichtung

Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Werte gelten nicht für die Differenzausgänge der CPU 1217.

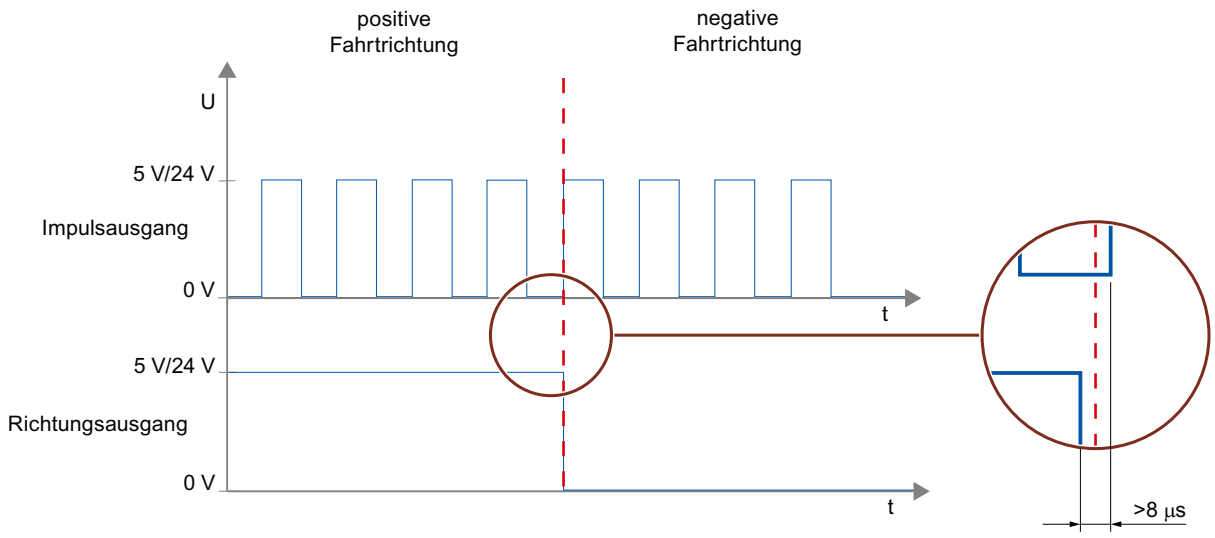

PTO – vorwärts takten A und rückwärts takten B (ab V4)

Bei dieser Signalart werden die Impulse eines Ausgangs ausgewertet.

Der Impuls für die positive Richtung wird über den "Impulsausgang vorwärts" ausgegeben. Der Impuls für die negative Richtung wird über den "Impulsausgang rückwärts" ausgegeben.

Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Werte gelten nicht für die Differenzausgänge der CPU 1217.

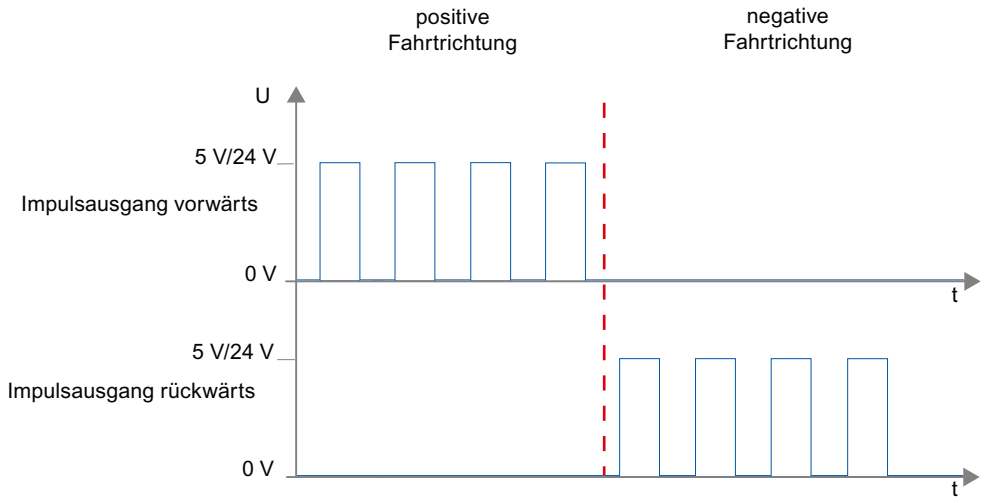

PTO – A/B phasenversetzt (ab V4)

Bei dieser Signalart werden die steigenden Flanken jeweils eines Ausgangs ausgewertet.

Der Impuls wird über den Ausgang "Signal A" und phasenverschoben über den Ausgang "Signal B" ausgegeben. Die Phasenverschiebung zwischen den Ausgängen definiert die Drehrichtung:

- Signal A um 90° voreilend gegenüber Signal B ⇒ positive Drehrichtung
- Signal B um 90° voreilend gegenüber Signal A ⇒ negative Drehrichtung

PTO – A/B phasenversetzt, vierfach (ab V4)

Bei dieser Signalart werden die steigenden und fallenden Flanken beider Ausgänge ausgewertet. Eine Impulse-Periode hat bei zwei Phasen (A und B) vier Flanken. Daher ist die Imuplsfrequenz am Ausgang auf ein viertel reduziert.

Der Impuls wird über den Ausgang "Signal A" und phasenverschoben über den Ausgang "Signal B" ausgegeben. Die Phasenverschiebung zwischen den Ausgängen definiert die Drehrichtung:

- Signal A um 90° voreilend gegenüber Signal B ⇒ positive Drehrichtung
- Signal B um 90° voreilend gegenüber Signal A ⇒ negative Drehrichtung

Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Werte gelten nicht für die Differenzausgänge der CPU 1217.

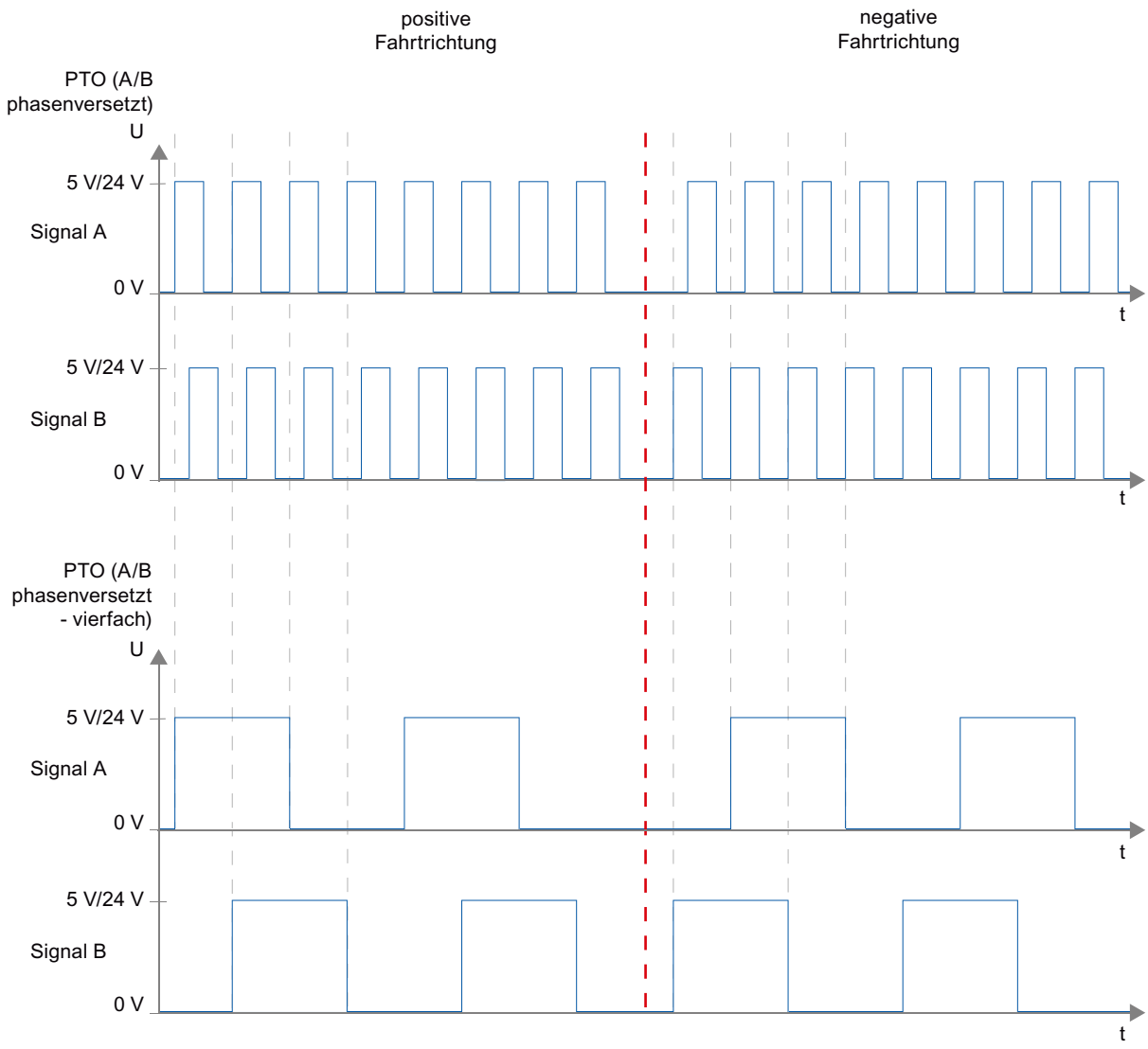

Richtungssinn invertieren

Wenn Sie die Option "Richtungssinn invertieren" aktivieren, wird die Richtungslogik invertiert:

- PTO – Impuls A und Richtung B
  - 0 V am Richtungsausgang (Low Pegel) ⇒ positive Drehrichtung
  - 5 V/24 V am Richtungsausgang (High Pegel) ⇒ negative Drehrichtung

  Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Spannungswerte gelten nicht für die Differenzausgänge der CPU 1217.
- PTO – vorwärts takten A und rückwärts takten B
  Die Ausgänge "Impulsausgang rückwärts" und "Impulsausgang vorwärts" werden vertauscht.

- PTO – A/B phasenversetzt
  Die Ausgänge "Signal A" und "Signal B" werden vertauscht.
- "PTO – A/B phasenversetzt, vierfach
  Die Ausgänge "Signal A" und "Signal B" werden vertauscht.

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Prinzip der Impulsschnittstelle (Seite 1347)

Hardware- und Software-Endschalter (Seite 1367)

Ruckbegrenzung (Seite 1368)

Referenzieren (Seite 1369)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

PROFIdrive-Antrieb/Analoge Antriebsanbindung

Antriebs- und Geberanschluss

Einer Positionierachse mit Antriebsanbindung über PROFIdrive / analoger Antriebsanbindung werden ein Antrieb und ein Geber zugeordnet.

PROFIdrive-fähige Antriebe werden über PROFIdrive-Telegramme angebunden. Der Sollwert wird über PROFIdrive-Telegramme vorgegeben.

Antriebe mit analoger Sollwertschnittstelle werden über einen Analogausgang und ein optionales Freigabesignal angebunden. Der Sollwert wird über einen Analogausgang vorgegeben.

Anschlussmöglichkeiten

PROFIdrive-fähige Antriebe werden über die PROFINET-Schnittstelle der CPU verbunden.

Antriebe mit analoger Sollwertschnittstelle werden über einen der folgenden Anschlüsse mit der CPU verbunden:

- Analogausgang über Signalboard
- Analogausgang Onboard
- Analogausgang über Analogausgabemodul

Für einen Geber sind folgende Anschlussmöglichkeiten möglich:

- Geber am PROFIdrive-Antrieb
- Geber am Technologiemodul

- PROFIdrive-Geber direkt am PROFINET IO
  (Bei diesen Gebern wird der Geberwert jeweils über PROFIdrive-Telegramme per PROFIBUS oder PROFINET übertragen.)
- Geber am schnellen Zähler (HSC - High Speed Counter)
  Bei dieser Anschlussmöglichkeit werden die Gebersignale direkt mit einem HSC verbunden, der daraus die Geberwerte bildet. Je nach eingesetzter CPU sind bis zu 6 HSC-Geber verwendbar.

Maximale Anzahl Achsen

Über PROFIdrive bzw. die analoge Antriebsanbindung lassen sich maximal 8 Antriebe ansteuern (die Anzahl ist unabhängig vom Simulationsstatus der Achse).

Automatische Übernahme der Antriebs- und Geberparameter im Gerät

Für den Betrieb müssen die Bezugsgrößen für die Antriebs- und Geberanbindung in der Steuerung und im Antrieb bzw. Geber identisch eingestellt sein.

Der Drehzahlsollwert NSOLL und der Drehzahlistwert NIST werden im PROFIdrive-Telegramm als Prozentwert bezogen auf die Bezugsdrehzahl übertragen. Der Bezugswert für die Drehzahl muss in der Steuerung und im Antrieb identisch eingestellt sein.

Die Auflösung des Istwertes im PROFIdrive-Telegramm muss ebenfalls in der Steuerung und im Antrieb bzw. Gebermodul identisch eingestellt sein.

Automatische Übernahme von Parametern

Bei SINAMICS-Antrieben ab V4.x und PROFIdrive-Gebern ab Ausgabestand A16 können die Antriebs-, bzw. Geberparameter automatisch in die CPU übernommen werden.

Die entsprechenden Parameter werden nach der (Neu-)Initialisierung des Technologieobjekts und (Wieder-)Anlauf des Antriebs und der CPU übernommen. Änderungen in der Konfiguration des Antriebs werden nach Wiederanlauf des Antriebs oder Restart des Technologieobjekts übernommen.

Die erfolgreiche Übernahme der Parameter kann in der Steuerung in den Variablen des Technologieobjekts <TO>.StatusDrive.AdaptionState = 1 und <TO>.StatusSensor[1].AdaptionState = 1 überprüft werden.

Parameter

Die Einstellungen der Steuerung werden im TIA Portal unter "Technologieobjekt > Konfiguration > Grundparameter > Antrieb/Geber" vorgenommen.

Die Einstellungen für Antrieb und Geber werden bei der Konfiguration der jeweiligen Hardware vorgenommen.

Folgende Tabelle stellt die Einstellungen im TIA-Portal, in der Steuerung und die entsprechenden Antriebs-/Geberparameter gegenüber:

Einstellung im TIA Portal	Steuerung Variable im Technologie-Datenbaustein	Antriebsparameter	Automatische Übernahme
Antrieb			
Telegrammnummer	Eingangsadresse Telegramm <TO>.Actor.Interface.AddressIn	Telegrammnummer P922	-
	Ausgangsadresse Telegramm <TO>.Actor.Interface.AddressOut		
Bezugsdrehzahl in [1/min]	<TO>.Actor.DriveParameter.Reference-Speed	SINAMICS-Antriebe: P2000	X
Maximale Drehzahl des Motors in [1/min]	<TO>.Actor.DriveParameter.MaxSpeed	SINAMICS-Antriebe: P1082	X
Antrieb	<TO>.Actor.Type 0 = Analoge Antriebsanbindung 1 = PROFIdrive 2 = PTO (Pulse Train Output)	-	-
Geber			
Telegramm	<TO>.Sensor[n].Interface.AddressIn	P922	-
	<TO>.Sensor[n].Interface.Addressout		
Gebertyp • Linear inkrementell • Linear absolut • Rotatorisch inkrementell • Rotatorisch absolut	<TO>.Sensor[n].System 0: Rotatorisch 1: Linear	P979[1] Bit0 Geber 1 P979[11] Bit0 Geber 2	X
	<TO>.Sensor[n].Type 0: Inkrementell 1: Absolut	P979[5] Geber 1 P979[15] Geber 2	-
Auflösung, linearer Geber Die Gitterteilung ist auf dem Typenschild des Gebers als Abstand der Striche auf dem linearen Messsystem angegeben.	<TO>.Sensor[n].Parameter.Resolution	P979[2] Geber 1 P979[12] Geber 2	X
Inkremente pro Umdrehung, rotatorischer Geber	<TO>.Sensor[n].Parameter.StepsPerRe-volution	P979[2] Geber 1 P979[12] Geber 2	X
Anzahl Bits für die Feinauflösung XIST1 Zyklischer Geberistwert, linearer oder rotatorischer Geber	<TO>.Sensor[n].Parameter.FineResoluti-onXist1	P979[3] Geber 1 P979[13] Geber 2	X
Anzahl Bits für die Feinauflösung XIST2 Absolutwert des Gebers, linearer oder rotatorischer Geber	<TO>.Sensor[n].Parameter.FineResoluti-onXist2	P979[4] Geber 1 P979[14] Geber 2	X
Unterscheidbare Geberumdrehungen, rotatorischer Absolutwertgeber	<TO>.Sensor[n].Parameter.Determinab-leRevolutions	P979[5] Geber 1 P979[15] Geber 2	X

Siehe auch

Konfiguration - Antrieb - PTO (Pulse Train Output) (Seite 1390)

Konfiguration - Antrieb - Analoge Antriebsanbindung (Seite 1393)

Konfiguration - Antrieb - PROFIdrive (Seite 1394)

Konfiguration - Geber - Geber am PROFINET/PROFIBUS (Seite 1396)

Konfiguration - Geber - Geber am schnellen Zähler (HSC) (Seite 1398)

PROFIdrive

PROFIdrive ist das genormte Standardprofil für Antriebstechnik bei der Anbindung von Antrieben und Gebern über PROFINET IO. Antriebe und Geber, die das PROFIdrive-Profil unterstützen, werden gemäß der PROFIdrive-Norm angebunden.

Die aktuelle PROFIdrive-Spezifikation finden Sie unter:

http://www.profibus.com (http://www.profibus.com)

Die Kommunikation zwischen Steuerung und Antrieb/Geber erfolgt über verschiedene PROFIdrive-Telegramme. Die Telegramme haben jeweils einen normierten Aufbau. Je nach Anwendung können Sie das passende Telegramm auswählen. In den PROFIdrive-Telegrammen werden Steuer- und Zustandsworte sowie Soll- und Istwerte übertragen.

Telegramme für PROFIdrive

Der Sollwert einer Positionierachse wird über PROFIdrive-Telegramm 1, 2, 3 oder 4 an einen Antrieb übertragen.

Der Geberwert wird entweder in einem Telegramm zusammen mit dem Sollwert (Telegramm 3 und 4) übertragen oder in einem separaten Gebertelegramm (Telegramm 81 oder 83).

Die folgenden Tabellen zeigen die unterstützten PROFIdrive-Telegramme für die Antriebs- und Geberzuordnung:

Standardtelegr-amme	Kurzbeschreibung
1	• Solldrehzahl 16 Bit (NSOLL) • Istdrehzahl 16 Bit (NIST)
2	• Solldrehzahl 32 Bit (NSOLL) • Istdrehzahl 32 Bit (NIST) • Lebenszeichen
3	• Solldrehzahl 32 Bit (NSOLL) • Istdrehzahl 32 Bit (NIST) • 1 Geber • Lebenszeichen
4	• Solldrehzahl 32 Bit (NSOLL) • Istdrehzahl 32 Bit (NIST) • 2 Geber • Lebenszeichen

Standardtelegr-amme für Geber	Kurzbeschreibung
81	• 1 Geber • Lebenszeichen
83	• Istdrehzahl 32 Bit (NIST) • 1 Geber • Lebenszeichen

Regelung

Antriebe, die über PROFIdrive oder einer analogen Antriebsschnittstelle angebunden sind, werden grundsätzlich lagegeregelt betrieben. Im Servicefall kann die Achse auch ohne Lageregelung betrieben werden.

Der Lageregler ist ein P-Regler mit Geschwindigkeitsvorsteuerung.

Reglerstruktur

Das folgende Bild zeigt die Reglerstruktur von S7-1200 Motion Control:

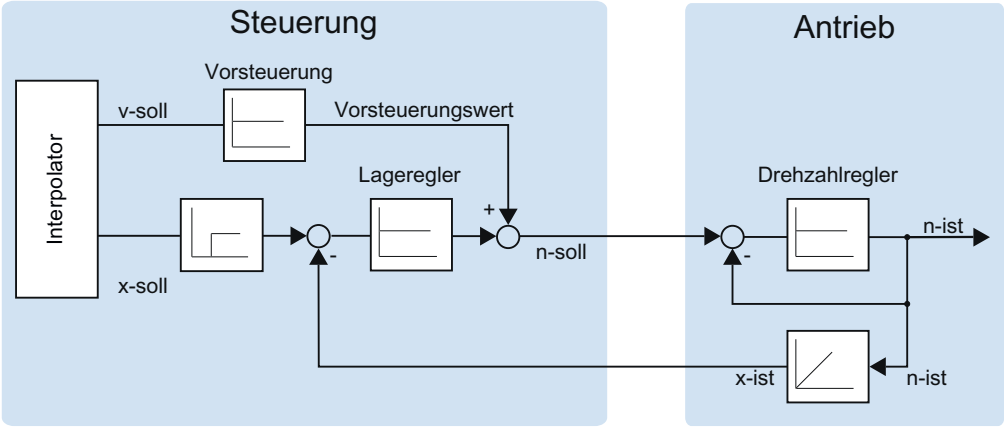

Der MC-Interpolator [OB92] berechnet die Sollposition für die Achse. Die Differenz von Soll- und Istposition wird mit dem Verstärkungsfaktor des Lagereglers multipliziert. Der resultierende Wert wird mit dem Vorsteuerungswert addiert und als Solldrehzahl über PROFIdrive oder Analogausgang an den Antrieb ausgegeben.

Der Geber erfasst die Istposition der Achse und gibt sie über ein PROFIdrive-Telegramm oder HSC (High Speed Counter) an die Steuerung zurück.

Datenanbindung PROFIdrive-Antrieb/PROFIdrive-Geber

Die Datenanbindung von PROFIdrive-Antrieben und PROFIdrive-Geber erfolgt entweder direkt über das PROFIdrive-Telegramm oder über einen Datenbaustein.

Nutzen Sie die Anbindung über Datenbaustein, wenn Sie im Anwenderprogramm prozessbedingt Telegramminhalte beeinflussen oder auswerten wollen.

Prinzip der Datenanbindung direkt zum Antrieb/Geber

Das nachfolgende vereinfachte Funktionsdiagramm zeigt die direkte Datenanbindung zu PROFIdrive-Antrieben und PROFIdrive-Gebern über Telegramme:

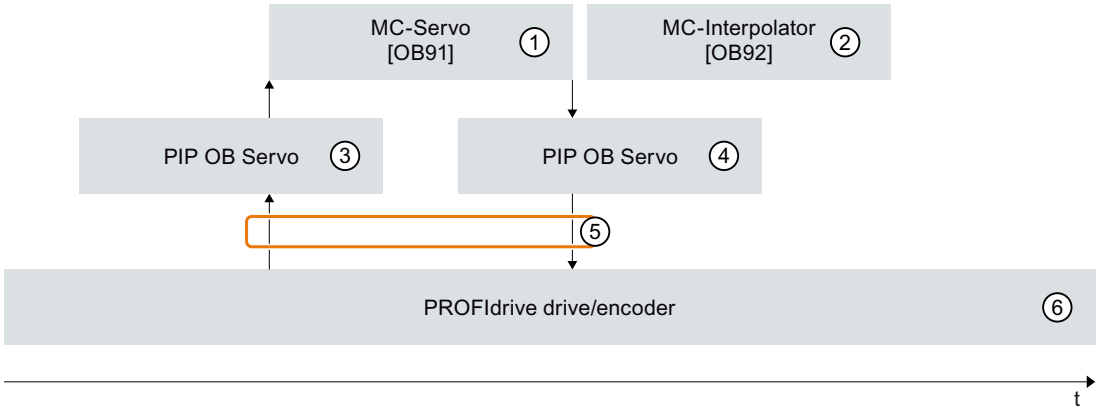

① Der Organisationsbaustein "MC-Servo" berechnet den Lageregler.

Zu Beginn des "MC-Servo" wird das Eingangstelegramm des Antriebs, bzw. Gebers gelesen (⑥ -> ⑤ -> ③). Wurde ein Organisationsbaustein "MC-PreServo" hinzugefügt, so wird das Telegramm zu Beginn des "MC-PreServo" gelesen.

Am Ende des "MC-Servo" wird das Ausgangstelegramm zum Antrieb, bzw. Geber geschrieben (④ -> ⑤ -> ⑥). Wurde ein Organisationsbaustein "MC-PostServo" hinzugefügt, so wird das Telegramm am Ende des "MC-PostServo" geschrieben.

② Der Organisationsbaustein "MC-Interpolator" wird in jedem Motion Applikationszyklus nach dem "MC-Servo" aufgerufen.

Im "MC-Interpolator" werden die Motion Control-Anweisungen ausgewertet, Sollwerte werden für den nächsten Motion Applikationszsyklus generiert und das Technologieobjekt überwacht.

③ Das Teilprozessabblid "TPA OB Servo" der Eingänge wird im Motion Applikationszyklus aktualisiert.

④ Das Teilprozessabblid "TPA OB Servo" der Ausgänge wird im Motion Applikationszyklus aktualisiert.

⑤ Telegrammaustausch über Peripherieadressen der Steuerung und des Antriebs, bzw. des Gebers.

⑥ PROFIdrive-Antrieb, bzw. PROFIdrive-Geber

Prinzip der Datenanbindung über Datenbaustein

Das nachfolgende vereinfachte Funktionsdiagramm zeigt die Datenanbindung zu PROFIdrive-Antrieben und PROFIdrive-Gebern über Datenbaustein. Details zur Ausführung folgen in den weiteren Abschnitten.

Um prozessbedingt Telegramminhalte beeinflussen oder auswerten zu können, wird ein Datenbaustein als Datenschnittstelle zwischengeschaltet (siehe ① und ③).

Verwenden Sie hierzu die Organisationsbausteine "MC-PreServo" und "MC-PostServo", um eine hohe Güte der Lageregelung zu erreichen.

Die Organisationsbausteine "MC-PreServo" und "MC-PostServo" können Sie in der Projektnavigation über den Befehl "Neuen Baustein hinzufügen" hinzufügen.

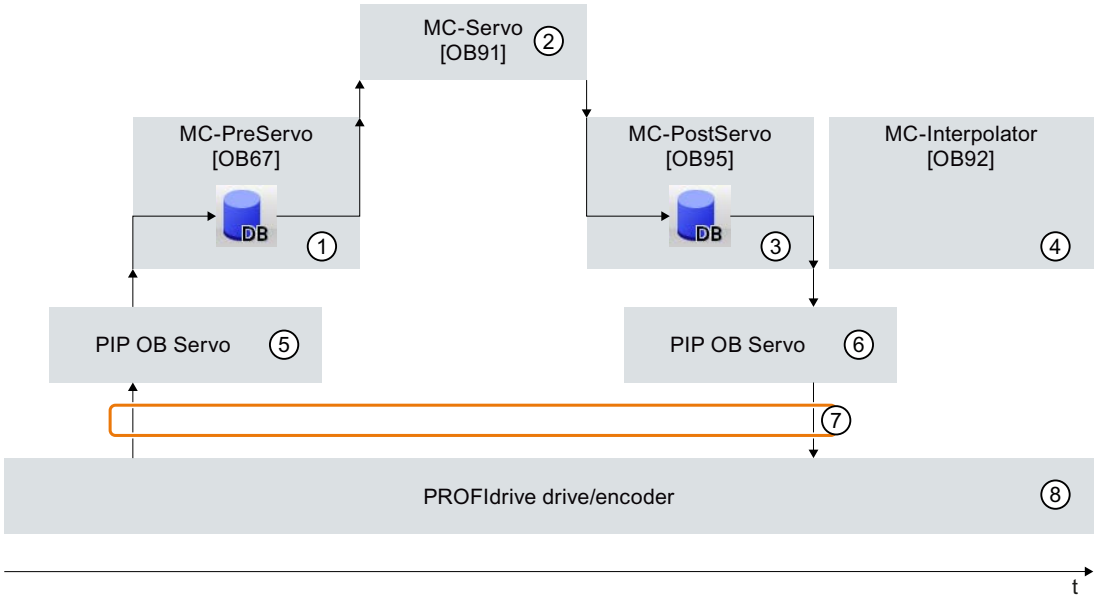

① Der "MC-PreServo" wird vor dem "MC-Servo" aufgerufen.

Übertragen Sie im Anwenderprogramm des "MC-PreServo" den Inhalt des Eingangstelegramms aus dem Teilprozessabbild "TPA OB Servo" ⑤ in den Datenbaustein der Datenschnittstelle.

Im weiteren Anwenderprogramm des "MC-PreServo" kann der Eingangsbereich des Telegramms bearbeitet, bzw. ausgewertet werden.

② Der Organisationsbaustein "MC-Servo" berechnet den Lageregler.

Zu Beginn des "MC-Servo" wird das Eingangstelegramm des Antriebs, bzw. Gebers aus dem Datenbaustein der Datenschnittstelle gelesen (① -> ②).

Am Ende des "MC-Servo" wird das Ausgangstelegramm des Antriebs, bzw. Gebers in den Datenbaustein der Datenschnittstelle geschrieben (② -> ③).

③ Der "MC-PostServo" wird nach dem "MC-Servo" aufgerufen.

Im Anwenderprogramm des "MC-PostServo" kann der Ausgangsbereich des Telegramms bearbeitet, bzw. ausgewertet werden.

Übertragen Sie am Ende des Anwenderprogramms des "MC-PostServo" den Inhalt des Ausgangstelegramms aus der Datenschnittstelle des Datenbausteins in das Teilprozessabbild "TPA OB Servo" ⑥.

④ Der Organisationsbaustein "MC-Interpolator" wird in jedem Motion Applikationszyklus nach dem "MC-PostServo" aufgerufen.

Im "MC-Interpolator" werden die Motion Control-Anweisungen ausgewertet, Sollwerte werden für den nächsten Motion Applikationszsyklus generiert und das Technologieobjekt überwacht.

⑤ Das Teilprozessabblid "TPA OB Servo" der Eingänge wird im Motion Applikationszyklus aktualisiert.

⑥ Das Teilprozessabblid "TPA OB Servo" der Ausgänge wird im Motion Applikationszyklus aktualisiert.

⑦ Telegrammaustausch über Peripherieadressen der Steuerung und des Antriebs, bzw. des Gebers.

⑧ PROFIdrive-Antrieb, bzw. PROFIdrive-Geber

Prinzipielle Vorgehensweise zur Datenanbindung über Datenbaustein

Gehen Sie wie nachfolgend beschrieben vor, um die Datenanbindung über Datenbaustein zu nutzen. Die Datenanbindung kann für PROFIdrive-Antrieb und Geber getrennt projektiert werden.

Datenbaustein für Datenanbindung erstellen

Der Datenbaustein für die Datenanbindung muss anwenderseitig erstellt werden. Der Datenbaustein muss für die Datenanbindung eine Datenstruktur vom Datentyp "PD_TELx" beinhalten. "x" steht für die in der Gerätekonfiguration konfigurierte Telegrammnummer des Antriebs, bzw. Gebers.

Gehen Sie zum Erstellen des Datenbausteins wie nachfolgend beschrieben vor:

1. Erstellen Sie einen neuen Datenbaustein vom Typ "Global-DB".
2. Markieren Sie den Datenbaustein in der Projektnavigation und wählen Sie den Kontextmenübefehl "Eigenschaften"
3. Deaktivieren Sie unter Attribute folgende Attribute und übernehmen Sie die Änderung mit OK
   "Nur im Ladespeicher ablegen"
   "Datenbaustein im Gerät schreibgeschützt"
   "Optimierter Bausteinzugriff"
4. Öffnen Sie den Datenbaustein im Bausteineditor.
5. Fügen Sie im Bausteineditor textlich eine Variable vom Typ "PD_TELx" ein.
6. Übersetzen Sie den Datenbaustein für die Datenanbindung, bevor Sie diesen in der Konfiguration der Achsen verwenden.

In dieser Variable finden Sie die Variablenstruktur "Input" für den Eingangsbereich des Telegramms und "Output" für den Ausgangsbereich des Telegramms.

Hinweis

"Input" und "Output" beziehen sich auf die Sicht der Lageregelung. Der Eingangsbereich beinhaltet z. B. die Istwerte des Antriebs, der Ausgangsbereich die Sollwerte für den Antrieb.

Der Datenbaustein darf die Datenstrukturen mehrerer Achsen, Geber und weitere Inhalte beinhalten.

Datenanbindung über Datenbaustein konfigurieren

Gehen Sie in der Konfiguration der Achse wie nachfolgend beschrieben vor:

1. Rufen Sie das Konfigurationsfenster Hardware-Schnittstelle > Antrieb, bzw. Geber auf.
2. Wählen Sie in der Klappliste Datenbaustein "Datenbaustein" aus.
3. Wählen Sie im Feld "Datenbaustein" den zuvor erstellten Datenbaustein aus.
   Öffnen Sie diesen und wählen Sie den für Antrieb und Geber definierten Variablennamen aus.

PLC-Variable für Telegrammzugriff hinzufügen

Um auf den Eingangs- und Ausgangsbereich des Telegramms zugreifen zu können, erzeugen Sie die folgende PLC-Variable.

Gehen Sie für die PLC-Variable des Eingangsbereichs wie nachfolgend beschrieben vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" und lassen Sie sich alle Variablen anzeigen.
2. Fügen Sie eine neue Variable hinzu und vergeben Sie einen eindeutigen Namen, z. B. bestehend aus dem Namen der Achse bzw. des Gebers, dem Telegrammtyp und dem Adressbereich.
3. Tragen Sie in der Spalte "Datentyp" textlich den Typ "PD_TELx_IN" ein.
4. Tragen Sie in der Spalte "Adresse" die Telegrammeingangsadresse des Antriebs/Gebers ein.
   Die Adresse finden Sie in der Gerätekonfiguration des Antriebs, bzw. Gebers.

Gehen Sie analog für die PLC-Variable des Ausgangsbereichs vor und wählen Sie für den Datentyp "PD_TELx_OUT" und als Adresse die Telegrammausgangsadresse des Antriebs/Gebers.

MC-PreServo und MC-PostServo programmieren

MC-PreServo

Das Anwenderprogramm des "MC-PreServo" muss den Eingangsbereich des Telegramms lesen und in den Datenbaustein der Datenanbindung übertragen.

Weisen Sie hierzu im Anwenderprogramm des "MC-PreServo" die zuvor definierten PLC-Variable des Eingangsbereichs der Variablenstruktur "Input" des Datenbausteins zu.

Mit weiteren Anweisungen können Sie die Daten der Variablenstruktur "Input" des Datenbaustein bearbeiten, bevor diese dem "MC-Servo" übergeben und vom "MC_Servo" verarbeitet werden.

MC-PostServo

Am Ende seiner Bearbeitung überträgt der "MC-Servo" den Ausgangsbereich des Telegramms in die Variablenstruktur "Output" des Datenbausteins.

Im Anwenderprogramm des "MC-PostServo" muss der Inhalt der Variablenstruktur "Output" des Datenbausteins in die Telegrammausgangsadresse geschrieben werden.

Weisen Sie hierzu im Anwenderprogramm des "MC-PostServo" die Variablenstruktur "Output" des Datenbausteins der zuvor definierten PLC-Variable des Ausgangsbereichs zu.

Soll der Ausgangsbereich modifiziert werden, so muss dies vor der Zuweisungsanweisung erfolgen.

ACHTUNG
Maschinenschäden Die unsachgemäße Manipulation der Antriebs- und Gebertelegramme kann zu unerwünschten Bewegungen des Antriebs führen. Prüfen Sie Ihr Anwenderprogramm auf Konsistenz in der Antriebs- und Geberanbindung.

Ein Anwendungsbeispiel für die Verwendung von MC-Pre- und MC-PostServo finden Sie unter:

https://support.industry.siemens.com/cs/document/109741575 (https://support.industry.siemens.com/cs/document/109741575)

Siehe auch

PROFIdrive-Telegramm (Seite 1503)

Konfiguration - Antrieb - PROFIdrive (Seite 1394)

Konfiguration - Geber - Geber am PROFINET/PROFIBUS (Seite 1396)

Datenanbindung Antrieb mit analoger Antriebsanbindung (Seite 1361)

Organisationsbausteine für Motion Control (Seite 1363)

Datenanbindung Antrieb mit analoger Antriebsanbindung

Die Datenanbindung von Antrieben mit analoger Antriebsschnittstelle kann alternativ über einen Datenbaustein erfolgen.

Nutzen Sie die Anbindung über Datenbaustein, wenn Sie im Anwenderprogramm prozessbedingt den analogen Sollwert anpassen wollen.

Prinzip der Datenanbindung über Datenbaustein

Am Ende der Lageregelung durch den MC-Servo [OB91] wird der Sollwert des analogen Antriebs auf den zugeordneten Analogausgang geschrieben.

Um prozessbedingt den analogen Sollwert anpassen zu können, wird eine Datenschnittstelle über Datenbaustein zwischengeschaltet.

Der Sollwert des analogen Antriebs kann über den Organisationsbaustein MC-PostServo [OB95] im Datenbaustein bearbeitet werden und anschließend in die Peripherieadresse geschrieben werden.

Der MC-PostServo wird nach dem MC-Servo aufgerufen. Der Organisationsbaustein MC-PostServo ist anwenderseitig programmierbar und muss über den Befehl "Neuen Baustein hinzufügen" hinzugefügt werden.

Prinzipielle Vorgehensweise

Gehen Sie wie nachfolgend beschrieben vor, um die Datenanbindung über Datenbaustein zu nutzen. Die Datenanbindung kann für Antriebe mit analoger Antriebsschnittstelle und PROFIdrive-Geber getrennt projektiert werden. Informationen zur Datenanbindung der PROFIdrive Geber entnehmen Sie dem Kapitel Datenanbindung PROFIdrive-Antrieb/ PROFIdrive-Geber (Seite 1356).

Datenbaustein für Datenanbindung erstellen

Der Datenbaustein muss anwenderseitig erstellt werden.

Gehen Sie zum Erstellen des Datenbausteins wie nachfolgend beschrieben vor:

1. Erstellen Sie einen neuen Datenbaustein vom Typ "Global-DB".
2. Markieren Sie den Datenbaustein in der Projektnavigation und wählen den Kontextmenübefehl "Eigenschaften"
3. Deaktivieren Sie unter Attribute folgende Attribute und übernehmen Sie die Änderung mit OK
   "Nur im Ladespeicher ablegen"
   "Datenbaustein im Gerät schreibgeschützt"
   "Optimierter Bausteinzugriff"
4. Öffnen Sie den Datenbaustein im Bausteineditor.
5. Fügen Sie im Bausteineditor eine Variable vom Datentyp "WORD" ein.
6. Übersetzen Sie den Datenbaustein für die Datenanbindung, bevor Sie diesen in der Konfiguration der Achsen verwenden.

Datenanbindung über Datenbaustein konfigurieren

Gehen Sie in der Konfiguration wie nachfolgend beschrieben vor (im Konfigurationsfenster Grundparameter > Allgemein muss die "Analoge Antriebsabindung" gewählt sein):

1. Rufen Sie das Konfigurationsfenster Grundparameter > Antrieb auf.
2. Wählen Sie im Feld "Analogausgang" die zuvor definierte Variable des Datenbausteins aus.

Analogausgangsadresse in das Prozessabbild TPA OB Servo legen

Um eine ausreichende Regelgüte zu erreichen, muss der Adressbereich des Analogausgangs im Prozessabblild "TPA OB Servo" liegen.

Gehen Sie wie nachfolgend beschrieben vor:

1. Öffnen Sie in der Gerätekonfiguration die Baugruppe des Analogausgangs.
2. Öffnen Sie das Register "Allgemein"
3. Wählen Sie "E/A-Adressen"
4. Wählen Sie als Organisationsbaustein den "MC-Servo". Als Prozessabbild wild automatisch "TPA OB Servo" gewählt.
5. Weisen Sie im Register "I/O-Variablen" dem Analogausgang noch einen Variablennamen zu.

MC-PostServo programmieren

Weisen Sie im Anwenderprogramm des MC-PostServo der Variable des Analogausgangs die Variable des Datenbausteins zu.

Am Ende des MC-PostServo wird der Ausgangsbereich des "TPA OB Servo" in die Peripherie geschrieben.

⚠ WARNUNG
Die unsachgemäße Manipulation des Antriebssollwerts kann Mensch und Maschine gefährden Treffen Sie ausreichende Vorsichtsmaßnahmen, um die Gefährdung von Mensch und Maschine zu verhindern.

Ablaufverhalten

Organisationsbausteine für Motion Control

Beschreibung

Wenn Sie ein Technologieobjekt "Positionierachse" mit PROFIdrive-Antrieb oder mit analoger Antriebsschnittstelle anlegen, werden automatisch Organisationsbausteine für die Bearbeitung der Technologieobjekte angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Motion Control-Applikationszyklus aufgerufen.

Folgende Organisationsbausteine werden angelegt:

- MC-Servo [OB91]
  Berechnung des Lagereglers
- MC-Interpolator [OB92]
  Auswertung der Motion Control-Anweisungen, Sollwertgenerierung und Überwachungsfunktionalität

Die Organisationsbausteine sind geschützt (Know-how-Schutz). Der Programmcode kann nicht eingesehen oder verändert werden.

Das Taktverhältnis der beiden Organisationsbausteine zueinander ist immer 1:1. MC-Servo [OB91] wird immer vor MC-Interpolator [OB92] ausgeführt.

Sie können den Motion Control-Applikationszyklus und die Priorität der Organisationsbausteine entsprechend Ihren Anforderungen an die Regelgüte und die Systembelastung einstellen.

Motion Control-Applikationszyklus

Den Motion Control-Applikationszyklus, in dem der MC-Servo [OB91] aufgerufen wird, können Sie in den Eigenschaften des Organisationsbausteins unter "Allgemein > Zykluszeit" einstellen.

Der MC-Servo [OB91] wird zyklisch mit dem angegebenen "Applikationszyklus" aufgerufen.

Den Motion Control-Applikationszyklus müssen Sie groß genug wählen, um alle Technologieobjekte in einem Zyklus bearbeiten zu können. Wenn die Bearbeitungszeit der Technologieobjekte größer ist als der Applikationszyklus, treten Überläufe (Seite 1365) auf.

Um Störungen im Programmablauf auf der CPU zu vermeiden, stellen Sie den Motion Control-Applikationszyklus abhängig von der Anzahl der eingesetzten Achsen folgendermaßen ein:

Motion Control-Applikationszyklus = (Anzahl der lagegeregelten Achsen × 2 ms) + 2 ms

Folgende Tabelle zeigt den resultierenden Motion Control-Applikationszyklus beispielhaft entsprechend der Anzahl der lagegeregelten Achsen:

Anzahl der Achsen	Motion Control-Applikationszyklus
1	4 ms
2	6 ms
4	10 ms
8	18 ms

Für SINAMICS sollte weiterhin gelten:

- Motion Control-Applikationszyklus (MC-Servo) ≥ SINAMICS Antriebsprozessabbild (Parameter P2048) ≥ Bustakt

Alle Zeiten sollten zueinander als ganzzahliges Vielfaches gewählt werden.

Priorität

Die Priorität der Organisationsbausteine können Sie bei Bedarf in deren Eigenschaften unter "Allgemein > Attribute > Priorität" einstellen:

- MC-Servo [OB91]
  Priorität 17 bis 26 (Default-Wert 25)
- MC-Interpolator [OB92]
  Priorität 17 bis 26 (Default-Wert 24)

Die Priorität des MC-Servo [OB91] muss mindestens um eins höher sein als die Priorität des MC-Interpolator [OB92].

MC-PreServo [OB67] und MC-PostServo [OB95]

Die Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] sind programmierbar und werden im konfigurierten Applikationszyklus aufgerufen. MC-PreServo [OB67] wird direkt vor dem MC-Servo [OB91] aufgerufen. MC-PostServo [OB95] wird direkt nach dem MC-Servo [OB91] aufgerufen.

Siehe auch

Datenanbindung PROFIdrive-Antrieb/PROFIdrive-Geber (Seite 1356)

Teilprozessabbild "TPA OB Servo"

Ordnen Sie für eine optimale Regelung alle von Motion Control verwendeten E/A-Baugruppen (z. B. Antriebe, Technologiemodule, digitale und analoge Ein-/Ausgabebaugruppen) dem Teilprozessabbild "TPA OB Servo" zu. Durch die Zuordnung werden die E/A-Baugruppen zeitlich synchron zum Technologieobjekt bearbeitet.

Ablaufverhalten und Überläufe

Bei der Bearbeitung der Motion Control-Funktionalität werden in jedem Applikationszyklus die Organisationsbausteine MC-Servo [OB91] und MC-Interpolator [OB92] aufgerufen und bearbeitet (die Abarbeitung erfolgt auch im Betrieszustand STOP der CPU). Die restliche Zykluszeit steht für die Bearbeitung Ihres Anwenderprogramms zur Verfügung.

Für einen fehlerfreien Programmablauf gelten folgende Regeln:

- In jedem Applikationszyklus muss der MC-Servo [OB91] gestartet und vollständig abgearbeitet werden.
- In jedem Applikationszyklus muss der zugehörige MC-Interpolator [OB92] zumindest gestartet werden.

Folgendes Bild zeigt das fehlerfreie Ablaufverhalten beispielhaft für die Abarbeitung des Organisationsbausteins OB1:

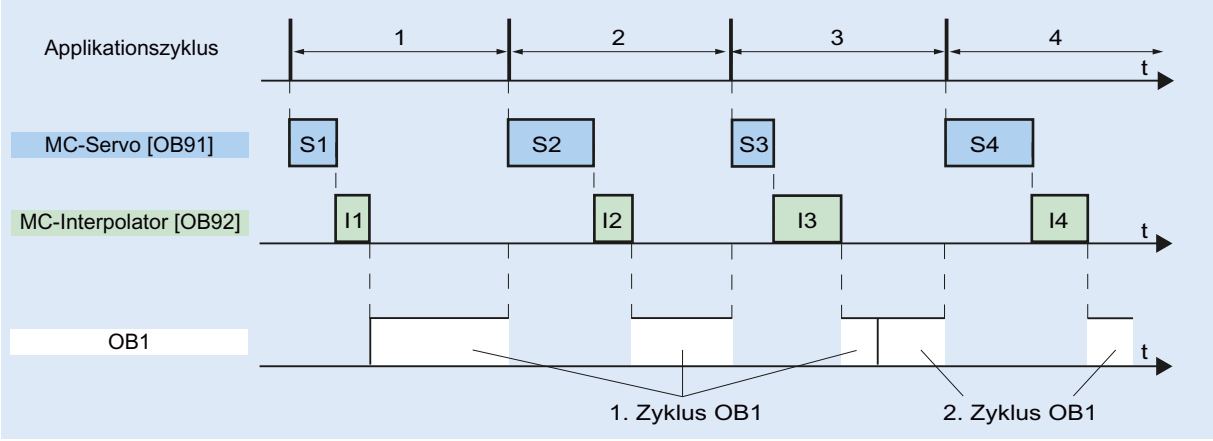

Überläufe

Wenn der eingestellte Applikationszyklus nicht eingehalten wird, z. B. aufgrund eines zu kurzen Applikationszyklus, können Überläufe auftreten.

Ein Überlauf des MC-Servo [OB91], MC-Interpolator [OB92], MC_PreServo [OB67] und MC_PostServo [OB95] wird im Diagnosepuffer der CPU eingetragen und führt zum STOP der CPU.

MC-PreServo, MC-Servo, MC-PostServo und MC-Interpolator werden gestoppt. Bei Bedarf können Sie den Eintrag im Diagnosepuffer über einen Zeitfehler-OB (OB 80) auswerten.

Betriebszustände

In diesem Abschnitt wird das Verhalten von Motion Control in den jeweiligen Betriebszuständen und bei Übergängen zwischen den Betriebszuständen betrachtet. Eine allgemeine Beschreibung der Betriebszustände finden Sie im Systemhandbuch S7-1200.

Betriebszustände und Übergänge

Die CPU hat drei Betriebszustände: STOP, ANLAUF (STARTUP) und RUN. Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge:

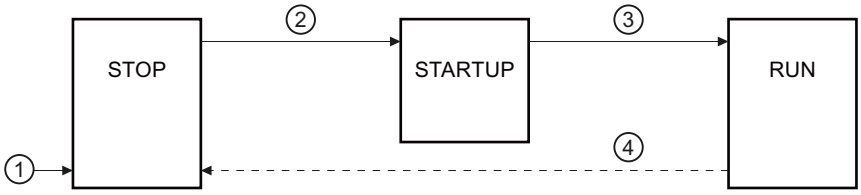

① NETZ-EIN → STOP
② STOP → ANLAUF
③ ANLAUF → RUN
④ RUN → STOP

Betriebszustand STOP

Im Betriebszustand STOP wird das Anwenderprogramm nicht bearbeitet und alle Prozessausgänge werden deaktiviert. Somit werden keine Motion Control-Aufträge ausgeführt.

Die Technologie-Datenbausteine der lagegeregelten Achsen werden aktualisiert.

Betriebszustand ANLAUF (STARTUP)

Bevor die CPU mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, werden die Anlauf-OBs einmalig abgearbeitet.

Im ANLAUF sind die Prozessausgänge gesperrt. Motion Control-Aufträge werden abgewiesen.

Die Technologie-Datenbausteine der lagegeregelten Achsen werden aktualisiert.

Betriebszustand RUN

Im Betriebszustand RUN wird das Anwenderprogramm abgearbeitet.

Im RUN werden die im OB1 programmierten Motion Control-Aufträge zyklisch aufgerufen und abgearbeitet (andere Ablaufebenen sind möglich).

Die Technologie-Datenbausteine werden aktualisiert.

Betriebzustandsübergänge

Die folgende Tabelle zeigt das Verhalten von Motion Control bei den Übergängen zwischen den Betriebszuständen:

Betriebszustandsüber-gang	Verhalten
NETZ-EIN → STOP	Die CPU führt einen Restart der Technologieobjekte durch. Die Technologieobjekte werden mit den Werten aus dem Ladespeicher neu initialisiert.
STOP → ANLAUF	Die Technologieobjekte werden auf die Startwerte der CPU initialisiert.
ANLAUF → RUN	Die Prozessausgänge werden frei gegeben.
RUN → STOP	Wenn die CPU vom Betriebszustand RUN nach STOP wechselt, werden alle Technologieobjekte gemäß der Fehlerreaktion "Freigabe wegnehmen" gesperrt. Laufende Motion Control-Aufträge werden abgebrochen.

Hardware- und Software-Endschalter

Begrenzen Sie mit Hardware- und Software-Endschaltern den "zulässigen Verfahrbereich" und den "Arbeitsbereich" ihres Technologieobjekts Positionierachse. Die Zusammenhänge können Sie der nachfolgenden Darstellung entnehmen:

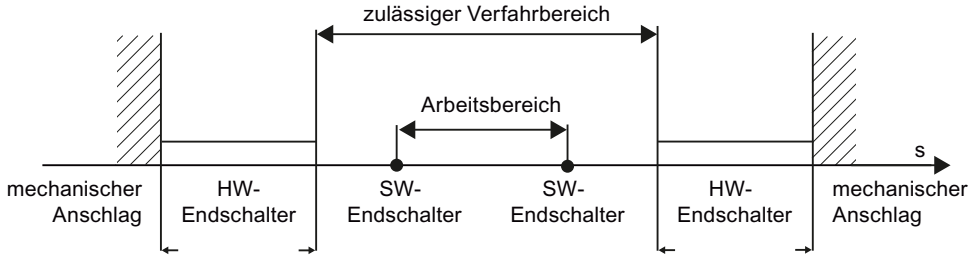

Bei den Hardware-Endschaltern handelt es sich um Endlagenschalter, welche den maximal "zulässigen Verfahrbereich" der Achse begrenzen. Hardware-Endschalter sind physikalische Schaltelemente, die an alarmfähigen Eingängen der CPU angeschlossen werden müssen.

Mit Software-Endschaltern wird der "Arbeitsbereich" der Achse begrenzt. Sie sollten bezogen auf den Verfahrbereich innerhalb der Hardware-Endschalter liegen. Da die Positionen der Software-Endschalter flexibel eingestellt werden können, kann der Arbeitsbereich der Achse je nach aktuellem Verfahrprofil individuell angepasst werden. Im Gegensatz zu Hardware-Endschaltern werden Software-Endschalter nur über die Software realisiert und benötigen keine eigenen Schaltelemente.

Hardware- und Software-Endschalter müssen vor der Verwendung in der Konfiguration, bzw. im Anwenderprogramm aktiviert werden. Software-Endschalter sind erst nach dem Referenzieren der Achse wirksam.

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Prinzip der Impulsschnittstelle (Seite 1347)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Ruckbegrenzung (Seite 1368)

Referenzieren (Seite 1369)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Positionsgrenzen (Seite 1403)

Ruckbegrenzung

Mit der Ruckbegrenzung reduzieren Sie die Belastungen ihrer Mechanik während einer Beschleunigungs- und Verzögerungsrampe. Der Wert für die Beschleunigung und Verzögerung wird bei aktiver Ruckbegrenzung nicht abrupt geändert, sondern sanft auf- und abgebaut. Das nachfolgende Bild zeigt den Verlauf von Geschwindigkeit und Beschleunigung ohne und mit Ruckbegrenzung:

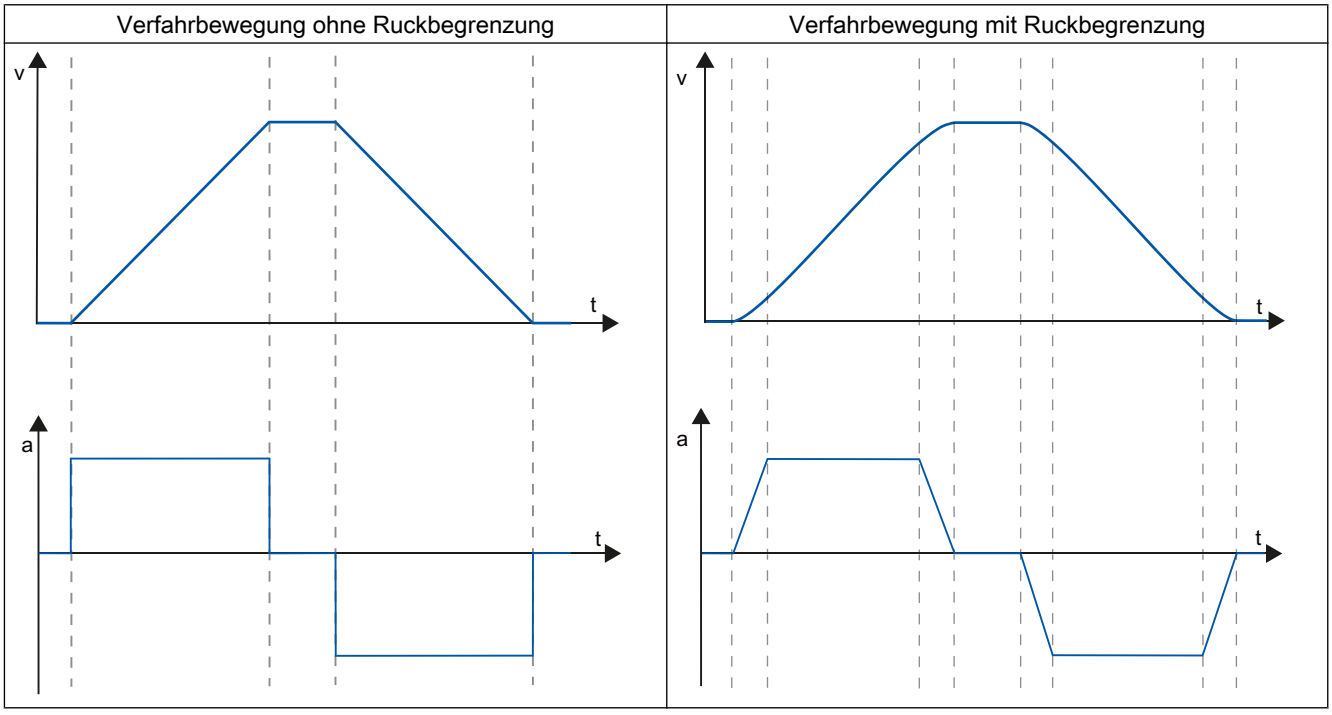

Mit der Ruckbegrenzung ergibt sich ein "verrundetes" Geschwindigkeitsprofil der Achsbewegung. Hiermit ist z. B. ein sanftes Anfahren und Abbremsen eines Förderbandes gewährleistet.

Siehe auch

Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 1413)

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Prinzip der Impulsschnittstelle (Seite 1347)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Hardware- und Software-Endschalter (Seite 1367)

Referenzieren (Seite 1369)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Referenzieren

Unter Referenzieren versteht man das Abgleichen der Achskoordinate des Technologieobjekts auf die reale, physikalische Position des Antriebs. Da sich bei positionsgesteuerten Achsen die Eingaben und Anzeigen zur Position auf genau diese Achskoordinate beziehen, ist eine Übereinstimmung zur realen Situation von äußerster Wichtigkeit. Nur so kann gewährleistet werden, dass die absolute Zielposition der Achse auch am Antrieb exakt erreicht wird.

Bei der CPU S7-1200 erfolgt das Referenzieren der Achse mit der Motion Control-Anweisung "MC_Home". Der Status "Referenziert" wird in der Variablen des Technologieobjekts <Achsname>.StatusBits.HomingDone angezeigt. Es wird zwischen den folgenden Referenziermodi unterschieden:

Referenziermodi

- Aktives Referenzieren
  Beim aktiven Referenzieren führt die Motion Control-Anweisung "MC_Home" die notwendige Referenzpunktfahrt durch. Beim Erkennen des Referenzpunktschalters wird die Achse entsprechend der Konfiguration referenziert. Laufende Verfahrbewegungen werden abgebrochen.
- Passives Referenzieren
  Beim passiven Referenzieren führt die Motion Control-Anweisung "MC_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse entsprechend der Konfiguration referenziert. Laufende Verfahrbewegungen werden beim Start des passiven Referenzierens nicht abgebrochen.
- Direktes Referenzieren Absolut
  Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Der Wert des Eingangsparameters "Position" der Motion Control-Anweisung "MC_Home" wird sofort als Referenzpunkt der Achse gesetzt.
- Direktes Referenzieren Relativ
  Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Für die Achsposition nach dem Referenzieren gilt:
  Neue Achsposition = Aktuelle Achsposition + Wert des Parameters "Position" der Anweisung "MC_Home".

Rücksetzen des Status "Referenziert"

Der Status "Referenziert" eines Technologieobjekts (<Achsname>.StatusBits.HomingDone) wird unter folgenden Bedingungen rückgesetzt:

- Antriebsanbindung über PTO (Pulse Train Output):
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Sperren der Achse durch die Motion Control-Anweisung "MC_Power"
  - Wechsel zwischen Automatikbetrieb und Handsteuerung
  - Nach NETZ-AUS -> NETZ-EIN der CPU
  - Nach Neustart der CPU (RUN-STOP -> STOP-RUN)
- Technologieobjekte mit inkrementellen Istwerten:
  - Starten eines "MC_Home"-Auftrags zum aktiven Referenzieren
    (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
  - Fehler im Gebersystem bzw. Geberausfall
  - Restart des Technologieobjekts
  - Nach NETZ-AUS → NETZ-EIN der CPU
  - Urlöschen
  - Veränderung der Geberkonfiguration
- Technologieobjekte mit absoluten Istwerten:
  - Fehler im Sensorsystem/Geberausfall
  - Austausch der CPU
  - Veränderung der Geberkonfiguration
  - Wiederherstellen der CPU-Werkseinstellung
  - Übertragen eines anderen Projekts in die Steuerung

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Prinzip der Impulsschnittstelle (Seite 1347)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Hardware- und Software-Endschalter (Seite 1367)

Ruckbegrenzung (Seite 1368)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Referenzieren (Technologieobjekt Positionierachse ab V2) (Seite 1416)

6.1.1.3 Leitfaden zum Einsatz von Motion Control

Der hier beschriebene Leitfaden zeigt die grundsätzliche Vorgehensweise, um Motion Control mit der CPU S7-1200 einzusetzen.

Voraussetzung

Um das Technologieobjekt Positionierachse einzusetzen, muss ein Projekt mit einer CPU S7-1200 angelegt sein.

Vorgehen

Gehen Sie in der nachfolgend empfohlenen Reihenfolge vor, um Motion Control mit der CPU S7-1200 einzusetzen. Folgen Sie hierzu den aufgelisteten Links:

1. Technologieobjekt Positionierachse hinzufügen (Seite 1386)
2. Arbeiten mit dem Konfigurationsdialog (Seite 1387)
3. Laden in CPU (Seite 1469)
4. Funktionstest der Achse im Inbetriebnahmefenster (Seite 1470)
5. Programmieren (Seite 1477)
6. Achssteuerung diagnostizieren (Seite 1499)

6.1.1.4 Versionen einsetzen

Versionsübersicht

Die Zusammenhänge der relevanten Versionen für S7-1200 Motion Control können Sie der nachfolgenden Aufstellung entnehmen:

Version der Technologie

Die aktuell ausgewählte Version der Technologie können Sie in der Task Card "Anweisungen" im Ordner "Technologie > Motion Control" und im Dialog "Technologieobjekte > Neues Objekt hinzufügen" überprüfen.

Die Version der Technologie wählen Sie in der Task Card "Anweisungen" im Ordner "Technologie > Motion Control".

Wird im Dialog "Neues Objekt hinzufügen" ein Technologieobjekt mit einer alternativen Version hinzugefügt, so wird die Version der Technologie ebenfalls umgeschaltet.

Hinweis

Die Auswahl einer alternativen Version der Technologie betrifft auch die Version der Motion Control-Anweisungen (Task Card).

Die Technologieobjekte und Motion Control-Anweisungen werden erst beim Übersetzen, bzw. beim "Laden in Gerät" entsprechend der gewählten Version der Technologie konvertiert.

Version des Technologieobjekts

Die Version eines Technologieobjekts kann im Inspektorfenster im Register "Eigenschaften > Allgemein > Information" im Feld "Version" überprüft werden.

Version der Motion Control-Anweisung

Die Version der Motion Control-Anweisung kann im Inspektorfenster im Register "Eigenschaften > Allgemein > Information" im Feld "Version" überprüft werden.

Entspricht die eingesetzte Version der Motion Control-Anweisung nicht der nachfolgenden Kompatibilitätsliste, so werden die entsprechenden Motion Control-Anweisungen im Programmeditor markiert.

Kompatibilitätsliste

Technologie		CPU	Technologieobjekt	Motion Control-Anweisung
V6.0	Neuerungen: • MC-PreServo • MC-PostServo • Datenanbindung direkt zum SINAMICS-Antrieb oder über Datenbaustein • Datenanbindung direkt zum Analogausgang eines Antriebs mit analoger Antriebsanbindung, oder über Datenbaustein • Übernahme der Antriebs- und Geberparameter im Gerät bei PROFIdrive-Antrieben und Gebern • Lagegeregelte Antriebe ohne Lageregelung zu Servicezwecken bewegen • Simulation lagegeregelter Antriebe ohne projektierte, bzw. vorhandene Hardware • Pegelauswahl bei der Konfiguration der Hardware-Endschalter • Unterstützung PROFIdrive Telegramm 4	V4.2	Positionierachse V6.0 Auftragstabelle V6.0	MC_Power V6.0 MC_Reset V6.0 MC_Home V6.0 MC_Halt V6.0 MC_MoveAbsolute V6.0 MC_MoveRelative V6.0 MC_MoveVelocity V6.0 MC_MoveJog V6.0 MC_CommandTable V6.0 MC_ChangeDynamic V6.0 MC_ReadParam V6.0 MC_WriteParam V6.0

Technologie		CPU	Technologieobjekt	Motion Control-Anweisung
V5.0	Neuerungen: • Antriebsanbindung über PROFIdrive • Analoge Antriebsanbindung • Lageregelung für PROFIdrive / analoge Antriebsanbindung • Positionierüberwachung für PROFIdrive / analoge Antriebsanbindung • MC-Servo [OB91] • MC-Interpolator [OB92]	V4.1	Positionierachse V5.0 Auftragstabelle V5.0	MC_Power V5.0 MC_Reset V5.0 MC_Home V5.0 MC_Halt V5.0 MC_MoveAbsolute V5.0 MC_MoveRelative V5.0 MC_MoveVelocity V5.0 MC_MoveJog V5.0 MC_CommandTable V5.0 MC_ChangeDynamic V5.0 MC_ReadParam V5.0 MC_WriteParam V5.0
V4.0	Neuerungen: • MC_ReadParam • MC_WriteParam • Vereinheitlichung der S7-1200 und S7-1500 Motion Control Technologie-Datenbausteine.	V4.0	Positionierachse V4.0 Auftragstabelle V4.0	MC_Power V4.0 MC_Reset V4.0 MC_Home V4.0 MC_Halt V4.0 MC_MoveAbsolute V4.0 MC_MoveRelative V4.0 MC_MoveVelocity V4.0 MC_MoveJog V4.0 MC_CommandTable V4.0 MC_ChangeDynamic V4.0 MC_ReadParam V4.0 MC_WriteParam V4.0
V3.0	Neuerung: Laden im Betriebszustand RUN	V2.2 V3.0 V4.0	Achse V3.0 Auftragstabelle V3.0	MC_Power V3.0 MC_Reset V3.0 MC_Home V3.0 MC_Halt V3.0 MC_MoveAbsolute V3.0 MC_MoveRelative V3.0 MC_MoveVelocity V3.0 MC_MoveJog V3.0 MC_CommandTable V3.0 MC_ChangeDynamic V3.0

Technologie		CPU	Technologieobjekt	Motion Control-Anweisung
V2.0	Neuerungen: • Ruckbegrenzung • Auftragstabelle • MC_ChangeDynamic	V2.1 V2.2 V3.0	Achse V2.0 Auftragstabelle V2.0	MC_Power V2.0 MC_Reset V2.0 MC_Home V2.0 MC_Halt V2.0 MC_MoveAbsolute V2.0 MC_MoveRelative V2.0 MC_MoveVelocity V2.0 MC_MoveJog V2.0 MC_CommandTable V2.0 MC_ChangeDynamic V2.0
V1.0		V1.0 V2.0 V2.1 V2.2 V3.0	Achse V1.0	MC_Power V1.0 MC_Reset V1.0 MC_Home V1.0 MC_Halt V1.0 MC_MoveAbsolute V1.0 MC_MoveRelative V1.0 MC_MoveVelocity V1.0 MC_MoveJog V1.0

Siehe auch

Technologieversion ändern (Seite 1375)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Status der Endschalter (Seite 1380)

Technologieversion ändern

Um die Vorteile einer neuen Technologieversion nutzen zu können, müssen Sie bei bestehenden Projekten ggf. die Version der Technologie einstellen / ändern.

Hinweis

Kompatibilität der Variablen des Technologieobjekts

Beachten Sie beim Wechsel zwischen V1...3 und ≥ V4 die Kompatibilitätsliste (Seite 1376), wenn Sie im Anwenderprogramm, in Beobachtungstabellen etc. Variablen des Technologieobjekts verwenden.

Technologieversion einstellen / ändern

Um die Technologieversion einzustellen/zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Programmiereditor (z. B. durch Öffnen des OB1).
2. Wählen Sie in der Task Card "Anweisungen" im Ordner "Technologie > Motion Control" die gewünschte Technologieversion.
3. Speichern und übersetzen Sie das Projekt. Beachten Sie evtl. Fehleranzeigen beim Übersetzen. Beseitigen Sie die Ursachen der angezeigten Fehler.
4. Überprüfen Sie die Konfiguration der Technologieobjekte.
5. Passen Sie ggf. die Variablennamen in folgenden Objekten entsprechend der Kompatibilitätsliste an.

- Anwenderprogramm
- Beobachtungstabellen
- Forcetabellen
- HMI-Projektierungen
- Tracekonfigurationen

Siehe auch

Versionsübersicht (Seite 1371)

Status der Endschalter (Seite 1380)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5

Im Rahmen der Technologie V4 wurden die Technologie-Datenbausteine für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergeben sich ab V4 neue Variablen und Variablennamen für die Technologieobjekte Positionierachse und Auftragstabelle.

Beachten Sie die nachfolgenden Tabellen, wenn Sie im Anwenderprogramm Variablen der Technologieobjekte verwendet haben und das Projekt von V1...3 nach ≥ V4 und größer (oder umgekehrt) konvertieren möchten.

Die in der Spalte "Automatische Konvertierung V1...3 nach ≥ V4" gekennzeichneten Variablen werden beim Übersetzen des Projekts automatisch konvertiert. Variablennamen in Beobachtungs-, Forcetabellen, HMI und Tracekonfigurationen werden nicht konvertiert.

Die nachfolgenden Variablen sind neu oder wurden angepasst und müssen evtl. im Anwenderprogramm, Beobachtungstabellen etc. korrigiert werden:

Config-Variablen (Positionierachse)

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
<Achsname>.Config.DynamicDefaults.Acceleration	<Achsname>.DynamicDefaults.Acceleration	Ja
<Achsname>.Config.DynamicDefaults.Deceleration	<Achsname>.DynamicDefaults.Deceleration	Ja
<Achsname>.Config.DynamicDefaults.EmergencyDeceleration	<Achsname>.DynamicDefaults.EmergencyDeceleration	Ja
<Achsname>.Config.DynamicDefaults.Jerk	<Achsname>.DynamicDefaults.Jerk	Ja
<Achsname>.Config.DynamicDefaults.JerkActive	Nicht verfügbar Der Ruck ist aktiviert, wenn der konfigurierte Ruck > 0.004 Impulse/s^3 ist.	Nein
<Achsname>.Config.DynamicLimits.MaxVelocity	<Achsname>.DynamicLimits.MaxVelocity	Ja
<Achsname>.Config.DynamicLimits.MinVelocity	<Achsname>.DynamicLimits.MinVelocity	Ja
<Achsname>.Config.General.LengthUnit	<Achsname>.Units.LengthUnit	Ja
<Achsname>.Config.Homing.AutoReversal	<Achsname>.Homing.AutoReversal	Ja
<Achsname>.Config.Homing.Direction	<Achsname>.Homing.ApproachDirection	Ja
<Achsname>.Config.Homing.FastVelocity	<Achsname>.Homing.ApproachVelocity	Ja
<Achsname>.Config.Homing.Offset	<Achsname>.Sensor[1].ActiveHoming.HomePositionOffset	Ja
<Achsname>.Config.Homing.SideActiveHoming	<Achsname>.Sensor[1].ActiveHoming.SideInput	Ja
<Achsname>.Config.Homing.SidePassiveHoming	<Achsname>.Sensor[1].PassiveHoming.SideInput	Ja
<Achsname>.Config.Homing.SlowVelocity	<Achsname>.Homing.ReferencingVelocity	Ja
<Achsname>.Config.Homing.SwitchedLevel	<Achsname>.Sensor[1].ActiveHoming.SwitchLevel <Achsname>.Sensor[1].PassiveHoming.SwitchLevel	Nein
<Achsname>.Config.Mechanics.InverseDirection	<Achsname>.Actor.InverseDirection	Ja
<Achsname>.Config.Mechanics.LeadScrew	<Achsname>.Mechanics.LeadScrew	Ja
<Achsname>.Config.Mechanics.PulsesPerDriveRevolution	<Achsname>.Actor.DriveParameter.PulsesPerDriveRevolution	Ja
<Achsname>.Config.PositionLimits_HW.Active	<Achsname>.PositionLimitsHW.Active	Ja
<Achsname>.Config.PositionLimits_HW.MaxSwitchedLevel	<Achsname>.PositionLimitsHW.MaxSwitchLevel	Ja
<Achsname>.Config.PositionLimits_HW.MinSwitchedLevel	<Achsname>.PositionLimitsHW.MinSwitchLevel	Ja
<Achsname>.Config.PositionLimits_SW.Active	<Achsname>.PositionLimitsSW.Active	Ja
<Achsname>.Config.PositionLimits_SW.MaxPosition	<Achsname>.PositionLimitsSW.MaxPosition	Ja
<Achsname>.Config.PositionLimits_SW.MinPosition	<Achsname>.PositionLimitsSW.MinPosition	Ja

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
Nicht verfügbar	<Achsname>.Actor.DirectionMode	Nein
Nicht verfügbar	<Achsname>.Actor.Type	Nein
Nicht verfügbar	<Achsname>.Sensor[1].ActiveHoming.Mode	Nein
Nicht verfügbar	<Achsname>.Sensor[1].PassiveHoming.Mode	Nein

ErrorBits-Variablen (Positionierachse)

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
<Achsname>.ErrorBits.HwLimitMax	<Achsname>.ErrorBits.HWLimit (Beachten Sie auch die neuen StatusBits und das Kapitel Status der Endschalter (Seite 1380).)	Nein
<Achsname>.ErrorBits.HwLimitMin		
<Achsname>.ErrorBits.SwLimitMaxExceeded	<Achsname>.ErrorBits.SWLimit (Beachten Sie auch die neuen StatusBits und das Kapitel Status der Endschalter (Seite 1380).)	Nein
<Achsname>.ErrorBits.SwLimitMaxReached		
<Achsname>.ErrorBits.SwLimitMinExceeded		
<Achsname>.ErrorBits.SwLimitMinReached		
Nicht verfügbar	<Achsname>.ErrorBits.DirectionFault	Nein

MotionStatus-Variablen (Positionierachse)

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
<Achsname>.MotionStatus.Distance	<Achsname>.StatusPositioning.Distance	Ja
<Achsname>.MotionStatus.Position	<Achsname>.Position	Ja
<Achsname>.MotionStatus.TargetPosition	<Achsname>.StatusPositioning.TargetPosition	Ja
<Achsname>.MotionStatus.Velocity	<Achsname>.Velocity	Ja

StatusBits-Variablen (Positionierachse)

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
<Achsname>.StatusBits.Homing	<Achsname>.StatusBits.HomingCommand	Ja
<Achsname>.StatusBits.SpeedCommand	<Achsname>.StatusBits.VelocityCommand	Ja
Nicht verfügbar	<Achsname>.StatusBits.HWLimitMaxActive	Nein
Nicht verfügbar	<Achsname>.StatusBits.HWLimitMinActive	Nein
Nicht verfügbar	<Achsname>.StatusBits.SWLimitMaxActive	Nein
Nicht verfügbar	<Achsname>.StatusBits.SWLimitMinActive	Nein

Variablen (Auftragstabelle)

Variablenname V1.0 bis V3.0	Variablenname ab V4.0	Automatische Konvertierung V1..3 nach ≥ V4
<Auftragstabelle>.Config.Command[n].Position	<Auftragstabelle>.Command[n].Position	Ja
<Auftragstabelle>.Config.Command[n].Velocity	<Auftragstabelle>.Command[n].Velocity	Ja
<Auftragstabelle>.Config.Command[n].Duration	<Auftragstabelle>.Command[n].Duration	Ja
<Auftragstabelle>.Config.Command[n].NextStep	<Auftragstabelle>.Command[n].NextStep	Ja
<Auftragstabelle>.Config.Command[n].StepCode	<Auftragstabelle>.Command[n].StepCode	Ja

Siehe auch

Versionsübersicht (Seite 1371)

Technologieversion ändern (Seite 1375)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Kompatibilitätsliste der Variablen V4...5 <-> V6

Im Rahmen der Technologie V6 wurden die Technologie-Datenbausteine für S7-1200 Motion Control und S7-1500 Motion Control weiter vereinheitlicht. Dadurch ergeben sich ab V6 neue Variablennamen für das Technologieobjekt Positionierachse.

Beachten Sie die nachfolgenden Tabellen, wenn Sie im Anwenderprogramm Variablen der Technologieobjekte verwendet haben und das Projekt von V4...5 nach ≥ V6 und größer (oder umgekehrt) konvertieren möchten.

Die in der Spalte "Automatische Konvertierung "V4...5 nach ≥ V6 " gekennzeichneten Variablen werden beim Übersetzen des Projekts automatisch konvertiert. Variablennamen in Beobachtungs-, Forcetabellen, HMI und Tracekonfigurationen werden nicht konvertiert.

Die nachfolgenden Variablen sind neu oder wurden angepasst und müssen evtl. im Anwenderprogramm, Beobachtungstabellen etc. korrigiert werden:

Config-Variablen (Positionierachse)

Variablenname V4.0 bis V5.0	Variablenname ab V6.0	Automatische Konvertierung V4...5 nach ≥ V6
<Achsname>.PositionLimitsSW.Active	<Achsname>.PositionLimits_SW.Active	Ja
<Achsname>.PositionLimitsSW.MinPosition	<Achsname>.PositionLimits_SW.MinPosition	Ja
<Achsname>.PositionLimitsSW.MaxPosition	<Achsname>.PositionLimits_SW.MaxPosition	Ja
<Achsname>.PositionLimitsHW.Active	<Achsname>.PositionLimits_HW.Active	Ja

Variablenname V4.0 bis V5.0	Variablenname ab V6.0	Automatische Konvertierung V4...5 nach ≥ V6
<Achsname>.PositionLimitsHW.MinSwitch-Level	<Achsname>.PositionLimits_HW.MinSwitchLevel	Ja
<Achsname>.PositionLimitsHW.MinSwit-chAddress	<Achsname>.PositionLimits_HW.MinSwitchAddress	Ja

Status der Endschalter

Die Status- und Errorbits für die Anzeige angefahrener Endschalter wurden in der Version V4 angepasst.

Um das Verhalten der Errorbits der Versionen V1...3 nachzubilden, verwenden Sie folgende logische Verknüpfungen:

V1...3	ab V4
<Achsname>.ErrorBits.HwLimitMin	<Achsname>.ErrorBits.HWLimit AND <Achsname>.Status-Bits.HWLimitMinActive
<Achsname>.ErrorBits.HwLimitMax	<Achsname>.ErrorBits.HWLimit AND <Achsname>.Status-Bits.HWLimitMaxActive
<Achsname>.ErrorBits.SwLimitMinReached	<Achsname>.ErrorBits.SWLimit AND (<Achsname>.Positi-on = <Achsname>.PositioningLimits_SW.MinPosition)
<Achsname>.ErrorBits.SwLimitMinExceeded	<Achsname>.ErrorBits.SWLimit AND (<Achsname>.Positi-on < <Achsname>.PositioningLimits_SW.MinPosition)
<Achsname>.ErrorBits.SwLimitMaxReached	<Achsname>.ErrorBits.SWLimit AND (<Achsname>.Positi-on = <Achsname>.PositioningLimits_SW.MaxPosition)
<Achsname>.ErrorBits.SwLimitMaxExceeded	<Achsname>.ErrorBits.SWLimit AND (<Achsname>.Positi-on > <Achsname>.PositioningLimits_SW.MaxPosition)

Siehe auch

Versionsübersicht (Seite 1371)

Technologieversion ändern (Seite 1375)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

6.1.1.5 Technologieobjekt Positionierachse

Einbindung des Technologieobjekts Positionierachse

In der nachfolgenden Darstellung werden die Zusammenhänge der Hardware- und Softwarekomponenten dargestellt, die bei der Verwendung des Technologieobjekts Positionierachse zum Einsatz kommen:

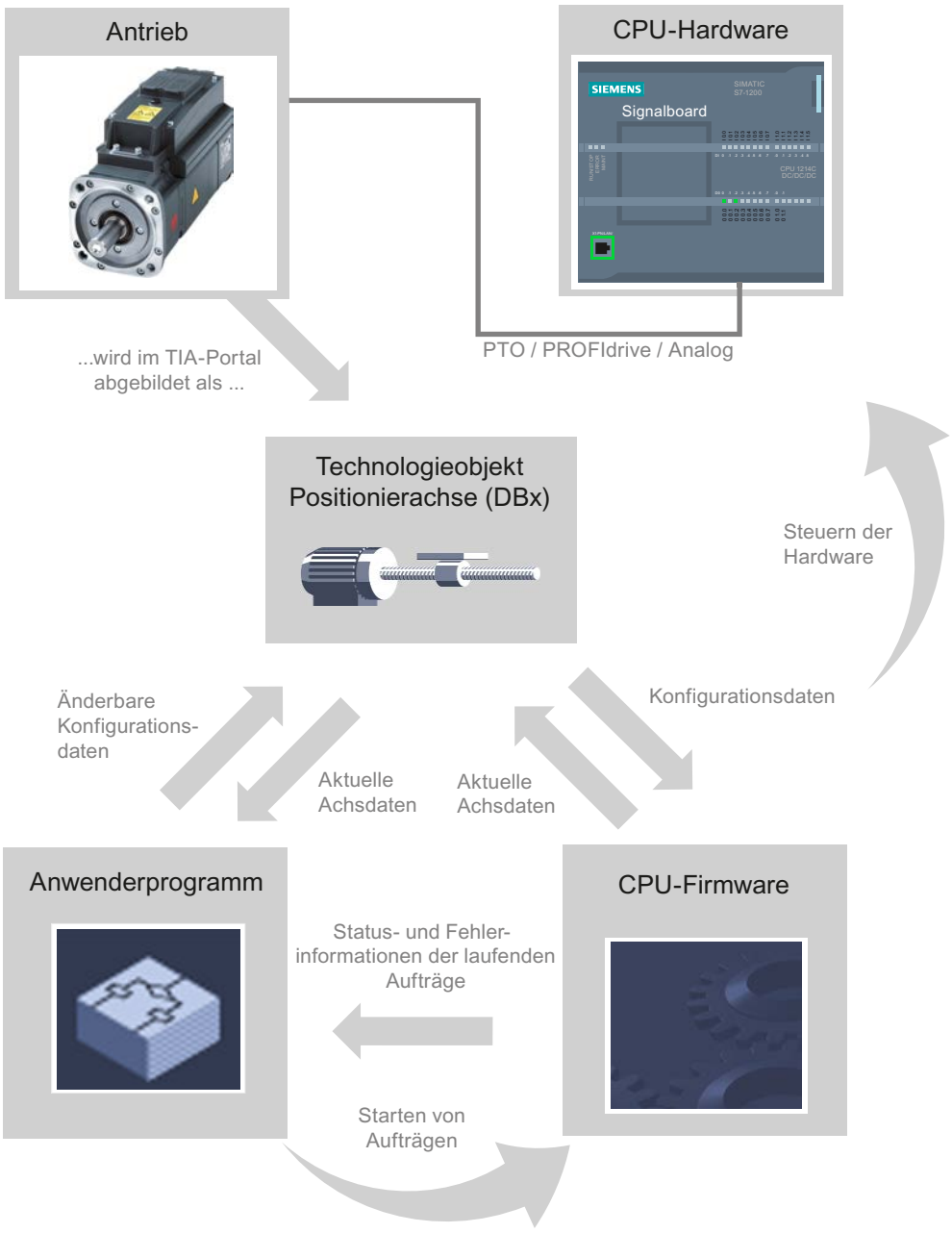

CPU-Hardware

Über die CPU-Hardware wird der physikalische Antrieb gesteuert und überwacht.

Antrieb

Der Antrieb stellt die Einheit aus Leistungsteil und Motor dar. Sie können Schrittmotoren und Servomotoren mit Impuls-, PROFIdrive- oder Analogschnittstelle verwenden.

Technologieobjekt Positionierachse

Der physikalische Antrieb inklusive Mechanik wird im TIA-Portal als Technologieobjekt Positionierachse abgebildet. Konfigurieren Sie hierzu das Technologieobjekt Positionierachse mit den nachfolgenden Parametern:

- Auswahl des zu verwendenden PTOs (Pulse Train Output) / PROFIdrive-Antriebs / Analogausgangs und Konfiguration der Antriebsschnittstelle
- Parameter zur Mechanik und zur Getriebeübersetzung des Antriebs (bzw. der Maschine oder Anlage)
- Parameter zu Positionsgrenzen und zur Positionierüberwachung
- Parameter zur Dynamik und zum Referenzieren
- Parameter zum Regelkreis

Die Konfiguration des Technologieobjekts Positionierachse wird im Technologieobjekt (Datenbaustein) gespeichert. Dieser Datenbaustein bildet gleichzeitig die Schnittstelle zwischen Anwenderprogramm und CPU-Firmware. Zur Laufzeit des Anwenderprogramms werden im Datenbaustein des Technologieobjekts die aktuellen Achsdaten abgelegt.

Anwenderprogramm

Über das Anwenderprogramm starten Sie über Motion Control-Anweisungen Aufträge in der CPU-Firmware. Folgende Aufträge zur Steuerung der Achse sind möglich:

- Achse freigeben und sperren
- Achse absolut positionieren
- Achse relativ positionieren
- Achse mit Drehzahlvorgabe fahren
- Achsaufträge als Bewegungssequenz ausführen (Technologie ab V2, nur PTO)
- Achse im Tippbetrieb fahren
- Achse anhalten
- Achse referenzieren; Referenzpunkt setzen
- Dynamikeinstellungen der Achse ändern
- Bewegungsdaten der Achse kontinuierlich lesen
- Variable der Achse lesen und schreiben
- Fehler quittieren

Über die Eingangsparameter der Motion Control-Anweisungen und die Konfiguration der Achse bestimmen Sie die Parameter des Auftrags. Über die Ausgangs-Parameter der Anweisung erhalten Sie zeitaktuelle Informationen zum Status und zu evtl. Fehlern des Auftrags.

Bevor Sie einen Auftrag für die Achse starten, müssen Sie die Achse mit der Motion Control-Anweisung "MC_Power" freigeben.

Über die Variablen des Technologieobjekts können Sie im Anwenderprogramm Konfigurationsdaten und aktuelle Achsdaten auslesen. Einzelne änderbare Variablen des Technologieobjekts (z. B. die aktuelle Beschleunigung) können Sie vom Anwenderprogramm aus ändern.

Zusätzlich können Sie mit der Motion Control-Anweisung "MC_ChangeDynamic" die Dynamikeinstellungen der Achse ändern und mit "MC_WriteParam" weitere Konfigurationsdaten schreiben. Mit der Motion Control-Anweisung "MC_ReadParam" können Sie den aktuellen Bewegungsstatus der Achse auslesen.

CPU Firmware

Die im Anwenderprogramm angestoßenen Motion Control-Aufträge werden in der CPU-Firmware abgearbeitet. Beim Einsatz der Achssteuertafel werden die Motion Control-Aufträge über die Bedienung der Achssteuertafel angestoßen. Entsprechend der Konfiguration der Achse erfüllt die CPU-Firmware folgende Aufgaben:

- Berechnen des genauen Bewegungsprofils für Bewegungsaufträge und Notstoppsituationen
- Lageregelung für Antriebsanbindung über PROFIdrive / analoge Antriebsanbindung
- Steuerung des Impuls- und Richtungssignals für Antriebsanbindung über PTO
- Steuerung der Antriebsfreigabe
- Überwachung des Antriebs sowie der Hardware- und Software-Endschalter
- Zeitaktuelles Rückmelden von Status- und Fehlerinformationen der Aufträge an die Motion Control-Anweisungen im Anwenderprogramm
- Schreiben von aktuellen Achsdaten in den Datenbaustein des Technologieobjekts

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Werkzeuge des Technologieobjekts Positionierachse (Seite 1384)

Hardware- und Software-Endschalter (Seite 1367)

Referenzieren (Seite 1369)

Werkzeuge des Technologieobjekts Positionierachse

Für das Technologieobjekt Positionierachse stehen Ihnen im TIA-Portal die Werkzeuge "Konfiguration", "Inbetriebnahme" und "Diagnose" zur Verfügung. Die nachfolgende Darstellung zeigt das Zusammenwirken der drei Werkzeuge mit dem Technologieobjekt und dem Antrieb:

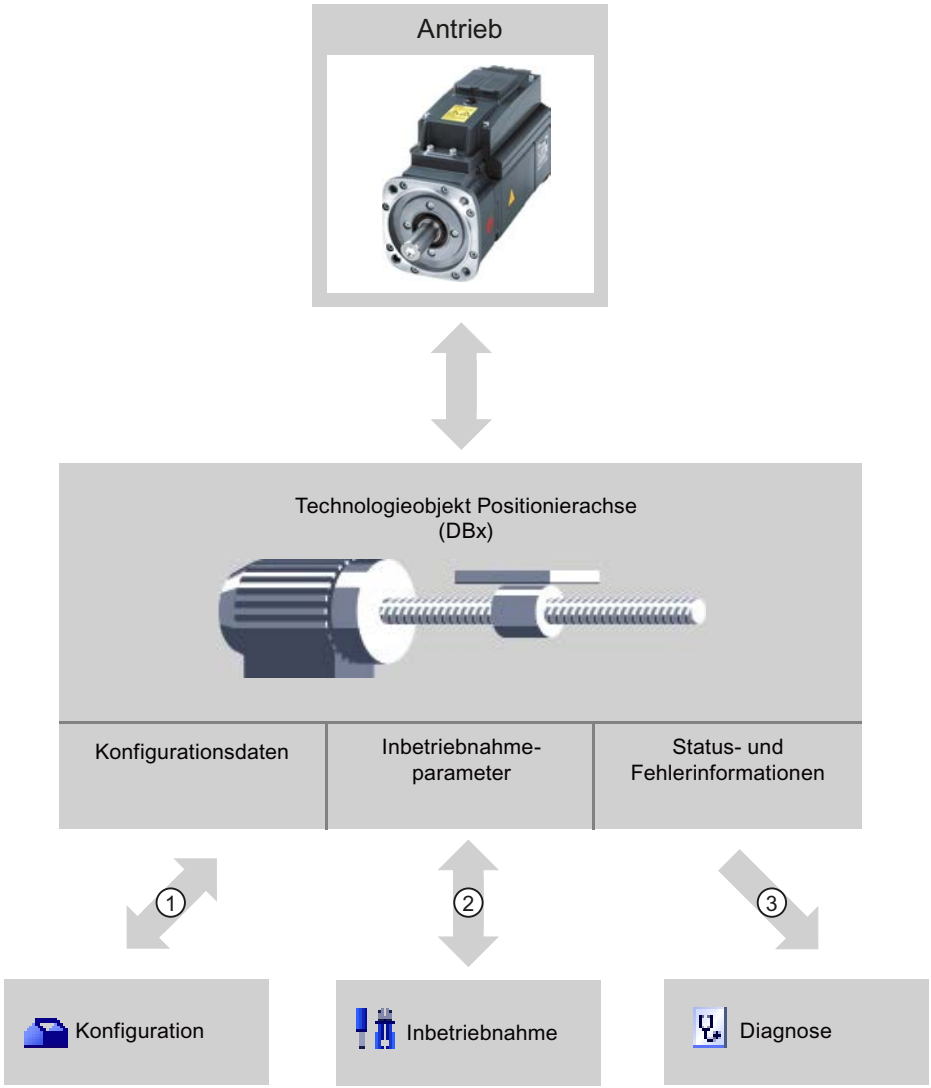

①	Schreiben und Lesen der Konfigurationsdaten des Technologieobjekts
②	Steuern des Antriebs über das Technologieobjekt. Lesen des Achsstatus zur Anzeige in der Achssteuertafel. Optimierung der Lageregelung.
③	Auslesen der aktuellen Status- und Fehlerinformationen des Technologieobjekts Bei PROFIdrive-Antrieben werden zusätzlich Telegramminformationen angezeigt

Konfiguration

Konfigurieren Sie mit dem Werkzeug "Konfiguration" folgende Eigenschaften des Technologieobjekts Positionierachse:

- Auswahl des zu verwendenden PTOs (Pulse Train Output) / PROFIdrive-Antriebs / Analogausgangs und Konfiguration der Antriebsschnittstelle
- Eigenschaften der Mechanik und der Getriebeübersetzung des Antriebs (bzw. der Maschine oder Anlage)
- Eigenschaften der Positionsgrenzen und der Positionierüberwachung
- Eigenschaften der Dynamik und des Referenzierens
- Parameter des Regelkreises

Die Konfiguration wird im Datenbaustein des Technologieobjekts gespeichert.

Inbetriebnahme

Testen Sie mit dem Werkzeug "Inbetriebnahme" die Funktion Ihrer Achse, ohne dass ein Anwenderprogramm angelegt sein muss. Mit dem Start des Werkzeugs öffnet sich die Achssteuertafel. In der Achssteuertafel stehen Ihnen folgende Befehle zur Verfügung:

- Freigeben und Sperren der Achse
- Verfahren der Achse im Tipp-Betrieb
- Absolutes und relatives Positionieren der Achse
- Referenzieren der Achse
- Quittieren von Fehlern

Für die Bewegungsbefehle können die Dynamikwerte entsprechend angepasst werden. Die Achssteuertafel zeigt zusätzlich den aktuellen Achsstatus an.

Bei Antriebsanbindung über PROFIdrive / Analogausgang unterstützt Sie die Optimierung bei der Ermittlung der optimalen Verstärkung für den Regelkreis.

Diagnose

Kontrollieren Sie mit dem Werkzeug "Diagnose" die aktuellen Status- und Fehlerinformationen von Achse und Antrieb.

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Einbindung des Technologieobjekts Positionierachse (Seite 1381)

Hardware- und Software-Endschalter (Seite 1367)

Referenzieren (Seite 1369)

Technologieobjekt Positionierachse konfigurieren (Seite 1387)

Achssteuertafel (Seite 1470)

Achse - Diagnose (Seite 1499)

Technologieobjekt Positionierachse hinzufügen

Voraussetzung

Ein Projekt mit einer CPU S7-1200 ist angelegt.

Vorgehen

Um ein Technologieobjekt Positionierachse in der Projektnavigation hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "CPU > Technologieobjekte".
2. Doppelklicken Sie auf den Befehl "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
3. Wählen Sie die Technologie "Motion Control".
4. Öffnen Sie den Ordner "Motion Control".
5. Wählen Sie in der Spalte "Version" die gewünschte Technologieversion.
6. Wählen Sie das Objekt "TO_PositioningAxis".
7. Geben Sie im Eingabefeld "Name" den Namen der Achse ein.
8. Um die automatisch vergebene Nummer des Datenbausteins zu ändern, wählen Sie die Option "manuell".
9. Um weitere Informationen zum Technologieobjekt anzuzeigen und zu ergänzen, klicken Sie auf "Weitere Informationen".
10. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

Im Ordner "Programmbausteine" werden automatisch die Organisationsbausteine MC-Servo [OB91] und MC-Interpolator [OB92] angelegt. In diesen Organisationsbausteinen werden die Technologieobjekte bearbeitet. Im MC-Servo [OB91] wird der Lageregler berechnet. Der MC-Interpolator [OB92] übernimmt die Auswertung der Motion Control-Anweisungen, die Sollwertgenerierung und die Überwachungsfunktionalität.

Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 1371)

Technologieobjekt Positionierachse konfigurieren

Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Gruppe des gewünschten Technologieobjekts.
2. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**
  Die Grundparameter enthalten alle Parameter, die für eine funktionsfähige Achse konfiguriert werden müssen.
- **Erweiterte Parameter**
  Die erweiterten Parameter enthalten Parameter zur Anpassung an Ihren Antrieb, bzw. an Ihre Anlage.

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.
	Die Konfiguration ist gültig, enthält jedoch Warnhinweise Z. B. nur ein Hardware-Endschalter ist konfiguriert. Je nach Anlage kann die fehlende Konfiguration eines Hardware-Endschalters zu einer Gefährdung führen. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt.

Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 1371)

Grundparameter (Seite 1388)

Erweiterte Parameter (Seite 1400)

Werte beobachten

Wenn eine Online-Verbindung zur CPU besteht, wird in den Konfigurationsdialogen des Technologieobjekts das Symbol "Alle beobachten" angezeigt.

Die Funktion "Alle beobachten" bietet folgende Möglichkeiten:

- Vergleich der konfigurierten Startwerte des Projekts mit den Startwerten in der CPU und den Aktualwerten
- Direktes Bearbeiten der Aktualwerte und der Startwerte des Projekts
- Sofortiges Erkennen und Anzeigen von Eingabefehlern mit Korrekturvorschlägen
- Sicherung der Aktualwerte im Projekt durch manuelle übernahme in den Startwert des Projekts

Symbole und Bedienelemente

Wenn eine Online-Verbindung zur CPU besteht, werden an den Parametern die Aktualwerte angezeigt.

Neben den Aktualwerten der Parameter werden folgende Symbole angezeigt:

Symbol	Beschreibung
	Startwert in CPU ist gleich dem konfigurierten Startwert im Projekt
	Startwert in CPU ist ungleich dem konfigurierten Startwert im Projekt
	Ein Vergleich des Startwert in CPU und des konfiguriertem Startwert im Projekt kann nicht durchgeführt werden, weil die gewählte CPU-Baugruppe diesen Vergleich nicht unterstützt.
	Der Wert ist nicht sinnvoll vergleichbar, da er in einer der Konfigurationen nicht relevant ist.
	Mit der Schaltfläche zeigen Sie für den jeweiligen Parameter den Startwert der CPU und den Startwert des Projekts an.

Den Aktualwert und den Startwert im Projekt können Sie direkt ändern und anschließend in die CPU laden. Bei direkt änderbaren Parametern wird die Änderung des Aktualwerts direkt in die CPU übernommen.

Grundparameter

Konfiguration - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" die grundlegenden Eigenschaften des Technologieobjekts Positionierachse.

Achsname

Definieren Sie in diesem Feld den Namen der Achse bzw. den Namen des Technologieobjekts Positionierachse. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

Antrieb

Wählen Sie die Art der Antriebsanbindung aus:

- **PTO (Pulse Train Output)**
  Der Antrieb wird über einen Impulsgeneratorausgang, einen optionalen Freigabe-Ausgang und einen optionalen Bereit-Eingang angebunden.
- **Analoge Antriebsanbindung**
  Der Antrieb wird über einen Analogausgang, einen Geber, einen optionalen Freigabe-Ausgang und einen optionalen Bereit-Eingang angebunden.
  Alle Bewegungen der Achse werden lagegeregelt.
- **PROFIdrive**
  Der Antrieb wird über PROFINET/PROFIBUS angebunden. Die Kommunikation zwischen Steuerung und Antrieb erfolgt über PROFIdrive-Telegramme.
  Alle Bewegungen der Achse werden lagegeregelt.

Wenn Sie "Analoge Antriebsanbindung" oder "PROFIdrive" auswählen, wird die Navigation der Konfiguration um zusätzliche Elemente erweitert:

- Geber
- Modulo
- Positionsüberwachungen (Positionierüberwachung, Schleppfehler und Stillstandssignal)
- Regelkreis

In den zusätzlichen Konfigurationsfenstern konfigurieren Sie den anzuschließenden Geber sowie die resultierenden Möglichkeiten zur Lageregelung und Positionsüberwachung.

Maßeinheit Position

Wählen Sie in der Klappliste die gewünschte Maßeinheit für das Maßsystem der Achse aus. Die gewählte Maßeinheit wird für die weitere Konfiguration des Technologieobjekts Positionierachse und für die Anzeige der aktuellen Achsdaten verwendet.

Die Werte an den Eingangsparametern (Position, Distance, Velocity, ...) der Motion Control-Anweisungen beziehen sich ebenfalls auf diese Maßeinheit.

Hinweis

Wählen Sie die Antriebsanbindung und die Maßeinheit der Position zu Beginn der Achskonfiguration.

Bei einer nachträglichen Änderung werden Parameter zurückgesetzt, bzw. neu initialisiert, was eine Kontrolle der Parameter der Konfigurationsdialoge erfordert.

Gegebenenfalls müssen Sie im Anwenderprogramm Werte an den Eingangsparametern der Motion Control-Anweisungen an die neue Maßeinheit anpassen.

Simulation

Wählen Sie in der Klappliste, ob der Antrieb und der Geber simuliert werden sollen oder nicht. Die Simulation kann für die analoge Antriebsanbindung oder für einen PROFIdrive-Antrieb gewählt werden. Die Konfiguration der Antriebs- und Geber-Hardware ist für den Simulationsbetrieb nicht erforderlich (evtl. Fehler in der Antriebs- oder Geberkonfiguration werden ignoriert).

Anwendung: Der Antrieb wird z. B. zur Inbetriebnahme simuliert und später mit der gegebenenfalls konfigurierten Hardware betrieben.

Die Betriebsart "Simulation" lässt sich während der Laufzeit des Anwenderprogramms per Download und anschließendem "MC_Reset" mit Parameter "Restart" = TRUE ändern.

Im Simulationsbetrieb werden keine Sollwerte an den Antrieb ausgegeben bzw. Istwerte vom Antrieb/Geber eingelesen. Hardware-Endschalter und Referenzpunktschalter haben keine Wirkung.

Die folgende Tabelle zeigt Motion Control-Anweisungen mit angepasstem Verhalten im Simulationsbetrieb:

Motion Control-Anweisung	Verhalten im Simulationsbetrieb
MC_Power	Die Achse wird unmittelbar freigegeben, ohne auf Rückmeldung vom Antrieb zu warten.
MC_Home	Referenzieraufträge werden unmittelbar ohne simulierte Achsbewegung ausgeführt.

PTO-Antriebe arbeiten ohne Regelkreis. Um einen PTO-Antrieb zu simulieren, wird keine eigene Simulationsfunktion benötigt, falls der PTO-Antrieb nicht angeschlossen ist.

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Konfiguration - Allgemein (Technologieobjekt "Achse" V1...3) (Seite 1585)

Konfiguration - Antrieb

Konfiguration - Antrieb - PTO (Pulse Train Output)

Konfigurieren Sie im Konfigurationsfenster "Antrieb" den Impulsgenerator sowie die Freigabe und Rückmeldung des Antriebs.

Hardware-Schnittstelle

Die Impulse werden über fest zugeordnete digitale Ausgänge an das Leistungsteil des Antriebs ausgegeben.

Bei CPUs mit Relaisausgängen kann das Impulssignal nicht auf diesen Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen.

Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, verwenden Sie ein Signalboard mit digitalen Ausgängen.

Hinweis

Der PTO benötigt die Funktionalität eines schnellen Zählers (HSC). Dazu wird ein interner HSC verwendet, dessen Zählerstand nicht ausgewertet werden kann.

Impulsgenerator

Wählen Sie in der Klappliste den PTO (Pulse Train Output) für die Ansteuerung des Schrittmotors oder Servomotors mit Impulsschnittstelle. Wenn Sie in der Gerätekonfiguration die Impulsgeneratoren und die schnellen Zähler nicht anderweitig verwendet haben, kann die Hardware-Schnittstelle automatisch konfiguriert werden. In diesem Fall wird der gewählte PTO in der Klappliste weiß hinterlegt angezeigt.

Bei ausgewähltem PTO (Pulse Train Output) gelangen Sie über die Schaltfläche "Gerätekonfiguration" zur Parametrierung der Impulsoptionen in der Gerätekonfiguration der CPU. Dies kann im Konfliktfall nötig sein, wenn der PTO anderseitig verwendet wurde, oder die Parameter anwenderseitig verändert wurden.

Signalart

Wählen Sie in der Klappliste die Signalart. Folgende Signalarten stehen zur Auswahl:

- **PTO (Impuls A und Richtung B)**
  Zur Ansteuerung des Schrittmotors wird ein Impulsausgang und ein Richtungsausgang verwendet.
- **PTO (Vorwärts takten A und rückwärts takten B)**
  Zur Ansteuerung des Schrittmotors wird je ein Impulsausgang zur Bewegung in positive Richtung und negative Richtung verwendet.
- **PTO (A/B phasenversetzt)**
  Beide Impuslausgänge für Phase A und für Phase B takten mit der gleichen Frequenz. Antriebsseitig wird die Periode der Impulsausgänge als Schritt ausgewertet.
  Die Phasenverschiebung zwischen Phase A und Phase B bestimmt die Bewegungsrichtung.
- **PTO (A/B phasenversetzt - vierfach)**
  Beide Impuslausgänge für Phase A und für Phase B takten mit der gleichen Frequenz. Antriebsseitig werden alle steigenden und alle fallenden Flanken von Phase A und Phase B als Schritt ausgewertet.
  Die Phasenverschiebung zwischen Phase A und Phase B bestimmt die Bewegungsrichtung.

Die folgende Tabelle zeigt die zu konfigurierenden Parameter in Abhängigkeit von der Signalart:

Signalart / Parameter		Beschreibung
PTO (Impuls A und Richtung B)		
	Impulsausgang	Wählen Sie in diesem Feld den Impulsausgang für Bewegungen in positive Richtung. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.
	Richtungsausgang aktivieren	Mit dieser Option aktivieren oder deaktivieren Sie den Richtungsausgang. Wenn Sie den Richtungsausgang deaktivieren, wird die Bewegungsrichtung eingeschränkt.
	Richtungsausgang	Wählen Sie in diesem Feld den Ausgang für den Richtungsausgang. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.
PTO (Vorwärts takten A und rückwärts takten B)		
	Impulsausgang vorwärts	Wählen Sie in diesem Feld den Impulsausgang für Bewegungen in positive Richtung. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.
	Impulsausgang rückwärts	Wählen Sie in diesem Feld den Impulsausgang für Bewegungen in negative Richtung. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.
PTO (A/B phasenversetzt) / PTO (A/B phasenversetzt - vierfach)		
	Signal A	Wählen Sie in diesem Feld den Impulsausgang für Phase A Signale. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.
	Signal B	Wählen Sie in diesem Feld den Impulsausgang für Phase B Signale. Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Freigabe und Rückmeldung des Antriebs

In diesem Bereich konfigurieren Sie den Ausgang für die Antriebsfreigabe und den Eingang für die Rückmeldung "Antrieb Bereit" des Antriebs:

- **Freigabe-Ausgang**
  Wählen Sie in diesem Feld den Freigabe-Ausgang zur Antriebsfreigabe.
- **Bereit-Eingang**
  Wählen Sie in diesem Feld den Bereit-Eingang für die Rückmeldung "Antrieb Bereit" des Antriebs

Die Antriebsfreigabe wird von der Motion Control-Anweisung "MC_Power" gesteuert und erteilt dem Antrieb die Leistungsfreigabe. Ist der Antrieb nach Erhalt der Antriebsfreigabe bereit, Bewegungen auszuführen, so meldet er "Antrieb Bereit" an die CPU.

Wenn der Antrieb nicht über derartige Schnittstellen verfügt, müssen Sie die Parameter nicht konfigurieren. Wählen Sie in diesem Fall für den Bereit-Eingang den Wert TRUE.

Konfiguration - Antrieb - Analoge Antriebsanbindung

Konfigurieren Sie im Konfigurationsfenster "Antrieb" den analogen Ausgang sowie die Freigabe und Rückmeldung des Antriebs.

Hardware-Schnittstelle

Die Solldrehzahl wird über einen fest zugeordneten analogen Ausgang an das Leistungsteil des Antriebs ausgegeben.

Konfigurieren Sie in diesem Bereich die Ein- und Ausgänge für die Ansteuerung des Antriebs:

- Analog-Ausgang
  Wählen Sie in diesem Feld die PLC-Variable des analogen Ausgangs, über den der Antrieb angesteuert wird.
  Wenn Sie die Autovervollständigung öffnen, werden alle Ausgangsadressen mit 16 Bit angezeigt (WORD, INT, UINT). Zur Datenanbindung über Datenbaustein kann auch die Variable eines Datenbausteins vom Datentyp WORD gewählt werden.
  Sie können auch eine Adresse (z. B. QW20) eingeben. Wenn die Adresse gültig ist, wird der Name "Axis_1_AnalogOutput" für diese Adresse generiert und in die Variablentabelle eingefügt. Damit die Adresse gültig ist, muss sie vom passenden Datentyp und durch eine HW-Baugruppe belegt sein.
- Auswahl Freigabe-Ausgang
  Wählen Sie in diesem Feld einen vorhandenen Ausgang als Freigabe-Ausgang zur Antriebsfreigabe.
- Auswahl Bereit-Eingang
  Wählen Sie in diesem Feld den Bereit-Eingang für die Rückmeldung "Antrieb Bereit" des Antriebs

Die Antriebsfreigabe wird von der Motion Control-Anweisung "MC_Power" gesteuert und erteilt dem Antrieb die Leistungsfreigabe. Ist der Antrieb nach Erhalt der Antriebsfreigabe bereit, Bewegungen auszuführen, so meldet er "Antrieb Bereit" an die CPU. Wenn der Antrieb nicht über derartige Schnittstellen verfügt, müssen Sie die Parameter nicht konfigurieren. Wählen Sie in diesem Fall für den Bereit-Eingang den Wert TRUE.

Datenaustausch Antrieb

Konfigurieren Sie in diesem Bereich die Skalierung der Solldrehzahl:

- **Bezugsdrehzahl**
  Die Bezugsdrehzahl des Antriebs ist die Drehzahl, mit welcher der Antrieb bei der Ausgabe von 100 % am Analogausgang dreht. Die Bezugsdrehzahl muss am Antrieb konfiguriert werden und in der Konfiguration des Technologieobjekts übernommen werden.
  Der bei 100 % ausgegebene Analogwert hängt vom Typ des Analogausgangs ab. Beispielsweise wird bei einem analogen Ausgang mit ±10 V bei 100 % der Wert 10 V ausgegeben.
- **Maximale Drehzahl**
  Geben Sie in diesem Feld die maximale Drehzahl des Antriebs an.
  Die maximale Drehzahl ist durch den Antrieb und durch den Wertebereich des Analogausgangs begrenzt. Im einfachsten Fall sind Bezugsdrehzahl und maximale Drehzahl identisch.
  Analogausgänge lassen sich um etwa 17 % übersteuern. Wenn der Antrieb die Übersteuerung zulässt, können Sie somit einen Analogausgang als Grenzwert im Bereich -117 % bis 117 % betreiben.
- **Antriebsrichtung invertieren**
  Aktivieren Sie das Optionskästchen, um den Drehsinn des Antriebs zu invertieren.

Siehe auch

Datenanbindung Antrieb mit analoger Antriebsanbindung (Seite 1361)

Konfiguration - Antrieb - PROFIdrive

Konfigurieren Sie im Konfigurationsfenster "Antrieb" die Datenanbindung und die Parameter des PROFIdrive-Antriebs.

PROFIdrive-Antrieb (Ab V6)

- **Datenanbindung**
  Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Antriebsgerät erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.
- **Antrieb** (bei Datenanbindung: "Antrieb")
  Wählen Sie im Feld "Antrieb" einen bereits konfigurierten PROFIdrive-Antrieb.
- **Datenbaustein** (bei Datenanbindung: "Datenbaustein")
  Wählen Sie einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Datenaustausch Antrieb

Konfigurieren Sie in diesem Bereich den Datenaustausch zwischen Antrieb und Steuerung:

- **Antriebstelegramm** (bei Datenanbindung: "Datenbaustein" nicht wählbar)
  Überprüfen Sie, bzw. wählen Sie in der Klappliste das Telegramm des Antrieb. Die Angabe muss mit der Gerätekonfiguration des Antriebs übereinstimmen.
- **Ein- / Ausgangsadresse**
  Die Felder zeigen die symbolische und absolute Ein- bzw. Ausgangsadresse des Telegramms.
- **Antriebsrichtung invertieren**
  Aktivieren Sie das Optionskästchen, um den Drehsinn des Antriebs zu invertieren.
- **Automatische Übernahme der Antriebsparameter im Gerät**
  Aktivieren Sie das Optionskästchen, wenn Sie die Antriebsparameter "Bezugsdrehzahl" und "Maximale Drehzahl" als Wert aus der Konfiguration des Antriebs in die CPU übernehmen möchten. Die Antriebsparameter werden nach der (Neu-)Initialisierung des Technologieobjekts und (Wieder-)Anlauf des Antriebs und der CPU vom Bus übernommen. Alternativ müssen Sie die folgenden Parameter händisch abgleichen:
  - **Bezugsdrehzahl**
    Konfigurieren Sie die Bezugsdrehzahl, wie in der Konfiguration des Antriebs eingestellt. Auf dem Bus wird dann z. B. der Wert 16#4000 übertragen, was 100% der Bezugsdrehzahl entspricht.
  - **Maximale Drehzahl**
    Konfigurieren Sie in diesem Feld die maximale Drehzahl des Antriebs.
    Die maximale Drehzahl ergibt sich aus der Konfiguration des Antriebes. Über den Bus können maximal -200% bis +200% der Bezugsdrehzahl übertragen werden. Die maximale Drehzahl kann hiermit maximal doppelt so groß wie die Bezugsdrehzahl sein.

Hinweis

Die automatische Übernahme der Antriebsparameter ist nur mit SINAMICS-Antrieben ab V4.x möglich. Dazu muss im Konfigurationsfenster als Datenanbindung "Antrieb" gewählt sein.

Siehe auch

Datenanbindung PROFIdrive-Antrieb/PROFIdrive-Geber (Seite 1356)

Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1353)

Konfiguration - Geber

Geberkopplung

Abhängig von der Wahl der Geberkopplung konfigurieren Sie im Konfigurationsfenster "Geber" verschiedene Parameter. Folgende Geberkopplungen sind möglich:

- Geber am schnellen Zähler (HSC) (Seite 1398)
- PROFIdrive-Geber am PROFINET/PROFIBUS (Geber am Antrieb, Geber am Technologiemodul, PROFIdrive-Geber) (Seite 1396)

Konfiguration - Geber - Geber am PROFINET/PROFIBUS

Auswahl Geber

Wählen Sie im Feld "PROFIdrive-Geber" den PROFIdrive-Geber am PROFINET.

- **Datenanbindung**
  Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Geber erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.
- **PROFIdrive-Geber/Datenbaustein**
  Wählen Sie in diesem Konfigurationsfeld einen bereits konfigurierten PROFIdrive-Geber. Folgende Geber können ausgewählt werden:
  - **Geber am Antrieb (nicht bei analoger Antriebsanbindung)**
    Der Geber ist am Antrieb angeschlossen. Die Gebersignale werden vom Antrieb ausgewertet und als Teil des Antriebstelegramms (Tel 3 oder 4) zur Steuerung übermittelt (die Geber der Telegramme anderer Antriebe können nicht verwendet werden).
    Die Konfiguration des Gebers erfolgt über die Konfiguration des PROFIdrive-Antriebs.
  - **Geber am Technologiemodul (TM)**
    Wählen Sie ein bereits konfiguriertes Technologiemodul und den zu verwendenden Kanal aus. Zur Auswahl werden nur Technologiemodule angezeigt, die auf den Betriebsmodus "Positionserfassung für Motion Control" eingestellt sind.
    Wenn kein Technologiemodul zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie ein Technologiemodul hinzu.
    Welche Technologiemodule zur Positionserfassung für Motion Control geeignet sind, entnehmen Sie der Dokumentation zum Technologiemodul und den Katalogdaten.
  - **PROFIdrive-Geber am PROFINET/PROFIBUS (PROFIdrive)**
    Wählen Sie im Feld "PROFIdrive-Geber" einen bereits konfigurierten Geber am PROFINET/PROFIBUS aus. Wechseln Sie in der Gerätekonfiguration in die Netzsicht und fügen Sie einen Geber hinzu, falls kein Geber zur Auswahl steht.

  Wurde unter Datenanbindung "Datenbaustein" gewählt, so ist hier ein zuvor erstellter Datenbaustein auszuwählen, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer über die der Geber angebunden ist). Es kann der Geber des gewählten Antriebstelegramms (Tel 3 oder 4) oder ein separater Geber (Tel 81 oder 83) verwendet werden.

Datenaustausch Geber

Konfigurieren Sie in diesem Bereich den Datenaustausch zwischen Geber und Steuerung:

- **Gebertelegramm** (bei Datenanbindung: "Datenbaustein" nicht wählbar)
  Wählen Sie in der Klappliste das Telegramm des Gebers. Die Angabe muss mit der Gerätekonfiguration übereinstimmen.
- **Ein- / Ausgangsadresse**
  Die Felder zeigen die symbolische und absolute Ein- bzw. Ausgangsadresse des Telegramms.
- **Geberrichtung invertieren**
  Aktivieren Sie das Optionskästchen, um den Istwert des Gebers zu invertieren.
- **Automatische Übernahme der Geberparameter im Gerät**
  Aktivieren Sie das Optionskästchen, wenn Sie die Geberparameter aus der Konfiguration des Gebers in die CPU übernehmen möchten. Die Geberparameter werden nach der (Neu-)Initialisierung des Technologieobjekts und (Wieder-)Anlauf des Gebers und der CPU vom Bus übernommen. Der Gebertyp muss in der Konfiguration der Achse und in der Konfiguration des Gebers übereinstimmen.

  Hinweis

  Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster als Datenanbindung "Geber" gewählt sein.

  Bei einem Geber am SINAMICS-Antrieb ist ein Ausgabestand > V4.x nötig

  Ohne automatische Übernahme der Geberparameter müssen die Parameter händisch abgeglichen werden. Die abzugleichenden Parameter finden Sie im Kapitel Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1353).

Gebertyp

Stellen Sie im Feld "Gebertyp" den verwendeten Gebertyp ein. Folgende Gebertypen sind wählbar:

- **Linear inkrementell**
- **Linear absolut**
- **Rotatorisch inkrementell**
- **Rotatorisch absolut**

Abhängig vom gewählten Gebertyp konfigurieren Sie verschiedene Parameter. Konfigurieren Sie je nach gewähltem Gebertyp folgende Parameter:

Gebertyp / Parameter		Beschreibung
Linear inkrementell		
	Abstand zwischen Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Schritten des Gebers.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).
Linear absolut		

Gebertyp / Parameter		Beschreibung
	Abstand zwischen Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Schritten des Gebers.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).
	Feinauflösung - Bits im abs. Istwert (Gn_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der reservierten Bits für den Multiplikationsfaktor des Absolutwerts der Feinauflösung (Gn_XIST2).
Rotatorisch inkrementell		
	Schritte pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Schritte, die der Geber pro Umdrehung auflöst.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).
Rotatorisch absolut		
	Schritte pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Schritte, die der Geber pro Umdrehung auflöst.
	Anzahl Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).
	Feinauflösung - Bits im abs. Istwert (Gn_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der reservierten Bits für den Multiplikationsfaktor des Absolutwerts der Feinauflösung (Gn_XIST2).

Siehe auch

Datenanbindung PROFIdrive-Antrieb/PROFIdrive-Geber (Seite 1356)

Technologiemodule für Motion Control konfigurieren (Seite 1446)

Konfiguration - Geber - Geber am schnellen Zähler (HSC)

Auswahl schneller Zähler (HSC)

Wählen Sie im Feld den schnellen Zähler, an den der Geber den Istwert überträgt.

Prüfen Sie die Filterzeiten der beiden verwendeten Digitaleingänge des schnellen Zählers. Die Filterzeiten sollten ausreichend klein sein, um die Impulse sicher erfassen zu können.

HSC-Schnittstelle

Wählen Sie im Feld "Betriebsart" die Betriebsart des schnellen Zählers.

Abhängig von der Betriebsart konfigurieren Sie verschiedene Eingänge:

Betriebsart / Parameter		Beschreibung
Zweiphasig		
	Taktgeber vorwärts	Wählen Sie in diesem Feld den Eingang zum Vorwärtszählen. Sie können den Eingang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen. Neben dem Adressfeld werden die Frequenz und der Ort (onboard, Signalboard) des Eingangs angezeigt.
	Taktgeber rückwärts	Wählen Sie in diesem Feld den Eingang zum Rückwärtszählen. Sie können den Eingang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen. Neben dem Adressfeld werden die Frequenz und der Ort (onboard, Signalboard) des Eingangs angezeigt.
A/B-Zähler / A/B-Zähler vierfach		
	Taktgeber A	Wählen Sie in diesem Feld den Eingang für Phase A Signale. Sie können den Eingang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen. Neben dem Adressfeld werden die Frequenz und der Ort (onboard, Signalboard) des Eingangs angezeigt.
	Taktgeber B	Wählen Sie in diesem Feld den Eingang für Phase B Signale. Sie können den Eingang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen. Neben dem Adressfeld werden die Frequenz und der Ort (onboard, Signalboard) des Eingangs angezeigt.

Geberrichtung invertieren

Aktivieren Sie das Optionskästchen, um den Istwert des Gebers zu invertieren.

Automatische Übernahme der Geberparameter im Gerät

Beim Einsatz von Gebern am schnellen Zähler (HSC) ist diese Auswahl nicht möglich.

Gebertyp

Wählen Sie im Feld "Gebertyp" den Gebertyp aus. Folgende Gebertypen sind wählbar:

- **Linear inkrementell**
- **Rotatorisch inkrementell**

Abhängig vom gewählten Gebertyp konfigurieren Sie verschiedene Parameter. Konfigurieren Sie je nach gewähltem Gebertyp folgende Parameter:

Gebertyp / Parameter		Beschreibung
Linear inkrementell		
	Abstand zwischen Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Schritten des Gebers.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).
Rotatorisch inkrementell		
	Schritte pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Schritte, die der Geber pro Umdrehung auflöst.
	Feinauflösung - Bits im inkr. Istwert (Gn_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gn_XIST1).

Erweiterte Parameter

Mechanik

Konfiguration - Mechanik - PTO (Pulse Train Output)

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die mechanischen Eigenschaften Ihres Antriebs.

Impulse pro Motorumdrehung

Konfigurieren Sie in diesem Feld, wie viele Impulse der Motor für eine Motorumdrehung benötigt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- 0 < Impulse pro Motorumdrehung ≤ 2147483647

Lastbewegung pro Motorumdrehung

Konfigurieren Sie in diesem Feld, welche Wegstrecke die Mechanik Ihrer Anlage pro Motorumdrehung zurücklegt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- 0.0 < Weg pro Motorumdrehung ≤ 1.0e12

Erlaubte Drehrichtung (Technologieversion ab V4)

Konfigurieren Sie in diesem Feld, ob sich die Mechanik Ihrer Anlage in beide Richtungen bewegen soll oder nur in positive oder negative Richtung.

Wenn Sie im Modus "PTO (Impuls A und Richtung B)" des Impulsgenerators den Richtungsausgang nicht aktiviert haben, ist die Auswahl auf die positive oder negative Richtung begrenzt.

Richtungssinn invertieren

Über das Optionskästchen "Richtungssinn invertieren" können Sie die Steuerung an die Richtungslogik des Antriebs anpassen.

Die Richtungslogik wird entsprechend dem ausgewählten Modus des Impulsgenerators invertiert:

- **PTO (Impuls A und Richtung B)**
  - 0 V am Richtungsausgang ⇒ positive Drehrichtung
  - 5 V/24 V am Richtungsausgang ⇒ negative Drehrichtung

  Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Werte gelten nicht für die Differenzausgänge der CPU 1217.
- **PTO (Vorwärts takten A, rückwärts takten B)**
  Die Ausgänge "Impulsausgang rückwärts" und "Impulsausgang vorwärts" werden vertauscht.
- **PTO (A/B phasenversetzt)**
  Die Ausgänge "Phase A" und "Phase B" werden vertauscht.
- **"PTO (A/B phasenversetzt - vierfach)**
  Die Ausgänge "Phase A" und "Phase B" werden vertauscht.

Konfiguration - Mechanik - PROFIdrive / Analoge Antriebsanbindung

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die mechanischen Eigenschaften Ihres Antriebs sowie Ihres Gebers.

Geberanbauart

Wählen Sie in der Klappliste, wie der Geber an der Mechanik angebaut ist. Folgende Geberanbauarten sind möglich:

- **An der Motorwelle**
- **Externes Messsystem** (nur rotatorische Geber)

Positionsparameter

Abhängig von der gewählten Geberanbauart konfigurieren Sie folgende Positionsparameter:

Geberanbauart / Positionsparameter		Beschreibung
An der Motorwelle		
	Lastbewegung pro Motorumdrehung	Konfigurieren Sie in diesem Feld den Weg der Last für eine Motorumdrehung.
Externes Messsystem		

Geberanbauart / Positionsparameter		Beschreibung
	Lastbewegung pro Motorumdrehung	Konfigurieren Sie in diesem Feld den Weg der Last für eine Motorumdrehung.
	Weg pro Geberumdrehung	Konfigurieren Sie in diesem Feld den vom externen Messsystem erfassten Weg pro Geberumdrehung.

Konfiguration - Modulo (nur PROFIdrive / analoge Antriebsanbindung)

Verwenden Sie die Einstellung "Modulo", wenn Sie den Verfahrbereich auf eine wiederkehrende Strecke, die sich an der Produktlänge / am Produktzyklus orientiert, beschränken möchten. Die Funktion Modulo ist nur im lagegeregelten Betrieb der Achse möglich.

Bei aktiviertem "Modulo" wird der Positionswert des Technologieobjekts auf einen sich wiederholenden Modulo-Bereich abgebildet. Der Modulo-Bereich ist durch den Startwert und die Länge definiert.

Um z. B. den Positionswert einer Achse auf eine volle Kreisbewegung zu beschränken, definieren Sie den Modulo-Bereich mit Startwert = 0° und Länge = 360°. Bei einer Geberauflösung von 0,1°/Geberschritt wird der Positionswert auf den Modulo-Bereich 0,0° bis 359,9° abgebildet. Fährt die Achse in diesem Beispiel auf die Position 400°, so wird die Istposition 40° (400° bis 360°) angefahren.

Bei aktiviertem "Modulo" geben Sie die Verfahrrichtung an der Motion Control-Anweisung "MC_MoveAbsolute" über den Eingangsparameter "Direction" vor. Folgende Parameterwerte stehen zur Verfügung:

- 0: Das Vorzeichen der Geschwindigkeit (Parameter "Velocity") bestimmt die Bewegungsrichtung.
- 1: Zielposition wird in positiver Richtung angefahren.
- 2: Zielposition wird in negativer Richtung angefahren.
- 3: Die Technologie wählt, ausgehend von der aktuellen Position, den kürzesten Weg zur Zielposition.

Modulo aktivieren

Aktivieren Sie das Optionskästchen "Modulo aktivieren", um für die Achse ein wiederkehrendes Bezugssystem zu verwenden (z. B. 0,0° bis 359,9°).

Modulostartwert

Definieren Sie in diesem Feld, an welcher Position der Modulo-Bereich beginnt (z. B. 0°).

Modulolänge

Definieren Sie in diesem Feld die Länge des Modulo-Bereichs (z. B. 360°).

Positionsgrenzen

Anforderungen an Hardware-Endschalter

Verwenden Sie nur Hardware-Endschalter, die nach dem Anfahren dauerhaft geschaltet bleiben. Dieser Schaltzustand darf erst nach der Rückkehr in den zulässigen Verfahrbereich zurück genommen werden.

Siehe auch

Konfiguration - Positionsgrenzen (Seite 1403)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Konfiguration der Positionsgrenzen im Anwenderprogramm ändern (Seite 1407)

Konfiguration - Positionsgrenzen

Konfigurieren Sie im Konfigurationsfenster "Positionsgrenzen" die Hardware- und Software-Endschalter der Achse.

HW-Endschalter aktivieren

Aktivieren Sie mit diesem Optionskästchen die Funktion des unteren und oberen Hardware-Endschalters. Während einer aktiven Referenzpunktfahrt können die Hardware-Endschalter zur Richtungsumkehr eingesetzt werden. Details finden Sie in der Konfigurationsbeschreibung zum Referenzieren.

SW-Endschalter aktivieren

Aktivieren Sie mit diesem Optionskästchen die Funktion des unteren und oberen Software-Endschalters.

Hinweis

Aktivierte Software-Endschalter wirken nur bei referenzierter Achse.

Eingang unterer / oberer HW-Endschalter

Wählen Sie in der Klappliste den digitalen Eingang für den unteren, bzw. für den oberen Hardware-Endschalter aus.

Bei PTO-Achsen muss der Eingang alarmfähig sein. Bei Antriebsanbindung über PROFIdrive / analoge Antriebsanbindung erreichen Sie über alarmfähige Eingänge die kürzeste Reaktionszeit. Alternativ können Sie die Eingänge dem Prozessabbild "TPA OB Servo" zuordnen und erhalten damit eine Reaktionszeit in der Zykluszeit des "TPA OB Servo". Die Zuordnung zum Standardprozessabbild des Organisationsbausteins OB1 wird nicht empfohlen, da hier die größten Reaktionszeiten entstehen.

Als alarmfähige Eingänge für die HW-Endschalter stehen die digitalen On-board CPU-Eingänge und die digitalen Eingänge eines gesteckten Signalboards zur Auswahl.

Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt. Bei der Verwendung als HW-Endschalter kann dies zu unerwünschten Verzögerungen führen. Verringern Sie in diesem Fall für die entsprechenden Digitaleingänge die Filterzeit.

Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Pegelauswahl

Wählen Sie in der Klappliste den, bei angefahrenem Hardware-Endschalter an der CPU anstehenden, Signalpegel aus.

- Auswahl "Unterer Pegel" (Öffner)
  0 V (FALSE) am CPU-Eingang entspricht Hardware-Endschalter angefahren
- Auswahl "Oberer Pegel" (Schließer)
  5 V / 24 V (TRUE) am CPU-Eingang entspricht Hardware-Endschalter angefahren (die angegebene Spannung ist abhängig von der verwendeten Hardware)

Position unterer / oberer SW-Endschalter

Legen Sie in diesen Feldern den Positionswert des unteren und oberen Software-Endschalters fest.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- -1.0e12 ≤ Unterer SW-Endschalter ≤ 1.0e12
- -1.0e12 ≤ Oberer SW-Endschalter ≤ 1.0e12

Der Wert des oberen Software-Endschalters muss größer gleich dem Wert des unteren Software-Endschalters sein.

Siehe auch

Anforderungen an Hardware-Endschalter (Seite 1403)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Konfiguration der Positionsgrenzen im Anwenderprogramm ändern (Seite 1407)

Konfiguration - Referenzieren - Aktiv (Seite 1416)

Verhalten der Achse beim Ansprechen der Positionsgrenzen

Verhalten der Achse beim Anfahren der Hardware-Endschalter

Wenn ein Hardware-Endschalter angefahren wird, verhält sich die Achse je nach Antriebsanbindung unterschiedlich:

- Antriebsanbindung über PROFIdrive / Analogausgang
  Beim Anfahren eines Hardware-Endschalters wird die Achse gesperrt und abhängig von der Konfiguration am Antrieb abgebremst und zum Stillstand gebracht. Die Verzögerung müssen Sie im Antrieb ausreichend groß wählen, sodass die Achse sicher vor dem mechanischen Anschlag zum Stehen kommt.
- Antriebsanbindung über PTO (Pulse Train Output)
  Beim Anfahren der Hardware-Endschalter bremst die Achse mit der konfigurierten Notstopp-Verzögerung bis auf Stillstand ab. Die Notstopp-Verzögerung müssen Sie ausreichend groß wählen, sodass die Achse sicher vor dem mechanischen Anschlag zum Stehen kommt. Das Verhalten der Achse nach dem Anfahren der Hardware-Endschalter entnehmen Sie der nachfolgenden Darstellung:

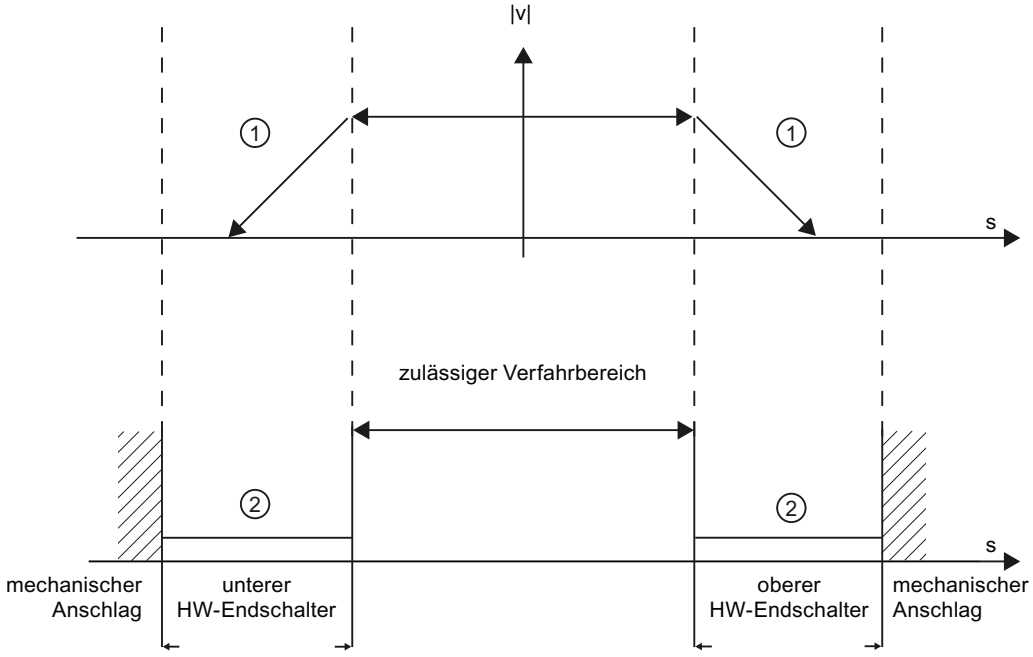

① Die Achse bremst mit der konfigurierten Notstopp-Verzögerung (PTO) oder mit der im Antrieb konfigurierten Verzögerung (PROFIdrive oder analoge Antriebsschnittstelle) bis auf Stillstand ab.

② Bereich in dem die HW-Endschalter den Status "angefahren" melden.

An der auslösenden Motion Control-Anweisung, am "MC_Power" und in den Variablen des Technologieobjekts wird der Fehler "HW-Endschalter angefahren" angezeigt. Eine Anleitung zur Beseitigung des Fehlers finden Sie im Anhang unter "Liste der ErrorIDs und ErrorInfos".

Verhalten der Achse beim Erreichen der Software-Endschalter

Eine laufende Bewegung kommt bei aktiven Software-Endschaltern auf der Position des Software-Endschalters zum Stehen. Die Achse wird mit der konfigurierten Verzögerung abgebremst.

Das Verhalten der Achse bis zum Erreichen der Software-Endschalter entnehmen Sie der nachfolgenden Darstellung:

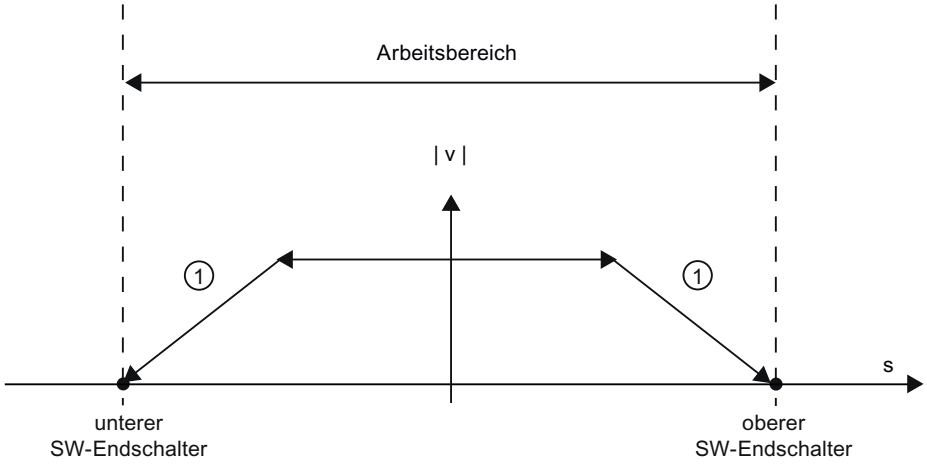

① Die Achse bremst mit der konfigurierten Verzögerung bis auf Stillstand ab.

An der auslösenden Motion Control-Anweisung, am "MC_Power" und in den Variablen des Technologieobjekts wird der Fehler "SW-Endschalter angefahren" angezeigt. Eine Anleitung zur Beseitigung des Fehlers finden Sie im Anhang unter "Liste der ErrorIDs und ErrorInfos".

Wenn ein Software-Endschalter überfahren wird, verhält sich die Achse je nach Antriebsanbindung unterschiedlich:

- Antriebsanbindung über PROFIdrive / Analogausgang
  Beim Überfahren eines Software-Endschalters wird die Achse gesperrt und abhängig von der Konfiguration am Antrieb abgebremst und zum Stillstand gebracht.
- Antriebsanbindung über PTO (Pulse Train Output)
  Das Verhalten der Achse beim Überfahren eines Software-Endschalters entnehmen Sie den Kapiteln "Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 1510)" und "Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 1515)".

Verwenden Sie zusätzlich Hardware-Endschalter, wenn sich hinter den Software-Endschaltern ein mechanischer Anschlag befindet und die Gefahr eines mechanischen Schadens besteht.

Siehe auch

Anforderungen an Hardware-Endschalter (Seite 1403)

Konfiguration - Positionsgrenzen (Seite 1403)

Konfiguration der Positionsgrenzen im Anwenderprogramm ändern (Seite 1407)

Konfiguration der Positionsgrenzen im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

Hardware-Endschalter

Sie können die Hardware-Endschalter auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsname>.PositionLimits_HW.Active

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1543) im Anhang.

Software-Endschalter

Sie können die Software-Endschalter auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren, bzw. deren Positionswerte ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsname>.PositionLimits_SW.Active
  zum Aktivieren und Deaktivieren der Software-Endschalter
- <Achsname>.PositionLimits_SW.MinPosition
  zum Ändern der Position des unteren Software-Endschalters
- <Achsname>.PositionLimits_SW.MaxPosition
  zum Ändern der Position des oberen Software-Endschalters

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1543)im Anhang.

Siehe auch

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Anforderungen an Hardware-Endschalter (Seite 1403)

Konfiguration - Positionsgrenzen (Seite 1403)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Dynamik

Konfiguration - Dynamik - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Dynamik Allgemein" die maximale Geschwindigkeit, die Start/Stopp-Geschwindigkeit, die Beschleunigung und Verzögerung als auch die Ruckbegrenzung (Technologieobjekt Positionierachse ab V2) der Achse.

Einheit der Geschwindigkeitsbegrenzung

Wählen Sie in der Klappliste die physikalische Einheit aus, mit der Sie die Geschwindigkeitsgrenzen festlegen möchten. Die hier eingestellte Einheit ist unabhängig von der unter "Konfiguration > Grundparameter > Allgemein" eingestellten Maßeinheit und dient nur zur vereinfachten Eingabe. So besteht z. B. die Möglichkeit, die maximale Geschwindigkeit als Drehzahlwert des Motors in U/min einzugeben.

Hinweis

Rundungsfehler

Wenn Sie in der Klappliste "Einheit der Geschwindigkeitsbegrenzung" eine andere Einheit als unter "Konfiguration > Grundparameter > Allgemein" auswählen, beachten Sie, dass Rundungsfehler auftreten können.

Maximale Geschwindigkeit/Start/Stopp-Geschwindigkeit

Definieren Sie in diesen Feldern die maximale zugelassene Geschwindigkeit und die Start/Stopp-Geschwindigkeit der Achse. Die Start/Stopp-Geschwindigkeit ist die minimal zulässige Geschwindigkeit der Achse und ist nur bei der Antriebsanbindung über PTO (Pulse Train Output) konfigurierbar.

Bei der Antriebsanbindung über PROFIdrive oder Analogausgang wird die Start/Stopp-Geschwindigkeit fest auf null gesetzt. Die maximale Geschwindigkeit beträgt bei der Anbindung über PROFdrive oder Analogausgang 1.0E12 der gewählten Maßeinheit (z. B. mm/s, °/s, ...).

Technologieobjekt Positionierachse (PTO) ab V4

Signalboard	Geschwindigkeit [Impuls/s]
20 kHz	1 ≤ Start/Stopp-Geschwindigkeit ≤ 20 000 1 ≤ Maximale Geschwindigkeit ≤ 20 000
200 kHz	1 ≤ Start/Stopp-Geschwindigkeit ≤ 200 000 1 ≤ Maximale Geschwindigkeit ≤ 200 000

On-board CPU-Ausgang	Geschwindigkeit [Impuls/s]
100 kHz	1 ≤ Start/Stopp-Geschwindigkeit ≤ 100 000 1 ≤ Maximale Geschwindigkeit ≤ 100 000
20 kHz	1 ≤ Start/Stopp-Geschwindigkeit ≤ 20 000 1 ≤ Maximale Geschwindigkeit ≤ 20 000
1 MHz CPU 1217	1 ≤ Start/Stopp-Geschwindigkeit ≤ 1000 000 1 ≤ Maximale Geschwindigkeit ≤ 1000 000

Die Grenzwerte für das Technologieobjekt Positionierachse < V4 entnehmen Sie dem Anhang Für Motion Control relevante Ausgänge der CPU (Technologieversion V1...3) (Seite 1581).

Der Wert der maximalen Geschwindigkeit muss größer gleich dem Wert der Start/Stopp-Geschwindigkeit sein.

Die Grenzwerte für andere Maßeinheiten müssen anwenderseitig entsprechend der gegebenen Mechanik umgerechnet werden.

Beschleunigung / Verzögerung - Hochlaufzeit / Rücklaufzeit

Stellen Sie die gewünschte Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$\textit{Hochlaufzeit} = \frac{\textit{Maximale Geschwindigkeit} - \textit{Start/Stopp-Geschwindigkeit}}{\textit{Beschleunigung}}$$

$$\textit{Rücklaufzeit} = \frac{\textit{Maximale Geschwindigkeit} - \textit{Start/Stopp-Geschwindigkeit}}{\textit{Verzögerung}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit der gewählten Beschleunigung / Verzögerung ausgeführt.

Die Grenzwerte für Beschleunigung und Verzögerung bei Antriebsanbindung über PTO (Pulse Train Output) finden Sie im Kapitel Für Motion Control relevante Ausgänge der CPU (Seite 1344).

Hinweis

Änderungen der Geschwindigkeitsgrenzen ("Start/Stopp-Geschwindigkeit", sowie "Maximale Geschwindigkeit") beeinflussen die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

Aktiviere Ruckbegrenzung, Technologieobjekt Positionierachse (ab V2)

Aktivieren Sie mit diesem Optionskästchen die Ruckbegrenzung.

Mit der Aktivierung der Ruckbegrenzung wird die Beschleunigung und Verzögerung der Achse nicht abrupt geändert, sondern entsprechend des eingestellten Rucks, bzw. der eingestellten Verrundungszeit sanft angeglichen.

Hinweis

Das Optionskästchen wird ab V4 nicht mehr als Parameter im Technologiedatenbaustein abgebildet. Eine Deaktivierung des Optionskästchens setzt den Ruckwert auf 0.0.

Verrundungszeit/Ruck, Technologieobjekt Positionierachse (ab V2)

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Ruck" ein.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

Hinweis

Verrundungszeit V2...3

Die eingestellte, und in der Konfiguration sichtbare Verrundungszeit gilt nur für die Beschleunigungsrampe.

Im Falle, dass sich die Werte von Beschleunigung und Verzögerung unterscheiden, wird die Verrundungszeit der Verzögerungsrampe entsprechend dem Ruck der Beschleunigungsrampe berechnet und verwendet. (siehe auch Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 1413)

Die Verrundungszeit der Verzögerung wird wie folgt angepasst:

- Beschleunigung > Verzögerung
  Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- Beschleunigung < Verzögerung
  Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- Beschleunigung = Verzögerung
  Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit dem gewählten Ruck ausgeführt.

Die Grenzwerte für den Ruck bei Antriebsanbindung über PTO (Pulse Train Output) finden Sie im Kapitel Für Motion Control relevante Ausgänge der CPU (Seite 1344).

Für PROFIdrive-Antriebe und Antriebe mit analoger Antriebsschnittstelle liegt der Grenzwert bei 1E12.

Siehe auch

Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 1413)

Hardwarekomponenten für Motion Control (Seite 1342)

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Konfiguration - Dynamik - Notstopp (Seite 1411)

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 1414)

Konfiguration - Dynamik - Notstopp

Konfigurieren Sie im Konfigurationsfenster "Dynamik Notstopp" die Notstopp-Verzögerung der Achse. Im Fehlerfall und beim Sperren der Achse mit der Motion-Control-Anweisung "MC_Power" (Eingangsparameter StopMode = 0 oder 2) wird die Achse mit dieser Verzögerung zum Stillstand gebracht.

Geschwindigkeit

In diesem Bereich werden die im Konfigurationsfenster "Dynamik Allgemein" parametrierten Geschwindigkeitswerte zur Veranschaulichung erneut angezeigt.

Verzögerung

Stellen Sie den Verzögerungswert für Notstopp in den Feldern "Notstopp-Verzögerung" oder "Notstopp-Rücklaufzeit" ein.

Den Zusammenhang zwischen Notstopp-Rücklaufzeit und Notstopp-Verzögerung können Sie der folgenden Gleichung entnehmen:

$$\textit{Notstopp-Rücklaufzeit} = \frac{\textit{Maximale Geschwindigkeit} - \textit{Start/Stopp-Geschwindigkeit}}{\textit{Notstopp-Verzögerung}}$$

Die Notstopp-Verzögerung muss ausreichend groß gewählt werden, um die Achse im Notfall rechtzeitig zum Stillstand zu bringen (z. B. beim Anfahren der Hardware-Endschalter, vor dem Erreichen des mechanischen Anschlags).

Bei der Wahl der Notstopp-Verzögerung muss die konfigurierte maximale Geschwindigkeit der Achse zugrunde gelegt werden.

Grenzwerte:

Die nachfolgend genannten Grenzwerte sind auf die Maßeinheit "Impulse/s^2" bezogen.

- ab CPU Firmware V3
  0.005 ≤ Notstopp-Verzögerung ≤ 9.5E9
- CPU Firmware V1...2
  0.28 ≤ Notstopp-Verzögerung ≤ 9.5E9

Die Grenzwerte für andere Maßeinheiten müssen entsprechend der gegebenen Mechanik umgerechnet werden.

Die Grenzwerte für den Ruck bei Antriebsanbindung über PTO (Pulse Train Output) finden Sie im Kapitel Für Motion Control relevante Ausgänge der CPU (Seite 1344).

Für PROFIdrive-Antriebe und Antriebe mit analoger Antriebsschnittstelle liegt der Grenzwert bei 1.0E12.

Siehe auch

Konfiguration - Dynamik - Allgemein (Seite 1408)

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 1414)

Verhalten der Achse beim Verwenden der Ruckbegrenzung

Mit der Aktivierung der Ruckbegrenzung wird die Beschleunigung und Verzögerung der Achse nicht abrupt geändert, sondern entsprechend des eingestellten Rucks, bzw. der eingestellten Verrundungszeit sanft angeglichen. Die nachfolgende Darstellung zeigt das Verhalten der Achse mit und ohne aktivierter Ruckbegrenzung im Detail.

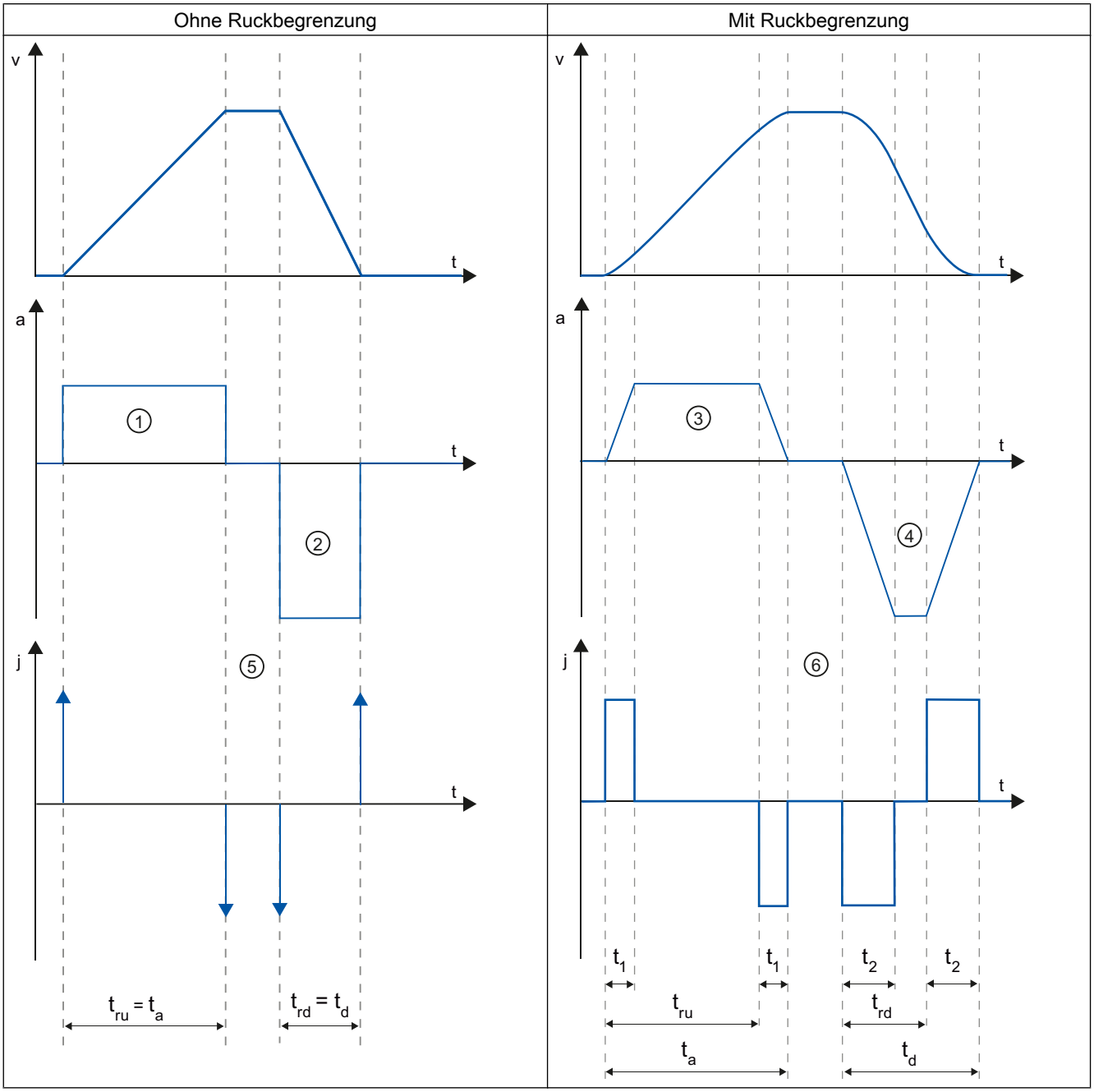

t	Zeitachse
v	Geschwindigkeit

a	Beschleunigung
j	Ruck
t_{ru}	Hochlaufzeit
t_a	Zeit zum Beschleunigen der Achse
t_{rd}	Rücklaufzeit
t_d	Zeit zum Verzögern der Achse
t_1	Verrundungszeit der Beschleunigungsrampe
t_2	Verrundungszeit der Verzögerungsrampe

Im Beispiel wird eine Verfahrbewegung dargestellt, bei der der Wert der Verzögerung ② doppelt so groß wie der der Beschleunigung ① ist. Hieraus ergibt sich eine Rücklaufzeit t_{rd}, die nur halb so groß wie die Hochlaufzeit t_{ru} ist.

Ohne Ruckbegrenzung wird die Beschleunigung ① und Verzögerung ② abrupt verändert. Mit aktivierter Ruckbegrenzung wird die Beschleunigung ③ und Verzögerung ④ sanft verändert. Da der Ruck für die gesamte Bewegung gilt, ergibt sich für die Zunahme und Abnahme der Beschleunigung und Verzögerung die gleiche Steigung.

Ohne Ruckbegrenzung werden die Werte für Ruck j im Moment der Änderung unendlich groß ⑤. Mit Ruckbegrenzung wird der Ruck auf den konfigurierten Wert ⑥ begrenzt.

Die in der Konfiguration angegebene Verrundungszeit t_1 gilt für die Beschleunigungsrampe. Die Verrundungszeit der Verzögerungsrampe t_2 wird aus dem konfigurierten Wert für Ruck und der konfigurierten Verzögerung berechnet.

Siehe auch

Konfiguration - Dynamik - Allgemein (Seite 1408)

Konfiguration der Dynamikwerte im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

Beschleunigung und Verzögerung

Sie können die Werte für Beschleunigung und Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsname>.DynamicDefaults.Acceleration
  zum Ändern der Beschleunigung
- <Achsname>.DynamicDefaults.Deceleration
  zum Ändern der Verzögerung

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1543) im Anhang.

Notstopp-Verzögerung

Sie können den Wert für die Notstopp-Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsname>.DynamicDefaults.EmergencyDeceleration

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

Hinweis

Nach dem Ändern dieses Parameters müssen Sie gegebenenfalls die Positionen der Hardware-Endschalter sowie weitere sicherheitsrelevante Einstellungen anpassen.

Ruckbegrenzung

Sie können die Ruckbegrenzung auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren und den Wert für den Ruck ändern. Verwenden Sie hierzu die Variable des Technologieobjekts <Achsname>.DynamicDefaults.Jerk. Bei Technologieobjekten < V4 muss die Variable <Achsname>.Config.DynamicDefaults.JerkActive auf TRUE gesetzt sein, damit die Ruckbegrenzung aktiviert ist und eine Wertänderung am Ruck damit sichtbar/wirksam wird.

Für PTO-Achsen gilt:

- Wenn Sie einen Wert ≥ 0.004 Impulse/s^3 für den Ruck eingeben, ist die Ruckbegrenzung mit dem eingegebenen Wert aktiviert.
- Wenn Sie einen Wert < 0.004 Impulse/s^3 für den Ruck eingeben, ist die Ruckbegrenzung deaktiviert.

Bei lagegeregelten Achsen ist die Ruckbegrenzung bei einem Wert 0.0 deaktiviert und bei Werten > 0.0 aktiviert.

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

Siehe auch

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Technologieobjekt "Achse" V1...3) (Seite 1593)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Konfiguration - Dynamik - Allgemein (Seite 1408)

Konfiguration - Dynamik - Notstopp (Seite 1411)

Referenzieren (Technologieobjekt Positionierachse ab V2)

Konfiguration - Referenzieren - Aktiv

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Aktiv" die für das aktive Referenzieren notwendigen Parameter. Das aktive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 3 gestartet.

Auswahl Referenziermodus (nur Antriebsanbindung über PROFIdrive ab V5)

Wählen Sie in der Auswahl einen der nachfolgenden Referenziermodi:

- Nullmarke über PROFIdrive-Telegramm und Näherungsschalter verwenden
- Nullmarke über PROFIdrive-Telegramm verwenden
- Referenzmarke über Digitaleingang verwenden

Wenn Sie die Antriebsanbindung über PTO (Pulse Train Output) oder analogen Ausgang mit HSC als Geber gewählt haben, steht ausschließlich der Referenziermodus "Referenzmarke über Digitaleingang verwenden" zur Verfügung.

Digitaleingänge

Konfigurieren Sie in diesem Bereich den Referenzpunktschalter:

- **Eingang Referenzpunktschalter**
  Wählen Sie in diesem Feld den digitalen Eingang für den Referenzpunktschalter.

  Hinweis

  Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

  Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

  Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

  Für Antriebsanbindung über PTO (Pulse Train Output):
  Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

- **Pegelauswahl**
  Wählen Sie in der Klappliste den Pegel des Referenzpunktschalters, mit dem referenziert werden soll.

- **Richtungsumkehr am Hardware-Endschalter erlauben**
  Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein (mindestens der Hardware-Endschalter in Anfahrrichtung muss konfiguriert sein).
  Wird der Hardware-Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.
  Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

  Hinweis

  Stellen Sie nach Möglichkeit durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:
  - Halten Sie die Anfahrgeschwindigkeit gering.
  - Vergrößern Sie die konfigurierte Beschleunigung/Verzögerung.
  - Vergrößern Sie den Abstand zwischen dem Hardware-Endschalter und dem mechanischen Anschlag.

Anfahr-/Referenzierrichtung

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die Anfahrrichtung zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest, mit welcher Fahrrichtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenziervorgang durchzuführen.

Seite des Referenzpunktschalters

Wählen Sie hier, ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Anfahrgeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Anfahrgeschwindigkeit ≤ Maximale Geschwindigkeit

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher in den Referenzpunktschalter zum Referenzieren eingefahren werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Referenziergeschwindigkeit ≤ Maximale Geschwindigkeit

Referenzpunktverschiebung

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- -1.0e12 ≤ Referenzpunktverschiebung ≤ 1.0e12

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Konfiguration - Referenzieren - Passiv

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Passiv" die für das passive Referenzieren notwendigen Parameter.

Zum passiven Referenzieren muss die Bewegung anwenderseitig (z. B. durch einen Fahrauftrag der Achse) angestoßen werden. Das passive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 2 gestartet.

Auswahl Referenziermodus (nur Antriebsanbindung über PROFIdrive ab V5)

Wählen Sie in der Auswahl einen der nachfolgenden Referenziermodi:

- **Nullmarke über PROFIdrive-Telegramm und Näherungsschalter verwenden**
  Das System prüft das Erreichen des Näherungsschalters. Nachdem der Näherungsschalter erreicht und in die parametrierte Referenzierrichtung wieder verlassen wurde, wird die Nullmarkenerfassung über das PROFIdrive-Telegramm aktiviert. Mit dem Erreichen der Nullmarke in der vorgewählten Richtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.
- **Nullmarke über PROFIdrive-Telegramm verwenden**
  Das System aktiviert die Nullmarkenerfassung, sobald sich der Istwert des Technologieobjekts in die parametrierte Referenzierrichtung bewegt. Mit dem Erreichen der Nullmarke in der vorgegebenen Referenzierrichtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.
- **Referenzmarke über Digitaleingang verwenden**
  Das System prüft den Zustand des Digitaleingangs, sobald sich der Istwert der Achse bzw. des Gebers in die parametrierte Referenzierrichtung bewegt. Mit dem Erreichen der Referenzmarke (Setzen des Digitaleingangs) in der vorgegebenen Referenzierrichtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.

Wenn Sie die Antriebsanbindung über PTO (Pulse Train Output) gewählt haben, wird standardmäßig eine Referenzmarke über einen Digitaleingang verwendet.

Digitaleingänge

Konfigurieren Sie in diesem Bereich den Referenzpunktschalter:

- **Eingang Referenzpunktschalter**
  Wählen Sie in diesem Feld den digitalen Eingang für den Referenzpunktschalter. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die Onboard CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

  Hinweis

  Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

  Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

  Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

- **Pegelauswahl**
  Wählen Sie in der Klappliste den Pegel des Referenzpunktschalters, mit dem referenziert werden soll.

Seite des Referenzpunktschalters

Wählen Sie hier, ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Hinweis

Erfolgt das passive Referenzieren ohne Fahrauftrag der Achse (Achse im Stillstand), so wird mit der nächsten steigenden oder fallenden Flanke am Referenzpunktschalter referenziert.

Ablauf - Aktives Referenzieren

Das Aktive Referenzieren starten Sie mit der Motion Control-Anweisung "MC_Home" (Eingangsparameter Mode = 3). Der Eingangsparameter "Position" gibt hierbei die absolute Referenzpunktposition an. Das aktive Referenzieren können Sie für Testzwecke alternativ in der Achssteuertafel starten.

Die folgende Darstellung zeigt beispielhaft den Verlauf einer aktiven Referenzpunktfahrt mit folgenden Konfigurationsparametern:

- "Referenziermodus" = "Referenzmarke über Digitaleingang verwenden"
- "Anfahr- / Referenzierrichtung" = "Positive Richtung"

- "Seite des Referenzpunktschalters" = "Obere Seite"
- Wert der "Referenzpunktverschiebung" > 0

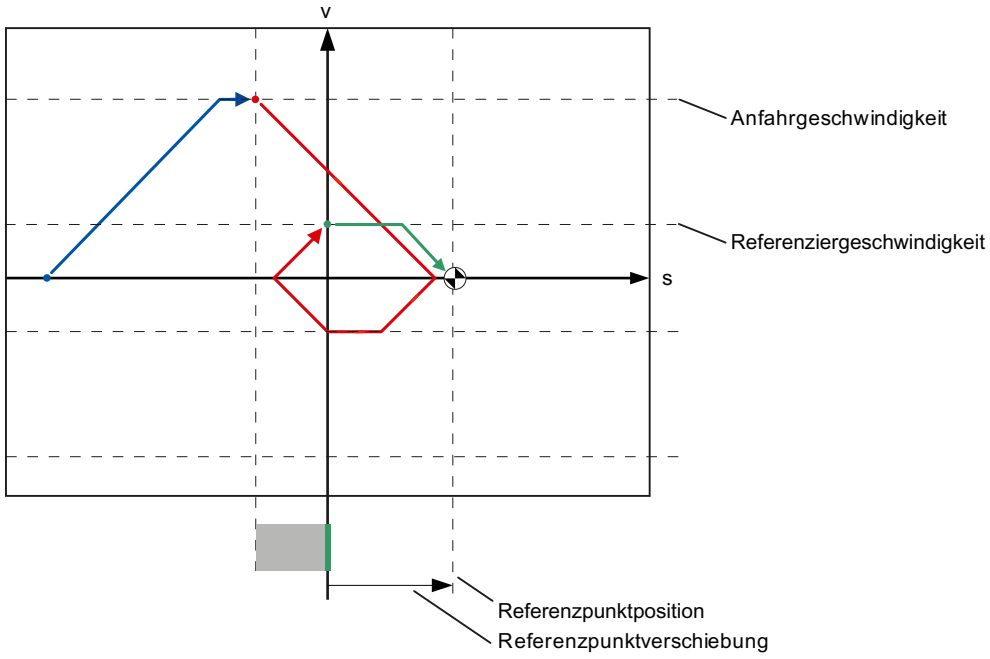

Suchen des Referenzpunktschalters (blauer Kurvenabschnitt)

Mit dem Start des Aktiven Referenzieren beschleunigt die Achse auf die konfigurierte "Anfahrgeschwindigkeit" und sucht mit dieser den Referenzpunktschalter. Die Variable <Achsname>.StatusBits.HomingDone wird auf FALSE gesetzt.

Referenzpunktfahrt (roter Kurvenabschnitt)

Mit dem Erkennen des Referenzpunktschalters bremst die Achse in diesem Beispiel ab, kehrt um, um mit der konfigurierten "Referenziergeschwindigkeit" auf der konfigurierten Seite des Referenzpunktschalters zu referenzieren. Mit dem Referenzieren wechselt die Variable <Achsname>.StatusBits.HomingDone nach TRUE.

Herausfahren der Referenzpunktverschiebung (grüner Kurvenabschnitt)

Nach dem Referenzieren verfährt die Achse mit der Referenziergeschwindigkeit die Wegstrecke der Referenzpunktverschiebung. Dort angekommen, befindet sich die Achse auf der Referenzpunktposition, die am Eingangsparameter "Position" der Motion Control-Anweisung "MC_Home" angegeben wurde.

Siehe auch

Konfiguration - Referenzieren - Allgemein (Technologieobjekt Achse V2...3) (Seite 1590)

Ablauf - Passives Referenzieren

Das Passive Referenzieren starten Sie mit der Motion Control-Anweisung "MC_Home" (Eingangsparameter Mode = 2). Der Eingangsparameter "Position" gibt die absolute Referenzpunktposition an.

Die folgende Darstellung zeigt beispielhaft den Verlauf eines passiven Referenziervorgangs mit dem folgenden Konfigurationsparameter:

- "Seite des Referenzpunktschalters" = "Obere Seite"
- "Referenziermodus" = "Referenzmarke über Digitaleingang verwenden"

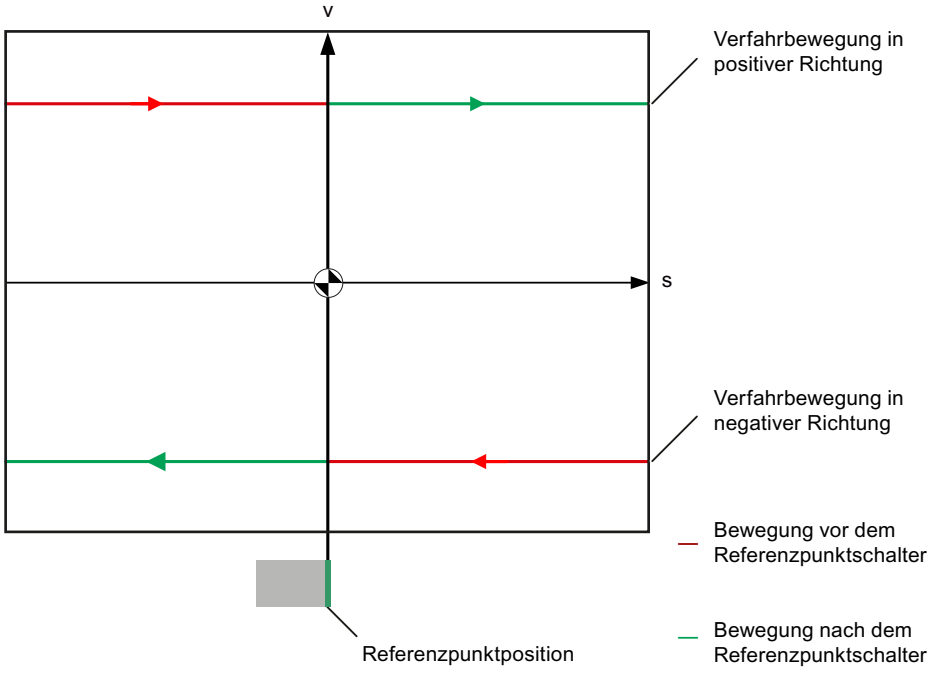

Bewegung vor dem Referenzpunktschalter (roter Kurvenabschnitt)

Mit dem Start des passiven Referenzierens führt die Motion Control-Anweisung "MC_Home" selbst keine Referenzierbewegung durch. Die zum Erreichen des Referenzpunktschalters notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen wie z. B. "MC_MoveRelative" realisiert werden. War die Achse zuvor schon referenziert, bleibt die Variable <Achsname>.StatusBits.HomingDone während des passiven Referenzierens TRUE.

Referenzieren der Achse (Übergang vom roten zum grünen Kurvenabschnitt)

Mit dem Erreichen der konfigurierten Seite des Referenzpunktschalters wird die Achse referenziert. Die aktuelle Position der Achse wird auf die Referenzpunktposition gesetzt. Diese wird am Parameter "Position" der Motion Control-Anweisung "MC_Home" angegeben. War die Achse zuvor noch nicht referenziert, so wird die Variable <Achsname>.StatusBits.HomingDone auf TRUE gesetzt. Die zuvor gestartete Verfahrbewegung wird nicht abgebrochen.

Bewegung nach dem Referenzpunktschalter (grüner Kurvenabschnitt)

Nach dem Referenzieren am Referenzpunktschalter führt die Achse die zuvor gestartete Verfahrbewegung mit der korrigierten Achsposition bis zum Abschluss fort.

Konfiguration zum Referenzieren im Anwenderprogramm ändern

Ab dem Technologieobjekt Positionierachse V2 können Sie folgende Konfigurationsparameter zur Laufzeit des Anwenderprogramms in der CPU ändern:

Passives Referenzieren

Sie können die Seite des Referenzpunktschalters für das passive Referenzieren zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsname>.Sensor[1].PassiveHoming.SideInput
  zum Ändern der Seite des Referenzpunktschalters
- <Achsname>.Sensor[1].PassiveHoming.Mode
  zum Ändern des Referenziermodus

Wann die Änderung des Konfigurationsparameters wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1652) im Anhang.

Aktives Referenzieren

Sie können die Anfahrrichtung, die Seite des Referenzpunktschalters, die Anfahrgeschwindigkeit, die Referenziergeschwindigkeit sowie die Referenzpunktverschiebung für das aktive Referenzieren zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsname>.Homing.AutoReversal
  zum Ändern der Richtungsumkehr am HW-Endschalter
- <Achsname>.Homing.ApproachDirection
  zum Ändern der Anfahr-/Referenzierrichtung
- <Achsname>.Sensor[1].ActiveHoming.SideInput
  zum Ändern der Seite des Referenzpunktschalters
- <Achsname>.Homing.ApproachVelocity
  zum Ändern der Anfahrgeschwindigkeit

- <Achsname>.Homing.ReferencingVelocity
  zum Ändern der Referenziergeschwindigkeit
- <Achsname>.Sensor[1].ActiveHoming.HomePositionOffset
  zum Ändern der Referenzpunktverschiebung
- <Achsname>.Sensor[1].ActiveHoming.Mode
  zum Ändern des Referenziermodus

Wann die Änderung des Konfigurationsparameters wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

Siehe auch

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Positionsüberwachungen

Konfiguration - Positionierüberwachung (nur PROFIdrive und analoge Antriebsanbindung)

Konfigurieren Sie im Konfigurationsfenster "Positionierüberwachung" die Kriterien für die Überwachung der Zielposition.

Die Positionierüberwachung überwacht das Verhalten der Istposition am Ende der Sollwertberechnung. Sobald der Geschwindigkeitssollwert den Wert Null erreicht, muss sich der Positionsistwert innerhalb einer Toleranzzeit im Positionierfenster befinden. Der Istwert darf während der Mindestverweildauer nicht das Positionierfenster verlassen.

Wenn die Istposition innerhalb der Toleranzzeit das Positionierfenster erreicht und für die Mindestverweildauer im Positionierfenster verbleibt, wird das Statusbit <Achsname>.StatusBits.Done gesetzt. Damit ist ein Bewegungsauftrag abgeschlossen.

Die Positionierüberwachung unterscheidet nicht, wie die Sollwertinterpolation beendet wird. Das Ende der Sollwertinterpolation wird z. B. folgendermaßen erreicht:

- Durch sollwertseitiges Erreichen der Zielposition
- Durch lagegeregeltes Anhalten während der Bewegung durch die Motion Control-Anweisung "MC_Halt"

In folgenden Fällen wird die Achse durch die Positionierüberwachung gestoppt und ein Positionierfehler (ErrorID 16#800F) an der Motion Control-Anweisung angezeigt:

- Der Istwert erreicht nicht innerhalb der Toleranzzeit das Positionierfenster.
- Der Istwert verlässt während der Mindestverweildauer das Positionierfenster.

Positionierfenster

Konfigurieren Sie in diesem Feld die Größe des Positionierfensters.

Toleranzzeit

Konfigurieren Sie in diesem Feld die Toleranzzeit, in welcher der Positionswert das Positionierfenster erreichen muss.

Minimale Verweildauer im Positionierfenster

Konfigurieren Sie in diesem Feld die minimale Verweildauer, für die sich der aktuelle Positionswert im Positionierfenster befinden muss.

Konfiguration - Schleppfehler (nur PROFIdrive und analoge Antriebsanbindung)

Konfigurieren Sie im Konfigurationsfenster "Schleppfehler" die zulässige Abweichung der Istposition der Achse zur Sollposition.

Der Schleppfehler ist die Differenz von Sollposition und Istposition der Achse. Die Übertragungszeiten des Sollwerts zum Antrieb und des Istwerts zur Steuerung werden bei der Berechnung des Schleppfehlers berücksichtigt.

Der Schleppfehler wird auf Basis einer geschwindigkeitsabhängigen Schleppfehlergrenze überwacht. Der zulässige Schleppfehler ist von der Sollgeschwindigkeit abhängig.

Bei Geschwindigkeiten kleiner als eine einstellbare untere Geschwindigkeit ist ein konstanter, zulässiger Schleppfehler vorgebbar. Oberhalb dieser unteren Geschwindigkeit wird der zulässige Schleppfehler proportional zur Sollgeschwindigkeit vergrößert. Bei maximaler Geschwindigkeit ist der maximale Schleppfehler zulässig.

Wenn der zulässige Schleppfehler überschritten wird, wird die Achse gestoppt und ein Fehler (ErrorID 16#800D) an der Motion Contro-Anweisung angezeigt.

Schleppfehlerüberwachung aktivieren

Aktivieren Sie das Optionskästchen, um die Schleppfehlerüberwachung zu aktivieren.

Bei aktivierter Schleppfehlerüberwachung wird die Achse im Fehlerbereich (orange) gestoppt.

Max. Schleppfehler

Konfigurieren Sie in diesem Feld den Schleppfehler, der bei maximaler Geschwindigkeit zulässig ist.

Schleppfehler

Konfigurieren Sie in diesem Feld den für geringe Geschwindigkeiten zulässigen Schleppfehler (ohne dynamische Anpassung).

Beginn der dynamischen Anpassung

Konfigurieren Sie in diesem Feld, ab welcher Geschwindigkeit der Schleppfehler dynamisch angepasst wird. Ab dieser Geschwindigkeit wird der Schleppfehler bis zur maximalen Geschwindigkeit auf den maximalen Schleppfehler angepasst.

Maximale Geschwindigkeit

Dieses Feld zeigt die unter "Dynamik > Allgemein" konfigurierte maximale Geschwindigkeit.

Konfiguration - Stillstandssignal (nur PROFIdrive und analoge Antriebsanbindung)

Konfigurieren Sie im Konfigurationsfenster "Stillstandssignal" die Kriterien zur Stillstandserkennung.

Zur Anzeige des Stillstands (<Achsname>.StatusBits.StandStill) muss sich die Geschwindigkeit der Achse für die minimale Verweildauer im Stillstandsfenster befinden.

Stillstandsfenster

Konfigurieren Sie in diesem Feld die Größe des Stillstandsfensters.

Minimale Verweildauer im Stillstandsfenster

Konfigurieren Sie in diesem Feld die minimale Verweildauer im Stillstandsfenster.

Konfiguration - Regelkreis (nur PROFIdrive und analoge Antriebsanbindung)

Konfigurieren Sie im Konfigurationsfenster "Regelkreis" die Vorsteuerung und die Verstärkung Kv des Lageregelkreises.

Der Kv-Faktor wirkt sich auf folgende Kenngrößen aus:

- Positioniergenauigkeit und Halteregelung
- Gleichförmigkeit der Bewegung
- Positionierzeit

Je besser die konstruktiven Voraussetzungen der Achse sind (große Steifigkeit), desto größer können Sie den Kv-Faktor einstellen. Damit verringert sich der Schleppfehler und eine höhere Dynamik wird erreicht.

Die Funktion "Optimierung (Seite 1473)" unterstützt Sie bei der Ermittlung der optimalen Verstärkung für die Lageregelung der Achse.

Vorsteuerung

Konfigurieren Sie in diesem Feld die prozentuale Geschwindigkeitsvorsteuerung des Lageregelkreises.

Verstärkung (Kv-Faktor)

Konfigurieren Sie in diesem Feld die Verstärkung Kv des Lageregelkreises.

Parametersicht

Einführung in die Parametersicht

Die Parametersicht bietet Ihnen eine Gesamtübersicht über alle relevanten Parameter eines Technologieobjektes. Sie erhalten einen Überblick über die Parametereinstellungen und können diese komfortabel im Offline- und Online-Betrieb ändern.

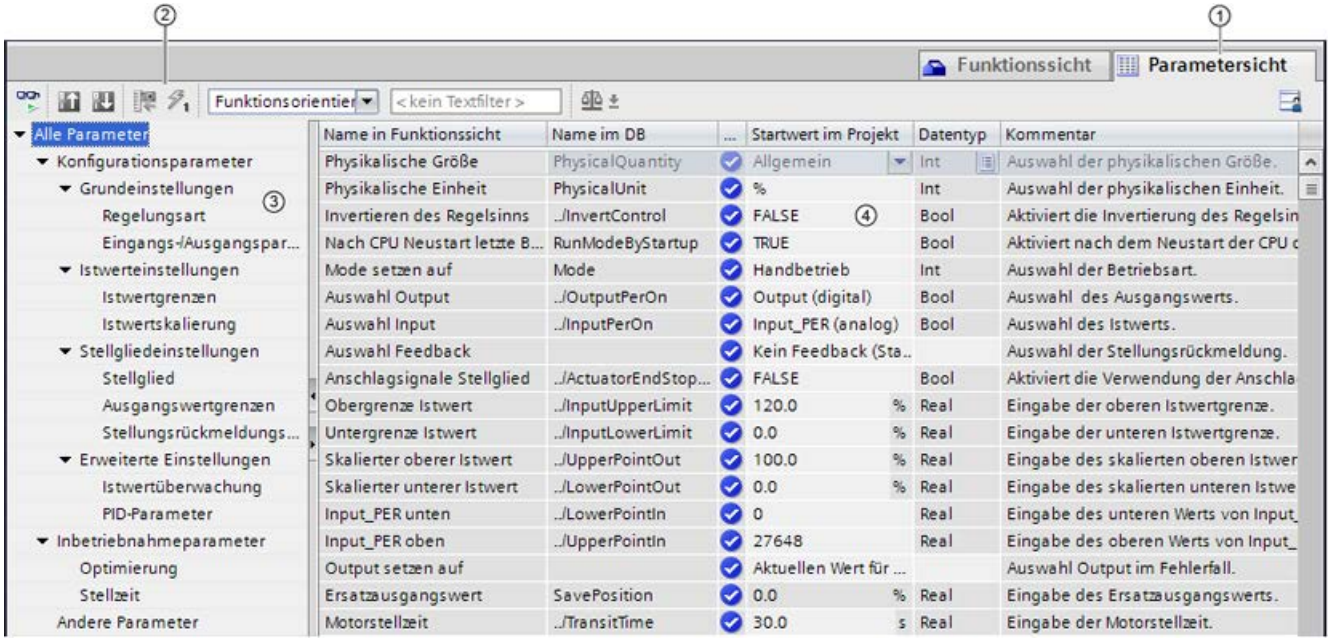

① Register "Parametersicht"

② Funktionsleiste (Seite 663)

③ Navigation (Seite 664)

④ Parametertabelle (Seite 664)

Funktionsumfang

Um die Parameter der Technologieobjekte zu analysieren und gezielt beobachten und steuern zu können, stehen folgende Funktionen zu Verfügung.

Anzeigefunktionen:

- Anzeige der Parameterwerte im Offline- und Online-Betrieb
- Anzeige von Statusinformationen der Parameter
- Anzeige von Werteabweichungen und Möglichkeit der direkten Korrektur
- Anzeige von Konfigurationsfehlern
- Anzeige von Wertänderungen in Folge von Parameterabhängigkeiten

- Anzeige aller Speicherwerte eines Parameters: Startwert in CPU, Startwert im Projekt, Beobachtungswert
- Anzeige des Parametervergleichs der Speicherwerte eines Parameters

Bedienfunktionen:

- Navigation, um schnell zwischen den Parametern und Parameterstrukturen zu wechseln.
- Text-Filter, um bestimmte Parameter schneller zu finden.
- Sortierfunktion, um die Reihenfolge von Parameter und Parametergruppen dem Bedarf anzupassen.
- Speicherfunktion, um strukturelle Einstellungen der Parametersicht zu sichern.
- Parameterwerte online beobachten und steuern.
- Anzeigeformat des Werts umstellen.
- Momentaufnahme von Parameterwerten der CPU speichern, um kurzzeitige Situationen abzubilden und darauf zu reagieren.
- Momentaufnahme von Parameterwerten als Startwerte übernehmen.
- Geänderte Startwerte in die CPU laden.
- Vergleichsfunktionen, um Parameterwerte miteinander zu vergleichen.

Gültigkeit

Die hier beschriebene Parametersicht steht für folgende Technologieobjekte zur Verfügung:

- PID_Compact
- PID_3Step
- PID_Temp
- CONT_C (nur S7-1500)
- CONT_S (nur S7-1500)
- TCONT_CP (nur S7-1500)
- TCONT_S (nur S7-1500)
- TO_Axis_PTO (S7-1200 Motion Control)
- TO_Positioning_Axis (S7-1200 Motion Control)
- TO_CommandTable_PTO (S7-1200 Motion Control)
- TO_CommandTable (S7-1200 Motion Control)

Aufbau der Parametersicht

Funktionsleiste

In der Funktionsleiste der Parametersicht sind folgende Funktionen anwählbar:

Symbol	Funktion	Erläuterung
	Alle beobachten	Startet das Beobachten der sichtbaren Parameter in der aktiven Parametersicht (Online-Betrieb).
	Momentaufnahme der Beobachtungswerte erstellen und Einstellwerte dieser Momentaufnahme als Startwerte übernehmen	Übernimmt die aktuellen Beobachtungswerte in die Spalte “Momentaufnahme“ und aktualisiert die Startwerte im Projekt. Nur im Online-Betrieb bei PID_Compact, PID_3Step und PID_Temp.
	Einstellwerte initialisieren	Überträgt die im Projekt aktualisierten Startwerte in die CPU. Nur im Online-Betrieb bei PID_Compact, PID_3Step und PID_Temp.
	Momentaufnahme der Beobachtungswerte erstellen	Übernimmt die aktuellen Beobachtungswerte in die Spalte “Momentaufnahme“. Nur im Online-Betrieb.
	Alle ausgewählten Parameter sofort und einmalig steuern	Dieser Befehl wird einmalig und schnellstmöglich ausgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm. Nur im Online-Betrieb.
Funktionsorientier	Navigationsstruktur auswählen	Wechselt zwischen der Funktionsorientierten Navigation und Datenorientierten Navigation.
< no text filter >	Textfilter...	Nach Eingabe einer Zeichenkette: Anzeige aller Parameter, die die eingegebene Zeichenkette in einer der aktuell sichtbaren Spalten enthalten.
	Vergleichswerte auswählen	Auswahl, welche Parameterwerte im Online-Betrieb miteinander verglichen werden sollen (Startwert im Projekt, Startwert in CPU, Momentaufnahme) Nur im Online-Betrieb.
	Anordnung merken	Speichert Ihre vorgenommenen Anzeigeeinstellungen der Parametersicht (z.B. ausgewählte Navigationsstruktur, aktivierte Tabellenspalten, usw.)

Navigation

Innerhalb des Registers "Parametersicht" sind folgende Navigationsstrukturen alternativ anwählbar:

Navigation		Erläuterung
Funktionsorientierte Navigation	Alle Parameter Konfigurationsparameter Inbetriebnahmeparameter Andere Parameter	In der Funktionsorientierten Navigation basiert die Struktur der Parameter auf der Struktur im Konfigurationsdialog (Register "Funktionssicht"), Inbetriebnahmedialog und Diagnosedialog. Die letzte Gruppe "Andere Parameter" enthält alle übrigen Parameter des Technologieobjektes.
Datenorientierte Navigation	Alle Parameter Input Output InOut Static Andere Parameter	In der Datenorientierten Navigation basiert die Struktur der Parameter auf der Struktur im Instanz-DB/Technologie-DB. Die letzte Gruppe "Andere Parameter" enthält diejenigen Parameter, die nicht im Instanz-DB/Technologie-DB enthalten sind.

Über die Klappliste "Navigationsstruktur auswählen" können Sie die Navigationsstruktur umstellen.

Parametertabelle

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten der Parametertabelle. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

- Spalte "Offline" = X: Spalte ist im Offline-Betrieb sichtbar.
- Spalte "Online"= X: Spalte ist im Online-Betrieb sichtbar (Online-Verbindung zur CPU).

Spalte	Erläuterung	Offline	Online
Name in Funktionssicht	Name des Parameters in der Funktionssicht. Das Anzeigefeld ist leer bei Parametern, die nicht über das Technologieobjekt konfiguriert werden.	X	X
Vollständiger Name im DB	Vollständiger Pfad des Parameters im Instanz-DB/Technologie-DB. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie-DB enthalten sind.	X	X
Name im DB	Name des Parameters im Instanz-DB/Technologie-DB. Wenn der Parameter Teil einer Struktur oder UDT ist, dann ist der Präfix ". ./" hinzugefügt. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/technologie-DB enthalten sind.	X	X
Status der Konfiguration	Anzeige der Vollständigkeit der Konfiguration durch Statussymbole. siehe Status der Konfiguration (offline) (Seite 672)	X	
Vergleichsergebnis	Ergebnis der Funktion "Werte vergleichen". Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist.		X
Startwert im Projekt	Konfigurierter Startwert im Projekt. Fehleranzeige bei syntaktisch oder technologisch falsch eingegebenen Werten.	X	X

Spalte	Erläuterung	Offline	Online
Defaultwert	Wert, mit dem der Parameter vorbelegt ist. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie-DB enthalten sind.	X	X
Momentaufnahme	Momentaufnahme der aktuellen Werte in der CPU (Beobachtungswerte). Fehleranzeige bei technologisch falschen Werten.	X	X
Startwert in CPU	Startwert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei technologisch falschen Werten.		X
Beobachtungswert	Aktueller Wert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei technologisch falschen Werten.		X
Steuerwert	Wert, mit dem der Beobachtungswert geändert werden soll. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und die Schaltfläche "Alles beobachten" angewählt ist. Fehleranzeige bei syntaktisch oder technologisch falsch eingegebenen Werten.		X
Auswahl Steuerwert	Auswahl der Steuerwerte, die mit Hilfe der Schaltfläche "Alle ausgewählten Parameter sofort und einmalig steuern" übertragen werden sollen. Diese Spalte wird zusammen mit der Spalte "Steuerwert" eingeblendet.		X
Minimalwert	Technologisch kleinster Wert des Parameters. Ist der Minimalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: Von den Startwerten im Projekt. • Online: Von den Beobachtungswerten.	X	X
Maximalwert	Technologisch höchster Wert des Parameters. Ist der Maximalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: Von den Startwerten im Projekt. • Online: Von den Beobachtungswerten.	X	X
Einstellwert	Kennzeichnet den Parameter als Einstellwert. Diese Parameter können online initialisiert werden.	X	X
Datentyp	Datentyp des Paramters. Das Anzeigefeld ist leer bei Parametern, die nicht im Instanz-DB/Technologie enthalten sind.	X	X
Remanenz	Kennzeichnet den Wert als remanent. Die Werte remanenter Parameter bleiben auch nach Ausschalten der Versorgungsspannung erhalten.	X	X
Erreichbar aus HMI	Zeigt an, ob HMI zur Laufzeit auf diesen Parameter zugreifen kann.	X	X
Sichtbar in HMI	Zeigt an, ob der Parameter in der Auswahlliste von HMI per Voreinstellung sichtbar ist.	X	X
Kommentar	Kurzbeschreibung des Parameters.	X	X

Siehe auch

Werte vergleichen (Seite 658)

Parametersicht öffnen

Voraussetzung

Das Technologieobjekt ist im Projektnavigator hinzufügt, d.h. der zugehörige Instanz-DB/ Technologie-DB der Anweisung ist erzeugt.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Konfiguration".
4. Wählen Sie in der rechten oberen Ecke das Register "Parametersicht".

Ergebnis

Die Parametersicht wird geöffnet. In der Parametertabelle wird jeder angezeigte Parameter durch eine Tabellenzeile repräsentiert.

Die anzeigbaren Parametereigenschaften (Tabellenspalten) sind abhängig davon, ob die Parametersicht im Offline- oder Online-Betrieb arbeitet.

Zusätzlich können Sie gezielt einzelne Tabellenspalten ein- und ausblenden.

Siehe auch

Voreinstellung der Parametersicht (Seite 666)

Voreinstellung der Parametersicht

Voreinstellungen

Für ein effektives Arbeiten mit der Parametersicht können Sie die Parameterdarstellung anpassen und die vorgenommen Einstellungen speichern.

Es sind folgende Anpassungen möglich und speicherbar:

- Spalten ein- und ausblenden
- Spaltenbreite ändern
- Reihenfolge der Spalten ändern
- Navigation umschalten
- Parametergruppe in der Navigation wählen
- Vergleichswerte auswählen

Spalten ein- und ausblenden

Um Spalten in der Parametertabelle ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

oder

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle.
2. Wählen Sie im Kontextmenü den Befehl "Alle Spalten anzeigen", wenn alle Spalten des Offline- oder Online-Betriebes angezeigt werden sollen.

Einige Spalten können nur im Online-Betrieb eingeblendet werden: siehe Parametertabelle (Seite 664).

Spaltenbreite ändern

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Parametertabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form eines Kreuzes annimmt.
2. Doppelklicken Sie auf diese Stelle.

oder

1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Parametertabelle.
2. Klicken Sie auf
   - "Spaltenbreite optimieren" oder
   - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

Reihenfolge der Spalten ändern

Die Spalten der Parametertabelle sind beliebig platzierbar.

Um die Reihenfolge der Spalten zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Spaltenkopf und ziehen Sie ihn per Drag&Drop an die gewünschte Stelle.
   Wenn Sie die Maustaste loslassen, wird die Spalte an der neuen Position verankert.

Navigation umschalten

Um die Anzeigestruktur der Parameter umzuschalten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste “Navigationsstruktur auswählen“ die gewünschte Navigation aus:
   - Datenorientierte Navigation
   - Funktionsorientierte Navigation

Siehe auch Navigation (Seite 664).

Parametergruppe in der Navigation wählen

Innerhalb der gewählten Navigation können Sie wählen zwischen der Anzeige “Alle Parameter“ oder der Anzeige einer gewünschten unterlagerten Parametergruppe.

1. Klicken Sie in der Navigation auf die gewünschte Parametergruppe.
   In der Parametertabelle werden nur die Parameter der Parametergruppe angezeigt.

Vergleichswerte auswählen (online)

Um die Vergleichswerte für die Funktion “Werte vergleichen“ einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste “Vergleichswerte auswählen“ die gewünschten Vergleichswerte aus:
   - Startwert im Projekt / Startwert in CPU
   - Startwert im Projekt / Momentaufnahme
   - Startwert in CPU / Momentaufnahme

Defaultmäßig ist die Option “Startwert im Projekt / Startwert in CPU“ eingestellt.

Voreinstellung der Parametersicht speichern

Um obige Anpassungen der Parametersicht zu speichern, gehen Sie folgendermaßen vor:

1. Passen Sie die Parametersicht nach Ihren Bedürfnissen an.
2. Klicken Sie auf die Schaltfläche “Anordnung merken“ rechts oben in der Parametersicht.

Arbeiten mit der Parametersicht

Übersicht

Die folgende Tabelle gibt einen Überblick über die nachfolgend beschriebenen Funktionen der Parametersicht im Online- und Offline-Betrieb.

- Spalte "Offline" = X: Diese Funktion ist im Offline-Betrieb möglich.
- Spalte "Online" = X: Diese Funktion ist im Online-Betrieb möglich.

Funktion/Aktion	Offline	Online
Parametertabelle filtern (Seite 669)	X	X
Parametertabelle sortieren (Seite 670)	X	X
Parameterdaten in andere Editoren übernehmen (Seite 670)	X	X
Fehler anzeigen (Seite 671)	X	X
Startwerte im Projekt bearbeiten (Seite 671)	X	X
Status der Konfiguration (offline) (Seite 672)	X	
Werte in der Parametersicht online beobachten (Seite 673)		X
Momentaufnahme der Beobachtungswerte erstellen (Seite 675)		X
Werte steuern (Seite 676)		X
Werte vergleichen (Seite 677)		X
Werte aus dem Online-Programm als Startwerte übernehmen (Seite 678)		X
Einstellwerte im Online-Programm initialisieren (Seite 679)		X

Parametertabelle filtern

Die Parameter in der Parametertabelle können Sie auf folgende Arten filtern:

- Mit dem Textfilter
- Mit den Untergruppen der Navigation

Beide Filtermethoden sind gleichzeitig anwendbar.

Mit dem Textfilter

Gefiltert werden kann nach Texten, die in der Parametertabelle sichtbar sind. D.h. es kann nur nach Texten in angezeigten Parameterzeilen und eingeblendeten Spalten gefiltert werden.

1. Geben Sie im Eingabefeld “Textfilter...“ die gewünschte Zeichenkette ein, nach der gefiltert werden soll.
   Die Parametertabelle zeigt nur noch diejenigen Parameter an, in denen die Zeichenkette enthalten ist.

Die Textfilterung wird zurückgesetzt:

- Bei Anwahl einer anderen Parametergruppe in der Navigation.
- Bei Wechsel zwischen Datenorientierter und Funktionsorientierter Navigation.

Mit den Untergruppen der Navigation

1. Klicken Sie in der Navigation auf die gewünschte Parametergruppe, z.B. "Static".
   In der Parametertabelle werden nur noch die Static-Parameter angezeigt. Sie können bei einigen Gruppen der Navigation weitere Untergruppen anwählen.
2. Klicken Sie in der Navigation auf “Alle Parameter“, wenn alle Parameter wieder angezeigt werden sollen.

Parametertabelle sortieren

Die Werte der Parameter sind zeilenweise angeordnet. Die Parametertabelle ist nach jeder angezeigten Spalte sortierbar.

- In Spalten mit numerischen Werten wird nach der Höhe des numerischen Wertes sortiert.
- In Text-Spalten wird alphabetisch sortiert.

Spaltenweise sortieren

1. Positionieren Sie den Mauszeiger in der Kopfzelle der gewünschten Spalte.
   Der Hintergrund dieser Zelle wird blau markiert.
2. Klicken Sie auf den Spaltenkopf.

Ergebnis

Die ganze Parametertabelle wird nach der markierten Spalte sortiert. Im Spaltenkopf erscheint ein Dreieck mit Spitze nach oben.

Durch wiederholtes Klicken auf den Spaltenkopf wird die Sortierung wie folgt geändert:

- Symbol "▲": Parametertabelle wird aufsteigend sortiert.
- Symbol "▼": Parametertabelle wird absteigend sortiert.
- Kein Symbol: Die Sortierung wird wieder aufgehoben. Die Parametertabelle nimmt die defaultmäßige Anzeige an.

Bei der Sortierung wird der Präfix "../" in der Spalte "Name im DB" ignoriert.

Parameterdaten in andere Editoren übernehmen

Nach Anwahl einer ganzen Parameterzeile der Parametertabelle können Sie mit

- Drag&Drop
- <Strg+C>/<Strg+V>
- Kopieren/Einfügen per Kontextmenü

Parameter in folgende Editoren des TIA-Portals übernehmen:

- In den Programmeditor
- In die Beobachtungstabelle
- In die Signaltabelle für Trace

Der Parameter wird mit vollständigen Namen eingefügt: vgl. Angabe in Spalte "Vollständiger Name im DB".

Fehler anzeigen

Fehleranzeige

Parametrierfehler, die zu Übersetzungsfehlern führen (z.B. Grenzwertüberschreitung), werden in der Parametersicht angezeigt.

Bei jeder Eingabe eines Wertes in der Parametersicht wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt.

Fehlerhafte Werte werden angezeigt durch:

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)

und/oder

- Tabellenfeld mit rotem Hintergrund
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der erforderlichen Syntax (Format)

Übersetzungsfehler

Von der Fehlermeldung des Compilers kann direkt die Parametersicht (Funktionsorientierte Navigation) mit dem fehlerverursachenden Parameter geöffnet werden, bei denjenigen Parametern, die nicht im Konfigurationsdialog angezeigt werden.

Startwerte im Projekt bearbeiten

Mit der Parametersicht können Sie im Offline-Betrieb und Online-Betrieb die Startwerte im Projekt bearbeiten:

- Wertänderungen nehmen Sie in der Spalte “Startwert im Projekt“ der Parametertabelle vor.
- In der Spalte “Status der Konfiguration“ der Parametertabelle wird der Fortschritt der Konfiguration durch die bekannten Statussymbole aus dem Konfigurationsdialog des Technologieobjektes angezeigt.

Randbedingungen

- Wenn vom Parameter, dessen Startwert geändert wurde, andere Parameter abhängen, dann wird der Startwert der abhängigen Parameter ebenfalls angepasst.
- Wenn ein Parameter eines Technologieobjektes nicht editierbar ist, dann ist er auch in der Parametersicht nicht editierbar. Die Editierbarkeit eines Parameters kann auch von den Werten anderer Parameter abhängen.

Neue Startwerte definieren

Um Startwerte für Parameter in der Parametersicht festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Parametersicht des Technologieobjektes.
2. Tragen Sie in die Spalte "Startwert im Projekt" die gewünschten Startwerte ein. Der Wert muss dem Datentyp des Parameters entsprechen und darf den Wertebereich des Parameters nicht überschreiten.
   In den Spalten “Maximalwert“ und “Minimalwert“ sind die Grenzwerte des Wertebereichs sichtbar.

In der Spalte "Status der Konfiguration" wird der “Fortschritt“ der Konfiguration durch farbliche Symbole angezeigt.

Siehe auch Status der Konfiguration (offline) (Seite 672)

Nach der Anpassung der Startwerte und dem Laden des Technologieobjekts in die CPU, nehmen die Parameter beim Anlauf den definierten Wert ein, sofern sie nicht als remanent deklariert sind (Spalte “Remanenz“).

Fehleranzeige

Bei Eingabe eines Startwertes wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt:

Fehlerhafte Startwerte werden angezeigt durch

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)

und/oder

- Roten Hintergrund im Feld “Startwert im Projekt“
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der notwendigen Syntax (Format)

Fehlerhafte Startwerte korrigieren

1. Korrigieren Sie fehlerhafte Startwerte mit Hilfe der Informationen aus der Roll-out-Fehlermeldung.
   Rotes Fehlersymbol, roter Feldhintergrund und Roll-out-Fehlermeldung werden nicht mehr angezeigt.

Nur mit fehlerfreien Startwerten kann das Projekt erfolgreich übersetzt werden.

Status der Konfiguration (offline)

Der Status der Konfiguration wird durch Symbole angezeigt:

- In Spalte “Status der Konfiguration“ in der Parametertabelle
- In der Navigationsstruktur der Funktionsorientierten Navigation bzw. Datenorientierten Navigation

Symbol in Spalte “Status der Konfiguration“

Symbol	Bedeutung
	Der Startwert des Parameters entspricht dem Defaultwert und ist gültig. Es wurde noch kein Startwert vom Anwender definiert.
	Der Startwert des Parameters enthält einen vom Anwender definierten oder automatisch angepassten Wert. Der Startwert unterscheidet sich vom Defaultwert. Der Startwert ist fehlerfrei und gültig.
	Der Startwert des Parameters ist ungültig (syntaktischer oder technologischer Fehler). Das Eingabefeld ist rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.
	Nur bei S7-1200 Motion Control: Der Startwert des Parameters ist gültig, enthält jedoch Warnhinweise. Das Eingabefeld ist gelb hinterlegt.
	Der Parameter ist in der aktuellen Konfiguration nicht relevant.

Symbol in der Navigation

Die Symbole in der Navigation zeigen den “Fortschritt“ der Konfiguration in gleicher Weise an wie im Konfigurationsdialog des Technologieobjektes.

Siehe auch

Technologieobjekte konfigurieren (Seite 654)

Werte in der Parametersicht online beobachten

Sie können die Werte, die die Parameter des Technologieobjektes aktuell in der CPU einnehmen (Beobachtungswerte), direkt in der Parametersicht beobachten.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Der Parametersicht des Technologieobjekt ist geöffnet.

Vorgehen

1. Starten Sie die Beobachtung, indem Sie auf das Symbol klicken.
   Sobald die Parametersicht online ist, werden folgende Spalten zusätzlich angezeigt:
   - Vergleichsergebnis
   - Startwert in CPU
   - Beobachtungswert
   - Steuerwert
   - Auswahl Steuerwert

   Die Spalte "Beobachtungswert" zeigt die aktuellen Parameterwerte auf der CPU an.
   Bedeutung der weiteren Spalten: siehe Parametertabelle (Seite 664)
2. Sie beenden die Beobachtung, indem Sie erneut auf das Symbol klicken.

Anzeige

Alle ausschließlich online verfügbaren Spalten haben einen orangen Hintergrund:

- Werte in den hellorangen Zellen sind änderbar.
- Werte in Zellen mit dunklorangem Hintergrund sind nicht änderbar.

Anzeigeformat des Werts umstellen

Das Anzeigeformat des Werts ist über das Kontextmenü einer Tabellenzeile in der Parametersicht des Technologieobjekts auswählbar.

Das Anzeigeformat der folgenden Werte kann sowohl im Online- als auch im Offline-Betrieb umgestellt werden:

- Startwert im Projekt
- Startwert in CPU
- Maximalwert
- Minimalwert
- Momentaufnahme
- Beobachtungswert
- Defaultwert
- Steuerwert

Das eingestellte Anzeigeformat gilt für alle Werte der Tabellenzeile.

Folgende Anzeigeformate des Werts sind änderbar:

- Voreinstellung
- Hex
- Oktal
- Bin

- Dec (+/-)
- DEC

Abhängig von dem in der Parametersicht ausgewählten Parameter sind nur die unterstützten Anzeigeformate auswählbar.

Voraussetzungen

- Die Parametersicht des Technologieobjekts ist geöffnet.

Vorgehen

Um das Anzeigeformat des Werts umzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Tabellenzeilen, in der Sie das Anzeigeformat umstellen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigeformat".
3. Wählen Sie das gewünschte Anzeigeformat aus.

Hinweis

Um das Anzeigeformat eines bestimmten Datentyps in mehreren Tabellenzeilen zu ändern, sortieren Sie die Parametersicht nach diesem Datentyp. Anschließend selektieren Sie die erste und letzte Tabellenzeile mit diesem Datentyp bei gedrückter <Shift>-Taste und ändern das Anzeigeformat für die selektierten Tabellenzeilen.

Momentaufnahme der Beobachtungswerte erstellen

Sie können die aktuellen Werte des Technologieobjektes auf der CPU (Beobachtungswerte) sichern und in der Parametersicht anzeigen.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um die aktuellen Parameterwerte anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Parametersicht auf das Symbol “Momentaufnahme der Beobachtungswerte erstellen".

Ergebnis

Die aktuellen Beobachtungswerte werden einmalig in die Spalte "Momentaufnahme" der Parametertabelle übernommen.

Sie können die so "eingefrorenen" Werte analysieren, während die Beobachtungswerte in der Spalte "Beobachtungswerte" weiterhin aktualisiert werden.

Werte steuern

Mit der Parametersicht können Sie Werte des Technologieobjektes in der CPU steuern.

Sie können dem Parameter einmalig Werte zuweisen (Steuerwert) und diese sofort steuern. Beim Ausführen wird der Steuerauftrag schnellstmöglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR
Gefahr beim Steuern: Ein Verändern der Parameterwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.
- Der Parameter kann gesteuert werden (zugehöriges Feld in der Spalte "Steuerwert" hat hellorangen Hintergrund).

Vorgehen

Um Parameter sofort zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Steuerwerte in die Spalte “Steuerwerte“ der Parametertabelle ein.
2. Prüfen Sie, ob das Optionskästchen für das Steuern in der Spalte "Auswahl Steuerwert" aktiviert ist.
   Die Steuerwerte und zugehörige Optionskästchen abhängiger Parameter werden automatisch mit angepasst.
3. Klicken Sie auf das Symbol “Alle ausgewählten Parameter sofort und einmalig steuern".

Die ausgewählten Parameter werden einmalig und sofort mit den vorgegebenen Werten gesteuert und können in der Spalte "Bebachtungswerte" beobachtet werden. Die Optionskästchen für das Steuern in der Spalte "Auswahl Steuerwert" werden nach Durchführung des Steuerauftrags automatisch deaktiviert.

Fehleranzeige

Bei Eingabe eines Steuerwertes wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt:

Fehlerhafte Steuerwerte werden angezeigt durch

- Roten Hintergrund im Feld “Steuerwert“

und

- Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der notwendigen Syntax (Format)

Fehlerhafte Steuerwerte

- Technologisch falsche Steuerwerte können übertragen werden.
- Syntaktisch falsche Steuerwerte können **nicht** übertragen werden.

Werte vergleichen

Über Vergleichfunktionen können Sie die folgenden Speicherwerte eines Parameters vergleichen:

- Startwert im Projekt
- Startwert in CPU
- Momentaufnahme

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um die Startwerte auf den verschiedenen Zielsystemen zu vergleichen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Vergleichswerte auswählen".
   Eine Auswahlliste mit den Vergleichsoptionen wird geöffnet:
   - Startwert im Projekt - Startwert in CPU (Defaulteinstellung)
   - Startwert im Projekt - Momentaufnahme
   - Startwert in CPU - Momentaufnahme
2. Wählen Sie die gewünschte Vergleichsoption.
   Die gewählte Vergleichsoption wird wie folgt ausgeführt:
   - In den Kopfzellen der beiden zum Vergleich ausgewählten Spalten erscheint ein Waage-Zeichen.
   - In der Spalte "Vergleichsergebnis" wird das Ergebnis des Vergleichs der gewählten Spalten durch Symbole angezeigt.

Symbol in Spalte "Vergleichergebnis"

Symbol	Bedeutung
	Die Vergleichswerte sind gleich und fehlerfrei.
	Die Vergleichswerte sind ungleich und fehlerfrei.
	Mindestens einer der beiden Vergleichswerte ist technologisch oder syntaktisch falsch.
	Der Vergleich kann nicht durchgeführt werden. Mindestens einer der beiden Vergleichswerte ist nicht verfügbar (z.B. Momentaufnahme).
	Der Wert ist nicht sinnvoll vergleichbar, da er in einer der Konfigurationen nicht relevant ist.

Symbol in der Navigation

Die Symbole werden in gleicher Weise in der Navigation angezeigt, wenn mindestens für einen der Parameter unterhalb der angezeigten Navigationsstruktur das Vergleichsergebnis zutrifft.

Werte aus dem Online-Programm als Startwerte übernehmen

Um in einem Schritt optimierte Werte aus der CPU als Startwerte in das Projekt zu übernehmen, erzeugen Sie eine Momentaufnahme der Beobachtungswerte. Als "Einstellwert" markierte Werte der Momentaufnahme werden anschließend als Startwerte in das Projekt übernommen.

Voraussetzungen

- Das Technologieobjekt ist vom Typ "PID_Compact" oder "PID_3Step".
- Eine Online-Verbindung besteht.

- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.

Vorgehen

Um optimierte Werte aus der CPU zu übernehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Momentaufnahme der Beobachtungswerte erstellen und Einstellwerte dieser Momentaufnahme als Startwerte übernehmen".

Ergebnis

Die aktuellen Beobachtungswerte werden in die Spalte "Momentaufnahme" übernommen und deren Einstellwerte als neue Startwerte in die Spalte "Startwert im Projekt" kopiert.

Hinweis

Werte einzelner Parameter übernehmen

Sie können auch die Werte einzelner Parameter, die nicht als Einstellwert markiert sind, aus der Spalte "Momentaufnahme" in die Spalte "Startwerte im Projekt" übernehmen. Kopieren Sie dazu die Werte mit Hilfe der Befehle "Kopieren" und "Einfügen" aus dem Kontextmenü und fügen Sie sie in die Spalte "Startwert im Projekt" ein.

Einstellwerte im Online-Programm initialisieren

Sie können alle Parameter, die in der Parametersicht als "Einstellwert" markiert sind, in einem Schritt in der CPU mit neuen Werten initialisieren. Dabei werden die Startwerte aus dem Projekt in die CPU geladen. Die CPU bleibt im Betriebszustand "RUN".

Um Datenverlust auf der CPU bei Kaltstart oder Neustart (Warmstart) zu vermeiden, müssen Sie zusätzlich noch das Technologieobjekt in die CPU laden.

⚠ GEFAHR

Gefahr beim Ändern von Parameterwerten

Ein Verändern der Parameterwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Einstellwerte neu initialisieren.

Voraussetzungen

- Das Technologieobjekt ist vom Typ "PID_Compact" oder "PID_3Step".
- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Die Schaltfläche “Alle beobachten“ ist aktiviert.
- Die Parameter, die als "Einstellwert" markiert sind, verfügen über einen technologisch und syntaktisch fehlerfreien "Startwert im Projekt".

Vorgehen

Um alle Einstellwerte zu initialisieren, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Werte in die Spalte "Startwert im Projekt" ein.
   Achten Sie darauf, dass die Startwerte technologisch und syntaktisch fehlerfrei sind.
2. Klicken Sie auf das Symbol "Einstellwerte initialisieren".

Ergebnis

Die Einstellwerte in der CPU werden mit den Startwerten aus dem Projekt initialisiert.

Technologiemodule für Motion Control konfigurieren

Übersicht

Folgende Technologiemodule können als Geberanbindung in S7-1200 Motion Control eingesetzt werden.

ET 200 MP	ET 200 SP
TM Count 2x24V (Seite 1448)	TM Count 1x24V (Seite 1448)
TM PosInput 2 (Seite 1447)	TM PosInput 1 (Seite 1447)

Die Technologiemodule können zentral oder auch dezentral im System eingesetzt werden.

Wie die Technologiemodule als Geber parametriert werden, finden Sie in den nachfolgenden Beschreibungen:

Siehe auch

TM Count 1x24V / TM Count 2x24V (Seite 1448)

TM PosInput 1 / TM PosInput 2 (Seite 1447)

TM PosInput 1 / TM PosInput 2

Für den Einsatz mit S7-1200 Motion Control ist die Konfiguration der folgenden Parameter erforderlich:

Konfiguration	
Technologiemodul **TM PosInput 1 / TM PosInput 2**	**Technologieobjekt** **Achse**
Grundparameter > Kanal 0/1 > Betriebsmodus	–
Betriebsmodus "Positionserfassung für Motion Control" wählen	
Grundparameter > Kanal 0/1 > Modulparameter	**Grundparameter > Geber**
–	Geber und entsprechenden Kanal wählen
Signalart • Inkrementalgeber • Absolutwertgeber	Gebertyp entsprechend Konfiguration am Technologiemodul wählen • Inkrementell • Absolut
–	**Datenaustausch Geber**
	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	"Automatische Übernahme der Geberparameterwerte im Gerät" abwählen
	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems
Signalauswertung • Einfach • Zweifach • Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul • Inkrementalgeber: – 0 = Einfach, – 1 = Zweifach *oder* – 2 = Vierfach • Absolutwertgeber: – 0 (= Einfach)
Schritte pro Umdrehung eingeben	• Rotatorische Ausführung: Inkemente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben • Lineare Ausführung: Abstand pro Umdrehung eingeben
–	**Hardware-Schnittstelle > Datenaustausch Antrieb**
• Rotatorische Ausführung: Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben • Lineare Ausführung: Konfiguration nicht relevant	Bezugsdrehzahl eingeben

Konfiguration	
Technologiemodul **TM PosInput 1 / TM PosInput 2**	**Technologieobjekt** **Achse**
E/A-Adressen	–
Organisationsbaustein: MC-Servo	
Prozessabbild: TPA OB Servo	
"Taktsynchroner Betrieb" anwählen	

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/Technologieobjekt erforderlich

Siehe auch

Übersicht (Seite 1446)

TM Count 1x24V / TM Count 2x24V (Seite 1448)

TM Count 1x24V / TM Count 2x24V

Für den Einsatz mit S7-1200 Motion Control ist die Konfiguration der folgenden Parameter erforderlich:

Konfiguration	
Technologiemodul **TM Count 1x24V / TM Count 2x24V**	**Technologieobjekt** **Achse**
Grundparameter > Kanal 0/1 > Betriebsmodus	–
Betriebsmodus "Positionserfassung für Motion Control" wählen	
Grundparameter > Kanal 0/1 > Modulparameter	**Hardware-Schnittstelle > Geber**
–	Geber und entsprechenden Kanal wählen
Signalart • Inkrementalgeber	Gebertyp entsprechend Konfiguration am Technologiemodul wählen • Inkrementell
–	**Hardware-Schnittstelle > Datenaustausch Geber**
	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	"Automatische Übernahme der Geberparameterwerte im Gerät" abwählen
	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems
Signalauswertung • Einfach • Zweifach • Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul • 0 = Einfach • 1 = Zweifach • 2 = Vierfach

Konfiguration	
Technologiemodul **TM Count 1x24V / TM Count 2x24V**	**Technologieobjekt** **Achse**
Schritte pro Umdrehung eingeben	• Rotatorische Ausführung: Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben • Lineare Ausführung: Weg pro Inkrement eingeben
–	**Hardware-Schnittstelle > Datenaustausch Antrieb**
• Rotatorische Ausführung: Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben • Lineare Ausführung: Konfiguration nicht relevant	Bezugsdrehzahl eingeben
E/A-Adressen	–
Organisationsbaustein: MC-Servo	
Prozessabbild: TPA OB Servo	
"Taktsynchroner Betrieb" anwählen	

Siehe auch

Übersicht (Seite 1446)

TM PosInput 1 / TM PosInput 2 (Seite 1447)

6.1.1.6 Technologieobjekt Auftragstabelle

Verwendung des Technolgieobjekts Auftragstabelle

Verwenden Sie das Technologieobjekt "Auftragstabelle", wenn Sie mehrere Einzelaufträge zur Steuerung einer Achse zu Bewegungssequenzen zusammenfassen möchten. Das Technologieobjekt kann ab der Technologieversion V2 für Achsen mit Antriebsanindung über PTO (Pulse Train Output) eingesetzt werden.

Die Bewegungssequenz konfigurieren Sie in Tabellenform in einem Konfigurationsdialog.

Das Bewegungsprofil der Bewegungssequenz kann bereits vor dem Laden des Projekts in die CPU grafisch überprüft werden. Die erstellten Auftragstabellen können anschließend im Anwenderprogramm über die Motion Control-Anweisung "MC_CommandTable" mit einer Achse verbunden und verwendet werden. Die Auftragstabelle kann komplett oder nur zum Teil abgearbeitet werden.

Werkzeuge des Technologieobjekts Auftragstabelle

Für das Technologieobjekt "Auftragstabelle" steht Ihnen im TIA-Portal das Werkzeug "Konfiguration" zur Verfügung. Die nachfolgende Darstellung zeigt das Zusammenwirken des Werkzeugs mit dem Technologieobjekt:

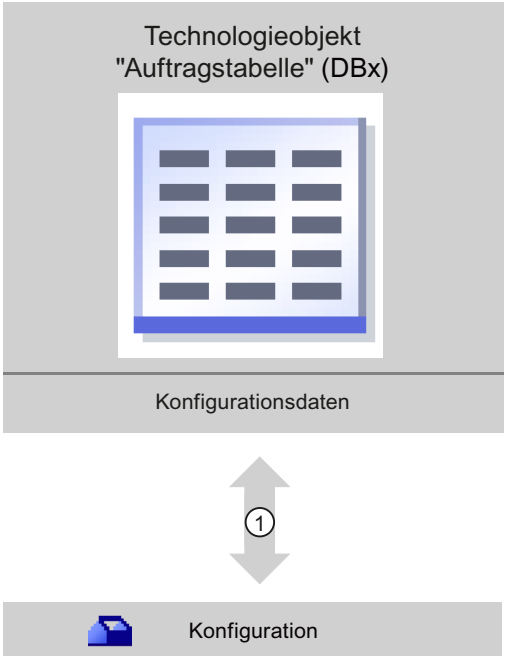

①	Schreiben und Lesen der Konfiguration des Technologieobjekts

Konfiguration

Konfigurieren Sie mit dem Werkzeug "Konfiguration" folgende Eigenschaften des Technologieobjekts "Auftragstabelle":

- Erstellen Sie eine oder mehrere Bewegungssequenzen durch Konfigurieren von Einzelaufträgen.
- Konfigurieren Sie die grafische Darstellung zur Überprüfung Ihrer Bewegungssequenz anhand einer bereits konfigurierten Achse oder einer konfigurierbaren Beispielachse.

Die Daten der Bewegungssequenz werden im Datenbaustein des Technologieobjekts gespeichert.

Technologieobjekt Auftragstabelle hinzufügen

Voraussetzungen

- Ein Projekt mit einer CPU S7-1200 ist angelegt.
- Die Firmwareversion der CPU ist größer oder gleich V2.1

Vorgehen

Um ein Technologieobjekt "Auftragstabelle" im Projektnavigator hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "CPU > Technologieobjekte".
2. Doppelklicken Sie auf den Befehl "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
3. Wählen Sie die Technologie "Motion Control".
4. Öffnen Sie den Ordner "Motion Control".
5. Wählen Sie in der Spalte "Version" die gewünschte Technologieversion.
6. Wählen Sie das Objekt "TO_CommandTable".
7. Geben Sie im Eingabefeld "Name" den Namen der Auftragstabelle ein.
8. Um die automatisch vergebene Nummer des Datenbausteins zu ändern, wählen Sie die Option "manuell".
9. Um weitere Informationen zum Technologieobjekt anzuzeigen und zu ergänzen, klicken Sie auf "Weitere Informationen".
10. Bestätigen Sie Ihre Eingabe mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

Technologieobjekt Auftragstabelle konfigurieren

Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Projektnavigator die Gruppe des gewünschten Technologieobjekts.
2. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**
  Die Grundparameter enthalten alle Parameter, die für eine funktionsfähige Auftragstabelle konfiguriert werden müssen.
- **Erweiterte Parameter**
  Die Erweiterten Parameter enthalten die Parameter der Beispielsachse, bzw. zeigen die Parameterwerte der gewählten Achse an.

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.
	Die Konfiguration enthält nicht miteinander vereinbare Parameterwerte Die Konfiguration enthält Parameterwerte, die sich entweder in ihrer Größe oder der Logik widersprechen. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt.

Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 1371)

Grundparameter (Seite 1453)

Erweiterte Parameter (Seite 1466)

Werte beobachten

Wenn eine Online-Verbindung zur CPU besteht, wird in den Konfigurationsdialogen des Technologieobjekts das Symbol "Alle beobachten" angezeigt.

Die Funktion "Alle beobachten" bietet folgende Möglichkeiten:

- Vergleich der konfigurierten Startwerte des Projekts mit den Startwerten in der CPU und den Aktualwerten
- Direktes Bearbeiten der Aktualwerte und der Startwerte des Projekts
- Sofortiges Erkennen und Anzeigen von Eingabefehlern mit Korrekturvorschlägen
- Sicherung der Aktualwerte im Projekt durch manuelle übernahme in den Startwert des Projekts

Symbole und Bedienelemente

Wenn eine Online-Verbindung zur CPU besteht, werden an den Parametern die Aktualwerte angezeigt.

Neben den Aktualwerten der Parameter werden folgende Symbole angezeigt:

Symbol	Beschreibung
	Startwert in CPU ist gleich dem konfigurierten Startwert im Projekt
	Startwert in CPU ist ungleich dem konfigurierten Startwert im Projekt

Symbol	Beschreibung
	Ein Vergleich des Startwert in CPU und des konfiguriertem Startwert im Projekt kann nicht durchgeführt werden, weil die gewählte CPU-Baugruppe diesen Vergleich nicht unterstützt.
	Der Wert ist nicht sinnvoll vergleichbar, da er in einer der Konfigurationen nicht relevant ist.
	Mit der Schaltfläche zeigen Sie für den jeweiligen Parameter den Startwert der CPU und den Startwert des Projekts an.

Den Aktualwert und den Startwert im Projekt können Sie direkt ändern und anschließend in die CPU laden. Bei direkt änderbaren Parametern wird die Änderung des Aktualwerts direkt in die CPU übernommen.

Grundparameter

Konfiguration - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" den Namen des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen der Auftragstabelle, bzw. den Namen des Technologieobjekts "Auftragstabelle". Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

Siehe auch

Konfiguration - Auftragstabelle (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Konfiguration - Auftragstabelle

Erstellen Sie im Konfigurationsfenster "Auftragstabelle" die gewünschte Bewegungssequenz und überprüfen Sie das Ergebnis anhand der grafischen Darstellung im Kurvendiagramm.

Hinweis

Geringe Abweichungen in der Darstellung des Zeitverhaltens und der Position im Kurvenverlauf zur realen Bewegung der Achse sind möglich. Bewegungsverläufe als Reaktion auf das Erreichen von SW-Endschaltern werden nicht dargestellt.

Warnungen aktivieren

Aktivieren Sie mit diesem Optionskästchen die Anzeige von Warnungen in der Auftragstabelle.

Verwende Achsparameter von

Wählen Sie in der Klappliste welche Achsparameter als Grundlage für die grafische Darstellung und die Überprüfung der Bewegungssequenz dienen sollen. Wählen Sie "Beispielachse", wenn Sie im Ordner "Technologieobjekte" noch keine Achse eingefügt haben oder Werte, die in keiner vorhandenen Achse konfiguriert wurden, verwenden möchten. Die Eigenschaften der Beispielachse konfigurieren Sie in unter "Erweiterte Parameter".

Für die Abarbeitung der Auftragstabelle im Anwenderprogramm werden die Achsparameter der am Parameter "Axis" gewählten Achse verwendet.

Spalte: Schritt

Zeigt die Schrittnummer des Auftrags an.

Spalte: Auftragstyp

Wählen Sie in dieser Spalte die Auftragtypen aus, welche bei der Abarbeitung der Auftragstabelle ausgeführt werden sollen. Es können bis zu 32 Aufträge eingetragen werden. Die Aufträge werden sequenziell abgearbeitet. Wählen Sie zwischen folgenden Einträgen und Auftragstypen:

- Empty
  Der Eintrag dient als Platzhalter für evtl. einzufügende Aufträge. Beim Abarbeiten der Auftragstabelle wir der Empty-Eintrag ignoriert.
- Halt
  Achse anhalten
  (der Auftrag wirkt nur nach einem "Velocity set point"-Auftrag)
- Positioning Relative
  Achse relativ positionieren
- Positioning Absolute
  Achse absolut positionieren (hierzu muss die Achse referenziert sein)
- Velocity set point
  Achse mit Geschwindigkeitsvorgabe verfahren
- Wait
  Wartet, bis die angegebene Dauer abgelaufen ist. Wait stoppt keine laufenden Verfahrbewegungen.
- Separator
  Fügt oberhalb der markierten Zeile eine Separator-Zeile ein. Die Separator-Zeile dient als Bereichsgrenze für die grafische Darstellung des Kurvendiagramms.
  Verwenden Sie Separator-Zeilen, wenn Sie Teile der Auftragstabelle abarbeiten möchten.

Spalte: Position/Verfahrweg

Geben Sie in dieser Spalte die Position, bzw. den Verfahrweg für den gewählten Auftrag an:

- **Auftrag "Positioning Relative"**
  Der Auftrag verfährt die Achse um den angegebenen Verfahrweg.
- **Auftrag "Positioning Absolute"**
  Der Auftrag fährt die Achse auf die angegebene Position.
  Hierzu muss die Achse referenziert sein.
- **Separator**
  Der angegebene Wert gibt die Startposition für die grafische Darstellung an.

Grenzwerte (unabhängig von der gewählten Anwendereinheit):

- -1.0e12 ≤ Position / Verfahrweg ≤ -1.0e-12
- 1.0e-12 ≤ Position / Verfahrweg ≤ 1.0e12
- Position / Verfahrweg = 0.0

Spalte: Geschwindigkeit

Geben Sie in dieser Spalte die Geschwindigkeit für den gewählten Auftrag an:

- **Auftrag "Positioning Relative"**
  Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.
  Ist der gewählte Verfahrweg nicht groß genug, so wird die angegebene Geschwindigkeit nicht erreicht.
- **Auftrag "Positioning Absolute"**
  Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.
  Liegt die Zielposition zu nahe an der Startposition, so wird die angegebene Geschwindigkeit nicht erreicht.
- **Auftrag " Velocity set point "**
  Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.
  Wird eine zu kurze Laufzeit gewählt, so wird die angegebene Geschwindigkeit während der Dauer des Auftrags nicht erreicht.

Grenzwerte (unabhängig von der gewählten Anwendereinheit):

- Für die Aufträge: "Positioning Relative" und "Positioning Absolute"
  - 1.0e-12 ≤ Geschwindigkeit ≤ 1.0e12
- Für den Auftrag: "Velocity set point"
  - -1.0e12 ≤ Geschwindigkeit ≤ -1.0e-12
  - 1.0e-12 ≤ Geschwindigkeit ≤ 1.0e12
  - Geschwindigkeit = 0.0

Spalte: Dauer

Geben Sie in dieser Spalte die Dauer für den gewählten Auftrag an:

- **Auftrag " Velocity set point "**
  Der Auftrag verfährt die Achse über die angegebene Dauer. Zur Dauer gehört sowohl die Beschleunigungsphase, als auch die Konstantfahrphase. Nach Ablauf der Dauer wir zum nächsten Auftrag übergegangen.
- **Auftrag "Wait"**
  Wartet, bis die angegebene Dauer abgelaufen ist.

Grenzwerte (unabhängig von der gewählten Anwendereinheit):

- 0.001s ≤ Dauer ≤ 64800s

Spalte: Nächster Schritt

Wählen Sie in der Klappliste den Übergangsmodus zum nächsten Schritt:

- **Auftrag abschließen**
  Der Auftrag wird abgeschlossen. Der nächste Auftrag wird ohne zeitliche Lücke angehängt.
- **Bewegung überschleifen**
  Die Bewegung des aktuellen Auftrags wird mit der Bewegung des nachfolgenden Auftrags überschliffen. Der Übergangsmodus "Bewegung überschleifen" ist beim Auftragstyp "Positioning Relative" und "Positioning Absolute" auswählbar.
  Die Bewegung wird mit Bewegungen folgender Auftragstypen überschliffen:
  - Positioning Relative
  - Positioning Absolute
  - Velocity set point

Mit anderen Auftragstypen wird nicht überschliffen.

Das detaillierte Verhalten der Achse beim Anhängen bzw. Überschleifen eines Auftrags entnehmen Sie dem Kapitel: Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Spalte: Schrittcode

Geben Sie in dieser Spalte einen Zahlenwert/Bitmuster ein, die am Ausgangsparameter "StepCode" der Motion Control-Anweisung "MC_CommandTable" während der Bearbeitung des Auftrags ausgegeben werden soll.

Grenzwerte:

- 0 ≤ Kennziffer ≤ 65535

Siehe auch

Konfiguration - Allgemein (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Kontextmenübefehle - Auftragstabelle

In der Auftragstabelle stehen folgende Kontextmenübefehle zur Verfügung:

Empty-Zeile einfügen

Fügt oberhalb der markierten Zeile eine Empty-Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.

Empty-Zeile hinzufügen

Fügt unterhalb der markierten Zeile eine Empty-Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.

Separator-Zeile einfügen

Fügt oberhalb der markierten Zeile eine Separator-Zeile ein.

Zwei aufeinander folgende Separator-Zeilen sind nicht möglich.

Separator-Zeile hinzufügen

Fügt unterhalb der markierten Zeile eine Separator-Zeile ein.

Zwei aufeinander folgende Separator-Zeilen sind nicht möglich; das Hinzufügen eine Separator-Zeile an das Ende der Auftragstabelle ist ebenfalls nicht möglich.

Ausschneiden

Entfernt die markierten Zeilen oder den Inhalt der markierten Zelle und legt sie in die Zwischenablage.

Markierte Zeilen werden entfernt; nachfolgende Zeilen der Auftragstabelle werden nach oben verschoben.

Kopieren

Kopiert die markierten Zeilen oder den Inhalt der markierten Zelle und legt sie in die Zwischenablage.

Einfügen

- Markierte Zeilen:
  Fügt die Zeilen der Zwischenablage oberhalb der markierten Zeile ein.
- Markierte Zelle:
  Fügt den Inhalt der Zwischenablage in die markierte Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.

Ersetzen

Ersetzt die markierten Zeilen durch die Zeilen der Zwischenablage.

Löschen

Löscht die markierten Zeilen. Nachfolgende Zeilen der Auftragstabelle werden nach oben verschoben.

Siehe auch

Konfiguration - Allgemein (Seite 1453)

Konfiguration - Auftragstabelle (Seite 1453)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Arbeiten mit dem Kurvendiagramm

Im Kurvendiagramm stehen Ihnen folgende Werkzeuge und Informationen zur Verfügung:

Ansicht und Komponenten des Kurvendiagramms

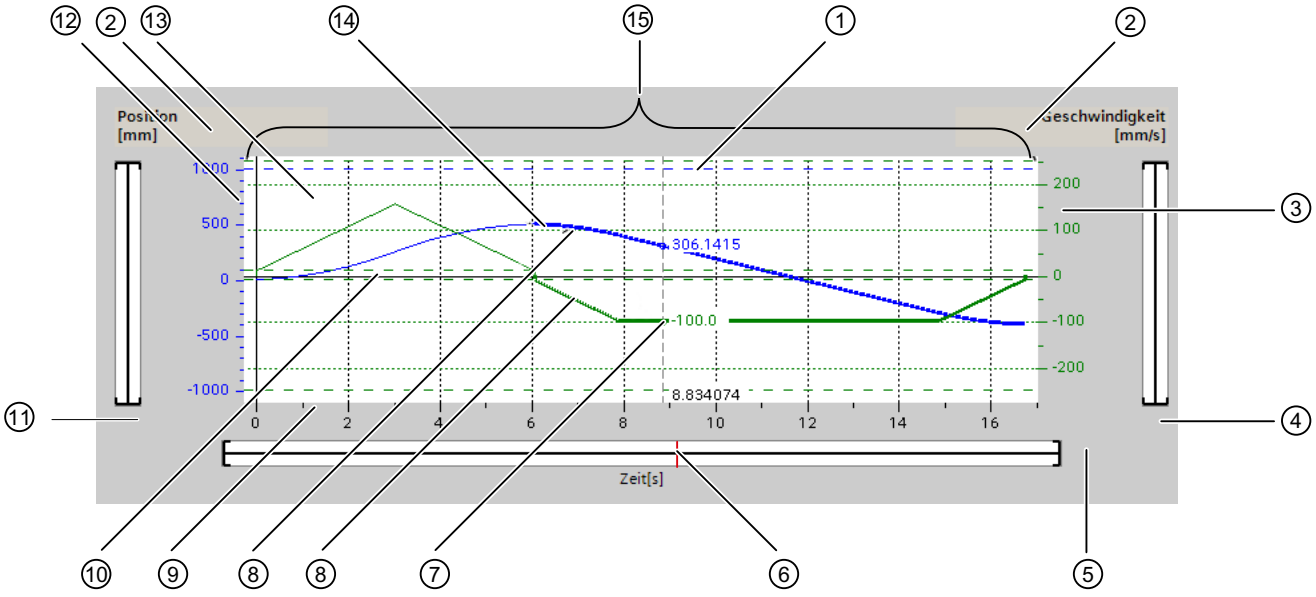

①	Messlineal
②	Wahl des Gitternetzes
③	Skalenbereich Geschwindigkeitsachse
④	Verschiebebalkenbereich Geschwindigkeitsachse
⑤	Verschiebebalkenbereich Zeitachse
⑥	Positionsmarkierung des Messlineals
⑦	Geschwindigkeitskurve
⑧	Kurvenabschnitt eines selektierten Auftrags
⑨	Skalenbereich Zeitachse
⑩	Start/Stopp-Geschwindigkeit
⑪	Verschiebebalkenbereich Positionsachse
⑫	Skalenbereich Positionsachse
⑬	Position der SW-Endschalter
⑭	Positionskurve
⑮	Kurvenfenster

Separator-Abschnitte auswählen

Besteht die Auftragstabelle aus mehreren durch Separator getrennten Abschnitten, so können die Abschnitte im Kurvendiagramm durch Auswahl eines Auftrags innerhalb des Abschnitts ausgewählt werden.

Aufträge auswählen

Aufträge können im Kurvendiagramm und in der Auftragstabelle ausgewählt werden:

- Klicken Sie im Kurvendiagramm auf einen Punkt der Geschwindigkeits- oder Positionskurve. Der dazugehörige Auftrag wird in der Auftragstabelle markiert.
- Markieren Sie in der Auftragstabelle einen Auftrag.
  Die entsprechenden Kurvenbereiche werden im Kurvenbereich hervorgehoben.

Sichtbaren Bereich des Kurvendiagramms auswählen

Gehen Sie wie nachfolgend beschrieben vor, um den darzustellenden Ausschnitt des Kurvendiagramms anzupassen:

Auswahl der Skalierung im Kontextmenü:

- Skalierung auf Kurven:
  Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve sichtbar ist.
- Skalierung auf Kurven und Grenzen:
  Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve, die Positionen der aktivierten Software-Endschalter und die minimale und maximale Geschwindigkeitsgrenze sichtbar ist.

Die jeweils ausgewählte Ansicht wird im Kontextmenü mit einem Haken markiert.

Darzustellenden Ausschnitt im Skalenbereich wählen:

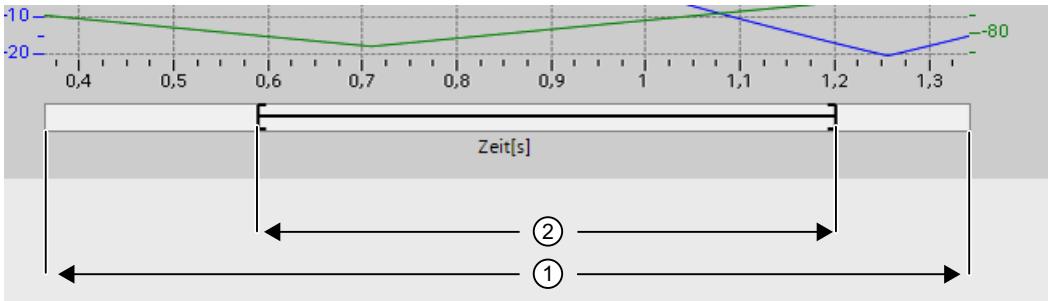

Bild 6-1 CmdTable_Scle01_neu

①	Bereich in dem sich Kurvenwerte und / oder Grenzwerte befinden. (siehe Auswahl im Kontextmenü)
②	Ausgewählter Bereich, der im Kurvenfenster dargestellt wird. Den Auswahlbereich stellen Sie am rechten und linken Rand mit dem Randcursor ein. Die Lage innerhalb des Bereichs ① stellen Sie mit dem Verschiebecursor ein. Die Lage kann auch durch Klick in Bereich ① bestimmt werden. 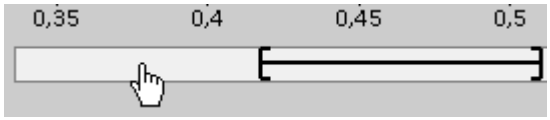

Darzustellenden Ausschnitt mit der Maus wählen:

Ziehen Sie mit der Maus durch klicken und ziehen einen Bereich im Kurvendiagramm auf. Nach dem Loslassen der Maus wird die Darstellung des Kurvenbereichs auf die Auswahl vergrößert.

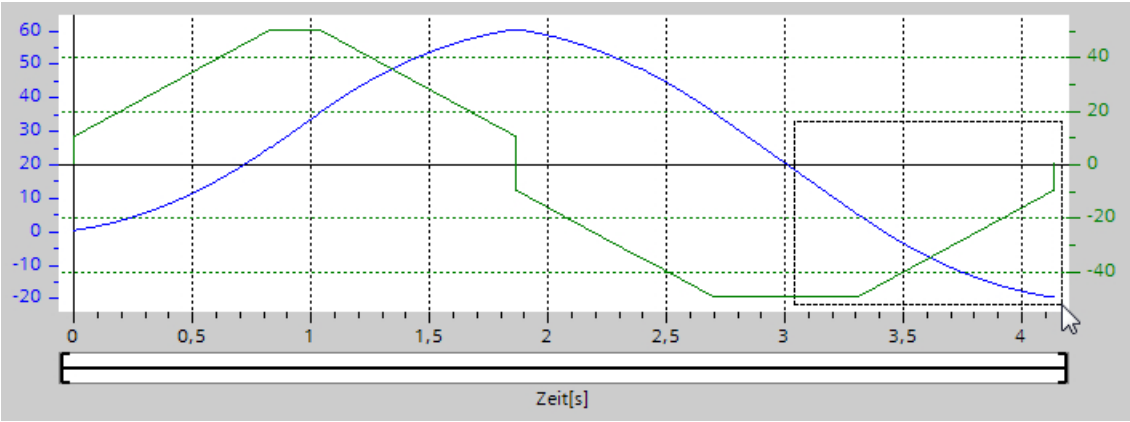

Letzte Änderung des Ausschnittes zurücknehmen:

Wählen Sie den Kontextmenübefehl "Zoom rückgängig", um die letzte Änderung des Ausschnitts zurück zu nehmen.

Gitternetz synchronisieren

Wählen Sie durch Klicken der Achsskalen, ob das Gitternetz mit der Positionsachse oder mit der Geschwindigkeitsachse synchronisiert werden soll.

Kurvenwerte am Messlineal auslesen

Aktivieren Sie das Messlineal über den Kontextmenübefehl "Messlineal einblenden".

Mit dem Messlinealcursor bewegen Sie das Messlineal auf beliebige Stellen Ihrer Kurven.

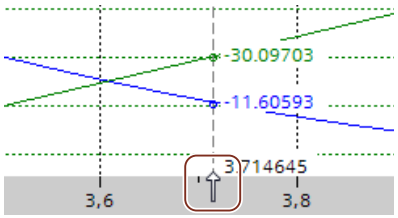

Siehe auch

Konfiguration - Allgemein (Seite 1453)

Konfiguration - Auftragstabelle (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Kontextmenübefehle - Kurvendiagramm

Im Bereich des Kurvenfensters stehen folgende Kontextmenübefehle zur Verfügung:

Zoom 100%

Wählt einen Zoomfaktor, mit dem 100% der Kurvenwerte und / oder Grenzwerte sichtbar sind.

Zoom rückgängig

Nimmt die letzte Zoomänderung zurück.

Skalierung auf Kurven

Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve sichtbar ist.

Skalierung auf Kurven und Grenzen

Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve, die Positionen der aktivierten Software-Endschalter und die minimale und maximale Geschwindigkeitsgrenze sichtbar ist.

Geschwindigkeitsgrenzen einblenden

Blendet die Linien der Geschwindigkeitsgrenzen ein.

Software-Endschalter einblenden

Blendet die Linien der Software-Endschalter ein.

Messlineal einblenden

Blendet das Messlineal ein / aus.

Verwenden Sie das Messlineal, wenn Sie einzelne Werte der Kurven betrachten möchten.

Siehe auch

Konfiguration - Allgemein (Seite 1453)

Konfiguration - Auftragstabelle (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen"

Die nachfolgenden Diagramme zeigen den Bewegungsübergang bei unterschiedlich gewählten Übergangsmodi in der Spalte "Nächster Schritt":

Bewegungsübergang bei vorausgehenden Positionieraufträgen

Folgende Diagramme stellen eine Auftragsabfolge mit zwei Bewegungsaufträgen dar. Der erste Auftrag ist ein Positionierauftrag (grün). Der zweite Auftrag ist ein Geschwindigkeitsauftrag (rot) bzw. Positionierauftrag (blau):

Auftrag abschließen	Bewegung überschleifen
Übergang kleinere Geschwindigkeit zu größerer Geschwindigkeit 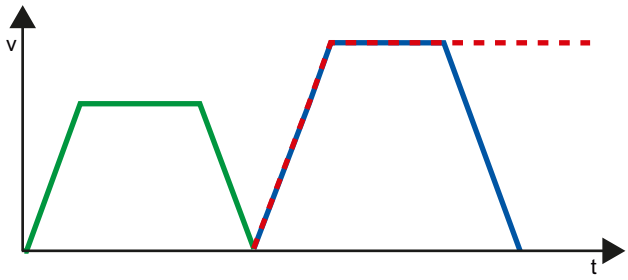 Ein Auftrag mit größerer Geschwindigkeit wird an einen vorherigen Positionierauftrag angehängt. Der erste Positionierauftrag endet an seiner Zielposition mit Geschwindigkeit "0". Der zweite Auftrag startet aus dem Stillstand.	**Übergang kleinere Geschwindigkeit zu größerer Geschwindigkeit** 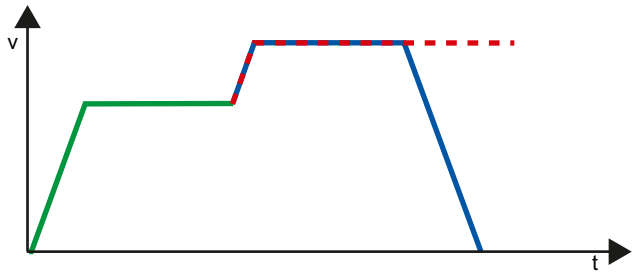 Ein Auftrag mit größerer Geschwindigkeit wird mit einem vorherigen Positionierauftrag überschliffen. Der erste Positionierauftrag endet ohne Stillstand an der Zielposition. Der zweite Auftrag überschleift auf die neue Geschwindigkeit.
Übergang größere Geschwindigkeit zu kleinerer Geschwindigkeit 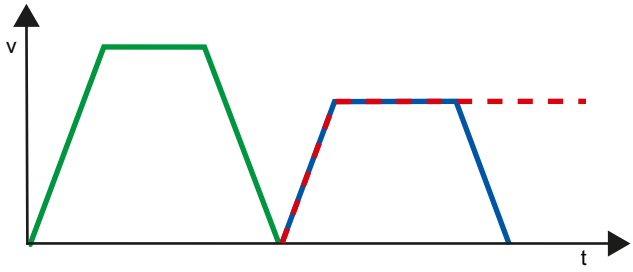 Ein Auftrag mit kleinerer Geschwindigkeit wird an ein vorherigen Positionierauftrag angehängt. Der erste Positionierauftrag endet an seiner Zielposition mit Geschwindigkeit "0". Der zweite Auftrag startet aus dem Stillstand.	**Übergang größere Geschwindigkeit zu kleinerer Geschwindigkeit** 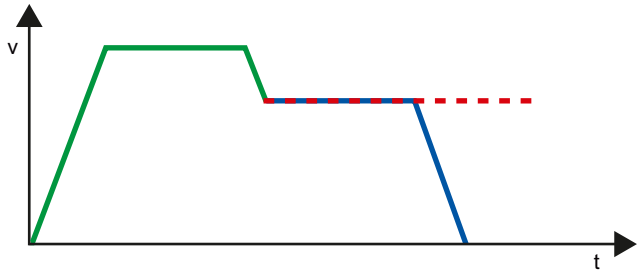 Ein Auftrag mit kleinerer Geschwindigkeit wird mit einem vorherigen Positionierauftrag überschliffen. Der erste Positionierauftrag endet ohne Stillstand an der Zielposition. Der erste Auftrag überschleift auf die neue Geschwindigkeit.

—	1. Auftrag "Positioning Relative" oder "Positioning Absolute"
- -	2. Auftrag "Velocity set point"
—	2. Auftrag "Positioning Relative" oder "Positioning Absolute"

Bewegungsübergang bei vorausgehenden Geschwindigkeitsaufträgen

Folgende Diagramme stellen eine Auftragsabfolge mit zwei Bewegungsaufträgen dar. Der erste Auftrag ist ein Geschwindigkeitsauftrag (violett). Der zweite Auftrag ist ein Geschwindigkeitsauftrag (rot) bzw. Positionierauftrag (blau):

Übergang kleinere Geschwindigkeit zu größerer Geschwindigkeit	Übergang größere Geschwindigkeit zu kleinerer Geschwindigkeit
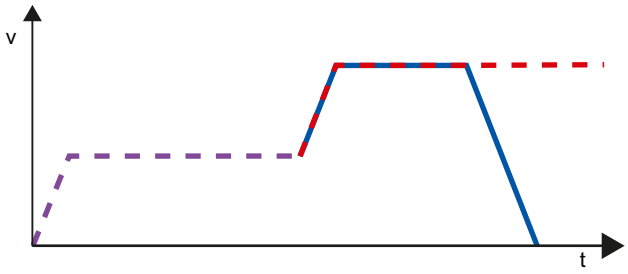	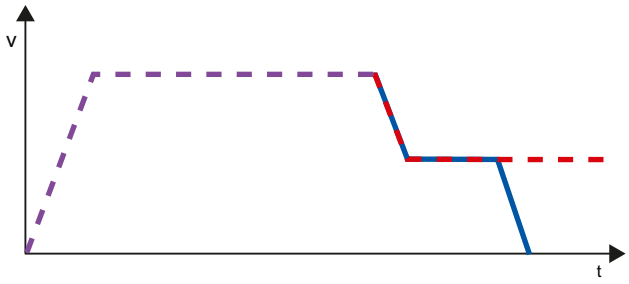
Ein Auftrag mit größerer Geschwindigkeit wird an einen vorausgegangenen Geschwindigkeitsauftrag angehängt. Der erste Geschwindigkeitsauftrag endet nach der angegebenen Laufzeit. Der zweite Auftrag überschleift auf die neue Geschwindigkeit.	Ein Auftrag mit kleinerer Geschwindigkeit wird mit einem vorausgegangenen Geschwindigkeitsauftrag überschliffen. Der zweite Auftrag überschleift auf die neue Geschwindigkeit.

- -	1. Auftrag "Velocity set point"
- -	2. Auftrag "Velocity set point"
—	2. Auftrag "Positioning Relative" oder "Positioning Absolute"

Siehe auch

Konfiguration - Allgemein (Seite 1453)

Konfiguration - Auftragstabelle (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 1465)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

Aufträge und zugehörige Werte

Sie können die Parameter der Auftragstabelle auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Tabellenname>.Command[1..32].Type
  zum Ändern des jeweiligen Auftragstyps
- <Tabellenname>.Command[1..32].Position
  zum Ändern der Position/des Verfahrwegs
- <Tabellenname>.Command[1..32].Velocity
  zum Ändern der Geschwindigkeit
- <Tabellenname>.Command[1..32].Duration
  zum Ändern der Dauer
- <Tabellenname>.Command[1..32].NextStep
  zum Ändern des Parameters "Nächster Schritt"
- <Tabellenname>.Command[1..32].StepCode
  zum Ändern des Schrittcodes

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1678) im Anhang.

Siehe auch

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Konfiguration - Allgemein (Seite 1453)

Konfiguration - Auftragstabelle (Seite 1453)

Kontextmenübefehle - Auftragstabelle (Seite 1457)

Arbeiten mit dem Kurvendiagramm (Seite 1458)

Kontextmenübefehle - Kurvendiagramm (Seite 1462)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 1463)

Erweiterte Parameter

Konfiguration - Erweiterte Parameter

Konfigurieren Sie im Konfigurationsfenster "Erweiterte Parameter" die grundlegenden Eigenschaften der Diagrammdarstellung des Technologieobjekts "Auftragstabelle".

Verwende Achsparameter von

Wählen Sie in der Klappliste, welche Achsparameter als Grundlage für die grafische Darstellung und die Überprüfung der Bewegungssequenz dienen sollen. Wählen Sie in der Klappliste "Beispielachse" aus, wenn Sie im Ordner "Technologieobjekte" noch keine Achse eingefügt haben oder Werte, die in keiner vorhandenen Achse konfiguriert wurden.

Für die Abarbeitung der Auftragstabelle im Anwenderprogramm werden die Achsparameter der am Parameter "Axis" gewählten Achse verwendet.

Maßeinheit Position

Wenn Sie in der Klappliste "Verwende Achsparameter von" eine Beispielachse auswählen, können Sie die Maßeinheit einstellen.

Wenn Sie in der Klappliste "Verwende Achsparameter von" eine konfigurierte Achse auswählen, wird die Maßeinheit der konfigurierten Achse angezeigt.

Achsparameter kopieren

Wählen Sie die Kopierrichtung und die Achse zum Kopieren der Achsparameter. Sie können die Achsparameter der Beispielachse auf die gewählte Achse kopieren oder die Achsparameter der gewählten Achse für die Beispielachse übernehmen. Mit der Schaltfläche "Konfiguration übernehmen" kopieren Sie die Achsparameter entsprechend Ihrer Konfiguration.

Konfiguration - Dynamik

Konfigurieren Sie im Konfigurationsfenster "Dynamik" die Beschleunigung und Verzögerung sowie die Ruckbegrenzung der Beispielachse.

Wenn Sie unter "Konfiguration > Erweiterte Parameter > Erweiterte Parameter" in der Klappliste "Verwende Achsparameter von" eine konfigurierte Achse auswählen, werden die Werte der konfigurierten Achse angezeigt.

Wenn Sie unter "Konfiguration > Erweiterte Parameter > Erweiterte Parameter" in der Klappliste "Verwende Achsparameter von" den Eintrag "Beispielachse" auswählen, können Sie die nachfolgend beschriebenen Felder editieren.

Beschleunigung/Verzögerung

Stellen Sie die gewünschte Beschleunigung der Beispielachse im Feld "Beschleunigung" ein. Die gewünschte Verzögerung kann im Feld "Verzögerung" eingestellt werden.

Die in der Auftragstabelle konfigurierten Verfahraufträge werden mit der gewählten Beschleunigung / Verzögerung berechnet.

Grenzwerte:

- 1.0E-12 ≤ Beschleunigung ≤ 1.0E12
- 1.0E-12 ≤ Verzögerung ≤ 1.0E12

Ruckbegrenzung aktivieren

Mit diesem Optionskästchen aktivieren Sie die Ruckbegrenzung.

Ruck

Stellen Sie den gewünschten Ruck für die Hochlauf- und Rücklauframpe im Feld "Ruck" ein.

Die in der Auftragstabelle konfigurierten Verfahraufträge werden mit dem gewählten Ruck berechnet.

Grenzwerte:

- 1.0E-12 ≤ Ruck ≤ 1.0E12

Konfiguration - Grenzwerte

Konfigurieren Sie im Konfigurationsfenster "Grenzwerte" die Maximale Geschwindigkeit, die Start/Stopp-Geschwindigkeit sowie die SW-Endschalter der Beispielachse.

Wenn Sie unter "Konfiguration > Erweiterte Parameter > Erweiterte Parameter" in der Klappliste "Verwende Achsparameter von" eine konfigurierte Achse auswählen, werden die Werte der konfigurierten Achse angezeigt.

Wenn Sie unter "Konfiguration > Erweiterte Parameter > Erweiterte Parameter" in der Klappliste "Verwende Achsparameter von" den Eintrag "Beispielachse" auswählen, können Sie die nachfolgend beschriebenen Felder editieren.

Maximale Geschwindigkeit / Start/Stopp-Geschwindigkeit

Definieren Sie in diesen Feldern die maximale zugelassene Geschwindigkeit und die Start/Stopp-Geschwindigkeit der Beispielachse. Die Start/Stopp-Geschwindigkeit ist die minimal zulässige Geschwindigkeit der Beispielachse.

Geschwindigkeit	Grenzwert
Start/Stopp-Geschwindigkeit	0.0
	1.0E-12 bis 1.0E12
Maximale Geschwindigkeit	0.0
	1.0E-12 bis 1.0E12

Der Wert der maximalen Geschwindigkeit muss größer gleich dem Wert der Start/Stopp-Geschwindigkeit sein.

SW-Endschalter aktivieren

Mit diesem Optionskästchen aktivieren Sie die Funktion des unteren und oberen Software-Endschalters. Im Kurvendiagramm werden Bewegungsverläufe als Reaktion auf das Erreichen von Software-Endschaltern nicht dargestellt.

Position unterer/oberer SW-Endschalter

Legen Sie in diesen Feldern den Positionswert des unteren und oberen Software-Endschalters fest.

Software-Endschalter	Grenzwert
Unterer Software-Endschalter	-1.0E12 bis -1.0E-12
	0.0
	1.0E-12 bis 1.0E12

Software-Endschalter	Grenzwert
Oberer Software-Endschalter	-1.0E12 bis -1.0E-12
	0.0
	1.0E-12 bis 1.0E12

Der Wert des oberen Software-Endschalters muss größer gleich dem Wert des unteren Software-Endschalters sein.

6.1.1.7 Laden in CPU

Die Daten der Motion Control Technologieobjekte werden in Datenbausteinen gespeichert. Zum Laden eines neuen oder geänderten Technologieobjekts gelten somit die Bedingungen zum Laden von "Bausteinen".

⚠ VORSICHT

Mögliche Fehlfunktionen der Achse beim Laden ohne Hardwarekonfiguration

Bei folgenden Änderungen in der Konfiguration der Achse oder Geber wird die Hardwarekonfiguration geändert:

- Änderung des Impulsgenerators (PTO)
- Änderung der Adresse der HW-Endschalter
- Änderung der Adresse des Referenzpunktschalters
- Änderung der Adresse des PROFIdrive-Telegramms
- Änderung der Adresse des Analogausgangs
- Änderung der Adresse von Freigabe-Ausgang oder Bereit-Eingang

Wird die geänderte Konfiguration der Achse oder Geber mit den Kontextmenübefehlen "Software" oder "Software (alle Bausteine") ohne Laden der Hardwarekonfiguration geladen, so kann dies zu Fehlfunktionen der Achse führen.

Stellen Sie sicher, dass unter den genannten Bedingungen die aktuelle Hardwarekonfiguration in die CPU geladen wird.

Laden im Betriebszustand RUN der CPU S7-1200 (ab Firmwareversion V2.2)

Ab der Firmwareversion V2.2 der CPU S7-1200 wird beim Laden im Betriebszustand RUN der CPU geprüft, ob ein Laden ohne STOP der CPU möglich ist.

Beim Laden von Datenbausteinen im Betriebszustand RUN gelten folgende Bedingungen:

	Laden in Ladespeicher	Laden in Arbeitsspeicher
Datenbaustein geänderte Werte	Ja	Nein
Datenbaustein geänderte Struktur	Ja (ab Firmware V4)	Ja (ab Firmware V4) • Beim Laden mit Reinitialisierung • Bei Variablen in der Reserve beim Laden ohne Reinitialsierung
	Nein (Firmware V2.2...3)	Nein (Firmware V2.2...3)

	Laden in Ladespeicher	Laden in Arbeitsspeicher
Datenbaustein neu	Ja	Ja
Datenbaustein gelöscht	Ja	Ja

Beachten Sie Beim Löschen und Laden mit Reinitialisierung von Datenbausteinen zusätzlich Folgendes:

- Beim Laden eines Technologieobjekts Positionierachse muss die Achse gesperrt sein.
- Beim Laden eines Technologieobjekts Auftragstabelle darf kein MC_CommandTable-Auftrag mit dieser Auftragstabelle aktiv sein (Parameter "Busy" = FALSE).
- Beim Laden eines MC_Power-Instanz-Datenbausteins darf die jeweilige MC_Power-Anweisung nicht aktiv sein (Parameter "Busy" = FALSE).

Ab Technologieversion V3.0 können Motion Control Technologieobjekte (Datenbausteine) auch im Betriebszustand RUN der CPU geladen werden.

Technologieobjekte kleiner V3.0 können im Betriebszustand RUN der CPU nicht geladen werden.

Wählen Sie eine der nachfolgend beschriebenen Aktionen, um die geänderte Konfiguration eines Motion Control Technologieobjekts (ab Version V3.0) in den Arbeitsspeicher zu laden:

- Technologieobjekt Positionierchse und Auftragstabelle
  Wechseln Sie den Betriebszustand der CPU von STOP nach RUN.
- Technologieobjekt Positionierachse
  Sperren Sie die Achse und führen Sie mit der Motion Control-Anweisung "MC_Reset" ein "Restart" aus.
- Technologieobjekt Auftragstabelle
  Stellen Sie sicher, dass die Auftragstabelle nicht verwendet wird. Laden Sie mit der erweiterten Anweisung "READ_DBL" den Datenbaustein der Auftragstabelle in den Arbeitsspeicher.

Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 1371)

6.1.1.8 Inbetriebnahme

Achssteuertafel

Verwenden Sie die Achsteuertafel, um die Achse im Handbetrieb zu verfahren, die Achseinstellungen zu optimieren und um Ihre Anlage zu testen.

Die Bedienung der Achssteuertafel ist nur möglich, wenn eine Onlineverbindung zur CPU aufgebaut ist. Es ist empfehlenswert, jede andere Online-Kommunikation während der Nutzung der Achssteuertafel und der Optimierung zu deaktivieren, um die Reaktionszeiten so kurz wie möglich zu halten.

Die Achssteuertafel ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Auftrag
- Aktuelle Werte
- Achsstatus

Hinweis

Reaktionszeiten der Achssteuertafel

Die Reaktionszeit während der Bedienung der Achssteuertafel ist von der Kommunikationslast der CPU abhängig. Schließen Sie alle weiteren Onlinefenster des TIA-Portals, um die Reaktionszeit niedrig zu halten.

Im Startdialog können Sie zusätzlich den Timeout anpassen.

Steuerungshoheit

In diesem Bereich übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben sie an Ihr Anwenderprogramm zurück:

- **Schaltfläche "Holen"**
  Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. Beachten Sie beim Holen der Steuerungshoheit Folgendes:
  - Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein.
  - Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler abgewiesen.

  ⚠ VORSICHT

  Weitere Achsen im Automatikbetrieb

  Die Steuerungshoheit wird nur für das ausgewählte Technologieobjekt übernommen. Wenn sich weitere Achsen im Automatikbetrieb befinden, kann dies zu gefährlichen Situationen führen.

  Sperren Sie in diesem Fall alle weiteren Achsen.

- **Schaltfläche "Abgeben"**
  Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.

Achse

In diesem Bereich geben Sie das Technologieobjekt für den Betrieb mit der Achssteuertafel/ Optimierung frei oder sperren es:

- **Schaltfläche "Freigeben"**
  Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
- **Schaltfläche "Sperren"**
  Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.

Auftrag

Die Bedienung im Bereich "Auftrag" ist nur mit freigegebener Achse möglich. Wählen Sie zwischen folgenden Befehlen:

- **Tippen**
  Der Befehl entspricht dem Motion Control-Auftrag "MC_MoveJog" im Anwenderprogramm.
- **Positionieren**
  Der Befehl entspricht den Motion Control-Aufträgen "MC_MoveAbsolute" und "MC_MoveRelative" im Anwenderprogramm. Zum absoluten Positionieren muss die Achse referenziert sein.
- **Referenzieren**
  Der Befehl entspricht dem Motion Control-Auftrag "MC_Home" im Anwenderprogramm.
  - Die Schaltfläche "Referenzpunkt setzen" entspricht Mode = 0 (Direktes Referenzieren Absolut)
  - Die Schaltfläche "Referenzieren" entspricht Mode = 3 (Aktives Referenzieren)

  Für das aktive Referenzieren muss in der Achskonfiguration der Referenzpunktschalter konfiguriert sein.
  Die Werte für Anfahr-, Referenziergeschwindigkeit und für die Referenzpunktverschiebung werden unverändert aus der Achskonfiguration übernommen.

Je nach Auswahl werden die relevanten Felder zur Eingabe der Sollwerte und Schaltflächen zum Starten des Befehls eingeblendet.

Aktivieren Sie das Optionskästchen "Ruckbegrenzung aktivieren", um die Ruckbegrenzung zu aktivieren. Defaultmäßig wird der Ruck mit 10 % des konfigurierten Werts übernommen. Dieser Wert kann je nach Anforderung geändert werden.

Aktuelle Werte

In diesem Bereich werden die folgenden Istwerte der Achse angezeigt:

- Position
- Geschwindigkeit

Achsstatus

Im Bereich "Achsstatus" wird der aktuelle Achs- und Antriebszustand angezeigt.

Statusmeldung	Beschreibung
Freigegeben	Die Achse ist freigegeben und bereit über Motion Control-Aufträge gesteuert zu werden.
Referenziert	Die Achse ist referenziert und kann absolute Positionieraufträge der Motion Control-Anweisung "MC_MoveAbsolute" ausführen.
Bereit	Der Antrieb ist betriebsbereit.
Achsfehler	Am Technologieobjekt Positionierachse ist ein Fehler aufgetreten. Die detaillierte Fehlerursache ist im Feld "Fehlermeldung" angezeigt.
Geberwerte gültig	Die Geberwerte sind gültig.
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben.
Antriebsfehler	Der Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt.
Restart erforderlich	Eine geänderte Konfiguration der Achse wurde im Betriebszustand RUN der CPU in den Ladespeicher geladen. Um die geänderte Konfiguration in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung "MC_Reset".

Das Feld "Info-Meldung" zeigt erweiterte Informationen zum Status der Achse an.

Das Feld "Fehlermeldung" zeigt den aktuellen Fehler an.

Quittieren Sie anstehende Fehler nach ihrer Beseitigung durch Klicken der Schaltfläche "Quittieren".

Hinweis

Initialwerte für Geschwindigkeit, Beschleunigung/Verzögerung und Ruck

Aus Sicherheitsgründen werden beim Aktivieren der Achssteuertafel die Parameter "Geschwindigkeit" und "Beschleunigung/Verzögerung" und "Ruck" mit lediglich 10% der in der Konfiguration eingestellten Werte initialisiert. Der Parameter "Ruck" wird erst ab Technologieobjekt "Achse" V2.0 eingesetzt.

Zur Initialisierung werden die Werte in der Konfigurationssicht "Erweiterte Parameter > Dynamik > Allgemein" zu Grunde gelegt.

Der Parameter "Geschwindigkeit" der Achssteuertafel wird aus der "Maximalen Geschwindigkeit", der Parameter "Beschleunigung/Verzögerung" aus der "Beschleunigung" der Konfiguration abgeleitet.

Die Parameter "Geschwindigkeit", "Beschleunigung/Verzögerung" und "Ruck" können in der Achssteuertafel geändert werden. Die Werte in der Konfiguration bleiben davon unberührt.

Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 1371)

Optimierung

Die Bewegung von Achsen mit Antriebsanbindung über PROFIdrive/Analogausgang wird lagegeregelt.

Die Funktion "Optimierung" unterstützt Sie bei der Ermittlung der optimalen Verstärkung (Kv-Faktor) für den Regelkreis (Seite 1426) der Achse. Hierzu wird der Geschwindigkeitsverlauf der Achse während einer vorgebbaren Positionierbewegung mit der Trace-Funktion aufgezeichnet. Anschließend können Sie die Aufzeichnung auswerten und die Verstärkung entsprechend anpassen. Es ist empfehlenswert, jede andere Online-Kommunikation während der Nutzung der Achssteuertafel und der Optimierung zu deaktivieren, um die Reaktionszeiten so kurz wie möglich zu halten.

In der Projektnavigation finden Sie die Funktion "Optimierung" für das Technologieobjekt Positionierachse unter "Technologieobjekt > Inbetriebnahme".

Der Dialog "Optimierung" ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Achsstatus
- Verstärkung optimieren
- Messung durchführen
- Trace

Hinweis

Keine Übernahme der Parameter

Die eingestellten Parameterwerte werden nach der Rückgabe der Steuerungshoheit verworfen. Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration.

Start der Optimierung

Beim Start der Optimierung wird gleichzeitig der Trace gestartet. Hierzu kann im Startdialog ein Timeout angepasst werden.

Steuerungshoheit

In diesem Bereich übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben sie an Ihr Anwenderprogramm zurück:

- **Schaltfläche "Holen"**
  Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. Beachten Sie beim Holen der Steuerungshoheit Folgendes:
  - Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein.
  - Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler abgewiesen.

  ⚠ VORSICHT

  Weitere Achsen im Automatikbetrieb

  Die Steuerungshoheit wird nur für das ausgewählte Technologieobjekt übernommen. Wenn sich weitere Achsen im Automatikbetrieb befinden, kann dies zu gefährlichen Situationen führen.

  Sperren Sie in diesem Fall alle weiteren Achsen.

- **Schaltfläche "Abgeben"**
  Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.

Achse

In diesem Bereich geben Sie das Technologieobjekt für den Betrieb mit der Achssteuertafel/ Optimierung frei oder sperren es:

- **Schaltfläche "Freigeben"**
  Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
- **Schaltfläche "Sperren"**
  Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.

Achsstatus

Im Bereich "Achsstatus" wird der aktuelle Achs- und Antriebszustand angezeigt.

Statusmeldung	Beschreibung
Freigegeben	Die Achse ist freigegeben und bereit über Motion Control-Aufträge gesteuert zu werden.
Referenziert	Die Achse ist referenziert und kann absolute Positionieraufträge der Motion Control-Anweisung "MC_MoveAbsolute" ausführen.
Bereit	Der Antrieb ist betriebsbereit.
Achsfehler	Am Technologieobjekt Positionierachse ist ein Fehler aufgetreten. Die detaillierte Fehlerursache ist im Feld "Fehlermeldung" angezeigt.
Geberwerte gültig	Die Geberwerte sind gültig.
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben.

Statusmeldung	Beschreibung
Antriebsfehler	Der Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt.
Restart erforderlich	Eine geänderte Konfiguration der Achse wurde im Betriebszustand RUN der CPU in den Ladespeicher geladen. Um die geänderte Konfiguration in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung MC_Reset.

Das Feld "Info-Meldung" zeigt erweiterte Informationen zum Status der Achse an.

Das Feld "Fehlermeldung" zeigt den aktuellen Fehler an.

Quittieren Sie anstehende Fehler nach ihrer Beseitigung durch Klicken der Schaltfläche "Quittieren".

Verstärkung optimieren

In diesem Bereich nehmen Sie die Einstellungen zum Optimieren der Verstärkung vor:

- Vorsteuerung
  Konfigurieren Sie in diesem Feld die aktuelle prozentuale Geschwindigkeitsvorsteuerung des Lagereglers.
- Weg
  Konfigurieren Sie in diesem Feld den Weg für einen Testschritt.
- Optionskästchen "Dynamik anpassen"
  Aktivieren Sie die Option, um die Beschleunigung und die maximale Beschleunigung für die Optimierung anzupassen.
- Geschwindigkeit
  Konfigurieren Sie in diesem Feld die maximale Geschwindigkeit für einen Testschritt.
- Beschleunigung
  Konfigurieren Sie in diesem Feld die Beschleunigung für einen Testschritt.
- Messdauer
  Die Messdauer wird jewels in Abhängigkeit von der gewählten Beschleunigung, Geschwindigkeit und Strecke neu berechnet und eingetragen.
  Sie können den Wert der Messdauer nachträglich anpassen.
- Verstärkung
  Konfigurieren Sie in diesem Feld die aktuelle Verstärkung des Lagereglers (Kv).
  Die Verstärkung wird bei der Eingabe wirksam. Eine zu große Verstärkung des Lagereglers (Kv) kann zu einem Fehler am Antrieb führen.

Messung durchführen

In diesem Bereich führen Sie die Testschritte aus:

- **Schaltfläche "Vorwärts"**
  Mit der Schaltfläche "Vorwärts" starten Sie einen Testschritt zur Optimierung in positive Richtung.
- **Schaltfläche "Rückwärts"**
  Mit der Schaltfläche "Rückwärts" starten Sie einen Testschritt zur Optimierung in negative Richtung.
- **Schaltfläche "Stopp"**
  Mit der Schaltfläche "Stopp" beenden sie die laufende Bewegung der Optimierung und beenden die Trace-Aufnahme.

Trace

Mit jedem Testschritt wird automatisch eine Trace-Aufzeichnung der benötigten Parameter gestartet und nach dem Beenden des Testschritts angezeigt. Nach der Rückgabe der Steuerungshoheit wird die Trace-Aufzeichnung gelöscht.

Eine vollständige Beschreibung der Trace-Funktion finden Sie in der Hilfe des TIA Portals im Kapitel Trace- und Logikanalysatorfunktion nutzen.

6.1.1.9 Programmieren

Übersicht über die Motion Control-Anweisungen

Über die Motion Control-Anweisungen steuern Sie die Achse vom Anwenderprogramm aus. Die Anweisungen starten Motion Control-Aufträge, die die gewünschten Funktionen ausführen.

Den Status der Motion Control-Aufträge, sowie eventuell bei deren Abarbeitung aufgetretene Fehler, können Sie den Ausgangsparametern der Motion Control-Anweisungen entnehmen. Folgende Motion Control-Anweisungen stehen zur Auswahl:

- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT
- AUTOHOTSPOT

Siehe auch

Anwenderprogramm erstellen (Seite 1478)

Programmierhinweise (Seite 1481)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 1483)

Verfolgung laufender Aufträge (Seite 1483)

Fehleranzeigen der Motion Control-Anweisungen (Seite 1495)

Anwenderprogramm erstellen

Nachfolgend erstellen Sie im Anwenderprogramm das Grundgerüst zur Steuerung Ihrer Achse. Mit Hilfe der einzufügenden Motion Control-Anweisungen werden alle zur Verfügung stehenden Funktionen der Achse gesteuert.

Voraussetzung

- Das Technologieobjekt wurde erzeugt und fehlerfrei konfiguriert.

Vor der Erstellung und Test des Anwenderprogramms ist es empfehlenswert, die Funktion der Achse und der entsprechenden Anlagenteile mit der Achssteuertafel zu testen.

Vorgehen

Gehen Sie zum Erstellen des Anwenderprogramms entsprechend dem nachfolgend beschriebenen Prinzip vor:

1. Doppelklicken Sie in der Projektnavigation Ihren Codebaustein (der Codebaustein muss im zyklischen Programm aufgerufen werden).
   Der Codebaustein wird im Programmiereditor geöffnet und die zur Verfügung stehenden Anweisungen werden eingeblendet.
2. Öffnen Sie die Kategorie "Technologie" und den Ordner "Motion Control".
3. Ziehen Sie per Drag & Drop die Anweisung "MC_Power" in das gewünschte Netzwerk des Codebausteins.
   Der Dialog zur Definition des Instanz-DBs wird geöffnet.
4. Wählen Sie im Dialog zwischen folgenden Alternativen:
   Einzelinstanz
   Klicken Sie die Schaltfläche "Einzelinstanz" und wählen Sie, ob Sie den Namen und die Nummer des Instanz-DBs automatisch oder manuell bestimmen möchten.
   Multiinstanz
   Klicken Sie die Schaltfläche "Multiinstanz" und wählen Sie, ob Sie den Namen der Multiinstanz automatisch oder manuell bestimmen möchten.

5. Klicken Sie die Schaltfläche "OK".
   Die Motion Control-Anweisung "MC_Power" wird in das Netzwerk eingefügt.

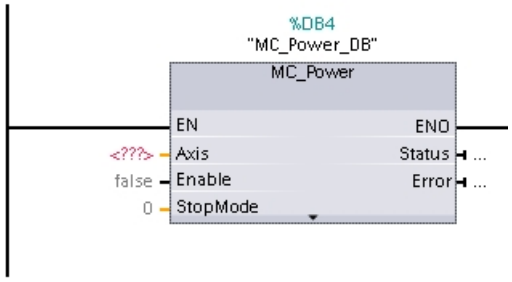

Mit "<???>" markierte Parameter müssen versorgt werden; alle anderen Parameter sind mit Voreinstellungswerten belegt.
Schwarz dargestellte Parameter sind zum Einsatz der Motion Control-Anweisung notwendig.

6. Markieren Sie in der Projektnavigation das Technologieobjekt und ziehen es per Drag & Drop auf <???>.

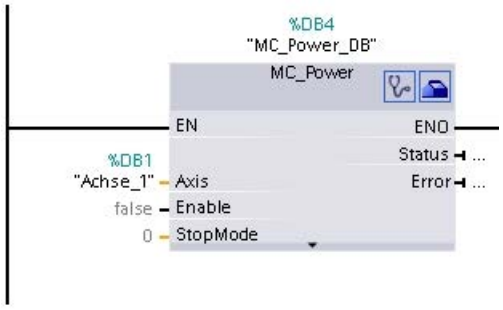

Nach der Auswahl des Datenbausteins des Technologieobjekts stehen Ihnen folgende Schaltflächen zur Verfügung:

Klicken Sie auf das Stethoskopsymbol, falls Sie die Diagnose des Technologieobjekts öffnen möchten.

Klicken Sie das Werkzeugkastensymbol, falls Sie die Konfigurationssicht des Technologieobjekts öffnen möchten.

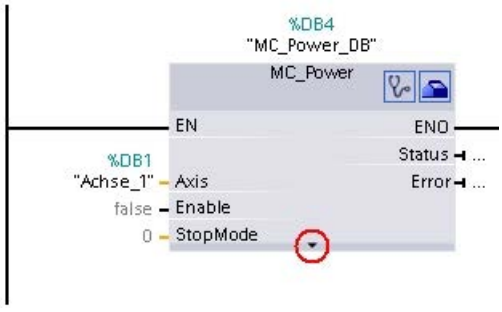

Mit Klicken des nach unten gerichteten Pfeils öffnen Sie die Sicht auf weitere Parameter der Motion Control-Anweisung.

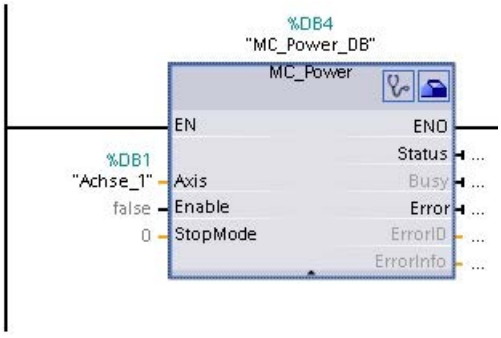

Die nun sichtbaren, grau dargestellten Parameter können optional verwendet werden.

7. Fügen Sie entsprechend Schritt 3 bis 6, weitere von Ihnen gewünschte Motion Control-Anweisungen ein.

Ergebnis

Das Grundgerüst zur Steuerung der Achse ist im Anwenderprogramm angelegt.

Versorgen Sie in weiteren Teilen des Anwenderprogramms die Eingangsparameter der Motion Control-Anweisungen, um die gewünschten Aufträge am Technologieobjekt "Achse" anzustoßen.

Werten Sie die Ausgangsparameter der Motion Control-Anweisungen und die Variablen des Datenbausteins der Achse aus, um die angestoßenen Aufträge und den Status der Achse zu verfolgen.

Details zu den Parametern der Motion Control-Anweisungen entnehmen Sie bitte deren Detailbeschreibung.

Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 1477)

Programmierhinweise (Seite 1481)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 1483)

Verfolgung laufender Aufträge (Seite 1483)

Fehleranzeigen der Motion Control-Anweisungen (Seite 1495)

Programmierhinweise

Berücksichtigen Sie bei der Erstellung Ihres Anwenderprogramms die nachfolgenden Hinweise:

- **Zyklischer Aufruf der verwendeten Motion Control-Anweisungen**
  Der aktuelle Status der Abarbeitung der Aufträge wird über die Ausgangsparameter der Motion Control-Anweisung zur Verfügung gestellt. Der Status wird mit jedem Aufruf der Motion Control-Anweisung aktualisiert. Sorgen deshalb Sie dafür, dass die verwendeten Motion Control-Anweisungen zyklisch aufgerufen werden.
- **Übernahme der Parameterwerte einer Motion Control-Anweisung**
  Die an den Eingangsparametern anstehenden Parameterwerte werden mit einer positiven Flanke am Eingangsparameter "Execute" beim Aufruf des Bausteins übernommen.
  Der Motion Control-Auftrag wird mit diesen Parameterwerten gestartet. Nachträglich an der Motion Control-Anweisung geänderte Parameterwerte werden erst beim nächsten Start des Motion Control-Auftrags übernommen.
  Eine Ausnahme bildet der Eingangsparameter "StopMode" der Motion Control-Anweisung "MC_Power" und "Velocity" der Motion Control-Anweisung "MC_MoveJog". Eine Änderung des Eingangsparameter wird auch bei "Enable" = TRUE, bzw. "JogForward" und "JogBackward" übernommen. .

- **Programmierung unter Berücksichtigung der Statusinformationen**
  Achten Sie bei einer schrittweisen Abarbeitung von Motion Control-Aufträgen vor dem Start eines neuen Auftrags auf den Abschluss des laufenden Auftrags. Prüfen Sie den Abschluss des laufenden Auftrags anhand der Statusmeldungen der Motion Control-Anweisung und der Variablen "StatusBits" des Technologieobjekts.
  In den nachfolgenden Beispielen muss die beschriebene Abfolge berücksichtigt werden. Wird sie nicht berücksichtigt, so wird ein Fehler der Achse oder des Auftrags angezeigt.
  - **Freigabe der Achse mit der Motion Control-Anweisung "MC_Power"**
    Bevor die Achse Aufträge zum Verfahren annehmen kann, muss die Achse freigegeben sein. Prüfen Sie die Freigabe der Achse über eine UND-Verknüpfung der Variable <Achsname>.StatusBits.Enable = TRUE mit dem Ausgangsparameter Status = TRUE der Motion Control-Anweisung "MC_Power".
  - **Fehler quittieren mit der Motion Control-Anweisung "MC_Reset"**
    Vor dem Start eines Motion Control-Auftrags müssen mit "MC_Reset" zu quittierende Fehler quittiert werden. Beheben Sie die Fehlerursache und quittieren Sie den Fehler mit der Motion Control-Anweisung "MC_Reset". Prüfen Sie vor dem Anstoß eines neuen Auftrags die erfolgreiche Quittierung des Fehlers. Verwenden Sie hierzu eine UND-Verknüpfung der Variable <Achsname>.StatusBits.Error = FALSE mit dem Ausgangsparameter Done = TRUE der Motion Control-Anweisung "MC_Reset".
  - **Referenzieren der Achse mit der Motion Control-Anweisung "MC_Home"**
    Bevor ein MC_MoveAbsolute-Auftrag gestartet werden kann, muss die Achse referenziert sein. Prüfen Sie nach dem Referenzieren der Achse den erfolgreichen Abschluss über eine UND-Verknüpfung der Variable <Achsname>.StatusBits.HomingDone = TRUE mit dem Ausgangsparameter Done = TRUE der Motion Control-Anweisung "MC_Home".
- **Ablösende Bearbeitung von Motion Control-Aufträgen**
  Motion Control-Aufträge zum Verfahren der Achse können auch ablösend ausgeführt werden.
  Wird während eines laufenden Motion Control-Auftrags ein neuer Motion Control-Auftrag der Achse gestartet, so wird der laufende Auftrag durch den neuen Auftrag abgelöst, ohne dass der laufende Auftrag komplett abgearbeitet wird. Der abgelöste Auftrag meldet dies durch CommandAborted = TRUE an der Motion Control-Anweisung. So kann z. B. ein laufender MC_MoveRelative-Auftrag durch einen MC_MoveAbsolute-Auftrag abgelöst werden.
- **Mehrfachverwendung der gleichen Instanz vermeiden**
  Alle relevanten Informationen eines Motion Control-Auftrags werden in seiner Instanz gespeichert.
  Starten Sie unter Verwendung dieser Instanz keinen neuen Auftrag, solange Sie den Status des aktuellen Auftrags verfolgen möchten. Verwenden Sie verschiedene Instanzen, wenn Sie die Aufträge getrennt verfolgen möchten. Wird die gleiche Instanz für mehrere Motion Control-Aufträge verwendet, so werden die Status- und Fehlerinformationen der einzelnen Aufträge gegenseitig überschrieben.
  Im Anwenderprogramm muss jede Achse mit einem eigenen Aufruf der Motion Control-Anweisung "MC_Power" mit eigenem Instanz- Datenbaustein aufgerufen werden.
- **Aufruf von Motion Control-Anweisungen in verschiedenen Prioritätsklassen (Ablaufebenen)**
  Motion Control-Anweisungen mit gleicher Instanz dürfen ohne Verriegelung nicht in unterschiedlichen Prioritätsklassen aufgerufen werden. Wie Sie Motion Control-Anweisungen zur Nachverfolgung verriegelt aufrufen können, finden Sie unter "Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 1508)".

Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 1477)

Anwenderprogramm erstellen (Seite 1478)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 1483)

Verfolgung laufender Aufträge (Seite 1483)

Fehleranzeigen der Motion Control-Anweisungen (Seite 1495)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 1508)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart

Mit einem NETZ-AUS oder CPU-STOP werden alle aktiven Motion Control-Aufträge abgebrochen. Alle CPU-Ausgänge inklusive der Impuls- und Richtungsausgänge werden zurückgesetzt.

Nach einem darauffolgenden NETZ-EIN oder CPU-Neustart (CPU-RUN) werden die Technologieobjekte und die Motion Control-Aufträge neu initialisiert.

Alle Aktualdaten der Technolgieobjekte sowie alle Status- und Fehlerinformationen der zuvor aktiven Motion Control-Aufträge werden auf Initialwerte zurück gesetzt.

Bevor die Achse wieder verwendet werden kann, muss sie erneut mit Hilfe der Motion Control-Anweisung "MC_Power" freigegeben werden. Ist eine Referenzierung gewünscht, so muss die Achse mit der Motion Control-Anweisung "MC_Home" ebenfalls erneut referenziert werden. Bei der Verwendung eines Absolutwertgebers bleibt die Referenzierung auch nach NETZ-AUS erhalten.

Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 1477)

Anwenderprogramm erstellen (Seite 1478)

Programmierhinweise (Seite 1481)

Verfolgung laufender Aufträge (Seite 1483)

Fehleranzeigen der Motion Control-Anweisungen (Seite 1495)

Verfolgung laufender Aufträge

Verfolgung laufender Aufträge

Bei der Verfolgung laufender Motion Control-Aufträge kann zwischen drei typischen Gruppen unterschieden werden:

- **Motion Control-Anweisungen mit dem Ausgangsparameter "Done"**
- **Die Motion Control-Anweisung "MC_MoveVelocity"**
- **Die Motion Control-Anweisung "MC_MoveJog"**

Motion Control-Anweisungen mit Ausgangsparameter Done

Motion Control-Anweisungen mit dem Ausgangsparameter "Done" werden über den Eingangsparameter "Execute" gestartet und haben einen definierten Abschluss (z. B. bei der Motion Control-Anweisung "MC_Home": Referenzieren war erfolgreich). Damit ist der Auftrag beendet und die Achse steht still.

Die Aufträge der nachfolgenden Motion Control-Anweisungen haben einen definierten Abschluss:

- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_CommandTable (ab Technologieobjekt V2)
- MC_ChangeDynamic (ab Technologieobjekt V2)
- MC_WriteParam (ab Technologieobjekt V4)
- MC_ReadParam (ab Technologieobjekt V4)

Der Ausgangsparameter "Done" zeigt den Wert TRUE, wenn der Auftrag erfolgreich abgeschlossen wurde.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht. Die Motion Control-Anweisungen "MC_Reset" kann nicht abgebrochen werden und besitzt deshalb keinen Ausgangsparameter "CommandAborted". Da die Motion Control-Anweisung "MC_ChangeDynamic" sofort abgeschlossen ist, besitzt die Anweisung keine Ausgangsparameter "Busy" und "CommandAborted".

Während der Bearbeitung des Motion Control-Auftrags zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wurde der Auftrag abgeschlossen, abgebrochen oder durch einen Fehler gestoppt, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE. Dies geschieht unabhängig vom Signal am Eingangsparameters "Execute".

Die Ausgangsparameter "Done", "CommandAborted" und "Error" zeigen den Wert TRUE für mindestens einen Zyklus an. Während der Eingangsparameter "Execute" auf TRUE gesetzt ist, werden diese Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft für unterschiedliche Situationen gezeigt:

Vollständige Abarbeitung des Auftrags

Wird der Motion Control-Auftrag bis zum Abschluss vollständig abgearbeitet, so wird dies am Ausgangsparameter "Done" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Done":

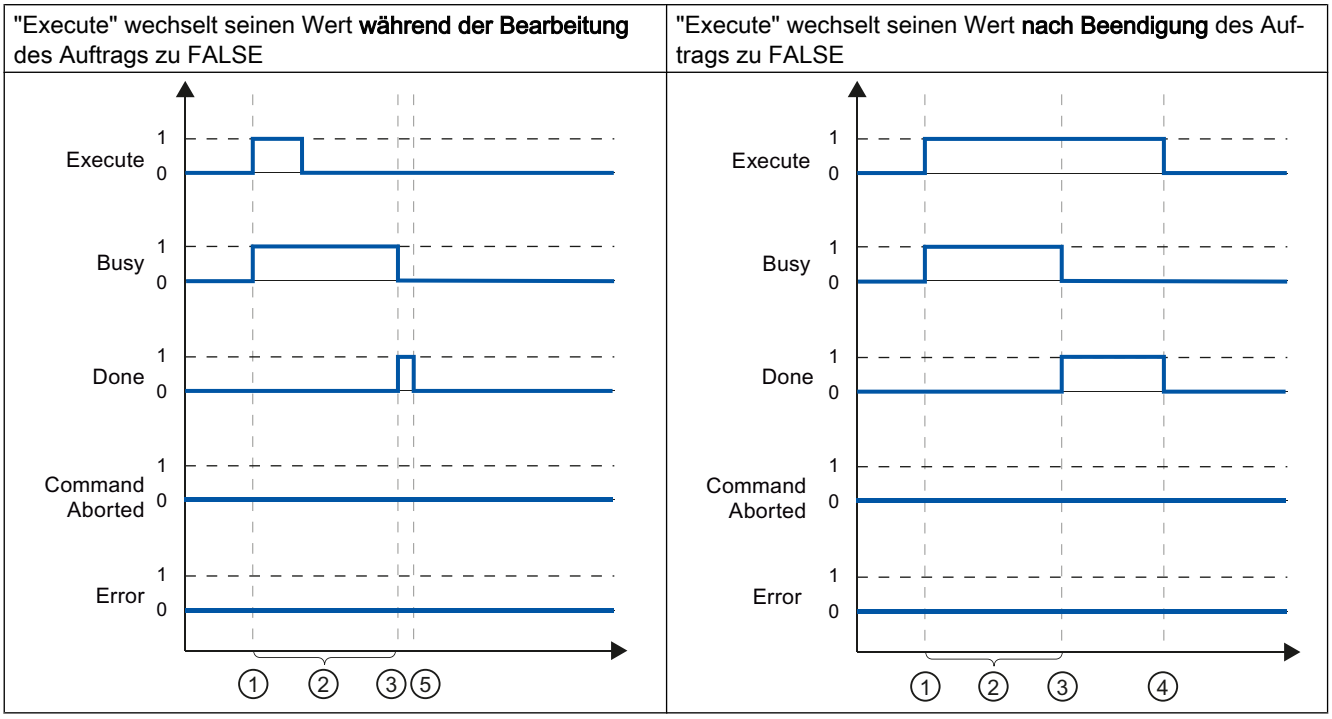

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten.
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Mit dem Abschluss des Auftrags (z. B. bei der Motion Control-Anweisung "MC_Home": Referenzieren war erfolgreich) wechselt der Ausgangsparameter "Busy" nach FALSE und "Done" nach TRUE.
④	Behält "Execute" den Wert TRUE bis nach Beendigung des Auftrags bei, so bleibt auch "Done" TRUE und wechselt seinen Wert gemeinsam mit "Execute" zu FALSE.
⑤	Wurde "Execute" schon vor Beendigung des Auftrags auf FALSE gesetzt, so zeigt "Done" nur für einen Abarbeitungszyklus den Wert TRUE.

Abbruch des Auftrags

Wird der Motion Control-Auftrag während der Bearbeitung abgebrochen, so wird dies am Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "CommandAborted":

"Execute" wechselt seinen Wert **vor** Abbruch des Auftrags zu FALSE	"Execute" wechselt seinen Wert **nach** Abbruch des Auftrags zu FALSE
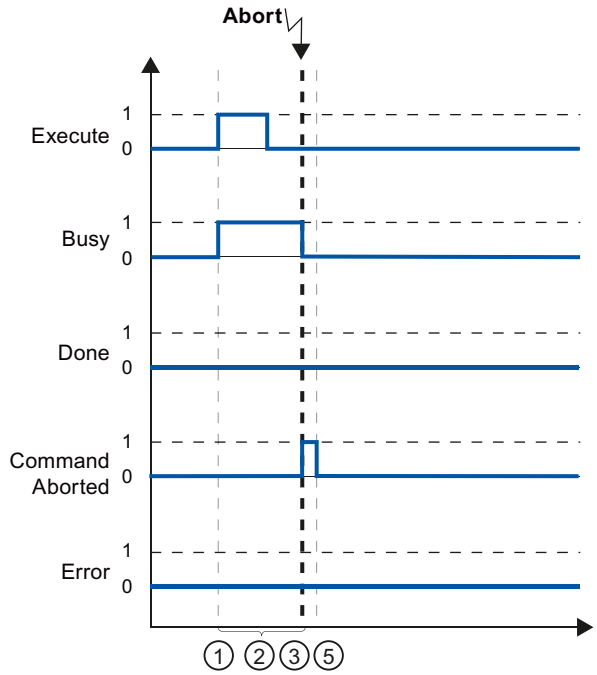	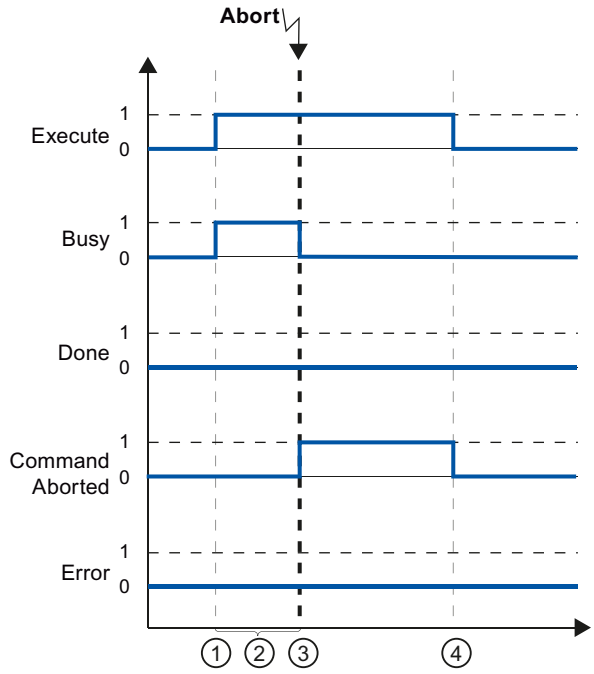

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten.
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE.
④	Behält "Execute" den Wert TRUE bis nach Abbruch des Auftrags bei, so bleibt auch "CommandAborted" TRUE und wechselt seinen Wert gemeinsam mit "Execute" zu FALSE.
⑤	Wurde "Execute" schon vor Abbruch des Auftrags auf FALSE gesetzt, so zeigt "CommandAborted" nur für einen Abarbeitungszyklus den Wert TRUE.

Fehler während der Auftragsbearbeitung

Tritt während der Bearbeitung des Motion Control-Auftrags ein Fehler auf, so wird dies am Ausgangsparameter "Error" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Error":

"Execute" wechselt seinen Wert **vor** Auftreten des Fehlers zu FALSE	"Execute" wechselt seinen Wert **nach** Auftreten des Fehlers zu FALSE
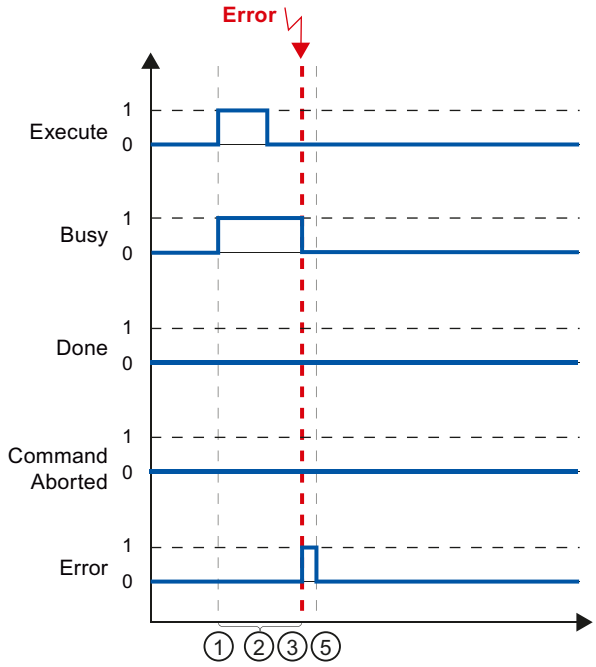	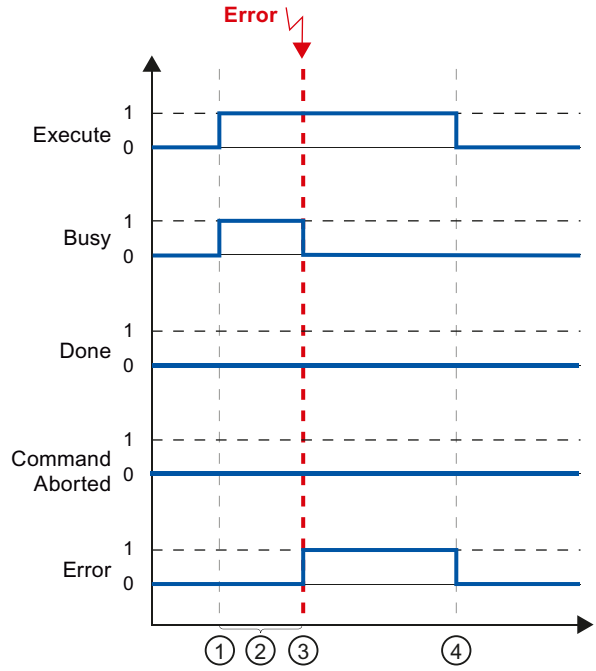

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE.
④	Behält "Execute" den Wert TRUE bis nach Auftreten des Fehlers bei, so bleibt "Error" TRUE und wechselt seinen Wert erst gemeinsam mit "Execute" zu FALSE.
⑤	Wurde "Execute" schon vor Auftreten des Fehlers auf FALSE gesetzt, so zeigt "Error" nur für einen Abarbeitungszyklus den Wert TRUE.

Motion Control-Anweisung MC_MoveVelocity

Ein "MC_MoveVelocity"-Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Das Auftragsziel ist erfüllt, wenn die parametrierte Geschwindigkeit erreicht wurde und die Achse mit konstanter Geschwindigkeit verfährt. Das Erreichen und Halten der parametrierten Geschwindigkeit wird am Parameter "InVelocity" mit dem Wert TRUE angezeigt.

Die Bewegung der Achse kann z. B. mit einem "MC_Halt"-Auftrag gestoppt werden.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht.

Während der Bearbeitung des Motion Control-Auftrags zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wenn der Auftrag durch einen anderen Auftrag oder durch einen Fehler abgebrochen wird, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE. Dies geschieht unabhängig vom Signal am Eingangsparameter "Execute".

Die Ausgangsparameter "CommandAborted" und "Error" zeigen den Wert TRUE für mindestens einen Zyklus an. Während der Eingangsparameter "Execute" auf TRUE gesetzt ist, werden diese Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft in unterschiedlichen Situationen gezeigt:

Die parametrierte Geschwindigkeit wird erreicht

Wird der Motion Control-Auftrag bis zum Erreichen der parametrierten Geschwindigkeit ausgeführt, so wird dies am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt. Der Parameter "Execute" hat keinen Einfluss auf die Anzeigedauer am Parameter "InVelocity".

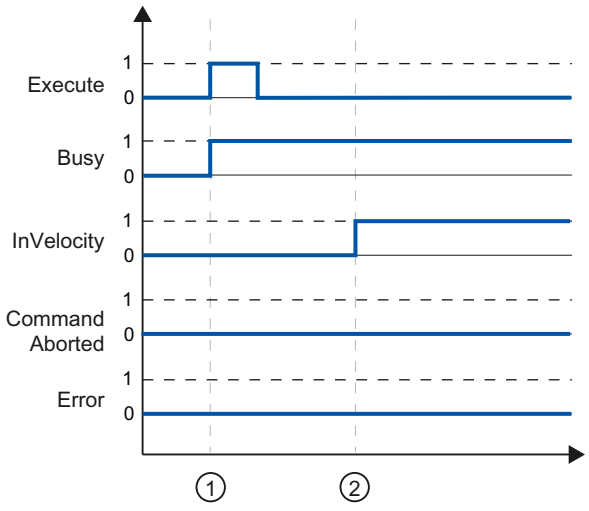

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Je nach Programmierung kann "Execute" noch vor oder erst nach Erreichen der parametrierten Geschwindigkeit auf den Wert FALSE zurückgesetzt werden. Während der Auftrag bearbeitet wird, zeigt Parameter "Busy" den Wert TRUE.
②	Mit dem Erreichen der parametrierten Geschwindigkeit wechselt Parameter "InVelocity" nach TRUE. Die Parameter "Busy" und "InVelocity" bleiben so lange auf dem Wert TRUE, bis der "MC_MoveVelocity"-Auftrag durch einen anderen Motion Control-Auftrag abgelöst oder mit Fehler abgebrochen wird.

Der Auftrag wird vor dem Erreichen der parametrierten Geschwindigkeit abgebrochen

Wird der Motion Control-Auftrag vor dem Erreichen der parametrierten Geschwindigkeit abgebrochen, so wird dies durch den Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameters "CommandAborted".

"Execute" wechselt seinen Wert **vor** Abbruch des Auftrags zu FALSE	"Execute" wechselt seinen Wert **nach** Abbruch des Auftrags zu FALSE
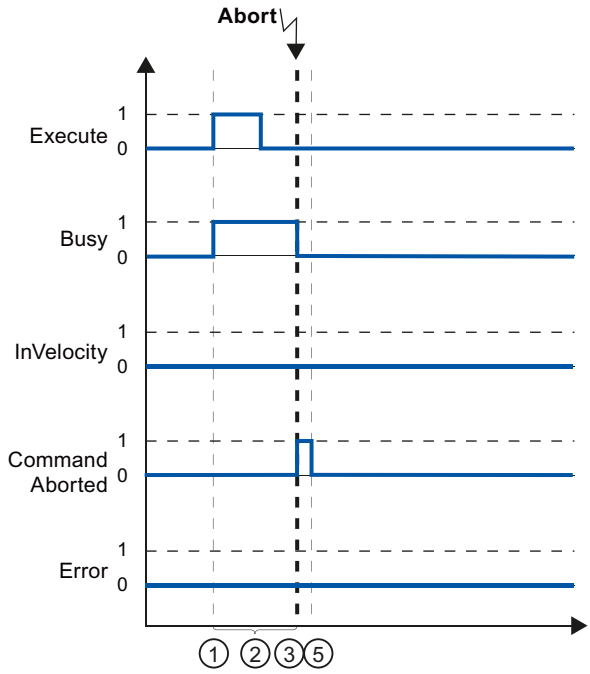	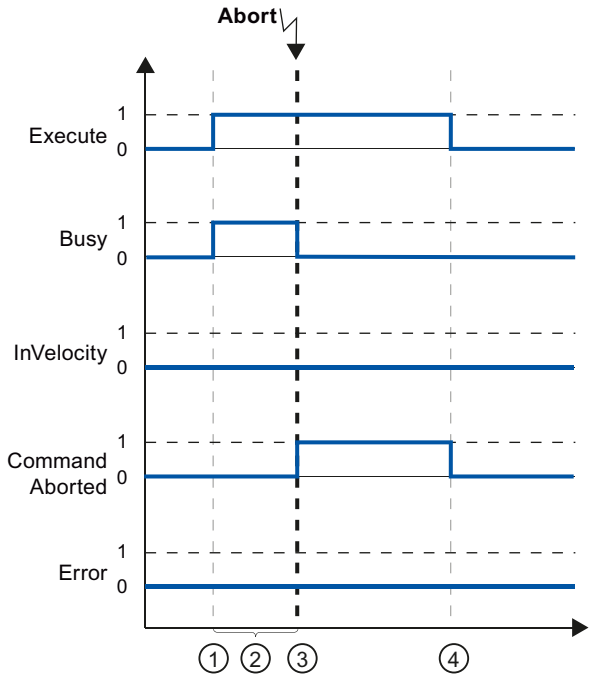

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Abbruch des Auftrags beibehalten
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE.
④	Behält "Execute" den Wert TRUE bis nach Abbruch des Auftrags bei, so bleibt auch "CommandAborted" TRUE und wechselt seinen Zustand gemeinsam mit "Execute" zu FALSE.
⑤	Wurde "Execute" schon vor Abbruch des Auftrags zurück auf FALSE gesetzt, so zeigt "CommandAborted" nur für einen Abarbeitungszyklus den Wert TRUE.

Hinweis

Unter folgenden Bedingungen wird am Ausgangsparameter "CommandAborted" kein Abbruch angezeigt:

Die parametrierte Geschwindigkeit wurde erreicht, der Eingangsparameter "Execute" hat den Wert FALSE und ein neuer Motion Control-Auftrag wird angestoßen.

Vor dem Erreichen der parametrierten Geschwindigkeit tritt ein Fehler auf

Tritt während der Bearbeitung des Motion Control-Auftrags vor dem Erreichen der parametrierten Geschwindigkeit ein Fehler auf, so wird dies am Ausgangsparameter "Error" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Error":

"Execute" wechselt seinen Wert **vor** Auftreten des Fehlers zu FALSE	"Execute" wechselt seinen Wert **nach** Auftreten des Fehlers zu FALSE
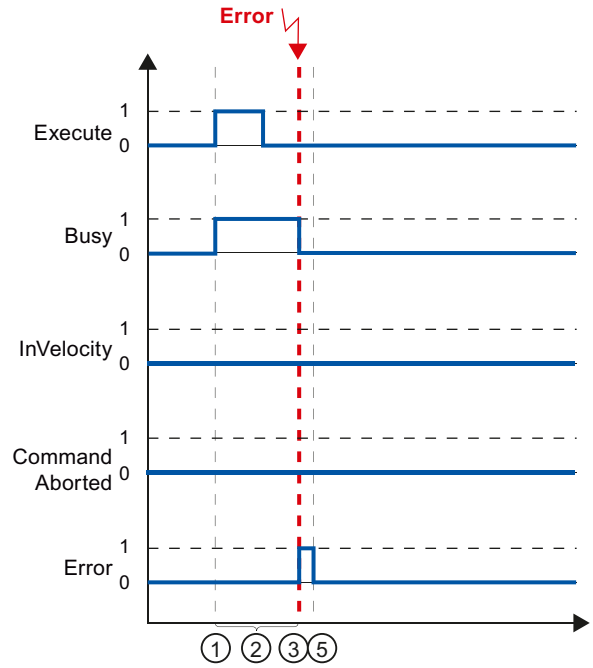	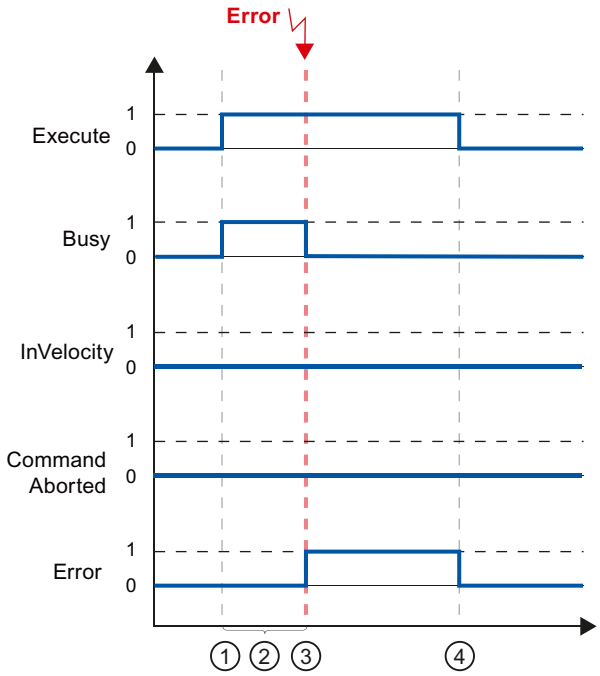

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Auftreten des Fehlers beibehalten
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE..
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE.
④	Behält "Execute" den Wert TRUE bis nach Auftreten des Fehlers bei, so bleibt auch "Error" TRUE und wechselt seinen Zustand erst gemeinsam mit "Execute" zu FALSE.
⑤	Wurde "Execute" schon vor Auftreten des Fehlers zurück auf FALSE gesetzt, so zeigt "Error" lediglich für einen Abarbeitungszyklus den Wert TRUE.

Hinweis

Unter folgenden Bedingungen wird am Ausgangsparameter "Error" kein Fehler angezeigt:

Die parametrierte Geschwindigkeit wurde erreicht, der Eingangsparameter "Execute" hat den Wert FALSE und ein Fehler an der Achse tritt ein (z. B. Software-Endschalter wird angefahren).

Der Fehler der Achse wird nur an der Motion Control-Anweisung "MC_Power" angezeigt.

Motion Control-Anweisung MC_MoveJog

Die Aufträge der Motion Control-Anweisung "MC_MoveJog" realisieren einen Tippbetrieb.

Die Motion Control-Aufträge "MC_MoveJog" haben kein definiertes Ende. Das Auftragsziel ist erfüllt, wenn die parametrierte Geschwindigkeit erstmalig erreicht wird und die Achse mit konstanter Geschwindigkeit fährt. Das Erreichen der parametrierten Geschwindigkeit wird am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

Der Auftrag ist abgeschlossen, wenn der Eingangsparameter "JogForward", bzw. "JogBackward" auf den Wert FALSE gesetzt wurde und die Achse den Stillstand erreicht hat.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht.

Während der Bearbeitung des Motion Control-Auftrags zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wurde der Auftrag abgeschlossen, abgebrochen oder durch einen Fehler gestoppt, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE.

Der Ausgangsparameter "InVelocity" zeigt den Zustand TRUE, solange die Achse mit der parametrierten Geschwindigkeit fährt. Die Ausgangsparameter "CommandAborted" und "Error" zeigen den Status für mindestens einen Zyklus an. Während einer der Eingangsparameter "JogForward" oder "JogBackward" auf TRUE gesetzt ist, werden die Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft in unterschiedlichen Situationen gezeigt:

Die parametrierte Geschwindigkeit wird erreicht und gehalten

Wird der Motion Control-Auftrag bis zum Erreichen der parametrierten Geschwindigkeit ausgeführt, so wird dies am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

Der Tippbetrieb wird durch den Eingangsparameter "JogForward" gesteuert	Der Tippbetrieb wird durch den Eingangsparameter "JogBackward" gesteuert.
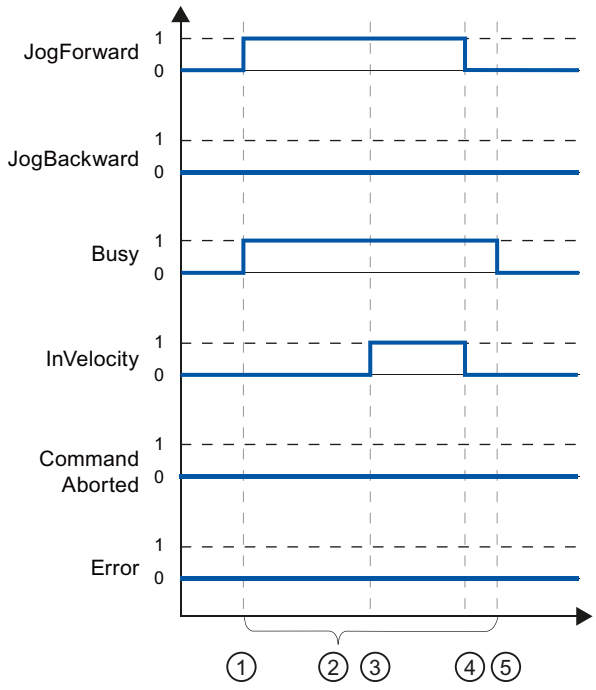	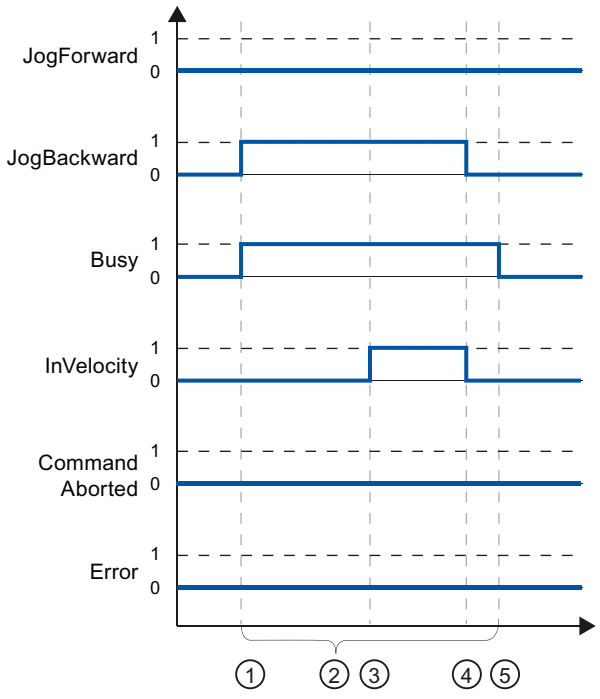

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Mit dem Erreichen der parametrierten Geschwindigkeit wechselt der Ausgangsparameter "InVelocity" nach TRUE.
④	Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wird die Bewegung der Achse beendet. Die Achse beginnt zu verzögern. Dadurch verfährt die Achse nicht länger mit konstanter Geschwindigkeit und der Ausgangsparameter "InVelocity" wechselt seinen Zustand zu FALSE.
⑤	Ist die Achse zum Stillstand gekommen, so ist der Motion Control-Auftrag abgeschlossen und der Ausgangsparameter "Busy" wechselt seinen Wert zu FALSE.

Der Auftrag wird während der Bearbeitung abgebrochen

Wird der Motion Control-Auftrag während der Bearbeitung abgebrochen, so wird dies am Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Das Verhalten ist unabhängig davon, ob die parametrierte Geschwindigkeit erreicht wurde, oder nicht.

Der Tippbetrieb wird durch den Eingangsparameter "JogForward" gesteuert.	Der Tippbetrieb wird durch den Eingangsparameter "JogBackward" gesteuert.
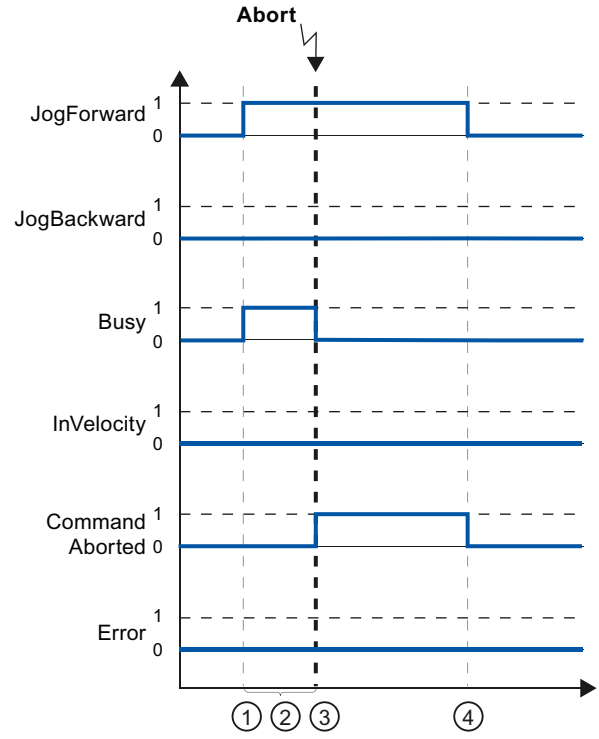	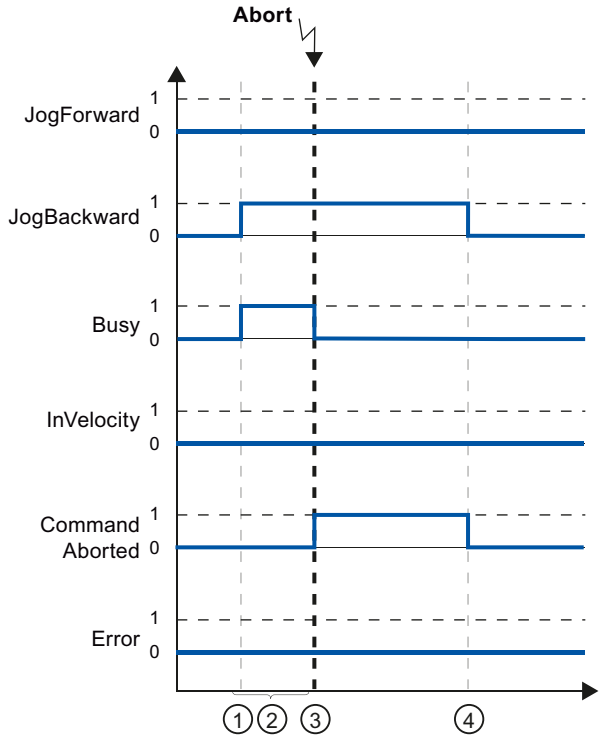

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE.
④	Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wechselt der Ausgangsparameter "CommandAborted" seinen Wert zu FALSE.

Hinweis

Der Auftragsabbruch wird am Ausgangsparameter "CommandAborted" nur für einen Bearbeitungszyklus angezeigt, wenn die nachfolgenden Bedingungen gemeinsam erfüllt sind:

Die Eingangsparameter "JogForward" und "JogBackward" haben den Wert FALSE (die Achse verzögert jedoch noch) und ein neuer Motion Control-Auftrag wird angestoßen.

Während der Auftragsbearbeitung tritt ein Fehler auf

Tritt bei der Bearbeitung des Motion Control-Auftrags ein Fehler auf, so wird dies am Ausgangsparameter "Error" mit dem Wert TRUE angezeigt. Das Verhalten ist unabhängig davon, ob die parametrierte Geschwindigkeit erreicht wurde, oder nicht.

Der Tippbetrieb wird durch den Eingangsparameter "JogForward" gesteuert.	Der Tippbetrieb wird durch den Eingangsparameter "JogBackward" gesteuert.
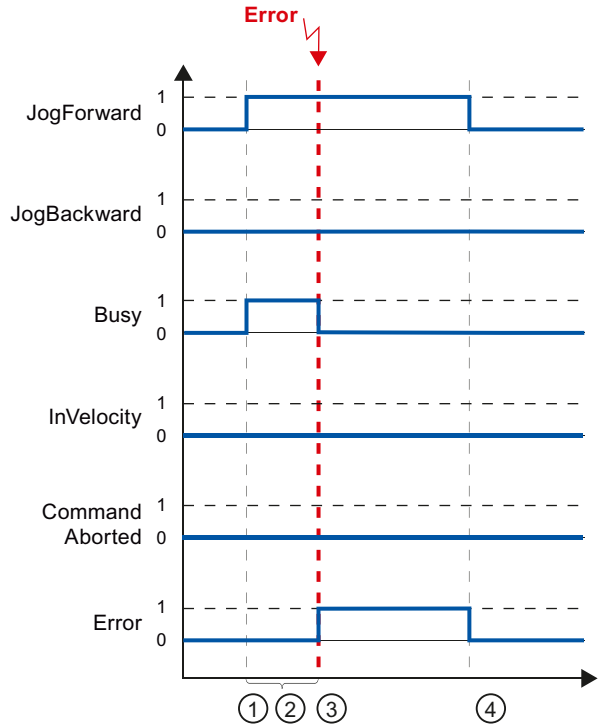	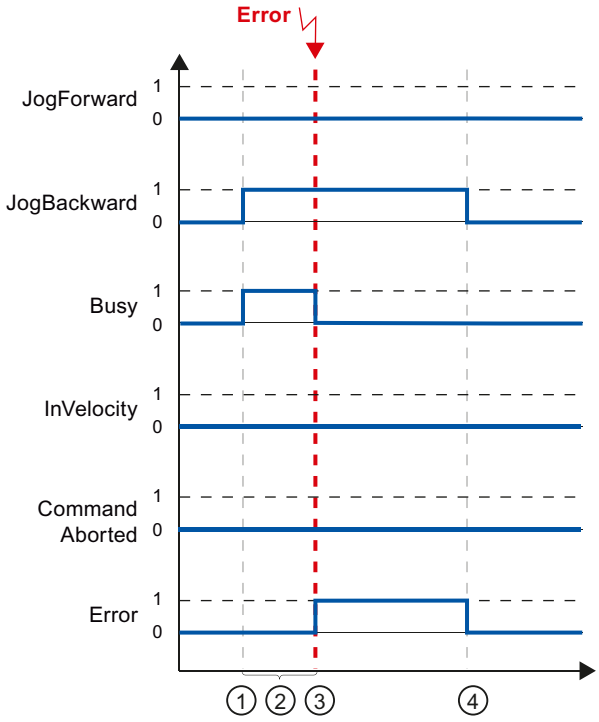

①	Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.
②	Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE.
④	Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wechselt der Ausgangsparameter "Error" seinen Wert zu FALSE.

Hinweis

Ein aufgetretener Fehler wird am Ausgangsparameter "Error" nur für einen Bearbeitungszyklus angezeigt, wenn die nachfolgenden Bedingungen gemeinsam erfüllt sind:

Die Eingangsparameter "JogForward" und "JogBackward" haben den Wert FALSE (die Achse verzögert jedoch noch) und ein Fehler tritt ein (z. B. Software-Endschalter wird angefahren).

Fehleranzeigen der Motion Control-Anweisungen

Die Motion Control-Anweisungen zeigen etwaige Fehler der Motion Control-Aufträge und des Technologieobjekts an den Ausgangsparametern "Error", "ErrorID" und "ErroInfo" der Motion Control-Anweisungen an.

Fehleranzeige an den Ausgangs-Parametern "Error", "ErrorID" und "ErrorInfo"

Zeigt der Ausgangsparameter "Error" den Wert TRUE, so konnte der Auftrag nicht oder nicht vollständig ausgeführt werden. Die Fehlerursache können Sie dem Wert am Ausgangsparameter "ErrorID" entnehmen. Eine Detailinformation zur Fehlerursache können Sie dem Wert am Ausgangs-Parameter "ErrorInfo" entnehmen. Bei der Fehleranzeige wird zwischen folgenden Fehlerklassen unterschieden:

- **Betriebsfehler mit Stopp der Achse (z. B. "HW-Endschalter wurde angefahren")**
  Betriebsfehler mit Stopp der Achse sind Fehler, die zur Laufzeit des Anwenderprogramms auftreten. Befindet sich die Achse in Bewegung, so wird sie je nach Fehler mit der konfigurierten oder der Notstopp-Verzögerung gestoppt. Die Fehler werden an der fehlerauslösenden Motion Control-Anweisung und an der Motion Control-Anweisung "MC_Power" angezeigt.
- **Betriebsfehler ohne Stopp der Achse (z. B. "Achse ist nicht referenziert")**
  Betriebsfehler ohne Stopp der Achse sind Fehler, die zur Laufzeit des Anwenderprogramms auftreten. Befindet sich die Achse in Bewegung, so wird die Bewegung fortgesetzt. Die Fehler werden nur an der fehlerauslösenden Motion Control-Anweisung angezeigt.
- **Parametrierfehler der Motion Control-Anweisung (z. B. "Falscher Wert am Parameter "Velocity")**
  Zu Parametrierfehlern kommt es bei fehlerhaften Angaben an den Eingangsparametern der Motion Control-Anweisungen. Befindet sich die Achse in Bewegung, so wird die Bewegung fortgesetzt. Die Fehler werden nur an der fehlerauslösenden Motion Control-Anweisung angezeigt.
- **Konfigurationsfehler am Technologieobjekt "Achse" (z. B. "Wert für "Beschleunigung" ist ungültig")**
  Ein Konfigurationsfehler liegt vor, wenn in der Konfiguration der Achse ein oder mehrere Parameter fehlerhaft konfiguriert oder änderbare Konfigurationsdaten während der Laufzeit des Programms fehlerhaft geändert wurden. Ist die Achse in Bewegung, so wird sie mit der konfigurierten Notstopp-Verzögerung gestoppt. Der Fehler wird an der fehlerauslösenden Motion Control-Anweisung und an der Motion Control-Anweisung "MC_Power" angezeigt.
- **Konfigurationsfehler am Technologieobjekt "Auftragstabelle" (z. B. "Wert für "Geschwindigkeit" ist ungültig")**
  Ein Konfigurationsfehler liegt vor, wenn in der Konfiguration der Auftragstabelle ein oder mehrere Parameter fehlerhaft konfiguriert oder änderbare Konfigurationsdaten während der Laufzeit des Programms fehlerhaft geändert wurden. Befindet sich die Achse in Bewegung, so wird diese fortgesetzt. Die Fehler werden nur an der Motion Control-Anweisung "MC_CommandTable" angezeigt.
- **Interner Fehler**
  Beim Auftreten eines internen Fehlers wird die Achse gestoppt. Die Fehler werden an der fehlerauslösenden Motion Control-Anweisung und teilweise an der Motion Control-Anweisung "MC_Power" angezeigt.

Eine detaillierte Beschreibung der ErrorIDs und ErrorInfos als auch deren Abhilfen finden Sie im Anhang (Seite 1604).

Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 1477)

Anwenderprogramm erstellen (Seite 1478)

Programmierhinweise (Seite 1481)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 1483)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Verfolgung laufender Aufträge (Seite 1483)

Restart von Technologieobjekten

Beschreibung

Systemseitig werden die Technologieobjekte automatisch nach dem Einschalten der CPU bzw. nach dem Laden in die CPU mit den Startwerten aus dem Technologie-Datenbaustein initialisiert. Wenn bei erneutem Laden in die CPU Restart-relevante Änderungen festgestellt werden, wird automatisch ein Restart des Technologieobjekts durchgeführt.

Wenn Restart-relevante Daten im RUN durch das Anwenderprogramm geändert wurden, muss das Technologieobjekt zur Übernahme der Änderungen anwenderseitig neu initialisiert werden.

Wenn Änderungen im Technologie-Datenbaustein auch nach dem Restart des Technologieobjekts erhalten bleiben sollen, müssen Sie die Änderungen mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert im Ladespeicher schreiben.

Restart erforderlich

Wenn ein Restart des Technologieobjekts erforderlich ist, wird dies unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Statusmeldungen > Restart erforderlich" und in der Variablen des Technologieobjekts <Achsname>.StatusBits.RestartRequired angezeigt.

Restart eines Technologieobjektes

Ein Restart des Technologieobjekts wird anwenderseitig durch die Motion Control-Anweisung "MC_Reset" mit Parameter "Restart" = TRUE ausgelöst.

Durch einen Restart wird der Status "Referenziert" eines Technologieobjekts mit inkrementellen Istwerten zurückgesetzt (<Achsname>.StatusBits.HomingDone).

Parameterübergabe für Funktionsbausteine

Wenn Sie einen Funktionsbaustein mit Motion Control-Anweisungen für verschiedene Technologieobjekte wiederverwenden wollen, erzeugen Sie in der Bausteinschnittstelle des Funktionsbausteins einen Eingangsparameter vom Datentyp des jeweiligen Technologieobjekts. Den Datentyp weisen Sie in der Bausteinschnittstelle über direkte Eingabe zu. Dieser Parameter wird dann als Referenz auf das Technologieobjekt an den Parameter "axis" der Motion Control-Anweisungen übergeben. Die Datentypen von Technologieobjekten entsprechen der Struktur des zugehörigen Technologie-Datenbausteins.

Durch die Angabe des Datentyps lassen sich die Variablen des Technologieobjekts im Funktionsbaustein adressieren (<Parameter der Bausteinschnittstelle>.<Variable des Technologieobjekts>).

Wenn Sie keinen Zugriff auf die Variablen des Technologieobjekts benötigen, können Sie den Datentyp "DB_ANY" verwenden. Mit Hilfe des Datentyps "DB_Any" ist eine variablere Programmierung möglich.

Folgende Tabelle zeigt die Datentypen für die Referenz auf die Technologieobjekte:

Technologieobjekt	Datentyp für die Referenz auf das Technologieobjekt
Positionierachse	TO_PositioningAxis
Auftragstabelle	TO_CommandTable

Beispiel 1

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

Operand	Deklaration	Datentyp	Beschreibung
axis	Input	TO_PositioningAxis	Referenz auf das Technologieobjekt
on	Input	BOOL	Signal zum Freigeben der Achse
actPosition	Output	Real	Abfrage der Istposition aus dem Technologie-Datenbaustein
instMC_POWER	Static	MC_POWER	Multiinstanz der Motion Control-Anwesiung MC_Power

Das folgende SCL-Programm zeigt die Umsetzung der Aufgabe:

SCL	Erläuterung
`#instMC_POWER(Axis := #axis, Enable := #on);`	`//Aufruf der Motion Control-Anweisung MC_Power mit Freigabe der Achse`
`#actPosition := #axis.ActualPosition;`	`//Abfrage der Istposition aus dem Technologie-Datenbaustein`

Beispiel 2

Eine weitere Möglichkeit für die Übergabe der Datentypen eines Technologieobjekts bietet der Datentyp "DB_Any". Der Datentyp "DB_Any" kann im Programm zur Laufzeit zugewiesen werden.

Das Beispiel zeigt zwei Möglichkeiten der Weitergabe von technologiespezifischen Datentypen an eine entsprechende Anweisung, z. B. "MC_CommandTable", die als Multiinstanz angelegt ist. Die 1. Möglichkeit zeigt die Verwendung des Datentyps "TO_PositioningAxis". Die 2. Möglichkeit zeigt die einfache Weitergabe des

Technologieobjekts Auftragstabelle in Abhängigkeit des Eingangs "cmdTablToUse". Je nach Wert am Eingang wird einer der drei Eingänge "cmdTablx" an die Anweisung "MC_CommandTable" mittels "tempCmdTableSel" weitergegeben.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Variable	Deklaration	Datentyp	Beschreibung
axis	Input	TO_PositioningAxis	Positionierachse
cmdTabl1	Input	DB_ANY	1. Auftragstabelle
cmdTabl2	Input	DB_ANY	2. Auftragstabelle
cmdTabl3	Input	DB_ANY	3. Auftragstabelle
cmdTablToUse	Input	Int	Auswahl Auftragstabelle 1 bis 3
instMC_Com-mandTable	Static	MC_CommandTable	Multiinstanz des MC_CommandTable
tempCmdTable-Sel	Temp	DB_ANY	Aktuelle Auftragstabelle

Das folgende Beispiel zeigt die prinzipielle Vorgehensweise:

```
SCL                                                 Erläuterung
CASE #cmdTablToUse OF
    1:  #tempCmdTableSel := #cmdTabl1;              //Programm für Fall 1
    2:  #tempCmdTableSel := #cmdTabl2;              //Programm für Fall 2
    3:  #tempCmdTableSel := #cmdTabl3;              //Programm für Fall 3
    ELSE                                            //Programm für alle anderen Werte
        #tempCmdTableSel := #cmdTabl1;              //-> Voreinstellung 1. Auftragstabelle
END_CASE;
                                                    //Aufruf der Anweisung "MC_CommandTable"
                                                    //mit variabler Übergabe der Technologieobjekte
#instMC_CommandTable(Axis:=#axis,                   //Zuweisung der Achse
    CommandTable:=#temCmdTableSel);                 //Indirekte Zuweisung der Auftragstabelle
```

Weitere Informationen

Weitere Programmbeispiele unter Verwendung des Datentyps "DB_Any" finden Sie im folgenden FAQ:

https://support.industry.siemens.com/cs/ww/de/view/109750880 (https://support.industry.siemens.com/cs/ww/de/view/109750880)

6.1.1.10 Achse - Diagnose

Status- und Fehlerbits (Technologieobjekte ab V4)

Mit der Diagnosefunktion "Status- und Fehlerbits" überwachen Sie im TIA-Portal die wichtigsten Status- und Fehlermeldungen der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Status- und Fehlermeldungen haben folgende Bedeutung:

Statusmeldungen

Statusmeldung - Achse	Beschreibung
Freigegeben	Die Achse ist freigegeben und bereit über Motion Control-Aufträge gesteuert zu werden. (Variable des Technologieobjekts: <Achsname>.StatusBits.Enable)
Referenziert	Die Achse ist referenziert und kann absolute Positionieraufträge der Motion Control-Anweisung "MC_MoveAbsolute" ausführen. Zum relativen Positionieren muss die Achse nicht referenziert sein. Sonderfälle: • Während des aktiven Referenzierens ist der Status FALSE. • Wird eine referenzierte Achse passiv referenziert, so ist der Status während des passiven Referenzierens TRUE. (Variable des Technologieobjekts: <Achsname>.StatusBits.HomingDone)
Achsfehler	Am Technologieobjekt "Achse" ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern ErrorID und ErrorInfo der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt. (Variable des Technologieobjekts: <Achsname>.StatusBits.Error)
Steuertafel aktiv	Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt "Achse". Die Achse kann nicht vom Anwenderprogramm gesteuert werden. (Variable des Technologieobjekts: <Achsname>.StatusBits.ControlPanelActive)
Restart erforderlich	Eine geänderte Konfiguration der Achse wurde im Betriebszustand RUN der CPU in den Ladespeicher geladen. Um die geänderte Konfiguration in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung MC_Reset. (Variable des Technologieobjekts: <Achsname>.StatusBits.RestartRequired)

Statusmeldung - Antrieb	Beschreibung
Bereit	Der Antrieb ist betriebsbereit. (Variable des Technologieobjekts: <Achsname>.StatusBits.DriveReady)
Antriebsfehler	Der Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt. (Variable des Technologieobjekts: <Achsname>.ErrorBits.DriveFault)

Statusmeldung - Bewegung	Beschreibung
Stillstand	Die Achse befindet sich im Stillstand. (Variable des Technologieobjekts: <Achsname>.StatusBits.StandStill)
Beschleunigung	Die Achse beschleunigt. (Variable des Technologieobjekts: <Achsname>.StatusBits.Accelerating)
Konstante Geschwindigkeit	Die Achse verfährt mit konstanter Geschwindigkeit. (Variable des Technologieobjekts: <Achsname>.StatusBits.ConstantVelocity)
Verzögerung	Die Achse verzögert (bremst ab). (Variable des Technologieobjekts: <Achsname>.StatusBits.Decelerating)

Statusmeldung - Bewegungsart	Beschreibung
Positionieren	Die Achse führt einen Positionierauftrag der Motion Control-Anweisung "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.PositioningCommand)
Mit Geschwindigkeitsvorgabe verfahren	Die Achse führt einen Auftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisung "MC_MoveVelocity", "MC_MoveJog" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.VelocityCommand)
Referenzieren	Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.HomingCommand)
Auftragstabelle aktiv	Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert. (Variable des Technologieobjekts: <Achsname>.StatusBits.CommandTableActive)

Endschalter-Statusmeldungen

Endschalter-Statusmeldung	Beschreibung
Unterer SW-Endschalter wurde angefahren	Ein Software-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.StatusBits.SWLimitMinActive)
Oberer SW-Endschalter wurde angefahren	Ein Hardware-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.StatusBits.SWLimitMaxActive)
Unterer HW-Endschalter wurde angefahren	Der untere Hardware-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.StatusBits.HWLimitMinActive)
Oberer HW-Endschalter wurde angefahren	Der obere Hardware-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.StatusBits.HWLimitMaxActive)

Fehlermeldungen

Fehlermeldung	Beschreibung
SW-Endschalter wurde angefahren	Ein Software-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SWLimit)
HW-Endschalter wurde angefahren	Ein Hardware-Endschalter wurde angefahren oder überfahren. (Variable des Technologieobjekts: <Achsname>.ErrorBits.HWLimit)
Unzulässige Bewegungsrichtung	Die Bewegungsrichtung des Auftrags entspricht nicht der konfigurierten Bewegungsrichtung. (Variable des Technologieobjekts: <Achsname>.ErrorBits.DirectionFault)
PTO bereits in Verwendung	Eine zweite Achse verwendet den selben PTO (Pulse Train Output) und HSC (High Speed Counter) und ist mit "MC_Power" freigegeben. (Variable des Technologieobjekts: <Achsname>.ErrorBits.HWUsed)
Geber	Fehler im Gebersystem. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SensorFault)
Datenaustausch	Die Kommunikation mit einem verbundenen Gerät ist gestört. (Variable des Technologieobjekts: <Achsname>.ErrorBits.CommunicationFault)
Positionieren	Die Achse wurde am Ende einer Positionierbewegung nicht korrekt positioniert. (Variable des Technologieobjekts: <Achsname>.ErrorBits.PositionigFault)
Schleppfehler	Der maximale zulässige Schleppfehler wurde überschritten. (Variable des Technologieobjekts: <Achsname>.ErrorBits.FollowingErrorFault)
Geberwerte sind ungültig	Die Geberwerte sind ungültig. (Variable des Technologieobjekts: <Achsname>.StatusSensor.State)
Konfigurationsfehler	Das Technologieobjekt "Achse" wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (Variable des Technologieobjekts: <Achsname>.ErrorBits.ConfigFault)
Interner Fehler	Es ist ein interner Fehler aufgetreten. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SystemFault)

Im darunter befindlichen Ausgabefenster wird der erste gemeldete und noch nicht quittierte Fehler angezeigt.

Siehe auch

Variablen StatusBits V4...5 (Seite 1670)

Variablen ErrorBits V4...5 (Seite 1675)

Diagnose - Status- und Fehlerbits (Technologieobjekt "Achse" V1...3) (Seite 1602)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Bewegungsstatus (Seite 1502)

Bewegungsstatus

Mit der Diagnosefunktion "Bewegungsstatus" überwachen Sie im TIA-Portal den Bewegungsstatus der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Statusinformationen haben folgende Bedeutung:

Status	Beschreibung
Istposition	Das Feld "Istposition" zeigt die gemessene Position der Achse an. Bei nicht referenzierter Achse zeigt der Wert den Positionswert relativ zur Freigabe-Position der Achse an. (Variable des Technologieobjekts: <Achsname>.ActualPosition)
Istgeschwindigkeit	Das Feld "Istgeschwindigkeit" zeigt die gemessene Geschwindigkeit der Achse an. (Variable des Technologieobjekts: <Achsname>.ActualVelocity)
Sollposition	Das Feld "Sollposition" zeigt die berechnete Sollposition der Achse an. Bei nicht referenzierter Achse zeigt der Wert den Positionswert relativ zur Freigabe-Position der Achse an. (Variable des Technologieobjekts: <Achsname>.Position)
Sollgeschwindigkeit	Das Feld "Sollgeschwindigkeit" zeigt die berechnete Sollgeschwindigkeit der Achse an. (Variable des Technologieobjekts: <Achsname>.Velocity)
Zielposition	Das Feld "Zielposition" zeigt die aktuelle Zielposition eines aktiven Positionierauftrags oder der Achssteuertafel an. Der Wert der "Zielposition" ist nur während der Ausführung eines Positionierauftrags gültig. (Variable des Technologieobjekts: <Achsname>.StatusPositioning.TargetPosition)
Verbleibender Verfahrweg	Das Feld "Verbleibender Verfahrweg" zeigt den aktuell verbleibenden Verfahrweg eines aktiven Positionierauftrags oder der Achssteuertafel an. Der Wert "Verbleibender Verfahrweg" ist nur während der Ausführung eines Positionierauftrags gültig. (Variable des Technologieobjekts: <Achsname>.StatusPositioning.Distance)

Siehe auch

Variablen zu Positions- und Geschwindigkeitswerten V4...5 (Seite 1652)

Variablen StatusPositioning V4...5 (Seite 1668)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

Variablen zu Positions- und Geschwindigkeitswerten ab V6 (Seite 1544)

Variablen MotionStatus V1...3 (Seite 1646)

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

Dynamikeinstellungen

Mit der Diagnosefunktion "Dynamikeinstellungen" kontrollieren Sie im TIA-Portal die konfigurierten Dynamikgrenzwerte der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Statusinformationen haben folgende Bedeutung:

Dynamische Grenze	Beschreibung
Beschleunigung	Das Feld "Beschleunigung" zeigt die aktuell konfigurierte Beschleunigung der Achse an. (Variable des Technologieobjekts: <Achsname>.DynamicDefaults.Acceleration)
Verzögerung	Das Feld "Verzögerung" zeigt die aktuell konfigurierte Verzögerung der Achse an. (Variable des Technologieobjekts: <Achsname>.DynamicDefaults.Deceleration)
Notstopp-Verzögerung	Das Feld "Notstopp-Verzögerung" zeigt die aktuell konfigurierte Notstopp-Verzögerung der Achse an. (Variable des Technologieobjekts: <Achsname>.DynamicDefaults.EmergencyDeceleration)
Ruck (ab Technologieobjekt Achse V2)	Das Feld zeigt den aktuell konfigurierten Ruck der Achse an. (Variable des Technologieobjekts: <Achsname>.DynamicDefaults.Jerk)

Siehe auch

Variablen DynamicDefaults V4...5 (Seite 1662)

Kompatibilitätsliste der Variablen V1...3 <-> V4...5 (Seite 1376)

PROFIdrive-Telegramm

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > PROFIdrive-Telegramm" überwachen Sie im TIA Portal die PROFIdrive-Telegramme von Antrieb und Geber. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Antrieb"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Antrieb zur Steuerung angezeigt:

- Die Zustandsworte "ZSW1" und "ZSW2"
- Die an den Antrieb ausgegebene Solldrehzahl (NSOLL)
- Die vom Antrieb gemeldete Istdrehzahl (NIST)

Bereich "Geber"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Geber zur Steuerung angezeigt:

- Das Zustandswort "G1_ZSW"
- Der Positionsistwert "G1_XIST1" (zyklischer Geberistwert)
- Der Positionsistwert "G1_XIST2" (Absolutwert des Gebers)

6.1.1.11 Anhang

Mehrere Achsen mit gleichem PTO einsetzen

Nutzen Sie die Motion Control-Funktionalität der CPU S7-1200, um mehrere Technologieobjekte Positionierachse mit dem gleichen PTO (Pulse Train Output) und somit mit den gleichen CPU-Ausgängen zu betreiben. Dies ist z. B. sinnvoll, wenn über einen PTO verschiedene Achskonfigurationen für unterschiedliche Produktionsabläufe genutzt werden sollen. Zwischen diesen Achskonfigurationen kann, wie nachfolgend beschrieben, beliebig gewechselt werden. Entnehmen Sie die prinzipiellen funktionalen Zusammenhänge der nachfolgenden Darstellung:

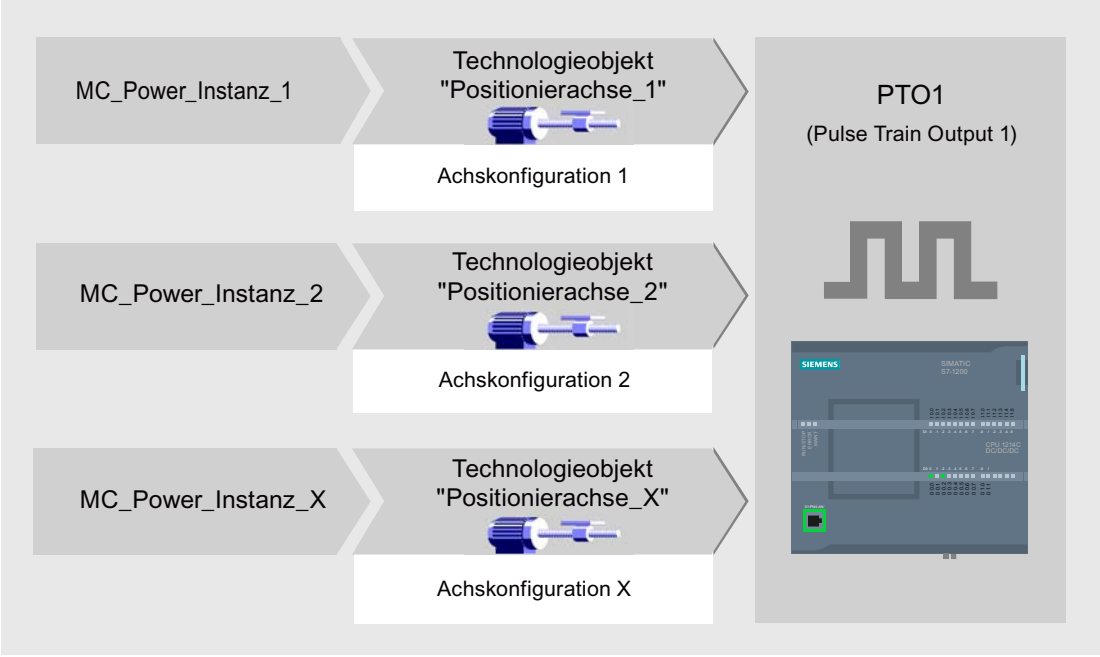

Im dargestellten Beispiel verwenden mehrere Technologieobjekte Positionierachse mit jeweils eigener Achskonfiguration den gleichen PTO. Im Anwenderprogramm muss jede Achse mit einem eigenen Aufruf der Motion Control-Anweisung "MC_Power" mit eigenem Instanz-Datenbaustein aufgerufen werden. Zu jedem Zeitpunkt darf nur jeweils eine Achse den PTO verwenden. Die Achse, die augenblicklich den PTO verwendet, zeigt dies mit der Variablen <Achsname>.StatusBits.Activated = TRUE an.

Wechsel des Technologieobjekts Positionierachse

Wie Sie zwischen verschiedenen Technologieobjekten und somit zwischen verschiedenen Achskonfigurationen wechseln, können Sie dem nachfolgend beschriebenen Programmschema entnehmen. Um mit mehreren Achsen ohne Fehleranzeigen den selben PTO verwenden zu können, dürfen nur die Motion Control-Anweisungen der aktuell zu verwendenden Achse aufgerufen werden.

Die folgende Darstellung zeigt dies am Beispiel der Motion Control-Anweisung "MC_Power":

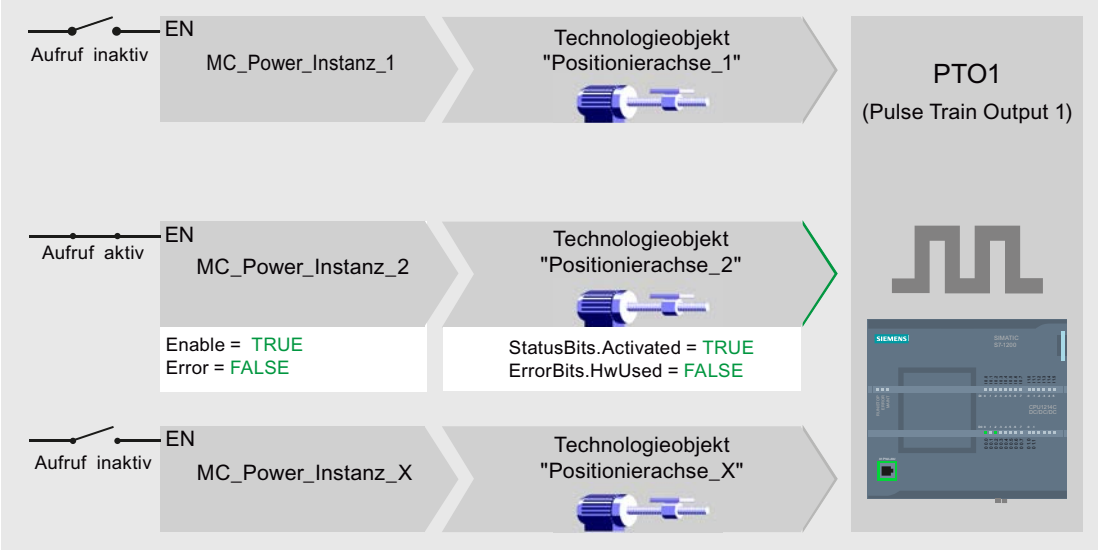

Die Variable der aktivierten Achse (hier "Positionierachse_2") zeigen im Anwenderprogramm folgende typischen Anzeigen:

- <Achsname>.StatusBits.Activated = TRUE
- <Achsname>.ErrorBits.HWUsed = FALSE

Gehen Sie wie nachfolgend beschrieben vor, um das Technologieobjekt Positionierachse zu wechseln. Im Beispiel soll von "Positionierachse_2" zu "Positionierachse_1" gewechselt werden:

1. Beenden Sie evt. laufende Verfahrbewegungen der aktivierten "Positionierachse_2"
2. Sperren Sie "Positionierachse_2" mit der dazugehörigen Motion Control-Anweisung "MC_Power" über den Eingangsparameter Enable = FALSE
3. Prüfen Sie das erfolgte Sperren der "Positionierachse_2" mit einer UND-Verknüpfung des Ausgangsparameters Status = FALSE der Motion Control-Anweisung "MC_Power" und der Variable des Technologieobjekts <Achsname>.StatusBits.Enable = FALSE.
4. Deaktivieren Sie den bedingten Aufruf der Motion Control-Anweisungen für "Positionierachse_2"
5. Aktivieren Sie den bedingten Aufruf der Motion Control-Anweisungen für "Positionierachse_1". Mit dem ersten Aufruf der entsprechenden Motion Control-Anweisung "MC_Power" wird "Positionierachse_2" deaktiviert und "Positionierachse_1" aktiviert.
6. Geben Sie "Positionierachse_1" mit der Motion Control-Anweisung "MC_Power" über den Eingangsparameter Enable = TRUE frei.
7. Prüfen Sie die erfolgte Freigabe der "Positionierachse_1" mit einer UND-Verknüpfung des Ausgangsparameters Status = TRUE der Motion Control-Anweisung "MC_Power" und der Variable des Technologieobjekts <Achsname>.StatusBits.Enable = TRUE.

Grundsätzlich ist auch der zyklische Aufruf aller Motion Control-Anweisungen aller mit einem PTO arbeitenden Achsen möglich:

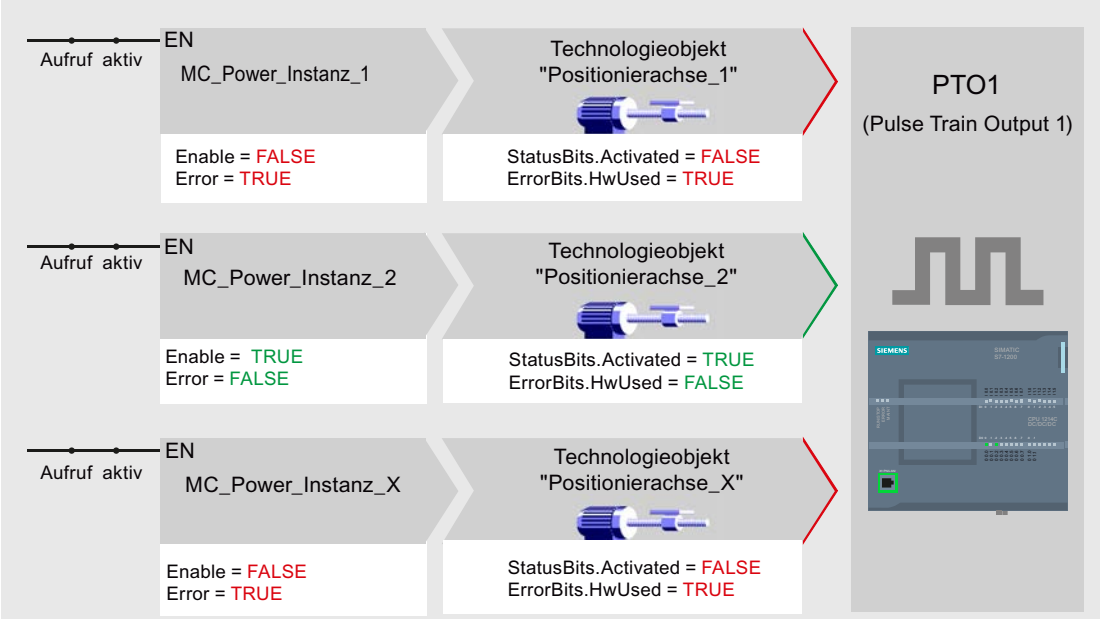

Mit dem Freigeben einer Achse (hier "Positionierachse_2") wird diese aktiviert.

Im Gegensatz zum bedingten Aufruf zeigen die Motion Control-Anweisungen der deaktivierten Achsen (hier "Positionierachse_1" und "Positionierachse_x") Fehler an. Die Variablen dieser Achsen zeigen den Zustand <Achsname>.StatusBits.Activated = FALSE und <Achsname>.ErrorBits.HWUsed = TRUE.

Setzen Sie den bedingten Aufruf der Motion Control-Anweisungen ein, wenn Sie das Anwenderprogramm ohne Fehleranzeigen realisieren möchten.

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Mehrere Antriebe mit gleichem PTO einsetzen (Seite 1507)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 1508)

Sonderfälle beim Einsatz von Software-Endschaltern bei Antriebsanbindung über PTO (Seite 1510)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Mehrere Antriebe mit gleichem PTO einsetzen

Sollen mehrere Antriebe alternativ genutzt werden, so können diese über eine Umschaltung an einem gemeinsamen PTO (Pulse Train Output) betrieben werden. Die nachfolgende Darstellung zeigt den prinzipiellen Schaltungsaufbau:

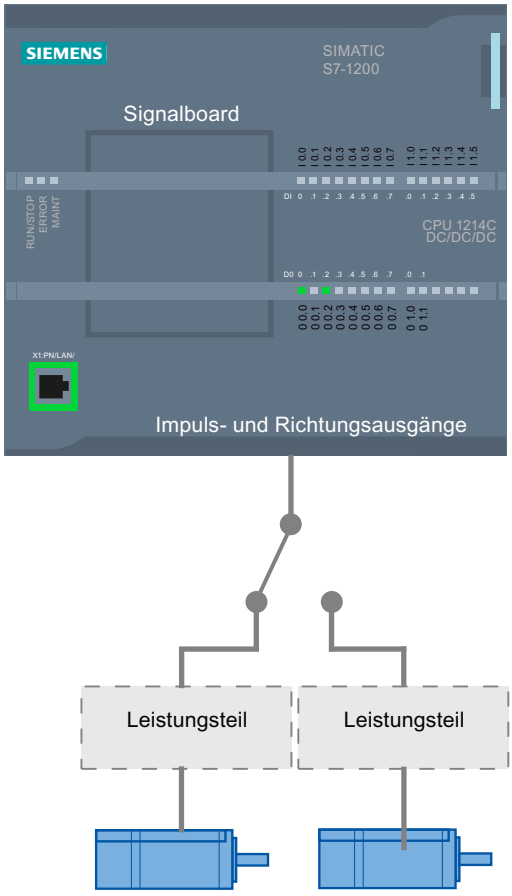

Die Umschaltung der Antriebe kann bei Bedarf durch das Anwenderprogramm über einen digitalen Ausgang gesteuert werden. Sind für die verschiedenen Antriebe unterschiedliche Achskonfigurationen nötig, so müssen diese für den PTO entsprechend umgeschaltet werden. Weitere Informationen hierzu finden Sie unter "Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)".

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 1508)

Sonderfälle beim Einsatz von Software-Endschaltern bei Antriebsanbindung über PTO (Seite 1510)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen

Je nach Anwendung kann es nötig sein, Motion Control-Aufträge (z. B. alarmgesteuert) in einer höheren Prioritätsklasse (Ablaufebene) zu starten.

Die Motion Control-Anweisungen müssen zur Statusverfolgung in kurzen Zeitabständen aufgerufen werden. Werden die Motion Control-Anweisungen in der höheren Prioritätsklasse nur einmalig oder in zu großen Zeitabständen aufgerufen, so können die Motion Control-Aufträge nicht ausreichend nachverfolgt werden. In diesem Fall besteht die Möglichkeit der Nachverfolgung im Zyklus-OB. Für jeden Start eines Motion Control-Auftrags in der höheren Prioritätsklasse muss ein momentan nicht verwendeter Instanz-Datenbaustein vorhanden sein. Entnehmen Sie dem nachfolgenden Flussdiagramm, wie Sie Motion Control-Aufträge in einer höheren Prioritätsklasse (z. B. Prozessalarm-OB) starten und im Zyklus-OB weiterverfolgen können:

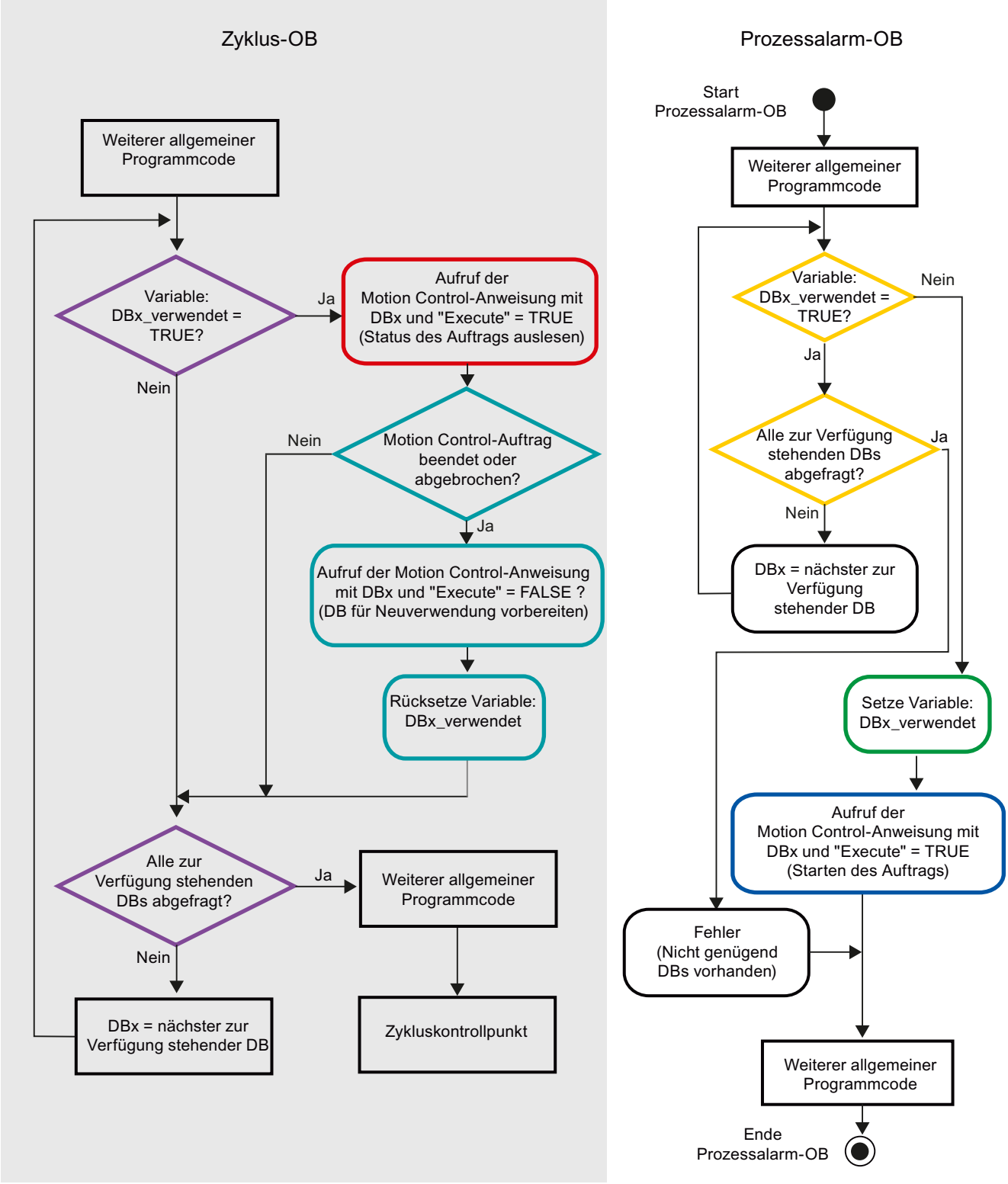

Je nach Häufigkeit der zu startenden Motion Control-Aufträge muss eine ausreichende Anzahl von Instanz-Datenbausteinen erzeugt worden sein. Welcher Instanz-Datenbaustein

augenblicklich in Verwendung ist, muss anwenderseitig in den Variablen DBx_verwendet verwaltet werden.

Start des Motion Control-Auftrags im Prozessalarm-OB

Über binäre Abfragen der Variablen DBx_verwendet (Orange) wird ein Instanz-Datenbaustein gesucht, der augenblicklich nicht in Verwendung ist. Wird ein solcher gefunden, so wird der verwendete Instanz-Datenbaustein als "verwendet" gekennzeichnet (Grün) und der Motion Control-Auftrag mit diesem Instanz-Datenbaustein gestartet (Blau).

Anschließend werden evt. weitere Programmteile des Prozessalarm-OBs bearbeitet und danach in den Zyklus-OB zurückgekehrt.

Verfolgung der gestarteten Motion-Control-Aufträge im Zyklus-OB

Im Zyklus-OB werden alle zur Verfügung stehenden Instanz-Datenbausteine auf ihre Verwendung über die Variable DBx_verwendet geprüft (Violett).

Ist ein Instanz-Datenbaustein in Verwendung (Motion Control-Auftrag ist in Bearbeitung), so wird die Motion Control-Anweisung mit diesem Instanz-Datenbaustein und dem Eingangsparameter Execute = TRUE aufgerufen, um die Statusmeldungen auszulesen (Rot).

Ist der Auftrag abgeschlossen oder wurde er abgebrochen, so wird mit folgenden Maßnahmen fortgesetzt (Blaugrün):

- Aufruf der Motion Control-Anweisung mit dem Eingangsparameter Execute = FALSE
- Rücksetzen der Variable DBx_verwendet

Hiermit ist die Auftragsverfolgung abgeschlossen und der Instanz-Datenbaustein steht für eine erneute Verwendung wieder zur Verfügung.

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)

Mehrere Antriebe mit gleichem PTO einsetzen (Seite 1507)

Sonderfälle beim Einsatz von Software-Endschaltern bei Antriebsanbindung über PTO (Seite 1510)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Sonderfälle beim Einsatz von Software-Endschaltern bei Antriebsanbindung über PTO

Software-Endschalter im Zusammenhang mit einem Referenziervorgang

Durch ungünstig parametrierte Referenzieraufträge kann der Bremsvorgang der Achse auf den Software-Endschalter beeinflusst werden. Berücksichtigen Sie die nachfolgenden Beispiele bei Ihrer Programmierung.

Beispiel 1:

Während eines Verfahrbefehls wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Es ist noch möglich, die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen:

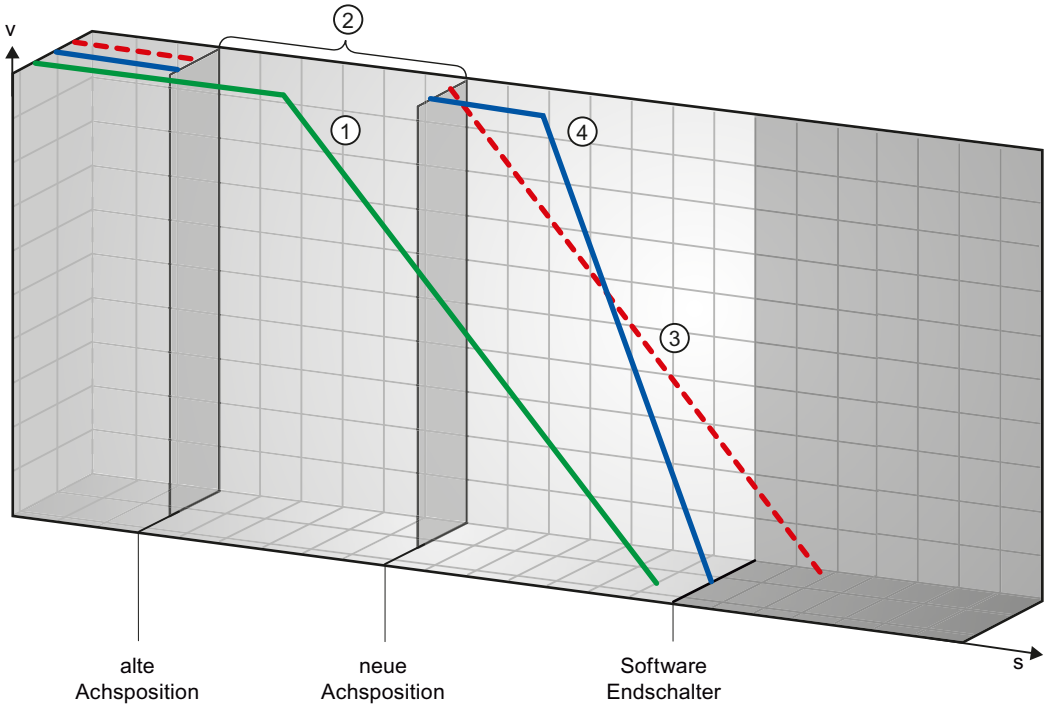

①	Der grüne Kurvenverlauf zeigt die Bewegung **ohne** Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".
③	Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
④	Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der konfigurierten Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen.

Beispiel 2:

Während eines Verfahrbefehls wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Im Gegensatz zu Beispiel 1 ist es nicht mehr möglich, die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen. Die Achse fährt über die Position des Software-Endschalters hinaus.

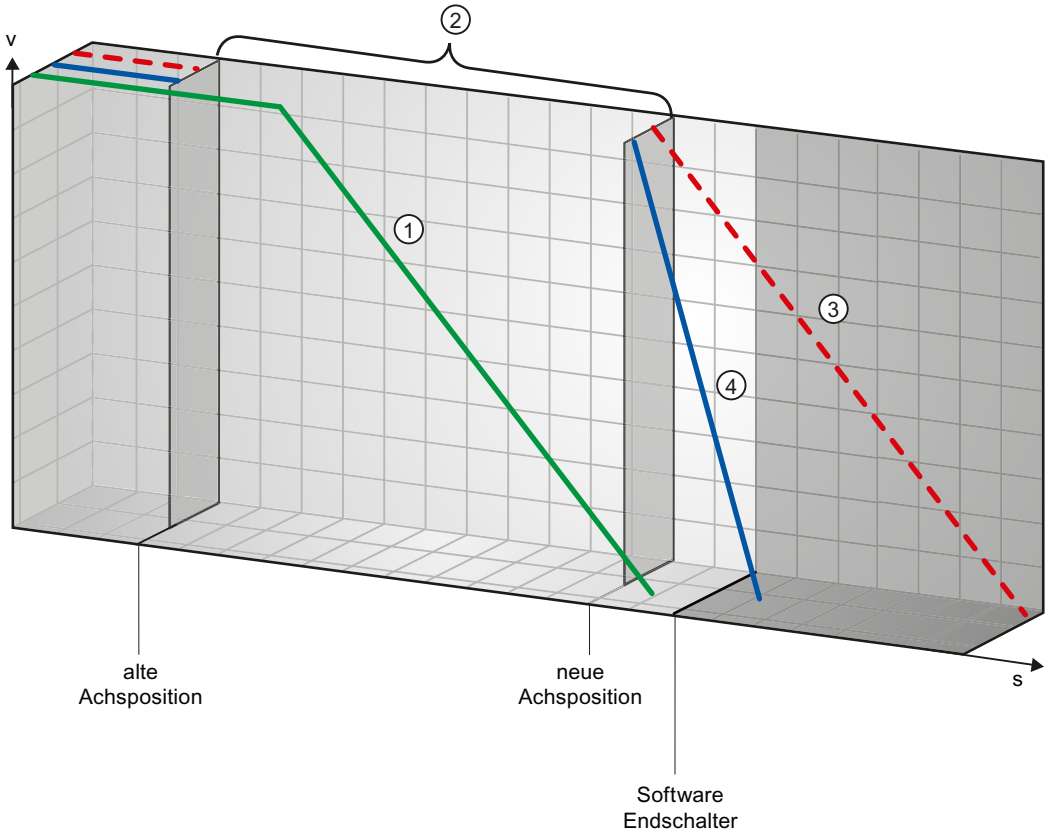

①	Der grüne Kurvenverlauf zeigt die Bewegung **ohne** Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".
③	Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
④	Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst mit der konfigurierten Notstopp-Verzögerung ab. Jedoch reicht die Notstopp-Verzögerung nicht aus, um die Achse auf der Position des Software-Endschalters zum Stehen zu bringen. Die Position des Software-Endschalters wird überfahren.

Beispiel 3:

Während eines Abbremsvorgangs wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Es ist noch möglich, die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen:

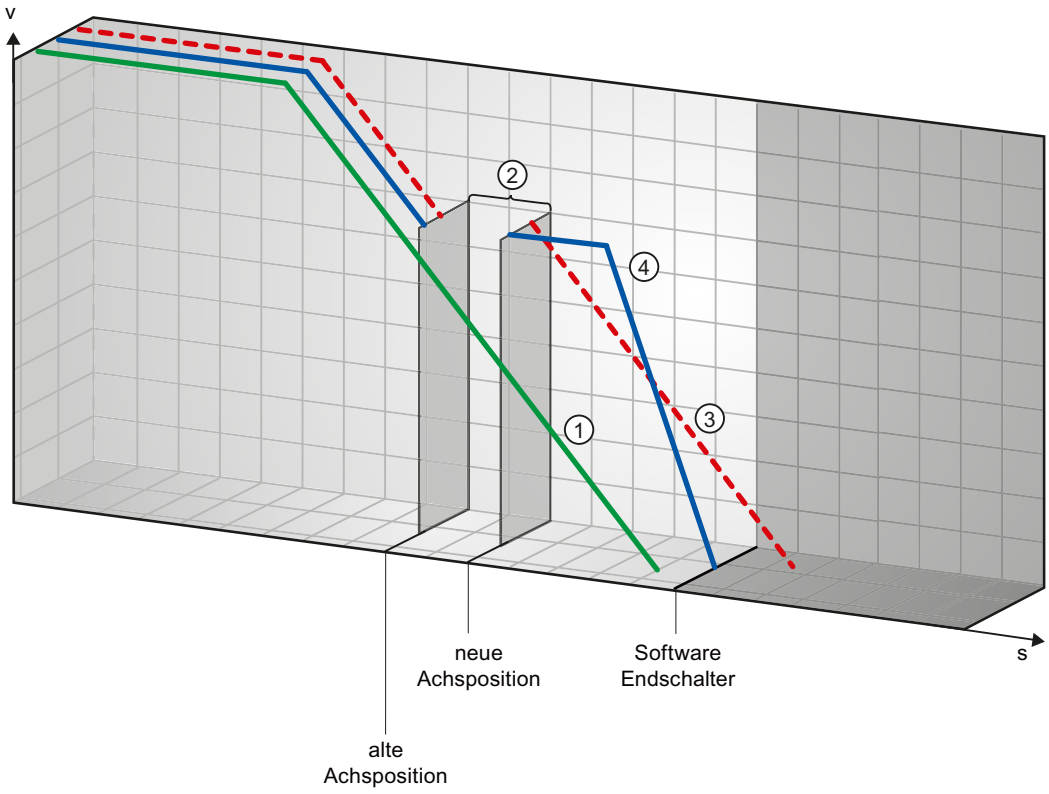

①	Der grüne Kurvenverlauf zeigt die Bewegung **ohne** Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".
③	Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
④	Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der konfigurierten Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen.

Beispiel 4:

Während eines Abbremsvorgangs wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Im Gegensatz zu Beispiel 3 ist es nicht mehr möglich, die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen. Die Achse fährt über die Position des Software-Endschalters hinaus.

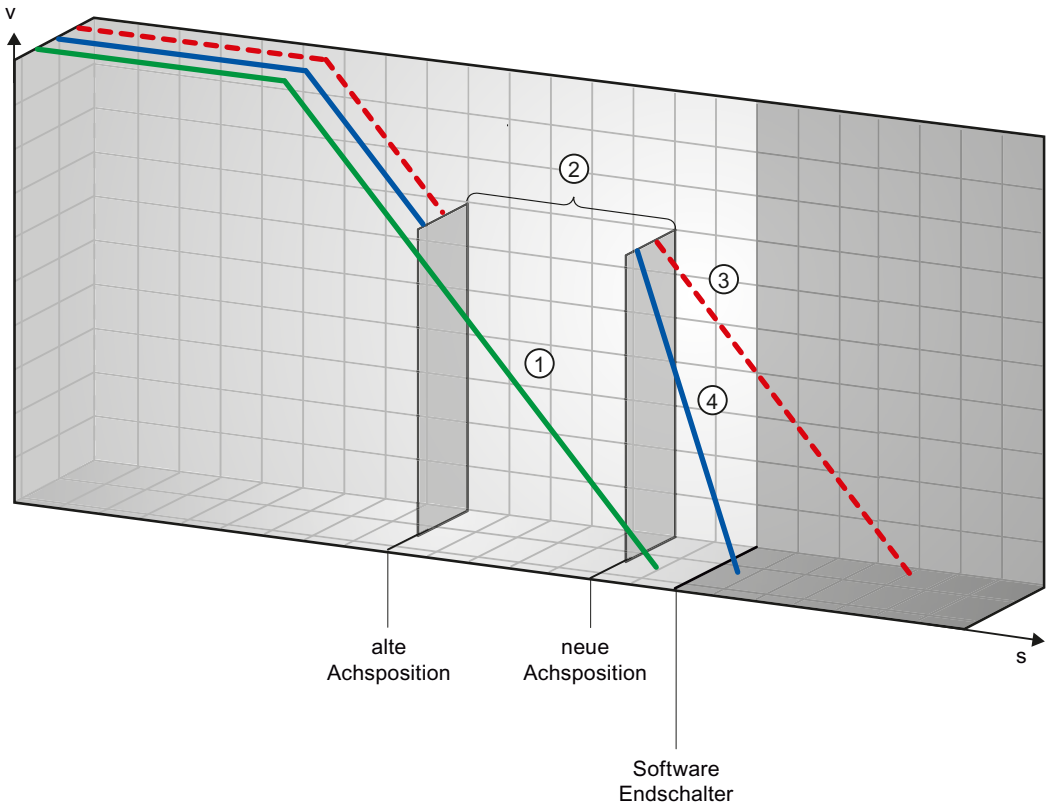

①	Der grüne Kurvenverlauf zeigt die Bewegung **ohne** Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".
③	Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
④	Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst mit der konfigurierten Notstopp-Verzögerung ab. Jedoch reicht die Notstopp-Verzögerung nicht aus, um die Achse auf der Position des Software-Endschalters zum Stehen zu bringen. Die Position des Software-Endschalters wird überfahren.

Siehe auch

Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter (Seite 1515)

Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 1515)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter

Durch eine ungünstige Änderung der Position des Software-Endschalters zur Laufzeit des Anwenderprogramms kann der Abstand zwischen aktueller Achsposition und der Position des Software-Endschalters abrupt verkürzt werden.

Die Reaktion der Achse erfolgt analog zur im Kapitel Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 1510) beschriebenen Reaktion.

Siehe auch

Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 1510)

Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 1515)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Software-Endschalter im Zusammenhang mit Dynamikänderungen

In Verbindung mit ablösenden Fahraufträgen kann der Verzögerungsvorgang der Achse im Bereich der Software-Endschalter beeinflusst werden. Dies gilt, wenn der ablösende Fahrauftrag mit einer geringeren Verzögerung (Variable <Achsname>.DynamicDefaults.Deceleration) gestartet wird. Berücksichtigen Sie die nachfolgenden Beispiele bei Ihrer Programmierung.

Beispiel 1:

Während des Verfahrens der Achse wird ein laufender Verfahrauftrag durch einen weiteren Verfahrauftrag mit einer geringeren Verzögerung abgelöst:

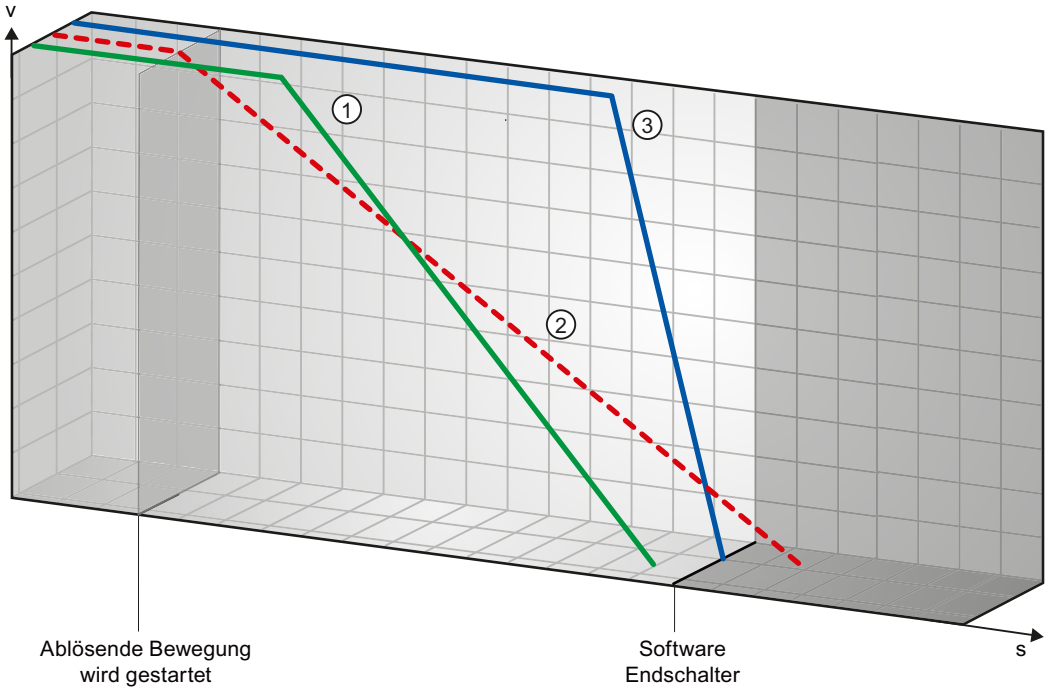

①	Der grüne Kurvenverlauf zeigt die Bewegung eines laufenden Auftrags, **ohne** dass dieser abgelöst wird. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Bedingt durch den ablösenden Bewegungsauftrag mit geringerer Verzögerung würde die Achse theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
③	Da ein Abbremsen mit der konfigurierten Verzögerung des ablösenden Bewegungsauftrags nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen.

Beispiel 2:

Während des Abbremsens der Achse wird ein laufender Verfahrauftrag durch einen weiteren Verfahrauftrag mit einer geringeren Verzögerung abgelöst:

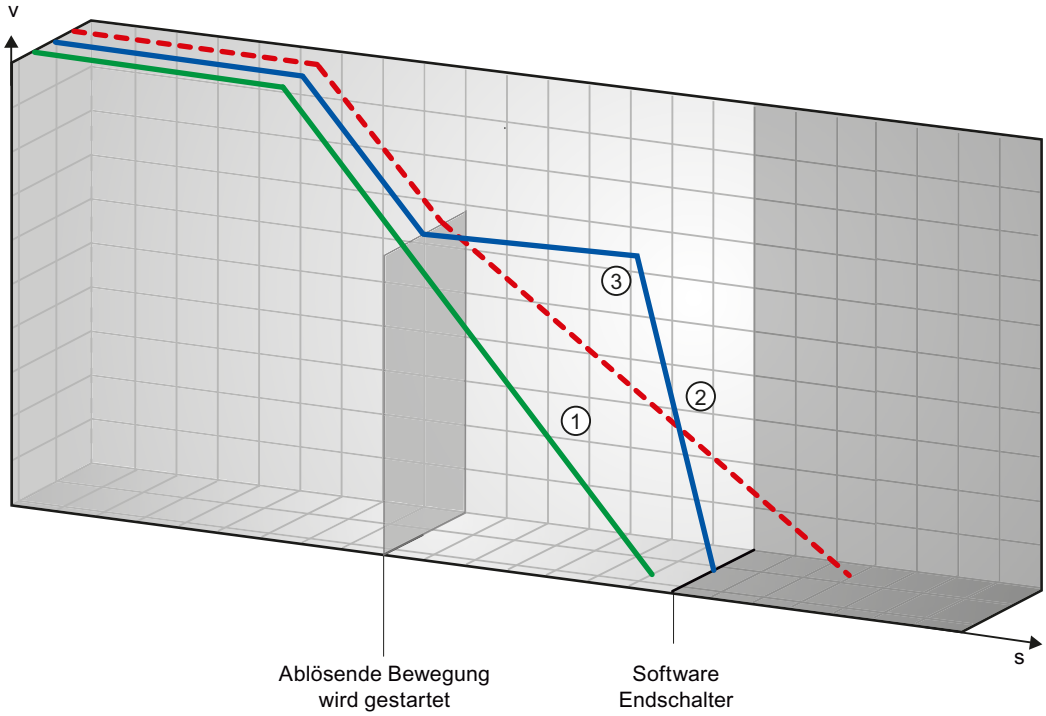

①	Der grüne Kurvenverlauf zeigt die Bewegung eines laufenden Auftrags, **ohne** dass dieser abgelöst wird. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.
②	Bedingt durch den ablösenden Bewegungsauftrag mit geringerer Verzögerung würde die Achse theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).
③	Da ein Abbremsen mit der konfigurierten Verzögerung des ablösenden Bewegungsauftrags nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen.

Siehe auch

Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 1510)

Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter (Seite 1515)

Verhalten der Achse beim Ansprechen der Positionsgrenzen (Seite 1405)

Reduzierung der Geschwindigkeit bei kurzer Positionierdauer

Bei einer geplanten Positionierdauer < 2 ms kann die Geschwindigkeit des Positionierauftrags durch die CPU reduziert werden.

Der Auftrag wird über den gesamten Verlauf mit einer reduzierten Geschwindigkeit ausgeführt. Die reduzierte Geschwindigkeit (Impulse/s) errechnet sich aus der nachfolgenden Formel:

- Reduzierte Geschwindigkeit = Anzahl auszugebender Impulse * 500Hz

Bei einer geplanten Positionierdauer >= 2 ms erfolgt **keine** Reduzierung der Geschwindigkeit.

Dynamische Anpassung der Start-/Stopp-Geschwindigkeit

Aufgrund der Konfiguration Ihrer Geschwindigkeitsgrenzen (Start/Stopp-Geschwindigkeit, Maximale Geschwindigkeit), der Dynamikwerte (Beschleunigung, Verzögerung, Ruck), sowie der Zielgeschwindigkeit des Verfahrauftrags kann die Start/Stopp-Geschwindigkeit unter bestimmten Umständen von der CPU dynamisch angepasst werden.

Dies ist z. B. der Fall, wenn aufgrund einer konfigurierten geringen Start/Stopp-Geschwindigkeit die benötigte Zeit für die ersten Impulse länger wäre, als für die gesamte Beschleunigung benötigt werden darf. In diesen Fällen wird der erste Impuls mit einer größeren Geschwindigkeit als der konfigurierten Start/Stopp-Geschwindigkeit ausgegeben. Die nachfolgenden Impulse werden ebenfalls dynamisch angepasst, so dass der Beschleunigungsvorgang in der vorgegebenen Zeit beendet werden kann.

Achten Sie bei eventuellen Impulsverlust darauf, dass Ihre verwendete Hardware (Antrieb) auf diese Situation abgestimmt ist, bzw. verändern Sie die Dynamikeinstellungen Ihrer Achse, um die dynamische Anpassung der Start/Stopp-Geschwindigkeit zu vermeiden.

Achse im Servicefall ohne Lageregelung bewegen

Im Servicefall kann es nötig sein, einen PROFIdrive-Antrieb oder einen Antrieb mit analoger Antriebsschnittstelle nicht lagegeregelt zu bewegen.

Dies kann z. B. bei nicht gültigen oder falschen Geberwerten der Fall sein.

Folgende Motion Control Anweisungen beinflussen den Zustand der Lageregelung:

Lagerregelte Antriebe nicht lagegeregelt freigeben mit MC_Power

Starten Sie die Achse mit der Motion Control Anweisung MC_Power StartMode = 0 im nicht lageregelten Betrieb.

Der nicht lagegereglete Betrieb gilt, bis eine andere Motion Control Anweisung den Zustand der Lageregelung ändert.

MC_MoveVelocity

MC_MoveVelocity mit PositionControlled = FALSE erzwingt den nicht lagegeregelten Betrieb.

MC_MoveVelocity mit PositionControlled = TRUE erzwingt den lagegeregelten Betrieb.

Der gewählte lagegeregelte Betrieb bleibt auch nach Beendigung des MC_MoveVelocity erhalten.

MC_MoveJog

MC_MoveJog mit PositionControlled = FALSE erzwingt den nicht lagegeregelten Betrieb.

MC_MoveJog mit PositionControlled = TRUE erzwingt den lagegeregelten Betrieb.

Der gewählte lagegeregelte Betrieb bleibt auch nach Beendigung des MC_MoveJog erhalten.

MC_Home, MC_MoveRelative, MC_MoveAbsolute

Die Motion Control Anweisungen MC_Home, MC_MoveRelative und MC_MoveAbsolute erzwingen den lagegeregelten Betrieb.

Die Lageregelung bleibt auch nach Beendigung des Auftrags aktiv.

MC_Halt

Die Motion Control Anweisung MC_Halt wird im lagegeregleten, als auch im nicht lagegeregelten Betrieb ausgeführt.

Der Zustand der Lageregelung wird nicht geändert.

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V6)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

Bei Betriebsfehlern mit Stopp der Achse wird die Achse abhängig von der Fehlerreaktion gestoppt. Folgende Fehlerreaktionen sind möglich:

- **Freigabe wegnehmen**
  Der Sollwert null wird ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
- **Stopp mit Notstopp-Rampe**
  Laufende Bewegungsbefehle werden abgebrochen. Die Achse wird mit der unter "Technologieobjekt > Erweiterte Parameter > Dynamik > Notstopp" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht.

Betriebsfehler mit Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8000		**Antriebsfehler, "Antrieb bereit" ausgefallen**		-
	16#0001	-	Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8001		Unterer SW-Endschalter wurde ausgelöst		-
	16#000E	Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren	
	16#000F	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht		
	16#0010	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten		Freigabe wegnehmen
16#8002		Oberer SW-Endschalter wurde ausgelöst		-
	16#000E	Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren	
	16#000F 0068	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht		
	16#0010	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten		Freigabe wegnehmen
16#8003		Unterer HW-Endschalter wurde angefahren		Bei Antriebsanbindung über PTO (Pulse Train Output): Stopp mit Notstopp-Rampe Bei Antriebsanbindung über PROFIdrive / Analogausgang: Freigabe wegnehmen
	16#000E	Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.	
16#8004		Oberer HW-Endschalter wurde angefahren		Bei Antriebsanbindung über PTO (Pulse Train Output): Stopp mit Notstopp-Rampe Bei Antriebsanbindung über PROFIdrive / Analogausgang: Freigabe wegnehmen
	16#000E	Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8005		**PTO / HSC werden bereits von einer anderen Achse verwendet**		-
	16#0001	-	**Die Achse wurde fehlerhaft konfiguriert:** Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden	
			Mehrere Achsen sollen mit einem PTO arbeiten: Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden. (siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504))	
16#8006		**Es ist ein Kommunikationsfehler in der Achssteuertafel eingetreten**		Freigabe wegnehmen
	16#0012	Es ist ein Zeitüberlauf eingetreten	Prüfen Sie die Kabelverbindung; betätigen Sie erneut die Schaltfläche "Handsteuerung"	
16#8007		**Freigabe der Achse nicht möglich**		-
	16#0025	Restart wird ausgeführt	Warten Sie, bis der Restart der Achse abgeschlossen ist.	
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie, bis der Ladevorgang abgeschlossen ist.	
16#8008		**Unzulässige Bewegungsrichtung**		-
	16#002E	Die gewählte Bewegungsrichtung ist nicht erlaubt.	• Passen Sie die Bewegungsrichtung an und starten Sie den Auftrag erneut. • Passen Sie in der Konfiguration des Technologieobjekts unter "Erweiterte Parameter > Mechanik" die erlaubte Drehrichtung an. Starten Sie den Auftrag erneut.	
	16#002F	Eine reversierende Bewegung ist mit der gewählten Bewegungsrichtung nicht möglich.		
16#8009		**Referenzschalter/Gebernullmarke nicht gefunden**		Stopp mit Notstopp-Rampe
	16#0033	Fehler in der Konfiguration, Hardware oder Montage des Gebers oder am Referenzpunktschalter.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#800A		Störungsmeldung vom Geber		Freigabe wegnehmen
	16#0001	-	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse.	
	16#0034	Hardware-Fehler am Geber		
	16#0035	Geber verschmutzt		
	16#0036	Fehler beim Lesen des Geberabsolutwerts	Vergleichen Sie den Gebertyp im Antriebs- bzw. Geberparameter P979 mit den Konfigurationsdaten des Technologieobjekts.	
	16#0037	Nullmarkenüberwachung des Gebers	Geber meldet Fehler bei der Nullmarkenüberwachung (Störcode 0x0002 im Gx_XIST2, siehe PROFIdrive-Profil). Prüfen Sie die Anlage auf elektromagnetische Verträglichkeit (EMV).	
	16#0038	Geber ist im Zustand "Parken"	• Suchen Sie die Fehlerursache im angeschlossenen Antrieb bzw. Geber. • Prüfen Sie, ob die Fehlermeldung eventuell infolge einer Inbetriebnahmehandlung am Antrieb bzw. Geber ausgelöst wurde.	
	16#0040	PROFIdrive: Geber am Bus ausgefallen (Stationsausfall)	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse.	
	16#0041	PROFIdrive: Lebenszeichen des Gebers gestört		
16#800B		Bereichsüberschreitung der Position		Freigabe wegnehmen
	16#0039	Bereichsüberschreitung in positiver Richtung	Referenzieren Sie die Achse in einen gültigen Istwertbereich.	
	16#003A	Bereichsüberschreitung in negativer Richtung		
	16#003B	Die Veränderung der Istposition ist in einem Lageregler-Takt grösser als die Modulolänge.	Passen Sie die Modulolänge dem verwendeten Geber an.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#800C		Störungsmeldung vom Antrieb		Freigabe wegnehmen
	16#0001	-	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse. Wählen Sie im Dialog "Optimierung" eine kleinere Verstärkung (Kv).	
	16#003C	PROFIdrive: Antriebssignal "Führung gefordert" ausgefallen		
	16#003D	PROFIdrive/Analoge Antriebsanbindung: Antrieb hat sich abgeschaltet		
	16#003E	PROFIdrive: Antrieb am Bus ausgefallen (Stationsausfall)		
	16#003F	PROFIdrive: Lebenszeichen des Antriebs gestört	• Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse. • Vergleichen Sie die Taktparameter von HW Konfig (PROFIBUS-Strang, Slave-OM für Antrieb bzw.Geber) und des Ablaufsystems. Tmapc und Servo müssen auf die gleiche Taktperiodenzeit parametriert sein.	
16#800D		Der zulässige Schleppfehler wurde überschritten		Freigabe wegnehmen
	16#0001	-	• Prüfen Sie die Konfiguration des Regelkreises. • Prüfen Sie den Richtungssinn des Gebers. • Prüfen Sie Konfiguration der Schleppfehlerüberwachung.	
16#800E		Fehler am HW-Endschalter		Freigabe wegnehmen
	16#0042	Unzulässige Freifahrrichtung bei aktivem HW-Endschalter	Die programmierte Bewegungsrichtung ist aufgrund des aktiven HW-Endschalters gesperrt. Fahren Sie die Achse in entgegengesetzter Richtung frei.	
	16#0043	HW-Endlagenschalter verpolt, kein Freifahren möglich	Prüfen Sie den mechanischen Aufbau der HW-Endschalter.	
	16#0044	Beide HW-Endschalter sind aktiv, kein Freifahren möglich		
16#800F		Fehler am Zielbereich		Freigabe wegnehmen
	16#0045	Zielbereich nicht erreicht	Der Zielbereich wurde nicht innerhalb der Positioniertoleranzzeit erreicht. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.	
	16#0046	Zielbereich wieder verlassen	Der Zielbereich wurde innerhalb der Mindestverweildauer verlassen. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8010		Position des unteren SW-Endschalters größer als die Position des oberen SW-Endschalters, wenn die Achse keine Moduloachse ist		Freigabe wegnehmen
	16#0001	-	Ändern Sie die Position der SW-Endschalter.	
16#8011		Anfahrgeschwindigkeit zum Referenzpunktschalter/Nullmarke ist gleich null.		Freigabe wegnehmen
	16#000A	Wert ist kleiner oder gleich 0.	Anfahrgeschwindigkeit > null wählen	
16#8012		Referenziergeschwindigkeit zum Setzen des Referenzpunkts ist gleich null.		Freigabe wegnehmen
	16#000A	Wert ist kleiner oder gleich 0.	Referenziergeschwindigkeit > null wählen	
16#8013		Die Achse kann den PTO nicht belegen, da dieser von "CTRL_PTO" verwendet wird.		Freigabe wegnehmen
	16#0001	-	Anderen PTO in der Konfiguration wählen.	

Betriebsfehler ohne Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8200		Achse ist nicht freigegeben	
	16#0001	-	Achse freigeben; Auftrag erneut starten
16#8201		Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben	
	16#0001	-	Achse nur über eine "MC_Power"-Instanz freigeben
16#8202		Maximale Anzahl gleichzeitig aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für Antriebsanbindung über PTO (Pulse Train Output), max. 100 Aufträge für Antriebsanbindung über PROFIdrive/Analogausgang)	
	16#0001	-	Anzahl der gleichzeitig aktiven Aufträge verringern; Auftrag erneut starten Ein aktiver Auftrag ist am Parameter "Busy" = TRUE der Motion Control-Anweisung zu erkennen.
16#8203		Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben	
	16#0001	-	"Handsteuerung" beenden; Auftrag erneut starten
16#8204		Achse ist nicht referenziert	
	16#0001	-	Achse mit der Anweisung "MC_Home" referenzieren; Auftrag erneut starten
16#8205		Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der Achssteuertafel angezeigt)	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren und in der Achssteuertafel erneut "Handsteuerung" anwählen
16#8206		Technologieobjekt ist noch nicht aktiviert	
	16#0001	-	Aktivieren Sie die Achse mit der Anweisung "MC_Power" Enable = TRUE, oder geben Sie die Achse in der Achssteuertafel frei.
16#8207		Auftrag abgewiesen	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0001	-	
	16#0016	Aktives Referenzieren läuft; eine andere Referenzierart kann nicht gestartet werden.	Warten Sie den Abschluss des aktiven Referenzierens ab oder brechen Sie das aktive Referenzieren durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
	16#0018	Die Achse kann nicht mit einer Auftragstabelle verfahren werden, während die Achse direkt oder passiv referenziert wird.	Warten Sie den Abschluss des direkten oder passiven Referenzierens ab.
	16#0019	Während der Abarbeitung einer Auftragstabelle, kann die Achse nicht direkt oder passiv referenziert werden.	Warten Sie den Abschluss der Auftragstabelle ab oder brechen Sie die Auftragstabelle durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
	16#0052	Die angegebene Position überschreitet die nummerische Zahlengrenze.	Geben Sie an der Motion Control-Anweisung einen gültigen Positionswert an.
	16#0053	Die Achse befindet sich im Hochlauf.	Warten Sie, bis die Achse betriebsbereit ist.
	16#0054	Istwert nicht gültig	Um einen "MC_Home"-Auftrag auszuführen, müssen die Istwerte gültig sein. Prüfen Sie den Status der Istwerte. Die Variable des Technologieobjekts <Achsname>.StatusSensor.State muss den Wert 2 (gültig) zeigen.
	16#0058	Auftrag wird bereits in einer anderen Ablaufebene verwendet.	Achse nur über eine "MC_Power"-Instanz aufrufen.
	16#006B	Aufruf im nicht lagegeregelten Betrieb unzulässig.	Geben Sie die Achse mit dem "MC_Power" mit StartMode = 1 lagegeregelt frei.
16#8208		**Geschwindigkeitsdifferenz zwischen maximaler Geschwindigkeit und Start/Stopp-Geschwindigkeit ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#8209		**Wert der Beschleunigung des Technologieobjekts "Achse" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#820A		**Restart der Achse nicht möglich**	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren; Restart erneut ausführen
	16#0027	Die Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben	"Handsteuerung" beenden; Restart erneut ausführen
	16#002C	Die Achse ist nicht gesperrt.	Achse sperren; Auftrag erneut starten
	16#0047	Das Technologieobjekt ist für den Restart nicht bereit.	Laden Sie das Projekt neu.
	16#0048	Bedingung für den Restart des Technologieobjekts ist nicht erfüllt.	Sperren Sie das Technologieobjekt.
16#820B		**Ausführen der Auftragstabelle nicht möglich**	
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie, bis der Ladevorgang abgeschlossen ist.
16#820C		**Keine Konfiguration verfügbar**	
	16#0001	-	Interner Fehler Wenden Sie sich an die Hotline.
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	

Bausteinparameterfehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8400		**Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	
16#8401		**Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
16#8402		**Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
	16#0024	Wert ist kleiner 0	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	
16#8403		**Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
16#8404		**Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
	16#0015	Aktives/Passives Referenzieren ist nicht konfiguriert	Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0017	Die Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind	• Aktivieren der HW-Endschalter über die Variable <Achsname>.PositionLimits_HW.Active = TRUE, Auftrag erneut starten • Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0055	Ungültiger Mode bei Inkrementalwertgeber	Starten Sie einen Referenziervorgang für einen Inkrementalgeber mit Parameter "Mode" = 0, 1, 2, 3.
	16#0056	Ungültiger Mode bei Absolutwertgeber	Bei einem Absolutwertgeber ist passives und aktives Referenzieren ("Mode" = 2, 3) nicht möglich. Starten Sie einen Referenziervorgang für einen Absolutwertgeber mit Parameter "Mode" = 0, 1.
16#8405		**Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Achse erneut freigeben
16#8406		**Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0001	-	Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.
16#8407		**Wechsel der Achse an der Anweisung "MC_Power" nur bei gesperrter Achse zulässig.**	
	16#0001	-	Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.
16#8408		**Wert am Parameter "Axis" der Motion Control-Anweisung ist ungültig**	
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#8409		**Wert am Parameter "CommandTable" der Motion Control-Anweisung ist ungültig**	
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#840A		**Wert am Parameter "StartStep" der Motion Control-Anweisung ist ungültig**	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001D	Der Startschritt ist größer als der Endschritt	
	16#001E	Wert ist größer 32	
16#840B		**Wert am Parameter "EndStep" der Motion Control-Anweisung ist ungültig**	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001E	Wert ist größer 32	
16#840C		**Wert am Parameter "RampUpTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840D		**Wert am Parameter "RampDownTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840E		**Wert am Parameter "EmergencyRampTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840F		**Wert am Parameter "JerkTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#8410		**Wert am Parameter "Parameter" der Motion Control-Anweisung ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000B	Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0028	Datentyp des VARIANT-Zeigers "Parameter" und "Value" passen nicht zueinander.	Geigneten Datentyp verwenden; Auftrag erneut starten
	16#0029	VARIANT-Zeiger "Parameter" zeigt nicht auf einen Datenbaustein des Technologieobjekts.	VARIANT-Zeiger korrigieren; Auftrag erneut starten
	16#002A	Der Wert am VARIANT-Zeiger "Parameter" kann nicht gelesen werden.	VARIANT-Zeiger korrigieren; Auftrag erneut starten
	16#002B	Der Wert am VARIANT-Zeiger "Parameter" kann nicht geschrieben werden.	VARIANT-Zeiger oder Wert korrigieren; Auftrag erneut starten
	16#002C	Die Achse ist nicht gesperrt.	Achse sperren; Auftrag erneut starten
16#8411		**Wert am Parameter "Value" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
16#8412		**Wert am Parameter "StartMode" der Motion Control Anweisung ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Achse erneut freigeben

Konfigurationsfehler der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8600		**Parametrierung des Impulsgenerators (PTO) ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8601		**Parametrierung des schnellen Zählers (HSC) ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8602		**Parametrierung des "Freigabe-Ausgangs" ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8603		**Parametrierung des "Bereit-Eingangs" ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8604		**Wert für "Impulse pro Motorumdrehung" ist ungültig**	
	16#000A	Wert ist kleiner oder gleich null	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8605		**Wert für "Weg pro Motorumdrehung" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#000A	Wert ist kleiner oder gleich null	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8606		**Wert für "Start/Stopp-Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0007	Die Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit	
16#8607		**Wert für "maximale Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8608		**Wert für "Beschleunigung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8609		**Wert für "Verzögerung" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#860A		Wert für "Notstopp-Verzögerung" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#860B		Wert für die Position des unteren SW-Endschalters ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
	16#0030	Der Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters	
16#860C		Wert für die Position des oberen SW-Endschalters ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#860D		Adresse des unteren HW-Endschalters ist ungültig	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860E		**Adresse des oberen HW-Endschalter ist ungültig**	
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860F		**Wert für "Referenzpunktverschiebung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8610		**Wert für "Anfahrgeschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8611		**Wert für "Referenziergeschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8612		**Adresse des Referenzpunktschalters ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#8613		**Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind**	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8614		**Wert für "Ruck" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#001F	Wert ist größer als der maximal zulässige Ruck	
	16#0020	Wert ist kleiner als der minimal zulässige Ruck	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8615		**Wert für "Maßeinheit" ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8616		**Adresse des Referenzpunktschalters ist ungültig (passives Referenzieren ab V4)**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8617		**Wert der Variable <Achsname>.Sensor.Sensor[1].ActiveHoming.Mode ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Referenzieren über Digitaleingang)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8618		**Wert der Variable <Achsname>.Sensor.Sensor[1].PassiveHoming.Mode ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Referenzieren über Digitaleingang)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8619		Wert der Variable <Achsname>.Actor.Type ist ungültig	
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Anbindung über Impulsschnittstelle)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861A		Wert für "Erlaubte Drehrichtung" ist ungültig	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#002D	"Beide Richtungen" bei abgeschalteten Richtungsausgang nicht erlaubt	
16#861B		Lastgetriebefaktoren fehlerhaft	
	16#0031	Wert ist ungültig.	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#861C		Unzulässige Kombination der Daten für Referenzieren beim inkrementellen Geber	
	16#0031	Wert ist ungültig.	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861D		Die eingestellte Gebersanbauart ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].MountingMode	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861E		Die Konfiguration des Messradumfangs des Gebers ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.DistancePerRevolution	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861F		Die Konfiguration der Auflösung des Lineargebers ist fehlerhaft. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.Resolution	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8620		**Die eingestellte Feinauflösung für Gn_XIST1 ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist1**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8621		**Die eingestellte Feinauflösung für Gn_XIST1 in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist1 ist nicht konsistent zu der Einstellung im PROFIdrive-Parameter P979**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8622		**Wert in <Achsname>.Actor.Interface.AddressIn bzw. <Achsname>.Actor.Interface.AddressOut unzulässig**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8623		**Der eingestellte Wert in der Variable <Achsname>.Sensor.Sensor[1].Type ist ungültig.**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8624		**Das eingestellte Gebersystem ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].System**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8625		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.MinDwellTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8626		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.Window**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8627		**Die Konfiguration der PROFIdrive-Schnittstelle des Istwerts ist fehlerhaft. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Interface.AddressIn bzw. <Achsname>.Sensor.Sensor[1].Interface.AddressOut**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8628		**Reglerfaktoren fehlerhaft**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Der Wert für die Verstärkung bzw. die Vorsteuerung des Regelkreises ist fehlerhaft. • Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten (<Achsname>.PositionControl.Kv, <Achsname>.PositionControl.Kpc)
16#8629		**Grenzwert für Stillstandssignal ist fehlerhaft. Unzulässiger Wert in <Achsname>.StandStillSignal.VelocityThreshold**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862A		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.ToleranceTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862B		**Inkonsistente PROFIBUS-Parametrierung; die Summe aus Ti und To ist grösser als ein DP-Takt**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#862C		**Parameter der Stillstandsüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.StandStillSignal.MinDwellTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862D		**Parameter der Schleppabstandsüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MinValue**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862E		Wert in <Achsname>.Modulo.Length unzulässig	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862F		Wert in <Achsname>.Modulo.StartValue unzulässig	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8630		Wert in <Achsname>.Actor.DriveParameter.ReferenceSpeed unzulässig	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8631		Die eingestellte Feinauflösung für Gn_XIST2 ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist2	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8632		Die Anzahl der auflösbaren Geberumdrehungen ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.DeterminableRevolutions	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8634		Parameter der Schleppabstandsüberwachung fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MaxValue	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8635		Parameter der Schleppabstandsüberwachung fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MinVelocity	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8636		Reglerfaktor fehlerhaft. Unzulässiger Wert des Vorsteuerfaktors <Achsname>.PositionControl.Kpc	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8637		Wert in <Achsname>.Sensor.Sensor[1].Interface.Type unzulässig	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8638		Wert in <Achsname>.Sensor.Sensor[1].Interface.HSC unzulässig	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8639		Fehler am Antrieb	
	16#0049	Konfigurationsfehler am Gerät	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#004A	Die Technologie benötigt einen kleineren Servo-Takt	Interner Systemfehler. Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
	16#004B	Gerätetreiber wurde im Hochlauf nicht initialisiert	Um ein Technologieobjekt freizugeben, muss der Aktortreiber fertig initialisiert sein. Setzen sie den Auftrag später erneut ab.
16#863A		Kommunikation zum Antrieb fehlerhaft	
	16#004C	Konfigurationsfehler am Gerät	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004D	Gerätereiber benötigt einen kleineren Servo-Takt	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004E	Fehler in der internen Kommunikation zum Gerät	Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
16#863B		Fehler am Geber	
	16#0049	Konfigurationsfehler am Gerät	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#004A	Die Technologie benötigt einen kleineren Servo-Takt	Interner Systemfehler. Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
	16#004B	Gerätetreiber wurde im Hochlauf nicht initialisiert	Um ein Technologieobjekt freizugeben, muss der Aktortreiber fertig initialisiert sein. Setzen sie den Auftrag später erneut ab.

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#863C		**Kommunikation zum Geber fehlerhaft**	
	16#004C	Konfigurationsfehler am Gerät	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004D	Gerätereiber benötigt einen kleineren Servo-Takt	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004E	Fehler in der internen Kommunikation zum Gerät	Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
16#863D		**Kommunikation zum Gerät (Antrieb oder Geber) fehlerhaft**	
	16#004F	Die angeforderte logische Adresse ist ungültig.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie die topologische Konfiguration in HW Konfig. • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#0050	Die angeforderte logische Input-Adresse ist ungültig.	
	16#0051	Die angeforderte logische Output-Adresse ist ungültig.	
16#863E		**Wert der Variable "ControlPanel.Input.TimeOut" ist ungültig (Achssteuertafel)**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Wert in der Variablen des Technologieobjekts <Achsname>.ControlPanel.Input.TimeOut. Der Wert wird in Millisekunden (ms) angegeben.
16#863F		**Wert in <Achsname>.Actor.DriveParameter.MaxSpeed unzuläsig**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf Actor.MaxSpeed/2. Korrigieren Sie bei analoger Antriebsanbindung den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf Actor.MaxSpeed/1,17.
16#8640		**Fehler bei der automatischen Übernahme der Abtriebsparameter im Gerät**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Wert.
	16#0059	Das Gerät ist keinem SINAMICS Antriebsgerät zugeordnet oder unterstützt die für die Adaption notwendigen Dienste nicht.	Deaktivieren Sie die automatische Übernahme der Parameter im Gerät. Ergänzen Sie in der Achskonfiguration die geforderten Parameter und laden Sie die Achskonfiguration erneut ins Gerät.
	16#005A	Die automatische Übernahme der Parameter wird wegen Resourcenmangel abgebrochen.	
	16#005B	Die automatische Übernahme der Parameter ist nur möglich, wenn das Gerät direkt auf einen I/O Bereich verschaltet wurde.	
	16#005C	Maximaldrehzahl/-geschwindigkeit (p1082): Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	Überprüfen Sie die Ursachen. Deaktivieren Sie die automatische Übernahme der Parameter im Gerät, falls die Ursachen nicht beseitigt werden können. Ergänzen Sie in der Achskonfiguration die geforderten Parameter und laden Sie die Achskonfiguration erneut ins Gerät.
	16#005D	Maximalmoment/-kraft (p1520): Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#005E	Maximalmoment/-kraft (p1521) : Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#005F	Feinauflösung Momenten-/Kraftbegrenzung (p1544): Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#0060	Nenndrehzahl/-geschwindigkeit (p2000): Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0061	Nennmoment/-kraft (p2003) : Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
16#8641		Fehler bei der automatischen Übernahme der Geberparameter im Gerät	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Wert.
	16#0059	Das Gerät ist keinem SINAMICS Antriebsgerät zugeordnet oder unterstützt die für die Adaption notwendigen Dienste nicht.	Deaktivieren Sie die automatische Übernahme der Parameter im Gerät. Ergänzen Sie in der Achskonfiguration die geforderten Parameter und laden Sie die Achskonfiguration erneut ins Gerät.
	16#005A	Die automatische Übernahme der Parameter wird wegen Resourcenmangel abgebrochen.	
	16#005B	Die automatische Übernahme der Parameter ist nur möglich, wenn das Gerät direkt auf einen I/O Bereich verschaltet wurde.	
	16#0062	Gebersystem (r0979[1/11].0): Ein Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	Überprüfen Sie die Ursachen. Deaktivieren Sie die automatische Übernahme der Parameter im Gerät, falls die Ursachen nicht beseitigt werden können. Ergänzen Sie in der Achskonfiguration die geforderten Parameter und laden Sie die Achskonfiguration erneut ins Gerät.
	16#0063	Geberauflösung (r0979[2/12]): Ein Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#0064	Geberfeinauflösung Gx_XIST1 (r0979[3/13]): Ein Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#0065	Geberfeinauflösung Gx_XIST2 (r0979[4/14]): Ein Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
	16#0066	Anzahl der auflösbaren Geberumdrehungen (r0979[5/15]): Ein Parameter existiert nicht oder dessen Wert kann entweder nicht gelesen werden oder liegt ausserhalb der erlaubten Grenzen. Das Lesen der Parameter wurde wegen eines von der Hardware gemeldeten Fehlers abgebrochen.	
16#8646		Wert in <Achsname>.Sensor.Interface.Number unzulässig	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs.	Wert korrigieren und Konfiguration in Gerät laden.
16#8647		**Simulation wird bei PTO-Achsen nicht unterstützt**	
	16#0001	-	Simulation deaktvieren

Konfigurationsfehler der Auftragstabelle

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8700		**Wert für "Auftragstyp" in der Auftragstabelle ist ungültig**	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
16#8701		**Wert für "Position / Verfahrweg" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E12)	
16#8702		**Wert für "Geschwindigkeit" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
16#8703		**Wert für "Dauer" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0021	Wert ist größer als 64800 s	
	16#0022	Wert ist kleiner als 0.001 s	
16#8704		**Wert für "Nächster Schritt" in der Auftragstabelle ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0023	Der Auftragsübergang ist für diesen Auftrag nicht erlaubt	

Interne Fehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8FFF		Interner Fehler	
	16#F0**	-	NETZ-AUS und NETZ-EIN der CPU Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit: • ErrorID • ErrorInfo • Diagnosepuffereinträge

Variablen des Technologieobjekts Positionierachse ab V6

Legende

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Startwert (Wertebereich der Variable - Minimalwert bis Maximalwert) Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
Zugriff	Zugriff auf die Variable im Anwenderprogramm	
	OPR	Die Variable kann von der Openness-Anwendung gelesen werden.
	OPRW	Die Variable kann von der Openness-Anwendung gelesen und geschrieben werden.
	R	Die Variable kann im Anwenderprogramm und im HMI gelesen werden.
	RCCP	Die Variable kann im Anwenderprogramm und im HMI gelesen werden und wird jeweils am Zykluskontrollpunkt aktualisiert.
	RP	Die Variable kann mit der Motion Control-Anweisung "MC_ReadParam" gelesen werden. Der aktuelle Wert der entsprechenden Variablen wird beim Start des Auftrags ermittelt.
	RW	Die Variable kann im Anwenderprogramm und im HMI gelesen und geschrieben werden. Die Variable kann mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WD BL	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Die Variable kann mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert im Ladespeicher geschrieben werden.
	WP	Unabhängig von der Antriebsanbindung: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WP_PD	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WP_PTO	Bei Antriebsanbindung über PTO: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	-	Die Variable kann im Anwenderprogramm nicht verwendet werden.

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	1	Bei Antriebsanbindung über PTO: Mit dem Aktivieren, Sperren oder Freigeben der Achse
	2	Bei Antriebsanbindung über PTO: Mit dem Freigeben der Achse
	5	Bei Antriebsanbindung über PTO: Mit dem nächsten Start eines MC_MoveAbsolute-, MC_MoveRelative-, MC_MoveVelocity-, MC_MoveJog-, MC_Halt-, MC_CommandTable- oder aktiven MC_Home-Auftrags (Mode = 3)
	6	Bei Antriebsanbindung über PTO: Mit dem Stoppen eines MC_MoveJog-Auftrags
	7	Bei Antriebsanbindung über PTO: Mit dem Start eines passiven Referenzierauftrags
	8	Bei Antriebsanbindung über PTO: Mit dem Start eines aktiven Referenzierauftrags
	9	Mit dem Restart des Technologieobjekts
	10	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Mit dem nächsten Aufruf des MC-Servo [OB91]
Beschreibung	Beschreibung der Variablen	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Variablen zu Positions- und Geschwindigkeitswerten ab V6

Die Variablenstruktur beinhaltet die Soll- und Istwerte der Position und der Geschwindigkeit der Achse.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Position	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Sollposition der Achse (Angabe in der konfigurierten Maßeinheit) Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabeposition der Achse. Name in Openness: Position
Velocity	REAL	0.0	RCCP, RP, OPR	-	Sollgeschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: Velocity
ActualPosition	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Istposition der Achse (Angabe in der konfigurierten Maßeinheit) Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabeposition der Achse. Name in Openness: ActualPosition
ActualVelocity	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Istgeschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: ActualVelocity

Siehe auch

Bewegungsstatus (Seite 1502)
Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)
Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variable Simulation ab V6

Die Variablenstruktur <Achsname>.Simulation.Mode beinhaltet den Simulationsbetrieb.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Simulation.						
Mode	UDINT	0 (0 bis 1)	R WDBL OPRW	- 2, 9	Simulationsbetrieb Name in Openness: Simulation.Mode	
					0	Keine Simulation, normaler Betrieb
					1	Simulationsbetrieb Im Simulationsbetrieb können Sie Achsen ohne reellen Antrieb in der CPU simulieren.

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)
Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Actor ab V6

Die Variablenstruktur <Achsname>.Actor.<Variablenname> beinhaltet die Parameter des Antriebs.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Actor.	STRUCT				TO_Struct_Actor

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Type	DINT	2 (0 bis 2)	R OPRW	-	Name in Openness: Actor.Type	
					Technologieobjekt Positionierachse ab V5:	
					0	Der Antrieb wird über einen Analogausgang angebunden. Alle Bewegungen der Achse werden lagegeregelt.
					1	Der Antrieb wird über PROFIdrive-Telegramme angebunden. Alle Bewegungen der Achse werden lagegeregelt.
					2	Der Antrieb wird über eine Impulsschnittstelle angebunden.
					Technologieobjekt Positionierachse V4:	
					2	Der Antrieb wird über eine Impulsschnittstelle angebunden.
InverseDirection	BOOL	FALSE	R WP_PTO OPRW	- 2	Name in Openness: Actor.InverseDirection	
					FALSE	Der Richtungssinn wird nicht invertiert.
					TRUE	Der Richtungssinn wird invertiert.
DirectionMode	INT	0 (0 bis 2)	R WP_PTO OPRW	- 2	Zugelassene Drehrichtung Name in Openness: Actor.DirectionMode	
					0	Beide Richtungen
					1	Positive Richtung
					2	Negative Richtung
DataAdaption	DINT	0 (0 bis 1)	R OPRW	-	Name in Openness: Actor.DataAdaption	
					0	Die automatische Übernahme der Antriebsparameter im Gerät ist deaktiviert.
					1	Die automatische Übernahme der Antriebsparameter im Gerät ist aktiviert.
Interface.	STRUCT				TO_Struct_ActorInterface	
AddressIn.	VREF	-	-	-	Eingangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
AREA	BYTE	-	OPR	-	Name in Openness: Actor.Interface.AddressIn.AREA	
DB_NUMBER	UINT	-	OPR	-	Name in Openness: Actor.Interface.AddressIn.DB_NUMBER	
OFFSET	UDINT	-	OPR	-	Name in Openness: Actor.Interface.AddressIn.OFFSET	
RID	DWORD	-	OPR	-	Name in Openness: Actor.Interface.AddressIn.RID	
ProfiDriveIn*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.ProfiDriveIn Gültige Eingangsadresse, die Teil eines Telegramms ist Gültiger Variablenname	
AddressOut.	VREF	-	-	-	Ausgangsadresse für das PROFIdrive-Telegramm (interner Parameter)	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
AREA	BYTE	-	OPR	-	Name in Openness: Actor.Interface.AddressOut.AREA
DB_NUMBER	UINT	-	OPR	-	Name in Openness: Actor.Interface.AddressOut.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Name in Openness: Actor.Interface.AddressOut.OFFSET
RID	DWORD	-	OPR	-	Name in Openness: Actor.Interface.AddressOut.RID
ProfiDriveOut*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.ProfiDriveOut Gültige Ausgangsadresse, die Teil eines Telegramms ist Gültiger Variablenname
DataBlock*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.DataBlock Gültige Datenbausteinadresse
Analog*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.Analog Gültiger Analogausgang, gültige Datenbausteinadresse, gültiger Variablenname
DataConnection*	INT	0 (0 bis 1)	OPRW	-	Name in Openness: _Actor.Interface.DataConnection 0 Antrieb 1 Datenbaustein
EnableDriveOutput	VREF	-	-	-	Freigabeausgang (interner Parameter)
AREA	BYTE	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyOutput.AREA
DB_NUMBER	UINT	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyOutput.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyOutput.OFFSET
RID	DWORD	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyOutput.RID
EnableDriveOutput*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.EnableDriveOutput Gültiger Eingang, gültiger Ausgang, gültige Speicheradresse, gültiger Variablenname
DriveReadyInput	VREF	-	-	-	Bereiteingang (interner Parameter)
AREA	BYTE	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyInput.AREA
DB_NUMBER	UINT	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyInput.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyInput.OFFSET
RID	DWORD	-	OPR	-	Name in Openness: Actor.Interface.DriveReadyInput.RID

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
DriveReadyInput*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.DriveReadyInput Gültiger Eingang, gültiger Ausgang, gültige Speicheradresse, gültiger Variablenname
PTO	DWORD	0	OPR	-	Impulsausgang (interner Parameter)
PTO*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.PTO Name der Impulsgeneratoren wie in der Hardwarekonfiguration
PTO_OutputA*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.PTO_OutputA Gültige Eingangsadresse, gültiger Variablenname Nur On-board CPU- oder Dignalboard-Adressen werden angenommen.
PTO_OutputBEnable*	BOOL		OPRW	-	Name in Openness: _Actor.Interface.PTO_OutputBEnable Nur möglich bei PTO_SignalType = 2 FALSE: Der Ausgang B ist gesperrt. TRUE: Der Ausgang B ist freigegeben.
PTO_OutputB*	STRING	-	OPRW	-	Name in Openness: _Actor.Interface.PTO_OutputB Gültige Eingangsadresse, gültiger Variablenname Nur On-board CPU- oder Dignalboard-Adressen werden angenommen.
PTO_SignalType*	INT	(2 bis 5)	OPRW	-	Name in Openness: _Actor.Interface.PTO_SignalType 2: Impuls A und Richtung B 3: Vorwärts takten A und rückwärts takten B 4: A/B phasenversetzt 5: A/B phasenversetzt - vierfach
DriveParameter.	STRUCT				TO_Struct_ActorDriveParameter
ReferenceSpeed	REAL	3000.0	R OPRW	-	Bezugswert (100 %) für die Solldrehzahl des Antriebs (NSOLL) Der Drehzahlsollwert wird im PROFIdrive-Telegramm als normierter Wert von -200 % bis 200 % von "ReferenceSpeed" übertragen. Bei Sollwertvorgabe über einen Analogwert kann der Analogausgang im Bereich -117 % bis 117 % betrieben werden, wenn der Antrieb das zulässt. Name in Openness: Actor.DriveParameter.ReferenceSpeed
MaxSpeed	REAL	3000.0	R OPRW	-	Maximalwert für die Solldrehzahl des Antriebs (NSOLL) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Analogsollwert: MaxSpeed ≤ 1.17 × ReferenceSpeed) Name in Openness: Actor.DriveParameter.MaxSpeed

Variable			Datentyp	Werte	Zugriff	W	Beschreibung
		PulsesPerDri-veRevolution	DINT	1000 (1 bis 2147483648)	R WP_PTO OPRW	- 2	Pulse pro Motorumdrehung Name in Openness: Actor.DriveParameter.PulsesPerDriveRevolution

*) In Openness verfügbar

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Sensor[1] ab V6

Die Variablenstruktur <Achsname>.Sensor[1].<Variablenname> beinhaltet die Parameter des Gebers.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Sensor[1].	STRUCT				ARRAY[1..1] TO_Struct_Sensor

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Type	DINT	0 (0 bis 1)	R, OPRW	-	Gebertyp (interner Parameter) Name in Openness: Sensor[1].Type	
					0	Inkrementell
					1	Absolut
InverseDirection	BOOL	FALSE	R, OPRW	-	Invertierung des Istwerts Name in Openness: Sensor[1].InverseDirection	
					FALSE	Istwert wird nicht invertiert
					TRUE	Istwert wird invertiert
System	DINT	1 (0 bis 1)	R, OPRW	-	Gebersystem Name in Openness: Sensor[1].System	
					0	Linearer Geber
					1	Rotatorischer Geber
MountingMode	DINT	0 (0 bis 2)	R, OPRW	-	Anbauart des Gebers Name in Openness: Sensor[1].MountingMode	
					0	An der Motorwelle
					2	Externes Messsystem
DataAdaption	DINT	0 (0 bis 1)	R, OPRW	-	Name in Openness: Sensor[1].DataAdaption	
					0	Die automatische Übernahme der Geberparameter im Gerät ist deaktiviert.
					1	Die automatische Übernahme der Geberparameter im Gerät ist aktiviert
Interface.	STRUCT				TO_Struct_SensorInterface	
Type	DINT	4 (0 bis 4)	OPR	-	Geberkopplung (interner Parameter) Name in Openness: Sensor[1].Interface.Type	
					0	PROFIdrive-Geber am PROFINET
					1	Geber am Technologiemodul (TM)
					2	Geber am Antrieb
					4	Geber am schnellen Zähler
AddressIn.	VREF	-	-	-	Eingangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
AREA	BYTE	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressIn.AREA	
DB_NUMBER	UINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressIn.DB_NUMBER	
OFFSET	UDINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressIn.OFFSET	
RID	DWORD	-	OPR	-	Interner Parameter	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
					Name in Openness: Sensor[1].Interface.AddressIn.RID
ProfiDriveIn*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.ProfiDriveIn Gültige Eingangsadresse, die Teil eines Telegramms ist Gültiger Variablenname
AddressOut.	VREF	-	-	-	Ausgangsadresse für das PROFIdrive-Telegramm (interner Parameter)
AREA	BYTE	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressOut.AREA
DB_NUMBER	UINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressOut.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressOut.OFFSET
RID	DWORD	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].Interface.AddressOut.RID
ProfiDriveOut*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.ProfiDriveOut Gültige Ausgangsadresse, die Teil eines Telegramms ist Gültiger Variablenname
DataBlock*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.DataBlock Gültige Datenbausteinadresse
DataConnection*	UDINT	(0 bis 1)	OPRW	-	Name in Openness: _Sensor[1].Interface.DataConnection 0 Geber 1 Datenbaustein
EncoderConnection*	INT	(4 bis 7)	OPRW	-	Name in Openness: _Sensor[1].Interface.EncoderConnection 4 Geber am schnellen Zähler (HSC) 7 Geber am PROFIBUS/PROFINET
Number	UDINT	1	OPRW	-	Gebernummern Name in Openness: Sensor[1].Interface.Number
HSC	DWORD	0	OPR	-	Schneller Zähler, an den der Geber den Istwert überträgt (interner Parameter)
HSC*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.HSC Namen von schnellen Zählern aus der Hardwarekonfiguration
HSC_OperatingMode*	INT	(1 bis 3)	OPRW	-	Name in Openness: _Sensor[1].Interface.HSC_OperatingMode

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					1	Zweiphasig
					2	A/B-Zähler
					3	A/B-Zähler vierfach
HSC_InputA*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.HSC_InputA Gültige Eingangsadresse, gültiger Variablenname	
HSC.InputB*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].Interface.HSC.InputB Gültige Eingangsadresse, gültiger Variablenname	
Parameter.	STRUCT				TO_Struct_SensorParameter	
Resolution	REAL	0.001 (-1.0E12 bis 1.0E12)	R, OPRW	-	Auflösung eines linearen Gebers (Abstand zwischen zwei Geberstrichen) Name in Openness: Sensor[1].Parameter.Resolution	
StepsPerRevolution	UDINT	2048 (1 bis 8388608)	R, OPRW	-	Inkremente pro Geberumdrehung bei einem rotatorischen Geber Name in Openness: Sensor[1].Parameter.StepsPerRevolution	
FineResolutionXist1	UDINT	11 (0 bis 31)	R, OPRW	-	Anzahl Bits für die Feinauflösung Gn_XIST1 (zyklischer Geberistwert) Name in Openness: Sensor[1].Parameter.FineResolutionXist1	
FineResolutionXist2	UDINT	9 (0 bis 31)	R, OPRW	-	Anzahl Bits für die Feinauflösung Gn_XIST2 (Absolutwert des Gebers) Name in Openness: Sensor[1].Parameter.FineResolutionXist2	
DeterminableRevolutions	UDINT	1 (0 bis 8388608)	R, OPRW	-	Anzahl unterscheidbarer Geberumdrehungen bei einem Multiturn-Absolutwertgeber Name in Openness: Sensor[1].Parameter.DeterminableRevolutions	
					0	Inkrementtalgeber
					1	Singlereturn-Absolutwertgeber
DistancePerRevolution	REAL	100.0 (0.0 bis 1.0E12)	R, OPRW	-	Weg der Last pro Geberumdrehung bei einem extern montierten Geber Name in Openness: Sensor[1].Parameter.DistancePerRevolution	
ActiveHoming.	STRUCT				TO_Struct_SensorActiveHoming	
Mode	DINT	2 (0 bis 2)	R, WP_PTO, OPRW	- 2	Modus aktives Referenzieren Name in Openness: Sensor[1].ActiveHoming.Mode	
					Technologieobjekt Positionierachse ab V5:	
					0	Nullmarke über PROFIdrive-Telegramm (nicht PTO)
					1	Nullmarke über PROFIdrive-Telegramm und Näherungsschalter (nicht PTO)
					2	Referenzieren über Digitaleingang
					Technologieobjekt Positionierachse V4:	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					2	Referenzieren über Digitaleingang
SideInput	BOOL	FALSE	RW, WP, OPRW	1, 8, 10	Seite des Referenzpunktschalters, auf die beim aktiven Referenzieren referenziert wird Name in Openness: Sensor[1].ActiveHoming.SideInput	
					FALSE	Untere Seite
					TRUE	Obere Seite
DigitalInputAdd-ress.	VREF	-	-	-	Symbolische Eingangsadresse des Referenzpunkt-schalters (interner Parameter)	
AREA	BYTE	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].ActiveHoming.DigitalInputAddress.AREA	
DB_NUMBER	UINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].ActiveHoming.DigitalInputAdd-ress.DB_NUMBER	
OFFSET	UDINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].ActiveHoming.DigitalInputAddress.OFFSET	
RID	DWORD	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].ActiveHoming.DigitalInputAddress.RID	
DigitalInput*	STRING	-	OPRW	-	Name in Openness: _Sensor[1].ActiveHoming.DigitalInput Gültige Eingangsadresse, gültiger Variablenname	
HomePosition-Offset	REAL	0.0 (-1.0E12 bis 1.0E12)	RW, WP, OPRW	1, 8, 10	Referenzpunktverschiebung (aktives Referenzieren) (Angabe in der konfigurierten Maßeinheit) Name in Openness: Sensor[1].ActiveHoming.HomePositionOffset	
SwitchLevel	BOOL	TRUE	RW, WP, OPRW	1, 8, 10	Auswahl des Signalpegels, der bei angefahrenem Re-ferenzpunktschalter am CPU-Eingang ansteht Name in Openness: Sensor[1].ActiveHoming.SwitchLevel	
					FALSE	Unterer Pegel (Low-aktiv)
					TRUE	Oberer Pegel (High-aktiv)
PassiveHoming.	STRUCT				TO_Struct_SensorPassiveHoming	
DigitalInputAdd-ress.	VREF	-	-	-	Symbolische Eingangsadresse des Referenzpunkt-schalters (interner Parameter)	
AREA	BYTE	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.AREA	
DB_NUMBER	UINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAdd-ress.DB_NUMBER	
OFFSET	UDINT	-	OPR	-	Interner Parameter	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
					Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.OFFSET
RID	DWORD	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.RID
Mode	DINT	2 (0 bis 2)	R WP_PTO OPRW	- 2	Modus passives Referenzieren Name in Openness: Sensor[1].PassiveHoming.Mode
					Technologieobjekt Positionierachse ab V5:
					0 Nullmarke über PROFIdrive-Telegramm (nicht PTO)
					1 Nullmarke über PROFIdrive-Telegramm und Näherungsschalter (nicht PTO)
					2 Referenzieren über Digitaleingang
					Technologieobjekt Positionierachse V4:
					2 Referenzieren über Digitaleingang
SideInput	BOOL	FALSE	RW, WP, OPRW	1, 7, 10	Seite des Referenzpunktschalters, auf die beim passiven Referenzieren referenziert wird Name in Openness: Sensor[1].PassiveHoming.SideInput
					FALSE Untere Seite
					TRUE Obere Seite
DigitalInputAddress.	VREF	-	-	-	Symbolische Eingangsadresse des Referenzpunktschalters (interner Parameter)
AREA	BYTE	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.AREA
DB_NUMBER	UINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.DB_NUMBER
OFFSET	UDINT	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.OFFSET
RID	DWORD	-	OPR	-	Interner Parameter Name in Openness: Sensor[1].PassiveHoming.DigitalInputAddress.RID
DigitalInput*	STRING		OPRW	-	Name in Openness: _Sensor[1].PassiveHoming.DigitalInput Gültige Eingangsadresse, gültiger Variablenname
SwitchLevel	BOOL	TRUE	RW, WP, OPRW	1, 7, 10	Auswahl des Pegels, der bei angefahrenem Referenzpunktschalter am CPU-Eingang ansteht Name in Openness: Sensor[1].PassiveHoming.SwitchLevel
					FALSE Unterer Pegel (Low-aktiv)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					TRUE	Oberer Pegel (High-aktiv)

*) In Openness verfügbar

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variable Units ab V6

Die Variablenstruktur <Achsname>.Units.LengthUnit beinhaltet die konfigurierten Maßeinheiten der Parameter.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Units.	STRUCT				TO_Struct_Units	
LengthUnit	INT	1013 (-32768 bis 32767)	R WP_PTO OPRW	- 2	Konfigurierte Maßeinheit der Parameter Name in Openness: Units.LengthUnit	
					-1	Impulse
					1005	°
					1010	m
					1013	mm
					1018	ft
					1019	in

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variable Mechanics ab V6

Die Variablenstruktur <Achsname>.Mechanics.LeadScrew beinhaltet den zurückgelegten Weg pro Motorumdrehung.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Mechanics.	STRUCT				TO_Struct_Mechanics
LeadScrew	REAL	10.0 (-1.0E12 bis 1.0E12)	R, WP_PTO, OPRW	-	Weg pro Motorumdrehung (Angabe in der konfigurierten Maßeinheit) Name in Openness: Mechanics.LeadScrew

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Modulo ab V6

Die Variablenstruktur <Achsname>.Modulo.<Variablenname> beinhaltet die Moduloeinstellungen.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Modulo.	STRUCT				TO_Struct_Modulo
Enable	BOOL	FALSE	R, OPRW	-	Name in Openness: Modulo.Enable
					FALSE: Moduloumrechnung deaktiviert
					TRUE: Moduloumrechnung aktiviert
					Bei aktivierter Moduloumrechnung wird auf Modullänge > 0.0 geprüft
Length	REAL	360.0 (0.001 bis 1.0E12)	R, OPRW	-	Modullänge Name in Openness: Modulo.Length
StartValue	REAL	0.0 (-1.0E12 bis 1.0E12)	R, OPRW	-	Modulostartwert Name in Openness: Modulo.StartValue

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen DynamicLimits ab V6

Die Variablenstruktur <Achsname>.DynamicLimits.<Variablenname> beinhaltet die Konfiguration der Dynamikgrenzen.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits
MaxVelocity	REAL	250.0	R WP_PTO OPRW	- 2	Maximale Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: DynamicLimits.MaxVelocity
MinVelocity	REAL	10.0	R WP_PTO OPRW	- 2	Start/Stopp-Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: DynamicLimits.MinVelocity

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen DynamicDefaults ab V6

Die Variablenstruktur <Achsname>.DynamicDefaults.<Variablenname> beinhaltet die Konfiguration der Dynamikvoreinstellungen.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults
Acceleration	REAL	48.0 (0.0 bis 1.0E12)	RW, WP, OPRW	5, 6, 10	Voreinstellung der Beschleunigung der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: DynamicDefaults.Acceleration
Deceleration	REAL	48.0 (0.0 bis 1.0E12)	RW, WP, OPRW	5, 6, 10	Voreinstellung der Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: DynamicDefaults.Deceleration
Jerk	REAL	192.0 (0.0 bis 1.0E12)	RW, WP, OPRW	5, 10	Voreinstellung des Ruck während der Beschleunigungs- und Verzögerungsrampe der Achse (Angabe in der konfigurierten Maßeinheit) Der Ruck ist aktiviert, wenn der konfigurierte Ruck größer als 0.00004 mm/s² ist. Name in Openness: DynamicDefaults.Jerk
EmergencyDeceleration	REAL	120.0 (0.0 bis 1.0E12)	RW, WP, OPRW	1, 5, 6, 10	Notstopp-Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit) Name in Openness: DynamicDefaults.EmergencyDeceleration

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionLimits_SW ab V6

Die Variablenstruktur <Achsname>.PositionLimits_SW.<Variablenname> beinhaltet die die Konfiguration der Positionsüberwachung mit Software-Endschaltern. Mit Software-Endschaltern begrenzen Sie den Arbeitsbereich einer Positionierachse.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionLimits_SW.	STRUCT				TO_Struct_PositionLimitsSW
Active	BOOL	FALSE	RW, WP, OPRW	1, 5, 6, 10	Name in Openness: PositionLimits_SW.Active FALSE: Die Software-Endschalter sind deaktiviert. TRUE: Die Software-Endschalter sind aktiviert.
MinPosition	REAL	-10000.0 (-1.0E12 bis 1.0E12)	RW, WP, OPRW	1, 5, 6, 10	Position des unteren Software-Endschalters(Angabe in der konfigurierten Maßeinheit) Name in Openness: PositionLimits_SW.MinPosition
MaxPosition	REAL	10000.0 (-1.0E12 bis 1.0E12)	RW, WP, OPRW	1, 5, 6, 10	Position des oberen Software-Endschalters (Angabe in der konfigurierten Maßeinheit) Name in Openness: PositionLimits_SW.MaxPosition

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionLimits_HW ab V6

Die Variablenstruktur <Achsname>.PositionLimits_HW.<Variablenname> beinhaltet die die Konfiguration der Positionsüberwachung mit Hardware-Endschaltern. Mit Hardware-Endschaltern begrenzen Sie den Verfahrbereich einer Positionierachse.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionLimits_HW.	STRUCT				TO_Struct_PositionLimitsHW
Active	BOOL	FALSE	RW, WP, OPRW	1, 5, 6, 10	Name in Openness: PositionLimits_HW.Active
					FALSE: Die Hardware-Endschalter sind deaktiviert.
					TRUE: Die Hardware-Endschalter sind aktiviert.
MinSwitchLevel	BOOL	FALSE	RW WP_PTO	- 2	Auswahl des Signalpegels, der bei angefahrenem unteren Hardware-Endschalter am CPU-Eingang ansteht Name in Openness: PositionLimits_HW.MinSwitchLevel
					FALSE: Unterer Pegel (Low-aktiv)
					TRUE: Oberer Pegel (High-aktiv)
MinSwitchAddress.	VREF	-	-	-	Symbolische Eingangsadresse des unteren Hardware-Endschalters (interner Parameter)
AREA	BYTE	-	OPR	-	Name in Openness: PositionLimits_HW.MinSwitchAddress.AREA
DB_NUMBER	US-HORT	-	OPR	-	Name in Openness: PositionLimits_HW.MinSwitchAddress.DB_NUMBER
OFFSET	UINT	-	OPR	-	Name in Openness: PositionLimits_HW.MinSwitchAddress.OFFSET
RID	UINT	-	OPR	-	Name in Openness: PositionLimits_HW.MinSwitchAddress.RID
MinSwitch*	STRING	-	OPRW	-	Name in Openness: _PositionLimits_HW.MinSwitch Gültige Eingangsadresse, gültiger Variablenname
MaxSwitchLevel	BOOL	FALSE	RW WP_PTO	- 2	Auswahl des Signalpegels, der bei angefahrenem oberen Hardware-Endschalter am CPU-Eingang ansteht Name in Openness: PositionLimits_HW.MaxSwitchLevel
					FALSE: Unterer Pegel (Low-aktiv)
					TRUE: Oberer Pegel (High-aktiv)
MaxSwitchAddress.	VREF	-	-	-	Eingangsadresse des oberen Hardware-Endschalters (interner Parameter)
AREA	BYTE	-	OPR	-	Name in Openness: PositionLimits_HW.MaxSwitchAddress.AREA
DB_NUMBER	US-HORT	-	OPR	-	Name in Openness: PositionLimits_HW.MaxSwitchAddress.DB_NUMBER
OFFSET	UINT	-	OPR	-	Name in Openness: PositionLimits_HW.MaxSwitchAddress.OFFSET
RID	UINT	-	OPR	-	Name in Openness: PositionLimits_HW.MaxSwitchAddress.RID
MaxSwitch*	STRING	-	OPRW	-	Name in Openness: _PositionLimits_HW.MaxSwitch Gültige Eingangsadresse, gültiger Variablenname

*) In Openness verfügbar

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Homing ab V6

Die Variablenstruktur <Achsname>.Homing.<Variablenname> beinhaltet die die Konfiguration für das Referenzieren der Achse.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	FALSE	RW, WP, OPRW	1, 8, 10	Name in Openness: Homing.AutoReversal	
					FALSE	Die Richtungsumkehr am Hardware-Endschalter ist deaktiviert.
					TRUE	Die Richtungsumkehr am Hardware-Endschalter ist aktiviert.
ApproachDirection	BOOL	TRUE	RW, WP, OPRW	1, 8, 10	Name in Openness: Homing.ApproachDirection	
					FALSE	Negative Anfahrrichtung zur Suche des Referenzpunktschalters und negative Referenzierrichtung
					TRUE	Positive Anfahrrichtung zur Suche des Referenzpunktschalters und positive Referenzierrichtung
ApproachVelocity	REAL	200.0 (0.0 bis 1.0E12)	RW, WP, OPRW	1, 8, 10	Anfahrgeschwindigkeit der Achse beim aktiven Referenzieren (Angabe in der konfigurierten Maßeinheit) Name in Openness: Homing.ApproachVelocity	
ReferencingVelocity	REAL	40.0 (0.0 bis 1.0E12)	RW, WP, OPRW	1, 8, 10	Referenziergeschwindigkeit der Achse beim aktiven Referenzieren (Angabe in der konfigurierten Maßeinheit) Name in Openness: Homing.ReferencingVelocity	

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionControl ab V6

Die Variablenstruktur <Achsname>.PositionControl.<Variablenname> beinhaltet Einstellungen der Lageregelung.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionControl.	STRUCT				TO_Struct_PositionControl
Kv	REAL	10.0 (0.0 bis 2147480.0)	R WP_PD OPRW	- 10	P-Verstärkung der Lageregelung ("Kv" > 0.0) Name in Openness: PositionControl.Kv
Kpc	REAL	100.0 (0.0 bis 150.0)	R WP_PD OPRW	- 10	Prozentuale Geschwindigkeitsvorsteuerung der Lageregelung Name in Openness: PositionControl.Kpc

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen FollowingError ab V6

Die Variablenstruktur <Achsname>.FollowingError.<Variablenname> beinhaltet die Konfiguration der dynamischen Schleppfehlerüberwachung.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
FollowingError.	STRUCT				TO_Struct_FollowingError
EnableMonitoring	BOOL	TRUE	R OPRW	-	Name in Openness: FollowingError.EnableMonitoring
					FALSE: Schleppfehlerüberwachung deaktiviert
					TRUE: Schleppfehlerüberwachung aktiviert
MinValue	REAL	10.0 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Zulässiger Schleppfehler bei Geschwindigkeiten unterhalb des Wertes von "MinVelocity". Name in Openness: FollowingError.MinValue
MaxValue	REAL	100.0 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Maximal zulässiger Schleppfehler, der beim Maximum der Geschwindigkeit erreicht werden darf. Name in Openness: FollowingError.MaxValue
MinVelocity	REAL	10.0 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	"MinValue" ist unterhalb dieser Geschwindigkeit zulässig und wird konstant gehalten. Name in Openness: FollowingError.MinVelocity

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositioningMonitoring ab V6

Die Variablenstruktur <Achsname>.PositionMonitoring.<Variablenname> beinhaltet die Konfiguration der Positionsüberwachung am Ende einer Positionierbewegung.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionMonitoring.	STRUCT				TO_Struct_PositionMonitoring
ToleranceTime	REAL	1.0 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Toleranzzeit Maximal erlaubte Zeitdauer vom Erreichen der Sollgeschwindigkeit null bis zum Eintritt in das Positionierfenster. Name in Openness: PositionMonitoring.ToleranceTime
MinDwellTime	REAL	0.1 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Minimale Verweildauer im Positionierfenster Name in Openness: PositionMonitoring.MinDwellTime
Window	REAL	1.0 (0.001 bis 1.0E12)	R WP_PD OPRW	- 10	Positionierfenster Name in Openness: PositionMonitoring.Window

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StandstillSignal ab V6

Die Variablenstruktur <Achsname>.StandstillSignal.<Variablenname> beinhaltet die Konfiguration des Stillstandsignals.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StandstillSignal.	STRUCT				TO_Struct_StandstillSignal
VelocityThreshold	REAL	5.0 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Geschwindigkeitsschwelle Wird diese unterschritten, beginnt die Mindestverweildauer. Name in Openness: StandStillSignal.VelocityThreshold
MinDwellTime	REAL	0.01 (0.0 bis 1.0E12)	R WP_PD OPRW	- 10	Mindestverweildauer Name in Openness: StandStillSignal.MinDwellTime

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusPositioning ab V6

Die Variablenstruktur <Achsname>.StatusPositioning.<Variablenname> zeigt den Status einer Positionierbewegung an.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StatusPositioning.	STRUCT				TO_Struct_StatusPositioning
Distance	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Aktueller Abstand der Achse zur Zielposition (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur gültig während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel. Name in Openness: StatusPositioning.Distance
TargetPosition	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Zielposition der Achse (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur gültig während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel. Name in Openness: StatusPositioning.TargetPosition
FollowingError	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Aktueller Schleppfehler der Achse (Angabe in der konfigurierten Maßeinheit FollowingError = 0.0 bei Antriebsanbindung über PTO (Pulse Train Output). Name in Openness: StatusPositioning.FollowingError

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusDrive ab V6

Die Variablenstruktur <Achsname>.StatusDrive.<Variablenname> zeigt den Status des Antriebs an.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
StatusDrive.	STRUCT				TO_Struct_StatusDrive	
InOperation	BOOL	FALSE	RCCP, RP, OPR	-	Operationsstatus des Antriebs Name in Openness: StatusDrive.InOperation	
					FALSE	Antrieb nicht bereit. Sollwerte werden nicht ausgeführt.
					TRUE	Antrieb bereit. Sollwerte können ausgeführt werden.
CommunicationOK	BOOL	FALSE	RCCP, RP, OPR	-	Zyklische BUS-Kommunikation zwischen Steuerung und Antrieb Name in Openness: StatusDrive.CommunicationOK	
					FALSE	Kommunikation nicht aufgebaut
					TRUE	Kommunikation aufgebaut
AdaptionState	DINT	0 (0 bis 4)	R, OPR	10	Übernahmestatus des Antriebs Name in Openness: StatusDrive.AdaptionState	
					0	Daten nicht übernommen
					1	Daten in Übernahme
					2	Daten übernommen
					3	Übernahme nicht möglich oder nicht ausgewählt
					4	Fehler bei der Übernahme

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusSensor ab V6

Die Variablenstruktur <Achsname>.StatusSensor[1].<Variablenname> zeigt den Status des Messsystems an.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
StatusSensor[1].	STRUCT				TO_Struct_StatusSensor	
State	DINT	0 (0 bis 2)	RCCP, RP, OPR	-	Status des Geberwerts Name in Openness: StatusSensor.State	
					0	Nicht gültig
					1	Warte auf Status gültig
					2	Gültig
CommunicationOK	BOOL	FALSE	RCCP, RP, OPR	-	Zyklische BUS-Kommunikation zwischen Steuerung und Geber Name in Openness: StatusSensor.CommunicationOK	
					FALSE	Kommunikation nicht aufgebaut
					TRUE	Kommunikation aufgebaut
AbsEncoderOffset	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP, OPR	-	Referenzpunktverschiebung zum Wert eines Absolutwertgebers. Der Wert wird remanent in der CPU gespeichert. Name in Openness: StatusSensor.AbsEncoderOffset	
AdaptionState	DINT	0 (0 bis 1)	R, OPR	10	Übernahmestatus des Gebers Name in Openness: StatusSensor.AdaptionState	
					0	Daten nicht übernommen
					1	Daten in Übernahme

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusBits ab V6

Die Variablenstruktur <Achsname>.StatusBits.<Variablenname> beinhaltet die Statusinformationen des Technologieobjekts.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StatusBits.	STRUCT				TO_Struct_StatusBits

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Activated	BOOL	FALSE	RCCP, RP, OPR	-	Aktivierung der Achse Name in Openness: StatusBits.Activated	
					FALSE	Die Achse ist nicht aktiviert.
					TRUE	Die Achse ist aktiviert. Die Achse ist mit dem zugeordneten PTO (Pulse Train Output) verbunden. Die Daten des Technologie-Datenbausteins werden zyklisch aktualisiert.
Enable	BOOL	FALSE	RCCP, RP, OPR	-	Freigabestatus der Achse Name in Openness: StatusBits.Enable	
					FALSE	Die Achse ist nicht freigegeben.
					TRUE	Die Achse ist freigegeben und bereit, Motion Control-Aufträge anzunehmen.
AxisSimulation	BOOL	FALSE	RCCP, RP, OPR	-	Name in Openness: StatusBits.AxisSimulation	
					FALSE	Die Simulation ist deaktiviert.
					TRUE	Die Simulation ist aktiviert.
NonPositionControlled	BOOL	FALSE	RCCP, RP, OPR	-	Name in Openness: StatusBits.NonPositionControlled	
					FALSE	Die Achse befindet sich im lagegeregelten Betrieb.
					TRUE	Die Achse befindet sich im nichtlagegeregelten Betrieb.
HomingDone	BOOL	FALSE	RCCP, RP, OPR	-	Referenzierungsstatus der Achse Name in Openness: StatusBits.HomingDone	
					FALSE	Die Achse ist nicht referenziert.
					TRUE	Die Achse ist referenziert und kann absolute Positionieraufträge ausführen.
					Zum relativen Positionieren muss die Achse nicht referenziert sein. Während des aktiven Referenzierens ist der Status FALSE. Während des passiven Referenzierens bleibt der Status TRUE, wenn die Achse zuvor bereits referenziert war.	
Done	BOOL	FALSE	RCCP, RP, OPR	-	Auftragsbearbeitung der Achse Name in Openness: StatusBits.Done	
					FALSE	An der Achse ist ein Motion Control-Auftrag aktiv.
					TRUE	An der Achse ist kein Motion Control-Auftrag aktiv.
Error	BOOL	FALSE	RCCP, RP, OPR	-	Fehlerstatus der Achse Name in Openness: StatusBits.Error	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					FALSE	An der Achse steht kein Fehler an.
					TRUE	An der Achse ist ein Fehler aufgetreten.
					Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern "ErrorID" und "ErrorInfo" der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt.	
Standstill	BOOL	FALSE	RCCP, RP, OPR	-	Stillstand der Achse Name in Openness: StatusBits.Standstill	
					FALSE	Die Achse ist in Bewegung.
					TRUE	Die Achse befindet sich im Stillstand.
PositioningCommand	BOOL	FALSE	RCCP, RP, OPR	-	Bearbeitung eines Positionierauftrags Name in Openness: StatusBits.PositioningCommand	
					FALSE	An der Achse ist kein Positionierauftrag aktiv.
					TRUE	Die Achse führt einen Positionierauftrag der Motion Control-Anweisungen "MC_MoveRelative" oder "MC_MoveAbsolute" aus.
VelocityCommand	BOOL	FALSE	RCCP, RP, OPR	-	Bearbeitung eines Auftrags mit Geschwindigkeitsvorgabe Name in Openness: StatusBits.VelocityCommand	
					FALSE	An der Achse ist kein Auftrag mit Geschwindigkeitsvorgabe aktiv.
					TRUE	Die Achse führt einen Verfahrauftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisungen "MC_MoveVelocity" oder "MC_MoveJog" aus.
HomingCommand	BOOL	FALSE	RCCP, RP, OPR	-	Bearbeitung eines Referenzierauftrags Name in Openness: StatusBits.HomingCommand	
					FALSE	An der Achse ist kein Referenzierauftrag aktiv.
					TRUE	Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home".
CommandTableActive	BOOL	FALSE	RCCP, RP, OPR	-	Bearbeitung einer Auftragstabelle Name in Openness: StatusBits.CommandTableActive	
					FALSE	An der Achse ist keine Auftragstabelle aktiv.
					TRUE	Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert.
ConstantVelocity	BOOL	FALSE	RCCP, RP, OPR	-	Konstante Geschwindigkeit Name in Openness: StatusBits.ConstantVelocity	
					FALSE	Die Achse wird beschleunigt, abgebremst oder befindet sich im Stillstand.

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					TRUE	Die Sollgeschwindigkeit wurde erreicht. Die Achse wird mit konstanter Geschwindigkeit verfahren.
Accelerating	BOOL	FALSE	RCCP, RP, OPR	-	Beschleunigungsvorgang Name in Openness: StatusBits.Accelerating	
					FALSE	Die Achse wird abgebremst, mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird beschleunigt.
Decelerating	BOOL	FALSE	RCCP, RP, OPR	-	Verzögerungsvorgang Name in Openness: StatusBits.Decelerating	
					FALSE	Die Achse wird beschleunigt, mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird abgebremst.
ControlPanelActive	BOOL	FALSE	RCCP, RP, OPR	-	Aktivierungsstatus der Achssteuertafel Name in Openness: StatusBits.ControlPanelActive	
					FALSE	Die Betriebsart "Automatikbetrieb" ist aktiviert. Das Anwenderprogramm hat die Steuerungshoheit über die Achse.
					TRUE	Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerungshoheit über die Achse. Die Achse kann nicht vom Anwenderprogramm gesteuert werden.
DriveReady	BOOL	FALSE	RCCP, RP, OPR	-	Operationsstatus des Antriebs Name in Openness: StatusBits.DriveReady	
					FALSE	Der Antrieb ist nicht bereit. Sollwerte werden nicht ausgeführt.
					TRUE	Der Antrieb ist bereit. Sollwerte können ausgeführt werden.
RestartRequired	BOOL	FALSE	RCCP, RP, OPR	-	Restart der Achse erforderlich Name in Openness: StatusBits.RestartRequired	
					FALSE	Ein Restart der Achse ist nicht erforderlich.
					TRUE	Werte wurden im Ladespeicher geändert.
					Um die Werte im Betriebszustand RUN der CPU in den Arbeitsspeicher zu laden, ist ein Restart der Achse erforderlich. Verwenden Sie hierzu die Motion Control-Anweisung "MC_Reset".	
SWLimitMinActive	BOOL	FALSE	RCCP, RP, OPR	-	Status des unteren Software-Endschalters Name in Openness: StatusBits.SWLimitMinActive	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					FALSE	Der konfigurierte Arbeitsbereich der Achse wird eingehalten.
					TRUE	Der untere Software-Endschalter wurde angefahren oder überfahren.
SWLimitMaxActive	BOOL	FALSE	RCCP, RP, OPR	-	Status des oberen Software-Endschalters Name in Openness: StatusBits.SWLimitMaxActive	
					FALSE	Der konfigurierte Arbeitsbereich wird eingehalten.
					TRUE	Der obere Software-Endschalter wurde angefahren oder überfahren.
HWLimitMinActive	BOOL	FALSE	RCCP, RP, OPR	-	Status des unteren Hardware-Endschalters Name in Openness: StatusBits.HWLimitMinActive	
					FALSE	Der konfigurierte zulässige Verfahrbereich wird eingehalten.
					TRUE	Der untere Hardware-Endschalter wurde angefahren oder überfahren.
HWLimitMaxActive	BOOL	FALSE	RCCP, RP, OPR	-	Status des oberen Hardware-Endschalters Name in Openness: StatusBits.HWLimitMaxActive	
					FALSE	Der konfigurierte zulässige Verfahrbereich wird eingehalten.
					TRUE	Der obere Hardware-Endschalter wurde angefahren oder überfahren.

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen ErrorBits ab V6

Die Variablenstruktur <Achsname>.ErrorBits.<Variablenname> zeigt Fehler am Technologieobjekt an.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	RCCP, RP, OPR	-	Interner Systemfehler Name in Openness: ErrorBits.SystemFault
ConfigFault	BOOL	FALSE	RCCP, RP, OPR	-	Fehlerhafte Konfiguration der Achse Name in Openness: ErrorBits.ConfigFault
DriveFault	BOOL	FALSE	RCCP, RP, OPR	-	Fehler im Antrieb. Ausfall des "Antrieb bereit"-Signals. Name in Openness: ErrorBits.DriveFault
SWLimit	BOOL	FALSE	RCCP, RP, OPR	-	Software-Endschalter angefahren oder überfahren Name in Openness: ErrorBits.SWLimit
HWLimit	BOOL	FALSE	RCCP, RP, OPR	-	Hardware-Endschalter angefahren oder überfahren Name in Openness: ErrorBits.HWLimit
DirectionFault	BOOL	FALSE	RCCP, RP, OPR	-	Unzulässige Bewegungsrichtung Name in Openness: ErrorBits.DirectionFault
HWUsed	BOOL	FALSE	RCCP, RP, OPR	-	Eine weitere Achse verwendet denselben PTO (Pulse Train Output) und ist mit "MC_Power" freigegeben. Name in Openness: ErrorBits.HWUsed
SensorFault	BOOL	FALSE	RCCP, RP, OPR	-	Fehler im Gebersystem Name in Openness: ErrorBits.SensorFault
Communication-Fault	BOOL	FALSE	RCCP, RP, OPR	-	Kommunikationsfehler Die Kommunikation mit einem verbundenen Gerät ist gestört. Name in Openness: ErrorBits.CommunicationFault
FollowingError	BOOL	FALSE	RCCP, RP, OPR	-	Maximal zulässiger Schleppfehler überschritten Name in Openness: ErrorBits.FollowingError
PositioningFault	BOOL	FALSE	RCCP, RP, OPR	-	Positionierfehler Die Achse wurde am Ende einer Positionierbewegung nicht korrekt positioniert. Name in Openness: ErrorBits.PositioningFault
AdaptionError	BOOL	FALSE	RCCP, RP, OPR	-	Fehler bei der Datenübernahme Name in Openness: ErrorBits.AdaptionError

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen ControlPanel ab V6

Die Variablen "ControlPanel" beinhalten keine anwenderrelevanten Daten. Im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Variablen

Folgende Variablen "ControlPanel" sind in Openness lesbar.

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
ControlPanel.	STRUCT				TO_Struct_ControlPanel

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Input	STRUCT				TO_Struct_Input
TimeOut	DINT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.TimeOut
EsLifeSign	DINT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.EsLifeSign
Command[1].	STRUCT				ARRAY[1..1] TO_Struct_Command
ReqCounter	DINT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].ReqCounter
Type	DINT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].Type
Position	REAL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Com-mand[1].Position
Velocity	REAL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].Velocity
Acceleration	REAL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].Acceleration
Jerk	REAL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].Jerk
Param	INT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Input.Command[1].Param
Output.	STRUCT		-		TO_Struct_Output
RTLifeSign	INT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.RTLifeSign
Command[1].	STRUCT		-		ARRAY[1..1] TO_Struct_Command
AckCounter	INT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.Command[1].AckCounter
ErrorID	US-HORT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.Command[1].ErrorID
ErrorInfo	US-HORT	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.Command[1].ErrorInfo
Done	BOOL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.Command[1].Done

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Aborted	BOOL	-	OPR	-	(interner Parameter) Name in Openness: ControlPanel.Output.Command[1].Aborted

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Internal ab V6

Die Variablen "Internal" beinhalten keine anwenderrelevanten Daten. Im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Internal[1..4].	STRUCT				ARRAY [1..4] TO_Struct_Internal
Id	INT	0 (-32768 bis 32767)	OPRW	-	(interner Parameter) Name in Openness: Internal[1..4].Id
Value	REAL	0 (-9.0E15 bis 9.0E15)	OPRW	-	(interner Parameter) Name in Openness: Internal[1..4].Value

Siehe auch

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Aktualisierung der Variablen des Technologieobjekts

Die in den Variablen des Technologieobjekts angezeigten Status- und Fehlerinformationen der Achse werden jeweils am Zykluskontrollpunkt aktualisiert.

Wertänderungen bei Konfigurationsvariablen werden nicht sofort wirksam. Unter welchen Bedingungen eine Änderung wirksam wird, entnehmen Sie der Detailbeschreibung der entsprechenden Variable.

Variablen des Technologieobjekts Auftragstabelle V6

Die Variablenstruktur <Auftragstabelle>.Command[n].<Variablenname> beinhaltet die konfigurierten Auftragsparameter.

Variablen

Legende (Seite 1543)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Command[n].	STRUCT				ARRAY[1..32] TO_Struct_Command	
Type	INT	0 (0 bis 151)	RW, OPRW	-	Auftragstyp Name in Openness: Command[n].Type	
					0	Auftrag "Empty"
					2	Auftrag "Halt"
					5	Auftrag "Positioning Relative"
					6	Auftrag "Positioning Absolute"
					7	Auftrag "Velocity setpoint"
					151	Auftrag "Wait"
Position	REAL	0.0	RW, OPRW	-	Zielposition/Verfahrweg des Auftrags Name in Openness: Command[n].Position	
Velocity	REAL	0.0	RW, OPRW	-	Geschwindigkeit des Auftrags Name in Openness: Command[n].Velocity	
Duration	REAL	0.0	RW, OPRW	-	Dauer des Auftrags Name in Openness: Command[n].Duration	
NextStep	INT	0 (0 bis 1)	RW, OPRW	-	Modus für den Übergang zum nächsten Auftrag Name in Openness: Command[n].NextStep	
					0	"Auftrag abschließen"
					1	"Bewegung überschleifen"
StepCode	WORD	0	RW, OPRW	-	Schrittcode des Auftrags Name in Openness: Command[n].StepCode	
WarningEnabled*	BOOL	FALSE	OPRW	-	Name in Openness: _WarningEnabled	
					FALSE	Warnung ist gesperrt
					TRUE	Warnung ist freigegeben
UseAxisParametersFrom*	INT/ STRING	-	OPRW	-	Name in Openness: _UseAxisParametersFrom Achsnummer, "Beispielachse", Name der Achse	

*) In Openness verfügbar

Siehe auch

Variablen des Technologieobjekts Auftragstabelle V4...5 (Seite 1678)

Variablen des Technologieobjekts Auftragstabelle V1...3 (Seite 1677)

Versionen V1...5

Für Motion Control relevante Ausgänge der CPU (Technologieversion V1...3)

Die Anzahl der nutzbaren Antriebe ist abhängig von der CPU, der Anzahl der PTOs (Pulse Train Outputs) und der zur Verfügung stehenden Anzahl an Impulsgeneratorausgängen.

Die folgenden Aufstellungen geben Auskunft, über die entsprechenden Abhängigkeiten:

Maximale Anzahl der PTOs

Die maximale Anzahl der ansteuerbaren PTOs (Antriebe) ist von der Artikelnummer der CPU abhängig:

Artikelnummer CPU	Anzahl der PTOs
xxxxxxx-1xx**30**-xxxx	2
xxxxxxx-1xx**31**-xxxx	4

Die maximale Anzahl der ansteuerbaren PTOs (Antriebe) gilt unabhängig vom Einsatz eines Signalboards.

Nutzbare Impulsgeneratorausgänge

Zur Ansteuerung eines Schrittmotorantriebs oder eines Servomotorantriebs mit Impulsschnittstelle stellt die CPU einen Impulsausgang und einen Richtungsausgang zur Verfügung. Über den Impulsausgang erhält der Antrieb die notwendigen Impulse zur Bewegung des Motors. Der Richtungsausgang steuert die Fahrtrichtung des Antriebs.

Impuls- und Richtungsausgang sind zueinander fest zugeordnet und bilden einen Impulsgeneratorausgang. Als Impulsgeneratorausgänge können On-board CPU-Ausgänge oder Ausgänge eines Signalboards verwendet werden. Die Auswahl zwischen On-board CPU-Ausgängen und Ausgängen des Signalboards treffen Sie in der Gerätekonfiguration unter Impulsgeneratoren (PTO/PWM) im Register "Eigenschaften".

Die nachfolgende Tabelle zeigt die Anzahl der nutzbaren Antriebe pro CPU, bzw. Signalboard:

CPU		On-board	Signalboard				
			DI2/DO2 x DC24V 20kHz	DI2/DO2 x DC24V 200kHz	DO4 x DC24V 200kHz	DI2/DO2 x DC5V 200kHz	DO4 x DC5V 200kHz
CPU 1211C, CPU 1212C, CPU 1214C (MLFB - Artikelnummer xxxxxxx-1xx**30**-xxxx)	DC/DC/DC	2	2	2	2	2	2
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2

CPU		On-board	Signalboard				
CPU 1211C (MLFB - Artikelnummer xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	2	3	3	4	3	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1212C (MLFB - Artikelnummer xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	3	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1214C (MLFB - Artikelnummer xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1215C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2

Die mögliche Adresszuordnung der Impuls- und Richtungsausgänge entnehmen Sie der nachfolgenden Tabelle:

CPU S7-1200		Ausgänge PTO1 [1]		Ausgänge PTO2 [2]		Ausgänge PTO3 [1]		Ausgänge PTO4 [2]	
		Imp.	Richt.	Imp.	Richt.	Imp.	Richt.	Imp.	Richt.
CPU 1211C, CPU 1212C, CPU 1214C, CPU1215C (DC/DC/DC)	CPU	Ax.0	Ax.1	Ax.2	Ax.3	Ax.4	Ax.5	Ax.6	Ax.7
	Signal-board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C (AC/DC/RLY)	CPU	-	-	-	-	-	-	-	-
	Signal-board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C (DC/DC/RLY)	CPU	-	-	-	-	-	-	-	-
	Signal-board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3

x = Anfangsbyteadresse der On-board CPU-Ausgänge (Standardwert = 0)

y = Anfangsbyteadresse der Signalboard-Ausgänge (Standardwert = 4)

[1] Wird eine CPU-Variante DC/DC/DC zusammen mit einem Signalboard DI2/DO2 eingesetzt, so können die Signale des PTO1/3 über die On-board CPU-Ausgänge oder über das Signalboard ausgegeben werden.

[2] Wird eine CPU-Variante DC/DC/DC zusammen mit einem Signalboard DO4 eingesetzt, so können sowohl die Signale für PTO1/3 als auch für PTO2/4 über die On-board CPU-Ausgänge oder über das Signalboard ausgegeben werden.

PTO3 und PTO4 stehen nur bei CPUs mit der Artikelnummer xxxxxxx-1xx**31**-xxxx zur Verfügung.

Hinweis

Zugriff auf Impulgeneratorausgänge über das Prozessabbild

Wurde der PTO (Pulse Train Output) aktiviert und einer Achse zugeordnet, so übernimmt die Firmware die Kontrolle über die zugehörigen Impulsgeneratorausgänge.

Mit der Übernahme der Kontrolle wird auch die Verbindung zwischen dem Prozessabbild und dem Peripherie-Ausgang getrennt. Der Anwender hat zwar die Möglichkeit das Prozessabbild der Impulgeneratorausgänge über Anwenderprogramm oder Beobachtungstabelle zu beschreiben, dieses wird jedoch nicht auf den Peripherie-Ausgang übertragen werden. Dementsprechend ist es auch nicht möglich den Peripherie-Ausgang über Anwenderprogramm oder Beobachtungstabelle zu beobachten. Die gelesenen Informationen spiegeln den Wert des Prozessabbilds wieder, welche nicht mit dem realen Zustand des Peripherie-Ausgangs übereinstimmen.

Bei allen anderen, nicht fest durch die CPU-Firmware verwendeten CPU-Ausgängen, kann der Status des Peripherie-Ausgangs wie gewohnt über das Prozessabbild gesteuert bzw. überwacht werden.

Grenzfrequenzen der Impulsausgänge

Für die Impulsausgänge gelten folgende Grenzfrequenzen:

Impulsausgang	Grenzfrequenzen für Technologieobjekt Positionierachse V1	Grenzfrequenzen Technologieobjekt Positionierachse V2/V3 bei CPU < V3.0	Grenzfrequenzen Technologieobjekt Positionierachse V2/V3 bei CPU V3.0
On-board (MLFB - Artikelnummer xxxxxxx-1xx**30**-xxxx)	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 100 kHz	1 Hz ≤ f ≤ 100 kHz
On-board (MLFB - Artikelnummer xxxxxxx-1xx**31**xxxx)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	1 Hz ≤ f ≤ 100 kHz (PTO 1+2) 1 Hz ≤ f ≤ 20 kHz (PTO 3+4)
Signalboard DI2/DO2 x DC24V 20kHz	2 Hz ≤ f ≤ 20 kHz	2 Hz ≤ f ≤ 20 kHz	1 Hz ≤ f ≤ 20 kHz
Signalboard DI2/DO2 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz

Impulsausgang	Grenzfrequenzen für Technologieobjekt Positionierachse V1	Grenzfrequenzen Technologieobjekt Positionierachse V2/V3 bei CPU < V3.0	Grenzfrequenzen Technologieobjekt Positionierachse V2/V3 bei CPU V3.0
Signalboard DO4 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
Signalboard DI2/DO2 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
Signalboard DO4 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz

Antriebssignale

Für Motion Control können Sie optional eine Antriebsschnittstelle für "Antriebsfreigabe" und "Antrieb bereit" parametrieren. Bei Verwendung der Antriebsschnittstelle kann der digitale Ausgang für die Antriebsfreigabe und der digitale Eingang für "Antrieb bereit" frei gewählt werden.

Grenzwerte Beschleunigung / Verzögerung

Für die Beschleunigung und Verzögerung gelten folgende Grenzwerte:

Beschleunigung / Verzögerung	Wert (CPU < V3.0)	Wert (CPU V3.0)
Minimale Beschleunigung / Verzögerung	2,8E-1 Pulse/s^2	5,0E-3 Pulse/s^2
Maximale Beschleunigung / Verzögerung	9,5E+9 Pulse/s^2	9,5E+9 Pulse/s^2

Grenzwerte Ruck

Für den Ruck gelten folgende Grenzwerte:

Ruck	Wert (CPU < V3.0)	Wert (CPU V3.0)
Minimaler Ruck	4,0E-2 Pulse/s^3	4,0E-3 Pulse/s^3
Maximaler Ruck	1,5E+8 Pulse/s^3	1,0E+10 Pulse/s^3

Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 1344)

Konfigurationsdialoge

V1...3

Konfiguration - Allgemein (Technologieobjekt "Achse" V1...3)

Konfigurieren Sie im Konfigurationsfenster "Allgemein" die grundlegenden Eigenschaften des Technologieobjekts "Achse".

Achsname:

Definieren Sie in diesem Feld den Namen der Achse, bzw. den Namen des Technologieobjekts "Achse". Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

Hardware - Schnittstelle

Die Impulse werden über fest zugeordnete digitale Ausgänge an das Leistungsteil des Antriebs ausgegeben.

Bei CPUs mit Relais-Ausgängen kann das Impulssignal nicht auf diesen Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen. Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, muss ein Signalboard mit digitalen Ausgängen verwendet werden.

Hinweis

Der PTO benötigt die Funktionalität eines schnellen Zählers (HSC). Bei CPU Version < V3.0 wird hierzu ein HSC verwendet, der anschließend dem Anwender nicht mehr zur Verfügung steht. Bei CPU Version ≥ V3.0 wird hierzu ein interner HSC verwendet.

Der Zählerstand kann nicht über seine Eingangsadresse ausgewertet werden.

Die Zuordnung zwischen PTO und HSC ist fest. Bei anwenderseitiger Aktivierung des PTO1 wird dieser mit dem HSC1 verbunden. Wird der PTO2 aktiviert, so wird dieser mit dem HSC2 verbunden.

Wählen Sie in der Klappliste "Auswahl Impulsgenerator" den PTO (Pulse Train Output), über welchen die Impulse für die Ansteuerung der Schrittmotoren oder Servomotoren mit Impulsschnittstelle zur Verfügung gestellt werden sollen. Wurden in der Gerätekonfiguration die Impulsgeneratoren und die schnellen Zähler nicht anderweitig verwendet, so kann die Hardware-Schnittstelle automatisch konfiguriert werden. In diesem Fall wird der gewählte PTO in der Klappliste weiß hinterlegt angezeigt. In den Ausgabefeldern "Ausgangsquelle", "Impulsausgang", "Richtungsausgang" und "Zugewiesener schneller Zähler" werden die verwendeten Schnittstellen aufgelistet.

Gehen Sie wie folgt vor, wenn Sie die Schnittstellen ändern möchten, oder der PTO nicht automatisch konfiguriert werden konnte (Eintrag in der Klappliste "Auswahl Impulsgenerator" ist rot hinterlegt):

1. Klicken Sie auf die Schaltfläche "Gerätekonfiguration".
   Die Gerätekonfiguration des Impulsgenerators öffnet sich.
   Vergrößern Sie den Eigenschaftsbereich der Gerätekonfiguration, falls die Konfiguration des Impulsgenerators nicht sichtbar ist.

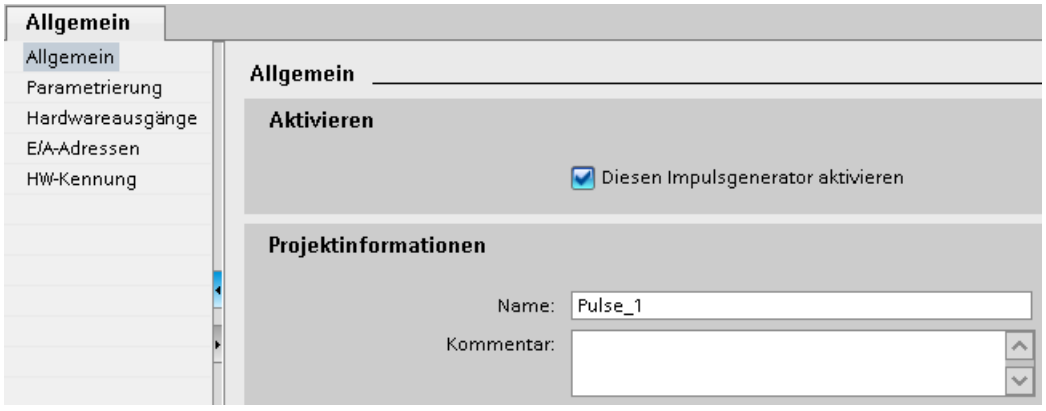

2. Aktivieren Sie das Optionskästchen "Diesen Impulsgenerator aktivieren".
3. Wählen Sie im Bereichsnavigator den Eintrag "Parametrierung"
   Die "Parametrierung öffnet sich.

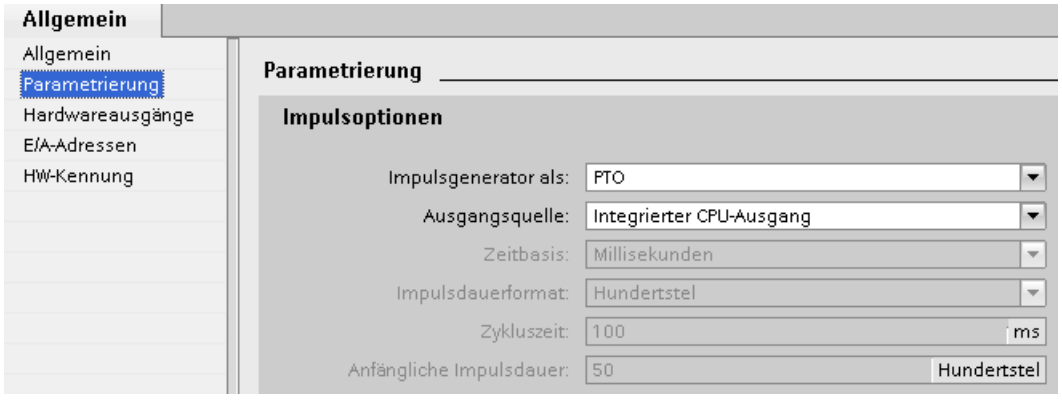

4. Wählen Sie in der Klappliste "Impulsgenerator als:" den Eintrag "PTO".
5. Wählen Sie in der Klappliste "Ausgangsquelle:" den Eintrag "Integrierter CPU-Ausgang" oder "Ausgang des Signalboards". Der Eintrag "Ausgang des Signalboards" ist je nach gestecktem Signalboard, nur für PTO1 oder für PTO1 und PTO2 auswählbar. Genauere Informationen entnehmen Sie dem Kapitel: Für Motion Control relevante Ausgänge der CPU (Seite 1344)
6. Kehren Sie in die Konfiguration der Achse zurück.
   Wurde der korrespondierende schnelle Zähler nicht bereits anderweitig verwendet, so sind die PTO-Felder der Achskonfiguration "Allgemein" ohne rote Hinterlegung. Korrigieren Sie die Konfiguration anhand der Fehlermeldungen falls dies nicht der Fall ist.

Anwendereinheit

Wählen Sie in der Klappliste die gewünschte Einheit für das Maßsystem der Achse aus. Die gewählte Einheit wird für die weitere Konfiguration des Technologieobjekts "Achse" und für die Anzeige der aktuellen Achsdaten verwendet.

Die Werte an den Eingangsparametern (Position, Distance, Velocity, ...) der Motion Control-Anweisungen beziehen sich ebenfalls auf diese Einheit.

Hinweis

Eine nachträgliche Änderung des Maßsystems kann u. U. nicht in allen Konfigurationsfenstern des Technologieobjekts korrekt umgerechnet werden. Kontrollieren Sie in diesem Fall die Konfiguration aller Achsparameter.

Gegebenenfalls müssen im Anwenderprogramm Werte an den Eingangsparametern der Motion Control-Anweisungen an die neue Maßeinheit angepasst werden.

Siehe auch

Konfiguration - Allgemein (Seite 1388)

Konfiguration - Referenzieren (Technologieobjekt "Achse" V1)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren" die Parameter für das aktive und das passive Referenzieren. Die Referenzierart wird über den Eingangsparameter "Mode" der Motion Control-Anweisung eingestellt. Hierbei entspricht Mode = 2 dem passiven und Mode = 3 dem aktiven Referenzieren.

Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

Erlaube Richtungsumkehr am HW-Endschalter (nur aktives Referenzieren)

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein. Wird die CPU-Firmware V1.0 eingesetzt, so müssen beide Hardware-Endschalter konfiguriert sein. Beim Einsatz der CPU-Firmware ab V2.0 muss nur der Hardware-Endschalter in Anfahrrichtung konfiguriert sein.

Wird der Hardware Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.

Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

Hinweis

Stellen Sie durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgeschwindigkeit gering
- Vergrößern Sie die konfigurierte Beschleunigung / Verzögerung
- Vergrößern Sie den Abstand zwischen Hardware-Endschalter und Hardwareanschlag

Anfahrrichtung- / Referenzierrichtung (aktives und passives Referenzieren)

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die "Anfahrrichtung" zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest mit welcher Fahrrichtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenziervorgang durchzuführen.

Die Wirkung der eingestellten Anfahrrichtung für das passive Referenzieren entnehmen Sie der Tabelle unter "Referenzpunktschalter".

Seite des Referenzpunktschalters (aktives und passives Referenzieren)

- **Aktives Referenzieren**
  Wählen Sie hier ob die Achse an der oberen oder der unteren Seite des Referenzpunktschalters referenziert werden soll.

  Hinweis

  Je nach Anfangsposition der Achse und Konfiguration der Referenzierparameter kann der Ablauf der Referenzpunktfahrt von der Grafik im Konfigurationsfenster abweichen.

- **Passives Referenzieren**
  Beim passiven Referenzieren müssen die Fahrbewegungen zum Referenzieren anwenderseitig durch Fahraufträge realisiert werden. An welcher Seite des Referenzpunktschalters referenziert wird, hängt von folgenden Faktoten ab:
  - Konfiguration "Anfahrrichtung"
  - Konfiguration "Referenzpunktschalter"
  - Aktuelle Fahrrichtung während des passiven Referenzierens

  Details zur Wirkung der Faktoren entnehmen Sie der nachfolgenden Tabelle:

Beeinflussende Faktoren:			Ergebnis:
Konfiguration - Anfahrrichtung	Konfiguration - Referenzpunktschalter	Aktuelle Fahrtrichtung	Referenzieren am Referenzpunktschalter
Positiv	"Untere Seite"	Positive Richtung	Obere Seite
		Negative Richtung	Untere Seite
Positiv	"Obere Seite"	Positive Richtung	Untere Seite
		Negative Richtung	Obere Seite
Negativ	"Untere Seite"	Positive Richtung	Untere Seite
		Negative Richtung	Obere Seite
Negativ	"Obere Seite"	Positive Richtung	Obere Seite
		Negative Richtung	Untere Seite

Geschwindigkeit (nur aktives Referenzieren)

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Anfahrgeschwindigkeit ≤ Maximale Geschwindigkeit

Referenziergeschwindigkeit (nur aktives Referenzieren)

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher die Achse in den Referenzpunktschalter zum Referenzieren einfahren soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Referenziergeschwindigkeit ≤ Maximale Geschwindigkeit

Referenzpunktverschiebung (nur aktives Referenzieren)

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- -1.0e12 ≤ Referenzpunktverschiebung ≤ 1.0e12

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Konfiguration - Referenzieren (Technologieobjekt "Achse" V2...3)

Konfiguration - Referenzieren - Allgemein (Technologieobjekt Achse V2...3)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Allgemein" den Eingang des Referenzpunktschalters für das aktive und das passive Referenzieren.

Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

Pegelauswahl

Wählen Sie in der Klappliste den Pegel des Referenzpunktschalters, mit dem referenziert werden soll.

Siehe auch

Ablauf - Aktives Referenzieren (Seite 1420)

Konfiguration - Referenzieren - Passiv (Technologieobjekt Achse V2...3)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Passiv" die für das passive Referenzieren notwendigen Parameter.

Zum passiven Referenzieren muss die Bewegung anwenderseitig (z. B. durch einen Fahrauftrag der Achse) angestoßen werden. Das passive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 2 gestartet.

Seite des Referenzpunktschalters

Wählen Sie hier, ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Hinweis

Erfolgt das passive Referenzieren ohne Fahrauftrag der Achse (Achse im Stillstand), so wird mit der nächsten steigenden oder fallenden Flanke am Referenzpunktschalter referenziert.

Konfiguration - Referenzieren - Aktiv (Technologieobjekt Achse V2...3)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren Aktiv" die für das aktive Referenzieren notwendigen Parameter. Das aktive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 3 gestartet.

Erlaube Richtungsumkehr am Hardware-Endschalter

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein (mindestens der Hardware-Endschalter in Anfahrrichtung muss konfiguriert sein).

Wird der Hardware-Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.

Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

Hinweis

Stellen Sie nach Möglichkeit durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgeschwindigkeit gering.
- Vergrößern Sie die konfigurierte Beschleunigung/Verzögerung.
- Vergrößern Sie den Abstand zwischen dem Hardware-Endschalter und dem mechanischen Anschlag.

Anfahr-/Referenzierrichtung

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die Anfahrrichtung zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest mit welcher Fahrrichtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenziervorgang durchzuführen.

Seite des Referenzpunktschalters

Wählen Sie hier ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Geschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Anfahrgeschwindigkeit ≤ Maximale Geschwindigkeit

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher in den Referenzpunktschalter zum Referenzieren eingefahren soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Referenziergeschwindigkeit ≤ Maximale Geschwindigkeit

Referenzpunktverschiebung

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- -1.0e12 ≤ Referenzpunktverschiebung ≤ 1.0e12

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Technologieobjekt "Achse" V1...3)

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

Beschleunigung und Verzögerung

Sie können die Werte für Beschleunigung und Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsname>.Config.DynamicDefaults.Acceleration
  zum Ändern der Beschleunigung
- <Achsname>.Config.DynamicDefaults.Deceleration
  zum Ändern der Verzögerung

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts (Seite 1641) im Anhang.

Notstopp-Verzögerung

Sie können den Wert für die Notstopp-Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsname>.Config.DynamicDefaults.EmergencyDeceleration

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

ACHTUNG
Nach dem Ändern dieses Parameters müssen u. U. die Positionen der Hardware-Endschalter sowie weitere sicherheitsrelevante Einstellungen angepasst werden.

Ruckbegrenzung (ab Technologieobjekt Achse V2.0)

Sie können die Ruckbegrenzung auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren und den Wert für den Ruck ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsname>.Config.DynamicDefaults.JerkActive
  zum Aktivieren und Deaktivieren der Ruckbegrenzung
- <Achsname>.Config.DynamicDefaults.Jerk
  zum Ändern des Rucks

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

Siehe auch

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 1414)

V4

Konfiguration - Allgemein (Technologieobjekt Positionierachse V4)

Konfigurieren Sie im Konfigurationsfenster "Allgemein" die grundlegenden Eigenschaften des Technologieobjekts Positionierachse.

Achsname

Definieren Sie in diesem Feld den Namen der Achse, bzw. den Namen des Technologieobjekts Positionierachse. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

Hardware - Schnittstelle

Die Impulse werden über fest zugeordnete digitale Ausgänge an das Leistungsteil des Antriebs ausgegeben.

Bei CPUs mit Relais-Ausgängen kann das Impulssignal nicht auf diesen Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen.

Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, muss ein Signalboard mit digitalen Ausgängen verwendet werden.

Hinweis

Der PTO benötigt die Funktionalität eines schnellen Zählers (HSC). Dazu wird ein interner HSC verwendet, dessen Zählerstand nicht ausgewertet werden kann.

Impulsgenerator auswählen

Wählen Sie in der Klappliste den PTO (Pulse Train Output), über welchen die Impulse für die Ansteuerung der Schrittmotoren oder Servomotoren mit Impulsschnittstelle zur Verfügung gestellt werden sollen. Wurden in der Gerätekonfiguration die Impulsgeneratoren und die schnellen Zähler nicht anderweitig verwendet, so kann die Hardware-Schnittstelle automatisch konfiguriert werden. In diesem Fall wird der gewählte PTO in der Klappliste weiß hinterlegt angezeigt.

Schaltfläche "Gerätekonfiguration"

Über diese Schaltfläche gelangen Sie zur Parametrierung der Impulsoptionen in der Gerätekonfiguration der CPU.

Signalart

Wählen Sie in der Klappliste die gewünschte Signalart. Folgende Signalarten stehen zur Auswahl:

- **PTO (Impuls A und Richtung B)**
  Zur Ansteuerung des Schrittmotors wird ein Impulsausgang und ein Richtungsausgang verwendet.
- **PTO (Vorwärts zählen A, rückwärts zählen B)**
  Zur Ansteuerung des Schrittmotors wird je ein Impulsausgang zur Bewegung in positive Richtung und negative Richtung verwendet.
- **PTO (A/B phasenversetzt)**
  Beide Impuslausgänge für Phase A und für Phase B takten mit der gleichen Frequenz. Antriebsseitig wird die Periode der Impulsausgänge als Schritt ausgewertet.
  Die Phasenverschiebung zwischen Phase A und Phase B bestimmt die Bewegungsrichtung.
- **PTO (A/B phasenversetzt - vierfach)**
  Beide Impuslausgänge für Phase A und für Phase B takten mit der gleichen Frequenz. Antriebsseitig werden alle steigenden und alle fallenden Flanken von Phase A und Phase B als Schritt ausgewertet.
  Die Phasenverschiebung zwischen Phase A und Phase B bestimmt die Bewegungsrichtung.

Impulsausgang (Signalart "PTO (Impuls A und Richtung B)")

Wählen Sie in diesem Feld den gewünschten Ausgang für den Impulsausgang.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Richtungsausgang aktivieren (Signalart "PTO (Impuls A und Richtung B)")

Im Modus "pulse and direction" können Sie den Richtungsausgang deaktivieren, bzw. aktivieren. Sie können mit dieser Option die Bewegungsrichtung einschränken.

Richtungsausgang (Signalart "PTO (Impuls A und Richtung B)")

Wählen Sie in diesem Feld den gewünschten Ausgang für den Richtungsausgang.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Impulsausgang vorwärts (Signalart "PTO (Vorwärts zählen A, rückwärts zählen B)")

Wählen Sie in diesem Feld den gewünschten Impulsausgang für Bewegungen in positive Richtung.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Impulsausgang rückwärts (Signalart "PTO (Vorwärts zählen A, rückwärts zählen B)")

Wählen Sie in diesem Feld den gewünschten Impulsausgang für Bewegungen in negative Richtung.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Phase A (Signalarten "PTO (A/B phasenversetzt)" und "PTO (A/B phasenversetzt - vierfach)")

Wählen Sie in diesem Feld den gewünschten Impulsausgang für Phase A Signale.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Phase B (Signalarten "PTO (A/B phasenversetzt)" und "PTO (A/B phasenversetzt - vierfach)")

Wählen Sie in diesem Feld den gewünschten Impulsausgang für Phase B Signale.

Sie können den Ausgang über eine symbolische Adresse auswählen oder einer absoluten Adresse zuordnen.

Anwendereinheit

Wählen Sie in der Klappliste die gewünschte Einheit für das Maßsystem der Achse aus. Die gewählte Einheit wird für die weitere Konfiguration des Technologieobjekts Positionierachse und für die Anzeige der aktuellen Achsdaten verwendet.

Die Werte an den Eingangsparametern (Position, Distance, Velocity, ...) der Motion Control-Anweisungen beziehen sich ebenfalls auf diese Einheit.

Hinweis

Eine nachträgliche Änderung des Maßsystems kann u. U. nicht in allen Konfigurationsfenstern des Technologieobjekts korrekt umgerechnet werden. Kontrollieren Sie in diesem Fall die Konfiguration aller Achsparameter.

Gegebenenfalls müssen im Anwenderprogramm Werte an den Eingangsparametern der Motion Control-Anweisungen an die neue Maßeinheit angepasst werden.

Konfiguration - Antriebssignale (Technologieobjekt Positionierachse V4)

Konfigurieren Sie im Konfigurationsfenster "Antriebssignale" den Ausgang für die Antriebsfreigabe und den Eingang für die Rückmeldung "Antrieb Bereit" des Antriebs.

Die Antriebsfreigabe wird von der Motion Control-Anweisung "MC_Power" gesteuert und erteilt dem Antrieb die Leistungsfreigabe. Das Signal wird dem Antrieb über den zu konfigurierenden Ausgang zur Verfügung gestellt.

Ist der Antrieb nach Erhalt der Antriebsfreigabe bereit Bewegungen auszuführen, so meldet er "Antrieb Bereit" an die CPU. Das Signal "Antrieb Bereit" wird der CPU über den zu konfigurierenden Eingang zurückgemeldet.

Verfügt der Antrieb über keine derartigen Schnittstellen, so müssen die Parameter nicht konfiguriert werden. Wählen Sie in diesem Fall für den Bereit-Eingang den Wert TRUE.

Siehe auch

Konfiguration - Mechanik (Technologieobjekt Positionierachse V4) (Seite 1597)

Positionsgrenzen (Seite 1403)

Dynamik (Seite 1408)

Referenzieren (Technologieobjekt Positionierachse ab V2) (Seite 1416)

Konfiguration - Mechanik (Technologieobjekt Positionierachse V4)

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die mechanischen Eigenschaften Ihres Antriebs.

Impulse pro Motorumdrehung

Konfigurieren Sie in diesem Feld, wie viele Impulse der Motor für eine Motorumdrehung benötigt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- 0 < Impulse pro Motorumdrehung ≤ 2147483647

Weg pro Motorumdrehung

Konfigurieren Sie in diesem Feld, welche Wegstrecke die Mechanik Ihrer Anlage pro Motorumdrehung zurücklegt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- 0.0 < Weg pro Motorumdrehung ≤ 1.0e12

Erlaubte Drehrichtung (Technologieversion ab V4)

Konfigurieren Sie in diesem Feld, ob sich die Mechanik Ihrer Anlage in beide Richtungen bewegen soll, oder nur in positive oder negative Richtung.

Wenn Sie im Modus "PTO (Impuls A und Richtung B)" des Impulsgenerators den Richtungsausgang nicht aktiviert haben, ist die Auswahl auf die positive oder negative Richtung begrenzt.

Richtungssinn invertieren

Über das Optionskästchen "Richtungssinn invertieren" können Sie die Steuerung an die Richtungslogik des Antriebs anpassen.

Die Richtungslogik wird entsprechend dem ausgewählten Modus des Impulsgenerators invertiert:

- **PTO (Impuls A und Richtung B)**
  - 0 V am Richtungsausgang ⇒ positive Drehrichtung
  - 5 V/24 V am Richtungsausgang ⇒ negative Drehrichtung

  Die angegebene Spannung ist abhängig von der verwendeten Hardware. Die genannten Werte gelten nicht für die Differenzausgänge der CPU 1217.
- **PTO (Vorwärts zählen A, rückwärts zählen B)**
  Die Ausgänge "Impulsausgang rückwärts" und "Impulsausgang vorwärts" werden vertauscht.
- **PTO (A/B phasenversetzt)**
  Die Ausgänge "Phase A" und "Phase B" werden vertauscht.
- **"PTO (A/B phasenversetzt - vierfach)**
  Die Ausgänge "Phase A" und "Phase B" werden vertauscht.

Siehe auch

Konfiguration - Antriebssignale (Technologieobjekt Positionierachse V4) (Seite 1597)

Zusammenhang zwischen der Signalart und der Fahrtrichtung (Seite 1348)

Positionsgrenzen (Seite 1403)

Dynamik (Seite 1408)

Referenzieren (Technologieobjekt Positionierachse ab V2) (Seite 1416)

Konfiguration - Referenzieren - Passiv (Technologieobjekt Positionierachse V4)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Passiv" die für das passive Referenzieren notwendigen Parameter.

Zum passiven Referenzieren muss die Bewegung anwenderseitig (z. B. durch einen Fahrauftrag der Achse) angestoßen werden. Das passive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 2 gestartet.

Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

Pegelauswahl

Wählen Sie in der Klappliste den Pegel des Referenzpunktschalters, mit dem referenziert werden soll.

Seite des Referenzpunktschalters

Wählen Sie hier, ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Hinweis

Erfolgt das passive Referenzieren ohne Fahrauftrag der Achse (Achse im Stillstand), so wird mit der nächsten steigenden oder fallenden Flanke am Referenzpunktschalter referenziert.

Konfiguration - Referenzieren - Aktiv (Technologieobjekt Positionierachse V4)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Aktiv" die für das aktive Referenzieren notwendigen Parameter. Das aktive Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit verwendetem Eingangsparameter "Mode" = 3 gestartet.

Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

Pegelauswahl

Wählen Sie in der Klappliste den Pegel des Referenzpunktschalters, mit dem referenziert werden soll.

Richtungsumkehr am Hardware-Endschalter erlauben

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein (mindestens der Hardware-Endschalter in Anfahrrichtung muss konfiguriert sein).

Wird der Hardware-Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.

Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

Hinweis

Stellen Sie nach Möglichkeit durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgeschwindigkeit gering.
- Vergrößern Sie die konfigurierte Beschleunigung/Verzögerung.
- Vergrößern Sie den Abstand zwischen dem Hardware-Endschalter und dem mechanischen Anschlag.

Anfahr-/Referenzierrichtung

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die Anfahrrichtung zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest mit welcher Fahrrichtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenziervorgang durchzuführen.

Seite des Referenzpunktschalters

Wählen Sie hier ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

Anfahrgeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Anfahrgeschwindigkeit ≤ Maximale Geschwindigkeit

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher in den Referenzpunktschalter zum Referenzieren eingefahren werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Referenziergeschwindigkeit ≤ Maximale Geschwindigkeit

Referenzpunktverschiebung

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- -1.0e12 ≤ Referenzpunktverschiebung ≤ 1.0e12

Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC_Home" parametrierte Position verwendet.

Diagnose - Status- und Fehlerbits (Technologieobjekt "Achse" V1...3)

Mit der Diagnosefunktion "Status- und Fehlerbits" überwachen Sie im TIA-Portal die wichtigsten Status- und Fehlermeldungen der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Status- Fehlermeldungen haben folgende Bedeutung:

Status der Achse

Status	Beschreibung
Freigegeben	Die Achse ist freigegeben und bereit über Motion Control-Aufträge gesteuert zu werden. (Variable des Technologieobjekts: <Achsname>.StatusBits.Enable)
Referenziert	Die Achse ist referenziert und kann absolute Positionieraufträge der Motion Control-Anweisung "MC_MoveAbsolute" ausführen. Zum relativen Positionieren muss die Achse nicht referenziert sein. Sonderfälle: • Während des aktiven Referenzierens ist der Status FALSE. • Wird eine referenzierte Achse passiv referenziert, so ist der Status während des passiven Referenzierens TRUE. (Variable des Technologieobjekts: <Achsname>.StatusBits.HomingDone)
Achsfehler	Am Technologieobjekt "Achse" ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern ErrorID und ErrorInfo der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt. (Variable des Technologieobjekts: <Achsname>.StatusBits.Error)
Achssteuertafel aktiv	Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt "Achse". Die Achse kann nicht vom Anwenderprogramm gesteuert werden. (Variable des Technologieobjekts: <Achsname>.StatusBits.ControlPanelActive)
Restart erforderlich	Eine geänderte Konfiguration der Achse wurde im Betriebszustand RUN der CPU in den Ladespeicher geladen. Um die geänderte Konfiguration in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung MC_Reset.

Status des Antriebs

Status	Beschreibung
Antrieb bereit	Der Antrieb ist betriebsbereit. (Variable des Technologieobjekts: <Achsname>.StatusBits.DriveReady)
Antriebsfehler	Der Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt. (Variable des Technologieobjekts: <Achsname>.ErrorBits.DriveFault)

Status der Achsbewegung

Status	Beschreibung
Stillstand	Die Achse befindet sich im Stillstand. (Variable des Technologieobjekts: <Achsname>.StatusBits.StandStill)
Beschleunigung	Die Achse beschleunigt. (Variable des Technologieobjekts: <Achsname>.StatusBits.Acceleration)
Konstante Geschwindigkeit	Die Achse verfährt mit konstanter Geschwindigkeit. (Variable des Technologieobjekts: <Achsname>.StatusBits.ConstantVelocity)
Verzögerung	Die Achse verzögert (bremst ab). (Variable des Technologieobjekts: <Achsname>.StatusBits.Deceleration)

Status der Bewegungsart

Status	Beschreibung
Positionieren	Die Achse führt einen Positionierauftrag der Motion Control-Anweisung "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.PositioningCommand)
Mit Geschwindigkeitsvorgabe fahren	Die Achse führt einen Auftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisung "MC_MoveVelocity", "MC_MoveJog" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.SpeedCommand)
Referenzieren	Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus. (Variable des Technologieobjekts: <Achsname>.StatusBits.Homing)
Auftragstabelle aktiv (ab Technologieobjekt Achse V2.0)	Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert. (Variable des Technologieobjekts: <Achsname>.StatusBits.CommandTableActive)

Fehlermeldungen

Fehler	Beschreibung
Unterer SW-Endschalter wurde erreicht	Der untere Software-Endschalter wurde erreicht. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SwLimitMinReached)
Unterer SW-Endschalter wurde überschritten	Der untere Software-Endschalter wurde unterschritten. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SwLimitMinExceeded)

Fehler	Beschreibung
Oberer SW-Endschalter wurde erreicht	Der obere Software-Endschalter wurde erreicht. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SwLimitMaxReached)
Oberer SW-Endschalter wurde überschritten	Der obere Software-Endschalter wurde überschritten. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SwLimitMaxExceeded)
Unterer HW-Endschalter wurde angefahren	Der untere Hardware-Endschalters wurde angefahren. (Variable des Technologieobjekts: <Achsname>.ErrorBits.HwLimitMin)
Oberer HW-Endschalter wurde angefahren	Der obere Hardware-Endschalters wurde angefahren. (Variable des Technologieobjekts: <Achsname>.ErrorBits.HwLimitMax)
PTO und HSC bereits in Verwendung	Eine zweite Achse verwendet den selben PTO (Pulse Train Output) und HSC (High Speed Counter) und ist mit "MC_Power" freigegeben. (Variable des Technologieobjekts: <Achsname>.ErrorBits.HwUsed)
Konfigurationsfehler	Das Technologieobjekt "Achse" wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (Variable des Technologieobjekts: <Achsname>.ErrorBits.ConfigFault)
Interner Fehler	Es ist ein interner Fehler aufgetreten. (Variable des Technologieobjekts: <Achsname>.ErrorBits.SystemFault)

Siehe auch

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

ErrorIDs und ErrorInfos

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

Bei Betriebsfehlern mit Stopp der Achse wird die Achse abhängig von der Fehlerreaktion gestoppt. Folgende Fehlerreaktionen sind möglich:

- **Freigabe wegnehmen**
  Der Sollwert null wird ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
- **Stopp mit Notstopp-Rampe**
  Laufende Bewegungsbefehle werden abgebrochen. Die Achse wird mit der unter "Technologieobjekt > Erweiterte Parameter > Dynamik > Notstopp" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht.

Betriebsfehler mit Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8000		**Antriebsfehler, "Antrieb bereit" ausgefallen**		-
	16#0001	-	Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten	
16#8001		**Unterer SW-Endschalter wurde ausgelöst**		-
	16#000E	Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren	
	16#000F	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht		
	16#0010	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten		Freigabe wegnehmen
16#8002		**Oberer SW-Endschalter wurde ausgelöst**		-
	16#000E	Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren	
	16#000F	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht		
	16#0010	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten		Freigabe wegnehmen
16#8003		**Unterer HW-Endschalter wurde angefahren**		Bei Antriebsanbindung über PTO (Pulse Train Output): Stopp mit Notstopp-Rampe Bei Antriebsanbindung über PROFIdrive / Analogausgang: Freigabe wegnehmen
	16#000E	Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden.)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#8004		Oberer HW-Endschalter wurde angefahren		Bei Antriebsanbindung über PTO (Pulse Train Output): Stopp mit Notstopp-Rampe Bei Antriebsanbindung über PROFIdrive / Analogausgang: Freigabe wegnehmen
	16#000E	Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden.)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.	
16#8005		PTO / HSC werden bereits von einer anderen Achse verwendet		-
	16#0001	-	**Die Achse wurde fehlerhaft konfiguriert:** Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden	
			Mehrere Achsen sollen mit einem PTO arbeiten: Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden. (siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504))	
16#8006		Ein Kommunikationsfehler ist in der Achssteuertafel eingetreten		Freigabe wegnehmen
	16#0012	Ein Zeitüberlauf ist eingetreten.	Prüfen Sie die Kabelverbindung; betätigen Sie erneut die Schaltfläche "Handsteuerung"	
16#8007		Freigabe der Achse nicht möglich		-
	16#0025	Restart wird ausgeführt	Warten Sie, bis der Restart der Achse abgeschlossen ist.	
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie, bis der Ladevorgang abgeschlossen ist.	
16#8008		Unzulässige Bewegungsrichtung		-
	16#002E	Die gewählte Bewegungsrichtung ist nicht erlaubt.	• Passen Sie die Bewegungsrichtung an und starten Sie den Auftrag erneut. • Passen Sie in der Konfiguration des Technologieobjekts unter "Erweiterte Parameter > Mechanik" die erlaubte Drehrichtung an. Starten Sie den Auftrag erneut.	
	16#002F	Eine reversierende Bewegung ist mit der gewählten Bewegungsrichtung nicht möglich.		
16#8009		Referenzschalter/Gebernullmarke nicht gefunden		Stopp mit Notstopp-Rampe
	16#0033	Fehler in der Konfiguration, Hardware oder Montage des Gebers oder am Referenzpunktschalter.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#800A		Störungsmeldung vom Geber		Freigabe wegnehmen
	16#0001	-	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse.	
	16#0034	Hardware-Fehler am Geber		
	16#0035	Geber verschmutzt		
	16#0036	Fehler beim Lesen des Geberabsolutwerts	Vergleichen Sie den Gebertyp im Antriebs- bzw. Geberparameter P979 mit den Konfigurationsdaten des Technologieobjekts.	
	16#0037	Nullmarkenüberwachung des Gebers	Geber meldet Fehler bei der Nullmarkenüberwachung (Störcode 0x0002 im Gx_XIST2, siehe PROFIdrive-Profil). Prüfen Sie die Anlage auf elektromagnetische Verträglichkeit (EMV).	
	16#0038	Geber ist im Zustand "Parken"	• Suchen Sie die Fehlerursache im angeschlossenen Antrieb bzw. Geber. • Prüfen Sie, ob die Fehlermeldung gegebenenfalls infolge einer Inbetriebnahmehandlung am Antrieb bzw. Geber ausgelöst wurde.	
	16#0040	PROFIdrive: Geber am Bus ausgefallen (Stationsausfall)	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse.	
	16#0041	PROFIdrive: Lebenszeichen des Gebers gestört		
16#800B		Bereichsüberschreitung der Position		Freigabe wegnehmen
	16#0039	Bereichsüberschreitung in positiver Richtung	Referenzieren Sie die Achse in einen gültigen Istwertbereich.	
	16#003A	Bereichsüberschreitung in negativer Richtung		
	16#003B	Die Veränderung der Istposition ist in einem Lageregler-Takt grösser als die Modulolänge.	Passen Sie die Modulolänge dem verwendeten Geber an.	

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#800C		**Störungsmeldung vom Antrieb**		Freigabe wegnehmen
	16#0001	-	Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse. Wählen Sie im Dialog "Optimierung" eine kleinere Verstärkung (Kv).	
	16#003C	PROFIdrive: Antriebssignal "Führung gefordert" ausgefallen		
	16#003D	PROFIdrive/Analoge Antriebsanbindung: Antrieb hat sich abgeschaltet		
	16#003E	PROFIdrive: Antrieb am Bus ausgefallen (Stationsausfall)		
	16#003F	PROFIdrive: Lebenszeichen des Antriebs gestört	• Prüfen Sie das Gerät bzgl. der Funktion, der Verbindungen und der Anschlüsse. • Vergleichen Sie die Taktparameter von HW Konfig (PROFIBUS-Strang, Slave-OM für Antrieb bzw. Geber) und des Ablaufsystems. Tmapc und Servo müssen auf die gleiche Taktperiodenzeit parametriert sein.	
16#800D		**Der zulässige Schleppfehler wurde überschritten**		Freigabe wegnehmen
	16#0001	-	• Prüfen Sie die Konfiguration des Regelkreises. • Prüfen Sie den Richtungssinn des Gebers. • Prüfen Sie Konfiguration der Schleppfehlerüberwachung.	
16#800E		**Fehler am HW-Endschalter**		Freigabe wegnehmen
	16#0042	Unzulässige Freifahrrichtung bei aktivem HW-Endschalter	Die programmierte Bewegungsrichtung ist wegen des aktiven HW-Endschalters gesperrt. Fahren Sie die Achse in entgegengesetzter Richtung frei.	
	16#0043	HW-Endlagenschalter verpolt, kein Freifahren möglich	Prüfen Sie den mechanischen Aufbau der HW-Endschalter.	
	16#0044	Beide HW-Endschalter sind aktiv, kein Freifahren möglich		

ErrorID	ErrorInfo	Beschreibung	Abhilfe	Fehlerreaktion
16#800F		**Fehler am Zielbereich**		Freigabe wegnehmen
	16#0045	Zielbereich nicht erreicht	Der Zielbereich wurde nicht innerhalb der Positioniertoleranzzeit erreicht. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.	
	16#0046	Zielbereich wieder verlassen	Der Zielbereich wurde innerhalb der Mindestverweildauer verlassen. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.	
16#8010		**Position des unteren SW-Endschalters größer als die Position des oberen SW-Endschalters, wenn die Achse keine Moduloachse ist**		Freigabe wegnehmen
	16#0001	-	Ändern Sie die Position der SW-Endschalter.	

Betriebsfehler ohne Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8200		**Achse ist nicht freigegeben**	
	16#0001	-	Achse freigeben; Auftrag erneut starten
16#8201		**Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben**	
	16#0001	-	Achse nur über eine "MC_Power"-Instanz freigeben
16#8202		**Maximale Anzahl gleichzeitig aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für Antriebsanbindung über PTO (Pulse Train Output), max. 100 Aufträge für Antriebsanbindung über PROFIdrive / Analogausgang)**	
	16#0001	-	Anzahl der gleichzeitig aktiven Aufträge verringern; Auftrag erneut starten Ein aktiver Auftrag ist am Parameter "Busy" = TRUE der Motion Control-Anweisung zu erkennen.
16#8203		**Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben**	
	16#0001	-	"Handsteuerung" beenden; Auftrag erneut starten
16#8204		**Achse ist nicht referenziert**	
	16#0001	-	Achse mit der Anweisung "MC_Home" referenzieren; Auftrag erneut starten
16#8205		**Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der Achssteuertafel angezeigt)**	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren und in der Achssteuertafel erneut "Handsteuerung" anwählen.
16#8206		**Technologieobjekt ist noch nicht aktiviert**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0001	-	Aktivieren Sie die Achse mit der Anweisung "MC_Power" Enable = TRUE oder geben Sie die Achse in der Achssteuertafel frei.
16#8207		Auftrag abgewiesen	
	16#0016	Aktives Referenzieren läuft; eine andere Referenzierart kann nicht gestartet werden.	Warten Sie den Abschluss des aktiven Referenzierens ab oder brechen Sie das aktive Referenzieren durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
	16#0018	Die Achse kann nicht mit einer Auftragstabelle verfahren werden, während die Achse direkt oder passiv referenziert wird.	Warten Sie den Abschluss des direkten oder passiven Referenzierens ab.
	16#0019	Während der Abarbeitung einer Auftragstabelle, kann die Achse nicht direkt oder passiv referenziert werden.	Warten Sie den Abschluss der Auftragstabelle ab oder brechen Sie die Auftragstabelle durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
	16#0052	Die angegebene Position überschreitet die nummerische Zahlengrenze.	Geben Sie an der Motion Control-Anweisung einen gültigen Positionswert an.
	16#0053	Die Achse befindet sich im Hochlauf.	Warten Sie, bis die Achse betriebsbereit ist.
	16#0054	Istwert nicht gültig	Um einen "MC_Home"-Auftrag auszuführen, müssen die Istwerte gültig sein. Prüfen Sie den Status der Istwerte. Die Variable des Technologieobjekts <Achsname>.StatusSensor.State muss den Wert 2 (gültig) zeigen.
16#8208		Geschwindigkeitsdifferenz zwischen maximaler Geschwindigkeit und Start/Stopp-Geschwindigkeit ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#8209		Wert der Beschleunigung des Technologieobjekts "Achse" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#820A		Restart der Achse nicht möglich	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren; Restart erneut ausführen
	16#0027	Die Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben	"Handsteuerung" beenden; Restart erneut ausführen
	16#002C	Die Achse ist nicht gesperrt.	Achse sperren; Auftrag erneut starten
	16#0047	Das Technologieobjekt ist für den Restart nicht bereit.	Laden Sie das Projekt neu.
	16#0048	Bedingung für den Restart des Technologieobjekts ist nicht erfüllt.	Sperren Sie das Technologieobjekt.
16#820B		Ausführen der Auftragstabelle nicht möglich	
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie, bis der Ladevorgang abgeschlossen ist.
16#820C		Keine Konfiguration verfügbar	
	16#0001	-	Interner Fehler Wenden Sie sich an die Hotline.

Bausteinparameterfehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8400		**Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
16#8401		**Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
16#8402		**Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
	16#0024	Wert ist kleiner 0	
16#8403		**Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
16#8404		**Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
	16#0015	Aktives / Passives Referenzieren ist nicht konfiguriert	Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0017	Die Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind	• Aktivieren der HW-Endschalter über die Variable <Achsname>.PositionLimitsHW.Active = TRUE, Auftrag erneut starten • Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0055	Ungültiger Mode bei Inkrementalwertgeber	Starten Sie einen Referenziervorgang für einen Inkrementalgeber mit Parameter "Mode" = 0, 1, 2, 3.
	16#0056	Ungültiger Mode bei Absolutwertgeber	Bei einem Absolutwertgeber ist passives und aktives Referenzieren ("Mode" = 2, 3) nicht möglich. Starten Sie einen Referenziervorgang für einen Absolutwertgeber mit Parameter "Mode" = 0, 1.
16#8405		**Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Achse erneut freigeben
16#8406		**Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt**	
	16#0001	-	Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8407		**Wechsel der Achse an der Anweisung "MC_Power" nur bei gesperrter Achse zulässig.**	
	16#0001	-	Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.
16#8408		**Wert am Parameter "Axis" der Motion Control-Anweisung ist ungültig**	
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#8409		**Wert am Parameter "CommandTable" der Motion Control-Anweisung ist ungültig**	
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#840A		**Wert am Parameter "StartStep" der Motion Control-Anweisung ist ungültig**	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001D	Der Startschritt ist größer als der Endschritt	
	16#001E	Wert ist größer 32	
16#840B		**Wert am Parameter "EndStep" der Motion Control-Anweisung ist ungültig**	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001E	Wert ist größer 32	
16#840C		**Wert am Parameter "RampUpTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840D		**Wert am Parameter "RampDownTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840E		**Wert am Parameter "EmergencyRampTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840F		**Wert am Parameter "JerkTime" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#8410		**Wert am Parameter "Parameter" der Motion Control-Anweisung ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000B	Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0028	Datentyp des VARIANT-Zeigers "Parameter" und "Value" passen nicht zueinander.	Geigneten Datentyp verwenden; Auftrag erneut starten
	16#0029	VARIANT-Zeiger "Parameter" zeigt nicht auf einen Datenbaustein des Technologieobjekts.	VARIANT-Zeiger korrigieren; Auftrag erneut starten
	16#002A	Der Wert am VARIANT-Zeiger "Parameter" kann nicht gelesen werden.	VARIANT-Zeiger korrigieren; Auftrag erneut starten
	16#002B	Der Wert am VARIANT-Zeiger "Parameter" kann nicht geschrieben werden.	VARIANT-Zeiger oder Wert korrigieren; Auftrag erneut starten
	16#002C	Die Achse ist nicht gesperrt.	Achse sperren; Auftrag erneut starten
16#8411		**Wert am Parameter "Value" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten

Konfigurationsfehler der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8600		**Parametrierung des Impulsgenerators (PTO) ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8601		**Parametrierung des schnellen Zählers (HSC) ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8602		**Parametrierung des "Freigabe-Ausgangs" ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8603		**Parametrierung des "Bereit-Eingangs" ist ungültig**	
	16#000B	Die Adresse ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8604		**Wert für "Impulse pro Motorumdrehung" ist ungültig**	
	16#000A	Wert ist kleiner oder gleich Null	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8605		**Wert für "Weg pro Motorumdrehung" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#000A	Wert ist kleiner oder gleich Null	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8606		**Wert für "Start/Stopp-Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0007	Die Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit	
16#8607		**Wert für "maximale Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8608		**Wert für "Beschleunigung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8609		**Wert für "Verzögerung" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#860A		Wert für "Notstopp-Verzögerung" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#860B		Wert für die Position des unteren SW-Endschalters ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
	16#0030	Der Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters	
16#860C		Wert für die Position des oberen SW-Endschalters ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#860D		**Adresse des unteren HW-Endschalters ist ungültig**	
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860E		**Adresse des oberen HW-Endschalter ist ungültig**	
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860F		**Wert für "Referenzpunktverschiebung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8610		**Wert für "Anfahrgeschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8611		**Wert für "Referenziergeschwindigkeit" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8612		Adresse des Referenzpunktschalters ist ungültig	
	16#000B	Adresse ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000C	Die Adresse der fallenden Flanke ist ungültig	
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#8613		Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8614		Wert für "Ruck" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#001F	Wert ist größer als der maximal zulässige Ruck	
	16#0020	Wert ist kleiner als der minimal zulässige Ruck	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8615		Wert für "Maßeinheit" ist ungültig	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8616		Adresse des Referenzpunktschalters ist ungültig (passives Referenzieren ab V4)	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8617		**Wert der Variable <Achsname>.Sensor.Sensor[1].ActiveHoming.Mode ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Referenzieren über Digitaleingang)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8618		**Wert der Variable <Achsname>.Sensor.Sensor[1].PassiveHoming.Mode ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Referenzieren über Digitaleingang)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8619		**Wert der Variable <Achsname>.Actor.Type ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig (Gültiger Wert: 2 = Anbindung über Impulsschnittstelle)	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861A		**Wert für "Erlaubte Drehrichtung" ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#002D	"Beide Richtungen" bei abgeschaltetem Richtungsausgang nicht erlaubt	
16#861B		**Lastgetriebefaktoren fehlerhaft**	
	16#0031	Wert ist ungültig.	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#861C		**Unzulässige Kombination der Daten für Referenzieren beim inkrementellen Geber**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0031	Wert ist ungültig.	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861D		**Die eingestellte Gebersanbauart ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].MountingMode**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861E		**Die Konfiguration des Messradumfangs des Gebers ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.DistancePerRevolution**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#861F		**Die Konfiguration der Auflösung des Lineargebers ist fehlerhaft. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.Resolution**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8620		**Die eingestellte Feinauflösung für Gn_XIST1 ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist1**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8621		**Die eingestellte Feinauflösung für Gn_XIST1 in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist1 ist nicht konsistent zu der Einstellung im PROFIdrive-Parameter P979**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8622		**Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Actor.Interface.AddressIn bzw. <Achsname>.Actor.Interface.AddressOut**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8623		**Der eingestellte Wert in der Variable <Achsname>.Sensor.Sensor[1].Type ist ungültig.**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8624		**Das eingestellte Gebersystem ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].System**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8625		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.MinDwellTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8626		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.Window**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8627		**Die Konfiguration der PROFIdrive-Schnittstelle des Istwerts ist fehlerhaft. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Interface.AddressIn bzw. <Achsname>.Sensor.Sensor[1].Interface.AddressOut**	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8628		**Reglerfaktoren fehlerhaft**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Der Wert für die Verstärkung bzw. die Vorsteuerung des Regelkreises ist fehlerhaft. • Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten >.PositionControl.Kv, <Achsname>.PositionControl.Kpc)

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8629		**Grenzwert für Stillstandssignal ist fehlerhaft. Unzulässiger Wert in <Achsname>.StandStillSignal.VelocityThreshold**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862A		**Parameter der Positionierüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.PositioningMonitoring.ToleranceTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862B		**Inkonsistente PROFIBUS-Parametrierung; die Summe aus Ti und To ist grösser als ein DP-Takt**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#862C		**Parameter der Stillstandsüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.StandStillSignal.MinDwellTime**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862D		**Parameter der Schleppabstandsüberwachung ist fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MinValue**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862E		**Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Modulo.Length**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#862F		**Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Modulo.StartValue**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8630		**Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Actor.DriveParameter.ReferenceSpeed**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8631		**Die eingestellte Feinauflösung für Gn_XIST2 ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.FineResolutionXist2**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8632		**Die Anzahl der auflösbaren Geberumdrehungen ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].Parameter.DeterminableRevolutions**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8633		**Die eingestellte Anfahrrichtung des Referenzpunktschalters für das passive Referenzieren ist ungültig. Unzulässiger Wert in <Achsname>.Sensor.Sensor[1].PassiveHoming.Direction**	
16#8634		**Parameter der Schleppabstandsüberwachung fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MaxValue**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8635		**Parameter der Schleppabstandsüberwachung fehlerhaft. Unzulässiger Wert in <Achsname>.FollowingError.MinVelocity**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8636		Reglerfaktor fehlerhaft. Unzulässiger Wert des Vorsteuerfaktors <Achsname>.PositionControl.Kpc	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8637		Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Sensor.Sensor[1].Interface.Type	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8638		Unzulässiger Wert das Konfgurationsdatum <Achsname>.Sensor.Sensor[1].Interface.HSC	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8639		Fehler am Antrieb	
	16#0049	Konfigurationsfehler am Gerät	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#004A	Die Technologie benötigt einen kleineren Servo-Takt	Interner Systemfehler. Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
	16#004B	Gerätetreiber wurde im Hochlauf nicht initialisiert	Um ein Technologieobjekt freizugeben, muss der Aktortreiber fertig initialisiert sein. Setzen Sie den Auftrag später erneut ab.
16#863A		Kommunikation zum Antrieb fehlerhaft	
	16#004C	Konfigurationsfehler am Gerät	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004D	Gerätereiber benötigt einen kleineren Servo-Takt	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004E	Fehler in der internen Kommunikation zum Gerät	Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
16#863B		Fehler am Geber	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0049	Konfigurationsfehler am Gerät	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#004A	Die Technologie benötigt einen kleineren Servo-Takt	Interner Systemfehler. Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
	16#004B	Gerätetreiber wurde im Hochlauf nicht initialisiert	Um ein Technologieobjekt freizugeben, muss der Aktortreiber fertig initialisiert sein. Setzen Sie den Auftrag später erneut ab.
16#863C		**Kommunikation zum Geber fehlerhaft**	
	16#004C	Konfigurationsfehler am Gerät	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004D	Gerätereiber benötigt einen kleineren Servo-Takt	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#004E	Fehler in der internen Kommunikation zum Gerät	Prüfen Sie das Projekt auf Konsistenz und laden Sie es erneut in die Steuerung.
16#863D		**Kommunikation zum Gerät (Antrieb oder Geber) ist gestört**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0055	Die angeforderte logische Adresse ist ungültig.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie die topologische Konfiguration in HW Konfig. • Vergleichen Sie die Konfiguration von HW Konfig und Technologieobjekt.
	16#0056	Die angeforderte logische Input-Adresse ist ungültig.	
	16#0057	Die angeforderte logische Output-Adresse ist ungültig.	
16#863E		**Wert der Variable "ControlPanel.Input.TimeOut" ist ungültig (Achssteuertafel)**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Wert in der Variablen des Technologieobjekts <Achsname>.ControlPanel.Input.TimeOut. Der Wert wird in Millisekunden (ms) angegeben.
16#863F		**Unzulässiger Wert für das Konfigurationsdatum <Achsname>.Actor.DriveParameter.MaxSpeed**	
	16#0030	Wert hat ein falsches Zahlenformat oder ist außerhalb des gültigen Zahlenbereichs	Korrigieren Sie den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf Actor.MaxSpeed/2. Korrigieren Sie bei analoger Antriebsanbindung den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf Actor.MaxSpeed/1,17.

Konfigurationsfehler der Auftragstabelle

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8700		**Wert für "Auftragstyp" in der Auftragstabelle ist ungültig**	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
16#8701		**Wert für "Position / Verfahrweg" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1E+12)	
16#8702		**Wert für "Geschwindigkeit" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
16#8703		**Wert für "Dauer" in der Auftragstabelle ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0021	Wert ist größer als 64800 s	
	16#0022	Wert ist kleiner als 0.001 s	
16#8704		**Wert für "Nächster Schritt" in der Auftragstabelle ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0023	Der Auftragsübergang ist für diesen Auftrag nicht erlaubt	

Interne Fehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8FFF		**Interner Fehler**	
	16#F0**	-	NETZ-AUS und NETZ-EIN der CPU Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit: • ErrorID • ErrorInfo • Diagnosepuffereinträge

Siehe auch

Fehleranzeigen der Motion Control-Anweisungen (Seite 1495)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

ErrorIDs und ErrorInfos (Seite 1604)

Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)

Mehrere Antriebe mit gleichem PTO einsetzen (Seite 1507)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 1508)

Sonderfälle beim Einsatz von Software-Endschaltern bei Antriebsanbindung über PTO (Seite 1510)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V2...3)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

Betriebsfehler mit Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8000		**Antriebsfehler, "Antrieb bereit" ausgefallen**	
	16#0001	-	Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten
16#8001		**Unterer SW-Endschalter wurde ausgelöst**	
	16#000E	Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren
	16#000F	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht	
	16#0010	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten	
16#8002		**Oberer SW-Endschalter wurde ausgelöst**	
	16#000E	Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren
	16#000F	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht	
	16#0010	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten	
16#8003		**Unterer HW-Endschalter wurde angefahren**	
	16#000E	Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8004		**Oberer HW-Endschalter wurde angefahren**	
	16#000E	Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.
16#8005		**PTO / HSC werden bereits von einer anderen Achse verwendet**	
	16#0001	-	**Die Achse wurde fehlerhaft konfiguriert:** Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden
			Mehrere Achsen sollen mit einem PTO arbeiten: Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden. (siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504))
16#8006		**Es ist ein Kommunikationsfehler in der Achssteuertafel eingetreten**	
	16#0012	Es ist ein Zeitüberlauf eingetreten	Prüfen Sie die Kabelverbindung; betätigen Sie erneut die Schaltfläche "Handsteuerung"
16#8007		**Freigabe der Achse nicht möglich**	
	16#0025	Restart wird ausgeführt	Warten Sie bis der Restart der Achse abgeschlossen ist.
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie bis der Ladevorgang abgeschlossen ist

Betriebsfehler ohne Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8200		**Achse ist nicht freigegeben**	
	16#0001	-	Achse freigeben; Auftrag erneut starten
16#8201		**Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben**	
	16#0001	-	Achse nur über eine "MC_Power"-Instanz freigeben
16#8202		**Maximale Anzahl gleichzeitig aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für alle Motion Control-Technologieobjekte)**	
	16#0001	-	Anzahl der gleichzeitig aktiven Aufträge verringern; Auftrag erneut starten Ein aktiver Auftrag ist am Parameter "Busy" = TRUE der Motion Control-Anweisung zu erkennen.
16#8203		**Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben**	
	16#0001	-	"Handsteuerung" beenden; Auftrag erneut starten
16#8204		**Achse ist nicht referenziert**	
	16#0001	-	Achse mit der Anweisung "MC_Home" referenzieren; Auftrag erneut starten

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8205		**Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der Achssteuertafel angezeigt)**	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren und in der Achssteuertafel erneut "Handsteuerung" anwählen
16#8206		**Technologieobjekt ist noch nicht aktiviert**	
	16#0001	-	Aktivieren Sie die Achse mit der Anweisung "MC_Power" Enable = TRUE, oder geben Sie die Achse in der Achssteuertafel frei.
16#8207		**Auftrag abgewiesen**	
	16#0016	Aktives Referenzieren läuft; eine andere Referenzierart kann nicht gestartet werden.	Warten Sie den Abschluss des aktiven Referenzierens ab oder brechen Sie das aktive Referenzieren durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
	16#0018	Die Achse kann nicht mit einer Auftragstabelle verfahren werden, während die Achse direkt oder passiv referenziert wird.	Warten Sie den Abschluss des direkten oder passiven Referenzierens ab.
	16#0019	Während der Abarbeitung einer Auftragstabelle, kann die Achse nicht direkt oder passiv referenziert werden.	Warten Sie den Abschluss der Auftragstabelle ab oder brechen Sie die Auftragstabelle durch einen Bewegungsauftrag z. B. "MC_Halt" ab.
16#8208		**Geschwindigkeitsdifferenz zwischen maximaler Geschwindigkeit und Start/Stopp-Geschwindigkeit ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#8209		**Wert der Beschleunigung des Technologieobjekts "Achse" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#820A		**Restart der Achse nicht möglich**	
	16#0013	Die Achse ist im Anwenderprogramm freigegeben	Achse mit der Anweisung "MC_Power" sperren; Restart erneut ausführen
	16#0027	Die Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben	"Handsteuerung" beenden; Restart erneut ausführen
16#820B		**Ausführen der Auftragstabelle nicht möglich**	
	16#0026	Ladevorgang im Betriebszustand RUN wird ausgeführt	Warten Sie bis der Ladevorgang abgeschlossen ist

Bausteinparameterfehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8400		**Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12)	
16#8401		**Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12)	
16#8402		**Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
	16#0024	Wert ist kleiner 0	
16#8403		**Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
16#8404		**Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Auftrag erneut starten
	16#0015	Aktives / Passives Referenzieren ist nicht konfiguriert	Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0017	Die Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind	• Aktivieren der HW-Endschalter über die Variable <Achse>.Config.PositionLimits_HW.Active = TRUE, Auftrag erneut starten • Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
16#8405		**Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	Auswahlwert korrigieren; Achse erneut freigeben
16#8406		**Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt**	
	16#0001	-	Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.
16#8407		**Wechsel der Achse an der Anweisung "MC_Power" nur bei gesperrter Achse zulässig.**	
	16#0001	-	Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.
16#8408		**Wert am Parameter "Axis" der Motion Control-Anweisung ist ungültig**	
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#8409		**Wert am Parameter "CommandTable" der Motion Control-Anweisung ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#001A	Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts	Wert korrigieren; Auftrag erneut starten
	16#001B	Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts	
	16#001C	Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein	
16#840A		Wert am Parameter "StartStep" der Motion Control-Anweisung ist ungültig	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001D	Der Startschritt ist größer als der Endschritt	
	16#001E	Wert ist größer 32	
16#840B		Wert am Parameter "EndStep" der Motion Control-Anweisung ist ungültig	
	16#000A	Wert ist kleiner oder gleich 0.	Wert korrigieren; Auftrag erneut starten
	16#001E	Wert ist größer 32	
16#840C		Wert am Parameter "RampUpTime" der Motion Control-Anweisung ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840D		Wert am Parameter "RampDownTime" der Motion Control-Anweisung ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840E		Wert am Parameter "EmergencyRampTime" der Motion Control-Anweisung ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	
16#840F		Wert am Parameter "JerkTime" der Motion Control-Anweisung ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert korrigieren; Auftrag erneut starten
	16#000A	Wert ist kleiner oder gleich 0.	

Konfigurationsfehler der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8600		Parametrierung des Impulsgenerators (PTO) ist ungültig	
	16#000B	Die Adresse ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8601		Parametrierung des schnellen Zählers (HSC) ist ungültig	
	16#000B	Die Adresse ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0014	Die ausgewählte Hardware wird von einer anderen Anwendung verwendet	
16#8602		Parametrierung des "Freigabe-Ausgangs" ist ungültig	
	16#000B	Die Adresse ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8603		Parametrierung des "Bereit-Eingangs" ist ungültig	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#000B	Die Adresse ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8604		**Wert für "Impulse pro Motorumdrehung" ist ungültig**	
	16#000A	Wert ist kleiner oder gleich Null	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
16#8605		**Wert für "Weg pro Motorumdrehung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#000A	Wert ist kleiner oder gleich Null	
16#8606		**Wert für "Start/Stopp-Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
	16#0007	Die Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit	
16#8607		**Wert für "maximale Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
16#8608		**Wert für "Beschleunigung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
16#8609		**Wert für "Verzögerung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
16#860A		**Wert für "Notstopp-Verzögerung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die obere Hardwaregrenze	
	16#0004	Wert ist kleiner als die untere Hardwaregrenze	
16#860B		**Wert für die Position des unteren SW-Endschalters ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12)	
	16#0007	Der Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters	
16#860C		Wert für die Position des oberen SW-Endschalters ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12)	
16#860D		Adresse des unteren HW-Endschalters ist ungültig	
	16#000C	Die Adresse der fallenden Flanke ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860E		Adresse des oberen HW-Endschalter ist ungültig	
	16#000C	Die Adresse der fallenden Flanke ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#860F		Wert für "Referenzpunktverschiebung" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12)	
16#8610		Wert für "Anfahrgeschwindigkeit" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
16#8611		Wert für "Referenziergeschwindigkeit" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0008	Die Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Die Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
16#8612		Adresse des Referenzpunktschalters ist ungültig	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#000C	Die Adresse der fallenden Flanke ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben
	16#000D	Die Adresse der steigenden Flanke ist ungültig	
16#8613		Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
16#8614		Wert für "Ruck" ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#001F	Wert ist größer als der maximal zulässige Ruck	
	16#0020	Wert ist kleiner als der minimal zulässige Ruck	
16#8615		Wert für "Maßeinheit" ist ungültig	
	16#0011	Der Auswahlwert ist ungültig	Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben

Konfigurationsfehler der Auftragstabelle

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8700		Wert für "Auftragstyp" in der Auftragstabelle ist ungültig	
	16#0001	-	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
16#8701		Wert für "Position / Verfahrweg" in der Auftragstabelle ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als 1.0E+12)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als -1.0E+12	
16#8702		Wert für "Geschwindigkeit" in der Auftragstabelle ist ungültig	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0008	Wert ist größer als die konfigurierte maximale Geschwindigkeit	
	16#0009	Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit	
16#8703		Wert für "Dauer" in der Auftragstabelle ist ungültig	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0021	Wert ist größer als 64800 s	
	16#0022	Wert ist kleiner als 0.001 s	
16#8704		**Wert für "Nächster Schritt" in der Auftragstabelle ist ungültig**	
	16#0011	Der Auswahlwert ist ungültig	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten
	16#0023	Der Auftragsübergang ist für diesen Auftrag nicht erlaubt	

Interne Fehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8FFF		**Interner Fehler**	
	16#F0**	-	NETZ-AUS und NETZ-EIN der CPU Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit: • ErrorID • ErrorInfo • Diagnosepuffereinträge

Siehe auch

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V1)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

Betriebsfehler mit Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8000		**Antriebsfehler, "Antrieb bereit" ausgefallen**	
	16#0001	-	Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten
16#8001		**Unterer SW-Endschalter wurde ausgelöst**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#000E	Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren
	16#000F	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht	
	16#0010	Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten	
16#8002		Oberer SW-Endschalter wurde ausgelöst	
	16#000E	Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht	Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren
	16#000F	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht	
	16#0010	Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten	
16#8003		Unterer HW-Endschalter wurde angefahren	
	16#000E	Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.
16#8004		Oberer HW-Endschalter wurde angefahren	
	16#000E	Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten. (Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)	Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.
16#8005		PTO / HSC werden bereits von einer anderen Achse verwendet	
	16#0001	-	**Die Achse wurde fehlerhaft konfiguriert:** Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden
			Mehrere Achsen sollen mit einem PTO arbeiten: Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden. (siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504))

Betriebsfehler ohne Stopp der Achse

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8200		Achse ist nicht freigegeben	
	16#0001	-	Achse freigeben; Auftrag erneut starten
16#8201		Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0001	-	Achse nur über eine Anweisung "MC_Power" freigeben
16#8202		**Maximale Anzahl gleichzeitiger aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für alle Motion Control-Technologieobjekte)**	
	16#0001	-	Anzahl der gleichzeitig aktiven Aufträge verringern; Auftrag erneut starten Ein aktiver Auftrag ist am Parameter "Busy" = TRUE der Motion Control-Anweisung zu erkennen.
16#8203		**Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben**	
	16#0001	-	"Handsteuerung" beenden; Auftrag erneut starten
16#8204		**Achse ist nicht referenziert**	
	16#0001	-	Achse mit der Anweisung "MC_Home" referenzieren; Auftrag erneut starten
16#8205		**Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der Achssteuertafel angezeigt)**	
	16#0001	-	Achse mit der Anweisung "MC_Power" sperren und in der Achssteuertafel erneut "Handsteuerung" anwählen
16#8206		**Technologieobjekt Achse ist noch nicht aktiviert**	
	16#0001	-	Aktivieren Sie die Achse mit der Anweisung "MC_Power" Enable = TRUE, oder geben Sie die Achse in der Achssteuertafel frei.
16#8207		**Auftrag abgewiesen**	
	16#0016	Aktives Referenzieren läuft; eine andere Referenzierart kann nicht gestartet werden.	Warten Sie den Abschluss des Aktiven Referenzierens ab oder brechen Sie das Aktive Referenzieren durch einen Bewegungsauftrag z. B. "MC_Halt" ab. Anschließend kann die andere Referenzierart gestartet werden.

Bausteinparameterfehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8400		**Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert "Position" korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$)	
16#8401		**Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Wert "Distance" korrigieren; Auftrag erneut starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$)	
16#8402		**Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Wert "Velocity" korrigieren; Auftrag erneut starten
	16#0008	Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
16#8403		**Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig**	
	16#0011	Auswahlwert ungültig	Auswahlwert korrigieren; Auftrag erneut starten
16#8404		**Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Auswahlwert ungültig	Auswahlwert korrigieren; Auftrag erneut starten
	16#0015	Aktives / Passives Referenzieren ist nicht konfiguriert	Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
	16#0017	Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind	• Aktivieren der HW-Endschalter über die Variable <Achse>.Config.PositionLimits_HW.Active = TRUE, Auftrag erneut starten • Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten
16#8405		**Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig**	
	16#0011	Auswahlwert ungültig	Auswahlwert korrigieren; Achse erneut freigeben
16#8406		**Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt**	
	16#0001	-	Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.
16#8407		**Wechsel der Achse an der Motion Control-Anweisung "MC_Power" nur bei gesperrter Achse zulässig**	
	16#0001	-	Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.

Konfigurationsfehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8600		**Parametrierung des Impulsgenerators (PTO) ist ungültig**	
	16#000B	Adresse ist ungültig	Konfiguration des PTO (Pulse Train Output) korrigieren und in Steuerung laden
16#8601		**Parametrierung des schnellen Zählers (HSC) ist ungültig**	
	16#000B	Adresse ist ungültig	Konfiguration des HSC (High Speed Counter) korrigieren und in Steuerung laden
16#8602		**Parametrierung des "Freigabe-Ausgangs" ist ungültig**	
	16#000D	Adresse ist ungültig	Konfiguration korrigieren und in Steuerung laden
16#8603		**Parametrierung des "Bereit-Eingangs" ist ungültig**	
	16#000D	Adresse ist ungültig	Konfiguration korrigieren und in Steuerung laden
16#8604		**Wert für "Impulse pro Motorumdrehung" ist ungültig**	
	16#000A	Wert ist kleiner oder gleich Null	Konfiguration korrigieren und in Steuerung laden
16#8605		**Wert für "Weg pro Motorumdrehung" ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#000A	Wert ist kleiner oder gleich Null	
16#8606		**Wert für "Start/Stopp-Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#0003	Wert ist größer als die Hardwaregrenze	
	16#0004	Wert ist kleiner als die Hardwaregrenze	
	16#0007	Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit	
16#8607		**Wert für "Maximale Geschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#0003	Wert ist größer als die Hardwaregrenze	
	16#0004	Wert ist kleiner als die Hardwaregrenze	
16#8608		**Wert für "Beschleunigung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die Hardwaregrenze	
	16#0004	Wert ist kleiner als die Hardwaregrenze	
16#8609		**Wert für "Verzögerung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die Hardwaregrenze	
	16#0004	Wert ist kleiner als die Hardwaregrenze	
16#860A		**Wert für "Notstopp-Verzögerung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0003	Wert ist größer als die Hardwaregrenze	
	16#0004	Wert ist kleiner als die Hardwaregrenze	
16#860B		**Wert für die Position des unteren SW-Endschalters ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$)	
	16#0007	Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters	
16#860C		**Wert für die Position des oberen SW-Endschalters ist ungültig**	

ErrorID	ErrorInfo	Beschreibung	Abhilfe
	16#0002	Wert hat kein gültiges Zahlenformat	• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben • Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$)	
16#860D		**Adresse des unteren HW-Endschalters ist ungültig**	
	16#000C	Adresse der fallenden Flanke ist ungültig	Konfiguration korrigieren und in Steuerung laden
	16#000D	Adresse der steigenden Flanke ist ungültig	
16#860E		**Adresse des oberen HW-Endschalter ist ungültig**	
	16#000C	Adresse der fallenden Flanke ist ungültig	Konfiguration korrigieren und in Steuerung laden
	16#000D	Adresse der steigenden Flanke ist ungültig	
16#860F		**Wert für "Referenzpunktverschiebung" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#0005	Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$)	
	16#0006	Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$)	
16#8610		**Wert für "Anfahrgeschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#0008	Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
16#8611		**Wert für "Referenziergeschwindigkeit" ist ungültig**	
	16#0002	Wert hat kein gültiges Zahlenformat	Konfiguration korrigieren und in Steuerung laden
	16#0008	Geschwindigkeit ist größer als die maximale Geschwindigkeit	
	16#0009	Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit	
16#8612		**Adresse des Referenzpunktschalters ist ungültig**	
	16#000C	Adresse der fallenden Flanke ist ungültig	Konfiguration korrigieren und in Steuerung laden
	16#000D	Adresse der steigenden Flanke ist ungültig	
16#8613		**Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind**	
	16#0001	-	Konfiguration korrigieren und in Steuerung laden

Interne Fehler

ErrorID	ErrorInfo	Beschreibung	Abhilfe
16#8FFF		Interner Fehler	
	16#F0**	-	NETZ-AUS und NETZ-EIN der CPU Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit: • ErrorID • ErrorInfo • Diagnosepuffereinträge

Siehe auch

Liste der ErrorIDs und ErrorInfos (Technologieobjekte V4...5) (Seite 1604)

Mehrere Achsen mit gleichem PTO einsetzen (Seite 1504)

Legende V1...5

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Startwert (Wertebereich der Variable - Minimalwert bis Maximalwert) Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
Zugriff	Zugriff auf die Variable im Anwenderprogramm	
	R	Die Variable kann im Anwenderprogramm und im HMI gelesen werden.
	RP	Die Variable kann mit der Motion Control-Anweisung "MC_ReadParam" gelesen werden. Der aktuelle Wert der entsprechenden Variablen wird beim Start des Auftrags ermittelt.
	RW	Die Variable kann im Anwenderprogramm und im HMI gelesen und geschrieben werden. Die Variable kann mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	RCCP	Die Variable kann im Anwenderprogramm und im HMI gelesen werden und wird jeweils am Zykluskontrollpunkt aktualisiert.
	WP	Unabhängig von der Antriebsanbindung: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WP_PD	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WP_PTO	Bei Antriebsanbindung über PTO: Wenn die Achse gesperrt ist (MC_Power.Status = FALSE), kann die Variable mit der Motion Control-Anweisung "MC_WriteParam" geschrieben werden.
	WD BL	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Die Variable kann mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert im Ladespeicher geschrieben werden.
	-	Die Variable kann im Anwenderprogramm nicht verwendet werden.

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	1	Bei Antriebsanbindung über PTO: Mit dem Aktivieren, Sperren oder Freigeben der Achse
	2	Bei Antriebsanbindung über PTO: Mit dem Freigeben der Achse
	3	Nach dem Freigeben der Achse (die Achse muss zuvor im Stillstand gewesen sein). Der Stillstand der Achse kann mit der Variable <Achsname>.StatusBits.Standstill geprüft werden.
	4	Nach einem Stillstand der Achse mit dem nächsten Start eines Motion Control-Auftrags. Der Stillstand der Achse kann mit der Variable <Achsname>.StatusBits.Standstill geprüft werden.
	5	Bei Antriebsanbindung über PTO: Mit dem nächsten Start eines MC_MoveAbsolute-, MC_MoveRelative-, MC_MoveVelocity-, MC_MoveJog-, MC_Halt-, MC_CommandTable- oder aktiven MC_Home-Auftrags (Mode = 3)
	6	Bei Antriebsanbindung über PTO: Mit dem Stoppen eines MC_MoveJog-Auftrags
	7	Bei Antriebsanbindung über PTO: Mit dem Start eines passiven Referenzierauftrags
	8	Bei Antriebsanbindung über PTO: Mit dem Start eines aktiven Referenzierauftrags
	9	Mit dem Restart des Technologieobjekts
	10	Bei Antriebsanbindung über PROFIdrive/Analogausgang: Mit dem nächsten Aufruf des MC-Servo [OB91]
Beschreibung	Beschreibung der Variablen	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Variablen des Technologieobjekts Achse V1...3

Variablen Config V1...3

Die Variablenstruktur <Achsname>.Config.<Variablenname> beinhaltet die konfigurierten Parameter.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Config.	STRUCT				TO_Struct_Config

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
General.	STRUCT				TO_Struct_General	
PTO	DWORD	DW#16#00000000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
HSC	DWORD	DW#16#00000000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
LengthUnit[2]	INT	1013 (-32768 bis 32767)	R WP	- 2	Konfigurierte Maßeinheit der Parameter	
					-1	Impulse
					1005	°
					1010	m
					1013	mm
					1018	ft
					1019	in
DriveInterface.	STRUCT				TO_Struct_DriveInterface	
EnableOutput	-	-	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
ReadyInput	-	-	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
Mechanics.	STRUCT				TO_Struct_Mechanics	
PulsesPerDriveRevolution	DINT	L#1000	R	-	Impulse pro Motorumdrehung	
LeadScrew	REAL	10.0	R	-	Weg pro Motorumdrehung (Angabe in der konfigurierten Maßeinheit)	
InverseDirection	BOOL	FALSE	R	-	Richtungssinn invertieren	
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits	
MinVelocity	REAL	10.0	R	-	Start/Stopp-Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)	
MaxVelocity	REAL	250.0	R	-	Maximale Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)	
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults	
Acceleration	REAL	48.0	RW	1[2], 5, 6[2]	Beschleunigung der Achse (Angabe in der konfigurierten Maßeinheit)	
Deceleration	REAL	48.0	RW	1[2], 5, 6	Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit)	
EmergencyDeceleration	REAL	120.0	RW	1[2], 2[1], 5, 6	Notstopp-Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit)	
JerkActive[2]	BOOL	FALSE	RW	1, 5	FALSE	Die Ruckbegrenzung ist deaktiviert.
					TRUE	Die Ruckbegrenzung ist aktiviert.
Jerk[2]	REAL	192.0	RW	1, 5	Ruck während der Beschleunigungs- und Verzögerungsrampe der Achse (Angabe in der konfigurierten Maßeinheit)	
PositionLimits_SW.	STRUCT				TO_Struct_PositionLimits_SW	
Active	BOOL	FALSE	RW	1[2], 4[1],	FALSE	Die Software-Endschalter sind deaktiviert.

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
				5[2], 6[2]	TRUE	Die Software-Endschalter sind aktiviert.
MinPosition	REAL	-1.0E4	RW	1[2], 4[1], 5[2], 6[2]	Position des unteren Software-Endschalters (Angabe in der konfigurierten Maßeinheit)	
MaxPosition	REAL	1.0E4	RW	1[2], 4[1], 5[2], 6[2]	Position des oberen Software-Endschalters (Angabe in der konfigurierten Maßeinheit)	
PositionLimits_HW.					TO_Struct_PositionLimits_HW	
Active	BOOL	FALSE	RW	1[2], 3[1], 4[1], 5[2], 6[2]	FALSE	Die Hardware-Endschalter sind deaktiviert.
					TRUE	Die Hardware-Endschalter sind aktiviert.
MinSwitchedLevel	BOOL	FALSE	R	-	FALSE	0 V am CPU-Eingang entspricht unterer Hardware-Endschalter angefahren.
					TRUE	24 V am CPU-Eingang entspricht unterer Hardware-Endschalter angefahren.
MinFallingEvent	DWORD	DW#16#000 00000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
MinRisingEvent	DWORD	DW#16#000 00000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
MaxSwitchedLevel	BOOL	FALSE	R	-	FALSE	0 V am CPU-Eingang entspricht oberer Hardware-Endschalter angefahren.
					TRUE	24 V am CPU-Eingang entspricht oberer Hardware-Endschalter angefahren.
MaxFallingEvent	DWORD	DW#16#000 00000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
MaxRisingEvent	DWORD	DW#16#000 00000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	TRUE	R[1] RW[2]	-[1] 1[2], 8[2]	FALSE	Richtungsumkehr am Hardware-Endschalter deaktiviert (aktives Referenzieren)
					TRUE	Richtungsumkehr am Hardware-Endschalter aktiviert (aktives Referenzieren)
Direction	BOOL	TRUE	R[1] RW[2]	-[1] 1[2], 8[2]	FALSE	Negative Anfahrrichtung zur Suche des Referenzpunktschalters und positive Referenzierrichtung (aktives Referenzieren)
					TRUE	Positive Anfahrrichtung zur Suche des Referenzpunktschalters und

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
						positive Referenzierrichtung (aktives Referenzieren)
SideActiveHoming[2]	BOOL	TRUE	RW	1, 8	FALSE	Referenzieren an der unteren Seite des Referenzpunktschalters (aktives Referenzieren)
					TRUE	Referenzieren an der oberen Seite des Referenzpunktschalters (aktives Referenzieren)
SidePassiveHoming[2]	BOOL	TRUE	RW	1, 7	FALSE	Referenzieren an der unteren Seite des Referenzpunktschalters (passives Referenzieren)
					TRUE	Referenzieren an der oberen Seite des Referenzpunktschalters (passives Referenzieren)
RisingEdge[1]	BOOL	FALSE	R	-	FALSE	Referenzieren mit der positiven Signalflanke des Referenzpunktschalters (aktives Referenzieren)
					TRUE	Referenzieren mit der negativen Signalflanke des Referenzpunktschalters (aktives Referenzieren)Den Einfluss der Variable auf das passive Referenzieren entnehmen Sie der Beschreibung unter "Konfiguration - Referenzieren".
Offset	REAL	0.0	R RW	-[1] 1[2], 8[2]	Referenzpunktverschiebung/Angabe in der konfigurierten Maßeinheit (aktives Referenzieren)	
FastVelocity	REAL	200.0	R RW	-[1] 1[2], 8[2]	Anfahrgeschwindigkeit/Angabe in der konfigurierten Maßeinheit (aktives Referenzieren)	
SlowVelocity	REAL	40.0	R RW	-[1], 1[2], 8[2]	Referenziergeschwindigkeit/Angabe in der konfigurierten Maßeinheit (aktives Referenzieren)	
FallingEvent	DWORD	DW#16#00000000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	
RisingEvent	DWORD	DW#16#00000000	-	-	Die Variable ist im Anwenderprogramm nicht auswertbar.	

1) Gilt nur für Technologieversion V1.0

2) Gilt ab Technologieversion V2.0

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen MotionStatus V1...3

Die Variablenstruktur <Achsname>.MotionStatus.<Variablenname> beinhaltet den Status der Bewegung.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
MotionStatus.	STRUCT				TO_Struct_MotionStatus
Position	REAL	0.0 (-9.0E15 bis 9.0E15)	R	-	Istposition der Achse (Angabe in der konfigurierten Maßeinheit) Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabeposition der Achse.
Velocity	REAL	0.0 (-9.0E15 bis 9.0E15)	R	-	Istgeschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)
Distance	REAL	0.0 (-9.0E15 bis 9.0E15)	R	-	Aktueller Abstand zur Zielposition der Achse (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel gültig.
TargetPosition	REAL	0.0 (-9.0E15 bis 9.0E15)	R	-	Zielposition der Achse (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel gültig.

Siehe auch

Bewegungsstatus (Seite 1502)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen StatusBits V1...3

Die Variablenstruktur <Achsname>.StatusBits.<Variablenname> beinhaltet die Statusinformationen des Technologieobjekts.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StatusBits.	STRUCT				TO_Struct_StatusBits

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Activated	BOOL	FALSE	R	-	Aktivierung der Achse	
					FALSE	Die Achse ist nicht aktiviert.
					TRUE	Die Achse ist aktiviert. Die Achse ist mit dem zugeordneten PTO (Pulse Train Output) verbunden. Die Daten des Technologie-Datenbausteins werden zyklisch aktualisiert.
Enable	BOOL	FALSE	R	-	Freigabestatus der Achse	
					FALSE	Die Achse ist nicht freigegeben.
					TRUE	Die Achse ist freigegeben und bereit, Motion Control-Aufträge anzunehmen.
HomingDone	BOOL	FALSE	R	-	Referenzierungsstatus der Achse	
					FALSE	Die Achse ist nicht referenziert.
					TRUE	Die Achse ist referenziert und kann absolute Positionieraufträge ausführen.
					Zum relativen Positionieren muss die Achse nicht referenziert sein. Während des aktiven Referenzierens ist der Status FALSE. Während des passiven Referenzierens bleibt der Status TRUE, wenn die Achse zuvor bereits referenziert war.	
Done	BOOL	FALSE	R	-	Auftragsbearbeitung der Achse	
					FALSE	An der Achse ist ein Motion Control-Auftrag aktiv.
					TRUE	An der Achse ist kein Motion Control-Auftrag aktiv.
Error	BOOL	FALSE	R	-	Fehlerstatus der Achse	
					FALSE	An der Achse steht kein Fehler an.
					TRUE	An der Achse ist ein Fehler aufgetreten.
					Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern "ErrorID" und "ErrorInfo" der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt.	
Standstill	BOOL	FALSE	R	-	Stillstand der Achse	
					FALSE	Die Achse ist in Bewegung.
					TRUE	Die Achse befindet sich im Stillstand.
PositioningCommand	BOOL	FALSE	R	-	Bearbeitung eines Positionierauftrags	
					FALSE	An der Achse ist kein Positionierauftrag aktiv.

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
					TRUE	Die Achse führt einen Positionierauftrag aus.
SpeedCommand	BOOL	FALSE	R	-	Bearbeitung eines Auftrags mit Geschwindigkeitsvorgabe	
					FALSE	An der Achse ist kein Auftrag mit Geschwindigkeitsvorgabe aktiv.
					TRUE	Die Achse führt einen Verfahrauftrag mit Geschwindigkeitsvorgabe aus.
Homing	BOOL	FALSE	R	-	Bearbeitung eines Referenzierauftrags	
					FALSE	An der Achse ist kein Referenzierauftrag aktiv.
					TRUE	Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus.
CommandTableActive	BOOL	FALSE	R	-	Bearbeitung einer Auftragstabelle	
					FALSE	An der Achse ist keine Auftragstabelle aktiv.
					TRUE	Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert.
ConstantVelocity	BOOL	FALSE	R	-	Konstante Geschwindigkeit	
					FALSE	Die Achse wird beschleunigt, abgebremst oder befindet sich im Stillstand.
					TRUE	Die Sollgeschwindigkeit wurde erreicht. Die Achse wird mit konstanter Geschwindigkeit verfahren.
Acceleration	BOOL	FALSE	R	-	Beschleunigungsvorgang	
					FALSE	Die Achse wird abgebremst, mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird beschleunigt.
Deceleration	BOOL	FALSE	R	-	Verzögerungsvorgang	
					FALSE	Die Achse wird beschleunigt, mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird abgebremst.
ControlPanelActive	BOOL	FALSE	R	-	Aktivierungsstatus der Achssteuertafel	
					FALSE	Die Betriebsart "Automatikbetrieb" ist aktiviert. Das Anwenderprogramm hat die Steuerungshoheit über die Achse.
					TRUE	Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
						Steuerungshoheit über die Achse. Die Achse kann nicht vom Anwenderprogramm gesteuert werden.
DriveReady	BOOL	FALSE	R	-	Operationsstatus des Antriebs	
					FALSE	Der Antrieb ist nicht bereit. Sollwerte werden nicht ausgeführt.
					TRUE	Der Antrieb ist bereit. Sollwerte können ausgeführt werden.
RestartRequired	BOOL	FALSE	R	-	Restart der Achse erforderlich	
					FALSE	Ein Restart der Achse ist nicht erforderlich.
					TRUE	Werte wurden im Ladespeicher geändert.
					Um die Werte im Betriebszustand RUN der CPU in den Arbeitsspeicher zu laden, ist ein Restart der Achse erforderlich. Verwenden Sie hierzu die Motion Control-Anweisung "MC_Reset".	

Siehe auch

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen ErrorBits V1...3

Die Variablenstruktur <Achsname>.ErrorBits.<Variablenname> zeigt Fehler am Technologieobjekt an.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	R	-	Interner Systemfehler
ConfigFault	BOOL	FALSE	R	-	Fehlerhafte Konfiguration der Achse
DriveFault	BOOL	FALSE	R	-	Fehler im Antrieb. Ausfall des "Antrieb bereit"-Signals.
SWLimitMinReached	BOOL	FALSE	R	-	Der untere Software-Endschalters wurde erreicht.
SWLimitMinExceeded	BOOL	FALSE	R	-	Der untere Software-Endschalters wurde überschritten.
SWLimitMaxReached	BOOL	FALSE	R	-	Der obere Software-Endschalters wurde erreicht.
SWLimitMaxExceeded	BOOL	FALSE	R	-	Der obere Software-Endschalters wurde überschritten.
HWLimitMin	BOOL	FALSE	R	-	Der untere Hardware -Endschalters wurde angefahren.
HWLimitMax	BOOL	FALSE	R	-	Der obere Hardware -Endschalters wurde angefahren.
HWUsed	BOOL	FALSE	R	-	Eine weitere Achse verwendet denselben PTO (Pulse Train Output) und ist mit "MC_Power" freigegeben.

Siehe auch

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen Internal V1...3

Die Variablen "Internal" beinhalten keine anwenderrelevanten Daten; im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Variablen ControlPanel V1...3

Die Variablen "ControlPanel" beinhalten keine anwenderrelevanten Daten; im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Positionierachse V4...5 (Seite 1652)

Aktualisierung der Variablen des Technologieobjekts

Die in den Variablen des Technologieobjekts angezeigten Status- und Fehlerinformationen der Achse werden jeweils am Zykluskontrollpunkt aktualisiert.

Die Änderung von Werten änderbarer Konfigurations-Variablen wird nicht sofort wirksam. Unter welchem Bedingungen eine Änderung wirksam wird, entnehmen Sie der Detailbeschreibung der entsprechenden Variable.

Variablen des Technologieobjekts Positionierachse V4...5

Variablen zu Positions- und Geschwindigkeitswerten V4...5

Die Variablenstruktur beinhaltet die Soll- und Istwerte der Position und der Geschwindigkeit der Achse.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Position	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Sollposition der Achse (Angabe in der konfigurierten Maßeinheit) Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabeposition der Achse.
Velocity	REAL	0.0	RCCP, RP	-	Sollgeschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)
ActualPosition	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Istposition der Achse (Angabe in der konfigurierten Maßeinheit) Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabeposition der Achse.
ActualVelocity	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Istgeschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)

Siehe auch

Bewegungsstatus (Seite 1502)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Actor V4...5

Die Variablenstruktur <Achsname>.Actor.<Variablenname> beinhaltet die Parameter des Antriebs.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Actor.	STRUCT				TO_Struct_Actor

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Type	DINT	1 (0 bis 2)	R	-	Technologieobjekt Positionierachse ab V5:	
					0	Der Antrieb wird über einen Analogausgang angebunden. Alle Bewegungen der Achse werden lagegeregelt.
					1	Der Antrieb wird über PROFIdrive-Telegramme angebunden. Alle Bewegungen der Achse werden lagegeregelt.
					2	Der Antrieb wird über eine Impulsschnittstelle angebunden.
					Technologieobjekt Positionierachse V4:	
					2	Der Antrieb wird über eine Impulsschnittstelle angebunden.
InverseDirection	BOOL	FALSE	R WP_PTO	- 2	FALSE	Der Richtungssinn wird nicht invertiert.
					TRUE	Der Richtungssinn wird invertiert.
DirectionMode	INT	0 (0 bis 2)	R WP_PTO	- 2	Zugelassene Drehrichtung	
					0	Beide Richtungen
					1	Positive Richtung
					2	Negative Richtung
Interface.	STRUCT				TO_Struct_ActorInterface	
AddressIn[3)]	VREF	-	-	-	Eingangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
AddressOut[3)]	VREF	-	-	-	Ausgangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
EnableDriveOutput	VREF	-	-	-	Freigabeausgang (interner Parameter)	
DriveReadyInput	VREF	-	-	-	Bereiteingang (interner Parameter)	
PTO	DWORD	0	-	-	Impulsausgang (interner Parameter)	
DriveParameter.	STRUCT				TO_Struct_ActorDriveParameter	
ReferenceSpeed[3)]	REAL	3000.0	R	-	Bezugswert (100%) für die Solldrehzahl des Antriebs (n-Soll) Der Drehzahlsollwert wird im PROFIdrive-Telegramm als normierter Wert von -200% bis 200% von "ReferenceSpeed" übertragen. Bei Sollwertvorgabe über einen Analogwert kann der Analogausgang im Bereich -117% bis 117% betrieben werden, wenn der Antrieb das zulässt.	
MaxSpeed[3)]	REAL	3000.0	R	-	Maximalwert für die Solldrehzahl des Antriebs (n-Soll) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Analogsollwert: MaxSpeed ≤ 1.17 × ReferenceSpeed)	
PulsesPerDriveRevolution	DINT	1000	R WP_PTO	- 2	Pulse pro Motorumdrehung	

Variable			Datentyp	Werte	Zugriff	W	Beschreibung
				(1 bis 2147483648)			

3) Gilt ab Technologieversion V5.0

Siehe auch

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen Sensor[1] V4...5

Die Variablenstruktur <Achsname>.Sensor.<Variablenname> beinhaltet die Parameter des Gebers.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Sensor.	STRUCT				TO_Struct_Sensor

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Sensor[1].	STRUCT				ARRAY[1..1] TO_Struct_Sensor	
Type	DINT	0 (0 bis 1)	R	-	Gebertyp (interner Parameter)	
					0	Inkrementell
					1	Absolut
InverseDirection	BOOL	FALSE	R	-	Invertierung des Istwerts	
					FALSE	Istwert wird nicht invertiert
					TRUE	Istwert wird invertiert
System	DINT	1 (0 bis 1)	R	-	Gebersystem	
					0	Linearer Geber
					1	Rotatorischer Geber
MountingMode	DINT	0 (0 bis 2)	R	-	Anbauart des Gebers	
					0	An der Motorwelle
					2	Externes Messsystem
Interface.	STRUCT				TO_Struct_SensorInterface	
AddressIn.	VREF	-	-	-	Eingangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
AREA	BYTE	-	-	-	Interner Parameter	
DB_NUMBER	UINT	-	-	-	Interner Parameter	
OFFSET	UDINT	-	-	-	Interner Parameter	
RID	DWORD	-	-	-	Interner Parameter	
AddressOut.	VREF	-	-	-	Ausgangsadresse für das PROFIdrive-Telegramm (interner Parameter)	
AREA	BYTE	-	-	-	Interner Parameter	
DB_NUMBER	UINT	-	-	-	Interner Parameter	
OFFSET	UDINT	-	-	-	Interner Parameter	
RID	DWORD	-	-	-	Interner Parameter	
Parameter.	STRUCT				TO_Struct_SensorParameter	
Resolution	REAL	0.001 (-1.0E12 bis 1.0E12)	R	-	Auflösung eines linearen Gebers (Abstand zwischen zwei Geberstrichen)	
StepsPerRevolution	UDINT	2048 (1 bis 8388608)	R	-	Inkremente pro Geberumdrehung bei einem rotatorischen Geber	
FineResolutionXist1	UDINT	11 (0 bis 31)	R	-	Anzahl Bits für die Feinauflösung Gn_XIST1 (zyklischer Geberistwert)	
FineResolutionXist2	UDINT	9 (0 bis 31)	R	-	Anzahl Bits für die Feinauflösung Gn_XIST2 (Absolutwert des Gebers)	
DeterminableRevolutions	UDINT	1 (0 bis 8388608)	R	-	Anzahl unterscheidbarer Geberumdrehungen bei einem Multiturn-Absolutwertgeber	
					0	Inkrementtalgeber
					1	Singlereturn-Absolutwertgeber
DistancePerRevolution	REAL	100.0 (0.0 bis 1.0E12)	R	-	Weg der Last pro Geberumdrehung bei einem extern montierten Geber	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
ActiveHoming.	STRUCT				TO_Struct_SensorActiveHoming	
Mode	DINT	2 (0 bis 2)	R WP_PTO	- 2	Modus aktives Referenzieren	
					Technologieobjekt Positionierachse ab V5:	
					0	Nullmarke über PROFIdrive-Telegramm (nicht PTO)
					1	Nullmarke über PROFIdrive-Telegramm und Näherungsschalter (nicht PTO)
					2	Referenzieren über Digitaleingang
					Technologieobjekt Positionierachse V4:	
					2	Referenzieren über Digitaleingang
SideInput	BOOL	FALSE	RW	1, 8, 10	Seite des Referenzpunktschalters, auf die beim aktiven Referenzieren referenziert wird	
					FALSE	Untere Seite
					TRUE	Obere Seite
DigitalInputAddress.	VREF	-	-	-	Symbolische Eingangsadresse des Referenzpunktschalters (interner Parameter)	
AREA	BYTE	-	-	-	Interner Parameter	
DB_NUMBER	UINT	-	-	-	Interner Parameter	
OFFSET	UDINT	-	-	-	Interner Parameter	
RID	DWORD	-	-	-	Interner Parameter	
HomePositionOffset	REAL	0.0 (-1.0E12 bis 1.0E12)	RW	1, 8, 10	Referenzpunktverschiebung (aktives Referenzieren) (Angabe in der konfigurierten Maßeinheit)	
SwitchLevel	BOOL	TRUE	RW	1, 8, 10	Auswahl des Signalpegels, der bei angefahrenem Referenzpunktschalter am CPU-Eingang ansteht	
					FALSE	Unterer Pegel (Low-aktiv)
					TRUE	Oberer Pegel (High-aktiv)
PassiveHoming.	STRUCT				TO_Struct_SensorPassiveHoming	
Mode	DINT	2 (0 bis 2)	R WP_PTO	- 2	Modus passives Referenzieren	
					Technologieobjekt Positionierachse ab V5:	
					0	Nullmarke über PROFIdrive-Telegramm (nicht PTO)
					1	Nullmarke über PROFIdrive-Telegramm und Näherungsschalter (nicht PTO)
					2	Referenzieren über Digitaleingang
					Technologieobjekt Positionierachse V4:	
					2	Referenzieren über Digitaleingang
SideInput	BOOL	FALSE	RW	1, 7, 10	Seite des Referenzpunktschalters, auf die beim passiven Referenzieren referenziert wird	
					FALSE	Untere Seite
					TRUE	Obere Seite
DigitalInputAddress.	VREF	-	-	-	Symbolische Eingangsadresse des Referenzpunktschalters (interner Parameter)	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
AREA	BYTE	-	-	-	Interner Parameter	
DB_NUMBER	UINT	-	-	-	Interner Parameter	
OFFSET	UDINT	-	-	-	Interner Parameter	
RID	DWORD	-	-	-	Interner Parameter	
SwitchLevel	BOOL	TRUE	RW	1, 7, 10	Auswahl des Pegels, der bei angefahrenem Referenzpunktschalter am CPU-Eingang ansteht	
					FALSE	Unterer Pegel (Low-aktiv)
					TRUE	Oberer Pegel (High-aktiv)

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variable Units V4...5

Die Variablenstruktur <Achsname>.Units.LengthUnit beinhaltet die konfigurierten Maßeinheiten der Parameter.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Units.	STRUCT				TO_Struct_Units	
LengthUnit	INT	1013 (-32768 bis 32767)	R WP_PTO	- 2	Konfigurierte Maßeinheit der Parameter	
					-1	Impulse
					1005	°
					1010	m
					1013	mm
					1018	ft
					1019	in

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variable Mechanics V4...5

Die Variablenstruktur <Achsname>.Mechanics.LeadScrew beinhaltet den zurückgelegten Weg pro Motorumdrehung.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
Mechanics.	STRUCT				TO_Struct_Mechanics
LeadScrew	REAL	10.0 (-1.0E12 bis 1.0E12)	R, WP_PTO	-	Weg pro Motorumdrehung (Angabe in der konfigurierten Maßeinheit)

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Modulo V4...5

Die Variablenstruktur <Achsname>.Modulo.<Variablenname> beinhaltet die Moduloeinstellungen.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Modulo.	STRUCT				TO_Struct_Modulo	
Enable	BOOL	FALSE	R	-	FALSE	Moduloumrechnung deaktiviert
					TRUE	Moduloumrechnung aktiviert Bei aktivierter Moduloumrechnung wird auf Modulolänge > 0.0 geprüft
Length	REAL	360.0 (0.001 bis 1.0E12)	R	-	Modulolänge	
StartValue	REAL	0.0 (-1.0E12 bis 1.0E12)	R	-	Modulostartwert	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen DynamicLimits V4...5

Die Variablenstruktur <Achsname>.DynamicLimits.<Variablenname> beinhaltet die Konfiguration der Dynamikgrenzen.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
DynamicLimits.	STRUCT				TO_Struct_DynamicLimits
MaxVelocity	REAL	250.0	R WP_PTO	- 2	Maximale Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)
MinVelocity	REAL	10.0	R WP_PTO	- 2	Start/Stopp-Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit)

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen DynamicDefaults V4...5

Die Variablenstruktur <Achsname>.DynamicDefaults.<Variablenname> beinhaltet die Konfiguration der Dynamikvoreinstellungen.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
DynamicDefaults.	STRUCT				TO_Struct_DynamicDefaults
Acceleration	REAL	48.0 (0.0 bis 1.0E12)	RW	1, 5, 6, 10	Voreinstellung der Beschleunigung der Achse (Angabe in der konfigurierten Maßeinheit)
Deceleration	REAL	48.0 (0.0 bis 1.0E12)	RW	1, 5, 6, 10	Voreinstellung der Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit)
Jerk	REAL	192.0 (0.0 bis 1.0E12)	RW	1, 5, 10	Voreinstellung des Ruck während der Beschleunigungs- und Verzögerungsrampe der Achse (Angabe in der konfigurierten Maßeinheit) Der Ruck ist aktiviert, wenn der konfigurierte Ruck größer als 0.00004 mm/s² ist.
EmergencyDeceleration	REAL	120.0 (0.0 bis 1.0E12)	RW	1, 5, 6, 10	Notstopp-Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit)

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionLimitsSW V4...5

Die Variablenstruktur <Achsname>.PositionLimits_SW.<Variablenname> beinhaltet die die Konfiguration der Positionsüberwachung mit Software-Endschaltern. Mit Software-Endschaltern begrenzen Sie den Arbeitsbereich einer Positionierachse.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
PositionLimits_SW.	STRUCT				TO_Struct_PositionLimitsSW	
Active	BOOL	FALSE	RW	1, 5, 6, 10	FALSE	Die Software-Endschalter sind deaktiviert.
					TRUE	Die Software-Endschalter sind aktiviert.
MinPosition	REAL	-10000.0 (-1.0E12 bis 1.0E12)	RW	1, 5, 6, 10	Position des unteren Software-Endschalters (Angabe in der konfigurierten Maßeinheit)	
MaxPosition	REAL	10000.0 (-1.0E12 bis 1.0E12)	RW	1, 5, 6, 10	Position des oberen Software-Endschalters (Angabe in der konfigurierten Maßeinheit)	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionLimitsHW V4...5

Die Variablenstruktur <Achsname>.PositionLimits_HW.<Variablenname> beinhaltet die die Konfiguration der Positionsüberwachung mit Hardware-Endschaltern. Mit Hardware-Endschaltern begrenzen Sie den Verfahrbereich einer Positionierachse.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
PositionLimits_HW.	STRUCT				TO_Struct_PositionLimitsHW	
Active	BOOL	FALSE	RW	1, 5, 6, 10	FALSE	Die Hardware-Endschalter sind deaktiviert.
					TRUE	Die Hardware-Endschalter sind aktiviert.
MinSwitchLevel	BOOL	FALSE	R WP_PTO	- 2	Auswahl des Signalpegels, der bei angefahrenem unteren Hardware-Endschalter am CPU-Eingang ansteht	
					FALSE	Unterer Pegel (Low-aktiv)
					TRUE	Oberer Pegel (High-aktiv)
MinSwitchAddress	VREF	-	-	-	Symbolische Eingangsadresse des unteren Hardware-Endschalters (interner Parameter)	
MaxSwitchLevel	BOOL	FALSE	R WP_PTO	- 2	Auswahl des Signalpegels, der bei angefahrenem oberen Hardware-Endschalter am CPU-Eingang ansteht	
					FALSE	Unterer Pegel (Low-aktiv)
					TRUE	Oberer Pegel (High-aktiv)
MaxSwitchAddress	VREF	-	-	-	Eingangsadresse des oberen Hardware-Endschalters (interner Parameter)	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Homing V4...5

Die Variablenstruktur <Achsname>.Homing.<Variablenname> beinhaltet die die Konfiguration für das Referenzieren der Achse.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Homing.	STRUCT				TO_Struct_Homing	
AutoReversal	BOOL	FALSE	RW	1, 8, 10	FALSE	Die Richtungsumkehr am Hardware-Endschalter ist deaktiviert.
					TRUE	Die Richtungsumkehr am Hardware-Endschalter ist aktiviert.
ApproachDirection	BOOL	TRUE	RW	1, 8, 10	FALSE	Negative Anfahrrichtung zur Suche des Referenzpunktschalters und negative Referenzierrichtung
					TRUE	Positive Anfahrrichtung zur Suche des Referenzpunktschalters und positive Referenzierrichtung
ApproachVelocity	REAL	200.0 (0.0 bis 1.0E12)	RW	1, 8, 10	Anfahrgeschwindigkeit der Achse beim aktiven Referenzieren (Angabe in der konfigurierten Maßeinheit)	
ReferencingVelocity	REAL	40.0 (0.0 bis 1.0E12)	RW	1, 8, 10	Referenziergeschwindigkeit der Achse beim aktiven Referenzieren (Angabe in der konfigurierten Maßeinheit)	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositionControl V5

Die Variablenstruktur <Achsname>.PositionControl.<Variablenname> beinhaltet Einstellungen der Lageregelung.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionControl.	STRUCT				TO_Struct_PositionControl
Kv	REAL	10.0 (0.0 bis 2147480.0)	R WP	- 10	P-Verstärkung der Lageregelung ("Kv" > 0.0)
Kpc	REAL	100.0 (0.0 bis 150.0)	R WP	- 10	Prozentuale Geschwindigkeitsvorsteuerung der Lageregelung

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen FollowingError V5

Die Variablenstruktur <Achsname>.FollowingError.<Variablenname> beinhaltet die Konfiguration der dynamischen Schleppfehlerüberwachung.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
FollowingError.	STRUCT				TO_Struct_FollowingError	
EnableMonitoring	BOOL	TRUE	R	-	FALSE	Schleppfehlerüberwachung deaktiviert
					TRUE	Schleppfehlerüberwachung aktiviert
MinValue	REAL	10.0 (0.0 bis 1.0E12)	R WP_PD	- 10	Zulässiger Schleppfehler bei Geschwindigkeiten unterhalb des Wertes von "MinVelocity".	
MaxValue	REAL	100.0 (0.0 bis 1.0E12)	R WP_PD	- 10	Maximal zulässiger Schleppfehler, der beim Maximum der Geschwindigkeit erreicht werden darf.	
MinVelocity	REAL	10.0 (0.0 bis 1.0E12)	R WP_PD	- 10	"MinValue" ist unterhalb dieser Geschwindigkeit zulässig und wird konstant gehalten.	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen PositioningMonitoring V5

Die Variablenstruktur <Achsname>.PositionMonitoring.<Variablenname> beinhaltet die Konfiguration der Positionsüberwachung am Ende einer Positionierbewegung.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
PositionMonitoring.	STRUCT				TO_Struct_PositionMonitoring
ToleranceTime	REAL	1.0 (0.0 bis 1.0E12)	R WP_PD	- 10	Toleranzzeit Maximal erlaubte Zeitdauer vom Erreichen der Sollgeschwindigkeit null bis zum Eintritt in das Positionierfenster.
MinDwellTime	REAL	0.1 (0.0 bis 1.0E12)	R WP_PD	- 10	Minimale Verweildauer im Positionierfenster
Window	REAL	1.0 (0.001 bis 1.0E12)	R WP_PD	- 10	Positionierfenster

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StandstillSignal V5

Die Variablenstruktur <Achsname>.StandstillSignal.<Variablenname> beinhaltet die Konfiguration des Stillstandsignals.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StandstillSignal.	STRUCT				TO_Struct_StandstillSignal
VelocityThreshold	REAL	5.0 (0.0 bis 1.0E12)	R WP_PD	- 10	Geschwindigkeitsschwelle Wird diese unterschritten, beginnt die Mindestverweildauer.
MinDwellTime	REAL	0.01 (0.0 bis 1.0E12)	R WP_PD	- 10	Mindestverweildauer

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusPositioning V4...5

Die Variablenstruktur <Achsname>.StatusPositioning.<Variablenname> zeigt den Status einer Positionierbewegung an.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StatusPositioning.	STRUCT				TO_Struct_StatusPositioning
Distance	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Aktueller Abstand der Achse zur Zielposition (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur gültig während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel.
TargetPosition	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Zielposition der Achse (Angabe in der konfigurierten Maßeinheit) Der Wert der Variable ist nur gültig während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel.
FollowingError	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Aktueller Schleppfehler der Achse (Angabe in der konfigurierten Maßeinheit FollowingError = 0.0 bei Antriebsanbindung über PTO (Pulse Train Output).

Siehe auch

Bewegungsstatus (Seite 1502)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusDrive V5

Die Variablenstruktur <Achsname>.StatusDrive.<Variablenname> zeigt den Status des Antriebs an.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
StatusDrive.	STRUCT				TO_Struct_StatusDrive	
InOperation	BOOL	FALSE	RCCP, RP	-	Operationsstatus des Antriebs	
					FALSE	Antrieb nicht bereit. Sollwerte werden nicht ausgeführt.
					TRUE	Antrieb bereit. Sollwerte können ausgeführt werden.
CommunicationOK	BOOL	FALSE	RCCP, RP	-	Zyklische BUS-Kommunikation zwischen Steuerung und Antrieb	
					FALSE	Kommunikation nicht aufgebaut
					TRUE	Kommunikation aufgebaut

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusSensor V5

Die Variablenstruktur <Achsname>.StatusSensor.<Variablenname> zeigt den Status des Messsystems an.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
StatusSensor.	STRUCT				TO_Struct_StatusSensor	
State	DINT	0 (0 bis 2)	RCCP, RP	-	Status des Geberwerts	
					0	Nicht gültig
					1	Warte auf Status gültig
					2	Gültig
CommunicationOK	BOOL	FALSE	RCCP, RP	-	Zyklische BUS-Kommunikation zwischen Steuerung und Geber	
					FALSE	Kommunikation nicht aufgebaut
					TRUE	Kommunikation aufgebaut
AbsEncoderOffset	REAL	0.0 (-9.0E15 bis 9.0E15)	RCCP, RP	-	Referenzpunktverschiebung zum Wert eines Absolutwertgebers. Der Wert wird remanent in der CPU gespeichert.	

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen StatusBits V4...5

Die Variablenstruktur <Achsname>.StatusBits.<Variablenname> beinhaltet die Statusinformationen des Technologieobjekts.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
StatusBits.	STRUCT				TO_Struct_StatusBits

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Activated	BOOL	FALSE	RCCP, RP	-	Aktivierung der Achse	
					FALSE	Die Achse ist nicht aktiviert.
					TRUE	Die Achse ist aktiviert. Die Achse ist mit dem zugeordneten PTO (Pulse Train Output) verbunden. Die Daten des Technologie-Datenbausteins werden zyklisch aktualisiert.
Enable	BOOL	FALSE	RCCP, RP	-	Freigabestatus der Achse	
					FALSE	Die Achse ist nicht freigegeben.
					TRUE	Die Achse ist freigegeben und bereit, Motion Control-Aufträge anzunehmen.
AxisSimulation	BOOL	FALSE	RCCP, RP	-	FALSE	Die Simulation ist deaktiviert.
					TRUE	Die Simulation ist aktiviert.
NonPositionControlled	BOOL	FALSE	RCCP, RP	-	FALSE	Die Achse befindet sich im lagegeregelten Betrieb.
					TRUE	Die Achse befindet sich im nichtlagegeregelten Betrieb.
HomingDone	BOOL	FALSE	RCCP, RP	-	Referenzierungsstatus der Achse	
					FALSE	Die Achse ist nicht referenziert.
					TRUE	Die Achse ist referenziert und kann absolute Positionieraufträge ausführen.
					Zum relativen Positionieren muss die Achse nicht referenziert sein. Während des aktiven Referenzierens ist der Status FALSE. Während des passiven Referenzierens bleibt der Status TRUE, wenn die Achse zuvor bereits referenziert war.	
Done	BOOL	FALSE	RCCP, RP	-	Auftragsbearbeitung der Achse	
					FALSE	An der Achse ist ein Motion Control-Auftrag aktiv.
					TRUE	An der Achse ist kein Motion Control-Auftrag aktiv.
Error	BOOL	FALSE	RCCP, RP	-	Fehlerstatus der Achse	
					FALSE	An der Achse steht kein Fehler an.
					TRUE	An der Achse ist ein Fehler aufgetreten.
					Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern "ErrorID" und "ErrorInfo" der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt.	

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Standstill	BOOL	FALSE	RCCP, RP	-	Stillstand der Achse	
					FALSE	Die Achse ist in Bewegung.
					TRUE	Die Achse befindet sich im Stillstand.
PositioningCommand	BOOL	FALSE	RCCP, RP	-	Bearbeitung eines Positionierauftrags	
					FALSE	An der Achse ist kein Positionierauftrag aktiv.
					TRUE	Die Achse führt einen Positionierauftrag der Motion Control-Anweisungen "MC_MoveRelative" oder "MC_MoveAbsolute" aus.
VelocityCommand	BOOL	FALSE	RCCP, RP	-	Bearbeitung eines Auftrags mit Geschwindigkeitsvorgabe	
					FALSE	An der Achse ist kein Auftrag mit Geschwindigkeitsvorgabe aktiv.
					TRUE	Die Achse führt einen Verfahrauftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisungen "MC_MoveVelocity" oder "MC_MoveJog" aus.
HomingCommand	BOOL	FALSE	RCCP, RP	-	Bearbeitung eines Referenzierauftrags	
					FALSE	An der Achse ist kein Referenzierauftrag aktiv.
					TRUE	Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home".
CommandTableActive	BOOL	FALSE	RCCP, RP	-	Bearbeitung einer Auftragstabelle	
					FALSE	An der Achse ist keine Auftragstabelle aktiv.
					TRUE	Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert.
ConstantVelocity	BOOL	FALSE	RCCP, RP	-	Konstante Geschwindigkeit	
					FALSE	Die Achse wird beschleunigt, abgebremst oder befindet sich im Stillstand.
					TRUE	Die Sollgeschwindigkeit wurde erreicht. Die Achse wird mit konstanter Geschwindigkeit verfahren.
Accelerating	BOOL	FALSE	RCCP, RP	-	Beschleunigungsvorgang	
					FALSE	Die Achse wird abgebremst, mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird beschleunigt.
Decelerating	BOOL	FALSE	RCCP, RP	-	Verzögerungsvorgang	
					FALSE	Die Achse wird beschleunigt, mit konstanter Geschwindigkeit ver-

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
						fahren oder befindet sich im Stillstand.
					TRUE	Die Achse wird abgebremst.
ControlPanelActive	BOOL	FALSE	RCCP, RP	-	Aktivierungsstatus der Achssteuertafel	
					FALSE	Die Betriebsart "Automatikbetrieb" ist aktiviert. Das Anwenderprogramm hat die Steuerungshoheit über die Achse.
					TRUE	Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerungshoheit über die Achse. Die Achse kann nicht vom Anwenderprogramm gesteuert werden.
DriveReady	BOOL	FALSE	RCCP, RP	-	Operationsstatus des Antriebs	
					FALSE	Der Antrieb ist nicht bereit. Sollwerte werden nicht ausgeführt.
					TRUE	Der Antrieb ist bereit. Sollwerte können ausgeführt werden.
RestartRequired	BOOL	FALSE	RCCP, RP	-	Restart der Achse erforderlich	
					FALSE	Ein Restart der Achse ist nicht erforderlich.
					TRUE	Werte wurden im Ladespeicher geändert.
					Um die Werte im Betriebszustand RUN der CPU in den Arbeitsspeicher zu laden, ist ein Restart der Achse erforderlich. Verwenden Sie hierzu die Motion Control-Anweisung "MC_Reset".	
SWLimitMinActive	BOOL	FALSE	RCCP, RP	-	Status des unteren Software-Endschalters	
					FALSE	Der konfigurierte Arbeitsbereich der Achse wird eingehalten.
					TRUE	Der untere Software-Endschalter wurde angefahren oder überfahren.
SWLimitMaxActive	BOOL	FALSE	RCCP, RP	-	Status des oberen Software-Endschalters	
					FALSE	Der konfigurierte Arbeitsbereich wird eingehalten.
					TRUE	Der obere Software-Endschalter wurde angefahren oder überfahren.
HWLimitMinActive	BOOL	FALSE	RCCP, RP	-	Status des unteren Hardware-Endschalters	
					FALSE	Der konfigurierte zulässige Verfahrbereich wird eingehalten.
					TRUE	Der untere Hardware-Endschalter wurde angefahren oder überfahren.
HWLimitMaxActive	BOOL	FALSE	RCCP, RP	-	Status des oberen Hardware-Endschalters	

Variable		Datentyp	Werte	Zugriff	W	Beschreibung	
						FALSE	Der konfigurierte zulässige Verfahrbereich wird eingehalten.
						TRUE	Der obere Hardware-Endschalter wurde angefahren oder überfahren.

Siehe auch

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen ErrorBits V4...5

Die Variablenstruktur <Achsname>.ErrorBits.<Variablenname> zeigt Fehler am Technologieobjekt an.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung
ErrorBits.	STRUCT				TO_Struct_ErrorBits
SystemFault	BOOL	FALSE	RCCP, RP	-	Interner Systemfehler
ConfigFault	BOOL	FALSE	RCCP, RP	-	Fehlerhafte Konfiguration der Achse
DriveFault	BOOL	FALSE	RCCP, RP	-	Fehler im Antrieb. Ausfall des "Antrieb bereit"-Signals.
SWLimit	BOOL	FALSE	RCCP, RP	-	Software-Endschalter angefahren oder überfahren
HWLimit	BOOL	FALSE	RCCP, RP	-	Hardware-Endschalter angefahren oder überfahren
DirectionFault	BOOL	FALSE	RCCP, RP	-	Unzulässige Bewegungsrichtung
HWUsed	BOOL	FALSE	RCCP, RP	-	Eine weitere Achse verwendet denselben PTO (Pulse Train Output) und ist mit "MC_Power" freigegeben.
SensorFault	BOOL	FALSE	RCCP, RP	-	Fehler im Gebersystem
CommunicationFault	BOOL	FALSE	RCCP, RP	-	Kommunikationsfehler Die Kommunikation mit einem verbundenen Gerät ist gestört.
FollowingErrorFault	BOOL	FALSE	RCCP, RP	-	Maximal zulässiger Schleppfehler überschritten
PositioningFault	BOOL	FALSE	RCCP, RP	-	Positionierfehler Die Achse wurde am Ende einer Positionierbewegung nicht korrekt positioniert.

Siehe auch

Status- und Fehlerbits (Technologieobjekte ab V4) (Seite 1499)

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen ControlPanel V4...5

Die Variablen "ControlPanel" beinhalten keine anwenderrelevanten Daten. Im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Variablen Internal V4...5

Die Variablen "Internal" beinhalten keine anwenderrelevanten Daten. Im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

Siehe auch

Variablen des Technologieobjekts Positionierachse ab V6 (Seite 1543)

Variablen des Technologieobjekts Achse V1...3 (Seite 1641)

Aktualisierung der Variablen des Technologieobjekts

Die in den Variablen des Technologieobjekts angezeigten Status- und Fehlerinformationen der Achse werden jeweils am Zykluskontrollpunkt aktualisiert.

Wertänderungen bei Konfigurations-Variablen werden nicht sofort wirksam. Unter welchen Bedingungen eine Änderung wirksam wird, entnehmen Sie der Detailbeschreibung der entsprechenden Variable.

Variablen des Technologieobjekts Auftragstabelle V1...3

Die Variablenstruktur <Auftragstabelle>.Config.Command.Command[x].<Variablenname> beinhaltet die konfigurierten Auftragsparameter.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Config.	STRUCT				TO_Struct_Config	
Command.	STRUCT				TO_Struct_Command	
Command[x].	STRUCT				ARRAY[1..32] TO_Struct_Command[x]	
Type	INT	0 (0 bis 151)	RW	-	Auftragstyp des Auftrags	
					0	Auftrag "Empty"
					2	Auftrag "Halt"
					5	Auftrag "Positioning Relative"
					6	Auftrag "Positioning Absolute"
					7	Auftrag "Velocity setpoint"
					151	Auftrag "Wait"
Position	REAL	0.0	RW	-	Zielposition/Verfahrweg des Auftrags	
Velocity	REAL	0.0	RW	-	Geschwindigkeit des Auftrags	
Duration	REAL	0.0	RW	-	Dauer des Auftrags	
BufferMode	INT	0 (0 bis 1)	RW	-	Wert für "Nächster Schritt" des Auftrags	
					0	"Auftrag abschließen"
					1	"Bewegung überschleifen"
StepCode	WORD	0	RW	-	Schrittcode des Auftrags	

Siehe auch

Variablen des Technologieobjekts Auftragstabelle V4...5 (Seite 1678)

Variablen des Technologieobjekts Auftragstabelle V6 (Seite 1579)

Variablen des Technologieobjekts Auftragstabelle V4...5

Die Variablenstruktur <Auftragstabelle>.Command[n].<Variablenname> beinhaltet die konfigurierten Auftragsparameter.

Variablen

Legende (Seite 1640)

Variable	Datentyp	Werte	Zugriff	W	Beschreibung	
Command[n].	STRUCT				ARRAY[1..32] TO_Struct_Command[n]	
Type	INT	0 (0 bis 151)	RW	-	Auftragstyp	
					0	Auftrag "Empty"
					2	Auftrag "Halt"
					5	Auftrag "Positioning Relative"
					6	Auftrag "Positioning Absolute"
					7	Auftrag "Velocity setpoint"
					151	Auftrag "Wait"
Position	REAL	0.0	RW	-	Zielposition/Verfahrweg des Auftrags	
Velocity	REAL	0.0	RW	-	Geschwindigkeit des Auftrags	
Duration	REAL	0.0	RW	-	Dauer des Auftrags	
NextStep	INT	0 (0 bis 1)	RW	-	Modus für den Übergang zum nächsten Auftrag	
					0	"Auftrag abschließen"
					1	"Bewegung überschleifen"
StepCode	WORD	0	RW	-	Schrittcode des Auftrags	

Siehe auch

Variablen des Technologieobjekts Auftragstabelle V1...3 (Seite 1677)

Variablen des Technologieobjekts Auftragstabelle V6 (Seite 1579)

6.1.2 S7-1500 Motion Control einsetzen

6.1.2.1 Einleitung

Zusammenspiel der verschiedenen Dokumentationen

Die Dokumentation der Motion Control-Funktionen ist für eine bessere Übersicht auf folgende Dokumentation aufgeteilt:

- S7-1500T Motion Control einsetzen
- S7-1500T Kinematikfunktionen einsetzen

Die Dokumentation "S7-1500 Motion Control einsetzen" beschreibt die Motion Control-Funktionen für folgende Technologieobjekte:

- Drehzahlachse
- Positionierachse
- Gleichlaufachse
- Externer Geber
- Messtaster
- Nocken
- Nockenspur
- Kurvenscheibe (S7-1500T)

Die Dokumentation "S7-1500T Kinematikfunktionen einsetzen" beschreibt die Motion Control-Funktionen für das Technologieobjekt Kinematik. Diese Dokumentation setzt die in "S7-1500 Motion Control einsetzen" beschriebenen Motion Control-Funktionen als bekannt voraus.

Integrierte Motion Control-Funktionalität

S7-1500 Motion Control unterstützt das geregelte Positionieren und Verfahren von Achsen und ist integrierter Bestandteil jeder CPU S7-1500 sowie jeder CPU S7-1500SP. Die Technologie-CPUs S7-1500T bieten erweiterte Funktionen.

Die Motion Control-Funktionalität unterstützt folgende Technologieobjekte:

- Drehzahlachse
- Positionierachse
- Gleichlaufachse
- Externer Geber
- Messtaster
- Nocken
- Nockenspur
- Kurvenscheibe (S7-1500T)
- Kinematik (S7-1500T)

PROFIdrive-fähige Antriebe, Antriebe mit analoger Sollwertschnittstelle und Schrittmotoren werden über standardisierte Motion Control-Anweisungen gemäß PLCopen gesteuert.

Die Achssteuertafel und umfangreiche Online- und Diagnosefunktionen unterstützen die einfache Inbetriebnahme und Optimierung von Antrieben.

S7-1500 Motion Control ist durchgehend in die Systemdiagnose der CPU S7-1500 eingebunden.

Funktionsweise von S7-1500 Motion Control

Mit dem TIA Portal erstellen Sie ein Projekt, konfigurieren Technologieobjekte und laden die Konfiguration in die CPU. Die Motion Control-Funktionalität wird in der CPU bearbeitet. Mit den Motion Control-Anweisungen in Ihrem Anwenderprogramm steuern Sie die Technologieobjekte. Zur Inbetriebnahme, Optimierung (Seite 2018) und Diagnose (Seite 2033) stellt das TIA Portal weitere Funktionen zur Verfügung.

Das folgende Bild zeigt schematisch die Anwenderschnittstellen und die Integration von Motion Control in die CPU S7-1500. Die Konzepte werden nachfolgend kurz erläutert:

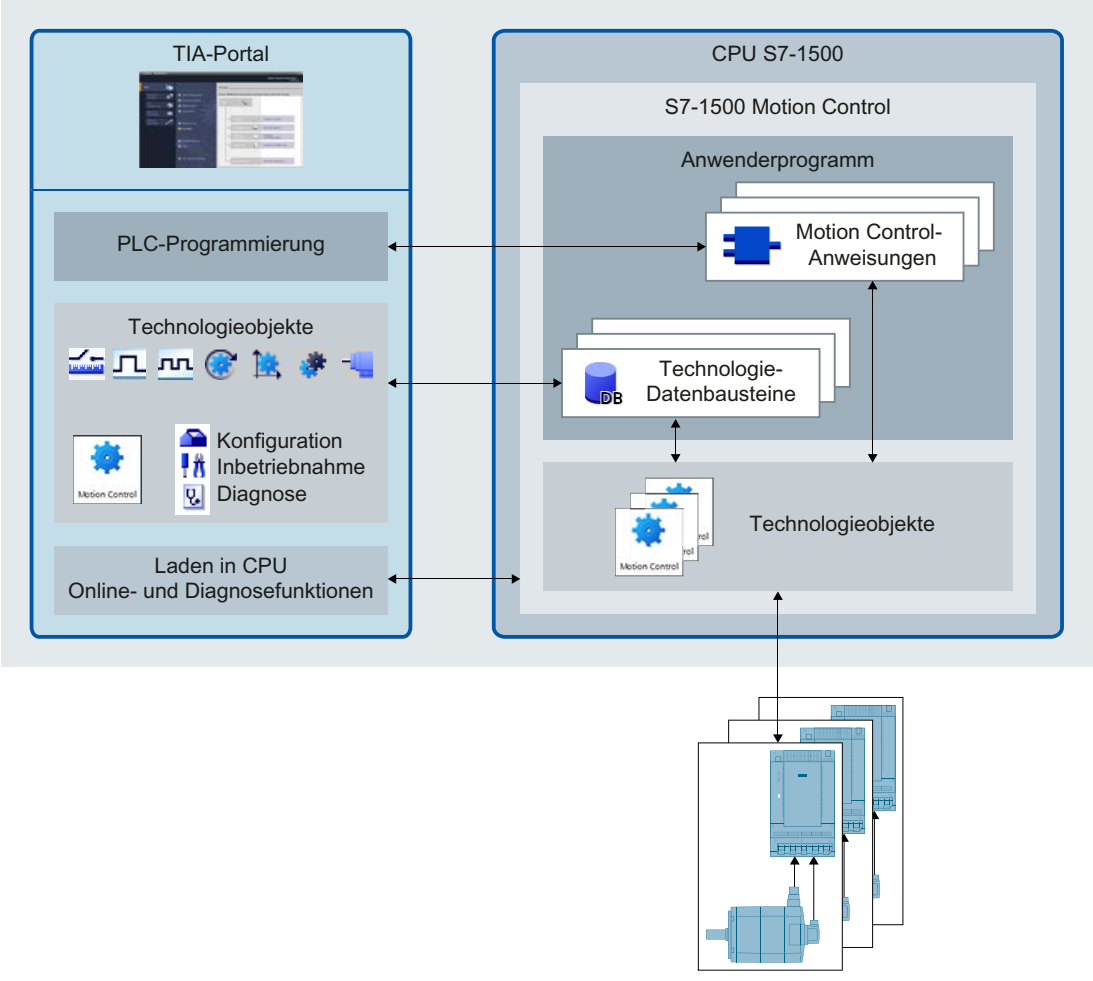

TIA Portal

Das TIA Portal unterstützt Sie bei der Projektierung und Inbetriebnahme der Motion Control-Funktionalität:

- Einbinden und Konfigurieren der Hardware
- Anlegen und Konfigurieren der Technologieobjekte
- Erstellen des Anwenderprogramms
- Laden in CPU
- Inbetriebnahme von Achsen
- Optimierung der Antriebe
- Diagnose

Mit dem TIA Portal projektieren Sie die Hardware, die Technologieobjekte sowie Ihr Anwenderprogramm. Das erstellte Projekt laden Sie in die CPU. Mit den Online- und Diagnosefunktionen des TIA Portals testen Sie Ihr Anwenderprogramm und diagnostizieren die Hardware.

Technologieobjekte

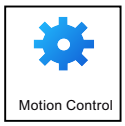

Technologieobjekte repräsentieren reale Objekte (z. B. einen Antrieb) in der Steuerung. Die Funktionen der Technologieobjekte rufen Sie über Motion Control-Anweisungen in Ihrem Anwenderprogramm auf. Die Technologieobjekte steuern bzw. regeln die Bewegung der realen Objekte und melden Statusinformation (z. B. die aktuelle Position) zurück.

Die Konfiguration der Technologieobjekte repräsentiert die Eigenschaften des realen Objekts. Die Konfigurationsdaten werden in einem Technologie-Datenbaustein gespeichert.

Folgende Technologieobjekte stehen für Motion Control zur Verfügung:

Symbol	Technologieobjekt	Beschreibung
	Drehzahlachse	Das Technologieobjekt Drehzahlachse ("TO_SpeedAxis") dient zur Vorgabe der Drehzahl für einen Antrieb. Die Bewegung der Achse programmieren Sie über Motion Control-Anweisungen.
	Positionierachse	Das Technologieobjekt Positionierachse ("TO_PositioningAxis") dient zum lagegeregelten Positionieren eines Antriebs. Mit Motion Control-Anweisungen erteilen Sie über Ihr Anwenderprogramm Positionieraufträge an die Achse.
	Gleichlaufachse	Das Technologieobjekt Gleichlaufachse ("TO_SynchronousAxis") enthält alle Funktionen des Technologieobjekts Positionierachse. Zusätzlich lässt sich die Achse mit einem Leitwert verschalten, sodass die Achse im Gleichlauf der Positionsänderung einer Leitachse folgt.
	Externer Geber	Das Technologieobjekt Externer Geber ("TO_ExternalEncoder") erfasst eine Position und stellt sie der Steuerung zur Verfügung. Die ermittelte Position lässt sich im Anwenderprogramm auswerten.
	Messtaster	Das Technologieobjekt Messtaster ("TO_MeasuringInput") erfasst Istpositionen schnell, genau und ereignisabhängig.
	Nocken	Das Technologieobjekt Nocken ("TO_OutputCam") erzeugt Schaltsignale abhängig von der Position einer Achse oder eines Externen Gebers. Die Schaltsignale können Sie im Anwenderprogramm auswerten oder auf digitale Ausgänge schalten.

Symbol	Technologieobjekt	Beschreibung
	Nockenspur	Das Technologieobjekt Nockenspur ("TO_CamTrack") erzeugt eine Schaltsignalfolge abhängig von der Position einer Achse oder eines Externen Gebers. Dabei werden bis zu 32 Einzelnocken überlagert und die Schaltsignale als Spur ausgegeben. Die Schaltsignale können Sie im Anwenderprogramm auswerten oder auf digitale Ausgänge schalten.
	Kurvenscheibe (S7-1500T)	Das Technologieobjekt Kurvenscheibe ("TO_Cam") definiert eine Funktion f(x) über Stützpunkte und/oder Segmente. Fehlende Funktionsbereiche werden interpoliert.
	Kinematik (S7-1500T)	Das Technologieobjekt Kinematik ("TO_Kinematics") dient zur Verschaltung von Positionierachsen und Gleichlaufachsen zu einer Kinematik. In der Konfiguration des Technologieobjekts Kinematik verschalten Sie die Achsen entsprechend des konfigurierten Kinematiktyps.

Technologie-Datenbaustein

Die Eigenschaften realer Objekte werden über die Technologieobjekte konfiguriert und in einem Technologie-Datenbaustein (Seite 1989) gespeichert. Der Technologie-Datenbaustein enthält alle Konfigurationsdaten, Soll- und Istwerte sowie Statusinformationen des Technologieobjekts. Das TIA Portal erzeugt beim Anlegen des Technologieobjekts automatisch den Technologie-Datenbaustein. Auf die Daten des Technologie-Datenbausteins greifen Sie lesend und schreibend über Ihr Anwenderprogramm zu.

Motion Control-Anweisungen

Mit den Motion Control-Anweisungen führen Sie die gewünschte Funktionalität an den Technologieobjekten aus. Die Motion Control-Anweisungen stehen im TIA Portal unter "Anweisungen > Technologie > Motion Control" zur Verfügung.

Die Motion Control-Anweisungen sind konform zu PLCopen (Version 2.0).

Anwenderprogramm

Die Motion Control-Anweisungen und der Technologie-Datenbaustein stellen die Programmierschnittstellen für die Technologieobjekte dar. Mit den Motion Control-Anweisungen übergeben Sie in Ihrem Anwenderprogramm Motion Control-Aufträge an die Technologieobjekte. Über die Ausgangsparameter der Motion Control-Anweisungen verfolgen Sie den Status laufender Aufträge. Über den Technologie-Datenbaustein rufen Sie Statusinformation des Technologieobjekts ab und ändern zur Laufzeit bestimmte Konfigurationsparameter.

Antriebe und Geber

Antriebe sorgen für die Bewegung der Achse. Sie werden in der Hardware-Konfiguration eingebunden.

Wenn Sie einen Motion Control-Auftrag in Ihrem Anwenderprogramm ausführen, übernimmt das Technologieobjekt die Ansteuerung des Antriebs und das Einlesen der Werte von Gebern.

PROFIdrive-fähige Antriebe und Geber werden über PROFIdrive-Telegramme angebunden. Folgende Anbindungen sind möglich:

- PROFINET IO
- PROFIBUS DP
- Technologiemodul (TM)

Antriebe mit analoger Sollwertschnittstelle werden über einen Analogausgang (AQ) und ein optionales Freigabesignal angebunden. Analoge Ein- und Ausgänge werden über entsprechende E/A-Baugruppen zur Verfügung gestellt.

Ein Antrieb wird auch als Aktor und ein Geber auch als Sensor bezeichnet.

Das folgende Bild zeigt eine Beispielkonfiguration, bei der alle Komponenten über PROFINET IO an die CPU angebunden sind:

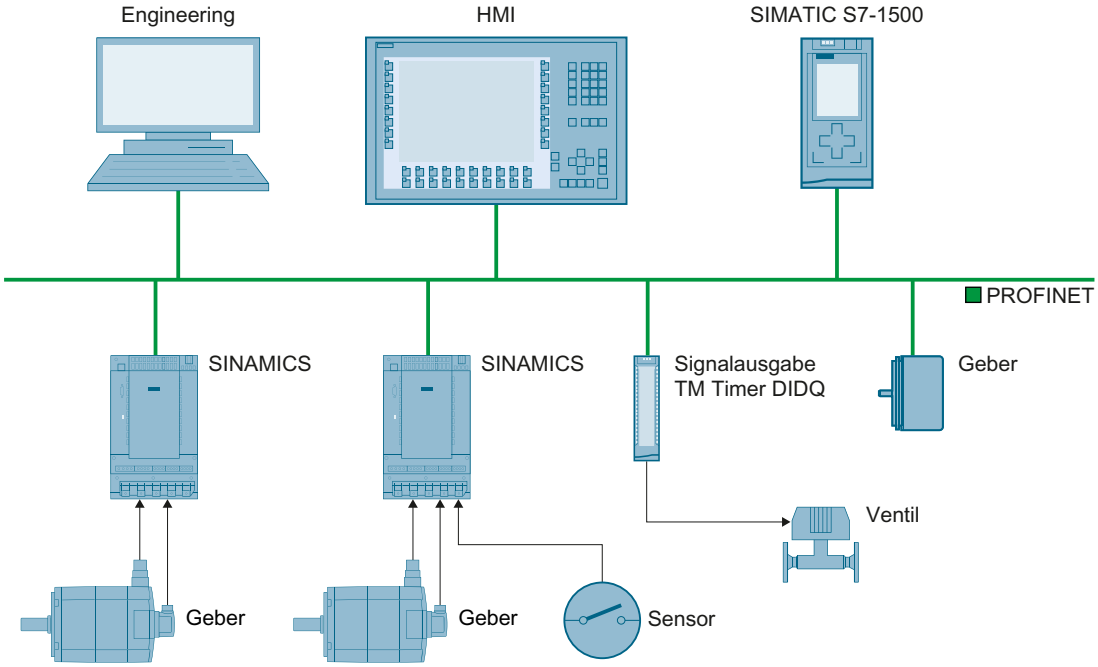

Funktionen

Funktionen - Achsen

Die Funktionen der Technologieobjekte Drehzahlachse, Positionierachse und Gleichlaufachse führen Sie über die Motion Control-Anweisungen in Ihrem Anwenderprogramm oder das TIA Portal (unter "Technologieobjekt > Inbetriebnahme") aus.

Die folgende Tabelle zeigt die von den Technologieobjekten unterstützten Funktionen:

Funktion	Technologieobjekt		
	Drehzahlachse (Seite 1762)	Positionierachse (Seite 1763)	Gleichlaufachse (Seite 1764)
Motion Control-Anweisungen (Anwenderprogramm)			
"MC_Power" Technologieobjekte freigeben, sperren	X	X	X
"MC_Home" Technologieobjekte referenzieren, Referenzpunkt setzen	-	X	X
"MC_MoveJog" Achsen im Tippbetrieb verfahren	X	X	X
"MC_MoveVelocity" Achsen mit Geschwindigkeitsvorgabe verfahren	X	X	X
"MC_MoveRelative" Achsen relativ positionieren	-	X	X
"MC_MoveAbsolute" Achsen absolut positionieren	-	X	X
"MC_MoveSuperimposed" Achsen überlagernd positionieren	-	X	X
"MC_GearIn" Getriebegleichlauf starten	-	-	X
"MC_Halt" Achsen anhalten	X	X	X
"MC_TorqueLimiting" Momentenbegrenzung über Momentenreduzierung	X	X	X
"MC_Reset" Alarme quittieren, Restart von Technologieobjekten	X	X	X
"MC_TorqueAdditive" Additives Moment vorgeben und aktivieren/deaktivieren	X	X	X
"MC_TorqueRange" Obere und untere Momentengrenze vorgeben	X	X	X
TIA Portal			
"Achssteuertafel (Seite 2020)" Verfahren und Referenzieren von Achsen über das TIA Portal	X	X	X
"Optimierung (Seite 2027)" Optimierung der Lageregelung	-	X	X

Funktionen - weitere Technologieobjekte

Die Funktionen der weiteren Technologieobjekte führen Sie über die Motion Control-Anweisungen in Ihrem Anwenderprogramm aus.

Die folgende Tabelle zeigt die von den Technologieobjekten unterstützten Motion Control-Anweisungen:

Motion Control-Anweisung	Technologieobjekt			
	Externer Geber (Seite 1780)	Messtaster (Seite 1781)	Nocken (Seite 1794)	Nockenspur (Seite 1808)
Motion Control-Anweisungen (Anwenderprogramm)				
"MC_Power" Technologieobjekte freigeben, sperren	X	-	-	-
"MC_Home" Technologieobjekte referenzieren, Referenzpunkt setzen	X	-	-	-
"MC_MeasuringInput" Einmaligen Messauftrag starten	-	X	-	-
"MC_MeasuringInputCyclic" Zyklischen Messauftrag starten	-	X	-	-
"MC_AbortMeasuringInput" Messauftrag beenden	-	X	-	-
"MC_OutputCam" Nocken aktivieren/deaktivieren	-	-	X	-
"MC_CamTrack" Nockenspur aktivieren/deaktivieren	-	-	-	X
"MC_Reset" Alarme quittieren, Restart von Technologieobjekten	X	X	X	X

Erweiterte Funktionen der Technologie-CPU

Die CPU S7-1500T bietet zusätzlich zur Funktionalität der CPU S7-1500 weitere Funktionen und die Technologieobjekte Kurvenscheibe und Kinematik:

Zusätzliche Funktion	Beschreibung
Mehrere Geber an Positionierachse/Gleichlaufachse (Seite 1700)	An eine Positionierachse/Gleichlaufachse lassen sich bis zu vier Geber anbinden. Die Geber lassen sich im Betrieb umschalten. Für die Lageregelung ist jeweils nur ein Geber aktiv.
Istwertkopplung (Seite 1767)	Alternativ zum Sollwert lässt sich der extrapolierte Istwert als Leitwert für den Gleichlauf verschalten. Somit lässt sich auch ein Technologieobjekt Externer Geber als Leitwert verwenden.
Getriebegleichlauf mit "MC_GearInPos" (Seite 1771)	Beim Getriebegleichlauf sind Leit- und Folgeachse wie bei einem mechanischen Getriebe durch eine lineare Gleichlauffunktion gekoppelt. Die Gleichlauffunktion geben Sie über den Getriebefaktor vor. Am "MC_GearInPos" lassen sich die Synchronpositionen von Leit- und Folgeachse vorgeben, ab denen die Achsen synchron verfahren.

Zusätzliche Funktion	Beschreibung
Technologieobjekt Kurvenscheibe (Seite 1821)	Das Technologieobjekt Kurvenscheibe ("TO_Cam") definiert eine Funktion f(x) über Stützpunkte und/oder Segmente. Lücken zwischen den definierten Stützpunkten und Segmenten der Kurvenscheibe werden zur Laufzeit des Anwenderprogramms durch Interpolation geschlossen.
Kurvenscheibengleichlauf (Seite 1773)	Beim Kurvenscheibengleichlauf sind Leit- und Folgeachse durch eine Gleichlauffunktion gekoppelt, die Sie über eine Kurvenscheibe vorgeben.
Vorlaufendes Aufsynchronisieren über Leitwertweg (Seite 1778) oder Dynamikparameter (Seite 1777)	Der Getriebegleichlauf mit "MC_GearInPos" und der Kurvenscheibengleichlauf mit "MC_CamIn" werden vorlaufend zu vorgebbaren Synchronpositionen aufsynchronisiert.
"MotionIn"-Funktionen (Seite 1721)	Mit den Motion Control-Anweisung "MC_MotionInVelocity" und "MC_MotionInPosition" geben Sie der Achse zyklisch applikativ berechnete Bewegungssollwerte als Basisbewegung vor. Dabei wird kein Geschwindigkeitsprofil berechnet, die Werte sind am Technologieobjekt direkt wirksam.
Technologieobjekt Kinematik	Das Technologieobjekt Kinematik ("TO_Kinematics") dient zur Verschaltung von Positionierachsen und Gleichlaufachsen zu einer Kinematik. In der Konfiguration des Technologieobjekts Kinematik verschalten Sie die Achsen entsprechend des konfigurierten Kinematiktyps.

Die erweiterten Funktionen der Technologie-CPU S7-1500T führen Sie über die Motion Control-Anweisungen in Ihrem Anwenderprogramm aus. Die folgende Tabelle zeigt die zusätzlich von den Technologieobjekten unterstützten Motion Control-Anweisungen:

Motion Control-Anweisung	Technologieobjekt		
	Positionierachse (Seite 1763)	Gleichlaufachse (Seite 1764)	Kurvenscheibe (Seite 1821)
"MC_SetSensor" Aktiven Geber umschalten	X	X	-
"MC_GearInPos" Getriebegleichlauf mit vorgegebenen Synchronpositionen starten	-	X	-
"MC_PhasingAbsolute" Absolute Verschiebung des Leitwerts beim Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos"	-	X	-
"MC_PhasingRelative" Relative Verschiebung des Leitwerts beim Getriebegleichlauf mit "MC_GearIn" und "MC_GearInPos"	-	X	-
"MC_CamIn" Kurvenscheibengleichlauf starten	-	X	-
"MC_InterpolateCam" Kurvenscheibe interpolieren	-	-	X
"MC_GetCamFollowingValue" Zu einem Leitwert den entsprechenden Folgewert mit der ersten und zweiten Ableitung an einer Kurvenscheibe auslesen	-	-	X
"MC_GetCamLeadingValue" Zu einem Folgewert den entsprechenden Leitwert an einer Kurvenscheibe auslesen	-	-	X

Motion Control-Anweisung	Technologieobjekt		
	Positionierachse (Seite 1763)	Gleichlaufachse (Seite 1764)	Kurvenscheibe (Seite 1821)
"MC_SynchronizedMotionSimulation" Gleichlauf in Simulation setzen und beim Sperren der Folgeachse aktiv halten	-	X	-
"MC_MotionInPosition" Bewegungssollwerte für Position, Geschwindigkeit und Beschleunigung vorgeben	X	X	-
"MC_MotionInVelocity" Bewegungssollwerte für Geschwindigkeit und Beschleunigung vorgeben	X[1)]	X	-

1) Gilt auch für das Technologieobjekt Drehzahlachse.

Weitere Informationen

Weitere Informationen zu den Motion Control-Anweisungen des Technologieobjekts Kinematik finden Sie im Funktionshandbuch "S7-1500T Kinematikfunktionen".

Siehe auch

S7-1500T Kinematikfunktionen einsetzen (https://support.industry.siemens.com/cs/ww/de/view/109749264)

Leitfaden zum Einsatz von Motion Control

Der hier beschriebene Leitfaden zeigt die grundsätzliche Vorgehensweise, um Motion Control mit der CPU S7-1500 einzusetzen. Dieser Leitfaden dient als Empfehlung.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500 ist angelegt.

Vorgehen

Gehen Sie zum Einsatz von Motion Control mit der CPU S7-1500 folgendermaßen vor:

1. Technologieobjekt hinzufügen (Seite 1843)
2. Arbeiten mit dem Konfigurationseditor (Seite 1846)
3. Programmieren (Seite 1988)
4. Laden in CPU (Seite 2017)
5. Funktionstest im Inbetriebnahmefenster (Seite 2018)
6. Diagnose (Seite 2033)

6.1.2.2 Technologieobjekte

Mengengerüst

Motion Control-Ressourcen

Jede CPU bietet eine definierte Menge an "Motion Control-Ressourcen". Die gesamt verfügbaren Motion Control-Ressourcen entnehmen Sie den technischen Daten der eingesetzten CPU.

Jedes Technologieobjekt belegt Motion Control-Ressourcen:

Technologieobjekt	Belegte Motion Control-Ressourcen
Drehzahlachse	40
Positionierachse	80
Gleichlaufachse	160
Externer Geber	80
Messtaster	40
Nocken	20
Nockenspur	160

Eine Übersicht über die Motion Control-Ressourcen einer CPU erhalten Sie im TIA Portal unter "Werkzeuge > Speicherauslastung".

Extended Motion Control-Ressourcen (S7-1500T)

Die Technologieobjekte Kurvenscheibe und Kinematik belegen "Extended Motion Control-Ressourcen". Die maximale Anzahl der zusätzlich zu den Motion Control-Ressourcen einsetzbaren Kurvenscheiben und Kinematiken entnehmen Sie den technischen Daten der eingesetzten CPU.

Technologieobjekt	Belegte Extended Motion Control-Ressourcen
Kurvenscheibe	2
Kinematik	30

Applikationszyklus

Mit steigender Anzahl eingesetzter Technologieobjekte benötigt die CPU mehr Rechenzeit zur Bearbeitung der Technologieobjekte. Der Motion Control-Applikationszyklus (Seite 1755) lässt sich entsprechend der Anzahl der eingesetzten Technologieobjekte anpassen.

Siehe auch

Extended Motion Control-Ressourcen (Seite 2224)

Grundlagen - Achsen

Achstypen

Achsen können mit unterschiedlichen Achstypen konfiguriert werden:

- Positionier- und Gleichlaufachsen können als rotatorische oder lineare Achse konfiguriert werden.
- Drehzahlachsen sind immer rotatorische Achsen.

Je nach Ausführung der Mechanik ist eine Achse als lineare Achse oder rotatorische Achse ausgeführt:

- Lineare Achse

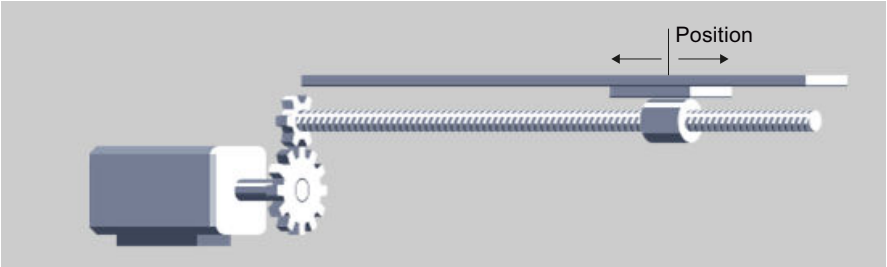

Bei linearen Achsen wird die Position der Achse als Längenmaß angegeben, z. B. Millimeter (mm).

- Rotatorische Achse

Bei rotatorischen Achsen wird die Position der Achse als Winkelmaß angegeben, z. B. Grad (°).

Maßeinheiten

Die unterstützten Maßeinheiten für die Drehzahl (Umdrehungen pro Zeiteinheit) sind 1/s, 1/min und 1/h.

Die folgende Tabelle zeigt die unterstützten Maßeinheiten für Position und Geschwindigkeit:

Position	Geschwindigkeit
nm, µm, mm, m, km	mm/s, mm/min, mm/h, m/s, m/min, m/h, km/min, km/h
in, ft, mi	in/s, in/min, ft/s, ft/min, mi/h
°, rad	°/s, °/min, rad/s, rad/min

Die Beschleunigung wird entsprechend als Maßeinheit der Position/s² eingestellt.

Der Ruck wird entsprechend als Maßeinheit der Position/s³ eingestellt.

Die folgende Tabelle zeigt die unterstützten Maßeinheiten für Kraft und Moment:

Kraft	Moment
N, kN	Nm, kNm
lbf, ozf, pdl	lbf in, lbf ft, ozf in, ozf ft, pdl in, pdl ft

Die Maßeinheit der Zeit ist für folgende Technologieobjekte fest vorgegeben:

Technologieobjekt	Zeit
Drehzahlachse, Positionier-/Gleichlaufachse, Externer Geber	s
Nocken, Nockenspur, Messtaster	ms

Hinweis

Beachten Sie bei der Einstellung oder Änderung der Maßeinheiten die Auswirkung auf die Anzeige der Parameterwerte und das Anwenderprogramm:

- Anzeige der Parameterwerte im Technologie-Datenbaustein
- Versorgung der Parameter im Anwenderprogramm
- Eingabe und Anzeige der Position und Geschwindigkeit im TIA Portal
- Sollwertvorgaben durch Leitachsen im Gleichlauf

Alle Angaben und Anzeigen erfolgen entsprechend der ausgewählten Maßeinheit.

Die eingestellten Einheiten werden in der Variablenstruktur "<TO>.Units" des Technologieobjekts angezeigt. Die Variablenstruktur ist im Anhang (Seite 2085) bei den Variablen des entsprechenden Technologieobjekts beschrieben.

Moduloeinstellung

Für die Technologieobjekte Positionierachse, Gleichlaufachse und Externer Geber kann die Einstellung "Modulo" aktiviert werden.

Wenn eine Achse in nur einer Richtung verfahren wird, wird der Positionswert stetig größer. Um den Positionswert auf ein wiederkehrendes Bezugssystem zu beschränken, kann die Einstellung "Modulo" aktiviert werden.

Bei aktiviertem "Modulo" wird der Positionswert des Technologieobjekts auf einen sich wiederholenden Modulobereich abgebildet. Der Modulobereich ist durch den Startwert und die Länge definiert.

Um z. B. den Positionswert einer rotatorischen Achse auf eine volle Kreisbewegung zu beschränken, kann der Modulobereich mit Startwert = 0° und Länge = 360° definiert werden. Damit wird der Positionswert auf den Modulobereich 0° bis 359,999° abgebildet.

Langzeitgenauigkeit

Langzeitgenauigkeit bedeutet, dass die technologische Position aus den aufgelaufenen Geberinkrementen eindeutig bestimmbar (ohne Rundungsfehler) und damit immer genau ist.

Die Langzeitgenauigkeit stellt Folgendes sicher:

- Die von der Steuerung angezeigte Achsposition entspricht immer der realen Position. Das heißt, die Achsposition ist eindeutig aus den aufgelaufenen Geberinkrementen darstellbar.
- Die berechnete Position in der Steuerung weicht nicht von der realen Position ab. Ohne Langzeitgenauigkeit ergeben sich Abweichungen, z. B. infolge von Rundungsfehlern oder Umrechnungen mit endlicher Genauigkeit.

Achsen ohne Modulo-Einstellung sind bis zum Erreichen der numerischen Verfahrbereichsgrenze langzeitstabil. Achsen mit Modulo-Einstellung sind dementsprechend mindestens so lange stabil, wie sie den numerischen Verfahrbereich von 9.0E12 mm bei 1000 Inkrementen pro Einheit nicht überschreiten.

Mit folgender Gleichung können Sie abschätzen, nach welcher Zeit der Verfahrbereich überschritten wird.

$$\text{Verfahrzeit} = \frac{\text{Maximalposition}}{\text{Geschwindigkeit}}$$

Beispiel zur maximalen Verfahrzeit

Maximalposition = 9.0E12 mm bei einer Auflösung von 1000 Ink/mm

Geschwindigkeit = 20.0 m/min = 2.0E4 mm/min

$$\text{Verfahrzeit} = \frac{\text{9.0E12 mm}}{\text{2.0E4 mm/min}} = \text{4.5E8 min} \quad \triangleq \text{856 Jahre}$$

Die Verfahrzeit beginnt neu, wenn eine der folgenden Bedingungen erfüllt ist:

- Sie haben die Achse mit "MC_Home" referenziert.
- Sie haben die Geschwindigkeit der Achse verändert.
  Eine Änderung der Geschwindigkeit hat zur Folge, dass sich die Verfahrzeit dementsprechend ändert.

Vermeidung der Überschreitung des Verfahrbereichs

Um einer möglichen Überschreitung des Verfahrbereichs vorzubeugen, führen Sie folgende Maßnahmen vor Ablauf der maximalen Verfahrzeit durch:

- Inkrementalgeber: Referenzieren Sie den Inkrementalgeber erneut.
- Absolutwertgeber: Führen Sie eine Absolutwertgeberjustage mit Vorgabe der aktuell bekannten Position durch.

Antriebs- und Geberanbindung

Dem Technologieobjekt "Drehzahlachse" wird ein Antrieb zugeordnet. Dem Technologieobjekt "Positionierachse" und "Gleichlaufachse" werden ein Antrieb und ein bis vier Geber (nur bei S7-1500T) zugeordnet. Dem Technologieobjekt "Externer Geber" wird ein Geber zugeordnet.

Der Sollwert an den Antrieb wird entweder über PROFIdrive-Telegramme oder über einen Analogausgang vorgegeben.

Für einen Geber sind folgende Anschlussmöglichkeiten gegeben:

- Geber am Antrieb
- Geber am Technologiemodul
- PROFIdrive-Geber direkt am PROFIBUS DP/PROFINET IO

Der Geberistwert wird ausschließlich über PROFIdrive-Telegramme übertragen.

PROFIdrive

PROFIdrive ist das genormte Standardprofil für Antriebstechnik bei der Anbindung von Antrieben und Gebern über PROFIBUS DP und PROFINET IO. Antriebe, die das PROFIdrive-Profil unterstützen, werden gemäß der PROFIdrive-Norm angebunden.

Die aktuelle PROFIdrive-Spezifikation finden Sie unter:

http://www.profibus.com (http://www.profibus.com)

Die Kommunikation zwischen Steuerung und Antrieb/Geber erfolgt über verschiedene PROFIdrive-Telegramme. Die Telegramme haben jeweils einen normierten Aufbau. Je nach Anwendung können Sie das passende Telegramm auswählen. In den PROFIdrive-Telegrammen werden Steuer- und Zustandsworte sowie Soll- und Istwerte übertragen.

Das PROFIdrive-Profil unterstützt ebenfalls das Regelungskonzept "Dynamic Servo Control" (DSC). DSC nutzt die schnelle Lageregelung im Antrieb. Damit lassen sich hochdynamische Motion Control-Aufgaben lösen.

Analoge Antriebsanbindung

Antriebe mit analoger Sollwertschnittstelle werden über einen Analogausgang und ein optionales Freigabesignal angebunden. Der Drehzahlsollwert wird über ein analoges Ausgangssignal (z. B. -10 V bis 10 V) der CPU vorgegeben.

Schrittmotoren

Die Anbindung von Antrieben mit einer Schrittmotoren-Schnittstelle erfolgt über Telegramm 3 und mithilfe von PTO (Pulse Train Output) Impulsgeneratoren.

Zur funktionalen Unterstützung des Schrittmotorbetriebs ist eine Quantisierung der Regeldifferenz einstellbar.

Durch die Vorgabe einer Quantisierung wird ein Bereich um die Zielposition festgelegt, in dem kein Ausregeln der Istposition erfolgen soll. Damit wird ein mögliches Pendeln des

Schrittmotors um die Zielposition verhindert. Zwei Arten der Quantisierung können eingestellt werden:

- Eine Quantisierung der Regeldifferenz entsprechend der Geberauflösung ("<TO>.PositionControl.ControlDifferenceQuantization.Mode" = 1)
  Damit wird z. B. ein Pendeln des Motors im Stillstand zwischen zwei Inkrementwerten verhindert. Dieser Mode ist insbesondere bei Verwendung von mehreren Gebern hilfreich. Bei dieser Einstellung wird die Quantisierung bei Geberumschaltung entsprechend angepasst. Dieser Mode ist bei Schrittmotoren mit Gebern, bei welchen die Auflösung des Gebers niedriger ist als die Schrittweite des Schrittmotors, hilfreich.
- Direkte Vorgabe eines Werts für die Quantisierung der Regeldifferenz ("<TO>.PositionControl.ControlDifferenceQuantization.Mode" = 2, Wertvorgabe in "<TO>.PositionControl.ControlDifferenceQuantization.Value")
  Dieser Mode ist bei Schrittmotoren mit Gebern, bei welchen die Auflösung des Gebers höher ist als die Schrittweite des Schrittmotors, hilfreich.

PROFIdrive-Telegramme

Über PROFIdrive-Telegramme werden Soll- und Istwerte, Steuer- und Zustandsworte sowie weitere Parameter zwischen Steuerung und Antrieb bzw. Geber übertragen.

Bei Anschaltung über PROFIdrive-Telegramm werden die Antriebe und Geber entsprechend dem PROFIdrive-Profil hantiert und eingeschaltet.

Die folgende Tabelle zeigt für verschiedene Technologieobjekte die möglichen PROFIdrive-Telegramme:

Technologieobjekt	Mögliche PROFIdrive-Telegramme
Drehzahlachse	• 1, 2 • 3, 4, 5, 6, 102, 103, 105, 106 (Geberistwert wird nicht ausgewertet)
Positionierachse/Gleichlaufachse	
Sollwert und Geberistwert in einem Antriebstelegramm	3, 4, 5, 6, 102, 103, 105, 106
Sollwert und Geberistwert getrennt	
Sollwert in Antriebstelegramm	1, 2, 3, 4, 5, 6, 102, 103, 105, 106
Istwert aus Telegramm	81, 83
Externer Geber	81, 83
Messtaster [1)]	391, 392, 393

1) Bei Verwendung von SINAMICS-Antrieben (Messen über SINAMICS Messtastereingang)

Telegrammtypen

Die folgende Tabelle zeigt unterstützte PROFIdrive-Telegrammtypen für die Antriebs- und Geberzuordnung:

Telegramm	Kurzbeschreibung
Standardtelegramme	
1[1)]	• Steuerwort STW1, Zustandswort ZSW1 • Drehzahlsollwert 16 Bit (NSOLL), Drehzahlistwert 16 Bit (NIST)
2	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST)
3	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2)
4	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) • Geberistwert 2 (G2_XIST1, G2_XIST2)
5	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) (Motorgeber) • Dynamic Servo Control (DSC)[2)] – Drehzahl-Vorsteuerwert – Positionsdifferenz (XERR) – Kpc - Geschwindigkeitsvorsteuerung der Lageregelung
6	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) (Motorgeber) • Geberistwert 2 (G2_XIST1, G2_XIST2) • Dynamic Servo Control (DSC)[2)] – Drehzahl-Vorsteuerwert – Positionsdifferenz (XERR) – Kpc - Geschwindigkeitsvorsteuerung der Lageregelung
SIEMENS-Telegramme (mit Momentenbegrenzung)	
102	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) • Momentenbegrenzung
103	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) • Geberistwert 2 (G2_XIST1, G2_XIST2) • Momentenbegrenzung

Telegramm	Kurzbeschreibung
105	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) (Motorgeber) • Dynamic Servo Control (DSC)[2)] – Drehzahl-Vorsteuerwert – Positionsdifferenz (XERR) – Kpc - Geschwindigkeitsvorsteuerung der Lageregelung • Momentenbegrenzung
106	• Steuerworte STW1 und STW2, Zustandsworte ZSW1 und ZSW2 • Drehzahlsollwert 32 Bit (NSOLL), Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2) (Motorgeber) • Geberistwert 2 (G2_XIST1, G2_XIST2) • Dynamic Servo Control (DSC)[2)] – Drehzahl-Vorsteuerwert – Positionsdifferenz (XERR) – Kpc - Geschwindigkeitsvorsteuerung der Lageregelung • Momentenbegrenzung
SIEMENS-Zusatztelegramme (Momentendaten)	
750[3)]	• Additives Sollmoment • Obere und untere Momentengrenze • Drehmomentenistwert
SIEMENS-Telegramme (Messtaster) [4)]	
391	• Steuerwort CU_STW1, Zustandswort CU_ZSW1 • Messtaster Steuerwort (MT_STW), Messtaster Zustandswort (MT_ZSW) • Messtaster Zeitstempel der fallenden (MT1…2_ZS_F) bzw. steigenden Flanken (MT1…2_ZS_S) • Digitalausgang 16 Bit, Digitaleingang 16 Bit
392	• Steuerwort CU_STW1, Zustandswort CU_ZSW1 • Messtaster Steuerwort (MT_STW), Messtaster Zustandswort (MT_ZSW) • Messtaster Zeitstempel der fallenden (MT1…6_ZS_F) bzw. steigenden Flanken (MT1…6_ZS_S) • Digitalausgang 16 Bit, Digitaleingang 16 Bit
393	• Steuerwort CU_STW1, Zustandswort CU_ZSW1 • Messtaster Steuerwort (MT_STW), Messtaster Zustandswort (MT_ZSW) • Messtaster Zeitstempel der fallenden (MT1…8_ZS_F) bzw. steigenden Flanken (MT1…8_ZS_S) • Digitalausgang 16 Bit, Digitaleingang 16 Bit • Analogeingang 16 Bit
Standardtelegramme Geber	

Telegramm	Kurzbeschreibung
81	• Steuerwort STW2_ENC, Zustandswort ZSW2_ENC • Geberistwert 1 (G1_XIST1, G1_XIST2)
83	• Steuerwort STW2_ENC, Zustandswort ZSW2_ENC • Drehzahlistwert 32 Bit (NIST) • Geberistwert 1 (G1_XIST1, G1_XIST2)

1) Kein taktsynchroner Betrieb möglich

2) Für den Einsatz von Dynamic Servo Control (DSC) muss der Motorgeber (erster Geber im Telegramm) des Antriebs als erster Geber für das Technologieobjekt verwendet werden.

3) Zusätzlich zu den Telegrammen 1, 2, 3, 4, 5, 6, 102, 103, 105, 106 verwendbar

4) Bei Verwendung von SINAMICS-Antrieben (Messen über SINAMICS Messtastereingang)

Siehe auch

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Istwerte

Für das lagegeregelte Verfahren und Positionieren muss der Steuerung der Lageistwert bekannt sein.

Der Lageistwert wird über ein PROFIdrive-Telegramm bereitgestellt.

Die Istwerte werden im PROFIdrive-Telegramm inkrementell oder absolut dargestellt. Die Istwerte werden in der Steuerung unter Berücksichtigung der Konfiguration der Mechanik auf die technologische Einheit normiert. Durch Referenzieren wird der Bezug zu einer physikalischen Position der Achse oder des Externen Gebers hergestellt.

Die Steuerung unterstützt folgende Istwertarten (Gebertypen):

- Inkrementeller Istwert
- Absoluter Istwert mit der Einstellung absolut (Messbereich > Verfahrbereich Achse)
- Absoluter Istwert mit der Einstellung zyklisch absolut (Messbereich < Verfahrbereich Achse)

Istwertberechnung bei virtueller Achse oder Achse in Simulation

Der Istwert einer virtuellen Achse oder einer Achse in Simulation wird aus dem Sollwert unter Berücksichtigung von Zeitverzügen gebildet.

Der jeweilige Zeitverzug vom Istwert zum Sollwert (T_t) ergibt sich wie folgt:

Berechnung	
Mit Vorsteuerung	$T_t = T_{ipo} + T_{servo} + T_{vtc} + T_{addPtc}$
Ohne Vorsteuerung, ohne DSC	$T_t = T_{ipo} + 1/Kv + T_{addPtc}$
Ohne Vorsteuerung, mit DSC bei einer Achse in Simulation	$T_t = T_{ipo} + T_{servo} + 1/Kv + T_{addPtc}$

T_t	Zeitverzug vom Istwert zum Sollwert
T_{ipo}	Rechenzeit des MC-Interpolator [OB92]
T_{servo}	Rechenzeit des MC-Servo [OB91]
T_{vtc}	Drehzahl-Regelkreis-Ersatzzeit (T_{vtc} aus "<TO>.DynamicAxisModel.VelocityTime-Constant")
T_{addPtc}	Additive Positions-Regelkreis-Ersatzzeit (T_{addPtc} aus "<TO>.DynamicAxisModel.Additional-PositionTimeConstant")
Kv	Verstärkungsfaktor (Kv aus "<TO>.PositionControl.Kv")

Siehe auch

Virtuelle Achse (Seite 1707)

Achse in Simulation (Seite 1706)

Inkrementeller Istwert

Der Istwert im PROFIdrive-Telegramm basiert auf einem inkrementellen Wert.

Nach NETZ-EIN wird Position null angezeigt. Mit dem Übergang in den Betriebszustand RUN der CPU beginnt die Istwertaktualisierung. Danach wird der Istwert auch im Betriebszustand STOP der CPU aktualisiert. Der Bezug zwischen dem Technologieobjekt und der mechanischen Position muss durch Referenzieren neu hergestellt werden.

Absoluter Istwert

Der Istwert im PROFIdrive-Telegramm basiert auf einem absoluten Wert.

Nach NETZ-EIN wird Position null angezeigt. Mit dem ersten Übergang in den Betriebszustand RUN der CPU beginnt die Istwertaktualisierung. Danach wird der Istwert auch im Betriebszustand STOP der CPU aktualisiert. Über die Absolutwertgeberjustage wird der gelieferte Absolutwert der dazugehörigen mechanischen Achsposition zugeordnet. Die Absolutwertgeberjustage muss einmalig vorgenommen werden. Der Absolutwertoffset wird über das Ein-/Ausschalten der Steuerung hinweg remanent gespeichert.

Unterscheidung der Absolutwerte:

- Der Messbereich des Gebers ist größer als der Verfahrbereich der Achse: Absolutwert mit Einstellung absolut
- Der Messbereich des Gebers ist kleiner als der Verfahrbereich der Achse: Absolutwert mit Einstellung zyklisch absolut

Absoluter Istwert mit der Einstellung absolut (Messbereich > Verfahrbereich)

Die Achsposition ergibt sich direkt aus dem aktuellen Geberistwert. Der Verfahrbereich muss innerhalb eines Gebermessbereichs liegen. Das heißt, dass der Nulldurchgang des Gebers nicht im Verfahrbereich liegen darf.

Beim Einschalten der Steuerung wird die Achsposition aus dem absoluten Geberistwert ermittelt.

Absoluter Istwert mit der Einstellung zyklisch absolut (Messbereich < Verfahrbereich)

Der Geber liefert innerhalb seines Messbereichs einen absoluten Wert. Die Steuerung zählt die durchlaufenen Messbereiche mit und ermittelt so auch über den Messbereich hinaus die korrekte Achsposition.

Beim Ausschalten der Steuerung werden die durchlaufenen Messbereiche im remanenten Speicherbereich der Steuerung gespeichert.
Beim nächsten Einschalten werden die gespeicherten Überläufe in der Berechnung des Lageistwerts berücksichtigt.

ACHTUNG
Bewegungen der Achse bei ausgeschalteter Steuerung können den Istwert verfälschen Wenn bei ausgeschalteter Steuerung die Achse bzw. der Geber um mehr als den halben Gebermessbereich bewegt wird, dann stimmt der Istwert in der Steuerung nicht mehr mit der mechanischen Achsstellung überein.

Siehe auch

Absolutwertgeberjustage (Seite 1746)

Variablen

Für die Anpassung der Istwerte sind die im Kapitel "Referenzieren (Seite 1748)" genannten Variablen relevant.

Automatische Übernahme der Antriebs- und Geberparameter im Gerät

Für den Betrieb müssen die Bezugsgrößen für die Antriebs- und Geberanbindung in der Steuerung und im Antrieb bzw. Geber identisch eingestellt sein.

Der Drehzahlsollwert NSOLL und der Drehzahlistwert NIST werden im PROFIdrive-Telegramm als Prozentwert bezogen auf die Bezugsdrehzahl übertragen. Der Bezugswert für die Drehzahl muss in der Steuerung und im Antrieb identisch eingestellt sein.

Die Auflösung des Istwertes im PROFIdrive-Telegramm muss ebenfalls in der Steuerung und im Antrieb bzw. Gebermodul identisch eingestellt sein.

Automatische Übernahme von Parametern

Für folgende Antriebe und Geber können die Antriebs- bzw. Geberparameter automatisch in die CPU übernommen werden:

- SINAMICS-Antriebe (siehe Kompatibilitätsliste (Seite 2222))
- PROFIdrive-Geber ab Ausgabestand A16

Die entsprechenden Parameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder (Wieder-)Anlauf des Antriebs oder der CPU übernommen. Änderungen in der Konfiguration des Antriebs werden nach Wiederanlauf des Antriebs oder Restart des Technologieobjekts übernommen.

Die erfolgreiche Übernahme der Parameter kann in der Steuerung über den Wert der Variablen "<TO>.StatusDrive.AdaptionState" = 2 und "<TO>.StatusSensor[n].AdaptionState" = 2 des Technologieobjekts überprüft werden.

Parameter

Die Einstellungen der Steuerung werden im TIA Portal unter "Technologieobjekt > Konfiguration > Hardware-Schnittstelle > Datenaustausch Antrieb/Geber" vorgenommen.

Die Einstellungen für Antrieb und Geber werden bei der Konfiguration der jeweiligen Hardware vorgenommen.

Folgende Tabelle stellt die Einstellungen im TIA Portal, in der Steuerung und die entsprechenden Antriebs-/Geberparameter gegenüber:

Einstellung im TIA Portal	Steuerung Variable im Technologie-Datenbaustein		Antriebsparameter	Automatische Übernahme
Antrieb				
Telegrammnummer	Eingangsadresse Telegramm <TO>.Actor.Interface.AddressIn		Telegrammnummer P922	-
	Ausgangsadresse Telegramm <TO>.Actor.Interface.AddressOut			
Bezugsdrehzahl in [1/min]	<TO>.Actor.DriveParameter.Reference-Speed		(SINAMICS-Antriebe: P2000)	X
Maximale Drehzahl des Motors in [1/min]	<TO>.Actor.DriveParameter.MaxSpeed		(SINAMICS-Antriebe: P1082)	X
Bezugsmoment in [Nm]	<TO>.Actor.DriveParameter.Reference-Torque		(SINAMICS-Antriebe: P2003)	X
Geber				
Telegramm	<TO>.Sensor[n].Interface.AddressIn		P922	-
	<TO>.Sensor[n].Interface.Addressout			
Gebertyp	<TO>.Sensor[n].Type		P979[5] Geber 1 P979[15] Geber 2	-
	0	Inkrementell		
	1	Absolut		
	2	Zyklisch absolut		
Messsystem	<TO>.Sensor[n].System		P979[1] Bit0 Geber 1 P979[11] Bit0 Geber 2	X
	0	Linear		
	1	Rotatorisch		
Auflösung (linearer Geber) Die Gitterteilung ist auf dem Typenschild des Gebers als Abstand der Striche auf dem linearen Messsystem angegeben.	<TO>.Sensor[n].Parameter.Resolution		P979[2] Geber 1 P979[12] Geber 2	X
Inkremente pro Umdrehung (rotatorischer Geber)	<TO>.Sensor[n].Parameter.StepsPerRe-volution		P979[2] Geber 1 P979[12] Geber 2	X
Anzahl Bits für die Feinauflösung XIST1 (zyklischer Geberistwert, linearer oder rotatorischer Geber)	<TO>.Sensor[n].Parameter.FineResoluti-onXist1		P979[3] Geber 1 P979[13] Geber 2	X

Einstellung im TIA Portal	Steuerung Variable im Technologie-Datenbaustein	Antriebsparameter	Automatische Übernahme
Anzahl Bits für die Feinauflösung XIST2 (Absolutwert des Gebers, linearer oder rotatorischer Geber)	<TO>.Sensor[n].Parameter.FineResolutionXist2	P979[4] Geber 1 P979[14] Geber 2	X
Unterscheidbare Geberumdrehungen (rotatorischer Absolutwertgeber)	<TO>.Sensor[n].Parameter.DeterminableRevolutions	P979[5] Geber 1 P979[15] Geber 2	X

Mehrere Geber verwenden

Die Technologie-CPU S7-1500T bietet die Möglichkeit je Positionier- und Gleichlaufachse bis zu 4 Geber-, bzw. Messsysteme als Istposition für die Lageregelung zu verwenden.

Für die Lageregelung ist jeweils nur ein Geber aktiv. Die 4 Geber-, bzw. Messsysteme können alternativ eingesetzt werden.

Allerdings können die Istwerte aller konfigurierten Geber im Anwenderprogramm ausgewertet werden.

Damit bieten sich z. B. folgende mögliche Einsatzgebiete:

- Einsatz von zusätzlichen Maschinengebern (neben dem Motorgeber), z. B. als direkte Messsysteme zur genaueren Erfassung der Istpositionen von Bearbeitungsprozessen.
- Einsatz von alternativen Gebersystemen bei Werkzeugwechsel in der flexiblen Fertigung.

Konfigurieren Sie die Geber in der Konfiguration der Achse. Die Umschaltung der Geber steuern Sie im Anwenderprogramm mit der Motion Control-Anweisung "MC_SetSensor".

Achse mit mehreren Gebern konfigurieren

Beachten Sie beim Einsatz mehrerer Geber folgende Konfigurationsfenster:

- Konfigurieren Sie im Konfigurationsfenster "Hardware-Schnittstelle > Geber", welche Geber alternativ verwendet werden sollen und welchem Gebertyp (Inkrementell, Absolut oder Zyklisch absolut) sie entsprechen.
  Alle als verwendet markierten Geber liefern unabhängig ihrer Nutzung zur Lageregelung laufend aktuelle Istwerte.
- Konfigurieren Sie im Konfigurationsfenster "Hardware-Schnittstelle > Geber" einen Geber als "Geber beim Hochlauf". Dies ist notwendig, da der Positionier- und Gleichlaufachse immer ein Geber zugeordnet sein muss.
- Konfigurieren Sie im Konfigurationsfenster "Hardware-Schnittstelle > Datenaustausch Geber" weitere Geberdetails und über welches Telegramm die Geber angebunden werden sollen. Die Konfiguration muss für jeden verwendeten Geber ausgeführt werden.
  Jeder zu verwendende Geber, bzw. jedes Messsystem darf in seiner Geberanbauart unterschiedlich ausgeprägt sein.

- Konfigurieren Sie im Konfigurationsfenster "Erweiterte Parameter > Mechanik" die Geberanbauart und gegebenfalls Getriebeparameter. Die Konfiguration muss für jeden verwendeten Geber ausgeführt werden.
- Die Achse kann mit jedem konfigurierten Geber referenziert werden. Konfigurieren Sie im Konfigurationsfenster "Erweiterte Parameter > Referenzieren" die Parameter zum aktiven und passiven Referenzieren der Achse. Die Konfiguration kann für jeden verwendeten Geber ausgeführt werden.
  Mit dem Referenzieren der Achse mit einem Geber ist die Achse referenziert und behält bei Geberumschaltung den Status "referenziert" bei.

Geberumschaltung im Anwenderprogramm

Zur Lageregelung der Positionier- und Gleichlaufachse muss immer ein Geber aktiv sein. Einzelne Geber dürfen ausfallen, solange sie nicht an der Lagerregelung beteiligt sind.

Mit der Motion Control-Anweisung "MC_SetSensor" schalten Sie den Geber für die Lageregelung der Achse um.

Die Umschaltung kann während eines laufenden Bewegungsauftrags oder im Stillstand erfolgen. Die Achse muss nicht freigegeben sein.

Eine Umschaltung während eines laufenden Referenzier- oder Restartauftrags ist nicht möglich.

Hinweis

Referenzieren

Referenzieren mit der Motion Control-Anweisung "MC-Home" oder der Achssteuertafel erfolgt immer mit dem an der Lageregelung beteiligten Geber.

Der Referenzierstatus der Achse wird bei einer Geberumschaltung nicht geändert.

Simulation

Bei der Simulation der Achse werden alle als "verwendet" konfigurierten Geber simuliert.

Bei der Umschaltung auf einen alternativen Geber, bzw. Gebersystem können Sie auswählen, wie mit einem Unterschied der Istpositionen der Geber umgegangen werden soll.

Über den Eingangsparameter "Mode" der Motion Control-Anweisung "MC_SetSensor" bestimmen Sie, wie mit der Differenz in den Istpositionen der Geber umgegangen werden soll.

- **Geber umschalten und aktuelle Istposition auf den umzuschaltenden Geber übertragen ("Mode" = 0)**
  Bei dieser Geberumschaltung werden Sprünge in der Istposition verhindert. Eine stoßfreie Umschaltung der Geber ist möglich.
- **Geber umschalten ohne die Istposition zu übertragen ("Mode" = 1)**
  Beim Umschalten auf einen Geber ohne Anpassung kann ein Sprung der Istposition auftreten. Dies kann gewünscht sein, wenn durch den neuen Geber gegebenenfalls mechanische Einflüsse (z. B. Schlupf) in der Positionierung kompensiert werden sollen. Die Positionsdifferenz wird nicht unmittelbar, sondern über die Zeitkonstante "<TO>.PositionControl.SmoothingTimeByChangeDifference" verzögert umgesetzt, um Sprünge in der Istposition bei aktiver Lageregelung zu verhindern.

- **Istposition übertragen ("Mode" = 2)**
  Die Istposition der Achse wird auf den am Parameter "Sensor" angegebenen Geber übertragen.
- **Istposition des Referenzgebers übertragen ("Mode" = 3)**
  Die Istposition des "Referenzgebers" (Parameter "ReferenceSensor") wird auf den am Parameter "Sensor" angegebenen Geber übertragen.

"Mode" = 2 und 3 können zum Vorbereiten einer Umschaltung dienen.

Safety-Funktionen im Antrieb

Safety-Funktionen ("Safety Integrated Basic Functions") im SINAMICS-Antrieb sind sicherheitsgerichtete antriebsinterne Funktionen mit dem Ziel des sicheren Stillsetzens des betreffenden Antriebs. Zusätzlich stehen zur Überwachung von vorgebbaren Grenzen weitere Sicherheitsfunktionen ("Safety Integrated Extended Functions") zur Verfügung. Das Ziel dieser Sicherheitsfunktionen ist, das Einhalten der betreffenden Grenze sicher zu überwachen und bei Verletzung einen Fehler zu melden bzw. den Antrieb dann in Folge sicher still zu setzen. Um ein Auslösen der Überwachungsfunktionen zu vermeiden, muss die Achse durch das Anwenderprogramm in den überwachten Betriebszustand geführt werden bzw. in diesem gehalten werden.

Das Zusammenwirken der Safety-Funktionen im Antrieb und der SIMATIC S7-1500 und S7-1500T ist erforderlich, um einen störungsfreien Anlagenbetrieb zu ermöglichen.

Die Technologieobjekte Drehzahl-, Positionier- und Gleichlaufachse unterstützen die "Safety Integrated Basic Functions" des Antriebs. Ein Auslösen der Basic Safety-Funktion wird vom Technologieobjekt erkannt und durch eine entsprechende Warnung (Technologie-Alarm 550 - Alarmreaktion: Sollwerte nachführen) bzw. Alarm (Technologie-Alarm 421 - Alarmreaktion: Freigabe wegnehmen) angezeigt.

Im Anwenderprogramm darf speziell für den Technologie-Alarm 550 keine zusätzliche Reaktion am Eingang "Enable" der Motion Control-Anweisung "MC_Power" im Sinne eines "Abschalten" erfolgen. Das anwenderseitige Abschalten der Motion Control-Anweisung "MC_Power" nach dem Technologie Alarm 421 ist möglich, aber nicht erforderlich.

Nach der Quittierung und Freigabe des Antriebs durch die eigentliche Sicherheitsfunktion kann am betreffenden Technologieobjekt durch die Motion Control-Anweisung "MC_Reset" der Technologie-Alarm ebenfalls quittiert werden. Danach ist das Technologieobjekt automatisch freigegeben, sofern "MC_Power.Enable" = "TRUE" geblieben ist.

Die "Safety Integrated Extended Functions" werden vom Technologieobjekt **nicht** eigenständig unterstützt.

Um ein Auslösen der erweiterten Sicherheitsfunktionen und damit eine Störung des Anlagenbetriebs zu vermeiden, werten Sie den Status der Sicherheitsfunktionen aus. Dies kann im Anwenderprogramm durch Verwendung bzw. die Auswertung von Zustandsinformationen des "Safety Info Channels" (SIC) erfolgen. Mithilfe der passenden Motion Control-Anweisung kann die Achse im überwachten Grenzbereich verbleiben bzw. diesen erreichen, bevor eine Abweichung erkannt wird.

Wenn eine Safety-Funktion an einer Folgeachse im aktiven Gleichlauf angewendet wird, ist eine der folgenden beiden Reaktionen erforderlich:

- Gleichlauf beenden
- Geschwindigkeit der Leitachse entsprechend anpassen

Im SIC stehen vier Zustandsworte zur Verfügung:

- S_ZSW1B
- S_ZSW2B
- S_ZSW3B
- S_V_LIMIT_B

Zur Übertragung stehen zwei vordefinierte PROFIdrive-Telegramme zur Verfügung:

- Tel. 700 (beinhaltet die Zustandsworte S_ZSW1B & S_V_LIMIT_B)
- Tel. 701 (beinhaltet alle vier Zustandsworte und zwei weitere Steuerworte)

Weiterführende Informationen

Weiterführende Informationen zu den Safety-Funktionen in SINAMICS Antrieben sowie zum SIC finden Sie im Funktionshandbuch SINAMICS S120 Safety Integrated.

https://support.industry.siemens.com/cs/ww/de/view/99668646 (https://support.industry.siemens.com/cs/ww/de/view/109754301)

Die folgenden Tabellen geben eine Übersicht der vier SIC-Zustandsworte und der jeweils erforderlichen Reaktion, um ein Stören des Anlagenbetriebs zu vermeiden.

S_ZSW1B

S_ZSW1B		Bedeutung		Empfohlene Reaktion der betroffenen Achse im Anwenderprogramm
Bit	Belegung			
0	STO (aktiv)	1	Safe Torque Off aktiv	"MC_Power" kann freigegeben bleiben (wartet).
		0	Nicht aktiv	Keine
1	SS1 (aktiv)	1	Safe Stop 1 aktiv	Antrieb bremst autark und geht in STO. "MC_Power" freigegeben lassen bis STO.
		0	Nicht aktiv	Keine
2	SS2 (aktiv)	1	Safe Stop 2 aktiv	Antrieb bremst autark und geht in SOS. "MC_Power" freigegeben lassen.
		0	Nicht aktiv	Keine
3	SOS (aktiv)	1	Safe Operating Stop aktiv	"MC_Power" freigegeben lassen. Der Antrieb darf sich nicht bewegen (Überwachung erfolgt im Antrieb)
		0	Nicht aktiv	Keine
4	SLS (aktiv)	1	Safety-Limited Speed aktiv	"MC_Power" freigegeben lassen. Geschwindigkeit muss kleiner als die aktive Geschwindigkeitsgrenze (siehe "Aktive SLS-Stufe" bzw. S_V_LIMIT_B) sein.
		0	Nicht aktiv	Keine
5	SOS (angewählt)	1	Safe Operating Stop angewählt	"MC_Power" freigegeben lassen. Innerhalb der durch SOS zugelassenen Zeit durch "MC_Halt" abbremsen.
		0	Abgewählt	Keine

S_ZSW1B		Bedeutung		Empfohlene Reaktion der betroffenen Achse im Anwenderprogramm
Bit	Belegung			
6	SLS (angewählt)	1	Safety-Limited Speed angewählt	Geschwindigkeitslimit innerhalb der durch SLS gegebenen Zeit unterschreiten. Z. B. durch die Vorgabe von Override oder einer neuen dynamischen Grenze (Einschränkung bei synchroner Bewegung).
		0	Abgewählt	Keine
7	Internes Ereignis	1	Sammelmeldung, dass eine Safety-Funktion angewählt bzw. aktiv wurde	Weitere Auswertung der Zustandsworte erforderlich, um die auslösende Safety-Funktion zu ermitteln. Das Bit zeigt an, dass eine Safety-Funktion aktiv ist. (Siehe auch "Safety Meldung")
		0	Kein Ereignis	Keine
8	Reserviert	-		-
9	Aktive SLS-Stufe	SLS Geschwindigkeitsgrenze Anzeige Bit 0		Zusatzinfo zum SLS (Bit 6) – Zeigt stufenweise (1… 4) die aktive Geschwindigkeitsgrenze für SLS an. Diese kann im Programm ausgewertet werden, um die aktuelle Geschwindigkeit der Achse entsprechend zu begrenzen.
10		SLS Geschwindigkeitsgrenze Anzeige Bit 1		
11	Reserviert	-		-
12	SDI positiv	1	Safe Direction positiv angewählt	Innerhalb der durch SDI vorgegebenen Zeit, den Stillstand oder die positive Geschwindigkeit des Istwerts der Achse erreichen (wenn SDI negativ = 0).
		0	Abgewählt	Keine Überwachung auf positive Richtung.
13	SDI negativ	1	Safe Direction negativ angewählt	Innerhalb der durch SDI vorgegebenen Zeit, den Stillstand oder die negative Geschwindigkeit des Istwerts der Achse erreichen (wenn SDI positiv = 0).
		0	Abgewählt	Keine Überwachung auf negative Richtung.
14	ESR Rückziehen	1	Erweitertes Stillsetzen und Rückziehen angefordert (ist keine Safety Funktion)	Individuell zu betrachten. Weiterführende Informationen finden Sie im Funktionshandbuch SINAMICS S120 Safety Integrated. https://support.industry.siemens.com/cs/ww/de/view/99668646 (https://support.industry.siemens.com/cs/ww/de/view/109754301)
		0	Nicht angefordert	Keine
15	Safety Meldung	1	Wirksam	Bei Bedarf das Bit als Sammelmeldung auswerten, ob eine Safety Meldung im Meldungspuffer vorliegt.
		0	Nicht wirksam	Keine

S_V_LIMIT_B

S_V_LIMIT_B		Bedeutung	Erläuterung
Bit	Belegung		
0 … 31	Sollwertgeschwindigkeitsbegrenzung	SLS-Speedlimit (32-Bit-Auflösung mit Vorzeichen)	Zusatzinfo zum SLS (S_ZSW1B Bit 6) Zeigt die angewählte/aktive Geschwindigkeitsgrenze für SLS an. Bei Bedarf die Geschwindigkeitsgrenze im Programm auswerten, um die aktuelle Achsgeschwindigkeit entsprechend zu begrenzen.

S_ZSW2B

S_ZSW2B		Bedeutung		Empfohlene Reaktion der betroffenen Achse
Bit	Belegung			
0 … 3	Reserviert	-		-
4	SLP angewählter Positionsbereich	1	SLP-Bereich 2 angewählt	Sichere Position liegt im Bereich 2. Position nicht mehr über das Anwenderprogramm verändern.
		0	SLP-Bereich 1 angewählt	Sichere Position liegt im Bereich 1. Position nicht mehr über das Anwenderprogramm verändern.
5, 6	Reserviert	-		-
7	SLP angewählt und Anwenderzustimmung gesetzt	1	Safety-Limited Position angewählt und Anwenderzustimmung ist gesetzt	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung (Bedeutet, dass das SLP angewählt ist und die sichere Position vom Anwender bestätigt wurde – vgl. "Sicheres Referenzieren")
		0	SLP nicht angewählt oder Anwenderzustimmung fehlt	Applikationsabhängige Auswertung
8	SDI positiv	1	Safe Direction positiv angewählt	Innerhalb der durch SDI vorgegebenen Zeit den Stillstand oder positive Geschwindigkeit des Achsistwerts erreichen. (Wenn SDI negativ = 0)
		0	Abgewählt	Keine Überwachung auf positive Richtung
9	SDI negativ	1	Safe Direction negativ angewählt	Innerhalb der durch SDI vorgegebenen Zeit den Stillstand oder negative Geschwindigkeit des Achsistwerts erreichen (Wenn SDI positiv = 0)
		0	Abgewählt	Keine Überwachung auf negative Richtung
10, 11	Reserviert	-		-
12	Teststopp aktiv	1	Teststopp aktiv	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Nicht aktiv	Keine
13	Teststopp erforderlich	1	Teststopp erforderlich	Teststopp durchführen
		0	Nicht erforderlich	Keine
14, 15	Reserviert	-		-

S_ZSW3B

S_ZSW3B		Bedeutung		Empfohlene Reaktion der betroffenen Achse
Bit	Belegung			
0	Bremsentest	1	Bremsentest angewählt	"MC_Power" freigegeben lassen. Keine Fahrbewegung durch Anwenderprogramm starten.
		0	Abgewählt	Keine – Bremsentest ist inaktiv (normaler Anlagenbetrieb)

S_ZSW3B		Bedeutung		Empfohlene Reaktion der betroffenen Achse
Bit	Belegung			
1	Sollwertvorgabe Antrieb/Extern	1	Vorgabe beim Antrieb	Der Drehzahlsollwert wird von der Funktion SBT vorgegeben. Applikationsabhängige Auswertung
		0	Vorgabe extern (Steuerung)	Der "normale" Drehzahlsollwert ist wirksam. Applikationsabhängige Auswertung, Sollwertvorgabe durch Anwenderprogramm erforderlich
2	Aktive Bremse	1	Test Bremse 2 aktiv	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Test Bremse 1 aktiv	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
3	Bremsentest aktiv	1	Test aktiv	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Inaktiv	Keine
4	Bremsentest Ergebnis	1	Test erfolgreich	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Fehlerhaft	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung, i. d. R. muss der Test erfolgreich sein, um die Sicherheit der Bremse zu gewährleisten.
5	Bremsentest beendet	1	Test durchlaufen	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Unvollständig	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung, i. d. R. Test wiederholen
6	Externe Bremse Anforderung	1	Bremse schließen	Externe Bremse schließen (wenn durch Anwenderprogramm gesteuert) Applikationsabhängige Auswertung
		0	Bremse öffnen	Externe Bremse öffnen (wenn durch Anwenderprogramm gesteuert) Applikationsabhängige Auswertung
7	Aktuelle Last Vorzeichen	1	Vorzeichen negativ	Status des Lastvorzeichens falls im Anwenderprogramm benötigt Applikationsabhängige Auswertung
		0	Vorzeichen positiv	
8 … 13	Reserviert	-		-
14	Abnahmetest SLP(SE) angewählt	1	Abnahmetest SLP(SE) angewählt	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Abgewählt	Keine
15	Abnahmetestmodus angewählt	1	Abnahmetestmodus angewählt	Statusmeldung (falls im Anwenderprogramm benötigt) Applikationsabhängige Auswertung
		0	Abgewählt	Keine

Achse in Simulation

S7-1500 Motion Control bietet die Möglichkeit, reale Achsen im Simulationsbetrieb zu verfahren. Damit lassen sich Drehzahl-, Positionier- und Gleichlaufachsen ohne verbundenen Antrieb und Geber in der CPU simulieren.

Bei aktiviertem Simulationsbetrieb braucht die Antriebs- und Geberanbindung in der Achskonfiguration noch nicht konfiguriert sein, z. B. wenn die Antriebsprojektierung zu diesem Zeitpunkt noch nicht verfügbar ist. Die Konfiguration "Simulation" lässt sich während der Laufzeit des Anwenderprogramms ändern (<TO>.Simulation.Mode). Beim Beenden der Simulation ist eine gültige Antriebs- und Geberanbindung erforderlich.

Um ein Technologieobjekt im Simulationsbetrieb oder mit SIMATIC S7-PLCSIM zu verwenden, müssen Sie Geber 1 für die Lageregelung der Achse verwenden.

Anwendungen:

- Eine Achse wird z. B. für die Programmierung der Maschinenapplikation simuliert und erst später, zur Inbetriebnahme, der konfigurierten Hardware zugeordnet.
- Bei der Inbetriebnahme sind z. B. noch nicht alle Hardware-Komponenten verfügbar.
- Bei der Inbetriebnahme sollen noch keine Achsbewegungen erfolgen.

Verhalten im Simulationsbetrieb

Eine Achse in Simulation gibt keine Sollwerte an den Antrieb aus und liest keine Istwerte des Gebers ein. Die Istwerte (Seite 1696) werden mit einem Zeitverzug aus den Sollwerten gebildet.

Hardware-Endschalter und Referenzpunktschalter haben keine Wirkung.

Die Technologieobjekte Messtaster (bei Signalerfassung über TM Timer DIDQ oder SINAMICS-Messtastereingang), Nocken und Nockenspur lassen sich auch an Achsen in Simulation verwenden.

Die folgende Tabelle zeigt die Motion Control-Anweisungen mit angepasstem Verhalten im Simulationsbetrieb:

Motion Control-Anweisung	Verhalten im Simulationsbetrieb
MC_Power	Die Achse wird unmittelbar freigegeben, ohne auf Rückmeldung vom Antrieb zu warten.
MC_Home	Referenzieraufträge werden unmittelbar ohne simulierte Achsbewegung ausgeführt.
MC_TorqueLimiting	Das vorgegebene Drehmoment wird nicht an den Antrieb ausgegeben.

Virtuelle Achse

S7-1500 Motion Control bietet die Möglichkeit, eine Achse als virtuelle Achse zu konfigurieren. Eine virtuelle Achse besitzt die Bewegungsführung, aber im Gegensatz zu realen Achsen keine Antriebs- und Geberanbindung. Die Sollwerte werden nur innerhalb der Steuerung verarbeitet und nie ein realer Antrieb angesteuert.

Anwendung:

Eine virtuelle Achse wird z. B. häufig als virtuelle Leitachse eingesetzt, um im Gleichlauf die Sollwerte für mehrere reale Folgeachsen zu erzeugen.

Die Konfiguration "Virtuelle Achse" ist nur über ein erneutes Laden in die CPU, im Betriebszustand STOP, änderbar (<TO>.VirtualAxis.Mode).

Das Verhalten einer virtuellen Achse ist identisch mit dem Verhalten einer Achse in Simulation (Seite 1706).

Datenanbindung Antrieb/Geber über Datenbaustein

Die Datenanbindung von PROFIdrive-Antrieben und Gebern erfolgt entweder direkt über das PROFIdrive-Telegramm oder über einen Datenbaustein.

Nutzen Sie die vom System generierten Tags der PROFIdrive-Telegramme, wenn Sie die Telegramminhalte auswerten wollen.

Nutzen Sie die Anbindung über Datenbaustein, wenn Sie im Anwenderprogramm prozessbedingt Telegramminhalte beeinflussen oder auswerten wollen.

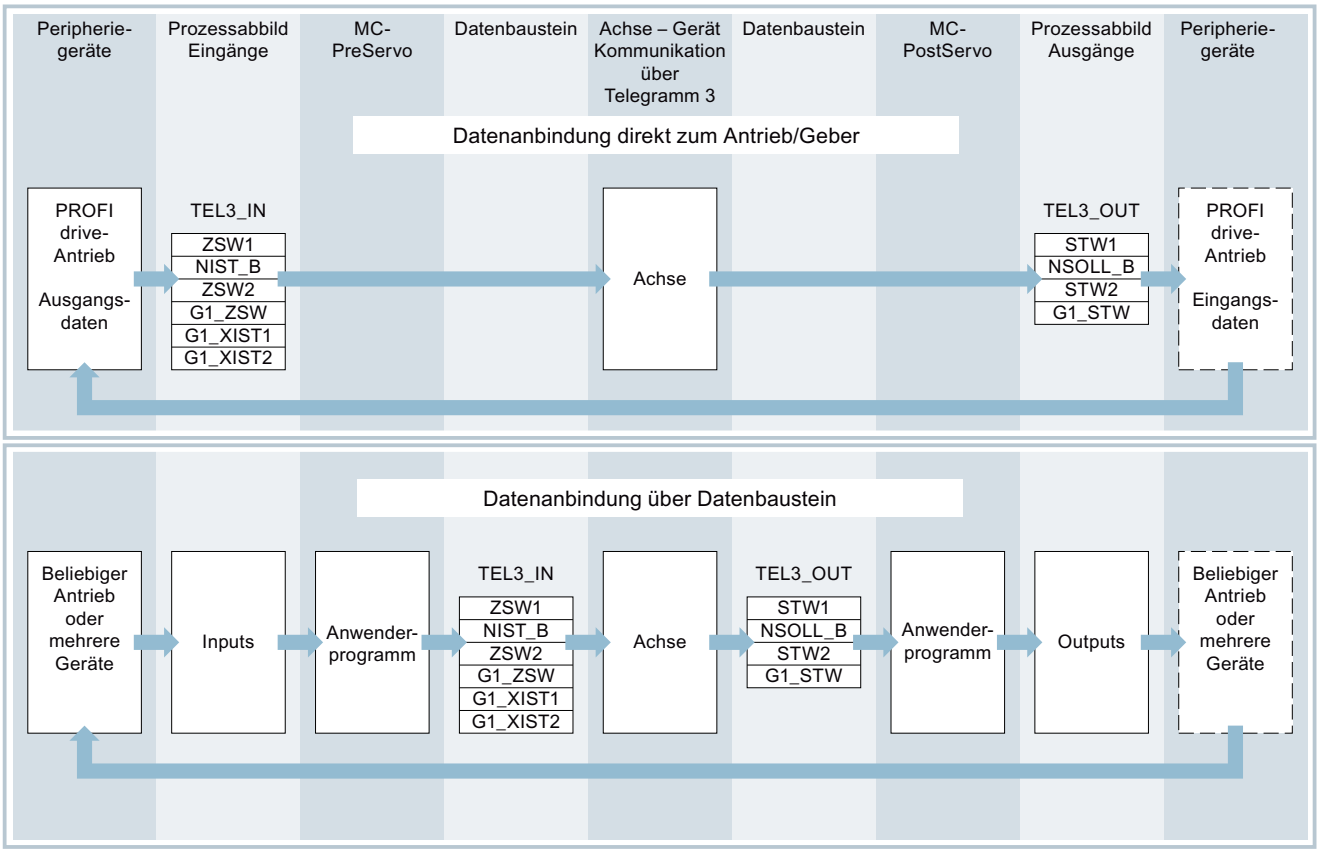

Prinzip der Datenanbindung über Datenbaustein

Grundsätzlich wird zu Beginn der Lageregelung der Achse (durch den MC-Servo [OB91]) der Eingangsbereich des Antriebs- bzw. Gebertelegramms gelesen.

Am Ende der Lageregelung wird der Ausgangsbereich des Antriebs- bzw. Gebertelegramms geschrieben.

Um prozessbedingt Telegramminhalte beeinflussen oder auswerten zu können, wird vor und nach der Lageregelung jeweils eine Datenschnittstelle über Datenbaustein zwischengeschaltet.

- Der Eingangsbereich des Telegramms kann über den Organisationsbaustein MC-PreServo [OB67] bearbeitet werden. Der MC-PreServo wird vor dem MC-Servo aufgerufen.
- Der Ausgangsbereich des Telegramms kann über den Organisationsbaustein MC-PostServo [OB95] bearbeitet werden. Der MC-PostServo wird nach dem MC-Servo aufgerufen.

Der Datenbaustein für die Datenanbindung muss anwenderseitig erstellt werden und eine Datenstruktur vom Datentyp "PD_TELx" beinhalten. "x" steht für die in der Gerätekonfiguration konfigurierte Telegrammnummer des Antriebs, bzw. Gebers.

Die Organisationsbausteine MC-PreServo und MC-PostServo sind anwenderseitig programmierbar und müssen über den Befehl "Neuen Baustein hinzufügen" hinzugefügt werden. Die Anbindung an die Peripherie über Telegramm muss in diesen Organisationsbausteinen programmiert werden. Bei der Verwendung von DSC müssen Sie selbst die Lebenszeichen im Telegramm in MC-Pre- und MC-PostServo, entsprechend der PROFIdrive-Norm, bearbeiten.

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Organisationsbausteine für Motion Control (Seite 1755)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Variablen

Folgende Variablen des Technologieobjekts sind für die Antriebs- und Geberanbindung relevant:

Antriebstelegramm	
Variable	Beschreibung
<TO>.Actor.Interface.AddressIn	Eingangsadresse für das PROFIdrive-Telegramm
<TO>.Actor.Interface.AddressOut	Ausgangsadresse für das PROFIdrive-Telegramm oder den Analogsollwert
<TO>.Actor.DriveParameter.ReferenceSpeed	Bezugswert (100 %) für die Solldrehzahl des Antriebs (NSOLL)
<TO>.Actor.DriveParameter.MaxSpeed	Maximalwert für die Solldrehzahl des Antriebs (NSOLL)
<TO>.Actor.DriveParameter.ReferenceTorque	Bezugsdrehmoment für das als Prozentwert übertragene Drehmoment

Gebertelegramm	
Variable	Beschreibung
<TO>.Sensor[n].Interface.AddressIn	Eingangsadresse für das PROFIdrive-Telegramm
<TO>.Sensor[n].Interface.AddressOut	Ausgangsadresse für das PROFIdrive-Telegramm

Gebertelegramm	
Variable	**Beschreibung**
<TO>.Sensor[n].System	Gebersystem linear oder rotatorisch
<TO>.Sensor[n].Type	Gebertyp inkrementell, absolut oder zyklisch absolut
<TO>.Sensor[n].Parameter.Resolution	Auflösung für lineare Geber Die Gitterteilung entspricht dem Abstand zwischen zwei Strichen.
<TO>.Sensor[n].Parameter.StepsPerRevolution	Inkremente pro Umdrehung für rotatorische Geber
<TO>.Sensor[n].Parameter.DeterminableRevolutions	Anzahl unterscheidbarer Geberumdrehungen bei einem Multiturn-Absolutwertgeber

Feinauflösung	
Variable	**Beschreibung**
<TO>.Sensor[n].Parameter.FineResolutionXist1	Anzahl Bits für die Feinauflösung XIST1 (zyklischer Geberistwert)
<TO>.Sensor[n].Parameter.FineResolutionXist2	Anzahl Bits für die Feinauflösung XIST2 (Absolutwert des Gebers)

Simulationsbetrieb		
Variable	**Beschreibung**	
<TO>.Simulation.Mode	Simulationsbetrieb	
	0	Keine Simulation, normaler Betrieb
	1	Simulationsbetrieb

Mechanik

Für die Anzeige und Verarbeitung der Position des Technologieobjekts ist entscheidend, ob die Position eine Längeneinheit (lineare Achse) oder eine Winkelgröße (rotatorische Achse) darstellt.

Beispiele für Längeneinheiten: mm, m, km

Beispiele für Winkelgrößen: °, rad

Für die Ermittlung der physikalischen Position aus einem Geberistwert müssen dem System die unterschiedlichen Eigenschaften und Anordnungen der Mechanik bekannt sein.

Positionierachse/Gleichlaufachse

Folgende Einstellmöglichkeiten zur Mechanik werden unterstützt:

- Lastgetriebe
- Spindelsteigung (nur lineare Achsen)

- Geberanbauart:
  - Motorseitig (vor dem Lastgetriebe)
  - Lastseitig (nach dem Lastgetriebe und ggf. Spindel)
  - Extern (z. B. Messrad)
- Invertierung der Antriebsrichtung
- Invertierung der Geberrichtung

Externer Geber

Folgende Einstellmöglichkeiten zur Mechanik werden unterstützt:

- Messgetriebe (bei rotatorischen Gebern)
- Spindelsteigung (nur bei linearem Einheitensystem und rotatorischen Gebern)
- Invertierung der Geberrichtung

Drehzahlachse

Folgende Einstellmöglichkeiten zur Mechanik werden unterstützt:

- Lastgetriebe
- Invertierung der Antriebsrichtung

Variablen

Folgende Variablen des Technologieobjekts sind für die Einstellung der Mechanik relevant:

Bewegungstyp		
Variable	**Beschreibung**	
<TO>.Properties.MotionType	Anzeige lineare oder rotatorische Bewegung	
	0	Lineare Bewegung
	1	Rotatorische Bewegung

Lastgetriebe	
Variable	**Beschreibung**
<TO>.LoadGear.Numerator	Lastgetriebe Zähler
<TO>.LoadGear.Denominator	Lastgetriebe Nenner

Spindelsteigung	
Variable	**Beschreibung**
<TO>.Mechanics.LeadScrew	Spindelsteigung

Geberanbauart	
Variable	**Beschreibung**
<TO>.Sensor[n].MountingMode	Geberanbauart
<TO>.Sensor[n].Parameter.Distance-PerRevolution	Weg der Last pro Geberumdrehung bei extern montierten Gebern

Invertierung	
Variable	**Beschreibung**
<TO>.Actor.InverseDirection	Invertierung Sollwert
<TO>.Actor.Efficiency	Wirkungsgrad der Spindelsteigung
<TO>.Sensor[n].InverseDirection	Invertierung Istwert

Modulo	
Variable	**Beschreibung**
<TO>.Modulo.Enable	Modulo aktivieren
<TO>.Modulo.Length	Modulolänge
<TO>.Modulo.StartValue	Modulostartwert

Verfahrbereichsbegrenzung

Hardware- und Software-Endschalter begrenzen den zulässigen Verfahr- und Arbeitsbereich der Positionierachse/Gleichlaufachse. Sie müssen vor der Verwendung in der Konfiguration bzw. im Anwenderprogramm aktiviert werden.

Der Zusammenhang zwischen Arbeitsbereich, maximalem Verfahrbereich und den Endschaltern ist im folgenden Bild dargestellt:

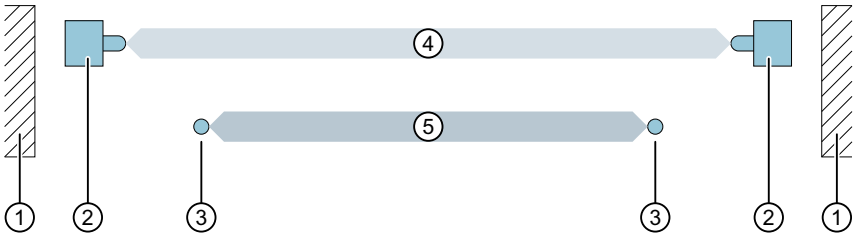

① Mechanischer Anschlag
② Hardware-Endschalter
③ Software-Endschalter
④ Maximaler Verfahrbereich
⑤ Arbeitsbereich

Hardware-Endschalter

Hardware-Endschalter sind Endlagenschalter, die den maximal zulässigen Verfahrbereich der Achse begrenzen.

Wählen Sie die Positionen der Hardware-Endschalter so, dass im Bedarfsfall genügend Bremsweg für die Achse vorhanden ist. Die Achse sollte vor einem mechanischen Anschlag zum Stillstand kommen.

Anfahren der Hardware-Endschalter

Bei der Überwachung der Bereichsbegrenzung wird nicht unterschieden, ob die Schalter angefahren oder überfahren werden.

Beim Anfahren eines Hardware-Endschalters wird der Technologiealarm 531 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen).

Ausnahme

Werden die Hardware-Endschalter beim Referenzieren als Umkehrnocken genutzt, dann ist die Überwachung der Hardware-Endschalter unwirksam.

Beim Einsatz als Umkehrnocken wird die Achse mit der in der Dynamik-Voreinstellung projektierten Verzögerung gebremst.

Bei der Planung des Abstands, Hardware-Endschalter zum mechanischen Anschlag, ist dies zu berücksichtigen.

Freifahren

Die Position der Achse beim Erkennen des Hardware-Endschalters wird intern in der CPU gespeichert. Erst wenn der Hardware-Endschalter verlassen wurde und sich die Achse wieder im maximalen Verfahrbereich befindet, wird der Status des angefahrenen Hardware-Endschalters zurückgesetzt.

Um nach dem Anfahren des Hardware-Endschalters die Achse freizufahren und den Status des Hardware-Endschalters zurückzusetzen, gehen Sie folgendermaßen vor:

1. Um Bewegungen in Freifahrtrichtung zu ermöglichen, quittieren Sie den Technologie-Alarm.
2. Fahren Sie die Achse in Freifahrtrichtung, bis der Hardware-Endschalter verlassen ist. Die Achse muss sich danach im maximalen Verfahrbereich befinden.
   Wenn Sie vor dem Verlassen des Hardware-Endschalters entgegen der Freifahrtrichtung fahren, wird die Überwachung erneut ausgelöst.

Folgendes Diagramm zeigt das Verhalten des Statusworts beim Anfahren des Hardware-Endschalters und beim Freifahren der Achse:

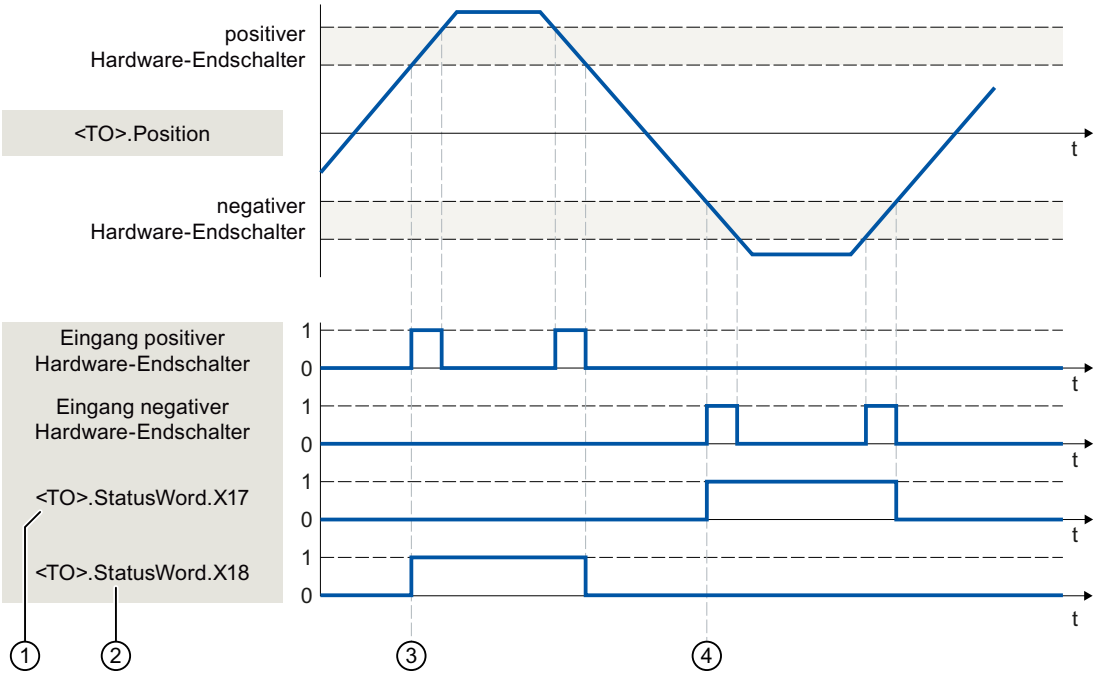

① <TO>.StatusWord.X17 (HWLimitMinActive)

0 Negativer Hardware-Endschalter nicht angefahren

1 Negativer Hardware-Endschalter angefahren oder überfahren

② <TO>.StatusWord.X18 (HWLimitMaxActive)

0 Positiver Hardware-Endschalter nicht angefahren

1 Positiver Hardware-Endschalter angefahren oder überfahren

③ Die Position der Achse wird beim Erkennen des **positiven** Hardware-Endschalters intern in der CPU gespeichert. Zum Rücksetzen des Status des Hardware-Endschalters muss diese Position unterschritten werden.

④ Die Position der Achse wird beim Erkennen des **negativen** Hardware-Endschalters intern in der CPU gespeichert. Zum Rücksetzen des Status des Hardware-Endschalters muss diese Position überschritten werden.

Software-Endschalter

Mit Software-Endschaltern wird der Arbeitsbereich der Achse begrenzt. Positionieren Sie die Software-Endschalter, bezogen auf den Verfahrbereich, immer innerhalb der Hardware-Endschalter. Da die Positionen der Software-Endschalter flexibel eingestellt werden können, kann der Arbeitsbereich der Achse je nach aktuellem Geschwindigkeitsprofil individuell angepasst werden.

Software-Endschalter sind erst bei gültigem Istwert nach dem Referenzieren des Technologieobjekts wirksam. Die Überwachung der Software-Endschalter wird auf den Sollwert bezogen.

Aktivierte Modulofunktion

Bei aktivierter Modulofunktion wird die Moduloposition überwacht.

Die Software-Endschalter werden in der Konfiguration der Achse konfiguriert und aktiviert. Im Anwenderprogramm können die Software-Endschalter über die Variable "<TO>.PositionLimits_SW.Active" aktiviert oder deaktiviert werden. Wenn die Positionen beider Software-Endschalter außerhalb des Modulobereichs liegen, ist die Überwachung nicht wirksam. Ob die Positionen der Software-Endschalter innerhalb des Modulobereichs liegen, wird nicht überprüft.

Anfahren der Software-Endschalter

Die Achse prüft während der Bewegung ständig die Position des Software-Endschalters und bremst ggf. genau auf diese Position ab.

Beim Fahren auf den Software-Endschalter wird der Technologiealarm 533 ausgegeben und die Achse mit den maximalen Dynamikwerten angehalten (Alarmreaktion: Stopp mit maximalen Dynamikwerten). Das Technologieobjekt bleibt freigegeben.

Überfahren der Software-Endschalter

Beim Überfahren des Software-Endschalters wird der Technologiealarm 534 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen).

Freifahren

Gehen Sie zum Freifahren der Achse nach Verletzung des Software-Endschalters folgendermaßen vor:

1. Quittieren Sie den Technologiealarm.
2. Fahren Sie die Achse in Freifahrtrichtung, bis der Software-Endschalter verlassen ist. Wenn Sie vor dem Verlassen des Software-Endschalters entgegen der Freifahrtrichtung fahren, wird die Überwachung erneut ausgelöst.

Variablen

Folgende Variablen des Technologieobjekts sind für Software-Endschalter relevant:

Statusanzeigen	
Variable	**Beschreibung**
<TO>.StatusWord.X15 (SWLimitMinActive)	Negativer Software-Endschalter ist aktiv.
<TO>.StatusWord.X16 (SWLimitMaxActive)	Positiver Software-Endschalter ist aktiv.
<TO>.ErrorWord.X8 (SWLimit)	Ein Alarm steht an, dass ein Software-Endschalter verletzt wurde.

Steuerbits	
Variable	**Beschreibung**
<TO>.PositionLimits_SW.Active	Aktiviert/deaktiviert die Überwachung der Software-Endschalter.

Positionswerte	
Variable	**Beschreibung**
<TO>.PositionLimits_SW.MinPosition	Position des negativen Software-Endschalters
<TO>.PositionLimits_SW.MaxPosition	Position des positiven Software-Endschalters

Folgende Variablen des Technologieobjekts sind für Hardware-Endschalter relevant:

Statusanzeigen	
Variable	**Beschreibung**
<TO>.StatusWord.X17 (HWLimitMinActive)	Negativer Hardware-Endschalter ist aktiv.
<TO>.StatusWord.X18 (HWLimitMaxActive)	Positiver Hardware-Endschalter ist aktiv.
<TO>.ErrorWord.X9 (HWLimit)	Ein Alarm steht an, dass ein Hardware-Endschalter verletzt wurde.

Steuerbits	
Variable	**Beschreibung**
<TO>.PositionLimits_HW.Active	Aktiviert/deaktiviert die Überwachung der Hardware-Endschalter.

Parameter		
Variable	**Beschreibung**	
<TO>.PositionLimits_HW.MinSwitchLevel	Pegelauswahl zur Aktivierung des unteren Hardware-Endschalters	
	FALSE	Bei Low-Pegel ist das Signal aktiv.
	TRUE	Bei High-Pegel ist das Signal aktiv.
<TO>.PositionLimits_HW.MinSwitchAddress	Bytenummer der Peripherieadresse des Hardware-Endschalters für die untere bzw. minimale Position	
<TO>.PositionLimits_HW.MaxSwitchLevel	Pegelauswahl zur Aktivierung des oberen Hardware-Endschalters	
	FALSE	Bei Low-Pegel ist das Signal aktiv.
	TRUE	Bei High-Pegel ist das Signal aktiv.
<TO>.PositionLimits_HW.MaxSwitchAddress	Bytenummer der Peripherieadresse des Hardware-Endschalters für die obere bzw. maximale Position	

Bewegungsführung und Dynamikgrenzen

Die Bewegungsführung der Achse erfolgt über Geschwindigkeitsprofile (Seite 1717). Die Geschwindigkeitsprofile werden entsprechend den Dynamikvorgaben berechnet. Ein Geschwindigkeitsprofil definiert das Verhalten der Achse beim Anfahren, Bremsen und bei Geschwindigkeitsänderungen. Beim Positionieren wird ein Geschwindigkeitsprofil berechnet, das die Achse auf den Zielpunkt verfährt.

Aus den Eigenschaften des Antriebs und der Mechanik ergeben sich Maximalwerte für Geschwindigkeit, Beschleunigung und Ruck. Diese Maximalwerte können in den

Dynamikgrenzen eingestellt werden. Die Dynamikgrenzen sind bei jeder über das Technologieobjekt erzeugten Bewegung als Grenzen wirksam. Bei einer Folgeachse im Gleichlauf sind die Dynamikgrenzen nicht wirksam.

Die einstellbare Notstopp-Verzögerung (Seite 1719) wird durch die Motion Control-Anweisung "MC_Power" oder einen Technologiealarm ausgelöst.

Die Ruckbegrenzung reduziert die Belastung der Mechanik während einer Beschleunigungs- oder Verzögerungsrampe. Ein "verrundetes" Geschwindigkeitsprofil ergibt sich.

Siehe auch

Dynamikgrenzen im Gleichlauf (Seite 1720)

Geschwindigkeitsprofil

Für die Bewegungsführung der Achse werden Geschwindigkeitsprofile mit oder ohne Ruckbegrenzung unterstützt.

Die Dynamikwerte für die Bewegung werden am Bewegungsauftrag vorgegeben. Alternativ können die Werte der Dynamik-Voreinstellung (Seite 1874) genutzt werden. Die Voreinstellungen und die Grenzen für Geschwindigkeit, Beschleunigung, Verzögerung und Ruck werden in der Konfiguration eingestellt.

Für die Beeinflussung der Geschwindigkeit kann der aktuellen Verfahrgeschwindigkeit ein Geschwindigkeits-Override überlagert werden.

Geschwindigkeitsprofil ohne Ruckbegrenzung

Das folgende Bild zeigt Geschwindigkeit, Beschleunigung und Ruck:

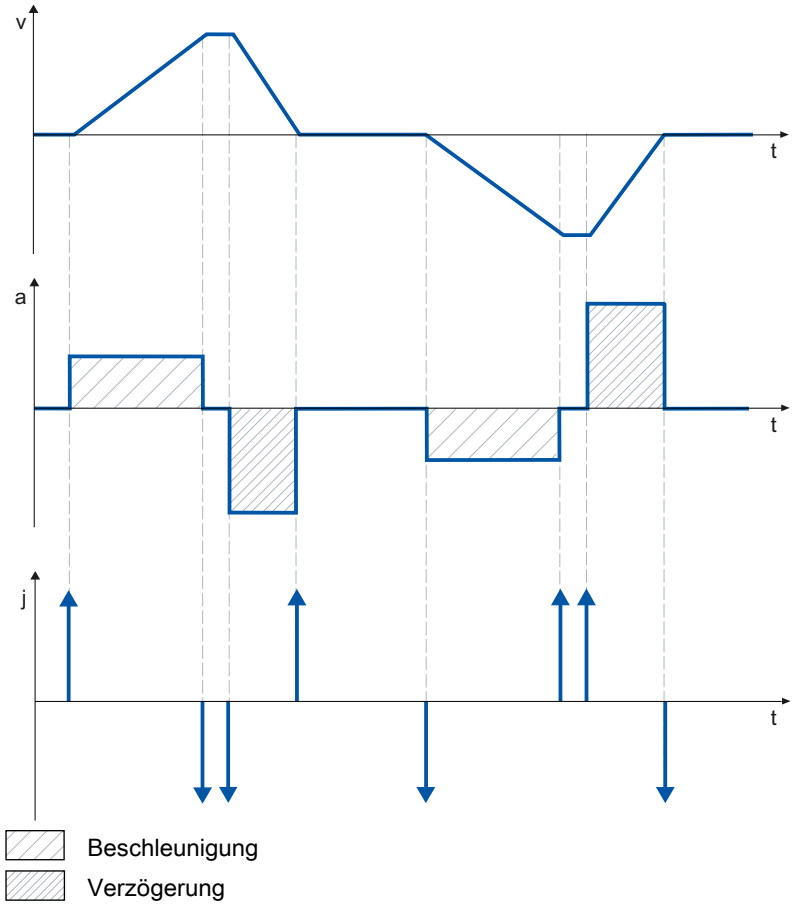

Geschwindigkeitsprofil mit Ruckbegrenzung

Das folgende Bild zeigt Geschwindigkeit, Beschleunigung und Ruck:

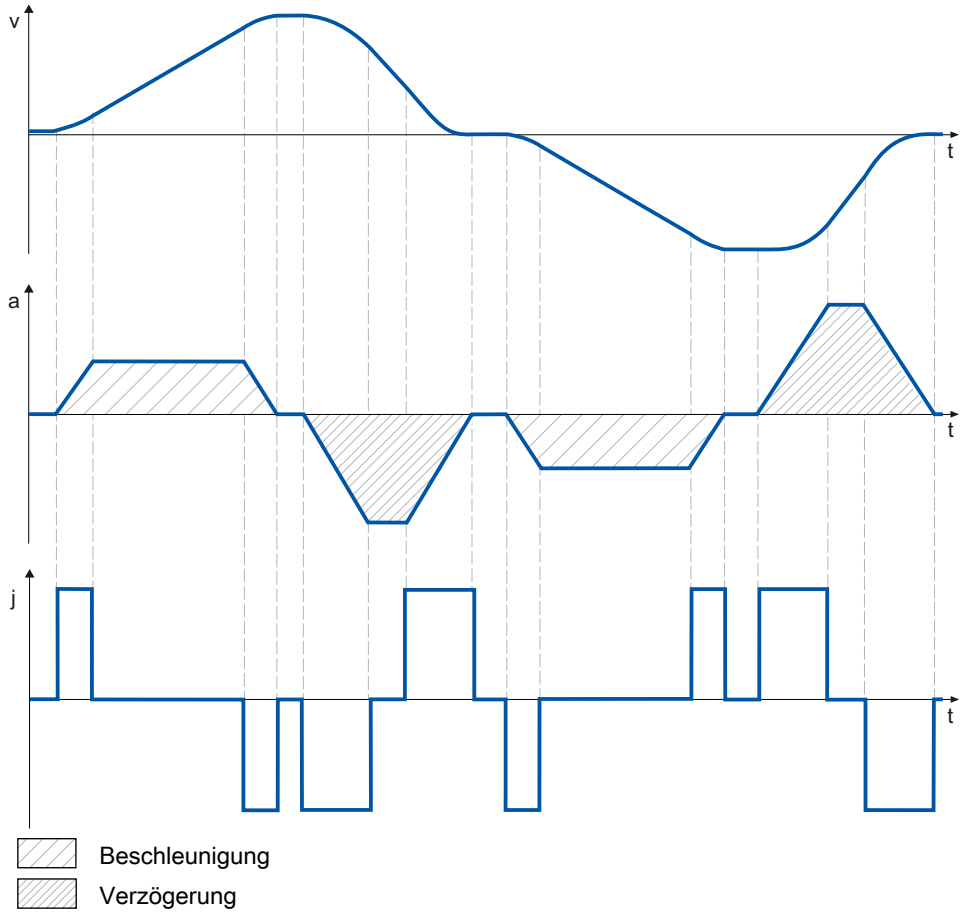

Ein Geschwindigkeitsprofil mit Ruckbegrenzung wird für einen stetigen Beschleunigungs- und Verzögerungsverlauf eingesetzt. Der Ruck ist vorgebbar.

Notstopp-Verzögerung

Bei einem Stopp mit der Notstopp-Rampe wird die Achse mit der eingestellten Notstopp-Verzögerung ohne Ruckbegrenzung bis zum Stillstand abgebremst.

In folgenden Fällen wird die eingestellte Notstopp-Verzögerung wirksam:

- Bei einer Notstopp-Rampe, die über die Motion Control-Anweisung "MC_Power" mit Parameter "StopMode" = 0 aktiviert wurde.
- Bei einem Technologie-Alarm mit der lokalen Alarmreaktion "Stopp mit Notstopp-Rampe".

Diese Notstopp-Verzögerung kann größer als die maximale Verzögerung eingestellt werden. Wenn die Notstopp-Verzögerung kleiner eingestellt wird, kann es im Fall "Halten auf Software-Endschalter" und beim Auftreten eines Technologie-Alarms mit der lokalen Alarmreaktion "Stopp mit Notstopp-Rampe" dazu kommen, dass die Achse erst nach dem Endschalter anhält.

Dynamikgrenzen im Gleichlauf

Dynamikgrenzen im Gleichlauf mit "MC_GearIn"

Leitachse

An der Leitachse sind immer die am Technologieobjekt konfigurierten Dynamikgrenzen wirksam.

Folgeachse

Wenn eine Gleichlaufachse als Folgeachse im Getriebegleichlauf mit "MC_GearIn" betrieben wird, gelten abhängig von der Phase des Gleichlaufs folgende Dynamikgrenzen:

- **Aufsynchronisieren**
  Beim Aufsynchronisieren gelten für die Folgeachse die am Technologieobjekt konfigurierten Dynamikgrenzen.
- **Synchron fahren**
  Wenn die Gleichlaufachse als Folgeachse synchron zur Leitachse fährt, wird die Dynamik der Folgeachse nur auf die maximale Drehzahl des Antriebs begrenzt (<TO>.Actor.DriveParameter.MaxSpeed). Die Dynamik der Folgeachse ergibt sich aus der Gleichlauffunktion.
  Wenn die an der Folgeachse konfigurierten Dynamikgrenzen überschritten werden, wird dies in der Variable "<TO.>StatusSynchronizedMotion.StatusWord" des Technologieobjekts angezeigt.
  Wenn die Folgeachse dem Leitwert nicht folgen kann, ergibt sich ein Schleppfehler, der über die Schleppfehlerüberwachung überwacht wird.
- **Gleichlauf ablösen**
  Sobald der Gleichlauf abgelöst wird, gelten für die Folgeachse wieder die am Technologieobjekt konfigurierten Dynamikgrenzen. Mit dem Start des ablösenden Auftrags wird die aktive Dynamik überführt (verschliffen) auf die konfigurierten Dynamikgrenzen und die Vorgaben an der Motion Control-Anweisung.

Siehe auch

Variable "StatusSynchronizedMotion" (Gleichlaufachse) (Seite 2129)

Dynamikgrenzen im Gleichlauf mit "MC_GearInPos"/"MC_CamIn"

Leitachse

An der Leitachse sind immer die am Technologieobjekt konfigurierten Dynamikgrenzen wirksam.

Folgeachse

Wenn eine Gleichlaufachse als Folgeachse im Gleichlauf mit der Motion Control-Anweisung "MC_GearInPos" oder "MC_CamIn" betrieben wird, gelten abhängig von der Phase des Gleichlaufs folgende Dynamikgrenzen:

- **Wartender Gleichlauf**
  Wenn noch kein Gleichlauf aktiv ist, gelten die konfigurierten Dynamikgrenzen. Wenn bereits ein Gleichlauf aktiv ist, sind diese Grenzen durch den vorherigen Gleichlauf aufgehoben.
- **Aufsynchronisieren/Synchron fahren**
  Beim Aufsynchronisieren/Synchron fahren wird die Dynamik der Folgeachse nur auf die maximale Drehzahl des Antriebs begrenzt (<TO>.Actor.DriveParameter.MaxSpeed). Die Dynamik der Folgeachse ergibt sich aus der Gleichlauffunktion.
  Wenn die an der Folgeachse konfigurierten Dynamikgrenzen überschritten werden, wird dies in der Variablen des Technologieobjekts "<TO.>StatusSynchronizedMotion.StatusWord" angezeigt.
  Wenn die Folgeachse dem Leitwert nicht folgen kann, ergibt sich ein Schleppfehler, der über die Schleppfehlerüberwachung überwacht wird.
- **Gleichlauf ablösen**
  Sobald der Gleichlauf abgelöst wird, gelten für die Folgeachse wieder die am Technologieobjekt konfigurierten Dynamikgrenzen. Mit dem Start des ablösenden Auftrags wird die aktive Dynamik überführt (verschliffen) auf die konfigurierten Dynamikgrenzen und die Vorgaben an der Motion Control-Anweisung.

Siehe auch

Variable "StatusSynchronizedMotion" (Gleichlaufachse) (Seite 2129)

Bewegungsvorgabe über "MotionIn"

Im Gegensatz zu den Motion Control-Anweisungen wie z. B. "MC_MoveAbsolute" und "MC_MoveRelative", wird bei der Verwendung von "MC_MotionInVelocity" und "MC_MotionInPosition" kein Bewegungsprofil vom System berechnet. Jeder einzelne Sollwert des Bewegungsprofils (Bewegungsvektor) muss mit der "MotionIn"-Anweisung im Applikationszyklus vorgegeben werden. Dadurch haben Sie die Möglichkeit, Ihr eigenes Bewegungsprofil zu berechnen.

Die Sollwerte werden typischerweise im Bearbeitungstakt des Technologieobjekts angepasst. Dazu wird die "MotionIn"-Anweisung im Organisationsbaustein MC-PreServo [OB67] oder MC-

PostServo [OB95] aufgerufen. Im Anschluss werden im MC-Interpolator [OB92] die Sollwerte direkt an die Achse übernommen.

> ⚠ WARNUNG
>
> **Unkontrollierte Achsbewegungen**
>
> Bei Verwendung der Bewegungsvorgabe über die Motion Control-Anweisungen "MC_MotionInVelocity" und "MC_MotionInPosition" kann die Achse unkontrollierte Bewegungen ausführen.
>
> Berücksichtigen Sie die aktuelle Dynamik der Achse bei der Vorgabe der neuen Bewegungsvektoren. Die Bewegungsvektoren müssen konsistent zueinander sein.
>
> Richten Sie vor dem Betrieb mit den Motion Control-Anweisungen "MC_MotionInVelocity" und "MC_MotionInPosition" folgende Schutzmaßnahmen ein:
>
> - Stellen Sie sicher, dass sich der NOT-AUS-Schalter in Reichweite des Bedieners befindet.
> - Aktivieren Sie die Hardware-Endschalter.
> - Aktivieren Sie die Software-Endschalter.
> - Stellen Sie sicher, dass die Schleppfehlerüberwachung aktiviert ist.
>
> Beachten Sie, dass eine mit der Achse gekoppelte Folgeachse ebenfalls verfahren wird.

Ablösen durch "MotionIn"-Anweisungen

Wenn eine Motion Control-Anweisung durch eine "MotionIn"-Anweisung abgelöst wird, werden die vorgegebenen Sollwerte sofort mit dem laufenden Applikationszyklus wirksam. Die Dynamik ergibt sich ausschließlich durch die Sollwertvorgaben des Anwenderprogramms. Sie wird weder begrenzt noch findet ein sanfter Übergang aus dem aktuellen Bewegungszustand statt. Berücksichtigen Sie die aktuelle Dynamik der Achse bei der Vorgabe der neuen Bewegungsvektoren. Beachten Sie dabei, dass am Technologieobjekt eingestellte Dynamikgrenzen nicht wirksam sind. Nur antriebsseitig eingestellte Grenzen sind wirksam.

Beenden der "MotionIn"-Anweisungen

Die "MotionIn"-Anweisungen können durch folgende Maßnahmen beendet werden:

- Ablösen durch eine andere Motion Control-Anweisung
  Die "MotionIn"-Anweisungen werden entsprechend dem im Kapitel "AUTOHOTSPOT" beschriebenen Verhalten abgelöst. In der Regel wird die aktuelle Dynamik auf die neue Bewegung überschliffen.
- Setzen des Parameters "Enable" auf "FALSE"
  Wenn Sie den Parameter "Enable" auf "FALSE" setzen, wird sofort der Sollwert null vorgegeben. Beachten Sie dabei, dass die am Technologieobjekt eingestellten Dynamikgrenzen nicht wirksam sind. Nur antriebsseitig eingestellte Grenzen sind wirksam.

"MC_MotionInVelocity"

Mit der Anweisung "MC_MotionInVelocity" können Sie die Geschwindigkeit und die Beschleunigung der Bewegung vorgeben. Die Anweisung ist für Drehzahl-, Positionier- und Gleichlaufachsen anwendbar.

Um die Anweisung ausführen zu können, müssen Sie mindestens die Geschwindigkeit vorgeben. Die Beschleunigung wird in der Regel nur für das Ablösen laufender Bewegungen benötigt. Standardmäßig beträgt der Wert der Beschleunigung null.

"MC_MotionInPosition"

Mit der Anweisung "MC_MotionInPosition" können Sie die Position, die Geschwindigkeit und die Beschleunigung der Bewegung vorgeben. Die Anweisung ist für Positionier- und Gleichlaufachsen anwendbar.

Um die Anweisung ausführen zu können, müssen Sie mindestens die Position und die Geschwindigkeit vorgeben. Die Beschleunigung wird in der Regel nur für das Ablösen laufender Bewegungen benötigt. Standardmäßig beträgt der Wert der Beschleunigung null. Die vorgegebenen Sollwerte müssen konsistent zueinander sein.

Die Positionsvorgabe wird lagegeregelt verarbeitet. Wenn Sie eine Geschwindigkeitsvorsteuerung verwenden, wird die Geschwindigkeitsvorgabe über die Geschwindigkeitsvorsteuerung verarbeitet.

Momentengrenzen

Kraft-/Momentenbegrenzung

Für die Technologieobjekte Drehzahlachse, Positionierachse und Gleichlaufachse ist eine einstellbare Kraft-/Momentenbegrenzung verfügbar. Die Kraft-/Momentenbegrenzung kann vor oder während eines Bewegungsauftrags aktiviert und deaktiviert werden. Voraussetzung zum Einsatz der Kraft-/Momentenbegrenzung ist, dass der Antrieb und das PROFIdrive-Telegramm die Momentenreduzierung unterstützen. Verwenden Sie z. B. ein Telegramm 10x.

Der Begrenzungswert kann als voreingestellter Wert in der Konfiguration der Achse konfiguriert werden oder im Anwenderprogramm über die Motion Control-Anweisung "MC_TorqueLimiting" definiert werden.

Der Anwender gibt die Begrenzungswerte in der konfigurierten Maßeinheit für Kraft oder Moment vor. Die Maßeinheiten werden im Konfigurationsfenster "Grundparameter" definiert.

Bezüglich Kraft-/Momentenbegrenzung stehen folgende Konfigurationsmöglichkeiten zur Verfügung:

- Achstyp "Linear"
  - Momentenbegrenzung wirksam an der Motorseite
  - Kraftbegrenzung wirksam an der Lastseite
- Achstyp "Rotatorisch"
  - Momentenbegrenzung wirksam an der Lastseite oder an der Motorseite

Intern wird die anwenderdefinierte Kraft-/Momentengrenze entsprechend der Festlegung in den PROFIdrive-Telegrammen 10x als prozentuale Momentenreduzierung an den Antrieb übertragen. Das im Konfigurationsdialog "Datenaustausch Antrieb" eingestellte Bezugsmoment muss mit dem am Antrieb eingestellten Bezugsmoment übereinstimmen.

Achstyp linear

Eine von Ihnen definierte lastseitige Kraftbegrenzung wird von der Technologie in eine Momentenreduzierung umgerechnet. Bei einer Begrenzung, die sich auf die Lastseite bezieht, werden die im Konfigurationsfenster "Mechanik" definierten Getriebe- und Spindelparameter berücksichtigt. Wenn der Wirkungsgrad von Getriebe und Spindel ausschlaggebend ist, können Sie diesen in der Variable "<TO>.Actor.Efficiency" einstellen.

Achstyp rotatorisch

Beim Achstyp rotatorisch wirkt lastseitig eine Momentenreduzierung. Die im Konfigurationsfenster "Mechanik" definierten Getriebeparameter werden berücksichtigt. Wenn der Wirkungsgrad des Getriebes ausschlaggebend ist, können Sie diesen in der Variable "<TO>.Actor.Efficiency" einstellen.

Die definierten Begrenzungswerte wirken als Betragswert und damit in gleicher Weise für positive wie auch negative Kräfte/Momente.

Positionier- und Schleppfehlerüberwachung bei aktiver Kraft-/Momentenbegrenzung

Infolge einer Kraft-/Momentenbegrenzung kann sich eine größere Soll-Ist-Differenz bei lagegeregelten Achsen aufbauen, was zu einem ungewollten Ansprechen der Positionier- und Schleppfehlerüberwachung führen kann.

Im Konfigurationsfenster "Momentenbegrenzung" ist daher die Positionier- und Schleppfehlerüberwachung der Achse bei wirksamer Kraft-/Momentenbegrenzung voreingestellt deaktiviert. Die Positionier- und Schleppfehlerüberwachung lässt sich bei Bedarf auch bei wirksamer Kraft-/Momentenbegrenzung aktiv halten.

Typisches Verhalten einer Positionier- oder Gleichlaufachse bei aktiver Kraft-/Momentenbegrenzung

Bei aktiver Kraft-/Momentenbegrenzung kann sich eine größere Soll-Ist-Differenz aufbauen, als beim Fahren ohne Kraft-/Momentenbegrenzung.

Die Achse versucht kontinuierlich, bei gleich bleibendem Sollwert den Schleppabstand abzubauen.

Bei Erhöhen der Begrenzungswerte oder Deaktivierung der Begrenzung während aktiver Lageregelung kann die Achse kurzzeitig beschleunigen, um den Schleppabstand abzubauen. Durch ein Schalten der Achse in den nicht lagegeregelten Betrieb, z. B. über "MC_MoveVelocity" mit "PositionControlled" = FALSE, ist der Schleppfehler nicht mehr wirksam.

Anhalten einer Achse bei aktiver Kraft-/Momentenbegrenzung

Beim Stoppen einer Achse im lagegeregelten Betrieb über "MC_Halt" wird auf die Sollposition und die Sollgeschwindigkeit aufgesetzt. Die Momentenbegrenzung bleibt weiterhin aktiv und ein gegebenenfalls aufgebauter Schleppabstand wird abgebaut. Die Achse befindet sich im Stillstand, wenn die Istgeschwindigkeit "0.0" erreicht und die minimale Verweildauer im Stillstandsfenster abgelaufen ist. Die Achse bleibt weiterhin freigegeben.

Beim Stoppen einer Achse über "MC_Power" und einer Notstopp-Rampe wird auf den Positionsistwert und die Istgeschwindigkeit aufgesetzt. Die Achse wird mit der konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht. Anschließend wird die Achse im Stillstand gesperrt.

In dem Moment, wo die Begrenzung deaktiviert wird, versucht der Antrieb den entstandenen Schleppfehler mit maximalen Dynamikwerten aufzuholen. Um dieses Verhalten zu vermeiden, verwenden Sie "MC_MoveVelocity" mit deaktivierter Lageregelung.

Siehe auch

Festanschlagserkennung (Seite 1725)

Konfiguration - Momentengrenzen (Seite 1857)

Konfiguration - Momentengrenzen (Seite 1880)

Festanschlagserkennung

Über die Motion Control-Anweisung "MC_TorqueLimiting" aktivieren und überwachen Sie eine Festanschlagserkennung. Zusammen mit einem lagegeregelten Bewegungsauftrag kann ein "Fahren auf Festanschlag" realisiert werden. Der Vorgang wird auch als Klemmen bezeichnet. Mit "Fahren auf Festanschlag" können z. B. Pinolen gegen das Werkstück mit einem vorgegebenen Moment gefahren werden.

Die Festanschlagserkennung wird im Konfigurationsfenster "Erweiterte Parameter > Begrenzungen > Festanschlagserkennung" konfiguriert.

Die Festanschlagserkennung ist nur im lagegeregelten Betrieb der Achse möglich. Wenn Antrieb und Telegramm die Kraft-/Momentenbegrenzung unterstützen, ist die Kraft-/Momentenbegrenzung beim Fahren auf den Festanschlag und beim Klemmen aktiv.

Erkennung des Festanschlags über Schleppabstand

Wenn der Antrieb während eines Bewegungsauftrags durch einen mechanischen Festanschlag gestoppt wird, vergrößert sich der Schleppabstand. Wenn der im Konfigurationsfenster "Erweiterte Parameter > Begrenzungen > Festanschlagserkennung" konfigurierte Schleppabstand überschritten wird, wird dies als Erreichen des Festanschlags gewertet.

Bei aktivierter Schleppfehlerüberwachung muss der konfigurierte Schleppfehler größer sein, als der Schleppabstand zur Erkennung des Festanschlags.

Klemmen am mechanischen Anschlag

Mit Erreichen des Festanschlags wird der laufende lagegeregelte Bewegungsauftrag mit "CommandAborted" abgebrochen. Der Sollwert wird nicht mehr verändert und der Schleppabstand bleibt somit konstant. Die Lageregelung bleibt weiter aktiv und die Überwachung der konfigurierten "Positioniertoleranz" wird aktiviert. Der Antrieb befindet sich im Zustand "Klemmen".

Wenn Antrieb und Telegramm die Kraft-/Momentenbegrenzung unterstützen, ist diese bei aktiver Festanschlagserkennung weiter aktiv. Während des Klemmens kann die Klemmkraft, bzw. das Klemmmoment verändert werden. Hierzu kann der Wert am Eingangsparameter "Limit" der Motion Control-Anweisung "MC_TorqueLimiting" verändert werden.

Überwachung der Klemmung

Wenn sich während der aktiven Klemmung die Istposition um einen Wert größer als die konfigurierte "Positioniertoleranz" ändert, wird dies als Wegbrechen oder Zurückdrücken des Festanschlags gewertet. Ein Alarm wird ausgelöst. Die Achse wird gesperrt und der Antrieb entsprechend der Antriebskonfiguration angehalten.

Wenn sich die Sollposition innerhalb der konfigurierten "Positioniertoleranz" befindet, kann das Wegbrechen oder Zurückdrücken des Festanschlags nicht erkannt werden.

Die konfigurierte Positionstoleranz muss kleiner als der konfigurierte Schleppabstand für die Erkennung der Klemmung sein.

Freifahren

Ein Freifahren vom Festanschlag ist nur mit einem lagegeregelten Bewegungsauftrag in die Gegenrichtung zum Festanschlag möglich.

Die Funktion "Fahren auf Festanschlag" bzw. das "Klemmen" ist beendet, wenn die "Positioniertoleranz" in Freifahrrichtung verlassen wird.

Siehe auch

Kraft-/Momentenbegrenzung (Seite 1723)

Konfiguration - Festanschlagserkennung (Seite 1881)

Additives Sollmoment

Die Motion Control-Anweisung "MC_TorqueAdditive" ermöglicht Ihnen ein zusätzliches Drehmoment im Antrieb aufzuschalten.

Das additive Sollmoment kommt z. B. zur Anwendung bei der Momentenvorsteuerung oder der Vorgabe des Zugmoments bei Wickelapplikationen.

Folgende Voraussetzungen sind zur Einstellung des additiven Sollmoments notwendig:

- SINAMICS-Antrieb (siehe Kompatibilitätsliste (Seite 2222))
- SIEMENS-Zusatztelegramm 750 zur Übertragung der Momentendaten an den Antrieb

Das zusätzliche Drehmoment kann sowohl positiv als auch negativ sein. Der in der Anweisung angegebene Wert ist ein technologischer Wert, kein Prozentwert. Die Maßeinheit für das Moment stellen Sie an der Achse ein (Standardwert: Nm).

Zulässiger Momentenbereich

Die Motion Control-Anweisung "MC_TorqueRange" ermöglicht Ihnen Momentengrenzen an den Antrieb vorzugeben.

Die Motion Control-Anweisung kommt z. B. zur Anwendung bei Wickelapplikationen, um ein Reißen des Materials zu verhindern.

Folgende Voraussetzungen sind zur Einstellung der Momentendaten notwendig:

- SINAMICS-Antrieb (siehe Kompatibilitätsliste (Seite 2222))
- SIEMENS-Zusatztelegramm 750 zur Übertragung der Momentendaten an den Antrieb

Der in der Anweisung angegebene Wert ist ein technologischer Wert, kein Prozentwert. Die Maßeinheit für das Moment stellen Sie an der Achse ein (Standardwert: Nm). Wenn Sie die Sollwerte am Technologieobjekt der Achse invertieren, werden auch die Werte für die obere und untere Momentengrenze invertiert und umgekehrt ausgegeben.

Wenn die Momentenbegrenzung über die Vorgabe der oberen und unteren Momentengrenze aktiviert wird, werden dadurch folgende Überwachungen und Begrenzungen deaktiviert:

- Schleppfehlerüberwachung
- Zeitbegrenzungen bei Positionierüberwachung
- Zeitbegrenzungen bei Stillstandsüberwachung

Wenn Sie unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Begrenzungen > Momentenbegrenzung" die Option "Positionsbezogene Überwachungen aktiv lassen" ausgewählt haben, bleiben die Überwachungen weiterhin wirksam.

Variablen

Folgende Variablen des Technologieobjekts sind für die Bewegungsführung relevant:

Status		
Variable	**Beschreibung**	
<TO>.StatusWord	Statusanzeige für eine aktive Bewegung	
<TO>.Position	Sollposition	
<TO>.Velocity	Sollgeschwindigkeit/Solldrehzahl	
<TO>.ActualPosition	Istposition	
<TO>.ActualVelocity	Istgeschwindigkeit	
<TO>.ActualSpeed	Istdrehzahl des Motors (nur bei Antriebstyp PROFIdrive)	
<TO>.Acceleration	Sollbeschleunigung	
<TO>.ActualAcceleration	Istbeschleunigung	
<TO>.StatusSynchronizedMotion.StatusWord.X0 (MaxVelocityExceeded)	Die Variable wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Geschwindigkeit überschritten wird.	
<TO>.StatusSynchronizedMotion.StatusWord.X1 (MaxAccelerationExceeded)	Die Variable wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Beschleunigung überschritten wird.	
<TO>.StatusSynchronizedMotion.StatusWord.X2 (MaxDecelerationExceeded)	Die Variable wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Verzögerung überschritten wird.	
<TO>.StatusMotionIn.FunctionState	Status der "MotionIn"-Funktion	
	0	Keine "MotionIn"-Funktion aktiv
	1	"MC_MotionInVelocity" aktiv
	2	"MC_MotionInPosition" aktiv

Überlagerung	
Variable	**Beschreibung**
<TO>.Override.Velocity	Geschwindigkeits- bzw. Drehzahl-Override

Dynamik-Grenzwerte	
Variable	**Beschreibung**
<TO>.DynamicLimits.MaxVelocity	Dynamikbegrenzung für maximale Geschwindigkeit (mechanisch)
<TO>.DynamicLimits.Velocity	Dynamikbegrenzung für maximale Geschwindigkeit (programmierbar)
<TO>.DynamicLimits.MaxAcceleration	Dynamikbegrenzung für maximale Beschleunigung
<TO>.DynamicLimits.MaxDeceleration	Dynamikbegrenzung für maximale Verzögerung
<TO>.DynamicLimits.MaxJerk	Dynamikbegrenzung für maximalen Ruck

Dynamik-Voreinstellungen	
Variable	**Beschreibung**
<TO>.DynamicDefaults.Velocity	Voreinstellung der Geschwindigkeit
<TO>.DynamicDefaults.Acceleration	Voreinstellung der Beschleunigung
<TO>.DynamicDefaults.Deceleration	Voreinstellung der Verzögerung
<TO>.DynamicDefaults.Jerk	Voreinstellung des Rucks
<TO>.DynamicDefaults.EmergencyDeceleration	Notstopp-Verzögerung

Momentenbegrenzung		
Variable	**Beschreibung**	
<TO>.TorqueLimiting.LimitDefaults.Torque	Begrenzungsdrehmoment	
<TO>.TorqueLimiting.LimitDefaults.Force	Begrenzungskraft	
<TO>.TorqueLimiting.LimitBase	Momentenbegrenzung motor- oder lastseitig	
	0	Motorseitig
	1	Lastseitig
<TO>.TorqueLimiting.PositionBasedMonitorings	Positionier- und Schleppfehlerüberwachung	
	0	Deaktiviert
	1	Aktiviert
<TO>.StatusTorqueData.CommandAdditiveTorqueActive	Funktion additives Sollmoment	
	0	Deaktiviert
	1	Aktiviert
<TO>.StatusTorqueData.CommandTorqueRangeActive	Funktion Momentengrenzen	
	0	Deaktiviert
	1	Aktiviert
<TO>.StatusTorqueData.ActualTorque	Istdrehmoment der Achse	

Festanschlagserkennung	
Variable	**Beschreibung**
<TO>.Clamping.FollowingErrorDeviation	Wert des Schleppfehlers, ab dem der Festanschlag erkannt wird
<TO>.Clamping.PositionTolerance	Positionstoleranz für die Klemmüberwachung

Referenzieren

Mit dem Referenzieren stellen Sie den Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung her. Der Positionsistwert am Technologieobjekt wird dabei einer Referenzmarke zugeordnet. Diese Referenzmarke repräsentiert eine bekannte mechanische Position.

Bei inkrementellen Istwerten wird dieser Vorgang als Referenzieren bezeichnet, bei absoluten Istwerten als Absolutwertgeberjustage.

Das Referenzieren ist Voraussetzung für die Anzeige der korrekten Position am Technologieobjekt und für das absolute Positionieren.

Das Referenzieren wird mit der Motion Control-Anweisung "MC_Home" aktiviert.

Referenzierstatus

Die Variable "<TO>.StatusWord.X5 (HomingDone)" des Technologieobjekts zeigt an, ob das Technologieobjekt Achse bzw. Externer Geber referenziert ist.

Referenzierart

Referenzieren kann über eine eigenständige Bewegung zum Referenzieren (Aktives Referenzieren), über das Erfassen einer Referenzmarke während einer anwenderseitig initiierten Bewegung (Passives Referenzieren) oder über direkte Positionszuordnung erfolgen.

Folgende Referenzierarten werden unterschieden:

- **Aktives Referenzieren**
  Das aktive Referenzieren initiiert eine Referenzierbewegung und führt die notwendige Fahrt auf die Referenzmarke aus. Beim Erkennen der Referenzmarke wird die Istposition auf den am "MC_Home" angegebenen Wert gesetzt. Die Angabe einer Referenzpunktverschiebung ist möglich. Die Referenzpunktverschiebung wird bei der Referenzpunktfahrt automatisch herausgefahren.
  Beim Start des aktiven Referenzierens werden laufende Verfahrbewegungen abgebrochen.
- **Passives Referenzieren**
  Der Referenzierauftrag führt keine eigene Referenzierbewegung durch. Beim Erkennen der Referenzmarke während einer anwenderseitig initiierten Bewegung wird die Istposition auf den am "MC_Home" angegebenen Wert gesetzt.
  Passives Referenzieren wird auch fliegendes Referenzieren genannt.

- **Direktes Referenzieren**
  Mit dem Referenzierauftrag wird die Istposition direkt auf den am "MC_Home" angegebenen Wert gesetzt oder um diesen verschoben.
- **Absolutwertgeberjustage**
  Die Absolutwertgeberjustage gleicht den vorliegenden absoluten Istwert mit dem am "MC_Home" angegebenen Wert ab.

Referenziermodus

Abhängig von der Art der Referenzmarke und der Referenzmarkensuche werden folgende Referenziermodi (Seite 1731) unterschieden:

- Referenzieren mit Referenznocken und Nullmarke über PROFIdrive-Telegramm
- Referenzieren mit Nullmarke über PROFIdrive-Telegramm
- Referenzieren mit Digitaleingang

Begriffe

Referenzmarke

Eine Referenzmarke ist ein Eingangssignal, bei dessen Auftreten den Istwerten eine bekannte mechanische Position zugeordnet werden kann.

Eine Referenzmarke kann sein:

- **Eine Nullmarke**
  Die Nullmarke eines Inkrementalgebers oder eine Externe Nullmarke wird als Referenzmarke verwendet.
  Die Nullmarke wird am Antriebsmodul bzw. Gebermodul erfasst und im PROFIdrive-Telegramm übertragen. Nehmen Sie die Einstellung und Auswertung als Gebernullmarke oder Externe Nullmarke am Antriebsmodul bzw. Gebermodul vor.
- **Eine Flanke am Digitaleingang**
  Die fallende oder steigende Flanke an einem Digitaleingang wird als Referenzmarke verwendet.

Referenznocken

Wenn mehrere Nullmarken im Verfahrbereich vorliegen, dient der Referenznocken zur Auswahl einer spezifischen Nullmarke vor oder hinter dem Referenznocken.

Referenzmarkenposition

Ist die der Referenzmarke zugeordnete Position.

Die Referenzmarkenposition entspricht der Referenzpunktposition minus Referenzpunktverschiebung.

Referenzpunkt

Die Achse fährt am Ende der aktiven Referenzierbewegung auf den Referenzpunkt.

Referenzpunktverschiebung

Die Differenz zwischen der Referenzpunktposition und der Referenzmarkenposition ist die Referenzpunktverschiebung.

Eine Referenzpunktverschiebung ist nur bei aktivem Referenzieren wirksam. Die Verschiebung wird nach der Synchronisation der Achse über die Motion Control-Anweisung "MC_Home" herausgefahren. Bei Achsen mit Moduloeinstellung wird die Referenzpunktverschiebung immer mit der Richtungseinstellung für den kürzesten Weg herausgefahren.

Richtungsumkehr am Hardware-Endschalter (Umkehrnocken)

Die Hardware-Endschalter können bei aktivem Referenzieren als Umkehrnocken verwendet werden. Falls die Referenzmarke nicht erkannt oder von der falschen Seite angefahren wurde, wird die Fahrt nach dem Umkehrnocken in entgegengesetzter Richtung fortgesetzt.

Referenziermodus

Für die Technologieobjekte Positionierachse/Gleichlaufachse und Externer Geber stehen bei Inkrementalgebern verschiedene Referenziermodi zur Verfügung. Der Referenziermodus wird in der Konfiguration eingestellt.

Referenzieren mit Referenznocken und Nullmarke über PROFIdrive-Telegramm

Das System prüft das Erreichen des Referenznockens. Nachdem der Referenznocken erreicht und in die parametrierte Referenzierrichtung wieder verlassen wurde, wird die Nullmarkenerfassung über das PROFIdrive-Telegramm aktiviert.

Mit dem Erreichen der Nullmarke in der vorgewählten Richtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.

Referenzieren mit Nullmarke über PROFIdrive-Telegramm

Das System aktiviert die Nullmarkenerfassung, sobald sich der Istwert des Technologieobjekts in die parametrierte Referenzierrichtung bewegt.

Mit dem Erreichen der Nullmarke in der vorgegebenen Referenzierrichtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.

Referenzieren mit Digitaleingang

Das System prüft den Zustand des Digitaleingangs, sobald sich der Istwert der Achse bzw. des Gebers in die parametrierte Referenzierrichtung bewegt.

Mit dem Erreichen der Referenzmarke (Setzen des Digitaleingangs) in der vorgegebenen Referenzierrichtung wird die Istposition des Technologieobjekts auf die Referenzmarkenposition gesetzt.

Hinweis

Die Digitaleingänge müssen in das Teilprozessabbild "TPA OB Servo" gelegt werden.

Die Filterzeit der Digitaleingänge muss kleiner als die Dauer des Eingangssignals am Referenzierschalter gewählt werden.

Siehe auch

Referenzieren bei SINAMICS-Antrieben mit Externer Nullmarke (Seite 2223)

Aktives Referenzieren mit Referenznocken und Nullmarke

Die folgenden Beispiele zeigen Referenzierbewegungen in positive und negative Richtung.

Beispiel für Referenzieren in positive Richtung

Die Fahrt auf die Referenzmarke und den Referenzpunkt erfolgt in positiver Richtung.

Das folgende Bild zeigt die Referenzierbewegung mit folgenden Einstellungen:

- Aktives Referenzieren mit Referenznocken und Nullmarke
- Anfahren in positive Richtung
- Referenzieren in positive Richtung
- Positive Referenzpunktverschiebung

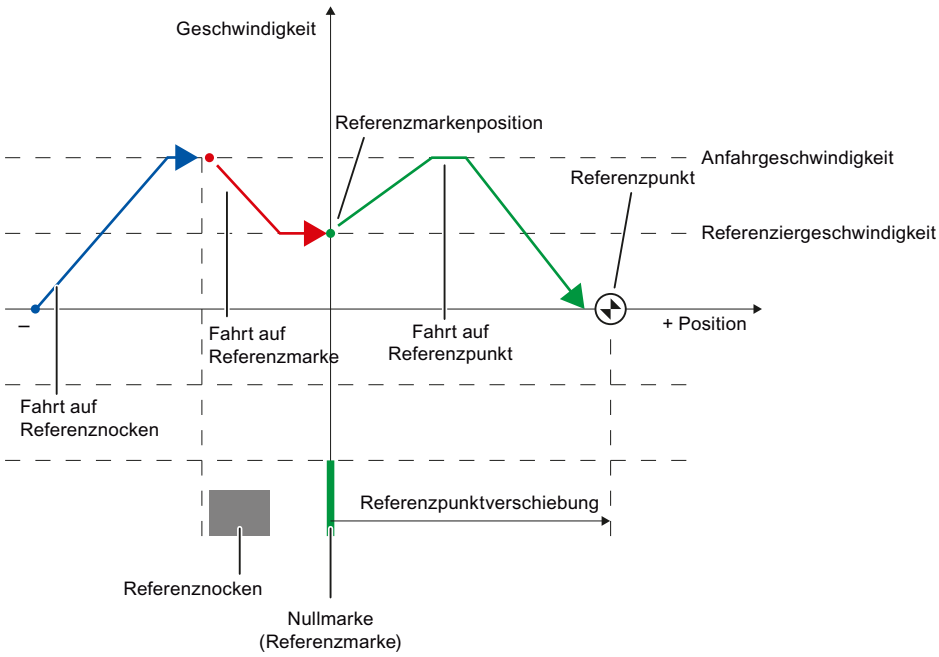

Beispiel für Referenzieren in negative Richtung

Die Fahrt auf die Referenzmarke erfolgt in negativer Richtung durch eine Richtungsumkehr während des Referenziervorgangs. Die Fahrt auf den Referenzpunkt bedingt eine weitere Richtungsumkehr und erfolgt in positiver Richtung.

Das folgende Bild zeigt die Referenzierbewegung mit folgenden Einstellungen:

- Aktives Referenzieren mit Referenznocken und Nullmarke
- Anfahren in positive Richtung
- Referenzieren in negative Richtung
- Positive Referenzpunktverschiebung

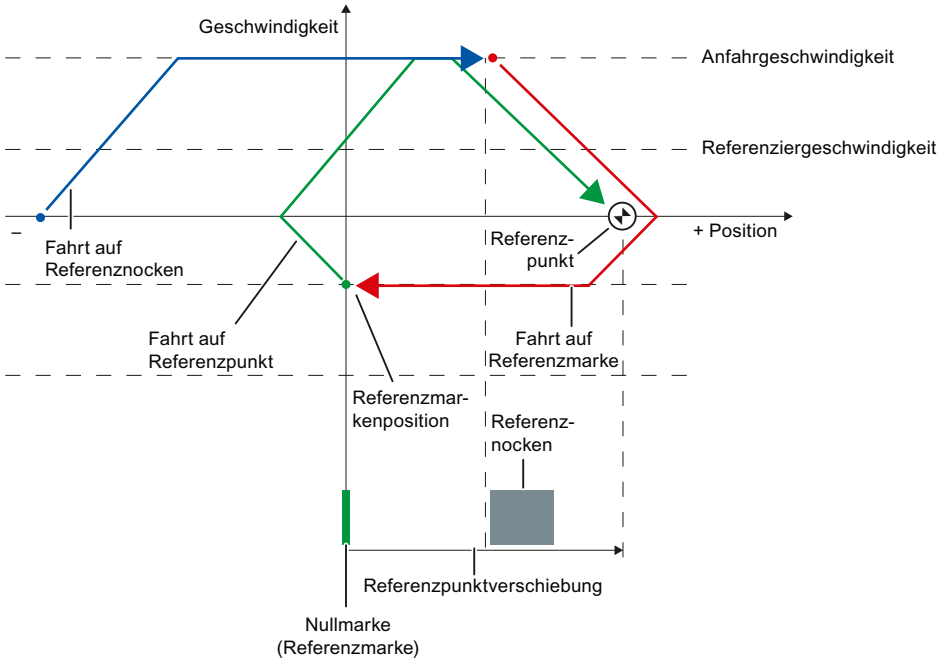

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Start des aktiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Fahrt auf den Referenznocken
3. Erkennen des Referenznockens in Referenzierrichtung und Fahren mit Referenziergeschwindigkeit
4. Verlassen des Referenznockens und Fahrt auf die Referenzmarke
   Mit dem Verlassen des Referenznockens wird die Erfassung der Referenzmarke aktiviert.

5. Erkennen der Referenzmarke
   Die Position des Technologieobjekts wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   - Parameter "Mode" an "MC_Home" = 3
     Position = Wert in Parameter "Position" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"
   - Parameter "Mode" an "MC_Home" = 5
     Position = Wert in der Variable "<TO>.Homing.HomePosition" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"

   Hinweis

   Parameter "MC_Home.Mode"

   Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

6. Fahrt auf den Referenzpunkt
   - Parameter "Mode" an "MC_Home" = 3
     Die Achse fährt auf die Position, die in Parameter "Position" angegeben ist.
   - Parameter "Mode" an "MC_Home" = 5
     Die Achse fährt auf die Position, die in der Variable "<TO>.Homing.HomePosition" angegeben ist.

   Hinweis

   Wenn die Geschwindigkeit auf der Strecke vom Erkennen des Referenznockens bis zur Nullmarke nicht auf die Referenziergeschwindigkeit reduziert werden kann, dann wird mit der Geschwindigkeit referenziert, die beim Überfahren der Nullmarke vorliegt.

Siehe auch

Referenzieren bei SINAMICS-Antrieben mit Externer Nullmarke (Seite 2223)

Aktives Referenzieren mit Nullmarke

Das folgende Bild zeigt beispielhaft die Referenzierbewegung mit folgenden Einstellungen:

- Aktives Referenzieren mit Nullmarke
- Referenzieren in positive Richtung
- Positive Referenzpunktverschiebung

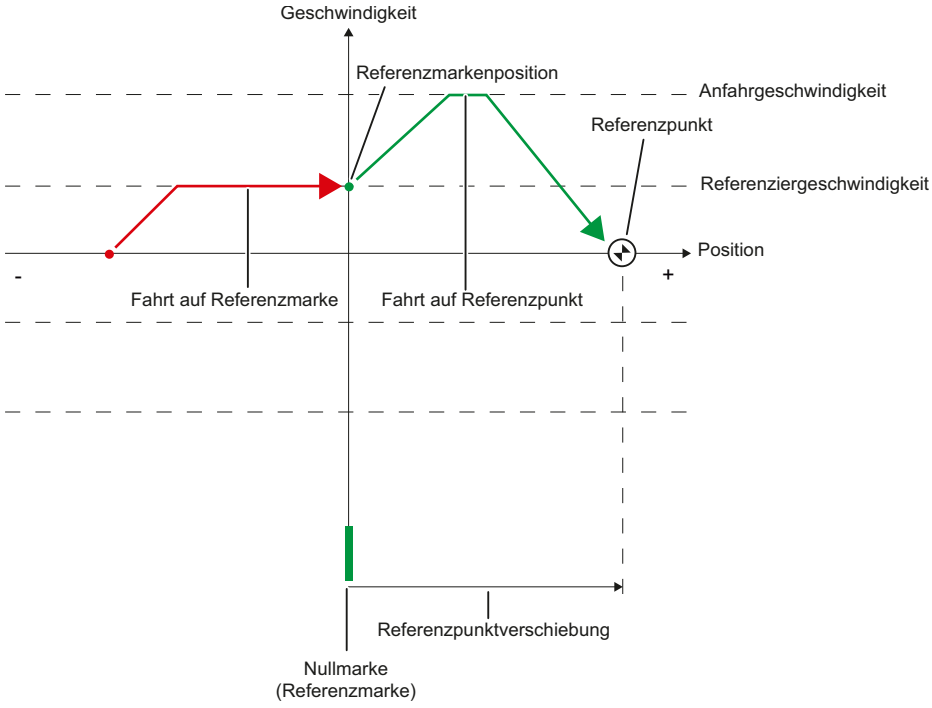

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Start des aktiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Fahrt auf die Referenzmarke in Referenzierrichtung mit der Referenziergeschwindigkeit

3. Erkennen der Referenzmarke
   Die Position der Achse bzw. des Gebers wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   - Parameter "Mode" an "MC_Home" = 3
     Position = Wert in Parameter "Position" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"
   - Parameter "Mode" an "MC_Home" = 5
     Position = Wert in der Variable "<TO>.Homing.HomePosition" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"

   Hinweis

   Parameter "MC_Home.Mode"

   Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

4. Fahrt auf den Referenzpunkt
   - Parameter "Mode" an "MC_Home" = 3
     Die Achse fährt auf die Position, die in Parameter "Position" angegeben ist.
   - Parameter "Mode" an "MC_Home" = 5
     Die Achse fährt auf die Position, die in der Variable "<TO>.Homing.HomePosition" angegeben ist.

Siehe auch

Referenzieren bei SINAMICS-Antrieben mit Externer Nullmarke (Seite 2223)

Aktives Referenzieren mit Digitaleingang

Das folgende Bild zeigt beispielhaft die Referenzierbewegung mit folgenden Einstellungen:

- Aktives Referenzieren mit Digitaleingang
- Anfahren in positive Richtung

- Referenzmarke an positiver Seite des Digitaleingangs
- Positive Referenzpunktverschiebung

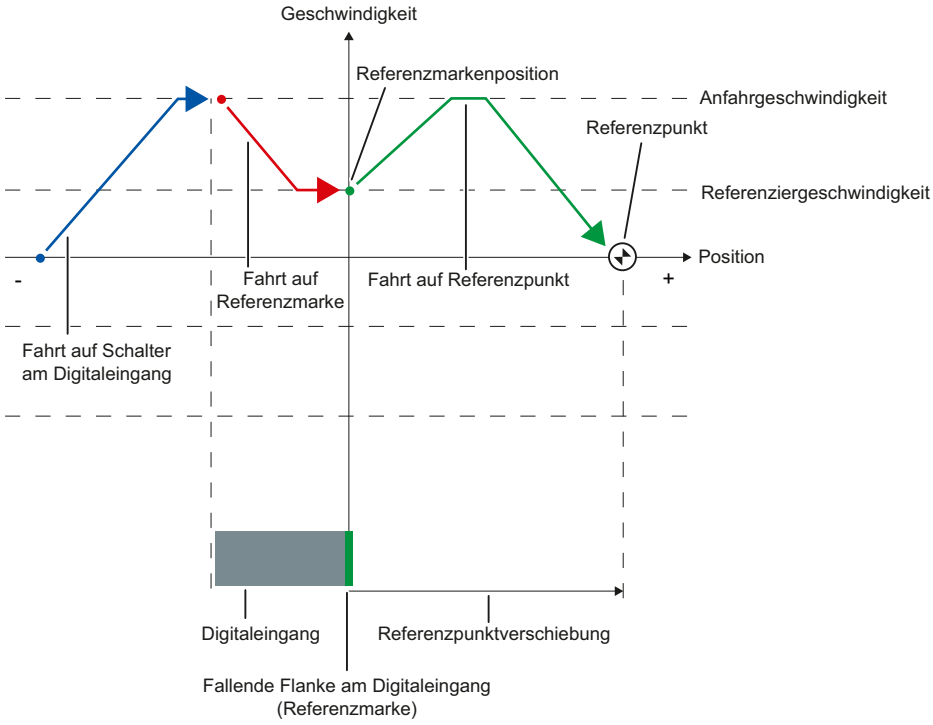

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Start des aktiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Erkennen der steigenden Flanke am Digitaleingang und Fahren mit Referenziergeschwindigkeit
3. Fahrt auf die Referenzmarke

4. Erkennen der Referenzmarke
   Im Beispiel stellt die fallende Flanke des Schalters am Digitaleingang die Referenzmarke dar.
   Die Position der Achse bzw. des Gebers wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   - Parameter "Mode" an "MC_Home" = 3
     Position = Wert in Parameter "Position" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"
   - Parameter "Mode" an "MC_Home" = 5
     Position = Wert in der Variable "<TO>.Homing.HomePosition" minus "<TO>.Sensor[n].ActiveHoming.HomePositionOffset"

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

5. Fahrt auf den Referenzpunkt
   - Parameter "Mode" an "MC_Home" = 3
     Die Achse fährt auf die Position, die in Parameter "Position" angegeben ist.
   - Parameter "Mode" an "MC_Home" = 5
     Die Achse fährt auf die Position, die in der Variable "<TO>.Homing.HomePosition" angegeben ist.

Hinweis

Wenn die Geschwindigkeit auf der Strecke vom Erkennen der steigenden Flanke bis zur fallenden Flanke nicht auf die Referenziergeschwindigkeit reduziert werden kann, dann wird mit der Geschwindigkeit referenziert, die beim Überfahren der Referenzmarke vorliegt.

Passives Referenzieren mit Referenznocken und Nullmarke

Das folgende Bild zeigt beispielhaft die Referenzierbewegung mit folgenden Einstellungen:

- Passives Referenzieren mit Referenznocken und Nullmarke
- Referenzieren in positive Richtung

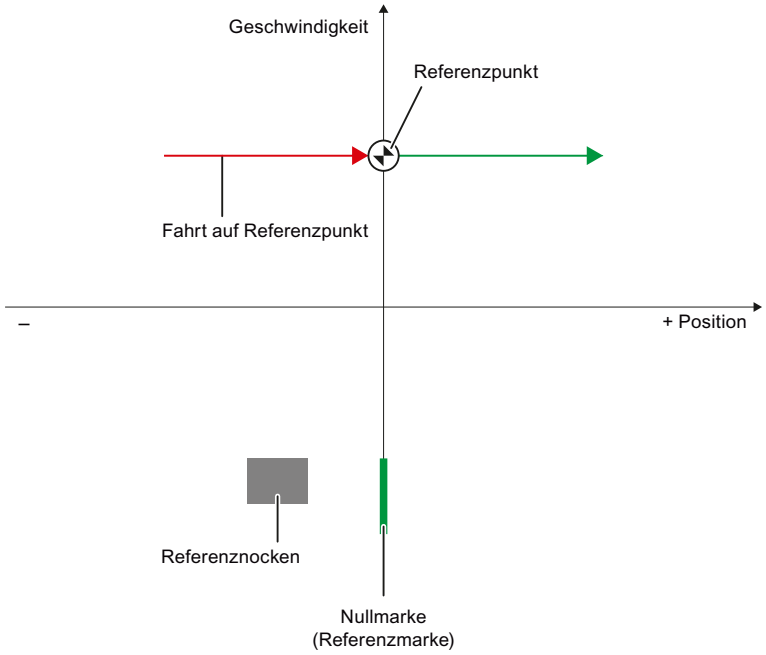

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Aktivieren des passiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Fahren durch einen Bewegungsauftrag vom Anwender
   Die Erfassung des Referenznockens und der Referenzmarke wird aktiviert, wenn sich der Positionsistwert der Achse bzw. des Gebers in die parametrierte Referenzierrichtung bewegt.
3. Erkennen des Referenznockens

4. Verlassen des Referenznockens
   Das Verlassen des Referenznockens aktiviert die Erfassung der Referenzmarke.
5. Erkennen der Referenzmarke
   Die Position der Achse bzw. des Gebers wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   – Parameter "Mode" an "MC_Home" = 2, 8
     Position = Wert in Parameter "Position"
   – Parameter "Mode" an "MC_Home" = 10
     Position = Wert in der Variable "<TO>.Homing.HomePosition"

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Hinweis

Wenn sich die Bewegungsrichtung nach dem Verlassen des Referenznockens und vor Erkennen der Referenzmarke ändert, dann muss der Referenznocken erneut erkannt werden. Die Motion Control-Anweisung "MC_Home" bleibt aktiviert.

Passives Referenzieren mit Nullmarke

Das folgende Bild zeigt beispielhaft die Referenzierbewegung mit folgenden Einstellungen:

- Passives Referenzieren mit Nullmarke
- Referenzieren in positive Richtung

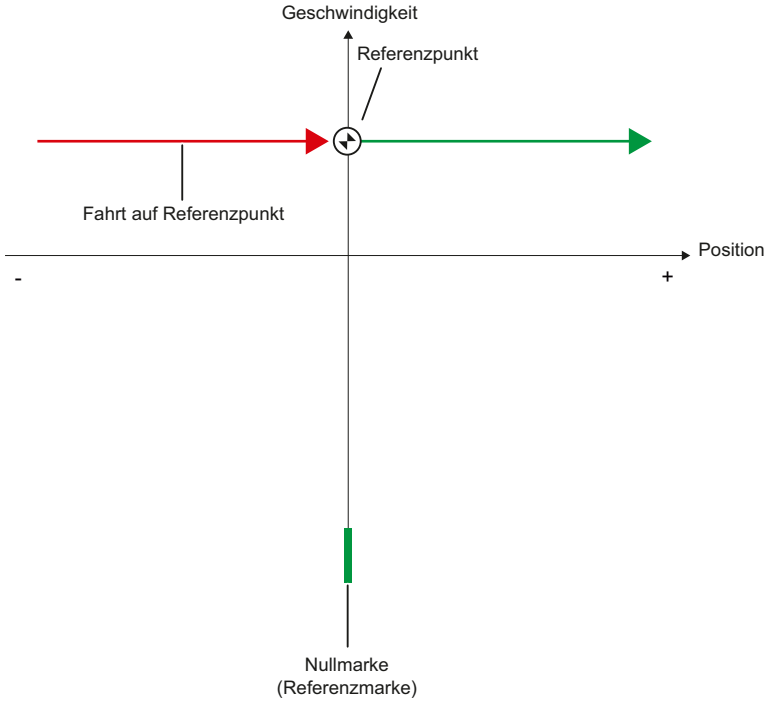

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Aktivieren des passiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Fahren durch einen Bewegungsauftrag vom Anwender
   Die Erfassung der Referenzmarke wird aktiviert, wenn sich der Positionsistwert der Achse bzw. des Gebers in die parametrierte Referenzierrichtung bewegt.
3. Erkennen der Referenzmarke
   Die Position der Achse bzw. des Gebers wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   - Parameter "Mode" an "MC_Home" = 2, 8
     Position = Wert in Parameter "Position"
   - Parameter "Mode" an "MC_Home" = 10
     Position = Wert in der Variable "<TO>.Homing.HomePosition"

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Passives Referenzieren mit Digitaleingang

Das folgende Bild zeigt beispielhaft die Referenzierbewegung mit folgenden Einstellungen:

- Passives Referenzieren mit Digitaleingang
- Referenzieren in positive Richtung
- Referenzmarke an positiver Seite des Digitaleingangs

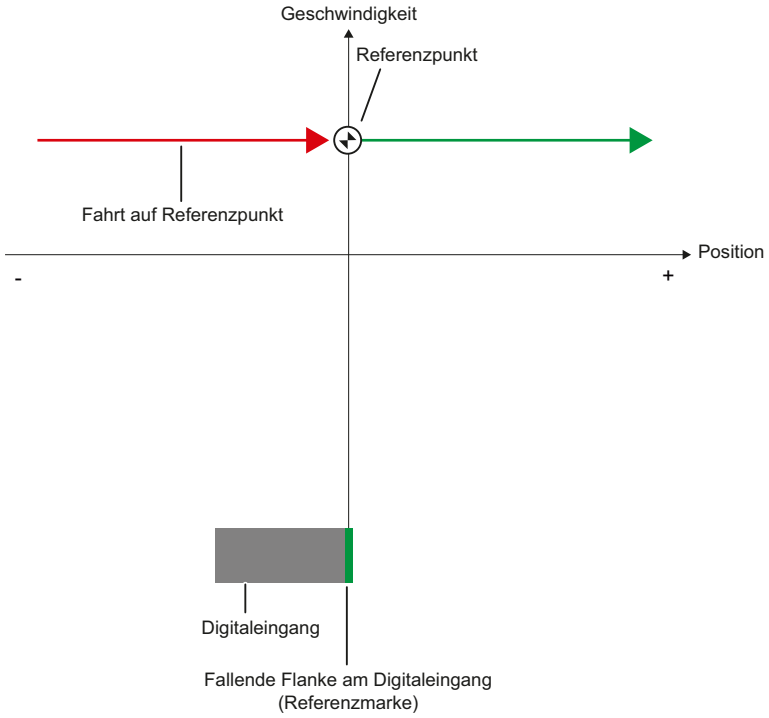

Bewegungsablauf

Die Bewegung läuft folgendermaßen ab:

1. Aktivieren des passiven Referenzierens über die Motion Control-Anweisung "MC_Home"
2. Fahren durch einen Bewegungsauftrag vom Anwender
   Die Erfassung der Referenzmarke am Digitaleingang wird aktiviert, wenn sich der Positionsistwert der Achse bzw. des Gebers in die parametrierte Referenzierrichtung bewegt.
3. Erkennen der Referenzmarke
   Im Beispiel stellt die fallende Flanke des Schalters am Digitaleingang die Referenzmarke dar.
   Die Position der Achse bzw. des Gebers wird bei Erkennen der Referenzmarke abhängig vom eingestellten Mode gesetzt:
   - Parameter "Mode" an "MC_Home" = 2, 8
     Position = Wert in Parameter "Position"
   - Parameter "Mode" an "MC_Home" = 10
     Position = Wert in der Variable "<TO>.Homing.HomePosition"

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Richtungsumkehr am Hardware-Endschalter (Umkehrnocken)

Beim aktiven Referenzieren können optional die Hardware-Endschalter als Umkehrnocken genutzt werden. Falls die Referenzmarke nicht erkannt oder nicht in Referenzierrichtung gefahren wurde, wird die Fahrt nach dem Umkehrnocken in entgegengesetzte Richtung mit Anfahrgeschwindigkeit fortgesetzt.

Beim Erreichen des Hardware-Endschalters werden die Dynamikvoreinstellungen wirksam. Dabei wird nicht mit der Notstopp-Verzögerung abgebremst.

ACHTUNG

Fahren auf einen mechanischen Anschlag vermeiden

Stellen Sie durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgeschwindigkeit gering.
- Vergrößern Sie die konfigurierte Beschleunigung/Verzögerung.
- Vergrößern Sie den Abstand zwischen Hardware-Endschalter und mechanischem Anschlag.

Direktes Referenzieren

Die Position der Technologieobjekte Positionierachse/Gleichlaufachse bzw. Externer Geber kann abhängig vom eingestellten Mode an "MC_Home" absolut oder relativ gesetzt werden.

Istposition absolut setzen

Um die Position absolut zu setzen, gehen Sie folgendermaßen vor:

1. Tragen Sie bei der Motion Control-Anweisung "MC_Home" in Parameter "Position" die absolute Position ein.
2. Rufen Sie die Motion Control-Anweisung "MC_Home" mit Parameter "Mode" = 0 auf.

Die Position wird auf den in Parameter "Position" vorgegebenen Wert gesetzt.

Istposition relativ setzen

Um die Position relativ zu setzen, gehen Sie folgendermaßen vor:

1. Tragen Sie bei der Motion Control-Anweisung "MC_Home" in Parameter "Position" die relative Position ein.
2. Rufen Sie die Motion Control-Anweisung "MC_Home" mit Parameter "Mode" = 1 auf.

Die Position wird auf die aktuelle Position plus den in Parameter "Position" vorgegebenen Wert gesetzt.

Absolutwertgeberjustage

Bei der Absolutwertgeberjustage ermittelt Motion Control einen Absolutwert-Offset, der remanent in der CPU gespeichert wird.

Die Istposition der Achse bzw. des Gebers wird abhängig vom eingestellten Mode an der Motion Control-Anweisung "MC_Home" absolut oder relativ gesetzt:

- Parameter "Mode" = 7 (Absolute Positionsvorgabe)
  Position = Wert in Parameter "Position"
- Parameter "Mode" = 6 (Relative Positionsvorgabe)
  Position = Aktuelle Position + Wert in Parameter "Position"

Rücksetzen des Status "Referenziert"

Inkrementalgeber

In folgenden Fällen wird der Status "Referenziert" zurückgesetzt und das Technologieobjekt muss neu referenziert werden:

- Fehler im Sensorsystem/Geberausfall
- Anstoß des aktiven Referenzierens mit der Motion Control-Anweisung "MC_Home" mit "Mode" = 3, 5 (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)

  Hinweis

  Parameter "MC_Home.Mode"

  Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

- Anstoß des passiven Referenzierens mit der Motion Control-Anweisung "MC_Home" mit "Mode" = 2, 8, 10 (Nach dem erfolgreichen Abschluss des Referenziervorgangs wird der Status "Referenziert" wieder gesetzt.)
- Tausch der CPU
- Tausch der SIMATIC Memory Card
- NETZ-AUS
- Urlöschen
- Veränderung der Geberkonfiguration
- Restart des Technologieobjekts
- Wiederherstellen der CPU-Werkseinstellung
- Übertragen eines anderen Projekts in die Steuerung

Absolutwertgeber

In folgenden Fällen wird der Status "Referenziert" zurückgesetzt und das Technologieobjekt muss neu referenziert werden:

- Tausch der CPU
- Veränderung der Geberkonfiguration
- Wiederherstellen der CPU-Werkseinstellung
- Übertragen eines anderen Projekts in die Steuerung

Wenn Sie einen neuen Absolutwertgeber einsetzen, müssen Sie den Absolutwertgeber neu referenzieren.

Das Urlöschen der CPU oder das Hochrüsten eines Projekts erfordert keine neue Absolutwertgeberjustage.

Variablen

Folgende Variablen des Technologieobjekts sind für das Referenzieren relevant:

Statusanzeigen	
Variable	**Beschreibung**
<TO>StatusWord.X11 (HomingCommand)	Referenzierbefehl aktiv
<TO>StatusWord.X5 (HomingDone)	Technologieobjekt referenziert
<TO>ErrorWord.X10 (HomingFault)	Fehler beim Referenzieren aufgetreten

Hinweis

Auswertung der Bits in StatusWord, ErrorWord und WarningWord

Beachten Sie die Hinweise in Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Fahren auf den Referenznocken	
Variable	**Beschreibung**
<TO>.Homing.ApproachDirection	Start- bzw. Anfahrrichtung beim Fahren auf den Referenznocken
<TO>.Homing.ApproachVelocity	Geschwindigkeit zum Fahren auf den Referenznocken

Fahren auf die Referenzmarke	
Variable	**Beschreibung**
<TO>.Sensor[n].ActiveHoming.Direction	Referenzierrichtung
<TO>.Homing.ReferencingVelocity	Geschwindigkeit zum Anfahren der Referenzmarke

Fahren auf den Referenzpunkt	
Variable	**Beschreibung**
<TO>.Homing.ApproachVelocity	Geschwindigkeit zum Fahren auf den Referenzpunkt

Positionen	
Variable	**Beschreibung**
<TO>.Homing.AutoReversal	Umkehren an den Hardware-Endschaltern
<TO>.Homing.HomePosition	Referenzpunkt
<TO>.StatusSensor[n].AbsEncoderOffset	Berechneter Offset nach der Absolutwertgeberjustage

Parameter für aktives Referenzieren	
Variable	**Beschreibung**
<TO>.Sensor[n].ActiveHoming.Mode	Referenziermodus
<TO>.Sensor[n].ActiveHoming.SideInput	Seite des Digitaleingangs
<TO>.Sensor[n].ActiveHoming.Direction	Referenzierrichtung bzw. Anfahrrichtung
<TO>.Sensor[n].ActiveHoming.DigitalInputAddress	Adresse digitaler Eingang
<TO>.Sensor[n].ActiveHoming.HomePositionOffset	Offset von der Referenzmarke zum Referenzpunkt

Parameter für passives Referenzieren	
Variable	**Beschreibung**
<TO>.Sensor[n].PassiveHoming.Mode	Referenziermodus
<TO>.Sensor[n].PassiveHoming.SideInput	Seite des Digitaleingangs
<TO>.Sensor[n].PassiveHoming.Direction	Referenzierrichtung bzw. Anfahrrichtung
<TO>.Sensor[n].PassiveHoming.DigitalInputAddress	Adresse digitaler Eingang

Positionsüberwachungen

Zur Überwachung der Positionierung und Bewegung stehen am Technologieobjekt Positionierachse/Gleichlaufachse folgende Funktionen zur Verfügung:

- Positionierüberwachung (Seite 1749)
  Der Positionsistwert muss innerhalb einer bestimmten Zeit ein Positionierfenster erreichen und für eine minimale Verweildauer in diesem Positionierfenster verbleiben.
- Schleppfehlerüberwachung (Seite 1750)
  Auf Basis einer geschwindigkeitsabhängigen Schleppfehlergrenze wird der Schleppfehler überwacht. Der zulässige maximale Schleppfehler ist von der Sollgeschwindigkeit abhängig.

Bei Verletzung einer Überwachung werden Technologiealarme ausgegeben. Das Technologieobjekt reagiert entsprechend der Alarmreaktion.

Positionierüberwachung

Die Positionierüberwachung überwacht das Verhalten der Istposition am Ende der Sollwertberechnung.

Sobald die Sollgeschwindigkeit den Wert null erreicht, muss sich die Istposition innerhalb einer Toleranzzeit im Positionierfenster befinden. Der Istwert darf während der minimalen Verweildauer nicht das Positionierfenster verlassen.

Wenn die Istposition am Ende einer Positionierbewegung innerhalb der Toleranzzeit das Positionierfenster erreicht und für die minimale Verweildauer im Positionierfenster verbleibt,

wird im Technologie-Datenbaustein "<TO>.StatusWord.X6 (Done)" gesetzt. Nach Ablauf der minimalen Verweildauer wird ebenfalls der Parameter "Done" der entsprechenden Motion Control-Anweisung gesetzt. Damit ist ein Bewegungsauftrag abgeschlossen.

Das folgende Bild zeigt den zeitlichen Ablauf und das Positionierfenster:

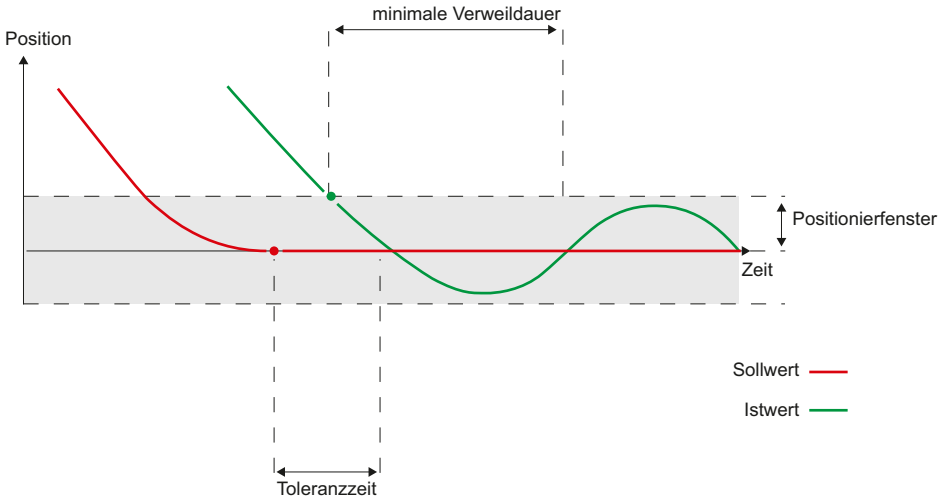

Die Positionierüberwachung unterscheidet nicht, wie die Sollwertinterpolation beendet wird. Das Ende der Sollwertinterpolation kann z. B. folgendermaßen erreicht werden:

- Durch sollwertseitiges Erreichen der Zielposition
- Durch lagegeregeltes Anhalten während der Bewegung durch die Motion Control-Anweisung "MC_Halt"

Verletzung der Positionierüberwachung

In folgenden Fällen wird durch die Positionierüberwachung der Technologiealarm 541 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen):

- Der Istwert erreicht innerhalb der Toleranzzeit nicht das Positionierfenster.
- Der Istwert verlässt während der minimalen Verweildauer das Positionierfenster.

Schleppfehlerüberwachung

Der Schleppfehler am Technologieobjekt Positionierachse/Gleichlaufachse wird auf Basis einer geschwindigkeitsabhängigen Schleppfehlergrenze überwacht. Der zulässige Schleppfehler ist von der Sollgeschwindigkeit abhängig.

Bei Geschwindigkeiten kleiner als eine einstellbare untere Geschwindigkeit ist ein konstanter, zulässiger Schleppfehler vorgebbar.

Oberhalb dieser unteren Geschwindigkeit wird der zulässige Schleppfehler proportional zur Sollgeschwindigkeit vergrößert. Der vorgebbare maximal zulässige Schleppfehler ist der Grenzwert bei maximaler Geschwindigkeit.

Berechnung des Schleppfehlers

Der Schleppfehler ist die Differenz von Positionssollwert und Positionsistwert. Die Übertragungszeiten des Sollwerts zum Antrieb und des Positionsistwerts zur Steuerung werden bei der Berechnung des Schleppfehlers berücksichtigt, d. h. herausgerechnet.

Warngrenze

Für den Schleppfehler kann eine Warngrenze vorgegeben werden. Die Warngrenze wird als Prozentwert eingestellt und wirkt relativ zum aktuell zulässigen Schleppfehler. Wenn die Warngrenze des Schleppfehlers erreicht ist, wird der Technologie-Alarm 522 ausgegeben. Dies ist eine Warnung und beinhaltet keine Alarmreaktion.

Überschreiten des zulässigen Schleppfehlers

Beim Überschreiten des zulässigen Schleppfehlers wird der Technologie-Alarm 521 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen).

Bei aktivierter Kraft-/Momentenbegrenzung kann die Überwachung des zulässigen Schleppfehlers deaktiviert werden.

Stillstandssignal

Wenn die Istgeschwindigkeit das Stillstandsfenster erreicht und für die minimale Verweildauer im Stillstandsfenster verbleibt, wird der Stillstand der Achse angezeigt.

Variablen

Folgende Variablen des Technologieobjekts sind bei der Positionierüberwachung und für das Stillstandssignal relevant:

Statusanzeigen	
Variable	**Beschreibung**
<TO>.StatusWord.X7 (Standstill)	Wird auf den Wert "TRUE" gesetzt, wenn die Istgeschwindigkeit das Stillstandsfenster erreicht und über die minimale Verweildauer nicht verlässt. Das Stillstandssignal ist nur an der Positionierachse/Gleichlaufachse vorhanden.
<TO>.StatusWord.X6 (Done)	**Positionierachse/Gleichlaufachse** Wird auf den Wert "TRUE" gesetzt, wenn der Geschwindigkeitsistwert innerhalb der Toleranzzeit das Positionierfenster erreicht und die minimale Verweildauer im Fenster verweilt.
	Drehzahlachse Wird "TRUE" gesetzt, wenn die Bewegung abgeschlossen und damit die Solldrehzahl gleich null ist.
<TO>.ErrorWord.X12 (Positioning-Fault)	Ein Positionierfehler ist aufgetreten.

Positionen und Zeiten	
Variable	**Beschreibung**
<TO>.PositioningMonitoring.Tolerance-Time	Maximal zulässige Zeit bis zum Erreichen des Positionierfensters Die Zeit wird mit dem Ende der Sollwertinterpolation gestartet.
<TO>.PositioningMonitoring.MinDwell-Time	Minimale Verweildauer im Positionsfenster
<TO>.PositioningMonitoring.Window	Positionierfenster

Stillstandssignal	
Variable	**Beschreibung**
<TO>.StandstillSignal.VelocityThres-hold	Geschwindigkeitsschwelle für das Stillstandssignal
<TO>.StandstillSignal.MinDwellTime	Minimale Verweildauer unter der Geschwindigkeitsschwelle

Folgende Variablen Technologieobjekts sind bei der Schleppfehlerüberwachung relevant:

Statusanzeigen	
Variable	**Beschreibung**
<TO>.StatusPositioning.FollowingError	Aktueller Schleppfehler
<TO>.ErrorWord.X11 (FollowingError-Fault)	Statusanzeige, dass der Schleppfehler zu groß ist
<TO>.WarningWord.X11 (FollowingEr-rorWarning)	Statusanzeige, dass die Schleppfehlerwarngrenze erreicht wurde

Steuerbits	
Variable	**Beschreibung**
<TO>.FollowingError.EnableMonitoring	Schleppfehlerüberwachung aktivieren/deaktivieren

Grenzwerte	
Variable	**Beschreibung**
<TO>.FollowingError.MinVelocity	Untere Sollgeschwindigkeit für die Kennlinie des maximalen Schleppfehlers
<TO>.FollowingError.MinValue	Zulässiger Schleppfehler unterhalb der "<TO>.FollowingEr-ror.MinVelocity"
<TO>.FollowingError.MaxValue	Maximal zulässiger Schleppfehler bei maximaler Geschwindigkeit der Achse
<TO>.FollowingError.WarningLevel	Warngrenze als Prozentwert bezogen auf den maximal zulässigen Schleppfehler (geschwindigkeitsabhängig gemäß Kennlinie)

Regelung

Der Lageregler der Positionierachse/Gleichlaufachse ist ein P-Regler mit Geschwindigkeitsvorsteuerung.

Geschwindigkeitsvorsteuerung

Die Geschwindigkeitsvorsteuerung können Sie verwenden, um den geschwindigkeitsabhängigen Schleppfehler bei der Lageregelung zu minimieren. Dadurch wird gegebenenfalls eine schnellere Positionierung erzielt, weil der Lageregler nur noch Störgrößen ausregeln muss.

Bei Verwendung der Geschwindigkeitsvorsteuerung wird zusätzlich der Geschwindigkeitssollwert additiv auf den Ausgang des Lagereglers geschaltet. Diesen zusätzlichen Sollwert können Sie mit einem Faktor wichten.

Dynamic Servo Control (DSC)

Bei Antrieben, die Dynamic Servo Control (DSC) unterstützen, können Sie optional den Lageregler im Antrieb verwenden. Wenn Sie Telegramme verwenden, die DSC unterstützen, wird DSC automatisch aktiviert. Der Lageregler im Antrieb wird üblicherweise im schnellen Drehzahlregeltakt ausgeführt. Dadurch wird die Regelgüte bei digital gekoppelten Antrieben verbessert.

Für den Einsatz von DSC müssen folgende Voraussetzungen erfüllt sein:

- Der Motorgeber (erster Geber im Telegramm) des Antriebs ist als erster Geber für das Technologieobjekt verwendet.
- Am Antrieb ist eines der folgenden PROFIdrive-Telegramme verwendet:
  - Standardtelegramm 5 oder 6
  - SIEMENS-Telegramm 105 oder 106

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Regelungsstruktur

Das folgende Bild zeigt die effektive Regelungsstruktur **ohne** DSC:

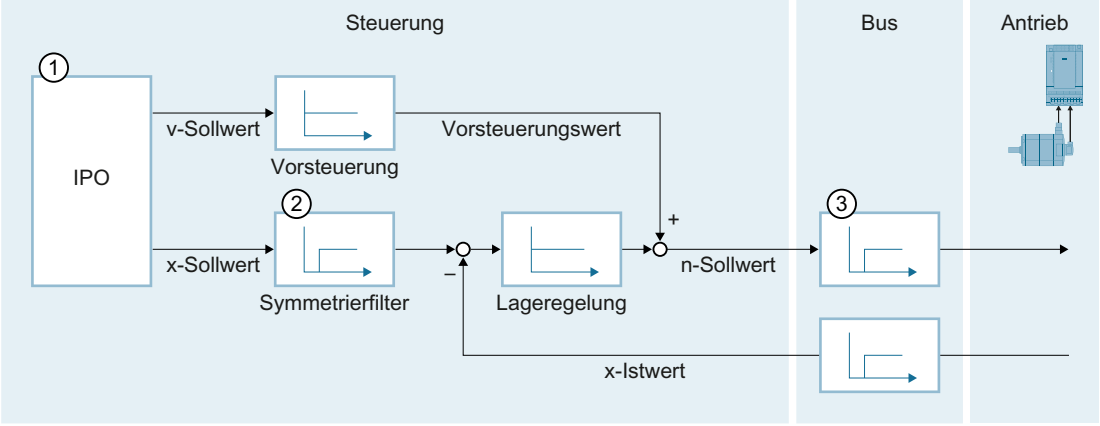

① Interpolator mit Bewegungsführung
② Interne Berücksichtigung der Signallaufzeiten und der Drehzahl-Regelkreis-Ersatzzeit
③ Kommunikation Steuerung - Antrieb

Das folgende Bild zeigt die effektive Regelungsstruktur **mit** DSC:

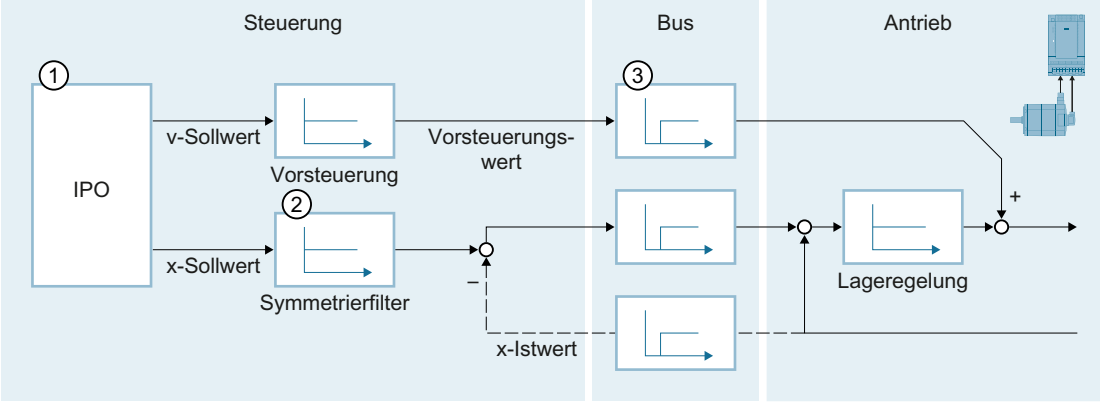

① Interpolator mit Bewegungsführung
② Interne Berücksichtigung der Drehzahl-Regelkreis-Ersatzzeit
③ Kommunikation Steuerung - Antrieb

Siehe auch

Konfiguration - Regelkreis (Seite 1894)

Variablen

Folgende Variablen des Technologieobjekts sind für die Regelung relevant:

Parameter		
Variable	**Beschreibung**	
<TO>.PositionControl.Kv	P-Verstärkung der Lageregelung	
<TO>.PositionControl.Kpc	Geschwindigkeitsvorsteuerung der Lageregelung (in %)	
<TO>.PositionControl.EnableDSC	DSC aktivieren	
<TO>.DynamicAxisModel.VelocityTimeConstant	Drehzahl-Regelkreis-Ersatzzeit [s]	
<TO>.PositionControl.ControlDifferenceQuantization.Mode	Art der Quantisierung Konfiguration einer Quantisierung bei Anschluss eines Antriebs mit Schrittmotor-Schnittstelle	
	0	Keine Quantisierung
	1	Quantisierung entsprechend Geberauflösung
	2	Quantisierung auf direkten Wert (Werteingabe in "<TO>.PositionControl.ControlDifferenceQuantization.Value")
	Die Konfiguration erfolgt über die Parameteransicht (Datenstruktur).	
<TO>.PositionControl.ControlDifferenceQuantization.Value	Wert der Quantisierung Konfiguration eines Werts bei Quantisierung auf direktem Wert ("<TO>.PositionControl.ControlDifferenceQuantization.Mode" = 2) Der Quantisierungswert wird in der Positionseinheit der Achse angegeben. Die Konfiguration erfolgt über die Parameteransicht (Datenstruktur).	

Ablaufverhalten

Organisationsbausteine für Motion Control

Wenn Sie ein Technologieobjekt anlegen, werden automatisch Organisationsbausteine für die Bearbeitung der Technologieobjekte angelegt. Die Motion Control-Funktionalität der Technologieobjekte erzeugt eine eigene Ablaufebene und wird entsprechend dem Motion Control-Applikationszyklus aufgerufen.

Folgende Organisationsbausteine werden angelegt:

- **MC-Servo [OB91]**
  Berechnung des Lagereglers
- **MC-Interpolator [OB92]**
  Auswertung der Motion Control-Anweisungen, Sollwertgenerierung und Überwachungsfunktionalität

Die Organisationsbausteine sind geschützt (Know-how-Schutz). Der Programmcode kann nicht eingesehen oder verändert werden.

Das Taktverhältnis der beiden Organisationsbausteine zueinander ist immer 1:1. MC-Servo [OB91] wird immer vor MC-Interpolator [OB92] ausgeführt.

Sie können den Applikationszyklus und die Priorität der Organisationsbausteine entsprechend Ihren Anforderungen an die Regelgüte und die Systembelastung einstellen.

MC-PreServo [OB67] und MC-PostServo [OB95]

Zusätzlich können Sie die Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] anlegen.

Die Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] sind programmierbar. Der Aufruf erfolgt im konfigurierten Applikationszyklus. So können die Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] für eine konsistente Datenverarbeitung oder zeitkritische Ereignisse, z. B. das Starten von Bewegungen oder Referenzierfunktionen zum Prozess, die gegebenenfalls synchron zum Applikationszyklus ablaufen müssen, eingesetzt werden.

Applikationszyklus

Den Applikationszyklus, in dem der MC-Servo [OB91] aufgerufen wird, können Sie in den Eigenschaften des Organisationsbausteins unter "Allgemein > Zykluszeit" einstellen:

- **Synchron zum Bus**
  Der MC-Servo [OB91] wird synchron oder untersetzt zu einem Bussystem aufgerufen. Den Sendetakt stellen Sie in den Eigenschaften des ausgewählten Bussystems ein. In der Klappliste "Dezentrale Peripherie" können Sie folgende Bussysteme auswählen:
  - Taktsynchroner PROFIBUS DP
  - Taktsynchrones PROFINET IO

  Den MC-Servo [OB91] können Sie nicht synchron zu einem Bussystem aufrufen, das über einen Kommunikationsprozessor/Kommunikationsmodul (CP/CM) an die CPU angebunden ist.
- **Zyklisch**
  Der MC-Servo [OB91] wird zyklisch mit dem angegebenen Applikationszyklus aufgerufen.

Den Applikationszyklus müssen Sie groß genug wählen, um alle Technologieobjekte in einem Zyklus bearbeiten zu können. Wenn die Bearbeitungszeit der Technologieobjekte größer ist als der Applikationszyklus, treten Überläufe (Seite 1758) auf. Um eine optimale Regelgüte zu erreichen, müssen die Berechnungen der Organisationsbausteine MC-PreServo [OB67], MC-Servo [OB91] und MC-PostServo [OB95] innerhalb eines Sendetakts durchgeführt werden.

Die Laufzeit von MC-Servo [OB91] und MC-Interpolator [OB92] können Sie mit der erweiterten Anweisung "RT_INFO" überprüfen.

Der aktuelle Applikationszyklus (Angabe in µs) der Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] lässt sich über die Startinformation auslesen.

Taktuntersetzung (ab CPU V1.5)

Den Applikationszyklus des MC-Servo [OB91] können Sie zum Sendetakt eines taktsynchronen PROFINET IO-Systems untersetzen. Als Faktor können Sie ein ganzzahliges Vielfaches des Sendetakts einstellen. Für den Applikationszyklus sind Werte bis zum 14-fachen des Sendetakts (maximal 32 ms) möglich.

Wenn Sie einen Taktsynchronalarm-OB und den MC-Servo [OB91] synchron zum selben PROFINET IO-System aufrufen, müssen Sie bei beiden Organisationsbausteinen den gleichen Applikationszyklus einstellen.

Priorität

Die Priorität der Organisationsbausteine können Sie bei Bedarf in deren Eigenschaften unter "Allgemein > Attribute > Priorität" einstellen:

- **MC-Servo [OB91]**
  Priorität 17 bis 26 (Standardwert 26)
- **MC-Interpolator [OB92]**
  Priorität 16 bis 25 (Standardwert 24)

Die Priorität des MC-Servo [OB91] muss mindestens um eins höher sein als die Priorität des MC-Interpolator [OB92].

Die Priorität der Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] entspricht der Priorität des MC-Servo [OB91]. MC-PreServo [OB67] wird direkt vor dem MC-Servo [OB91] aufgerufen. MC-PostServo [OB95] wird direkt nach dem MC-Servo [OB91] aufgerufen.

Siehe auch

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Teilprozessabbild "TPA OB Servo"

Das Teilprozessabbild "TPA OB Servo" wird taktsynchron beim Aufruf des MC-Servo [OB91] für Motion Control zur Verfügung gestellt. Alle von Motion Control verwendeten Antriebe und Geber werden diesem Teilprozessabbild zugeordnet.

Da die Organisationsbausteine MC-PreServo [OB67] und MC-PostServo [OB95] automatisch durch den MC-Servo [OB91] aufgerufen werden, steht auch hier das Teilprozessabbild automatisch zur Verfügung. Wenn Sie einen MC-PreServo [OB67] einsetzen, werden die Daten beim Start des MC-PreServo [OB67] eingelesen. Wenn Sie einen MC-PostServo [OB95] einsetzen, werden die Daten nach dem MC-PostServo [OB95] ausgegeben.

Zusätzlich sollten Sie diesem Teilprozessabbild alle von Motion Control verwendeten E/A-Baugruppen zuordnen (z. B. Hardware-Endschalter). Durch die Zuordnung erfolgt eine zeitlich synchrone Bearbeitung zum Technologieobjekt.

Das Eingangs-Teilprozessabbild wird auch im STOP aktualisiert.

Teilprozessabbild im Anwenderprogramm

Ab der CPU-Version V1.5 können Sie in Ihrem Anwenderprogramm auf das Teilprozessabbild "TPA OB Servo" zugreifen. Damit ist auch die Auswertung des Teilprozessabbildes über die Trace-Funktion möglich.

Ablaufverhalten und Überläufe

Bei der Bearbeitung der Motion Control-Funktionalität werden in jedem Applikationszyklus die Organisationsbausteine MC-Servo [OB91] und MC-Interpolator [OB92] aufgerufen und bearbeitet. Die restliche Zykluszeit steht für die Bearbeitung Ihres Anwenderprogramms zur Verfügung.

Für einen fehlerfreien Programmablauf gelten folgende Regeln:

- In jedem Applikationszyklus muss der MC-Servo [OB91] gestartet und vollständig abgearbeitet werden.
- In jedem Applikationszyklus muss der zugehörige MC-Interpolator [OB92] zumindest gestartet werden.

Folgendes Bild zeigt das fehlerfreie Ablaufverhalten beispielhaft für die Abarbeitung des Organisationsbausteins OB1:

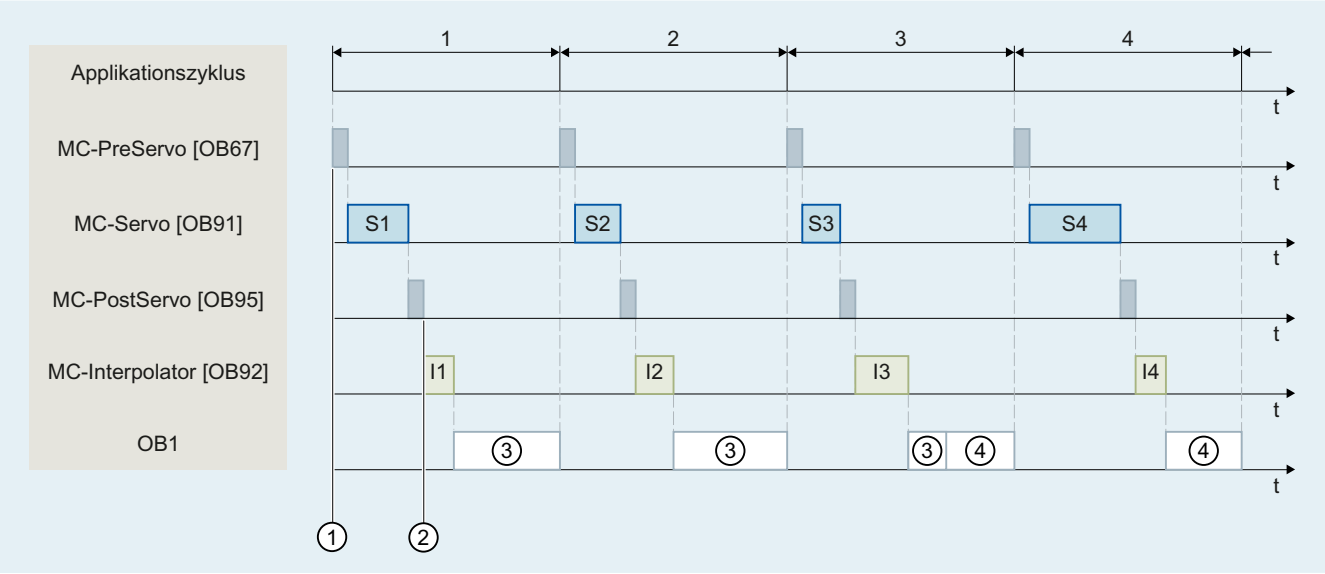

① Eingang "TPA OB Servo"
② Ausgang "TPA OB Servo"
③ Erster OB1-Zyklus
④ Zweiter OB1-Zyklus

Überläufe

Wenn der eingestellte Applikationszyklus nicht eingehalten wird, z. B. durch Hinzufügen weiterer Technologieobjekte oder Programme im MC-PreServo [OB67] bzw. MC-PostServo [OB95], können Überläufe auftreten. Der Applikationszyklus muss in diesem Fall angepasst werden, um Überläufe zu vermeiden.

Bei einem Überlauf des MC-Servo [OB91] im Sendetakt wird eine Meldung im Diagnosepuffer der CPU angezeigt. Der Regler läuft nicht mehr taktsynchron.

Bei einem Überlauf des MC-Servo [OB91] im Applikationszyklus geht die CPU in den Betriebszustand STOP.

Folgendes Bild zeigt das Ablaufverhalten bei einem Überlauf des MC-Servo [OB91] im Applikationszyklus und im Sendetakt bei einer Taktuntersetzung von 2:

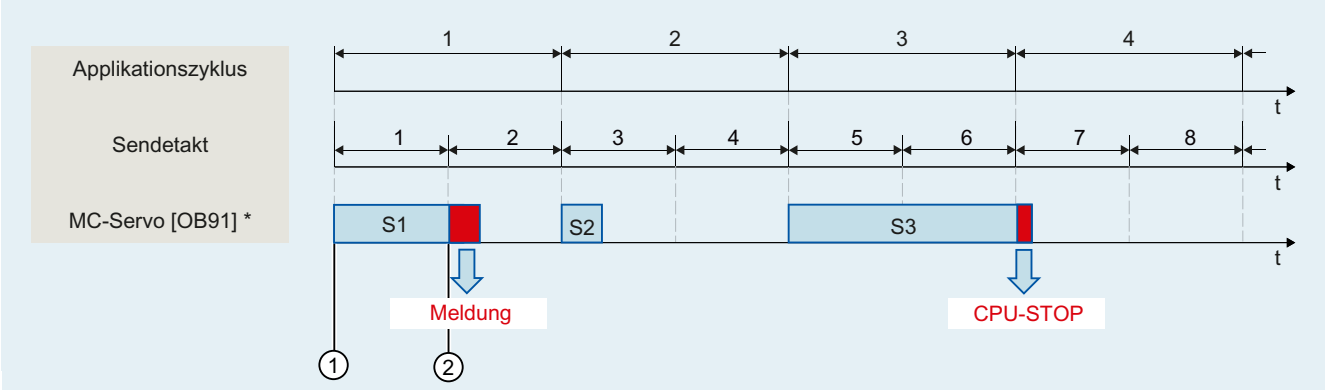

① Eingang "TPA OB Servo"

② Ausgang "TPA OB Servo"

* Inklusive MC-PreServo [OB67] und/oder MC-PostServo [OB95], falls vorhanden

Die Bearbeitung eines MC-Interpolator [OB92] darf maximal durch einen Aufruf des MC-Servo [OB91] unterbrochen werden. Wenn mehr Unterbrechungen auftreten, geht die CPU in den Betriebszustand STOP.

Folgendes Bild zeigt das Ablaufverhalten bei einer Unterbrechung eines MC-Interpolator [OB92] über zwei Zeitscheiben:

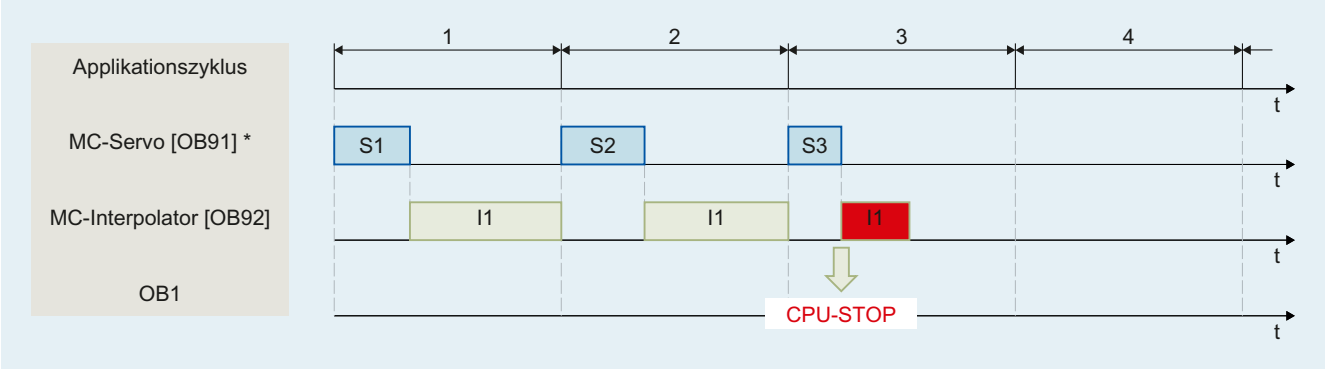

* Inklusive MC-PreServo [OB67] und/oder MC-PostServo [OB95], falls vorhanden

Die CPU toleriert maximal drei aufeinander folgende Überläufe des MC-Interpolator [OB92]. Wenn mehr Überläufe auftreten, geht die CPU in den Betriebszustand STOP.

Folgendes Bild zeigt das Ablaufverhalten bei vier Einzelüberläufen des MC-Interpolator [OB92] hintereinander:

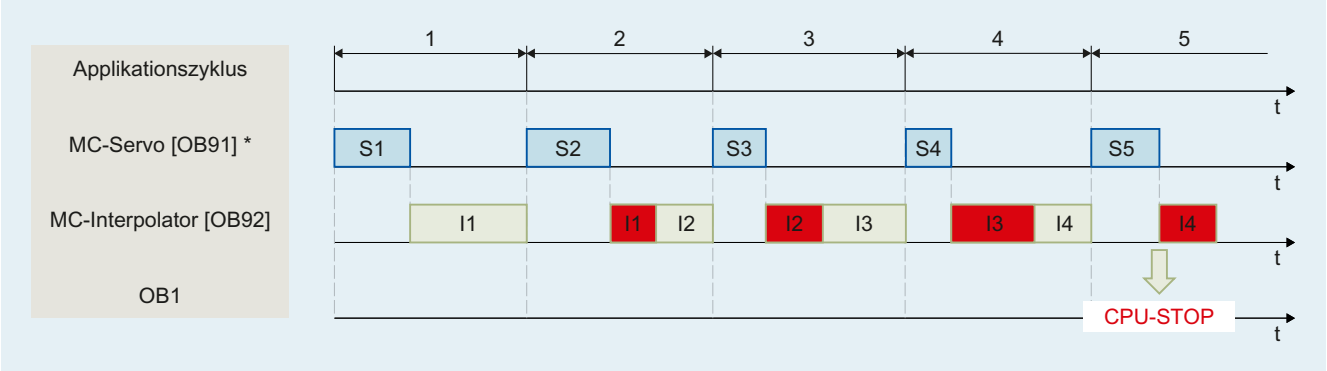

* Inklusive MC-PreServo [OB67] und/oder MC-PostServo [OB95], falls vorhanden

Betriebszustände

In diesem Abschnitt wird das Verhalten von Motion Control in den jeweiligen Betriebszuständen und bei Übergängen zwischen den Betriebszuständen betrachtet. Eine allgemeine Beschreibung der Betriebszustände finden Sie im Systemhandbuch S7-1500.

Betriebszustände und Übergänge

Die CPU hat folgende Betriebszustände:

- STOP
- STARTUP
- RUN
- HALT

Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge:

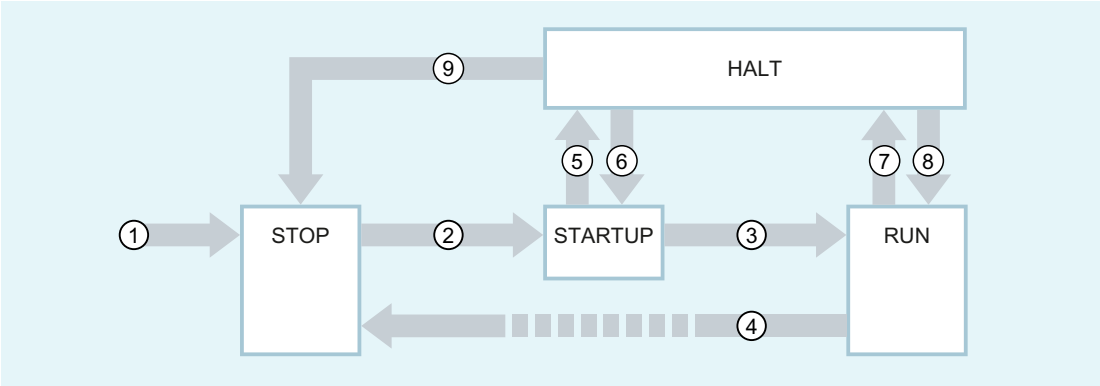

Betriebszustandsübergänge

Die folgende Tabelle zeigt das Verhalten von Motion Control bei den Übergängen zwischen den Betriebszuständen:

Nr.	Betriebszustands-übergang	Verhalten
①	NETZ-EIN → STOP	Die CPU führt einen Restart der Technologieobjekte durch. Die Technologieobjekte werden mit den Werten aus dem Ladespeicher neu initialisiert.
②	STOP → STARTUP	Keine Relevanz für Motion Control.
③	STARTUP → RUN	Die Prozessausgänge werden freigegeben.
④	RUN → STOP	Wenn die CPU vom Betriebszustand RUN nach STOP wechselt, werden alle Technologieobjekte gemäß der Alarmreaktion "Freigabe wegnehmen" gesperrt. Laufende Motion Control-Aufträge werden abgebrochen. Wenn im Betriebszustand RUN an Technologieobjekten Restart-relevante Daten geändert wurden, führt die CPU einen Restart der entsprechenden Technologieobjekte durch.
⑤	STARTUP → HALT	Haltepunkt im Anlaufprogramm erreicht.
⑥	HALT → STARTUP	Nicht möglich bei Verwendung von Technologieobjekten.
⑦	RUN → HALT	Haltepunkt erreicht.
⑧	HALT → RUN	Nicht möglich bei Verwendung von Technologieobjekten.
⑨	HALT → STOP	Bedienung Schalter/Display oder vom Programmiergerät auf STOP gesetzt.

Betriebszustand STOP

Im Betriebszustand STOP wird das Anwenderprogramm nicht bearbeitet und alle Prozessausgänge werden deaktiviert. Somit werden keine Motion Control-Aufträge ausgeführt.

Die Technologie-Datenbausteine werden aktualisiert.

Betriebszustand STARTUP

Bevor die CPU mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, werden die Anlauf-OBs einmalig abgearbeitet.

Im STARTUP sind die Prozessausgänge gesperrt. Motion Control-Aufträge werden abgewiesen.

Die Technologie-Datenbausteine werden aktualisiert.

Betriebszustand RUN

Im Betriebszustand RUN wird das Anwenderprogramm abgearbeitet.

Im RUN werden die programmierten Motion Control-Aufträge zyklisch aufgerufen und abgearbeitet.

Die Technologie-Datenbausteine werden aktualisiert.

Betriebszustand HALT

Das Arbeiten mit Haltepunkten wird bei Verwendung von Technologieobjekten nicht unterstützt. Dabei tritt ein Überlaufen des MC-Servo auf. Dadurch wird sofort in den Betriebszustand STOP gewechselt.

Im Betriebszustand HALT werden keine Ereignisse ausgelöst, das Anwenderprogramm wird nicht ausgeführt.

Alle Ausgänge sind deaktiviert bzw. reagieren wie parametriert. Ausgänge liefern einen parametrierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.

Wenn Sie einen Haltepunkt erreichen, führt die CPU einen impliziten Restart der Technologieobjekte aus. Referenzieren Sie die Technologieobjekte erneut.

Technologieobjekt Drehzahlachse

Das Technologieobjekt Drehzahlachse berechnet unter Berücksichtigung der Dynamikvorgaben Drehzahlsollwerte und gibt sie an den Antrieb aus. Alle Bewegungen der Drehzahlachse finden drehzahlgesteuert statt. Ein vorhandenes Lastgetriebe wird systemseitig berücksichtigt.

Eine Übersicht über die Funktionen des Technologieobjekts Drehzahlachse finden Sie im Kapitel "Funktionen (Seite 1683)".

Jeder Drehzahlachse wird ein Antrieb über ein PROFIdrive-Telegramm oder über eine analoge Sollwertschnittstelle zugeordnet.

Die Drehzahl wird in Umdrehungen pro Zeiteinheit angegeben.

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Drehzahlachse:

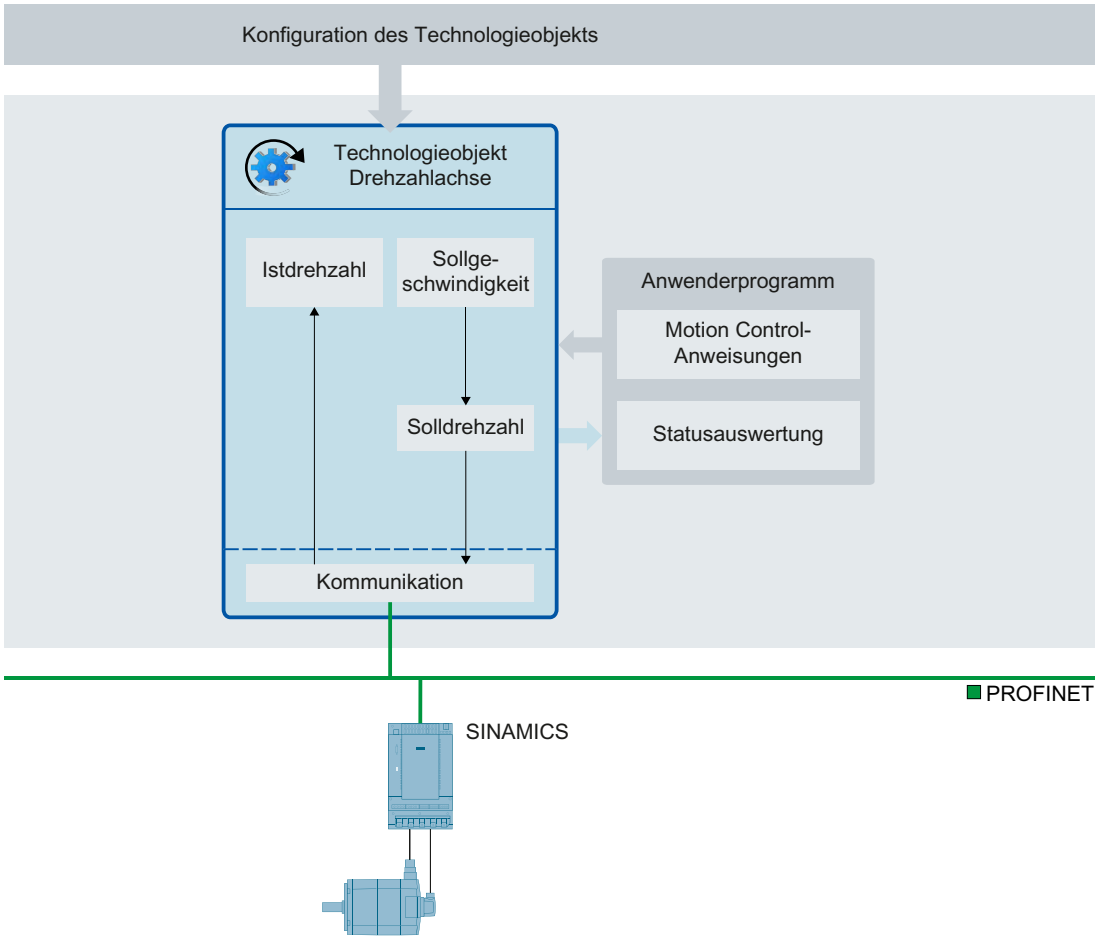

Technologieobjekt Positionierachse

Das Technologieobjekt Positionierachse berechnet unter Berücksichtigung der Dynamikvorgaben Positionssollwerte und gibt entsprechende Drehzahlsollwerte an den Antrieb aus. Im lagegeregelten Betrieb finden alle Bewegungen der Positionierachse lagegeregelt statt. Für das absolute Positionieren muss dem Technologieobjekt Positionierachse die physikalische Position bekannt sein.

Eine Übersicht über die Funktionen des Technologieobjekts Positionierachse finden Sie im Kapitel "Funktionen (Seite 1683)".

Jeder Positionierachse werden ein Antrieb über ein PROFIdrive-Telegramm oder über eine analoge Sollwertschnittstelle und ein Geber über ein PROFIdrive-Telegramm zugeordnet.

Der Bezug der Geberistwerte zu einer definierten Position wird durch die Parametrierung der mechanischen Eigenschaften und Gebereinstellungen sowie einen Referenziervorgang hergestellt. Bewegungen ohne Positionsbezug und relative Positionierbewegungen führt das Technologieobjekt auch im nicht referenzierten Zustand aus.

Je nach Ausführung der Mechanik lässt sich eine Positionierachse als lineare Achse oder rotatorische Achse (Seite 1689) konfigurieren.

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Positionierachse:

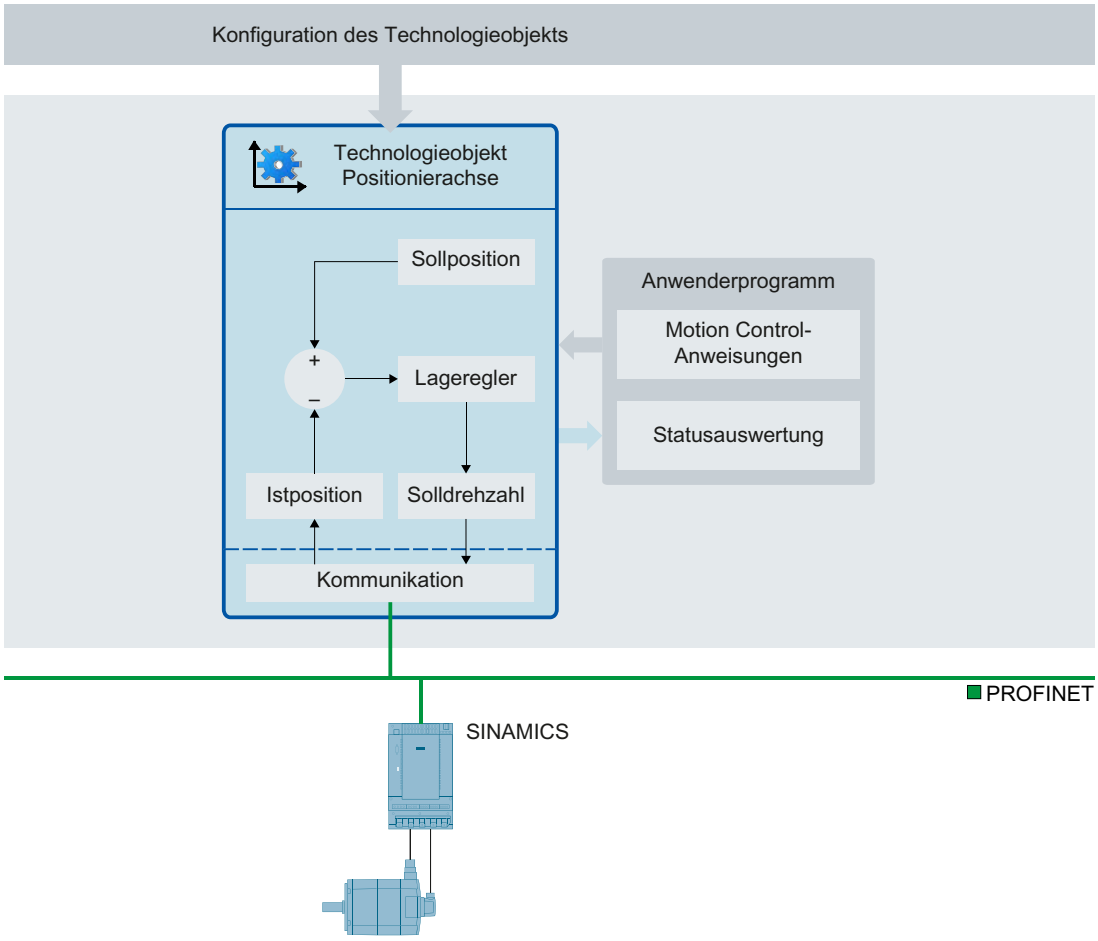

Technologieobjekt Gleichlaufachse

Das Technologieobjekt Gleichlaufachse enthält alle Funktionen des Technologieobjekts Positionierachse.

Zusätzlich kann eine Gleichlaufachse den Bewegungen einer Leitachse folgen. Die Gleichlaufbeziehung zwischen Leit- und Folgeachse wird durch eine Gleichlauffunktion vorgegeben.

Eine Übersicht über die Funktionen des Technologieobjekts Gleichlaufachse finden Sie im Kapitel "Funktionen (Seite 1683)".

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Gleichlaufachse:

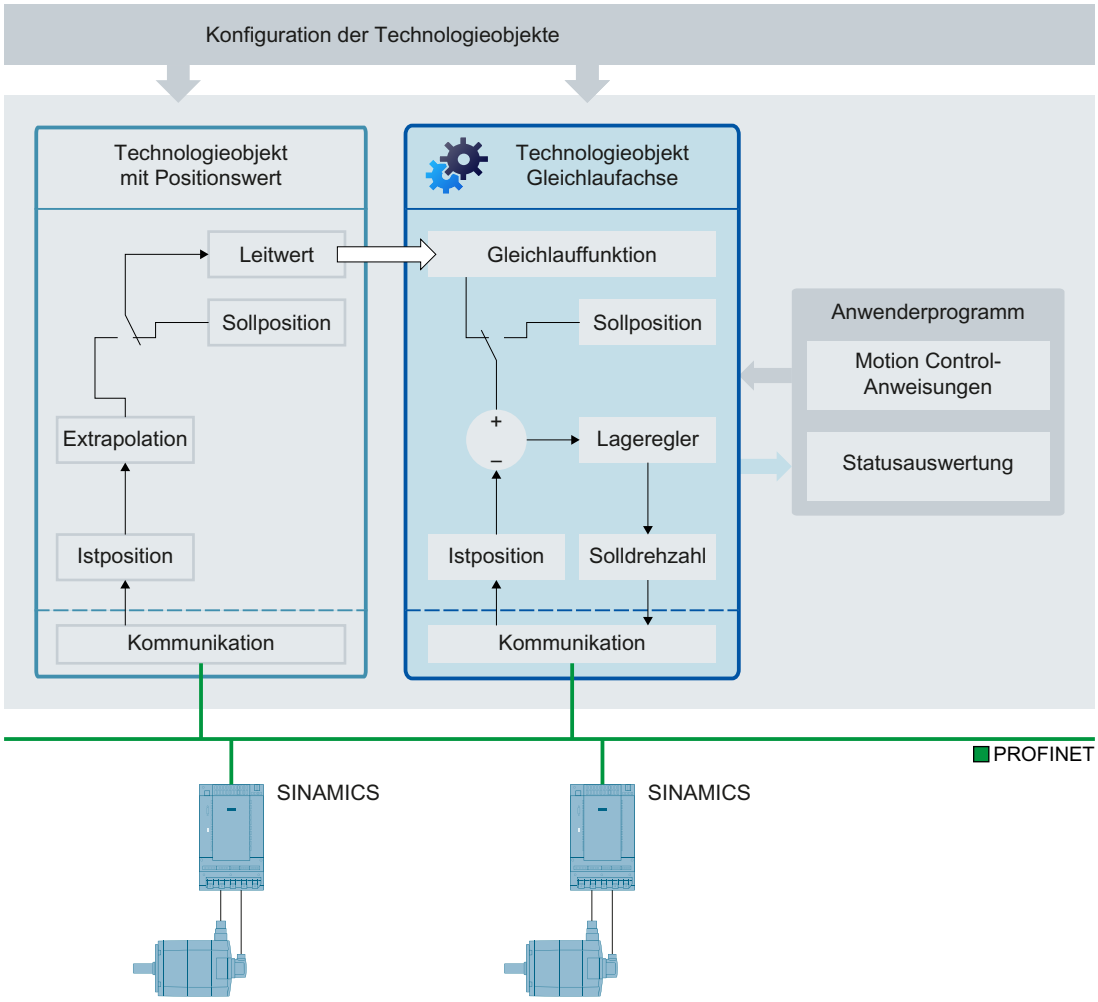

Siehe auch

Achstypen (Seite 1689)

Phasen des Gleichlaufs

Über einen Gleichlauf lässt sich eine Folgeachse an eine Leitachse koppeln und synchron zu dieser verfahren.

Der Gleichlauf verläuft in folgenden Phasen:

- Wartender Gleichlauf (S7-1500T)
  Die Folgeachse wartet auf die Erfüllung der Startbedingungen für die Aufsynchronisierbewegung.
- Aufsynchronisieren
  Die Folgeachse wird auf den Leitwert aufsynchronisiert.

- Synchron fahren
  Die Folgeachse folgt entsprechend der Gleichlauffunktion der Position der Leitachse.
- Gleichlauf ablösen
  Ein aktiver Gleichlauf wird durch Bewegungsaufträge (z. B. "MC_Halt") an die Folgeachse abgelöst.

In den Phasen gelten die Dynamikgrenzen, wie im Kapitel "Dynamikgrenzen im Gleichlauf (Seite 1720)" beschrieben.

Vermeiden Sie während eines aktiven Gleichlaufs das Referenzieren der Leitachse. Das Referenzieren der Leitachse im Gleichlauf entspricht einem Sollwertsprung an der Folgeachse. Die Folgeachse gleicht den Sprung entsprechend der Gleichlauffunktion und nur auf die maximale Drehzahl des Antriebs begrenzt aus.

Hinweis

Die Leitwerte und Folgewerte werden ohne Umrechnung in der jeweils parametrierten Anwendereinheit gekoppelt. Wenn z. B. eine lineare Leitachse um 10 mm verfährt, verfährt eine rotatorische Folgeachse bei einem Getriebefaktor von 1:1 um 10°.

Leitwertkopplung

Der Leitwert für den Gleichlauf wird von einer Leitachse bereitgestellt. Der Leitwert wird im Anwenderprogramm mit dem Aufruf der entsprechenden Motion Control-Anweisung für den Gleichlauf vorgegeben und gekoppelt. Mit dem erneuten Aufruf der Motion Control-Anweisung mit Angabe einer anderen Leitachse wird der Leitwert umgeschaltet.

Für die Leitwertkopplung gelten folgende Regeln:

- Eine Leitachse kann den Leitwert für mehrere Folgeachsen liefern.
- Die Gleichlaufachse lässt sich mit verschiedenen Leitwerten verschalten. Alle im Betrieb benötigten Verschaltungen müssen Sie bei der Konfiguration des Technologieobjekts Gleichlaufachse einrichten.
- Nur jeweils ein Leitwert wird gekoppelt und ausgewertet.

Sollwertkopplung

Bei der Sollwertkopplung wird die Sollposition der Leitachse als Leitwert für den Gleichlauf verwendet.

Die Sollposition folgender Technologieobjekte lässt sich als Leitwert für den Gleichlauf verschalten:

- Positionierachse
- Gleichlaufachse

Istwertkopplung

Für Applikationen, bei denen eine Sollwertkopplung nicht möglich ist, z. B. beim Einsatz eines externen Gebers, bietet die CPU S7-1500T zusätzlich eine Istwertkopplung für den Gleichlauf. Bei der Istwertkopplung wird die extrapolierte Istposition (Seite 1767) eines Technologieobjekts als Leitwert verwendet.

Die Istposition folgender Technologieobjekte lässt sich als Leitwert verwenden:

- Positionierachse
- Gleichlaufachse
- Externer Geber

Extrapolation der Leitwerte bei Istwertkopplung

Bei der Istwertkopplung (Seite 1767) entstehen durch die Verarbeitung der Istwerte Verzugszeiten. Um diese Verzugszeiten zu kompensieren, wird der Istwert leitwertseitig extrapoliert. Das heißt, der Leitwert wird auf Basis der bereits bekannten Werte hochgerechnet.

Mit der Extrapolation lassen sich Istwertabweichungen bei konstanter Geschwindigkeit oder bei konstanter Beschleunigung/Verzögerung kompensieren. Veränderungen von Beschleunigung/Verzögerung (Ruck) während der Extrapolation haben technisch bedingt immer einen Versatz der Folgeachse zum Leitwert zur Folge.

Die Extrapolationszeit ergibt sich aus einem leitachsbedingten und einem vom Anwender vorgegebenen folgeachsbedingten Anteil:

- Leitachsbedingter Anteil
  Der leitachsbedingte Anteil wird automatisch berechnet und an der Leitachse in der Variable "<TO>.Extrapolation.LeadingAxisDependentTime" des Technologieobjekts angezeigt.
- Folgeachsbedingter Anteil
  Der folgeachsbedingte Anteil wird automatisch berechnet und an der Folgeachse in der Variable "<TO>.StatusPositioning.SetpointExecutionTime" des Technologieobjekts angezeigt. Den Wert dieser Variable geben Sie (unverändert oder mit anwenderspezifischen Laufzeiten verrechnet) in der Konfiguration der Leitachse an (<TO>.Extrapolation.FollowingAxisDependentTime).

Der extrapolierte Istwert wird vor der Ausgabe als Leitwert mit einer konfigurierbaren Hysterese bewertet. Mit der Hysteresebewertung wird eine Umkehr des Leitwerts verhindert, die sich aus der Extrapolation eines verrauschten Werts ergeben kann.

Folgendes Bild zeigt den Ablauf der Istwertextrapolation:

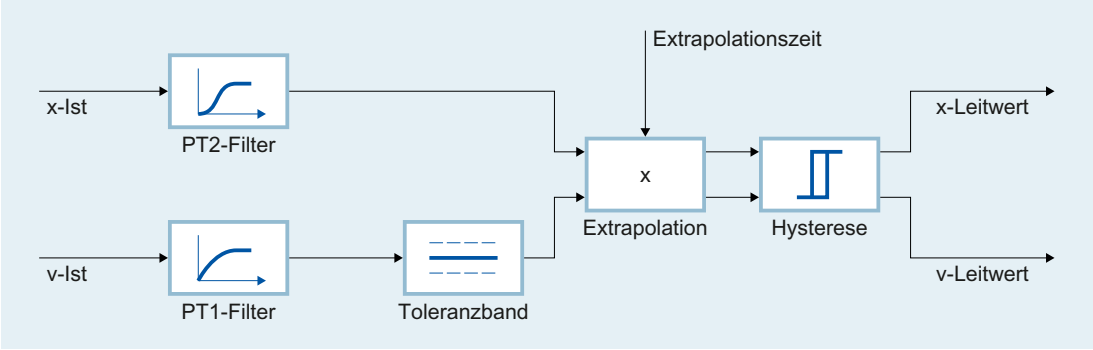

ACHTUNG
Maschinenschaden Wenn Sie die Extrapolationszeit während der Laufzeit Ihres Anwenderprogramms in zu großen Schritten ändern, können Schäden an der Maschine auftreten. Ändern Sie die Extrapolationszeit nur um einen geringen Betrag.

Filtern der Istwerte

Verrauschte Gebersignale führen zu hohen Geschwindigkeitssprüngen, die sich auch auf die Extrapolation auswirken. Durch geeignete Filtereinstellungen lassen sich diese Sprünge reduzieren oder kompensieren. Der Istpositionsfilter ist ein PT2-Filter. Der Geschwindigkeitsfilter ist ein PT1-Filter mit konfigurierbarer Toleranzbandbreite.

Der Istwert wird zuerst durch den Istpositionsfilter geglättet. Die Geschwindigkeit wird durch den Geschwindigkeitsfilter geglättet und durch das Toleranzband weiter "beruhigt". Die gefilterte Istposition wird dann unter Berücksichtigung der gefilterten Geschwindigkeit extrapoliert.

Einstellungsempfehlung:

Stellen Sie die Summe der Zeitkonstanten T1 und T2 des Positionsfilters deutlich kleiner ein, als die Zeitkonstante T1 des Geschwindigkeitsfilters.

Toleranzband

Das Toleranzband wirkt auf den gefilterten Geschwindigkeitswert im Interpolationstakt. Die Lage des Toleranzbands wird automatisch in die Richtung des Geschwindigkeitswerts verschoben, sobald sich dieser um mehr als die Hälfte des Toleranzbands vom zuletzt ausgegebenen Ausgangswert in eine Richtung verändert. Mit der Verschiebung des Toleranzbands wird gleichzeitig ein neuer Ausgangswert gebildet. Dieser entspricht dem gefilterten Geschwindigkeitswert abzüglich der Hälfte des Toleranzbands. Solange der Geschwindigkeitswert innerhalb des Toleranzbands verbleibt, wird kein neuer Ausgangswert gebildet.

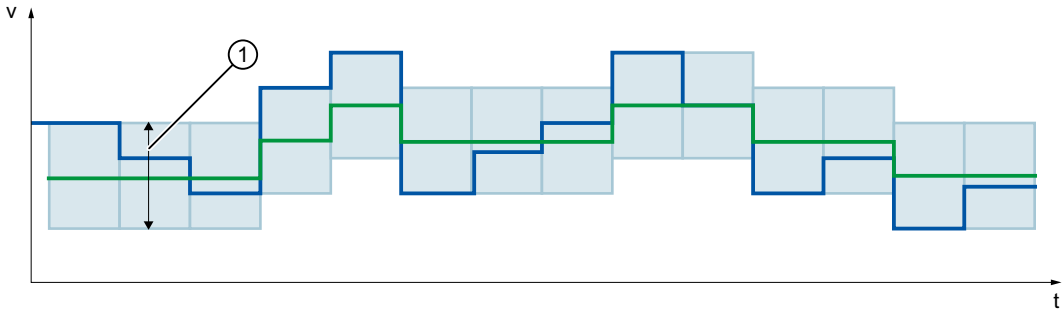

① Toleranzband

Extrapolationsgeschwindigkeit vor Toleranzband

Extrapolationsgeschwindigkeit nach Toleranzband

Hysterese

Die Hysterese wirkt auf den gefilterten extrapolierten Positionswert im Interpolationstakt. Ein Richtungswechsel wird erst wirksam, wenn sich der Positionswert mindestens um den Hysteresewert in Gegenrichtung verändert hat.

Variablen

Folgende Variablen des Technologieobjekts sind für die Istwertextrapolation relevant:

Konfiguration	
Variable	**Beschreibung**
<TO>.Extrapolation.LeadingAxisDependent-Time	(An der Leitachse) Leitachsbedingter Anteil der Extrapolationszeit, die sich aus T_i, T_{ipo} und T_{Filter} ergibt.
<TO>.Extrapolation.FollowingAxisDependent-Time	(An der Leitachse) Folgeachsbedingter Anteil der Extrapolationszeit Geben Sie den Wert aus der Variable "<TO>.StatusPositioning.SetpointExecutionTime" der Folgeachse ein (unverändert oder mit anwenderspezifischen Zeiten verrechnet).
<TO>.StatusPositioning.SetpointExecutionTime	(An der Folgeachse) Sollwert-Ausführungszeit, die sich aus T_{Ipo}, T_{vtc} bzw. 1/kv, T_{Send} und T_O der Achse ergibt.
<TO>.Extrapolation.PositionFilter.T1	Positionsfilter Zeitkonstante T1

Konfiguration	
Variable	**Beschreibung**
<TO>.Extrapolation.PositionFilter.T2	Positionsfilter Zeitkonstante T2
<TO>.Extrapolation.VelocityFilter.T1	Geschwindigkeitsfilter Zeitkonstante T1
<TO>.Extrapolation.VelocityTolerance.Range	Toleranzbandbreite für die Geschwindigkeit
<TO>.Extrapolation.Hysteresis.Value	Hysteresewert (in der konfigurierten Längeneinheit)

Getriebegleichlauf mit "MC_GearIn"

Beim Getriebegleichlauf ergibt sich die Position der Folgeachse aus der Position der Leitachse multipliziert mit dem Getriebefaktor. Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen vor. Dadurch ergibt sich eine lineare Gleichlauffunktion.

Der Gleichlauf mit "MC_GearIn" beginnt nach dem Aufsynchronisieren, wenn die Folgeachse unter Berücksichtigung des Getriebefaktors die Geschwindigkeit und Beschleunigung der Leitachse erreicht hat.

Aufsynchronisieren

Das Aufsynchronisieren beginnt mit dem Start eines "MC_GearIn"-Auftrags. Wenn eine Gleichlaufachse auf einen Leitwert aufsynchronisiert, wird dies in der Variable "<TO>.StatusWord.X21 (Synchronizing)" des Technologieobjekts angezeigt. Laufende Bewegungsaufträge werden abgelöst. Die Dynamikwerte (Beschleunigung, Verzögerung, Ruck) der Folgeachse für das Aufsynchronisieren werden am "MC_GearIn" angegeben.

Dauer und Wegstrecke des Aufsynchronisierens sind von folgenden Parametern abhängig:

- Startzeitpunkt des "MC_GearIn"-Auftrags
- Dynamik der Folgeachse zum Startzeitpunkt
- Dynamikvorgaben am "MC_GearIn"
- Dynamik der Leitachse

Wenn die Folgeachse unter Berücksichtigung des Getriebefaktors die Geschwindigkeit und die Beschleunigung der Leitachse erreicht hat, ist die Folgeachse aufsynchronisiert. Die Folgeachse fährt synchron zur Leitachse.

Synchron fahren

Wenn eine Gleichlaufachse auf einen Leitwert aufsynchronisiert ist, wird der Status "synchron" durch den Parameter "MC_GearIn.InGear" = TRUE sowie in der Variable "<TO>.StatusWord.X22 (Synchronous)" des Technologieobjekts angezeigt. Die Folgeachse folgt entsprechend dem Getriebefaktor der Dynamik der Leitachse.

Das Übertragungsverhalten beim Getriebegleichlauf wird durch eine lineare Beziehung zwischen Leitwert und Folgewert ausgedrückt.

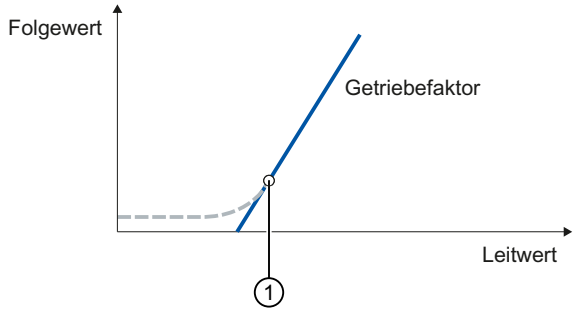

Getriebefaktor	Steigung der Geraden/Übersetzungsverhältnis Getriebefaktor = "MC_GearIn.RatioNumerator"/"MC_GearIn.RatioDenominator"
	Aufsynchronisieren
①	Position, ab der Leit- und Folgeachse synchron verfahren Position ① ist von den oben genannten Parametern abhängig (siehe Aufsynchronisieren).

Der Folgewert wird nach folgender Funktion berechnet:

Position der Folgeachse (Folgewert) = Position ① der Folgeachse + Getriebefaktor × (Position der Leitachse - Position ① der Leitachse)

Richtung

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- Positiver Getriebefaktor
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- Negativer Getriebefaktor
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Siehe auch

Variablen für Gleichlauf (Seite 1779)

Dynamikgrenzen im Gleichlauf (Seite 1720)

Getriebegleichlauf mit "MC_GearInPos" mit vorgegebenen Synchronpositionen

Beim Getriebegleichlauf ergibt sich die Position der Folgeachse aus der Position der Leitachse multipliziert mit dem Getriebefaktor. Den Getriebefaktor geben Sie als Verhältnis zweier ganzer Zahlen vor. Dadurch ergibt sich eine lineare Gleichlauffunktion.

Der Gleichlauf mit "MC_GearInPos" beginnt nach dem Aufsynchronisieren mit dem Erreichen der vorgegebenen Synchronpositionen.

Aufsynchronisieren

Die Synchronpositionen der Leit- und Folgeachse, ab denen die Achsen synchron verfahren sollen, geben Sie an der Motion Control-Anweisung "MC_GearInPos" an. Der Gleichlauf wird vorlaufend zu den angegebenen Synchronpositionen für die Leit- und Folgeachse aufsynchronisiert.

Das vorlaufende Aufsynchronisieren (Seite 1777) erfolgt wahlweise über die Dynamikparameter (Geschwindigkeit, Beschleunigung, Verzögerung, Ruck) oder über den Leitwertweg. Die Dynamikparameter und der Leitwertweg werden am "MC_GearInPos" angegeben und beeinflussen die Startposition für das Aufsynchronisieren. Nach dem Start eines "MC_GearInPos"-Auftrags wird so aufsynchronisiert, dass Leit- und Folgeachse beim Erreichen der Synchronpositionen synchron verfahren.

Das Aufsynchronisieren wird durch den Parameter "MC_GearInPos.StartSync" = TRUE sowie in der Variable "<TO>.StatusWord.X21 (Synchronizing)" des Technologieobjekts angezeigt.

Beim Aufsynchronisieren darf der Leitwert nicht reversieren.

Synchron fahren

Der Gleichlauf ist ab den angegebenen Synchronpositionen synchron. Der Status "synchron" wird durch den Parameter "MC_GearInPos.InSync" = TRUE sowie in der Variable "<TO>.StatusWord.X22 (Synchronous)" des Technologieobjekts angezeigt. Die Folgeachse folgt entsprechend dem Getriebefaktor der Position der Leitachse.

Das Übertragungsverhalten beim Getriebegleichlauf wird durch eine lineare Beziehung zwischen Leitwert und Folgewert ausgedrückt.

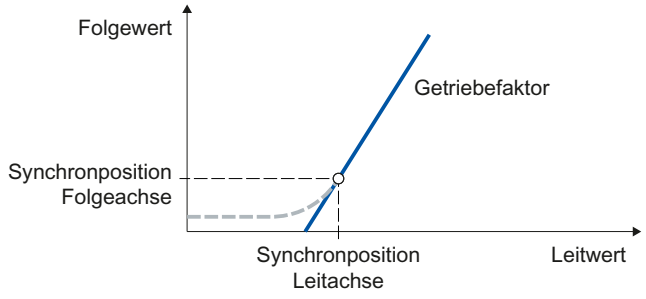

Getriebefaktor	Steigung der Geraden/Übersetzungsverhältnis Getriebefaktor = "MC_GearInPos.RatioNumerator"/"MC_GearInPos.RatioDenominator"
- - - -	Aufsynchronisieren

Der Folgewert wird nach folgender Funktion berechnet:

Position der Folgeachse (Folgewert) = Synchronposition der Folgeachse + Getriebefaktor × (Position der Leitachse - Synchronposition der Leitachse)

Richtung

Der Zähler des Getriebefaktors wird positiv oder negativ angegeben. Dadurch ergibt sich folgendes Verhalten:

- Positiver Getriebefaktor
  Die Leitachse und die Folgeachse bewegen sich in die gleiche Richtung.
- Negativer Getriebefaktor
  Die Folgeachse bewegt sich in entgegengesetzter Richtung zur Leitachse.

Siehe auch

Dynamikgrenzen im Gleichlauf (Seite 1720)

Variablen für Getriebegleichlauf mit "MC_GearInPos"

Folgende Variablen des Technologieobjekts sind für den Getriebegleichlauf mit "MC_GearInPos" relevant:

Statusanzeigen		
Variable	**Beschreibung**	
<TO>.StatusSynchronizedMotion.FunctionState	Anzeige, welche Gleichlauffunktion aktiv ist	
	0	Kein Gleichlauf aktiv
	1	Getriebegleichlauf ("MC_GearIn")
	2	Getriebegleichlauf mit vorgegebenen Synchronpositionen ("MC_GearInPos")
	3	Kurvenscheibengleichlauf ("MC_CamIn")
<TO>.StatusSynchronizedMotion.PhaseShift	Aktuelle absolute Leitwertverschiebung bei Getriebegleichlauf	

Siehe auch

Variablen für Gleichlauf (Seite 1779)

Leitwertverschiebung beim Getriebegleichlauf

Über die Motion Control-Anweisungen "MC_PhasingRelative" und "MC_PhasingAbsolute" lässt sich beim Getriebegleichlauf über "MC_GearIn" und "MC_GearInPos" der Leitwert in der Gleichlauffunktion verschieben. Die Leitwertverschiebung wird an der Folgeachse ausgeführt. Die Leitachse wird nicht beeinflusst.

Die Leitwertverschiebung ist nur im Status "synchron" wirksam. Wenn der Gleichlauf abgelöst wird, wird die Leitwertverschiebung auf null zurückgesetzt.

Kurvenscheibengleichlauf

Beim Kurvenscheibengleichlauf sind Leit- und Folgeachse durch eine Gleichlauffunktion gekoppelt, die Sie über eine Kurvenscheibe (Seite 1821) vorgeben.

Der Gleichlauf beginnt nach dem Aufsynchronisieren mit dem Erreichen der vorgegebenen Synchronposition der Leitachse. Die entsprechende Synchronposition der Folgeachse ergibt sich durch die Kurvenscheibe.

Die verwendete Kurvenscheibe lässt sich auftragsbezogen skaliert und verschoben anwenden. Um die Kurvenscheibe für den Kurvenscheibengleichlauf zu verwenden, muss sie interpoliert sein. Die Kurvenscheibe interpolieren Sie in Ihrem Anwenderprogramm mit der Motion Control-Anweisung "MC_InterpolateCam".

Vorlaufendes Aufsynchronisieren

Die Synchronposition der Leitachse, zu der die Achsen synchron verfahren sollen, geben Sie an der Motion Control-Anweisung "MC_CamIn" an. Der Gleichlauf wird vorlaufend zu der angegebenen Synchronposition aufsynchronisiert.

Die Synchronposition der Leitachse ergibt sich durch folgende Parameter:

- Anfangsposition der Kurvenscheibe (<TO_Cam>.StatusCam.StartLeadingValue)
- "MC_CamIn.MasterScaling" - Skalierung der Leitwerte der Kurvenscheibe
- "MC_CamIn.MasterOffset" - Verschiebung/Lage der Kurvenscheibe
- "MC_CamIn.MasterSyncPosition" - Startpunkt innerhalb der Kurvenscheibe

Die Synchronposition berechnet sich mit folgender Gleichung:

Synchronposition = (Anfangsposition der Kurvenscheibe × "MasterScaling") + "MasterOffset" + "MasterSyncPosition"

Das vorlaufende Aufsynchronisieren (Seite 1777) erfolgt wahlweise über die Dynamikparameter (Geschwindigkeit, Beschleunigung, Verzögerung, Ruck) oder den Leitwertweg. Die Dynamikparameter und der Leitwertweg werden am "MC_CamIn" angegeben und beeinflussen die Startposition für das Aufsynchronisieren. Nach dem Start eines "MC_CamIn"-Auftrags wird so aufsynchronisiert, dass Leit- und Folgeachse beim Erreichen der Synchronposition synchron verfahren.

Das Aufsynchronisieren wird durch den Parameter "MC_CamIn.StartSync" = TRUE sowie in der Variable "<TO>.StatusWord.X21 (Synchronizing)" des Technologieobjekts angezeigt.

Beim Aufsynchronisieren darf der Leitwert nicht reversieren.

Direktes Synchronisieren

Wenn Sie den Parameter "MC_CamIn.SyncProfileReference" = 2 setzen, wird unmittelbar an der aktuellen Leitwertposition und an der aktuellen Folgewertposition der Status synchron gesetzt.

Synchron fahren

Der Gleichlauf ist erreicht, wenn die Folgeachse die Synchronposition und entsprechend der Gleichlauffunktion die Dynamik der Leitachse erreicht hat.

Wenn eine Gleichlaufachse auf einen Leitwert aufsynchronisiert ist, wird der Status "synchron" durch den Parameter "MC_CamIn.InSync" = TRUE sowie in der Variable "<TO>.StatusWord.X22 (Synchronous)" des Technologieobjekts angezeigt. Die Folgeachse folgt entsprechend der konfigurierten Gleichlauffunktion der Position der Leitachse.

Das Übertragungsverhalten beim Kurvenscheibengleichlauf wird durch den Verlauf der Kurvenscheibe ausgedrückt:

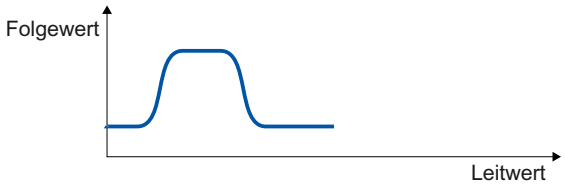

Folgewert = f(Leitwert)

Siehe auch

Dynamikgrenzen im Gleichlauf (Seite 1720)

Skalierung und Verschiebung der Kurvenscheibe

Die Skalierung und die Verschiebung der Kurvenscheibe lassen sich beim Kurvenscheibengleichlauf an der Motion Control-Anweisung "MC_CamIn" angeben. Die konfigurierte Kurvenscheibe wird durch den Aufruf von "MC_CamIn" nicht verändert.

Das folgende Bild zeigt den prinzipiellen Ablauf der Skalierung/Verschiebung der Kurvenscheibe:

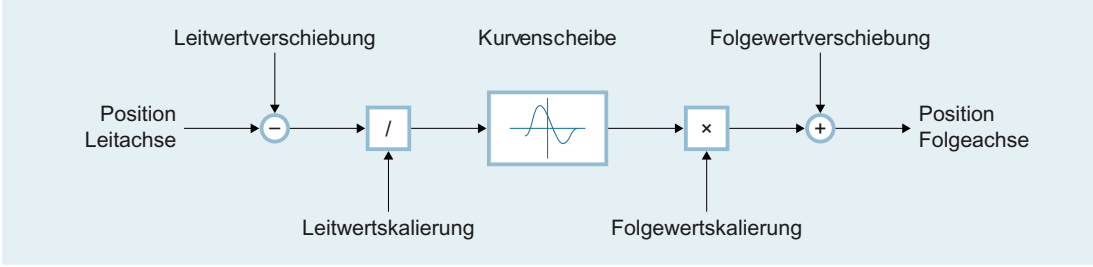

Position Folgeachse = f[(Position Leitachse - Leitwertverschiebung) / Leitwertskalierung] × Folgewertskalierung + Folgewertverschiebung

Zyklische und nicht zyklische Anwendung der Kurvenscheibe

Über den Parameter "MC_CamIn.ApplicationMode" lässt sich einstellen, ob die Kurvenscheibe zyklisch oder nicht zyklisch für den Gleichlauf angewendet wird:

- **Nicht zyklisch**
  Die Kurvenscheibe wird genau einmal durchlaufen. Beim Durchlaufen in positiver Richtung wird der Gleichlauf beim Erreichen des Endpunkts der Kurvenscheibe beendet. Beim Durchlaufen in negativer Richtung wird der Gleichlauf beim Erreichen des Anfangspunkts der Kurvenscheibe beendet. Zur Vermeidung von Dynamiksprüngen muss die Geschwindigkeit der Folgeachse im Anfangs- und Endpunkt der Kurvenscheibe null sein.

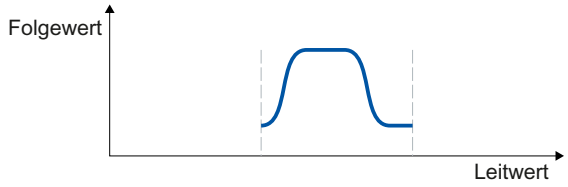

- **Zyklisch**
  Die Kurvenscheibe wird zyklisch durchlaufen. Beim Durchlaufen in positiver Richtung wird die Kurvenscheibe beim Erreichen des Endpunkts der Kurvenscheibe ab dem Startpunkt wiederholt. Beim Durchlaufen in negativer Richtung wird die Kurvenscheibe beim Erreichen des Anfangspunkts der Kurvenscheibe ab dem Endpunkt wiederholt. Zur Vermeidung von Dynamiksprüngen müssen Start- und Endpunkt der Kurvenscheibe übereinstimmen.

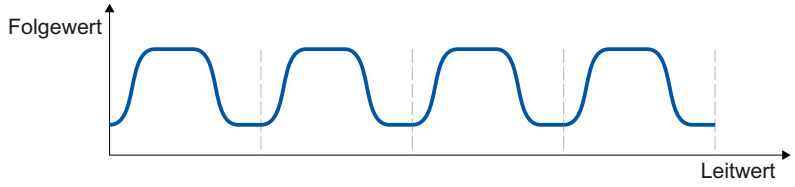

- **Zyklisch anhängend**
  Die Kurvenscheibe wird zyklisch durchlaufen. Beim Durchlaufen in positiver Richtung wird der Endpunkt der Kurvenscheibe als Startpunkt für den nächsten Durchlauf verwendet. Beim Durchlaufen in negativer Richtung wird der Startpunkt der Kurvenscheibe als Startpunkt für den nächsten Durchlauf verwendet. Der folgewertseitige Positionsunterschied zwischen Start- und Endpunkt wird richtungsabhängig aufaddiert. Zur Vermeidung von Dynamiksprüngen muss die Geschwindigkeit in den Randpunkten stetig sein.

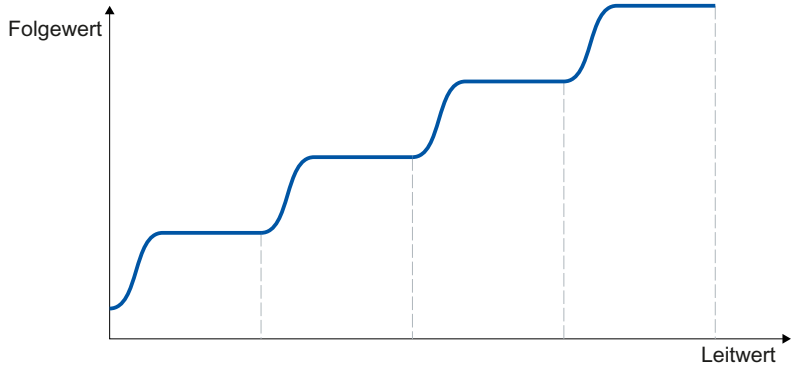

Variablen für Kurvenscheibengleichlauf

Folgende Variablen des Technologieobjekts sind für den Kurvenscheibengleichlauf relevant:

Statusanzeigen		
Variable	**Beschreibung**	
<TO>.StatusSynchronizedMotion.FunctionState	Anzeige, welche Gleichlauffunktion aktiv ist	
	0	Kein Gleichlauf aktiv
	1	Getriebegleichlauf ("MC_GearIn")
	2	Getriebegleichlauf mit vorgegebenen Synchronpositionen ("MC_GearInPos")
	3	Kurvenscheibengleichlauf ("MC_CamIn")
<TO>.StatusSynchronizedMotion.CurrentCam	Aktuell für den Kurvenscheibengleichlauf verwendete Kurvenscheibe	
<TO>.StatusSynchronizedMotion.MasterOffset	Aktuelle Verschiebung des Leitwertbereichs der Kurvenscheibe	
<TO>.StatusSynchronizedMotion.MasterScaling	Aktuelle Skalierung des Leitwertbereichs der Kurvenscheibe	
<TO>.StatusSynchronizedMotion.SlaveOffset	Aktuelle Verschiebung des Folgewertbereichs der Kurvenscheibe	
<TO>.StatusSynchronizedMotion.SlaveScaling	Aktuelle Skalierung des Folgewertbereichs der Kurvenscheibe	

Siehe auch

Variablen für Gleichlauf (Seite 1779)

Vorlaufendes Aufsynchronisieren mit "MC_GearInPos"/"MC_CamIn"

Aufsynchronisieren über Dynamikparameter

Beim Aufsynchronisieren über Dynamikparameter geben Sie die Dynamik vor (Geschwindigkeit, Beschleunigung, Verzögerung, Ruck). Die benötigte Wegstrecke (Aufsynchronisierlänge) wird berechnet.

Nach dem Start des entsprechenden Motion Control-Auftrags wird fortlaufend ein Bewegungsprofil für die Folgeachse berechnet. Das Bewegungsprofil wird abhängig von folgenden Parametern berechnet:

- Vorgegebene Synchronpositionen an der Motion Control-Anweisung
- Vorgegebene Dynamik an der Motion Control-Anweisung
- Aktuelle Position und Dynamik der Leit- und Folgeachse
- Gleichlauffunktion

Aus der Berechnung ergibt sich eine Aufsynchronisierlänge.

Die Berechnung wird beendet und die Folgeachse zur vorgegebenen Synchronposition verfahren, wenn folgende Position erreicht wurde:

Startposition = Synchronposition der Leitachse - Aufsynchronisierlänge

Der Start des Aufsynchronisierens wird an der Motion Control-Anweisung mit dem Parameter "StartSync" = TRUE angezeigt.

Die Dynamik der Folgeachse beim Aufsynchronisieren ergibt sich aus dem berechneten Bewegungsprofil und aus der aktuellen Dynamik der Leitachse. Änderungen der Dynamik der Leitachse beim Aufsynchronisieren werden dem berechneten Bewegungsprofil überlagert. Durch die Überlagerung werden gegebenenfalls die Dynamikvorgaben an der Folgeachse überhöht.

Wenn die Folgeachse die Synchronposition erreicht hat, ist die Folgeachse aufsynchronisiert. Die Folgeachse fährt synchron zur Leitachse. Der Status "synchron" wird an der Motion Control-Anweisung mit dem Parameter "InSync" = TRUE angezeigt.

Wenn beim Start des entsprechenden Motion Control-Auftrags nur die Leitachse auf ihrer Synchronposition steht, muss die Leitachse erst die Startposition überfahren, um das Aufsynchronisieren zu starten.

Aufsynchronisieren über Leitwertweg

Beim Aufsynchronisieren über den Leitwertweg geben Sie die Aufsynchronisierlänge vor.

Nach dem Start des entsprechenden Motion Control-Auftrags wird abhängig vom vorgegebenen Leitwertweg ein Bewegungsprofil mit der benötigten Dynamik für die Folgeachse berechnet. Dazu muss die Leitachse mindestens um den Leitwertweg von der Synchronposition entfernt sein.

Die Folgeachse wird zur vorgegebenen Synchronposition verfahren, sobald der Leitwert folgende Position erreicht:

Startposition = Synchronposition der Leitachse - Aufsynchronisierlänge

Der Start des Aufsynchronisierens wird an der Motion Control-Anweisung mit dem Parameter "StartSync" = TRUE angezeigt.

Die Dynamik der Folgeachse beim Aufsynchronisieren ergibt sich aus dem berechneten Bewegungsprofil und aus der aktuellen Dynamik der Leitachse. Änderungen der Dynamik der Leitachse beim Aufsynchronisieren werden dem berechneten Bewegungsprofil entsprechend der Gleichlauffunktion überlagert. Durch die Überlagerung werden gegebenenfalls die Dynamikvorgaben an der Folgeachse überhöht.

Wenn die Folgeachse die Synchronposition erreicht hat, ist die Folgeachse aufsynchronisiert. Die Folgeachse fährt synchron zur Leitachse. Der Status "synchron" wird an der Motion Control-Anweisung mit dem Parameter "InSync" = TRUE angezeigt.

Direkt synchron setzen mit "MC_CamIn"

Diese Einstellung der Art des Aufsynchronisierens am "MC_CamIn" ist vor allem für das Synchronisieren im Stillstand geeignet.

Nach dem Start des entsprechenden Motion Control-Auftrags wird unmittelbar an der aktuellen Leitwertposition und an der aktuellen Folgewertposition der Status synchron gesetzt ("InSync" = TRUE).

Der über den Parameter "MasterSyncPosition" angegebene Punkt in der Kurvenscheibe wird im Leitwertbereich der Sollposition der Leitachse und im Folgewertbereich der Sollposition der Folgeachse zugeordnet. Die Kurvenscheibe wird entsprechend verschoben. Die aktuelle Verschiebung ergibt sich aus der Kurvenscheibe und wird an den Variablen "<TO>.StatusSynchronizedMotion.MasterOffset" und "<TO>.StatusSynchronizedMotion.SlaveOffset" des Technologieobjekts angezeigt.

Gleichlauf in Simulation setzen

Beim Wegnehmen der Achsfreigaben oder bei Bewegungsaufträgen an einer Folgeachse wird eine aktive Gleichlaufverbindung gelöst. Durch Setzen des Gleichlaufs in Simulation halten Sie den Gleichlauf aktiv, ohne den Gleichlauf abzulösen.

Mit der Motion Control-Anweisung "MC_SynchronizedMotionSimulation" können Sie einen aktiven Gleichlauf in Simulation setzen. Die Leitachse soll zu diesem Zeitpunkt stehen.

Die Gleichlaufsimulation wirkt nur auf den Gleichlauf der Folgeachse. Aus dem Gleichlauf werden keine Sollwerte an den Antrieb ausgegeben. Aus gegebenenfalls aktiv überlagerten Bewegungen der Folgeachse erfolgt weiterhin die Sollwertausgabe an den Antrieb. Gleiches gilt bei Einzelachsbefehlen während der Gleichlaufsimulation.

Starten Sie einen "MC_SynchronizedMotionSimulation"-Auftrag nur, wenn sich die Folgeachse im Gleichlauf befindet. Dann ist das Statusbit "Synchron" aktiv, "<TO>.StatusWord.X22" = TRUE. Ist die Folgeachse nicht oder noch nicht im Gleichlauf, wird die Anweisung mit Fehler abgebrochen.

Der Gleichlauf bleibt in Simulation aktiv, auch bei Bewegungen durch Einzelachsbefehle oder Sperren der Leit- und/oder Folgeachse, z. B. durch Öffnen einer Schutztüre. Die Folgeachse muss nach dem Beenden der Gleichlaufsimulation nicht neu aufsynchronisieren.

Um zu vermeiden, dass der Sollwert der Folgeachse beim Beenden der Simulation springt, beachten Sie folgende Punkte:

- Leit- und Folgeachse müssen beim Beenden der Simulation jeweils auf den Ausgangspositionen stehen, die sie beim Einschalten der Simulation hatten.
- Wenn die Leitachse an einer anderen Position steht, als beim Einschalten der Simulation, muss die Folgeachse an der Position stehen, die sich aus der Gleichlaufbeziehung ergibt. Diese Position der Folgeachse können Sie vorher im Anwenderprogramm berechnen und über einen Bewegungsauftrag anfahren.

Variablen für Gleichlauf

Folgende Variablen des Technologieobjekts sind für den Gleichlauf relevant:

Statusanzeigen		
Variable	**Beschreibung**	
<TO>.StatusSynchronizedMotion.ActualMaster	Beim Start eines Gleichlaufauftrags wird die Nummer des Technologie-Datenbausteins der aktuell verwendeten Leitachse angezeigt.	
	0	Gleichlauf inaktiv
<TO>.StatusSynchronizedMotion.PhaseShift	Aktuelle absolute Leitwertverschiebung	
<TO>.StatusWord.X21 (Synchronizing)	Wird auf den Wert "TRUE" gesetzt, wenn die Gleichlaufachse auf einen Leitwert aufsynchronisiert.	
<TO>.StatusWord.X22 (Synchronous)	Wird auf den Wert "TRUE" gesetzt, wenn die Gleichlaufachse aufsynchronisiert ist und synchron zur Leitachse fährt.	
<TO>.ErrorWord.X14 (SynchronousError)	Fehler beim Gleichlauf Die an der Motion Control-Anweisung angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert.	
<TO>.StatusSynchronizedMotion.Status-Word.X0 (MaxVelocityExceeded)	Wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Geschwindigkeit überschritten wird.	

Statusanzeigen	
Variable	**Beschreibung**
<TO>.StatusSynchronizedMotion.StatusWord.X1 (MaxAccelerationExceeded)	Wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Beschleunigung überschritten wird.
<TO>.StatusSynchronizedMotion.StatusWord.X2 (MaxDecelerationExceeded)	Wird auf den Wert "TRUE" gesetzt, wenn im Gleichlauf die an der Folgeachse konfigurierte maximale Verzögerung überschritten wird.

Siehe auch

Variablen für Kurvenscheibengleichlauf (Seite 1777)

Variablen für Getriebegleichlauf mit "MC_GearInPos" (Seite 1773)

Technologieobjekt Externer Geber

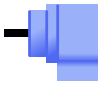

Das Technologieobjekt Externer Geber erfasst eine Position und stellt sie der Steuerung zur Verfügung.

Die durch den Externen Geber erfasste Istposition kann beispielsweise für folgende Funktionen verwendet werden:

- Messwerterfassung durch einen Messtaster
- Positionsabhängige Erzeugung von Schaltsignalen und Schaltsignalfolgen durch Nocken und Nockenspur mit Istwertbezug
- Als Leitwert einer Gleichlaufachse (S7-1500T)

Eine Übersicht über die Funktionen des Technologieobjekts Externer Geber finden Sie im Kapitel "Funktionen (Seite 1683)".

Der Bezug der Geberistwerte zu einer definierten Position wird durch die Parametrierung der mechanischen Eigenschaften und Gebereinstellungen sowie einen Referenziervorgang hergestellt.

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Externer Geber:

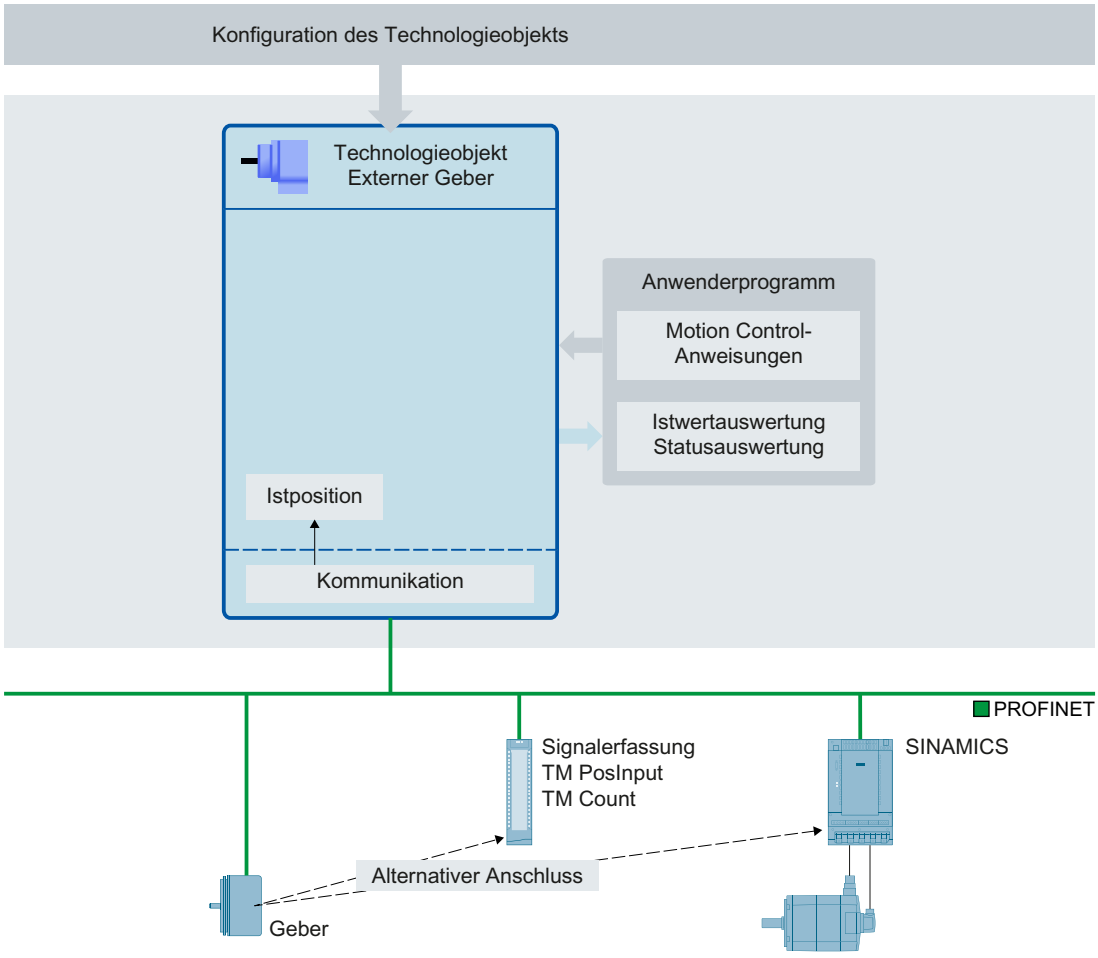

Die Angabe der Position erfolgt entsprechend dem gewählten Einheitensystem:

- **Lineares Einheitensystem**
  Die Position wird als Längenmaß angegeben, z. B. Millimeter.
- **Rotatorisches Einheitensystem**
  Die Position wird als Winkelmaß angegeben, z. B. Grad.

Technologieobjekt Messtaster

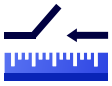

Das Technologieobjekt Messtaster erfasst bei einem Signalwechsel am Messeingang die Istposition einer Achse oder eines Externen Gebers.

Eine Übersicht über die Funktionen des Technologieobjekts Messtaster finden Sie im Kapitel "Funktionen (Seite 1683)".

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Messtaster:

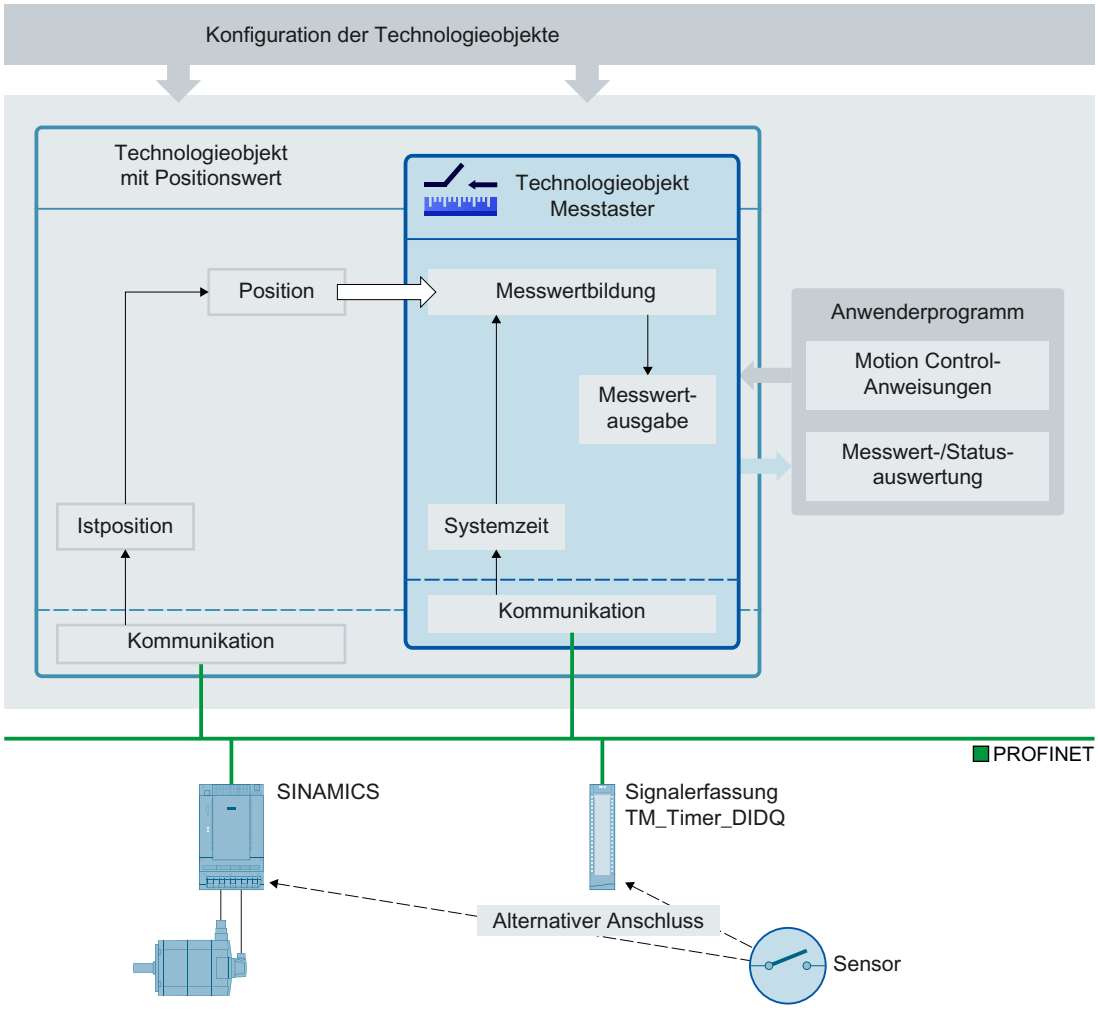

Messarten

Folgende Arten der Messung können durchgeführt werden:

- Einmaliges Messen (Seite 1784)
  Mit einem Messauftrag werden flankengenau bis zu zwei Messwerte erfasst. Ein einmaliger Messauftrag wird mit "MC_MeasuringInput" gestartet.
- Zyklisches Messen (Seite 1786)
  Mit dem zyklischen Messen werden in jedem Servotakt bis zu zwei Messwerte flankengenau erfasst.
  Ein zyklischer Messauftrag wird mit "MC_MeasuringInputCyclic" gestartet. Die Messungen werden zyklisch fortgesetzt, bis sie per Befehl beendet werden.

Die Auswahl der zu erfassenden Flanken erfolgt beim Starten der Messung über die Motion Control-Anweisung "MC_MeasuringInput" bzw. "MC_MeasuringInputCyclic".

Zuordnung

Das Technologieobjekt Messtaster benötigt immer eine Zuordnung zu einem anderen Technologieobjekt, dessen Position vom Messtaster ausgewertet wird.

Das Technologieobjekt Messtaster kann den folgenden Technologieobjekten zugeordnet werden:

- Gleichlaufachse
- Positionierachse
- Externer Geber

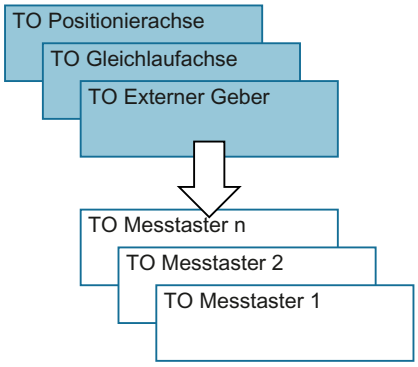

Einem Technologieobjekt Messtaster kann genau eine Achse oder ein Externer Geber zugeordnet werden.

Einer Achse oder einem Externen Geber können mehrere Technologieobjekte Messtaster zugeordnet werden.

Messwertermittlung

Die Positionserfassung kann mittels hardwareseitiger Unterstützung auf eine der folgenden Arten erfolgen:

- Messen über TM Timer DIDQ (Time-based)
- Messen über SINAMICS-Messtastereingang (Time-based)
  Messungen über einen Messtastereingang sind nur bei SINAMICS-Antrieben möglich, siehe "Kompatibilitätsliste (Seite 2222)".
- Messen über PROFIdrive-Telegramm (Antrieb oder Externer Geber)

Ein Messauftrag wird über die Motion Control-Anweisung "MC_MeasuringInput" oder "MC_MeasuringInputCyclic" aktiviert.

Bei einer Messwerterfassung über Zeitwert ("TIME_BASED") wird ein Signalwechsel des externen Triggersignals für die Messung über ein entsprechendes Modul wie TM Timer DIDQ oder SINAMICS Drive erfasst. Der Zeitstempel wird an die Steuerung übertragen und im Technologieobjekt die dazugehörende Istposition ermittelt.

Bei einer direkten Positionserfassung ("PROFIDRIVE") ist der Messeingang am Antriebsgerät angeschlossen und die Messung wird im Antrieb realisiert. Der Antrieb bzw. das Gebermodul liefert über das PROFIdrive-Telegramm direkt den erfassten Positionswert an das Technologieobjekt.

Der resultierende Messwert wird am jeweiligen Ausgang der Motion Control-Anweisung "MC_MeasuringInput" oder "MC_MeasuringInputCyclic" angezeigt.

Das zyklische Messen ist nur beim Messen über TM Timer DIDQ möglich. Das einmalige Messen ist immer möglich.

Das Messen über TM Timer DIDQ setzt zwingend den taktsynchronen Betrieb voraus.

Beim Messen über PROFIdrive-Telegramm kann nur ein Messtaster auf einen Istwert bzw. Geber im PROFIdrive-Telegramm zu einer Zeit aktiv sein. Auf einen Istwert bzw. Geber im PROFIdrive-Telegramm können maximal zwei Messtaster über PROFIdrive konfiguriert werden, siehe Kapitel "Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1698)".

Das Technologieobjekt Messtaster lässt sich nicht mit SIMATIC S7-PLCSIM verwenden. Das Technologieobjekt Messtaster und die im Anwenderprogramm verwendeten Messtasteraufträge lassen sich in SIMATIC S7-PLCSIM laden, sind aber ohne Funktion. Messwerte werden nicht gebildet.

Korrekturzeit

Durch die Einstellung einer Korrekturzeit (<TO>.Parameter.CorrectionTime) am Technologieobjekt Messtaster kann der Messzeitpunkt korrigiert werden.

Für die folgenden Beispiele können Korrekturen erforderlich sein:

- Zeiten für die mechanische Auslenkung des Tasters
- Zeiten für die Erzeugung des Messsignals vor dem Eingang am Messmodul
- Filterzeiten am Eingang oder Filterzeiten für die Messeingänge an dem SINAMICS-Antriebsgerät

Die Korrekturzeit wird bei einer Messwerterfassung über Zeitwert ("TIME_BASED") wie auch bei direkter Positionserfassung ("PROFIDRIVE") vom Technologieobjekt Messtaster eingerechnet.

Beachten Sie zusätzlich den Zeitverzug beim Messen auf einer Virtuellen Achse, siehe Kapitel "Istwerte (Seite 1696)".

Messen

Einmaliges Messen

Mit dem einmaligen Messen können mit einem Messauftrag bis zu zwei Flanken erfasst werden. Die zugehörigen Istpositionen werden am Funktionsbaustein und im Technologie-Datenbaustein zurückgemeldet und können im Anwenderprogramm weiterverarbeitet werden.

Messauftrag

Ein Messauftrag wird über die Motion Control-Anweisung "MC_MeasuringInput" gestartet. Die Variable "<TO>.Status" im Technologie-Datenbaustein wechselt auf "WAITING_FOR_TRIGGER". Das Technologieobjekt aktiviert die Messung, wenn die ausgewählte Flanke erkannt wird.

Die Messung erfolgt am Messeingang in Form von bis zu zwei Systemzeiten. Aus diesen wird nach Berücksichtigung einer vorhandenen Korrekturzeit die zugehörige Position bestimmt und ausgegeben.

Bei einer direkten Positionserfassung wird der erfasste Positionswert direkt vom Antrieb bzw. Gebermodul über das PROFIdrive-Telegramm an das Technologieobjekt geliefert.

Die Messung ist damit beendet. Ein weiteres Messen muss über die Motion Control-Anweisung "MC_MeasuringInput" neu gestartet werden.

Konfigurierter Parameter "Mode" in der Motion Control-Anweisung "MC_MeasuringInput"	Ausgang Motion Control-Anweisung "MC_MeasuringInput"	
	"MeasuredValue1"	"MeasuredValue2"
Nur steigende Flanke messen	Istposition zum Zeitpunkt der Flanke	–
Nur fallende Flanke messen		
Die nächsten beiden Flanken messen	Istposition zum Zeitpunkt der ersten Flanke	Istposition zum Zeitpunkt der zweiten Flanke
Die nächsten beiden Flanken messen, beginnend mit der steigenden Flanke	Istposition zum Zeitpunkt der steigenden Flanke	Istposition zum Zeitpunkt der fallenden Flanke
Die nächsten beiden Flanken messen, beginnend mit der fallenden Flanke	Istposition zum Zeitpunkt der fallenden Flanke	Istposition zum Zeitpunkt der steigenden Flanke

Im Technologie-Datenbaustein stehen speichernd die letzten erfassten Werte. Wird ein neuer Auftrag am Funktionsbaustein angestoßen, werden die Ausgänge am Funktionsbaustein initialisiert. Der Technologie-Datenbaustein wird nicht initialisiert. Nach Erfassen des ersten gültigen Messzyklus sind die Werte im Technologie-Datenbaustein und im Funktionsbaustein zueinander konsistent.

Der beendete Messauftrag wird am Funktionsbaustein in "MC_MeasuringInput.DONE" = TRUE bzw. im Technologie-Datenbaustein in "<TO>.Status" = "TRIGGER_OCCURRED" angezeigt.

Zeitliche Anforderungen für Messaufträge über "MC_MeasuringInput"

Durch hardwarebedingte Einschränkungen bei der Messung über PROFIdrive-Telegramm oder Messen über SINAMICS-Messtastereingang entstehen zeitliche Anforderungen für die Dauer, bis ein Messereignis erfasst werden kann.

Beim einmaligen Messen über "MC_MeasuringInput" mit "Mode" = 3 (Messung an beiden Flanken, beginnend mit der steigenden Flanke) oder "Mode" = 4 (Messung an beiden Flanken, beginnend mit der fallenden Flanke) ist deshalb zwischen der ersten zu messenden Flanke und der vorhergehenden Flanke ein Mindestabstand von mehreren Servotakten erforderlich, damit die erste zu messende Flanke erfasst werden kann.

Informationen zu den zeitlichen Randbedingungen finden Sie im Kapitel "Zeitliche Randbedingungen (Seite 1791)".

Siehe auch

Zeitliche Randbedingungen (Seite 1791)

Zyklisches Messen

Mit dem zyklischen Messen können in jedem Servotakt des Technologieobjekts bis zu zwei Messereignisse vom System erfasst und die dazugehörigen Messpositionen angezeigt werden. Die Messungen werden zyklisch fortgesetzt, bis sie per Befehl beendet werden.

Die ermittelten Messwerte werden angezeigt und können vom Anwenderprogramm ausgelesen werden.

Das zyklische Messen setzt eine Messwertermittlung mittels hardwareseitiger Unterstützung der Art "Messen über TM Timer DIDQ" voraus.

Messflanken, zu welchen kein Messwert ermittelt werden konnte, werden in einem Lost Edge Counter im Technologie-Datenbaustein als auch am Funktionsbaustein "MC_MeasuringInputCyclic" angezeigt.

Messauftrag

Ein zyklischer Messauftrag wird mit der Motion Control-Anweisung "MC_MeasuringInputCyclic" gestartet und der Messauftrag an das entsprechende Technologiemodul gegeben. Pro Servotakt können abhängig von der Funktionalität des Technologiemoduls, flankenbezogen bis zu zwei Messereignisse und damit Messzeitwerte erfasst und anschließend an das Technologieobjekt weitergegeben werden. Das Technologieobjekt ermittelt die Messpositionen zu den Messzeitwerten unter Berücksichtigung der gegebenenfalls angegebenen Korrekturzeiten.

Die Technologie-Datenbaustein Variable "<TO>.Status" wechselt von "INACTIVE" auf "WAITING_FOR_TRIGGER" und bleibt im weiteren Verlauf in diesem Status, solange auf weitere Ereignisse gewartet wird.

Der in der Motion Control-Anweisung eingestellte Mode gibt vor, für welche Flanken die Messwerte erfasst werden sollen. In einem Servotakt können maximal erfasst werden:

- Zwei steigende Flanken bei Erfassung der steigenden Flanken
- Zwei fallende Flanken bei Erfassung der fallenden Flanken
- Eine steigende und eine fallende Flanke bei Erfassung der steigenden und fallenden Flanken

Messwerte und Zähler

Bei positiver Flanke am Eingang "MC_MeasuringInputCyclic.Execute" werden die Ausgänge "MeasuredValue1Counter" und "MeasuredValue2Counter" auf "0" zurückgesetzt. Damit sind neue Ereignisse unmittelbar verfolgbar und neue Messwerteinträge erkennbar.

Alle erfolgten Messereignisse des Messauftrags werden im Technologie-Datenbaustein in den entsprechenden Ereigniszähler "<TO>.MeasuredValues.MeasuredValue1Counter" und "<TO>.MeasuredValues.MeasuredValue2Counter" jeweils um "1" hochgezählt.

Die erfassten Messwerte werden am Technologie-Datenbaustein unabhängig von einzelnen Aufträgen fortlaufend erfasst und die Werte erst mit dem Hochlauf des Technologieobjekts bzw. einem Restart am Technologieobjekt auf "0" zurückgesetzt.

Nach erfolgter Messung werden die Messwerte am Funktionsbaustein ausgegeben. Mit einem neuen Messauftrag werden die Zähler am Funktionsbaustein auf "0" gestellt. Die

Messwertausgabe am Technologie-Datenbaustein zeigt immer den letzten erfassten Messwert.

Lost Edge Counter (LEC)

Treten mehr als zwei zu erfassende Flanken innerhalb eines Servotakts auf, kann für die weiteren zu erfassenden Flanken kein Messwert ausgewertet werden. Die Anzahl der verloren gegangenen Flanken wird im LEC erfasst.

Welche verloren gegangenen Flanken im LEC erfasst werden, hängt von dem in der Motion Control-Anweisung eingestellten Mode ab. Sollen z. B. nur steigende Flanken gemessen werden, erfasst der LEC nur die nicht gemessenen steigenden Flanken.

Im LEC können maximal sieben verloren gegangene Flanken gezählt und angezeigt werden.

Die Anzahl der verloren gegangenen Flanken wird angezeigt im Funktionsbaustein und im Technologie-Datenbaustein in:

- "LostEdgeCounter1"
  Verloren gegangene Flanken aus dem Servotakt, in dem der "MeasuredValue1" erfasst wurde.
  ⇒ Der angezeigte Wert in "LostEdgeCounter1" wird aktualisiert, wenn der Zähler "MeasuredValueCounter1" inkrementiert wird.
- "LostEdgeCounter2"
  Verloren gegangene Flanken aus dem Servotakt, in dem der "MeasuredValue2" erfasst wurde.
  ⇒ Der angezeigte Wert in "LostEdgeCounter2" wird aktualisiert, wenn der Zähler "MeasuredValueCounter2" inkrementiert wird.

Anzeige der Messergebnisse beim zyklischen Messen

Im Befehl ausgewählte Flanken	Anzeige pro Servotakt			
	"MeasuredValue1"	"MeasuredValue2"	"LostEdgeCounter1"	"LostEdgeCounter2"
Nur steigende Flanken erfassen "MC_MeasuringInputCyclic.Mode" = 0	Istposition zum Zeitpunkt der ersten steigenden Flanke	Istposition zum Zeitpunkt der zweiten steigenden Flanke	Anzahl der mehr als zwei steigenden bzw. fallenden Flanken im Servotakt der Erfassung von "MeasuredValue1" und "MeasuredValue2" Dabei gilt: • Wenn ein "MeasuredValue1" und ein "MeasuredValue2" erfasst werden, ist die unter "LostEdgeCounter1" und "LostEdgeCounter2" angezeigte Anzahl an erfassten verloren gegangenen Flanken gleich. • Wenn nur ein "MeasuredValue1" erfasst wird, wird der "LostEdgeCounter1" auf "0" zurückgesetzt. Der Wert im "LostEdgeCounter2" bleibt unverändert.	
Nur fallende Flanken erfassen "MC_MeasuringInputCyclic.Mode" = 1	Istposition zum Zeitpunkt der ersten fallenden Flanke	Istposition zum Zeitpunkt der zweiten fallenden Flanke		
Steigende und fallende Flanken erfassen "MC_MeasuringInputCyclic.Mode" = 2	Istposition zum Zeitpunkt der ersten steigenden Flanke im Servotakt	Istposition zum Zeitpunkt der ersten fallenden Flanke im Servotakt	Anzahl der mehr als zwei Flanken im Servotakt der Erfassung von "MeasuredValue1" und "MeasuredValue2" Dabei gilt: • Wenn ein "MeasuredValue1" und ein "MeasuredValue2" erfasst werden, ist die unter "LostEdgeCounter1" und "LostEdgeCounter2" angezeigte Anzahl an erfassten verloren gegangenen Flanken gleich. • Wenn nur ein "MeasuredValue1" erfasst wird, wird der "LostEdgeCounter1" auf "0" zurückgesetzt. Der Wert im "LostEdgeCounter2" bleibt unverändert. • Wenn nur ein "MeasuredValue2" erfasst wird, wird der "LostEdgeCounter2" auf "0" zurückgesetzt. Der Wert im "LostEdgeCounter1" bleibt unverändert.	

Beispiele

Die nachfolgenden Abbildungen zeigen beispielhaft das Auseinanderlaufen von "MeasuredValue1Counter" und "MeasuredValue2Counter" infolge verloren gegangener Flanken.

Beispiel: Messung an steigenden Flanken (Mode = 0)

Bearbeitungstakt TO Messtaster	N	N+1	N+2	N+3	N+4	N+5	N+6	N+7	N+8	N+9	N+10
Measuring input											
MeasuredValue1Counter	...	6	7	7	8	9	9	10	11	12	13
LostEdgeCounter1	...	0	0	0	0	0	0	1	0	1	0
MeasuredValue2Counter	...	6	7	7	7	7	7	8	8	9	9
LostEdgeCounter2	...	0	0	0	0	0	0	1	1	1	1

Beispiel: Messung an fallenden Flanken (Mode = 1)

Bearbeitungstakt TO Messtaster	N	N+1	N+2	N+3	N+4	N+5	N+6	N+7	N+8	N+9	N+10
Measuring input											
MeasuredValue1Counter	...	6	7	8	9	10	10	11	11	12	13
LostEdgeCounter1	...	0	0	0	1	0	0	1	1	1	0
MeasuredValue2Counter	...	6	7	7	8	8	8	9	9	10	10
LostEdgeCounter2	...	0	0	0	1	1	1	1	1	1	1

Beispiel: Messung an steigenden und fallenden Flanken (Mode = 2)

Bearbeitungstakt TO Messtaster	N	N+1	N+2	N+3	N+4	N+5	N+6	N+7	N+8	N+9	N+10
Measuring input											
MeasuredValue1Counter	...	6	7	7	8	9	9	10	11	12	13
LostEdgeCounter1	...	0	2	2	0	0	0	3	0	4	0
MeasuredValue2Counter	...	6	7	8	8	9	10	11	12	13	14
LostEdgeCounter2	...	0	2	0	0	0	0	3	0	4	0

Messen mit Messbereich

Ein Messauftrag kann unmittelbar aktiviert oder auf einen definierten Messbereich begrenzt werden.

Die folgende Grafik zeigt ein Beispiel für das Messen mit Messbereich im "Mode" = 0 (Messung der nächsten steigenden Flanke):

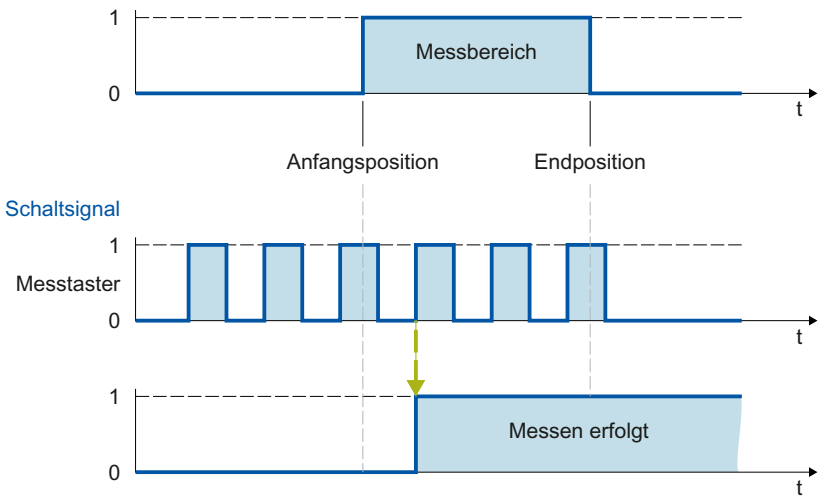

Nur Messwerte innerhalb des Messbereichs werden am Technologieobjekt angezeigt.

- Wird beim einmaligen Messen im Messbereich keine Messflanke erfasst, wird der Messauftrag abgebrochen und ein Alarm ausgelöst.
- Das zyklische Messen bleibt weiter aktiv, auch wenn keine Messflanke im Messbereich erfasst wurde.

Bei Achsen ohne Modulofunktion ist die Reihenfolge der Angabe von Anfangs- und Endposition unerheblich. Ist die Anfangsposition größer als die Endposition, werden beide Werte in der Anwendung getauscht. Ist bei einer Achse mit Modulofunktion die Anfangsposition größer als die Endposition, erstreckt sich der Messbereich von der Anfangsposition über den Moduloübergang der Achse bis zur Endposition.

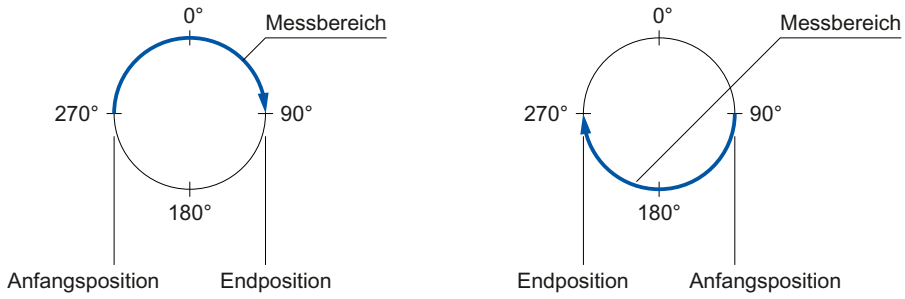

Die Messbereichspositionen werden in der Motion Control-Anweisung "MC_MeasuringInput" bzw. "MC_MeasuringInputCyclic" angegeben.

Aktivierungszeit für den Messbereich

Die Messfunktion muss beim Erreichen des Messbereichsanfangs am Messeingang aktiv sein. Um z. B. die Kommunikationszeit für die Aktivierung im TM Timer DIDQ bzw. Antrieb zu kompensieren, erfolgt die Aktivschaltung der Messung im Technologieobjekt um die Aktivierungszeit früher, als der Messbereichsanfang beginnt.

Die Aktivierungszeit für das Messen mit Messbereich ist wie folgt aufgeteilt:

- Der systemseitig bestimmte und wirksame Anteil der Aktivierungszeit wird in der Variable "<TO>.Parameter.MeasuringRangeActivationTime" angezeigt.
- Zusätzlich kann anwenderseitig eine additive Aktivierungszeit über die Variable "<TO>.Parameter.MeasuringRangeAdditionalActivationTime" eingestellt werden.

Zeitliche Randbedingungen

Abhängig vom Hardware-Ausbau und der Auswahl zu erfassender Flanken ergeben sich systembedingt unterschiedliche zeitliche Anforderungen, bis eine Messung nach Aufruf der Motion Control-Anweisung "MC_MeasuringInput" bzw. "MC_MeasuringInputCyclic" erfolgen kann und die Ergebnisse dort angezeigt werden.

Dabei ist zwischen den folgenden Zeiten zu unterscheiden:

- Zeit, bis das Messereignis erfasst werden kann
- Zeit, bis das Messergebnis angezeigt wird bzw. die Messung zu Ende ist

Die unter Berücksichtigung der aktuellen Einstellungen ermittelten Zeiten werden im Konfigurationsfenster "Erweiterte Parameter" eines Messtasters angezeigt.

Messen über TM Timer DIDQ/Messen über SINAMICS-Messtastereingang

- Zeit vom Absetzen eines Auftrags "MC_MeasuringInput" bis Messereigniserfassung wirksam ist:
  - "MeasuringRangeActivationTime": 2 x T_{servo}
- Zeit nach Messereignis bis Messwert in der Steuerung verfügbar:
  - Einmaliges Messen: 2 x T_{servo} + T_{Send}
  - Zyklisches Messen: T_{servo} + T_{Send}
- Aktivierungszeit für eine Messung mit Messbereich:
  - "MeasuringRangeActivationTime" + "MeasuringRangeAdditionalActivationTime"

Messen über PROFIdrive-Telegramm (Antrieb oder Externer Geber)

- "MeasuringRangeActivationTime": 4 x T_{servo}
- Zeit vom Absetzen eines Auftrags "MC_MeasuringInput" bis Messereigniserfassung wirksam ist:
  - Messen einer steigenden/fallenden Flanke oder zwei Flanken: "MeasuringRangeActivationTime" + 2 x T_{servo}
  - Messen von zwei dedizierten Flanken: "MeasuringRangeActivationTime" + 3 x T_{servo}

- Zeit nach Messereignis bis Messwert in der Steuerung verfügbar:
  - Messen einer Flanke: 7 x T_{servo}
  - Messen von zwei Flanken: 13 x T_{servo}
- Aktivierungszeit für eine Messung mit Messbereich:
  - Messen einer steigenden/fallenden Flanke oder zwei Flanken: "MeasuringRangeActivationTime" + "MeasuringRangeAdditionalActivationTime" + 2 x T_{servo}
  - Messen von zwei dedizierten Flanken: "MeasuringRangeActivationTime" + "MeasuringRangeAdditionalActivationTime" + 3 x T_{servo}

Definition Variablen

- T_{servo} = Aufrufintervall des Technologieobjekts im Servotakt [ms]
- T_{Send} = Sendetakt [ms]
- "MeasuringRangeActivationTime" = Siehe Kapitel "Messen mit Messbereich (Seite 1789)"

Um eine Überschreibung eines gerade ermittelten Messwerts durch asynchrone Bearbeitung zu vermeiden, wird empfohlen, einen neuen einmaligen Messauftrag erst nach Abschluss der aktiven Messung zu starten. Zeitlich muss dabei die Summe der Aktivierungszeit und der Zeit bis zur Anzeige bzw. dem Abschluss berücksichtigt werden.

Beim zyklischen Messen ist eine Auswertung bzw. Zwischenspeicherung der Messergebnisse aus Anwendersicht im synchronen MC-PostServo [OB95] empfohlen.

Variablen

Folgende Variablen des Technologieobjekts sind für das Messen relevant:

Statusanzeige		
Variable	**Beschreibung**	
<TO>.Status	Status der Messtasterfunktion	
	0	Messung ist nicht aktiv ("INACTIVE")
	1	Der Messtaster wartet auf ein Messereignis ("WAITING_FOR_TRIGGER")
	2	Der Messtaster hat einen oder mehrere Messwerte erfasst ("TRIGGER_OCCURRED")
	3	Fehler bei der Messung ("MEASURING_ERROR")
<TO>.InputState	Status des Messtastereingangs	

Parameter	
Variable	**Beschreibung**
<TO>.Parameter.MeasuringInputType	Messtastertyp
<TO>.Parameter.PROFIdriveProbeNumber	Nummer des zu verwendenden Messeingangs bei einer Messung über PROFIdrive-Telegramm

Parameter	
Variable	**Beschreibung**
<TO>.Parameter.MeasuringRangeActivation-Time	Systemseitig eingestellter Anteil der Aktivierungszeit [ms]
<TO>.Parameter.MeasuringRangeAdditional-ActivationTime	Zusätzlicher, anwenderseitig einstellbarer Anteil der Aktivierungszeit [ms]
<TO>.Parameter.CorrectionTime	Anwenderseitig eingestellte Korrekturzeit für das Messergebnis [ms]

Interface	
Variable	**Beschreibung**
<TO>.Interface.Address	Peripherieadresse für den digitalen Messeingang

Units	
Variable	**Beschreibung**
<TO>.Units.LengthUnit	Einheit der Längenangabe
<TO>.Units.TimeUnit	Einheit der Zeitangabe

MeasuredValues	
Variable	**Beschreibung**
<TO>.MeasuredValues.MeasuredValue1	Erster Messwert
<TO>.MeasuredValues.MeasuredValue2	Zweiter Messwert (bei Messung von zwei oder mehreren Flanken in einem Servotakt)
<TO>.MeasuredValues.MeasuredValue1Counter	Zählwert für den ersten Messwert
<TO>.MeasuredValues.MeasuredValue2Counter	Zählwert für den zweiten Messwert
<TO>.MeasuredValues.LostEdgeCounter1	Verloren gegangene Flanken im Takt der Erfassung vom ersten Messwert (beim einmaligen Messen gleich null)
<TO>.MeasuredValues.LostEdgeCounter2	Verloren gegangene Flanken im Takt der Erfassung vom zweiten Messwert (beim einmaligen Messen gleich null)

StatusWord	
Variable	**Beschreibung**
<TO>.StatusWord.X0 (Control)	Das Technologieobjekt ist in Betrieb.
<TO>.StatusWord.X1 (Error)	Am Technologieobjekt ist ein Fehler aufgetreten.
<TO>.StatusWord.X2 (RestartActive)	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert.
<TO>.StatusWord.X3 (OnlineStartValuesChanged)	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen.
<TO>.StatusWord.X5 (CommunicationOk)	Der Messtaster ist mit dem Messmodul synchronisiert und kann verwendet werden.

ErrorWord	
Variable	**Beschreibung**
<TO>.ErrorWord.X0 (SystemFault)	Ein systeminterner Fehler ist aufgetreten.
<TO>.ErrorWord.X1 (ConfigFault)	Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert.
<TO>.ErrorWord.X2 (UserFault)	Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
<TO>.ErrorWord.X3 (CommandNotAccepted)	Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
<TO>.ErrorWord.X13 (PeripheralError)	Fehler beim Zugriff auf eine logische Adresse

ErrorDetail		
Variable	**Beschreibung**	
<TO>.ErrorDetail.Number	Alarmnummer	
<TO>.ErrorDetail.Reaction	Wirksame Alarmreaktion	
	0	Keine Reaktion
	6	Messtasterbearbeitung beenden

Technologieobjekt Nocken

Das Technologieobjekt Nocken erzeugt Schaltsignale abhängig von der Position einer Achse oder eines Externen Gebers. Die Schaltzustände können im Anwenderprogramm ausgewertet werden und auf digitale Ausgänge geschaltet werden.

Eine Übersicht über die Funktionen des Technologieobjekts Nocken finden Sie im Kapitel "Funktionen (Seite 1683)".

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Nocken:

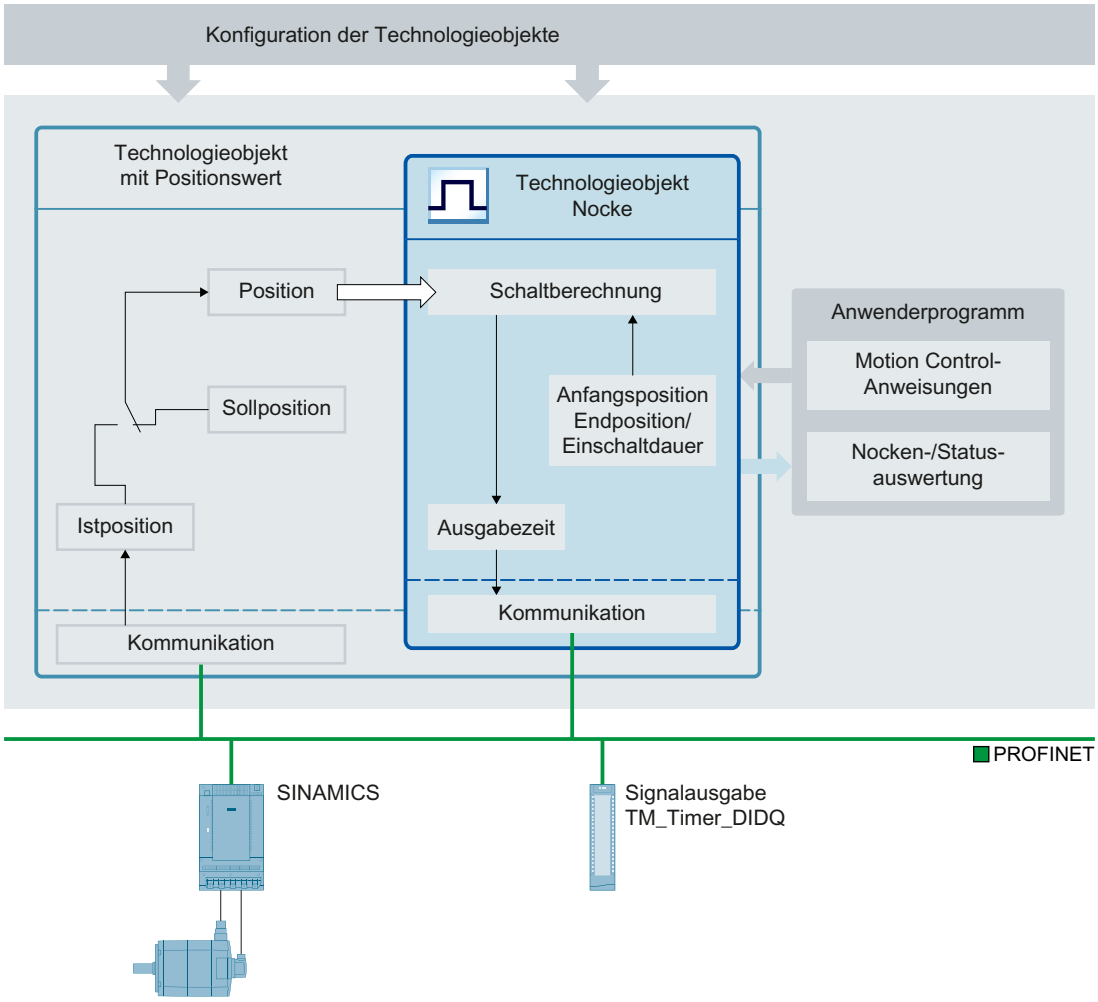

Nockentypen

Folgende Nockentypen können verwendet werden:

- Wegnocken (Seite 1797)
  Wegnocken schalten zwischen Anfangsposition und der Endposition ein. Außerhalb dieses Bereichs ist der Wegnocken ausgeschaltet.
- Zeitnocken (Seite 1799)
  Zeitnocken schalten mit Erreichen der Anfangsposition für eine definierte Zeitdauer ein.

Zuordnung

Das Technologieobjekt Nocken benötigt immer eine Zuordnung zu einem anderen Technologieobjekt, dessen Position ausgewertet wird.

Das Technologieobjekt Nocken kann den folgenden Technologieobjekten zugeordnet werden:

- Gleichlaufachse
- Positionierachse
- Externer Geber

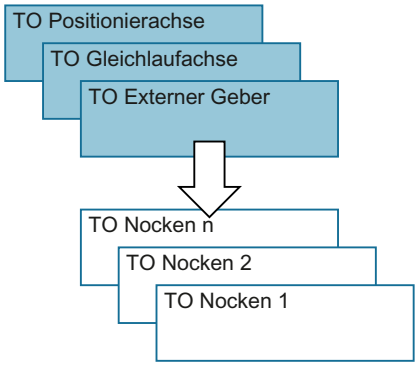

Einem Nocken kann genau eine Achse oder ein Externer Geber zugeordnet werden.

Einer Achse oder einem Externer Geber können mehrere Nocken zugeordnet werden.

Nockenberechnung und Nockenausgabe

Das Technologieobjekt Nocken berechnet den genauen Schaltzeitpunkt und sorgt damit für ein exaktes Einhalten der Schaltpositionen. Die Berechnung des Schaltzeitpunkts erfolgt zwei Servotakte vor der Ausgabe.

Folgende Ausgabemöglichkeiten stehen für den digitalen Nockenausgang zur Verfügung:

- TM Timer DIDQ
  Digitalausgang mit hoher Genauigkeit und Reproduzierbarkeit im Mikrosekundenbereich auf Time-based IO-Modulen ET 200MP TM Timer DIDQ 16x24V und ET 200SP TM Timer DIDQ 10x24V
  Die Ausgabe über TM Timer DIDQ setzt zwingend den taktsynchronen Betrieb voraus. Der taktsynchrone Betrieb ist nur mit PROFINET Interfacemodulen möglich.
- Digitalausgabemodul
  Digitalausgang mit Schaltgenauigkeit abhängig vom Ausgabezyklus der verwendeten Peripherie

Bei deaktivierter Ausgabe wird der Nockenstatus nicht am Hardware-Ausgang ausgegeben. Der Nockenstatus kann durch die Auswertung der betreffenden Variable "<TO>.CamOutput" intern im Anwenderprogramm verwendet werden.

Invertierte Ausgabe

Bei der invertierten Ausgabe werden der Bereich, in dem der Nockenausgang eingeschaltet ist, und der Bereich, in dem der Nockenausgang ausgeschaltet ist, miteinander vertauscht.

Die invertierte Ausgabe wird in der Motion Control-Anweisung "MC_OutputCam" eingestellt und ist aktiv, wenn die Anweisung freigegeben ist.

Die invertierte Ausgabe kann sowohl für Wegnocken als auch für Zeitnocken eingestellt werden.

Ausgabe mehrerer Nocken auf einen Ausgang

Die Ausgabe von mehreren Nocken auf einen Ausgang erfolgt wahlweise mit einer logischen UND- oder einer ODER-Verknüpfung der Nockensignale auf den Ausgang.

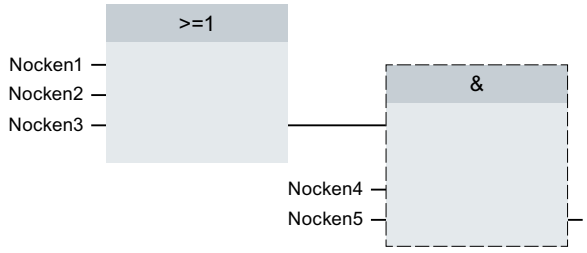

Anzeige des Schaltzustands

Der Schaltzustand des Nockens wird im zugehörigen Technologie-Datenbaustein in "<TO>.CamOutput" angezeigt.

Positionsbezug

Die Schaltpunkte der Nocken können sich abhängig vom verschalteten Technologieobjekt auf die folgenden Positionen beziehen.

- Istposition einer Gleichlauf-/Positionierachse
- Sollposition einer Gleichlauf-/Positionierachse
- Position eines Externen Gebers

Referenzieren des verschalteten Technologieobjekts

Eine Änderung der Position einer Achse oder des Externen Gebers über die Motion Control Anweisung "MC_Home" wird als Positionssprung gewertet.

- Wegnocken werden möglicherweise übersprungen oder werden entsprechend ausgegeben.
- Zeitnocken werden übersprungen. Erst mit Überfahren der Anfangsposition wird ein Zeitnocken geschaltet und bleibt für die Einschaltdauer eingeschaltet.
- Geschaltete Zeitnocken werden durch eine Referenzierung nicht abgebrochen.

Wegnocken

Einschaltbereich

Der Einschaltbereich von Wegnocken ist grundsätzlich definiert durch die Anfangsposition und die Endposition.

Anfangsposition kleiner als die Endposition

Wenn die Anfangsposition kleiner als die Endposition ist, beginnt der Einschaltbereich mit der Anfangsposition und endet mit der Endposition.

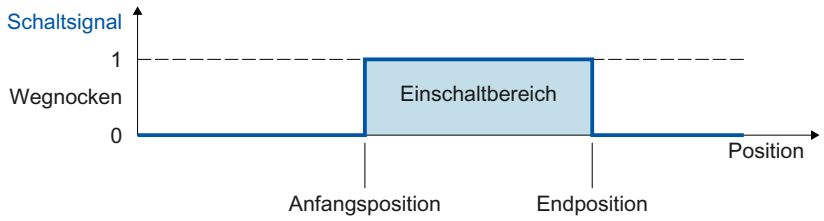

Anfangsposition größer als die Endposition

Wenn die Anfangsposition größer als die Endposition ist, sind die beiden Einschaltbereiche wie folgt:

- Einschaltbereich beginnt mit der Anfangsposition und endet mit dem positiven Bereichsende (z. B. positiver Software-Endschalter, Ende Modulobereich)
- Einschaltbereich beginnt mit dem negativen Bereichsende (z. B. negativer Software-Endschalter, Modulobereichsanfang) und endet mit der Endposition

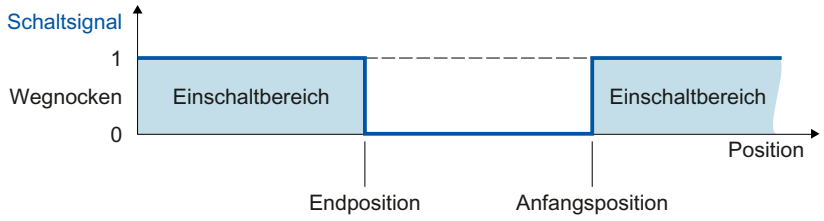

Abbildung auf eine Achse mit Modulofunktion

Bei aktiver Modulofunktion des verschalteten Technologieobjekts werden die Anfangs- und die Endposition des Nockens automatisch auf Werte innerhalb des Modulobereichs abgebildet.

Beispiel

- Modulobereich = 0° bis 50°
- Nocken Anfangsposition = 80°
- Nocken Endposition = 220°

⇒ Der Wegnocken schaltet ein bei 30° und schaltet bei 20° wieder aus.

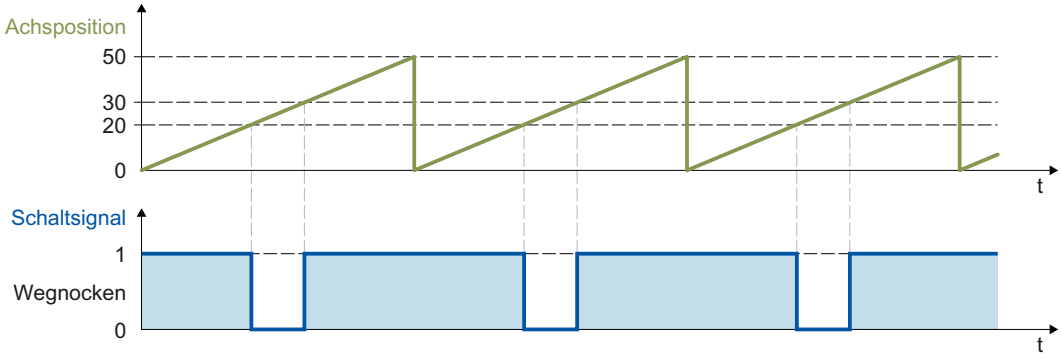

Schaltverhalten

Ein Wegnocken schaltet nach der Aktivierung in den folgenden Fällen ein:

- Die Position des verschalteten Technologieobjekts erreicht die Anfangs- bzw. Endposition mit der an der Motion Control-Anweisung "MC_OutputCam" konfigurierten Wirkrichtung.
- Die Position des verschalteten Technologieobjekts wird mit der an der Motion Control-Anweisung "MC_OutputCam" konfigurierten Wirkrichtung in den Einschaltbereich des Nockens verschoben (z. B. bei Referenzierung). Sind in der Motion Control-Anweisung "MC_OutputCam" beide Wirkrichtungen freigegeben, schaltet der Nocken auch im Stillstand ein.
- Der Nocken wird über die Motion Control-Anweisung "MC_OutputCam" mit "Mode" = 3 permanent eingeschaltet.

Ein aktiver Wegnocken schaltet in den folgenden Fällen aus:

- Die Position liegt außerhalb des Einschaltbereichs des Nockens.
- Der Positionswert wird außerhalb des Einschaltbereichs des Nockens verschoben.
- Die Motion Control-Anweisung "MC_OutputCam" wird auf "Enable" = FALSE gesetzt.
- Die Bewegungsrichtung des verschalteten Technologieobjekts wird umgekehrt und stimmt nicht mehr mit der freigegebenen Wirkrichtung überein.

Zeitnocken

Ein Zeitnocken schaltet an der Anfangsposition ein und bleibt für die Einschaltdauer gesetzt.

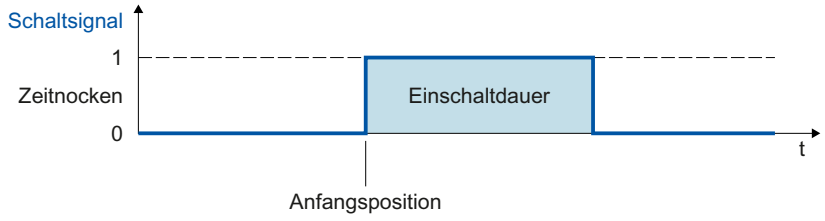

Abbildung auf eine Achse mit Modulofunktion

Bei aktiver Modulofunktion des verschalteten Technologieobjekts wird die Anfangsposition des Nockens automatisch auf den Wert innerhalb des Modulobereichs abgebildet.

Beispiel

- Modulobereich = 0° bis 50°
- Nocken Anfangspositionen:
  - Nocken 1 = 20°
  - Nocken 2 = 30°
  - Nocken 3 = 80°

⇒ Der Zeitnocken 1 schaltet ein bei 20°, die Zeitnocken 2 und 3 schalten ein bei 30°. Alle drei Nocken bleiben für die jeweils eingestellte Einschaltdauer aktiv.

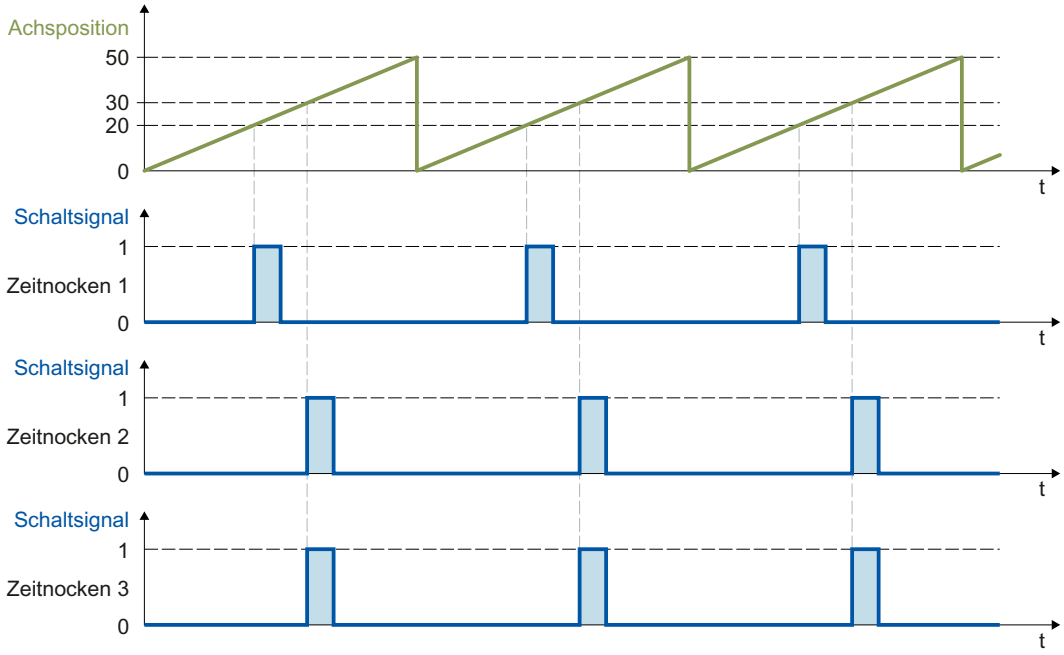

Schaltverhalten

Ein Zeitnocken schaltet in den folgenden Fällen ein:

- Die Anfangsposition wurde erreicht und die Bewegungsrichtung des verschalteten Technologieobjekts entspricht der durch die Anweisung freigegebenen Wirkrichtung.

Hinweis

- Wird die Anfangsposition während eines eingeschalteten Nockens erneut erreicht, wird die Einschaltdauer nicht nachgetriggert.
- Wird durch die Motion Control-Anweisung "MC_Home" der Positionswert des verschalteten Technologieobjekts während der Bewegung direkt auf oder hinter die Anfangsposition des Nockens gesetzt, schaltet der Nocken nicht ein.

Ein Zeitnocken schaltet in den folgenden Fällen aus:

- Die konfigurierte Einschaltdauer ist abgelaufen.
- Die Motion Control-Anweisung "MC_OutputCam" wird auf "Enable" = FALSE gesetzt.

Wirkrichtung von Nocken

Ein Nocken kann in Abhängigkeit von der Bewegungsrichtung des verschalteten Technologieobjekts geschaltet werden. Eine Ausgabe des Nockens in positiver bzw. negativer Bewegungsrichtung oder auch richtungsunabhängig ist möglich.

Die Wirkrichtung wird im "MC_OutputCam.Direction" eingestellt.

Die nachfolgenden Beispiele zeigen das Verhalten der Nocken abhängig von der eingestellten Wirkrichtung.

Beispiel Wirkrichtung "positiv" ("Direction" = 1)

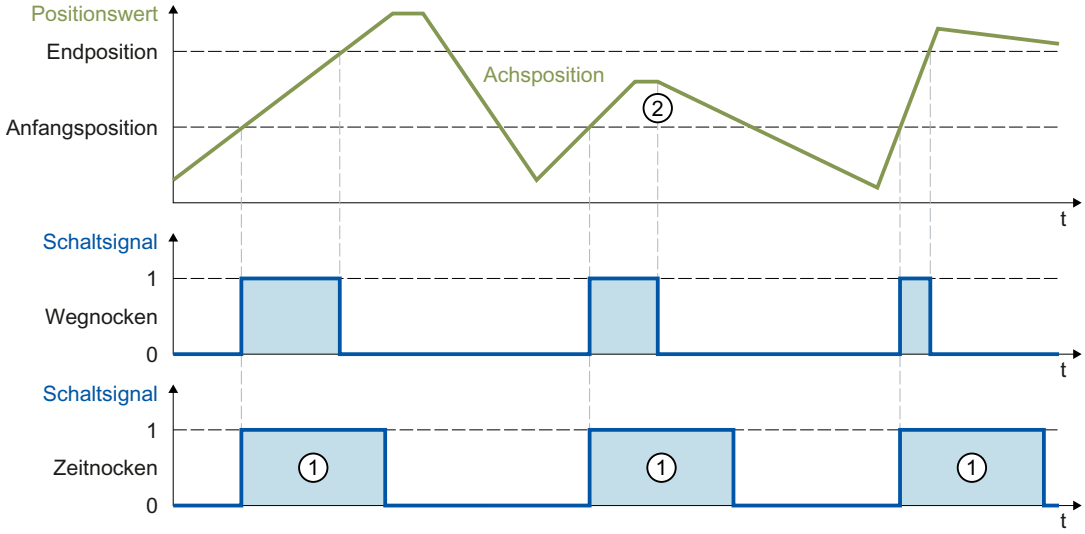

Wegnocken

Der Nocken schaltet ein, wenn der Einschaltbereich in **positiver** Richtung erreicht wird. Bei einer Richtungsumkehr ② schaltet der Nocken aus.

Wird der Positionswert in den Einschaltbereich des Nockens verschoben, schaltet der Nocken bei positiver Bewegungsrichtung des verschalteten Technologieobjekts ein. Im Stillstand bleibt der Nocken ausgeschaltet.

Zeitnocken

Der Nocken schaltet ein, wenn die Anfangsposition in **positiver** Richtung erreicht wird. Bei einer Richtungsumkehr bleibt der Nocken weiterhin für die eingestellte Einschaltdauer ① eingeschaltet.

Wird der Positionswert des verschalteten Technologieobjekts beim Referenzieren, während der Bewegung direkt auf oder hinter die Anfangsposition des Nockens gesetzt, schaltet der Nocken nicht ein.

Beispiel Wirkrichtung "negativ" ("Direction" = 2)

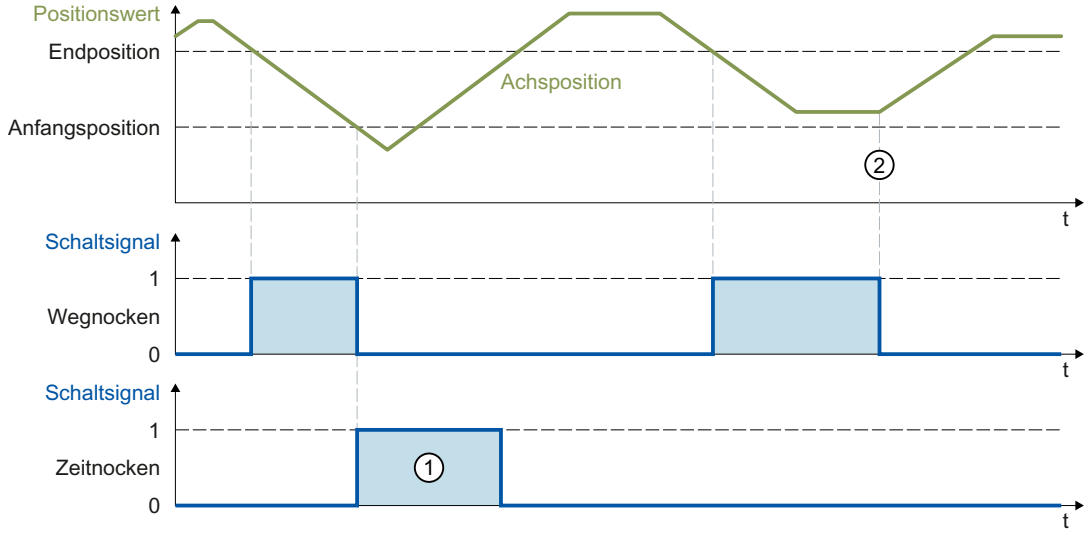

Wegnocken

Der Nocken schaltet ein, wenn der Einschaltbereich in **negativer** Richtung erreicht wird. Bei einer Richtungsumkehr ② schaltet der Nocken aus.

Wird der Positionswert in den Einschaltbereich des Nockens verschoben, schaltet der Nocken bei negativer Bewegungsrichtung des verschalteten Technologieobjekts ein. Im Stillstand bleibt der Nocken ausgeschaltet.

Zeitnocken

Der Nocken schaltet ein, wenn die Anfangsposition in **negativer** Richtung erreicht wird. Bei einer Richtungsumkehr bleibt der Nocken weiterhin für die eingestellte Einschaltdauer ① eingeschaltet.

Wird der Positionswert des verschalteten Technologieobjekts beim Referenzieren, während der Bewegung direkt auf oder hinter die Anfangsposition des Nockens gesetzt, schaltet der Nocken nicht ein.

Beispiel Wirkrichtung "beide Richtungen" ("Direction" = 3)

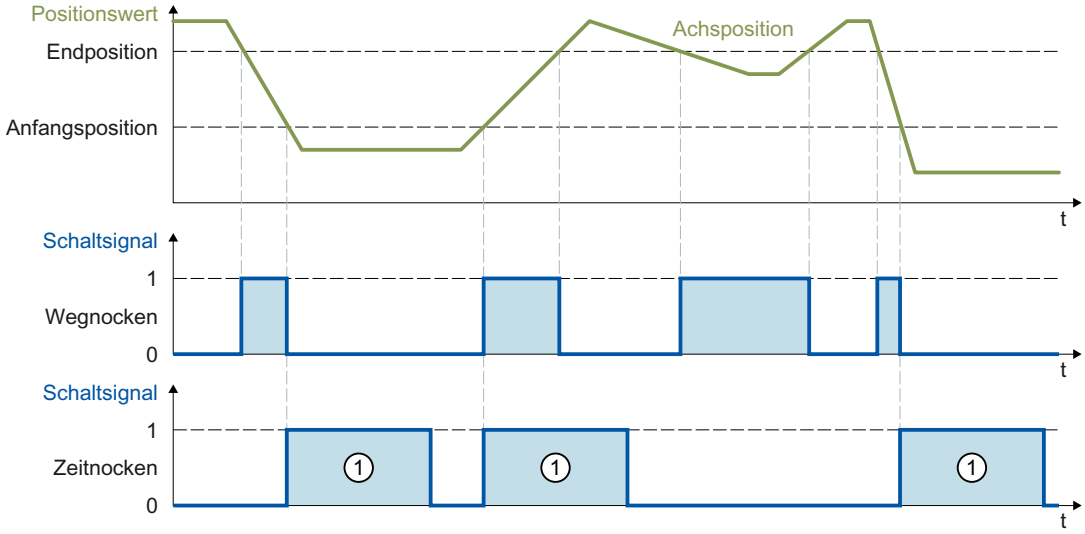

Wegnocken

Der Nocken schaltet ein, wenn die Position des verschalteten Technologieobjekts im Einschaltbereich liegt.

Wird der Positionswert des verschalteten Technologieobjekts in den Einschaltbereich des Nockens verschoben, schaltet der Nocken auch im Stillstand ein.

Zeitnocken

Der Nocken schaltet ein, wenn die Anfangsposition erreicht wird. Bei einer Richtungsumkehr bleibt der Nocken weiterhin für die eingestellte Einschaltdauer ① eingeschaltet.

Wird der Positionswert des verschalteten Technologieobjekts beim Referenzieren, während der Bewegung direkt auf oder hinter die Anfangsposition des Nockens gesetzt, schaltet der Nocken nicht ein.

Hysterese

Mögliche Schwankungen in der Ist-/Sollposition können bei Nocken zu ungewolltem Ein- bzw. Ausschalten führen.

Minimale Änderungen des Istwerts einer Achse im Stillstand können zum Ein- bzw. Ausschalten eines Istwertnockens mit vorgegebener positiver bzw. negativer Wirkrichtung führen. Auch minimale Veränderungen der Sollwerte einer abgeschalteten Achse im Nachführbetrieb können zum Ein- bzw. Ausschalten eines Sollwertnockens mit vorgegebener positiver bzw. negativer Wirkrichtung führen. Durch die Konfiguration einer Hysterese (> 0.0) können solche ungewollten Schaltzustände unterbunden werden. Insbesondere beim Bezug auf die Istposition wird die Konfiguration eines Hysteresewerts (> 0.0) empfohlen.

Die Hysterese ist eine Positionstoleranz, innerhalb der sich die Positionswerte ändern dürfen, ohne dass der Schaltzustand des Nockens geändert wird. Innerhalb der Hysterese erkannte Richtungswechsel werden ignoriert.

Die Hysterese wird am Technologieobjekt in "<TO>.Parameter.Hysteresis" eingestellt.

Verhalten

- Die Hysterese wird bei Richtungsumkehr aktiviert.
- Innerhalb der Hysterese gilt Folgendes:
  - Der Schaltzustand von Wegnocken wird nicht verändert.
  - Die Bewegungsrichtung wird nicht neu bestimmt.
  - Liegt die Anfangsposition eines Zeitnockens innerhalb der Hysterese, wird der Zeitnocken mit Verlassen der Hysterese bei entsprechender Wirkrichtung eingeschaltet.
  - Die Einschaltdauer von Zeitnocken bleibt unverändert.
- Nach Verlassen des Hysteresebereichs werden Wegnocken entsprechend den Nockeneinstellungen gesetzt.

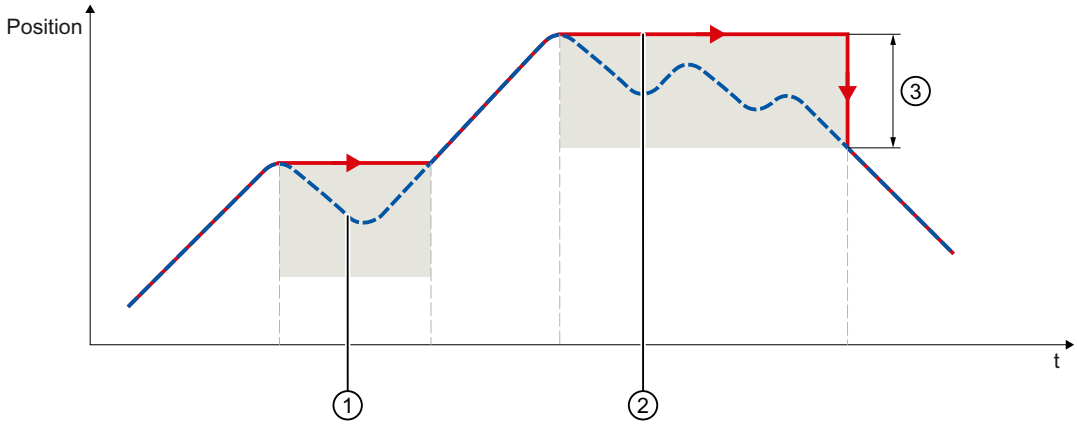

① Tatsächliche Position
② Wirksame Position
③ Hysteresebereich

Die nachfolgenden Beispiele zeigen die Auswirkungen der Hysterese auf das Schaltverhalten der Nocken mit positiver Wirkrichtung.

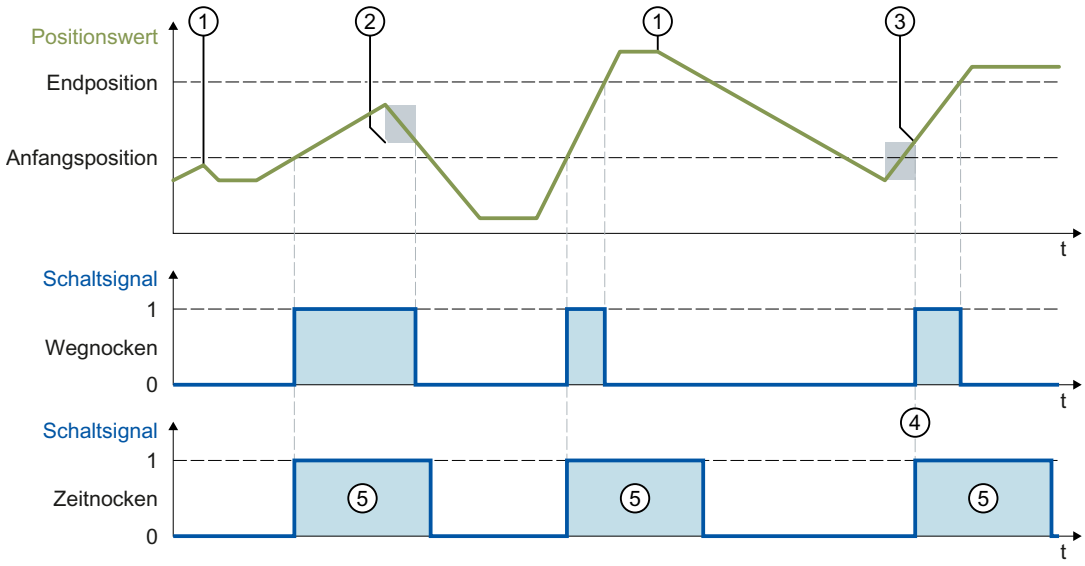

① Richtungsumkehr ohne Hysteresewirkung

② Hysterese wirksam

③ Die Einschaltposition des Wegnockens wird entsprechend der Richtungsumkehr und Hysterese beeinflusst.

④ Die Anfangsposition des Zeitnockens liegt innerhalb der Hysterese. Der Zeitnocken wird mit Verlassen der Hysterese bei entsprechender Wirkrichtung eingeschaltet.

⑤ Einschaltdauer

Hysteresebereich

Als maximale Größe des Hysteresebereichs gilt im System für eine Achse mit Modulofunktion ein Viertel des Modulobereichs und für eine Achse ohne Modulofunktion ein Viertel des Arbeitsbereichs.

Kompensation von Aktorschaltzeiten

Mithilfe der Aktivierungs- bzw. Deaktivierungszeit am Technologieobjekt Nocken können Schaltzeiten des Ausgangs und des angeschlossenen Aktors (z. B. Ventil) ausgeglichen werden.

Die Aktivierungszeit wird als Vorhaltezeit an der Einschaltflanke, die Deaktivierungszeit als Vorhaltezeit an der Ausschaltflanke angegeben.

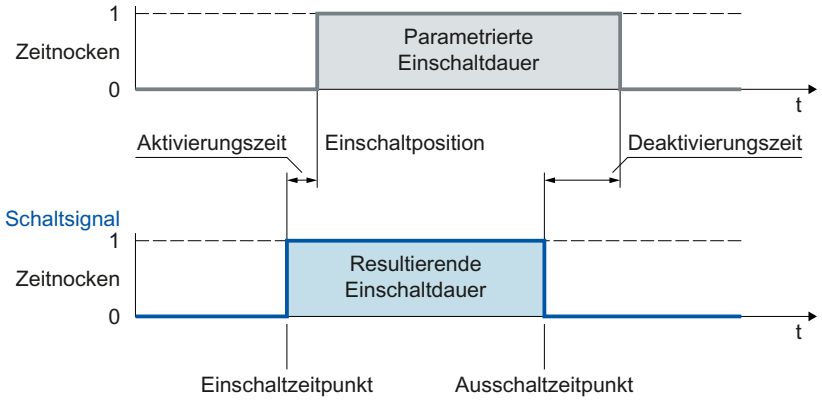

Die Aktivierungszeit wird am Technologieobjekt über die Variable "<TO>.Parameter.OnCompensation" eingestellt.

Die Deaktivierungszeit wird am Technologieobjekt über die Variable "<TO>.Parameter.OffCompensation" eingestellt.

Hinweis

- Für die Zeiten muss Folgendes erfüllt sein:
  Einschaltdauer > Deaktivierungszeit - Aktivierungszeit
  (Die Einschaltdauer wird bei Wegnocken durch die Schaltpositionen und die aktuelle Geschwindigkeit bestimmt.)
- Wenn der Nocken unter Berücksichtigung der Vorhaltezeiten geschaltet wurde, ist diese Aktion aus Systemsicht erfolgt und wird auch bei einer sich gegebenenfalls anschließend ändernden Aktualgeschwindigkeit nicht neu aufgesetzt.
- Systembedingte Laufzeiten werden bei der Verwendung der Ausgabe über TM Timer DIDQ automatisch berücksichtigt.
  Bei Ausgabe über ein Digitalausgabemodul werden systembedingte Laufzeiten nicht berücksichtigt und sind in der Aktivierungszeit bzw. Deaktivierungszeit als Korrekturzeiten einzustellen.

Variablen

Folgende Variablen des Technologieobjekts sind relevant:

Statusanzeige	
Variable	**Beschreibung**
<TO>.CamOutput	Der Nocken ist geschaltet.

Parameter		
Variable	**Beschreibung**	
<TO>.Parameter.OutputCamType	Nockentyp	
	0	Wegnocken
	1	Zeitnocken

Parameter		
Variable	**Beschreibung**	
<TO>.Parameter.PositionType	Positionsbezug	
	0	Sollposition
	1	Istposition
<TO>.ParameterOnCompensation	Aktivierungszeit (Vorhaltezeit an der Einschaltflanke)	
<TO>.Parameter.OffCompensation	Deaktivierungszeit (Vorhaltezeit an der Ausschaltflanke)	
<TO>.Parameter.Hysteresis	Hysteresewert	

Interface		
Variable	**Beschreibung**	
<TO>.Interface.EnableOutput	Aktivierung der Nockenausgabe	
	FALSE	Keine Ausgabe
	TRUE	Ausgabe
<TO>.Interface.Address	Peripherieadresse des Nockens	
<TO>.Interface.LogicOperation	Logische Verknüpfung der Nockensignale am Ausgang	
	0	ODER-Verknüpfung
	1	UND-Verknüpfung

Units	
Variable	**Beschreibung**
<TO>.Units.LengthUnit	Einheit der Längenangabe
<TO>.Units.TimeUnit	Einheit der Zeitangabe

StatusWord	
Variable	**Beschreibung**
<TO>.StatusWord.X0 (Control)	Das Technologieobjekt ist in Betrieb.
<TO>.StatusWord.X1 (Error)	Am Technologieobjekt ist ein Fehler aufgetreten.
<TO>.StatusWord.X2 (RestartActive)	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert.
<TO>.StatusWord.X3 (OnlineStartValuesChanged)	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen.
<TO>.StatusWord.X4 (OutputInverted)	Die Nockenausgabe erfolgt invertiert.
<TO>.StatusWord.X5 (CommunicationOk)	Der Nocken ist mit dem Ausgabemodul synchronisiert und kann verwendet werden.

ErrorWord	
Variable	**Beschreibung**
<TO>.ErrorWord.X0 (SystemFault)	Ein systeminterner Fehler ist aufgetreten.
<TO>.ErrorWord.X1 (ConfigFault)	Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert.
<TO>.ErrorWord.X2 (UserFault)	Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
<TO>.ErrorWord.X3 (CommandNotAccepted)	Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
<TO>.ErrorWord.X13 (PeripheralError)	Fehler beim Zugriff auf eine logische Adresse

ErrorDetail		
Variable	**Beschreibung**	
<TO>.ErrorDetail.Number	Alarmnummer Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".	
<TO>.ErrorDetail.Reaction	Wirksame Alarmreaktion	
	0	Keine Reaktion
	6	Nockenbearbeitung wird beendet
	Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".	

Technologieobjekt Nockenspur

Das Technologieobjekt Nockenspur erzeugt eine Schaltsignalfolge abhängig von der Position einer Achse oder eines Externen Gebers. Eine Nockenspur kann aus bis zu 32 einzelnen Nocken bestehen und auf einen Ausgang ausgegeben werden. Die Schaltzustände können im Anwenderprogramm ausgewertet oder auf digitale Ausgänge geschaltet werden.

Eine Übersicht über die Funktionen des Technologieobjekts Nockenspur finden Sie im Kapitel "Funktionen (Seite 1683)".

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Nockenspur:

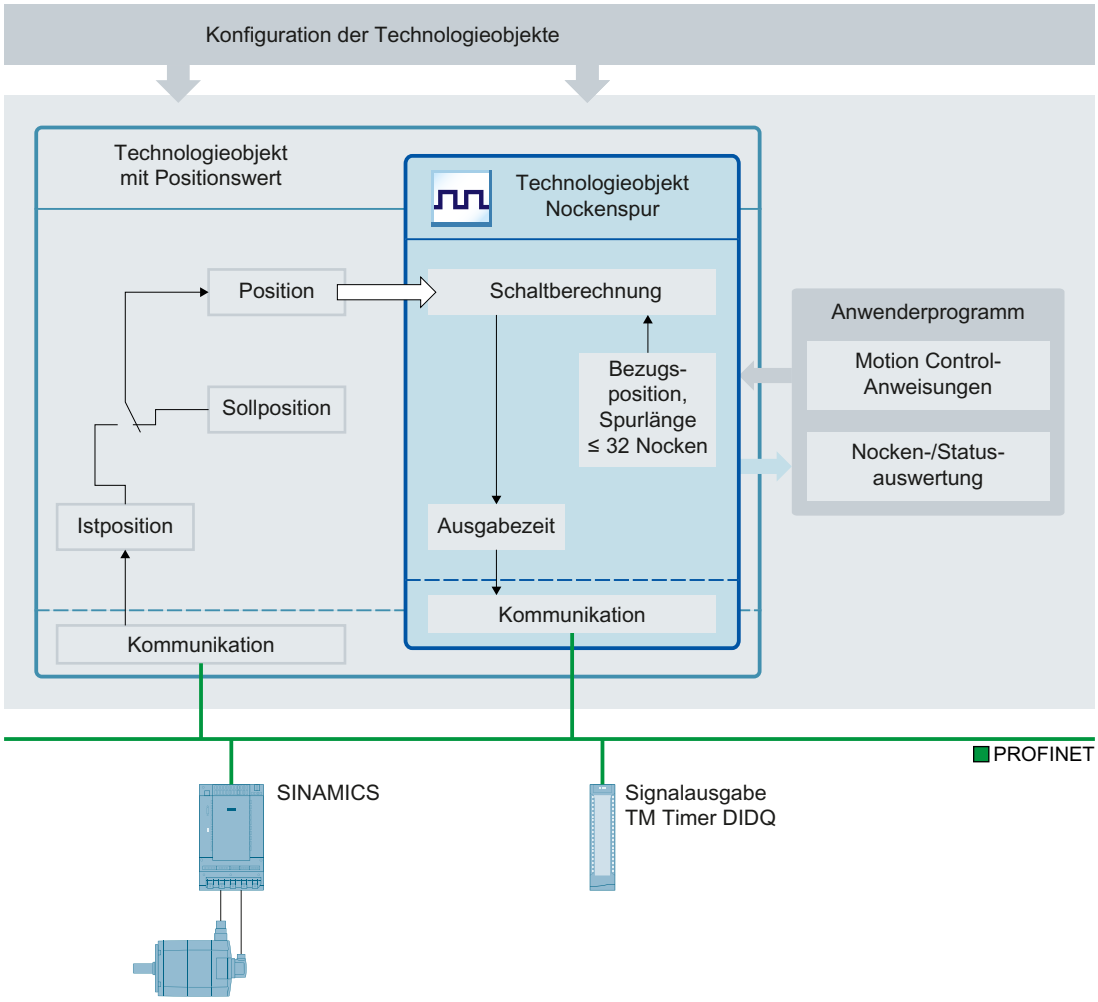

Definition Nockenspur

Eine Nockenspur besteht aus bis zu 32 einzelnen Nocken, die innerhalb einer einstellbaren Spurlänge vorgegeben werden.

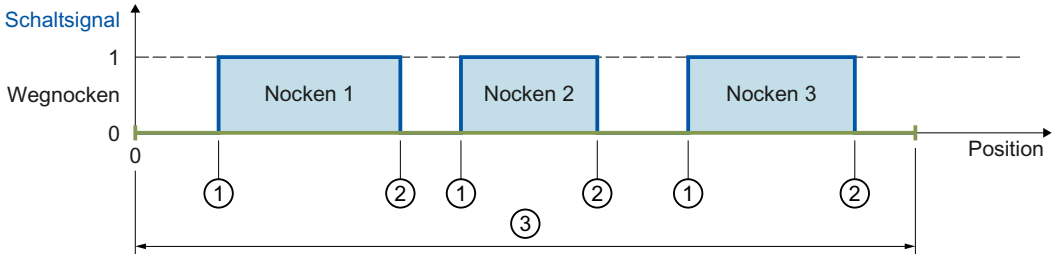

① Anfangsposition
② Endposition
③ Nockenspurlänge

Die Nockenpositionen werden bezogen auf die Nockenspur definiert. Der Nockenspuranfang ist immer 0.0. Die Nockenpositionen auf der Nockenspur sind damit immer positiv.

Bei der Bearbeitung der Nockenspur werden Nocken mit einer Anfangsposition innerhalb der Spurlänge berücksichtigt.

Die Nocken der Nockenspur können als Wegnocken oder Zeitnocken eingestellt werden, wobei in einer Nockenspur nur einer der beiden Nockentypen verwendet werden kann.

Wenn das Ende einer Nockenspur überfahren wird, verhalten sich innerhalb der Nockenspur geschaltete Nocken wie folgt:

- Zeitnocken bleiben für die eingestellte Einschaltdauer weiterhin eingeschaltet.
- Wegnocken, deren Endposition außerhalb der Nockenspur liegt, werden mit Verlassen der Nockenspur abgeschaltet.

Nocken, deren Anfangs- und Endposition außerhalb der Nockenspurlänge liegen, werden nicht berücksichtigt. Sie werden erst wirksam, wenn die Nockenspurlänge vergrößert wird, sodass mindestens die jeweilige Anfangsposition eines Nockens innerhalb der neuen Spurlänge liegt.

Zuordnung

Das Technologieobjekt Nockenspur benötigt immer eine Zuordnung zu einem anderen Technologieobjekt, dessen Position ausgewertet wird.

Das Technologieobjekt Nockenspur kann den folgenden Technologieobjekten zugeordnet werden:

- Gleichlaufachse
- Positionierachse
- Externer Geber

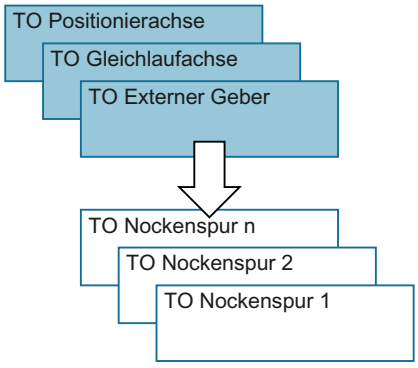

Einer Nockenspur kann genau eine Achse oder ein Externer Geber zugeordnet werden.

Einer Achse oder einem Externer Geber können mehrere Nockenspuren zugeordnet werden.

Positionsbezug

Die Schaltpunkte der Nocken einer Nockenspur können sich abhängig vom verschalteten Technologieobjekt auf die folgenden Positionen beziehen.

- Istposition einer Gleichlauf-/Positionierachse
- Sollposition einer Gleichlauf-/Positionierachse
- Position eines Externen Gebers

Abbildung der Nockenspur auf die Position des Technologieobjekts

Der Nockenspuranfang wird auf die angegebene Bezugsposition des verschalteten Technologieobjekts gelegt. Somit ergeben sich die Schaltpositionen aus den ab der Bezugsposition auf das verschaltete Technologieobjekt abgebildeten Nockenspurpositionen. Die Nockenspur wird in beide Richtungen des verschalteten Technologieobjekts fortgesetzt.

Die Bezugsposition kann positiv als auch negativ gesetzt werden.

Beispiel

- Achsbereich = -1000 mm bis +1000 mm
- Gewünschte Schaltpunkte des Nockens mit Bezug auf Achsposition:
  - Anfangsposition = -200 mm
  - Endposition = -100 mm
- Nockenspurlänge = 2000 mm
- Definition des Nockens auf der Spur:
  - Anfangsposition = 800 mm
  - Endposition = 900 mm

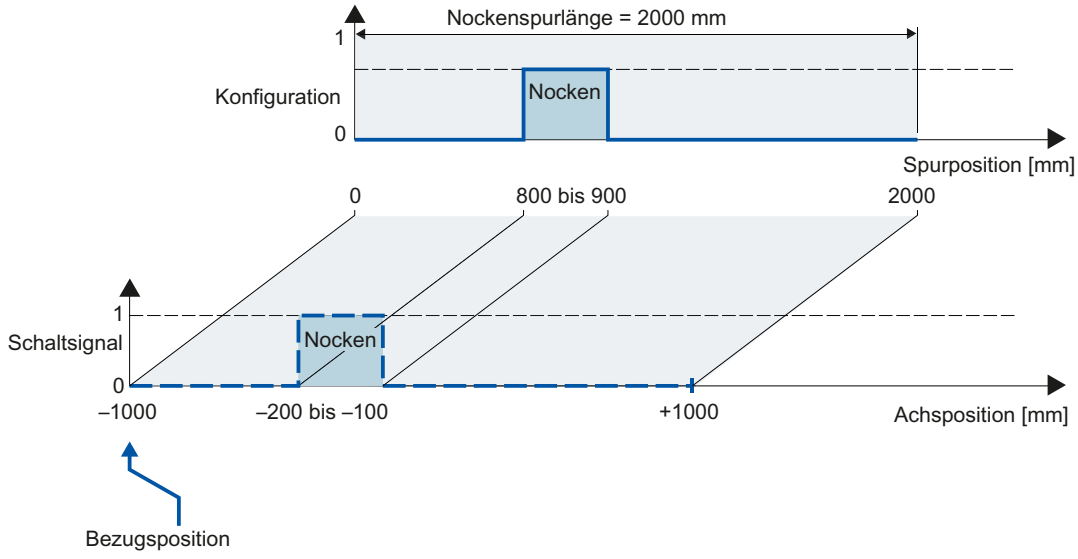

Bearbeitung einer Nockenspur

Die Bearbeitung einer Nockenspur erfolgt zyklisch.

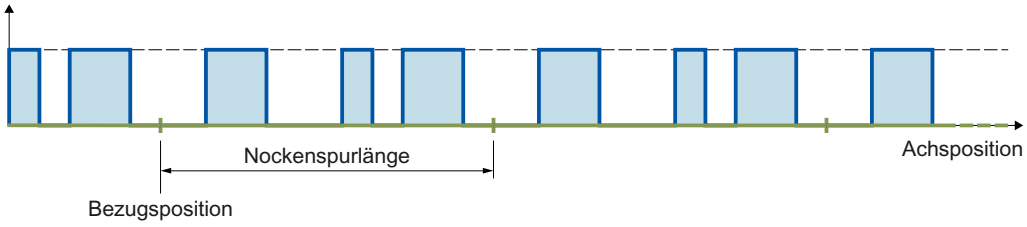

Die Nockenspur wird auf die Position des verschalteten Technologieobjekts ab der Bezugsposition abgebildet und in beide Richtungen zyklisch fortgesetzt.

Ausgabe einer Nockenspur

Folgende Ausgabemöglichkeiten stehen für den digitalen Nockenspurausgang zur Verfügung:

- TM Timer DIDQ
  Digitalausgang mit hoher Genauigkeit und Reproduzierbarkeit im Mikrosekundenbereich auf Time-based IO-Modulen ET 200MP TM Timer DIDQ 16x24V und ET 200SP TM Timer DIDQ 10x24V
  Die Ausgabe über TM Timer DIDQ setzt zwingend den taktsynchronen Betrieb voraus. Der taktsynchrone Betrieb ist nur mit PROFINET Interface-Modulen möglich.
- Digitalausgabemodul
  Digitalausgang mit Schaltgenauigkeit abhängig vom Ausgabezyklus der verwendeten Peripherie

Pro Servotakt können maximal zwei Flanken (über TM Timer DIDQ, steigend und fallend) bzw. eine Flanke (über Digitalausgabemodul, steigend oder fallend) ausgegeben werden. Werden in einem Servotakt mehrere Einschaltflanken oder mehrere Ausschaltflanken abgesetzt, sind die jeweils zuletzt geschriebenen Werte gültig.

Maskierung einzelner Nocken einer Nockenspur

Damit Nocken bearbeitet werden, müssen sie am Technologie-Datenbaustein mit "<TO>.Parameter.Cam[1 … 32].Existent" = TRUE als gültig konfiguriert werden. Zusätzlich können Nocken einer Nockenspur über die Bitmaskierung ("<TO>.Parameter.CamMasking") im Anwenderprogramm als gültig definiert werden. In der Standardeinstellung sind alle gültigen Nocken freigegeben ("<TO>.Parameter.CamMasking" = 0xFFFFFFFF). Die Nockenspur selbst wird über die Motion Control-Anweisung "MC_CamTrack" aktiviert/deaktiviert.

Modulofunktion

Spurlänge und Abbildung auf eine Achse oder einen Externen Geber mit Modulofunktion

Bei der Abbildung einer Nockenspur auf eine Achse mit Modulofunktion wird eine außerhalb des Modulobereichs angegebene Bezugsposition in den Modulobereich abgebildet.

Die Spurlänge kann kleiner oder größer als die Modulolänge der Achse sein. Damit die Nockenspur ohne Versatz in den Modulobereich abgebildet wird und keine unerwünschten

Überlagerungen entstehen, ist ein ganzzahliges Verhältnis von Modulolänge zu Spurlänge und umgekehrt, erforderlich.

Beispiel Abbildung ohne Versatz

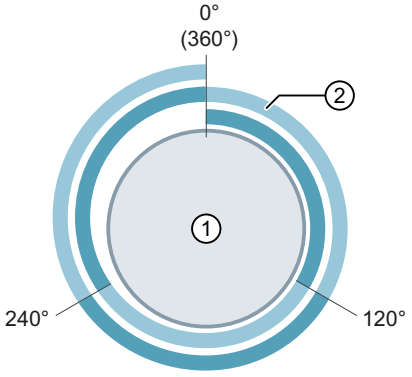

① Achse mit Modulolänge 360°
② Nockenspur mit Spurlänge 120°
⇒ Verhältnis = 360° / 120° = **3**
Die Nockenspur wird 3 Mal auf einer Modulolänge ausgegeben.

Beispiel Abbildung mit Versatz

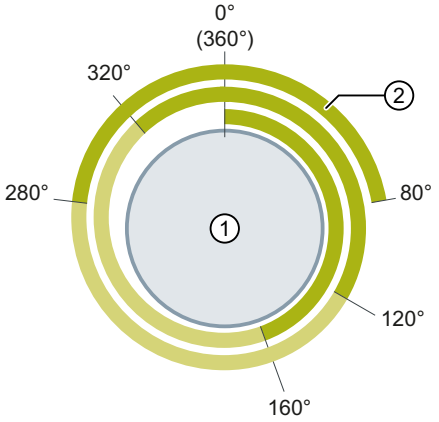

① Achse mit Modulolänge 360°
② Nockenspur mit Spurlänge 160°
⇒ Verhältnis = 360° / 160° = **2,25**
Die Nockenspur wird 2,25 Mal auf der ersten Modulolänge ausgegeben und in den weiteren Modulolängen entsprechend fortgesetzt.

Bei zyklischer Bearbeitung der Nockenspur wird die fortgeführte Bezugsposition der aktuellen Nockenspur in der Variable "<TO>.MatchPosition" angezeigt. Die fortgeführte Bezugsposition ist richtungsunabhängig, immer die Position des linken Randes der Nockenspur. Die eindeutige Erfassung und Ausgabe der Position ist nur möglich, wenn das zugeordnete Technologieobjekt in Bewegung ist. Der Abstand zur aktuellen Bezugsposition der aktuellen Nockenspur (<TO>.MatchPosition) wird in der Variable "<TO>.TrackPosition" angezeigt.

Beim Freigeben der Nockenspur durch einen "MC_CamTrack"-Auftrag wird die Lage der Nockenspur (<TO_CamTrack>.MatchPosition) so gesetzt, als wäre die aktuelle Position ausgehend von der Bezugsposition (<TO_CamTrack>.ReferencePosition) durch eine Bewegung in positiver Richtung erreicht worden.

Referenzieren einer Achse oder eines Gebers ohne Modulofunktion

Eine Änderung der Position einer Achse oder des Externen Gebers über die Motion Control Anweisung "MC_Home" wird als Positionssprung gewertet. Eine freigegebene Nockenspur wird auf die umgesetzte Position bezogen und von dort weiter bearbeitet.

- Das Referenzieren wirkt sich auf die aktuelle Position der Nockenspur (<TO>.TrackPosition) aus. Diese wird aufgrund der Verschiebung auf kürzestem Weg neu gebildet.
- Wegnocken werden möglicherweise übersprungen oder werden entsprechend ausgegeben.

- Zeitnocken werden übersprungen. Erst mit Überfahren der Anfangsposition wird ein Zeitnocken geschaltet und bleibt für die Einschaltdauer eingeschaltet.
- Geschaltete Zeitnocken werden durch eine Referenzierung nicht abgebrochen.

Empfehlung:

Sperren Sie die Nockenspur vor oder während des Referenzierens.

Referenzieren einer Achse oder eines Gebers mit Modulofunktion

- Das Referenzieren wirkt sich auf die aktuelle Position der Nockenspur (<TO>.TrackPosition) aus.
  Durch das Referenzieren des zugeordneten Technologieobjekts wird die Lage der danach aktiven Nockenspur richtungsabhängig beeinflusst. Dies ist abhängig von der Positionsdifferenz, der Differenz zwischen der neuen Position minus der ursprünglichen Position. Wenn die Positionsdifferenz negativ ist, addieren Sie die Modulolänge (<TO_Axis/TO_Encoder>.Modulo.Length).
  Wenn diese Positionsdifferenz kleiner gleich der Hälfte der Modulolänge ist, wird die neue Lage der Nockenspur (<TO_CamTrack>.MatchPosition) so gesetzt, als wäre die neue Position ausgehend von der ursprünglichen Position durch eine Bewegung in positiver Richtung erreicht worden.
  Wenn diese Positionsdifferenz größer als die Hälfte der Modulolänge ist, wird die neue Lage der Nockenspur (<TO_CamTrack>.MatchPosition) so gesetzt, als wäre die neue Position ausgehend von der ursprünglichen Position durch eine Bewegung in negativer Richtung erreicht worden.
- Wegnocken werden möglicherweise übersprungen oder werden entsprechend ausgegeben.
- Zeitnocken werden übersprungen. Erst mit Überfahren der Anfangsposition wird ein Zeitnocken geschaltet und bleibt für die Einschaltdauer eingeschaltet.
- Geschaltete Zeitnocken werden durch eine Referenzierung nicht abgebrochen.

Empfehlung:

Sperren Sie die Nockenspur vor oder während des Referenzierens.

Wirkrichtung

Die Nockenspur ist immer für beide Richtungen der Position des verschalteten Technologieobjekts aktiv.

Ausgabe einer Nockenspur mit Wegnocken

Die Wegnocken werden beim Überfahren des Einschaltbereichs geschaltet.

Die folgende Grafik zeigt das Ausführen einer Nockenspur mit Wegnocken abhängig von der Bewegungsrichtung der Achse.

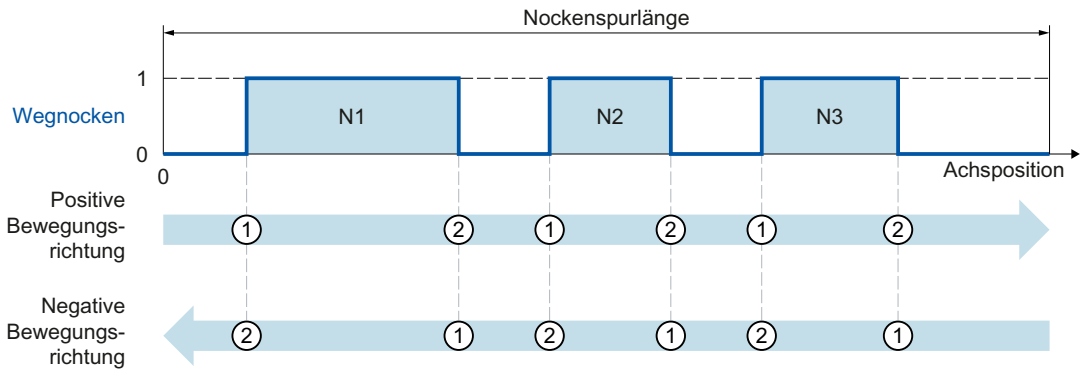

Bei positiver Bewegungsrichtung werden die Nocken der Nockenspur in der Reihenfolge Nocken N1, Nocken N2, Nocken N3 ausgegeben. Bei negativer Bewegungsrichtung werden die Nocken der Nockenspur in der Reihenfolge Nocken N3, Nocken N2, Nocken N1 ausgegeben. Die Wegnocken schalten ein bei ① und schalten aus bei ②.

Ausgabe einer Nockenspur mit Zeitnocken

Die Zeitnocken werden beim Überfahren der Anfangsposition geschaltet.

Die folgende Grafik zeigt das Ausführen einer Nockenspur mit Zeitnocken abhängig von der Bewegungsrichtung der Achse.

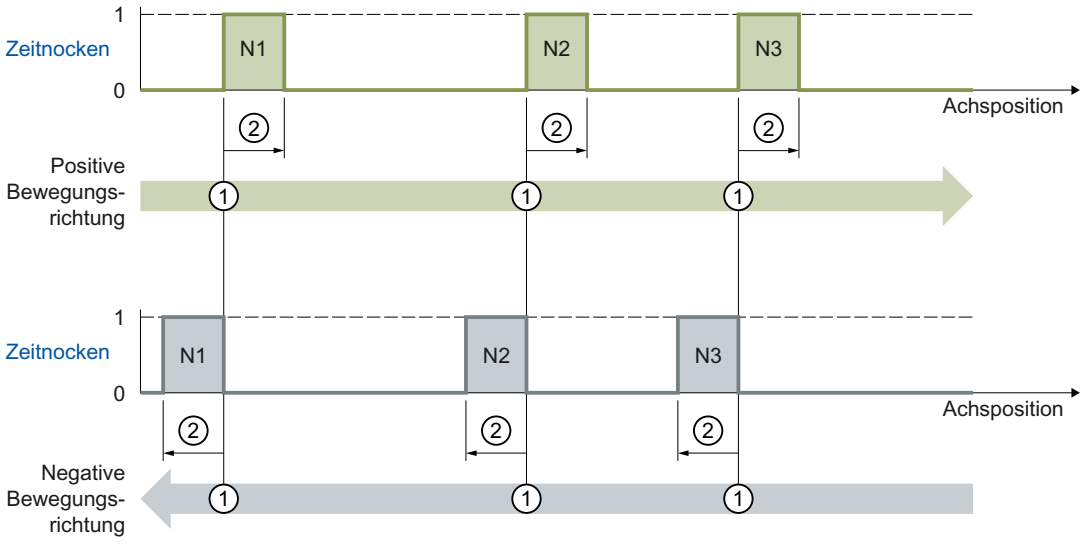

Bei positiver Bewegungsrichtung werden die Nocken der Nockenspur in der Reihenfolge Nocken N1, Nocken N2, Nocken N3 ausgegeben. Bei negativer Bewegungsrichtung werden die Nocken der Nockenspur in der Reihenfolge Nocken N3, Nocken N2, Nocken N1 ausgegeben. Die Zeitnocken schalten ein bei ① und bleiben für die eingestellte Einschaltdauer ② eingeschaltet.

Ändern der Nockenspurdaten im Betrieb

Die Daten einer Nockenspur und die Parameter der dazugehörigen Motion Control-Anweisung "MC_CamTrack" können während einer freigegebenen Spurbearbeitung geändert werden. Dabei wird die aktive Motion Control-Anweisung "MC_CamTrack" nicht abgebrochen. Die geänderten Parameter werden jedoch erst mit dem nächsten Aufruf der Motion Control-Anweisung "MC_CamTrack" wirksam.

Folgende Parameter können im Betrieb geändert werden und sind nach dem erneuten Aufruf der Motion Control-Anweisung "MC_CamTrack" wirksam:

- Daten Nockenspur am Technologie-Datenbaustein
  - Bezugsposition (<TO>.Parameter.ReferencePosition)
  - Spurlänge (<TO>.Parameter.CamTrackLength)
  - Bitmaskierung der Einzelnocken (<TO>.Parameter.CamMasking)
  - Aktivierungszeit (<TO>.Parameter.OnCompensation)
  - Deaktivierungszeit (<TO>.Parameter.OffCompensation)
  - Hysteresewert (<TO>.Parameter.Hysteresis)
  - Nockendaten (<TO>.Parameter.Cam[1 … 32])
- Parameter am Funktionsbaustein
  - Freigabe (MC_CamTrack.Enable)
  - Modus (MC_CamTrack.Mode)
  - Invertierte Ausgabe (MC_CamTrack.InvertOutput)

Berücksichtigen Sie bei einer Änderung der Nockenspurdaten das unterschiedliche Aktivierungsverhalten (Seite 1816).

Aktivierungsverhalten

Durch den Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE wird eine Nockenspur aktiviert. Dabei ist zu unterscheiden zwischen:

- Erstmaliger Aktivierung der Nockenspur
- Aufruf nach einer Änderung der Nockenspurdaten während der aktiven Nockenspurbearbeitung

Der Unterschied zwischen den beiden Arten liegt im Verhalten bei der Übernahme der Nockenspurdaten. Je nach eingestelltem Mode ("MC_CamTrack.Mode") erfolgt die

Übernahme der Konfiguration (Nockenspurdaten, Daten am Funktionsbaustein) zu unterschiedlichen Zeitpunkten.

- **Erstmaliges Einschalten einer Nockenspur**
  Durch den Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE wird die Nockenspur sofort aktiviert ("<TO>.Status" wechselt auf 1) und konfigurierte Nockenspurdaten sind sofort wirksam. Dieses Verhalten ist bei eingestelltem "MC_CamTrack.Mode" = 0 und "MC_CamTrack.Mode" = 1 gleich.
- **Änderung der Nockenspurdaten einer bereits aktivierten Nockenspur** ("<TO>.Status" = 1)
  - Durch den Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE und "Mode" = 0 werden die geänderten Nockenspurdaten sofort wirksam.
    Bereits angesteuerte Wegnocken werden abgebrochen, wenn deren Spursignale durch die geänderten Nockenspurdaten nicht weiterhin angesteuert werden. Bereits angesteuerte Zeitnocken werden in jedem Fall abgebrochen.
  - Erfolgt der Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE und "Mode" = 1, wird die Nockenspur bis Nockenspurende weiterhin mit vorheriger Konfiguration ausgegeben. Geänderte Nockenspurdaten werden mit dem Ende des aktuellen Spurzyklus wirksam.
    Wenn Sie eine Nockenspur mit "MC_CamTrack.Mode" = 1 zur Laufzeit des Anwenderprogramms ändern, berücksichtigen Sie die Vorhaltezeit der Nockenspur als Reserve für den ersten Nocken. Definieren Sie die erste Nockenposition in der Nockenspur erst nach folgender Position:
    Position erster Nocken > Geschwindigkeit der Achse × Vorhaltezeit der Nockenspur (<TO>.Parameter.OnCompensation)
    Berücksichtigen Sie dabei ebenfalls die systeminterne Zeit zur Nockenberechnung, auch wenn Sie die Vorhaltezeit 0.0 einstellen.

Ändern von Nockenspurdaten, wenn die Modulolänge kein Vielfaches der Spurlänge ist

Damit die Schaltzeitpunkte korrekt gesetzt werden, wird die folgende Vorgehensweise für Änderungen empfohlen:

- Geben Sie Änderungen möglichst frühzeitig, nach Beginn einer neuen Nockenspur ein
- Geben Sie eine neue Bezugsposition (<TO>.Parameter.ReferencePosition) für die geänderte Nockenspur ein. Die neue Bezugsposition setzt sich richtungsabhängig wie folgt zusammen:
  - Positive Wirkrichtung: Aktuelle Bezugsposition (<TO>.MatchPosition) + Nockenspurlänge (<TO>.Parameter.CamTrackLength)
  - Negative Wirkrichtung: Aktuelle Bezugsposition (<TO>.MatchPosition) - Nockenspurlänge (<TO>.Parameter.CamTrackLength)
- Geben Sie Änderungen der Nockenspur mit dem Aufruf der Motion Control-Anweisung "MC_CamTrack" und "Mode" = 1 aus.

Beispiel

Die nachfolgende Abbildung zeigt beispielhaft die Unterschiede im Aktivierungsverhalten.

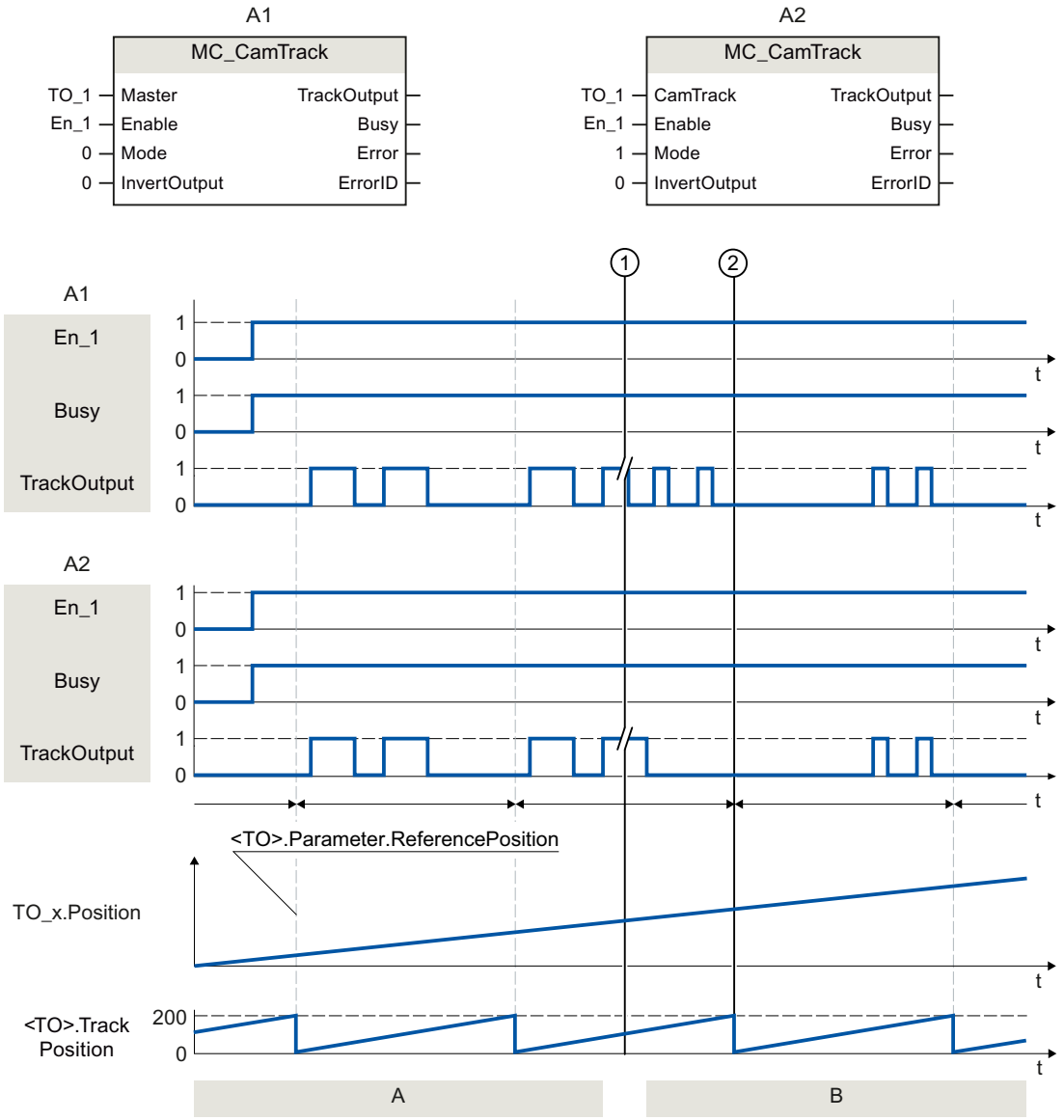

A1	A	Die Nockenspur wird erstmalig mit "MC_CamTrack.Enable" = TRUE aktiviert und die Nocken werden mit eingestelltem "MC_CamTrack.Mode" = 0 sofort ausgegeben.
	B	Nachdem Nockenspurdaten geändert wurden (①), wird die Nockenspur durch Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE aktiviert und die geänderten Daten sind mit eingestelltem "MC_CamTrack.Mode" = 0 sofort wirksam (①).
A2	A	Die Nockenspur wird erstmalig mit "MC_CamTrack.Enable" = TRUE aktiviert und die Nocken werden mit eingestelltem "MC_CamTrack.Mode" = 1 sofort ausgegeben.
	B	Nachdem Nockenspurdaten geändert wurden (①), wird die Nockenspur durch Aufruf der Motion Control-Anweisung "MC_CamTrack" mit "Enable" = TRUE aktiviert und die geänderten Daten sind mit eingestelltem "MC_CamTrack.Mode" = 1 mit dem Ende des aktuellen Spurzyklus wirksam (②).

Hysterese

Die Einstellung der Hysterese erfolgt am Technologieobjekt Nockenspur. Das Verhalten und Auswirkung der eingestellten Hysterese entspricht der Hysterese (Seite 1803) am Technologieobjekt Nocken.

Zeitlicher Versatz von Nockenschaltpunkten

Mithilfe der Aktivierungs- bzw. Deaktivierungszeit am Technologieobjekt Nockenspur können Schaltzeiten des Ausgangs und des angeschlossenen Aktors (z. B. Ventil) ausgeglichen werden.

Der zeitliche Versatz von Nockenschaltpunken entspricht der Aktivierungs- bzw. Deaktivierungszeit (Seite 1805) am Technologieobjekt Nocken.

Variablen

Folgende Variablen des Technologieobjekts sind relevant:

Statusanzeige		
Variable	**Beschreibung**	
<TO>.Status	Status der Nockenspurfunktion	
	0	Inaktiv
	1	Aktiv
	2	Aktiv und wartet auf nächste Spur
<TO>.TrackOutput	Ein Nocken der Nockenspur ist geschaltet.	
<TO>.SingleCamState	Eingeschalteter Nocken (Bitmaskiert)	
<TO>.TrackPosition	Anzeige der aktuellen Position innerhalb der Nockenspur Der Abstand zur aktuellen Bezugsposition der aktuellen Nockenspur (<TO>.MatchPosition) wird angezeigt.	
<TO>.MatchPosition	Bezugsposition der aktuellen Nockenspur Bei zyklischer Bearbeitung der Nockenspur wird die fortgeführte Bezugsposition der aktuellen Nockenspur angezeigt. Die eindeutige Erfassung und Ausgabe der Position ist nur möglich, wenn das zugeordnete Technologieobjekt in Bewegung ist.	

Parameter		
Variable	**Beschreibung**	
<TO>.Parameter.CamTrackType	Nockentyp	
	0	Wegnocken
	1	Zeitnocken
<TO>.Parameter.PositionType	Positionsbezug	
	0	Sollposition
	1	Istposition
<TO>.Parameter.ReferencePosition	Bezugsposition	
<TO>.Parameter.CamTrackLength	Spurlänge	
<TO>.Parameter.CamMasking	Bitmaskierung der Einzelnocken	

Parameter		
Variable	**Beschreibung**	
<TO>.Parameter.OnCompensation	Aktivierungszeit (Vorhaltezeit an der Einschaltflanke)	
<TO>.Parameter.OffCompensation	Deaktivierungszeit (Vorhaltezeit an der Ausschaltflanke)	
<TO>.Parameter.Hysteresis	Hysteresewert	
<TO>.Parameter.Cam[1..32].OnPosition	Startposition (Wegnocken und Zeitnocken)	
<TO>.Parameter.Cam[1..32].OffPosition	Endposition (Wegnocken)	
<TO>.Parameter.Cam[1..32].Duration	Einschaltdauer (Zeitnocken)	
<TO>.Parameter.Cam[1..32].Existent	Gültigkeit eines Nockens	
	FALSE	Nocken wird nicht verwendet.
	TRUE	Nocken wird verwendet.

Interface		
Variable	**Beschreibung**	
<TO>.Interface.EnableOutput	Nockenausgabe an dem unter "Address" angegebenen Bit	
	FALSE	Keine Ausgabe
	TRUE	Ausgabe
<TO>.Interface.Address	Peripherieadresse für digitalen Nockenausgang	

Units	
Variable	**Beschreibung**
<TO>.Units.LengthUnit	Einheit der Längenangabe
<TO>.Units.TimeUnit	Einheit der Zeitangabe

StatusWord	
Variable	**Beschreibung**
<TO>.StatusWord.X0 (Control)	Das Technologieobjekt ist in Betrieb.
<TO>.StatusWord.X1 (Error)	Am Technologieobjekt ist ein Fehler aufgetreten.
<TO>.StatusWord.X2 (RestartActive)	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert.
<TO>.StatusWord.X3 (OnlineStartValuesChanged)	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen.
<TO>.StatusWord.X4 (OutputInverted)	Die Nockenausgabe erfolgt invertiert.
<TO>.StatusWord.X5 (CommunicationOk)	Die Nockenspur ist mit dem Ausgabemodul synchronisiert und kann verwendet werden.
<TO>.StatusWord.X6 (CamDataChanged)	Die Daten der Einzelnocken wurden verändert, aber noch nicht mit der Motion Control-Anweisung "MC_CamTrack" wirksam gesetzt.

ErrorWord	
Variable	**Beschreibung**
<TO>.ErrorWord.X0 (SystemFault)	Ein systeminterner Fehler ist aufgetreten.
<TO>.ErrorWord.X1 (ConfigFault)	Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert.
<TO>.ErrorWord.X2 (UserFault)	Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung.
<TO>.ErrorWord.X3 (CommandNotAccepted)	Befehl nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
<TO>.ErrorWord.X13 (PeripheralError)	Fehler beim Zugriff auf eine logische Adresse.

ErrorDetail		
Variable	**Beschreibung**	
<TO>.ErrorDetail.Number	Alarmnummer	
<TO>.ErrorDetail.Reaction	Wirksame Alarmreaktion	
	0	Keine Reaktion
	5	Nockenspurbearbeitung wird beendet

Technologieobjekt Kurvenscheibe

Das Technologieobjekt Kurvenscheibe definiert eine Übertragungsfunktion y = f(x). In dieser Übertragungsfunktion wird einheitenneutral die Abhängigkeit einer Ausgangsgröße von einer Eingangsgröße beschrieben. Ein Technologieobjekt Kurvenscheibe lässt sich mehrfach anwenden.

Eine Übersicht über die Funktionen des Technologieobjekts Kurvenscheibe finden Sie im Kapitel "Funktionen (Seite 1683)".

Die Funktion y = f(x) definieren Sie in der Konfiguration des Technologieobjekts (Seite 1919) über Stützpunkte und/oder Segmente. Bereiche zwischen Stützpunkten bzw. Segmenten werden über die Motion Control-Anweisung "MC_InterpolateCam" interpoliert. Zur Laufzeit des Anwenderprogramms lassen sich die Einstellungen über den Technologie-Datenbaustein gemäß Anhang "Variablen des Technologieobjekts Kurvenscheibe (Seite 2176)" ändern/neu definieren.

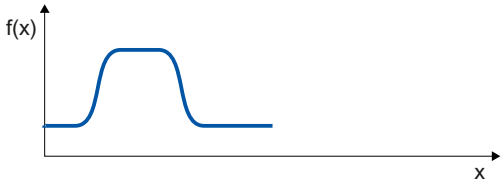

Eine interpolierte Kurvenscheibe lässt sich als Gleichlauffunktion für einen Kurvenscheibengleichlauf (Seite 1773) anwenden.

Das folgende Bild zeigt die prinzipielle Funktionsweise des Technologieobjekts Kurvenscheibe:

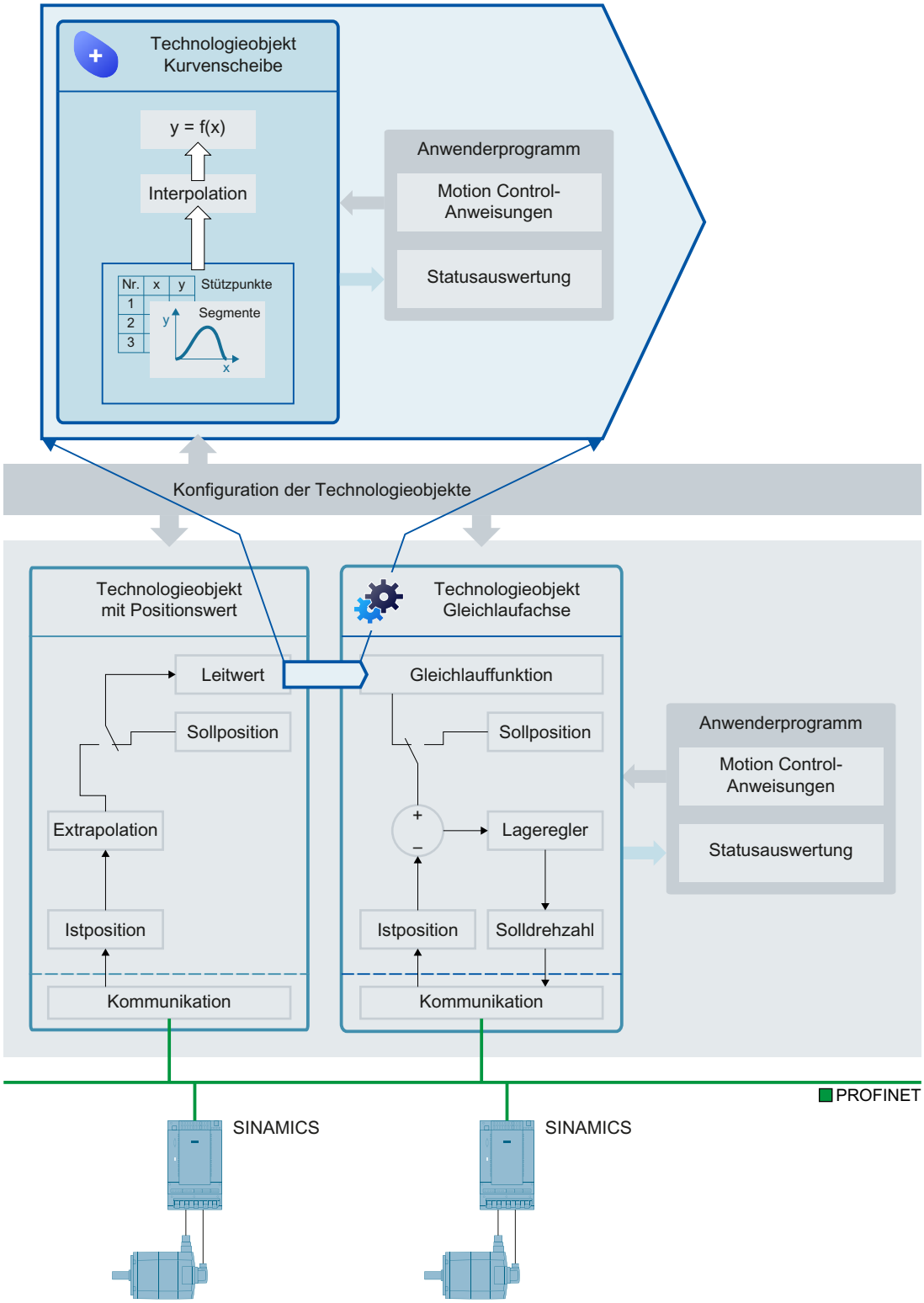

Interpolation

Um eine Kurvenscheibe im Anwenderprogramm zu verwenden, müssen Sie die Kurvenscheibe nach dem Laden in die CPU interpolieren. Die Interpolation schließt die Lücken zwischen den definierten Stützpunkten und Segmenten der Kurvenscheibe. Die Kurvenscheibe wird vom minimalen Wert im Leitwertbereich bis zum maximalen Wert interpoliert. Der minimale Wert im Leitwertbereich ist der erste definierte Stützpunkt/der Beginn des ersten Segments der Kurvenscheibe (<TO>.StatusCam.StartLeadingValue). Der maximale Wert im Leitwertbereich ist der letzte definierte Stützpunkt/das Ende des letzten Segments der Kurvenscheibe (<TO>.StatusCam.EndLeadingValue).

Eine Kurvenscheibe interpolieren Sie in ihrem Anwenderprogramm mit der Motion Control-Anweisung "MC_InterpolateCam". Wenn eine Kurvenscheibe interpoliert ist, wird dies über den Parameter "MC_InterpolateCam.Done" = TRUE und über die Variable "<TO>.StatusWord.X5 (Interpolated)" = 1 im Technologie-Datenbaustein angezeigt.

Die Interpolation stellen Sie in der Konfiguration des Technologieobjekts (Seite 1919) ein. Folgende Methoden sind möglich:

- Systeminterpolation
- Optimierung von Übergängen gemäß VDI-Richtlinie 2143

Systeminterpolation

Bei der Systeminterpolation werden die Übergänge gemäß der Interpolationsart und dem Verhalten in den Randpunkten des Übergangssegments interpoliert. Folgende Interpolationsarten sind möglich:

- **Lineare Interpolation**
  Lücken im Kurvenverlauf werden über eine Gerade geschlossen.

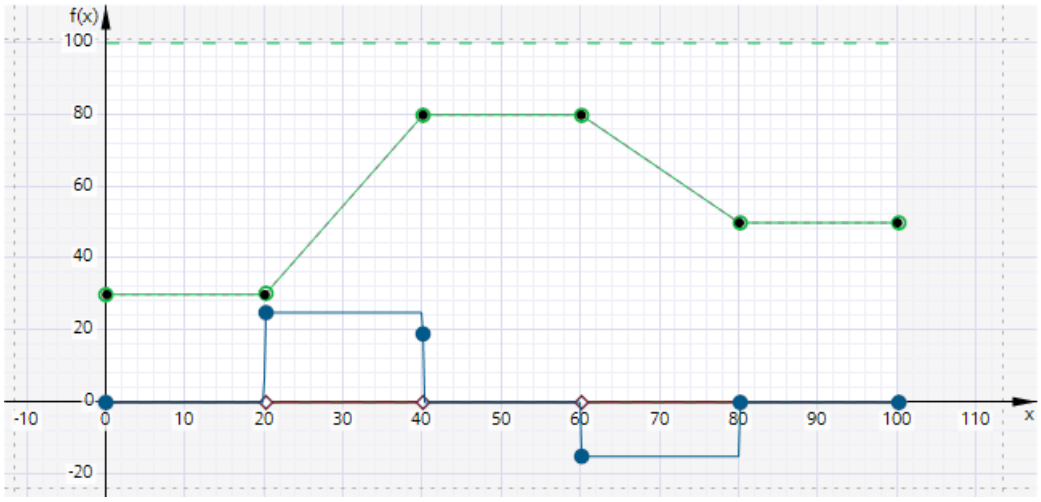

Vorgegebene Position (Punkt)
Interpolierte Position
Resultierende Geschwindigkeit (skaliert)
Resultierende Beschleunigung
Resultierender Ruck

- **Interpolation mit kubischen Splines**
  Der interpolierte Kurvenverlauf läuft durch die Stützpunkte bzw. Segmente der Kurve. Nach Abschluss der Interpolation kann der Wertebereich der Kurvenscheibe größer sein als vor der Interpolation.

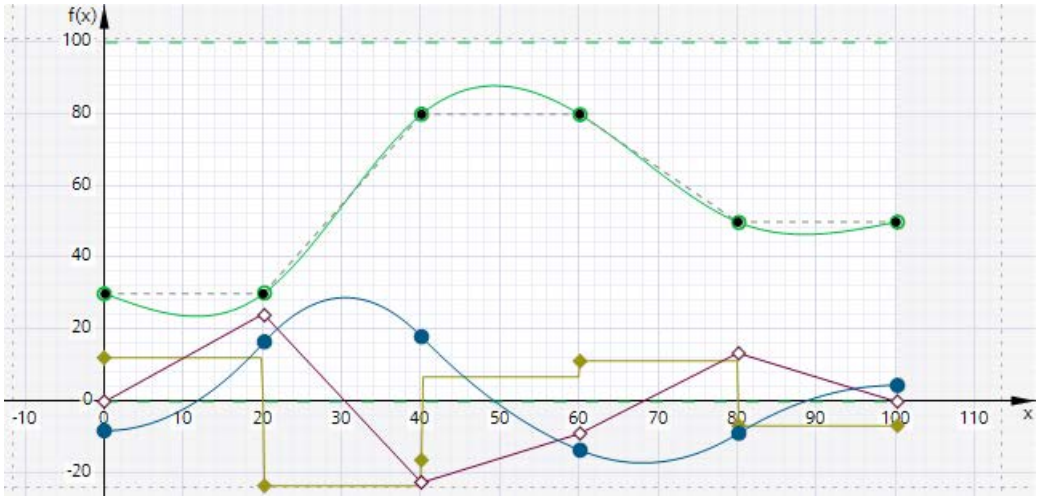

Vorgegebene Position (Punkt)
Interpolierte Position
Resultierende Geschwindigkeit (skaliert)
Resultierende Beschleunigung (skaliert)
Resultierender Ruck (skaliert)

- **Interpolation mit Bézier-Splines**
  Der interpolierte Kurvenverlauf läuft entlang der Stützpunkte bzw. Kurvensegmente. Der Wertebereich der Kurvenscheibe ändert sich durch die Interpolation nicht.

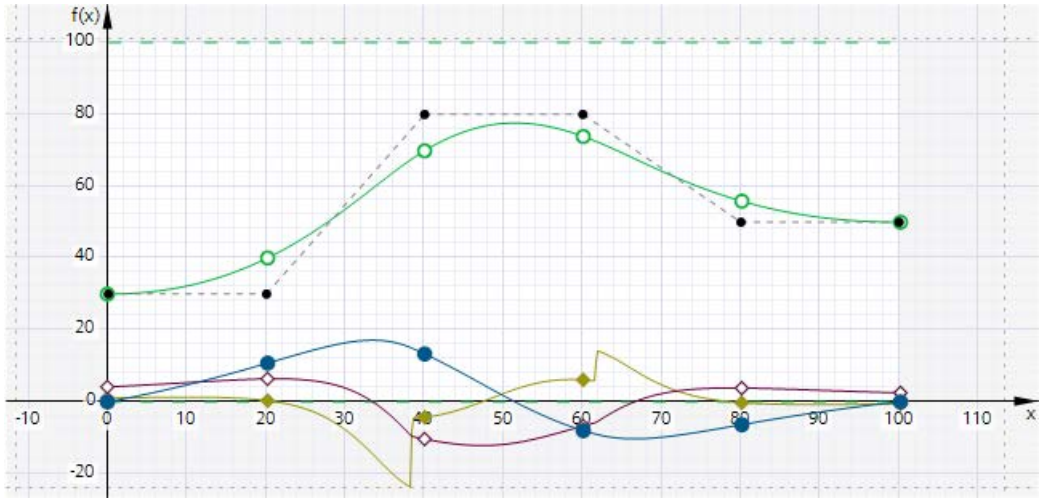

Vorgegebene Position (Punkt)
Interpolierte Position
Resultierende Geschwindigkeit (skaliert)
Resultierende Beschleunigung (skaliert)
Resultierender Ruck (skaliert)

Optimierung von Übergängen gemäß VDI-Richtlinie 2143

Die Übergänge werden entsprechend der Bewegungsaufgabe und den Optimierungseinstellungen gemäß VDI-Richtlinie 2143 vorgegeben.

Beachten Sie, dass die Optimierung von Übergangen gemäß VDI-Richtlinie 2143 im Gegensatz zur Systeminterpolation direkt Segmente im Technologieobjekt-Datenbaustein belegt. Diese Optimierungsart ist somit über "MC_InterpolateCam" während der Laufzeit nicht möglich.

Technologieobjekt Kinematik

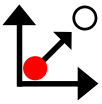

Das Technologieobjekt Kinematik wird beschrieben im Funktionshandbuch "S7-1500T Kinematikfunktionen".

6.1.2.3 Versionen einsetzen

Versionsübersicht

Bei S7-1500 Motion Control wird unterschieden zwischen der Version der Technologie, der Technologieobjekte und der Motion Control-Anweisungen. Die nachfolgend dargestellte Übersicht umfasst die S7-1500 und S7-1500T. Auf einer CPU kann jeweils nur eine Technologieversion betrieben werden.

Beim Wechsel auf eine CPU ≥ V1.6 müssen Sie die Technologieversion entsprechend ändern. Der Kartentausch von einer CPU < V1.6 zu einer CPU ≥ V1.6 wird unterstützt. Im TIA Portal können Sie mit einer CPU ≥ V1.6 nur noch Projekte mit der entsprechend höheren Technologieversion bearbeiten.

Die Technologieversion können Sie auf zwei Wegen ändern:

- Version der Motion Control-Anweisungen ändern
  Die Version der Motion Control-Anweisungen ändern Sie in der Task Card "Anweisungen" im Ordner "Technologie > Motion Control > S7-1500 Motion Control".
  Wenn die eingesetzte Version der Motion Control-Anweisung nicht der Kompatibilitätsliste entspricht, werden die entsprechenden Motion Control-Anweisungen im Programmiereditor rot markiert.
- Technologieobjekt mit einer alternativen Version hinzufügen
  Wenn Sie im Dialog "Neues Objekt hinzufügen" ein Technologieobjekt mit einer alternativen Version hinzufügen, wird die Technologieversion auf die alternative Version geändert.

Die Technologieobjekte und Motion Control-Anweisungen werden erst beim Übersetzen entsprechend der gewählten Version der Technologie konvertiert.

Die Version eines Technologieobjekts bzw. einer Motion Control-Anweisung wird in den Eigenschaften des Technologieobjekts im Register "Allgemein > Information" im Feld "Version" angezeigt.

Kompatibilitätsliste

Die folgende Tabelle zeigt die Kompatibilität der Version der Technologie mit der Version der CPU:

CPU	Technologie	Technologieobjekt
V2.5	V4.0	Drehzahlachse V4.0 Positionierachse V4.0 Externer Geber V4.0 Gleichlaufachse V4.0 Messtaster V4.0 Nocken V4.0 Nockenspur V4.0 Kurvenscheibe V4.0 (S7-1500T) Kinematik V4.0 (S7-1500T)
V2.0, V2.1	V3.0	Drehzahlachse V3.0 Positionierachse V3.0 Externer Geber V3.0 Gleichlaufachse V3.0 Messtaster V3.0 Nocken V3.0 Nockenspur V3.0 Kurvenscheibe V3.0 (S7-1500T)
V1.6, V1.7, V1.8	V2.0	Drehzahlachse V2.0 Positionierachse V2.0 Externer Geber V2.0 Gleichlaufachse V2.0
V1.0, V1.1, V1.5[1]	V1.0	Drehzahlachse V1.0 Positionierachse V1.0 Externer Geber V1.0

1) Der Kartentausch von einer CPU < V1.6 zu einer CPU ≥ V1.6 wird unterstützt.

Parameter "Mode" der Motion Control-Anweisung "MC_Home"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode".

Folgende Tabelle zeigt die Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologie V1.0 und ≥ V2.0:

MC_Home.HomingMode V1.0	Parameter-wert	MC_Home.Mode ≥ V2.0
Direktes Referenzieren (Absolut)	0	Direktes Referenzieren (Absolut)
Direktes Referenzieren (Relativ)	1	Direktes Referenzieren (Relativ)
Passives Referenzieren	2	Passives Referenzieren (ohne Rücksetzen)

MC_Home.HomingMode V1.0	Parameter-wert	MC_Home.Mode ≥ V2.0
Passives Referenzieren (mit konfigurierter Referenzpunktposition)	3	Aktives Referenzieren
Aktives Referenzieren	4	Reserviert
Aktives Referenzieren (mit konfigurierter Referenzpunktposition)	5	Aktives Referenzieren (mit konfigurierter Referenzpunktposition)
Absolutwertgeberjustage (Relativ)	6	Absolutwertgeberjustage (Relativ)
Absolutwertgeberjustage (Absolut)	7	Absolutwertgeberjustage (Absolut)
Passives Referenzieren (ohne Rücksetzen)	8	Passives Referenzieren
Abbruch Passives Referenzieren	9	Abbruch Passives Referenzieren
-	10	Passives Referenzieren (mit konfigurierter Referenzpunktposition)

Weitere Informationen zum Parameter "MC_Home.Mode" finden Sie in der Beschreibung der Motion Control-Anweisung "MC_Home".

Variablen des Technologieobjekts

Ab der Technologieversion V3.0 werden alle Ein- und Ausgangsadressen über den Datentyp "VREF" vorgegeben. Dadurch ergeben sich folgende Änderungen bei den Variablen des Technologieobjekts:

Variable des Technologieobjekts	Änderungen ab V3.0
<TO>.Actor.Interface.AddressIn	Datentyp: VREF
<TO>.Actor.Interface.AddressOut	Datentyp: VREF
<TO>.Sensor[n].Interface.AddressIn	Datentyp: VREF
<TO>.Sensor[n].Interface.AddressOut	Datentyp: VREF
<TO>.Actor.Interface.EnableDriveOutputAddress	Datentyp: VREF
<TO>.Actor.Interface.EnableDriveOutputBitNumber	Variable entfällt
<TO>.Actor.Interface.DriveReadyInputAddress	Datentyp: VREF
<TO>.Actor.Interface.DriveReadyInputBitNumber	Variable entfällt
<TO>.Sensor[n].ActiveHoming.DigitalInputAddress	Datentyp: VREF
<TO>.Sensor[n].ActiveHoming.DigitalInputBitNumber	Variable entfällt
<TO>.Sensor[n].PassiveHoming.DigitalInputAddress	Datentyp: VREF
<TO>.Sensor[n].PassiveHoming.DigitalInputBitNumber	Variable entfällt
<TO>.PositionLimits_HW.MinInputAddress	Datentyp: VREF
<TO>.PositionLimits_HW.MinInputBitNumber	Variable entfällt
<TO>.PositionLimits_HW.MaxInputAddress	Datentyp: VREF
<TO>.PositionLimits_HW.MaxInputBitNumber	Variable entfällt

Version V4.0

Neuerungen

Die Technologieversion V4.0 enthält folgende Neuerungen:

- Austausch von Momentendaten mit dem Antrieb in den technologischen Einheiten des Technologieobjekts
  - Additives Sollmoment
  - Aktuelles Istmoment
  - Zulässiger Momentenbereich
- Erweiterung der Datenstruktur von Positionierachse und Gleichlaufachse zur Verwendung des Technologieobjekts Kinematik
- Verwendung optimierter Datenbausteine (Antriebs-/Geberanbindung)

Zusätzliche Neuerungen S7-1500T

Die Technologieversion V4.0 enthält folgende zusätzliche Neuerungen:

- Technologieobjekt Kinematik (S7-1500T)
- Bewegungsvorgabe über "MotionIn"-Anweisungen (S7-1500T)
- Direkt synchron setzen mit "MC_CamIn" V4.0 (S7-1500T)

Motion Control-Anweisungen

Die Technologieversion V4.0 enthält folgende Motion Control-Anweisungen:

- MC_Power V4.0
- MC_Reset V4.0
- MC_Home V4.0
- MC_Halt V4.0
- MC_MoveAbsolute V4.0
- MC_MoveRelative V4.0
- MC_MoveVelocity V4.0
- MC_MoveJog V4.0
- MC_MoveSuperimposed V4.0
- MC_SetSensor V4.0 (S7-1500T)
- MC_TorqueLimit V4.0
- MC_MeasuringInput V4.0
- MC_MeasuringInputCyclic V4.0
- MC_AbortMeasuringInput V4.0
- MC_OutputCam V4.0

- MC_CamTrack V4.0
- MC_GearIn V4.0
- MC_GearInPos V4.0 (S7-1500T)
- MC_PhasingAbsolute V4.0 (S7-1500T)
- MC_PhasingRelative V4.0 (S7-1500T)
- MC_CamIn V4.0 (S7-1500T)
- MC_SynchronizedMotionSimulation V4.0 (S7-1500T)
- MC_InterpolateCam V4.0 (S7-1500T)
- MC_GetCamFollowingValue V4.0 (S7-1500T)
- MC_GetCamLeadingValue V4.0 (S7-1500T)
- MC_MotionInVelocity V4.0 (S7-1500T)
- MC_MotionInPosition V4.0 (S7-1500T)
- MC_TorqueAdditive V4.0
- MC_TorqueRange V4.0
- MC_GroupInterrupt V4.0 (S7-1500T)
- MC_GroupContinue V4.0 (S7-1500T)
- MC_GroupStop V4.0 (S7-1500T)
- MC_MoveLinearAbsolute V4.0 (S7-1500T)
- MC_MoveLinearRelative V4.0 (S7-1500T)
- MC_MoveCircularAbsolute V4.0 (S7-1500T)
- MC_MoveCircularRelative V4.0 (S7-1500T)
- MC_DefineWorkspaceZone V4.0 (S7-1500T)
- MC_DefineKinematicsZone V4.0 (S7-1500T)
- MC_SetWorkspaceZoneActive V4.0 (S7-1500T)
- MC_SetWorkspaceZoneInactive V4.0 (S7-1500T)
- MC_SetKinematicsZoneActive V4.0 (S7-1500T)
- MC_SetKinematicsZoneInactive V4.0 (S7-1500T)
- MC_DefineTool V4.0 (S7-1500T)
- MC_SetTool V4.0 (S7-1500T)
- MC_SetOcsFrame V4.0 (S7-1500T)

Versionsabhängige UDT-Namen

Die folgende Tabelle zeigt die versionsabhängigen UDT-Namen für die Steuerworte und Zustandsworte der SIEMENS-Telegramme 10x:

UDT-Name <V4.0	UDT-Name ≥V4.0	Datentyp word
PD_STW1	PD_STW1_611Umode	Control word 1 (Steuerwort 1 (STW1))
PD_STW2	PD_STW2_611Umode	Control word 2 (Steuerwort 2 (STW2))
PD_ZSW1	PD_ZSW1_611Umode	Status word 1 (Zustandswort 1 (ZSW1))
PD_ZSW2	PD_ZSW2_611Umode	Status word 2 (Zustandswort 2 (ZSW2))

Wenn Sie von einer Technologieversion <V4.0 auf ≥V4.0 oder umgekehrt wechseln, tritt ein Fehler bei der Übersetzung auf. Sie müssen die UDT-Namen händisch anpassen.

Version V3.0

Neuerungen

Die Technologieversion V3.0 enthält folgende Neuerungen:

- Technologieobjekt Messtaster
- Technologieobjekt Nocken
- Technologieobjekt Nockenspur
- Kraft-/Momentenbegrenzung
- Festanschlagserkennung
- Achstyp virtuelle Achse
- MC-PreServo [OB67] und MC-PostServo [OB95]
- Technologie-CPU S7-1500T

Zusätzliche Neuerungen S7-1500T

Die Technologieversion V3.0 enthält folgende zusätzliche Neuerungen:

- Technologieobjekt Kurvenscheibe (S7-1500T)
- Getriebegleichlauf mit "MC_GearInPos" (S7-1500T)
- Kurvenscheibengleichlauf mit "MC_CamIn" (S7-1500T)
- Mehrere Geber einsetzen (S7-1500T)

Motion Control-Anweisungen

Die Technologieversion V3.0 enthält folgende Motion Control-Anweisungen:

- MC_Power V3.0
- MC_Reset V3.0
- MC_Home V3.0

- MC_Halt V3.0
- MC_MoveAbsolute V3.0
- MC_MoveRelative V3.0
- MC_MoveVelocity V3.0
- MC_MoveJog V3.0
- MC_MoveSuperimposed V3.0
- MC_SetSensor V3.0 (S7-1500T)
- MC_TorqueLimit V3.0
- MC_MeasuringInput V3.0
- MC_MeasuringInputCyclic V3.0
- MC_AbortMeasuringInput V3.0
- MC_OutputCam V3.0
- MC_CamTrack V3.0
- MC_GearIn V3.0
- MC_GearInPos V3.0 (S7-1500T)
- MC_PhasingAbsolute V3.0 (S7-1500T)
- MC_PhasingRelative V3.0 (S7-1500T)
- MC_CamIn V3.0 (S7-1500T)
- MC_SynchronizedMotionSimulation V3.0 (S7-1500T)
- MC_InterpolateCam V3.0 (S7-1500T)
- MC_GetCamFollowingValue V3.0 (S7-1500T)
- MC_GetCamLeadingValue V3.0 (S7-1500T)

Version V2.0

Neuerungen

Die Technologieversion V2.0 enthält folgende Neuerungen:

- Technologieobjekt Gleichlaufachse
- Getriebegleichlauf mit "MC_GearIn"
- Überlagerndes Positionieren mit "MC_MoveSuperimposed"
- Vereinheitlichung des Parameters "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control
- Simulationsbetrieb
- Unterstützung von Safety-Funktionen des Antriebs

Motion Control-Anweisungen

Die Technologieversion V2.0 enthält folgende Motion Control-Anweisungen:

- MC_Power V2.0
- MC_Reset V2.0
- MC_Home V2.0
- MC_Halt V2.0
- MC_MoveAbsolute V2.0
- MC_MoveRelative V2.0
- MC_MoveVelocity V2.0
- MC_MoveJog V2.0
- MC_MoveSuperimposed V2.0
- MC_GearIn V2.0

Version V1.0

Motion Control-Anweisungen

Die Technologieversion V1.0 enthält folgende Motion Control-Anweisungen:

- MC_Power V1.0
- MC_Reset V1.0
- MC_Home V1.0
- MC_Halt V1.0
- MC_MoveAbsolute V1.0
- MC_MoveRelative V1.0
- MC_MoveVelocity V1.0
- MC_MoveJog V1.0

Technologieversion ändern

Um die Vorteile einer neuen Technologieversion nutzen zu können, müssen Sie bei bestehenden Projekten die Technologieversion ändern.

Technologieversion ändern

Um die Technologieversion zu ändern, gehen Sie folgendermaßen vor:

1. Tauschen Sie die CPU im Projekt gegen eine CPU mit entsprechend höherer Version aus.
2. Öffnen Sie den Programmiereditor (z. B. durch Öffnen des OB1).
   Die Technologieobjekte und Motion Control-Anweisungen werden nach dem Tausch der CPU rot als ungültig markiert.

3. Wählen Sie in der Task Card "Anweisungen" im Ordner "Technologie > Motion Control" die entsprechend höhere Technologieversion.
4. Speichern und übersetzen Sie das Projekt.
   Beim Übersetzen des Projekts wird die Version der Technologieobjekte und Motion Control-Anweisungen auf die entsprechend höhere Technologieversion geändert.
   Beachten Sie evtl. Fehleranzeigen beim Übersetzen. Beseitigen Sie die Ursachen der angezeigten Fehler.
5. Überprüfen Sie die Konfiguration der Technologieobjekte.

Parameter "Mode" der Motion Control-Anweisung "MC_Home" neu einstellen

Beim Wechsel der Technologieversion von V1.0 auf ≥ V2.0 wird der Parameter "MC_Home.HomingMode" (V1.0) umbenannt in "MC_Home.Mode" (≥ V2.0). Die Belegung der Parameterwerte wird ebenfalls geändert.

Um den Parameter "MC_Home.Mode" (≥ V2.0) neu einzustellen, gehen Sie folgendermaßen vor:

1. Um die Version der Technologie zu ändern, folgen Sie der obigen Handlungsanweisung. Beim Übersetzen des Projekts wird der Parameter "MC_Home.HomingMode" (V1.0) umbenannt in "MC_Home.Mode" (≥ V2.0):
   - Die Belegung der Parameterwerte wird geändert. Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und ≥ V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".
     Weitere Informationen zum Parameter "MC_Home.Mode" finden Sie in der Beschreibung der Motion Control-Anweisung "MC_Home".
   - Der konfigurierte Wert am Parameter "MC_Home.HomingMode" (V1.0) geht verloren. Als Hinweis zur Umbenennung wird am Parameter "MC_Home.Mode" (≥ V2.0) folgender Text als Parameterwert eingefügt:
     "Die Schnittstelle hat sich geändert. Weitere Informationen finden Sie in der Beschreibung der Motion Control-Anweisung MC_Home."
   - In der Registerkarte "Info > Übersetzen" im Inspektorfenster wird gemeldet, dass der Operand den falschen Datentyp hat.
2. Ändern Sie entsprechend Ihres Anwenderprogramms den Wert des Parameters "MC_Home.Mode" (≥ V2.0) gemäß der neuen Belegung.
3. Speichern und übersetzen Sie das Projekt.

Gerät tauschen

Sie können eine S7-1500 gegen eine S7-1500T tauschen und umgekehrt. Je nach Art des Tauschs ist das Verhalten hinsichtlich Funktionen und bereits vorhandener Konfiguration unterschiedlich.

- S7-1500 ⇒ S7-1500T
  Die Funktionen der S7-1500 werden um zusätzliche Parameter für die erweiterten Funktionen der S7-1500T erweitert. Die zusätzlichen Parameter sind mit Standardwerten vorbelegt und müssen entsprechend konfiguriert werden.
- S7-1500T ⇒ S7-1500
  - Erweiterte Funktionen werden nur durch eine S7-1500T unterstützt und sind nach einem Tausch an der S7-1500 nicht mehr verfügbar.
  - Die nicht unterstützten Funktionsbausteine werden gekennzeichnet.
  - Die nicht unterstützen Technologieobjekte werden nach dem Übersetzen in einer Fehlermeldung angezeigt und müssen gelöscht werden.

6.1.2.4 Konfigurieren

Antriebe in der Gerätekonfiguration hinzufügen und konfigurieren

Im folgenden Kapitel wird das Hinzufügen und Konfigurieren anhand eines SINAMICS S120 Antriebs beschrieben. Wenn Sie einen SINAMICS V90 PN oder einen SINAMICS-Antrieb mit Startdrive verwenden, finden Sie weitere Informationen in den folgenden Dokumentationen.

Verwendung von SINAMICS V90 PN

Für das Hinzufügen und Konfigurieren eines SINAMICS V90 PN Antriebs im TIA Portal benötigen Sie das Hardware Support Package (HSP) V90. Informationen zur Projektierung eines SINAMICS V90 PN Antriebs mit SIMATIC S7-1500 im TIA Portal finden Sie im Getting Started "SIMATIC/SINAMICS Erste Schritte SINAMICS V90 PN an S7-1500 Motion Control":

https://support.industry.siemens.com/cs/ww/de/view/109739497 (https://support.industry.siemens.com/cs/ww/de/view/109739497)

Verwendung von Startdrive

Wenn Sie einen SINAMICS-Antrieb mit Startdrive verwenden, finden Sie die Antriebe im Hardware-Katalog im Ordner "Antriebe & Starter". Weitere Informationen zur Anbindung über Startdrive finden Sie unter:

- "Getting Started SINAMICS S120 im Startdrive V14 SP1"
  https://support.industry.siemens.com/cs/ww/de/view/109747452 (https://support.industry.siemens.com/cs/ww/de/view/109747452)
- Anwendungsbeispiel "Positionieren eines S120 (Startdrive) mit SIMATIC S7-1500 (TO)"
  https://support.industry.siemens.com/cs/document/109743270 (https://support.industry.siemens.com/cs/ww/de/view/109743270)

Kompatibilitätsliste der Antriebe

Eine Übersicht, welche Antriebe Sie mit einer S7-1500 CPU verschalten können, finden Sie im Anhang (Seite 2222).

PROFINET IO-Antriebe hinzufügen und konfigurieren

Das Hinzufügen und Konfigurieren eines PROFINET IO-Antriebs wird im Folgenden anhand eines SINAMICS S120 Antriebs beschrieben. Das Hinzufügen und Konfigurieren anderer PROFINET IO-Antriebe kann in einzelnen Punkten von der Beschreibung abweichen.

Bei Verwendung einer CPU S7-1500C können Sie die Ein-/Ausgänge der CPU als Schnittstelle zum Antrieb verwenden.

Voraussetzung

- Das Gerät SIMATIC S7-1500 ist im Projekt angelegt.
- Der gewünschte Antrieb steht im Hardware-Katalog zur Auswahl.

Wenn der Antrieb im Hardware-Katalog nicht zur Verfügung steht, müssen Sie diesen im Menü "Extras" als Gerätebeschreibungsdatei (GSD) installieren.

Antrieb und Telegramm in der Gerätekonfiguration hinzufügen

1. Öffnen Sie die Gerätekonfiguration und wechseln Sie in die Netzsicht.
2. Öffnen Sie im Hardware-Katalog den Ordner "Weitere Feldgeräte > PROFINET IO > Drives > Siemens AG > SINAMICS".
3. Wählen Sie den gewünschten Antrieb in der gewünschten Version und ziehen Sie diesen per Drag&Drop in die Netzsicht.
4. Ordnen Sie den Antrieb der PROFINET-Schnittstelle der CPU zu.
5. Öffnen Sie den Antrieb in der Gerätesicht.
6. Ziehen Sie per Drag&Drop aus dem Hardware-Katalog ein Drive Object (DO) und ein Telegramm auf einen Steckplatz der Geräteübersicht des Antriebs.
7. Stellen Sie sicher, dass die Telegrammreihenfolge in der Gerätekonfiguration und der Antriebsparametrierung die gleiche ist.

Wählen Sie je nach Version des SINAMICS S120 Antriebs zur Telegrammauswahl "DO mit Telegramm X" oder "DO Servo" und ein "Telegramm X".

Informationen zu geeigneten Telegrammen finden Sie im Kapitel "PROFIdrive-Telegramme (Seite 1693)".

Wiederholen Sie Schritt 6, wenn Sie einen weiteren Antrieb und ein weiteres Standardtelegramm hinzufügen möchten.

Taktsynchronität des Antriebs in der Gerätekonfiguration aktivieren

PROFINET-Antriebe können grundsätzlich taktsynchron oder nicht taktsynchron betrieben werden. Die Taktsynchronität erhöht die Güte der Lageregelung des Antriebs und wird daher bei Antrieben, wie dem SINAMICS S120 empfohlen.

Um den Antrieb taktsynchron anzusteuern, gehen Sie folgendermaßen vor:

1. Wählen Sie die Gerätesicht des Antriebs.
2. Wählen Sie im Eigenschaftsfenster das Register "PROFINET-Schnittstelle [X1] > Erweiterte Optionen > Taktsynchronisation".
3. Aktivieren Sie in diesem Register das Kontrollkästchen "Taktsynchroner Betrieb".

In der Detailansicht muss der Eintrag zum Telegramm ebenfalls für die Taktsynchronisation markiert sein.

Port der CPU mit dem Port des Antriebs verschalten

1. Wählen Sie die Topologiesicht der Gerätekonfiguration
2. Verschalten Sie den Port des Antriebs wie im realen Aufbau mit dem Port der CPU. Beachten Sie dabei die Regeln zum Topologieaufbau.

CPU als Sync-Master konfigurieren und isochronen Takt einstellen

1. Wählen Sie die Gerätesicht der CPU.
2. Wählen Sie im Eigenschaftsfenster das Register "PROFINET-Schnittstelle [X1] > Erweiterte Optionen > Echtzeit-Einstellungen > Synchronisation".
3. Wählen Sie in der Klappliste "Synchronisationsrolle" den Eintrag "Sync-Master".
4. Betätigen Sie die Schaltfläche "Domain-Einstellungen".
5. Wählen Sie das Register "Domain-Management > Sync-Domains" und stellen Sie den gewünschten "Sendetakt" (isochronen Takt) ein.

Antrieb in der Konfiguration des Technologieobjekts auswählen

1. Fügen Sie ein neues Technologieobjekt Achse hinzu, bzw. öffnen Sie die Konfiguration einer existierenden Achse.
2. Öffnen Sie die Konfiguration "Hardware-Schnittstelle > Antrieb".
3. Wählen Sie in der Klappliste "Antriebstyp" den Eintrag "PROFIdrive".
4. Wählen Sie in der Liste "Antrieb" das Drive Object des PROFINET-Antriebs aus.

Wie Sie ein Technologieobjekt hinzufügen können, finden Sie im Kapitel "Technologieobjekt hinzufügen (Seite 1843)".

Ergebnis

Das Technologieobjekt ist mit dem Antrieb verbunden und der Organisationsbaustein "MC-Servo" kann überprüft/konfiguriert werden.

Das Telegramm des konfigurierten Antriebs wird dem Prozessabbild "TPA OB Servo" zugeordnet.

Eigenschaften des MC-Servo überprüfen/konfigurieren

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Markieren Sie den Organisationsbaustein "MC-Servo".
3. Wählen Sie den Kontextmenübefehl "Eigenschaften".
4. Wählen Sie in der Bereichsnavigation den Eintrag "Zykluszeit".
5. Im Dialogfenster muss die Option "Synchron zum Bus" ausgewählt sein.
6. In der Klappliste "Dezentrale Peripherie" muss ein "PROFINET IO_System" ausgewählt sein.
7. Der Applikationszyklus des "MC-Servo" muss dem Sendetakt des Busses entsprechen bzw. ganzzahlig zum Sendetakt des Busses untersetzt sein.

Ergebnis

Der PROFINET IO-Antrieb ist so konfiguriert, dass er im PROFINET IO-Netz taktsynchron angesteuert werden kann.

Die Eigenschaften des SINAMICS-Antriebs müssen entsprechend der Konfiguration der Achse mit der Software STARTER oder SINAMICS Startdrive konfiguriert werden.

Überprüfen der Taktsynchronisation am Antrieb

Wenn bei der Konfiguration der Achse die obige Reihenfolge nicht eingehalten wurde und beim Übersetzen des Projekts treten antriebsspezifische Fehler auf, kann die Einstellung zur Taktsynchronisation am Antrieb überprüft werden.

1. Wählen Sie die Gerätesicht des Antriebs.
2. Wählen Sie in der Geräteübersicht den Eintrag des Standardtelegramms.
3. Wählen Sie den Eigenschaftsdialog "Allgemein > E/A-Adressen".
4. Für die Eingangs- und Ausgangsadressen müssen folgende Einstellungen gelten:
   - "Taktsynchroner Betrieb" ist aktiviert.
   - Für "Organisationsbaustein" muss "MC-Servo" ausgewählt sein.
   - Für "Prozessabbild" muss "TPA OB Servo" ausgewählt sein.

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Technologieobjekt hinzufügen (Seite 1843)

PROFIBUS DP-Antriebe hinzufügen und konfigurieren

Das Hinzufügen und Konfigurieren eines PROFIBUS-Antriebs wird im Folgenden anhand eines SINAMICS S120 Antriebs beschrieben. Das Hinzufügen und Konfigurieren anderer PROFIBUS-Antriebe kann in einzelnen Punkten der Beschreibung abweichen.

Bei Verwendung einer CPU S7-1500C können Sie die Ein-/Ausgänge der CPU als Schnittstelle zum Antrieb verwenden.

Voraussetzung

- Das Gerät SIMATIC S7-1500 ist im Projekt angelegt.
- Der gewünschte Antrieb steht im Hardware-Katalog zur Auswahl.

Wenn der Antrieb im Hardware-Katalog nicht zur Verfügung steht, müssen Sie den Antrieb im Menü "Extras" als Gerätebeschreibungsdatei (GSD) installieren.

Antrieb und Telegramm in der Gerätekonfiguration hinzufügen

1. Öffnen Sie die Gerätekonfiguration und wechseln Sie in die Netzsicht.
2. Öffnen Sie im Hardware-Katalog den Ordner "Weitere Feldgeräte > PROFIBUS DP > Antriebe > Siemens AG > SINAMICS".
3. Wählen Sie den Ordner des gewünschten Antriebs in der gewünschten Version und ziehen Sie das Objekt des Antriebs per Drag&Drop in die Netzsicht.
4. Ordnen Sie den Antrieb der PROFIBUS-Schnittstelle der CPU zu.
5. Öffnen Sie den Antrieb in der Gerätesicht.
6. Ziehen Sie per Drag&Drop aus dem Hardware-Katalog ein Telegramm auf einen Steckplatz der Geräteübersicht des Antriebs.

Informationen zu geeigneten Telegrammen finden Sie im Kapitel "PROFIdrive-Telegramme (Seite 1693)".

Wenn Sie einen weiteren Antrieb und ein weiteres Telegramm in die Geräteübersicht hinzufügen möchten, verwenden Sie den "Achstrenner" im Hardware-Katalog.

Taktsynchronität des Antriebs in der Gerätekonfiguration aktivieren

PROFIBUS-Antriebe können zyklisch oder taktsynchron betrieben werden. Die Taktsynchronität erhöht jedoch die Güte der Lageregelung des Antriebs.

Wenn Sie den Antrieb taktsynchron ansteuern möchten, gehen Sie folgendermaßen vor:

1. Wählen Sie die Gerätesicht des Antriebs.
2. Wählen Sie im Eigenschaftsdialog das Register "Allgemein > Taktsynchronisation".
3. Aktivieren Sie das Kontrollkästchen "DP-Slave auf äquidistanten DP-Zyklus synchronisieren".

Isochronen Takt einstellen

1. Wählen Sie die Netzsicht.
2. Markieren Sie das DP-Mastersystem.
3. Wählen Sie im Eigenschaftsdialog das Register "Allgemein > Äquidistanz".
4. Wählen Sie den gewünschten "Äquidistanten DP-Zyklus".

Antrieb in der Konfiguration des Technologieobjekts auswählen

1. Fügen Sie ein neues Technologieobjekt Achse hinzu, bzw. öffnen Sie die Konfiguration einer existierenden Achse.
2. Öffnen Sie die Konfiguration "Hardware-Schnittstelle > Antrieb".
3. Wählen Sie in der Klappliste "Antriebstyp" den Eintrag "PROFIdrive".
4. Wählen Sie in der Liste "Antrieb" das Telegramm des PROFIBUS-Antriebs aus.

Wie Sie ein Technologieobjekt hinzufügen können, finden Sie im Kapitel "Technologieobjekt hinzufügen (Seite 1843)".

Ergebnis

Das Technologieobjekt ist mit dem Antrieb verbunden und der Organisationsbaustein "MC-Servo" kann überprüft/konfiguriert werden.

Das Telegramm des konfigurierten Antriebs wird dem Prozessabbild "TPA OB Servo" zugeordnet.

Eigenschaften des MC-Servo überprüfen/konfigurieren

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Markieren Sie den Organisationsbaustein "MC-Servo".
3. Wählen Sie den Kontextmenübefehl "Eigenschaften" aus.
   Der Dialog "MC-Servo" wird geöffnet.
4. Wählen Sie unter "Allgemein > Zykluszeit" die Option "Synchron zum Bus" aus.
5. Wählen Sie in der Klappliste "Dezentrale Peripherie" ein "PROFIBUS DP-System" aus.
   Der Applikationszyklus des "MC-Servo" muss dem Sendetakt des Busses entsprechen bzw. ganzzahlig zum Sendetakt des Busses untersetzt sein.

Einen Antrieb, der über einen Kommunikationsprozessor/Kommunikationsmodul (CP/CM) an die CPU angebunden ist, können Sie in der Konfiguration des Technologieobjekts auswählen. Das DP-Mastersystem des CP/CM können Sie nicht als Taktquelle für den MC-Servo [OB91] auswählen.

Ergebnis

Der PROFIBUS DP-Antrieb ist so konfiguriert, dass er im PROFIBUS-Netz taktsynchron angesteuert werden kann.

Die Eigenschaften des SINAMICS-Antriebs müssen entsprechend der Konfiguration der Achse mit der Sofarwae STARTER oder SINAMICS Startdrive konfiguriert werden.

Überprüfen der Taktsynchronisation am Antrieb

Wurde bei der Konfiguration der Achse die obige Reihenfolge nicht eingehalten und beim Übersetzen des Projekts treten antriebsspezifische Fehler auf, so kann die Taktsynchronisation am Antrieb überprüft werden.

1. Wählen Sie die Gerätesicht des Antriebs.
2. Wählen Sie in der Geräteübersicht den Eintrag des Telegramms.
3. Wählen Sie den Eigenschaftsdialog "Allgemein > E/A-Adressen".
4. Für die Eingangs- und Ausgangsadressen müssen folgende Einstellungen gelten:
   – Für "Organisationsbaustein" muss "MC-Servo" gewählt sein.
   – Für "Prozessabbild" muss "TPA OB Servo" gewählt sein.

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Technologieobjekt hinzufügen (Seite 1843)

Antriebe mit analoger Antriebsanbindung hinzufügen und konfigurieren

Im Folgenden wird das Hinzufügen und Konfigurieren eines Antriebs mit analoger Antriebsanbindung und eines Gebers beschrieben. Die Anbindung erfolgt beispielhaft als Positionierachse mit einem inkrementellen Geber und über ein Technologiemodul im Baugruppenträger der CPU.

Bei Verwendung einer CPU S7-1500C können Sie die Ein-/Ausgänge der CPU als Schnittstelle zum Antrieb verwenden.

Voraussetzung

Das Gerät SIMATIC S7-1500 ist im Projekt angelegt.

Analogausgabemodul in der Gerätekonfiguration hinzufügen und konfigurieren

1. Öffnen Sie die Gerätekonfiguration der CPU.
2. Wählen Sie aus dem Hardware-Katalog ein Analogausgabemodul und ziehen Sie dieses per Drag&Drop in den Baugruppenträger der CPU.
3. Markieren Sie das Analogausgabemodul in der Geräteübersicht.
4. Wählen Sie im Eigenschaftsdialog das Register "Allgemein" und dort "Name des Analogausgabemoduls > E/A-Adressen".
5. Tragen Sie die gewünschte Anfangsadresse ein.
6. Wählen Sie im Eigenschaftsdialog das Register "IO-Variablen".
7. Geben Sie dem gewünschten Analogausgang einen Variablennamen.

Technologiemodul hinzufügen und konfigurieren

1. Wechseln Sie in die Gerätesicht der CPU.
2. Öffnen Sie im Hardware-Katalog den Ordner "TM > Zählen > TM Count 2X24V".
3. Ziehen Sie das Zählermodul per Drag&Drop auf einen freien Steckplatz des Baugruppenträgers. Beim Einsatz im Baugruppenträger der CPU kann das Technologiemodul nicht taktsynchron betrieben werden.
4. Markieren Sie das Technologiemodul in der Gerätesicht.
5. Öffnen Sie im Eigenschaftsdialog das Register "Allgemein" und dort "Count 2x24V > Grundparameter > Kanal X > Betriebsart" des zu verwendenden Kanals.
6. Wählen Sie für "Auswahl des Betriebsmodus für den Kanal" die Option "Positionserfassung für Motion Control".
7. Passen Sie unter "Modulparameter" die Parameter des inkrementellen Gebers an (Schritte pro Umdrehung = Inkremente pro Umdrehung).
8. Wählen Sie unter "Verhalten bei CPU-Stop" den Eintrag "Weiterarbeiten".

Antrieb und Geber in der Konfiguration des Technologieobjekts auswählen

1. Fügen Sie ein neues Technologieobjekt Positionierachse/Gleichlaufachse hinzu, bzw. öffnen Sie die Konfiguration einer existierenden Positionierachse/Gleichlaufachse.
2. Öffnen Sie die Konfiguration "Hardware-Schnittstelle > Antrieb".
3. Wählen Sie in der Klappliste "Antriebstyp" den Eintrag "Analoge Antriebsanbindung".
4. Wählen Sie in der Liste "Ausgang" den zuvor definierten Variablennamen des Analogausgangs aus.
5. Öffnen Sie die Konfiguration "Hardware-Schnittstelle > Geber".
6. Wählen Sie unter "Geberkopplung auswählen" die Option "Geber am Technologiemodul (TM)".
7. Wählen Sie in der Liste "Technologiemodul" unter "local modules" den Kanal des inkrementellen Gebers.
8. Öffnen Sie die Konfiguration "Hardware-Schnittstelle > Datenaustausch Geber > Feinauflösung" und geben Sie für "Bits im inkr. Istwert" den Wert "0" ein.

Wie Sie ein Technologieobjekt hinzufügen können, finden Sie im Kapitel "Technologieobjekt hinzufügen (Seite 1843)".

Ergebnis

Die analoge Antriebsanbindung und die Geberanbindung sind konfiguriert.

Die Analogadressen und die Adressen des TM-Moduls werden dem Prozessabbild "TPA OB Servo" zugeordnet.

Überprüfung der Geberanbindung/Antriebsanbindung

Die Geberdaten werden im Lagereglertakt übernommen. Überprüfen Sie im Zweifelsfall folgende Einstellungen:

1. Wechseln Sie in die Gerätesicht der CPU.
2. Markieren Sie das Technologiemodul.
3. Öffnen Sie den Eigenschaftsdialog "Grundparameter > E/A-Adressen".
4. Für die Eingangs- und Ausgangsadressen müssen folgende Einstellungen gelten:
   – Für "Organisationsbaustein" muss "MC-Servo" gewählt sein.
   – Für "Prozessabbild" muss "TPA OB Servo" gewählt sein.
5. Markieren Sie das Analogmodul.
6. Öffnen Sie den Eigenschaftsdialog "Name des Analogmoduls > E/A-Adressen".
7. Für die Eingangs- und Ausgangsadressen müssen die Einstellungen aus Schritt 4 gelten

Siehe auch

Technologieobjekt hinzufügen (Seite 1843)

Grundlagen - Konfiguration

Technologieobjekt hinzufügen

Im Folgenden ist beschrieben, wie Sie ein Technologieobjekt in der Projektnavigation hinzufügen.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500 ist angelegt.
- Für die Technologieobjekte Nocken, Nockenspur, Messtaster:
  Ein Technologieobjekt Drehzahlachse, Positionierachse, Gleichlaufachse oder Externer Geber ist angelegt.

Vorgehen

Um ein Technologieobjekt hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie das gewünschte Technologieobjekt aus. Der eingeblendeten Beschreibung können Sie die Funktion des Technologieobjekts entnehmen.

5. Wenn Sie ein Technologieobjekte Nocken, Nockenspur, Messtaster hinzufügen, wählen Sie in der Klappliste "Zuzuordnende Achse oder Externer Geber" das übergeordnete Technologieobjekt aus.
6. Passen Sie im Eingabefeld "Name" den Namen Ihren Bedürfnissen an.
7. Um die vorgeschlagene Datenbausteinnummer zu ändern, wählen Sie die Option "Manuell" aus.
8. Um eigene Informationen zum Technologieobjekt zu ergänzen, klicken Sie auf "Weitere Informationen".
9. Um nach dem Hinzufügen des Technologieobjekts die Konfiguration zu öffnen, aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen".
10. Um das Technologieobjekt hinzuzufügen, klicken Sie auf die Schaltfläche "OK".

Ergebnis

Das neue Technologieobjekt wurde erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" angelegt.

Falls die Organisationsbausteine "MC-Servo" und "MC-Interpolator" noch nicht vorhanden waren, wurden diese hinzugefügt.

Technologieobjekt kopieren

Sie können ein Technologieobjekt auf folgende Weisen kopieren:

- Kopieren eines Technologieobjekts innerhalb einer CPU
- Kopieren eines Technologieobjekts von einer CPU S7-1500 auf eine CPU S7-1500T Zusätzliche Parameter für die erweiterten Funktionen werden mit Standardwerten vorbelegt. Diese müssen Sie entsprechend konfigurieren.
- Kopieren eines Technologieobjekts von einer CPU S7-1500T auf eine CPU S7-1500 Zusätzliche und durch die CPU S7-1500 nicht unterstützte Parameter werden auf die Standardwerte zurückgesetzt.

Wenn Sie ein Technologieobjekt kopieren, das unterlagerte Technologieobjekte, wie Nocken, Nockenspur oder Messtaster, besitzt, werden die unterlagerten Technologieobjekte ebenfalls kopiert.

Im Folgenden ist beschrieben, wie Sie ein Technologieobjekt innerhalb einer CPU kopieren. Das Vorgehen gilt entsprechend auch für die anderen genannten Kopierweisen.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500 ist angelegt.
- Im Projekt ist ein Technologieobjekt angelegt.

Vorgehen

Um ein Technologieobjekt zu kopieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Öffnen Sie gegebenenfalls das übergeordnete Technologieobjekt.
4. Markieren Sie das zu kopierende Technologieobjekt.
5. Wählen Sie im Kontextmenü den Befehl "Kopieren".
6. Markieren Sie den Ordner "Technologieobjekte" oder das überzuordnende Technologieobjekt.
7. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

Das ausgewählte Technologieobjekt inklusive der unterlagerten Technologieobjekte wurde kopiert und in der Projektnavigation im Ordner "Technologieobjekte" angelegt.

Technologieobjekt löschen

Technologieobjekte können Sie in der Projektnavigation löschen.

Wenn Sie ein Technologieobjekt löschen, das unterlagerte Technologieobjekte, wie Nocken, Nockenspur oder Messtaster, besitzt, werden die unterlagerten Technologieobjekte ebenfalls gelöscht.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500 ist angelegt.
- Im Projekt ist ein Technologieobjekt angelegt.

Vorgehen

Um ein Technologieobjekt zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Markieren Sie das zu löschende Technologieobjekt.
4. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Dialog "Löschen bestätigen" wird geöffnet.
5. Um das Technologieobjekt zu löschen, klicken Sie auf die Schaltfläche "Ja".

Ergebnis

Das ausgewählte Technologieobjekt wurde gelöscht.

Siehe auch

Werte vergleichen (Seite 1846)

Arbeiten mit dem Konfigurationseditor

Die Eigenschaften eines Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um in der Projektsicht das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Gruppe "Technologieobjekte" des Geräts.
2. Wählen Sie das Technologieobjekt und doppelklicken Sie auf "Konfiguration".

Die Konfiguration ist abhängig vom Objekttyp in Kategorien aufgeteilt, z. B. Grundparameter, Hardware-Schnittstelle, Erweiterte Parameter.

Symbole des Konfigurationseditors

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

Symbol	Beschreibung
	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig. Alle Eingabefelder der Konfiguration enthalten gültige Werte und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft. Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken des Feldes zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.

Werte vergleichen

Wenn eine Online-Verbindung zur CPU besteht, wird in der Konfiguration des Technologieobjekts die Funktion "Alle beobachten" angezeigt.

Die Funktion "Alle beobachten" bietet folgende Möglichkeiten:

- Vergleich der konfigurierten Startwerte des Projekts mit den Startwerten in der CPU und den Aktualwerten
- Direktes Bearbeiten der Aktualwerte und der Startwerte des Projekts
- Sofortiges Erkennen und Anzeigen von Eingabefehlern mit Korrekturvorschlägen
- Sicherung der Aktualwerte im Projekt
- Übertragen der Startwerte des Projekts in die CPU als Aktualwerte

Hinweis

Unterschiede von Online- und Ofline-Werten

Durch Hinzufügen oder Löschen von Technologieobjekten, die eine Verbindung zu anderen Technologieobjekten haben, wie Nocke, Nockenspur, Messtaster oder Synchronachse, können beim Vergleich der Online- und Offline-Werte Unterschiede auftreten. Durch eine erneute Übersetzung des Projekts und anschließendes Laden in die CPU können Sie die Unterschiede beheben.

Symbole und Bedienelemente

Wenn eine Online-Verbindung zur CPU besteht, werden an den Parametern die Aktualwerte angezeigt.

Neben den Aktualwerten der Parameter werden folgende Symbole angezeigt:

Symbol	Beschreibung
	Startwert in CPU ist gleich dem konfigurierten Startwert im Projekt
	Startwert in CPU ist ungleich dem konfigurierten Startwert im Projekt
	Software-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Software-Komponente sind Online- und Offline-Version verschieden.
	Der Vergleich Startwert in CPU und konfiguriertem Startwert im Projekt kann nicht durchgeführt werden.
	Ein Vergleich der Online- und Offline-Werte ist nicht sinnvoll.
	Mit dieser Schaltfläche zeigen Sie für den jeweiligen Parameter den Startwert der CPU und den Startwert des Projekts an.

Den Startwert der CPU können Sie direkt ändern und anschließend in die CPU laden. Bei direkt änderbaren Parametern kann auch der Aktualwert geändert werden und die Änderung wird direkt in die CPU übernommen.

Technologieobjekt Drehzahlachse konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen der Drehzahlachse. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen der Drehzahlachse können Sie im Anwenderprogramm unter diesem Namen verwenden.

Achstyp

Wenn Sie die Achse ausschließlich virtuell in der CPU verwenden möchten, z. B. als virtuelle Leitachse für den Gleichlauf, aktivieren Sie das Optionskästchen "Virtuelle Achse". Die Konfiguration einer Antriebs- und Geberanbindung ist nicht relevant.

Maßeinheiten

Wählen Sie in den Klapplisten die gewünschten Maßeinheiten für die Drehzahl und das Moment aus.

Simulation

Wenn Sie eine reale Achse im Simulationsbetrieb verfahren möchten, aktivieren Sie das Optionskästchen "Simulation aktivieren".

Im Simulationsbetrieb lassen sich Drehzahl-, Positionier- und Gleichlaufachsen ohne angebundene Antriebe und Geber in der CPU simulieren. Der Simulationsbetrieb ist ab Technologieversion V3.0 auch ohne konfigurierte Antriebs- und Geberanbindung möglich.

Für den Simulationsbetrieb ohne an der CPU angeschlossene Hardware können Sie die Anlaufzeit der CPU über den Parameter "Parametrierungszeit für zentrale und dezentrale Peripherie" beeinflussen. Den Parameter finden Sie in den Eigenschaften der CPU in der Bereichsnavigation "Anlauf".

Siehe auch

Maßeinheiten (Seite 1689)

Achse in Simulation (Seite 1706)

Hardware-Schnittstelle

Konfiguration - Antrieb

Konfigurieren Sie im Konfigurationsfenster "Antrieb", welchen Antriebstyp und welchen Antrieb Sie verwenden möchten.

Antriebstyp

Wählen Sie in der Klappliste, ob Sie einen PROFIdrive-Antrieb oder einen Antrieb mit analoger Antriebsanbindung einsetzen möchten.

PROFIdrive-Antriebe werden über ein digitales Kommunikationssystem (PROFINET oder PROFIBUS) mit der Steuerung verbunden. Die Kommunikation erfolgt über PROFIdrive-Telegramme.

Antriebe mit analoger Antriebsanbindung erhalten den Drehzahlsollwert über ein analoges Ausgangssignal (z. B. -10 V bis +10 V) der CPU.

Antriebstyp: PROFIdrive

Datenanbindung

Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Antriebsgerät erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.

Antrieb/Datenbaustein

Wählen Sie im Feld "Antrieb" einen bereits konfigurierten PROFIdrive-Antrieb/Slot aus. Wenn Sie einen PROFIdrive-Antrieb ausgewählt haben, können Sie den PROFIdrive-Antrieb über die Schaltfläche "Gerätekonfiguration" konfigurieren.

Wenn kein PROFIdrive-Antrieb zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie in der Netzsicht einen PROFIdrive-Antrieb hinzu.

Hinweis

Option "Zeige alle Module"

Wenn ein bereits konfigurierter PROFIdrive-Antrieb nicht zur Auswahl steht, zeigen Sie mit der Option "Zeige alle Module" alle erreichbaren Module an.

Wenn Sie die Option "Zeige alle Module" aktivieren, wird für alle angezeigten Module nur der Adressbereich der angezeigten Module überprüft. Wenn der Adressbereich eines Moduls groß genug für das gewählte PROFIdrive-Telegramm ist, können Sie das Modul auswählen. Stellen Sie daher sicher, dass Sie einen PROFIdrive-Antrieb auswählen.

Wenn Sie unter Datenanbindung "Datenbaustein" ausgewählt haben, wählen Sie hier einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Antriebstyp: Analoge Antriebsanbindung

Analogausgang

Wählen Sie im Feld "Analogausgang" die PLC-Variable des Analogausgangs, über welche der Antrieb angesteuert werden soll.

Um einen Ausgang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Analogausgangsmodul hinzugefügt haben und den PLC-Variablennamen für den Analogausgang definiert haben.

Freigabe-Ausgang aktivieren

Aktivieren Sie das Optionskästchen "Freigabe-Ausgang aktivieren", wenn der Antrieb eine Freigabe unterstützt.

Wählen Sie im entsprechenden Feld die PLC-Variable des Digitalausgangs zur Freigabe des Antriebs. Mit dem Freigabe-Ausgang wird der Drehzahlregler im Antrieb freigegeben, bzw. gesperrt.

Um einen Freigabe-Ausgang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Digitalausgangsmodul hinzugefügt haben und den PLC-Variablennamen für den Digitalausgang definiert haben.

Hinweis

Wenn Sie keinen Freigabe-Ausgang verwenden, kann der Antrieb infolge von Fehlerreaktionen oder Überwachungsfunktionen systemseitig nicht unmittelbar gesperrt werden. Ein kontrolliertes Stoppen des Antriebs ist nicht gewährleistet.

Bereit-Eingang aktivieren

Aktivieren Sie das Optionskästchen "Bereit-Eingang aktivieren", wenn der Antrieb seine Bereitschaft zurückmelden kann.

Wählen Sie im entsprechenden Feld die PLC-Variable des Digitaleingangs, über welchen der Antrieb seine Betriebsbereitschaft an das Technologieobjekt zurückmeldet. Das Leistungsteil ist eingeschaltet und der analoge Drehzahl-Sollwerteingang ist aktiv.

Um einen Bereit-Eingang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Digitaleingangsmodul hinzugefügt haben und den PLC-Variablennamen für den Digitaleingang definiert haben.

Hinweis

Der Freigabe-Ausgang und der Bereit-Eingang können voneinander unabhängig aktiviert werden.

Für den aktivierten Bereit-Eingang gelten folgende Randbedingungen:

- Die Achse wird erst freigegeben ("MC_Power Status" = TRUE), wenn am Bereit-Eingang ein Signal ansteht.
- Wenn bei einer freigegebenen Achse kein Signal am Bereit-Eingang anliegt, wird die Achse der Alarmreaktion "Freigabe wegnehmen" gesperrt.
- Wenn die Achse über die Anweisung "MC_Power" gesperrt wird ("Enable" = FALSE), wird die Achse auch mit anstehendem Signal am Bereit-Eingang gesperrt.

Siehe auch

Konfiguration - Datenaustausch Antrieb (Seite 1850)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Antriebs- und Geberanbindung (Seite 1692)

Konfiguration - Datenaustausch Antrieb

Konfigurieren Sie im Konfigurationsfenster "Datenaustausch Antrieb" den Datenaustausch mit dem Antrieb.

Die Konfiguration unterscheidet sich je nach gewähltem Antriebstyp:

Antriebstyp: PROFIdrive

Antriebstelegramm

Das in der Gerätekonfiguration eingestellte Telegramm zum Antrieb ist in der Klappliste vorausgewählt.

Automatischer Datenaustausch für Antriebswerte (offline)

Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Antriebs "Bezugsdrehzahl", "Maximale Drehzahl" und "Bezugsmoment" in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.

Automatischer Datenaustausch für Antriebswerte (online)

Aktivieren Sie das Optionskästchen, wenn Sie die online im Antrieb wirksamen Werte "Bezugsdrehzahl", "Maximale Drehzahl" und "Bezugsmoment" zur Laufzeit in die CPU übernehmen möchten. Die Antriebsparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Antriebs oder der CPU vom Bus übernommen.

Alternativ müssen Sie die folgenden Parameter händisch abgleichen:

- **Bezugsdrehzahl**
  Konfigurieren Sie in diesem Feld die Bezugsdrehzahl des Antriebs entsprechend den Angaben des Herstellers. Die Vorgabe der Antriebsdrehzahl erfolgt prozentual zur Bezugsdrehzahl im Bereich -200 % bis 200 %.
- **Maximale Drehzahl**
  Konfigurieren Sie in diesem Feld die maximale Drehzahl des Antriebs.
- **Bezugsmoment**
  Konfigurieren Sie in diesem Feld das Bezugsmoment des Antriebs entsprechend dessen Konfiguration.
  Das Bezugmoment ist zur Kraft-/Momentenreduzierung nötig, welches mit Telegramm 10x unterstützt wird.

Zusatzdaten

Aktivieren Sie das Optionskästchen "Momentendaten", wenn Sie die Datenanbindung der Momentendaten konfigurieren wollen. Wenn Sie einen Antrieb ausgewählt haben, bei dem das Zusatztelegramm 750 projektiert wurde, ist das Optionskästchen "Momentendaten" vorausgewählt.

Datenanbindung

Definieren Sie in der Klappliste, ob die Datenanbindung über Zusatztelegramm oder Datenbaustein erfolgen soll:

- Wenn Sie in der Klappliste "Datenanbindung" den Eintrag "Zusatztelegramm" auswählen, können Sie die Klappliste "Zusatztelegramm" bearbeiten.
- Wenn Sie in der Klappliste "Datenanbindung" den Eintrag "Datenbaustein" auswählen, können Sie einen zuvor erstellten Datenbaustein auswählen, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Zusatztelegrammnummer).

Datenbaustein/Zusatztelegramm

Wählen Sie im Feld "Zusatztelegramm" ein bereits konfiguriertes Zusatztelegramm aus.

Aktivieren Sie das Optionskästchen "Zeige alle Module", wenn Sie sich alle Submodule des angebundenen Antriebs anzeigen lassen wollen. Mit dieser Funktion finden Sie auch selbstdefinierte Zusatztelegramme.

Wählen Sie im Feld "Datenbaustein" den Datenbaustein aus, über den Sie die Momentendaten einbinden möchten.

Hinweis

Die automatische Übernahme der Antriebsparameter ist nur mit SINAMICS-Antrieben ab V4.x möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Antrieb" als Datenanbindung "Antrieb" gewählt sein.

Antriebstyp: Analoge Antriebsanbindung

Bezugsdrehzahl

Die Bezugsdrehzahl des Antriebs ist die Drehzahl, mit welcher der Antrieb bei der Ausgabe von 100 % am Analogausgang dreht. Die Bezugsdrehzahl muss am Antrieb konfiguriert werden und in der Konfiguration des Technologieobjekts übernommen werden.

Der bei 100 % ausgegebene Analogwert hängt vom Typ des Analogausgangs ab. Beispielsweise wird bei einem Analogausgang mit +/- 10 V bei 100 % der Wert 10 V ausgegeben.

Analogausgänge lassen sich um etwa 17 % übersteuern. Dies bedeutet, dass ein Analogausgang im Bereich -117 % bis 117 % betrieben werden kann, sofern dies der Antrieb zulässt.

Siehe auch

Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1698)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Antriebs- und Geberanbindung (Seite 1692)

Erweiterte Parameter

Konfiguration - Mechanik

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die Anbindung der Last an den Antrieb.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Siehe auch

Mechanik (Seite 1710)

Konfiguration - Dynamik-Voreinstellungen

Konfigurieren Sie im Konfigurationsfenster "Dynamik-Voreinstellung" die Voreinstellungswerte für Drehzahl, Beschleunigung, Verzögerung und Ruck der Achse.

Die Voreinstellungswerte wirken, wenn an den Motion Control-Anweisungen für die Parameter "Velocity", "Acceleration", "Deceleration" oder "Jerk" Werte < 0 angegeben werden. Die Voreinstellungswerte können einzeln für jeden genannten Parameter übernommen werden.

Drehzahl

Definieren Sie in diesem Feld den Voreinstellungswert für die Drehzahl der Achse.

Beschleunigung/Verzögerung - Hochlaufzeit/Rücklaufzeit

Stellen Sie den gewünschten Voreinstellungswert für Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$\textit{Hochlaufzeit} = \frac{\textit{Drehzahl}}{\textit{Beschleunigung}}$$

$$\textit{Rücklaufzeit} = \frac{\textit{Drehzahl}}{\textit{Verzögerung}}$$

Hinweis

Eine Änderung der Geschwindigkeit beeinflusst die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

Verrundungszeit/Ruck

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Ruck" ein. Der Wert 0 bedeutet, dass die Ruckbegrenzung deaktiviert wird.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

Hinweis

Der Ruck-Wert ist für Beschleunigungs- und Verzögerungsrampe identisch. Die bei der Verzögerungsrampe wirksame Verrundungszeit ergibt sich aus folgenden Beziehungen:

- **Beschleunigung > Verzögerung**
  Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- **Beschleunigung < Verzögerung**
  Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- **Beschleunigung = Verzögerung**
  Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.

Im Fehlerfall verzögert die Achse mit der konfigurierten Notstopp-Verzögerung. Eine konfigurierte Ruckbegrenzung wird hierbei nicht berücksichtigt.

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit dem gewählten Ruck ausgeführt.

Siehe auch

Geschwindigkeitsprofil (Seite 1717)

Konfiguration - Notstopp

Konfigurieren Sie im Konfigurationsfenster "Notstopp" die Notstopp-Verzögerung der Achse. Im Fehlerfall und beim Sperren der Achse mit der Motion Control-Anweisung "MC_Power" (Eingangsparameter "StopMode" = 0) wird die Achse mit dieser Verzögerung zum Stillstand gebracht.

Notstopp-Verzögerung/Notstopp-Rücklaufzeit

Stellen Sie den Verzögerungswert für Notstopp in den Feldern "Notstopp-Verzögerung" oder "Notstopp-Rücklaufzeit" ein.

Den Zusammenhang zwischen Notstopp-Rücklaufzeit und Notstopp-Verzögerung können Sie der folgenden Gleichung entnehmen:

$$\textit{Notstopp-Rücklaufzeit} = \frac{\textit{Maximale Drehzahl}}{\textit{Notstopp-Verzögerung}}$$

Die Konfiguration der Notstopp-Verzögerung bezieht sich auf die konfigurierte maximale Drehzahl der Achse. Wenn die maximale Drehzahl der Achse verändert wird, verändert sich auch der Wert der Notstopp-Verzögerung (die Notstopp-Rücklaufzeit bleibt unverändert).

Siehe auch

Notstopp-Verzögerung (Seite 1719)

Begrenzungen

Konfiguration - Dynamikgrenzen

Konfigurieren Sie im Konfigurationsfenster "Dynamikgrenzen" die Maximalwerte für Drehzahl, Beschleunigung, Verzögerung und Ruck der Achse.

Maximale Drehzahl

Definieren Sie in diesem Feld die maximal zugelassene Drehzahl der Achse.

Maximale Beschleunigung/Maximale Verzögerung - Hochlaufzeit/Rücklaufzeit

Stellen Sie die gewünschte Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$\textit{Hochlaufzeit} = \frac{\textit{Maximale Drehzahl}}{\textit{Beschleunigung}}$$

$$\textit{Rücklaufzeit} = \frac{\textit{Maximale Drehzahl}}{\textit{Verzögerung}}$$

Hinweis

Änderung der maximalen Drehzahl

Eine Änderung der maximalen Drehzahl beeinflusst die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

Verrundungszeit/Ruck

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Maximaler Ruck" ein. Der Wert 0 bedeutet, dass der Ruck nicht begrenzt wird.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

Hinweis

Unterschiedliche Werte von Beschleunigung und Verzögerung

Die eingestellte und in der Konfiguration angezeigte Verrundungszeit gilt nur für die Beschleunigungsrampe.

Im Falle, dass sich die Werte von Beschleunigung und Verzögerung unterscheiden, wird die Verrundungszeit der Verzögerungsrampe entsprechend dem Ruck der Beschleunigungsrampe berechnet und verwendet.

Die Verrundungszeit der Verzögerung wird wie folgt angepasst:

- **Beschleunigung > Verzögerung**
  Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- **Beschleunigung < Verzögerung**
  Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
- **Beschleunigung = Verzögerung**
  Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.

Im Fehlerfall verzögert die Achse mit der konfigurierten Notstopp-Verzögerung (Seite 1717) (Alarmreaktion "Stopp mit maximalen Dynamikwerten"). Eine konfigurierte Ruckbegrenzung wird hierbei nicht berücksichtigt.

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit dem gewählten Ruck ausgeführt.

Siehe auch

Notstopp-Verzögerung (Seite 1719)

Dynamikgrenzen im Gleichlauf (Seite 1720)

Konfiguration - Momentengrenzen

Konfigurieren Sie im Konfigurationsfenster "Momentengrenzen" die Momentenbegrenzung des Antriebs.

Die Konfiguration steht nur zur Verfügung, wenn ein Antrieb gewählt wurde, der die Kraft-/Momentenbegrenzung unterstützt und ein Telegramm 10x eingesetzt wird.

Wirksam

Wählen Sie in der Klappliste, ob der Begrenzungswert "An der Lastseite" oder "An der Motorseite" wirken soll.

Momentenbegrenzung

Geben Sie in diesem Feld einen Voreinstellungswert für die Momentenbegrenzung in der vorgegebenen Maßeinheit ein.

Der Voreinstellungswert wirkt, wenn die Momentenbegrenzung über die Motion Control-Anweisung "MC_TorqueLimiting" Eingangsparameter "Limit" < 0 vorgegeben wird.

Wenn der Wirkungsgrad des Getriebes ausschlaggebend ist, können Sie diesen in der Variable "<TO>.Actor.Efficiency" einstellen.

Siehe auch

Kraft-/Momentenbegrenzung (Seite 1723)

Technologieobjekt Positionierachse/Gleichlaufachse konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen der Positionierachse/Gleichlaufachse. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen des Technologieobjekts können im Anwenderprogramm unter diesem Namen verwendet werden.

Achstyp

Wenn Sie die Achse ausschließlich virtuell in der CPU verwenden möchten, z. B. als virtuelle Leitachse für den Gleichlauf, aktivieren Sie das Optionskästchen "Virtuelle Achse". Die Konfiguration einer Antriebs- und Geberanbindung ist nicht relevant.

Konfigurieren Sie in dieser Auswahl, ob die Achse lineare oder rotatorische Bewegungen ausführen soll.

Maßeinheiten

Wählen Sie in den Klapplisten die gewünschten Maßeinheiten für Position, Geschwindigkeit, Moment und Kraft der Achse aus.

Modulo

Aktivieren Sie das Optionskästchen "Modulo aktivieren", wenn Sie für die Achse ein wiederkehrendes Maßsystem einsetzen möchten (z. B. 0 bis 360° bei einer Achse vom Achstyp "Rotatorisch").

- **Modulostartwert**
  Definieren Sie in diesem Feld, an welcher Position der Modulobereich beginnen soll (z. B. 0° bei einer Achse vom Achstyp "Rotatorisch").
- **Modulolänge**
  Definieren Sie in diesem Feld die Länge des Modulobereichs (z. B. 360° bei einer Achse vom Achstyp "Rotatorisch").

Simulation

Wenn Sie eine reale Achse im Simulationsbetrieb verfahren möchten, aktivieren Sie das Optionskästchen "Simulation aktivieren".

Im Simulationsbetrieb lassen sich Drehzahl-, Positionier- und Gleichlaufachsen ohne angebundene Antriebe und Geber in der CPU simulieren. Der Simulationsbetrieb ist ab Technologieversion V3.0 auch ohne konfigurierte Antriebs- und Geberanbindung möglich.

Für den Simulationsbetrieb ohne an der CPU angeschlossene Hardware können Sie die Anlaufzeit der CPU über den Parameter "Parametrierungszeit für zentrale und dezentrale Peripherie" beeinflussen. Den Parameter finden Sie in den Eigenschaften der CPU in der Bereichsnavigation "Anlauf".

Siehe auch

Achse in Simulation (Seite 1706)

Maßeinheiten (Seite 1689)

Moduloeinstellung (Seite 1690)

Mechanik (Seite 1710)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Hardware-Schnittstelle

Konfiguration - Antrieb

Konfigurieren Sie im Konfigurationsfenster "Antrieb", welchen Antriebstyp und welchen Antrieb Sie verwenden möchten.

Antriebstyp

Wählen Sie in der Klappliste, ob Sie einen PROFIdrive-Antrieb oder einen Antrieb mit analoger Antriebsanbindung einsetzen möchten.

PROFIdrive-Antriebe werden über ein digitales Kommunikationssystem (PROFINET oder PROFIBUS) mit der Steuerung verbunden. Die Kommunikation erfolgt über PROFIdrive-Telegramme.

Antriebe mit analoger Antriebsanbindung erhalten den Drehzahlsollwert über ein analoges Ausgangssignal (z. B. -10 V bis +10 V) der CPU.

Antriebstyp: PROFIdrive

Datenanbindung

Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Antriebsgerät erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.

Antrieb/Datenbaustein

Wählen Sie im Feld "Antrieb" einen bereits konfigurierten PROFIdrive-Antrieb/Slot aus. Wenn Sie einen PROFIdrive-Antrieb ausgewählt haben, können Sie den PROFIdrive-Antrieb über die Schaltfläche "Gerätekonfiguration" konfigurieren.

Wenn kein PROFIdrive-Antrieb zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie in der Netzsicht einen PROFIdrive-Antrieb hinzu.

Hinweis

Option "Zeige alle Module"

Wenn ein bereits konfigurierter PROFIdrive-Antrieb nicht zur Auswahl steht, zeigen Sie mit der Option "Zeige alle Module" alle erreichbaren Module an.

Wenn Sie die Option "Zeige alle Module" aktivieren, wird für alle angezeigten Module nur der Adressbereich der angezeigten Module überprüft. Wenn der Adressbereich eines Moduls groß genug für das gewählte PROFIdrive-Telegramm ist, können Sie das Modul auswählen. Stellen Sie daher sicher, dass Sie einen PROFIdrive-Antrieb auswählen.

Wenn Sie unter Datenanbindung "Datenbaustein" gewählt haben, wählen Sie hier einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Antriebstyp: Analoge Antriebsanbindung

Analogausgang

Wählen Sie im Feld "Analogausgang" die PLC-Variable des Analogausgangs, über welche der Antrieb angesteuert werden soll.

Um einen Ausgang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Analogausgangsmodul hinzugefügt haben und den PLC-Variablennamen für den Analogausgang definiert haben.

Freigabe-Ausgang aktivieren

Aktivieren Sie das Optionskästchen "Freigabe-Ausgang aktivieren", wenn der Antrieb eine Freigabe unterstützt.

Wählen Sie im entsprechenden Feld die PLC-Variable des Digitalausgangs zur Freigabe des Antriebs. Mit dem Freigabe-Ausgang wird der Drehzahlregler im Antrieb freigegeben, bzw. gesperrt.

Um einen Freigabe-Ausgang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Digitalausgangsmodul hinzugefügt haben und den PLC-Variablennamen für den Digitalausgang definiert haben.

Hinweis

Wenn Sie keinen Freigabe-Ausgang verwenden, kann der Antrieb infolge von Fehlerreaktionen oder Überwachungsfunktionen systemseitig nicht unmittelbar gesperrt werden. Ein kontrolliertes Stoppen des Antriebs ist nicht gewährleistet.

Bereit-Eingang aktivieren

Aktivieren Sie das Optionskästchen "Bereit-Eingang aktivieren", wenn der Antrieb seine Bereitschaft zurückmelden kann.

Wählen Sie im entsprechenden Feld die PLC-Variable des Digitaleingangs, über welchen der Antrieb seine Betriebsbereitschaft an das Technologieobjekt zurückmeldet. Das Leistungsteil ist eingeschaltet und der analoge Drehzahl-Sollwerteingang ist aktiv.

Um einen Bereit-Eingang auswählen zu können, müssen Sie in der Gerätekonfiguration ein Digitaleingangsmodul hinzugefügt haben und den PLC-Variablennamen für den Digitaleingang definiert haben.

Hinweis

Der Freigabe-Ausgang und der Bereit-Eingang können voneinander unabhängig aktiviert werden.

Für den aktivierten Bereit-Eingang gelten folgende Randbedingungen:

- Die Achse wird erst freigegeben ("MC_Power Status" = TRUE), wenn am Bereit-Eingang ein Signal ansteht.
- Wenn bei einer freigegebenen Achse kein Signal am Bereit-Eingang anliegt, wird die Achse mit Fehler gesperrt.
- Wenn die Achse über die Anweisung "MC_Power" gesperrt wird ("Enable" = FALSE), wird die Achse auch mit anstehendem Signal am Bereit-Eingang gesperrt.

Siehe auch

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Geber

Positionierachsen/Gleichlaufachsen benötigen für die Lageregelung einen Lageistwert in Form einer Geberposition. Die Geberposition wird mittels PROFIdrive-Telegramm an die Steuerung übertragen.

Konfigurieren Sie im Konfigurationsfenster "Geber" den Geber und den Gebertyp.

Datenanbindung

Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Geber erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.

Die Auswahl ist nur für Geber möglich, die über PROFIdrive angebunden sind und den Parameter P979 unterstützen.

Geber/Datenbaustein

Wählen Sie in diesem Konfigurationsfeld einen bereits konfigurierten Geber.

Folgende Geber können ausgewählt werden:

- **Geber am Antrieb (nicht bei analoger Antriebsanbindung)**
  Die Konfiguration des Gebers erfolgt über die Konfiguration des PROFIdrive-Antriebs. Der Antrieb wertet die Gebersignale aus und sendet sie im PROFIdrive-Telegramm an die Steuerung.
- **Geber am Technologiemodul (TM)**
  Wählen Sie ein bereits konfiguriertes Technologiemodul und den zu verwendenden Kanal aus. Zur Auswahl werden nur Technologiemodule angezeigt, die auf den Betriebsmodus "Positionserfassung für Motion Control" eingestellt sind.
  Wenn kein Technologiemodul zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie ein Technologiemodul hinzu. Wenn Sie ein Technologiemodul ausgewählt haben, gelangen Sie über die Schaltfläche "Gerätekonfiguration" zur Konfiguration des Technologiemoduls.
  Das Technologiemodul können Sie zentral an einer CPU S7-1500 oder dezentral an einer dezentralen Peripherie betreiben. Beim zentralen Betrieb in der CPU ist kein taktsynchroner Betrieb möglich.
  Welche Technologiemodule zur Positionserfassung für Motion Control geeignet sind, entnehmen Sie der Dokumentation zum Technologiemodul und den Katalogdaten.
- **PROFIdrive-Geber am PROFINET/PROFIBUS (PROFIdrive)**
  Wählen Sie im Feld "PROFIdrive-Geber" einen bereits konfigurierten Geber am PROFINET/PROFIBUS aus. Wenn Sie einen Geber ausgewählt haben, können Sie diesen über die Schaltfläche "Gerätekonfiguration" konfigurieren.
  Wechseln Sie in der Gerätekonfiguration in die Netzsicht und fügen Sie einen Geber hinzu, falls kein Geber zur Auswahl steht.

Hinweis

Option "Zeige alle Module"

Wenn ein bereits konfigurierter PROFIdrive-Geber nicht zur Auswahl steht, zeigen Sie mit der Option "Zeige alle Module" alle erreichbaren Module an.

Wenn Sie die Option "Zeige alle Module" aktivieren, wird für alle angezeigten Module nur der Adressbereich der angezeigten Module überprüft. Wenn der Adressbereich eines Moduls groß genug für das gewählte PROFIdrive-Telegramm ist, können Sie das Modul auswählen. Stellen Sie daher sicher, dass Sie einen PROFIdrive-Geber auswählen.

Wenn Sie unter Datenanbindung "Datenbaustein" ausgewählt haben, wählen Sie hier einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Gebertyp

Wählen Sie in der Klappliste den Gebertyp des Gebers. Folgende Gebertypen stehen zur Verfügung:

- Inkrementell
- Absolut
- Zyklisch absolut

Siehe auch

Konfiguration - Geber (mehrere Geber) (Seite 1863)

Technologieobjekt Positionierachse (Seite 1763)

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Geber (mehrere Geber)

Mit der Technologie-CPU S7-1500T können Sie zwischen 4 Gebersystemen wechseln. Den Wechsel steuern Sie im Anwenderprogramm mit der Motion Control-Anweisung "MC_SetSensor".

Konfigurieren Sie im Konfigurationsfenster "Geber" die möglichen 4 Geber, deren Verwendung und deren Typ.

Geber beim Hochlauf

Wählen Sie in der Klappliste den Geber, der nach dem Anlauf der CPU (STARTUP) aktiv sein soll. Der Geber muss konfiguriert sein und als "verwendet" markiert sein.

Dieser Geber wird nach dem Anlauf der CPU und nach einem Restart des Technologieobjekts (Seite 2016) verwendet. Bei einem Betriebszustandsübergang STOP → RUN der CPU (ohne Restart des Technologieobjekts) wird der Geber weiterverwendet, der auch vor dem STOP aktiv war.

Geber verwenden

Aktivieren Sie das Optionskästchen "Geber verwenden", wenn Sie diesen Geber alternativ für die Lageregelung verwenden möchten.

Siehe auch

Mehrere Geber verwenden (Seite 1700)

Konfiguration - Geber (Seite 1861)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Datenaustausch Antrieb

Konfigurieren Sie im Konfigurationsfenster "Datenaustausch Antrieb" den Datenaustausch mit dem Antrieb.

Die Konfiguration unterscheidet sich je nach gewähltem Antriebstyp:

Antriebstyp: PROFIdrive

Antriebstelegramm

Das in der Gerätekonfiguration eingestellte Telegramm zum Antrieb ist in der Klappliste vorausgewählt.

Automatischer Datenaustausch für Antriebswerte (offline)

Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Antriebs "Bezugsdrehzahl", "Maximale Drehzahl" und "Bezugsmoment" in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.

Automatischer Datenaustausch für Antriebswerte (online)

Aktivieren Sie das Optionskästchen, wenn Sie die online im Antrieb wirksamen Werte "Bezugsdrehzahl", "Maximale Drehzahl" und "Bezugsmoment" zur Laufzeit in die CPU übernehmen möchten. Die Antriebsparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Antriebs oder der CPU vom Bus übernommen.

Alternativ müssen Sie die folgenden Parameter händisch abgleichen:

- **Bezugsdrehzahl**
  Konfigurieren Sie in diesem Feld die Bezugsdrehzahl des Antriebs entsprechend den Angaben des Herstellers. Die Vorgabe der Antriebsdrehzahl erfolgt prozentual zur Bezugsdrehzahl im Bereich -200 % bis 200 %.
- **Maximale Drehzahl**
  Konfigurieren Sie in diesem Feld die maximale Drehzahl des Antriebs.
- **Bezugsmoment**
  Konfigurieren Sie in diesem Feld das Bezugsmoment des Antriebs entsprechend dessen Konfiguration.
  Das Bezugmoment ist zur Kraft-/Momentenreduzierung nötig, welches mit Telegramm 10x unterstützt wird.

Zusatzdaten

Aktivieren Sie das Optionskästchen "Momentendaten", wenn Sie die Datenanbindung der Momentendaten konfigurieren wollen. Wenn Sie einen Antrieb ausgewählt haben, bei dem das Zusatztelegramm 750 projektiert wurde, ist das Optionskästchen "Momentendaten" vorausgewählt.

Datenanbindung

Definieren Sie in der Klappliste, ob die Datenanbindung über Zusatztelegramm oder Datenbaustein erfolgen soll:

- Wenn Sie in der Klappliste "Datenanbindung" den Eintrag "Zusatztelegramm" auswählen, können Sie die Klappliste "Zusatztelegramm" bearbeiten.
- Wenn Sie in der Klappliste "Datenanbindung" den Eintrag "Datenbaustein" auswählen, können Sie einen zuvor erstellten Datenbaustein auswählen, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Zusatztelegrammnummer).

Datenbaustein/Zusatztelegramm

Wählen Sie im Feld "Zusatztelegramm" ein bereits konfiguriertes Zusatztelegramm aus.

Aktivieren Sie das Optionskästchen "Zeige alle Module", wenn Sie sich alle Submodule des angebundenen Antriebs anzeigen lassen wollen. Mit dieser Funktion finden Sie auch selbstdefinierte Zusatztelegramme.

Wählen Sie im Feld "Datenbaustein" den Datenbaustein aus, über den Sie die Momentendaten einbinden möchten.

Hinweis

Die automatische Übernahme der Antriebsparameter ist nur mit SINAMICS-Antrieben ab V4.x möglich. Legen Sie dazu im Konfigurationsfenster "Hardware-Schnittstelle > Antrieb" die Datenanbindung "Antrieb" fest.

Antriebstyp: Analoge Antriebsanbindung

Bezugsdrehzahl

Die Bezugsdrehzahl des Antriebs ist die Drehzahl, mit welcher der Antrieb bei der Ausgabe von 100 % am Analogausgang dreht. Die Bezugsdrehzahl muss am Antrieb konfiguriert werden und in der Konfiguration des Technologieobjekts übernommen werden.

Der bei 100 % ausgegebene Analogwert hängt vom Typ des Analogausgangs ab. Beispielsweise wird bei einem Analogausgang mit +/- 10 V bei 100 % der Wert 10 V ausgegeben.

Analogausgänge lassen sich um etwa 17 % übersteuern. Dies bedeutet, dass ein Analogausgang im Bereich -117 % bis 117 % betrieben werden kann, sofern dies der Antrieb zulässt.

Maximale Drehzahl

Geben Sie in diesem Feld die maximale Drehzahl des Antriebs an.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1698)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Datenaustausch Geber

Konfigurieren Sie im Konfigurationsfenster "Datenaustausch Geber" detaillierte Geberparameter und den Datenaustausch mit dem Geber.

Die Anzeige und Auswahl der hier beschriebenen Konfigurationsparameter ist von folgenden Parametern abhängig:

- Konfigurationsfenster "Grundparameter": Antriebstyp (Linear/Rotatorisch)
- Konfigurationsfenster "Hardware-Schnittstelle > Geber": Gebertyp (Inkrementell/Absolut/ Zyklisch absolut)
- Konfigurationsfenster "Erweiterte Parameter > Mechanik": Geberanbauart

Gebertelegramm

Das in der Gerätekonfiguration eingestellte Telegramm zum Geber ist in der Klappliste vorausgewählt.

Automatischer Datenaustausch für Geberwerte (Offline)

Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Gebers in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Automatischer Datenaustausch für Geberwerte (Online)

Aktivieren Sie das Optionskästchen, wenn Sie die online im Geber wirksamen Werte zur Laufzeit in die CPU übernehmen möchten. Die Geberparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Gebers oder der CPU vom Bus übernommen.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Alternativ müssen Sie die folgenden Parameter, je nach Gebertyp, händisch abgleichen.

Gebertyp

Konfigurieren Sie je nach gewähltem Gebertyp die nachfolgend beschriebenen Parameter:

Gebertyp	Rotatorisch inkrementell
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Rotatorisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl der Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Rotatorisch zyklisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl der Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear inkrementell
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Linear absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear zyklisch absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Siehe auch

Konfiguration - Datenaustausch Geber (mehrere Geber) (Seite 1868)

Technologieobjekt Positionierachse (Seite 1763)

Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1698)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Datenaustausch (Seite 1899)

Konfiguration - Datenaustausch Geber (mehrere Geber)

Einstellungen für

Wählen Sie in der Klappliste den Geber, für den die nachfolgenden Konfigurationen gelten sollen.

Siehe auch

Mehrere Geber verwenden (Seite 1700)

Konfiguration - Datenaustausch Geber (Seite 1865)

Konfiguration - Mechanik (Seite 1869)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Leitwertverschaltungen (nur Gleichlaufachse)

Sie können eine Gleichlaufachse mit mehreren leitwertfähigen Technologieobjekten verschalten. Leitwertfähige Technologieobjekte sind:

- Positionierachsen
- Gleichlaufachsen
- Externe Geber (nur mit CPU S7-1500T)

Zur Laufzeit Ihres Anwenderprogramms können Sie jeweils nur einen Leitwert auswählen.

Alle im Betrieb benötigten Verschaltungen müssen Sie bei der Konfiguration des Technologieobjekts einrichten.

Mögliche Leitwerte

Fügen Sie in der Liste "Leitwertverschaltungen" in der Spalte "Mögliche Leitwerte" alle leitwertfähigen Technologieobjekte hinzu, die Sie im Betrieb als Leitwert für die Gleichlaufachse benötigen.

Die in der Liste hinzugefügten Achsen können Sie mit der entsprechenden Motion Control-Anweisung als Leitwert mit der Gleichlaufachse verschalten.

In der Querverweisliste des Technologieobjekts werden alle konfigurierten Leitwertverschaltungen für das Technologieobjekt angezeigt.

Art der Kopplung

Konfigurieren Sie in der Spalte "Art der Kopplung", ob der Leitwert über Sollwert oder Istwert gekoppelt werden soll.

Die Auswahl "Istwert" ist nur bei der CPU S7-1500T möglich.

Siehe auch

Leitwertkopplung (Seite 1766)

Technologieobjekt Gleichlaufachse (Seite 1764)

Erweiterte Parameter

Konfiguration - Mechanik

Konfiguration - Mechanik

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die Anbauart des Gebers und die Anpassung des Geberistwerts an die mechanischen Gegebenheiten.

Einstellungen für (S7-1500T)

Wählen Sie in der Klappliste den Geber, für den die nachfolgenden Konfigurationen gelten sollen.

Geberanbauart

Wählen Sie in der Klappliste, wie der Geber an der Mechanik angebaut ist.

Die Konfiguration unterscheidet sich entsprechend dem im Konfigurationsfenster "Grundparameter" gewählten Achstyp und der Geberanbauart:

Achstyp: Linear

- Linear - An der Motorwelle (Seite 1870)
- Linear - An der Lastseite (Seite 1871)
- Linear - Externes Messsystem (Seite 1872)

Achstyp: Rotatorisch

- Rotatorisch - An der Motorwelle (Seite 1872)
- Rotatorisch - An der Lastseite (Seite 1873)
- Rotatorisch - Externes Messsystem (Seite 1873)

Geberrichtung invertieren

Aktivieren Sie das Optionskästchen, wenn Sie den Drehsinn des Gebers invertieren müssen.

Siehe auch

Konfiguration - Datenaustausch Geber (mehrere Geber) (Seite 1868)

Technologieobjekt Positionierachse (Seite 1763)

Mehrere Geber verwenden (Seite 1700)

Technologieobjekt Gleichlaufachse (Seite 1764)

Achstyp: Linear

Linear - An der Motorwelle

Der Geber ist mechanisch fest mit der Motorwelle verbunden. Motor und Geber bilden eine Einheit.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Positionsparameter

Konfigurieren Sie im Konfigurationsfeld "Spindelsteigung", um welche Strecke die Last bewegt wird, wenn sich die Spindel um eine Umdrehung dreht.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Linear - An der Lastseite

Der Geber ist mechanisch mit der Lastseite des Getriebes verbunden.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Positionsparameter

Konfigurieren Sie im Konfigurationsfeld "Spindelsteigung", um welche Strecke die Last bewegt wird, wenn sich die Spindel um eine Umdrehung dreht.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Linear - Externes Messsystem

Ein externes Messsystem liefert die Positionswerte der linearen Lastbewegung.

Weg pro Geberumdrehung

Konfigurieren Sie in diesem Konfigurationsfeld den linearen Lastweg pro Geberumdrehung.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Positionsparameter

Konfigurieren Sie im Konfigurationsfeld "Spindelsteigung", um welche Strecke die Last bewegt wird, wenn sich die Spindel um eine Umdrehung dreht.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Achstyp: Rotatorisch

Rotatorisch - An der Motorwelle

Der Geber ist mechanisch fest mit der Motorwelle verbunden. Motor und Geber bilden eine Einheit.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Rotatorisch - An der Lastseite

Der Geber ist mechanisch mit der Lastseite des Getriebes verbunden.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Rotatorisch - Externes Messsystem

Ein externes Messsystem liefert die Positionswerte der rotatorischen Lastbewegung.

Weg pro Geberumdrehung

Konfigurieren Sie in diesem Konfigurationsfeld den linearen Lastweg pro Geberumdrehung.

Antriebsmechanik

Aktivieren Sie das Optionskästchen "Antriebsrichtung invertieren", wenn der Drehsinn des Antriebs invertiert werden soll.

Lastgetriebe

Die Getriebeübersetzung des Lastgetriebes wird als Verhältnis zwischen Motor- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Motorumdrehungen" die ganzzahlige Anzahl der Motorumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Dynamik-Voreinstellungen

Konfigurieren Sie im Konfigurationsfenster "Dynamik-Voreinstellung" die Voreinstellungswerte für Geschwindigkeit, Beschleunigung, Verzögerung und Ruck der Achse.

Die Voreinstellungswerte wirken, wenn an den Motion Control-Anweisungen für die Parameter "Velocity", "Acceleration", "Deceleration" oder "Jerk" Werte < 0 angegeben werden. Die Voreinstellungswerte können einzeln für jeden genannten Parameter übernommen werden.

Die Voreinstellungswerte für Beschleunigung und Verzögerung wirken zusätzlich bei den Verfahrbewegungen des aktiven Referenzierens.

Geschwindigkeit

Definieren Sie in diesem Feld den Voreinstellungswert für die Geschwindigkeit der Achse.

Beschleunigung/Verzögerung - Hochlaufzeit/Rücklaufzeit

Konfigurieren Sie den gewünschten Voreinstellungswert für Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$Hochlaufzeit = \frac{Geschwindigkeit}{Beschleunigung}$$

$$\text{Rücklaufzeit} = \frac{\text{Geschwindigkeit}}{\text{Verzögerung}}$$

Hinweis

Eine Änderung der Geschwindigkeit beeinflusst die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

Verrundungszeit/Ruck

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Ruck" ein. Der Wert 0 bedeutet, dass die Ruckbegrenzung deaktiviert wird.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

  Hinweis

  Der Ruck-Wert ist für Beschleunigungs- und Verzögerungsrampe identisch. Die bei der Verzögerungsrampe wirksame Verrundungszeit ergibt sich aus folgenden Beziehungen:

  - **Beschleunigung > Verzögerung**
    Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung < Verzögerung**
    Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung = Verzögerung**
    Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.

  Im Fehlerfall verzögert die Achse mit der konfigurierten Notstopp-Verzögerung. Eine konfigurierte Ruckbegrenzung wird hierbei nicht berücksichtigt.

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit dem gewählten Ruck ausgeführt.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Geschwindigkeitsprofil (Seite 1717)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Notstopp

Konfigurieren Sie im Konfigurationsfenster "Notstopp" die Notstopp-Verzögerung der Achse. Im Fehlerfall und beim Sperren der Achse mit der Motion Control-Anweisung "MC_Power" (Eingangsparameter "StopMode" = 0) wird die Achse mit dieser Verzögerung zum Stillstand gebracht.

Notstopp-Verzögerung/Notstopp-Rücklaufzeit

Stellen Sie den Verzögerungswert für Notstopp in den Feldern "Notstopp-Verzögerung" oder "Notstopp-Rücklaufzeit" ein.

Den Zusammenhang zwischen Notstopp-Rücklaufzeit und Notstopp-Verzögerung können Sie der folgenden Gleichung entnehmen:

$$\textit{Notstopp-Rücklaufzeit} = \frac{\textit{Maximale Geschwindigkeit}}{\textit{Notstopp-Verzögerung}}$$

Die Konfiguration der Notstopp-Verzögerung bezieht sich auf die konfigurierte maximale Geschwindigkeit der Achse. Wenn die maximale Geschwindigkeit der Achse verändert wird, verändert sich auch der Wert der Notstopp-Verzögerung (die Notstopp-Rücklaufzeit bleibt unverändert).

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Notstopp-Verzögerung (Seite 1719)

Technologieobjekt Gleichlaufachse (Seite 1764)

Begrenzungen

Konfiguration - Positionsgrenzen

Konfigurieren Sie im Konfigurationsfenster "Positionsgrenzen" die Hardware-Endschalter und Software-Endschalter der Achse.

HW-Endschalter aktivieren

Das Optionskästchen aktiviert die Funktion des negativen und positiven Hardware-Endschalters. Der negative Hardware-Endschalter befindet sich auf der Seite in negativer Fahrtrichtung, der positive Hardware-Endschalter auf der Seite in positiver Fahrtrichtung.

Beim Anfahren eines Hardware-Endschalters wird der Technologiealarm 531 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen).

Ausnahme: Wenn ein Hardware-Endschalter während einer aktiven Referenzpunktfahrt mit aktivierter Richtungsumkehr am Hardware-Endschalter überfahren wird, stoppt die Achse mit der konfigurierten maximalen Verzögerung und setzt die Referenzpunktfahrt in umgekehrter Richtung fort.

Hinweis

Verwenden Sie nur Hardware-Endschalter, die nach dem Anfahren dauerhaft geschaltet bleiben. Dieser Schaltzustand darf erst nach der Rückkehr in den zulässigen Verfahrbereich zurückgenommen werden.

Die digitalen Eingänge der Hardware-Endschalter werden standardmäßig im zyklischen Datenaustausch ausgewertet. Wenn die Hardware-Endschalter im Lagereglertakt des Antriebs ausgewertet werden sollen, wählen Sie in den Einstellungen des Eingangsmoduls unter "E/A-Adressen" für "Organisationsbaustein" den Eintrag "MC-Servo" und für "Prozessabbild" den Eintrag "TPA OB Servo".

Eingang negativer/positiver HW-Endschalter

Wählen Sie in den Feldern die PLC-Variable des Digitaleingangs für den negativen und für den positiven Hardware-Endschalter aus.

Um einen Eingang auswählen zu können, muss in der Gerätekonfiguration ein Digitaleingangsmodul hinzugefügt worden sein und der PLC-Variablenname für den Digitaleingang definiert sein.

⚠ VORSICHT

Bei der Anbringung der Hardware-Endschalter muss auf die Filterzeiten der Digitaleingänge geachtet werden.

Aufgrund der Zeit für einen Servotakt und der Filterzeit der Digitaleingänge müssen die resultierenden Verzögerungszeiten berücksichtigt werden.

Die Filterzeit ist bei einzelnen Digitaleingangsmodulen in der Gerätekonfiguration einstellbar.

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt. Bei der Verwendung als Hardware-Endschalter kann dies zu unerwünschten Verzögerungen führen. Verringern Sie in diesem Fall für die entsprechenden Digitaleingänge die Filterzeit.

Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfilter" eingestellt werden.

Pegelauswahl negativer/positiver HW-Endschalter

Wählen Sie in der Klappliste den auslösenden Signalpegel ("Unterer Pegel"/"Oberer Pegel") des Hardware-Endschalters aus. Bei "Unterer Pegel" ist das Eingangssignal "FALSE", wenn der Hardware-Endschalter an- oder überfahren ist. Bei "Oberer Pegel" ist das Eingangssignal "TRUE", wenn der Hardware-Endschalter an- oder überfahren ist.

SW-Endschalter aktivieren

Das Optionskästchen aktiviert die Funktion des unteren und oberen Software-Endschalters. Eine laufende Bewegung kommt bei aktivierten Software-Endschaltern auf der Position des Software-Endschalters zum Stehen. Das Technologieobjekt meldet einen Fehler. Nach Quittierung des Fehlers kann die Achse wieder in Richtung Arbeitsbereich verfahren werden.

Hinweis

Aktivierte Software-Endschalter wirken nur bei referenzierter Achse.

Position negativer/positiver SW-Endschalter

Konfigurieren Sie mit den Positionen des negativen und positiven Software-Endschalters den Arbeitsbereich der Achse.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Dynamikgrenzen

Konfigurieren Sie im Konfigurationsfenster "Dynamikgrenzen" die Maximalwerte für Geschwindigkeit, Beschleunigung, Verzögerung und Ruck der Achse.

Maximale Geschwindigkeit

Definieren Sie in diesem Feld die maximal zugelassene Geschwindigkeit der Achse.

Maximale Beschleunigung/Maximale Verzögerung - Hochlaufzeit/Rücklaufzeit

Stellen Sie die gewünschte Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$\textit{Hochlaufzeit} = \frac{\textit{Maximale Geschwindigkeit}}{\textit{Beschleunigung}}$$

$$\textit{Rücklaufzeit} = \frac{\textit{Maximale Geschwindigkeit}}{\textit{Verzögerung}}$$

Hinweis

Eine Änderung der maximalen Geschwindigkeit beeinflusst die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

Die "Maximale Verzögerung" muss für das aktive Referenzieren mit Richtungsumkehr am Hardware-Endschalter ausreichend groß gewählt werden, um die Achse vor dem Erreichen des mechanischen Anschlags abbremsen zu können.

Verrundungszeit/Ruck

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Ruck" ein. Der Wert 0 bedeutet, dass der Ruck nicht begrenzt wird.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

  Hinweis

  Die eingestellte, und in der Konfiguration angezeigte Verrundungszeit gilt nur für die Beschleunigungsrampe.

  Im Falle, dass sich die Werte von Beschleunigung und Verzögerung unterscheiden, wird die Verrundungszeit der Verzögerungsrampe entsprechend dem Ruck der Beschleunigungsrampe berechnet und verwendet.

  Die Verrundungszeit der Verzögerung wird wie folgt angepasst:

  - **Beschleunigung > Verzögerung**
    Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung < Verzögerung**
    Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung = Verzögerung**
    Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.

  Im Fehlerfall verzögert die Achse mit der konfigurierten Notstopp-Verzögerung. Eine konfigurierte Ruckbegrenzung wird hierbei nicht berücksichtigt.

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahraufträge werden mit dem gewählten Ruck ausgeführt.

Siehe auch

Geschwindigkeitsprofil (Seite 1717)

Dynamikgrenzen im Gleichlauf (Seite 1720)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Momentengrenzen

Konfigurieren Sie im Konfigurationsfenster "Momentenbegrenzung" die Kraft-/ Momentenbegrenzung des Antriebs.

Die Konfiguration steht nur zur Verfügung, wenn ein Antrieb gewählt wurde, der die Kraft-/ Momentenbegrenzung unterstützt und ein Telegramm 10x eingesetzt wird. Die Verwendung des Telegramms 101 ist nicht möglich.

Wirksam

Wählen Sie in der Klappliste, ob der Begrenzungswert "An der Lastseite" oder "An der Motorseite" wirken soll.

Momentengrenzen

Geben Sie in diesem Feld einen Voreinstellungswert für die Momentenbegrenzung in der vorgegebenen Maßeinheit ein.

Der Voreinstellungswert wirkt, wenn die Momentenbegrenzung über die Motion Control-Anweisung "MC_TorqueLimiting" Eingangsparameter "Limit" < 0 vorgegeben wird.

Die Momentenbegrenzung gilt für folgende Achskonfigurationen:

- Achstyp ist "Rotatorisch" und Begrenzungswert wirksam "An der Lastseite" oder "An der Motorseite"
- Achstyp ist "Linear" und Begrenzungswert wirksam "An der Motorseite"

Kraftbegrenzung

Geben Sie in diesem Feld einen Voreinstellungswert für die Kraftbegrenzung in der vorgegebenen Maßeinheit ein.

Der Voreinstellungswert wirkt, wenn die Kraftbegrenzung über die Motion Control-Anweisung "MC_TorqueLimiting" Eingangsparameter "Limit" < 0 vorgegeben wird.

Die Kraftbegrenzung gilt für folgende Achskonfiguration: Achstyp ist "Linear" und Begrenzungswert wirksam "An der Lastseite".

Wenn der Wirkungsgrad von Getriebe und Spindel ausschlaggebend ist, können Sie diesen in der Variable "<TO>.Actor.Efficiency" einstellen.

Positionsbezogene Überwachung

Durch die Kraft-/Momentenbegrenzung am Antrieb kann sich gegebenenfalls ein größerer Schleppfehler einstellen oder der Stillstand der Achse wird in der Positionierüberwachung nicht zuverlässig erkannt.

Wählen Sie die Option "Positionsbezogenen Überwachungen deaktivieren" aus, um die Überwachung des Schleppfehlers und die Positionierüberwachung während einer Kraft-/Momentenbegrenzung zu deaktivieren. Wenn Sie die positionsbezogene Überwachung aktivieren wollen, wählen Sie die Option "Positionsbezogene Überwachungen aktiv lassen" aus.

Verschaltung im SINAMICS-Antrieb

Die folgende Verschaltung ist im SINAMICS-Antrieb erforderlich:

- P1522 auf einen Festwert von +100 %
- P1523 auf einen Festwert von -100 % (z. B. durch Verschaltung auf Festwertparameter P2902[i])

Siehe auch

Kraft-/Momentenbegrenzung (Seite 1723)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Festanschlagserkennung

Konfigurieren Sie im Konfigurationsfenster die Festanschlagserkennung.

Bei Aktivierung der Festanschlagserkennung durch die Motion Control-Anweisung "MC_TorqueLimiting" und einem lagegeregelten Bewegungsauftrag kann ein "Fahren auf Festanschlag" realisiert werden. Der Vorgang wird auch als Klemmen bezeichnet.

Positioniertoleranz

Konfigurieren Sie in diesem Konfigurationsfeld die Positioniertoleranz, deren Überschreitung als Wegbrechen oder Zurückdrücken des Festanschlags gewertet wird. Um das Wegbrechen oder Zurückdrücken des Festanschlags zu erkennen, muss sich die Sollposition außerhalb der Positioniertoleranz befinden. Die konfigurierte Positionstoleranz muss kleiner als der konfigurierte Schleppabstand sein.

Schleppabstand

Wenn der Antrieb während eines Bewegungsauftrags durch einen mechanischen Festanschlag gestoppt wird, vergrößert sich der Schleppfehler. Der sich aufbauende Schleppabstand dient als Kriterium zur Festanschlagserkennung. Konfigurieren Sie im Konfigurationsfeld "Schleppabstand" den Wert des Schleppabstands, ab welchem die Festanschlagserkennung wirken soll. Der konfigurierte Schleppabstand muss größer als die konfigurierte Positionstoleranz sein.

Hinweis

Wenn in der Konfiguration der Positionsüberwachungen die Schleppfehlerüberwachung aktiviert wurde, muss der dort konfigurierte "Maximale Schleppfehler" größer sein als der "Schleppabstand" der Festanschlagserkennung.

Siehe auch

Festanschlagserkennung (Seite 1725)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren

Unter Referenzieren versteht man das Abgleichen des Positionswerts eines Technologieobjekts auf die reale, physikalische Position des Antriebs. Nur mit einer referenzierten Achse können absolute Zielpositionen der Achse angefahren werden.

Betriebsarten der Motion Control-Anweisung "MC_Home"

Bei S7-1500 Motion Control wird die Achse mit der Motion Control-Anweisung "MC_Home" referenziert. Dabei werden folgende Betriebsarten unterschieden:

- **Aktives Referenzieren (Inkrementeller Geber)**
  Beim aktiven Referenzieren führt die Motion Control-Anweisung "MC_Home" die konfigurierte Referenzpunktfahrt durch. Laufende Verfahrbewegungen werden abgebrochen. Beim Erkennen der Referenzmarke wird die Position der Achse entsprechend der Konfiguration gesetzt.
- **Passives Referenzieren (Inkrementeller Geber)**
  Beim passiven Referenzieren führt die Motion Control-Anweisung "MC_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Laufende Verfahrbewegungen werden beim Start des passiven Referenzierens nicht abgebrochen. Beim Erkennen der Referenzmarke wird die Achse entsprechend der Konfiguration gesetzt.

- **Direktes Referenzieren Absolut (Inkrementeller Geber oder Absolutwertgeber)**
  Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Der Wert des Eingangsparameters "Position" der Motion Control-Anweisung "MC_Home" wird sofort als aktuelle Istposition der Achse gesetzt.
- **Direktes Referenzieren Relativ (Inkrementeller Geber oder Absolutwertgeber)**
  Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Für die Achsposition nach dem Referenzieren gilt:
  Neue Achsposition = Aktuelle Achsposition + Wert des Parameters "Position" der Anweisung "MC_Home".

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Aktives Referenzieren

Konfiguration - Aktives Referenzieren

Konfigurieren Sie im Konfigurationsfenster "Aktives Referenzieren" die Parameter für das aktive Referenzieren. "Aktives Referenzieren" wird über die Motion Control-Anweisung "MC_Home" mit "Mode" = 3 und 5 ausgeführt.

Hinweis

Parameter "MC_Home.Mode" (CPU S7-1500)

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Einstellungen für

Wählen Sie in der Klappliste den Geber, für den die Referenziereinstellungen gelten sollen (nur für S7-1500T).

Auswahl Referenziermodus

Wählen Sie in der Auswahl unter den nachfolgenden Referenziermodi:

- Nullmarke über PROFIdrive-Telegramm verwenden (Seite 1884)
- Referenznocken und Nullmarke über PROFIdrive-Telegramm verwenden (Seite 1885)
- Referenzmarke über Digitaleingang verwenden (Seite 1886)

Siehe auch

Referenzieren (Seite 1729)

Mehrere Geber verwenden (Seite 1700)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden"

Richtungsumkehr am Hardware-Endschalter aktivieren

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Wenn der Hardware-Endschalter während des aktiven Referenzierens erreicht wird, bremst die Achse mit der konfigurierten maximalen Verzögerung ab und führt eine Richtungsumkehr durch. Die Nullmarke wird anschließend in umgekehrter Richtung gesucht. Ist diese Funktion nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird der Antrieb gesperrt und mit der am Antrieb konfigurierten Rampe gebremst.

Referenzierrichtung

Wählen Sie, in welcher Richtung die nächste Nullmarke zum Referenzieren angefahren werden soll.

"Positiv" ist die Referenzierrichtung in Richtung positiver Positionswerte, "Negativ" in Richtung negativer Positionswerte.

Anfahrgeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, die zum Herausfahren der Referenzpunktverschiebung verwendet wird.

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher die Achse zum Referenzieren in die Nullmarke einfahren soll.

Referenzpunktverschiebung

Geben Sie bei unterschiedlicher Position von Nullmarke und Referenzpunktposition in diesem Feld die entsprechende Referenzpunktverschiebung ein. Die Achse fährt die Referenzpunktposition mit der Anfahrgeschwindigkeit an.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 5 ausgeführt wird.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Referenziermodus "Referenznocken und Nullmarke über PROFIdrive-Telegramm verwenden"

Richtungsumkehr am Hardwareendschalter aktivieren

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Wenn der Hardware-Endschalter während des aktiven Referenzierens erreicht wird, bremst die Achse mit der konfigurierten maximalen Verzögerung ab und führt eine Richtungsumkehr durch. Der Referenznocken wird anschließend in umgekehrter Richtung gesucht. Ist diese Funktion nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird der Antrieb gesperrt und mit der am Antrieb konfigurierten Rampe gebremst.

Anfahrrichtung

Wählen Sie die Anfahrrichtung zur Suche des Referenznockens aus.

"Positiv" ist die Anfahrrichtung in Richtung positiver Positionswerte, "Negativ" in Richtung negativer Positionswerte.

Referenzierrichtung

Wählen Sie in welcher Richtung die Nullmarke zum Referenzieren angefahren werden soll.

Anfahrgeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenznocken während Referenzpunktfahrt gesucht wird. Eine eventuell eingestellte Referenzpunktverschiebung wird mit der gleichen Geschwindigkeit herausgefahren.

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher die Achse zum Referenzieren in die Nullmarke einfahren soll. Zur Nullmarkenerkennung muss der Referenznocken verlassen sein.

Referenzpunktverschiebung

Geben Sie bei unterschiedlicher Position von Nullmarke und Referenzpunktposition in diesem Feld die entsprechende Referenzpunktverschiebung ein. Die Achse fährt die Referenzpunktposition mit der Anfahrgeschwindigkeit an.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 5 ausgeführt wird.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Referenziermodus "Referenzmarke über Digitaleingang verwenden"

Wenn ein Digitaleingang als Referenzmarke verwendet wird, ist die Genauigkeit des Referenziervorgangs nicht so hoch wie bei Hardware-unterstütztem Referenzieren über Nullmarken.

Sie können die Genauigkeit durch kleine Referenziergeschwindigkeit verbessern.

Achten Sie auch auf die Einstellung kurzer Filterzeiten am Digitaleingang.

Digitaleingang Referenzmarke/-nocken

Wählen Sie in diesem Konfigurationsfeld die PLC-Variable des Digitaleingangs, der als Referenzmarke (Referenznocken) wirken soll. Wählen Sie zusätzlich den Pegel, bei welchem die Referenzmarke erkannt werden soll.

Um einen Eingang auswählen zu können, muss in der Gerätekonfiguration ein Digitaleingangsmodul hinzugefügt worden sein und der PLC-Variablenname für den Digitaleingang definiert sein.

Richtungsumkehr am Hardware-Endschalter aktivieren

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Wenn der Hardware-Endschalter während des aktiven Referenzierens erreicht wird, bremst die Achse mit der konfigurierten maximalen Verzögerung ab und führt eine Richtungsumkehr durch. Die Referenzmarke wird anschließend in umgekehrter Richtung gesucht. Ist diese Funktion nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird der Antrieb gesperrt und mit der am Antrieb konfigurierten Rampe gebremst.

Anfahrrichtung

Wählen Sie die Anfahrrichtung zur Suche der Referenzmarke aus.

"Positiv" ist die Anfahrrichtung in Richtung positiver Positionswerte, "Negativ" in Richtung negativer Positionswerte.

Referenzierrichtung

Wählen Sie, in welcher Richtung die Referenzmarke zum Referenzieren angefahren werden soll.

Referenzmarke

Wählen Sie, welche Schaltposition des "Digitaleingangs" als Referenzmarke verwendet werden soll.

Beim Überfahren eines "Digitaleingangs" werden zwei Schaltflanken erzeugt, die räumlich auseinander liegen. Mit der Wahl der positiven oder negativen Seite wird sichergestellt, dass die Referenzmarke immer an der gleichen mechanischen Position ausgewertet wird.

Die positive Seite ist die Schaltposition mit einem größeren Positionswert, die negative Seite ist die Schaltposition mit dem kleineren Positionswert.

Die Auswahl der Seite ist unabhängig von der Fahrtrichtung und unabhängig davon, ob sie eine steigende oder eine fallende Flanke bewirkt.

Anfahrgeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der "Digitaleingang" während Referenzpunktfahrt gesucht werden soll. Eine eventuell eingestellte Referenzpunktverschiebung wird mit der gleichen Geschwindigkeit herausgefahren.

Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher die Achse zum Referenzieren in den Referenzpunkt einfahren soll.

Referenzpunktverschiebung

Geben Sie bei unterschiedlicher Position von Referenzpunkt und Referenzpunktposition in diesem Feld die entsprechende Referenzpunktverschiebung ein. Die Achse fährt die Referenzpunktposition mit der Anfahrgeschwindigkeit an.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 5 ausgeführt wird.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Passives Referenzieren

Konfiguration - Passives Referenzieren

Konfigurieren Sie im Konfigurationsfenster "Passives Referenzieren" (fliegendes Referenzieren) die Parameter für das passive Referenzieren. Die Referenzierfunktion "Passives Referenzieren" wird über die Motion Control-Anweisung "MC_Home" mit "Mode" = 2, 8 und 10 ausgeführt.

Hinweis

Parameter "MC_Home.Mode" (CPU S7-1500)

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Einstellungen für

Wählen Sie in der Klappliste den Geber, für den die Referenziereinstellungen gelten sollen (nur für S7-1500T).

Auswahl Referenziermodus

Wählen Sie in der Auswahl unter den nachfolgenden Referenziermodi:

- Nullmarke über PROFIdrive-Telegramm verwenden (Seite 1889)
- Referenznocken und Nullmarke über PROFIdrive-Telegramm verwenden (Seite 1890)
- Referenzmarke über Digitaleingang verwenden (Seite 1891)

Siehe auch

Referenzieren (Seite 1729)

Mehrere Geber verwenden (Seite 1700)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden"

Referenzierrichtung

Wählen Sie in welcher Richtung die nächste Nullmarke zum Referenzieren angefahren werden soll. Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Die Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Die Achse bewegt sich in Richtung niedrigerer Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Referenziermodus "Referenznocken und Nullmarke über PROFIdrive-Telegramm verwenden"

Referenzierrichtung

Wählen Sie, in welcher Richtung die Nullmarke zum Referenzieren angefahren werden soll. Die nächste Nullmarke nach Verlassen des Referenznockens wird verwendet.

Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Die Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Die Achse bewegt sich in Richtung niedrigerer Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Referenziermodus "Referenzmarke über Digitaleingang verwenden"

Digitaleingang Referenzmarke/-nocken

Wählen Sie in diesem Dialogfeld einen Digitaleingang, der als Referenzmarke (Referenznocken) wirken soll. Wählen Sie zusätzlich den Pegel, bei welchem die Referenzmarke erkannt werden soll.

Referenzierrichtung

Wählen Sie in welcher Richtung die Referenzmarke zum Referenzieren angefahren werden soll.

Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Die Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Die Achse bewegt sich in Richtung niedriger Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzmarke

Wählen Sie welche Schaltposition des "Digitaleingangs" als Referenzmarke verwendet werden soll.

Beim Überfahren eines "Digitaleingangs" werden zwei Schaltflanken erzeugt, die räumlich auseinander liegen. Mit der Wahl der positiven oder negativen Seite wird sichergestellt, dass die Referenzmarke immer an der gleichen mechanischen Position ausgewertet wird.

Die positive Seite ist die Schaltposition mit einem größeren Positionswert, die negative Seite ist die Schaltposition mit dem kleineren Positionswert.

Die Auswahl der Seite ist unabhängig von der Fahrtrichtung und unabhängig davon, ob sie eine steigende oder eine fallende Flanke bewirkt.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Referenzieren (Seite 1729)

Positionsüberwachungen

Konfiguration - Positionierüberwachung

Konfigurieren Sie im Konfigurationsfenster "Positionierüberwachung" die Kriterien für die Überwachung der Zielposition.

Positionierfenster

Konfigurieren Sie in diesem Feld die Größe des Positionierfensters. Wenn sich die Achse innerhalb dieses Fensters befindet, gilt die Position als "erreicht".

Toleranzzeit

Konfigurieren Sie in diesem Feld die Toleranzzeit, in welcher der Positionswert das Positionierfenster erreichen muss.

Minimale Verweildauer im Positionierfenster

Konfigurieren Sie in diesem Feld die minimale Verweildauer. Der aktuelle Positionswert muss sich mindestens für die "Minimale Verweildauer" im Positionierfenster befinden.

Wenn eines der Kriterien verletzt wird, wird die Achse gestoppt und der Technologie-Alarm 541 "Fehler Positionierüberwachung" angezeigt (Alarmreaktion: Freigabe wegnehmen).

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Positionsüberwachungen (Seite 1749)

Konfiguration - Schleppfehler

Konfigurieren Sie im Konfigurationsfenster "Schleppfehler" die zulässige Abweichung der Ist-Position der Achse zur Soll-Position. Der Schleppfehler kann dynamisch zur aktuellen Geschwindigkeit der Achse angepasst werden.

Schleppfehlerüberwachung aktivieren

Aktivieren Sie das Optionskästchen, wenn Sie die Schleppfehlerüberwachung aktivieren möchten. Bei aktivierter Schleppfehlerüberwachung wird die Achse im Fehler-Bereich (orange) gestoppt. Der Technologie-Alarm 521 "Schleppfehler" wird angezeigt (Alarmreaktion: Freigabe wegnehmen).

Bei deaktivierter Schleppfehlerüberwachung sind die eingestellten Grenzen ohne Wirkung.

Maximaler Schleppfehler

Konfigurieren Sie in diesem Feld den Schleppfehler, der bei maximaler Geschwindigkeit zulässig ist.

Warnpegel

Konfigurieren Sie in diesem Feld einen Prozentwert der aktuellen Schleppfehlergrenze, ab der eine Warnung ausgegeben werden soll.

Beispiel: Der aktuelle maximale Schleppfehler beträgt 100 mm. Der Warnpegel ist auf 90 % konfiguriert. Wenn der aktuelle Schleppfehler einen Wert von 90 mm überschreitet, wird der Technologie-Alarm 522 "Warnung Schleppfehlertoleranz" ausgegeben. Dies ist eine Warnung und beinhaltet keine Alarmreaktion.

Schleppfehler

Konfigurieren Sie in diesem Feld den für geringe Geschwindigkeiten zulässigen Schleppfehler (ohne dynamische Anpassung des Schleppfehlers).

Beginn der dynamischen Anpassung

Konfigurieren Sie in diesem Feld, ab welcher Geschwindigkeit der Schleppfehler dynamisch angepasst werden soll. Ab dieser Geschwindigkeit wird der Schleppfehler bis zur maximalen Geschwindigkeit auf den maximalen Schleppfehler angepasst.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Positionsüberwachungen (Seite 1749)

Konfiguration - Stillstandssignal

Konfigurieren Sie im Konfigurationsfenster "Stillstandssignal" die Kriterien zur Stillstandserkennung.

Stillstandsfenster

Konfigurieren Sie in diesem Feld die Größe des Stillstandsfensters. Zur Stillstandsanzeige muss die Geschwindigkeit der Achse innerhalb dieses Fensters sein.

Minimale Verweildauer im Stillstandsfenster

Konfigurieren Sie in diesem Feld die minimale Verweildauer im Stillstandsfenster. Die Geschwindigkeit der Achse muss sich mindestens für die angegebene Dauer im Stillstandsfenster befinden.

Wenn beide Kriterien erfüllt sind, wird der Stillstand der Achse angezeigt.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Stillstandssignal (Seite 1751)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Regelkreis

Konfigurieren Sie im Konfigurationsfenster "Regelkreis" die Vorsteuerung und die Verstärkung Kv des Lageregelkreises.

Der Kv-Faktor wirkt sich auf folgende Kenngrößen aus:

- Positioniergenauigkeit und Halteregelung
- Gleichförmigkeit der Bewegung
- Positionierzeit

Je besser die konstruktiven Voraussetzungen der Achse sind (große Steifigkeit), desto größer kann der Kv-Faktor eingestellt werden. Damit verringert sich der Schleppfehler und eine höhere Dynamik wird erreicht.

Grundlegende Informationen finden Sie im Kapitel "Regelung (Seite 1753)".

Vorsteuerung

Konfigurieren Sie in diesem Feld die prozentuale Geschwindigkeitsvorsteuerung.

Drehzahl-Regelkreis-Ersatzzeit

Konfigurieren Sie in diesem Feld Drehzahl-Regelkreis-Ersatzzeit (T_{vtc}).

Bei aktivierter Geschwindigkeitsvorsteuerung wird der Lagesollwert vor Bildung der Regeldifferenz um die Drehzahl-Regelkreis-Ersatzzeit verzögert. Damit wird eine Übersteuerung oder Voreilen der Istposition gegen die Sollposition verhindert. Die Drehzahl-Regelkreis-Ersatzzeit ist ein vereinfachtes Ersatzmodell des dynamischen Verhaltens des Drehzahlregelkreises. Die Drehzahl-Regelkreis-Ersatzzeit wird im Symmetrierfilter eingerechnet.

Verstärkung (Kv-Faktor)

Tragen Sie in dem Eingabefeld die Verstärkung Kv des Lageregelkreises ein.

Dynamic Servo Control (DSC)

Bei lagegeregelten Achsen (Positionierachsen/Gleichlaufachsen) kann die Lageregelung entweder in der CPU oder im Antrieb erfolgen, falls dieser Dynamic Servo Control (DSC) unterstützt. Wählen Sie das von Ihnen gewünschte Regelungsverfahren:

- **Lageregelung im Antrieb (DSC aktiviert)**
  Bei der Funktion Dynamic Servo Control (DSC) wird der Lageregler im Antrieb im Takt des Drehzahlregelkreises ausgeführt. Die Einstellung eines wesentlich größeren Lageregler-Verstärkungsfaktor Kv wird damit ermöglicht. Dies erhöht die Dynamik für Führungsgrößenfolge und Störgrößenausregelung bei hochdynamischen Antrieben.
- **Lageregelung in der PLC**

Hinweis

Dynamic Servo Control (DSC) ist nur mit einem der folgenden PROFIdrive-Telegramme möglich:

- Standardtelegramm 5 oder 6
- SIEMENS-Telegramm 105 oder 106

Siehe auch

Regelung (Seite 1753)

Regelungsstruktur (Seite 1754)

Funktion und Aufbau der Optimierung (Seite 2027)

Technologieobjekt Positionierachse (Seite 1763)

Technologieobjekt Gleichlaufachse (Seite 1764)

Konfiguration - Istwertextrapolation

Konfigurieren Sie im Konfigurationsfenster "Istwertextrapolation" die Eigenschaften der Extrapolation bei einer Istwertkopplung für den Gleichlauf. Die hier eingestellten Werte gelten nur, wenn die Istwerte dieser Achse als Leitwert verwendet werden.

Positionsfilter T1 und T2

Geben Sie die Zeitkonstanten für die Filterung der Istposition ein.

Geschwindigkeitsfilter T1 und Toleranzbandbreite

Geben Sie die Zeitkonstante für die Glättung der Geschwindigkeit und die Toleranzbandbreite ein.

Für eine optimierte Anwendung des Toleranzbandes geben Sie die gleiche Breite für das Toleranzband ein, wie die Breite des Rausschsignals ist.

Folgeachsbedingte Extrapolationszeit

Geben Sie den durch die Folgeachse bedingten Anteil für die Berechnung der Extrapolationszeit ein.

Hysteresewert

Geben Sie einen Wert für die Anwendung der Hysteresefunktion auf den extrapolierten Istwert ein. Die Angabe erfolgt in der konfigurierten Längeneinheit.

Siehe auch

Technologieobjekt Positionierachse (Seite 1763)

Extrapolation der Leitwerte bei Istwertkopplung (Seite 1767)

Technologieobjekt Gleichlaufachse (Seite 1764)

Technologieobjekt Externer Geber konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen des Externen Gebers. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen des Externen Gebers können im Anwenderprogramm unter diesem Namen verwendet werden.

Typ Externer Geber

Konfigurieren Sie in dieser Auswahl, ob der Externe Geber lineare oder rotatorische Bewegungen aufnimmt.

Maßeinheiten

Wählen Sie in den Klapplisten die gewünschten Maßeinheiten für Position und Geschwindigkeit des Externen Gebers aus.

Modulo

Aktivieren Sie das Optionskästchen "Modulo aktivieren", wenn Sie für den Externen Geber ein wiederkehrendes Maßsystem einsetzen möchten (z. B. 0-360° bei einem Externen Geber vom Typ "Rotatorisch").

- **Modulostartwert**
  Definieren Sie in diesem Feld, an welcher Position der Modulobereich beginnen soll (z. B. 0° bei einem Externen Geber Typ "Rotatorisch").
- **Modulolänge**
  Definieren Sie in diesem Feld die Länge des Modulobereichs (z. B. 360° bei einem Externen Geber vom Typ "Rotatorisch").

Siehe auch

Maßeinheiten (Seite 1689)

Moduloeinstellung (Seite 1690)

Technologieobjekt Externer Geber (Seite 1780)

Hardware-Schnittstelle

Konfiguration - Geber

Der Externe Geber nimmt die Position eines extern angesteuerten Antriebs auf. Der hierzu benötigte Geber übermittelt die Geberposition mittels PROFIdrive-Telegramm an die Steuerung. Konfigurieren Sie im Konfigurationsfenster "Geber" den Geber und den Gebertyp.

Datenanbindung

Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Geber erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.

Die Auswahl ist nur für Geber möglich, die über PROFIdrive angebunden sind und den Parameter P979 unterstützen.

Geber/Datenbaustein

Wählen Sie in diesem Bereich den Geber, den Sie in der Gerätekonfiguration konfiguriert haben.

Folgende Geber können verwendet werden:

- **Geber am Technologiemodul (TM)**
  Wählen Sie im Konfigurationsfeld "Geber" ein bereits konfiguriertes Technologiemodul und den zu verwendenden Kanal aus. Zur Auswahl werden nur Technologiemodule angezeigt, die auf den Betriebsmodus "Positionserfassung für Motion Control" eingestellt sind.
  Das Technologiemodul können Sie zentral an einer CPU S7-1500 oder dezentral an einer dezentralen Peripherie betreiben. Beim zentralen Betrieb in der CPU ist kein taktsynchroner Betrieb möglich.
  Welche Technologiemodule zur Positionserfassung für Motion Control geeignet sind, entnehmen Sie der Dokumentation zum Technologiemodul und den Katalogdaten.
- **Geber über PROFINET/PROFIBUS (PROFIdrive)**
  Wählen Sie im Konfigurationsfeld "Geber" einen bereits konfigurierten Geber am PROFINET/PROFIBUS aus.

Hinweis

Option "Zeige alle Module"

Wenn ein bereits konfigurierter PROFIdrive-Geber nicht zur Auswahl steht, zeigen Sie mit der Option "Zeige alle Module" alle erreichbaren Module an.

Wenn Sie die Option "Zeige alle Module" aktivieren, wird für alle angezeigten Module nur der Adressbereich der angezeigten Module überprüft. Wenn der Adressbereich eines Moduls groß genug für das gewählte PROFIdrive-Telegramm ist, können Sie das Modul auswählen. Stellen Sie daher sicher, dass Sie einen PROFIdrive-Geber auswählen.

Wenn Sie unter Datenanbindung "Datenbaustein" ausgewählt haben, wählen Sie hier einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Gebertyp

Wählen Sie in diesem Konfigurationsfeld den Gebertyp. Folgende Gebertypen stehen zu Auswahl:

- Inkrementeller Geber
- Absolutwertgeber
- Absolutwertgeber zyklisch absolut

Siehe auch

Technologieobjekt Externer Geber (Seite 1780)

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Antrieb/Geber über Datenbaustein anbinden (Seite 1976)

Konfiguration - Datenaustausch

Konfiguration - Datenaustausch

Konfigurieren Sie im Konfigurationsfenster "Datenaustausch Geber" detaillierte Geberparameter und den Datenaustausch mit dem Geber. Die Konfiguration unterscheidet sich entsprechend der Geberkopplung:

- Geber am Technologiemodul (Seite 1899)
- Geber am PROFINET/PROFIBUS (Seite 1902)

Siehe auch

Technologieobjekt Externer Geber (Seite 1780)

Geber am Technologiemodul

Datenaustausch Geber

Konfigurieren Sie in diesem Bereich das Gebertelegramm und die Kriterien, wie die Geberdaten auszuwerten sind. Die Angaben müssen mit den Angaben in der Gerätekonfiguration übereinstimmen.

Gebertelegramm

Wählen Sie in der Klappliste für das Technologiemodul das Telegramm, welches Sie am Technologiemodul konfiguriert haben.

Automatischer Datenaustausch für Geberwerte (Offline)

Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Gebers in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Automatischer Datenaustausch für Geberwerte (Online)

Aktivieren Sie das Optionskästchen, wenn Sie die online im Geber wirksamen Werte zur Laufzeit in die CPU übernehmen möchten. Die Geberparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Gebers oder der CPU vom Bus übernommen.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Gebertyp

Konfigurieren Sie abhängig vom ausgewählten Gebertyp die nachfolgend beschriebenen Parameter. Die Angaben müssen mit den Angaben in der Gerätekonfiguration übereinstimmen.

Gebertyp	Rotatorisch inkrementell
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Rotatorisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Rotatorisch zyklisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.

Gebertyp	Rotatorisch zyklisch absolut
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear inkrementell
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Linear absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear zyklisch absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Siehe auch

Konfiguration - Datenaustausch (Seite 1899)

Geber am PROFINET/PROFIBUS (Seite 1902)

Technologieobjekt Externer Geber (Seite 1780)

Geber am PROFINET/PROFIBUS

Datenaustausch Geber

Konfigurieren Sie in diesem Bereich das Gebertelegramm und die Kriterien, wie die Geberdaten auszuwerten sind. Die Angaben müssen mit den Angaben in der Gerätekonfiguration übereinstimmen.

Gebertelegramm

Das in der Gerätekonfiguration eingestellte Telegramm zum Geber ist in der Klappliste vorausgewählt.

Automatischer Datenaustausch für Geberwerte (Offline)

Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Gebers in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Automatischer Datenaustausch für Geberwerte (Online)

Aktivieren Sie das Optionskästchen, wenn Sie die online im Geber wirksamen Werte zur Laufzeit in die CPU übernehmen möchten. Die Geberparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Gebers oder der CPU vom Bus übernommen.

Hinweis

Die automatische Übernahme der Geberparameter ist nur mit PROFIdrive-Gebern ab Ausgabestand A16 möglich. Dazu muss im Konfigurationsfenster "Hardware-Schnittstelle > Geber" als Datenanbindung "Geber" gewählt sein.

Gebertyp

Konfigurieren Sie abhängig vom ausgewählten Gebertyp die nachfolgend beschriebenen Parameter:

Gebertyp	Rotatorisch inkrementell
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Rotatorisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl der Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Rotatorisch zyklisch absolut
Inkremente pro Umdrehung	Konfigurieren Sie in diesem Feld die Anzahl der Inkremente, die der Geber pro Umdrehung auflöst.
Anzahl der Umdrehungen	Konfigurieren Sie in diesem Feld die Anzahl der Umdrehungen, die der Absolutwertgeber erfassen kann.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear inkrementell
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).

Gebertyp	Linear absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Gebertyp	Linear zyklisch absolut
Abstand zwischen zwei Inkrementen	Konfigurieren Sie in diesem Feld den Weg zwischen zwei Inkrementen des Gebers.
Bits für Feinauflösung im inkrementellen Istwert (Gx_XIST1)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des inkrementellen Istwerts (Gx_XIST1).
Bits für Feinauflösung im absoluten Istwert (Gx_XIST2)	Konfigurieren Sie in diesem Feld die Anzahl der Bits für die Feinauflösung innerhalb des absoluten Istwerts (Gx_XIST2).

Siehe auch

Konfiguration - Datenaustausch (Seite 1899)

Geber am Technologiemodul (Seite 1899)

Technologieobjekt Externer Geber (Seite 1780)

Erweiterte Parameter

Konfiguration - Mechanik

Konfiguration - Mechanik

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die Geberparameter zur Erfassung der Position des extern angesteuerten Antriebs.

Die Konfiguration unterscheidet sich entsprechend dem Gebertyp:

- Linear (Seite 1904)
- Rotatorisch (Seite 1905)

Siehe auch

Technologieobjekt Externer Geber (Seite 1780)

Mechanik (Seite 1710)

Linear

Geber

Aktivieren Sie das Optionskästchen "Geberrichtung invertieren", wenn Sie den Istwert des Gebers invertieren möchten.

Lastgetriebe

Die Getriebeübersetzung des Messgetriebes wird als Verhältnis zwischen Geber- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Geberumdrehungen" die ganzzahlige Anzahl der Geberumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Wählen Sie für die Anzahl der Geber- und Lastumdrehungen gleiche Werte, wenn kein Lastgetriebe vorhanden ist.

Positionsparameter

Konfigurieren Sie im Konfigurationsfeld "Spindelsteigung", um welche Strecke die Last bewegt wird, wenn sich die Spindel um eine Umdrehung dreht.

Siehe auch

Konfiguration - Mechanik (Seite 1904)

Rotatorisch (Seite 1905)

Technologieobjekt Externer Geber (Seite 1780)

Mechanik (Seite 1710)

Rotatorisch

Geber

Aktivieren Sie das Optionskästchen "Geberrichtung invertieren", wenn Sie den Istwert des Gebers invertieren möchten.

Lastgetriebe

Die Getriebeübersetzung des Messgetriebes wird als Verhältnis zwischen Geber- und Lastumdrehungen angegeben.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Geberumdrehungen" die ganzzahlige Anzahl der Geberumdrehungen.

Konfigurieren Sie im Konfigurationsfeld "Anzahl Lastumdrehungen" die ganzzahlige Anzahl der Lastumdrehungen.

Wählen Sie für die Anzahl der Geber- und Lastumdrehungen gleiche Werte, wenn kein Lastgetriebe vorhanden ist.

Siehe auch

Konfiguration - Mechanik (Seite 1904)

Linear (Seite 1904)

Technologieobjekt Externer Geber (Seite 1780)

Mechanik (Seite 1710)

Referenzieren

Konfiguration - Referenzieren

Konfigurieren Sie im Konfigurationsfenster "Referenzieren" die Parameter zum Referenzieren des Externen Gebers. Das Referenzieren wird über die Motion Control-Anweisung "MC_Home" mit "Mode" = 2, 8 und 10 ausgeführt.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Auswahl Referenziermodus

Wählen Sie in der Auswahl unter den nachfolgenden Referenziermodi:

- Referenznocken und Nullmarke über PROFIdrive-Telegramm (Seite 1906)
- Nullmarke über PROFIdrive-Telegramm (Seite 1907)
- Referenzmarke über Digitaleingang (Seite 1908)

Siehe auch

Technologieobjekt Externer Geber (Seite 1780)

Referenznocken und Nullmarke über PROFIdrive-Telegramm

Referenzierrichtung

Wählen Sie in welcher Richtung die Nullmarke zum Referenzieren angefahren werden soll. Die nächste Nullmarke nach Verlassen des Referenznockens wird verwendet.

Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Achse bewegt sich in Richtung niedriger Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Konfiguration - Referenzieren (Seite 1906)

Nullmarke über PROFIdrive-Telegramm (Seite 1907)

Referenzmarke über Digitaleingang (Seite 1908)

Technologieobjekt Externer Geber (Seite 1780)

Nullmarke über PROFIdrive-Telegramm

Referenzierrichtung

Wählen Sie in welcher Richtung die nächste Nullmarke zum Referenzieren angefahren werden soll. Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Achse bewegt sich in Richtung niedriger Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Konfiguration - Referenzieren (Seite 1906)

Referenznocken und Nullmarke über PROFIdrive-Telegramm (Seite 1906)

Referenzmarke über Digitaleingang (Seite 1908)

Technologieobjekt Externer Geber (Seite 1780)

Referenzmarke über Digitaleingang

Digitaleingang Referenzmarke

Wählen Sie in diesem Konfigurationsfeld einen Digitaleingang, der als Referenzmarke (Referenznocken) wirken soll. Wählen Sie zusätzlich den Pegel, bei welchem die Referenzmarke erkannt werden soll.

Referenzierrichtung

Wählen Sie in welcher Richtung die Referenzmarke zum Referenzieren angefahren werden soll.

Folgende Optionen stehen zur Verfügung:

- **Positiv**
  Achse bewegt sich in Richtung höherer Positionswerte.
- **Negativ**
  Achse bewegt sich in Richtung niedriger Positionswerte.
- **Aktuell**
  Zum Referenzieren wird die aktuell wirksame Fahrtrichtung verwendet.

Referenzmarke

Wählen Sie welche Schaltposition des "Digitaleingangs" als Referenzmarke verwendet werden soll.

Beim Überfahren eines "Digitaleingangs" werden zwei Schaltflanken erzeugt, die räumlich auseinander liegen. Mit der Wahl der positiven oder negativen Seite wird sichergestellt, dass die Referenzmarke immer an der gleichen mechanischen Position ausgewertet wird.

Die positive Seite ist die Schaltposition mit einem größeren Positionswert, die negative Seite ist die Schaltposition mit dem kleineren Positionswert.

Die Auswahl der Seite ist unabhängig von der Fahrtrichtung und unabhängig davon, ob sie eine steigende oder eine fallende Flanke bewirkt.

Referenzpunktposition

Konfigurieren Sie in diesem Feld die absolute Referenzpunktkoordinate der Referenzpunktposition. Die hier konfigurierte Referenzpunktposition wirkt, wenn die Motion Control-Anweisung "MC_Home" mit "Mode" = 10 ausgeführt wird.

Hinweis

Parameter "MC_Home.Mode"

Im Rahmen der Technologieversion V2.0 wurde der Parameter "MC_Home.Mode" für S7-1200 Motion Control und S7-1500 Motion Control vereinheitlicht. Dadurch ergibt sich auch eine neue Belegung der Parameterwerte für den Parameter "MC_Home.Mode". Eine Gegenüberstellung des Parameters "MC_Home.Mode" für die Technologieversionen V1.0 und V2.0 finden Sie im Kapitel "Versionsübersicht (Seite 1826)".

Siehe auch

Konfiguration - Referenzieren (Seite 1906)

Nullmarke über PROFIdrive-Telegramm (Seite 1907)

Referenznocken und Nullmarke über PROFIdrive-Telegramm (Seite 1906)

Technologieobjekt Externer Geber (Seite 1780)

Konfiguration - Istwertextrapolation

Konfigurieren Sie im Konfigurationsfenster "Istwertextrapolation" die Eigenschaften der Extrapolation bei einer Istwertkopplung für den Gleichlauf. Die hier eingestellten Werte gelten nur, wenn die Istwerte dieses Gebers als Leitwert verwendet werden.

Positionsfilter T1 und T2

Geben Sie die Zeitkonstanten für die Filterung der Istposition ein.

Geschwindigkeitsfilter T1 und Toleranzbandbreite

Geben Sie die Zeitkonstante für die Glättung der Geschwindigkeit und die Toleranzbandbreite ein.

Für eine optimierte Anwendung des Toleranzbandes geben Sie die gleiche Breite für das Toleranzband ein, wie die Breite des Rausschsignals ist.

Folgeachsbedingte Extrapolationszeit

Geben Sie den durch die Folgeachse bedingten Anteil für die Berechnung der Extrapolationszeit ein.

Hysteresewert

Geben Sie einen Wert für die Anwendung der Hysteresefunktion auf den extrapolierten Istwert ein. Die Angabe erfolgt in der konfigurierten Längeneinheit.

Siehe auch

Technologieobjekt Externer Geber (Seite 1780)

Extrapolation der Leitwerte bei Istwertkopplung (Seite 1767)

Technologieobjekt Messtaster konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen des Messtasters. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen des Messtasters können im Anwenderprogramm unter diesem Namen verwendet werden.

Zugeordnete Achse oder Externer Geber

Die dem Messtaster zugeordnete Achse oder der Externe Geber wird angezeigt. Über die Schaltfläche können Sie direkt die Konfiguration der Grundparameter des übergeordneten Technologieobjekts aufrufen.

Maßeinheit

Die angezeigte Maßeinheit zur Position des Messtasters entspricht der Maßeinheit des übergeordneten Technologieobjekts.

Siehe auch

Maßeinheiten (Seite 1689)

Technologieobjekt Messtaster (Seite 1781)

Konfiguration - Hardware-Schnittstelle

Messtastertyp

Wählen Sie den Messtastertyp aus:

- Messen über TM Timer DIDQ
- Messen über SINAMICS-Messtastereingang
- Messen über PROFIdrive-Telegramm (Antrieb oder Externer Geber)

Messen über TM Timer DIDQ

Wählen Sie bei einer Messung über einen Timer-DI einen Messeingang aus. Im Auswahlfeld werden alle Kanäle angezeigt, die korrekt konfiguriert sind.

Messen über SINAMICS-Messtastereingang

Wählen Sie bei einer Messung über SINAMICS-Messtastereingang einen Messeingang aus. Im Auswahlfeld werden alle passenden Telegrammtypen angezeigt. Ihnen werden alle Klemmen angezeigt, die Sie potentiell als Messtaster verwenden können. Beachten Sie bei der Zuordnung, dass die Eingänge den Messtastern (P680) in lückenloser und aufsteigender Reihenfolge im Telegramm zugeordnet werden müssen.

Über P728.8 bis P728.15 konfigurieren Sie an der Control Unit alle als Messeingang genutzten DI/DOs. Über P680 der Control Unit legen Sie die Klemmen für die globalen Messtastereingänge fest.

Messen über PROFIdrive-Telegramm (Antrieb oder Externer Geber)

Wählen Sie bei einer Messung über PROFIdrive-Telegramm in der Klappliste "Nummer des Messtasters" die Nummer des Messtasters im Telegramm aus. Das Eingabefeld ist mit dem Wert "1" voreingestellt.

Für die Messwertübertragung sind im PROFIdrive-Telegramm zwei Kommunikationskanäle verfügbar. Die Kommunikationskanäle werden im Antrieb jeweils einem Messeingang/

Digitaleingang zugeordnet. Konfigurieren Sie über PROFIdrive-Parameter, welcher Digitaleingang am Antrieb für den konfigurierten Kommunikationskanal verwendet werden soll.

- Messeingang für den ersten Kommunikationskanal
  ("<TO>.Parameter.PROFIdriveProbeNumber" = 1)
  Wenn Sie zwei Geber verwenden, müssen Sie für jeden Geber den zugehörigen DI im SINAMICS auswählen. Abhängig vom gewählten Geber werden dann verschiedene Ergebnisse an das Technologieobjekt übermittelt. Die Geber werden über die Parameter P488[0] und P488[1] parametriert.
- Messeingang für den zweiten Kommunikationskanal
  ("<TO>.Parameter.PROFIdriveProbeNumber" = 2)
  Wenn Sie zwei Geber verwenden, müssen Sie für jeden Geber den zugehörigen DI im SINAMICS auswählen. Abhängig vom gewählten Geber werden dann verschiedene Ergebnisse an das Technologieobjekt übermittelt. Die Geber werden über die Parameter P489[0] und P489[1] parametriert.

Korrekturzeit für das Messsignal

Geben Sie eine Korrekturzeit an, wenn mögliche Verzugszeiten im Messsignal ausgeglichen werden sollen.

Siehe auch

Automatische Übernahme der Antriebs- und Geberparameter im Gerät (Seite 1698)

Technologieobjekt Messtaster (Seite 1781)

Technologiemodule für Motion Control konfigurieren (Seite 1959)

Konfiguration - Erweiterte Parameter

Anpassung für Aktivierungszeit Messbereich

Um die systemseitig bestimmte Aktivierungszeit anzupassen, geben Sie hier eine zusätzliche Aktivierungszeit ein.

Im Konfigurationsfenster werden zusätzlich die systemseitig ermittelten Zeiten angezeigt:

- Zeit, nach dem Absetzen des Messaufrags bis das Messereignis erfasst werden kann
- Zeit, nach dem Messereignis bis das Messergebnis angezeigt wird (jeweils für das Messen von einer oder zwei Flanken)

Siehe auch

Messen mit Messbereich (Seite 1789)

Zeitliche Randbedingungen (Seite 1791)

Technologieobjekt Messtaster (Seite 1781)

Technologieobjekt Nocken konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen des Nockens. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen des Nockens können im Anwenderprogramm unter diesem Namen verwendet werden.

Zugeordnete Achse oder Externer Geber

Die dem Nocken zugeordnete Achse oder der Externe Geber wird angezeigt. Über die Schaltfläche können Sie direkt die Konfiguration der Grundparameter des übergeordneten Technologieobjekts aufrufen.

Nockentyp

Wählen Sie anhand des gewünschten Schaltverhaltens einen Nockentyp aus:

- Wegnocken (positionsabhängiges Ein- und Ausschalten)
- Zeitnocken (positionsabhängiges Einschalten und positionsunabhängiges bzw. zeitabhängiges Ausschalten)

Nockenbezug

Konfigurieren Sie in dieser Auswahl, ob sich die Schaltpunkte der Nocken auf die Istposition oder die Sollposition beziehen sollen.

Maßeinheit

Die angezeigte Maßeinheit zur Position des Nockens entspricht der Maßeinheit des übergeordneten Technologieobjekts.

Bei ausgewähltem Nockentyp Zeitnocken, wird zusätzlich die Maßeinheit für die Einschaltdauer und andere Zeiten angezeigt. Diese ist bei Nocken immer ms.

Siehe auch

Maßeinheiten (Seite 1689)

Technologieobjekt Nocken (Seite 1794)

Konfiguration - Hardware-Schnittstelle

Nockenausgabe

Wählen Sie aus, ob die erzeugten Schaltsignale am Digitalausgang ausgegeben werden sollen.

- **Ausgabe aktivieren**
  Wählen Sie für die Nockenausgabe eine der folgenden beiden Ausgabemöglichkeiten aus:
  - **Ausgabe über TM Timer DIDQ**
    Bei Ausgabe über einen TM Timer DIDQ wählen Sie im Feld "Ausgang" ein bereits konfiguriertes Technologiemodul und den zu verwendenden Kanal aus.
    Wenn kein Technologiemodul zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie ein Technologiemodul hinzu.
  - **Ausgabe über Digitalausgabemodul**
    Bei Ausgabe über ein Digitalausgabemodul, wählen Sie diesen im Feld "Nockenausgang" aus. Zur Auswahl werden nur die Digitalausgänge mit bereits definierten PLC-Variablen angezeigt.

  Wählen Sie die logische Verknüpfung des Nockensignals am Ausgang aus. Die Verknüpfung bezieht sich auf das letztlich auszugebende Signal nach der ggf. eingestellten Invertierung.
  Alle Nocken, die den ausgewählten Ausgang verwenden, werden grafisch dargestellt.
- **Ausgabe deaktiviert**
  Bei deaktivierter Ausgabe wird der Nocken nur in der Software ausgewertet.

Siehe auch

Technologieobjekt Nocken (Seite 1794)

Technologiemodule für Motion Control konfigurieren (Seite 1959)

Erweiterte Parameter

Konfiguration - Aktivierungszeit

Im oberen Bereich des Konfigurationsfensters "Aktivierungszeit" wird der eingestellte Nockentyp angezeigt.

Aktivierungszeit und Deaktivierungszeit

Für eine zeitliche Verschiebung des Einschalt- und Ausschaltzeitpunkts eines Nockens, geben Sie eine Aktivierungszeit und eine Deaktivierungszeit ein.

Siehe auch

Kompensation von Aktorschaltzeiten (Seite 1805)

Technologieobjekt Nocken (Seite 1794)

Konfiguration - Hysterese

Um ungewollte Veränderungen im Schaltzustand der Nocken einer Nockenspur zu verhindern, tragen Sie einen Hysteresewert ein.

Bei der Verwendung von Nocken mit Bezug auf die Istposition wird empfohlen, einen Hysteresewert (> 0.0) einzugeben.

Siehe auch

Hysterese (Seite 1803)

Technologieobjekt Nocken (Seite 1794)

Technologieobjekt Nockenspur konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts.

Name

Definieren Sie in diesem Feld den Namen der Nockenspur. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen der Nockenspur können im Anwenderprogramm unter diesem Namen verwendet werden.

Zugeordnete Achse oder Externer Geber

Die der Nockenspur zugeordnete Achse oder der Externe Geber wird angezeigt. Über die Schaltfläche können Sie direkt die Konfiguration der Grundparameter des übergeordneten Technologieobjekts aufrufen.

Nockentyp

Wählen Sie anhand des gewünschten Schaltverhaltens einen Nockentyp für die Nockenspur aus:

- Wegnocken (positionsabhängiges Ein- und Ausschalten)
- Zeitnocken (positionsabhängiges Einschalten und positionsunabhängiges bzw. zeitabhängiges Ausschalten)

Nockenbezug

Konfigurieren Sie in dieser Auswahl, ob sich die Schaltpunkte der Nockenspur auf die Istposition oder die Sollposition beziehen sollen.

Maßeinheit

Die angezeigte Maßeinheit zur Position der Nockenspur entspricht der Maßeinheit des übergeordneten Technologieobjekts.

Bei ausgewähltem Nockentyp Zeitnocken, wird zusätzlich die Maßeinheit für die Einschaltdauer und andere Zeiten angezeigt. Diese ist bei Nocken immer ms.

Siehe auch

Maßeinheiten (Seite 1689)

Technologieobjekt Nockenspur (Seite 1808)

Konfiguration - Hardware-Schnittstelle

Ausgabe Nockenspur

Wählen Sie aus, ob die erzeugten Schaltsignale am Digitalausgang ausgegeben werden sollen.

- **Ausgabe aktivieren**
  Wählen Sie für die Ausgabe der Nockenspur eine der folgenden beiden Ausgabemöglichkeiten aus:
  - Ausgabe über TM Timer DIDQ
    Bei Ausgabe über einen TM Timer DIDQ wählen Sie im Feld "Ausgang" ein bereits konfiguriertes Technologiemodul und den zu verwendenden Kanal aus.
    Wenn kein Technologiemodul zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie ein Technologiemodul hinzu.
  - Ausgabe über Digitalausgabemodul
    Bei Ausgabe über ein Digitalausgabemodul, wählen Sie diesen im Feld "Nockenausgang" aus. Zur Auswahl werden nur die Digitalausgänge mit bereits definierten PLC-Variablen angezeigt.
- **Ausgabe deaktivieren**
  Bei deaktivierter Ausgabe wird die Nockenspur nur in der Software ausgewertet.

Siehe auch

Technologieobjekt Nockenspur (Seite 1808)

Technologiemodule für Motion Control konfigurieren (Seite 1959)

Erweiterte Parameter

Spurdaten

Konfiguration - Aktivierungszeit

Der eingestellte Nockentyp wird angezeigt.

Aktivierungszeit und Deaktivierungszeit

Geben Sie die Aktivierungszeit und die Deaktivierungszeit ein.

Für eine zeitliche Verschiebung der Einschalt- und Ausschaltzeitpunkte der Nocken einer Nockenspur, geben Sie eine Aktivierungszeit und eine Deaktivierungszeit ein.

Siehe auch

Zeitlicher Versatz von Nockenschaltpunkten (Seite 1819)

Technologieobjekt Nockenspur (Seite 1808)

Konfiguration - Hysterese

Um ungewollte Veränderungen im Schaltzustand der Nocken einer Nockenspur zu verhindern, tragen Sie einen Hysteresewert ein.

Bei der Verwendung von Nocken mit Bezug auf die Istposition wird empfohlen, einen Hysteresewert (> 0.0) einzugeben.

Siehe auch

Hysterese (Seite 1819)

Technologieobjekt Nockenspur (Seite 1808)

Konfiguration - Spurmaße

Spurlänge

Geben Sie die entsprechende Spurlänge ein.

Berücksichtigen Sie bei der Bestimmung der Spurlänge ebenfalls die Nockendaten der einzelnen Nocken. Nocken, deren Anfangsposition außerhalb der Nockenspurlänge liegt, werden nicht berücksichtigt. Sie werden erst wirksam, wenn die Nockenspurlänge vergrößert wird, sodass mindestens die jeweilige Anfangsposition eines Nockens innerhalb der neuen Spurlänge liegt.

Achsbezugsposition

Geben Sie ein, ab welcher Position einer Achse oder eines Externen Gebers die Ausgabe der Nockenspur erfolgen soll. Der Anfang der Nockenspur wird auf die eingegebene Position gelegt.

Sie können einen negativen als auch positiven Wert für die Bezugsposition eingeben.

Modulolänge der Achse

Bei Anwendung einer Achse mit Modulofunktion wird die Modulolänge der Achse angezeigt.

Siehe auch

Technologieobjekt Nockenspur (Seite 1808)

Konfiguration - Nockendaten

Der eingestellte Nockentyp wird angezeigt.

Geben Sie die Eigenschaften für die auszugebenden Nocken der Nockenspur ein. Auf einer Nockenspur können Sie bis zu 32 einzelne Nocken einstellen.

Berücksichtigen Sie bei der Bestimmung der Nockendaten auch die gegebenenfalls bereits definierte Spurlänge. Nocken, deren Anfangsposition außerhalb der Nockenspurlänge liegt, werden nicht berücksichtigt. Sie werden erst wirksam, wenn die Nockenspurlänge vergrößert wird, sodass mindestens die jeweilige Anfangsposition eines Nockens innerhalb der neuen Spurlänge liegt.

Abhängig von dem eingestellten Nockentyp werden im Konfigurationsfenster Nockendaten die nachfolgend beschriebenen Eingabemöglichkeiten angezeigt.

- **Gültig**
  Nur als "Gültig" gesetzte Nocken werden ausgegeben und besitzen eine Statusanzeige.
- **Anfangsposition**
  - Die Anfangsposition darf bei Wegnocken nicht größer als die Endposition sein.
  - Wenn die Anfangsposition gleich der Endposition ist, schaltet der Wegnocken nicht.
  - Überlappungen der Schaltbereiche von einzelnen Nocken sind zulässig.
- **Endposition**
  - Die Spalte "Endposition" wird nur bei eingestelltem Nockentyp Wegnocken angezeigt.
  - Die Endposition darf nicht kleiner als die Anfangsposition sein.
- **Einschaltdauer**
  Die Spalte "Einschaltdauer" wird nur bei eingestelltem Nockentyp Zeitnocken angezeigt.

Siehe auch

Technologieobjekt Nockenspur (Seite 1808)

Technologieobjekt Kurvenscheibe konfigurieren

Das Technologieobjekt Kurvenscheibe (Seite 1821) konfigurieren Sie mit einem Editor.

Die Kurvenscheibe erstellen Sie über ein Diagramm, eine Tabelle mit den Elementen der Kurve sowie über die Eigenschaften der Elemente. Zwischen den einzelnen Elementen der Kurvenscheibe (z. B. Punkte, Geraden, Polynome) werden Übergänge berechnet. Der Kurvenverlauf gibt die wegbezogene Abhängigkeit zwischen Leitachse (Leitwerte, Abszisse im Diagramm) und Folgeachse (Folgewerte, Ordinate im Diagramm) wieder.

Folgendes Bild zeigt den Aufbau des Editors:

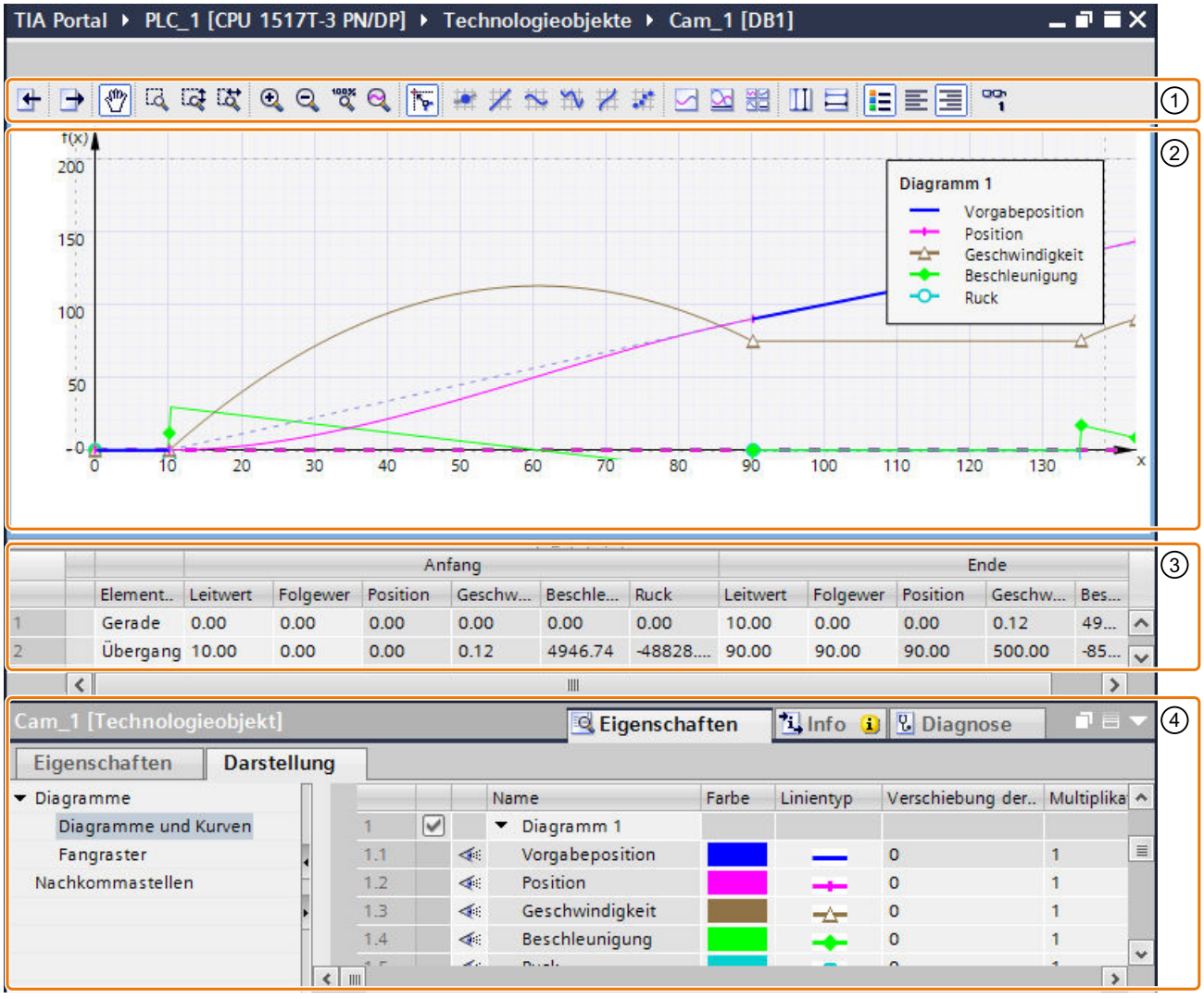

① Werkzeugleiste

② Grafischer Editor

Der Leitwertbereich (Definitionsbereich) wird auf der Abszisse (x-Achse) angezeigt.

Der Folgewertbereich (Wertebereich) wird auf der Ordinate (y-Achse) angezeigt.

③ Tabellarischer Editor

④ Eigenschaften (Inspektorfenster)

Werkzeugleiste

Über die Werkzeugleiste bedienen Sie den grafischen Editor und importieren/exportieren Kurvenscheiben.

Grafischer Editor

Im grafischen Editor bearbeiten Sie grafisch die Elemente der Kurve. Die Elemente lassen sich hinzufügen, bearbeiten und löschen. Bis zu vier Diagramme lassen sich übereinander mit synchronisierter Abszisse anzeigen. In den Diagrammen lassen sich die Vorgabekurve sowie die Kurven für die effektive Position, Geschwindigkeit, Beschleunigung und Ruck anzeigen.

Tabellarischer Editor

Im tabellarischen Editor werden alle Elemente der Kurve aufgelistet. Vorhandene Elemente lassen sich editieren. Neue Elemente lassen sich hinzufügen.

Eigenschaften (Inspektorfenster)

Im Inspektorfenster konfigurieren Sie in der Registerkarte "Eigenschaften" die Eigenschaften der Kurve und des selektierten Elements und in der Registerkarte "Darstellung" die grafische Ansicht:

- Profil (z. B. Leit- und Folgewertbereich, Optimierung und Interpolation des Profils, Anzahl verwendeter Elemente)
- Element (z. B. Ableitungen, Polynomkoeffizienten, Optimierung des Elements)
- Grafische Ansicht (z. B. Linienart, Linienfarbe, Skalierung der Ansicht)

Elemente der Kurve

Folgende Tabelle zeigt die Elemente, mit denen sich die Kurve definieren lässt:

Element	Beschreibung
Punkt	Ein Punkt ordnet einem Leitwert einen Folgewert zu. Die Kurve verläuft durch den Punkt mit diesen Koordinaten. Über die erste, zweite und dritte Ableitung können Sie die Geschwindigkeit, die Beschleunigung und den Ruck in diesem Punkt definieren.
Punktegruppe	Eine Punktegruppe fasst zwei oder mehr Punkte zu einem gemeinsam interpolierten Element zusammen und ermöglicht genaue Interpolationsvorgaben zwischen diesen Punkten.
Gerade	Eine Gerade beschreibt eine Bewegung mit konstanter Geschwindigkeit vom Anfangspunkt der Gerade bis zum Endpunkt. Die Steigung der Gerade gibt die konstante Geschwindigkeit vor.
Sinus	Ein Sinuselement beschreibt eine Bewegung gemäß der Sinusfunktion. Die Sinusfunktion lässt sich mit dem Phasenwinkel im Anfangs- und im Endpunkt, der Periodenlänge, der Amplitude sowie dem Schwingungsnullpunkt (Offset) anpassen.

Element	Beschreibung
Polynom	Ein Polynom beschreibt eine Bewegung gemäß einer Polynomfunktion maximal 7. Grades. Polynome lassen sich über die Eingabe der Randbedingungen oder der Polynomkoeffizienten definieren. Optional lässt sich ein trigonometrischer Polynomanteil konfigurieren.
Inverser Sinus (approximiert)	Ein inverser Sinus beschreibt eine Bewegung gemäß der Arkussinusfunktion. Ein inverser Sinus wird über Stützpunkte der Arkussinusfunktion angenähert.
Übergang	Übergänge interpolieren den Bereich zwischen zwei Elementen. Die Bereiche werden automatisch durch die Steuerung oder über eine konfigurierbare Optimierung gemäß VDI-Richtlinie 2143 interpoliert. Übergänge werden automatisch hinzugefügt.

Kurvenscheibeneditor bedienen

Das hier beschriebene Vorgehen zeigt die grundsätzliche Bedienung des Kurvenscheibeneditors. Dieses Vorgehen dient als Empfehlung.

Die grundsätzliche Bedienung teilt sich folgendermaßen auf:

- Voreinstellungen anpassen
- Kurve erstellen und anpassen
- Interpolation/Optimierung der Übergänge einstellen

Voreinstellungen anpassen

Um den Leit- und Folgewertbereich des Kurvenscheibenprofils sowie die grafische Ansicht anzupassen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in den Eigenschaften (Inspektorfenster) das Konfigurationsfenster "Profil > Allgemein (Seite 1935)".
2. Konfigurieren Sie den Leitwertbereich und den Folgewertbereich der Kurvendefinition. Die grafische Ansicht wird automatisch auf die Eingaben angepasst.
3. Öffnen Sie in der Bereichsnavigation des Inspektorfensters die Registerkarte "Darstellung (Seite 1953)".
4. Konfigurieren Sie in den Konfigurationsfenstern:
   - Die Anzeige der Diagramme und Kurven
   - Die Rasterabstände für die Ausrichtung von Eingaben im grafischen Editor
   - Die im Kurvenscheibeneditor angezeigten Nachkommastellen

Kurve erstellen und anpassen

Um die Kurve zu erstellen und anzupassen, gehen Sie folgendermaßen vor:

1. Fügen Sie über den grafischen Editor und/oder den tabellarischen Editor die Elemente der Kurvenscheibe ein:
   - Wählen Sie in der Werkzeugleiste das Werkzeug zum Einfügen des entsprechenden Elements aus. Platzieren Sie das Element an der gewünschten Position im grafischen Editor.
   - Fügen Sie im tabellarischen Editor über <Hinzufügen> in der Spalte "Elementtyp" die entsprechenden Elemente ein. Passen Sie Position der Elemente über die Anfangs- und Endwerte an.

   Übergänge zwischen den Elementen werden automatisch eingefügt.
2. Um ein Element zu bearbeiten, selektieren Sie das Element im grafischen oder tabellarischen Editor.
   Das Element wird im grafischen und im tabellarischen Editor hervorgehoben. In den Eigenschaften (Inspektorfenster) wird das Konfigurationsfenster "Element > Parameter/ Charakteristik" angezeigt.
3. Die Elemente lassen sich folgendermaßen anpassen:
   - Verschieben Sie das Element oder die Ziehpunkte des Elements im grafischen Editor.
   - Passen Sie die Anfangs- und Endwerte im tabellarischen Editor an.
   - Konfigurieren Sie weitere elementspezifische Parameter in den Eigenschaften (Inspektorfenster) im Konfigurationsfenster "Element > Parameter (Seite 1941)".
   - Stellen Sie die Interpolation der Übergänge über die Eigenschaften (Inspektorfenster) ein.

Die Anzahl der verbrauchten Elemente wird in den Eigenschaften (Inspektorfenster) im Eigenschaftsfenster "Profil > Statistik (Seite 1939)" angezeigt.

Interpolation der Übergänge einstellen (Systeminterpolation)

Die Interpolation (Seite 1823) lässt sich für jeden Übergang separat einstellen. Die Voreinstellung für die Interpolation der Übergänge ist die Systeminterpolation. Die Systeminterpolation konfigurieren Sie für alle Übergänge in den Eigenschaften (Inspektorfenster) im Konfigurationsfenster "Profil > Systeminterpolation (Seite 1936)".

Optimierung der Übergänge einstellen (VDI-Richtlinie 2143)

Jeder Übergang lässt sich auch separat nach VDI-Richtlinie 2143 anpassen. Dabei werden voreingestellt die Einstellungen in den Eigenschaften (Inspektorfenster) im Konfigurationsfenster "Profil > Optimierungsvoreinstellungen (Seite 1936)" berücksichtigt.

Um die Optimierung eines Übergangs nach VDI-Richtlinie 2143 anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Übergang im grafischen oder tabellarischen Editor.
2. Öffnen Sie in den Eigenschaften (Inspektorfenster) das Konfigurationsfenster "Element > Charakteristik (Seite 1949)".

3. Wählen Sie in der Klappliste "Optimierungsmethode" die Optimierungsmethode "VDI-basierte Optimierung".
4. Passen Sie gegebenenfalls die Voreinstellungen an.
   Die Auswahl der Parameter wird automatisch auf die gemäß VDI-Richtlinie 2143 anwendbaren Einstellungen eingeschränkt.

Die Optimierung der Übergänge nach VDI-Richtlinie verbraucht zusätzliche Punkte bzw. Segmente (Seite 1939) in der Kurvenscheibe.

Siehe auch

Konfiguration Diagramme - Diagramme und Kurven (Seite 1953)

Grafischer Editor

Aufbau des grafischen Editors

Der grafische Editor ist in folgende Bereiche unterteilt:

- Werkzeugleiste
- Kurvendiagramm

Werkzeugleiste

Die Werkzeugleiste am oberen Rand des grafischen Editors stellt Ihnen über Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	Kurvenscheibe aus Datei importieren	Siehe Kapitel "Kurvenscheibe importieren/exportieren (Seite 1955)"
	Kurvenscheibe in Datei exportieren	Siehe Kapitel "Kurvenscheibe importieren/exportieren (Seite 1955)"
	Elemente bearbeiten/Ansicht verschieben	• Selektieren und Verschieben einzelner Elemente und Element • Verschieben der Ansicht per Drag & Drop Um von einem beliebigen Werkzeug auf das Werkzeug "Elemente bearbeiten/Ansicht verschieben" umzuschalten, drücken Sie die Taste <Esc>.
	Zoomauswahl aktivieren	Zoom auf ausgewählten Bereich
	Vertikalen Zoom aktivieren	Vertikaler Zoom auf ausgewählten Bereich ohne horizontale Skalierung Alternativ: <Strg> + Ziehen mit gedrückter Maustaste auf Ordinate
	Horizontalen Zoom aktivieren	Horizontaler Zoom auf ausgewählten Bereich ohne vertikale Skalierung Alternativ: <Strg> + Ziehen mit gedrückter Maustaste auf Abszisse
	Zoom in	Vergrößern der Anzeige Alternativ: <Strg> + Mausrad aufwärts im Kurvendiagramm
	Zoom out	Verkleinern der Anzeige Alternativ: <Strg> + Mausrad abwärts im Kurvendiagramm
	Alles anzeigen	Anzeige des gesamten Definitions- und Wertebereichs

Schaltfläche	Funktion	Beschreibung
	Zoom auf Kurve	Zoom auf den Folgewertbereich der Kurve, die Sie in der Legende des Diagramms ausgewählt haben
	Fangraster aktivieren	Eingaben und Elementendpunkte werden am konfigurierbaren Fangraster und an anderen Elementendpunkten ausgerichtet.
	Punkt einfügen	Hinzufügen eines Punkts zum Diagramm
	Gerade einfügen	Hinzufügen einer Gerade zum Diagramm
	Sinus einfügen	Hinzufügen eines Sinuselements zum Diagramm
	Polynom einfügen	Hinzufügen eines Polynoms zum Diagramm
	Inversen Sinus einfügen	Hinzufügen eines inversen Sinus zum Diagramm
	Punktegruppe einfügen	Hinzufügen einer Punktegruppe zum Diagramm
	Ansicht: Ein Diagramm mit Positionen	Anzeige eines Diagramms mit folgenden Kurven der im Editor geöffneten Kurvenscheibe: • Vorgabekurve • Effektive Position
	Ansicht: Ein Diagramm mit allen Kurven	Anzeige eines Diagramms mit folgenden Kurven der im Editor geöffneten Kurvenscheibe: • Vorgabekurve • Effektive Position • Effektive Geschwindigkeit • Effektive Beschleunigung • Effektiver Ruck
	Ansicht: Vier Diagramme mit allen Kurven	Anzeige von vier Diagrammen mit folgenden Kurven der im Editor geöffneten Kurvenscheibe: • Diagramm mit Vorgabekurve und effektiver Position • Diagramm mit effektiver Geschwindigkeit • Diagramm mit effektiver Beschleunigung • Diagramm mit effektivem Ruck
	Vertikale Messlinien	Anzeigen und Verschieben der vertikalen Messlinien Ziehen Sie mit gedrückter Maustaste einen Messbereich auf. Die vertikale Position der Messlinien lässt sich verschieben. Im Diagramm werden die Funktionswerte an den Messlinienpositionen angezeigt. Zwischen den Messlinien wird die Differenz der Messlinien angezeigt.
	Horizontale Messlinien	Anzeigen und Verschieben der horizontalen Messlinien Ziehen Sie mit gedrückter Maustaste einen Messbereich auf. Die horizontale Position der Messlinien lässt sich verschieben. Im Diagramm werden die Funktionswerte an den Messlinienpositionen angezeigt. Zwischen den Messlinien wird die Differenz der Messlinien angezeigt.
	Legende anzeigen	Ein- bzw. Ausblenden der Legende im Kurvendiagramm. Um auf der Ordinate die Werte für eine bestimmte Kurve anzuzeigen, klicken Sie in der Legende auf den Namen der entsprechenden Kurve.

Schaltfläche	Funktion	Beschreibung
	Legende links anzeigen	Anzeige der Legende auf der linken Seite des Kurvendiagramms.
	Legende rechts anzeigen	Anzeige der Legende auf der rechten Seite des Kurvendiagramms.
	Online-Kurve einmalig auslesen und anzeigen	Anzeige der aus der CPU zurückgelesenen Positionswerte der Kurvenscheibe (orange) Der Kurvenscheibeneditor liest die Kurvenscheibe aus, die bereits in die CPU geladen wurde. Die ausgelesene "Online-Kurve" wird im grafischen Editor angezeigt.

Kurvendiagramm

Im Kurvendiagramm geben Sie die Elemente der Kurve grafisch ein und passen die Kurve über Auswahl und Verschiebung der Elemente an.

Diagrammflächen außerhalb des in "Profil > Allgemein (Seite 1935)" konfigurierten Leitwert-/Folgewertbereichs werden ausgegraut. Elemente außerhalb des Leitwert-/Folgewertbereichs werden mit Warnung („Element liegt außerhalb des Definitionsbereichs“) dargestellt.

Über die Konfiguration der grafischen Ansicht können Sie verschiedene Kurven (Position, Geschwindigkeit, Beschleunigung und Ruck) in bis zu vier Diagrammen übereinander anzeigen. Wenn mehrere Diagramme angezeigt werden, können Sie die Diagramme über die Trennlinien in der Höhe anpassen.

Die Ansicht zoomen Sie im Handmodus über <Strg> + Mausrad und <Strg> + Ziehen mit gedrückter Maustaste auf Abszisse/Ordinate.

Über Warndreiecke zeigt der Editor Meldungen zur Überprüfung der eingegebenen Kurve an. Der Tooltip des Warndreiecks zeigt den Meldungstext. Die Überprüfung der Kurve konfigurieren Sie im Konfigurationsfenster Überprüfung (Seite 1938).

Anzeige der Online-Kurve

Wenn Sie auf die Schaltfläche klicken, liest der Kurvenscheibeneditor die Daten aus dem Technologieobjekt-Datenbaustein auf der CPU aus und zeigt die Kurve im grafischen Editor an:

Status Kurvenscheibe	Status Interpolation	
Daten geändert (CamDataChanged = 0)	Nicht interpoliert (Interpolated = 0)	Nur die Punkte und Segmente der Kurvenscheibe werden angezeigt.
	Interpoliert (Interpolated = 1)	Die gesamte Kurvenscheibe wird angezeigt.
Daten geändert (CamDataChanged = 1)	Nicht interpoliert (Interpolated = 0)	Nur die Punkte und Segmente der Kurvenscheibe werden angezeigt.
	Interpoliert (Interpolated = 1)	Nur die Punkte und Segmente der Kurvenscheibe werden angezeigt und interpoliert.

Punkt einfügen

Ein Punkt ordnet einem Leitwert einen Folgewert zu. Die Kurve verläuft durch den Punkt mit diesen Koordinaten.

Über die erste, zweite und dritte Ableitung lassen sich die Geschwindigkeit, die Beschleunigung und der Ruck in diesem Punkt definieren. Die Ableitungen werden bei der VDI-basierten Optimierung von Übergängen des Punkts zu anderen Elementen berücksichtigt.

Punkt einfügen

Um einen Punkt zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Punkt einfügen" aus.
2. Klicken Sie im Diagramm 1 auf die Position, an der Sie den Punkt einfügen wollen. Der Punkt wird eingefügt. Am Punkt werden die Koordinaten angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch eingefügt.

Punkt verschieben

Um einen Punkt im grafischen Editor zu verschieben, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie den Punkt im Diagramm 1.
3. Verschieben Sie den Punkt per Drag & Drop auf die gewünschte Position.

Parameter anpassen

Die Parameter des Punkts lassen sich im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1941)" anpassen.

Punktegruppe einfügen

Eine Punktegruppe fasst zwei oder mehr Punkte zu einem gemeinsam interpolierten Element zusammen und ermöglicht genaue Interpolationsvorgaben zwischen den Punkten.

Punktegruppe einfügen

Um eine Punktegruppe zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Punktegruppe einfügen" aus.
2. Klicken Sie im Diagramm 1 auf die Position, an der Sie die Punktegruppe einfügen wollen. Die Punktegruppe wird eingefügt. An der Punktegruppe werden die Koordinaten des Anfangspunkts und des Endpunkts angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Wenn bereits ein anderes Element vorhanden ist, wird automatisch ein Übergang zu dem vorhandenen Element eingefügt.

Punktegruppe anpassen

Um eine Punktegruppe im grafischen Editor anzupassen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie die Punktegruppe im Diagramm 1.
   Die Punktegruppe wird grafisch mit Ziehpunkten hervorgehoben. Folgende Ziehpunkte werden angezeigt:
   – Anfangswert der Punktegruppe
   – Endwert der Punktegruppe
3. Verschieben Sie per Drag & Drop die Ziehpunkte oder die gesamte Punktegruppe auf die gewünschte Position.
   Wenn zwischen Anfangs- und Endpunkt weitere Stützpunkte in der Punktegruppe konfiguriert sind, behandelt der Kurvenscheibeneditor die Stützpunkte folgendermaßen:
   – Definitionsart der Leitwerte "Relativ zum Segment"
     Die Stützpunkte werden relativ zum Anfangs- und Endpunkt mitverschoben.
   – Definitionsart der Leitwerte "Absolut im Profil"
     Die Stützpunkte werden nicht verschoben.

Parameter anpassen

Die Parameter der Punktegruppe können Sie im grafischen Editor, im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1942)" anpassen.

Gerade einfügen

Eine Gerade beschreibt eine Bewegung mit konstanter Geschwindigkeit vom Anfangspunkt der Gerade bis zum Endpunkt. Die Steigung der Gerade gibt die konstante Geschwindigkeit vor.

Gerade einfügen

Um eine Gerade zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Gerade einfügen" aus.
2. Ziehen Sie per Drag & Drop im Diagramm 1 die Gerade von der Anfangsposition bis zur Endposition.
   Die Gerade wird eingefügt. An der Gerade werden die Koordinaten des Anfangspunkts und des Endpunkts angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch eingefügt.

Gerade verschieben

Um eine Gerade im grafischen Editor zu verschieben, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie die Gerade im Diagramm 1.
   Die Gerade wird grafisch mit Ziehpunkten hervorgehoben. Folgende Ziehpunkte werden angezeigt:
   – Anfangspunkt der Gerade
   – Endpunkt der Gerade
3. Verschieben Sie per Drag & Drop die Ziehpunkte oder die gesamte Gerade auf die gewünschte Position.

Parameter anpassen

Die Parameter der Gerade lassen sich im grafischen Editor, im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1944)" anpassen.

Sinus einfügen

Ein Sinuselement beschreibt eine Bewegung gemäß der Sinusfunktion. Die Sinusfunktion lässt sich mit dem Phasenwinkel im Anfangs- und im Endpunkt, der Periodenlänge, der Amplitude sowie dem Schwingungsnullpunkt (Offset) anpassen.

Sinus einfügen

Um einen Sinus zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Sinus einfügen" aus.
2. Klicken Sie im Diagramm 1 auf die Position, an der Sie den Sinus einfügen wollen. Der Mauszeiger zeigt dabei auf die Anfangsposition des Sinus.
   Der Sinus wird eingefügt. Am Sinus werden die Koordinaten des Anfangspunkts und des Endpunkts angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch eingefügt.

Sinus anpassen

Um einen Sinus im grafischen Editor anzupassen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie den Sinus im Diagramm 1.
   Der Sinus wird grafisch mit Ziehpunkten sowie Hilfslinien für die Null-Linie und die Amplitude hervorgehoben. Folgende Ziehpunkte werden angezeigt:
   - Leitwert/Verschiebung am linken/rechten Rand
     Bei einem geneigten Sinus lässt sich über diese Ziehpunkte auch die Neigung anpassen.
   - Leitwert am linken/rechten Rand
   - Phase am linken/rechten Rand
   - Amplitude
3. Verschieben Sie per Drag & Drop die Ziehpunkte oder den gesamten Sinus auf die gewünschte Position.

Parameter anpassen

Die Parameter des Sinus lassen sich im grafischen Editor, im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1945)" anpassen.

Polynom einfügen

Ein Polynom beschreibt eine Bewegung gemäß einer Polynomfunktion maximal 6. Grades. Polynome lassen sich über die Eingabe der Randbedingungen oder der Polynomkoeffizienten definieren. Optional lässt sich ein trigonometrischer Polynomanteil konfigurieren.

Polynom einfügen

Um ein Polynom zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Polynom einfügen" aus.
2. Klicken Sie im Diagramm 1 auf die Position, an der Sie das Polynom einfügen wollen. Der Mauszeiger zeigt dabei auf die Anfangsposition des Polynoms.
   Das Polynom wird eingefügt. Am Polynom werden die Koordinaten des Anfangspunkts und des Endpunkts angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Wenn bereits ein anderes Element vorhanden ist, wird automatisch ein Übergang zu dem vorhandenen Element eingefügt.

Polynom anpassen

Um ein Polynom im grafischen Editor anzupassen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie das Polynom im Diagramm 1.
   Das Polynom wird grafisch mit Ziehpunkten hervorgehoben. Folgende Ziehpunkte werden angezeigt:
   – Leitwert/Folgewert am linken/rechten Rand
   – Lage des Wendepunkts
3. Verschieben Sie per Drag & Drop die Ziehpunkte oder den gesamten Sinus auf die gewünschte Position.

Parameter anpassen

Die Parameter des Polynoms können Sie im grafischen Editor, im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1946)" anpassen.

Inversen Sinus einfügen

Ein inverser Sinus beschreibt eine Bewegung gemäß der Arkussinusfunktion. Die Arkussinusfunktion ist die Umkehrfunktion der Sinusfunktion. Ein inverser Sinus wird über Stützpunkte der Arkussinusfunktion angenähert.

Inversen Sinus einfügen

Um einen inversen Sinus zum Kurvenverlauf hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Inversen Sinus einfügen" aus.
2. Klicken Sie im Diagramm 1 auf die Position, an der Sie den inversen Sinus einfügen wollen.
   Der Mauszeiger zeigt dabei auf die Anfangsposition des inversen Sinus.
   Der inverse Sinus wird eingefügt. Am Punkt werden die Koordinaten angezeigt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch eingefügt.

Inversen Sinus anpassen

Um einen inversen Sinus im grafischen Editor anzupassen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Werkzeugleiste das Werkzeug "Elemente bearbeiten/Ansicht verschieben" aus.
2. Selektieren Sie den inversen Sinus im Diagramm 1.
   Der inverse Sinus wird grafisch mit Ziehpunkten hervorgehoben. Folgende Ziehpunkte werden angezeigt:
   – Anfangspunkt des inversen Sinus
   – Endpunkt des inversen Sinus
3. Verschieben Sie per Drag & Drop die Ziehpunkte oder den gesamten inversen Sinus auf die gewünschte Position.

Parameter anpassen

Die Parameter des inversen Sinus lassen sich im grafischen Editor, im tabellarischen Editor sowie in den Eigenschaften (Inspektorfenster) unter "Element > Parameter (Seite 1948)" anpassen.

Element löschen

Um ein Element im grafischen Editor zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Element.
2. Drücken Sie die Taste <Entf>.
   Das Element wird gelöscht. Der grafische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird ebenfalls gelöscht.

Kontextmenü im grafischen Editor

Folgende Tabelle zeigt die Funktionen im Kontextmenü des grafischen Editors:

Funktion	Beschreibung
Alles anzeigen	Anzeige des gesamten Definitions- und Wertebereichs
Zoom auf Kurve	Anzeige der in der Legende des Diagramms ausgewählten Kurve
Zoom in	Vergrößern der Anzeige
Zoom out	Verkleinern der Anzeige
Ausschneiden	Entfernen der ausgewählten Elemente und Kopieren in die Zwischenablage
Kopieren	Kopieren der ausgewählten Elemente in die Zwischenablage
Einfügen	Einfügen der Elemente aus der Zwischenablage nach dem letzten Element
Löschen	Löschen der ausgewählten Elemente Übergänge zu gegebenenfalls vorhandenen Elementen werden ebenfalls gelöscht.
Einfügen spezial	Aufruf des Dialogs "Elemente einfügen (Seite 1958)"

Funktion	Beschreibung
Punkte gruppieren	Zusammenfassen der selektierten Punkte zu einer Punktegruppe Der Eintrag wird unter folgenden Bedingungen angezeigt: • Im grafischen/tabellarischen Editor sind nur Punkte selektiert. • Zwischen den selektierten Punkten liegen keine weiteren Elemente.
Gruppierung der Punkte auflösen	Auflösen der selektierten Punktegruppe in einzelne Punkte
Kennzeichnung der Messpunkte anzeigen/verbergen	Ein- bzw. Ausblenden der Messpunkte Der Eintrag wird unter folgenden Bedingungen angezeigt: • Messlinien werden angezeigt. • Messpunkte sind verborgen/angezeigt.
Verschieben	Aufruf des Dialogs "Elemente verschieben (Seite 1958)"
Skalieren	Aufruf des Dialogs "Elemente skalieren (Seite 1958)"

Siehe auch

Dialoge im Kontextmenü (Seite 1958)

Tabellarischer Editor

Aufbau des tabellarischen Editors

Im tabellarischen Editor werden alle Elemente der Kurve nach ihren Leitwerten sortiert angezeigt. Die Elemente lassen sich anpassen. Neue Elemente lassen sich hinzufügen.

Zu jedem Element der Kurve werden in der entsprechenden Spalte folgende Eigenschaften angezeigt:

Spalte/Eigenschaft	Beschreibung
Erste Spalte	Laufende Nummer des Elements
Zweite Spalte	Anzeige von eventuellen Berechnungsproblemen mit Warndreieck Der Meldungstext wird über den Tooltip des Warndreiecks angezeigt.
Elementtyp	• Anzeige/Ändern des Elementtyps • Hinzufügen von Elementen Mögliche Elementtypen: • Punkt • Punktegruppe • Gerade • Sinus • Polynom • Inverser Sinus • Übergang
Anfang	Parameterwerte am Anfangspunkt des Elements

Spalte/Eigenschaft	Beschreibung
Leitwert	Leitwertvorgabe am Anfangspunkt des Elements
Folgewert	Folgewertvorgabe am Anfangspunkt des Elements
Position[1)]	Berechnete effektive Position am Anfangspunkt des Elements
Geschwindigkeit[1)]	Berechnete effektive Geschwindigkeit am Anfangspunkt des Elements
Beschleunigung[1)]	Berechnete effektive Beschleunigung am Anfangspunkt des Elements
Ruck[1)]	Berechneter effektiver Ruck am Anfangspunkt des Elements
Ende	Parameterwerte am Endpunkt des Elements
Leitwert	Leitwertvorgabe am Endpunkt des Elements
Folgewert	Folgewertvorgabe am Endpunkt des Elements
Position[1)]	Berechnete effektive Position am Endpunkt des Elements
Geschwindigkeit[1)]	Berechnete effektive Geschwindigkeit am Endpunkt des Elements
Beschleunigung[1)]	Berechnete effektive Beschleunigung am Endpunkt des Elements
Ruck[1)]	Berechneter effektiver Ruck am Endpunkt des Elements
Kommentar	Optionaler Kommentar zum Element

1) Wird entsprechend der Konfiguration in "Eigenschaften (Inspektorfenster) > Grafische Ansicht > Diagramme und Kurven" angezeigt.

Kurve editieren

Der tabellarische Editor bietet Ihnen folgende Möglichkeiten zum Editieren der Kurve:

- Einfügen von Elementen
- Löschen von Elementen
- Elementtyp ändern
- Leitwert und Folgewert der Randpunkte anpassen

Element einfügen

Um ein Element im tabellarischen Editor hinzuzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Spalte "Elementtyp" aus der Klappliste "Hinzufügen" den gewünschten Elementtyp. "Hinzufügen" wird immer in der Zeile nach dem zuletzt hinzugefügten Element angezeigt.
   Das Element wird hinter dem letzten Element mit geeigneten Werten eingefügt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch eingefügt.
   Die Parameter des Elements lassen sich im grafischen Editor, im tabellarischen Editor und in den Eigenschaften (Inspektorfenster) anpassen.

Element löschen

Um ein Element im tabellarischen Editor zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile des Elements.
2. Drücken Sie die Taste <Entf>.
   Das Element wird gelöscht. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird ebenfalls gelöscht.

Elementtyp umwandeln

Um den Elementtyp eines Elements im tabellarischen Editor zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile des Elements.
2. Wählen Sie in der Spalte "Elementtyp" aus der Klappliste den gewünschten Elementtyp.
   Der Elementtyp des Elements wird in den gewählten Elementtyp umgewandelt. Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch angepasst.

Leitwert und Folgewert der Randpunkte anpassen

Um den Leitwert/Folgewert der Randpunkte eines Elements im tabellarischen Editor anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Eingabefeld des zu ändernden Parameters.
2. Geben Sie den gewünschten Wert ein.
   Der tabellarische Editor und die Ansicht in den Eigenschaften (Inspektorfenster) werden aktualisiert. Ein Übergang zu einem gegebenenfalls vorhandenen Element wird automatisch angepasst.

Kontextmenü im tabellarischen Editor

Folgende Tabelle zeigt die Funktionen im Kontextmenü des tabellarischen Editors:

Funktion	Beschreibung
Zeile davor einfügen	Einfügen einer Tabellenzeile/eines Elements vor der selektierten Zeile/dem selektierten Element Wenn vor dem Element kein Übergang vorhanden ist, werden das selektierte Element und angrenzende Elemente verändert.
Zeile danach einfügen	Einfügen einer Tabellenzeile/eines Elements nach der selektierten Zeile/dem selektierten Element Wenn nach dem Element kein Übergang vorhanden ist, werden das selektierte Element und angrenzende Elemente verändert.
Ausschneiden	Entfernen der ausgewählten Elemente und Kopieren in die Zwischenablage
Kopieren	Kopieren der ausgewählten Elemente in die Zwischenablage
Einfügen	Einfügen der Elemente aus der Zwischenablage nach dem letzten Element

Funktion	Beschreibung
Löschen	Löschen der ausgewählten Elemente Übergänge zu gegebenenfalls vorhandenen Elementen werden ebenfalls gelöscht.
Einfügen spezial	Aufruf des Dialogs "Elemente einfügen (Seite 1958)"
Punkte gruppieren	Zusammenfassen der selektierten Punkte zu einer Punktegruppe Der Eintrag wird unter folgenden Bedingungen angezeigt: • Im grafischen/tabellarischen Editor sind nur Punkte selektiert. • Zwischen den selektierten Punkten liegen keine weiteren Elemente.
Gruppierung der Punkte auflösen	Auflösen der selektierten Punktegruppe in einzelne Punkte
Verschieben	Aufruf des Dialogs "Elemente verschieben (Seite 1958)"
Skalieren	Aufruf des Dialogs "Elemente skalieren (Seite 1958)"

Eigenschaften (Inspektorfenster)

Kontextsensitive Anzeige

In den Eigenschaften (Inspektorfenster) werden die Parameter für das Profil der Kurvenscheibe sowie für die Elemente angezeigt. Je nach selektiertem Element werden die entsprechenden Parameter angezeigt. Wenn kein Element der Kurve selektiert ist, werden nur die Einstellungen für das Profil der Kurvenscheibe angezeigt. Wenn ein Element der Kurve selektiert ist, werden zusätzlich die Parameter des Elements angezeigt.

Konfiguration Profil - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" den Leitwert- und den Folgewertbereich (Definitions- und Wertebereich) der Kurvenscheibe.

Die Eingaben des Leit- und Folgewertbereichs wirken sich nur auf die Anzeige im grafischen Editor aus. Die Kurvenscheibe wird im Definitionsbereich zwischen folgenden Werten interpoliert:

- Erster definierter Stützpunkt/Beginn des ersten Segments der Kurvenscheibe (<TO>.StatusCam.StartLeadingValue)
- Letzter definierter Stützpunkt/Ende des letzten Segments der Kurvenscheibe (<TO>.StatusCam.EndLeadingValue)

Leitwertbereich der Kurvendefinition

Konfigurieren Sie in diesem Bereich den Leitwertbereich (Definitionsbereich) der Kurvenscheibe:

Parameter	Beschreibung
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt des Leitwertbereichs der Kurvenscheibe.
Ende	Konfigurieren Sie in diesem Feld den Endpunkt des Leitwertbereichs der Kurvenscheibe.

Folgewertbereich der Kurvendefinition

Konfigurieren Sie in diesem Bereich die Beschränkung des Folgewertbereichs (Wertebereich) der Kurvenscheibe:

Parameter	Beschreibung
Minimum	Konfigurieren Sie in diesem Feld den kleinsten zulässigen Wert für den Folgewertbereich der Kurvenscheibe.
Maximum	Konfigurieren Sie in diesem Feld den größten zulässigen Wert für den Folgewertbereich der Kurvenscheibe.

Konfiguration Profil - Optimierungsvoreinstellungen

Im Konfigurationsfenster "Optimierungsvoreinstellungen" konfigurieren Sie die Voreinstellungen zur Optimierung von Übergängen gemäß VDI-Richtlinie 2143. Die Voreinstellungen werden verwendet, wenn Sie für einen Übergang (Seite 1949) die Optimierungsmethode "VDI-basierte Optimierung" verwenden und die Einstellung "Optimierungsvoreinstellung" für die Stetigkeit oder das Optimierungsziel wählen.

Die Kurvenscheibe wird mit der Motion Control-Anweisung "MC_InterpolateCam" gemäß den Einstellungen für die VDI-Optimierung interpoliert.

Voreinstellungen für die VDI-Optimierung

Konfigurieren Sie in diesem Bereich die Voreinstellungen für die Stetigkeitsforderung und das Optimierungsziel:

Parameter	Beschreibung
Stetigkeit	Wählen Sie in der Klappliste, welcher Parameter in den Randpunkten stetig ist und zur Optimierung berücksichtigt werden soll: • Position • Geschwindigkeit (stoßfrei) • Beschleunigung (ruckfrei) • Ruck
Optimierungsziel	Wählen Sie in der Klappliste das Optimierungsziel gemäß VDI-Richtlinie: • Nicht vorgegeben • Geschwindigkeit (Cv) • Beschleunigung (Ca) • Ruck (Cj) • Minimales Dynamisches Moment (Cmdyn)

Konfiguration Profil - Systeminterpolation

Im Konfigurationsfenster "Systeminterpolation" konfigurieren Sie die Interpolation von Übergängen nach Systemvorgaben. Diese Einstellungen werden verwendet, wenn Sie für einen Übergang (Seite 1949) die Optimierungsmethode "Systeminterpolation" wählen (Voreinstellung).

Die Kurvenscheibe wird mit der Motion Control-Anweisung "MC_InterpolateCam" interpoliert.

Einstellungen Systeminterpolation

Konfigurieren Sie in diesem Bereich die Interpolationsart und das Verhalten der Randpunkte:

Parameter	Beschreibung
Interpolationsart	Wählen Sie in der Klappliste, nach welcher Interpolationsart die Kurve interpoliert wird: • Lineare Interpolation • Interpolation mit kubischen Splines • Interpolation mit Bézier-Splines
Verhalten am Rand	Wählen Sie in der Klappliste, welches Verhalten der Randpunkte für die Interpolation gilt: • Keine Einschränkungen • Erste Ableitung stetig (geschwindigkeitsstetig) Die Kurvenscheibe wird so interpoliert, dass die erste Ableitung (Geschwindigkeit) am Anfang und am Ende der Kurvenscheibe gleich ist.

Konfiguration Profil - Effektive Runtime-Kurven

Konfigurieren Sie im Konfigurationsfenster "Effektive Runtime-Kurven" die Werte für Leit- und Folgeachse, die für die effektive Kurve angenommen werden. Die Runtime-Emulation berechnet die effektive Kurve mit diesen Annahmen und zeigt sie im grafischen Editor mit den angenommenen Grenzwerten an.

Die Eingaben werden nicht in die CPU geladen. Die Kurvenscheibe wird somit ohne diese Eingaben interpoliert. Mit den angenommenen Werten lässt sich testen und visualisieren, wie sich Kurvenscheibe im Betrieb verhält, z. B. bei Angabe einer Skalierung am "MC_CamIn".

Einstellungen der Leitachse

Konfigurieren Sie in diesem Bereich leitwertseitig die Berechnung und Anzeige der Kurve:

Parameter	Beschreibung
Von Achse kopieren	Wählen Sie über die Schaltfläche und den Dialog "Leitwerteinstellungen von Achse kopieren" eine Achse aus, deren maximale Geschwindigkeit als Geschwindigkeit für die Leitachse angenommen wird.
Skalierungsfaktor	Konfigurieren Sie in diesem Feld einen leitwertseitigen Skalierungsfaktor. Dies erlaubt die Annahme, dass an einem "MC_CamIn"-Auftrag eine Skalierung angegeben wird.
Maßeinheit	Wählen Sie in der Auswahlliste die Maßeinheit für den Leitwert.
Maßeinheit der ersten Ableitung	Wählen Sie in der Auswahlliste die Maßeinheit für die erste Ableitung des Leitwerts.
Geschwindigkeit	Konfigurieren Sie in diesem Feld die Geschwindigkeit der Leitachse, die für die Runtime-Emulation der Kurve angenommen wird.

Einstellungen der Folgeachse

Konfigurieren Sie in diesem Bereich folgewertseitig die Berechnung und Anzeige der Kurve:

Parameter	Beschreibung
Von Achse kopieren	Wählen Sie über die Schaltfläche und den Dialog "Folgewerteinstellungen von Achse kopieren" eine Achse aus, deren maximale Dynamikwerte als zu überprüfende Grenzwerte bei der Berechnung und Anzeige der Kurve übernommen werden.
Skalierungsfaktor	Konfigurieren Sie in diesem Feld einen folgewertseitigen Skalierungsfaktor. Dies erlaubt die Annahme, dass an einem "MC_CamIn"-Auftrag eine Skalierung angegeben wird.
Maßeinheit	Wählen Sie in der Auswahlliste die Maßeinheit für den Folgewert.
Maßeinheit der ersten Ableitung	Wählen Sie in der Auswahlliste die Maßeinheit für die erste Ableitung des Folgewerts.
Maximale Geschwindigkeit	Konfigurieren Sie in diesem Feld die maximale Geschwindigkeit für die Folgeachse.
Maximale Beschleunigung	Konfigurieren Sie in diesem Feld die maximale Beschleunigung für die Folgeachse.
Maximaler Ruck	Konfigurieren Sie in diesem Feld den maximalen Ruck für die Folgeachse.

Konfiguration - Überprüfung

Im Konfigurationsfenster "Überprüfung" konfigurieren Sie, welche Kriterien der Kurvenscheibeneditor bei der Eingabe der Kurve prüft. Wenn Sie eine Überprüfung aktivieren, zeigen der grafische und der tabellarische Editor entsprechende Meldungen über ein Warndreieck am Element an. Über den Tooltip am Warndreieck blenden Sie den Meldungstext ein.

Überprüfung von Grenzüberschreitungen

Konfigurieren Sie in diesem Bereich die Überprüfungen auf Einhaltung der konfigurierten Grenzen:

Überprüfung/Element	Beschreibung
Einhaltung der Kurvendefiniton des Leit- und Folgewertbereichs überprüfen	Aktivieren Sie das Optionskästchen "Einhaltung der Kurvendefiniton des Leit- und Folgewertbereichs überprüfen", damit der Kurvenscheibeneditor die Kurve entsprechend prüft.
Einhaltung der Maximalwerte der Ableitungen der effektiven Runtime-Kurve überprüfen	Aktivieren Sie das Optionskästchen "Einhaltung der Maximalwerte der Ableitungen der effektiven Runtime-Kurve überprüfen", damit der Kurvenscheibeneditor die Kurve entsprechend prüft.

Überprüfung der VDI-Eignung

Aktivieren Sie das Optionskästchen "Eignung von Übergängen nach VDI überprüfen", damit der Kurvenscheibeneditor die VDI-Eignung der Kurve prüft.

Dabei prüft der Kurvenscheibeneditor Folgendes:

- Unterstützung der Übergangsklassifikation des aktuell ausgewählten VDI-Übergangs
- Randwertanpassung nach VDI

Überprüfung der Stetigkeit

Wählen Sie in der Klappliste "Geforderte Stetigkeit", welchen Parameter der Kurvenscheibeneditor auf Stetigkeit prüft:

- Position
- Geschwindigkeit
- Beschleunigung
- Ruck

Wenn eine Funktion bzw. eine Ableitung unstetig ist, sind alle höheren Ableitungen ebenfalls unstetig.

Profil - Statistik

Das Eigenschaftsfenster "Statistik" zeigt eine Übersicht über die Anzahl der Elemente der Kurvenscheibe sowie die Minimal- und Maximalwerte der effektiven Kurven für den Folgewert und die Ableitungen. Eine Kurvenscheibe besteht aus maximal 1000 Punkten und maximal 50 Segmenten.

Verbrauchte Elemente

Dieser Bereich zeigt die Anzahl der verbrauchten Elemente der Kurve:

Parameter	Beschreibung
Punkte	Dieses Feld zeigt die Anzahl der verbrauchten Punkte der Kurvenscheibe. Eine Kurvenscheibe besteht aus maximal 1000 Punkten.
Segmente	Dieses Feld zeigt die Anzahl der verbrauchten Segmente der Kurvenscheibe. Eine Kurvenscheibe besteht aus maximal 50 Segmenten.

Der Verbrauch von Punkten und Segmenten ist abhängig von der Zusammenstellung und der Konfiguration der Elemente. Folgende Tabelle zeigt den Verbrauch von Punkten und Segmenten je Element:

Element	Anzahl verbrauchter Punkte	Anzahl verbrauchter Segmente
Punkt	1	0
Punkt an einem Übergang mit VDI-basierter Optimierung	0	0
Punktegruppe mit Abbildungsmethode Punktapproximation	Anzahl der konfigurierten Stützpunkte ("Eigenschaften (Inspektorfenster) > Element > Parameter > Approximation > Anzahl der Stützpunkte") Voreinstellung: 32	0

Element	Anzahl verbrauchter Punkte	Anzahl verbrauchter Segmente
Punktegruppe mit Abbildungsmethode Segmentapproximation	0	Anzahl der konfigurierten Stützpunkte - 1
Gerade	0	1
Sinus	0	1
Polynom	0	1
Inverser Sinus	Anzahl der konfigurierten Stützpunkte ("Eigenschaften (Inspektorfenster) > Element > Parameter > Approximation > Anzahl der Stützpunkte") Voreinstellung: 32	0
Inverser Sinus rechts von einem Übergang mit VDI-basierter Optimierung	Anzahl der konfigurierten Stützpunkte - 1	0
Übergang mit Systeminterpolation	0	0
Übergang mit VDI-basierter Optimierung		
Bewegungsgesetz		
Sinus	0	1
Sinus mit relativem Lambda ≠ 0,5	0	2
Geneigter Sinus	0	1
Geneigter Sinus mit relativem Lambda ≠ 0,5	0	2
Polynom	0	1
Polynom mit relativem Lambda ≠ 0,5	0	2
Modifiziertes Beschleunigungstrapez		
Bewegungsaufgabe		
Rast-in-Umkehr	0	5
Umkehr-in-Rast	0	5
Rast-in-Rast	0	6
Modifizierter Sinus		
Bewegungsaufgabe		
Rast-in-Rast	0	3
Konstante Geschwindigkeit-in-konstante Geschwindigkeit	0	4
Konstante Geschwindigkeit-in-Rast	0	4
Rast-in-konstante Geschwindigkeit	0	4
Sinus-Geraden-Kombination	0	3
Harmonische Kombination	0	3
Doppelt-harmonischer Übergang	Anzahl der konfigurierten Stützpunkte ("Eigenschaften (Inspektorfenster) > Element > Parameter > Approximation > Anzahl der Stützpunkte") Voreinstellung: 32	0
Quadratische Parabel	0	2

Lambda = Wendepunkt der Kurve

Wertebereiche

Dieser Bereich zeigt die Minimal- und Maximalwerte der effektiven Kurven für den Folgewert und die Ableitungen.

Randbedingungen

Für die Eingabe und Verwendung von Punkten und Segmenten gelten folgende Randbedingungen:

- **Punkte**
  Bei Punkten mit gleichen Leitwerten wird der Punkt wirksam, den Sie als letztes eingegeben haben bzw. der im tabellarischen Editor weiter unten steht.
- **Segmente**
  - Lücken zwischen Segmenten werden mit einem Übergangssegment gefüllt.
  - Bei Lücken im Leitwertbereich kleiner 1.0E-4 werden Segmentendpunkt und Segmentanfangspunkt zusammengezogen.
  - Bei Lücken im Leitwertbereich größer 1.0E-4 wird ein neues Übergangssegment eingefügt.
  - Bei Überlappungen wird das neue Segment ab dem Anfangspunkt eingefügt und vollständig verwendet. Wenn das vorherige Segment über das neue Segment hinaus definiert ist, wird nach dem Endpunkt des neuen Segments das vorherige Segment weiter verwendet.
- **Stützpunkte und Segmente** (gemischte Kurvenscheiben)
  Wenn Punkte im gleichen Bereich wie Segmente definiert sind, wird das Segment verwendet.

Konfiguration Elemente - Parameter

Konfigurieren Sie im Konfigurationsfenster "Parameter/Charakteristik" die Parameter des selektierten Elements der Kurve. Die Eingaben werden im tabellarischen und im grafischen Editor übernommen. Je nach selektiertem Element werden die elementspezifischen Parameter angezeigt:

- Punkt (Seite 1941)
- Punktegruppe (Seite 1942)
- Gerade (Seite 1944)
- Sinus (Seite 1945)
- Polynom (Seite 1946)
- Inverser Sinus (Seite 1948)
- Übergang (Charakteristik) (Seite 1949)

Konfiguration Elemente - Parameter (Punkt)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter des selektierten Punkts:

Parameter/Option	Beschreibung
Leitwert des Punkts	
Leitwert	Konfigurieren Sie in diesem Feld den Leitwert des Punkts (Wert im Definitionsbereich).
Folgewerte des Punkts	
Folgewert	Konfigurieren Sie in diesem Feld den Folgewert des Punkts (Wert im Wertebereich).
Erste Ableitung vorgeben	Aktivieren Sie das Optionskästchen, um die erste Ableitung im selektierten Punkt vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Erste Ableitung	Konfigurieren Sie in diesem Feld den Wert der ersten Ableitung im selektierten Punkt.
Zweite Ableitung vorgeben	Aktivieren Sie das Optionskästchen, um die zweite Ableitung im selektierten Punkt vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Zweite Ableitung	Konfigurieren Sie in diesem Feld den Wert der zweiten Ableitung im selektierten Punkt.
Dritte Ableitung vorgeben	Aktivieren Sie das Optionskästchen, um die dritte Ableitung im selektierten Punkt vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Dritte Ableitung	Konfigurieren Sie in diesem Feld den Wert der dritten Ableitung im selektierten Punkt.

Die Ableitungen werden nur bei der VDI-basierten Optimierung von Übergängen des Punkts zu anderen Elementen berücksichtigt.

Siehe auch

Punkt einfügen (Seite 1926)

Konfiguration Elemente - Parameter (Punktegruppe)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter der selektierten Punktegruppe:

Parameter/Option	Beschreibung
Leitwerte der Punktegruppe	
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt der Punktegruppe im Leitwertbereich (Definitionsbereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt der Punktegruppe im Leitwertbereich (Definitionsbereich).
Stützpunkte	

Parameter/Option	Beschreibung
Defintionsart der Leitwerte	Wählen Sie in der Klappliste, wie die Leitwerte der Stützpunkte angegeben sind: • Relativ zum Segment Sie geben die Leitwerte der Stützpunkte relativ zur Punktegruppe von 0.0 bis 1.0 an. Der Wert 0.0 entspricht dem Anfang der Punktegruppe. Der Wert 1.0 entspricht dem Ende der Punktegruppe. • Absolut im Profil Sie geben die Leitwerte der Stützpunkte als absolute Werte an.
Defintionsart der Folgewerte	Wählen Sie in der Klappliste, wie die Folgewerte der Stützpunkte angegeben sind: • Relativ zum Segment Sie geben die Folgewerte der Stützpunkte relativ zum Folgewertbereich der Punktegruppe von 0.0 bis 1.0 an. Der Wert 0.0 entspricht dem konfigurierten minimalen Folgewert der Punktegruppe. Der Wert 1.0 entspricht dem konfigurierten maximalen Folgewert der Punktegruppe. • Absolut im Profil Sie geben die Folgewerte der Stützpunkte als absolute Werte an.
Minimaler Folgewert	Konfigurieren Sie in diesem Feld den minimalen Folgewert der Punktegruppe im Folgewertbereich (Wertebereich).
Maximaler Folgewert	Konfigurieren Sie in diesem Feld den maximalen Folgewert der Punktegruppe im Folgewertbereich (Wertebereich).
	Fügen Sie mit der Schaltfläche "Stützpunkt hinzufügen" einen Stützpunkt zur Punktegruppe hinzu.
Stützpunkte	Diese Tabelle zeigt die konfigurierten Stützpunkte sortiert nach aufsteigendem Leitwert. Fügen Sie Stützpunkte über die Schaltfäche hinzu. Löschen Sie Stützpunkte, indem Sie eine Zeile markieren und <Entf> drücken. Wenn Sie alle Punkte bis auf einen löschen, wird der Elementtyp von "Punktegruppe" zu "Punkt" geändert.
Leitwert	Konfigurieren Sie in diesem Feld den Leitwert des Stützpunkts (Wert im Definitionsbereich).
Folgewert	Konfigurieren Sie in diesem Feld den Folgewert des Stützpunkts (Wert im Wertebereich).
Interpolation	
Interpolationsart	Wählen Sie in der Klappliste, nach welcher Interpolationsart die Punktegruppe interpoliert wird: • Interpolation mit kubischen Splines • Interpolation mit Bézier-Splines
Approximation	

Parameter/Option	Beschreibung
Abbildungsmethode	Wählen Sie in der Klappliste die Abbildungsmethode: • Punktapproximation • Segmentapproximation
Anzahl der Stützpunkte	Konfigurieren Sie in diesem Feld die Anzahl der Stützpunkte für die Punktapproximation.
Maximale Folgewerttoleranz	Geben Sie in diesem Feld die maximal erlaubte Abweichung (absolut) der Approximation von den Stützpunkten an. Wenn der konfigurierte Wert überschritten wird, wird im grafischen Editor eine Warnung an der Punktegruppe angezeigt.

Siehe auch

Punktegruppe einfügen (Seite 1926)

Konfiguration Elemente - Parameter (Gerade)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter der selektierten Gerade:

Parameter	Beschreibung
Leitwerte der Gerade	
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt der Gerade im Leitwertbereich (Definitionsbereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt der Gerade im Leitwertbereich (Definitionsbereich).
Folgewerte der Gerade	
Definition über	Wählen Sie in der Auswahlliste, über welche Parameter die Gerade definiert wird: • Folgewerte am Anfang und am Ende • Folgewert am Anfang und Steigung • Steigung und Folgewert am Ende Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt der Gerade im Folgewertbereich (Wertebereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt der Gerade im Folgewertbereich (Wertebereich).
Steigung	Konfigurieren Sie in diesem Feld die Steigung der Gerade.

Siehe auch

Gerade einfügen (Seite 1927)

Konfiguration Elemente - Parameter (Sinus)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter des selektierten Sinuselements:

Parameter	Beschreibung
Leitwerte des Sinus	
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt des Sinuselements im Leitwertbereich (Definitionsbereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt des Sinuselements im Leitwertbereich (Definitionsbereich).
Trigonometrische Parameter	
Amplitude	Konfigurieren Sie in diesem Feld die Amplitude des Sinuselements.
Definition über	Wählen Sie in der Klappliste, wie das Sinuselement definiert wird: • Phase am Anfang und am Ende • Phase am Anfang und Periodenlänge • Phase am Anfang und Frequenz • Periodenlänge und Phase am Ende • Frequenz und Phase am Ende Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
Phasenwinkel am Anfang [°]	Konfigurieren Sie in diesem Feld den Phasenwinkel am Anfang des Sinuselements.
Phasenwinkel am Ende [°]	Konfigurieren Sie in diesem Feld den Phasenwinkel am Ende des Sinuselements.
Periodenlänge	Konfigurieren Sie in diesem Feld Periodenlänge des Sinuselements.
Frequenz	Konfigurieren Sie in diesem Feld die Frequenz des Sinuselements.
Erweiterte Parameter	

Parameter	Beschreibung
Segmentvariante	Wählen Sie in der Klappliste die Variante des Sinuselements: • Sinus • Geneigter Sinus Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt. Wenn Sie einen geneigten Sinus konfigurieren, zeigt der grafische Editor zusätzliche Orientierungslinien für die Amplitude und die Mittellage.
Verschiebung	Konfigurieren Sie in diesem Feld den Schwingungsmittelpunkt des Sinuselements.
Definition Steigung über	Wählen Sie in der Klappliste, wie der geneigte Sinus definiert wird: • Folgewerte am Anfang und am Ende • Folgewert am Anfang und Neigung • Neigung und Folgewert am Ende Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
Verschiebung am Anfang	Konfigurieren Sie in diesem Feld den Schwingungsmittelpunkt am Anfang des Sinuselements.
Verschiebung am Ende	Konfigurieren Sie in diesem Feld den Schwingungsmittelpunkt am Ende des Sinuselements.
Neigung	Konfigurieren Sie in diesem Feld die Neigung des Sinuselements.

Siehe auch

Sinus einfügen (Seite 1928)

Konfiguration Elemente - Parameter (Polynom)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter des selektierten Polynoms:

Parameter	Beschreibung
Leitwerte des Polynoms	
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt des Polynoms im Leitwertbereich (Definitionsbereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt des Polynoms im Leitwertbereich (Definitionsbereich).
Polynomielle Parameter	

Parameter	Beschreibung
Definition über	Wählen Sie in der Auswahlliste, wie das Polynom definiert wird: • Koeffizienten • Randwerte Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
Koeffizienten	Konfigurieren Sie in diesen Feldern die Koeffizienten der Polynomfunktion 6. Grades: $P(x) = a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ Die Koeffizienten werden in wissenschaftlicher Zahlendarstellung angezeigt, z. B. "9.6450617283e-11".
Folgewert - linker Randwert	Konfigurieren Sie in diesem Feld den Folgewert am Anfang des Polynoms.
Folgewert - rechter Randwert	Konfigurieren Sie in diesem Feld den Folgewert am Ende des Polynoms.
Erste Ableitung vorgeben	Aktivieren Sie die Optionskästchen, um die erste Ableitung im linken/rechten Randwert des Polynoms vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Erste Ableitung - linker Randwert	Konfigurieren Sie in diesem Feld die erste Ableitung (Geschwindigkeit) für den Folgewert am Anfang des Polynoms.
Erste Ableitung - rechter Randwert	Konfigurieren Sie in diesem Feld die erste Ableitung (Geschwindigkeit) für den Folgewert am Ende des Polynoms.
Zweite Ableitung vorgeben	Aktivieren Sie die Optionskästchen, um die zweite Ableitung im linken/rechten Randwert des Polynoms vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Zweite Ableitung - linker Randwert	Konfigurieren Sie in diesem Feld die zweite Ableitung (Beschleunigung) für den Folgewert am Anfang des Polynoms.
Zweite Ableitung - rechter Randwert	Konfigurieren Sie in diesem Feld die zweite Ableitung (Beschleunigung) für den Folgewert am Ende des Polynoms.
Dritte Ableitung vorgeben	Aktivieren Sie die Optionskästchen, um die dritte Ableitung im linken/rechten Randwert des Polynoms vorzugeben und in die Interpolation der Kurvenscheibe mit einzubeziehen.
Dritte Ableitung - linker Randwert	Konfigurieren Sie in diesem Feld die dritte Ableitung (Ruck) für den Folgewert am Anfang des Polynoms.
Dritte Ableitung - rechter Randwert	Konfigurieren Sie in diesem Feld die dritte Ableitung (Ruck) für den Folgewert am Ende des Polynoms.
Lambda	Wählen Sie in der Auswahlliste, wie der Wendepunkt des Polynoms im Feld "Lambda-Position" angegeben ist: • Kein Lambda Sie geben keinen Wert an. Die Position des Wendepunkts wird automatisch berechnet. • Relativ zum Element Sie geben den Leitwert des Wendepunkts relativ zum Polynom von 0.0 bis 1.0 an. Der Wert 0.0 entspricht dem Anfang des Polynoms. Der Wert 1.0 entspricht dem Ende des Polynoms. • Absolut im Profil Sie geben den Leitwert des Wendepunkts als absoluten Wert an.
	Konfigurieren Sie in diesem Feld den Leitwert des Wendepunkts des Polynoms entsprechend der Auswahl in der Auswahlliste.

Parameter	Beschreibung
Erweiterte Parameter	
Segmentvariante	Wählen Sie in der Auswahlliste, ob das Polynom einen trigonometrischen Anteil haben soll. Bei Auswahl "Polynom mit trigonometrischem Anteil" werden analog zum Sinus die entsprechenden trigonometrischen Parameter angezeigt. Beim Umwandeln eines Sinuselements in ein Polynom wird das Sinuselement als Polynom mit trigonometrischem Anteil parametriert. Die Form des Elements bleibt erhalten.
Amplitude	Konfigurieren Sie in diesem Feld die Amplitude des trigonometrischen Anteils.
Definition über	Wählen Sie in der Auswahlliste, wie der trigonometrische Anteil definiert wird: • Phase am Anfang und am Ende • Phase am Anfang und Periodenlänge • Phase am Anfang und Frequenz • Periodenlänge und Phase am Ende • Frequenz und Phase am Ende Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
Phase am Anfang	Konfigurieren Sie in diesem Feld den Phasenwinkel am Anfang des trigonometrischen Anteils.
Phase am Ende	Konfigurieren Sie in diesem Feld den Phasenwinkel am Ende des trigonometrischen Anteils.
Periodenlänge	Konfigurieren Sie in diesem Feld Periodenlänge des trigonometrischen Anteils.
Frequenz	Konfigurieren Sie in diesem Feld die Frequenz des trigonometrischen Anteils.

Siehe auch

Polynom einfügen (Seite 1929)

Konfiguration Elemente - Parameter (Inverser Sinus)

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Der inverse Sinus ist im Definitionsbereich [-1, 1] definiert. Der inverse Sinus lässt sich für den gesamten oder einen eingeschränkten Definitionsbereich der Arkussinusfunktion berechnen.

Ein inverser Sinus wird über Stützpunkte der Arkussinusfunktion angenähert.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter des selektierten inversen Sinus:

Parameter	Beschreibung
Leitwerte des inversen Sinus	
Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt des inversen Sinus im Leitwertbereich (Definitionsbereich).
Ende	Konfigurieren Sie in diesem Feld den Endpunkt des inversen Sinus im Leitwertbereich (Definitionsbereich).
Folgewerte des inversen Sinus	
Minimum	Konfigurieren Sie in diesem Feld den Minimalwert des inversen Sinus im Folgewertbereich (Wertebereich).
Maximum	Konfigurieren Sie in diesem Feld den Maximalwert des inversen Sinus im Folgewertbereich (Wertebereich).
Definitionsbereich	
Nicht Gespiegelt/Gespiegelt	Wählen Sie, ob der inverse Sinus an der Abszisse gespiegelt werden soll.
Anfang	Konfigurieren Sie in diesem Feld den zu verwendenden Anfangspunkt im Definitionsbereich der Arkussinusfunktion.
Ende	Konfigurieren Sie in diesem Feld den zu verwendenden Endpunkt im Definitionsbereich der Arkussinusfunktion.
Approximation	
Anzahl der Stützpunkte	Konfigurieren Sie in diesem Feld die Anzahl der Stützpunkte für die Approximation.
Maximale Folgewerttoleranz	Geben Sie in diesem Feld die maximal erlaubte Abweichung (absolut) der Approximation von der Arkussinusfunktion an. Wenn der konfigurierte Wert überschritten wird, wird im grafischen Editor eine Warnung am Arkussinuselement angezeigt.

Siehe auch

Inversen Sinus einfügen (Seite 1930)

Konfiguration Elemente - Charakteristik (Übergang)

Im Konfigurationsfenster "Charakteristik" konfigurieren Sie die Parameter des selektierten Übergangs.

Charakteristik

Konfigurieren Sie in diesem Bereich die Vorgaben für die Optimierung des Übergangs:

Parameter	Beschreibung
Interpolationseinstellungen des Übergangs	
Optimierungsmethode	Wählen Sie in der Klappliste die Optimierungsmethode: • Systeminterpolation Die CPU bestimmt die Parameter der Optimierung automatisch gemäß den Einstellungen der Systeminterpolation (Seite 1936). • VDI-basierte Optimierung Sie passen die Optimierung händisch an. Die Eingaben werden automatisch entsprechend der VDI-Richtlinie 2143 angewendet.
Bewegungsaufgabe	Der Übergangstyp wird aus den Eigenschaften der Nachbarelemente des Übergangs ermittelt und in diesem Feld angezeigt.
Stetigkeit am Anfang/Ende	Wählen Sie in den Klapplisten, welcher Parameter in den Randpunkten stetig ist und zur Optimierung berücksichtigt wird: • Optimierungsvoreinstellung (Einstellung unter "Profil > Optimierungsvoreinstellungen (Seite 1936)") • Position • Geschwindigkeit (stoßfrei) • Beschleunigung (ruckfrei) • Ruck (Ruckstetigkeit nur einseitig erlaubt)
Optimierungsziel	Wählen Sie in der Klappliste das Optimierungsziel: • Optimierungsvoreinstellung (Einstellung unter "Profil > Optimierungsvoreinstellungen") • Nicht vorgegeben • Geschwindigkeit (Cv) • Beschleunigung (Ca) • Ruck (Cj) • Minimales Dynamisches Moment (Cmdyn)
Auswahl des Bewegungsgesetzes	

Parameter	Beschreibung
Bewegungsgesetz	Wählen Sie in der Klappliste das Bewegungsgesetz, nach dem optimiert wird: • Gerade • Quadratische Parabel • Sinus • Polynom • Geneigter Sinus • Modifiziertes Beschleunigungstrapez • Modifizierter Sinus • Harmonische Kombination • Doppelt-harmonischer Übergang • Sinus-Gerade-Kombination Die Auswahl wird automatisch auf die entsprechend der Bewegungsaufgabe und der gewählten Randbedingungen anwendbaren Bewegungsgesetze eingeschränkt. Abhängig vom gewählten Bewegungsgesetz werden weitere Parameter angezeigt.
Verwendeter Parameter	Wählen Sie in der Klappliste, welcher Parameter für die Optimierung berücksichtigt wird: • Lambda • Maximale Beschleunigung (Ca) • Maximale Verzögerung (Ca*) Die Auswahl wird automatisch auf die entsprechend des Bewegungsgesetzes anwendbaren Parameter eingeschränkt.
Lambda	Wählen Sie in der Klappliste, wie der Wendepunkt des Übergangs im Feld "Lambda-Position" angegeben ist: • Kein Lambda Sie geben keinen Wert an. Die Position des Wendepunkts wird automatisch berechnet. • Relativ zum Segment Sie geben den Leitwert des Wendepunkts relativ zum Übergang von 0.0 bis 1.0 an. Der Wert 0.0 entspricht dem Anfang des Übergangs. Der Wert 1.0 entspricht dem Ende des Übergangs. • Absolut im Profil Sie geben den Leitwert des Wendepunkts als absoluten Wert an.
Lambda-Position	Konfigurieren Sie in diesem Feld den Leitwert des Wendepunkts für den Übergang entsprechend der Auswahl in der Klappliste "Lambda".
Maximale Beschleunigung (Ca)	Konfigurieren Sie in diesem Feld die maximale Beschleunigung (Ca) für den Übergang.
Maximale Verzögerung (Ca*)	Konfigurieren Sie in diesem Feld die maximale Verzögerung (Ca*) für den Übergang.
Approximation	
Anzahl der Stützpunkte	Konfigurieren Sie in diesem Feld die Anzahl der Stützpunkte für die Approximation des Übergangs.
Maximale Folgewerttoleranz	Geben Sie in diesem Feld die maximal erlaubte Abweichung (absolut) der Approximation vom Bewegungsgesetz an.

Parameter	Beschreibung
	Wenn der konfigurierte Wert überschritten wird, wird im grafischen Editor eine Warnung am Übergang angezeigt.
Kennwerte des Übergangs	In diesem Bereich werden die nach VDI 2143 relevanten Kennwerte des Übergangs angezeigt. Für folgende Kennwerte werden jeweils der maximale Wert und der normierte Wert angezeigt: • Geschwindigkeit (Cv) • Beschleunigung (Ca) • Verzögerung (Ca*) • Ruck (Cj) • Dynamisches Moment (Cmdyn)

Bewegungsaufgaben nach VDI-Richtlinie 2143

Die VDI-Richtlinie 2143 unterscheidet zwischen Nutzbereichen und Bewegungsübergängen:

- Nutzbereiche entsprechen den Arbeitsschritten in einem Prozess, also den eingefügten Elementen der Kurvenscheibe.
- Bewegungsübergänge sind Übergänge zwischen Nutzbereichen, die nicht unmittelbar prozessrelevant sind, aber gewissen Randbedingungen genügen müssen (z. B. Geschwindigkeitsstetigkeit).

In Anlehnung an die VDI-Richtlinie 2143 sind folgende Bewegungsaufgaben definiert:

Bewegungsaufgaben	Bezeichnung	Eigenschaften
Rast	R	Geschwindigkeit = 0 Beschleunigung = 0
Konstante Geschwindigkeit	G	Geschwindigkeit ≠ 0 Beschleunigung = 0
Umkehr	U	Geschwindigkeit = 0 Beschleunigung ≠ 0
Bewegung	B	Geschwindigkeit ≠ 0 Beschleunigung ≠ 0

Folgendes Bild zeigt beispielhaft die Bewegungsaufgaben:

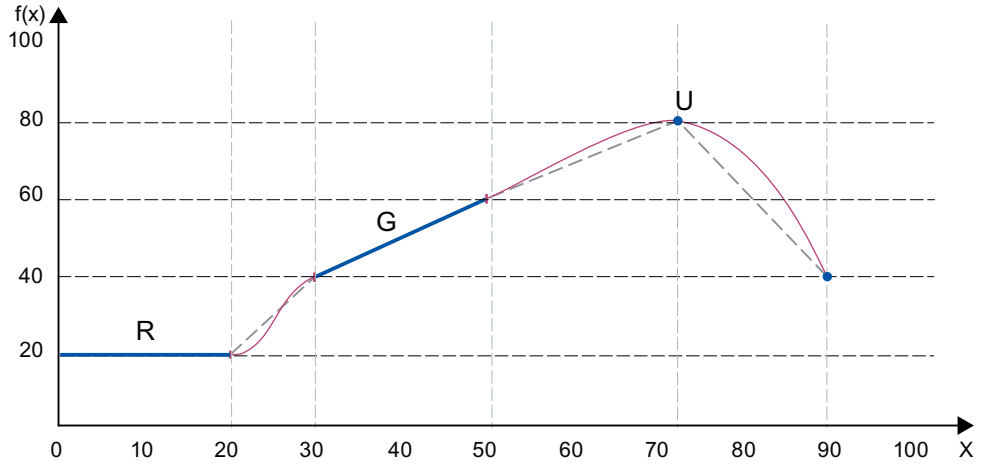

Folgendes Bild zeigt die möglichen Kombinationen von Bewegungsaufgaben:

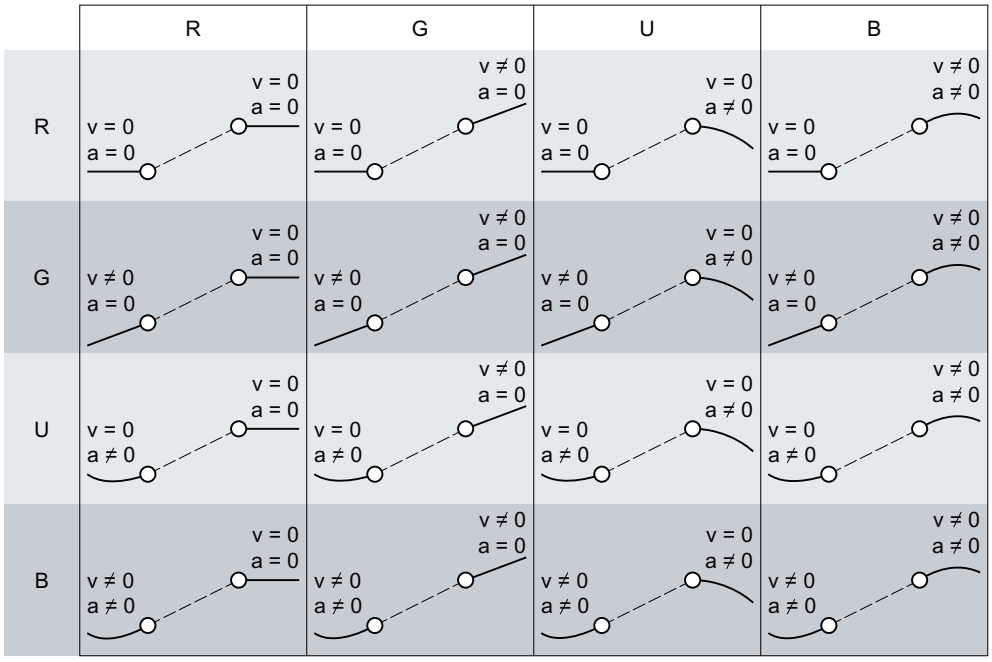

Darstellung (Inspektorfenster)

Konfiguration Diagramme - Diagramme und Kurven

Im Konfigurationsfenster "Diagramme und Kurven" konfigurieren Sie die Anzeige im grafischen Editor.

Schaltfläche "Auf Voreinstellungen zurücksetzen"

Mit dieser Schaltfläche setzen Sie alle Einstellungen der Ansicht für Diagramme und Kurven auf die Voreinstellungen zurück.

Konfigurationstabelle

In der Tabelle konfigurieren Sie die Anzeige im grafischen Editor:

Spalte	Beschreibung
Anzeigen	Einblenden/Ausblenden der Diagramme 1 bis 4
Sichtbar	Einblenden/Ausblenden von Kurven im Diagramm
Name	Name des Diagramms bzw. der Kurve Neue Kurven lassen sich hinzufügen. Vorhandene Kurven lassen sich entfernen. Kurven von anderen Kurvenscheiben lassen sich ebenfalls anzeigen. Der Name der anderen Kurvenscheibe wird ebenfalls in der Tabelle und in der Legende des Diagramms angezeigt. Eine Kurve lässt sich mehrfach in ein Diagramm einfügen, z. B. um sie mit unterschiedlichen Skalierungen anzuzeigen.
Farbe	Linienfarbe der Kurve
Linientyp	Linientyp der Kurve
Verschiebung der Leitwerte[1)]	Verschiebung der Kurve auf der Abszisse
Multiplikator für Leitwerte[1)]	Skalierung der Abszisse
Verschiebung der Folgewerte[1)]	Verschiebung der Kurve auf der Ordinate
Multiplikator für Folgewerte[1)]	Skalierung der Ordinate

1) Nur die Anzeige der Kurve im Diagramm wird beeinflusst. Die Skalierung und die Verschiebung der Kurvenscheibe im Kurvenscheibengleichlauf geben Sie an der Motion Control-Anweisung "MC_CamIn" an.

Konfiguration Diagramme - Fangraster

Im Konfigurationsfenster "Fangraster" konfigurieren Sie die Rasterabstände für die Ausrichtung von Eingaben im grafischen Editor am Raster. Bei aktiviertem "Fangen" werden Eingaben und Elementendpunkte an diesem Raster und an anderen Elementendpunkten ausgerichtet.

Abstände des Fangrasters

Konfigurieren Sie in diesem Bereich die Rasterabstände des Fangrasters:

Parameter	Beschreibung
Rasterabstand Leitwert	Konfigurieren Sie in diesem Feld den Rasterabstand auf der Abszisse (Leitwerte).
Rasterabstand Folgewert	Konfigurieren Sie in diesem Feld den Rasterabstand auf der Ordinate (Folgewerte).

Konfiguration - Nachkommastellen

Im Konfigurationsfenster "Nachkommastellen" konfigurieren Sie, mit wie vielen Nachkommstellen die Werte im grafischen und tabellarischen Editor sowie in den Konfigurationsfenstern angezeigt werden. Die Werte in den Anzeigen werden gerundet. Die Einstellungen beeinflussen nicht die Berechnung der Kurven. Die Kurven werden unabhängig von den Einstellungen mit höherer Genauigkeit berechnet.

Angezeigte Nachkommastellen

Konfigurieren Sie in diesem Bereich die angezeigten Nachkommastellen:

Parameter	Beschreibung
Tabellarischer Editor und Konfigurationsfenster	Konfigurieren Sie in diesem Feld, mit wie vielen Nachkommastellen die Werte im tabellarischen Editor und in den Konfigurationsfenstern angezeigt werden.
Grafischer Editor	Konfigurieren Sie in diesem Feld, mit wie vielen Nachkommastellen die Werte im grafischen Editor angezeigt werden.

Kurvenscheibe importieren/exportieren

Über die Werkzeugleiste exportieren Sie Kurvenscheiben aus dem Kurvenscheibeneditor und importieren Kurvenscheiben in den Kurvenscheibeneditor.

Kurvenscheibe importieren

ACHTUNG
Maschinenschäden Der Import fehlerhafter Dateien (.txt, .csv) kann zu einem unerwünschten Verhalten der Achsen führen. Prüfen Sie bei jedem Import einer Kurvenscheibe aus einer Datei die Integrität der importierten Daten.

Folgende Tabelle zeigt die unterstützten Dateiformate für den Import/Export einer Kurvenscheibe:

Dateiformat	Anmerkung
Importformat	
SIMOTION SCOUT-Format/MCD *.txt, *.csv	MCD-Exchange Format wird automatisch erkannt, importierte Daten: • Interpolierte Punkte • Geraden • Sinuselemente • Inverse Sinuselemente • Polynome • Übergänge
Proprietäres Binärformat *.bin	Das Binärformat dient zum Austausch von Kurvenscheiben zwischen mehreren TIA Portal-Installationen und externen Anwendungen.

Um eine Kurvenscheibe zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Werkzeugleiste auf das Symbol "Kurvenscheibe aus Datei importieren".
   Der Dialog "Kurvenscheibenimport" wird geöffnet.
2. Wählen Sie den Dateitypen der Datei aus, die Sie importieren wollen.
3. Wählen Sie aus dem Dateiverzeichnis die Datei aus, die Sie importieren wollen.
4. Klicken Sie auf die Schaltfläche "Öffnen".
   Die Kurvenscheibe wird im Kurvenscheibeneditor geöffnet. Alle vorherigen Eingaben im Editor werden verworfen.

Kurvenscheibe exportieren

Folgende Tabelle zeigt den Aufbau des Dialogs "Kurvenscheibenexport":

Parameter/Element	Beschreibung
Exportformat	
Export als	Wählen Sie in der Klappliste das Exportformat aus: • MCD-Exchange Format[1)] • Standard Vektorformat • Punkteliste • Binäres Format[1)]
Trennzeichen	Wählen Sie in der Klappliste, mit welchem Trennzeichen die Datenfelder in der Datei getrennt werden sollen: • Komma • Tabulator
Anzahl der Punkte	Konfigurieren Sie in diesem Feld die Anzahl der Punkte, die in eine Punkteliste exportiert werden sollen. Je mehr Punkte exportiert werden, desto genauer bildet die Punkteliste die konfigurierte Kurvenscheibe ab. Mögliche Werte: 0 bis 1E5 Voreinstellung: 360
Zusätzliche Kurven	Nur Punkteliste
Geschwindigkeit	Aktivieren Sie das Optionskästchen "Geschwindigkeit", wenn zusätzlich zur Position die Ableitungskurve der Geschwindigkeit mit exportiert werden soll.
Beschleunigung	Aktivieren Sie das Optionskästchen "Beschleunigung", wenn zusätzlich zur Position die Ableitungskurve der Beschleunigung mit exportiert werden soll.
Ruck	Aktivieren Sie das Optionskästchen "Ruck", wenn zusätzlich zur Position die Ableitungskurve des Rucks mit exportiert werden soll.
Verzeichnis für Export	
Dateiname	Geben Sie in diesem Feld einen Dateinamen ein.
Verzeichnis	Geben Sie in diesem Feld das Verzeichnis ein, in das die Datei geschrieben werden soll.
Export	Export der Datei ausführen
Abbrechen	Abbruch des Exports und Schließen des Dialogs

1) Im TIA Portal importierbar

Um eine Kurvenscheibe zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Werkzeugleiste auf das Symbol "Kurvenscheibe in Datei exportieren". Der Dialog "Kurvenscheibenexport" wird geöffnet.
2. Wählen Sie in der Klappliste "Export als" das Exportformat aus.
3. Konfigurieren Sie optional das Trennzeichen, die Anzahl der Punkte und die zusätzlichen Kurven für den Export.
4. Geben Sie im Feld "Dateiname" einen Dateinamen ein.
5. Wählen Sie das Verzeichnis aus, in das die Datei geschrieben wird.
6. Klicken Sie auf die Schaltfläche "Export".

Siehe auch

Aufbau des grafischen Editors (Seite 1923)

Dialoge im Kontextmenü

Über das Kontextmenü des grafischen und des tabellarischen Editors lassen sich folgende Dialoge aufrufen:

- Elemente einfügen
- Elemente verschieben
- Elemente skalieren

Dialog "Elemente einfügen"

Folgende Tabelle zeigt den Aufbau des Dialogs "Elemente einfügen":

Parameter/Element	Beschreibung	
Einfügemodus	Wählen Sie in der Auswahlliste den Einfügemodus:	
	Vom Ende aus nach links überschreiben	Sie überschreiben die selektierten Elemente vom Ende aus in Richtung kleinerer Leitwerte mit den Elementen in der Zwischenablage. Das Ende der eingefügten Elemente liegt dann auf dem Ende der selektierten Elemente. Elemente, die sich im Leitwertbereich der Elemente in der Zwischenablage befinden, werden überschrieben oder abgeschnitten.
	Vom Anfang aus nach rechts überschreiben	Sie überschreiben die selektierten Elemente vom Anfang aus in Richtung größerer Leitwerte mit den Elementen inder Zwischenablage. Der Anfang der eingefügten Elemente liegt dann auf dem Anfang der selektierten Elemente. Elemente, die sich im Leitwertbereich der Elemente in der Zwischenablage befinden, werden überschrieben oder abgeschnitten.
	Von der Mitte aus überschreiben	Sie überschreiben die selektierten Elemente von der Mitte aus mit den Elementen in der Zwischenablage. Die Mitte der eingfügten Elemente liegt dann auf der Mitte der selektierten Elemente. Elemente, die sich im Leitwertbereich der Elemente in der Zwischenablage befinden, werden überschrieben oder abgeschnitten.
	Auf den Leitwertbereich der Auswahl skalieren	Die Elemente in der Zwischenablage werden auf den Leitwertbereich der selektierten Elemente skaliert. Anfang und Ende der eingefügten Elemente liegen dann auf dem Anfang bzw. Ende der selektierten Elemente.
Einfügen	Einfügen der Elemente in der Zwischenablage mit dem gewählten Modus	
Abbrechen	Abbruch des Einfügens und Schließen des Dialogs	

Dialog "Elemente verschieben"

Folgende Tabelle zeigt den Aufbau des Dialogs "Elemente verschieben":

Parameter/Element	Beschreibung
Horizontale Entfernung	Geben Sie in diesem Feld die Verschiebung der Auswahl auf der Abszisse (x-Achse) ein.
Vertikale Entfernung	Geben Sie in diesem Feld die Verschiebung der Auswahl auf der Ordinate (y-Achse) ein.
Verschieben	Verschieben der Auswahl um die eingegebene Entfernung
Abbrechen	Abbruch der Verschiebung und Schließen des Dialogs

Dialog "Elemente skalieren"

Folgende Tabelle zeigt den Aufbau des Dialogs "Elemente skalieren":

Parameter/Element	Beschreibung	
Anpassen auf Leitwert-bereich	Geben Sie in diesem Feld die Skalierlänge (leitwertseitig) ein, auf die Sie die Auswahl skalieren wollen.	
Ankerpunkt	Wählen Sie in der Auswahlliste die Richtung der Skalierung:	
	Linker Rand	Die Auswahl wird vom linken Randpunkt auf die Skalierlänge angepasst.
	Mitte	Die Auswahl wird vom Mittelpunkt auf die Skalierlänge angepasst.
	Rechter Rand	Die Auswahl wird vom rechten Randpunkt auf die Skalierlänge angepasst.
Skalieren	Skalieren mit den gewählten Parameterwerten	
Abbrechen	Abbruch der Skalierung und Schließen des Dialogs	

Siehe auch

Kontextmenü im tabellarischen Editor (Seite 1934)

Kontextmenü im grafischen Editor (Seite 1931)

Technologiemodule für Motion Control konfigurieren

Einsatz von Technologiemodulen mit Motion Control

Sie können Technologiemodule und auch Kompakt-CPUs mit Motion Control einsetzen. Damit die Technologiefunktionen für Motion Control genutzt werden können, müssen Parameter der Gerätekonfiguration des Technologiemoduls bzw. der Kompakt-CPU und Parameter der Konfiguration des Technologieobjekts entsprechend konfiguriert werden. Welche Parameter für die Funktion relevant sind, entnehmen Sie der nachfolgenden Beschreibung. Weitere, hier nicht genannte Parameter können Sie optional einstellen. Die Beschreibung der Parameter finden Sie in der Dokumentation des jeweiligen Technologiemoduls.

Folgende Technologiemodule unterstützen Motion Control-Funktionalitäten:

S7-1500/ET 200 MP	ET 200 SP
TM Count 2x24V (Seite 1960) [1]	TM Count 1x24V (Seite 1960) [1]
TM PosInput 2 (Seite 1962) [1]	TM PosInput 1 (Seite 1962) [1]
TM Timer DIDQ 16x24V (Seite 1965) [2]	TM Timer DIDQ 10x24V (Seite 1965) [2]
–	TM Pulse 2x24V (Seite 1966) [1]
TM PTO 4 (Seite 1967) [3]	–
CPU 1511C-1 PN/CPU 1512C-1 PN (Seite 1969) [3]	–

[1] Automatischer Datenaustausch für Geberwerte wird unterstützt

[2] ET 200MP und ET 200SP: Taktsynchroner Betrieb erforderlich

[3] Automatischer Datenaustausch für Antriebs- bzw. Geberwerte wird unterstützt

Taktsynchronisation

Die Technologiemodule können zentral oder auch dezentral im System eingesetzt werden. Die Taktsynchronisation wird jedoch nur beim dezentralen Betrieb mit geeigneten PROFINET Interfacemodulen unterstützt.

Automatischer Datenaustausch für Antriebs- bzw. Geberwerte

Durch die Aktivierung der Optionskästchen für den automatischen Datenaustausch werden Antriebs- bzw. Geberparameter automatisch in die CPU übernommen.

Alternativ gleichen Sie die in den folgenden Tabellen beschriebenen und gekennzeichneten Parameter, je nach Antriebs- bzw. Gebertyp, händisch ab.

Die folgenden Arten für den automatischen Datenaustausch stehen zur Auswahl:

- **Offline**
  Aktivieren Sie das Optionskästchen, wenn Sie die Offline-Werte des Antriebs- bzw. des Gebers in die Konfiguration des Technologieobjekts im Projekt übernehmen möchten.
- **Online**
  Aktivieren Sie das Optionskästchen, wenn Sie die online im Antrieb- bzw. Geber wirksamen Werte zur Laufzeit in die CPU übernehmen möchten. Die Antriebs- bzw. Geberparameter werden nach der (Neu-)Initialisierung des Technologieobjekts oder dem (Wieder-)Anlauf des Antriebs bzw. Gebers oder der CPU vom Bus übernommen.

TM Count 1x24V/TM Count 2x24V

Für den Einsatz mit Motion Control ist die Konfiguration der folgenden Parameter erforderlich:

Konfiguration		
Technologiemodul	Technologieobjekt	
TM Count 1x24V/TM Count 2x24V	Achse	Externer Geber
TM Count 1x24V/TM Count 2x24V > Kanal 0/1 > Betriebsmodus	–	–
Betriebsmodus "Positionserfassung für Technologieobjekt Motion Control" wählen		

Konfiguration		
Technologiemodul TM Count 1x24V/TM Count 2x24V	Technologieobjekt	
	Achse	Externer Geber
TM Count 1x24V/TM Count 2x24V > Kanal 0/1 > Modulparameter	**Hardware-Schnittstelle > Geber**	**Hardware-Schnittstelle > Geber**
–	Datenanbindung "Geber" und den auf dem Technologiemodul für Motion Control konfigurierten Kanal als Geber wählen	Datenanbindung "Geber" und den auf dem Technologiemodul für Motion Control konfigurierten Kanal als Geber wählen
Signalart • Inkrementalgeber	Gebertyp entsprechend Konfiguration am Technologiemodul wählen • Inkrementell [1)]	Gebertyp entsprechend Konfiguration am Technologiemodul wählen • Inkrementell [1)]
–	**Hardware-Schnittstelle > Datenaustausch Geber**	**Hardware-Schnittstelle > Datenaustausch**
	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems [1)]	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems [1)]
Signalauswertung • Einfach • Zweifach • Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul eingeben [1)] • 0 = Einfach • 1 = Zweifach • 2 = Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul eingeben [1)] • 0 = Einfach • 1 = Zweifach • 2 = Vierfach
• Rotatorische Ausführung: Schritte pro Umdrehung eingeben • Lineare Ausführung: Konfiguration nicht relevant	• Rotatorische Ausführung: Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben [1)] • Lineare Ausführung: Abstand zwischen Inkrementen eingeben [1)]	• Rotatorische Ausführung: Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben [1)] • Lineare Ausführung: Abstand zwischen Inkrementen eingeben [1)]

Konfiguration		
Technologiemodul TM Count 1x24V/TM Count 2x24V	Technologieobjekt	
	Achse	Externer Geber
–	**Hardware-Schnittstelle > Datenaustausch Antrieb**	–
• Rotatorische Ausführung: Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben • Lineare Ausführung: Konfiguration nicht relevant	Bezugsdrehzahl eingeben	
–	**Referenzieren**	**Referenzieren**
Wählen Sie das Referenzsignal für Referenzmarke 0: • Signal N des Inkrementalgebers • DI0	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".
TM Count 2x24V > E/A-Adressen	–	–
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des Kanals in der Geberkonfiguration am Technologieobjekt automatisch ausgewählt.		
Prozessabbild: TPA OB Servo		

[1] Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Geberwerte (Offline)" aktiviert ist

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/Technologieobjekt erforderlich

TM PosInput 1/TM PosInput 2

Für den Einsatz mit Motion Control ist die Konfiguration der folgenden Parameter erforderlich:

Konfiguration		
Technologiemodul TM PosInput 1 / TM PosInput 2	Technologieobjekt	
	Achse	Externer Geber
TM PosInput 1/2 > Kanal 0/1 > Betriebsmodus	–	–
Betriebsmodus "Positionserfassung für Technologieobjekt Motion Control" wählen		

Konfiguration		
Technologiemodul TM PosInput 1 / TM PosInput 2	Technologieobjekt	
	Achse	Externer Geber
TM PosInput 1/2 > Kanal 0/1 > Modulparameter Im Betriebsmodus "Positionserfassung für Motion Control" stellen Sie unter "Modulparameter" die Parameter für die Gebersignale des Kanals ein. Die Parameter sind abhängig vom verwendeten Geber einzustellen. Für die Verwendung mit einem SSI-Absolutwertgeber, ist die Konfiguration des Gebers erforderlich. Informationen zur Konfiguration finden Sie in der Dokumentation zum jeweiligen Technologiemodul.	**Hardware-Schnittstelle > Geber**	**Hardware-Schnittstelle > Geber**
–	Datenanbindung "Geber" und den auf dem Technologiemodul aktivierten und konfigurierten Kanal als Geber wählen	Datenanbindung "Geber" und den auf dem Technologiemodul aktivierten und konfigurierten Kanal als Geber wählen
Signalart • Inkrementalgeber • Absolutwertgeber	Gebertyp entsprechend Konfiguration am Technologiemodul wählen: • Inkrementell [1)] • Absolut/Zyklisch absolut	Gebertyp entsprechend Konfiguration am Technologiemodul wählen: • Inkrementell [1)] • Absolut/Zyklisch absolut
–	**Hardware-Schnittstelle > Datenaustausch Geber**	**Hardware-Schnittstelle > Datenaustausch**
	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems [1)]	Wählen Sie die rotatorische oder lineare Ausführung des Messsystems [1)]

Konfiguration		
Technologiemodul **TM PosInput 1 / TM PosInput 2**	**Technologieobjekt**	
	Achse	**Externer Geber**
Signalauswertung • Einfach • Zweifach • Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul eingeben [1] • Inkrementalgeber: – 0 = Einfach – 1 = Zweifach – 2 = Vierfach • Absolutwertgeber: – 0 = Einfach	Feinauflösung entsprechend Konfiguration am Technologiemodul eingeben [1] • Inkrementalgeber: – 0 = Einfach – 1 = Zweifach – 2 = Vierfach • Absolutwertgeber: – 0 = Einfach
• Rotatorische Ausführung: Schritte pro Umdrehung eingeben • Lineare Ausführung: Konfiguration nicht relevant	• Rotatorische Ausführung: Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben [1] • Lineare Ausführung: Abstand zwischen Inkrementen eingeben [1]	• Rotatorische Ausführung: Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben [1] • Lineare Ausführung: Abstand zwischen Inkrementen eingeben [1]
–	**Hardware-Schnittstelle > Datenaustausch Antrieb**	–
• Rotatorische Ausführung: Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben • Lineare Ausführung: Konfiguration nicht relevant	Bezugsdrehzahl eingeben	
–	**Referenzieren**	**Referenzieren**
Wählen Sie das Referenzsignal für Referenzmarke 0: • Signal N des Inkrementalgebers • DI0	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".
TM PosInput 1/2 > E/A-Adressen	–	–
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des Kanals in der Geberkonfiguration am Technologieobjekt automatisch ausgewählt.		

[1] Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Geberwerte (Offline)" aktiviert ist

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/Technologieobjekt erforderlich

TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V

Das Technologiemodul TM Timer DIDQ können Sie zentral an einer CPU S7-1500 oder dezentral an einer dezentralen Peripherie betreiben. Für die Anwendung mit einem Messtaster, einem Nocken oder einer Nockenspur ist ein dezentraler Einsatz mit taktsynchronem Betrieb erforderlich.

Für den Einsatz mit Motion Control ist die Konfiguration der folgenden Parameter erforderlich:

Anwendung mit Technologieobjekt Nocken/Nockenspur

Konfiguration	
Technologiemodul **TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V**	**Technologieobjekt** **Nocken / Nockenspur**
Grundparameter	–
Gewünschte Anzahl an Ausgängen unter Kanalkonfiguration auswählen (nur ET 200MP TM Timer DIDQ 16x24V)	
Kanalparameter	**Hardware-Schnittstelle > Nockenausgabe/Ausgabe Nockenspur**
–	Ausgabe aktivieren
	Ausgabe über TM Timer DIDQ wählen
Betriebsart "Timer-DQ" für den jeweiligen Ausgang auswählen	Nockenausgang wählen
E/A-Adressen	–
"Taktsynchroner Betrieb" anwählen	
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des Kanals in der Ausgabekonfiguration am Technologieobjekt automatisch aktualisiert.	

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/Technologieobjekt erforderlich

Anwendung mit Technologieobjekt Messtaster

Konfiguration	
Technologiemodul **TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V**	**Technologieobjekt** **Messtaster**
Grundparameter	–
Gewünschte Anzahl an Eingängen unter Kanalkonfiguration auswählen	
Kanalparameter	**Hardware-Schnittstelle > Eingang Messtaster**
Konfiguration DI-Gruppe: Eingänge einzeln verwenden	–
Betriebsart "Timer-DI" für den jeweiligen Eingang wählen	Messen über TM Timer DIDQ wählen
–	Messeingang wählen

Konfiguration	
Technologiemodul TM Timer DIDQ 10x24V/TM Timer DIDQ 16x24V	Technologieobjekt Messtaster
Eingangsverzögerung applikationsabhängig wählen	–
E/A-Adressen	
"Taktsynchroner Betrieb" anwählen	
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des Kanals in der Eingangskonfiguration am Technologieobjekt automatisch aktualisiert.	

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/Technologieobjekt erforderlich

TM Pulse 2x24V

Für den Einsatz mit Motion Control ist die jeweilige Konfiguration der im Folgenden beschriebenen Parameter erforderlich.

Antriebsanbindung über PWM (Pulsweitenmodulation)

Konfiguration	
TM Pulse 2x24V	Technologieobjekt Achse
TM Pulse 2x24V > Kanalkonfiguration	–
Wählen Sie, ob 1 oder 2 Kanäle verwendet werden sollen.	
TM Pulse 2x24V > Kanal > Betriebsart	
Betriebsart "Pulsweitenmodulation PWM" oder "PWM mit DC-Motor" wählen	
TM Pulse 2x24V > Kanal > Parameter	**Hardware-Schnittstelle > Antrieb**
Ausgabeformat "S7-Analogausgabe" wählen	Analoge Antriebsanbindung wählen Für die Auswahl des Analogausgangs, legen Sie eine PLC-Variable des Typs "Int" an mit entsprechender Adresse an. Der Offset für die PLC-Variable zur Anfangsadresse ist 2. Um die Ausgabe des PWM-Signals zu aktivieren, setzen Sie im Anwenderprogramm die folgenden beiden Bits der Steuerschnittstelle des PWM-Kanals: • SW_ENABLE (=Bit 0 in Byte 9) • TM_CTRL_DQ (= Bit 1 in Byte 9) Der Offset für Byte 9 zur Anfangsadresse des PWM-Kanals ist 9.

Konfiguration	
TM Pulse 2x24V	Technologieobjekt
	Achse
TM Pulse 2x24V > Kanal > E/A-Adressen	–
Wählen Sie den Organisationsbaustein "MC-Servo" für die Eingangs- und die Ausgangsadressen aus. Das Prozessabbild "TPA OB Servo" wird für die Eingangs- und die Ausgangsadressen durch die Auswahl des Organisationsbausteins automatisch ausgewählt.	

"–" zu diesen Parametern ist keine Konfiguration am Technologieobjekt erforderlich

TM PTO 4

Für den Einsatz mit Motion Control ist die Konfiguration der folgenden Parameter erforderlich.

Konfiguration	
Technologiemodul TM PTO 4	Technologieobjekt
	Achse
TM PTO 4 > Kanalkonfiguration	–
Konfigurieren Sie, wie viele Kanäle (1 bis 4) verwendet werden sollen.	
TM PTO 4 > Kanal 0…3 > Betriebsart	
Signalart wählen: • PTO (Impuls (P) und Richtung (D)) • PTO (Vorwärts zählen (A) und rückwärts zählen (B)) • PTO (A, B phasenversetzt) • PTO (A, B phasenversetzt - vierfach)	
Signalschnittstelle wählen: • RS422, symmetrisch/TTL (5V), asymmetrisch • 24 V asymmetrisch	
Konfigurieren Sie die Impulspause bei Richtungsumkehr.	
–	Hardware-Schnittstelle > Antrieb
	Antriebstyp "PROFIdrive" und Datenanbindung "Antrieb" wählen. Als Antrieb den konfigurierten Pulsausgang am Technologiemodul wählen.
	Hardware-Schnittstelle > Geber
	Der Geber des Aktortelegramms (simulierter Geber) wird automatisch ausgewählt. Alternativ kann eine vorhandene Geberschnittstelle ausgewählt werden.

Konfiguration	
Technologiemodul TM PTO 4	Technologieobjekt Achse
TM PTO 4 > Kanal 0...3 > Diagnosealarme	-
Mit Aktivierung des Optionskästchens "Diagnosealarme freigeben" werden die Diagnosealarme aktiviert, wenn: • Versorgungsspannung fehlt • Fehler an Digitalausgängen auftreten Der erkannte Fehler wird für den jeweiligen Kanal mit dem Rückmeldebit "Fault_Present" und "Sensor_Error" angezeigt.	
TM PTO 4 > Kanal 0…3 > Achsparameter	**Datenaustausch Antrieb**
–	Telegramm "DP_TEL3_STANDARD" wird nach Auswahl des Antriebs automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Antriebswerte (online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Antriebswerte (offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben	Bezugsdrehzahl des Antriebs eingeben [1)]
Maximale Drehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben	Maximale Drehzahl des Antriebs eingeben [1)] Bei einer Überschreitung der Maximaldrehzahl wird der Technologie-Alarm 102 ausgelöst und angezeigt.
–	**Datenaustausch Geber**
	Telegramm "DP_TEL3_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
	Wählen Sie die rotatorische Ausführung des Messsystems aus [2)]
Inkremente pro Umdrehung eingeben	Inkremente pro Umdrehung entsprechend Konfiguration am Technologiemodul (1:1) eingeben [2)]
Feinauflösung konfigurieren • 0 = Einfach • 2 = Vierfach	Feinauflösung entsprechend Konfiguration am Technologiemodul eingeben [2)] • 0 = Einfach • 2 = Vierfach
Stoppverhalten konfigurieren • Zeit Schnellhalt • Zeit Rampenstopp	–

Konfiguration	
Technologiemodul TM PTO 4	Technologieobjekt Achse
TM PTO 4 > Kanal 0…3 > Hardwareein-/ausgänge	–
Wenn Sie einen Hardware-Ausgang für die Freigabe des Antriebs verwenden möchten, aktivieren Sie das Optionskästchen "Antriebsfreigabe verwenden". Wählen Sie anschließend einen der beiden Hardware-Ausgänge DQ0 oder DIQ2 aus.	Keine Einstellung am Technologieobjekt nötig. Der Ausgang wird automatisch durch den "MC_Power" gesteuert.
	Referenzieren
Aktivieren Sie den Hardware-Eingang (DI0) für den Referenznocken.	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".
Wählen Sie aus, welche Flanke am Hardware-Eingang die Referenznocken-Funktion auslösen soll.	
Bei Nutzung eines Messtasters, aktivieren Sie das Optionskästchen "DI1 als Messtaster verwenden".	**Technologieobjekt Messtaster > Konfiguration > Hardware-Schnittstelle**
	Wählen Sie den Messtastertyp "Messen über PROFIdrive-Telegramm (Antrieb oder Externer Geber)" aus. Wählen Sie unter Hardwareverbindung den Messeingang "1" aus.
Aktivieren Sie das Optionskästchen ""Antrieb bereit" verwenden". Wählen Sie in der Klappliste "Eingang "Antrieb bereit"" den Hardware-Eingang aus, der verwendet werden soll, um anzuzeigen, ob der Antrieb bereit ist.	Keine Einstellung am Technologieobjekt nötig. Bei Verwendung des Eingangs wartet der "MC_Power", bis das Eingangssignal anliegt, bevor er die Antriebsfreigabe setzt.
Eingangsverzögerung konfigurieren	–
TM PTO 4 > Kanal 0…3 > Lebenszeichen-Fehler	
Tolerierte Anzahl Lebenszeichen-Fehler konfigurieren	
TM PTO 4 > E/A-Adressen	
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des PTO-Kanals am Technologieobjekt automatisch ausgewählt.	

1) Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Antriebswerte (offline)" aktiviert ist

2) Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Geberwerte (Offline)" aktiviert ist

"–" zu diesen Parametern ist keine Konfiguration am Technologiemodul/am Technologieobjekt erforderlich

CPU 1511C-1 PN/CPU 1512C-1 PN

Für den Einsatz mit Motion Control ist die Konfiguration der im Folgenden beschriebenen Parameter erforderlich.

Antriebsanbindung über PTO (Pulse Train Output)

Konfiguration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt
	Achse
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Allgemein	–
Um einen Kanal für den PTO-Betrieb zu aktivieren, wählen Sie eine der folgenden Betriebsarten: • PTO (Impuls (A) und Richtung (B)) • PTO (Vorwärts zählen (A), rückwärts zählen (B)) • PTO (A,B phasenversetzt) • PTO (A,B phasenversetzt, vierfach)	
–	**Hardware-Schnittstelle > Antrieb**
	Antriebstyp "PROFIdrive" und Datenanbindung "Antrieb" wählen. Als Antrieb den für den PTO-Betrieb konfigurierten Impulsgenerator der CPU wählen.
	Hardware-Schnittstelle > Geber
	Der Geber des Aktortelegramms (simulierter Geber) wird automatisch ausgewählt. Alternativ kann eine vorhandene Geberschnittstelle ausgewählt werden.
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Achsparameter	**Hardware-Schnittstelle > Datenaustausch Antrieb**
–	Telegramm "DP_TEL3_STANDARD" wird nach Auswahl des Antriebs automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Antriebswerte (online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Antriebswerte (offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben	Bezugsdrehzahl des Antriebs eingeben [1)]
Maximaldrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben	Maximale Drehzahl des Antriebs eingeben [1)] Bei einer Überschreitung der Maximaldrehzahl wird der Technologie-Alarm 102 ausgelöst und angezeigt.

Konfiguration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt
	Achse
–	**Hardware-Schnittstelle > Datenaustausch Geber**
	Telegramm "DP_TEL3_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
	Wählen Sie die rotatorische Ausführung des Messsystems aus [2)]
Inkremente pro Umdrehung eingeben	Inkremente pro Umdrehung entsprechend Konfiguration an der CPU (1:1) eingeben [2)]
Die Feinauflösung ist mit festem Wert "0 Bit" (= einfach) vorbelegt und nicht änderbar.	Feinauflösung entsprechend der Konfiguration an der CPU eingeben [2)] Bits im inkr. Istwert (G1_XIST1): 0 (= einfach)
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Hardwareein-/ausgänge	**Referenzieren**
Wählen Sie den Hardware-Eingang für den Referenzschalter Konfigurieren Sie zusätzlich die Eingangsverzögerung für den ausgewählten Hardware-Eingang. Die Konfiguration der Eingangsverzögerung erfolgt in der Gerätekonfiguration am entsprechenden DI-Kanal (DI 16/DQ 16 > Eingänge > Kanal > Eingangsparameter > Eingangsverzögerung).	Für die Antriebsanbindung über PTO verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".
Wählen Sie aus, welche Flanke am Hardware-Eingang die Referenznocken-Funktion auslösen soll.	

Konfiguration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt Achse
Bei Nutzung eines Messtasters, wählen Sie den Hardware-Eingang des Messeingangs aus. Die Kofigurationsbeschreibung finden Sie in der nachfolgenden Tabelle.	–
Wählen Sie den Hardware-Eingang aus, der verwendet werden soll, um anzuzeigen, ob der Antrieb bereit ist. Konfigurieren Sie zusätzlich die Eingangsverzögerung für den ausgewählten Hardware-Eingang. Die Konfiguration der Eingangsverzögerung erfolgt in der Gerätekonfiguration am entsprechenden DI-Kanal (DI 16/DQ 16 > Eingänge > Kanal > Eingangsparameter > Eingangsverzögerung).	
Bei ausgewählter Betriebsart "PTO (Impuls (A) und Richtung (B))" wird der Hardware-Ausgang für das PTO-Signal A ("Impulsausgang (A)") durch die Gerätekonfiguration automatisch ausgewählt und kann nicht geändert werden. Für das PTO-Signal B ("Richtungsausgang (B)") wählen Sie einen der im Auswahlfeld angebotenen Hardware-Ausgänge aus. Die Hardware-Ausgänge für die PTO-Signale werden für die folgenden Betriebsarten durch die Gerätekonfiguration ausgewählt und können nicht geändert werden: • PTO (Vorwärts zählen (A), rückwärts zählen (B)) • PTO (A,B phasenversetzt) • PTO (A,B phasenversetzt, vierfach)	
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > E/A-Adressen	
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des PTO-Kanals am Technologieobjekt automatisch ausgewählt.	

1) Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Antriebswerte (offline)" aktiviert ist

2) Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Geberwerte (Offline)" aktiviert ist

"–" zu diesen Parametern ist keine Konfiguration an der CPU/am Technologieobjekt erforderlich

Zusätzliche Konfiguration für die Anwendung mit Technologieobjekt Messtaster

Konfiguration	
Technologiemodul CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt Messtaster
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Hardwareein-/ausgänge	Hardware-Schnittstelle > Eingang Messtaster
Wählen Sie den Hardware-Eingang des Messeingangs aus. Konfigurieren Sie zusätzlich die Eingangsverzögerung für den ausgewählten Hardware-Eingang. Die Konfiguration der Eingangsverzögerung erfolgt in der Gerätekonfiguration am entsprechenden DI-Kanal (DI 16/DQ 16 > Eingänge > Kanal > Eingangsparameter > Eingangsverzögerung).	Messen über PROFIdrive Telegramm (Antrieb oder Externer Geber)
	Wählen Sie im Auswahlfeld "Nummer des Messtasters" die "1" (Messtaster 1) aus.

Antriebsanbindung über PWM (Pulsweitenmodulation)

Beachten Sie, dass bei der Antriebsanbindung über die integrierte PWM-Funktion der Kompakt-CPU nur das Fahren in positiver Richtung möglich ist.

Konfiguration	
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt
	Drehzahlachse
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Allgemein	–
Betriebsart "Pulsweitenmodulation PWM" wählen	
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Hardwareein-/ausgänge	
Wählen Sie den Hardware-Ausgang aus, der für die Impulsausgabe verwendet werden soll.	
Wählen Sie aus, ob der eingestellte Hardware-Ausgang als schneller Gegentakt- oder als P-Schalter arbeiten soll.	
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > Parameter	**Hardware-Schnittstelle > Antrieb**
Ausgabeformat "S7-Analogausgabe" wählen	Analoge Antriebsanbindung wählen Für die Auswahl des Analogausgangs legen Sie eine PLC-Variable des Typs "Int" mit entsprechender Adresse an. Der Offset für die PLC-Variable der Steuerschnittstelle des PWM-Kanals ist 2. Um die Ausgabe des PWM-Signals zu aktivieren, setzen Sie im Anwenderprogramm die folgenden beiden Bits der Steuerschnittstelle des PWM-Kanals: • SW_ENABLE (=Bit 0 in Byte 9) • TM_CTRL_DQ (= Bit 1 in Byte 9) Der Offset für Byte 9 zur Anfangsadresse des PWM-Kanals ist 9.
Mindestimpulsdauer von 0 µs wählen	–
Gewünschte Periodendauer wählen (z. B. 100 µs)	
Impulsgeneratoren (PTO/PWM) > PTO1…4/PWM1…4 > E/A-Adressen	
Wählen Sie den Organisationsbaustein "MC-Servo" für die Eingangs- und die Ausgangsadressen aus. Das Prozessabbild "TPA OB Servo" wird für die Eingangs- und die Ausgangsadressen durch die Auswahl des Organisationsbausteins automatisch ausgewählt.	

"–" zu diesen Parametern ist keine Konfiguration am Technologieobjekt erforderlich

Geberanbindung über HSC (High Speed Counter)

Konfiguration		
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt	
	Achse	Externer Geber
Schnelle Zähler (HSC) > HSC 1…6 > Allgemein > Freigabe	–	–
Schnellen Zähler aktivieren		
Schnelle Zähler (HSC) > HSC 1…6 > Grundparameter > Betriebsmodus		
Betriebsmodus "Positionserfassung für Motion Control" wählen		
Schnelle Zähler (HSC) > HSC 1…6 > Grundparameter > Modulparameter	**Hardware-Schnittstelle > Geber**	**Hardware-Schnittstelle > Geber**
–	Datenanbindung "Geber" und den auf der CPU aktivierten und konfigurierten schnellen Zähler als Geber wählen	Datenanbindung "Geber" und den auf der CPU aktivierten und konfigurierten schnellen Zähler als Geber wählen
Signalart • Inkrementalgeber	Gebertyp entsprechend Gerätekonfiguration der CPU wählen [1)] • Inkrementell	Gebertyp entsprechend Gerätekonfiguration der CPU wählen [1)] • Inkrementell
–	**Hardware-Schnittstelle > Datenaustausch Geber**	**Hardware-Schnittstelle > Datenaustausch**
	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.	Telegramm "DP_TEL83_STANDARD" wird nach Auswahl des Gebers automatisch ausgewählt.
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Online)" deaktivieren
	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.	Optionskästchen "Automatischer Datenaustausch für Geberwerte (Offline)" aktivieren Ist das Optionskästchen deaktiviert, gleichen Sie die in dieser Tabelle beschriebenen und gekennzeichneten Parameter, händisch ab.
	Wählen Sie die rotatorische Ausführung des Messsystems aus [1)]	Wählen Sie die rotatorische Ausführung des Messsystems aus [1)]
Signalauswertung • Einfach • Zweifach • Vierfach	Feinauflösung entsprechend der am schnellen Zähler (HSC) konfigurierten Signalauswertung eingeben [1)] • 0 = Einfach • 1 = Zweifach • 2 = Vierfach	Feinauflösung entsprechend der am schnellen Zähler (HSC) konfigurierten Signalauswertung eingeben [1)] • 0 = Einfach • 1 = Zweifach • 2 = Vierfach
Inkremente pro Umdrehung eingeben	Inkremente pro Umdrehung entsprechend Gerätekonfiguration der CPU (1:1) eingeben [1)]	Inkremente pro Umdrehung entsprechend Gerätekonfiguration der CPU (1:1) eingeben [1)]

Konfiguration		
CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt	
	Achse	Externer Geber
–	**Hardware-Schnittstelle > Datenaustausch Antrieb**	–
Bezugsdrehzahl entsprechend Konfiguration am Technologieobjekt (1:1) eingeben	Bezugsdrehzahl eingeben	
–	**Referenzieren**	**Referenzieren**
Wählen Sie das Referenzsignal für Referenzmarke 0: • Signal N des Inkrementalgebers • DI0 (einstellbar über die Hardwareein-/ausgänge) Konfigurieren Sie zusätzlich die Eingangsverzögerung für den ausgewählten Hardware-Eingang. Die Konfiguration der Eingangsverzögerung erfolgt in der Gerätekonfiguration am entsprechenden DI-Kanal (DI 16/DQ 16 > Eingänge > Kanal > Eingangsparameter > Eingangsverzögerung).	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".	Verwenden Sie den Referenziermodus "Nullmarke über PROFIdrive-Telegramm verwenden".
Schnelle Zähler (HSC) > HSC 1…6 > E/A-Adressen	–	–
Der Organisationsbaustein ("MC-Servo") und das Prozessabbild ("TPA OB Servo") werden für die Eingangs- und die Ausgangsadressen durch die Auswahl des HSC-Kanals am Technologieobjekt automatisch ausgewählt.		

[1] Parameter werden automatisch übernommen wenn "Automatischer Datenaustausch für Geberwerte (Offline)" aktiviert ist

"–" zu diesen Parametern ist keine Konfiguration an der CPU/am Technologieobjekt erforderlich

Zusätzliche Konfiguration für die Anwendung mit Technologieobjekt Messtaster

Konfiguration	
Technologiemodul CPU 1511C-1 PN/CPU 1512C-1 PN	Technologieobjekt
	Messtaster
Schnelle Zähler (HSC) > HSC 1…6 > Hardwareein-/ausgänge	**Hardware-Schnittstelle > Eingang Messtaster**
Wählen Sie den Hardware-Eingang des Messeingangs aus. Konfigurieren Sie zusätzlich die Eingangsverzögerung für den ausgewählten Hardware-Eingang. Die Konfiguration der Eingangsverzögerung erfolgt in der Gerätekonfiguration am entsprechenden DI-Kanal (DI 16/DQ 16 > Eingänge > Kanal > Eingangsparameter > Eingangsverzögerung).	Messen über PROFIdrive Telegramm (Antrieb oder Externer Geber)
	Wählen Sie im Auswahlfeld "Nummer des Messtasters" die "1" (Messtaster 1) aus.

Antrieb/Geber über Datenbaustein anbinden

Datenbaustein für Datenanbindung erstellen

1. Erstellen Sie einen neuen Datenbaustein vom Typ "Global-DB".
2. Markieren Sie den Datenbaustein in der Projektnavigation und wählen den Kontextmenübefehl "Eigenschaften".
3. Deaktivieren Sie unter Attribute folgende Attribute und übernehmen Sie die Änderung mit "OK":
   – "Nur im Ladespeicher ablegen"
   – "Datenbaustein im Gerät schreibgeschützt"
   – "Optimierter Bausteinzugriff" für Technologieversion < V4.0
4. Öffnen Sie den Datenbaustein im Bausteineditor.
5. Fügen Sie im Bausteineditor textlich eine Variablenstruktur vom Typ "PD_TELx" ein.

In dieser Variablenstruktur finden Sie die Variablenstruktur "Input" für den Eingangsbereich des Telegramms und "Output" für den Ausgangsbereich des Telegramms.

Hinweis

"Input" und "Output" beziehen sich auf die Sicht der Lageregelung. Der Eingangsbereich beinhaltet z. B. die Istwerte des Antriebs, der Ausgangsbereich die Sollwerte für den Antrieb.

Der Datenbaustein darf die Datenstrukturen mehrerer Achsen, Geber und weitere Inhalte beinhalten.

Datenanbindung über Datenbaustein konfigurieren

1. Rufen Sie das Konfigurationsfenster "Hardware-Schnittstelle > Antrieb" bzw. "Hardware-Schnittstelle > Geber" auf.
2. Wählen Sie in der Klappliste "Datenbaustein" den Eintrag "Datenbaustein" aus.
3. Wählen Sie im Feld "Datenbaustein" den zuvor erstellten Datenbaustein aus.
   Öffnen Sie diesen und wählen Sie den für Antrieb und Geber definierten Variablennamen aus.

MC-PreServo und MC-PostServo programmieren

1. Weisen Sie dem MC-PreServo die zuvor definierte PLC-Variable des Eingangsbereichs des Datenbausteins zu.
2. Weisen Sie dem MC-PostServo die zuvor definierte PLC-Variable des Ausgangsbereichs des Datenbausteins zu.

ACHTUNG
Maschinenschäden Die unsachgemäße Manipulation der Antriebs- und Gebertelegramme kann zu unerwünschten Bewegungen des Antriebs führen. Prüfen Sie Ihr Anwenderprogramm auf Konsistenz in der Antriebs- und Geberanbindung.

Ein Anwendungsbeispiel für die Verwendung von MC-Pre- und MC-PostServo finden Sie unter:

https://support.industry.siemens.com/cs/document/109741575 (https://support.industry.siemens.com/cs/document/109741575)

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Datenanbindung Antrieb/Geber über Datenbaustein (Seite 1708)

Parametersicht

Die Parametersicht bietet Ihnen eine Gesamtübersicht über alle relevanten Parameter eines Technologieobjekts. Sie erhalten einen Überblick über die Parametereinstellungen und können diese komfortabel im Offline- und Online-Betrieb ändern.

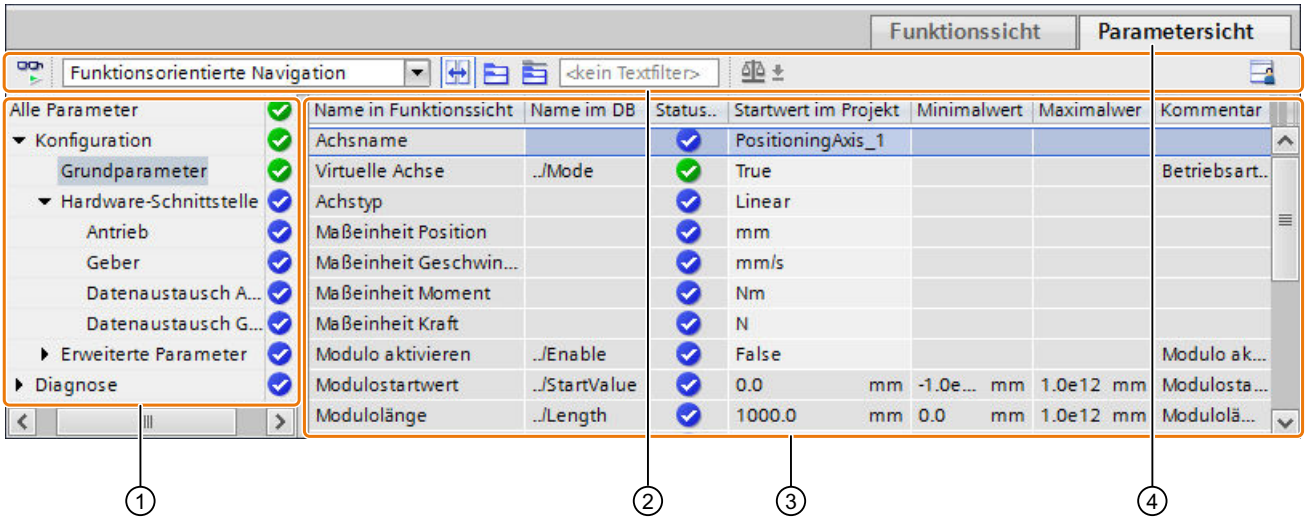

① Navigation (Seite 1979)
② Funktionsleiste (Seite 1978)
③ Parametertabelle (Seite 1980)
④ Register "Parametersicht"

Funktionsumfang

Um die Parameter der Technologieobjekte zu analysieren und gezielt beobachten und steuern zu können, stehen folgende Funktionen zur Verfügung.

Anzeigefunktionen:

- Anzeige der Parameterwerte im Offline- und Online-Betrieb
- Anzeige von Statusinformationen der Parameter
- Anzeige von Werteabweichungen und Möglichkeit der direkten Korrektur
- Anzeige von Konfigurationsfehlern
- Anzeige von Wertänderungen infolge von Parameterabhängigkeiten
- Anzeige aller Speicherwerte eines Parameters: Startwert in CPU, Startwert im Projekt, Beobachtungswert
- Anzeige des Parametervergleichs der Speicherwerte eines Parameters

Bedienfunktionen:

- Navigation, um schnell zwischen den Parametern und Parameterstrukturen zu wechseln.
- Text-Filter, um bestimmte Parameter schneller zu finden.
- Sortierfunktion, um die Reihenfolge von Parameter und Parametergruppen dem Bedarf anzupassen.
- Speicherfunktion, um strukturelle Einstellungen der Parametersicht zu sichern.
- Parameterwerte online beobachten und steuern.
- Momentaufnahme von Parameterwerten der CPU speichern, um kurzzeitige Situationen abzubilden und darauf zu reagieren.
- Momentaufnahme von Parameterwerten als Startwerte übernehmen.
- Geänderte Startwerte in die CPU laden.
- Vergleichsfunktionen, um Parameterwerte miteinander zu vergleichen.

Aufbau der Parametersicht

Funktionsleiste

In der Funktionsleiste der Parametersicht sind folgende Funktionen anwählbar:

Symbol	Funktion	Erläuterung
	Alle beobachten	Startet das Beobachten der sichtbaren Variablen in der aktiven Tabelle.
Funktionsorientierte Naviga	Navigationsstruktur auswählen	Wechselt zwischen der funktionsorientierten Navigation und der Sicht auf die Datenstruktur des Technologie-Datenbausteins.
	Koppelt die Funktions- und Parametersicht für die in der Navigation markierten Objekte	Ermöglicht den gezielten Wechsel von Parametersicht zur funktionsorientierten Sicht.

Symbol	Funktion	Erläuterung
	Ein-/Ausklappen aller Knoten und Objekte	Klappt in der jeweils aktiven Sicht alle Knoten und Objekte der Navigation bzw. der Datenstruktur ein und aus.
	Ein-/Ausklappen der Knoten unterhalb des markierten Knoten	Klappt in der jeweils aktiven Sicht die markierten Knoten und Objekte der Navigation bzw. der Datensicht ein und aus.
< kein Textfilter >	Textfilter...	Nach Eingabe einer Zeichenkette: Anzeige aller Parameter, welche die eingegebene Zeichenkette in einer der aktuell sichtbaren Spalten enthalten.
	Vergleichswerte auswählen	Auswahl, welche Parameterwerte im Online-Betrieb miteinander verglichen werden sollen (Startwert im Projekt, Startwert in CPU) Nur im Online-Betrieb.
	Anordnung merken	Speichert Ihre vorgenommenen Anzeigeeinstellungen der Parametersicht (z. B. ausgewählte Navigationsstruktur, aktivierte Tabellenspalten, usw.)

Navigation

Innerhalb des Registers "Parametersicht" sind folgende Navigationsstrukturen alternativ anwählbar:

Navigation		Erläuterung
Funktionsorientierte Navigation	Alle Parameter Konfiguration Grundparameter Hardware-Schnittstelle Antrieb Geber	In der Funktionsorientierten Navigation basiert die Struktur der Parameter auf der Struktur im Konfigurationsfenster (Register "Funktionssicht"), Inbetriebnahmefenster und Diagnosefenster.
Datenstruktur	Velocity Acceleration ActualSpeed Actor Type InverseDirection	In der Navigation "Datenstruktur" basiert die Struktur der Parameter auf der Struktur des Technologie-Datenbausteins.

Über die Klappliste "Navigationsstruktur auswählen" können Sie die Navigationsstruktur umstellen.

Parametertabelle

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten der Parametertabelle. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

Spalte	Erläuterung	Offline	Online
Name in Funktionssicht	Name des Parameters in der Funktionssicht. Das Anzeigefeld ist leer bei Parametern, die nicht über das Technologieobjekt konfiguriert werden.	X	X
Name im DB	Name des Parameters im Technologie-Datenbaustein. Wenn der Parameter Teil einer Struktur oder UDT ist, dann ist das Präfix "../" hinzugefügt. Das Anzeigefeld ist leer bei Parametern, die nicht im Technologie-Datenbaustein enthalten sind.	X	X
Vollständiger Name im DB	Vollständiger Pfad des Parameters im Instanz-Datenbaustein. Das Anzeigefeld ist leer bei Parametern, die nicht im Technologie-Datenbaustein enthalten sind.	X	X
Status der Konfiguration	Anzeige der Vollständigkeit der Konfiguration durch Statussymbole.	X	-
Vergleichsergebnis	Ergebnis der Funktion "Werte vergleichen". Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht.	-	X
Startwert im Projekt	Konfigurierter Startwert im Projekt. Fehleranzeige bei syntaktisch oder technologisch falsch eingegebenen Werten.	X	X
Defaultwert	Wert, mit dem der Parameter vorbelegt ist. Das Anzeigefeld ist leer bei Parametern, die nicht im Technologie-Datenbaustein enthalten sind.	X	X
Startwert in CPU	Startwert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht.	-	X
Beobachtungswert	Aktueller Wert in der CPU. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht.	-	X
Steuerwert	Wert, mit dem der Beobachtungswert geändert werden soll. Diese Spalte wird eingeblendet, wenn eine Online-Verbindung besteht.	-	X
Minimalwert	Technologisch kleinster Wert des Parameters. Ist der Minimalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: von den Startwerten im Projekt. • Online: von den Beobachtungswerten.	X	X
Maximalwert	Technologisch höchster Wert des Parameters. Ist der Maximalwert von anderen Parametern abhängig, dann wird er bestimmt: • Offline: von den Startwerten im Projekt. • Online: von den Beobachtungswerten.	X	X

Spalte	Erläuterung	Offline	Online
Einstellwert	Kennzeichnet den Parameter als Einstellwert. Diese Parameter können online initialisiert werden.	X	X
Datentyp	Datentyp des Parameters. Das Anzeigefeld ist leer bei Parametern, die nicht im Technologie-Datenbaustein enthalten sind.	X	X
Remanenz	Kennzeichnet den Wert als remanent. Die Werte remanenter Parameter bleiben auch nach Ausschalten der Versorgungsspannung erhalten.	X	X
Erreichbar aus HMI	Zeigt an, ob HMI zur Laufzeit auf diesen Parameter zugreifen kann.	X	X
Sichtbar in HMI	Zeigt an, ob der Parameter in der Auswahlliste von HMI per Voreinstellung sichtbar ist.	X	X
Kommentar	Kurzbeschreibung des Parameters.	X	X

X Diese Funktion ist im Offline-/Online-Betrieb sichtbar.

- Diese Funktion ist im Offline-/Online-Betrieb nicht sichtbar.

Parametersicht öffnen

Voraussetzung

Das Technologieobjekt wurde im Projektnavigator hinzufügt.

Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Konfiguration".
4. Wählen Sie in der rechten oberen Ecke das Register "Parametersicht".

Ergebnis

Die Parametersicht wird geöffnet. In der Parametertabelle wird jeder angezeigte Parameter durch eine Tabellenzeile repräsentiert.

Die anzeigbaren Parametereigenschaften (Tabellenspalten) sind abhängig davon, ob die Parametersicht im Offline- oder Online-Betrieb arbeitet.

Zusätzlich können Sie gezielt einzelne Tabellenspalten ein- und ausblenden.

Arbeiten mit der Parametersicht

Die folgende Tabelle gibt einen Überblick über die nachfolgend beschriebenen Funktionen der Parametersicht im Online- und Offline-Betrieb:

Funktion/Aktion	Offline	Online
Parametertabelle filtern (Seite 1982)	X	X
Parametertabelle sortieren (Seite 1983)	X	X
Parameterdaten in andere Editoren übernehmen (Seite 1983)	X	X
Fehler anzeigen (Seite 1984)	X	X
Startwerte im Projekt bearbeiten (Seite 1984)	X	X
Werte in der Parametersicht online beobachten (Seite 1985)	-	X
Werte steuern (Seite 1986)	-	X
Werte vergleichen (Seite 1987)	-	X

X Diese Funktion ist im Offline-/Online-Betrieb möglich.

- Diese Funktion ist im Offline-/Online-Betrieb nicht möglich.

Parametertabelle filtern

Sie können die Parameter in der Parametertabelle auf folgende Arten filtern:

- Mit dem Textfilter
- Mit den Untergruppen der Navigation

Beide Filtermethoden sind gleichzeitig anwendbar.

Mit dem Textfilter

Der Textfilter kann ausschließlich auf Texte in angezeigten Parameterzeilen und eingeblendeten Spalten angewendet werden.

Um mit dem Textfilter zu arbeiten, gehen Sie folgendermaßen vor:

1. Um nach einer gewünschten Zeichenkette zu filtern, geben Sie im Eingabefeld "Textfilter..." die gewünschte Zeichenkette ein.
   Die Parametertabelle zeigt nur noch diejenigen Parameter an, in denen die Zeichenkette enthalten ist.

Um die Textfilterung zurückzusetzen, stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Wählen Sie eine andere Parametergruppe in der Navigation.
- Wechseln Sie zwischen datenorientierter und funktionsorientierter Navigation.

Mit den Untergruppen der Navigation

Um mit den Untergruppen der Navigation zu filtern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Navigation auf die gewünschte Parametergruppe, z. B. "Static".
   In der Parametertabelle werden nur noch die "Static"-Parameter angezeigt. Sie können bei einigen Gruppen der Navigation weitere Untergruppen anwählen.
2. Wenn alle Parameter wieder angezeigt werden sollen, klicken Sie in der Navigation auf "Alle Parameter".

Parametertabelle sortieren

Die Werte der Parameter sind zeilenweise angeordnet. Die Parametertabelle ist nach jeder angezeigten Spalte sortierbar.

- In Spalten mit numerischen Werten wird nach der Höhe des numerischen Wertes sortiert.
- In Text-Spalten wird alphabetisch sortiert.

Spaltenweise sortieren

1. Positionieren Sie den Mauszeiger in der Kopfzelle der gewünschten Spalte.
   Der Hintergrund dieser Zelle wird blau markiert.
2. Klicken Sie auf den Spaltenkopf.

Ergebnis

Die ganze Parametertabelle wird nach der markierten Spalte sortiert. Im Spaltenkopf erscheint ein Dreieck mit Spitze nach oben.

Durch wiederholtes Klicken auf den Spaltenkopf wird die Sortierung wie folgt geändert:

Symbol	Beschreibung
▲	Parametertabelle wird aufsteigend sortiert.
▼	Parametertabelle wird absteigend sortiert.
Kein Symbol	Die Sortierung wird wieder aufgehoben. Die Parametertabelle nimmt die standardmäßige Anzeige an.

Bei der Sortierung wird der Präfix "../" in der Spalte "Name im DB" ignoriert.

Parameterdaten in andere Editoren übernehmen

Parameter der Parametersicht können Sie in folgende Editoren einfügen:

- Programmeditor
- Beobachtungstabelle
- Signaltabelle für Trace

Zum Einfügen stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Drag&Drop
- <Strg+C>/<Strg+V>
- Kopieren/Einfügen per Kontextmenü

Fehler anzeigen

Fehleranzeige

Parametrierfehler, die zu Übersetzungsfehlern führen (z. B. Grenzwertüberschreitung), werden in der Parametersicht angezeigt.

Bei jeder Eingabe eines Wertes in der Parametersicht wird die technologische und syntaktische Korrektheit überprüft und über folgende Anzeigen angezeigt:

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)
- Tabellenfeld mit rotem Hintergrund
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der erforderlichen Syntax (Format)

Übersetzungsfehler

Von der Fehlermeldung des Compilers kann direkt die Parametersicht (Funktionsorientierte Navigation) mit dem fehlerverursachenden Parameter geöffnet werden, bei denjenigen Parametern, die nicht im Konfigurationsfenster angezeigt werden.

Startwerte im Projekt bearbeiten

Mit der Parametersicht können Sie im Offline-Betrieb und Online-Betrieb die Startwerte im Projekt bearbeiten:

- Wertänderungen nehmen Sie in der Spalte “Startwert im Projekt“ der Parametertabelle vor.
- In der Spalte “Status der Konfiguration“ der Parametertabelle wird der Fortschritt der Konfiguration durch die bekannten Statussymbole aus dem Konfigurationsfenster des Technologieobjekts angezeigt.

Randbedingungen

- Wenn vom Parameter, dessen Startwert geändert wurde, andere Parameter abhängen, dann wird der Startwert der abhängigen Parameter ebenfalls angepasst.
- Wenn ein Parameter eines Technologieobjekts nicht editierbar ist, dann ist er auch in der Parametersicht nicht editierbar. Die Editierbarkeit eines Parameters kann auch von den Werten anderer Parameter abhängen.

Neue Startwerte definieren

Um Startwerte für Parameter in der Parametersicht festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Parametersicht des Technologieobjekts.
2. Tragen Sie in die Spalte "Startwert im Projekt" die gewünschten Startwerte ein. Der Wert muss dem Datentyp des Parameters entsprechen und darf den Wertebereich des Parameters nicht überschreiten.
   In den Spalten “Maximalwert“ und “Minimalwert“ sind die Grenzwerte des Wertebereichs sichtbar.

In der Spalte "Status der Konfiguration" wird der “Fortschritt“ der Konfiguration durch farbliche Symbole angezeigt.

Nach der Anpassung der Startwerte und dem Laden des Technologieobjekts in die CPU, nehmen die Parameter beim Anlauf den definierten Wert ein, sofern sie nicht als remanent deklariert sind (Spalte “Remanenz“).

Fehleranzeige

Bei Eingabe eines Startwertes wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt:

Fehlerhafte Startwerte werden angezeigt durch

- Rotes Fehlersymbol in den Spalten "Status der Konfiguration" (Offline-Betrieb) bzw. "Vergleichsergebnis" (Online-Betrieb, abhängig von der gewählten Vergleichsart)

und/oder

- Roten Hintergrund im Feld “Startwert im Projekt“
  Bei Klick auf das fehlerhafte Feld: Roll-out-Fehlermeldung mit Angabe des zulässigen Wertebereiches oder der notwendigen Syntax (Format)

Fehlerhafte Startwerte korrigieren

1. Korrigieren Sie fehlerhafte Startwerte mit Hilfe der Informationen aus der Roll-out-Fehlermeldung.
   Rotes Fehlersymbol, roter Feldhintergrund und Roll-out-Fehlermeldung werden nicht mehr angezeigt.

Nur mit fehlerfreien Startwerten kann das Projekt erfolgreich übersetzt werden.

Werte in der Parametersicht online beobachten

Sie können die Werte, die die Parameter des Technologieobjekts aktuell in der CPU einnehmen (Beobachtungswerte), direkt in der Parametersicht beobachten.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Der Parametersicht des Technologieobjekts ist geöffnet.

Vorgehen

Sobald die Parametersicht online ist, werden folgende Spalten zusätzlich angezeigt:

- Vergleichsergebnis
- Startwert in CPU
- Beobachtungswert
- Steuerwert
- Auswahl Steuerwert

Die Spalte "Beobachtungswert" zeigt die aktuellen Parameterwerte auf der CPU an.

Anzeige

Alle ausschließlich online verfügbaren Spalten werden wie folgt farbig dargestellt:

Farbe	Beschreibung
	Die Werte sind änderbar.
	Die Werte sind nicht änderbar.

Siehe auch

Parametertabelle (Seite 1980)

Werte steuern

Mit der Parametersicht können Sie Werte des Technologieobjekts in der CPU steuern. Sie können einem Parameter einmalig Werte zuweisen (Steuerwert) und diese sofort steuern. Beim Ausführen wird der Steuerauftrag schnellstmöglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

⚠ GEFAHR

Gefahr beim Steuern

Ein Verändern der Parameterwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen.

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Parametersicht des Technologieobjekts ist geöffnet.
- Der Parameter ist steuerbar. (Das zugehörige Feld in der Spalte "Steuerwert" hat die entsprechende Hintergrundfarbe.)

Vorgehen

Um Werte sofort zu steuern, gehen Sie folgendermaßen vor:

1. Geben Sie den gewünschten Steuerwert in die Spalte "Steuerwerte" der Parametertabelle ein.
   Der Parameter wird einmalig und sofort mit dem vorgegebenen Wert gesteuert. Sie können den Wert in der Spalte "Beobachtungswerte" beobachten. Das Optionskästchen für das Steuern in der Spalte "Auswahl Steuerwert" wird automatisch deaktiviert, nachdem der Steuerauftrag ausgeführt ist.

Fehleranzeige

Bei der Eingabe eines Steuerwerts wird die technologische und syntaktische Korrektheit sofort überprüft und angezeigt.

Fehlerhafte Steuerwerte erkennen Sie wie folgt:

- Das Feld “Steuerwert“ wird mit roter Hintergrundfarbe dargestellt.
- Wenn Sie auf das fehlerhafte Feld klicken, wird eine Roll-out-Fehlermeldung mit dem zulässigen Wertebereich oder der notwendigen Syntax angezeigt.

Fehlerhafte Steuerwerte

- Technologisch falsche Steuerwerte können übertragen werden.
- Syntaktisch falsche Steuerwerte können nicht übertragen werden.

Werte vergleichen

Über Vergleichsfunktionen können Sie die folgenden Speicherwerte eines Parameters vergleichen:

- Startwert im Projekt
- Startwert in CPU

Voraussetzungen

- Eine Online-Verbindung besteht.
- Das Technologieobjekt ist in die CPU geladen.
- Die Parametersicht des Technologieobjekts ist geöffnet.

Vorgehen

Um die Startwerte auf den verschiedenen Zielsystemen zu vergleichen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Vergleichswerte auswählen".
   Eine Auswahlliste mit den Vergleichsoptionen wird geöffnet:
   – Startwert im Projekt - Startwert in CPU (Standardeinstellung)
2. Wählen Sie die gewünschte Vergleichsoption.
   Die gewählte Vergleichsoption wird wie folgt ausgeführt:
   – In den Kopfzellen der beiden zum Vergleich ausgewählten Spalten erscheint ein Waage-Zeichen.
   – In der Spalte "Vergleichsergebnis" wird das Ergebnis des Vergleichs der gewählten Spalten durch Symbole angezeigt.

Symbol in Spalte "Vergleichsergebnis"

Symbol	Bedeutung
	Die Vergleichswerte sind gleich und fehlerfrei.
	Die Vergleichswerte sind ungleich und fehlerfrei.
	Mindestens einer der beiden Vergleichswerte ist technologisch oder syntaktisch falsch.
	Der Vergleich kann nicht durchgeführt werden. Mindestens einer der beiden Vergleichswerte ist nicht verfügbar (z. B. Momentaufnahme).

Symbol in der Navigation

Die Symbole werden in gleicher Weise in der Navigation angezeigt, wenn mindestens für einen der Parameter unterhalb der angezeigten Navigationsstruktur das Vergleichsergebnis zutrifft.

6.1.2.5 Programmieren

Das Kapitel "Programmieren" beinhaltet allgemeine Informationen zum Versorgen und Auswerten der Motion Control-Anweisungen und zum Technologie-Datenbaustein.

Eine Übersicht der Motion Control-Anweisungen finden Sie im Kapitel "Funktionen (Seite 1683)".

Über die Motion Control-Anweisungen im Anwenderprogramm können Sie Aufträge an das Technologieobjekt absetzen. Über die Eingangsparameter der Motion Control-Anweisungen definieren Sie den Auftrag. Der aktuelle Auftragsstatus wird an den Ausgangsparametern angezeigt.

Als weitere Schnittstelle zum Technologieobjekt steht Ihnen der Technologie-Datenbaustein zur Verfügung.

Technologie-Datenbaustein

Die Eigenschaften realer Objekte (z. B. Antriebe) werden über die Technologieobjekte konfiguriert und in einem Technologie-Datenbaustein gespeichert. Der Technologie-Datenbaustein enthält alle Konfigurationsdaten, Soll- und Istwerte sowie Statusinformationen des Technologieobjekts. Das TIA Portal erzeugt beim Anlegen des Technologieobjekts automatisch den Technologie-Datenbaustein. Auf die Daten des Technologie-Datenbausteins greifen Sie lesend und schreibend über Ihr Anwenderprogramm zu.

Eine Auflistung und Beschreibung der Variablen finden Sie im Anhang (Seite 2085).

Auswerten des Technologie-Datenbausteins

Der Zugriff auf Daten im Technologie-Datenbaustein erfolgt entsprechend dem Zugriff auf Standard-Datenbausteine. Im Technologie-Datenbaustein lässt sich nur auf Variablen mit elementaren Datentypen zugreifen. Der Zugriff auf Variablen mit zusammengesetzten Datentypen (z. B. STRUCT, ARRAY) ist nicht möglich.

Lesen von Werten aus dem Technologie-Datenbaustein

Sie können in Ihrem Anwenderprogramm Istwerte (z. B. aktuelle Position) und Statusinformationen lesen oder auch Fehlermeldungen am Technologieobjekt erkennen. Wenn Sie in Ihrem Anwenderprogramm eine Abfrage (z. B. aktuelle Geschwindigkeit) programmieren, wird der Wert direkt vom Technologieobjekt gelesen.

Das Lesen von Werten aus dem Technologie-Datenbaustein dauert länger als bei anderen Datenbausteinen. Wenn Sie Variablen mehrfach in einem Zyklus Ihres Anwenderprogramms verwenden, wird empfohlen, die Variablenwerte auf lokale Variablen zu kopieren und diese in Ihrem Programm zu verwenden.

Schreiben von Werten in den Technologie-Datenbaustein

Durch die Konfiguration des Technologieobjekts im TIA Portal werden die entsprechenden Daten im Technologie-Datenbaustein geschrieben. Nach dem Laden in die CPU sind diese Daten in der CPU auf der SIMATIC Memory Card (Ladespeicher) gespeichert.

Im Anwenderprogramm kann das Schreiben von Werten in den Technologie-Datenbaustein beispielsweise in folgenden Fällen notwendig werden:

- Anpassung der Konfiguration des Technologieobjekts (z. B. Dynamikgrenzen, Software-Endschalter)
- Verwendung von Overrides
- Anpassung der Lageregelung (z. B. Parameter "Kv")

Wertänderungen im Technologie-Datenbaustein durch Ihr Anwenderprogramm können zu unterschiedlichen Zeitpunkten wirksam werden. Die jeweilige Eigenschaft der einzelnen Variablen entnehmen Sie deren Beschreibungen im Anhang (Seite 2085):

Wirksamkeit von Änderungen	Beschreibung	
Direkt (DIR)	Änderungen schreiben Sie über direkte Zuweisungen. Die Änderungen werden mit dem Start des nächsten MC-Servo [OB91] wirksam. Die Änderungen bleiben bis zum nächsten NETZ-AUS der CPU bzw. Restart des Technologieobjekts erhalten.	
	LREAL (z. B. <TO>.Override.Velocity)	Das Technologieobjekt führt eine Bereichsprüfung des geschriebenen Werts durch und arbeitet sofort mit dem neuen Wert. Wenn beim Schreiben Bereichsgrenzen verletzt werden, korrigiert das Technologieobjekt die Werte automatisch. Bei Bereichsunterschreitung wird der Wert auf die untere Bereichsgrenze gesetzt, bei Bereichsüberschreitung auf die obere Bereichsgrenze.
	DINT/BOOL (z. B. <TO>.PositionLimits_SW.Active)	Änderungen sind nur im definierten Wertebereich zulässig. Wertänderungen außerhalb des Wertebereichs werden nicht übernommen. Wenn Sie unzulässige Werte eingeben, wird der Programmierfehler-OB (OB 121) gestartet.
Mit Aufruf der Motion Control-Anweisung (CAL) (z. B. <TO>.Sensor[n].ActiveHoming.HomePositionOffset)	Änderungen schreiben Sie über direkte Zuweisungen. Die Änderungen werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam. Die Änderungen bleiben bis zum nächsten NETZ-AUS der CPU bzw. Restart des Technologieobjekts erhalten.	
Restart (RES) (z. B. <TO>.Homing.AutoReversal)	Da bei Restart-relevanten Variablen Abhängigkeiten zu anderen Variablen bestehen, können Wertänderungen nicht zu einem beliebigen Zeitpunkt übernommen werden. Die Änderungen werden nur bei der Neuinitialisierung (Restart) des Technologieobjekts übernommen. Bei einem Restart wird das Technologieobjekt mit den Daten im Ladespeicher neu initialisiert. Änderungen schreiben Sie daher mit der erweiterten Anweisung "WRIT_DBL" (in Datenbaustein im Ladespeicher schreiben) auf den Startwert im Ladespeicher. Den Restart lösen Sie in Ihrem Anwenderprogramm über die Motion Control-Anweisung "MC_Reset" mit Parameter "Restart" = TRUE aus. Weitere Informationen zum Restart finden Sie im Kapitel "Restart von Technologieobjekten (Seite 2016)".	
Read Only (RON) (z. B. <TO>.Position)	Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.	

Hinweis

Änderungen sichern mit "WRIT_DBL"

Änderungen an direkt wirksamen Variablen gehen mit NETZ-AUS der CPU oder Restart des Technologieobjekts verloren.

Wenn Änderungen im Technologie-Datenbaustein auch nach NETZ-AUS der CPU oder Restart des Technologieobjekts erhalten bleiben sollen, müssen Sie die Änderungen mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert im Ladespeicher schreiben.

Hinweis

Einsatz der Datenbausteinfunktionen "READ_DBL" und "WRIT_DBL"

Die Datenbausteinfunktionen "READ_DBL" und "WRIT_DBL" dürfen in Zusammenhang mit den Variablen des Technologieobjekts nur auf einzelne Variablen angewendet werden. Die Datenbausteinfunktionen "READ_DBL" und "WRIT_DBL" dürfen nicht auf Datenstrukturen des Technologieobjekts angewendet werden.

Taktsynchrone Auswertung von Daten

Wenn Sie Daten des Technologie-Datenbausteins taktsynchron aus einem Motion Control-Applikationszyklus verarbeiten wollen, besteht ab der Technologieversion V3.0 die Möglichkeit, diese im MC-PreServo [OB67]/MC-PostServo [OB95] auszuwerten.

Siehe auch

Organisationsbausteine für Motion Control (Seite 1755)

StatusWord, ErrorWord und WarningWord auswerten

Um einzelne Status- und Fehlerinformationen aus den Datendoppelwörtern "StatusWord", "ErrorWord" und "WarningWord" symbolisch zu verwenden, können Sie sie wie nachfolgend beschrieben auswerten. Für eine konsistente Auswertung sollten Sie Bit-Adressierungen auf diese Datendoppelwörter im Technologie-Datenbaustein vermeiden. Der Zugriff auf ein einzelnes Bit im Technologie-Datenbaustein dauert genauso lange wie der Zugriff auf das gesamte Datenwort.

Kopieren Sie das benötigte Datendoppelwort bei Bedarf in Variablen einer Datenstruktur und fragen Sie die einzelnen Bits der Variable ab.

Die Belegung der einzelnen Bits in den Datendoppelwörtern finden Sie im Anhang (Seite 2085) in der Beschreibung der Variablen des entsprechenden Technologieobjekts.

Voraussetzung

Das Technologieobjekt ist angelegt.

Vorgehen

Gehen Sie zum Auswerten der einzelnen Bits im Datenwort "StatusWord" folgendermaßen vor:

1. Legen Sie eine globale Datenstruktur an. Benennen Sie die Datenstruktur z. B. als "Status".
2. Legen Sie in der Datenstruktur "Status" ein Doppelwort (DWORD) an. Benennen Sie das Doppelwort z. B. als "Temp".
3. Legen Sie in der Datenstruktur "Status" 32 boolesche Variablen an. Benennen Sie die einzelnen booleschen Variablen zur besseren Übersicht identisch zu den Bits im Technologie-Datenbaustein (z. B. die fünfte boolesche Variable als "HomingDone" benennen).

4. Kopieren Sie die Variable "<TO>.StatusWord" aus dem Technologie-Datenbaustein bei Bedarf auf das Doppelwort "Temp" in Ihrer Datenstruktur.
5. Kopieren Sie die einzelnen Bits des Doppelwortes "Temp" mit Bitzugriffen auf die entsprechenden booleschen Variablen.
6. Fragen Sie die einzelnen Statusbits über die booleschen Variablen ab.

Werten Sie die Datenwörter "ErrorWord" und "WarningWord" entsprechend Schritt 1 bis 6 aus.

Beispiel

Folgendem Beispiel entnehmen Sie, wie Sie das fünfte Bit "HomingDone" des Datenworts "StatusWord" auslesen und sichern können:

SCL	Erläuterung
`#Status.Temp := "TO".StatusWord;`	`//Statuswort kopieren`
`#Status.HomingDone := #Status.Temp.%X5;`	`//Kopieren des einzelnen Bits per Bitzugriff`

AWL	Erläuterung
`L "TO".StatusWord`	`//Statuswort kopieren`
`T #Status.Temp`	
`U #Status.Temp.%X5`	`//Kopieren des einzelnen Bits per Bitzugriff`
`= #Status.HomingDone`	

Restart-relevante Daten ändern

Um Restart-relevante Daten im Technologie-Datenbaustein zu ändern, schreiben Sie mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert der Variablen im Ladespeicher. Damit die Änderungen übernommen werden, muss ein Restart des Technologieobjekts durchgeführt werden.

Ob Wertänderungen einer Variable Restart-relevant sind, entnehmen Sie der Beschreibung der Variablen im Anhang (Seite 2102).

Voraussetzung

Das Technologieobjekt ist angelegt.

Vorgehen

Gehen Sie zum Ändern Restart-relevanter Daten folgendermaßen vor:

1. Legen Sie einen Datenbaustein an und füllen Sie ihn mit den Restart-relevanten Werten, die Sie im Technologie-Datenbaustein ändern wollen. Dabei müssen die Datentypen den zu ändernden Variablen entsprechen.
2. Schreiben Sie mit der erweiterten Anweisung "WRIT_DBL" die Variablen aus Ihrem Datenbaustein auf den Startwert der Variablen des Technologie-Datenbausteins im Ladespeicher.
   Wenn Restart-relevante Daten geändert wurden, wird dies in der Variable "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" des Technologieobjekts angezeigt.
3. Führen Sie mit der Motion Control-Anweisung "MC_Reset" mit Parameter "Restart" = TRUE einen Restart des Technologieobjekts durch.

Nach dem Restart des Technologieobjekts ist der neue Wert in den Technologie-Datenbaustein im Arbeitsspeicher übernommen und wirksam.

Motion Control-Anweisungen

Parameter der Motion Control-Anweisungen

Die einzelnen Motion Control-Anweisungen sind ausführlich im Kapitel "AUTOHOTSPOT" beschrieben.

Berücksichtigen Sie bei der Erstellung Ihres Anwenderprogramms die nachfolgenden Erläuterungen zu den Parametern der Motion Control-Anweisungen.

Referenz auf das Technologieobjekt

Die Angabe des Technologieobjekts an der Motion Control-Anweisung erfolgt folgendermaßen:

- **Parameter "Axis"**
  Am Eingangsparameter "Axis" einer Motion Control-Anweisung wird eine Referenz auf das Technologieobjekt angegeben, welches den entsprechenden Auftrag ausführen soll.
  An folgenden Parametern wird ebenfalls auf das entsprechende Technologieobjekt referenziert:
  - Parameter "Master"
  - Parameter "Slave"
  - Parameter "Cam"
  - Parameter "MeasuringInput"
  - Parameter "OutputCam"
  - Parameter "CamTrack"

  Ab der Technologieversion V3.0 lässt sich die Referenz auf das Technologieobjekt auch über den Datentyp "DB_ANY" angeben.

Auftragsstart und Übernahme der Eingangsparameter einer Motion Control-Anweisung

Beim Start von Aufträgen und bei der Übernahme geänderter Parameterwerte wird zwischen folgenden Motion Control-Anweisungen unterschieden:

- **Motion Control-Anweisungen mit Parameter "Execute"**
  Mit einer steigenden Flanke am Parameter "Execute" wird der Auftrag gestartet und die an den Eingangsparametern anstehenden Werte übernommen.
  Nachträglich geänderte Parameterwerte werden erst beim nächsten Auftragsstart übernommen.
  Das Rücksetzen des Parameters "Execute" beendet den Auftrag nicht, hat aber Einfluss auf die Anzeigedauer des Auftragsstatus. Solange "Execute" auf "TRUE" gesetzt ist, werden die Ausgangsparameter aktualisiert. Wenn "Execute" vor Abschluss eines Auftrags rückgesetzt wird, werden die Parameter "Done", "Error" und "CommandAborted" entsprechend nur für einen Aufrufzyklus gesetzt.
- **Motion Control-Anweisungen mit Parameter "Enable"**
  Mit dem Setzen des Parameters "Enable" wird der Auftrag gestartet.
  Solange "Enable" = TRUE ist, bleibt der Auftrag aktiv und geänderte Parameterwerte werden jeweils beim Aufruf der Anweisung im Anwenderprogramm übernommen.
  Mit dem Rücksetzen des Parameters "Enable" wird der Auftrag beendet.
  Die Eingangsparameter "JogForward" und "JogBackward" der Motion Control-Anweisung "MC_MoveJog" entsprechen in ihrem Verhalten dem Parameter "Enable".

Auftragsstatus

Die folgenden Ausgangsparameter zeigen den Status der Auftragsbearbeitung an:

- **Motion Control-Anweisungen mit Parameter "Done"**
  Mit Parameter "Done" = TRUE wird der ordnungsgemäße Abschluss eines Auftrags angezeigt.
- **Motion Control-Anweisungen ohne Parameter "Done"**
  Das Erreichen des Auftragsziels wird über andere Parameter (z. B. "Status", "InVelocity") angezeigt. Weitere Informationen dazu finden Sie im Kapitel "Verfolgung laufender Aufträge (Seite 2004)".
- **Parameter "Busy"**
  Solange ein Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE". Wenn ein Auftrag beendet oder abgebrochen wurde, zeigt "Busy" den Wert "FALSE".
- **Parameter "CommandAborted"**
  Wenn ein Auftrag durch einen anderen Auftrag abgebrochen wurde, zeigt der Parameter "CommandAborted" den Wert "TRUE".
- **Parameter "Error"**
  Wenn an der Motion Control-Anweisung ein Fehler auftritt, zeigt der Parameter "Error" den Wert "TRUE". Am Parameter "ErrorID" wird die entsprechende Fehlerkennung angezeigt.

Solange der Parameter "Execute" bzw. "Enable" auf "TRUE" gesetzt ist, werden die Ausgangsparameter aktualisiert. Andernfalls werden die Parameter "Done", "Error" und "CommandAborted" entsprechend nur für einen Zyklus gesetzt.

Abbruch laufender Aufträge

Ein laufender Bewegungsauftrag wird durch Anstoß eines neuen Bewegungsauftrags abgebrochen. Dabei werden die aktuellen Dynamiksollwerte (Beschleunigung, Verzögerung, Ruck, Geschwindigkeit) auf die Werte des ablösenden Auftrags geführt.

Beispiel für das Verhalten der Parameter

Das Verhalten der Parameter von Motion Control-Anweisungen wird im folgenden Diagramm beispielhaft für zwei "MC_MoveAbsolute"-Aufträge dargestellt:

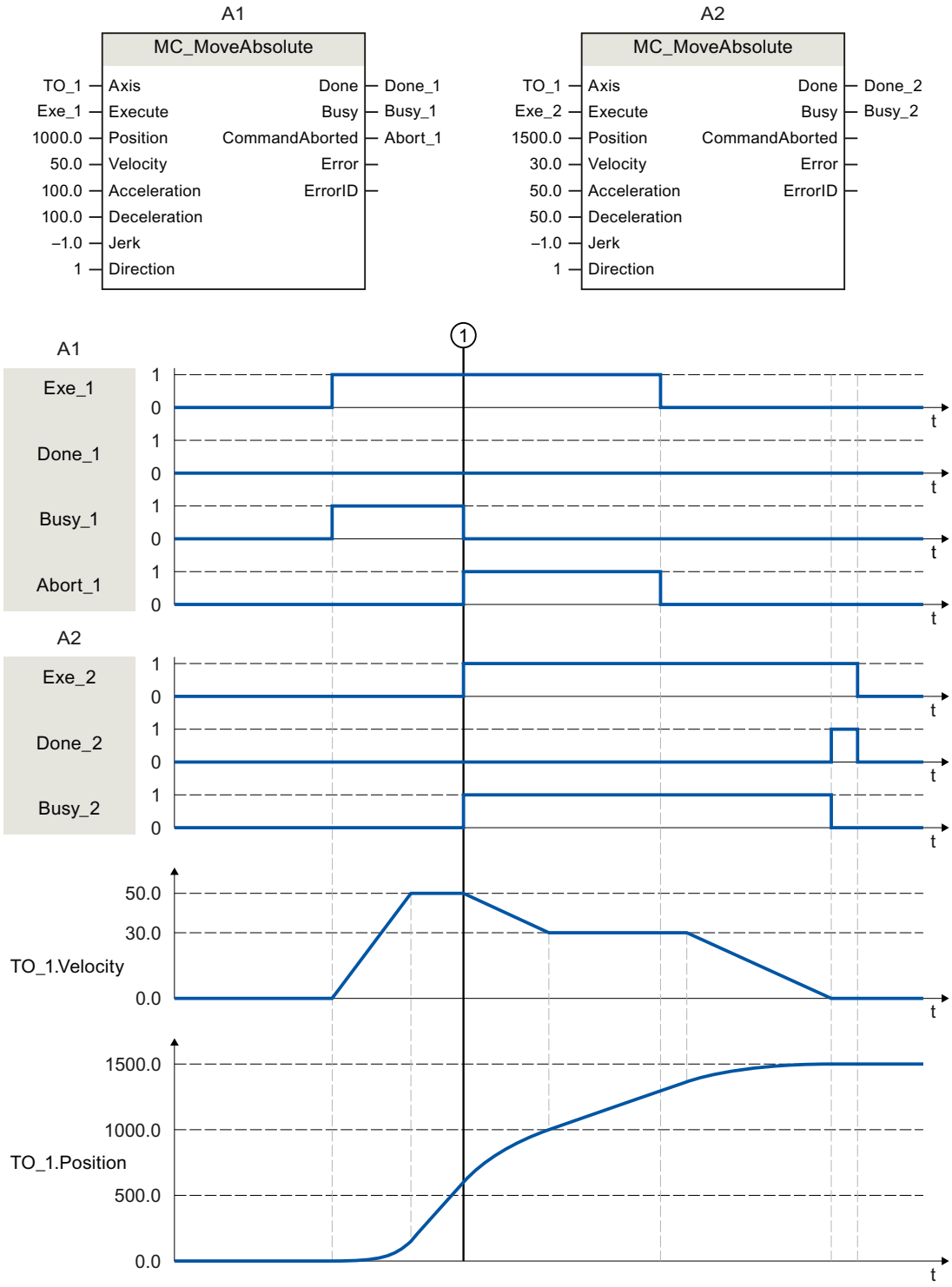

Über "Exe_1" wird ein "MC_MoveAbsolute"-Auftrag (A1) mit Zielposition 1000.0 angestoßen. "Busy_1" wird auf "TRUE" gesetzt. Die Achse wird auf die angegebene Geschwindigkeit beschleunigt und auf die Zielposition verfahren (siehe TO_1.Velocity und TO_1.Position). Vor dem Erreichen der Zielposition wird der Auftrag zum Zeitpunkt ① durch einen weiteren "MC_MoveAbsolute"-Auftrag (A2) abgelöst. Der Abbruch wird über "Abort_1" gemeldet und "Busy_1" wird auf "FALSE" gesetzt. Die Achse wird auf die an A2 angegebene Geschwindigkeit abgebremst und auf die neue Zielposition 1500.0 verfahren. Das Erreichen der Zielposition wird über "Done_2" gemeldet.

Nicht lagegeregelter Betrieb

Über folgende Parameter lässt sich die Lageregelung der Achse abschalten:

- MC_Power.StartMode = 0
- MC_MoveVelocity.PositionControlled = FALSE
- MC_MoveJog.PositionControlled = FALSE

Beachten Sie dazu auch das Kapitel "Nicht lagegeregelter Betrieb (Seite 2001)".

Motion Control-Anweisungen einfügen

Motion Control-Anweisungen fügen Sie wie andere Anweisungen in einen Programmbaustein ein. Mit den Motion Control-Anweisungen steuern Sie alle zur Verfügung stehenden Funktionen des Technologieobjekts.

Voraussetzung

Das Technologieobjekt wurde angelegt.

Vorgehen

Gehen Sie zum Einfügen der Motion Control-Anweisungen in Ihr Anwenderprogramm folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation Ihren Programmbaustein (der Programmbaustein muss im zyklischen Programm aufgerufen werden).
   Der Programmbaustein wird im Programmiereditor geöffnet und die zur Verfügung stehenden Anweisungen werden eingeblendet.
2. Öffnen Sie in der Task Card "Anweisungen" den Ordner "Technologie > Motion Control".
3. Ziehen Sie per Drag & Drop die Motion Control-Anweisung, z. B. "MC_Power", in das gewünschte Netzwerk des Programmbausteins.
   Der Dialog "Aufrufoptionen" wird geöffnet.
4. Vergeben Sie im Dialog einen Namen und eine Nummer für den Instanz-Datenbaustein der Motion Control-Anweisung.

5. Klicken Sie auf die Schaltfläche "OK".
   Die Motion Control-Anweisung "MC_Power" wird in das Netzwerk eingefügt.

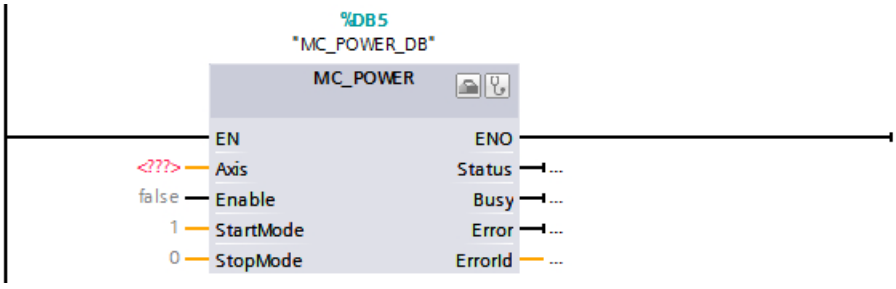

   Der Instanz-Datenbaustein wird unter "Programmbausteine > Systembausteine > Programmressourcen" automatisch angelegt.

6. Eingangsparameter ohne voreingestellten Wert (z. B. "Axis") müssen versorgt werden. Markieren Sie in der Projektnavigation das Technologieobjekt und ziehen Sie das Technologieobjekt per Drag & Drop auf <...> am Parameter "Axis".

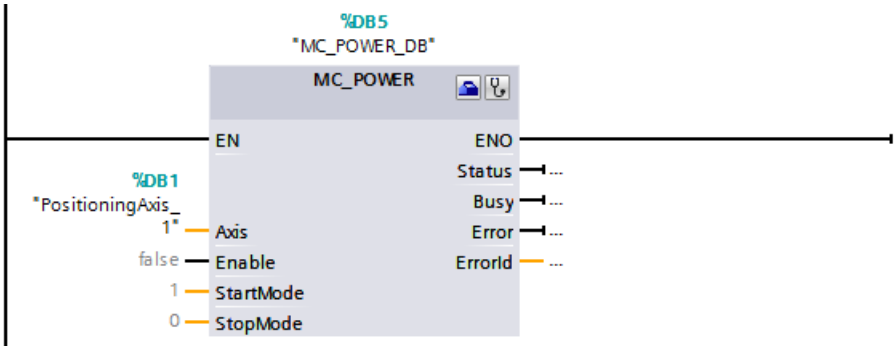

   Nach der Angabe des Technologieobjekts am Parameter "Axis" stehen Ihnen folgende Schaltflächen zur Verfügung:

	Um die Konfiguration des Technologieobjekts zu öffnen, klicken Sie auf das Werkzeugkastensymbol.
	Um die Diagnose des Technologieobjekts zu öffnen, klicken Sie auf das Stethoskopsymbol.

7. Fügen Sie weitere Motion Control-Anweisungen entsprechend Schritt 3 bis 6 ein.

Siehe auch

Verfolgung laufender Aufträge (Seite 2004)

Variablen des Technologieobjekts Positionierachse/Gleichlaufachse (Seite 2102)

Parameterübergabe für Funktionsbausteine

Um einen Funktionsbaustein mit Motion Control-Anweisungen für verschiedene Technologieobjekte wiederzuverwenden, erzeugen Sie in der Bausteinschnittstelle des aufrufenden Funktionsbausteins einen Eingangsparameter vom Datentyp des jeweiligen Technologieobjekts. Den Datentyp weisen Sie in der Bausteinschnittstelle über eine direkte Eingabe zu. Dieser Parameter wird dann als Referenz auf das Technologieobjekt an den Parameter "Axis" der Motion Control-Anweisungen übergeben. Die Datentypen von Technologieobjekten entsprechen der Struktur des zugehörigen Technologie-Datenbausteins.

Anders als Standarddatentypen werden die Datentypen der Technologieobjekte immer als Pointer an den Funktionsbaustein übergeben (Call by reference). Das gilt auch, wenn Sie die Datentypen der Technologieobjekte im Bereich "Input" der Bausteinschnittstelle deklarieren. Schreibende Zugriffe im Funktionsbaustein führen immer direkt zur Veränderung des referenzierten Technologieobjekts.

Beispiel 1: Variablenübergabe mit bestimmtem Datentyp

Durch die Angabe des Datentyps lassen sich die Variablen des Technologieobjekts im Funktionsbaustein adressieren (<Parameter der Bausteinschnittstelle>.<Variable des Technologieobjekts>). Die Datentypen für die Referenz auf die Technologieobjekte finden Sie im Anhang (Seite 2223).

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Variable	Deklaration	Datentyp	Beschreibung
axis	Input	TO_PositioningAxis	Referenz auf das Technologieobjekt
on	Input	BOOL	Signal zum Freigeben der Achse
actPosition	Output	LReal	Abfrage der Istposition aus dem Technologie-Datenbaustein
instMC_POWER	Static	MC_POWER	Multiinstanz der Motion Control-Anweisung "MC_Power"

Das folgende SCL-Programm zeigt die Variablenübergabe mit einem bestimmten Datentyp:

SCL	Erläuterung
`#instMC_POWER(Axis := #axis, Enable := #on);`	`//Aufruf der Motion Control-Anweisung "MC_Power" mit Freigabe der Achse`
`#actPosition := #axis.ActualPosition;`	`//Abfrage der Istposition aus dem Technologie-Datenbaustein`

Beispiel 2: Variablenübergabe mit "DB_Any"

Eine weitere Möglichkeit für die Übergabe bei bestimmten Datentypen des Technologieobjekts bietet der Datentyp "DB_Any". Im Gegensatz zu den Datentypen des Technologieobjekts im Programm kann "DB_Any" zur Laufzeit zugewiesen werden.

An diesem Beispiel wird gezeigt, wie Sie eine variable Umschaltung von bis zu vier Kurvenscheiben am "MC_CamIn" programmieren können.

Hierzu werden zunächst als Eingangsparameter des Bausteins Variablen vom Datentyp "DB_Any" angelegt. Durch einen weiteren Eingangsparameter wird die zu verwendende Kurvenscheibe zugewiesen.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Variable	Deklaration	Datentyp	Beschreibung
cam1	Input	DB_ANY	Kurvenscheibe 1
cam2	Input	DB_ANY	Kurvenscheibe 2
cam3	Input	DB_ANY	Kurvenscheibe 3
cam4	Input	DB_ANY	Kurvenscheibe 4
camToUse	Input	Int	Auswahl Kurvenscheibe 1 bis 4
instMC_CAMIN	Static	MC_CAMIN	Multiinstanz der Motion Control-Anweisung "MC_CamIn"
tempCamSel	Temp	DB_ANY	Aktuelle Kurvenscheibe

Das folgende SCL-Programm zeigt die Variablenübergabe mit "DB_Any":

```
SCL                                                  Erläuterung
                                                     //Auswahl der gewünschten Kurvenscheibe 1..4
                                                     //Unter Verwendung einer Eingangsvariable vom Da-
                                                     tentyp Int
CASE #camToUse OF
     1:  #tempCamSel := #cam1;                       //Anweisung für Fall 1
     2:  #tempCamSel := #cam2;                       //Anweisung für Fall 2
     3:  #tempCamSel := #cam3;                       //Anweisung für Fall 3
     4:  #tempCamSel := #cam4;                       //Anweisung für Fall 4
   ELSE                                              //Anweisung für Int ≤ 0 oder Int > 4
     #tempCamSel := #cam1;                           //Entspricht Voreinstellung Kurvenscheibe 1
END_CASE;
                                                     //Aufruf der Motion Control-Anweisung "MC_CamIn"
                                                     mit variabler Übergabe des Technologieobjekts Kur-
                                                     venscheibe unter Verwendung der temporären Variab-
                                                     le "tempCamSel"

#instMC_CAMIN(Master := "PositioningAxis_1",         //direkte Zuweisung des Technologieobjekts der
                                                     Leitachse
              Slave := "SynchronousAxis_1",          //direkte Zuweisung des Technologieobjekts der Fol-
                                                     geachse
              Cam := #tempCamSel);                   //indirekte Zuweisung des Technologieobjekts Kur-
                                                     venscheibe
```

Weitere Informationen

Weitere Programmbeispiele für die Verwendung des Datentyps "DB_Any" finden Sie im folgenden FAQ-Eintrag:

https://support.industry.siemens.com/cs/ww/de/view/109750880 (https://support.industry.siemens.com/cs/ww/de/view/109750880)

Nicht lagegeregelter Betrieb

Die Lageregelung einer Achse lässt sich über folgende Motion Control-Anweisungen abschalten/umschalten:

- MC_Power
- MC_MoveVelocity
- MC_MoveJog
- MC_MotionInVelocity

Der nicht lagegeregelte Betrieb wird über "<TO>.StatusWord.X28 (NonPositionControlled)" = TRUE angezeigt.

MC_Power

Mit "MC_Power" und dem Parameter "StartMode" = 0 wird die Achse ohne Lageregelung freigegeben. Die Lageregelung bleibt ausgeschaltet, bis eine andere Motion Control-Anweisung den Zustand der Lageregelung ändert.

MC_MoveVelocity und MC_MoveJog

Ein "MC_MoveVelocity"- oder "MC_MoveJog"-Auftrag mit "PositionControlled" = FALSE erzwingt den nicht lagegeregelten Betrieb.

Ein "MC_MoveVelocity"- oder "MC_MoveJog"-Auftrag mit "PositionControlled" = TRUE erzwingt den lagegeregelten Betrieb.

Der gewählte Betrieb bleibt nach dem Beenden des Auftrags erhalten.

MC_MotionInVelocity und MC_MotionInPosition

Ein "MC_MotionInVelocity"-Auftrag mit "PositionControlled" = FALSE erzwingt den nicht lagegeregelten Betrieb.

Ein "MC_MotionInVelocity"-Auftrag mit "PositionControlled" = TRUE erzwingt den lagegeregelten Betrieb.

Der gewählte Betrieb bleibt nach dem Beenden des Auftrags erhalten.

Ein "MC_MotionInPosition"-Auftrag erzwingt den lagegeregelten Betrieb.

Einfluss weiterer Motion Control-Anweisungen

Der Start folgender Motion Control-Anweisungen erzwingt den lagegeregelten Betrieb der Achse:

- MC_Home mit "Mode" = 3, 5
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveSuperimposed
- MC_MotionInPosition

- MC_GearIn
- MC_GearInPos (S7-1500T)
- MC_CamIn (S7-1500T)

Die Lageregelung bleibt nach dem Beenden der entsprechenden Aufträge aktiv.

Die Motion Control-Anweisung "MC_Halt" wird sowohl im lagegeregelten als auch im nicht lagegeregelten Betrieb ausgeführt. Der Zustand der Lageregelung wird durch "MC_Halt" nicht geändert.

Eine über "MC_TorqueLimiting" aktivierte Momentenbegrenzung ist auch im nicht lagegeregelten Betrieb wirksam.

Gleichlauf mit Sollwertkopplung

Eine Folgeachse wird mit dem Start eines Gleichlaufauftrags in den lagegeregelten Betrieb gesetzt. Wenn sich die Leitachse beim Start des Gleichlaufauftrags im nicht lagegeregelten Betrieb befindet, bleibt der Gleichlaufauftrag wartend. Das Aufsynchronisieren wird erst nach Aktivieren der Lageregelung und dem Erreichen der Startposition des Aufsynchronisierens gestartet.

Hinweis

Wenn die Leitachse während eines aktiven Gleichlaufs in den nicht lagegeregelten Betrieb gesetzt wird, dann wird ihr Sollwert auf null gesetzt. In Folge der Kopplung erhält der Sollwert der Folgeachse einen Sollwertsprung. Der Sollwertsprung wird entsprechend der Gleichlauffunktion ausgeglichen. Dabei wirkt einzig die maximale Drehzahl des Antriebs begrenzend.

Gleichlauf mit Istwertkopplung (S7-1500T)

Eine Folgeachse wird mit dem Start eines Gleichlaufauftrags in den lagegeregelten Betrieb gesetzt. Wenn sich die Leitachse beim Start des Gleichlaufauftrags im nicht lagegeregelten Betrieb befindet und die Istwerte gültig sind, wird aufsynchronisiert.

Wenn die Leitachse während eines aktiven Gleichlaufs in den nicht lagegeregelten Betrieb gesetzt wird, bleibt der Gleichlauf aktiv.

Start von Motion Control-Aufträgen

Das Starten von Motion Control-Aufträgen erfolgt durch Setzen des Parameters "Execute" bzw. "Enable" der Motion Control-Anweisung. Die Aufrufe der Motion Control-Anweisungen sollten für ein Technologieobjekt in einer Ablaufebene erfolgen.

Beachten Sie bei der Ausführung von Motion Control-Aufträgen auch den Status des Technologieobjekts.

Das Starten von Motion Control-Aufträgen sollte in folgenden Schritten ausgeführt werden:

1. Zustand des Technologieobjekts abfragen.
2. Neuen Auftrag für das Technologieobjekt anstoßen.
3. Auftragsstatus prüfen.

Die Schritte werden am Beispiel eines Auftrags zum absoluten Positionieren erläutert.

1. Zustand des Technologieobjekts abfragen

Stellen Sie sicher, dass sich das Technologieobjekt im entsprechenden Zustand befindet, um den gewünschten Auftrag auszuführen:

- **Ist das Technologieobjekt freigegeben?**
  Zur Ausführung von Bewegungsaufträgen muss das Technologieobjekt freigegeben sein. Die Freigabe erfolgt über die Motion Control-Anweisung "MC_Power".
  Der Parameter "MC_Power.Status" (<TO>.StatusWord.X0 (Enable)) muss den Wert "TRUE" zeigen.
- **Steht ein Technologie-Alarm an?**
  Zur Ausführung von Bewegungsaufträgen dürfen keine Technologie-Alarme oder Alarmreaktionen anstehen. Die Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts müssen den Wert null anzeigen. Quittieren Sie anstehende Alarme nach der Fehlerbehebung mit der Motion Control-Anweisung "MC_Reset".
  Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".
- **Ist das Technologieobjekt referenziert?**
  Um einen Auftrag zum absoluten Positionieren auszuführen, muss das Technologieobjekt Positionierachse/Gleichlaufachse referenziert sein. Das Referenzieren erfolgt über die Motion Control-Anweisung "MC_Home". Die Variable "<TO>.StatusWord.X5 (HomingDone)" des Technologieobjekts muss den Wert "TRUE" zeigen.

2. Neuen Auftrag für das Technologieobjekt anstoßen

Geben Sie am Parameter "Position" der "MC_MoveAbsolute"-Anweisung die Position an, auf die die Achse verfahren werden soll. Starten Sie den Auftrag mit einer steigenden Flanke am Parameter "Execute".

3. Auftragsstatus prüfen

Der fehlerfreie Abschluss eines Auftrags (hier das Erreichen der Zielposition) wird über den Parameter "Done" der Motion Control-Anweisung angezeigt.

Wenn ein Fehler erkannt wird, wird der Parameter "Error" der Motion Control-Anweisung auf "TRUE" gesetzt und der Auftrag abgelehnt.

Sie können eine Fehlerroutine für den Motion Control-Auftrag programmieren. Werten Sie dazu einen am Parameter "Error" angezeigten Fehler aus. Die Fehlerursache wird am Parameter "ErrorID" angezeigt. Starten Sie den Auftrag nach dem Beseitigen der Fehlerursache neu.

Wenn während der Auftragsbearbeitung "Error" = TRUE und "ErrorID" = 16#8001 angezeigt wird, ist ein Technologie-Alarm aufgetreten.

Eine Liste der ErrorIDs finden Sie im Anhang "Fehlerkennung (Seite 2208)".

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Status-, Fehler- und Warnungsbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Verfolgung laufender Aufträge

Der aktuelle Status der Auftragsbearbeitung wird über die Ausgangsparameter der Motion Control-Anweisung zur Verfügung gestellt. Diese Parameter werden mit jedem Aufruf der Motion Control-Anweisung aktualisiert.

Bei der Verfolgung von Aufträgen wird zwischen drei Gruppen unterschieden:

- Motion Control-Anweisungen mit Parameter "Done" (Seite 2004)
- Motion Control-Anweisungen ohne Parameter "Done" (Seite 2007)
- Motion Control-Anweisung "MC_MoveJog" (Seite 2011)

Motion Control-Anweisungen mit Parameter "Done"

Aufträge von Motion Control-Anweisungen mit Parameter "Done" werden mit einer positiven Flanke am Parameter "Execute" gestartet. Wenn der Auftrag fehlerfrei und ohne Unterbrechung durch einen anderen Auftrag abgeschlossen wurde (z. B. "MC_MoveAbsolute": Zielposition erreicht), zeigt der Parameter "Done" den Wert "TRUE".

Folgende Motion Control-Anweisungen haben einen Parameter "Done":

- MC_Home
- MC_MoveRelative
- MC_MoveAbsolute
- MC_MoveSuperimposed
- MC_Halt
- MC_Reset
- MC_MeasuringInput
- MC_AbortMeasuringInput

Nachfolgend wird das Verhalten der Parameter beispielhaft für unterschiedliche Situationen gezeigt.

Vollständige Abarbeitung des Auftrags

Wenn der Motion Control-Auftrag bis zum Abschluss vollständig abgearbeitet wurde, wird dies am Parameter "Done" mit dem Wert "TRUE" angezeigt. Der Signalzustand des Parameters "Execute" beeinflusst die Anzeigedauer am Parameter "Done":

Sie setzen "Execute" **nach Beendigung** des Auftrags auf "FALSE".	Sie setzen "Execute" **während der Bearbeitung** des Auftrags auf "FALSE".
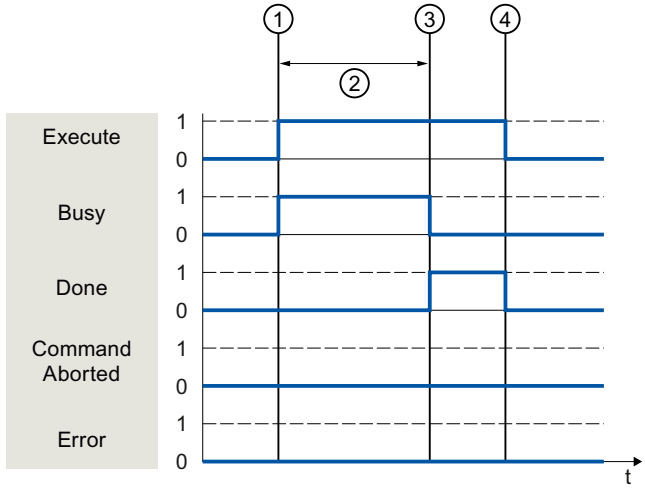	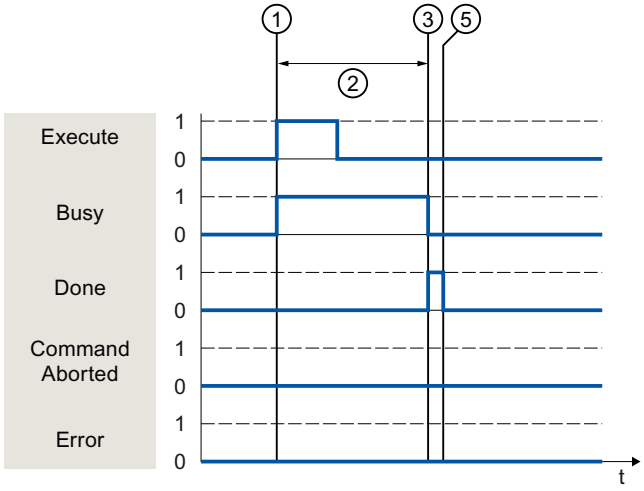

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Je nach Programmierung kann "Execute" noch während des Auftrags auf den Wert "FALSE" zurückgesetzt werden oder den Wert "TRUE" bis nach Beendigung des Auftrags beibehalten.
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Mit dem Abschluss des Auftrags (z. B. bei der Motion Control-Anweisung "MC_MoveAbsolute": Zielposition erreicht) wechselt der Parameter "Busy" nach "FALSE" und der Parameter "Done" nach "TRUE".
④	Solange der Parameter "Execute" nach Beendigung des Auftrags den Wert "TRUE" behält, bleibt auch der Parameter "Done" auf dem Wert "TRUE".
⑤	Wenn der Parameter "Execute" schon vor Beendigung des Auftrags auf "FALSE" gesetzt wurde, zeigt der Parameter "Done" nur für einen Abarbeitungszyklus den Wert "TRUE".

Abbruch des Auftrags

Wenn der Motion Control-Auftrag während der Bearbeitung durch einen anderen Auftrag abgebrochen wird, wird dies am Parameter "CommandAborted" mit dem Wert "TRUE" angezeigt. Der Signalzustand des Parameters "Execute" beeinflusst die Anzeigedauer am Parameter "CommandAborted":

Sie setzen "Execute" **nach Abbruch** des Auftrags auf "FALSE".	Sie setzen "Execute" **vor** Abbruch des Auftrags auf "FALSE".
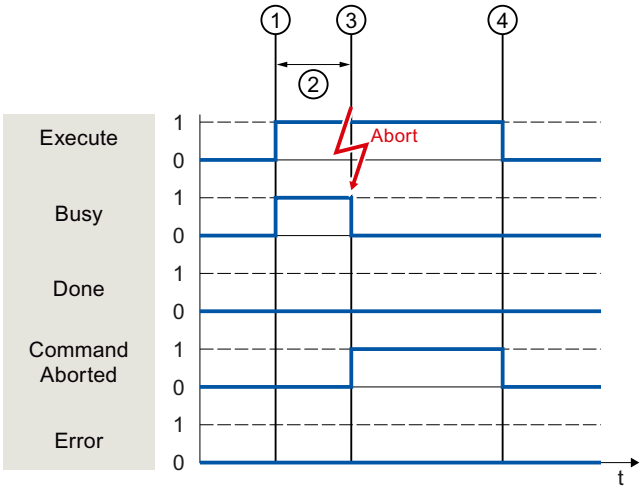	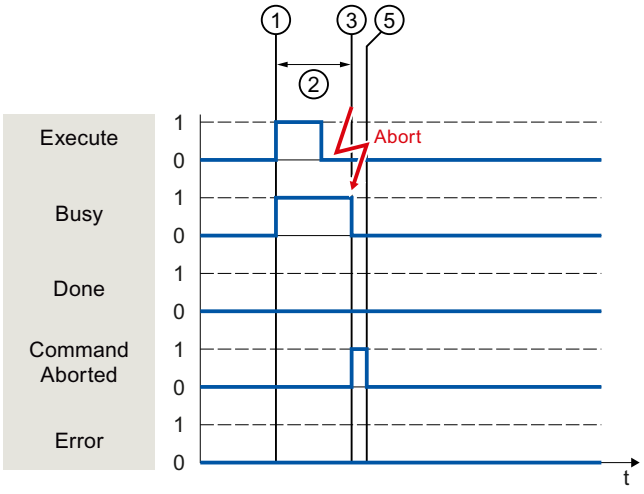

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Je nach Programmierung kann "Execute" noch während des Auftrags auf den Wert "FALSE" zurückgesetzt werden oder den Wert "TRUE" bis nach Beendigung des Auftrags beibehalten.
②	Während der Auftrag bearbeitet wird, zeigt Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Parameter "Busy" nach "FALSE" und "CommandAborted" nach "TRUE".
④	Solange der Parameter "Execute" nach Beendigung des Auftrags den Wert "TRUE" behält, bleibt auch der Parameter "CommandAborted" auf dem Wert "TRUE".
⑤	Wenn der Parameter "Execute" schon vor Abbruch des Auftrags auf "FALSE" gesetzt wurde, zeigt der Parameter "CommandAborted" nur für einen Abarbeitungszyklus den Wert "TRUE".

Fehler während der Auftragsbearbeitung

Wenn während der Bearbeitung des Motion Control-Auftrags ein Fehler auftritt, wird dies am Parameter "Error" mit dem Wert "TRUE" angezeigt. Der Signalzustand des Parameters "Execute" beeinflusst die Anzeigedauer am Parameter "Error":

Sie setzen "Execute" **nach** Auftreten des Fehlers auf "FALSE".	Sie setzen "Execute" **vor** Auftreten des Fehlers auf "FALSE".
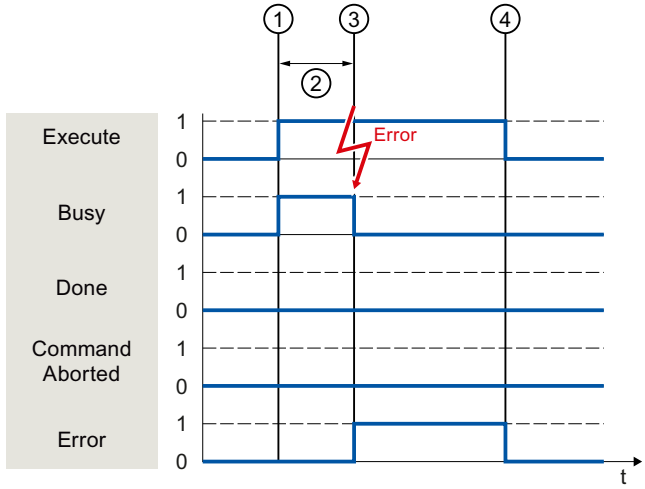	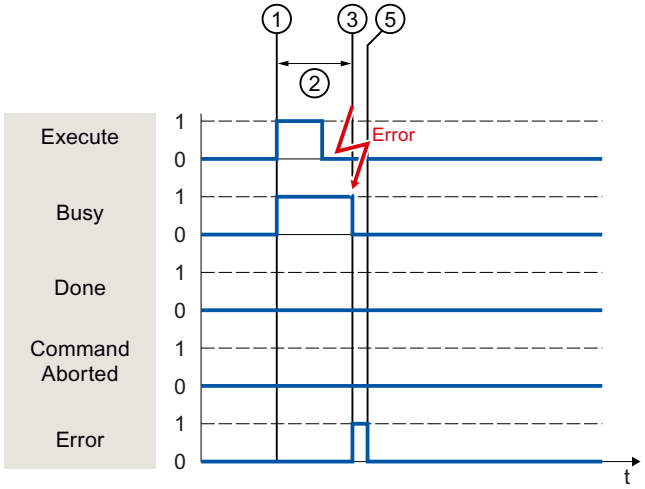

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Je nach Programmierung kann "Execute" noch während des Auftrags auf den Wert "FALSE" zurückgesetzt werden oder den Wert "TRUE" bis nach Beendigung des Auftrags beibehalten
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Parameter "Busy" nach "FALSE" und der Parameter "Error" nach "TRUE".
④	Solange der Parameter "Execute" nach Auftreten des Fehlers den Wert "TRUE" behält, bleibt auch der Parameter "Error" auf dem Wert "TRUE".
⑤	Wenn der Parameter "Execute" schon vor Auftreten des Fehlers auf "FALSE" gesetzt wurde, zeigt der Parameter "Error" nur für einen Abarbeitungszyklus den Wert "TRUE".

Motion Control-Anweisungen ohne Parameter "Done"

Motion Control-Anweisungen ohne Parameter "Done" zeigen über einen speziellen Parameter an, dass das Auftragsziel (z. B. "InVelocity", "InGear") erreicht wurde. Der Zielzustand bzw. die Bewegung wird gehalten, bis der Auftrag abgebrochen wird oder ein Fehler auftritt.

Folgende Motion Control-Anweisungen haben einen speziellen Parameter zur Anzeige des Auftragsstatus:

- MC_Power (Parameter "Status")
- MC_MoveVelocity (Parameter "InVeloctity")
- MC_GearIn (Parameter "InGear")

- MC_MoveJog (Parameter "InVelocity")
  Das besondere Verhalten von "MC_MoveJog" wird im Kapitel "Motion Control-Anweisung "MC_MoveJog" (Seite 2011)" beschrieben.
- MC_TorqueLimiting (Parameter "InClamping" und "InLimitation")

Folgende Motion Control-Anweisungen haben keinen speziellen Parameter zur Anzeige des Auftragsstatus. Die Rückmeldung erfolgt über die folgenden Variablen:

- MC_MeasuringInputCyclic
  Die Bearbeitung eines Messauftrags wird am Parameter "Busy" mit dem Wert "TRUE" angezeigt. Erfolgte Messereignisse werden im Technologie-Datenbaustein in den entsprechenden Ereigniszähler "<TO>.MeasuredValues.MeasuredValue1Counter" und "<TO>.MeasuredValues.MeasuredValue2Counter" angezeigt.
- MC_OutputCam
  Die Bearbeitung eines Auftrags wird am Parameter "Busy" mit dem Wert "TRUE" angezeigt. Die Variable "CamOutput" im zugehörigen Technologie-Datenbaustein zeigt den Schaltzustand des Nockens an.
- MC_CamTrack
  Die Bearbeitung eines Auftrags wird am Parameter "Busy" mit dem Wert "TRUE" angezeigt. Die Variable "TrackOutput" im zugehörigen Technologie-Datenbaustein zeigt den Schaltzustand der Nockenspur an.

Nachfolgend wird das Verhalten der Parameter am Beispiel der Motion Control-Anweisung "MC_MoveVelocity" für unterschiedliche Situationen gezeigt:

Beispiel "MC_MoveVelocity"

Ein "MC_MoveVelocity"-Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Das Auftragsziel ist erfüllt, wenn die parametrierte Geschwindigkeit erreicht wurde und die Achse mit konstanter Geschwindigkeit verfährt. Das Erreichen und Halten der parametrierten Geschwindigkeit wird am Parameter "InVelocity" mit dem Wert "TRUE" angezeigt.

Die Bewegung der Achse kann z. B. mit einem "MC_Halt"-Auftrag gestoppt werden.

Die parametrierte Geschwindigkeit wird erreicht und gehalten

Das Erreichen der parametrierten Geschwindigkeit wird am Parameter "InVelocity" mit dem Wert "TRUE" angezeigt. Der Parameter "Execute" hat keinen Einfluss auf die Anzeigedauer am Parameter "InVelocity".

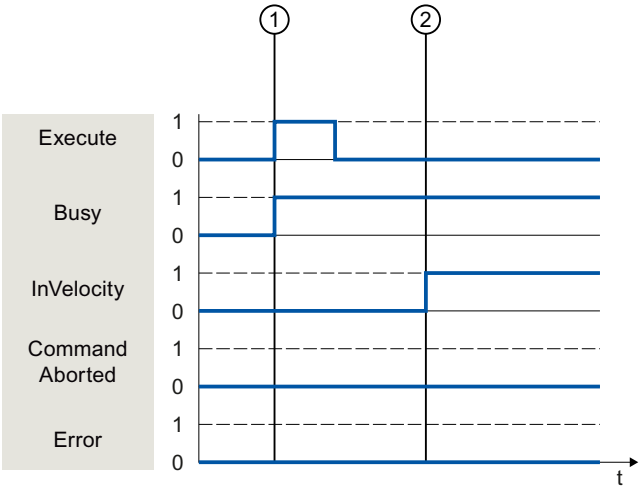

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Je nach Programmierung kann "Execute" noch vor oder erst nach Erreichen der parametrierten Geschwindigkeit auf den Wert "FALSE" zurückgesetzt werden. Während der Auftrag bearbeitet wird, zeigt Parameter "Busy" den Wert "TRUE".
②	Mit dem Erreichen der parametrierten Geschwindigkeit wechselt Parameter "InVelocity" nach "TRUE". Die Parameter "Busy" und "InVelocity" bleiben so lange auf dem Wert "TRUE", bis der "MC_MoveVelocity"-Auftrag durch einen anderen Motion Control-Auftrag abgelöst wird.

Der Auftrag wird vor dem Erreichen der parametrierten Geschwindigkeit abgebrochen

Wenn der Motion Control-Auftrag vor dem Erreichen der parametrierten Geschwindigkeit von einem anderen Auftrag abgebrochen wird, wird der Abbruch durch den Parameter "CommandAborted" mit dem Wert "TRUE" angezeigt. Der Signalzustand des Parameters "Execute" beeinflusst die Anzeigedauer am Parameter "CommandAborted".

Sie setzen "Execute" **nach** Abbruch des Auftrags auf "FALSE".	Sie setzen "Execute" **vor** Abbruch des Auftrags auf "FALSE".
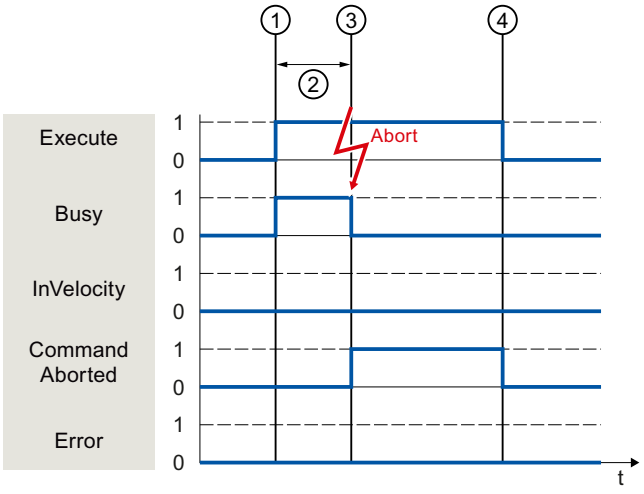	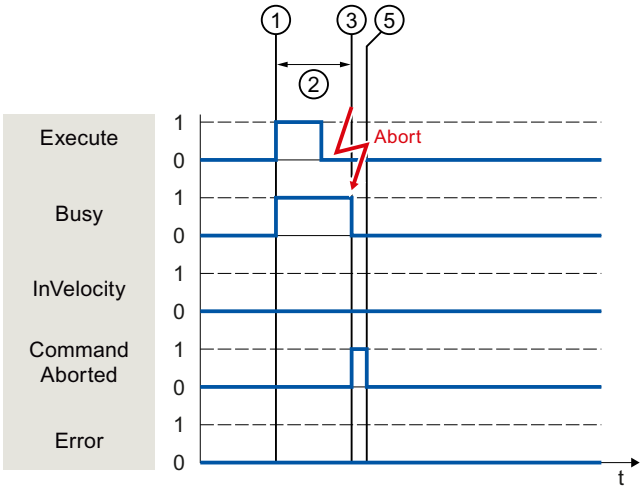

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Abhängig von der Programmierung kann "Execute" noch während des Auftrags auf den Wert "FALSE" zurückgesetzt werden oder den Wert "TRUE" bis nach Abbruch des Auftrags beibehalten.
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Parameter "Busy" nach "FALSE" und "CommandAborted" nach "TRUE".
④	Solange der Parameter "Execute" nach Beendigung des Auftrags den Wert "TRUE" behält, bleibt auch der Parameter "CommandAborted" auf dem Wert "TRUE".
⑤	Wenn der Parameter "Execute" schon vor Abbruch des Auftrags auf "FALSE" gesetzt wurde, zeigt der Parameter "CommandAborted" nur für einen Abarbeitungszyklus den Wert "TRUE".

Vor dem Erreichen der parametrierten Geschwindigkeit tritt ein Fehler auf

Wenn während der Bearbeitung des Motion Control-Auftrags vor dem Erreichen der parametrierten Geschwindigkeit ein Fehler auftritt, wird dies am Parameter "Error" mit dem Wert "TRUE" angezeigt. Der Signalzustand des Parameters "Execute" beeinflusst die Anzeigedauer am Parameter "Error".

Sie setzen "Execute" **nach** Auftreten des Fehlers auf "FALSE".	Sie setzen "Execute" **vor** Auftreten des Fehlers auf "FALSE".
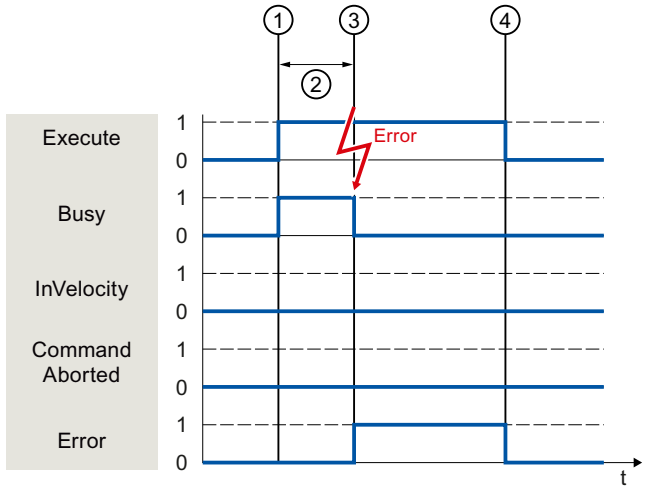	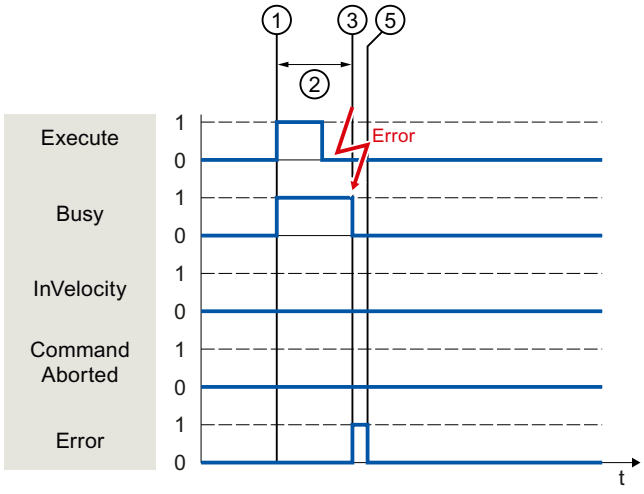

①	Der Auftrag wird mit einer positiven Flanke am Parameter "Execute" gestartet. Abhängig von der Programmierung kann "Execute" noch während des Auftrags auf den Wert "FALSE" zurückgesetzt werden oder den Wert "TRUE" bis nach Auftreten des Fehlers beibehalten.
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Parameter "Busy" nach "FALSE" und der Parameter "Error" nach "TRUE".
④	Solange der Parameter "Execute" nach Beendigung des Auftrags den Wert "TRUE" behält, bleibt auch der Parameter "Error" auf dem Wert "TRUE".
⑤	Wenn der Parameter "Execute" schon vor Abbruch des Auftrags auf "FALSE" gesetzt wurde, zeigt der Parameter "Error" nur für einen Abarbeitungszyklus den Wert "TRUE".

Motion Control-Anweisung "MC_MoveJog"

Ein "MC_MoveJog"-Auftrag wird mit dem Setzen des Parameters "JogForward" bzw. "JogBackward" gestartet. Das Auftragsziel ist erfüllt, wenn die parametrierte Geschwindigkeit erreicht wurde und die Achse mit konstanter Geschwindigkeit verfährt. Das Erreichen und Halten der parametrierten Geschwindigkeit wird am Parameter "InVelocity" mit dem Wert "TRUE" angezeigt.

Der Auftrag ist abgeschlossen, wenn der Parameter "JogForward" bzw. "JogBackward" auf den Wert "FALSE" gesetzt wurde und die Achse den Stillstand erreicht hat.

Nachfolgend wird das Verhalten der Parameter beispielhaft in unterschiedlichen Situationen gezeigt.

Die parametrierte Geschwindigkeit wird erreicht und gehalten

Wenn der Motion Control-Auftrag bis zum Erreichen der parametrierten Geschwindigkeit ausgeführt wurde, wird dies am Parameter "InVelocity" mit dem Wert "TRUE" angezeigt.

Der Tippbetrieb wird durch den Parameter "JogForward" gesteuert	Der Tippbetrieb wird durch den Parameter "JogBackward" gesteuert.
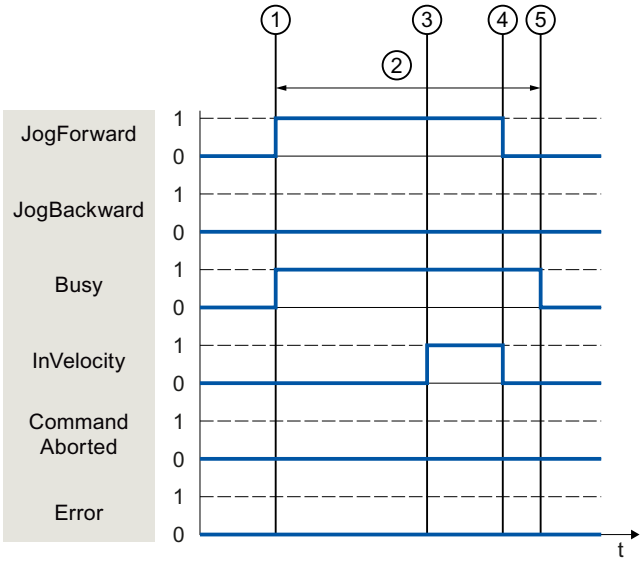	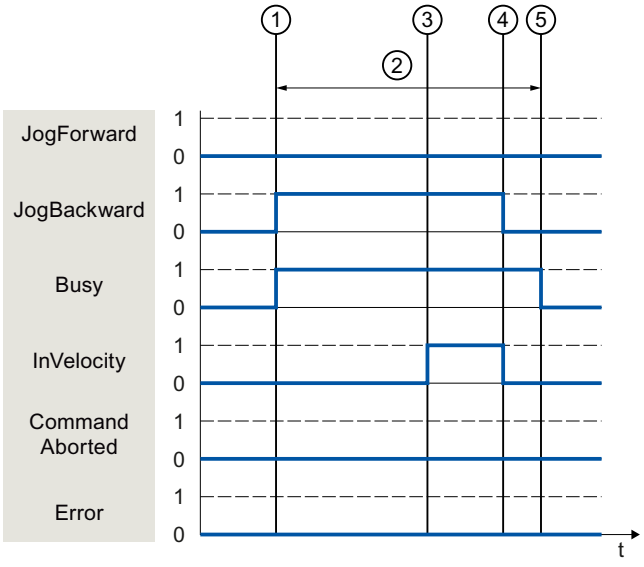

①	Der Auftrag wird mit dem Setzen des Parameters "JogForward" bzw. "JogBackward" gestartet.
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Mit dem Erreichen der parametrierten Geschwindigkeit wechselt der Parameter "InVelocity" nach "TRUE".
④	Mit dem Rücksetzen des Parameters "JogForward" bzw. "JogBackward" wird die Bewegung der Achse beendet. Die Achse bremst ab. Der Parameter "InVelocity" wechselt nach "FALSE".
⑤	Wenn die Achse zum Stillstand gekommen ist, ist der Motion Control-Auftrag abgeschlossen und der Parameter "Busy" wechselt nach "FALSE".

Der Auftrag wird während der Bearbeitung abgebrochen

Wenn der Motion Control-Auftrag während der Bearbeitung durch einen anderen Auftrag abgebrochen wird, wird dies am Parameter "CommandAborted" mit dem Wert "TRUE" angezeigt. Das Verhalten des Parameters "CommandAborted" ist unabhängig vom Erreichen der parametrierten Geschwindigkeit.

Der Tippbetrieb wird durch den Parameter "JogForward" gesteuert.	Der Tippbetrieb wird durch den Parameter "JogBackward" gesteuert.
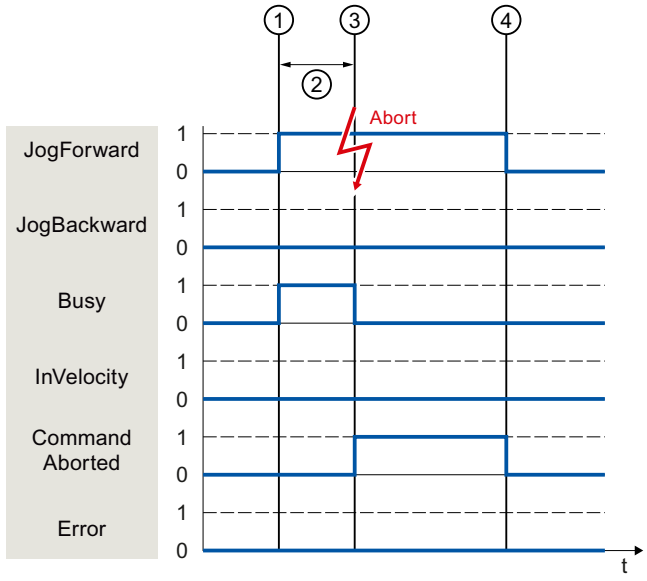	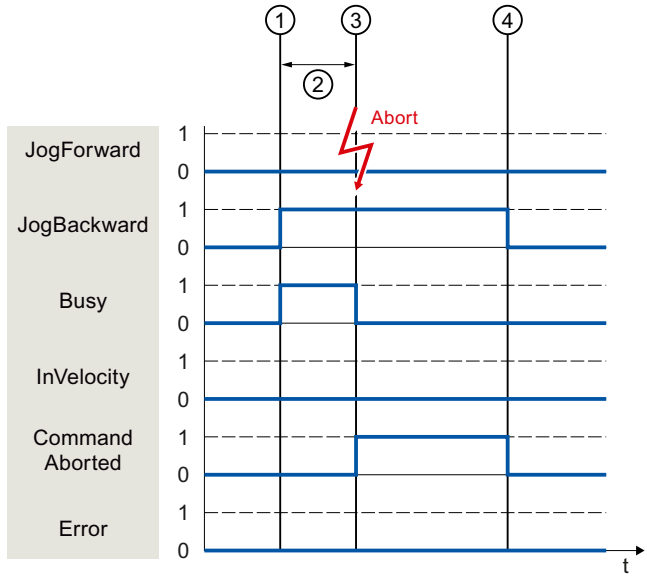

①	Der Auftrag wird mit dem Setzen des Parameters "JogForward" bzw. "JogBackward" gestartet.
②	Während der Auftrag bearbeitet, zeigt der Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Parameter "Busy" nach "FALSE" und "CommandAborted" nach "TRUE".
④	Mit dem Rücksetzen des Parameters "JogForward" bzw. "JogBackward" wechselt der Parameter "CommandAborted" ebenfalls nach "FALSE".

Während der Auftragsbearbeitung tritt ein Fehler auf

Wenn bei der Bearbeitung des Motion Control-Auftrags ein Fehler auftritt, wird dies am Parameter "Error" mit dem Wert "TRUE" angezeigt. Das Verhalten des Parameters "Error" ist unabhängig vom Erreichen der parametrierten Geschwindigkeit.

Der Tippbetrieb wird durch den Parameter "JogForward" gesteuert.	Der Tippbetrieb wird durch den Parameter "JogBackward" gesteuert.
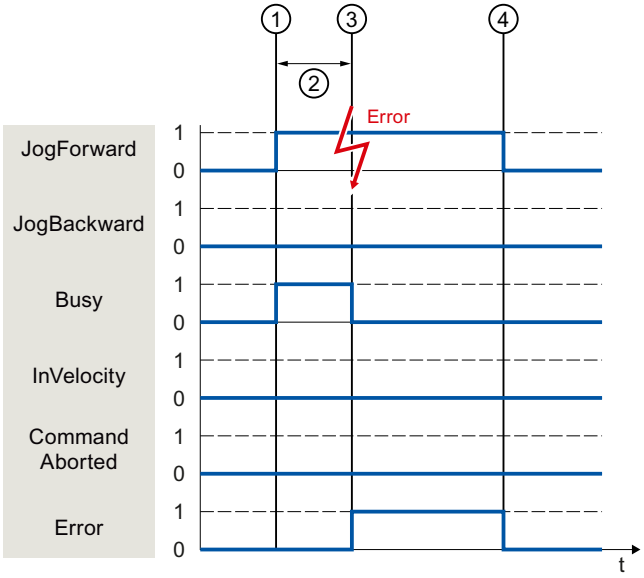	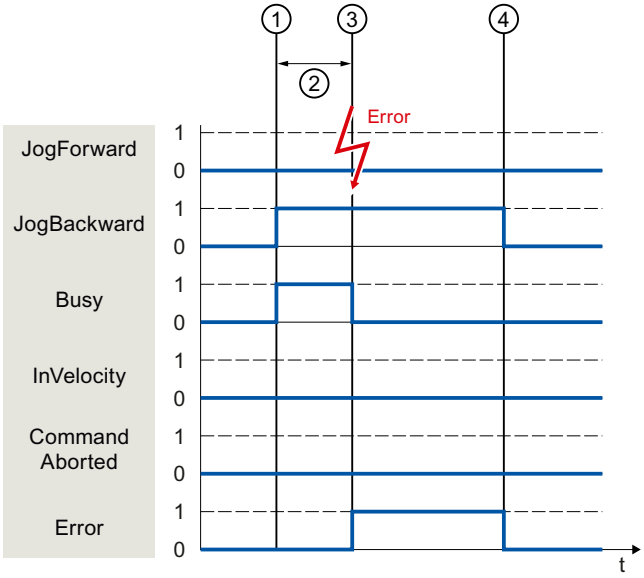

①	Der Auftrag wird mit dem Setzen des Parameters "JogForward" bzw. "JogBackward" gestartet.
②	Während der Auftrag bearbeitet wird, zeigt der Parameter "Busy" den Wert "TRUE".
③	Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Parameter "Busy" nach "FALSE" und "Error" nach "TRUE".
④	Mit dem Rücksetzen Parameters "JogForward" bzw. "JogBackward" auf den Wert "FALSE" wechselt der Parameter "Error" ebenfalls nach "FALSE".

Zusätzliche Motion Control-Anweisungen der Technologie-CPU

Die zusätzlichen Motion Control-Anweisungen der CPU S7-1500T zeigen das gleiche Parameterverhalten wie bei der CPU S7-1500.

Motion Control-Anweisungen mit Parameter "Done"

Folgende Motion Control-Anweisungen haben einen Parameter "Done":

- MC_PhasingAbsolute
- MC_PhasingRelative
- MC_InterpolateCam
- MC_GetCamFollowingValue
- MC_GetCamLeadingValue

Motion Control-Anweisungen ohne Parameter "Done"

Folgende Motion Control-Anweisungen haben einen speziellen Parameter zur Anzeige des Auftragsstatus:

- MC_GearInPos (Parameter "InSync")
- MC_CamIn (Parameter "InSync")
- MC_SynchronizedMotionSimulation (Parameter "InSimulation")
- MC_MotionInVeloctiy (Parameter "Busy")
- MC_MotionInPosition (Parameter "Busy")

Motion Control-Anweisungen

Weitere Informationen zu den Motion Control-Anweisungen des Technologieobjekts Kinematik finden Sie im Funktionshandbuch "S7-1500T Kinematikfunktionen".

Siehe auch

S7-1500T Kinematikfunktionen (https://support.industry.siemens.com/cs/ww/de/view/109749264)

Beenden von Motion Control-Aufträgen

Beim Beenden eines Auftrags wird zwischen dem fehlerfreien Abschluss des Auftrags und dem Abbruch einer Bewegung unterschieden.

Abschluss des Auftrags

Der Abschluss eines Motion Control-Auftrags wird wie im Kapitel "Verfolgung laufender Aufträge (Seite 2004)" beschrieben angezeigt.

Abbruch eines Auftrags

Der Abbruch und das Ablöseverhalten sind in Kapitel "AUTOHOTSPOT" beschrieben. Speziell wartende Aufträge können durch den "MC_Power" abgebrochen werden.

Abbruch einer Bewegung

Wenn eine Bewegung abgebrochen werden muss, können Sie folgende Maßnahmen durchführen:

- "MC_Halt" ausführen
  Um eine Bewegung abzubrechen und die Achse anzuhalten, können Sie die "MC_Halt"-Anweisung verwenden.
- "MC_Power" deaktivieren
  Im Notfall können Sie die Achse über eine Schnellhalterampe anhalten. Setzen Sie dazu den Parameter "Enable" der "MC_Power"-Anweisung auf "FALSE". Die Achse wird gemäß dem gewählten "StopMode" abgebremst und alle Aufträge an das Technologieobjekt werden abgebrochen.

Abbruch eines Messauftrags

Mit der Motion Control-Anweisung "MC_AbortMeasuringInput" wird ein aktiver einmaliger oder zyklischer Messauftrag abgebrochen.

Abbruch eines aktiven Nockens/einer aktiven Nockenspur

- "MC_OutputCam"
  Ein aktiver Nocken wird gesperrt, wenn der Parameter "Enable" der Motion Control-Anweisung "MC_OutputCam" auf "FALSE" gesetzt wird.
- "MC_CamTrack"
  Eine aktive Nockenspur wird gesperrt, wenn der Parameter "Enable" der Motion Control-Anweisung "MC_CamTrack" auf "FALSE" gesetzt wird.

Restart von Technologieobjekten

Systemseitig werden die Technologieobjekte automatisch nach dem Einschalten der CPU bzw. nach dem Laden in die CPU mit den Startwerten aus dem Technologie-Datenbaustein initialisiert. Wenn bei erneutem Laden in die CPU Restart-relevante Änderungen festgestellt werden, wird automatisch ein Restart des Technologieobjekts durchgeführt.

Wenn Restart-relevante Daten im RUN durch das Anwenderprogramm geändert wurden, muss das Technologieobjekt zur Übernahme der Änderungen anwenderseitig neu initialisiert werden. Beim Betriebszustandsübergang RUN → STOP führt die CPU automatisch einen Restart von Technologieobjekten mit Restart-relevanten Änderungen durch.

Wenn Änderungen im Technologie-Datenbaustein auch nach dem Restart des Technologieobjekts erhalten bleiben sollen, müssen Sie die Änderungen mit der erweiterten Anweisung "WRIT_DBL" auf den Startwert im Ladespeicher schreiben.

Restart erforderlich

Wenn ein Restart des Technologieobjekts erforderlich ist, wird dies unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Achse bzw. Status Geber > Online-Startwert geändert" und in der Variable "<TO>.StatusWord.X3 (OnlineStartValuesChanged)" des Technologieobjekts angezeigt.

Restart eines Technologieobjekts

Ein Restart des Technologieobjekts wird anwenderseitig durch die Motion Control-Anweisung "MC_Reset" mit Parameter "Restart" = TRUE ausgelöst.

Bei einem Restart werden alle Konfigurationsdaten des Technologieobjekts vom Ladespeicher in den Arbeitsspeicher geladen. Dabei werden die Aktualwerte im Technologie-Datenbaustein überschrieben.

Beachten Sie bei einem Restart des Technologieobjekts folgende Hinweise:

- Durch einen Restart wird der Status "Referenziert" eines Technologieobjekts mit inkrementellen Istwerten zurückgesetzt (<TO>.StatusWord.X5 (HomingDone)).
- Während ein Restart ausgeführt wird, kann das Technologieobjekt keine Aufträge ausführen. Ein aktiver Restart wird unter "Technologieobjekt > Diagnose > Status- und Fehlerbits > Status Achse bzw. Status Geber > Restart aktiv" und in der Variable "<TO>.StatusWord.X2 (RestartActive)" des Technologieobjekts angezeigt.
- Motion Control-Aufträge werden während eines Restarts mit Parameter "Error" = TRUE und "ErrorID" = 16#800D (Auftrag nicht ausführbar, da Restart aktiv) abgelehnt.
- Während ein Restart ausgeführt wird, können Sie nicht auf den Technologie-Datenbaustein zugreifen.

Siehe auch

Restart-relevante Daten ändern (Seite 1992)

6.1.2.6 Laden in CPU

Beim Laden in die CPU S7-1500 wird immer sichergestellt, dass Projektdaten nach dem Laden online und offline konsistent sind.

Die Daten der Technologieobjekte werden in Technologie-Datenbausteinen gespeichert. Zum Laden neuer oder geänderter Technologieobjekte gelten somit die Bedingungen zum Laden von Bausteinen.

Laden im Betriebszustand RUN

Beim Laden im Betriebszustand RUN der CPU wird geprüft, ob ein Laden ohne Restart des Technologieobjekts möglich ist.

Wenn Restart-relevante Konfigurationswerte geändert wurden, wird nach dem Laden in die CPU automatisch ein Restart des Technologieobjekts durchgeführt.

Das Laden eines Technologieobjekts ist nur möglich, wenn das Technologieobjekt gesperrt ist.

Folgende Änderungen können Sie **nicht** im Betriebszustand RUN in die CPU laden:

- Änderungen der MC-Servo Zeittakte
- Änderungen an der Hardware-Schnittstelle des Technologieobjekts unter "Technologieobjekt > Konfiguration > Hardware-Schnittstelle"

6.1.2.7 Inbetriebnahme

Der nachfolgende Leitfaden beschreibt die Schritte, die Sie bei der Inbetriebnahme Ihrer Motion Control-spezifischen Anlagenteile beachten sollten.

Die Inbetriebnahme anderer Teile Ihres Automatisierungssystems ist abhängig von der jeweiligen Anlagenkonfiguration. Die Inbetriebnahme (nicht Motion Control) ist im Systemhandbuch Automatisierungssystem S7-1500 (http://support.automation.siemens.com/WW/view/de/59191792) beschrieben.

Leitfaden zur Inbetriebnahme

Dieser Leitfaden dient als Empfehlung für die Inbetriebnahme einer Anlage mit Motion Control. Die Vorgehensweise wird am Beispiel eines Technologieobjekts Positionierachse beschrieben.

Voraussetzung

- Die Konfiguration folgender Bestandteile ist abgeschlossen:
  - CPU
  - BUS-Kommunikation
  - Antriebe
  - Technologieobjekte
- Das Anwenderprogramm ist erstellt.
- Die Verdrahtung der CPU und der zugehörigen Peripherie ist abgeschlossen.
- Die Inbetriebnahme und Optimierung des Antriebs ist abgeschlossen.

Vorgehensweise

Gehen Sie zur Inbetriebnahme Ihrer Motion Control-spezifischen Anlagenteile folgendermaßen vor:

Schritt	Durchzuführende Aktion	Unterstützt durch TIA Portal
CPU einschalten	Schalten Sie die Spannungsversorgung und die CPU ein.	-
Lageregler "deaktivieren"	Stellen Sie die Verstärkung des Lagereglerkreises (Kv-Faktor) auf null. (Dies dient zur Vermeidung ungewollter Antriebsbewegungen durch evtl. Parametrierfehler im Lageregelkreis.)	"Technologieobjekt > Konfiguration > Erweiterte Parameter > Regelkreis"
Vorsteuerung aktivieren	Stellen Sie die Vorsteuerung auf 100 %.	"Technologieobjekt > Konfiguration > Erweiterte Parameter > Regelkreis"
Projekt in die CPU laden	Bringen Sie die CPU in den Betriebszustand STOP. Laden Sie Ihr Projekt in die CPU (Hardware und Software laden).	• "Funktionsleiste > CPU stoppen" • "Funktionsleiste > Laden in Gerät"

Schritt	Durchzuführende Aktion	Unterstützt durch TIA Portal
Onlineverbindung zur CPU herstellen	Aktivieren Sie unter "Online & Diagnose > Online-Zugänge" das Optionskästchen "Meldungen empfangen". Stellen Sie die Schnittstelle des TIA Portals ein und stellen Sie eine Onlineverbindung mit der CPU her.	• Gerätekonfiguration • "Online & Diagnose > Online-Zugänge"
Motion Control-spezifisches Anwenderprogramm deaktivieren	Um Konflikte mit der Achssteuertafel zu vermeiden, verriegeln Sie die Freigabe der Technologieobjekte in Ihrem Anwenderprogramm (MC_Power.Enable = FALSE).	• PLC-Programmierung • Motion Control-Anweisungen
Anstehende Meldungen auswerten	Werten Sie die Meldungsanzeige im Inspektorfenster aus. Beheben Sie die Ursachen anstehender Technologie-Alarme. Quittieren Sie die Technologie-Alarme (Seite 2033).	"Inspektorfenster > Diagnose > Meldungsanzeige"
Hardware-Endschalter prüfen	Betätigen Sie die Hardware-Endschalter. Prüfen Sie die korrekte Meldungsanzeige (Technologie-Alarm 531). Quittieren Sie den Technologie-Alarm.	"Inspektorfenster > Diagnose > Meldungsanzeige"
Anbindung und Konfiguration des Antriebs prüfen (Sollwert)	Bringen Sie die CPU in den Betriebszustand RUN. Öffnen Sie die Achssteuertafel (Seite 2020) und übernehmen Sie die Steuerungshoheit. Führen Sie folgende Schritte aus: • Geben Sie das Technologieobjekt frei. ⇒ Der Antrieb muss sich einschalten und ggf. die Bremse lösen. Die Position wird gehalten. • Verfahren Sie die Achse im Tippbetrieb mit kleiner Geschwindigkeit in positiver Richtung. ⇒ Der Antrieb muss sich bewegen. Der Positionsistwert muss steigen (positive Richtung). • Sperren Sie das Technologieobjekt. ⇒ Der Antrieb muss sich abschalten und ggf. die Bremse schließen.	"Technologieobjekt > Inbetriebnahme > Achssteuertafel"
Anbindung und Konfiguration des Gebers prüfen (Istwert)	Prüfen Sie die Skalierung der Istwerte (Drehrichtung, Wegbewertung, und Auflösung des Gebers). ⇒ Die tatsächliche mechanische Positionsänderung muss mit der Änderung der Istwerte übereinstimmen.	• "Technologieobjekt > Diagnose > PROFIdrive-Telegramm" • "Technologieobjekt > Inbetriebnahme > Achssteuertafel"
Bezugsdrehzahl prüfen	Verfahren Sie die Achse im Tippbetrieb mit kleiner Geschwindigkeit in positiver Richtung. ⇒ Die angezeigte aktuelle Geschwindigkeit muss mit der Sollgeschwindigkeit übereinstimmen. Wenn die angezeigte aktuelle Geschwindigkeit stark von der Sollgeschwindigkeit abweicht, passen Sie die Bezugsdrehzahl an.	• "Technologieobjekt >Hardware-Schnittstelle > Datenaustausch" • "Technologieobjekt > Inbetriebnahme > Achssteuertafel"
Lageregler optimieren	Ermitteln Sie die optimale Verstärkung des Lagereglerkreises (Kv) mit der Inbetriebnahmefunktion Optimierung (Seite 2027). Passen Sie dafür bei Bedarf die Schleppfehlergrenzen an.	"Technologieobjekt > Inbetriebnahme > Optimierung"
Verstärkung Kv in das Projekt übernehmen.	Tragen Sie die mit der Optimierungsfunktion ermittelte Verstärkung Kv in Ihre Konfiguration ein. Laden Sie Ihr Projekt in die CPU.	"Technologieobjekt > Konfiguration > Erweiterte Parameter > Regelkreis"

Schritt	Durchzuführende Aktion	Unterstützt durch TIA Portal
Motion Control-spezifisches Anwenderprogramm aktivieren	Heben Sie die Verriegelung der Freigabe der Technologieobjekte in Ihrem Anwenderprogramm auf (MC_Power.Enable = TRUE).	• PLC-Programmierung • Motion Control-Anweisungen
Funktion des Anwenderprogramms prüfen	Prüfen Sie die programmierten Funktionen Ihres Anwenderprogramms.	• Beobachtungs- und Forcetabellen • Online- und Diagnosefunktionen
Ende der Inbetriebnahme für ein Technologieobjekt Positionierachse	Führen Sie zur Inbetriebnahme weiterer Technologieobjekte die entsprechenden Schritte erneut aus.	Siehe oben.

Achssteuertafel

Funktion und Aufbau der Achssteuertafel

Mit der Achssteuertafel verfahren Sie einzelne Achsen. Für den Betrieb der Achssteuertafel ist kein Anwenderprogramm notwendig. Mit der Achssteuertafel übernehmen Sie die Steuerungshoheit für ein Technologieobjekt und steuern die Bewegungen der Achse.

⚠ WARNUNG

Unkontrollierte Achsbewegungen

Beim Betrieb mit der Achssteuertafel kann die Achse unkontrollierte Bewegungen ausführen (z. B. wegen fehlerhafter Konfiguration des Antriebs oder des Technologieobjekts). Weiterhin wird beim Verfahren einer Leitachse mit der Achssteuertafel eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren.

Führen Sie daher vor dem Betrieb mit der Achssteuertafel folgende Schutzmaßnahmen durch:

- Stellen Sie sicher, dass sich der NOT-AUS-Schalter in Reichweite des Bedieners befindet.
- Aktivieren Sie die Hardware-Endschalter.
- Aktivieren Sie die Software-Endschalter.
- Stellen Sie sicher, dass die Schleppfehlerüberwachung aktiviert ist.
- Stellen Sie sicher, dass keine Folgeachse mit der zu verfahrenden Achse gekoppelt ist.

In der Projektnavigation finden Sie die Achssteuertafel der Technologieobjekte Drehzahlachse, Positionierachse und Gleichlaufachse unter "Technologieobjekt > Inbetriebnahme".

Die Achssteuertafel ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Betriebsart
- Steuern

- Status Achse
- Istwerte

Elemente der Achssteuertafel

Die folgende Tabelle zeigt die Elemente der Achssteuertafel:

Bereich	Element	Beschreibung
Steuerungshoheit		Im Bereich "Steuerungshoheit" übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben Sie an Ihr Anwenderprogramm zurück.
	Schaltfläche "Holen"	Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. • Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein. • Beim Verfahren einer Leitachse mit der Achssteuertafel wird eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren. • Wenn die Onlineverbindung zur CPU während des Betriebs mit der Achssteuertafel ausfällt, wird die Achse nach Ablauf der Lebenszeichenüberwachung mit maximaler Verzögerung angehalten. In diesem Fall wird eine Fehlermeldung angezeigt ("ErrorID" = 16#8013) und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. • Wenn während des Betriebs mit der Achssteuertafel ein Dialog, z. B. "Speichern unter", die Achssteuertafel überdeckt, wird die Achse mit maximaler Verzögerung angehalten und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. Wenn während des Betriebs mit der Achssteuertafel die Schaltfläche "Stopp", z. B. durch Scrollen oder durch ein anderes Fenster, überdeckt wird, bleibt die Steuerungshoheit bestehen, aber die Achse wird mit maximaler Verzögerung angehalten. Wenn Sie während des Betriebs mit der Achssteuertafel innerhalb des TIA Portals in ein anderes Fenster, z. B. in die Projektnavigation, wechseln, bleibt die Steuerungshoheit und Bewegung der Achse bestehen, sofern die Achssteuertafel im TIA Portal eingebettet ist. Wenn die Achssteuertafel vom TIA Portal abgelöst ist und Sie innerhalb des TIA Portals in ein anderes Fenster, z. B. in die Projektnavigation, wechseln, bleibt die Steuerungshoheit bestehen, aber die Achse wird mit maximaler Verzögerung angehalten. Wenn Sie während des Betriebs mit der Achssteuertafel außerhalb des TIA Portals in ein anderes Fenster wechseln, bleibt die Steuerungshoheit bestehen, aber die Achse wird mit maximaler Verzögerung angehalten. • Wenn Sie auf die Schaltfläche "Holen" klicken, wird ein Warnhinweis angezeigt. Im Warnhinweis können Sie die Lebenszeichenüberwachung anpassen (100 bis 60000 ms). Wenn die Steuerungshoheit der Achssteuertafel wiederholt ohne direkte Fehlermeldung verloren geht, könnte die Onlineverbindung zur CPU durch eine zu hohe Kommunikationslast beeinträchtigt sein. In diesem Fall wird im Archiv der Meldungsanzeige die Meldung "Fehler bei Inbetriebnahme. Ausfall des Lebenszeichens zwischen Steuerung und TIA Portal" angezeigt. Um diesen Fehler zu beheben, passen Sie die Lebenszeichenüberwachung im Warnhinweis an. • Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler ("ErrorID" = 16#8012: Achssteuertafel aktiviert) abgewiesen. • Mit der Übernahme der Steuerungshoheit wird die Konfiguration des Technologieobjekts übernommen. Änderungen an der Konfiguration des Technologieobjekts werden erst wirksam, nachdem die Steuerungshoheit

Bereich	Element	Beschreibung
		zurückgegeben wurde. Nehmen Sie daher gegebenenfalls erforderliche Änderungen vor der Übernahme der Steuerungshoheit vor. • Wenn die Steuerungshoheit für das Technologieobjekt übernommen wurde, ist die Achssteuertafel für den Zugriff durch eine andere Instanz des TIA Portals gesperrt (Team Engineering ab CPU V1.5).
	Schaltfläche "Abgeben"	Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.
Achse		Im Bereich "Achse" geben Sie das Technologieobjekt frei oder sperren dieses.
	Schaltfläche "Freigeben"	Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
	Schaltfläche "Sperren"	Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.
Betriebsart		In der Klappliste "Betriebsart" wählen Sie die gewünschte Betriebsart der Achssteuertafel aus.
Steuern		Im Bereich "Steuern" werden die Parameter für das Verfahren mit der Achssteuertafel entsprechend der ausgewählten Betriebsart angezeigt.
	Position	Position, auf welche die Achse referenziert wird. (Nur Betriebsarten "Referenzieren" und "Referenzpunkt setzen")
	Weg	Wegstrecke, um welche die Achse verfahren wird. (Nur Betriebsart "Positionieren relativ")
	Zielposition	Position, auf welche die Achse verfahren wird. (Nur Betriebsart "Positionieren absolut")
	Geschwindigkeit/ Sollgeschwindigkeit	Geschwindigkeit bzw. Drehzahl, mit der die Achse verfahren wird. Vorbelegung: 10 % des Standardwerts (Nur Betriebsarten "Geschwindigkeits-/Drehzahlvorgabe", "Tippen" und "Positionieren")
	Beschleunigung	Beschleunigung, mit der die Achse verfahren wird. Vorbelegung: 10 % des Standardwerts
	Verzögerung	Verzögerung, mit der die Achse verfahren wird. Vorbelegung: 10 % des Standardwerts
	Ruck	Ruck, mit dem die Achse verfahren wird. Vorbelegung: 100 % des Standardwerts
	Schaltfläche "Start"	Mit der Schaltfläche "Start" stoßen Sie einen Auftrag gemäß der ausgewählten Betriebsart an.
	Schaltfläche "Vorwärts"	Mit der Schaltfläche "Vorwärts" stoßen Sie eine Bewegung gemäß der ausgewählten Betriebsart in positive Richtung an.
	Schaltfläche "Rückwärts"	Mit der Schaltfläche "Rückwärts" stoßen Sie eine Bewegung gemäß der ausgewählten Betriebsart in negative Richtung an.
	Schaltfläche "Stopp"	Mit der Schaltfläche "Stopp" brechen Sie einen Auftrag ab bzw. halten Sie die Achse an. Wenn während des Betriebs mit der Achssteuertafel die Schaltfläche "Stopp", z. B. durch Scrollen oder durch ein anderes Fenster, überdeckt wird, bleibt die Steuerungshoheit bestehen, aber die Achse wird mit maximaler Verzögerung angehalten.
Status Achse		Im Bereich "Status Achse" wird der Status der Achse und der Status des Antriebs angezeigt.

Bereich	Element	Beschreibung
	Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen.
	Fehler	Am Technologieobjekt ist ein Fehler aufgetreten.
	Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden.
	Referenziert	Das Technologieobjekt ist referenziert.
	Mehr	Über den Link "Mehr" gelangen Sie zum Fenster "Technologieobjekt > Diagnose > Status- und Fehlerbits".
	Anstehender Fehler	Im Textfeld "Anstehender Fehler" wird der zuletzt aufgetretene Fehler angezeigt.
	Schaltfläche "Quittieren"	Mit der Schaltfläche "Quittieren" quittieren Sie anstehende Fehler.
	Meldungsanzeige	Über den Link "Meldungsanzeige" gelangen Sie zur Meldungsanzeige im Inspektorfenster.
Istwerte		Im Bereich "Aktuelle Werte" werden die Istwerte der Achse angezeigt.
	Position	Istposition der Achse
	Geschwindigkeit	Istgeschwindigkeit der Achse

Hinweis

Keine Übernahme der Parameter

Die eingestellten Parameterwerte werden mit der Rückgabe der Steuerungshoheit verworfen. Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration.

Wenn Sie während des Betriebs mit der Achssteuertafel Konfigurationswerte geändert haben, haben diese Änderungen keine Auswirkung auf den Betrieb der Achssteuertafel.

Betriebsart

Die folgende Tabelle zeigt die Betriebsarten der Achssteuertafel:

Betriebsart	Beschreibung
Referenzieren	Diese Funktion entspricht dem aktiven Referenzieren. Die Parameter für das Referenzieren (Seite 1729) müssen konfiguriert sein. Das Referenzieren ist mit einem Absolutwertgeber nicht möglich. Das Technologieobjekt wird bei Verwendung dieser Betriebsart mit einem Absolutwertgeber nicht referenziert.
Referenzpunkt setzen	Diese Funktion entspricht dem direkten Referenzieren (absolut). Mit der Schaltfläche "Start" setzen Sie die Istposition auf den in "Position" angegebenen Wert und der Status "Referenziert" wird gesetzt.
Tippen	Die Bewegungsbefehle erfolgen durch Tippen. Mit der Schaltfläche "Vorwärts" bzw. "Rückwärts" stoßen Sie Bewegung in positive bzw. negative Richtung an. Die Bewegung läuft, solange Sie die linke Maustaste gedrückt halten.
Geschwindigkeits-/ Drehzahlvorgabe	Die Achse wird so lange mit der angegebenen Geschwindigkeit bzw. Drehzahl verfahren, bis Sie die Bewegung stoppen. Die Bewegungsbefehle werden entsprechend den unter "Steuerung" parametrierten Sollwerten ausgeführt.

Betriebsart	Beschreibung
Relatives Positionieren	Das Positionieren wird als geregelte, relative Verfahrbewegung entsprechend den unter "Steuerung" parametrierten Vorgaben ausgeführt.
Absolutes Positionieren	Das Positionieren wird als geregelte, absolute Verfahrbewegung entsprechend den unter "Steuerung" parametrierten Vorgaben ausgeführt. Wenn Sie die Einstellung "Modulo" des Technologieobjekts aktiviert haben, werden im Bereich "Steuerung" die Schaltflächen "Vorwärts" und "Rückwärts" angezeigt. Die Achse wird innerhalb des Modulobereichs positioniert. Positionsvorgaben außerhalb des Modulobereichs werden entsprechend auf den Modulobereich umgerechnet. Wenn Sie die Einstellung "Modulo" des Technologieobjekts nicht aktiviert haben, wird im Bereich "Steuerung" nur die Schaltfläche "Start" angezeigt. Sie können direkt die eingegebene Position anfahren.

Achssteuertafel einsetzen

Mit der Achssteuertafel verfahren Sie einzelne Achsen. Sie übernehmen die Steuerungshoheit für ein Technologieobjekt und steuern die Bewegungen der Achse.

Voraussetzung

- Das Projekt ist erstellt und in die CPU geladen.
- Die CPU ist im Betriebszustand RUN.
- Das Technologieobjekt ist über Ihr Anwenderprogramm gesperrt ("MC_Power.Enable" = FALSE).
- Die Achssteuertafel für das Technologieobjekt wird nicht von einer anderen Instanz des TIA Portals verwendet (Team Engineering ab CPU V1.5).

Vorgehen

Gehen Sie zum Steuern der Achse mit der Achssteuertafel folgendermaßen vor:

1. Um die Steuerungshoheit für das Technologieobjekt zu übernehmen und eine Onlineverbindung zur CPU aufzubauen, klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Holen".
   Ein Warnhinweis wird angezeigt.
2. Passen Sie gegebenenfalls die Lebenszeichenüberwachung an und klicken Sie auf die Schaltfläche "OK".
3. Um das Technologieobjekt freizugeben, klicken Sie im Bereich "Achse" auf die Schaltfläche "Freigeben".
4. Wählen Sie in der Klappliste im Bereich "Betriebsart" die gewünschte Funktion der Achssteuertafel aus.
5. Geben Sie im Bereich "Steuern" die entsprechenden Parameterwerte für Ihren Auftrag an.
6. Klicken Sie abhängig von der eingestellten Betriebsart auf die Schaltfläche "Start", "Vorwärts" oder "Rückwärts", um den Auftrag zu starten.
7. Um den Auftrag zu stoppen, klicken Sie auf die Schaltfläche "Stopp".
8. Wiederholen Sie für weitere Aufträge die Schritte 4 bis 6.

9. Um das Technologieobjekt zu sperren, klicken Sie im Bereich "Achse" auf die Schaltfläche "Sperren".
10. Um die Steuerungshoheit an Ihr Anwenderprogramm zurückzugeben, klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Abgeben".

Optimierung

Funktion und Aufbau der Optimierung

Die Funktion "Optimierung" unterstützt Sie bei der Ermittlung der optimalen Vorsteuerung und Verstärkung (Kv-Faktor) für die Lageregelung der Achse. Hierzu wird der Geschwindigkeitsverlauf der Achse während einer vorgebbaren Positionierbewegung mit der Trace-Funktion aufgezeichnet. Anschließend können Sie die Aufzeichnung auswerten und die Vorsteuerung und die Verstärkung entsprechend anpassen.

⚠ WARNUNG

Unkontrollierte Achsbewegungen

Beim Betrieb mit der Optimierung kann die Achse unkontrollierte Bewegungen ausführen (z. B. wegen fehlerhafter Konfiguration des Antriebs oder des Technologieobjekts). Weiterhin wird beim Verfahren einer Leitachse eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren.

Führen Sie daher vor dem Betrieb mit der Optimierung folgende Schutzmaßnahmen durch:

- Stellen Sie sicher, dass sich der NOT-AUS-Schalter in Reichweite des Bedieners befindet.
- Aktivieren Sie die Hardware-Endschalter.
- Aktivieren Sie die Software-Endschalter.
- Stellen Sie sicher, dass die Schleppfehlerüberwachung aktiviert ist.
- Stellen Sie sicher, dass keine Folgeachse mit der zu verfahrenden Achse gekoppelt ist.

In der Projektnavigation finden Sie die Funktion "Optimierung" für die Technologieobjekte Positionierachse und Gleichlaufachse unter "Technologieobjekt > Inbetriebnahme".

Der Dialog "Optimierung" ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Messkonfiguration
- Lageregler optimieren
- Messung durchführen
- Trace

Die folgende Tabelle zeigt die Elemente der Optimierung:

Bereich	Element	Beschreibung
Steuerungshoheit		Im Bereich "Steuerungshoheit" übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben sie an Ihr Anwenderprogramm zurück.
	Schaltfläche "Holen"	Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. • Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein. • Beim Verfahren einer Leitachse mit der Achssteuertafel wird eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren. • Wenn die Onlineverbindung zur CPU während des Betriebs mit der Achssteuertafel ausfällt, wird die Achse nach Ablauf der Lebenszeichenüberwachung mit maximaler Verzögerung angehalten. In diesem Fall wird eine Fehlermeldung angezeigt ("ErrorID" = 16#8013) und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. • Wenn Sie auf die Schaltfläche "Holen" klicken, wird ein Warnhinweis angezeigt. Hier können Sie die Lebenszeichenüberwachung anpassen (100 bis 60000 ms). Wenn die Steuerungshoheit der Achssteuertafel wiederholt ohne direkte Fehlermeldung verloren geht, könnte die Onlineverbindung zur CPU durch eine zu hohe Kommunikationslast beeinträchtigt sein. In diesem Fall wird im Archiv der Meldungsanzeige die Meldung "Fehler bei Inbetriebnahme. Ausfall des Lebenszeichens zwischen Steuerung und TIA Portal" angezeigt. Um diesen Fehler zu beheben, passen Sie die Lebenszeichenüberwachung im Warnhinweis an. • Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler ("ErrorID" = 16#8012: Achssteuertafel aktiviert) abgewiesen. • Mit der Übernahme der Steuerungshoheit wird die Konfiguration des Technologieobjekts übernommen. Änderungen an der Konfiguration des Technologieobjekts werden erst wirksam, nach dem die Steuerungshoheit zurückgegeben wurde. Nehmen Sie daher eventuell erforderliche Änderungen vor der Übernahme der Steuerungshoheit vor. • Wenn die Steuerungshoheit für das Technologieobjekt übernommen wurde, ist die Achssteuertafel für den Zugriff durch eine andere Installation des TIA Portals gesperrt (Team Engineering ab CPU V1.5).
	Schaltfläche "Abgeben"	Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.
Achse		Im Bereich "Achse" geben Sie das Technologieobjekt frei oder sperren dieses.
	Schaltfläche "Freigeben"	Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
	Schaltfläche "Sperren"	Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.

Bereich	Element	Beschreibung
Messkonfiguration		Im Bereich "Messkonfiguration" konfigurieren Sie die Einstellungen für einen Testschritt.
	Weg	Wegstrecke für einen Testschritt
	Messdauer	Zeitdauer für einen Testschritt
	Optionskästchen "Dynamik anpassen"	Mit dem Optionskästchen "Dynamik anpassen" wird Ihnen die Anpassung der Beschleunigung, der Verzögerung und der maximalen Geschwindigkeit ermöglicht. Solange der Arbeitsbereich "Inbetriebnahme" geöffnet ist, werden bei erneuter Aktivierung des Optionskästchens die zuvor eingestellten Werte angezeigt.
	Beschleunigung	Voreinstellung der Beschleunigung für einen Testschritt
	Verzögerung	Voreinstellung der Verzögerung für einen Testschritt
	Maximale Geschwindigkeit	Voreinstellung der maximalen Geschwindigkeit für einen Testschritt
Lageregler optimieren		Im Bereich "Lageregler optimieren" nehmen Sie die Einstellungen zum Optimieren der Regelparameter vor. Über das Symbol neben einem Feld öffnen Sie eine Werteliste. Die Werteliste beinhaltet folgende Werte des jeweiligen Parameters: • Online-Aktualwert (Istwert) • Online-Startwert • Startwert Projekt Tragen Sie im Eingabefeld für den Istwert den neuen Wert ein. Der neue Wert wird durch Klicken auf die Schaltfläche "Vorwärts" bzw. "Rückwärts" im Bereich "Messung durchführen" übernommen.
	Vorsteuerung	Aktuelle prozentuale Geschwindigkeitsvorsteuerung des Lagereglers
	Drehzahl-Regelkreis-Ersatzzeit	Aktuelle Drehzahl-Regelkreis-Ersatzzeit Bei einer Geschwindigkeitsvorsteuerung kann über die Drehzahl-Regelkreis-Ersatzzeit ein vereinfachtes Modell des Drehzahlregelkreises gebildet werden. Dadurch wird ein Übersteuern der Geschwindigkeitsstellgröße durch den Lageregler in den Beschleunigungs- und Verzögerungsphasen verhindert. Dazu wird der Positionssollwert des Lagereglers um die Drehzahl-Regelkreis-Ersatzzeit im Bezug zur Geschwindigkeitsvorsteuerung verzögert.
	Verstärkung	Aktuelle Verstärkung des Lagereglers (Kv)
Messung durchführen		Im Bereich "Messung durchführen" führen Sie die Testschritte aus.
	Schaltfläche "Vorwärts"	Mit der Schaltfläche "Vorwärts" starten Sie einen Testschritt zur Optimierung in positive Richtung.
	Schaltfläche "Rückwärts"	Mit der Schaltfläche "Rückwärts" starten Sie einen Testschritt zur Optimierung in negative Richtung.
	Schaltfläche "Stopp"	Mit der Schaltfläche "Stopp" beenden Sie einen Testschritt. Die Achse wird mit der konfigurierten maximalen Verzögerung abgebremst. Wenn ein anderes Fenster die Schaltfläche "Stopp" überlagert, wird die Achse angehalten.
Trace		Im unteren Bereich des Dialogs "Optimierung" wird die Trace-Funktion angezeigt. Mit jedem Testschritt wird automatisch eine Trace-Aufzeichnung der benötigten Parameter gestartet und nach dem Beenden des Testschritts angezeigt. Nach der Rückgabe der Steuerungshoheit wird die Trace-Aufzeichnung gelöscht. Eine vollständige Beschreibung der Trace-Funktion finden Sie im Kapitel "Trace- und Logikanalysatorfunktion nutzen".

Hinweis

Keine automatische Übernahme der Parameter in das Technologieobjekt

Die eingestellten Parameterwerte werden nach der Rückgabe der Steuerungshoheit verworfen.

Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration. Die Werte der Verstärkung, der Vorsteuerung und der Drehzahl-Regelkreis-Ersatzzeit können Sie über den Wert "Startwert Projekt" in Ihre Konfiguration übernehmen.

Siehe auch

Regelung (Seite 1753)

Lageregler optimieren

Voraussetzung

- Die CPU ist im Betriebszustand RUN.
- Das Projekt ist erstellt und in die CPU geladen.
- Das Technologieobjekt ist über Ihr Anwenderprogramm gesperrt (MC_Power.Enable = FALSE).
- Die Achssteuertafel für das Technologieobjekt wird nicht von einer anderen Installation des TIA Portals verwendet (Team Engineering ab CPU V1.5).

Verstärkung des Lagereglers (Kv) optimieren

Gehen Sie zum Optimieren der Verstärkung (Kv) folgendermaßen vor:

1. Klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Holen", um die Steuerungshoheit für das Technologieobjekt zu holen und eine Onlineverbindung zur CPU aufzubauen.
   Ein Warnhinweis wird angezeigt.
2. Passen Sie gegebenenfalls die Lebenszeichenüberwachung an und bestätigen Sie mit "OK".
3. Klicken Sie im Bereich "Achse" auf die Schaltfläche "Freigeben", um das Technologieobjekt freizugeben.
4. Konfigurieren Sie gegebenenfalls im Bereich "Messkonfiguration" die Werte für Weg, Dauer und Dynamik eines Testschritts.
5. Konfigurieren Sie gegebenenfalls im Bereich "Lageregler konfigurieren" die Werte für die Vorsteuerung und die Drehzahl-Regelkreis-Ersatzzeit.
6. Tragen Sie einen Startwert für die Verstärkung ein. Starten Sie mit einem niedrigen Wert.

7. Klicken Sie auf die Schaltfläche "Vorwärts" bzw. "Rückwärts", um einen Testschritt für die Optimierung in positive bzw. negative Richtung zu starten.
   Für die angegebene Dauer wird ein Sollwert entsprechend des angegebenen Wegs ausgegeben. Die Achse verfährt um die angegebene Wegstrecke. Eine Trace-Aufzeichnung der Bewegung (Soll- und Istwerte) wird automatisch angelegt.
   Eine vollständige Beschreibung der Trace-Funktion finden Sie im Kapitel "Trace- und Logikanalysatorfunktion nutzen".

   Hinweis

   Schleppfehlergrenzen anpassen

   Wenn bei der Optimierung wiederholt Fehlermeldungen durch die Schleppfehlerüberwachung angezeigt werden, passen Sie zeitweise die Schleppfehlergrenzen an.

8. Werten Sie den Kurvenverlauf der Trace-Aufzeichnung aus. Passen Sie die Verstärkung schrittweise an. Klicken Sie nach jeder Werteingabe auf die Schaltfläche "Vorwärts" bzw. "Rückwärts". Damit übernehmen Sie den Wert und starten jeweils eine neue Bewegung und Trace-Aufzeichnung.

Achten Sie bei der Anpassung der Verstärkung auf folgende Eigenschaften des Kurvenverlaufs:

- Die Kurve zeigt eine kurze Ausregelungszeit.
- Die Kurve weist keine Bewegungsumkehr der Istposition auf.
- Beim Anfahren der Sollposition tritt kein Überschwingen auf.
- Der Kurvenverlauf zeigt ein stabiles Gesamtverhalten (schwingungsfreier Kurvenverlauf).

Die folgende Trace-Aufzeichnung zeigt einen Kurvenverlauf mit einer langen Ausregelungszeit:

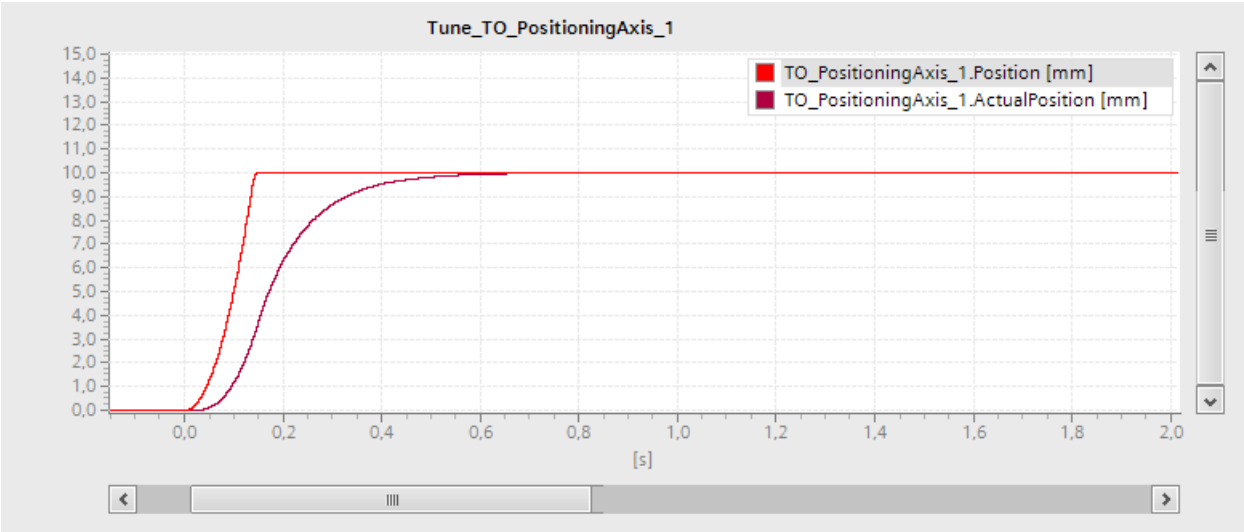

Die folgende Trace-Aufzeichnung zeigt einen Kurvenverlauf mit einem Überschwingen beim Anfahren des Sollwerts:

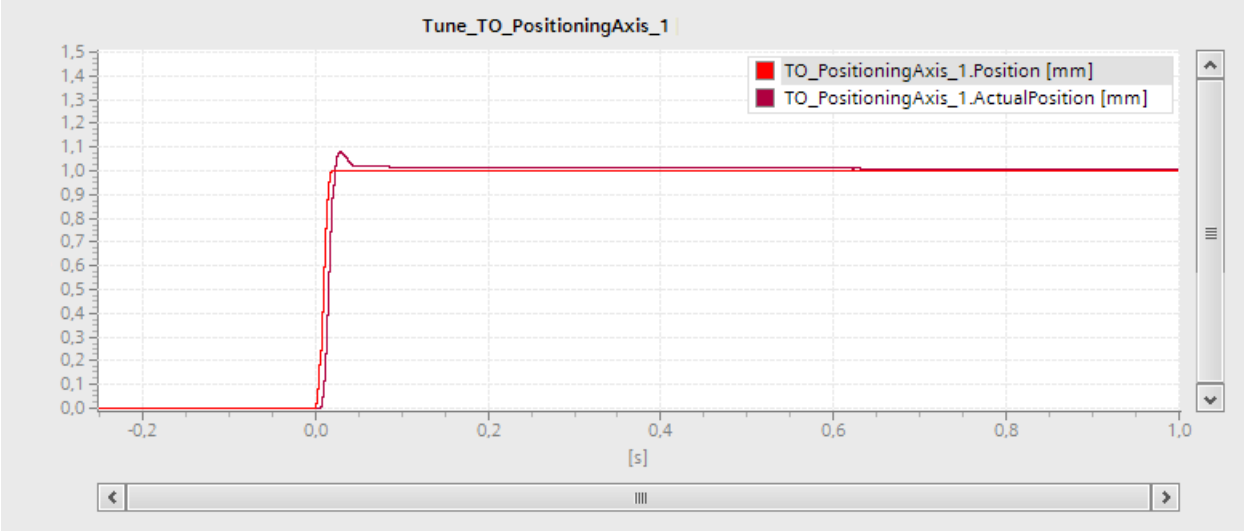

Die folgende Trace-Aufzeichnung zeigt einen Kurvenverlauf mit einer optimierten Verstärkung und stabilem Gesamtverhalten:

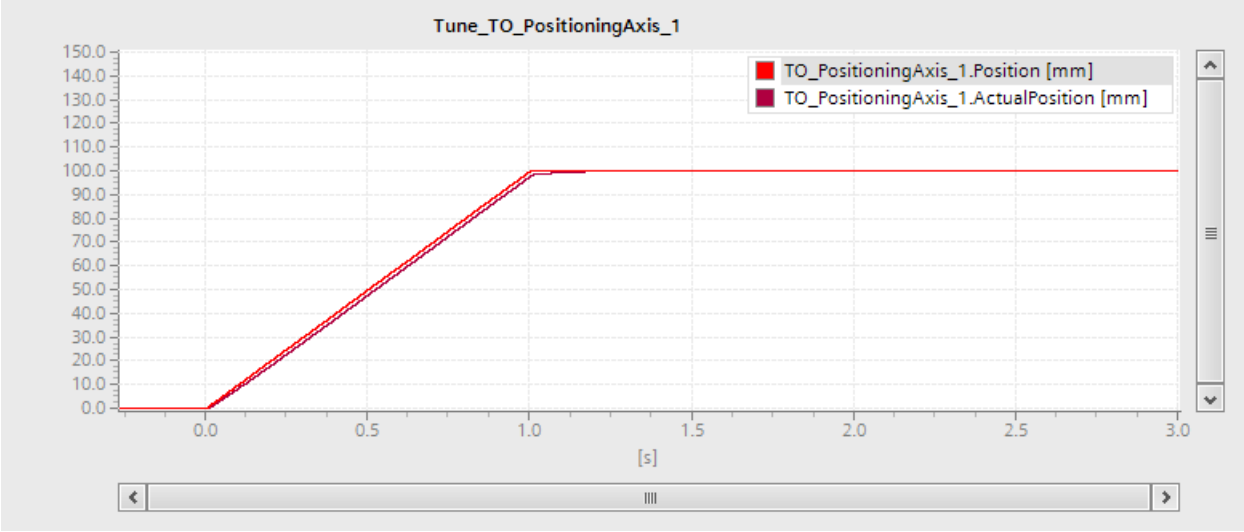

Parameterwerte des Lagereglers ins Projekt übernehmen

Um die ermittelten Parameterwerte des Lagereglers in Ihr Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol neben dem Feld des jeweiligen Parameters.
   Eine Werteliste wird geöffnet.
2. Tragen Sie den ermittelten Wert in das Feld "Startwert Projekt" der Werteliste ein.
3. Klicken Sie im Bereich "Achse" auf die Schaltfläche "Sperren", um das Technologieobjekt zu sperren.

4. Klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Abgeben", um die Steuerungshoheit an Ihr Anwenderprogramm zurückzugeben.
5. Laden Sie Ihr Projekt in die CPU.

6.1.2.8 Diagnose

Das Kapitel "Diagnose" beschränkt sich auf die Beschreibung des Diagnosekonzepts für Motion Control und die Beschreibung der Diagnosesicht der einzelnen Technologieobjekte im TIA Portal.

Eine umfassende Beschreibung der Systemdiagnose der CPU S7-1500 finden Sie im Funktionshandbuch "Diagnose" (http://support.automation.siemens.com/WW/view/de/59192926).

Diagnosekonzept

Das Diagnosekonzept umfasst Alarme und dazugehörige Meldungen sowie Fehlermeldungen an den Motion Control-Anweisungen. Zusätzlich unterstützt Sie das TIA Portal durch Konsistenzprüfungen bereits bei der Konfiguration der Technologieobjekte und der Erstellung Ihres Anwenderprogramms.

Alle Alarme im Betrieb (von CPU, Technologie, Hardware usw.) werden im Inspektorfenster des TIA Portals angezeigt. Diagnoseinformationen, die sich auf Technologieobjekte beziehen (Technologie-Alarme, Statusinformationen), werden zusätzlich im Diagnosefenster des jeweiligen Technologieobjekts angezeigt.

Wenn bei der Bewegungsführung ein Fehler an einem Technologieobjekt auftritt (z. B. Anfahren eines Hardware-Endschalters), wird ein Technologie-Alarm (Seite 2033) ausgelöst und eine entsprechende Meldung im TIA Portal sowie an HMI-Geräten angezeigt.

In Ihrem Anwenderprogramm werden Technologie-Alarme grundsätzlich über Fehlerbits im Technologie-Datenbaustein gemeldet. Zusätzlich wird die Nummer des Technologie-Alarms mit der höchsten Priorität angezeigt. Um die Fehlerauswertung zu vereinfachen, wird zusätzlich über die Parameter "Error" und "ErrorID" der Motion Control-Anweisungen angezeigt, dass ein Technologie-Alarm ansteht.

Programmfehler (Seite 2037) können bei der Parametrierung oder bei der Bearbeitungsreihenfolge der Motion Control-Anweisungen auftreten (z. B. unzulässige Parameterangabe beim Aufruf der Anweisung, Anstoß eines Auftrags ohne Freigabe mit "MC_Power"). Fehler an Motion Control-Anweisungen werden beim Aufruf der Anweisungen über die Parameter "Error" und "ErrorID" angezeigt.

Technologie-Alarme

Wenn am Technologieobjekt ein Fehler auftritt (z. B. Anfahren eines Hardware-Endschalters), wird ein Technologie-Alarm ausgelöst und angezeigt. Die Auswirkungen eines Technologie-Alarms auf das Technologieobjekt sind durch die Alarmreaktion festgelegt.

Alarmklassen

Technologie-Alarme sind in drei Klassen eingeteilt:

- **Quittierbare Warnung**
  Die Bearbeitung der Motion Control-Aufträge wird fortgesetzt. Die laufende Bewegung der Achse kann beeinflusst werden, z. B. durch Begrenzung der aktuellen Dynamikwerte auf die konfigurierten Grenzwerte.
- **Quittierpflichtiger Alarm**
  Bewegungsaufträge werden gemäß der Alarmreaktion abgebrochen. Um die Bearbeitung neuer Aufträge nach dem Beheben der Fehlerursache fortzusetzen, müssen Sie Alarme quittieren.
- **Schwerwiegender Fehler**
  Bewegungsaufträge werden gemäß der Alarmreaktion abgebrochen.
  Um das Technologieobjekt nach dem Beheben der Fehlerursache erneut einsetzen zu können, müssen Sie einen Restart des Technologieobjekts (Seite 2016) durchführen.

Anzeige von Technologie-Alarmen

Ein Technologie-Alarm wird an folgenden Stellen angezeigt:

- TIA Portal
  - "Technologieobjekt > Diagnose > Status- und Fehlerbits"
    Anzeige anstehender Technologie-Alarme je Technologieobjekt.
  - "Technologieobjekt > Inbetriebnahme > Achssteuertafel"
    Anzeige des letzten anstehenden Technologie-Alarms je Technologieobjekt.
  - "Inspektorfenster > Diagnose > Meldungsanzeige"
    Um Technologie-Alarme über die Meldungsanzeige anzuzeigen, aktivieren Sie unter "Online & Diagnose > Online-Zugänge" das Optionskästchen "Meldungen empfangen". Bei Online-Verbindung zur CPU werden die anstehenden Technologie-Alarme aller Technologieobjekte angezeigt. Zusätzlich steht Ihnen die Archivansicht zur Verfügung. Die Meldungsanzeige kann auch an einem angeschlossenen HMI aktiviert und angezeigt werden.
  - "CPU > Online & Diagnose"
    Anzeige der im Diagnosepuffer der CPU eingetragenen Technologie-Alarme.
- Anwenderprogramm
  - Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction"
    Anzeige der Nummer und der Reaktion des Technologie-Alarms mit der höchsten Priorität.
  - Variable "<TO>.StatusWord"
    Über das Bit 1 (Error) wird angezeigt, dass ein Technologie-Alarm ansteht.
  - Variable "<TO>.ErrorWord"
    Anzeige von Alarmen und schwerwiegenden Fehlern.
  - Variable "<TO>.WarningWord"
    Anzeige von Warnungen.
  - Parameter "Error" und "ErrorID"
    An einer Motion Control-Anweisung wird mit den Parametern "Error" = TRUE und "ErrorID" = 16#8001 angezeigt, dass ein Technologie-Alarm ansteht.
- Display der CPU
  Um Technologie-Alarme auf dem Display der CPU anzuzeigen, nehmen Sie beim Laden in die CPU folgende Einstellung vor:
  Im Dialog "Vorschau Laden" wählen Sie für den Eintrag "Textbibliotheken" die Aktion "Konsistentes Laden" aus.

Alarmreaktion

Ein Technologie-Alarm beinhaltet immer eine Alarmreaktion, welche die Auswirkung auf das Technologieobjekt beschreibt. Die Alarmreaktion ist systemseitig vorgegeben.

Folgende Tabelle zeigt mögliche Alarmreaktionen:

Alarmreaktion	Beschreibung
Achsen (Drehzahlachse, Positionierachse, Gleichlaufachse)	
Keine Reaktion (nur Warnungen) <TO>.ErrorDetail.Reaction = 0	Die Bearbeitung der Motion Control-Aufträge wird fortgesetzt. Die laufende Bewegung der Achse kann beeinflusst werden, z. B. durch Begrenzung der aktuellen Dynamikwerte auf die konfigurierten Grenzwerte.
Stopp mit aktuellen Dynamikwerten <TO>.ErrorDetail.Reaction = 1	Laufende Bewegungsbefehle werden abgebrochen. Die Achse wird mit den an der Motion Control-Anweisung anstehenden Dynamikwerten abgebremst und zum Stillstand gebracht.
Stopp mit maximalen Dynamikwerten <TO>.ErrorDetail.Reaction = 2	Laufende Bewegungsbefehle werden abgebrochen. Die Achse wird mit den unter "Technologieobjekt > Erweiterte Parameter > Dynamikgrenzen" konfigurierten maximalen Dynamikwerten abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt.
Stopp mit Notstopp-Rampe <TO>.ErrorDetail.Reaction = 3	Laufende Bewegungsbefehle werden abgebrochen. Die Achse wird mit der unter "Technologieobjekt > Erweiterte Parameter > Notstopp-Rampe" konfigurierten Notstopp-Verzögerung ohne Ruckbegrenzung abgebremst und zum Stillstand gebracht.
Freigabe wegnehmen <TO>.ErrorDetail.Reaction = 4	Laufende Bewegungsbefehle werden abgebrochen. Der Sollwert null wird ausgegeben und die Freigabe weggenommen. Die Achse wird abhängig von der Konfiguration im Antrieb abgebremst und zum Stillstand gebracht.
Sollwerte nachführen <TO>.ErrorDetail.Reaction = 5	Laufende Bewegungsbefehle werden abgebrochen. Der Sollwert null wird ausgegeben. Die vom Antrieb gelieferten Istwerte werden automatisch als Sollwerte nachgeführt.
Andere Technologieobjekte (Nocken, Nockenspur, Messtaster, Kurvenscheibe, Externer Geber)	
Keine Reaktion (nur Warnungen) <TO>.ErrorDetail.Reaction = 0	Die Bearbeitung der Motion Control-Aufträge wird fortgesetzt. Die laufende Bewegung der Achse kann beeinflusst werden, z. B. durch Begrenzung der aktuellen Dynamikwerte auf die konfigurierten Grenzwerte.
Bearbeitung des Technologieobjekts beenden: • Nocken <TO>.ErrorDetail.Reaction = 6 • Nockenspur <TO>.ErrorDetail.Reaction = 7 • Messtaster <TO>.ErrorDetail.Reaction = 8 • Kurvenscheibe <TO>.ErrorDetail.Reaction = 9 • Externer Geber <TO>.ErrorDetail.Reaction = 10	Die Bearbeitung des Technologieobjekts wird beendet. Alle laufenden Motion Control-Aufträge werden abgebrochen.

Technologie-Alarme quittieren

Sie können Technologie-Alarme folgendermaßen quittieren:

- TIA Portal
  - "Technologieobjekt > Inbetriebnahme > Achssteuertafel"
    Klicken Sie auf die Schaltfläche "Quittieren", um alle anstehenden Alarme und Warnungen für das ausgewählte Technologieobjekt zu quittieren.
  - "Inspektorfenster > Diagnose > Meldungsanzeige"
    Sie können die Alarme und Warnungen aller Technologieobjekte einzeln oder gesamt quittieren.
- HMI
  An einem HMI können Sie bei aktivierter Meldungsanzeige die Alarme und Warnungen aller Technologieobjekte einzeln oder gesamt quittieren.
- Anwenderprogramm
  Quittieren Sie an einem Technologieobjekt anstehende Technologie-Alarme mit der Motion Control-Anweisung "MC_Reset".

Weitere Informationen

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Fehler an Motion Control-Anweisungen

Fehler an Motion Control-Anweisungen (z. B. Angabe eines ungültigen Parameterwerts) werden durch die Ausgangsparameter "Error" und "ErrorID" angezeigt.

Unter folgenden Bedingungen wird an der Motion Control-Anweisung "Error" = TRUE und "ErrorID" = 16#8xxx angezeigt:

- Unzulässiger Status des Technologieobjekts, der die Ausführung des Auftrags verhindert.
- Unzulässige Parametrierung der Motion Control-Anweisung, die die Ausführung des Auftrags verhindert.
- Infolge der Alarmreaktion eines Fehlers am Technologieobjekt.

Fehleranzeige

Wenn ein Fehler an einer Motion Control-Anweisung ansteht, zeigt der Parameter "Error" den Wert "TRUE". Die Fehlerursache können Sie dem Wert am Parameter "ErrorID" entnehmen.

Aufträge an das Technologieobjekt werden bei "Error" = TRUE abgelehnt. Laufende Aufträge werden durch abgelehnte Aufträge nicht beeinflusst.

Wenn während der Auftragsbearbeitung "Error" = TRUE und "ErrorID" = 16#8001 angezeigt wird, ist ein Technologie-Alarm aufgetreten. Werten Sie in diesem Fall die Anzeige des Technologie-Alarms aus.

Wenn während der Auftragsbearbeitung eines "MC_MoveJog"-Auftrags "Error" = TRUE angezeigt wird, wird die Achse abgebremst und zum Stillstand gebracht. In diesem Fall ist die Verzögerung wirksam, die an der "MC_MoveJog"-Anweisung parametriert wurde.

Fehler quittieren

Das Quittieren von Fehlern an Motion Control-Anweisungen ist nicht erforderlich.

Starten Sie einen Auftrag nach der Behebung des Fehlers erneut.

Weitere Informationen

Eine Liste der ErrorIDs finden Sie im Anhang "Fehlerkennung (Seite 2208)".

Technologieobjekt Drehzahlachse

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Achse

Die folgende Tabelle zeigt mögliche Zustände der Achse:

Status	Beschreibung
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben.
Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden. (<TO>.StatusWord.X0 (Enable))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Achssteuertafel aktiv	Die Achssteuertafel ist aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt. Die Achse kann nicht vom Anwenderprogramm gesteuert werden. (<TO>.StatusWord.X4 (ControlPanelActive))
Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen. (<TO>.StatusDrive.InOperation)
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Achsbewegung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Bewegungsauftrag aktiv. (<TO>.StatusWord.X6 (Done))
Tippen	Die Achse wird mit einem Auftrag zum Tippbetrieb der Motion Control-Anweisung "MC_MoveJog" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X9 (JogCommand))
Drehzahlvorgabe	Die Achse wird mit einem Auftrag mit Drehzahlvorgabe der Motion Control-Anweisung "MC_MoveVelocity" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X10 (VelocityCommand))
Konstante Drehzahl	Die Achse wird mit konstanter Drehzahl verfahren oder befindet sich im Stillstand. (<TO>.StatusWord.X12 (ConstantVelocity))
Beschleunigen	Die Achse wird beschleunigt. (<TO>.StatusWord.X13 (Accelerating))
Verzögern	Die Achse wird abgebremst. (<TO>.StatusWord.X14 (Decelerating))
Momentenbegrenzung aktiv	An der Achse wirkt der konfigurierte Grenzwert für die Kraft/das Moment. (<TO>.StatusWord.X27 (InLimitation))

Warnungen

Die folgende Tabelle zeigt mögliche Warnungen:

Warnung	Beschreibung
Konfiguration	Einer oder mehrere Konfigurationsparameter werden zeitweise intern angepasst. (<TO>.WarningWord.X1 (ConfigWarning))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind. (<TO>.WarningWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.WarningWord.X6 (DynamicError))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Antrieb	Ein Fehler im Antrieb ist aufgetreten. (<TO>.ErrorWord.X4 (DriveFault))
Datenaustausch	Die Kommunikation mit einem verbundenen Gerät ist gestört. (<TO>.ErrorWord.X7 (CommunicationFault))
Peripherie	Ein Fehler beim Zugriff auf eine logische Adresse ist aufgetreten. (<TO>.ErrorWord.X13 (PeripheralError))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. Technologieobjekt nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.ErrorWord.X6 (DynamicError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Drehzahlachse) (Seite 2094)

Variable "ErrorWord" (Drehzahlachse) (Seite 2097)

Variable "WarningWord" (Drehzahlachse) (Seite 2099)

Status Bewegung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal den Bewegungsstatus der Achse. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Sollwerte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Solldrehzahl	Solldrehzahl der Achse (<TO>.Velocity)
Drehzahl-Override	Prozentuale Korrektur der Drehzahlvorgabe Die von Motion Control-Anweisungen oder von der Achssteuertafel vorgegebene Solldrehzahl wird durch einen Override überlagert und prozentual angepasst. Als Drehzahlkorrektur sind Werte von 0.0 % bis 200.0 % zulässig. (<TO>.Override.Velocity)

Bereich "Aktuelle Werte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Istdrehzahl	Istdrehzahl der Achse (<TO>.ActualSpeed)

Bereich "Dynamikgrenzen"

In diesem Bereich werden die konfigurierten Grenzwerte der Dynamikparameter angezeigt.

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Drehzahl	Konfigurierte maximale Drehzahl (<TO>.DynamicLimits.MaxVelocity)
Beschleunigung	Konfigurierte maximale Beschleunigung (<TO>.DynamicLimits.MaxAcceleration)
Verzögerung	Konfigurierte maximale Verzögerung (<TO>.DynamicLimits.MaxDeceleration)
Ruck	Konfigurierter maximaler Ruck (<TO>.DynamicLimits.MaxJerk)

PROFIdrive-Telegramm

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > PROFIdrive-Telegramm" überwachen Sie im TIA Portal das PROFIdrive-Telegramm vom Antrieb zur Steuerung. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Antrieb"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Antrieb zur Steuerung angezeigt:

- Die Zustandsworte "ZSW1" und "ZSW2"
- Die an den Antrieb ausgegebene Solldrehzahl (NSOLL)
- Die vom Antrieb gemeldete Istdrehzahl (NIST)

Technologieobjekt Positionierachse/Gleichlaufachse

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Achse

Die folgende Tabelle zeigt mögliche Zustände der Achse:

Status	Beschreibung
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben. (<TO>.StatusWord.X25 (AxisSimulation))
Freigegeben	Das Technologieobjekt ist freigegeben. Sie können die Achse mit Bewegungsaufträgen verfahren. (<TO>.StatusWord.X0 (Enable))
Lagegeregelter Betrieb	Die Achse befindet sich im lagegeregelten Betrieb. (Invertierung von <TO>.StatusWord.X28 (NonPositionControlled))
Referenziert	Das Technologieobjekt ist referenziert. Der Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung wurde erfolgreich hergestellt. (<TO>.StatusWord.X5 (HomingDone))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Achssteuertafel aktiv	Die Achssteuertafel ist aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt. Sie können die Achse nicht vom Anwenderprogramm aus steuern. (<TO>.StatusWord.X4 (ControlPanelActive))

Status	Beschreibung
Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen. (<TO>.StatusDrive.InOperation)
Geberistwerte gültig	Die Geberistwerte sind gültig. (<TO>.StatusSensor[n].State)
Aktiver Geber	Aktueller Geber an der Achse (<TO>.OperativeSensor)
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Endschalter

Die folgende Tabelle zeigt mögliche Aktivierungen der Software- und Hardware-Endschalter:

Status	Beschreibung
Negativer SW-Endschalter angefahren	Der negative Software-Endschalter wurde angefahren. (<TO>.StatusWord.X15 (SWLimitMinActive))
Positiver SW-Endschalter angefahren	Der positive Software-Endschalter wurde angefahren. (<TO>.StatusWord.X16 (SWLimitMaxActive))
Negativer HW-Endschalter angefahren	Der negative Hardware-Endschalter wurde angefahren oder überfahren. (<TO>.StatusWord.X17 (HWLimitMinActive))
Positiver HW-Endschalter angefahren	Der positive Hardware-Endschalter wurde angefahren oder überfahren. (<TO>.StatusWord.X18 (HWLimitMaxActive))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Achsbewegung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Auftrag aktiv. (<TO>.StatusWord.X6 (Done))
Referenzierauftrag	Das Technologieobjekt führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus. (<TO>.StatusWord.X11 (HomingCommand))
Tippen	Die Achse wird mit einem Auftrag zum Tippbetrieb der Motion Control-Anweisung "MC_MoveJog" verfahren. (<TO>.StatusWord.X9 (JogCommand))
Geschwindigkeitsvorgabe	Die Achse wird mit einem Auftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisung "MC_MoveVelocity" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X10 (VelocityCommand))
Positionierauftrag	Die Achse wird mit einem Positionierauftrag der Motion Control-Anweisung "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X8 (PositioningCommand))

Status	Beschreibung
Konstante Geschwindigkeit	Die Achse wird mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand. (<TO>.StatusWord.X12 (ConstantVelocity))
Stillstand	Die Achse befindet sich im Stillstand. (<TO>.StatusWord.X7 (StandStill))
Beschleunigen	Die Achse wird beschleunigt. (<TO>.StatusWord.X13 (Accelerating))
Verzögern	Die Achse wird abgebremst. (<TO>.StatusWord.X14 (Decelerating))
Momentenbegrenzung aktiv	An der Achse wirkt der konfigurierte Grenzwert für die Kraft/das Moment. (<TO>.StatusWord.X27 (InLimitation))
Synchronisierung	Nur Gleichlaufachse Die Achse wird auf den Leitwert einer Leitachse aufsynchronisiert. (<TO>.StatusWord.X21 (Synchronizing))
Gleichlauf	Nur Gleichlaufachse Die Achse ist aufsynchronisiert und fährt synchron zur Leitachse. (<TO>.StatusWord.X22 (Synchronous))

Warnungen

Die folgende Tabelle zeigt mögliche Warnungen:

Warnung	Beschreibung
Konfiguration	Einer oder mehrere Konfigurationsparameter werden zeitweise intern angepasst. (<TO>.WarningWord.X1 (ConfigWarning))
Auftrag abgewiesen	Der Auftrag ist nicht ausführbar. Sie können keine Motion Control-Anweisung ausführen, weil notwendige Voraussetzungen nicht erfüllt sind. (<TO>.WarningWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.WarningWord.X6 (DynamicError))
Synchronisierung	Nur Gleichlaufachse Ein Fehler beim Gleichlauf ist aufgetreten. Die an der entsprechenden Motion Control-Anweisung angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert. (<TO>.WarningWord.X14 (SynchronousError))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Antrieb	Ein Fehler im Antrieb ist aufgetreten. (<TO>.ErrorWord.X4 (DriveFault))
Geber	Ein Fehler im Gebersystem ist aufgetreten. (<TO>.ErrorWord.X5 (SensorFault))
Datenaustausch	Die Kommunikation mit einem verbundenen Gerät ist gestört. (<TO>.ErrorWord.X7 (CommunicationFault))
Peripherie	Ein Fehler beim Zugriff auf eine logische Adresse ist aufgetreten. (<TO>.ErrorWord.X13 (PeripheralError))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Sie können keine Motion Control-Anweisung ausführen, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. Technologieobjekt nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Referenzieren	Ein Fehler bei einem Referenziervorgang ist aufgetreten. (<TO>.ErrorWord.X10 HomingFault))
Positionieren	Die Achse wurde am Ende einer Positionierbewegung nicht korrekt positioniert. (<TO>.ErrorWord.X12 (PositioningFault))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.ErrorWord.X6 (DynamicError))
Schleppfehler	Der maximale zulässige Schleppfehler wurde überschritten. (<TO>.ErrorWord.X11 (FollowingErrorFault))
SW-Endschalter	Ein Software-Endschalter wurde erreicht. (<TO>.ErrorWord.X8 (SwLimit))
HW-Endschalter	Ein Hardware-Endschalter wurde erreicht oder überfahren. (<TO>.ErrorWord.X9 (HWLimit))

Fehler	Beschreibung
Adaption	Ein Fehler bei der Datenadaption ist aufgetreten. (<TO>.ErrorWord.X15 (AdaptionError))
Synchronisierung	Nur Gleichlaufachse Ein Fehler beim Gleichlauf ist aufgetreten. Die an der entsprechenden Motion Control-Anweisung angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert. (<TO>.ErrorWord.X14 (SynchronousError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Positionierachse/Gleichlaufachse) (Seite 2132)

Variable "ErrorWord" (Positionierachse/Gleichlaufachse) (Seite 2137)

Variable "WarningWord" (Positionierachse/Gleichlaufachse) (Seite 2140)

Zusätzliche Anzeigen für die Technologie-CPU (Seite 2048)

Status Bewegung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal den Bewegungsstatus der Achse. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Sollwerte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Zielposition	Aktuelle Zielposition eines aktiven Positionierauftrags Der Wert der Zielposition ist nur während der Ausführung eines Positionierauftrags gültig. (<TO>.StatusPositioning.TargetPosition)
Sollposition	Sollposition der Achse (<TO>.Position)

Status	Beschreibung
Sollgeschwindigkeit	Sollgeschwindigkeit der Achse (<TO>.Velocity)
Geschwindigkeits-Override	Prozentuale Korrektur der Geschwindigkeitsvorgabe Die von Motion Control-Anweisungen oder von der Achssteuertafel vorgegebene Sollgeschwindigkeit wird durch einen Override überlagert und prozentual angepasst. Als Geschwindigkeitskorrektur sind Werte von 0.0 % bis 200.0 % zulässig. (<TO>.Override.Velocity)

Bereich "Aktuelle Werte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Operativer Geber	Operativ wirksamer Geber der Achse
Istposition	Istposition der Achse Wenn das Technologieobjekt nicht referenziert ist, wird der Wert relativ zu der Position angezeigt, die zum Zeitpunkt der Freigabe des Technologieobjekts vorlag. (<TO>.ActualPosition)
Istgeschwindigkeit	Istgeschwindigkeit der Achse (<TO>.ActualVelocity)
Schleppfehler	Schleppfehler der Achse (<TO>.StatusPositioning.FollowingError)

Bereich "Dynamikgrenzen"

In diesem Bereich werden die konfigurierten Grenzwerte der Dynamikparameter angezeigt.

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Geschwindigkeit	Konfigurierte maximale Geschwindigkeit (<TO>.DynamicLimits.MaxVelocity)
Beschleunigung	Konfigurierte maximale Beschleunigung (<TO>.DynamicLimits.MaxAcceleration)
Verzögerung	Konfigurierte maximale Verzögerung (<TO>.DynamicLimits.MaxDeceleration)
Ruck	Konfigurierter maximaler Ruck (<TO>.DynamicLimits.Jerk)

Siehe auch

Zusätzliche Anzeigen für die Technologie-CPU (Seite 2048)

PROFIdrive-Telegramm

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > PROFIdrive-Telegramm" überwachen Sie im TIA Portal die PROFIdrive-Telegramme von Antrieb und Geber. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Antrieb"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Antrieb zur Steuerung angezeigt:

- Die Zustandsworte "ZSW1" und "ZSW2"
- Die an den Antrieb ausgegebene Solldrehzahl (NSOLL)
- Die vom Antrieb gemeldete Istdrehzahl (NIST)

Bereich "Geber"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Geber zur Steuerung angezeigt:

- Das Zustandswort "Gx_ZSW"
- Der Positionsistwert "Gx_XIST1" (zyklischer Geberistwert)
- Der Positionsistwert "Gx_XIST2" (Absolutwert des Gebers)

Siehe auch

Zusätzliche Anzeigen für die Technologie-CPU (Seite 2048)

Zusätzliche Anzeigen für die Technologie-CPU

Die Diagnosefunktionen enthalten für die Technologie-CPU S7-1500T erweiterte/zusätzliche Anzeigen

Status- und Fehlerbits

Die folgende Tabelle zeigt die erweiterten/zusätzlichen Anzeigen der Status- und Fehlerbits (Seite 2042):

Status Achse	
Geberistwerte gültig	Die Geberistwerte (Geber 1, Geber 2, Geber 3 oder Geber 4) sind gültig. (<TO>.StatusSensor[n].State)
Aktiver Geber	Geber 1, Geber 2, Geber 3 oder Geber 4 ist der operativ wirksame Geber. (<TO>.OperativeSensor)

Status Bewegung

Die folgende Tabelle zeigt die erweiterten/zusätzlichen Anzeigen für den Status der Bewegung (Seite 2046):

Aktuelle Werte	Beschreibung
Operativer Geber	Operativ wirksamer Geber der Achse

PROFIdrive-Telegramm

In den Bereichen "Geber 1" bis "Geber 4" werden folgende Parameter aus dem PROFIdrive-Telegramm (Seite 2048) des entsprechenden Gebers zur Steuerung angezeigt:

- Das Zustandswort "Gx_ZSW"
- Der Positionsistwert "Gx_XIST1" (zyklischer Geberistwert)
- Der Positionsistwert "Gx_XIST2" (Absolutwert des Gebers)

Siehe auch

Status- und Fehlerbits (Seite 2042)

Status Bewegung (Seite 2046)

PROFIdrive-Telegramm (Seite 2048)

Technologieobjekt Externer Geber

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Geberstatus

Die folgende Tabelle zeigt mögliche Zustände des Externen Gebers:

Status	Beschreibung
Geber freigegeben	Das Technologieobjekt ist freigegeben. (<TO>.StatusWord.X0 (Enable))
Referenziert	Das Technologieobjekt ist referenziert. Der Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung wurde erfolgreich hergestellt. (<TO>.StatusWord.X5 (HomingDone))

Status	Beschreibung
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Geberistwerte gültig	Die Geberistwerte sind gültig. (<TO>.StatusSensor[n].State)
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Auftragsbearbeitung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Motion Control-Auftrag aktiv. (Freigabe durch "MC_Power"-Auftrag ausgenommen) (<TO>.StatusWord.X6 (Done))
Referenzierauftrag	Das Technologieobjekt führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" aus. (<TO>.StatusWord.X11 (HomingCommand))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 UserFault))
Geber	Ein Fehler im Gebersystem ist aufgetreten. (<TO>.ErrorWord.X5 (SensorFault))

Fehler	Beschreibung
Datenaustausch	Fehlende oder fehlerhafte Kommunikation (<TO>.ErrorWord.X7 (CommunicationFault))
Adaption	Ein Fehler bei der Datenadaption ist aufgetreten. (<TO>.ErrorWord.X15 (AdaptionError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Externer Geber) (Seite 2153)

Variable "ErrorWord" (Externer Geber) (Seite 2154)

Variable "WarningWord" (Externer Geber) (Seite 2156)

Status Bewegung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal die Geberistwerte. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Aktuelle Werte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Istposition	Istposition der Achse Wenn das Technologieobjekt nicht referenziert ist, wird der Wert relativ zu der Position angezeigt, die zum Zeitpunkt der Freigabe des Technologieobjekts vorlag. (<TO>.ActualPosition)
Istgeschwindigkeit	Istgeschwindigkeit der Achse (<TO>.ActualVelocity)

PROFIdrive-Telegramm

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > PROFIdrive-Telegramm" überwachen Sie im TIA Portal das PROFIdrive-Telegramm des Gebers. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb Technologieobjekt zur Verfügung.

Bereich "Geber"

In diesem Bereich werden folgende Parameter aus dem PROFIdrive-Telegramm vom Geber zur Steuerung angezeigt:

- Das Zustandswort "G1_ZSW"
- Der Positionsistwert "G1_XIST1" (zyklischer Geberistwert)
- Der Positionsistwert "G1_XIST2" (Absolutwert des Gebers)

Technologieobjekt Messtaster

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Messtaster

Die folgende Tabelle zeigt mögliche Zustände des Messtasters:

Status	Beschreibung
Aktiv	Das Technologieobjekt ist in Betrieb. (<TO>.StatusWord.X0 (Control))
Warten auf Messereignis	Der Messtaster wartet auf ein Messereignis. Die Technologie-Datenbaustein Variable "<TO>.Status" hat den Wert 1 ("WAITING_FOR_TRIGGER").
Messwert vorhanden	Der Messtaster hat einen oder mehrere Messwerte erfasst. Die Technologie-Datenbaustein Variable "<TO>.Status" hat den Wert 2 ("TRIGGER_OCCURRED").
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert. (<TO>.StatusWord.X2 (RestartActive))
Messtaster bereit	Der Messtaster ist mit dem Messmodul synchronisiert und kann verwendet werden. (<TO>.StatusWord.X5 (CommunicationOK))
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. die dem Messtaster zugeordnete Achse ist nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Peripherie	Ein Fehler beim Zugriff auf eine logische Adresse ist aufgetreten. (<TO>.ErrorWord.X13 (PeripheralError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Messtaster) (Seite 2161)

Variable "ErrorWord" (Messtaster) (Seite 2162)

Variable "WarningWord" (Messtaster) (Seite 2164)

Technologieobjekt Nocken

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Nocken

Die folgende Tabelle zeigt mögliche Zustände des Nockens:

Status	Beschreibung
Aktiv	Das Technologieobjekt ist in Betrieb. (<TO>.StatusWord.X0 (Control))
Geschaltet	Der Nocken ist geschaltet. (<TO>.CamOutput)
Nockenausgabe invertiert	Die Nockenausgabe erfolgt invertiert. (<TO>.StatusWord.X4 (OutputInverted))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert. (<TO>.StatusWord.X2 (RestartActive))
Nockenausgabe bereit	Der Nocken ist mit dem Ausgabemodul synchronisiert und kann verwendet werden. (<TO>.StatusWord.X5 (CommunicationOk))
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. die dem Nocken zugewiesene Achse ist nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Peripherie	Ein Fehler beim Zugriff auf eine logische Adresse ist aufgetreten. (<TO>.ErrorWord.X13 (PeripheralError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Nocken) (Seite 2167)

Variable "ErrorWord" (Nocken) (Seite 2168)

Variable "WarningWord" (Nocken) (Seite 2170)

Technologieobjekt Nockenspur

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Nockenspur

Die folgende Tabelle zeigt mögliche Zustände der Nockenspur:

Status	Beschreibung
Aktiv	Das Technologieobjekt ist in Betrieb. (<TO>.StatusWord.X0 (Control))
Geschaltet	Ein Nocken der Nockenspur ist geschaltet. (<TO>.TrackOutput)
Nockenausgabe invertiert	Die Nockenausgabe erfolgt invertiert. (<TO>.StatusWord.X4 (OutputInverted))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction" des Technologieobjekts. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. Die Variablen des Technologie-Datenbausteins werden im aktiven Restart nicht aktualisiert. (<TO>.StatusWord.X2 (RestartActive))
Nockenspurausgabe bereit	Die Nockenspur ist mit dem Ausgabemodul synchronisiert und kann verwendet werden. (<TO>.StatusWord.X5 (CommunicationOk))
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. die der Nockenspur zugewiesene Achse ist nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Peripherie	Ein Fehler beim Zugriff auf eine logische Adresse ist aufgetreten. (<TO>.ErrorWord.X13 (PeripheralError))

Meldungsanzeige

Für weitere Informationen und zum Quittieren des Fehlers gelangen Sie über den Link "Meldungsanzeige" in das Inspektorfenster.

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Siehe auch

Variable "StatusWord" (Nockenspur) (Seite 2173)

Variable "ErrorWord" (Nockenspur) (Seite 2174)

Variable "WarningWord" (Nockenspur) (Seite 2176)

Status Nockenspur

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Nockenspur" überwachen Sie im TIA Portal den Status der Nockenspur. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Gültigkeit und Maskierung der Nocken"

In diesem Bereich werden die einzelnen Nocken einer Nockenspur mit dem Status zu den folgenden Eigenschaften angezeigt:

Status	Beschreibung
Gültig	Gültigkeit der einzelnen Nocken der Nockenspur (<TO>.Parameter.Cam[1..32].Existent)
Maskiert	Bitmaskierung der einzelnen Nocken der Nockenspur (<TO>.Parameter.CamMasking)
Wirksam	Eingeschalteter Nocken (Bitmaskiert) (<TO>.SingleCamState)

Bereich "Positionen"

In diesem Bereich werden die folgenden Statuswerte angezeigt:

Status	Beschreibung
Aktuelle Position in der Nockenspur	Position bei der Nockenspurbearbeitung innerhalb eines Nockenspurzyklus Der Abstand zur aktuellen Bezugsposition der aktuellen Nockenspur (<TO>.MatchPosition) wird angezeigt. (<TO>.TrackPosition)
Aktueller Nockenspur-anfang	Bezugsposition der aktuellen Nockenspur Bei zyklischer Bearbeitung der Nockenspur wird die fortgeführte Bezugsposition der aktuellen Nockenspur angezeigt. Die eindeutige Erfassung und Ausgabe der Position ist nur möglich, wenn das zugeordnete Technologieobjekt in Bewegung ist. (<TO>.MatchPosition)

6.1.2.9 Technologieversion V3.0

Konfigurieren

Technologieobjekt Kurvenscheibe konfigurieren

Grafischer Editor

Aufbau des grafischen Editors V3.0

Der grafische Editor ist in folgende Bereiche unterteilt:

- Werkzeugleiste
- Kurvendiagramm

Werkzeugleiste

Die Werkzeugleiste am oberen Rand des grafischen Editors stellt Ihnen über Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	Kurvenscheibe aus Datei importieren	Siehe Kapitel Kurvenscheibe importieren/exportieren (Seite 1955)
	Kurvenscheibe in Datei exportieren	Siehe Kapitel Kurvenscheibe importieren/exportieren (Seite 1955)
	Elemente bearbeiten/Ansicht verschieben	• Selektieren und Verschieben einzelner Elemente und Element • Verschieben der Ansicht per Drag & Drop Um von einem beliebigen Werkzeug auf das Werkzeug "Elemente bearbeiten/ Ansicht verschieben" umzuschalten, drücken Sie die Taste <Esc>.
	Zoomauswahl aktivieren	Zoom auf ausgewählten Bereich
	Vertikalen Zoom aktivieren	Vertikaler Zoom auf ausgewählten Bereich ohne horizontale Skalierung Alternativ: <Strg> + Ziehen mit gedrückter Maustaste auf Ordinate
	Horizontalen Zoom aktivieren	Horizontaler Zoom auf ausgewählten Bereich ohne vertikale Skalierung Alternativ: <Strg> + Ziehen mit gedrückter Maustaste auf Abszisse
	Zoom in	Vergrößern der Anzeige Alternativ: <Strg> + Mausrad aufwärts im Kurvendiagramm
	Zoom out	Verkleinern der Anzeige Alternativ: <Strg> + Mausrad abwärts im Kurvendiagramm
	Alles anzeigen	Anzeige des gesamten Definitions- und Wertebereichs
	Fangraster aktivieren	Eingaben und Elementendpunkte werden am konfigurierbaren Fangraster und an anderen Elementendpunkten ausgerichtet.
	Punkt einfügen	Hinzufügen eines Punkts zum Diagramm
	Gerade einfügen	Hinzufügen einer Gerade zum Diagramm
	Sinus einfügen	Hinzufügen eines Sinuselements zum Diagramm
	Polynom einfügen	Hinzufügen eines Polynoms zum Diagramm
	Inversen Sinus einfügen	Hinzufügen eines inversen Sinus zum Diagramm
	Ansicht: Ein Diagramm	Anzeige eines Diagramms mit folgenden Kurven der im Editor geöffneten Kurvenscheibe: • Vorgabekurve • Effektive Position • Effektive Geschwindigkeit • Effektive Beschleunigung • Effektiver Ruck

Schaltfläche	Funktion	Beschreibung
	Ansicht: Vier Diagramme	Anzeige von vier Diagrammen mit folgenden Kurven der im Editor geöffneten Kurvenscheibe: • Diagramm mit Vorgabekurve und effektiver Position • Diagramm mit effektiver Geschwindigkeit • Diagramm mit effektiver Beschleunigung • Diagramm mit effektivem Ruck
	Vertikale Messlinien	Anzeigen und Verschieben der vertikalen Messlinien Ziehen Sie mit gedrückter Maustaste einen Messbereich auf. Die vertikale Position der Messlinien lässt sich verschieben. Im Diagramm werden die Funktionswerte an den Messlinienpositionen angezeigt. Zwischen den Messlinien wird die Differenz der Messlinien angezeigt. Wenn Sie den Mauszeiger auf einer Messlinie positionieren, wird der Funktionswert angezeigt.
	Horizontale Messlinien	Anzeigen und Verschieben der horizontalen Messlinien Ziehen Sie mit gedrückter Maustaste einen Messbereich auf. Die horizontale Position der Messlinien lässt sich verschieben. Im Diagramm werden die Funktionswerte an den Messlinienpositionen angezeigt. Zwischen den Messlinien wird die Differenz der Messlinien angezeigt. Wenn Sie den Mauszeiger auf einer Messlinie positionieren, wird der Funktionswert angezeigt.
	Legende anzeigen	Ein- bzw. Ausblenden der Legende im Kurvendiagramm. Um auf der Ordinate die Werte für eine bestimmte Kurve anzuzeigen, klicken Sie in der Legende auf den Namen der entsprechenden Kurve.
	Legende links anzeigen	Anzeige der Legende auf der linken Seite des Kurvendiagramms.
	Legende rechts anzeigen	Anzeige der Legende auf der rechten Seite des Kurvendiagramms.
	Online-Kurve anzeigen	Anzeige der aus der CPU zurückgelesenen Kurvenscheibe (orange)

Kurvendiagramm

Im Kurvendiagramm geben Sie die Elemente der Kurve grafisch ein und passen die Kurve über Auswahl und Verschiebung der Elemente an.

Diagrammflächen außerhalb des in "Profil > Allgemein (Seite 1935)" konfigurierten Leitwert-/Folgewertbereichs werden ausgegraut. Elemente außerhalb werden mit Warnung („Element liegt außerhalb des Definitionsbereichs“) dargestellt.

Über die Konfiguration der grafischen Ansicht lassen sich verschiedene Kurven (Position, Geschwindigkeit, Beschleunigung und Ruck) in bis zu vier Diagrammen übereinander anzeigen. Wenn mehrere Diagramme angezeigt werden, lassen sich die Diagramme über die Trennlinien in der Höhe anpassen.

Die Ansicht lässt sich im Handmodus über <Strg> + Mausrad und <Strg> + Ziehen mit gedrückter Maustaste auf Abszisse/Ordinate zoomen.

Über Warndreiecke werden eventuelle Berechnungsprobleme gemeldet. Der Meldungstext wird über den Tooltip des Warndreiecks angezeigt.

Kontextmenü im grafischen Editor V3.0

Folgende Tabelle zeigt die Funktionen im Kontextmenü des grafischen Editors:

Funktion	Beschreibung
Alles anzeigen	Anzeige des gesamten Definitions- und Wertebereichs
Zoom in	Vergrößern der Anzeige
Zoom out	Verkleinern der Anzeige
Ausschneiden	Entfernen der ausgewählten Elemente und Kopieren in die Zwischenablage
Kopieren	Kopieren der ausgewählten Elemente in die Zwischenablage
Einfügen	Einfügen der Elemente aus der Zwischenablage nach dem letzten Element
Löschen	Löschen der ausgewählten Elemente Übergänge zu gegebenenfalls vorhandenen Elementen werden ebenfalls gelöscht.
Einfügen spezial	Aufruf des Dialogs "Elemente einfügen (Seite 1958)"
Verschieben	Aufruf des Dialogs "Elemente verschieben (Seite 1958)"
Skalieren	Aufruf des Dialogs "Elemente skalieren (Seite 1958)"

Siehe auch

Dialoge im Kontextmenü (Seite 1958)

Tabellarischer Editor

Kontextmenü im tabellarischen Editor V3.0

Folgende Tabelle zeigt die Funktionen im Kontextmenü des tabellarischen Editors:

Funktion	Beschreibung
Zeile einfügen	Einfügen eines Elements (Punkt) vor dem selektierten Element
Zeile hinzufügen	Einfügen eines Elements (Punkt) nach dem selektierten Element
Ausschneiden	Entfernen der ausgewählten Elemente und Kopieren in die Zwischenablage
Kopieren	Kopieren der ausgewählten Elemente in die Zwischenablage
Einfügen	Einfügen der Elemente aus der Zwischenablage nach dem letzten Element
Löschen	Löschen der ausgewählten Elemente Übergänge zu gegebenenfalls vorhandenen Elementen werden ebenfalls gelöscht.
Einfügen spezial	Aufruf des Dialogs "Elemente einfügen (Seite 1958)"
Verschieben	Aufruf des Dialogs "Elemente verschieben (Seite 1958)"
Skalieren	Aufruf des Dialogs "Elemente skalieren (Seite 1958)"

Eigenschaften (Inspektorfenster)

Profil - Statistik V3.0

Das Eigenschaftsfenster "Statistik" zeigt eine Übersicht über die Anzahl der Elemente der Kurvenscheibe. Eine Kurvenscheibe kann aus bis zu 1000 Punkten und bis zu 50 Segmenten bestehen.

Verbrauchte Elemente

Dieser Bereich zeigt die Anzahl der verbrauchten Elemente der Kurve:

Parameter	Beschreibung
Punkte	Dieses Feld zeigt die Anzahl der verbrauchten Punkte der Kurvenscheibe. Eine Kurvenscheibe kann aus bis zu 1000 Punkten bestehen.
Segmente	Dieses Feld zeigt die Anzahl der verbrauchten Segmente der Kurvenscheibe. Eine Kurvenscheibe kann aus bis zu 50 Segmenten bestehen.

Der Verbrauch von Punkten und Segmenten ist abhängig von der Zusammenstellung und der Konfiguration der Elemente. Folgende Tabelle zeigt den Verbrauch von Punkten und Segmenten je Element:

Element	Anzahl verbrauchter Punkte	Anzahl verbrauchter Segmente
Punkt	1	0
Punkt an einem Übergang mit VDI-basierter Optimierung	0	0
Gerade	0	1
Sinus	0	1
Polynom	0	1
Inverser Sinus	Anzahl der konfigurierten Stützpunkte (Eigenschaften (Inspektorfenster) > Element > Parameter > Approximation > Anzahl der Stützpunkte) Voreinstellung: 32	0
Inverser Sinus rechts von einem Übergang mit VDI-basierter Optimierung	Anzahl der konfigurierten Stützpunkte - 1	0
Übergang mit Systeminterpolation	0	0

Element	Anzahl verbrauchter Punkte	Anzahl verbrauchter Segmente
Übergang mit VDI-basierter Optimierung		
Bewegungsgesetz		
Sinus	0	1
Geneigter Sinus	0	1
Polynom	0	1
Modifiziertes Beschleunigungstrapez		
Bewegungsaufgabe		
Rast-in-Umkehr	0	5
Umkehr-in-Rast	0	5
Rast-in-Rast	0	6
Modifizierter Sinus		
Bewegungsaufgabe		
Rast-in-Rast	0	3
Konstante Geschwindigkeit-in-konstante Geschwindigkeit	0	4
Konstante Geschwindigkeit-in-Rast	0	4
Rast-in-konstante Geschwindigkeit	0	4
Sinus-Geraden-Kombination	0	3
Harmonische Kombination	0	3

Randbedingungen

Für die Eingabe und Verwendung von Punkten und Segmenten gelten folgende Randbedingungen:

- **Punkte**
  Bei Punkten mit gleichen Leitwerten wird der Punkt wirksam, der als letztes eingegeben wurde bzw. im tabellarischen Editor weiter unten steht.
- **Segmente**
  - Lücken zwischen Segmenten werden mit einem Übergangssegment gefüllt.
  - Bei Lücken im Leitwertbereich kleiner 1.0E-4 werden Segmentendpunkt und Segmentanfangspunkt zusammengezogen.
  - Bei Lücken im Leitwertbereich größer 1.0E-4 wird ein neues Übergangssegment eingefügt.
  - Bei Überlappungen wird das neue Segment ab dem Anfangspunkt eingefügt und vollständig verwendet. Wenn das vorherige Segment über das neue Segment hinaus definiert ist, wird nach dem Endpunkt des neuen Segments das vorherige Segment weiter verwendet.
- **Stützpunkte und Segmente** (gemischte Kurvenscheiben)
  Wenn Segmente und Punkte im gleichen Bereich definiert sind, wird das Segment verwendet.

Konfiguration Elemente - Parameter (Polynom) V3.0

Im Konfigurationsfenster "Parameter" konfigurieren Sie die Parameter des selektierten Elements.

Parameter

Konfigurieren Sie in diesem Bereich die Parameter des selektierten Polynoms:

Parameter		Beschreibung
Leitwerte des Polynoms		
	Anfang	Konfigurieren Sie in diesem Feld den Anfangspunkt des Polynoms im Leitwertbereich (Definitionsbereich).
	Ende	Konfigurieren Sie in diesem Feld den Endpunkt des Polynoms im Leitwertbereich (Definitionsbereich).
Polynomielle Parameter		
	Definition über	Wählen Sie in der Klappliste, wie das Polynom definiert wird: • Koeffizienten • Randwerte und Wendepunkt • Randwerte und linksseitiger Ruck • Randwerte und rechtsseitiger Ruck Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
	Koeffizienten	Konfigurieren Sie in diesen Feldern die Koeffizienten der Polynomfunktion 6. Grades: $P(x) = a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ Die Koeffizienten werden in wissenschaftlicher Zahlendarstellung angezeigt, z. B. "9.6450617283e-11".
	Folgewert - Anfang	Konfigurieren Sie in diesem Feld den Folgewert am Anfang des Polynoms.
	Folgewert - Ende	Konfigurieren Sie in diesem Feld den Folgewert am Ende des Polynoms.
	Erste Ableitung - Anfang	Konfigurieren Sie in diesem Feld die erste Ableitung (Geschwindigkeit) für den Folgewert am Anfang des Polynoms.
	Erste Ableitung - Ende	Konfigurieren Sie in diesem Feld die erste Ableitung (Geschwindigkeit) für den Folgewert am Ende des Polynoms.
	Zweite Ableitung - Anfang	Konfigurieren Sie in diesem Feld die zweite Ableitung (Beschleunigung) für den Folgewert am Anfang des Polynoms.
	Zweite Ableitung - Ende	Konfigurieren Sie in diesem Feld die zweite Ableitung (Beschleunigung) für den Folgewert am Ende des Polynoms.
	Dritte Ableitung - Anfang	Konfigurieren Sie in diesem Feld die dritte Ableitung (Ruck) für den Folgewert am Anfang des Polynoms.
	Dritte Ableitung - Ende	Konfigurieren Sie in diesem Feld die dritte Ableitung (Ruck) für den Folgewert am Ende des Polynoms.
	Leitwert des Wendepunkts	Konfigurieren Sie in diesem Feld den Leitwert des Wendepunkts des Polynoms.
Erweiterte Parameter		

Parameter		Beschreibung
	Segmentvariante	Wählen Sie in der Klappliste, ob das Polynom einen trigonometrischen Anteil haben soll. Bei Auswahl "Polynom mit trigonometrischem Anteil" werden analog zum Sinus die entsprechenden trigonometrischen Parameter angezeigt. Beim Umwandeln eines Sinuselements in ein Polynom wird das Sinuselement als Polynom mit trigonometrischem Anteil parametriert. Die Form des Elements bleibt erhalten.
	Amplitude	Konfigurieren Sie in diesem Feld die Amplitude des trigonometrischen Anteils.
	Definition über	Wählen Sie in der Klappliste, wie der trigonometrische Anteil definiert wird: • Phase am Anfang und am Ende • Phase am Anfang und Periodenlänge • Phase am Anfang und Frequenz • Periodenlänge und Phase am Ende • Frequenz und Phase am Ende Abhängig von der Auswahl werden die entsprechenden Parameter angezeigt.
	Phase am Anfang	Konfigurieren Sie in diesem Feld den Phasenwinkel am Anfang des trigonometrischen Anteils.
	Phase am Ende	Konfigurieren Sie in diesem Feld den Phasenwinkel am Ende des trigonometrischen Anteils.
	Periodenlänge	Konfigurieren Sie in diesem Feld Periodenlänge des trigonometrischen Anteils.
	Frequenz	Konfigurieren Sie in diesem Feld die Frequenz des trigonometrischen Anteils.

Konfiguration Elemente - Charakteristik (Übergang) V3.0

Im Konfigurationsfenster "Charakteristik" konfigurieren Sie die Parameter des selektierten Übergangs.

Parameter

Konfigurieren Sie in diesem Bereich die Vorgaben für die Optimierung des Übergangs:

Parameter		Beschreibung
Interpolationseinstellungen des Übergangs		
	Optimierungsmethode	Wählen Sie in der Klappliste die Optimierungsmethode: • Systeminterpolation Die CPU bestimmt die Parameter der Optimierung automatisch gemäß den Einstellungen der Systeminterpolation (Seite 1936). • VDI-basierte Optimierung Sie passen die Optimierung händisch an. Die Eingaben werden automatisch entsprechend der VDI-Richtlinie 2143 angewendet.
	Bewegungsaufgabe	Der Übergangstyp wird aus den Eigenschaften der Nachbarelemente des Übergangs ermittelt und in diesem Feld angezeigt.
	Stetigkeit am Anfang/Ende	Wählen Sie in den Klapplisten, welcher Parameter in den Randpunkten stetig ist und zur Optimierung berücksichtigt wird: • Optimierungsvoreinstellung (Einstellung unter "Profil > Optimierungsvoreinstellungen (Seite 1936)") • Position • Geschwindigkeit (stoßfrei) • Beschleunigung (ruckfrei) • Ruck (Ruckstetigkeit nur einseitig erlaubt)
	Optimierungsziel	Wählen Sie in der Klappliste das Optimierungsziel: • Optimierungsvoreinstellung (Einstellung unter "Profil > Optimierungsvoreinstellungen") • Nicht vorgegeben • Geschwindigkeit (Cv) • Beschleunigung (Ca) • Ruck (Cj) • Minimales Dynamikmoment (Cmdyn)
Auswahl des Bewegungsgesetzes		

Parameter		Beschreibung
	Bewegungsgesetz	Wählen Sie in der Klappliste das Bewegungsgesetz, nach dem optimiert wird: • Gerade • Quadratische Parabel • Sinus • Polynom • Geneigter Sinus • Modifiziertes Beschleunigungstrapez • Modifizierter Sinus • Harmonische Kombination • Sinus-Gerade-Kombination Die Auswahl wird automatisch auf die entsprechend der Bewegungsaufgabe und der gewählten Randbedingungen anwendbaren Bewegungsgesetze eingeschränkt. Abhängig vom gewählten Bewegungsgesetz werden weitere Parameter angezeigt.
	Verwendeter Parameter	Wählen Sie in der Klappliste, welcher Parameter für die Optimierung berücksichtigt wird: • Lambda • Maximale Beschleunigung (Ca) • Maximale Verzögerung (Ca*) Die Auswahl wird automatisch auf die entsprechend des Bewegungsgesetzes anwendbaren Parameter eingeschränkt.
	Lambda	Konfigurieren Sie in diesem Feld die Lage des Wendepunkts (Lambda) für den Übergang.
	Maximale Beschleunigung (Ca)	Konfigurieren Sie in diesem Feld die maximale Beschleunigung (Ca) für den Übergang.
	Maximale Verzögerung (Ca*)	Konfigurieren Sie in diesem Feld die maximale Verzögerung (Ca*) für den Übergang.
Kennwerte des Übergangs		In diesem Bereich werden die nach VDI 2143 relevanten Kennwerte des Übergangs angezeigt. Für folgende Kennwerte werden jeweils der maximale Wert und der normierte Wert angezeigt: • Geschwindigkeit (Cv) • Beschleunigung (Ca) • Verzögerung (Ca*) • Ruck (Cj) • Dynamisches Moment (Cmdyn)

Bewegungsaufgaben nach VDI-Richtlinie 2143

Die VDI-Richtlinie 2143 unterscheidet zwischen Nutzbereichen und Bewegungsübergängen:

- Nutzbereiche entsprechen den Arbeitsschritten in einem Prozess, also den eingefügten Elementen der Kurvenscheibe.
- Bewegungsübergänge sind Übergänge zwischen Nutzbereichen, die nicht unmittelbar prozessrelevant sind, aber gewissen Randbedingungen genügen müssen (z. B. Geschwindigkeitsstetigkeit).

Die VDI-Richtlinie 2143 definiert folgende Nutzbereiche:

Nutzbereich	Bezeichnung	Eigenschaften
Rast	R	Geschwindigkeit = 0 Beschleunigung = 0
Konstante Geschwindigkeit	G	Geschwindigkeit ≠ 0 Beschleunigung = 0
Umkehr	U	Geschwindigkeit = 0 Beschleunigung ≠ 0
Bewegung	B	Geschwindigkeit ≠ 0 Beschleunigung ≠ 0

Folgendes Bild zeigt beispielhaft die Nutzbereiche:

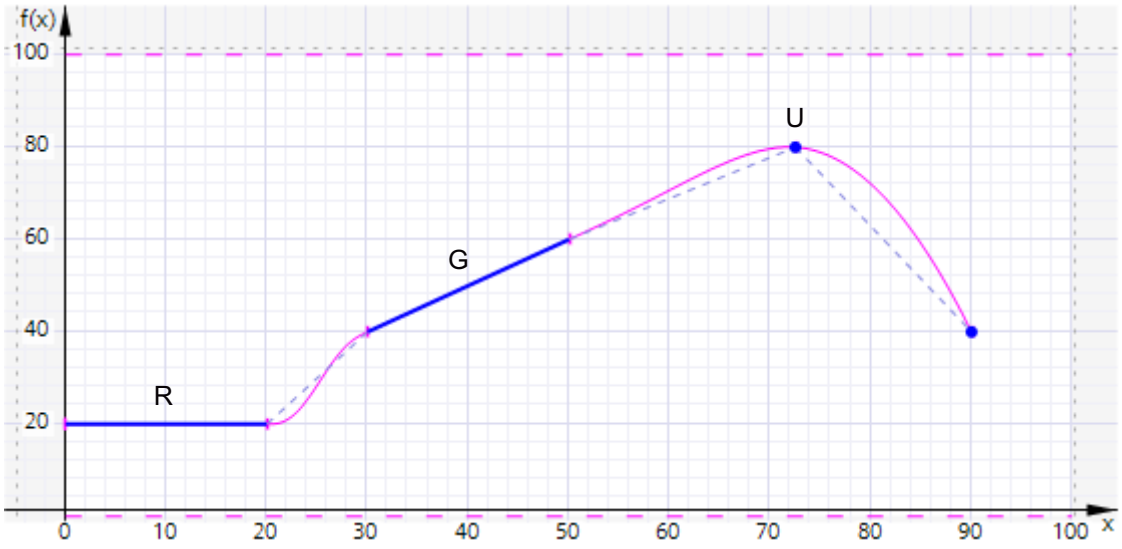

Folgendes Bild zeigt die möglichen Übergänge zwischen den Nutzbereichen:

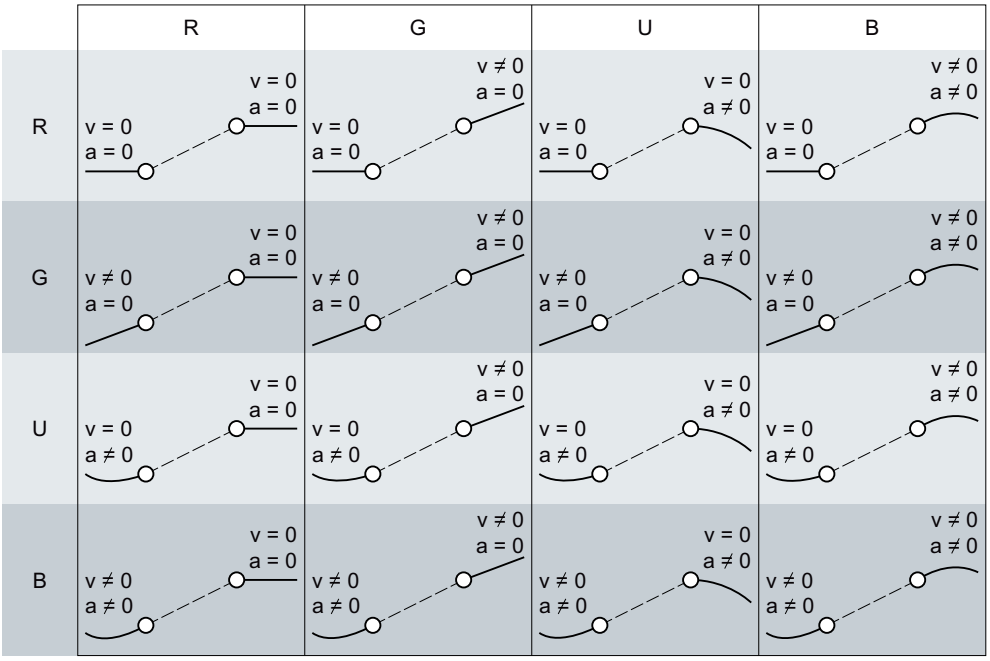

Inbetriebnahme

Achssteuertafel

Funktion und Aufbau der Achssteuertafel V3.0

Beschreibung

Mit der Achssteuertafel verfahren Sie einzelne Achsen.

Für den Betrieb der Achssteuertafel ist kein Anwenderprogramm notwendig. Mit der Achssteuertafel übernehmen Sie die Steuerungshoheit für ein Technologieobjekt und steuern die Bewegungen der Achse.

⚠ WARNUNG

Unkontrollierte Achsbewegungen

Beim Betrieb mit der Achssteuertafel kann die Achse unkontrollierte Bewegungen ausführen (z. B. aufgrund fehlerhafter Konfiguration des Antriebs oder des Technologieobjekts). Weiterhin wird beim Verfahren einer Leitachse mit der Achssteuertafel eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren.

Führen Sie daher vor dem Betrieb mit der Achssteuertafel folgende Schutzmaßnahmen durch:

- Stellen Sie sicher, dass sich der NOT-AUS-Schalter in Reichweite des Bedieners befindet.
- Aktivieren Sie die Hardware-Endschalter.
- Aktivieren Sie die Software-Endschalter.
- Stellen Sie sicher, dass die Schleppfehlerüberwachung aktiviert ist.
- Stellen Sie sicher, dass keine Folgeachse mit der zu verfahrenden Achse gekoppelt ist.

In der Projektnavigation finden Sie die Achssteuertafel der Technologieobjekte Drehzahlachse, Positionierachse und Gleichlaufachse unter "Technologieobjekt > Inbetriebnahme".

Die Achssteuertafel ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Betriebsart
- Steuern
- Status Achse
- Aktuelle Werte

Die folgende Tabelle zeigt die Elemente der Achssteuertafel:

Bereich	Element	Beschreibung
Steuerungshoheit		Im Bereich "Steuerungshoheit" übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben sie an Ihr Anwenderprogramm zurück.
	Schaltfläche "Holen"	Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. • Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein. • Beim Verfahren einer Leitachse mit der Achssteuertafel wird eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren. • Wenn die Onlineverbindung zur CPU während des Betriebs mit der Achssteuertafel ausfällt, wird die Achse nach Ablauf der Lebenszeichenüberwachung mit maximaler Verzögerung angehalten. In diesem Fall wird eine Fehlermeldung angezeigt ("ErrorID" = 16#8013) und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. • Wenn Sie auf die Schaltfläche "Holen" klicken, wird ein Warnhinweis angezeigt. Im Warnhinweis können Sie die Lebenszeichenüberwachung anpassen (100 bis 60000 ms). Wenn die Steuerungshoheit der Achssteuertafel wiederholt ohne direkte Fehlermeldung verloren geht, könnte die Onlineverbindung zur CPU durch eine zu hohe Kommunikationslast beeinträchtigt sein. In diesem Fall wird im Archiv der Meldungsanzeige die Meldung "Fehler bei Inbetriebnahme. Ausfall des Lebenszeichens zwischen Steuerung und TIA Portal" angezeigt. Um diesen Fehler zu beheben, passen Sie die Lebenszeichenüberwachung im Warnhinweis an. • Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler ("ErrorID" = 16#8012: Achssteuertafel aktiviert) abgewiesen. • Mit der Übernahme der Steuerungshoheit wird die Konfiguration des Technologieobjekts übernommen. Änderungen an der Konfiguration des Technologieobjekts werden erst wirksam, nach dem die Steuerungshoheit zurückgegeben wurde. Nehmen Sie daher eventuell erforderliche Änderungen vor der Übernahme der Steuerungshoheit vor. • Wenn die Steuerungshoheit für das Technologieobjekt übernommen wurde, ist die Achssteuertafel für den Zugriff durch eine andere Installation des TIA Portals gesperrt (Team Engineering ab CPU V1.5).
	Schaltfläche "Abgeben"	Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.
Achse		Im Bereich "Achse" geben Sie das Technologieobjekt frei oder sperren es.
	Schaltfläche "Freigeben"	Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
	Schaltfläche "Sperren"	Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.
Betriebsart		In der Klappliste "Betriebsart" wählen Sie die gewünschte Betriebsart der Achssteuertafel aus.
Steuern		Im Bereich "Steuern" werden die Parameter für das Verfahren mit der Achssteuertafel entsprechend der ausgewählten Betriebsart angezeigt.

Bereich	Element	Beschreibung
	Position	Position, auf welche die Achse referenziert wird. (Nur Betriebsarten Referenzieren, Referenzpunkt setzen)
	Weg	Wegstrecke, um welche die Achse verfahren wird. (Nur Betriebsart Positionieren relativ)
	Zielposition	Position, auf welche die Achse verfahren wird. (Nur Betriebsart Positionieren absolut)
	Geschwindigkeit/ Sollgeschwindigkeit	Geschwindigkeit bzw. Drehzahl, mit der die Achse verfahren wird. Vorbelegung: 10 % des Defaultwerts (Nur Betriebsarten Geschwindigkeits-/Drehzahlvorgabe, Tippen, Positionieren)
	Beschleunigung	Beschleunigung, mit der die Achse verfahren wird. Vorbelegung: 10 % des Defaultwerts
	Verzögerung	Verzögerung, mit der die Achse verfahren wird. Vorbelegung: 10 % des Defaultwerts
	Ruck	Ruck, mit dem die Achse verfahren wird. Vorbelegung: 100 % des Defaultwerts
	Schaltfläche "Start"	Mit der Schaltfläche "Start" stoßen Sie einen Auftrag gemäß der gewählten Betriebsart an.
	Schaltfläche "Vorwärts"	Mit der Schaltfläche "Vorwärts" stoßen Sie eine Bewegung gemäß der gewählten Betriebsart in positive Richtung an.
	Schaltfläche "Rückwärts"	Mit der Schaltfläche "Rückwärts" stoßen Sie eine Bewegung gemäß der gewählten Betriebsart in negative Richtung an.
	Schaltfläche "Stopp"	Mit der Schaltfläche "Stopp" brechen Sie einen Auftrag ab bzw. halten Sie die Achse an.
Status Achse		Im Bereich "Status Achse" wird der Status der Achse und der Status des Antriebs angezeigt.
	Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen.
	Fehler	Am Technologieobjekt ist ein Fehler aufgetreten.
	Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden.
	Referenziert	Das Technologieobjekt ist referenziert.
	Mehr	Über den Link "Mehr" gelangen Sie zum Fenster "Technologieobjekt > Diagnose > Status- und Fehlerbits".
	Anstehender Fehler	Im Textfeld "Anstehender Fehler" wird der zuletzt aufgetretene Fehler angezeigt.
	Schaltfläche "Quittieren"	Mit der Schaltfläche "Quittieren" quittieren Sie anstehende Fehler.
Aktuelle Werte		Im Bereich "Aktuelle Werte" werden die Istwerte der Achse angezeigt.
	Position	Istposition der Achse
	Geschwindigkeit	Istgeschwindigkeit der Achse

Hinweis

Keine Übernahme der Parameter

Die eingestellten Parameterwerte werden mit der Rückgabe der Steuerungshoheit verworfen. Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration.

Wenn Sie während des Betriebs mit der Achssteuertafel Konfigurationswerte geändert haben, so haben diese Änderungen keine Auswirkung auf den Betrieb der Achssteuertafel.

Betriebsart

Die folgende Tabelle zeigt die Betriebsarten der Achssteuertafel:

Betriebsart	Beschreibung
Referenzieren	Diese Funktion entspricht dem aktiven Referenzieren. Die Parameter für das Referenzieren (Seite 1729) müssen konfiguriert sein. Das Referenzieren ist mit einem Absolutwertgeber nicht möglich. Das Technologieobjekt wird bei Verwendung dieser Betriebsart mit einem Absolutwertgeber nicht referenziert.
Referenzpunkt setzen	Diese Funktion entspricht dem direkten Referenzieren (absolut). Mit der Schaltfläche "Start" setzen Sie die Istposition auf den in "Position" angegebenen Wert und der Status "Referenziert" wird gesetzt.
Tippen	Die Bewegungsbefehle erfolgen durch Tippen. Mit der Schaltfläche "Vorwärts" bzw. "Rückwärts" stoßen Sie Bewegung in positive bzw. negative Richtung an. Die Bewegung läuft, solange Sie die linke Maustaste gedrückt halten.
Geschwindigkeits-/ Drehzahlvorgabe	Die Achse wird so lange mit der angegebenen Geschwindigkeit bzw. Drehzahl verfahren, bis Sie die Bewegung stoppen. Die Bewegungsbefehle werden entsprechend den unter "Steuerung" parametrierten Sollwerten ausgeführt.
Relatives Positionieren	Das Positionieren wird als geregelte, relative Verfahrbewegung entsprechend den unter "Steuerung" parametrierten Vorgaben ausgeführt.
Absolutes Positionieren	Das Positionieren wird als geregelte, absolute Verfahrbewegung entsprechend den unter "Steuerung" parametrierten Vorgaben ausgeführt. Wenn Sie die Einstellung "Modulo" des Technologieobjekts aktiviert haben, werden im Bereich "Steuerung" die Schaltflächen "Vorwärts" und "Rückwärts" angezeigt. Die Achse wird innerhalb des Modulobereichs positioniert. Positionsvorgaben außerhalb des Modulobereichs werden entsprechend auf den Modulobereich umgerechnet. Wenn Sie die Einstellung "Modulo" des Technologieobjekts nicht aktiviert haben, wird im Bereich "Steuerung" nur die Schaltfläche "Start" angezeigt. Sie können direkt die eingegebene Position anfahren.

Optimierung

Funktion und Aufbau der Optimierung V3.0

Beschreibung

Die Funktion "Optimierung" unterstützt Sie bei der Ermittlung der optimalen Vorsteuerung und Verstärkung (Kv-Faktor) für die Lageregelung der Achse. Hierzu wird der Geschwindigkeitsverlauf der Achse während einer vorgebbaren Positionierbewegung mit der Trace-Funktion aufgezeichnet. Anschließend können Sie die Aufzeichnung auswerten und die Vorsteuerung und die Verstärkung entsprechend anpassen.

In der Projektnavigation finden Sie die Funktion "Optimierung" für die Technologieobjekte Positionierachse und Gleichlaufachse unter "Technologieobjekt > Inbetriebnahme".

Der Dialog "Optimierung" ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Achse
- Messkonfiguration
- Lageregler optimieren
- Messung durchführen
- Trace

Die folgende Tabelle zeigt die Elemente der Optimierung:

Bereich	Element	Beschreibung
Steuerungshoheit		Im Bereich "Steuerungshoheit" übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben sie an Ihr Anwenderprogramm zurück.
	Schaltfläche "Holen"	Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. • Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein. • Beim Verfahren einer Leitachse mit der Achssteuertafel wird eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren. • Wenn die Onlineverbindung zur CPU während des Betriebs mit der Achssteuertafel ausfällt, wird die Achse nach Ablauf der Lebenszeichenüberwachung mit maximaler Verzögerung angehalten. In diesem Fall wird eine Fehlermeldung angezeigt ("ErrorID" = 16#8013) und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. • Wenn Sie auf die Schaltfläche "Holen" klicken, wird ein Warnhinweis angezeigt. Hier können Sie die Lebenszeichenüberwachung anpassen (100 bis 60000 ms). Wenn die Steuerungshoheit der Achssteuertafel wiederholt ohne direkte Fehlermeldung verloren geht, könnte die Onlineverbindung zur CPU durch eine zu hohe Kommunikationslast beeinträchtigt sein. In diesem Fall wird im Archiv der Meldungsanzeige die Meldung "Fehler bei Inbetriebnahme. Ausfall des Lebenszeichens zwischen Steuerung und TIA Portal" angezeigt. Um diesen Fehler zu beheben, passen Sie die Lebenszeichenüberwachung im Warnhinweis an. • Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler ("ErrorID" = 16#8012: Achssteuertafel aktiviert) abgewiesen. • Mit der Übernahme der Steuerungshoheit wird die Konfiguration des Technologieobjekts übernommen. Änderungen an der Konfiguration des Technologieobjekts werden erst wirksam, nach dem die Steuerungshoheit zurückgegeben wurde. Nehmen Sie daher eventuell erforderliche Änderungen vor der Übernahme der Steuerungshoheit vor. • Wenn die Steuerungshoheit für das Technologieobjekt übernommen wurde, ist die Achssteuertafel für den Zugriff durch eine andere Installation des TIA-Portals gesperrt (Team Engineering ab CPU V1.5).
	Schaltfläche "Abgeben"	Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.
Achse		Im Bereich "Achse" geben Sie das Technologieobjekt frei oder sperren es.
	Schaltfläche "Freigeben"	Mit der Schaltfläche "Freigeben" geben Sie das ausgewählte Technologieobjekt frei.
	Schaltfläche "Sperren"	Mit der Schaltfläche "Sperren" sperren Sie das ausgewählte Technologieobjekt.

Bereich	Element	Beschreibung
Messkonfiguration		Im Bereich "Messkonfiguration" konfigurieren Sie die Einstellungen für einen Testschritt.
	Weg	Wegstrecke für einen Testschritt
	Messdauer	Zeitdauer für einen Testschritt
	Optionskästchen "Dynamikwerte anpassen"	Mit dem Optionskästchen "Dynamikwerte anpassen" ermöglichen Sie die Anpassung der Beschleunigung und der maximalen Geschwindigkeit.
	Beschleunigung	Voreinstellung der Beschleunigung für einen Testschritt
	Maximale Geschwindigkeit	Voreinstellung der maximalen Geschwindigkeit für einen Testschritt
Lageregler optimieren		Im Bereich "Lageregler optimieren" nehmen Sie die Einstellungen zum Optimieren der Regelparameter vor. Über das Symbol neben einem Feld öffnen Sie eine Werteliste. Die Werteliste beinhaltet folgende Werte des jeweiligen Parameters: • Online-Aktualwert (Istwert) • Online-Startwert • Startwert Projekt Tragen Sie im Eingabefeld für den Istwert den neuen Wert ein. Der neue Wert wird durch Klicken auf die Schaltfläche "Vorwärts" bzw. "Rückwärts" im Bereich "Messung durchführen" übernommen.
	Vorsteuerung	Aktuelle prozentuale Geschwindigkeitsvorsteuerung des Lagereglers
	Drehzahl-Regelkreis-Ersatzzeit	Aktuelle Drehzahl-Regelkreis-Ersatzzeit Bei einer Geschwindigkeitsvorsteuerung kann über die Drehzahl-Regelkreis-Ersatzzeit ein vereinfachtes Modell des Drehzahlregelkreises gebildet werden. Dadurch wird ein Übersteuern der Geschwindigkeitsstellgröße durch den Lageregler in den Beschleunigungs- und Verzögerungsphasen verhindern. Dazu wird der Positionssollwert des Lagereglers um die Drehzahl-Regelkreis-Ersatzzeit im Bezug zur Geschwindigkeitsvorsteuerung verzögert.
	Verstärkung	Aktuelle Verstärkung des Lagereglers (Kv)
Messung durchführen		Im Bereich "Messung durchführen" führen Sie die Testschritte aus.
	Schaltfläche "Vorwärts"	Mit der Schaltfläche "Vorwärts" starten Sie einen Testschritt zur Optimierung in positive Richtung.
	Schaltfläche "Rückwärts"	Mit der Schaltfläche "Rückwärts" starten Sie einen Testschritt zur Optimierung in negative Richtung.
	Schaltfläche "Stopp"	Mit der Schaltfläche "Stopp" beenden Sie einen Testschritt.
Trace		Im unteren Bereich des Dialogs "Optimierung" wird die Trace-Funktion angezeigt. Mit jedem Testschritt wird automatisch eine Trace-Aufzeichnung der benötigten Parameter gestartet und nach dem Beenden des Testschritts angezeigt. Nach der Rückgabe der Steuerungshoheit wird die Trace-Aufzeichnung gelöscht. Eine vollständige Beschreibung der Trace-Funktion finden Sie im Kapitel Trace- und Logikanalysatorfunktion nutzen.

Hinweis

Keine Übernahme der Parameter

Die eingestellten Parameterwerte werden nach der Rückgabe der Steuerungshoheit verworfen.

Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration. Die Werte der Verstärkung, der Vorsteuerung und der Drehzahl-Regelkreis-Ersatzzeit können Sie über den Wert "Startwert Projekt" in Ihre Konfiguration übernehmen.

Diagnose

Technologieobjekt Drehzahlachse

Status und Fehlerbits V3.0

Beschreibung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Achse

Die folgende Tabelle zeigt mögliche Zustände der Achse:

Status	Beschreibung
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben.
Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden. (<TO>.StatusWord.X0 (Enable))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen des Technologieobjekts <TO>.ErrorDetail.Number und <TO>.ErrorDetail.Reaction. (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Achssteuertafel aktiv	Die Achssteuertafel ist aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt. Die Achse kann nicht vom Anwenderprogramm gesteuert werden. (<TO>.StatusWord.X4 (ControlPanelActive))

Status	Beschreibung
Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen. (<TO>.StatusDrive.InOperation)
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen.(<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Achsbewegung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Bewegungsauftrag aktiv. (<TO>.StatusWord.X6 (Done))
Tippen	Die Achse wird mit einem Auftrag zum Tippbetrieb der Motion Control-Anweisung "MC_MoveJog" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X9 (JogCommand))
Drehzahlvorgabe	Die Achse wird mit einem Auftrag mit Drehzahlvorgabe der Motion Control-Anweisung "MC_MoveVelocity" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X10 (VelocityCommand))
Konstante Drehzahl	Die Achse wird mit konstanter Drehzahl verfahren oder befindet sich im Stillstand. (<TO>.StatusWord.X12 (ConstantVelocity))
Beschleunigen	Die Achse wird beschleunigt. (<TO>.StatusWord.X13 (Accelerating))
Verzögern	Die Achse wird abgebremst. (<TO>.StatusWord.X14 (Decelerating))
Momentenbegrenzung aktiv	An der Achse wirkt der konfigurierte Grenzwert für die Kraft/das Moment. (<TO>.StatusWord.X27 (InLimitation))

Warnungen

Die folgende Tabelle zeigt mögliche Warnungen:

Warnung	Beschreibung
Konfiguration	Einer oder mehrere Konfigurationsparameter werden zeitweise intern angepasst.. (<TO>.WarningWord.X1 (ConfigWarning))
Auftrag abgewiesen	Befehl nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind. (<TO>.WarningWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.WarningWord.X6 (DynamicError))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung. (<TO>.ErrorWord.X2 (UserFault))
Antrieb	Fehler im Antrieb. (<TO>.ErrorWord.X4 (DriveFault))
Datenaustausch	Die Kommunikation mit einem verbundenen Gerät ist gestört. (<TO>.ErrorWord.X7 (CommunicationFault))
Peripherie	Fehler beim Zugriff auf eine logische Adresse. (<TO>.ErrorWord.X13 (PeripheralError))
Auftrag abgewiesen	Befehl nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. Technologieobjekt nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.ErrorWord.X6 (DynamicError))

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten (Seite 1991).

Status Bewegung V3.0

Beschreibung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal den Bewegungsstatus der Achse. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Aktuelle Werte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Solldrehzahl	Solldrehzahl der Achse (<TO>.Velocity)
Drehzahl-Override	Prozentuale Korrektur der Drehzahlvorgabe Die von Motion Control-Anweisungen oder von der Achssteuertafel vorgegebene Solldrehzahl wird durch einen Override überlagert und prozentual angepasst. Als Drehzahlkorrektur sind Werte von 0.0 % bis 200.0 % zulässig. (<TO>.Override.Velocity)

Bereich "Dynamikgrenzen"

In diesem Bereich werden die konfigurierten Grenzwerte der Dynamikparameter angezeigt.

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Drehzahl	Konfigurierte maximale Drehzahl (<TO>.DynamicLimits.MaxVelocity)
Beschleunigung	Konfigurierte maximale Beschleunigung (<TO>.DynamicLimits.MaxAcceleration)
Verzögerung	Konfigurierte maximale Verzögerung (<TO>.DynamicLimits.MaxDeceleration)

Technologieobjekt Positionierachse/Gleichlaufachse

Status und Fehlerbits V3.0

Beschreibung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Achse

Die folgende Tabelle zeigt mögliche Zustände der Achse:

Status	Beschreibung
Simulation aktiv	Die Achse wird in der CPU simuliert. Sollwerte werden nicht an den Antrieb ausgegeben. (<TO>.StatusWord.X25 (AxisSimulation))
Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden. (<TO>.StatusWord.X0 (Enable))
Lagegeregelter Betrieb	Die Achse befindet sich im lagegeregelten Betrieb. (Invertierung von <TO>.StatusWord.X28 (NonPositionControlled))
Referenziert	Das Technologieobjekt ist referenziert. Der Bezug zwischen der Position am Technologieobjekt und der mechanischen Stellung wurde erfolgreich hergestellt. (<TO>.StatusWord.X5 (HomingDone))
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen des Technologieobjekts <TO>.ErrorDetail.Number und <TO>.ErrorDetail.Reaction (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Achssteuertafel aktiv	Die Achssteuertafel ist aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt. Die Achse kann nicht vom Anwenderprogramm gesteuert werden. (<TO>.StatusWord.X4 (ControlPanelActive))
Antrieb bereit	Der Antrieb ist bereit, Sollwerte auszuführen. (<TO>.StatusDrive.InOperation)
Geberistwerte gültig	Die Geberistwerte sind gültig. (<TO>.StatusSensor[n].State)
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Endschalter

Die folgende Tabelle zeigt mögliche Aktivierungen der Software- und Hardware-Endschalter:

Status	Beschreibung
Negativer SW-Endschalter angefahren	Der negative Software-Endschalter wurde angefahren. (<TO>.StatusWord.X15 (SWLimitMinActive))
Positiver SW-Endschalter angefahren	Der positive Software-Endschalter wurde angefahren. (<TO>.StatusWord.X16 (SWLimitMaxActive))
Negativer HW-Endschalter angefahren	Der negative Hardware-Endschalter wurde angefahren oder überfahren. (<TO>.StatusWord.X17 (HWLimitMinActive))
Positiver HW-Endschalter angefahren	Der positive Hardware-Endschalter wurde angefahren oder überfahren. (<TO>.StatusWord.X18 (HWLimitMaxActive))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Achsbewegung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Auftrag aktiv. (<TO>.StatusWord.X6 (Done))
Referenzierauftrag	Das Technologieobjekt führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus. (<TO>.StatusWord.X11 (HomingCommand))
Tippen	Die Achse wird mit einem Auftrag zum Tippbetrieb der Motion Control-Anweisung "MC_MoveJog" verfahren. (<TO>.StatusWord.X9 (JogCommand))
Geschwindigkeitsvorgabe	Die Achse wird mit einem Auftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisung "MC_MoveVelocity" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X10 (VelocityCommand))
Positionierauftrag	Die Achse wird mit einem Positionierauftrag der Motion Control-Anweisung "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel verfahren. (<TO>.StatusWord.X8 (PositioningCommand))
Konstante Geschwindigkeit	Die Achse wird mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand. (<TO>.StatusWord.X12 (ConstantVelocity))
Stillstand	Die Achse befindet sich im Stillstand. (<TO>.StatusWord.X7 (StandStill))
Beschleunigen	Die Achse wird beschleunigt. (<TO>.StatusWord.X13 (Accelerating))
Verzögern	Die Achse wird abgebremst. (<TO>.StatusWord.X14 (Decelerating))
Momentenbegrenzung aktiv	An der Achse wirkt der konfigurierte Grenzwert für die Kraft/das Moment. (<TO>.StatusWord.X27 (InLimitation))
Synchronisierung	Nur Gleichlaufachse Die Achse wird auf den Leitwert einer Leitachse aufsynchronisiert. (<TO>.StatusWord.X21 (Synchronizing))
Gleichlauf	Nur Gleichlaufachse Die Achse ist aufsynchronisiert und fährt synchron zur Leitachse. (<TO>.StatusWord.X22 (Synchronous))

Warnungen

Die folgende Tabelle zeigt mögliche Warnungen:

Warnung	Beschreibung
Konfiguration	Einer oder mehrere Konfigurationsparameter werden zeitweise intern angepasst. (<TO>.WarningWord.X1 (ConfigWarning))
Auftrag abgewiesen	Befehl nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind. (<TO>.WarningWord.X3 (CommandNotAccepted))

Warnung	Beschreibung
Dynamikbegrenzung	Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.WarningWord.X6 (DynamicError))
Synchronisieren	Nur Gleichlaufachse Fehler beim Gleichlauf. Die an der entsprechenden Motion Control-Anweisung angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert. (<TO>.WarningWord.X14 (SynchronousError))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Anwenderprogramm	Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung. (<TO>.ErrorWord.X2 (UserFault))
Antrieb	Fehler im Antrieb. (<TO>.ErrorWord.X4 (DriveFault))
Geber	Fehler im Gebersystem. (<TO>.ErrorWord.X5 (SensorFault))
Datenaustausch	Die Kommunikation mit einem verbundenen Gerät ist gestört. (<TO>.ErrorWord.X7 (CommunicationFault))
Peripherie	Fehler beim Zugriff auf eine logische Adresse. (<TO>.ErrorWord.X13 (PeripheralError))
Auftrag abgewiesen	Befehl nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. Technologieobjekt nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Referenzieren	Fehler bei einem Referenziervorgang. (<TO>.ErrorWord.X10 HomingFault))
Positionieren	Die Achse wurde am Ende einer Positionierbewegung nicht korrekt positioniert. (<TO>.ErrorWord.X12 (PositioningFault))
Dynamikbegrenzung	Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.ErrorWord.X6 (DynamicError))
Schleppfehler	Der maximale zulässige Schleppfehler wurde überschritten. (<TO>.ErrorWord.X11 (FollowingErrorFault))
SW-Endschalter	Ein Software-Endschalter wurde erreicht. (<TO>.ErrorWord.X8 (SwLimit))

Fehler	Beschreibung
HW-Endschalter	Ein Hardware-Endschalter wurde erreicht oder überfahren. (<TO>.ErrorWord.X9 (HWLimit))
Synchronisieren	Nur Gleichlaufachse Fehler beim Gleichlauf. Die an der entsprechenden Motion Control-Anweisung angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert. (<TO>.ErrorWord.X14 (SynchronousError))

Weitere Informationen

Eine Möglichkeit zur Auswertung der einzelnen Statusbits finden Sie im Kapitel StatusWord, ErrorWord und WarningWord auswerten (Seite 1991).

Status Bewegung V3.0

Beschreibung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal den Bewegungsstatus der Achse. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Aktuelle Werte"

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Operativer Geber	Operativ wirksamer Geber der Achse
Zielposition	Aktuelle Zielposition eines aktiven Positionierauftrags Der Wert der Zielposition ist nur während der Ausführung eines Positionierauftrags gültig. (<TO>.StatusPositioning.TargetPosition)
Istposition	Istposition der Achse Wenn das Technologieobjekt nicht referenziert ist, wird der Wert relativ zu der Position angezeigt, die zum Zeitpunkt der Freigabe des Technologieobjekts vorlag. (<TO>.ActualPosition)
Istgeschwindigkeit	Istgeschwindigkeit der Achse (<TO>.ActualVelocity)
Geschwindigkeits-Override	Prozentuale Korrektur der Geschwindigkeitsvorgabe Die von Motion Control-Anweisungen oder von der Achssteuertafel vorgegebene Sollgeschwindigkeit wird durch einen Override überlagert und prozentual angepasst. Als Geschwindigkeitskorrektur sind Werte von 0.0 % bis 200.0 % zulässig. (<TO>.Override.Velocity)

Bereich "Dynamikgrenzen"

In diesem Bereich werden die konfigurierten Grenzwerte der Dynamikparameter angezeigt.

Die folgende Tabelle beschreibt die Bedeutung der Statusinformationen:

Status	Beschreibung
Geschwindigkeit	Konfigurierte maximale Geschwindigkeit (<TO>.DynamicLimits.MaxVelocity)
Beschleunigung	Konfigurierte maximale Beschleunigung (<TO>.DynamicLimits.MaxAcceleration)
Verzögerung	Konfigurierte maximale Verzögerung (<TO>.DynamicLimits.MaxDeceleration)

6.1.2.10 Anhang

Variablen des Technologieobjekts Drehzahlachse

Legende

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Istwerte und Sollwerte (Drehzahlachse)

Die folgenden Variablen zeigen die Soll- und Istwerte des Technologieobjekts an.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
Velocity	LREAL	-	RON	Sollgeschwindigkeit/Solldrehzahl
ActualSpeed	LREAL	-	RON	Istdrehzahl des Motors (Bei Analogsollwert = 0.0)
Acceleration	LREAL	-	RON	Sollbeschleunigung

Variable "Simulation" (Drehzahlachse)

Die Variablenstruktur "<TO>.Simulation.<Variablenname>" beinhaltet die Konfiguration des Simulationsbetriebs. Im Simulationsbetrieb können Sie Achsen ohne reellen Antrieb in der CPU simulieren.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
Simulation.	STRUCT				
Mode	UDINT	-	RES[1]	Simulationsbetrieb	
				0	Keine Simulation, normaler Betrieb
				1	Simulationsbetrieb

1) Technologieversion V2.0: RON

Variable "VirtualAxis" (Drehzahlachse)

Die Variablenstruktur "<TO>.VirtualAxis.<Variablenname>" beinhaltet die Konfiguration des Simulationsbetriebs. Im Simulationsbetrieb können Sie Achsen ohne reellen Antrieb in der CPU simulieren.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
VirtualAxis.	STRUCT				
Mode	UDINT	-	RON	Virtuelle Achse	
				0	Keine virtuelle Achse
				1	Achse wird immer und ausschließlich als virtuelle Achse betrieben

Variable "Actor" (Drehzahlachse)

Die Variablenstruktur "<TO>.Actor.<Variablenname>" beinhaltet die steuerungsseitige Konfiguration des Antriebs.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
Actor.	STRUCT			

Variable	Datentyp	Werte	W	Beschreibung	
Type	DINT	-	RON	Antriebsanbindung	
				0	Analoger Ausgang
				1	PROFIdrive-Telegramm
InverseDirection	BOOL	-	RES	Invertierung des Sollwerts	
				FALSE	Nein
				TRUE	Ja
DataAdaption	DINT	-	RES	Automatische Übernahme der Antriebswerte Bezugsdrehzahl, maximale Drehzahl und Bezugsmoment im Gerät	
				0	Keine automatische Übernahme, händische Konfiguration der Werte
				1	Automatische Übernahme der im Antrieb konfigurierten Werte in die Konfiguration des Technologieobjekts
Efficiency	LREAL	0.0 bis 1.0	RES	Wirkungsgrad des Getriebes	
Interface.	STRUCT				
AddressIn	VREF	0 bis 65535	RON	Eingangsadresse für das PROFIdrive-Telegramm	
AddressOut	VREF	0 bis 65535	RON	Ausgangsadresse für das PROFIdrive-Telegramm oder den Analogsollwert	
EnableDriveOutput	BOOL	-	RES	"Freigabe-Ausgang" für analoge Antriebe	
				FALSE	Deaktiviert
				TRUE	Aktiviert
EnableDriveOutputAddress	VREF	0 bis 65535	RON	Adresse für den "Freigabe-Ausgang" bei Analogsollwert	
DriveReadyInput	BOOL	-	RES	"Bereit-Eingang" für analoge Antriebe Der analoge Antrieb meldet seine Bereitschaft zum Empfangen von Drehzahlsollwerten.	
				FALSE	Deaktiviert
				TRUE	Aktiviert
DriveReadyInputAddress	VREF	0 bis 65535	RON	Adresse für den "Freigabe-Eingang" bei Analogsollwert	
EnableTorqueData	BOOL	-	RES	Momentendaten	
				FALSE	Deaktiviert
				TRUE	Aktiviert
TorqueDataAddressIn	VREF	0 bis 65535	RON	Eingangsadresse des Telegramms 750	
TorqueDataAddressOut	VREF	0 bis 65535	RON	Ausgangsadresse des Telegramms 750	
DriveParameter.					
ReferenceSpeed	LREAL	0.0 bis 1.0E12	RES	Bezugswert (100 %) für die Solldrehzahl des Antriebs (N-soll) Der Drehzahlsollwert wird im PROFIdrive-Telegramm als normierter Wert von -200 % bis 200 % von "ReferenceSpeed" übertragen. Bei Sollwertvorgabe über einen Analogwert kann der Analogausgang im Bereich -117 % bis 117 % betrieben werden, sofern der Antrieb dies zulässt.	

Variable	Datentyp	Werte	W	Beschreibung
MaxSpeed	LREAL	0.0 bis 1.0E12	RES	Maximalwert für die Solldrehzahl des Antriebs (N-soll) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Analogsollwert: MaxSpeed ≤ 1.17 × ReferenceSpeed)
ReferenceTorque	LREAL	0.0 bis 1.0E12	RES	Bezugsdrehmoment des Antriebs (p2003). Gültig bei Einstellung Standardmotor.

Variable "TorqueLimiting" (Drehzahlachse)

Die Variablenstruktur "<TO>.TorqueLimiting.<Variablenname>" beinhaltet die Konfiguration der Momentenbegrenzung.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
TorqueLimiting.	STRUCT				
LimitBase	DINT	-	RES	Momentenbegrenzung	
				0	Motorseitig
				1	Lastseitig
PositionBasedMonitorings	DINT	-	RES	Positionier- und Schleppfehlerüberwachung	
				0	Überwachungen deaktiviert
				1	Überwachungen aktiviert
LimitDefaults.	STRUCT				
Torque	LREAL	0.0 bis 1.0E12	CAL	Begrenzungsdrehmoment	
Force	LREAL	0.0 bis 1.0E12	CAL	Begrenzungskraft	

Variable "LoadGear" (Drehzahlachse)

Die Variablenstruktur "<TO>.LoadGear.<Variablenname>" beinhaltet die Konfiguration des Lastgetriebes.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Wertebereich	W	Beschreibung
LoadGear.	STRUCT			
Numerator	UDINT	1 bis 4294967295	RES	Lastgetriebe Zähler
Denominator	UDINT	1 bis 4294967295	RES	Lastgetriebe Nenner

Variable "Units" (Drehzahlachse)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
VelocityUnit	UDINT	-	RON	Einheit für Geschwindigkeit	
				1082	1/s
				1083	1/min
				1528	1/h
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1054	s
TorqueUnit	UDINT	-	RON	Einheit für Drehmoment	
				1126	Nm
				1128	kNm
				1529	lbf in (pound-force-inch)
				1530	lbf ft
				1531	ozf in (ounce-force-inch)
				1532	ozf ft
				1533	pdl in (poundal-inch)
				1534	pdl ft
ForceUnit	UDINT	-	RON	Einheit für Kraft	
				1120	N
				1122	kN
				1094	lbf (pound-force)
				1093	ozf (ounce-force)
				1535	pdl (poundals)

Variable "DynamicLimits" (Drehzahlachse)

Die Variablenstruktur "<TO>.DynamicLimits.<Variablenname>" beinhaltet die Konfiguration der Dynamikgrenzen. Bei der Bewegungsführung werden keine Dynamikwerte größer der Dynamikgrenzen zugelassen. Wenn Sie an einer Motion Control-Anweisung größere Werte angegeben, wird mit den Dynamikgrenzen verfahren und eine Warnung (Alarm 501 bis 503 - Dynamikwerte werden begrenzt) wird angezeigt.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
DynamicLimits.	STRUCT			
MaxVelocity	LREAL	0.0 bis 1.0E12	RES	Maximal zulässige Geschwindigkeit der Achse
MaxAcceleration	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässige Beschleunigung der Achse
MaxDeceleration	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässige Verzögerung der Achse
MaxJerk	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässiger Ruck an der Achse

Variable "DynamicDefaults" (Drehzahlachse)

Die Variablenstruktur "<TO>.DynamicDefaults.<Variablenname>" beinhaltet die Konfiguration der Dynamikvoreinstellungen. Diese Einstellungen werden verwendet, wenn Sie an einer Motion Control-Anweisung einen Dynamikwert kleiner 0.0 angeben (Ausnahmen: "MC_MoveJog.Velocity", "MC_MoveVelocity.Velocity"). Änderungen der Dynamikvoreinstellungen werden mit der nächsten steigenden Flanke am Parameter "Execute" einer Motion Control-Anweisung übernommen.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
DynamicDefaults.	STRUCT			
Velocity	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Geschwindigkeit
Acceleration	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Beschleunigung
Deceleration	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Verzögerung
Jerk	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung des Rucks
EmergencyDecelerati-on	LREAL	0.0 bis 1.0E12	DIR	Notstopp-Verzögerung

Variable "Override" (Drehzahlachse)

Die Variablenstruktur "<TO>.Override.<Variablenname>" beinhaltet die Konfiguration von Override-Parametern. Mit Override-Parametern nehmen Sie eine prozentuale Korrektur vorgegebener Werte vor. Eine Override-Änderung ist sofort wirksam und wird mit den an der Motion Control-Anweisung wirksamen Dynamikeinstellungen herausgefahren.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
Override.	STRUCT			
Velocity	LREAL	0.0 bis 200.0 %	DIR	Geschwindigkeits- bzw. Drehzahl-Override Prozentuale Korrektur der Geschwindigkeit/Drehzahl

Variable "StatusDrive" (Drehzahlachse)

Die Variablenstruktur "<TO>.StatusDrive.<Variablenname>" zeigt den Status des Antriebs an.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
StatusDrive.	STRUCT				
InOperation	BOOL	-	RON	Operationsstatus des Antriebs	
				FALSE	Antrieb nicht bereit. Sollwerte werden nicht ausgeführt.
				TRUE	Antrieb bereit. Sollwerte können ausgeführt werden.
CommunicationOK	BOOL	-	RON	Zyklische BUS-Kommunikation zwischen Steuerung und Antrieb	
				FALSE	Nicht aufgebaut
				TRUE	Aufgebaut
Error	BOOL	-	RON	FALSE	Kein Fehler am Antrieb
				TRUE	Fehler am Antrieb
AdaptionState	DINT	-	RON	Status der automatischen Datenübernahme der Antriebsparameter	
				0	"NOT_ADAPTED" Daten nicht übernommen
				1	"IN_ADAPTION" Datenübernahme in Bearbeitung
				2	"ADAPTED" Datenübernahme abgeschlossen
				3	"NOT_APPLICABLE" Datenübernahme nicht angewählt, nicht möglich
				4	"ADAPTION_ERROR" Fehler bei der Datenübernahme

Variable "StatusTorqueData" (Drehzahlachse)

Die Variablenstruktur "<TO>.StatusTorqueData.<Variablenname>" zeigt den Status des Moments an.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Wertebereich	W	Beschreibung	
StatusTorqueData.	STRUCT				
CommandAdditiveTorqueActive	DINT	-	RON	Funktion additives Sollmoment	
				0	Deaktiviert
				1	Aktiviert
CommandTorqueRangeActive	DINT	-	RON	Funktion Momentenbereich über obere und untere Momentengrenze	
				0	Deaktiviert
				1	Aktiviert
ActualTorque	LREAL	-1.0E12 bis 1.0E12	RON	Istdrehmoment der Achse in der technologischen Einheit des Technologieobjekts für Moment	

Variable "StatusMotionIn" (Drehzahlachse)

Die Variablenstruktur "<TO>.StatusMotionIn.<Variablenname>" zeigt den Bewegungsstatus an.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Wertebereich	W	Beschreibung	
StatusMotionIn.	STRUCT				
FunctionState	DINT	-	RON	0	Keine "MotionIn"-Funktion aktiv
				1	"MotionInVelocity"-Funktion aktiv

Variable "StatusWord" (Drehzahlachse)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 0 "Enable") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts

Variable	Datentyp	Werte	W	Beschreibung	
Bit 0	-	-	-	"Enable" Freigabestatus	
				0	Technologieobjekt gesperrt
				1	Technologieobjekt freigegeben
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"ControlPanelActive" Achssteuertafel	
				0	Deaktiviert
				1	Aktiviert
Bit 5	-	-	-	Reserviert	
Bit 6	-	-	-	"Done"	
				0	Bewegungsauftrag in Bearbeitung bzw. Achssteuertafel aktiviert
				1	Kein Bewegungsauftrag in Bearbeitung und Achssteuertafel deaktiviert
Bit 7	-	-	-	Reserviert	
Bit 8	-	-	-	Reserviert	
Bit 9	-	-	-	"JogCommand"	
				0	Kein "MC_MoveJog"-Auftrag aktiv
				1	"MC_MoveJog"-Auftrag aktiv
Bit 10	-	-	-	"VelocityCommand"	
				0	Kein "MC_MoveVelocity"-Auftrag aktiv
				1	"MC_MoveVelocity"-Auftrag aktiv
Bit 11	-	-	-	Reserviert	
Bit 12	-	-	-	"ConstantVelocity"	
				0	Die Achse wird beschleunigt oder abgebremst.
				1	Die Sollgeschwindigkeit ist erreicht. Eine konstante Sollgeschwindigkeit wird ausgegeben.
Bit 13	-	-	-	"Accelerating"	
				0	Kein Beschleunigungsvorgang aktiv
				1	Beschleunigungsvorgang aktiv
Bit 14	-	-	-	"Decelerating"	
				0	Kein Verzögerungsvorgang aktiv

Variable	Datentyp	Werte	W	Beschreibung	
				1	Verzögerungsvorgang aktiv
Bit 15 … Bit 24	-	-	-	Reserviert	
Bit 25	-	-	-	"AxisSimulation"	
				0	Keine Simulation
				1	Simulation ist aktiv
Bit 26	-	-	-	"TorqueLimitingCommand"	
				0	Kein "MC_TorqueLimiting"-Auftrag ist aktiv
				1	"MC_TorqueLimiting"-Auftrag ist aktiv
Bit 27	-	-	-	"InLimitation"	
				0	Antrieb arbeitet nicht an der Momentengrenze
				1	Antrieb arbeitet an der Momentengrenze
Bit 28 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Drehzahlachse)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Systemfehler
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	"DriveFault" Fehler im Antrieb
Bit 5	-	-	-	Reserviert
Bit 6	-	-	-	"DynamicError" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7	-	-	-	"CommunicationFault" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14	-	-	-	Reserviert
Bit 15	-	-	-	"AdaptionError" Fehler bei der Datenübernahme
Bit 16 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Drehzahlachse)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0 bis 5	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				1	Stopp mit aktuellen Dynamikwerten
				2	Stopp mit maximalen Dynamikwerten
				3	Stopp mit Notstopp-Rampe
				4	Freigabe wegnehmen
				5	Sollwerte nachführen

Variable "WarningWord" (Drehzahlachse)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 13 "PeripheralWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	"DriveWarning" Fehler im Antrieb
Bit 5	-	-	-	Reserviert
Bit 6	-	-	-	"DynamicWarning" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7	-	-	-	"CommunicationWarning" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14	-	-	-	Reserviert
Bit 15	-	-	-	"AdaptionWarning" Fehler bei der automatischen Datenübernahme
Bit 16 … Bit 31	-	-	-	Reserviert

Variable "ControlPanel" (Drehzahlachse)

Die Variablenstruktur "<TO>.ControlPanel.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
ControlPanel.	STRUCT			
Input.	STRUCT			
TimeOut	LREAL	100 bis 60000	DIR	-
EsLifeSign	UDINT	-	DIR	-
Command.	ARRAY [1..2] OF STRUCT			
ReqCounter	UDINT	-	DIR	-
Type	UDINT	-	DIR	-
Position	LREAL	-	DIR	-
Velocity	LREAL	-	DIR	-
Acceleration	LREAL	-	DIR	-
Deceleration	LREAL	-	DIR	-
Jerk	LREAL	-	DIR	-
Param	LREAL	-	DIR	-
Output.	STRUCT			
RTLifeSign	UDINT	-	RON	-
Command.	ARRAY [1..2] OF STRUCT			
AckCounter	UDINT	-	RON	-
Error	BOOL	-	RON	-
ErrorID	UDINT	-	RON	-
Done	BOOL	-	RON	-
Aborted	BOOL	-	RON	-

Variable "InternalToTrace" (Drehzahlachse)

Die Variablenstruktur "<TO>.InternalToTrace.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Variablen

Legende (Seite 2085)

Variable	Datentyp	Werte	W	Beschreibung
InternalToTrace.	ARRAY [1..4] OF STRUCT			
Id	DINT	-	DIR	-
Value	LREAL	-	DIR	-

Variablen des Technologieobjekts Positionierachse/Gleichlaufachse

Legende

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert (L = lineare Angabe, R = rotatorische Angabe) Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Istwerte und Sollwerte (Positionierachse/Gleichlaufachse)

Die folgenden Variablen zeigen die Soll- und Istwerte des Technologieobjekts an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Position	LREAL	-	RON	Sollposition
Velocity	LREAL	-	RON	Sollgeschwindigkeit/Solldrehzahl
ActualPosition	LREAL	-	RON	Istposition
ActualVelocity	LREAL	-	RON	Istgeschwindigkeit
ActualSpeed	LREAL	-	RON	Istdrehzahl des Motors (Bei Analogsollwert = 0.0)
Acceleration	LREAL	-	RON	Sollbeschleunigung
ActualAcceleration	LREAL	-	RON	Istbeschleunigung
OperativeSensor	UDINT	1 bis 4	RON	Operativ wirksamer Geber

Variable "Simulation" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Simulation.<Variablenname>" beinhaltet die Konfiguration des Simulationsbetriebs. Im Simulationsbetrieb können Sie Achsen ohne reellen Antrieb in der CPU simulieren.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
Simulation.	STRUCT				
Mode	UDINT	-	RES[1]	Simulationsbetrieb	
				0	Keine Simulation, normaler Betrieb
				1	Simulationsbetrieb

1) Technologieversion V2.0: RON

Variable "VirtualAxis" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.VirtualAxis.<Variablenname>" beinhaltet die Konfiguration des Simulationsbetriebs. Im Simulationsbetrieb können Sie Achsen ohne reellen Antrieb in der CPU simulieren.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
VirtualAxis.	STRUCT				
Mode	UDINT	-	RON	Virtuelle Achse	
				0	Keine virtuelle Achse
				1	Achse wird immer und ausschließlich als virtuelle Achse betrieben

Variable "Actor" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Actor.<Variablenname>" beinhaltet die steuerungsseitige Konfiguration des Antriebs.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Actor.	STRUCT			

Variable	Datentyp	Werte	W	Beschreibung	
Type	DINT	-	RON	Antriebsanbindung	
				0	Analoger Ausgang
				1	PROFIdrive-Telegramm
InverseDirection	BOOL	-	RES	Invertierung des Sollwerts	
				FALSE	Nein
				TRUE	Ja
DataAdaption	DINT	-	RES	Automatische Übernahme der Antriebswerte Bezugsdrehzahl, maximale Drehzahl und Bezugsmoment	
				0	Keine automatische Übernahme, händische Konfiguration der Werte
				1	Automatische Übernahme der im Antrieb konfigurierten Werte in die Konfiguration des Technologieobjekts
Efficiency	LREAL	0.0 bis 1.0	RES	Wirkungsgrad der Mechanik (Getriebe und Spindel)	
Interface.	STRUCT				
AddressIn	VREF	0 bis 65535	RON	Eingangsadresse für das PROFIdrive-Telegramm	
AddressOut	VREF	0 bis 65535	RON	Ausgangsadresse für das PROFIdrive-Telegramm oder den Analogsollwert	
EnableDriveOutput	BOOL	-	RES	"Freigabe-Ausgang" für analoge Antriebe	
				FALSE	Deaktiviert
				TRUE	Aktiviert
EnableDriveOutputAddress	VREF	0 bis 65535	RON	Adresse für den "Freigabe-Ausgang" bei Analogsollwert	
DriveReadyInput	BOOL	-	RES	"Bereit-Eingang" für analoge Antriebe Der analoge Antrieb meldet seine Bereitschaft zum Empfangen von Drehzahlsollwerten.	
				FALSE	Deaktiviert
				TRUE	Aktiviert
DriveReadyInputAddress	VREF	0 bis 65535	RON	Adresse für den "Freigabe-Eingang" bei Analogsollwert	
EnableTorqueData	BOOL	-	RES	Momentendaten	
				FALSE	Deaktiviert
				TRUE	Aktiviert
TorqueDataAddressIn	VREF	0 bis 65535	RON	Eingangsadresse des Zusatztelegramms	
TorqueDataAddressOut	VREF	0 bis 65535	RON	Ausgangsadresse des Zusatztelegramms	
DriveParameter.					
ReferenceSpeed	LREAL	0.0 bis 1.0E12	RES	Bezugswert (100 %) für die Solldrehzahl des Antriebs (N-soll) Der Drehzahlsollwert wird im PROFIdrive-Telegramm als normierter Wert von -200 % bis 200 % von "ReferenceSpeed" übertragen. Bei Sollwertvorgabe über einen Analogwert kann der Analogausgang im Bereich -117 % bis 117 % betrieben werden, sofern der Antrieb dies zulässt.	

Variable	Datentyp	Werte	W	Beschreibung
MaxSpeed	LREAL	0.0 bis 1.0E12	RES	Maximalwert für die Solldrehzahl des Antriebs (N-soll) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Analogsollwert: MaxSpeed ≤ 1.17 × ReferenceSpeed)
ReferenceTorque	LREAL	0.0 bis 1.0E12	RES	Bezugswert (100 %) für das Drehmoment des Antriebs

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "TorqueLimiting" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.TorqueLimiting.<Variablenname>" beinhaltet die Konfiguration der Momentenbegrenzung.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
TorqueLimiting.	STRUCT				
LimitBase	DINT	-	RES	Momentenbegrenzung	
				0	Motorseitig
				1	Lastseitig
PositionBasedMonitorings	DINT	-	RES	Positionier- und Schleppfehlerüberwachung	
				0	Überwachungen deaktiviert
				1	Überwachungen aktiviert
LimitDefaults.	STRUCT				
Torque	LREAL	0.0 bis 1.0E12	CAL	Begrenzungsdrehmoment	
Force	LREAL	0.0 bis 1.0E12	CAL	Begrenzungskraft	

Variable "Clamping" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Clamping.<Variablenname>" beinhaltet die Konfiguration der Festanschlagserkennung.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Clamping.	STRUCT			
FollowingErrorDeviation	LREAL	0.001 bis 1.0E12	DIR	Wert des Schleppfehlers, ab dem der Festanschlag erkannt wird.
PositionTolerance	LREAL	0.001 bis 1.0E12	DIR	Positionstoleranz für die Klemmüberwachung

Variablen "Sensor[n]" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Sensor[n].<Variablenname>" beinhaltet die steuerungsseitige Konfiguration des Gebers und die Konfiguration des aktiven und passiven Referenzierens.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Sensor[n].	ARRAY [1..4] OF STRUCT			

Variable	Datentyp	Werte	W	Beschreibung	
Existent	BOOL	-	RON	Anzeige angelegter Sensoren	
Type	DINT	-	RON	Gebertyp	
				0	"INCREMENTAL" (inkrementell)
				1	"ABSOLUTE" (absolut)
				2	"CYCLIC_ABSOLUTE" (zyklisch absolut)
InverseDirection	BOOL	-	RES	Invertierung des Istwerts	
				FALSE	Nein
				TRUE	Ja
System	DINT	-	RES	Gebersystem	
				0	"LINEAR" (linearer Geber)
				1	"ROTATORY" (rotatorischer Geber)
MountingMode	DINT	-	RES	Anbauart des Gebers	
				0	An der Motorwelle
				1	An der Lastseite
				2	Externes Messsystem
DataAdaption	DINT	-	RES	Automatische Übernahme der Antriebswerte Bezugsdrehzahl, maximale Drehzahl und Bezugsmoment im Gerät	
				0	Keine automatische Übernahme, händische Konfiguration der Werte
				1	Automatische Übernahme der im Antrieb konfigurierten Werte in die Konfiguration des Technologieobjekts
Interface.					
AddressIn	VREF	0 bis 65535	RON	Eingangsadresse für das PROFIdrive-Telegramm	
AddressOut	VREF	0 bis 65535	RON	Ausgangsadresse für das PROFIdrive-Telegramm	
Number	UDINT	1 bis 2	RON	Nummer des Gebers im Telegramm	
Parameter.					
Resolution	LREAL	1.0E-12 bis 1.0E12	RES	Auflösung eines linearen Gebers (Abstand zwischen zwei Geberstrichen)	
StepsPerRevolution	UDINT	1 bis 8388608	RES	Inkremente pro Geberumdrehung bei einem rotatorischen Geber	
FineResolutionXist1	UDINT	0 bis 31	RES	Anzahl Bits für die Feinauflösung Gx_XIST1 (zyklischer Geberistwert)	
FineResolutionXist2	UDINT	0 bis 31	RES	Anzahl Bits für die Feinauflösung Gx_XIST2 (Absolutwert des Gebers)	
DeterminableRevolutions	UDINT	0 bis 8388608	RES	Anzahl unterscheidbarer Geberumdrehungen bei einem Multiturn-Absolutwertgeber (Bei Singleturn-Absolutwertgeber = 1; bei Inkrementalgeber = 0)	
DistancePerRevolution	LREAL	0.0 bis 1.0E12	RES	Weg der Last pro Geberumdrehung bei einem extern montierten Geber	
ActiveHoming.	STRUCT				
Mode	DINT	-	RES	Referenziermodus	
				0	Nullmarke über PROFIdrive-Telegramm verwenden

Variable	Datentyp	Werte	W	Beschreibung	
				1	Nullmarke über PROFIdrive-Telegramm und Referenznocken verwenden
				2	Referenzmarke über Digitaleingang verwenden
SideInput	BOOL	-	CAL	Seite des Digitaleingangs beim aktiven Referenzieren	
				FALSE	Negative Seite
				TRUE	Positive Seite
Direction	DINT	-	CAL	Referenzierrichtung/Anfahrrichtung auf die Referenzmarke	
				0	Positive Referenzierrichtung
				1	Negative Referenzierrichtung
DigitalInputAddress	VREF	0 bis 65535	RON	Adresse digitaler Eingang	
HomePositionOffset	LREAL	-1.0E12 bis 1.0E12	CAL	Referenzpunktverschiebung	
SwitchLevel	BOOL	-	RES	Signalpegel, der bei angefahrener Referenzmarke am digitalen Eingang ansteht	
				FALSE	Unterer Pegel
				TRUE	Oberer Pegel
PassiveHoming.	STRUCT				
Mode	DINT	-	RES	Referenziermodus	
				0	Nullmarke über PROFIdrive-Telegramm verwenden
				1	Nullmarke über PROFIdrive-Telegramm und Referenznocken verwenden
				2	Referenzmarke über Digitaleingang verwenden
SideInput	BOOL	-	CAL	Seite des digitalen Eingangs beim passiven Referenzieren	
				FALSE	Negative Seite
				TRUE	Positive Seite
Direction	DINT	-	CAL	Referenzierrichtung/Anfahrrichtung auf die Referenzmarke	
				0	Positive Referenzierrichtung
				1	Negative Referenzierrichtung
				2	Aktuelle Referenzierrichtung
DigitalInputAddress	VREF	0 bis 65535	RON	Adresse digitaler Eingang	
SwitchLevel	BOOL	-	RES	Signalpegel, der bei angefahrener Referenzmarke am digitalen Eingang ansteht	
				FALSE	Unterer Pegel
				TRUE	Oberer Pegel

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "Extrapolation" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Extrapolation.<Variablenname>" beinhaltet die Konfiguration der Istwertextrapolation.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Extrapolation.	STRUCT			
LeadingAxisDependentTime	LREAL	-	RON	Anteil der Extrapolationszeit (bedingt durch Leitachse) Ergibt sich aus den folgenden Zeiten: • Zeit der Istwerterfassung an der Leitachse • Interpolator-Takt • Zeit des Istpositionsfilters der Leitachse (T1 + T2)
FollowingAxisDependentTime	LREAL	0.0 bis 1.0E12	DIR	Anteil der Extrapolationszeit (bedingt durch Folgeachse) Ergibt sich aus den folgenden Zeiten: • Für eine Folgeachse mit eingestellter Geschwindigkeitsvorsteuerung: – Kommunikationstakt – Interpolator-Takt – Drehzahl-Regelkreis-Ersatzzeit an der Folgeachse – Ausgabezeit des Sollwertes an der Folgeachse • Für eine Folgeachse ohne Geschwindigkeitsvorsteuerung: – Kommunikationstakt – Interpolator-Takt – Lage-Regelkreis-Ersatzzeit (1/Kv aus "<TO>.PositionControl.Kv") – Ausgabezeit des Sollwertes an der Folgeachse
PositionFilter.	STRUCT			
T1	LREAL	0.0 bis 1.0E12	DIR	Positionsfilter Zeitkonstante T1
T2	LREAL	0.0 bis 1.0E12	DIR	Positionsfilter Zeitkonstante T2
VelocityFilter.	STRUCT			
T1	LREAL	0.0 bis 1.0E12	DIR	Geschwindigkeitsfilter Zeitkonstante T1
VelocityTolerance.	STRUCT			
Range	LREAL	0.0 bis 1.0E12	DIR	Toleranzbandbreite für die Geschwindigkeit
Hysteresis.	STRUCT			
Value	LREAL	0.0 bis 1.0E12	DIR	Hysteresewert

Variable "LoadGear" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.LoadGear.<Variablenname>" beinhaltet die Konfiguration des Lastgetriebes.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Wertebereich	W	Beschreibung
LoadGear.	STRUCT			
Numerator	UDINT	1 bis 4294967295	RES	Lastgetriebe Zähler
Denominator	UDINT	1 bis 4294967295	RES	Lastgetriebe Nenner

Variable "Properties" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Properties.<Variablenname>" beinhaltet die Konfiguration des Achs- bzw. Bewegungstyps.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Wertebereich	W	Beschreibung	
Properties.	STRUCT				
MotionType	DINT	-	RON	Anzeige des Achstyps bzw. Bewegungstyps	
				0	Lineare Achse bzw. Bewegung
				1	Rotatorische Achse bzw. Bewegung

Variable "Units" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Units.	STRUCT			

Variable	Datentyp	Werte	W	Beschreibung	
LengthUnit	UDINT	-	RON	Einheit für Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
				1004	rad
				1005	°
VelocityUnit	UDINT	-	RON	Einheit für Geschwindigkeit	
				1521	°/s
				1522	°/min
				1086	rad/s
				1523	rad/min
				1062	mm/s
				1061	m/s
				1524	mm/min
				1525	m/min
				1526	mm/h
				1063	m/h
				1527	km/min
				1064	km/h
				1066	in/s
				1069	in/min
				1067	ft/s
				1070	ft/min
				1075	mi/h
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1054	s
TorqueUnit	UDINT	-	RON	Einheit für Drehmoment	
				1126	Nm
				1128	kNm
				1529	lbf in (pound-force-inch)
				1530	lbf ft
				1531	ozf in (ounce-force-inch)
				1532	ozf ft
				1533	pdl in (poundal-inch)
				1534	pdl ft
ForceUnit	UDINT	-	RON	Einheit für Kraft	
				1120	N
				1122	kN

Variable		Datentyp	Werte	W	Beschreibung	
					1094	lbf (pound-force)
					1093	ozf (ounce-force)
					1535	pdl (poundals)

Variable "Mechanics" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Mechanics.<Variablenname>" beinhaltet die Konfiguration der Mechanik.

Variablen

Legende (Seite 2102)

Variable		Datentyp	Wertebereich	W	Beschreibung
Mechanics.		STRUCT			
	LeadScrew	LREAL	0.0 bis 1.0E12	RES	Spindelsteigung

Variable "Modulo" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Modulo.<Variablenname>" beinhaltet die Konfiguration der Modulofunktion.

Variablen

Legende (Seite 2102)

Variable		Datentyp	Werte	W	Beschreibung	
Modulo.		STRUCT				
	Enable	BOOL	-	RES	FALSE	Moduloumrechnung deaktiviert
					TRUE	Moduloumrechnung aktiviert
					Bei aktivierter Moduloumrechnung wird auf Modulolänge > 0.0 geprüft.	
	Length	LREAL	0.001 bis 1.0E12	RES	Modulolänge	
	StartValue	LREAL	-1.0E12 bis 1.0E12	RES	Modulostartwert	

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "DynamicLimits" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.DynamicLimits.<Variablenname>" beinhaltet die Konfiguration der Dynamikgrenzen. Bei der Bewegungsführung werden keine Dynamikwerte größer der Dynamikgrenzen zugelassen. Wenn Sie an einer Motion Control-Anweisung größere Werte angegeben, wird mit den Dynamikgrenzen verfahren und eine Warnung (Alarm 501 bis 503 - Dynamikwerte werden begrenzt) wird angezeigt.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
DynamicLimits.	STRUCT			
MaxVelocity	LREAL	0.0 bis 1.0E12	RES	Maximal zulässige Geschwindigkeit der Achse
Velocity	LREAL	0.0 bis 1.0E12	DIR	Aktuell maximale Geschwindigkeit der Achse
MaxAcceleration	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässige Beschleunigung der Achse
MaxDeceleration	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässige Verzögerung der Achse
MaxJerk	LREAL	0.0 bis 1.0E12	DIR	Maximal zulässiger Ruck an der Achse

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "DynamicDefaults" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.DynamicDefaults.<Variablenname>" beinhaltet die Konfiguration der Dynamikvoreinstellungen. Diese Einstellungen werden verwendet, wenn Sie an einer Motion Control-Anweisung einen Dynamikwert kleiner 0.0 angeben (Ausnahmen: "MC_MoveJog.Velocity", "MC_MoveVelocity.Velocity"). Änderungen der Dynamikvoreinstellungen werden mit der nächsten steigenden Flanke am Parameter "Execute" einer Motion Control-Anweisung übernommen.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
DynamicDefaults.	STRUCT			
Velocity	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Geschwindigkeit
Acceleration	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Beschleunigung
Deceleration	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung der Verzögerung
Jerk	LREAL	0.0 bis 1.0E12	CAL	Voreinstellung des Rucks
EmergencyDecelerati-on	LREAL	0.0 bis 1.0E12	DIR	Notstopp-Verzögerung

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "PositionLimits_SW" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.PositionLimits_SW.<Variablenname>" beinhaltet die Konfiguration der Positionsüberwachung mit Software-Endschaltern. Mit Software-Endschaltern begrenzen Sie den Arbeitsbereich einer Positionierachse.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
PositionLimits_SW.	STRUCT				
Active	BOOL	-	DIR	FALSE	Überwachung deaktiviert
				TRUE	Überwachung aktiviert
MinPosition	LREAL	-1.0E12 bis 1.0E12	DIR	Position negativer Software-Endschalter	
MaxPosition	LREAL	-1.0E12 bis 1.0E12	DIR	Position positiver Software-Endschalter ("MaxPosition" > "MinPosition")	

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "PositionLimits_HW" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.PositionLimits_HW.<Variablenname>" beinhaltet die Konfiguration der Positionsüberwachung mit Hardware-Endschaltern. Mit Hardware-Endschaltern begrenzen Sie den Verfahrbereich einer Positionierachse.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
PositionLimits_HW.	STRUCT				
Active	BOOL	-	RES	FALSE	Überwachung deaktiviert
				TRUE	Überwachung aktiviert
				Mit "Active" werden beide (negativer und positiver) Hardware-Endschalter aktiviert bzw. deaktiviert.	
MinSwitchLevel	BOOL	-	RES	Pegelauswahl zur Aktivierung des negativen Hardware-Endschalters	
				FALSE	Unterer Pegel (Low-aktiv)
				TRUE	Oberer Pegel (High-aktiv)
MinSwitchAddress	VREF	0 bis 65535	RON	Adresse für den negativen Hardware-Endschalter	
MaxSwitchLevel	BOOL	-	RES	Pegelauswahl zur Aktivierung des positiven Hardware-Endschalters	
				FALSE	Unterer Pegel (Low-aktiv)
				TRUE	Oberer Pegel (High-aktiv)
MaxSwitchAddress	VREF	0 bis 65535	RON	Adresse für den positiven Hardware-Endschalter	

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "Homing" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Homing.<Variablenname>" beinhaltet die Konfiguration für das Referenzieren des Technologieobjekts.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Homing.	STRUCT			
AutoReversal	BOOL	-	RES	Umkehren an Hardware-Endschaltern FALSE: Nein TRUE: Ja
ApproachDirection	BOOL	-	CAL	Anfahrrichtung auf den Referenzpunktschalter FALSE: Positive Richtung TRUE: Negative Richtung
ApproachVelocity	LREAL	Linear: 0.0 bis 10000.0 mm/s Rotatorisch: 0.0 bis 360000.0 °/s	CAL	Anfahrgeschwindigkeit Geschwindigkeit beim aktiven Referenzieren, mit der auf den Referenznocken und auf den Referenzpunkt zugefahren wird.
ReferencingVelocity	LREAL	Linear: 0.0 bis 1000.0 mm/s Rotatorisch: 0.0 bis 36000.0 °/s	CAL	Referenziergeschwindigkeit Geschwindigkeit beim aktiven Referenzieren, mit der auf die Referenzpunktposition gefahren wird.
HomePosition	LREAL	-1.0E12 bis 1.0E12	CAL	Referenzpunktposition

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "Override" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.Override.<Variablenname>" beinhaltet die Konfiguration von Override-Parametern. Mit Override-Parametern nehmen Sie eine prozentuale Korrektur vorgegebener Werte vor. Eine Override-Änderung ist sofort wirksam und wird mit den an der Motion Control-Anweisung wirksamen Dynamikeinstellungen herausgefahren.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Override.	STRUCT			
Velocity	LREAL	0.0 bis 200.0 %	DIR	Geschwindigkeits- bzw. Drehzahl-Override Prozentuale Korrektur der Geschwindigkeit/Drehzahl

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "PositionControl" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.PositionControl.<Variablenname>" beinhaltet Einstellungen der Lageregelung.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
PositionControl.	STRUCT			

Variable	Datentyp	Werte	W	Beschreibung
Kv	LREAL	0.0 bis 2147480.0	DIR	P-Verstärkung der Lageregelung ("Kv" > 0.0)
Kpc	LREAL	0.0 bis 150.0 %	DIR	Geschwindigkeitsvorsteuerung der Lageregelung Empfohlene Einstellung: • Taktsynchrone Antriebsanbindung über PROFIdrive: 100.0 % • Nicht taktsynchrone Antriebsanbindung über PROFIdrive: 0.0 bis 100.0 % • Analoge Antriebsanbindung: 0.0 bis 100.0 %
EnableDSC	BOOL	-	RES	Dynamic Servo Control (DSC) FALSE: DSC deaktiviert TRUE: DSC aktiviert DSC ist nur bei Verwendung von einem der folgenden PROFIdrive-Telegramme möglich: • Standardtelegramm 5 oder 6 • SIEMENS-Telegramm 105 oder 106
SmoothingTimeBy-ChangeDifference	LREAL	0.0 bis 1.0E+12	DIR	Stellgrößenrampe bei Sprung der Regeldifferenz durch Geberumschaltung.
InitialOperativeSensor	UDINT	-	RES	Nach Initialisierung der Achse wirksamer Sensor. (Sensornummer 1 bis 4) Dieser Geber wird nach dem Anlauf der CPU und nach einem Restart des Technologieobjekts (Seite 2016) verwendet. Bei einem Betriebszustandsübergang STOP → RUN der CPU (ohne Restart des Technologieobjekts) wird der Geber weiterverwendet, der auch vor dem STOP aktiv war.
ControlDifference-Quantization.	STRUCT			
Mode	DINT	-	RES	Art der Quantisierung Konfiguration einer Quantisierung bei Anschluss eines Antriebs mit Schrittmotor-Schnittstelle 0: Keine Quantisierung 1: Quantisierung entsprechend Geberauflösung 2: Quantisierung auf direkten Wert (Konfiguration erfolgt über die Parameteransicht (Datenstruktur))
Value	LREAL	0.001 bis 1.0E +12	RES	Wert der Quantisierung Konfiguration eines Werts bei Quantisierung auf direktem Wert (<TO>.PositionControl.ControlDifferenceQuantization.Mode = 2) (Konfiguration erfolgt über die Parameteransicht (Datenstruktur))

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "DynamicAxisModel" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.DynamicAxisModel.<Variablenname>" beinhaltet Einstellungen des Symmetriefilter.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
DynamicAxisModel.	STRUCT			
VelocityTimeConstant	LREAL	0.0 bis 1.0E12	DIR	Drehzahl-Regelkreis-Ersatzzeit [s]
AdditionalPositionTi-meConstant	LREAL	0.0 bis 1.0E12	DIR	Additive Positions-Regelkreis-Ersatzzeit [s]

Variable "FollowingError" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.FollowingError.<Variablenname>" beinhaltet die Konfiguration der dynamischen Schleppfehlerüberwachung.

Bei Überschreitung des zulässigen Schleppfehlers wird der Technologie-Alarm 521 ausgegeben und das Technologieobjekt gesperrt (Alarmreaktion: Freigabe wegnehmen).

Beim Erreichen des Warnpegels wird eine Warnung ausgegeben (Technologie-Alarm 522).

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
FollowingError.	STRUCT				
EnableMonitoring	BOOL	-	RES	FALSE	Schleppfehlerüberwachung deaktiviert
				TRUE	Schleppfehlerüberwachung aktiviert
MinValue	LREAL	Linear: 0.0 bis 1.0E12	DIR	Zulässiger Schleppfehler bei Geschwindigkeiten unterhalb des Wertes von "MinVelocity"	
		Rotatorisch: 0.001 bis 1.0E12			
MaxValue	LREAL	Linear: 0.0 bis 1.0E12	DIR	Maximal zulässiger Schleppfehler, der beim Maximum der Geschwindigkeit erreicht werden darf.	
		Rotatorisch: 0.002 bis 1.0E12			
MinVelocity	LREAL	0.0 bis 1.0E12	DIR	"MinValue" ist unterhalb dieser Geschwindigkeit zulässig und wird konstant gehalten.	
WarningLevel	LREAL	0.0 bis 100.0	DIR	Warnpegel Prozentualer Wert bezogen auf den maximal zulässigen Schleppfehler	

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "PositioningMonitoring" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.PositioningMonitoring.<Variablenname>" beinhaltet die Konfiguration der Positionierüberwachung am Ende einer Positionierbewegung.

Wenn der Positionsistwert am Ende einer Positionierbewegung innerhalb der Toleranzzeit das Positionierfenster erreicht und für die minimale Verweildauer im Positionierfenster verbleibt, wird im Technologie-Datenbaustein "<TO>.StatusWord.X5 (Done)" gesetzt. Damit ist ein Bewegungsauftrag abgeschlossen.

Bei Überschreitung der Toleranzzeit wird der Technologie-Alarm 541 "Positionierüberwachung" mit Zusatzwert 1: "Zielbereich nicht erreicht" angezeigt.

Bei Unterschreitung der minimalen Verweildauer wird der Technologie-Alarm 541 "Positionierüberwachung" mit Zusatzwert 2: "Zielbereich wieder verlassen" angezeigt.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
PositioningMonitoring.	STRUCT			
ToleranceTime	LREAL	0.0 bis 1.0E12	DIR	Toleranzzeit Maximal erlaubte Zeitdauer vom Erreichen der Sollgeschwindigkeit null bis zum Eintritt in das Positionierfenster
MinDwellTime	LREAL	0.0 bis 1.0E12	DIR	Minimale Verweildauer im Positionierfenster
Window	LREAL	0.0 bis 1.0E12	DIR	Positionierfenster

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "StandstillSignal" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StandstillSignal.<Variablenname>" beinhaltet die Konfiguration des Stillstandsignals.

Wenn der Geschwindigkeitsistwert die Geschwindigkeitsschwelle unterschreitet und während der minimalen Verweildauer nicht überschreitet, wird das Stillstandssignal "<TO>.StatusWord.X7 (Standstill)" gesetzt.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
StandstillSignal.	STRUCT			Konfiguration des Stillstandssignals
VelocityThreshold	LREAL	0.0 bis 1.0E12	DIR	Geschwindigkeitsschwelle Wird diese unterschritten, beginnt die minimale Verweildauer.
MinDwellTime	LREAL	0.0 bis 1.0E12	DIR	Miniimale Verweildauer

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "StatusPositioning" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusPositioning.<Variablenname>" zeigt den Status einer Positionierbewegung an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
StatusPositioning.	STRUCT			
Distance	LREAL	-1.0E12 bis 1.0E12	RON	Abstand zur Zielposition
TargetPosition	LREAL	-1.0E12 bis 1.0E12	RON	Zielposition
FollowingError	LREAL	-1.0E12 bis 1.0E12	RON	Aktueller Schleppfehler
SetpointExecutionTime	LREAL	-1.0E12 bis 1.0E12	RON	Sollwert-Ausführungszeit der Achse (Ergibt sich aus T_{lpo}, T_{vtc} bzw. 1/kv, T_{Send} und T_O der Achse)

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "StatusDrive" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusDrive.<Variablenname>" zeigt den Status des Antriebs an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
StatusDrive.	STRUCT				
InOperation	BOOL	-	RON	Operationsstatus des Antriebs	
				FALSE	Antrieb nicht bereit. Sollwerte werden nicht ausgeführt.
				TRUE	Antrieb bereit. Sollwerte können ausgeführt werden.
CommunicationOK	BOOL	-	RON	Zyklische BUS-Kommunikation zwischen Steuerung und Antrieb	
				FALSE	Nicht aufgebaut
				TRUE	Aufgebaut
Error	BOOL	-	RON	FALSE	Kein Fehler am Antrieb
				TRUE	Fehler am Antrieb
AdaptionState	DINT	-	RON	Status der automatischen Datenübernahme der Antriebsparameter	
				0	"NOT_ADAPTED" Daten nicht übernommen
				1	"IN_ADAPTION" Datenübernahme in Bearbeitung
				2	"ADAPTED" Datenübernahme abgeschlossen
				3	"NOT_APPLICABLE" Datenübernahme nicht angewählt, nicht möglich
				4	"ADAPTION_ERROR" Fehler bei der Datenübernahme

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "StatusServo" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusServo.<Variablenname>" zeigt den Status zum Symmetriefilter an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
StatusServo.	STRUCT			
BalancedPosition	LREAL	-	RON	Position nach dem Symmetriefilter
ControlDifference	LREAL	-	RON	Regelfehler

Variablen "StatusSensor[n]" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusSensor[n].<Variablenname>" zeigt den Status des Messsystems an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
StatusSensor[n].	ARRAY [1..4] OF STRUCT				
State	DINT	-	RON	Status des Geberistwerts	
				0	"NOT_VALID" Nicht gültig
				1	"WAITING_FOR_VALID" Warte auf Status "Gültig"
				2	"VALID" Gültig
CommunicationOK	BOOL	-	RON	Zyklische BUS-Kommunikation zwischen Steuerung und Geber	
				FALSE	Nicht aufgebaut
				TRUE	Aufgebaut
Error	BOOL	-	RON	FALSE	Kein Fehler im Messsystem
				TRUE	Fehler im Messsystem
AbsEncoderOffset	LREAL	-	RON	Referenzpunktverschiebung zum Wert eines Absolutwertgebers. Der Wert wird remanent in der CPU gespeichert.	
Control	BOOL	-	RON	FALSE	Geber ist nicht aktiv
				TRUE	Geber ist aktiv
Position	LREAL	-	RON	Geberposition	
Velocity	LREAL	-	RON	Gebergeschwindigkeit	
AdaptionState	DINT	-	RON	Status der automatischen Datenübernahme der Geberparameter	
				0	"NOT_ADAPTED" Daten nicht übernommen
				1	"IN_ADAPTION" Datenübernahme in Bearbeitung
				2	"ADAPTED" Datenübernahme abgeschlossen
				3	"NOT_APPLICABLE" Datenübernahme nicht angewählt, nicht möglich
				4	"ADAPTION_ERROR" Fehler bei der Datenübernahme

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "StatusExtrapolation" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusExtrapolation.<Variablenname>" zeigt den Status der Istwertextrapolation.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
Extrapolation.	STRUCT			
FilteredPosition	LREAL	-1.0E12 bis 1.0E12	RON	Position nach Positionsfilter
FilteredVelocity	LREAL	-1.0E12 bis 1.0E12	RON	Geschwindigkeit nach Toleranzband
ExtrapolatedPosition	LREAL	-1.0E12 bis 1.0E12	RON	Extrapolierte Position
ExtrapolatedVelocity	LREAL	-1.0E12 bis 1.0E12	RON	Extrapolierte Geschwindigkeit

Variable "StatusSynchronizedMotion" (Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusSynchronizedMotion.<Variablenname>" zeigt den Status des Gleichlaufs an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Wertebereich	W	Beschreibung	
StatusSynchronizedMotion.	STRUCT				
FunctionState	DINT	-	RON	Anzeige, welche Gleichlauffunktion aktiv ist	
				0	Kein Gleichlauf aktiv
				1	Getriebegleichlauf (MC_GearIn)
				2	Getriebegleichlauf mit vorgegebenen Synchronpositionen (MC_GearInPos)
				3	Kurvenscheibengleichlauf (MC_CamIn)
PhaseShift	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle absolute Leitwertverschiebung	
ActualMaster	DB_ANY	0 bis 65535	RON	Beim Start eines "MC_GearIn"-Auftrags wird die Nummer des Technologie-Datenbausteins der aktuell verwendeten Leitachse angezeigt. "ActualMaster" = 0 bei inaktivem Gleichlauf	
ActualCam	DB_ANY	0 bis 65535	RON	Aktuell für den Kurvenscheibengleichlauf verwendete Kurvenscheibe	
MasterOffset	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Verschiebung des Leitwertbereichs der Kurvenscheibe	
MasterScaling	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Skalierung des Leitwertbereichs der Kurvenscheibe	
SlaveOffset	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Verschiebung des Folgewertbereichs der Kurvenscheibe	
SlaveScaling	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Skalierung des Folgewertbereichs der Kurvenscheibe	
StatusWord.	STRUCT				
Bit 0	BOOL	-	RON	"MaxVelocityExceeded" Im Gleichlauf wird die konfigurierte maximale Geschwindigkeit überschritten.	
Bit 1	BOOL	-	RON	"MaxAccelerationExceeded" Im Gleichlauf wird die konfigurierte maximale Beschleunigung überschritten.	
Bit 2	BOOL	-	RON	"MaxDecelerationExceeded" Im Gleichlauf wird die konfigurierte maximale Verzögerung überschritten.	
Bit 3	BOOL	-	RON	"InSimulation"	
				FALSE	Gleichlauf nicht in Simulation
				TRUE	Gleichlauf in Simulation
Bit 4 … Bit 31	BOOL	-	RON	Reserviert	

Siehe auch

Technologieobjekt Gleichlaufachse (Seite 1764)

Variable "StatusKinematicsMotion" (Positionierachse/Gleichlaufachse)

Die Variable "<TO>.StatusKinematicsMotion" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "MaxDecelerationExceeded") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
StatusKinematicsMotion	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"MaxVelocityExceeded"	
				0	Das Technologieobjekt Kinematik hat eine niedrigere Sollgeschwindigkeit berechnet als die maximale Geschwindigkeit an der Achse.
				1	Das Technologieobjekt Kinematik hat eine höhere Sollgeschwindigkeit berechnet als die maximale Geschwindigkeit an der Achse.
Bit 1	-	-	-	"MaxAccelerationExceeded"	
				0	Das Technologieobjekt Kinematik hat eine niedrigere Sollbeschleunigung berechnet als die maximale Beschleunigung der Achse.
				1	Das Technologieobjekt Kinematik hat eine höhere Sollbeschleunigung berechnet als die maximale Beschleunigung der Achse.
Bit 2	-	-	-	"MaxDecelerationExceeded"	
				0	Das Technologieobjekt Kinematik hat eine niedrigere Sollverzögerung berechnet als die maximale Verzögerung der Achse.
				1	Das Technologieobjekt Kinematik hat eine höhere Sollverzögerung berechnet als die maximale Verzögerung der Achse.

Variable "StatusTorqueData" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusTorqueData.<Variablenname>" zeigt den Status der Momentendaten an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Wertebereich	W	Beschreibung	
StatusTorqueData.	STRUCT				
CommandAdditiveTorqueActive	DINT	-	RON	Funktion additives Sollmoment aktiv	
				0	Nein
				1	Ja
CommandTorqueRangeActive	DINT	-	RON	Funktion Momentengrenzen B+, B- aktiv	
				0	Nein
				1	Ja
ActualTorque	LREAL	-1.0E12 bis 1.0E12	RON	Istdrehmoment der Achse	

Variable "StatusMotionIn" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.StatusMotionIn.<Variablenname>" zeigt den Status der "MotionIn"-Funktion an.

Variablen

Legende (Seite 2102)

Variable	Datentyp	Wertebereich	W	Beschreibung	
StatusMotionIn.	STRUCT				
FunctionState	DINT	-	RON	0	Keine "MotionIn"-Funktion aktiv
				1	"MC_MotionInVelocity" aktiv
				2	"MC_MotionInPosition" aktiv

Variable "StatusWord" (Positionierachse/Gleichlaufachse)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 5 "HomingDone") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts

Variable	Datentyp	Werte	W	Beschreibung	
Bit 0	-	-	-	"Enable" Freigabestatus	
				0	Ein Technologieobjekt ist gesperrt.
				1	Ein Technologieobjekt ist freigegeben.
Bit 1	-	-	-	"Error"	
				0	Kein Fehler ist vorhanden.
				1	Ein Fehler ist vorhanden.
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart ist aktiv.
				1	Ein Restart ist aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Die Restart-Variablen sind unverändert.
				1	Die Restart-Variablen wurden verändert. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"ControlPanelActive"	
				0	Die Achssteuertafel ist deaktiviert.
				1	Die Achssteuertafel ist aktiviert.
Bit 5	-	-	-	"HomingDone" Referenzierungsstatus	
				0	Das Technologieobjekt ist nicht referenziert.
				1	Das Technologieobjekt ist referenziert.
Bit 6	-	-	-	"Done"	
				0	Ein Bewegungsauftrag ist in Bearbeitung bzw. die Achssteuertafel ist aktiviert.
				1	Kein Bewegungsauftrag ist in Bearbeitung und die Achssteuertafel ist deaktiviert.
Bit 7	-	-	-	"Standstill" Stillstandssignal	
				0	Die Achse ist in Bewegung.
				1	Die Achse ist im Stillstand.
Bit 8	-	-	-	"PositioningCommand"	
				0	Kein Positionierauftrag ist aktiv.
				1	Ein Positionierauftrag ist aktiv ("MC_MoveRelative", "MC_MoveAbsolute").
Bit 9	-	-	-	"JogCommand"	
				0	Kein "MC_MoveJog"-Auftrag ist aktiv.
				1	Ein "MC_MoveJog"-Auftrag ist aktiv.
Bit 10	-	-	-	"VelocityCommand"	
				0	Kein "MC_MoveVelocity"-Auftrag ist aktiv.
				1	Ein "MC_MoveVelocity"-Auftrag ist aktiv.
Bit 11	-	-	-	"HomingCommand"	
				0	Kein "MC_Home"-Auftrag ist in Bearbeitung.

Variable	Datentyp	Werte	W	Beschreibung	
				1	Ein "MC_Home"-Auftrag ist in Bearbeitung.
Bit 12	-	-	-	"ConstantVelocity"	
				0	Die Achse wird beschleunigt oder abgebremst.
				1	Die Sollgeschwindigkeit ist erreicht. Eine konstante Sollgeschwindigkeit wird ausgegeben.
Bit 13	-	-	-	"Accelerating"	
				0	Kein Beschleunigungsvorgang ist aktiv.
				1	Ein Beschleunigungsvorgang ist aktiv.
Bit 14	-	-	-	"Decelerating"	
				0	Kein Verzögerungsvorgang ist aktiv.
				1	Ein Verzögerungsvorgang ist aktiv.
Bit 15	-	-	-	"SWLimitMinActive"	
				0	Kein negativer Software-Endschalter wurde angefahren.
				1	Ein negativer Software-Endschalter wurde angefahren oder überfahren.
Bit 16	-	-	-	"SWLimitMaxActive"	
				0	Kein positiver Software-Endschalter wurde angefahren.
				1	Ein positiver Software-Endschalter wurde angefahren oder überfahren.
Bit 17	-	-	-	"HWLimitMinActive"	
				0	Kein negativer Hardware-Endschalter wurde angefahren.
				1	Ein negativer Hardware-Endschalter wurde angefahren oder überfahren.
Bit 18	-	-	-	"HWLimitMaxActive"	
				0	Kein positiver Hardware-Endschalter wurde angefahren.
				1	Ein positiver Hardware-Endschalter wurde angefahren oder überfahren.
Bit 19	-	-	-	Reserviert	
Bit 20	-	-	-	Reserviert	
Bit 21	-	-	-	Technologieobjekt Positionierachse: Reserviert	
				Technologieobjekt Gleichlaufachse: "Synchronizing"	
				0	Die Achse synchronisiert nicht auf einen Leitwert auf.
				1	Die Achse synchronisiert auf einen Leitwert auf.
Bit 22	-	-	-	Technologieobjekt Positionierachse: Reserviert	
				Technologieobjekt Gleichlaufachse: "Synchronous"	

Variable	Datentyp	Werte	W	Beschreibung	
				0	Die Achse verfährt nicht synchron zu einem Leitwert.
				1	Die Achse verfährt synchron zu einem Leitwert.
Bit 23	-	-	-	"SuperimposedMotionCommand"	
				0	Keine überlagerte Bewegung ist aktiv.
				1	Eine überlagerte Bewegung ist aktiv.
Bit 24	-	-	-	Technologieobjekt Positionierachse: Reserviert	
				Technologieobjekt Gleichlaufachse: "PhasingCommand"	
				0	Keine Motion Control-Anweisung zur Leitwertverschiebung ist aktiv.
				1	Eine Motion Control-Anweisung zur Leitwertverschiebung ist aktiv.
Bit 25	-	-	-	"AxisSimulation"	
				0	Die Simulation ist nicht aktiv.
				1	Die Simulation ist aktiv.
Bit 26	-	-	-	"TorqueLimitingCommand"	
				0	Kein "MC_TorqueLimiting"-Auftrag ist aktiv.
				1	Ein "MC_TorqueLimiting"-Auftrag ist aktiv.
Bit 27	-	-	-	"InLimitation"	
				0	Der Antrieb arbeitet nicht an der Momentengrenze.
				1	Der Antrieb arbeitet an der Momentengrenze.
Bit 28	-	-	-	"NonPositionControlled"	
				0	Die Achse ist im lagegeregelten Betrieb.
				1	Die Achse ist im nicht-lagegeregelten Betrieb.
Bit 29	-	-	-	"KinematicsMotionCommand"	
				0	Die Achse wird nicht für einen Kinematikauftrag verwendet.
				1	Die Achse wird für einen Kinematikauftrag verwendet.
Bit 30	-	-	-	"InClamping"	
				0	Die Achse steht nicht an einem Festanschlag in Klemmung.
				1	Die Achse steht an einem Festanschlag in Klemmung.
Bit 31	-	-	-	"MotionInCommand"	
				0	Kein "MotionIn"-Auftrag ist aktiv.
				1	Ein "MotionIn"-Auftrag ist aktiv.

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "ErrorWord" (Positionierachse/Gleichlaufachse)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	

Variable	Datentyp	Werte	W	Beschreibung
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	"DriveFault" Fehler im Antrieb
Bit 5	-	-	-	"SensorFault" Fehler im Gebersystem
Bit 6	-	-	-	"DynamicError" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7	-	-	-	"CommunicationFault" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8	-	-	-	"SWLimit" Software-Endschalter angefahren oder überfahren.
Bit 9	-	-	-	"HWLimit" Hardware-Endschalter angefahren oder überfahren.
Bit 10	-	-	-	"HomingError" Fehler beim Referenziervorgang Das Referenzieren kann nicht abgeschlossen werden.
Bit 11	-	-	-	"FollowingErrorFault" Schleppfehlergrenzen überschritten
Bit 12	-	-	-	"PositioningFault" Positionierfehler
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14	-	-	-	Technologieobjekt Positionierachse: Reserviert
				Technologieobjekt Gleichlaufachse: "SynchronousError" Fehler beim Gleichlauf Die an der Motion Control-Anweisung "MC_GearIn" angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert.

Variable	Datentyp	Werte	W	Beschreibung
Bit 15	-	-	-	"AdaptionError" Fehler bei der automatischen Datenübernahme
Bit 16 … Bit 31	-	-	-	Reserviert

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "ErrorDetail" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0 bis 5	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				1	Stopp mit aktuellen Dynamikwerten
				2	Stopp mit maximalen Dynamikwerten
				3	Stopp mit Notstopp-Rampe
				4	Freigabe wegnehmen
				5	Sollwerte nachführen

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Technologie-Alarme (Seite 2033)

Variable "WarningWord" (Positionierachse/Gleichlaufachse)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 13 "PeripheralWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2102)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	

Variable	Datentyp	Werte	W	Beschreibung
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	"DriveWarning" Fehler im Antrieb
Bit 5	-	-	-	"SensorWarning" Fehler im Gebersystem
Bit 6	-	-	-	"DynamicError" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7	-	-	-	"CommunicationWarning" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8	-	-	-	"SWLimitMin"
Bit 9	-	-	-	"SWLimitMax"
Bit 10	-	-	-	"HomingWarning" Fehler beim Referenziervorgang Das Referenzieren kann nicht abgeschlossen werden.
Bit 11	-	-	-	"FollowingErrorWarning" Warnpegel der Schleppfehlerüberwachung erreicht/überschritten
Bit 12	-	-	-	"PositioningWarning" Positionierfehler
Bit 13	-	-	-	"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14	-	-	-	Technologieobjekt Positionierachse: Reserviert
				Technologieobjekt Gleichlaufachse: "SynchronousWarning" Fehler beim Gleichlauf Die an der Motion Control-Anweisung "MC_GearIn" angegebene Leitachse wurde nicht als mögliche Leitachse konfiguriert.
Bit 15	-	-	-	"AdaptionWarning"

Variable		Datentyp	Werte	W	Beschreibung
					Fehler bei der automatischen Datenübernahme
	Bit 16 … Bit 31	-	-	-	Reserviert

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "ControlPanel" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.ControlPanel.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variable "InternalToTrace" (Positionierachse/Gleichlaufachse)

Die Variablenstruktur "<TO>.InternalToTrace.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Siehe auch

Auswerten des Technologie-Datenbausteins (Seite 1989)

Variablen des Technologieobjekts Externer Geber

Legende

Variable	Name der Variable
Datentyp	Datentyp der Variable
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Istwerte und Sollwerte (Externer Geber)

Die folgenden Variablen zeigen die Soll- und Istwerte des Technologieobjekts an.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
ActualPosition	LREAL	-	RON	Istposition
ActualVelocity	LREAL	-	RON	Istgeschwindigkeit
ActualAcceleration	LREAL	-	RON	Istbeschleunigung

Variable "Sensor" (Externer Geber)

Die Variablenstruktur "<TO>.Sensor.<Variablenname>" beinhaltet die steuerungsseitige Konfiguration des Gebers und die Konfiguration des aktiven und passiven Referenzierens.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
Sensor.	STRUCT			

Variable	Datentyp	Werte	W	Beschreibung	
Type	DINT	-	RES	Gebertyp	
				0	"INCREMENTAL" Inkrementell
				1	"ABSOLUTE" Absolut
				2	"CYCLIC_ABSOLUTE" Zyklisch absolut
InverseDirection	BOOL	-	RES	Invertierung des Istwerts	
				FALSE	Nein
				TRUE	Ja
System	DINT	-	RES	Gebersystem	
				0	"LINEAR" Linearer Geber
				1	"ROTATORY" Rotatorischer Geber
MountingMode	DINT	-	RES	Anbauart des Gebers	
				0	An der Motorwelle
				1	An der Lastseite
				2	Externes Messsystem
DataAdaption	DINT	-	RES	Automatische Übernahme der Antriebswerte Bezugsdrehzahl, maximale Drehzahl und Bezugsmoment im Gerät	
				0	Keine automatische Übernahme, händische Konfiguration der Werte
				1	Automatische Übernahme der im Antrieb konfigurierten Werte in die Konfiguration des Technologieobjekts
Interface.					
AddressIn	VREF	0 bis 65535	RON	Eingangsadresse für das PROFIdrive-Telegramm	
AddressOut	VREF	0 bis 65535	RON	Ausgangsadresse für das PROFIdrive-Telegramm	
Number	UDINT	1 bis 2	RON	Nummer des Gebers im Telegramm	
Parameter.					
Resolution	LREAL	-1.0E12 bis 1.0E12	RES	Auflösung eines linearen Gebers (Abstand zwischen zwei Geberstrichen)	
StepsPerRevolution	UDINT	1 bis 8388608	RES	Inkremente pro Geberumdrehung bei einem rotatorischen Geber	
FineResolutionXist1	UDINT	0 bis 31	RES	Anzahl Bits für die Feinauflösung Gx_XIST1 (zyklischer Geberistwert)	
FineResolutionXist2	UDINT	0 bis 31	RES	Anzahl Bits für die Feinauflösung Gx_XIST2 (Absolutwert des Gebers)	
DeterminableRevolutions	UDINT	0 bis 8388608	RES	Anzahl unterscheidbarer Geberumdrehungen bei einem Multiturn-Absolutwertgeber (Bei Singleturn-Absolutwertgeber = 1; bei Inkrementalgeber = 0)	
DistancePerRevolution	LREAL	0.0 bis 1.0E12	RES	Weg der Last pro Geberumdrehung bei einem extern montierten Geber	

Variable	Datentyp	Werte	W	Beschreibung	
PassiveHoming.	STRUCT				
Mode	DINT	-	RES	Referenziermodus	
				0	Nullmarke über PROFIdrive-Telegramm verwenden
				1	Nullmarke über PROFIdrive-Telegramm und Referenznocken verwenden
				2	Referenzmarke über Digitaleingang verwenden
SideInput	BOOL	-	CAL	Seite des digitalen Eingangs beim passiven Referenzieren	
				FALSE	Negative Seite
				TRUE	Positive Seite
Direction	DINT	-	CAL	Referenzierrichtung/Anfahrrichtung auf die Referenzmarke	
				0	Positive Referenzierrichtung
				1	Negative Referenzierrichtung
				2	Aktuelle Referenzierrichtung
DigitalInputAddress	VREF	0 bis 65535	RES	Adresse des digitalen Eingangs	
SwitchLevel	BOOL	-	RON	Signalpegel, der bei angefahrener Referenzmarke am digitalen Eingang ansteht	
				FALSE	Unterer Pegel
				TRUE	Oberer Pegel

Variable "Extrapolation" (Externer Geber)

Die Variablenstruktur "<TO>.Extrapolation.<Variablenname>" beinhaltet die Konfiguration der Istwertextrapolation.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
Extrapolation.	STRUCT			
LeadingAxisDependentTime	LREAL	-	RON	Anteil der Extrapolationszeit (bedingt durch Leitachse) Ergibt sich aus den folgenden Zeiten: • Zeit der Istwerterfassung an der Leitachse • Interpolator-Takt • Zeit des Istpositionsfilters der Leitachse (T1 + T2)
FollowingAxisDependentTime	LREAL	0.001 bis 1.0E12	DIR	Anteil der Extrapolationszeit (bedingt durch Folgeachse) Ergibt sich aus den folgenden Zeiten: • Für eine Folgeachse mit eingestellter Geschwindigkeitsvorsteuerung: – Kommunikationstakt – Interpolator-Takt – Drehzahl-Regelkreis-Ersatzzeit an der Folgeachse – Ausgabezeit des Sollwertes an der Folgeachse • Für eine Folgeachse ohne Geschwindigkeitsvorsteuerung: – Kommunikationstakt – Interpolator-Takt – Lage-Regelkreis-Ersatzzeit (1/Kv aus <TO>.PositionControl.Kv) – Ausgabezeit des Sollwertes an der Folgeachse
PositionFilter.	STRUCT			
T1	LREAL	0.001 bis 1.0E12	DIR	Positionsfilter Zeitkonstante T1
T2	LREAL	0.001 bis 1.0E12	DIR	Positionsfilter Zeitkonstante T2
VelocityFilter.	STRUCT			
T1	LREAL	0.001 bis 1.0E12	DIR	Geschwindigkeitsfilter Zeitkonstante T1
VelocityTolerance.	STRUCT			
Range	LREAL	0.001 bis 1.0E12	DIR	Toleranzbandbreite für die Geschwindigkeit
Hysteresis.	STRUCT			
Value	LREAL	0.001 bis 1.0E12	DIR	Hysteresewert

Variable "LoadGear" (Externer Geber)

Die Variablenstruktur "<TO>.LoadGear.<Variablenname>" beinhaltet die Konfiguration des Lastgetriebes.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Wertebereich	W	Beschreibung
LoadGear.	STRUCT			
Numerator	UDINT	1 bis 4294967295	RES	Lastgetriebe Zähler
Denominator	UDINT	1 bis 4294967295	RES	Lastgetriebe Nenner

Variable "Properties" (Externer Geber)

Die Variablenstruktur "<TO>.Properties.<Variablenname>" beinhaltet die Konfiguration des Achs- bzw. Bewegungstyps.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Wertebereich	W	Beschreibung	
Properties.	STRUCT				
MotionType	DINT	-	RON	Anzeige des Achs- bzw. Bewegungstyps	
				0	Lineare Achse bzw. Bewegung
				1	Rotatorische Achse bzw. Bewegung

Variable "Units" (Externer Geber)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
LengthUnit	UDINT	-	RON	Einheit für Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
				1004	rad
				1005	°
VelocityUnit	UDINT	-	RON	Einheit für Geschwindigkeit	
				1521	°/s
				1522	°/min
				1086	rad/s
				1523	rad/min
				1062	mm/s
				1061	m/s
				1524	mm/min
				1525	m/min
				1526	mm/h
				1063	m/h
				1527	km/min
				1064	km/h
				1066	in/s
				1069	in/min
				1067	ft/s
				1070	ft/min
				1075	mi/h
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1054	s

Variable "Mechanics" (Externer Geber)

Die Variablenstruktur "<TO>.Mechanics.<Variablenname>" beinhaltet die Konfiguration der Mechanik.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Wertebereich	W	Beschreibung
Mechanics.	STRUCT			
LeadScrew	LREAL	0.0 bis 1.0E12	RES	Spindelsteigung

Variable "Modulo" (Externer Geber)

Die Variablenstruktur "<TO>.Modulo.<Variablenname>" beinhaltet die Konfiguration der Modulofunktion.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung	
Modulo.	STRUCT				
Enable	BOOL	-	RES	FALSE	Moduloumrechnung deaktiviert
				TRUE	Moduloumrechnung aktiviert
				Bei aktivierter Moduloumrechnung wird auf Modulolänge > 0.0 geprüft.	
Length	LREAL	0.001 bis 1.0E12	RES	Modulolänge	
StartValue	LREAL	-1.0E12 bis 1.0E12	RES	Modulostartwert	

Variable "Homing" (Externer Geber)

Die Variablenstruktur "<TO>.Homing.<Variablenname>" beinhaltet die Konfiguration für das Referenzieren des Technologieobjekts.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
Homing.	STRUCT			
HomePosition	LREAL	-1.0E12 bis 1.0E12	CAL	Referenzpunktposition

Variable "StatusSensor" (Externer Geber)

Die Variablenstruktur "<TO>.StatusSensor.<Variablenname>" zeigt den Status des Messsystems an.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung	
StatusSensor.	STRUCT				
State	DINT	-	RON	Status des Geberistwerts	
				0	"NOT_VALID" Nicht gültig
				1	"WAITING_FOR_VALID" Warte auf Status "Gültig"
				2	"VALID" Gültig
CommunicationOK	BOOL	-	RON	Zyklische BUS-Kommunikation zwischen Steuerung und Geber	
				FALSE	Nicht aufgebaut
				TRUE	Aufgebaut
Error	BOOL	-	RON	FALSE	Kein Fehler im Messsystem
				TRUE	Fehler im Messsystem
AbsEncoderOffset	LREAL	-	RON	Referenzpunktverschiebung zum Wert eines Absolutwertgebers. Der Wert wird remanent in der CPU gespeichert.	
Control	BOOL	-	RON	FALSE	Geber ist nicht aktiv
				TRUE	Geber ist aktiv
Position	LREAL	-	RON	Geberposition	
Velocity	LREAL	-	RON	Gebergeschwindigkeit	
AdaptionState	DINT	-	RON	Status der automatischen Datenübernahme der Geberparameter	
				0	"NOT_ADAPTED" Daten nicht übernommen
				1	"IN_ADAPTION" Datenübernahme in Bearbeitung
				2	"ADAPTED" Datenübernahme abgeschlossen
				3	"NOT_APPLICABLE" Datenübernahme nicht angewählt, nicht möglich
				4	"ADAPTION_ERROR" Fehler bei der Datenübernahme

Variable "StatusExtrapolation" (Externer Geber)

Die Variablenstruktur "<TO>.StatusExtrapolation.<Variablenname>" zeigt den Status der Istwertextrapolation.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
Extrapolation.	STRUCT			
FilteredPosition	LREAL	-1.0E12 bis 1.0E12	RON	Position nach Positionsfilter
FilteredVelocity	LREAL	-1.0E12 bis 1.0E12	RON	Geschwindigkeit nach Toleranzband
ExtrapolatedPosition	LREAL	-1.0E12 bis 1.0E12	RON	Extrapolierte Position
ExtrapolatedVelocity	LREAL	-1.0E12 bis 1.0E12	RON	Extrapolierte Geschwindigkeit

Variable "StatusWord" (Externer Geber)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 5 "HomingDone") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung	
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"Enable" Freigabestatus	
				0	Technologieobjekt gesperrt
				1	Technologieobjekt freigegeben
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	Reserviert	
Bit 5	-	-	-	"HomingDone" Referenzierungsstatus	
				0	Technologieobjekt nicht referenziert
				1	Technologieobjekt referenziert
Bit 6	-	-	-	"Done"	
				0	Bewegungsauftrag in Bearbeitung bzw. Achssteuertafel aktiviert
				1	Kein Bewegungsauftrag in Bearbeitung und Achssteuertafel deaktiviert
Bit 7	-	-	-	Reserviert	
Bit 8	-		-	Reserviert	
Bit 9	-	-	-	Reserviert	
Bit 10	-	-	-	Reserviert	
Bit 11	-	-	-	"HomingCommand"	
				0	Kein "MC_Home"-Auftrag in Bearbeitung
				1	"MC_Home"-Auftrag in Bearbeitung
Bit 12 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Externer Geber)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	Reserviert
Bit 5	-	-	-	"SensorFault" Fehler im Gebersystem
Bit 6	-	-	-	Reserviert
Bit 7	-	-	-	"CommunicationFault" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8	-	-	-	Reserviert
Bit 9	-	-	-	Reserviert
Bit 10	-	-	-	"HomingError" Fehler beim Referenziervorgang Das Referenzieren kann nicht abgeschlossen werden.
Bit 11	-	-	-	Reserviert
Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Externer Geber)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2143)

Variable		Datentyp	Werte	W	Beschreibung	
ErrorDetail.		STRUCT				
	Number	UDINT	-	RON	Alarmnummer	
	Reaction	DINT	0, 10	RON	Wirksame Alarmreaktion	
					0	Keine Reaktion
					10	Freigabe wegnehmen

Variable "WarningWord" (Externer Geber)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 13 "PeripheralWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	Reserviert
Bit 5	-	-	-	"SensorWarning" Fehler im Gebersystem
Bit 6	-	-	-	Reserviert
Bit 7	-	-	-	"CommunicationWarning" Kommunikationsfehler Fehlende oder fehlerhafte Kommunikation.
Bit 8	-	-	-	Reserviert
Bit 9	-	-	-	Reserviert
Bit 10	-	-	-	"HomingWarning" Fehler beim Referenziervorgang Das Referenzieren kann nicht abgeschlossen werden.
Bit 11	-	-	-	Reserviert
Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variable "InternalToTrace" (Externer Geber)

Die Variablenstruktur "<TO>.InternalToTrace.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Variablen

Legende (Seite 2143)

Variable	Datentyp	Werte	W	Beschreibung
InternalToTrace.	ARRAY [1..4] OF STRUCT			
Id	DINT	-	DIR	-
Value	LREAL	-	RON	-

Variablen des Technologieobjekts Messtaster

Legende

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Anzeigedaten (Messtaster)

Die Variablen "<TO>.Status" und "<TO>.InputState" zeigen den Status der Messtasterfunktion und des Messtastereingangs.

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung	
Status	DINT	-	RON	Status Messtasterfunktion	
				0	"INACTIVE" Messung ist nicht aktiv.
				1	"WAITING_FOR_TRIGGER" Messtaster wartet auf Messereignis.
				2	"TRIGGER_OCCURRED" Ein oder mehrere Messwerte wurden erfasst.
				3	"MEASURING_ERROR" Fehler bei der Messung
InputState	BOOL	-	RON	Status des Messtastereingangs	
				FALSE	Messtastereingang nicht aktiv
				TRUE	Messtastereingang aktiv

Variable "Parameter" (Messtaster)

Die Variablenstruktur "<TO>.Parameter.<Variablenname>" beinhaltet die Konfiguration der Grundparameter des Technologieobjekts Messtaster.

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung	
Parameter.	STRUCT				
MeasuringInputType	DINT	-	RON	Messtastertyp	
				0	"TIME_BASED" Messung über Zeitstempel
				1	"PROFIDRIVE" Messereignis über PROFIdrive-Telegramm
PROFIdriveProbeNum-ber	UDINT	1, 2	RES	Nummer des zu verwendenden Messeingangs bei einer Messung über PROFIdrive-Telegramm	
MeasuringRangeActi-vationTime	LREAL	0.0 bis 1.0E12	RON	Systemanteil für die Aktivierungszeit des Messbereichs	
MeasuringRangeAdditi-onalActivationTime	LREAL	0.0 bis 1.0E12	RES	Additive Aktivierungszeit bei Verwendung von Messbe-reichsgrenzen [ms]	
CorrectionTime	LREAL	0.0 bis 1.0E12	RES	Korrekturzeit für das Messergebnis [ms]	

Variable "Interface" (Messtaster)

Die Variablenstruktur "<TO>.Interface.<Variablenname>" beinhaltet die Konfiguration der Eingabeeigenschaften für das Technologieobjekt Messtaster.

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung
Interface.	STRUCT			
Address	VREF	-	RON	Peripherieadresse für den digitalen Messeingang

Variable "Units" (Messtaster)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
LengthUnit	UDINT	-	RON	Einheit für Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
				1004	rad
				1005	°
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1056	ms

Variable "MeasuredValues" (Messtaster)

Die Variablenstruktur "<TO>.MeasuredValues.<Variablenname>" zeigt die Messergebnisse an.

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung
MeasuredValues.	STRUCT			
MeasuredValue1	LREAL	-1.0E12 bis 1.0E12	RON	Erster Messwert
MeasuredValue2	LREAL	-1.0E12 bis 1.0E12	RON	Zweiter Messwert
MeasuredValue1Counter	UDINT	0 bis 2147483647	RON	Zählwert für den ersten Messwert
MeasuredValue2Counter	UDINT	0 bis 2147483647	RON	Zählwert für den zweiten Messwert
LostEdgeCounter1	UDINT	0 bis 7	RON	LEC für Messwert 1 (Beim einmaligen Messen gleich null)
LostEdgeCounter2	UDINT	0 bis 7	RON	LEC für Messwert 2 (Beim einmaligen Messen gleich null)

Variable "StatusWord" (Messtaster)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "RestartActive") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung	
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"Control"	
				0	Technologieobjekt nicht in Betrieb
				1	Technologieobjekt in Betrieb
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	Reserviert	
Bit 5	-	-	-	"CommunicationOK" Kommunikation zwischen Messtaster und Messmodul	
				0	Nicht aufgebaut
				1	Aufgebaut
Bit 6 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Messtaster)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Messtaster)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0, 8	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				8	Messtasterbearbeitung beenden

Variable "WarningWord" (Messtaster)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 1 "ConfigWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13				"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variablen des Technologieobjekts Nocken

Legende

Variable	Name der Variable
Datentyp	Datentyp der Variable
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Anzeigedaten (Nocken)

Die Variable "<TO>.CamOutput" zeigt den Schaltzustand des Nockens an.

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
CamOutput	BOOL	-	RON	Schaltzustand des Nockens	
				FALSE	Nicht geschaltet
				TRUE	Geschaltet

Variable "Parameter" (Nocken)

Die Variablenstruktur "<TO>.Parameter.<Variablenname>" beinhaltet die Konfiguration der Grundparameter des Technologieobjekts Nocken.

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
Parameter.	STRUCT				
OutputCamType	DINT	-	RES	Nockentyp	
				0	Wegnocken
				1	Zeitnocken
PositionType	DINT	-	RES	Positionsbezug	
				0	Sollposition
				1	Istposition
OnCompensation	LREAL	0.0 bis 1.0E12	DIR	Aktivierungszeit Vorhaltezeit an der Einschaltflanke	
OffCompensation	LREAL	0.0 bis 1.0E12	DIR	Deaktivierungszeit Vorhaltezeit an der Ausschaltflanke	
Hysteresis	LREAL	0.0 bis 1.0E12	DIR	Hysteresewert Bei Nocken mit Bezug auf die Istposition wird empfohlen, einen Hysteresewert (> 0.0) einzugeben.	

Variable "Interface" (Nocken)

Die Variablenstruktur "<TO>.Interface.<Variablenname>" beinhaltet die Konfiguration der Ausgabeeigenschaften für das Technologieobjekt Nocken.

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
Interface.	STRUCT				
EnableOutput	BOOL	-	RES	Aktivierung der Nockenausgabe	
				FALSE	Ausgabe ist deaktiviert
				TRUE	Ausgabe ist aktiviert
Address	VREF	-	RON	Peripherieadresse des Nockens	
LogicOperation	DINT	-	RON	Logische Verknüpfung der Nockensignale am Ausgang	
				0	ODER-Verknüpfung
				1	UND-Verknüpfung

Variable "Units" (Nocken)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
LengthUnit	UDINT	-	RON	Einheit für Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
				1004	rad
				1005	°
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1056	ms

Variable "StatusWord" (Nocken)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "RestartActive") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"Control"	
				0	Technologieobjekt nicht in Betrieb
				1	Technologieobjekt in Betrieb
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"OutputInverted"	
				0	Nockenausgabe erfolgt nicht invertiert
				1	Nockenausgabe erfolgt invertiert
Bit 5	-	-	-	"CommunicationOK" Kommunikation zwischen Nocken und Ausgabemodul	
				0	Nicht aufgebaut
				1	Aufgebaut
Bit 6 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Nocken)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Nocken)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0, 6	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				6	Nockenbearbeitung wird beendet

Variable "WarningWord" (Nocken)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 1 "ConfigWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2164)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13				"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variablen des Technologieobjekts Nockenspur

Legende

Variable	Name der Variable
Datentyp	Datentyp der Variable
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Anzeigedaten (Nockenspur)

Die folgenden Variablen zeigen den Status der Nockenspur an.

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
Status	DINT	-	RON	0	"INACTIVE" (inaktiv)
				1	"ACTIVE" (aktiv)
				2	"ACTIVE_WAITING_FOR_NEXT_CYCLE" (aktiv und wartet auf nächste Spur)
TrackOutput	BOOL	-	RON	FALSE	Nockenspur wird nicht ausgegeben
				TRUE	Nockenspur wird ausgegeben
SingleCamState	DWORD	16#0 bis 16#FFFF_FFFF	RON	Eingeschalteter Nocken (Bitmaskiert)	
				0	Nocken ist nicht eingeschaltet
				1	Nocken ist eingeschaltet
TrackPosition	LREAL	-1.0E12 bis 1.0E12	RON	Anzeige der aktuellen Position innerhalb der Nockenspur	
MatchPosition	LREAL	-1.0E12 bis 1.0E12	RON	Bezugsposition der aktuellen Nockenspur Bei zyklischer Bearbeitung der Nockenspur wird die fortgeführte Bezugsposition der aktuellen Nockenspur angezeigt. Die eindeutige Erfassung und Ausgabe der Position ist nur möglich, wenn das zugeordnete Technologieobjekt in Bewegung ist.	

Variable "Parameter" (Nockenspur)

Die Variablenstruktur "<TO>.Parameter.<Variablenname>" beinhaltet die Konfiguration der Grundparameter des Technologieobjekts Nockenspur.

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
Parameter.	STRUCT				
CamTrackType	DINT	-	RES	Nockentyp	
				0	Wegnocken
				1	Zeitnocken
PositionType	DINT	-	RES	Positionsbezug	
				0	Sollposition
				1	Istposition
ReferencePosition	LREAL	-1.0E12 bis 1.0E12	DIR	Bezugsposition	
CamTrackLength	LREAL	0.001 bis 1.0E12	DIR	Spurlänge	
CamMasking	DWORD	16#0 bis 16#FFFF_FFFF	DIR	Bitmaskierung der Einzelnocken	
OnCompensation	LREAL	0.0 bis 1.0E12	DIR	Aktivierungszeit Vorhaltezeit an der Einschaltflanke	
OffCompensation	LREAL	0.0 bis 1.0E12	DIR	Deaktivierungszeit Vorhaltezeit an der Ausschaltflanke	
Hysteresis	LREAL	0.0 bis 1.0E12	DIR	Hysteresewert Bei Nocken mit Bezug auf die Istposition wird empfohlen, einen Hysteresewert (> 0.0) einzugeben.	
Cam[1..32].	STUCT				
OnPosition	LREAL	0.0 bis 1.0E12	CAL	Startposition (Wegnocken und Zeitnocken)	
Offposition	LREAL	0.0 bis 1.0E12	CAL	Endposition (Wegnocken)	
Duration	LREAL	0.001 bis 1.0E12	CAL	Einschaltdauer (Zeitnocken)	
Existent	BOOL	-	CAL	FALSE	Nocken wird nicht verwendet.
				TRUE	Nocken wird verwendet.

Variable "Interface" (Nockenspur)

Die Variablenstruktur "<TO>.Interface.<Variablenname>" beinhaltet die Konfiguration der Ausgabeeigenschaften für das Technologieobjekt Nockenspur.

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
Interface.	STRUCT				
EnableOutput	BOOL	-	RES	Nockenausgabe an dem unter "<TO>.Interface.Address" angegebenen Bit	
				FALSE	Keine Ausgabe
				TRUE	Ausgabe
Address	VREF	-	RON	Peripherieadresse für digitalen Nockenausgang	

Variable "Units" (Nockenspur)

Die Variablenstruktur "<TO>.Units.<Variablenname>" zeigt die eingestellten technologischen Einheiten.

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
LengthUnit	UDINT	-	RON	Einheit für Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
				1004	rad
				1005	°
TimeUnit	UDINT	-	RON	Einheit für Zeit	
				1056	ms

Variable "StatusWord" (Nockenspur)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "RestartActive") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"Control"	
				0	Technologieobjekt nicht in Betrieb
				1	Technologieobjekt in Betrieb
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"OutputInverted"	
				0	Nockenausgabe erfolgt nicht invertiert
				1	Nockenausgabe erfolgt invertiert
Bit 5	-	-	-	"CommunicationOK" Kommunikation zwischen Nockenspur und Ausgabemodul	
				0	Nicht aufgebaut
				1	Aufgebaut
Bit 6 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Nockenspur)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralError" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Nockenspur)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2170)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0, 7	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				7	Nockenspurbearbeitung wird beendet

Variable "WarningWord" (Nockenspur)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 1 "ConfigWarning") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variable

Legende (Seite 2158)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 12	-	-	-	Reserviert
Bit 13	-	-	-	"PeripheralWarning" Fehler beim Zugriff auf eine logische Adresse
Bit 14 … Bit 31	-	-	-	Reserviert

Variablen des Technologieobjekts Kurvenscheibe

Legende

Variable	Name der Variable
Datentyp	Datentyp der Variable
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".

W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Variable "Point" (Kurvenscheibe)

Die Variablenstruktur "<TO>.Point.<Variablenname>" beinhaltet die definierten Punkte der Kurvenscheibe.

Variablen

Legende (Seite 2176)

Variable		Datentyp	Werte	W	Beschreibung
Point.		ARRAY [1..1000] OF STRUCT			
	x	LREAL	-1.0E12 bis 1.0E12	CAL	Wert des Punkts im Definitionsbereich
	y	LREAL	-1.0E12 bis 1.0E12	CAL	Wert des Punkts im Wertebereich

Variable "ValidPoints" (Kurvenscheibe)

Die Variablenstruktur "<TO>.ValidPoint.<Variablenname>" zeigt die Gültigkeit der definierten Punkte der Kurvenscheibe an.

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung	
ValidPoint.	ARRAY [1..1000] OF BOOL				
ValidPoint	BOOL	-	CAL	Zeigt an, ob der definierte Punkt gültig ist.	
				FALSE	Nicht gültig
				TRUE	Gültig

Variable "Segment" (Kurvenscheibe)

Die Variablenstruktur "<TO>.Segment.<Variablenname>" beinhaltet die definierten Segmente der Kurvenscheibe.

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung
Segment.	ARRAY [1..50] OF STRUCT			
xmin	LREAL	-1.0E12 bis 1.0E12	CAL	Anfangskoordinate des Segments
xmax	LREAL	-1.0E12 bis 1.0E12	CAL	Endkoordinate des Segments
a0	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A0 für x^0 des Polynoms für das Segment
a1	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A1 für x^1 des Polynoms für das Segment
a2	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A2 für x^2 des Polynoms für das Segment
a3	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A3 für x^3 des Polynoms für das Segment
a4	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A4 für x^4 des Polynoms für das Segment
a5	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A5 für x^5 des Polynoms für das Segment
a6	LREAL	-1.0E12 bis 1.0E12	CAL	Koeffizient A6 für x^6 des Polynoms für das Segment
sineAmplitude	LREAL	-1.0E12 bis 1.0E12	CAL	Amplitude des Sinuselements
sinePeriod	LREAL	-1.0E12 bis 1.0E12	CAL	Periodenlänge des Sinuselements [rad]
sinePhase	LREAL	-1.0E12 bis 1.0E12	CAL	Phasenversatz des Sinuselements [rad]

Variable "ValidSegments" (Kurvenscheibe)

Die Variablenstruktur "<TO>.ValidSegment.<Variablenname>" zeigt die Gültigkeit der definierten Segmente der Kurvenscheibe an.

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung	
ValidSegment.	ARRAY [1..50] OF BOOL				
ValidSegment	BOOL	-	CAL	Zeigt an, ob das definierte Segment gültig ist.	
				FALSE	Nicht gültig
				TRUE	Gültig

Variable "InterpolationSettings" (Kurvenscheibe)

Die Variablenstruktur "<TO>.InterpolationSettings.<Variablenname>" beinhaltet die Konfiguration für die Interpolation der Kurvenscheibe.

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung	
InterpolationSettings.	STRUCT				
InterpolationMode	DINT	-1.0E12 bis 1.0E12	CAL	Interpolationsart	
				0	Linear
				1	C-Splines
				2	B-Splines
BoundaryConditions	DINT	-1.0E12 bis 1.0E12	CAL	Verhalten der Randpunkte	
				0	Keine Profilstart- und Profilendbedingungen
				1	Erste Ableitung am Profilanfang und -ende gleich

Variable "StatusCam" (Kurvenscheibe)

Die Variablenstruktur "<TO>.StatusCam.<Variablenname>" zeigt den Status der Kurvenscheibe an.

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung
StatusCam.	STRUCT			
StartLeadingValue	LREAL	-1.0E12 bis 1.0E12	RON	Erster definierter Stützpunkt/Beginn des ersten Segments der Kurvenscheibe (Anfangswert des Definitionsbereichs der Kurvenscheibe)
EndLeadingValue	LREAL	-1.0E12 bis 1.0E12	RON	Letzter definierter Stützpunkt/Ende des letzten Segments der Kurvenscheibe (Endwert des Definitionsbereichs der Kurvenscheibe)

Variable "StatusWord" (Kurvenscheibe)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 4 "CamDataChanged") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung	
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts	
Bit 0	-	-	-	"Control" Verwendungsstatus	
				0	Kurvenscheibe nicht in Verwendung
				1	Kurvenscheibe in Verwendung
Bit 1	-	-	-	"Error"	
				0	Kein Fehler vorhanden
				1	Fehler vorhanden
Bit 2	-	-	-	"RestartActive"	
				0	Kein Restart aktiv
				1	Restart aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	Restart-Variablen unverändert
				1	Änderung an Restart-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"CamDataChanged"	
				0	Keine Änderung
				1	Die Definition der Kurvenscheibe hat sich im Technologie-Datenbaustein geändert.
Bit 5	-	-	-	"Interpolated"	
				0	Kurvenscheibe ist nicht interpoliert
				1	Kurvenscheibe ist interpoliert
Bit 6	-	-	-	"InInterpolation"	
				0	Kurvenscheibe nicht in Interpolation
				1	Kurvenscheibe in Interpolation
Bit 7 … Bit 31	-	-	-	Reserviert	

Variable "ErrorWord" (Kurvenscheibe)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 31	-	-	-	Reserviert

Variable "ErrorDetail" (Kurvenscheibe)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0, 9	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion
				9	Bearbeitung des Technologieobjekts beenden

Variable "WarningWord" (Kurvenscheibe)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten (Seite 1991)".

Variablen

Legende (Seite 2176)

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemWarning" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigWarning" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserWarning" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4 … Bit 31	-	-	-	Reserviert

Technologie-Alarme

Übersicht der Technologie-Alarme

Folgende Tabelle zeigt eine Übersicht über die Technologie-Alarme und die entsprechenden Alarmreaktionen. Werten Sie beim Auftreten eines Technologie-Alarms den gesamten angezeigten Alarmtext aus, um die genaue Ursache zu finden.

Legende

Nr.	Nummer des Technologie-Alarms (entspricht <TO>.ErrorDetail.Number)
Reaktion	Wirksame Alarmreaktion (entspricht <TO>.ErrorDetail.Reaction)
Fehlerbit	Beim Auftreten des Technologie-Alarms gesetztes Bit in "<TO>.ErrorWord" Eine Beschreibung der Bits finden Sie im Anhang (Seite 2137).
Warnungsbit	Beim Auftreten des Technologie-Alarms gesetztes Bit in "<TO>.WarningWord" Eine Beschreibung der Bits finden Sie im Anhang (Seite 2140).
Restart	Zum Quittieren des Technologie-Alarms muss das Technologieobjekt neu initialisiert werden (Restart).
Diagnosepuffer	Der Alarm wird im Diagnosepuffer eingetragen.
Alarmtext	Angezeigter Alarmtext (eingeschränkt)

Liste der Technologie-Alarme

Nr.	Reaktion	Fehler-bit	Warnungs-bit	Restart	Diagnose-puffer	Alarmtext
101	Freigabe wegnehmen	X1	-	X	X	Konfigurationsfehler.
102	Freigabe wegnehmen	X15	-	X	X	Fehler Adaption Antriebskonfiguration.
103	Freigabe wegnehmen	X15	-	X	X	Fehler Adaption Geberkonfiguration.
104	Stopp mit maximalen Dynamikwerten	X1	-	-	-	Fehler Angabe SW-Endschalter.
105	Freigabe wegnehmen	X1	-	X	X	Konfigurationsfehler Antrieb.
106	Freigabe wegnehmen	X1	-	-	X	Konfigurationsfehler Antriebsanbindung.
107	Freigabe wegnehmen	X1	-	X	X	Konfigurationsfehler Geber.
108	Freigabe wegnehmen	X1	-	-	X	Konfigurationsfehler Geberanbindung.
109	Freigabe wegnehmen	X1	-	X	-	Konfigurationsfehler.
110	Keine Reaktion	-	X1	-	-	Konfiguration wird intern angepasst.
111	Keine Reaktion	-	X15	-	X	TO- und Antriebskonfiguration inkonsistent.
112	Keine Reaktion	-	X15	-	X	TO- und Geberkonfiguration inkonsistent.
113	Freigabe wegnehmen	X2	-	X	-	Taktsynchroner Betrieb nicht möglich.
201	Freigabe wegnehmen	X0	-	X	X	Interner Fehler.
202	Keine Reaktion	X0	-	X	-	Interner Konfigurationsfehler.
203	Freigabe wegnehmen	X0	-	X	-	Interner Fehler.
204	Freigabe wegnehmen	X0	-	-	-	Inbetriebnahmefehler.
304	Stopp mit Notstopp-Rampe	X2	-	-	-	Grenzwert der Geschwindigkeit ist null.
305	Stopp mit Notstopp-Rampe	X2	-	-	-	• Grenzwert der Beschleunigung ist null. • Grenzwert der Verzögerung ist null.
306	Stopp mit Notstopp-Rampe	X2	-	-	-	Grenzwert des Rucks ist null.
307	Stopp mit maximalen Dynamikwerten	X2	-	-	X	• Negativer numerischer Wertebereich der Position erreicht. • Positiver numerischer Wertebereich der Position erreicht.
308	Freigabe wegnehmen	X2	-	-	X	• Negativer numerischer Wertebereich der Position überschritten. • Positiver numerischer Wertebereich der Position überschritten.
321	Stopp mit Notstopp-Rampe	X3	-	-	-	Achse nicht referenziert.
322	Keine Reaktion	-	X3	-	-	Restart nicht ausgeführt.
323	Freigabe wegnehmen	X3	-	-	-	MC_Home konnte nicht durchgeführt werden.
341	Stopp mit maximalen Dynamikwerten	X10	-	-	-	Referenzierdaten fehlerhaft.
342	Stopp mit Notstopp-Rampe	X10	-	-	-	Referenznocken/Gebernullmarke nicht gefunden.
343	Freigabe wegnehmen	X1	-	-	-	Funktion Referenzieren wird durch das Gerät nicht unterstützt.

Nr.	Reaktion	Fehler-bit	Warnungs-bit	Restart	Diagnose-puffer	Alarmtext
401	Freigabe wegnehmen	X13	-	-	X	Fehler bei Zugriff auf log. Adresse.
411	Freigabe wegnehmen	X5	-	-	X	Geber unter der log. Adresse gestört.
412	Freigabe wegnehmen	X5	-	-	-	Zulässiger Istwertbereich überschritten.
421	Freigabe wegnehmen	X4	-	-	X	Antrieb unter der log. Adresse gestört.
431	Freigabe wegnehmen	X7	-	-	X	Kommunikation zum Gerät log. Adresse gestört.
501	Keine Reaktion	-	X6	-	-	Programmierte Geschwindigkeit wird begrenzt.
502	Keine Reaktion	-	X6	-	-	• Programmierte Beschleunigung wird begrenzt. • Programmierte Verzögerung wird begrenzt.
503	Keine Reaktion	-	X6	-	-	Programmierter Ruck wird begrenzt.
504	Keine Reaktion	-	X6	-	-	Drehzahlsollwertüberwachung aktiv.
511	Keine Reaktion	-	X6	-	-	Dynamikgrenze wird durch die Kinematikbewegung verletzt.
521	Freigabe wegnehmen	X11	-	-	-	Schleppfehler.
522	Keine Reaktion	-	X11	-	-	Warnung Schleppfehlertoleranz.
531	Freigabe wegnehmen	X9	-	-	-	• Positiver HW-Endschalter angefahren. • Negativer HW-Endschalter angefahren. • Unzulässige Freifahrrichtung aktiver HW-Endschalter. • HW-Endschalter verpolt, kein Freifahren möglich. • Beide HW-Endschalter aktiv, kein Freifahren möglich.
533	Stopp mit maximalen Dynamikwerten	X8	-	-	-	• Negativer SW-Endschalter angefahren. • Positiver SW-Endschalter angefahren.
534	Freigabe wegnehmen	X8	-	-	-	• Negativer SW-Endschalter wird überfahren. • Positiver SW-Endschalter wird überfahren.
541	Freigabe wegnehmen	X12	-	-	-	Fehler Positionierüberwachung.
542	Freigabe wegnehmen	X2	-	-	-	Fehler Klemmungsüberwachung: Achse verlässt Klemmungstoleranzfenster.
550	Sollwerte nachführen	X4	-	-	-	Antriebsautarke Bewegung wird ausgeführt.
551	Keine Reaktion	X2	X6	-	-	Max. Geschwindigkeit mit Antriebs-/Achsparametern nicht erreichbar.
552	Freigabe wegnehmen	X15	-	-	-	Adaptionsfehler des Gebers bei Hochlauf.
601	Stopp mit maximalen Dynamikwerten	X14	-	-	-	Leitachse nicht zugeordnet oder fehlerhaft.
603	Freigabe wegnehmen	X14	-	-	-	Leitachse befindet sich nicht im lagegeregelten Betrieb.
608	Stopp mit maximalen Dynamikwerten	X14	-	-	-	Fehler beim Aufsynchronisieren.

Nr.	Reaktion	Fehler-bit	Warnungs-bit	Restart	Diagnose-puffer	Alarmtext
611	Freigabe wegnehmen	X2	-	-	-	Die am Parameter MC_CamIn.Cam angegebene Kurvenscheibe wurde nicht konfiguriert bzw. ist nicht verfügbar oder ist nicht interpoliert.
612	Freigabe wegnehmen	X2	-	-	-	Angegebene Kurvenscheibe wurde nicht interpoliert.
700	Freigabe wegnehmen	X2	-	-	-	Fehler bei der Nockenbegrenzung.
701	Freigabe wegnehmen	X13	-	-	-	Peripherieausgabefehler.
702	Freigabe wegnehmen	X2	-	-	-	Positionswert ungültig.
703	Freigabe wegnehmen	X2	-	-	-	Nockenspurdaten fehlerhaft.
704	Freigabe wegnehmen	X2	-	-	-	Nockendaten fehlerhaft.
750	Freigabe wegnehmen	X2	-	-	-	Messauftrag beim Referenzieren der zugeordneten Achse nicht möglich.
752	Freigabe wegnehmen	X2	-	-	-	Gültigkeitsbereich des Messauftrags nicht erkannt.
753	Freigabe wegnehmen	X2	-	-	-	Zeitgleich kann nur ein Messtaster auf einen Geber zugreifen.
754	Freigabe wegnehmen	X2	-	-	-	Messeingangsprojektierung in externem Gerät nicht korrekt.
755	Freigabe wegnehmen	X13	-	-	-	Messauftrag nicht möglich.
758	Keine Reaktion	X2	-	-	-	Eine Messflanke wurde nicht ausgewertet.

Siehe auch

Technologie-Alarme (Seite 2033)

Technologie-Alarme 101-113

Technologie-Alarm 101

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler.	
Wert in <Variable> unzulässig.	Passen Sie den angegebenen Wert an.
Lastgetriebefaktoren fehlerhaft.	Passen Sie die Lastgetriebefaktoren in den Parametern "<TO>.LoadGear.Numerator" bzw. "<TO>.LoadGear.Denominator" an.
Mindestens ein Geber notwendig. Sensor[].existent	Konfigurieren Sie mindestens einen Geber.
Für DSC muss Sensor[1] konfiguriert sein.	Konfigurieren Sie Sensor[1].
Werte in Sensor.Parameter.FineResolutionXist1 und P979 ungleich.	Stellen Sie am Technologieobjekt die gleiche Feinauflösung wie am Antrieb ein.
Reglerparameter fehlerhaft.	Passen Sie den Wert des Parameters "<TO>.PositionController.Kv" an.
PROFIBUS-Parametrierung inkonsistent; Summe Ti und To größer Sendetakt.	Passen Sie den Sendetakt in der Hardware-Konfiguration an.
Antrieb oder Antriebstelegrammtyp bzw. Geber nicht für DSC geeignet.	Prüfen Sie, ob der Antrieb mit DSC betrieben werden kann und passen Sie gegebenenfalls das Antriebstelegramm an.
Parameter TimeOut außerhalb der Grenzen.	Setzen Sie die Überwachungszeit der Achssteuertafel auf einen gültigen Wert.
Parameter Simulation.Mode außerhalb der Grenzen.	Setzen Sie den Parameter auf einen gültigen Wert.
Telegramm in Actor.Interface.AddressIn und AddressOut ungleich.	Stellen Sie für Sende- und Empfangsrichtung den gleichen Antriebstelegrammtyp ein.
Unzulässige Kombination für Referenzierdaten inkrement. Geber.	Prüfen Sie die Einstellungen zum aktiven und passiven Referenzieren.
Telegramm in Sensor.Interface.AddressIn und AddressOut ungleich.	Stellen Sie für Sende- und Empfangsrichtung den gleichen Gebertelegrammtyp ein.

Technologie-Alarm 102

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Fehler Adaption Antriebskonfiguration.	
Antrieb ist keinem SINAMICS-Gerät zugeordnet.	Die Antriebsadaption ist nur bei SINAMICS-Antrieben verfügbar.
Antrieb nicht direkt auf I/O-Bereich verschaltet.	Bei der Konfiguration der Achse wurden die log Adressen z. B. auf einen Datenbaustein- bzw. Merkerbereich gelegt. Die Adaption ist nur möglich, wenn der Geber direkt auf einen I/O-Bereich verschaltet wurde.
Abbruch Adaption wegen Ressourcenmangel.	Prüfen Sie, ob Ihr Gerät die azyklische Datenkommunikation nach PROFIdrive unterstützt.
Parameter inexistent, Wert unlesbar oder unzulässig.	
Maximaldrehzahl	
Maximalmoment/-kraft (P1520)	
Maximalmoment/-kraft (P1521)	
Auflösung Moment	
Nenndrehzahl	
Nennmoment	
Motortyp	

Technologie-Alarm 103

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Fehler Adaption Geberkonfiguration.	
Geber ist keinem SINAMICS-Gerät zugeordnet.	Die Geberadaption ist nur für SINAMICS-Geräte und externe Siemens Geber verfügbar.
Geber nicht direkt auf I/O-Bereich verschaltet.	Bei der Konfiguration der Achse wurden die log Adressen z. B. auf einen Datenbaustein- bzw. Merkerbereich gelegt. Die Adaption ist nur möglich, wenn der Geber direkt auf einen I/O-Bereich verschaltet wurde.
Abbruch Adaption wegen Ressourcenmangel.	Prüfen Sie, ob Ihr Gerät die azyklische Datenkommunikation nach PROFIdrive unterstützt.
Parameter inexistent, Wert unlesbar oder unzulässig.	
Gebersystem	
Geberauflösung	
Geberfeinauflösung Gx_XIST1	
Geberfeinauflösung Gx_XIST2	
Geberumdrehungen	

Technologie-Alarm 104

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler Angabe SW-Endschalter.	
Neg. SW-Endschalter größer pos. SW-Endschalter.	Ändern Sie die Position der Software-Endschalter.

Technologie-Alarm 105

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler Antrieb.	
HW-Konfiguration.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie den topologischen Aufbau des Projekts. • Vergleichen Sie die Gerätekonfiguration und die Konfiguration des Technologieobjekts. • Wenden Sie sich an den Kundendienst.
TO benötigt kleineren Servo-Takt.	
Fehler in der internen Kommunikation.	• Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung. • Wenden Sie sich an den Kundendienst.
Adresse für Antriebsdaten im Projekt nicht vorhanden.	Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung.

Technologie-Alarm 106

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler Antriebsanbindung.	
System hat keine Kommunikation zum Antrieb.	Interner Systemfehler. • Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung. • Wenden Sie sich an den Kundendienst.
Antrieb im Hochlauf nicht initialisiert.	• Stellen Sie sicher, dass die Kommunikation zwischen Steuerung und Antrieb aufgebaut ist. Werten Sie dazu vor dem Freigeben der Achse den Parameter "<TO>.StatusDrive.CommunicationOK" aus. • Um ein Technologieobjekt freizugeben, muss der Antrieb fertig initialisiert sein. Setzen Sie den Auftrag später erneut ab.

Technologie-Alarm 107

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler Geber.	
HW-Konfiguration TO benötigt kleineren Servo-Takt.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie den topologischen Aufbau des Projekts. • Vergleichen Sie die Gerätekonfiguration und die Konfiguration des Technologieobjekts. • Wenden Sie sich an den Kundendienst.
Fehler interne Kommunikation.	• Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung. • Wenden Sie sich an den Kundendienst.

Technologie-Alarm 108

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler Geberanbindung.	
System ohne Kommunikation zum Geber.	Interner Systemfehler. • Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung. • Wenden Sie sich an den Kundendienst.
Geber im Hochlauf nicht initialisiert.	• Stellen Sie sicher, dass die Kommunikation zwischen Steuerung und Geber aufgebaut ist. Werten Sie dazu vor dem Freigeben der Achse den Parameter "<TO>.StatusSensor[1..4].CommunicationOK" aus und prüfen Sie zusätzlich, ob der Status des Geberistwerts "<TO>.StatusSensor[1..4].State" = VALID (2) ist. • Um ein Technologieobjekt freizugeben, muss der Geber fertig initialisiert sein. Setzen Sie den Auftrag später erneut ab.
Adresse Geberdaten fehlt im Projekt.	Prüfen Sie das Projekt auf Konsistenz und laden Sie das Projekt erneut in die Steuerung.

Technologie-Alarm 109

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler.	
Neg. HW-Endschalter.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie den topologischen Aufbau des Projekts. • Vergleichen Sie die Gerätekonfiguration und die Konfiguration des Technologieobjekts. • Wenden Sie sich an den Kundendienst.
Pos. HW-Endschalter.	
Referenznocken Aktives Referenzieren.	
Referenznocken Passives Referenzieren.	
Enable-Bit der analogen Antriebsschnittstelle.	
Driveready-Bit der analogen Antriebsschnittstelle.	
Messtastereingang fehlerhaft.	
Nockenausgang fehlerhaft.	

Technologie-Alarm 110

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Konfiguration wird intern angepasst.	
Actor.DriveParameter.MaxSpeed wird begrenzt.	• Korrigieren Sie den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf "<TO>.Actor.MaxSpeed" / 2. • Korrigieren Sie bei analoger Antriebsanbindung den Bezugswert im Antrieb sowie in der Konfiguration des Technologieobjekts auf "<TO>.Actor.MaxSpeed" / 1,17. • Der Wert kann im Antrieb beispielsweise im p2000 = p1082 eingestellt werden.
PositioningMonitoring.ToleranceTime wird begrenzt.	Ändern Sie das Konfigurationsdatum.
DynamicDefaults.EmergencyDeceleration wird begrenzt.	
DriveParameter.ReferenceTorque zu klein.	

Technologie-Alarm 111

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
TO- und Antriebskonfiguration inkonsistent.	
Telegramm unterschiedlich.	Gleichen Sie die Telegrammkonfiguration am Technologieobjekt mit der Telegrammkonfiguration im Antrieb ab. (P922 im Antrieb)
Momentenauflösung inkompatibel.	Stellen Sie am Antrieb die hohe Momentenauflösung ein.
Applikationszyklus des Antriebs und Servotakt ungleich.	Passen Sie den Applikationszyklus des Antriebs in der Gerätekonfiguration für den PROFIBUS-Antrieb an.
Applikationszyklus des Antriebs und Bearbeitungstakt des TOs ungleich.	
Linearmotor konfiguriert.	Stellen Sie im Antrieb Rundmotor (P300) ein.

Technologie-Alarm 112

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
TO- und Geberkonfiguration inkonsistent.	
Telegrammtyp unterschiedlich.	Gleichen Sie die Telegrammkonfiguration am Technologieobjekt mit der Telegrammkonfiguration im Geber ab.
Geber ist kein Absolutwertgeber.	Konfigurieren Sie den Geber am Technologieobjekt als Inkrementalgeber.
Applikationszyklus des Gebers und Servotakt ungleich.	Passen Sie den Applikationszyklus des Gebers in der Gerätekonfiguration für den PROFIBUS-Geber an.
Applikationszyklus des Gebers und Bearbeitungstakt des TOs ungleich.	

Technologie-Alarm 113

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Taktsynchroner Betrieb nicht möglich.	• Der konfigurierte Ausgang am Technologieobjekt Nocken, Nockenspur bzw. der Eingang am Technologieobjekt Messtaster kann nicht taktsynchron verwendet werden. Konfigurieren Sie die Peripherie in der Gerätekonfiguration taktsynchron. • Stellen Sie sicher, dass der Organisationsbaustein MC_Servo [OB91] synchron zum Bussystem aufgerufen wird.

Technologie-Alarme 201-204

Technologie-Alarm 201

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Interner Fehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 202

Alarmreaktion: keine Reaktion

Restart: erforderlich

Alarmtext	Abhilfe
Interner Konfigurationsfehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 203

Alarmreaktion: Freigabe wegnehmen

Restart: erforderlich

Alarmtext	Abhilfe
Interner Fehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 204

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Inbetriebnahmefehler.	
Verbindung zum TIA Portal unterbrochen.	Prüfen Sie die Verbindungseigenschaften.

Technologie-Alarme 304-343

Technologie-Alarm 304

Alarmreaktion: Stopp mit Notstopp-Rampe

Restart: nicht erforderlich

Alarmtext	Abhilfe
Grenzwert der Geschwindigkeit ist null.	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die maximale Geschwindigkeit (DynamicLimits.MaxVelocity) ein.

Technologie-Alarm 305

Alarmreaktion: Stopp mit Notstopp-Rampe

Restart: nicht erforderlich

Alarmtext		Abhilfe
Grenzwert der Beschleunigung/Verzögerung ist null.		
	Beschleunigung	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die maximale Beschleunigung (DynamicLimits.MaxAcceleration) ein.
	Verzögerung	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die maximale Verzögerung (DynamicLimits.MaxDeceleration) ein.

Technologie-Alarm 306

Alarmreaktion: Stopp mit Notstopp-Rampe

Restart: nicht erforderlich

Alarmtext	Abhilfe
Grenzwert des Rucks ist null.	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für den maximalen Ruck (DynamicLimits.MaxJerk) ein.

Technologie-Alarm 307

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext		Abhilfe
Negativer/Positiver numerischer Wertebereich der Position erreicht.		
	Negativer	Aktivieren Sie die Einstellung "Modulo" für das Technologieobjekt.
	Positiver	

Technologie-Alarm 308

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Negativer/Positiver numerischer Wertebereich der Position überschritten.	
Negativer	Aktivieren Sie die Einstellung "Modulo" für das Technologieobjekt.
Positiver	

Technologie-Alarm 321

Alarmreaktion: Stopp mit Notstopp-Rampe

Restart: nicht erforderlich

Alarmtext	Abhilfe
Achse nicht referenziert.	Um eine absolute Positionierbewegung auszuführen, müssen Sie das Technologieobjekt referenzieren.

Technologie-Alarm 322

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Restart nicht ausgeführt.	
Das Technologieobjekt ist für den Restart nicht bereit.	Laden Sie das Projekt neu.
Die Bedingung für den Restart des Technologieobjekts ist nicht erfüllt.	Sperren Sie das Technologieobjekt.

Technologie-Alarm 323

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
MC_Home konnte nicht durchgeführt werden.	• Aktivieren Sie die Einstellung "Modulo" für das Technologieobjekt. • Passen Sie bei Verwendung der Motion Control-Anweisung "MC_Home" den Positionswert an.

Technologie-Alarm 341

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext	Abhilfe
Referenzierdaten fehlerhaft.	
Anfahrgeschwindigkeit ist null.	Prüfen Sie die Konfiguration für das Referenzieren (Homing.ApproachVelocity).
Referenziergeschwindigkeit ist null.	Prüfen Sie die Konfiguration für das Referenzieren (Homing.ReferencingVelocity).

Technologie-Alarm 342

Alarmreaktion: Stopp mit Notstopp-Rampe

Restart: nicht erforderlich

Alarmtext	Abhilfe
Referenznocken/Gebernullmarke nicht gefunden.	Der zum Referenzieren konfigurierte Referenznocken wurde im Verfahrbereich der Achse nicht gefunden.

Technologie-Alarm 343

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Funktion Referenzieren wird durch das Gerät nicht unterstützt.	Konfigurieren Sie In den Eigenschaften der C-CPU einen Referenzschalter-Eingang für den verwendeten Impulsgeneratorausgang. ("Impulsgeneratoren (PTO/PWM) > PTO[n]/PWN[n] > Hardwareein-/ausgänge") Beim Referenzieren über eine Nullmarke überträgt die CPU den Referenzschalter-Eingang als Nullmarke.

Technologie-Alarme 401-431

Technologie-Alarm 401

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler bei Zugriff auf log. Adresse.	
Adresse ungültig.	• Schließen Sie ein geeignetes Gerät an. • Prüfen Sie das Gerät (Anschlüsse). • Prüfen Sie den topologischen Aufbau des Projekts. • Vergleichen Sie die Gerätekonfiguration und die Konfiguration des Technologieobjekts. • Wenden Sie sich an den Kundendienst.
Input-Adresse ungültig.	
Output-Adresse ungültig.	

Technologie-Alarm 411

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Geber unter log. Adresse gestört.	
Störungsmeldung Geber.	Prüfen Sie die Funktion, Verbindungen und Anschlüsse des Gebers.
HW-Fehler Geber.	
Geber verschmutzt.	
Lesefehler Geber-Absolutwert.	Vergleichen Sie den Gebertyp im Antriebs- bzw. Geberparameter P979 mit den Konfigurationsdaten des Technologieobjekts.
Nullmarkenüberwachung Geber.	Geber meldet Fehler bei der Nullmarkenüberwachung (Störcode 0x0002 im Gx_XIST2, siehe PROFIdrive-Profil).
Geber im Zustand Parken.	• Suchen Sie die Fehlerursache im angeschlossenen Antrieb bzw. Geber. • Prüfen Sie, ob der Alarm eventuell infolge einer Inbetriebnahmehandlung am Antrieb bzw. Geber ausgelöst wurde.

Technologie-Alarm 412

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Zulässiger Istwertbereich überschritten.	
Positiv.	Referenzieren Sie die Achse/den Geber in einen gültigen Istwertbereich.
Negativ.	
Modulolänge.	Passen Sie die Modulolänge dem verwendeten Geber an.

Technologie-Alarm 421

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Antrieb unter log. Adresse gestört.	
Störungsmeldung Antrieb.	• Funktion und Anschlüsse des Antriebs prüfen. • Sicherheitsfunktion im Antrieb freigeben und quittieren. • Prüfen Sie bei analog angebundenen Achsen, ob die Variable "<TO>.StatusDrive.InOperation" = TRUE ist.
Keine Antriebsführung gefordert.	
Antrieb hat sich abgeschaltet.	
Antriebsfreigabe nicht möglich.	

Technologie-Alarm 431

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Kommunikation zum Gerät log. Adresse gestört.	
Antrieb ausgefallen.	Prüfen Sie die Funktion, Verbindungen und Anschlüsse des Antriebs.
Lebenszeichen Antrieb gestört.	• Prüfen Sie die Funktion, Verbindungen und Anschlüsse des Antriebs. • Vergleichen Sie die Taktparameter in der Gerätekonfiguration (PROFIBUS-Strang, Slave-OM für Antrieb bzw. Geber) und des Ablaufsystems. Tmapc und Servo müssen auf die gleiche Taktperiodenzeit parametriert sein. (Fehlparametrierung wird durch Grund 0x0080 angezeigt.) • Wenn Sie den Applikationszyklus des MC-Servo [OB91] untersetzt zum Sendetakt eines PROFINET IO-Systems aufrufen und wiederholt der Technologie-Alarm 431 (Lebenszeichen Antrieb gestört.) angezeigt wird, erhöhen Sie die Aktualisierungszeit des Sendetakts.
Geber ausgefallen.	Prüfen Sie die Funktion, Verbindungen und Anschlüsse des Gebers.
Lebenszeichen Geber gestört.	• Prüfen Sie die Funktion, Verbindungen und Anschlüsse des Gebers. • Vergleichen Sie die Taktparameter in der Gerätekonfiguration (PROFIBUS-Strang, Slave-OM für Antrieb bzw. Geber) und des Ablaufsystems. Tmapc und Servo müssen auf die gleiche Taktperiodenzeit parametriert sein.

Technologie-Alarme 501-552

Technologie-Alarm 501

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierte Geschwindigkeit wird begrenzt.	• Prüfen Sie den Wert für die Geschwindigkeit an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 502

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierte Beschleunigung/Verzögerung wird begrenzt.	
Beschleunigung	• Prüfen Sie den Wert für die Beschleunigung an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.
Verzögerung	• Prüfen Sie den Wert für die Verzögerung an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 503

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierter Ruck wird begrenzt.	• Prüfen Sie den Wert für den Ruck an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 504

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Drehzahlsollwertüberwachung aktiv.	• Prüfen Sie den mechanischen Aufbau. • Prüfen Sie den Geberanschluss. • Prüfen Sie die Konfiguration der Drehzahlsollwertschnittstelle. • Prüfen Sie die Konfiguration des Regelkreises. • Prüfen Sie den Wert für die maximale Geschwindigkeit (<TO>.DynamicLimits.MaxVelocity).

Technologie-Alarm 511

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Dynamikgrenze wird durch die Kinematikbewegung verletzt.	
Geschwindigkeit	Reduzieren Sie die Geschwindigkeit der Kinematikbewegung.
Beschleunigung	Reduzieren Sie die Beschleunigung der Kinematikbewegung.
Verzögerung	Reduzieren Sie die Verzögerung der Kinematikbewegung.

Technologie-Alarm 521

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Schleppfehler.	• Prüfen Sie die Konfiguration des Regelkreises. • Prüfen Sie den Richtungssinn des Gebers. • Prüfen Sie die Konfiguration der Schleppfehlerüberwachung.

Technologie-Alarm 522

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Warnung Schleppfehlertoleranz.	• Prüfen Sie die Konfiguration des Regelkreises. • Prüfen Sie den Richtungssinn des Gebers. • Prüfen Sie die Konfiguration der Schleppfehlerüberwachung.

Technologie-Alarm 531

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext		Abhilfe
	Pos. HW-Endschalter angefahren.	Quittieren Sie den Alarm. Nach der Quittierung sind Bewegungen in negativer Richtung zugelassen.
	Neg. HW-Endschalter angefahren.	Quittieren Sie den Alarm. Nach der Quittierung sind Bewegungen in positiver Richtung zugelassen
	Unzulässige Freifahrrichtung aktiver HW-Endschalter.	Die programmierte Bewegungsrichtung ist wegen des aktiven Hardware-Endschalters gesperrt. Fahren Sie die Achse in entgegengesetzter Richtung frei.
	HW-Endschalter verpolt, kein Freifahren möglich.	• Prüfen Sie den mechanischen Aufbau der Hardware-Endschalter. • Prüfen Sie die Endschalter. • Der Fehler kann durch Aus- und Einschalten der Steuerung bzw. "MC_Reset" mit "Restart" = TRUE quittiert werden.
	Beide HW-Endschalter aktiv, kein Freifahren möglich.	

Technologie-Alarm 533

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext		Abhilfe
Negativer/Positiver SW-Endschalter angefahren.		
	Negativer	Der Software-Endschalter wurde aus einer laufenden Bewegung angefahren. Fahren Sie die Achse in positiver Richtung weg vom Software-Endschalter.
	Positiver	Der Software-Endschalter wurde aus einer laufenden Bewegung angefahren. Fahren Sie die Achse in negativer Richtung weg vom Software-Endschalter.

Technologie-Alarm 534

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Negativer/Positiver SW-Endschalter wird überfahren.	
Negativer	Der Software-Endschalter wurde überfahren. Quittieren Sie den Alarm. Nach der Quittierung sind Bewegungen in positiver Richtung zugelassen.
Positiver	Der Software-Endschalter wurde überfahren. Quittieren Sie den Alarm. Nach der Quittierung sind Bewegungen in negativer Richtung zugelassen.

Technologie-Alarm 541

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler Positionierüberwachung.	
Zielbereich nicht erreicht.	Der Zielbereich wurde nicht innerhalb der Toleranzzeit erreicht. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.
Zielbereich wieder verlassen.	Der Zielbereich wurde innerhalb der minimalen Verweildauer verlassen. • Prüfen Sie die Konfiguration der Positionierüberwachung. • Prüfen Sie Konfiguration des Regelkreises.

Technologie-Alarm 542

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler Klemmungsüberwachung: Achse verlässt Klemmungstoleranzfenster.	Die Achse hat am Festanschlag eine Bewegung ausgeführt, die größer als die zulässige Toleranz ist. Prüfen Sie, ob der Festanschlag weggebrochen ist.

Technologie-Alarm 550

Alarmreaktion: Sollwerte nachführen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Antriebsautarke Bewegung wird ausgeführt.	Der Antrieb führt eine Bewegung aus, die nicht vom Technologieobjekt vorgegeben wurde. Prüfen Sie, ob eine Safety-Funktion im Antrieb aktiv ist.

Technologie-Alarm 551

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Max. Geschwindigkeit mit Antriebs-/Achsparametern nicht erreichbar.	Die konfigurierte maximale Geschwindigkeit kann mit der konfigurierten Mechanik der Achse nicht erreicht werden. Überprüfen Sie die Konfiguration der Mechanik und die eingestellte Referenzdrehzahl der Achse.

Technologie-Alarm 552

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Adaptionsfehler des Gebers bei Hochlauf.	
Geber ist keinem SINAMICS-Gerät zugeordnet.	• Der operativ wirksame Geber konnte nicht adaptiert werden. Weitere Geber, die verwendet werden können, sind konfiguriert. Verwenden Sie die Geberumschaltung (MC_SetSensor). • Der als operativ wirksam eingestellte Geber konnte nicht adaptiert werden. • Geben Sie einen anderen Sensor für die Initialisierung des Technologieobjekts an.
Geber nicht direkt auf I/O-Bereich verschaltet.	Bei der konfiguration der Achse wurden die log. Adressen z. B. auf einen Datenbaustein- bzw. Merkerbereich gelegt. Die Adaption ist nur möglich, wenn der Geber direkt auf einen I/O-Bereich verschaltet wurde.
Abbruch Adaption wegen Ressourcenmangel.	Prüfen Sie, ob Ihr Gerät die azyklische Datenkommunikation nach PROFIdrive unterstützt.
Parameter inexistent, Wert unlesbar oder unzulässig.	
Gebersystem	
Geberauflösung	
Geberfeinauflösung	
Geberumdrehungen	

Technologie-Alarme 601-618

Technologie-Alarm 601

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext	Abhilfe
Leitachse nicht zugeordnet oder fehlerhaft.	Konfigurieren Sie die möglichen Leitwertachsen an der Folgeachse unter "Konfiguration > Leitwertverschaltungen".

Technologie-Alarm 603

Alarmreaktion: keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Leitachse befindet sich nicht im lagegeregelten Betrieb.	Für die Gleichlauffunktionalität muss die Folgeachse im lagegeregelten Betrieb betrieben werden.

Technologie-Alarm 608

Alarmreaktion: Stopp mit maximalen Dynamikwerten

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler beim Aufsynchronisieren.	Verhindern Sie eine reversierende Leitwertbewegung während des Aufsynchronisiervorgangs.

Technologie-Alarm 611

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Die am Parameter MC_CamIn.Cam angegebene Kurvenscheibe wurde nicht konfiguriert bzw. ist nicht verfügbar.	Konfigurieren und interpolieren Sie die Kurvenscheibe. Starten Sie den Auftrag erneut.

Technologie-Alarm 612

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Angegebene Kurvenscheibe wurde nicht interpoliert.	Interpolieren Sie die zum Kurvenscheibengleichlauf verwendete Kurvenscheibe mit der Motion Control-Anweisung "MC_InterpolateCam".

Technologie-Alarme 700-758

Technologie-Alarm 700

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler bei der Nockenbegrenzung.	
Nockenposition: OnPosition	Die Position für den Parameter "OnPosition" konnte nicht berechnet werden. Wegen den Vorhaltezeiten wurden nicht gültige Positionen berechnet (z. B. "OnPosition" > "OffPosition"). Der Nocken kann wegen der Achsdynamik und den Kompensationszeiten nicht geschaltet werden.
Nockenposition: OffPosition	Die Position für den Parameter "OffPosition" konnte nicht berechnet werden. Wegen den Vorhaltezeiten wurden nicht gültige Positionen berechnet (z. B. "OffPosition" < "OnPosition"). Der Nocken kann wegen der Achsdynamik und den Kompensationszeiten nicht geschaltet werden.

Technologie-Alarm 701

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Peripherieausgabefehler.	Der digitale Ausgang für das Technologieobjekt Nocken oder Nockenspur kann nicht angesprochen werden. Laden sie die Gerätekonfiguration erneut.

Technologie-Alarm 702

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Positionswert ungültig.	• Auf die Achse wird ein Motion Control-Auftrag "MC_Reset" ausgeführt. Warten Sie bis der Restart des Technologieobjekts abgeschlossen ist. • Die Geberwerte sind wegen eines Geberfehlers ungültig. Überprüfen Sie den Geber und passen Sie die Konfiguration gegebenenfalls an.

Technologie-Alarm 703

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Nockendaten fehlerhaft.	
Nocken: Nockennummer	Überprüfen Sie die Konfiguration des betroffenen Nockens in der Nockenspur und passen Sie ggf. die Werte an. Beispiele für eine korrekte Konfiguration: • "<TO>.Parameter.Cam[].OnPosition" < "<TO>.Parameter.Cam[].OffPosition" • "<TO>.Parameter.Cam[].Duration" > "<TO>.Parameter.OffCompensation" - "<TO>.Parameter.OnCompensation"

Technologie-Alarm 704

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Nockendaten fehlerhaft.	Überprüfen Sie die Konfiguration des Nockens und passen Sie ggf. die Werte an. Beispiele für eine korrekte Konfiguration: • "MC_OutputCam.OnPosition" < "MC_OutputCam.OffPosition" • "MC_OutputCam.Duration" > "<TO>.Parameter.OffCompensation" - "<TO>.Parameter.OnCompensation"

Technologie-Alarm 750

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Messauftrag beim Referenzieren der zugeordneten Achse nicht möglich.	Verwenden Sie die Motion Control-Anweisungen "MC_Home" und "MC_MeasuringInput" nicht gleichzeitig.

Technologie-Alarm 752

Alarmreaktion: Keine Reaktion

Restart: nicht erforderlich

Alarmtext	Abhilfe
Gültigkeitsbereich des Messauftrags nicht erkannt.	Der in der Motion Control-Anweisung "MC_MeasuringInput" angegebene Messbereich wurde nicht erkannt. Passen sie den Messbereich an.

Technologie-Alarm 753

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Zeitgleich kann nur ein Messtaster auf einen Geber zugreifen.	Verwenden Sie nur eine Motion Control-Anweisung "MC_MeasurinInput" für einen Geber.

Technologie-Alarm 754

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Messeingangsprojektierung in externem Gerät nicht korrekt.	Überprüfen Sie die Projektierung der Messeingänge am externen Gerät.

Technologie-Alarm 755

Alarmreaktion: Freigabe wegnehmen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Messauftrag nicht möglich.	
Gerät hat Fehler gemeldet.	Die Messung wurde mit Fehler abgebrochen. Prüfen sie die Messtasterfunktionalität im verwendeten Gerät
Zyklisches Messen mit Telegramm 39x nicht möglich.	• Verwenden Sie die Motion Control-Anweisung "MC_MeasuringInput" für das Starten einer einmaligen Messung. • Das zyklische Messen ist nur beim Messen über TM Timer DIDQ möglich. Ändern Sie die Konfiguration des Messtastertyps auf "TM Timer DIDQ".

Technologie-Alarm 758

Alarmreaktion: keine

Restart: nicht erforderlich

Alarmtext	Abhilfe
Eine Messflanke wurde nicht ausgewertet.	Am Messtastereingang wurde bereits eine Flanke erkannt, obwohl das Modul noch nicht bereit war. Der Messwert wird mit der nächsten Flanke bereitgestellt.

Fehlerkennung Motion Control-Anweisungen

Fehler an Motion Control-Anweisungen werden über die Parameter "Error" und "ErrorID" gemeldet.

Unter folgenden Bedingungen wird an der Motion Control-Anweisung "Error" = TRUE und "ErrorID" = 16#8xxx angezeigt:

- Unzulässiger Status des Technologieobjekts, der die Ausführung des Auftrags verhindert.
- Unzulässige Parametrierung der Motion Control-Anweisung, welche die Ausführung des Auftrags verhindert.
- Infolge der Alarmreaktion eines Fehlers am Technologieobjekt.

Die folgenden Tabellen zeigen eine Auflistung aller "ErrorIDs", die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

16#0000 - 16#800F

ErrorID	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8001	Während der Bearbeitung der Motion Control-Anweisung ist ein Technologie-Alarm (Fehler am Technologieobjekt) aufgetreten.	Im Technologie-Datenbaustein wird an der Variable "ErrorDetail.Number" eine Fehlermeldung ausgegeben. Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2183)".
16#8002	Unzulässige Angabe des Technologieobjekts	• Prüfen Sie die Angabe des Technologieobjekts am Parameter "Axis", "Master", "Slave", "OutputCam", "CamTrack", "MeasuringInput" oder "Cam". • Ein Technologieobjekt Kinematik können Sie nur am Parameter "AxesGroup" verwenden. • Bei "MC_MeasuringInputCyclic": Geben Sie am Parameter "MeasuringInputType" einen für die zyklische Messung gültigen Messtastertyp an.
16#8003	Unzulässige Geschwindigkeitsangabe	Geben Sie am Parameter "Velocity" einen zulässigen Wert für die Geschwindigkeit an.
16#8004	Unzulässige Beschleunigungsangabe	Geben Sie am Parameter "Acceleration" einen zulässigen Wert für die Beschleunigung an.
16#8005	Unzulässige Verzögerungsangabe	Geben Sie am Parameter "Deceleration" einen zulässigen Wert für die Verzögerung an.
16#8006	Unzulässige Ruckangabe	Geben Sie am Parameter "Jerk" einen zulässigen Wert für den Ruck an.
16#8007	Unzulässige Richtungsangabe	Geben Sie am Parameter "Direction" bzw. "SyncDirection" einen zulässigen Wert für die Drehrichtung an.
	Unzulässige Eingabe Beide Parameter "JogForward" und "JogBackward" sind gleichzeitig auf TRUE gesetzt. Die Achse wird mit der zuletzt gültigen Verzögerung abgebremst.	Setzen Sie beide Parameter "JogForward" und "JogBackward" zurück.
16#8008	Unzulässige Distanzangabe	Geben Sie am Parameter "Distance" einen zulässigen Wert für die Distanz an.
16#8009	Unzulässige Positionsangabe	Geben Sie am Parameter "Position" einen zulässigen Wert für die Position an.
16#800A	Unzulässige Betriebsart	Geben Sie am Parameter "Mode" eine zulässige Betriebsart an.
16#800B	Unzulässige Stoppmodusangabe	Geben Sie am Parameter "StopMode" einen zulässigen Wert für den Stoppmodus an.
16#800C	Nur eine Instanz der Anweisung je Technologieobjekt ist zulässig.	Die Anweisung wird an mehreren Stellen des Anwenderprogramms mit identischem Wert am Parameter "Axis", "Master", "Slave" oder "Cam" aufgerufen. Stellen Sie sicher, dass nur eine Anweisung mit dem Wert am Parameter "Axis", "Master", "Slave" oder "Cam" aufgerufen wird.

ErrorID	Beschreibung	Abhilfe
16#800D	Der Auftrag ist im aktuellen Zustand nicht erlaubt. "Restart" wird ausgeführt.	Während ein "Restart" ausgeführt wird, kann das Technologieobjekt keine Aufträge ausführen. Warten Sie, bis der "Restart" des Technologieobjekts abgeschlossen ist.
16#800E	Wenn das Technologieobjekt freigegeben ist, ist ein "Restart" nicht möglich.	Sperren Sie vor einem "Restart" das Technologieobjekt mit "MC_Power.Enable" = FALSE.
16#800F	Der Auftrag ist nicht ausführbar, da das Technologieobjekt gesperrt ist.	Geben Sie das Technologieobjekt mit "MC_Power.Enable" = TRUE frei. Starten Sie den Auftrag erneut.

16#8010 - 16#802F

ErrorID	Beschreibung	Abhilfe
16#8010	Unzulässige Referenzierungsfunktion bei inkrementellem Geber	Bei einem inkrementellen Geber ist keine Absolutwertgeberjustage ("Mode" = 6, 7) möglich. Starten Sie einen Referenziervorgang für einen inkrementellen Geber mit Parameter "Mode" = 0, 1, 2, 3, 5, 8, 10.
16#8011	Unzulässige Referenzierungsfunktion bei absolutem Geber	Bei einem Absolutwertgeber ist passives und aktives Referenzieren ("Mode" = 2, 3, 5, 8, 10) nicht möglich. Starten Sie einen Referenziervorgang für einen absoluten Geber mit Parameter "Mode" = 0, 1, 6, 7.
16#8012	Der Auftrag ist nicht ausführbar, da die Achssteuertafel aktiv ist.	Geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück. Starten Sie den Auftrag erneut.
16#8013	Die Onlineverbindung zwischen CPU und TIA Portal ist ausgefallen.	Prüfen Sie die Onlineverbindung zur CPU.
16#8014	Kein interner Auftragsspeicher verfügbar.	Die maximal mögliche Anzahl an Motion Control-Aufträgen ist erreicht. Reduzieren Sie die Anzahl der zu bearbeitenden Aufträge (Parameter "Execute" = FALSE).
16#8015	Fehlerquittierung mit "MC_Reset" ist nicht möglich. Fehler bei der Konfiguration des Technologieobjekts.	Prüfen Sie die Konfiguration des Technologieobjekts.
16#8016	Die Istwerte sind nicht gültig.	Um einen "MC_Home"- bzw. Positionierauftrag auszuführen, müssen die Istwerte gültig sein. Prüfen Sie den Status der Istwerte. Die Variable "<TO>.StatusSensor[n].State" des Technologieobjekts muss den Wert 2 (gültig) zeigen.
16#8017	Unzulässiger Wert für Getriebefaktor Zähler	Geben Sie am Parameter "RatioNumerator" einen zulässigen Wert für den Zähler des Getriebefaktors an. Zulässige ganzzahlige Werte: -2147483648 bis 2147483647 (Wert 0 nicht zulässig)
16#8018	Unzulässiger Wert für Getriebefaktor Nenner	Geben Sie am Parameter "RatioDenominator" einen zulässigen Wert für den Nenner des Getriebefaktors an. Zulässige ganzzahlige Werte: 1 bis 2147483647

ErrorID	Beschreibung	Abhilfe
16#8019	Der Auftrag ist nicht ausführbar. Die angegebene Folgeachse ist der ursprüngliche Leitwert für die Gleichlaufkette.	Rekursive Verschaltungen sind nicht möglich. Eine Leitachse lässt sich nicht als Folgeachse auf den eigenen Leitwert verschalten. Geben Sie am Parameter "Slave" eine zulässige Folgeachse an.
16#8021	Unzulässiger Wert für Verschiebung des Leitwertbereichs	Geben Sie am Parameter "MasterOffset" einen zulässigen Wert für die Verschiebung des Leitwertbereichs an.
16#8022	Unzulässiger Wert für Verschiebung des Folgewertbereichs	Geben Sie am Parameter "SlaveOffset" einen zulässigen Wert für die Verschiebung des Leitwertbereichs an.
16#8023	Unzulässiger Wert für Skalierung des Leitwertbereichs	Geben Sie am Parameter "MasterScaling" einen zulässigen Wert für die Skalierung des Leitwertbereichs an.
16#8024	Unzulässiger Wert für Skalierung des Folgewertbereichs	Geben Sie am Parameter "SlaveScaling" einen zulässigen Wert für die Skalierung des Folgewertbereichs an.
16#8026	Unzulässiger Wert für Leitwertweg	Geben Sie am Parameter "MasterStartDistance" einen zulässigen Wert für den Leitwertweg an.
16#8027	Unzulässiger Wert für Anwendung der Kurvenscheibe	Geben Sie am Parameter "ApplicationMode" einen zulässigen Wert für die zyklische/nicht zyklische Anwendung der Kurvenscheibe an.

16#8030 - 16#807F

ErrorID	Beschreibung	Abhilfe
16#8034	Unzulässiger Wert für Synchronposition der Leitachse	Geben Sie am Parameter "MasterSyncPosition" einen zulässigen Wert für die Synchronposition der Leitachse an.
16#8035	Unzulässiger Wert für Synchronposition der Folgeachse	Geben Sie am Parameter "SlaveSyncPosition" einen zulässigen Wert für die Synchronposition der Folgeachse an.
16#8036	Unzulässiger Wert für Art des Aufsynchronisierens	Geben Sie am Parameter "SyncProfileReference" einen zulässigen Wert für die Art des Aufsynchronisierens an.
16#8040	Unzulässiger Wert für die Anfangsposition des Nockens	Geben Sie am Parameter "OnPosition" einen zulässigen Wert für die Anfangsposition des Nockens an.
16#8041	Unzulässiger Wert für die Endposition des Wegnockens	Geben Sie am Parameter "OffPosition" einen zulässigen Wert für die Endposition des Wegnockens an.
16#8042	Unzulässiger Wert für die Einschaltdauer des Zeitnockens	Geben Sie am Parameter "Duration" einen zulässigen Wert für die Einschaltdauer des Zeitnockens an.
16#8043	Unzulässiger Wert für die Kraft-/Momentenbegrenzung	Geben Sie am Parameter "Limit" einen Wert innerhalb des zulässigen Wertebereichs an. Zulässige ganzzahlige Werte: -2147483648 bis 2147483648
16#8044	Die Achse ist nicht für Momentenreduzierung konfiguriert.	Antriebstelegramm 102, 103, 105 oder 106 wählen.
16#8045	Der Auftrag ist nicht ausführbar, da ein Auftrag zum Fahren auf Festanschlag aktiv ist.	Bei aktivem Fahren auf Festanschlag kann nicht in den nicht lagegeregelten Betrieb umgeschaltet werden.
16#8046	Der Auftrag "MC_TorqueLimiting" kann im Zustand "InClamping" nicht deaktiviert werden.	Fahren Sie die Achse frei und deaktivieren Sie "MC_TorqueLimiting".
16#8047	Die Bewegung führt hin zum Festanschlag.	Nur Bewegungen, die vom Festanschlag weg führen, sind zulässig.
16#804A	Unzulässiger Wert für additives Sollmoment	Geben Sie am Parameter "Value" einen zulässigen Wert für das additive Sollmoment an.

ErrorID	Beschreibung	Abhilfe
16#804B	Unzulässiger Wert für obere Momentengrenze	Geben Sie am Parameter "UpperLimit" einen zulässigen Wert für die obere Momentengrenze an.
16#804C	Unzulässiger Wert für untere Momentengrenze	Geben Sie am Parameter "LowerLimit" einen zulässigen Wert für die untere Momentengrenze an.
16#804D	Der Wert der oberen Momentengrenze ist kleiner als oder gleich dem Wert der unteren Momentengrenze.	Passen Sie die Werte der Parameter "UpperLimit" und "LowerLimit" an, sodass der Wert der oberen Momentengrenze größer als der Wert der unteren Momentengrenze ist.
16#804E	Der Auftrag ist nicht ausführbar, da ein "MC_TorqueLimiting"-Auftrag aktiv ist.	Beenden Sie die Kraft-/Momentenbegrenzung bzw. Festanschlagserkennung. Starten Sie den "MC_TorqueRange"-Auftrag erneut.
	Der Auftrag ist nicht ausführbar, da ein "MC_TorqueRange"-Auftrag aktiv ist.	Beenden Sie die Vorgabe der oberen und unteren Momentengrenzen. Starten Sie den "MC_TorqueLimiting"-Auftrag erneut.
16#804F	Die Achse ist nicht für zusätzliche Momentenwerte konfiguriert.	Verwenden Sie das Zusatztelegramm 750.
16#8050	Unzulässige Gebernummer	Geben Sie am Parameter "MC_SetSensor.Sensor" eine zulässige Nummer des neuen Gebers (1 bis 4) an.
16#8051	Unzulässige Nummer des Referenzgebers	Geben Sie am Parameter "MC_SetSensor.ReferenceSensor" eine zulässige Nummer des Referenzgebers an.
16#8062	Unzulässiger Näherungswert	Geben Sie am Parameter "ApproachLeadingValue" einen zulässigen Näherungswert zum gesuchten Leitwert an.
16#8063	Für den angegebenen Folgewert existiert keine gültige Abbildung in den Definitionsbereich (Leitwerte).	Geben Sie am Parameter "FollowingValue" einen zulässigen Folgewert an.
16#8064	Für den angegebenen Leitwert existiert keine gültige Abbildung in den Wertebereich (Folgewerte).	Geben Sie am Parameter "LeadingValue" einen zulässigen Leitwert an.
16#8070	Unzulässiger Wert für Leitwertverschiebung	Geben Sie am Parameter "PhaseShift" einen zulässigen Wert für die Leitwertverschiebung an.
16#8071	Der Auftrag ist nicht ausführbar, da sich die Achse nicht im lagegeregelten Betrieb befindet.	Aktivieren Sie den lagegeregelten Betrieb.
16#8074	Der Auftrag ist nicht ausführbar, da ein "MC_Home"-Auftrag aktiv ist.	Bei einem Aktiven oder Passiven Referenzieren wird eine Geberumschaltung abgelehnt. Warten Sie, bis der "MC_Home"-Auftrag abgeschlossen ist. Starten Sie den Auftrag erneut.
16#8075	Der Auftrag ist nicht ausführbar, da kein Gleichlauf an der Achse aktiv ist.	Schalten Sie die Gleichlauffunktion ein. Starten Sie den Auftrag erneut.
16#8076	Der Auftrag ist nicht ausführbar, da an der angegebenen Achse ein Gleichlauf simuliert wird.	Beenden Sie die Simulation des Gleichlaufs. Starten Sie den Auftrag erneut.

16#80A0 - 16#8FFF

ErrorID	Beschreibung	Abhilfe
16#80A1	Der Auftrag ist nicht ausführbar, da ein Gleichlaufauftrag aktiv ist.	Ein "MC_Home"-Auftrag an einer Folgeachse wird nicht ausgeführt, wenn ein "MC_CamIn"- oder "MC_GearInPos"-Auftrag aktiv ist. Beenden Sie den Gleichlaufauftrag. Starten Sie den Auftrag erneut.
16#80A2	Mit den konfigurierten Moduloachseneinstellungen ist der Messbereich ungültig.	Überprüfen und justieren Sie den Messeingang und passen Sie bei Bedarf die Messbereichspositionen an.
16#80A3	Der Messtasterauftrag über PROFIdrive-Telegramm konnte nicht gestartet werden, da ein Referenzierauftrag aktiv ist.	Das gleichzeitige Ausführen eines Referenzierauftrags und eines Messtasterauftrags über PROFIdrive-Telegramm ist nicht möglich. Warten Sie, bis der Referenzierauftrag beendet ist. Starten Sie den Messauftrag über PROFIdrive-Telegramm erneut.
16#80A5	Unzulässiger Wert für die Anfangsposition des Messbereichs	Geben Sie am Parameter "MC_MeasuringInput.StartPosition" bzw."MC_MeasuringInputCyclic.StartPosition" einen zulässigen Wert für die Anfangsposition des Messbereichs an.
16#80A6	Unzulässiger Wert für die Endposition des Messbereichs	Geben Sie am Parameter "MC_MeasuringInput.EndPosition" bzw. "MC_MeasuringInputCyclic.EndPosition" einen zulässigen Wert für die Endposition des Messbereichs an.
16#80A7	Beim Messen mit Messbereich ist eine Messung erfolgt, die berechnete Position liegt jedoch außerhalb des angegebenen Messbereichs. Der Messwert wird verworfen.	Überprüfen und justieren Sie den Messeingang und passen Sie bei Bedarf die Messbereichspositionen an.
16#80A8	Der Auftrag ist nicht ausführbar, da ein Kurvenscheibengleichlauf an der Achse aktiv ist.	Die Motion Control-Anweisungen "MC_PhasingRelative" und "MC_PhasingAbsolute" sind nur auf einen aktiven Getriebegleichlauf mit "MC_GearIn" oder "MC_GearInPos" ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE) anwendbar.
16#80A9	Der Auftrag ist nicht ausführbar, da die Folgeachse aufsynchronisiert wird ("MC_GearInPos.StartSync" = TRUE) oder eine Kinematikbewegung aktiv ist.	Die Motion Control-Anweisungen "MC_PhasingRelative" und "MC_PhasingAbsolute" sind nur auf einen aktiven Getriebegleichlauf mit "MC_GearIn" oder "MC_GearInPos" ("MC_GearIn.InGear" = TRUE oder "MC_GearInPos.InSync" = TRUE) anwendbar.
16#80AA	Die Kurvenscheibe enthält keine Punkte oder Segmente und lässt sich nicht interpolieren.	Befüllen Sie die Kurvenscheibe mit Punkten/Segmenten. Starten Sie den Auftrag erneut.
16#80AB	Die Kurvenscheibe wird aktuell verwendet und lässt sich nicht interpolieren.	Beenden Sie die aktuelle Verwendung der Kurvenscheibe. Starten Sie den Auftrag erneut.
16#80AC	Die Kurvenscheibe enthält fehlerhafte Punkte oder Segmente und lässt sich nicht interpolieren. (Die Kurvenscheibe enthält z. B. nur einen Punkt.)	Befüllen Sie die Kurvenscheibe mit zulässigen Punkten/Segmenten. Starten Sie den Auftrag erneut.
16#80AD	Die vorgegebene Synchronposition liegt außerhalb des Definitionsbereichs der Kurvenscheibe.	Geben Sie am Parameter "MasterSyncPosition" eine zulässige Synchronposition an. Starten Sie den Auftrag erneut.

ErrorID	Beschreibung	Abhilfe
16#80AE	Der Auftrag ist nicht ausführbar, da eine Kinematikbewegung aktiv ist.	Beenden Sie die aktuelle Kinematikbewegung. Starten Sie den Auftrag erneut.
16#8FFF	Unspezifizierter Fehler	Wenden Sie sich an Ihren Siemens-Ansprechpartner in den für Sie zuständigen Vertretungen und Geschäftsstellen. Ihren Ansprechpartner bei Industry Automation and Drive Technologies finden Sie unter: http://www.siemens.com/automation/partner (http://www.siemens.com/automation/partner)

Siehe auch

Fehler an Motion Control-Anweisungen (Seite 2037)

"MC_Power"-Funktionsdiagramme

Antriebsanbindung über PROFIdrive

"StopMode" = 0, 2

Funktionsdiagramm: Freigeben eines Technologieobjekts und Sperren mit "StopMode" = 0, 2

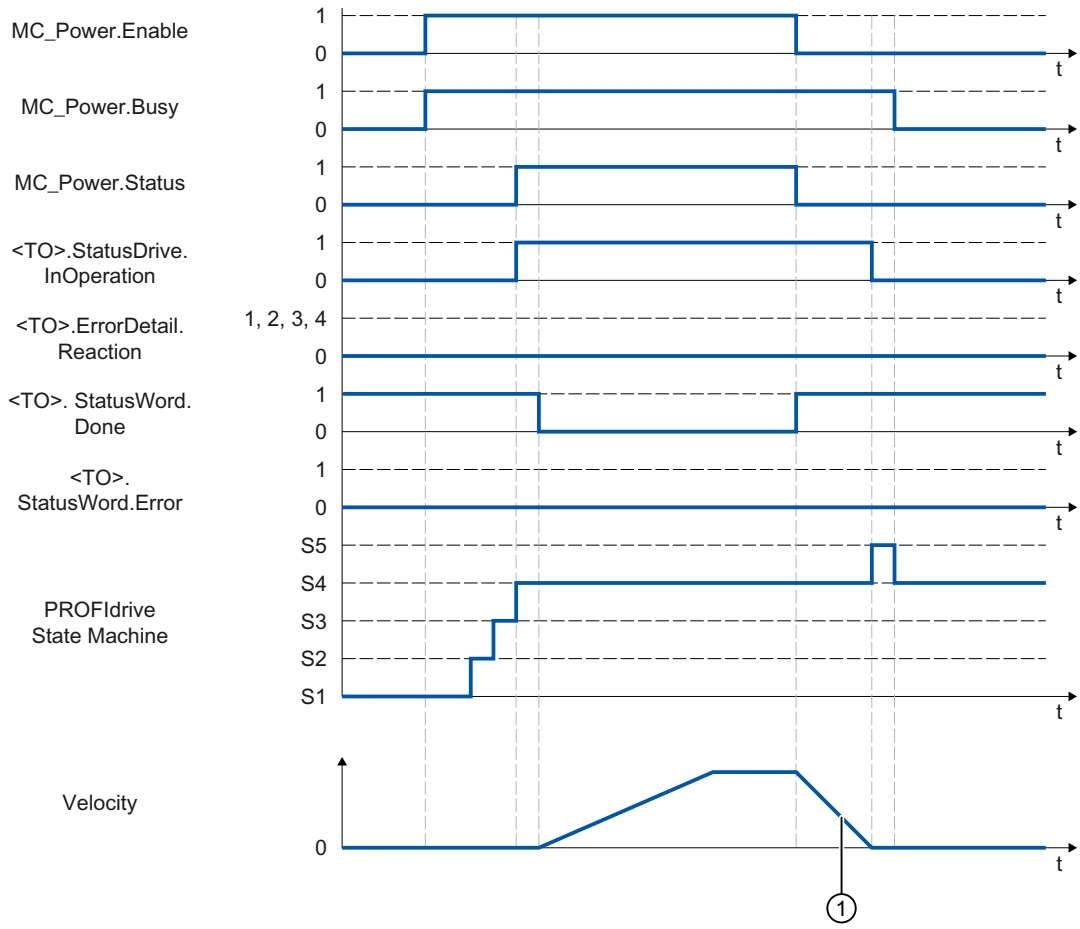

①
- "StopMode" = 0
  Die Achse wird mit der konfigurierten Notstopp-Verzögerung abgebremst.
- "StopMode" = 2
  Die Achse wird mit der konfigurierten maximalen Verzögerung abgebremst.

"StopMode" = 1

Funktionsdiagramm: Freigeben eines Technologieobjekts und Sperren mit "StopMode" = 1

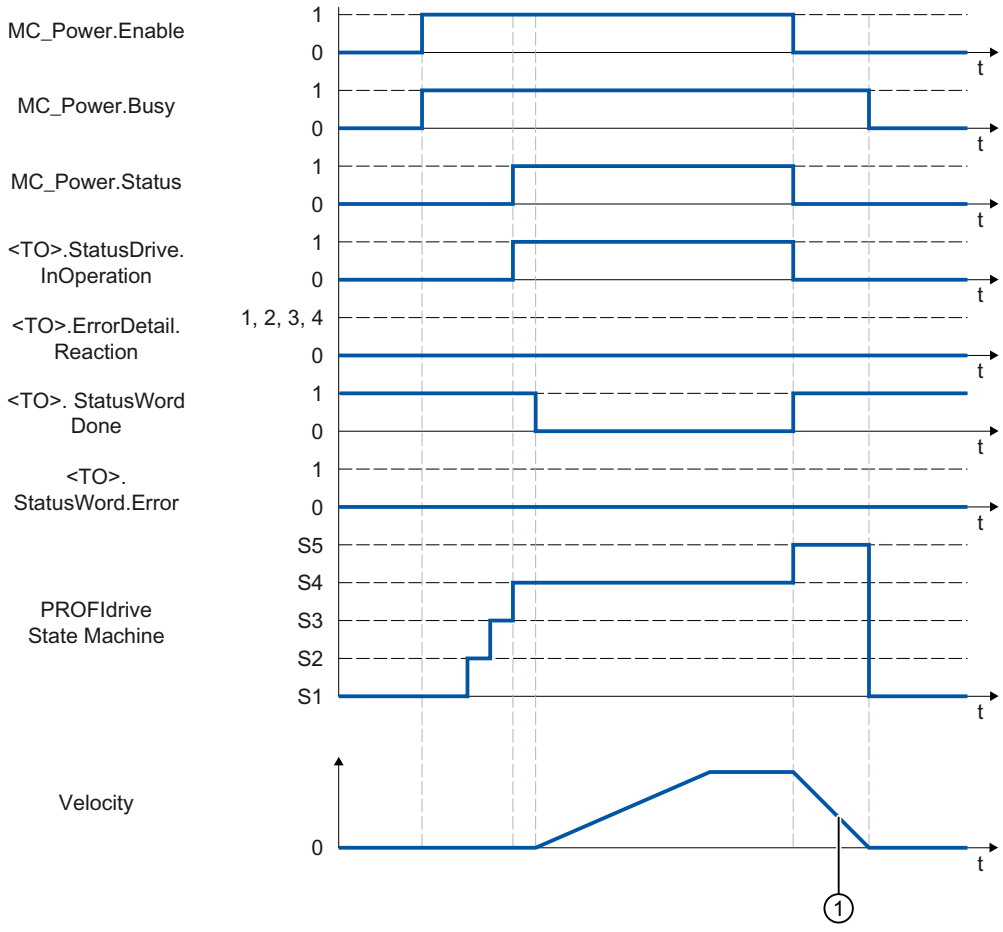

① Die Bremsrampe ist abhängig von der Konfiguration im Antrieb.

Alarmreaktionen mit Bremsrampe über das Technologieobjekt

Funktionsdiagramm: Freigeben eines Technologieobjekts und Auftreten eines Technologie-Alarms mit Bremsrampe über das Technologieobjekt

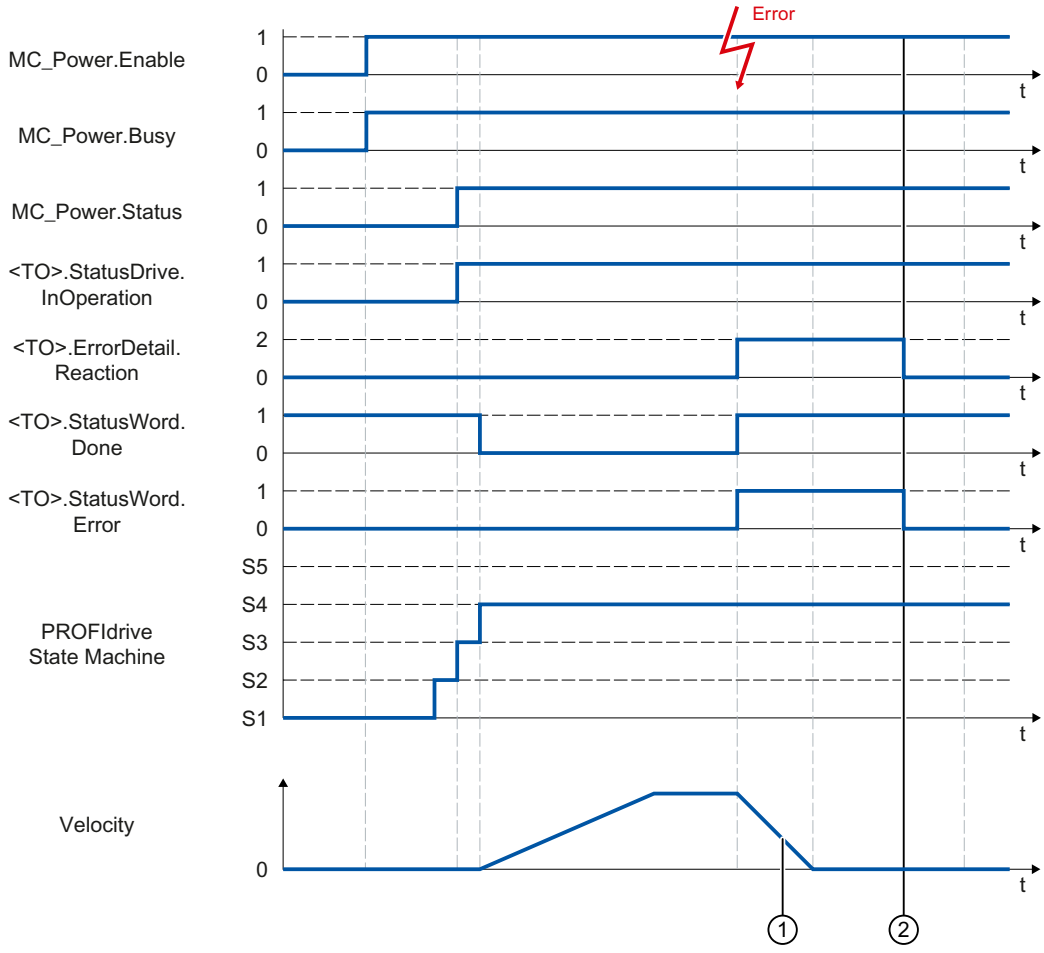

① Die Achse wird gemäß der Alarmreaktion abgebremst:

- Stopp mit aktuellen Dynamikwerten (<TO>.ErrorDetail.Reaction = 1)
  Die Achse wird mit der an der Motion Control-Anweisung anstehenden Verzögerung abgebremst.
- Stopp mit maximalen Dynamikwerten (<TO>.ErrorDetail.Reaction = 2)
  Die Achse wird mit der konfigurierten maximalen Verzögerung abgebremst.
- Stopp mit Notstopp-Rampe (<TO>.ErrorDetail.Reaction = 3)
  Die Achse wird mit der konfigurierten Notstopp-Verzögerung abgebremst.

② Der Technologie-Alarm wird quittiert.

Alarmreaktion "Freigabe wegnehmen"

Funktionsdiagramm: Freigeben eines Technologieobjekts und Auftreten eines Technologie-Alarms mit Alarmreaktion "Freigabe wegnehmen"

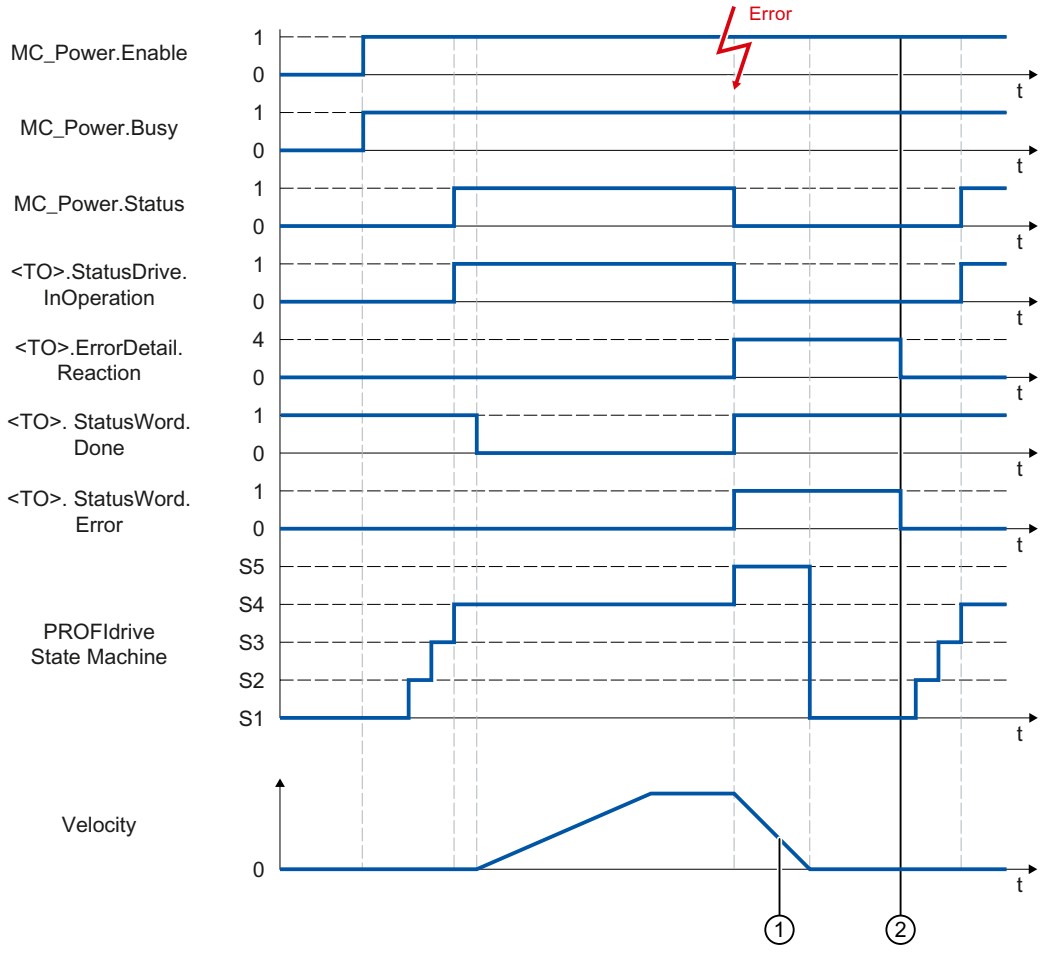

① Die Bremsrampe ist abhängig von der Konfiguration im Antrieb.

② Der Technologie-Alarm wird zum Zeitpunkt ② quittiert.

Analoge Antriebsanbindung

"StopMode" = 0, 2

Funktionsdiagramm: Freigeben eines Technologieobjekts und Sperren mit "StopMode" = 0, 2

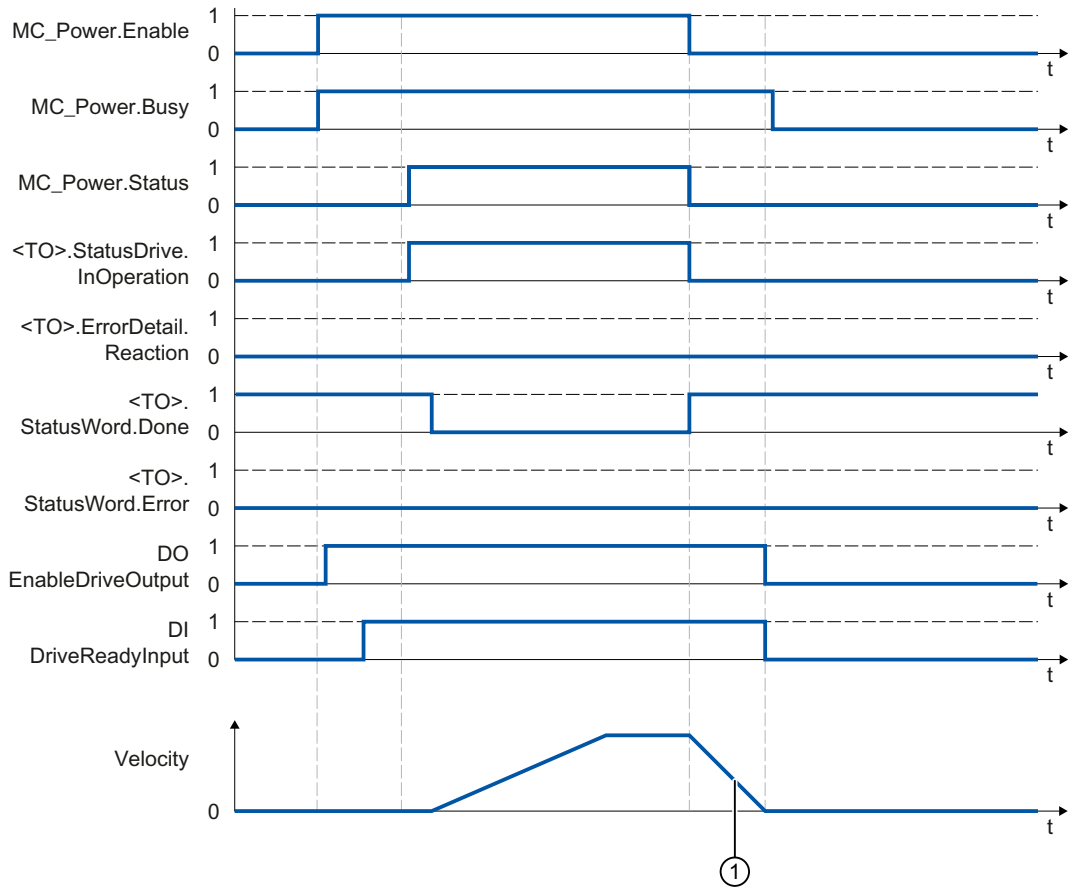

①
- "StopMode" = 0
  Die Achse wird mit der konfigurierten Notstopp-Verzögerung abgebremst.
- "StopMode" = 2
  Die Achse wird mit der konfigurierten maximalen Verzögerung abgebremst.

"StopMode" = 1

Funktionsdiagramm: Freigeben eines Technologieobjekts und Sperren mit "StopMode" = 1

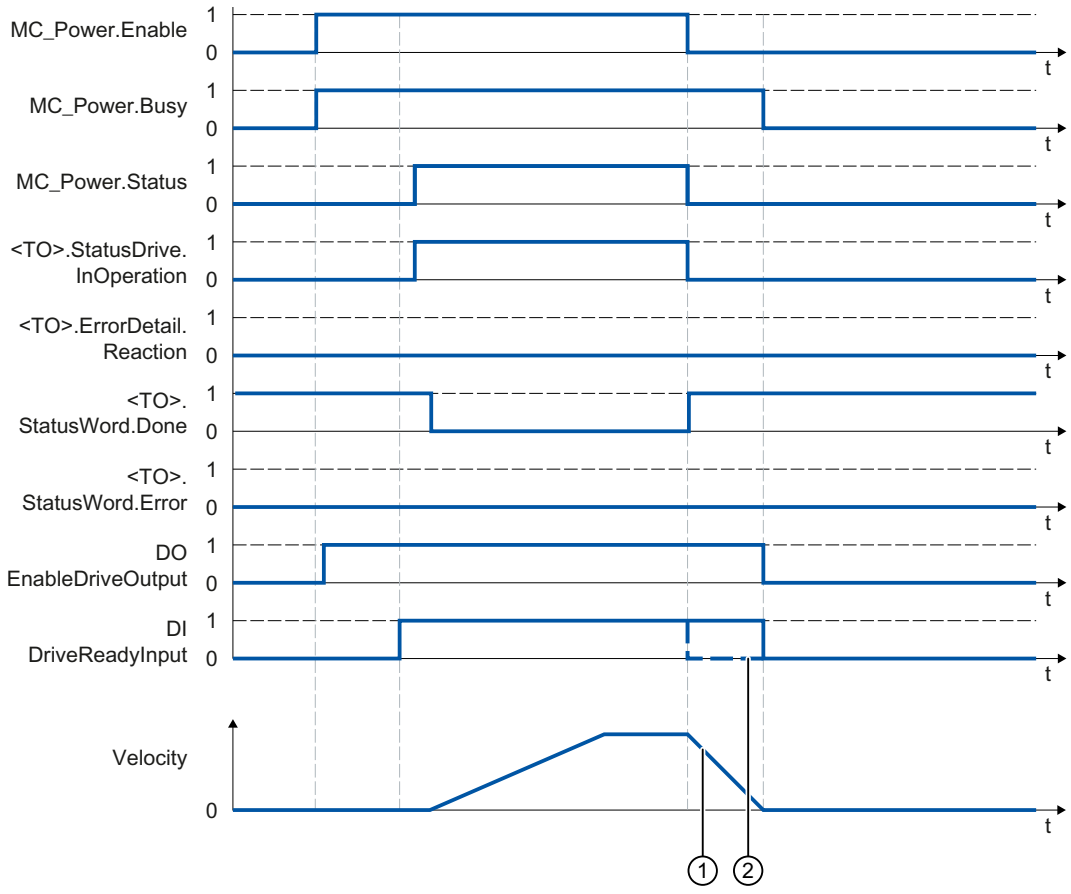

① Die Bremsrampe ist abhängig von der Konfiguration im Antrieb.

② Das Verhalten des Bereit-Signals des Antriebs "DI DriveReadyInput" ist herstellerspezifisch.

Alarmreaktionen mit Bremsrampe über das Technologieobjekt

Funktionsdiagramm: Freigeben eines Technologieobjekts und Auftreten eines Technologie-Alarms mit Bremsrampe über das Technologieobjekt

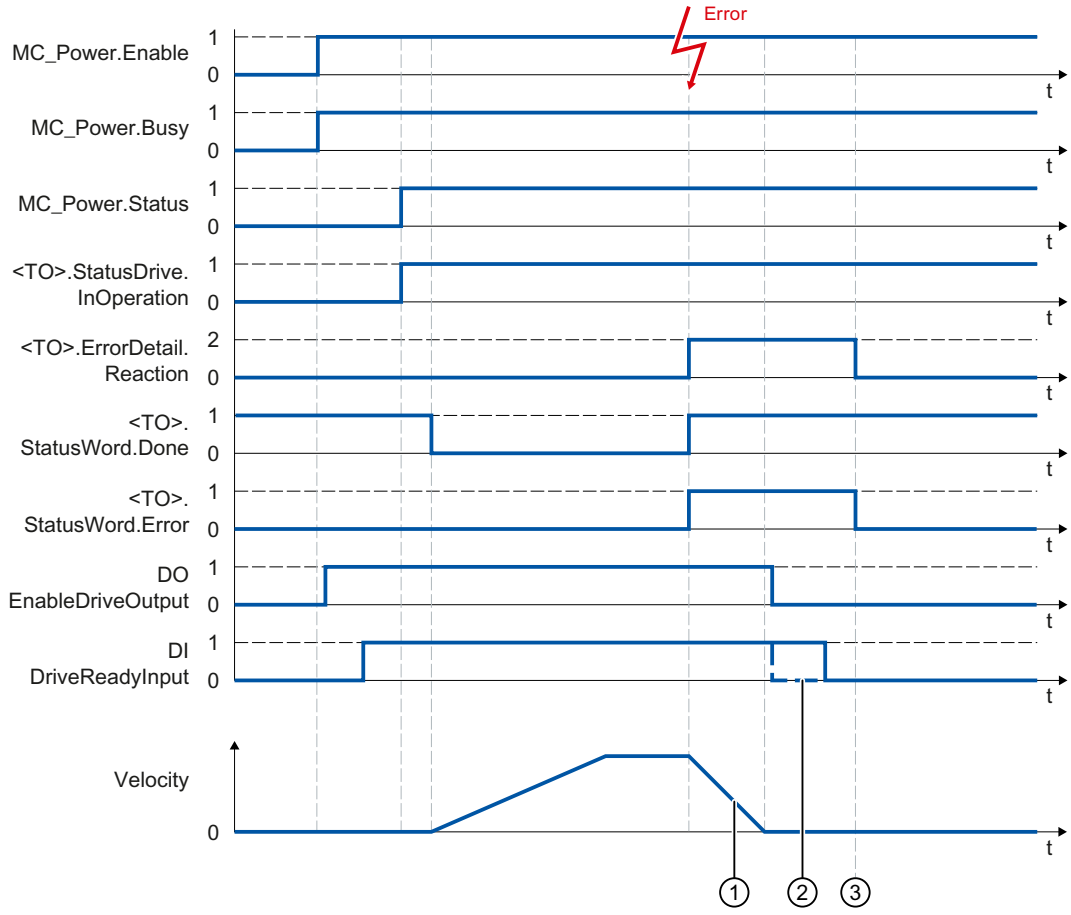

① Die Achse wird gemäß der Alarmreaktion abgebremst:

- Stopp mit aktuellen Dynamikwerten (<TO>.ErrorDetail.Reaction = 1)
  Die Achse wird mit der an der Motion Control-Anweisung anstehenden Verzögerung abgebremst.
- Stopp mit maximalen Dynamikwerten (<TO>.ErrorDetail.Reaction = 2)
  Die Achse wird mit der konfigurierten maximalen Verzögerung abgebremst.
- Stopp mit Notstopp-Rampe (<TO>.ErrorDetail.Reaction = 3)
  Die Achse wird mit der konfigurierten Notstopp-Verzögerung abgebremst.

② Das Verhalten des Bereit-Signals des Antriebs "DI DriveReadyInput" ist herstellerspezifisch.

③ Der Technologie-Alarm wird zum Zeitpunkt ③ quittiert.

Alarmreaktion "Freigabe wegnehmen"

Funktionsdiagramm: Freigeben eines Technologieobjekts und Auftreten eines Technologie-Alarms mit Alarmreaktion "Freigabe wegnehmen"

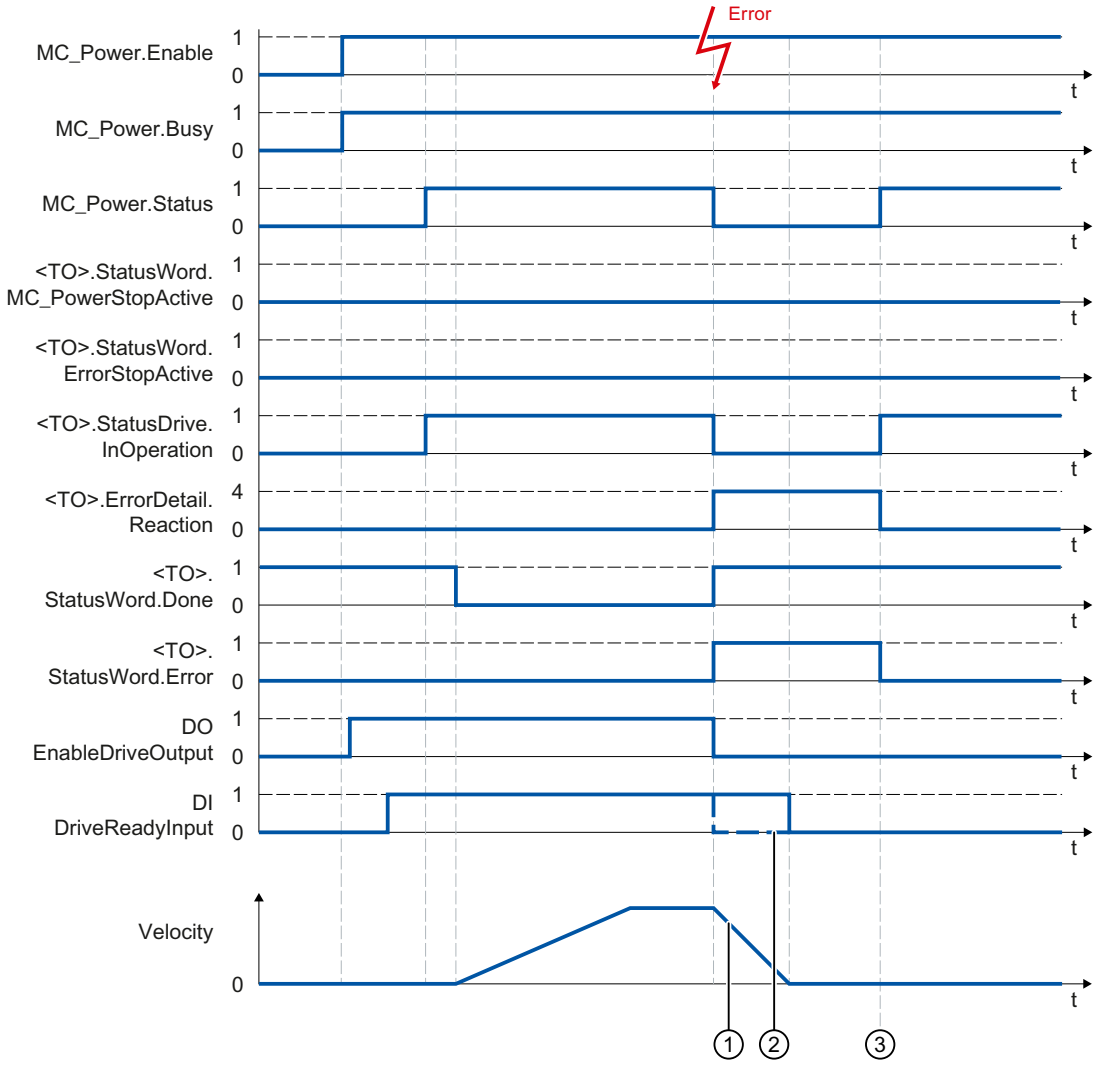

① Die Bremsrampe ist abhängig von der Konfiguration im Antrieb.

② Das Verhalten des Bereit-Signals des Antriebs "DI DriveReadyInput" ist herstellerspezifisch.

③ Der Technologie-Alarm wird zum Zeitpunkt ③ quittiert.

SINAMICS-Antriebe

Kompatibilitätsliste

Eine Übersicht, welche Antriebe Sie mit einer S7-1500 CPU verschalten können, finden Sie unter:

https://support.industry.siemens.com/cs/ww/de/view/109750431 (https://support.industry.siemens.com/cs/ww/de/view/109750431)

Referenzieren bei SINAMICS-Antrieben mit Externer Nullmarke

Bei SINAMICS-Antrieben mit Externer Nullmarke wird beim Referenzieren immer auf die linke Seite des Signals der Externen Nullmarke synchronisiert. D. h. bei positiver Fahrrichtung wird auf eine positive Flanke und bei negativer Fahrrichtung auf eine negative Flanke synchronisiert.

Durch Invertierung des Signals kann auch an der rechten Seite des Signals der Externer Nullmarke synchronisiert werden. Die Invertierung kann am Antrieb mit SINAMICS-Parameter P490 eingestellt werden.

Das Referenzieren auf eine Geber-Nullmarke oder eine Externe Nullmarke wird in SINAMICS-Parameter P495 eingestellt.

Datentypen

Datentypen für die Verwendung mit Technologie ab V4.0

Die folgende Tabelle enthält die Datentypen für die Referenz auf das jeweilige Technologieobjekt:

Datentyp	Struktur
TO_SpeedAxis	Drehzahlachse
TO_PositioningAxis	Positionierachse
TO_SynchronousAxis	Gleichlaufachse
TO_Encoder	Externer Geber
TO_OutputCam	Nocken
TO_CamTrack	Nockenspur
TO_MeasuringInput	Messtaster
TO_Cam	Kurvenscheibe (S7-1500T)
TO_Kinematics	Kinematik (S7-1500T)
PD_TELx	Telegramm Nr. "x"
PD_STW1_611Umode	Steuerwort 1 (STW1)
PD_STW2_611Umode	Steuerwort 2 (STW2)
PD_ZSW1_611Umode	Zustandswort 1 (ZSW1)
PD_ZSW2_611Umode	Zustandswort 2 (ZSW2)

Siehe auch

PROFIdrive-Telegramme (Seite 1693)

Extended Motion Control-Ressourcen

Wenn Sie mehrere Technologieobjekte Kinematik und mehrere Technologieobjekte Kurvenscheibe verwenden, können Sie die "Extended Motion Control-Ressourcen" der CPU nicht vollständig belegen.

CPU	Extended Motion Control-Ressourcen	Anzahl Technologieobjekte Kinematik	Maximale Anzahl Technologieobjekte Kurvenscheibe
CPU 1511T(F)	40	0	20
		1	0
CPU 1515T(F)	120	0	60
		1	45
		2	30
		3	15
		4	0
CPU 1516T(F)	192	0	96
		1	80
		2	64
		3	48
		4	32
		5	16
		6	0
CPU 1517T(F)	256	0	128
		1	112
		2	96
		3	80
		4	64
		5	48
		6	32
		7	16
		8	0

6.1.3 S7-1500T Kinematikfunktionen einsetzen

6.1.3.1 Einleitung

Zusammenspiel der verschiedenen Dokumentationen

Die Dokumentation der Motion Control-Funktionen ist für eine bessere Übersicht auf folgende Dokumentation aufgeteilt:

- S7-1500T Motion Control einsetzen
- S7-1500T Kinematikfunktionen einsetzen

Die Dokumentation "S7-1500 Motion Control einsetzen" beschreibt die Motion Control-Funktionen für folgende Technologieobjekte:

- Drehzahlachse
- Positionierachse
- Gleichlaufachse
- Externer Geber
- Messtaster
- Nocken
- Nockenspur
- Kurvenscheibe (S7-1500T)

Die Dokumentation "S7-1500T Kinematikfunktionen einsetzen" beschreibt die Motion Control-Funktionen für das Technologieobjekt Kinematik. Diese Dokumentation setzt die in "S7-1500 Motion Control einsetzen" beschriebenen Motion Control-Funktionen als bekannt voraus.

Siehe auch

Funktionshandbuch "S7-1500T Motion Control V4.0 im TIA Portal V15" (https://support.industry.siemens.com/cs/ww/de/view/109749263)

Kinematiken für Handhabungsaufgaben

Kinematiken sind frei programmierbare mechanische Systeme, bei denen mehrere mechanisch gekoppelte Achsen die Bewegung eines Arbeitspunkts bewirken. Die Technologie-CPUs S7-1500T stellen mit dem Technologieobjekt Kinematik Funktionen für das Steuern von Kinematiken bereit, z. B. für Handhabungsaufgaben. Typische Anwendungsfälle sind:

- Pick & Place
- Montage
- Palettieren

Die Kinematiksteuertafel und umfangreiche Online- und Diagnosefunktionen unterstützen die einfache Inbetriebnahme von Kinematiken. Das Technologieobjekt Kinematik ist durchgehend in die Systemdiagnose der CPU S7-1500 eingebunden.

Begriffsdefinition

Kinematik

Kinematiken sind frei programmierbare mechanische Systeme, bei denen mehrere mechanisch gekoppelte Achsen die Bewegung eines Arbeitspunkts bewirken.

Kinematikachsen

Kinematikachsen sind die Achsen, die eine Kinematik bewegen. Jede Kinematikachse verschalten Sie mit einem Technologieobjekt Positionierachse/Gleichlaufachse.

Kinematiknullpunkt (KNP)

Der Koordinatenursprung des Kinematikkoordinatensystems (KCS) ist der KNP. Vom KNP ausgehend konfigurieren Sie die Geometrieparameter der Kinematik.

Nullpunkt des Flanschkoordinatensystems (FNP)

Der Koordinatenursprung des Flanschkoordinatensystems (FCS) ist der FNP. Vom FNP ausgehend definieren Sie z. B. die Flanschzonen der Kinematik.

Werkzeugarbeitspunkt (TCP)

Der Koordinatenursprung des Werkzeugkoordinatensystems (TCS) ist der Werkzeugarbeitspunkt oder TCP. Der TCP ist der Arbeitspunkt der Kinematik.

Freiheitsgrade einer Kinematik

Die Freiheitsgrade einer Kinematik sind die Dimensionen, in denen sich das Werkzeug bewegen kann. 2D-Kinematiken bewegen das Werkzeug in der xz-Ebene und haben somit zwei translatorische Freiheitsgrade. 3D-Kinematiken bewegen das Werkzeug im xyz-Raum und haben somit drei translatorische Freiheitsgrade. Die optionale Orientierung des Werkzeugs ist ein weiterer Freiheitsgrad (Rotation des Werkzeugs um die z-Achse).

Maschinenkoordinatensystem (MCS)

Das MCS enthält die Positionsdaten der verschalteten Kinematikachsen und kombiniert somit bis zu vier eindimensionale Systeme in einem System.

Auftragskette

Die Auftragskette des Technologieobjekts Kinematik ist der Speicher, in den bewegungsrelevante Motion Control-Aufträge als anstehende, inaktive Aufträge eingetragen werden. Bei der Bewegungsaufbereitung werden alle Aufträge in der Auftragskette berücksichtigt.

AxesGroup

Kinematikbezogene Motion Control-Anweisungen besitzen den Eingangsparameter "AxesGroup". Das Technologieobjekt Kinematik gruppiert die verschalteten Kinematikachsen. Deshalb können Sie dem Eingangsparameter "AxesGroup" direkt das Technologieobjekt Kinematik zuweisen.

Funktionen

Die Funktionen des Technologieobjekts Kinematik führen Sie über die Motion Control-Anweisungen in Ihrem Anwenderprogramm oder das TIA Portal (unter "Technologieobjekt > Inbetriebnahme") aus.

Die folgende Tabelle zeigt die von dem Technologieobjekt unterstützten Funktionen:

Funktion	Kurzbeschreibung
Motion Control-Anweisungen (Anwenderprogramm)	
"MC_GroupInterrupt"	Bewegungsausführung unterbrechen
"MC_GroupContinue"	Bewegungsausführung fortsetzen
"MC_GroupStop"	Bewegung stoppen
"MC_MoveLinearAbsolute"	Kinematik mit linearer Bahnbewegung positionieren
"MC_MoveLinearRelative"	Kinematik mit linearer Bahnbewegung relativ positionieren
"MC_MoveCircularAbsolute"	Kinematik mit zirkularer Bahnbewegung positionieren
"MC_MoveCircularRelative"	Kinematik mit zirkularer Bahnbewegung relativ positionieren
"MC_DefineWorkspaceZone"	Arbeitsraumzone definieren
"MC_DefineKinematicsZone"	Kinematikzone definieren
"MC_SetWorkspaceZoneActive"	Arbeitsraumzone aktivieren
"MC_SetWorkspaceZoneInactive"	Arbeitsraumzone deaktivieren
"MC_SetKinematicsZoneActive"	Kinematikzone aktivieren
"MC_SetKinematicsZoneInactive"	Kinematikzone deaktivieren
"MC_DefineTool"	Werkzeug neu definieren
"MC_SetTool"	Aktives Werkzeug wechseln
"MC_SetOcsFrame"	Objektkoordinatensysteme neu definieren
TIA Portal	
"Kinematiksteuertafel (Seite 2369)"	Referenzieren von Kinematikachsen und Verfahren von Kinematiken oder einzelnen Kinematikachsen über das TIA Portal

6.1.3.2 Grundlagen

Technologieobjekt Kinematik

Das Technologieobjekt Kinematik berechnet unter Berücksichtigung der Dynamikvorgaben Bewegungssollwerte für den Werkzeugarbeitspunkt (TCP) einer Kinematik. Über die Kinematiktransformation berechnet das Technologieobjekt Kinematik die Bewegungssollwerte für die einzelnen Achsen und umgekehrt aus den aktuellen Werten der Achsen die Bewegung der Kinematik. Die achsspezifischen Bewegungssollwerte gibt das Technologieobjekt Kinematik an die verschalteten Positionierachsen aus.

Das Technologieobjekt Kinematik stellt systemseitig die Kinematiktransformation (Seite 2315) für die vordefinierten Kinematiktypen bereit. Bei anwenderdefinierten Kinematiken müssen Sie die Anwendertransformation (Seite 2320) in einem eigenen Programm bereitstellen.

Die einzelnen Achsen der Kinematik legen Sie im TIA Portal als Technologieobjekte vom Typ "Positionierachse" oder "Gleichlaufachse" an. In der Konfiguration des Technologieobjekts Kinematik verschalten Sie die Achsen entsprechend des konfigurierten Kinematiktyps.

Eine Übersicht über die Funktionen des Technologieobjekts Kinematik finden Sie im Kapitel "Funktionen" (Seite 2227).

Die folgende Grafik zeigt die prinzipielle Funktionsweise des Technologieobjekts Kinematik:

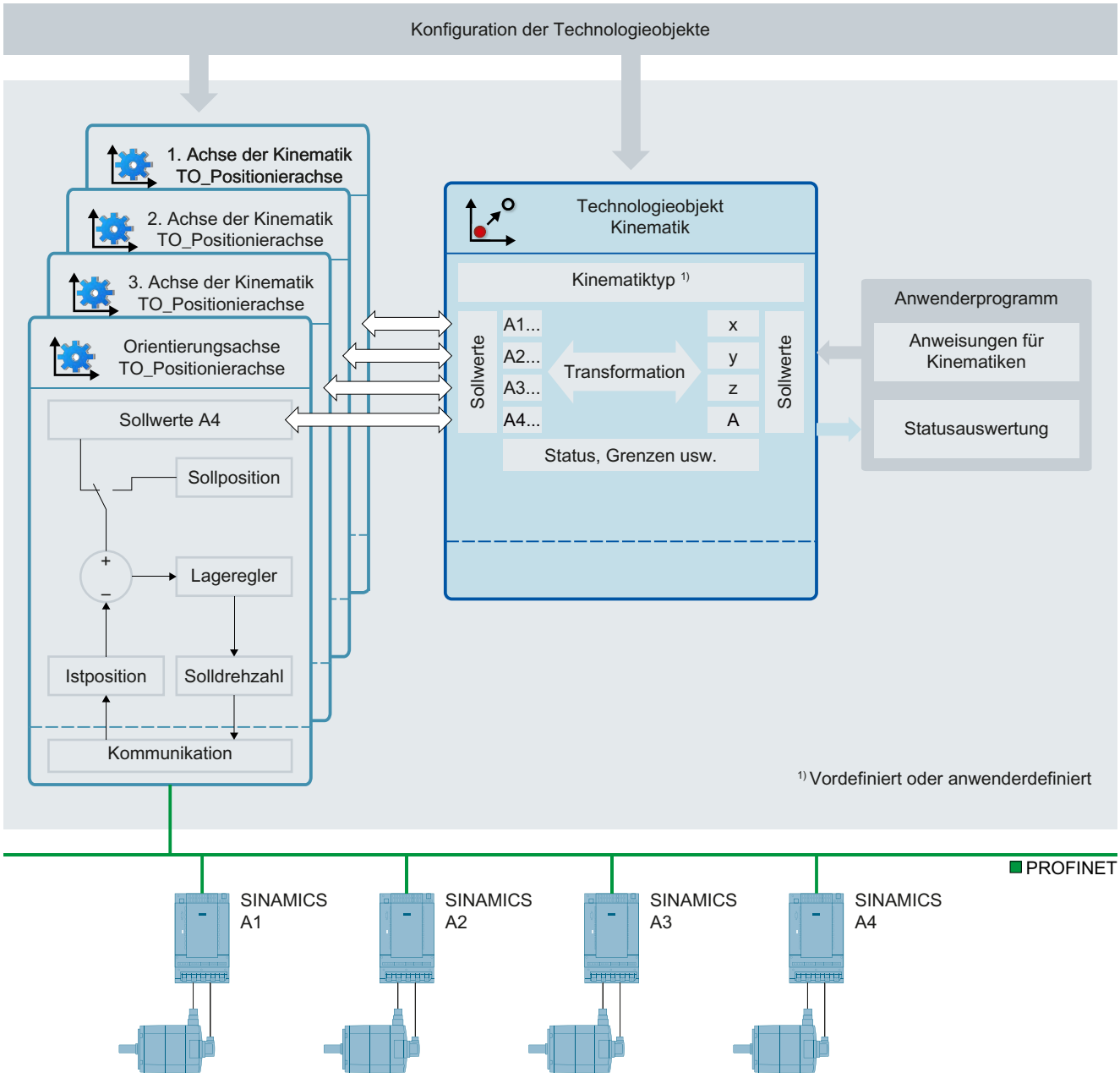

Verschaltungsregeln

Ein Technologieobjekt Kinematik können Sie mit Positionierachsen und Gleichlaufachsen verschalten. Der Bezug zwischen dem Technologieobjekt Kinematik und den verschalteten Achsen muss eindeutig sein. Ein zweites Technologieobjekt Kinematik können Sie nicht mit bereits verschalteten Achsen verwenden.

Eine Neuverschaltung der Achsen im laufenden Betrieb ist nicht vorgesehen.

Virtuelle Achse/Simulation

Das Technologieobjekt Kinematik können Sie auch mit Achsen in Simulation und virtuellen Achsen verschalten.

Mengengerüst für Kinematiken

Motion Control-Ressourcen

Jede CPU bietet eine definierte Menge an "Motion Control-Ressourcen". Die gesamt verfügbaren Motion Control-Ressourcen entnehmen Sie den technischen Daten der eingesetzten CPU.

Eine Übersicht über die Motion Control-Ressourcen einer CPU erhalten Sie im TIA Portal unter "Werkzeuge > Speicherauslastung".

Extended Motion Control-Ressourcen (S7-1500T)

Zusätzlich zu den Motion Control-Ressourcen der verschalteten Achsen belegt ein Technologieobjekt Kinematik 30 "Extended Motion Control-Ressourcen". Die maximale Anzahl der einsetzbaren Kinematiken entnehmen Sie den technischen Daten der eingesetzten CPU.

Die technischen Daten der S7-15xxT CPUs finden Sie im jeweiligen Gerätehandbuch.

Applikationszyklus

Mit steigender Anzahl eingesetzter Technologieobjekte benötigt die CPU mehr Rechenzeit zur Bearbeitung der Technologieobjekte. Der Motion Control-Applikationszyklus lässt sich entsprechend der Anzahl der eingesetzten Technologieobjekte anpassen.

Maßeinheiten

Das Technologieobjekt Kinematik unterstützt folgende Maßeinheiten für Position und Geschwindigkeit von linearen Achsen:

Position	Geschwindigkeit
nm, µm, mm, m, km	mm/s, mm/min, mm/h, m/s, m/min, m/h, km/min, km/h
in, ft, mi	in/s, in/min, ft/s, ft/min, mi/h

Das Technologieobjekt Kinematik unterstützt folgende Maßeinheiten für Winkel und Winkelgeschwindigkeit von rotatorischen Achsen:

Winkel	Winkelgeschwindigkeit
°, rad	°/s, °/min, rad/s, rad/min

Die Beschleunigung wird entsprechend als Maßeinheit der Position/s² (Winkel/s²) eingestellt.

Der Ruck wird entsprechend als Maßeinheit der Position/s³ (Winkel/s³) eingestellt.

Hinweis

Beachten Sie bei der Einstellung oder Änderung der Maßeinheiten die Auswirkung auf die Anzeige der Parameterwerte und das Anwenderprogramm:

- Anzeige der Parameterwerte im Technologie-Datenbaustein
- Versorgung der Parameter im Anwenderprogramm
- Eingabe und Anzeige der Position und Geschwindigkeit im TIA Portal
- Sollwertvorgaben durch Leitachsen im Gleichlauf

Alle Angaben und Anzeigen werden entsprechend der ausgewählten Maßeinheit dargestellt.

Die eingestellten Einheiten werden in der Variablenstruktur des Technologieobjekts <TO>.Units angezeigt. Die Variablenstruktur ist im Anhang (Seite 2409) beschrieben.

Maßeinheiten der Achsen und des Technologieobjekts Kinematik

Die Technologieobjekte übertragen Werte immer ohne Maßeinheiten.

Wenn Sie z. B. an einer Achse [mm] und am Technologieobjekt Kinematik [m] einstellen, verrechnet das Technologieobjekt Kinematik die Positionswerte der linearen Achse in [m]. Wenn in diesem Beispiel das Technologieobjekt Kinematik einen Sollwert für eine Bewegung um einen Meter ausgibt, verfährt die Achse nur um einen Millimeter.

Das Technologieobjekt Kinematik gibt lineare und rotatorische Sollwerte entsprechend des Kinematiktyps an die verschalteten Achsen aus. Das Technologieobjekt Kinematik prüft nicht den Achstyp der verschalteten Achse (linear oder rotatorisch).

Beachten Sie bei der Konfiguration der Maßeinheiten folgende Vorgaben:

- Konfigurieren Sie die verschalteten Technologieobjekte entsprechend des Kinematiktyps als lineare oder rotatorische Achsen.
- Konfigurieren Sie an den entsprechend des Kinematiktyps verschalteten Achsen dieselben linearen/rotatorischen Maßeinheiten wie am Technologieobjekt Kinematik.

Moduloeinstellung

Das Technologieobjekt Kinematik hat selbst keine Moduloeinstellung. Wenn Sie Achsen mit aktiver Moduloeinstellung am Technologieobjekt Kinematik verschalten, muss der Modulobereich der Achsen mindestens den Verfahrbereich der Kinematik abdecken. Die Nullstellung der Achse muss mit der Nullstellung der Kinematikachse übereinstimmen. Mit Ausnahme der Orientierungsachse kann während einer Kinematikbewegung der Modulobereich der Achsen nicht gewechselt werden.

Typischerweise findet die Moduloeinstellung bei der Orientierungsachse Anwendung. Bei der Orientierungsachse (Achse A4 in der Kinematik) deckt die Kinematiktransformation den gesamten Verfahrbereich der Achse ab. Die Orientierungsachse verfährt ohne Einschränkungen durch eine aktivierte Moduloeinstellung.

Für die kartesische Orientierung kann ein Winkel größer als 360° vorgegeben werden. Eine relative Bewegung fährt diesen Winkel aus. Eine absolute Bewegung bildet diesen Winkel in

dem Bereich von 0° bis 360° ab. Grundsätzlich wird für die Koordinate A des Werkzeugarbeitspunkts (TCP) der Bereich -180° bis +180° festgelegt.

Koordinatensysteme und Frames

An einer Handhabungsaufgabe sind viele Objekte beteiligt, z. B. Kinematiken, Werkzeuge, Paletten und Produkte. Diese Objekte und deren Lage zueinander beschreiben Sie mit Koordinatensystemen und Frames. Das Technologieobjekt Kinematik berechnet alle Bewegungen für den Werkzeugarbeitspunkt (TCP).

Frames

Frames geben die Verschiebung und die Verdrehung eines Koordinatensystems zu einem anderen Koordinatensystem an.

Koordinatensysteme

Das Technologieobjekt Kinematik verwendet folgende rechtsdrehende, rechtwinklige Koordinatensysteme nach DIN 66217:

- Weltkoordinatensystem (WCS)
- Kinematikkoordinatensystem (KCS)
- Flanschkoordinatensystem (FCS)
- Werkzeugkoordinatensystem (TCS)
- Objektkoordinatensysteme (OCS)

Folgende Grafik zeigt am Beispiel eines Arbeitsraums die Lage der Koordinatensysteme zueinander:

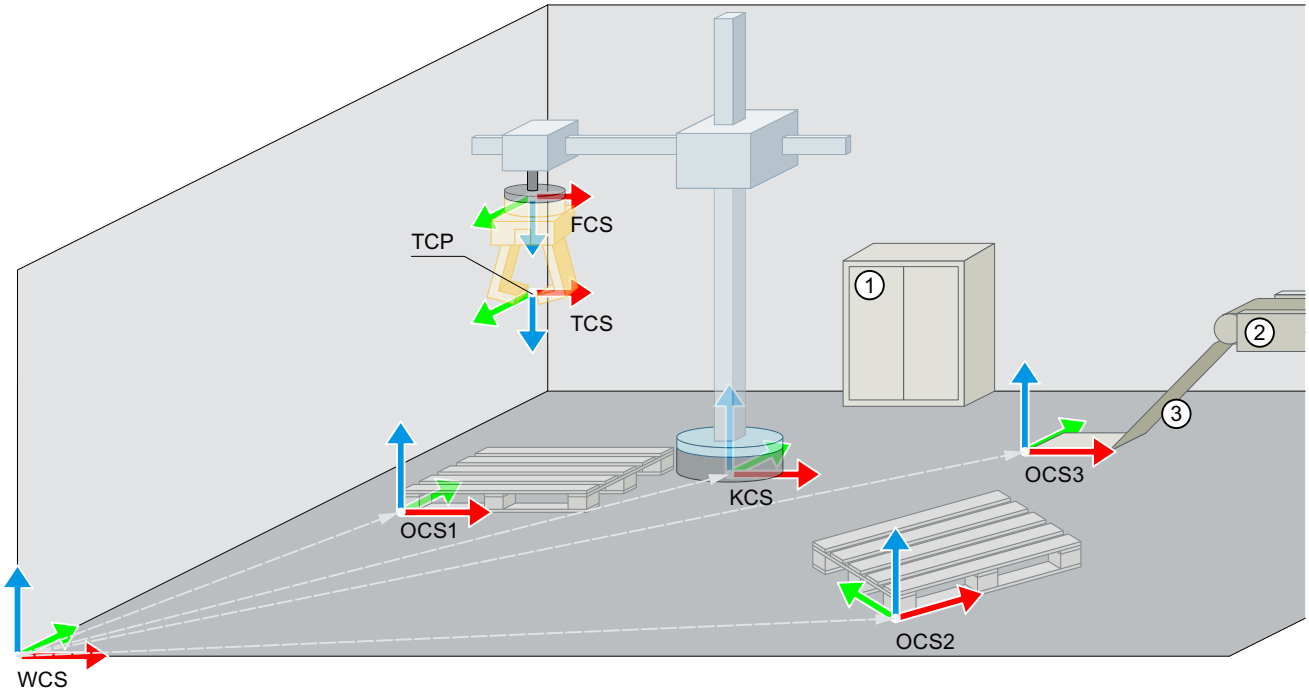

①	Schaltschrank
②	Förderband
③	Rutsche
WCS	Weltkoordinatensystem
KCS	Kinematikkoordinatensystem
FCS	Flanschkoordinatensystem
TCS	Werkzeugkoordinatensystem
TCP	Werkzeugarbeitspunkt
OCS	Objektkoordinatensystem

Weltkoordinatensystem (WCS - World Coordinate System)

Das WCS ist das ortsfeste Koordinatensystem der Umgebung bzw. des Arbeitsraums einer Kinematik. Der Nullpunkt des WCS ist der Bezugspunkt für Objekte und Bewegungen am Technologieobjekt Kinematik. Ausgehend vom Nullpunkt des WCS (z. B. Ecke eines Arbeitsraums) definieren Sie die Lage der Objekte durch Frames.

Kinematikkoordinatensystem (KCS - Kinematic Coordinate System)

Das KCS ist mit der Kinematik verbunden. Die Lage des KCS innerhalb der Kinematik ist für jeden vordefinierten Kinematiktyp festgelegt. Der Koordinatenursprung des KCS ist der Kinematiknullpunkt (KNP). Vom KNP ausgehend konfigurieren Sie die Geometrieparameter der Kinematik.

Die Lage des KCS im WCS konfigurieren Sie über den KCS-Frame.

Flanschkoordinatensystem (FCS - Flange Coordinate System)

Das FCS hängt an der Werkzeugaufnahme (Flansch) der Kinematik. Dadurch verändert sich die Lage des FCS mit Kinematikbewegungen.

Die Lage des FCS in der Nullstellung der Kinematik ergibt sich aus der Konfiguration der Geometrieparameter der Kinematik. Das Technologieobjekt Kinematik berechnet aus den Geometrieparametern den Transformations-Frame. Der Transformations-Frame beschreibt die Lage des FCS im KCS. Die z-Achse des FCS zeigt immer in die negative z-Richtung des KCS.

Die folgende Grafik zeigt am Beispiel der Kinematik "Zylindrischer Roboter" die Lagen des FCS und des KCS und den Transformations-Frame:

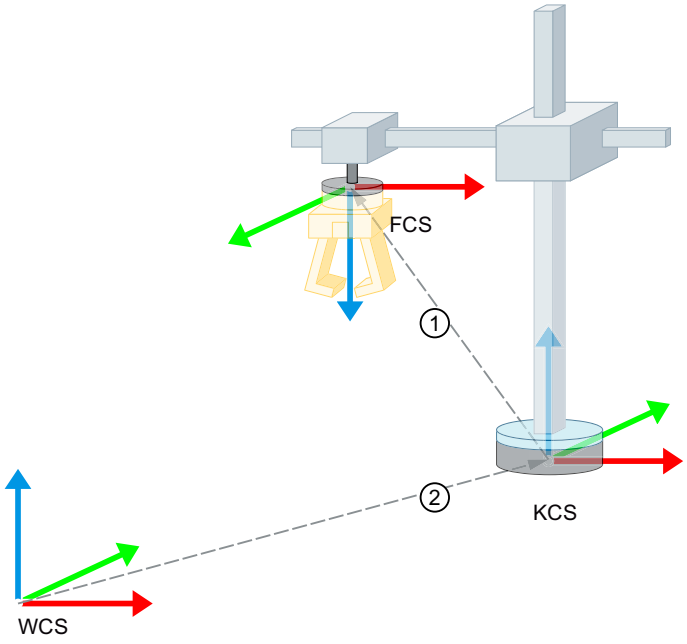

① Transformations-Frame

② KCS-Frame

Werkzeugkoordinatensystem (TCS - Tool Coordinate System) und Werkzeugarbeitspunkt (TCP - Tool Center Point)

Das TCS hängt am FCS und definiert im Koordinatenursprung den Werkzeugarbeitspunkt (TCP). Der TCP ist der Arbeitspunkt des Werkzeugs. Kinematikbewegungen beziehen sich immer auf den TCP (mit Bezug auf WCS/OCS). Die Lage des TCS im FCS wird über den aktiven Tool-Frame bestimmt. Die z-Achse des TCS zeigt immer in die negative z-Richtung des KCS. Sie können Tool-Frames für bis zu drei Werkzeuge definieren, wovon jeweils nur ein Werkzeug und somit ein Tool-Frame gleichzeitig aktiv ist.

Die folgende Grafik zeigt die Lage des TCS und des TCP im Arbeitsraum:

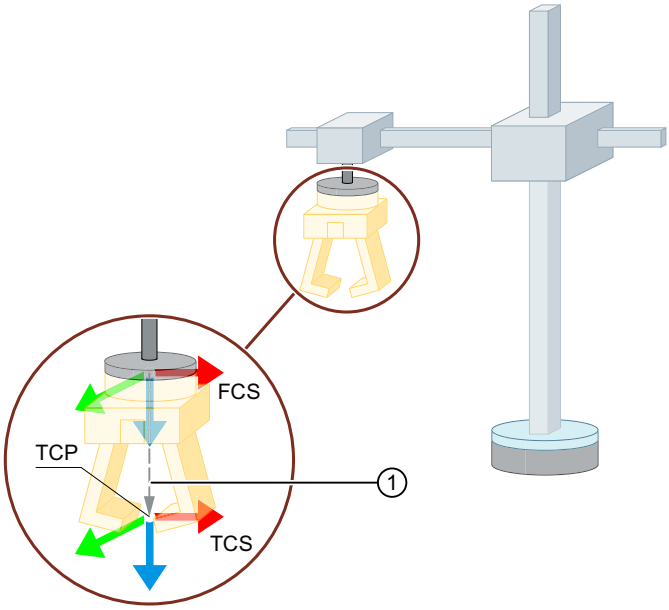

① Tool-Frame

Objektkoordinatensystem (OCS - Object Coordinate System)

Das OCS ist ein benutzerdefiniertes Koordinatensystem. Mit einem OCS definieren Sie z. B. die Lage einer Palette im Arbeitsraum. Die Lage des OCS im WCS definieren Sie mit einem OCS-Frame. Sie können bis zu drei OCS-Frames vorgeben, die gleichzeitig aktiv sind.

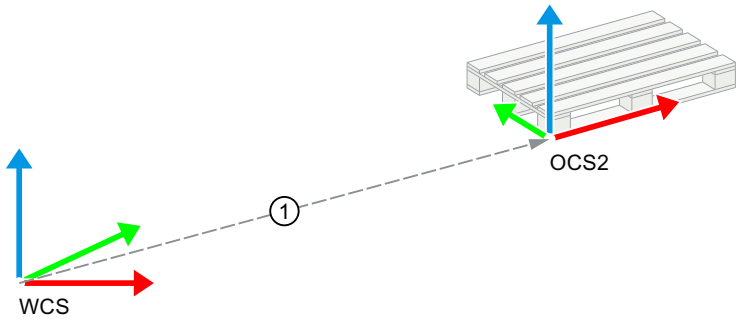

① OCS-Frame

Frames

Die folgende Tabelle zeigt die Frames für das Technologieobjekt Kinematik:

Frame	Beschreibung
KCS-Frame	Lage des Kinematikkoordinatensystems (KCS) im Weltkoordinatensystem (WCS)
Transformations-Frame	Lage des Flanschkoordinatensystems (FCS) im KCS Der Transformations-Frame ergibt sich aus der Kinematiktransformation und wird in der Variable "<TO>.FlangeInKcs" des Technologieobjekts angezeigt.
Tool-Frame	Lage des Werkzeugkoordinatensystems (TCS) im FCS

Frame	Beschreibung
OCS[1..3]-Frame	Lage der Objektkoordinatensysteme 1 bis 3 (OCS[1..3]) im WCS
Zielposition	Zielposition an einer Kinematikbewegung

Frame-Definition

Frames definieren die Verschiebung und die Verdrehung eines Koordinatensystems zu einem anderen Koordinatensystem mit folgenden Werten:

Wert im Frame	Beschreibung
x	Verschiebung in x-Richtung im Bezugskoordinatensystem
y	Verschiebung in y-Richtung im Bezugskoordinatensystem
z	Verschiebung in z-Richtung im Bezugskoordinatensystem
A	Drehung um die z-Achse
B	Drehung um die y-Achse
C	Drehung um die x-Achse

Die folgende Tabelle zeigt die Einschränkungen für Frames abhängig vom Kinematiktyp. Die Angaben "x", "y" und "z" bedeuten, dass eine Verschiebung in die jeweilige Richtung möglich ist. Die Angaben "A", "B" und "C" bedeuten, dass eine Verdrehung in die jeweilige Richtung möglich ist. Die Angabe "= 0.0" bedeutet, dass eine Verschiebung oder Verdrehung in der jeweiligen Richtung unzulässig oder für den Kinematiktyp nicht relevant ist.

Kinematiktyp	KCS-Frame/OCS-Frame		Tool-Frame		Zielposition	
2D	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung
	x	A = 0.0	x	A = 0.0	x	A = 0.0
	y = 0.0	B	y = 0.0	B = 0.0	y = 0.0	B = 0.0
	z	C = 0.0	z	C = 0.0	z	C = 0.0
2D mit Orientierung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung
	x	A = 0.0	x = 0.0	A	x	A
	y = 0.0	B = 0.0	y = 0.0	B = 0.0	y = 0.0	B = 0.0
	z	C = 0.0	z	C = 0.0	z	C = 0.0
3D	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung
	x	A	x	A = 0.0	x	A = 0.0
	y	B	y	B = 0.0	y	B = 0.0
	z	C	z	C = 0.0	z	C = 0.0
3D mit Orientierung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung	Verschie-bung	Verdrehung
	x	A	x	A	x	A
	y	B = 0.0	y	B = 0.0	y	B = 0.0
	z	C = 0.0	z	C = 0.0	z	C = 0.0

x, y, z, A, B, C: Verschiebung/Verdrehung möglich

Wert = 0.0: Verschiebung/Verdrehung unzulässig oder nicht relevant

Die folgende Tabelle zeigt die Wertebereiche für die Verdrehungen von KCS-, OCS- und Tool-Frames abhängig vom Kinematiktyp:

Kinematiktyp	Wertebereiche			
	KCS-Frame/OCS-Frame		Tool-Frame	
2D	A	0.0	A	0.0
	B	-180.0° bis 179.999°	-	-
	C	0.0	-	-
2D mit Orientierung	A	0.0	A	-180.0° bis 179.999°
	B	0.0	-	-
	C	0.0	-	-
3D	A	-180.0° bis 179.999°	A	0.0
	B	-90.0° bis 90.0°	-	-
	C	-180.0° bis 179.999°	-	-
3D mit Orientierung	A	-180.0° bis 179.999°	A	-180.0° bis 179.999°
	B	0.0	-	-
	C	0.0	-	-

Wert = 0.0: Verdrehung unzulässig

Keine Angabe (-): Parameter nicht vorhanden

Variablen Koordinatensysteme und Frames

Für Koordinatensysteme und Frames sind folgende Variablen des Technologieobjekts Kinematik relevant:

Variable	Beschreibung
Konfiguration	
<TO>.KcsFrame	KCS-Frame x, y, z, A, B, C
<TO>.OcsFrame[1..3]	OCS-Frame x, y, z, A, B, C
<TO>.Tool[1..3]	Tool-Frame x, y, z, A
Statuswerte	
<TO>.Tcp	Position des Werkzeugarbeitspunkts (TCP), TCP-Frame im Weltkoordinatensystem (WCS) x, y, z, A
<TO>.TcpInWcs	Parameter zum Werkzeugarbeitspunkt im Weltkoordinatensystem x, y, z, A
<TO>.TcpInOcs[1..3]	Parameter zum Werkzeugarbeitspunkt in den Objektkoordinatensystemen 1 bis 3 (OCS) x, y, z, A
<TO>.FlangeInKcs	Parameter zum Flanschkoordinatensystem (FCS) im Kinematikkoordinatensystem (KCS) x, y, z, A
<TO>.StatusOcsFrame	Anzeige der OCS-Frames x, y, z, A, B, C

Kinematiktypen

Die Art des mechanischen Systems und die Anzahl der Achsen bestimmen den Kinematiktyp. Die mechanisch gekoppelten Achsen bewirken die Bewegung des Werkzeugarbeitspunkts (TCP). Abhängig vom Kinematiktyp konfigurieren Sie die Kinematik über entsprechende Geometrieparameter.

Das Technologieobjekt Kinematik unterstützt folgende Kinematiktypen:

Kategorie	Kinematiktyp
Vordefinierte Kinematiken	
Kartesisches Portal (Seite 2239)	Kartesisches Portal 2D
	Kartesisches Portal 2D mit Orientierung
	Kartesisches Portal 3D
	Kartesisches Portal 3D mit Orientierung
Rollen-Picker (Seite 2250)	Rollen-Picker 2D
	Rollen-Picker 2D mit Orientierung
	Rollen-Picker 3D (vertikal)
	Rollen-Picker 3D mit Orientierung (vertikal)
	Rollen-Picker 3D mit Orientierung (horizontal)
SCARA (Seite 2264)	SCARA 3D mit Orientierung
Knickarm (Seite 2269)	Knickarm 2D
	Knickarm 2D mit Orientierung
	Knickarm 3D
	Knickarm 3D mit Orientierung
Delta-Picker (Seite 2285)	Delta-Picker 2D
	Delta-Picker 2D mit Orientierung
	Delta-Picker 3D
	Delta-Picker 3D mit Orientierung
Zylindrischer Roboter (Seite 2298)	Zylindrischer Roboter 3D
	Zylindrischer Roboter 3D mit Orientierung
Tripod (Seite 2307)	Tripod 3D
	Tripod 3D mit Orientierung
Anwenderdefinierte Kinematiken	
Anwenderdefinierte Kinematiken (Seite 2314)	Anwenderdefiniert 2D
	Anwenderdefiniert 2D mit Orientierung
	Anwenderdefiniert 3D
	Anwenderdefiniert 3D mit Orientierung

Legende zur Darstellung der Kinematiken

Die folgende Tabelle zeigt die grafischen Elemente und Symbole, die in der Darstellung der Kinematiken verwendet werden:

Grafisches Element	Bedeutung
	Basis der Kinematik
	Kinematikarm
	Kinematik aus Nullstellung ausgelenkt
	Aktive rotatorische Achse
	• Passives Gelenk • Führung einer Achse
	Aktive lineare Achse
	Rotatorische Achse an der Werkzeugaufnahme (Orientierungsachse)
	Werkzeugaufnahme
	Werkzeug (Greifer)
	Koordinatenachse aus der Abbildungsebene herausgerichtet
	Koordinatenachse in die Abbildungsebene hineingerichtet
	Farbe x-Achse
	Farbe y-Achse
	Farbe z-Achse

Kartesisches Portal

Portal 2D

Die Kinematik "Portal 2D" unterstützt zwei Achsen und zwei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

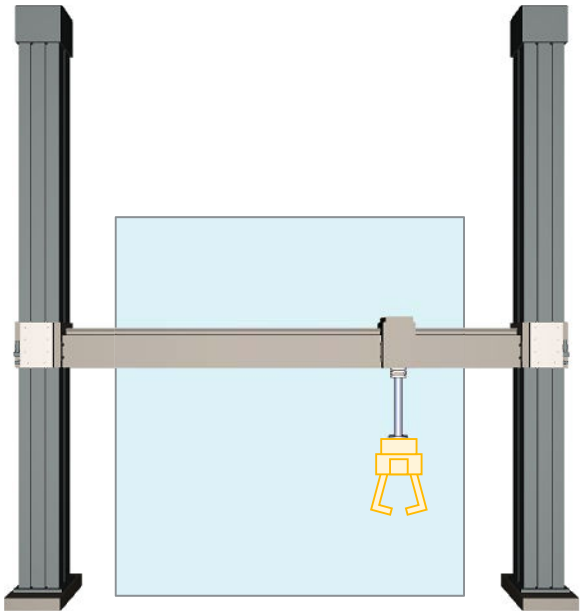

Die Kinematik besteht aus zwei orthogonalen linearen Achsen A1 und A2. Die Achsen umschließen einen rechteckigen Arbeitsbereich.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive Auslenkung der Kinematik

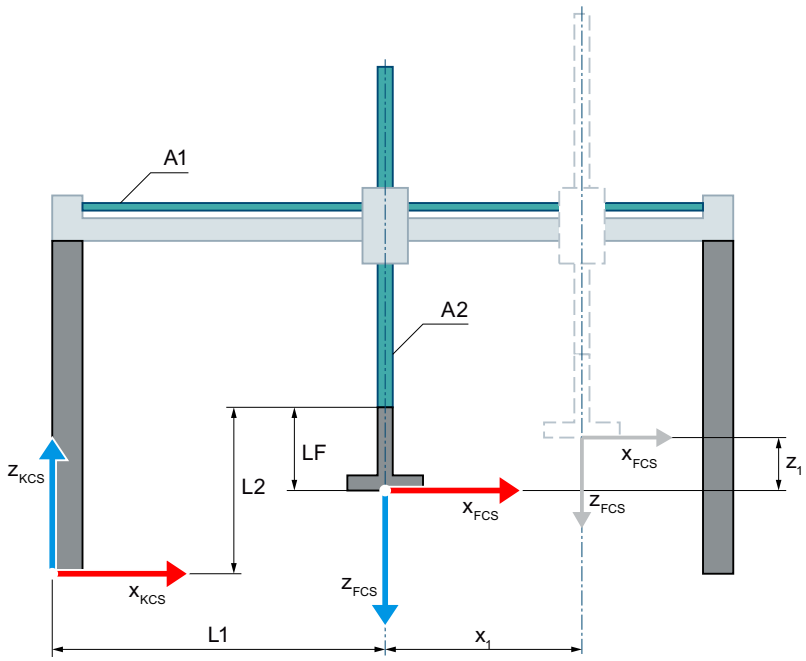

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum KNP in x-Richtung des KCS

L2 Bei Nullstellung der Achse A2:
Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

z_1 Auslenkung der Achse A2 in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt im Abstand LF zur Nullposition der Achse A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullpositionen der Achsen A1 und A2 im KCS. Über die Längen L1 und L2 definieren Sie die Abstände der Nullpositionen der Achsen gegenüber dem Kinematiknullpunkt.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Portal 2D mit Orientierung

Die Kinematik "Portal 2D mit Orientierung" unterstützt drei Achsen und drei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

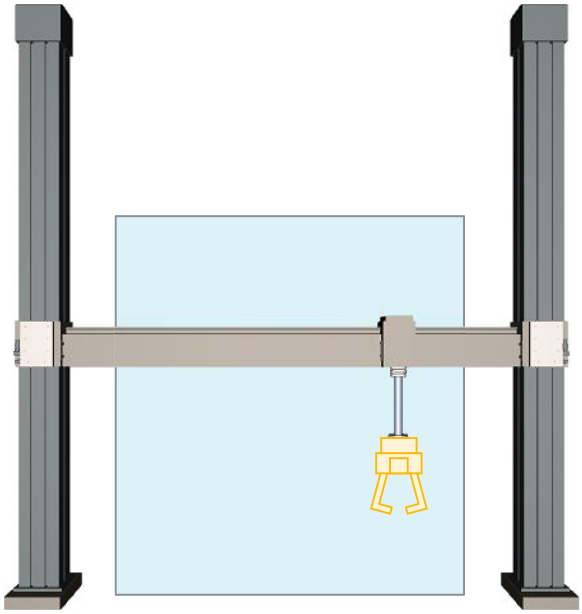

Die Kinematik besteht aus folgenden Achsen:

- Zwei orthogonale lineare Achsen A1 und A2
- Eine rotatorische Achse A4 (Orientierungsachse)

Die linearen Achsen umschließen einen rechteckigen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive Auslenkung der Kinematik

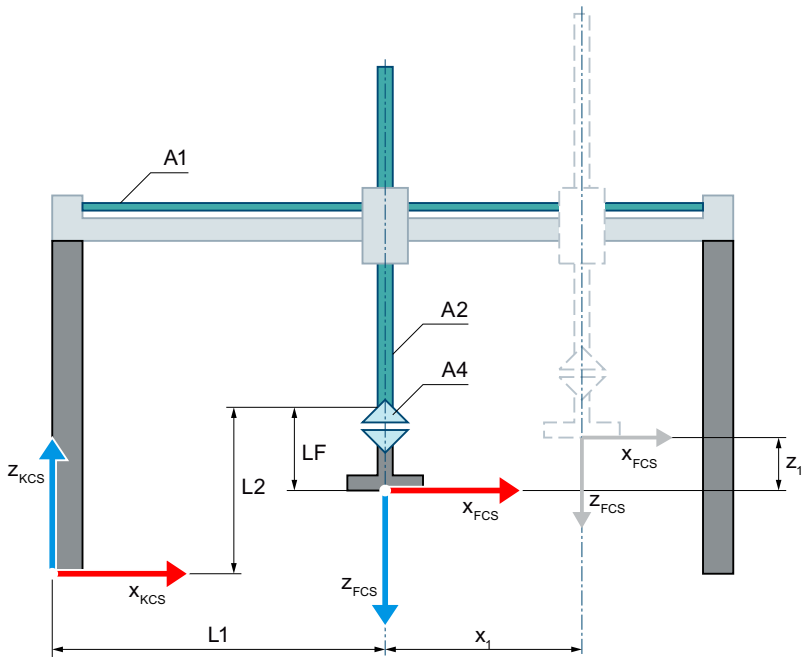

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum Kinematiknullpunkt (KNP) in x-Richtung des KCS

L2 Bei Nullstellung der Achse A2:
Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

z_1 Auslenkung der Achse A2 in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt im Abstand LF zur Nullposition der Achse A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullpositionen der Achsen A1 und A2 im KCS. Mit den Längen L1 und L2 definieren Sie die Abstände der Nullpositionen der Achsen A1 und A2 zum Kinematiknullpunkt. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Portal 3D

Die Kinematik "Portal 3D" unterstützt drei Achsen und drei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

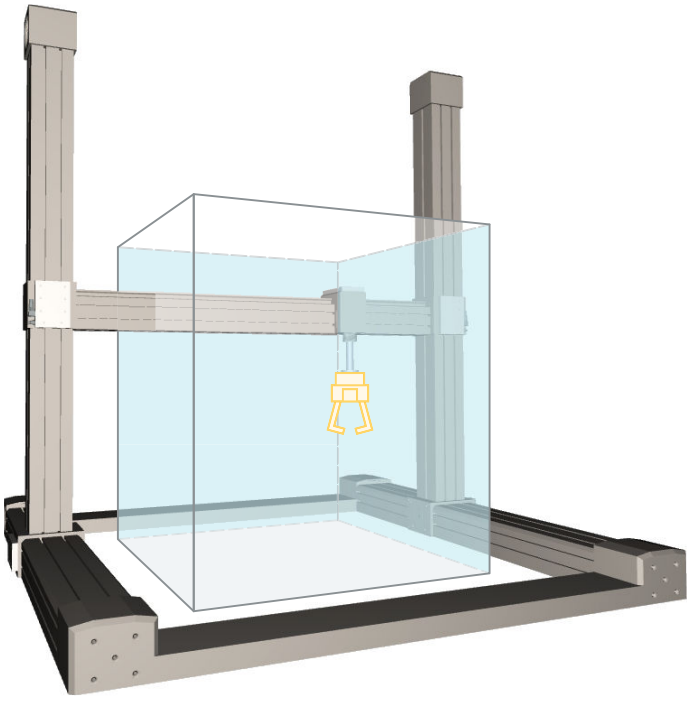

Die Kinematik besteht aus drei orthogonalen linearen Achsen A1, A2 und A3. Die linearen Achsen umschließen einen quaderförmigen Arbeitsbereich.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achsen A1 und A3

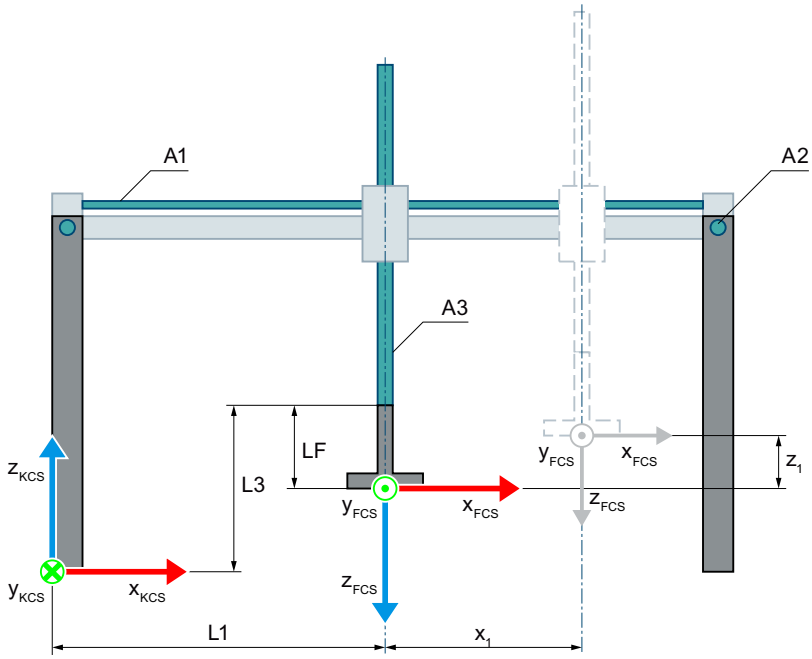

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum Kinematiknullpunkt (KNP) in x-Richtung des KCS

L3 Bei Nullstellung der Achse A3:
Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

z_1 Auslenkung der Achse A3 in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achsen A1 und A2

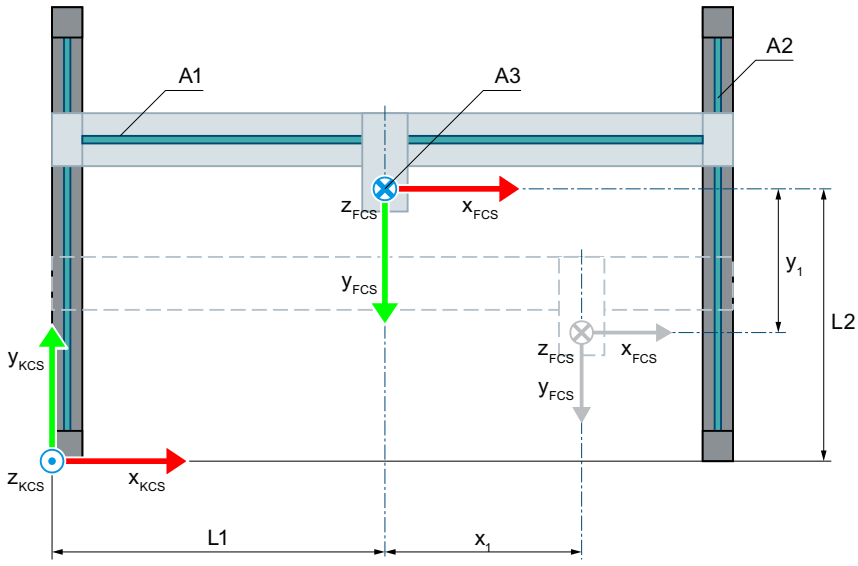

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum KNP in x-Richtung des KCS

L2 Bei Nullstellung der Achse A2:
Abstand des FCS zum KNP in y-Richtung des KCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

y_1 Auslenkung der Achse A2 in negativer y-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt im Abstand LF zur Nullposition der Achse A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullpositionen der Achsen A1, A2 und A3 im KCS. Mit den Längen L1, L2 und L3 definieren Sie die Abstände der Nullpositionen der Achsen zum Kinematiknullpunkt.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Portal 3D mit Orientierung

Die Kinematik "Portal 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

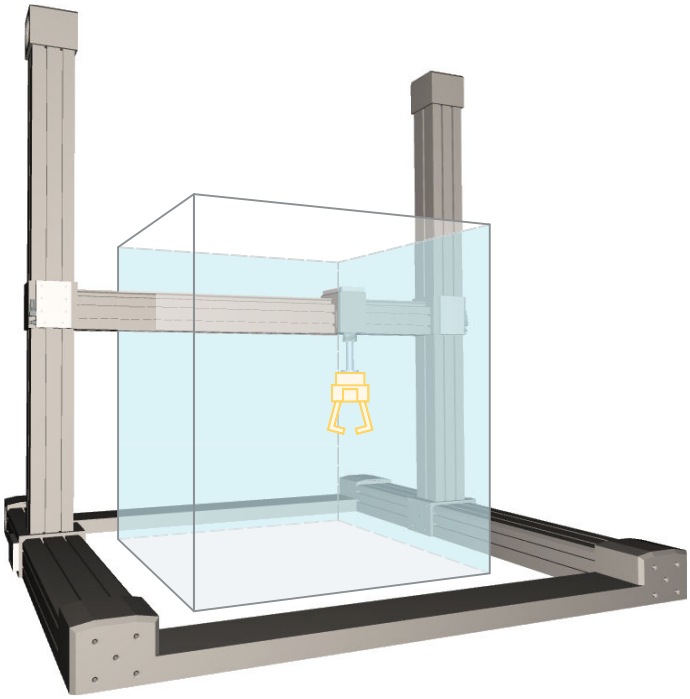

Die Kinematik besteht aus folgenden Achsen:

- Drei orthogonale lineare Achsen A1, A2 und A3
- Eine rotatorische Achse A4 (Orientierungsachse)

Die linearen Achsen umschließen einen quaderförmigen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achsen A1 und A3

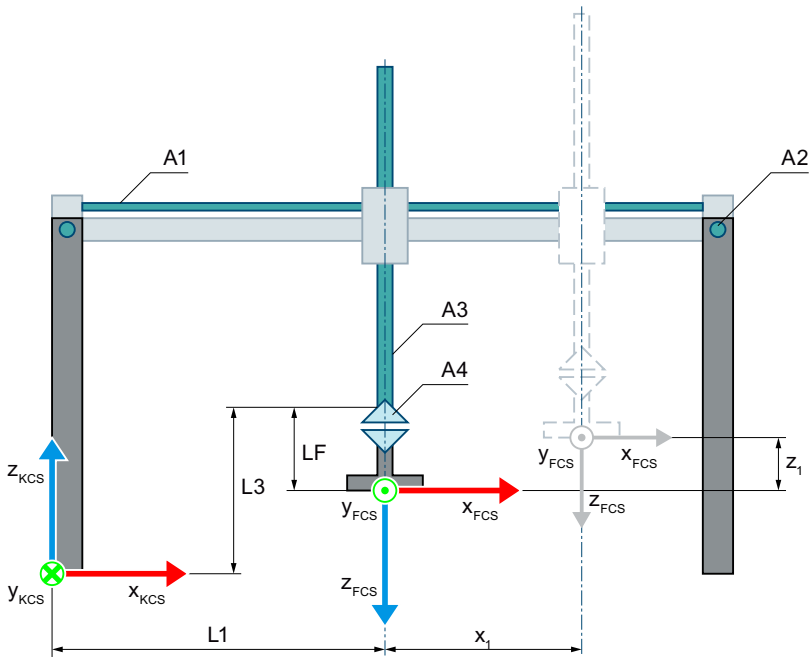

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum Kinematiknullpunkt (KNP) in x-Richtung des KCS

L3 Bei Nullstellung der Achse A3:
Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

z_1 Auslenkung der Achse A3 in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achsen A1 und A2

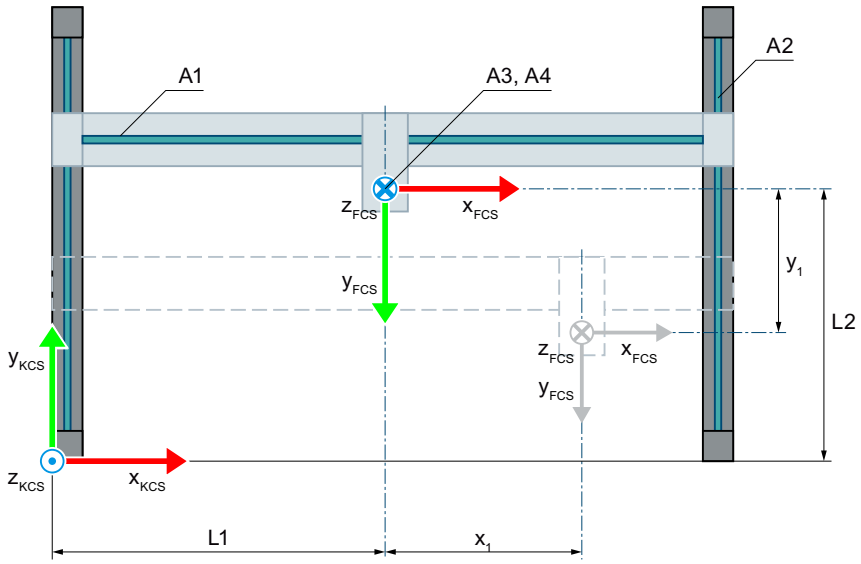

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A1:
Abstand des FCS zum KNP in x-Richtung des KCS

L2 Bei Nullstellung der Achse A2:
Abstand des FCS zum KNP in y-Richtung des KCS

Auslenkung der Kinematik

x_1 Auslenkung der Achse A1 in positiver x-Richtung

y_1 Auslenkung der Achse A2 in negativer y-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt im Abstand LF zur Nullposition der Achse A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullpositionen der Achsen A1, A2 und A3 im KCS. Mit den Längen L1, L2 und L3 definieren Sie die Abstände der Nullpositionen der Achsen A1, A2 und A3 zum Kinematiknullpunkt. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Variablen Portal

Portal 2D

Die 2D Portal-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	1	Portal 2D
	2	Portal 2D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Nullposition der Achse A1 zum Kinematik-nullpunkt (KNP) in x-Richtung des Kinematikkoordinaten-systems (KCS)
<TO>.Kinematics.Parameter[2]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der Achse A2 in negativer z-Richtung des KCS
<TO>.Kinematics.Parameter[3]	-1.0E12 bis 1.0E12	Abstand L2 der Nullposition der Achse A2 zum KNP in z-Richtung des KCS

Portal 3D

Die 3D Portal-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	3	Portal 3D
	4	Portal 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Nullposition der Achse A1 zum KNP in x-Richtung des KCS
<TO>.Kinematics.Parameter[2]	-1.0E12 bis 1.0E12	Abstand L2 der Nullposition der Achse A2 zum KNP in y-Richtung des KCS
<TO>.Kinematics.Parameter[3]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der Achse A3 in negativer z-Richtung des KCS
<TO>.Kinematics.Parameter[4]	-1.0E12 bis 1.0E12	Abstand L3 der Nullposition der Achse A3 zum KNP in z-Richtung des KCS

Siehe auch

Variablen des Technologieobjekts Kinematik (Seite 2387)

Rollen-Picker

Rollen-Picker 2D

Die Kinematik "Rollen-Picker 2D" unterstützt zwei Achsen und zwei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

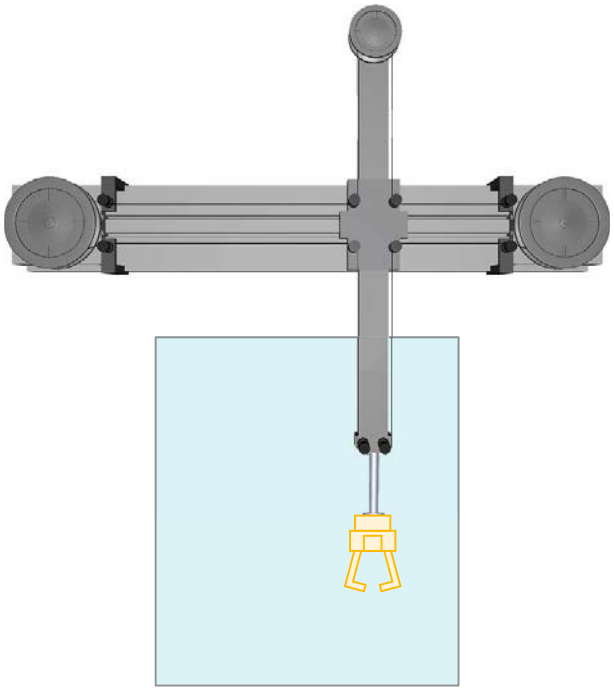

Die Kinematik besteht aus zwei rotatorischen Achsen A1 und A2 und einem System aus Umlenkrollen. Wenn beide Achsen A1 und A2 in dieselbe Richtung drehen, bewegt sich der Flansch horizontal in x-Richtung des KCS. Wenn beide Achsen A1 und A2 in entgegengesetzte Richtungen drehen, bewegt sich der Flansch vertikal in z-Richtung des KCS. Die Kinematik ermöglicht einen rechteckigen Arbeitsbereich.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

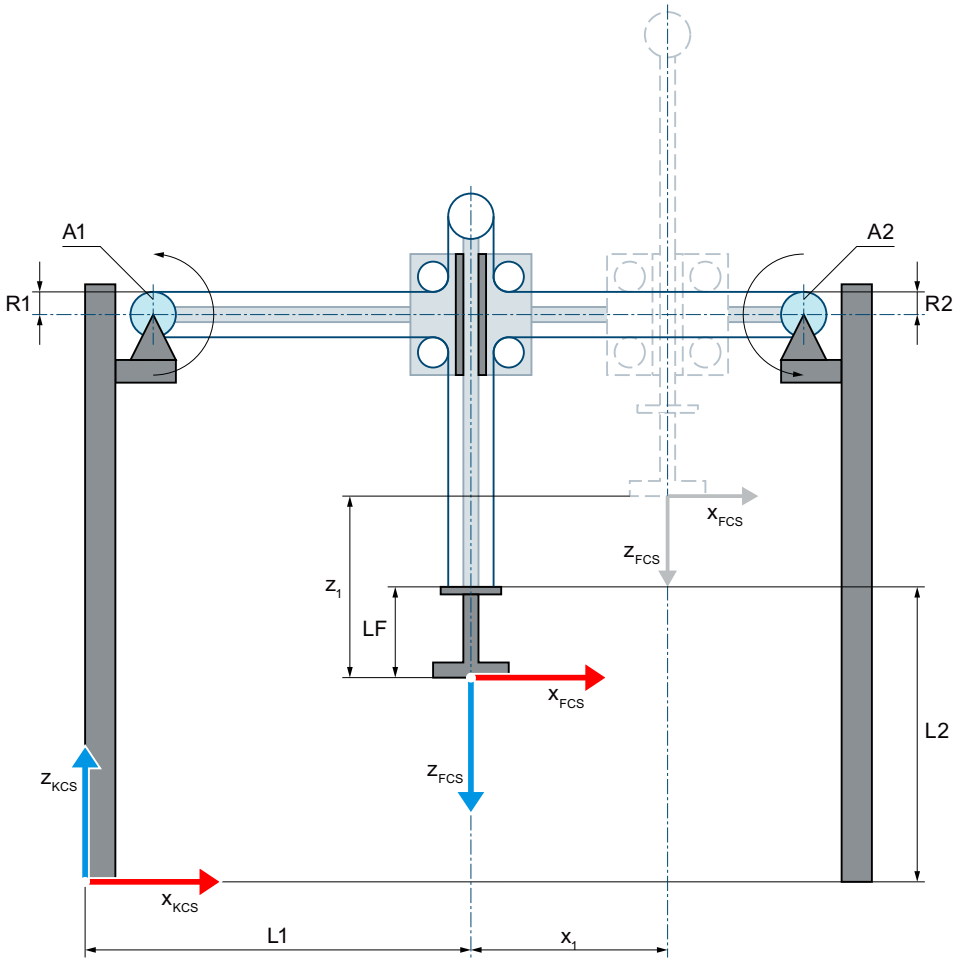

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L2	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
z_1	Auslenkung der Kinematik in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt zwischen den Achsen A1 und A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullstellung der Achsen A1 und A2. Über die Abstände L1 und L2 definieren Sie die Lage des FCS bei Nullstellung der Achsen A1 und A2. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Rollen-Picker 2D mit Orientierung

Die Kinematik "Rollen-Picker 2D mit Orientierung" unterstützt drei Achsen und drei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

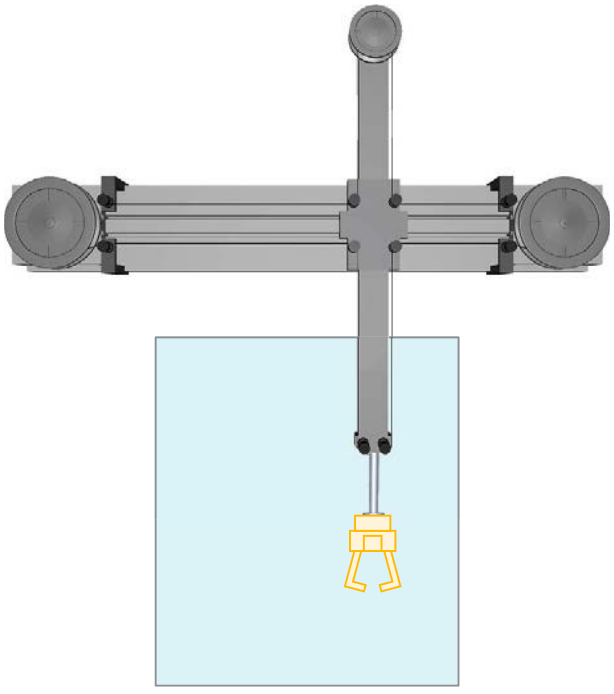

Die Kinematik besteht aus einem System aus Umlenkrollen und folgenden Achsen:

- Zwei rotatorische Achsen A1 und A2
- Eine rotatorische Achse A4 (Orientierungsachse) mit Drehung um z im KCS

Wenn beide Achsen A1 und A2 in dieselbe Richtung drehen, bewegt sich der Flansch horizontal in x-Richtung des KCS. Wenn beide Achsen A1 und A2 in entgegengesetzte Richtungen drehen, bewegt sich der Flansch vertikal in z-Richtung des KCS. Die Kinematik ermöglicht einen rechteckigen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

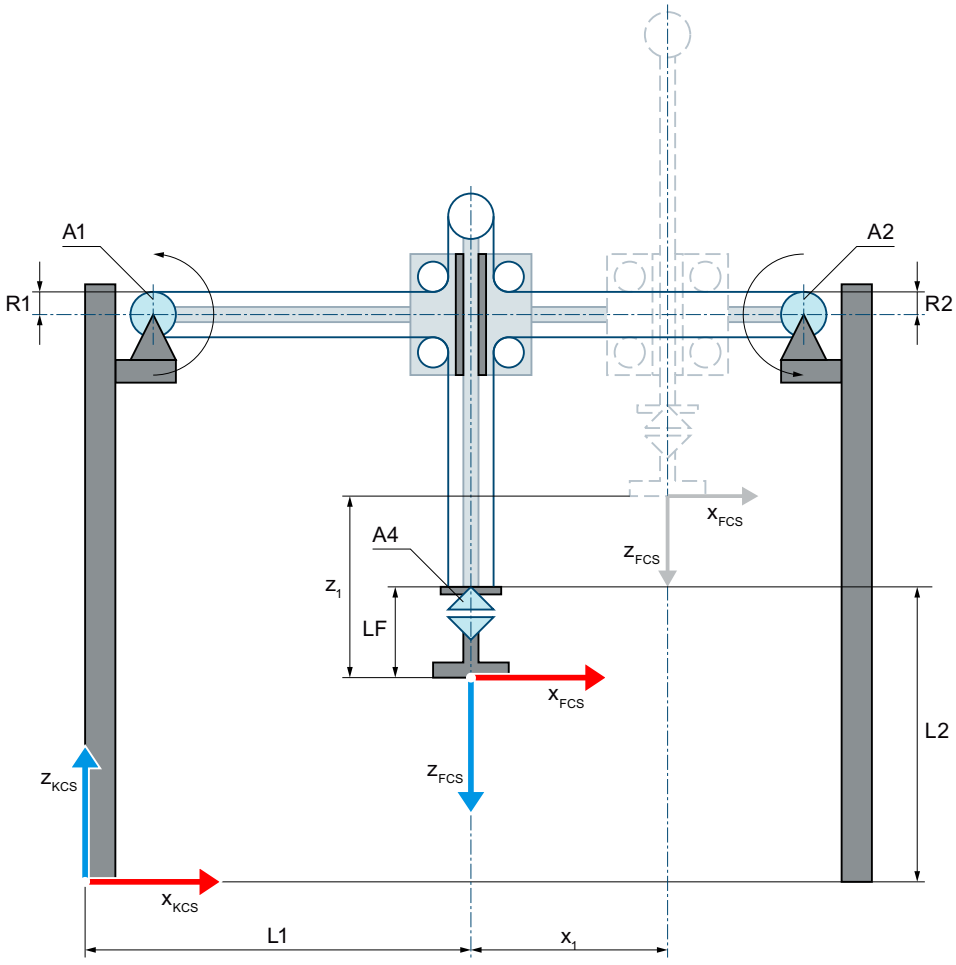

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L2	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
z_1	Auslenkung der Kinematik in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt zwischen den Achsen A1 und A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullstellung der Achsen A1 und A2. Über die Längen L1 und L2 definieren Sie die Lage des FCS bei Nullstellung der Achsen A1 und A2. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Rollen-Picker 3D (vertikal)

Die Kinematik "Rollen-Picker 3D (vertikal)" unterstützt drei Achsen und drei Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

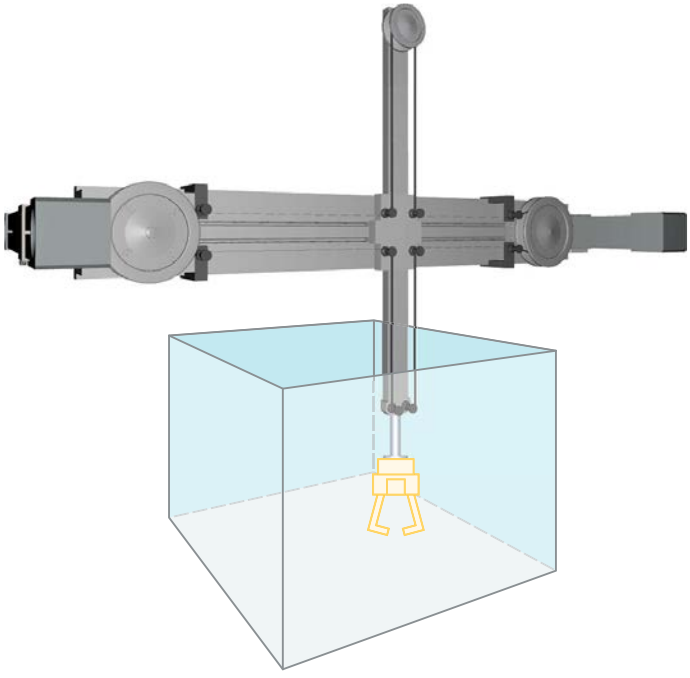

Die Kinematik besteht aus einem System aus Umlenkrollen und folgenden Achsen:

- Zwei rotatorische Achsen A1 und A2
- Eine lineare Achse A3 in y-Richtung des KCS

Wenn beide Achsen A1 und A2 in dieselbe Richtung drehen, bewegt sich der Flansch horizontal in x-Richtung des KCS. Wenn beide Achsen A1 und A2 in entgegengesetzte Richtungen drehen, bewegt sich der Flansch vertikal in z-Richtung des KCS. Die lineare Portalachse A3 bewegt das System aus Umlenkrollen horizontal in y-Richtung des KCS. Die Kinematik ermöglicht einen quaderförmigen Arbeitsbereich.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

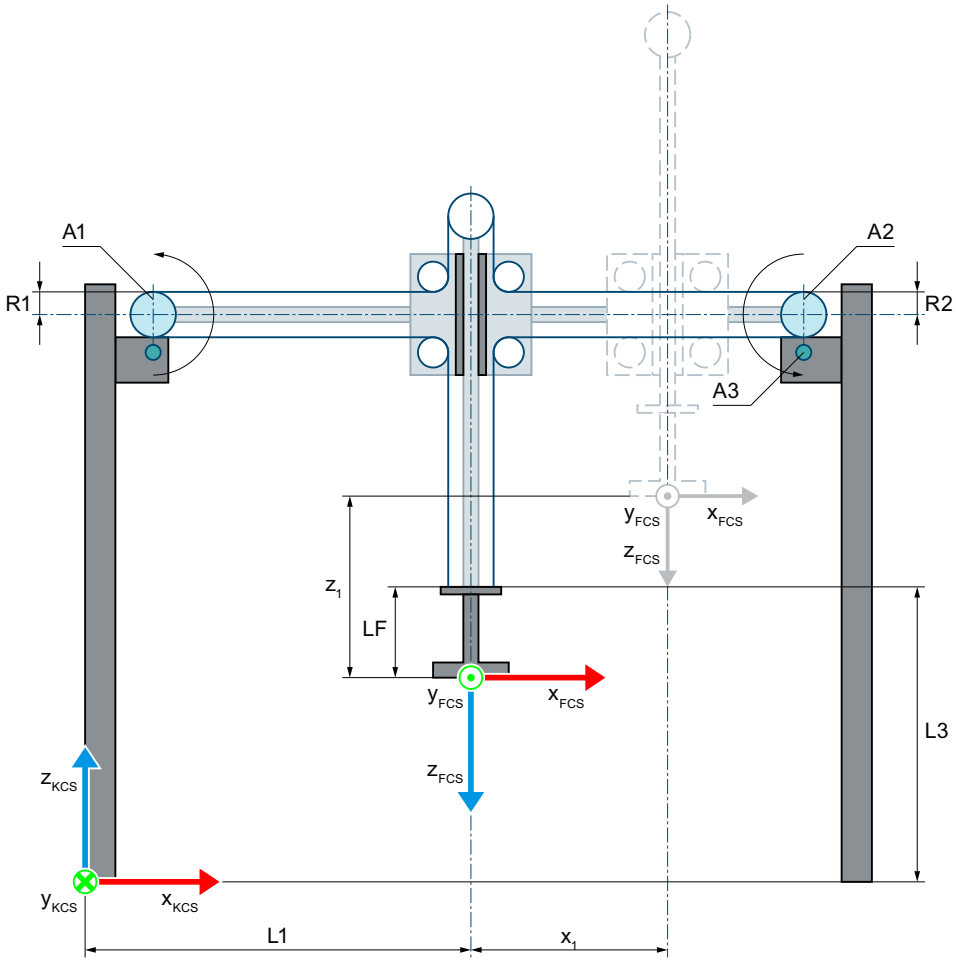

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L3	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
z_1	Auslenkung der Kinematik in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

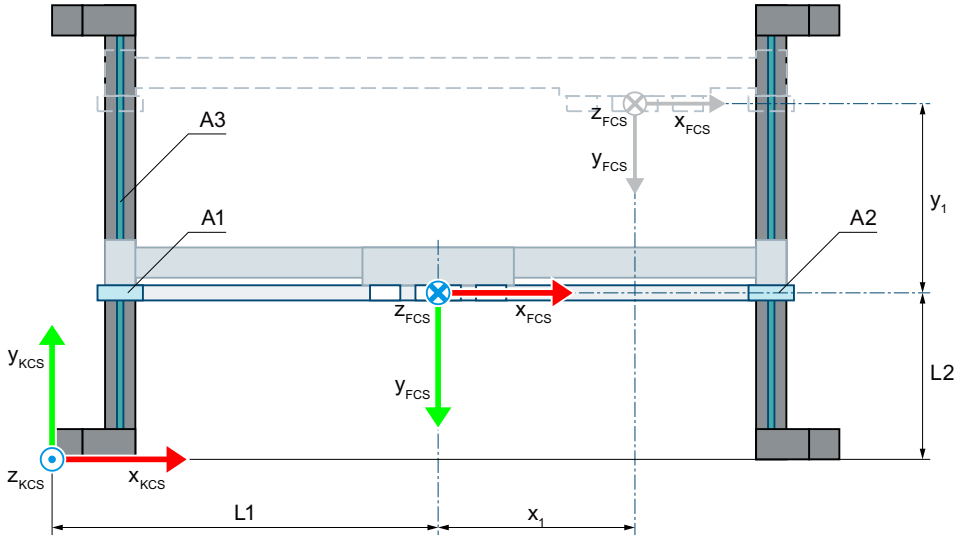

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L2	Bei Nullstellung der Achse A3: Abstand des FCS zum KNP in y-Richtung des KCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
y_1	Auslenkung der Kinematik in positiver y-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt zwischen den Achsen A1 und A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullstellung der Achsen A1 und A2 sowie die Nullposition der Achse A3 im KCS. Über die Länge L2 definieren Sie den Abstand der Nullposition der Achse A3 zum KNP in y-Richtung des KCS.

Über die Längen L1 und L3 definieren Sie die Lage des FCS bei Nullstellung der Achsen A1 und A2. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Rollen-Picker 3D mit Orientierung (vertikal)

Die Kinematik "Rollen-Picker 3D mit Orientierung (vertikal)" unterstützt vier Achsen und vier Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

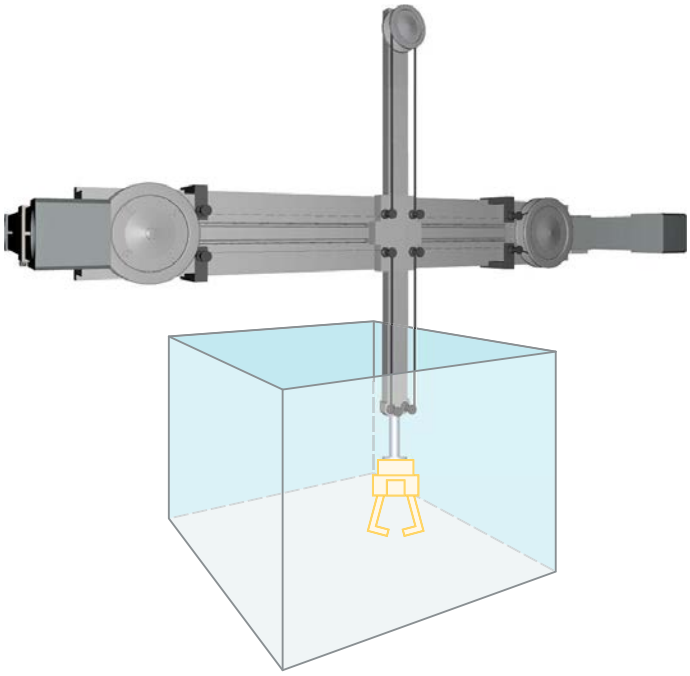

Die Kinematik besteht aus einem System aus Umlenkrollen und folgenden Achsen:

- Zwei rotatorische Achsen A1 und A2
- Eine lineare Achse A3 in y-Richtung des KCS
- Eine rotatorische Achse A4 (Orientierungsachse) mit Drehung um z im KCS

Wenn beide Achsen A1 und A2 in dieselbe Richtung drehen, bewegt sich der Flansch horizontal in x-Richtung des KCS. Wenn beide Achsen A1 und A2 in entgegengesetzte Richtungen drehen, bewegt sich der Flansch vertikal in z-Richtung des KCS. Die lineare Portalachse A3 bewegt das System aus Umlenkrollen horizontal in y-Richtung des KCS. Die Kinematik ermöglicht einen quaderförmigen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

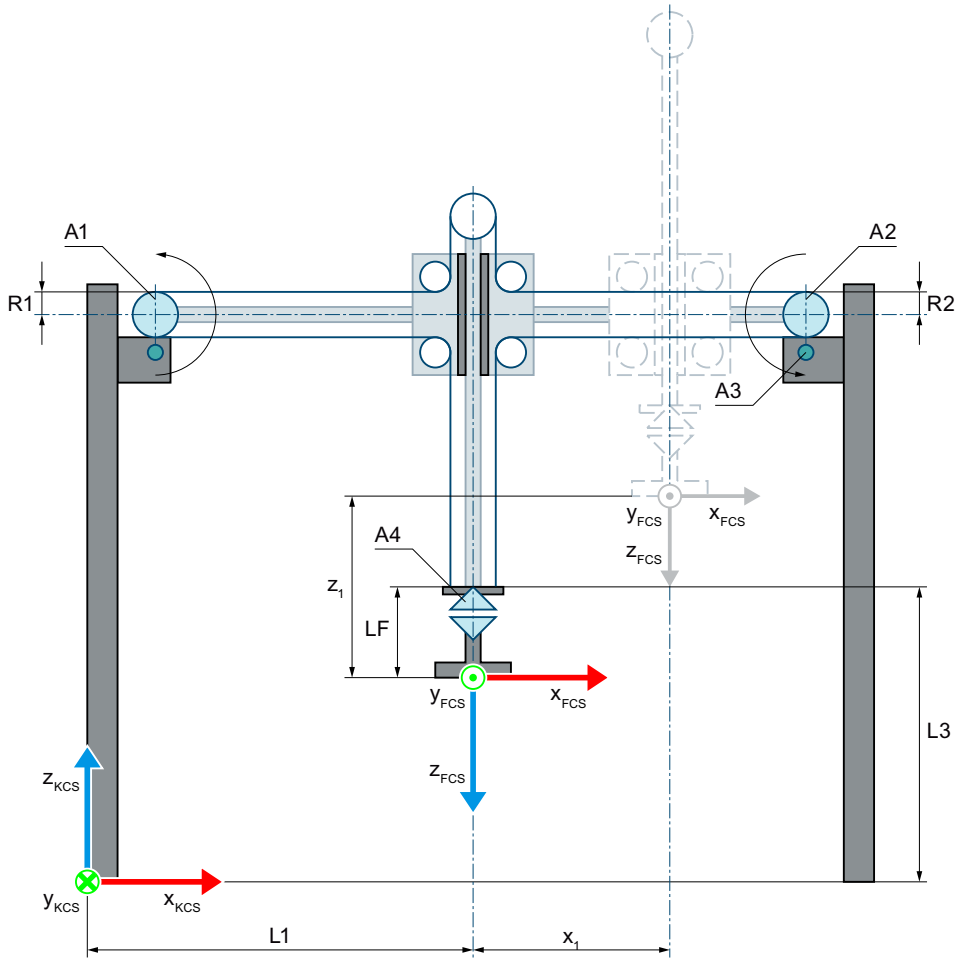

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L3	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
z_1	Auslenkung der Kinematik in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

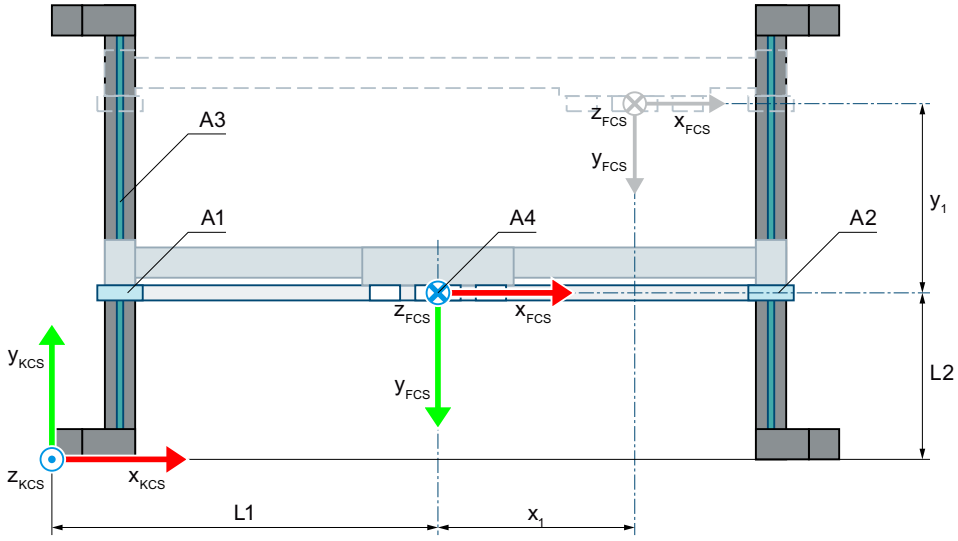

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L2	Bei Nullstellung der Achse A3: Abstand des FCS zum KNP in y-Richtung des KCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
y_1	Auslenkung der Kinematik in positiver y-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt zwischen den Achsen A1 und A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullstellung der Achsen A1 und A2 sowie die Nullposition der Achse A3 im KCS. Über die Länge L2 definieren Sie den Abstand der Nullposition der Achse A3 zum KNP in y-Richtung des KCS.

Über die Längen L1 und L3 definieren Sie die Lage des FCS bei Nullstellung der Achsen A1 und A2. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Rollen-Picker 3D mit Orientierung (horizontal)

Die Kinematik "Rollen-Picker 3D mit Orientierung (horizontal)" unterstützt vier Achsen und vier Freiheitsgrade. Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

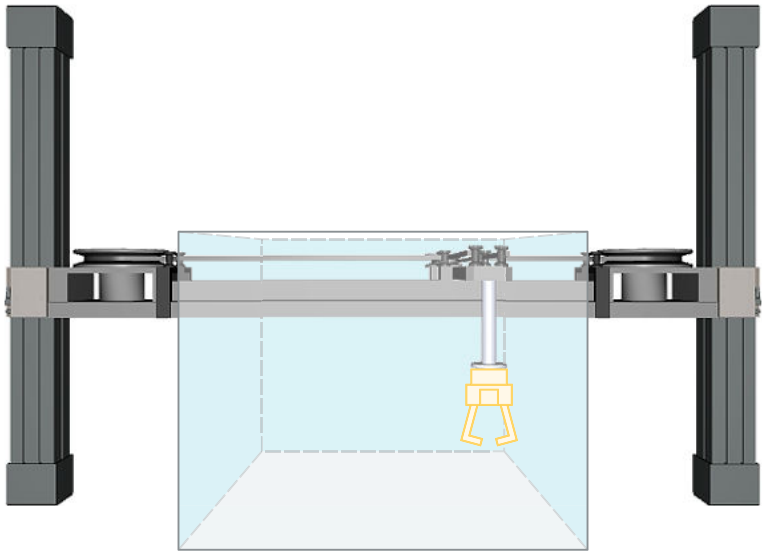

Die Kinematik besteht aus einem System aus Umlenkrollen und folgenden Achsen:

- Zwei rotatorische Achsen A1 und A2
- Eine lineare Achse A3 in z-Richtung des KCS
- Eine rotatorische Achse A4 (Orientierungsachse) mit Drehung um z im KCS

Wenn beide Achsen A1 und A2 in dieselbe Richtung drehen, bewegt sich der Flansch horizontal in x-Richtung des KCS. Wenn beide Achsen A1 und A2 in entgegengesetzte Richtungen drehen, bewegt sich der Flansch horizontal in y-Richtung des KCS. Die lineare Portalachse A3 bewegt das System aus Umlenkrollen vertikal in z-Richtung des KCS. Die Kinematik ermöglicht einen quaderförmigen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

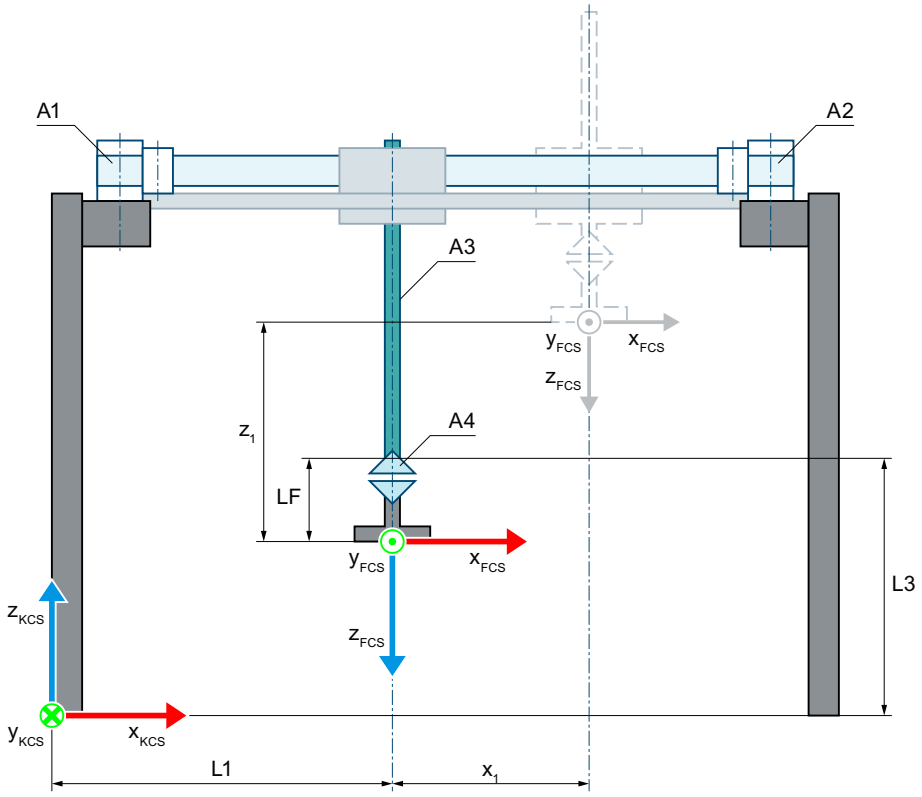

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L3	Bei Nullstellung der Achse A3: Abstand des FCS zum KNP und Flanschlänge LF in z-Richtung des KCS
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
z_1	Auslenkung der Kinematik in positiver z-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

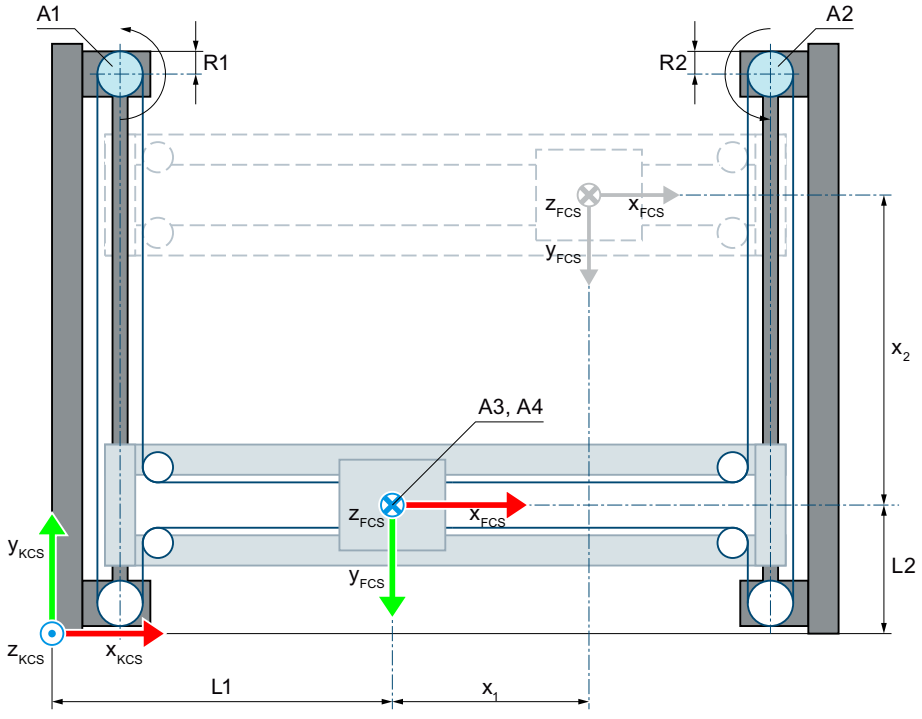

Nullstellung der Kinematik

L1	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in x-Richtung des KCS
L2	Bei Nullstellung der Achsen A1 und A2: Abstand des FCS zum KNP in y-Richtung des KCS
R1	Scheibenradius für Achse A1
R2	Scheibenradius für Achse A2

Auslenkung der Kinematik

x_1	Auslenkung der Kinematik in positiver x-Richtung
y_1	Auslenkung der Kinematik in positiver y-Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt zwischen den Achsen A1 und A2.

Die Position 0.0 am jeweilig verschalteten Technologieobjekt definiert die Nullstellung der Achsen A1 und A2 sowie die Nullposition der Achse A3 im KCS. Über die Länge L2 definieren Sie den Abstand der Nullposition der Achse A3 zum KNP in y-Richtung des KCS.

Über die Längen L1 und L3 definieren Sie die Lage des FCS bei Nullstellung der Achsen A1 und A2. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

Transformationsbereich

Die Kinematiktransformation deckt den gesamten Verfahrbereich (Seite 2316) der Achsen ab.

Variablen Rollen-Picker

Rollen-Picker 2D

Die 2D Rollen-Picker-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	5	Rollen-Picker 2D
	6	Rollen-Picker 2D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Bei Nullstellung der Achsen A1 und A2: Abstand L1 des FCS zum KNP in x-Richtung des Kinematikkoordinatensystems (KCS)
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Scheibenradius R1 für Achse 1
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Scheibenradius R2 für Achse 2
<TO>.Kinematics.Parameter[4]	-1.0E12 bis 1.0E12	Flanschlänge LF vor dem Flanschkoordinatensystem (FCS) in negativer z-Richtung des KCS
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Bei Nullstellung der Achsen A1 und A2: Abstand L2 des FCS zum KNP in z-Richtung des KCS

Rollen-Picker 3D

Die 3D Rollen-Picker-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung	
<TO>.Kinematics.TypeOfKinematics	7	Rollen-Picker 3D (vertikal)	
	8	Rollen-Picker 3D mit Orientierung (vertikal)	
	9	Rollen-Picker 3D mit Orientierung (horizontal)	
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Bei Nullstellung der Achsen A1 und A2: Abstand L1 des FCS zum KNP in x-Richtung des KCS	
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Scheibenradius R1 für Achse 1	
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Scheibenradius R2 für Achse 2	
<TO>.Kinematics.Parameter[4]	-1.0E12 bis 1.0E12	Flanschlänge LF vor dem FCS in negativer z-Richtung des KCS	
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Rollen-Picker vertikal	Abstand L2 der Nullposition der Achse A3 zum KNP in y-Richtung des KCS
		Rollen-Picker horizontal	Bei Nullstellung der Achsen A1 und A2: Abstand L2 des FCS zum Kinematiknullpunkt (KNP) in y-Richtung des KCS

Variablen	Werte	Beschreibung	
<TO>.Kinematics.Parameter[6]	-1.0E12 bis 1.0E12	Rollen-Picker vertikal	Bei Nullstellung der Achsen A1 und A2: Abstand L3 des FCS zum KNP in z-Richtung des KCS
		Rollen-Picker horizontal	Abstand L3 der Nullposition der Achse A3 zum KNP in z-Richtung des KCS

SCARA

SCARA 3D mit Orientierung

Die Kinematik "SCARA (Selective Compliance Assembly Robot Arm) 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die Achsen sind als serielle Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

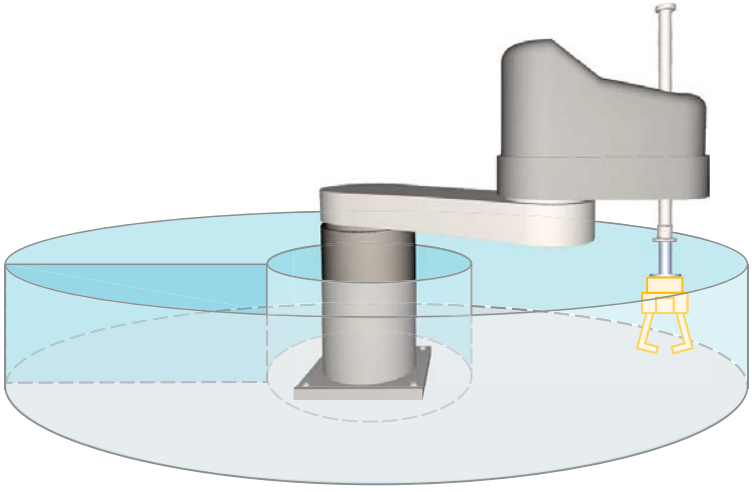

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit Drehung um die z-Achse des Kinematikkoordinatensystems (KCS)
- Eine rotatorische Achse A2 im Abstand L2 zu A1 mit Drehung um z des KCS
- Eine lineare Achse A3 im Abstand L3 zu A2 mit Bewegung in z-Richtung des KCS
- Eine rotatorische Achse A4 (Orientierungsachse) mit Drehung um z im KCS

Die Kinematik besteht aus einer Basis und zwei Hebeln zur horizontalen Ausrichtung, die durch Drehgelenke (Achse A1 und A2) verbunden sind. Am Ende dieses Gelenkarms ist für die vertikale Ausrichtung eine lineare Achse (Achse A3) befestigt. Das Werkzeug ist am Ende der linearen Achse befestigt. Die Orientierungsachse A4 ermöglicht die rotatorische Bewegung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik

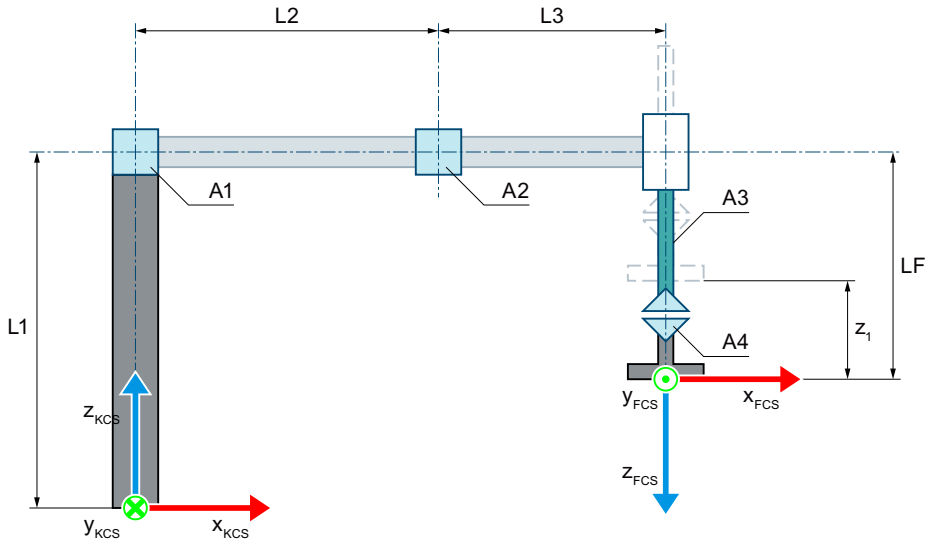

Nullstellung der Kinematik

L1	Abstand der Achse A1 zum KNP in z-Richtung des KCS
L2	Abstand der Achse A2 zu der Achse A1 in x-Richtung des KCS
L3	Abstand der Achse A3 zu der Achse A2 in x-Richtung des KCS
LF	Abstand des FCS zu der Achse A2 in z-Richtung des FCS

Auslenkung der Kinematik

z_1	Auslenkung der Achse A3 in positiver Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

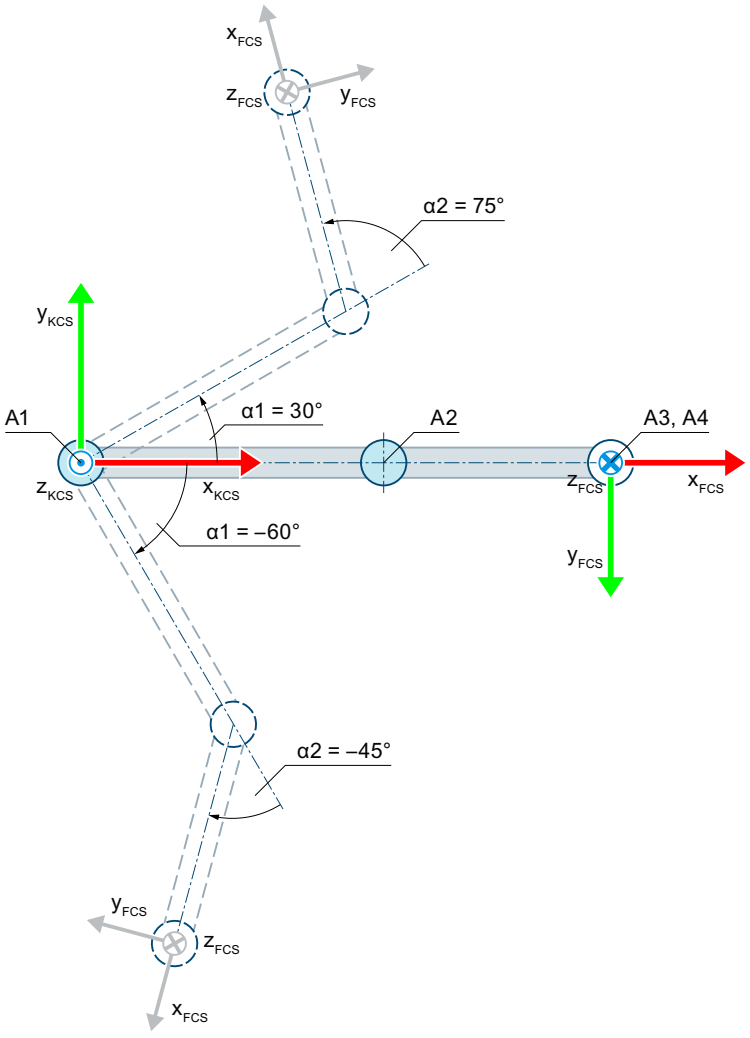

Nullstellung der Kinematik

Auslenkung der Kinematik in positiver Richtung bei α1 = 30.0° mit positiver Gelenkstellung bei α2 = 75.0°

Auslenkung der Kinematik in negativer Richtung bei α1 = -60.0° mit negativer Gelenkstellung bei α2 = -45.0°

α1 Auslenkung der Achse A1 in positiver Richtung bei α1 = 30.0°

Auslenkung der Achse A1 in negativer Richtung bei α1 = -60.0°

α2 Die Auslenkung der Achse A2 in positiver Richtung bei α2 = 75.0° ergibt eine positive Gelenkstellung.

Die Auslenkung der Achse A2 in negativer Richtung bei α2 = -45.0° ergibt eine negative Gelenkstellung.

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Das Flanschkoordinatensystem (FCS) liegt am Ende der Achse A3.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1 und A2	Die Kinematik ist in x_{KCS}-Richtung gestreckt.
A3	Das FCS ist um den Abstand L1-LF vom KCS in z-Richtung entfernt.
A4	Bei Nullstellung von Achse A1 und A2 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie folgende mechanische Achskopplungen konfigurieren:

- Mechanische Kopplung von Achse A1 auf Achse A2
- Mechanische Kopplung von Achse A4 auf Achse A3

Die Kinematiktransformation kompensiert die konfigurierten mechanischen Achskopplungen. Bei einem Kopplungsfaktor > 0.0 geht die Kinematiktransformation davon aus, dass eine positive Bewegung der Achse A1 zu einer negativen Bewegung an der Achse A2 führt. Die Achskopplung zwischen der Achse A4 und der Achse A3 ist als Spindelsteigung realisiert. Bei einem Kopplungsfaktor von 1.0 entsprechen 360.0° an der Achse A4 einem Weg von -1.0 mm an der Achse A3.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: -180.0° ≤ α1 ≤ 180.0°
- Achse A2: -180.0° ≤ α2 ≤ 180.0°
- Achse A3: Keine Begrenzung
- Achse A4: Keine Begrenzung
  Für die Orientierung kann ein Winkel größer als 360° vorgegeben werden. Aber die Koordinate A des Werkzeugarbeitspunkts (TCP) wird im Bereich -180° bis +180° abgebildet.

Hinweis

Singuläre Stellungen

Die Kinematik besitzt singuläre Stellungen (Seite 2318).

Eine singuläre Stellung liegt vor, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. In diesem Bereich ist die Rückwärtstransformation nicht möglich. Diese Position kann sich z. B. bei einem hängenden Aufbau ergeben, wenn die Längen L2 und L3 gleichgroß sind.

Die folgende Grafik zeigt beispielhaft eine Bewegung in Richtung der singulären Gelenkstellung:

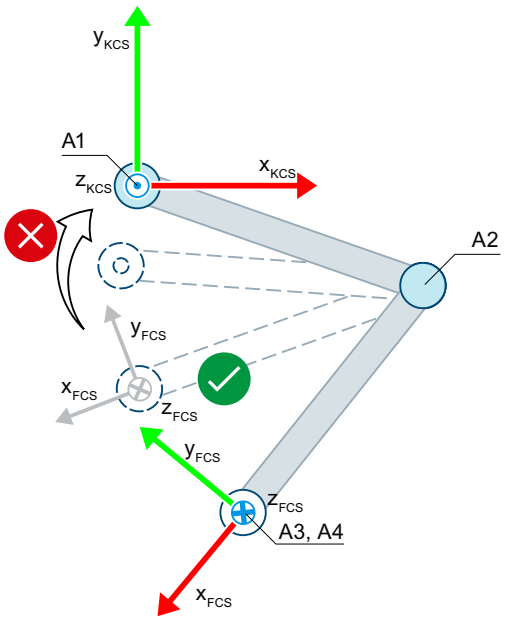

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation bei L2 = L3

Variablen SCARA

Die SCARA-Kinematik definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	10	SCARA 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand der Achse A1 vom Kinematiknullpunkt in z-Richtung des Kinematikkoordinatensystems (KCS)
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Abstand L2 der Achse A2 von der Achse A1 in x-Richtung des KCS
<TO>.Kinematics.Parameter[3]	-	Mechanische Achskopplung von Achse A1 auf Achse A2 vorhanden/nicht vorhanden
	0	Nicht vorhanden
	1	Vorhanden
<TO>.Kinematics.Parameter[4]	-1.0E12 bis 1.0E12	Mechanischer Achskopplungsfaktor von Achse A1 auf Achse A2
<TO>.Kinematics.Parameter[5]	0.001 bis 1.0E12	Abstand L3 der Achse A3 von der Achse A2 in x-Richtung des KCS
<TO>.Kinematics.Parameter[6]	-	Mechanische Achskopplung von Achse A4 auf Achse A3 vorhanden/nicht vorhanden
	0	Nicht vorhanden
	1	Vorhanden
<TO>.Kinematics.Parameter[7]	-1.0E12 bis 1.0E12	Mechanischer Achskopplungsfaktor von Achse A4 auf Achse A3
<TO>.Kinematics.Parameter[8]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der Achse A2 in negativer z-Richtung des KCS

Knickarm

Knickarm 2D

Die Kinematik "Knickarm 2D" unterstützt zwei Achsen und zwei Freiheitsgrade. Die Achsen sind als serielle Kinematik mit Zwangskopplung des Flanschsystems aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

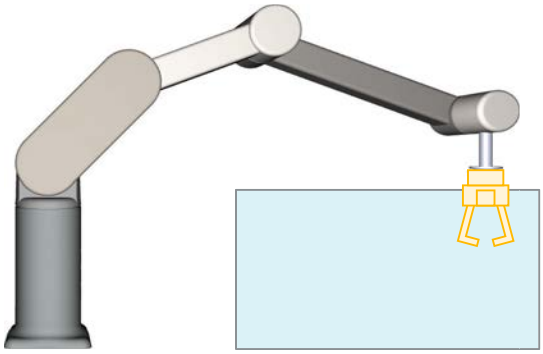

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit den Abständen L1 in z-Richtung des KCS und L2 in x-Richtung des KCS zum Kinematiknullpunkt
- Eine rotatorische Achse A2 im Abstand L3 zur Achse A1

Die Kinematik besteht aus einer Basis und Gelenkarmen, die durch Drehgelenke verbunden sind (Achsen A1, A2). Die Achsen A1 und A2 bewegen die Gelenkarme vertikal. Durch eine Zwangskopplung zwischen der Achse A2 und dem Flanschsystem zeigt die z-Achse des FCS immer in die negative z-Richtung des KCS.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht:

- Die Lage der Achsen und des Zwangskoppelpunkts
- Die Lage Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

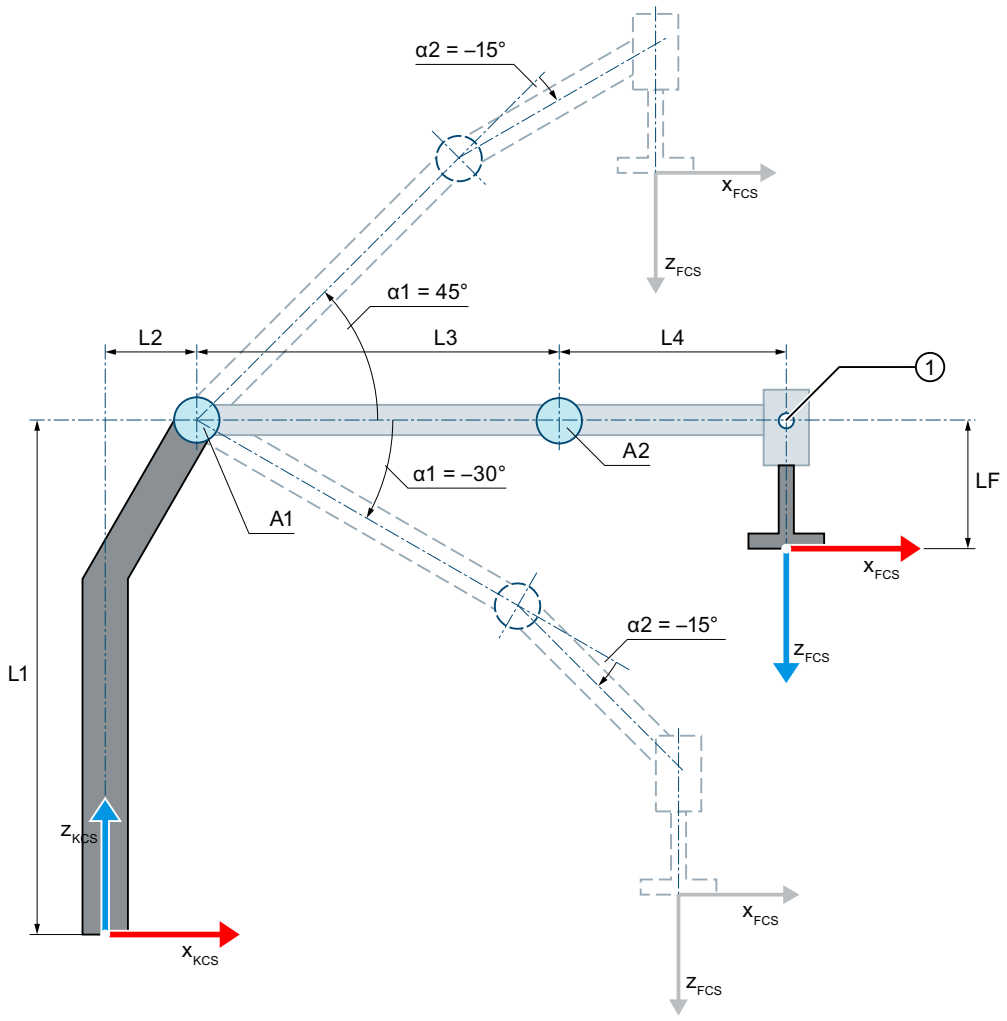

① Zwangskoppelpunkt

Nullstellung der Kinematik

- L1 Abstand der Achse A1 zum Kinematiknullpunkt (KNP) in z-Richtung des KCS
- L2 Abstand der Achse A1 zum KNP in x-Richtung des KCS
- L3 Abstand der Achse A2 zu der Achse A1 in x-Richtung des KCS
- L4 Abstand des Zwangskoppelpunkts zu der Achse A2 in x-Richtung des KCS
- LF Abstand des FCS zum Zwangskoppelpunkt in z-Richtung des FCS

Auslenkung der Kinematik

- α1 Positive Auslenkung der Achse A1 bei α1 = 45.0°
  Negative Auslenkung der Achse A1 bei α1 = -30.0°
- α2 Negative Auslenkung der Achse A2 bei α2 = -15.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Längen L1 und L2 definieren Sie die Lage der Achse A1 gegenüber dem KNP. Die Achse A2 liegt im Abstand L3 in x-Richtung des KCS zur Achse A1.

Das Flanschkoordinatensystem (FCS) liegt in folgenden Abständen zur Achse A2 und zum Zwangskoppelpunkt:

- Abstand L4 zur Achse A2 in x-Richtung des KCS
- Abstand LF zum Zwangskoppelpunkt in negativer z-Richtung des KCS

Die Achse A2 und das Flanschsystem sind zwangsgekoppelt. Mit der Zwangskopplung zeigt die z-Achse des FCS immer in negative z-Richtung des KCS. Der Zwangskoppelpunkt liegt im Abstand L4 in x-Richtung des KCS zur Achse A2.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Die Länge L3 zeigt in x-Richtung des KCS.
A2	Bei Nullstellung der Achse A1 zeigt die Länge L4 in x-Richtung des KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie eine mechanische Achskopplung von Achse A1 auf Achse A2 konfigurieren. Die Kinematiktransformation kompensiert die konfigurierte mechanische Achskopplung. Bei einem Kopplungsfaktor > 0.0 geht die Kinematiktransformation davon aus, dass eine positive Bewegung der Achse A1 zu einer negativen Bewegung an der Achse A2 führt.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: $-180.0° \leq \alpha 1 < 180.0°$
- Achse A2: $-180.0° \leq \alpha 2 < 180.0°$

Knickarm 2D mit Orientierung

Die Kinematik "Knickarm 2D mit Orientierung" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als serielle Kinematik mit Zwangskopplung des Flanschsystems aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

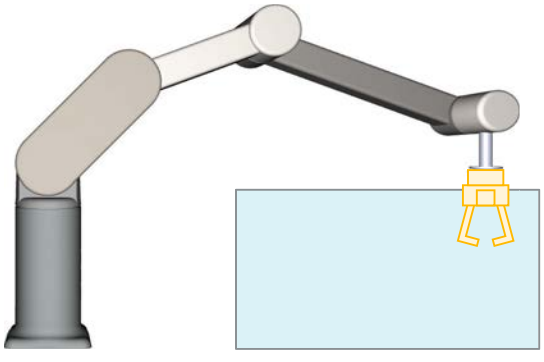

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit den Abständen L1 in z-Richtung des KCS und L2 in x-Richtung des KCS zum Kinematiknullpunkt
- Eine rotatorische Achse A2 im Abstand L3 zur Achse A1
- Eine rotatorische Achse A4 (Orientierungsachse) im Abstand L4 in x-Richtung des KCS zur Achse A2

Die Kinematik besteht aus einer Basis und Gelenkarmen, die durch Drehgelenke verbunden sind (Achsen A1, A2). Die Achsen A1 und A2 bewegen die Gelenkarme vertikal. Durch eine Zwangskopplung zwischen der Achse A2 und dem Flanschsystem zeigt die z-Achse des FCS immer in die negative z-Richtung des KCS. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht:

- Die Lage der Achsen und des Zwangskoppelpunkts
- Die Lage Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

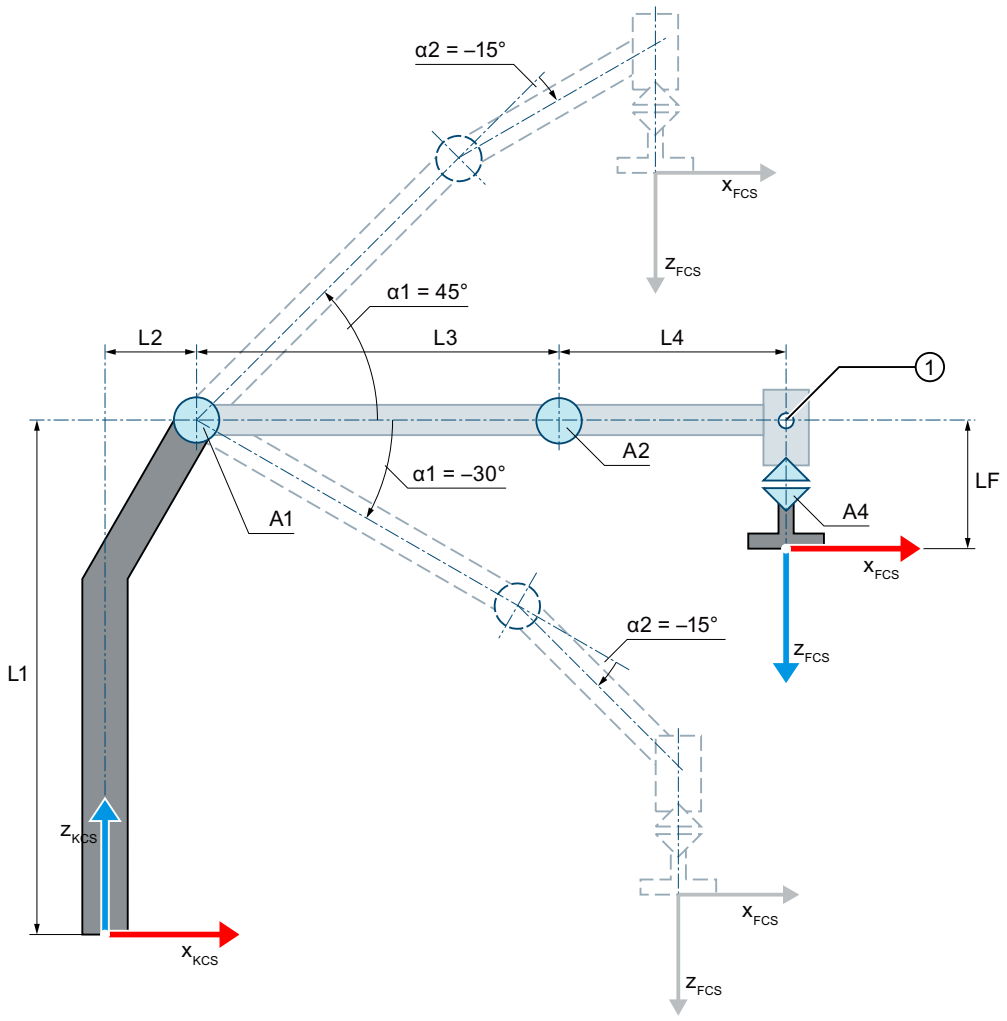

① Zwangskoppelpunkt

Nullstellung der Kinematik

- L1 Abstand der Achse A1 zum Kinematiknullpunkt (KNP) in z-Richtung des KCS
- L2 Abstand der Achse A1 zum KNP in x-Richtung des KCS
- L3 Abstand der Achse A2 zu der Achse A1 in x-Richtung des KCS
- L4 Abstand des Zwangskoppelpunkts zu der Achse A2 in x-Richtung des KCS
- LF Abstand des FCS zum Zwangskoppelpunkt in z-Richtung des FCS

Auslenkung der Kinematik

- α1 Positive Auslenkung der Achse A1 bei α1 = 45.0°
  Negative Auslenkung der Achse A1 bei α1 = -30.0°
- α2 Negative Auslenkung der Achse A2 bei α2 = -15.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Längen L1 und L2 definieren Sie die Lage der Achse A1 gegenüber dem KNP. Die Achse A2 liegt im Abstand L3 in x-Richtung des KCS zur Achse A1.

Das Flanschkoordinatensystem (FCS) liegt in folgenden Abständen zur Achse A2 und zum Zwangskoppelpunkt:

- Abstand L4 zur Achse A2 in x-Richtung des KCS
- Abstand LF zum Zwangskoppelpunkt in negativer z-Richtung des KCS

Die Achse A2 und das Flanschsystem sind zwangsgekoppelt. Mit der Zwangskopplung zeigt die z-Achse des FCS immer in negative z-Richtung des KCS. Der Zwangskoppelpunkt liegt im Abstand L4 in x-Richtung des KCS zur Achse A2.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Die Länge L3 zeigt in x-Richtung des KCS.
A2	Bei Nullstellung der Achse A1 zeigt die Länge L4 in x-Richtung des KCS.
A4	Bei Nullstellung der Achsen A1 und A2 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie eine mechanische Achskopplung von Achse A1 auf Achse A2 konfigurieren. Die Kinematiktransformation kompensiert die konfigurierte mechanische Achskopplung. Bei einem Kopplungsfaktor > 0.0 geht die Kinematiktransformation davon aus, dass eine positive Bewegung der Achse A1 zu einer negativen Bewegung an der Achse A2 führt.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: $-180.0° \leq \alpha 1 < 180.0°$
- Achse A2: $-180.0° \leq \alpha 2 < 180.0°$
- Achse A4: Keine Begrenzung
  Für die Orientierung kann ein Winkel größer als 360° vorgegeben werden. Aber die Koordinate A des Werkzeugarbeitspunkts (TCP) wird im Bereich -180° bis +180° abgebildet.

Knickarm 3D

Die Kinematik "Knickarm 3D" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als serielle Kinematik mit Zwangskopplung des Flanschsystems aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

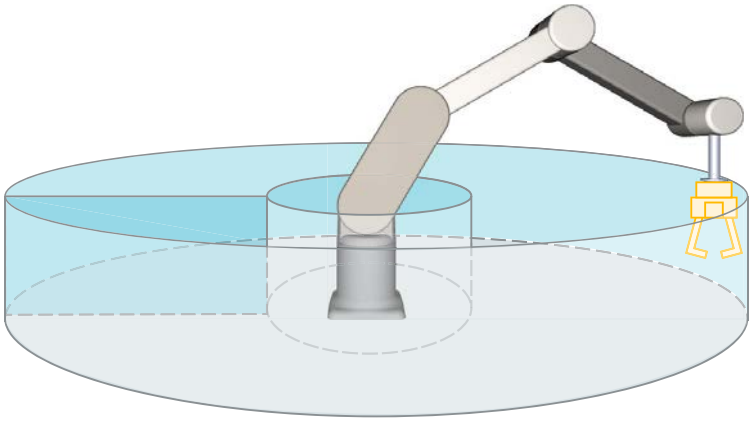

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit Drehung um die z-Achse des Kinematikkoordinatensystems (KCS)
- Eine rotatorische Achse A2 mit den Abständen L1 in z-Richtung des KCS und L2 in x-Richtung des KCS zum Kinematiknullpunkt
- Eine rotatorische Achse A3 im Abstand L3 zur Achse A2

Die Kinematik besteht aus einer Basis und Gelenkarmen, die durch Drehgelenke verbunden sind (Achsen A1, A2 und A3). Die Achse A1 dreht die Kinematik horizontal um die Basis. Die Achsen A2 und A3 bewegen die Gelenkarme vertikal. Die Kinematik ermöglicht einen annähernd kugelförmigen Arbeitsbereich. Durch eine Zwangskopplung zwischen der Achse A2 und dem Flanschsystem zeigt die z-Achse des FCS immer in die negative z-Richtung des KCS.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und des Zwangskoppelpunkts
- Die Lage Koordinatensysteme KCS und FCS
- Die Nullstellung der Achsen
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achsen A2 und A3

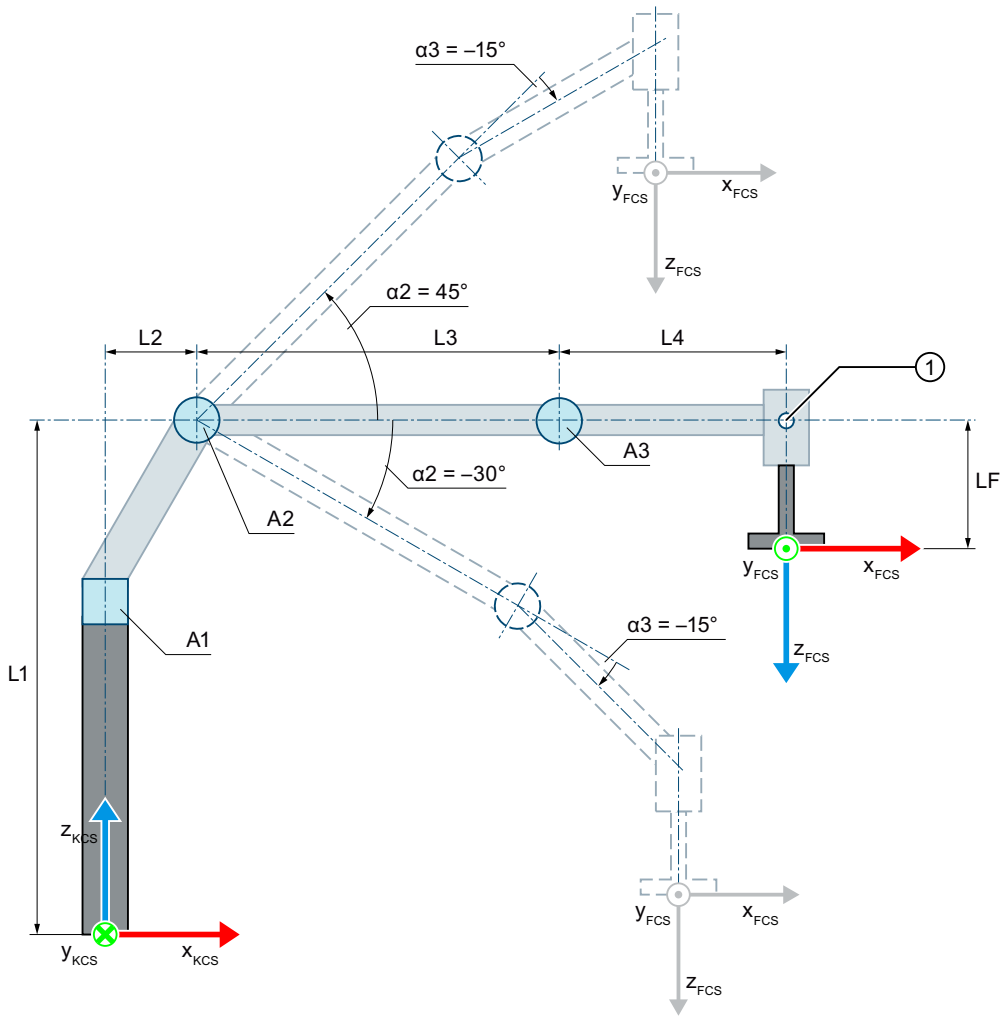

① Zwangskoppelpunkt

Nullstellung der Kinematik

- L1 Abstand der Achse A2 zum Kinematiknullpunkt (KNP) in z-Richtung des KCS
- L2 Abstand der Achse A2 zum KNP in x-Richtung des KCS
- L3 Abstand der Achse A3 zu der Achse A2 in x-Richtung des KCS
- L4 Abstand des Zwangskoppelpunkts zu der Achse A3 in x-Richtung des KCS
- LF Abstand des FCS zum Zwangskoppelpunkt in z-Richtung des FCS

Auslenkung der Kinematik

- α2 Positive Auslenkung der Achse A2 bei α2 = 45.0°
  Negative Auslenkung der Achse A2 bei α2 = -30.0°
- α3 Negative Auslenkung der Achse A3 bei α3 = -15.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung Kinematik

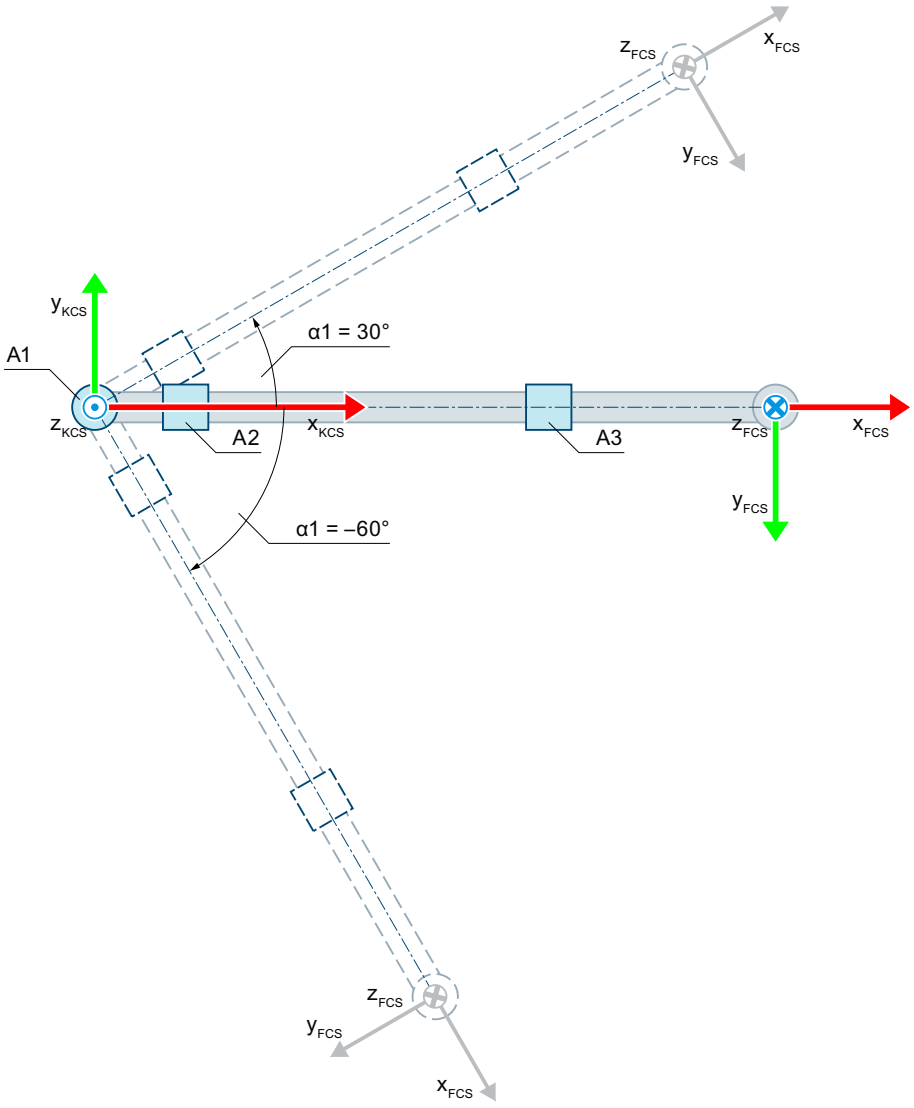

Nullstellung der Kinematik

Auslenkung der Kinematik

α1 Positive Auslenkung der Achse A1 bei α1 = 30.0°

Negative Auslenkung der Achse A1 bei α1 = -60.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Längen L1 und L2 definieren Sie die Lage der Achse A2 gegenüber dem KNP. Die Achse A3 liegt im Abstand L3 in x-Richtung des KCS zur Achse A2.

Das Flanschkoordinatensystem (FCS) liegt in folgenden Abständen zur Achse A3 und zum Zwangskoppelpunkt:

- Abstand L4 zur Achse A3 in x-Richtung des KCS
- Abstand LF zum Zwangskoppelpunkt in negativer z-Richtung des KCS

Die Achse A3 und das Flanschsystem sind zwangsgekoppelt. Mit der Zwangskopplung zeigt die z-Achse des FCS immer in negative z-Richtung des KCS. Der Zwangskoppelpunkt liegt im Abstand L4 in x-Richtung des KCS zur Achse A3.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Die Gelenkarme der Kinematik zeigen in x-Richtung des KCS.
A2	Bei Nullstellung von Achse A1 zeigt die Länge L3 in x-Richtung des KCS.
A3	Bei Nullstellung der Achsen A1 und A2 zeigt die Länge L4 in x-Richtung KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie eine mechanische Achskopplung von Achse A2 auf Achse A3 konfigurieren. Die Kinematiktransformation kompensiert die konfigurierte mechanische Achskopplung. Bei einem Kopplungsfaktor > 0.0 geht die Kinematiktransformation davon aus, dass eine positive Bewegung der Achse A2 zu einer negativen Bewegung an der Achse A3 führt.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: -180.0° ≤ α1 < 180.0°
- Achse A2: -180.0° ≤ α2 < 180.0°
- Achse A3: -180.0° ≤ α3 < 180.0°

Hinweis

Singuläre Stellungen

Die Kinematik besitzt singuläre Stellungen (Seite 2318).

Eine singuläre Stellung liegt vor, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. In diesem Bereich ist die Rückwärtstransformation nicht möglich.

Die folgende Grafik zeigt beispielhaft zulässige und für die Transformation ungültige Gelenkstellungen:

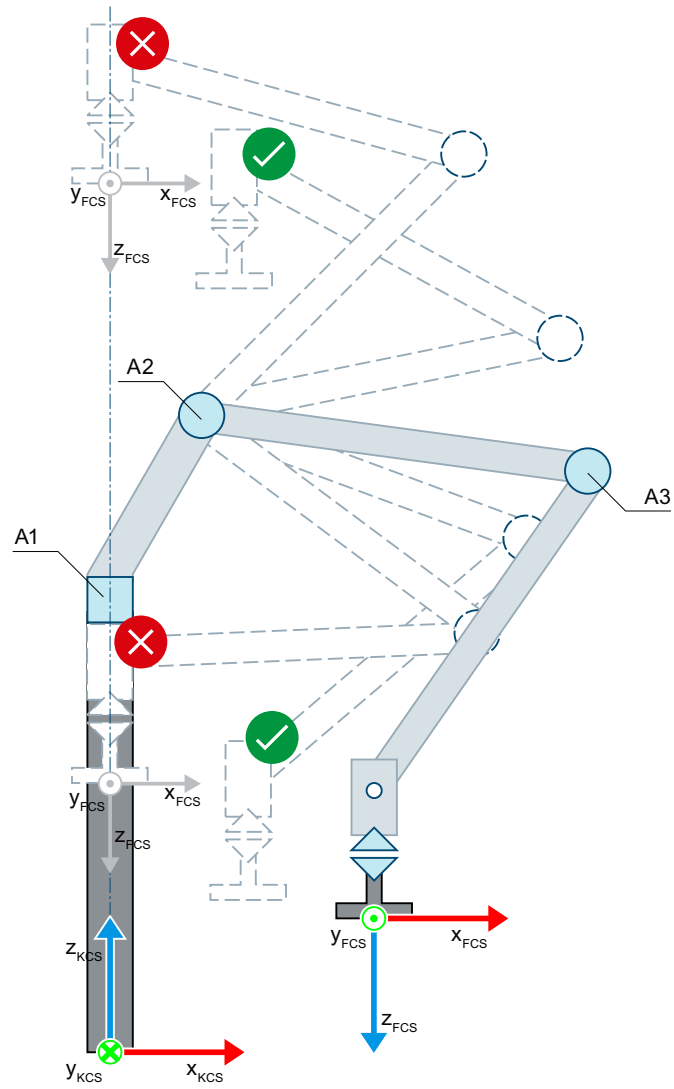

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation

Knickarm 3D mit Orientierung

Die Kinematik "Knickarm 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die Achsen sind als serielle Kinematik mit Zwangskopplung des Flanschsystems aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

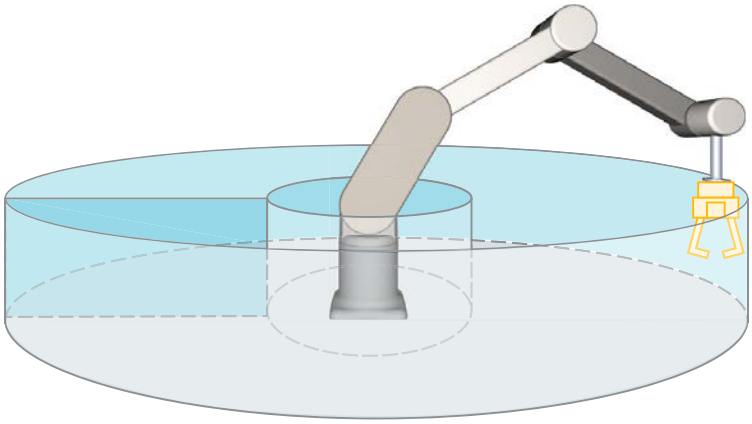

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit Drehung um die z-Achse des Kinematikkoordinatensystems (KCS)
- Eine rotatorische Achse A2 mit den Abständen L1 in z-Richtung des KCS und L2 in x-Richtung des KCS zum Kinematiknullpunkt
- Eine rotatorische Achse A3 im Abstand L3 zur Achse A2
- Eine rotatorische Achse A4 (Orientierungsachse) im Abstand L4 in x-Richtung des KCS zur Achse A3

Die Kinematik besteht aus einer Basis und Gelenkarmen, die durch Drehgelenke verbunden sind (Achsen A1, A2 und A3). Die Achse A1 dreht die Kinematik horizontal um die Basis. Die Achsen A2 und A3 bewegen die Gelenkarme vertikal. Die Kinematik ermöglicht einen annähernd kugelförmigen Arbeitsbereich. Durch eine Zwangskopplung zwischen der Achse A2 und dem Flanschsystem zeigt die z-Achse des FCS immer in die negative z-Richtung des KCS. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und des Zwangskoppelpunkts
- Die Lage Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

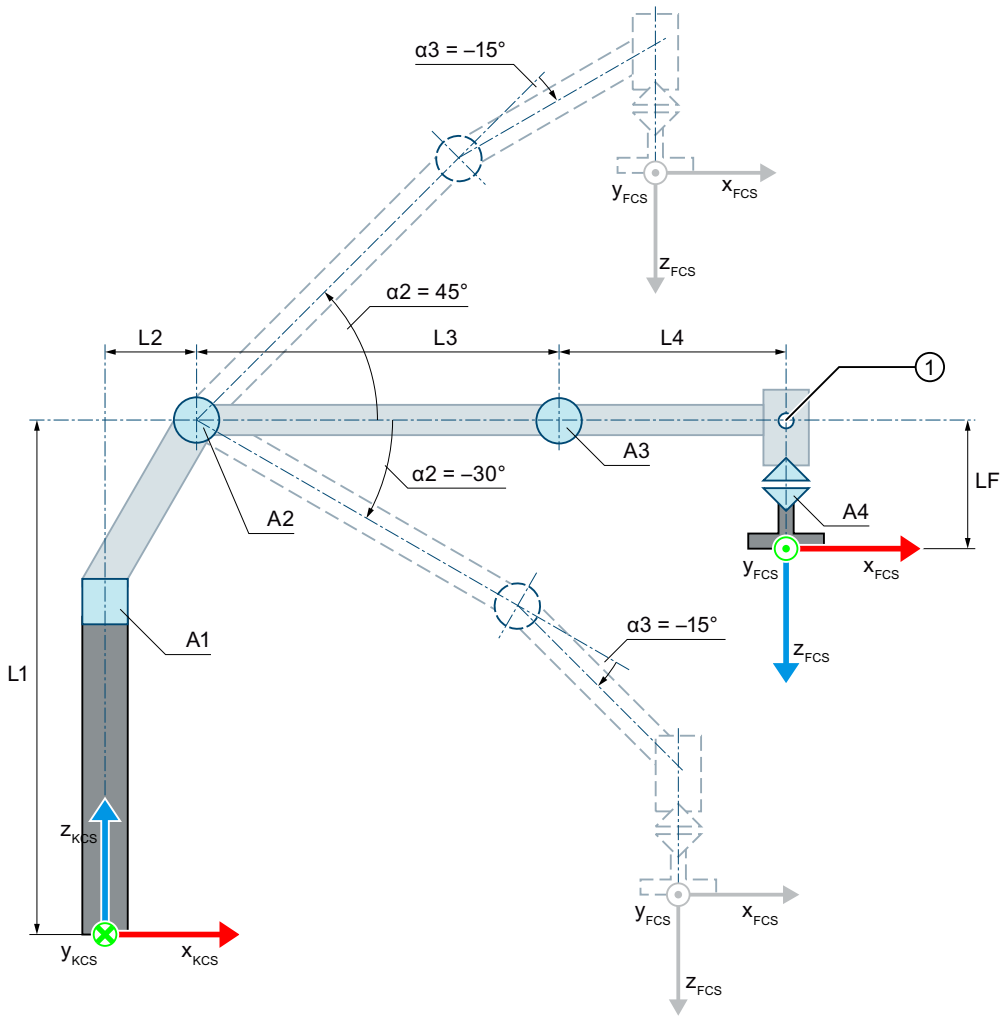

① Zwangskoppelpunkt

Nullstellung der Kinematik

- L1 Abstand der Achse A2 zum Kinematiknullpunkt (KNP) in z-Richtung des KCS
- L2 Abstand der Achse A2 zum KNP in x-Richtung des KCS
- L3 Abstand der Achse A3 zu der Achse A2 in x-Richtung des KCS
- L4 Abstand des Zwangskoppelpunkts zu der Achse A3 in x-Richtung des KCS
- LF Abstand des FCS zum Zwangskoppelpunkt in z-Richtung des FCS

Auslenkung der Kinematik

- α2 Positive Auslenkung der Achse A2 bei α2 = 45.0°
  Negative Auslenkung der Achse A2 bei α2 = -30.0°
- α3 Negative Auslenkung der Achse A3 bei α3 = -15.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen
- Die Lage Koordinatensysteme KCS und FCS

- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

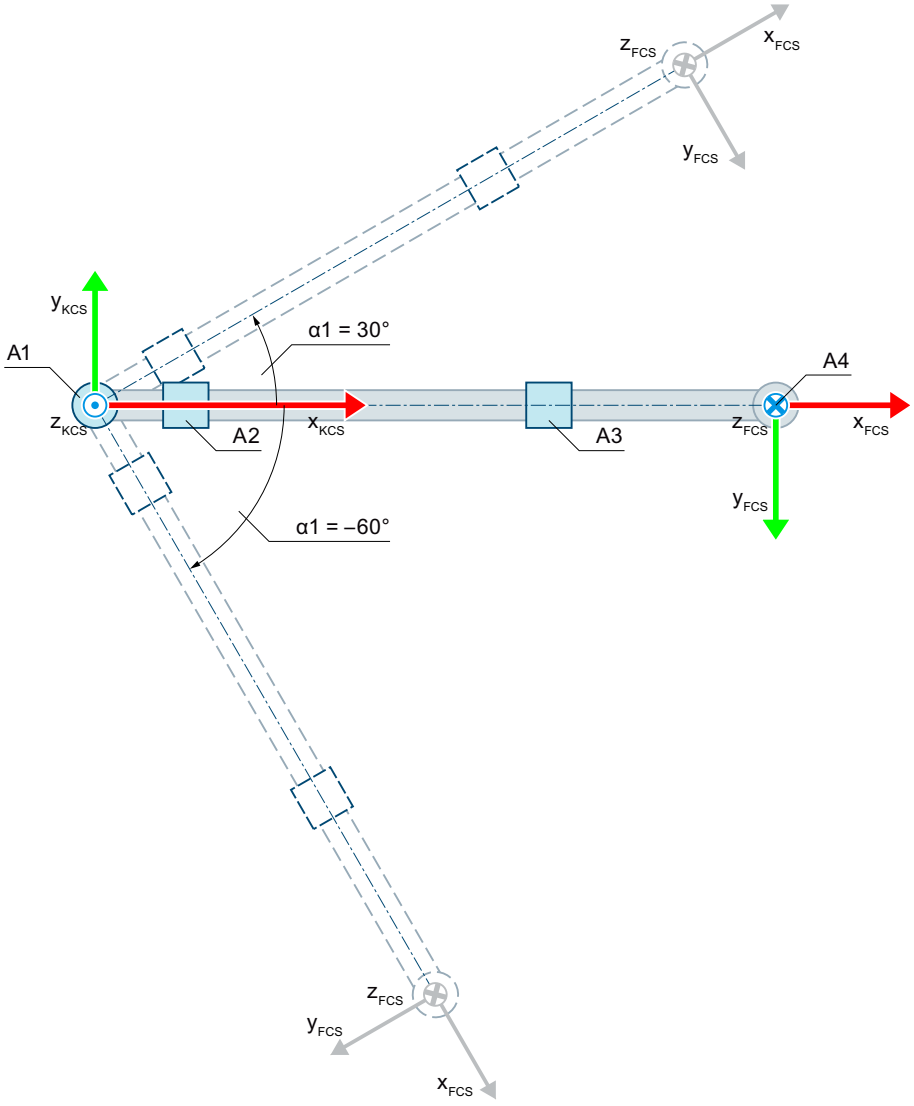

Nullstellung der Kinematik

Auslenkung der Kinematik

α1 Positive Auslenkung der Achse A1 bei α1 = 30.0°

Negative Auslenkung der Achse A1 bei α1 = -60.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Längen L1 und L2 definieren Sie die Lage der Achse A2 gegenüber dem KNP. Die Achse A3 liegt im Abstand L3 in x-Richtung des KCS zur Achse A2.

Das Flanschkoordinatensystem (FCS) liegt in folgenden Abständen zur Achse A3 und zum Zwangskoppelpunkt:

- Abstand L4 zur Achse A3 in x-Richtung des KCS
- Abstand LF zum Zwangskoppelpunkt in negativer z-Richtung des KCS

Die Achse A3 und das Flanschsystem sind zwangsgekoppelt. Mit der Zwangskopplung zeigt die z-Achse des FCS immer in negative z-Richtung des KCS. Der Zwangskoppelpunkt liegt im Abstand L4 in x-Richtung des KCS zur Achse A3.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Die Gelenkarme der Kinematik zeigen in x-Richtung des KCS.
A2	Bei Nullstellung von Achse A1 zeigt die Länge L3 in x-Richtung des KCS.
A3	Bei Nullstellung der Achsen A1 und A2 zeigt die Länge L4 in x-Richtung KCS.
A4	Bei Nullstellung der Achsen A1, A2 und A3 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie eine mechanische Achskopplung von Achse A2 auf Achse A3 konfigurieren. Die Kinematiktransformation kompensiert die konfigurierte mechanische Achskopplung. Bei einem Kopplungsfaktor > 0.0 geht die Kinematiktransformation davon aus, dass eine positive Bewegung der Achse A2 zu einer negativen Bewegung an der Achse A3 führt.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: -180.0° ≤ α1 < 180.0°
- Achse A2: -180.0° ≤ α2 < 180.0°
- Achse A3: -180.0° ≤ α3 < 180.0°
- Achse A4: Keine Begrenzung
  Für die Orientierung kann ein Winkel größer als 360° vorgegeben werden. Aber die Koordinate A des Werkzeugarbeitspunkts (TCP) wird im Bereich -180° bis +180° abgebildet.

Hinweis

Singuläre Stellungen

Die Kinematik besitzt singuläre Stellungen (Seite 2318).

Eine singuläre Stellung liegt vor, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. In diesem Bereich ist die Rückwärtstransformation nicht möglich.

Die folgende Grafik zeigt beispielhaft zulässige und für die Transformation ungültige Gelenkstellungen:

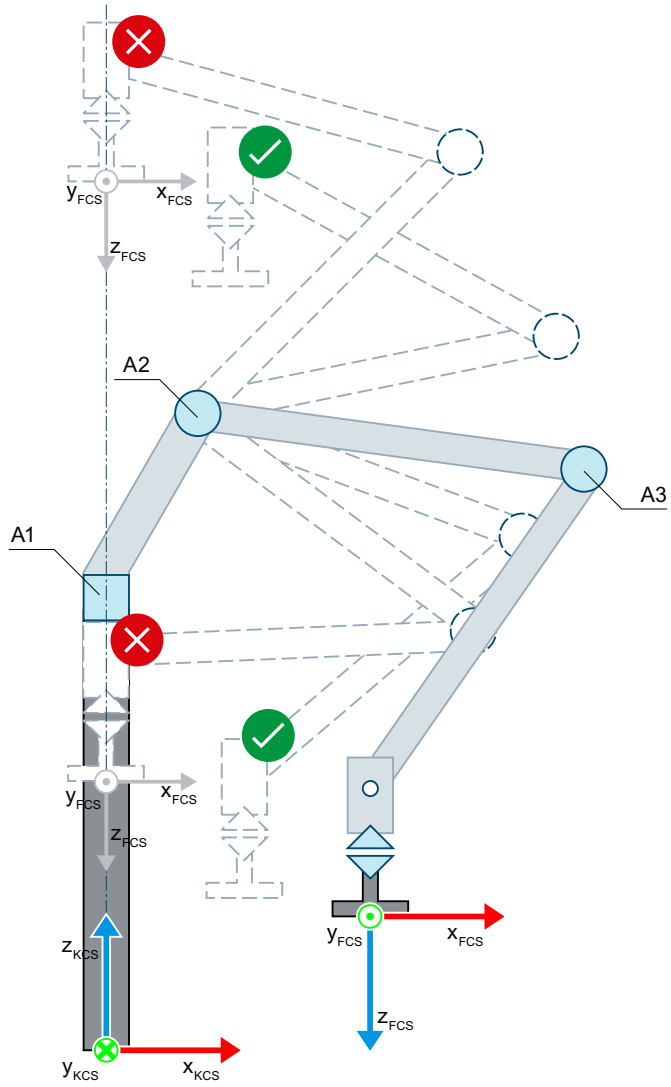

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation

Variablen Knickarm

Knickarm 2D

Die 2D Knickarm-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	11	Knickarm 2D
	12	Knickarm 2D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Achse A1 zum Kinematiknullpunkt in z-Richtung des Kinematikkoordinatensystems (KCS)

Variablen	Werte	Beschreibung
<TO>.Kinematics.Parameter[2]	-1.0E12 bis 1.0E12	Abstand L2 der Achse A1 zum Kinematiknullpunkt in x-Richtung des KCS
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Schenkellänge L3 zwischen den Achsen A1 und A2
<TO>.Kinematics.Parameter[4]	-	Mechanische Achskopplung von Achse A1 auf Achse A2 vorhanden/nicht vorhanden
	0	Nicht vorhanden
	1	Vorhanden
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Mechanischer Achskopplungsfaktor von Achse A1 auf Achse A2
<TO>.Kinematics.Parameter[6]	0.001 bis 1.0E12	Schenkellänge L4 zwischen Achse A2 und Zwangskoppelpunkt
<TO>.Kinematics.Parameter[7]	-1.0E12 bis 1.0E12	Abstand LF des Flanschkoordinatensystems (FCS) vom Zwangskoppelpunkt in negativer z-Richtung des KCS

Knickarm 3D

Die 3D Knickarm-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	13	Knickarm 3D
	14	Knickarm 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Achse A2 zum Kinematiknullpunkt in z-Richtung des KCS
<TO>.Kinematics.Parameter[2]	0.0 bis 1.0E12	Abstand L2 der Achse A2 zum Kinematiknullpunkt in x-Richtung des KCS
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Schenkellänge L3 zwischen den Achsen A2 und A3
<TO>.Kinematics.Parameter[4]	-	Mechanische Achskopplung von Achse A2 auf Achse A3 vorhanden/nicht vorhanden
	0	Nicht vorhanden
	1	Vorhanden
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Mechanischer Achskopplungsfaktor von Achse A2 auf Achse A3
<TO>.Kinematics.Parameter[6]	0.001 bis 1.0E12	Schenkellänge L4 zwischen Achse A3 und Zwangskoppelpunkt
<TO>.Kinematics.Parameter[7]	-1.0E12 bis 1.0E12	Abstand LF des FCS vom Zwangskoppelpunkt in negativer z-Richtung des KCS

Delta-Picker

Delta-Picker 2D

Die Kinematik "Delta-Picker 2D" unterstützt zwei Achsen und zwei Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsraum der Kinematik:

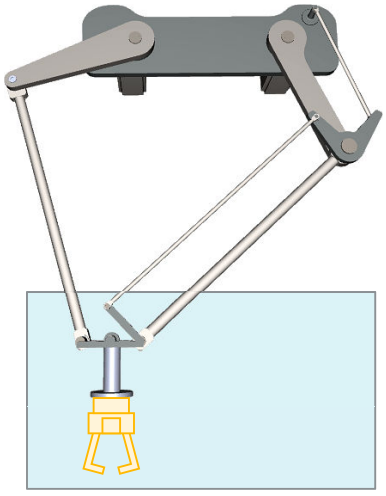

Die Kinematik besteht aus zwei rotatorischen Achsen A1 und A2.

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, zwei Oberarmen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme (Achsen A1, A2) sind an der oberen Verbindungsplatte befestigt. Die Oberarme und Verbindungsstangen verbinden die obere und die untere Verbindungsplatte. Das Werkzeug hängt an der unteren Verbindungsplatte. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

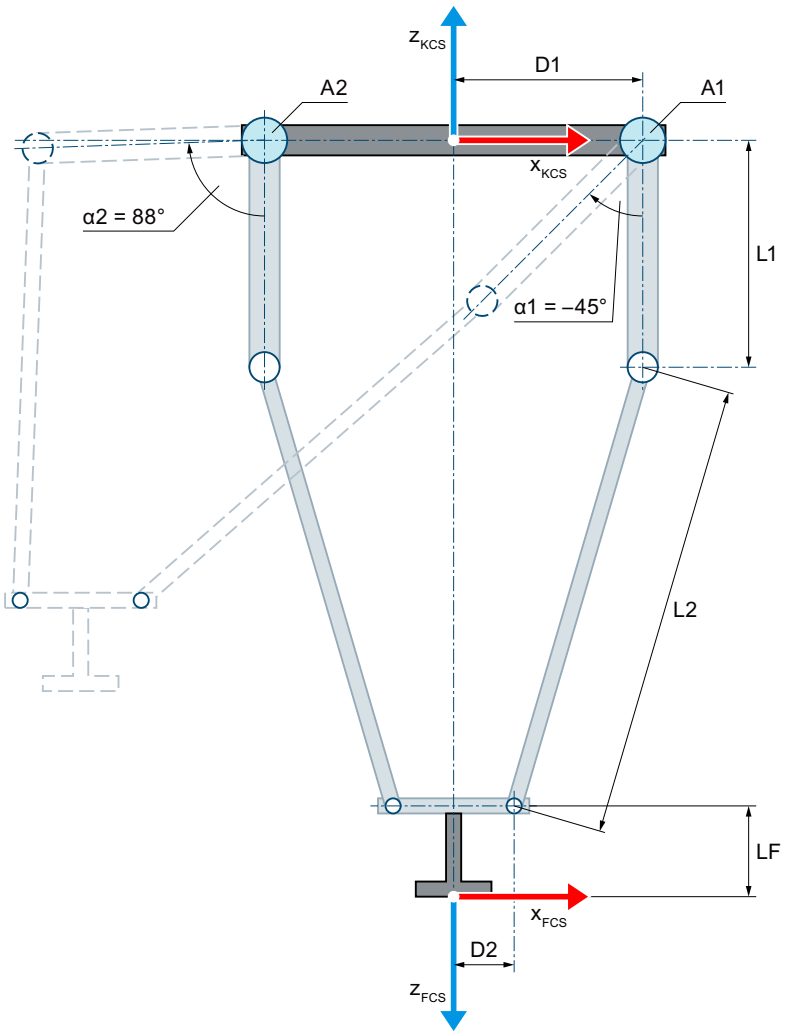

Nullstellung der Kinematik

D1 Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte)

D2 Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte)

L1 Länge der Oberarme

L2 Länge der Verbindungsstangen

D1, D2, L1 und L2 sind bei beiden Armen der Kinematik identisch.

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

Die Bewegung der Achsen in positive Richtung ist die Drehung der Oberarme nach außen.

α1 Auslenkung der Achse A1 in negativer Richtung bei α1 = -45.0°

α2 Auslenkung der Achse A2 in positiver Richtung bei α2 = 88.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Die Achsen A1 und A2 sind vom gemeinsamen Mittelpunkt (Kinematiknullpunkt) jeweils um den Abstand D1 entfernt.

Das Flanschkoordinatensystem (FCS) liegt im Mittelpunkt an der Unterseite der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Arme. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung der Achsen A1 und A2 zeigen die Oberarme in die negative z-Richtung des KCS.

Transformationsbereich

Für die Arme der Kinematik ist nur die nach außen geknickte Gelenkstellung (Seite 2297) zulässig. Sie können die Achsen nicht über die Strecklage der Arme hinaus verfahren.

Delta-Picker 2D mit Orientierung

Die Kinematik "Delta-Picker 2D mit Orientierung" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsraum der Kinematik:

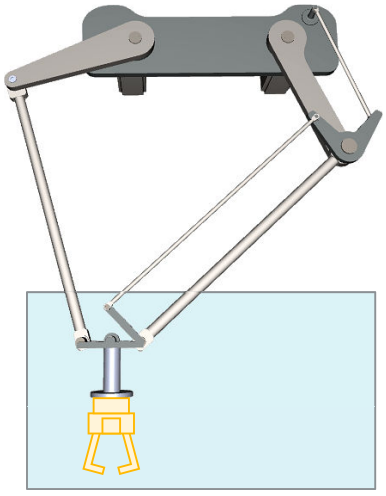

Die Kinematik besteht aus folgenden Achsen:

- Zwei rotatorische Achsen A1, A2
- Eine rotatorische Achse A4 (Orientierungsachse)

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, zwei Oberarmen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme (Achsen A1, A2) sind an der oberen Verbindungsplatte befestigt. Die Oberarme und Verbindungsstangen verbinden die obere und die untere Verbindungsplatte. Das Werkzeug hängt an der unteren Verbindungsplatte. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Vorderansicht:

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellungen der Achsen A1 und A4
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

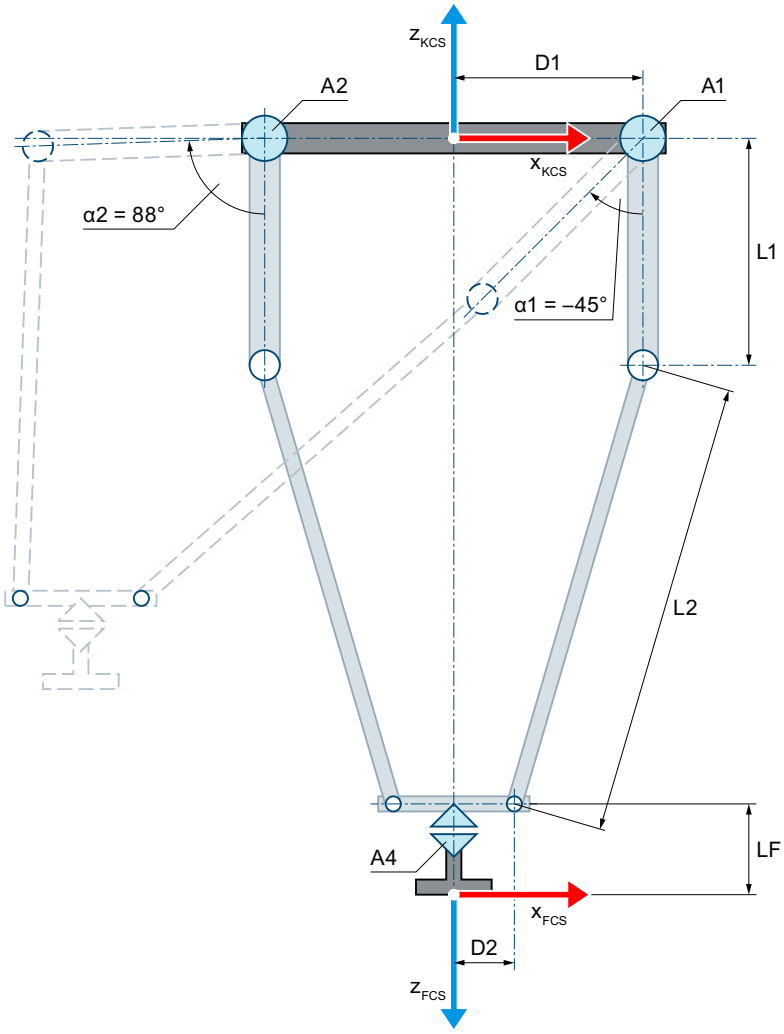

Nullstellung der Kinematik

D1 Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte)

D2 Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte)

L1 Länge der Oberarme

L2 Länge der Verbindungsstangen

D1, D2, L1 und L2 sind bei beiden Armen der Kinematik identisch.

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

Die Bewegung der Achsen in positive Richtung ist die Drehung der Oberarme nach außen.

α1 Auslenkung der Achse A1 in negativer Richtung bei α1 = -45.0°

α2 Auslenkung der Achse A2 in positiver Richtung bei α2 = 88.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das Kinematikkoordinatensystem (KCS) mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Die Achsen A1 und A2 sind vom gemeinsamen Mittelpunkt (Kinematiknullpunkt) jeweils um den Abstand D1 entfernt.

Das Flanschkoordinatensystem (FCS) liegt im Mittelpunkt an der Unterseite der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Arme. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung der Achsen A1 und A2 zeigen die Oberarme in die negative z-Richtung des KCS. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Für die Arme der Kinematik ist nur die nach außen geknickte Gelenkstellung (Seite 2297) zulässig. Sie können die Achsen nicht über die Strecklage der Arme hinaus verfahren.

Delta-Picker 3D

Die Kinematik "Delta-Picker 3D" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsraum der Kinematik:

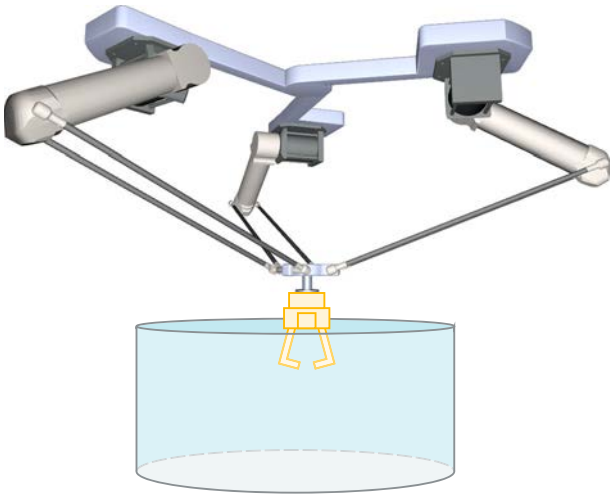

Die Kinematik besteht aus drei rotatorischen Achsen A1, A2 und A3.

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, drei Oberarmen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme (Achsen A1, A2 und A3) sind an der oberen Verbindungsplatte befestigt. Die Oberarme und Verbindungsstangen verbinden die obere und die untere Verbindungsplatte. Das Werkzeug hängt an der unteren Verbindungsplatte. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage des Kinematikkoordinatensystems (KCS)
- Die Winkel der Achsen A1, A2 und A3 zueinander

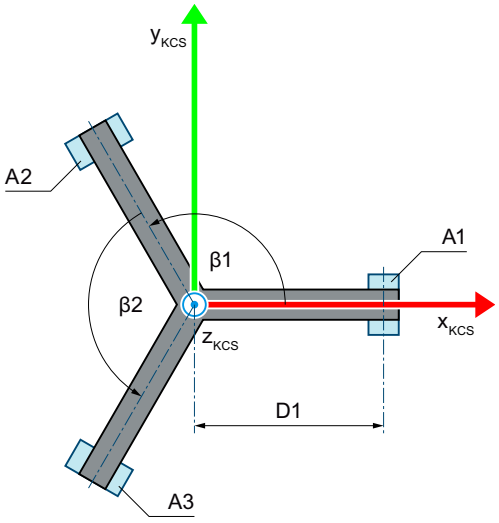

β1 Winkel zwischen den Achsen A1 und A2

β2 Winkel zwischen den Achsen A2 und A3

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht die Lage des Flanschkoordinatensystems (FCS) in der xy-Ebene der unteren Verbindungsplatte:

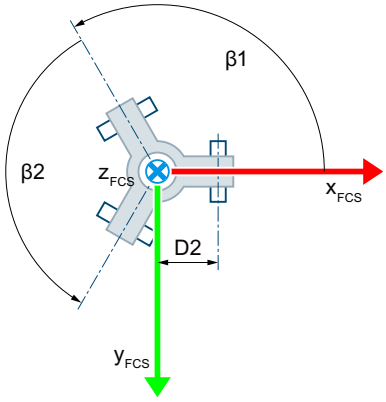

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achse A1 und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Achse A1
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achse A1

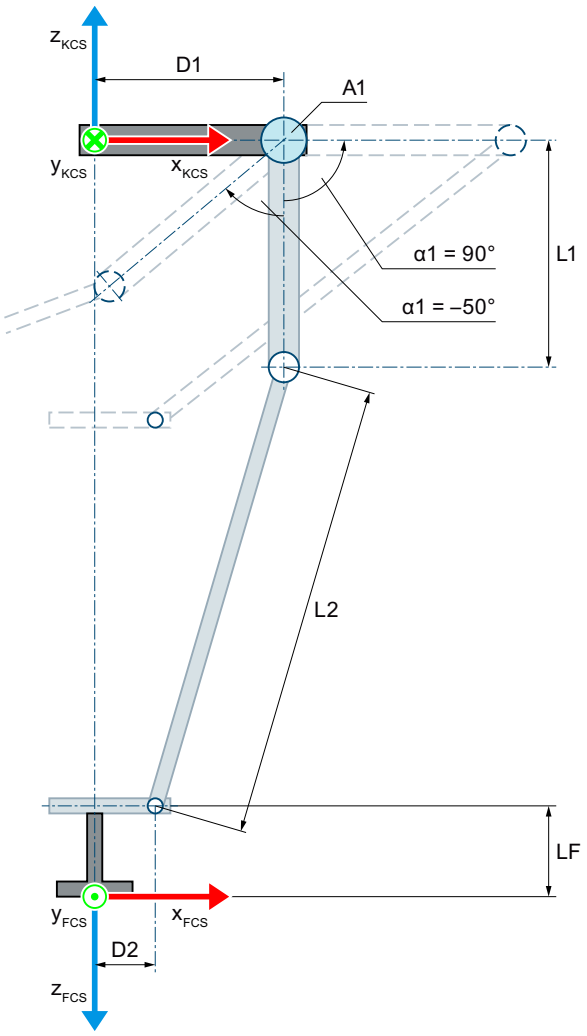

Nullstellung der Kinematik

D1 Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte)

D2 Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte)

L1 Länge der Oberarme

L2 Länge der Verbindungsstangen

D1, D2, L1 und L2 sind bei den drei Armen der Kinematik identisch.

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

Die Bewegung der Achsen in positive Richtung ist die Drehung der Oberarme nach außen.

α1 Auslenkung der Achse A1 in negativer Richtung bei α1 = -50.0°

Auslenkung der Achse A1 in positiver Richtung bei α1 = 90.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Die Achsen A1, A2 und A3 sind vom gemeinsamen Mittelpunkt (Kinematiknullpunkt) jeweils um den Abstand D1 entfernt.

Das FCS liegt im Mittelpunkt an der Unterseite der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Arme. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung der Achsen A1, A2 und A3 zeigen die Oberarme in die negative z-Richtung des KCS.

Transformationsbereich

Für die Arme der Kinematik ist nur die nach außen geknickte Gelenkstellung (Seite 2297) zulässig. Sie können die Achsen nicht über die Strecklage der Arme hinaus verfahren.

Delta-Picker 3D mit Orientierung

Die Kinematik "Delta-Picker 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsraum der Kinematik:

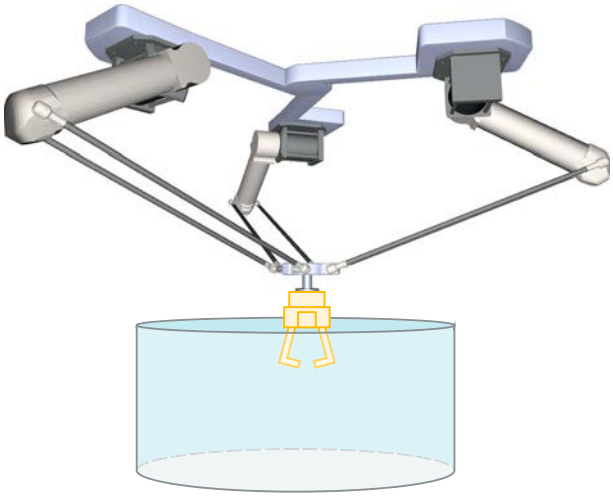

Die Kinematik besteht aus folgenden Achsen:

- Drei rotatorische Achsen A1, A2 und A3
- Eine rotatorische Achse A4 (Orientierungsachse)

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, drei Oberarmen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme (Achsen A1, A2 und A3) sind an der oberen Verbindungsplatte befestigt. Die Oberarme und Verbindungsstangen verbinden die obere und die untere Verbindungsplatte. Das Werkzeug ist an der unteren Verbindungsplatte befestigt. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS. Die Orientierungsachse A4 ermöglicht die Drehung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage des Kinematikkoordinatensystems (KCS)
- Die Winkel der Achsen A1, A2 und A3 zueinander

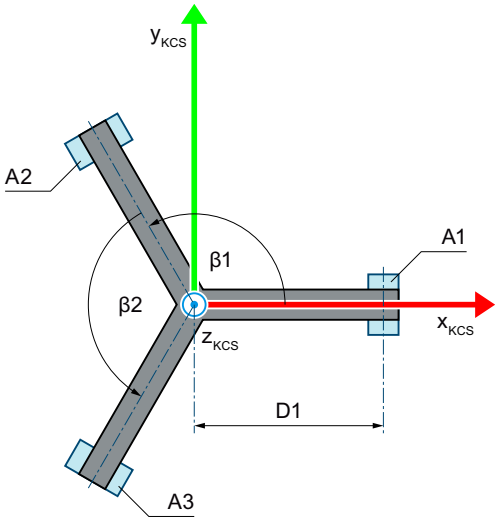

β1 Winkel zwischen den Achsen A1 und A2

β2 Winkel zwischen den Achsen A2 und A3

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht die Lage des Flanschkoordinatensystems (FCS) in der xy-Ebene der unteren Verbindungsplatte:

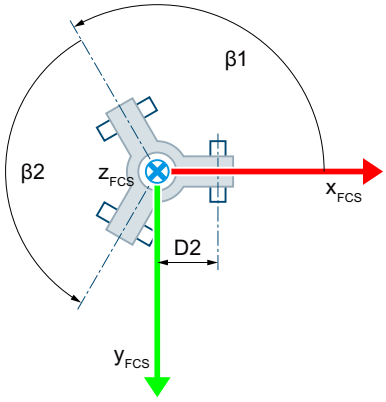

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achse A1 und der Koordinatensysteme KCS und FCS
- Die Nullstellungen der Achsen A1 und A4
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Achse A1

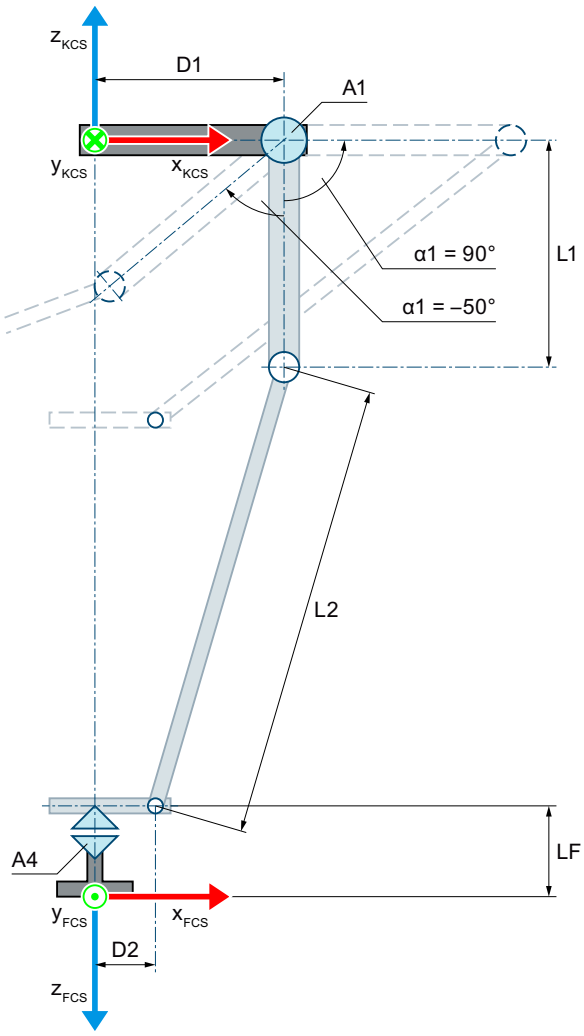

Nullstellung der Kinematik

D1 Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte)

D2 Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte)

L1 Länge der Oberarme

L2 Länge der Verbindungsstangen

D1, D2, L1 und L2 sind bei den drei Armen der Kinematik identisch.

LF Flanschlänge vor dem FCS in z-Richtung des FCS

Auslenkung der Kinematik

Die Bewegung der Achsen in positive Richtung ist die Drehung der Oberarme nach außen.

α1 Auslenkung der Achse A1 in negativer Richtung bei α1 = -50.0°

Auslenkung der Achse A1 in positiver Richtung bei α1 = 90.0°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Die Achsen A1, A2 und A3 sind vom gemeinsamen Mittelpunkt (Kinematiknullpunkt) jeweils um den Abstand D1 entfernt.

Das FCS liegt im Mittelpunkt an der Unterseite der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Arme. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung der Achsen A1, A2 und A3 zeigen die Oberarme in die negative z-Richtung des KCS. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Für die Arme der Kinematik ist nur die nach außen geknickte Gelenkstellung (Seite 2297) zulässig. Sie können die Achsen nicht über die Strecklage der Arme hinaus verfahren.

Zulässige Gelenkstellung für Delta-Picker

Für die Arme der Delta-Picker-Kinematiken ist nur die nach außen geknickte Gelenkstellung zulässig. Die folgende Grafik zeigt beispielhaft zulässige und für die Transformation ungültige Gelenkstellungen:

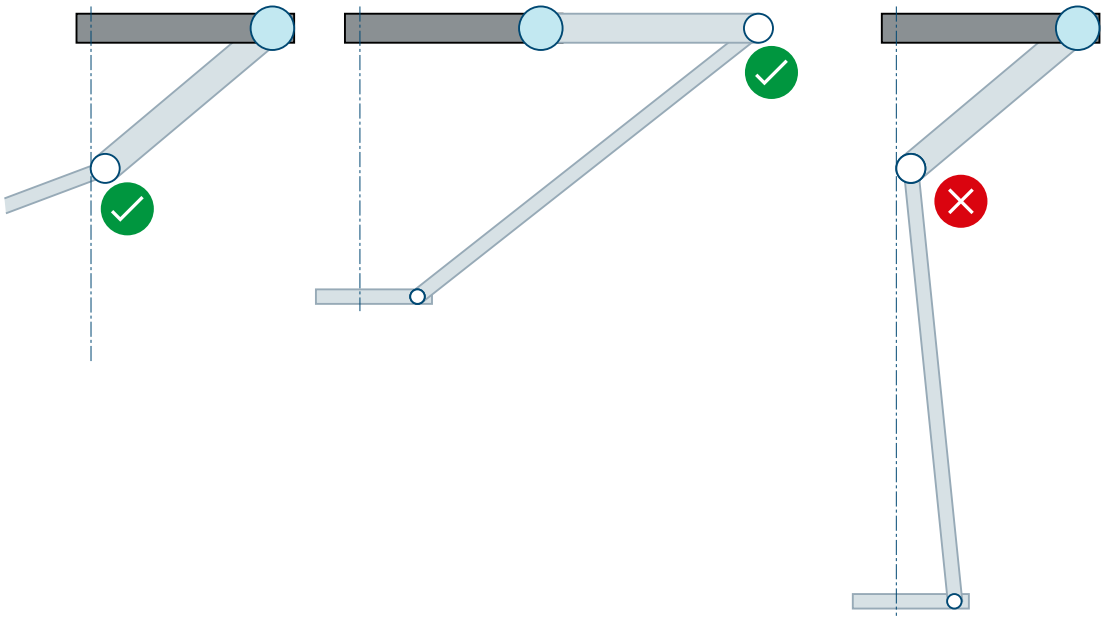

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation

Variablen Delta-Picker

Delta-Picker 2D

Die 2D Delta-Picker-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	15	Delta-Picker 2D
	16	Delta-Picker 2D mit Orientierung
<TO>.Kinematics.Parameter[1]	0.0 bis 1.0E12	Abstand D1 (Radius der oberen Verbindungsplatte)
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Länge L1 der Oberarme
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Länge L2 der Verbindungsstangen
<TO>.Kinematics.Parameter[4]	0.0 bis 1.0E12	Abstand D2 (Radius der unteren Verbindungsplatte)
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Abstand LF des FCS von der unteren Verbindungsplatte in negativer z-Richtung des KCS

Delta-Picker 3D

Die 3D Delta-Picker-Kinematiken definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	17	Delta-Picker 3D
	18	Delta-Picker 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	0.0 bis 1.0E12	Abstand D1 (Radius der oberen Verbindungsplatte)
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Länge L1 der Oberarme
<TO>.Kinematics.Parameter[3]	0.001 bis 1.0E12	Länge L2 der Verbindungsstangen
<TO>.Kinematics.Parameter[4]	0.0 bis 1.0E12	Abstand D2 (Radius der unteren Verbindungsplatte)
<TO>.Kinematics.Parameter[5]	90.001° bis 179.998°	Winkel β1 zwischen den Achsen A1 und A2
<TO>.Kinematics.Parameter[6]	90.001° bis 179.998°	Winkel β2 zwischen den Achsen A2 und A3
<TO>.Kinematics.Parameter[7]	-1.0E12 bis 1.0E12	Abstand LF des FCS von der unteren Verbindungsplatte in negativer z-Richtung des KCS

Zylindrischer Roboter

Zylindrischer Roboter 3D

Die Kinematik "Zylindrischer Roboter 3D" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als serielle Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

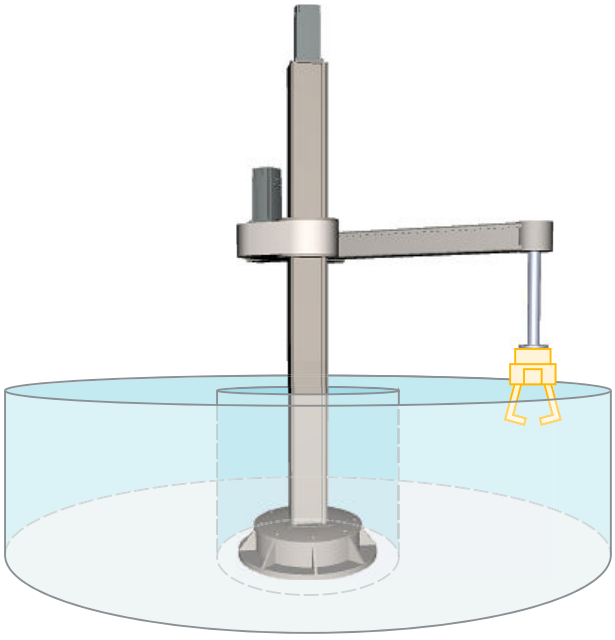

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit Drehung um die z-Achse des Kinematikkoordinatensystems (KCS)
- Eine lineare Achse A2 in z-Richtung des KCS
- Eine lineare Achse A3 in x-Richtung des KCS

Die Kinematik besteht aus einer Basis, einer Tragsäule und einem Ausleger. Die Achse A1 dreht die Tragsäule mit Ausleger horizontal um die Basis. Die Achse A2 bewegt den Ausleger vertikal. Die Achse A3 bewegt das Flanschsystem auf dem Ausleger horizontal. Die Kinematik ermöglicht einen zylindrischen Arbeitsbereich.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellungen der Achsen A1 und A2
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

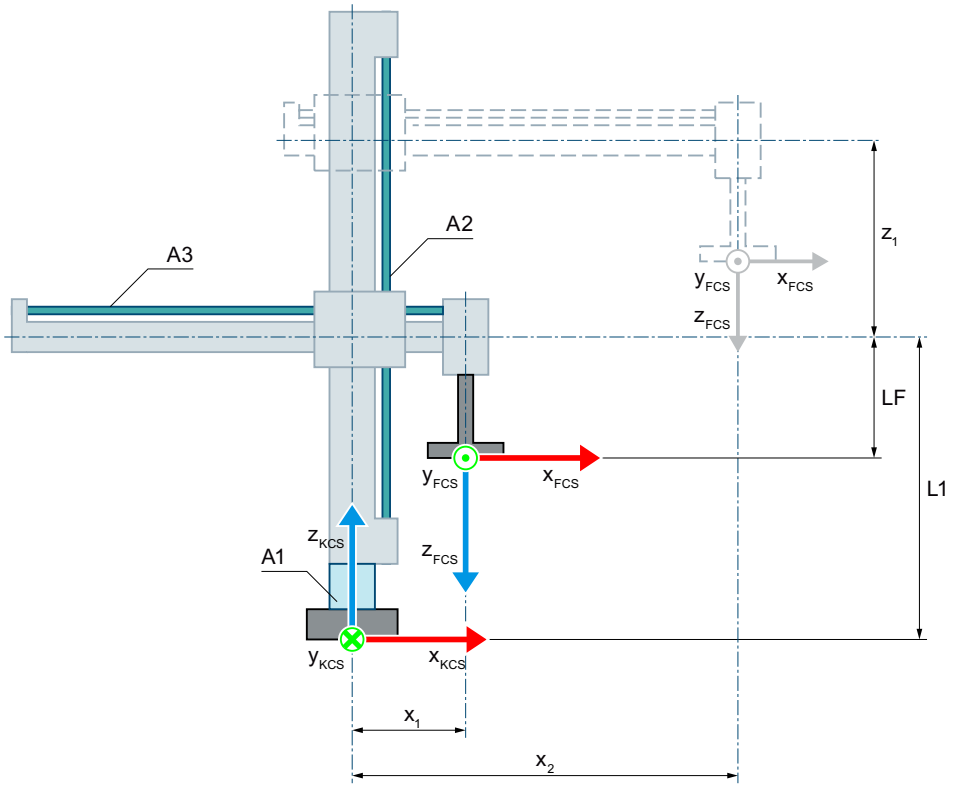

Nullstellung der Achsen A1 und A2

L1 Bei Nullstellung der Achse A2:
Abstand des FCS zum Kinematiknullpunkt (KNP) und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

x_1 Positive Auslenkung der Achse A3
In der Nullstellung der Achse A3 liegt die z-Achse des FCS auf der z-Achse des KCS. Die dargestellte Kinematik kann die Nullposition der Achse A3 mechanisch bedingt nicht anfahren.

Auslenkung der Kinematik

x_2 Positive Auslenkung der Achse A3

z_1 Positive Auslenkung der Achse A2

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

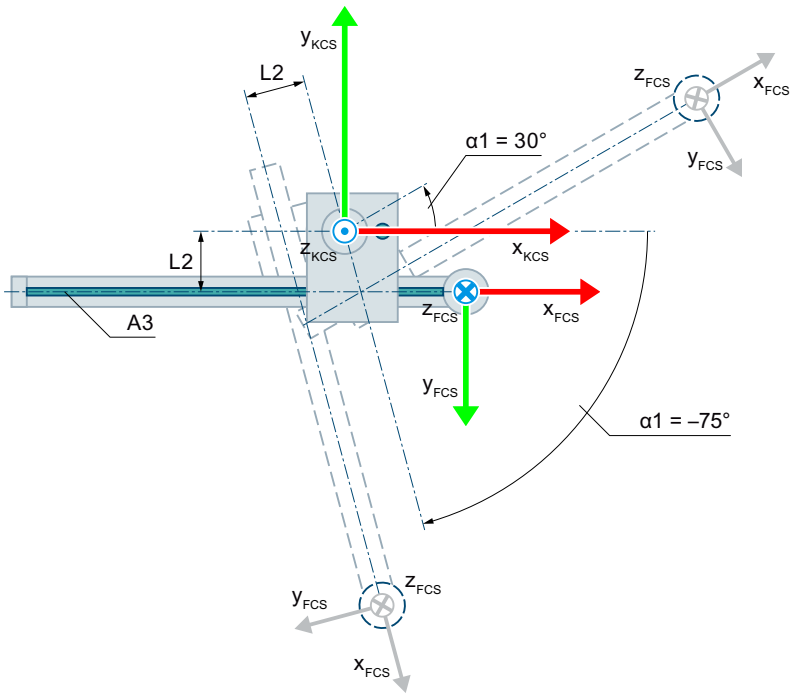

Nullstellung der Kinematik

L2 Abstand der Achse A3 vom KNP in y-Richtung des KCS (hier negativer Wert)

Auslenkung der Kinematik

α1 Positive Auslenkung der Achse A1 bei α1 = 30°

Negative Auslenkung der Achse A1 bei α1 = -75°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Länge L1 definieren Sie den Abstand der Nullposition der Achse A2 in z-Richtung des KCS vom KNP. Über die Länge L2 definieren Sie den Abstand der Achse A3 vom KNP in y-Richtung des KCS.

Das Flanschkoordinatensystem (FCS) liegt um die Länge LF in negativer z-Richtung des KCS verschoben auf der Achse A3.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Der Ausleger mit der Achse A3 zeigt in x_{KCS}-Richtung.
A2	Die Achse A2 steht auf der Position 0.0 des verschalteten Technologieobjekts.
A3	Die Achse A3 steht auf der Position 0.0 des verschalteten Technologieobjekts.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: -180.0° ≤ α1 < 180.0°
- Achse A2: Keine Begrenzung
- Achse A3: Keine Begrenzung

Hinweis

Singuläre Stellungen

Die Kinematik besitzt singuläre Stellungen (Seite 2318).

Eine singuläre Stellung liegt vor, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. In diesem Bereich ist die Rückwärtstransformation nicht möglich. Diese Position kann sich ergeben, wenn die Länge L2 bauartbedingt 0.0 ist.

Die folgende Grafik zeigt beispielhaft zulässige und für die Transformation ungültige Gelenkstellungen:

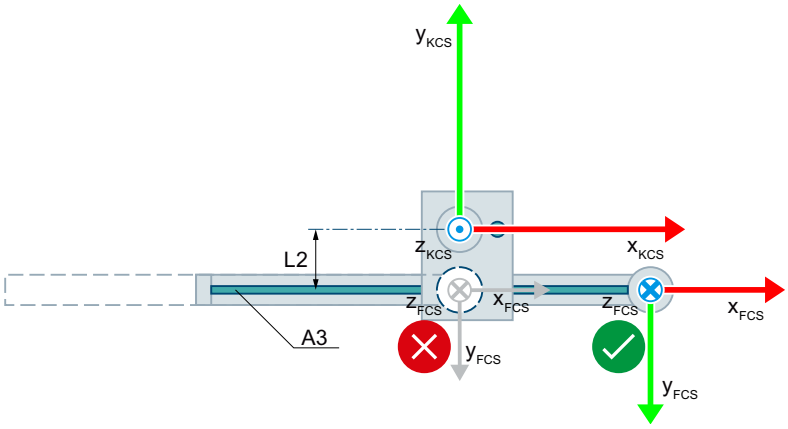

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation bei L2 = 0.0

Zylindrischer Roboter 3D mit Orientierung

Die Kinematik "Zylindrischer Roboter 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die Achsen sind als serielle Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

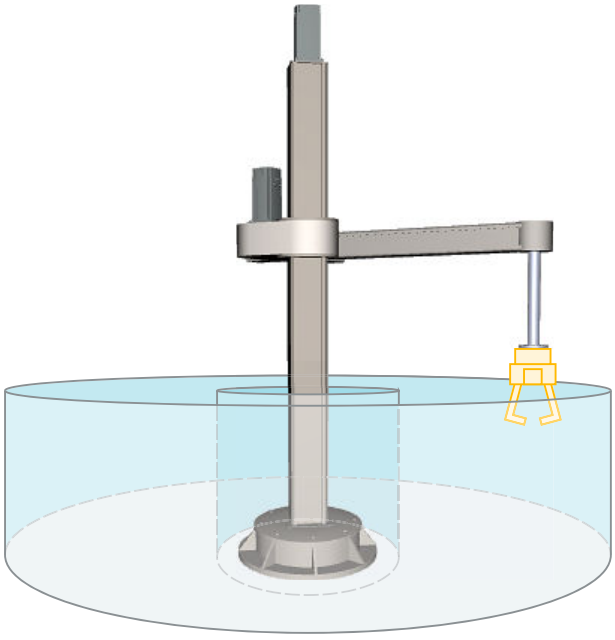

Die Kinematik besteht aus folgenden Achsen:

- Eine rotatorische Achse A1 mit Drehung um die z-Achse des Kinematikkoordinatensystems (KCS)
- Eine lineare Achse A2 in z-Richtung des KCS
- Eine lineare Achse A3 in x-Richtung des KCS
- Eine rotatorische Achse A4 (Orientierungsachse)

Die Kinematik besteht aus einer Basis, einer Tragsäule und einem Ausleger. Die Achse A1 dreht die Tragsäule mit Ausleger horizontal um die Basis. Die Achse A2 bewegt den Ausleger vertikal. Die Achse A3 bewegt das Flanschsystem auf dem Ausleger horizontal. Die Kinematik ermöglicht einen zylindrischen Arbeitsbereich. Die Orientierungsachse A4 ermöglicht die rotatorische Bewegung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Seitenansicht (xz-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellungen der Achsen A1 und A2
- Angedeutet (gestrichelt) die Auslenkung der Kinematik

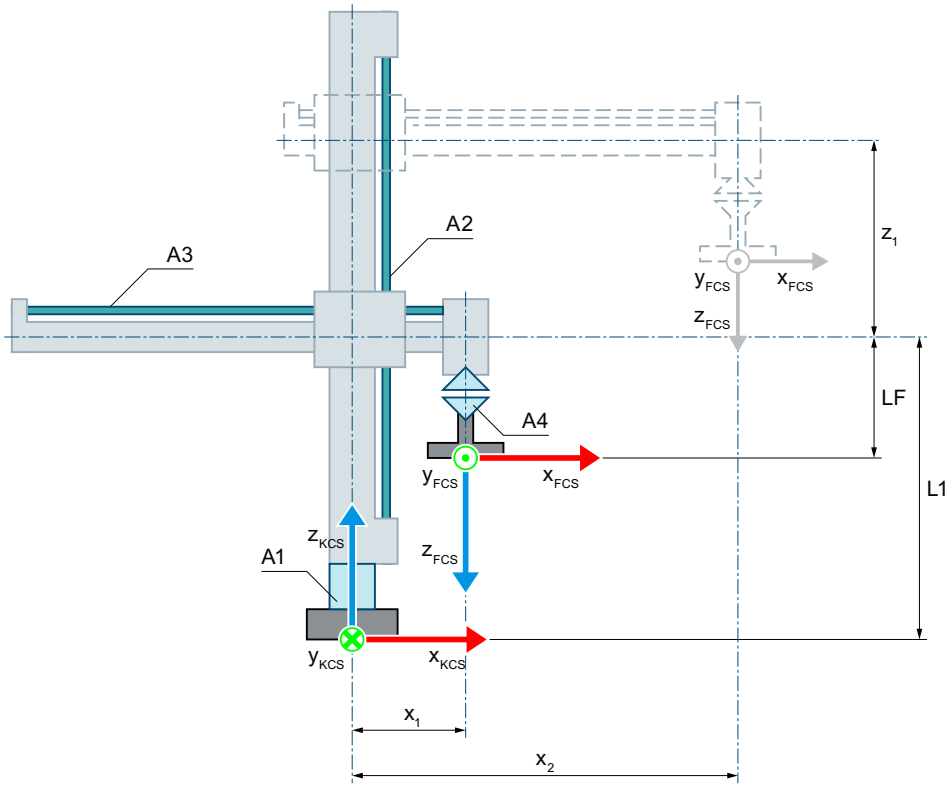

Nullstellung der Kinematik

L1 Bei Nullstellung der Achse A2:
Abstand des FCS zum Kinematiknullpunkt (KNP) und Flanschlänge LF in z-Richtung des KCS

LF Flanschlänge vor dem FCS in z-Richtung des FCS

x_1 Positive Auslenkung der Achse A3
In der Nullstellung der Achse A3 liegt die z-Achse des FCS auf der z-Achse des KCS. Die dargestellte Kinematik kann die Nullposition der Achse A3 mechanisch bedingt nicht anfahren.

Auslenkung der Kinematik

x_2 Positive Auslenkung der Achse A3

z_1 Positive Auslenkung der Achse A2

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage der Achsen und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Kinematik
- Angedeutet (gestrichelt) die positive/negative Auslenkung der Kinematik

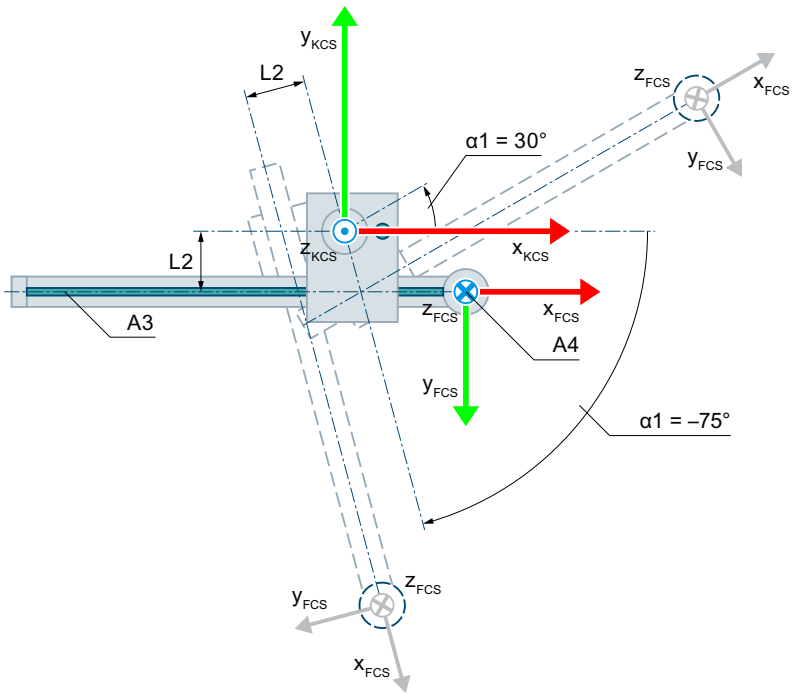

Nullstellung der Kinematik

L2 Abstand der Achse A3 zum KNP in y-Richtung des KCS (hier negativer Wert)

Auslenkung der Kinematik

α1 Positive Auslenkung der Achse A1 bei α1 = 30°

Negative Auslenkung der Achse A1 bei α1 = -75°

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt in der Basis der Kinematik. Über die Länge L1 definieren Sie den Abstand der Nullposition der Achse A2 in z-Richtung des KCS vom KNP. Über die Länge L2 definieren Sie den Abstand der Achse A3 vom KNP in y-Richtung des KCS.

Das Flanschkoordinatensystem (FCS) liegt um die Länge LF in negativer z-Richtung des KCS verschoben auf der Achse A3.

Folgende Tabelle zeigt die Nullstellung der Achsen:

Achse	Nullstellung
A1	Der Ausleger mit der Achse A3 zeigt in x_{KCS}-Richtung.
A2	Die Achse A2 steht auf der Position 0.0 des verschalteten Technologieobjekts.
A3	Die Achse A3 steht auf der Position 0.0 des verschalteten Technologieobjekts.
A4	Bei Nullstellung von Achse A1 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Kompensation von mechanischen Achskopplungen

Für die Kinematik können Sie eine mechanische Achskopplung von Achse A4 auf Achse A2 konfigurieren. Die Kinematiktransformation kompensiert die konfigurierte mechanische Achskopplung. Die Achskopplung zwischen der Achse A4 und der Achse A2 ist als Spindelsteigung realisiert. Bei einem Kopplungsfaktor von 1.0 entsprechen 360.0° an der Achse A4 einem Weg von -1.0 mm an der Achse A2.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achse A1: -180.0° ≤ α1 < 180.0°
- Achse A2: Keine Begrenzung
- Achse A3: Keine Begrenzung
- Achse A4: Keine Begrenzung
  Für die Orientierung kann ein Winkel größer als 360° vorgegeben werden. Aber die Koordinate A des Werkzeugarbeitspunkts (TCP) wird im Bereich -180° bis +180° abgebildet.

Hinweis

Singuläre Stellungen

Die Kinematik besitzt singuläre Stellungen (Seite 2318).

Eine singuläre Stellung liegt vor, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. In diesem Bereich ist die Rückwärtstransformation nicht möglich. Diese Position kann sich ergeben, wenn die Länge L2 bauartbedingt 0.0 ist.

Die folgende Grafik zeigt beispielhaft zulässige und für die Transformation ungültige Gelenkstellungen:

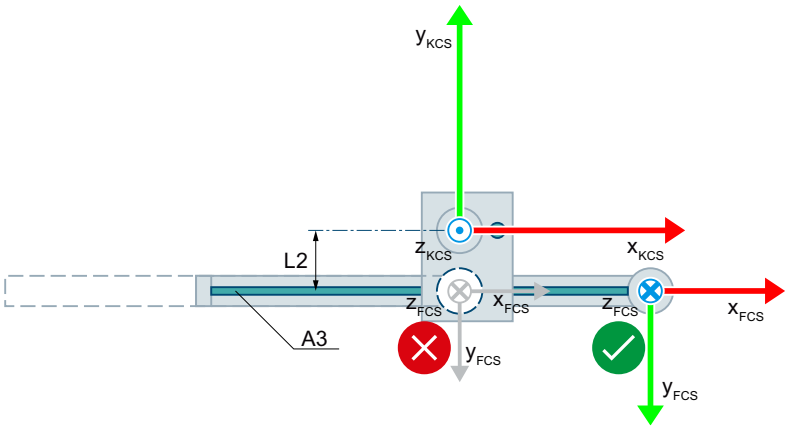

Zulässige Gelenkstellung

Ungültige Gelenkstellung für die Transformation bei L2 = 0.0

Variablen zylindrischer Roboter

Zylindrischer Roboter 3D

Die Kinematik "Zylindrischer Roboter 3D" definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	21	Zylindrischer Roboter 3D
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Nullposition der Achse A2 zum Kinematik-nullpunkt in z-Richtung des Kinematikkoordinatensystems (KCS)
<TO>.Kinematics.Parameter[2]	-1.0E12 bis 1.0E12	Abstand L2 zwischen den Achsen A2 und A3 in y-Richtung des KCS
<TO>.Kinematics.Parameter[3]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der Achse A3 in negativer z-Richtung des KCS

Zylindrischer Roboter 3D mit Orientierung

Die Kinematik "Zylindrischer Roboter 3D mit Orientierung" definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	22	Zylindrischer Roboter 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	-1.0E12 bis 1.0E12	Abstand L1 der Nullposition der Achse A2 zum Kinematik-nullpunkt in z-Richtung des KCS
<TO>.Kinematics.Parameter[2]	-1.0E12 bis 1.0E12	Abstand L2 zwischen den Achsen A2 und A3 in y-Richtung des KCS
<TO>.Kinematics.Parameter[3]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der Achse A3 in negativer z-Richtung des KCS
<TO>.Kinematics.Parameter[4]	-	Mechanische Achskopplung von Achse A4 auf Achse A2 vorhanden/nicht vorhanden
	0	Nicht vorhanden
	1	Vorhanden
<TO>.Kinematics.Parameter[5]	-1.0E12 bis 1.0E12	Mechanischer Achskopplungsfaktor von Achse A4 auf Achse A2

Tripod

Tripod 3D

Die Kinematik "Tripod 3D" unterstützt drei Achsen und drei Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

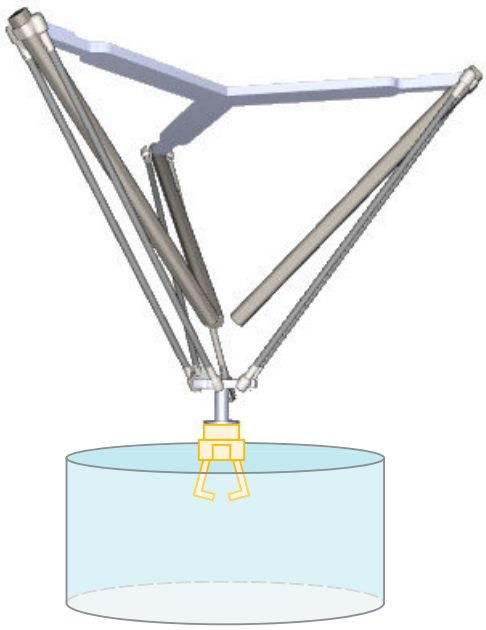

Die Kinematik besteht aus drei linearen Achsen A1, A2 und A3.

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, drei Armen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme bestehen aus Schienen mit Verfahrschlitten. Die Schienen mit den Verfahrschlitten sind an der oberen Verbindungsplatte befestigt. Verbindungsstangen verbinden die Verfahrschlitten mit der unteren Verbindungsplatte. Das Werkzeug hängt an der unteren Verbindungsplatte. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage des Kinematikkoordinatensystems (KCS)
- Die Winkel der Achsen A1, A2 und A3 zueinander

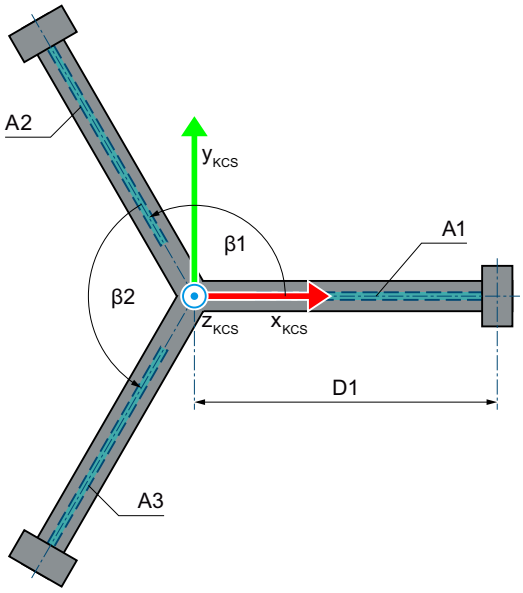

β1 Winkel zwischen den Achsen A1 und A2

β2 Winkel zwischen den Achsen A2 und A3

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht die Lage des Flanschkoordinatensystems (FCS) in der xy-Ebene der unteren Verbindungsplatte:

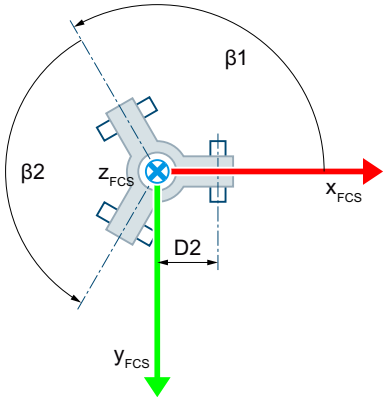

Die folgende Grafik zeigt in der Vorderansicht (xz-Ebene):

- Die Lage der Achse A1 und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Achse A1
- Angedeutet (gestrichelt) die positive Auslenkung der Achse A1

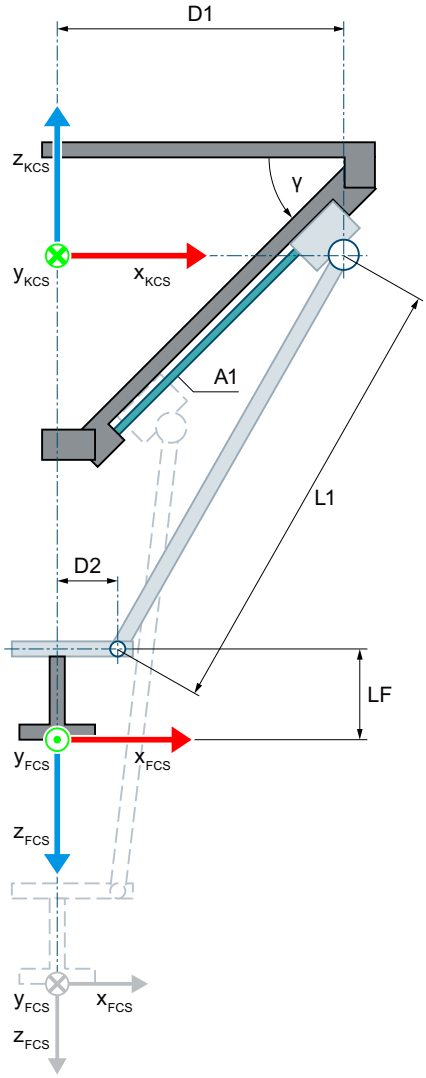

Nullstellung der Kinematik

D1	Abstand der oberen Gelenkpunkte der Verbindungsstangen zur Mitte der oberen Verbindungsplatte
D2	Abstand der unteren Gelenkpunkte der Verbindungsstangen zur Mitte der unteren Verbindungsplatte
L1	Länge der Verbindungsstangen
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
γ	Winkel zwischen der oberen Verbindungsplatte (xy-Ebene des KCS) und der Schiene der Achse A1 (0.0° ≤ γ < 90.0°)

D1, D2, L1 und γ sind bei den drei Armen der Kinematik identisch.

Auslenkung der Kinematik mit Auslenkung der Achse A1 in positiver Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Der Kinematiknullpunkt liegt mittig zu den Nullstellungen der Achsen A1, A2 und A3.

Das FCS liegt im Mittelpunkt der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Verbindungsstangen. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung stehen die Achsen A1, A2 und A3 in der x-y-Ebene des KCS.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achsen A1, A2 und A3: 0.0 ≤ Verfahrweg

Tripod 3D mit Orientierung

Die Kinematik "Tripod 3D mit Orientierung" unterstützt vier Achsen und vier Freiheitsgrade. Die Achsen sind als parallele Kinematik aufgebaut.

Die folgende Grafik zeigt den prinzipiellen Aufbau und den typischen Arbeitsbereich der Kinematik:

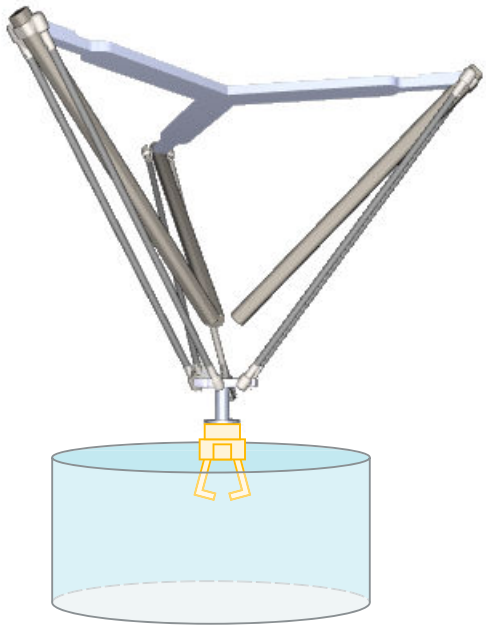

Die Kinematik besteht aus folgenden Achsen:

- Drei lineare Achsen A1, A2 und A3
- Eine rotatorische Achse A4 (Orientierungsachse)

Die Kinematik ist hängend modelliert und besteht aus einer oberen Verbindungsplatte, drei Armen und einer unteren Verbindungsplatte. Die Achsen zur Bewegung der Arme bestehen aus Schienen mit Verfahrschlitten. Die Schienen mit den Verfahrschlitten sind an der oberen Verbindungsplatte befestigt. Verbindungsstangen verbinden die Verfahrschlitten mit der unteren Verbindungsplatte. Das Werkzeug hängt an der unteren Verbindungsplatte. Die Parallelogrammstrukturen der Verbindungsstangen halten die untere Verbindungsplatte parallel zur xy-Ebene des KCS. Die Orientierungsachse A4 ermöglicht die rotatorische Bewegung des Werkzeugs.

Koordinatensysteme und Nullstellung

Die folgende Grafik zeigt in der Draufsicht (xy-Ebene):

- Die Lage des Kinematikkoordinatensystems (KCS)
- Die Winkel der Achsen A1, A2 und A3 zueinander

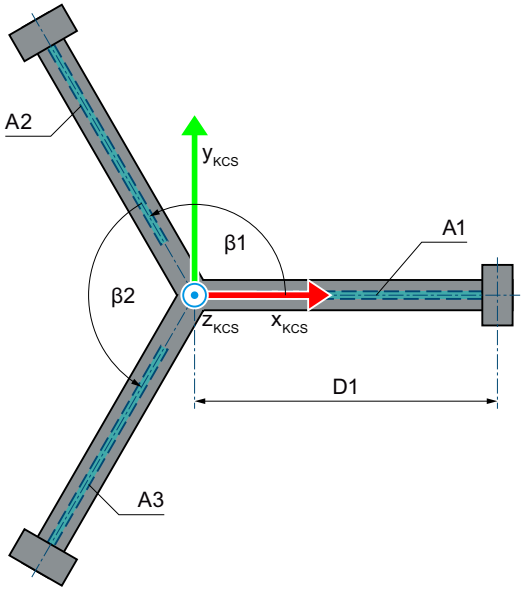

β1 Winkel zwischen den Achsen A1 und A2

β2 Winkel zwischen den Achsen A2 und A3

Legende zur Darstellung der Kinematiken (Seite 2238)

Die folgende Grafik zeigt in der Draufsicht die Lage des Flanschkoordinatensystems (FCS) in der xy-Ebene der unteren Verbindungsplatte:

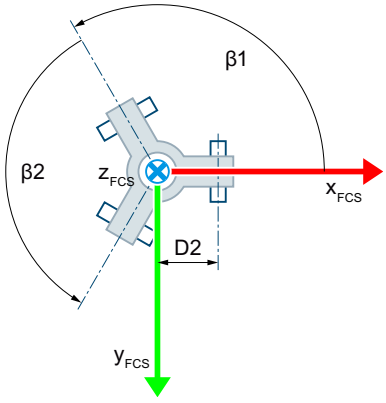

Die folgende Grafik zeigt in der Seitenansicht:

- Die Lage der Achse A1 und der Koordinatensysteme KCS und FCS
- Die Nullstellung der Achse A1
- Angedeutet (gestrichelt) die positive Auslenkung der Achse A1

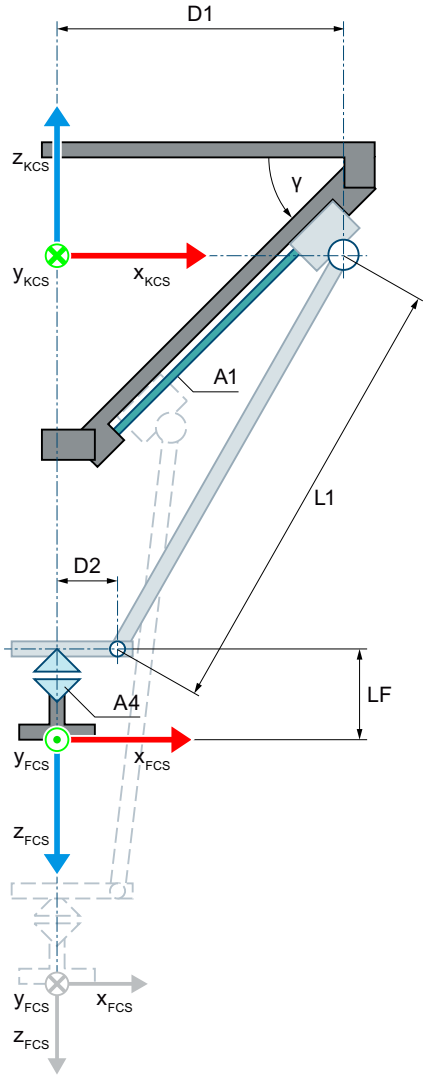

Nullstellung der Kinematik

D1	Abstand der oberen Gelenkpunkte der Verbindungsstangen zur Mitte der oberen Verbindungsplatte
D2	Abstand der unteren Gelenkpunkte der Verbindungsstangen zur Mitte der unteren Verbindungsplatte
L1	Länge der Verbindungsstangen
LF	Flanschlänge vor dem FCS in z-Richtung des FCS
γ	Winkel zwischen der oberen Verbindungsplatte (xy-Ebene des KCS) und der Schiene der Achse A1 (0.0° ≤ γ < 90.0°)

D1, D2, L1 und γ sind bei den drei Armen der Kinematik identisch.

Auslenkung der Kinematik mit Auslenkung der Achse A1 in positiver Richtung

Legende zur Darstellung der Kinematiken (Seite 2238)

Das KCS mit dem Kinematiknullpunkt (KNP) liegt im Mittelpunkt der oberen Verbindungsplatte. Der Kinematiknullpunkt liegt mittig zu den Nullstellungen der Achsen A1, A2 und A3.

Das FCS liegt im Mittelpunkt der unteren Verbindungsplatte mit jeweils gleichem Abstand D2 zu den Gelenkpunkten der Verbindungsstangen. Über die Länge LF verschieben Sie das FCS in negativer z-Richtung des KCS.

In der Nullstellung stehen die Achsen A1, A2 und A3 in der x-y-Ebene des KCS. In der Nullstellung der Achse A4 zeigt die x-Achse des FCS in die Richtung der x-Achse des KCS.

Transformationsbereich

Die Kinematiktransformation deckt folgenden Verfahrbereich (Seite 2316) der Achsen ab:

- Achsen A1, A2 und A3: 0.0 ≤ Verfahrweg
- Achse A4: Keine Begrenzung
  Für die Orientierung kann ein Winkel größer als 360° vorgegeben werden. Aber die Koordinate A des Werkzeugarbeitspunkts (TCP) wird im Bereich -180° bis +180° abgebildet.

Variablen Tripod

Die Tripod-Kinematik definieren Sie über folgende Variablen des Technologieobjekts:

Variablen	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	23	Tripod 3D
	24	Tripod 3D mit Orientierung
<TO>.Kinematics.Parameter[1]	0.0 bis 1.0E12	Abstand D1 (Radius der oberen Verbindungsplatte)
<TO>.Kinematics.Parameter[2]	0.001 bis 1.0E12	Länge L1 der Verbindungsstangen
<TO>.Kinematics.Parameter[3]	0.0° bis 89.999°	Winkel γ zwischen den linearen Achsen und der x-y-Ebene des Kinematikkoordinatensystems
<TO>.Kinematics.Parameter[4]	0.0 bis 1.0E12	Abstand D2 (Radius der unteren Verbindungsplatte)
<TO>.Kinematics.Parameter[5]	90.001° bis 179.998°	Winkel β1 zwischen den Achsen A1 und A2
<TO>.Kinematics.Parameter[6]	90.001° bis 179.998°	Winkel β2 zwischen den Achsen A2 und A3
<TO>.Kinematics.Parameter[7]	-1.0E12 bis 1.0E12	Abstand des Flanschkoordinatensystems von der unteren Verbindungsplatte in negativer z-Richtung des Kinematikkoordinatensystems

Anwenderdefinierte Kinematiken

Sie können folgende anwenderdefinierte Kinematiken mit entsprechenden Achsverschaltungen konfigurieren:

- Anwenderdefiniert 2D
- Anwenderdefiniert 2D mit Orientierung
- Anwenderdefiniert 3D
- Anwenderdefiniert 3D mit Orientierung

Die Konfiguration unterstützt Sie bei einer anwenderdefinierten Kinematik bei der Verschaltung der Positionierachsen. Zusätzlich stehen Ihnen systemseitig bis zu 32 Variablen für die Definition der Geometrie Ihrer Kinematik zur Verfügung.

Die Anwendertransformation (Seite 2320) der kartesischen Positionen sowie der Achsstellungen und Achsdynamik müssen Sie programmieren. Systemseitig stehen dafür vordefinierte Schnittstellen zur Verfügung.

Siehe auch

Anwendertransformation (Seite 2320)

Variablen anwenderdefinierte Kinematiken

Die anwenderdefinierten Kinematiken konfigurieren Sie über folgende Variablen des Technologieobjekts:

Variable	Werte	Beschreibung
<TO>.Kinematics.TypeOfKinematics	31	Anwenderdefiniert 2D
	32	Anwenderdefiniert 2D mit Orientierung
	33	Anwenderdefiniert 3D
	34	Anwenderdefiniert 3D mit Orientierung
<TO>.Kinematics.Parameter[1..32]	-	Bis zu 32 anwenderspezifische Parameter

Kinematiktransformation

Die Kinematiktransformation ist die Umrechnung zwischen den kartesischen Koordinaten der Kinematikbewegung und den Sollwerten für die einzelnen Kinematikachsen:

- Vorwärtstransformation
  Berechnung der kartesischen Koordinaten aus den Achsstellungen der Kinematikachsen
- Rückwärtstransformation
  Berechnung der Achsstellungen der Kinematikachsen aus den kartesischen Koordinaten

Die Kinematiktransformation rechnet die Positionswerte sowie die Dynamikwerte (Geschwindigkeit, Beschleunigung) um.

Das Technologieobjekt Kinematik stellt systemseitig die Kinematiktransformation für die vordefinierten Kinematiktypen bereit. Bei anwenderdefinierten Kinematiken müssen Sie die Anwendertransformation (Seite 2320) in einem eigenen Programm berechnen.

Transformation für vordefinierte Kinematiken

Bezugspunkte

Die Kinematiktransformation verwendet folgende Bezugspunkte:

- Kinematiknullpunkt (KNP)
- Nullstellungen der Kinematikachsen
- Werkzeugarbeitspunkt (TCP)

Die positive Richtung der Achsen für die Kinematiktransformation ist abhängig vom Kinematiktyp (Seite 2237). Konfigurieren Sie die positive Richtung am Technologieobjekt

Positionierachse/Gleichlaufachse entsprechend der positiven Richtung der Achse in der Kinematik.

Kinematiknullpunkt (KNP)

Der Koordinatenursprung des Kinematikkoordinatensystems (KCS) ist der KNP. Vom KNP ausgehend konfigurieren Sie die Geometrieparameter der Kinematik.

Nullstellungen der Kinematikachsen

Die Position 0.0 am Technologieobjekt Positionierachse/Gleichlaufachse definiert die Nullposition der Kinematikachse. Referenzieren Sie die Achsen so, dass die Achsen in der Kinematiknullstellung die Position 0.0 anzeigen. Die Nullstellung der Kinematik ist abhängig vom Kinematiktyp (Seite 2237).

Werkzeugarbeitspunkt (TCP)

Aus den Geometrieparametern und dem Tool-Frame ergibt sich die Lage des TCP.

Verfahrbereich und Transformationsbereich

Der Transformationsbereich ist der Bereich der Achsstellungen, den die Kinematiktransformation abdeckt. Der Kinematiktyp definiert den Transformationsbereich für die einzelnen Kinematikachsen. Die Angaben zum Transformationsbereich finden Sie bei der Beschreibung der einzelnen Kinematiken (Seite 2237).

Die Hardware- und Software-Endschalter einer Achse definieren den maximalen Verfahrbereich und den Arbeitsbereich. Der Arbeitsbereich einer Kinematikachse kann je nach Achskonfiguration größer oder kleiner sein als der Transformationsbereich:

- Arbeitsbereich > Transformationsbereich
  Wenn eine Kinematikachse während einer Kinematikbewegung den Transformationsbereich verlässt, gibt das Technologieobjekt Kinematik den Technologie-Alarm 803 aus. Die Kinematikbewegung wird abgebrochen und die Achsen halten mit den an den Achsen konfigurierten maximalen Dynamikwerten an (Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen).
- Arbeitsbereich ≤ Transformationsbereich
  Wenn eine Kinematikachse auf den Software-Endschalter fährt, gibt das Technologieobjekt Positionierachse/Gleichlaufachse den Technologie-Alarm 533 aus. Die Achse hält mit den an der Achse konfigurierten maximalen Dynamikwerten an (Alarmreaktion: Stopp mit maximalen Dynamikwerten). Mit dem Achsstopp gibt das Technologieobjekt Kinematik den Technologie-Alarm 801 aus. Die Kinematikbewegung wird abgebrochen und die Achsen halten mit den an den Achsen konfigurierten maximalen Dynamikwerten an (Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen).

Die folgende Grafik zeigt den Zusammenhang zwischen dem Arbeitsbereich der Achse und dem Transformationsbereich:

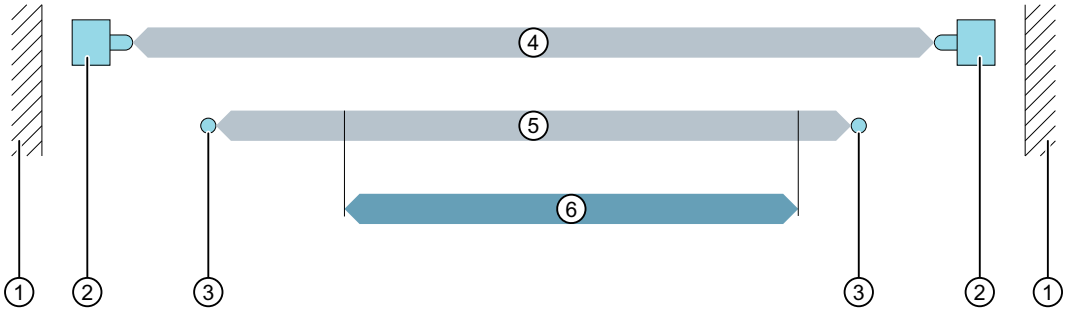

① Mechanischer Anschlag
② Hardware-Endschalter am Technologieobjekt Positionierachse/Gleichlaufachse
③ Software-Endschalter am Technologieobjekt Positionierachse/Gleichlaufachse
④ Maximaler Verfahrbereich der Achse
⑤ Arbeitsbereich der Achse
⑥ Transformationsbereich
(hier Arbeitsbereich > Transformationsbereich)

Gelenkstellungsräume (kinematikabhängig)

Abhängig vom Kinematiktyp kann eine Kinematik kartesische Koordinaten über verschiedene Gelenkstellungen erreichen. Der Kinematiktyp (Seite 2237) definiert die möglichen Gelenkstellungen sowie den positiven und den negativen Gelenkstellungsraum. Die Gelenkstellungsräume sind durch die jeweiligen Transformationsbereiche begrenzt. Zusätzlich ergeben sind beim Kinematiktyp "Delta-Picker" weitere Einschränkungen durch unzulässige Gelenkstellungen sowie durch singulare Stellungen (Seite 2318) bei den Kinematiktypen "Knickarm", "SCARA" und "Zylindrischer Roboter". Beachten Sie auch sich ergebende bauliche Begrenzungen durch den Aufstellort der Kinematik.

Das Technologieobjekt Kinematik zeigt die aktuelle Gelenkstellung in der Variable "<TO>.StatusKinematics.LinkConstellation" an.

Die Kinematik kann den Gelenkstellungsraum während einer Kinematikbewegung nicht verlassen. Den Gelenkstellungsraum können Sie über Einzelachsbewegungen wechseln.

Beispiel: Kinematiktyp "SCARA"

Eine Kinematik "SCARA" soll ein Objekt von einer Palette auf eine andere umlagern. Die Kinematik kann die zweite Palette wegen einer Wand nicht erreichen, ohne dass die Achse A2 den Gelenkstellungsraum wechselt.

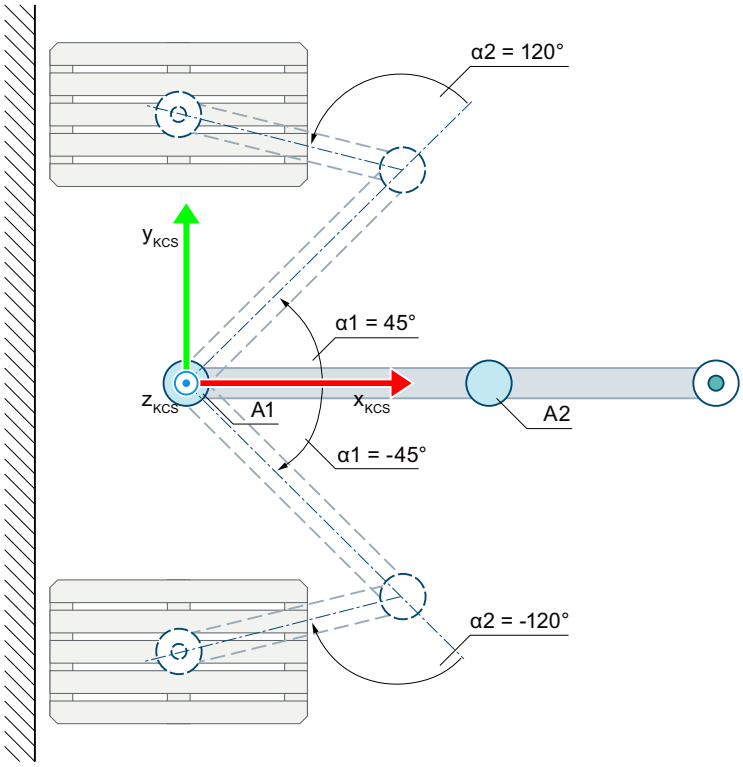

Nullstellung der Kinematik

Auslenkung der Kinematik in positiver Richtung bei α1 = 45.0° mit positiver Gelenkstellung bei α2 = 120.0°

Auslenkung der Kinematik in negativer Richtung bei α1 = -45.0° mit negativer Gelenkstellung bei α2 = -120.0°

α1 Auslenkung der Achse A1 in positiver Richtung bei α1 = 45.0°

Auslenkung der Achse A1 in negativer Richtung bei α1 = -45.0°

α2 Die Auslenkung der Achse A2 in positiver Richtung bei α2 = 120.0° ergibt eine positive Gelenkstellung.

Die Auslenkung der Achse A2 in negativer Richtung bei α2 = -120.0° ergibt eine negative Gelenkstellung.

Legende zur Darstellung der Kinematiken (Seite 2238)

Singuläre Stellungen

Abhängig vom Kinematiktyp sind bei der Rückwärtstransformation kartesische Koordinaten möglich, die nicht eindeutig in Achsstellungen der Kinematikachsen umgerechnet werden können. Dies tritt auf, wenn sich der Nullpunkt des Flanschkoordinatensystems (FCS) auf der z-Achse des Kinematikkoordinatensystems (KCS) befindet. Die kartesischen Koordinaten, an denen dieses Verhalten auftritt, werden als singuläre Stellungen bezeichnet.

Folgende Kinematiktypen des Technologieobjekts Kinematik besitzen singuläre Stellungen:

- Knickarm 3D
- Knickarm 3D mit Orientierung
- SCARA 3D mit Orientierung
- Zylindrischer Roboter 3D
- Zylindrischer Roboter 3D mit Orientierung

Verhalten in singulären Stellungen

Eine Bahnbewegung auf oder durch eine singuläre Stellung ist nicht möglich. Beim Erreichen der singulären Stellung wird der Technologie-Alarm 803 "Fehler bei Berechnung der Transformation" ausgegeben (Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen). Die Kinematik baut sollwertseitige Dynamiküberhöhungen mit der maximalen Dynamik der Achse ab. Der Abbau der Dynamiküberhöhungen kann zu unerwarteten Verfahrbewegungen der Achse führen.

Dynamiküberhöhung in der Nähe von singulären Stellungen

Wenn die Bahnbewegung in der Nähe von singulären Stellungen verläuft, können Kinematikachsen stark beschleunigen bzw. verzögern und mit sehr hoher Geschwindigkeit verfahren. Dadurch können die Dynamikgrenzen der Achsen überschritten werden. Wenn die Dynamikgrenzen der Achsen bezüglich Geschwindigkeit, Beschleunigung oder Verzögerung überschritten werden, wird dies am Technologieobjekt-Datenbaustein der entsprechenden Achse angezeigt und der Technologie-Alarm 511 "Dynamikgrenze wird durch die Kinematikbewegung verletzt" ausgelöst. Dies ist eine Warnung und beinhaltet keine Alarmreaktion. Die Kinematikbewegung wird nicht gestoppt.

Die Größe des Bereichs, in dem dieses Verhalten auftreten kann, hängt von der verwendeten Kinematik ab.

Mögliche Maßnahmen

Um ein Fahren in oder in die Nähe von singulären Stellungen zu vermeiden, treffen Sie folgende Maßnahmen:

1. Planen Sie die Bahnbewegungen so, dass die Kinematik nicht in oder in die Nähe von singulären Stellungen fährt.
2. Prüfen Sie, ob bei Ihrer Kinematik ein Fahren in oder in die Nähe von singulären Stellungen durch Sperrzonen oder Software-Endschalter vermieden werden kann.
3. Nehmen Sie eine Kinematik mit singulären Stellungen innerhalb oder am Rande des Transformationsbereichs mit aktivierter Dynamikadaption in Betrieb. Mit der Dynamikadaption vermeiden Sie unerwartete Dynamiküberhöhungen und daraus resultierende Verfahrbewegungen der Achse.

Mechanische Kopplungen (kinematikabhängig)

Wenn sich durch die Bewegung einer Kinematikachse die Stellung einer weiteren Kinematikachse verändert, sind diese beiden Achsen mechanisch gekoppelt. Mechanische Kopplungen zwischen zwei Kinematikachsen können sich konstruktionsbedingt ergeben. Wenn z. B. bei einer Kinematik "SCARA" die Orientierungsachse auf der Spindel einer Linearachse befestigt ist, ändert sich die Orientierung durch die Bewegung der Linearachse.

Die Kinematiktransformation kompensiert mechanische Kopplungen mit einem Kopplungsfaktor. Die mechanischen Kopplungen und Kopplungsfaktoren geben Sie abhängig vom Kinematiktyp bei der Konfiguration der Kinematik an.

Transformation für anwenderdefinierte Kinematiken

Anwendertransformation

Im Gegensatz zu den vordefinierten Kinematiktypen müssen Sie bei anwenderdefinierten Kinematiken die Transformation in einem eigenen Programm berechnen. Wie bei den vordefinierten Kinematiktypen übernimmt das Technologieobjekt Kinematik folgende Aufgaben:

- Bearbeitung der Motion Control-Anweisungen
- Überwachungsfunktionen
- Kommunikation mit den verschalteten Achsen

Die Anwendertransformation der kartesischen Koordinaten und der achsspezifischen Sollwerte programmieren Sie im Organisationsbaustein MC-Transformation [OB98]. Dabei umfasst die Programmierung die Transformation der Positionen und der Dynamikwerte (Geschwindigkeit, Beschleunigung, Ruck). Die Parameter einer anwenderdefinierten Kinematik definieren Sie frei in den Variablen des Technologieobjekts Kinematik "<TO>.Kinematics.Parameter[1..32]" bzw. unter "Technologieobjekt > Konfiguration > Geometrie".

Wenn Sie den MC-Transformation [OB98] im TIA Portal hinzufügen, wird automatisch der Systemdatenbaustein "TransformationParameter" unter "Programmbausteine > Systembausteine > Programmressourcen" angelegt. In den Eigenschaften des Organisationsbausteins unter "Allgemein > Transformation" zeigt der MC-Transformation [OB98] die Nummer des Systemdatenbausteins "TransformationParameter" an. Im Systemdatenbaustein "TransformationParameter" schreiben und lesen Sie die achsspezifischen Daten bzw. die kartesischen Daten der zu transformierenden Kinematik.

Programmierung

Die folgende Grafik zeigt die Schnittstellen und das Zusammenwirken von Systemleistung und Anwendertransformation:

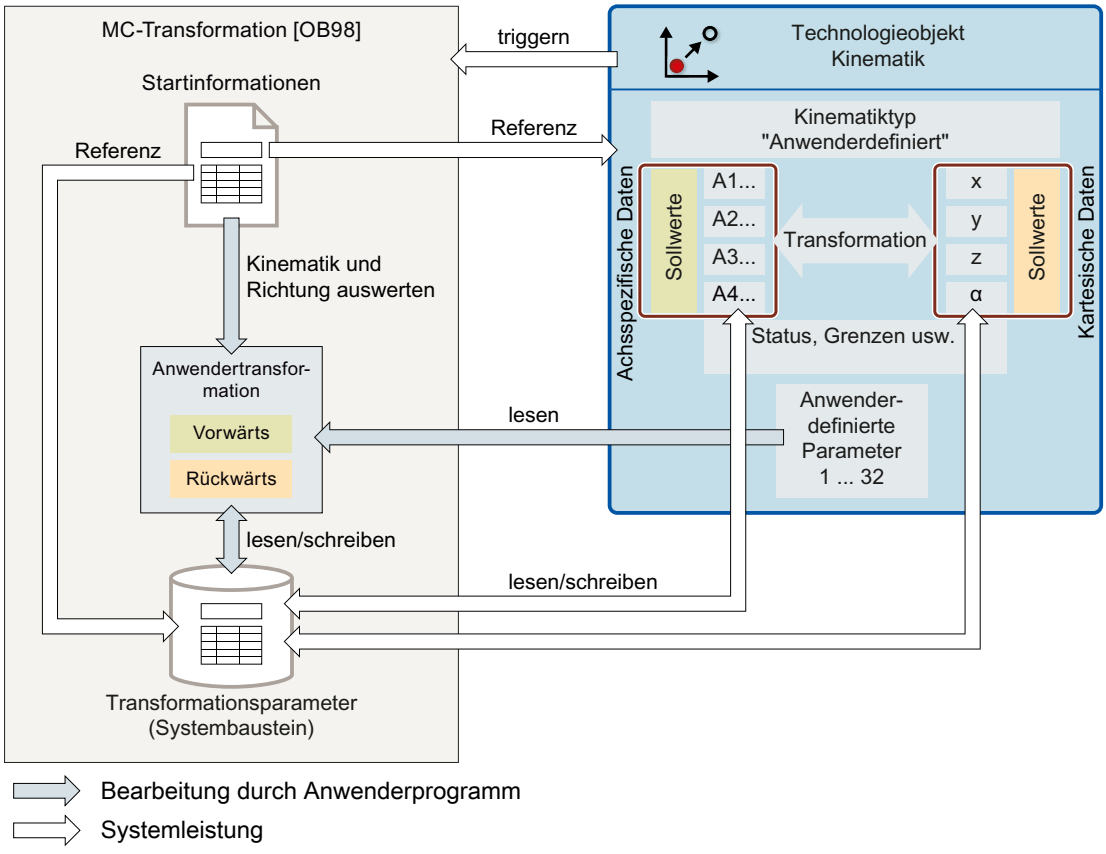

Das Technologieobjekt Kinematik ruft automatisch den MC-Transformation [OB98] auf. Der MC-Transformation [OB98] enthält dabei folgende Startinformationen:

- Technologieobjekt Kinematik, das den MC-Transformation [OB98] aufruft
- Benötigte Richtung der Transformation (Vorwärts- oder Rückwärtstransformation)
- Bearbeitungskontext der Transformation (aktuelle Bewegung oder Bewegungsplanung)
- Zeiger auf den Systemdatenbaustein "TransformationParameter" (VARIANT)

Diese Startinformationen werten Sie in Ihrem Anwenderprogramm aus. Im MC-Transformation [OB98] programmieren Sie die Algorithmen für die Berechnung der achsspezifischen Daten bzw. der kartesischen Daten aller anwenderdefinierten Kinematiken. Die dafür benötigten Kinematikparameter lesen Sie aus den Variablen des Technologieobjekts "<TO>.Kinematics.Parameter[1..32]" aus. Das Ergebnis der Berechnung schreiben Sie in die Schnittstelle "TransformationParameter".

Anschließend werden die Transformationsparameter automatisch in das Technologieobjekt Kinematik übernommen und weiterverarbeitet. Das Technologieobjekt Kinematik gibt die Sollwerte an die Kinematikachsen aus.

Einschränkungen bei Transformation von mehreren Kinematiken

Jedes Technologieobjekt Kinematik ruft den MC-Transformation [OB98] auf. In einem MC-Transformation [OB98] kann jeweils nur eine Kinematik transformiert werden. In einem Servo-Takt kann jeweils nur ein MC-Transformation [OB98] aufgerufen werden.

MC-Transformation [OB98]

Referenzdeklaration für Systemdatenbaustein "TransformationParameter"

Im MC-Transformation [OB98] müssen Sie für den Systemdatenbaustein "TransformationParameter" eine Referenz auf den Datentyp angeben. Im "Temp"-Bereich der Bausteinschnittstelle legen Sie dazu eine Variable mit dem folgenden Datentyp an:

"REF_TO TO_Struct_TransformationParameter_V1"

Um auf den Systemdatenbaustein "TransformationParameter" zugreifen zu können, weisen Sie über den folgenden Casting-Befehl den Datentyp "TO_Struct_TransformationParameter_V1" zu:

```
#P ?= #TransformationParameters;
```

Die Deklaration ist in einem Programmbeispiel (Seite 2324) beschrieben.

Bausteinaufruf

Der MC-Transformation [OB98] wird im Motion Control-Applikationszyklus entsprechend der konfigurierten Priorität aufgerufen. Beim Aufruf versorgt das Technologieobjekt Kinematik die Parameter des MC-Transformation [OB98]:

Parameter	Deklaration	Datentyp	Default-wert	Beschreibung	
KinematicsObject	INPUT	DB_ANY	-	Technologieobjekt Kinematik, für das der MC-Transformation [OB98] bei Aufruf die Transformation berechnet.	
ExecutionContext	INPUT	DINT	-	Bearbeitungskontext des OB MC-Transformation	
				0	MOTION_EXECUTION Berechnung der Achssollwerte in der Bewegungsausführung im MC-Interpolator [OB92]. Die berechneten Werte sind für die aktuelle Bewegungsführung notwendig.
				1	NON_MOTION_EXECUTION Die Transformation ist für die Bewegungsplanung notwendig (aktuell keine Bewegung).
TransformationType	INPUT	DINT	-	Angeforderte Berechnung	
				0	Vorwärtstransformation Berechnung der kartesischen Parameter aus den Achsstellungen
				1	Rückwärtstransformation Berechnung der achsspezifischen Parameter aus den kartesischen Parametern

Parameter	Deklaration	Datentyp	Default-wert	Beschreibung	
TransformationParameters	InOut	VARIANT	-	Zeiger auf den Systemdatenbaustein "Transformation-Parameter"	
FunctionResult	OUTPUT	DINT	-	Rückgabewert des MC-Transformation [OB98] an das Technologieobjekt Kinematik	
				0	Berechnung durchgeführt und Parameter ausgegeben
				≠ 0	Fehler bei der Berechnung (anwenderdefiniert) Wenn bei der Berechnung Fehler auftreten, stoppt das Technologieobjekt Kinematik die Bewegung. Das Technologieobjekt Kinematik gibt einen Technologie-Alarm mit der Fehlerkennung als Begleitwert aus und löscht die Auftragskette. Der Wert kann größer oder kleiner als null sein.

Priorität

Die Priorität des MC-Transformation [OB98] konfigurieren Sie in den Eigenschaften des Organisationsbausteins unter "Allgemein > Attribute > Priorität". Für die Priorität können Sie Werte von 17 bis 25 einstellen (Voreinstellung 25):

- Die Priorität des MC-Transformation [OB98] muss mindestens um eins niedriger sein als die Priorität des MC-Servo [OB91].
- Die Priorität des MC-Transformation [OB98] muss mindestens um eins höher sein als die Priorität des MC-Interpolator [OB92].

Variablen des Systemdatenbausteins "TransformationParameter"

Die folgende Tabelle zeigt die Variablen im Systemdatenbaustein "TransformationParameter":

Variable	Datentyp	Beschreibung
AxisData.	STRUCT_ Transformation AxisData_V1	Achsspezifische Parameter
a1Position	LREAL	Sollposition der Achse A1
a1Velocity	LREAL	Sollgeschwindigkeit der Achse A1
a1Acceleration	LREAL	Sollbeschleunigung der Achse A1
a2Position	LREAL	Sollposition der Achse A2
a2Velocity	LREAL	Sollgeschwindigkeit der Achse A2
a2Acceleration	LREAL	Sollbeschleunigung der Achse A2
a3Position	LREAL	Sollposition der Achse A3
a3Velocity	LREAL	Sollgeschwindigkeit der Achse A3
a3Acceleration	LREAL	Sollbeschleunigung der Achse A3
a4Position	LREAL	Sollposition der Achse A4
a4Velocity	LREAL	Sollgeschwindigkeit der Achse A4
a4Acceleration	LREAL	Sollbeschleunigung der Achse A4

Variable	Datentyp	Beschreibung
CartesianData	STRUCT_ Transformation CartesianDa-ta_V1	Kartesische Parameter und Gelenkstellung
xPosition	LREAL	x-Position
xVelocity	LREAL	x-Geschwindigkeit
xAcceleration	LREAL	x-Beschleunigung
yPosition	LREAL	y-Position
yVelocity	LREAL	y-Geschwindigkeit
yAcceleration	LREAL	y-Beschleunigung
zPosition	LREAL	z-Position
zVelocity	LREAL	z-Geschwindigkeit
zAcceleration	LREAL	z-Beschleunigung
aPosition	LREAL	A-Position (Orientierung)
aVelocity	LREAL	A-Geschwindigkeit (Orientierung)
aAcceleration	LREAL	A-Beschleunigung (Orientierung)
LinkConstellation	DWORD	Gelenkstellungsbereich

Programmbeispiel für eine anwenderdefinierte Kinematik

Im Folgenden wird ein Beispiel für die Anwendertransformation im MC-Transformation [OB98] einer 2D-Kinematik mit dem Namen "KinematicsUserDefined2D" beschrieben. Für diese Kinematik wurden zwei Transformationsparameter unter "Technologieobjekt > Konfiguration > Geometrie" definiert.

Die folgende Tabelle zeigt die Deklaration der verwendeten Variablen:

Variable	Deklaration	Datentyp	Beschreibung
KinematicsOb-ject	Input	DB_ANY	Referenz auf das Technologieobjekt
Transformation-Type	Input	DInt	Transformationsrichtung 0: Vorwärtstransformation 1: Rückwärtstransformation
FunctionResult	Output	DInt	Transformationsergebnis 0: Erfolgreich < 0: Fehler
Transformation-Parameters	InOut	Variant	Referenz auf den Systemdatenbaustein "TransformationParame-ter"
P	Temp	REF_TO TO_Struct_Tran sformationPara-meter_V1	Temporäre Variable für den Casting-Befehl
GearRatioA1	Temp	LReal	Temporäre Variable zum Auslesen der definierten Transformations-parameter
GearRatioA2	Temp	LReal	Temporäre Variable zum Auslesen der definierten Transformations-parameter
InvalidCast	Constant	DInt	Rückgabewert für nicht erfolgreiches Casting

Das Programmbeispiel ist wie folgt aufgebaut:

- Casting-Befehl zum Zugriff auf den Systemdatenbaustein "TransformationParameter"
- Auswertung des Technologieobjekts
- Auslesen der definierten Transformationsparameter
- Auswertung der Transformationsrichtung
- Berechnung der kartesischen Koordinaten aus den Achsstellungen der Kinematikachsen (Vorwärtstransformation)
- Berechnung der Achsstellungen der Kinematikachsen aus den kartesischen Koordinaten (Rückwärtstransformation)

SCL

```
//Caste of the variant "TransformationParameters" to the referenced datatype
//"TO_Struct_TransformationParameter_V1".
//This has to be done in order to access the variant pointer, which references
//the "TransformationParameters" where the "AxisData" and "CartesianData" for
//the calculation of user transformation are stored.
#P ?= #TransformationParameters;

//Check if cast of "TransformationParameters" was successfull. Otherwise abort calculation.
IF #P = NULL THEN
    #FunctionResult := #InvalidCast;
    RETURN;
END_IF;

//Check if "KinematicsUserDefined2D" needs transformation.
IF #KinematicsObject = "KinematicsUserDefined2D" THEN

    //Read the user defined cartesian parameters.
    #GearRatioA1 := "KinematicsUserDefined2D".Kinematics.Parameter[1];
    #GearRatioA2 := "KinematicsUserDefined2D".Kinematics.Parameter[2];

    //Calculate the forward transformation "AxisData" -> "CartesianData".
    //The system fills the "AxisData" of "TransformationParameters" with values.
    //To calculate the "CartesianData" evaluate "AxisData".
    IF #TransformationType = 0 THEN

        //Calculate the position, velocity and acceleration component for the x-vector.
        #P^.CartesianData.xPosition := #P^.AxisData.a1Position * #GearRatioA1;
        #P^.CartesianData.xVelocity := #P^.AxisData.a1Velocity * #GearRatioA1;
        #P^.CartesianData.xAcceleration := #P^.AxisData.a1Acceleration * #GearRatioA1;

        //Calculate the position, velocity and acceleration component for the z-vector.
        #P^.CartesianData.zPosition := #P^.AxisData.a2Position * #GearRatioA2;
        #P^.CartesianData.zVelocity := #P^.AxisData.a2Velocity * #GearRatioA2;
        #P^.CartesianData.zAcceleration := #P^.AxisData.a2Acceleration * #GearRatioA2;
        //Link constellation can be set to 0 here, hence it is not needed.
        #P^.CartesianData.LinkConstellation := 16#0000;
```

SCL

```
        //Transformation was successfull.
        #FunctionResult := 0;

    //Calculate the backward transformation "CartesianData" -> "AxisData".
    //The system fills the "CartesianData" of "TransformationParameters" with values.
    //To calculate the "AxisData" evaluate "CartesianData".
    ELSIF #TransformationType = 1 THEN

        //Calculate the position, velocity and acceleration component for the first axis.
        #P^.AxisData.a1Position := #P^.CartesianData.xPosition / #GearRatioA1;
        #P^.AxisData.a1Velocity := #P^.CartesianData.xVelocity / #GearRatioA1;
        #P^.AxisData.a1Acceleration := #P^.CartesianData.xAcceleration / #GearRatioA1;

        //Calculate the position, velocity and acceleration component for the second axis.
        #P^.AxisData.a2Position := #P^.CartesianData.zPosition / #GearRatioA2;
        #P^.AxisData.a2Velocity := #P^.CartesianData.zVelocity / #GearRatioA2;
        #P^.AxisData.a2Acceleration := #P^.CartesianData.zAcceleration / #GearRatioA2;

        //Transformation was successfull.
        #FunctionResult := 0;

    END_IF;
END_IF;
```

Siehe auch

MC-Transformation [OB98] (Seite 2322)

Variablen Kinematiktransformation

Für die Kinematiktransformation sind folgende Variablen des Technologieobjekts Kinematik relevant:

Variable	Beschreibung	
Statuswerte		
<TO>.StatusKinematics.Valid	TRUE	Transformation/kartesische Werte gültig
	FALSE	Transformation/kartesische Werte ungültig
<TO>.StatusKinematics.LinkConstellation	Gelenkstellung	
<TO>.FlangeInKcs	Aktueller Transformations-Frame (mit Dynamik, Sollwertbezug)	

Kinematikbewegungen

Mit Kinematikbewegungen bewegen Sie die Kinematik durch den dreidimensionalen Raum. Planen Sie die Kinematikbewegung voraus. Beachten Sie dabei Folgendes:

- Erreichbare Punkte der Kinematik
- Zonen
- Transformationsbereiche

- Gelenkstellungsräume
- Software-Endschalter der Achsen

Die Orientierungsbewegung ist die Bewegung der kartesischen Orientierung und erfolgt gleichzeitig zur Kinematikbewegung. Beim Überschleifen von Bewegungen wird die Orientierungsbewegung ebenfalls überschliffen. Wenn die Kinematikbewegung stoppt, stoppt auch die Orientierungsbewegung.

Bezugssystem

Die Zielposition und die Zielorientierung, die Sie für eine Kinematikbewegung angeben, können sich auf das Weltkoordinatensystem (WCS) oder auf ein Objektkoordinatensystem (OCS) beziehen.

Bewegungsarten

Linearbewegung

Sie können eine Kinematik mit einer linearen Bewegung verfahren. Über die Motion Control-Anweisungen "MC_MoveLinearAbsolute" und "MC_MoveLinearRelative" definieren Sie die Linearbewegung. Während mit einem "MC_MoveLinearAbsolute"-Auftrag die Kinematik auf eine absolute Position verfahren wird, wird mit einem "MC_MoveLinearRelative"-Auftrag die Kinematik relativ zu der aktuellen Position verfahren. Von der aktuellen Position verfährt die Kinematik mit einer Linearbewegung zur definierten Zielposition.

Kreisbewegung

Sie können eine Kinematik mit einer zirkularen Bewegung verfahren. Über die Motion Control-Anweisungen "MC_MoveCircularAbsolute" und "MC_MoveCircularRelative" definieren Sie die Kreisbewegung. Während mit einem "MC_MoveCircularAbsolute"-Auftrag die Kinematik auf eine absolute Position verfahren wird, wird mit einem "MC_MoveCircularRelative"-Auftrag die Kinematik relativ zu der aktuellen Position verfahren.

Definition der Kreisbahn ("CircMode")

Mit dem Parameter "CircMode" legen Sie die Definition der Kreisbahn fest. Abhängig von diesem Parameterwert wird die Kreisbahn wie folgt berechnet:

- Über einen Zwischen- und den Endpunkt ("CircMode" = 0)
  Mit dem Zwischenpunkt geben Sie einen Punkt auf der Kreisbahn an, über den der Endpunkt angefahren werden soll. Aus dem Start-, Zwischen- und Endpunkt wird die Kreisbahn berechnet. Dabei können nur Kreisbahnen unter 360° verfahren werden.
- Über den Kreismittelpunkt und den Winkel in einer Hauptebene ("CircMode" = 1)
  Mit dem Kreismittelpunkt und dem Winkel wird der Endpunkt der Kreisbahn berechnet. Dabei legen Sie mit dem Parameter "PathChoice" fest, ob die Kreisbahn in positiver oder negativer Drehrichtung verfahren werden soll. Zusätzlich geben Sie über den Parameter "CirclePlane" die Hauptebene an, in welcher die Kreisbahn verfahren werden soll.
- Über den Kreisradius und den Endpunkt in einer Hauptebene ("CircMode" = 2)
  Mit dem Kreisradius und dem Endpunkt wird die Kreisbahn berechnet. Dabei können sich bis zu vier mögliche Kreisbahnen ergeben. Mit dem Parameter "PathChoice" legen Sie fest, welche der vier möglichen Kreisbahnen verfahren werden soll. Zusätzlich geben Sie über den Parameter "CirclePlane" die Hauptebene an, in welcher die Kreisbahn verfahren werden soll.

Achten Sie bei der Definition von Zwischen- und Endpunkt, Kreismittelpunkt und Winkel bzw. Kreisradius und Endpunkt darauf, dass die Angaben konsistent zueinander sind.

Orientierungsrichtung der Kreisbahn ("PathChoice")

Wenn die Kreisbahn über den Kreismittelpunkt und den Winkel berechnet werden soll, definieren Sie mit dem Parameter "PathChoice", ob die Kreisbahn in positiver oder negativer Drehrichtung verfahren werden soll.

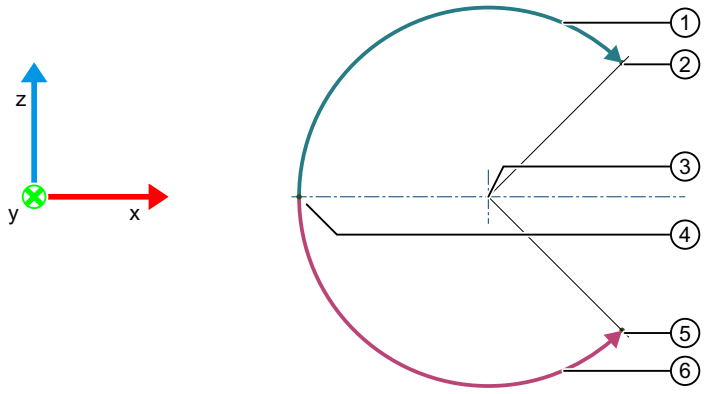

① Positive Drehrichtung ("PathChoice" = 0)
② Endpunkt
③ Kreismittelpunkt
④ Startpunkt
⑤ Endpunkt
⑥ Negative Drehrichtung ("PathChoice" = 1)

Wenn die Kreisbahn über den Kreisradius und den Endpunkt berechnet werden soll, definieren Sie mit dem Parameter "PathChoice", welche der vier möglichen Kreisbahnen verfahren

werden soll. Dabei wird sowohl zwischen der positiven und negativen Drehrichtung als auch zwischen dem längeren und kürzeren Kreissegment unterschieden.

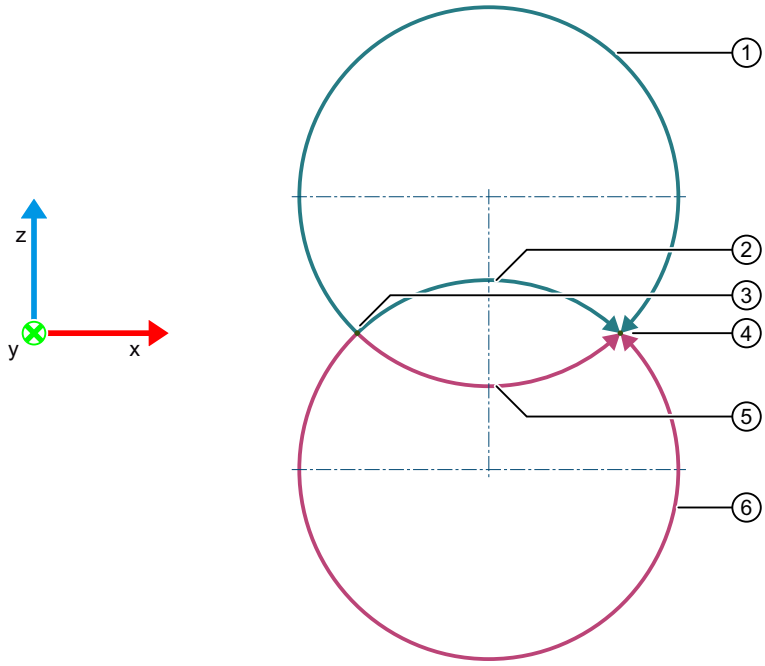

① Längeres positives Kreissegment ("PathChoice" = 2)
② Kürzeres positives Kreissegment ("PathChoice" = 0)
③ Startpunkt
④ Endpunkt
⑤ Kürzeres negatives Kreissegment ("PathChoice" = 1)
⑥ Längeres negatives Kreissegment ("PathChoice" = 3)

Bewegungsdynamik

Dynamik von Kinematik- und Orientierungsbewegung

Die Dynamikwerte (Geschwindigkeit, Beschleunigung, Ruck) einer Kinematikbewegung geben Sie an der entsprechenden Motion Control-Anweisung an.

Dynamikvoreinstellungen

Wenn Sie für Bewegungsaufträge keine Dynamikwerte angeben (Standardwert "-1.0"), werden die unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierten Dynamikvoreinstellungen für die Kinematikbewegung verwendet. Für die Orientierungsbewegung können Sie die Dynamikwerte nur über die Dynamikvoreinstellungen angeben.

Wenn Sie die Dynamikvoreinstellungen während einer laufenden Bewegung ändern, werden die geänderten Werte erst mit dem nächsten Bewegungsauftrag wirksam.

Dynamikgrenzen der Kinematik

Bei der Bewegungsführung werden die unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierten Dynamikgrenzen der Kinematik berücksichtigt. Die Dynamik einer Bewegung wird gegebenenfalls beschränkt, sodass die Dynamikgrenzen der Kinematik nicht überschritten werden. Wenn Sie die Dynamikgrenzen der Kinematik ändern, werden die Änderungen für die Kinematik- und Orientierungsbewegung direkt wirksam.

Dynamikgrenzen der Achsen

Beim Absetzen eines Bewegungsauftrags werden die unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Begrenzungen > Dynamikgrenzen" konfigurierten Dynamikgrenzen der Achsen nur berücksichtigt, wenn die Dynamikadaption aktiv ist. Die Dynamik der Bewegung wird gegebenenfalls beschränkt, sodass die Dynamikgrenzen der Achsen nicht überschritten werden. Wenn Sie die Dynamikgrenzen der Achsen während einer laufenden Bewegung ändern, werden die geänderten Werte erst bei dem nächsten Bewegungsauftrag wirksam.

Dynamikadaption

Die Dynamikadaption stellen Sie unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" ein. Bei aktivierter Dynamikadaption wird ein Geschwindigkeitsprofil für die gesamte Bewegung berechnet, das die Dynamikgrenzen der Achsen und der Kinematik berücksichtigt (<TO>.StatusPath.DynamicAdaption). In die Dynamikadaption werden Geschwindigkeit und Beschleunigung einbezogen. Bei der Beschleunigung wird die Tangential- und Radialbeschleunigung der Bahn berücksichtigt. Der Ruck wird bei der Dynamikadaption nicht begrenzt.

Die folgende Grafik zeigt beispielhaft einen Geschwindigkeitsverlauf mit und ohne Dynamikadaption:

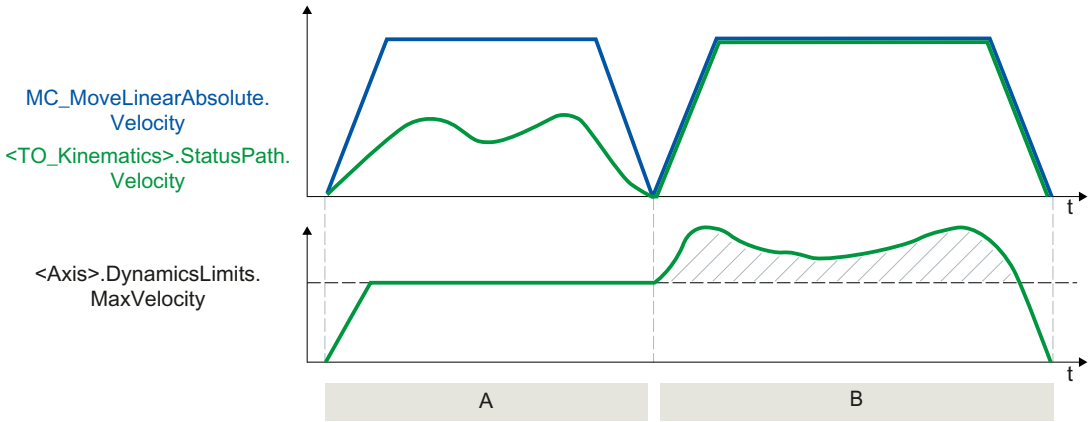

A	Mit Dynamikadaption
B	Ohne Dynamikadaption

Bei der Dynamikadaption ohne Segmentierung der Bahn wird das Geschwindigkeitsprofil unter der Berücksichtigung der Dynamikgrenzen der Achsen, die für die gesamte Bewegung gelten, berechnet.

Bei der Dynamikadaption mit Segmentierung wird die Bahn in einzelne äquidistante Segmente unterteilt. Für jedes dieser Segmente wird das Geschwindigkeitsprofil unter der Berücksichtigung der Dynamikgrenzen der Achsen, die für einzelne Abschnitte der Bewegung gelten, berechnet. Die Dynamik wird somit für einzelne Abschnitte einer Bewegung angepasst.

Override

Über den Technologieobjekt-Datenbaustein können Sie einen Geschwindigkeits-Override für die Kinematik festlegen (<TO>.Override.Velocity). Sie können einen Wert zwischen 0 % und 200 % angeben. Der Geschwindigkeits-Override wirkt auf die Geschwindigkeit des Werkzeugarbeitspunkts (TCP) entlang der Bahn. Wenn Sie den Geschwindigkeits-Override der Kinematik ändern, wird die Änderung für die Kinematikbewegung direkt wirksam.

Die Sollgeschwindigkeit der Bewegung ist die an der Motion Control-Anweisung angegebene Geschwindigkeit multipliziert mit dem prozentualen Wert des Geschwindigkeits-Override.

Die achsspezifischen Geschwindigkeits-Override-Werte sind bei Kinematikbewegungen nicht wirksam.

Variablen Bewegungsführung und Dynamik

Folgende Variablen des Technologieobjekts sind für die Bewegungsführung relevant:

Variable	Beschreibung	
Statuswerte		
<TO>.StatusWord	Statusanzeige für eine aktive Bewegung	
<TO>.StatusPath.CoordSystem	Koordinatensystem des aktiven Bewegungsauftrags	
	0	Weltkoordinatensystem
	1, 2, 3	Objektkoordinatensystem 1, 2, 3
<TO>.Tcp	Zielkoordinaten der Kinematikbewegung im Weltkoordinatensystem x, y, z, A	
<TO>.StatusPath.Velocity	Aktuelle Bahngeschwindigkeit (Sollwertbezug)	
<TO>.StatusPath.Acceleration	Aktuelle Bahnbeschleunigung (Sollwertbezug)	
<TO>.StatusPath.DynamicAdaption	Dynamikadaption	
	0	Keine Dynamikadaption
	1	Dynamikadaption mit Segmentierung der Bahn
	2	Dynamikadaption ohne Segmentierung der Bahn
<TO>.StatusMotionQueue.NumberOfCommands	Anzahl der Aufträge in der Auftragskette	
Überlagerung		
<TO>.Override.Velocity	Geschwindigkeits-Override	
Dynamikgrenzwerte		
<TO>.DynamicLimits.Path.Velocity	Dynamikbegrenzung für die maximale Geschwindigkeit der Bahn	
<TO>.DynamicLimits.Path.Acceleration	Dynamikbegrenzung für die maximale Beschleunigung der Bahn	
<TO>.DynamicLimits.Path.Deceleration	Dynamikbegrenzung für die maximale Verzögerung der Bahn	
<TO>.DynamicLimits.Path.Jerk	Dynamikbegrenzung für den maximalen Ruck der Bahn	
<TO>.DynamicLimits.Orientation.Velocity	Dynamikbegrenzung für die maximale Geschwindigkeit der kartesischen Orientierung	

Variable	Beschreibung	
<TO>.DynamicLimits.Orientation.Acceleration	Dynamikbegrenzung für die maximale Beschleunigung der kartesischen Orientierung	
<TO>.DynamicLimits.Orientation.Deceleration	Dynamikbegrenzung für die maximale Verzögerung der kartesischen Orientierung	
<TO>.DynamicLimits.Orientation.Jerk	Dynamikbegrenzung für den maximalen Ruck der kartesischen Orientierung	
Dynamikvoreinstellungen		
<TO>.DynamicDefaults.Path.Velocity	Voreinstellung der Geschwindigkeit der Bahn	
<TO>.DynamicDefaults.Path.Acceleration	Voreinstellung der Beschleunigung der Bahn	
<TO>.DynamicDefaults.Path.Deceleration	Voreinstellung der Verzögerung der Bahn	
<TO>.DynamicDefaults.Path.Jerk	Voreinstellung des Rucks der Bahn	
<TO>.DynamicDefaults.Orientation.Velocity	Voreinstellung der Geschwindigkeit der kartesischen Orientierung	
<TO>.DynamicDefaults.Orientation.Acceleration	Voreinstellung der Beschleunigung der kartesischen Orientierung	
<TO>.DynamicDefaults.Orientation.Deceleration	Voreinstellung der Verzögerung der kartesischen Orientierung	
<TO>.DynamicDefaults.Orientation.Jerk	Voreinstellung des Rucks der kartesischen Orientierung	
<TO>.DynamicDefaults.DynamicAdaption	Voreinstellung der Dynamikadaption	
	0	Keine Dynamikadaption
	1	Dynamikadaption mit Segmentierung der Bahn
	2	Dynamikadaption ohne Segmentierung der Bahn

Zonenüberwachung

Die Zonenüberwachung erfüllt folgende Aufgaben:

- Schutz vor Kollisionen mit mechanischen Einbauten
- Auslösen von prozessbedingten Aktionen (Meldezonen)

⚠ WARNUNG

Kein Schutz von Menschen

Die Zonenüberwachung ist nicht zum Schutz von Menschen geeignet.

Führen Sie zum Schutz von Menschen geeignete Schutzmaßnahmen durch, z. B. Schutzzäune aufstellen, Sicherheitstüren installieren usw.

Zonen sind geometrische Körper, mit denen Sie den Arbeitsraum einer Kinematik beschreiben und unterteilen können. Am Technologieobjekt Kinematik können Sie Arbeitsraumzonen und Kinematikzonen konfigurieren. Arbeitsraumzonen beschreiben die Umgebung einer Kinematik. Kinematikzonen umhüllen den Endpunkt einer Kinematik (Flansch oder Werkzeug).

Die folgende Grafik zeigt die Zonen einer Kinematik:

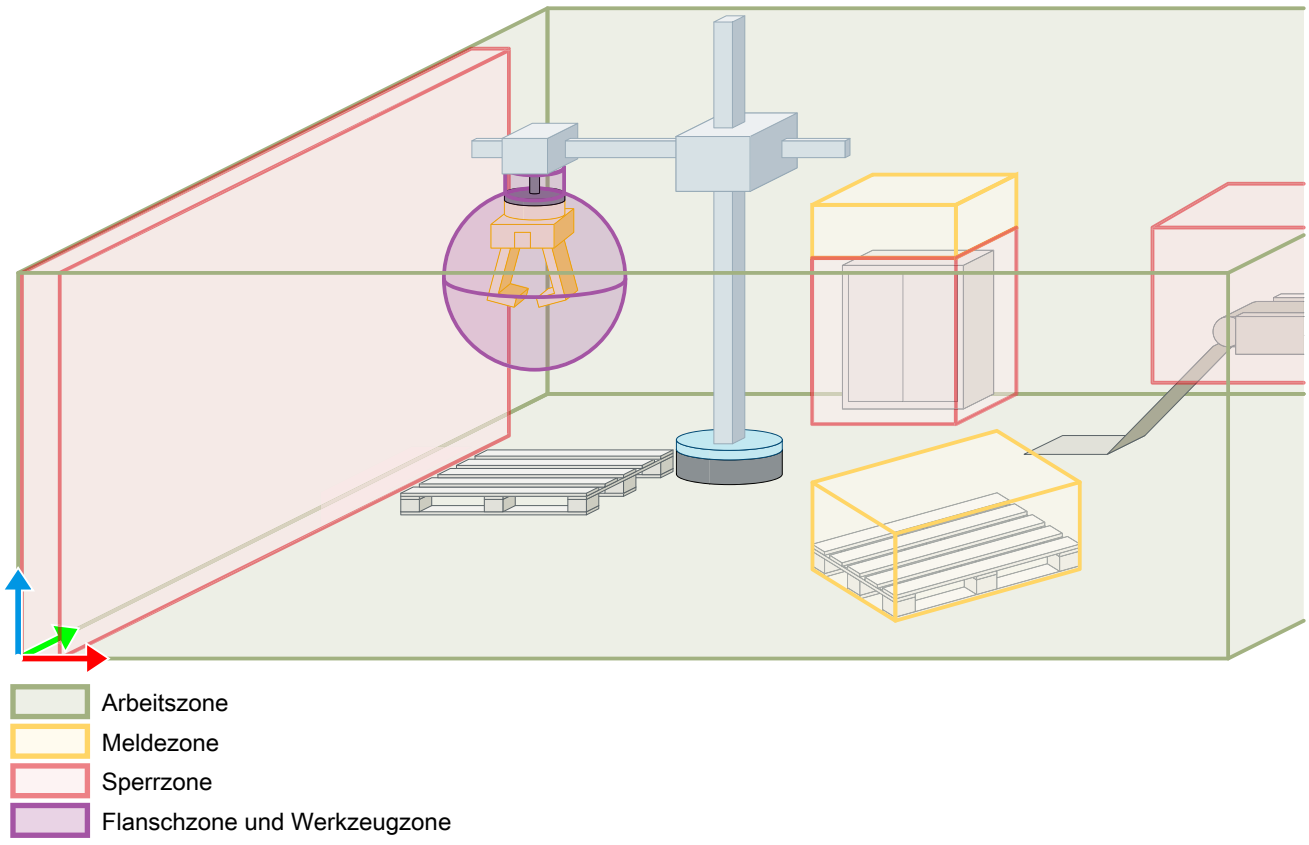

Zonenkonfiguration

Die Zonen können Sie über die Konfiguration des Technologieobjekts Kinematik oder in Ihrem Anwenderprogramm über Motion Control-Anweisungen vorgeben und aktivieren/deaktivieren.

Zonenüberwachung

Die Zonenüberwachung prüft alle aktivierten Arbeitsraumzonen (Arbeitszonen, Meldezonen, Sperrzonen) auf Kollision mit allen aktivierten Kinematikzonen (Flanschzonen, Werkzeugzonen). Die Zonenüberwachung überwacht die Zonen bei allen Bewegungen der Kinematik:

- Kinematikbewegungen über das Anwenderprogramm oder die Kinematiksteuertafel
- Einzelachsbewegungen über das Anwenderprogramm oder die Achssteuertafel

Der Status der Zonenüberwachung wird in der Diagnose (Seite 2386) und in den Variablen (Seite 2339) des Technologieobjekts Kinematik angezeigt.

Wenn die Zonenüberwachung eine Zonenverletzung durch eine Kinematikbewegung erkennt, treten folgende Reaktionen auf:

Zonenverletzung	Reaktion	Beschreibung
Verlassen der Arbeitszone	Alarm mit Halt	Das Technologieobjekt Kinematik gibt einen Technologie-Alarm aus. Die Kinematikbewegung wird gestoppt.
Eintreten in eine Meldezone	Alarm ohne Halt	Das Technologieobjekt Kinematik gibt einen Technologie-Alarm aus. Die Kinematikbewegung wird fortgesetzt.
Eintreten in eine Sperrzone	Alarm mit Halt	Das Technologieobjekt Kinematik gibt einen Technologie-Alarm aus. Die Kinematikbewegung wird gestoppt. Die Kinematik verletzt die Zone dabei mindestens um den Bremsweg.

Bei Zonenverletzungen durch Einzelachsbewegungen gibt das Technologieobjekt Kinematik einen Technologie-Alarm aus. Das Technologieobjekt Positionierachse/Gleichlaufachse gibt keinen Technologie-Alarm aus. Die Einzelachsbewegung wird nicht abgebrochen. Sie können die Einzelachsbewegung applikativ abbrechen.

Zusätzlich zu den Zonen am Technologieobjekt Kinematik begrenzen Sie den Verfahrraum der Kinematik durch die Software-Endschalter an den Achsen.

Freifahren nach Zonenverletzung

Nachdem Sie den Technologie-Alarm am Technologieobjekt Kinematik quittiert haben, können Sie die Kinematik wieder verfahren.

ACHTUNG
Zonenüberwachung für verletzte Zone nach Quittierung deaktiviert Nachdem Sie den Technologie-Alarm am Technologieobjekt Kinematik quittiert haben, ist die Zonenüberwachung für die verletzte Zone deaktiviert, bis die Kinematik die verletzte Sperrzone/Meldezone verlässt bzw. wieder in die verletzte Arbeitszone eintritt. Sie können die Kinematik in alle Richtungen verfahren, also auch weiter in die verletzte Sperrzone/Meldezone hinein oder weiter aus der Arbeitszone heraus. Beachten Sie die Fahrrichtung beim Freifahren der Kinematik.

Überwachen Sie das Freifahren applikativ. Der Status der Zonenüberwachung wird weiterhin im Technologieobjekt-Datenbaustein angezeigt.

Nachdem die Kinematik die verletzte Sperrzone/Meldezone wieder verlassen hat bzw. wieder in die verletzte Arbeitszone eingetreten ist, ist die Zonenüberwachung für diese Zone wieder aktiv. Somit wird beim erneuten Verletzten der Zone ein neuer Technologie-Alarm ausgelöst.

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Arbeitsraumzonen

Arbeitsraumzonen beschreiben die Umgebung einer Kinematik. Arbeitsraumzonen definieren Sie im Weltkoordinatensystem (WCS) oder im Objektkoordinatensystem (OCS). Sie können bis zu zehn Arbeitsraumzonen konfigurieren und aktivieren/deaktivieren. Die folgende Tabelle zeigt die Arbeitsraumzonen am Technologieobjekt Kinematik:

Arbeitsraumzone	Beschreibung
Arbeitszone	Arbeitszonen definieren Bereiche, in denen sich Kinematikzonen bewegen dürfen.
Meldezone	Meldezonen zeigen Folgendes an: • Kinematikzone tritt in die Meldezonen ein • Kinematikzone befindet sich in der Meldezone
Sperrzone	Sperrzonen definieren Bereiche, in die eine Kinematikzone nicht eintreten darf.

Arbeitszone

Mit Arbeitszonen schränken Sie den möglichen Verfahrraum der Kinematik ein oder definieren mehrere mögliche Arbeitsbereiche. Sie können mehrere Arbeitszonen vorgeben. Zu einem Zeitpunkt darf nur eine Arbeitszone aktiviert sein. Wenn keine Arbeitszone aktiviert ist, gilt der gesamte Verfahrraum der Kinematik als Arbeitsbereich.

Kinematikzonen müssen sich innerhalb von Arbeitszonen befinden. Wenn eine Kinematikzone eine Arbeitszone verlässt, gibt das Technologieobjekt Kinematik den Technologie-Alarm 806 aus (Alarmreaktion: Stopp mit maximalen Dynamikwerten der Kinematik). Die an der Kinematikbewegung beteiligten Achsen halten mit den am Technologieobjekt Kinematik konfigurierten maximalen Dynamikwerten an. Alle Aufträge in der Auftragskette werden abgebrochen.

Meldezone

Meldezonen sind Bereiche innerhalb des Verfahrraums der Kinematik. Meldezonen zeigen eine Zonenverletzung durch eine Kinematikzone an, lösen aber keinen Stopp der Kinematikbewegung aus. Meldezonen können teilweise außerhalb der Arbeitszone liegen.

Wenn eine Kinematikzone eine Meldezone verletzt, gibt das Technologieobjekt Kinematik den Technologie-Alarm 807 aus (keine Alarmreaktion).

Sperrzone

Sperrzonen sind Bereiche innerhalb des Verfahrraums der Kinematik, in die eine Kinematikzone nicht eintreten darf (z. B. Schaltschrank, Schutzwand). Sperrzonen können teilweise außerhalb der Arbeitszone liegen.

Wenn eine Kinematikzone eine Sperrzone verletzt, gibt das Technologieobjekt Kinematik den Technologie-Alarm 806 aus (Alarmreaktion: Stopp mit maximalen Dynamikwerten der Kinematik). Die an der Kinematikbewegung beteiligten Achsen halten mit den am Technologieobjekt Kinematik konfigurierten maximalen Dynamikwerten an. Alle Aufträge in der Auftragskette werden abgebrochen.

Kinematikzonen

Kinematikzonen sind bezogen auf den Arbeitspunkt/Flansch einer Kinematik und bewegen sich mit der Kinematik. Die Zonenüberwachung prüft Kinematikzonen auf Durchdringung mit Arbeitsraumzonen. Mit Kinematikzonen erweitern Sie den überwachten Bereich über den Werkzeugarbeitspunkt (TCP) hinaus. Sie können bis zu neun Kinematikzonen konfigurieren und aktivieren/deaktivieren. Die folgende Tabelle zeigt die Kinematikzonen am Technologieobjekt Kinematik:

Kinematikzone	Bezugssystem	Beschreibung
Werkzeugzone	TCS	Werkzeugzonen umhüllen das Werkzeug oder Teile davon.
Flanschzone	FCS	Flanschzonen umhüllen den Flansch oder Teile davon.

Werkzeugzone

Werkzeugzonen definieren Sie im Werkzeugkoordinatensystem (TCS). Die folgende Grafik zeigt eine kugelförmige Werkzeugzone:

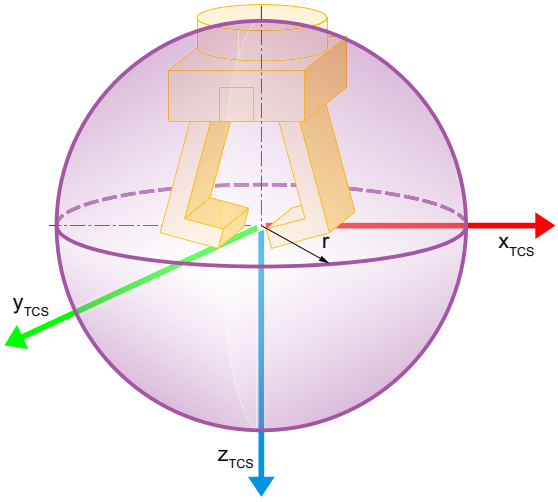

Flanschzone

Flanschzonen definieren Sie im Flanschkoordinatensystem (FCS). Die folgende Grafik zeigt eine zylindrische Flanschzone:

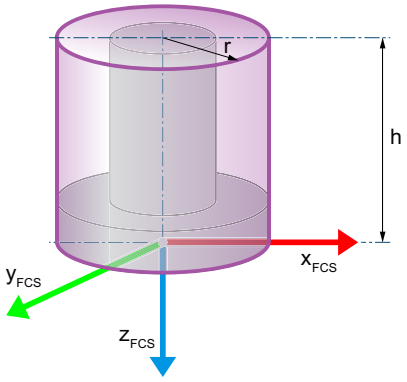

In diesem Beispiel ist eine Verschiebung um die Höhe der Flanschzone in negativer z-Richtung des FCS definiert worden.

Zonengeometrie

Zonen können Sie mit folgenden geometrischen Körpern konfigurieren:

- Kugel
- Quader
- Zylinder

Im Bezugskoordinatensystem geben Sie die Lage des Nullpunkts des Zonenkoordinatensystems vor. Von diesem Nullpunkt ausgehend geben Sie die Dimensionen und die Drehung des Körpers an.

Kugel

Eine Kugel definieren Sie vom Nullpunkt ausgehend über den Radius:

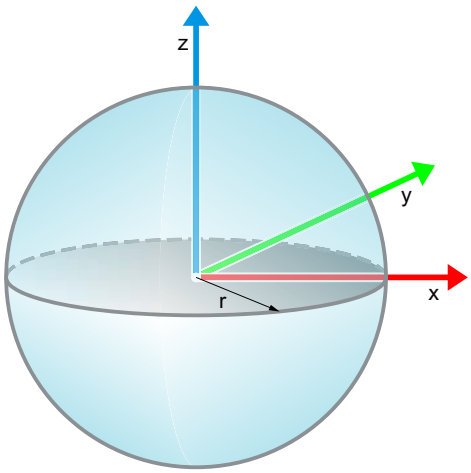

r Radius der Kugel

Quader

Einen Quader definieren Sie vom Nullpunkt ausgehend über die Kantenlängen in x-, y- und z-Richtung:

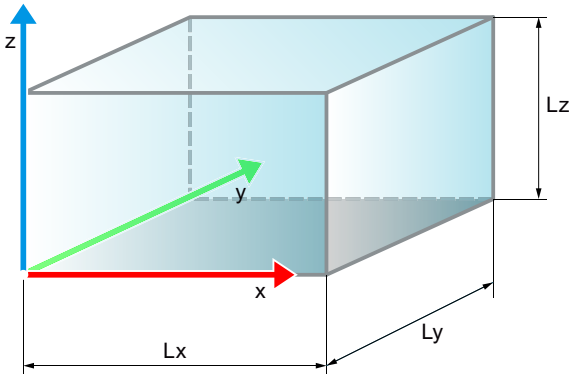

Lx Kantenlänge in x-Richtung des Zonenkoordinatensystems

Ly Kantenlänge in y-Richtung des Zonenkoordinatensystems

Lz Kantenlänge in z-Richtung des Zonenkoordinatensystems

Zylinder

Einen Zylinder definieren Sie vom Nullpunkt ausgehend über den Radius der Grundfläche und die Zylinderhöhe:

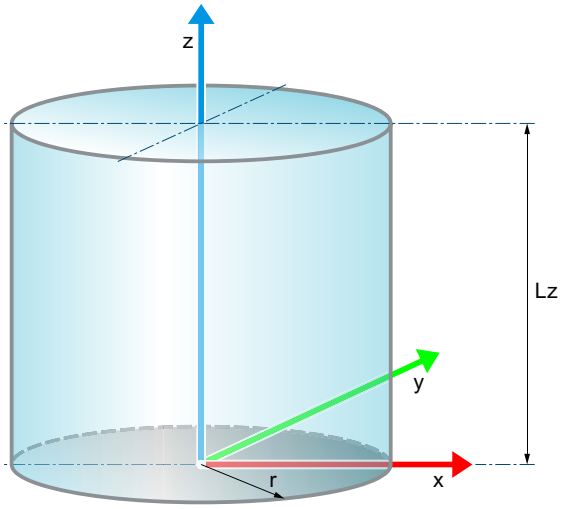

r Radius der Grundfläche des Zylinders

Lz Höhe des Zylinders in z-Richtung des Zonenkoordinatensystems

Variablen Zonenüberwachung

Für die Zonenüberwachung sind folgende Variablen des Technologieobjekts Kinematik relevant:

Variable	Beschreibung
Zonenkonfiguration	
<TO>.WorkspaceZone[1..10]	Konfiguration der Arbeitsraumzonen
<TO>.KinematicsZone[2..10]	Konfiguration der Kinematikzonen Die Zone <TO>.KinematicsZone[1] ist der Werkzeugarbeitspunkt (TCP) und immer aktiviert.
Statuswerte	
<TO>.StatusWorkspaceZone[1..10]	Status der Arbeitsraumzonen
<TO>.StatusKinematicsZone[2..10]	Status der Kinematikzonen
<TO>.StatusZoneMonitoring.WorkingZones	Anzeige von verletzten Arbeitszonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
<TO>.StatusZoneMonitoring.BlockedZones	Anzeige von verletzten Sperrzonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
<TO>.StatusZoneMonitoring.SignalizingZones	Anzeige von angefahrenen Meldezonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
<TO>.StatusZoneMonitoring.KinematicsZones	Anzeige von Kinematikzonen, die Arbeitsraumzonen verletzen Die Bitnummer 1 zeigt den Überwachungsstatus des TCP an. Die Bitnummern 2 bis 10 entsprechen den konfigurierten Zonennummern.

6.1.3.3 Versionsübersicht

Bei S7-1500T Motion Control wird unterschieden zwischen der Version der Technologie, der Technologieobjekte und der Motion Control-Anweisungen.

Die Version eines Technologieobjekts bzw. einer Motion Control-Anweisung wird in den Eigenschaften des Technologieobjekts im Register "Allgemein > Information" im Feld "Version" angezeigt.

Kompatibilitätsliste

Die folgende Tabelle zeigt die Kompatibilität der Version der Technologie mit der Version der CPU:

CPU	Technologie	Technologieobjekt	Motion Control-Anweisung
V2.5	V4.0	Kinematik V4.0	MC_GroupInterrupt V4.0
			MC_GroupContinue V4.0
			MC_GroupStop V4.0
			MC_MoveLinearAbsolute V4.0
			MC_MoveLinearRelative V4.0
			MC_MoveCircularAbsolute V4.0
			MC_MoveCircularAbsolute V4.0
			MC_DefineWorkspaceZone V4.0
			MC_DefineKinematicsZone V4.0
			MC_SetWorkspaceZoneActive V4.0
			MC_SetWorkspaceZoneInactive V4.0
			MC_SetKinematicsZoneActive V4.0
			MC_SetKinematicsZoneInactive V4.0
			MC_DefineTool V4.0
			MC_SetTool V4.0
			MC_SetOcsFrame V4.0

6.1.3.4 Konfigurieren

Technologieobjekt Kinematik hinzufügen

Im Folgenden ist beschrieben, wie Sie ein Technologieobjekt Kinematik in der Projektnavigation hinzufügen.

Voraussetzung

Ein Projekt mit einer CPU S7-1500T ist angelegt.

Vorgehen

Um ein Technologieobjekt Kinematik hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie "TO_Kinematics" aus. Der eingeblendeten Beschreibung können Sie die Funktion des Technologieobjekts entnehmen.
5. Passen Sie im Eingabefeld "Name" den Namen der Kinematik Ihren Bedürfnissen an.
6. Um die vorgeschlagene Datenbausteinnummer zu ändern, wählen Sie die Option "Manuell" aus.
7. Um eigene Informationen zum Technologieobjekt zu ergänzen, klicken Sie auf "Weitere Informationen".
8. Um nach dem Hinzufügen des Technologieobjekts die Konfiguration zu öffnen, aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen".
9. Um das Technologieobjekt hinzuzufügen, klicken Sie auf die Schaltfläche "OK".

Ergebnis

Das neue Technologieobjekt Kinematik wurde erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" angelegt.

Technologieobjekt Kinematik konfigurieren

Konfiguration - Grundparameter

Konfigurieren Sie im Konfigurationsfenster "Grundparameter" die Basiseigenschaften des Technologieobjekts Kinematik.

Kinematikname

Definieren Sie in diesem Feld den Namen der Kinematik. Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet. Die Variablen des Technologieobjekts können im Anwenderprogramm unter diesem Namen verwendet werden.

Kinematiktyp

Wählen Sie in dieser Klappliste den gewünschten Kinematiktyp (Seite 2237) aus.

Maßeinheiten

Wählen Sie in den Klapplisten die gewünschten Maßeinheiten (Seite 2229) für Position, Geschwindigkeit, Winkel und Winkelgeschwindigkeit der Kinematik aus.

Konfiguration - Verschaltungen

Konfigurieren Sie im Konfigurationsfenster "Verschaltungen" die Achsen der Kinematik.

Kinematikachsen

Ein Technologieobjekt Kinematik können Sie mit Positionierachsen und Gleichlaufachsen verschalten, die im Projekt bereits angelegt sind. Wählen Sie abhängig vom Kinematiktyp (Seite 2237) in den Klapplisten die gewünschten Achsen aus. Über die Schaltfläche können Sie direkt die Konfiguration des ausgewählten Technologieobjekts aufrufen. Konfigurieren Sie die verschalteten Technologieobjekte entsprechend des Kinematiktyps als lineare oder rotatorische Achsen.

Abhängig vom Kinematiktyp sind folgende Kinematikachsen relevant:

Kinematiktyp	Kinematikachse A1	Kinematikachse A2	Kinematikachse A3	Orientierungsachse A4
2D	x	x	-	-
2D mit Orientierung	x	x	-	x
3D	x	x	x	-
3D mit Orientierung	x	x	x	x

x Relevant

- Nicht relevant

Siehe auch

Verschaltungsregeln (Seite 2228)

Konfiguration - Geometrie

Konfiguration - Geometrie (Kartesisches Portal)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS) abhängig vom Kinematiktyp:

- Kinematiktyp "Kartesisches Portal 2D" und "Kartesisches Portal 2D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A1 zum Kinematiknullpunkt (KNP) in x-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A2 zum Kinematiknullpunkt in z-Richtung des KCS.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) von der Achse A2 in negativer z-Richtung des KCS.

- Kinematiktyp "Kartesisches Portal 3D" und "Kartesisches Portal 3D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A1 zum Kinematiknullpunkt in x-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A2 zum Kinematiknullpunkt in y-Richtung des KCS.
Länge L3	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A3 zum Kinematiknullpunkt in z-Richtung des KCS.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems von der Achse A3 in negativer z-Richtung des KCS.

Darstellung im Kinematik-Trace

Definieren Sie in diesen Feldern, in welcher Skalierung die Kinematik im Kinematik-Trace angezeigt wird, abhängig vom Kinematiktyp:

- Kinematiktyp "Kartesisches Portal 2D" und "Kartesisches Portal 2D mit Orientierung"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

- Kinematiktyp "Kartesisches Portal 3D" und "Kartesisches Portal 3D mit Orientierung"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
y minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer y-Richtung.
y maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver y-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

Siehe auch

Kartesisches Portal (Seite 2239)

Konfiguration - Geometrie (Rollen-Picker)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS) abhängig vom Kinematiktyp:

- Kinematiktyp "Rollen-Picker 2D" und "Rollen-Picker 2D mit Orientierung"

Feld	Beschreibung
Radius R1	Definieren Sie in diesem Feld den Scheibenradius für die Achse A1.
Radius R2	Definieren Sie in diesem Feld den Scheibenradius für die Achse A2.
Länge L1	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) zum Kinematiknullpunkt (KNP) in x-Richtung des KCS bei Nullstellung der Achsen A1 und A2.
Länge L2	Definieren Sie in diesem Feld den Abstand des FCS zum Kinematiknullpunkt inklusive der Flanschlänge LF in z-Richtung des KCS bei Nullstellung der Achsen A1 und A2.
Flanschlänge LF	Definieren Sie in diesem Feld die Flanschlänge vor dem FCS in negativer z-Richtung des KCS.

- Kinematiktyp "Rollen-Picker 3D (vertikal)", "Rollen-Picker 3D mit Orientierung (vertikal)" und "Rollen-Picker 3D mit Orientierung (horizontal)"

Feld	Beschreibung
Radius R1	Definieren Sie in diesem Feld den Scheibenradius für die Achse A1.
Radius R2	Definieren Sie in diesem Feld den Scheibenradius für die Achse A2.
Länge L1	Definieren Sie in diesem Feld den Abstand des FCS zum Kinematiknullpunkt in x-Richtung des KCS.
Länge L2	Für Kinematiktyp "Rollen-Picker 3D (vertikal)" und "Rollen-Picker 3D mit Orientierung (vertikal)": Definieren Sie in diesem Feld den Abstand der Achse A3 zum Kinematiknullpunkt in y-Richtung des KCS.
	Für Kinematiktyp "Rollen-Picker 3D mit Orientierung (horizontal)": Definieren Sie in diesem Feld den Abstand des FCS zum Kinematiknullpunkt in y-Richtung des KCS.
Länge L3	Für Kinematiktyp "Rollen-Picker 3D (vertikal)" und "Rollen-Picker 3D mit Orientierung (vertikal)": Definieren Sie in diesem Feld den Abstand des FCS zum Kinematiknullpunkt in z-Richtung des KCS.
	Für Kinematiktyp "Rollen-Picker 3D mit Orientierung (horizontal)": Definieren Sie in diesem Feld den Abstand der Achse A3 zum Kinematiknullpunkt in z-Richtung des KCS.
Flanschlänge LF	Definieren Sie in diesem Feld die Flanschlänge vor dem FCS in negativer z-Richtung des KCS.

Darstellung im Kinematik-Trace

Definieren Sie in diesen Feldern, in welcher Skalierung die Kinematik im Kinematik-Trace angezeigt wird, abhängig vom Kinematiktyp:

- Kinematiktyp "Rollen-Picker 2D" und "Rollen-Picker 2D mit Orientierung"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

- Kinematiktyp "Rollen-Picker 3D (vertikal)", "Rollen-Picker 3D mit Orientierung (vertikal)" und "Rollen-Picker 3D mit Orientierung (horizontal)"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
y minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer y-Richtung.
y maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver y-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

Siehe auch

Rollen-Picker (Seite 2250)

Konfiguration - Geometrie (SCARA)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS):

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Achse A1 zum Kinematiknullpunkt (KNP) in z-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Achse A1 zur Achse A2 in x-Richtung des KCS.
Länge L3	Definieren Sie in diesem Feld den Abstand der Achse A2 zur Achse A3 in x-Richtung des KCS.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) von der Achse A3 in negativer z-Richtung des KCS.

Mechanische Achskopplung

Die Kinematiktransformation kompensiert die konfigurierten mechanischen Achskopplungen. Wenn Sie eine mechanische Achskopplung von einer ersten Achse auf eine zweite Achse einstellen, darf für die erste Achse die Einstellung "Modulo" nicht aktiviert sein.

Für die Kinematik können Sie folgende mechanische Achskopplungen konfigurieren:

- Mechanische Kopplung von Achse A1 auf Achse A2
- Mechanische Kopplung von Achse A4 auf Achse A3

Geben Sie in den Feldern "Kompensationsfaktor" jeweils den gewünschten Kopplungsfaktor an.

Darstellung im Kinematik-Trace

Definieren Sie in diesen Feldern, in welcher Skalierung die Kinematik im Kinematik-Trace angezeigt wird:

Feld	Beschreibung
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

Siehe auch

SCARA (Seite 2264)

Konfiguration - Geometrie (Knickarm)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS) abhängig vom Kinematiktyp:

- Kinematiktyp "Knickarm 2D" und "Knickarm 2D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Achse A1 vom Kinematiknullpunkt (KNP) in z-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Achse A1 vom Kinematiknullpunkt in x-Richtung des KCS.
Länge L3	Definieren Sie in diesem Feld den Abstand der Achse A2 von der Achse A1.
Länge L4	Definieren Sie in diesem Feld den Abstand des Zwangskoppelpunkts von der Achse A2.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) vom Zwangskoppelpunkt in negativer z-Richtung des KCS.

- Kinematiktyp "Knickarm 3D" und "Knickarm 3D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Achse A2 vom Kinematiknullpunkt in z-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Achse A2 vom Kinematiknullpunkt in x-Richtung des KCS.
Länge L3	Definieren Sie in diesem Feld den Abstand der Achse A3 von der Achse A2.
Länge L4	Definieren Sie in diesem Feld den Abstand des Zwangskoppelpunkts von der Achse A3.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des FCS vom Zwangskoppelpunkt in negativer z-Richtung des KCS.

Mechanische Achskopplung

Die Kinematiktransformation kompensiert die konfigurierten mechanischen Achskopplungen. Wenn Sie eine mechanische Achskopplung von einer ersten Achse auf eine zweite Achse einstellen, darf für die erste Achse die Einstellung "Modulo" nicht aktiviert sein.

Abhängig vom Kinematiktyp können Sie folgende mechanische Achskopplungen konfigurieren:

- Kinematiktyp "Knickarm 2D" und "Knickarm 2D mit Orientierung":
  Mechanische Kopplung von Achse A1 auf Achse A2
- Kinematiktyp "Knickarm 3D" und "Knickarm 3D mit Orientierung":
  Mechanische Kopplung von Achse A2 auf Achse A3

Geben Sie im Feld "Kompensationsfaktor" den gewünschten Kopplungsfaktor an.

Siehe auch

Knickarm (Seite 2269)

Konfiguration - Geometrie (Delta-Picker)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS) abhängig vom Kinematiktyp:

- Kinematiktyp "Delta-Picker 2D" und "Delta-Picker 2D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld die Länge der Oberarme.
Länge L2	Definieren Sie in diesem Feld die Länge der Verbindungsstangen.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) von der unteren Verbindungsplatte in negativer z-Richtung des KCS.
Abstand D1	Definieren Sie in diesem Feld den Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte).
Abstand D2	Definieren Sie in diesem Feld den Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte).

- Kinematiktyp "Delta-Picker 3D" und "Delta-Picker 3D mit Orientierung"

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld die Länge der Oberarme.
Länge L2	Definieren Sie in diesem Feld die Länge der Verbindungsstangen.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des FCS von der unteren Verbindungsplatte in negativer z-Richtung des KCS.
Abstand D1	Definieren Sie in diesem Feld den Abstand der Achsen zur Mitte der oberen Verbindungsplatte (Radius der oberen Verbindungsplatte).
Abstand D2	Definieren Sie in diesem Feld den Abstand der Gelenkpunkte der Verbindungsstangen zur unteren Verbindungsplatte (Radius der unteren Verbindungsplatte).
Winkel A1 zu A2	Definieren Sie in diesem Feld den Winkel zwischen den Achsen A1 und A2.
Winkel A2 zu A3	Definieren Sie in diesem Feld den Winkel zwischen den Achsen A2 und A3.

Siehe auch

Delta-Picker (Seite 2285)

Konfiguration - Geometrie (Zylindrischer Roboter)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS):

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld den Abstand der Nullposition der Achse A2 vom Kinematiknullpunkt (KNP) in z-Richtung des KCS.
Länge L2	Definieren Sie in diesem Feld den Abstand der Achse A3 vom Kinematiknullpunkt in y-Richtung des KCS.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) von der Achse A3 in negativer z-Richtung des KCS.

Mechanische Achskopplung

Die Kinematiktransformation kompensiert die konfigurierten mechanischen Achskopplungen. Wenn Sie eine mechanische Achskopplung von einer ersten Achse auf eine zweite Achse einstellen, darf für die erste Achse die Einstellung "Modulo" nicht aktiviert sein.

Für den Kinematiktyp "Zylindrischer Roboter 3D mit Orientierung" können Sie die folgende mechanische Achskopplung konfigurieren:

- Mechanische Kopplung von Achse A4 auf Achse A2

Geben Sie im Feld "Kompensationsfaktor" den gewünschten Kopplungsfaktor an.

Darstellung im Kinematik-Trace

Definieren Sie in diesen Feldern, in welcher Skalierung die Kinematik im Kinematik-Trace angezeigt wird:

Feld	Beschreibung
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.
A3 maximal	Definieren Sie in diesem Feld die maximale Verfahrlänge der Achse A3.

Siehe auch

Zylindrischer Roboter (Seite 2298)

Konfiguration - Geometrie (Tripod)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in diesen Feldern die Transformationsparameter der Kinematik im Kinematikkoordinatensystem (KCS):

Feld	Beschreibung
Länge L1	Definieren Sie in diesem Feld die Länge der Verbindungsstangen.
Flanschlänge LF	Definieren Sie in diesem Feld den Abstand des Flanschkoordinatensystems (FCS) von der unteren Verbindungsplatte in negativer z-Richtung des KCS.
Abstand D1	Definieren Sie in diesem Feld den Abstand der oberen Gelenkpunkte der Verbindungsstangen zur Mitte der oberen Verbindungsplatte.
Abstand D2	Definieren Sie in diesem Feld den Abstand der unteren Gelenkpunkte der Verbindungsstangen zur Mitte der unteren Verbindungsplatte.
Winkel zwischen der Achse A1 und der xy-Ebene des KCS	Definieren Sie in diesem Feld den Winkel zwischen der oberen Verbindungsplatte (xy-Ebene des KCS) und der Schiene der Achse A1 ($0.0° \leq \gamma < 90.0°$).
Winkel A1 zu A2	Definieren Sie in diesem Feld den Winkel zwischen den Achsen A1 und A2.
Winkel A2 zu A3	Definieren Sie in diesem Feld den Winkel zwischen den Achsen A2 und A3.

Siehe auch

Tripod (Seite 2307)

Konfiguration - Geometrie (Anwenderdefiniert)

Konfigurieren Sie im Konfigurationsfenster "Geometrie" die geometrischen Parameter der Kinematik.

Transformationsparameter

Definieren Sie in dieser Tabelle die Startwerte der Parameter 1 bis 32 der Kinematik (<TO>.Kinematics.Parameter[1..32]).

Darstellung im Kinematik-Trace

Definieren Sie in diesen Feldern, in welcher Skalierung die Kinematik im Kinematik-Trace angezeigt wird, abhängig vom Kinematiktyp:

- Kinematiktyp "Anwenderdefiniert 2D" und "Anwenderdefiniert 2D mit Orientierung"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

- Kinematiktyp "Anwenderdefiniert 3D" und "Anwenderdefiniert 3D mit Orientierung"

Feld	Beschreibung
x minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer x-Richtung.
x maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver x-Richtung.
y minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer y-Richtung.
y maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver y-Richtung.
z minimal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in negativer z-Richtung.
z maximal	Definieren Sie in diesem Feld die Dimensionierung der Kinematik in positiver z-Richtung.

Siehe auch

Anwendertransformation (Seite 2320)

Anwenderdefinierte Kinematiken (Seite 2314)

Erweiterte Parameter

Konfiguration - Dynamik

Konfigurieren Sie im Konfigurationsfenster "Dynamik" die Voreinstellungswerte für die Dynamik, die Dynamikgrenzen und die Dynamikadaption der Kinematikbewegung und der Orientierungsbewegung.

Voreinstellungen und Grenzen

Um die Voreinstellungswerte für die Kinematikbewegung festzulegen, wählen Sie in der Klappliste "Einstellungen für" den Eintrag "Bahn" aus. Um die Voreinstellungswerte für die Orientierungsbewegung festzulegen, wählen Sie in der Klappliste "Einstellungen für" den Eintrag "Orientierungsbewegung" aus.

Definieren Sie in den Feldern "Geschwindigkeit", "Beschleunigung", "Verzögerung" und "Ruck" die Voreinstellungswerte der Dynamik. Im Anwenderprogramm angestoßene Kinematikbewegungsaufträge werden mit diesen Voreinstellungswerten ausgeführt, wenn für die Aufträge keine gesonderten Dynamikwerte vorgegeben sind.

Definieren Sie in den Feldern "Maximale Geschwindigkeit", "Maximale Beschleunigung", "Maximale Verzögerung" und "Maximaler Ruck" die Voreinstellungswerte der Dynamikgrenzen.

Dynamikadaption

Wählen Sie in der Klappliste die Voreinstellung für die Dynamikadaption aus. Bei aktiver Dynamikadaption wird ein Geschwindigkeitsprofil für die gesamte Bewegung berechnet, das die Dynamikgrenzen der Achsen und der Kinematik berücksichtigt.

Modus	Beschreibung
Nicht begrenzen	Die Dynamikgrenzen der Achsen werden nicht berücksichtigt.
Begrenzen mit Segmentierung der Bahn	Die Bahn wird in Segmente unterteilt. Für jedes dieser Segmente wird die Dynamik so angepasst, dass die Dynamikgrenzen der Achsen nicht überschritten werden.
Begrenzen ohne Segmentierung der Bahn	Die Dynamik wird so angepasst, dass über die gesamte Bahn die Dynamikgrenzen der Achsen nicht überschritten werden.

Siehe auch

Dynamik von Kinematik- und Orientierungsbewegung (Seite 2329)

Konfiguration - Kinematikkoordinatensystem

Konfigurieren Sie im Konfigurationsfenster "Kinematikkoordinatensystem" den KCS-Frame (Seite 2234) und damit die Lage des Kinematikkoordinatensystems (KCS) im Weltkoordinatensystem (WCS).

Kinematiknullpunkt im WCS

Definieren Sie in diesen Feldern die Lage des Kinematikkoordinatensystems:

Feld	Beschreibung
Position x	Definieren Sie in diesem Feld die Verschiebung des KCS in x-Richtung des WCS.
Position y	Definieren Sie in diesem Feld die Verschiebung des KCS in y-Richtung des WCS.
Position z	Definieren Sie in diesem Feld die Verschiebung des KCS in z-Richtung des WCS.
Drehung A	Definieren Sie in diesem Feld die Drehung des KCS um die z-Achse.

Feld	Beschreibung
Drehung B	Definieren Sie in diesem Feld die Drehung des KCS um die y-Achse.
Drehung C	Definieren Sie in diesem Feld die Drehung des KCS um die x-Achse.

Siehe auch

Koordinatensysteme und Frames (Seite 2231)

Konfiguration - Objektkoordinatensysteme

Konfigurieren Sie im Konfigurationsfenster "Objektkoordinatensystem" die OCS-Frames (Seite 2234) und damit die Lage der Objektkoordinatensysteme (OCS) im Weltkoordinatensystem (WCS).

Objektkoordinatensystem (OCS)

Wählen Sie in dieser Klappliste das zu definierende Objektkoordinatensystem aus. Sie können bis zu drei Objektkoordinatensysteme definieren.

OCS im Weltkoordinatensystem (WCS)

Definieren Sie in diesen Feldern die Lage des ausgewählten Objektkoordinatensystems:

Feld	Beschreibung
Position x	Definieren Sie in diesem Feld die Verschiebung des OCS in x-Richtung des WCS.
Position y	Definieren Sie in diesem Feld die Verschiebung des OCS in y-Richtung des WCS.
Position z	Definieren Sie in diesem Feld die Verschiebung des OCS in z-Richtung des WCS.
Drehung A	Definieren Sie in diesem Feld die Drehung des OCS um die z-Achse.
Drehung B	Definieren Sie in diesem Feld die Drehung des OCS um die y-Achse.
Drehung C	Definieren Sie in diesem Feld die Drehung des OCS um die x-Achse.

Siehe auch

Koordinatensysteme und Frames (Seite 2231)

Konfiguration - Werkzeuge

Konfigurieren Sie im Konfigurationsfenster "Werkzeuge" die Tool-Frames (Seite 2234) und damit jeweils die Lage des Werkzeugarbeitspunkts (TCP) der Werkzeuge im Flanschkoordinatensystem (FCS).

Werkzeug

Wählen Sie in dieser Klappliste das zu definierende Werkzeug aus. Sie können bis zu drei Werkzeuge definieren.

Werkzeugarbeitspunkt im FCS

Definieren Sie in diesen Feldern die Lage des Werkzeugarbeitspunkts des ausgewählten Werkzeugs:

Feld	Beschreibung
Position x	Definieren Sie in diesem Feld die Verschiebung des TCP in x-Richtung des FCS.
Position y	Definieren Sie in diesem Feld die Verschiebung des TCP in y-Richtung des FCS.
Position z	Definieren Sie in diesem Feld die Verschiebung des TCP in z-Richtung des FCS.
Drehung A	Definieren Sie in diesem Feld die Drehung des TCP um die z-Achse.

Siehe auch

Koordinatensysteme und Frames (Seite 2231)

Konfiguration - Zonen

Konfigurieren Sie im Konfigurationsfenster "Zonen" die Arbeitsraumzonen und die Kinematikzonen des Technologieobjekts.

Das Konfigurationsfenster ist in folgende Bereiche unterteilt:

- Grafische Ansicht
- Tabellarischer Editor
  - Arbeitsraumzonen
  - Kinematikzonen

Grafische Ansicht

In der grafischen Ansicht werden Ihnen die Arbeitsraumzonen oder die Kinematikzonen angezeigt, die Sie in dem entsprechenden tabellarischen Editor definieren. Mithilfe der Maus können Sie die Ansicht drehen, hinein- oder herauszoomen.

Die Werkzeugleiste am oberen Rand der grafischen Ansicht stellt Ihnen abhängig vom jeweiligen tabellarischen Editor über die Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	An Bildschirmgröße anpassen	Die Ansicht wird an die Fenstergröße angepasst angezeigt.
	Gitternetz ein-/ausblenden	Die Gitternetzlinien des Koordinatensystems werden ein-/ausgeblendet.
WCS	Koordinatensystem auswählen	Wählen Sie ein Koordinatensystem aus.
Werkzeug 1	Werkzeug auswählen	Wählen Sie ein Werkzeug aus.
	2D-Darstellung anzeigen	Die 2D-Darstellung wird angezeigt.
	3D-Darstellung anzeigen	Die 3D-Darstellung wird angezeigt.

Schaltfläche	Funktion	Beschreibung
	xy-Ebene anzeigen	Die xy-Ebene wird angezeigt.
	xy-Ebene gedreht anzeigen	Die xy-Ebene wird um die x-Achse gedreht angezeigt.
	xz-Ebene anzeigen	Die xz-Ebene wird angezeigt.
	xz-Ebene gedreht anzeigen	Die xz-Ebene wird um die z-Achse gedreht angezeigt.
	xz-Ebene gedreht anzeigen	Die xz-Ebene wird um die x-Achse gedreht angezeigt.
	xz-Ebene gedreht anzeigen	Die xz-Ebene wird um die x-Achse und die z-Achse gedreht angezeigt.
	yz-Ebene anzeigen	Die yz-Ebene wird angezeigt.
	yz-Ebene gedreht anzeigen	Die yz-Ebene wird um die z-Achse gedreht angezeigt.
	yz-Ebene gedreht anzeigen	Die yz-Ebene wird um die y-Achse gedreht angezeigt.
	yz-Ebene gedreht anzeigen	Die yz-Ebene wird um die y-Achse und die z-Achse gedreht angezeigt.

Arbeitsraumzonen

Arbeitsraumzonen (Seite 2335) beschreiben die Umgebung einer Kinematik. Sie können in der Tabelle bis zu zehn Arbeitsraumzonen konfigurieren:

Spalte	Beschreibung	
Sichtbar	Mit dem Symbol in dieser Spalte blenden Sie die Zone in der oberen Ansicht ein und aus.	
Nummer	In dieser Spalte wird die Zonennummer angezeigt.	
Status	Wählen Sie in dieser Spalte den Aktivierungsstatus der Zone aus.	
	Aktiv	Die Zonenüberwachung ist für die Zone aktiviert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_SetWorkspaceZoneInactive"-Auftrag deaktivieren.
	Inaktiv	Die Zonenüberwachung ist für die Zone deaktiviert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_SetWorkspaceZoneActive"-Auftrag aktivieren.
	Ungültig	Die Zone ist nicht definiert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_DefineWorkspaceZone"-Auftrag definieren.

Spalte	Beschreibung	
Zonentyp	Wählen Sie in dieser Spalte den Typ der Zone aus.	
	Arbeitszone	Arbeitszonen definieren Bereiche, in denen sich Kinematikzonen bewegen dürfen. Sie können mehrere Arbeitszonen vorgeben. Zu einem Zeitpunkt darf aber nur eine Arbeitszone aktiviert sein. Wenn keine Arbeitszone aktiviert ist, gilt der gesamte Verfahrraum der Kinematik als Arbeitsbereich.
	Sperrzone	Sperrzonen definieren Bereiche, in die eine Kinematikzone nicht eintreten darf.
	Meldezone	Meldezonen sind Bereiche innerhalb des Verfahrraums der Kinematik. Meldezonen zeigen eine Zonenverletzung durch eine Kinematikzone an, lösen aber keinen Stopp der Kinematikbewegung aus.
Geometrie	Wählen Sie in dieser Spalte die Geometrie der Zone (Seite 2337) aus.	
		Kugel
		Quader
		Zylinder
Länge x	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Länge der Zone in x-Richtung.	
Länge y	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Breite der Zone in y-Richtung.	
Länge z	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Höhe der Zone in z-Richtung.	
	Bei einer zylindrischen Zone definieren Sie in dieser Spalte die Höhe der Zone in z-Richtung.	
Radius	Bei einer kugelförmigen Zone definieren Sie in dieser Spalte den Radius der Zone.	
	Bei einer zylindrischen Zone definieren Sie in dieser Spalte den Radius der Zone.	
KS	Wählen Sie in dieser Spalte das Bezugskoordinatensystem aus.	
	WCS	Weltkoordinatensystem
	OCS 1	Objektkoordinatensystem 1
	OCS 2	Objektkoordinatensystem 2
	OCS 3	Objektkoordinatensystem 3
x	Definieren Sie in dieser Spalte die Position der Zone in x-Richtung.	
y	Definieren Sie in dieser Spalte die Position der Zone in y-Richtung.	
z	Definieren Sie in dieser Spalte die Position der Zone in z-Richtung.	
A	Definieren Sie in dieser Spalte die Drehung der Zone um die z-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -180.0° und 179.999° an.	
B	Definieren Sie in dieser Spalte die Drehung der Zone um die y-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -90.0° und 90.0° an.	
C	Definieren Sie in dieser Spalte die Drehung der Zone um die x-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -180.0° und 179.999° an.	

Kinematikzonen

Kinematikzonen (Seite 2336) beziehen sich auf den Arbeitspunkt/Flansch einer Kinematik und bewegen sich mit diesem. Die Zonenüberwachung prüft Kinematikzonen auf Durchdringung mit Arbeitsraumzonen. Sie können in der Tabelle bis zu neun Kinematikzonen konfigurieren:

Spalte	Beschreibung	
Sichtbar	Mit dem Symbol in dieser Spalte blenden Sie die Zone in der oberen Ansicht ein und aus.	
Nummer	In dieser Spalte wird die Zonennummer angezeigt.	
Status	Wählen Sie in dieser Spalte den Aktivierungsstatus der Zone aus.	
	Aktiv	Die Zonenüberwachung ist für die Zone aktiviert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_SetKinematicsZoneInactive"-Auftrag deaktivieren.
	Inaktiv	Die Zonenüberwachung ist für die Zone deaktiviert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_SetKinematicsZoneActive"-Auftrag aktivieren.
	Ungültig	Die Zone ist nicht definiert. Sie können die Zone in Ihrem Anwenderprogramm über einen "MC_DefineKinematicsZone"-Auftrag definieren.
Zonentyp	Wählen Sie in dieser Spalte den Typ der Zone aus.	
	Flanschzone	Flanschzonen umhüllen den Flansch oder Teile davon.
	Werkzeugzone	Werkzeugzonen umhüllen das Werkzeug oder Teile davon.
Geometrie	Wählen Sie in dieser Spalte die Geometrie der Zone (Seite 2337) aus.	
		Kugel
		Quader
		Zylinder
Länge x	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Länge der Zone in x-Richtung.	
Länge y	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Breite der Zone in y-Richtung.	
Länge z	Bei einer quaderförmigen Zone definieren Sie in dieser Spalte die Höhe der Zone in z-Richtung.	
	Bei einer zylindrischen Zone definieren Sie in dieser Spalte die Höhe der Zone in z-Richtung.	
Radius	Bei einer kugelförmigen Zone definieren Sie in dieser Spalte den Radius der Zone.	
	Bei einer zylindrischen Zone definieren Sie in dieser Spalte den Radius der Zone.	
KS	Wählen Sie in dieser Spalte das Bezugskoordinatensystem aus.	
	FCS	Flanschkoordinatensystem
	TCS	Werkzeugkoordinatensystem
x	Definieren Sie in dieser Spalte die Position der Zone in x-Richtung.	
y	Definieren Sie in dieser Spalte die Position der Zone in y-Richtung.	
z	Definieren Sie in dieser Spalte die Position der Zone in z-Richtung.	
A	Definieren Sie in dieser Spalte die Drehung der Zone um die z-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -180.0° und 179.999° an.	

Spalte	Beschreibung
B	Definieren Sie in dieser Spalte die Drehung der Zone um die y-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -90.0° und 90.0° an.
C	Definieren Sie in dieser Spalte die Drehung der Zone um die x-Achse (für eine kugelförmige Zone nicht relevant). Geben Sie einen Wert zwischen -180.0° und 179.999° an.

Online-Sicht

Wenn Sie auf das Symbol klicken, werden die aus der CPU zurückgelesenen Werte der Arbeitsraumzonen und der Kinematikzonen zum aktuellen Zeitpunkt angezeigt. Sie können in der Tabelle für jede Zone zwei weitere Zeilen über das Symbol in der Spalte "Nummer" einblenden. Diese drei Zeilen enthalten folgende Werte:

Zeile	Kennzeichnung in der Spalte "Nummer"	Beschreibung
1	Nummer der Zone	Diese Zeile enthält die aktuellen Werte der Zone.
2		Diese Zeile enthält die Startwerte der CPU.
3		Diese Zeile enthält die Startwerte im Projekt. Sie können die Werte bearbeiten.

Siehe auch

Zonenüberwachung (Seite 2332)

Technologieobjekt Kinematik kopieren

Im Folgenden ist beschrieben, wie Sie ein Technologieobjekt Kinematik in der Projektnavigation kopieren.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500T ist angelegt.
- Im Projekt ist ein Technologieobjekt Kinematik angelegt.

Vorgehen

Um ein Technologieobjekt Kinematik zu kopieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Markieren Sie das zu kopierende Technologieobjekt Kinematik.
4. Um die verbundenen Achsen mitzukopieren, markieren Sie diese ebenfalls. Zur Mehrfachauswahl halten Sie die Taste <Strg> gedrückt.
5. Wählen Sie im Kontextmenü den Befehl "Kopieren".

6. Markieren Sie den Ordner "Technologieobjekte".
7. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Ergebnis

Das ausgewählte Technologieobjekt Kinematik wurde zusammen mit den gegebenenfalls ausgewählten verbundenen Achsen kopiert und in der Projektnavigation im Ordner "Technologieobjekte" angelegt.

Technologieobjekt Kinematik löschen

Im Folgenden ist beschrieben, wie Sie ein Technologieobjekt Kinematik in der Projektnavigation löschen.

Voraussetzung

- Ein Projekt mit einer CPU S7-1500T ist angelegt.
- Im Projekt ist ein Technologieobjekt Kinematik angelegt.

Vorgehen

Um ein Technologieobjekt Kinematik zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Markieren Sie das zu löschende Technologieobjekt Kinematik.
4. Wählen Sie im Kontextmenü den Befehl "Löschen".
   Der Dialog "Löschen bestätigen" wird geöffnet.
5. Um das Technologieobjekt zu löschen, klicken Sie auf die Schaltfläche "Ja".

Ergebnis

Das ausgewählte Technologieobjekt Kinematik wurde gelöscht. Die mit dem Technologieobjekt Kinematik verbundenen Achsen bleiben erhalten.

Funktionsleiste der Konfiguration

In der Funktionsleiste der Funktionssicht sind folgende Funktionen verfügbar:

Symbol	Funktion	Erläuterung
	Online-Werte anzeigen	Zeigt die aus der CPU zurückgelesenen Werte zum aktuellen Zeitpunkt an.
	Koppelt die Funktions- und Parametersicht für die in der Navigation markierten Objekte	Ermöglicht den gezielten Wechsel von Parametersicht zur funktionsorientierten Sicht.

Symbol	Funktion	Erläuterung
	Ein-/Ausklappen aller Knoten und Objekte	Klappt in der jeweils aktiven Sicht alle Knoten und Objekte der Navigation bzw. der Datenstruktur ein und aus.
	Ein-/Ausklappen der Knoten unterhalb des markierten Knotens	Klappt in der jeweils aktiven Sicht die markierten Knoten und Objekte der Navigation bzw. der Datensicht ein und aus.

6.1.3.5 Programmieren

Das Kapitel "Programmieren" beinhaltet allgemeine Informationen zum Versorgen und Auswerten der Motion Control-Anweisungen.

Eine Übersicht der Motion Control-Anweisungen für das Technologieobjekt Kinematik finden Sie im Kapitel "Funktionen" (Seite 2227).

Über die Motion Control-Anweisungen im Anwenderprogramm können Sie Aufträge an das Technologieobjekt absetzen. Mit den Eingangsparametern der Motion Control-Anweisungen definieren Sie den Auftrag. Der aktuelle Auftragsstatus wird an den Ausgangsparametern angezeigt.

Da das Technologieobjekt Kinematik die Kinematikachsen gruppiert, können Sie dem Eingangsparameter "AxesGroup" direkt das Technologieobjekt Kinematik zuweisen.

Das Technologieobjekt Kinematik selbst können Sie nicht über einen "MC_Power"-Auftrag freigeben oder über einen "MC_Home"-Auftrag referenzieren. Für Kinematikbewegungen müssen stattdessen die verschalteten Achsen freigegeben ("MC_Power.Enable" = TRUE) bzw. referenziert sein.

Fehler des Technologieobjekts Kinematik können Sie mit einem "MC_Reset"-Auftrag quittieren oder einen Restart des Technologieobjekts durchführen.

Auftragskette

In die Auftragskette des Technologieobjekts Kinematik werden bewegungsrelevante Aufträge eingetragen.

Die folgenden Aufträge reihen sich in die Anweisungskette ein:

Auftrag	Kurzbeschreibung
Kinematikbewegungen	
"MC_MoveLinearAbsolute"	Kinematik mit linearer Bahnbewegung positionieren
"MC_MoveLinearRelative"	Kinematik mit linearer Bahnbewegung relativ positionieren
"MC_MoveCircularAbsolute"	Kinematik mit zirkularer Bahnbewegung positionieren
"MC_MoveCircularRelative"	Kinematik mit zirkularer Bahnbewegung relativ positionieren
Zonen	
"MC_DefineWorkspaceZone"	Arbeitsraumzone definieren
"MC_DefineKinematicsZone"	Kinematikzone definieren
"MC_SetWorkspaceZoneActive"	Arbeitsraumzone aktivieren
"MC_SetWorkspaceZoneInactive"	Arbeitsraumzone deaktivieren
"MC_SetKinematicsZoneActive"	Kinematikzone aktivieren

Auftrag	Kurzbeschreibung
"MC_SetKinematicsZoneInactive"	Kinematikzone deaktivieren
Koordinatensysteme	
"MC_SetOcsFrame"	Objektkoordinatensysteme neu definieren

Die Aufträge werden in derselben Reihenfolge abgearbeitet, in der diese in die Auftragskette eingetragen wurden. Die Reihenfolge der Aufträge kann nachträglich nicht verändert werden. Wenn ein weiterer Bewegungsauftrag der Auftragskette hinzugefügt wird, werden alle Aufträge in der Auftragskette neu berechnet. Die Kinematikbewegungsaufträge brechen sich nicht gegenseitig ab. Da für die Bewegungsaufbereitung und für die Berechnung des Geschwindigkeitsprofils alle Aufträge in der Auftragskette berücksichtigt werden, können auch Aufträge mit kurzen Verfahrlängen und Überschleifen mit höheren Geschwindigkeiten verfahren werden, als sie für die Einzelbewegungen möglich sind. Bei der Neuberechnung wird auch der laufende Auftrag mit einbezogen, sodass der laufende Auftrag mit dem nächsten Auftrag überschliffen werden kann.

Sie können auch die Abarbeitung der Aufträge durch einen "MC_GroupInterrupt"-Auftrag unterbrechen, die Auftragskette befüllen und anschließend die Abarbeitung mit einem "MC_GroupContinue"-Auftrag fortsetzen.

Standardmäßig kann die Auftragskette bis zu fünf Aufträge enthalten. Sie können die maximale Anzahl der Aufträge (Seite 2394) über die Parametersicht ändern. Die Auftragskette kann maximal zehn Aufträge enthalten.

Siehe auch

Variable "MotionQueue" (Kinematik) (Seite 2394)

Unterbrechen, Fortsetzen und Stoppen von Kinematikbewegungen (Seite 2363)

Bewegungsstatus und Restweg

Den Status und den Restweg eines Bewegungsauftrags können Sie den Parametern der entsprechenden Motion Control-Anweisung entnehmen.

Status eines Bewegungsauftrags

Den Status eines Bewegungsauftrags können Sie an den Parametern "Busy" und "Active" erkennen. Mit dem Absetzen des Auftrags wird der Parameter "Busy" auf TRUE gesetzt und der Auftrag der Auftragskette hinzugefügt. Solange sich der Auftrag noch in der Auftragskette befindet, ist der Parameter "Active" auf FALSE gesetzt. Sobald der Auftrag in der Bewegungsführung wirksam ist, wird der Parameter "Active" auf TRUE gesetzt. Wenn der Bewegungsauftrag abgeschlossen ist, werden die Parameter "Busy" und "Active" auf FALSE und der Parameter "Done" auf TRUE gesetzt.

Alle inaktiven Aufträge in der Auftragskette werden neu berechnet, wenn ein weiterer Bewegungsauftrag der Auftragskette hinzugefügt wird. Bei der Neuberechnung wird auch der laufende Auftrag mit einbezogen, sodass der laufende Auftrag mit dem nächsten Auftrag überschliffen werden kann. Bei einer unterbrochenen Bewegungsführung durch einen "MC_GroupInterrupt"-Auftrag werden die Aufträge in der Auftragskette erst berechnet, wenn die Bewegungsführung mit einem "MC_GroupContinue"-Auftrag fortgesetzt wird.

Restweg eines Bewegungsauftrags

Den Restweg eines Bewegungsauftrags können Sie dem Parameter "RemainingDistance" entnehmen. Wenn die Bewegung nicht überschliffen wird, enthält der Parameter "RemainingDistance" den Weg auf der Bahn bis zur Zielposition. Wenn die laufende Bewegung mit der nächsten Bewegung überschliffen wird, enthält der Parameter "RemainingDistance" den Weg auf der Bahn bis zum Beginn des Überschleifsegments. Wenn bei einem Bewegungsauftrag ausschließlich die Orientierungsachse bewegt wird (Umorientierung), enthält der Parameter "RemainingDistance" den Wert "-1.0".

Unterbrechen, Fortsetzen und Stoppen von Kinematikbewegungen

Laufende Kinematikbewegungen können Sie unterbrechen und fortsetzen oder stoppen und somit auch wartende Bewegungsaufträge abbrechen.

Unterbrechen von Kinematikbewegungen

Mit der Motion Control-Anweisung "MC_GroupInterrupt" unterbrechen Sie die Bewegungsausführung am Technologieobjekt Kinematik. Über den Parameter "Mode" bestimmen Sie das dynamische Verhalten. Die Kinematik kann entweder mit der Dynamik des zu unterbrechenden Bewegungsauftrags oder mit maximaler Dynamik angehalten werden. Beim Anhalten wird die aktuelle Bahn nicht verlassen. Wenn sich die Kinematik bereits im Stillstand befindet, ist die Bewegungsführung ebenso für folgende Bewegungsaufträge unterbrochen.

Das Technologieobjekt Kinematik befindet sich im Zustand "Interrupted" (<TO>.StatusWord.X17).

Für eine Bahnplanung können Sie die Abarbeitung der Aufträge unterbrechen, die Auftragskette (Seite 2361) befüllen und anschließend die Abarbeitung der Aufträge fortsetzen.

Fortsetzen von Kinematikbewegungen

Mit der Motion Control-Anweisung "MC_GroupContinue" setzen Sie eine Kinematikbewegung fort, die mit einem "MC_GroupInterrupt"-Auftrag unterbrochen wurde. Die Kinematikbewegung lässt sich auch fortsetzen, wenn die Kinematik durch den "MC_GroupInterrupt"-Auftrag den Stillstand noch nicht erreicht hat.

Der "MC_GroupContinue"-Auftrag hat nur eine Wirkung, wenn sich das Technologieobjekt Kinematik im Zustand "Interrupted" befindet (<TO>.StatusWord.X17).

Stoppen von Kinematikbewegungen

Mit der Motion Control-Anweisung "MC_GroupStop" stoppen Sie die Bewegungsführung am Technologieobjekt Kinematik. Dabei werden sowohl der aktive Bewegungsauftrag als auch alle anstehenden Aufträge in der Auftragskette abgebrochen und die Auftragskette geleert. Wenn die Kinematikbewegung bereits mit einem "MC_GroupInterrupt"-Auftrag unterbrochen wurde, wird diese ebenfalls abgebrochen. Solange der Parameter "Execute" auf TRUE gesetzt ist, werden auch folgende Kinematikaufträge abgelehnt ("ErrorID" = 16#80CD).

Über den Parameter "Mode" bestimmen Sie das dynamische Verhalten. Die Kinematik kann entweder mit der Dynamik des zu stoppenden Bewegungsauftrags oder mit maximaler Dynamik angehalten werden. Beim Anhalten wird die aktuelle Bahn nicht verlassen.

Bewegungsaufbereitung über mehrere Aufträge

Verbinden von mehreren Kinematikbewegungen mit geometrischen Übergängen

Mehrere Bewegungen können einander angehängt werden, wobei die Kinematik zwischen den einzelnen Bewegungen zum Stillstand kommt. Um eine durchgängige Bewegungsführung zu erreichen, können die einzelnen Bewegungen mit geometrischen Übergängen überschliffen werden. Die entsprechenden Parameter definieren Sie an dem neuen Bewegungsauftrag (A2), in den der vorherige Auftrag (A1) überschliffen werden soll.

Übergänge von Linearbewegungen

Mit den Motion Control-Anweisungen "MC_MoveLinearAbsolute" und "MC_MoveLinearRelative" verfahren Sie eine Kinematik mit einer linearen Bewegung. Mit dem Parameter "BufferMode" bestimmen Sie die Art des Bewegungsübergangs und mit dem Parameter "TransitionParameter[1]" den Überschleifabstand. Die folgende Tabelle zeigt anhand von zwei Linearbewegungen, wie sich diese Parameter auf den Bewegungsübergang auswirken:

Überschleifabstand ("TransitionParameter[1]")	Bewegungsübergang ("BufferMode")	Beschreibung
Nicht relevant	"BufferMode" = 1 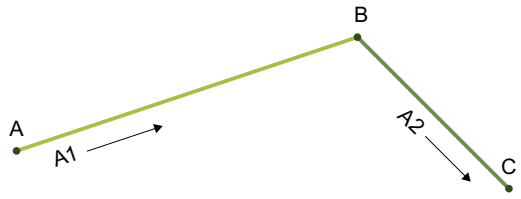 	Bewegung anhängen Die laufende Linearbewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die nächste Linearbewegung ausgeführt.
d > 0.0	"BufferMode" = 2, 5 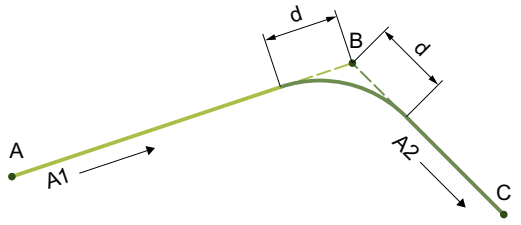 	Bewegung überschleifen Mit dem Erreichen des Überschleifabstands zur Zielposition wird die laufende Linearbewegung mit der nächsten Linearbewegung überschliffen. Bei "BufferMode" = 2 wird mit der niedrigeren und bei "BufferMode" = 5 mit der höheren Geschwindigkeit beider Bewegungsaufträge überschliffen.

Überschleifabstand ("TransitionParameter[1]")	Bewegungsübergang ("BufferMode")	Beschreibung
d = 0.0	"BufferMode" = 2, 5 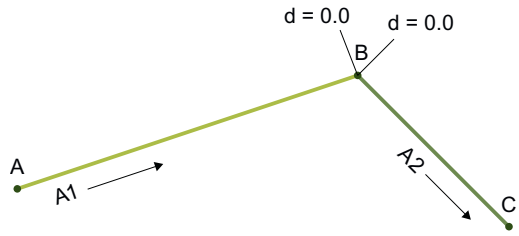	Bewegung überschleifen Da der Überschleifabstand 0.0 ist, wird wie bei "BufferMode" = 1 verfahren. Die laufende Linearbewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die nächste Linearbewegung ausgeführt.
d < 0.0	"BufferMode" = 2, 5 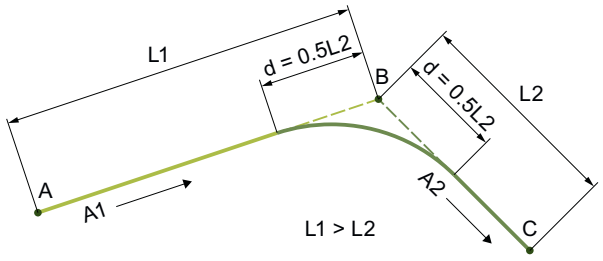	Bewegung überschleifen Da der Überschleifabstand negativ ist, wird der maximale Überschleifabstand verwendet. Mit dem Erreichen des Überschleifabstands zur Zielposition wird die laufende Linearbewegung mit der nächsten Linearbewegung überschliffen. Bei "BufferMode" = 2 wird mit der niedrigeren und bei "BufferMode" = 5 mit der höheren Geschwindigkeit beider Bewegungsaufträge überschliffen.

Übergänge von Kreisbewegungen

Mit den Motion Control-Anweisungen "MC_MoveCircularAbsolute" und "MC_MoveCircularRelative" verfahren Sie eine Kinematik mit einer Kreisbewegung. Mit dem Parameter "BufferMode" bestimmen Sie die Art des Bewegungsübergangs und mit dem Parameter "TransitionParameter[1]" den Überschleifabstand. Die folgende Tabelle zeigt anhand von einer Linear- und einer Kreisbewegung, wie sich diese Parameter auf den Bewegungsübergang auswirken:

Überschleifabstand ("TransitionParameter[1]")	Bewegungsübergang ("BufferMode")	Beschreibung
Nicht relevant	"BufferMode" = 1 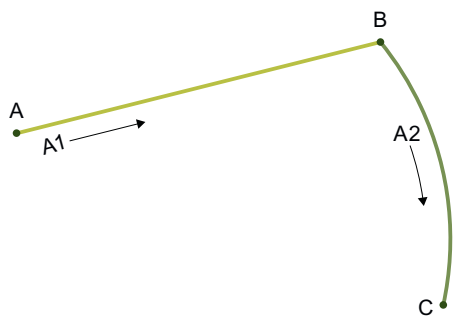 	Bewegung anhängen Die laufende Linearbewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Kreisbewegung ausgeführt.
d > 0.0	"BufferMode" = 2, 5 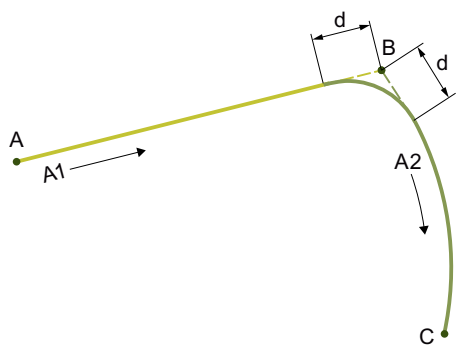 	Bewegung überschleifen Mit dem Erreichen des Überschleifabstands zur Zielposition wird die laufende Linearbewegung mit der Kreisbewegung überschliffen. Bei "BufferMode" = 2 wird mit der niedrigeren und bei "BufferMode" = 5 mit der höheren Geschwindigkeit beider Bewegungsaufträge überschliffen.

Überschleifabstand ("TransitionParameter[1]")	Bewegungsübergang ("BufferMode")	Beschreibung
d = 0.0	"BufferMode" = 2, 5 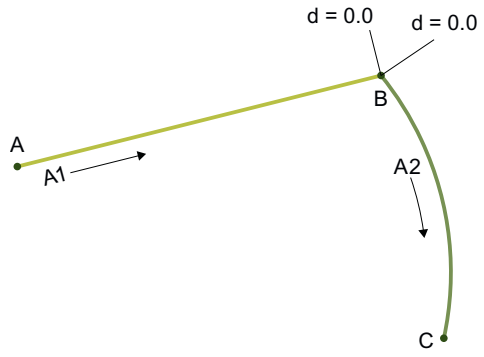	Bewegung überschleifen Da der Überschleifabstand 0.0 ist, wird wie bei "BufferMode" = 1 verfahren. Die laufende Linearbewegung wird zu Ende geführt und die Kinematik kommt zum Stillstand. Anschließend wird die Kreisbewegung ausgeführt.
d < 0.0	"BufferMode" = 2, 5 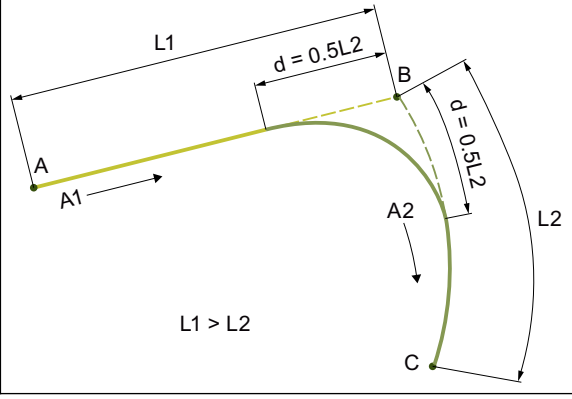	Bewegung überschleifen Da der Überschleifabstand negativ ist, wird der maximale Überschleifabstand verwendet. Mit dem Erreichen des Überschleifabstands zur Zielposition wird die laufende Linearbewegung mit der Kreisbewegung überschliffen. Bei "BufferMode" = 2 wird mit der niedrigeren und bei "BufferMode" = 5 mit der höheren Geschwindigkeit beider Bewegungsaufträge überschliffen.

Maximaler Überschleifabstand

Der maximale Überschleifabstand wird verwendet, wenn der Wert des Parameters "TransitionParameter[1]" < 0.0 ist. Der maximale Überschleifabstand berechnet sich aus der Hälfte der kürzeren Strecke beider Bewegungen.

Voraussetzung	Maximaler Überschleifabstand
L1 > L2	$d_{max} = ½ \cdot L2$
L1 < L2	$d_{max} = ½ \cdot L1$

L1 Streckenlänge des ersten Auftrags

L2 Streckenlänge des zweiten Auftrags

Dynamisches Verhalten beim Anhängen/Überschleifen von Bewegungen

Mit den Parametern "BufferMode" und "DynamicAdaption" bestimmen Sie das dynamische Verhalten beim Übergang von Kinematikbewegungen.

Sie können mehrere Bewegungen einander anhängen, wobei die Kinematik zwischen den einzelnen Bewegungen zum Stillstand kommt ("BufferMode" = 1). Um eine durchgängige Bewegung zu erreichen, können die einzelnen Bewegungen mit einem Überschleifsegment

überschliffen werden. Dabei kann mit der niedrigeren Geschwindigkeit der aufeinander folgenden Bewegungen ("BufferMode" = 2) oder mit der höheren Geschwindigkeit ("BufferMode" = 5) überschliffen werden.

Dynamikadaption

Bei aktiver Dynamikadaption mit Segmentierung wird die Bahn inklusive des Überschleifsegments in weitere Segmente unterteilt ("DynamicAdaption" = 1). Für jedes dieser Segmente wird das Geschwindigkeitsprofil unter der Berücksichtigung der Dynamikgrenzen der Achsen, die für einzelne Abschnitte der Bewegung gelten, berechnet. Die Dynamik wird somit für einzelne Abschnitte einer Bewegung angepasst.

Bei aktiver Dynamikadaption ohne Segmentierung der Bahn wird das Geschwindigkeitsprofil unter der Berücksichtigung der Dynamikgrenzen der Achsen, die für die gesamte Bewegung gelten, berechnet ("DynamicAdaption" = 2).

In die Dynamikadaption werden Geschwindigkeit und Beschleunigung einbezogen. Bei der Beschleunigung wird die Tangential- und Radialbeschleunigung der Bahn berücksichtigt. Der Ruck wird bei der Dynamikadaption nicht begrenzt.

Wenn die Dynamikadaption deaktiviert ist, werden die Dynamikgrenzen der Achsen nicht berücksichtigt ("DynamicAdaption" = 0).

Zusammenwirken von Kinematikbewegungen und Einzelachsbewegungen

Kinematikbewegungen sind nur möglich, wenn an den Kinematikachsen keine Einzelachsbewegungen aktiv sind. Einzelachsbewegungen wirken ablösend auf Kinematikbewegungen. Die Bewegung der entsprechenden Achse wird von der Einzelachsbewegung abgelöst und die Auftragskette geleert. Die anderen Kinematikachsen stoppen mit maximaler Dynamik.

Folgende Funktionen sind während einer laufenden Kinematikbewegung zulässig:

- Momentenreduzierung an den Achsen/Fahren auf Festanschlag ("MC_TorqueLimiting") Beim Erreichen des Festanschlags wird die Kinematikbewegung abgebrochen.
- Vorgabe eines additiven Moments ("MC_TorqueAdditive")
- Vorgabe der oberen und unteren Momentengrenze ("MC_TorqueRange")
- Geberumschaltung ("MC_SetSensor")

Folgende Funktionen werden während einer laufenden Kinematikbewegung abgelehnt:

- Überlagerte Bewegung an den Achsen ("MC_MoveSuperimposed")
- Referenzieren der Achsen ("MC_Home")

6.1.3.6 Inbetriebnahme

Funktion und Aufbau der Kinematiksteuertafel

Mit der Kinematiksteuertafel übernehmen Sie die Steuerungshoheit für ein Technologieobjekt Kinematik und steuern die Bewegungen der Kinematik oder der einzelnen Achsen.

⚠ WARNUNG

Unkontrollierte Achsbewegungen

Beim Betrieb mit der Kinematiksteuertafel kann die Kinematik unkontrollierte Bewegungen ausführen (z. B. wegen fehlerhafter Konfiguration des Antriebs oder des Technologieobjekts). Weiterhin wird beim Verfahren einer Leitachse mit der Kinematiksteuertafel eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren.

Führen Sie daher vor dem Betrieb mit der Kinematiksteuertafel folgende Schutzmaßnahmen durch:

- Stellen Sie sicher, dass sich der NOT-AUS-Schalter in Reichweite des Bedieners befindet.
- Aktivieren Sie die Hardware-Endschalter.
- Aktivieren Sie die Software-Endschalter.
- Stellen Sie sicher, dass die Schleppfehlerüberwachung aktiviert ist.
- Stellen Sie sicher, dass keine Folgeachse mit der zu verfahrenden Achse gekoppelt ist.

In der Projektnavigation finden Sie die Kinematiksteuertafel des Technologieobjekts Kinematik unter "Technologieobjekt > Inbetriebnahme".

Die Kinematiksteuertafel ist in folgende Bereiche unterteilt:

- Steuerungshoheit
- Kinematik
- Betriebsart
- Steuern
- Status
- Aktuelle Positionswerte

Elemente der Kinematiksteuertafel

Die folgende Tabelle zeigt die Elemente der Kinematiksteuertafel:

Bereich	Element	Beschreibung
Steuerungshoheit		Im Bereich "Steuerungshoheit" übernehmen Sie die Steuerungshoheit für das Technologieobjekt oder geben Sie an Ihr Anwenderprogramm zurück.
	Schaltfläche "Holen"	Mit der Schaltfläche "Holen" stellen Sie eine Onlineverbindung zur CPU her und übernehmen die Steuerungshoheit für das ausgewählte Technologieobjekt. • Zur Übernahme der Steuerungshoheit muss das Technologieobjekt im Anwenderprogramm gesperrt sein. • Mit der Übernahme der Steuerungshoheit der Kinematik wird automatisch die Steuerungshoheit über alle mit der Kinematik verschalteten Achsen übernommen. Die Steuerungshoheit kann nur mit der Kinematiksteuertafel übernommen werden, wenn keine Achssteuertafel der verschalteten Achsen aktiv ist. • Beim Verfahren einer Leitachse mit der Kinematiksteuertafel wird eine gegebenenfalls aufsynchronisierte Folgeachse mitverfahren. • Wenn Sie auf die Schaltfläche "Holen" klicken, wird ein Warnhinweis angezeigt. Im Warnhinweis können Sie die Lebenszeichenüberwachung anpassen (100 bis 60000 ms). Wenn die Steuerungshoheit der Kinematiksteuertafel wiederholt ohne direkte Fehlermeldung verloren geht, kann die Onlineverbindung zur CPU durch eine zu hohe Kommunikationslast beeinträchtigt sein. In diesem Fall wird im Archiv der Meldungsanzeige die Meldung "Fehler bei Inbetriebnahme. Ausfall des Lebenszeichens zwischen Steuerung und TIA Portal" angezeigt. Um diesen Fehler zu beheben, passen Sie die Lebenszeichenüberwachung im Warnhinweis an. • Das Anwenderprogramm hat bis zur Rückgabe der Steuerungshoheit keinen Einfluss auf die Funktionen des Technologieobjekts. Motion Control-Aufträge vom Anwenderprogramm an das Technologieobjekt werden mit Fehler ("ErrorID" = 16#8012: Kinematiksteuertafel aktiviert) abgewiesen. • Mit der Übernahme der Steuerungshoheit wird die Konfiguration des Technologieobjekts übernommen. Änderungen an der Konfiguration des Technologieobjekts werden erst wirksam, nachdem die Steuerungshoheit zurückgegeben wurde. Nehmen Sie daher gegebenenfalls erforderliche Änderungen vor der Übernahme der Steuerungshoheit vor. • Wenn die Steuerungshoheit für das Technologieobjekt übernommen wurde, sind die Kinematiksteuertafel und die Achssteuertafeln der verschalteten Achsen für den Zugriff durch eine andere Instanz des TIA Portals gesperrt (Team Engineering ab CPU V1.5). • Wenn die Onlineverbindung zur CPU während des Betriebs mit der Kinematiksteuertafel ausfällt, wird die Kinematik bzw. die Achse nach Ablauf der Lebenszeichenüberwachung mit maximaler Verzögerung angehalten. In diesem Fall wird eine Fehlermeldung angezeigt ("ErrorID" = 16#8013) und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. • Wenn während des Betriebs mit der Kinematiksteuertafel ein Dialog, z. B. "Speichern unter", die Kinematiksteuertafel überdeckt, wird die Kinematik bzw. die Achse mit maximaler Verzögerung angehalten und die Steuerungshoheit an das Anwenderprogramm zurückgegeben. Wenn Sie während des Betriebs mit der Kinematiksteuertafel innerhalb des TIA Portals in ein anderes Fenster, z. B. in die Projektnavigation, wechseln, bleibt die Steuerungshoheit und Bewegung der Kinematik bzw. der Achse bestehen,

Bereich	Element	Beschreibung
		sofern die Kinematiksteuertafel im TIA Portal eingebettet ist. Wenn die Kinematiksteuertafel vom TIA Portal abgelöst ist und Sie innerhalb des TIA Portals in ein anderes Fenster, z. B. in die Projektnavigation, wechseln, bleibt die Steuerungshoheit bestehen, aber die Kinematik bzw. die Achse wird mit maximaler Verzögerung angehalten. Wenn Sie während des Betriebs mit der Kinematiksteuertafel außerhalb des TIA Portals in ein anderes Fenster wechseln, bleibt die Steuerungshoheit bestehen, aber die Kinematik bzw. die Achse wird mit maximaler Verzögerung angehalten.
	Schaltfläche "Abgeben"	Mit der Schaltfläche "Abgeben" geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück.
Kinematik		Im Bereich "Kinematik" geben Sie das Technologieobjekt frei oder sperren dieses.
	Schaltfläche "Freigeben"	Mit der Schaltfläche "Freigeben" geben Sie die verschalteten Achsen des ausgewählten Technologieobjekts Kinematik frei.
	Schaltfläche "Sperren"	Mit der Schaltfläche "Sperren" sperren Sie die verschalteten Achsen des ausgewählten Technologieobjekts Kinematik.
Betriebsart		In der Klappliste "Betriebsart" wählen Sie die gewünschte Betriebsart der Kinematiksteuertafel aus.

Bereich	Element	Beschreibung
Steuern		Im Bereich "Steuern" werden die Parameter für das Verfahren mit der Kinematiksteuertafel entsprechend der ausgewählten Betriebsart angezeigt.
	Koordinatensystem	In der Klappliste "Koordinatensystem" wählen Sie das gewünschte Koordinatensystem aus, in dem die Kinematik verfahren wird. (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Aktives Werkzeug	In der Klappliste "Aktives Werkzeug" wählen Sie das gewünschte Werkzeug aus.
	Optionskästchen "Dynamik anpassen"	Wenn Sie das Optionskästchen aktivieren, können Sie die Werte für Beschleunigung, Verzögerung und Ruck bearbeiten. (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Beschleunigung	Beschleunigung, mit der die Kinematik abhängig vom Kinematiktyp in x-, y-, z-Richtung und die Orientierung verfahren wird. Vorbelegung: 10 % des Defaultwerts Die Werte können Sie nur bearbeiten, wenn das Optionskästchen "Dynamik anpassen" aktiviert ist. (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Verzögerung	Verzögerung, mit der die Kinematik abhängig vom Kinematiktyp in x-, y-, z-Richtung und die Orientierung verfahren wird. Vorbelegung: 100 % des Defaultwerts Die Werte können Sie nur bearbeiten, wenn das Optionskästchen "Dynamik anpassen" aktiviert ist. (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Ruck	Ruck, mit dem die Kinematik abhängig vom Kinematiktyp in x-, y-, z-Richtung und die Orientierung verfahren wird. Vorbelegung: 100 % des Defaultwerts Die Werte können Sie nur bearbeiten, wenn das Optionskästchen "Dynamik anpassen" aktiviert ist. (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Geschwindigkeit	Geschwindigkeit, mit der die Kinematik abhängig vom Kinematiktyp in x-, y-, z-Richtung und die Orientierung verfahren wird. Vorbelegung: 10 % des Defaultwerts Mit dem Schieberegler stellen Sie die Geschwindigkeit prozentual zu den eingestellten Geschwindigkeitswerten zwischen 0 % und 200 % ein (Standardwert 100 %). (Nur Betriebsarten "Tippen" und "Auf Zielposition tippen")
	Zielposition	Position, auf welche die Kinematik bzw. eine Achse verfahren wird. (Nur Betriebsarten "Auf Zielposition tippen" und "Einzelne Achsen: Referenzpunkt setzen")
		Position, auf welche der Referenzpunkt gesetzt wird. (Nur Betriebsart "Einzelne Achsen: Referenzpunkt setzen")
	Schaltfläche "Setzen"	Mit der Schaltfläche "Setzen" setzen Sie einen Referenzpunkt. (Nur Betriebsart "Einzelne Achsen: Referenzpunkt setzen")
	Schaltfläche "Start"	Mit der Schaltfläche "Start" stoßen Sie eine Bewegung gemäß der gewählten Betriebsart an. (Nur Betriebsarten "Auf Zielposition tippen" und "Einzelne Achsen: Referenzieren")
	Schaltfläche "Vorwärts"	Mit der Schaltfläche "Vorwärts" stoßen Sie eine Bewegung gemäß der gewählten Betriebsart in positive Richtung an.

Bereich	Element	Beschreibung
	Schaltfläche "Rückwärts"	Mit der Schaltfläche "Rückwärts" stoßen Sie eine Bewegung gemäß der gewählten Betriebsart in negative Richtung an.
Status		Im Bereich "Status Achse" wird der Status der Achse und der Status des Antriebs angezeigt.
	Freigegeben	Das Technologieobjekt ist freigegeben. Die Achse kann mit Bewegungsaufträgen verfahren werden.
	Referenziert	Das Technologieobjekt ist referenziert.
	Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Fehlermeldungen werden im Inspektorfenster unter "Diagnose > Meldungsanzeige" angezeigt.
Aktuelle Positionswerte		Im Bereich "Aktuelle Positionswerte" werden die Istwerte der Achse angezeigt.
	Koordinatensystem	Koordinatensystem, in dem die Kinematik bzw. eine Achse aktuell verfahren wird. In der Klappliste auf der rechten Seite können Sie ein weiteres Koordinatensystem auswählen, um die Istposition des aktiven Werkzeugs entsprechend in diesem Koordinatensystem anzuzeigen.
	Position x	Aktuelle Position und Drehung des Werkzeugarbeitspunkts im eingestellten Koordinatensystem.
	Position y	
	Position z	
	Drehung A	

Hinweis

Keine Übernahme der Parameter

Die eingestellten Parameterwerte werden mit der Rückgabe der Steuerungshoheit verworfen. Übertragen Sie die Werte bei Bedarf in Ihre Konfiguration.

Wenn Sie während des Betriebs mit der Kinematiksteuertafel Konfigurationswerte geändert haben, haben diese Änderungen keine Auswirkung auf den Betrieb der Kinematiksteuertafel.

Betriebsart

Die folgende Tabelle zeigt die Betriebsarten der Kinematiksteuertafel:

Betriebsart	Beschreibung
Tippen	Mit der Schaltfläche "Vorwärts" verfahren Sie eine Achse durch Tippen in positiver Richtung. Mit der Schaltfläche "Rückwärts" verfahren Sie eine Achse durch Tippen in negativer Richtung. Die jeweilige Achse verfährt, solange Sie die Schaltfläche "Vorwärts" bzw. "Rückwärts" gedrückt halten.
Auf Zielposition tippen	Mit der Schaltfläche "Start" verfahren Sie die Kinematik bzw. eine Achse durch Tippen auf die unter "Zielposition" angegebene Position. Dabei beschränkt sich die Bewegungsrichtung der kartesischen Orientierung nicht nur auf die positive Richtung. Die Orientierungsachse wird mit dem Modus "Kürzester Weg" verfahren. Die Kinematik verfährt, solange Sie die Schaltfläche "Start" gedrückt halten. Beim Erreichen der Zielposition hält die Kinematik automatisch an. Die angegebene Position bezieht sich auf das Koordinatensystem, das in der Klappliste "Koordinatensystem" ausgewählt ist.

Betriebsart	Beschreibung
Einzelne Achsen: Referenzpunkt setzen	Mit der Schaltfläche "Setzen" setzen Sie den Referenzpunkt der jeweiligen Achse auf den unter "Zielposition" angegebenen Wert. Für die entsprechende Achse wird der Status "Referenziert" gesetzt. Die angegebene Position bezieht sich auf das Maschinenkoordinatensystem (MCS), das mit dieser Betriebsart in der Klappliste "Koordinatensystem" voreingestellt ist. Diese Funktion entspricht dem direkten Referenzieren (absolut). Das Referenzieren ist mit einem Absolutwertgeber nicht möglich. Das Technologieobjekt wird bei Verwendung dieser Betriebsart mit einem Absolutwertgeber nicht referenziert.
Einzelne Achsen: Referenzieren	Mit der Schaltfläche "Start" verfahren Sie eine Achse durch Tippen auf die festgelegte Referenzposition. Die jeweilige Achse verfährt, solange Sie die Schaltfläche "Start" gedrückt halten. Beim Erreichen der Referenzposition hält die Achse automatisch an.

Kinematiksteuertafel einsetzen

Mit der Kinematiksteuertafel übernehmen Sie die Steuerungshoheit für ein Technologieobjekt Kinematik und steuern die Bewegungen der Kinematik oder der einzelnen Achsen.

Voraussetzung

- Das Projekt ist erstellt und in die CPU geladen.
- Die CPU ist im Betriebszustand RUN.
- Die verschalteten Achsen der Kinematik sind über Ihr Anwenderprogramm gesperrt ("MC_Power.Enable" = FALSE).
- Die Kinematiksteuertafel für das Technologieobjekt wird nicht von einer anderen Instanz des TIA Portals verwendet (Team Engineering ab CPU V1.5).
- Die Antriebe sind betriebsbereit.

Vorgehen

Gehen Sie zum Steuern der Kinematik bzw. der Kinematikachsen mit der Kinematiksteuertafel folgendermaßen vor:

1. Um die Steuerungshoheit für das Technologieobjekt zu übernehmen und eine Onlineverbindung zur CPU aufzubauen, klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Holen".
   Ein Warnhinweis wird angezeigt.
2. Passen Sie gegebenenfalls die Lebenszeichenüberwachung an und klicken Sie auf die Schaltfläche "OK".
3. Um das Technologieobjekt freizugeben, klicken Sie im Bereich "Kinematik" auf die Schaltfläche "Freigeben".
4. Wählen Sie in der Klappliste im Bereich "Betriebsart" die gewünschte Funktion der Kinematiksteuertafel aus.
5. Wählen Sie abhängig von der eingestellten Betriebsart im Bereich "Steuern" in der Klappliste "Koordinatensystem" das Bezugskoordinatensystem aus.
6. Geben Sie im Bereich "Steuern" die entsprechenden Parameterwerte für Ihren Auftrag an.

7. Klicken Sie abhängig von der eingestellten Betriebsart auf die Schaltfläche "Setzen", "Start", "Vorwärts" oder "Rückwärts", um den Auftrag zu starten.
8. Wiederholen Sie für weitere Aufträge die Schritte 4 bis 7.
9. Um das Technologieobjekt zu sperren, klicken Sie im Bereich "Achse" auf die Schaltfläche "Sperren".
10. Um die Steuerungshoheit an Ihr Anwenderprogramm zurückzugeben, klicken Sie im Bereich "Steuerungshoheit" auf die Schaltfläche "Abgeben".

Kinematik-Trace

Der Kinematik-Trace bietet folgende Funktionen:

- 3D-Visualisierung der aktuellen Bewegung des Werkzeugarbeitspunkts (TCP)
- Aufzeichnen und abspielen der Kinematikbewegungen
- Exportieren und importieren der Aufzeichnungen

In der Projektnavigation finden Sie den Kinematik-Trace des Technologieobjekts Kinematik unter "Technologieobjekt > Kinematik-Trace".

3D-Visualisierung

Die Ansicht "3D-Visualisierung" des Kinematik-Trace ist in folgende Bereiche aufgeteilt:

- Grafische Anzeige
  In der grafischen Anzeige wird die Kinematik und die Bewegung des Werkzeugarbeitspunkts (TCP) als Leuchtspur dargestellt.

  Hinweis

  Die Skalierung der Kinematik im Kinematik-Trace stellen Sie unter "Technologieobjekt > Konfiguration > Geometrie (Seite 2342)" ein.

- Tabellarischer Editor
  Im tabellarischen Editor werden die Aufzeichnungen aufgelistet.

Werkzeugleiste

Die Werkzeugleiste des Kinematik-Trace stellt Ihnen über Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	Alle beobachten	Online-Verbindung aufbauen/trennen Der Kinematik-Trace baut eine Online-Verbindung zum Gerät auf. Wenn die Online- und Offline-Trace-Konfigurationen unterschiedlich sind, wird die Trace-Konfiguration auf das Gerät geladen.
	Aufzeichnung starten	Siehe Kapitel "Bahnbewegungen aufzeichnen und abspielen (Seite 2378)"
	Aufzeichnung beenden	

Schaltfläche	Funktion	Beschreibung
	Aufzeichnung aus Datei importieren	Siehe Kapitel "Aufzeichnungen importieren und exportieren (Seite 2381)"
	Markierte Aufzeichnung in Datei exportieren	
	Markierte Aufzeichnung zu Messungen hinzufügen	Aufzeichnungen werden den Messungen unter Trace hinzugefügt. Siehe Kapitel "Aufzeichnungen importieren und exportieren (Seite 2381)"

Die Werkzeugleiste am oberen Rand der grafischen Anzeige stellt Ihnen über Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	Zu Live-Anzeige wechseln	Live-Anzeige der Kinematik einschalten/ausschalten Mit der Live-Anzeige können Sie sich ansehen, wie die Kinematik sich aktuell bewegt. Diese Funktion ist nur im Online-Modus verfügbar.
	Dunkler	Helligkeit der grafischen Ansicht einstellen
100	Helligkeit wählen	
	Heller	
	2D-Darstellung anzeigen	Ansicht auf 2D-Darstellung umschalten
	3D-Darstellung anzeigen	Ansicht auf 3D-Darstellung umschalten
	TCP fokussieren	Ansicht auf "TCP fokussieren" umschalten Während der Bahnbewegung der Kinematik liegt der Fokus immer auf dem TCP. Im Stillstand können Sie die Ansicht mit der Maus verschieben oder rotieren.
	An Bildschirmgröße anpassen	Die Ansicht wird zentriert und die gesamte Kinematik dargestellt.
	Koordinatensystem anzeigen	Ausgewähltes Koordinatensystem einschließlich der Gitternetzlinien ausblenden/einblenden
WCS	Koordinatensystem wählen	Koordinatensystem auswählen
	xz-Ebene anzeigen	Die xz-Ebene wird angezeigt.
	xz-Ebene gedreht anzeigen	Die xz-Ebene wird um die z-Achse gedreht angezeigt.
	yz-Ebene anzeigen	Die yz-Ebene wird angezeigt.
	yz-Ebene gedreht anzeigen	Die yz-Ebene wird um die z-Achse gedreht angezeigt.
	xy-Ebene anzeigen	Die xy-Ebene wird angezeigt.
	xy-Ebene gedreht anzeigen	Die xy-Ebene wird um die x-Achse gedreht angezeigt.

Schaltfläche	Funktion	Beschreibung
	Vereinfachtes Modell anzeigen	Vereinfachte Darstellung der Kinematik oder Kinematik ausblenden/einblenden
	TCS hervorheben	Werkzeugkoordinatensystem einblenden/ausblenden

Die Werkzeugleiste am unteren Rand der grafischen Anzeige stellt Ihnen über Schaltflächen folgende Funktionen zur Verfügung:

Schaltfläche	Funktion	Beschreibung
	Springe zu Anfang	Zum Anfang der Aufzeichnung springen
	Schritt rückwärts	In der Aufzeichnung schrittweise rückwärts springen
	Abspielen	Aufzeichnung abspielen
	Pause	Aufzeichnung pausieren
	Stopp	Aufzeichnung stoppen
	Schritt vorwärts	In der Aufzeichnung schrittweise vorwärts springen
	Springe zu Ende	Zum Ende der Aufzeichnung springen

Bedienung der grafischen Anzeige

Innerhalb der grafischen Anzeige haben Sie mit der Maus folgende Bedienmöglichkeiten:

- Koordinatensystem drehen und zoomen
- Wenn Sie den Mauszeiger auf der Leuchtspur positionieren, werden für die entsprechende Position folgende Werte eingeblendet:
  - Messpunktnummer
  - Koordinaten x, y, z und A

Bahnbewegungen aufzeichnen und abspielen

Bahnbewegung aufzeichnen

Um eine Bahnbewegung der Kinematik aufzuzeichnen, klicken Sie auf die Schaltfläche . Wenn noch keine Online-Verbindung besteht, wird automatisch eine Online-Verbindung aufgebaut.

Die Aufzeichnung erfolgt mit den Vorgaben, die unter "Konfiguration" (Seite 2380) hinterlegt sind.

Während der Aufzeichnung zeigt die grafische Anzeige ein Live-Bild der Kinematik. Bei Anwendertransformationen zeigt die grafische Anzeige stattdessen den bewegten TCP (Werkzeugarbeitspunkt).

Die "Aktuelle Aufzeichnung" sowie die Konfiguration bleiben bis zum Schließen der Benutzeroberfläche "Kinematik-Trace" erhalten.

Wenn Sie eine neue Aufzeichnung starten, wird die "Aktuelle Aufzeichnung" überschrieben.

Aktuelle Aufzeichnung speichern

1. Editieren Sie in der "Aktuellen Aufzeichnung" die Einträge für Name der Leuchtspur, Farbe der Leuchtspur, Kommentar entsprechend Ihren Erfordernissen.
2. Um die "Aktuelle Aufzeichnung" zu speichern, verwenden Sie das Symbol in der Tabelle oder das Kontextmenü.
   Die Aufzeichnung wird im tabellarischen Editor eingefügt. Eine gespeicherte Aufzeichnung enthält folgende Kinematikdaten:
   - Kinematikkoordinaten
   - Kinematiktyp
   - Zum Zeitpunkt der Aufzeichnung gültige Online-Geometrie

Hinweis

Der tabellarischen Editor kann maximal 20 Aufzeichnungen speichern. Wenn Sie weitere Aufzeichnungen speichern wollen, müssen Sie nicht benötigte Aufzeichnungen löschen.

Werkzeugleiste für Aufzeichnungen

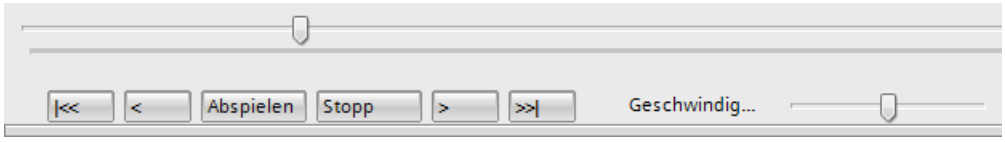

Die Werkzeugleiste für Aufzeichnungen bietet folgende Bedienmöglichkeiten:

- Zeitleiste mit Schieberegler
  Um das Abspielen an einer bestimmten Position zu stoppen, klicken Sie direkt auf die gewünschte Position in der Zeitleiste.
- Schaltflächen zum Abspielen und Stoppen der Aufzeichnung
- Einen Messpunkt vor bzw. zurück
- Sprung an den Anfang bzw. an das Ende der Aufzeichnung
- Schieberegler zum Einstellen der Abspielgeschwindigkeit

Bahnbewegung als Leuchtspur anzeigen und abspielen

1. Wenn der Live-Monitor der Kinematik aktiviert ist, schalten Sie den Live-Monitor über die Schaltfläche in der Werkzeugleiste aus.
2. Um die Leuchtspur einzuschalten, klicken Sie auf die Schaltfläche der Aufzeichnung, die Sie abspielen wollen.
   Die Aufzeichnung ist dadurch ausgewählt. Die grafische Anzeige zeigt die vollständige Bahnbewegung der Aufzeichnung an.
3. Klicken Sie auf die Schaltfläche "Abspielen" in der Werkzeugleiste.
   Die ausgewählte Aufzeichnung wird abgespielt. Die Aufzeichnung zeigt:
   – Die aufgezeichnete Bahn in der grafischen Ansicht als Leuchtspur
   – Die Kinematik mit der zum Zeitpunkt der Aufzeichnung gültigen Online-Geometrie
     Wenn mehrere Leuchtspuren sichtbar sind, wird nur die Kinematik der ausgewählten Aufzeichnung in der grafischen Anzeige angezeigt.

Konfiguration

Unter "Konfiguration" legen Sie die Parameterwerte für die Aufzeichnung fest.

Die Aufzeichnung der Bahnbewegungen können Sie über die Werkzeugleiste mit der Schaltfläche starten.

Abtastung

Parameter	Beschreibung
Aufzeichnungszeitpunkt	Auswahl zwischen folgenden OBs: • MC-Servo • MC-Interpolator
Aufzeichnen alle	Angabe des Werts für das Aufzeichnungsintervall. Bei MC-Servo haben Sie über eine Klappliste folgende Auswahlmöglichkeiten: • Angabe in "Zyklen" • Angabe in "Sekunden"
Max. Aufzeichnungsdauer	Anzeige der berechneten maximalen Aufzeichnungsdauer. Bei einer vorgegebenen maximalen Anzahl Messpunkte ist die maximale Aufzeichnungsdauer abhängig vom angegebenen Aufzeichnungsintervall.
Max. Aufzeichnungsdauer verwenden	Mit Aktivierung des Optionskästchens wird die Aufzeichnungsdauer auf die maximal mögliche Aufzeichnungsdauer gesetzt.
Aufzeichnungsdauer (a)	Angabe der Aufzeichnungsdauer: • In Sekunden • Anzahl der Messpunkte

Trigger

Parameter	Beschreibung
Triggermodus	• Sofort aufzeichnen Die Aufzeichnung startet sofort nach dem Laden der Konfiguration. • Trigger auf Variable Das System wartet auf ein Trigger-Ereignis, das die Aufzeichnung auslöst.
Triggervariable	Für den Modus "Trigger auf Variable" benötigen Sie eine verknüpfte Variable vom Typ BOOL.
Ereignis	Wählen Sie das Ereignis aus, das als Trigger genutzt werden soll: • Steigende Flanke • Fallende Flanke
Vortrigger	Mit dem "Vortrigger" definieren Sie die Messpunkte, die bereits vor der Erfüllung der eigentlichen Triggerbedingung aufgezeichnet werden. Die Angabe erfolgt: • In Sekunden • Anzahl der Messpunkte

Aufzeichnungen importieren und exportieren

Für die aktuelle oder für die gespeicherten Aufzeichnungen gibt es verschiedene Export- und Importmöglichkeiten.

Aufzeichnung als Datei exportieren bzw. importieren

Der Export beinhaltet die TCP-Positionswerte sowie die Konfiguration des Technologieobjekts Kinematik.

Um eine Aufzeichnung zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die zu exportierende Aufzeichnung aus.
2. Klicken Sie auf die Schaltfläche .
3. Wählen Sie das gewünschte Dateiformat "*.csv" oder "*.ltr" aus.
4. Wählen Sie den gewünschten Ordner aus.
5. Klicken Sie auf "Datei exportieren".

Eine exportierte *.ltr-Datei kann in das Technologieobjekt Kinematik erneut importiert werden.

Um eine Aufzeichnung zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekt Kinematik > Kinematik-Trace".
2. Klicken Sie auf die Schaltfläche .

3. Wählen Sie den gewünschten Ordner aus.
4. Klicken Sie auf "Datei importieren".
   Nach dem Import wird unter 3D-Visualisierung Folgendes angezeigt:
   – Die importierte Kinematik
   – Die Bahnbewegungen

Aufzeichnung als Messung speichern

Um die Aufzeichnung als Messung zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie die "Aktuelle Aufzeichnung" oder eine "Gespeicherte Aufzeichnung".
2. Speichern Sie die Aufzeichnung über die Schaltfläche in der Werkzeugleiste.
   Die Ablage der Aufzeichnung erfolgt unter "Traces > Messungen".

Die Datei beinhaltet die TCP-Koordinaten (x, y, z, A) sowie die Positionen der verbundenen Achsen.

Die unter "Messungen" gespeicherten Aufzeichnungen können Sie in den Trace laden und anzeigen. Mit Trace haben Sie erweiterte Auswertemöglichkeiten, um Bahnbewegungen im Detail zu analysieren.

Hinweis

Die unter Messungen gespeicherten Aufzeichnungen können Sie nicht wieder in den Kinematik-Trace importieren.

6.1.3.7 Diagnose

Das Kapitel "Diagnose" beschränkt sich auf die Beschreibung der Diagnosesicht der einzelnen Technologieobjekte im TIA Portal.

Eine umfassende Beschreibung der Systemdiagnose der CPU S7-1500 finden Sie im Funktionshandbuch "Diagnose" (http://support.automation.siemens.com/WW/view/de/59192926).

Die Beschreibung des Diagnosekonzepts für Motion Control finden Sie im Kapitel "Diagnosekonzept".

Siehe auch

Funktionshandbuch "S7-1500T Motion Control" Kapitel "Diagnosekonzept" (https://support.industry.siemens.com/cs/ww/de/view/109749263)

Technologieobjekt Kinematik

Status- und Fehlerbits

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status- und Fehlerbits" überwachen Sie im TIA Portal die Status- und Fehlermeldungen des Technologieobjekts. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

In den folgenden Tabellen wird die Bedeutung der Status- und Fehlermeldungen beschrieben. In Klammern wird die zugehörige Variable des Technologieobjekts angegeben.

Status Kinematik

Die folgende Tabelle zeigt mögliche Zustände der Kinematik:

Status	Beschreibung
Fehler	Am Technologieobjekt ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie dem Bereich "Fehler" und den Variablen des Technologieobjekts "<TO>.ErrorDetail.Number" und "<TO>.ErrorDetail.Reaction". (<TO>.StatusWord.X1 (Error))
Restart aktiv	Das Technologieobjekt wird neu initialisiert. (<TO>.StatusWord.X2 (RestartActive))
Kinematiksteuertafel aktiv	Die Kinematiksteuertafel ist aktiviert. Die Kinematiksteuertafel hat die Steuerhoheit über das Technologieobjekt. Die Kinematik kann nicht vom Anwenderprogramm gesteuert werden. (<TO>.StatusWord.X4 (ControlPanelActive))
Restart erforderlich	Restart-relevante Daten wurden verändert. Die Änderungen werden erst mit dem Restart des Technologieobjekts übernommen. (<TO>.StatusWord.X3 (OnlineStartValuesChanged))

Status Bewegung

Die folgende Tabelle zeigt mögliche Zustände der Kinematikbewegung:

Status	Beschreibung
Done (kein Auftrag aktiv)	Am Technologieobjekt ist kein Bewegungsauftrag aktiv. (<TO>.StatusWord.X6 (Done))
Linearbewegung aktiv	Am Technologieobjekt ist ein Linearbewegungsauftrag aktiv. (<TO>.StatusWord.X8 (LinearCommand))
Kreisbewegung aktiv	Am Technologieobjekt ist ein Kreisbewegungsauftrag aktiv. (<TO>.StatusWord.X9 (CircularCommand))
Konstante Geschwindigkeit	Die Kinematik wird mit konstanter Geschwindigkeit verfahren oder befindet sich im Stillstand. (<TO>.StatusWord.X12 (ConstantVelocity))
Beschleunigen	Die Kinematik wird beschleunigt. (<TO>.StatusWord.X13 (Accelerating))
Verzögern	Die Kinematik wird abgebremst. (<TO>.StatusWord.X14 (Decelerating))

Status	Beschreibung
Bewegung unterbrochen	Die aktive Kinematikbewegung wird durch einen "MC_GroupInterrupt"-Auftrag unterbrochen. (<TO>.StatusWord.X17 (Interrupted))
Orientierungsbewegung aktiv	Am Technologieobjekt ist eine Orientierungsbewegung aktiv. (<TO>.StatusWord.X15 (OrientationMotion))

Fehler

Die folgende Tabelle zeigt mögliche Fehler:

Fehler	Beschreibung
System	Ein systeminterner Fehler ist aufgetreten. (<TO>.ErrorWord.X0 (SystemFault))
Konfiguration	Ein Konfigurationsfehler ist aufgetreten. Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig. Das Technologieobjekt wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert. (<TO>.ErrorWord.X1 (ConfigFault))
Transformation	Ein Transformationsfehler ist aufgetreten. (<TO>.ErrorWord.X4 (TransformationFault))
Anwenderprogramm	Ein Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung ist aufgetreten. (<TO>.ErrorWord.X2 (UserFault))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind (z. B. Technologieobjekt nicht referenziert). (<TO>.ErrorWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.ErrorWord.X6 (DynamicError))

Warnungen

Die folgende Tabelle zeigt mögliche Warnungen:

Warnung	Beschreibung
Konfiguration	Einer oder mehrere Konfigurationsparameter werden zeitweise intern angepasst. (<TO>.WarningWord.X1 (ConfigWarning))
Auftrag abgewiesen	Ein Auftrag ist nicht ausführbar. Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind. (<TO>.WarningWord.X3 (CommandNotAccepted))
Dynamikbegrenzung	Die Dynamikwerte werden auf die Dynamikgrenzen begrenzt. (<TO>.WarningWord.X6 (DynamicWarning))

Siehe auch

Variable "StatusWord" (Kinematik) (Seite 2410)

Variable "ErrorWord" (Kinematik) (Seite 2413)

Variable "WarningWord" (Kinematik) (Seite 2415)

Status Bewegung

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Status Bewegung" überwachen Sie im TIA Portal den Bewegungsstatus der Kinematik. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Werkzeugarbeitspunkt (TCP)"

Die folgende Tabelle beschreibt die Bedeutung der Positionsinformationen des Werkzeugarbeitspunkts:

Status	Beschreibung
Koordinatensystem	Bezugskoordinatensystem In diesem Feld ist das Weltkoordinatensystem (WCS) voreingestellt.
Position x	x-Koordinate des TCP im WCS (<TO>.TcpInWcs.x.Position)
Position y	y-Koordinate des TCP im WCS (<TO>.TcpInWcs.y.Position)
Position z	z-Koordinate des TCP im WCS (<TO>.TcpInWcs.z.Position)
Drehung A	A-Koordinate des TCP im WCS (<TO>.TcpInWcs.a.Position)
Koordinatensystem	Bezugskoordinatensystem In der Klappliste können Sie ein weiteres Koordinatensystem auswählen, um die Istposition des aktiven Werkzeugs entsprechend in diesem Koordinatensystem anzuzeigen.
Position x	x-Koordinate des aktiven Werkzeugs im eingestellten Koordinatensystem
Position y	y-Koordinate des aktiven Werkzeugs im eingestellten Koordinatensystem
Position z	z-Koordinate des aktiven Werkzeugs im eingestellten Koordinatensystem
Drehung A	A-Koordinate des aktiven Werkzeugs im eingestellten Koordinatensystem

Bereich "Dynamikwerte der Kinematik"

Die folgende Tabelle beschreibt die Bedeutung der Dynamikinformationen:

Status	Beschreibung
Bahndynamik auf Achsdynamik begrenzen	Anzeige der Dynamikwerte mit oder ohne konfigurierter Dynamikadaption
Geschwindigkeit	Bahngeschwindigkeit (<TO>.StatusPath.Velocity)

Status	Beschreibung
Beschleunigung	Bahnbeschleunigung (<TO>.StatusPath.Acceleration)
Override	Prozentuale Korrektur der Geschwindigkeitsvorgabe Die von Motion Control-Anweisungen oder von der Kinematiksteuertafel vorgegebene Sollgeschwindigkeit wird durch einen Override überlagert und prozentual angepasst. Als Geschwindigkeitskorrektur sind Werte von 0.0 % bis 200.0 % zulässig. (<TO>.Override.Velocity)

Bereich "Auftragskette"

Die folgende Tabelle beschreibt die Bedeutung der Auftragsinformationen:

Status	Beschreibung
Aufträge in der Auftragskette	Aktuelle Anzahl der Aufträge für das Technologieobjekt Kinematik in der Auftragskette (<TO>.StatusMotionQueue.NumberOfCommands)

Zonen und Werkzeuge

Mit der Diagnosefunktion "Technologieobjekt > Diagnose > Zonen und Werkzeuge" überwachen Sie im TIA Portal den Zonen- und Werkzeugstatus der Kinematik. Die Diagnosefunktion steht im Onlinebetrieb zur Verfügung.

Bereich "Zonen"

Die Tabellen "Arbeitsraumzonen" und "Kinematikzonen" zeigen den Status der einzelnen Zonen an. Dazu werden folgende Symbole angezeigt:

Symbol	Beschreibung
	Die Zone ist ungültig.
	Die Zone ist inaktiv.
	Die Zone ist aktiv.
	Die Zone wurde verletzt.

Bereich "Aktive Werkzeugparameter"

Die folgende Tabelle beschreibt die Bedeutung der Werkzeuginformationen:

Status	Beschreibung
Aktives Werkzeug	Aktuell wirksames Werkzeug (<TO>.StatusTool.ActiveTool)
Werkzeugarbeitspunkt im FCS	Im Bereich "Werkzeugarbeitspunkt im FCS" werden die Werte des aktuellen Tool-Frames im Flanschkoordinatensystem (FCS) angezeigt.

Status	Beschreibung
Position x	x-Koordinate (<TO>.StatusTool.Frame[1].x)
Position y	y-Koordinate (<TO>.StatusTool.Frame[1].y)
Position z	z-Koordinate (<TO>.StatusTool.Frame[1].z)
Drehung A	A-Koordinate (<TO>.StatusTool.Frame[1].a)

6.1.3.8 Anhang

Variablen des Technologieobjekts Kinematik

Legende

Variable	Name der Variable	
Datentyp	Datentyp der Variable	
Werte	Wertebereich der Variable - Minimalwert bis Maximalwert (L = lineare Angabe, R = rotatorische Angabe) Ohne spezifische Wertangabe gelten die Wertebereichsgrenzen des jeweiligen Datentyps bzw. die Angabe unter "Beschreibung".	
W	Wirksamkeit von Änderungen im Technologie-Datenbaustein	
	DIR	Direkt: Wertänderungen erfolgen über direkte Zuweisung und werden mit dem Start des nächsten MC-Servo [OB91] wirksam.
	CAL	Mit Aufruf der Motion Control-Anweisung: Wertänderungen erfolgen über direkte Zuweisung und werden nach dem Aufruf der entsprechenden Motion Control-Anweisung im Anwenderprogramm mit dem Start des nächsten MC-Servo [OB91] wirksam.
	RES	Restart: Änderungen des Startwerts im Ladespeicher erfolgen über die erweiterte Anweisung "WRIT_DBL" (In DB im Ladespeicher schreiben). Änderungen werden erst nach Restart des Technologieobjekts wirksam.
	RON	Read only: Die Variable kann bzw. darf zur Laufzeit des Anwenderprogramms nicht verändert werden.
Beschreibung	Beschreibung der Variable	

Der Zugriff auf die Variablen erfolgt über "<TO>.<Variablenname>". Der Platzhalter <TO> repräsentiert den Namen des Technologieobjekts.

Variable "Tcp" (Kinematik)

Die Variablenstruktur "<TO>.Tcp.<Variablenname>" beinhaltet die Position des Werkzeugarbeitspunkts (TCP), den TCP-Frame im Weltkoordinatensystem (WCS)

Variablen

Variable	Datentyp	Werte	W	Beschreibung
Tcp.	STRUCT			
x	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate
y	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate
z	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate
a	LREAL	-180 bis 180	RON	A-Koordinate

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "Kinematics" (Kinematik)

Die Variablenstruktur "<TO>.Kinematics.<Variablenname>" beinhaltet den definierten Kinematiktyp.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
Kinematics.	STRUCT				
TypeOfKinematics	DINT	1 bis 34	RON	Kinematiktyp Die Konfiguration erfolgt über die Funktionssicht.	
				1	Kartesisches Portal 2D
				2	Kartesisches Portal 2D mit Orientierung
				3	Kartesisches Portal 3D
				4	Kartesisches Portal 3D mit Orientierung
				5	Rollen-Picker 2D
				6	Rollen-Picker 2D mit Orientierung
				7	Rollen-Picker 3D (vertikal)
				8	Rollen-Picker 3D mit Orientierung (vertikal)
				9	Rollen-Picker 3D mit Orientierung (horizontal)
				10	SCARA 3D mit Orientierung
				11	Knickarm 2D
				12	Knickarm 2D mit Orientierung
				13	Knickarm 3D
				14	Knickarm 3D mit Orientierung
				15	Delta-Picker 2D
				16	Delta-Picker 2D mit Orientierung
				17	Delta-Picker 3D
				18	Delta-Picker 3D mit Orientierung
				19 … 20	Reserviert
				21	Zylindrischer Roboter 3D
				22	Zylindrischer Roboter 3D mit Orientierung
				23	Tripod 3D
				24	Tripod 3D mit Orientierung
				25 … 30	Reserviert
				31	Anwenderdefiniert 2D
				32	Anwenderdefiniert 2D mit Orientierung
				33	Anwenderdefiniert 3D
				34	Anwenderdefiniert 3D mit Orientierung
Parameter[1..32]	ARRAY [1..32] OF LREAL	-1.0E12 bis 1.0E12	RES	Kinematikspezifische Parameter	

Siehe auch

Variablen Portal (Seite 2249)
Variablen Delta-Picker (Seite 2298)
Variablen Rollen-Picker (Seite 2263)
Variablen Knickarm (Seite 2284)

Variablen zylindrischer Roboter (Seite 2307)

Variablen Tripod (Seite 2314)

Variablen SCARA (Seite 2268)

Variablen anwenderdefinierte Kinematiken (Seite 2315)

Variable "KcsFrame" (Kinematik)

Die Variablenstruktur "<TO>.KcsFrame.<Variablenname>" beinhaltet den Frame des Kinematikkoordinatensystems (KCS) im Weltkoordinatensystem (WCS).

Variablen

Variable	Datentyp	Wertebereich	W	Beschreibung
KcsFrame.	STRUCT			
x	LREAL	-1.0E12 bis 1.0E12	RES	x-Koordinate
y	LREAL	-1.0E12 bis 1.0E12	RES	y-Koordinate
z	LREAL	-1.0E12 bis 1.0E12	RES	z-Koordinate
a	LREAL		RES	A-Koordinate
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		-180.0 bis 179.999		Bei Kinematiktyp "3D mit Orientierung"
b	LREAL		RES	B-Koordinate
		-180.0 bis 179.999		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-90.0 bis 90.0		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"
c	LREAL		RES	C-Koordinate
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "OcsFrame" (Kinematik)

Die Variablenstruktur "<TO>.OcsFrame[1..3].<Variablenname>" beinhaltet die Frames der Objektkoordinatensysteme 1 bis 3 (OCS) im Weltkoordinatensystem (WCS).

Variablen

Variable	Datentyp	Wertebereich	W	Beschreibung
OcsFrame[1..3].	ARRAY [1..3] OF STRUCT			
x	LREAL	-1.0E12 bis 1.0E12	RES	x-Koordinate
y	LREAL	-1.0E12 bis 1.0E12	RES	y-Koordinate
z	LREAL	-1.0E12 bis 1.0E12	RES	z-Koordinate
a	LREAL		RES	A-Koordinate
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		-180.0 bis 179.999		Bei Kinematiktyp "3D mit Orientierung"
b	LREAL		RES	B-Koordinate
		-180.0 bis 179.999		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-90.0 bis 90.0		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"
c	LREAL		RES	C-Koordinate
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "Tool" (Kinematik)

Die Variablenstruktur "<TO>.Tool[1..3].<Variablenname>" beinhaltet den Tool-Frame im Flanschkoordinatensystem (FCS).

Variablen

Variable	Datentyp	Wertebereich	W	Beschreibung
Tool[1..3].	ARRAY [1..3] OF STRUCT			
Frame.	STRUCT	-1.0E12 bis 1.0E12	RES	
x	LREAL	-1.0E12 bis 1.0E12	RES	x-Koordinate im FCS
y	LREAL	-1.0E12 bis 1.0E12	RES	y-Koordinate im FCS
z	LREAL	-1.0E12 bis 1.0E12	RES	z-Koordinate im FCS
a	LREAL		RES	A-Koordinate
		0.0		Bei Kinematiktyp "2D"
		-180.0 bis 179.999		Bei Kinematiktyp "2D mit Orientierung"
		0.0		Bei Kinematiktyp "3D"
		-180.0 bis 179.999		Bei Kinematiktyp "3D mit Orientierung"

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "DynamicDefaults" (Kinematik)

Die Variablenstruktur "<TO>.DynamicDefaults.<Variablenname>" beinhaltet die Konfiguration der Dynamikvoreinstellungen. Diese Einstellungen werden verwendet, wenn Sie an einer Motion Control-Anweisung einen Dynamikwert kleiner 0.0 angeben. Änderungen der Dynamikvoreinstellungen werden mit der nächsten steigenden Flanke am Parameter "Execute" einer Motion Control-Anweisung übernommen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
DynamicDefaults.	STRUCT				
Path.	STRUCT				
Velocity	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Geschwindigkeit der Bahn	
Acceleration	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Beschleunigung der Bahn	
Deceleration	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Verzögerung der Bahn	
Jerk	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung des Rucks der Bahn	
Orientation.	STRUCT				
Velocity	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Geschwindigkeit der kartesischen Orientierung	
Acceleration	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Beschleunigung der kartesischen Orientierung	
Deceleration	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung der Verzögerung der kartesischen Orientierung	
Jerk	LREAL	0.0 bis 1.0E12	DIR	Voreinstellung des Rucks der kartesischen Orientierung	
DynamicAdaption	DINT	0 bis 2	CAL	Voreinstellung der Dynamikadaption	
				0	Keine Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "DynamicLimits" (Kinematik)

Die Variablenstruktur "<TO>.DynamicLimits.<Variablenname>" beinhaltet die Konfiguration der Dynamikgrenzen. Bei der Bewegungsführung werden keine Dynamikwerte größer der Dynamikgrenzen zugelassen. Wenn Sie an einer Motion Control-Anweisung größere Werte angegeben, wird mit den Dynamikgrenzen verfahren und eine Warnung (Alarm 501 bis 503 - Dynamikwerte werden begrenzt) wird angezeigt.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
DynamicLimits.	STRUCT			
Path.	STRUCT			
Velocity	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Geschwindigkeit der Bahn
Acceleration	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Geschwindigkeit der Bahn
Deceleration	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Verzögerung der Bahn
Jerk	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für den maximalen Ruck der Bahn
Orientation.	STRUCT			
Velocity	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Geschwindigkeit der kartesischen Orientierung
Acceleration	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Geschwindigkeit der kartesischen Orientierung
Deceleration	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für die maximale Verzögerung der kartesischen Orientierung
Jerk	LREAL	0.0 bis 1.0E12	DIR	Dynamikbegrenzung für den maximalen Ruck der kartesischen Orientierung

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "MotionQueue" (Kinematik)

Die Variablenstruktur "<TO>.MotionQueue.<Variablenname>" beinhaltet die Konfiguration von Parametern der Auftragskette.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
MotionQueue.	STRUCT			
MaxNumberOfCommands	DINT	1 bis 10	RON	Maximale Anzahl der Aufträge in der Auftragskette Die Konfiguration erfolgt über die Parameteransicht (Datenstruktur).

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "Override" (Kinematik)

Die Variablenstruktur "<TO>.Override.<Variablenname>" beinhaltet die Konfiguration von Override-Parametern.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
Override.	STRUCT			
Velocity	LREAL	0.0 bis 200.0	DIR	Geschwindigkeits-Override

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "WorkspaceZone" (Kinematik)

Die Variablenstruktur "<TO>.WorkspaceZone[1..10].<Variablenname>" beinhaltet die Parameter zu Arbeitsraumzonen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
WorkspaceZone[1..10].	ARRAY [1..10] OF STRUCT				
Active	BOOL	-	RES	FALSE	Arbeitsraumzone deaktiviert
				TRUE	Arbeitsraumzone aktiviert
Valid	BOOL	-	RES	FALSE	Zone ist nicht definiert
				TRUE	Zone ist definiert
Type	DINT	0 bis 2	RES	Typ der Arbeitsraumzone	
				0	Sperrzone
				1	Arbeitszone
				2	Meldezone
ReferenceSystem	DINT	0 bis 3	RES	Referenzkoordinatensystem für die Arbeitsraumzone	
				0	WCS
				1	OCS1
				2	OCS2
				3	OCS3
Frame.	STRUCT				
x	LREAL	-1.0E12 bis 1.0E12	RES	x-Koordinate	
y	LREAL	-1.0E12 bis 1.0E12	RES	y-Koordinate	
z	LREAL	-1.0E12 bis 1.0E12	RES	z-Koordinate	
a	LREAL	-180.0 bis 179.999	RES	A-Koordinate	
b	LREAL	-90.0 bis 90.0	RES	B-Koordinate	
c	LREAL	-180.0 bis 179.999	RES	C-Koordinate	
Geometry.	STRUCT				
Type	DINT	0 bis 2	RES	Zonengeometrie	
				0	Quader
				1	Kugel
				2	Zylinder
Parameter[1..3]	ARRAY [1..3] OF LREAL	0.0 bis 1.0E12	RES	1	Länge x (Quader) oder Radius (Kugel, Zylinder)
				2	Länge y (Quader) oder Höhe (Zylinder)
				3	Länge z (Quader)

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Variable "KinematicsZone" (Kinematik)

Die Variablenstruktur "<TO>.KinematicsZone[2..10].<Variablenname>" beinhaltet die Parameter zu Kinematikzonen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
KinematicsZone[2..10].	ARRAY [2..10] OF STRUCT				
Active	BOOL	-	RES	FALSE	Kinematikzone deaktiviert
				TRUE	Kinematikzone aktiviert
Valid	BOOL	-	RES	FALSE	Zone ist nicht definiert
				TRUE	Zone ist definiert
ReferenceSystem	DINT	0 bis 1	RES	Referenzkoordinatensystem für die Kinematikzone	
				0	FCS
				1	TCS
Frame.	STRUCT				
x	LREAL	-1.0E12 bis 1.0E12	RES	x-Koordinate	
y	LREAL	-1.0E12 bis 1.0E12	RES	y-Koordinate	
z	LREAL	-1.0E12 bis 1.0E12	RES	z-Koordinate	
a	LREAL	-180.0 bis 179.999	RES	A-Koordinate	
b	LREAL	-90.0 bis 90.0	RES	B-Koordinate	
c	LREAL	-180.0 bis 179.999	RES	C-Koordinate	
Geometry.	STRUCT				
Type	DINT	0 bis 2	RES	Zonengeometrie	
				0	Quader
				1	Kugel
				2	Zylinder
Parameter[1..3]	ARRAY [1..3] OF LREAL	0.0 bis 1.0E12	RES	1	Länge x (Quader) oder Radius (Kugel, Zylinder)
				2	Länge y (Quader) oder Höhe (Zylinder)
				3	Länge z (Quader)

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Variable "StatusPath" (Kinematik)

Die Variablenstruktur "<TO>.StatusPath.<Variablenname>" beinhaltet die Parameter der aktuellen Kinematikbewegung.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
StatusPath.	STRUCT				
CoordSystem	DINT	0 bis 3	RON	Koordinatensystem des aktiven Bewegungsauftrags	
				0	Weltkoordinatensystem
				1, 2, 3	Objektkoordinatensystem 1, 2, 3
Velocity	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Bahngeschwindigkeit (Sollwertbezug)	
Acceleration	LREAL	-1.0E12 bis 1.0E12	RON	Aktuelle Bahnbeschleunigung (Sollwertbezug)	
DynamicAdaption	DINT	0 bis 2	RON	Dynamikadaption	
				0	Keine Dynamikadaption
				1	Dynamikadaption mit Segmentierung der Bahn
				2	Dynamikadaption ohne Segmentierung der Bahn

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "TcpInWcs" (Kinematik)

Die Variablenstruktur "<TO>.TcpInWcs.<Variablenname>" beinhaltet die Parameter zum Werkzeugarbeitspunkt (TCP) im Weltkoordinatensystem (WCS).

Variablen

Variable	Datentyp	Werte	W	Beschreibung
TcpInWcs.	STRUCT			
x.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der Bahnkoordinate x
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	Position der Bahnkoordinate x
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der Bahnkoordinate x
y.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der Bahnkoordinate y
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	Position der Bahnkoordinate y
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der Bahnkoordinate y
z.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der Bahnkoordinate z
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	Position der Bahnkoordinate z
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der Bahnkoordinate z
a.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der Drehung A
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	Position der Drehung A
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der Drehung A

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "TcpInOcs" (Kinematik)

Die Variablenstruktur "<TO>.TcpInOcs.<Variablenname>" beinhaltet die Parameter zum Werkzeugarbeitspunkt (TCP) in den Objektkoordinatensystemen 1 bis 3 (OCS).

Variablen

Variable	Datentyp	Werte	W	Beschreibung
TcpInOcs[1..3].	ARRAY [1..3] OF STRUCT			
x.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der x-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der x-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
y.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der y-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der y-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
z.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der z-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der z-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3

Variable	Datentyp	Werte	W	Beschreibung
a.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der A-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	A-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der A-Koordinate des Werkzeugarbeitspunkts im Objektkoordinatensystem 1 bis 3

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "StatusOcsFrame" (Kinematik)

Die Variablenstruktur "<TO>.StatusOcsFrame.<Variablenname>" beinhaltet die Frames der Objektkoordinatensysteme 1 bis 3 (OCS) im Weltkoordinatensystem (WCS).

Variablen

Variable	Datentyp	Wertebereich	W	Beschreibung
StatusOcsFrame[1..3].	ARRAY [1..3] OF STRUCT			
x	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate im WCS
y	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate im WCS
z	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate im WCS
a	LREAL		RON	A-Koordinate im WCS
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		-180.0 bis 179.999		Bei Kinematiktyp "3D mit Orientierung"
b	LREAL		RON	B-Koordinate im WCS
		-180.0 bis 179.999		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-90.0 bis 90.0		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"
c	LREAL		RON	C-Koordinate im WCS
		0.0		Bei Kinematiktyp "2D"
		0.0		Bei Kinematiktyp "2D mit Orientierung"
		-180.0 bis 179.999		Bei Kinematiktyp "3D"
		0.0		Bei Kinematiktyp "3D mit Orientierung"

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "StatusKinematics" (Kinematik)

Die Variablenstruktur "<TO>.StatusKinematics.<Variablenname>" beinhaltet den Status der Kinematik.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
StatusKinematics.	STRUCT				
Valid	BOOL	-	RON	Gültigkeit der Transformationswerte	
				FALSE	nicht gültig
				TRUE	gültig
LinkConstellation	DWORD	0 bis n	RON	Gelenkstellung	
Bit 0	-	-	-	Winkel α1 der Achse A1 im vorderen/rückwärtigen Bereich (Standardbereich/Überkopfbereich)	
				0	Der Nullpunkt des FCS befindet sich im vorderen Bereich (Standardbereich) der Gelenkstellungsgeraden für die Achse A1. $\alpha 1 = \arctan(y_{FCS}/x_{FCS})$
				1	Der Nullpunkt des FCS befindet sich im rückwärtigen Bereich (Überkopfbereich) der Gelenkstellungsgeraden für die Achse A1. $\alpha 1 = -\arctan(y_{FCS}/x_{FCS})$
Bit 1	-	-	-	Winkel α2 der Achse A2 positiv/negativ unter Berücksichtigung der mechanischen Achskopplung	
				0	α2 positiv
				1	α2 negativ
Bit 2	-	-	-	Winkel α3 der Achse A3 positiv/negativ unter Berücksichtigung der mechanischen Achskopplung	
				0	α3 positiv
				1	α3 negativ

Siehe auch

Variablen Kinematiktransformation (Seite 2326)

Variable "FlangeInKcs" (Kinematik)

Die Variablenstruktur "<TO>.FlangeInKcs.<Variablenname>" beinhaltet die Parameter zum Flanschkoordinatensystem (FCS) im Kinematikkoordinatensystem (KCS).

Variablen

Variable	Datentyp	Werte	W	Beschreibung
FlangeInKcs.	STRUCT			
x.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der x-Koordinate des Flanschkoordinatensystems (FCS) im Kinematikkoordinatensystem (KCS)
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der x-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
y.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der y-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der y-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
z.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der z-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der z-Koordinate des Flanschkoordinatensystems im Kinematikkoordinatensystem
a.	STRUCT			
Acceleration	LREAL	-1.79769E308 bis 1.79769E308	RON	Beschleunigung der Drehung A des Flanschkoordinatensystems im Kinematikkoordinatensystem
Position	LREAL	-1.79769E308 bis 1.79769E308	RON	Position der Drehung A des Flanschkoordinatensystems im Kinematikkoordinatensystem
Velocity	LREAL	-1.79769E308 bis 1.79769E308	RON	Geschwindigkeit der Drehung A des Flanschkoordinatensystems im Kinematikkoordinatensystem

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "StatusTool" (Kinematik)

Die Variablenstruktur "<TO>.StatusTool.<Variablenname>" beinhaltet die Parameter zum Werkzeug.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
StatusTool.	STRUCT			
ActiveTool	DINT	1 bis 3	RON	Aktuell wirksames Werkzeug
Frame[1..1].	ARRAY [1..1] OF STRUCT			
x	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate des Werkzeugs 1
y	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate des Werkzeugs 1
z	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate des Werkzeugs 1
a	LREAL	-1.79769E308 bis 1.79769E308	RON	A-Koordinate des Werkzeugs 1

Siehe auch

Variablen Koordinatensysteme und Frames (Seite 2236)

Variable "StatusWorkspaceZone" (Kinematik)

Die Variablenstruktur "<TO>.StatusWorkspaceZone.<Variablenname>" beinhaltet den Status der Arbeitsraumzonen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
StatusWorkspaceZone[1..10].	ARRAY [1..10] OF STRUCT				
Active	BOOL	-	RON	FALSE	Arbeitsraumzone deaktiviert
				TRUE	Arbeitsraumzone aktiviert
Valid	BOOL	-	RON	FALSE	Zone ist nicht definiert
				TRUE	Zone ist definiert
Type	DINT	0 bis 2	RON	Typ der Arbeitsraumzone	
				0	Sperrzone
				1	Arbeitszone
				2	Meldezone
ReferenceSystem	DINT	0 bis 3	RON	Referenzkoordinatensystem für die Arbeitsraumzone	
				0	WCS
				1	OCS1
				2	OCS2
				3	OCS3
Frame.	STRUCT				
x	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate	
y	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate	
z	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate	
a	LREAL	-180.0 bis 179.999	RON	A-Koordinate	
b	LREAL	-90.0 bis 90.0		B-Koordinate	
c	LREAL	-180.0 bis 179.999	RON	C-Koordinate	
Geometry.	STRUCT		RON		
Type	DINT	0 bis 2	RON	Zonengeometrie	
				0	Quader
				1	Kugel
				2	Zylinder
Parameter[1..3]	ARRAY [1..3] OF LREAL	0.0 bis 1.0E12	RON	1	Länge x (Quader) oder Radius (Kugel, Zylinder)
				2	Länge y (Quader) oder Höhe (Zylinder)
				3	Länge z (Quader)

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Variable "StatusKinematicsZone" (Kinematik)

Die Variablenstruktur "<TO>.StatusKinematicsZone.<Variablenname>" beinhaltet den Status der Kinematikzonen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
StatusKinematicsZone[2..10].	ARRAY [2..10] OF STRUCT				
Active	BOOL	-	RON	FALSE	Kinematikzone deaktiviert
				TRUE	Kinematikzone aktiviert
Valid	BOOL	-	RON	FALSE	Zone ist nicht vorhanden
				TRUE	Zone ist vorhanden
ReferenceSystem	DINT	0 bis 1	RON	Referenzkoordinatensystem für die Kinematikzone	
				0	FCS
				1	TCS
Frame.	STRUCT				
x	LREAL	-1.79769E308 bis 1.79769E308	RON	x-Koordinate	
y	LREAL	-1.79769E308 bis 1.79769E308	RON	y-Koordinate	
z	LREAL	-1.79769E308 bis 1.79769E308	RON	z-Koordinate	
a	LREAL	-180.0 bis 179.999	RON	A-Koordinate	
b	LREAL	-90.0 bis 90.0	RON	B-Koordinate	
c	LREAL	-180.0 bis 179.999	RON	C-Koordinate	
Geometry.	STRUCT				
Type	DINT	0 bis 2	RON	Zonengeometrie	
				0	Quader
				1	Kugel
				2	Zylinder
Parameter[1..3]	ARRAY [1..3] OF LREAL	0.0 bis 1.0E12	RON	1	Länge x (Quader) oder Radius (Kugel, Zylinder)
				2	Länge y (Quader) oder Höhe (Zylinder)
				3	Länge z (Quader)

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Variable "StatusZoneMonitoring" (Kinematik)

Die Variablenstruktur "<TO>.StatusZoneMonitoring.<Variablenname>" beinhaltet den Status von verletzten Zonen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
StatusZoneMonitoring.	STRUCT			
WorkingZones	DWORD	-	RON	Anzeige von verletzten Arbeitszonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
BlockedZones	DWORD	-	RON	Anzeige von verletzten Sperrzonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
SignalizingZones	DWORD	-	RON	Anzeige von angefahrenen Meldezonen Die Bitnummern 1 bis 10 entsprechen den konfigurierten Zonennummern.
KinematicsZones	DWORD	-	RON	Anzeige von Kinematikzonen, die Arbeitsraumzonen verletzen Die Bitnummer 1 zeigt den Überwachungsstatus des TCP (Werkzeugarbeitspunkts) an. Die Bitnummern 2 bis 10 entsprechen den konfigurierten Zonennummern.

Siehe auch

Variablen Zonenüberwachung (Seite 2339)

Variable "StatusMotionQueue" (Kinematik)

Die Variablenstruktur "<TO>.StatusMotionQueue.<Variablenname>" beinhaltet den Status der Auftragskette.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
StatusMotionQueue.	STRUCT			
NumberOfCommands	DINT	-	RON	Anzahl der wartenden Aufträge in der Auftragskette

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Variable "KinematicsAxis" (Kinematik)

Die Variablenstruktur "<TO>.KinematicsAxis.<Variablenname>" beinhaltet die definierten Kinematikachsen.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
KinematicsAxis.	STRUCT			
A1	DB_ANY	-	RON	Technologieobjekt-Datenbaustein der Kinematikachse A1
A2	DB_ANY	-	RON	Technologieobjekt-Datenbaustein der Kinematikachse A2
A3	DB_ANY	-	RON	Technologieobjekt-Datenbaustein der Kinematikachse A3
A4	DB_ANY	-	RON	Technologieobjekt-Datenbaustein der Kinematikachse A4

Variable "Units" (Kinematik)

Die Variablenstruktur "<TO>.Units.<Variablenname>" beinhaltet die eingestellten technologischen Einheiten.

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
Units.	STRUCT				
LengthUnit	UDINT	0 bis 4294967295	RON	Einheit für die Position	
				1010	m
				1013	mm
				1011	km
				1014	µm
				1015	nm
				1019	in
				1018	ft
				1021	mi
LengthVelocityUnit	UDINT	0 bis 4294967295	RON	Einheit für die Geschwindigkeit	
				1062	mm/s
				1061	m/s
				1524	mm/min
				1525	m/min
				1526	mm/h
				1063	m/h
				1527	km/min
				1064	km/h
				1066	in/s
				1069	in/min
				1067	ft/s
				1070	ft/min
				1075	mi/h
AngleUnit	UDINT	0 bis 4294967295	RON	Einheit für die Position der Orientierungsachse	
				1004	rad
				1005	°
AngleVelocityUnit	UDINT	0 bis 4294967295	RON	Einheit für die Geschwindigkeit der Orientierungsachse	
				1521	°/s
				1522	°/min
				1086	rad/s
				1523	rad/min

Variable "StatusWord" (Kinematik)

Die Variable "<TO>.StatusWord" beinhaltet die Statusinformationen des Technologieobjekts.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "RestartActive") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Variablen

Variable	Datentyp	Werte	W	Beschreibung
StatusWord	DWORD	-	RON	Statusinformationen des Technologieobjekts

Variable	Datentyp	Werte	W	Beschreibung	
Bit 0	-	-	-	Reserviert	
Bit 1	-	-	-	"Error"	
				0	Kein Fehler ist vorhanden.
				1	Ein Fehler ist vorhanden.
Bit 2	-	-	-	"RestartActive"	
				0	Kein "Restart" ist aktiv.
				1	Ein "Restart" ist aktiv. Das Technologieobjekt wird neu initialisiert.
Bit 3	-	-	-	"OnlineStartValuesChanged"	
				0	"Restart"-Variablen unverändert
				1	Änderung an "Restart"-Variablen. Zur Übernahme der Änderungen muss das Technologieobjekt neu initialisiert werden.
Bit 4	-	-	-	"ControlPanelActive"	
				0	Die Kinematiksteuertafel ist deaktiviert.
				1	Die Kinematiksteuertafel ist aktiviert.
Bit 5	-	-	-	Reserviert	
Bit 6	-	-	-	"Done"	
				0	Ein Bewegungsauftrag ist in Bearbeitung bzw. die Kinematiksteuertafel ist aktiviert.
				1	Kein Bewegungsauftrag ist in Bearbeitung und die Kinematiksteuertafel ist deaktiviert.
Bit 7	-	-	-	Reserviert	
Bit 8	-	-	-	"LinearCommand"	
				0	Keine Linearbewegung ist aktiv.
				1	Eine Linearbewegung ist aktiv.
Bit 9	-	-	-	"CircularCommand"	
				0	Keine Kreisbewegung ist aktiv.
				1	Eine Kreisbewegung ist aktiv.
Bit 10	-	-	-	Reserviert	
Bit 11	-	-	-	Reserviert	
Bit 12	-	-	-	"ConstantVelocity"	
				0	Die Kinematik wird beschleunigt oder abgebremst.
				1	Die Sollgeschwindigkeit ist erreicht. Die Kinematik wird mit dieser Geschwindigkeit konstant verfahren oder befindet sich im Stillstand.
Bit 13	-	-	-	"Accelerating"	
				0	Kein Beschleunigungsvorgang ist aktiv.
				1	Ein Beschleunigungsvorgang ist aktiv.
Bit 14	-	-	-	"Decelerating"	
				0	Kein Verzögerungsvorgang ist aktiv.
				1	Ein Verzögerungsvorgang ist aktiv.
Bit 15	-	-	-	"OrientationMotion"	

Variable	Datentyp	Werte	W	Beschreibung	
				Orientierungsbewegung aktiv	
Bit 16	-	-	-	"Stopping"	
				0	Kein "MC_GroupStop"-Auftrag ist aktiv.
				1	Ein "MC_GroupStop"-Auftrag ist aktiv. Die Bewegung am Technologieobjekt Kinematik wird abgebrochen.
Bit 17	-	-	-	"Interrupted"	
				0	Die Bewegung am Technologieobjekt Kinematik ist nicht unterbrochen.
				1	Die Bewegung am Technologieobjekt Kinematik ist mit einem "MC_GroupInterrupt"-Auftrag unterbrochen. Die Bewegung kann mit einem "MC_GroupContinue"-Auftrag fortgesetzt werden.
Bit 18	-	-	-	"Blending"	
				0	Kein Überschleifsegment ist aktiv.
				1	Ein Überschleifsegment ist aktiv.
Bit 19 … Bit 31	-	-	-	Reserviert	

Siehe auch

Variablen Bewegungsführung und Dynamik (Seite 2331)

Funktionshandbuch "S7-1500T Motion Control V4.0 im TIA Portal V15" Kapitel "StatusWord, WarningWord und ErrorWord auswerten" (https://support.industry.siemens.com/cs/ww/de/view/109749263)

Variable "ErrorWord" (Kinematik)

Die Variable "<TO>.ErrorWord" zeigt Fehler am Technologieobjekt (Technologie-Alarme) an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 3 "CommandNotAccepted") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Variablen

Variable	Datentyp	Werte	W	Beschreibung
ErrorWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter sind inkonsistent bzw. unzulässig.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	"TransformationFault" Fehler bei der Kinematiktransformation
Bit 5	-	-	-	Reserviert
Bit 6	-	-	-	"DynamicError" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7 … Bit 31	-	-	-	Reserviert

Siehe auch

Funktionshandbuch "S7-1500T Motion Control V4.0 im TIA Portal V15" Kapitel "StatusWord, WarningWord und ErrorWord auswerten" (https://support.industry.siemens.com/cs/ww/de/view/109749263)

Variable "ErrorDetail" (Kinematik)

Die Variablenstruktur "<TO>.ErrorDetail.<Variablenname>" beinhaltet die Alarmnummer und die wirksame lokale Alarmreaktion zum aktuell am Technologieobjekt anstehenden Technologie-Alarm.

Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang "Technologie-Alarme (Seite 2418)".

Variablen

Variable	Datentyp	Werte	W	Beschreibung	
ErrorDetail.	STRUCT				
Number	UDINT	-	RON	Alarmnummer	
Reaction	DINT	0, 11, 12	RON	Wirksame Alarmreaktion	
				0	Keine Reaktion (nur Warnungen)
				11	Stopp mit maximalen Dynamikwerten der Kinematik
				12	Stopp mit maximalen Dynamikwerten der Achsen

Variable "WarningWord" (Kinematik)

Die Variable "<TO>.WarningWord" zeigt am Technologieobjekt anstehende Warnungen an.

Hinweise zur Auswertung der einzelnen Bits (z. B. Bit 2 "UserFault") finden Sie im Kapitel "StatusWord, ErrorWord und WarningWord auswerten".

Variablen

Variable	Datentyp	Werte	W	Beschreibung
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Ein systeminterner Fehler ist aufgetreten.
Bit 1	-	-	-	"ConfigFault" Konfigurationsfehler Einer oder mehrere Konfigurationsparameter werden intern angepasst.
Bit 2	-	-	-	"UserFault" Fehler im Anwenderprogramm an einer Motion Control-Anweisung oder deren Verwendung
Bit 3	-	-	-	"CommandNotAccepted" Befehl nicht ausführbar Eine Motion Control-Anweisung kann nicht ausgeführt werden, weil notwendige Voraussetzungen nicht erfüllt sind.
Bit 4	-	-	-	Reserviert
Bit 5	-	-	-	Reserviert
Bit 6	-	-	-	"DynamicWarning" Vorgaben von Dynamikwerten werden auf zulässige Werte beschränkt.
Bit 7 … Bit 31	-	-	-	Reserviert

Siehe auch

Funktionshandbuch "S7-1500T Motion Control V4.0 im TIA Portal V15" Kapitel "StatusWord, WarningWord und ErrorWord auswerten" (https://support.industry.siemens.com/cs/ww/de/view/109749263)

Variable "ControlPanel" (Kinematik)

Die Variablenstruktur "<TO>.ControlPanel.<Variablenname>" beinhaltet keine anwenderrelevanten Daten. Diese Variablenstruktur wird intern verwendet.

Variablen

Variable	Datentyp	Werte	W	Beschreibung
ControlPanel.	STRUCT			
Input.	STRUCT			
TimeOut	LREAL	100 bis 60000	DIR	-
EsLifeSign	UDINT	-	DIR	-
Command[1..2].	ARRAY [1..2] OF STRUCT			
ReqCounter	UDINT	-	DIR	-
Type	UDINT	-	DIR	-
Position[1..4]	ARRAY [1..4] OF LREAL	-	DIR	-
Velocity[1..4]	ARRAY [1..4] OF LREAL	-	DIR	-
Acceleration[1..4]	ARRAY [1..4] OF LREAL	-	DIR	-
Deceleration[1..4]	ARRAY [1..4] OF LREAL	-	DIR	-
Jerk[1..4]	ARRAY [1..4] OF LREAL	-	DIR	-
Param[1..9]	ARRAY [1..9] OF LREAL	-	DIR	-
CoordinateSystem	UDINT	-	DIR	-
ToolNumber	UDINT	-	DIR	-
Output.	STRUCT			
RtLifeSign	UDINT	-	RON	-
Command[1..2].	ARRAY [1..2] OF STRUCT	-		
AckCounter	UDINT	-	RON	-
Error	BOOL	-	RON	-
ErrorID	WORD	-	RON	-
Done	BOOL	-	RON	-
Aborted	BOOL	-	RON	-

Technologie-Alarme

Übersicht

Folgende Tabelle zeigt eine Übersicht über die Technologie-Alarme und die entsprechenden Alarmreaktionen. Werten Sie beim Auftreten eines Technologie-Alarms den gesamten angezeigten Alarmtext aus, um die genaue Ursache zu finden.

Legende

Nr.	Nummer des Technologie-Alarms (entspricht <TO>.ErrorDetail.Number)
Reaktion	Wirksame Alarmreaktion (entspricht <TO>.ErrorDetail.Reaction)
Fehlerbit	Beim Auftreten des Technologie-Alarms gesetztes Bit in <TO>.ErrorWord Eine Beschreibung der Bits finden Sie im Anhang (Seite 2413).
Warnungsbit	Beim Auftreten des Technologie-Alarms gesetztes Bit in <TO>.WarningWord Eine Beschreibung der Bits finden Sie im Anhang (Seite 2415).
Restart	Zum Quittieren des Technologie-Alarms muss das Technologieobjekt neu initialisiert werden (Restart).
Diagnosepuffer	Der Alarm wird im Diagnosepuffer eingetragen.
Alarmtext	Angezeigter Alarmtext (eingeschränkt)

Liste der Technologie-Alarme

Nr.	Reaktion	Fehler-bit	Warnungs-bit	Restart	Diagnose-puffer	Alarmtext
101	Stopp mit maximalen Dynamikwerten der Achsen	X1	-	X	X	Konfigurationsfehler.
201	Stopp mit maximalen Dynamikwerten der Achsen	X0	-	X	X	Interner Fehler.
202	Stopp mit maximalen Dynamikwerten der Achsen	X0	-	X	-	Interner Konfigurationsfehler.
203	Stopp mit maximalen Dynamikwerten der Achsen	X0	-	X	-	Interner Fehler.
204	Stopp mit maximalen Dynamikwerten der Achsen	X0	-	-	-	Inbetriebnahmefehler.
304	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Grenzwert der Geschwindigkeit ist null.
305	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	• Grenzwert der Beschleunigung ist null. • Grenzwert der Verzögerung ist null.

Nr.	Reaktion	Fehler-bit	Warnungs-bit	Restart	Diagnose-puffer	Alarmtext
306	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Grenzwert des Rucks ist null.
501	Keine Reaktion (nur Warnungen)	-	X6	-	-	Programmierte Geschwindigkeit wird begrenzt.
502	Keine Reaktion (nur Warnungen)	-	X6	-	-	• Programmierte Beschleunigung wird begrenzt. • Programmierte Verzögerung wird begrenzt.
503	Keine Reaktion (nur Warnungen)	-	X6	-	-	Programmierter Ruck wird begrenzt.
561	Keine Reaktion (nur Warnungen)	-	X6	-	-	Programmierte Geschwindigkeit der Orientierungsbewegung wird begrenzt.
562	Keine Reaktion (nur Warnungen)	-	X6	-	-	• Programmierte Beschleunigung der Orientierungsbewegung wird begrenzt. • Programmierte Verzögerung der Orientierungsbewegung wird begrenzt.
563	Keine Reaktion (nur Warnungen)	-	X6	-	-	Programmierter Ruck der Orientierungsbewegung wird begrenzt.
801	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Kinematikachse nicht bereit.
802	Stopp mit maximalen Dynamikwerten der Achsen	X3	-	-	-	Berechnung des Geometrieelements nicht möglich.
803	Stopp mit maximalen Dynamikwerten der Achsen	X4	-	-	-	Fehler bei der Berechnung der Transformation.
804	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Die Kinematikbewegung kann am Ende nicht gestoppt werden.
805	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Begrenzung der Bahndynamik durch die Dynamik der Kinematikachsen.
806	Stopp mit maximalen Dynamikwerten der Kinematik	X2	-	-	-	Kollision mit Arbeits- oder Sperrzonen festgestellt.
807	Keine Reaktion (nur Warnungen)	-	X2	-	-	Kollision mit Meldezonen festgestellt.
808	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Uneindeutigkeit durch mehrere aktive Arbeitszonen.
809	Stopp mit maximalen Dynamikwerten der Achsen	X2	-	-	-	Bahndynamikbegrenzung durch Dynamik der Orientierungsbewegung fehlerhaft.

Alarmreaktion

Ein Technologie-Alarm beinhaltet immer eine Alarmreaktion, welche die Auswirkung auf das Technologieobjekt beschreibt. Die Alarmreaktion ist systemseitig vorgegeben.

Folgende Tabelle zeigt mögliche Alarmreaktionen:

Alarmreaktion	Beschreibung
Keine Reaktion (nur Warnungen) <TO>.ErrorDetail.Reaction = 0	Die Bearbeitung der Motion Control-Aufträge wird fortgesetzt. Die laufende Bewegung der Kinematik kann beeinflusst werden, z. B. durch Begrenzung der aktuellen Dynamikwerte auf die konfigurierten Grenzwerte.
Stopp mit maximalen Dynamikwerten der Kinematik <TO>.ErrorDetail.Reaction = 11	Laufende und anstehende Bewegungsaufträge werden abgebrochen. Die Kinematik wird mit den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Dynamik" konfigurierten maximalen Dynamikwerten abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt.
Stopp mit maximalen Dynamikwerten der Achsen <TO>.ErrorDetail.Reaction = 12	Laufende und anstehende Bewegungsaufträge werden abgebrochen. Die Achsen werden mit den unter "Technologieobjekt > Konfiguration > Erweiterte Parameter > Begrenzungen > Dynamikgrenzen" konfigurierten maximalen Dynamikwerten abgebremst und zum Stillstand gebracht. Dabei wird der konfigurierte maximale Ruck berücksichtigt.

Technologie-Alarme 101

Technologie-Alarm 101

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: erforderlich

Alarmtext	Abhilfe
Konfigurationsfehler.	
Wert in <Variable> unzulässig.	Passen Sie den angegebenen Wert an.
Verschaltung Achse <Nr.> fehlt.	Verschalten Sie die Achse neu.
Verschaltung Orientierungsachse fehlt.	
Delta-2D-Picker: Keine Bildung einer geschlossenen parallelen Struktur.	Passen Sie die Geometriedaten der Mechanik an.
Delta-3D-Picker: Keine Bildung einer geschlossenen parallelen Struktur.	
Delta-3D-Picker: Winkelversatz lässt keinen dritten Arm zu.	
Rollen-Picker: Radius fehlerhaft.	
Gelenkabstände unzulässig.	
Grenzen bei Vorgabe von Orientierungen nicht eingehalten.	Passen Sie den entsprechenden Wert an.

Technologie-Alarme 201 - 204

Technologie-Alarm 201

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: erforderlich

Alarmtext	Abhilfe
Interner Fehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 202

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: erforderlich

Alarmtext	Abhilfe
Interner Konfigurationsfehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 203

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: erforderlich

Alarmtext	Abhilfe
Interner Algorithmenfehler.	Wenden Sie sich an den Kundendienst.

Technologie-Alarm 204

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: erforderlich

Alarmtext	Abhilfe
Inbetriebnahmefehler.	
Verbindung zum TIA Portal unterbrochen.	Prüfen Sie die Verbindungseigenschaften.

Technologie-Alarme 304 - 306

Technologie-Alarm 304

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Der Grenzwert der Geschwindigkeit ist null.	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die Geschwindigkeit (DynamicLimits.Path.Velocity) ein.

Technologie-Alarm 305

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext		Abhilfe
Der Grenzwert der Beschleunigung/Verzögerung ist null.		
	Beschleunigung	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die Beschleunigung (DynamicLimits.Path.Acceleration) ein.
	Verzögerung	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für die Verzögerung (DynamicLimits.Path.Deceleration) ein.

Technologie-Alarm 306

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Der Grenzwert des Rucks ist null.	Geben Sie in den Dynamikgrenzen einen Wert ungleich null für den Ruck (DynamicLimits.Path.Jerk) ein.

Technologie-Alarme 501 - 563

Technologie-Alarm 501

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierte Geschwindigkeit wird begrenzt.	• Prüfen Sie den Wert für die Geschwindigkeit an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 502

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext		Abhilfe
Programmierte Beschleunigung/Verzögerung wird begrenzt.		
	Beschleunigung	• Prüfen Sie den Wert für die Beschleunigung an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.
	Verzögerung	• Prüfen Sie den Wert für die Verzögerung an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 503

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierter Ruck wird begrenzt.	• Prüfen Sie den Wert für den Ruck an der Motion Control-Anweisung. • Prüfen Sie die Konfiguration der Dynamikgrenzen.

Technologie-Alarm 561

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierte Geschwindigkeit der Orientierungsbewegung wird begrenzt.	• Prüfen Sie die Konfiguration der Geschwindigkeit der Orientierungsbewegung.

Technologie-Alarm 562

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext		Abhilfe
Programmierte Beschleunigung/Verzögerung der Orientierungsbewegung wird begrenzt.		
	Beschleunigung	• Prüfen Sie die Konfiguration der Beschleunigung der Orientierungsbewegung.
	Verzögerung	• Prüfen Sie die Konfiguration der Verzögerung der Orientierungsbewegung.

Technologie-Alarm 563

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext	Abhilfe
Programmierter Ruck der Orientierungsbewegung wird begrenzt.	• Prüfen Sie die Konfiguration des Rucks der Orientierungsbewegung.

Technologie-Alarme 801 - 809

Technologie-Alarm 801

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Kinematikachse <Nr.> nicht bereit.	
Achse nicht freigegeben.	Geben Sie das Technologieobjekt frei.
Achsbefehl programmiert.	Um einen weiteren Kinematikbefehl absetzen zu können, setzen Sie die angegebene Achse in den Stillstand.
Alarm an der Achse.	Prüfen und quittieren Sie die Technologie-Alarme der angegebenen Kinematikachse.

Technologie-Alarm 802

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Berechnung des Geometrieelements nicht möglich.	
Radius bei "CircMode" = 2 kleiner als die Hälfte der Strecke.	Passen Sie den Radius an.
Start-, Zwischen- oder Endpunkt sind bei "CircMode" = 0 identisch.	Geben Sie unterschiedliche Werte für Start-, Zwischen- und Endpunkt an.
Zwischenpunkt kann bei "CircMode" = 0 nicht erreicht werden.	Passen Sie den Zwischenpunkt an.
Start- und Endpunkt sind bei der Angabe von "CircMode" = 2 sowie "PathChoice" = 2 oder 3 identisch.	Definieren Sie unterschiedliche Start- und Endpunkte.
Dynamikadaption konnte nicht durchgeführt werden.	Schalten Sie die Dynamikadaption aus.

Technologie-Alarm 803

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Fehler bei Berechnung der Transformation.	
Transformation der Achs- in Kinematikkoordinaten führte zum Fehler.	• Korrigieren Sie Ihre vorgegebene Bewegung in Bezug auf den Gelenkstellungsraum und die Transformationsbereiche. • Positionieren Sie die Kinematikachsen mit Einzelachsbewegungen in einen zulässigen Transformationsbereich. • Bei Anwendertransformation: Prüfen Sie die Berechnung im MC-Transformation [OB98].
Bei anwenderdefinierter Kinematik: "FunctionResult“ des MC-Transformation [OB98]	
Transformation der Kinematik- in Achskoordinaten führte zum Fehler.	
Bei vordefinierter Kinematik: Zusatzinfo:	
0 Kartesische Position nicht erreichbar	
1 Singuläre Stellung	
Bei anwenderdefinierter Kinematik: "FunctionResult“ des MC-Transformation [OB98]	

Technologie-Alarm 804

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Kinematikbewegung kann am Ende nicht gestoppt werden.	Sorgen Sie für eine ausreichend lange Bahn.

Technologie-Alarm 805

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Begrenzung der Bahndynamik durch Achsdynamik fehlerhaft.	
Bahngeschwindigkeit wird auf null begrenzt.	Konfigurieren Sie eine größere Bahngeschwindigkeit an der Kinematikachse.
Bahnbeschleunigung bzw. -verzögerung wird auf null begrenzt.	Konfigurieren Sie eine größere Bahnbeschleunigung bzw. -verzögerung an der Kinematikachse.

Technologie-Alarm 806

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Kinematik

Restart: nicht erforderlich

Alarmtext	Abhilfe
Kollision mit Arbeits- oder Sperrzonen festgestellt.	Fahren Sie das Kinematikobjekt aus der Arbeits- oder Sperrzone.

Technologie-Alarm 807

Alarmreaktion: Keine Reaktion (nur Warnungen)

Restart: nicht erforderlich

Alarmtext	Abhilfe
Kollision mit Meldezonen festgestellt.	-

Technologie-Alarm 808

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Uneindeutigkeit durch mehrere aktive Arbeitszonen.	
<Nummer der aktuell aktiven Zone>	Aktivieren Sie lediglich eine Arbeitszone.

Technologie-Alarm 809

Alarmreaktion: Stopp mit maximalen Dynamikwerten der Achsen

Restart: nicht erforderlich

Alarmtext	Abhilfe
Bahndynamikbegrenzung durch Dynamik der Orientierungsbewegung fehlerhaft.	
Geschwindigkeit wird auf null begrenzt.	Konfigurieren Sie eine größere Maximalgeschwindigkeit an den an der Orientierungsbewegung beteiligten Achsen.
Beschleunigung/Verzögerung wird auf null begrenzt.	Konfigurieren Sie eine größere Maximalbeschleunigung bzw. -verzögerung an den an der Orientierungsbewegung beteiligten Achsen.

Fehlerkennung (Kinematik)

Fehler an Motion Control-Anweisungen werden über die Parameter "Error" und "ErrorID" gemeldet.

Unter folgenden Bedingungen wird an der Motion Control-Anweisung "Error" = TRUE und "ErrorID" = 16#8xxx angezeigt:

- Unzulässiger Status des Technologieobjekts, der die Ausführung des Auftrags verhindert.
- Unzulässige Parametrierung der Motion Control-Anweisung, welche die Ausführung des Auftrags verhindert.
- Infolge der Alarmreaktion eines Fehlers am Technologieobjekt.

Die folgenden Tabellen zeigen eine Auflistung aller "ErrorIDs", die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

16#0000 - 16#800F

ErrorID	Beschreibung	Abhilfe
16#0000	Kein Fehler	-
16#8001	Während der Bearbeitung der Motion Control-Anweisung ist ein Technologie-Alarm (Fehler am Technologieobjekt) aufgetreten.	Im Technologie-Datenbaustein wird an der Variable "ErrorDetail.Number" eine Fehlermeldung ausgegeben. Eine Liste der Technologie-Alarme und Alarmreaktionen finden Sie im Anhang Technologie-Alarme (Seite 2420).
16#8002	Unzulässige Angabe des Technologieobjekts	Prüfen Sie die Angabe des Technologieobjekts am Parameter "Axis" oder "AxesGroup". Ein Technologieobjekt Kinematik können Sie nur am Parameter "AxesGroup" verwenden.
16#8003	Unzulässige Geschwindigkeitsangabe	Geben Sie am Parameter "Velocity" einen zulässigen Wert für die Geschwindigkeit an.
16#8004	Unzulässige Beschleunigungsangabe	Geben Sie am Parameter "Acceleration" einen zulässigen Wert für die Beschleunigung an.
16#8005	Unzulässige Verzögerungsangabe	Geben Sie am Parameter "Deceleration" einen zulässigen Wert für die Verzögerung an.
16#8006	Unzulässige Ruckangabe	Geben Sie am Parameter "Jerk" einen zulässigen Wert für den Ruck an.
16#8007	Unzulässige Richtungsangabe	Geben Sie am Parameter "DirectionA" einen zulässigen Wert für die Bewegungsrichtung an.
16#8008	Unzulässige Angabe der relativen Zielkoordinate	Geben Sie am Parameter "Distance" zulässige Werte für die relative Zielkoordinate an.
16#8009	Unzulässige Angabe der absoluten Zielkoordinate	Geben Sie am Parameter "Position" einen zulässigen Wert für die absolute Zielkoordinate an.
16#800A	Unzulässige Modusangabe	Geben Sie am Parameter "Mode" einen zulässigen Wert für den Modus an.
16#800D	Der Auftrag ist im aktuellen Zustand nicht erlaubt. Restart wird ausgeführt.	Während ein Restart ausgeführt wird, kann das Technologieobjekt keine Aufträge ausführen. Bei einem Restart des Technologieobjekts Kinematik werden zunächst alle verschalteten Achsen reinitialisiert. Anschließend wird das Technologieobjekt Kinematik reinitialisiert. Dies kann bis zu einer Sekunde dauern. Warten Sie, bis der Restart des Technologieobjekts abgeschlossen ist.
16#800F	Der Auftrag ist nicht ausführbar, da das Technologieobjekt gesperrt ist.	Geben Sie das Technologieobjekt mit "MC_Power.Enable" = TRUE frei. Starten Sie den Auftrag erneut.

16#8010 - 16#807F

ErrorID	Beschreibung	Abhilfe
16#8012	Der Auftrag ist nicht ausführbar, da die Kinematiksteuertafel aktiv ist.	Geben Sie die Steuerungshoheit an Ihr Anwenderprogramm zurück. Starten Sie den Auftrag erneut.
16#8014	Kein interner Auftragsspeicher verfügbar.	Die maximal mögliche Anzahl an Motion Control-Aufträgen ist erreicht. Reduzieren Sie die Anzahl der zu bearbeitenden Aufträge (Parameter "Execute" = FALSE).
16#8015	Fehlerquittierung mit "MC_Reset" ist nicht möglich. Fehler bei der Konfiguration des Technologieobjekts.	Prüfen Sie die Konfiguration des Technologieobjekts.

16#80A0 - 16#8FFF

ErrorID	Beschreibung	Abhilfe
16#80B1	Unzulässige Angabe des Koordinatensystems	Geben Sie am Parameter "CoordSystem" einen zulässigen Wert für das Koordinatensystem an.
16#80B2	Unzulässige Angabe des Bewegungsübergangs	Geben Sie am Parameter "BufferMode" einen zulässigen Wert für den Bewegungsübergang an.
16#80B3	Unzulässige Angabe des Überschleifabstands	Geben Sie am Parameter "TransitionParameter" einen zulässigen Wert für den Überschleifabstand an.
16#80B5	Unzulässige Angabe der Dynamikadaption	Geben Sie am Parameter "DynamicAdaption" einen zulässigen Wert für die Dynamikadaption an.
16#80B6	Unzulässige Angabe für die Definition der Kreisbahn	Geben Sie am Parameter "CircMode" einen zulässigen Wert für die Definition der Kreisbahn an.
16#80B7	Unzulässige Angabe für den Kreisbahn-Hilfspunkt	Geben Sie am Parameter "AuxPoint" einen zulässigen Wert für den Kreisbahn-Hilfspunkt an.
16#80B8	Unzulässige Angabe der Zielposition	Geben Sie am Parameter "EndPoint" einen zulässigen Wert für die Zielposition an.
16#80B9	Unzulässige Angabe der Orientierung der Kreisbahn	Geben Sie am Parameter "PathChoice" einen zulässigen Wert für die Orientierung der Kreisbahn an.
16#80BA	Unzulässige Angabe für die Hauptebene der Kreisbahn	Geben Sie am Parameter "CirclePlane" einen zulässigen Wert für die Hauptebene der Kreisbahn an.
16#80BB	Unzulässige Radiusangabe	Geben Sie am Parameter "Radius" einen zulässigen Wert für den Radius der Kreisbewegung an.
16#80BC	Unzulässige Winkelangabe	Geben Sie am Parameter "Arc" einen zulässigen Wert für den Winkel der Kreisbewegung an.
16#80C1	Unzulässige Angabe des Zonentyps	Geben Sie am Parameter "ZoneType" einen zulässigen Wert für den Zonentyp an.
16#80C2	Unzulässige Angabe der Zonennummer	Geben Sie am Parameter "ZoneNumber" einen zulässigen Wert für die Zonennummer an.
16#80C3	Unzulässige Angabe des Bezugssystems	Geben Sie am Parameter "ReferenceSystem" einen zulässigen Wert für das Bezugssystem an.
16#80C4	Unzulässige Koordinatenangabe	Geben Sie am Parameter "Frame" zulässige Werte für die Koordinaten an.
16#80C5	Unzulässige Angabe der Zonengeometrie	Geben Sie am Parameter "GeometryType" einen zulässigen Wert für die Zonengeometrie an.

ErrorID	Beschreibung	Abhilfe
16#80C6	Unzulässige Angabe der geometrischen Parameter	Geben Sie am Parameter "GeometryParameter" zulässige Werte für die geometrischen Parameter an.
16#80C7	Die Zone ist nicht definiert.	Definieren Sie eine Arbeitsraumzone über den Auftrag "MC_DefineWorkspaceZone" bzw. eine Kinematikzone über den Auftrag "MC_DefineKinematicsZone".
16#80C8	Ein Werkzeug kann während einer Bewegung nicht neu definiert werden.	Beenden Sie die aktive Bewegung. Starten Sie den Auftrag "MC_DefineTool" erneut.
	Ein aktives Werkzeug kann während einer Bewegung nicht gewechselt werden.	Beenden Sie die aktive Bewegung. Starten Sie den Auftrag "MC_SetTool" erneut.
16#80CA	Unzulässige Angabe der Werkzeugnummer	Geben Sie am Parameter "ToolNumber" einen zulässigen Wert für die Werkzeugnummer an.
16#80CB	Unzulässige Angabe des Objektkoordinatensystems	Geben Sie am Parameter "OcsNumber" einen zulässigen Wert für das Objektkoordinatensystem an.
16#80CC	Der Auftrag ist nicht ausführbar, da an einer Kinematikachse eine Einzelachsbewegung aktiv ist.	Beenden Sie die aktuelle Einzelachsbewegung. Starten Sie den Auftrag erneut.
16#80CD	Der Auftrag ist nicht ausführbar, da ein "MC_GroupStop"-Auftrag aktiv ist.	Setzen Sie den Parameter "MC_GroupStop.Execute" auf FALSE. Starten Sie den Auftrag erneut.
16#80CE	Die Auftragskette ist ausgelastet.	Die maximal möglichen Motion Control-Aufträge wurden abgesetzt.
16#8FFF	Unspezifizierter Fehler	Wenden Sie sich an Ihren Siemens-Ansprechpartner in den für Sie zuständigen Vertretungen und Geschäftsstellen. Ihren Ansprechpartner bei Industry Automation and Drive Technologies finden Sie unter: http://www.siemens.com/automation/partner (http://www.siemens.com/automation/partner)

6.2 Motion Control (S7-300, S7-400)

6.2.1 Grundlagen zur Wegerfassung

6.2.1.1 Was ist Wegerfassung?

Wegerfassung

Wegerfassung ist das Erfassen von Istwerten, die auf Längeneinheiten normiert sind.

Abhängig vom eingesetzten Geber gibt es folgende Unterschiede:

- Mit Inkrementalgeber
  Die Baugruppe zählt die Signale des Inkrementalgebers. Um einen Bezugspunkt für die Geberposition zu schaffen, ist vor dem Wegerfassen eine Referenzpunktfahrt erforderlich. Der Zähler kann zum Beispiel an einer Referenzmarke auf "0" gesetzt werden.
  Nach der Referenzpunktfahrt (mit synchronisierter Achse) ist der Zählerstand ein Maß für die aktuelle Position.
- Mit Absolutwertgeber
  Eine Referenzpunktfahrt ist nicht erforderlich. Das ausgegebene Signal entspricht einer festen Position.

Die ermittelten Werte können unmittelbar im Anwenderprogramm verarbeitet werden. Damit ist eine direkte Reaktion auf Geberwerte aus bewegten Systemen möglich.

6.2.1.2 Geberwerterfassung

Beschreibung

Der Absolutwertgeber überträgt seine Geberwerte in Telegrammen zur Baugruppe. Die Übertragung der Telegramme wird von der Baugruppe veranlasst.

- Im nicht taktsynchronen Betrieb erfolgt die Geberwerterfassung freilaufend.
- Im taktsynchronen Betrieb erfolgt die Geberwerterfassung synchron zum Busyklus zu jedem T_i.

Freilaufende Geberwerterfassung

Die Baugruppe veranlasst die Übertragung eines Telegramms jeweils nach Ablauf der parametrierten Monoflopzeit.

Asynchron zu diesen freilaufenden Telegrammen verarbeitet die Baugruppe den erfassten Geberwert im Zyklus seiner Aktualisierungsrate.

Dadurch ergeben sich bei der freilaufenden Geberwerterfassung unterschiedlich alte Geberwerte. Die Differenz zwischen maximalem und minimalem Alter ist der Jitter.

Taktsynchrone Geberwerterfassung

Taktsynchrone Geberwerterfassung stellt sich automatisch dann ein, wenn das Bussystem und die Peripheriemodule synchronisiert arbeiten.

Die Baugruppe veranlasst die Übertragung eines Telegramms in jedem Buszyklus zum Zeitpunkt T_i.

Taktsynchron zum Buszyklus verarbeitet die Baugruppe den übertragenen Geberwert.

6.2.1.3 Gray-/Dualwandler

Gray-/Dual-Wandler

Mit dem Parameter "Gray-/Dual-Wandler" bestimmen Sie die Behandlung des vom Geber gelieferten Codes:

- In der Einstellung "Gray" wird der vom Absolutwertgeber in Graycode gelieferte Geberwert in Dualcode umgewandelt.
- In der Einstellung "Dual" bleibt der gelieferte Geberwert unverändert.

Hinweis

Haben Sie die Einstellung "Gray" gewählt, wandelt die Baugruppe immer den gesamten Geberwert um (13 bis 25 Bit). Dadurch beeinflussen vorangestellte Sonderbits den Geberwert und nachgestellte Bits können unter Umständen verfälscht werden.

6.2.1.4 Geberwert und Normierung

Geberwert und Normierung

Der übertragene Geberwert enthält die Geberposition des Absolutwertgebers. Je nach verwendetem Geber werden neben der Geberposition weitere Bits übertragen, die sich vor und nach der Geberposition befinden.

Damit die Baugruppe die Geberposition ermitteln kann, braucht sie folgende Angaben:

- Geberart
- Anzahl nachgestellter Bits
- Inkremente/Geberumdrehung

Mit der Normierung legen Sie die Darstellung des Geberwertes in der Rückmeldeschnittstelle fest.

- Mit dem Eintrag "Anzahl nachgestellter Bits = 1, 2 … 12" (bei SM 338) bzw. "Normierung ein" (bei ET 200S 1SSI) legen Sie fest, dass nachgestellte, nicht relevante Bits im Geberwert weggeschoben werden. Der Geberwert wird rechtsbündig im Adressbereich angeordnet (siehe nachfolgendes Beispiel).
- Mit dem Eintrag "Anzahl nachgestellter Bits = 0" (bei SM 338) bzw. "Normierung aus" (bei ET 200S 1SSI) legen Sie fest, dass nachgestellte Bits erhalten bleiben und zur Auswertung bereitstehen.
  Dies kann sinnvoll sein, wenn Sie einen Absolutwertgeber einsetzen, der in den nachgestellten Bits Informationen überträgt (siehe Herstellerangaben) und Sie diese auswerten möchten. Beachten Sie dazu auch die Informationen zum Gray-/Dualwandler (Seite 2431).

Beispiel zur Normierung eines Geberwertes

Vorgaben:

Sie verwenden einen Singleturn-Geber mit 2^9 Schritten (entspricht 9 Bit) = 512 Inkremente/ Geberumdrehung (Auflösung / 360°) mit folgender Parametrierung:

- Geberart: SSI-13 Bit
- Anzahl nachgestellter Bits: 4 Stellen
- Inkremente/Geberumdrehung: 512

Das folgende Bild zeigt die Lage der relevanten Bits vor und nach der Normierung.

Vor der Normierung: zyklisch erfasster Geberwert 100

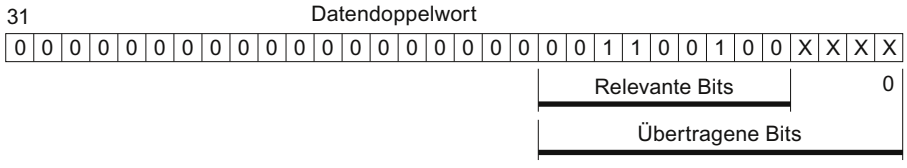

Nach der Normierung: Geberwert 100

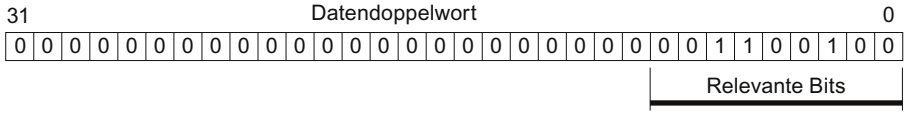

Ergebnis: Bits 0 bis 3 (4 Stellen, mit “x” gekennzeichnet) sind entfallen.

6.2.2 SM 338 einsetzen

6.2.2.1 Anschlussbelegung des Frontsteckers der SM 338

Verdrahtungsregeln

Beachten Sie bei der Verdrahtung der Baugruppe die folgenden Regeln:

- Die Masse der Geberversorgung ist potenzialgebunden zur Masse der CPU. Verbinden Sie deshalb Pin 2 der SM 338 (M) niederohmig mit der Masse der CPU.
- Die Geberleitungen (Pin 3 bis 14) müssen geschirmt und paarweise verdrillt sein. Legen Sie den Schirm beidseitig auf.
  Zur Schirmauflage an der SM 338 verwenden Sie das Schirmauflageelement (Artikelnummer: 6ES7390-5AA00-0AA0).
- Wenn Sie den maximalen Ausgangsstrom (900 mA) der Geberversorgung überschreiten, dann müssen Sie eine externe Spannungsversorgung anschließen.

Anschlussbelegung des Frontsteckers und Prinzipschaltbild

Das folgende Bild zeigt die Anschlussbelegung des Frontsteckers der SM 338 sowie das Prinzipschaltbild.

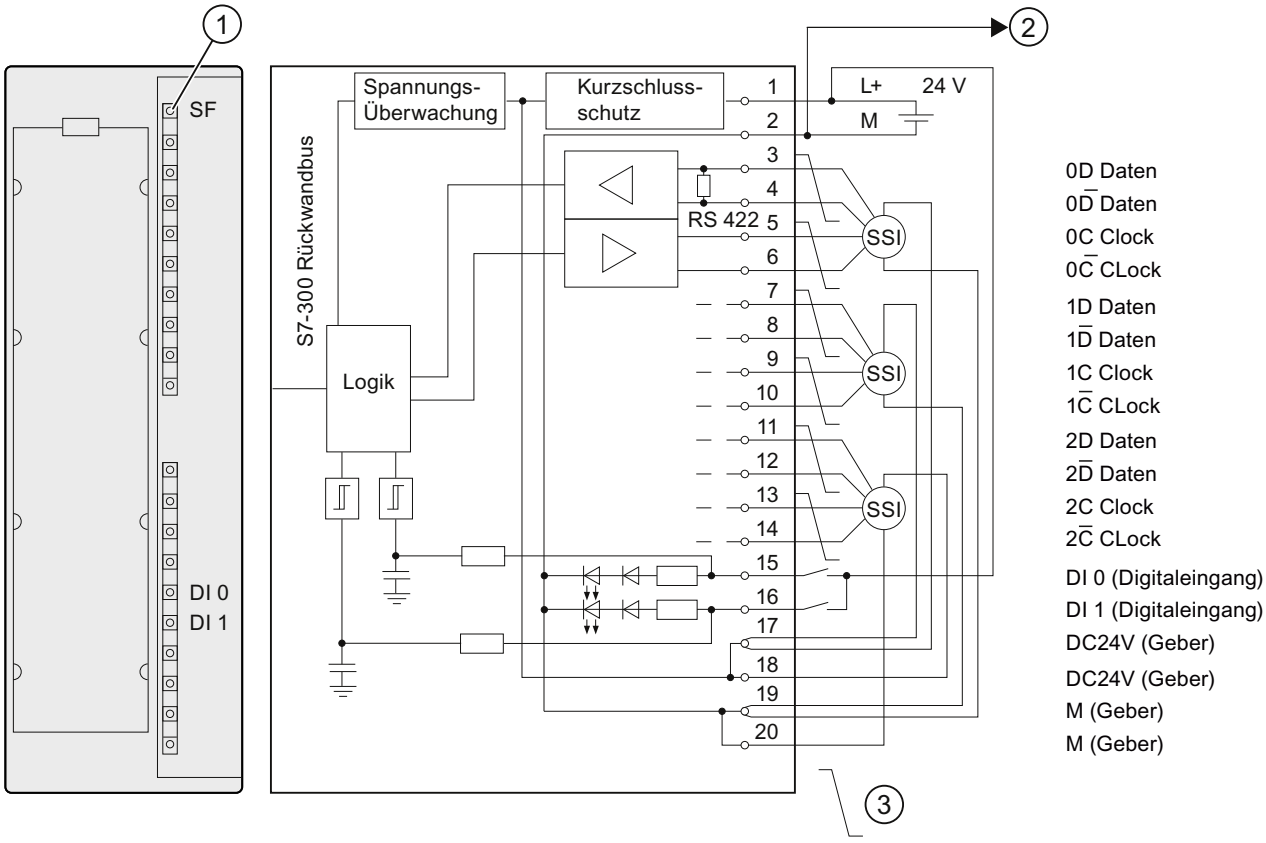

① Fehleranzeige - rot

② Verbindung zur Masse der CPU

③ Leitungen paarweise verdrillen

6.2.2.2 SM 338 konfigurieren

6.2.2.3 SM 338 parametrieren

Betriebsart

Fast Mode aktivieren

Fast Mode

Fast Mode aktivieren Sie im entsprechenden Optionskästchen.

Im Fast Mode arbeitet die Baugruppe mit schnellerer Geberwerterfassung und komprimierter Rückmeldeschnittstelle. Die Freeze-Funktion ist nicht möglich.

Taktsynchroner Betrieb

Hardware-Voraussetzungen

Für den taktsynchronen Betrieb der SM 338 benötigen Sie:

- CPU, die Taktsynchronität unterstützt
- DP-Master, der den äquidistanten Buszyklus unterstützt
- Interfacemodul IM 153-x, das den taktsynchronen Betrieb unterstützt

Eigenschaften der SM 338 im taktsynchronen Betrieb

Abhängig von der Systemparametrierung arbeitet die SM 338 entweder im nicht taktsynchronen oder im taktsynchronen Betrieb.

Im taktsynchronen Betrieb ist der Datenaustausch zwischen DP-Master und SM 338 taktsynchron zum PROFIBUS DP-Zyklus.

Im taktsynchronen Betrieb sind alle 16 Byte der Rückmeldeschnittstelle konsistent.

Bei einem Verlust der Taktsynchronität durch Störungen oder durch Ausfall oder Verzug von Global Control (GC) geht die SM 338 im nächsten Zyklus ohne Fehlerreaktion wieder in den taktsynchronen Betrieb.

Bei einem Verlust der Taktsynchronität wird die Rückmeldeschnittstelle nicht aktualisiert.

Hinweis

Die Grundlagen des taktsynchronen Betriebs sind im Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045) beschrieben.

Freigabe

Diagnosealarm aktivieren

Diagnosealarm

Diagnosealarme sind nicht voreingestellt, d. h. sie sind ohne entsprechende Aktivierung gesperrt.

Diagnosealarme aktivieren Sie im entsprechenden Optionskästchen.

Wenn Sie Diagnosealarme freigegeben haben, dann werden Ihnen kommende (erstes Auftreten des Fehlers) und gehende Fehlerereignisse (Meldung nach Fehlerbeseitigung aller Fehler) über Alarm (Seite 2441) gemeldet.

Die CPU unterbricht die Bearbeitung des Anwenderprogramms und bearbeitet den Diagnosealarm-OB.

Die Diagnoseinformationen sind bis zum Verlassen des Diagnosealarm-OB konsistent. Mit dem Verlassen des Diagnosealarm-OB wird der Diagnosealarm auf der Baugruppe quittiert.

Gebereingang n

Geberart auswählen

Geberart

Die SM 338 unterstützt folgende Geberarten:

- Absolutwertgeber (SSI) mit 13 Bit Telegrammlänge
- Absolutwertgeber (SSI) mit 21 Bit Telegrammlänge
- Absolutwertgeber (SSI) mit 25 Bit Telegrammlänge

Die Geberart finden Sie in den technische Daten Ihres eingesetzten Absolutwertgebers.

Mit der Auswahl "Kein Geber" ist der Gebereingang ausgeschaltet.

Codeart auswählen

Codeart

Mit dem Parameter "Codeart" bestimmen Sie die Behandlung des vom Geber gelieferten Codes:

- In der Einstellung "Gray" wird der vom Absolutwertgeber in Graycode gelieferte Geberwert in Dualcode umgewandelt.
- In der Einstellung "Dual" bleibt der gelieferte Geberwert unverändert.

Baudrate auswählen

Baudrate

Mit der "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur SM 338.

Wählen Sie die gewünschte Baudrate in der Klappliste aus.

Beachten Sie den Zusammenhang zwischen Leitungslänge und Baudrate:

- bis max. 320 m → 125 kHz
- bis max. 160 m → 250 kHz
- bis max. 60 m → 500 kHz
- bis max. 20 m → 1 MHz

Monoflopzeit auswählen

Monoflopzeit

Die Monoflopzeit ist die Pausenzeit zwischen 2 SSI-Telegrammen.

Die parametrierte Monoflopzeit muss größer sein als die Monoflopzeit des eingesetzten Absolutwertgebers (siehe Technische Daten des Herstellers).

Zu dem parametrierten Wert addiert sich noch die Zeit 2 × (1 / Baudrate). Bei einer Baudrate von 125 kHz ist bei parametrierter Monoflopzeit von 16 µs tatsächlich eine Monoflopzeit von 32 µs wirksam.

Für die Monoflopzeit des Absolutwertgebers gilt folgende Einschränkung:

(1 / Baudrate) < Monoflopzeit des Absolutwertgebers < 64 µs + 2 × (1 / Baudrate)

Wählen Sie die entsprechende Monoflopzeit in der Klappliste aus.

Die Angabe der Monoflopzeit ist bei freilaufender Geberwerterfassung relevant.

Anzahl nachgestellter Bits eintragen

Anzahl nachgestellter Bits [0..12]

Für die Ermittlung der Geberposition und zur korrekten Erkennung der Bewegungsrichtung des Gebers benötigt die SM 338 die Anzahl nachgestellter Bits.

Sie finden diesen Wert in den technischen Daten des eingesetzten Absolutwertgebers.

Schritte/Umdrehung auswählen

Schritte/Umdrehung

Maximal stehen 13 Bit für die Schritte/Umdrehung zur Verfügung. Entsprechend der Angabe "Anzahl nachgestellter Bits (Seite 2437)" wird automatisch die resultierende Anzahl Schritte/Umdrehung errechnet.

Digitaleingang für FREEZE-Funktion auswählen

FREEZE-Funktion

Mit der FREEZE-Funktion "frieren" Sie die aktuellen Geberwerte der SM 338 ein. Die FREEZE-Funktion ist an die Digitaleingänge DI 0 und DI 1 der SM 338 gekoppelt.

Auslöser für das Einfrieren ist ein Flankenwechsel (steigende Flanke) an Digitaleingang DI 0 oder DI 1. Ein eingefrorener Geberwert ist durch das gesetzte Bit 31 (Ausgangsadresse) gekennzeichnet. Mit einem Digitaleingang können Sie einen, zwei oder drei Geberwerte einfrieren.

Die Geberwerte bleiben bis zum Beenden der FREEZE-Funktion erhalten und sind somit ereignisabhängig auswertbar.

FREEZE-Funktion einschalten

Die FREEZE-Funktion müssen Sie bei der Parametrierung einschalten.

Wählen Sie dazu den Digitaleingang aus, an dem eine steigende Flanke den Geberwert einfriert:

- aus (Geberwert wird nicht eingefroren)
- DI 0
- DI 1

FREEZE-Funktion beenden

Die FREEZE-Funktion müssen Sie je Gebereingang beenden. Sie quittieren die Funktion im Anwenderprogramm, indem Sie entsprechend dem Gebereingang das Bit 0, 1 bzw. 2 mit der Operation T PAB "xyz" setzen (siehe Programmbeispiel (Seite 2439)).

Nach der Quittierung ist das Bit 31 des entsprechen Geberwertes wieder gelöscht und die Geberwerte werden wieder aktualisiert. Ein erneutes Einfrieren der Geberwerte ist wieder möglich, sobald Sie das Quittungsbit in der Ausgangsadresse der Baugruppe gelöscht haben.

Im taktsynchronen Betrieb wird die Quittierung zum Zeitpunkt T_o verarbeitet. Ab diesem Zeitpunkt kann ein erneutes Einfrieren der Geberwerte über die Digitaleingänge erfolgen.

Hinweis

Die FREEZE-Funktion wird automatisch quittiert, wenn Sie den entsprechenden Gebereingang mit unterschiedlichen Parametern neu parametrieren.
Bei identischen Parametern bleibt die FREEZE-Funktion unbeeinflusst.

SM 338 adressieren

Datenbereiche für die Geberwerte

Die Ein- und Ausgänge der SM 338 werden ab der Baugruppenanfangsadresse adressiert. Die Ein- und Ausgangsadresse ermitteln Sie bei der Konfigurierung der SM 338.

Eingangsadressen

Die folgende Tabelle zeigt die Eingangsadressen der SM 338.

Gebereingang	Eingangsadresse (aus Konfigurierung) + Adress-Offset
0	"Baugruppenanfangsadresse"
1	"Baugruppenanfangsadresse" + 4 Byte Adress-Offset
2	"Baugruppenanfangsadresse" + 8 Byte Adress-Offset

Aufbau des Datendoppelwortes im Standard Mode

Das folgende Bild zeigt je Gebereingang den Aufbau des Datendoppelwortes im Standard Mode.

0 = Geberwert ist nicht eingefroren. Der Wert wird ständig aktualisiert.
1 = Geberwert ist eingefroren. Der Wert bleibt konstant bis zur Quittierung.

Aufbau des Datendoppelwortes im Fast Mode

Das folgende Bild zeigt je Gebereingang den Aufbau des Datendoppelwortes im Fast Mode.

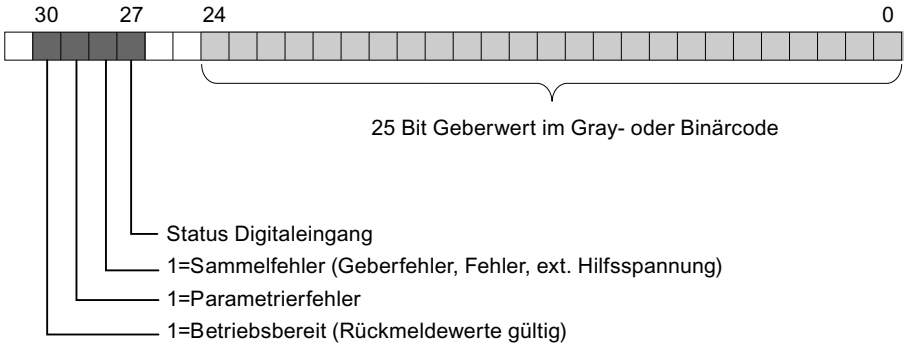

Im Datendoppelwert vom Gebereingang 0 wird im Bit 27 (Status Digitaleingang) der Status vom Digitaleingang DI 0 und im Datendoppelwort vom Gebereingang 1 der Status des Digitaleingag DI 1 gemeldet.

Im Datendoppelwort vom Gebereingang 2 ist das Bit 27 immer = 0.

Ausgangsdaten im Standard Mode

Das folgende Bild zeigt den Aufbau der Ausgangsdaten im Standard Mode.

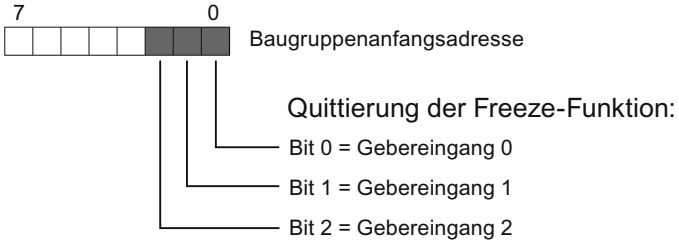

Im Fast Mode werden keine Ausgangsdaten unterstützt.

Datenbereiche auslesen

Sie können die Datenbereiche in Ihrem Anwenderprogramm mit der Operation L PED "xyz" auslesen.

Programmbeispiel für den Zugriff auf Geberwerte und Nutzung der FREEZE-Funktion

Sie wollen den Wert des Gebers an den Gebereingängen auslesen und auswerten. Die Baugruppenanfangsadresse ist 256.

AWL	Erklärung
L PED256	// Geberwert in dem Adressbereich für Gebereingang 0 lesen
T MD 100	// Geberwert in Merkerdoppelwort ablegen

```
AWL                 Erklärung
UD W#16#1FFFFFF     // die unteren 25 Bit verunden, um die oberen 7 Bit komplett
                    auszublenden
T MD 120            // Geberwert ohne FREEZE oder andere Status-Bits auch im Fast
                    Mode nutzbar, da nur 25 Bit zur Anzeige kommen
U M 100.7           // FREEZE-Zustand für spätere Quittierung
= M 99.0            // ermitteln und ablegen

L PED 260           // Geberwert in dem Adressbereich für Gebereingang 1 lesen

T MD 104            // Geberwert in Merkerdoppelwort ablegen
UD W#16#1FFFFFF     // die unteren 25 Bit verunden, um die oberen 7 Bit komplett
                    auszublenden
T MD 124            // Geberwert ohne FREEZE oder andere Status-Bits auch im Fast
                    Mode nutzbar, da nur 25 Bit zur Anzeige kommen
U M 104.7           // FREEZE-Zustand für spätere Quittierung
= M 99.1            // ermitteln und ablegen

L PED 264           // Geberwert in dem Adressbereich für Gebereingang 2 lesen

T MD 108            // Geberwert in Merkerdoppelwort ablegen
UD W#16#1FFFFFF     // die unteren 25 Bit verunden, um die oberen 7 Bit komplett
                    auszublenden
T MD 128            // Geberwert ohne FREEZE oder andere Status-Bits auch im Fast
                    Mode nutzbar, da nur 25 Bit zur Anzeige kommen
U M 108.7           // FREEZE-Zustand für spätere Quittierung
= M 99.2            // ermitteln und ablegen

L MB 99             // FREEZE-Zustand laden und
T PAB 256           // quittieren (SM 338:Ausgangsadresse 256)
```

Danach können Sie die Geberwerte aus dem Merkerbereich MD 100, MD 104 und MD 108 weiterverarbeiten. Der Geberwert steht in den Bits 0 bis 30 des Merkerdoppelwortes.

6.2.2.4 SM 338: Online & Diagnose

Diagnose der SM 338

Die SM 338 stellt Diagnosemeldungen zur Verfügung. D. h., alle Diagnosemeldungen werden ohne Ihr Zutun immer von der SM 338 bereitgestellt.

Aktionen nach Diagnosemeldung

Jede Diagnosemeldung führt zu folgenden Aktionen:

- Die Diagnosemeldung wird in die Diagnose der Baugruppe eingetragen und an die CPU weitergeleitet.
- Die SF-LED auf der Baugruppe leuchtet.
- Wenn Sie "Freigabe Diagnosealarm (Seite 2436)" parametriert haben, dann wird ein Diagnosealarm ausgelöst und der Diagnosealarm-OB aufgerufen.

Diagnosemeldungen auslesen

Sie können die detaillierten Diagnosemeldungen mittels Anweisungen im Anwenderprogramm auslesen.

Die Fehlerursache können Sie in der Baugruppendiagnose anzeigen lassen.

Diagnosemeldung über SF-LED

Die SM 338 zeigt Ihnen Fehler über ihre SF-LED (Sammelfehler-LED) an. Die SF-LED leuchtet, sobald eine Diagnosemeldung von der SM 338 ausgelöst wird. Sie erlischt, wenn alle Fehler behoben sind.

Die SF-LED leuchtet auch bei externen Fehlern (Kurzschluss der Geberversorgung), unabhängig vom Betriebszustand der CPU (bei NETZ-EIN).

Die SF-LED leuchtet im Anlauf kurzzeitig während des Selbsttests der SM 338.

Diagnosemeldungen der SM 338

Die folgende Tabelle zeigt Ihnen die Diagnosemeldungen der SM 338.

Diagnosemeldung	LED	Wirkungsbereich der Diagnose
Baugruppenstörung	SF	Baugruppe
Fehler intern	SF	Baugruppe
Fehler extern	SF	Baugruppe
Kanalfehler vorhanden	SF	Baugruppe
externe Hilfsspannung fehlt	SF	Baugruppe
Baugruppe nicht parametriert	SF	Baugruppe
falsche Parameter	SF	Baugruppe
Kanalinformation vorhanden	SF	Baugruppe
Zeitüberwachung angesprochen	SF	Baugruppe
Kanalfehler vorhanden	SF	Kanal (Gebereingang)
Projektierungs- / Parametrierfehler	SF	Kanal (Gebereingang)
externer Kanalfehler (Geberfehler)	SF	Kanal (Gebereingang)

Fehlerursachen und Abhilfemaßnahmen

Die folgende Tabelle zeigt Fehlerursachen und Abhilfemaßnahmen bei Diagnosemeldungen der SM 338.

Diagnosemeldung	mögliche Fehlerursache	Abhilfemaßnahme
Baugruppenstörung	Beliebiger, durch die Baugruppe erkannter Fehler ist aufgetreten.	
Fehler intern	Baugruppe hat einen Fehler innerhalb des Automatisierungssystems erkannt.	
Fehler extern	Baugruppe hat einen Fehler außerhalb des Automatisierungssystems erkannt.	
Kanalfehler vorhanden	Zeigt an, dass nur bestimmte Kanäle fehlerbehaftet sind.	
externe Hilfsspannung fehlt	Die Versorgungsspannung L+ der Baugruppe fehlt.	Versorgung L+ zuführen
Baugruppe nicht parametriert	Baugruppe benötigt die Information, ob sie mit systemseitig voreingestellten Parametern oder mit Ihren Parametern arbeiten soll.	Meldung steht nach NETZ-EIN bis zur abgeschlossenen Übertragung der Parameter von der CPU an; Baugruppe ggf. parametrieren.
falsche Parameter	Ein Parameter oder die Kombination von Parametern ist unplausibel.	Baugruppe neu parametrieren
Kanalinformation vorhanden	Kanalfehler vorhanden; Baugruppe kann zusätzliche Kanalinformation liefern.	
Zeitüberwachung angesprochen (watch dog)	Zeitweise hohe elektromagnetische Störungen.	Beseitigung der Störungen
Kanalfehler vorhanden	Beliebiger, durch die Baugruppe erkannter Fehler ist an einem Gebereingang aufgetreten.	
Projektierungs- / Parametrierfehler	Fehlerhafter Parameter wurde an Baugruppe übertragen.	Baugruppe neu parametrieren
externer Kanalfehler (Geberfehler)	Drahtbruch des Geberkabels, Geberkabel nicht angeschlossen oder Geber defekt.	angeschlossenen Geber überprüfen

6.2.3 ET 200S 1SSI einsetzen

6.2.3.1 Anschlussbelegung des 1SSI

Verdrahtungsregeln

Die Leitungen (Klemmen 1 und 5 sowie Klemmen 4 und 8) müssen geschirmt und paarweise verdrillt sein. Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung des 1SSI

Die folgende Tabelle zeigt die Anschlussbelegung für das 1SSI.

Ansicht	Anschlussbelegung	Bemerkungen	
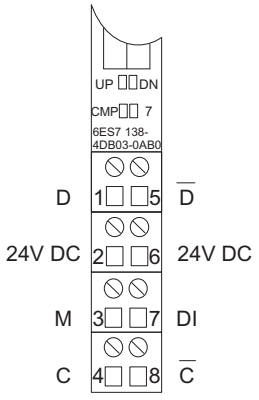	TM-E15S24-01 und EM 1SSI 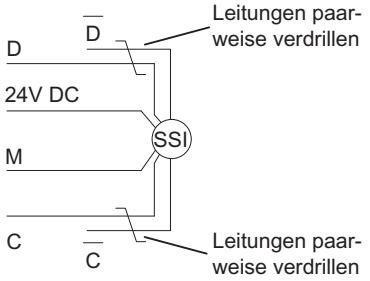	Klemmen 1 bis 8:	
		1/5:	Daten vom SSI-Geber [1]
		2/6:	Spannungsversorgung für Absolutwertgeber und Schalter [2]
		3:	Masse
		7:	Digitaleingang Latchfunktion
		4/8:	SSI-Clock (Taktleitung) [1]
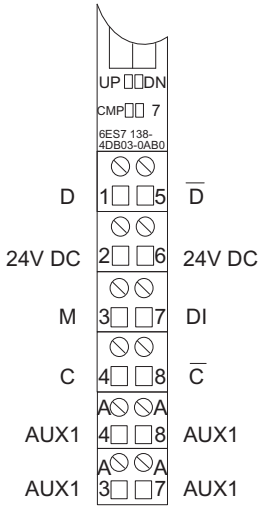	TM-E15S26-A1 und EM 1SSI 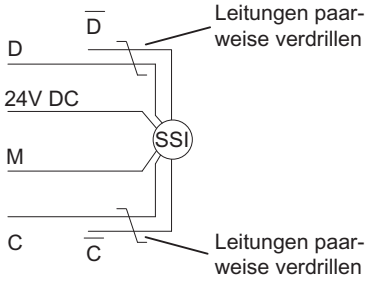		

[1] Beim Verdrahten unbedingt auf die richtige Polarität achten, da sonst Fehler Absolutwertgeber gemeldet wird. Signale nach RS 422

[2] Kurzschlussfest, max. 0,5 A.

6.2.3.2 1SSI konfigurieren und parametrieren

Betrieb

Betriebsmodus auswählen

Betriebsmodus

Mit der Vorgabe eines Betriebsmodus wählen Sie, mit welcher Funktionalität Sie das 1SSI betreiben wollen.

Betriebsmodus	Beschreibung
Standard Mode (Seite 2445)	Standard-Betriebsmodus des 1SSI mit vollständigem Funktionsumfang: • Normierung des Geberwertes • Drehrichtungsumkehr zur Anpassung der Bewegungsrichtung des Absolutwertgebers an die Achse • Art der Geberwerterfassung wählbar: – freilaufend – synchron zur Aktualisierungsrate • Berücksichtigung der maximalen Geberabtastrate (z. B. bei Ultraschallgebern) im taktsynchronen Betrieb • Lebenszeichen im taktsynchronen Betrieb • Paritätsprüfung des Geberwertes möglich • Gray-/Dual-Wandler • Latchfunktion zum Einfrieren das aktuellen Geberwertes • Vergleichsfunktion zwischen dem aktuellem Geberwert und ladbaren Vergleichswerten Dieser Betriebsmodus eignet sich zum Überwachen bzw. Erkennen von Wegpunkten, zur Längenmessung, Kantenerfassung und Synchronisation mit Werkstücken.
Fast Mode (Seite 2457)	In diesem Betriebsmodus hat das 1SSI komprimierte Funktionalität. Nicht verfügbar sind die Latch- und die Vergleichsfunktion. Dieser Betriebsmodus eignet sich für schnelle Geberwerterfassung sowie Regelungsapplikationen, wie z. B. Lageregelungen mit Weg als Istwert.

Parameter für Standard Mode

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Geberwerterfassung auswählen

Beschreibung

Für die Geberwerterfassung haben Sie folgende Alternativen:

- freilaufende Geberwerterfassung
- synchrone Geberwerterfassung
- taktsynchrone Geberwerterfassung

Freilaufende oder synchrone Geberwerterfassung wählen Sie bei der Parametrierung mit dem Parameter "Erfassung" aus. Dieser Parameter wirkt nur im nicht taktsynchronen Betrieb.

Taktsynchrone Geberwerterfassung erhalten Sie, wenn das 1SSI im taktsynchronen Betrieb arbeitet. Der Parameter "Erfassung" wird dann nicht ausgewertet.

Die folgende Tabelle zeigt diese Zusammenhänge der Geberwerterfassung.

Betrieb	Parameter "Erfassung"	Geberwerterfassung
Nicht taktsynchroner Betrieb	freilaufend	freilaufende Geberwerterfassung
	synchron	synchrone Geberwerterfassung
Taktsynchroner Betrieb	- (irrelevant)	taktsynchrone Geberwerterfassung

Freilaufende Geberwerterfassung

Bei freilaufender Geberwerterfassung erhalten Sie höchste Genauigkeit beim Latchen des Geberwertes (Seite 2451).

Das 1SSI veranlasst die Übertragung eines Telegramms jeweils nach Ablauf der parametrierten Monoflopzeit (Seite 2448).

Asynchron zu diesen freilaufenden Telegrammen verarbeitet das 1SSI den erfassten Geberwert im Zyklus seiner Aktualisierungsrate.

Dadurch ergeben sich bei der freilaufenden Geberwerterfassung unterschiedlich alte Geberwerte. Die Differenz zwischen maximalem und minimalem Alter ist der Jitter.

Synchrone Geberwerterfassung

Bei synchroner Geberwerterfassung erhalten Sie höchste Genauigkeit bei der Geberwerterfassung.

Das 1SSI veranlasst die Übertragung eines Telegramms im Zyklus seiner Aktualisierungsrate.

Synchron zu seiner Aktualisierungsrate verarbeitet das 1SSI den übertragenen Geberwert.

Geberart auswählen

Geberart

Das 1SSI unterstützt folgende Geberarten:

- Absolutwertgeber (SSI) mit 13 Bit
  bis
- Absolutwertgeber (SSI) mit 25 Bit

Die Geberart finden Sie in den technischen Daten Ihres eingesetzten Absolutwertgebers.

Mit der Auswahl "Kein Geber" ist der Gebereingang ausgeschaltet.

Gray-/Dualwandler auswählen

Gray-/Dual-Wandler

Mit dem Parameter "Gray-/Dual-Wandler" bestimmen Sie die Behandlung des vom Geber gelieferten Codes:

- In der Einstellung "Gray" wird der vom Absolutwertgeber in Graycode gelieferte Geberwert in Dualcode umgewandelt.
- In der Einstellung "Dual" bleibt der gelieferte Geberwert unverändert.

Baudrate auswählen

Baudrate

Mit der "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur 1SSI.

Wählen Sie die gewünschte Baudrate in der Klappliste aus.

Beachten Sie, dass die Baudrate die Genauigkeit und Aktualität der Geberwerte beeinflusst.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz
- 10 m → 1,5 MHz
- 8 m → 2 MHz

Siehe auch

Monoflopzeit auswählen (Seite 2448)

Parität auswählen

Parität

Mit der "Parität" bestimmen Sie, ob für den Absolutwertgeber keine, ungerade oder gerade Parität übertragen werden soll.

Ein parametriertes Paritäts-Bit zählt beim Parameter "Geberart (Seite 2447)" nicht mit. Ist ein 25 Bit-Geber mit Parität parametriert, so werden vom Geber 26 Bit eingelesen.

Das dem LSB (Least significant Bit) folgende Bit des Gebers wird als Paritäts-Bit ausgewertet. Ein Paritäts-Fehler wird in der Rückmeldeschnittstelle (Seite 2456) über das Bit ERR_SSI gemeldet.

Monoflopzeit auswählen

Monoflopzeit

Die Monoflopzeit ist die Pausenzeit zwischen 2 SSI-Telegrammen.

Die parametrierte Monoflopzeit muss größer sein als die Monoflopzeit des eingesetzten Absolutwertgebers (siehe Technische Daten des Herstellers).

Für die Monoflopzeit des Absolutwertgebers gilt folgende Einschränkung:

(1 / Baudrate) < Monoflopzeit des Absolutwertgebers < 64 µs

Wählen Sie die entsprechende Monoflopzeit in der Klappliste aus.

Die Angabe der Monoflopzeit ist bei freilaufender Geberwerterfassung (Seite 2446) relevant.

Siehe auch

Baudrate auswählen (Seite 2447)

Normierung aktivieren

Geberwert und Normierung

Der übertragene Geberwert enthält die Geberposition des Absolutwertgebers. Je nach verwendetem Geber werden neben der Geberposition weitere Bits übertragen, die sich vor und nach der Geberposition befinden.

Damit das 1SSI die Geberposition ermitteln kann, braucht es folgende Angaben:

- Geberart (Seite 2447)
- Anzahl nachgestellter Bits (Seite 2449)
- Inkremente/Geberumdrehung (Seite 2450)

Mit der Normierung legen Sie die Darstellung des Geberwertes in der Rückmeldeschnittstelle (Seite 2456) fest.

- Mit "Normierung **ein**" legen Sie fest, dass nachgestellte, nicht relevante Bits im Geberwert weggeschoben werden.
- Mit "Normierung **aus**" legen Sie fest, dass nachgestellte Bits erhalten bleiben und zur Auswertung bereitstehen.

Anzahl nachgestellter Bits eintragen

Anzahl nachgestellter Bits

Für die Ermittlung der Geberposition und zur korrekten Erkennung der Bewegungsrichtung des Gebers benötigt das 1SSI die Anzahl nachgestellter Bits.

Sie finden diesen Wert in den technischen Daten des eingesetzten Absolutwertgebers.

Drehrichtungsumkehr aktivieren

Richtungserkennung und Drehrichtungsumkehr

Zur korrekten Erkennung der Bewegungsrichtung des Gebers benötigt das 1SSI folgende Informationen:

- Geberart (Seite 2447)
- Anzahl nachgestellter Bits (Seite 2449)
- Inkremente/Geberumdrehung (Seite 2450)

Die ermittelte Bewegungsrichtung wird in der Rückmeldeschnittstelle (Seite 2456) und an den LED angezeigt.

- LED UP: Änderung der Geberposition vom kleineren zum größeren Wert
- LED DN: Änderung der Geberposition vom größeren zum kleineren Wert

Mit der **Drehrichtungsumkehr** wird die Bewegungsrichtung des Gebers an die der Achse angepasst.

Dabei sind zwei Einstellungen möglich:

- **Aus:** Die Richtung der übertragenen Geberposition bleibt erhalten.
- **Ein:** Die Richtung der übertragenen Geberposition wird umgekehrt. D. h., obwohl der Geber aufsteigende Werte liefert, werden absteigende Werte angezeigt.
  Die Umkehr bezieht sich auf die in den Parametern eingestellten Inkremente/Geberumdrehung.

Beispiel zur Drehrichtungsumkehr

Vorgaben:

Sie verwenden einen Singleturn-Geber mit 2^{10} (entspricht 10 Bit) = 1024 Inkremente/ Geberumdrehung (Auflösung / 360°) mit folgender Parametrierung:

- Geberart: SSI-13 Bit
- Anzahl nachgestellter Bits: 3 Stellen
- Drehrichtungsumkehr: ein
- Inkremente/Geberumdrehung: 1024

Geberwert vor Drehrichtungsumkehr: Zyklisch erfasste Geberposition 1023

Geberwert nach Drehrichtungsumkehr: Angezeigte Geberposition 0

Inkremente pro Geberumdrehung eintragen

Inkremente pro Geberumdrehung

Der Parameter "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt.

Sie finden diesen Wert in den technischen Daten des eingesetzten Absolutwertgebers.

Zulässige Wertebereiche

Für die Grenzen muss zwischen den einzelnen Gebermodellen unterschieden werden. Als Eingabe sind nur Werte in 2er Potenzen erlaubt.

- Geberart 13 Bit: 16 bis 8192
- Geberart 14 Bit: 16 bis 16384
- Geberart 15 Bit: 16 bis 32768
- Geberart 16 Bit: 16 bis 65536
- Geberart 17 Bit: 16 bis 131072
- Geberart 18 Bit: 16 bis 262144
- Geberart 19 Bit: 16 bis 524288
- Geberart 20 Bit: 16 bis 1048576
- Geberart 21 Bit: 16 bis 2097152
- Geberart 22 Bit: 16 bis 4194304
- Geberart 23 Bit: 16 bis 8388608
- Geberart 24 Bit: 16 bis 16777216
- Geberart 25 Bit: 16 bis 33554432

Latchen des Geberwertes auswählen

Latchen des Geberwertes

Mit der Latch-Funktion "frieren" Sie bei einer Flanke am Digitaleingang (DI) den aktuellen Geberwert des 1SSI ein.

Der Geberwert ist damit ereignisabhängig auswertbar.

Ein eingefrorener Geberwert ist durch das gesetzte Bit 31 gekennzeichnet und bleibt bis zum Beenden der Latch-Funktion erhalten.

Der eingefrorene Geberwert wird in der Rückmeldeschnittstelle (Seite 2456) an der Stelle des ansonsten zyklisch erfassten Wertes eingetragen und mit Kennung "Bit 31 gesetzt" versehen.

Hinweis

Auch bei eingefrorenem Geberwert erfolgt eine Richtungsermittlung und Vergleichsfunktion sowie eine Fehlerüberwachung.

Voraussetzung zur Nutzung der Latch-Funktion

Über die Parametrierung wählen Sie aus, welche Flanke am Digitaleingang den Geberwert einfriert:

- fallende Flanke am DI
- steigende Flanke am DI
- beide Flanken am DI

Mit der Flankenauswahl wird die Latch-Funktion eingeschaltet, die an den Digitaleingang gekoppelt ist.

Mit der Auswahl "inaktiv" ist das Einfrieren des Geberwertes nicht möglich.

Beenden der Latch-Funktion

Die Latch-Funktion muss quittiert werden. Wenn das Steuerungsprogramm die Übernahme des Geberwertes quittiert, wird Bit 31 gelöscht und der Geberwert wieder aktualisiert. Danach ist erneutes Einfrieren wieder möglich.

Das folgende Bild zeigt den Ablauf der Latch-Funktion.

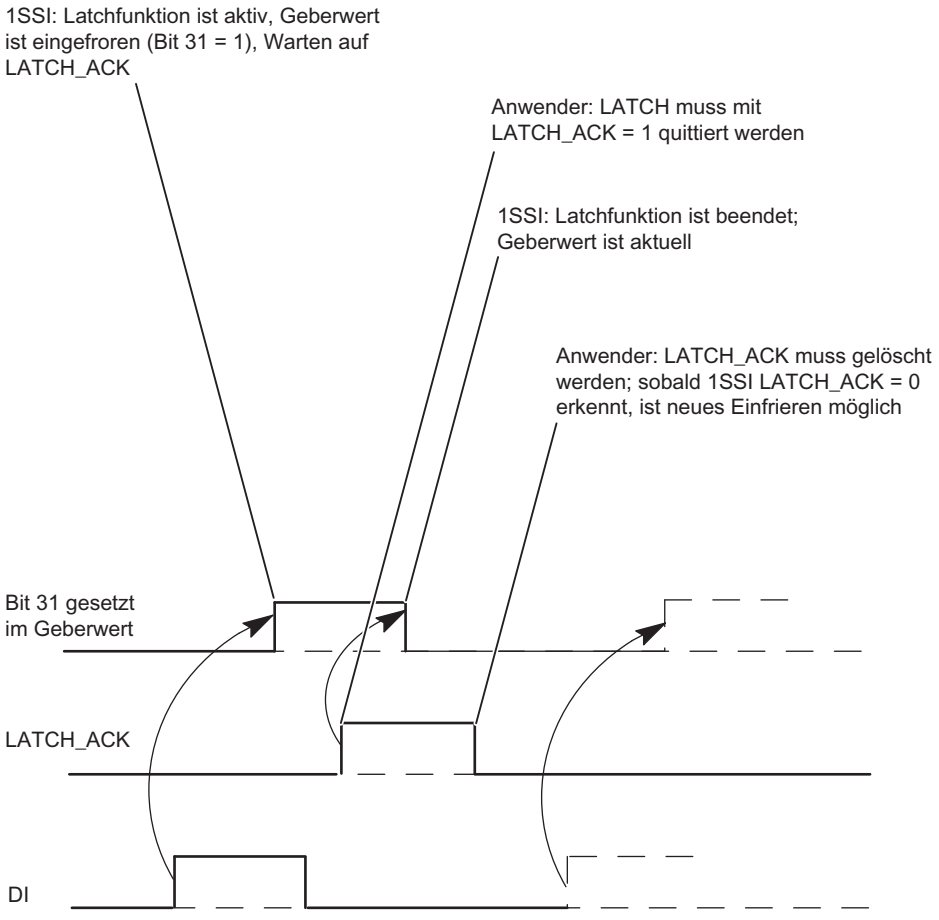

Verhalten der Vergleicher auswählen

Einstellen der Vergleicher

Die erfasste Geberposition kann mit bis zu zwei ladbaren Werten verglichen werden. Die beiden Vergleichsergebnisse sind in der Rückmeldeschnittstelle (Seite 2456) abgelegt. Erst nach Laden des Vergleichswertes ist der zugehörige Vergleicher aktiv.

Die beiden Vergleicher werden in den Parametern "Vergleicher 1" und "Vergleicher 2" eingestellt:

Einstellung	Wirkung auf das Vergleichsergebnis (CMPx)
inaktiv	Der Geberwert wird nicht verglichen. Das Rückmeldebit CMPx = 0.
vorwärts	Der Geberwert wird in Richtung vorwärts (UP) verglichen. • Ist der Geberwert ≥ Vergleichswert, wird das Rückmeldebit CMPx = 1. • Ist der Geberwert < Vergleichswert, wird das Rückmeldebit CMPx = 0. • Ist die Richtung rückwärts, bleibt das Rückmeldebit CMPx unverändert. • Wird keine Geberwertänderung erkannt, bleibt das Rückmeldebit CMPx unverändert.

Einstellung	Wirkung auf das Vergleichsergebnis (CMPx)
rückwärts	Der Geberwert wird in Richtung rückwärts (DN) verglichen. • Ist der Geberwert ≤ Vergleichswert, wird das Rückmeldebit CMPx = 1. • Ist der Geberwert > Vergleichswert, wird das Rückmeldebit CMPx = 0. • Ist die Richtung vorwärts, bleibt das Rückmeldebit CMPx unverändert. • Wird keine Geberwertänderung erkannt, bleibt das Rückmeldebit CMPx unverändert.
in beide Richtungen	Der Geberwert wird in beide Richtungen verglichen. Ist die Richtung vorwärts, gilt: • Ist der Geberwert ≥ Vergleichswert, wird das Rückmeldebit CMPx = 1. • Ist der Geberwert < Vergleichswert, wird das Rückmeldebit CMPx = 0. • Wird keine Geberwertänderung erkannt, bleibt das Rückmeldebit CMPx unverändert. Ist die Richtung rückwärts, gilt: • Ist der Geberwert ≤ Vergleichswert, wird das Rückmeldebit CMPx = 1. • Ist der Geberwert > Vergleichswert, wird das Rückmeldebit CMPx = 0. • Wird keine Geberwertänderung erkannt, bleibt das Rückmeldebit CMPx unverändert.

Sobald Sie einen Vergleichswert laden, wird das Vergleichsergebnis gelöscht und danach das neue Vergleichsergebnis entsprechend der richtungsabhängigen Einstellung eingetragen.

Hinweis

Es darf immer nur ein Steuerbit gesetzt sein:

CMP_VAL1 oder CMP_VAL2.

Ansonsten wird der Fehler ERR_LOAD gemeldet, bis beide Steuerbits gelöscht sind.

Vergleichswert laden

Das folgende Bild zeigt den Ablauf einer Wertübergabe.

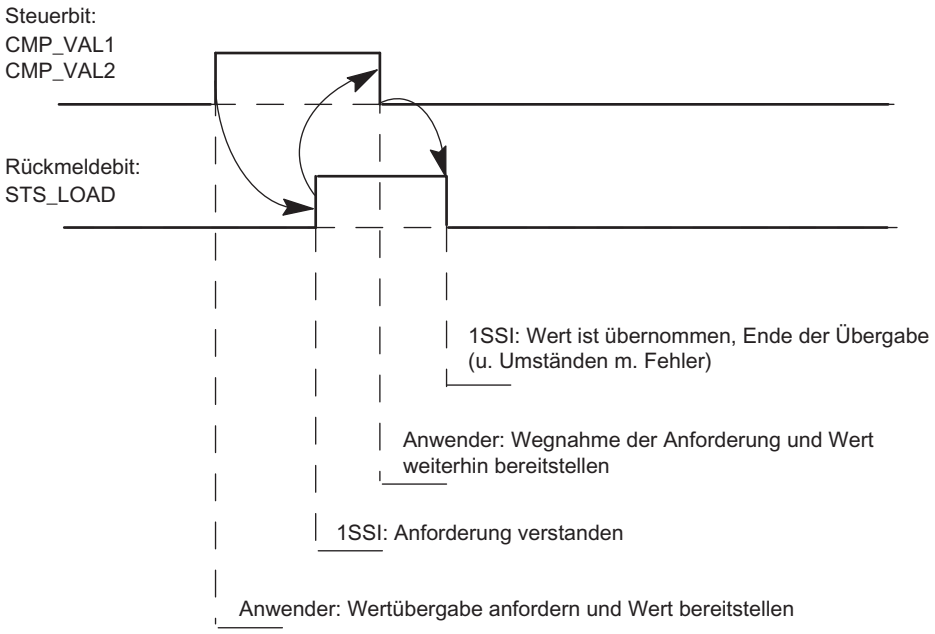

Vergleicher bei taktsynchronem Betrieb

Bei taktsynchronem Betrieb werden die Vergleichswerte zum Zeitpunkt T_o geladen und wirken ab dem Zeitpunkt T_i im gleichen Buszyklus.

Lebenszeichen aktivieren (Standard Mode)

Lebenszeichen

Der Parameter ist nur im taktsynchronen Betrieb wirksam.

Bei aktiviertem Lebenszeichen wird im taktsynchronen Betrieb bei jedem eingelesenen Geberwert das Lebenszeichen-Bit getoggelt, d. h., der zuletzt gesendete Wert wird invertiert. Ist über den Parameter "Geberabtastrate (Seite 2454)" eine Untersetzung parametriert, so wird der Wert nur dann getoggelt, wenn auch tatsächlich ein Geberwert eingelesen wurde.

Das Lebenszeichen-Bit ist in der Rückmeldeschnittstelle (Seite 2456) im Byte 4 / Bit 7 zu finden.

Geberabtastrate einstellen

Geberabtastrate

Der Parameter ist nur im taktsynchronen Betrieb wirksam.

Hier stellen Sie eine eventuell zu berücksichtigende Geberabtastrate ein. Damit ist es möglich, langsamere Geber (z. B. Ultraschallgeber) auch bei einem schnellen Bearbeitungszyklus einzusetzen.

Anhand der eingestellten Frequenz wird eine ganzzahlige Untersetzung n berechnet. Der Geber wird dann nur bei jedem n-ten Taktzyklus neu eingelesen.

Beispiel

- Bearbeitungszyklus 500 µs
- Geberabtastrate: 1,2 kHz (à ca. 833 µs)

Das ergibt eine Untersetzung n = 2.

Der Geber wird also nur bei jedem zweiten Bearbeitungszyklus neu eingelesen, d. h. jede Millisekunde.

Steuer- und Rückmeldeschnittstelle für Standard Mode

Steuerschnittstelle für Standard Mode

Hinweis

Für das 1SSI sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1SSI im Standard Mode.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Vergleichswert 1 oder 2 (Doppelwort)		
Byte 4	Bit 7	Fehlerquittung	EXTF_ACK
	Bit 6	Quittung Latch-Funktion	LATCH_ACK
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	reserviert = 0	
	Bit 2	reserviert = 0	
	Bit 1	Vergleichswert 2 laden	CMP_VAL2
	Bit 0	Vergleichswert 1 laden	CMP_VAL1
Byte 5	reserviert = 0		
Byte 6 bis 7	reserviert = 0		

Erläuterungen zu den Steuerbits

Bit	Erläuterung
CMP_VAL1	Vergleichswert 1 laden
CMP_VAL2	Vergleichswert 2 laden
EXTF_ACK	Fehlerquittung für "Fehler Absolutwertgeber" ERR_SSI und "Kurzschluss Geberversorgung" ERR_24V
LATCH_ACK	Quittung für Latch-Funktion

Rückmeldeschnittstelle für Standard Mode

Hinweis

Für das 1SSI sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0...3
- Byte 4...7

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1SSI im Standard Mode.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Geberwert (Doppelwort) (Bit 31 gesetzt, Geberwert eingefroren)		
Byte 4	Bit 7	reserviert = 0 oder Lebenszeichen	LZ
	Bit 6	betriebsbereit	RDY
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	Fehler Absolutwertgeber	ERR_SSI
	Bit 3	Kurzschluss Geberversorgung	ERR_24V
	Bit 2	Status DI	STS_DI
	Bit 1	Status DN	STS_DN
	Bit 0	Status UP	STS_UP
Byte 5	Bit 7	reserviert = 0	
	Bit 6	reserviert = 0	
	Bit 5	reserviert = 0	
	Bit 4	reserviert = 0	
	Bit 3	Vergleichswert 2 erreicht	CMP2
	Bit 2	Vergleichswert 1 erreicht	CMP
	Bit 1	Fehler bei Ladefunktion	ERR_LOAD
	Bit 0	Ladefunktion läuft	STS_LOAD
Byte 6 bis 7	reserviert = 0		

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
CMP	Vergleichsergebnis von Vergleicher 1
CMP2	Vergleichsergebnis von Vergleicher 2
ERR_24V	Die Geberversorgung ist kurzgeschlossen. ERR_24V wird zurückgesetzt, wenn der Kurzschluss beseitigt und mit dem Steuerbit EXTF_ACK quittiert ist.
ERR_LOAD	Fehler beim Laden der Vergleichswerte, da beide Steuerbits CMP_VAL1 und CMP_VAL2 gesetzt sind.
ERR_PARA	Bei der Parametrierung der ET 200S-Station sind Parameter fehlerhaft. Ursache: Inkremente/Geberumdrehung nicht im Wertebereich für die Geberart. Das Parameterbit wird gelöscht, wenn Sie eine korrekte Parametrierung übertragen.
ERR_SSI	Das 1SSI erkennt den "Fehler Absolutwertgeber", wenn die Telegramme an der SSI-Schnittstelle gestört sind. Ursachen: kein Geber angeschlossen; Drahtbruch des Geberkabels; Unterspannung der Geberversorgung; Geberart, Baudrate, Monoflopzeit entsprechen nicht dem angeschlossenen Geber; programmierbare Geber entsprechen nicht den Einstellungen auf dem 1SSI; Geber ist defekt oder es liegen Störungen oder Paritäts-Fehler vor. ERR_SSI wird zurückgesetzt, wenn die Fehlerursache beseitigt und mit dem Steuerbit EXTF_ACK quittiert ist.
LZ	Der Parameter ist nur im taktsynchronen Betrieb wirksam. Bei eingeschaltetem Lebenszeichen wird im taktsynchronen Betrieb bei jedem eingelesenen Geberwert das Lebenszeichen-Bit getoggelt, d. h., der zuletzt gesendete Wert wird invertiert. Ist über den Parameter "Geberabtastrate" eine Untersetzung parametriert, so wird der Wert nur dann getoggelt, wenn auch tatsächlich ein Geberwert eingelesen wurde.
STS_DI	Das Bit zeigt den Status des Digitaleingangs DI an.
STS_DN	Status Richtung rückwärts; bei Geberwertänderung von größeren zu kleineren Geberpositionen (inkl. Nulldurchgang)
STS_LOAD	Rückmeldebit zu CMP_VAL1 bzw. CMP_VAL2. Mit diesem Bit zeigt das 1SSI an, dass ein Vergleichswert geladen wird.
STS_UP	Status Richtung vorwärts; bei Geberwertänderung von kleineren zu größeren Geberpositionen (inkl. Nulldurchgang)
RDY	Das 1SSI ist korrekt parametriert und führt seine Funktionen aus. Die angezeigten Rückmeldungen sind gültig. Bei Fehler Absolutwertgeber ist ERR_SSI auch gesetzt.

Parameter für Fast Mode

Sammeldiagnose aktivieren

Sammeldiagnose

Wenn Sie die Sammeldiagnose in Ihrer Parametrierung aktivieren, dann wird eine kanalbezogene Diagnose durchgeführt.

Geberwerterfassung auswählen

Beschreibung

Der Absolutwertgeber überträgt seine Geberwerte in Telegrammen zum 1SSI. Die Übertragung der Telegramme wird vom 1SSI veranlasst. Für die Geberwerterfassung haben Sie folgende Alternativen:

- freilaufende Geberwerterfassung
- synchrone Geberwerterfassung
- taktsynchrone Geberwerterfassung

Freilaufende oder synchrone Geberwerterfassung wählen Sie bei der Parametrierung mit dem Parameter "Erfassung" aus. Dieser Parameter wirkt nur im nicht taktsynchronen Betrieb.

Taktsynchrone Geberwerterfassung erhalten Sie, wenn das 1SSI im taktsynchronen Betrieb arbeitet. Der Parameter "Erfassung" wird dann nicht ausgewertet.

Die folgende Tabelle zeigt diese Zusammenhänge der Geberwerterfassung.

Betrieb	Parameter "Erfassung"	Geberwerterfassung
Nicht taktsynchroner Betrieb	freilaufend	freilaufende Geberwerterfassung
	synchron	synchrone Geberwerterfassung
Taktsynchroner Betrieb	- (irrelevant)	taktsynchrone Geberwerterfassung

Freilaufende Geberwerterfassung

Das 1SSI veranlasst die Übertragung eines Telegramms jeweils nach Ablauf der parametrierten Monoflopzeit (Seite 2460).

Asynchron zu diesen freilaufenden Telegrammen verarbeitet das 1SSI den erfassten Geberwert im Zyklus seiner Aktualisierungsrate.

Dadurch ergeben sich bei der freilaufenden Geberwerterfassung unterschiedlich alte Geberwerte. Die Differenz zwischen maximalem und minimalem Alter ist der Jitter.

Synchrone Geberwerterfassung

Bei synchroner Geberwerterfassung erhalten Sie höchste Genauigkeit bei der Geberwerterfassung.

Das 1SSI veranlasst die Übertragung eines Telegramms im Zyklus seiner Aktualisierungsrate.

Synchron zu seiner Aktualisierungsrate verarbeitet das 1SSI den übertragenen Geberwert.

Taktsynchrone Geberwerterfassung

Taktsynchrone Geberwerterfassung stellt sich automatisch dann ein, wenn das Bussystem und die Peripheriemodule synchronisiert arbeiten.

Das 1SSI veranlasst die Übertragung eines Telegramms in jedem Buszyklus zum Zeitpunkt T_i, sofern die projektierte maximale Geberabtastrate nicht zu einer Untersetzung führt.

Taktsynchron zum Buszyklus verarbeitet das 1SSI den übertragenen Geberwert.

Geberart auswählen

Geberart

Das 1SSI unterstützt folgende Geberarten:

- Absolutwertgeber (SSI) mit 13 Bit
  bis
- Absolutwertgeber (SSI) mit 25 Bit

Die Geberart finden Sie in den technischen Daten Ihres eingesetzten Absolutwertgebers.

Mit der Auswahl "Kein Geber" ist der Gebereingang ausgeschaltet.

Gray-/Dualwandler auswählen

Gray-/Dual-Wandler

Mit dem Parameter "Gray-/Dual-Wandler" bestimmen Sie die Behandlung des vom Geber gelieferten Codes:

- In der Einstellung "Gray" wird der vom Absolutwertgeber in Graycode gelieferte Geberwert in Dualcode umgewandelt.
- In der Einstellung "Dual" bleibt der gelieferte Geberwert unverändert.

Baudrate auswählen

Baudrate

Mit der "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur 1SSI.

Wählen Sie die gewünschte Baudrate in der Klappliste aus.

Beachten Sie, dass die Baudrate die Genauigkeit und Aktualität der Geberwerte beeinflusst.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz
- 10 m → 1,5 MHz
- 8 m → 2 MHz

Siehe auch

Monoflopzeit auswählen (Seite 2460)

Parität auswählen

Parität

Mit der "Parität" bestimmen Sie, ob für den Absolutwertgeber keine, ungerade oder gerade Parität übertragen werden soll.

Ein parametriertes Paritäts-Bit zählt beim Parameter "Geberart (Seite 2459)" nicht mit. Ist ein 25 Bit-Geber mit Parität parametriert, so werden vom Geber 26 Bit eingelesen.

Das dem LSB (Least significant Bit) folgende Bit des Gebers wird als Paritäts-Bit ausgewertet. Ein Paritäts-Fehler wird in der Rückmeldeschnittstelle (Seite 2464) über das Bit EXTF gemeldet.

Monoflopzeit auswählen

Monoflopzeit

Die Monoflopzeit ist die Pausenzeit zwischen 2 SSI-Telegrammen.

Die parametrierte Monoflopzeit muss größer sein als die Monoflopzeit des eingesetzten Absolutwertgebers (siehe Technische Daten des Herstellers).

Für die Monoflopzeit des Absolutwertgebers gilt folgende Einschränkung:

(1 / Baudrate) < Monoflopzeit des Absolutwertgebers < 64 µs

Wählen Sie die entsprechende Monoflopzeit in der Klappliste aus.

Die Angabe der Monoflopzeit ist bei freilaufender Geberwerterfassung (Seite 2458) relevant.

Siehe auch

Baudrate auswählen (Seite 2459)

Normierung aktivieren

Geberwert und Normierung

Der übertragene Geberwert enthält die Geberposition des Absolutwertgebers. Je nach verwendetem Geber werden neben der Geberposition weitere Bits übertragen, die sich vor und nach der Geberposition befinden.

Damit das 1SSI die Geberposition ermitteln kann, braucht es folgende Angaben:

- Geberart (Seite 2459)
- Anzahl nachgestellter Bits (Seite 2461)
- Inkremente/Geberumdrehung (Seite 2462)

Mit der Normierung legen Sie die Darstellung des Geberwertes in der Rückmeldeschnittstelle (Seite 2464) fest.

- Mit "Normierung **ein**" legen Sie fest, dass nachgestellte, nicht relevante Bits im Geberwert weggeschoben werden.
- Mit "Normierung **aus**" legen Sie fest, dass nachgestellte Bits erhalten bleiben und zur Auswertung bereitstehen.

Anzahl nachgestellter Bits eintragen

Anzahl nachgestellter Bits

Für die Ermittlung der Geberposition und zur korrekten Erkennung der Bewegungsrichtung des Gebers benötigt das 1SSI die Anzahl nachgestellter Bits.

Sie finden diesen Wert in den technischen Daten des eingesetzten Absolutwertgebers.

Drehrichtungsumkehr aktivieren

Richtungserkennung und Drehrichtungsumkehr

Zur korrekten Erkennung der Bewegungsrichtung des Gebers benötigt das 1SSI folgende Informationen:

- Geberart (Seite 2459)
- Anzahl nachgestellter Bits (Seite 2461)
- Inkremente/Geberumdrehung (Seite 2462)

Die ermittelte Bewegungsrichtung wird in der Rückmeldeschnittstelle (Seite 2464) und an den LED angezeigt.

- LED UP: Änderung der Geberposition vom kleineren zum größeren Wert
- LED DN: Änderung der Geberposition vom größeren zum kleineren Wert

Mit der **Drehrichtungsumkehr** wird die Bewegungsrichtung des Gebers an die der Achse angepasst.

Dabei sind zwei Einstellungen möglich:

- **Aus:** Die Richtung der übertragenen Geberposition bleibt erhalten.
- **Ein:** Die Richtung der übertragenen Geberposition wird umgekehrt. D. h., obwohl der Geber aufsteigende Werte liefert, werden absteigende Werte angezeigt.
  Die Umkehr bezieht sich auf die in den Parametern eingestellten Inkremente/Geberumdrehung.

Beispiel zur Drehrichtungsumkehr

Vorgaben:

Sie verwenden einen Singleturn-Geber mit 2^{10} (entspricht 10 Bit) = 1024 Inkremente/ Geberumdrehung (Auflösung / 360°) mit folgender Parametrierung:

- Geberart: SSI-13 Bit
- Anzahl nachgestellter Bits: 3 Stellen
- Drehrichtungsumkehr: ein
- Inkremente/Geberumdrehung: 1024

Geberwert vor Drehrichtungsumkehr: Zyklisch erfasste Geberposition 1023

Geberwert nach Drehrichtungsumkehr: Angezeigte Geberposition 0

Inkremente pro Geberumdrehung eintragen

Inkremente pro Geberumdrehung

Der Parameter "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt.

Sie finden diesen Wert in den technischen Daten des eingesetzten Absolutwertgebers.

Zulässige Wertebereiche

Für die Grenzen muss zwischen den einzelnen Gebermodellen unterschieden werden. Als Eingabe sind nur Werte in 2er Potenzen erlaubt.

- Geberart 13 Bit: 16 bis 8192
- Geberart 14 Bit: 16 bis 16384
- Geberart 15 Bit: 16 bis 32768
- Geberart 16 Bit: 16 bis 65536
- Geberart 17 Bit: 16 bis 131072
- Geberart 18 Bit: 16 bis 262144
- Geberart 19 Bit: 16 bis 524288
- Geberart 20 Bit: 16 bis 1048576
- Geberart 21 Bit: 16 bis 2097152
- Geberart 22 Bit: 16 bis 4194304
- Geberart 23 Bit: 16 bis 8388608
- Geberart 24 Bit: 16 bis 16777216
- Geberart 25 Bit: 16 bis 33554432

Lebenszeichen aktivieren (Fast Mode)

Lebenszeichen

Der Parameter ist nur im taktsynchronen Betrieb wirksam.

Bei aktiviertem Lebenszeichen wird im taktsynchronen Betrieb bei jedem eingelesenen Geberwert das Lebenszeichen-Bit getoggelt, d. h., der zuletzt gesendete Wert wird invertiert. Ist über den Parameter "Geberabtastrate (Seite 2463)" eine Untersetzung parametriert, so wird der Wert nur dann getoggelt, wenn auch tatsächlich ein Geberwert eingelesen wurde.

Das Lebenszeichen-Bit ist in der Rückmeldeschnittstelle (Seite 2464) im Byte 0 bis 3 / Bit 31 zu finden.

Geberabtastrate einstellen

Geberabtastrate

Der Parameter ist nur im taktsynchronen Betrieb wirksam.

Hier stellen Sie eine eventuell zu berücksichtigende Geberabtastrate ein. Damit ist es möglich, langsamere Geber (z. B. Ultraschallgeber) auch bei einem schnellen Bearbeitungszyklus einzusetzen.

Anhand der eingestellten Frequenz wird eine ganzzahlige Untersetzung n berechnet. Der Geber wird dann nur bei jedem n-ten Taktzyklus neu eingelesen.

Beispiel

- Bearbeitungszyklus 500 µs
- Geberabtastrate: 1,2 kHz (à ca. 833 µs)

Das ergibt eine Untersetzung n = 2.

Der Geber wird also nur bei jedem zweiten Bearbeitungszyklus neu eingelesen, d. h. jede Millisekunde.

Rückmeldeschnittstelle für Fast Mode

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1SSI im Fast Mode.

Adresse	Belegung		Bezeichnung
Byte 0 bis 3	Bit 31	reserviert = 0 oder Lebenszeichen	LZ
	Bit 30	betriebsbereit (Rückmeldungen sind gültig)	RDY
	Bit 29	Parametrierfehler	ERR_PARA
	Bit 28	Sammelfehler Absolutwertgeber oder Kurzschluss Geberversorgung	EXTF
	Bit 27	Status DI	STS_DI
	Bit 26	Status DN	STS_DN
	Bit 25	Status UP	STS_UP
	Bit 0 bis 24	Geberwert	

Erläuterungen zu den Rückmeldebits

Bit	Erläuterung
ERR_PARA	Bei der Parametrierung der ET 200S-Station sind Parameter fehlerhaft. Ursache: Inkremente/Geberumdrehung nicht im Wertebereich für die Geberart. Das Parameterbit wird gelöscht, wenn Sie eine korrekte Parametrierung übertragen.
EXTF	Sammelfehler Absolutwertgeber oder Kurzschluss Geberversorgung Ursachen: • Die Geberversorgung ist kurzgeschlossen • Kein Geber angeschlossen; Drahtbruch des Geberkabels; Geberart, Baudrate, Monoflopzeit entsprechen nicht dem angeschlossenen Geber; programmierbare Geber entsprechen nicht den Einstellungen auf dem 1SSI; Geber ist defekt oder es liegen Störungen oder Paritäts-Fehler vor. EXTF wird zurückgesetzt, wenn die Fehlerursachen beseitigt sind.
LZ	Der Parameter ist nur im taktsynchronen Betrieb wirksam. Bei eingeschaltetem Lebenszeichen wird im taktsynchronen Betrieb bei jedem eingelesenen Geberwert das Lebenszeichen-Bit getoggelt, d. h., der zuletzt gesendete Wert wird invertiert. Ist über den Parameter "Geberabtastrate" eine Untersetzung parametriert, so wird der Wert nur dann getoggelt, wenn auch tatsächlich ein Geberwert eingelesen wurde.
STS_DI	Das Bit zeigt den Status des Digitaleingangs DI an.
STS_DN	Status Richtung rückwärts; bei Geberwertänderung von größeren zu kleineren Geberpositionen (inkl. Nulldurchgang)
STS_UP	Status Richtung vorwärts; bei Geberwertänderung von kleineren zu größeren Geberpositionen (inkl. Nulldurchgang)
RDY	Das 1SSI ist korrekt parametriert und führt seine Funktionen aus. Die angezeigten Rückmeldungen sind gültig. Bei Fehler Absolutwertgeber ist EXTF auch gesetzt.

6.2.3.3 Diagnose des 1SSI

Diagnose durch LED-Anzeige

LED-Anzeige am 1SSI

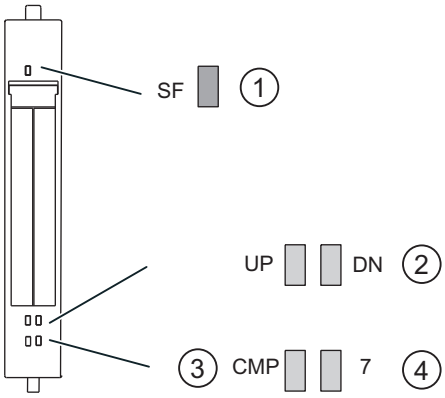

① Sammelfehler (rot)
② Statusanzeige Geberwertänderung (grün)
③ Statusanzeige Vergleichsergebnis
④ Statusanzeige für Digitaleingang (grün)

Status- und Fehleranzeigen durch LEDs am 1SSI

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 1SSI.

Ereignis (LED)					Ursache	Maßnahme
SF	UP	DN	CMP	7		
ein					Keine Parametrierung. Diagnosemeldung liegt vor.	Überprüfen Sie die Parametrierung. Werten Sie die Diagnose aus.
	ein				bei Geberwertänderung von kleineren zu größeren Geberwerten (incl. Nulldurchgang)	
		ein			bei Geberwertänderung von größeren zu kleineren Geberwerten (incl. Nulldurchgang)	
			ein		bei Vergleichsergebnis CMP 1 gesetzt	
				ein	DI (Latch) aktiviert	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

1SSI Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 1SSI.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Versorgung zum Absolutgeber.	Korrektur der Prozessverdrahtung.
9_D	01001: Fehler	Interner Modulfehler ist aufgetreten.	Austausch des Moduls.
		Lastspannung vom Powermodul zu niedrig.	Korrektur der Prozessverdrahtung. Überprüfen Sie die Lastspannung.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.
26_D	11010: externer Fehler	Start-/Stoppbitfehler (Fehler Absolutwertgeber) Drahtbruch des Geberkabels bzw. Geberkabel ist nicht angeschlossen Geberart, Baudrate, Monoflopzeit entsprechen nicht dem angeschlossenen Geber Programmierbare Geber entsprechen nicht den Einstellungen auf dem 1SSI Geber ist defekt oder es liegen Störungen vor.	Austausch des Gebers Korrektur der Prozessverdrahtung Korrektur der Parametrierung

6.2.4 Grundlagen zum Positionieren mit Funktionsbaugruppen

6.2.4.1 Was ist Positionieren?

Positionieren

Positionieren ist das Einnehmen einer konkreten räumlichen, vor allem auch nach Koordinaten festgelegten Position.

Bei SIMATIC Technology verstehen wir unter Positionieren das Verstellen und Positionieren von mechanischen Achsen. Dabei werden verschiedene Verfahren angewendet:

- Gesteuertes Positionieren mit Eil-/Schleichgang
- Lagegeregeltes Positionieren
- Positionieren mit Schrittmotor

Ansteuerbar sind Linearachsen mit begrenztem Verfahrbereich und endlos drehende Rundachsen.

6.2.4.2 Gesteuertes Positionieren mit Eil-/Schleichgang

Gesteuertes Positionieren

Jeder Positioniervorgang ist gekennzeichnet durch:

- eine Startposition
- ein Ziel, auf das positioniert wird
- Parameter, die den Ablauf des Positionierens bestimmen

Das Ziel wird zunächst mit einer höheren Geschwindigkeit, dem Eilgang, angefahren. In einem vorgegebenen Abstand zum Ziel wird auf eine niedrigere Geschwindigkeit, den Schleichgang, umgeschaltet. Kurz bevor die Achse das Ziel erreicht, ebenfalls in einem vorgegebenen Abstand zum Ziel, wird der Antrieb abgeschaltet. Dabei überwacht die Baugruppe den Zieleinlauf.

Der Antrieb wird über Digitalausgänge mit Eilgang oder Schleichgang und der entsprechenden Richtung angesteuert.

6.2.4.3 Bereiche und Schaltpunkte der Positionierbaugruppe

Ziel

Das Ziel ist die absolute bzw. relative Position auf der Achse, die bei einem Positioniervorgang angefahren wird.

Definition der Schaltpunkte und Schaltbereiche

Für gesteuertes Positionieren sind folgende Bereiche und Positionen parametrierbar:

Bereich / Position	Erklärung
Arbeitsbereich	definiert den Bereich, den Sie für Ihre Aufgabe durch die Software-Endschalter bzw. das Ende der Rundachse bestimmen.
Umschaltdifferenz	definiert den Abstand zum Ziel, an dem der Antrieb vom Eilgang auf den Schleichgang umgeschaltet wird.
Umschaltpunkt	definiert die Position, an welcher der Antrieb vom Eilgang auf den Schleichgang umgeschaltet wird.
Abschaltdifferenz	definiert den Abstand zum Ziel, an dem der Antrieb abgeschaltet wird.
Abschaltpunkt	definiert die Position, an welcher der Antrieb abgeschaltet wird. Die Positionierbaugruppe übernimmt ab diesem Punkt Überwachungsfunktionen.
Zielbereich	definiert die Positioniergenauigkeit Ihrer Anwendung und liegt symmetrisch um das Ziel.
Stillstandsbereich	definiert einen symmetrischen Bereich um das Ziel, der von der Positionierbaugruppe überwacht wird.

Schaltpunkte und Schaltdifferenzen

Das folgende Bild zeigt, wie die Schaltpunkte und Schaltdifferenzen für einen Positioniervorgang angeordnet sein können. Zur Vereinfachung wird hier angenommen, dass sich die Istgeschwindigkeit linear über dem verfahrenen Weg ändert. Die entstandenen Rampen erklären sich durch mechanische Trägheit oder durch die Parametriermöglichkeiten des Leistungsteils.

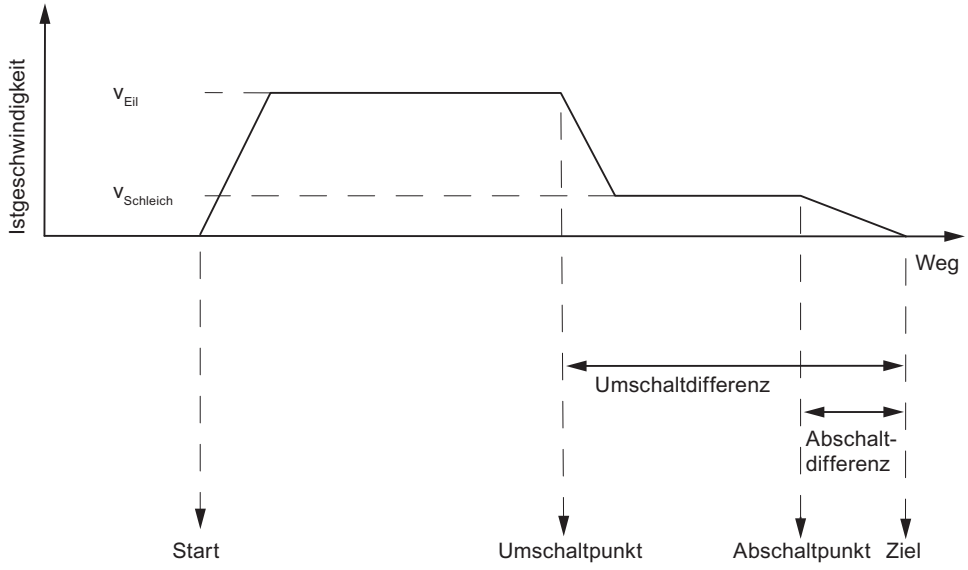

Schaltbereiche

Das folgende Bild zeigt, wie die Schaltbereiche um das Ziel herum angeordnet sein können.

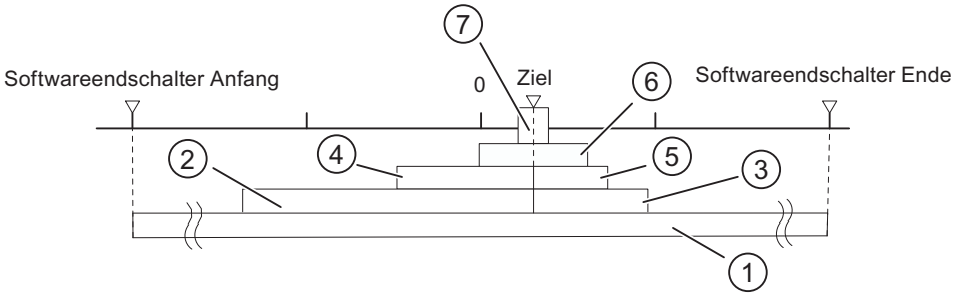

① Arbeitsbereich
② Umschaltdifferenz in Fahrtrichtung plus
③ Umschaltdifferenz in Fahrtrichtung minus
④ Abschaltdifferenz in Fahrtrichtung plus
⑤ Abschaltdifferenz in Fahrtrichtung minus
⑥ Stillstandsbereich
⑦ Zielbereich

6.2.4.4 Gebersignale

Anschließbare Geber

Anschließbare Inkrementalgeber

Es werden Inkrementalgeber (Seite 2470) mit zwei um 90° versetzten Impulsen mit oder ohne Nullmarke unterstützt:

- Geber mit asymmetrischen Ausgangssignalen 24 V-Pegel
  - Grenzfrequenz = 50 kHz
  - Max. 100 m Leitungslänge geschirmt
- Geber mit symmetrischen Ausgangssignalen mit 5 V-Differenzschnittstelle nach RS 422
  - Grenzfrequenz = 400 kHz
  - Bei 5 V-Spannungsversorgung: max. 32 m Leitungslänge geschirmt
  - Bei 24 V-Spannungsversorgung: max. 100 m Leitungslänge geschirmt

Hinweis

Wenn der 5 V-Geber kein Nullmarkensignal ausgibt und Sie die Drahtbruchüberwachung aktiviert haben, dann müssen Sie die Nullmarkeneingänge N und /N extern verschalten, damit diese Eingänge einen unterschiedlichen Pegel aufweisen (z. B. N auf 5 V, /N auf Masse).

Anschließbare Absolutwertgeber

Es werden Absolutwertgeber (Seite 2471) mit serieller Schnittstelle unterstützt. Die Übertragung der Weginformation erfolgt synchron nach dem SSI-Protokoll (**s**ynchron-**s**erielles-**I**nterface). Die FM x51 unterstützt nur GRAY-Code. Durch die Anordnung der Datenbits im Telegrammrahmen ergeben sich die Datenformate 25 Bit (Tannenbaum) und 13 Bit (halber Tannenbaum).

Absolutwertgeber werden unterschieden in

- Singleturn-Geber
  Singleturn-Geber bilden den gesamten Messbereich auf eine Umdrehung des Gebers ab.
- Multiturn-Geber
  Multiturn-Geber bilden den gesamten Messbereich auf mehreren Umdrehungen des Gebers ab.

Geberart	Telegrammlänge
Singleturn-Geber	13 Bit
Singleturn-Geber	25 Bit
Multiturn-Geber	13 Bit … 25 Bit

Die Baudrate zur Datenübertragung ist abhängig von der Leitungslänge:

- Max. 188 kHz bei 200 m Leitungslänge geschirmt
- Max. 375 kHz bei 100 m Leitungslänge geschirmt
- Max. 750 kHz bei 40 m Leitungslänge geschirmt
- Max. 1,5 MHz bei 12 m Leitungslänge geschirmt

Inkrementalgeber

Signalformen

Die folgende Tabelle zeigt die Signalformen von Gebern mit asymmetrischen und symmetrischen Ausgangssignalen.

asymmetrisch	symmetrisch
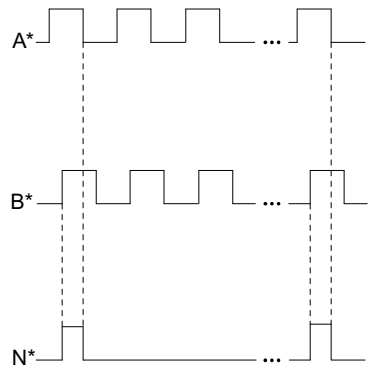	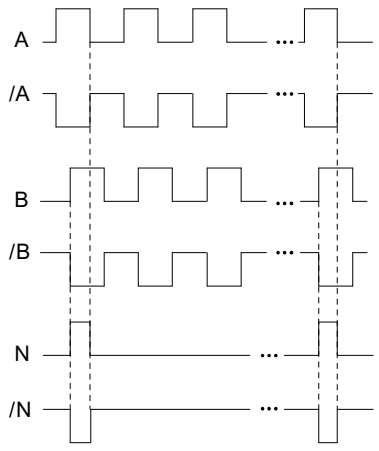

Signalauswertung

Inkremente

Ein Inkrement kennzeichnet eine Signalperiode der beiden Signale A und B eines Gebers. Dieser Wert ist in den Technischen Daten eines Gebers und/oder auf dessen Typenschild angegeben.

Das folgende Bild zeigt das Verhältnis von Inkrementen und Impulsen.

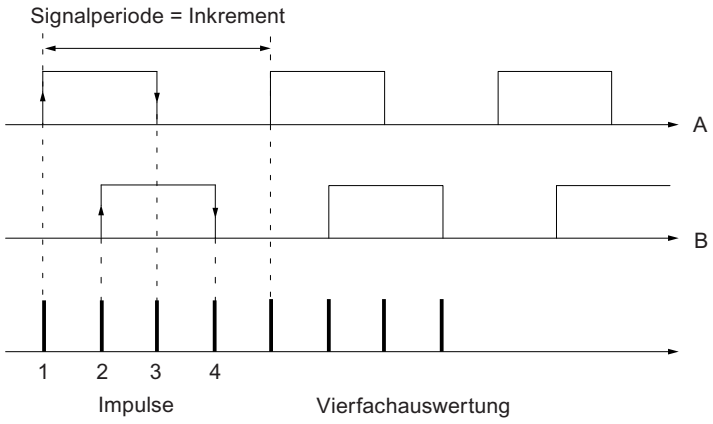

Impulse

Die Positionierbaugruppe wertet alle 4 Flanken der Signale A und B in jedem Inkrement aus (Vierfachauswertung).

Impulse
1 Inkrement (Gebervorgabe) = 4 Impulse (Auswertung auf der Baugruppe)

Reaktionszeiten

Die Positionierbaugruppe hat für angeschlossene Inkrementalgeber folgende Reaktionszeiten:

Reaktionszeiten
Reaktionszeit = Schaltzeit der angeschlossenen Schaltelemente

Hinweis

Die minimale Reaktionszeit können Sie durch entsprechende Parametrierung der Umschaltdifferenz und der Abschaltdifferenz kompensieren.

Unschärfe

Die Unschärfe beeinflusst die Genauigkeit der Positionierung. Bei Inkrementalgebern ist die Unschärfe vernachlässigbar.

Absolutwertgeber

Impulsauswertung Absolutwertgeber

Impulsauswertung Absolutwertgeber
1 Inkrement (Gebervorgabe) = 1 Impuls (FM-Auswertung)

Reaktionszeiten

Die FM x51 hat für Absolutwertgeber folgende Reaktionszeiten:

Reaktionszeiten
Minimale Reaktionszeit = Telegrammlaufzeit + Schaltzeit der angeschlossenen Schaltelemente
Maximale Reaktionszeit = 2 × Telegrammlaufzeit + Monoflopzeit + Schaltzeit der angeschlossenen Schaltelemente
Bei programmierbaren Absolutwertgebern: Maximale Reaktionszeit = Telegrammlaufzeit + Monoflopzeit + Schaltzeit der angeschlossenen Schaltelemente +1/max. Schrittfolgefrequenz

Monoflopzeit

Die Monoflopzeit beträgt 64 µs.

Geber mit Werten größer der hier genannten Grenze sind nicht zulässig.

Telegrammlaufzeiten

Die Telegrammlaufzeiten sind abhängig von der Baudrate:

Baudrate	Telegrammlaufzeit bei 13 Bit	Telegrammlaufzeit bei 25 Bit
0,188 MHz	75 µs	139 µs
0,375 MHz	38 µs	70 µs
0,750 MHz	19 µs	35 µs
1,500 MHz	10 µs	18 µs

Beispiel Reaktionszeiten

Dieses Beispiel zeigt Ihnen, wie Sie die minimale und maximale Reaktionszeit berechnen. Im Beispiel wird kein programmierbarer Geber verwendet.

- Schaltzeit der Hardware: ca. 150 µs
- Telegrammlaufzeit: 18 µs bei 1,5 MHz Baudrate (25 Bit-Telegrammrahmen)
- Monoflopzeit: 64 µs

Minimale Reaktionszeit = 18 µs + 150 µs = 168 µs

Maximale Reaktionszeit = 2 × 18 µs + 64 µs + 150 µs = 250 µs

Hinweis

Die minimale Reaktionszeit können Sie durch entsprechende Parametrierung der Um- und Abschaltdifferenz kompensieren.

Unschärfe

Die Unschärfe ist die Differenz aus maximaler und minimaler Reaktionszeit. Bei einem Absolutwertgeber beträgt sie

Unschärfe
Unschärfe = Telegrammlaufzeit + Monoflopzeit
Bei programmierbaren Absolutwertgebern: Unschärfe = Telegrammlaufzeit + Monoflopzeit + 1/max. Schrittfolgefrequenz

6.2.5 FM 351 einsetzen

6.2.5.1 Anschlussbelegung der Frontstecker der FM 351

Beschreibung der Geberschnittstelle

Lage der D-SUB-Buchsen

Das folgende Bild zeigt die Einbaulage und die Bezeichnung der Buchsen auf der Baugruppe. An die beiden D-SUB-Buchsen können Sie Inkrementalgeber oder Absolutwertgeber (SSI) anschließen.

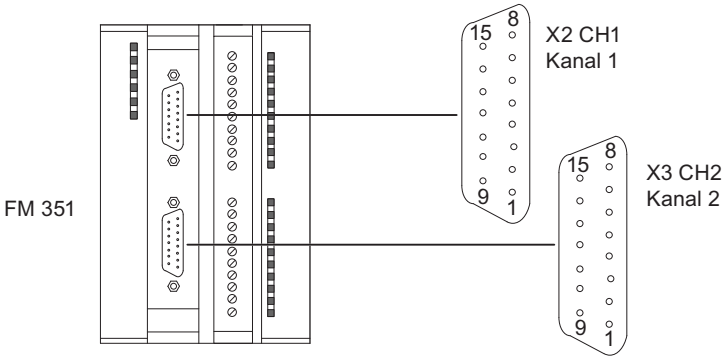

Belegung der Geberschnittstellen (X2 und X3)

Pin	Name	Inkrementalgeber (24 V)	Inkrementalgeber (5 V)	Absolutwertgeber
1	A*	Gebersignal A	---	---
2	CLS	---	---	SSI-Schiebetakt
3	/CLS	---	---	SSI-Schiebetakt invers
4	B*	Gebersignal B	---	---
5	DC 24 V	Geberversorgung	Geberversorgung	Geberversorgung
6	DC 5.2 V	---	Geberversorgung	Geberversorgung
7	M	Masse	Masse	Masse

Pin	Name	Inkrementalgeber (24 V)	Inkrementalgeber (5 V)	Absolutwertgeber
8	N*	Nullmarkensignal	---	---
9	RE	P/M-schaltend [1]	---	---
10	N	---	Nullmarkensignal	---
11	/N	---	Nullmarkensignal invers	---
12	/B	---	Gebersignal B invers	---
13	B	---	Gebersignal B	---
14	/A / /DAT	---	Gebersignal A invers	SSI-Daten invers
15	A / DAT	---	Gebersignal A	SSI-Daten

[1] **Hinweis:** Wenn Sie einen Inkrementalgeber eines Fremdherstellers in Gegentaktschaltung (P/M-schaltend) anschließen, müssen Sie Folgendes beachten:

- P-schaltend: Verbinden Sie RE (9) mit Masse (7).
- M-schaltend: Verbinden Sie RE (9) mit +24 V (5).

Belegung des Frontsteckers

Frontstecker

Am 20-poligen Frontstecker schließen Sie an:

- die den Kanälen zugeordnete digitale Ausgänge und Eingänge
- die Versorgungsspannungen der Geber und der Digitalausgänge

Anschlussbelegung des Frontsteckers (X1)

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 351.

An-schluss	Name	Funktion	Inkrementalgeber	Absolutwertgeber
1	1L+	DC 24 V Hilfsspannung für die Geberversorgung		
2	1M	Masse Geberversorgung		
3	1I0	Kanal 1: Digitaleingang 0	Referenzpunktschalter	nicht verwendet
4	1I1	Kanal 1: Digitaleingang 1	Umkehrschalter	nicht verwendet
5	1I2	Kanal 1: Digitaleingang 2	Freigabeeingang	
6	1I3	Kanal 1: Digitaleingang 3	nicht verwendet	
7	2I0	Kanal 2: Digitaleingang 0	Referenzpunktschalter	nicht verwendet
8	2I1	Kanal 2: Digitaleingang 1	Umkehrschalter	nicht verwendet
9	2I2	Kanal 2: Digitaleingang 2	Freigabeeingang	
10	2I3	Kanal 2: Digitaleingang 3	nicht verwendet	
11	1Q0	Kanal 1: Digitalausgang 0		
12	1Q1	Kanal 1: Digitalausgang 1		
13	1Q2	Kanal 1: Digitalausgang 2		
14	1Q3	Kanal 1: Digitalausgang 3		
15	2Q0	Kanal 2: Digitalausgang 0		
16	2Q1	Kanal 2: Digitalausgang 1		

An-schluss	Name	Funktion	Inkrementalgeber	Absolutwertgeber
17	2Q2	Kanal 2: Digitalausgang 2		
18	2Q3	Kanal 2: Digitalausgang 3		
19	2L+	DC 24 V Hilfsspannung für die Laststromversorgung		
20	2M	Masse Lastromversorgung		

Weitere Informationen zu Aufbau und Verdrahtung finden Sie in der Betriebsanleitung (http://support.automation.siemens.com/WW/view/de/2103043) der Positionierbaugruppe FM 351.

Hilfsspannung für die Geberversorgung (1L+, 1M)

Hier schließen Sie eine DC 24 V-Hilfsspannung für die Geberversorgung an. Das Bezugspotential dieser Versorgung (1M) ist in der FM 351 mit der Masse der Laststromversorgung (2M) nicht verbunden.

Die DC 24 V-Hilfsspannung für die Geberversorgung wird auf Unterspannung und Masse-Drahtbruch überwacht.

An der Geberschnittstelle (D-SUB-Buchsen X2 und X3) werden DC 24 V und DC 5,2 V für die unterschiedlichen Geberarten zur Verfügung gestellt.

⚠ VORSICHT

Achten Sie auf die richtige Polung der DC 24 V-Hilfsspannung für die Geberversorgung (1L+, 1M).

Wenn Sie die DC 24 V-Hilfsspannung für die Geberversorgung verpolt anschließen, dann wird die Baugruppe beschädigt und muss ausgetauscht werden.

Hilfsspannung für die Laststromversorgung (2L+, 2M)

An die Klemmen 2L+ und 2M müssen Sie eine DC 24 V-Hilfsspannung für die Laststromversorgung der Digitalausgänge anschließen.

⚠ VORSICHT

Achten Sie auf die richtige Polung der DC 24 V-Hilfsspannung für die Laststromversorgung (2L+, 2M).

Wenn Sie die DC 24 V-Hilfsspannung für die Laststromversorgung verpolt anschließen, dann wird die Baugruppe beschädigt und muss ausgetauscht werden.

Verdrahtungshinweis für DC 24 V

Beachten Sie bei der Verdrahtung, dass die Klemmen 1L+, 1M und 2L+, 2M verschaltet sein müssen, damit die Baugruppe fehlerfrei läuft.

Wenn Sie 1L+, 1M und 2L+, 2M an getrennte Spannungsversorgungen anschließen, bleibt nach einem Spannungsausfall an der Hilfsspannung für die Laststromversorgung (2L+, 2M) die Synchronisation der Achsen erhalten.

Laststromversorgungen

Die DC-Laststromversorgung muss folgenden Anforderungen genügen:

Als Laststromversorgung darf nur vom Netz sicher getrennte Kleinspannung DC ≤ 60 V verwendet werden. Die sichere Trennung kann realisiert sein nach den Anforderungen u. a. in

- VDE 0100 Teil 410 / HD 384-4-41 / IEC 364-4-41
  (als Funktionskleinspannung mit sicherer Trennung) bzw.
- VDE 0805 / EN 60950 / IEC 950
  (als Sicherheitskleinspannung SELV) bzw.
- VDE 0106 Teil 101.

8 Digitaleingänge (1I0 bis 2I3)

Die FM 351 verfügt je Kanal über 4 Digitaleingänge.

An die 8 Digitaleingänge können Sie prellfreie Schalter (24 V P-schaltend) oder berührungslose Sensoren (2- oder 3-Draht Näherungsschalter) anschließen.

Die Digitaleingänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgetrennt zur Masse der Geberversorgung und zur Masse der CPU.

Der Zustand jedes Eingangs ist an der zugehörigen LED ablesbar.

8 Digitalausgänge (1Q0 bis 2Q3)

Die FM 351 verfügt je Kanal über 4 Digitalausgänge.

Mit den Digitalausgängen wird das Leistungsteil angesteuert. Die Funktion der Digitalausgänge ist abhängig von der Ansteuerart (Seite 2515).

Die Digitalausgänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgetrennt zur Masse der Geberversorgung und zur Masse der CPU.

Der Zustand jedes Ausgangs ist an der zugehörigen LED ablesbar.

6.2.5.2 Funktionen der FM 351

Funktionsübersicht

Mit der Vorgabe einer Betriebsart, eines Auftrags oder Funktionsschalters wählen Sie, mit welcher Funktionalität Sie die Baugruppe betreiben wollen. Die folgende Tabelle zeigt einen Überblick über die Funktionalität der Baugruppe.

Funktionalität	Beschreibung
Maschinendaten schreiben und lesen (Seite 2477)	Mit den Maschinendaten passen Sie die Positionierbaugruppe an die Achse und den Geber an.
Schrittmaßtabellen schreiben und lesen (Seite 2479)	Mit den Schrittmaßen geben Sie Ziele vor, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ / absolut" angesteuert werden.

Funktionalität	Beschreibung
Ende eines Positioniervorgangs (Seite 2481)	Das Ende eines Positioniervorgangs wird auf drei verschiedene Arten erreicht: • Zieleinlauf • Absteuern • Abbrechen
Betriebsart "Tippen" (Seite 2488)	In der Betriebsart "Tippen" bewegen Sie den Antrieb auf Tastendruck in eine Richtung.
Betriebsart "Referenzpunktfahrt" (Seite 2491)	Mit der Betriebsart "Referenzpunktfahrt" synchronisieren Sie die Achse durch ein wiederkehrendes externes Ereignis.
Betriebsart "Schrittmaßfahrt" (Seite 2498)	In der Betriebsart "Schrittmaßfahrt" kann die Positionierbaugruppe den Antrieb • auf absolute Ziele bewegen, • relativ um ein Wegstück in eine vorgegebene Richtung bewegen.
Auftrag "Istwert setzen" / "Istwert setzen rückgängig" (Seite 2503)	Mit dem Auftrag "Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Mit dem Auftrag "Istwert setzen rückgängig" setzen Sie alle Arbeitsbereichverschiebungen, die durch "Istwert setzen" erzeugt wurden, wieder zurück.
Auftrag "Bezugspunkt setzen" (Seite 2505)	Mit dem Auftrag "Bezugspunkt setzen" synchronisieren Sie die Achse.
Funktionsschalter "Schleifenfahrt" (Seite 2507)	Mit "Schleifenfahrt" legen Sie die Richtung fest, in die ein Ziel mit Kraftschluss angefahren wird.
Funktionsschalter "Freigabeeingang" (Seite 2510)	Der Freigabeeingang ist ein externer Eingang, mit dem eine Positionierung durch ein externes Ereignis freigegeben werden kann.
Auftrag "Positionsdaten lesen" (Seite 2510)	Mit dem Auftrag "Positionsdaten lesen" können Sie Schrittmaß, Restweg und Geschwindigkeit zum aktuellen Zeitpunkt lesen.
Auftrag "Geberdaten lesen" (Seite 2511)	Mit dem Auftrag "Geberdaten lesen" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.
Rückmeldesignale für die Positionierung (Seite 2511)	Mit den "Rückmeldesignalen für die Positionierung" werden Sie über den aktuellen Zustand der Positionierung informiert.
Rückmeldesignale für die Diagnose (Seite 2512)	Mit den "Rückmeldesignalen für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Maschinendaten schreiben und lesen

Maschinendaten schreiben, aktivieren und lesen

Mit den Maschinendaten passen Sie die Positionierbaugruppe an die Achse und den Geber an.

Über das Anwenderprogramm können Sie die Parameter während des Betriebs ändern und auslesen.

Alle Parameter sind im Parameter-DB abgelegt:

- Maschinendaten liegen im Parameter-DB auf den Adressen 4.0 bis 116.0.

Die Nummer des Parameter-DB müssen Sie in den jeweils zugehörigen Kanal-DB eintragen.

Die Parameter können Sie mit dem DB-Editor eingeben oder auch komfortabel in den Dialogen "Antrieb", "Achse" und "Geber" eintragen und mit der Funktion "Export" in den Parameter-DB schreiben.

Die Parameter in einem bereits vorhandenen Parameter-DB können Sie mit der Funktion "Import" in die Dialoge importieren.

Erstparametrierung

Falls der Kanal noch keine Maschinendaten enthält, gehen Sie bei einer Erstparametrierung ohne die Parametrierdialoge folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein und speichern diesen ab.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie das folgende Anstoßbit im Kanal-DB für den Auftrag:
   – Maschinendaten schreiben (MDWR_EN = 1).
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Maschinendaten ändern

Um Maschinendaten per Anwenderprogramm zu ändern, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   – Maschinendaten schreiben (MDWR_EN = 1)
   – Maschinendaten aktivieren (MD_EN = 1)
3. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Wenn Sie die Anstoßbits für diese Aufträge gleichzeitig setzen, sorgt die Anweisung ABS_CTRL dafür, dass die Aufträge in der richtigen Reihenfolge abgearbeitet werden.

Ansonsten ändern Sie die Maschinendaten immer in folgender Reihenfolge:

- Maschinendaten schreiben
- Maschinendaten aktivieren

Maschinendaten lesen

Um die aktuellen Maschinendaten aus einem Kanal zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie das folgende Anstoßbit im Kanal-DB:
   – Maschinendaten lesen (MDRD_EN = 1)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Damit werden die aktuellen Maschinendaten im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
36.5	MDRD_EN	BOOL	FALSE	1 = Maschinendaten lesen

Hinweis

Wenn synchronisationsrelevante Parameter verändert werden, dann werden beim Aktivieren der Maschinendaten folgende Aktionen von der Baugruppe für den betroffenen Kanal durchgeführt:

- Die Synchronisation wird gelöscht.
- Die Funktionsschalter und die Nullpunktverschiebung werden zurückgesetzt.
- Alle bisherigen Maschinendaten und Schrittmaßtabellen werden ungültig.

Synchronisationsrelevante Parameter sind:

- Achsart
- Ende der Rundachse
- Signalart
- Weg pro Geberumdrehung
- Inkremente pro Geberumdrehung
- Anzahl Umdrehungen
- Referenzpunktkoordinate
- Absolutwertgeberjustage
- Art der Referenzpunktfahrt
- Zählrichtung

Schrittmaßtabellen schreiben und lesen

Schrittmaßtabellen schreiben und lesen

Mit den Schrittmaßen geben Sie Ziele vor, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ / absolut" angesteuert werden.

Über das Anwenderprogramm können Sie die Parameter während des Betriebs ändern und auslesen.

Alle Parameter sind im Parameter-DB abgelegt:

- Schrittmaßtabellen liegen im Parameter-DB auf den Adressen 120.0 bis 516.0.

Die Nummer des Parameter-DBs müssen Sie in den jeweils zugehörigen Kanal-DB eintragen.

Die Parameter können Sie mit dem DB-Editor eingeben oder auch komfortabel in den Dialog "Schrittmaße" eintragen und mit der Funktion "Export" in den Parameter-DB schreiben.

Die Parameter in einem bereits vorhandenen Parameter-DB können Sie mit der Funktion "Import" in den Dialog importieren.

Erstparametrierung

Falls der Kanal noch keine Schrittmaßtabellen enthält, gehen Sie bei einer Erstparametrierung ohne den Parametrierdialog folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein und speichern diesen ab.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   – Schrittmaßtabelle 1 schreiben (TRGL1WR_EN = 1) und / oder Schrittmaßtabelle 2 schreiben (TRGL2WR_EN = 1)
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Schrittmaßtabellen ändern

Um Schrittmaßtabellen per Anwenderprogramm zu ändern, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   – Schrittmaßtabelle 1 schreiben (TRGL1WR_EN = 1) und / oder Schrittmaßtabelle 2 schreiben (TRGL2WR_EN = 1)
3. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Schrittmaßtabellen lesen

Um die Schrittmaßtabellen aus einem Kanal zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   – Schrittmaßtabelle 1 lesen (TRGL1RD_EN = 1) und / oder Schrittmaßtabelle 2 lesen (TRGL2RD_EN = 1)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL auf.

Damit werden die Schrittmaßtabellen im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (1 ... 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (51 ... 100) schreiben
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (1 ... 50) lesen
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (51 ... 100) lesen

Hinweis

Wenn synchronisationsrelevante Parameter verändert werden, dann werden beim Aktivieren der Maschinendaten folgende Aktionen von der Baugruppe für den betroffenen Kanal durchgeführt:

- Die Synchronisation wird gelöscht.
- Die Funktionsschalter und die Nullpunktverschiebung werden zurückgesetzt.
- Alle bisherigen Maschinendaten und Schrittmaßtabellen werden ungültig.

Synchronisationsrelevante Parameter sind:

- Achsart
- Ende der Rundachse
- Signalart
- Weg pro Geberumdrehung
- Inkremente pro Geberumdrehung
- Anzahl Umdrehungen
- Referenzpunktkoordinate
- Absolutwertgeberjustage
- Art der Referenzpunktfahrt
- Zählrichtung

Ende eines Positioniervorgangs

Definition

Das Ende eines Positioniervorgangs wird durch das Rückmeldesignal WORKING = 0 angezeigt. Es wird auf drei verschiedene Arten erreicht:

- Zieleinlauf
- Absteuern
- Abbrechen

Überwachungen

Zum Ende eines Positioniervorgangs hin sind folgende Überwachungen aktiv:

- Überwachungszeit
  Die Überwachungszeit wird am Abschaltpunkt zum letzten Mal nachgetriggert und verliert ihre Gültigkeit mit dem Ende des Positioniervorgangs.
  Innerhalb dieser Zeit muss das Ende des Positioniervorgangs erreicht werden. Sonst werden die Ausgänge abgeschaltet und der Betriebsfehler "Fehler beim Zieleinlauf" (Fehlernummer 5) gemeldet.
- Überwachung des Zielbereichs
  Die FM x51 legt um jedes Ziel einen symmetrischen Bereich und definiert damit die Positioniergenauigkeit Ihrer Anwendung. Innerhalb dieses Bereichs muss die Achse bei einem Zieleinlauf zum Stillstand kommen. Eine Wertvorgabe 0 schaltet die Toleranz beim Zieleinlauf ab.
- Überwachung der Stillstandsgeschwindigkeit
  Über die Stillstandsgeschwindigkeit lässt sich feststellen, ob der Antrieb innerhalb des Zielbereichs zum Stillstand kommt. Nach dem Erreichen des Abschaltpunktes überprüft die FM x51, ob die Istgeschwindigkeit des Antriebs kleiner ist als die Stillstandsgeschwindigkeit.
  Die Stillstandsgeschwindigkeit muss innerhalb des Zielbereichs unterschritten werden. Sonst meldet die FM x51 den Betriebsfehler "Zielbereich überfahren" (Fehlernummer 10).
  Das Unterschreiten der Stillstandsgeschwindigkeit wird nur einmal je Zieleinlauf überwacht.
  Beachten Sie hierbei, dass die Stillstandsgeschwindigkeit für die Geschwindigkeitsermittlung der Baugruppe kurzzeitig unterschritten werden kann, wenn die Achse sich mit sehr kleiner Positioniergeschwindigkeit bewegt (weniger als 2 Impulse pro 8 ms).
- Überwachung des Stillstandsbereichs
  Nach dem Ende eines Positioniervorgangs wird überwacht, ob der Antrieb auf einer angefahrenen Zielposition stehen bleibt oder von dieser wegdriftet.
  Der Stillstandsbereich wird überwacht,
  - nachdem die FM x51 das Rückmeldesignal "PEH" gemeldet hat,
  - wenn die Überwachungszeit überschritten wird,
  - wenn die Stillstandsgeschwindigkeit unterschritten wird.

  Wenn der Stillstandsbereich ohne einen gültigen Fahrauftrag verlassen wird, meldet die FM x51 den Betriebsfehler "Stillstandsbereich verlassen" (Fehlernummer 6).

Zieleinlauf

Der Zieleinlauf in den Betriebsarten "Schrittmaßfahrt absolut" und "Schrittmaßfahrt relativ" beginnt mit dem Erreichen des Abschaltpunktes. Ab diesem Punkt wird der Antrieb abgeschaltet, und die FM x51 übernimmt Überwachungsfunktionen.

Je nachdem, welche Überwachungen Sie parametriert haben, ergeben sich verschiedene Fälle für die Erzeugung des Rückmeldesignals "PEH (POS_RCD)". Der Positioniervorgang wird abgebrochen, wenn kein Rückmeldesignal "PEH (POS_RCD)" erzeugt wird.

Fall 1: Sie haben parametriert:

- Zielbereich (TRG_RANGE) > 0
- Stillstandsgeschwindigkeit (ZSPEED_L) > 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn die Stillstandsgeschwindigkeit unterschritten und der Zielbereich erreicht wird. Welche Bedingung zuerst erfüllt wird, spielt hierbei keine Rolle.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit den Zielbereich nicht erreicht bzw. der Zielbereich überfahren wird, ohne die Stillstandsgeschwindigkeit zu unterschreiten.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich > 0 und Stillstandsgeschwindigkeit > 0.

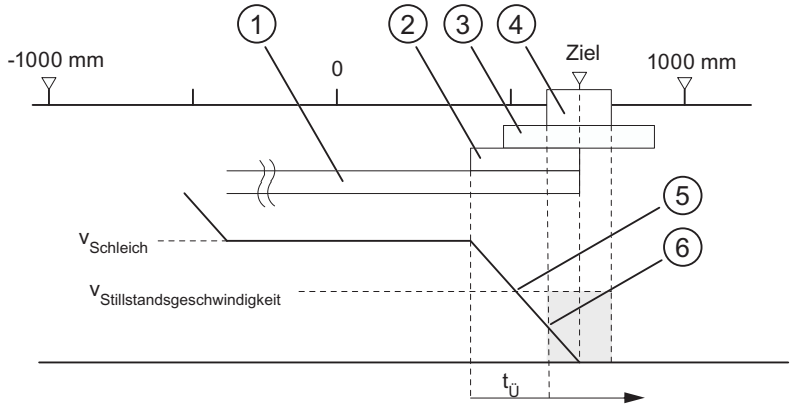

①	Umschaltdifferenz plus
②	Abschaltdifferenz plus
③	Stillstandsbereich
④	Zielbereich
⑤	Stillstandsgeschwindigkeit erreicht
⑥	Zielbereich mit V_{Still} erreicht: PEH wird gesetzt
$t_Ü$	Überwachungszeit

Fall 2: Sie haben parametriert:

- Zielbereich (TRG_RANGE) > 0
- Stillstandsgeschwindigkeit (ZSPEED_L) = 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn der Zielbereich erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit den Zielbereich nicht erreicht.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich > 0 und Stillstandsgeschwindigkeit = 0.

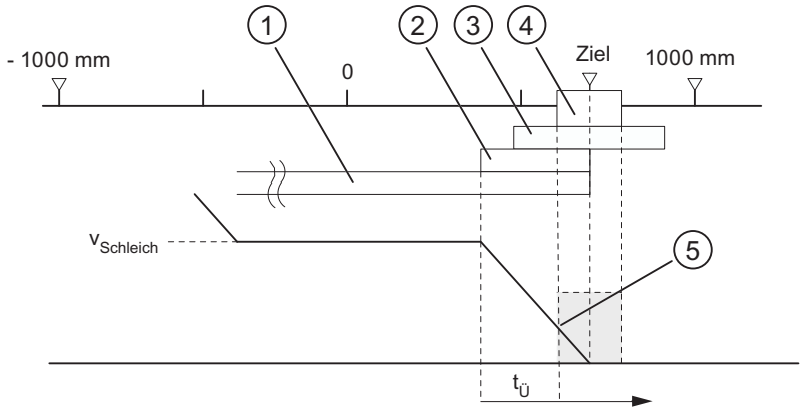

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Zielbereich
⑤ Zielbereich mit V_{Still} erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 3: Sie haben parametriert:

- Zielbereich (TRG_RANGE) = 0
- Stillstandsgeschwindigkeit (ZSPEED_L) > 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn die Stillstandsgeschwindigkeit unterschritten und dann das Ziel erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit das Ziel nicht erreicht bzw. der Zielbereich überfahren wird, ohne die Stillstandsgeschwindigkeit zu unterschreiten.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich = 0 und Stillstandsgeschwindigkeit > 0.

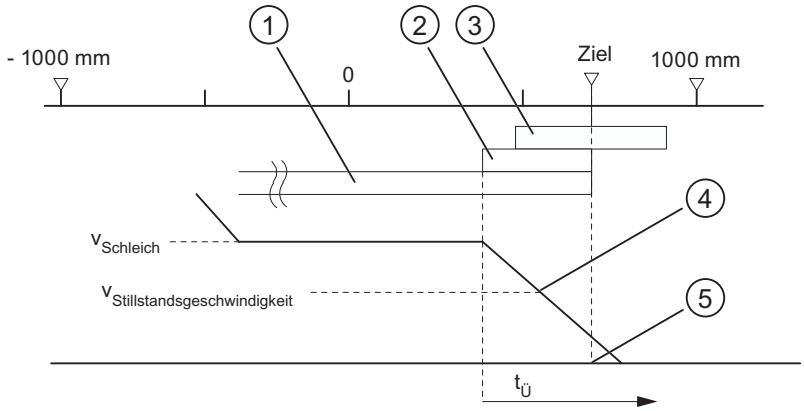

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Stillstandsgeschwindigkeit erreicht
⑤ Ziel erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 4: Sie haben parametriert:

- Zielbereich (TRG_RANGE) = 0
- Stillstandsgeschwindigkeit (ZSPEED_L) = 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn das Ziel erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit das Ziel nicht erreicht.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich = 0 und Stillstandsgeschwindigkeit = 0.

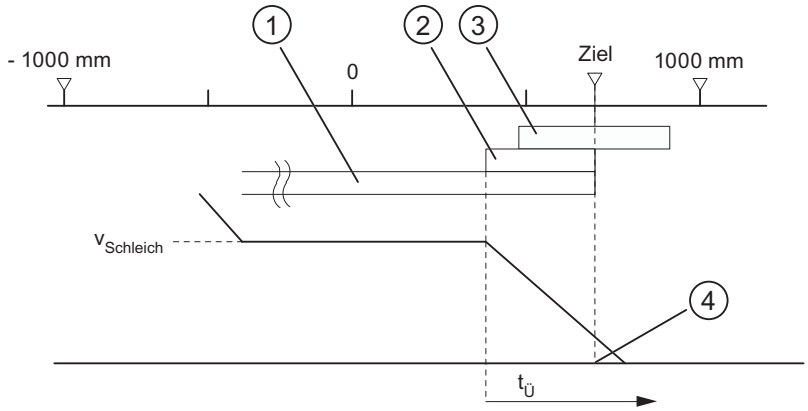

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Ziel erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 5: Sie haben parametriert:

- Zielbereich (TRG_RANGE) ≥ 0
- Stillstandsgeschwindigkeit (ZSPEED_L) ≥ 0
- Überwachungszeit (MON_TIME) = 0

Wenn in diesem Fall der Antrieb vor dem Zielbereich zum Stillstand kommt, wird das Ende des Positioniervorgangs nicht erkannt. PEH wird nicht generiert und das Rückmeldesignal WORKING bleibt gesetzt. Sie können den Positioniervorgang nur durch Löschen der Antriebsfreigabe abbrechen (DRV_EN = 0).

Absteuern ohne vorgegebenes Ziel

Absteuern bedeutet: Der Positioniervorgang wird gezielt unter Einhaltung der Differenzen von Eilgang über Schleichgang beendet.

Der Positioniervorgang wird abgesteuert, wenn

- die FM x51 ein STOP-Signal empfängt (STOP = 1).
- die Betriebsarten "Tippen" und "Referenzpunktfahrt" beendet werden.
- Bedienfehler oder der Betriebsfehler "Ziel überfahren" (Fehlernummer 9) auftreten.

Das Rückmeldesignal "PEH (POS_RCD)" wird nicht gesetzt. Die Abläufe sind analog zum Zieleinlauf.

Das folgende Bild zeigt das Absteuern eines Positioniervorgangs.

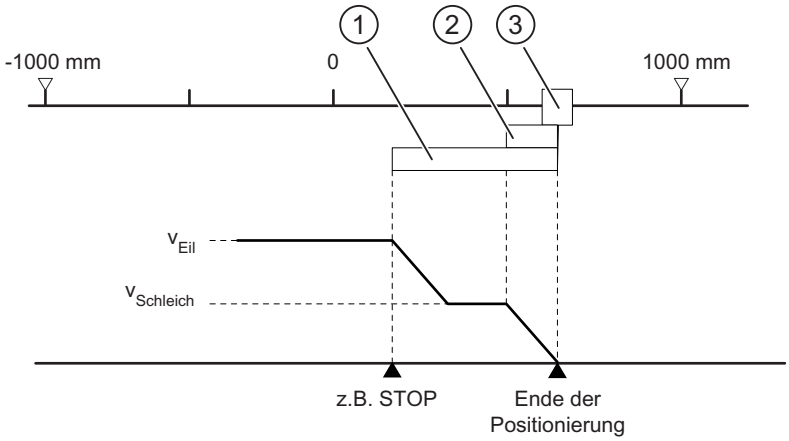

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Zielbereich

Abbrechen

Abbrechen bedeutet: Der Positioniervorgang wird sofort ohne Verwendung der Umschalt- und Abschaltdifferenz von Eilgang bzw. Schleichgang nach Stillstand beendet. Dazu werden alle relevanten Ausgänge der jeweiligen Ansteuerart sofort ausgeschaltet und außerdem wird:

- Schrittmaß = Istwert
- Restweg = Null

Der Positioniervorgang wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0).
- die CPU nach STOP geht.
- Diagnosefehler oder alle Betriebsfehler außer dem Betriebsfehler "Ziel überfahren" (Fehlernummer 9) auftreten.

Das Rückmeldesignal "PEH (POS_RCD)" wird bei der Betriebsart "Schrittmaßfahrt" nicht gesetzt.

Wenn die Stillstandsgeschwindigkeit parametriert ist, wird die Stillstandsüberwachung aktiv, sobald diese unterschritten ist. Falls die Stillstandsgeschwindigkeit nicht parametriert ist, wird die Stillstandsüberwachung mit dem Abschalten der Ausgänge aktiv.

Das folgende Bild zeigt das Abbrechen eines Positioniervorgangs.

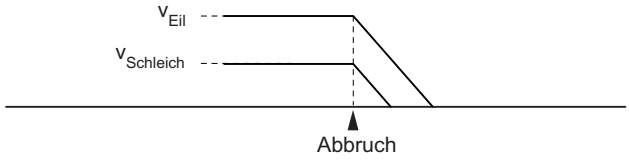

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	TRG_RANGE	DINT	L#1000	Zielbereich
80.0	MON_TIME	DINT	L#2000	Überwachungszeit
84.0	ZSPEED_R	DINT	L#1000	Stillstandsbereich
88.0	ZSPEED_L	DINT	L#30000	Stillstandsgeschwindigkeit

Rückmeldesignale im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht

Betriebsart "Tippen"

Definition

In der Betriebsart "Tippen" bewegen Sie den Antrieb auf Tastendruck in eine Richtung. Für beide Richtungen (plus und minus) müssen Sie je einen Taster installieren. Die Betriebsart "Tippen" können Sie sowohl für eine synchronisierte als auch für eine nicht synchronisierte Achse verwenden.

Bei nicht synchronisierter Achse ist Tippen ein Positionieren in die vorgegebene Richtung.

Bei synchronisierter Achse (Linearachse) ist Tippen ein Positionieren auf die Software-Endschalter.

Voraussetzung

Die Achse muss parametriert sein.

Ablauf der Betriebsart "Tippen"

1. Setzen Sie das Steuersignal für die Betriebsart "Tippen" (MODE_IN = 1).
2. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
3. Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie für den entsprechenden Kanal den Freigabeeingang.
4. Tragen Sie die Startgeschwindigkeit ein.
   - Eilgang (MODE_TYPE = 1)
   - Schleichgang (MODE_TYPE = 0)
5. Setzen Sie das Steuersignal für die Fahrtrichtung plus oder minus (DIR_P = 1 oder DIR_M = 1).
6. Rufen Sie die Anweisung ABS_CTRL auf.

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Tippen".

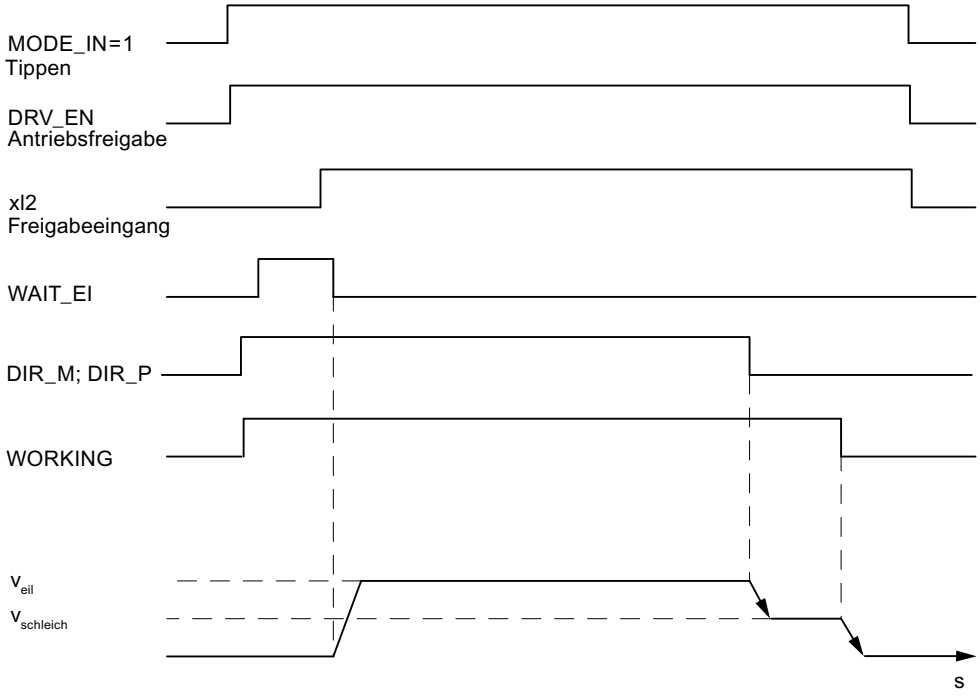

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	1 = Tippen
17.0	MODE_TYPE	BYTE	B#16#0	1 = Eilgang 0 = Schleichgang
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Absteuern des Tippens

Die Betriebsart "Tippen" wird abgesteuert, wenn

- Sie den Taster, mit dem Sie "Tippen", wieder öffnen (DIR_M = 0 bzw. DIR_P = 0),
- die FM 351 ein STOP-Signal empfängt (STOP = 1),
- der Istwert bei einer synchronisierten Linearachse die Grenze des Arbeitsbereichs erreicht. Eine Weiterfahrt ist nur in der entgegengesetzten Richtung möglich.

Nach dem Absteuern der Fahrt ist eine Weiterfahrt in einer beliebigen Richtung möglich.

Abbrechen des Tippens

Die Betriebsart "Tippen" wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0),
- eine Verfahrbereichsgrenze bei einer Linearachse überfahren wurde.

Überwachungen

Bei der Betriebsart "Tippen" sind am Ende der Positionierung folgende Überwachungen nicht aktiv:

- Überwachung des Stillstandsbereichs
- Überwachung des Zielbereichs
- Überwachung der Stillstandsgeschwindigkeit

Tippen führt nicht zu einer ordnungsgemäßen Positionierung, bei der am Ende das Signal "Position erreicht" gesetzt wird.

Arbeitsbereichsgrenze bei einer Linearachse

Die Grenzen für die Betriebsart "Tippen" unterscheiden sich zwischen einer synchronisierten und einer nicht synchronisierten Achse.

Die folgende Tabelle zeigt "Tippen" bei synchronisierter und nicht synchronisierter Achse.

Achse ist nicht synchronisiert	Achse ist synchronisiert
Wenn beim Tippen die Verfahrbereichsgrenze überfahren wird, • ist die Istwertanzeige nicht mehr gültig. • wird der Positioniervorgang abgebrochen. 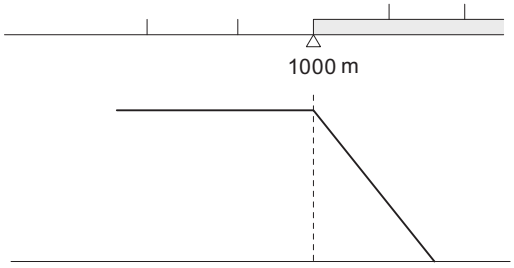 Die Istwertanzeige ist nicht mehr gültig	Tippen ist ein Positionieren auf Ziele, die um den ganzen Zielbereich von den Software-Endschaltern entfernt liegen. Die Arbeitsbereichsgrenzen ermitteln sich aus • **SEE - ½ Zielbereich** für das Ende der Linearachse in Richtung plus • **SEE + ½ Zielbereich** für das Ende der Linearachse in Richtung minus Wenn Sie nicht vorher den Taster öffnen, steuert die FM 351 auf einen Zielpunkt ab, der um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegt. Alle Bereiche, die zum korrekten Absteuern notwendig sind, werden von der FM 351 um diesen Zielpunkt gesetzt. 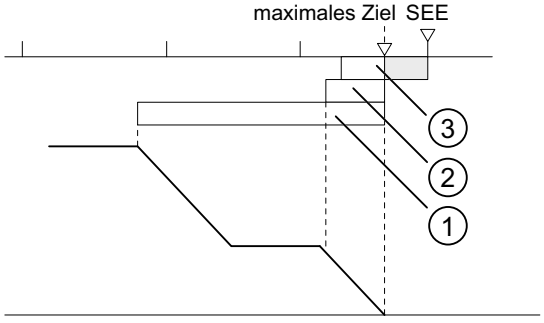 = Teil des Arbeitsbereichs, in dem keine Zielposition liegen darf ① = Umschaltdifferenz plus ② = Abschaltdifferenz plus ③ = ½ Zielbereich

Betriebsart "Referenzpunktfahrt"

Definition

Mit der Betriebsart "Referenzpunktfahrt" synchronisieren Sie die Achse durch ein wiederkehrendes externes Ereignis.

Voraussetzungen

- Ein Inkrementalgeber mit Nullmarke.
- Die Achse muss parametriert sein:

Anschlüsse	Kanal 1	Kanal 2
Referenzpunktschalter	Digitaleingang 1I0	Digitaleingang 2I0
	Der Referenzpunktschalter soll so beschaffen sein, dass der Antrieb im Bereich des Schalters sicher vom Eilgang auf Schleichgang abbremsen kann.	
Umkehrschalter	Digitaleingang 1I1	Digitaleingang 2I1
	Achten Sie bei der Parametrierung darauf, dass der Start der Referenzpunktfahrt in Richtung Umkehrschalter parametriert ist. Nur so können Sie sicherstellen, dass der Referenzpunktschalter auch immer gefunden wird.	
Freigabeeingang	Digitaleingang 1I2	Digitaleingang 2I2

Ablauf der Betriebsart "Referenzpunktfahrt"

Die Taster für die positive und negative Verfahrrichtung müssen je Kanal installiert sein.

1. Tragen Sie den Wert der Referenzpunktkoordinate in den Parameter-DB ein (REFPT).
2. Tragen Sie die Art der "Referenzpunktfahrt" in den Parameter-DB ein.
   Hierbei haben Sie folgende Möglichkeiten:

Start in Richtung ...	Zur Synchronisation führt ...	
plus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **plus**	REFPT_TYPE = 0
plus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **minus**	REFPT_TYPE = 1
minus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **plus**	REFPT_TYPE = 2
minus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **minus**	REFPT_TYPE = 3

3. Tragen Sie die Startgeschwindigkeit ein:
   - Eilgang (REFPT_SPD = 0)
   - Schleichgang (REFPT_SPD = 1)
4. Schreiben und aktivieren Sie die Maschinendaten.
5. Setzen Sie das Steuersignal für die Betriebsart "Referenzpunktfahrt" (MODE_IN = 3).
6. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
7. Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten"(EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.
8. Setzen Sie das Steuersignal für die Fahrtrichtung plus bzw. minus oder das Startsignal (DIR_P = 1, DIR_M = 1 oder START = 1)
9. Rufen Sie die Anweisung ABS_CTRL auf.

Die folgende Tabelle zeigt die Startkommandos für eine Referenzpunktfahrt.

Startbefehl	Aufgabe	Bemerkung
DIR_P	Der Antrieb startet in Richtung positiver Werte. Er bewegt sich also in Richtung Verfahrbereich Ende.	Wenn in den Maschinendaten eine negative Richtung eingetragen ist, meldet die FM 351 einen Bedienfehler. Es wird keine Referenzpunktfahrt durchgeführt.
DIR_M	Der Antrieb startet in Richtung negativer Werte. Er bewegt sich also in Richtung Verfahrbereich Anfang.	Wenn in den Maschinendaten eine positive Richtung eingetragen ist, meldet die FM 351 einen Bedienfehler. Es wird keine Referenzpunktfahrt durchgeführt.
START	Der Antrieb startet in die Richtung, die in den Maschinendaten eingegeben wurde.	

Hinweis

Für die Rundachse gilt: Die Reproduzierbarkeit des Referenzpunktes ist nur dann sichergestellt, wenn zwischen dem Wert **Ende der Rundachse** und dem Wert **Weg pro Geberumdrehung** ein ganzzahliges Verhältnis besteht.

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Referenzpunktfahrt".

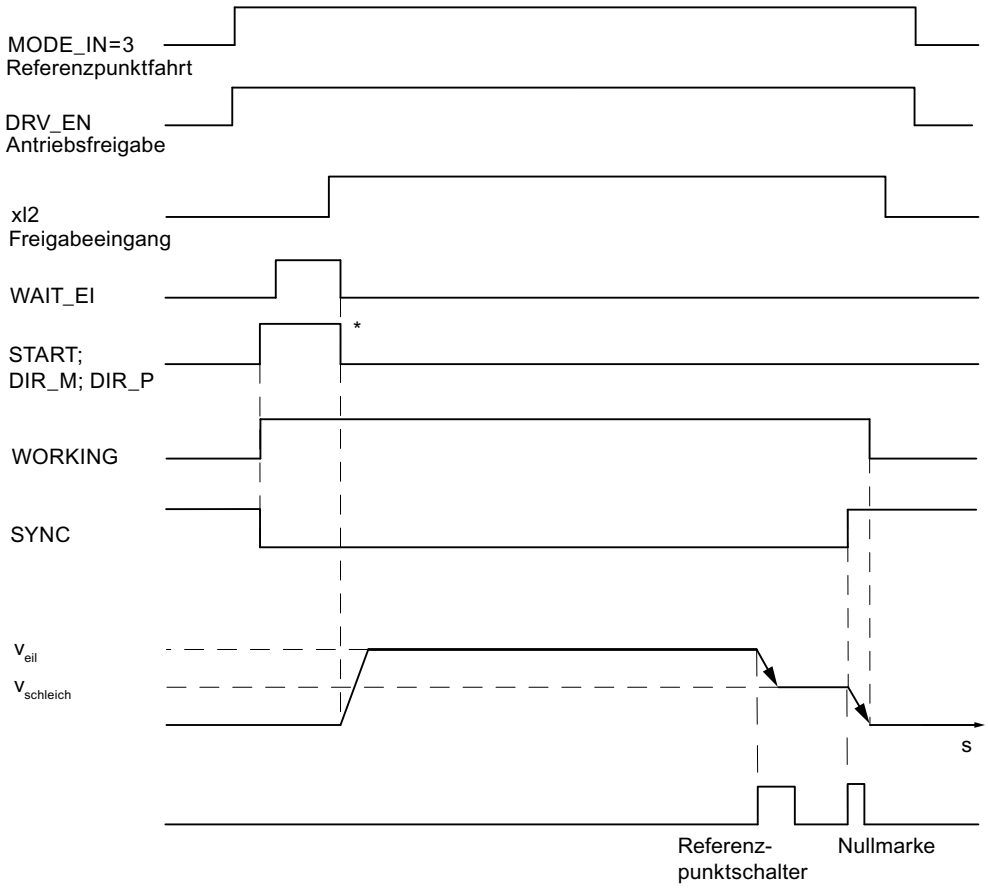

* Die Startsignale werden von der Anweisung ABS_CTRL zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positioniervorgang starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	3 = Referenzpunktfahrt
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
52.0	REFPT_TYPE	DINT	L#0	Art der Referenzpunktfahrt
99.0	REFPT_SPD	BOOL	TRUE	Startgeschwindigkeit für Referenzpunktfahrt 0 = Eilgang 1 = Schleichgang

Auswirkungen der Betriebsart

- Mit Start der Fahrt wird die Synchronisation weggenommen.
- Die Istposition wird auf den Wert der Referenzpunktkoordinate gesetzt, wenn das Rückmeldesignal SYNC gesetzt wird.
- Der Arbeitsbereich wird auf der Achse festgelegt.
- Die einzelnen Punkte innerhalb des Arbeitsbereichs behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.

Abbrechen der Referenzpunktfahrt

Die Betriebsart "Referenzpunktfahrt" wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0),
- eine Verfahrbereichsgrenze bei einer Linearachse überfahren wurde.

Referenzpunktfahrt in Abhängigkeit der Startposition

Bei einer Referenzpunktfahrt müssen Sie verschiedene Fälle unterscheiden, die abhängig sind

- von der Lage des Antriebs beim Start einer Referenzpunktfahrt,
- von der parametrierten Startrichtung,
- von der parametrierten Lage der Nullmarke zum Referenzpunktschalter.

Die folgende Tabelle zeigt die Fälle für REFPT_TYPE = 0 und REFPT_TYPE = 1. Für REFPT_TYPE = 2 und REFPT_TYPE = 3 gelten die Bilder entsprechend.

Bedingungen der Referenzpunktfahrt	Ablauf der Referenzpunktfahrt
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert.	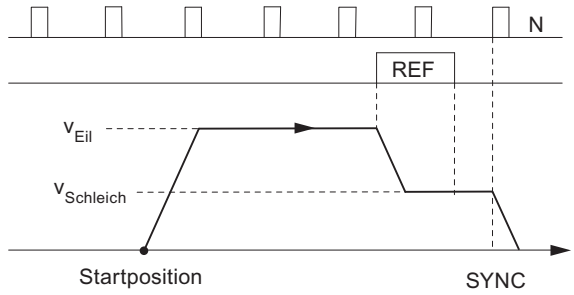
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 1): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung minus parametriert.	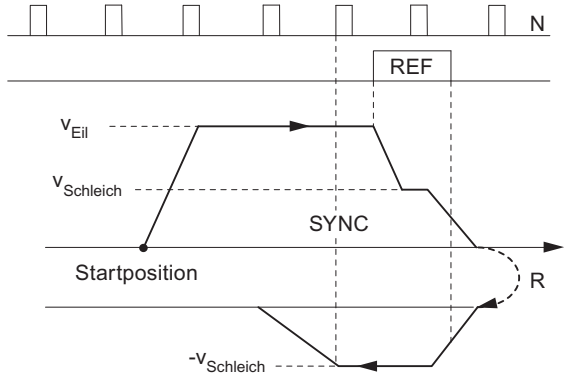
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung muss in Richtung plus parametriert sein. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Der Umkehrschalter ist positiver als der Referenzpunktschalter.	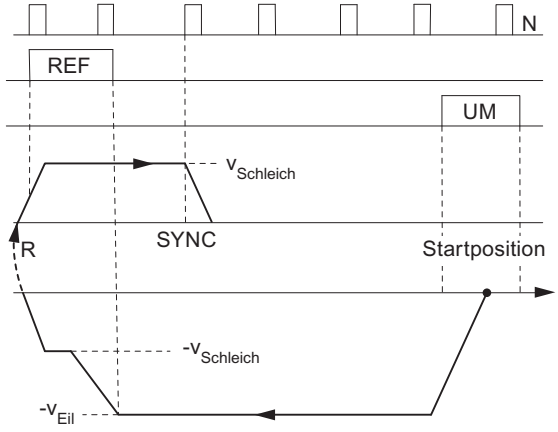

Bedingungen der Referenzpunktfahrt	Ablauf der Referenzpunktfahrt
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 1): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung minus parametriert. • Startposition der Referenzpunktfahrt ist auf dem Referenzpunktschalter.	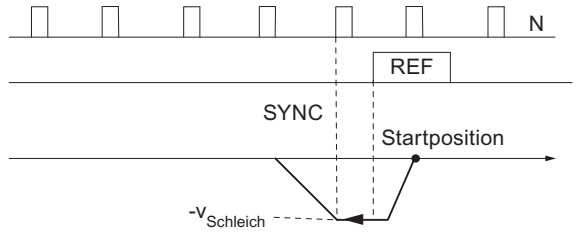
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Der Umkehrschalter ist positiver als der Referenzpunktschalter.	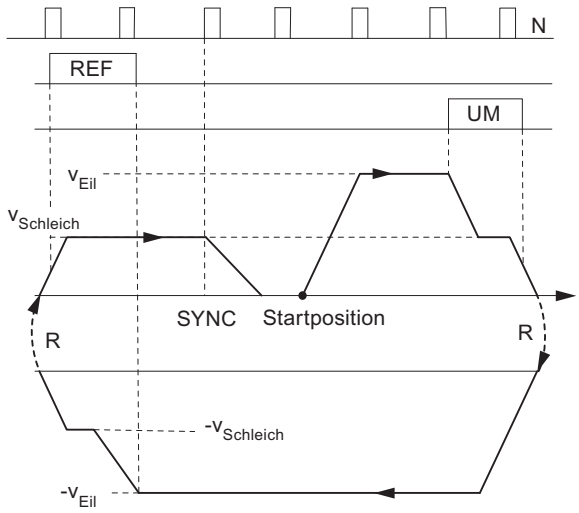
Beispiel für Referenzpunktfahrt (REFPT_TYPE=0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Startgeschwindigkeit = Schleichgeschwindigkeit	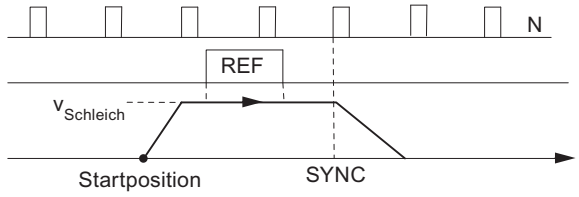
R = Richtungsumkehr REF = Referenzpunktschalter UM = Umkehrschalter N = Nullmarke des Gebers SYNC = Synchronisation wurde erreicht	

Betriebsart "Schrittmaßfahrt"

Definition

In der Betriebsart "Schrittmaßfahrt" kann die FM 351 den Antrieb

- auf **absolute** Ziele bewegen,
- **relativ** um ein Wegstück in eine vorgegebene Richtung bewegen.

Das Ziel bzw. die relativen Wege geben Sie als Schrittmaße der FM 351 vor. Sie haben die Möglichkeit, maximal 100 Schrittmaße in eine Tabelle einzutragen, die sowohl für die Betriebsart "Schrittmaßfahrt relativ" als auch für "Schrittmaßfahrt absolut" gültig sind. Unabhängig von der Schrittmaßtabelle können Sie mit dem Schrittmaß 254 (Seite 2542) und Schrittmaß 255 (Seite 2542) den Weg vorgeben.

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.
- Die Schrittmaße müssen auf der Baugruppe vorhanden sein.

Interpretation der Schrittmaße

Je nachdem welche Schrittmaßfahrt Sie wählen, interpretiert die FM 351 die Vorgaben unterschiedlich.

- Schrittmaßfahrt absolut: Die Schrittmaße werden als absolute Zielposition interpretiert.
- Schrittmaßfahrt relativ: Die Schrittmaße werden als Wegdifferenz von der Startposition interpretiert.

Hinweis

Für die Betriebsart "Schrittmaßfahrt relativ" sind nur positive Schrittmaße zulässig.

Die Schrittmaße erhalten Ihr Vorzeichen durch die Richtungsvorgaben DIR_P bzw. DIR_M.

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 1 bis 100

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 1 bis 100	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Tragen Sie die Schrittmaße in die Tabellen (TRGL1; TRGL2) ein.	
3	Schreiben Sie die Schrittmaßtabellen (TRGL1/2WR_EN = 1).	
4	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 1 bis 100	
5	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
6	Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 1 … 100).	
7	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
8	Rufen Sie die Anweisung ABS_CTRL auf.	
Schritte 2 und 3 sind nur erforderlich, wenn noch keine Schrittmaße vorhanden sind oder die vorhandenen Schrittmaße geändert werden sollen.		

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 254

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 254	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
3	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
4	Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 254).	
5	Tragen Sie das Schrittmaß für die Schrittmaßnummer 254 ein (TRG252_254).	
6	Setzen Sie das Anstoßbit zum Schreiben des Schrittmaßes (TRG252_254_EN = 1).	
7	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
8	Rufen Sie die Anweisung ABS_CTRL auf.	

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 255

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 255	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
3	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
4	Tragen sie die Schrittmaßnummer ein (MODE_TYPE = 255).	
5	Tragen Sie das Schrittmaß für die Schrittmaßnummer 255 ein (TRG255).	
6	Tragen Sie den Wert für die Umschaltdifferenz der Schrittmaßnummer 255 ein (CHGDIF255).	
7	Tragen Sie den Wert für die Abschaltdifferenz der Schrittmaßnummer 255 ein (CUTDIF255).	
8	Setzen Sie das Anstoßbit zum Schreiben von Schrittmaß, Ab- und Umschaltdifferenz (TRG255_EN = 1).	
9	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
10	Rufen Sie die Anweisung ABS_CTRL auf.	

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Schrittmaßfahrt".

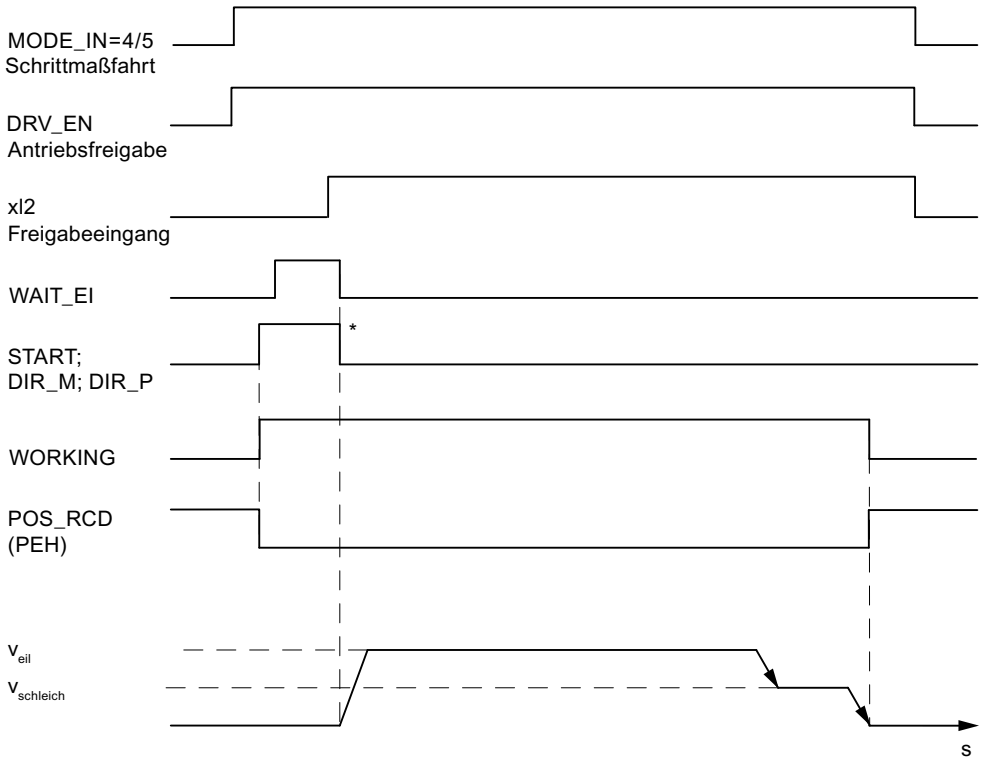

* Die Startsignale werden von der Anweisung ABS_CTRL zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positionierung starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	4 = Schrittmaßfahrt relativ 5 = Schrittmaßfahrt absolut
17.0	MODE_TYPE	BYTE	B#16#0	Schrittmaßnummer 1 … 100, 254 oder 255
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positionierung läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 254 schreiben
36.3	TRG255_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 255 schreiben
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaßnummer 1 … 50) schreiben

Adresse	Name	Typ	Anfangswert	Kommentar
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaßnummer 51 … 100) schreiben
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 254
100.0	TRG255	DINT	L#0	Schrittmaß für Schrittmaßnummer 255
104.0	CHGDIF_255	DINT	L#0	Umschaltdifferenz für Schrittmaßnummer 255
108.0	CUTDIF_255	DINT	L#0	Abschaltdifferenz für Schrittmaßnummer 255

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus	
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus	
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus	
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus	
120.0	TRGL1.TRG[1]	DINT	L#0	Schrittmaßnummer 1	Schrittmaßtabelle 1
. . .	. . .	. . .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Schrittmaßnummer 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Schrittmaßnummer 51	Schrittmaßtabelle 2
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Schrittmaßnummer 100	

Restweg

Der Restweg ist die Differenz zwischen Ziel (Schrittmaß) und Istwert. Der Restweg ist

- positiv, wenn das Ziel noch nicht erreicht wurde.
- negativ, wenn das Ziel bereits überfahren wurde.

Bei einer Rundachse ist der angezeigte Restweg nicht verwendbar.

Absteuern der Schrittmaßfahrt

Die Betriebsart "Schrittmaßfahrt" wird abgesteuert, wenn die FM 351 ein STOP-Signal empfängt (STOP = 1).

Nach dem Absteuern der Fahrt bleibt ein Restweg bestehen.

Der verbleibende Restweg bei einer "Schrittmaßfahrt relativ" kann zu Ende verfahren werden, wenn

- die Betriebsart unverändert ist, und
- die Schrittmaßnummer unverändert ist, und

- die Richtung unverändert ist, und
- der verbleibende Restweg größer als die parametrierte Abschaltdifferenz ist.

Den Restweg verfahren Sie, indem Sie die "Schrittmaßfahrt relativ" unverändert noch einmal starten.

Abbrechen der Schrittmaßfahrt

Die Betriebsart "Schrittmaßfahrt" wird abgebrochen, wenn das Signal "Antriebsfreigabe" gelöscht wird (DRV_EN = 0).

Restweg löschen

Mit dem Auftrag "Restweg löschen" löschen Sie einen anstehenden Restweg.

Mit dem Aufruf einer anderen Betriebsart oder dem Start der Betriebsart in die andere Richtung löschen Sie ebenfalls den anstehenden Restweg.

Verwendetes Datum im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.2	DELDIST_EN	BOOL	FALSE	1 = Restweg löschen

Auftrag "Istwert setzen" / "Istwert setzen rückgängig"

Definition

Mit dem Auftrag "Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Der Arbeitsbereich wird auf einen anderen Bereich der Achse projiziert.

Die Verschiebung des Arbeitsbereiches ermitteln Sie mit (IST_{neu} - $IST_{aktuell}$).

- IST_{neu} ist der Vorgabewert
- $IST_{aktuell}$ ist der Istwert zum Zeitpunkt der Ausführung

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.

Ablauf des Auftrags

1. Tragen Sie die Koordinate AVAL für den Istwert (IST_{neu}) ein.
   - Linearachse:
     Den angegebenen Istwert müssen Sie so wählen, dass die Software-Endschalter nach dem Aufruf des Auftrags noch innerhalb des zulässigen Verfahrbereichs liegen.
     Der Betrag der Verschiebung, der sich aus (IST_{neu} - $IST_{aktuell}$) ergibt, muss kleiner oder gleich dem Betrag des zulässigen Verfahrbereichs sein (maximal 100 m bzw. 1000 m).
   - Rundachse:
     Für den angegebenen Istwert muss gelten:
     0 ≤ Istwert < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit (AVAL_EN = 1).

Wenn der Auftrag "Istwert setzen" während einer Positionierung abgegeben wird, dann wird er bis zum Ende der Positionierung zurückgehalten und erst bei dem danach folgenden Aufruf des Bausteins durchgeführt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.7	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
84.0	AVAL	DINT	L#0	Koordinate für "Istwert setzen"

Auswirkungen des Auftrags

Am Beispiel "Istwert setzen" auf 300 mm erkennen Sie, wie dieser Auftrag den Arbeitsbereich auf eine bestimmte Position der Achse projiziert. Es ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Istwertkoordinate gesetzt.
- Der Arbeitsbereich wird auf der Achse verschoben.
- Die einzelnen Punkte (z. B. Software-Endschalter Ende) innerhalb des Arbeitsbereichs behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Istwert setzen".

Istwert setzen	SEA	IST	SEE
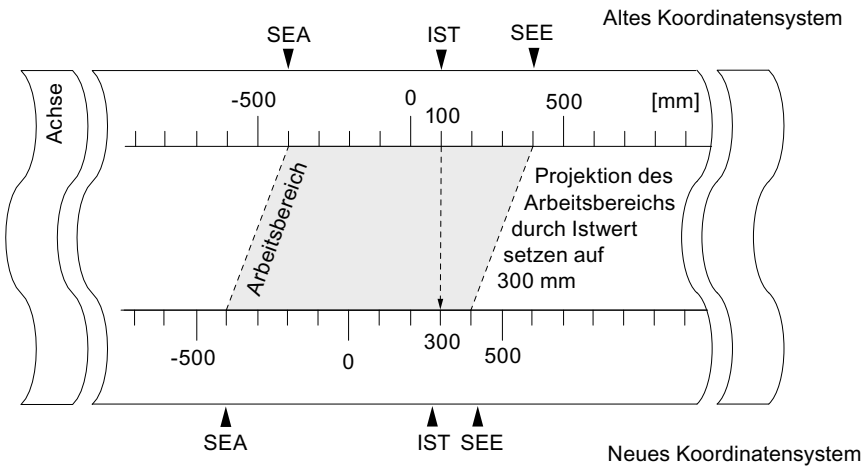	-400	100	400
	-400	300	400

Rücknahme des Auftrags ("Istwert setzen rückgängig")

Mit dem Auftrag "Istwert setzen rückgängig" setzen Sie alle Arbeitsbereichverschiebungen, die durch "Istwert setzen" erzeugt wurden, wieder zurück.

Die Summe aller Arbeitsbereichverschiebungen darf nicht größer als der Verfahrbereich sein, damit dieser Auftrag richtig ausgeführt werden kann.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.3	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen rückgängig

Auftrag "Bezugspunkt setzen"

Definition

Mit dem Auftrag "Bezugspunkt setzen" synchronisieren Sie die Achse. Der Auftrag verschiebt den Arbeitsbereich. Alle Verschiebungen, die durch "Istwert setzen" erzeugt wurden, bleiben erhalten.

Die Einstellung projiziert den Arbeitsbereich auf die Achse. Durch unterschiedliche Werteingaben kann deshalb der Arbeitsbereich an beliebigen Positionen im physikalischen Bereich der Achse liegen.

Voraussetzungen

- Die Positionierung muss beendet sein.
- Die Achse muss parametriert sein.

Ablauf des Auftrags

1. Tragen Sie den Wert REFPT für die Bezugspunktkoordinate ein.
   – Linearachse:

   Die Bezugspunktkoordinate darf nicht außerhalb der Software-Endschalter liegen. Das gilt auch für die Bezugspunktkoordinate in einem verschobenen Koordinatensystem.

   – Rundachse:
   Für die Bezugspunktkoordinate muss gelten:
   0 ≤ Bezugspunktkoordinate < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit (REFPT_EN = 1).

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
35.6	REFPT_EN	BOOL	FALSE	1 = Bezugspunkt setzen
92.0	REFPT	DINT	L#0	Bezugspunktkoordinate

Auswirkungen des Auftrags

Am Beispiel "Bezugspunkt setzen" auf 400 mm erkennen Sie, wie dieser Auftrag den Arbeitsbereich auf eine bestimmte physikalische Position der Achse projiziert. Es ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Bezugspunktkoordinate gesetzt.
- Der Arbeitsbereich wird auf der Achse verschoben.
- Die einzelnen Punkte (z. B. Software-Endschalter Ende) behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.
- Das Bit SYNC in den Rückmeldesignalen wird gesetzt.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Bezugspunkt setzen".

Bezugspunkt setzen	SEA	IST	SEE
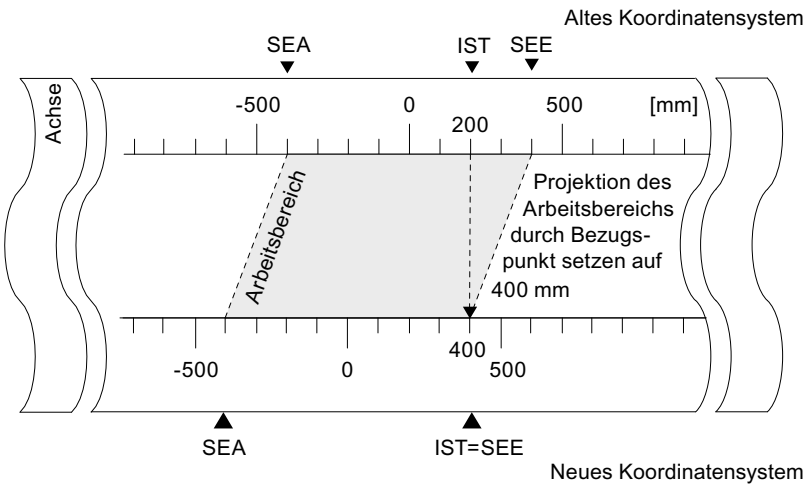	-400	200	400
	-400	400	400

Besonderheiten Absolutwertgeber

Dieser Auftrag ist für eine Absolutwertgeberjustage (Seite 2529) notwendig.

Funktionsschalter "Schleifenfahrt"

Definition

Mit "Schleifenfahrt" legen Sie die Richtung fest, in die ein Ziel mit Kraftschluss angefahren wird. Die Schleifenfahrt können Sie verwenden, wenn nur in einer Richtung Kraftschluss zwischen Motor und Achse sichergestellt werden kann.

Ein Ziel, das entgegen der vorgegebenen Richtung angefahren wird, wird zunächst überfahren. Anschließend nimmt die Baugruppe eine Richtungsumkehr vor und fährt das Ziel dann in vorgegebener Richtung an.

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.
- Bei einer Schleifenfahrt entgegen der Verfahrrichtung zum Ziel liegt die maximale Zielposition:
  - in Verfahrrichtung plus
    Ziel < SEE - ½ Zielbereich - Abschaltdifferenz plus - Umschaltdifferenz minus
  - in Verfahrrichtung minus
    Ziel > SEA + ½ Zielbereich + Abschaltdifferenz minus + Umschaltdifferenz plus

- Eine Schleifenfahrt wird nicht durchgeführt, wenn das Ziel in Richtung der Schleifenfahrt angefahren wird. In diesem Fall wird eine Schrittmaßfahrt ohne Richtungsumkehr durchgeführt.
- Der Ablauf der Betriebsart "Schrittmaßfahrt" (Seite 2498) muss bekannt sein.

Ablauf der Schleifenfahrt

1. Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut / relativ" (MODE_IN = 5/4).
2. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
3. Setzen Sie den Funktionsschalter "Freigabeeingang" (EI_OFF = 1) oder verdrahten Sie für den entsprechenden Kanal den Freigabeeingang.
4. Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 1 … 100, 254, 255).
5. Setzen Sie den Funktionsschalter "Schleifenfahrt" (PLOOP_ON / MLOOP_ON = 1).
6. Starten Sie die Schrittmaßfahrt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positionierung starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	4 / 5 = Schrittmaßfahrt relativ / absolut
17.0	MODE_TYPE	BYTE	B#16#0	Schrittmaßnummer 1 … 100, 254 oder 255
34.0	PLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung plus
34.1	MLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung minus
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Fiktives Ziel

Wenn Sie eine Positionierung auf ein Ziel starten, das entgegen der Richtung der parametrierten Schleifenfahrt liegt, ermittelt die Baugruppe zu diesem Ziel ein fiktives Ziel, an dem sie eine Richtungsumkehr ausführt und dann in richtiger Richtung das Ziel anfährt.

Dieses fiktive Ziel **muss** mindestens um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegen.

Die Entfernung des fiktiven Ziels vom parametrierten Ziel wird richtungsabhängig ermittelt. Die folgende Tabelle zeigt die Berechnung der Lage des fiktiven Ziels bei Schleifenfahrt.

Vorgaben	Lage des fiktiven Ziels
Parametrierung: Schleife + (Kraftschluss plus) und Fahrt in Richtung minus	Das fiktive Ziel (**Ziel**$_f$) hat den Wert: **Ziel**$_f$ **= Ziel - Abschaltdifferenz minus - Umschaltdifferenz plus** 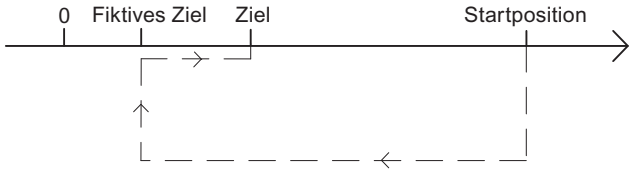
Parametrierung: Schleife - (Kraftschluss minus) und Fahrt in Richtung plus	Das fiktive Ziel (**Ziel**$_f$) hat den Wert: **Ziel**$_f$**= Ziel + Abschaltdifferenz plus + Umschaltdifferenz minus** 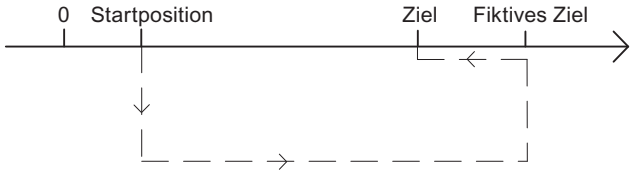

Beispiel

Anhand einer Positionierung mit Schleifenfahrt minus auf ein maximales Ziel zeigen wir Ihnen die Lage des fiktiven Ziels.

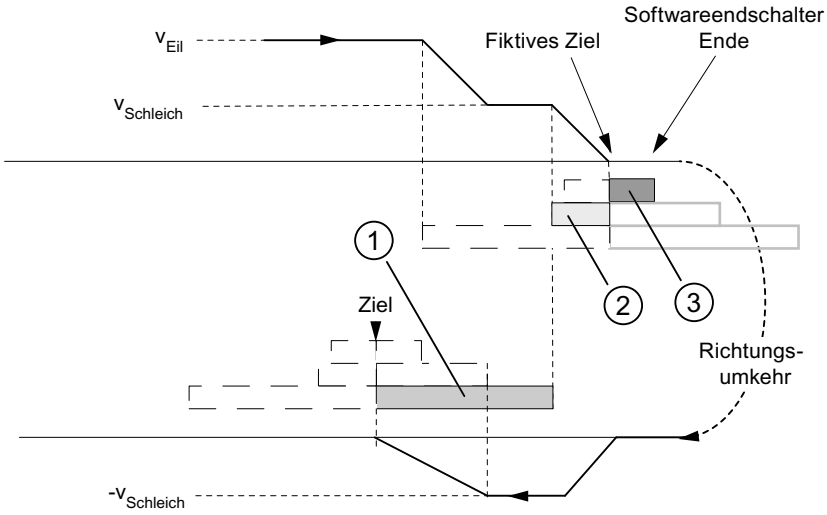

① Umschaltdifferenz negativ
② Abschaltdifferenz positiv
③ ½ Zielbereich

Funktionsschalter "Freigabeeingang"

Definition

Der Freigabeeingang ist ein externer Eingang, mit dem eine Positionierung durch ein externes Ereignis freigegeben werden kann.

Freigabeeingang auswerten (EI_OFF = 0)

Der entsprechende Freigabeeingang (xI2) für den Kanal muss verdrahtet sein.

Sie haben damit die Möglichkeit, den Start einer Positionierung vorzubereiten. Sie starten die Positionierung unabhängig vom Programmablauf Ihres Anwenderprogramms, indem Sie an den Freigabeeingang ein "1"-Signal anlegen.

Die Fahrt beginnt, wenn Sie an den Freigabeeingang ein "1"-Signal anlegen und wird abgesteuert, wenn Sie an den Freigabeeingang ein "0"-Signal anlegen.

Freigabeeingang nicht auswerten (EI_OFF = 1)

Wenn Sie die Auswertung des Freigabeeingangs abschalten, startet eine Betriebsart unmittelbar nach dem Erkennen des Startsignals. Es ist dann nicht möglich, eine Betriebsart vorzubereiten und zu einem definierten späteren Zeitpunkt zu starten.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Auftrag "Positionsdaten lesen"

Definition

Mit dem Auftrag "Positionsdaten lesen" können Sie Schrittmaß, Restweg und Geschwindigkeit zum aktuellen Zeitpunkt lesen.

Ablauf des Auftrags

1. Setzen Sie das Anstoßbit im Kanal-DB (ACTSPD_EN = 1).
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.1	ACTSPD_EN	BOOL	FALSE	1 = Positionsdaten lesen
112.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
116.0	DIST_TO_GO	DINT	L#0	Restweg
120.0	ACT_TRG	DINT	L#0	Aktuelles Schrittmaß

Auftrag "Geberdaten lesen"

Definition

Mit dem Auftrag "Geberdaten lesen" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.

Voraussetzungen

Sie können den Wert für die Absolutwertgeberjustage (Seite 2529) auslesen, nachdem Sie den Auftrag "Bezugspunkt setzen" (Seite 2505) ausgeführt haben.

Ablauf des Auftrags

1. Setzen Sie das Anstoßbit im Kanal-DB (ENCVAL_EN = 1).
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.2	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
124.0	ENCVAL	DINT	L#0	Geberistwert (interne Darstellung)
128.0	ZEROVAL	DINT	L#0	Letzter Nullmarkenwert (interne Darstellung)
132.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage

Rückmeldesignale für die Positionierung abfragen

Definition

Mit den "Rückmeldesignalen für die Positionierung" werden Sie über den aktuellen Zustand der Positionierung informiert.

Ablauf

1. Die Daten werden bei jedem Aufruf der Anweisung ABS_CTRL im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positionierung läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
23.4	SPEED_OUT	BOOL	FALSE	0 = Schleichgang 1 = Eilgang
23.5	ZSPEED	BOOL	FALSE	1 = Achse befindet sich im Stillstandsbereich
23.6	CUTOFF	BOOL	FALSE	1 = Achse befindet sich im Abschaltbereich
23.7	CHGOVER	BOOL	FALSE	1 = Achse befindet sich im Umschaltbereich
24.0	MODE_OUT	BYTE	B#16#0	Aktive Betriebsart
25.2	GO_M	BOOL	FALSE	1 = Achse fährt in Richtung minus
25.3	GO_P	BOOL	FALSE	1 = Achse fährt in Richtung plus
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht
26.0	ACT_POS	DINT	L#0	Aktueller Istwert (aktuelle Position der Achse)

Rückmeldesignale für die Diagnose abfragen

Definition

Mit den "Rückmeldesignalen für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Ablauf

1. Wenn die Baugruppe ein neues Ereignis in den Diagnosepuffer einträgt, setzt sie in der Rückmeldeschnittstelle das Bit DIAG in allen Kanälen. Jedes Auftreten eines Fehlers (Seite 2555) bewirkt einen Eintrag in den Diagnosepuffer. Beim Löschen des Diagnosepuffers wird das Bit DIAG auch gesetzt.
2. Wenn der Aufruf einer Betriebsart bzw. das Steuern einer aktiven Betriebsart nicht möglich ist oder fehlerhaft durchgeführt wurde, setzt die Baugruppe in der Rückmeldeschnittstelle einen Bedienfehler OT_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen. Solange der Bedienfehler ansteht, können Sie weder eine neue Betriebsart starten noch die angehaltene Betriebsart fortsetzen. Einen anstehenden Bedienfehler quittieren Sie mit OT_ERR_A = 1.
3. Wenn die Baugruppe einen Schreibauftrag mit fehlerhaften Daten erkennt, setzt sie in der Rückmeldeschnittstelle das Bit DATA_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen.
4. Die Rückmeldesignale werden im Kanal-DB abgelegt.
5. Wenn der Diagnosepuffer gelesen ist, setzt die Baugruppe das Bit DIAG in allen Kanälen wieder auf 0.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.3	OT_ERR	BOOL	FALSE	1 = Bedienfehler
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler

6.2.5.3 FM 351 konfigurieren

6.2.5.4 FM 351 parametrieren

Kanäle spezifizieren

Maßsystem auswählen

Maßsystem auswählen

Sie können für die Eingabe und Ausgabe der Daten unter den folgenden Maßsystemen wählen:

- Millimeter (Voreinstellung)
- Inch
- Grad

Hinweis

Wenn Sie das Maßsystem im Dialog "Kanäle spezifizieren" verändern, dann werden die Werte in das neue System umgerechnet. Dabei können Rundungsfehler auftreten.

Wenn Sie das Maßsystem über das Maschinendatum ändern, werden die Werte nicht automatisch umgerechnet.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
8.0	UNITS	DINT	L#1	**Maßsystem** Zulässige Werte: • 1 = 10^{-3} mm • 2 = 10^{-4} inch • 3 = 10^{-4} grad • 4 = 10^{-2} grad • 6 = 10^{-3} grad

Standard-Maßsystem in der Online-Hilfe

Wir verwenden für die Angabe der minimalen und maximalen Werte das Maßsystem mm. Für die Bestimmung der Grenzwerte in den anderen Maßsystemen nehmen Sie deshalb folgende Umrechnung vor:

Für die Umrechnung von		berechnen Sie
mm → inch		Grenzwert (inch) = Grenzwert (mm) × 0,1 [1]
mm → grad	10^{-4} (4 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 0,1
	10^{-3} (3 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 1
	10^{-2} (2 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 10
inch → grad		Grenzwert (grad) = Grenzwert (inch)

[1] Die Anzahl der Nachkommastellen hat beim Maximalwert Auswirkungen auf die Anzahl der Stellen vor dem Komma. Im Maßsystem Inch werden 4 Nachkommastellen verwendet, somit können maximal 100 000,0000 inch eingegeben werden. Das Maßsystem Millimeter verwendet 3 Nachkommastellen, hier können maximal 1 000 000,000 mm eingegeben werden.

Zusammenhang zwischen Inkrementen und Maßsystem

Die Gebersignale eines angeschlossenen Gebers werden von der Baugruppe ausgewertet und in das aktuelle Maßsystem umgerechnet. Für die Umrechnung wird die Auflösung (Seite 2535) verwendet.

Wenn die Positionierbaugruppe

- 10 Inkremente gezählt hat und
- eine Auflösung von 100 µm pro Inkrement durch die parametrierten Geberdaten eingestellt ist,

bedeutet das, dass die Achse um eine Strecke von 1 mm bewegt wurde.

Kanäle kopieren

Kanäle kopieren

Sie können die Parameter eines Kanals auf einen anderen Kanal kopieren.

1. Wählen Sie den Kanal aus, dessen Parameter kopiert werden sollen (Kanal Quelle).
2. Aktivieren Sie den Kanal, auf den die Parameter kopiert werden sollen (Kanal Ziel).
3. Drücken Sie die Schaltfläche "Kopieren".

Kanal n Antrieb

Ansteuerart auswählen

Übersicht der Ansteuerarten

Übersicht

Die Ansteuerart legt fest, wie ein Motor an den 4 Digitalausgängen (Q) eines Kanals über das Leistungsteil betrieben wird. Für die Positionierbaugruppe gibt es 4 Ansteuerarten je Kanal:

- Ansteuerart 1 (Seite 2516)
- Ansteuerart 2 (Seite 2516)
- Ansteuerart 3 (Seite 2517)
- Ansteuerart 4 (Seite 2518)

Die folgende Tabelle zeigt die Zustände der 4 Digitalausgänge für jede Ansteuerart, x steht für Kanal 1 und 2.

	Eilgang		Schleichgang		PEH Position erreicht Halt
	Richtung +	Richtung -	Richtung +	Richtung -	
Ansteuerart 1					
xQ0	1	1	0	0	-
xQ1	0	0	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Ansteuerart 2					
xQ0	1	1	0	0	0
xQ1	0	0	0	0	1
xQ2	1	0	1	0	0
xQ3	0	1	0	1	0
Ansteuerart 3					
xQ0	1	1	0	0	-
xQ1	1	1	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Ansteuerart 4					
xQ0	1	0	0	0	-
xQ1	1	0	1	0	-
xQ2	0	1	0	0	-
xQ3	0	1	0	1	-

Ansteuerart 1

Ansteuerart 1

Bei Ansteuerart 1 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang
- xQ1: Schleichgang
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 1.

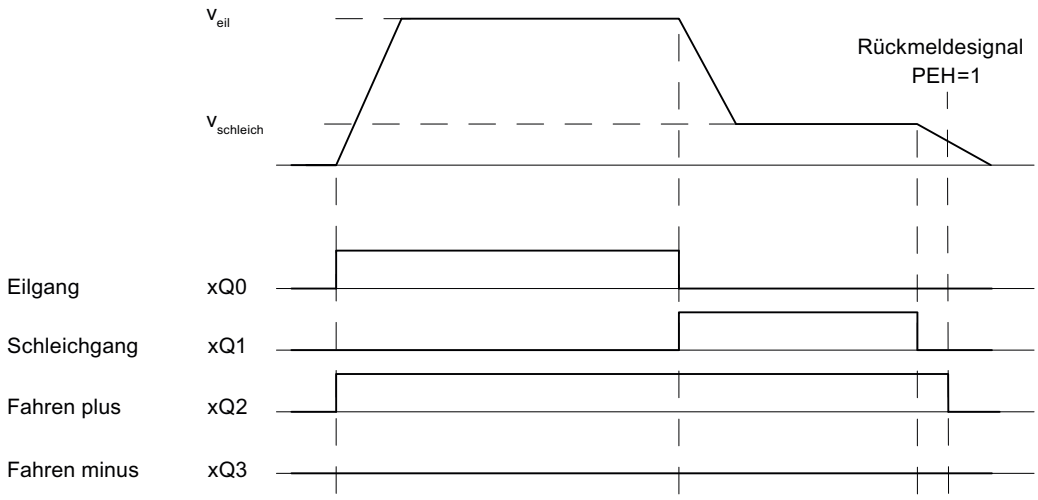

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	1 = Ansteuerart 1

Ansteuerart 2

Ansteuerart 2

Bei Ansteuerart 2 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eil-/Schleichgang
- xQ1: Position erreicht
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 2.

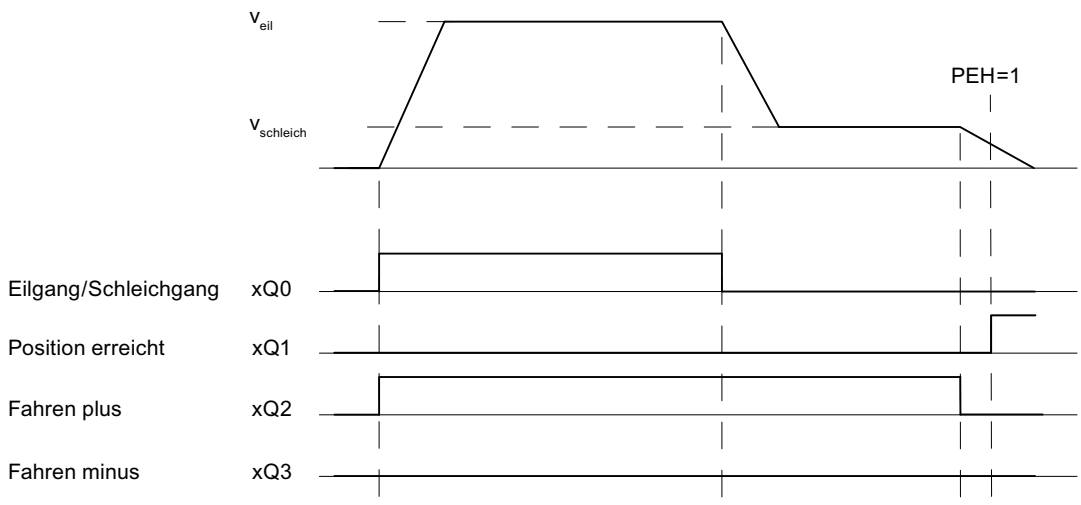

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	2 = Ansteuerart 2

Ansteuerart 3

Ansteuerart 3

Bei Ansteuerart 3 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang
- xQ1: Schleichgang
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 3.

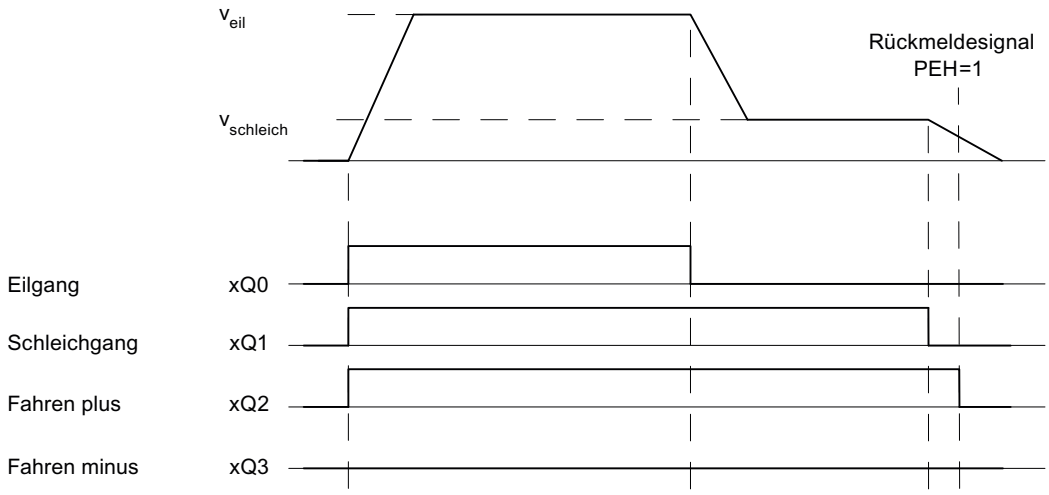

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	3 = Ansteuerart 3

Ansteuerart 4

Ansteuerart 4

Bei Ansteuerart 4 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang plus
- xQ1: Schleichgang plus
- xQ2: Eilgang minus
- xQ3: Schleichgang minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 4.

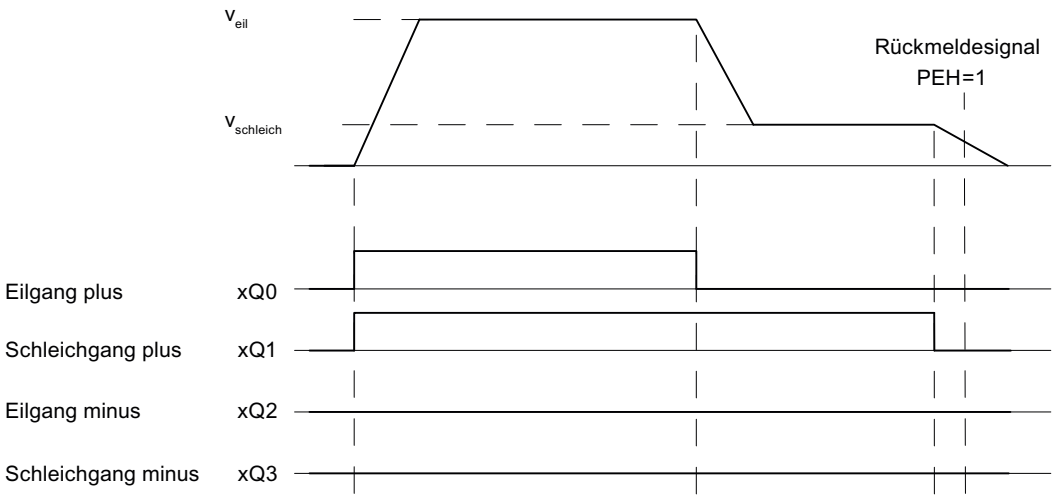

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	4 = Ansteuerart 4

Umschaltdifferenz festlegen

Umschaltdifferenz

Die "Umschaltdifferenz" definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang umgeschaltet wird.

Die vorgegebenen Werte gelten für alle Ziele, die die Positionierbaugruppe anfährt. Ausnahme ist eine Schrittmaßfahrt mit dem Schrittmaß 255, dafür wird eine gesonderte Umschaltdifferenz festgelegt.

Regeln für die Umschaltdifferenz

- Die Umschaltdifferenz plus und die Umschaltdifferenz minus müssen nicht symmetrisch um das Ziel angeordnet sein. Die Werte können sich unterscheiden.
- Die Umschaltdifferenz muss größer als die Abschaltdifferenz sein.
- Die Umschaltdifferenz muss innerhalb des Arbeitsbereichs liegen.
- Die Umschaltdifferenz muss kleiner als das Ende der Rundachse sein.
- Der Abstand zwischen dem Umschaltpunkt und dem Abschaltpunkt muss so groß gewählt werden, dass der Antrieb sicher vom Eilgang in den Schleichgang schalten kann.
- Der Umschaltpunkt und der Abschaltpunkt müssen einen zeitlichen Abstand von mindestens 8 ms voneinander haben.

Das folgende Bild zeigt die Lage der Schaltbereiche um das Ziel.

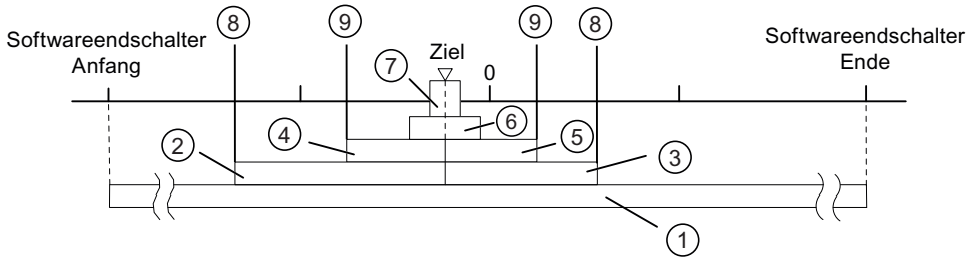

① Arbeitsbereich
② Umschaltdifferenz plus / minus
③
④ Abschaltdifferenz plus / minus
⑤
⑥ Stillstandsbereich
⑦ Zielbereich
⑧ Umschaltpunkt
⑨ Abschaltpunkt

Weitere Informationen über die Anordnung der Bereiche finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CHGDIF_P	DINT	L#5000	**Umschaltdifferenz plus**
104.0	CHGDIF_M	DINT	L#5000	**Umschaltdifferenz minus** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Abschaltdifferenz festlegen

Abschaltdifferenz

Die "Abschaltdifferenz" definiert den Abstand zum Ziel, an dem der Antrieb (Schleichgang) abgeschaltet wird. Die Überwachungsfunktionen übernimmt ab diesem Punkt die Positionierbaugruppe.

Die vorgegebenen Werte gelten für alle Ziele, die die Positionierbaugruppe anfährt. Ausnahme ist eine Schrittmaßfahrt mit dem Schrittmaß 255, dafür wird eine gesonderte Abschaltdifferenz festgelegt.

Regeln für die Abschaltdifferenz

- Die Abschaltdifferenz plus und die Abschaltdifferenz minus müssen nicht symmetrisch um das Ziel angeordnet sein. Die Werte können sich unterscheiden.
- Die Abschaltdifferenz muss größer als der halbe Zielbereich sein.

- Der Abstand zwischen Abschaltpunkt und Ziel muss so gewählt werden, dass der Antrieb innerhalb des Zielbereichs zum Stillstand kommt.
- Der Abschaltpunkt und der Beginn des Zielbereichs müssen einen zeitlichen Abstand von mindestens 8 ms voneinander haben.

Das folgende Bild zeigt die Lage der Schaltbereiche um das Ziel.

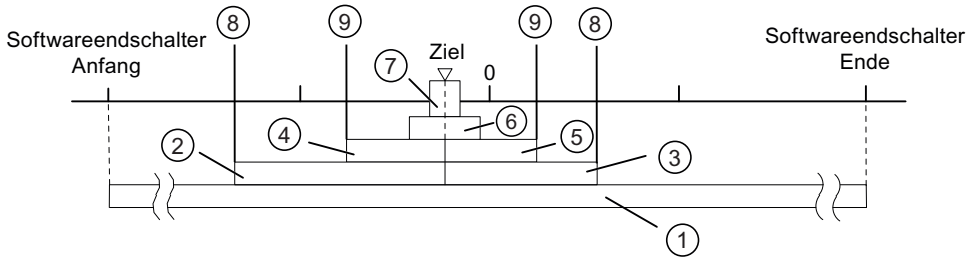

①	Arbeitsbereich	⑦	Zielbereich
②	Umschaltdifferenz plus / minus	⑧	Umschaltpunkt
③			
④	Abschaltdifferenz plus / minus	⑨	Abschaltpunkt
⑤			
⑥	Stillstandsbereich		

Weitere Informationen über die Anordnung der Bereiche finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
108.0	CUTDIF_P	DINT	L#2000	**Abschaltdifferenz plus**
112.0	CUTDIF_M	DINT	L#2000	**Abschaltdifferenz minus** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/ Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/ Impuls

Zieleinlauf festlegen

Zielbereich festlegen

Zielbereich

Die Positionierbaugruppe legt um jedes Ziel einen symmetrischen Bereich, den Zielbereich. Innerhalb dieses Bereichs muss bei einem Zieleinlauf der Istwert liegen.

Eine Wertvorgabe "0" schaltet die Toleranz beim Zieleinlauf ab. Die Geschwindigkeit bei Erreichen des Zieles muss dann kleiner sein als die parametrierte Stillstandsgeschwindigkeit (Seite 2524).

Regeln für den Zielbereich

- Der Zielbereich liegt symmetrisch um das Ziel.
- Eine Wertvorgabe "0" schaltet die Überwachung des Zielbereichs ab.

Das folgende Bild zeigt die Lage der Bereiche um das Ziel.

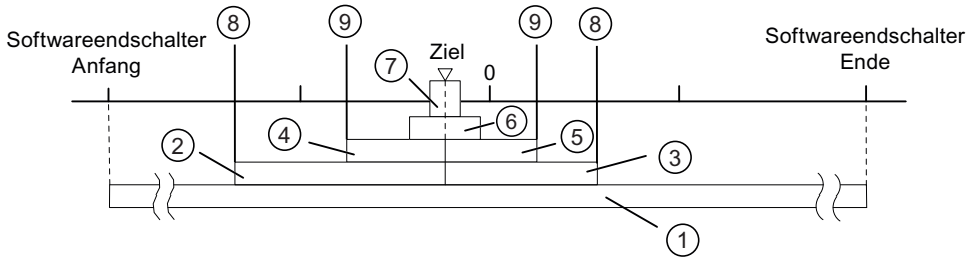

①	Arbeitsbereich	⑦	Zielbereich
② ③	Umschaltdifferenz plus / minus	⑧	Umschaltpunkt
④ ⑤	Abschaltdifferenz plus / minus	⑨	Abschaltpunkt
⑥	Stillstandsbereich		

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe" sowie unter "Ende eines Positioniervorgangs (Seite 2481)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	TRG_RANGE	DINT	L#1000	**Zielbereich** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/ Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/ Impuls

Überwachungszeit festlegen

Überwachungszeit

Während der Überwachungszeit überwacht die Baugruppe:

- die Bewegung der Achse bis zum Abschaltpunkt.
  Die Überwachungszeit startet mit dem Beginn einer Positionierung und wird bei jeder Istwertänderung in Fahrtrichtung nachgetriggert.
  Innerhalb der Überwachungszeit muss sich der Istwert mindestens um ein Inkrement (das Wegstück der Auflösung) verändert haben.
- den Zieleinlauf.
  Die Überwachungszeit wird mit dem Erreichen des Abschaltpunkts zum letzen Mal nachgetriggert.
  Innerhalb der Überwachungszeit muss der Zielbereich erreicht werden.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Ende eines Positioniervorgangs (Seite 2481)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
80.0	MON_TIME	DINT	L#2000	**Überwachungszeit** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 bis 100 000 ms
Tatsächliche Überwachungszeit Sie können für die Überwachungszeit alle Werte aus dem Wertebereich vorgeben. Die Baugruppe rundet die vorgegebene Zeit auf ein Vielfaches von 8 ms (Baugruppenzyklus) auf.				

Stillstandsbereich festlegen

Stillstandsbereich

Der Stillstandsbereich liegt symmetrisch um das jeweilige Ziel. In diesem Bereich wird überwacht, ob der Antrieb auf der angefahrenen Zielposition stehenbleibt oder von ihr wegdriftet. Die Überwachung beginnt, nachdem die Positionierbaugruppe "Position erreicht" (PEH) gemeldet hat.

Wenn der Stillstandsbereich ohne einen gültigen Fahrauftrag verlassen wird, meldet die Positionierbaugruppe einen Fehler.

Regeln für den Stillstandsbereich

- Der Stillstandsbereich liegt symmetrisch um das jeweilige Ziel:
  (Ziel – ½ × Stillstandsbereich) ≤ Ziel ≤ (Ziel + ½ × Stillstandsbereich)
- Der Stillstandsbereich muss größer oder gleich dem Zielbereich sein.

- Die Koordinatenwerte für den Stillstandsbereich müssen im Arbeitsbereich liegen.
- Mit dem Wert "0" ist die Stillstandsüberwachung ausgeschaltet.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe" sowie unter "Ende eines Positioniervorgangs (Seite 2481)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	ZSPEED_R	DINT	L#1000	**Stillstandsbereich** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Stillstandsgeschwindigkeit festlegen

Stillstandsgeschwindigkeit

Die Stillstandsgeschwindigkeit dient als Referenzgeschwindigkeit. Erreicht die Istgeschwindigkeit während einer Positionierung die Stillstandsgeschwindigkeit und befindet sich der Antrieb im Zielbereich, dann wird "Position erreicht" (PEH) gemeldet. Der Positioniervorgang ist damit erfolgreich beendet.

Die Genauigkeit der Positionierung steigt, wenn der Wert der Stillstandsgeschwindigkeit herabgesetzt wird.
Mit dem Wert "0" ist die Stillstandsüberwachung ausgeschaltet.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Ende eines Positioniervorgangs (Seite 2481)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
88.0	ZSPEED_L	DINT	L#30000	**Stillstandsgeschwindigkeit** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm/min bis 100 000 µm/min

Kanal n Achse

Achsart festlegen

Achsart auswählen

Achsarten

Die Positionierbaugruppe kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

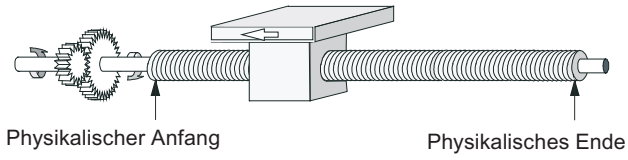

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

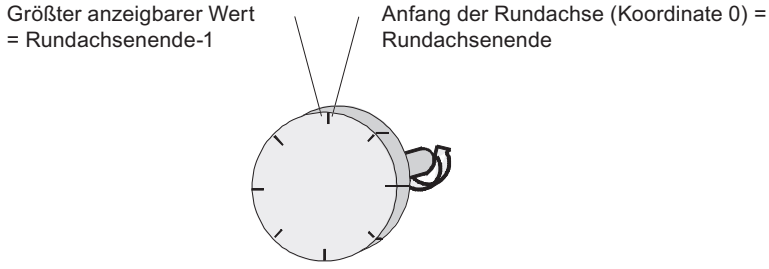

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	AXIS_TYPE	DINT	L#0	**Achsart** Zulässige Werte: • 0 = Linearachse • 1 = Rundachse

Software-Endschalter festlegen

Software-Endschalter

Software-Endschalter gibt es nur für die Linearachse.

Der Bereich, der durch die Software-Endschalter begrenzt wird, ist der Arbeitsbereich.

Der Software-Endschalter Anfang (SEA) muss immer kleiner sein als der Software-Endschalter Ende (SEE).

Die Software-Endschalter werden überwacht, nachdem die Achse synchronisiert ist.

- Bei Verwendung eines Inkrementalgebers:
  Die Synchronisation der Achse wird erreicht durch Referenzpunktfahrt oder Bezugspunkt setzen.
- Bei Verwendung eines Absolutwertgebers (SSI):
  Die Achse ist synchronisiert, nachdem die Baugruppe ein vollständiges, fehlerfreies Telegramm für den zugehörigen Kanal empfangen hat.
  Der Absolutwertgeber muss mindestens den Arbeitsbereich inklusive der Software-Endschalter abdecken.

Zusammenhang zwischen Arbeitsbereich, Geberbereich und Verfahrbereich

- Der Arbeitsbereich ist der Bereich, den Sie für Ihre Aufgabe durch die Software-Endschalter bestimmen.
- Der Geberbereich ist der vom Geber eindeutig abgedeckte Bereich. Der Geberbereich wird bei einer Linearachse von der Baugruppe symmetrisch über den Arbeitsbereich gelegt. D. h., die Baugruppe verschiebt den Geberbereich so, dass die Abstände zwischen den Software-Endschaltern und den Enden des Geberbereichs jeweils gleich sind (siehe Bild unten).
- Der Verfahrbereich ist der Wertebereich, den die Positionierbaugruppe verarbeiten kann. Der Verfahrbereich ist abhängig von der Auflösung.

Das folgende Bild zeigt den Zusammenhang zwischen Software-Endschaltern, Arbeitsbereich, Geberbereich und Verfahrbereich.

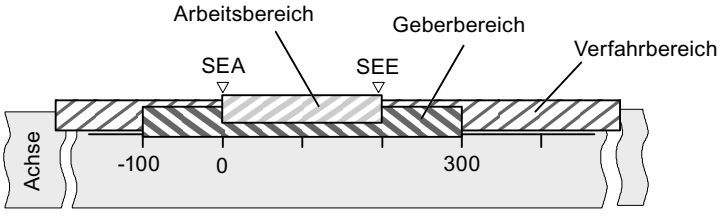

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L#-100000000	**Software-Endschalter Anfang**
68.0	SSW_END	DINT	L#100000000	**Software-Endschalter Ende** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Ende der Rundachse festlegen

Ende der Rundachse

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Der größte Wert, der bei einer Rundachse angezeigt wird, hat den Wert:

Ende der Rundachse [µm] - Auflösung [µm/Impuls] × 1 [Impuls]

Beispiel: Ende der Rundachse 1000 mm, Auflösung 1000 µm/Impuls

Die Anzeige springt:

- bei positiver Drehrichtung von 999 mm auf 0 mm
- bei negativer Drehrichtung von 0 mm auf 999 mm.

Rundachse mit Absolutwertgebern

Bei einer Rundachse mit einem Absolutwertgeber muss sich der Rundachsenbereich (0 bis Ende der Rundachse) genau mit dem Geberbereich des Absolutwertgebers decken:
Ende der Rundachse [µm] = Anzahl Geberumdrehungen × Weg/Geberumdrehung [µm]

Verendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
16.0	ENDROTAX	DINT	L#100000	**Ende der Rundachse** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Referenzpunktfahrt festlegen

Art der Referenzpunktfahrt auswählen

Art der Referenzpunktfahrt

Mit "Art der Referenzpunktfahrt" bestimmen Sie die Bedingungen für die Synchronisation der Achse:

- Die erste Angabe definiert die Startrichtung, in die die Referenzpunktfahrt startet.
- Die zweite Angabe definiert die Lage der Nullmarke, die zur Synchronisation führt, in Bezug auf den Referenzpunktschalter.

Detaillierte Informationen finden Sie unter Betriebsart "Referenzpunktfahrt" (Seite 2491).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
52.0	REFPT_TYPE	DINT	L#0	**Art der Referenzpunktfahrt** Zulässige Werte: • 0 = plus, Referenzpunktschalter in Richtung + • 1 = plus, Referenzpunktschalter in Richtung - • 2 = minus, Referenzpunktschalter in Richtung + • 3 = minus, Referenzpunktschalter in Richtung -

Referenzpunktkoordinate festlegen

Referenzpunktkoordinate

- **Inkrementalgeber:**
  Die Referenzpunktkoordinate benötigen Sie für die Betriebsart "Referenzpunktfahrt". Wenn nach dem Schreiben und Aktivieren von Maschinendaten die Achse nicht synchronisiert ist, wird der Istwert auf den Wert der Referenzpunktkoordinate gesetzt.
- **Absolutwertgeber (SSI)**
  Die Referenzpunktkoordinate benötigen Sie für die mechanische Justage des Gebers. Lesen Sie hierzu auch die Beschreibung der Absolutwertgeberjustage (Seite 2529), die das Zusammenspiel der Absolutwertgeberjustage mit den anderen Daten beschreibt.

Der Wert der Referenzpunktkoordinate muss innerhalb des Arbeitsbereichs liegen:

- **Linearachse**
  Einschließlich der Software-Endschalter.
- **Rundachse**
  Größer oder gleich 0 und kleiner als der Wert "Ende der Rundachse"
  (0 ≤ Referenzpunktkoordinate < Ende der Rundachse).

Verwendete Daten in Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Absolutwertgeberjustage festlegen

Absolutwertgeberjustage ermitteln

Absolutwertgeberjustage

Absolutwertgeberjustage ist nur für einen Absolutwertgeber (SSI) erforderlich.

Mit der Absolutwertgeberjustage und der Referenzpunktkoordinate wird der Wertebereich des Gebers eindeutig auf das Koordinatensystem der Achse abgebildet.

Weitere Informationen finden Sie unter "Vorgehen für eine Absolutwertgeberjustage (Seite 2529)" sowie in einem Beispiel (Seite 2530).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls
48.0	ENC_ADJ	DINT	L#0	**Absolutwertgeberjustage** Zulässiger Wertebereich: • 0 bis (2^{25} - 1)

Vorgehen für eine Absolutwertgeberjustage

Richtige Absolutwertgeberjustage ermitteln

Nach dem ersten Parametrieren müssen Sie weitere Schritte ausführen, damit eine korrekte Beziehung zwischen Geber und Koordinatensystem hergestellt wird. Der Ablauf ist unter Verwendung der Parametrierdialoge dargestellt.

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Ende" sein.
2. Rufen Sie den Auftrag "Bezugspunkt setzen (Seite 2505)" mit der Koordinate des unter 1. definierten Punktes auf.
   Die Positionierbaugruppe ermittelt nun für die im Kanal-DB eingetragene Referenzpunktkoordinate (REFPT im Kanal-DB) einen Geberwert, die Absolutwertgeberjustage. Diesen Wert können Sie mit dem Auftrag "Geberdaten lesen (Seite 2511)" auslesen.
3. Tragen Sie den ausgelesenen Wert in das Feld "Absolutwertgeberjustage" ein.
4. Speichern Sie Ihre Parametrierung mit der Exportfunktion in den entsprechenden Parameter-DB.

5. Laden Sie die Daten in HWCN zur CPU.
6. Damit die Daten aktiv werden, führen Sie einen Neustart der CPU durch.

Hinweis

Diesen Abgleich führen Sie bei der Inbetriebnahme einmal durch. Die Positionierbaugruppe ist nach einer Parametrierung im Anlauf synchronisiert, sobald ein vollständiges, fehlerfreies Telegramm des Gebers nach dem Anlauf empfangen wurde.

Alternative: Mechanische Justage eines Gebers

Eine korrekte Beziehung zwischen dem Koordinatensystem und dem Geber erreichen Sie auch folgendermaßen:

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Anfang" sein.
2. Tragen Sie diesen Koordinatenwert in den Maschinendaten als "Referenzpunktkoordinate" ein.
3. Lesen Sie den an dieser Position angezeigten Geberwert mit dem Auftrag "Geberdaten lesen (Seite 2511)" aus.
4. Tragen Sie diesen Wert als "Absolutwertgeberjustage" in die Maschinendaten ein.

Nach der Parametrierung wird dann immer ein korrekter Istwert angezeigt.

Anstelle der Schritte 3 und 4 können Sie auch den Geber über "Reset", falls vorhanden, auf Null setzen. Dann tragen Sie den Wert "0" als Absolutwertgeberjustage in die Maschinendaten ein.

Beispiel für eine Absolutwertgeberjustage

Beispiel einer Absolutwertgeberjustage

Für das Beispiel gelten folgende Annahmen:

- Referenzpunktkoordinate = -125 mm
- Arbeitsbereich von SSW_STRT = -1000 mm bis SSW_END = 1000 mm
- Absolutwertgeberjustage = 0
- Geberbereich = 2048 Inkremente mit einer Auflösung von 1 mm/Impuls
- Der verwendete Absolutwertgeber lässt sich nicht exakt mechanisch justieren und besitzt auch nicht die Möglichkeit, den Geberwert gezielt zu setzen.

Das folgende Bild zeigt ein Beispiel einer Absolutwertgeberjustage.

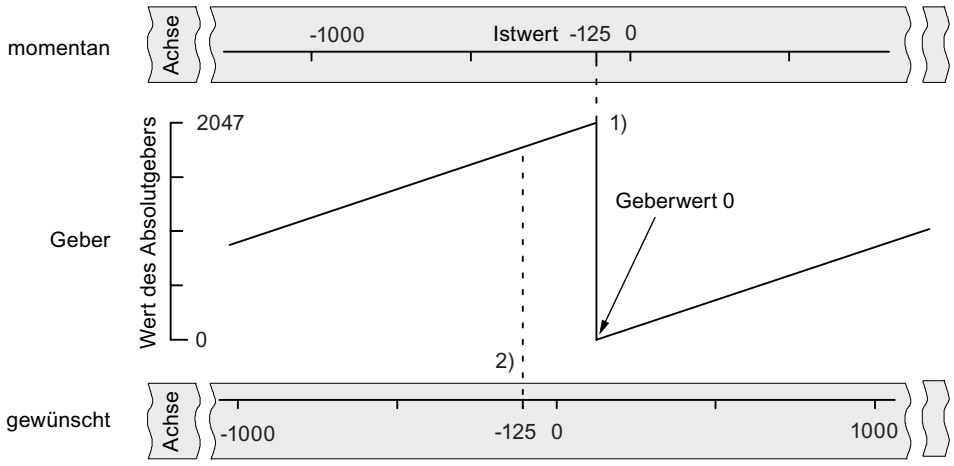

1) Zuordnung des Koordinatensystems zu den Geberwerten mit der eingestellten Absolutwertgeberjustage. Der Geberwert 0 entspricht dem Istwert -125.

2) Gewünschte Zuordnung des Koordinatensystems zum Geber. An dieser Position soll die Koordinate -125 sein.

Ergebnis nach "Bezugspunkt setzen"

Nach "Bezugspunkt setzen" sieht die Beziehung wie folgt aus:

Der Referenzpunktkoordinate auf der Achse (-125) wird der aus der Absolutwertgeberjustage ermittelte Geberwert (1798) zugeordnet. Das folgende Bild zeigt diese Zuordnung.

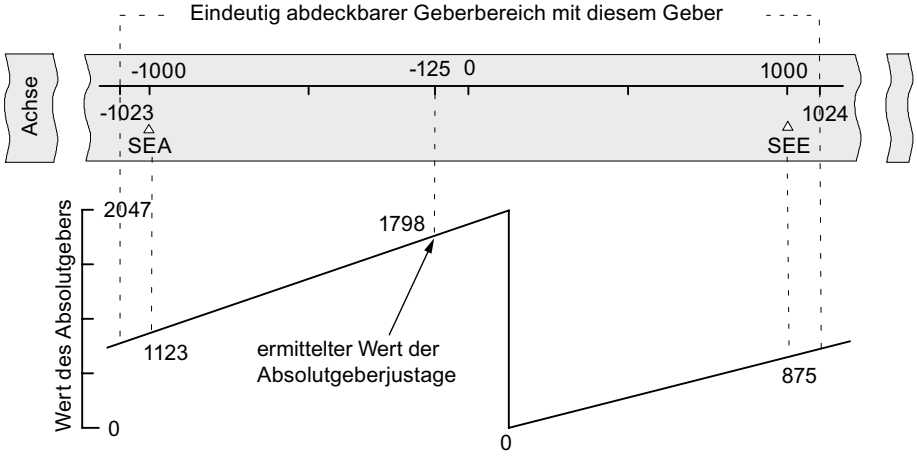

Der Geber liefert 2048 eindeutige Werte. Der Arbeitsbereich wird durch die Software-Endschalter festgelegt. Aufgrund der gewählten Auflösung von 1 mm pro Impuls kann der Geber aber einen größeren Arbeitsbereich abdecken als durch die Software-Endschalter vorgesehen ist.

Bei der eingestellten Auflösung ist der Arbeitsbereich bereits mit 2001 Werten abgedeckt. Es bleiben im Beispiel deshalb 47 Impulse "übrig", die sich symmetrisch um den Arbeitsbereich legen.

Startgeschwindigkeit auswählen

Startgeschwindigkeit bei Referenzpunktfahrt

Mit diesem Datum wählen Sie die Geschwindigkeit für den Start einer Referenzpunktfahrt.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
99.0	REFPT_SPD	BOOL	TRUE	**Startgeschwindigkeit bei Referenzpunktfahrt** Zulässige Werte: • 0 = Eilgang • 1 = Schleichgang

Kanal n Geber

Signalart auswählen

Signalart

Der Geber liefert Weginformationen zur Baugruppe, die diese auswertet und mit der Auflösung in einen Istwert umrechnet.

Sie können die Signalart von einem der folgenden Geber wählen:

- 5 V-Inkrementalgeber
  Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B und N, /N einsetzen.
- 24 V-Inkrementalgeber
  Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.
- Absolutwertgeber (SSI)
  Wählen Sie die Signalart "SSI absolut", wenn Sie einen Absolutwertgeber mit serieller Schnittstelle einsetzen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
20.0	ENC_TYPE	DINT	L#1	**Signalart** Zulässige Werte: • 1 = 5 V Differenzsignal • 2 = 24 V Spur A+B phasenversetzt • 3 = SSI 13 Bit Telegrammlänge [1)] • 4 = SSI 25 Bit Telegrammlänge [1)] • 5 = SSI 14 Bit Telegrammlänge [1)] • 6 = SSI 15 Bit Telegrammlänge [1)] • 7 = SSI 16 Bit Telegrammlänge [1)] • : • : • 14 = SSI 23 Bit Telegrammlänge [1)] • 15 = SSI 24 Bit Telegrammlänge [1)]

[1)] Die Auswahl der Telegrammlänge erfolgt im Dialogfeld "SSI".

Zählrichtung auswählen

Zählrichtung

Mit dem Maschinendatum "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an. Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

- normal = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen aufsteigenden Lageistwerten
- invertiert = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen absteigenden Lageistwerten

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
59.0	CNT_DIR	BOOL	FALSE	**Zählrichtung** Zulässige Werte: • 0 = normal • 1 = invertiert

Auflösung festlegen

Weg pro Geberumdrehung festlegen

Weg pro Geberumdrehung

Mit dem Maschinendatum "Weg pro Geberumdrehung" geben Sie der Positionierbaugruppe bekannt, welchen Weg das Antriebssystem je Geberumdrehung zurücklegt.

Der Wert "Weg pro Geberumdrehung" ist abhängig vom Aufbau der Achse und von der Art, wie der Geber angebracht ist. Sie müssen alle Übertragungsglieder wie Kupplungen oder Getriebe berücksichtigen. Das folgende Bild zeigt beispielhaft zwei unterschiedliche Geberanordnungen.

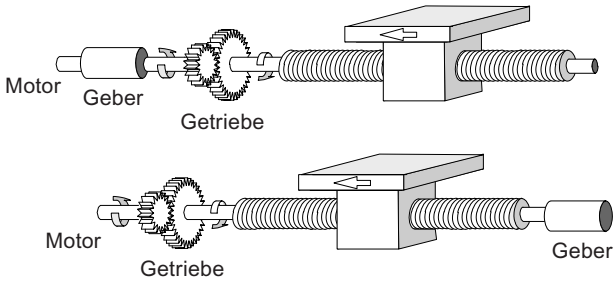

Der Zusammenhang zwischen den Maschinendaten "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" ist unter Auflösung (Seite 2535) beschrieben.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
24.0	DISP_REV	DINT	L#80000	**Weg pro Geberumdrehung** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm

Inkremente pro Geberumdrehung festlegen

Inkremente pro Geberumdrehung

Das Maschinendatum "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt. Aus diesem Wert und dem Maschinendatum "Weg pro Geberumdrehung" ermittelt die Positionierbaugruppe die Auflösung (Seite 2535).

Inkrementalgeber

Als Eingabe ist jeder Wert aus dem Wertebereich möglich. Die Baugruppe wertet die Inkremente 4fach aus.

Absolutwertgeber

Für die Grenzen muss zwischen den einzelnen Gebermodellen unterschieden werden. Als Eingabe sind nur Werte in 2er Potenzen erlaubt.

- Singleturn-Geber (Anzahl Umdrehungen = 1) mit 13 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192
- Multiturn-Geber (Anzahl Umdrehungen > 1) mit 25 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192
- Singleturn-Geber (Anzahl Umdrehungen = 1) mit 25 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 2^{25}

Linearmaßstäbe werden als Multiturn-Geber wie folgt parametriert:

- Inkremente pro Geberumdrehung = 8192
- Anzahl Umdrehungen × 8192 ≥ Anzahl der Schritte des Linearmaßstabs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
32.0	INC_REV	DINT	L#500	**Inkremente pro Geberumdrehung** Zulässiger Wertebereich: • 1 bis 2^{25}

Auflösung

Auflösung

Die Auflösung gibt an, welchem Verfahrweg ein Impuls entspricht. Sie ist ein Maß für die Genauigkeit der Positionierung und bestimmt auch den möglichen maximalen Verfahrbereich der Positionierbaugruppe.

Die Auflösung (AUFL) berechnen Sie wie in der folgenden Tabelle gezeigt:

	Inkrementalgeber	Absolutwertgeber
Eingangs-werte	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – 4fach-Auswertung – 1 Inkrement = 4 Impulse	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – 1 Inkrement = 1 Impuls
Berechnung	$\text{AUFL} = \frac{\frac{\text{Weg}}{\text{Geberumdrehung}}}{\frac{\text{Impulse}}{\text{Geberumdrehung}}}$	

Hinweis

Alle Positionsangaben werden auf das ganzzahlige Vielfache der Auflösung gerundet. Somit können sich die eingegebenen und die verwendeten Werte unterscheiden.

Wertebereich der Auflösung

Das gewählte Maßsystem bestimmt den Wertebereich der Auflösung:

Maßsystem	Angaben in ...	Wertebereich der Auflösung
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ … 1000×10^{-3} mm/Impuls
inch	10^{-4} inch	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} inch/Impuls
grad	10^{-4} grad 10^{-3} grad 10^{-2} grad	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} grad/Impuls $0{,}1 \times 10^{-3}$ … 1000×10^{-3} grad/Impuls $0{,}1 \times 10^{-2}$ … 1000×10^{-2} grad/Impuls

Beispiel

- Ein Inkrementalgeber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 5000
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 4 Impulse

  Damit ergibt sich die Auflösung (4fach-Auswertung):

$$\text{Auflösung} = \frac{1000\ \text{mm}}{5000\ \text{Inkremente}} = 0{,}2000\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{Impulse}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impuls}}$$

- Ein SSI-Geber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 4096
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 1 Impuls

  Damit ergibt sich die Auflösung:

$$\text{Auflösung} = \frac{1000\ \text{mm}}{4096\ \text{Inkremente}} = 0{,}2441\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2441\ \frac{\text{mm}}{\text{Impuls}}$$

Abhängigkeit zwischen Verfahrbereich und Auflösung

Der Verfahrbereich wird begrenzt durch die Zahlendarstellung in der Positionierbaugruppe. Diese Zahlendarstellung variiert in Abhängigkeit der Auflösung. Achten Sie deshalb darauf, dass Sie bei Vorgaben immer innerhalb der zulässigen Grenzen liegen.

Der maximale Verfahrbereich ist in der folgenden Tabelle dargestellt:

Auflösung (AUFL) liegt im Bereich	Maximaler Verfahrbereich
0,1 µm/Impuls ≤ AUFL < 1 µm/Impuls	-10^8 µm bis 10^8 µm (-100 m bis +100 m)
1 µm/Impuls ≤ AUFL ≤ 1000 µm/Impuls	-10^9 µm bis 10^9 µm (-1000 m bis +1000 m)

Daten für Absolutwertgeber (SSI) auswählen

Anzahl Geberumdrehungen

Anzahl Geberumdrehungen

Das Maschinendatum "Anzahl Geberumdrehungen" gibt es nur für Absolutwertgeber. Damit geben Sie die Anzahl der Umdrehungen an, die mit diesem Geber möglich sind.

Die **Gesamtschrittzahl des Gebers** ist kein Maschinendatum. Sie errechnet sich wie folgt:

Gesamtschrittzahl = Inkremente pro Geberumdrehung × Anzahl Umdrehungen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.0	NO_REV	DINT	L#1	**Anzahl Geberumdrehungen** Zulässiger Wertebereich: • 1 (Singleturn-Geber) • 2 bis 4096 in 2er Potenzen (Multiturn-Geber)

Baudrate

Baudrate

Mit dem Maschinendatum "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur Positionierbaugruppe.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 200 m → 188 kHz
- 100 m → 375 kHz
- 40 m → 750 kHz
- 12 m → 1500 kHz

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
40.0	BAUDRATE	DINT	L#0	**Baudrate** Zulässige Werte: • 0 = 188 kHz • 1 = 375 kHz • 2 = 750 kHz • 3 = 1500 kHz

Telegrammlänge

Telegrammlänge

Mit der "Telegrammlänge" bestimmen Sie den Taktrahmen, den die Positionierbaugruppe für Absolutwertgeber ausgibt.

Zulässig ist eine Telegrammlänge von 13 Bit bis 25 Bit.

Verwendete Daten im Parameter-DB

Die hier ausgewählte Telegrammlänge ist zusammen mit der Signalart unter dem Datum ENC_TYPE (Seite 2532) im Parameter-DB abgelegt.

Parität

Parität

Mit der "Parität" bestimmen Sie, ob für Absolutwertgeber keine, gerade oder ungerade Parität übertragen werden soll.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
28.0	PARITY	DINT	L#0	**Parität** Zulässige Werte: • 0 = keine • 1 = gerade Parität (even) • 2 = ungerade Parität (odd)

Überwachung auswählen

Überwachung

Drahtbruch

Mit dem Aktivieren der Überwachung überwacht die Positionierbaugruppe bei einem 5 V-Inkrementalgeber und einem Absolutwertgeber alle Leitungen. Die Überwachung erkennt:

- Drahtbruch
- Kurzschluss der einzelnen Leitungen
- Flankenabstand der Zählimpulse (auch bei 24 V-Inkrementalgeber)

Zur Überwachung bei einem 24 V-Inkrementalgeber müssen Sie eine Überwachungszeit MON_TIME (Seite 2523) > 0 einstellen.

Bei 5 V-Inkrementalgebern ohne Nullmarke müssen Sie entweder die Drahtbruch-Überwachung ausschalten oder die Signale N und /N extern verschalten.

Telegrammfehler

Die Baugruppe überwacht das Telegramm eines Absolutwertgebers (SSI) auf:

- Start- und Stoppbitfehler

Die Überwachung auf Telegrammfehler ist bei Absolutwertgebern (SSI) nicht abschaltbar.

Fehlimpulse (Inkrementalgeber)

Ein Inkrementalgeber muss zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Inkremente liefern.

Die Positionierbaugruppe überprüft, ob die Nullmarke eines Inkrementalgebers zum richtigen Geberstand auftritt.

Für Geber ohne Nullmarken müssen Sie die Überwachung der Fehlimpulse abschalten.

Ebenso müssen Sie die Drahtbruchüberwachung abschalten oder die Nullmarkeneingänge N und /N extern verschalten.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Überwachungen:
63.0	MON_WIRE	BOOL	TRUE	1 = Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1 = Telegrammfehler (muss immer 1 sein)
63.2	MON_PULSE	BOOL	TRUE	1 = Fehlimpulse

Kanal n Schrittmaße

Schrittmaße festlegen

Definition

Schrittmaße sind Zielvorgaben, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ" bzw. "Schrittmaßfahrt absolut" angesteuert werden können.

Möglicher Bereich für Schrittmaße

Das Ziel, das angefahren werden soll, muss mindestens um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegen. Das folgende Bild zeigt die Grenzen für eine Schrittmaßvorgabe.

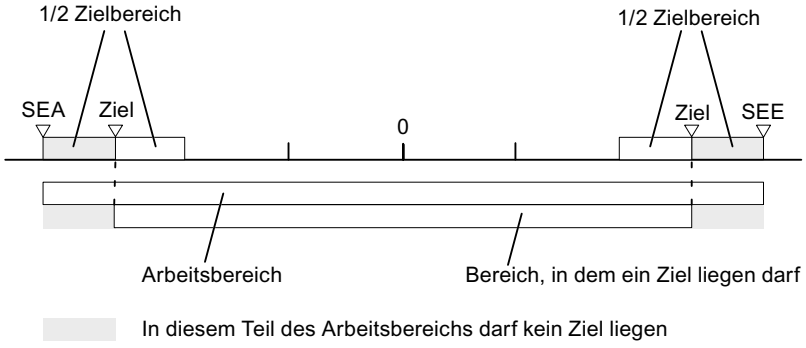

Schrittmaßnummer 1 bis 100

Schrittmaßnummer 1 bis 100

Sie haben die Möglichkeit, maximal 100 Schrittmaße in eine Tabelle einzutragen, die sowohl für die Betriebsart "Schrittmaßfahrt relativ" als auch für "Schrittmaßfahrt absolut" gültig sind.

Beachten Sie, dass die Positionierbaugruppe für die "Schrittmaßfahrt relativ" keine negativen Werte erlaubt. Die Werte werden von der Positionierbaugruppe je nach Bewegungsrichtung als positive oder als negative Differenz interpretiert.

Hinweis

Der Eintrag erfolgt in der Einheit des eingestellten Maßsystems. Beachten Sie dabei insbesondere die Nachkommastellen.

Zahlenbeispiel:

- Schrittmaß: 800 mm
- Maßsystem: 10^{-3} mm
- Eingabe in Parameter-DB: 800000

Empfehlung: Definieren Sie in der Schrittmaßtabelle getrennte Bereiche für relative und absolute Schrittmaße.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaße 1 ... 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaße 51 ... 100) schreiben
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaße 1 ... 50) lesen
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaße 51 ... 100) lesen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
120.0	TRGL1.TRG[1]	DINT	L#0	Schrittmaßnummer 1	Schrittmaßtabelle 1
. . .	. . .	. .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Schrittmaßnummer 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Schrittmaßnummer 51	Schrittmaßtabelle 2
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Schrittmaßnummer 100	

Schrittmaßnummer 254

Schrittmaßnummer 254

Unabhängig von der Schrittmaßtabelle können Sie die Schrittmaßnummer 254 als eine weitere Vorgabe des Weges verwenden. Für die Umschalt- und Abschaltdifferenzen haben die Einträge aus dem Parameter-DB für dieses Schrittmaß Gültigkeit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 254 schreiben
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 254

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus

Schrittmaßnummer 255

Schrittmaßnummer 255

Mit der Schrittmaßnummer 255 steht Ihnen eine weitere Vorgabe des Weges zur Verfügung.

Die Abschaltdifferenzen und die Umschaltdifferenzen übergeben Sie zusammen mit dem Schrittmaß. Anders als bei den anderen Schrittmaßen verwendet das Schrittmaß 255 die im Kanal-DB festgelegten Werte für die Abschalt- und Umschaltdifferenz. Die Einträge aus den Maschinendaten haben für dieses Schrittmaß keine Gültigkeit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.3	TRG255_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 255 schreiben
100.0	TRG255	DINT	L#0	Schrittmaß für Schrittmaßnummer 255
104.0	CHGDIF255	DINT	L#0	Umschaltdifferenzen für Schrittmaßnummer 255
108.0	CUTDIF255	DINT	L#0	Abschaltdifferenzen für Schrittmaßnummer 255

Steuer- und Rückmeldeschnittstelle der FM 351

Anwendung

In speziellen Anwendungen oder in einer Alarmebene kann ein besonders schneller Zugriff auf Rückmelde- und Steuersignale erforderlich sein. Diese Daten erreichen Sie direkt über die Ein- und Ausgangsbereiche der Baugruppe.

Zur Anlaufkoordinierung nach jedem Anlauf der Baugruppe, z. B. nach Baugruppe stecken oder nach einem STOP-RUN-Übergang der CPU, müssen Sie die Anweisung ABS_CTRL solange aufrufen, bis durch RETVAL = 0 das Ende des Anlaufs angezeigt wird. Anschließend dürfen Sie die Anweisung ABS_CTRL nicht mehr anwenden.

Hinweis

Die Verwendung der Anweisung ABS_CTRL zusammen mit einem Schreibzugriff ist nicht möglich.

Rückmeldesignale lesen durch Direktzugriff

Die Byte-Adressen sind relativ zur Anfangsadresse der Ausgänge des jeweiligen Kanals angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

Anfangsadresse Kanal 1 = Anfangsadresse Baugruppe

Anfangsadresse Kanal 2 = Anfangsadresse Baugruppe + 8

In AWL greifen Sie mit den Befehlen PEB (1 Byte lesen), PEW (2 Byte lesen) und PED (4 Byte lesen) auf die Daten zu.

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) der FM 351.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	PARA	intern	intern	DATA_ERR	OT_ERR	DIAG	intern	intern
Byte 1	CHGOVER	CUTOFF	ZSPEED	SPEED_OUT	0	WAIT_EI	WORKING	ST_ENBLD
Byte 2	MODE_OUT							
Byte 3	POS_RCD	0	0	0	GO_P	GO_M	0	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								

Beispiel: Lageistwert ACT_POS

Die Anfangsadresse der Baugruppe ist 512.

AWL	Beschreibung
L PED 516	//Aktuellen Lageistwert (ACT_POS) von Kanal 1
	//mit Direktzugriff lesen:
	//Anfangsadresse des Kanals + 4

Steuersignale schreiben durch Direktzugriff

Die Byte-Adressen sind relativ zur Anfangsadresse der Eingänge des jeweiligen Kanals angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

Anfangsadresse Kanal 1 = Anfangsadresse Baugruppe

Anfangsadresse Kanal 2 = Anfangsadresse Baugruppe + 8

In AWL greifen Sie mit den Befehlen PAB (1 Byte schreiben), PAW (2 Byte schreiben) und PAD (4 Byte schreiben) auf die Daten zu.

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) der FM 351.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	OT_ERR_A	0	0	0
Byte 1	DRV_EN	0	0	0	DIR_P	DIR_M	STOP	START
Byte 2	MODE_IN							
Byte 3	MODE_TYPE							
Byte 4	Reserviert							
Byte 5								
Byte 6								
Byte 7								

Beispiel: START-Signale Kanal 2

Die Anfangsadresse der Baugruppe ist 512.

AWL	Beschreibung
L 2#10001000	//DRV_EN und DIR_P auf 1 setzen
T PAB 521	//Signale mit Direktzugriff für Kanal 2 schreiben:
	//Anfangsadresse der Baugruppe + 8 + 1

Parameterübertragungswege

Unter Parameter werden nachfolgend Maschinendaten und Schrittmaße verstanden.

Das folgende Bild zeigt die Parameterübertragungswege zur FM x51.

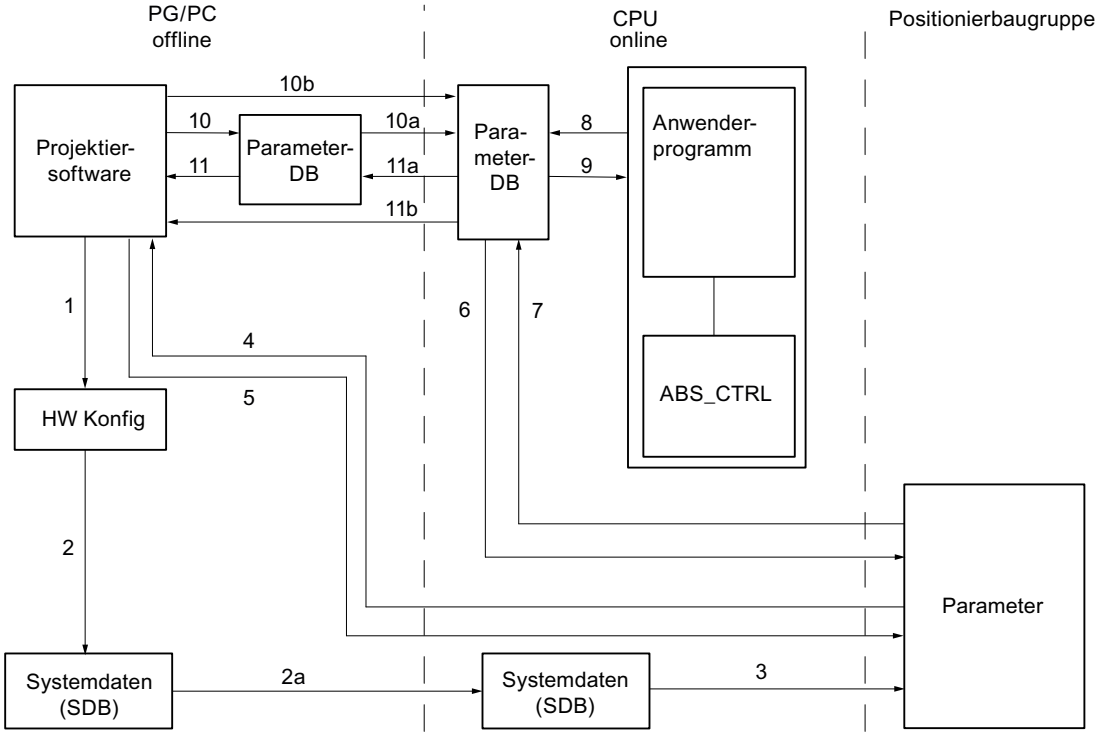

1 Parameter in der Projektiersoftware speichern.
2 HW-Konfiguration speichern und übersetzen.
2a HW-Konfiguration zur CPU laden. Die CPU führt automatisch Schritt 3 aus.
3 Die CPU schreibt die Parameter bei Systemparametrierungen zur Baugruppe.
4 Parameter eines Kanals der Baugruppe mit dem Befehl "Zielsystem laden in PG" ins PG laden.
5 Parameter aus der Projektiersoftware mit dem Befehl "Zielsystem laden" in einen Kanal der Baugruppe laden.
6 Parameter über Aufträge des Anwenderprogramms in einen Kanal der Baugruppe schreiben.
7 Parameter über Aufträge des Anwenderprogramms aus einem Kanal der Baugruppe lesen.
8 Parameter aus dem Anwenderprogramm in dem Online-DB ablegen.
9 Parameter aus dem Online-DB in das Anwenderprogramm einlesen.
10 Parameter aus der Projektiersoftware in den Offline-DB exportieren.
10a Offline-DB in die CPU laden.
10b Parameter aus der Projektiersoftware in den Online-DB exportieren.
11 Parameter aus dem Offline-DB in die Prokjektiersoftware importieren.
11a Parameter aus dem Online-DB in das PG laden.
11b Parameter aus dem Online-DB in die Projektiersoftware importieren.

Einige Anwendungsfälle für die Übertragung von Parametern

Anwendungsfall	Schritte
Sie bearbeiten die Parameter mit der Projektiersoftware. Die Kanäle der Baugruppe sollen anschließend automatisch beim Anlauf parametriert werden.	Führen Sie die Schritte 1, 2 und 2a aus.
Sie ändern Parameter bei der Inbetriebnahme im Testbetrieb in der Projektiersoftware.	Führen Sie die Schritte 4 und 5 aus.
Die bei der Inbetriebnahme veränderten Parameter sollen anschließend beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 1, 2 und 2a aus.
Sie erstellen die Parameter mit der Projektiersoftware. Die Kanäle der Baugruppe sollen beim Anlauf nur vom Anwenderprogramm über Datenbausteine parametriert werden.	Führen Sie die Schritte 10, 10a und 6 oder 10b und 6 aus.
Sie wollen komfortabel Vorratsdaten für Rezepturen erstellen.	Führen Sie die Schritte 10 und 10a aus.
Sie erstellen die Parameter mit der Projektiersoftware. Diese sollen dem Anwenderprogramm für temporäre Änderungen zur Verfügung stehen.	Führen Sie die Schritte 1, 2 und 2a für die automatische Parametrierung aus.
	Führen Sie die Schritte 10, 10a, 9, 8 und 6 für den Zugriff durch das Anwenderprogramm aus.
Sie ändern vorhandene Parameter ausschließlich über das Anwenderprogramm.	Führen Sie die Schritte 7, 9, 8 und 6 aus.
Sie wollen die über das Anwenderprogramm geänderten Daten mit der Projektiersoftware ansehen.	Führen Sie die Schritte 11a und 11 oder nur 11b aus.
Die vom Anwenderprogramm veränderten Parameter sollen auch beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 11b oder 11a, 11 und dann 1, 2, 2a aus.

6.2.5.5 FM 351 aus dem Anwenderprogramm verwenden

Einbinden der FM 351 in ein Anwenderprogramm

Für das Einbinden der FM 351 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen ABS_INIT und ABS_CTRL werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 351. Der Kanal-DB wird automatisch als Instanz-DB zur Anweisung ABS_CTRL angelegt, wenn Sie die Anweisung ABS_CTRL in das Projekt einbinden.

Alle zu einem Kanal der Baugruppe gehörenden Parameterdaten (Maschinendaten, Schrittmaßtabellen) liegen im Parameter-DB für die Baugruppe. Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ ABS_PARATYPE in Ihr Projekt ein.

Außerdem gibt es für die FM 351 die Anweisung ABS_DIAG, mit der Sie Diagnoseinformationen in den Diagnose-DB übertragen können. Der Diagnose-DB wird automatisch als Instanz-DB zur Anweisung ABS_DIAG angelegt, wenn Sie die Anweisung ABS_DIAG in das Projekt einbinden.

Inbetriebnahme der Baugruppe

Auch ohne Anwenderprogramm ist ein erster Test der Baugruppe mit dem Dialog "Inbetriebnahme" möglich. Beachten Sie dabei insbesondere die nachfolgende Warnung.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Zur Vermeidung von Schäden an Personen und Gegenständen beachten Sie folgende Punkte:

- Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners.
  Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Software-Ausfalls die Anlage sicher ausgeschaltet werden kann.
- Installieren Sie Hardware-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
- Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.
- Durch paralleles Steuern und Beobachten der FM 351 von Ihrem Programm und aus dem Dialog "Inbetriebnahme" kann es zu Konflikten kommen, deren Auswirkungen nicht eindeutig sind. Schalten Sie deshalb immer die CPU in den STOP-Zustand, wenn Sie mit dem Dialog "Inbetriebnahme" arbeiten, oder deaktivieren Sie Ihr Programm.

6.2.5.6 FM 351 in Betrieb nehmen

6.2.5.7 FM 351: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Fehler-LEDs der FM 351

Wo wird der Fehler angezeigt?

Leuchtet die rote Sammelfehler-LED, dann ist entweder ein Fehler auf der Baugruppe (Interner Fehler) oder bei den Leitungsanschlüssen (Externer Fehler) aufgetreten.

Die roten LEDs CH 1 und CH 2 zeigen einen Kanalfehler an.

Das folgende Bild zeigt die Lage der Fehler-LEDs an der FM 351.

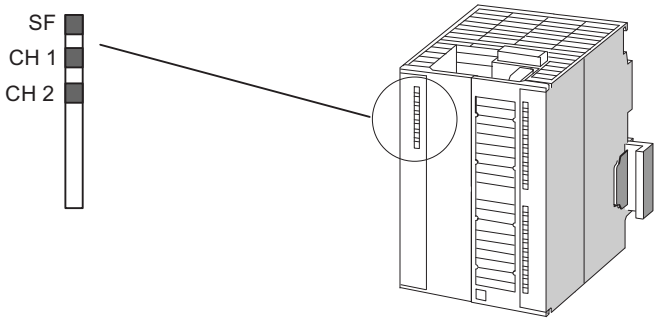

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Fehler-LEDs angezeigt werden.

LED	Bedeutung	Erläuterungen
SF	Sammelfehler	Diese LED zeigt einen Fehlerzustand der FM 351 an. Diagnosealarm (interner oder externer (Kanal-)Fehler) Zur Beseitigung des Fehlers siehe Fehlerklassen (Seite 2555).
CH 1 CH 2	Kanalfehler 1 Kanalfehler 2	Diese LEDs zeigen einen Kanalfehler am Kanal 1 bzw. am Kanal 2 an. • Drahtbruch am Geber • Fehler Absolutwertgeber • Fehlimpulse Inkrementalgeber • Betriebsfehler • Parametrierfehler

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Alarmbearbeitung

Die Positionierbaugruppe kann Diagnosealarme auslösen. Diese Alarme bearbeiten Sie in einem Alarm-OB. Wenn ein Alarm ausgelöst wird, ohne dass der zugehörige OB geladen ist, geht die CPU in STOP.

Die Bearbeitung der Diagnosealarme geben Sie folgendermaßen frei:

1. Wählen Sie die Baugruppe in HWCN aus.
2. Geben Sie über "Bearbeiten > Objekteigenschaften > Grundparameter" den Diagnosealarm frei.
3. Speichern und übersetzen Sie die Konfiguration.
4. Laden Sie die Konfiguration in die CPU.

Übersicht über die Diagnosealarme

Folgende Ereignisse und Fehler lösen einen Diagnosealarm aus:

- Betriebsfehler
- Fehlerhafte Maschinendaten
- Diagnosefehler

Sie finden diese Fehler unter "Fehlerklassen (Seite 2555)" ausführlich erläutert.

Reaktion der Positionierbaugruppe bei einem Fehler mit Diagnosealarm

- Die Positionierung wird abgebrochen.
- Die Synchronisation wird bei folgenden Diagnosealarmen gelöscht:
  - Frontstecker fehlt, externe Hilfsspannung für die Geberversorgung fehlt
  - ein Nullmarkenfehler wurde erkannt, Leitungsfehler (5 V-Gebersignale)
  - der Verfahrbereich wurde verlassen (wird durch einen Betriebsfehler gemeldet)
  - Istwert setzen nicht ausführbar (wird durch einen Betriebsfehler gemeldet)
- Steuersignale START, DIR_P und DIR_M werden, bis auf eine Ausnahme, nicht mehr bearbeitet.
  Ausnahme:
  Bei einem Betriebsfehler ist noch ein Tippen in Richtung des Arbeitsbereichs möglich.
- Funktionsschalter und Aufträge werden weiter bearbeitet.

Positionierbaugruppe erkennt einen Fehler ("kommend")

Ein Diagnosealarm ist "kommend", wenn mindestens ein Fehler ansteht. Wurden nicht alle Fehler beseitigt, werden die restlichen anstehenden Fehler nochmals "kommend" gemeldet.

Ablauf:

1. Die Positionierbaugruppe erkennt einen oder mehrere Fehler und löst einen Diagnosealarm aus. Die LED "SF" und je nach Fehler die LEDs "CH1" / "CH2" leuchten. Der Fehler wird in den Diagnosepuffer eingetragen.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Sie können die Startinfo des OB 82 auswerten.
4. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
5. Weitere Informationen erhalten Sie, wenn Sie die Anweisung ABS_DIAG aufrufen.

Positionierbaugruppe erkennt den Übergang in den fehlerfreien Zustand ("gehend")

Ein Diagnosealarm ist nur dann "gehend", wenn der letzte Fehler auf der Baugruppe behoben wurde.

Ablauf:

1. Die Positionierbaugruppe erkennt, dass alle Fehler behoben wurden und löst einen Diagnosealarm aus. Die LED "SF" leuchtet nicht mehr. Der Diagnosepuffer wird nicht verändert.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
4. Werten Sie das Bit OB82_MDL_DEFECT aus.
   Wenn dieses Bit "0" ist, sind keine Fehler auf der Baugruppe vorhanden. Ihre Auswertung kann hier enden.

Diagnosealarme in Abhängigkeit des CPU-Status

- Im STOP-Zustand der CPU sind die Diagnosealarme von der Positionierbaugruppe gesperrt.
- Werden im STOP-Zustand der CPU nicht alle der anstehenden Fehler behoben, meldet die Baugruppe die noch nicht beseitigten Fehler noch einmal nach dem Übergang in den RUN-Zustand als "kommend".
- Werden alle anstehenden Fehler im STOP-Zustand der CPU behoben, dann wird der fehlerfreie Zustand der Baugruppe nach dem Übergang in den RUN-Zustand **nicht** mit einem Diagnosealarm gemeldet.

Fehlerauswertung im Anwenderprogramm

Fehlerreaktion im Anwenderprogramm

Im Anwenderprogramm können Sie gezielt auf Fehler reagieren. Dafür stehen Ihnen folgende Mittel zur Verfügung:

- Die Rückgabewerte RETVAL der eingebundenen Standard-FB:
  Dieser Wert wird bei jedem Aufruf des Bausteins neu ermittelt.
  RETVAL = -1 ist eine Sammelanzeige für einen synchronen Fehler in einem Auftrag oder bei der Kommunikation mit der Baugruppe.
- Zu jedem Auftrag gehört ein Fehlerbit _ERR als Sammelanzeige für einen Fehler im Auftrag oder einem seiner Vorgänger in einer Auftragskette:
  Das Fehlerbit wird für einen Schreibauftrag und seine Folgeaufträge gesetzt, wenn ein Datenfehler von der Baugruppe gemeldet wurde oder ein Kommunikationsfehler aufgetreten ist.
  Bei Leseaufträgen wird das Fehlerbit für den betroffenen Auftrag gesetzt, wenn ein Kommunikationsfehler aufgetreten ist.
  Die Fehlerbits werden nach der Bearbeitung eines Auftrags von der Anweisung ABS_CTRL neu gesetzt. Diese sollten aber für eine Fehlerauswertung vom Anwenderprogramm zurückgenommen werden.
- Das Rückmeldesignal DATA_ERR als Sammelanzeige für einen Fehler, den die Baugruppe bei einem Schreibauftrag erkannt hat. Das Signal wird beim nächsten Schreibauftrag neu ermittelt.

- Das Rückmeldesignal OT_ERR (Bedienfehler) als Sammelanzeige für einen Fehler, den die Baugruppe beim Starten einer Fahrt erkannt hat. Der Fehler muss nach Behebung der Ursache mit OT_ERR_A = 1 quittiert werden.
- Das Rückmeldesignal DIAG wird gesetzt, wenn sich der Inhalt des Diagnosepuffers geändert hat. Dieses Signal kann später als die Signale DATA_ERR und OT_ERR kommen.
- Der Kommunikationsfehler JOB_ERR enthält den Fehlercode (Seite 2554) bei einem Kommunikationsproblem zwischen Anweisung und Baugruppe. Der Wert wird nach der Bearbeitung eines Auftrags neu ermittelt und bei der Anweisung ABS_CTRL im Kanal-DB und bei der Anweisung ABS_DIAG im Diagnose-DB abgelegt.
- Die Anweisung ABS_DIAG zum Auslesen des Diagnosepuffers der Baugruppe. Hier können Sie die Fehlerursachen für synchrone und asynchrone Ereignisse erfahren.
- Diagnosealarme für die schnelle Reaktion auf Ereignisse im Diagnosealarm-OB (z. B. OB 82).

Auswertung eines Diagnosealarms im Anwenderprogramm

Folgende Einträge in den Lokaldaten des Diagnosealarm-OBs (z. B. OB 82) werden von der Positionierbaugruppe gesetzt. Die Fehler werden auch im Diagnosepuffer eingetragen (Fehlerklasse 128 (Seite 2562)).

Adresse	Name	Typ	Kommentar
0.0	OB82_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#38: gehendes Ereignis B#16#39: kommendes Ereignis
1.0	OB82_FLT_ID	BYTE	Fehlercode (B#16#42)
2.0	OB82_PRIORITY	BYTE	Prioriätsklasse: B#16#1A im Betriebszustand RUN B#16#1C im Betriebszustand ANLAUF
3.0	OB82_OB_NUMBR	BYTE	OB-Nummer (82)
4.0	OB82_RESERVED_1	BYTE	Reserviert
5.0	OB82_IO_FLAG	BYTE	Eingabebaugruppe: B#16#54
6.0	OB82_MDL_ADDR	INT	Logische Basisadresse der Baugruppe, in der der Fehler aufgetreten ist
8.0	OB82_MDL_DEFECT	BOOL	Baugruppenstörung
8.1	OB82_INT_FAULT	BOOL	Interner Fehler
8.2	OB82_EXT_FAULT	BOOL	Externer Fehler
8.3	OB82_PNT_INFO	BOOL	Kanalfehler vorhanden
8.4	OB82_EXT_VOLTAGE	BOOL	Externe Hilfsspannung fehlt
...	nicht verwendet		
10.3	OB82_WTCH_DOG_FLT	BOOL	Zeitüberwachung hat angesprochen
...	nicht verwendet		
12.0	OB82_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Diagnosepuffer der Baugruppe

Diagnoseereignisse

Der Diagnosepuffer der Baugruppe enthält maximal 9 Diagnoseereignisse und ist als Ringpuffer organisiert.

Ein Diagnoseereignis wird in den Puffer geschrieben, wenn eine (Fehler-) Meldung "kommend" erkannt wird. Dies kann eine Meldung, ein synchroner Fehler (Datenfehler, Bedienfehler) oder auch ein asynchroner Fehler (Betriebsfehler und Diagnosefehler) sein. Aus einer Fehlerursache können auch mehrere Einträge als Folgefehler entstehen. Gehende Meldungen erzeugen keine Einträge im Diagnosepuffer.

Für jedes Diagnoseereignis werden angegeben:

- Status (immer kommend)
- Interner Fehler
- Externer Fehler
- Fehlerklasse
- Fehlernummer
- Kanalnummer
- Schrittmaßnummer (bei Schrittmaßtabellenfehler)

Wenn ein Diagnoseereignis in den Diagnosepuffer geschrieben wird, wird das Rückmeldesignal DIAG = 1 in allen parametrierten Kanälen gesetzt.

Der Diagnosepuffer kann als Ganzes mit der Anweisung ABS_DIAG in einen Datenbaustein (Diagnose-DB) übertragen werden oder über den Fehlerauswertungsdialog der Projektiersoftware angezeigt werden. Wird der Diagnosepuffer gelesen, setzt die Baugruppe das Rückmeldesignal DIAG = 0.

Hinweis

Wird der Diagnosepuffer gleichzeitig von der Anweisung ABS_DIAG und dem Fehlerauswertungsdialog gelesen, kann es sein, dass ein neu eingetroffenes Diagnoseereignis vom Programm nicht erkannt wird.

Daten und Aufbau des Diagnose-DB

Daten und Aufbau des Diagnose-DB

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Die folgende Tabelle zeigt den Aufbau des Diagnose-DB.

Adresse	Name	Typ	Anfangswert	Kommentar
0.0	MOD_ADDR	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Fehlernummer des Kommunikationsfehlers
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten vierter Eintrag
302.0	DIAG[5]	STRUCT		Diagnosedaten fünfter Eintrag
312.0	DIAG[6]	STRUCT		Diagnosedaten sechster Eintrag
322.0	DIAG[7]	STRUCT		Diagnosedaten siebter Eintrag
332.0	DIAG[8]	STRUCT		Diagnosedaten achter Eintrag
342.0	DIAG[9]	STRUCT		Diagnosedaten neunter Eintrag

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 2: Bedienfehler 4: Datenfehler 5: Maschinendatenfehler 6: Schrittmaßtabellen-Fehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer
+6.0	CH_NO	INT	0	Kanalnummer
+8.0	TRG_NO	INT	0	Schrittmaßnummer

Liste der JOB_ERR-Meldungen

JOB_ERR-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80A0	32928	-32608	Negative Quittung beim Lesen von der Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2 [1]	32930	-32606	Protokollfehler bei Layer 2
80A3 [1]	32931	-32605	Protokollfehler bei User-Interface / User
80A4 [1]	32932	-32604	Kommunikation am K-Bus gestört
80B0	32944	-32592	Datensatz/Auftrag unbekannt.
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0 [1]	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1 [1]	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2 [1]	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3 [1]	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4 [1]	32964	-32572	Kommunikationsfehler
80C5 [1]	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6 [1]	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)

[1] Die Fehler 80A2 … 80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein.

Fehlerklassen

Betriebsfehler

Fehlerklasse 1: Betriebsfehler

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt. Die Betriebsfehler führen zum Abbruch der Positionierung, außer bei Fehlernummer 9. Fehlernummer 9 führt zum Absteuern der Positionierung.

Die folgende Tabelle zeigt die möglichen Betriebsfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
2	**Software-Endschalter Ende überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
3	**Verfahrbereichsanfang überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
4	**Verfahrbereichsende überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
5	**Fehler beim Zieleinlauf**		ja
	Ursache	Zielbereich wurde innerhalb der Überwachungszeit nicht erreicht.	
6	**Stillstandsbereich verlassen**		ja
	Ursache	Der Istwert liegt außerhalb des Stillstandsbereichs.	
7	**Mitkopplung**		ja
	Ursache	Istwertänderung > ½ Stillstandsbereich in die falsche Richtung.	
8	**Fehlende oder zu geringe Istwertänderung**		ja
	Ursache	Keine Istwertänderung oder eine Istwertänderung entgegen der Sollrichtung innerhalb der Überwachungszeit.	
9	**Ziel überfahren (FM 451)**		ja
	Ursache	Beim "Fliegend Istwert setzen" wurde das Ziel überfahren.	
10	**Zielbereich überfahren**		ja
	Ursache	Zielbereich wurde nach Zieleinlauf überfahren.	
11	**Umschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Umschaltpunkt.	
12	**Abschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Abschaltpunkt/Umkehrpunkt.	
13	**Zielbereichsanfang fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Zielbereich.	
14	**Änderung größer als der halbe Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwertsprünge.	

Nr.	Bedeutung		Diagnosealarm
15	**Änderung größer als der Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwert-sprünge.	
16	**Schrittmaß für Schrittmaßnummer 252 nicht übertragen (FM 451)**		ja
	Ursache	Das Schrittmaß wurde nicht übertragen.	
17	**Schrittmaß für Schrittmaßnummer 252 nicht anfahrbar (FM 451)**		ja
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Schrittmaß ist kleiner als die Umschalt- bzw. Abschaltdifferenz.	
18	**Falsches Schrittmaß für Schrittmaßnummer 252 (FM 451)**		ja
	Ursache	Das Schrittmaß liegt außerhalb des Arbeitsbereichs.	

Bedienfehler

Fehlerklasse 2: Bedienfehler

Bedienfehler werden bei Änderung der Steuersignale im Nutzdatenbereich erkannt. Die Bedienfehler führen zum Absteuern der Positionierung.

Die folgende Tabelle zeigt die möglichen Bedienfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Unzulässige Betriebsart**		nein
	Ursache	Die angewählte Betriebsart ist unzulässig.	
3	**Unzulässiger Schnittstellenauftrag**		nein
	Ursache	Das angewählte Signal ist mit dieser Betriebsart unzulässig.	
4	**Falscher Betriebsartenparameter**		nein
	Ursache	• Bei der Betriebsart "Tippen" ist die Geschwindigkeitsvorgabe ungleich der Eil- oder Schleichgeschwindigkeit. • Bei der Betriebsart "Schrittmaßfahrt" ist das Schrittmaß ungleich 1 bis 100 oder ungleich 252, 254 und 255.	
5	**Startfreigabe nicht vorhanden**		nein
	Ursache	Startfreigabe beim Starten nicht vorhanden.	
7	**Ziel/Zielbereich liegt außerhalb des Arbeitsbereichs**		nein
	Ursache	Vorgegebenes bzw. berechnetes Ziel liegt außerhalb der Software-Endschalter.	
8	**Achse nicht parametriert**		nein
	Ursache	Falsche oder keine Maschinendaten wurden für die Achse parametriert.	
9	**Achse nicht synchronisiert**		nein
	Ursache	Die Betriebsart "Schrittmaßfahrt" ist nur mit einer bereits synchronisierten Achse möglich.	
10	**Ziel/ Wegstück nicht positionierbar**		nein
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Ziel ist kleiner als die Abschaltdifferenz.	
17	**Referenzpunktfahrt nicht möglich**		nein
	Ursache	Es wurde ein SSI-Geber angeschlossen.	

Nr.	Bedeutung		Diagnosealarm
18	**Schrittmaßfahrt relativ oder absolut nicht möglich**		nein
	Ursache	Das Schrittmaß ist ungültig.	
19	**Abschaltdifferenz nicht größer als ½ Zielbereich bei Schrittmaßnummer 255**		nein
	Ursache	Die Abschaltdifferenz für das Schrittmaß 255 ist kleiner als der halbe Zielbereich.	
20	**Fahrt in die vorgegebene Richtung unzulässig**		nein
	Ursache	Der Abstand zum Software-Endschalter ist zu gering.	

Datenfehler

Fehlerklasse 4: Datenfehler

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt. Die Datenfehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Datenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück/ Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
10	**Fehlerhafte Nullpunktverschiebung (FM 451)**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen, fehlerfreien Maschinendaten auf der Baugruppe vorhanden.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
29	**Unzulässige Bitcodierung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und einer Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereichs liegen.	

Nr.	Bedeutung		Diagnosealarm
36	**Fehlerhafte Umschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
37	**Fehlerhafte Abschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Die Abschaltdifferenz muss kleiner als die Umschaltdifferenz sein.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind entweder keine Maschinendaten vorhanden oder sie sind nicht aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Einer der Aufträge "Istwert setzen" und "Istwert setzen rückgängig" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	

Maschinendatenfehler

Fehlerklasse 5: Maschinendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst. Die Maschinendatenfehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Maschinendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	
6	**Falscher minimaler Kantenabstand (FM 451)**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben.	
7	**Falsches Maßsystem**		ja
	Ursache	Der Wert für das Maßsystem liegt außerhalb des zulässigen Bereichs von 1 bis 4 und 6.	
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben.	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 4.	
11	**Falscher Weg pro Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Inkremente pro Geberumdrehung** (siehe "Inkremente pro Geberumdrehung (Seite 2534)")		ja
14	**Falsche Anzahl Umdrehungen** (siehe "Anzahl Geberumdrehungen (Seite 2537)")		ja

Nr.	Bedeutung		Diagnosealarm
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereichs von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m (je nach Auflösung). Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Weggeber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art der Referenzpunktfahrt**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 2 und 3 angegeben.	
19	**Falsche Zählrichtung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung Telegrammfehler im Parameter-DB auf "FALSE" gesetzt. Beim verwendeten Geber ist die Überwachung von Fehlimpulsen nicht möglich. Deaktivieren Sie den Parameter MON_PULSE.	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw. -1000 m (je nach Auflösung).	
22	**Falscher Software-Endschalter Ende**		ja
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Linearachse: Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
23	**Falsche Maximalgeschwindigkeit**		ja
	Ursache	Die nicht aufgelisteten Daten im Parameter-DB müssen 0 sein.	
24	**Falscher Zielbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	
25	**Falsche Überwachungszeit**		ja
	Ursache	Der Wert für die Überwachungszeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 ms.	
26	**Falscher Stillstandsbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	

Nr.	Bedeutung		Diagnosealarm
127	**Falsche Stillstandsgeschwindigkeit**		ja
	Ursache	Der Wert für die Stillstandsgeschwindigkeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 µm/min.	
128	**Falsche Ansteuerart**		ja
	Ursache	Sie haben für die Ansteuerart einen Wert außerhalb des zulässigen Bereichs von 1 bis 4 angegeben.	
129	**Falsche Startgeschwindigkeit für die Referenzpunktfahrt**		ja
	Ursache	Sie haben als Startgeschwindigkeit weder 0 noch 1 angegeben.	
130	**Falsche Umschaltdifferenz in Richtung +**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
131	**Falsche Umschaltdifferenz in Richtung -**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
132	**Falsche Abschaltdifferenz in Richtung +**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz plus, kleiner ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
133	**Falsche Abschaltdifferenz in Richtung -**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz minus, kleiner ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben eine Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Geber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Geber und Rundachse: Der Geber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Geber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Schrittmaßtabellen-Fehler

Fehlerklasse 6: Schrittmaßtabellen-Fehler

Die Schrittmaßtabellen-Fehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Schrittmaßtabellen-Fehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß in der Schrittmaßtabelle zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück/Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	

Meldungen

Fehlerklasse 15: Meldungen

Die Meldungen führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Meldungen und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	
11	**Abstand zum Umschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
12	**Abstand zum Umkehrpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
14	**Abstand zum Abschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
15	**Abstand zum Zielbereichsanfang zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	

Diagnosefehler

Fehlerklasse 128: Diagnosefehler

Die folgende Tabelle zeigt die möglichen Diagnosefehler mit ihren Ursachen und Wirkungen sowie Abhilfemöglichkeiten.

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	• Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen • Sicherung auf der Baugruppe ist defekt • Unterspannung • Masse-Drahtbruch • Kurzschluss (z. B. am angeschlossenen Geber)	
	Wirkung	• Die Positionierung wird auf allen Kanälen abgebrochen • Abschalten der Ausgänge • Löschen der Synchronisation bei Inkrementalgebern, wenn die Hilfsspannung für Geberversorgung fehlt • Die Positionierbaugruppe ist nicht parametriert • Löschen der Startfreigabe	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (Wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
5	**Frontstecker fehlt (FM 451)**		ja
	Ursache	Fronstecker der Positionierbaugruppe ist nicht gesteckt	
	Wirkung	• Externe Hilfsspannung 24 V fehlt • Baugruppe ist nicht betriebsbereit	
	Behebung	Frontstecker auf die Positionierbaugruppe stecken	
51	**Zeitüberwachung angesprochen (Watchdog)**		ja
	Ursache	• Starke Störeinflüsse auf die Positionierbaugruppe • Fehler in der Positionierbaugruppe	
	Wirkung	• Baugruppe wird rückgesetzt • Abschalten alle Ausgänge • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind • Tauschen der Positionierbaugruppe	

Nr.	Bedeutung		Diagnosealarm
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale • Flankenfehler der Gebersignale • Maximale Eingangsfrequenz des Gebereingangs überschritten • Ausfall der Geberversorgung	
	Wirkung	• die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Synchronisation bei Inkrementalgebern • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden • Technische Daten der Baugruppe einhalten	
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen der Positionierbaugruppe und dem Absolutwert-geber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • Falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt • Monoflopzeit des Gebers größer als 64 µs	
	Wirkung	• Die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und Positionierbaugruppe	

Nr.	Bedeutung		Diagnosealarm
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben (Parametrierdialog) • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden	

6.2.6 FM 451 einsetzen

6.2.6.1 Anschlussbelegung der Frontstecker der FM 451

Beschreibung der Geberschnittstelle

Lage der D-SUB-Buchsen

Das folgende Bild zeigt die Einbaulage und die Bezeichnung der Buchsen auf der Baugruppe. An die D-SUB-Buchsen können Sie Inkrementalgeber oder Absolutwertgeber (SSI) anschließen.

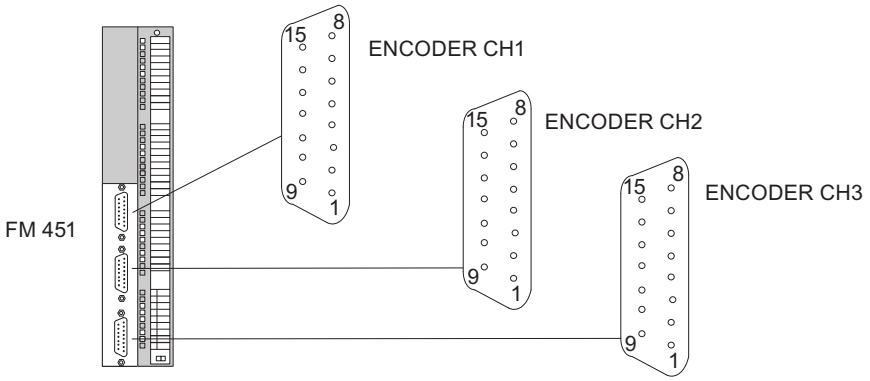

CH = Kanal

Belegung der Geberschnittstellen (Encoder CH1, CH2 und CH3)

Pin	Name	Inkrementalgeber (24 V)	Inkrementalgeber (5 V)	Absolutwertgeber
1	A*	Gebersignal A	---	---
2	CLS	---	---	SSI-Schiebetakt
3	/CLS	---	---	SSI-Schiebetakt invers
4	B*	Gebersignal B	---	---
5	DC 24 V	Geberversorgung	Geberversorgung	Geberversorgung
6	DC 5.2 V	---	Geberversorgung	Geberversorgung
7	M	Masse	Masse	Masse
8	N*	Nullmarkensignal	---	---
9	RE	P/M-schaltend [1)]	---	---
10	N	---	Nullmarkensignal	---
11	/N	---	Nullmarkensignal invers	---
12	/B	---	Gebersignal B invers	---
13	B	---	Gebersignal B	---
14	/A / /DAT	---	Gebersignal A invers	SSI-Daten invers
15	A / DAT	---	Gebersignal A	SSI-Daten

[1)] **Hinweis:** Wenn Sie einen Inkrementalgeber eines Fremdherstellers in Gegentaktschaltung (P/M-schaltend) anschließen, müssen Sie Folgendes beachten:

- P-schaltend: Verbinden Sie RE (9) mit Masse (7).
- M-schaltend: Verbinden Sie RE (9) mit +24 V (5).

Belegung des Frontsteckers

Frontstecker

Am 48-poligen Frontstecker schließen Sie an:

- die den Kanälen zugeordnete digitale Ausgänge und Eingänge
- die Versorgungsspannungen der Geber und der Digitalausgänge

Anschlussbelegung des Frontsteckers (X1)

Die folgende Tabelle zeigt die Anschlussbelegung des Frontsteckers der FM 451.

Anschluss	Name	Funktion	Inkrementalgeber	Absolutwertgeber
1		Belegt; enthält Kabelbrücke zur Erkennung des gesteckten Frontsteckers.		
2				
3	1L+	DC 24 V Hilfsspannung für die Geberversorgung		
4 bis 7		Nicht verwendet		
8	1I0	Kanal 1: Digitaleingang 0	Referenzpunktschalter	Nicht verwendet
9	1I1	Kanal 1: Digitaleingang 1	Umkehrschalter	Nicht verwendet
10	1I2	Kanal 1: Digitaleingang 2	Freigabeeingang	

Anschluss	Name	Funktion	Inkrementalgeber	Absolutwertgeber
11	1I3	Kanal 1: Digitaleingang 3	Eingang für "fliegendes Istwert setzen" / Längenmessung / Kantenerfassung / Externes Ereignis für Schrittmaßnummer 252	
12		Nicht verwendet		
13	2L+	DC 24 V Hilfsspannung für die Laststromversorgung		
14	2L+	Die beiden Klemmen sind auf der Baugruppe intern verbunden.		
15	2I0	Kanal 2: Digitaleingang 0	Referenzpunktschalter	Nicht verwendet
16	2I1	Kanal 2: Digitaleingang 1	Umkehrschalter	Nicht verwendet
17	2I2	Kanal 2: Digitaleingang 2	Freigabeeingang	
18	2I3	Kanal 2: Digitaleingang 3	Eingang für "fliegendes Istwert setzen" / Längenmessung / Kantenerfassung / Externes Ereignis für Schrittmaßnummer 252	
19	3I0	Kanal 3: Digitaleingang 0	Referenzpunktschalter	Nicht verwendet
20	3I1	Kanal 3: Digitaleingang 1	Umkehrschalter	Nicht verwendet
21	3I2	Kanal 3: Digitaleingang 2	Freigabeeingang	
22	3I3	Kanal 3: Digitaleingang 3	Eingang für "fliegendes Istwert setzen" / Längenmessung / Kantenerfassung / Externes Ereignis für Schrittmaßnummer 252	
23 bis 24		Nicht verwendet		
25	3L+	DC 24 V Hilfsspannung für die Laststromversorgung		
26	3L+	Die beiden Klemmen sind auf der Baugruppe intern verbunden.		
27	1Q0	Kanal 1: Digitalausgang 0		
28	1Q1	Kanal 1: Digitalausgang 1		
29	1Q2	Kanal 1: Digitalausgang 2		
30	1Q3	Kanal 1: Digitalausgang 3		
31	2Q0	Kanal 2: Digitalausgang 0		
32	2Q1	Kanal 2: Digitalausgang 1		
33	2Q2	Kanal 2: Digitalausgang 2		
34	2Q3	Kanal 2: Digitalausgang 3		
35 bis 36		Nicht verwendet		
37	4L+	DC 24 V Hilfsspannung für die Laststromversorgung		
38	4L+	Die beiden Klemmen sind auf der Baugruppe intern verbunden.		
39	3Q0	Kanal 3: Digitalausgang 0		
40	3Q1	Kanal 3: Digitalausgang 1		
41	3Q2	Kanal 3: Digitalausgang 2		
42	3Q3	Kanal 3: Digitalausgang 3		
43 bis 47		Nicht verwendet		
48	M	Masse der Hilfsspannungen 1L+, 2L+, 3L+ und 4L+		

Weitere Informationen zu Aufbau und Verdrahtung finden Sie in der Betriebsanleitung (http://support.automation.siemens.com/WW/view/de/1118070) der Positionierbaugruppe FM 451.

Hilfsspannung für die Geberversorgung (1L+, M)

Hier schließen Sie eine DC 24 V-Hilfsspannung für die Geberversorgung an. Das Bezugspotential dieser Versorgung (M) ist in der FM 451 mit der Masse der Laststromversorgung (M) verbunden.

Die DC 24 V-Hilfsspannung für die Geberversorgung wird auf Unterspannung und Drahtbruch überwacht.

An der Geberschnittstelle (D-SUB-Buchsen Encoder CH1, CH2 und CH3) werden DC 24 V und DC 5,2 V für die unterschiedlichen Geberarten zur Verfügung gestellt.

⚠ VORSICHT
Achten Sie auf die richtige Polung der DC 24 V-Hilfsspannung für die Geberversorgung (1L +, M). Wenn Sie die DC 24 V-Hilfsspannung für die Geberversorgung verpolt anschließen, dann wird die Baugruppe beschädigt und muss ausgetauscht werden.

Hilfsspannung für die Laststromversorgung (2L+, 3L+, 4L+, M)

An die Klemmen 2L+, 3L+, 4L+ und M müssen Sie die DC 24 V-Hilfsspannungen für die Laststromversorgung der Digitalausgänge anschließen.

⚠ VORSICHT
Achten Sie auf die richtige Polung der DC 24 V-Hilfsspannung für die Laststromversorgung (2L+, 3L+, 4L+, M). Wenn Sie die DC 24 V-Hilfsspannung für die Laststromversorgung verpolt anschließen, dann wird die Baugruppe beschädigt und muss ausgetauscht werden.

Verdrahtungshinweis für DC 24 V

Beachten Sie bei der Verdrahtung, dass alle Klemmen 1L+ bis 4L+ verschaltet sein müssen, damit die Baugruppe fehlerfrei läuft.

Beginnen Sie mit der Verdrahtung der DC 24 V an Klemme 38 und verbinden Sie die Hilfsspannung der Klemme 37 mit der Klemme 26. Mit den weiteren Klemmen verfahren Sie analog.

Alternativ können Sie auch an die Anschlüsse 1L+ (Geber), 2L+ (Kanal 1), 3L+ (Kanal 2) und 4L+ (Kanal 3) getrennte Spannungsversorgungen anschließen. Achten Sie aber darauf, dass alle Spannungsversorgungen ein gemeinsames Massepotential besitzen. In diesem Fall bleibt nach einem Spannungsausfall an den Hilfsspannungen für die Laststromversorgung die Synchronisation der Achsen erhalten.

Laststromversorgungen

Die DC-Laststromversorgung muss folgenden Anforderungen genügen:

Als Laststromversorgung darf nur vom Netz sicher getrennte Kleinspannung DC ≤ 60 V verwendet werden. Die sichere Trennung kann realisiert sein nach den Anforderungen u. a. in

- VDE 0100 Teil 410 / HD 384-4-41 / IEC 364-4-41
  (als Funktionskleinspannung mit sicherer Trennung) bzw.
- VDE 0805 / EN 60950 / IEC 950
  (als Sicherheitskleinspannung SELV) bzw.
- VDE 0106 Teil 101.

12 Digitaleingänge (1I0 bis 3I3)

Die FM 451 verfügt je Kanal über 4 Digitaleingänge.

An die 12 Digitaleingänge können Sie prellfreie Schalter (24 V P-schaltend) oder berührungslose Sensoren (2- oder 3-Draht Näherungsschalter) anschließen.

Die Digitaleingänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgebunden zur Masse der Geberversorgung.

Der Zustand jedes Eingangs ist an der zugehörigen LED ablesbar.

12 Digitalausgänge (1Q0 bis 3Q3)

Die FM 451 verfügt je Kanal über 4 Digitalausgänge.

Mit den Digitalausgängen wird das Leistungsteil angesteuert. Die Funktion der Digitalausgänge ist abhängig von der Ansteuerart (Seite 2613).

Die Digitalausgänge werden nicht auf Kurzschluss oder Drahtbruch überwacht und sind potentialgebunden zur Masse der Geberversorgung.

Der Zustand jedes Ausgangs ist an der zugehörigen LED ablesbar.

6.2.6.2 Funktionen der FM 451

Funktionsübersicht

Mit der Vorgabe einer Betriebsart, eines Auftrags oder Funktionsschalters wählen Sie, mit welcher Funktionalität Sie die Baugruppe betreiben wollen. Die folgende Tabelle zeigt einen Überblick über die Funktionalität der Baugruppe.

Funktionalität	Beschreibung
Maschinendaten schreiben und lesen (Seite 2569)	Mit den Maschinendaten passen Sie die Positionierbaugruppe an die Achse und den Geber an.
Schrittmaßtabellen schreiben und lesen (Seite 2572)	Mit den Schrittmaßen geben Sie Ziele vor, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ / absolut" angesteuert werden.

Funktionalität	Beschreibung
Ende eines Positioniervorgangs (Seite 2574)	Das Ende eines Positioniervorgangs wird auf drei verschiedene Arten erreicht: • Zieleinlauf • Absteuern • Abbrechen
Betriebsart "Tippen" (Seite 2580)	In der Betriebsart "Tippen" bewegen Sie den Antrieb auf Tastendruck in eine Richtung.
Betriebsart "Referenzpunktfahrt" (Seite 2583)	Mit der Betriebsart "Referenzpunktfahrt" synchronisieren Sie die Achse durch ein wiederkehrendes externes Ereignis.
Betriebsart "Schrittmaßfahrt" (Seite 2590)	In der Betriebsart "Schrittmaßfahrt" kann die Positionierbaugruppe den Antrieb • auf absolute Ziele bewegen, • relativ um ein Wegstück in eine vorgegebene Richtung bewegen.
Auftrag "Istwert setzen" / Fliegendes Istwert setzen / "Istwert setzen rückgängig" (Seite 2596)	Mit dem Auftrag "Istwert setzen" bzw. "Fliegendes Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Mit dem Auftrag "Istwert setzen rückgängig" setzen Sie alle Arbeitsbereichverschiebungen, die durch "Istwert setzen" erzeugt wurden, wieder zurück.
Auftrag "Nullpunktverschiebung" (Seite 2598)	Mit dem Auftrag "Nullpunktverschiebung" verschieben Sie den Nullpunkt im Koordinatensystem um den eingegebenen Wert.
Auftrag "Bezugspunkt setzen" (Seite 2601)	Mit dem Auftrag "Bezugspunkt setzen" synchronisieren Sie die Achse.
Funktionsschalter "Längenmessung" / "Kantenerfassung" (Seite 2602)	Mit "Längenmessung" und "Kantenerfassung" können Sie die Länge eines Werkstücks ermitteln.
Funktionsschalter "Schleifenfahrt" (Seite 2605)	Mit "Schleifenfahrt" legen Sie die Richtung fest, in die ein Ziel mit Kraftschluss angefahren wird.
Funktionsschalter "Freigabeeingang" (Seite 2607)	Der Freigabeeingang ist ein externer Eingang, mit dem eine Positionierung durch ein externes Ereignis freigegeben werden kann.
Auftrag "Positionsdaten lesen" (Seite 2608)	Mit dem Auftrag "Positionsdaten lesen" können Sie Schrittmaß, Restweg und Geschwindigkeit zum aktuellen Zeitpunkt lesen.
Auftrag "Geberdaten lesen" (Seite 2608)	Mit dem Auftrag "Geberdaten lesen" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.
Rückmeldesignale für die Positionierung (Seite 2609)	Mit den "Rückmeldesignalen für die Positionierung" werden Sie über den aktuellen Zustand der Positionierung informiert.
Rückmeldesignale für die Diagnose (Seite 2610)	Mit den "Rückmeldesignalen für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Maschinendaten schreiben und lesen

Maschinendaten schreiben, aktivieren und lesen

Mit den Maschinendaten passen Sie die Positionierbaugruppe an die Achse und den Geber an.

Über das Anwenderprogramm können Sie die Parameter während des Betriebs ändern und auslesen.

Alle Parameter sind im Parameter-DB abgelegt:

- Maschinendaten liegen im Parameter-DB auf den Adressen 4.0 bis 116.0.

Die Nummer des Parameter-DB müssen Sie in den jeweils zugehörigen Kanal-DB eintragen.

Die Parameter können Sie mit dem DB-Editor eingeben oder auch komfortabel in den Dialogen "Antrieb", "Achse" und "Geber" eintragen und mit der Funktion "Export" in den Parameter-DB schreiben.

Die Parameter in einem bereits vorhandenen Parameter-DB können Sie mit der Funktion "Import" in die Dialoge importieren.

Erstparametrierung

Falls der Kanal noch keine Maschinendaten enthält, gehen Sie bei einer Erstparametrierung ohne die Parametrierdialoge folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein und speichern diesen ab.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie das folgende Anstoßbit im Kanal-DB für den Auftrag:
   - Maschinendaten schreiben (MDWR_EN = 1).
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Maschinendaten ändern

Um Maschinendaten per Anwenderprogramm zu ändern, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   - Maschinendaten schreiben (MDWR_EN = 1)
   - Maschinendaten aktivieren (MD_EN = 1)
3. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Wenn Sie die Anstoßbits für diese Aufträge gleichzeitig setzen, sorgt die Anweisung ABS_CTRL (bzw. ABS_CTRL_451) dafür, dass die Aufträge in der richtigen Reihenfolge abgearbeitet werden.

Ansonsten ändern Sie die Maschinendaten immer in folgender Reihenfolge:

- Maschinendaten schreiben
- Maschinendaten aktivieren

Maschinendaten lesen

Um die aktuellen Maschinendaten aus einem Kanal zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie das folgende Anstoßbit im Kanal-DB:
   – Maschinendaten lesen (MDRD_EN = 1)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Damit werden die aktuellen Maschinendaten im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
36.5	MDRD_EN	BOOL	FALSE	1 = Maschinendaten lesen

Hinweis

Wenn synchronisationsrelevante Parameter verändert werden, dann werden beim Aktivieren der Maschinendaten folgende Aktionen von der Baugruppe für den betroffenen Kanal durchgeführt:

- Die Synchronisation wird gelöscht.
- Die Funktionsschalter und die Nullpunktverschiebung werden zurückgesetzt.
- Alle bisherigen Maschinendaten und Schrittmaßtabellen werden ungültig.

Synchronisationsrelevante Parameter sind:

- Achsart
- Ende der Rundachse
- Signalart
- Weg pro Geberumdrehung
- Inkremente pro Geberumdrehung
- Anzahl Umdrehungen
- Referenzpunktkoordinate
- Absolutwertgeberjustage
- Art der Referenzpunktfahrt
- Zählrichtung

Schrittmaßtabellen schreiben und lesen

Schrittmaßtabellen schreiben und lesen

Mit den Schrittmaßen geben Sie Ziele vor, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ / absolut" angesteuert werden.

Über das Anwenderprogramm können Sie die Parameter während des Betriebs ändern und auslesen.

Alle Parameter sind im Parameter-DB abgelegt:

- Schrittmaßtabellen liegen im Parameter-DB auf den Adressen 120.0 bis 516.0.

Die Nummer des Parameter-DBs müssen Sie in den jeweils zugehörigen Kanal-DB eintragen.

Die Parameter können Sie mit dem DB-Editor eingeben oder auch komfortabel in den Dialog "Schrittmaße" eintragen und mit der Funktion "Export" in den Parameter-DB schreiben.

Die Parameter in einem bereits vorhandenen Parameter-DB können Sie mit der Funktion "Import" in den Dialog importieren.

Erstparametrierung

Falls der Kanal noch keine Schrittmaßtabellen enthält, gehen Sie bei einer Erstparametrierung ohne den Parametrierdialog folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein und speichern diesen ab.
2. Laden Sie den Parameter-DB in die CPU.
3. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   - Schrittmaßtabelle 1 schreiben (TRGL1WR_EN = 1) und / oder Schrittmaßtabelle 2 schreiben (TRGL2WR_EN = 1)
4. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Schrittmaßtabellen ändern

Um Schrittmaßtabellen per Anwenderprogramm zu ändern, gehen Sie folgendermaßen vor:

1. Tragen Sie die neuen Werte in den Parameter-DB ein.
2. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   - Schrittmaßtabelle 1 schreiben (TRGL1WR_EN = 1) und / oder Schrittmaßtabelle 2 schreiben (TRGL2WR_EN = 1)
3. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Schrittmaßtabellen lesen

Um die Schrittmaßtabellen aus einem Kanal zu lesen, gehen Sie folgendermaßen vor:

1. Setzen Sie die Anstoßbits im Kanal-DB für die Aufträge:
   – Schrittmaßtabelle 1 lesen (TRGL1RD_EN = 1) und / oder Schrittmaßtabelle 2 lesen (TRGL2RD_EN = 1)
2. Rufen Sie im zyklischen Anwenderprogramm die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Damit werden die Schrittmaßtabellen im Parameter-DB auf der CPU abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (1 ... 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (51 ... 100) schreiben
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (1 ... 50) lesen
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (51 ... 100) lesen

Hinweis

Wenn synchronisationsrelevante Parameter verändert werden, dann werden beim Aktivieren der Maschinendaten folgende Aktionen von der Baugruppe für den betroffenen Kanal durchgeführt:

- Die Synchronisation wird gelöscht.
- Die Funktionsschalter und die Nullpunktverschiebung werden zurückgesetzt.
- Alle bisherigen Maschinendaten und Schrittmaßtabellen werden ungültig.

Synchronisationsrelevante Parameter sind:

- Achsart
- Ende der Rundachse
- Signalart
- Weg pro Geberumdrehung
- Inkremente pro Geberumdrehung
- Anzahl Umdrehungen
- Referenzpunktkoordinate
- Absolutwertgeberjustage
- Art der Referenzpunktfahrt
- Zählrichtung

Ende eines Positioniervorgangs

Definition

Das Ende eines Positioniervorgangs wird durch das Rückmeldesignal WORKING = 0 angezeigt. Es wird auf drei verschiedene Arten erreicht:

- Zieleinlauf
- Absteuern
- Abbrechen

Überwachungen

Zum Ende eines Positioniervorgangs hin sind folgende Überwachungen aktiv:

- Überwachungszeit
  Die Überwachungszeit wird am Abschaltpunkt zum letzten Mal nachgetriggert und verliert ihre Gültigkeit mit dem Ende des Positioniervorgangs.
  Innerhalb dieser Zeit muss das Ende des Positioniervorgangs erreicht werden. Sonst werden die Ausgänge abgeschaltet und der Betriebsfehler "Fehler beim Zieleinlauf" (Fehlernummer 5) gemeldet.
- Überwachung des Zielbereichs
  Die FM x51 legt um jedes Ziel einen symmetrischen Bereich und definiert damit die Positioniergenauigkeit Ihrer Anwendung. Innerhalb dieses Bereichs muss die Achse bei einem Zieleinlauf zum Stillstand kommen. Eine Wertvorgabe 0 schaltet die Toleranz beim Zieleinlauf ab.
- Überwachung der Stillstandsgeschwindigkeit
  Über die Stillstandsgeschwindigkeit lässt sich feststellen, ob der Antrieb innerhalb des Zielbereichs zum Stillstand kommt. Nach dem Erreichen des Abschaltpunktes überprüft die FM x51, ob die Istgeschwindigkeit des Antriebs kleiner ist als die Stillstandsgeschwindigkeit.
  Die Stillstandsgeschwindigkeit muss innerhalb des Zielbereichs unterschritten werden. Sonst meldet die FM x51 den Betriebsfehler "Zielbereich überfahren" (Fehlernummer 10).
  Das Unterschreiten der Stillstandsgeschwindigkeit wird nur einmal je Zieleinlauf überwacht. Beachten Sie hierbei, dass die Stillstandsgeschwindigkeit für die Geschwindigkeitsermittlung der Baugruppe kurzzeitig unterschritten werden kann, wenn die Achse sich mit sehr kleiner Positioniergeschwindigkeit bewegt (weniger als 2 Impulse pro 8 ms).
- Überwachung des Stillstandsbereichs
  Nach dem Ende eines Positioniervorgangs wird überwacht, ob der Antrieb auf einer angefahrenen Zielposition stehen bleibt oder von dieser wegdriftet.
  Der Stillstandsbereich wird überwacht,
  - nachdem die FM x51 das Rückmeldesignal "PEH" gemeldet hat,
  - wenn die Überwachungszeit überschritten wird,
  - wenn die Stillstandsgeschwindigkeit unterschritten wird.

  Wenn der Stillstandsbereich ohne einen gültigen Fahrauftrag verlassen wird, meldet die FM x51 den Betriebsfehler "Stillstandsbereich verlassen" (Fehlernummer 6).

Zieleinlauf

Der Zieleinlauf in den Betriebsarten "Schrittmaßfahrt absolut" und "Schrittmaßfahrt relativ" beginnt mit dem Erreichen des Abschaltpunktes. Ab diesem Punkt wird der Antrieb abgeschaltet, und die FM x51 übernimmt Überwachungsfunktionen.

Je nachdem, welche Überwachungen Sie parametriert haben, ergeben sich verschiedene Fälle für die Erzeugung des Rückmeldesignals "PEH (POS_RCD)". Der Positioniervorgang wird abgebrochen, wenn kein Rückmeldesignal "PEH (POS_RCD)" erzeugt wird.

Fall 1: Sie haben parametriert:

- Zielbereich (TRG_RANGE) > 0
- Stillstandsgeschwindigkeit (ZSPEED_L) > 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn die Stillstandsgeschwindigkeit unterschritten und der Zielbereich erreicht wird. Welche Bedingung zuerst erfüllt wird, spielt hierbei keine Rolle.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit den Zielbereich nicht erreicht bzw. der Zielbereich überfahren wird, ohne die Stillstandsgeschwindigkeit zu unterschreiten.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich > 0 und Stillstandsgeschwindigkeit > 0.

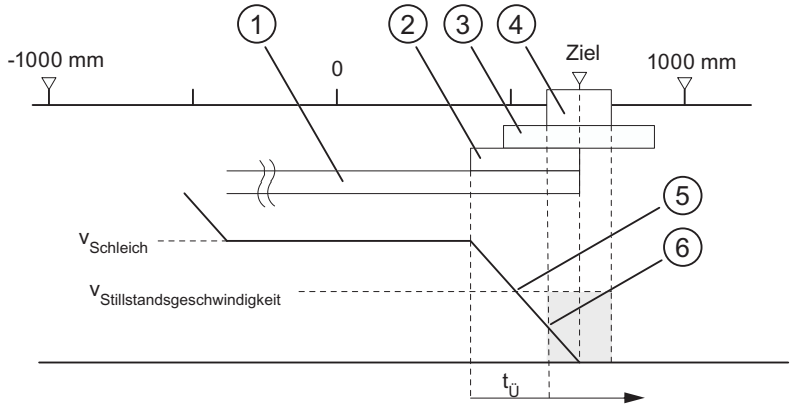

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Zielbereich
⑤ Stillstandsgeschwindigkeit erreicht
⑥ Zielbereich mit V_{Still} erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 2: Sie haben parametriert:

- Zielbereich (TRG_RANGE) > 0
- Stillstandsgeschwindigkeit (ZSPEED_L) = 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn der Zielbereich erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit den Zielbereich nicht erreicht.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich > 0 und Stillstandsgeschwindigkeit = 0.

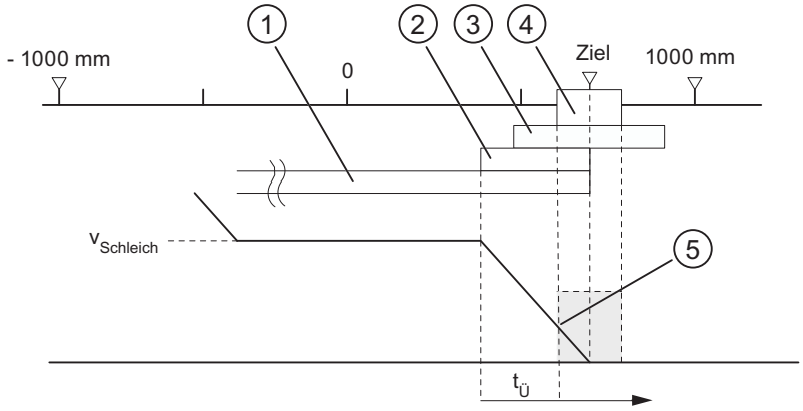

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Zielbereich
⑤ Zielbereich mit V_{Still} erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 3: Sie haben parametriert:

- Zielbereich (TRG_RANGE) = 0
- Stillstandsgeschwindigkeit (ZSPEED_L) > 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn die Stillstandsgeschwindigkeit unterschritten und dann das Ziel erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit das Ziel nicht erreicht bzw. der Zielbereich überfahren wird, ohne die Stillstandsgeschwindigkeit zu unterschreiten.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich = 0 und Stillstandsgeschwindigkeit > 0.

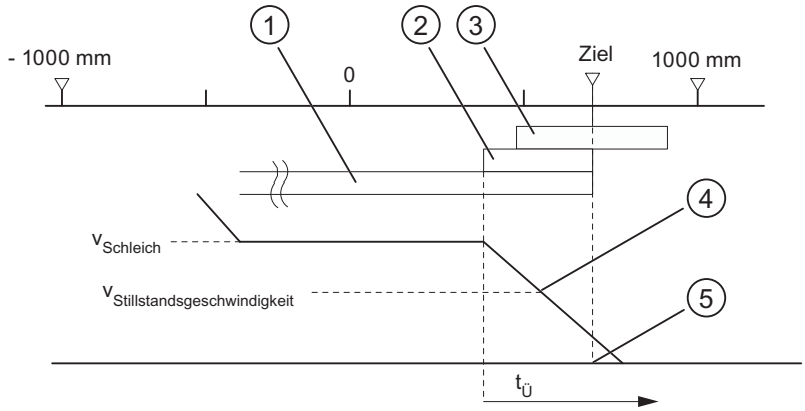

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Stillstandsbereich
④ Stillstandsgeschwindigkeit erreicht
⑤ Ziel erreicht: PEH wird gesetzt
$t_Ü$ Überwachungszeit

Fall 4: Sie haben parametriert:

- Zielbereich (TRG_RANGE) = 0
- Stillstandsgeschwindigkeit (ZSPEED_L) = 0
- Überwachungszeit (MON_TIME) > 0

PEH wird generiert, wenn das Ziel erreicht wird.

PEH wird nicht generiert, wenn der Istwert innerhalb der Überwachungszeit das Ziel nicht erreicht.

Das folgende Bild zeigt den Zieleinlauf einer Schrittmaßfahrt mit Zielbereich = 0 und Stillstandsgeschwindigkeit = 0.

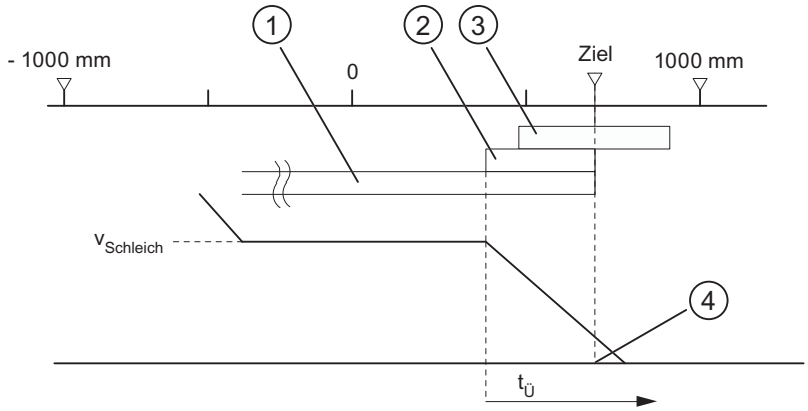

① Umschaltdifferenz plus

② Abschaltdifferenz plus

③ Stillstandsbereich

④ Ziel erreicht: PEH wird gesetzt

$t_Ü$ Überwachungszeit

Fall 5: Sie haben parametriert:

- Zielbereich (TRG_RANGE) ≥ 0
- Stillstandsgeschwindigkeit (ZSPEED_L) ≥ 0
- Überwachungszeit (MON_TIME) = 0

Wenn in diesem Fall der Antrieb vor dem Zielbereich zum Stillstand kommt, wird das Ende des Positioniervorgangs nicht erkannt. PEH wird nicht generiert und das Rückmeldesignal WORKING bleibt gesetzt. Sie können den Positioniervorgang nur durch Löschen der Antriebsfreigabe abbrechen (DRV_EN = 0).

Absteuern ohne vorgegebenes Ziel

Absteuern bedeutet: Der Positioniervorgang wird gezielt unter Einhaltung der Differenzen von Eilgang über Schleichgang beendet.

Der Positioniervorgang wird abgesteuert, wenn

- die FM x51 ein STOP-Signal empfängt (STOP = 1).
- die Betriebsarten "Tippen" und "Referenzpunktfahrt" beendet werden.
- Bedienfehler oder der Betriebsfehler "Ziel überfahren" (Fehlernummer 9) auftreten.

Das Rückmeldesignal "PEH (POS_RCD)" wird nicht gesetzt. Die Abläufe sind analog zum Zieleinlauf.

Das folgende Bild zeigt das Absteuern eines Positioniervorgangs.

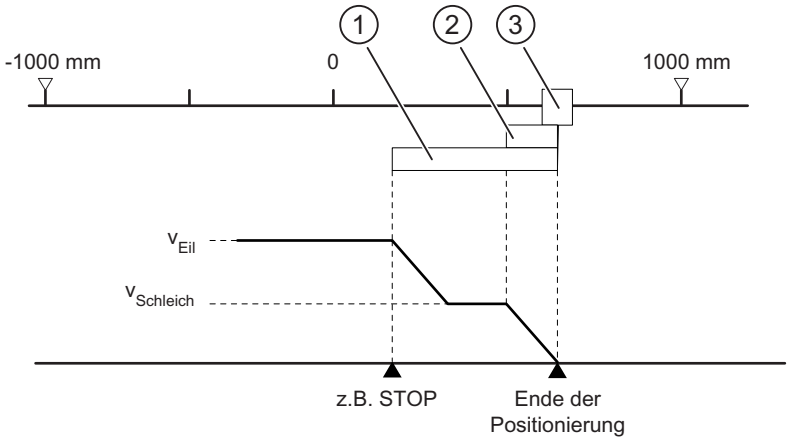

① Umschaltdifferenz plus
② Abschaltdifferenz plus
③ Zielbereich

Abbrechen

Abbrechen bedeutet: Der Positioniervorgang wird sofort ohne Verwendung der Umschalt- und Abschaltdifferenz von Eilgang bzw. Schleichgang nach Stillstand beendet. Dazu werden alle relevanten Ausgänge der jeweiligen Ansteuerart sofort ausgeschaltet und außerdem wird:

- Schrittmaß = Istwert
- Restweg = Null

Der Positioniervorgang wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0).
- die CPU nach STOP geht.
- Diagnosefehler oder alle Betriebsfehler außer dem Betriebsfehler "Ziel überfahren" (Fehlernummer 9) auftreten.

Das Rückmeldesignal "PEH (POS_RCD)" wird bei der Betriebsart "Schrittmaßfahrt" nicht gesetzt.

Wenn die Stillstandsgeschwindigkeit parametriert ist, wird die Stillstandsüberwachung aktiv, sobald diese unterschritten ist. Falls die Stillstandsgeschwindigkeit nicht parametriert ist, wird die Stillstandsüberwachung mit dem Abschalten der Ausgänge aktiv.

Das folgende Bild zeigt das Abbrechen eines Positioniervorgangs.

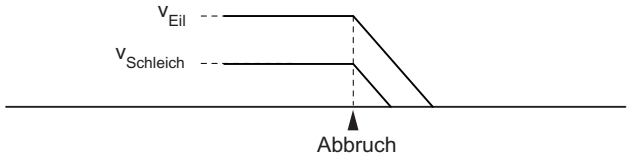

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	TRG_RANGE	DINT	L#1000	Zielbereich
80.0	MON_TIME	DINT	L#2000	Überwachungszeit
84.0	ZSPEED_R	DINT	L#1000	Stillstandsbereich
88.0	ZSPEED_L	DINT	L#30000	Stillstandsgeschwindigkeit

Rückmeldesignale im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht

Betriebsart "Tippen"

Definition

In der Betriebsart "Tippen" bewegen Sie den Antrieb auf Tastendruck in eine Richtung. Für beide Richtungen (plus und minus) müssen Sie je einen Taster installieren. Die Betriebsart "Tippen" können Sie sowohl für eine synchronisierte als auch für eine nicht synchronisierte Achse verwenden.

Bei nicht synchronisierter Achse ist Tippen ein Positionieren in die vorgegebene Richtung.

Bei synchronisierter Achse (Linearachse) ist Tippen ein Positionieren auf die Software-Endschalter.

Voraussetzung

Die Achse muss parametriert sein.

Ablauf der Betriebsart "Tippen"

1. Setzen Sie das Steuersignal für die Betriebsart "Tippen" (MODE_IN = 1).
2. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
3. Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie für den entsprechenden Kanal den Freigabeeingang.
4. Tragen Sie die Startgeschwindigkeit ein.
   - Eilgang (MODE_TYPE = 1)
   - Schleichgang (MODE_TYPE = 0)
5. Setzen Sie das Steuersignal für die Fahrtrichtung plus oder minus (DIR_P = 1 oder DIR_M = 1).
6. Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Tippen".

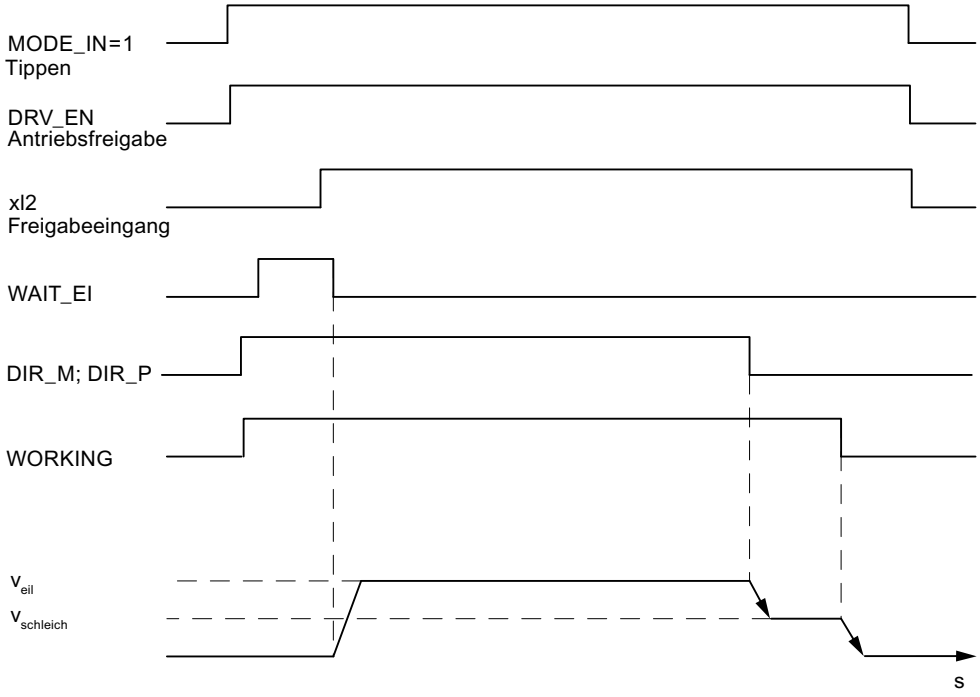

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	1 = Tippen
17.0	MODE_TYPE	BYTE	B#16#0	1 = Eilgang 0 = Schleichgang
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Absteuern des Tippens

Die Betriebsart "Tippen" wird abgesteuert, wenn

- Sie den Taster, mit dem Sie "Tippen", wieder öffnen (DIR_M = 0 bzw. DIR_P = 0),
- die FM 451 ein STOP-Signal empfängt (STOP = 1),
- der Istwert bei einer synchronisierten Linearachse die Grenze des Arbeitsbereichs erreicht. Eine Weiterfahrt ist nur in der entgegengesetzten Richtung möglich.

Nach dem Absteuern der Fahrt ist eine Weiterfahrt in einer beliebigen Richtung möglich.

Abbrechen des Tippens

Die Betriebsart "Tippen" wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0),
- eine Verfahrbereichsgrenze bei einer Linearachse überfahren wurde.

Überwachungen

Bei der Betriebsart "Tippen" sind am Ende der Positionierung folgende Überwachungen nicht aktiv:

- Überwachung des Stillstandsbereichs
- Überwachung des Zielbereichs
- Überwachung der Stillstandsgeschwindigkeit

Tippen führt nicht zu einer ordnungsgemäßen Positionierung, bei der am Ende das Signal "Position erreicht" gesetzt wird.

Arbeitsbereichsgrenze bei einer Linearachse

Die Grenzen für die Betriebsart "Tippen" unterscheiden sich zwischen einer synchronisierten und einer nicht synchronisierten Achse.

Die folgende Tabelle zeigt "Tippen" bei synchronisierter und nicht synchronisierter Achse.

Achse ist nicht synchronisiert	Achse ist synchronisiert
Wenn beim Tippen die Verfahrbereichsgrenze überfahren wird, • ist die Istwertanzeige nicht mehr gültig. • wird der Positioniervorgang abgebrochen. 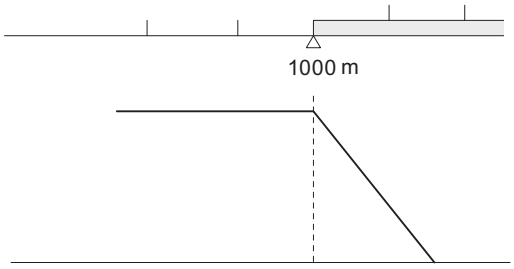 Die Istwertanzeige ist nicht mehr gültig	Tippen ist ein Positionieren auf Ziele, die um den ganzen Zielbereich von den Software-Endschaltern entfernt liegen. Die Arbeitsbereichsgrenzen ermitteln sich aus • **SEE - ½ Zielbereich** für das Ende der Linearachse in Richtung plus • **SEE + ½ Zielbereich** für das Ende der Linearachse in Richtung minus Wenn Sie nicht vorher den Taster öffnen, steuert die FM 451 auf einen Zielpunkt ab, der um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegt. Alle Bereiche, die zum korrekten Absteuern notwendig sind, werden von der FM 451 um diesen Zielpunkt gesetzt. 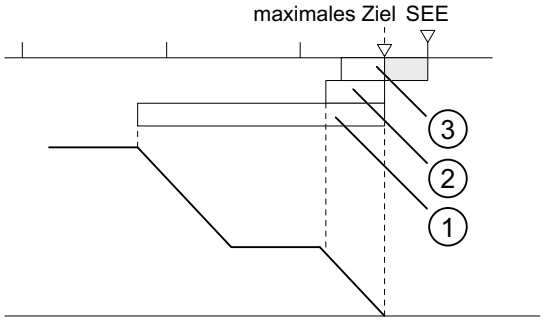 = Teil des Arbeitsbereichs, in dem keine Zielposition liegen darf ① = Umschaltdifferenz plus ② = Abschaltdifferenz plus ③ = ½ Zielbereich

Betriebsart "Referenzpunktfahrt"

Definition

Mit der Betriebsart "Referenzpunktfahrt" synchronisieren Sie die Achse durch ein wiederkehrendes externes Ereignis.

Voraussetzungen

- Ein Inkrementalgeber mit Nullmarke.
- Die Achse muss parametriert sein:

Anschlüsse	Kanal 1	Kanal 2	Kanal 3
Referenzpunktschalter	Digitaleingang 1I0	Digitaleingang 2I0	Digitaleingang 3I0
	Der Referenzpunktschalter soll so beschaffen sein, dass der Antrieb im Bereich des Schalters sicher vom Eilgang auf Schleichgang abbremsen kann.		
Umkehrschalter	Digitaleingang 1I1	Digitaleingang 2I1	Digitaleingang 3I1
	Achten Sie bei der Parametrierung darauf, dass der Start der Referenzpunktfahrt in Richtung Umkehrschalter parametriert ist. Nur so können Sie sicherstellen, dass der Referenzpunktschalter auch immer gefunden wird.		
Freigabeeingang	Digitaleingang 1I2	Digitaleingang 2I2	Digitaleingang 3I2

Ablauf der Betriebsart "Referenzpunktfahrt"

Die Taster für die positive und negative Verfahrrichtung müssen je Kanal installiert sein.

1. Tragen Sie den Wert der Referenzpunktkoordinate in den Parameter-DB ein (REFPT).
2. Tragen Sie die Art der "Referenzpunktfahrt" in den Parameter-DB ein.
   Hierbei haben Sie folgende Möglichkeiten:

Start in Richtung ...	Zur Synchronisation führt ...	
plus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **plus**	REFPT_TYPE = 0
plus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **minus**	REFPT_TYPE = 1
minus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **plus**	REFPT_TYPE = 2
minus	die erste Nullmarke nach Verlassen des Referenzpunktschalters in Richtung **minus**	REFPT_TYPE = 3

3. Tragen Sie die Startgeschwindigkeit ein:
   - Eilgang (REFPT_SPD = 0)
   - Schleichgang (REFPT_SPD = 1)
4. Schreiben und aktivieren Sie die Maschinendaten.
5. Setzen Sie das Steuersignal für die Betriebsart "Referenzpunktfahrt" (MODE_IN = 3).
6. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
7. Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten"(EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.
8. Setzen Sie das Steuersignal für die Fahrtrichtung plus bzw. minus oder das Startsignal (DIR_P = 1, DIR_M = 1 oder START = 1)
9. Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.

Die folgende Tabelle zeigt die Startkommandos für eine Referenzpunktfahrt.

Startbefehl	Aufgabe	Bemerkung
DIR_P	Der Antrieb startet in Richtung positiver Werte. Er bewegt sich also in Richtung Verfahrbereich Ende.	Wenn in den Maschinendaten eine negative Richtung eingetragen ist, meldet die FM 451 einen Bedienfehler. Es wird keine Referenzpunktfahrt durchgeführt.
DIR_M	Der Antrieb startet in Richtung negativer Werte. Er bewegt sich also in Richtung Verfahrbereich Anfang.	Wenn in den Maschinendaten eine positive Richtung eingetragen ist, meldet die FM 451 einen Bedienfehler. Es wird keine Referenzpunktfahrt durchgeführt.
START	Der Antrieb startet in die Richtung, die in den Maschinendaten eingegeben wurde.	

Hinweis

Für die Rundachse gilt: Die Reproduzierbarkeit des Referenzpunktes ist nur dann sichergestellt, wenn zwischen dem Wert **Ende der Rundachse** und dem Wert **Weg pro Geberumdrehung** ein ganzzahliges Verhältnis besteht.

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Referenzpunktfahrt".

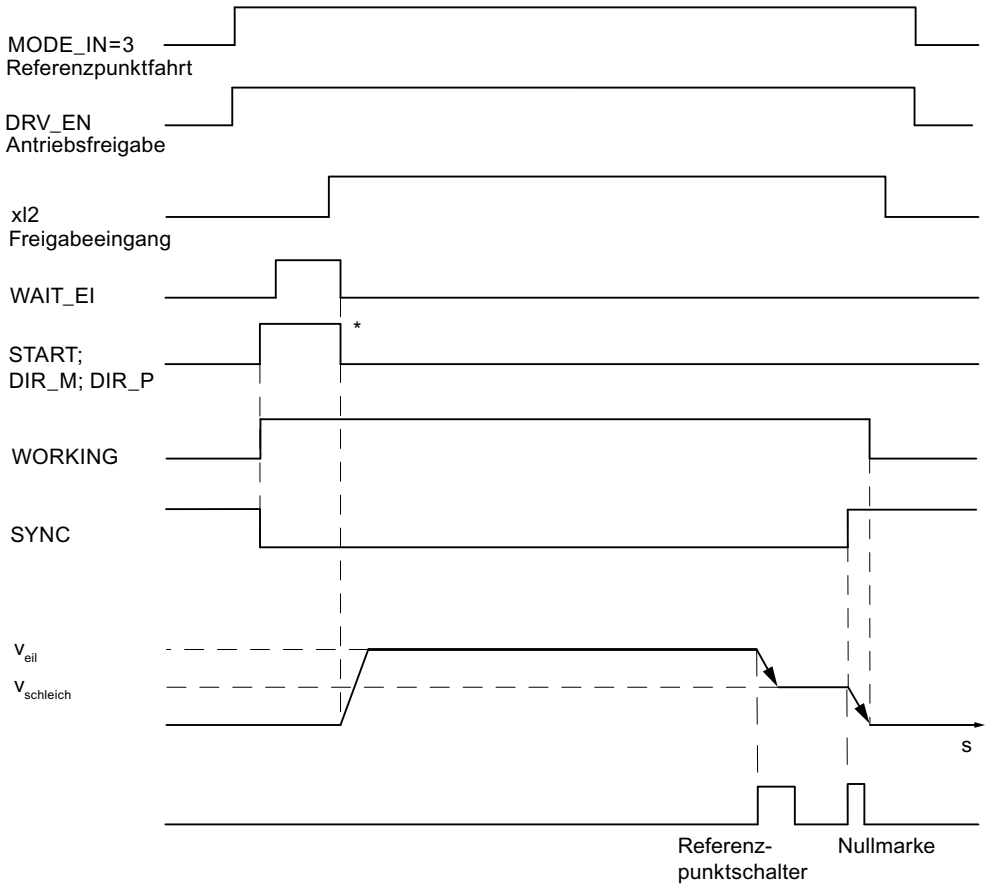

* Die Startsignale werden von der Anweisung ABS_CTRL bzw. ABS_CTRL_451 zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positioniervorgang starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	3 = Referenzpunktfahrt
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positioniervorgang läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	Referenzpunktkoordinate
52.0	REFPT_TYPE	DINT	L#0	Art der Referenzpunktfahrt
99.0	REFPT_SPD	BOOL	TRUE	Startgeschwindigkeit für Referenzpunktfahrt 0 = Eilgang 1 = Schleichgang

Auswirkungen der Betriebsart

- Mit Start der Fahrt wird die Synchronisation weggenommen.
- Die Istposition wird auf den Wert der Referenzpunktkoordinate gesetzt, wenn das Rückmeldesignal SYNC gesetzt wird.
- Der Arbeitsbereich wird auf der Achse festgelegt.
- Die einzelnen Punkte innerhalb des Arbeitsbereichs behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.

Abbrechen der Referenzpunktfahrt

Die Betriebsart "Referenzpunktfahrt" wird abgebrochen, wenn

- das Signal Antriebsfreigabe gelöscht wird (DRV_EN = 0),
- eine Verfahrbereichsgrenze bei einer Linearachse überfahren wurde.

Referenzpunktfahrt in Abhängigkeit der Startposition

Bei einer Referenzpunktfahrt müssen Sie verschiedene Fälle unterscheiden, die abhängig sind

- von der Lage des Antriebs beim Start einer Referenzpunktfahrt,
- von der parametrierten Startrichtung,
- von der parametrierten Lage der Nullmarke zum Referenzpunktschalter.

Die folgende Tabelle zeigt die Fälle für REFPT_TYPE = 0 und REFPT_TYPE = 1. Für REFPT_TYPE = 2 und REFPT_TYPE = 3 gelten die Bilder entsprechend.

Bedingungen der Referenzpunktfahrt	Ablauf der Referenzpunktfahrt
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert.	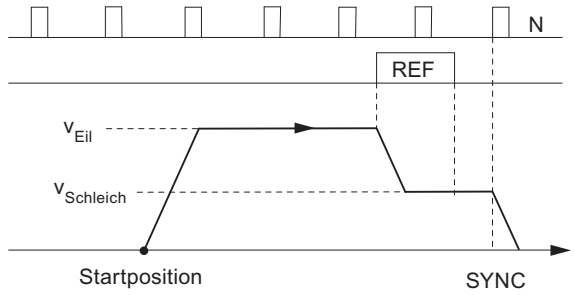
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 1): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung minus parametriert.	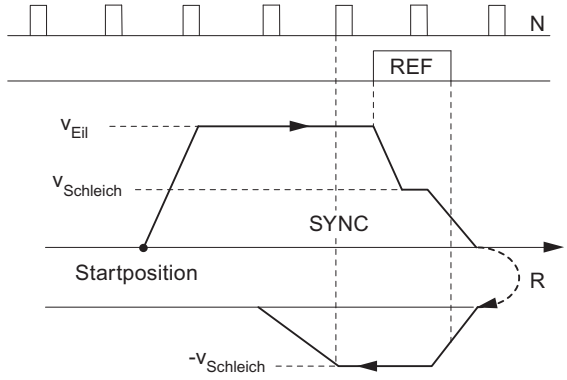
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung muss in Richtung plus parametriert sein. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Der Umkehrschalter ist positiver als der Referenzpunktschalter.	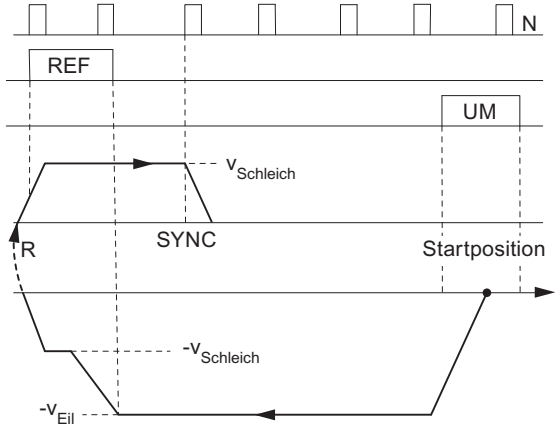

Bedingungen der Referenzpunktfahrt	Ablauf der Referenzpunktfahrt
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 1): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung minus parametriert. • Startposition der Referenzpunktfahrt ist auf dem Referenzpunktschalter.	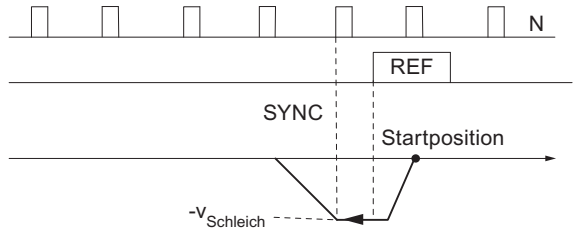
Beispiel für Referenzpunktfahrt (REFPT_TYPE = 0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Der Umkehrschalter ist positiver als der Referenzpunktschalter.	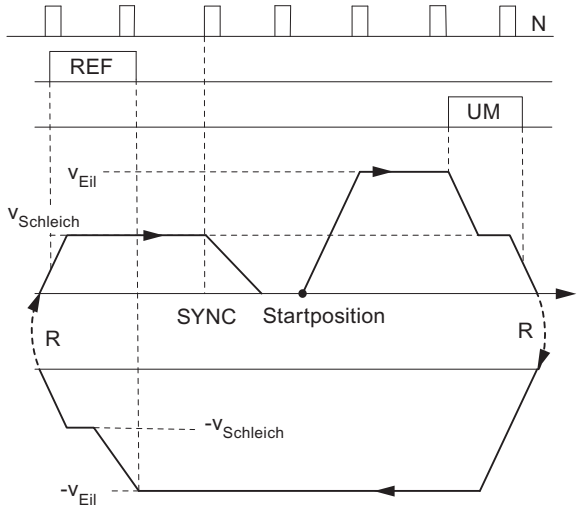
Beispiel für Referenzpunktfahrt (REFPT_TYPE=0): • Startrichtung ist plus. • Lage der Nullmarke vom Referenzpunktschalter ist in Richtung plus parametriert. • Startgeschwindigkeit = Schleichgeschwindigkeit	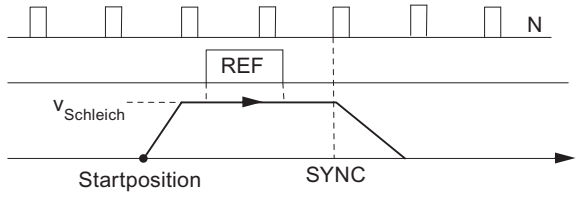
R = Richtungsumkehr REF = Referenzpunktschalter UM = Umkehrschalter N = Nullmarke des Gebers SYNC = Synchronisation wurde erreicht	

Betriebsart "Schrittmaßfahrt"

Definition

In der Betriebsart "Schrittmaßfahrt" kann die FM 451 den Antrieb

- auf **absolute** Ziele bewegen,
- **relativ** um ein Wegstück in eine vorgegebene Richtung bewegen.

Das Ziel bzw. die relativen Wege geben Sie als Schrittmaße der FM 451 vor. Sie haben die Möglichkeit, maximal 100 Schrittmaße in eine Tabelle einzutragen, die sowohl für die Betriebsart "Schrittmaßfahrt relativ" als auch für "Schrittmaßfahrt absolut" gültig sind. Unabhängig von der Schrittmaßtabelle können Sie mit dem Schrittmaß 254 (Seite 2640) und Schrittmaß 255 (Seite 2641) den Weg vorgeben.

Mit dem Schrittmaß 252 (Seite 2640) haben Sie die Möglichkeit, den Antrieb zunächst ohne Ziel in vorgegebener Richtung zu bewegen (Endlosfahren). Beim Eintreffen eines externen Ereignisses (xI3) wird dann auf das parametrierte Ziel positioniert.

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.
- Die Schrittmaße müssen auf der Baugruppe vorhanden sein.

Interpretation der Schrittmaße

Je nachdem welche Schrittmaßfahrt Sie wählen, interpretiert die FM 451 die Vorgaben unterschiedlich.

- Schrittmaßfahrt absolut: Die Schrittmaße werden als absolute Zielposition interpretiert.
  Bei einer Rundachse darf dabei das Ende der Rundachse nicht überfahren werden.
- Schrittmaßfahrt relativ: Die Schrittmaße werden als Wegdifferenz von der Startposition interpretiert.
  Bei einer Rundachse darf dabei das Ende der Rundachse überfahren werden (Mehrfachumdrehungen sind zulässig).

Hinweis

Für die Betriebsart "Schrittmaßfahrt relativ" sind nur positive Schrittmaße zulässig.

Die Schrittmaße erhalten Ihr Vorzeichen durch die Richtungsvorgaben DIR_P bzw. DIR_M.

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 1 bis 100

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 1 bis 100	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Tragen Sie die Schrittmaße in die Tabellen (TRGL1; TRGL2) ein.	
3	Schreiben Sie die Schrittmaßtabellen (TRGL1/2WR_EN = 1).	
4	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
5	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
6	Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 1 … 100).	
7	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
8	Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.	
Schritte 2 und 3 sind nur erforderlich, wenn noch keine Schrittmaße vorhanden sind oder die vorhandenen Schrittmaße geändert werden sollen.		

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 252

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 252	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
3	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
4	Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 252).	
5	Tragen Sie die Startgeschwindigkeit ein: • Eilgang (SPEED252 = 1) • Schleichgang (SPEED252 = 0)	
6	Tragen Sie das Schrittmaß für die Schrittmaßnummer 252 ein (TRG252_254).	
7	Setzen Sie das Anstoßbit zum Schreiben des Schrittmaßes (TRG252_254_EN = 1).	

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 252	
8	Setzen Sie das Steuersignal: • Linearachse / Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	
9	Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.	
10	Beim Eintreffen eines externen Ereignisses (Digitaleingang xI3) wird das Positionieren auf das parametrierte Ziel (Schrittmaß) durchgeführt.	
Die unter Schritt 6 und 7 aufgeführten Aktionen können auch erst während der Fahrt ausgeführt werden. In diesem Fall steuert die Baugruppe FM 451 bis zur Schrittmaßangabe ein Ziel an, das um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegt.		

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 254

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 254	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
3	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
4	Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 254).	
5	Tragen Sie das Schrittmaß für die Schrittmaßnummer 254 ein (TRG252_254).	
6	Setzen Sie das Anstoßbit zum Schreiben des Schrittmaßes (TRG252_254_EN = 1).	
7	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
8	Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.	

Ablauf der Betriebsart "Schrittmaßfahrt" mit Schrittmaßnummer 255

Schritt	Schrittmaßfahrt absolut	Schrittmaßfahrt relativ
	Schrittmaßnummer 255	
1	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut" (MODE_IN = 5).	Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt relativ" (MODE_IN = 4).
2	Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).	
3	Setzen Sie den Funktionsschalter für "Freigabeeingang nicht auswerten" (EI_OFF = 1) oder verdrahten Sie den Freigabeeingang für den entsprechenden Kanal.	
4	Tragen sie die Schrittmaßnummer ein (MODE_TYPE = 255).	
5	Tragen Sie das Schrittmaß für die Schrittmaßnummer 255 ein (TRG255).	
6	Tragen Sie den Wert für die Umschaltdifferenz der Schrittmaßnummer 255 ein (CHGDIF255).	
7	Tragen Sie den Wert für die Abschaltdifferenz der Schrittmaßnummer 255 ein (CUTDIF255).	
8	Setzen Sie das Anstoßbit zum Schreiben von Schrittmaß, Ab- und Umschaltdifferenz (TRG255_EN = 1).	
9	Setzen Sie das Steuersignal: • Linearachse: – START; die Richtung ist durch das Ziel und den aktuellen Istwert eindeutig bestimmt. • Rundachse: – START; das Ziel wird auf dem kürzesten Weg angefahren. – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus	Setzen Sie das Steuersignal: • Linearachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus • Rundachse: – DIR_P; Start in Richtung plus – DIR_M; Start in Richtung minus
10	Rufen Sie die Anweisung ABS_CTRL bzw. ABS_CTRL_451 auf.	

Das folgende Bild zeigt ein Beispiel für die Betriebsart "Schrittmaßfahrt".

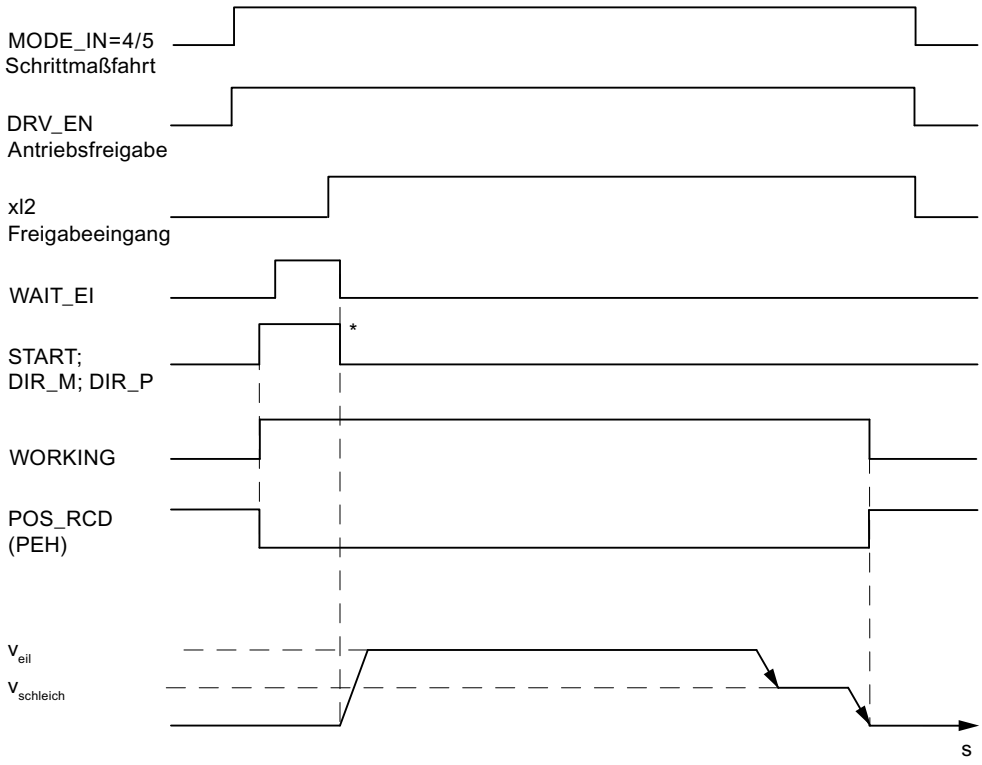

* Die Startsignale werden von der Anweisung ABS_CTRL bzw. ABS_CTRL_451 zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positionierung starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus
15.6	SPEED252	BOOL	FALSE	Startgeschwindigkeit für Schrittmaßfahrt mit Schrittmaß-nummer 252 • 0 = Schleichgang • 1 = Eilgang
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	4 = Schrittmaßfahrt relativ 5 = Schrittmaßfahrt absolut
17.0	MODE_TYPE	BYTE	B#16#0	Schrittmaßnummer 1 … 100, 252, 254 oder 255
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positionierung läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Adresse	Name	Typ	Anfangswert	Kommentar
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben
36.3	TRG255_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 255 schreiben
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaßnummer 1 … 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaßnummer 51 … 100) schreiben
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 252 oder 254
100.0	TRG255	DINT	L#0	Schrittmaß für Schrittmaßnummer 255
104.0	CHGDIF_255	DINT	L#0	Umschaltdifferenz für Schrittmaßnummer 255
108.0	CUTDIF_255	DINT	L#0	Abschaltdifferenz für Schrittmaßnummer 255

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus	
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus	
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus	
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus	
120.0	TRGL1.TRG[1]	DINT	L#0	Schrittmaßnummer 1	Schrittmaßtabelle 1
. . .	. . .	. . .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Schrittmaßnummer 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Schrittmaßnummer 51	Schrittmaßtabelle 2
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Schrittmaßnummer 100	

Restweg

Der Restweg ist die Differenz zwischen Ziel (Schrittmaß) und Istwert. Der Restweg ist

- positiv, wenn das Ziel noch nicht erreicht wurde.
- negativ, wenn das Ziel bereits überfahren wurde.

Bei einer Rundachse ist der angezeigte Restweg nicht verwendbar.

Absteuern der Schrittmaßfahrt

Die Betriebsart "Schrittmaßfahrt" wird abgesteuert, wenn die FM 451 ein STOP-Signal empfängt (STOP = 1).

Nach dem Absteuern der Fahrt bleibt ein Restweg bestehen.

Der verbleibende Restweg bei einer "Schrittmaßfahrt relativ" und "Schrittmaßfahrt absolut mit der Schrittmaßnummer 252" kann zu Ende verfahren werden, wenn

- die Betriebsart unverändert ist, und
- die Schrittmaßnummer unverändert ist, und
- die Richtung unverändert ist, und
- der verbleibende Restweg größer als die parametrierte Abschaltdifferenz ist.

Den Restweg verfahren Sie, indem Sie die "Schrittmaßfahrt relativ" oder die "Schrittmaßfahrt absolut mit der Schrittmaßnummer 252" unverändert noch einmal starten.

Abbrechen der Schrittmaßfahrt

Die Betriebsart "Schrittmaßfahrt" wird abgebrochen, wenn das Signal "Antriebsfreigabe" gelöscht wird (DRV_EN = 0).

Restweg löschen

Mit dem Auftrag "Restweg löschen" löschen Sie einen anstehenden Restweg.

Bei der "Schrittmaßfahrt mit Schrittmaßnummer 252" beginnt die Fahrt nach dem "Restweg löschen" wieder mit dem Endlosfahren.

Mit dem Aufruf einer anderen Betriebsart oder dem Start der Betriebsart in die andere Richtung löschen Sie ebenfalls den anstehenden Restweg.

Verwendetes Datum im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.2	DELDIST_EN	BOOL	FALSE	1 = Restweg löschen

Auftrag "Istwert setzen" / "Fliegendes Istwert setzen" / "Istwert setzen rückgängig"

Definition

Mit dem Auftrag "Istwert setzen" bzw. "Fliegendes Istwert setzen" ordnen Sie dem aktuellen Geberstand eine neue Koordinate zu. Der Arbeitsbereich wird auf einen anderen Bereich der Achse projiziert.

Die Verschiebung des Arbeitsbereiches ermitteln Sie mit (IST_{neu} - $IST_{aktuell}$).

- IST_{neu} ist der Vorgabewert
- $IST_{aktuell}$ ist der Istwert zum Zeitpunkt der Ausführung

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.

Für "Istwert setzen" muss der Positioniervorgang beendet sein.

Ablauf des Auftrags

1. Tragen Sie die Koordinate AVAL bzw. FVAL für den Istwert (**IST_{neu}**) ein.
   - Linearachse:
     Den angegebenen Istwert müssen Sie so wählen, dass die Software-Endschalter nach dem Aufruf des Auftrags noch innerhalb des zulässigen Verfahrbereichs liegen.
     Der Betrag der Verschiebung, der sich aus (**IST_{neu} - $IST_{aktuell}$**) ergibt, muss kleiner oder gleich dem Betrag des zulässigen Verfahrbereichs sein (maximal 100 m bzw. 1000 m).
   - Rundachse:
     Für den angegebenen Istwert muss gelten:
     0 ≤ Istwert < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit (AVAL_EN = 1 bzw. FVAL_EN = 1).

"Istwert setzen" wird sofort ausgeführt.

"Fliegendes Istwert setzen" wird mit der nächsten steigenden Flanke am Digitaleingang xI3 ausgeführt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.7	AVAL_EN	BOOL	FALSE	1 = Istwert setzen
36.0	FVAL_EN	BOOL	FALSE	1 = Fliegendes Istwert setzen
84.0	AVAL	DINT	L#0	Koordinate für "Istwert setzen"
88.0	FVAL	DINT	L#0	Koordinate für "Fliegendes Istwert setzen"

Auswirkungen des Auftrags

Am Beispiel "Istwert setzen" auf 300 mm erkennen Sie, wie dieser Auftrag den Arbeitsbereich auf eine bestimmte Position der Achse projiziert. Es ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Istwertkoordinate gesetzt.
- Der Arbeitsbereich wird auf der Achse verschoben.
- Die einzelnen Punkte (z. B. Software-Endschalter Ende) innerhalb des Arbeitsbereichs behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Istwert setzen".

Istwert setzen	SEA	IST	SEE
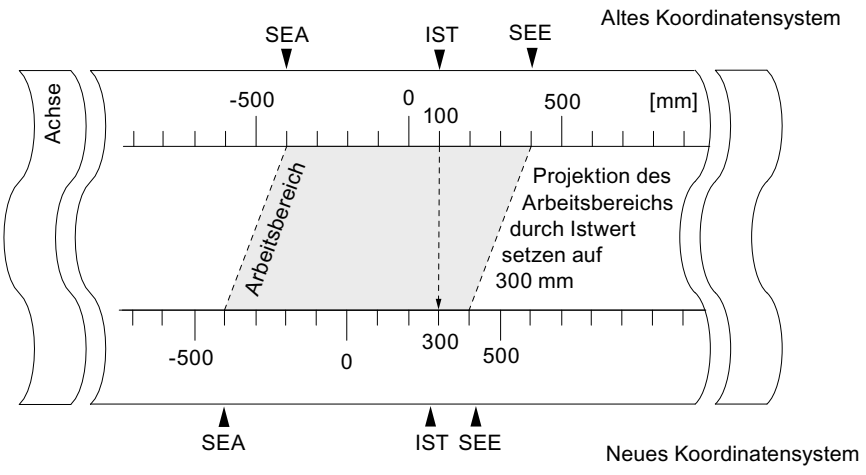	-400	100	400
	-400	300	400

Rücknahme des Auftrags ("Istwert setzen rückgängig")

Mit dem Auftrag "Istwert setzen rückgängig" setzen Sie die **letzte** Arbeitsbereichverschiebung, die durch "Istwert setzen" erzeugt wurden, wieder zurück.

Ein einmal angestoßenes "Fliegendes Istwert setzen" kann vor der Ausführung durch eine steigende Flanke am Digitaleingang xI3 nicht mehr gelöscht werden. Es kann aber durch ein neues "Fliegendes Istwert setzen" überschrieben werden. Bei einem Baugruppenanlauf werden diese Aufträge zurückgesetzt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.3	AVALREM_EN	BOOL	FALSE	1 = Istwert setzen rückgängig

Auftrag "Nullpunktverschiebung"

Definition

Mit dem Auftrag "Nullpunktverschiebung" verschieben Sie den Nullpunkt im Koordinatensystem um den eingegebenen Wert.

Das Vorzeichen bestimmt die Richtung der Verschiebung im Koordinatensystem.

Neue Koordinate ermitteln

Alle Werte im verschobenen Koordinatensystem berechnen Sie mit der folgenden Formel:

$\text{Koordinate}_{neu} = \text{Koordinate}_{alt} - (\text{NPV}_{neu} - \text{NPV}_{alt})$

NPV_{alt} kennzeichnet dabei eine eventuell bestehende alte Nullpunktverschiebung.

Wenn vor dem Aufruf keine Nullpunktverschiebung aktiv war, setzen Sie für NPV_{alt} den Wert 0 ein.

Sie können damit ermitteln, welche Koordinatenwerte z. B. die Software-Endschalter annehmen.

Voraussetzungen

- Die Positionierung muss beendet sein.
- Die Achse muss parametriert sein.

Ablauf des Auftrags

1. Tragen Sie den Wert ZOFF für die Nullpunktverschiebung ein.
   - Linearachse:
     Die Nullpunktverschiebung muss so gewählt werden, dass die Software-Endschalter nach dem Aufruf des Auftrags noch innerhalb des zulässigen Zahlenbereichs liegen.
   - Rundachse:
     Für die Nullpunktverschiebung muss gelten:
     Betrag der Nullpunktverschiebung ≤ Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit (ZOFF_EN = 1).

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.1	ZOFF_EN	BOOL	FALSE	1 = Nullpunktverschiebung setzen
80.0	ZOFF	DINT	L#0	Nullpunktverschiebung

Auswirkungen des Auftrags bei einer Linearachse

Am Beispiel einer Nullpunktverschiebung von -200 mm erkennen Sie, dass dieser Auftrag das Koordinatensystem positiv verschiebt. Es ergeben sich folgende Auswirkungen:

- Der Arbeitsbereich wird physikalisch **nicht** verschoben.
- Den einzelnen Punkten (wie z. B. den Software-Endschaltern) werden neue Koordinatenwerte zugeordnet.

Die folgende Tabelle zeigt die Verschiebung des Koordinatensystems durch "Nullpunktverschiebung".

Nullpunktverschiebung	SEA	IST	SEE
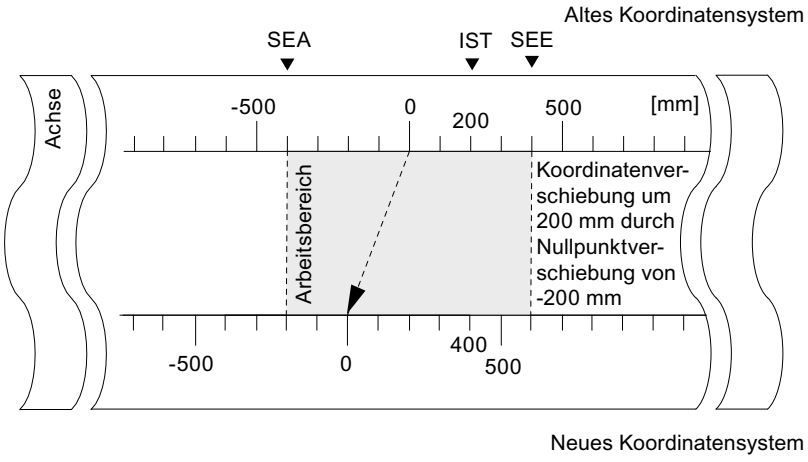	-400	200	400
	-200	400	600

Auswirkungen des Auftrags bei einer Rundachse

Am Beispiel einer Nullpunktverschiebung um -45° erkennen Sie, wie dieser Auftrag das Koordinatensystem **dreht**.

Das folgende Bild zeigt die Drehung des Koordinatensystems durch eine Nullpunktverschiebung.

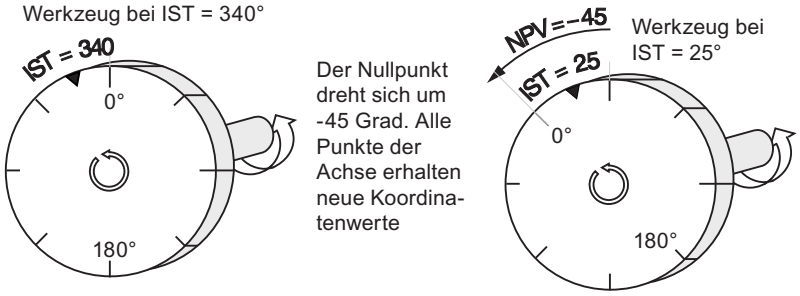

Unter Berücksichtigung einer **NPV_{alt} = 0** ergibt sich ein neuer Wert von 385°.

Da am Ende der Rundachse bei einer positiven Drehrichtung der Istwert wieder bei 0 beginnt, errechnet sich der tatsächliche Istwert von 25° aus:

$Koordinate_{neu}$ = $Koordinate_{alt}$ - (NPV_{neu} - NPV_{alt}) - Ende der Rundachse

Wegfall der Synchronisation

Wenn durch einen Fehler die Synchronisation verloren geht oder durch "Referenzpunktfahrt" neu gesetzt wird, **bleibt** eine Nullpunktverschiebung aktiv.

Rücknahme des Auftrags

Mit der Vorgabe einer Nullpunktverschiebung von 0 mm setzen Sie eine bestehende Nullpunktverschiebung wieder zurück.

Auftrag "Bezugspunkt setzen"

Definition

Mit dem Auftrag "Bezugspunkt setzen" synchronisieren Sie die Achse. Der Auftrag verschiebt den Arbeitsbereich. Alle Verschiebungen, die durch "Istwert setzen" erzeugt wurden, bleiben erhalten.

Die Einstellung projiziert den Arbeitsbereich auf die Achse. Durch unterschiedliche Werteingaben kann deshalb der Arbeitsbereich an beliebigen Positionen im physikalischen Bereich der Achse liegen.

Voraussetzungen

- Die Positionierung muss beendet sein.
- Die Achse muss parametriert sein.

Ablauf des Auftrags

1. Tragen Sie den Wert REFPT für die Bezugspunktkoordinate ein.
   - Linearachse:

     Die Bezugspunktkoordinate darf nicht außerhalb der Software-Endschalter liegen. Das gilt auch für die Bezugspunktkoordinate in einem verschobenen Koordinatensystem.
   - Rundachse:
     Für die Bezugspunktkoordinate muss gelten:
     0 ≤ Bezugspunktkoordinate < Ende der Rundachse
2. Setzen Sie das entsprechende Anstoßbit (REFPT_EN = 1).

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
25.0	SYNC	BOOL	FALSE	1 = Achse ist synchronisiert
35.6	REFPT_EN	BOOL	FALSE	1 = Bezugspunkt setzen
92.0	REFPT	DINT	L#0	Bezugspunktkoordinate

Auswirkungen des Auftrags

Am Beispiel "Bezugspunkt setzen" auf 400 mm erkennen Sie, wie dieser Auftrag den Arbeitsbereich auf eine bestimmte physikalische Position der Achse projiziert. Es ergeben sich folgende Auswirkungen:

- Die Istposition wird auf den Wert der Bezugspunktkoordinate gesetzt.
- Der Arbeitsbereich wird auf der Achse verschoben.
- Die einzelnen Punkte (z. B. Software-Endschalter Ende) behalten ihren ursprünglichen Wert, liegen aber auf neuen Positionen.
- Das Bit SYNC in den Rückmeldesignalen wird gesetzt.

Die folgende Tabelle zeigt die Verschiebung des Arbeitsbereichs auf der Achse durch "Bezugspunkt setzen".

Bezugspunkt setzen	SEA	IST	SEE
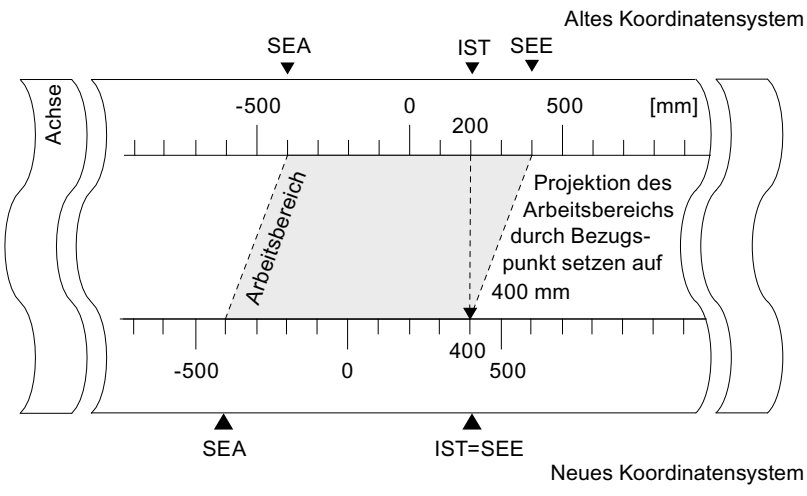	-400	200	400
	-400	400	400

Besonderheiten Absolutwertgeber

Dieser Auftrag ist für eine Absolutwertgeberjustage (Seite 2627) notwendig.

Funktionsschalter "Längenmessung" / "Kantenerfassung"

Definition

Mit "Längenmessung" und "Kantenerfassung" können Sie die Länge eines Werkstücks ermitteln.

Der Funktionsschalter "Längenmessung" bzw. "Kantenerfassung" ist und bleibt so lange aktiv, bis Sie ihn wieder ausschalten oder die jeweils andere Messmethode wählen. Falls Sie beide Messmethoden gleichzeitig wählen, wird ABS_CTRL bzw. ABS_CTRL_451 die Längenmessung einschalten.

Voraussetzungen

- Die Positionierung muss beendet sein.
- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.
- Am Eingang xI3 muss ein **prellfreier** Schalter angeschlossen sein.
- Wenn diese Funktionsschalter während einer Positionierung gewählt werden, können "Längenmessung" und "Kantenerfassung" erst bei der nächsten Positionierung ausgeführt werden.

Ablauf der Messungen

Je nach der Art der Messung aktualisiert die FM 451 die Daten auf der Baugruppe zu einem unterschiedlichen Zeitpunkt. Jede Aktualisierung meldet die FM 451 in einem Parameter an der Rückmeldeschnittstelle.

Die folgende Tabelle zeigt den Ablauf der Messungen.

Ablauf	Längenmessung	Kantenerfassung
	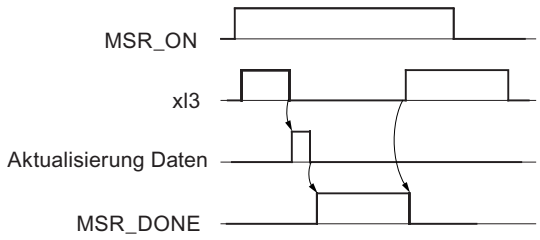	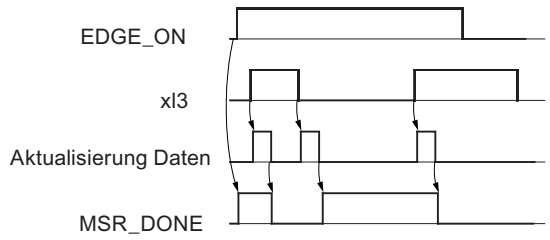
1	Setzen Sie den Funktionsschalter für "Längenmessung" (MSR_ON = 1).	Tragen Sie ggf. einen Wert EDGEDIST für den minimalen Kantenabstand im Parameter-DB ein. Schreiben und aktivieren Sie die Maschinendaten.
2	Starten Sie eine Positionierung.	Setzen Sie den Funktionsschalter für "Kantenerfassung" (EDGE_ON = 1). Der Parameter MSR_DONE wird gesetzt.
3	Die steigende Flanke des Eingangs xI3 startet die Längenmessung.	Starten Sie eine Positionierung.
4	Die fallende Flanke des Eingangs xI3 beendet eine laufende Messung. Die FM 451 aktualisiert die Daten Anfangswert, Endwert und Länge.	Die steigende Flanke des Eingangs xI3 startet die Messung. Nach dem Verfahren des Wegstücks "Minimaler Kantenabstand" wird die Messung gültig. Dies wird mit MSR_DONE = 0 gemeldet. Die Messergebnisse werden aktualisiert und können ausgelesen werden, der Anfangswert der Messung wird eingetragen; Endwert und Länge werden -1.
5	Mit dem gesetzten Parameter MSR_DONE meldet die FM 451 das Aktualisieren der Daten. Der Parameter zeigt an, dass die Messung beendet ist. Die Messergebnisse können ausgelesen werden.	Nach der Aktualisierung meldet die FM 451 die Veränderung durch das Rücksetzen des Parameters MSR_DONE.
6	Der Start einer weiteren Messung mit der steigenden Flanke von xI3 setzt den Parameter MSR_DONE zurück.	Die fallende Flanke des Eingangs xI3 beendet eine laufende Messung. Die FM 451 aktualisiert die Daten für Endwert der Messung und Länge.

Ablauf	Längenmessung	Kantenerfassung
7	-	Nach der Aktualisierung meldet die FM 451 die Veränderung durch das Setzen des Parameters MSR_DONE. Die Messergebnisse können ausgelesen werden.
8	-	Der Start einer weiteren Messung mit der steigenden Flanke von xI3 setzt den Parameter MSR_DONE zurück.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
25.1	MSR_DONE	BOOL	FALSE	1 = Längenmessung/Kantenerfassung abgeschlossen
34.3	EDGE_ON	BOOL	FALSE	1 = Kantenerfassung ein
34.4	MSR_ON	BOOL	FALSE	1 = Längenmessung ein
35.0	MDWR_EN	BOOL	FALSE	1 = Maschinendaten schreiben
35.1	MD_EN	BOOL	FALSE	1 = Maschinendaten aktivieren
37.0	MSRRD_EN	BOOL	FALSE	1 = Messwerte lesen
136.0	BEG_VAL	DINT	L#0	Anfangswert der Längenmessung/Kantenerfassung
140.0	END_VAL	DINT	L#0	Endwert der Längenmessung/Kantenerfassung
144.0	LEN_VAL	DINT	L#0	Länge

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	Minimaler Kantenabstand bei Kantenerfassung
Wenn das Ende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen (BEG_VAL, END_VAL und LEN_VAL = -1).				

Randbedingungen für eine Längenmessung

- Der Abstand zwischen Ausschaltflanke und Einschaltflanke am Eingang xI3 muss so groß sein, dass Ihr Programm in der CPU das Ergebnis der Messung korrekt auswerten kann, bevor eine neue Messung beginnt.
- Der minimale Abstand zwischen der steigenden und der fallenden Flanke am Eingang xI3 wie auch zwischen der fallenden und der nächsten steigenden Flanke am Eingang xI3 muss größer als 8 ms sein.

Fehlerhafte Messung

Im Fall einer fehlerhaften Längenmessung / Kantenerfassung liefert die FM 451 den Wert -1 für die Länge zurück. Eine Messung ist fehlerhaft, wenn

- bei einer Rundachse die gemessene Länge größer als 2^{31} ist,
- gleichzeitig die Einschalt- und die Ausschaltflanke von der FM 451 erkannt wird, z. B. durch Schalterprellen.

Funktionsschalter "Schleifenfahrt"

Definition

Mit "Schleifenfahrt" legen Sie die Richtung fest, in die ein Ziel mit Kraftschluss angefahren wird. Die Schleifenfahrt können Sie verwenden, wenn nur in einer Richtung Kraftschluss zwischen Motor und Achse sichergestellt werden kann.

Ein Ziel, das entgegen der vorgegebenen Richtung angefahren wird, wird zunächst überfahren. Anschließend nimmt die Baugruppe eine Richtungsumkehr vor und fährt das Ziel dann in vorgegebener Richtung an.

Voraussetzungen

- Die Achse muss parametriert sein.
- Die Achse muss synchronisiert sein.
- Bei einer Schleifenfahrt entgegen der Verfahrrichtung zum Ziel liegt die maximale Zielposition:
  - in Verfahrrichtung plus
    Ziel < SEE - ½ Zielbereich - Abschaltdifferenz plus - Umschaltdifferenz minus
  - in Verfahrrichtung minus
    Ziel > SEA + ½ Zielbereich + Abschaltdifferenz minus + Umschaltdifferenz plus
- Eine Schleifenfahrt wird nicht durchgeführt, wenn das Ziel in Richtung der Schleifenfahrt angefahren wird. In diesem Fall wird eine Schrittmaßfahrt ohne Richtungsumkehr durchgeführt.
- Der Ablauf der Betriebsart "Schrittmaßfahrt" (Seite 2590) muss bekannt sein.

Ablauf der Schleifenfahrt

1. Setzen Sie das Steuersignal für die Betriebsart "Schrittmaßfahrt absolut / relativ" (MODE_IN = 5/4).
2. Setzen Sie das Steuersignal für die Antriebsfreigabe (DRV_EN = 1).
3. Setzen Sie den Funktionsschalter "Freigabeeingang" (EI_OFF = 1) oder verdrahten Sie für den entsprechenden Kanal den Freigabeeingang.
4. Tragen Sie die Schrittmaßnummer ein (MODE_TYPE = 1 … 100, 254, 255).
5. Setzen Sie den Funktionsschalter "Schleifenfahrt" (PLOOP_ON / MLOOP_ON = 1).
6. Starten Sie die Schrittmaßfahrt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
15.0	START	BOOL	FALSE	1 = Positionierung starten
15.2	DIR_M	BOOL	FALSE	1 = Richtung minus
15.3	DIR_P	BOOL	FALSE	1 = Richtung plus

Adresse	Name	Typ	Anfangswert	Kommentar
15.7	DRV_EN	BOOL	FALSE	1 = Antriebsfreigabe einschalten
16.0	MODE_IN	BYTE	B#16#0	4 / 5 = Schrittmaßfahrt relativ / absolut
17.0	MODE_TYPE	BYTE	B#16#0	Schrittmaßnummer 1 … 100, 254 oder 255
34.0	PLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung plus
34.1	MLOOP_ON	BOOL	FALSE	1 = Schleifenfahrt in Richtung minus
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Fiktives Ziel

Wenn Sie eine Positionierung auf ein Ziel starten, das entgegen der Richtung der parametrierten Schleifenfahrt liegt, ermittelt die Baugruppe zu diesem Ziel ein fiktives Ziel, an dem sie eine Richtungsumkehr ausführt und dann in richtiger Richtung das Ziel anfährt.

Dieses fiktive Ziel **muss** mindestens um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegen.

Die Entfernung des fiktiven Ziels vom parametrierten Ziel wird richtungsabhängig ermittelt. Die folgende Tabelle zeigt die Berechnung der Lage des fiktiven Ziels bei Schleifenfahrt.

Vorgaben	Lage des fiktiven Ziels
Parametrierung: Schleife + (Kraftschluss plus) und Fahrt in Richtung minus	Das fiktive Ziel (**Ziel**$_f$) hat den Wert: **Ziel**$_f$ **= Ziel - Abschaltdifferenz minus - Umschaltdifferenz plus** 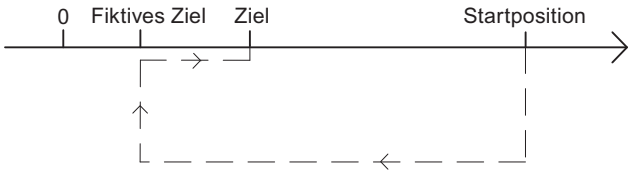
Parametrierung: Schleife - (Kraftschluss minus) und Fahrt in Richtung plus	Das fiktive Ziel (**Ziel**$_f$) hat den Wert: **Ziel**$_f$**= Ziel + Abschaltdifferenz plus + Umschaltdifferenz minus** 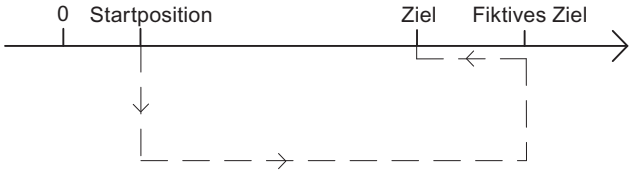

Beispiel

Anhand einer Positionierung mit Schleifenfahrt minus auf ein maximales Ziel zeigen wir Ihnen die Lage des fiktiven Ziels.

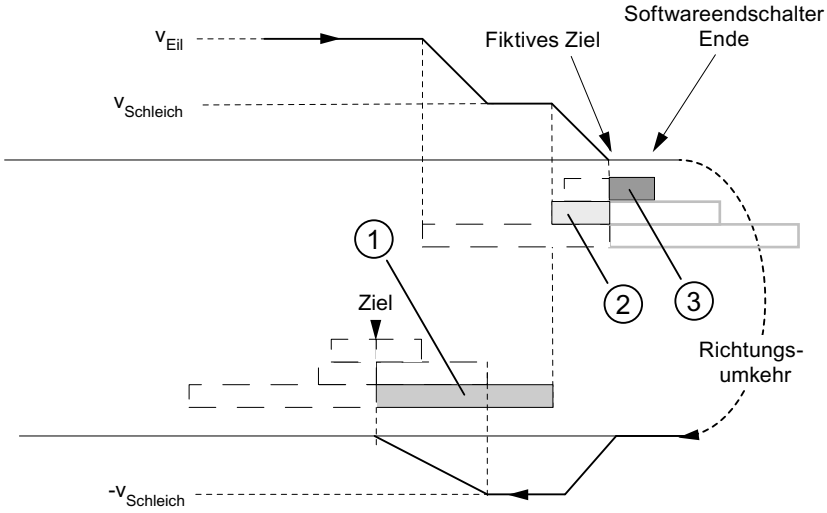

① Umschaltdifferenz negativ
② Abschaltdifferenz positiv
③ ½ Zielbereich

Funktionsschalter "Freigabeeingang"

Definition

Der Freigabeeingang ist ein externer Eingang, mit dem eine Positionierung durch ein externes Ereignis freigegeben werden kann.

Freigabeeingang auswerten (EI_OFF = 0)

Der entsprechende Freigabeeingang (xI2) für den Kanal muss verdrahtet sein.

Sie haben damit die Möglichkeit, den Start einer Positionierung vorzubereiten. Sie starten die Positionierung unabhängig vom Programmablauf Ihres Anwenderprogramms, indem Sie an den Freigabeeingang ein "1"-Signal anlegen.

Sie haben folgende Möglichkeiten für die Art der Auswertung des Freigabeeingangs:

- **pegelgesteuert (EI_TYPE = 0)**
  Die Fahrt beginnt, wenn Sie an den Freigabeeingang ein "1"-Signal anlegen, und wird abgesteuert, wenn Sie an den Freigabeeingang ein "0"-Signal anlegen.
- **flankengesteuert (EI_TYPE = 1)**
  Die Fahrt beginnt mit der Erkennung einer steigenden Flanke am Freigabeeingang. Der weitere Signalverlauf an diesem Eingang nimmt keinen Einfluss mehr auf den Ablauf der begonnenen Fahrt.

Freigabeeingang nicht auswerten (EI_OFF = 1)

Wenn Sie die Auswertung des Freigabeeingangs abschalten, startet eine Betriebsart unmittelbar nach dem Erkennen des Startsignals. Es ist dann nicht möglich, eine Betriebsart vorzubereiten und zu einem definierten späteren Zeitpunkt zu starten.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
34.2	EI_OFF	BOOL	FALSE	1 = Freigabeeingang nicht auswerten

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
99.1	EI_TYPE	BOOL	FALSE	0 = Pegelgesteuert 1 = Flankengesteuert

Auftrag "Positionsdaten lesen"

Definition

Mit dem Auftrag "Positionsdaten lesen" können Sie Schrittmaß, Restweg und Geschwindigkeit zum aktuellen Zeitpunkt lesen.

Ablauf des Auftrags

1. Setzen Sie das Anstoßbit im Kanal-DB (ACTSPD_EN = 1).
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.1	ACTSPD_EN	BOOL	FALSE	1 = Positionsdaten lesen
112.0	ACTSPD	DINT	L#0	Aktuelle Geschwindigkeit
116.0	DIST_TO_GO	DINT	L#0	Restweg
120.0	ACT_TRG	DINT	L#0	Aktuelles Schrittmaß

Auftrag "Geberdaten lesen"

Definition

Mit dem Auftrag "Geberdaten lesen" lesen Sie die aktuellen Daten des Gebers sowie den Wert für die Absolutwertgeberjustage.

Voraussetzungen

Sie können den Wert für die Absolutwertgeberjustage (Seite 2627) auslesen, nachdem Sie den Auftrag "Bezugspunkt setzen" (Seite 2601) ausgeführt haben.

Ablauf des Auftrags

1. Setzen Sie das Anstoßbit im Kanal-DB (ENCVAL_EN = 1).
2. Die Daten werden im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
37.2	ENCVAL_EN	BOOL	FALSE	1 = Geberwerte lesen
124.0	ENCVAL	DINT	L#0	Geberistwert (interne Darstellung)
128.0	ZEROVAL	DINT	L#0	Letzter Nullmarkenwert (interne Darstellung)
132.0	ENC_ADJ	DINT	L#0	Absolutwertgeberjustage

Rückmeldesignale für die Positionierung

Definition

Mit den "Rückmeldesignalen für die Positionierung" werden Sie über den aktuellen Zustand der Positionierung informiert.

Ablauf

Die Daten werden bei jedem Aufruf der Anweisung ABS_CTRL bzw. ABS_CTRL_451 im Kanal-DB abgelegt.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
23.0	ST_ENBLD	BOOL	FALSE	1 = Start freigegeben
23.1	WORKING	BOOL	FALSE	1 = Positionierung läuft
23.2	WAIT_EI	BOOL	FALSE	1 = Achse wartet auf externe Freigabe
23.4	SPEED_OUT	BOOL	FALSE	0 = Schleichgang 1 = Eilgang
23.5	ZSPEED	BOOL	FALSE	1 = Achse befindet sich im Stillstandsbereich
23.6	CUTOFF	BOOL	FALSE	1 = Achse befindet sich im Abschaltbereich
23.7	CHGOVER	BOOL	FALSE	1 = Achse befindet sich im Umschaltbereich
24.0	MODE_OUT	BYTE	B#16#0	Aktive Betriebsart
25.2	GO_M	BOOL	FALSE	1 = Achse fährt in Richtung minus
25.3	GO_P	BOOL	FALSE	1 = Achse fährt in Richtung plus

Adresse	Name	Typ	Anfangswert	Kommentar
25.5	FVAL_DONE	BOOL	FALSE	1 = Fliegendes Istwert setzen abgeschlossen
25.7	POS_RCD	BOOL	FALSE	1 = Position erreicht
26.0	ACT_POS	DINT	L#0	Aktueller Istwert (aktuelle Position der Achse)

Rückmeldesignale für die Diagnose abfragen

Definition

Mit den "Rückmeldesignalen für die Diagnose" werden Sie über aufgetretene Diagnoseereignisse informiert.

Ablauf

1. Wenn die Baugruppe ein neues Ereignis in den Diagnosepuffer einträgt, setzt sie in der Rückmeldeschnittstelle das Bit DIAG in allen Kanälen. Jedes Auftreten eines Fehlers (Seite 2654) bewirkt einen Eintrag in den Diagnosepuffer. Beim Löschen des Diagnosepuffers wird das Bit DIAG auch gesetzt.
2. Wenn der Aufruf einer Betriebsart bzw. das Steuern einer aktiven Betriebsart nicht möglich ist oder fehlerhaft durchgeführt wurde, setzt die Baugruppe in der Rückmeldeschnittstelle einen Bedienfehler OT_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen. Solange der Bedienfehler ansteht, können Sie weder eine neue Betriebsart starten noch die angehaltene Betriebsart fortsetzen. Einen anstehenden Bedienfehler quittieren Sie mit OT_ERR_A = 1.
3. Wenn die Baugruppe einen Schreibauftrag mit fehlerhaften Daten erkennt, setzt sie in der Rückmeldeschnittstelle das Bit DATA_ERR. Die Fehlerursache wird in den Diagnosepuffer eingetragen.
4. Die Rückmeldesignale werden im Kanal-DB abgelegt.
5. Wenn der Diagnosepuffer gelesen ist, setzt die Baugruppe das Bit DIAG in allen Kanälen wieder auf 0.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
22.2	DIAG	BOOL	FALSE	1 = Diagnosepuffer geändert
22.3	OT_ERR	BOOL	FALSE	1 = Bedienfehler
22.4	DATA_ERR	BOOL	FALSE	1 = Datenfehler

6.2.6.3 FM 451 konfigurieren

6.2.6.4 FM 451 parametrieren

Kanäle spezifizieren

Maßsystem auswählen

Maßsystem auswählen

Sie können für die Eingabe und Ausgabe der Daten unter den folgenden Maßsystemen wählen:

- Millimeter (Voreinstellung)
- Inch
- Grad

Hinweis

Wenn Sie das Maßsystem im Dialog "Kanäle spezifizieren" verändern, dann werden die Werte in das neue System umgerechnet. Dabei können Rundungsfehler auftreten.

Wenn Sie das Maßsystem über das Maschinendatum ändern, werden die Werte nicht automatisch umgerechnet.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
8.0	UNITS	DINT	L#1	**Maßsystem** Zulässige Werte: • 1 = 10^{-3} mm • 2 = 10^{-4} inch • 3 = 10^{-4} grad • 4 = 10^{-2} grad • 6 = 10^{-3} grad

Standard-Maßsystem in der Online-Hilfe

Wir verwenden für die Angabe der minimalen und maximalen Werte das Maßsystem mm. Für die Bestimmung der Grenzwerte in den anderen Maßsystemen nehmen Sie deshalb folgende Umrechnung vor:

Für die Umrechnung von		berechnen Sie
mm → inch		Grenzwert (inch) = Grenzwert (mm) × 0,1 [1]
mm → grad	10^{-4} (4 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 0,1
	10^{-3} (3 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 1
	10^{-2} (2 Nachkommastellen)	Grenzwert (grad) = Grenzwert (mm) × 10
inch → grad		Grenzwert (grad) = Grenzwert (inch)

[1] Die Anzahl der Nachkommastellen hat beim Maximalwert Auswirkungen auf die Anzahl der Stellen vor dem Komma. Im Maßsystem Inch werden 4 Nachkommastellen verwendet, somit können maximal 100 000,0000 inch eingegeben werden. Das Maßsystem Millimeter verwendet 3 Nachkommastellen, hier können maximal 1 000 000,000 mm eingegeben werden.

Zusammenhang zwischen Inkrementen und Maßsystem

Die Gebersignale eines angeschlossenen Gebers werden von der Baugruppe ausgewertet und in das aktuelle Maßsystem umgerechnet. Für die Umrechnung wird die Auflösung (Seite 2633) verwendet.

Wenn die Positionierbaugruppe

- 10 Inkremente gezählt hat und
- eine Auflösung von 100 µm pro Inkrement durch die parametrierten Geberdaten eingestellt ist,

bedeutet das, dass die Achse um eine Strecke von 1 mm bewegt wurde.

Kanäle kopieren

Kanäle kopieren

Sie können die Parameter eines Kanals auf einen anderen Kanal kopieren.

1. Wählen Sie den Kanal aus, dessen Parameter kopiert werden sollen (Kanal Quelle).
2. Aktivieren Sie den Kanal, auf den die Parameter kopiert werden sollen (Kanal Ziel).
3. Drücken Sie die Schaltfläche "Kopieren".

Kanal n Antrieb

Ansteuerart auswählen

Übersicht der Ansteuerarten

Übersicht

Die Ansteuerart legt fest, wie ein Motor an den 4 Digitalausgängen (Q) eines Kanals über das Leistungsteil betrieben wird. Für die Positionierbaugruppe gibt es 4 Ansteuerarten je Kanal:

- Ansteuerart 1 (Seite 2614)
- Ansteuerart 2 (Seite 2614)
- Ansteuerart 3 (Seite 2615)
- Ansteuerart 4 (Seite 2616)

Die folgende Tabelle zeigt die Zustände der 4 Digitalausgänge für jede Ansteuerart, x steht für Kanal 1, 2 und 3.

	Eilgang		Schleichgang		PEH Position erreicht Halt
	Richtung +	Richtung -	Richtung +	Richtung -	
Ansteuerart 1					
xQ0	1	1	0	0	-
xQ1	0	0	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Ansteuerart 2					
xQ0	1	1	0	0	0
xQ1	0	0	0	0	1
xQ2	1	0	1	0	0
xQ3	0	1	0	1	0
Ansteuerart 3					
xQ0	1	1	0	0	-
xQ1	1	1	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Ansteuerart 4					
xQ0	1	0	0	0	-
xQ1	1	0	1	0	-
xQ2	0	1	0	0	-
xQ3	0	1	0	1	-

Ansteuerart 1

Ansteuerart 1

Bei Ansteuerart 1 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang
- xQ1: Schleichgang
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 1.

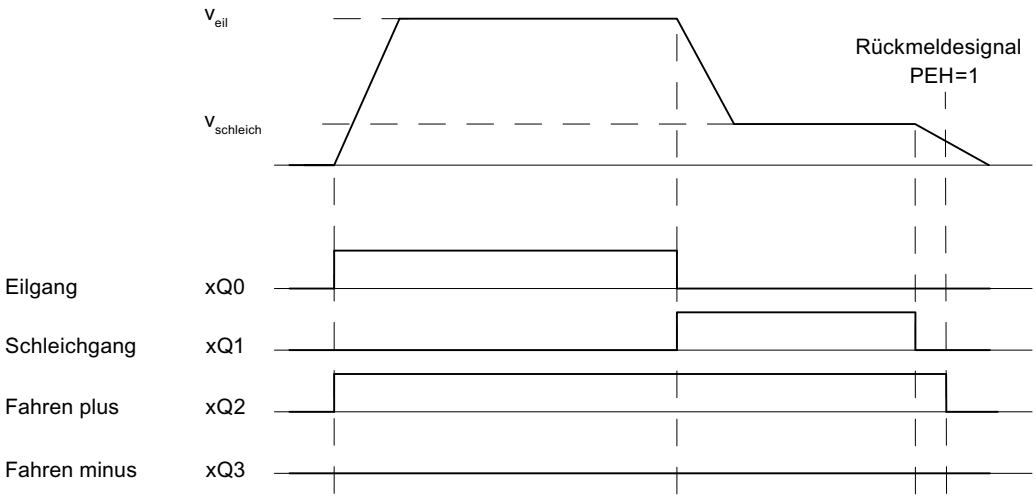

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	1 = Ansteuerart 1

Ansteuerart 2

Ansteuerart 2

Bei Ansteuerart 2 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eil-/Schleichgang
- xQ1: Position erreicht
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 2.

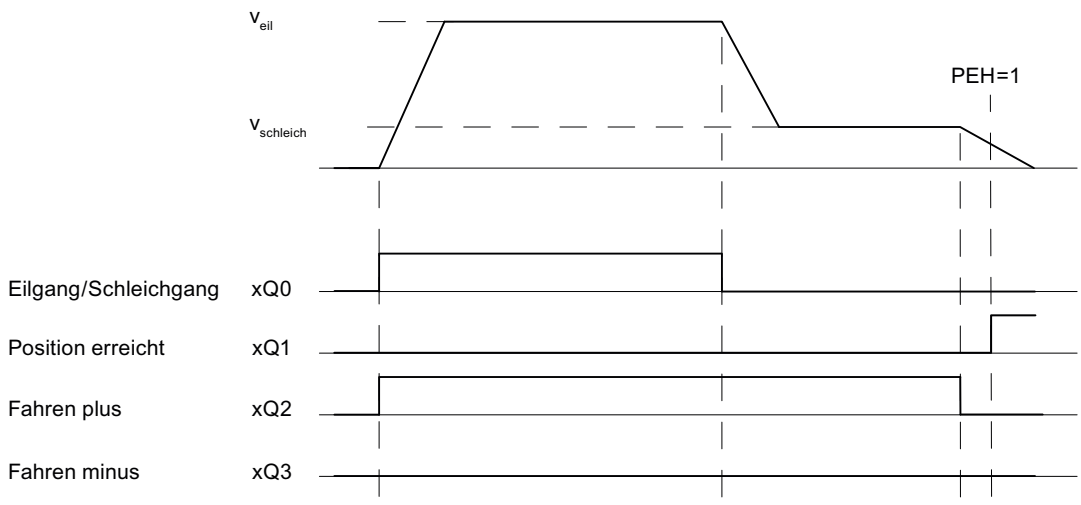

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	2 = Ansteuerart 2

Ansteuerart 3

Ansteuerart 3

Bei Ansteuerart 3 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang
- xQ1: Schleichgang
- xQ2: Fahren plus
- xQ3: Fahren minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 3.

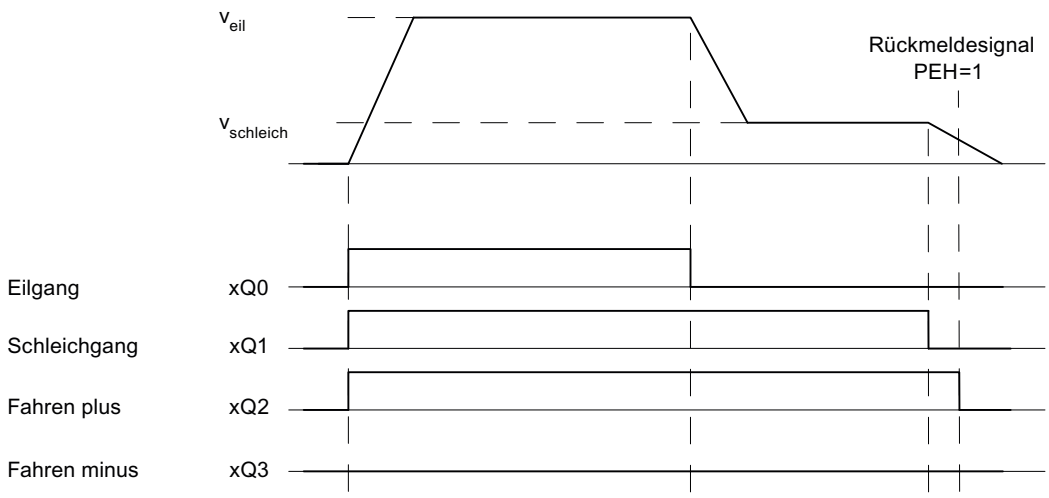

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	3 = Ansteuerart 3

Ansteuerart 4

Ansteuerart 4

Bei Ansteuerart 4 gelten folgende Einstellungen (x steht für Kanal n):

- xQ0: Eilgang plus
- xQ1: Schleichgang plus
- xQ2: Eilgang minus
- xQ3: Schleichgang minus

Das folgende Bild zeigt die Funktion der Digitalausgänge bei der Ansteuerart 4.

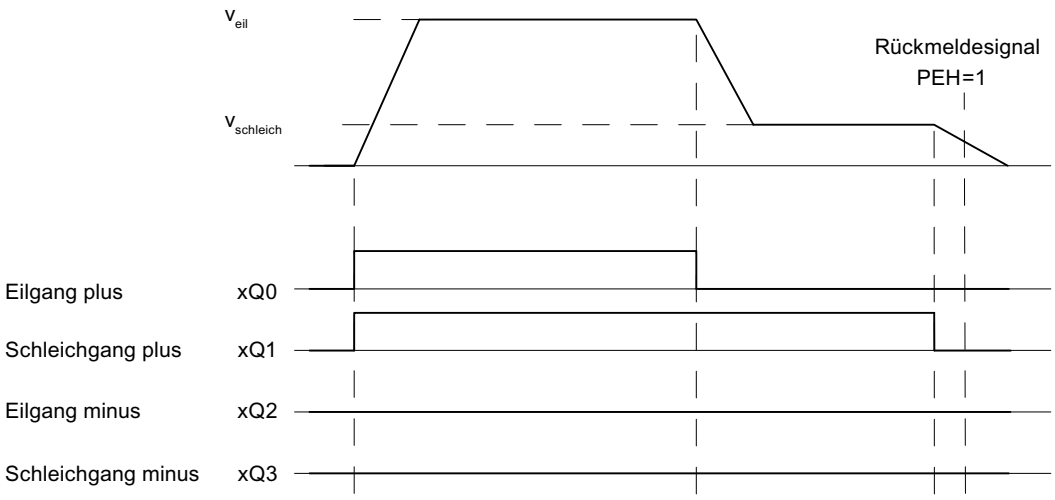

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	CTRL_TYPE	DINT	L#1	4 = Ansteuerart 4

Umschaltdifferenz festlegen

Umschaltdifferenz

Die "Umschaltdifferenz" definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang umgeschaltet wird.

Die vorgegebenen Werte gelten für alle Ziele, die die Positionierbaugruppe anfährt. Ausnahme ist eine Schrittmaßfahrt mit dem Schrittmaß 255, dafür wird eine gesonderte Umschaltdifferenz festgelegt.

Regeln für die Umschaltdifferenz

- Die Umschaltdifferenz plus und die Umschaltdifferenz minus müssen nicht symmetrisch um das Ziel angeordnet sein. Die Werte können sich unterscheiden.
- Die Umschaltdifferenz muss größer als die Abschaltdifferenz sein.
- Die Umschaltdifferenz muss innerhalb des Arbeitsbereichs liegen.
- Die Umschaltdifferenz muss kleiner als das Ende der Rundachse sein.
- Der Abstand zwischen dem Umschaltpunkt und dem Abschaltpunkt muss so groß gewählt werden, dass der Antrieb sicher vom Eilgang in den Schleichgang schalten kann.
- Der Umschaltpunkt und der Abschaltpunkt müssen einen zeitlichen Abstand von mindestens 8 ms voneinander haben.

Das folgende Bild zeigt die Lage der Schaltbereiche um das Ziel.

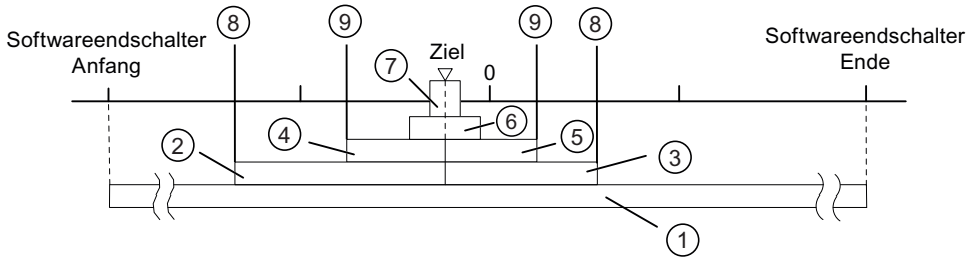

① Arbeitsbereich
② Umschaltdifferenz plus / minus
③
④ Abschaltdifferenz plus / minus
⑤
⑥ Stillstandsbereich
⑦ Zielbereich
⑧ Umschaltpunkt
⑨ Abschaltpunkt

Weitere Informationen über die Anordnung der Bereiche finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CHGDIF_P	DINT	L#5000	**Umschaltdifferenz plus**
104.0	CHGDIF_M	DINT	L#5000	**Umschaltdifferenz minus** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Abschaltdifferenz festlegen

Abschaltdifferenz

Die "Abschaltdifferenz" definiert den Abstand zum Ziel, an dem der Antrieb (Schleichgang) abgeschaltet wird. Die Überwachungsfunktionen übernimmt ab diesem Punkt die Positionierbaugruppe.

Die vorgegebenen Werte gelten für alle Ziele, die die Positionierbaugruppe anfährt. Ausnahme ist eine Schrittmaßfahrt mit dem Schrittmaß 255, dafür wird eine gesonderte Abschaltdifferenz festgelegt.

Regeln für die Abschaltdifferenz

- Die Abschaltdifferenz plus und die Abschaltdifferenz minus müssen nicht symmetrisch um das Ziel angeordnet sein. Die Werte können sich unterscheiden.
- Die Abschaltdifferenz muss größer als der halbe Zielbereich sein.

- Der Abstand zwischen Abschaltpunkt und Ziel muss so gewählt werden, dass der Antrieb innerhalb des Zielbereichs zum Stillstand kommt.
- Der Abschaltpunkt und der Beginn des Zielbereichs müssen einen zeitlichen Abstand von mindestens 8 ms voneinander haben.

Das folgende Bild zeigt die Lage der Schaltbereiche um das Ziel.

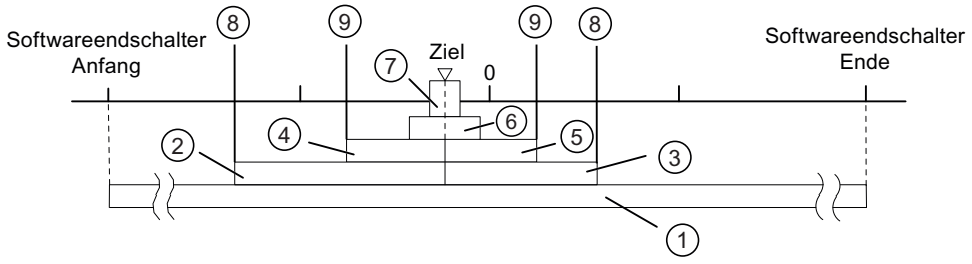

① Arbeitsbereich
② Umschaltdifferenz plus / minus
③
④ Abschaltdifferenz plus / minus
⑤
⑥ Stillstandsbereich
⑦ Zielbereich
⑧ Umschaltpunkt
⑨ Abschaltpunkt

Weitere Informationen über die Anordnung der Bereiche finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
108.0	CUTDIF_P	DINT	L#2000	**Abschaltdifferenz plus**
112.0	CUTDIF_M	DINT	L#2000	**Abschaltdifferenz minus** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/ Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/ Impuls

Zieleinlauf festlegen

Zielbereich festlegen

Zielbereich

Die Positionierbaugruppe legt um jedes Ziel einen symmetrischen Bereich, den Zielbereich. Innerhalb dieses Bereichs muss bei einem Zieleinlauf der Istwert liegen.

Eine Wertvorgabe "0" schaltet die Toleranz beim Zieleinlauf ab. Die Geschwindigkeit bei Erreichen des Zieles muss dann kleiner sein als die parametrierte Stillstandsgeschwindigkeit (Seite 2622).

Regeln für den Zielbereich

- Der Zielbereich liegt symmetrisch um das Ziel.
- Eine Wertvorgabe "0" schaltet die Überwachung des Zielbereichs ab.

Das folgende Bild zeigt die Lage der Bereiche um das Ziel.

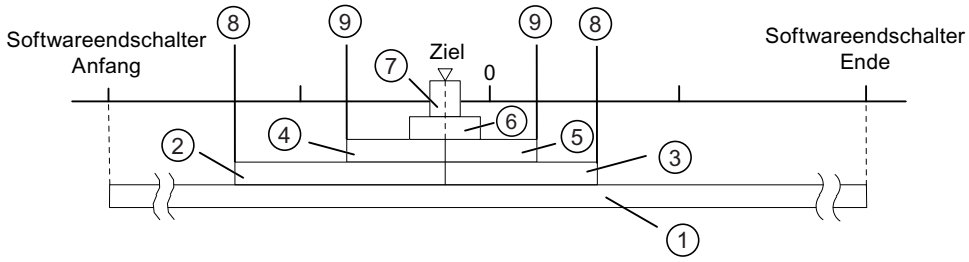

①	Arbeitsbereich	⑦	Zielbereich
② ③	Umschaltdifferenz plus / minus	⑧	Umschaltpunkt
④ ⑤	Abschaltdifferenz plus / minus	⑨	Abschaltpunkt
⑥	Stillstandsbereich		

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe" sowie unter "Ende eines Positioniervorgangs (Seite 2574)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
76.0	TRG_RANGE	DINT	L#1000	**Zielbereich** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/ Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/ Impuls

Überwachungszeit festlegen

Überwachungszeit

Während der Überwachungszeit überwacht die Baugruppe:

- die Bewegung der Achse bis zum Abschaltpunkt.
  Die Überwachungszeit startet mit dem Beginn einer Positionierung und wird bei jeder Istwertänderung in Fahrtrichtung nachgetriggert.
  Innerhalb der Überwachungszeit muss sich der Istwert mindestens um ein Inkrement (das Wegstück der Auflösung) verändert haben.
- den Zieleinlauf.
  Die Überwachungszeit wird mit dem Erreichen des Abschaltpunkts zum letzen Mal nachgetriggert.
  Innerhalb der Überwachungszeit muss der Zielbereich erreicht werden.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Ende eines Positioniervorgangs (Seite 2574)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
80.0	MON_TIME	DINT	L#2000	**Überwachungszeit** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 bis 100 000 ms
Tatsächliche Überwachungszeit Sie können für die Überwachungszeit alle Werte aus dem Wertebereich vorgeben. Die Baugruppe rundet die vorgegebene Zeit auf ein Vielfaches von 8 ms (Baugruppenzyklus) auf.				

Stillstandsbereich festlegen

Stillstandsbereich

Der Stillstandsbereich liegt symmetrisch um das jeweilige Ziel. In diesem Bereich wird überwacht, ob der Antrieb auf der angefahrenen Zielposition stehenbleibt oder von ihr wegdriftet. Die Überwachung beginnt, nachdem die Positionierbaugruppe "Position erreicht" (PEH) gemeldet hat.

Wenn der Stillstandsbereich ohne einen gültigen Fahrauftrag verlassen wird, meldet die Positionierbaugruppe einen Fehler.

Regeln für den Stillstandsbereich

- Der Stillstandsbereich liegt symmetrisch um das jeweilige Ziel:
  (Ziel – ½ × Stillstandsbereich) ≤ Ziel ≤ (Ziel + ½ × Stillstandsbereich)
- Der Stillstandsbereich muss größer oder gleich dem Zielbereich sein.

- Die Koordinatenwerte für den Stillstandsbereich müssen im Arbeitsbereich liegen.
- Mit dem Wert "0" ist die Stillstandsüberwachung ausgeschaltet.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Bereiche und Schaltpunkte der Positionierbaugruppe" sowie unter "Ende eines Positioniervorgangs (Seite 2574)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
84.0	ZSPEED_R	DINT	L#1000	**Stillstandsbereich** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Stillstandsgeschwindigkeit festlegen

Stillstandsgeschwindigkeit

Die Stillstandsgeschwindigkeit dient als Referenzgeschwindigkeit. Erreicht die Istgeschwindigkeit während einer Positionierung die Stillstandsgeschwindigkeit und befindet sich der Antrieb im Zielbereich, dann wird "Position erreicht" (PEH) gemeldet. Der Positioniervorgang ist damit erfolgreich beendet.

Die Genauigkeit der Positionierung steigt, wenn der Wert der Stillstandsgeschwindigkeit herabgesetzt wird.
Mit dem Wert "0" ist die Stillstandsüberwachung ausgeschaltet.

Weitere Informationen zum Thema Zieleinlauf finden Sie unter "Ende eines Positioniervorgangs (Seite 2574)".

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
88.0	ZSPEED_L	DINT	L#30000	**Stillstandsgeschwindigkeit** Zulässiger Wertebereich: • 0 = keine Überwachung • 1 µm/min bis 100 000 µm/min

Kanal n Achse

Achsart festlegen

Achsart auswählen

Achsarten

Die Positionierbaugruppe kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

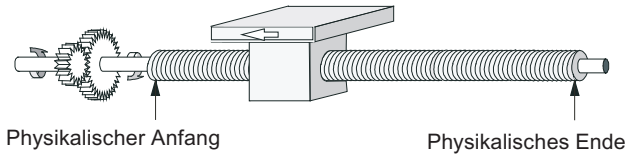

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

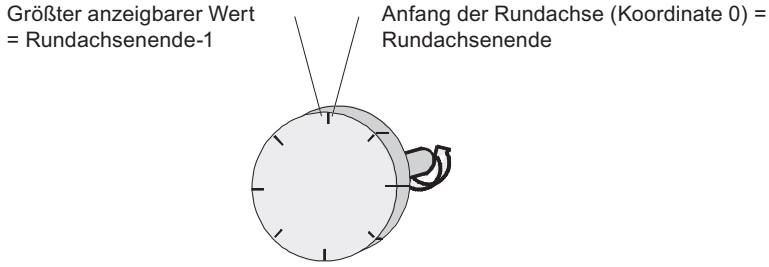

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
12.0	AXIS_TYPE	DINT	L#0	**Achsart** Zulässige Werte: • 0 = Linearachse • 1 = Rundachse

Software-Endschalter festlegen

Software-Endschalter

Software-Endschalter gibt es nur für die Linearachse.

Der Bereich, der durch die Software-Endschalter begrenzt wird, ist der Arbeitsbereich.

Der Software-Endschalter Anfang (SEA) muss immer kleiner sein als der Software-Endschalter Ende (SEE).

Die Software-Endschalter werden überwacht, nachdem die Achse synchronisiert ist.

- Bei Verwendung eines Inkrementalgebers:
  Die Synchronisation der Achse wird erreicht durch Referenzpunktfahrt oder Bezugspunkt setzen.
- Bei Verwendung eines Absolutwertgebers (SSI):
  Die Achse ist synchronisiert, nachdem die Baugruppe ein vollständiges, fehlerfreies Telegramm für den zugehörigen Kanal empfangen hat.
  Der Absolutwertgeber muss mindestens den Arbeitsbereich inklusive der Software-Endschalter abdecken.

Zusammenhang zwischen Arbeitsbereich, Geberbereich und Verfahrbereich

- Der Arbeitsbereich ist der Bereich, den Sie für Ihre Aufgabe durch die Software-Endschalter bestimmen.
- Der Geberbereich ist der vom Geber eindeutig abgedeckte Bereich. Der Geberbereich wird bei einer Linearachse von der Baugruppe symmetrisch über den Arbeitsbereich gelegt. D. h., die Baugruppe verschiebt den Geberbereich so, dass die Abstände zwischen den Software-Endschaltern und den Enden des Geberbereichs jeweils gleich sind (siehe Bild unten).
- Der Verfahrbereich ist der Wertebereich, den die Positionierbaugruppe verarbeiten kann. Der Verfahrbereich ist abhängig von der Auflösung.

Das folgende Bild zeigt den Zusammenhang zwischen Software-Endschaltern, Arbeitsbereich, Geberbereich und Verfahrbereich.

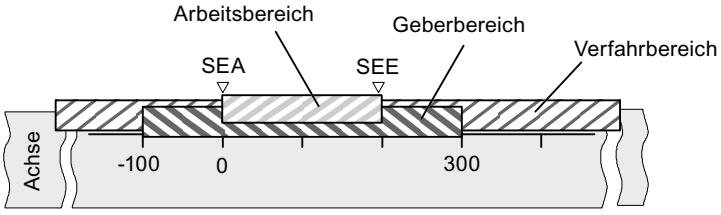

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
64.0	SSW_STRT	DINT	L#-100000000	**Software-Endschalter Anfang**
68.0	SSW_END	DINT	L#100000000	**Software-Endschalter Ende** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Ende der Rundachse festlegen

Ende der Rundachse

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Der größte Wert, der bei einer Rundachse angezeigt wird, hat den Wert:

Ende der Rundachse [µm] - Auflösung [µm/Impuls] × 1 [Impuls]

Beispiel: Ende der Rundachse 1000 mm, Auflösung 1000 µm/Impuls

Die Anzeige springt:

- bei positiver Drehrichtung von 999 mm auf 0 mm
- bei negativer Drehrichtung von 0 mm auf 999 mm.

Rundachse mit Absolutwertgebern

Bei einer Rundachse mit einem Absolutwertgeber muss sich der Rundachsenbereich (0 bis Ende der Rundachse) genau mit dem Geberbereich des Absolutwertgebers decken:
Ende der Rundachse [µm] = Anzahl Geberumdrehungen × Weg/Geberumdrehung [µm]

Verendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
16.0	ENDROTAX	DINT	L#100000	**Ende der Rundachse** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 1 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Referenzpunktfahrt festlegen

Art der Referenzpunktfahrt auswählen

Art der Referenzpunktfahrt

Mit "Art der Referenzpunktfahrt" bestimmen Sie die Bedingungen für die Synchronisation der Achse:

- Die erste Angabe definiert die Startrichtung, in die die Referenzpunktfahrt startet.
- Die zweite Angabe definiert die Lage der Nullmarke, die zur Synchronisation führt, in Bezug auf den Referenzpunktschalter.

Detaillierte Informationen finden Sie unter Betriebsart "Referenzpunktfahrt" (Seite 2583).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
52.0	REFPT_TYPE	DINT	L#0	**Art der Referenzpunktfahrt** Zulässige Werte: • 0 = plus, Referenzpunktschalter in Richtung + • 1 = plus, Referenzpunktschalter in Richtung - • 2 = minus, Referenzpunktschalter in Richtung + • 3 = minus, Referenzpunktschalter in Richtung -

Referenzpunktkoordinate festlegen

Referenzpunktkoordinate

- **Inkrementalgeber:**
  Die Referenzpunktkoordinate benötigen Sie für die Betriebsart "Referenzpunktfahrt". Wenn nach dem Schreiben und Aktivieren von Maschinendaten die Achse nicht synchronisiert ist, wird der Istwert auf den Wert der Referenzpunktkoordinate gesetzt.
- **Absolutwertgeber (SSI)**
  Die Referenzpunktkoordinate benötigen Sie für die mechanische Justage des Gebers. Lesen Sie hierzu auch die Beschreibung der Absolutwertgeberjustage (Seite 2627), die das Zusammenspiel der Absolutwertgeberjustage mit den anderen Daten beschreibt.

Der Wert der Referenzpunktkoordinate muss innerhalb des Arbeitsbereichs liegen:

- **Linearachse**
  Einschließlich der Software-Endschalter.
- **Rundachse**
  Größer oder gleich 0 und kleiner als der Wert "Ende der Rundachse"
  (0 ≤ Referenzpunktkoordinate < Ende der Rundachse).

Verwendete Daten in Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Absolutwertgeberjustage festlegen

Absolutwertgeberjustage ermitteln

Absolutwertgeberjustage

Absolutwertgeberjustage ist nur für einen Absolutwertgeber (SSI) erforderlich.

Mit der Absolutwertgeberjustage und der Referenzpunktkoordinate wird der Wertebereich des Gebers eindeutig auf das Koordinatensystem der Achse abgebildet.

Weitere Informationen finden Sie unter "Vorgehen für eine Absolutwertgeberjustage (Seite 2627)" sowie in einem Beispiel (Seite 2628).

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	REFPT	DINT	L#0	**Referenzpunktkoordinate** Zulässiger Wertebereich: • -1 000 000 000 µm bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • -100 000 000 µm bis 100 000 000 µm bei Auflösung < 1 µm/Impuls
48.0	ENC_ADJ	DINT	L#0	**Absolutwertgeberjustage** Zulässiger Wertebereich: • 0 bis (2^{25} - 1)

Vorgehen für eine Absolutwertgeberjustage

Richtige Absolutwertgeberjustage ermitteln

Nach dem ersten Parametrieren müssen Sie weitere Schritte ausführen, damit eine korrekte Beziehung zwischen Geber und Koordinatensystem hergestellt wird. Der Ablauf ist unter Verwendung der Parametrierdialoge dargestellt.

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Ende" sein.
2. Rufen Sie den Auftrag "Bezugspunkt setzen (Seite 2601)" mit der Koordinate des unter 1. definierten Punktes auf.
   Die Positionierbaugruppe ermittelt nun für die im Kanal-DB eingetragene Referenzpunktkoordinate (REFPT im Kanal-DB) einen Geberwert, die Absolutwertgeberjustage. Diesen Wert können Sie mit dem Auftrag "Geberdaten lesen (Seite 2608)" auslesen.
3. Tragen Sie den ausgelesenen Wert in das Feld "Absolutwertgeberjustage" ein.
4. Speichern Sie Ihre Parametrierung mit der Exportfunktion in den entsprechenden Parameter-DB.

5. Laden Sie die Daten in HWCN zur CPU.
6. Damit die Daten aktiv werden, führen Sie einen Neustart der CPU durch.

Hinweis

Diesen Abgleich führen Sie bei der Inbetriebnahme einmal durch. Die Positionierbaugruppe ist nach einer Parametrierung im Anlauf synchronisiert, sobald ein vollständiges, fehlerfreies Telegramm des Gebers nach dem Anlauf empfangen wurde.

Alternative: Mechanische Justage eines Gebers

Eine korrekte Beziehung zwischen dem Koordinatensystem und dem Geber erreichen Sie auch folgendermaßen:

1. Fahren Sie die Achse an einen definierten reproduzierbaren Punkt, dem eine eindeutige Koordinate zugeordnet ist.
   Das kann z. B. der "Software-Endschalter Anfang" sein.
2. Tragen Sie diesen Koordinatenwert in den Maschinendaten als "Referenzpunktkoordinate" ein.
3. Lesen Sie den an dieser Position angezeigten Geberwert mit dem Auftrag "Geberdaten lesen (Seite 2608)" aus.
4. Tragen Sie diesen Wert als "Absolutwertgeberjustage" in die Maschinendaten ein.

Nach der Parametrierung wird dann immer ein korrekter Istwert angezeigt.

Anstelle der Schritte 3 und 4 können Sie auch den Geber über "Reset", falls vorhanden, auf Null setzen. Dann tragen Sie den Wert "0" als Absolutwertgeberjustage in die Maschinendaten ein.

Beispiel für eine Absolutwertgeberjustage

Beispiel einer Absolutwertgeberjustage

Für das Beispiel gelten folgende Annahmen:

- Referenzpunktkoordinate = -125 mm
- Arbeitsbereich von SSW_STRT = -1000 mm bis SSW_END = 1000 mm
- Absolutwertgeberjustage = 0
- Geberbereich = 2048 Inkremente mit einer Auflösung von 1 mm/Impuls
- Der verwendete Absolutwertgeber lässt sich nicht exakt mechanisch justieren und besitzt auch nicht die Möglichkeit, den Geberwert gezielt zu setzen.

Das folgende Bild zeigt ein Beispiel einer Absolutwertgeberjustage.

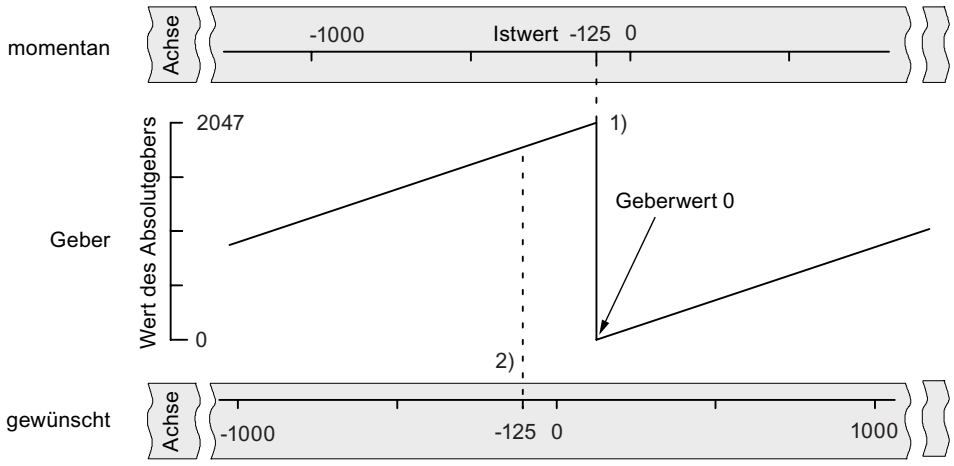

1) Zuordnung des Koordinatensystems zu den Geberwerten mit der eingestellten Absolutwertgeberjustage. Der Geberwert 0 entspricht dem Istwert -125.

2) Gewünschte Zuordnung des Koordinatensystems zum Geber. An dieser Position soll die Koordinate -125 sein.

Ergebnis nach "Bezugspunkt setzen"

Nach "Bezugspunkt setzen" sieht die Beziehung wie folgt aus:

Der Referenzpunktkoordinate auf der Achse (-125) wird der aus der Absolutwertgeberjustage ermittelte Geberwert (1798) zugeordnet. Das folgende Bild zeigt diese Zuordnung.

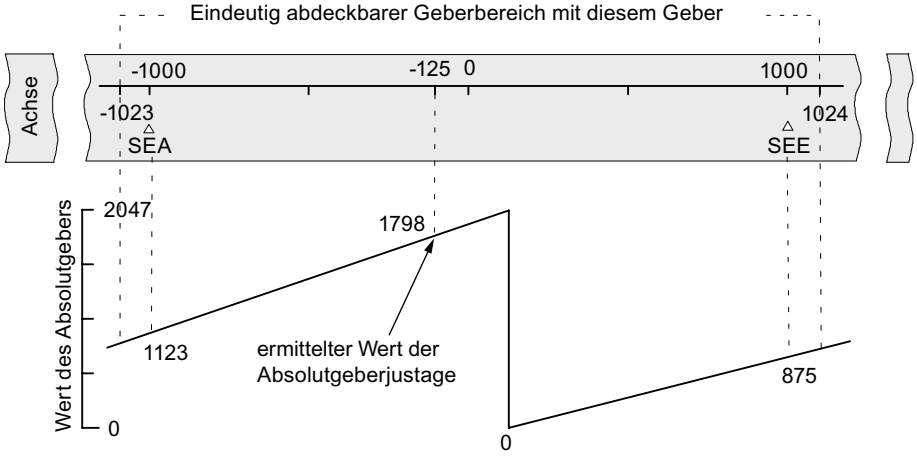

Der Geber liefert 2048 eindeutige Werte. Der Arbeitsbereich wird durch die Software-Endschalter festgelegt. Aufgrund der gewählten Auflösung von 1 mm pro Impuls kann der Geber aber einen größeren Arbeitsbereich abdecken als durch die Software-Endschalter vorgesehen ist.

Bei der eingestellten Auflösung ist der Arbeitsbereich bereits mit 2001 Werten abgedeckt. Es bleiben im Beispiel deshalb 47 Impulse "übrig", die sich symmetrisch um den Arbeitsbereich legen.

Startgeschwindigkeit auswählen

Startgeschwindigkeit bei Referenzpunktfahrt

Mit diesem Datum wählen Sie die Geschwindigkeit für den Start einer Referenzpunktfahrt.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
99.0	REFPT_SPD	BOOL	TRUE	**Startgeschwindigkeit bei Referenzpunktfahrt** Zulässige Werte: • 0 = Eilgang • 1 = Schleichgang

Minimalen Kantenabstand festlegen

Minimaler Kantenabstand

Mit dem Maschinendatum "Minimaler Kantenabstand" definieren Sie einen Bereich nach dem Erkennen des Messbeginns bei einer Kantenerfassung. Wenn das Ende des Messvorgangs innerhalb dieses Bereichs liegt, wird die Messung verworfen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.0	EDGEDIST	DINT	L#0	**Minimaler Kantenabstand** Zulässiger Wertebereich: • 0 bis 1 000 000 000 µm bei Auflösung ≥ 1 µm/Impuls • 0 bis 100 000 000 µm bei Auflösung < 1 µm/Impuls

Kanal n Geber

Signalart auswählen

Signalart

Der Geber liefert Weginformationen zur Baugruppe, die diese auswertet und mit der Auflösung in einen Istwert umrechnet.

Sie können die Signalart von einem der folgenden Geber wählen:

- 5 V-Inkrementalgeber
  Wählen Sie die Signalart "5 V Differenzsignal", wenn Sie einen 5 V-Inkrementalgeber mit den Differenzsignalen A, /A, B, /B und N, /N einsetzen.
- 24 V-Inkrementalgeber
  Wählen Sie die Signalart "24 V Spur A+B phasenversetzt", wenn Sie einen 24 V-Inkrementalgeber mit den Signalen A*, B* und N* einsetzen.
- Absolutwertgeber (SSI)
  Wählen Sie die Signalart "SSI absolut", wenn Sie einen Absolutwertgeber mit serieller Schnittstelle einsetzen.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
20.0	ENC_TYPE	DINT	L#1	**Signalart** Zulässige Werte: • 1 = 5 V Differenzsignal • 2 = 24 V Spur A+B phasenversetzt • 3 = SSI 13 Bit Telegrammlänge [1)] • 4 = SSI 25 Bit Telegrammlänge [1)] • 5 = SSI 14 Bit Telegrammlänge [1)] • 6 = SSI 15 Bit Telegrammlänge [1)] • 7 = SSI 16 Bit Telegrammlänge [1)] • : • : • 14 = SSI 23 Bit Telegrammlänge [1)] • 15 = SSI 24 Bit Telegrammlänge [1)]

[1)] Die Auswahl der Telegrammlänge erfolgt im Dialogfeld "SSI".

Zählrichtung auswählen

Zählrichtung

Mit dem Maschinendatum "Zählrichtung" passen Sie die Richtung der Wegerfassung der Bewegungsrichtung der Achse an. Berücksichtigen Sie dabei auch alle Drehrichtungen der Übertragungsglieder (wie z. B. Kupplungen und Getriebe).

- normal = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen aufsteigenden Lageistwerten
- invertiert = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen absteigenden Lageistwerten

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
59.0	CNT_DIR	BOOL	FALSE	**Zählrichtung** Zulässige Werte: • 0 = normal • 1 = invertiert

Auflösung festlegen

Weg pro Geberumdrehung festlegen

Weg pro Geberumdrehung

Mit dem Maschinendatum "Weg pro Geberumdrehung" geben Sie der Positionierbaugruppe bekannt, welchen Weg das Antriebssystem je Geberumdrehung zurücklegt.

Der Wert "Weg pro Geberumdrehung" ist abhängig vom Aufbau der Achse und von der Art, wie der Geber angebracht ist. Sie müssen alle Übertragungsglieder wie Kupplungen oder Getriebe berücksichtigen. Das folgende Bild zeigt beispielhaft zwei unterschiedliche Geberanordnungen.

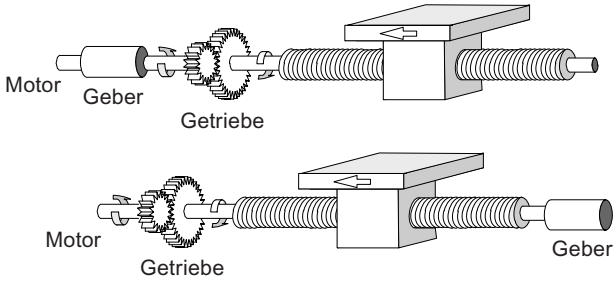

Der Zusammenhang zwischen den Maschinendaten "Weg pro Geberumdrehung" und "Inkremente pro Geberumdrehung" ist unter Auflösung (Seite 2633) beschrieben.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
24.0	DISP_REV	DINT	L#80000	**Weg pro Geberumdrehung** Zulässiger Wertebereich: • 1 µm bis 1 000 000 000 µm

Inkremente pro Geberumdrehung festlegen

Inkremente pro Geberumdrehung

Das Maschinendatum "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Geber je Umdrehung abgibt. Aus diesem Wert und dem Maschinendatum "Weg pro Geberumdrehung" ermittelt die Positionierbaugruppe die Auflösung (Seite 2633).

Inkrementalgeber

Als Eingabe ist jeder Wert aus dem Wertebereich möglich. Die Baugruppe wertet die Inkremente 4fach aus.

Absolutwertgeber

Für die Grenzen muss zwischen den einzelnen Gebermodellen unterschieden werden. Als Eingabe sind nur Werte in 2er Potenzen erlaubt.

- Singleturn-Geber (Anzahl Umdrehungen = 1) mit 13 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192
- Multiturn-Geber (Anzahl Umdrehungen > 1) mit 25 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192
- Singleturn-Geber (Anzahl Umdrehungen = 1) mit 25 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 2^{25}

Linearmaßstäbe werden als Multiturn-Geber wie folgt parametriert:

- Inkremente pro Geberumdrehung = 8192
- Anzahl Umdrehungen × 8192 ≥ Anzahl der Schritte des Linearmaßstabs

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
32.0	INC_REV	DINT	L#500	**Inkremente pro Geberumdrehung** Zulässiger Wertebereich: • 1 bis 2^{25}

Auflösung

Auflösung

Die Auflösung gibt an, welchem Verfahrweg ein Impuls entspricht. Sie ist ein Maß für die Genauigkeit der Positionierung und bestimmt auch den möglichen maximalen Verfahrbereich der Positionierbaugruppe.

Die Auflösung (AUFL) berechnen Sie wie in der folgenden Tabelle gezeigt:

	Inkrementalgeber	Absolutwertgeber
Eingangs-werte	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – 4fach-Auswertung – 1 Inkrement = 4 Impulse	• Weg pro Geberumdrehung • Inkremente pro Geberumdrehung: – 1 Inkrement = 1 Impuls
Berechnung	$\text{AUFL} = \dfrac{\dfrac{\text{Weg}}{\text{Geberumdrehung}}}{\dfrac{\text{Impulse}}{\text{Geberumdrehung}}}$	

Hinweis

Alle Positionsangaben werden auf das ganzzahlige Vielfache der Auflösung gerundet. Somit können sich die eingegebenen und die verwendeten Werte unterscheiden.

Wertebereich der Auflösung

Das gewählte Maßsystem bestimmt den Wertebereich der Auflösung:

Maßsystem	Angaben in ...	Wertebereich der Auflösung
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ … 1000×10^{-3} mm/Impuls
inch	10^{-4} inch	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} inch/Impuls
grad	10^{-4} grad 10^{-3} grad 10^{-2} grad	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} grad/Impuls $0{,}1 \times 10^{-3}$ … 1000×10^{-3} grad/Impuls $0{,}1 \times 10^{-2}$ … 1000×10^{-2} grad/Impuls

Beispiel

- Ein Inkrementalgeber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 5000
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 4 Impulse

  Damit ergibt sich die Auflösung (4fach-Auswertung):

$$\text{Auflösung} = \frac{1000\ \text{mm}}{5000\ \text{Inkremente}} = 0{,}2000\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{Impulse}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impuls}}$$

- Ein SSI-Geber hat folgende Daten:
  - Inkremente pro Geberumdrehung: 4096
  - Weg pro Geberumdrehung: 1000 mm
  - 1 Inkrement = 1 Impuls

  Damit ergibt sich die Auflösung:

$$\text{Auflösung} = \frac{1000\ \text{mm}}{4096\ \text{Inkremente}} = 0{,}2441\ \frac{\text{mm}}{\text{Inkrement}} = 0{,}2441\ \frac{\text{mm}}{\text{Impuls}}$$

Abhängigkeit zwischen Verfahrbereich und Auflösung

Der Verfahrbereich wird begrenzt durch die Zahlendarstellung in der Positionierbaugruppe. Diese Zahlendarstellung variiert in Abhängigkeit der Auflösung. Achten Sie deshalb darauf, dass Sie bei Vorgaben immer innerhalb der zulässigen Grenzen liegen.

Der maximale Verfahrbereich ist in der folgenden Tabelle dargestellt:

Auflösung (AUFL) liegt im Bereich	Maximaler Verfahrbereich
0,1 µm/Impuls ≤ AUFL < 1 µm/Impuls	-10^8 µm bis 10^8 µm (-100 m bis +100 m)
1 µm/Impuls ≤ AUFL ≤ 1000 µm/Impuls	-10^9 µm bis 10^9 µm (-1000 m bis +1000 m)

Daten für Absolutwertgeber

Anzahl Geberumdrehungen

Anzahl Geberumdrehungen

Das Maschinendatum "Anzahl Geberumdrehungen" gibt es nur für Absolutwertgeber. Damit geben Sie die Anzahl der Umdrehungen an, die mit diesem Geber möglich sind.

Die **Gesamtschrittzahl des Gebers** ist kein Maschinendatum. Sie errechnet sich wie folgt:

Gesamtschrittzahl = Inkremente pro Geberumdrehung × Anzahl Umdrehungen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.0	NO_REV	DINT	L#1	**Anzahl Geberumdrehungen** Zulässiger Wertebereich: • 1 (Singleturn-Geber) • 2 bis 4096 in 2er Potenzen (Multiturn-Geber)

Baudrate

Baudrate

Mit dem Maschinendatum "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur Positionierbaugruppe.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
40.0	BAUDRATE	DINT	L#0	**Baudrate** Zulässige Werte: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Telegrammlänge

Telegrammlänge

Mit der "Telegrammlänge" bestimmen Sie den Taktrahmen, den die Positionierbaugruppe für Absolutwertgeber ausgibt.

Zulässig ist eine Telegrammlänge von 13 Bit bis 25 Bit.

Verwendete Daten im Parameter-DB

Die hier ausgewählte Telegrammlänge ist zusammen mit der Signalart unter dem Datum ENC_TYPE (Seite 2630) im Parameter-DB abgelegt.

Parität

Parität

Mit der "Parität" bestimmen Sie, ob für Absolutwertgeber keine, gerade oder ungerade Parität übertragen werden soll.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
28.0	PARITY	DINT	L#0	**Parität** Zulässige Werte: • 0 = keine • 1 = gerade Parität (even) • 2 = ungerade Parität (odd)

Überwachung auswählen

Überwachung

Drahtbruch

Mit dem Aktivieren der Überwachung überwacht die Positionierbaugruppe bei einem 5 V-Inkrementalgeber und einem Absolutwertgeber alle Leitungen. Die Überwachung erkennt:

- Drahtbruch
- Kurzschluss der einzelnen Leitungen
- Flankenabstand der Zählimpulse (auch bei 24 V-Inkrementalgeber)

Zur Überwachung bei einem 24 V-Inkrementalgeber müssen Sie eine Überwachungszeit MON_TIME (Seite 2621) > 0 einstellen.

Bei 5 V-Inkrementalgebern ohne Nullmarke müssen Sie entweder die Drahtbruch-Überwachung ausschalten oder die Signale N und /N extern verschalten.

Telegrammfehler

Die Baugruppe überwacht das Telegramm eines Absolutwertgebers (SSI) auf:

- Start- und Stoppbitfehler

Die Überwachung auf Telegrammfehler ist bei Absolutwertgebern (SSI) nicht abschaltbar.

Fehlimpulse (Inkrementalgeber)

Ein Inkrementalgeber muss zwischen zwei aufeinanderfolgenden Nullmarken immer die gleiche Anzahl Inkremente liefern.

Die Positionierbaugruppe überprüft, ob die Nullmarke eines Inkrementalgebers zum richtigen Geberstand auftritt.

Für Geber ohne Nullmarken müssen Sie die Überwachung der Fehlimpulse abschalten.

Ebenso müssen Sie die Drahtbruchüberwachung abschalten oder die Nullmarkeneingänge N und /N extern verschalten.

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
				Überwachungen:
63.0	MON_WIRE	BOOL	TRUE	1 = Drahtbruch
63.1	MON_FRAME	BOOL	TRUE	1 = Telegrammfehler (muss immer 1 sein)
63.2	MON_PULSE	BOOL	TRUE	1 = Fehlimpulse

Kanal n Schrittmaße

Schrittmaße festlegen

Definition

Schrittmaße sind Zielvorgaben, die von der Positionierbaugruppe mit der Betriebsart "Schrittmaßfahrt relativ" bzw. "Schrittmaßfahrt absolut" angesteuert werden können.

Möglicher Bereich für Schrittmaße

Das Ziel, das angefahren werden soll, muss mindestens um den halben Zielbereich vor dem jeweiligen Software-Endschalter liegen. Das folgende Bild zeigt die Grenzen für eine Schrittmaßvorgabe.

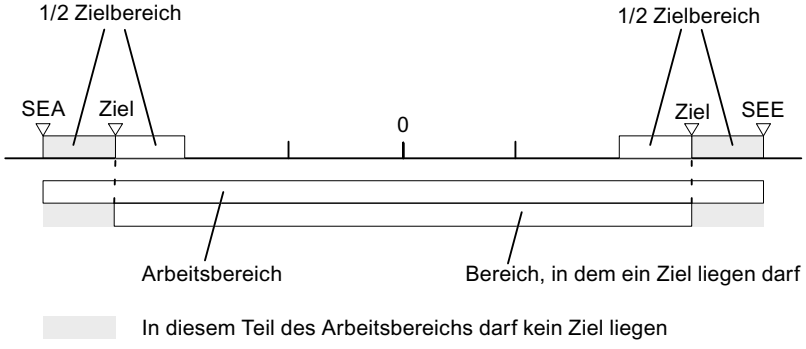

Schrittmaßnummer 1 bis 100

Schrittmaßnummer 1 bis 100

Sie haben die Möglichkeit, maximal 100 Schrittmaße in eine Tabelle einzutragen, die sowohl für die Betriebsart "Schrittmaßfahrt relativ" als auch für "Schrittmaßfahrt absolut" gültig sind.

Beachten Sie, dass die Positionierbaugruppe für die "Schrittmaßfahrt relativ" keine negativen Werte erlaubt. Die Werte werden von der Positionierbaugruppe je nach Bewegungsrichtung als positive oder als negative Differenz interpretiert.

Hinweis

Der Eintrag erfolgt in der Einheit des eingestellten Maßsystems. Beachten Sie dabei insbesondere die Nachkommastellen.

Zahlenbeispiel:

- Schrittmaß: 800 mm
- Maßsystem: 10^{-3} mm
- Eingabe in Parameter-DB: 800000

Empfehlung: Definieren Sie in der Schrittmaßtabelle getrennte Bereiche für relative und absolute Schrittmaße.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaße 1 ... 50) schreiben
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaße 51 ... 100) schreiben
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 1 (Schrittmaße 1 ... 50) lesen
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Schrittmaßtabelle 2 (Schrittmaße 51 ... 100) lesen

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
120.0	TRGL1.TRG[1]	DINT	L#0	Schrittmaßnummer 1	Schrittmaßtabelle 1
. . .	. . .	. .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Schrittmaßnummer 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Schrittmaßnummer 51	Schrittmaßtabelle 2
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Schrittmaßnummer 100	

Schrittmaßnummer 252

Schrittmaßnummer 252

Mit der Schrittmaßnummer 252 haben Sie die Möglichkeit, den Antrieb zunächst ohne Ziel in vorgegebener Richtung zu bewegen (Endlosfahren) und beim Eintreffen eines externen Ereignisses am Digitaleingang xI3 eine Positionierung auf das parametrierte Ziel (Schrittmaß) durchzuführen.

Für die Umschalt- und Abschaltdifferenzen haben die Einträge aus dem Parameter-DB für dieses Schrittmaß Gültigkeit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 252 oder 254

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus

Schrittmaßnummer 254

Schrittmaßnummer 254

Unabhängig von der Schrittmaßtabelle können Sie die Schrittmaßnummer 254 als eine weitere Vorgabe des Weges verwenden. Für die Umschalt- und Abschaltdifferenzen haben die Einträge aus dem Parameter-DB für dieses Schrittmaß Gültigkeit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.2	TRG252_254_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 252 oder 254 schreiben
96.0	TRG252_254	DINT	L#0	Schrittmaß für Schrittmaßnummer 252 oder 254

Verwendete Daten im Parameter-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	CHGDIF_P	DINT	L#5000	Umschaltdifferenz plus
104.0	CHGDIF_M	DINT	L#5000	Umschaltdifferenz minus
108.0	CUTDIF_P	DINT	L#2000	Abschaltdifferenz plus
112.0	CUTDIF_M	DINT	L#2000	Abschaltdifferenz minus

Schrittmaßnummer 255

Schrittmaßnummer 255

Mit der Schrittmaßnummer 255 steht Ihnen eine weitere Vorgabe des Weges zur Verfügung.

Die Abschaltdifferenzen und die Umschaltdifferenzen übergeben Sie zusammen mit dem Schrittmaß. Anders als bei den anderen Schrittmaßen verwendet das Schrittmaß 255 die im Kanal-DB festgelegten Werte für die Abschalt- und Umschaltdifferenz. Die Einträge aus den Maschinendaten haben für dieses Schrittmaß keine Gültigkeit.

Verwendete Daten im Kanal-DB

Adresse	Name	Typ	Anfangswert	Kommentar
36.3	TRG255_EN	BOOL	FALSE	1 = Schrittmaß für Schrittmaßnummer 255 schreiben
100.0	TRG255	DINT	L#0	Schrittmaß für Schrittmaßnummer 255
104.0	CHGDIF255	DINT	L#0	Umschaltdifferenzen für Schrittmaßnummer 255
108.0	CUTDIF255	DINT	L#0	Abschaltdifferenzen für Schrittmaßnummer 255

Steuer- und Rückmeldeschnittstelle der FM 451

Anwendung

In speziellen Anwendungen oder in einer Alarmebene kann ein besonders schneller Zugriff auf Rückmelde- und Steuersignale erforderlich sein. Diese Daten erreichen Sie direkt über die Ein- und Ausgangsbereiche der Baugruppe.

Zur Anlaufkoordinierung nach jedem Anlauf der Baugruppe, z. B. nach Baugruppe stecken oder nach einem STOP-RUN-Übergang der CPU, müssen Sie die Anweisung ABS_CTRL solange aufrufen, bis durch RETVAL = 0 das Ende des Anlaufs angezeigt wird. Anschließend dürfen Sie die Anweisung ABS_CTRL nicht mehr anwenden.

Hinweis

Die Verwendung der Anweisung ABS_CTRL zusammen mit einem Schreibzugriff ist nicht möglich.

Rückmeldesignale lesen durch Direktzugriff

Die Byte-Adressen sind relativ zur Anfangsadresse der Ausgänge des jeweiligen Kanals angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

Anfangsadresse Kanal 1 = Anfangsadresse Baugruppe

Anfangsadresse Kanal 2 = Anfangsadresse Baugruppe + 8

Anfangsadresse Kanal 3 = Anfangsadresse Baugruppe + 16

In AWL greifen Sie mit den Befehlen PEB (1 Byte lesen), PEW (2 Byte lesen) und PED (4 Byte lesen) auf die Daten zu.

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) der FM 451.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	PARA	intern	intern	DATA_ERR	OT_ERR	DIAG	intern	intern
Byte 1	CHGOVER	CUTOFF	ZSPEED	SPEED_OUT	0	WAIT_EI	WORKING	ST_ENBLD
Byte 2	MODE_OUT							
Byte 3	POS_RCD	0	FVAL_DONE	0	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								

Beispiel: Lageistwert ACT_POS

Die Anfangsadresse der Baugruppe ist 512.

AWL	Beschreibung
L PED 516	//Aktuellen Lageistwert (ACT_POS) von Kanal 1
	//mit Direktzugriff lesen:
	//Anfangsadresse des Kanals + 4

Steuersignale schreiben durch Direktzugriff

Die Byte-Adressen sind relativ zur Anfangsadresse der Eingänge des jeweiligen Kanals angegeben. Die Namen der Parameter entsprechen den Namen im Kanal-DB.

Anfangsadresse Kanal 1 = Anfangsadresse Baugruppe

Anfangsadresse Kanal 2 = Anfangsadresse Baugruppe + 8

Anfangsadresse Kanal 3 = Anfangsadresse Baugruppe + 16

In AWL greifen Sie mit den Befehlen PAB (1 Byte schreiben), PAW (2 Byte schreiben) und PAD (4 Byte schreiben) auf die Daten zu.

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) der FM 451.

Adresse	Bitnummer							
	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	OT_ERR_A	0	0	0
Byte 1	DRV_EN	SPEED252	0	0	DIR_P	DIR_M	STOP	START
Byte 2	MODE_IN							
Byte 3	MODE_TYPE							
Byte 4	Reserviert							
Byte 5								
Byte 6								
Byte 7								

Beispiel: START-Signale Kanal 2

Die Anfangsadresse der Baugruppe ist 512.

```
AWL                 Beschreibung
L 2#10001000        //DRV_EN und DIR_P auf 1 setzen
T PAB 521           //Signale mit Direktzugriff für Kanal 2 schreiben:
                    //Anfangsadresse der Baugruppe + 8 + 1
```

Parameterübertragungswege

Unter Parameter werden nachfolgend Maschinendaten und Schrittmaße verstanden.

Das folgende Bild zeigt die Parameterübertragungswege zur FM x51.

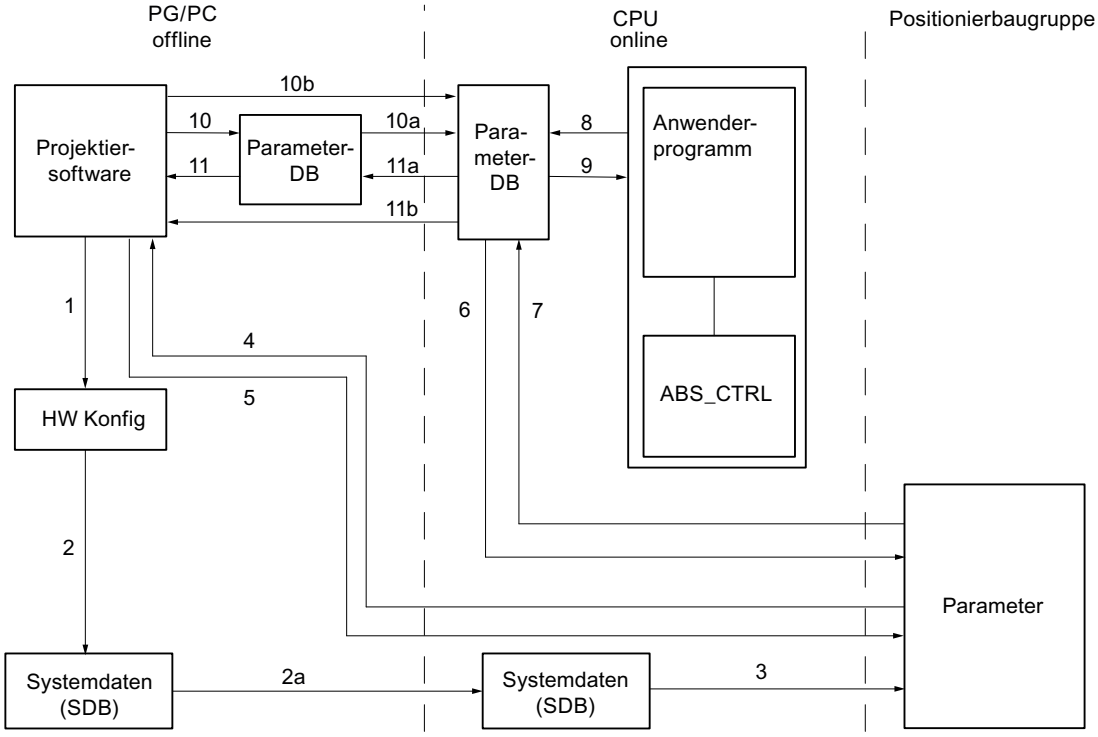

1 Parameter in der Projektiersoftware speichern.
2 HW-Konfiguration speichern und übersetzen.
2a HW-Konfiguration zur CPU laden. Die CPU führt automatisch Schritt 3 aus.
3 Die CPU schreibt die Parameter bei Systemparametrierungen zur Baugruppe.
4 Parameter eines Kanals der Baugruppe mit dem Befehl "Zielsystem laden in PG" ins PG laden.
5 Parameter aus der Projektiersoftware mit dem Befehl "Zielsystem laden" in einen Kanal der Baugruppe laden.
6 Parameter über Aufträge des Anwenderprogramms in einen Kanal der Baugruppe schreiben.
7 Parameter über Aufträge des Anwenderprogramms aus einem Kanal der Baugruppe lesen.
8 Parameter aus dem Anwenderprogramm in dem Online-DB ablegen.
9 Parameter aus dem Online-DB in das Anwenderprogramm einlesen.
10 Parameter aus der Projektiersoftware in den Offline-DB exportieren.
10a Offline-DB in die CPU laden.
10b Parameter aus der Projektiersoftware in den Online-DB exportieren.
11 Parameter aus dem Offline-DB in die Prokjektiersoftware importieren.
11a Parameter aus dem Online-DB in das PG laden.
11b Parameter aus dem Online-DB in die Projektiersoftware importieren.

Einige Anwendungsfälle für die Übertragung von Parametern

Anwendungsfall	Schritte
Sie bearbeiten die Parameter mit der Projektiersoftware. Die Kanäle der Baugruppe sollen anschließend automatisch beim Anlauf parametriert werden.	Führen Sie die Schritte 1, 2 und 2a aus.
Sie ändern Parameter bei der Inbetriebnahme im Testbetrieb in der Projektiersoftware.	Führen Sie die Schritte 4 und 5 aus.
Die bei der Inbetriebnahme veränderten Parameter sollen anschließend beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 1, 2 und 2a aus.
Sie erstellen die Parameter mit der Projektiersoftware. Die Kanäle der Baugruppe sollen beim Anlauf nur vom Anwenderprogramm über Datenbausteine parametriert werden.	Führen Sie die Schritte 10, 10a und 6 oder 10b und 6 aus.
Sie wollen komfortabel Vorratsdaten für Rezepturen erstellen.	Führen Sie die Schritte 10 und 10a aus.
Sie erstellen die Parameter mit der Projektiersoftware. Diese sollen dem Anwenderprogramm für temporäre Änderungen zur Verfügung stehen.	Führen Sie die Schritte 1, 2 und 2a für die automatische Parametrierung aus.
	Führen Sie die Schritte 10, 10a, 9, 8 und 6 für den Zugriff durch das Anwenderprogramm aus.
Sie ändern vorhandene Parameter ausschließlich über das Anwenderprogramm.	Führen Sie die Schritte 7, 9, 8 und 6 aus.
Sie wollen die über das Anwenderprogramm geänderten Daten mit der Projektiersoftware ansehen.	Führen Sie die Schritte 11a und 11 oder nur 11b aus.
Die vom Anwenderprogramm veränderten Parameter sollen auch beim Anlauf automatisch geladen werden.	Führen Sie die Schritte 11b oder 11a, 11 und dann 1, 2, 2a aus.

6.2.6.5 FM 451 aus dem Anwenderprogramm verwenden

Einbinden der FM 451 in ein Anwenderprogramm

Für das Einbinden der FM 451 in ein Anwenderprogramm werden Ihnen Anweisungen zur Verfügung gestellt, die Ihnen eine möglichst einfache Handhabung der gewünschten Funktionen ermöglichen.

Die Anweisungen ABS_INIT und ABS_CTRL bzw. ABS_CTRL_451 werden im Anwenderprogramm der CPU aufgerufen. Sie realisieren die Kommunikation zwischen der CPU und der FM 451.

Wenn Sie die Anweisung ABS_CTRL in das Projekt einbinden, dann wird der Kanal-DB automatisch als Instanz-DB zur Anweisung ABS_CTRL angelegt.
Wenn Sie die Anweisung ABS_CTRL_451 in das Projekt einbinden, dann müssen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ ABS_CHANTYPE als Kanal-DB in Ihr Projekt einfügen.

Alle zu einem Kanal der Baugruppe gehörenden Parameterdaten (Maschinendaten, Schrittmaßtabellen) liegen im Parameter-DB für die Baugruppe. Fügen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ ABS_PARATYPE in Ihr Projekt ein.

Außerdem gibt es für die FM 451 die Anweisung ABS_DIAG bzw. ABS_DIAG_451, mit der Sie Diagnoseinformationen in den Diagnose-DB übertragen können.

Wenn Sie die Anweisung ABS_DIAG in das Projekt einbinden, dann wird der Diagnose-DB automatisch als Instanz-DB zur Anweisung ABS_DIAG angelegt.
Wenn Sie die Anweisung ABS_DIAG_451 in das Projekt einbinden, dann müssen Sie über "Programmbausteine > Neuen Baustein hinzufügen" einen Datenbaustein vom Typ ABS_DIAGTYPE als Diagnose-DB in Ihr Projekt einfügen.

Inbetriebnahme der Baugruppe

Auch ohne Anwenderprogramm ist ein erster Test der Baugruppe mit dem Dialog "Inbetriebnahme" möglich. Beachten Sie dabei insbesondere die nachfolgende Warnung.

⚠ WARNUNG

Es kann zu Personen- und Sachschäden kommen.

Zur Vermeidung von Schäden an Personen und Gegenständen beachten Sie folgende Punkte:

- Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners.
  Nur so können Sie sicherstellen, dass im Falle eines Rechner- oder Software-Ausfalls die Anlage sicher ausgeschaltet werden kann.
- Installieren Sie Hardware-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
- Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.
- Durch paralleles Steuern und Beobachten der FM 451 von Ihrem Programm und aus dem Dialog "Inbetriebnahme" kann es zu Konflikten kommen, deren Auswirkungen nicht eindeutig sind. Schalten Sie deshalb immer die CPU in den STOP-Zustand, wenn Sie mit dem Dialog "Inbetriebnahme" arbeiten, oder deaktivieren Sie Ihr Programm.

6.2.6.6 FM 451 in Betrieb nehmen

6.2.6.7 FM 451: Online & Diagnose

Aktuelle Informationen zur Baugruppe

Standarddiagnose

Fehleranzeige durch die Fehler-LEDs der FM 451

Wo wird der Fehler angezeigt?

Die roten Fehler-LED auf der Baugruppe zeigen verschiedene Fehlerzustände an. Die LED leuchten auch bei kurzfristig auftretenden Fehlern mindestens 3 s lang.

Das folgende Bild zeigt die Lage der Fehler-LED an der FM 451.

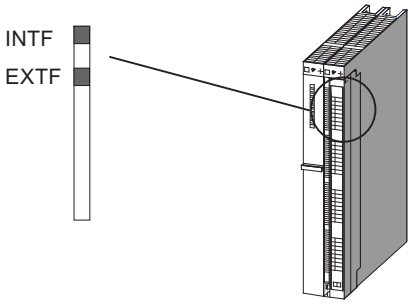

Welche Fehler werden angezeigt?

Die folgende Tabelle zeigt die Fehlerarten, die durch die Fehler-LED angezeigt werden.

LED	Bedeutung	Erläuterungen
INTF	Sammelfehler für interne Fehler	Diese LED zeigt folgende Fehlerzustände der FM 451 an: • einen Baugruppendefekt • einen internen (Kanal-)Fehler • Die FM 451 ist nicht projektiert. Projektieren Sie die Baugruppe. Zur Beseitigung des Fehlers siehe Fehlerklassen (Seite 2654).
EXTF	Sammelfehler für externe Fehler	Diese LED zeigt einen externen (Kanal-)Fehler an.

Kanalspezifische Diagnose

Auslösen von Diagnosealarmen

Alarmbearbeitung

Die Positionierbaugruppe kann Diagnosealarme auslösen. Diese Alarme bearbeiten Sie in einem Alarm-OB. Wenn ein Alarm ausgelöst wird, ohne dass der zugehörige OB geladen ist, geht die CPU in STOP.

Die Bearbeitung der Diagnosealarme geben Sie folgendermaßen frei:

1. Wählen Sie die Baugruppe in HWCN aus.
2. Geben Sie über "Bearbeiten > Objekteigenschaften > Grundparameter" den Diagnosealarm frei.
3. Speichern und übersetzen Sie die Konfiguration.
4. Laden Sie die Konfiguration in die CPU.

Übersicht über die Diagnosealarme

Folgende Ereignisse und Fehler lösen einen Diagnosealarm aus:

- Betriebsfehler
- Fehlerhafte Maschinendaten
- Diagnosefehler

Sie finden diese Fehler unter "Fehlerklassen (Seite 2654)" ausführlich erläutert.

Reaktion der Positionierbaugruppe bei einem Fehler mit Diagnosealarm

- Die Positionierung wird abgebrochen.
- Die Synchronisation wird bei folgenden Diagnosealarmen gelöscht:
  - Frontstecker fehlt, externe Hilfsspannung für die Geberversorgung fehlt
  - ein Nullmarkenfehler wurde erkannt, Leitungsfehler (5 V-Gebersignale)
  - der Verfahrbereich wurde verlassen (wird durch einen Betriebsfehler gemeldet)
  - Istwert setzen nicht ausführbar (wird durch einen Betriebsfehler gemeldet)
- Steuersignale START, DIR_P und DIR_M werden, bis auf eine Ausnahme, nicht mehr bearbeitet.
  Ausnahme:
  Bei einem Betriebsfehler ist noch ein Tippen in Richtung des Arbeitsbereichs möglich.
- Funktionsschalter und Aufträge werden weiter bearbeitet.

Positionierbaugruppe erkennt einen Fehler ("kommend")

Ein Diagnosealarm ist "kommend", wenn mindestens ein Fehler ansteht. Wurden nicht alle Fehler beseitigt, werden die restlichen anstehenden Fehler nochmals "kommend" gemeldet.

Ablauf:

1. Die Positionierbaugruppe erkennt einen oder mehrere Fehler und löst einen Diagnosealarm aus. Eine der LED "INTF" bzw. "EXTF" leuchtet. Der Fehler wird in den Diagnosepuffer eingetragen.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Sie können die Startinfo des OB 82 auswerten.
4. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
5. Weitere Informationen erhalten Sie, wenn Sie die Anweisung ABS_DIAG bzw. ABS_DIAG_451 aufrufen.

Positionierbaugruppe erkennt den Übergang in den fehlerfreien Zustand ("gehend")

Ein Diagnosealarm ist nur dann "gehend", wenn der letzte Fehler auf der Baugruppe behoben wurde.

Ablauf:

1. Die Positionierbaugruppe erkennt, dass alle Fehler behoben wurden und löst einen Diagnosealarm aus. Die LED "INTF" bzw. "EXTF" leuchtet nicht mehr. Der Diagnosepuffer wird nicht verändert.
2. Das Betriebssystem der CPU ruft den Diagnosealarm-OB, z. B. OB 82 auf.
3. Über den Parameter OB82_MOD_ADDR lesen Sie, welche Baugruppe den Alarm ausgelöst hat.
4. Werten Sie das Bit OB82_MDL_DEFECT aus.
   Wenn dieses Bit "0" ist, sind keine Fehler auf der Baugruppe vorhanden. Ihre Auswertung kann hier enden.

Diagnosealarme in Abhängigkeit des CPU-Status

- Im STOP-Zustand der CPU sind die Diagnosealarme von der Positionierbaugruppe gesperrt.
- Werden im STOP-Zustand der CPU nicht alle der anstehenden Fehler behoben, meldet die Baugruppe die noch nicht beseitigten Fehler noch einmal nach dem Übergang in den RUN-Zustand als "kommend".
- Werden alle anstehenden Fehler im STOP-Zustand der CPU behoben, dann wird der fehlerfreie Zustand der Baugruppe nach dem Übergang in den RUN-Zustand **nicht** mit einem Diagnosealarm gemeldet.

Fehlerauswertung im Anwenderprogramm

Fehlerreaktion im Anwenderprogramm

Im Anwenderprogramm können Sie gezielt auf Fehler reagieren. Dafür stehen Ihnen folgende Mittel zur Verfügung:

- Die Rückgabewerte RET_VAL der eingebundenen Anweisungen:
  Dieser Wert wird bei jedem Aufruf der Anweisung neu ermittelt.
  RET_VAL = -1 ist eine Sammelanzeige für einen synchronen Fehler in einem Auftrag oder bei der Kommunikation mit der Baugruppe.
- Zu jedem Auftrag gehört ein Fehlerbit _ERR als Sammelanzeige für einen Fehler im Auftrag oder einem seiner Vorgänger in einer Auftragskette:
  Das Fehlerbit wird für einen Schreibauftrag und seine Folgeaufträge gesetzt, wenn ein Datenfehler von der Baugruppe gemeldet wurde oder ein Kommunikationsfehler aufgetreten ist.
  Bei Leseaufträgen wird das Fehlerbit für den betroffenen Auftrag gesetzt, wenn ein Kommunikationsfehler aufgetreten ist.
  Die Fehlerbits werden nach der Bearbeitung eines Auftrags von der Anweisung ABS_CTRL bzw. ABS_CTRL_451 neu gesetzt. Diese sollten aber für eine Fehlerauswertung vom Anwenderprogramm zurückgenommen werden.
- Das Rückmeldesignal DATA_ERR als Sammelanzeige für einen Fehler, den die Baugruppe bei einem Schreibauftrag erkannt hat. Das Signal wird beim nächsten Schreibauftrag neu ermittelt.

- Das Rückmeldesignal OT_ERR (Bedienfehler) als Sammelanzeige für einen Fehler, den die Baugruppe beim Starten einer Fahrt erkannt hat. Der Fehler muss nach Behebung der Ursache mit OT_ERR_A = 1 quittiert werden.
- Das Rückmeldesignal DIAG wird gesetzt, wenn sich der Inhalt des Diagnosepuffers geändert hat. Dieses Signal kann später als die Signale DATA_ERR und OT_ERR kommen.
- Der Kommunikationsfehler JOB_ERR enthält den Fehlercode (Seite 2653) bei einem Kommunikationsproblem zwischen Anweisung und Baugruppe. Der Wert wird nach der Bearbeitung eines Auftrags neu ermittelt und bei der Anweisung ABS_CTRL bzw. ABS_CTRL_451 im Kanal-DB und bei der Anweisung ABS_DIAG bzw. ABS_DIAG_451 im Diagnose-DB abgelegt.
- Die Anweisung ABS_DIAG bzw. ABS_DIAG_451 zum Auslesen des Diagnosepuffers der Baugruppe. Hier können Sie die Fehlerursachen für synchrone und asynchrone Ereignisse erfahren.
- Diagnosealarme für die schnelle Reaktion auf Ereignisse im Diagnosealarm-OB (z. B. OB 82).

Auswertung eines Diagnosealarms im Anwenderprogramm

Folgende Einträge in den Lokaldaten des Diagnosealarm-OBs (z. B. OB 82) werden von der Positionierbaugruppe gesetzt. Die Fehler werden auch im Diagnosepuffer eingetragen (Fehlerklasse 128 (Seite 2661)).

Adresse	Name	Typ	Kommentar
0.0	OB82_EV_CLASS	BYTE	Ereignisklasse und Kennungen: B#16#38: gehendes Ereignis B#16#39: kommendes Ereignis
1.0	OB82_FLT_ID	BYTE	Fehlercode (B#16#42)
2.0	OB82_PRIORITY	BYTE	Prioriätsklasse: B#16#1A im Betriebszustand RUN B#16#1C im Betriebszustand ANLAUF
3.0	OB82_OB_NUMBR	BYTE	OB-Nummer (82)
4.0	OB82_RESERVED_1	BYTE	Reserviert
5.0	OB82_IO_FLAG	BYTE	Eingabebaugruppe: B#16#54
6.0	OB82_MDL_ADDR	INT	Logische Basisadresse der Baugruppe, in der der Fehler aufgetreten ist
8.0	OB82_MDL_DEFECT	BOOL	Baugruppenstörung
8.1	OB82_INT_FAULT	BOOL	Interner Fehler
8.2	OB82_EXT_FAULT	BOOL	Externer Fehler
8.3	OB82_PNT_INFO	BOOL	Kanalfehler vorhanden
8.4	OB82_EXT_VOLTAGE	BOOL	Externe Hilfsspannung fehlt
...	nicht verwendet		
10.3	OB82_WTCH_DOG_FLT	BOOL	Zeitüberwachung hat angesprochen
...	nicht verwendet		
12.0	OB82_DATE_TIME	DATE_AND_TIME	Datum und Uhrzeit, zu denen der OB angefordert wurde

Diagnosepuffer der Baugruppe

Diagnoseereignisse

Der Diagnosepuffer der Baugruppe enthält maximal 9 Diagnoseereignisse und ist als Ringpuffer organisiert.

Ein Diagnoseereignis wird in den Puffer geschrieben, wenn eine (Fehler-) Meldung "kommend" erkannt wird. Dies kann eine Meldung, ein synchroner Fehler (Datenfehler, Bedienfehler) oder auch ein asynchroner Fehler (Betriebsfehler und Diagnosefehler) sein. Aus einer Fehlerursache können auch mehrere Einträge als Folgefehler entstehen. Gehende Meldungen erzeugen keine Einträge im Diagnosepuffer.

Für jedes Diagnoseereignis werden angegeben:

- Status (immer kommend)
- Interner Fehler
- Externer Fehler
- Fehlerklasse
- Fehlernummer
- Kanalnummer
- Schrittmaßnummer (bei Schrittmaßtabellenfehler)

Wenn ein Diagnoseereignis in den Diagnosepuffer geschrieben wird, wird das Rückmeldesignal DIAG = 1 in allen parametrierten Kanälen gesetzt.

Der Diagnosepuffer kann als Ganzes mit der Anweisung ABS_DIAG bzw. ABS_DIAG_451 in einen Datenbaustein (Diagnose-DB) übertragen werden oder über den Fehlerauswertungsdialog der Projektiersoftware angezeigt werden. Wird der Diagnosepuffer gelesen, setzt die Baugruppe das Rückmeldesignal DIAG = 0.

Hinweis

Wird der Diagnosepuffer gleichzeitig von der Anweisung ABS_DIAG bzw. ABS_DIAG_451 und dem Fehlerauswertungsdialog gelesen, kann es sein, dass ein neu eingetroffenes Diagnoseereignis vom Programm nicht erkannt wird.

Daten und Aufbau des Diagnose-DB

Daten und Aufbau des Diagnose-DB

Hinweis

Sie dürfen Daten, die in dieser Tabelle nicht aufgeführt werden, nicht verändern.

Die folgende Tabelle zeigt den Aufbau des Diagnose-DB.

Adresse	Name	Typ	Anfangswert	Kommentar
0.0	MOD_ADDR	INT	0	Baugruppenadresse
256.0	JOB_ERR	INT	0	Fehlernummer des Kommunikationsfehlers
258.0	JOBBUSY	BOOL	FALSE	1 = Auftrag aktiv
258.1	DIAGRD_EN	BOOL	FALSE	1 = Diagnosepuffer unbedingt lesen
260.0	DIAG_CNT	INT	0	Anzahl gültiger Einträge der Liste
262.0	DIAG[1]	STRUCT		Diagnosedaten neuester Eintrag
272.0	DIAG[2]	STRUCT		Diagnosedaten zweiter Eintrag
282.0	DIAG[3]	STRUCT		Diagnosedaten dritter Eintrag
292.0	DIAG[4]	STRUCT		Diagnosedaten vierter Eintrag
302.0	DIAG[5]	STRUCT		Diagnosedaten fünfter Eintrag
312.0	DIAG[6]	STRUCT		Diagnosedaten sechster Eintrag
322.0	DIAG[7]	STRUCT		Diagnosedaten siebter Eintrag
332.0	DIAG[8]	STRUCT		Diagnosedaten achter Eintrag
342.0	DIAG[9]	STRUCT		Diagnosedaten neunter Eintrag

Die folgende Tabelle zeigt die Struktur eines Diagnoseeintrags DIAG[n].

Adresse	Name	Typ	Anfangswert	Kommentar
+0.0	STATE	BOOL	FALSE	0 = Ereignis gehend 1 = Ereignis kommend
+0.1	INTF	BOOL	FALSE	1 = interner Fehler
+0.2	EXTF	BOOL	FALSE	1 = externer Fehler
+2.0	FCL	INT	0	Fehlerklasse: 1: Betriebsfehler 2: Bedienfehler 4: Datenfehler 5: Maschinendatenfehler 6: Schrittmaßtabellen-Fehler 15: Meldungen 128: Diagnosefehler
+4.0	FNO	INT	0	Fehlernummer
+6.0	CH_NO	INT	0	Kanalnummer
+8.0	TRG_NO	INT	0	Schrittmaßnummer

Liste der JOB_ERR-Meldungen

JOB_ERR-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80A0	32928	-32608	Negative Quittung beim Lesen von der Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2 [1]	32930	-32606	Protokollfehler bei Layer 2
80A3 [1]	32931	-32605	Protokollfehler bei User-Interface / User
80A4 [1]	32932	-32604	Kommunikation am K-Bus gestört
80B0	32944	-32592	Datensatz/Auftrag unbekannt.
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0 [1]	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1 [1]	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2 [1]	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3 [1]	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4 [1]	32964	-32572	Kommunikationsfehler
80C5 [1]	32965	-32571	Dezentrale Peripherie nicht verfügbar.
80C6 [1]	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden. (Schreibauftrag)
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß. (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden. (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers. (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden. (Leseauftrag)
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag)
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß. (Leseauftrag)
873A	34618	-30918	Parameter-DB nicht vorhanden. (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers. (Leseauftrag)

[1] Die Fehler 80A2 … 80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein.

Fehlerklassen

Betriebsfehler

Fehlerklasse 1: Betriebsfehler

Betriebsfehler werden asynchron zu einer Bedienung/Steuerung erkannt. Die Betriebsfehler führen zum Abbruch der Positionierung, außer bei Fehlernummer 9. Fehlernummer 9 führt zum Absteuern der Positionierung.

Die folgende Tabelle zeigt die möglichen Betriebsfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Software-Endschalter Anfang überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
2	**Software-Endschalter Ende überfahren**		ja
	Ursache	Der Istwert liegt außerhalb des Arbeitsbereichs.	
3	**Verfahrbereichsanfang überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
4	**Verfahrbereichsende überfahren**		ja
	Ursache	Verfahrbereichsgrenze überfahren (die Koordinaten der Verfahrbereichsgrenzen gehören mit zum Verfahrbereich).	
5	**Fehler beim Zieleinlauf**		ja
	Ursache	Zielbereich wurde innerhalb der Überwachungszeit nicht erreicht.	
6	**Stillstandsbereich verlassen**		ja
	Ursache	Der Istwert liegt außerhalb des Stillstandsbereichs.	
7	**Mitkopplung**		ja
	Ursache	Istwertänderung > ½ Stillstandsbereich in die falsche Richtung.	
8	**Fehlende oder zu geringe Istwertänderung**		ja
	Ursache	Keine Istwertänderung oder eine Istwertänderung entgegen der Sollrichtung innerhalb der Überwachungszeit.	
9	**Ziel überfahren (FM 451)**		ja
	Ursache	Beim "Fliegend Istwert setzen" wurde das Ziel überfahren.	
10	**Zielbereich überfahren**		ja
	Ursache	Zielbereich wurde nach Zieleinlauf überfahren.	
11	**Umschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Umschaltpunkt.	
12	**Abschaltpunkt fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Abschaltpunkt/Umkehrpunkt.	
13	**Zielbereichsanfang fehlerhaft geschaltet**		ja
	Ursache	Achse pendelt im Zielbereich.	
14	**Änderung größer als der halbe Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwertsprünge.	

Nr.	Bedeutung		Diagnosealarm
15	**Änderung größer als der Rundachsenbereich**		ja
	Ursache	Die Geschwindigkeit/Frequenz ist zu hoch oder es existieren fehlerhafte Istwert-sprünge.	
16	**Schrittmaß für Schrittmaßnummer 252 nicht übertragen (FM 451)**		ja
	Ursache	Das Schrittmaß wurde nicht übertragen.	
17	**Schrittmaß für Schrittmaßnummer 252 nicht anfahrbar (FM 451)**		ja
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Schrittmaß ist kleiner als die Umschalt- bzw. Abschaltdifferenz.	
18	**Falsches Schrittmaß für Schrittmaßnummer 252 (FM 451)**		ja
	Ursache	Das Schrittmaß liegt außerhalb des Arbeitsbereichs.	

Bedienfehler

Fehlerklasse 2: Bedienfehler

Bedienfehler werden bei Änderung der Steuersignale im Nutzdatenbereich erkannt. Die Bedienfehler führen zum Absteuern der Positionierung.

Die folgende Tabelle zeigt die möglichen Bedienfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Unzulässige Betriebsart**		nein
	Ursache	Die angewählte Betriebsart ist unzulässig.	
3	**Unzulässiger Schnittstellenauftrag**		nein
	Ursache	Das angewählte Signal ist mit dieser Betriebsart unzulässig.	
4	**Falscher Betriebsartenparameter**		nein
	Ursache	• Bei der Betriebsart "Tippen" ist die Geschwindigkeitsvorgabe ungleich der Eil- oder Schleichgeschwindigkeit. • Bei der Betriebsart "Schrittmaßfahrt" ist das Schrittmaß ungleich 1 bis 100 oder ungleich 252, 254 und 255.	
5	**Startfreigabe nicht vorhanden**		nein
	Ursache	Startfreigabe beim Starten nicht vorhanden.	
7	**Ziel/Zielbereich liegt außerhalb des Arbeitsbereichs**		nein
	Ursache	Vorgegebenes bzw. berechnetes Ziel liegt außerhalb der Software-Endschalter.	
8	**Achse nicht parametriert**		nein
	Ursache	Falsche oder keine Maschinendaten wurden für die Achse parametriert.	
9	**Achse nicht synchronisiert**		nein
	Ursache	Die Betriebsart "Schrittmaßfahrt" ist nur mit einer bereits synchronisierten Achse möglich.	
10	**Ziel/ Wegstück nicht positionierbar**		nein
	Ursache	Der Abstand zwischen aktueller Istposition und dem vorgegebenen Ziel ist kleiner als die Abschaltdifferenz.	
17	**Referenzpunktfahrt nicht möglich**		nein
	Ursache	Es wurde ein SSI-Geber angeschlossen.	

Nr.	Bedeutung		Diagnosealarm
18	**Schrittmaßfahrt relativ oder absolut nicht möglich**		nein
	Ursache	Das Schrittmaß ist ungültig.	
19	**Abschaltdifferenz nicht größer als ½ Zielbereich bei Schrittmaßnummer 255**		nein
	Ursache	Die Abschaltdifferenz für das Schrittmaß 255 ist kleiner als der halbe Zielbereich.	
20	**Fahrt in die vorgegebene Richtung unzulässig**		nein
	Ursache	Der Abstand zum Software-Endschalter ist zu gering.	

Datenfehler

Fehlerklasse 4: Datenfehler

Datenfehler werden synchron zu einer Bedienung/Steuerung erkannt. Die Datenfehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Datenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück/ Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
10	**Fehlerhafte Nullpunktverschiebung (FM 451)**		nein
	Ursache	Die Nullpunktverschiebung ist größer als ±100 m bzw. ±1000 m. Die Software-Endschalter liegen nach der Nullpunktverschiebung außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m). Rundachse: Der Betrag der Nullpunktverschiebung ist größer als das Ende der Rundachse.	
11	**Fehlerhafte Istwertvorgabe**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
12	**Fehlerhafter Bezugspunkt**		nein
	Ursache	Linearachse: Die Koordinate liegt außerhalb der momentanen (evtl. verschobenen) Software-Endschalter. Rundachse: Die Koordinate ist < 0 oder größer als das Ende der Rundachse.	
20	**Maschinendaten aktivieren nicht zulässig**		nein
	Ursache	Es sind keine neuen, fehlerfreien Maschinendaten auf der Baugruppe vorhanden.	
27	**Unerlaubte bitcodierte Einstellung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
29	**Unzulässige Bitcodierung**		nein
	Ursache	Nicht verwendete und hier nicht beschriebene Bits sind ungleich 0.	
34	**Istwertsetzen rückgängig nicht möglich**		nein
	Ursache	Der Lageistwert würde bei einem SSI-Geber und einer Linearachse nach der Ausführung der Einstellung außerhalb des Arbeitsbereichs liegen.	

Nr.	Bedeutung		Diagnosealarm
36	**Fehlerhafte Umschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	
37	**Fehlerhafte Abschaltdifferenz bei der Schrittmaßnummer 255**		nein
	Ursache	Der Wert liegt außerhalb des zulässigen Zahlenbereichs von ±100 m bzw. ±1000 m. Die Abschaltdifferenz muss kleiner als die Umschaltdifferenz sein.	
107	**Achse nicht parametriert**		nein
	Ursache	Auf der Achse sind entweder keine Maschinendaten vorhanden oder sie sind nicht aktiviert.	
108	**Achse nicht synchronisiert**		nein
	Ursache	Einer der Aufträge "Istwert setzen" und "Istwert setzen rückgängig" wurde angestoßen, obwohl die Achse nicht synchronisiert ist.	

Maschinendatenfehler

Fehlerklasse 5: Maschinendatenfehler

Der Diagnosealarm wird nur bei einem fehlerhaften Systemdatenbaustein (SDB) ausgelöst. Die Maschinendatenfehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Maschinendatenfehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
5	**Fehler in Prozessalarmeinstellung**		ja
	Ursache	Sie haben versucht, einen Prozessalarm anzuwählen, den die Baugruppe nicht unterstützt.	
6	**Falscher minimaler Kantenabstand (FM 451)**		ja
	Ursache	Sie haben als minimalen Kantenabstand einen Wert < 0 oder > 10^9 µm eingegeben.	
7	**Falsches Maßsystem**		ja
	Ursache	Der Wert für das Maßsystem liegt außerhalb des zulässigen Bereichs von 1 bis 4 und 6.	
8	**Falsche Achsart**		ja
	Ursache	Sie haben als Achsart weder 0 noch 1 angegeben.	
9	**Falsches Ende der Rundachse**		ja
	Ursache	Der Wert für das Ende der Rundachse liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm bzw. 1 bis 10^8 µm (je nach Auflösung).	
10	**Falscher Gebertyp**		ja
	Ursache	Der Wert für den Gebertyp liegt außerhalb des zulässigen Bereichs von 1 bis 4.	
11	**Falscher Weg pro Geberumdrehung**		ja
	Ursache	Der Wert für Weg/Geberumdrehung liegt außerhalb des zulässigen Bereichs von 1 bis 10^9 µm (unabhängig von der Auflösung).	
13	**Falsche Inkremente pro Geberumdrehung** (siehe "Inkremente pro Geberumdrehung (Seite 2633)")		ja
14	**Falsche Anzahl Umdrehungen** (siehe "Anzahl Geberumdrehungen (Seite 2635)")		ja

Nr.	Bedeutung		Diagnosealarm
15	**Falsche Baudrate**		ja
	Ursache	Sie haben für die Baudrate einen Wert außerhalb des zulässigen Bereichs von 0 bis 3 angegeben.	
16	**Falsche Referenzpunktkoordinate**		ja
	Ursache	Die Koordinate liegt außerhalb des Bereichs von -100 m bis +100 m bzw. -1000 m bis +1000 m (je nach Auflösung). Linearachse: Die Koordinate liegt außerhalb des Arbeitsbereichs. Rundachse: Die Koordinate ist größer als das Ende der Rundachse oder < 0.	
17	**Falsche Absolutwertgeberjustage**		ja
	Ursache	SSI-Weggeber: Der Wert der Absolutwertgeberjustage liegt nicht im Geberbereich (Inkremente pro Geberumdrehung × Anzahl Umdrehungen - 1).	
18	**Falsche Art der Referenzpunktfahrt**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0, 1, 2 und 3 angegeben.	
19	**Falsche Zählrichtung**		ja
	Ursache	Sie haben einen Wert außerhalb der zulässigen Wertemenge von 0 und 1 angegeben.	
20	**Hardwareüberwachung nicht möglich**		ja
	Ursache	Sie haben die Überwachung Telegrammfehler im Parameter-DB auf "FALSE" gesetzt. Beim verwendeten Geber ist die Überwachung von Fehlimpulsen nicht möglich. Deaktivieren Sie den Parameter MON_PULSE.	
21	**Falscher Software-Endschalter Anfang**		ja
	Ursache	Linearachse: Der Software-Endschalter Anfang liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung). Linearachse: Der Software-Endschalter Anfang ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) kleiner als -100 m bzw. -1000 m (je nach Auflösung).	
22	**Falscher Software-Endschalter Ende**		ja
	Ursache	Linearachse: Der Software-Endschalter Ende liegt außerhalb des Verfahrbereichs (-100 m...+100 m bzw. -1000 m...+1000 m, je nach Auflösung) oder ist kleiner als der Software-Endschalter Anfang. Linearachse: Der Software-Endschalter Ende ist (ggf. inklusive einer vorhandenen Nullpunktverschiebung) größer als +100 m bzw. +1000 m (je nach Auflösung).	
23	**Falsche Maximalgeschwindigkeit**		ja
	Ursache	Die nicht aufgelisteten Daten im Parameter-DB müssen 0 sein.	
24	**Falscher Zielbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	
25	**Falsche Überwachungszeit**		ja
	Ursache	Der Wert für die Überwachungszeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 ms.	
26	**Falscher Stillstandsbereich**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m, je nach Auflösung. Rundachse: Bereich größer als das Ende der Rundachse.	

Nr.	Bedeutung		Diagnosealarm
127	**Falsche Stillstandsgeschwindigkeit**		ja
	Ursache	Der Wert für die Stillstandsgeschwindigkeit liegt außerhalb des zulässigen Bereichs von 0 bis 100 000 µm/min.	
128	**Falsche Ansteuerart**		ja
	Ursache	Sie haben für die Ansteuerart einen Wert außerhalb des zulässigen Bereichs von 1 bis 4 angegeben.	
129	**Falsche Startgeschwindigkeit für die Referenzpunktfahrt**		ja
	Ursache	Sie haben als Startgeschwindigkeit weder 0 noch 1 angegeben.	
130	**Falsche Umschaltdifferenz in Richtung +**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
131	**Falsche Umschaltdifferenz in Richtung -**		ja
	Ursache	Linearachse: Bereich zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung). Rundachse: Bereich größer als das Ende der Rundachse und kleiner ½ Zielbereich.	
132	**Falsche Abschaltdifferenz in Richtung +**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz plus, kleiner ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
133	**Falsche Abschaltdifferenz in Richtung -**		ja
	Ursache	Die Abschaltdifferenz ist größer als Umschaltdifferenz minus, kleiner ½ Zielbereich oder liegt außerhalb des zulässigen Bereichs zwischen 0 bis 100 m bzw. 1000 m (je nach Auflösung).	
200	**Falsche Auflösung**		ja
	Ursache	Sie haben eine Auflösung < 0,1 µm/Impuls oder > 1000 µm/Impuls angegeben. Sie haben einen Weg/Geberumdrehung und eine Anzahl Impulse/Geberumdrehung angegeben, aus denen sich eine Auflösung < 0,1 oder > 1000 ergibt.	
201	**Geber passt nicht zum Arbeitsbereich / Rundachsenbereich**		ja
	Ursache	SSI-Geber und Rundachse: Der Geber deckt nicht genau den Rundachsenbereich ab. Linearachse: Der Geber deckt nicht mindestens den Arbeitsbereich (inkl. Software-Endschalter) ab.	

Schrittmaßtabellen-Fehler

Fehlerklasse 6: Schrittmaßtabellen-Fehler

Die Schrittmaßtabellen-Fehler führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Schrittmaßtabellen-Fehler und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
6	**Vorgegebenes Schrittmaß in der Schrittmaßtabelle zu groß**		nein
	Ursache	Der Wert liegt außerhalb von ±100 m bzw. ±1000 m. Das Wegstück/Ziel darf nicht größer als der Verfahrbereich sein. Bei einer Rundachse muss die Koordinate ≥ 0 und kleiner als das Ende der Rundachse sein.	

Meldungen

Fehlerklasse 15: Meldungen

Die Meldungen führen zu keiner Fehlerreaktion.

Die folgende Tabelle zeigt die möglichen Meldungen und ihre Ursachen.

Nr.	Bedeutung		Diagnosealarm
1	**Beginn Parametrierung**		nein
	Ursache	Die Baugruppe hat eine Parametrierung über einen Systemdatenbaustein erkannt.	
2	**Ende Parametrierung**		nein
	Ursache	Die Baugruppe hat die Parametrierung über einen Systemdatenbaustein fehlerfrei abgearbeitet.	
11	**Abstand zum Umschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
12	**Abstand zum Umkehrpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
14	**Abstand zum Abschaltpunkt zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	
15	**Abstand zum Zielbereichsanfang zu gering**		nein
	Ursache	Die Hardware-Reaktionszeiten können nicht eingehalten werden, da der Abstand zwischen den Schaltpunkten zu gering ist.	

Diagnosefehler

Fehlerklasse 128: Diagnosefehler

Die folgende Tabelle zeigt die möglichen Diagnosefehler mit ihren Ursachen und Wirkungen sowie Abhilfemöglichkeiten.

Nr.	Bedeutung		Diagnosealarm
4	**Externe Hilfsspannung fehlt**		ja
	Ursache	• Externe Hilfsspannung 24 V ist nicht angeschlossen bzw. ausgefallen • Sicherung auf der Baugruppe ist defekt • Unterspannung • Masse-Drahtbruch • Kurzschluss (z. B. am angeschlossenen Geber)	
	Wirkung	• Die Positionierung wird auf allen Kanälen abgebrochen • Abschalten der Ausgänge • Löschen der Synchronisation bei Inkrementalgebern, wenn die Hilfsspannung für Geberversorgung fehlt • Die Positionierbaugruppe ist nicht parametriert • Löschen der Startfreigabe	
	Behebung	Korrekten 24 V-Anschluss sicherstellen (Wenn 24 V-Anschluss korrekt, dann Baugruppe defekt.)	
5	**Frontstecker fehlt (FM 451)**		ja
	Ursache	Fronstecker der Positionierbaugruppe ist nicht gesteckt	
	Wirkung	• Externe Hilfsspannung 24 V fehlt • Baugruppe ist nicht betriebsbereit	
	Behebung	Frontstecker auf die Positionierbaugruppe stecken	
51	**Zeitüberwachung angesprochen (Watchdog)**		ja
	Ursache	• Starke Störeinflüsse auf die Positionierbaugruppe • Fehler in der Positionierbaugruppe	
	Wirkung	• Baugruppe wird rückgesetzt • Abschalten alle Ausgänge • Sofern nach dem Rücksetzen der Baugruppe keine Baugruppendefekte erkannt werden, ist die Baugruppe wieder betriebsbereit • Die Baugruppe meldet den abgelaufenen Watchdog mit "kommend" und "gehend"	
	Behebung	• Beseitigung von Störeinflüssen • Wenden Sie sich an die zuständige Vertriebsabteilung, wobei die genauen Umstände, die zum Fehler führten, von großer Wichtigkeit sind • Tauschen der Positionierbaugruppe	

Nr.	Bedeutung		Diagnosealarm
144	**Drahtbruch Geber**		ja
	Ursache	• Geberkabel nicht gesteckt oder abgeschert • Geber ohne Quersignale • Anschlussbelegung falsch • Kabellänge zu groß • Kurzschluss der Gebersignale • Flankenfehler der Gebersignale • Maximale Eingangsfrequenz des Gebereingangs überschritten • Ausfall der Geberversorgung	
	Wirkung	• die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Synchronisation bei Inkrementalgebern • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geberspezifikation einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden • Technische Daten der Baugruppe einhalten	
145	**Telegrammfehler Absolutwertgeber**		ja
	Ursache	Der Telegrammverkehr zwischen der Positionierbaugruppe und dem Absolutwert-geber (SSI) ist fehlerhaft oder unterbrochen: • Geberkabel nicht gesteckt oder abgeschert • Falsche Geberart • Geber falsch eingestellt (programmierbare Geber) • Telegrammlänge falsch vorgegeben • Geber liefert fehlerhafte Werte (Geber ist defekt) • Störeinstreuung auf Messsystemkabel • Baudrate zu hoch gewählt • Monoflopzeit des Gebers größer als 64 µs	
	Wirkung	• Die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Geberkabel kontrollieren • Geber kontrollieren • Überprüfung des Telegrammverkehrs zwischen Geber und Positionierbaugruppe	

Nr.	Bedeutung		Diagnosealarm
146	**Fehlimpulse Inkrementalgeber**		ja
	Ursache	• Geberüberwachung hat Fehlimpulse festgestellt • Anzahl Inkremente pro Geberumdrehung ist falsch eingegeben • Geber defekt: liefert nicht die angegebene Impulszahl • fehlerhafte oder keine Nullmarke • Einstreuungen auf das Geberkabel	
	Wirkung	• die Positionierung wird abgebrochen • Abschalten der Ausgänge • Löschen der Startfreigabe	
	Behebung	• Anzahl der Inkremente/Geberumdrehung korrekt eingeben (Parametrierdialog) • Geber und Geberkabel kontrollieren • Schirmungs- und Erdungsvorschriften einhalten • Überwachung kann mit dem Parametrierdialog vorübergehend unter Verantwortung des Betreibers ausgeblendet werden	

6.2.7 ET 200S 1PosU einsetzen

6.2.7.1 Anschlussbelegung des 1PosU

Verdrahtungsregeln

Bei Verwendung eines Weggebers mit 5 V-Differenzsignalen müssen die Leitungen an den Klemmen 9 und 13, den Klemmen 12 und 16, sowie bei Inkrementalgeber die Leitungen an den Klemmen 11 und 15 paarweise verdrillt und geschirmt sein.

Bei Verwendung eines Inkrementalgebers mit 24 V-Signalen müssen die Leitungen an den Klemmen 9, 11 und 12 geschirmt sein.

Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung

Die folgende Tabelle zeigt die Anschlussbelegung für das 1PosU.

Ansicht	Anschlussbelegung	Bemerkungen			
		Anschluss der Schalter und des Antriebs: Klemmen 1 … 8		**Anschluss des Weggebers 5 V-Differenzsignale bzw. 24 V-Signale: Klemmen 9 … 16**	
1 POS Universal Digital SF 1 5 UP DN 2 POS 6ES7 138-4DL00-0AB0		1: IN0	Endschalter Minus	9: A / D	Spur A / Daten vom SSI-Geber
		5: IN1	Endschalter Plus	13: /A / /D	
		2: IN2	Reduziernocken; Latchsignal	10: DC24V	Spannungsversorgung für den Weggeber
		6: DC24V	Versorgung für die Schalter	14: M	
		3: OUT0	Fahren Minus bzw. Eilgang	11: B	Spur B
		7: 2L+	Einspeisung der Lastspannung für OUT0, OUT1 und OUT2	15: /B	
1 5 9 13 2 6 10 14 3 7 11 15 4 8 12 16	9 A/D 13 $\overline{A}/\overline{D}$ 10 DC24V 14 M 11 B 15 $\overline{B}$ 12 N/C 16 $\overline{N}/\overline{C}$ Enc	4: OUT1	Fahren Plus bzw. Schleichgang	12: N / C	Spur N / SSI-Clock (Taktleitung)
		8: OUT2	Eil-/Schleichgang bzw. Fahren Plus/ Minus	16: /N / /C	

Anschluss von Relais und Schützen an die Digitalausgänge

Hinweis

Der direkte Anschluss von Induktivitäten (z. B. von Relais und Schützen) ist ohne externe Beschaltung möglich. Wenn SIMATIC-Ausgabestromkreise durch zusätzlich eingebaute Kontakte (z. B. Relaiskontakte) abgeschaltet werden können, müssen Sie bei Induktivitäten zusätzliche Überspannungsschutz-Einrichtungen vorsehen (siehe nachfolgendes Beispiel für Überspannungsschutz).

Beispiel für Überspannungsschutz

Das folgende Bild zeigt einen Ausgabestromkreis, der zusätzliche Überspannungsschutz-Einrichtungen notwendig macht. Gleichstrombetätigte Spulen werden mit Dioden oder Z-Dioden beschaltet.

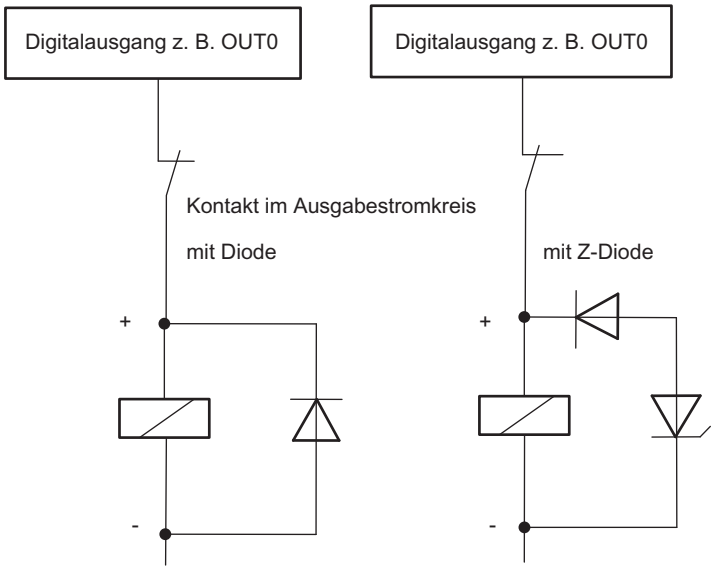

6.2.7.2 Grundlagen des gesteuerten Positionierens über Eil-/Schleichgang

Positioniervorgang

Ab der Startposition wird zunächst mit einer höheren Geschwindigkeit (Eilgang) auf das Ziel zugefahren. In einem vorgegebenen Abstand zum Ziel (Umschaltpunkt) wird auf eine niedrigere Geschwindigkeit (Schleichgang) umgeschaltet. Kurz bevor die Achse das Ziel erreicht, ebenfalls in einem vorgegebenen Abstand zum Ziel, wird der Antrieb abgeschaltet (Abschaltpunkt).

Der Antrieb wird über Digitalausgänge mit Eil- oder Schleichgang und der entsprechenden Richtung angesteuert.

Das folgende Bild zeigt die Schaltpunkte und Schaltdifferenzen. Zur Vereinfachung wird die Änderung der Geschwindigkeit über dem verfahrenen Weg dargestellt.

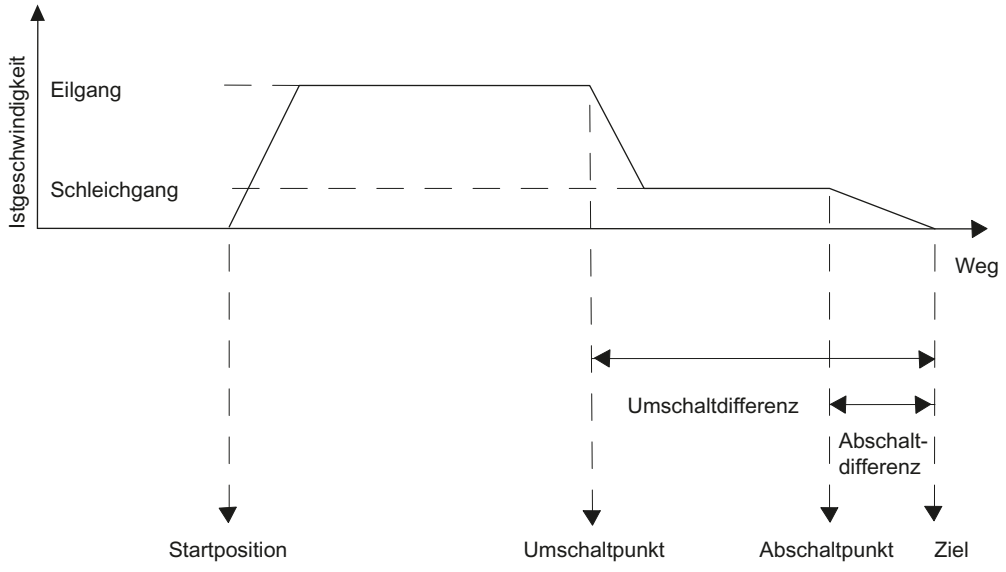

Definitionen

Die folgende Tabelle enthält die Grundbegriffe zum gesteuerten Positionieren über Eil-/ Schleichgang.

Begriff	Erklärung
Arbeitsbereich	definiert den Bereich, den Sie für Ihre Aufgabe durch die Hardware-Endschalter bestimmen. Bei einem SSI-Geber wird der Arbeitsbereich zusätzlich durch den vom SSI-Geber abgedeckten Bereich begrenzt. Den Geberbereich tragen Sie in den Parametern • Impulse pro Geberumdrehung und • Umdrehungen ein. Geberbereich = Impulse pro Geberumdrehung × Umdrehungen Maximaler Arbeitsbereich: • Linearachse max. 0 bis (Geberbereich – 1) • Rundachse von 0 bis (Geberbereich – 1) Bei einem Inkrementalgeber ist der Arbeitsbereich begrenzt auf: • max. 0 bis 16 777 215 Schritte bei einer Linearachse • 0 bis zum parametrierten Ende der Rundachse bei einer Rundachse
Umschaltdifferenz	definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang umgeschaltet wird.
Umschaltpunkt	definiert die Position, an der der Antrieb von Eilgang auf Schleichgang umgeschaltet wird.
Abschaltdifferenz	definiert den Abstand zum Ziel, an dem der Antrieb abgeschaltet wird. Ist die Abschaltdifferenz ≥ Umschaltdifferenz, existiert kein Umschaltpunkt. Es wird nicht von Eil- auf Schleichgang umgeschaltet.
Abschaltpunkt	definiert die Position, an der der Antrieb abgeschaltet wird. Das 1PosU meldet an diesem Punkt das Ende der Fahrt.

Begriff	Erklärung
Startposition	definiert die Position des Antriebs innerhalb des Arbeitsbereichs, von der aus die Fahrt gestartet wird. Liegt die Startposition innerhalb der Abschaltdifferenz, wird der Antrieb nicht angesteuert. Das 1PosU meldet an diesem Punkt das Ende der Fahrt. Liegt die Startposition innerhalb der Umschaltdifferenz, wird die Fahrt nur im Schleichgang durchgeführt.
Ziel	definiert die absolute bzw. relative Position der Achse, die bei der Positionierung angefahren wird. Das Ziel ist die Position einer Achse, die bei einer Fahrt erreicht werden soll. Bei einer absoluten Fahrt geben Sie das Ziel durch Ihr Steuerungsprogramm direkt vor. Bei einer relativen Fahrt wird das Ziel aus der Startposition und dem im Steuerungsprogramm vorgegebenen Wegstück errechnet. Wenn Sie ermitteln wollen, wie genau Sie das Ziel erreicht haben, müssen Sie den Istwert mit der Positionsangabe vergleichen.
Linearachse	definiert den Achstyp mit begrenztem Arbeitsbereich. Er wird begrenzt durch: • den Geberbereich • den darstellbaren Zahlenbereich (0 bis 16 777 215 Schritte) • die Hardware-Endschalter
Rundachse	definiert den Achstyp mit endlosem Arbeitsbereich. Dabei wird die Achsposition nach einer Umdrehung (parametriertes Ende der Rundachse bei Inkrementalgeber bzw. parametrierter Geberbereich bei SSI-Geber) wieder auf 0 gesetzt.
Richtung Minus	bewegt sich der Antrieb in Richtung Minus, wird der angezeigte Istwert kleiner.
Richtung Plus	bewegt sich der Antrieb in Richtung Plus, wird der angezeigte Istwert größer.

6.2.7.3 Funktionen des 1PosU

Funktionsübersicht

Übersicht

Das 1PosU bietet Ihnen folgende Funktionen zur Bewegung Ihrer Achse an:

- Stoppen (Seite 2673)
- Tippen (Seite 2673)
- Referenzpunktfahrt (Seite 2675)
- Positionieren relativ (Seite 2681)
- Positionieren absolut (Seite 2683)

Zusätzlich zu den Bewegungsarten kennt das 1PosU die Funktionen:

- JOB-Bearbeitung abbrechen (Seite 2687)
- Istwert setzen (Seite 2688)
- Geberbereich verschieben (Seite 2690)

- Abschaltdifferenz ändern (Seite 2691)
- Umschaltdifferenz ändern (Seite 2692)
- Referenziersignal auswerten (Seite 2693)
- Latch-Funktion (Seite 2695)
- Drehrichtungsüberwachung einstellen (Seite 2697)
- Aktuelle Werte anzeigen (Seite 2698)
- Fehlererkennung / Diagnose (Seite 2700)
- Verhalten bei CPU/Master-STOP (Seite 2707)

Parameter

In den Parametern legen Sie die von Antrieb, Achse und Geber abhängigen Größen einmalig fest.

Dosierbetrieb

Mit Inkrementalgebern können Sie das 1PosU zum Dosieren verwenden. Den Dosierbetrieb stellen Sie in den Parametern ein. Im Dosierbetrieb wertet das 1PosU nur das Gebersignal A (/A) aus. Bei jeder positiven Flanke wird der Istwert inkrementiert.

Im Dosierbetrieb stehen Ihnen nur die Funktionen "Tippen" und "Positionieren relativ" zur Ansteuerung der Digitalausgänge zur Verfügung.

Das Dosieren selbst stoßen Sie über die Funktion "Positionieren relativ" an. Die Dosiermenge geben Sie beim Starten über die Steuerschnittstelle (Wegstück) vor.

Bei jedem Starten wird der Istwert auf 0 gesetzt und die Digitalausgänge in Abhängigkeit der Um- und Abschaltdifferenz angesteuert.

Arbeitsweise des 1PosU

Das folgende Bild zeigt die Arbeitsweise des 1PosU.

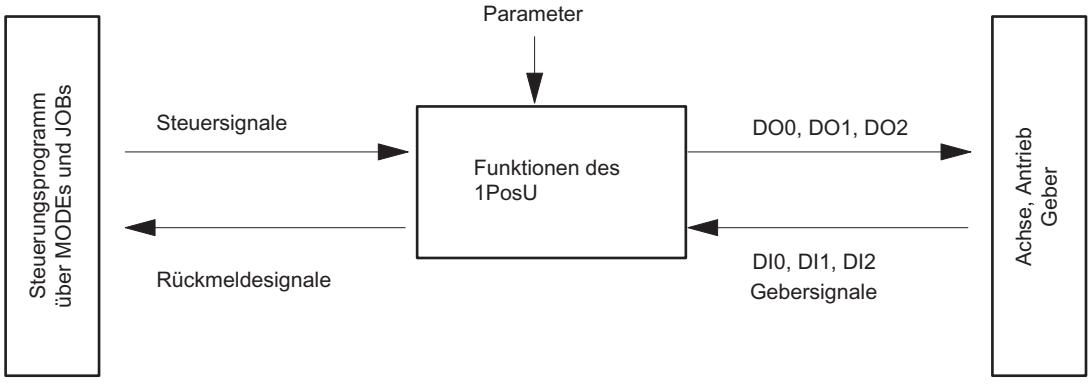

Schnittstellen zum Steuerungsprogramm und zur Achse

Um die Funktion auszuführen, hat das 1PosU als Schnittstelle zur Achse Digitaleingänge, Gebersignale für den Anschluss eines Gebers und Digitalausgänge zur Ansteuerung des Antriebs.

Starten von Bewegungsarten (MODEs)

Die Bewegungsarten (MODEs) steuern und beobachten Sie mit Ihrem Steuerungsprogramm über Steuer- und Rückmeldesignale.

Starten von MODEs

Die folgende Tabelle zeigt, wie Sie MODEs starten und welche Reaktionen dabei vom 1PosU kommen.

Ihre Aktivität	Reaktion des 1PosU
Versorgen Sie die Steuerschnittstelle (Seite 2726) je nach MODE mit den erforderlichen Daten. Prüfen Sie, ob das Rückmeldebit (Seite 2728) POS_ACK = 0 ist.	
Setzen Sie das Steuerbit START von 0 auf 1.	Das 1PosU setzt das Rückmeldebit POS_ACK = 1 und POS_DONE = 0. Sie erkennen daran, dass der Start vom 1PosU erkannt wurde und bei POS_ERR = 0 der MODE ausgeführt wird. Bei POS_ERR = 1 wird der MODE nicht ausgeführt.
Setzen Sie das Steuerbit START von 1 auf 0.	Das 1PosU setzt das Rückmeldebit POS_ACK = 0.
	Bei Stoppen, Referenzpunktfahrt, Positionieren absolut und Positionieren relativ setzt das 1PosU das Rückmeldebit POS_DONE = 1, wenn der MODE ohne Fehler beendet ist. Bei POS_ERR = 1 ist der MODE mit Fehler beendet.
Erst wenn POS_ACK = 0 ist, können Sie wieder einen neuen MODE starten. Wenn Sie während eines laufenden MODEs starten, dann übernimmt das 1PosU die neue Bewegung und führt bei Bedarf einen Richtungswechsel durch.	

Das folgende Bild zeigt das Zusammenspiel der Steuer- und Rückmeldesignale beim Starten von MODEs.

Aktivieren weiterer Funktionen (JOBs)

Die weiteren Funktionen (JOBs) steuern und beobachten Sie mit Ihrem Steuerungsprogramm über Steuer- und Rückmeldesignale.

Aktivieren von JOBs

Die folgende Tabelle zeigt, wie Sie JOBs starten und welche Reaktionen dabei vom 1PosU kommen.

Ihre Aktivität	Reaktion des 1PosU
Versorgen Sie die Steuerschnittstelle (Seite 2726) je nach JOB mit den erforderlichen Daten. Prüfen Sie, ob das Rückmeldebit (Seite 2728) JOB_ACK = 0 ist.	
Setzen Sie das Steuerbit JOB_REQ von 0 auf 1.	Das 1PosU setzt das Rückmeldebit JOB_ACK = 1. Sie erkennen daran, dass der Anstoß vom 1PosU erkannt wurde und bei JOB_ERR = 0 der JOB ausgeführt wird. • Bei "Referenziersignal auswerten" setzt das 1PosU gleichzeitig das Rückmeldebit SYNC = 0. • Bei der Latch-Funktion setzt das 1PosU gleichzeitig das Rückmeldebit LATCH_DONE = 0. • Alle anderen JOBs sind damit ausgeführt. Bei JOB_ERR = 1 wird der JOB nicht ausgeführt.
Setzen Sie das Steuerbit JOB_REQ von 1 auf 0.	Das 1PosU setzt das Rückmeldebit JOB_ACK = 0.
	Bei "Referenziersignal auswerten" setzt das 1PosU das Rückmeldebit SYNC = 1, wenn die Funktion ausgeführt ist. Bei Latch setzt das 1PosU das Rückmeldebit LATCH_DONE = 1, wenn die Funktion ausgeführt ist.
Erst wenn JOB_ACK = 0 ist, können Sie wieder einen neuen JOB aktivieren.	

Das folgende Bild zeigt das Zusammenspiel der Steuer- und Rückmeldesignale beim Aktivieren von JOBs.

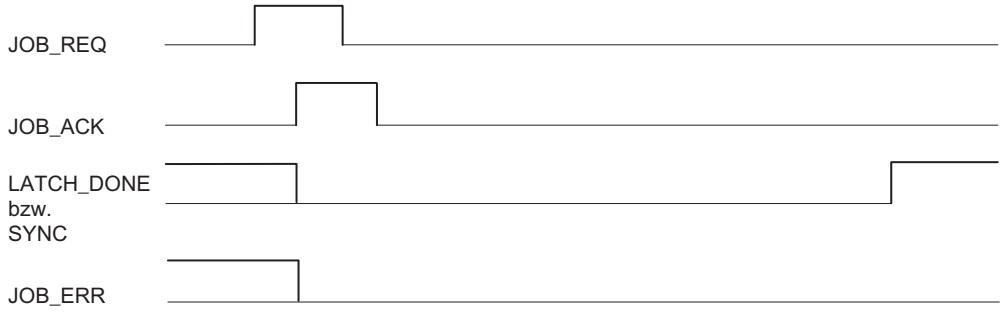

Achse, Antrieb und Geber

Auswertung der Gebersignale

Das 1PosU wertet die vom Geber gelieferten Signale entsprechend dem Gebertyp unterschiedlich aus:

- SSI-Geber:
  Das 1PosU wertet den vom SSI-Geber gelieferten Geberwert in Schritten direkt aus und bildet daraus den Istwert in Schritten (Istwert = Geberwert).
  Der Istwert liegt im Geberbereich von 0 … (Anzahl Umdrehungen × Anzahl Schritte) – 1.
  An den Geberbereichsgrenzen erzeugt das 1PosU einen Über- bzw. Unterlauf des Istwerts.
- Inkrementalgeber:
  Das 1PosU wertet die Impulse des Gebers 4fach aus und summiert diese richtungsspezifisch zum Istwert. Sie müssen bei allen Wegvorgaben in den Parametern und der Steuer- bzw. Rückmeldeschnittstelle die Vierfachauswertung berücksichtigen:
  1 Impuls des Inkrementalgebers entspricht 4 Schritten des 1PosU.
  Der Istwert liegt im Arbeitsbereich von 0 … 16 777 215 Schritten. An den Arbeitsbereichsgrenzen erzeugt das 1PosU einen Über- bzw. Unterlauf des Istwerts im Arbeitsbereich.
- Inkrementalgeber bei Dosierbetrieb:
  Das 1PosU wertet nur die positiven Flanken des Gebersignals A aus und summiert diese zum Istwert.
  Der Istwert liegt im Arbeitsbereich von 0 … 16 777 215 Schritten. An der oberen Arbeitsbereichsgrenze erzeugt das 1PosU einen Überlauf des Istwerts.

Ansteuerung des Antriebs

Über die 3 Digitalausgänge des 1PosU wird der Antrieb angesteuert.

Die Geschwindigkeit wählen Sie über das Steuerbit SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang). Die Geschwindigkeit können Sie auch während der Fahrt ändern.

Den Richtungswechsel (Seite 2714) beeinflussen Sie mit dem Parameter "T_{min} Richtungswechsel".

Sie können den Zustand des jeweiligen Ausgangs aus der Rückmeldeschnittstelle (Seite 2728) (DO 0, DO 1 und DO 2) auslesen.

Die Funktion der Digitalausgänge ist abhängig von der Ansteuerart:

Das folgende Bild zeigt die Zustände der Digitalausgänge bei **Ansteuerart 0**.

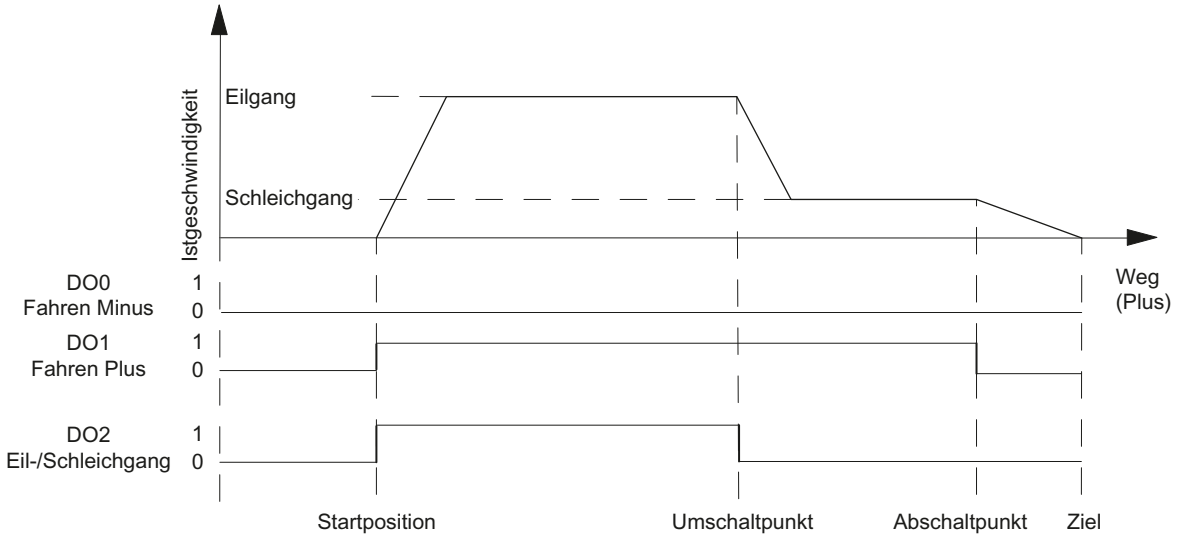

Das folgende Bild zeigt die Zustände der Digitalausgänge bei **Ansteuerart 1**.

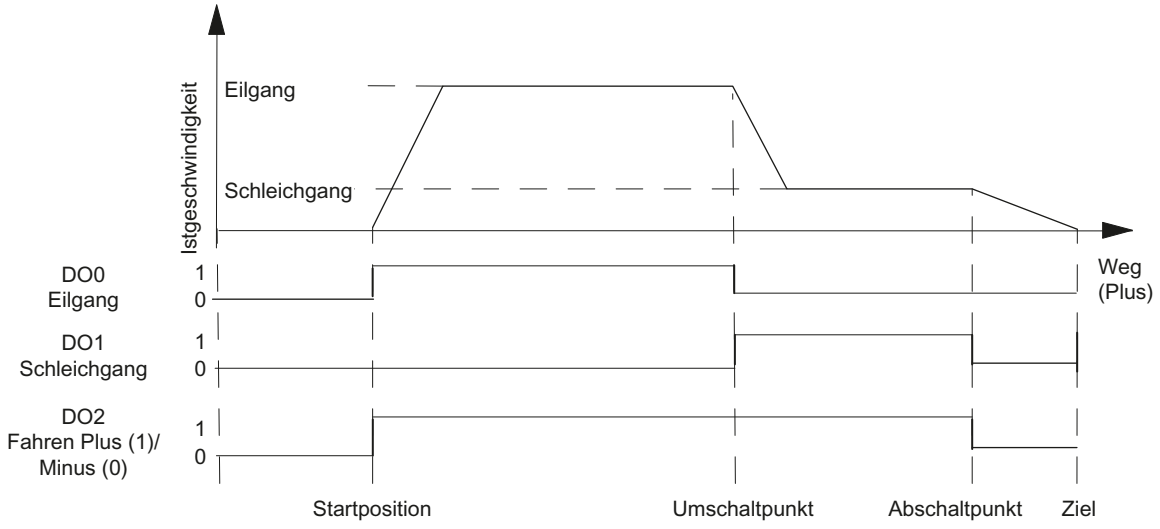

Wirkung der richtungsabhängigen Freigaben

Beschreibung

Mit den Steuerbits DIR_M und DIR_P geben Sie die Digitalausgänge richtungsabhängig frei.

- Mit DIR_M = 1 fahren Sie in Richtung Minus.
- Mit DIR_P = 1 fahren Sie in Richtung Plus.

Unterbrechen und Fortsetzen der Fahrt

Wenn Sie während einer Fahrt die relevante richtungsabhängige Freigabe zurücksetzen, wird die Bewegung der Achse gestoppt, alle 3 Digitalausgänge werden "0" und die Fahrt ist unterbrochen.

Wenn Sie die relevante richtungsabhängige Freigabe wieder setzen, wird die Fahrt fortgesetzt.

Stoppen (MODE 0)

Stoppen

Wenn Sie MODE 0 starten, dann stoppt das 1PosU die laufende Fahrt, alle 3 Digitalausgänge werden "0" und die Fahrt ist beendet (POS_ERR = 0, POS_DONE = 1).

Eine mit MODE 0 beendete Fahrt können Sie nicht mehr fortsetzen. Zum erneuten Bewegen der Achse starten Sie einen neuen MODE.

Steuersignale

Die folgende Tabelle zeigt die für "Stoppen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	0	0	MODE 0 = Stoppen
	Bit 0: START					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Stoppen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK

Tippen (MODE 1)

Tippen

Mit "Tippen" steuern Sie den Antrieb direkt über die Steuerbits (Seite 2672) DIR_M oder DIR_P in eine Richtung.

Wenn Sie MODE 1 starten, dann bewegt das 1PosU den Antrieb mit der vorgegebenen Geschwindigkeit (Steuerbit (Seite 2726) SPEED) in die vorgegebene Richtung (Steuerbits (Seite 2726) DIR_M oder DIR_P).

Sie stoppen den Antrieb durch Setzen der Steuerbits DIR_M = 0 und DIR_P = 0.

Ein Richtungswechsel (Seite 2714) wird nach Ablauf der Zeit T_{min} ausgeführt.

Tippen können Sie auch bei einer nicht synchronisierten Achse (Rückmeldebit SYNC = 0) oder bei anstehendem Geberfehler (Rückmeldebit ERR_ENCODER = 1) bzw. ohne angeschlossenen Geber.

Das folgende Bild zeigt den Ablauf der Bewegungsart "Tippen".

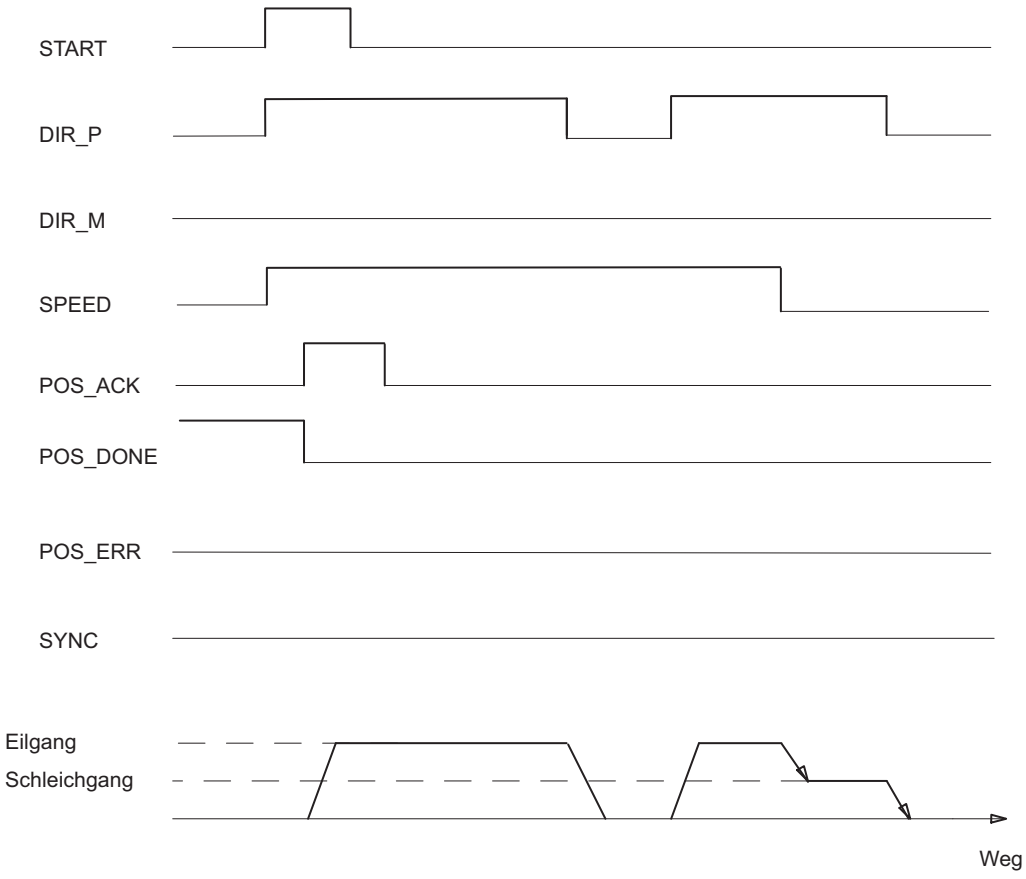

Steuersignale

Die folgende Tabelle zeigt die für "Tippen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	0	1	MODE 1 = Tippen
	Bit 3: SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Tippen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Byte 1 bis 3	Istwert: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1

Fehlerursachen für POS_ERR

Die Fehlerursachen müssen Sie mit dem JOB 15 ermitteln (Aktuelle Werte anzeigen (Seite 2698)).

Die folgende Tabelle zeigt die bei "Tippen" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Fehlerursache	Abhilfe
2	ERR_2L+ wird angezeigt	Überprüfen Sie die Lastspannung (2L+) an der Klemme 7.
5	Endschalter, in dessen Richtung der Antrieb angesteuert wird, ist aktiv	Überprüfen Sie Ihre Schalter und die Verdrahtung sowie die Parameter "DI0 Endschalter Minus (Seite 2711)" und "DI1 Endschalter Plus (Seite 2711)".
7	Tippen: DIR_P = 1 und DIR_M = 1	
13	Drehrichtung von Antrieb und Geber unterschiedlich	Überprüfen Sie die Verdrahtung von Antrieb und Geber sowie den Parameter "Drehrichtungsumkehr (Seite 2710)".
15	Im Dosierbetrieb: DIR_M = 1	

Referenzpunktfahrt (MODE 3)

Referenzpunktfahrt (MODE 3)

Referenzpunktfahrt

Eine Referenzpunktfahrt können Sie nur bei Inkrementalgeber und nicht aktiviertem Dosierbetrieb starten.

Bei der Referenzpunktfahrt synchronisieren Sie die Achse anhand eines externen Referenziersignals. Als Referenziersignal stehen Ihnen wahlweise die 3 Digitaleingänge und die Nullmarke (Seite 2678) zur Verfügung.

Die Digitaleingänge DI0 (Endschalter Minus), DI1 (Endschalter Plus) und DI2 (Reduziernocken) können Sie als Öffner oder Schließer parametrieren.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der Referenzpunktkoordinate und starten Sie MODE 3. Das 1PosU setzt das Rückmeldesignal SYNC = 0 und bewegt den Antrieb mit der vorgegebenen Geschwindigkeit (Steuerbit SPEED) in die parametrierte Startrichtung und sucht das Referenziersignal. Dabei führt das 1PosU notwendige Richtungswechsel an den Endschaltern bzw. dem Reduziernocken automatisch durch.

Setzen Sie die notwendigen richtungsabhängigen Freigaben (Seite 2672) (DIR_M, DIR_P), damit der Antrieb angesteuert wird.

Wenn das 1PosU das parametrierte Referenziersignal erkennt, dann steuert es den Antrieb mit Schleichgang in die Referenzierrichtung. Diese ergibt sich aus den Parametern "Referenziersignal" und "Referenzschalter":

<table>
<tr><th></th><th>Referenzschalter: Reduziernocken nach Minus</th><th>Referenzschalter: Reduziernocken nach Plus</th><th>Referenzschalter: Endschalter Minus</th><th>Referenzschalter: Endschalter Plus</th></tr>
<tr><td>Referenziersignal: Referenzschalter und Nullmarke</td><td rowspan="2">Referenzierrich-tung Minus</td><td rowspan="2">Referenzierrich-tung Plus</td><td rowspan="2">Referenzierrich-tung Plus</td><td rowspan="2">Referenzierrich-tung Minus</td></tr>
<tr><td>Referenziersignal: Referenzschalter</td></tr>
<tr><td>Referenziersignal: Nullmarke</td><td colspan="4">Die Referenzierrichtung ist nicht festgelegt. Mit der nächsten Nullmarke wird die Achse synchronisiert.</td></tr>
</table>

Nach Überfahren des Referenziersignals ist die Achse synchronisiert. Das 1PosU setzt das Rückmeldesignal SYNC = 1 und ordnet dem Istwert die Referenzpunktkoordinate zu.

Das folgende Bild zeigt den Ablauf der Bewegungsart "Referenzpunktfahrt".

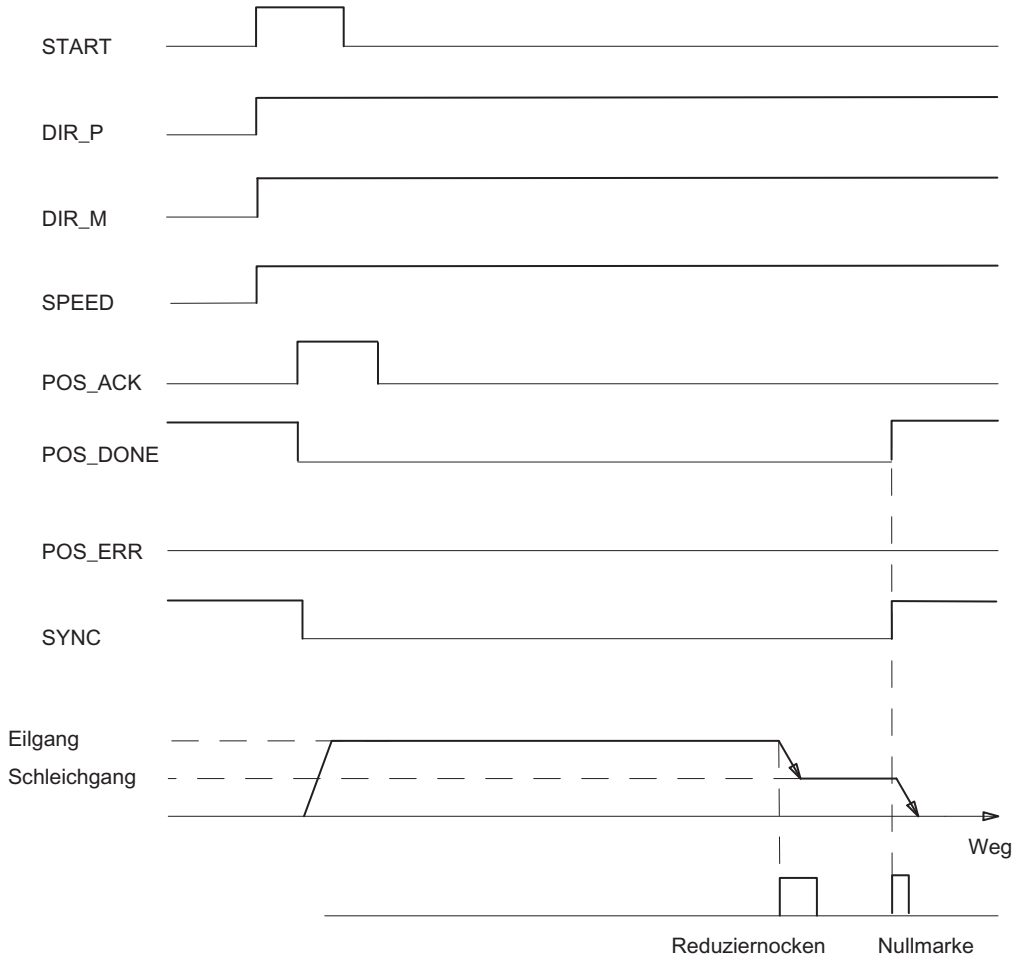

Steuersignale

Die folgende Tabelle zeigt die für "Referenzpunktfahrt" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	1	1	MODE 3 = Referenzpunktfahrt
	Bit 3: SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Byte 1 bis 3	Referenzpunktkoordinate: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … Ende der Rundachse – 1					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Referenzpunktfahrt" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Byte 1 bis 3	Istwert: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … Ende der Rundachse – 1

Fehlerursachen für POS_ERR

Die Fehlerursachen müssen Sie mit dem JOB 15 ermitteln (Aktuelle Werte anzeigen (Seite 2698)).

Die folgende Tabelle zeigt die bei "Referenzpunktfahrt" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Fehlerursache	Abhilfe
1	Unzulässiger MODE im Dosierbetrieb	
2	ERR_2L+ wird angezeigt	Überprüfen Sie die Lastspannung (2L+) an der Klemme 7.
3	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
10	Referenzpunktfahrt: Referenzpunktkoordinate ≥ Ende der Rundachse	
11	kein Referenziersignal bis zum Endschalter bzw. zwischen den Endschaltern gefunden	Überprüfen Sie Ihre Schalter, den Geber und die Verdrahtung.
13	Drehrichtung von Antrieb und Geber unterschiedlich	Überprüfen Sie die Verdrahtung von Antrieb und Geber sowie den Parameter "Drehrichtungsumkehr (Seite 2710)".

Verlauf einer Referenzpunktfahrt

Verlauf einer Referenzpunktfahrt abhängig von Parametrierung und Startposition

Der Verlauf einer Referenzpunktfahrt ist abhängig von der Parametrierung und der Startposition.

Bei einer Referenzpunktfahrt gibt es daher verschiedene Fälle, die abhängig sind

- von der Startposition des Antriebs beim Start der Referenzpunktfahrt,
- von der parametrierten Startrichtung,

- vom parametrierten Referenziersignal und
- vom parametrierten Referenzschalter.

Beispiel 1: Referenzpunktfahrt mit Reduziernocken und Nullmarke

Startposition und Parametrierung:

- Startpositon: zwischen Endschalter Minus und Reduziernocken
- Startrichtung: Plus
- Referenziersignal: Referenzschalter und Nullmarke
- Referenzschalter: Reduziernocken nach Plus

Das folgende Bild zeigt eine Referenzpunktfahrt mit Reduziernocken und Nullmarke.

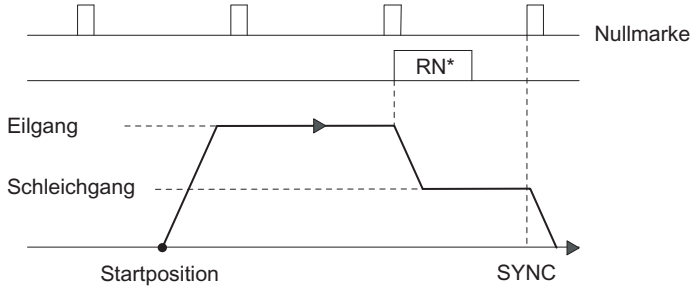

*RN = Reduziernocken

Sie können auch an dem Reduziernocken ohne Nullmarke synchronisieren.

Wenn die Startposition auf dem Reduziernocken liegt, dann steuert das 1PosU den Antrieb mit Schleichgang direkt in die Referenzierrichtung.

Beispiel 2: Referenzpunktfahrt mit Endschalter Minus

Startposition und Parametrierung:

- Startpositon: zwischen Endschalter Minus und Endschalter Plus
- Startrichtung: Minus
- Referenziersignal: Referenzschalter
- Referenzschalter: Endschalter Minus

Das folgende Bild zeigt eine Referenzpunktfahrt mit Endschalter Minus.

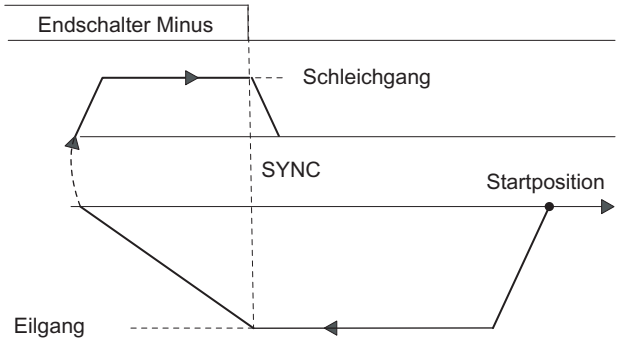

Sie können auch am Endschalter mit der nachfolgenden Nullmarke synchronisieren.

Wenn die Startposition auf dem Endschalter liegt, dann steuert das 1PosU den Antrieb mit Schleichgang direkt in die Referenzierrichtung.

Beispiel 3: Referenzpunktfahrt mit Richtungsumkehr am Endschalter Plus

Startposition und Parametrierung:

- Startpositon: zwischen Endschalter Minus und Reduziernocken
- Startrichtung: Plus
- Referenziersignal: Referenzschalter und Nullmarke
- Referenzschalter: Reduziernocken nach Plus

Das folgende Bild zeigt eine Referenzpunktfahrt mit Richtungsumkehr am Endschalter Plus.

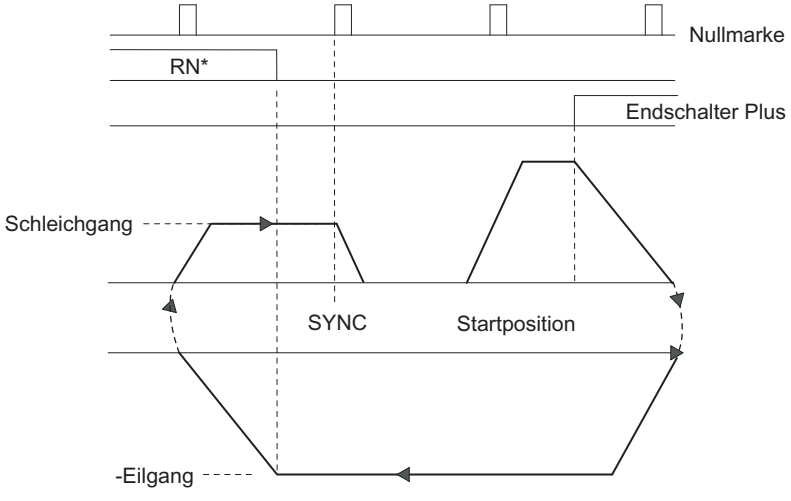

*RN = Reduziernocken

Wenn die Startposition auf dem Endschalter Plus liegt, dann steuert das 1PosU den Antrieb mit Eilgang direkt entgegen der parametrierten Startrichtung.

Beispiel 4: Referenzpunktfahrt nur mit Nullmarke

Startposition und Parametrierung:

- Startpositon: zwischen Endschalter Minus und Endschalter Plus
- Startrichtung: Minus
- Referenziersignal: Nullmarke
- Referenzschalter: irrelevant

Das folgende Bild zeigt eine Referenzpunktfahrt nur mit Nullmarke.

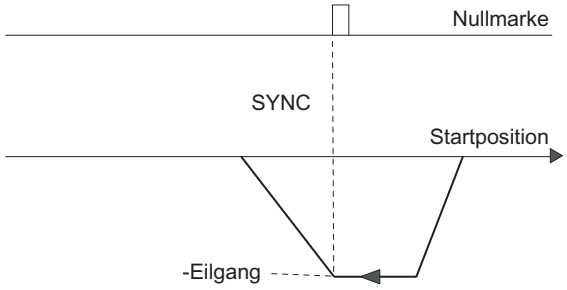

Positionieren relativ (MODE 4)

Positionieren relativ

Mit "Positionieren relativ" steuert das 1PosU den Antrieb von der Startposition um ein vorgegebenes Wegstück in eine vorgegebene Richtung.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit dem Wegstück und starten Sie MODE 4 mit der Richtungsvorgabe (DIR_M oder DIR_P). Das 1PosU bewegt den Antrieb mit der vorgegebenen Geschwindigkeit (Steuerbit SPEED) um das Wegstück. Am Umschaltpunkt schaltet das 1PosU von Eil- auf Schleichgang und am Abschaltpunkt beendet das 1PosU die Fahrt.

Wenn Sie während einer laufenden Fahrt starten, dann wird vom 1PosU ein notwendiger Richtungswechsel (Seite 2714) nach Ablauf der Zeit T_{min} ausgeführt.

Das vorgegebene Wegstück wird vom 1PosU nicht überprüft. Dadurch können Sie bei Rundachsen auch um mehr als eine Umdrehung fahren.

Im Dosierbetrieb (Seite 2667) ist "Positionieren relativ" nur in Richtung Plus möglich. Dabei wird bei jedem Starten der Istwert auf 0 gesetzt.

Das folgende Bild zeigt den Ablauf der Bewegungsart "Positionieren relativ".

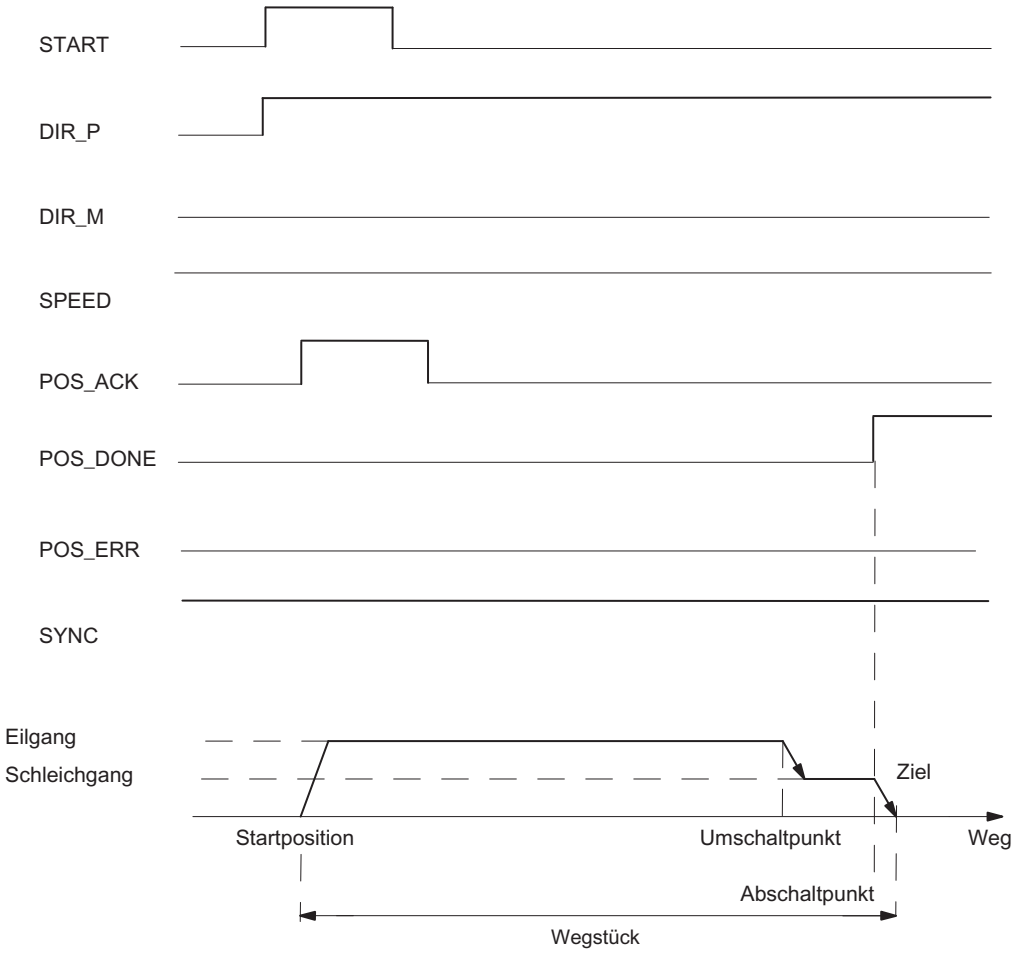

Steuersignale

Die folgende Tabelle zeigt die für "Positionieren relativ" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	1	0	0	MODE 4 = Positionieren relativ
	Bit 3: SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Byte 1 bis 3	Wegstück: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … 16 777 215					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Positionieren relativ" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Byte 1 bis 3	Istwert: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1

Fehlerursachen für POS_ERR

Die Fehlerursachen müssen Sie mit dem JOB 15 ermitteln (Aktuelle Werte anzeigen (Seite 2698)).

Die folgende Tabelle zeigt die bei "Positionieren relativ" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Fehlerursache	Abhilfe
2	ERR_2L+ wird angezeigt	Überprüfen Sie die Lastspannung (2L+) an der Klemme 7.
3	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
5	Endschalter, in dessen Richtung der Antrieb angesteuert wird, ist aktiv	Überprüfen Sie Ihre Schalter und die Verdrahtung sowie die Parameter "DI0 Endschalter Minus (Seite 2711)" und "DI1 Endschalter Plus (Seite 2711)".
7	Positionieren relativ: Starten mit DIR_P = 0 und DIR_M = 0 oder DIR_P = 1 und DIR_M = 1	
13	Drehrichtung von Antrieb und Geber unterschiedlich	Überprüfen Sie die Verdrahtung von Antrieb und Geber sowie den Parameter "Drehrichtungsumkehr (Seite 2710)".
15	Im Dosierbetrieb: DIR_M = 1	

Positionieren absolut (MODE 5)

Positionieren absolut

Mit "Positionieren absolut" bewegt das 1PosU den Antrieb auf absolute Ziele zu. Dazu muss die Achse synchronisiert sein.

"Positionieren absolut" ist nicht möglich bei aktiviertem Dosierbetrieb.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit dem Ziel und starten Sie MODE 5 mit der notwendigen richtungsabhängigen Freigabe (Seite 2672) (DIR_M, DIR_P). Das 1PosU bewegt den Antrieb mit der vorgegebenen Geschwindigkeit (Steuerbit SPEED) zum Ziel. Am Umschaltpunkt schaltet das 1PosU von Eil- auf Schleichgang und am Abschaltpunkt beendet das 1PosU die Fahrt.

Wenn Sie während einer laufenden Fahrt starten, dann wird vom 1PosU ein notwendiger Richtungswechsel (Seite 2714) nach Ablauf der Zeit T_{min} ausgeführt.

Linearachse

Das 1PosU ermittelt die Richtung, in der das Ziel angefahren wird. Sie müssen zum Start die notwendige richtungsabhängige Freigabe (DIR_M, DIR_P) setzen. Sie können auch beide Freigaben setzen.

Rundachse

Die Richtung, in der das Ziel angefahren wird, bestimmen Sie durch die Wahl der richtungsabhängigen Freigabe (DIR_M, DIR_P):

Steuerbits DIR_P und DIR_M	Richtung
DIR_P = 1 DIR_M = 0	Das Ziel wird in Richtung Plus angefahren.
DIR_P = 0 DIR_M = 1	Das Ziel wird in Richtung Minus angefahren.
DIR_P = 1 DIR_M = 1	Das Ziel wird auf dem kürzesten Weg angefahren. Das 1PosU ermittelt die Richtung, in der das Ziel angefahren wird. Ergibt sich eine zu fahrende Strecke kleiner als die Abschaltdifferenz, so wird keine Fahrt gestartet (POS_DONE = 1).

Das folgende Bild zeigt den Ablauf der Bewegungsart "Positionieren absolut".

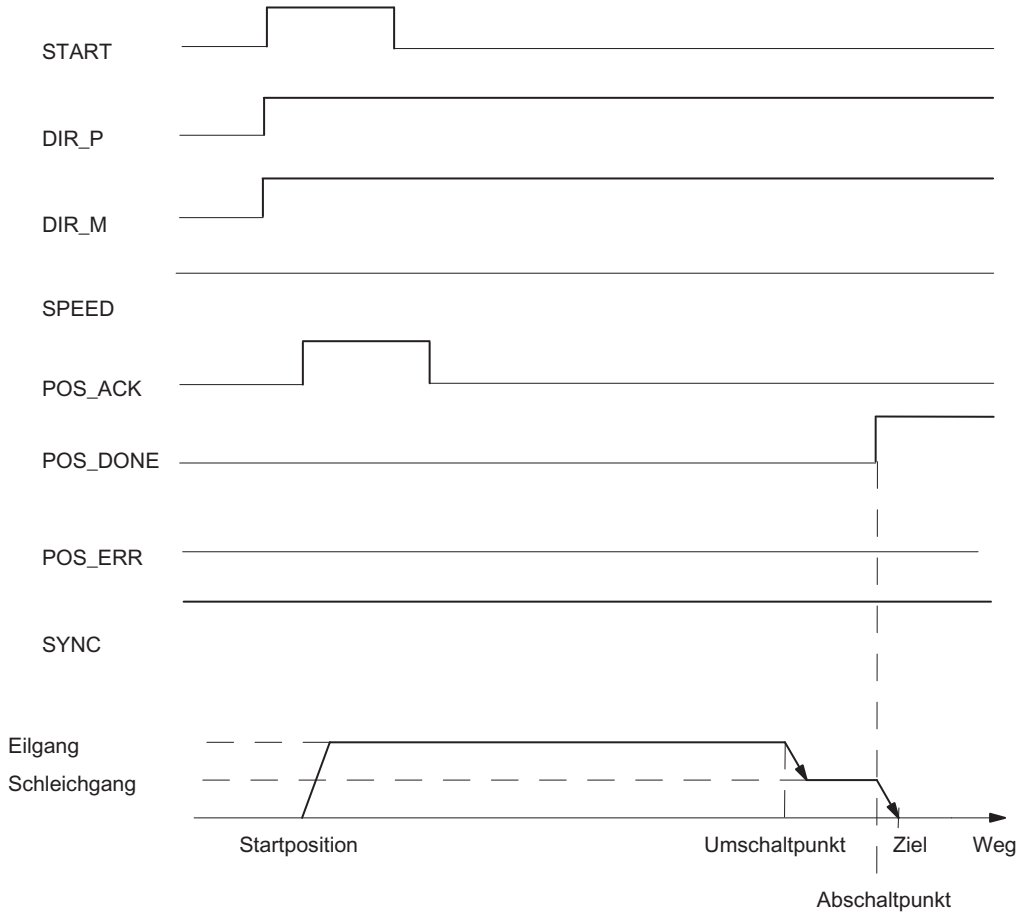

Steuersignale

Die folgende Tabelle zeigt die für "Positionieren absolut" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	1	0	1	MODE 5 = Positionieren absolut
	Bit 3: SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Byte 1 bis 3	Ziel: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Positionieren absolut" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Byte 1 bis 3	Istwert: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1

Fehlerursachen für POS_ERR

Die Fehlerursachen müssen Sie mit dem JOB 15 ermitteln (Aktuelle Werte anzeigen (Seite 2698)).

Die folgende Tabelle zeigt die bei "Positionieren absolut" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Fehlerursache	Abhilfe
1	Unzulässiger MODE im Dosierbetrieb	
2	ERR_2L+ wird angezeigt	Überprüfen Sie die Lastspannung (2L+) an der Klemme 7.
3	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
4	Achse ist nicht synchronisiert (SYNC = 0)	Sie können die Achse synchronisieren: • mit Referenzpunktfahrt • Referenziersignal auswerten • Istwert setzen
5	Endschalter, in dessen Richtung der Antrieb angesteuert wird, ist aktiv	Überprüfen Sie Ihre Schalter und die Verdrahtung sowie die Parameter "DI0 Endschalter Minus (Seite 2711)" und "DI1 Endschalter Plus (Seite 2711)".
7	Starten mit DIR_P = 0 und DIR_M = 0 oder relevantes Steuerbit DIR_P = 0 bzw. DIR_M = 0	
8	Ziel ≥ Ende der Rundachse (bei Inkrementalgeber) bzw. Ziel ≥ Geberbereich (bei SSI-Geber)	

Fehlernummer	Fehlerursache	Abhilfe
9	Positionieren absolut abgebrochen, weil JOB 9 angestoßen wurde (nur bei Inkrementalgeber)	
13	Drehrichtung von Antrieb und Geber unterschiedlich	Überprüfen Sie die Verdrahtung von Antrieb und Geber sowie den Parameter "Drehrichtungsumkehr (Seite 2710)".

JOB-Bearbeitung abbrechen (JOB 0)

JOB-Bearbeitung abbrechen

Wenn Sie JOB 0 aktivieren, dann reagiert das 1PosU folgendermaßen:

- es bricht einen laufenden JOB 9 "Referenziersignal auswerten" ab,
- es bricht den laufenden JOB 10 "Latch-Funktion" ab,
- es setzt einen anstehenden JOB_ERR = 0.

Sie können in jedem Zustand der Achse den JOB 0 aktivieren.

Auswirkung auf die MODEs

MODEs werden durch den JOB 0 nicht beeinflusst.

Steuersignale

Die folgende Tabelle zeigt die für "JOB-Bearbeitung abbrechen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	JOB 0 = JOB-Bearbeitung abbrechen
		0	0	0	0	
	Bit 0: JOB_REQ					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "JOB-Bearbeitung abbrechen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Istwert setzen (JOB 1)

Istwert setzen

"Istwert setzen" ordnet dem angezeigten Istwert eine neue Koordinate zu. Dadurch wird der Arbeitsbereich auf einen anderen Bereich der Achse verschoben.

Bei Inkrementalgeber und nicht aktiviertem Dosierbetrieb wird die Achse synchronisiert.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der neuen Istwertkoordinate und aktivieren Sie JOB 1.

Das 1PosU setzt die vorgegebene Istwertkoordinate auf den in der Rückmeldeschnittstelle (Seite 2728) angezeigten Istwert und setzt das Rückmeldebit SYNC = 1.

Auswirkung auf die MODEs

Die folgende Tabelle zeigt die Auswirkungen von JOB 1 auf die verschiedenen MODEs.

MODE	Was passiert?
Referenzpunktfahrt	Beachten Sie bei Inkrementalgeber und nicht aktiviertem Dosierbetrieb bei der Auswertung der Referenzpunktfahrt, dass das Rückmeldebit SYNC = 1 sofort gesetzt wird. Die Referenzpunktfahrt läuft dennoch weiter.
Tippen	-
Positionieren absolut	Folgende Reaktionen sind möglich: • Differenz zum Ziel ≤ Abschaltdifferenz Abschaltpunkt ist erreicht bzw. übersprungen; es wird direkt abgeschaltet und die Fahrt mit POS_DONE = 1 beendet. In diesem Fall ist unter Umständen das Ziel überfahren. • Differenz zum Ziel ≤ Umschaltdifferenz Umschaltpunkt ist erreicht bzw. übersprungen; es wird direkt von Eilgang auf Schleichgang reduziert. In diesem Fall ist der im Schleichgang zurückgelegte Weg kleiner als (Umschaltdifferenz – Abschaltdifferenz). • Differenz zum Ziel > Umschaltdifferenz Der Antrieb wird mit Eilgang angesteuert, auch wenn er vorher auf Schleichgang umgeschaltet war.
Positionieren relativ	Das vorgegebene Wegstück wird weiter abgefahren.

Steuersignale

Die folgende Tabelle zeigt die für "Istwert setzen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		0	0	0	1	JOB 1 = Istwert setzen
	Bit 0: JOB_REQ					
Byte 5 bis 7	Istwertkoordinate: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Istwert setzen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 3: SYNC
Byte 1 bis 3	Istwert: • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei "Istwert setzen" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
23	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
34	Istwertkoordinate ≥ Ende der Rundachse (bei Inkrementalgeber) bzw. Istwertkoordinate ≥ Geberbereich (bei SSI-Geber)	

Geberbereich verschieben (JOB 2)

Geberbereich verschieben

Die Funktion "Geberbereich verschieben" können Sie nur bei SSI-Geber ausführen.

"Geberbereich verschieben" justiert den Geberwert so, dass der angezeigte Istwert dem tatsächlichen Istwert entspricht. Dazu muss eine laufende Fahrt beendet sein.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit dem Offset und aktivieren Sie JOB 2.

Sie bestimmen den Offset wie folgt:

- Offset = angezeigter Istwert – tatsächlicher Istwert

Ergibt sich ein negativer Offset, rechnen Sie:

- Offset = angezeigter Istwert – tatsächlicher Istwert +
  (Anzahl Umdrehungen × Anzahl Schritte)

Das 1PosU übernimmt den vorgegebenen Offset und zeigt die tatsächliche Istwertkoordinate in der Rückmeldeschnittstelle (Seite 2728) an.

Auswirkung auf die MODEs

MODEs werden durch den JOB 2 nicht beeinflusst.

Steuersignale

Die folgende Tabelle zeigt die für "Geberbereich verschieben" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		0	0	1	0	JOB 2 = Geberbereich verschieben
	Bit 0: JOB_REQ					
Byte 5 bis 7	Offset (0...Geberbereich)					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Geberbereich verschieben" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 1 bis 3	Istwert: • 0 … Geberbereich – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei "Geberbereich verschieben" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
21	Unzulässiger JOB bei Inkrementalgeber	
23	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
26	JOB 2 "Geberbereich verschieben" kann nicht angestoßen werden, da eine Fahrt läuft	
33	bei JOB 2: Offset nicht im Geberbereich	

Abschaltdifferenz ändern (JOB 3)

Abschaltdifferenz ändern

Mit "Abschaltdifferenz ändern" können Sie die Antriebsansteuerung an veränderte Last- und Mechanikbedingungen anpassen.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der neuen Abschaltdifferenz und aktivieren Sie JOB 3.

Das 1PosU übernimmt die vorgegebene Abschaltdifferenz.

Die Abschaltdifferenz bleibt dann bis zu einer neuen Parametrierung (Seite 2707) des 1PosU mit geänderten Parametern gültig.

Auswirkung auf die MODEs

Die folgende Tabelle zeigt die Auswirkungen von JOB 3 auf die verschiedenen MODEs.

MODE	Was passiert?
Referenzpunktfahrt	-
Tippen	
Positionieren absolut	Differenz zum Ziel ≤ Abschaltdifferenz Abschaltpunkt ist erreicht bzw. übersprungen; es wird direkt abgeschaltet und die Fahrt mit POS_DONE = 1 beendet. In diesem Fall ist unter Umständen das Ziel überfahren.
Positionieren relativ	

Steuersignale

Die folgende Tabelle zeigt die für "Abschaltdifferenz ändern" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		0	0	1	1	JOB 3 = Abschaltdifferenz ändern
	Bit 0: JOB_REQ					
Byte 5 bis 7	Abschaltdifferenz: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … 16 777 215					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Abschaltdifferenz ändern" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 0: JOB_ACK

Umschaltdifferenz ändern (JOB 4)

Umschaltdifferenz ändern

Mit "Umschaltdifferenz ändern" können Sie die Antriebsansteuerung an veränderte Last- und Mechanikbedingungen anpassen.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der neuen Umschaltdifferenz und aktivieren Sie JOB 4.

Das 1PosU übernimmt die vorgegebene Umschaltdifferenz.

Die Umschaltdifferenz bleibt dann bis zu einer neuen Parametrierung (Seite 2707) des 1PosU mit geänderten Parametern gültig.

Auswirkung auf die MODEs

Die folgende Tabelle zeigt die Auswirkungen von JOB 4 auf die verschiedenen MODEs.

MODE	Was passiert?
Referenzpunktfahrt	-
Tippen	

MODE	Was passiert?
Positionieren absolut	Folgende Reaktionen sind möglich: • Differenz zum Ziel ≤ Umschaltdifferenz Der Umschaltpunkt ist erreicht bzw. übersprungen; es wird direkt von Eilgang auf Schleichgang reduziert. In diesem Fall ist der im Schleichgang zurückgelegte Weg kleiner als (Umschaltdifferenz – Abschaltdifferenz). • Differenz zum Ziel > Umschaltdifferenz Der Antrieb wird mit Eilgang angesteuert, auch wenn er vorher auf Schleichgang umgeschaltet war.
Positionieren relativ	

Steuersignale

Die folgende Tabelle zeigt die für "Umschaltdifferenz ändern" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		0	1	0	0	JOB 4 = Umschaltdifferenz ändern
	Bit 0: JOB_REQ					
Byte 5 bis 7	Umschaltdifferenz: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … 16 777 215					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Umschaltdifferenz ändern" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 0: JOB_ACK

Referenziersignal auswerten (JOB 9)

Referenziersignal auswerten

Die Funktion "Referenziersignal auswerten" steht Ihnen nur bei Inkrementalgeber und nicht aktiviertem Dosierbetrieb zur Verfügung.

Mit "Referenziersignal auswerten" können Sie die Achse anhand eines externen Referenziersignals während einer laufenden Fahrt in den MODEs "Tippen" und "Positionieren relativ" synchronisieren. Als Referenziersignal stehen Ihnen wahlweise die 3 Digitaleingänge und die Nullmarke (Seite 2678) zur Verfügung.

Die Digitaleingänge DI0 (Endschalter Minus), DI1 (Endschalter Plus) und DI2 (Reduziernocken) können Sie als Öffner oder Schließer parametrieren.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der Referenzpunktkoordinate und aktivieren Sie JOB 9. Das 1PosU setzt das Rückmeldesignal SYNC = 0.

Wenn das 1PosU das Überfahren des parametrierten Referenziersignals in der Referenzierrichtung erkennt, dann ist die Achse synchronisiert. Das 1PosU setzt das Rückmeldesignal SYNC = 1 und ordnet dem Istwert die Referenzpunktkoordinate zu.

Die Referenzierrichtung ergibt sich aus den Parametern "Referenziersignal" und "Referenzschalter":

	Referenzschalter: Reduziernocken nach Minus	**Referenzschalter: Reduziernocken nach Plus**	**Referenzschalter: Endschalter Minus**	**Referenzschalter: Endschalter Plus**
Referenziersignal: Referenzschalter und Nullmarke	Referenzierrichtung Minus	Referenzierrichtung Plus	Referenzierrichtung Plus	Referenzierrichtung Minus
Referenziersignal: Referenzschalter				
Referenziersignal: Nullmarke	Die Referenzierrichtung ist nicht festgelegt. Mit der nächsten Nullmarke wird die Achse synchronisiert.			

Auswirkung auf die MODEs

Die folgende Tabelle zeigt die Auswirkungen von JOB 9 auf die verschiedenen MODEs.

MODE	Was passiert?
Referenzpunktfahrt	Die mit dem JOB 9 übergebene Referenzkoordinate ist gültig.
Tippen	-
Positionieren absolut	Abbruch der Fahrt mit POS_ERR = 1, da SYNC gelöscht ist
Positionieren relativ	-

Steuersignale

Die folgende Tabelle zeigt die für "Referenziersignal auswerten" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	0	0	1	JOB 9 = Referenziersignal auswerten
	Bit 0: JOB_REQ					
Byte 5 bis 7	Referenzpunktkoordinate: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … Ende der Rundachse – 1					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Referenziersignal auswerten" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 0	Bit 3: SYNC
Byte 1 bis 3	Istwert: • Linearachse: 0 … 16 777 215 • Rundachse: 0 … Ende der Rundachse – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei "Referenziersignal auswerten" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
21	Unzulässiger JOB bei SSI-Geber oder im Dosierbetrieb	
23	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
30	Referenzpunktkoordinate ≥ Ende der Rundachse	

Latch-Funktion (JOB 10)

Latch-Funktion

Mit der "Latch-Funktion" können Sie den Istwert bei einer Flanke am Digitaleingang DI2 einmalig abspeichern. Diese Funktion können Sie z. B. zur Kantenerfassung oder Längenmessung benutzen.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der gewünschten Flanke und aktivieren Sie JOB 10.

Wenn das 1PosU die vorgegebene Flanke am Digitaleingang DI2 erkennt, dann speichert es den dazugehörenden Istwert, zeigt ihn als Rückmeldewert (Seite 2728) an und setzt das Rückmeldebit LATCH_DONE = 1.

Danach können Sie die "Latch-Funktion" wieder aktivieren.

Latch-Funktion und Referenzpunktfahrt bzw. "Referenziersignal auswerten"

Synchronisiert das 1PosU an der selben Flanke, speichert es den Istwert, bevor es die Referenzpunktkoordinate zuordnet.

Auswirkung auf die MODEs

MODEs werden durch den JOB 10 nicht beeinflusst.

Steuersignale

Die folgende Tabelle zeigt die für die "Latch-Funktion" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	0	1	0	JOB 10 = Latch-Funktion
	Bit 0: JOB_REQ					
Byte 5	Bit 1: Latch bei negativer Flanke am DI2 Bit 0: Latch bei positiver Flanke am DI2					

Rückmeldesignale

Die folgende Tabelle zeigt die bei der "Latch-Funktion" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 2: LATCH_DONE Bit 1: JOB_ERR Bit 0: JOB_ACK
Byte 5 bis 7	Rückmeldewert: Istwert bei der Flanke am DI2 • bei Inkrementalgeber: – Linearachse: 0 … 16 777 215 – Rundachse: 0 … Ende der Rundachse – 1 • bei SSI-Geber: – 0 … Geberbereich – 1

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei der "Latch-Funktion" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
23	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
36	Flankenauswahl unbekannt	

Drehrichtungsüberwachung einstellen (JOB 11)

Drehrichtungsüberwachung einstellen

Die Funktion "Drehrichtungsüberwachung einstellen" steht Ihnen nicht im Dosierbetrieb zur Verfügung.

Mit "Drehrichtungsüberwachung einstellen" können Sie die Drehrichtungsüberwachung des 1PosU an Ihre Last- und Mechanikbedingungen anpassen.

Die Drehrichtungsüberwachung ist immer aktiv. Das 1PosU erkennt, ob die Drehrichtung von Antrieb und Geber gleich ist. Eine unterschiedliche Drehrichtung von Antrieb und Geber wird bis zur vorgegebenen Wegdifferenz für die Drehrichtungsüberwachung toleriert. Wenn die vorgegebene Wegdifferenz überschritten wird, dann meldet das 1PosU POS_ERR = 1.

Solange Sie den JOB 11 noch nicht aktiviert haben, wird als Wegdifferenz für die Drehrichtungsüberwachung die doppelte Abschaltdifferenz aus den Parametern verwendet. JOB 3 "Abschaltdifferenz ändern" beeinflusst die Wegdifferenz für die Drehrichtungsüberwachung nicht.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit der neuen Wegdifferenz und aktivieren Sie JOB 11.

Das 1PosU übernimmt die vorgegebene Wegdifferenz für die Drehrichtungsüberwachung.

Die Wegdifferenz für die Drehrichtungsüberwachung bleibt dann bis zu einer neuen Parametrierung (Seite 2707) des 1PosU mit geänderten Parametern gültig.

Abschalten der Drehrichtungsüberwachung

Mit der Wegdifferenz 0 ist die Drehrichtungsüberwachung abgeschaltet.

Auswirkung auf die MODEs

MODEs werden durch den JOB 11 nicht beeinflusst.

Steuersignale

Die folgende Tabelle zeigt die für "Drehrichtungsüberwachung einstellen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	0	1	1	JOB 11 = Drehrichtungsüberwachung einstellen
	Bit 0: JOB_REQ					
Byte 5	0					
Byte 6, 7	Wegdifferenz für Drehrichtungsüberwachung: • 0 … 65 535					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Drehrichtungsüberwachung einstellen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei "Drehrichtungsüberwachung einstellen" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
21	Unzulässiger JOB im Dosierbetrieb	
38	Drehrichtungsüberwachung Wegdifferenz > 65 535	

Aktuelle Werte anzeigen (JOB 15)

Aktuelle Werte anzeigen

Sie können folgende Werte in der Rückmeldeschnittstelle als Rückmeldewert anzeigen lassen:

- Restweg
- Istgeschwindigkeit
- Fehlerursachen zu POS_ERR und JOB_ERR

Voreingestellt durch das 1PosU ist als Rückmeldewert der Restweg.

Unabhängig vom angewählten Rückmeldewert zeigt das 1PosU ständig den Istwert in der Rückmeldeschnittstelle an.

Versorgen Sie die Steuerschnittstelle (Seite 2726) mit dem gewünschten Rückmeldewert und aktivieren Sie JOB 15.

Der angewählte Rückmeldewert bleibt dann bis zu einer neuen Parametrierung (Seite 2707) des 1PosU mit geänderten Parametern gültig.

"Aktuelle Werte anzeigen" und Latch-Funktion

Wenn Sie die Latch-Funktion aktivieren, dann setzt das 1PosU den Rückmeldewert = 0 und zeigt den Istwert bei der Flanke am Digitaleingang DI2 an.

Den JOB 15 können Sie erst wieder nach Beenden der Latch-Funktion aktivieren.

Restweg

Das 1PosU berechnet in den MODEs "Positionieren absolut" und "Positionieren relativ" den Abstand zum Ziel als Restweg. Der Restweg ist positiv, solange sich der Istwert vor dem Ziel befindet. Er wird negativ beim Überfahren des Ziels. In den restlichen MODEs ist der Restweg = 0.

Das 1PosU zeigt den Restweg mit Vorzeichen zwischen -8 388 608 und 8 388 607 Schritten an. Negative Werte werden im Zweierkomplement dargestellt. Liegt der tatsächliche Restweg außerhalb dieser Grenzen, wird der Grenzwert angezeigt.

Istgeschwindigkeit

Das 1PosU berechnet die Istgeschwindigkeit als Geberwertänderung in Schritten pro 10 ms. Es zeigt diese zwischen 0 und 16 777 215 an.

Fehlerursachen zu POS_ERR und JOB_ERR

Das 1PosU zeigt die Fehlerursachen zu POS_ERR und JOB_ERR sowie den in der Steuerschnittstelle (Seite 2726) eingetragenen MODE und JOB an.

Auswirkung auf die MODEs

MODEs werden durch den JOB 15 nicht beeinflusst.

Steuersignale

Die folgende Tabelle zeigt die für "Aktuelle Werte anzeigen" relevanten Daten in der Steuerschnittstelle (Seite 2726).

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	1	1	1	JOB 15 = Aktuelle Werte anzeigen
	Bit 0: JOB_REQ					
Byte 5	0: Restweg 1: Istgeschwindigkeit 2: Fehlerursachen zu POS_ERR und JOB_ERR					

Rückmeldesignale

Die folgende Tabelle zeigt die bei "Aktuelle Werte anzeigen" relevanten Daten in der Rückmeldeschnittstelle (Seite 2728).

Adresse	Belegung
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK
Byte 5 bis 7	entsprechend des gewählten Rückmeldewerts: • bei Restweg: –8 388 608 … 8 388 607 • bei Istgeschwindigkeit: 0 … 16 777 215 • bei Fehlerursachen zu POS_ERR und JOB_ERR: – Byte 5: Fehlerursachen (Seite 2705) zu POS_ERR – Byte 6: Fehlerursachen (Seite 2706) zu JOB_ERR – Bit 7.3 … 7.0: MODE (= Bit 0.7 … 0.4 aus den Steuersignalen) – Bit 7.7 … 7.4: JOB (= Bit 4.7 … 4.4 aus den Steuersignalen)

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt die bei "Aktuelle Werte anzeigen" relevanten Fehlernummern mit Fehlerursache und Abhilfemöglichkeiten.

Fehlernummer	Bedeutung	Abhilfe
35	Aktuelle Werte anzeigen: Anwahl unbekannt	
37	Aktuelle Werte anzeigen: JOB 15 bei laufender Latch-Funktion nicht aktivierbar.	

Fehlererkennung / Diagnose

Übersicht zu Fehlern mit 1PosU

Beim Arbeiten mit dem 1PosU können verschiedene Fehler vorkommen. Dabei werden folgende Fehlerarten unterschieden:

- Parametrierfehler (Seite 2701)
- Externe Fehler (Seite 2701)
- Fehler bei der Steuerung von MODEs und JOBs (Seite 2704)

Die Fehler "Lastspannung 2L+ nicht vorhanden", "Kurzschluss Geberversorgung" und "Drahtbruch/Kurzschluss Gebersignale" müssen Sie quittieren (Seite 2704).

Wenn beim Starten von MODEs bzw. Aktivieren von JOBs Fehler auftreten, dann finden Sie hier eine Zusammenstellung:

- Fehlerursachen für POS_ERR (Seite 2705)
- Fehlerursachen für JOB_ERR (Seite 2706)

Parametrierfehler

Parametrierfehler

Die folgende Tabelle zeigt Parametrierfehler mit dem 1PosU.

Parametrierfehler	Reaktion des 1PosU
Ursachen: • Das 1PosU kann vorhandene Parameter nicht als eigene identifizieren. • Der von Ihnen projektierte Steckplatz des 1PosU stimmt nicht mit dem Aufbau überein. Nur bei SSI-Geber: • Unzulässiger Wert im Parameter "Impulse pro Geberumdrehung". • Unzulässiger Wert im Parameter "Umdrehungen". • Impulse pro Geberumdrehung × Umdrehungen ist größer als 4096 × 4096. Nur bei Dosierbetrieb: • Aktivierte "Diagnose Gebersignale" • Aktivierte "Drehrichtungsumkehr" Abhilfe: • Überprüfen Sie Projektierung und Aufbau.	• Das 1PosU ist nicht parametriert und kann seine Funktionen nicht ausführen. • kanalbezogene Diagnose erzeugen

Externe Fehler

Externe Fehler

Die folgende Tabelle zeigt externe Fehler mit dem 1PosU.

Lastspannung 2L+ nicht vorhanden	Reaktion des 1PosU
Ursachen: • Fehlende oder zu geringe Lastspannung 2L+ an der Klemme 7 Abhilfe: • Überprüfen Sie die Verdrahtung. Beseitigen Sie einen eventuellen Kurzschluss. • Quittieren (Seite 2704) Sie den Fehler mit dem Steuerbit EXTF_ACK.	• Die laufende Fahrt wird gestoppt; Start einer neuen Fahrt nicht möglich: – Alle 3 Digitalausgänge werden 0. – Rückmeldebit POS_ERR = 1 – Rückmeldebit POS_DONE = 0 • Rückmeldebit ERR_2L+ = 1 • kanalbezogene Diagnose erzeugen • wartet auf Fehlerquittung EXTF_ACK
Kurzschluss Geberversorgung	**Reaktion des 1PosU**

Ursachen: • Kurzschluss der Geberversorgung, die an den Klemmen 6 und 10 zur Verfügung gestellt wird Abhilfe: • Überprüfen Sie die Verdrahtung und beseitigen Sie den Kurzschluss. • Quittieren (Seite 2704) Sie den Fehler mit dem Steuerbit EXTF_ACK.	• Die laufenden MODEs "Referenzpunktfahrt", "Positionieren relativ" und "Positionieren absolut" werden gestoppt; Start einer neuen Fahrt in diesen MODEs nicht möglich: – Alle 3 Digitalausgänge werden 0. – Rückmeldebit POS_ERR = 1 – Rückmeldebit POS_DONE = 0 • Rückmeldebit ERR_ENCODER = 1 • Rückmeldebit SYNC = 0 • kanalbezogene Diagnose erzeugen • wartet auf Fehlerquittung EXTF_ACK • Der MODE "Tippen" wird durch diesen Fehler nicht beeinflusst. • Der laufende JOB "Referenziersignal auswerten" wird abgebrochen.

Drahtbruch / Kurzschluss Gebersignale	Reaktion des 1PosU
Bei Verwendung eines SSI-Gebers gilt: Voraussetzung: • Zur Fehlererkennung an den Gebersignalen müssen Sie den Parameter "Diagnose Gebersignale" freigeben. Ursachen: • Drahtbruch oder Kurzschluss der Gebersignale an den Klemmen 9 und 13 oder 12 und 16. • Parameter für den SSI-Geber stimmen nicht mit dem angeschlossenen Geber überein. Abhilfe: • Überprüfen Sie die Verdrahtung und beseitigen Sie den Kurzschluss. • Vergleichen Sie die Parametrierung mit den technischen Daten des Gebers. • Quittieren (Seite 2704) Sie den Fehler mit dem Steuerbit EXTF_ACK. **Bei Verwendung eines Inkrementalgebers gilt:** Voraussetzung: • Zur Fehlererkennung an den Signalen A, /A und B, /B bei 5 V-Differenzsignalen bzw. an den Signalen A* und B* bei 24 V-Signalen müssen Sie den Parameter "Diagnose Gebersignale" freigeben. • Zur Fehlererkennung an den Signalen N, /N bei 5 V-Differenzsignalen müssen Sie den Parameter "Diagnose Nullmarkensignale" freigeben. Wenn Sie einen Geber ohne Nullmarke verwenden, dann schalten Sie die Fehlererkennung aus. Bei aktiviertem Dosierbetrieb ist die "Diagnose Nullmarkensignale" nicht möglich. Ursachen: • Nur bei 5 V-Differenzsignalen: Drahtbruch oder Kurzschluss der Gebersignale an den Klemmen 9 und 13 oder 11 und 15 oder 12 und 16. • Flankenfehler der Gebersignale erkannt, so dass das 1PosU keine eindeutige Richtungserkennung durchführen kann. Abhilfe: • Überprüfen Sie die Verdrahtung und beseitigen Sie den Kurzschluss. • Quittieren (Seite 2704) Sie den Fehler mit dem Steuerbit EXTF_ACK.	• Die laufenden MODEs "Positionieren relativ" und "Positionieren absolut" werden gestoppt; Start einer neuen Fahrt in diesen MODEs nicht möglich: – Alle 3 Digitalausgänge werden 0. – Rückmeldebit POS_ERR = 1 – Rückmeldebit POS_DONE = 0 • Rückmeldebit ERR_ENCODER = 1 • Rückmeldebit SYNC = 0 • kanalbezogene Diagnose erzeugen • wartet auf Fehlerquittung EXTF_ACK • Der MODE "Tippen" wird durch diesen Fehler nicht beeinflusst. • Der laufende JOB "Referenziersignal auswerten" wird abgebrochen.

Fehler bei der Steuerung von MODEs und JOBs

Fehler bei der Steuerung von MODEs und JOBs

Die folgende Tabelle zeigt Fehler bei der Steuerung von MODEs und JOBs mit dem 1PosU.

POS_ERR	Reaktion des 1PosU
Ursachen: • Beim Start eines MODEs sind bestimmte Voraussetzungen oder Bedingungen nicht erfüllt.	• Der gestartete MODE wird nicht ausgeführt. • Die laufende Fahrt wird gestoppt: – Alle 3 Digitalausgänge werden 0. – Rückmeldebit POS_ERR = 1 – Rückmeldebit POS_DONE = 0
JOB_ERR	**Reaktion des 1PosU**
Ursachen: • Beim Aktivieren eines JOBs sind bestimmte Voraussetzungen oder Bedingungen nicht erfüllt.	• Der aktivierte JOB wird nicht ausgeführt: – Rückmeldebit JOB_ERR = 1

Fehlerquittung EXTF_ACK

Fehlerquittung EXTF_ACK

Die beseitigten externen Fehler (Seite 2701) "Lastspannung 2L+ nicht vorhanden", "Kurzschluss Geberversorgung" und "Drahtbruch/Kurzschluss Gebersignale" müssen Sie quittieren. Die folgende Tabelle zeigt den dazu notwendigen Ablauf.

Ihre Aktivität	Reaktion des 1PosU
	• Rückmeldebit ERR_2L+ = 1 und/oder • Rückmeldebit ERR_ENCODER = 1
Ihr Steuerungsprogramm erkennt das gesetzte Rückmeldebit ERR_2L+ oder ERR_ENCODER. Führen Sie Ihre applikationsspezifische Fehlerreaktion aus. Beseitigen Sie die Fehlerursache.	
Setzen Sie das Steuerbit (Seite 2726) EXTF_ACK von 0 auf 1	Das 1PosU setzt die Rückmeldebits (Seite 2728) ERR_2L + = 0 und ERR_ENCODER = 0. Sie erkennen daran, dass die Fehlerursache beseitigt und quittiert ist. Bleibt ERR_2L+ = 1 und/oder ERR_ENCODER = 1, ist die Fehlerursache noch nicht beseitigt.
Setzen Sie das Steuerbit (Seite 2726) EXTF_ACK von 1 auf 0.	
Bei ständiger Fehlerquittung (EXTF_ACK = 1) oder im CPU/Master-STOP meldet das 1PosU die Fehler, sobald sie erkannt sind, und löscht die Fehler, sobald sie beseitigt sind.	

Fehlerursachen für POS_ERR

Fehlerursachen für POS_ERR

Die folgende Tabelle zeigt Fehlerursachen und Abhilfemöglichkeiten für POS_ERR.

Fehlernummer	Fehlerursache	Abhilfe
1	MODE unbekannt oder unzulässig	zulässige MODEs sind: • MODE 0 • MODE 1 • MODE 3 (nicht möglich bei SSI-Geber oder im Dosierbetrieb) • MODE 4 • MODE 5 (nicht möglich im Dosierbetrieb)
2	ERR_2L+ wird angezeigt	Überprüfen Sie die Lastspannung (2L+) an der Klemme 7.
3	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
4	Achse ist nicht synchronisiert (SYNC = 0)	SSI-Geber: • Beseitigen Sie den Geberfehler. Bei Inkrementalgeber und nicht aktiviertem Dosierbetrieb können Sie die Achse synchronisieren mit: • Referenzpunktfahrt • Referenziersignal auswerten • Istwert setzen
5	Endschalter, in dessen Richtung der Antrieb angesteuert wird, ist aktiv	Überprüfen Sie Ihre Schalter und die Verdrahtung sowie die Parameter "DI0 Endschalter Minus (Seite 2711)" und "DI1 Endschalter Plus (Seite 2711)".
7	Tippen: DIR_P = 1 und DIR_M = 1 Positionieren absolut: Starten mit DIR_P = 0 und DIR_M = 0 oder relevantes Steuerbit DIR_P = 0 bzw. DIR_M = 0 Positionieren relativ: Starten mit DIR_P = 0 und DIR_M = 0 oder DIR_P = 1 und DIR_M = 1	
8	Positionieren absolut: Ziel ≥ Ende der Rundachse bei Inkrementalgeber oder Ziel ≥ Geberbereich bei SSI-Geber	
9	Positionieren absolut abgebrochen, weil JOB 9 angestoßen wurde	
10	Referenzpunktfahrt: Referenzpunktkoordinate ≥ Ende der Rundachse	
11	Referenzpunktfahrt: kein Referenziersignal bis zum Endschalter bzw. zwischen den Endschaltern gefunden	Überprüfen Sie Ihre Schalter, den Geber und die Verdrahtung.

Fehlernummer	Fehlerursache	Abhilfe
13	Drehrichtung von Antrieb und Geber unterschiedlich	Überprüfen Sie die Verdrahtung von Antrieb und Geber sowie den Parameter "Drehrichtungsumkehr (Seite 2710)".
15	Tippen, Positionieren relativ: Im Dosierbetrieb DIR_M = 1	

Fehlerursachen für JOB_ERR

Fehlerursachen für JOB_ERR

Die folgende Tabelle zeigt Fehlerursachen und Abhilfemöglichkeiten für POS_ERR.

Fehlernummer	Bedeutung	Abhilfe
21	JOB unbekannt oder unzulässig	zulässige JOBs sind: • JOB 0 • JOB 1 • JOB 2 (nur bei SSI-Geber möglich) • JOB 3 • JOB 4 • JOB 9 (nicht bei SSI-Geber oder im Dosierbetrieb (Seite 2667) möglich) • JOB 10 • JOB 11 (nicht im Dosierbetrieb möglich) • JOB 15
23	ERR_ENCODER wird angezeigt	Überprüfen Sie die Verdrahtung des Gebers.
26	JOB 2 "Geberbereich verschieben" kann nicht angestoßen werden, da eine Fahrt läuft	
29	Referenziersignal auswerten: Referenzpunktkoordinate ≥ Ende der Rundachse	
34	Istwert setzen: Istwertkoordinate ≥ Geberbereich	
35	Aktuelle Werte anzeigen: Anwahl unbekannt	
36	Latch-Funktion: Flankenauswahl unbekannt	
37	Aktuelle Werte anzeigen: JOB 15 bei laufender Latch-Funktion nicht aktivierbar.	
38	Drehrichtungsüberwachung Wegdifferenz > 65 535	

CPU/Master-STOP und RESET-Zustand

Verhalten bei CPU/Master-STOP

Die folgende Tabelle zeigt das Verhalten des 1PosU bei CPU/Master-STOP.

Verhalten bei CPU/Master-STOP	Reaktion des 1PosU
• durch NETZ-AUS der CPU / des Bus-Masters oder • durch NETZ-AUS der IM 151 / IM 151 FO oder • durch Ausfall der Bus-Übertragung oder • durch Übergang von RUN nach STOP	• Die laufende Fahrt wird gestoppt. • Alle 3 Digitalausgänge werden 0. • Rückmeldebit (Seite 2728) POS_ERR = 0 • Rückmeldebit (Seite 2728) POS_DONE = 1

Verlassen des Zustands CPU/Master-STOP

Die folgende Tabelle zeigt das Verhalten des 1PosU nach Verlassen des Zustands CPU/ Master-STOP.

Verlassen des Zustands CPU/Master-STOP	Reaktion des 1PosU
• mit NETZ-EIN der CPU / des Bus-Masters oder • mit NETZ-EIN der IM 151 / IM 151 FO oder • nach Ausfall der Bus-Übertragung oder • nach Übergang von STOP nach RUN	• Die Rückmeldeschnittstelle (Seite 2728) des 1PosU bleibt aktuell. • Die Achse bleibt synchronisiert und der Istwert ist aktuell. • Der verschobene Geberbereich bleibt gültig. • Die geänderte Ab- und Umschaltdifferenz und die Wegdifferenz zur Drehrichtungsüberwachung bleiben gültig. • Ein angestoßener JOB 9 "Referenziersignal auswerten (Seite 2693)" und JOB 10 "Latch-Funktion (Seite 2695)" bleiben aktiv. • Der mit JOB 15 angewählte Rückmeldewert (Seite 2698) ist aktuell.

RESET-Zustand des 1PosU

Die folgende Tabelle zeigt das Verhalten des 1PosU im RESET-Zustand.

RESET-Zustand des 1PosU und Ändern der Parameter des 1PosU	Reaktion des 1PosU
• durch Ändern der Parameter des 1PosU und Laden der Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU / den Bus-Master oder • durch NETZ-EIN am Powermodul des 1PosU oder • Stecken des 1PosU unter Spannung	• Die Achse ist synchronisiert und der Istwert entspricht dem aktuellen Geberwert. • Der Geberbereich ist nicht verschoben. • Die Ab- und Umschaltdifferenz wird aus den Parametern übernommen. • Die Wegdifferenz zur Drehrichtungsüberwachung wird auf die doppelte Abschaltdifferenz gesetzt. • JOB 9 "Referenziersignal auswerten (Seite 2693)" und JOB 10 "Latch-Funktion (Seite 2695)" sind nicht aktiv. • Der Restweg wird als Rückmeldewert (Seite 2698) angezeigt.

6.2.7.4 1PosU konfigurieren und parametrieren

Betrieb

Geberschnittstelle auswählen

Geberschnittstelle

Mit der Vorgabe der Geberschnittstelle wählen Sie, mit welcher Signalart Sie das 1PosU ansteuern wollen.

Geberschnittstelle	Beschreibung
Inkremental 5 V (Seite 2709)	Das 1PosU empfängt die Signale von Inkrementalgebern mit 5 V-Differenzsignalen.
Inkremental 24 V (Seite 2714)	Das 1PosU empfängt die Signale von Inkrementalgebern mit 24 V-Signalen.
SSI absolut (Seite 2720)	Das 1PosU empfängt die Signale von Absolutwertgebern (SSI).

Parameter bei Inkrementalgebern mit 5 V-Differenzsignalen

Freigaben

Sammeldiagnose aktivieren

Sammeldiagnose

Die Sammeldiagnose aktivieren Sie im entsprechenden Optionskästchen.

Ein Geberfehler ERR_ENCODER, eine fehlende Lastspannung ERR_2L+ oder ein Parametrierfehler führen bei freigegebener Sammeldiagnose zu einer kanalbezogenen Diagnose.

Diagnose Gebersignale aktivieren

Diagnose der Gebersignale

Die Diagnose der Gebersignale aktivieren Sie im entsprechenden Optionskästchen.

Bei freigegebener "Diagnose Gebersignale" werden die Gebersignale A, /A und B, /B auf Kurzschluss und Drahtbruch überwacht. Bei nicht aktiviertem Dosierbetrieb wird zusätzlich die Signalfolge überwacht. Ein gleichzeitiger Flankenwechsel der Signale A und B führt zu einem Fehler.

Diagnose Nullmarkensignale aktivieren

Diagnose der Nullmarkensignale

Die Diagnose der Nullmarkensignale aktivieren Sie im entsprechenden Optionskästchen.

Bei freigegebener "Diagnose Nullmarkensignale" werden die Nullmarkensignale N, /N auf Kurzschluss und Drahtbruch überwacht.

Geber und Achsen

Dosierbetrieb aktivieren

Dosierbetrieb

Den Dosierbetrieb aktivieren Sie im entsprechenden Optionskästchen.

Im Dosierbetrieb (Seite 2667) wertet das 1PosU nur die Gebersignale A und /A aus. Somit ist nur eine Wegänderung in positive Richtung möglich. Bei jeder positiven Flanke wird der Istwert inkrementiert.

Drehrichtungsumkehr aktivieren

Drehrichtungsumkehr

Die Drehrichtungsumkehr aktivieren Sie im entsprechenden Optionskästchen.

Mit dem Parameter "Drehrichtungsumkehr" können Sie die Drehrichtung des Gebers an die des Antriebs und der Achse anpassen.

Im Dosierbetrieb (Seite 2667) ist die Drehrichtungsumkehr nicht möglich.

Achsart auswählen

Achsarten

Das Positioniermodul 1PosU kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

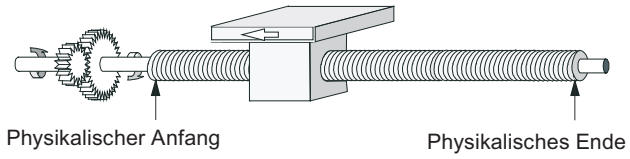

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

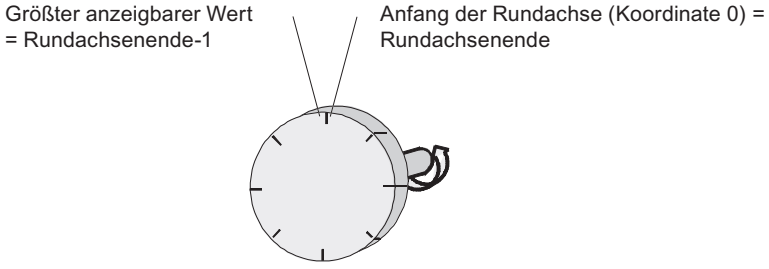

Ende der Rundachse festlegen

Ende der Rundachse

Dieser Parameter ist nur relevant für die Achsart "Rundachse".

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Die Anzeige springt bei

- Überlauf von (Ende der Rundachse – 1) auf 0
- Unterlauf von 0 auf (Ende der Rundachse – 1)

Zulässiger Wertebereich für das Ende der Rundachse: 1 … 16 777 215
Die Eingabe "0" führt zu einem Parametrierfehler.

Digitaleingänge

Endschalter Minus und Endschalter Plus auswählen

Wirkung der Hardware-Endschalter

Die beiden Digitaleingänge DI0 und DI1 werden vom 1PosU als Hardware-Endschalter ausgewertet:

- DI0 ist Endschalter Minus und begrenzt den Arbeitsbereich in Richtung Minus
- DI1 ist Endschalter Plus und begrenzt den Arbeitsbereich in Richtung Plus

Sie können die Hardware-Endschalter **einzeln als Öffner oder Schließer** parametrieren.

Die Hardware-Endschalter werden bei Linearachsen und bei Rundachsen ausgewertet.

Ausgewertet wird immer nur der Hardware-Endschalter, in dessen Richtung der Antrieb angesteuert wird.

Dadurch können Sie nach An- oder Überfahren eines Hardware-Endschalters diesen durch Fahren in die andere Richtung ohne weitere Fehlerquittung wieder verlassen.

Der aktuelle Pegel der Digitaleingänge ist, verzögert um die Aktualisierungsrate, in der Rückmeldeschnittstelle abgebildet.

Die folgende Tabelle zeigt, wie die Hardware-Endschalter in den einzelnen MODEs wirken.

MODE	Wirkung der Hardware-Endschalter
Referenzpunktfahrt	Am Hardware-Endschalter führt das 1PosU eine automatische Richtungsumkehr durch.
Tippen	Am Hardware-Endschalter wird die Bewegung der Achse gestoppt, alle 3 Digitalausgänge werden 0 und das Rückmeldebit POS_ERR wird gemeldet.
Positionieren absolut	
Positionieren relativ	

Starten auf dem Hardware-Endschalter

Die folgende Tabelle zeigt die Reaktion des 1PosU beim Starten eines MODEs auf dem Hardware-Endschalter.

Richtung	Reaktion des 1PosU
Starten in den Arbeitsbereich	Das 1PosU startet den vorgegeben MODE.
Starten weg vom Arbeitsbereich	Das Rückmeldebit POS_ERR = 1 wird gesetzt.

Reduziernocken auswählen

Wirkung des Reduziernockens

Der Digitaleingang DI2 wird vom 1PosU als Reduziernocken oder Latch-Eingang ausgewertet:

- als Reduziernocken bei MODE 3 "Referenzpunktfahrt (Seite 2675)" und bei JOB 9 "Referenziersignal auswerten (Seite 2693)"
- als Latch-Eingang bei JOB 10 "Latch-Funktion (Seite 2695)"

Sie können den Reduziernocken **als Öffner oder Schließer** parametrieren.

Referenzpunktfahrt und Referenziersignal auswerten

Referenziersignal auswählen

Referenziersignal

Der Parameter "Referenziersignal" bestimmt für MODE 3 "Referenzpunktfahrt (Seite 2675)" und JOB 9 "Referenziersignal auswerten (Seite 2693)" den relevanten Schalter bzw. die Kombination Schalter und Nullmarke. Auswählbar sind:

- Referenzschalter und Nullmarke
- Referenzschalter
- Nullmarke

Die Art des Referenzschalters wählen Sie mit dem Parameter "Referenzschalter (Seite 2712)" aus.

Referenzschalter auswählen

Referenzschalter

Der Parameter "Referenzschalter" bestimmt den relevanten Schalter und die Referenzierrichtung, in der der Schalter überfahren werden muss. Auswählbar sind:

- Reduziernocken nach Minus
- Reduziernocken nach Plus
- Endschalter Minus
- Endschalter Plus

Dieser Parameter ist nur relevant, wenn Sie bei "Referenziersignal (Seite 2712)" ausgewählt haben:

- Referenzschalter und Nullmarke
  oder
- Referenzschalter

Startrichtung auswählen

Startrichtung

Der Parameter "Startrichtung" bestimmt die Startrichtung für den MODE 3 "Referenzpunktfahrt (Seite 2675)". Auswählbar sind:

- Plus
- Minus

Antrieb

Ansteuerart auswählen

Ansteuerart

Der Parameter "Ansteuerart" bestimmt die Funktion der Digitalausgänge für die Ansteuerung des Antriebs (Seite 2671). Auswählbar sind:

- Ansteuerart 0:
  - DO0 Fahren Minus
  - DO1 Fahren Plus
  - DO2 Eil-/Schleichgang
- Ansteuerart 1:
  - DO0 Eilgang
  - DO1 Schleichgang (Eilgang ist dann 0)
  - DO2 Fahren Plus (1) / Fahren Minus (0)

Abschaltdifferenz eintragen

Abschaltdifferenz

Die Abschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb vom Schleichgang auf 0 verzögert wird.

Wenn die Abschaltdifferenz ≥ Umschaltdifferenz ist, dann existiert kein Umschaltpunkt. Es wird nicht von Eil- auf Schleichgang verzögert sondern direkt die Reaktion am Abschaltpunkt ausgeführt.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 3 (Seite 2691) können Sie die Abschaltdifferenz ändern.

Umschaltdifferenz eintragen

Umschaltdifferenz

Die Umschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang verzögert wird.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 4 (Seite 2692) können Sie die Umschaltdifferenz ändern.

Verzögerungszeit für den Richtungswechsel eintragen

T_{min} Richtungswechsel

Der Parameter "T_{min} Richtungswechsel" bestimmt die Verzögerungszeit für den Richtungswechsel. Die Digitalausgänge werden abgeschaltet und danach die Umsteuerung der Richtung um T_{min} verzögert durchgeführt. T_{min} wirkt bei jedem Richtungswechsel während einer Fahrt.

T_{min} wirkt nicht beim Starten nach POS_DONE = 1 oder POS_ERR = 1.

Zulässiger Wertebereich:

- 0 bis 255

Ihr Eingabewert wird mit 10 multipliziert. Sie geben T_{min} also in Schritten von 10 ms vor (z. B: 0 ms, 10 ms oder 2550 ms).

Parameter bei Inkrementalgebern mit 24 V-Signalen

Freigaben

Sammeldiagnose aktivieren

Sammeldiagnose

Die Sammeldiagnose aktivieren Sie im entsprechenden Optionskästchen.

Ein Geberfehler ERR_ENCODER, eine fehlende Lastspannung ERR_2L+ oder ein Parametrierfehler führen bei freigegebener Sammeldiagnose zu einer kanalbezogenen Diagnose.

Diagnose Gebersignale aktivieren

Diagnose der Gebersignale

Die Diagnose der Gebersignale aktivieren Sie im entsprechenden Optionskästchen.

Bei freigegebener "Diagnose Gebersignale" führt ein gleichzeitiger Flankenwechsel der Signale A* und B* zu einem Fehler.

Geber und Achsen

Dosierbetrieb aktivieren

Dosierbetrieb

Den Dosierbetrieb aktivieren Sie im entsprechenden Optionskästchen.

Im Dosierbetrieb (Seite 2667) wertet das 1PosU nur das Gebersignal A* aus. Somit ist nur eine Wegänderung in positive Richtung möglich. Bei jeder positiven Flanke wird der Istwert inkrementiert.

Gebereingänge auswählen

Gebereingänge

An den Gebereingängen können unterschiedlichen Sensoren angeschlossen werden:

- P-Schalter/Gegentakt
- M-Schalter

Hinweis

Wenn Sie beim Parameter "Gebereingänge" die Einstellung "M-Schalter" gewählt haben, müssen Sie auch M-schaltende Sensoren verwenden.

Wählen Sie aus der Klappliste die Ausgangsschaltung des eingesetzten Gebers für die Signale A*, B* und N*.

Drehrichtungsumkehr aktivieren

Drehrichtungsumkehr

Die Drehrichtungsumkehr aktivieren Sie im entsprechenden Optionskästchen.

Mit dem Parameter "Drehrichtungsumkehr" können Sie die Drehrichtung des Gebers an die des Antriebs und der Achse anpassen.

Im Dosierbetrieb (Seite 2667) ist die Drehrichtungsumkehr nicht möglich.

Achsart auswählen

Achsarten

Das Positioniermodul 1PosU kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

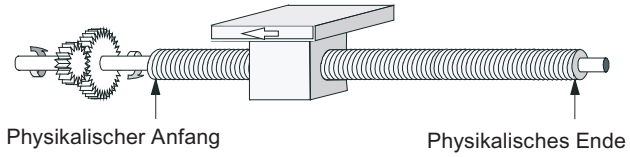

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

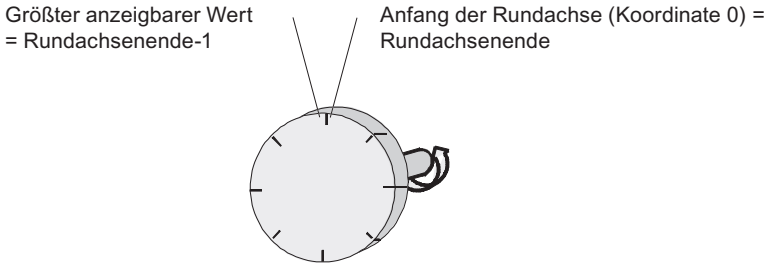

Ende der Rundachse festlegen

Ende der Rundachse

Dieser Parameter ist nur relevant für die Achsart "Rundachse".

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann. Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse (0).

Die Anzeige springt bei

- Überlauf von (Ende der Rundachse – 1) auf 0
- Unterlauf von 0 auf (Ende der Rundachse – 1)

Zulässiger Wertebereich für das Ende der Rundachse: 1 … 16 777 215
Die Eingabe "0" führt zu einem Parametrierfehler.

Digitaleingänge

Endschalter Minus und Endschalter Plus auswählen

Wirkung der Hardware-Endschalter

Die beiden Digitaleingänge DI0 und DI1 werden vom 1PosU als Hardware-Endschalter ausgewertet:

- DI0 ist Endschalter Minus und begrenzt den Arbeitsbereich in Richtung Minus
- DI1 ist Endschalter Plus und begrenzt den Arbeitsbereich in Richtung Plus

Sie können die Hardware-Endschalter **einzeln als Öffner oder Schließer** parametrieren.

Die Hardware-Endschalter werden bei Linearachsen und bei Rundachsen ausgewertet.

Ausgewertet wird immer nur der Hardware-Endschalter, in dessen Richtung der Antrieb angesteuert wird.

Dadurch können Sie nach An- oder Überfahren eines Hardware-Endschalters diesen durch Fahren in die andere Richtung ohne weitere Fehlerquittung wieder verlassen.

Der aktuelle Pegel der Digitaleingänge ist, verzögert um die Aktualisierungsrate, in der Rückmeldeschnittstelle abgebildet.

Die folgende Tabelle zeigt, wie die Hardware-Endschalter in den einzelnen MODEs wirken.

MODE	Wirkung der Hardware-Endschalter
Referenzpunktfahrt	Am Hardware-Endschalter führt das 1PosU eine automatische Richtungsumkehr durch.
Tippen	Am Hardware-Endschalter wird die Bewegung der Achse gestoppt, alle 3 Digitalausgänge werden 0 und das Rückmeldebit POS_ERR wird gemeldet.
Positionieren absolut	
Positionieren relativ	

Starten auf dem Hardware-Endschalter

Die folgende Tabelle zeigt die Reaktion des 1PosU beim Starten eines MODEs auf dem Hardware-Endschalter.

Richtung	Reaktion des 1PosU
Starten in den Arbeitsbereich	Das 1PosU startet den vorgegeben MODE.
Starten weg vom Arbeitsbereich	Das Rückmeldebit POS_ERR = 1 wird gesetzt.

Reduziernocken auswählen

Wirkung des Reduziernockens

Der Digitaleingang DI2 wird vom 1PosU als Reduziernocken oder Latch-Eingang ausgewertet:

- als Reduziernocken bei MODE 3 "Referenzpunktfahrt (Seite 2675)" und bei JOB 9 "Referenziersignal auswerten (Seite 2693)"
- als Latch-Eingang bei JOB 10 "Latch-Funktion (Seite 2695)"

Sie können den Reduziernocken **als Öffner oder Schließer** parametrieren.

Referenzpunktfahrt und Referenziersignal auswerten

Referenziersignal auswählen

Referenziersignal

Der Parameter "Referenziersignal" bestimmt für MODE 3 "Referenzpunktfahrt (Seite 2675)" und JOB 9 "Referenziersignal auswerten (Seite 2693)" den relevanten Schalter bzw. die Kombination Schalter und Nullmarke. Auswählbar sind:

- Referenzschalter und Nullmarke
- Referenzschalter
- Nullmarke

Die Art des Referenzschalters wählen Sie mit dem Parameter "Referenzschalter (Seite 2718)" aus.

Referenzschalter auswählen

Referenzschalter

Der Parameter "Referenzschalter" bestimmt den relevanten Schalter und die Referenzierrichtung, in der der Schalter überfahren werden muss. Auswählbar sind:

- Reduziernocken nach Minus
- Reduziernocken nach Plus
- Endschalter Minus
- Endschalter Plus

Dieser Parameter ist nur relevant, wenn Sie bei "Referenziersignal (Seite 2718)" ausgewählt haben:

- Referenzschalter und Nullmarke
  oder
- Referenzschalter

Startrichtung auswählen

Startrichtung

Der Parameter "Startrichtung" bestimmt die Startrichtung für den MODE 3 "Referenzpunktfahrt (Seite 2675)". Auswählbar sind:

- Plus
- Minus

Antrieb

Ansteuerart auswählen

Ansteuerart

Der Parameter "Ansteuerart" bestimmt die Funktion der Digitalausgänge für die Ansteuerung des Antriebs (Seite 2671). Auswählbar sind:

- Ansteuerart 0:
  - DO0 Fahren Minus
  - DO1 Fahren Plus
  - DO2 Eil-/Schleichgang
- Ansteuerart 1:
  - DO0 Eilgang
  - DO1 Schleichgang (Eilgang ist dann 0)
  - DO2 Fahren Plus (1) / Fahren Minus (0)

Abschaltdifferenz eintragen

Abschaltdifferenz

Die Abschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb vom Schleichgang auf 0 verzögert wird.

Wenn die Abschaltdifferenz ≥ Umschaltdifferenz ist, dann existiert kein Umschaltpunkt. Es wird nicht von Eil- auf Schleichgang verzögert sondern direkt die Reaktion am Abschaltpunkt ausgeführt.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 3 (Seite 2691) können Sie die Abschaltdifferenz ändern.

Umschaltdifferenz eintragen

Umschaltdifferenz

Die Umschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang verzögert wird.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 4 (Seite 2692) können Sie die Umschaltdifferenz ändern.

Verzögerungszeit für den Richtungswechsel eintragen

T_{min} Richtungswechsel

Der Parameter "T_{min} Richtungswechsel" bestimmt die Verzögerungszeit für den Richtungswechsel. Die Digitalausgänge werden abgeschaltet und danach die Umsteuerung der Richtung um T_{min} verzögert durchgeführt. T_{min} wirkt bei jedem Richtungswechsel während einer Fahrt.

T_{min} wirkt nicht beim Starten nach POS_DONE = 1 oder POS_ERR = 1.

Zulässiger Wertebereich:

- 0 bis 255

Ihr Eingabewert wird mit 10 multipliziert. Sie geben T_{min} also in Schritten von 10 ms vor (z. B: 0 ms, 10 ms oder 2550 ms).

Parameter bei SSI-Gebern

Freigaben

Sammeldiagnose aktivieren

Sammeldiagnose

Die Sammeldiagnose aktivieren Sie im entsprechenden Optionskästchen.

Ein Geberfehler ERR_ENCODER, eine fehlende Lastspannung ERR_2L+ oder ein Parametrierfehler führen bei freigegebener Sammeldiagnose zu einer kanalbezogenen Diagnose.

Diagnose Gebersignale aktivieren

Diagnose der Gebersignale

Die Diagnose der Gebersignale aktivieren Sie im entsprechenden Optionskästchen.

Bei freigegebener "Diagnose Gebersignale" werden die Gebersignale D, /D auf Kurzschluss und Drahtbruch überprüft. Zusätzlich erfolgt eine Überwachung des Telegramms (Startbit und Stoppbit).

Geber und Achsen

Geberart auswählen

Geberart

Wählen Sie aus der Klappliste den eingesetzten Absolutwertgeber:

- Singleturn-Geber (SSI-13 Bit)
- Multiturn-Geber (SSI-25 Bit)

Baudrate auswählen

Baudrate

Mit dem Parameter "Baudrate" bestimmen Sie die Geschwindigkeit der Datenübertragung vom SSI-Geber zur Positioniermodul.

Die maximale Baudrate ist von der Leitungslänge abhängig:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz

Inkremente pro Geberumdrehung auswählen

Inkremente pro Geberumdrehung

Der Parameter "Inkremente pro Geberumdrehung" gibt die Anzahl der Inkremente an, die ein Absolutwertgeber je Umdrehung abgibt.

Für die Grenzen muss zwischen den einzelnen Gebermodellen unterschieden werden. Es sind nur Werte in 2er Potenzen auswählbar.

- Singleturn-Geber (Anzahl Umdrehungen = 1) mit 13 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192
- Multiturn-Geber (Anzahl Umdrehungen > 1) mit 25 Bit Telegrammlänge:
  - minimaler Wert = 4
  - maximaler Wert = 8192

Anzahl Geberumdrehungen auswählen

Anzahl Geberumdrehungen

Mit diesem Parameter geben Sie die Anzahl der Umdrehungen an, die mit dem Geber möglich sind.

Die Angabe ist nur für Multiturn-Geber relevant. Bei Singleturn-Gebern setzt das 1PosU die Anzahl Umdrehungen gleich 1.

Zulässiger Wertebereich:

- 2 bis 4096 in 2er Potenzen

Drehrichtungsumkehr aktivieren

Drehrichtungsumkehr

Die Drehrichtungsumkehr aktivieren Sie im entsprechenden Optionskästchen.

Mit dem Parameter "Drehrichtungsumkehr" können Sie die Drehrichtung des Gebers an die des Antriebs und der Achse anpassen.

Achsart auswählen

Achsarten

Das Positioniermodul 1PosU kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

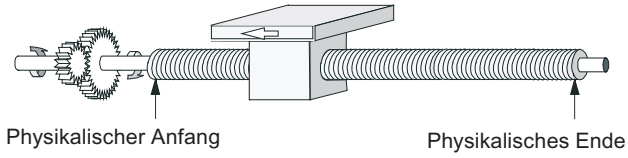

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

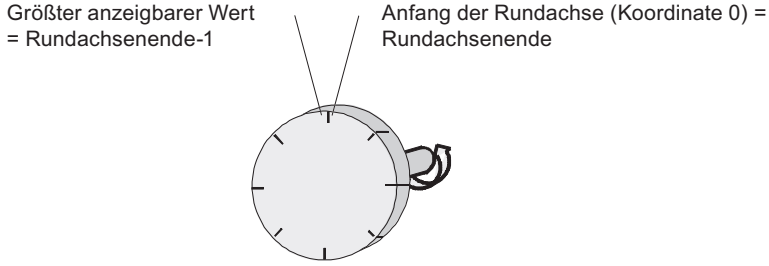

Digitaleingänge

Endschalter Minus und Endschalter Plus auswählen

Wirkung der Hardware-Endschalter

Die beiden Digitaleingänge DI0 und DI1 werden vom 1PosU als Hardware-Endschalter ausgewertet:

- DI0 ist Endschalter Minus und begrenzt den Arbeitsbereich in Richtung Minus
- DI1 ist Endschalter Plus und begrenzt den Arbeitsbereich in Richtung Plus

Sie können die Hardware-Endschalter **einzeln als Öffner oder Schließer** parametrieren.

Die Hardware-Endschalter werden bei Linearachsen und bei Rundachsen ausgewertet.

Ausgewertet wird immer nur der Hardware-Endschalter, in dessen Richtung der Antrieb angesteuert wird.

Dadurch können Sie nach An- oder Überfahren eines Hardware-Endschalters diesen durch Fahren in die andere Richtung ohne weitere Fehlerquittung wieder verlassen.

Der aktuelle Pegel der Digitaleingänge ist, verzögert um die Aktualisierungsrate, in der Rückmeldeschnittstelle abgebildet.

Die folgende Tabelle zeigt, wie die Hardware-Endschalter in den einzelnen MODEs wirken.

MODE	Wirkung der Hardware-Endschalter
Referenzpunktfahrt	Am Hardware-Endschalter führt das 1PosU eine automatische Richtungsumkehr durch.
Tippen	Am Hardware-Endschalter wird die Bewegung der Achse gestoppt, alle 3 Digitalausgänge werden 0 und das Rückmeldebit POS_ERR wird gemeldet.
Positionieren absolut	
Positionieren relativ	

Starten auf dem Hardware-Endschalter

Die folgende Tabelle zeigt die Reaktion des 1PosU beim Starten eines MODEs auf dem Hardware-Endschalter.

Richtung	Reaktion des 1PosU
Starten in den Arbeitsbereich	Das 1PosU startet den vorgegeben MODE.
Starten weg vom Arbeitsbereich	Das Rückmeldebit POS_ERR = 1 wird gesetzt.

Reduziernocken auswählen

Wirkung des Reduziernockens

Der Digitaleingang DI2 wird vom 1PosU als Reduziernocken oder Latch-Eingang ausgewertet:

- als Reduziernocken bei MODE 3 "Referenzpunktfahrt (Seite 2675)" und bei JOB 9 "Referenziersignal auswerten (Seite 2693)"
- als Latch-Eingang bei JOB 10 "Latch-Funktion (Seite 2695)"

Sie können den Reduziernocken **als Öffner oder Schließer** parametrieren.

Antrieb

Ansteuerart auswählen

Ansteuerart

Der Parameter "Ansteuerart" bestimmt die Funktion der Digitalausgänge für die Ansteuerung des Antriebs (Seite 2671). Auswählbar sind:

- Ansteuerart 0:
  - DO0 Fahren Minus
  - DO1 Fahren Plus
  - DO2 Eil-/Schleichgang
- Ansteuerart 1:
  - DO0 Eilgang
  - DO1 Schleichgang (Eilgang ist dann 0)
  - DO2 Fahren Plus (1) / Fahren Minus (0)

Abschaltdifferenz eintragen

Abschaltdifferenz

Die Abschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb vom Schleichgang auf 0 verzögert wird.

Wenn die Abschaltdifferenz ≥ Umschaltdifferenz ist, dann existiert kein Umschaltpunkt. Es wird nicht von Eil- auf Schleichgang verzögert sondern direkt die Reaktion am Abschaltpunkt ausgeführt.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 3 (Seite 2691) können Sie die Abschaltdifferenz ändern.

Umschaltdifferenz eintragen

Umschaltdifferenz

Die Umschaltdifferenz (Seite 2665) definiert den Abstand zum Ziel, an dem der Antrieb von Eilgang auf Schleichgang verzögert wird.

Zulässiger Wertebereich:

- 0 bis 65 535

Mit dem JOB 4 (Seite 2692) können Sie die Umschaltdifferenz ändern.

Verzögerungszeit für den Richtungswechsel eintragen

T_{min} Richtungswechsel

Der Parameter "T_{min} Richtungswechsel" bestimmt die Verzögerungszeit für den Richtungswechsel. Die Digitalausgänge werden abgeschaltet und danach die Umsteuerung der Richtung um T_{min} verzögert durchgeführt. T_{min} wirkt bei jedem Richtungswechsel während einer Fahrt.

T_{min} wirkt nicht beim Starten nach POS_DONE = 1 oder POS_ERR = 1.

Zulässiger Wertebereich:

- 0 bis 255

Ihr Eingabewert wird mit 10 multipliziert. Sie geben T_{min} also in Schritten von 10 ms vor (z. B: 0 ms, 10 ms oder 2550 ms).

Steuer- und Rückmeldeschnittstelle des 1PosU

Steuerschnittstelle des 1PosU

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1PosU.

Adresse	Belegung					
Byte 0	Bit 0.7 … 0.4 stehen für die MODEs					
	Bit	7	6	5	4	
		0	0	0	0	MODE 0 = Stoppen
		0	0	0	1	MODE 1 = Tippen
		0	0	1	1	MODE 3 = Referenzpunktfahrt
		0	1	0	0	MODE 4 = Positionieren relativ
		0	1	0	1	MODE 5 = Positionieren absolut
	Bit 3: SPEED (SPEED = 0 ist Schleichgang; SPEED = 1 ist Eilgang) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Byte 1 bis 3	bei MODE 3 = Referenzpunktfahrt: Referenzpunktkoordinate					
	bei MODE 4 = Positionieren relativ: Wegstück					
	bei MODE 5 = Positionieren absolut: Ziel					

Adresse	Belegung					
Byte 4	Bit 4.7 … 4.4 stehen für die JOBs					
	Bit	7	6	5	4	
		0	0	0	0	JOB 0 = JOB-Bearbeitung abbrechen
		0	0	0	1	JOB 1 = Istwert setzen
		0	0	1	0	JOB 2 = Geberbereich verschieben (nur bei SSI-Geber)
		0	0	1	1	JOB 3 = Abschaltdifferenz ändern
		0	1	0	0	JOB 4 = Umschaltdifferenz ändern
		1	0	0	1	JOB 9 = Referenziersignal auswerten
		1	0	1	0	JOB 10 = Latch-Funktion
		1	0	1	1	JOB 11 = Drehrichtungsüberwachung einstellen
		1	1	1	1	JOB 15 = Aktuelle Werte anzeigen
	Bit 3: EXTF_ACK Bit 2: reserviert = 0 Bit 1: reserviert = 0 Bit 0: JOB_REQ					
Byte 5 bis 7	entsprechend des gewählten JOBs: • bei JOB 1 = Istwertkoordinate • bei JOB 2 = Offset • bei JOB 3 = Abschaltdifferenz • bei JOB 4 = Umschaltdifferenz • bei JOB 9 = Referenzpunktkoordinate • bei JOB 10 – Byte 5: Bit 0 = Latch bei positiver Flanke am DI2 – Byte 5: Bit 1 = Latch bei negativer Flanke am DI2 • bei JOB 11 = Wegdifferenz für Drehrichtungsüberwachung • bei JOB 15 – Byte 5: 0 = Restweg – Byte 5: 1 = Istgeschwindigkeit – Byte 5: 2 = Fehlerinformation					

Rückmeldeschnittstelle des 1PosU

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1PosU.

Adresse	Belegung
Byte 0	Bit 7: ERR_ENCODER Bit 6: STATUS DO 2 Bit 5: STATUS DO 1 Bit 4: STATUS DO 0 Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Byte 1 bis 3	Istwert
Byte 4	Bit 7: ERR_2L+ Bit 6: STATUS DI 2 Reduziernocken Bit 5: STATUS DI 1 Endschalter Plus Bit 4: STATUS DI 0 Endschalter Minus Bit 3: reserviert = 0 Bit 2: LATCH_DONE Bit 1: JOB_ERR Bit 0: JOB_ACK
Byte 5 bis 7	Rückmeldewert

6.2.7.5 Diagnose des 1PosU

Diagnose durch LED-Anzeige

LED-Anzeige am 1PosU

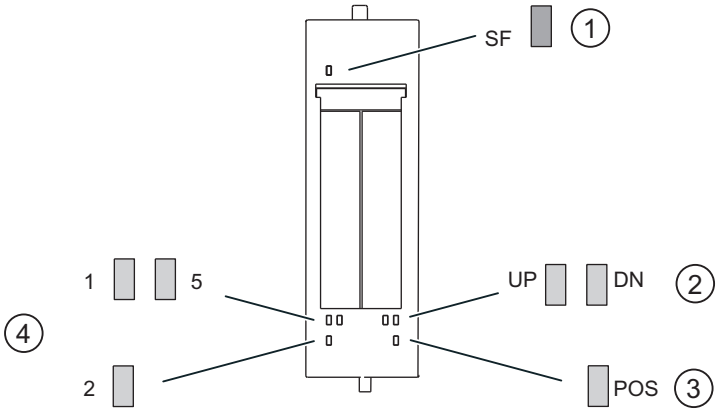

① Sammelfehler (rot)
② Statusanzeige Istwertänderung (grün)
③ Positionierung läuft (grün)
④ Statusanzeigen für Digitaleingänge (grün)

Status- und Fehleranzeigen durch LEDs am 1PosU

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 1PosU.

Ereignis (LED)							Ursache	Maßnahme
SF	1	5	2	UP	DN	POS		
ein							Keine Parametrierung. Diagnosemeldung liegt vor.	Überprüfen Sie die Parametrierung. Werten Sie die Diagnose aus.
	ein						DI 0 ist aktiviert.	
		ein					DI 1 ist aktiviert.	
			ein				DI 2 ist aktiviert.	
				ein			bei Istwertänderung von kleineren zu größeren Istwerten	
					ein		bei Istwertänderung von größeren zu kleineren Istwerten	
						ein	Die Positionierung läuft und einer der 3 Digitalausgänge ist gesetzt.	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

1PosU Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 1PosU.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Geberversorgung.	Überprüfen Sie die Verdrahtung zum Geber. Korrektur der Prozessverdrahtung.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.
17_D	10001: Lastspannung 2L+ fehlt	Versorgungsspannung nicht vorhanden oder zu niedrig.	Korrektur der Prozessverdrahtung. Überprüfen Sie die Versorgungsspannung.
26_D	11010: externer Fehler	Leitungsbruch / Kurzschluss der Gebersignale. Drahtbruch des Geberkabels bzw. Geberkabel ist nicht angeschlossen. Geber ist defekt oder es liegen Störungen vor. Geberart, Baudrate, Monoflopzeit entsprechen nicht dem angeschlossenen Geber; Programmierbare Geber entsprechen nicht den Einstellungen auf dem Modul.	Korrektur der Prozessverdrahtung Korrektur der Parametrierung Austausch des Gebers

6.2.8 ET 200S 1STEP 5V einsetzen

6.2.8.1 Anschlussbelegung des 1STEP 5V

Verdrahtungsregeln

Die Leitungen (Klemmen 1 und 5 sowie Klemmen 4 und 8) zum Leistungsteil müssen geschirmt und paarweise verdrillt sein. Der Schirm muss beidseitig aufgelegt werden. Verwenden Sie hierzu die Schirmauflage (siehe im Anhang der Betriebsanleitung Dezentrales Peripheriesystem ET 200S (http://support.automation.siemens.com/WW/view/de/1144348)).

Anschlussbelegung

Die folgende Tabelle zeigt die Anschlussbelegung für das 1STEP 5V.

Ansicht	Anschlussbelegung	Bemerkungen
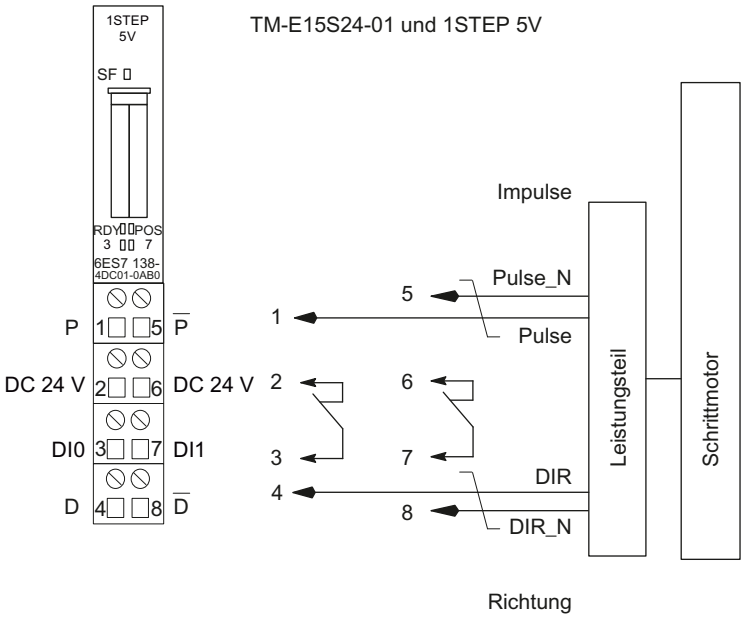		DC 24 V: Lastspannung DI0: externe Impulsfreigabe, externer STOP oder Endschalter in Richtung vorwärts / rückwärts DI1: Referenzschalter oder Referenzschalter und Endschalter in Richtung vorwärts / rückwärts P, /P und D, /D sind Signale nach RS 422.
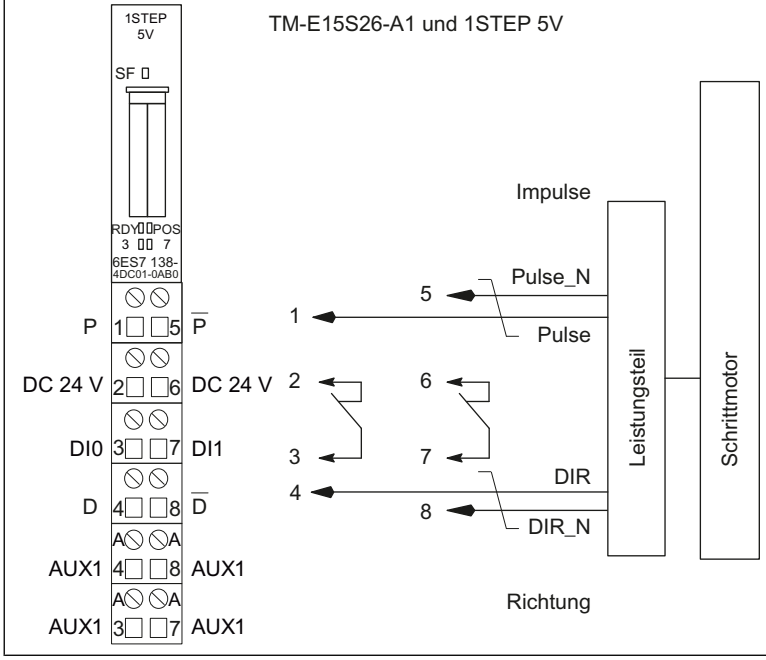		DC 24 V: Lastspannung DI0: externe Impulsfreigabe, externer STOP oder Endschalter in Richtung vorwärts / rückwärts DI1: Referenzschalter oder Referenzschalter und Endschalter in Richtung vorwärts / rückwärts P, /P und D, /D sind Signale nach RS 422.

6.2.8.2 Grundlagen des Positionierens mit Schrittmotor

Zusammenspiel der Komponenten

Nachfolgend erfahren Sie, wie sich die einzelnen Komponenten für eine Positionierung mit Schrittmotor beeinflussen.

1STEP 5V

Das 1STEP 5V erzeugt Impulse und ein Richtungssignal für Schrittmotor-Leistungsteile. Die Anzahl der ausgegebenen Impulse bestimmt die Länge des Verfahrwegs. Die Impulsfrequenz ist ein Maß für die Geschwindigkeit. Die Änderung der Impulsfrequenz pro Zeiteinheit (Sekunde) ist ein Maß für die Beschleunigung bzw. Verzögerung. Die Arbeitsweise des 1STEP 5V ist durch seine Parameter und Einstellungen beeinflussbar.

Leistungsteil für Schrittmotoren

Das Leistungsteil ist das Bindeglied zwischen 1STEP 5V und Schrittmotor. Das 1STEP 5V sendet RS 422-Differenzsignale für Frequenz und Richtung. Diese Signale werden im Leistungsteil in Motorströme umgewandelt, die die Motorbewegungen mit höchster Präzision steuern.

Schrittmotoren

Schrittmotoren werden zum Positionieren von Achsen eingesetzt. Sie sind die einfache und kostengünstige Lösung für hochgenaue Positionieraufgaben in weiten Leistungsbereichen.

Die Welle eines Schrittmotors dreht sich bei jedem Impuls um einen bestimmten Winkel, bei schnellen Impulsfolgen geht diese Schrittbewegung in eine stetige Drehbewegung über.

Parameter und Einstellungen

Benötigte Informationen

Um das optimale Zusammenspiel der einzelnen Komponenten zu erreichen, müssen Sie dem 1STEP 5V Informationen zur Verfügung stellen:

- **einmalig; bei der Projektierung der Parameter mit Ihrer Projektiersoftware**
  - Basisfrequenz (Seite 2736) F_b
  - Multiplikator n (Seite 2734) für die Einstellung der Start-Stop-Frequenz F_{ss}
  - Multiplikator i (Seite 2734) für die Einstellung der Beschleunigung / Verzögerung
  - Funktion und Verhalten der Digitaleingänge (Seite 2755)
  - Verfahrbereich (Seite 2751)
- **im Betrieb; Bewegen des Motors durch einen Fahrauftrag in Ihrem Anwenderprogramm**
  - Multiplikator G (Seite 2734) für die Geschwindigkeit / Ausgangsfrequenz F_a
  - Reduzierfaktor R (Seite 2736)
  - Wegstrecke, Position oder Frequenz
  - Betriebsart
  - Richtungsvorgabe (Fahrauftrag) zum Start
- **im Betrieb; zum Anpassen an unterschiedliche Lastverhältnisse als Parametrierauftrag in Ihrem Anwenderprogramm**
  - Basisfrequenz (Seite 2736) F_b
  - Multiplikator n (Seite 2734) für die Einstellung der Start-Stop-Frequenz F_{ss}
  - Multiplikator i (Seite 2734) für die Einstellung der Beschleunigung / Verzögerung

Verfahrkurve des 1STEP 5V

Verfahrkurve

Eine Schrittmaßfahrt wird vom 1STEP 5V nach der folgenden Verfahrkurve ausgeführt. Die Eckgrößen (Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung / Verzögerung) der Verfahrkurve bildet das 1STEP 5V mit einer von Ihnen gewählten Basisfrequenz (Seite 2736).

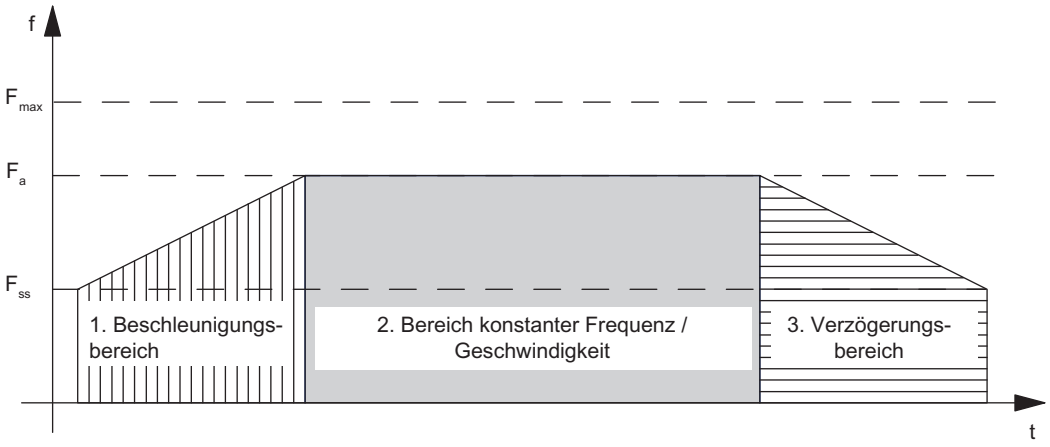

F_{ss} = Start-Stop-Frequenz
F_a = Ausgangsfrequenz
F_{max} = maximale Frequenz des Schrittmotors

Start-Stop-Frequenz F_{ss}

Die Start-Stop-Frequenz F_{ss} ist die Frequenz, auf die der Motor unter Belastung aus dem Stillstand beschleunigt werden kann.

Die Größe von F_{ss} hängt vom Trägheitsmoment der Last ab. Sie kann am einfachsten durch Ausprobieren ermittelt werden.

Die Start-Stop-Frequenz F_{ss} ist gleichzeitig die minimale Ausgangsfrequenz F_a, mit der Sie den Schrittmotor bewegen können.

Einstellen der Start-Stop-Frequenz F_{ss}

Das 1STEP 5V ermöglicht durch Parametrierung eine gestufte Einstellung der Start-Stop-Frequenz F_{ss}. Dazu wählen Sie den Multiplikator n zwischen 1 und 255, der mit der Basisfrequenz F_b multipliziert wird. Mit dem Reduzierfaktor R (1 oder 0,1) im Fahrauftrag können Sie die Start-Stop-Frequenz F_{ss} nochmals absenken.

Die Start-Stop-Frequenz berechnet sich nach der Formel:

$F_{ss} = F_b \times n \times R$

Nähere Informationen dazu finden Sie in der Tabelle (Seite 2736) mit den Bereichen für Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung.

Maximale Frequenz / Geschwindigkeit der Achse F_{max}

Bei der Auswahl Ihres Schrittmotors müssen Sie Folgendes beachten:

Die maximale Frequenz / Geschwindigkeit ist durch Ihre Anwendung bestimmt. Bei dieser Frequenz muss der Motor noch ein genügend großes Drehmoment aufbringen, um seine Last zu bewegen.

Beachten Sie, dass hier nicht die maximal mögliche Frequenz gemeint ist, die der Motor oder das Leistungsteil verkraften können.

Mit der entsprechenden Kennlinie können Sie die maximale Frequenz F_{max} ermitteln. Das folgende Bild zeigt die Drehmomenten-Kennlinie eines Schrittmotors.

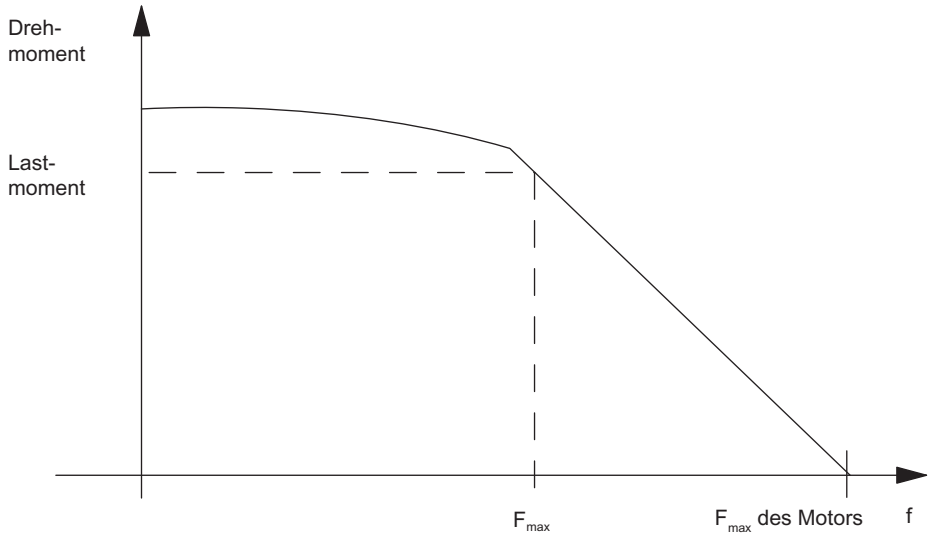

Ausgangsfrequenz / Geschwindigkeit F_a

Die Ausgangsfrequenz kann für jede Fahrt unterschiedlich eingestellt werden.

Beachten Sie bei der Wahl der Ausgangsfrequenz auch die minimale Impulsdauer Ihres Leistungsteils (siehe Formel und Tabelle (Seite 2736) mit den Bereichen für Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung).

Ist die gewählte Ausgangsfrequenz kleiner als die eingestellte Start-Stop-Frequenz F_{ss}, wird die Ausgangsfrequenz vom 1STEP 5V auf die Start-Stop-Frequenz F_{ss} gesetzt.

Sie müssen F_a immer kleiner als F_{max} wählen.

Einstellen der Ausgangsfrequenz / Geschwindigkeit F_a

Das 1STEP 5V ermöglicht eine gestufte Einstellung der Ausgangsfrequenz F_a. Dazu wählen Sie den Multiplikator G zwischen 1 und 255, der mit der Basisfrequenz F_b multipliziert wird. Mit dem Reduzierfaktor R (1 oder 0,1) im Fahrauftrag können Sie die Ausgangsfrequenz F_a nochmals absenken.

Die Ausgangsfrequenz berechnet sich nach der Formel:

$F_a = F_b \times G \times R$

Nähere Informationen dazu finden Sie in der Tabelle (Seite 2736) mit den Bereichen für Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung.

Beschleunigung / Verzögerung a

Die maximal zulässige Beschleunigung / Verzögerung hängt von der zu bewegenden Last ab.

Der Motor muss ein genügend großes Drehmoment aufbringen, um die Last ohne Schrittverlust zu beschleunigen oder zu verzögern.

Weitere Kriterien zur Festlegung der Beschleunigung / Verzögerung, wie z. B. sanftes Anfahren und Abbremsen, müssen Sie je nach Anwendung ebenfalls beachten.

Einstellen der Beschleunigung / Verzögerung a

Das 1STEP 5V ermöglicht durch Parametrierung eine gestufte Einstellung der Beschleunigung / Verzögerung über den Multiplikator i.

Während der Beschleunigungsphase wird die Frequenz, ausgehend von der Start-Stop-Frequenz F_{ss}, kontinuierlich erhöht, bis die Ausgangsfrequenz F_a erreicht ist.

Das Zeitintervall für die kontinuierliche Erhöhung der Frequenz ist gestuft einstellbar. Dazu wählen Sie einen Multiplikator i zwischen 1 und 255.

In der Verzögerungsphase wird die Ausgangsfrequenz auf die gleiche Weise verringert.

Mit dem Reduzierfaktor R (1 oder 0,1) im Fahrauftrag können Sie die Beschleunigung / Verzögerung a nochmals absenken.

Die Beschleunigung / Verzögerung berechnet sich nach der Formel:

$a = F_b \times R / (i \times 0{,}128\ \text{ms})$

Nähere Informationen dazu finden Sie in der Tabelle (Seite 2736) mit den Bereichen für Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung.

Einstellen der Basisfrequenz

Einleitung

Das 1STEP 5V ermöglicht durch Parametrierung eine gestufte Einstellung der Basisfrequenz.

Die Basisfrequenz bestimmt den Bereich für die Start-Stop-Frequenz, die Ausgangsfrequenz und die Beschleunigung.

Vorgehen

1. Wählen Sie nach der Priorität Ihrer Anforderung aus der folgenden Tabelle einen passenden Bereich entweder der Start-Stop-Frequenz F_{ss} und der Ausgangsfrequenz F_a oder der Beschleunigung a nach folgenden Kriterien:
   - Bereich Start-Stop-Frequenz F_{ss},
     z. B. für möglichst schnelles Anfahren und Bremsen
   - Bereich Ausgangsfrequenz F_a,
     z. B. für möglichst genaue Geschwindigkeitseinstellung
   - Bereich Beschleunigung a,
     z. B. für möglichst schnelle Positioniervorgänge
2. Bestimmen Sie die Basisfrequenz F_b über die Tabelle.
   Zur Optimierung der Basisfrequenz F_b gehen Sie folgendermaßen vor:
3. Überprüfen Sie, ob die entsprechenden anderen Werte Ihre Anforderungen abdecken. Gegebenenfalls wählen Sie eine andere Basisfrequenz F_b, bei der Ihre Anforderungen besser erfüllt werden.
4. Bestimmen Sie die Multiplikatoren zur Einstellung der Ausgangsfrequenz F_a, der Beschleunigung / Verzögerung a und der Start-Stop-Frequenz F_{ss}.
5. Entnehmen Sie den zugehörigen Reduzierfaktor R aus der Tabelle.

Die folgende Tabelle zeigt die Bereiche für Start-Stop-Frequenz, Ausgangsfrequenz und Beschleunigung.

Basisfrequenz F_b in Hz	Reduzierfak-tor R	Bereich Start-Stop-Frequenz F_{ss} Ausgangsfrequenz F_a in Hz	Bereich Beschleunigung a in Hz/ms
		Formel: $F_{ss} = F_b \times n \times R$ $F_a = F_b \times G \times R$	Formel: $a = F_b \times R / (i \times 0{,}128\ ms)$
4	0,1	0,4 … 102	0,01 … 3,13
8	0,1	0,8 … 204	0,02 … 6,25
20	0,1	2 … 510	0,06 … 15,6
4	1	4 … 1020	0,12 … 31,3
8	1	8 … 2040	0,25 … 62,5
20	1	20 … 5100	0,61 … 156
40	1	40 … 10200	1,23 … 313
80	1	80 … 20400	2,45 … 625
200	1	200 … 51000	6,13 … 1563
400	1	400 … 102000	12,25 … 3125
800	1	800 … 204000	24,51 … 6250

Basisfrequenz F_b in Hz	Reduzierfaktor R	Bereich Start-Stop-Frequenz F_{ss} Ausgangsfrequenz F_a in Hz	Bereich Beschleunigung a in Hz/ms
		Formel: $F_{ss} = F_b \times n \times R$ $F_a = F_b \times G \times R$	Formel: $a = F_b \times R / (i \times 0{,}128\ ms)$
2000 [1]	1	2000 … 510000	61,27 … 15625
F_b = Basisfrequenz F_{ss} = Start-Stop-Frequenz F_a = Ausgangsfrequenz a = Beschleunigung / Verzögerung R = Reduzierfaktor n = Multiplikator für gestufte Einstellung der Start-Stop-Frequenz G = Multiplikator für gestufte Einstellung der Ausgangsfrequenz i = Multiplikator für gestufte Einstellung der Beschleunigung / Verzögerung			

[1] ab 6ES7138-4DD01-0AB0

Die minimale Impulsdauer ergibt sich anhand der eingestellten Ausgangsfrequenz F_a und berechnet sich nach der Formel $T_{Puls} = 1 / (2 \times F_a)$.

6.2.8.3 Funktionen des 1STEP 5V

Funktionsübersicht

Übersicht

Die Aufgabe des 1STEP 5V ist die Positionierung eines Antriebs auf bestimmte vorgegebene Ziele (Schrittmaßfahrten) und das endlose Fahren mit vorgebbaren Frequenzen (Drehzahlbetrieb).

Für diese Aufgabe stehen Ihnen folgende Funktionen zur Verfügung:

- Referenzpunktfahrt (Seite 2740): Die Achse wird auf einen Bezugspunkt synchronisiert
- Referenzpunkt setzen (Seite 2745) (ab 6ES7138-4DC01-0AB0): Der aktuellen Position wird ein Wert zugewiesen.
- Schrittmaßfahrt relativ (Seite 2746) (relativ Positionieren): Die Achse wird um eine vorgegebene Wegstrecke verfahren
- Schrittmaßfahrt absolut (Seite 2747) (absolut Positionieren) (ab 6ES7138-4DC01-0AB0): Die Achse wird an eine vorgegebene Position gefahren.
- Drehzahlbetrieb (Seite 2748) (ab 6ES7138-4DC01-0AB0): Der Antrieb wird mit einer flexibel vorgebbaren Geschwindigkeit (Impulsfrequenz) bewegt.
- Fahrauftrag anhalten (Seite 2749)
- Parameter ändern im Betrieb (Seite 2754)

Wie Sie die Funktionen ansteuern, finden Sie in der Beschreibung der Steuerschnittstelle (Seite 2762).

Starten eines Fahrauftrags

Starten des Fahrauftrags

Das folgende Bild zeigt den Ablauf von der Startanforderung für einen Fahrauftrag bis zum Beenden der Fahrt.

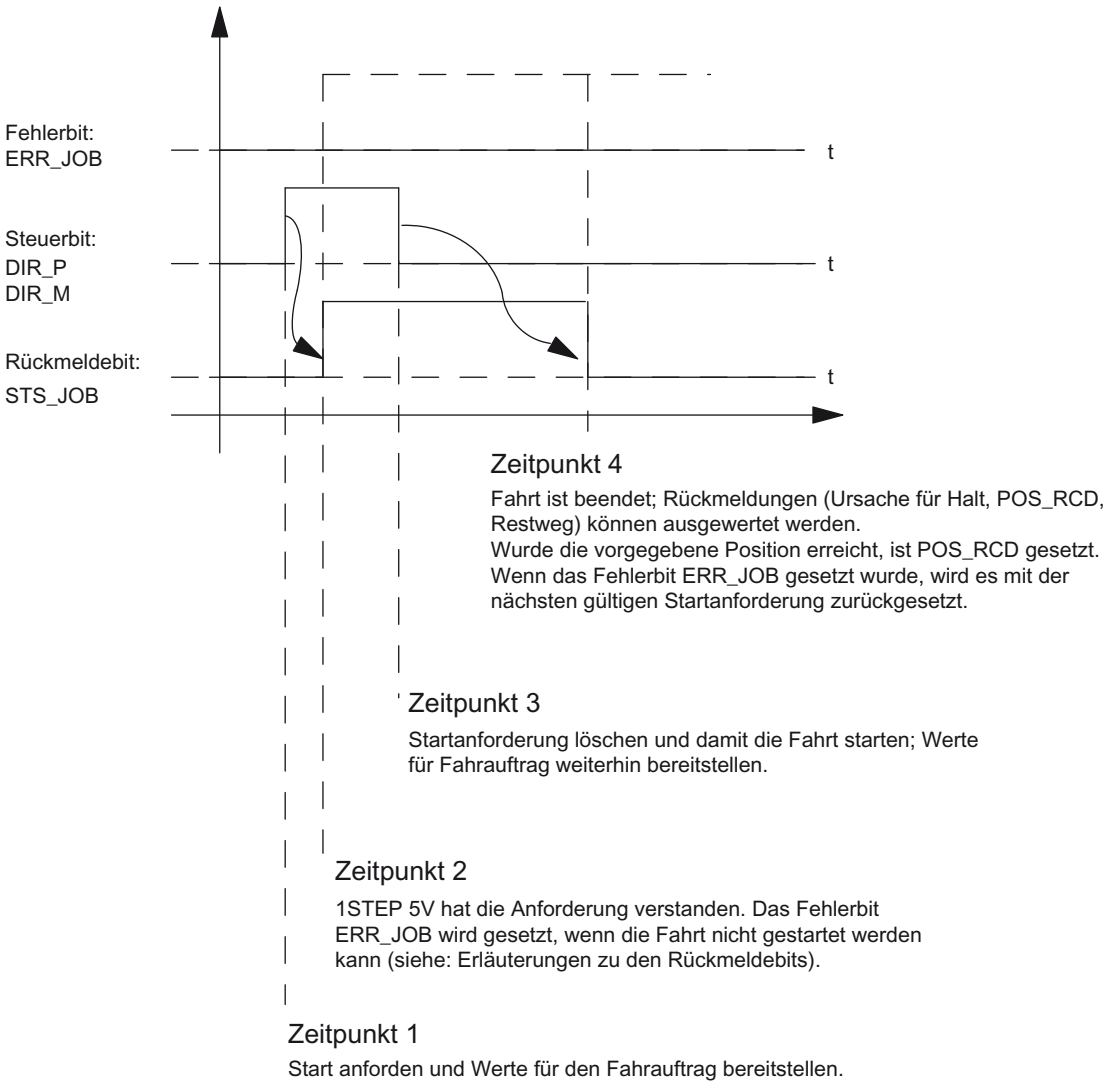

Auswertung des Fehlerbits ERR_JOB

Sobald das Rückmeldebit (Seite 2766) STS_JOB zum Zeitpunkt 4 gelöscht ist, werten Sie das Fehlerbit ERR_JOB aus. Beachten Sie, dass das Rückmeldebit STS_JOB erst gelöscht wird, wenn die Steuerbits (Seite 2762) DIR_P, DIR_M und C_PAR gelöscht sind.

Referenzpunktfahrt

Funktionsbeschreibung

Der Referenzpunkt markiert den Bezugspunkt Ihres Antriebssystems (Referenznocken) für die folgenden Fahraufträge. Sie können den Referenzpunkt ermitteln, wenn Sie z. B. einen Initiator am Referenznocken montieren und am Digitaleingang DI1 anschließen.

Das 1STEP 5V sorgt für die schrittgenaue Reproduzierbarkeit des Referenzpunktes, indem er immer aus der gleichen Richtung angefahren wird. Diese Richtung legen Sie fest, indem Sie die Referenzpunktfahrt immer in die gleiche Richtung starten.

Fahrauftrag für Referenzpunktfahrt

Der Fahrauftrag enthält folgende Informationen:

- Multiplikator G für die Geschwindigkeit / Ausgangsfrequenz F_a
- Reduzierfaktor R für die parametrierte Basisfrequenz F_b
- Position des Referenzpunktes
- Betriebsart = 1 für Referenzpunktfahrt
- Anhalten am Referenznocken (Seite 2749)
- Richtungsvorgabe als Start (Seite 2762)

Hinweis

Das 1STEP 5V überprüft die vorgegebene Position auf Grenzwerte (minimal 0 und maximal 16777215). Der Endwert kann parametriert werden.

Hinweis

Wenn Sie das Verhalten des Digitaleingangs (Seite 2755) DI1 (7) als "Referenzschalter und Endschalter" parametriert haben, dann wählt das 1STEP 5V die Startrichtung selbständig in Richtung Endschalter, unabhängig von der Richtungsvorgabe im Fahrauftrag.

Hinweis

Status der Endschalter

Bei der Verwendung von Endschaltern ist es möglich, dass diese beim Starten der Referenzpunktfahrt noch nicht den korrekten Status liefern. Ein solcher Fall kann z. B. auftreten, wenn die Endschalter als Öffner parametriert sind (aktiv bei 0-Pegel) und die Spannungsversorgung der Endschalter vor dem Start der Referenzpunktfahrt noch nicht vorhanden ist.

Dies kann dazu führen, dass die Referenzpunktfahrt in die falsche Richtung gestartet wird.

Achten Sie darauf, dass die Endschalter jeweils bereits den korrekten Status liefern, bevor Sie eine Fahrt starten.

Statusbit SYNC

Das Statusbit (Seite 2766) SYNC meldet Ihnen die Synchronisation der Achse, d. h. nach der korrekten Referenzpunktfahrt ist dieses Statusbit gesetzt und während der Fahrt gelöscht.

Das Statusbit SYNC ist gelöscht

- nach der Parametrierung Ihrer ET 200S-Station
- nach dem Löschen der Impulsfreigabe
- nach einem CPU/Master-STOP

In diesen Fällen ist es sinnvoll, eine Referenzpunktfahrt durchzuführen.

Statusbits POS und POS_RCD

Solange die Referenzpunktfahrt läuft, wird dies durch das gesetzte Rückmeldebit (Seite 2766) POS angezeigt.

Nach korrekter Referenzpunktfahrt wird das Erreichen der Position durch das gesetzte Rückmeldebit POS_RCD angezeigt.

Falls die Referenzpunktfahrt abgebrochen wird, bleibt das Rückmeldebit POS_RCD zurückgesetzt.

Restweg, Position, Frequenz

Der zurückgemeldete Wert (Seite 2766) ist während der Referenzpunktfahrt irrelevant.

Hinweis

Damit das 1STEP 5V den Referenzpunkt wiederholt genau anfahren kann, muss die Periodendauer der Start-Stop-Frequenz größer sein als die Laufzeit eines einzelnen Schrittes vom 1STEP 5V zum Schrittmotor und über den Referenznocken zurück zum 1STEP 5V (Eingangsverzögerung der Digitaleingänge: 4 ms).

Beim Anhalten am Referenznocken oder an einem der Endschalter während der Beschleunigungsphase gibt das 1STEP 5V noch maximal 50 ms lang Impulse mit der bereits erreichten Frequenz aus, bevor es den Bremsvorgang einleitet. Dadurch werden abrupte Frequenzänderungen vermieden, die zu Schrittverlusten führen können.

Ablauf der Referenzpunktfahrt

Abschnitte der Referenzpunktfahrt

Eine Referenzpunktfahrt besteht aus maximal drei Abschnitten.

Im **ersten Abschnitt** (1) und **zweiten Abschnitt** (2) wird sichergestellt, dass der Referenznocken gefunden wird.

Diese beiden Abschnitte werden mit der vorgegebenen Ausgangsfrequenz F_a durchfahren.

Im **dritten Abschnitt** (3) wird der Referenznocken mit Start-Stop-Frequenz F_{ss} in der gewählten Richtung bis zum Referenzpunkt ⊕ reproduzierbar angefahren.

Hinweis

Die Anzahl der ausgegebenen Impulse eines Abschnitts ist maximal die parametrierte Länge des Verfahrbereichs minus 1.

Verschiedene Abläufe

Je nach Standort (P) beim Start der Referenzpunktfahrt ergeben sich unterschiedliche Ablaufmuster der Fahrt (REF ist der Referenznocken, verdrahtet am Digitaleingang DI1). Die Darstellung gilt für die Startrichtung vorwärts (DIR_P).

- **Start vor dem REF oder auf Endschalter LIMIT_M**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start vor dem REF.

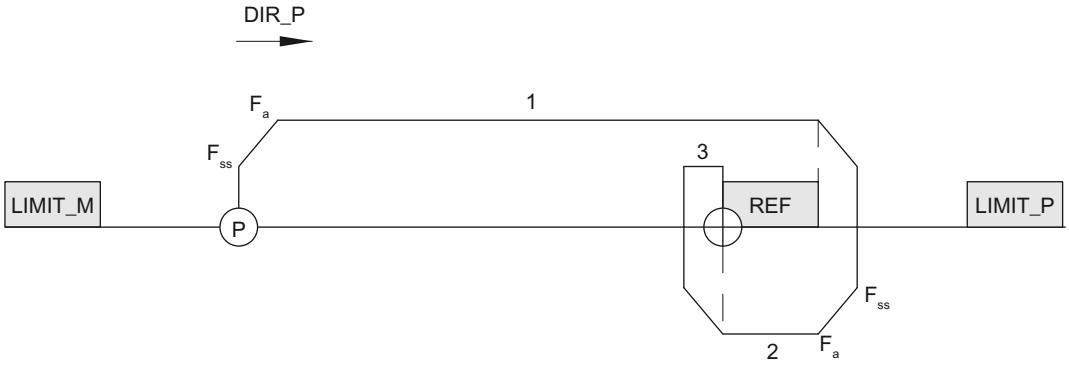

- **Start nach dem REF**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start nach dem REF.

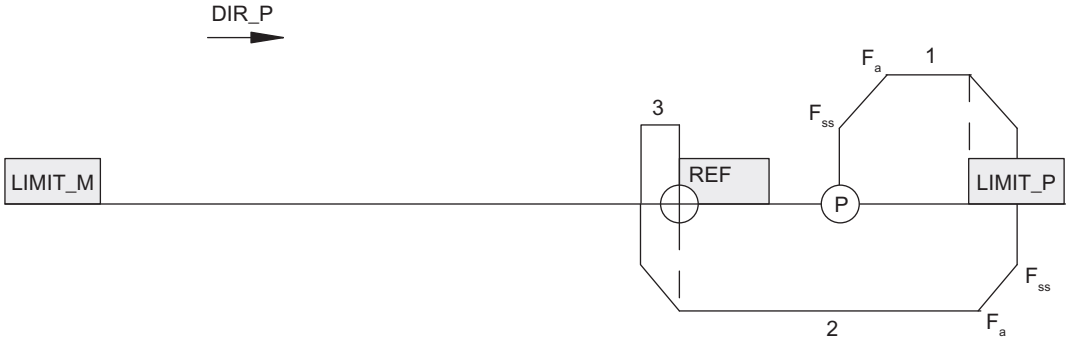

- **Start auf dem REF**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start auf dem REF.

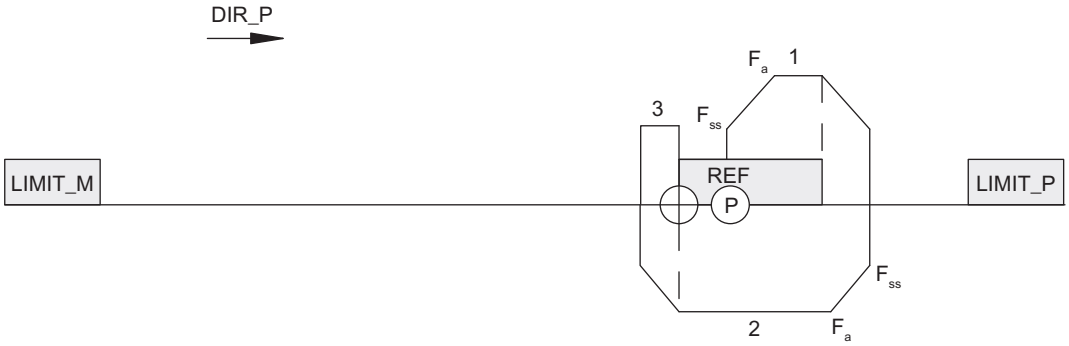

- **Start auf dem Endschalter in Startrichtung**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start auf dem Endschalter in Startrichtung.

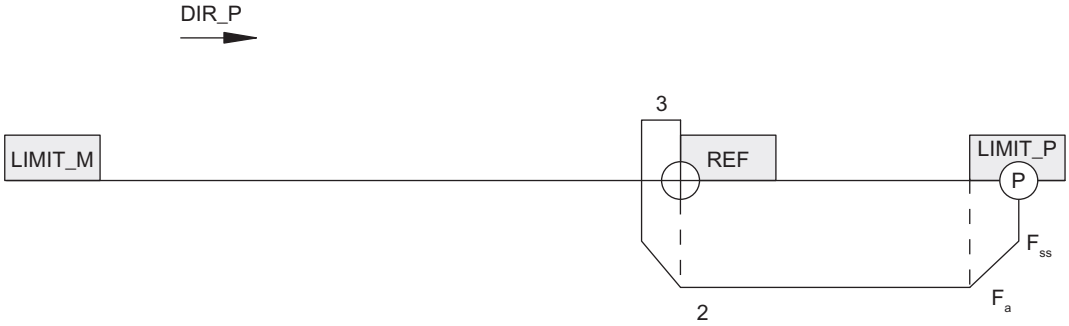

Referenzschalter ist gleichzeitig Endschalter

Hinweis

Wenn der Referenzschalter REF gleichzeitig ein Endschalter ist, wählt das 1STEP 5V die Startrichtung selbständig in Richtung des Endschalters, unabhängig von der Richtungsvorgabe im Fahrauftrag.

- **Start innerhalb des Verfahrbereichs**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start innerhalb des Verfahrbereichs, wobei REF gleichzeitig Endschalter LIMIT_M ist.

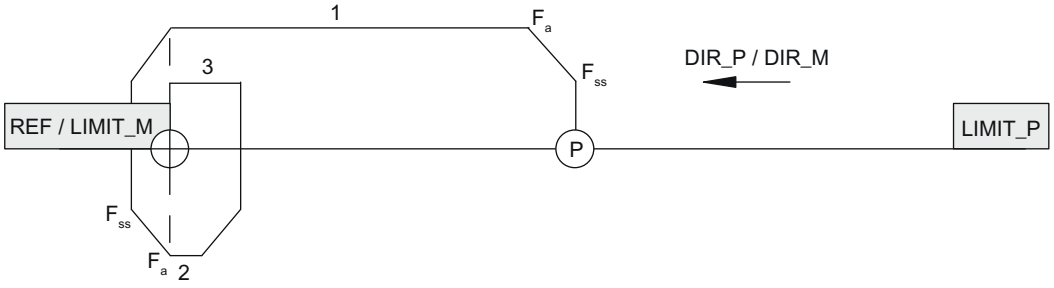

- **Start auf dem Endschalter**

Das folgende Bild zeigt eine Referenzpunktfahrt mit Start auf dem Endschalter LIMIT_P, wobei REF gleichzeitig LIMIT_P ist.

Verhalten bei defektem Referenznocken mit Endschalter (Abbruch der Fahrt)

Das folgende Bild zeigt eine Referenzpunktfahrt mit defektem Referenznocken und Start vor dem REF.

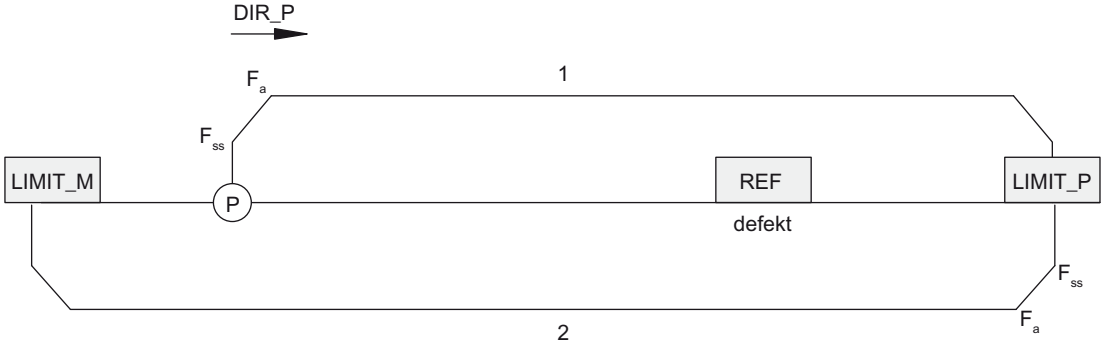

Das folgende Bild zeigt eine Referenzpunktfahrt mit defektem Referenznocken und Start auf LIMIT_P.

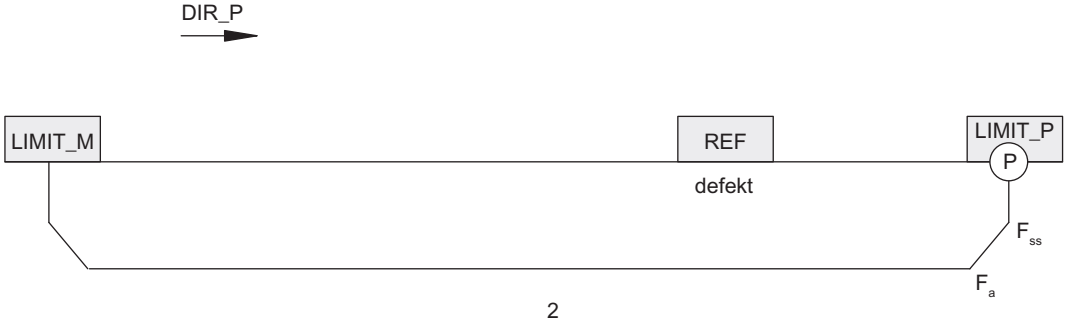

Verhalten bei ständig gesetztem Referenznocken ohne Endschalter

Am Ende des ersten Abschnitts nach Ausgabe von 16777215 Impulsen wird die Fahrt mit gelöschten Statusbits SYNC und POS_RCD beendet.

Verhalten bei Ausfall des Referenznockens ohne Endschalter

Alle drei Abschnitte der Fahrt werden mit der Ausgabe von jeweils 16777215 Impulsen durchgeführt. Danach wird die Fahrt mit gelöschten Statusbits SYNC und POS_RCD abgebrochen.

Referenzpunkt setzen (ab 6ES7138-4DC01-0AB0)

Funktionsbeschreibung

Der Referenzpunkt markiert den Bezugspunkt Ihres Antriebssystems, auf den sich nachfolgende absolute Schrittmaßfahrten und der Positionswert in der Rückmeldeschnittstelle beziehen. Sie legen den Referenzpunkt fest, indem Sie für die aktuelle Position des Schrittmotors den absoluten Positionswert vorgeben.

Auftrag zum Setzen des Referenzpunkts

Ein Auftrag zum Setzen des Referenzpunkts ist ein virtueller Auftrag ohne Fahrbewegung. Er enthält folgende Informationen:

- Position des Referenzpunkts
- Betriebsart = 4 für Referenzpunkt setzen
- Beliebige Richtungsvorgabe als Start (Seite 2762)

Hinweis

Das 1STEP 5V überprüft die vorgegebene Position auf Grenzwerte (minimal 0 und maximal 16777215). Der Endwert kann parametriert werden.

Hinweis

Da Sie den Referenzpunkt innerhalb des Verfahrbereichs auf Werte von 0 bis 16777215 setzen können, sind Situationen möglich, in denen ein Endschalter nicht mehr erreichbar ist.

Beispiele:

- Wenn Sie den Referenzpunkt innerhalb des Fahrbereichs auf "0" setzen, ist der Endschalter in Richtung rückwärts nicht mehr über einen Fahrauftrag erreichbar, da Sie keine negativen Positionen anfahren können.
  Achten Sie darauf, dass alle anzufahrenden Positionen im positiven Wertebereich liegen.
- Die Achse kann im nicht synchronisierten Zustand keine Strecke fahren, die größer ist als der parametrierte Verfahrbereich. Wenn der parametrierte Verfahrbereich kleiner ist als die Strecke zwischen beiden Endschaltern und die Achse auf dem Endschalter in Richtung vorwärts steht, kann die Achse mit einer Relativfahrt nicht bis zum Endschalter in Richtung rückwärts fahren.

Rückmeldungen

Nach korrekter Ausführung des Auftrags sind die Rückmeldebits (Seite 2766) SYNC und POS_RCD gesetzt.

Wenn das Rückmeldebit SYNC vor Ausführung des Auftrags bereits gesetzt ist, bleibt es gesetzt. Die korrekte Ausführung erkennen Sie auch an der Änderung des Positionswerts in der Rückmeldeschnittstelle.

Schrittmaßfahrt relativ (relativ Positionieren)

Funktionsbeschreibung

Mit der relativen Schrittmaßfahrt können Sie den Schrittmotor um eine vorgegebene Strecke weiterbewegen und somit eine bestimmte Position anfahren.

Die Richtung der Fahrt und die Geschwindigkeit legen Sie beim Start fest.

Fahrauftrag für Schrittmaßfahrt relativ

Der Fahrauftrag enthält folgende Informationen:

- Wegstrecke (Anzahl der auszuführenden Impulse)
- Multiplikator G für die Geschwindigkeit / Ausgangsfrequenz F_a
- Reduzierfaktor R für die parametrierte Basisfrequenz F_b
- Betriebsart = 0 für Schrittmaßfahrt relativ
- Anhalten am Referenznocken (Seite 2749)
- Richtungsvorgabe als Start (Seite 2762)

Hinweis

Das 1STEP 5V überprüft die vorgegebene Wegstrecke auf Grenzwerte (minimal 1 Impuls und maximal 16777215 Impulse). Der Abstand zum Endschalter wird vom 1STEP 5V nicht überprüft. Spätestens beim Erreichen des Endschalters wird die Fahrt angehalten.

Rückmeldungen

Am Beginn der Schrittmaßfahrt wird das Rückmeldebit (Seite 2766) POS_RCD zurückgesetzt.

Solange die Schrittmaßfahrt läuft, wird dies durch das gesetzte Rückmeldebit POS angezeigt.

Nach korrekt verlaufener Schrittmaßfahrt wird das Erreichen der Position durch das gesetzte Rückmeldebit POS_RCD angezeigt.

Falls die Schrittmaßfahrt abgebrochen wird, bleibt das Rückmeldebit POS_RCD zurückgesetzt. Nach dem Anhalten der Schrittmaßfahrt wird der noch zu verfahrende Weg angezeigt, wenn der Rückmeldewert (Seite 2766) auf "Restweg" eingestellt ist.

Schrittmaßfahrt absolut (absolut Positionieren) (ab 6ES7138-4DC01-0AB0)

Funktionsbeschreibung

Mit der absoluten Schrittmaßfahrt können Sie den Schrittmotor an eine vorgegebene Position bewegen und somit eine bestimmte Position anfahren.

Die Geschwindigkeit legen Sie beim Start fest. Die Richtung und die Wegstrecke der Fahrt bestimmt das 1STEP 5V selbständig anhand der Ausgangsposition (Positions-Istwert). Für eine Moduloachse können Sie die Richtung auch vorgeben.

Hinweis

Wenn Sie bei einer Moduloachse Start vorwärts und Start rückwärts (DIR_P und DIR_M) gleichzeitig setzen, dann wählt das 1STEP 5V selbständig den kürzesten Weg zum Erreichen (Seite 2751) der Zielposition.

Fahrauftrag für Schrittmaßfahrt absolut

Der Fahrauftrag enthält folgende Informationen:

- Zielposition
- Multiplikator G für die Geschwindigkeit / Ausgangsfrequenz F_a
- Reduzierfaktor R für die parametrierte Basisfrequenz F_b
- Betriebsart = 2 für Schrittmaßfahrt absolut
- Beliebige Richtungsvorgabe als Start (Seite 2762)

Hinweis

Das 1STEP 5V überprüft die vorgegebene Position auf Grenzwerte (minimal 0 und maximal 16777215). Der Endwert kann parametriert werden.

Der Fahrauftrag wird nur dann ausgeführt, wenn Sie zuvor die Position des Referenzpunkts ermittelt oder festgelegt haben (das SYNC-Bit muss gesetzt sein, siehe "Referenzpunktfahrt (Seite 2740)" bzw. "Referenzpunkt setzen (Seite 2745)").

Das Steuersignal (Seite 2762) "Halt am Referenznocken" wird nicht berücksichtigt.

Rückmeldungen

Am Beginn der Schrittmaßfahrt wird das Rückmeldebit (Seite 2766) POS_RCD zurückgesetzt.

Solange die Schrittmaßfahrt läuft, wird dies durch das gesetzte Rückmeldebit POS angezeigt.

Nach korrekt verlaufener Schrittmaßfahrt wird das Erreichen der Position durch das gesetzte Rückmeldebit POS_RCD angezeigt.

Falls die Schrittmaßfahrt abgebrochen wird, bleibt das Rückmeldebit POS_RCD zurückgesetzt. Nach dem Anhalten der Schrittmaßfahrt wird der noch zu verfahrende Weg angezeigt, wenn der Rückmeldewert (Seite 2766) auf "Restweg" eingestellt ist.

Drehzahlbetrieb (ab 6ES7138-4DC01-0AB0)

Funktionsbeschreibung

In dieser Betriebsart geben Sie die Frequenz vor, mit der die Ausgabe der Impulse (Schritte) erfolgen soll. Wenn Sie die Frequenz ändern, werden die Impulse nach einer Beschleunigungs- oder Verzögerungsphase mit der neuen Frequenz ausgegeben. Die Ausgabe erfolgt solange endlos, bis Sie den Fahrauftrag anhalten oder bei einer Linearachse eine Verfahrbereichsgrenze erreicht wird.

Das folgende Bild zeigt den Drehzahlbetrieb mit einer Moduloachse.

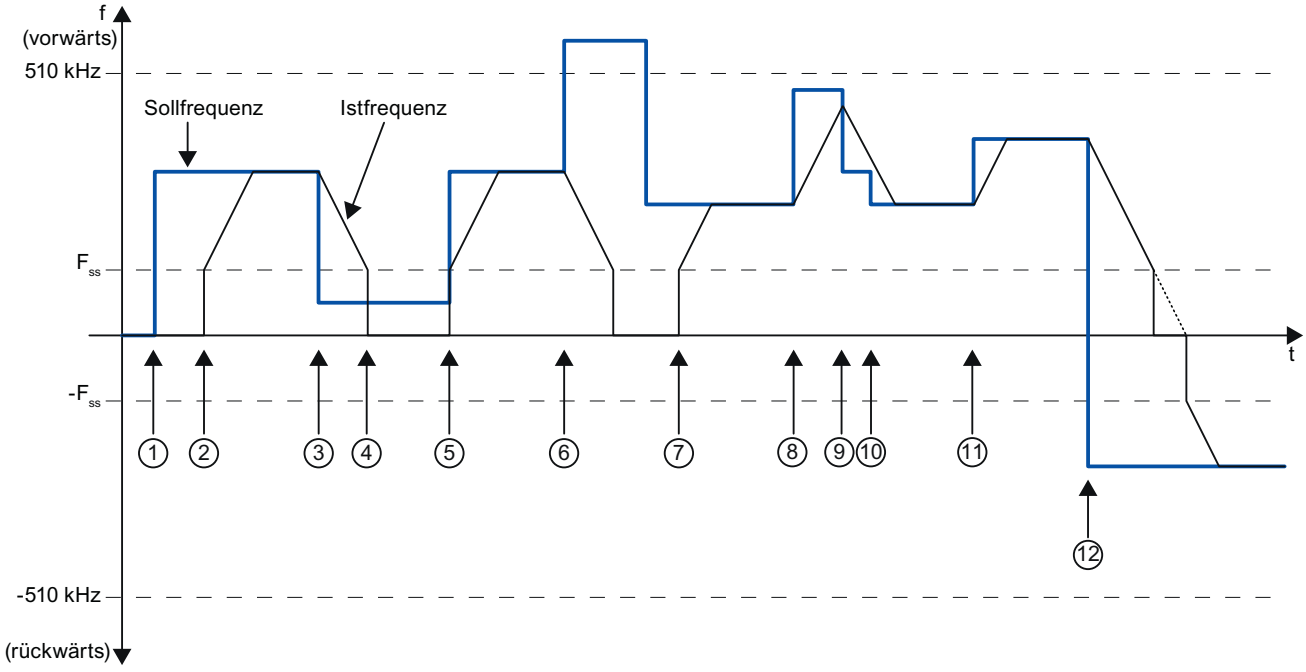

① Vorgabe Sollfrequenz
② Start des Drehzahlbetriebs mit Frequenz F_{SS}, Beschleunigungsrampe bis zur Sollfrequenz
③ Neue Sollfrequenz (< F_{SS})
④ Abbremsrampe bis F_{SS}, anschließend Stopp der Impulsausgabe (kein Stopp des Drehzahlbetriebs)
⑤ Vorgabe Sollfrequenz: Fortführung des Drehzahlbetriebs mit F_{SS}, Beschleunigungsrampe bis zur Sollfrequenz
⑥ Vorgabe ungültige Sollfrequenz: Stopp des Drehzahlbetriebs
⑦ Start des Drehzahlbetriebs mit Frequenz F_{SS}, Beschleunigungsrampe bis zur Sollfrequenz
⑧ Vorgabe Sollfrequenz
⑨ Vorgabe Sollfrequenz: Wechsel von Beschleunigungsrampe zu Abbremsrampe
⑩ Vorgabe Sollfrequenz: Fortführung der Abbremsrampe
⑪ Vorgabe Sollfrequenz
⑫ Vorgabe negative Sollfrequenz: Richtungswechsel

Fahrauftrag für Drehzahlbetrieb

Der Fahrauftrag enthält folgende Informationen:

- Sollfrequenz als 32 Bit-Wert (STEP 7-Datentyp REAL)
- Richtungsvorgabe durch das Vorzeichen der Sollfrequenz (positiv: vorwärts)
- Betriebsart = 3 für Drehzahlbetrieb
- Beliebige Richtungsvorgabe als Start (Seite 2762)

Hinweis

Das 1STEP 5V überprüft die vorgegebene Frequenz auf Grenzwerte (minimal -510,0 kHz und maximal +510,0 kHz).

Die vorgegebene Frequenz wird unter Berücksichtigung der Start-Stop-Frequenz F_{ss} mit der parametrierten Beschleunigung a angefahren. Bei Frequenzen, die betragsmäßig kleiner als F_{ss} sind, erfolgt keine Impulsausgabe.

Die endlose Ausgabe der Frequenz wird bei folgenden Ereignissen beendet:

- Erreichen der Grenzen des parametrierten Verfahrbereichs (0 in Richtung rückwärts), wenn keine Moduloachse parametriert ist
- Sonstige Abbruch-Bedingungen für Fahraufträge (Seite 2749)

Rückmeldungen

Solange der Fahrauftrag läuft, wird dies durch das gesetzte Rückmeldebit (Seite 2766) POS angezeigt.

Wenn eine neue Frequenz vorgegeben wird, dann wird das Rückmeldebit POS_RCD gelöscht. Wenn nach der Beschleunigungs- oder Verzögerungsphase die neue Frequenz erreicht ist, dann wird POS_RCD wieder gesetzt.

Die aktuelle Frequenz wird in der Rückmeldeschnittstelle als 32 Bit-Wert (STEP 7-Datentyp REAL) angezeigt, wenn der Rückmeldewert (Seite 2766) auf "Frequenz" eingestellt ist.

Hinweis

Wenn Sie während eines laufenden Fahrauftrags mit einer Frequenz von 0,0 Hz eine Frequenz ungleich 0,0 Hz und ein Anhalten des Fahrauftrags über das Steuerbit STOP anfordern, wird der Fahrauftrag angehalten. Der Zustand des Rückmeldebits POS_RCD ist dann undefiniert.

Fahrauftrag anhalten

Gezieltes Anhalten des Fahrauftrags

- verursacht durch	angezeigt durch Rückmeldebit
STOP durch Steuerbit	-
externer STOP am Digitaleingang	STOP_EXT

- verursacht durch	angezeigt durch Rückmeldebit
Endschalter in Richtung vorwärts erreicht (LIMIT_P oder Digitaleingang)	STOP_LIMIT_P
Endschalter in Richtung rückwärts erreicht (LIMIT_M oder Digitaleingang)	STOP_LIMIT_M
STOP am Referenznocken	STOP_REF

Hinweis

Beachten Sie, dass die Endschalter in der Betriebsart Referenzpunktfahrt auch zur Suche des Referenznocken verwendet werden.

Wenn ein Fahrauftrag an LIMIT_P angehalten wurde, dann können Sie die Achse mit einem neuen Fahrauftrag mit DIR_M vom Endschalter wegfahren. Bei Anhalten an LIMIT_M können Sie die Achse mit DIR_P vom Endschalter wegfahren.

Halt am Referenznocken

Wenn beim Start einer Fahrt die Funktion "Halt am Referenznocken" angewählt ist (das Steuerbit STOP_REF_EN ist gesetzt) und während der Fahrt der Referenznocken erkannt wird, wird angehalten und die Fahrt ist beendet.

Anhalten des Fahrauftrags im Ausnahmefall

In folgenden Fällen wird der Fahrauftrag mit Verlust der Synchronisation angehalten:

- Fehlbedienung der Steuerschnittstelle während des laufenden Fahrauftrags
- externer Fehler ERR_24V durch Überlastung der Geberversorgung (z. B. Kurzschluss)
- CPU/Master-STOP
- Bei Linearachse: Erreichen der Grenzen des Verfahrbereichs

Auswirkungen

Tritt einer der genannten Gründe zum Anhalten der laufenden Positionierung auf, wird diese durch eine Verzögerungsrampe beendet.

Auch beim Anhalten des Fahrauftrags im Ausnahmefall wird der Rückgabewert weiter aktualisiert. Dadurch können Sie einen nach dem Anhalten noch anstehenden Restweg durch einen neuen Fahrauftrag in der Betriebsart "Schrittmaßfahrt relativ" verfahren.

Endschalter und externer STOP

Mittels Parametrierung legen Sie fest, ob Sie für den externen STOP und die Endschalter Öffner oder Schließer verdrahten.

- **Öffner bedeutet:** Der externe STOP und die Wirkung der Endschalter werden durch ein 0-Signal ausgelöst. Beim Erreichen der Endschalter löschen Sie das zugehörige Steuerbit.
- **Schließer bedeutet:** Der externe STOP und die Wirkung der Endschalter werden durch ein 1-Signal ausgelöst. Beim Erreichen der Endschalter setzen Sie das zugehörige Steuerbit.

Hinweis

Beim Anhalten während der Beschleunigungsphase gibt das 1STEP 5V noch maximal 50 ms lang Impulse mit der bereits erreichten Frequenz aus, bevor es den Bremsvorgang einleitet. Dadurch werden abrupte Frequenzänderungen vermieden, die zu Schrittverlusten führen könnten.

Achstyp und Verfahrbereich (ab 6ES7138-4DC01-0AB0)

Übersicht

Bei der Parametrierung legen Sie fest, welchen Achstyp der vom 1STEP 5V gesteuerte Schrittmotor ansteuert. Sie können wählen zwischen den folgenden Achstypen:

- Linearachse
- Moduloachse

Funktionsbeschreibung

Linearachse

Der Verfahrbereich einer Linearachse kann eingestellt werden. Die Untergrenze ist immer 0, die Obergrenze wird parametriert und hat einen Wertebereich von 1 bis 16777215.
Der Verfahrbereich kann durch Endschalter weiter eingeschränkt werden (Arbeitsbereich).

Das folgende Bild zeigt eine Linearachse.

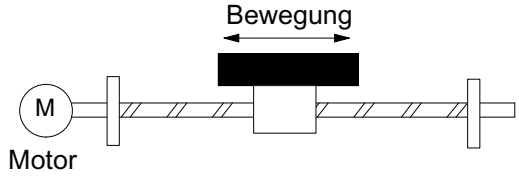

Moduloachse

Eine Moduloachse ist eine Ausprägung der Rundachse. Das folgende Bild zeigt eine Moduloachse.

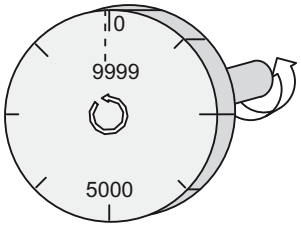

Ende der Moduloachse

Mit dem Parameter "Verfahrbereich" legen Sie das Ende der Moduloachse fest.

Der Positions-Istwert kann den Verfahrbereichswert nicht erreichen, weil dieser höchste Wert physikalisch auf der gleichen Position liegt wie der Anfang der Moduloachse (0).

Beispiel:

Sie geben als Verfahrbereich den Wert 10000 vor, siehe obiges Bild.

Bei Vorwärtsbewegung springt der Positionswert in der Rückmeldeschnittstelle von 9999 auf 0, bei Rückwärtsbewegung von 0 auf 9999.

Referenzpunktfahrt

Wenn Sie bei der Parametrierung die Moduloachse gewählt und Ihr Antriebssystem mit einem Referenznocken versehen haben, dann können Sie eine Referenzpunktfahrt (Seite 2740) durchführen.

Die Fahrt wird erfolglos abgebrochen, wenn nach Ausgabe einer Impulsanzahl, die dem parametrierten Verfahrbereich entspricht, der Referenznocken nicht gefunden wurde. Dann bleiben die Statusbits (Seite 2766) SYNC und POS_RCD gelöscht.

Referenzpunkt setzen

Sie dürfen für die Position des Referenzpunkts nur Werte von 0 bis parametriertes Ende des Verfahrbereichs – 1 angeben.

Positionieren relativ

Das Ende des Verfahrbereichs (Ende der Moduloachse) darf in beiden Richtungen überfahren werden.

Positionieren absolut

Wenn Sie bei der Parametrierung die Moduloachse gewählt haben, dann dürfen Sie für die Zielposition nur Werte von 0 bis parametriertes Ende des Verfahrbereichs – 1 angeben.

Im Unterschied zur Linearachse geben Sie über die Richtungsvorgabe (Seite 2762) beim Start des Fahrauftrags an, auf welchem Weg die Zielposition erreicht werden soll:

- Start rückwärts (DIR_M): Das 1STEP 5V fährt die Zielposition in Richtung niedrigerer Positions-Istwerte an (Möglichkeit 1 im folgenden Bild).
- Start vorwärts (DIR_P): Das 1STEP 5V fährt die Zielposition in Richtung höherer Positions-Istwerte an (Möglichkeit 2 im folgenden Bild).
- Start vorwärts und Start rückwärts gleichzeitig (DIR_P und DIR_M): Das 1STEP 5V wählt selbständig den kürzesten Weg zum Erreichen der Zielposition (Möglichkeit 1 im folgenden Bild).

Das folgende Bild zeigt Positionieren absolut mit einer Moduloachse.

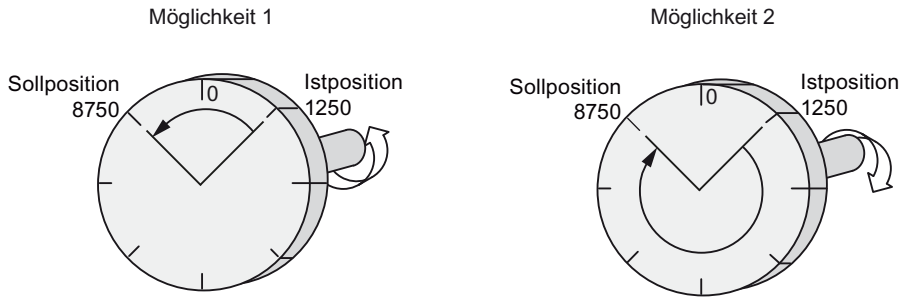

Impulsfreigabe

Funktionsbeschreibung

Die Impulsfreigabe gibt die Ausgabe der Impulse vom 1STEP 5V an das Leistungsteil frei. Ohne Impulsfreigabe ist keine Fahrt möglich.

Impulsfreigabe erteilen

Sie erteilen die Impulsfreigabe entweder

- über den Digitaleingang DI0, wenn "Funktion DI0" als externe Impulsfreigabe (Seite 2755) parametriert ist

oder

- über das Steuerbit DRV_EN, wenn "Funktion DI0" als externer STOP (Seite 2755) bzw. Endschalter vorwärts oder rückwärts (Seite 2755) parametriert ist

Sie erkennen die erteilte Impulsfreigabe daran, dass

- bei korrekter Parametrierung die LED (Seite 2769) RDY am 1STEP 5V leuchtet.
- das Rückmeldebit (Seite 2766) STS_DRV_EN gesetzt ist.

Löschen der Impulsfreigabe

Ein Löschen der Impulsfreigabe während der Fahrt beendet diese sofort, da keine Impulse mehr zum Leistungsteil ausgegeben werden. Restweg und Positions-Istwert sind nicht mehr gültig. Die Synchronisation der Achse über den Referenzpunkt geht verloren. Das Rückmeldebit SYNC und die LED RDY sind gelöscht.

Ein Löschen der Impulsfreigabe im Stillstand führt zum Löschen des Rückmeldebits SYNC und der LED RDY.

In diesem Fall ist es eventuell erforderlich, eine Referenzpunktfahrt (Seite 2740) durchzuführen.

Parameter ändern im Betrieb

Einleitung

Sie können einige Parameter des 1STEP 5V im Betrieb ändern, ohne dass eine Neuparametrierung der gesamten ET 200S-Station erforderlich ist.

Veränderbare Parameter

Folgende Parameter sind veränderbar:

- Basisfrequenz F_b
- Multiplikator n für Start-Stop-Frequenz F_{ss}
- Multiplikator i für Beschleunigung / Verzögerung
- Rückmeldewert in der Rückmeldeschnittstelle

Beim Anstoß der Parameteränderung über das Steuerbit (Seite 2762) C_PAR werden die Parameter auf zulässige Werte geprüft. Haben Sie unzulässige Werte verwendet, wird das Rückmeldebit (Seite 2766) ERR_JOB gesetzt.

Durch den Parametrierauftrag werden nur die Rückmeldebits für die Auftragsabwicklung ERR_JOB und STS_JOB beeinflusst.

Durchführung einer Parameteränderung

Das folgende Bild zeigt den Ablauf bei einer Parameteränderung.

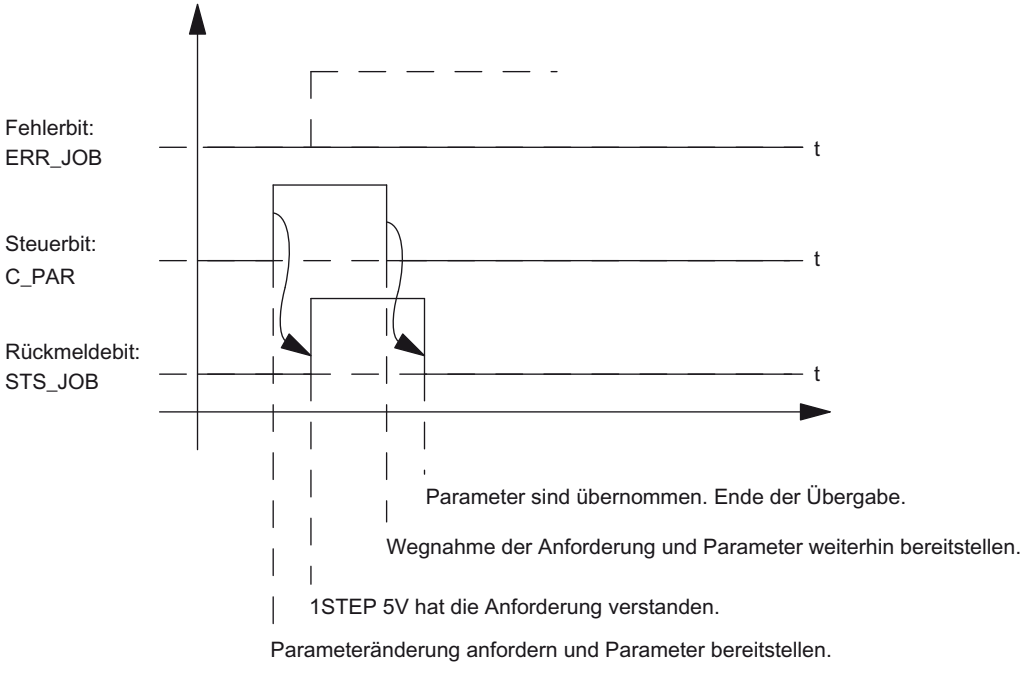

Hinweis

Es darf immer nur eines der folgenden Steuerbits gesetzt sein:

DIR_P oder DIR_M oder C_PAR.

Ansonsten wird der Fehler ERR_JOB gemeldet. Die Fehlermeldung des Auftrags wird durch den Start des nächsten Auftrags gelöscht.

Verhalten der Digitaleingänge

Einleitung

Sie können die Funktion und das Verhalten (aktive Pegel) der Digitaleingänge DI0 (3) und DI1 (7) parametrieren. Diese Parameter können nicht über das Anwenderprogramm geändert werden.

Digitaleingang DI0 (3)

Die Funktion des Digitaleingangs DI0 (3) können Sie parametrieren als:

- externe Impulsfreigabe
- externer STOP
- Endschalter in Richtung vorwärts
- Endschalter in Richtung rückwärts

Außerdem können Sie das Verhalten des Digitaleingangs DI0 (3) parametrieren als:

- Öffner
- Schließer

Digitaleingang DI0 (3) als externe Impulsfreigabe

Der Eingang muss im Betrieb gesetzt (Seite 2753) (aktiviert) sein. Ist der Eingang gesetzt und ist die Parametrierung fehlerfrei, so ist das 1STEP 5V betriebsbereit.

Digitaleingang DI0 (3) als externer STOP

Mit dieser Eingangsfunktion können Sie einen laufenden Fahrauftrag durch ein externes Signal anhalten (Seite 2749).

Digitaleingang DI0 (3) als Endschalter in Richtung vorwärts / rückwärts

Mit diesen Eingangsfunktionen begrenzen Sie durch ein externes Signal den Verfahrbereich in Vorwärts- oder in Rückwärtsrichtung. Das Signal hat dieselbe Wirkung wie eines der beiden Steuerbits LIMIT_P bzw. LIMIT_M in der Steuerschnittstelle (Seite 2762).

Digitaleingang DI1 (7)

Die Funktion des Digitaleingangs DI1 (7) können Sie parametrieren als:

- Referenzschalter: Referenznocken
- Referenz- und Endschalter in Richtung vorwärts
  Diese Parameterauswahl ist nur möglich, wenn "Funktion DI0" nicht als "Endschalter in Richtung vorwärts" parametriert ist.
- Referenz- und Endschalter in Richtung rückwärts
  Diese Parameterauswahl ist nur möglich, wenn "Funktion DI0" nicht als "Endschalter in Richtung rückwärts" parametriert ist.

Außerdem können Sie das Verhalten des Digitaleingangs DI1 (7) parametrieren als:

- Öffner
- Schließer

Digitaleingang DI1 (7) als Referenzschalter

An diesen Eingang verdrahten Sie einen Schalter für den Referenznocken.

Sie benötigen einen Referenznocken

- für eine Referenzpunktfahrt
- für eine Schrittmaßfahrt mit Halt am Referenznocken

Digitaleingang DI1 (7) als Referenzschalter und Endschalter in Richtung vorwärts / rückwärts

Mit diesen Eingangsfunktionen begrenzen Sie durch den Referenznocken zusätzlich den Verfahrbereich in Vorwärts- oder in Rückwärtsrichtung. Das Signal hat zusätzlich dieselbe Wirkung wie eines der beiden Steuerbits LIMIT_P bzw. LIMIT_M in der Steuerschnittstelle (Seite 2762).

Wenn Sie das Verhalten des Digitaleingangs DI1 (7) als "Referenz- und Endschalter" parametriert haben, dann wählt das 1STEP 5V die Startrichtung selbständig in Richtung Endschalter, unabhängig von der Richtungsvorgabe im Fahrauftrag.

Verhalten bei CPU/Master-STOP

Einleitung

Das 1STEP 5V erkennt den CPU/Master-STOP. Es reagiert darauf, indem es den laufenden Fahrauftrag anhält (Seite 2749).

Verlassen des Zustands CPU/Master-STOP

ET 200S-Station	1STEP 5V
Ohne Neuparametrierung der ET 200S-Station	• Die Rückmeldeschnittstelle des 1STEP 5V bleibt aktuell. • Die über Parametrierauftrag geänderten Werte bleiben erhalten. • War zum Zeitpunkt des CPU/Master-STOP ein Steuerbit gesetzt (DIR_P, DIR_M, C_PAR), so sind beim Verlassen des Zustands CPU/Master-STOP die Bits STS_JOB und ERR_JOB gesetzt. Löschen Sie das Steuerbit. Die Fahrt / der Parametrierauftrag wird nicht durchgeführt. Danach können Sie eine neue Fahrt über das Steuerbit anstoßen. • Nach der Verzögerungsrampe werden die Impulsfreigabe, die LED RDY und das Statusbit SYNC gelöscht.
Mit Neuparametrierung der ET 200S-Station	• Die Informationen über frühere Fahrten und Parametrieraufträge werden zurückgesetzt. • War zum Zeitpunkt des CPU/Master-STOP die Impulsfreigabe über das Steuerbit DRV_EN erteilt, werden nach der Verzögerungsrampe die Impulsfreigabe, die LED RDY und das Statusbit SYNC gelöscht.

Neuparametrierung der ET 200S-Station

Eine Neuparametrierung der ET 200S-Station durch Ihre CPU / DP-Master erfolgt bei:

- NETZ-EIN der CPU / DP-Master
- NETZ-EIN der IM 151 / IM 151 FO
- nach Ausfall der DP-Übertragung
- nach Laden einer geänderten Parametrierung bzw. Konfiguration der ET 200S-Station in die CPU / DP-Master
- beim Stecken des 1STEP 5V
- NETZ-EIN oder Stecken des zugehörigen Powermoduls

Siehe auch

Impulsfreigabe (Seite 2753)

Verhalten der Digitaleingänge (-4DC00-)

Digitaleingang DI 3

Die Funktion des Digitaleingangs DI 3 können Sie parametrieren als:

- externe Impulsfreigabe
- externer STOP

Außerdem können Sie das Verhalten des Digitaleingangs DI 3 parametrieren als:

- Öffner
- Schließer

Digitaleingang DI 3 als externe Impulsfreigabe

Der Eingang muss im Betrieb gesetzt (Seite 2758) (aktiviert) sein. Ist der Eingang gesetzt und ist die Parametrierung fehlerfrei, so ist das 1STEP 5V/204kHz betriebsbereit.

Digitaleingang DI 3 als externer STOP

Mit dieser Eingangsfunktion können Sie einen laufenden Fahrauftrag durch ein externes Signal anhalten.

Digitaleingang REF

An diesen Eingang verdrahten Sie einen Schalter für den Referenznocken.

Sie benötigen einen Referenznocken

- für eine Referenzpunktfahrt
- für eine Schrittmaßfahrt mit Halt am Referenznocken

Impulsfreigabe (-4DC00-)

Funktionsbeschreibung

Die Impulsfreigabe gibt die Ausgabe der Impulse vom 1STEP 5V/204kHz an das Leistungsteil frei. Ohne Impulsfreigabe ist keine Fahrt möglich.

Impulsfreigabe erteilen

Sie erteilen die Impulsfreigabe entweder

- über den Digitaleingang DI 3, wenn "Funktion DI" als externe Impulsfreigabe (Seite 2758) parametriert ist

oder

- über das Steuerbit DRV_EN, wenn "Funktion DI" als externer STOP (Seite 2758) parametriert ist

Sie erkennen die erteilte Impulsfreigabe daran, dass

- bei korrekter Parametrierung die LED (Seite 2769) RDY am 1STEP 5V/204kHz leuchtet.
- das Rückmeldebit STS_DRV_EN gesetzt ist.

Löschen der Impulsfreigabe

Ein Löschen der Impulsfreigabe während der Fahrt beendet diese sofort, da keine Impulse mehr zum Leistungsteil ausgegeben werden. Der Schrittmotor läuft ohne Impulsansteuerung aus. Dadurch ist der Restweg nicht mehr gültig. Die Synchronisation der Achse über den Referenzpunkt geht verloren. Das Rückmeldebit SYNC und die LED RDY sind gelöscht.

Ein Löschen der Impulsfreigabe im Stillstand führt zum Löschen des Rückmeldebits SYNC und der LED RDY.

In diesem Fall ist es eventuell erforderlich, eine Referenzpunktfahrt durchzuführen.

6.2.8.4 1STEP 5V konfigurieren und parametrieren

Parameter

Sammeldiagnose aktivieren

Sammeldiagnose

Die Sammeldiagnose aktivieren Sie im entsprechenden Optionskästchen.

Ein "Kurzschluss Geberversorgung" ERR_24V oder ein "Parametrierfehler" ERR_PARA führen bei freigegebener Sammeldiagnose zu einer kanalbezogenen Diagnose.

Basisfrequenz auswählen

Basisfrequenz F_b

Die Basisfrequenz F_b ist der Basiswert für die Einstellung der Start-Stop-Frequenz, der Ausgangsfrequenz und der Beschleunigung / Verzögerung.

Wählen Sie die für Ihre Anwendung geeignete Basisfrequenz aus.

Detaillierte Informationen dazu finden Sie unter Einstellen der Basisfrequenz (Seite 2736).

Multiplikator n festlegen

Multiplikator n

Mit dem Multiplikator n können Sie die Start-Stop-Frequenz gestuft einstellen:

$F_{ss} = F_b \times n$

Zulässiger Wertebereich: 1 … 255

Siehe auch

Einstellen der Basisfrequenz (Seite 2736)

Zeitintervall i festlegen

Zeitintervall i

Mit dem Multiplikator i können Sie die Beschleunigung / Verzögerung gestuft einstellen:

$a = F_b / (i \times 0{,}128\ ms)$

Zulässiger Wertebereich: 1 … 255

Siehe auch

Einstellen der Basisfrequenz (Seite 2736)

Rückmeldewert auswählen

Rückmeldewert

Wählen Sie aus, welcher Rückmeldewert in der Rückmeldeschnittstelle (Seite 2766) zur Verfügung gestellt wird:

- Restweg
- Position
- Frequenz

Funktion DI0 auswählen

Funktion DI0

Wählen Sie aus, welche Funktion der Digitaleingang DI0 (3) (Seite 2755) haben soll:

- externe Impulsfreigabe
- externer STOP
- Endschalter in Richtung vorwärts
- Endschalter in Richtung rückwärts

Funktion DI1 auswählen

Funktion DI1

Wählen Sie aus, welche Funktion der Digitaleingang DI1 (7) (Seite 2755) haben soll:

- Referenzschalter (Referenznocken)
- Referenzschalter und Endschalter in Richtung vorwärts
  Diese Parameterauswahl ist nur möglich, wenn "Funktion DI0" nicht als "Endschalter vorwärts" parametriert ist.
- Referenzschalter und Endschalter in Richtung rückwärts
  Diese Parameterauswahl ist nur möglich, wenn "Funktion DI0" nicht als "Endschalter rückwärts" parametriert ist.

Verhalten der Eingänge DI0 und DI1 auswählen

Eingang DI0, Eingang DI1

Wählen Sie das Verhalten der Digitaleingänge DI0 (3) und DI1 (7) aus:

- Öffner
- Schließer

Siehe auch

Verhalten der Digitaleingänge (Seite 2755)

Fahrauftrag anhalten (Seite 2749)

Verhalten der Endschalter in der Steuerschnittstelle auswählen

Endschalter in der Steuerschnittstelle

Wählen Sie das Verhalten der Endschalter in der Steuerschnittstelle (Seite 2762) aus:

- Öffner
- Schließer

Siehe auch

Fahrauftrag anhalten (Seite 2749)

Moduloachse aktivieren

Moduloachse

Am 1STEP 5V ist der Achstyp "Linearachse" voreingestellt. Wenn Sie eine "Moduloachse" ansteuern wollen, dann müssen Sie das Optionskästchen "Moduloachse" aktivieren.

Siehe auch

Achstyp und Verfahrbereich (ab 6ES7138-4DC01-0AB0) (Seite 2751)

Verfahrbereich festlegen

Verfahrbereich

Mit dem Parameter "Verfahrbereich" legen Sie das Ende der Moduloachse fest.

Zulässiger Wertebereich: 1 … 16777216

Siehe auch

Achstyp und Verfahrbereich (ab 6ES7138-4DC01-0AB0) (Seite 2751)

Steuer- und Rückmeldeschnittstelle des 1STEP 5V

Steuerschnittstelle des 1STEP 5V

Hinweis

Für das 1STEP 5V sind folgende Daten der Steuerschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7

Belegung der Steuerschnittstelle

Die folgende Tabelle zeigt die Belegung der Steuerschnittstelle (Ausgänge) des 1STEP 5V.

Adresse	Belegung	
Byte 0 bis 3	**Schrittmaßfahrt relativ (Byte 4 Bit 2 bis Bit 0 ist 0), Schrittmaßfahrt absolut (Byte 4 Bit 2 bis Bit 0 ist 2)**	
	Byte 0	Multiplikator G: $F_a = F_b \times R \times G$ (Wertebereich 1 … 255)
	Byte 1	Wegstrecke bzw. Position Bit 23 … Bit 16
	Byte 2	Wegstrecke bzw. Position Bit 15 … Bit 8
	Byte 3	Wegstrecke bzw. Position Bit 7 … Bit 0
	Referenzpunktfahrt (Byte 4 Bit 2 bis Bit 0 ist 1)	
	Byte 0	Multiplikator G: $F_a = F_b \times R \times G$ (Wertebereich 1 … 255)
	Byte 1	Position Bit 23 … Bit 16
	Byte 2	Position Bit 15 … Bit 8
	Byte 3	Position Bit 7 … Bit 0
	Referenzpunkt setzen (Byte 4 Bit 2 bis Bit 0 ist 4)	
	Byte 0	Reserve = 0
	Byte 1	Position Bit 23 … Bit 16
	Byte 2	Position Bit 15 … Bit 8
	Byte 3	Position Bit 7 … Bit 0
	Drehzahlbetrieb (Byte 4 Bit 2 bis Bit 0 ist 3)	
	Byte 0 bis 3	Frequenz als STEP 7-Datentyp REAL
	Parametrierauftrag (Byte 5 Bit 6 ist 1)	
	Byte 0	Reserve = 0
	Byte 1	Multiplikator i: $a = F_b \times R / (i \times 0{,}128\ ms)$ (Wertebereich 1 … 255)
	Byte 2	Multiplikator n: $F_{ss} = F_b \times n \times R$ (Wertebereich 1 … 255)
	Byte 3	Basisfrequenz F_b: • 0 = 800 Hz • 1 = 400 Hz • 2 = 200 Hz • 3 = 80 Hz • 4 = 40 Hz • 5 = 20 Hz • 6 = 8 Hz • 7 = 4 Hz • 8 = 2000 Hz

Adresse	Belegung		
Byte 4	**Fahrauftrag**		
	Bit 7	Reduzierfaktor • 0 = Faktor 1,0 (keine Reduzierung) • 1 = Faktor 0,1	R
	Bit 6	Fahrauftrag anhalten	STOP
	Bit 5	Start rückwärts	DIR_M
	Bit 4	Start vorwärts	DIR_P
	Bit 3	Firmware-Version V1.0.0: Reserve = 0 Ab Firmware-Version V1.0.1: Freigabe von Frequenzen unterhalb von F_{SS} im Drehzahlbetrieb	EN_LF
	Bit 2 bis Bit 0	Betriebsart • 0 = Schrittmaßfahrt relativ (relativ Positionieren) • 1 = Referenzpunkfahrt • 2 = Schrittmaßfahrt absolut (absolut Positionieren) • 3 = Drehzahlbetrieb • 4 = Referenzpunkt setzen	MODE
	Parametrierauftrag		
	Bit 7 bis Bit 0	Wert ist nicht relevant und wird ignoriert	
Byte 5	**Fahrauftrag**		
	Bit 7	Diagnosefehlerquittung	EXTF_ACK
	Bit 6	Muss auf 0 gesetzt sein	C_PAR
	Bit 5 bis Bit 4	Wert ist nicht relevant und wird ignoriert	FEEDBACK
	Bit 3	Halt am Referenznocken	STOP_REF_EN
	Bit 2	Impulsfreigabe	DRV_EN
	Bit 1	Endschalter in Richtung vorwärts	LIMIT_P
	Bit 0	Endschalter in Richtung rückwärts	LIMIT_M
	Parametrierauftrag		
	Bit 7	Diagnosefehlerquittung	EXTF_ACK
	Bit 6	Parameter ändern; Muss auf 1 gesetzt sein	C_PAR
	Bit 5 bis Bit 4	Rückmeldewert in der Rückmeldeschnittstelle • 00 = Restweg • 01 = Position • 10 = Frequenz • 11 = reserviert	FEEDBACK
	Bit 3	Wert ist nicht relevant und wird ignoriert	STOP_REF_EN
	Bit 2	Impulsfreigabe	DRV_EN
	Bit 1	Wert ist nicht relevant und wird ignoriert	LIMIT_P
	Bit 0	Wert ist nicht relevant und wird ignoriert	LIMIT_M
Byte 6	Reserve = 0		
Byte 7	Reserve = 0		

Erläuterungen zu den Steuerbits

Bits	Erläuterung
Basisfrequenz F_b	Kodierung für eine gestufte Einstellung der Basisfrequenz: • 0 = 800 Hz • 1 = 400 Hz • 2 = 200 Hz • 3 = 80 Hz • 4 = 40 Hz • 5 = 20 Hz • 6 = 8 Hz • 7 = 4 Hz • 8 = 2000 Hz Diesen Parameter können Sie im Betrieb über einen Parametrierauftrag (Seite 2754) mit dem Steuerbit C_PAR ändern.
Betriebsart	Kodierung für Betriebsart: • 0 = Schrittmaßfahrt relativ (relativ Positionieren) • 1 = Referenzpunktfahrt • 2 = Schrittmaßfahrt absolut (absolut Positionieren) • 3 = Drehzahlbetrieb • 4 = Referenzpunkt setzen
C_PAR	Mit diesem Bit wird eine Parameteränderung angefordert.
DIR_M	Mit diesem Bit wird ein Fahrauftrag in Rückwärtsrichtung angefordert und gestartet.
DIR_P	Mit diesem Bit wird ein Fahrauftrag in Vorwärtsrichtung angefordert und gestartet.
DRV_EN	Wenn Sie den Digitaleingang DI0 (3) nicht als externe Impulsfreigabe verwenden, dann wird dieses Bit als Impulsfreigabe ausgewertet. Wenn keine externe Impulsfreigabe parametriert ist und DRV_EN auf 0 geht, kann die Referenzierung (SYNC) verloren gehen.
EN_LF	Ab Firmware-Version V1.0.1 erlauben Sie mit diesem Bit, im Drehzahlbetrieb auch Frequenzen auszugeben, die betragsmäßig kleiner sind als die parametrierte Start-Stopp-Frequenz. Wenn die vorgegebene Frequenz betragsmäßig unterhalb der Start-Stopp-Frequenz liegt, wird sie ohne Beschleunigungs- bzw. Verzögerungsrampe ausgegeben. Frequenzen oberhalb der Start-Stopp-Frequenz werden mit Beschleunigungs- oder Verzögerungsrampe ausgegeben. Die betragsmäßig kleinste ausgebbare Frequenz ist 0,1 Hz. Das Bit wird nur beim Start der Betriebsart Drehzahlbetrieb ausgewertet. Eine Änderung des Bits während eines laufenden Fahrauftrags Drehzahlbetrieb ist nicht wirksam.
Endschalter LIMIT_M	Dieser Endschalter begrenzt den Verfahrbereich in Rückwärtsrichtung. Sie setzen bzw. löschen dieses Bit in Ihrem Anwenderprogramm.
Endschalter LIMIT_P	Dieser Endschalter begrenzt den Verfahrbereich in Vorwärtsrichtung. Sie setzen bzw. löschen dieses Bit in Ihrem Anwenderprogramm.
EXTF_ACK	Quittungsbit für Diagnosemeldung

Bits	Erläuterung
FEEDBACK	Kodierung für den Rückmeldewert in der Rückmeldeschnittstelle: • 00 = Restweg • 01 = Position • 10 = Frequenz • 11 = reserviert Diesen Parameter können Sie im Betrieb über einen Parametrierauftrag (Seite 2754) mit dem Steuerbit C_PAR ändern.
Frequenz	32 Bit-Wert (STEP 7-Datentyp REAL), der die auszugebende Impulsfrequenz enthält.
Multiplikator G	Faktor für eine gestufte Einstellung der Geschwindigkeit / Ausgangsfrequenz
Multiplikator i	Faktor für eine gestufte Einstellung der Beschleunigung / Verzögerung Diesen Parameter können Sie im Betrieb über einen Parametrierauftrag (Seite 2754) mit dem Steuerbit C_PAR ändern.
Multiplikator n	Faktor für eine gestufte Einstellung der Start-Stop-Frequenz Diesen Parameter können Sie im Betrieb über einen Parametrierauftrag (Seite 2754) mit dem Steuerbit C_PAR ändern.
Position	24 Bit-Wert, der die anzufahrende Zielposition enthält
Reduzierfaktor R	Bei gesetztem Bit wird die Basisfrequenz F_b mit 0,1 multipliziert. Dadurch verkleinern sich in gleichem Maß Ausgangsfrequenz F_a, Start-Stop-Frequenz F_{ss} und Beschleunigung / Verzögerung a.
STOP	Mit diesem Bit können Sie jederzeit einen Fahrauftrag mit einer Verzögerungsrampe anhalten (siehe auch "Fahrauftrag anhalten (Seite 2749)").
STOP_REF_EN	Bei gesetztem Bit ist die Funktion "Halt am Referenznocken" aktiv. Beim Erkennen des Referenznockens wird der Fahrauftrag mit einer Verzögerungsrampe angehalten (siehe auch "Fahrauftrag anhalten (Seite 2749)").
Wegstrecke	24 Bit-Wert, der die Anzahl der zu verfahrenden Impulse enthält (ohne Vorzeichen)

Rückmeldeschnittstelle des 1STEP 5V

Hinweis

Für das 1STEP 5V sind folgende Daten der Rückmeldeschnittstelle zusammengehörende, also konsistente Daten:

- Byte 0 … 3
- Byte 4 … 7

Belegung der Rückmeldeschnittstelle

Die folgende Tabelle zeigt die Belegung der Rückmeldeschnittstelle (Eingänge) des 1STEP 5V.

Adresse	Belegung		
Byte 0 bis 3	Restweg (Bit 23 … Bit 0 von 32 Bit) / Position (Bit 23 … Bit 0 von 32 Bit) / Frequenz (32 Bit, STEP 7-Datentyp REAL)		
Byte 4	Bit 7	Kurzschluss Geberversorgung	ERR_24V
	Bit 6	Reserve = 0	
	Bit 5	Parametrierfehler	ERR_PARA
	Bit 4	Referenzpunkt ermittelt	SYNC
	Bit 3	Reserve = 0	
	Bit 2	Position erreicht	POS_RCD
	Bit 1	Fehler bei Auftragsübergabe	ERR_JOB
	Bit 0	Auftragsübergabe läuft	STS_JOB
Byte 5	Bit 7	Fahrauftrag läuft	POS
	Bit 6	Endschalter vorwärts ist Ursache für Halt	STOP_LIMIT_P
	Bit 5	Endschalter rückwärts ist Ursache für Halt	STOP_LIMIT_M
	Bit 4	externer STOP ist Ursache für Halt	STOP_EXT
	Bit 3	Referenznocken ist Ursache für Halt	STOP_REF
	Bit 2	Status DI0	STS_DI0
	Bit 1	Status DI1	STS_DI1
	Bit 0	Status Impulsfreigabe aktiv	STS_DRV_EN
Byte 6	Fehlernummer bei einem Fehler in der Auftragsübergabe		
Byte 7	Reserve = 0		

Erläuterungen zu den Rückmeldebits

Bits	Erläuterung
Frequenz	Ein 32 Bit-Wert (STEP 7-Datentyp REAL), der die aktuelle Impulsfrequenz enthält. Die Impulsfrequenz kann positive (Richtung vorwärts) und negative (Richtung rückwärts) Werte annehmen.
ERR_JOB	Dieses Bit wird gesetzt, wenn der Auftrag nicht eindeutig oder nicht möglich ist. Die Fehlerursache wird durch die zurückgegebene Fehlernummer näher spezifiziert (siehe nachfolgende Tabelle "Fehlernummer in der Rückmeldeschnittstelle").
ERR_PARA	Bei der Parametrierung der ET 200S-Station sind Parameter fehlerhaft. Die Fehlerursache wird durch die zurückgegebene Fehlernummer näher spezifiziert (siehe nachfolgende Tabelle "Fehlernummer in der Rückmeldeschnittstelle"). Das Parameterfehlerbit wird gelöscht, wenn Sie eine korrekte Parametrierung übertragen.
ERR_24V	Die Geberversorgung wurde überlastet (z. B. durch einen Kurzschluss) und ist nun abgeschaltet. ERR_24V wird zurückgesetzt, wenn es mit dem Steuerbit EXTF_ACK quittiert wurde. Wenn die Überlast beseitigt ist, wird die Geberversorgung wieder eingeschaltet und ERR_24V bleibt gelöscht.
Fehlernummer	Gibt die Fehlerursache an, wenn ERR_JOB oder ERR_PARA gesetzt sind (siehe nachfolgende Tabelle "Fehlernummer in der Rückmeldeschnittstelle").
POS	Fahrt: Dieses Bit ist gesetzt, solange der Fahrauftrag läuft.

Bits	Erläuterung
POS_RCD	Beim Start einer Schrittmaßfahrt oder bei Vorgabe einer neuen Sollfrequenz im Drehzahlbetrieb wird POS_RCD gelöscht. Nach korrekt verlaufener Schrittmaßfahrt oder bei erreichter Sollfrequenz im Drehzahlbetrieb ist POS_RCD gesetzt. Wurde die Fahrt abgebrochen (durch Anhalten des Fahrauftrags (Seite 2749) oder Löschen der Impulsfreigabe (Seite 2753)), dann bleibt POS_RCD gelöscht.
Position	Ein 24 Bit-Wert, der die aktuelle absolute Position enthält (ohne Vorzeichen). Byte 0 der Rückmeldeschnittstelle ist 0.
Restweg	Ein 24 Bit-Wert, der die Anzahl der Impulse enthält, die noch verfahren werden müssen (ohne Vorzeichen). Byte 0 der Rückmeldeschnittstelle ist 0.
STOP_EXT	Ursache für Halt: Dieses Bit ist gesetzt, wenn der Fahrauftrag durch externen STOP angehalten wurde.
STOP_LIMIT_M	Ursache für Halt: Dieses Bit ist gesetzt, wenn der Fahrauftrag durch Erreichen des Endschalters rückwärts angehalten wurde.
STOP_LIMIT_P	Ursache für Halt: Dieses Bit ist gesetzt, wenn der Fahrauftrag durch Erreichen des Endschalters vorwärts angehalten wurde.
STOP_REF	Ursache für Halt: Dieses Bit ist gesetzt, wenn der Fahrauftrag durch Erreichen des Referenznockens angehalten wurde.
STS_DI0	Das Bit zeigt den Status des Digitaleingangs DI0 (3) an.
STS_DI1	Das Bit zeigt den Status des Digitaleingangs DI1 (7) an.
STS_DRV_EN	Dieses Bit ist gesetzt, wenn abhängig von der parametrierten Funktion des Digitaleingangs DI0 entweder • die externe Impulsfreigabe gesetzt ist oder • das Steuerbit DRV_EN für die Impulsfreigabe gesetzt ist.
STS_JOB	Dieses Bit wird als Rückmeldung beim Erkennen einer Auftragsanforderung für einen Fahr- oder Parametrierauftrag gesetzt und dann zurückgesetzt, wenn der Auftrag ausgeführt wurde.
SYNC	Dieses Bit wird nach einer korrekt verlaufenen Referenzpunktfahrt oder nach dem manuellen Festlegen des Referenzpunkts gesetzt. Nach der Parametrierung mit neuen Parametern der ET 200S-Station oder nach dem Löschen der Impulsfreigabe ist das Bit SYNC gelöscht.

Fehlernummer

Wenn in der Rückmeldeschnittstelle ein Fehler bei der Auftragsübergabe (ERR_JOB ist gesetzt) oder ein Fehler in der Grundparametrierung (ERR_PARA ist gesetzt) angezeigt wird, wird über eine Fehlernummer die Fehlerursache genauer spezifiziert.

Die folgende Tabelle zeigt die Fehlernummern in der Rückmeldeschnittstelle.

Fehlernummer	Bedeutung
Allgemeine Fehlerursachen	
0_D	Kein Fehler (dann sind auch ERR_JOB bzw. ERR_PARA nicht gesetzt)
1_D	Kombination der Steuerbits (DIR_P, DIR_M, C_PAR) ist unzulässig
2_D	Anderer Auftrag läuft bereits
Fehlerursachen bei einem Fahrauftrag	
16_D	Start vorwärts (DIR_P) bei Endschalter vorwärts (LIMIT_P) aktiv
17_D	Start rückwärts (DIR_M) bei Endschalter rückwärts (LIMIT_M) aktiv

Fehlernummer	Bedeutung
18_D	Start mit gesetztem Steuerbit STOP
19_D	Start bei externem STOP aktiv
20_D	Start bei fehlender Impulsfreigabe (intern oder extern)
21_D	Start mit gesetztem STOP_REF_EN bei aktivem Referenznocken
22_D	Start ohne Referenz (bei Schrittmaßfahrt absolut)
23_D	Start bei anstehendem Diagnosefehler
24_D	Start wurde durch CPU/Master-STOP abgebrochen
25_D	Start mit falscher Betriebsart (ungleich der Anforderung)
26_D	Wegstrecke bzw. Positionsangabe ist unzulässig
27_D	Multiplikator G für die Geschwindigkeit ist Null
28_D	Frequenz bei Drehzahlbetrieb ist unzulässig
Fehlerursachen bei einem Parametrierauftrag bzw. bei der Grundparametrierung	
32_D	Angabe für die Basisfrequenz ist unzulässig
33_D	Multiplikator n für die Start-Stop-Frequenz ist Null
34_D	Multiplikator i für die Beschleunigung / Verzögerung ist Null
35_D	Rückmeldewert für die Rückmeldeschnittstelle ist unzulässig
36_D	Kombination der Funktionen von DI0 und DI1 ist unzulässig (Endschalter)
37_D	Angabe für das Ende des Verfahrbereichs ist unzulässig

6.2.8.5 Diagnose des 1STEP 5V

Diagnose durch LED-Anzeige

LED-Anzeige am 1STEP 5V

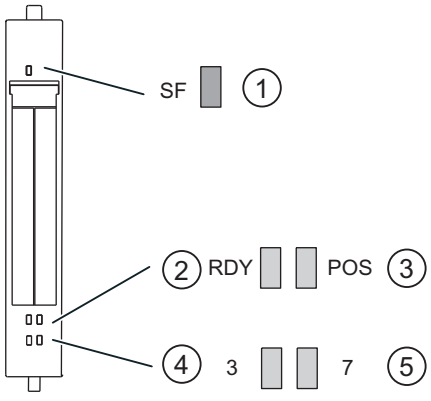

① Sammelfehler (rot)
② bereit für Fahrauftrag (grün)
③ Positionierung läuft (grün)
④ Statusanzeige für Digitaleingang 0 (grün)
⑤ Statusanzeige für Digitaleingang 1 (grün)

Status- und Fehleranzeigen durch LEDs am 1STEP 5V

Die folgende Tabelle zeigt die Status- und Fehleranzeigen am 1STEP 5V.

Ereignis (LED)					Ursache	Maßnahme
SF	RDY	POS	3	7		
ein					Keine Parametrierung oder falsches Modul gesteckt. Diagnosemeldung liegt vor.	Überprüfen Sie die Parametrierung. Werten Sie die Diagnose aus.
	ein				wenn Modul korrekt parametriert und Impulsfreigabe erteilt ist	
		ein			wenn Fahrauftrag läuft	
			ein		DI 0 ist aktiviert.	
				ein	DI 1 ist aktiviert.	

Fehlertypen

Informationen zum Aufbau der kanalbezogenen Diagnose finden Sie im Gerätehandbuch zum Interfacemodul, das Sie in Ihrer ET 200S-Station eingesetzt haben.

1STEP 5V Fehlertypen

Die folgende Tabelle zeigt die Fehlertypen am 1STEP 5V.

Fehlertyp		Bedeutung	Abhilfe
1_D	00001: Kurzschluss	Kurzschluss der Geberversorgung.	Überprüfen Sie die Verdrahtung zu den Schaltern. Korrektur der Prozessverdrahtung.
9_D	01001: Fehler	Interner Modulfehler ist aufgetreten.	Austausch des Moduls.
16_D	10000: Parametrierfehler	Modul ist nicht parametriert.	Korrektur der Parametrierung.

6.2.9 IM 174 einsetzen

6.2.9.1 Übersicht

Interface Modul für analoge Antriebe und Schrittmotoren

Eine IM 174-Baugruppe (Interface Module) ist eine Interface-Baugruppe, über die bis zu 4 Antriebe mit analoger Sollwertschnittstelle mit einem TTL- oder SSI-Geber pro Achse am taktsynchronen PROFIBUS betrieben werden können.

Die Kommunikation zwischen Steuerung und IM 174 erfolgt ausschließlich über PROFIBUS, über einen IM 174-spezifischen Telegrammtyp, der neben digitalen Ein-/Ausgangsdaten pro Antrieb einen nach PROFIDrive-Profil spezifizierten Telegrammtyp (Standardtelegramm 3 und 81) enthält. Im Rahmen der zyklischen DP-Kommunikation werden die Antriebs-Istwerte

(Geberwerte) von der IM 174-Baugruppe über den taktsynchronen PROFIBUS zur Steuerung und die von der Steuerung errechneten Drehzahlsollwerte zur IM 174-Baugruppe übertragen.

Die übertragenen Drehzahlsollwerte werden dann von der IM 174-Baugruppe als Analogwerte bzw. Pulse an die Antriebe ausgegeben.

Anwendungsgebiet

IM 174 dient folgenden Steuerungsaufgaben:

- analoge Antriebe steuern
- Schrittmotoren steuern
- Geberdaten einlesen

6.2.9.2 IM174 projektieren und programmieren

IM174 projektieren

Erforderliche Schritte - Überblick

Sie projektieren die IM 174 im Inspektorfenster unter "Eigenschaften". Folgende Schritte müssen Sie durchführen:

1. Eine PROFIBUS-Baugruppe der Hardware-Konfiguration hinzufügen
2. Taktsynchronen PROFIBUS parametrieren
3. Antrieb- und Geberparameter für die Achsen festlegen

Taktsynchronen PROFIBUS parametrieren

1. Im Bereich "PROFIBUS-Adresse" wählen Sie das PROFIBUS-Subnetz und weisen Sie eine PROFIBUS-Adressesse zu.

   Hinweis

   PROFIBUS-Adresse am DIL-Schalter

   An der PROFIBUS-Baugruppe IM 174 wird die PROFIBUS-Adresse über DIL-Schalter eingestellt. Damit eine am DIL-Schalter eingestellte PROFIBUS-Adresse aktiv wird, ist ein Netz-Aus/Netz-Ein der IM 174 notwendig. Die zugewiesene PROFIBUS-Adresse muss mit der am DIL-Schalter eingestellten PROFIBUS-Adresse übereinstimmen.

2. Im Bereich "Taktsynchronisation" aktivieren Sie das Optionskästchen "DP-Slave auf äquidistanten DP-Zyklus synchronisieren" und stellen Sie die Ti/To-Zeiten ein.
3. Im Bereich "Geber und Antriebe" wählen Sie den Telegrammtyp (Seite 2781) aus.
4. Im Bereich "Adressen" legen Sie die Adressbereiche für die Ein- und Ausgangsparameter der Achsen fest und weisen Sie ein Prozessabbild zu.

Antrieb- und Geberparameter für die Achsen festlegen

1. Im Bereich "Geber und Antriebe" wählen Sie für jede Achse den Antriebstyp (Seite 2772) und stellen Sie die Antriebsparameter (Seite 2772) ein.
2. Im Bereich "Geber und Antriebe" wählen Sie für jede Achse den Gebertyp (Seite 2772) und stellen Sie bei Bedarf die erforderlichen Geberparameter (Seite 2777) ein.
3. Im Bereich "Parameter" können Sie die Parameter (Seite 2780) für das Abschaltverhalten einstellen.

Geber und Antriebe

Über die Eigenschaft "Geber und Antriebe " in der Bereichsnavigation können Sie die, an IM 174 angeschlossenen, Antriebe und Geber parametrieren.

Antriebstypen

Folgende Antriebstypen können gewählt werden:

- Servo (Analogantrieb)
- Stepper (Schrittantrieb)

Gebertypen für analoge Antriebe

Folgende Gebertypen können für analoge Antriebe ausgewählt werden

- Gebertyp nicht vorhanden
- Gebertyp TTL
- Gebertyp SSI

Gebertypen für Schrittantriebe

Folgende Gebertypen können für Schrittantriebe ausgewählt werden

- Gebertyp nicht vorhanden
- Gebertyp TTL
- Gebertyp SSI
- Gebertyp Stepper

Antriebsparameter

Antriebstypen

Folgende Antriebstypen können gewählt werden:

- Servo (Analogantrieb)
- Stepper (Schrittantrieb)

Antriebtyp Servo

Wenn Sie den Antriebtyp Servo gewählt haben, können Sie über das Optionsfeld "unipolar" den Spannungsbereich der analogen Ausgangsspannung umschalten.

unipolar nicht angewählt

Ist die Option "unipolar" **nicht** aktiv, wird als Sollwert eine analoge Spannung im Bereich von **-10V** bis **+10V** ausgegeben. Dadurch kann der Antrieb in zwei Richtungen verfahren werden.

unipolar angewählt

Ist die Option "unipolar" **aktiv**, wird als Sollwert eine analoge Spannung im Bereich von **0V** bis **+10V** ausgegeben. Die Drehrichtung wird dann von der IM 174, abhängig vom aktuellen Drehzahlsollwert, über einen digitalen Ausgang der IM 174 ausgegeben:

- Drehrichtungssignal für Achse 1 -> Digitaler Ausgang X11, Pin 13
- Drehrichtungssignal für Achse 2 -> Digitaler Ausgang X11, Pin 15
- Drehrichtungssignal für Achse 3 -> Digitaler Ausgang X11, Pin 17
- Drehrichtungssignal für Achse 4 -> Digitaler Ausgang X11, Pin 19

Alt.DrvRdy

Abhängig von Ihrem verwendeten Antrieb liefert das anliegende Signal Drive Ready unterschiedliche Zustandsmeldungen:

- Der Antrieb meldet durch das Signal Drive Ready den Zustand "bereit zum Einschalten". Um den Antrieb einzuschalten, ist erst noch eine Freigabe durch die Steuerung erforderlich.
- Der Antrieb meldet durch das Signal Drive Ready den Zustand "Bereitschaft". In diesem Zustand ist der Antrieb regelungsbereit und folgt direkt einem Sollwert.

Aktivieren Sie die Funktion "Alt.DrvRdy" **nicht**, wenn Ihr Antrieb bei einem Drive Ready-Signal den Zustand "bereit zum Einschalten" meldet (meist bei Analogantrieben).

Aktivieren Sie die Funktion "Alt.DrvRdy", wenn Ihr Antrieb bei einem Drive Ready-Signal den Zustand "Bereitschaft" meldet (vor allem bei Schrittantrieben).

Informationen, welche Statusmeldung Ihr verwendeter Antrieb liefert, entnehmen Sie der Produktinformation Ihres Antriebes.

Antriebstyp Stepper

Beim Antrieb "Stepper" müssen Sie in den Feldern "Max. Frequenz [Hz]" und "Norm. Frequenz [Hz]" antriebsspezifische bzw. prozessspezifische Frequenzwerte [Hz] eingeben. Zum Betrieb eines Schrittmotors an einer IM 174 wird die nötige Leistungselektronik (z.B. FM Stepdrive) sowie der Schrittmotor benötigt.

In folgendem Bild wird ein prinzipieller Aufbau einer IM 174 mit Schrittantrieben mit und ohne Geber dargestellt:

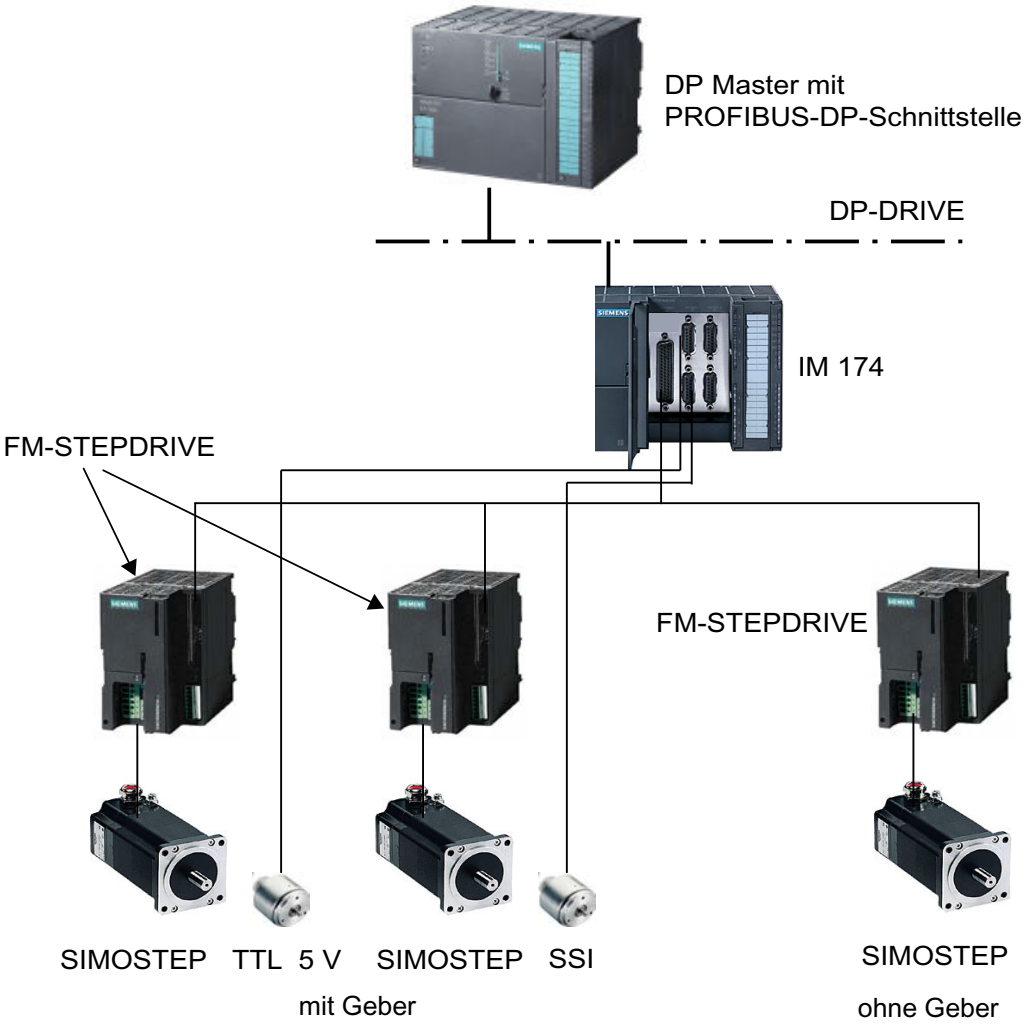

Bild 6-2 Prinzipieller Aufbau einer IM 174 mit FM-STEPDRIVE und SIMOSTEP-Motor

Alt.DrvRdy

Abhängig von Ihrem verwendeten Antrieb liefert das anliegende Signal Drive Ready unterschiedliche Zustandsmeldungen:

- Der Antrieb meldet durch das Signal Drive Ready den Zustand "einschaltbereit". Um den Antrieb einzuschalten, ist erst noch eine Freigabe durch die Steuerung erforderlich.
- Der Antrieb meldet durch das Signal Drive Ready den Zustand "Bereitschaft". In diesem Zustand ist der Antrieb regelungsbereit und folgt direkt einem Sollwert.

Aktivieren Sie die Funktion "Alt.DrvRdy" **nicht**, wenn Ihr Antrieb bei einem Drive Ready-Signal den Zustand "einschaltbereit" meldet (meist bei Analogantrieben).

Aktivieren Sie die Funktion "Alt.DrvRdy", wenn Ihr Antrieb bei einem Drive Ready-Signal den Zustand "Bereitschaft" meldet (vor allem bei Schrittantrieben).

Informationen, welche Statusmeldung Ihr verwendeter Antrieb liefert, entnehmen Sie der Produktinformation Ihres Antriebes.

Max. Frequenz [Hz] und Norm. Frequenz [Hz]

In den Feldern "Max. Frequenz [Hz]" (maximale Frequenz) und "Norm. Frequenz [Hz]" (Nenn-Frequenz) müssen Sie antriebsspezifische bzw. prozessspezifische Frequenzwerte [Hz] eingeben.

Berechnung der Norm. Frequenz

Die Norm. Frequenz lässt sich nach folgender Formel berechnen:

Norm. Frequenz [Hz] = n [U/min] / 60 * Auflösung am Schrittmotor

n [U/min]:	Drehzahl des Schrittmotors (charakteristische Werte liegen zwischen 500 und 1000 Umdrehungen pro Minute)
Auflösung am Schrittmotor:	Anzahl der Inkremente am Schrittmotor (charakteristische Werte: 500, 1000, 5000)

Berechnung der Max. Frequenz

Die Max. Frequenz lässt sich nach folgender Formel berechnen:

Max. Frequenz [Hz] = n_{max} [U/min] / 60 * Auflösung am Schrittmotor

n_{max} [U/min]:	Maximale Drehzahl des Schrittmotors (charakteristische Werte liegen zwischen 500 und 1000 Umdrehungen pro Minute)
Auflösung am Schrittmotor:	Anzahl der Inkremente am Schrittmotor (charakteristische Werte: 500, 1000, 5000)

Die maximale Drehzahl n_{max} lässt sich aus der charakteristischen Betriebskennlinie für Schrittmotoren bestimmen. Dabei ist über das technologisch erforderliche Moment M_t eine Maximaldrehzahl n_{max} des Motors zu ermitteln, die einer Maximalgeschwindigkeit der Achse entspricht.

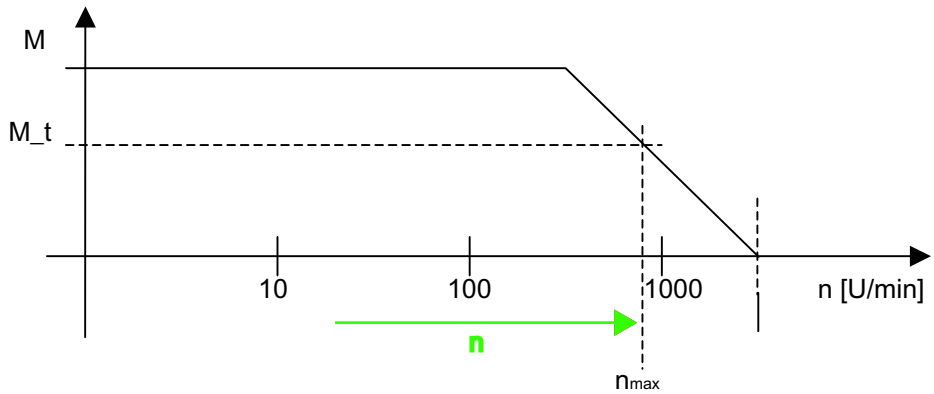

Bild 6-3 Charakteristische Betriebskennlinie eines Schrittmotors

Hinweis

Überlastungsbereich

Um einen Betrieb im Überlastungsbereich zu vermeiden, verwenden Sie den Antrieb nur bei Norm. Frequenz!

Beschreibung des Verhaltens eines Schrittmotors

Für prozessspezifische Anforderungen ist die Positionsgenauigkeit, die Drehzahl n sowie das vom Motor aufgebrachte Drehmoment M von entscheidender Bedeutung. Um diese Werte optimal bestimmen zu können, muss das Verhalten des Schrittmotors betrachtet werden. Ab einer bestimmten Drehzahl des Schrittmotors (etwa bei 500U/min) fällt das vom Motor aufgebrachte Drehmoment logarithmisch ab und geht bei einer Maximaldrehzahl gegen den Wert 0 (etwa bei 3000U/min). Die konkreten Daten sind dem Datenblatt des eingesetzten Motors zu entnehmen.

Folgen des Betriebes im Überlastungsbereich

Wenn der Schrittmotor einmal das abgeforderte Moment nicht aufbringen kann verliert er die Synchronisation auf die vorgegebene Frequenz und seine Drehzahl bricht zusammen. Das kann bis zum Stillstand führen. Eine erneute Bewegungsaufnahme in diesem Zustand ist nur nach einer zwischenzeitlichen Sollwertvorgabe von 0 sichergestellt. Bei einer Positionierachse ohne zusätzlichen Geber geht dadurch die Wegposition und damit die Synchronisation der Achse verloren.

Hinweis

Schrittantrieb mit Gebertyp "Stepper" (Betrieb PULSE REFEED)

Im Betrieb mit einem Gebertyp "Stepper" (Betrieb PULSE REFEED) kann es im Stillstand zu einem Schwingen des Schrittmotors kommen.

Dies tritt mit der default-Parametrierung eines Projekts auf, wenn der Stepper durch eine vorausgehende Verfahrbewegung auf einen Positionswert gefahren wurde, der genau zwischen zwei natürlichen Schritt-Teilungen des Steppers liegt.

Schrittantrieb mit TTL / SSI - Gebern

Bei Verwendung von TTL / SSI Gebern ist auf das Verhältnis aus Geberauflösung (incl. Feinauflösung) und Schrittauflösung des Steppers zu achten.

Um im Stillstand ein Schwingen zwischen zwei Positionswerten zu verhindern, muss die Schrittweite des Steppers wegmäßig kleiner sein, als die Wegrepräsentation in der Geberrückmeldung (d.h. die Schrittweite des Steppers muss größer sein als die Geberauflösung). Gegebenenfalls müssen niedriger auflösende Geber eingesetzt werden oder eine höhere Schrittauflösung am Stepper eingestellt werden.

Geberparameter

Gebertypen für analoge Antriebe

Folgende Gebertypen können für analoge Antriebe ausgewählt werden:

- Gebertyp nicht vorhanden
- Gebertyp TTL
- Gebertyp SSI

Gebertypen für Schrittantriebe

Folgende Gebertypen können für Schrittantriebe ausgewählt werden:

- Gebertyp nicht vorhanden
- Gebertyp TTL
- Gebertyp SSI
- Gebertyp Stepper

Gebertyp "nicht vorhanden"

Für **Antriebstyp "Servo"** gilt: Die Achse x ist nicht vorhanden bzw. soll nicht betrieben werden. Im PROFIBUS-Telegramm werden für diese Achse leere Nutzdaten übertragen.

Für **Antriebstyp "Stepper"** gilt: Der Stepper wird geberlos betrieben.

Wird der Stepper für den geberlosen Betrieb parametriert, ist ausschließlich die Nutzung als Drehzahlachse möglich.

Wird kein Geber (SSI/TTL) verwendet, kann der Gebertyp "Stepper" gewählt werden, um eine Positionierung zu ermöglichen, auch wenn kein Geber angeschlossen ist.

Gebertyp TTL

Beim Gebertyp "TTL" können Sie folgende Geberparameter einstellen:

- **Auflösung**: Geberauflösung in Geberimpulse pro Geberumdrehung
- **Drehzahlberechnung aktiviert**: IM 174 berechnet die Drehzahl, wenn das Optionskästchen aktiviert ist.

- **Normierungsdrehzahl**: Dieses Eingabefeld ist nur sichtbar wenn die Option "Drehzahlberechnung aktiviert" gesetzt wird. Geben Sie die Nenndrehzahl des Motors in U/min an. Durch die Berechnung der Drehzahl durch IM174 wird die Geschwindigkeit mit weniger Rauschen ermittelt.
- **Reservierte Bits für Feinauflösung**: Einstellung 0 - 15
  Über den Parameter "Reservierte Bits für Feinauflösung", wird die gewünschte Impulsvervielfachung der im Geberistwert G1_XIST1 und G1_XIST2 übertragenen Geberinkremente angegeben.
  Die Einstellung entspricht einer Impulsvervielfachung zwischen 2^0 = 1 bis 2^{15} = 32768.

Hinweis

Minimale Feinauflösung bei TTL-Gebern

Bei TTL-Gebern ist immer eine minimale Feinauflösung von 2^2 (= 4) zu berücksichtigen und einzustellen.

übertragener Geberistwert G1_XIST1 bzw. G1_XIST2

31		0
Überläufe	akt. Geberinkremente	Reservierte Bits für Feinauflösung

Gebertyp SSI

Beim Gebertyp "SSI" können Sie folgende Geberparameter einstellen:

- **Parität**: Diese Option ist anzuwählen, wenn die Übertragung der Geberdaten vom Geber zur IM174 mit Parity-Bit erfolgt.
- **MsgLänge**: Einstellung 0 - 25
  Anzahl der vom Geber gesendeten Nutzdatenbits
- **Encoding**: Es werden folgende Geberkodierungen unterstützt:
  - Binary
  - Gray

- **Baudrate**: Die Baudrate muss für alle SSI-Geber identisch eingestellt werden. Bei unterschiedlich eingestellten Baudraten wird die Baudrate des SSI-Gebers mit der höchsten Geber-Nummer vergeben.
  Es werden folgende Baudraten unterstützt:
  - 187,5 kBit/s
  - 375 kBit/s
  - 750 kBit/s
  - 1,5 MBit/s
  - 3,0 MBit/s

  Hinweis

  Stellen Sie maximal 187,5 kBit/s ein. Schnellere Übertragungsraten sind bei dieser Baugruppe nicht sinnvoll.

- **Reservierte Bits für Feinauflösung**: Einstellung 0 - 15
  Über den Parameter "Reservierte Bits für Feinauflösung", wird die gewünschte Impulsvervielfachung der im Geberistwert G1_XIST1 und G1_XIST2 übertragenen Geberinkremente angegeben.
  Die Einstellung entspricht einer Impulsvervielfachung zwischen 2^0 = 1 bis 2^{15} = 32768.

Gebertyp Stepper

Schrittantrieb im Betrieb PULSE REFEED

Eine Achse x mit dem Antriebtyp "Stepper" lässt sich mit einem Gebertyp "Stepper" betreiben. In diesem Betrieb werden die Sollwerte als Istwerte von der IM 174 an die Steuerung zurückgemeldet.

Beim Gebertyp "Stepper" können Sie folgende Geberparameter einstellen:

- **Motorüberwachung**: Mit der Aktivierung des Kontrollkästchens "Motorüberwachung" wird die Anzahl der Schritte eines Schrittmotors für einen vorgegebenen Referenzpunktabstand (siehe "Bero-Distanz") ständig überwacht. Im geberlosen Betrieb dient der verwendete Schrittantrieb somit gleichzeitig als Istwertmelder (pulse refeed).
- **Bero-Distanz**: Bei "Bero-Distanz" wird die Anzahl der Schritte (für den verwendeten Schrittantrieb) zwischen zwei Bero Signalen eingegeben. Die Strecke zwischen beiden Bero Signalen ist der Referenzpunktabstand.
- **Bero-Toleranz**: Einstellung 0 - 65535
  Bei "Bero-Toleranz" wird die zulässige Abweichung der Schritte von der Bero-Distanz eingegeben. Der resultierender Schrittanzahlbereich liegt innerhalb folgender Toleranz:
  Resultierender Schrittanzahlbereich = Bero-Distanz ± 1/2 * Bero-Toleranz
- **Reservierte Bits für Feinauflösung**: Einstellung 0 - 15
  Über den Parameter "Reservierte Bits für Feinauflösung", wird die gewünschte Impulsvervielfachung der im Geberistwert G1_XIST1 und G1_XIST2 übertragenen Geberinkremente angegeben.
  Die Einstellung entspricht einer Impulsvervielfachung zwischen 2^0 = 1 bis 2^{15} = 32768.

Parameter

Über die Eigenschaft "Parameter" in der Bereichsnavigation bestimmen Sie wichtige Parameter für IM 174.

Shutdown Rampe [ms]

Der Wertebereich liegt bei 0 - 65535 ms.

Über den Parameter "Shutdown Rampe" kann eine zeitlich lineare Funktion vorgegeben werden, nach der bei einem erkannten Fehler alle Antriebe, die an IM 174 angeschlossen sind, auf den Sollwert 0 abgebremst werden.

Bei folgenden Fehlern wird die Shutdown Rampe wirksam:

- Temperaturalarm der Baugruppe (ca. 90°C EIN, ca. 85°C AUS)
- Lebenszeichenfehler
- Synchronisationsfehler zwischen Master und Slave

Hinweis

Parameterwert 0

Ein Parameterwert von 0 bewirkt bei einem Fehler ein Austrudeln des Antriebs.

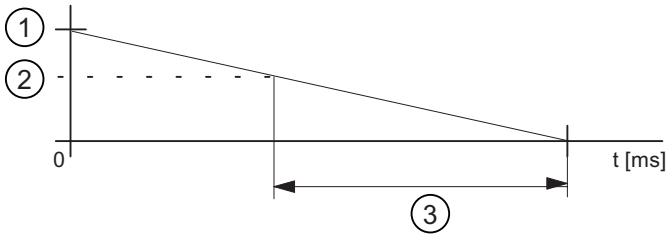

1) maximaler Sollwert
2) aktueller Sollwert
3) Parameter-Wert: Shutdown Rampe

Bild 6-4 Parameter: Shutdown Rampe

Shutdown Verzögerungszeit

Der Wertebereich liegt bei 0 - 65535 s.

Über den Parameter "Shutdown Verzögerungszeit" kann eine Zeit vorgegeben werden, nach der, nach einem aufgetretenen **Temperaturalarm,** die Shutdown Rampe aktiviert wird.

Die Shutdown Verzögerungszeit wird, bei Überschreiten einer Temperatur von 90 °C in der Baugruppe, gestartet. Geht, während die Verzögerungszeit aktiv ist, die Temperatur auf ca. 85 °C zurück, wird die Shutdown Rampe nicht aktiviert und die Shutdown Verzögerungszeit zurückgesetzt.

Tolerierbare Lebenszeichenfehler

Der Wertebereich liegt bei 0 - 15.

Über den Parameter "Tolerierbare Lebenszeichenfehler" wird die Anzahl tolerierbarer Lebenszeichenfehler des DP-Master vorgegeben. Bei Überschreitung der parametrierten Anzahl wird die Shutdown Rampe aktiviert. Der Wertebereich liegt bei

611U-Konformmodus

Im 611U-Konformmodus wird die Signal-Quelle für das Referenzieren der Achsen nicht mehr über das PROFIDrive-Standardtelegramm (STD3, Gebersteuerwort G1_STW), sondern über das zusätzliche digitale Ausgangswort im PROFIBUS-Telegramm von IM 174 (siehe Tabelle "Telegrammaufbau" im Abschnitt "Telegrammtyp") vorgegeben.

611U-Konformmodus:

- Nicht angewählt
  Die Vorgabe der Signal-Quelle für das Referenzieren erfolgt über das Gebersteuerwort Gx_STW im PROFIDrive-Standardtelegramm.
- Angewählt
  Die Vorgabe der Signal-Quelle für das Referenzieren erfolgt über das zusätzliche digitale Ausgangswort im PROFIBUS-Telegramm.

Telegrammtyp auswählen

Telegrammtyp

Der DP-Slave IM 174 wird mit zwei spezifischen Telegrammtypen betrieben: vier Achsen mit jeweils einem Geber (Standardtelegramm 3 und 81) und E/A-Daten.

Hinweis

Standardtelegramm 81

Das Standardtelegramm 81 kann nur für Geber verwendet werden.

Telegrammaufbau

Telegrammaufbau

In der folgenden Übersicht wird der Telegrammaufbau erläutert.

Tabelle 6-1

Telegrammtyp	Beschreibung
4 Achsen mit jeweils einem Geber, Standardtelegramm 3 + EA, PZD-5/9 A/E 1/1 oder Telegramm 81 + EA, PZD-2/6 + A/E 1/1	4 mal Standardtelegramm 3 oder 81 und je 1 PZD-Wort für digitale E/A-Daten
PZD x/y Anzahl der Prozessdaten-Worte, x: Sollwert, y: Istwert, z.B. PZD-5/9: 5 Prozessdaten-Worte für Sollwerte 9 Prozessdaten-Worte für Istwerte	
IM174-Telegrammaufbau Achse1: STD 3 oder STD 81; Achse2: STD 3; Achse3: STD 3; Achse4: STD 3; E/A: A-Wort — Sollwerte (Master -> Slave) Low — High Achse1: STD 3 oder STD 81; Achse2: STD 3; Achse3: STD 3; Achse4: STD 3; E/A: E-Wort — Istwerte (Slave -> Master) STD 3: Standardtelegramm 3 nach PROFIDrive-Spezifikation V3.0 A-Wort: digitale Ausgangsdaten (16Bit) E-Wort: digitale Eingangsdaten (16Bit)	
Standardtelegramm 3: Drehzahlsollwert-Schnittstelle 32Bit mit 1 Geber PZD1: STW1; PZD1 PZD3: NSOLL_B; PZD4: STW2; PZD5: G1_STW — Sollwert (Master -> Slave) Low — High PZD1: ZSW1; PZD2 PZD3: NIST_B; PZD4: ZSW2; PZD5: G1_ZSW; PZD6 PZD7: G1_XIST1 PZD8 PZD9: G1_XIST2 — Istwert (Slave -> Master)	
Standardtelegramm 81: Drehzahlsollwert-Schnittstelle 32Bit mit 1 Geber PZD1: STW2; PZD2: G1_STW1 — Sollwert (Master -> Slave) PZD1: ZSW2; PZD2: G1_ZSTW1; PZD3 PZD4: G1_XIST1; PZD5 PZD6: G1_XIST2 — Istwert (Slave -> Master)	

Telegrammtyp	Beschreibung
A-Wort (dig. Ausgangsdaten 16Bit)	
E-Wort (dig. Eingangsdaten 16Bit)	

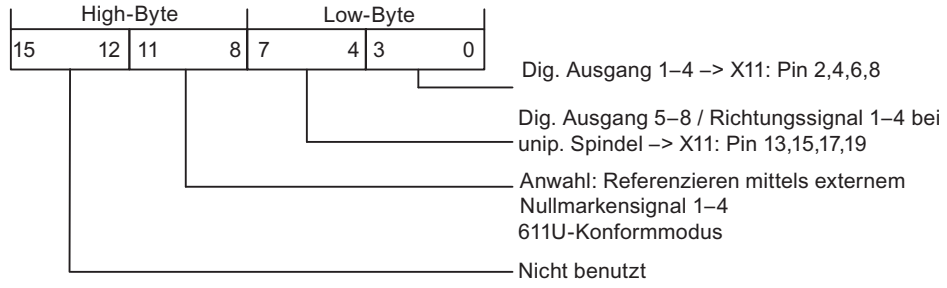

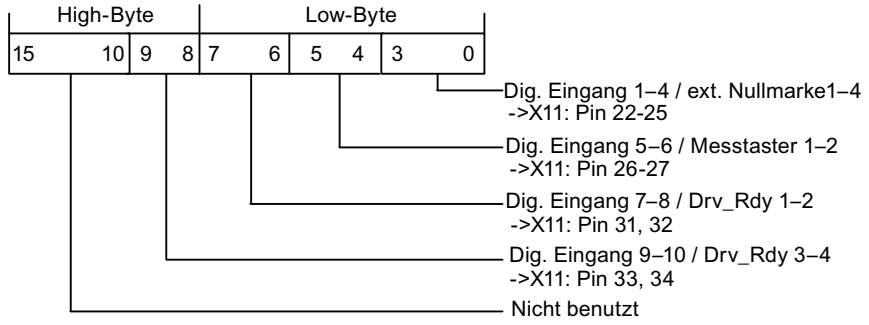

Hinweis

Der eingestellte Telegrammtyp des DP-Slaves IM 174 muss mit dem in der Steuerung eingestellten Telegrammtyp übereinstimmen. Es findet kein automatischer Abgleich statt.

Gebersteuerwort Gx_STW

In der folgenden Übersicht wird die Beschreibung des Gebersteuerwortes (Auszug) erläutert. Die Übersicht bezieht sich auf folgende Anforderungen:

- Referenzmarkensuche
- Fliegendes Messen
- Geberfehler

Bit	Name		Signalzustand, Beschreibung		
0	Referenz-markensuche oder Fliegendes Messen	Funktionen	Bit 7 = 0 => Anforderung: Referenzmarkensuche		
			Bit	Bedeutung	Referenzieren mittels:
			0	Funktion 1:	Geber-Nullmarke (außer "611 U-Konformmodus")
			1	Funktion 2:	Nicht verwendet
			2	Funktion 3:	Nicht verwendet
			3	Funktion 4:	Nicht verwendet
1			Bit 7 = 1 => Anforderung: Fliegendes Messen		
			Bit	Bedeutung	
			0	Funktion 1:	Messtaster 1 steigende Flanke
			1	Funktion 2:	Messtaster 1 fallende Flanke
			2	Funktion 3:	Messtaster 2 steigende Flanke
			3	Funktion 4:	Messtaster 2 fallende Flanke
2			Hinweis • Bit x = 1 Funktion angefordert Bit x = 0 Funktion nicht angefordert • Wird mehr als eine Funktion aktiviert, können die Werte zu allen Funktionen erst dann gelesen werden, wenn alle Funktionen beendet sind und dies über das entsprechende Zustandsbit signalisiert wurde (G1_ZSW, Bit0 - Bit3 = 0). • Fliegendes Messen Die steigende und fallende Flanke des Messtasters können gleichzeitig aktiviert werden. Das Messtastersignal wird abhängig von der Richtung der Signaländerung erkannt. Die Messwerte werden nacheinander ausgelesen.		
3					
4		Kommando	Bit 6, 5, 4	Bedeutung	
5			000	--	
6			001	Funktion x aktivieren	
			010	Wert x lesen	
			011	Funktion x abbrechen	
7		Mode	0	Referenzmarkensuche	
			1	Fliegendes Messen	
15	Geberfehler quittieren		0 1	-- Geberfehler quittieren	

Hinweis

Messen auf steigende oder fallende Flanke

IM 174 unterstützt nur das Messen auf steigende oder fallende Flanke.

Zusätzlicher Geberistwert Gx_XIST2

In der folgenden Übersicht werden die Fehlercodes in Gx_XIST2 bei G1_ZSW, Bit15 == 1 beschrieben:

Tabelle 6-2 Fehlercodes in Gx_XIST2

G1_XIST2	Bedeutung	Mögliche Ursachen/Beschreibung
1_{Hex}	Gebersummenfehler	Die Signalpegel des Gebers sind zu klein, gestört (unzulängliche Schirmung) oder die Leitungsbruch-Überwachung hat angesprochen.
2_{Hex}	Nullmarken-Überwachung	Zwischen 2 Nullmarken des Gebers ist eine Schwankung der gemessenen Rotorlage aufgetreten (evtl. Geberstriche verloren).

Zustandswort ZSW_1 - Bit 11 bis Bit 14

In der folgenden Übersicht wird das Zustandswort ZSW_1 bezüglich der baugruppenspezifischen Meldungen Bit 11 bis Bit 14 beschrieben:

Tabelle 6-3 Zustandswort ZSW_1

Bit	Name	Beschreibung
11	Temperaturfehler	Eine bestimmte Temperatur im Gehäuse der Baugruppe IM 174 wurde überschritten.
12	PLL Synchronisationsfehler	IM 174 lässt sich nicht mit dem dem globalen Kontrollsignal synchronisieren.
13	Master Life Sign Fehler	IM 174 lässt sich nicht mit dem Master Lebenszeichensignal Synchronisieren.
14	Antriebsfehler	Es liegt ein Antriebsfehler vor, z. B. fehlt das Drive Ready Signal.

Nullmarkenüberwachung

Funktionsweise der Geber-Nullmarkenüberwachung

Die Nullmarkenüberwachung vergleicht die erkannten Geberinkremente zwischen zwei Nullmarken mit der konfigurierten Geberauflösung. Wird dabei eine Differenz festgestellt, wird ein Geberfehler gemeldet.

Spur A und Spur B sowie Spur Z des Gebers müssen sich gleichzeitig auf dem Level High befinden, damit die Nullmarkenüberwachung korrekt ausgeführt wird.

Einschränkungen:

- Der Ausfall einer Nullmarke an sich führt nicht unweigerlich zu einem Fehler, da erst bei Erreichen einer Nullmarke die Überprüfung angestoßen wird.
- Das Ergebnis der gezählten Geberinkremente MOD 16 muss ungleich null sein, damit ein Fehler gemeldet wird.
- Das Ergebnis der gezählten Geberinkremente MOD 10 muss ungleich null sein, damit ein Fehler gemeldet wird.

Beispiel:

Geberkonfiguration: TTL 2048 Striche

gezählte Geberinkremente 2049 -> Geberfehler

gezählte Geberinkremente 1024 -> kein Geberfehler

IM 174 programmieren

Für die Programmierung der IM 174 wird das Optionspaket "Easy Motion Control" empfohlen.

Anweisungen für IM 174

In der Anweisungsliste Easy Motion Control stehen folgende Anweisungen für IM 174 zur Verfügung:

- EncoderIM174 - Einsatz der IM 174 als Wegerfassungsbaugruppe für Inkremental- und Absolutgeber
- OutputIM174 - Einsatz eines Kanals der Baugruppe IM 174 als Ausgabebaugruppe

6.2.10 Easy Motion Control einsetzen

6.2.10.1 Positionieren mit Easy Motion Control

Produktübersicht

Überblick

Mit Easy Motion Control können Sie Antriebe geregelt positionieren. Die Bewegung wird allein durch Software (Anweisungen) in der CPU gesteuert. Zur Anbindung an die Antriebe ist **keine spezielle Hardware** notwendig, obwohl natürlich zur Erfassung der Lageistwerte und zur Ausgabe der Drehzahlsollwerte entsprechende Peripherie vorhanden sein muss.

Die Anweisungen von Easy Motion Control wurden nach der "Technical Specification Function blocks for motion control V1.0" von **PLCopen** erstellt.

Sie parametrieren über eine Projektiersoftware, die auch die Inbetriebnahme durch Testfunktionen unterstützt. Alle Parameter einer Achse sind in einem Datenbaustein hinterlegt, dem Achs-DB.

Sie können die Antriebe an zentraler Peripherie oder über einen in die CPU integrierten DP-Master / IO-Controller an dezentrale Peripherie anschließen.

Die Fahrt führen Sie durch, indem Sie die entsprechenden Anweisungen verschalten und diese im Anwenderprogramm ablaufen lassen.

Komponenten für die Positionierung

Das folgende Bild zeigt eine Beispielkonfiguration mit Easy Motion Control.

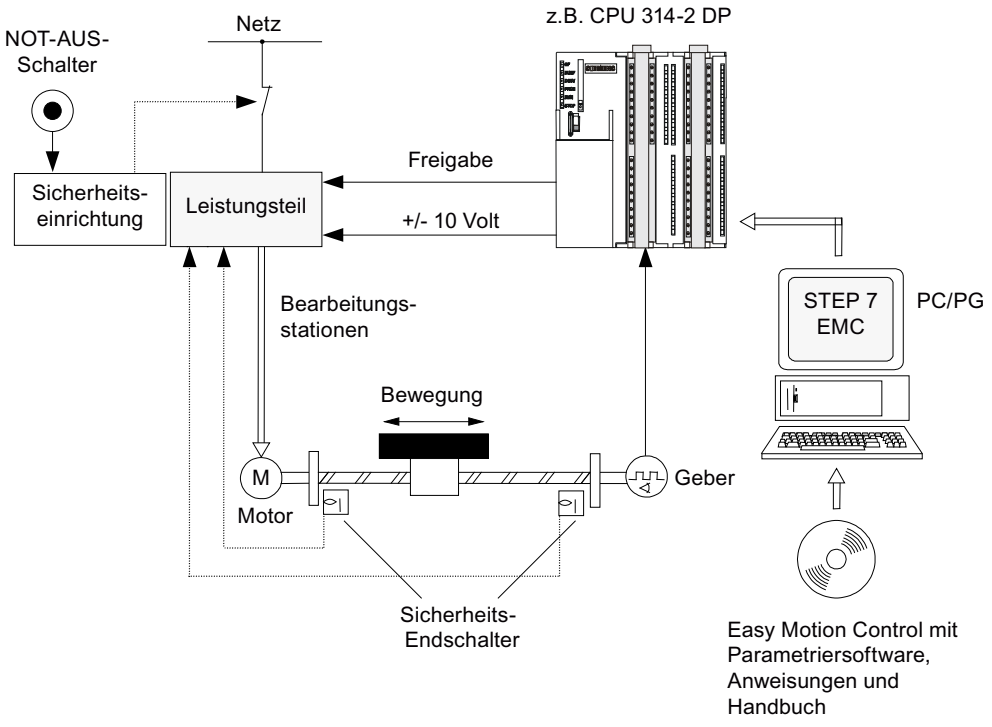

Sicherheitseinrichtung

Beim Ansprechen der Sicherheitseinrichtung schaltet der NOT-AUS-Schalter die Stromversorgung ab.

Leistungsteil

Das Leistungsteil steuert den Motor an. Die Sicherheits-Endschalter schalten das Leistungsteil ab.

Motor

Der Motor wird vom Leistungsteil angesteuert und treibt die Achse an.

Geber

Der Geber liefert Weg- und Richtungsinformationen.

Easy Motion Control

Easy Motion Control stellt folgende Funktionen zur Verfügung:

- Tippen
- Referenzpunktfahren
- Positionieren absolut/relativ
- Elektronisches Getriebe
- Bezugspunktsetzen
- Alle gängigen Überwachungsfunktionen für Positionieren
- Geschwindigkeitsoverride
- Lageregelung
- Simulation
- Vorkonfektionierte Eingangstreiber für Geberwerterfassung bzw. Ausgangstreiber für Analogausgabebaugruppen

Baugruppen zur Wegerfassung und Sollwertausgabe

Die Baugruppen bilden die Schnittstelle zwischen CPU und Maschine. Sie lesen die Achspositionen und geben Sollwerte an das Leistungsteil aus.

CPU

Die CPU führt das Anwenderprogramm aus, in dem die Anweisungen von Easy Motion Control eingebettet sind.

PC/PG

Ein PC/PG dient zum

- Parametrieren: Sie parametrieren Easy Motion Control entweder mit der Projektiersoftware oder direkt im Achs-Datenbaustein (siehe Technologieobjekt Achse konfigurieren (Seite 2799)).
- Programmieren: Sie binden die Anweisungen von Easy Motion Control direkt in Ihr Programm ein.

Unterstützte Hardware-Konfigurationen

Die nachfolgend dargestellten Hardware-Konfigurationen werden durch Treiber-Anweisungen komfortabel unterstützt. Diese werden mit Easy Motion Control mitgeliefert.

Eine Anpassung an nicht direkt unterstützte Konfigurationen ist durch ebenfalls mitgelieferte freie Treiber-Anweisungen möglich.

Das folgende Bild zeigt **dezentrale** Konfigurationen für Easy Motion Control.

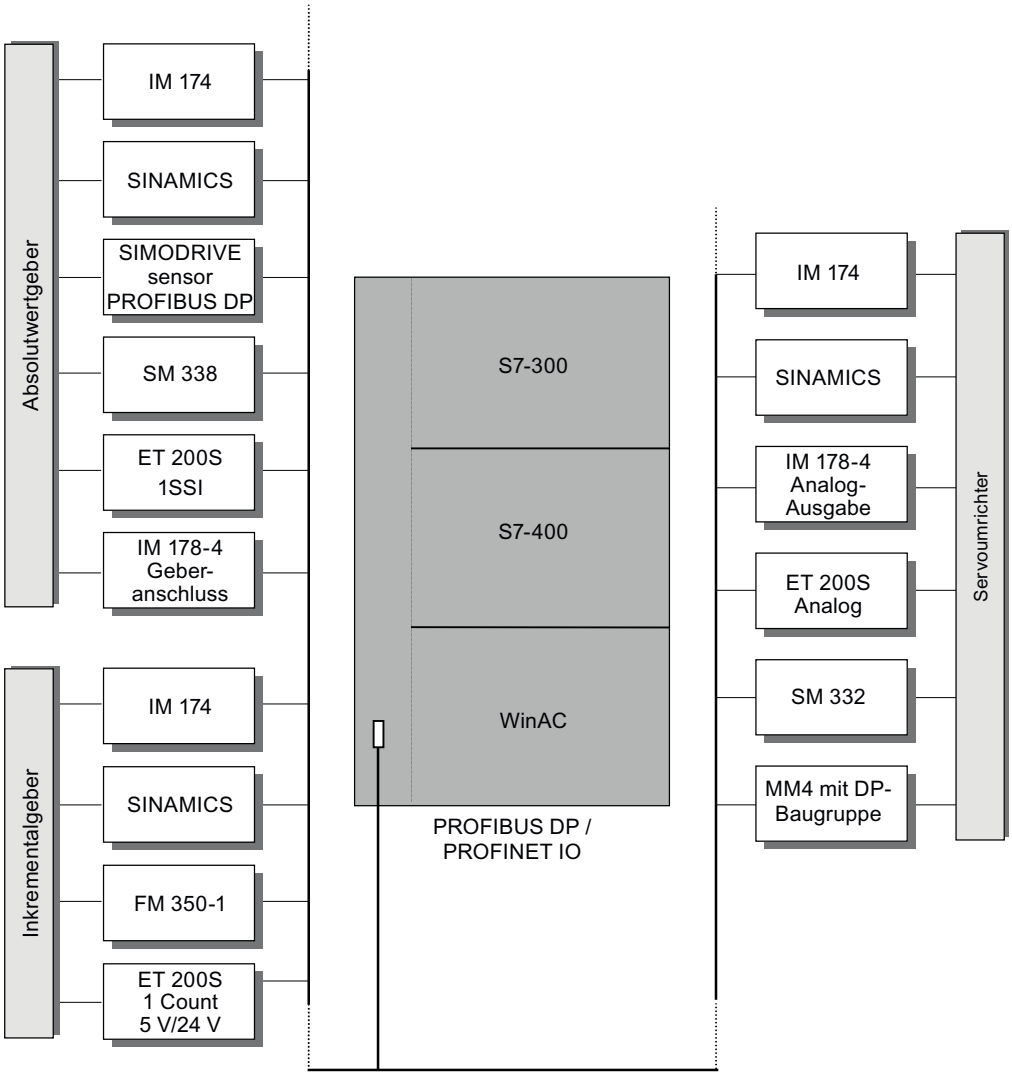

Das folgende Bild zeigt **zentrale** Konfigurationen für Easy Motion Control.

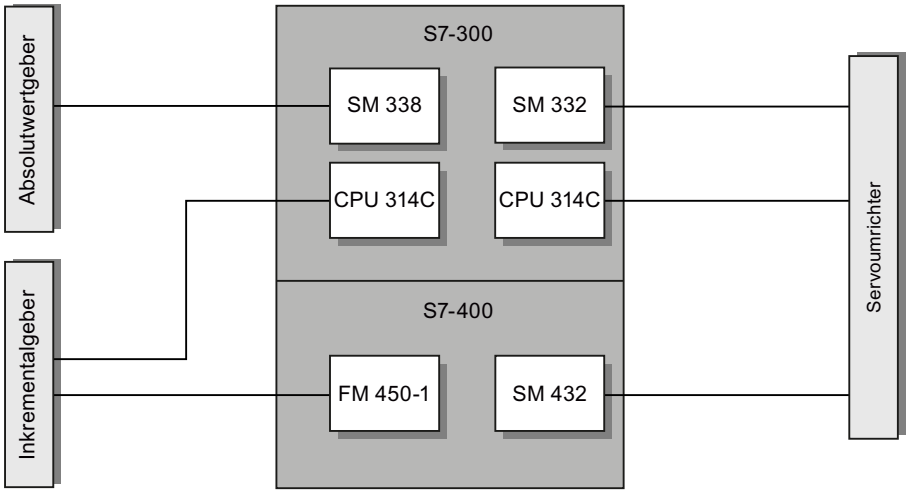

Grundlagen der Positionierung mit Easy Motion Control

Aufbau der Lageregelung

Bei der geregelten Positionierung wird die aktuelle Position der Achse mit der Sollposition verglichen. Abweichungen werden über den Lageregler sowohl während einer Fahrbewegung als auch im Stillstand ausgeregelt.

Das folgende Bild zeigt die Prinzipdarstellung für die geregelte Positionierung mit Easy Motion Control. Eine Fahranweisung (Seite 2822) gibt Positionssollwerte vor. Der Eingangstreiber erfasst den Positionsistwert der Achse. Der Lageregler (Seite 2809) vergleicht den Positionsistwert mit dem Positionssollwert und erzeugt aus der Differenz einen Geschwindigkeitssollwert. Der Ausgangstreiber gibt den normierten Geschwindigkeitssollwert aus.

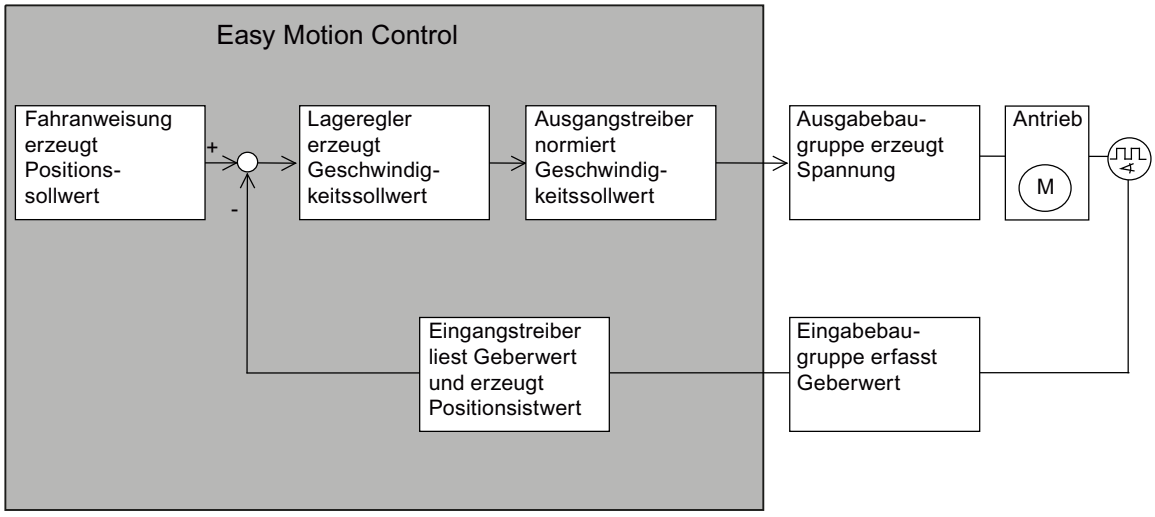

Das folgende Bild zeigt den Zusammenhang von Geschwindigkeit und zurückgelegtem Weg der Achse während einer Fahrbewegung.

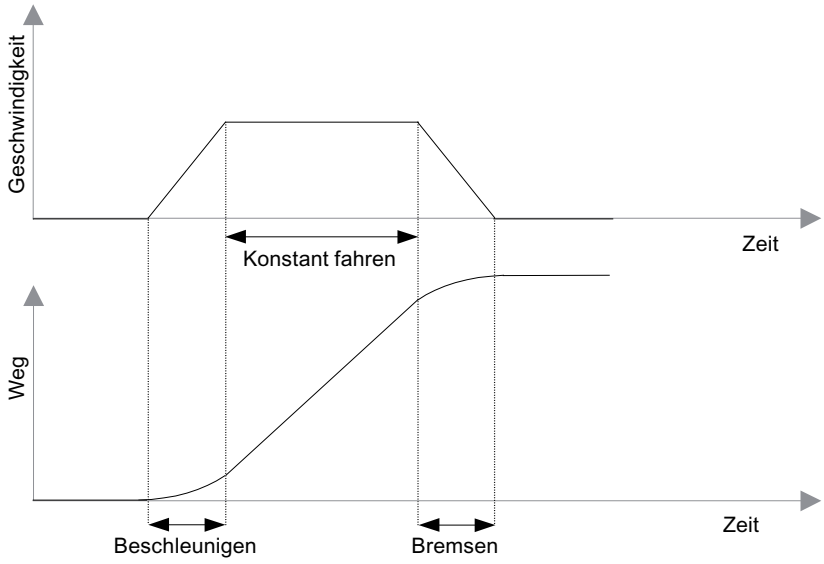

Begriffsdefinitionen

Ziel

Das Ziel ist die absolute bzw. relative Position der Achse, die bei einer Positionierung angefahren wird.

Fahrbewegung

Die Fahrbewegungen werden durch den Aufruf von Anweisungen ausgelöst, die im Folgenden Fahranweisungen genannt werden.

Restweg

Der Restweg ist die Differenz zwischen Zielposition und aktuellem Positionssollwert.

Schleppabstand / Schleppfehler

Während einer Fahrbewegung baut sich aufgrund der Achsträgheit eine Differenz zwischen dem aktuellen Positionssollwert und dem Positionsistwert auf, die als Schleppabstand bezeichnet wird. Überschreitet der Schleppabstand einen parametrierbaren Wert, so liegt ein Schleppfehler vor.

Überwachungsbereiche

- Arbeitsbereich:
  definiert den erlaubten Verfahrbereich Ihrer Achse, den Sie für Ihre Aufgabe durch die Software-Endschalter (Seite 2792) bzw. das Ende der Rundachse (Seite 2793) bestimmen.
- Stillstandsbereich:
  definiert einen parametrierbaren Bereich, den die Achse im Stillstand nicht verlassen darf. Der Stillstandsbereich liegt links und rechts vom Ziel.
- Zielbereich:
  Wenn der Sollwert am Ende einer Fahrt das Ziel erreicht hat, wird das Einlaufen der Achse in den Zielbereich überwacht. Der Zielbereich dient zur Überwachung der Positioniergenauigkeit Ihrer Anwendung und liegt links und rechts vom Ziel.
  Unmittelbar nach dem Erreichen des Zielbereichs beginnt die Überwachung des Stillstandsbereichs. Deshalb sollte der Stillstandsbereich größer sein als der Zielbereich.

Das folgende Bild zeigt die Lage der einzelnen Bereiche um das Ziel.

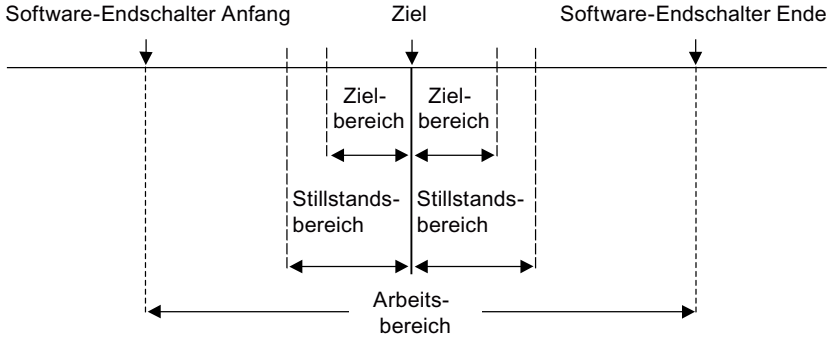

Linearachse

Beschreibung

Der **Arbeitsbereich** reicht vom Software-Endschalter Anfang bis zum Software-Endschalter Ende und ist begrenzt auf 1×2^{24} Längeneinheiten (im Koordinatenbereich zwischen $-2^{24}+6$ und $+2^{24}-6$).

Die **physikalische Achslänge** ist durch die mechanischen Anschläge Ihrer Achse begrenzt.

Der **Zahlenbereich** von Easy Motion Control ist nur relevant, wenn die Überwachung des Arbeitsbereichs ausgeschaltet ist. Er ist begrenzt auf $2 \times 2^{24}-12$ Längeneinheiten (von $-2^{24}+6$ bis $+2^{24}-6$).

Der **Geberbereich** ist

- bei einem Absolutwertgeber der vom Geber abgedeckte Bereich.
- bei einem Inkrementalgeber die Wegstrecke von -2^{31} bis $+2^{31}$ Schritten.

Bei Linearachsen ist der **Arbeitsbereich** (= Verfahrbereich) wie folgt begrenzt:

Arbeitsbereich < Physikalische Achslänge < Zahlenbereich von Easy Motion Control < Geberbereich

Das folgende Bild zeigt die verschiedenen Bereiche einer Linearachse.

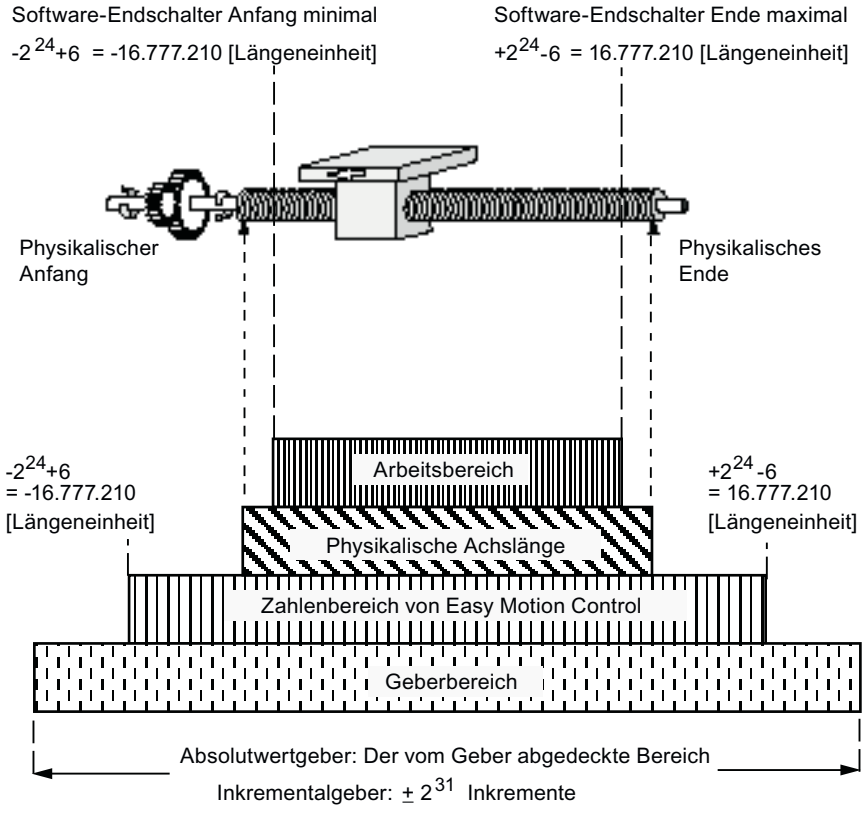

Rundachse

Beschreibung

Der **Arbeitsbereich** reicht vom Anfang der Rundachse bis zum Ende der Rundachse und ist begrenzt auf 1×2^{24} Längeneinheiten (im Koordinatenbereich zwischen $-2^{24}+6$ bis $+2^{24}-6$).

Der **Geberbereich**

- ist bei einem Inkrementalgeber beliebig.
- bestimmt sich bei einem Absolutwertgeber nach der Vorschrift:
  Geberbereich = Arbeitsbereich × n
  mit:
  - n = 1 bei Singleturn-Gebern
  - n > 1 und ganzzahlig bei Multiturn-Gebern

Bei einer Rundachse wird der Istwert nach jeder Umdrehung wieder auf den Anfang der Rundachse zurückgesetzt. Rundachsen können somit über das Ende bzw. den Anfang der Rundachse hinaus endlos verfahren werden. Die Achsposition wird immer als Wert zwischen Anfang und Ende der Rundachse angezeigt.

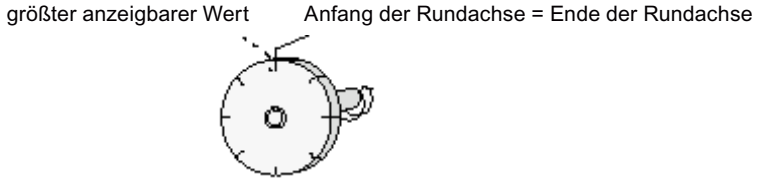

6.2.10.2 Funktionen mit Easy Motion Control

Übersicht

Easy Motion Control dient dazu, eine Achse mit wählbarer Beschleunigung, Geschwindigkeit und Verzögerung auf ein Ziel zu positionieren oder mit konstanter Geschwindigkeit zu verfahren. Die dazu nötigen Fahrbewegungen werden durch den Aufruf von Fahranweisungen ausgelöst.

Beim Starten eines Fahrauftrags wird an der Fahranweisung das Ziel oder die Wegstrecke der Fahrbewegung und evtl. die gewünschte Beschleunigung, Geschwindigkeit oder Verzögerung parametriert. Das Maßsystem (die Längeneinheit) dieser Werte kann frei gewählt werden (z. B. mm).

Tippen

Mit der Funktion Tippen bewegen Sie den Antrieb in positiver bzw. negativer Richtung. Ein Ziel wird nicht vorgegeben.

Die Funktion Tippen wird mit der Anweisung MC_MoveJog realisiert.

Synchronisieren

Um der realen Position einen reproduzierbaren Geberwert zuzuordnen, muss ein Bezug (Synchronisation) zwischen Achsposition und Geberwert hergestellt werden.

Die Funktion Synchronisieren wird mit der Anweisung MC_Home realisiert.

Synchronisieren mit Inkrementalgeber

Ein Inkrementalgeber liefert beim Einschalten der Anlage keine Angabe über die augenblickliche Lage der Achse. Statt dessen fängt er an der aktuellen Position der Achse zu zählen an.

Beim Wiedereinschalten der Anlage, nach Auftreten von Fehlern sowie nach einem Tausch des Gebers ist immer eine Synchronisation erforderlich.

Die Synchronisation eines Inkrementalgebers kann erfolgen

- durch eine Referenzpunktfahrt
- durch Bezugspunktsetzen

Synchronisieren mit Absolutwertgeber

Bei der Inbetriebnahme von Absolutwertgebern muss der vom Geber angezeigte Wert auf die tatsächliche Achsposition synchronisiert werden.

Eine erneute Synchronisation ist beim Wiedereinschalten der Anlage oder nach Auftreten von Fehlern nicht erforderlich. Nach einem Tausch des Gebers muss die Synchronisation dagegen erneut durchgeführt werden.

Die Synchronisation eines Absolutwertgebers kann nur durch Bezugspunktsetzen (auch Absolutwertgeberjustage genannt) erfolgen.

Referenzpunktfahrt

Die Achse wird in Richtung des Referenzpunktes bewegt. Nach Überfahren des Referenzpunktes ist die Achse synchronisiert, d. h. dem Zahlenwert des Inkrementalgebers ist ein Positionswert zugeordnet. Dem Referenzpunkt kann bei jeder Referenzpunktfahrt ein neuer Koordinatenwert zugewiesen werden.

Eine Referenzpunktfahrt ist nur mit Inkrementalgebern möglich.

Bezugspunktsetzen

Beim Bezugspunktsetzen wird der aktuellen Position ein neuer Koordinatenwert zugeordnet.

Bezugspunktsetzen ist mit Inkremental- und Absolutwertgebern möglich.

Positionieren absolut oder relativ

Easy Motion Control kann die Achse

- auf **absolute** Ziele bewegen.
- **relativ** um eine Wegstrecke in eine vorgegebene Richtung bewegen.

Die Funktionen Positionieren absolut und Positionieren relativ werden mit den Anweisungen MC_MoveAbsolute und MC_MoveRelative realisiert.

Getriebe

Easy Motion Control realisiert ein elektronisches Getriebe zwischen zwei Achsen mit der Anweisung MC_GearIn.

Stoppen einer Fahrt

Eine laufende Fahrt können Sie jederzeit stoppen.

Die Funktion Stoppen wird mit der Anweisung MC_StopMotion realisiert.

Ablösen einer Fahrt

Eine gerade aktive Fahranweisung kann jederzeit durch

- den Start einer anderen Fahranweisung abgelöst werden.
- die Anweisung MC_StopMotion abgebrochen werden (siehe Ablauf von Fahrbewegungen (Seite 2828)).

Nachführen

Beim Nachführen wird im Lageregler der Positionssollwert ständig dem Positionsistwert angepasst (nachgeführt). Die Achse wird während des Nachführens nicht auf Verlassen des Stillstandsbereichs überwacht.

Die Funktion wird durch Wegnahme des Signals "EnableDrive" an der Anweisung MC_Control oder nach Auftreten eines Fehlers aktiviert.

Geschwindigkeits-Override

Mit dem Geschwindigkeits-Override kann die parametrierte Geschwindigkeit einer Fahranweisung dynamisch verändert werden. Die Beschleunigungs- und Verzögerungswerte werden nicht beeinflusst.

Weitere Informationen zum Geschwindigkeits-Override siehe Geschwindigkeits-Override festlegen (Seite 2815).

Simulation

Im Simulationsbetrieb können Sie Ihr Verfahrprogramm testen, ohne Ihre Achse zu verfahren.

Die Simulation wird mit der Anweisung MC_Simulation realisiert.

Fehlerreaktionen und Diagnose

Alle von den Anweisungen erkannten Fehler werden im Achs-DB angezeigt durch

- Setzen eines Sammelfehlers und
- eine fehlerspezifische Anzeige.

Bei quittierpflichtigen Fehlern muss nach der Fehlerbeseitigung eine Fehlerquittung gesetzt werden.

Ist der Fehler durch eine Fahranweisung (z. B. MC_MoveJog oder MC_MoveAbsolute) verursacht oder erkannt worden, dann wird bei dieser Anweisung der Fehlerausgang "Error" gesetzt.

Wurde der Fehler aber durch eine Treiberanweisung erkannt, wird die genaue Fehlerursache in den statischen Daten der Anweisung angezeigt (siehe Eingangstreiber).

Ausführliche Informationen zur Fehlerreaktion, Fehleranzeige und Fehlerquittierung finden Sie unter Fehleranzeigen und Fehlerquittung im Achs-DB (Seite 2843).

6.2.10.3 Easy Motion Control konfigurieren

Leitfaden zum Einsatz von Easy Motion Control

Der hier beschriebene Leitfaden zeigt die grundsätzliche Vorgehensweise, um Easy Motion Control mit einer S7-300/400-CPU einzusetzen.

Voraussetzung

Um das Technologieobjekt "Achse" einzusetzen, muss ein Projekt mit einer CPU angelegt sein.

Vorgehen

Gehen Sie in der nachfolgend empfohlenen Reihenfolge vor, um Easy Motion Control mit der S7-300/400-CPU einzusetzen. Folgen Sie hierzu den aufgelisteten Links:

1. Technologieobjekt Achse hinzufügen (Seite 2798)
2. Arbeiten mit dem Konfigurationsdialog (Seite 2799)
3. Laden in CPU (Seite 2820)
4. Funktionstest der Achse im Inbetriebnahmefenster
5. Easy Motion Control programmieren (Seite 2822)
6. Achssteuerung diagnostizieren

Technologieobjekt Achse hinzufügen

Technologieobjekt Achse im Projektnavigator hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Instanz-DB zu diesem Technologieobjekt erzeugt. In diesem Instanz-DB wird die Konfiguration der Achse hinterlegt.

Voraussetzung

Ein Projekt mit einer CPU ist angelegt.

Vorgehen

Um ein Technologieobjekt Achse im Projektnavigator hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie die Technologie "Motion".
5. Öffnen Sie den Ordner "Motion Control".
6. Öffnen Sie den Ordner "Easy Motion Control".
7. Wählen Sie die Anweisung für das Technologieobjekt: hier AXIS_REF.
8. Geben Sie im Eingabefeld "Name" einen individuellen Namen für die Achse ein.
9. Wählen Sie die Option "Manuell", falls Sie die vorgeschlagene Datenbausteinnummer des Instanz-DB ändern möchten.
10. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt ergänzen möchten.
11. Klicken Sie auf die Schaltfläche "OK", wenn Sie das Technologieobjekt hinzufügen möchten.
    Klicken Sie auf die Schaltfläche "Abbrechen", wenn Sie die Eingaben verwerfen möchten.

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt. Das Technologieobjekt wird verwendet, wenn die Anweisungen zu diesem Technologieobjekt in einem Weckalarm- oder Taktsynchronalarm-OB aufgerufen wird.

Nähere Hinweise hierzu finden Sie in "Aufbau eines Anwenderprogramms (Seite 2822)".

Hinweis

Sie können im unteren Bereich des Dialogfelds das Optionskästchen "Neu hinzufügen und öffnen" aktivieren. Dadurch wird die Konfiguration des Technologieobjekts nach dem Hinzufügen geöffnet.

Technologieobjekt Achse konfigurieren

Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Projektnavigator die Gruppe des gewünschten Technologieobjekts.
2. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**
  Die Grundparameter enthalten alle Parameter, die für eine funktionsfähige Achse konfiguriert werden müssen.
- **Erweiterte Parameter**
  Die erweiterten Parameter enthalten Parameter zur Anpassung an Ihren Antrieb bzw. an Ihre Anlage.

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig. Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein voreingestellter Wert wurde geändert.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.
	Die Konfiguration ist gültig, enthält jedoch Warnhinweise Nur ein Hardware-Endschalter ist konfiguriert. Je nach Anlage kann die fehlende Konfiguration eines Hardware-Endschalters zu einer Gefährdung führen. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt.

Grundparameter

Achsnamen festlegen

Achsname

Definieren Sie in diesem Feld den Namen der Achse, bzw. den Namen des Technologieobjekts "Achse". Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

Achsart auswählen

Achsarten

Easy Motion Control kann folgende Achsarten steuern:

- Linearachse
  Die Linearachse ist eine Achse, die einen begrenzten physikalischen Verfahrbereich hat.

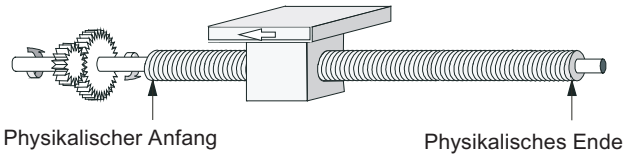

- Rundachse
  Die Rundachse ist eine Achse, deren Verfahrbereich nicht durch mechanische Anschläge begrenzt ist.

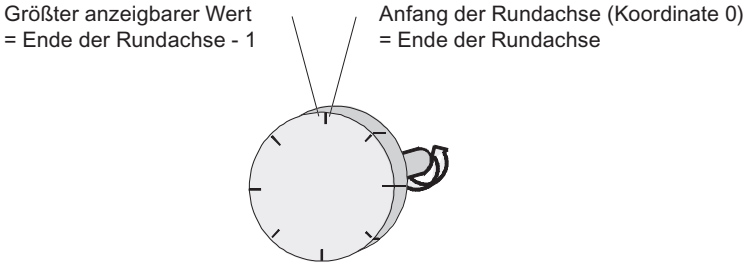

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
4.0	AxisType	BOOL	FALSE	**Achsart** Zulässige Werte: • FALSE = Linearachse • TRUE = Rundachse	I

Längeneinheit auswählen

Längeneinheit

Wählen Sie in der Klappliste die gewünschte Längeneinheit für das Maßsystem der Achse aus. Die gewählte Einheit wird für die weitere Konfiguration des Technologieobjekts "Achse" und für die Anzeige der aktuellen Achsdaten verwendet.

Sie können für die Eingabe und Ausgabe der Daten unter den folgenden Längeneinheiten wählen:

- Millimeter (Voreinstellung)
- Grad
- Inch

Die Werte an den Eingangsparametern (Position, Distance, Velocity, …) der Easy Motion Control-Anweisungen beziehen sich ebenfalls auf diese Einheit.

Hinweis

Eine Änderung der Längeneinheit hat keinen Einfluss auf die Werte der Parameter. Es findet keine Umrechnung statt.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
196.0	LengthUnit	STRING [4]	mm	**Längeneinheit** für Projektiersoftware Zulässige Werte: • mm (Voreinstellung) • Grad • Inch

Simulationsmodus aktivieren

Simulationsmodus

Aktivieren Sie das Optionskästchen, wenn Sie Ihr Verfahrprogramm testen wollen, ohne die Achse zu verfahren.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
4.2	Sim	BOOL	FALSE	**Simulationsmodus** Zulässige Werte: • FALSE = ausgeschaltet • TRUE = eingeschaltet

Erweiterte Parameter

Treiberschnittstellen

Eingangstreiber

Eingabemodul auswählen

Eingabemodul

Wählen Sie in der Klappliste das Eingabemodul aus, an dem Ihr Geber angeschlossen ist.

Passend zu Ihrer Auswahl wird im Feld "Anweisung" der Name der Anweisung angezeigt, die Sie zur Istwerterfassung aufrufen müssen.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
202.0	InputModuleType	STRING [24]	IM 174	Eingangstreiber **zum Modul**. Dieser Parameter darf nicht manuell überschrieben werden.

Geberart auswählen

Geberart

Aktivieren Sie das Optionsfeld zu dem eingesetzten Geber.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
4.1	EncoderType	BOOL	FALSE	**Geberart** Zulässige Werte: • FALSE = Inkrementalgeber • TRUE = Absolutwertgeber	I

Eingabemodul: Baugruppenadresse Eingänge

Baugruppenadresse Eingänge

Tragen Sie in das Eingabefeld den in der Hardware-Konfiguration für das Eingabemodul vergebenen Wert ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
58.0	InputModuleInAddr	INT	0	**Anfangsadresse der Eingänge der Wegerfassungsbaugruppe** Bereich: • bestimmt durch die Hardware-Konfiguration	I

Eingabemodul: Baugruppenadresse Ausgänge

Baugruppenadresse Ausgänge

Der Parameter wird nur bei Baugruppen verwendet, bei denen die "Baugruppenadresse Ausgänge" separat zur "Baugruppenadresse Eingänge" in der Hardware-Kofiguration einstellbar ist.

Tragen Sie in das Eingabefeld den in der Hardware-Konfiguration für das Eingabemodul vergebenen Wert ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
60.0	InputModuleOutAddr	INT	0	**Anfangsadresse der Ausgänge der Wegerfassungsbaugruppe** Bereich: • bestimmt durch die Hardware-Konfiguration	I

Eingabemodul: Kanalnummer

Kanalnummer

- Einkanalige Baugruppen: Das Eingabefeld ist mit "0" vorbelegt und nicht editierbar.
- Mehrkanalige Baugruppen: Tragen Sie in das Eingabefeld die Nummer des verwendeten Kanals ein. Für den ersten Kanal geben Sie "0" ein, usw.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
62.0	InputChannelNo	INT	0	**Kanalnummer der Wegerfassungsbaugruppe** Bereich: • abhängig von Baugruppe	I

Ausgangstreiber

Ausgabemodul auswählen

Ausgabemodul

Wählen Sie in der Klappliste das Ausgabemodul aus, an dem Ihr Antrieb angeschlossen ist.

Passend zu Ihrer Auswahl wird im Feld "Anweisung" der Name der Anweisung angezeigt, die Sie zur Ausgabe des Drehzahlsollwertes aufrufen müssen.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
228.0	OutputModuleType	STRING [24]	IM 174	Ausgangstreiber **zum Modul**. Dieser Parameter darf nicht manuell überschrieben werden.

Ausgabemodul: Baugruppenadresse Eingänge

Baugruppenadresse Eingänge

Der Parameter wird nur bei Baugruppen verwendet, bei denen die "Baugruppenadresse Eingänge" separat zur "Baugruppenadresse Ausgänge" in der Hardware-Konfiguration einstellbar ist.

Tragen Sie in das Eingabefeld den in der Hardware-Konfiguration für das Ausgabemodul vergebenen Wert ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
82.0	OutputModuleInAddr	INT	0	**Anfangsadresse der Eingänge der Ausgabe-baugruppe** Bereich: • bestimmt durch die Hardware-Konfiguration	I

Ausgabemodul: Baugruppenadresse Ausgänge

Baugruppenadresse Ausgänge

Tragen Sie in das Eingabefeld den in der Hardware-Konfiguration für das Ausgabemodul vergebenen Wert ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
80.0	OutputModuleOutAddr	INT	0	**Anfangsadresse der Ausgänge der Ausgabebaugruppe** Bereich: • bestimmt durch die Hardware-Konfiguration	I

Ausgabemodul: Kanalnummer

Kanalnummer

- Einkanalige Baugruppen: Das Eingabefeld ist mit "0" vorbelegt und nicht editierbar.
- Mehrkanalige Baugruppen: Tragen Sie in das Eingabefeld die Nummer des verwendeten Kanals ein. Für den ersten Kanal geben Sie "0" ein, usw.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
84.0	OutputChannelNo	INT	0	**Kanalnummer der Ausgabebaugruppe** Bereich: • abhängig von Baugruppe	I

Mechanik

Geber

Schritte pro Geberumdrehung festlegen

Schritte pro Geberumdrehung

Tragen Sie in das Eingabefeld die Anzahl der Schritte ein, die Ihre Wegerfassungsbaugruppe pro Geberumdrehung ausgibt.

Mit diesem Wert und dem Parameter "Achsweg pro Geberumdrehung" (Achs-DB Adresse 68.0 (Seite 2807)) erfolgt die Umrechnung von Schritten nach Längeneinheit.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
64.0	StepsPerRev	DINT	L#1	**Schritte pro Geberumdrehung** Zulässiger Wertebereich: • ≥ 1 [Schritte/Geberumdrehung]	I

Inkrementalgeber

Wird der Inkrementalgeber mit einer Baugruppe eingelesen, die eine 2fach- oder 4fach-Auswertung der Impulse durchführt, so muss dies bei der Eingabe der Parameter eingerechnet werden.

Beispiel:

- Geberimpulse pro Umdrehung: 500
- 4fach-Auswertung

Sie geben in das Eingabefeld "Schritte pro Geberumdrehung" ein: 2000

Absolutwertgeber

Bei Absolutwertgebern ist die Zahl der Schritte pro Geberumdrehung üblicherweise eine Zweierpotenz.

Gängige Werte sind:

- Beim Singleturngeber: 4096 und 8192
- Beim Multiturngeber (12 × 12 Bit im 25 Bit-Telegramm): 4096

Im Zweifelsfall lässt sich der richtige Wert feststellen, indem Sie den Geber um eine Umdrehung bewegen und die Änderung des Geberwertes in der Projektiersoftware beobachten.

Bei einem Linearmaßstab geben Sie für "Schritte pro Geberumdrehung" die Gesamtzahl der Schritte ein, die der Länge Ihres Linearmaßstabs entspricht.

Anzahl Geberumdrehungen festlegen

Anzahl Geberumdrehungen

Tragen Sie in das Eingabefeld die Anzahl der Geberumdrehungen bei einem Absolutwertgeber ein.

Bei einem Singleturn-Geber geben Sie hier "1" ein.

Bei einem **Linearmaßstab** geben Sie für "Anzahl Geberumdrehungen" eine "1" ein.

Bei einem Inkrementalgeber ist dieser Parameter ohne Bedeutung.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
72.0	NumberRevs	INT	1	**Anzahl Geberumdrehungen** eines Absolutwertgebers Zulässiger Wertebereich: • > 0	I

Achsweg pro Geberumdrehung festlegen

Achsweg pro Geberumdrehung

Tragen Sie in das Eingabefeld den Weg ein, den die Achse bei einer Umdrehung des Gebers zurücklegt. Der Wert ist abhängig vom Aufbau der Achse und von der Art, wie der Geber montiert ist. Sie müssen alle Übertragungsglieder wie Kupplungen oder Getriebe berücksichtigen.

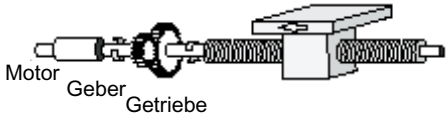

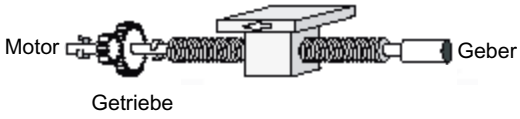

Bei einem **Linearmaßstab** geben Sie für "Achsweg pro Geberumdrehung" die Länge Ihres Linearmaßstabs ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
68.0	DisplacementPerRev	REAL	1.0	**Achsweg pro Geberumdrehung** Zulässiger Wertebereich: • > 0.0 [Längeneinheit/Geberumdrehung]	I

Besonderheiten bei Rundachsen

Bei einer Rundachsenumdrehung muss sich der Geber um eine ganzzahlige Anzahl von Schritten bewegen:

$$\frac{\text{(Ende der Rundachse - Anfang der Rundachse)}}{\text{Achsweg pro Geberumdrehung}} \times \text{Schritte pro Geberumdrehung} = \text{eine ganze Zahl}$$

Besonderheiten bei Rundachsen mit Absolutwertgebern

Um die Reproduzierbarkeit jeder Position zu gewährleisten, muss der vom Geber abgedeckte Achsweg ein ganzzahliges Vielfaches des Weges sein, den die Rundachse bei einer Umdrehung zurücklegt:

$$\frac{\text{(Anzahl Geberumdrehungen) x (Achsweg pro Geberumdrehung)}}{\text{(Ende der Rundachse - Anfang der Rundachse)}} = \text{eine ganze Zahl}$$

Beispiel:

Absolutwertgeber 24 Bit

=> Anzahl Geberumdrehungen = 4096

Anfang der Rundachse = 0.0°

Ende der Rundachse = 360.0°

=> Ende der Rundachse - Anfang der Rundachse = 360.0°

Achsweg pro Geberumdrehung = 45°

$$\frac{4096 \times 45°}{360°} = 512 \text{ (ganzzahlig)}$$

Richtungsanpassung Geber auswählen

Richtungsanpassung Geber

Hier passen Sie die Richtung der Wegerfassung an die Bewegungsrichtung der Achse an:

- positiv = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen aufsteigenden Lageistwerten.
- negativ = aufsteigende Zählimpulse (Inkrementalgeber) bzw. Geberwerte (Absolutwertgeber) entsprechen fallenden Lageistwerten.

Wählen Sie in der Klappliste die entsprechende Option aus.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
74.0	PolarityEncoder	INT	1	**Richtungsanpassung Geber** Zulässige Werte: • +1 oder -1	I

Regler

Reglerverstärkung festlegen

Reglerverstärkung

Die optimale Reglerverstärkung kann an der Achse experimentell ermittelt werden.

Erhöhen Sie die Reglerverstärkung in Schritten von 1.0, bis die Achse beim Fahren oder im Stillstand zu schwingen beginnt. Wenn dies der Fall ist, verringern Sie die Reglerverstärkung, bis keine Schwingungsneigung mehr sichtbar ist.

Tragen Sie in das Eingabefeld den entsprechenden Wert ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
44.0	FactorP	REAL	1.0	**Reglerverstärkung** Zulässiger Wertebereich: • > 0.0 [1/s]

Sollgeschwindigkeit im Handbetrieb festlegen

Sollgeschwindigkeit im Handbetrieb

Tragen Sie in das Eingabefeld die gewünschte Sollgeschwindigkeit für den Handbetrieb ein.

Bei eingeschaltetem Handbetrieb gibt der Regler den Wert von "Sollgeschwindigkeit im Handbetrieb" als Stellgröße aus und begrenzt ihn auf "MaxVelocity". Der Lagesollwert wird dem Lageistwert nachgeführt.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
48.0	ManVelocity	REAL	0.0	**Sollgeschwindigkeit im Handbetrieb** Zulässiger Wertebereich: • ≤ ± MaxVelocity (Seite 2814) [Längeneinheit/s]
52.0	ManEnable	BOOL	FALSE	**Handbetrieb einschalten** Zulässige Werte: • FALSE = ausgeschaltet • TRUE = eingeschaltet

Motor

Bezugswert für 100% Drehzahl festlegen

Bezugswert für 100% Drehzahl

Wenn Sie Leistungsteile verwenden, die mit ±10 V angesteuert werden, ist dieser Parameter nicht relevant.

Wenn Sie den Ausgangstreiber OutputMM4_DP (busgesteuerte Baugruppe MICROMASTER DP) verwenden, geben Sie in diesem Parameter die Frequenz ein, mit dem der Antrieb angesteuert werden muss, um 100 % seiner Drehzahl zu erreichen.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
96.0	DriveInputAt100	REAL	10.0	**Bezugswert für 100% Drehzahl** Zulässiger Wertebereich: • -10.0 bis +10.0 [V]	I

Beispiel

- Nenndrehzahl des Motors (100% Drehzahl): = 1380 U/min
- Motor ist abgeglichen auf: = 1380 U/min bei 50 Hz

=> Geben Sie ein für "Bezugswert für 100% Drehzahl" = 50 Hz

Bezugswert für maximale Achsgeschwindigkeit festlegen

Bezugswert für maximale Achsgeschwindigkeit

Wenn Sie Leistungsteile verwenden, die mit **±10 V** angesteuert werden, geben Sie in das Eingabefeld den Wert der Spannung ein, bei welcher die "Maximale Geschwindigkeit" des Antriebs erreicht wird (siehe Achs-DB Adresse 14.0 (Seite 2814)).

Beispiel:

- Nenndrehzahl des Motors: = 3000 U/min
- Motor ist abgeglichen auf: = 9 V bei 3000 U/min
- Sie wollen maximal fahren: = 1500 U/min

=> Geben Sie ein für "Bezugswert für maximale Achsgeschwindigkeit": = 4.5 V

Wenn Sie den Ausgangstreiber **OutputMM4_DP** verwenden, geben Sie in das Eingabefeld den Wert ein, mit dem der Antrieb angesteuert werden muss, um die "Maximale Geschwindigkeit" zu erreichen (siehe Achs-DB Adresse 14.0 (Seite 2814)).

Beispiel:

- Nenndrehzahl des Motors: = 1380 U/min
- Motor ist abgeglichen auf: = 1380 U/min bei 50 Hz
- Sie wollen maximal fahren: = 690 U/min

=> Geben Sie ein für "Bezugswert für maximale Achsgeschwindigkeit": = 25 Hz

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
88.0	DriveInputAtMaxVel	REAL	9.0	**Bezugswert für maximale Achsgeschwindigkeit [V]** Zulässiger Wertebereich: • Abhängig vom Leistungsteil	I

Nullpunktkompensation festlegen

Nullpunktkompensation

Mit diesem Parameter können Sie eine Nullpunktkompensation realisieren, falls Ihr Antrieb diese Möglichkeit nicht bietet. Der hier parametrierte Wert wird auf den Ausgabewert addiert.

Der Parameter wird beim Ausgangstreiber OutputMM4_DP nicht berücksichtigt.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
92.0	OffsetCompensation	REAL	0.0	**Nullpunktkompensation** Zulässiger Wertebereich: • -10.0 bis +10.0 [V]

Richtungsanpassung Antrieb auswählen

Richtungsanpassung Antrieb

Hier wählen Sie die Richtung der Antriebsbewegung:

- positiv = positive Geschwindigkeitssollwerte bewirken eine Bewegung in positiver Richtung.
- negativ = positive Geschwindigkeitssollwerte bewirken eine Bewegung in negativer Richtung.

Wählen Sie in der Klappliste die entsprechende Option aus.

Wenn Sie die "Richtungsanpassung Antrieb" ändern, müssen Sie auch die "Richtungsanpassung Geber" (Achs-DB Adresse 74.0 (Seite 2808)) ändern.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
86.0	PolarityDrive	INT	1	**Schritte pro Geberumdrehung** Zulässige Werte: • +1 oder -1	I

Endschalter

Software-Endschalter aktivieren

Software-Endschalter

Software-Endschalter gibt es nur bei Linearachsen.

Die Überwachung beginnt, sobald die Achse synchronisiert und die Überwachung der Software-Endschalter eingeschaltet ist.

Bei ausgeschalteter Überwachung der Software-Endschalter haben die Parameter "Software-Endschalter Anfang" und "Software-Endschalter Ende" (Achs-DB Adresse 6.0 und 10.0 (Seite 2812)) keine Bedeutung.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
42.1	SWLimitEnable	BOOL	TRUE	**Software-Endschalter überwachen** Zulässige Werte: • TRUE = Überwachung eingeschaltet • FALSE = Überwachung ausgeschaltet

Software-Endschalter festlegen

Software-Endschalter

Software-Endschalter gibt es nur bei Linearachsen.

Bei Linearachse: Software-Endschalter Anfang/Ende

Die Überwachung beginnt, sobald die Achse synchronisiert und die Überwachung der Software-Endschalter eingeschaltet ist (siehe "Software-Endschalter aktivieren (Seite 2812)"). Der Arbeitsbereich wird durch die Software-Endschalter begrenzt.

Der "Software-Endschalter Anfang" muss immer kleiner sein als der "Software-Endschalter Ende".

Inkrementalgeber

Nach jedem Anlauf der CPU ist die Achse zunächst nicht synchronisiert. Erst nach Referenzpunktfahrt bzw. Bezugspunktsetzen ist die Achse synchronisiert und die Software-Endschalter werden überwacht.

Absolutwertgeber

Ein einmal durchgeführtes Bezugspunktsetzen geht mit dem nächsten CPU-Anlauf nicht verloren. Die Software-Endschalter können überwacht werden.
Der Geberbereich des Absolutwertgebers muss mindestens den Arbeitsbereich abdecken.

Bei Rundachse: Rundachsenanfang / Rundachsenende

Mit "Rundachsenanfang" und "Rundachsenende" bestimmen Sie, in welchem Wertebereich die Achspositionen während einer Achsumdrehung angezeigt werden.

Der Wert "Ende der Rundachse" ist der theoretisch größte Wert, den der Istwert erreichen kann:

$$\text{größter anzeigbarer Wert} = \text{Ende der Rundachse} - \frac{\text{Weg pro Geberumdrehung}}{\text{Inkremente pro Geberumdrehung}}$$

Der theoretisch höchste Wert wird allerdings nie angezeigt, weil er physikalisch die gleiche Position kennzeichnet wie der Anfang der Rundachse.

Beispiel:

- AxisLimitMin = 0.0°
- AxisLimitMax = 360.0°

In positiver Richtung zählt der Istwert ...359.8, 359.9, 0.0, 0.1, …

Beachten Sie bei Verwendung von Absolutwertgebern den Parameter "Achsweg pro Geberumdrehung (Seite 2807)".

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
6.0	AxisLimitMax	REAL	$1.0e^{+6}$	Bei Linearachse: **Software-Endschalter Ende** Bei Rundachse: **Ende der Rundachse** Zulässiger Wertebereich: • $-2^{24}+6 \leq$ AxisLimitMin < AxisLimitMax $\leq +2^{24}-6$ [Längeneinheit]	I
10.0	AxisLimitMin	REAL	$-1.0e^{+6}$	Bei Linearachse: **Software-Endschalter Anfang** Bei Rundachse: **Anfang der Rundachse** Zulässiger Wertebereich: • $-2^{24}+6 \leq$ AxisLimitMin < AxisLimitMax $\leq +2^{24}-6$ [Längeneinheit]	I

Dynamik

Allgemein

Maximale Achsgeschwindigkeit festlegen

Maximale Achsgeschwindigkeit

Hier parametrieren Sie die Geschwindigkeit, die die Achse erreicht, wenn der Motor mit dem im Parameter "Bezugswert für maximale Achsgeschwindigkeit" (Achs-DB Adresse 88.0 (Seite 2810)) eingegebenen Wert angesteuert wird.

Die "Maximale Achsgeschwindigkeit" dient sowohl als Faktor bei der Berechnung des Ausgangssignals zum Antrieb als auch zur Begrenzung der Achsgeschwindigkeit.

Besonderheiten bei Rundachsen mit Absolutwertgebern

Damit jede Rundachsenumdrehung erfasst werden kann, darf die maximale Achsgeschwindigkeit nur so groß sein, dass pro Abtastzeit ein Weg verfahren wird, der kleiner ist als der halbe Geberbereich des Absolutwertgebers.

Beispiel für Singleturn-Geber:

Voraussetzung => eine Achsumdrehung entspricht einer Geberumdrehung.

Verfahrweg / Geberbereich = 360.0°

Verfahrweg / halber Geberbereich = 180.0°

Abtastzeit = 0.01 s

$$\Rightarrow \text{„Maximale Geschwindigkeit“} - \frac{\text{Verfahrweg / halber Geberbereich}}{\text{Abtastzeit}}$$

=> "Maximale Geschwindigkeit" < 180.0° / 0.01s = 18000°/s entspricht 50 Achsumdrehungen/s

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
14.0	MaxVelocity	REAL	0.1	**Maximale Achsgeschwindigkeit** Zulässiger Wertebereich: • > 0.0 [Längeneinheit/s]	I

Geschwindigkeits-Override festlegen

Geschwindigkeits-Override

Der "Geschwindigkeits-Override" wirkt ständig auf die parametrierte Geschwindigkeit einer Fahranweisung.
Die Beschleunigungs- und Verzögerungswerte werden nicht beeinflusst.

Beispiel: Fahrkurven mit Override von 50 % und 100 %

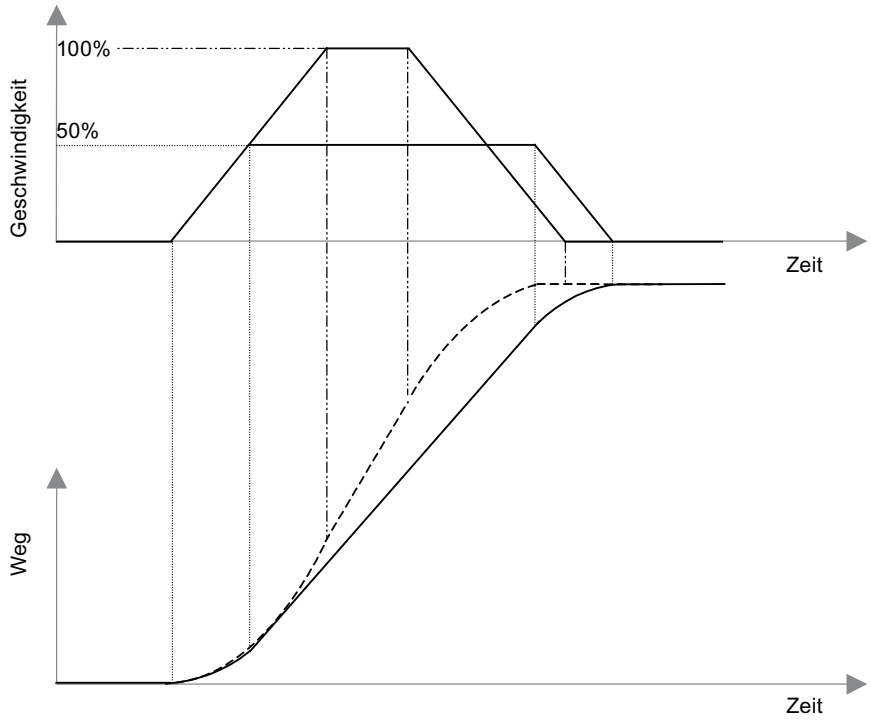

Beim Umschalten von 50 % auf 100 % wird die Geschwindigkeit verdoppelt. Weil Beschleunigung und Verzögerung durch den Parameter "Geschwindigkeits-Override" jedoch nicht beeinflusst werden, wird die Positionierzeit **nicht** halbiert.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
100.0	Override	REAL	100.0	**Geschwindigkeits-Override** Zulässiger Wertebereich: • 0.0 bis +100.0 [%]

Maximale Achsbeschleunigung festlegen

Maximale Achsbeschleunigung

Die Steilheit der Beschleunigungsrampe der Achse wird im geregelten Betrieb auf diesen Wert begrenzt.

Die maximal mögliche Beschleunigung kann experimentell ermittelt werden.

Sie sollte nur so groß sein, dass beim Beschleunigen auf maximale Geschwindigkeit (mit Last) die Stromgrenze des Antriebs gerade nicht erreicht wird.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
18.0	MaxAcceleration	REAL	0.1	**Maximale Achsbeschleunigung** Zulässiger Wertebereich: • > 0.0 [Längeneinheit/s^2]	I

Maximale Achsverzögerung festlegen

Maximale Achsverzögerung

Die Steilheit der Bremsrampe der Achse wird im geregelten Betrieb auf diesen Wert begrenzt.

Die maximal mögliche Verzögerung kann experimentell ermittelt werden.

Sie sollte nur so groß sein, dass beim Abbremsen von maximaler Geschwindigkeit (mit Last) die Stromgrenze des Antriebs gerade nicht erreicht wird.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
22.0	MaxDeceleration	REAL	0.1	**Maximale Achsverzögerung** Zulässiger Wertebereich: • > 0.0 [Längeneinheit/s^2]	I

Abtastzeit festlegen

Abtastzeit

Für die korrekte Funktion einer Lageregelung ist es wichtig, dass alle Funktionen einer Achse in einem festen Zeitraster bearbeitet werden. Dieses Zeitraster wird dadurch realisiert, dass Sie alle Anweisungen einer Achse im selben Weckalarm- bzw. Taktsynchronalarm-OB aufrufen. Das Zeitraster dieses OB wählen Sie in der Hardware-Konfiguration der CPU aus.

Als "Abtastzeit" geben Sie das Zeitraster des OB ein, in dem Sie alle Anweisungen einer Achse aufrufen.

Beispiel:

Sie rufen das Programm mit den Anweisungen von Easy Motion Control im OB 35 auf. In der Hardware-Konfiguration haben Sie das Weckalarmzeitraster des OB 35 auf 10 ms eingestellt.

Als "Abtastzeit" geben Sie 0.010 ein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
0.0	Sample_T	REAL	0.01	**Abtastzeit** der Achsanweisungen Vorschlag: • Bei S7-300 bis CPU 316 ≥ 0.020 [s] • Bei S7-400 und ab CPU 317 ≥ 0.004 [s]	I

Hochlaufzeit anzeigen

Hochlaufzeit

Die Hochlaufzeit ist die Zeit, während der die Achse vom Stillstand auf die Maximalgeschwindigkeit beschleunigt wird.

Die Hochlaufzeit berechnet sich aus folgenden Parametern:

Hochlauf-zeit = Maximale Achsgeschwindigkeit × Geschwindigkeits-Override / Maximale Beschleunigung

In dem Ausgabefeld wird die berechnete Hochlaufzeit angezeigt.

Rücklaufzeit anzeigen

Rücklaufzeit

Die Rücklaufzeit ist die Zeit, während der die Achse ohne Ruckbegrenzung von der Maximalgeschwindigkeit auf Stillstand verzögert wird.

Die Rücklaufzeit berechnet sich aus folgenden Parametern:

Rücklauf-zeit = Maximale Achsgeschwindigkeit × Geschwindigkeits-Override / Maximale Verzögerung

In dem Ausgabefeld wird die berechnete Rücklaufzeit angezeigt.

Notstopp

Verzögerung für Notstopp festlegen

Verzögerung für Notstopp (harter Stopp)

Hier parametrieren Sie die Steilheit der zeitgesteuerten Bremsrampe, mit welcher der Antrieb abgebremst werden soll, wenn ein Fehler mit hartem Stopp auftritt.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
54.0	EmergencyDec	REAL	1.0	**Verzögerung für Notstopp** Zulässiger Wertebereich: • > 0.0 [Längeneinheit/s^2]	I

Rücklaufzeit anzeigen

Rücklaufzeit

Die Rücklaufzeit ist die Zeit, während der die Achse im Notstopp-Betrieb von der Maximalgeschwindigkeit auf Stillstand verzögert wird.

Die Rücklaufzeit berechnet sich aus folgenden Parametern:

Rücklauf- = Maximale Achsgeschwindigkeit / Notstopp-Verzögerung
zeit

In dem Ausgabefeld wird die berechnete Rücklaufzeit angezeigt.

Überwachungen

Zielbereich festlegen

Zielbereich

Der "Zielbereich" liegt links und rechts um das Ziel (siehe Begriffsdefinitionen (Seite 2791)).

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
30.0	TargetRange	REAL	1.0	**Zielbereich** Zulässiger Wertebereich: • > 0.0 [Längeneinheit]	I

Stillstandsbereich festlegen

Stillstandsbereich

Im Stillstand wird überwacht, ob die Achse auf einer Position stehenbleibt oder von ihr wegdriftet. Sobald der "Stillstandsbereich" ohne einen aktiven Fahrauftrag verlassen wird, wird im Achs-DB das Fehlerbit "Stillstandsbereich verlassen" (Achs-DB Adresse 130.3 (Seite 2848)) gesetzt. Der "Stillstandsbereich" liegt links und rechts um das Ziel (siehe Begriffsdefinitionen (Seite 2791)).

Empfehlung: Der "Stillstandsbereich" sollte größer als der "Zielbereich" (Achs-DB Adresse 30.0 (Seite 2818)) sein.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
34.0	StandstillRange	REAL	1.5	**Stillstandsbereich** Zulässiger Wertebereich: • > 0.0 [Längeneinheit]	I

Zieleinlauf überwachen

Überwachung des Zieleinlaufs

Bei eingeschalteter Überwachung des Zieleinlaufs ist eine Fahrt beendet, sobald der Lageistwert innerhalb der "Überwachungszeit für Zieleinlauf" (Achs-DB Adresse 26.0) den "Zielbereich" (Achs-DB Adresse 30.0) erreicht hat.

Bei ausgeschalteter Überwachung des Zieleinlaufs ist eine Fahrt beendet, sobald der Lagesollwert das Ziel erreicht hat (siehe Begriffsdefinitionen (Seite 2791)).

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar
42.0	MonitorTargetAppr	BOOL	TRUE	**Zieleinlauf überwachen** Zulässige Werte: • TRUE = Überwachung eingeschaltet • FALSE = Überwachung ausgeschaltet

Überwachungszeit für Zieleinlauf festlegen

Überwachungszeit für Zieleinlauf

Nachdem bei einer Fahrt der **Lagesollwert** im Ziel angekommen ist, startet die "Überwachungszeit für den Zieleinlauf", wenn das Bit "Zieleinlauf überwachen (Achs-DB Adresse 42.0 (Seite 2819)) gesetzt ist. Wenn der **Lageistwert** nach Ablauf dieser Zeit den im Parameter "Zielbereich" (Achs-DB Adresse 30.0 (Seite 2818)) definierten Bereich nicht erreicht hat, so wird im Achs-DB das Fehlerbit "Fehler bei Zieleinlauf" (Achs-DB Adresse 130.4 (Seite 2848)) gesetzt.

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
26.0	MonTimeTargetAppr	REAL	1.0	**Überwachungszeit für Zieleinlauf** Zulässiger Wertebereich: • > 2 × Sample_T [s] (Abtastzeit (Seite 2816))	I

Maximal zulässigen Schleppabstand festlegen

Maximal zulässiger Schleppabstand

Mit der Schleppabstandsüberwachung wird geprüft, ob der Lageistwert während einer Fahrt unzulässig weit hinter dem Lagesollwert zurückbleibt. Wird der maximale Schleppabstand überschritten, so wird im Achs-DB das Fehlerbit "Schleppabstand überschritten" (Achs-DB Adresse 130.2 (Seite 2848)) gesetzt.

Vorschlag für die Einstellung: ("Maximale Geschwindigkeit" / "Reglerverstärkung") × 1.1 (siehe Achs-DB Adresse 14.0 (Seite 2814) und 44.0 (Seite 2809))

Verwendete Daten im Achs-DB

Adresse	Name	Typ	Anfangswert	Kommentar	
38.0	MaxFollowingDist	REAL	5.0	**Maximal zulässiger Schleppabstand** Zulässiger Wertebereich: • > 0.0 [Längeneinheit]	I

6.2.10.4 Laden in CPU

Neue oder geänderte Projektdaten müssen in die CPU geladen werden.

Laden der Programmbausteine und der Konfiguration des Technologieobjekts in die CPU

Gehen Sie zum Laden wie folgt vor:

1. Markieren Sie in der Projektnavigation das Objekt "Programmbausteine" oder das Objekt "Technologieobjekte".
2. Wählen Sie den Befehl "Online > Laden in Gerät".

Hinweis

Beim Laden von Bausteinen in die CPU stellt das TIA-Portal stets sicher, dass nach dem Ladevorgang die Offline-Bausteine im Projekt und die Online-Bausteine in der CPU zueinander konsistent sind.

Unabhängig von der Markierung in der Projektnavigation werden immer alle geänderten und neuen Offline-Bausteine und Technologieobjekte in die CPU geladen.

Bei jedem Ladevorgang werden alle Aktualwerte der Datenbausteine auf Initialwerte zurück gesetzt.

Laden der Hardware-Konfiguration in die CPU

Gehen Sie zum Laden der Hardware-Konfiguration wie folgt vor:

1. Markieren Sie in der Projektnavigation das Objekt der CPU.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Hardwarekonfiguration".

Laden der Hardware-Konfiguration, der Programmbausteine und der Konfiguration des Technologieobjekts in die CPU

Gehen Sie zum Laden wie folgt vor:

1. Markieren Sie in der Projektnavigation das Objekt der CPU.
2. Wählen Sie den Menübefehl "Online > Laden in Gerät".

6.2.10.5 Easy Motion Control programmieren

Aufbau eines Anwenderprogramms

Zusammenwirken der Anweisungen

Das folgende Bild zeigt für Easy Motion Control das Zusammenwirken der Anweisungen.

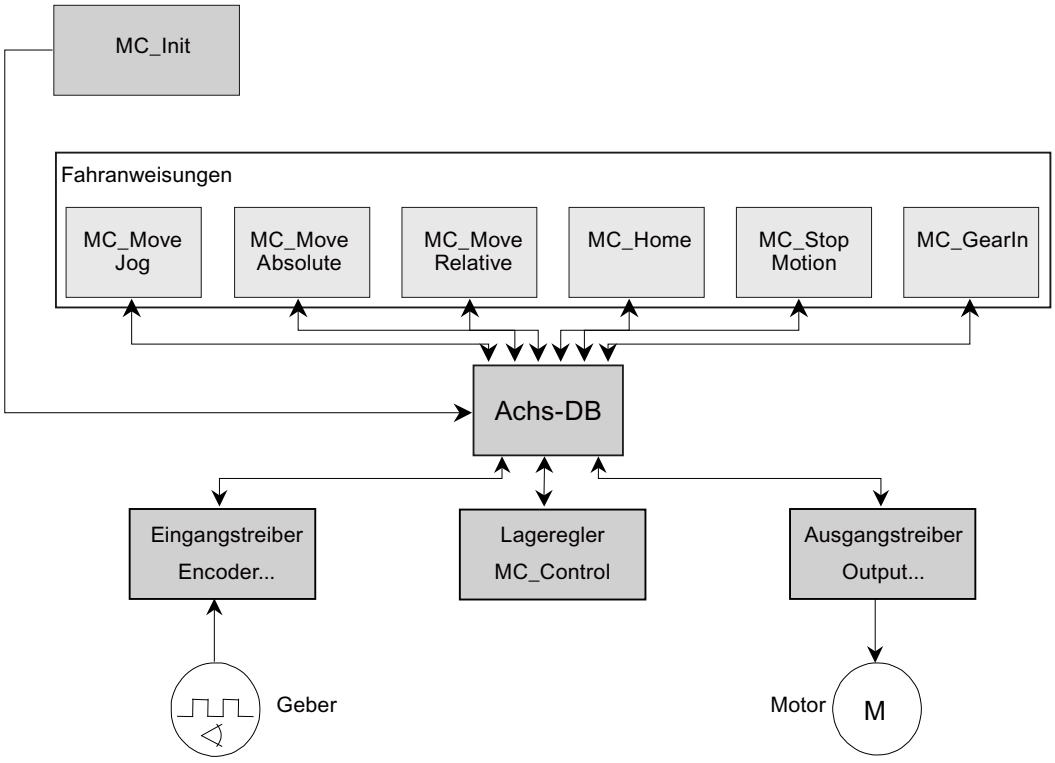

Die Anweisung MC_Init dient zur **Initialisierung** der Anweisungen.

Der **Eingangstreiber** (Anweisung Encoder…) liest den Positionsistwert einer Achse aus der Peripheriebaugruppe ein und normiert diesen in die von Ihnen gewählte Längeneinheit.

Eine der **Fahranweisungen** erzeugt den Positionssollwert.

Der **Lageregler** (Anweisung MC_Control) vergleicht den Positionsistwert mit dem Positionssollwert und erzeugt aus der Differenz einen Geschwindigkeitssollwert.

Der **Ausgangstreiber** (Anweisung Output…) normiert den Geschwindigkeitssollwert in das passende Format und schreibt ihn zur angeschlossenen Peripheriebaugruppe.

Das Programm für einen **geschlossenen Regelkreis** besteht aus einem Eingangstreiber (Encoder…), mindestens einer Fahranweisung, dem Regler (MC_Control) und einem Ausgangstreiber (Output…).

Alle Anweisungen einer Achse arbeiten mit einem **Achs-DB** zusammen. Dieser enthält alle Achsdaten, die zum Betreiben der Achse notwendig sind, und ist für jede Achse vorhanden. Beachten Sie dazu die Hinweise zum Laden des Achs-DB (Seite 2832).

Aufrufumgebung

Für die korrekte Funktion einer Lageregelung ist es wichtig, dass alle Funktionen einer Achse in einem festen **Zeitraster** bearbeitet werden. Dieses Zeitraster wird dadurch bestimmt, dass Sie die Anweisungen von Easy Motion Control in **einer** Ablaufebene aufrufen. Dafür kommen je nach eingesetzter CPU und je nach Art der Peripherieanbindung zwei Verfahren in Betracht:

Weckalarm-OB (OB 30 bis OB 38)

Wird verwendet bei zentraler Anbindung der Peripherie oder bei dezentraler Anbindung über einen nicht-äquidistanten Peripheriebus. Die Weckalarmzeit entspricht dem Lageregeltakt.

Taktsynchronalarm-OB (OB 61 bis OB 64)

Bei der Peripherieanbindung über einen durchgängig taktsynchronen Peripheriebus entspricht der Lageregeltakt dem Wert, den Sie bei der Busparametrierung als "Äquidistanten Zyklus" gewählt haben.

In beiden Fällen müssen Sie diesen Wert in den Achs-DB eintragen.

Allgemeine Hinweise zur Taktsynchronität finden Sie in der TIA-Portal Online-Hilfe sowie im Funktionshandbuch Taktsynchronität (http://support.automation.siemens.com/WW/view/de/15218045).

Aus Sicht von Easy Motion Control sollten Sie folgende Besonderheiten beachten:

- Bei der Projektierung des Peripheriebus und der DP-Slaves / IO-Devices (evtl. auch deren Module) müssen Sie die "Äquidistanz" wählen und parametrieren.
- Die zur Lageerfassung bzw. zur Sollwertausgabe benutzten Peripherieadressen einer Achse müssen in ein und demselben Teilprozessabbild (TPA) liegen. Bei Getriebeachsen müssen beide Achsen im selben TPA liegen.
- Bei der Projektierung der CPU wird dieses TPA dem benutzten Taktsynchronalarm zugeordnet.
- Dieses TPA müssen Sie im Alarm-OB vor/nach dem Aufruf der Anweisungen von Easy Motion Control mit den Systemanweisungen "SYNC_PI" / "SYNC_PO" aktualisieren.
- Werten Sie in Ihrem Programm die Rückmeldungen dieser Systemanweisungen aus und reagieren Sie darauf. (Im Fehlerfall konnte das TPA eventuell nicht aktualisiert werden und damit könnte der gelesene Istwert falsch sein.)

Empfohlene Aufrufreihenfolge

Das folgende Bild zeigt die empfohlene Aufrufreihenfolge für die Anweisungen von Easy Motion Control.

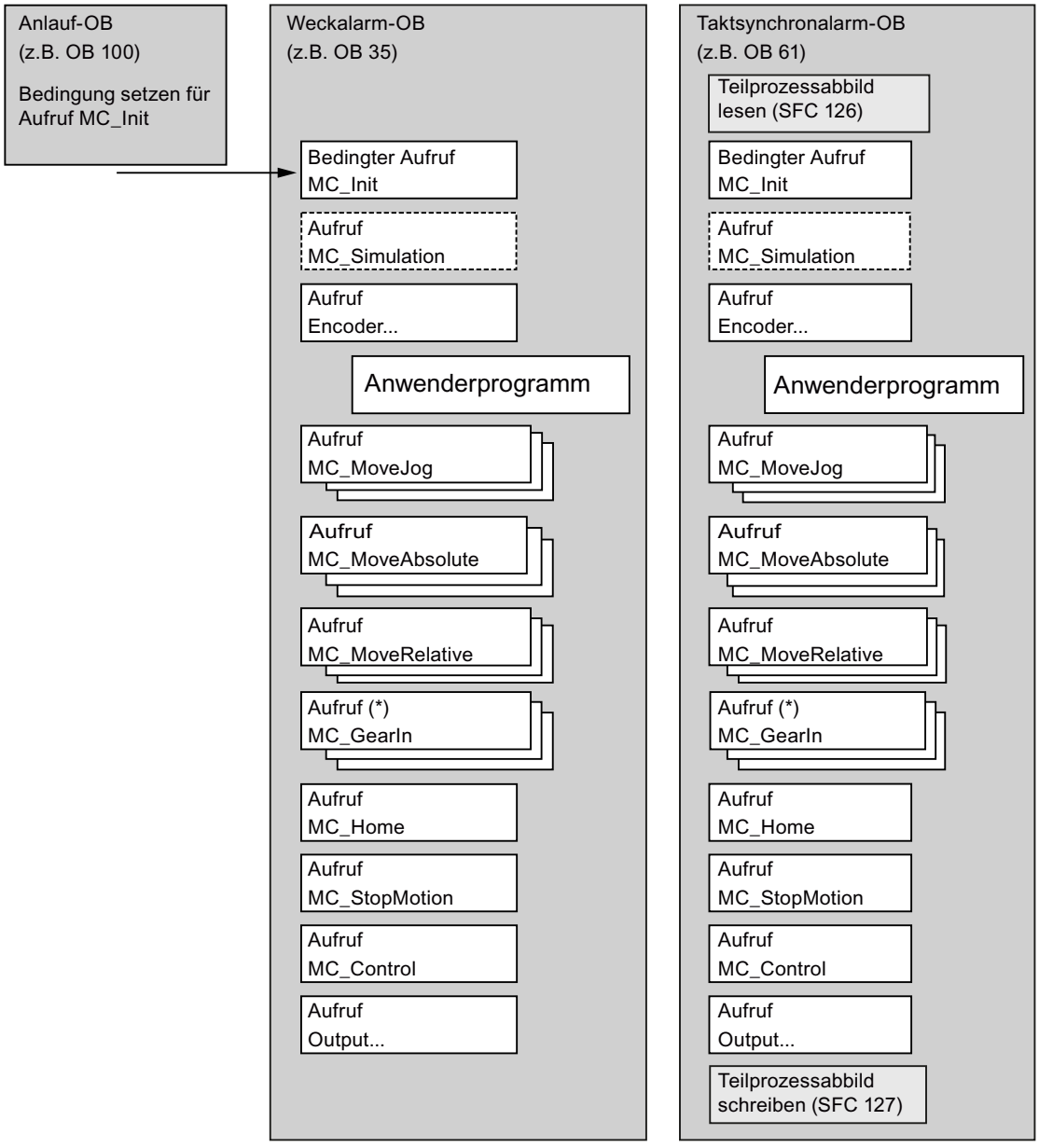

(*) Der Aufruf der Getriebeanweisung MC_GearIn ist nur bei Folgeachsen sinnvoll.

Mehrere Achsen

Für den Betrieb mehrerer Achsen in einer CPU werden für das Anwenderprogramm folgende Strukturen empfohlen:

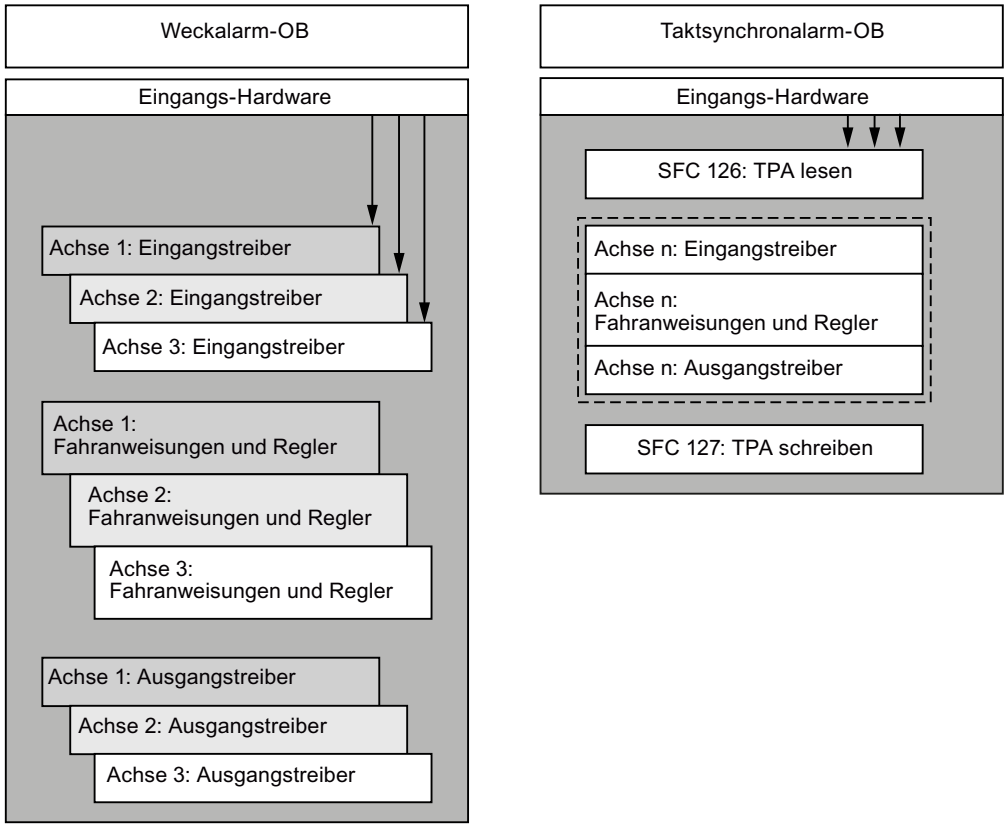

Empfohlen für Weckalarm-OB.	Empfohlen für Taktsynchronalarm-OB.
Vorteil: Aufruf der Eingangstreiber am Anfang des OB bedeutet minimalen Zeitversatz und minimalen Jitter beim Lesen der Peripherie.	**Vorteil:** Zusammenfassen *aller* Anweisungen einer Achse in *einerm* Multiinstanz-Anweisung ist möglich.
Nachteil: Zusammenfassen *aller* Anweisungen einer Achse in *einer* Multiinstanz-Anweisung nicht möglich.	Wegen des bei Taktsynchronalarm-OB notwendigen TPA sind die gelesenen Peripheriedaten unabhängig vom Treiberaufruf konsistent.

Alle Achsen, die untereinander in Beziehung stehen, müssen im selben OB aufgerufen werden.

Bei einer taktsynchronen Anwendung müssen **alle** beteiligten Achsen im selben Teilprozessabbild liegen.

Handhabung der Instanzdaten

Die **Instanzdaten** können Sie für jede Anweisung in einem eigenen DB ablegen oder für alle Anweisungen in einem Multiinstanz-DB.

Nach dem Laden der Instanzdaten einer Anweisung in die Baugruppe ist eine Initialisierung der Anweisung erforderlich. Dazu muss der Eingang "Init" der Anweisung gesetzt werden. Siehe auch Initialisierung und Parameteränderungen (Seite 2827) sowie Speichern und Laden des Achs-Datenbausteins (Seite 2832).

Verschalten der Anweisungen

Die oben im Bild unter Zusammenwirken der Anweisungen dargestellten **Verbindungen** zwischen den Anweisungen und dem Achs-DB erfolgen über diese Parameter:

- Axis
- Init

(siehe Bedeutung der gemeinsamen Parameter (Seite 2827))

Weiterhin sind folgende Verschaltungen notwendig:

- Der Eingang "EnableDrive" der Anweisung MC_Control muss mit einem Freigabeschalter verschaltet werden. Nur wenn "EnableDrive" = TRUE ist, gibt die Anweisung MC_Control einen Drehzahlsollwert aus.
- Der Eingang "EnableDrive" der Anweisung Output… muss mit dem Ausgang "DriveEnabled" der Anweisung MC_Control verschaltet werden. Nur wenn "EnableDrive" = TRUE ist, gibt die Anweisung Output… einen Wert ungleich Null aus.
- Den Ausgang "DriveEnabled" der Anweisung MC_Control sollten Sie mit einem Freigabeeingang Ihres Antriebs verschalten.

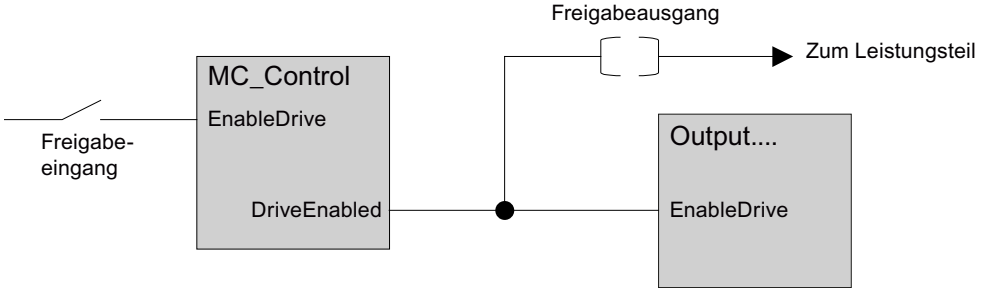

Einflüsse auf die Lageregelung in der CPU

Empfehlungen für eine optimale Lageregelung:

- Die Abtastzeit muss so exakt wie möglich eingehalten werden.
  - Parametrieren Sie in der Hardware-Konfiguration für die CPU eine möglichst hohe Priorität für den Weckalarm-OB, in dem Sie die Anweisungen aufrufen.
  - Parametrieren Sie in der Hardware-Konfiguration für die CPU eine möglichst geringe Zyklusbelastung durch Kommunikation.
  - Verwenden Sie eine taktsynchrone Umgebung; hier sind keine weiteren Maßnahmen erforderlich
- Stellen Sie bei Anbindung der Geberwerterfassung über dezentrale Peripherie (PROFIBUS DP / PROFINET IO) eine möglichst hohe Übertragungsgeschwindigkeit des Peripheriebus ein (z. B. 12 Mbit/s).

Bedeutung der gemeinsamen Parameter

Die folgende Tabelle zeigt die Parameter, die in allen Anweisungen benutzt werden.

Parameter	Deklaration	Datentyp	Beschreibung
EN	INPUT	BOOL	Freigabeeingang der KOP-/FUP-Box
ENO	OUTPUT	BOOL	Freigabeausgang der KOP-/FUP-Box
Axis	IN-OUT	STRUCT	Verschalten Sie diesen Parameter mit dem Achs-DB. Bei einem DB vom Typ AXIS_REF lautet die Verschaltung: • Axis := DBname.Ax Bei einem DB, der eine Variable vom Typ AXIS_REF enthält, lautet die Verschaltung: • Axis := DBname.Variablenname.Ax
Init	IN-OUT	BOOL	Verschalten Sie diesen Parameter mit einem Initialisierungsbit im Achs-DB, das von keiner anderen Anweisung benutzt wird ("Init.I0" … "Init.I31", siehe Bitfeld für die Initialisierungsfunktion (Seite 2835)). Die Anweisung führt bei Init = TRUE ihre Initialisierung durch und setzt anschließend das Bit zurück. Bei einem DB vom Typ AXIS_REF lautet die Verschaltung: • Init := DBname.Ax.Init.Ix Bei einem DB, der eine Variable vom Typ AXIS_REF enthält, lautet die Verschaltung: • Init := DBname.Variablenname.Ax.Init.Ix mit Ix = I0 … I31

Initialisierung und Parameteränderungen

Die Funktion der Anweisungen von Easy Motion Control basiert auf den im Achs-DB hinterlegten Daten. Dort sind auch Parameter zu finden, die die Achse selbst grundlegend beschreiben. Werden diese Parameter geändert, ist es erforderlich, dass **jede** in Ihrem Anwenderprogramm aufgerufene Anweisung von Easy Motion Control nach der Änderung **einmal** ihre Initialisierung durchführt. Diese Parameter sind in den Abschnitten zur Konfiguration (Seite 2799) durch folgendes Symbol gekennzeichnet:

Zur Initialisierung ist im Achs-DB das Bitfeld "Init.Ix" definiert und jede Anweisung besitzt den Parameter "Init". Die Anweisungen durchlaufen ihre Initialisierungsroutine, wenn ihr Eingang "Init" gesetzt ist, und setzen anschließend das verschaltete Initialisierungs-Bit zurück.

Die Bits im Bitfeld "Init.Ix" im Achs-DB werden gesetzt:

- bei jedem Aufruf der Anweisung MC_Init.
- von der Projektiersoftware bei Änderung entsprechender Parameter des Achs-DB.

Verschalten Sie den Parameter "Init" bei jeder in Ihrem Anwenderprogramm aufgerufenen Anweisung von Easy Motion Control mit einem Initialisierungsbit im Achs-DB, das von keiner anderen Anweisung benutzt wird. Damit ist gewährleistet, dass die Anweisungen sich initialisieren, wenn dies aufgrund einer Parameteränderung oder eines CPU-Anlaufs notwendig wird ("Init.I0" … "Init.I31", siehe Bitfeld für die Initialisierungsfunktion (Seite 2835)).

Sie rufen einmalig die Anweisung MC_Init in diesen Fällen auf:

- nach jedem Anlauf der CPU
- wenn Sie ohne die Projektiersoftware einen der Parameter geändert haben, die in den Abschnitten zur Konfiguration (Seite 2799) durch folgendes Symbol gekennzeichnet sind:

Ergänzende Informationen zur Initialisierungsanweisung siehe MC_Init.

⚠ WARNUNG
Nur im Stillstand der Achse dürfen Sie Parameter ändern, die in den Abschnitten zur Konfiguration (Seite 2799) durch das nachfolgende Symbol gekennzeichnet sind:
Beachten Sie diese Vorschrift nicht, ist eine kontrollierte Achsbewegung nicht gewährleistet.

⚠ WARNUNG
Bei der Initialisierung werden Hilfswerte berechnet, die in den statischen Parametern der Anweisungen und im Achs-DBs abgelegt werden. Daher dürfen Sie die Instanz-Datenbausteine und den Achs-DB nur im Stillstand der Achse überschreiben.
Beachten Sie diese Vorschrift nicht, ist eine kontrollierte Achsbewegung nicht gewährleistet.

Ablauf von Fahrbewegungen

Start eines Fahrauftrags

Die Eingangsparameter der Fahranweisungen werden mit dem Start eines Fahrauftrags einmal interpretiert. Eine Änderung dieser Werte bei laufendem Auftrag hat keinen Einfluss auf die laufende Fahrt.

Maximaler Fahrweg

Der Fahrweg ist pro Fahrt beschränkt auf den Weg, der 2^{24} Schritten des Gebers entspricht. Er errechnet sich wie folgt:

$$\text{Maximaler Fahrweg} = 2^{24} \times \frac{\text{„Achsweg pro Geberumdrehung“}}{\text{„Schritte pro Geberumdrehung“}}$$

Beispiel:

Achsweg pro Geberumdrehung	= 2,5 mm
Schritte pro Geberumdrehung	= 4096
maximaler Weg = 2^{24} x 2,5 mm / 4096	= 10240 mm

Ausnahme:

Die Anweisungen MC_MoveJog und MC_GearIn haben bei einer Rundachse keine Begrenzung des Fahrwegs.

Ablösen einer Fahrt

Eine gerade aktive Fahranweisung kann jederzeit

- durch den Start einer anderen Fahranweisung abgelöst werden,
- durch den Start der Anweisung MC_StopMotion abgebrochen werden,
- durch Setzen des Handbetriebs abgebrochen werden.

Beim Ablösen durch eine andere Fahranweisung wird die Achse auf die Geschwindigkeit der ablösenden Anweisung gebremst oder beschleunigt und bewegt sich dann ohne Zwischenstopp auf das neue Ziel zu. Die abgelöste Fahranweisung setzt den Ausgang "CommandAborted".

Erfordert die neue Fahrt einen Richtungswechsel der Achse, so wird sie zunächst gestoppt und dann in die Gegenrichtung bewegt (Umkehren).

Aus Sicherheitsgründen müssen die Verzögerungen der beiden sich ablösenden Fahranweisungen übereinstimmen. So ist beim Umkehren gewährleistet, dass das **ursprüngliche** Ziel der Bewegung nicht überfahren wird.

Mehrmalige Verwendung einer Fahranweisung

Sie können eine Fahranweisung, wenn sie in Ihrem Programm nötig ist, mehrmals für eine Achse aufrufen.

Fehlerreaktion

Alle auftretenden Fehler werden im Achs-DB angezeigt und sind dort vom Anwender zu quittieren (siehe Fehleranzeigen und Fehlerquittung im Achs-DB (Seite 2843)). Jeder Fehler führt zum Abbruch eines gerade aktiven Fahrauftrags. Der Start eines neuen Fahrauftrags ist erst möglich, wenn kein Fehler mehr ansteht.

Signalablaufdiagramme

Das folgende Bild zeigt die Signale bei einer Fahrt ohne Fehler.

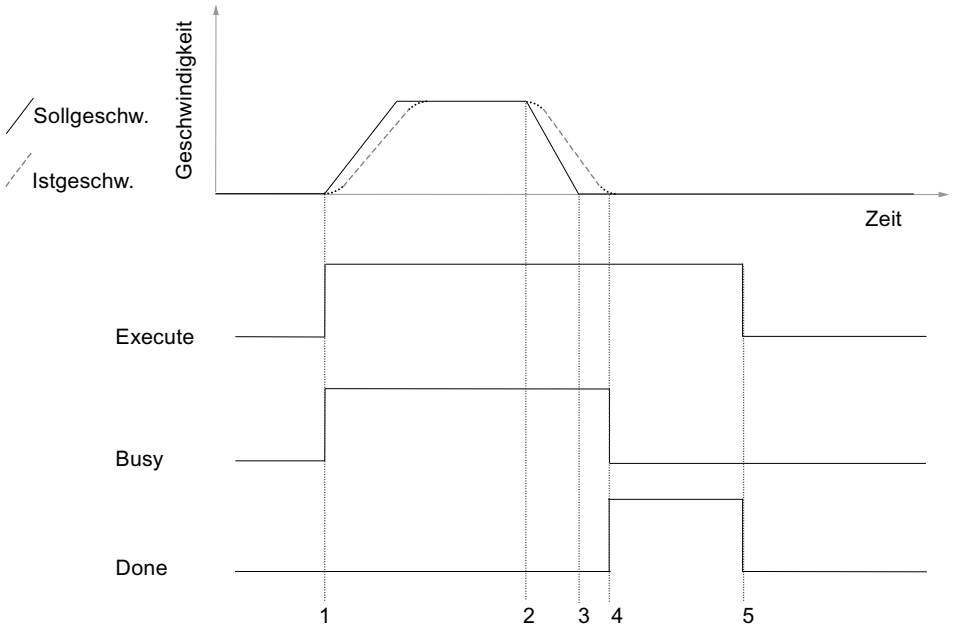

1 Start der Fahranweisung

2 Bremseinsatzpunkt

3 Sollwert hat Zielposition erreicht

4 Achse hat den Zielbereich erreicht

5 Mit Rücksetzen von „Execute“ wird „Done“ rückgesetzt. Der Vorgang ist beendet. Wenn „Execute“ vor dem Erreichen der Zielposition rückgesetzt wird, dann wird „Done“ beim Erreichen der Zielposition für einen Zyklus gesetzt und dann wieder rückgesetzt.

Das folgende Bild zeigt die Signale bei einer Fahrt mit Fehler.

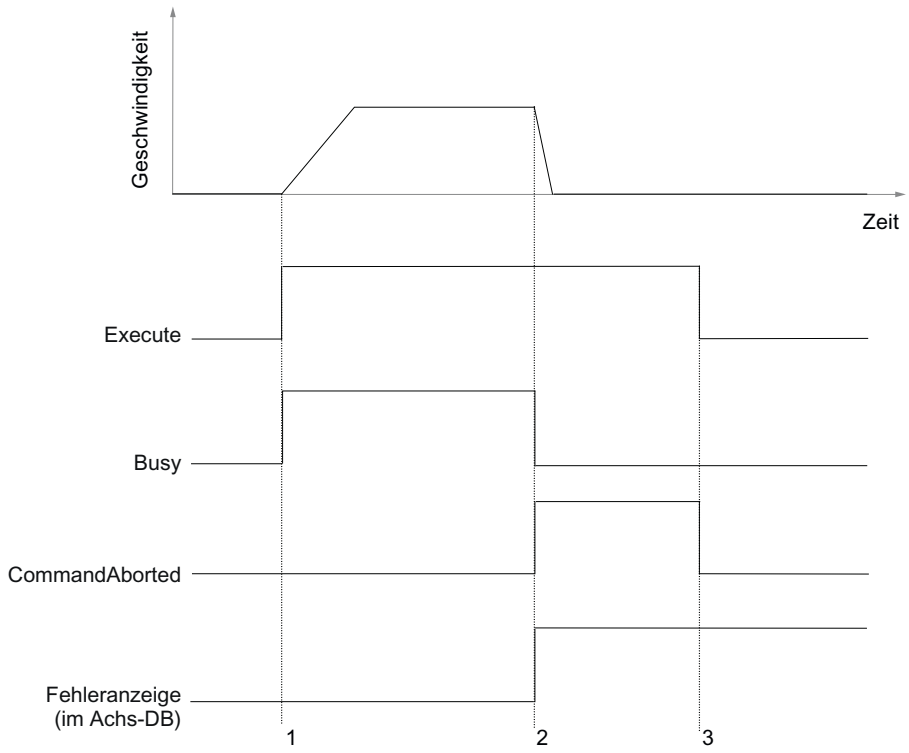

1 Start der Fahranweisung

2 Fehler ist aufgetreten

3 Mit Rücksetzen von „Execute“ wird die Rückmeldung „CommandAborted“ rückgesetzt. Wenn „Execute“ vor Auftreten rückgesetzt wird, dann wird „MotionAborted“ beim Auftreten des Fehlers für einen Zyklus gesetzt und dann wieder rückgesetzt.

Achs-Datenbaustein

Aufgabe

Der Achs-DB ist die zentrale Datenhaltung von Easy Motion Control. Er enthält:

- Parameter, die die Achse grundlegend beschreiben (z. B. Peripherieadressen).
- Parameter der Lageregelung, die bei der Initialisierung der Anweisungen von Easy Motion Control aus diesen Grundparametern berechnet werden.
- Aktualwerte der Lageregelung (z. B. Position, Fehlerzustände, ...).
- Initialisierungs-Bits zur Koordination des Anlaufs der Anweisungen.

Das Zusammenspiel dieser Parametertypen ist dann gewährleistet, wenn Sie eine der folgenden Maßnahmen treffen:

- Bearbeiten und übertragen Sie den Achs-DB mit der Projektiersoftware.
- Sorgen Sie dafür, dass nach Änderung bestimmter Parameter die Anweisung MC_Init aufgerufen wird.

In beiden Fällen wird soweit möglich die Plausibilität der eingegebenen Parameter geprüft und es werden die von Easy Motion Control intern benötigten Werte berechnet.

Die Projektiersoftware analysiert darüber hinaus die Art der Änderungen und steuert darüber die Ladevorgänge.

Speichern und Laden des Achs-Datenbausteins

⚠ WARNUNG
Das Überschreiben des kompletten Achs-DB ist nur im Stillstand der Achse erlaubt. Sie dürfen den Achs-DB im RUN **NICHT** laden. Wenn Sie den Achs-DB in die Baugruppe laden, ohne die Hinweise in den folgenden Abschnitten zu beachten, ist eine korrekte Bewegung der Achse nicht gewährleistet.

Offline

Nach Ändern bestimmter Parameter ist es erforderlich, dass alle Anweisungen von Easy Motion Control eine Initialisierung durchführen.

Diese Parameter sind in der Beschreibung des Achs-DB durch folgendes Symbol gekennzeichnet:

Ändern und Laden dieser Parameter

- ist nur im Stillstand der Achse möglich,
- muss das Setzen der Initialisierungsbits einschließen (siehe Initialisierung und Parameteränderungen (Seite 2827)).

Online

Ein Übernehmen von online geänderten Paramertern (z. B. die Koordinate des Bezugspunkts bei Absolutwertgebern) in die Offline-Datenhaltung wird nicht unterstützt.

Achs-Parameter

Alle Längen-, Geschwindigkeits- bzw. Verzögerungsangaben im Abschnitt "Easy Motion Control konfigurieren (Seite 2797)" beziehen sich auf eine beliebige, aber einheitliche Längeneinheit. Alle Zeitangaben erfolgen in Sekunden.

In den dortigen Tabellen ist in der Spalte "Name" der Parametername aus dem Achs-DB angegeben und in der Spalte "Kommentar" der in der Projektiersoftware verwendete Parametername.

Hinweis

Die beim Konfigurieren nicht aufgelisteten Parameter sind für den internen Gebrauch durch die Anweisungen von Easy Motion Control reserviert und dürfen nicht verändert werden.

Aktualwerte

Übersicht

Folgende Parameter im Achs-DB zeigen während einer Fahrt die aktuellen Informationen zum Zustand Ihrer Achse. Sie dürfen den Inhalt dieser Parameter nicht verändern.

Lageistwert

Hier wird der aktuelle Lageistwert angezeigt.

Adresse	Name	Typ	Anfangswert	Kommentar
104.0	ActPosition	REAL	0.0	**Lageistwert** Wertebereich: Gleitpunktzahl [Längeneinheit]

Schleppabstand

Schleppabstand = Aktueller Lagesollwert – Aktueller Lageistwert

Adresse	Name	Typ	Anfangswert	Kommentar
108.0	FollowingDistance	INT	0	**Schleppabstand** Wertebereich: Gleitpunktzahl [Längeneinheit]

Restweg

Hier wird während einer Fahrt mit Zielangabe (MC_MoveAbsolute, MC_MoveRelative) der Restweg bis zur Zielposition angezeigt.

Eine Fahrt ohne Ziel (MC_MoveJog, MC_Home) sowie ein Fehler mit hartem Stopp setzen den Restweg auf 0.

Adresse	Name	Typ	Anfangswert	Kommentar
112.0	RemainingDistance	REAL	0.0	**Restweg** zum Ziel Wertebereich: Gleitpunktzahl [Längeneinheit]

Sollgeschwindigkeit

Hier wird während einer Fahrt die von der Fahranweisung berechnete Sollgeschwindigkeit der Achse angezeigt.

Adresse	Name	Typ	Anfangswert	Kommentar
116.0	NomVelocity	REAL	0.0	**Sollgeschwindigkeit** Wertebereich: Gleitpunktzahl [Längeneinheit/s]

Istgeschwindigkeit

Hier wird die tatsächliche Geschwindigkeit der Achse angezeigt.

Adresse	Name	Typ	Anfangswert	Kommentar
120.0	ActVelocity	REAL	0.0	**Istgeschwindigkeit** Wertebereich: Gleitpunktzahl [Längeneinheit/s]

Achse synchronisiert

Bei Verwendung eines Inkrementalgebers:

- FALSE: Nach CPU-Anlauf oder Erkennen eines Geberfehlers
- TRUE: Nach Durchführung einer Referenzpunktfahrt oder Bezugspunktsetzen

Bei Verwendung eines Absolutwertgebers:

- FALSE: Vor dem erstmaligen Bezugspunktsetzen
- TRUE: Nach dem Bezugspunktsetzen

Adresse	Name	Typ	Anfangswert	Kommentar
124.0	Sync	BOOL	FALSE	**Achse synchronisiert** FALSE = nicht synchronisiert TRUE = synchronisiert

Aktueller Geberwert

Hier wird der aktuelle Geberwert angezeigt.

Adresse	Name	Typ	Anfangswert	Kommentar
136.0	EncoderValue	DINT	L#0	**Aktueller Geberwert** Wertebereich: Dezimal [Schritte]

Bitfeld für die Initialisierungsfunktion

Funktion des Bitfeldes

Alle Anweisungen von Easy Motion Control müssen nach jedem CPU-Anlauf und nach Änderung bestimmter Parameter initialisiert werden.

Zu diesem Zweck ist im Achs-DB das Bitfeld "Init.Ix" definiert und jede Anweisung besitzt den Parameter "Init". Die Anweisungen durchlaufen ihre Initialisierungsroutine, wenn ihr Eingang "Init" gesetzt ist. Anschließend setzen sie das verschaltete Initialisierungs-Bit zurück.

Sie müssen bei jedem Aufruf einer Anweisung den Parameter "Init" mit einem Bit aus dem Bitfeld verschalten. Dieses Bit darf von keiner anderen Anweisung benutzt werden (siehe Initialisierung und Parameteränderungen (Seite 2827)).

Immer wenn die Anweisung MC_Init aufgerufen wird oder Sie in der Projektiersoftware einen Parameter ändern, der nur nach Initialisierung der Anweisungen aktiv wird, werden gleichzeitig alle Bits in diesem Feld auf TRUE gesetzt. Dadurch führen alle Anweisungen mit ihrem nächsten Aufruf eine Initialisierung durch.

Adresse	Name	Typ	Anfangswert	Kommentar
278.0	Init	STRUCT	TRUE	Bitfeld für Initialisierung der Anweisungen

Anwenderprogramm als Multiinstanz-FB

Wenn Sie alle Parameter Ihrer Fahranweisungen in einem DB ablegen oder für mehrere Achsen das gleiche Programm verwenden wollen, dann erstellen Sie Ihr Anwenderprogramm in einem Multiinstanz-FB.

Legen Sie die Achsdaten ebenfalls in diesen Multiinstanz-DB. Definieren Sie dazu in Ihrem Multiinstanz-FB eine statische Variable vom Typ der AXIS_REF aus der Bibliothek "Easy Motion Control".

Verschalten Sie die Parameter Axis und Init der Anweisungen von Easy Motion Control mit dieser Variablen.

Rufen Sie die Anweisung MC_Init in Ihrem Multiinstanz-Baustein als erste Anweisung bedingt auf. Die Bedingung muss im Anlauf-OB (z. B. OB 100) gesetzt und nach dem Aufruf der Anweisung MC_Init rückgesetzt werden.

Neue Multiinstanz-DB können Sie nur mit dem TIA-Portal oder dem KOP/AWL/FUP-Editor anlegen.

6.2.10.6 Easy Motion Control in Betrieb nehmen

Sicherheitshinweise

Einleitung

Beachten Sie vor der Inbetriebnahme die nachfolgend aufgeführten Hinweise.

> ⚠ WARNUNG
>
> **Die Inbetriebnahmefunktionen bewirken eine Bewegung der Achse.**
>
> Bei der Inbetriebnahme und im Betrieb können unkontrollierte Verfahrbewegungen der Achse zu schweren Personen- und Sachschäden führen.
>
> Zur Vermeidung von Schäden an Personen und Gegenständen treffen Sie folgende Maßnahmen:
>
> - Installieren Sie einen NOT-AUS-Schalter im Umgebungsbereich des Rechners. Nur so können Sie sicherstellen, dass im Falle eines Ausfalls der Hardware und Software die Anlage sicher ausgeschaltet werden kann.
> - Installieren Sie Sicherheits-Endschalter, die direkt auf die Leistungsteile aller Antriebe wirken.
> - Stellen Sie sicher, dass niemand Zutritt zu dem Bereich der Anlage hat, in dem bewegte Teile vorhanden sind.

Technologieobjekt Achse in Betrieb nehmen

Der Inbetriebnahme-Editor hilft Ihnen bei der Inbetriebnahme und dem Funktionstest des Technologieobjekts Achse. Sie können folgende Tests und Messungen vornehmen:

- Verdrahtungstest (Seite 2837)
- Geschwindigkeitsmessung (Seite 2837)
- Referenzieren der Achse (Seite 2838)
- Wegmessung (Seite 2839)

Vorgehen

Um den Inbetriebnahme-Editor des Technologieobjekts Achse zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation die entsprechende Instanz des Technologieobjekts AXIS_REF (z. B. EMC_AXIS_1).
3. Doppelklicken Sie auf das Objekt "Inbetriebnahme".

Abhilfe bei Fehlern

Wenn während der Inbetriebnahme Fehler auftreten, finden Sie Informationen über mögliche Abhilfe im Abschnitt Fehlerbilder und Abhilfemaßnahmen bei der Inbetriebnahme des Anwenderprogramms (Seite 2840).

Zusätzliche Informationen

Detaillierte Informationen zu den Parametern und Schaltflächen der Inbetriebnahmeoberflächen erhalten Sie in den jeweiligen Tooltipps.

Verdrahtung testen

Der Verdrahtungstest dient zur Ermittlung der Drehrichtung der Achse.

Vorgehen

Um den Verdrahtungstest durchzuführen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektieransicht auf "Verdrahtungstest".
2. Klicken Sie auf die Schaltfläche "Aktivieren".
   Eine Rückfrage zur Aktivierung der Handsteuerung wird eingeblendet.
3. Klicken Sie auf die Schaltfläche "OK".
   Eine Online-Verbindung zwischen STEP 7 (TIA-Portal) und CPU wird hergestellt.
4. Geben Sie eine Geschwindigkeit als Prozentsatz der maximalen Geschwindigkeit in das Eingabefeld "Vorgegebene Geschwindigkeit" ein.
5. Geben Sie einen Wert in das Eingabefeld "Hochlaufzeit" ein.
6. Stellen Sie sicher, dass die Anfangsposition eine Bewegung der Achse in beide Richtungen erlaubt.
7. Klicken Sie auf die Schaltfläche "Tippen rückwärts" oder "Tippen vorwärts", um die Achse zu bewegen.
8. Geben Sie bei "Ergebnis" an, ob sich die Achse in die gewünschte Richtung bewegt hat.
9. Wenn Sie als Ergebnis "Ja" gewählt haben, klicken Sie im sich einblendenden Fenster auf die Schaltfläche "OK".
10. Wenn Sie als Ergebnis "Nein" gewählt haben, klicken Sie im sich einblendenden Fenster auf die Schaltfläche "Ja".
    Die Parameter "Richtungsanpassung Geber" und "Richtungsanpassung Antrieb" werden geändert.

Achsgeschwindigkeit messen

Die Geschwindigkeitsmessung dient zur Ermittlung der maximalen Achsgeschwindigkeit.

Vorgehen

Um die Geschwindigkeitsmessung durchzuführen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektieransicht auf "Geschwindigkeitsmessung".
2. Klicken Sie auf die Schaltfläche "Aktivieren".
   Eine Rückfrage zur Aktivierung der Handsteuerung wird eingeblendet.
3. Klicken Sie auf die Schaltfläche "OK".
   Eine Online-Verbindung zwischen STEP 7 (TIA-Portal) und CPU wird hergestellt.
4. Geben Sie eine Geschwindigkeit als Prozentsatz der maximalen Geschwindigkeit in das Eingabefeld "Vorgegebene Geschwindigkeit" ein.
5. Geben Sie einen Wert in das Eingabefeld "Hochlaufzeit" ein.
6. Stellen Sie sicher, dass die Anfangsposition eine ausreichend große Verfahrstrecke in mindestens eine Richtung erlaubt.
7. Klicken Sie auf die Schaltfläche "Tippen rückwärts" oder "Tippen vorwärts" und halten Sie diese, bis der Wert im Ausgabefeld "Ausgang-Ist" das "Ausgang-Soll" erreicht hat.
   Easy Motion Control berechnet die maximale Achsgeschwindigkeit und zeigt sie im Ausgabefeld "Ermittelter Wert" an.
8. Wenn Sie die berechnete Achsgeschwindigkeit im Projekt speichern möchten, klicken Sie auf die Schaltfläche "Übernehmen".
   Der berechnete Wert überschreibt die "Maximale Geschwindigkeit" in der Online-/Offline-Datenhaltung des Technologieobjekts.

Bezugspunkt setzen

Das Setzen eines Bezugspunkts ist für Absolutwertgeber vorgesehen und wird in diesem Fall als Absolutgeberjustage bezeichnet.

Vorgehen

Um einen Bezugspunkt für einen Absolutwertgeber zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektieransicht auf "Referenzieren der Achse".
2. Klicken Sie auf die Schaltfläche "Aktivieren".
   Eine Rückfrage zur Aktivierung der Handsteuerung wird eingeblendet.
3. Klicken Sie auf die Schaltfläche "OK".
   Eine Online-Verbindung zwischen STEP 7 (TIA-Portal) und CPU wird hergestellt.
4. Geben Sie eine Geschwindigkeit als Prozentsatz der maximalen Geschwindigkeit in das Eingabefeld "Vorgegebene Geschwindigkeit" ein.
5. Geben Sie einen Wert in das Eingabefeld "Hochlaufzeit" ein.
6. Klicken Sie auf die Schaltfläche "Tippen rückwärts" oder "Tippen vorwärts", bis die Achse zum Bezugspunkt gefahren ist.
   Alternativ:
   Bewegen Sie die Achse manuell zum Bezugspunkt.

7. Geben Sie einen Positionswert als Bezugspunkt in das Eingabefeld "Eingabe der Referenzposition" ein.
   Der Offset zwischen der Referenzposition und der physikalischen Achsposition wird ermittelt und im Achs-DB gespeichert.
8. Bestätigen Sie im sich einblendenden Fenster, ob Sie den Offset für die Absolutgeberjustage in der Offline-Datenhaltung des Technologieobjekts speichern möchten.

Hinweis

Wenn Sie den Absolutwertgeber austauschen, müssen Sie den Bezugspunkt erneut setzen.

Achsweg messen

Die Wegmessung dient zur Ermittlung des Achswegs pro Geberumdrehung.

Vorgehen

Um die Wegmessung durchzuführen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektieransicht auf "Wegmessung".
2. Klicken Sie auf die Schaltfläche "Aktivieren".
   Eine Rückfrage zur Aktivierung der Handsteuerung wird eingeblendet.
3. Klicken Sie auf die Schaltfläche "OK".
   Eine Online-Verbindung zwischen STEP 7 (TIA-Portal) und CPU wird hergestellt.
4. Geben Sie eine Geschwindigkeit als Prozentsatz der maximalen Geschwindigkeit in das Eingabefeld "Vorgegebene Geschwindigkeit" ein.
5. Geben Sie einen Wert in das Eingabefeld "Hochlaufzeit" ein.
6. Klicken Sie auf die Schaltfläche "Tippen rückwärts" oder "Tippen vorwärts", bis die Achse auf die Anfangsposition gefahren ist.
   Alternativ:
   Bewegen Sie die Achse manuell auf die Anfangsposition.
7. Geben Sie einen Positionswert als Anfangsposition in das Eingabefeld "Eingabe der Anfangsposition" ein.
8. Klicken Sie auf die Schaltfläche "Tippen rückwärts" oder "Tippen vorwärts", bis die Achse auf die Endposition gefahren ist.
   Alternativ:
   Bewegen Sie die Achse manuell auf die Endposition.

9. Geben Sie einen Positionswert als Endposition in das Eingabefeld "Eingabe der Endposition" ein.
   Easy Motion Control berechnet den Achsweg pro Geberumdrehung und zeigt ihn im Ausgabefeld "Ermittelter Wert" an.
10. Wenn Sie den berechneten Achsweg pro Geberumdrehung im Projekt speichern möchten, klicken Sie auf die Schaltfläche "Übernehmen".
    Der berechnete Wert wird in die Online-/Offline-Datenhaltung des Technologieobjekts übernommen.

Fehlerbilder und Abhilfemaßnahmen bei der Inbetriebnahme des Anwenderprogramms

Achse steht trotz Fahrauftrag

Wenn die Achse trotz eines Fahrauftrags steht, dann prüfen Sie folgende Sachverhalte:

Fehler	Abhilfemaßnahmen
Verdrahtung des Antriebs fehlerhaft	• Prüfen Sie, ob der Ausgang "DriveEnabled" der Anweisung MC_Control durchgängig mit dem Freigabeeingang des Leistungsteils verschaltet ist. • Prüfen Sie, ob der Ausgang "DriveEnabled" der Anweisung MC_Control mit dem Eingang "EnableDrive" des Ausgangstreibers verschaltet ist.
Peripherieadressen oder Kanalnummern fehlerhaft	Stellen Sie die Peripherieadressen und Kanalnummern korrekt ein: Im Achs-DB werden die **Anfangs**adressen der Baugruppen erwartet. Aus Anfangsadresse und Kanalnummer (Zählweise: ab Kanal 0) errechnen die Treiberbausteine die baugruppenspezifischen Peripherieadressen für Soll- und Istwert.
Simulationsmodus eingeschaltet	Stellen Sie den Simulationsmodus aus ("Sim" = FALSE). Wenn der Simulationsmodus eingeschaltet ist ("Sim" = TRUE), dann werden **keine** Peripheriewerte gelesen oder geschrieben.
Handbetrieb eingeschaltet bei fehlender Sollgeschwindigkeit	Wenn der Handbetrieb eingeschaltet ist ("ManEnable" = TRUE), dann geben Sie eine "Sollgeschwindigkeit für Handbetrieb" vor ("ManVelocity" <> 0.0 [Längeneinheit/s]).
Antriebsfreigabe nicht erteilt	Setzen Sie den Eingang "EnableDrive" der Anweisung MC_Control auf TRUE.
Achsfehler steht an	Wenn im Achs-DB das Bit "Sammelfehler" ("Error") gesetzt ist, ist in den Bitfeldern zur Fehleranzeige die Art des Achsfehlers ersichtlich: • Nach dem Aufruf der Anweisung MC_Init ist immer der Fehler "Quittierpflichtiger Stopp-Zustand" ("Err.StoppedMotion") gesetzt. Quittieren Sie den Fehler durch Setzen der "Sammelquittung" "ErrorAck" = TRUE. • Starten Sie bei einem Parametrierfehler nach der Korrektur der fehlerhaften Werte einen erneuten Durchlauf von MC_Init. • Prüfen Sie bei einem Geberfehler, ob die Wegerfassungsbaugruppe vollständig parametriert ist (gegebenenfalls unter Zuhilfenahme der Baugruppen-Projektiersoftware). • Andere Fehler: Beheben und quitiieren Sie den Fehler durch Setzen der "Sammelquittung" "ErrorAck" = TRUE.
"Sammelquittung" wird nicht rückgesetzt	Wenn eine von Ihnen gesetzte "Sammelquittung" ("ErrorAck") nicht durch Easy Motion Control erkannt und rückgesetzt wird, dann: • Vermeiden Sie, dass die Anweisungen von Easy Motion Control wegen ständigen Aufrufs der Anweisung MC_Init in ihrem Initialisierungszweig bleiben. • Prüfen Sie, ob die Wegerfassungsbaugruppe auf die Fehlerquittung reagiert.

Fehler	Abhilfemaßnahmen
"Geschwindigkeits-Override" nicht 0 %	Parametrieren Sie "Geschwindigkeits-Override" = 0 % ("Override").
Fahrauftrag vom Programm nicht bearbeitet	Setzen Sie in der Fahranweisung den Parameter "Busy" = TRUE.
"Parameter einer Fahranweisung unzulässig" ("Err.DataErr") oder "Ziel außerhalb des Verfahrbereichs" ("Err.TargetErr")	Wenn Sie den Achs-DB ohne Nutzung der Easy Motion Control-Projektiersoftware in die CPU laden, stellen Sie sicher, dass anschließend die Anweisung MC_Init aufgerufen wird. MC_Init legt Hilfswerte im Achs-DB ab.

Achse fährt ohne Fahrauftrag

Wenn die Achse ohne Fahrauftrag fährt, dann kann das folgende Ursachen haben:

Fehler	Ursachen	Abhilfemaßnahmen
Die Achse fährt sehr schnell los.	Der Wirksinn der Regelung ist falsch eingestellt.	Stellen Sie mit Hilfe des Assistenten "Verdrahtungstest" die Parameter "Richtungsanpassung Geber" ("PolarityEncoder") und "Richtungsanpassung Antrieb" ("PolarityDrive") ein. Dadurch passen Sie die Anweisungen an Ihre Hardwareinstallation an.
Die Achse driftet langsam.	Das Leistungsteil des Antriebs ist freigegeben und die Achse ist **nicht** in Regelung (Anweisung MC_Control gibt "DriveEnabled" = FALSE aus).	—
	Der Analogausgang ist nicht korrekt mit dem Leistungsteil des Antriebs verdrahtet.	Korrigieren Sie die Verdrahtung zwischen dem Analogausgang und dem Leistungsteil des Antriebs.
	Der Eingangstreiber oder der Ausgangstreiber greift nicht auf die zum Antrieb gehörenden Baugruppen zu, weil die Baugruppenadresse oder die Kanalnummer im Achs-DB nicht korrekt eingestellt ist.	Stellen Sie die Parameter mit Hilfe des Assistenten "Verdrahtungstest" ein.

Geschwindigkeit der Achse anders als erwartet oder Achse schwingt

Stimmen die Parameter, die die Achse beschreiben?

Zwischen diesen Parametern bestehen gewisse Zusammenhänge, die durch folgendes Beispiel verdeutlicht werden sollen.

- "Reglerverstärkung" ("FactorP"):
  Ist die gewählte "Reglerverstärkung" ("FactorP") zu groß, kann die Achse schwingen.
- "Maximale Achsgeschwindigkeit" ("MaxVelocity")

- "Bezugswert für maximale Achsgeschwindigkeit" ("DriveInputAtMaxVel"):
  Beispiel für eine ±10 V-Ansteuerung:

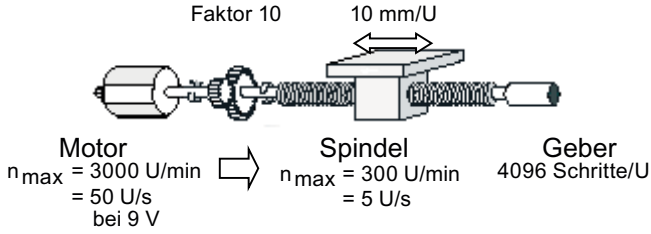

  Es gilt folgender Zusammenhang:
  Gibt die Ausgabebaugruppe 9 V aus, dreht sich der Antrieb mit 3000 U/min. An der Spindel sind das 300 U/min oder 5 U/s. Der Schlitten bewegt sich um 10 mm pro Spindelumdrehung. Somit ergibt sich eine Maximalgeschwindigkeit des Schlittens von 10 mm/U • 300 U/min = 50 mm/s.
  Stellen Sie also ein:

  – "Maximale Achsgeschwindigkeit" = 50 mm/s und "Bezugswert für maximale Achsgeschwindigkeit" = 9 V.

- "Schritte pro Geberumdrehung" ("StepsPerRev"):
  Ob die vom Geber gelieferte Schrittzahl mit dem parametrierten Wert übereinstimmt, können Sie überprüfen, indem Sie mit der Projektiersoftware den Parameter "Geberwert" beobachten, während Sie den **Geber** von Hand **einmal** drehen.
  Beispiel: Der "Geberwert" ändert sich bei einer Spindelumdrehung um 4096 Schritte.
- "Achsweg pro Geberumdrehung" ("DisplacementPerRev"):
  Dieser Parameter ordnet einer Umdrehung des Gebers einen bestimmten Weg zu. Ob dieser stimmt, können Sie überprüfen, indem Sie mit der Projektiersoftware den Parameter "Lageistwert" beobachten, während Sie den **Geber** von Hand **einmal** drehen.
  Beispiel: Der "Lageistwert" ändert sich bei einer Spindelumdrehung um 10 mm.
- "Abtastzeit" ("Sample_T"):
  Die Abtastzeit wird zusammen mit dem seit der letzten Abtastung zurückgelegten Weg zur Berechnung der Achsgeschwindigkeit benutzt. Sie muss der Aufrufzeit des OB entsprechen, in dem die Anweisungen von Easy Motion Control aufgerufen werden.
  Im Achs-DB muss "Sample_T" in der Einheit Sekunden eingetragen werden.

Positionierung nicht genau genug

Wenn die Positionierung nicht genau genug erfolgt, dann:

- überprüfen Sie die Angabe des Parameters "Schritte pro Geberumdrehung" ("StepsPerRev").
- ermitteln Sie den Parameter "Achsweg pro Geberumdrehung" ("DisplacementPerRev") so genau wie möglich (Assistent "Wegmessung").

Siehe auch

Achsweg messen (Seite 2839)

Maximaler Schleppabstand überschritten

Für das Überschreiten des maximalen Schleppabstands gibt es verschiedene Gründe:

- Damit die "Maximale Achsgeschwindigkeit" ("MaxVelocity") überhaupt erreicht werden kann, muss gelten:

$$„MaxFollowingDistance“ \quad >= \frac{„MaxVelocity“}{„FactorP“}$$

- Die parametrierte Beschleunigung kann von der realen Achse nicht erreicht werden.
- Die Parameter "Maximale Achsgeschwindigkeit" ("MaxVelocity") und "Bezugswert für maximale Achsgeschwindigkeit" ("DriveInputAtMaxVel") sowie ggf. "Bezugswert für 100 % Drehzahl" ("DriveInputAt100") sind nicht korrekt eingestellt (siehe unter Geschwindigkeit der Achse anders als erwartet oder Achse schwingt (Seite 2841)).

Die Anweisung MC_GearIn zeigt weder "InGear" noch "Coupled" an

Die Anweisung MC_GearIn zeigt bei folgenden Ursachen weder "InGear" noch "Coupled" an:

- Die Folgeachse koppelt an eine bereits fahrende Leitachse und hat (noch) nicht die Getriebegeschwindigkeit erreicht.
- Die Folgeachse hat die Kopplung aufgrund eines Fehlers aufgelöst.

Die Anweisung MC_GearIn zeigt "InGear", aber nicht "Coupled" an

Die Anweisung MC_GearIn zeigt bei folgenden Ursachen "InGear", aber nicht "Coupled" an:

- Die Folgeachse hat (möglicherweise nur für einen Zyklus) die Getriebegeschwindigkeit erreicht, aber anschließend wieder verloren:
  - Die Leitachse hat stärker beschleunigt oder gebremst, als es der Folgeachse gemäß ihrer Parametrierung und dem Getriebefaktor möglich ist.
  - Die Folgeachse kann die Geschwindigkeit nicht erreichen, die sich aus der Sollgeschwindigkeit der Leitachse und dem Getriebefaktor ergibt.

6.2.10.7 Diagnose

Fehleranzeigen und Fehlerquittung im Achs-DB

Fehleranzeigen und Fehlerquittung

Alle von den Anweisungen erkannten Fehler werden im Achs-DB folgendermaßen angezeigt:

- Durch das Setzen des Sammelfehlers "Error"
- Durch eine fehlerspezifische Anzeige "Err.xxxx"

Bei quittierpflichtigen Fehlern muss nach der Fehlerbeseitigung eine Fehlerquittung gesetzt werden (Parameter "ErrorAck" im Achs-DB, siehe unten).

Es gibt drei Arten von Fehlern mit unterschiedlicher Fehlerreaktion:

- Fehler, bei denen die einwandfreie Funktion der Lageregelung **nicht** mehr gewährleistet ist, führen zu einem **harten Stopp**:
  Die Anweisung MC_Control bringt die Achse mit der in "EmergencyDec" hinterlegten Bremsrampe zum Stillstand. Anschließend setzt die Anweisung MC_Control ihren Ausgang "DriveEnabled" auf "0".
  Nach Beheben des Fehlers setzen Sie die Fehlerquittung im Achs-DB. Danach wird "DriveEnabled" wieder gesetzt.
- Fehler, bei denen die Lageregelung noch funktioniert, führen zu einem **weichen Stopp**:
  Die aktive Fahranweisung bremst die Achse mit der parametrierten Verzögerung. Die Achse bleibt in Regelung, "DriveEnabled" wird **nicht** rückgesetzt.
  In diesem Zustand kann die Fahrt nur noch durch MC_StopMotion abgelöst werden.
  Nach Beheben des Fehlers setzen Sie die Fehlerquittung im Achs-DB.
- Fehler in den Achsparametern (**Parametrierfehler**):
  Die Anweisung MC_Init prüft soweit möglich die im Achs-DB eingetragenen Parameter. Im Fehlerfall setzt sie das Bit "Err.ConfigErr" und trägt die genaue Fehlerursache in "Config.Err_xxx" ein.
  Parametrierfehler können durch Quittierung **nicht** beseitigt werden. Ein Anfahren der Achse wird verhindert. Nach der Korrektur des falsch eingegebenen Parameters muss die Anweisung MC_Init erneut aufgerufen werden. Dabei werden die Parametrierfehleranzeigen rückgesetzt, wenn kein Fehler mehr ansteht.

Hinweis

Für die Fehler mit hartem und weichem Stopp gilt:

Solange ein oder mehrere Fehler anstehen und nicht quittiert wurden, ist kein Fahrauftrag möglich.

Ausnahme:

Wird nur ein Software-Endschalter überfahren, dann ist ein Fahrauftrag in die entgegengesetzte Richtung möglich. Nach Verlassen des Software-Endschalters wird dieser Fehler automatisch quittiert.

Hinweis

Wenn der Fehler durch eine Fahranweisung (z. B. MC_MoveJog oder MC_MoveAbsolute) erkannt wurde, dann wird bei dieser Anweisung der Fehlerausgang "Error" gesetzt.

Wenn der Fehler durch eine Treiberanweisung erkannt wurde, dann wird bei dieser Anweisung die genaue Fehlerursache angezeigt (siehe Eingangstreiber).

Fehleranzeige im Achs-DB

Der Sammelfehler ist TRUE, solange mindestens ein Fehler ansteht.

Adresse	Name	Typ	Anfangswert	Kommentar
124.1	Error	BOOL	TRUE	**Sammelfehler** FALSE = kein Fehler TRUE = Sammelfehler

Fehlerquittung im Achs-DB

Mit Setzen der Sammelquittung werden alle quittierbaren Fehleranzeigen gelöscht.

Die Sammelquittung wirkt erst im Stillstand der Achse und nach der Initialisierung.

Nach Auswertung der Fehlerquittung wird "ErrorAck" automatisch zurückgesetzt.

Adresse	Name	Typ	Anfangswert	Kommentar
124.2	ErrorAck	BOOL	FALSE	**Sammelquittung** für alle Fehler FALSE = keine Quittung TRUE = Fehler quittieren

Achsfehler

Fehler mit weichem Stopp

Beschreibung

Für alle nachfolgend beschriebenen Fehler gilt: TRUE = der Fehler steht an.

Wenn für den Fehler eine Fehlerquittung notwendig ist, dann ist das mit folgendem Symbol gekennzeichnet:

Die folgende Tabelle zeigt alle Fehler mit weichem Stopp.

Adresse	Name: Err.xxx	Beschreibung		
128.0	SWLimitMinExceeded	**Software-Endschalter Anfang überfahren**		—
		Ursache:	• Der Lageistwert liegt außerhalb des Arbeitsbereichs. • Ein Gebertausch bei einem Absolutwertgeber.	
		Abhilfe:	• Fahren Sie die Achse aus dem Endschalter heraus. • Schalten Sie die Software-Endschalter aus ("SWLimitEnable" = FALSE), synchronisieren Sie die Achse neu und schalten Sie die Software-Endschalter wieder ein.	

Adresse	Name: Err.xxx	Beschreibung		Q
128.1	SWLimitMaxExceeded	**Software-Endschalter Ende überfahren**		—
		Ursache:	• Der Lageistwert liegt außerhalb des Arbeitsbereichs. • Ein Gebertausch bei einem Absolutwertgeber.	
		Abhilfe:	• Fahren Sie die Achse aus dem Endschalter heraus. • Schalten Sie die Softwareendschalter aus ("SWLimitEnable" = FALSE), synchronisieren Sie die Achse neu und schalten Sie die Software-Endschalter wieder ein.	
128.2	TargetErr	**Ziel außerhalb des Verfahrbereichs**		Q
		Ursache:	• **MC_MoveAbsolute**: – Das Ziel liegt auf einem Software-Endschalter. – Das Ziel ist < "Anfang der Rundachse" oder ≥ "Ende der Rundachse". – Die berechnete Fahrstrecke ist größer als der Weg, der 2^{24} Schritten Ihres Gebers entspricht. – Hilfsgrößen im Achs-DB wurden nicht berechnet, weil die Anweisung MC_Init nicht aufgerufen wurde. • **MC_MoveRelative**: – Die eingegebene Fahrstrecke führt zu einem Ziel auf einem Software-Endschalter. – Die vorgegebene Wegstrecke ist größer als der Weg, der 2^{24} Schritten Ihres Gebers entspricht. – Hilfsgrößen im Achs-DB wurden nicht berechnet, weil die Anweisung MC_Init nicht aufgerufen wurde. • **MC_Home**: – Die Referenzpunktkoordinate liegt auf einem Software-Endschalter. – Die Referenzpunktkoordinate ist < "Anfang der Rundachse" oder ≥ "Ende der Rundachse". – Hilfsgrößen im Achs-DB wurden nicht berechnet, weil die Anweisung MC_Init nicht aufgerufen wurde.	
		Abhilfe:	• Korrigieren Sie die Parameter. • Rufen Sie die Anweisung MC_Init auf.	
128.3	NoSync	**Achse nicht synchronisiert**		Q
		Ursache:	Die Anweisung MC_MoveAbsolute wurde bei unsynchronisierter Achse gestartet.	
		Abhilfe:	Synchronisieren Sie die Achse (Referenzpunktfahrt, Bezugspunktsetzen).	
128.4	DirectionErr	**Fahrt in vorgegebene Richtung unzulässig**		Q
		Ursache:	Eine Bewegung weiter in einen Software-Endschalter ist unzulässig.	
		Abhilfe:	Starten Sie einen Fahrauftrag weg vom Software-Endschalter.	

Adresse	Name: Err.xxx	Beschreibung		Q
128.5	DataErr	**Parameter einer Fahranweisung unzulässig**		Q
		Ursache:	**MC_MoveHome**: • "Velocity" < 0 • "Velocity" > 0 und "Acceleration" ≤ 0 • "Velocity" > 0 und "Deceleration" ≤ 0 • "Velocity" > 0 bei Absolutwertgeber • Bei einer Linearachse liegt "Position" außerhalb des Zahlenbereichs. • Hilfsgrößen im Achs-DB wurden nicht berechnet, weil die Anweisung MC_Init nicht aufgerufen wurde. **MC_MoveAbsolute**, **MC_MoveRelative** und **MC_MoveJog**: • "Velocity" ≤ 0 • "Acceleration" ≤ 0 • "Deceleration" ≤ 0 • Die eingegebene oder berechnete Zielkoordinate liegt außerhalb des Zahlenbereichs (bei MC_MoveAbsolute und MC_MoveRelative). • Nenner des Getriebefaktors (RatioDenominator") ≤ 0 (MC_GearIn) • Zähler des Getriebefaktors ("RatioNumerator") = 0 (MC_GearIn) • Eine Fahranweisung wurde unterbrochen und "Deceleration" der unterbrechenden Anweisung stimmt nicht mit der an der unterbrochenen Anweisung parametrierten Verzögerung überein (siehe Ablösen einer Fahrt (Seite 2796)). • Hilfsgrößen im Achs-DB wurden nicht berechnet, weil die Anweisung MC_Init nicht aufgerufen wurde.	
		Abhilfe:	• Geben Sie zulässige Werte vor. • Initialisieren Sie die Anweisungen. • Rufen Sie die Anweisung MC_Init auf.	

Adresse	Name: Err.xxx	Beschreibung		Q
128.6	StartErr	**Start im aktuellen Achszustand nicht möglich**		Q
		Ursache:	• An einer Fahranweisung oder an MC_StopMotion wurde der Execute-Eingang zu einem Zeitpunkt gesetzt, als sich die Achse in einem Fehlerzustand oder im Handbetrieb ("ManEnable" = TRUE) befand oder während eine Anweisung MC_StopMotion aktiv war. • Die Anweisung MC_Home wurde nicht im Stillstand der Achse gestartet. • Bei der Anweisung MC_MoveJog sind beide Richtungen gesetzt. • Die Leitachse für die Anweisung MC_GearIn ist in einem unzulässigen Zustand: – Fehler mit hartem Stopp – MC_Home ist aktiv – Handebetrieb ("ManEnable" = TRUE)	
		Abhilfe:	• Beseitigen Sie eventuell anstehende Fehler. • Sorgen Sie dafür, dass die Leitachse in einem zulässigen Zustand ist.	
128.7	DistanceErr	**Fahrt zu weit**		Q
		Ursache:	Eine Fahrt ohne Ziel (MC_Home, MC_MoveJog, MC_GearIn) hat eine Wegstrecke zurückgelegt, die mehr als 2^{24} Schritten des Gebers entspricht. Die Achse wird etwa auf diesen Punkt gebremst. Bei einer synchronisierten Linearachse hat die Anweisung MC_MoveJog bzw. MC_GearIn auf einen Software-Endschalter gebremst. Außerdem ist in den meisten Fällen "SWLimitMinExceeded" bzw. "SWLimitMaxExceeded" gesetzt.	
		Abhilfe:	Sorgen Sie dafür, dass die Wegstrecke die oben angegebene Grenze nicht überschreitet (siehe Maximaler Fahrweg (Seite 2828)).	
129.0	MasterErr	**Leitachse in unzulässigem Zustand**		Q
		Ursache:	Während der Getriebekopplung wurde die Leitachse in einen unzulässigen Zustand versetzt: • Fehler mit hartem Stopp • MC_Home ist aktiv • Handbetrieb ("ManEnable" = TRUE)	
		Abhilfe:	Beseitigen Sie den unzulässigen Zustand der Leitachse.	

Fehler mit hartem Stopp

Beschreibung

Für alle nachfolgend beschriebenen Fehler gilt: TRUE = der Fehler steht an.

Wenn für den Fehler eine Fehlerquittung notwendig ist, dann ist das mit folgendem Symbol gekennzeichnet:

Die folgende Tabelle zeigt alle Fehler mit hartem Stopp.

Adresse	Name: Err.xxx	Beschreibung	
130.0	StoppedMotion	**Achse im quittierpflichtigen Stopp-Zustand**	
		Ursache:	Die Achse befindet sich im quittierpflichtigen Stopp-Zustand • nach Initialisierung durch Aufruf der Anweisung MC_Init. • durch einen der in dieser Tabelle aufgelisteten Fehler.
		Abhilfe:	Beseitigen Sie einen evtl. anstehenden Fehler.
130.1	EnableDriveErr	**Antriebsfreigabe fehlt**	
		Ursache:	An der Anweisung MC_Control ist "EnableDrive" nicht gesetzt.
		Abhilfe:	Setzen Sie "EnableDrive".
130.2	FollowingDistErr	**Schleppabstand überschritten**	
		Ursache:	Die Achse folgt den Sollwerten nicht oder zu langsam.
		Abhilfe:	• Schalten Sie den Antrieb ein. • Optimieren Sie die "Reglerverstärkung" (Achs-DB Adresse 44.0 (Seite 2809)). • Passen Sie die "Richtungsanpassung Geber" (Achs-DB Adresse 74.0 (Seite 2808)) an die "Richtungsanpassung Antrieb" (Achs-DB Adresse 86.0 (Seite 2811)) an. • Vergrößern Sie den maximal zulässigen Schleppabstand (Achs-DB Adresse 38.0 (Seite 2820)).
130.3	StandstillErr	**Stillstandsbereich verlassen**	
		Ursache:	• Die Achse hat sich ohne Fahrauftrag aus dem "Stillstandsbereich" bewegt. • Der "Stillstandsbereich" ist kleiner als der "Zielbereich" • Die "Richtungsanpassung Geber" oder die "Richtungsanpassung Antrieb" ist falsch parametriert.
		Abhilfe:	• Prüfen Sie die Verbindung zum Sollwerteingang des Leistungsteils. • Korrigieren Sie die Parameter "Stillstandsbereich" / "Zielbereich" (Achs-DB Adresse 34.0 (Seite 2819) / 30.0 (Seite 2818)) • während Erstinbetriebnahme: Führen Sie einen Verdrahtungstest durch (Achs-DB Adresse 74.0 (Seite 2808) / 86.0 (Seite 2811)).

Adresse	Name: Err.xxx	Beschreibung		Q
130.4	TargetApproachErr	**Fehler beim Zieleinlauf**		Q
		Ursache:	Der Istwert hat den "Zielbereich" innerhalb der "Überwachungszeit für Zieleinlauf" nicht erreicht.	
		Abhilfe:	• Optimieren Sie den "Zielbereich" und die "Überwachungszeit für Zieleinlauf" (Achs-DB Adresse 30.0 (Seite 2818) / 26.0 (Seite 2820)). • Überprüfen Sie Antrieb und Achse.	
130.5	EncoderErr	**Geberfehler**		Q
		Ursache:	• Der Eingangstreiber hat einen Geberfehler erkannt. Die genaue Fehlerursache wird im Instanz-DB der Treiberanweisung angezeigt. Siehe Eingangstreiber. • Am Eingangstreiber wurde der Eingang "EncErr" gesetzt (siehe Eingangstreiber , Abschnitt Fehlerbehandlung).	
		Abhilfe:	Beseitigen Sie die Fehlerursache.	
130.6	OutputErr	**Fehler am Ausgangstreiber**		Q
		Ursache:	Am Ausgangstreiber wurde der Eingang "OutErr" gesetzt (siehe Ausgangstreiber , Abschnitt Fehlerbehandlung).	
		Abhilfe:	Beseitigen Sie die Fehlerursache.	
130.7	ConfigErr	**Sammelanzeige: Achsdaten fehlerhaft parametriert**		**Quittierung nicht möglich**
		Ursache:	Die Parameter im Achs-DB sind fehlerhaft. Genaue Fehlerursache siehe Parametrierfehler (Seite 2850).	
		Abhilfe:	Ändern Sie den fehlerhaften Parameter und sorgen Sie dafür, dass die Anweisung MC_Init aufgerufen wird.	
131.0	DriveErr	**Antriebsfehler**		**Quittierung nicht möglich**
		Ursache:	Der Ausgangstreiber hat einen Fehler am Leistungsteil / Antrieb erkannt. Die genaue Fehlerursache wird im Instanz-DB der Treiberanweisung angezeigt. Siehe Ausgangstreiber).	
		Abhilfe:	Beseitigen Sie die Fehlerursache.	

Parametrierfehler

Parametrierfehler

Die folgende Tabelle zeigt die möglichen Parametrierfehler mit Easy Motion Control.

Die Spalte "Adresse" enthält die Adresse der Fehleranzeige im Achs-DB. Die Spalte "Parameteradresse" verweist auf die Adresse des fehlerhaften Parameters im Achs-DB (siehe Aufbau des Achsdatenbausteins).

Adresse	Name Config.xxx	Parametrierfehler	Parameteradresse
132.0	Err_AxisLimit	"Software-Endschalter Anfang" < $-2^{24}+6$ [Längeneinheit] "Software-Endschalter Ende" > $+2^{24}-6$ [Längeneinheit] "Software-Endschalter Ende" ≤ "Software-Endschalter Anfang"	6.0 (Seite 2812) , 10.0 (Seite 2812)
132.1	Err_MaxVelocity	"Maximale Achsgeschwindigkeit" ≤ 0.0 [Längeneinheit/s]	14.0 (Seite 2814)
132.2	Err_MaxAcceleration	"Maximale Beschleunigung" ≤ 0.0 [Längeneinheit/s^2]	18.0 (Seite 2816)
132.3	Err_MaxDeceleration	"Maximale Verzögerung" ≤ 0.0 [Längeneinheit/s^2]	22.0 (Seite 2816)
132.4	Err_MonTimeTargetAppr	"Überwachungszeit für Zieleinlauf" ≤ 0.0 [s]	26.0 (Seite 2820)
132.5	Err_TargetRange	"Zielbereich" ≤ 0.0 [Längeneinheit]	30.0 (Seite 2818)
132.6	Err_StandstillRange	"Stillstandsbereich" ≤ 0.0 [Längeneinheit]	34.0 (Seite 2819)
132.7	Err_MaxFollowingDist	"Maximal zulässiger Schleppabstand" ≤ 0.0 [Längeneinheit]	38.0 (Seite 2820)
133.0	Err_EmergencyDec	"Verzögerung für harten Stopp" ≤ 0.0 [Längeneinheit/s^2]	54.0 (Seite 2818)
133.1	Err_StepsPerRev	"Schritte pro Geberumdrehung" < 1	64.0 (Seite 2805)
133.2	Err_DisplacementPerRev	"Achsweg pro Geberumdrehung" ≤ 0.0 [Längeneinheit]	68.0 (Seite 2807)
133.3	Err_NumberRevs	"Anzahl Geberumdrehungen" < 1 bei Absolutwertgeber	72.0 (Seite 2806)
133.4	Err_PolarityEncoder	"Richtungsanpassung" Geber ungleich ±1	74.0 (Seite 2808)
133.5	Err_PolarityDrive	"Richtungsanpassung Antrieb" ungleich ±1	86.0 (Seite 2811)
133.6	Err_DriveInputAtMaxVel	"Bezugswert für maximale Achsgeschwindigkeit" ≤ 0.0 oder > "DriveInputAt100"	88.0 (Seite 2810) , 96.0 (Seite 2810)
133.7	Err_AxisLength	Achslänge > 2^{24} [Längeneinheit] oder Achslänge > (2^{31} -1) [Schritte]	6.0 (Seite 2812) , 10.0 (Seite 2812)
134.0	Err_EncoderRange	Bereich des Gebers passt nicht zur Achslänge (siehe "Achsweg pro Geberumdrehung", Achs-DB Adresse 68.0 (Seite 2807))	6.0 (Seite 2812) , 10.0 (Seite 2812) , 68.0 (Seite 2807) , 72.0 (Seite 2806)
134.1	Err_MaxVelRotaryAxis	Maximalgeschwindigkeit und Abtastzeit passen nicht zur Länge der Rundachse (siehe "Maximale Achsgeschwindigkeit", Achs-DB Adresse 14.0 (Seite 2814))	14.0 (Seite 2814) , 0.0 (Seite 2816) , 68.0 (Seite 2807) , 72.0 (Seite 2806)
134.2	Err_DriveInputAt100	"Bezugswert für 100 % Drehzahl" ≤ 0	96.0 (Seite 2810)

JOB_ERR-Meldungen

JOB_ERR-Meldungen

JOB_ERR (Hex)	JOB_ERR (Dez)	JOB_ERR (Int)	Bedeutung
80A0	32928	-32608	Negative Quittung beim Lesen von der Baugruppe. Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt.
80A1	32929	-32607	Negative Quittung beim Schreiben zur Baugruppe. Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt.
80A2 1)	32930	-32606	Protokollfehler bei Layer 2
80A3 1)	32931	-32605	Protokollfehler bei User-Interface / User
80A4 1)	32932	-32604	Kommunikation am K-Bus gestört
80B0	32944	-32592	Datensatz / Auftrag unbekannt
80B1	32945	-32591	Längenangabe falsch. Parameter FM_TYPE im Kanal-DB für die verwendete Baugruppe nicht richtig gesetzt.
80B2	32946	-32590	Der projektierte Steckplatz ist nicht belegt.
80B3	32947	-32589	Ist-Baugruppentyp ungleich Soll-Baugruppentyp.
80C0 1)	32960	-32576	Die Baugruppe hat die zu lesenden Daten noch nicht bereit.
80C1 1)	32961	-32575	Die Daten eines gleichartigen Schreibauftrags sind auf der Baugruppe noch nicht verarbeitet.
80C2 1)	32962	-32574	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen.
80C3 1)	32963	-32573	Benötigte Betriebsmittel (Speicher usw.) sind momentan belegt.
80C4 1)	32964	-32572	Kommunikationsfehler
80C5 1)	32965	-32571	Dezentrale Peripherie nicht verfügbar
80C6 1)	32966	-32570	Prioritätsklassenabbruch (Wiederanlauf oder Hintergrund)
8522	34082	-31454	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht aus dem DB gelesen werden (Schreibauftrag).
8532	34098	-31438	DB-Nummer des Parameter-DB zu groß (Schreibauftrag)
853A	34106	-31430	Parameter-DB nicht vorhanden (Schreibauftrag)
8544	34116	-31420	Fehler beim n-ten (n > 1) Lesezugriff auf einen DB nach Auftreten eines Fehlers (Schreibauftrag)
8723	34595	-30941	Kanal-DB bzw. Parameter-DB zu kurz. Die Daten können nicht in den DB geschrieben werden (Leseauftrag).
8730	34608	-30928	Parameter-DB in der CPU schreibgeschützt. Die Daten können nicht in den DB geschrieben werden (Leseauftrag).
8732	34610	-30926	DB-Nummer des Parameter-DB zu groß (Leseauftrag).
873A	34618	-30918	Parameter-DB nicht vorhanden (Leseauftrag)
8745	34629	-30907	Fehler beim n-ten (n > 1) Schreibzugriff auf einen DB nach Auftreten eines Fehlers (Leseauftrag).

1) Die Fehler 80A2 … 80A4 sowie 80Cx sind temporär, d. h. sie können nach einer Wartezeit ohne Ihr Zutun behoben sein.

SIMATIC Ident 7

7.1 Ident-Systeme konfigurieren

7.1.1 Technologieobjekt "SIMATIC Ident"

Einleitung

Das Technologieobjekt für SIMATIC Ident-Produkte ermöglicht die einfache Parametrierung eines Ident-Systems im TIA Portal. Es schafft eine leicht zu parametrierende Schnittstelle zwischen der projektierten Hardware und den Funktionsbausteinen in der Bibliothek "SIMATIC Ident".

Mit Hilfe des Objekts "Konfiguration" können Sie ein projektiertes Gerät auswählen, den Kanal / die Lesestelle bestimmen und ggf. das angeschlossene Ident-System festlegen. Das Technologieobjekt bestimmt dadurch automatisch alle, für den Betrieb benötigten Parameter.

Im Objekt "Diagnose" werden die 5 zuletzt gemeldeten Fehlermeldungen angezeigt, welche das Ident-Gerät an die Funktionsbausteine gemeldet hat. Zusätzlich werden bei jeder Fehlermeldung, eine Fehlerbeschreibung, sowie detaillierte Informationen über den ausgeführten Befehl angezeigt.

Die nachfolgende Übersicht zeigt die grundsätzliche Vorgehensweise, um ein Technologieobjekt "SIMATIC Ident" hinzuzufügen und zu projektieren.

Vorgehen

Gehen Sie folgendermaßen vor, um das Technologieobjekt "SIMATIC Ident" zu projektieren:

Schritt	Beschreibung
1	Technologieobjekt hinzufügen (Seite 2854)
2	AUTOHOTSPOT
3	Ident-Anweisung im Anwenderprogramm aufrufen (Seite 2861)
4	Technologieobjekt in CPU laden
5	Geräte diagnostizieren (Seite 2863)

7.1.2 Technologieobjekt hinzufügen

Technologieobjekt im Projektnavigator hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Datenbaustein erzeugt. In diesem Datenbaustein wird die Konfiguration des Technologieobjekts hinterlegt.

Voraussetzungen

Um das Technologieobjekt "SIMATIC Ident" einzusetzen, müssen folgende Voraussetzungen erfüllt sein:

- Ein Projekt (V14 SP1 oder höher) wurde angelegt.
- Entsprechend der verwendeten Hardware wurden die angeschlossenen Geräte in die Hardware-Konfiguration ("Geräte & Netze") hinzugefügt, miteinander vernetzt und projektiert.
- Das Projekt enthält mindestens eine S7-Steuerung und mindestens ein Ident-Gerät (z. B. Kommunikationsmodul, Reader, Optisches Lesegerät).
  Folgende Steuerungen sind kompatibel:
  - S7-1200 ab V4.0
  - S7-1500 ab V1.8
  - ET200pro (CPU 1516pro) ab V1.8
  - ET200SP ab V1.8

Vorgehen

Gehen Sie folgendermaßen vor, um ein Technologieobjekt hinzuzufügen:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".
   Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "SIMATIC Ident".
5. Wählen Sie in der Spalte "Version" die neueste Version (ab "V5.0").
6. Wählen Sie das Objekt "TO_Ident".
7. Geben Sie im Eingabefeld "Name" einen individuellen Namen für das Technologieobjekt ein.
8. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt hinterlegen möchten.
9. Bestätigen Sie mit "OK".

Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

7.1.3 Geräte mithilfe des Technologieobjekts konfigurieren

7.1.3.1 Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Gehen Sie folgendermaßen vor, um das Konfigurationsfenster des Technologieobjekts zu öffnen:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- Grundparameter
- Ident-Gerät-Parameter
- Reader-Parameter

Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation des Konfigurations- und Inspektorfensters zeigen weitere Details zur Vollständigkeit der Konfiguration:

	Die Konfiguration enthält Voreinstellungswerte und ist vollständig Die Konfiguration enthält ausschließlich Voreinstellungswerte. Mit diesen Voreinstellungswerten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.
	Die Konfiguration enthält vom Anwender definierte oder automatisch angepasste Werte und ist vollständig Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein Voreinstellungswert wurde geändert.
	Die Konfiguration ist gültig, enthält jedoch unerwartete Werte Mindestens ein Eingabefeld oder eine Klappliste enthält einen unerwarteten Wert. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Warnmeldung die Ursache für die Warnung an.
	Die Konfiguration ist unvollständig oder fehlerhaft Mindestens ein Eingabefeld oder eine Klappliste enthält keinen oder einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an.

7.1.3.2 Grundparameter konfigurieren

In dieser Parametergruppe können Sie die baugruppenspezifischen Parameter Ihres Systems konfigurieren.

Tabelle 7-1 Parameter der Parametergruppe "Grundparameter"

Parameter	Beschreibung
Ident-Gerät	Klicken Sie auf das Symbol "..." und wählen Sie im Ordner "Dezentrale Peripherie" bzw. "Lokale Module" das zu konfigurierende Ident-Gerät aus. Folgende Ident-Geräte können ausgewählt werden: • ASM 456 • RF120C, RF170C, RF180C • RF18xC • RF615R • RF680R, RF685R • MV400, MV500
Kanal	Öffnen Sie die Klappliste und wählen Sie ggf. den Kanal des Kommunikationsmoduls aus, an dem das zu konfigurierende Gerät angeschlossen ist.
Reader-Parametrierung	Öffnen Sie die Klappliste und wählen Sie das an das Ident-Gerät angeschlossene Gerät/ System aus, das parametriert werden soll.
Lesestelle	Voraussetzung: Im Parameter "Reader-Parametrierung" wurde ein RF600-Reader ausgewählt. Öffnen Sie die Klappliste und wählen Sie die Lesestelle aus, die verwendet werden soll.

7.1.3.3 Ident-Gerät-Parameter konfigurieren

In dieser Parametergruppe werden abhängig von dem ausgewählten Ident-Gerät, nachfolgende Ident-Gerät-Parameter angezeigt. Die angezeigten Ident-Gerät-Parameter werden automatisch aus der Gerätekonfiguration der angeschlossenen Geräte übernommen.

Beachten Sie, dass nicht alle aus der Gerätekonfiguration der angeschlossenen Geräte übernommen und angezeigten Parameterwerte in Verbindung mit dem Technologieobjekt zulässig sind. Sind nicht zulässige Werte in der Gerätekonfiguration eingestellt, können Sie

diese entweder über die Schaltfläche "Auto-Konfiguration" automatisch oder über Gerätekonfiguration manuell ändern.

Tabelle 7-2 Parameter der Parametergruppe "Ident-Gerät-Parameter"

Parameter	Mögl. angezeigte Parameterwert	Zulässige Parameterwerte	Beschreibung
User Mode	Ident-Profil/RFID-Normprofil FB 45 / FC 45 FB 55 / FC 55	Ident-Profil/RFID-Normprofil	Mit diesem Parameter wählen Sie den Baustein aus. Beachten Sie, dass ausschließlich die nachfolgenden Parameterwerte in Verbindung mit dem Technologieobjekt zulässig sind: • Ident-Profil/RFID-Normprofil: In der Steuerung kommt der Programmbaustein für das Ident-Profil/RFID-Normprofil zum Einsatz.
MOBY Mode	RF200/RF300/RF600; MV400/MV500; MOBY U/D RF680R/RF685R MOBY I/E Normaladressierung RF300 Filehandler MOBY U Filehandler MOBY I Filehandler	RF200/RF300/RF600; MV400/MV500; MOBY U/D RF680R/RF685R	Mit diesem Parameter stellen Sie die Betriebsart des Kommunikationsmoduls ein. Beachten Sie, dass ausschließlich die nachfolgenden Parameterwerte in Verbindung mit dem Technologieobjekt zulässig sind: • RF200/RF300/RF600; MV400/MV500; MOBY U/D • RF680R/RF685R
Angeschlossenes Gerät	Parameter über FB / Optische Lesegeräte	Parameter über FB / Optische Lesegeräte	Dieser Parameter wird nur in Verbindung mit dem Kommunikationsmodul RF120C angezeigt. Beachten Sie, dass ausschließlich die nachfolgenden Parameterwerte in Verbindung mit dem Technologieobjekt zulässig sind: • Parameter über FB / Optische Lesegeräte

Schaltfläche "Auto-Konfiguration"

Die angezeigten HWCN-Parameter werden automatisch aus der Gerätekonfiguration der angeschlossenen Geräte übernommen. Beachten Sie, dass nicht alle Parameterwert in Verbindung mit dem Technologieobjekt zulässig sind. Nicht zulässige Parameterwerte werden farblich hervorgehoben.

Klicken Sie auf die Schaltfläche "Auto-Konfiguration", um automatisch die passenden Parameterwerte einzustellen (z. B. User Mode). Die Parameterwerte des angeschlossenes Geräts werden für die Verwendung mit dem Technologieobjekt vorbereitet.

Hinweis

Verhalten bei MV400/MV500

Beachten Sie, dass bei den optischen Lesegeräten MV400 und MV500, die Schaltfläche "Auto-Konfiguration" nicht funktioniert. Bei diesen Geräten müssen Sie die HWCN-Parameter manuell in der Gerätekonfiguration der angeschlossenen Geräte eintragen.

7.1.3.4 Reader-Parameter konfigurieren

In dieser Parametergruppe können Sie abhängig von den eingestellten Grundparametern, nachfolgende Reader-Parameter konfigurieren.

Tabelle 7-3 Übersicht über die "Reader-Parameter"

Reader-Parametrierung bei "Grundparameter"	Übertragungsgeschwindigkeit	Anwesenheitskontrolle	Error-LED zurücksetzen	HF-Leistung	Max. Anzahl von Transpondern	Transponder-Typ			Programmanwahl bei Reset	Byte-Sequenz des Reset-Parameters [hex]
						ISO Mode	RF300	MOBYE		
RF200 allgemein	✓	✓	✓	-	✓	✓	-	-	-	-
RF290R	✓	✓	✓	✓	✓	✓	-	-	-	-
RF300 allgemein	✓	✓	✓	-	✓	✓	✓	-	-	-
RF380R	✓	✓	✓	✓	✓	✓	✓	-	-	-
RF300 Gen 2 allgemein	✓	✓	✓	-	✓	✓	✓	✓	-	-
RF380R Gen 2	✓	✓	✓	✓	✓	✓	✓	✓	-	-
RF615R [1)]	-	-	-	-	-	-	-	-	-	-
RF680R [1)]	-	-	-	-	-	-	-	-	-	-
RF685R [1)]	-	-	-	-	-	-	-	-	-	-
MV400 [1)]	-	-	-	-	-	-	-	-	✓	-
MV500 [1)]	-	-	-	-	-	-	-	-	✓	-
Allgemeiner Reader	✓	-	-	-	-	-	-	-	-	✓

[1)] Die gerätespezifischen Parameter dieser Geräte werden im WBM der jeweiligen Geräte eingestellt.

Tabelle 7-4 Parameter der Parametergruppe "Reader-Parameter"

Parameter	Parameterwert	Default-Wert	Beschreibung
Übertragungs-geschwindigkeit	19,2 kBd 57,6 kBd 115,2 kBd	115,2 kBd	Die Auswahl ist abhängig vom verwendeten Kommunikationsmodul und Ident-System. Beachten Sie, dass der hier angegebene Wert automatisch aus der Gerätekonfiguration der angeschlossenen Geräte übernommen wird. Mit diesem Parameter stellen Sie die Datenübertragungsgeschwindigkeit zwischen Kommunikationsmodul und Reader ein. Bei angeschlossenem RFID-Reader: Nach der Umstellung der Übertragungsgeschwindigkeit muss der Reader aus- und wieder eingeschaltet werden (Spannung aus/ein). Bei angeschlossenem optischen Lesegerät: Die hier ausgewählte Übertragungsgeschwindigkeit muss mit der in der Firmware des Lesegerätes ausgewählten Übertragungsgeschwindigkeit übereinstimmen.
Anwesenheitskontrolle	An Aus (HF-Feld an) Aus (HF-Feld aus)	An	An = Sobald sich ein Transponder im Antennenfeld des Readers befindet, wird dessen Anwesenheit gemeldet. Aus (HF-Feld an) = Die Anwesenheitsanzeige am FB wird unterdrückt. Die Antenne am Reader ist jedoch eingeschaltet, solange diese nicht durch einen Befehl abgeschaltet wurde. Aus (HF-Feld aus) = Die Antenne wird nur eingeschaltet, wenn ein Befehl abgeschickt wird und schaltet sich danach wieder ab.
Error-LED zurücksetzen	An Aus	An	An = Das Blinken der Error-LED am Kommunikationsmodul wird durch jeden Reset des FBs zurückgesetzt. Aus = An der Error-LED wird immer der letzte Fehler angezeigt. Ein Rücksetzen der Anzeige ist nur durch Ausschalten des Kommunikationsmoduls möglich.
HF-Leistung [1)]	0,50 ... 2,00 0,50 ... 5,00	1,00 1,25	Einstellung für die Ausgangsleistung des Readers. Die auswählbaren Werte sind abhängig vom angeschlossenen Gerät.
Max. Anzahl von Transpondern	1 ... 4	1	Anzahl der erwarteten Transponder im Antennenfeld. Die Auswahl ist abhängig vom angeschlossenen Gerät.
Reader-Betriebsart	Normal-Betrieb P2P-Master P2P-Slave	Normal-Betrieb	Auswahl der gewünschten Reader-Betriebsart.
Transponder-Typ	[1)]	[1)]	Auswahl der verwendeten Transponder-Typen. Die Auswahl ist abhängig vom angeschlossenen Gerät.

Parameter	Parameterwert	Default-Wert	Beschreibung
Programmanwahl bei Reset	Keine Programmänderung 1 ... 15	Keine Programmänderung	Auswahl des Programms das nach einem Neustart des Optischen Lesegeräts verwendet werden soll. Wird der Parameterwert "Keine Programmänderung" eingestellt, wird nach dem Neustart des Geräts das zuletzt aktive Programm gewählt.
Byte-Sequenz des Reset-Parameters [hex]	[1]	[1]	Hexadezimale Darstellung der Byte-Sequenz des Reset-Parameters des Readers.

[1] Eine ausführliche Beschreibung zu diesen Parametern finden Sie in den nachfolgenden Absätzen.

Der Parameter "HF-Leistung"

Die auswählbaren Werte sind abhängig von dem angegebenen Wert in dem Parameter "Reader-Parametrierung".

- RF290R
  - Wertebereich: 0,50 ... 5,00 W
  - Default-Wert: 1,00 W
- RF380R; RF380R Gen 2
  - Wertebereich: 0,50 ... 2,00 W
  - Default-Wert: 1,25 W

Der Parameter "Transponder-Typ"

Auswahl der verwendeten Transponder. Die Auswahl ist abhängig von dem angegebenen Wert in dem Parameter "Reader-Parametrierung". Es können folgende Transponder-Typen ausgewählt werden:

- ISO Mode
- RF300
- MOBY E

Wenn Sie "ISO Mode" auswählen, werden Ihnen abhängig von der Reader-Parametrierung die folgenden Auswahlmöglichkeiten angezeigt:

Reader-Parametrierung	Werte
RF200 allgemein	ISO 15693 allgemein
RF290R	ISO 15693 allgemein ISO 15693 (Infineon, MDS D3xx)

Reader-Parametrierung	Werte
RF300 allgemein RF380R	ISO 15693 allgemein ISO 15693 (Infineon, MDS D3xx) ISO 15693 (Fujitsu - 4 kB, MDS D4xx) ISO 15693 (NXP, MDS D1xx) ISO 15693 (TI, MDS D2xx) ISO 15693 (STM, MDS D261)
RF300 Gen 2 allgemein RF380R Gen 2	ISO 15693 allgemein ISO 15693 (Infineon, MDS D3xx) ISO 15693 (Fujitsu - 4 kB, MDS D4xx) ISO 15693 (NXP, MDS D1xx) ISO 15693 (TI, MDS D2xx) ISO 15693 (STM, MDS D261) ISO 15693 (Fujitsu - 8 kB, MDS D5xx)

Der Parameter "Byte-Sequenz des Reset-Parameters [hex]"

Über diese Funktion können Sie die Reset-Parameter in hexadezimaler Darstellung angeben. Diese Einstellung richten sich ausschließlich an geschulte Anwender.

Folgende Reset-Parameter stehen zur Verfügung:

Byte	1	2...5 [1]	6	7...8	9	10	11	12	13...14	15	16
Wert	0x04	0x00	0x0A	0x00	scan-ning_ time	param	opti-on_1	distance_ limiting	multitag	field_on_ control	field_on_ time

[1] Bei dem Kommunikationsmodul RF180C ab V2.2 in Verbindung mit MOBY U sind Byte 4 mit Kalenderwoche und 5 mit Jahr vorbelegt.

7.1.4 Ident-Anweisung im Anwenderprogramm aufrufen

Vorgehen

Gehen Sie folgendermaßen vor, um die Anweisung im Anwenderprogramm aufzurufen:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Programmbausteine".
3. Doppelklicken Sie auf den OB für die zyklische Programmbearbeitung.
   Der Baustein wird im Arbeitsbereich geöffnet.
4. Öffnen Sie im Fenster "Anweisungen" die Gruppe "Optionspakete" und den Ordner "SIMATIC Ident".
   Der Ordner, sowie die Unterordner enthalten die Anweisungen.
5. Wählen Sie eine Anweisung und ziehen Sie diese per 'Drag & Drop' in Ihren OB.
   Der Dialog "Aufrufoptionen" wird geöffnet.

6. Wählen Sie aus der Liste "Name" einen Instanz-Datenbaustein oder geben Sie den Namen für einen neuen Instanz-Datenbaustein ein.
7. Bestätigen Sie mit "OK".
8. Wählen Sie das projektierte Technologieobjekt und ziehen Sie dieses per 'Drag & Drop' in den Parameter "HW_CONNECT".

Rücksetzen der Reader mit Hilfe des "Ident-Profils" oder des Bausteins "Reset_Reader"

Es gibt verschiedene Methoden, um Reader zurückzusetzen:

- Wenn Sie mit den Ident-Bausteinen arbeiten, setzen Sie alle Reader-Typen mit dem Baustein "Reset_Reader" zurück. Dadurch werden die in dem Technologieobjekt eingestellten Werten an den Reader übertragen.
- Wenn Sie mit Ident-Profil arbeiten, werden beim Setzen des "INIT"-Bits, die in dem Technologieobjekt eingestellten Werten an den Reader übertragen.
- Wenn Sie einen eigenen "WRITE-CONFIG"-Befehl absetzen wollen, müssen Sie anstelle des "INIT"-Bits das "EXECUTE"-Bit verwenden.

Soll während der Laufzeit ein Reset-Parameter des Technologieobjekts geändert werden, kann der entsprechende Wert im Datenbaustein des Technologieobjekts mit Hilfe des Anwenderprogramms überschrieben werden. Die Reset-Parameter befinden sich im Datenbaustein des Technologieobjekts unter "Static.CONFIGURATION.param_buffer[]".

Ausführliche Informationen zum Ident-Profil und den Ident-Bausteinen finden Sie im Handbuch "Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme" auf den Seiten des "Industry Online Support (https://support.industry.siemens.com/cs/ww/de/ps/14971/man)".

7.1.5 Projektierung in CPU laden

Nach dem Konfigurieren müssen Sie die Änderungen aus dem Technologieobjekt bzw. der S7-Steuerung in die angeschlossenen Geräte laden.

Laden in Gerät

Beachten Sie, dass beim Laden in die Geräte zwischen "Software (nur Änderungen)" und "Hardware und Software (nur Änderungen)" unterschieden wird. Wurden ausschließlich Änderungen an der Software vorgenommen, erfolgt das Laden über das Technologieobjekt. Wurden Änderungen an Hard- und Software vorgenommen, erfolgt das Laden über die S7-Steuerung.

ACHTUNG
Laden in Gerät bei Verwendung der Schaltfläche "Auto-Konfiguration" Wenn beim Konfigurieren die Schaltfläche "Auto-Konfiguration" verwendet wurde, müssen Sie die Änderungen über die S7-Steuerung ("Hardware und Software (nur Änderungen)") in die angeschlossenen Geräte laden.

7.2 Ident-Systeme diagnostizieren

Nachdem die Konfiguration des Technologieobjekts abgeschlossen ist, können Sie mithilfe der Diagnosefunktion die im Technologieobjekt konfigurierten Geräte überwachen.

Voraussetzungen

- Es besteht eine Online-Verbindung zwischen STEP 7 (TIA Portal) und S7-Steuerung.
- Die Ident-Anweisung wird im Anwenderprogramm aufgerufen.
- Ein Technologieobjekt wurde angelegt und in die S7-Steuerung geladen.
- Die S7-Steuerung ist im Betriebszustand "RUN".

Vorgehen

Gehen Sie folgendermaßen vor, um den Anzeige-Editor für die Diagnosefunktionen zu öffnen:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das betreffende Technologieobjekt.
3. Doppelklicken Sie auf das Objekt "Diagnose".
4. Klicken Sie auf das Symbol "Alle Beobachten", um den Beobachtungsmodus zu aktivieren.

7.2.1 Baustein-Diagnose

In der Baustein-Diagnose können Sie sich die zuletzt aufgetretenen Fehlermeldungen anzeigen lassen, die bei der Befehlsausführung aufgetreten sind.

Folgende Werte werden vom Technologieobjekt gelesen und angezeigt:

Tabelle 7-5 Angezeigte Werte der "Baustein-Diagnose"

Angezeigte Werte	Beschreibung
Datum und Uhrzeit	Zeigt das Datum und die Uhrzeit an, wann der Fehler aufgetreten ist.
Befehl	Zeigt den Befehl an, bei dem der Fehler aufgetreten ist.
Fehlerbeschreibung	Zeigt eine kurze Beschreibung des aufgetretenen Fehlers an.
	Hinweis: In der Tabelle werden die letzten fünf Fehlermeldungen angezeigt und beschrieben. Wird eine neue Fehlermeldung angezeigt, werden die bisher angezeigten Fehlermeldungen eine Zeile nach unten geschoben. Es werden maximal die letzten fünf Fehlermeldungen angezeigt.
Fehlermeldung	Zeigt die Fehlermeldung des aufgetretenen Fehlers an.
Fehlerbeschreibung	Zeigt eine detailliertere Beschreibung des aufgetretenen Fehlers an.
Datum und Uhrzeit	Zeigt das Datum und die Uhrzeit an, wann der Fehler aufgetreten ist.
Befehlscode	Zeigt den Befehlscode des Befehls an, bei dem der Fehler aufgetreten ist.
Befehl	Zeigt den Befehl an, bei dem der Fehler aufgetreten ist.
Attribut	Zeigt das Attribut des Befehls an, bei dem der Fehler aufgetreten ist.

7.2.2 Hardware-Diagnose

In der Hardware-Diagnose können Sie sich die Status-Parameter des ausgewählten Readers und des sich aktuell im Antennenfeld befindlichen Transponders anzeigen lassen.

Monitoring-Status

In diesem Bereich können Sie die Reader- und Transponder-Parameter aktualisieren und die Fehlerzähler-Stände des Readers zurücksetzen.

Reader-Status

Folgende Parameter werden vom Technologieobjekt gelesen und angezeigt:

Tabelle 7-6 Angezeigte Parameter des Bereichs "Reader-Status"

Angezeigte Parameter	Beschreibung	
Hardware-Typ	Hardware-Typ des Readers	
Hardware-Version	Hardware-Version des Readers	
Bootloader-Version	Bootloader-Version des Readers	
Firmware-Version	Firmware-Version des Readers	
Treiber-Typ	Treiber-Typ der seriellen Schnittstelle des Readers	
Treiber-Version	Treiber-Version der seriellen Schnittstelle des Readers	
Schnittstelle	Verwendete serielle Schnittstelle des Readers Mögliche Werte: RS232, RS422	
Übertragungsgeschwindigkeit	Verwendete Übertragungsgeschwindigkeit des Readers Mögliche Werte: 19,2; 57,6; 115,2 kBd	
Distance limiting	Eingestellte Sendeleistung des Readers (RF380R) Folgende Werte sind möglich:	
	Wert	Bedeutung
	02	0,5 W
	03	0,75 W
	04	1,0 W
	05	1,25 W
	06	1,5 W
	07	1,75 W
	08	2,0 W
Max. Transponder-Anzahl	Maximale Anzahl der zu erwartenden Transponder, die sich zeitgleich im Antennenfeld des Readers befinden dürfen.	

Angezeigte Parameter	Beschreibung	
Transponder-Typ	Eingestelltes Transponder-Typen-Profil Folgende Werte sind möglich:	
	Wert	Bedeutung
	00	RF300 (RF3xxT)
	01	ISO 15693 allgemein
	03	ISO 15693 (Infineon, MDS D3xx)
	04	ISO 15693 (Fujitsu - 2 kB, MDS D4xx)
	05	ISO 15693 (NXP, MDS D1xx)
	06	ISO 15693 (TI, MDS D2xx)
	07	ISO 15693 (STM, MDS D261)
	08	ISO 15693 (Fujitsu - 8 kB, MDS D5xx)
	10	RF300 (RF3xxT)
	20	ISO 14443 (MOBY E, E6xx)
	31	General Mode
	40	P2P-Master
	41	P2P-Master & Mischbetrieb mit ISO-Transponder (MDS Dxxx)
	50	P2P-Master & Mischbetrieb mit RF300-Transponder
	4F	P2P-Slave
Antennenstatus	Status der Antenne	
Anwesenheitskontrolle	Eingestelltes Anwesenheitskontroll-Profil	

Transponder-Status

Folgende Parameter werden vom Technologieobjekt gelesen und angezeigt:

Tabelle 7-7 Angezeigte Parameter des Bereichs "Transponder-Status"

Angezeigte Parameter	Beschreibung
UID	Unique Identifier des Transponders
Transponder-Typ	Transponder-Typ (Hersteller, Bezeichnung)
Version	Version des Transponder-Chips
Lock-Status des OTP-Bereichs	Gesperrte Blöcke des OTP-Bereichs auf dem Chip
Größe des Anwenderspeichers	Speichergröße des Anwenderspeichers in Byte
Größe eines Speicherblocks	Größe der Speicherblöcke des Transponder-Chips
Anzahl der Speicherblöcke	Anzahl der Speicherblöcke des Transponder-Chips
Messwert Leistungsflussdichte	Strahlungsleistung die beim Transponder ankommt. Je niedriger der Wert ist, umso mehr Leistung erhält der Transponder.
Passiver Fehlerzähler	Anzahl der aufgetretenen Ruhefehler
Aktiver Fehlerzähler	Anzahl der aufgetretenen Fehler Summe der Signatur- und CRC-Fehler
Anwesenheitszähler	Dauer in [ms] die der Transponder sich im Antennenfeld befunden hat.

Fehlerzähler

Folgende Fehlertypen werden vom Technologieobjekt gelesen und angezeigt:

Tabelle 7-8 Angezeigte Fehlertypen des Bereichs "Fehlerzähler"

Angezeigte Fehlertypen	Beschreibung
FZP	Passiver Fehlerzähler (Ruhefehlerzähler) Dieser Fehlerzähler ist ein Indikator für eine gestörte Umgebung (Interferenzen). Ausschließlich für den Transponder-Typ "RF300" relevant.
ABZ	Abbruchszähler Zähler für Protokollfehler auf der Luftschnittstelle, bei denen der Transponder die Kommunikation abgebrochen hat.
CFZ	Codefehlerzähler Zähler für Störungen oder Kollisionen auf der Luftschnittstelle, durch die die Kommunikation gestört wurde.
SFZ	Signaturfehlerzähler Zähler für fehlgeschlagene Signaturverschlüsselungen von geschriebenen Datenblöcken
CRCFZ	CRC-Fehlerzähler Zähler für fehlgeschlagene CRC-Überprüfungen
ASMFZ	Fehlerzähler für Schnittstellenprobleme zum Host (CM/PC) Zähler für Fehler auf der seriellen Schnittstelle.
Signalstärke	
AMLI	AM-Leistungsindikator Empfangssignalstärke [dB] für die Amplituden-Modulation mit der der Transponder erkannt wurde.
PMLI	PM-Leistungsindikator Empfangssignalstärke [dB] für die Phasen-Modulation mit der der Transponder erkannt wurde.

Hinweis

Fehlerzähler-Stände

Die angezeigten Fehlerzähler-Stände werden ab dem letzten Neustart des Readers bzw. dem letzten manuellen Zurücksetzen der Fehlerzähler-Stände gezählt.

Index

1

1Count24V, 325
1Count5V, 389
1PosU
Abschaltdifferenz, 2713, 2719, 2725
Abschaltdifferenz ändern (JOB 3), 2691
Achsart, 2710, 2716, 2723
Aktivieren von JOBs, 2670
Aktuelle Werte anzeigen (JOB 15), 2698
Ansteuerung des Antriebs, 2671, 2713, 2719, 2725
Anzahl Geberumdrehungen, 2722
Auswertung der Gebersignale, 2671
Baudrate, 2721
CPU/Master-STOP, 2707
Dosierbetrieb, 2668, 2709, 2715
Drehrichtungsüberwachung einstellen (JOB 11), 2697
Drehrichtungsumkehr, 2710, 2715, 2722
Ende der Rundachse, 2710, 2716
Externe Fehler, 2701
Fehler bei der Steuerung von MODEs und JOBs, 2704
Fehlerquittung EXTF_ACK, 2704
Fehlerursachen für JOB_ERR, 2706
Fehlerursachen für POS_ERR, 2705
Funktionen, 2667
Geberart, 2721
Geberbereich verschieben (JOB 2), 2690
Gebereingänge, 2715
Hardware-Endschalter, 2711, 2717, 2723
Inkremente pro Geberumdrehung, 2721
Istwert setzen (JOB 1), 2688
JOB-Bearbeitung abbrechen (JOB 0), 2687
Parametrierfehler, 2701
Positionieren absolut (MODE 5), 2683
Positionieren relativ (MODE 4), 2681
Reduziernocken, 2712, 2718, 2724
Referenziersignal, 2712, 2718
Referenziersignal auswerten (JOB 9), 2693
Referenzpunktfahrt (MODE 3), 2675
Referenzschalter, 2712, 2718
RESET-Zustand, 2708
richtungsabhängige Freigaben, 2672
Rückmeldeschnittstelle, 2728
Starten von MODEs, 2669
Startrichtung für Referenzpunktfahrt, 2713, 2719
Steuerschnittstelle, 2726
Stoppen (MODE 0), 2673
Tippen (MODE 1), 2673
Tmin Richtungswechsel, 2714, 2720, 2726
Umschaltdifferenz, 2714, 2720, 2725
Umschaltdifferenz ändern (JOB 4), 2692
1SSI
Geberwerterfassung, 2430, 2446, 2458
1STEP 5V
Achstyp, 2751
Ausgangsfrequenz/Geschwindigkeit Fa, 2735
Basisfrequenz, 2736
Beschleunigung/Verzögerung a, 2736
Digitaleingang DI0 (3), 2755
Digitaleingang DI1 (7), 2756
Drehzahlbetrieb, 2748
Fahrauftrag anhalten, 2749
Impulsfreigabe, 2753
Maximale Frequenz / Geschwindigkeit der Achse Fmax, 2735
Moduloachse, 2751
Parameter ändern im Betrieb, 2754
Referenzpunkt setzen, 2745
Referenzpunktfahrt, 2740
Referenzpunktfahrt, Ablauf, 2741
Schrittmaßfahrt absolut, 2747
Schrittmaßfahrt relativ, 2746
Starten eines Fahrauftrags, 2739
Start-Stop-Frequenz Fss, 2734
Verfahrkurve, 2734
Verhalten bei CPU/Master-STOP, 2756

2

24 V-Gebersignale, 209
Eingangsfilter, 226, 227, 305
2PULSE
Direktes Steuern des DO, 1174
Ein-/Ausschaltverzögerung, 1159
Frequenzausgabe, 1167
Impulsausgabe, 1137
Impulskette, 1153
Parallelschaltung beider Kanäle, 1177
Pulsweitenmodulation, 1143
Sequenzzähler, 1173
Strommessung, 1173

5

5 V-Gebersignale, 208
Überwachung, 209

6

611U-Konformmodus, 2781

A

Abbrechen, 2487, 2579
Referenzpunktfahrt, 2495, 2587
Schrittmaßfahrt, 2503, 2596
Tippen, 2490, 2582
Abbrechende Torfunktion, 206
Ablauf
Bezugspunkt setzen, 2506, 2601
Fliegendes Istwert setzen, 2597
Geberdaten lesen, 2511, 2609
Istwert setzen, 2504, 2597
Kantenerfassung, 2603
Längenmessung, 2603
Nullpunktverschiebung, 2599
Positionsdaten lesen, 2510, 2608
Rückmeldesignale für die Diagnose, 2512, 2610
Rückmeldesignale für die Positionierung, 2511, 2609
Schleifenfahrt, 2508, 2605
Schrittmaßfahrt mit Schrittmaßnummer 1-100, 2498, 2591
Schrittmaßfahrt mit Schrittmaßnummer 252, 2591
Schrittmaßfahrt mit Schrittmaßnummer 254, 2499, 2592
Schrittmaßfahrt mit Schrittmaßnummer 255, 2500, 2593
Tippen, 2488, 2580
Ablösen einer Fahrt
Easy Motion Control, 2796, 2829
Abschaltdifferenz, 2467, 2520, 2618
Abschaltdifferenz minus, 2521, 2619
Abschaltdifferenz plus, 2521, 2619
Abschaltpunkt, 2467
Absoluter Istwert, 1696, 1697
Absolutwertgeber, 22, 462, 2469
Impulsauswertung, 465, 2471
Inkremente pro Geberumdrehung, 2534, 2633
Mithören, 465
Monoflopzeit, 466, 2472
Reaktionszeiten, 466, 2472
Telegrammlaufzeiten, 466, 2472
Absolutwertgeberjustage, 504, 582, 1730, 1746, 2529, 2627
Beispiel, 505, 583, 2530, 2628
ermitteln, 504, 582, 2529, 2627
Mechanische Justage, 505, 583, 2530, 2628
Absteuern, 2486, 2578
Schrittmaßfahrt, 2502, 2595
Tippen, 2489, 2581
Achsart, 2525, 2623
Achsarten, 500, 578
Achs-Datenbaustein
Laden, 2832
Speichern, 2832
Achs-DB
Aufgabe, 2831
Achse
Absolutwertgeberjustage, 504, 582, 2529, 2627
Achsart, 2525, 2623
Achsarten, 500, 578
Art der Referenzpunktfahrt, 2527, 2625
Ende der Rundachse, 2527, 2625
Hysterese, 507, 585
Minimaler Kantenabstand, 2630
Referenzpunktkoordinate, 503, 581, 2528, 2626
Simulationsgeschwindigkeit, 509, 587
Software-Endschalter, 500, 578, 2525, 2623
Startgeschwindigkeit bei Referenzpunktfahrt, 2532, 2630
Achse fährt ohne Fahrauftrag
Easy Motion Control, 2841
Achse steht
Easy Motion Control, 2840
Achssteuertafel, 2020, 2026, 2069
Achstyp, 1689
Additives Sollmoment, 1726
AEV-Modell
AEV verstehen, 1200
Aktives Referenzieren, 1729, 1732, 1736, 1737, 1883
Aktivieren
Maschinendaten, 467, 2478, 2570
Aktor, 1683
Aktualisierungszeit, 51, 190
Aktualwerte
Easy Motion Control, 2833
Ändern
Maschinendaten, 2478, 2570
Schrittmaßtabellen, 2480, 2572
Anschlussbelegung
Frontstecker der FM 350-1, 212

Frontstecker der FM 350-2, 266
Frontstecker der FM 351, 2474
Frontstecker der FM 352, 472
Frontstecker der FM 450-1, 294
Frontstecker der FM 451, 2565
Frontstecker der FM 452, 549
Frontstecker der SM 338, 2434
Geberschnittstelle der FM 351, 2473
Geberschnittstelle der FM 352, 471
Geberschnittstelle der FM 451, 2565
Ansteuerart 1, 2516, 2614
Ansteuerart 2, 2516, 2614
Ansteuerart 3, 2517, 2615
Ansteuerart 4, 2518, 2616
Antrieb
Abschaltdifferenz, 2520, 2618
Ansteuerart 1, 2516, 2614
Ansteuerart 2, 2516, 2614
Ansteuerart 3, 2517, 2615
Ansteuerart 4, 2518, 2616
Kompatibilitätsliste, 2222
Maschinendaten, 2515, 2613
Stillstandsbereich, 2523, 2621
Stillstandsgeschwindigkeit, 2524, 2622
Überwachungszeit, 2523, 2621
Umschaltdifferenz, 2519, 2617
Zielbereich, 2521, 2619
Antriebsanbindung S7-1200 Motion Control, 1353
Antriebsanbindung S7-1500 Motion Control, 1682, 1692, 1693, 1698, 1709, 1835, 1836, 1838, 1841, 1859
Antriebtyp, 2772, 2773, 2780
Antriebtyp Servo, 2773
Antriebtyp Stepper, 2773
Anwenderdefinierte Kinematik, 2314
Anwenderprogramm, (Synchronous Cycle)
Fehlerauswertung, 537, 617, 2550, 2649
Anwendertransformation, 2320
Beispiel, 2324
Anwendungsbeispiele Time-based IO, 1193
Anwendungsgebiet, 2771
Applikationszyklus, 1200, 1205
Arbeitsbereich, 2467
Arbeitsbereichsgrenze
Tippen, 2490, 2582
Arbeitsraumzonen, 2333, 2335
Arbeitszonen, 2333, 2335
Art der Referenzpunktfahrt, 2527, 2625
Arten
Referenzpunktfahrt, 2495, 2587
Auflösung, 513, 591, 2535, 2633
Wertebereich, 513, 591, 2536, 2634
Auflösung der Periodendauer, 222, 363, 424
Aufrufreihenfolge, 1205
Auftrag
Bezugspunkt setzen, 2505, 2601
Fliegendes Istwert setzen, 2596
Istwert setzen, 2503, 2596
Istwert setzen rückgängig, 2505, 2598
Nullpunktverschiebung, 2598
Auftragskette, 2226, 2361
Auslösen
Diagnosealarm, 1093
Auswertung
Diagnosealarme, 2551, 2650
Auswirkungen
Bezugspunkt setzen, 2506, 2602
Istwert setzen, 2504, 2597
Nullpunktverschiebung, 2599
Referenzpunktfahrt, 2495, 2587
AxesGroup, 2226

B

Baudrate, 2447, 2459
Bedienfehler, 266, 325
Belegung
Diagnosedatensatz DS 1 der FM 355-2, 1096
Diagnosedatensatz DS0, 1095
Betriebsart, 1317, 1327, 1334
Digitalausgang, 1212, 1215, 1222
Digitaleingang, 1218
Dosieren, 200
Drehzahlmessung, 196
Einmaliges Zählen, 186
Endlos Zählen, 185
Frequenzmessung, 195
High_Speed_Counter, 97
Periodendauermessung, 198
Periodisches Zählen, 187
Referenzpunktfahrt, 2491, 2583
Schrittmaßfahrt, 2498, 2590
Tippen, 2488, 2580
Betriebsart Ein-/Ausschaltverzögerung, 1258
Ausschaltverzögerung, 1263
Eingangs- und Ausgangssignale, 1267
Einschaltverzögerung, 1263
Mindestimpulsdauer, 1260
Neuauslösen der Ausschaltverzögerung, 1262
Neuauslösen der Einschaltverzögerung, 1262
Parameter, 1264
Steuer- und Rückmeldesignale, 1264
Taktsynchroner Betrieb, 1264
Wahrheitstabelle, 1260

Betriebsart Frequenzausgabe, 1267
Ausgabewert (Frequenz), 1270
Eingangs- und Ausgangssignale, 1274
Einschaltverzögerung, 1270
Parameter, 1271
Steuer- und Rückmeldesignale, 1271
taktsynchroner Betrieb, 1270
Wahrheitstabelle, 1269
Betriebsart Gleichstrommotor, 1274
Eingangs- und Ausgangssignale der Betriebsart PWM, 1283
Einstellen und Ändern der Impulseinschaltdauer und Drehrichtung, 1277
Einstellen und Ändern der Periodendauer, 1278
Parameter der Betriebsart Gleichstrommotor, 1279
Steuer- und Rückmeldesignale der Betriebsart PWM, 1280
taktsynchroner Betrieb, 1278
Wahrheitstabelle, 1276
Betriebsart Impulsausgabe (Einzelimpuls), 1229
Eingangs- und Ausgangssignale, 1236
Einschaltverzögerung, 1232
Impulsdauer, 1232
Parameter, 1233
Sequenzzähler, 1233
Steuer- und Rückmeldesignale, 1234
Wahrheitstabelle, 1231
Betriebsart Impulskette, 1248
Eingangs- und Ausgangssignale, 1257
Einschaltdauer, 1252
Einschaltverzögerung, 1252
Impulsanzahl, 1251
Periodendauer, 1251
Steuer- und Rückmeldesignale, 1255
Strommessung, 1253
Wahrheitstabelle, 1250
Betriebsart Impulsweitenmodulation (PWM), 1237
Eingangs- und Ausgangssignale, 1248
Einschaltdauer, 1240
Einschaltverzögerung, 1243
Einstellen der Mindestimpulsdauer, 1243
Mindestimpulsdauer, 1239
Parameter, 1244
Periodendauer, 1241
Steuer- und Rückmeldesignale, 1245
Strommessung, 1244
Stromregelung, 1244
Taktsynchroner Betrieb, 1241
Wahrheitstabelle, 1239
Betriebsarten
Ein-/Ausschaltverzögerung, 1258
Frequenzausgabe, 1267
Gleichstrommotor, 1274
Impulsausgabe (Einzelimpuls), 1229
Impulskette, 1248
Impulsweitenmodulation (PWM), 1237
Übersicht, 1226
Betriebsmodus, 135
Betriebszustand S7-1500 Motion Control, 1760
Bewegungsübergang, 2364
Bezugspunkt setzen, 480, 557, 2505, 2601
Ablauf, 480, 558, 2506, 2601
Auswirkungen, 481, 558, 2506, 2602
Besonderheiten Absolutwertgeber, 481, 559
Daten im Kanal-DB, 2506, 2601
Definition, 480, 557
Synchronisation, 2505, 2601
verwendete Daten im Kanal-DB, 481, 558
Voraussetzung, 480, 558, 2506, 2601
Bremsnockenspur, 457

C

Capture, 21, 22, 28, 31, 93, 115, 149, 162
CircMode, 2328
CM 4xIO-Link, (IO-Link Device)
CPU-STOP, 132, 1318, 1335
CPU-Zykluszeit, 1193

D

Daten im Kanal-DB
Bezugspunkt setzen, 2506, 2601
Fliegendes Istwert setzen, 2597
Freigabeeingang, 2510, 2608
Geberdaten, 2511, 2609
Istwert setzen, 2504, 2597
Istwert setzen rückgängig, 2505, 2598
Kantenerfassung, 2604
Längenmessung, 2604
Maschinendaten, 2479, 2571
Nullpunktverschiebung, 2599
Positionsdaten, 2511, 2608
Referenzpunktfahrt, 2494, 2586
Rückmeldesignale für die Diagnose, 2513, 2610
Rückmeldesignale für die Positionierung, 2512, 2609
Schleifenfahrt, 2508, 2605
Schrittmaßfahrt, 2501, 2503, 2594, 2596
Schrittmaßtabellen, 2480, 2573
Tippen, 2489, 2581

Daten im Parameter-DB
Ende eines Positioniervorgangs, 2488, 2580
Freigabeeingang, 2608
Kantenerfassung, 2604
Referenzpunktfahrt, 2495, 2587
Schrittmaßfahrt, 2502, 2595
Datenaktualisierung, (PIP_Mode)
Datenfehler, 264, 324
Datenprüfung, 485, 562
durch die Baugruppe, 485, 562
Datentyp
DB_Any, 1999
DB_Any, 1999
Defaulteinstellung
Diagnosealarm, 1094
Default-Einstellung, 1203
Definition
Diagnosealarm, 1093
Delta-Picker
2D, 2285
2D mit Orientierung, 2288
3D, 2291
3D mit Orientierung, 2294
Determinismus, (Vorhersagbarkeit)
Diagnose
Easy Motion Control, 2797, 2843
High_Speed_Counter, 106
S7-1500 Motion Control, 2033, 2037
SSI_Absolute_Encoder, 128
Technologiemodul, 169, 181, 1223, 1325, 1332
Diagnosealarm, 61, 261, 288, 320
auslösen, 1043, 1044, 1093, 1126
bei Fehlern, 1126
Defaulteinstellung, 1044, 1094, 1127
Definition, 1044, 1093
Freigabe, 134, 1211
freigeben, 1044, 1094, 1127
OB 82, 1044, 1094, 1127
was ist ein Diagnosealarm?, 1093, 1126
Diagnosealarme, 535, 615, 2548, 2647
Auswertung, 2551, 2650
Diagnosealarme freigeben, 1328
Diagnosedatensatz, 1044, 1045, 1094, 1127
Diagnosedatensatz DS0
Belegung, 1046, 1095, 1129
Diagnosedatensatz DS1
Belegung, 1047, 1096, 1130
Diagnosedatensatz DS1 der FM 355-2
Belegung, 1096
Diagnose-DB
Aufbau, 538, 618, 2552, 2651
Daten, 538, 618, 2552, 2651
Diagnoseereignisse, 537, 617, 2552, 2651
Diagnosetext, 1045, 1095, 1128
Digitalausgabemodul
Leistungsmerkmale, 1225
Digitalausgang
Betriebsart, 1212, 1215, 1222
Funktionen, 98, 118, 153, 163
High-Speed-Ausgang, 1212
Vergleichswert, 40, 98, 118, 153, 163
Digitalausgänge
freigeben, 234, 244, 313
setzen und rücksetzen, 234
Verhalten bei Messbetriebsarten, 244
Verhalten bei Zählbetriebsarten, 233, 312
Digitalein-/Digitalausgabemodul
Leistungsmerkmale, 14
Digitaleingabemodul
Einsatzmöglichkeiten, 74
Hardware-Konfiguration, 176
Leistungsmerkmale, 11, 13
Digitaleingang
Betriebsart, 1218
Funktionen, 93, 115, 149, 162
Konfiguration, 1219
Digitaleingänge
Eingangsfilter, 227, 305
Direktansteuerung eines Digitalausgangs, 1295
Direktes Referenzieren, 1730, 1746
Dosieren, 200, (Anwendungsbeispiele Time-based IO)
Drahtbruch, 134, 515, 593, 1048, 1097, 1132, 2539, 2637
Drehkonventionen, 2235
Drehrichtungssignal, 2773
Drehzahlachse
Diagnose, 2038, 2041, 2077, 2079
Funktionen, 1684
Grundlagen, 1762
hinzufügen, 1843
Konfiguration, 1847
kopieren, 1845
löschen, 1845
Variablen, 2085
Drehzahlmessung, 196
DSC (Dynamic Servo Control), 1753, 1754, 1755, 1894
Dynamic Servo Control (DSC), 1753, 1754, 1755, 1894
Dynamikgrenzen, 1716, 1878
Dynamik-Voreinstellung, 1717, 1874
Dynamische Verstellung, 459, 526, 604

E

Easy Motion Control
Ablösen einer Fahrt, 2796, 2829
Absolutwertgeberjustage, 2795
Achs-DB, 2831
Achse fährt ohne Fahrauftrag, 2841
Achse schwingt, 2841
Achse steht, 2840
Achs-Parameter, 2833
Aktualwerte, 2833
Arbeitsbereich, 2792
Aufbau des Anwenderprogramms, 2822
Aufrufreihenfolge der Anweisungen, 2824
Bezugspunktsetzen, 2795
Bitfeld für Initialisierung, 2835
Diagnose, 2797, 2843
Fehler mit hartem Stopp, 2848
Fehler mit weichem Stopp, 2845
Fehlerreaktionen, 2797
Gemeinsame Parameter aller Anweisungen, 2827
Geschwindigkeit der Achse, 2841
Geschwindigkeitsmessung, 2837
Geschwindigkeits-Override, 2796
Getriebe, 2796
Hardware-Konfigurationen, 2788
Linearachse, 2792
Multiinstanz-FB, 2835
Nachführen, 2796
NOT-AUS-Schalter, 2836
Parametrierfehler, 2844, 2850
Positionieren absolut, 2796
Positionieren relativ, 2796
Positionierung, 2790
Referenzpunktfahrt, 2795
Restweg, 2791
Rundachse, 2793
Schleppabstand, 2791
Schleppfehler, 2843
Simulation, 2796
Start eines Fahrauftrags, 2828
Stillstandsbereich, 2792
Stoppen, 2796
Synchronisieren, 2795
Tippen, 2795
Überwachungsbereiche, 2792
Verdrahtungstest, 2837
Verfahrbereich, 2792
Wegmessung, 2839
Ziel, 2791
Zielbereich, 2792
Zusammenwirken der Anweisungen, 2822
Easy Motion Cotrol
Aufrufumgebung, 2823
Einfachauswertung, 211
Eingangsfilter, 227, 305
Eingangsverzögerung
Technologiemodul, 1213, 1217, 1220
Eingeben
Telegrammtyp, 2781
Einmaliges Zählen, 186
Einsatzgebiet, 2771
Empty-Zeile einfügen, 1457
Empty-Zeile hinzufügen, 1457
Ende
Positioniervorgang, 2481, 2574
Ende der Rundachse, 2527, 2625
Ende eines Positioniervorgangs
Daten im Parameter-DB, 2488, 2580
Rückmeldesignale im Kanal-DB, 2488, 2580
Endlos Zählen, 185
Endschalter, 1712, 1714, 1876
ENSET_DN, 229, 307
ENSET_UP, 229, 307
ErrorID
Grundlagen, 2033, 2037
Liste der ErrorIDs, 2208, 2426
Ersatzwert, 235, 283
ET 200, 1197, 1202
TM Timer DIDQ, 1197
EVA-Modell
EVA verstehen, 1199
Externe Fehler, 1043, 1093, 1126
Externer Geber
Diagnose, 2049, 2051
Funktionen, 1685
Grundlagen, 1780
hinzufügen, 1843
Konfiguration, 1896
kopieren, 1845
löschen, 1845
Variablen, 2144

F

FCS, 2233
Fehler
externe, 1043, 1093, 1126
interne, 1043, 1093, 1126
Fehler an Motion Control-Anweisungen, 2033, 2037, 2208, 2426

Fehler mit hartem Stopp
Easy Motion Control, 2848
Fehler mit weichem Stopp
Easy Motion Control, 2845
Fehleranzeige, 1043, 1092, 1125
Fehlerauswertung
Anwenderprogramm, 537, 617, 2550, 2649
Fehlercodes
Gx_XIST2, 2785
Fehlerkennung, 2037, 2208, 2426
Fehlerklassen
Bedienfehler, 2556, 2655
Betriebsfehler, 540, 620, 2555, 2654
Datenfehler, 541, 621, 2557, 2656
Diagnosefehler, 546, 626, 2562, 2661
Maschinendatenfehler, 542, 622, 2558, 2657
Meldungen, 546, 626, 2561, 2660
Nockendatenfehler, 544, 624
Schrittmaßtabellen-Fehler, 2560, 2659
Fehler-LEDs, 1125
Fehlerreaktionen
Easy Motion Control, 2797
Fehlimpulse Inkrementalgeber, 516, 594, 2539, 2637
Fiktives Ziel, 2508, 2606
Lage, 2509, 2606
Filterfrequenz, 88, 145
Flanschkoordinatensystem, 2233
Flanschzonen, 2337
Fliegendes Istwert setzen, 475, 552, 2596
Ablauf, 475, 553, 2597
Daten im Kanal-DB, 476, 553, 2597
Voraussetzung, 475, 552, 2596
Fliegendes Messen, 2784
FM 350-1, 212
FM 350-2, 266
Kanaleinstellung, 270
FM 351, 2473
FM 352, 471
FM 355, 1017
FM 355-2, 1050
FM 450-1, 294
FM 451, 2564
FM 452, 549
FM 455, 1099
FNP, 2226
Frame-Definition, 2235
Freigabe
Diagnosealarm, 61
Prozessalarm, 61, 80
Freigabeeingang, 2492, 2510, 2584, 2607
auswerten, 2510, 2607
Daten im Kanal-DB, 2510, 2608
Daten im Parameter-DB, 2608
nicht auswerten, 2510, 2608
Freigeben
Diagnosealarm, 1094
Frequenzmessung, 51, 195
Funktion
Bezugspunkt setzen, 480, 557
Fliegendes Istwert setzen, 475, 552
Istwert setzen, 475, 552
Istwert setzen rückgängig, 475, 552
Nockenflanken ändern, 482, 559
Nullpunktverschiebung, 477, 555
Funktion Schnelle Impulsausgabe, 1283
Funktion Strommessung, 1285
Genauigkeit, 1286
Funktionen
Direktansteuerung eines Digitalausgangs, 1295
PWM-Dither-Ausgabe, 1290
Schnelle Impulsausgabe, 1283
Sequenzzähler, 1284
Strommessung, 1285
Stromregelung, 1287
taktsynchroner Betrieb, 1294
Überblick, 1227

G

Geber
anschließbare, 208
Anzahl Geberumdrehungen, 517, 595, 2537, 2635
Baudrate, 517, 595, 2537, 2636
Impuls-, 208
Inkremental-, 208
Inkremente pro Geberumdrehung, 512, 590, 2534, 2633
Parität, 2538, 2637
Richtung Initiator, 518, 596
Signalart, 510, 588, 2532, 2630
Signalart auswählen, 223, 275, 302
Telegrammlänge, 518, 596, 2538, 2636
Überwachung, 515, 593, 2539, 2637
Weg pro Geberumdrehung, 511, 589, 2534, 2632
Zählrichtung, 515, 593, 2533, 2631
Geberanbauart, 1711, 1869
Geberanbindung S7-1200 Motion Control, 1353
Geberanbindung S7-1500 Motion Control, 1682, 1692, 1693, 1698, 1709, 1835, 1836, 1838, 1841, 1861, 1863
Geberdaten, 2511, 2608
Ablauf beim Lesen, 2511, 2609

Daten im Kanal-DB, 2511, 2609
Voraussetzung, 2511, 2609
Geberistwert
zusätzlicher, 2785
Gebersteuerwort, 2784
Gebertyp, 2772, 2777
SSI, 2778
TTL, 2777
Gebertyp nichtvorhanden, 2772, 2777
Geberwerterfassung
freilaufend, 2430, 2446, 2458
synchron, 2446, 2458
taktsynchron, 2431, 2458
Gelenkstellungsraum, 2317
Genauigkeit, 1195
TIO_Time berechnen, 1201
Geschwindigkeitsmessung, 51
Easy Motion Control, 2837
Geschwindigkeits-Override
Easy Motion Control, 2796
Geschwindigkeitsprofil, 1717
Geschwindigkeitsvorsteuerung, 1753
Getriebe
Easy Motion Control, 2796
Getriebefaktor, 1770, 1771
Getriebegleichlauf, 1770, 1771
Gleichlauf, 1766, 1770, 1771, 1773, 1869
Gleichlaufachse
Diagnose, 2042, 2046, 2048, 2080, 2084
Funktionen, 1684
Grundlagen, 1764
hinzufügen, 1843
kopieren, 1845
löschen, 1845
Variablen, 2102
Grenzwertüberwachung, 193
Grundparameter
High_Speed_Counter, 86
SSI_Absolute_Encoder, 111
Technologiemodul, 132
Gx_STW, 2784, 2785
Gx_XIST2, 2785

H

Hardware- und Software-Endschalter
Funktion, 1367
Hardwareaufbau für Motion Control S7-1200, 1342
Hardware-Endschalter, 1712, 1876
Hardware-Tor, 203
Hardware-Voraussetzungen, 1196
Hauptzählrichtung, 188, 222, 275, 341, 403
High_Speed_Counter, 82
Diagnose, 106
Grundparameter, 86
in Betrieb nehmen, 105
konfigurieren, 84
programmieren, 104
High-Speed-Ausgang, 1212
Hochlaufzeit, 1874, 1878
HW-Tor, 21, 26, 75, 77, 93, 149
flankengesteuert öffnen und schließen, 205
pegelgesteuert öffnen und schließen, 205
Hysterese, 22, 23, 57, 59, 101, 121, 157, 166, 246, 277, 507, 585
Wegnocken, 507, 585
Zeitnocken, 508, 586
Hysteresebereich, 458

I

IM 174
projektieren, 2771
Impulsdauer, 245, 311
Voreinstellung, 245, 311
Wertebereich, 245, 311
Impulsschnittstelle
Prinzip, 1347
Inbetriebnahme
High_Speed_Counter, 105
SSI_Absolute_Encoder, 127
Inbetriebnahme S7-1500 Motion Control, 2018, 2020, 2026, 2027, 2030, 2069, 2074
Initiator, 462, 465
Inkrementalgeber, 462, 2469
Impuls, 464, 2471
Inkrement, 463, 2470
Inkremente pro Geberumdrehung, 2534, 2633
Reaktionszeiten, 464, 2471
Signalformen, 463, 2470
Inkrementeller Istwert, 1696, 1697
Integrationszeit, 190
Interne Fehler, 1043, 1093, 1126
Internes Tor, 203
Interpolator-OB, 1363, 1365, 1755, 1758
IODD, (IO-Link Device)
IO-Link, 1197, 1198
Ports einstellen, 1203
IO-Link Device, 1197, 1201, 1202, 1203
Istwert S7-1500 Motion Control, 1696, 1697
Istwert setzen, 475, 552, 2503, 2596
Ablauf, 475, 553, 2504, 2597
Auswirkung, 476, 553
Auswirkungen, 2504, 2597

Datan im Kanal-DB, 476, 553
Daten im Kanal-DB, 2504, 2597
Voraussetzung, 475, 552, 2503, 2596
Istwert setzen rückgängig, 475, 552, 2505, 2598
AVALREM_EN, 477, 554
Daten im Kanal-DB, 2505, 2598
Istwertveränderung, 474, 551

J

Jitter, 1201

K

Kanaleinstellung
FM 350-2, 270
Kanalkonfiguration, 1316, 1327
Technologiemodul, 1210
Kantenerfassung, 2602
Ablauf, 2603
Daten im Kanal-DB, 2604
Daten im Parameter-DB, 2604
fehlerhaft, 2604
Voraussetzung, 2603
KCS, 2232
Kinematik, 2225, 2237
Anwendertransformation, 2320
Begriffsdefinition, 2225
Bewegung, 2326
Diagnose, 2383, 2385, 2386
Dynamikadaption, 2330
Dynamikgrenzen, 2330
Dynamikvoreinstellungen, 2329
Dynamikvorgaben, 2331
Frames, 2231
Freiheitsgrade, 2226
Funktionen, 2227
Gelenkstellungsraum, 2317
hinzufügen, 2340
Kinematiksteuertafel, 2370
Konfiguration, 2341
Koordinatensysteme, 2231
kopieren, 2359
Kreisbewegung, 2327
Linearbewegung, 2327
löschen, 2360
Mechanische Kopplungen, 2320
Simulation, 2229
Singuläre Stellungen, 2318
Variablen, 2387
Verschaltungsregeln, 2228
Zonenüberwachung, 2332
Kinematikachsen, 2226
Kinematikbewegung, 2326
Bewegungsübergang, 2364
Bewegungsübergangsdynamik, 2367
Dynamik, 2329
fortsetzen, 2363
Kreisbewegung, 2366
Linearbewegung, 2364
Restweg, 2363
Status, 2362
stoppen, 2363
unterbrechen, 2363
Kinematikkoordinatensystem, 2232
Kinematiknullpunkt, 2226
Kinematiksteuertafel, 2370, 2375
Kinematiktransformation, 2315, 2316
Kinematiktyp
Anwenderdefiniert, 2314
Delta-Picker 2D, 2285
Delta-Picker 2D mit Orientierung, 2288
Delta-Picker 3D, 2291
Delta-Picker 3D mit Orientierung, 2294
Knickarm 2D, 2269
Knickarm 2D mit Orientierung, 2271
Knickarm 3D, 2274
Knickarm 3D mit Orientierung, 2279
Portal 2D, 2239
Portal 2D mit Orientierung, 2241
Portal 3D, 2243
Portal 3D mit Orientierung, 2246
Rollen-Picker 2D, 2250
Rollen-Picker 2D mit Orientierung, 2252
Rollen-Picker 3D (vertikal), 2254
Rollen-Picker 3D mit Orientierung (horizontal), 2260
Rollen-Picker 3D mit Orientierung (vertikal), 2257
SCARA 3D mit Orientierung, 2264
Tripod 3D, 2307
Tripod 3D mit Orientierung, 2311
Zylindrischer Roboter 3D, 2298
Zylindrischer Roboter 3D mit Orientierung, 2302
Kinematiktypen, 2237
Kinematikzonen, 2333, 2336
Knickarm
2D, 2269
2D mit Orientierung, 2271
3D, 2274
3D mit Orientierung, 2279
KNP, 2226

Kompakt-CPU
Leistungsmerkmale, 11, 13, 1225
Kompakt-CPU S7-1500, (Technologiemodul)
Kompatibilitätsliste, 2222
Kompatibilitätsliste Antriebe, 1835
Konfiguration, (Systemumgebung)
Standard-Konfiguration, 1201
Koordinate ermitteln
Nullpunktverschiebung, 477, 555
Kopplungsfaktor, 2320
Kreisbahn
Definition, 2328
Kreisbewegung, 2327
Kurvenscheibe, 1821, 1919
Funktionen, 1685
Grundlagen, 1821
Import/Export, 1955
Interpolation, 1823
Konfiguration, 1919, 1921, 1955
Variablen, 2177
Kurvenscheibeneditor, 1773, 1919, 1921, 1955
Kurvenscheibengleichlauf, 1773

L

Laden
Achs-Datenbaustein, 2832
Ladewert, 202, 228, 306
direkt, 202
vorbereitend, 202
Lageregelung, 1426, 1753, 1754, 1755, 1894, 1895
Lageregler optimieren, 1473, 2027, 2030, 2074
Längenmessung, 2602, (Anwendungsbeispiele Time-based IO)
Ablauf, 2603
Daten im Kanal-DB, 2604
fehlerhaft, 2604
Flankenabstand, 2604
Voraussetzung, 2603
Lastgetriebe, 1710, 1711, 1869
Latch, (Capture)
Latch/Retrigger, 219
Latchen, 218
Lebenszeichenausfälle, 2780
Lebenszeichen-Fehler, 1332
Leitwertkopplung, 1766
Lesen
Maschinendaten, 469, 2478, 2571
Nockendaten, 470
Schrittmaßtabellen, 2480, 2573
Linearachse
Easy Motion Control, 2792
Linearbewegung, 2327
Lineare Achse, 1689

M

Maschinendaten, 2477, 2569
aktivieren, 467
Antrieb, 2515, 2613
Daten im Kanal-DB, 2479, 2571
Freigabeeingang, 520, 598
lesen, 469
schreiben, 467
schreiben, aktivieren, ändern, lesen, 2478, 2570
Maschinendaten des Gebers
Daten im Parameter-DB, 513, 591
Maschinenkoordinatensystem, 2226
Maßeinheit, 1689, 2229
Maßsystem
wählen, 498, 576, 2513, 2611
Max. Frequenz, 2775
MC_TorqueAdditive, 1726
MC_TorqueRange, 1726
MC-Interpolator-OB, 1363, 1365, 1755, 1758
MCS, 2226
MC-Servo-OB, 1363, 1365, 1755, 1758
MC-Transformation-OB, 2322
Mechanik S7-1500 Motion Control, 1710, 1711, 1869
Mechanische Kopplungen, 2320
Meldezonen, 2333, 2335
Messbereich, 52, 55
Messbereichsüberschreitung, 1048, 1097, 1132
Messbereichsunterschreitung, 1048, 1097, 1132
Messen
Drehzahl, 196
Frequenz, 195
Periodendauer, 198
Messen von Zeiten, 232
Messfunktionen, 51
Messintervall, 52, 55
Messtaster
Diagnose, 2052
Funktionen, 1685
Grundlagen, 1781
Konfiguration, 1910
Variablen, 2158
Messumformerstörung, 1048, 1097, 1132
Messzeit, 190
Mindestimpulsbreite, 227, 305
Minimaler Kantenabstand, 509, 587, 2630
Modi, (PIP_Mode)
Modell der Programmbearbeitung wählen, 1199

Modulo, 1690, 1857
Momentengrenzen, 1726
Motion Control CPU S7-1200
Leitfaden, 1371
Motion Control S7-1200
Antriebs- und Geberanbindung, 1353
PROFIdrive, 1353
Telegramm, 1353
Motion Control S7-1500
Ablaufverhalten, 1755, 1757, 1758, 1760
Achstyp, 1689
Antriebs- und Geberanbindung, 1682, 1692, 1693, 1698, 1709, 1835, 1836, 1838, 1841, 1859, 1861, 1863
Diagnose, 2033, 2037
Dynamikvorgaben, 1716, 1717, 1719, 1727, 1874, 1878
Einführung, 1680
Funktionsweise, 1680
Gleichlauf, 1766, 1770, 1771, 1773, 1779, 1869
Inbetriebnahme, 2018, 2020, 2026, 2027, 2030, 2069, 2074
Istwert, 1696, 1697
Konfiguration, 1843, 1845, 1846, 1857, 1919
Laden in CPU, 2017
Leitfaden zum Einsatz, 1687
Maßeinheit, 1689, 2229
Mechanik, 1710, 1711, 1869
Mengengerüst, 1688
Modulo, 1690, 1857
Motion Control-Anweisung, 1682, 1684, 1685, 1993, 1997, 2002
Positionsgrenzen, 1712, 1714, 1715, 1876
Positionsüberwachung, 1749, 1750, 1751, 1892, 1893
PROFIdrive, 1692, 1698
Programmieren, 1988, 1989, 1993, 2002, 2016
Referenzieren, 1729, 1730, 1731, 1732, 1740, 1745, 1746, 1747, 1748, 1882, 1883, 1888
Regelung, 1753, 1754, 1755, 1894, 1895
Technologie-Alarme, 2033, 2183, 2418
Technologie-Datenbaustein, 1682, 1989, 1991, 1992
Technologieobjekt, 1681, 1684, 1685, 1762, 1763, 1764, 1780, 1781, 1794, 1808, 1821, 1843, 1845, 1857, 1919, 2016
Telegramm, 1693, 1698, 1709
Versionen, 1826, 1833, 2340
Motion Control S7-1500T
Inbetriebnahme, 2375
Motion Control-Anweisung S7-1500, 1682
einfügen, 1997
Fehler an Motion Control-Anweisungen, 2033, 2037, 2208
Motion Control-Auftrag beenden, 2015
Motion Control-Auftrag starten, 2002
Motion Control-Auftrag verfolgen, 2004
Parameter, 1993
Übersicht, 1684, 1685
Motion Control-Anweisung S7-1500T
Fehler an Motion Control-Anweisungen, 2426
M-Schalter, 88, 145
Multiinstanz-FB
Easy Motion Control, 2835

N

Nachführen
Easy Motion Control, 2796
NAMUR-Geber, 270
Neuinitialisieren von Technologieobjekten, 1496, 2016
Nocken
Diagnose, 2054
Dynamische Verstellung, 526, 604
Einschaltzeit, 524, 602
Funktionen, 1685
Grundlagen, 1794
Konfiguration, 1913
Mengengerüst, 521, 599
Nocken "gültig", 522, 600
Nockenanfang bei Wegnocken, 523, 601
Nockenanfang bei Zeitnocken, 524, 602
Nockenart, 523, 601
Nockendaten, 521, 599
Nockenende bei Wegnocken, 523, 601
Nockenende bei Zeitnocken, 524, 602
Prozessalarm, 529, 607
Richtungsumkehr, 459
Schaltverhalten, 528, 606
Spurnummer, 522, 600
Variablen, 2165
Vorhaltezeit, 525, 603
Wirkrichtung, 527, 605
Nockendaten
lesen, 470
schreiben, 470
Nockenflanken ändern, 482, 559
Ablauf, 482, 559
Auswirkungen, 482, 560
Definition, 482, 559
Rücknahme, 483, 561
Verwendete Daten im Kanal-DB, 482, 560
Voraussetzung, 482, 559

Nockenspur
Diagnose, 2056, 2057
Funktionen, 1685
Grundlagen, 1808
Konfiguration, 1915
Variablen, 2171
Nockenspuren, 455
Nockensteuerwerk, (Anwendungsbeispiele Time-based IO)
Nockenzykluszeit, 521, 599
Norm. Frequenz, 2775
NOT-AUS-Schalter
Easy Motion Control, 2836
Notstopp-Verzögerung, 1719, 1876
Nulldurchgang, 221, 301
Nullmarke, 1730
Nullpunktverschiebung, 2598
Ablauf, 478, 555, 2599
Auswirkungen, 2599
bei einer Linearachse, 478, 555
bei einer Rundachse, 479, 556
Daten im Kanal-DB, 2599
Definition, 477, 555
Rücknahme, 480, 557, 2601
verwendete Daten im Kanal-DB, 478, 555
Voraussetzung, 2599
ZOFF_EN, 477, 555

O

OB 82
Diagnosealarm, 1044, 1094, 1127
OB Servo, 1365
Obere Zählgrenze, 217, 274, 347, 410
Oberer Grenzwert, 243, 365, 426
Objektkoordinatensystem, 2234
OCS, 2234
Optimierung, 1473
Optimierung S7-1500 Motion Control, 2027, 2030, 2074
Orientierungsbewegung, 2327

P

Parametrierfehler
Easy Motion Control, 2844, 2850
Parametrierung
Kompakt-CPU, 19
Technologiemodul, 19, 132
Passives Referenzieren, 1729, 1740, 1742, 1744, 1888
PathChoice, 2328
Periodendauer
Auflösung, 222, 363, 424
Periodendauermessung, 51, 198
Periodisches Zählen, 187
PID_Temp
Kaskadierung, 790
Mehrzonenanwendungen, 797
PID-Parameter, 1288
PIP_Mode
PIP_Mode einstellen, 1205
Port Configuration Tool, 1197, 1201, 1203
Portal
2D, 2239
2D mit Orientierung, 2241
3D, 2243
3D mit Orientierung, 2246
Ports
Ports konfigurieren, 1203
Positionierachse
Diagnose, 2042, 2046, 2048, 2080, 2084
Funktionen, 1684
Grundlagen, 1763
hinzufügen, 1843
Konfiguration, 1857
kopieren, 1845
löschen, 1845
Variablen, 1544, 1641, 2102
Positionieren absolut
Easy Motion Control, 2796
Positionieren relativ
Easy Motion Control, 2796
Positionierüberwachung, 1424, 1749, 1751, 1892
Positioniervorgang
Ende, 2481, 2574
Positionsdaten, 2510, 2608
Ablauf beim Lesen, 2510, 2608
Daten im Kanal-DB, 2511, 2608
Positionserfassung
mit SSI-Absolutwertgeber, 22
Vergleichswert, 118, 163
Positionsgrenzen, 1712, 1714, 1715, 1876
prellfreier Schalter, 2603
PROFIdrive, 1353, 1692, 1698
PROFINET, 1197, 1201, 1202
PROFINET IRT nutzen, 1198
Programmierbeispiel, 1207
Programmieren, 1205
Projektiersoftware, 1197
Prozessalarm, 61, 80, 247, 283, 316
Freigabe, 141

freigeben, 247, 284, 316
verlorener, 140
Prozessalarme
freigeben, 529, 607
P-Schalter, 88, 145
PULSE REFEED, 2776
PWM
Technologiemodul, 1210
PWM-Dither-Ausgabefunktion, 1290
Amplitude, 1291
Beispielschemen, 1293
Hochlauf und Rücklauf, 1292
Periode, 1292

R

Reaktionszeit, 1196
definierte Reaktionszeiten,
(Anwendungsbeispiele Time-based IO)
minimale Reaktionszeit, 1196
Referenzieren
Referenziermodi, 1369
Referenzieren S7-1500 Motion Control
Absolutwertgeberjustage, 1730, 1746
aktiv, 1729, 1732, 1736, 1737, 1883
direkt, 1730, 1746
fliegend, 1729, 1740, 1742, 1744, 1888
Grundlagen, 1729
Konfiguration, 1882
Nullmarke, 1730
passiv, 1729, 1740, 1742, 1744, 1888
Referenziermodus, 1730, 1731
Referenzmarke, 1730
Referenznocken, 1730
Referenzpunkt, 1731
Umkehrnocken, 1731, 1745, 1885
Referenzmarke, 1730
Referenzmarkensuche, 2784
Referenznocken, 1730
Referenzpunkt, 1731
Referenzpunkt nachtriggern, 502, 580
Referenzpunktfahrt, 2491, 2583
Abbrechen, 2495, 2587
Arten, 2495, 2587
Auswirkungen, 2495, 2587
Daten im Kanal-DB, 2494, 2586
Daten im Parameter-DB, 2495, 2587
Startgeschwindigkeit, 2532, 2630
Synchronisation, 2491, 2583
Voraussetzung, 2492, 2584
Referenzpunktkoordinate, 503, 581, 2528, 2626
Referenzpunktschalter, 2492, 2584
Regelung, 1426, 1753, 1754, 1755, 1894, 1895
Reproduzierbarkeit, 1193
Restart von Technologieobjekten, 1496, 2016
Restweg, 2502, 2595
Restweg löschen, 2503, 2596
RET_VALU-Meldungen, 1049, 1050, 1098, 1099, 1133, 1134
Richtungsausgang und Fahrtrichtung
Zusammenhang, 1348
Richtungsumkehr, 2508, 2606
Richtungsumkehr am Hardware-Endschalter, 1731, 1745, 1885
Rollen-Picker
2D, 2250
2D mit Orientierung, 2252
3D (vertikal), 2254
3D mit Orientierung (horizontal), 2260
3D mit Orientierung (vertikal), 2257
Rotatorische Achse, 1689
Ruckbegrenzung, 1717, 1874, 1878
Funktion, 1368
Rücklaufzeit, 1874, 1878
Rückmeldeschnittstelle, 172
1Count24V für Fast Mode, 387
1Count24V für Messbetriebsarten, 369
1Count24V für Wegerfassen, 380
1Count24V für Zählbetriebsarten, 350
1Count5V für Fast Mode, 449
1Count5V für Messbetriebsarten, 430
1Count5V für Wegerfassen, 442
1Count5V für Zählbetriebsarten, 413
1SSI im Fast Mode, 2464
1STEP 5V, 2767
2PULSE, 1187
FM 350-1 für Messbetriebsarten, 257
FM 350-1 für Zählbetriebsarten, 252
FM 351, 2543
FM 352, 530
FM 451, 2642
FM 452, 609
Rückmeldesignale für die Diagnose, 2512, 2610
Ablauf, 2512, 2610
Daten im Kanal-DB, 2513, 2610
Rückmeldesignale für die Positionierung, 2511, 2609
Ablauf, 2511, 2609
Daten im Kanal-DB, 2512, 2609
Rückmeldesignale im Kanal-DB
Ende eines Positioniervorgangs, 2488, 2580
Rücknahme
Nullpunktverschiebung, 2601

Rücknahme der Funktion
Istwert setzen rückgängig, 477, 554
Rundachse
Easy Motion Control, 2793

S

S7-1500 Motion Control, 1685
S7-1500T, 1685
S7-PCT, (Port Configuration Tool)
Sammelfehler-LED, 1043, 1092
FM 350-1, 260
FM 351, 2547
FM 352, 534
SM 338, 2442
SCARA
3D mit Orientierung, 2264
Schaltbereiche, 2467
Schaltpunkte, 2467
Schaltverhalten
von Zeitnocken, 474, 551
Schleifenfahrt, 2507, 2605
Ablauf, 2508, 2605
Daten im Kanal-DB, 2508, 2605
Voraussetzung, 2507, 2605
Schleppfehlerüberwachung, 1425, 1749, 1750, 1751, 1892
Schnelles Nockenändern
Ablauf, 483, 561
Auswirkungen, 485, 562
Datenprüfung, 485, 562
Definition, 483, 561
Rücknahme, 485, 562
Verwendete Daten im Kanal-DB, 484, 561
Voraussetzung, 483, 561
Schreiben
Maschinendaten, 467, 2478, 2570
Nockendaten, 470
Schrittmaßtabellen, 2480, 2572
Schrittantrieb, 2773
Schrittantrieb mit Gebertyp Stepper Betrieb PULSE REFEED, 2776
Schrittmaße, 2540, 2638
Schrittmaßfahrt, 2498, 2590
Abbrechen, 2503, 2596
absolut, 2498, 2590
Absteuern, 2502, 2595
Daten im Kanal-DB, 2501, 2503, 2594, 2596
Daten im Parameter-DB, 2502, 2595
relativ, 2498, 2590
Schrittmaßnummer 1-100:, 2498, 2591
Schrittmaßnummer 255, 2500, 2593
Schrittmaßnummer252, 2591
Schrittmaßnummer254, 2499, 2592
Voraussetzung, 2498, 2590
Schrittmaßfahrt mit Schrittmaßnummer 252
Ablauf, 2591
Schrittmaßfahrt mit Schrittmaßnummer 254
Ablauf, 2499, 2592
Schrittmaßfahrt mit Schrittmaßnummer 255
Ablauf, 2500, 2593
Schrittmaßnummer 1 bis 100, 2540, 2638
Schrittmaßnummer 1-100, 2498, 2591
Schrittmaßnummer 252, 2591, 2640
Schrittmaßnummer 254, 2499, 2542, 2592, 2640
Schrittmaßnummer 255, 2500, 2542, 2593, 2641
Schrittmaßtabellen, 2479, 2572
ändern, 2480, 2572
Daten im Kanal-DB, 2480, 2573
lesen, 2480, 2573
schreiben, 2480, 2572
Schrittmotor, 1342
Sensor, 1683
Sensortyp, 88, 145
Separator-Zeile einfügen, 1457
Separator-Zeile hinzufügen, 1457
Sequenzzähler, 1284
Servomotor, 1342
Servo-OB, 1363, 1365, 1755, 1758
SET, 227
Shutdown Verzögerungszeit, 2780
Signal N, 88, 145
Signalart, 87, 143, 159, 1327
Signalauswertung, 88, 144, 210
SIMATIC-Funktionen, 1198
Simulation
Easy Motion Control, 2796
Simulationsgeschwindigkeit, 509, 587
SINAMICS V90 PN, 1835
Singuläre Stellungen, 2318
SM 338
adressieren, 2439
Baudrate, 2437
Diagnosealarm, 2436
Diagnosemeldungen, 2442
Taktsynchroner Betrieb, 2435
Software-Endschalter, 500, 578, 1712, 1714, 1876, 2525, 2623
Software-Regler
Projektieren, 651
Software-Tor, 203
Sollmoment, 1726
Sonderspuren, 457

Speichern
Achs-Datenbaustein, 2832
Sperrzonen, 2333, 2335
Spindelsteigung, 1710, 1711, 1869
Spuren
Ansteuerung der Spurausgänge, 519, 597
Bremsnockenspur, 519, 597
Externe Freigabe, 456
freischalten, 456
Oberer Zählwert für Zählnockenspur, 520, 598
Sonderspuren, 519, 597
Spurergebnis, 455
Spursignal setzen, 456
Zählnockenspur, 519, 597
SSI_Absolute_Encoder, 108
Diagnose, 128
Grundparameter, 111
in Betrieb nehmen, 127
konfigurieren, 111
programmieren, 126
SSI-Absolutwertgeber, 112, 159
Start eines Fahrauftrags
Easy Motion Control, 2828
Startdrive, 1835
Startgeschwindigkeit bei Referenzpunktfahrt, 2532, 2630
Startwert, 21, 75, 92, 148, 180
STEP 7, 1197
Steuerschnittstelle, 170
1Count24V für Messbetriebsarten, 367
1Count24V für Wegerfassen, 379
1Count24V für Zählbetriebsarten, 348
1Count5V für Messbetriebsarten, 428
1Count5V für Wegerfassen, 441
1Count5V für Zählbetriebsarten, 411
1SSI im Standard Mode, 2455, 2456
1STEP 5V, 2763
2PULSE (kurz), 1184
2PULSE (lang), 1185
FM 350-1 für Messbetriebsarten, 255
FM 350-1 für Zählbetriebsarten, 249
FM 351, 2544
FM 352, 531
FM 451, 2642
FM 452, 609
Steuerungsaufgaben, 2771
Steuerungshoheit, 2020, 2069
Stillstandsbereich, 2467, 2523, 2621
Stillstandsgeschwindigkeit, 2524, 2622
Überwachung, 2482, 2574
Stillstandssignal, 1426
Stoppen
Easy Motion Control, 2796
Stromregelungsfunktion, 1287
Dithering und Zyklusdauer, 1289
Grenze erreichte Merker, 1289
PID-Funktionsplan, 1287
PID-Parameter, 1288
Rücksetzen des PID-Reglers, 1290
Sollwertregler, 1289
SW_GATE, 25, 77
SW-Tor, 21, 25, 75, 77
öffnen und schließen, 204
Symbol
für Wertevergleich, 658
Symmetrische Geber, 208
SYNC_PI, 1199
SYNC_PO, 1199
Synchronisation, 22, 33, 93, 149
bei Signal N, 38, 93, 149
Bezugspunkt setzen, 2505, 2601
durch Digitaleingang, 36
Referenzpunktfahrt, 2491, 2583
Wegfall, 480, 557, 2600
Synchronisieren
Easy Motion Control, 2795
Synchronous Cycle, 1205
TIO-Anweisungen verstehen, 1199
Systemumgebung, 1196

T

Taktsynchroner Betrieb
Betriebsart Ein-/Ausschaltverzögerung, 1264
Betriebsart Frequenzausgabe, 1270
Betriebsart Gleichstrommotor, 1278
Betriebsart Impulsausgabe (Einzelimpuls), 1233
Betriebsart Impulsweitenmodulation (PWM), 1241
Überblick, 1294
Taktsynchronisierung, 1198
Taktsynchronität, 68, 73, 81
Taktuntersetzung, 1755
TAPP, (Applikationszyklus)
TCP, 2226, 2233
T-CPU, 1685
TCS, 2233
Technologie-Alarme
Grundlagen, 2033
Liste der Technologie-Alarme, 2183, 2418
Technologie-Datenbaustein
auswerten, 1989
Grundlagen, 1682, 1989

Restart-relevante Daten ändern, 1992
StatusWord ErrorWord und WarningWord auswerten, 1991
Variablen des Technologieobjekts Drehzahlachse, 2085
Variablen des Technologieobjekts Externer Geber, 2144
Variablen des Technologieobjekts Gleichlaufachse, 2102
Variablen des Technologieobjekts Kinematik, 2387
Variablen des Technologieobjekts Kurvenscheibe, 2177
Variablen des Technologieobjekts Messtaster, 2158
Variablen des Technologieobjekts Nocken, 2165
Variablen des Technologieobjekts Nockenspur, 2171
Variablen des Technologieobjekts Positionierachse, 2102
Technologiemodul
Eingangsverzögerung, 1213, 1217, 1220
Einsatzmöglichkeiten, 15, 69
Grundparameter, 132
Hardware-Konfiguration, 130, 1209, 1315, 1326, 1333
Kanalkonfiguration, 1210
Leistungsmerkmale, 11, 13, 1225
Online & Diagnose, 169, 181, 1223, 1325, 1332
parametrieren, 132
Projektnavigation, 130
PWM, 1210
Rückmeldeschnittstelle, 172
Steuerschnittstelle, 170
Verhalten bei CPU-STOP, 1211
Technologieobjekt
Datentypen, 1497, 1999
Drehzahlachse, 1684, 1762, 1843, 1845, 1847, 2038, 2041, 2077, 2079
Externer Geber, 1685, 1780, 1843, 1845, 1896, 2049, 2051
Gleichlaufachse, 1684, 1764, 1843, 1845, 1857, 2042, 2046, 2048, 2080, 2084
High_Speed_Counter, 82
Kinematik, 2227, 2340
Kurvenscheibe, 1685, 1821, 1823, 1919, 1955
Messtaster, 1685, 1781, 2052
Mestaster, 1910
Nocken, 1685, 1794, 1913, 2054
Nockenspur, 1685, 1808, 1915, 2056, 2057
Positionierachse, 1684, 1763, 1843, 1845, 1857, 2042, 2046, 2048, 2080, 2084
SSI_Absolute_Encoder, 108
Technologieobjekt Achse
Variable Config, 1641
Variable ErrorBits, 1650
Variable MotionStatus, 1646
Variable StatusBits, 1646
Technologieobjekt Achse und Auftragstabelle
Liste der ErrorIDs und ErrorInfos, 1519, 1604
Technologieobjekt Auftragstabelle
Erweiterte Parameter, 1466
Grundparameter, 1453
Konfiguration Aktiviere Warnungen, 1454
Konfiguration Allgemein, 1453
Konfiguration Auftragstabelle, 1453
Konfiguration Auftragstyp, 1454
Konfiguration Dauer, 1456
Konfiguration Geschwindigkeit, 1455
Konfiguration Nächster Schritt, 1456
Konfiguration Position / Verfahrweg, 1455
Konfiguration Schrittcode, 1456
Konfiguration Verwende Achsparameter von, 1454
Neues Objekt hinzufügen, 1451
Symbole des Konfigurationsfensters, 1452
Variable Command[1...32], 1579, 1678
Variable Config.Command.Command[1...32], 1677
Verwendung, 1449
Werkzeuge, 1450
Technologieobjekt Auftragstabelle: Kontextmenübefehle, 1457
Technologieobjekt Positionierachse
Ablauf aktives Referenzieren, 1420
Ablauf passives Referenzieren, 1422
Aktualisierung der Variablen, 1579, 1677
Erweiterte Parameter, 1597
Grundparameter, 1388, 1594
Hardware- und Softwarekomponenten, 1381
Konfiguration Achsname, 1388, 1594
Konfiguration Anfahr- / Referenzierrichtung, 1418, 1601
Konfiguration Antrieb Bereit, 1597
Konfiguration Antriebsfreigabe, 1597
Konfiguration Antriebssignale, 1392, 1597
Konfiguration Anwendereinheit, 1596
Konfiguration Beschleunigung, 1409
Konfiguration Dynamik Allgemein, 1408
Konfiguration Eingang Referenzpunktschalter, 1417, 1420, 1599, 1600

Konfiguration Einheit der Geschwindigkeitsbegrenzung, 1408
Konfiguration Erlaube Richtungsumkehr am HW-Endschalter, 1600
Konfiguration Hardware - Schnittstelle, 1390, 1594
Konfiguration Hochlaufzeit, 1409
Konfiguration Impulse pro Motorumdrehung, 1400, 1597
Konfiguration Maßeinheit, 1389
Konfiguration Maximale Geschwindigkeit / Start/ Stopp-Geschwindigkeit, 1408
Konfiguration Mechanik, 1400, 1401, 1597
Konfiguration Notstopp-Verzögerung, 1411
Konfiguration PTO und HSC, 1391, 1595
Konfiguration Referenzieren - Passiv, 1419, 1599
Konfiguration Referenzieren Aktiv, 1416, 1600
Konfiguration Referenziergeschwindigkeit, 1418, 1601
Konfiguration Referenzpunktposition, 1418, 1420, 1599, 1602
Konfiguration Referenzpunktverschiebung, 1418, 1601
Konfiguration Richtungssinn invertieren, 1401, 1598
Konfiguration Ruckbegrenzung, 1410
Konfiguration Rücklaufzeit, 1409
Konfiguration Seite des Referenzpunktschalters, 1418, 1420, 1599, 1601
Konfiguration Verrundungszeit, 1410
Konfiguration Verzögerung, 1409
Konfiguration Weg pro Motorumdrehung, 1400, 1598
Konfigurationsparameter Dynamik im Anwenderprogramm ändern, 1414
Konfigurationsparameter Referenzieren im Anwenderprogramm ändern, 1423
Neues Objekt hinzufügen, 1386
Symbole des Konfigurationsfensters, 1387
Übersicht Diagnose, 1385
Übersicht Inbetriebnahme, 1385
Übersicht Konfiguration, 1385
Variable ActualPosition, 1544
Variable ActualVelocity, 1544
Variable DynamicDefaults, 1559, 1662
Variable DynamicLimits, 1559, 1661
Variable ErrorBits, 1574, 1675
Variable FollowingError, 1564, 1666
Variable Homing, 1563, 1664
Variable Mechanics, 1557, 1660
Variable Modulo, 1558, 1661
Variable Position, 1544, 1652
Variable PositionControl, 1564, 1665
Variable PositionLimitsHW, 1561, 1663
Variable PositionLimitsSW, 1560, 1663
Variable PositionMonitoring, 1565, 1666
Variable Sensor, 1550, 1656
Variable Simulation, 1545
Variable StandstillSignal, 1566, 1667
Variable StatusBits, 1569, 1670
Variable StatusDrive, 1567, 1668
Variable StatusPositioning, 1567, 1668
Variable StatusSensor, 1568, 1669
Variable Units, 1557, 1660
Variable Velocity, 1544
Variablen Actor, 1545, 1653
Verhalten bei aktivierter Ruckbegrenzung, 1413
Werkzeuge, 1384
Technologieobjekte, 1681
CONT_C, 804
CONT_S, 808
PID_3Step, 719
PID_Compact, 681
PID_CP, 834
PID_ES, 883
PID_Temp, 758
TCONT_CP, 810
TCONT_S, 831
Teilprozessabbild
PIP, 1205
TPA1, 1202
Teilprozessabbild "TPA OB Servo", 1757
Telegramm S7-1200 Motion Control, 1353
Telegramm S7-1500 Motion Control, 1693, 1698, 1709
Telegrammaufbau, 2782
Telegrammfehler, 516, 594, 2539, 2637
Telegrammtyp
eingeben, 2781
TIA-Welt, 2771
Time-based IO
Default-Einstellung verwenden, 1203
SIMATIC-Funktionen kennen, 1198
Time-based IO anwenden, 1207
Time-based IO einstellen, 1202
TIO_SYNC
synchronisierte TIO-Module, 1199
TIO_Time, 1194, 1204
TIO_Time verstehen, 1198
TIO-Anweisungen, 1198
TIO-Anweisungen aufzählen, 1204
TM Timer DIDQ, 1198
TIO-Modul, 1197, 1202

Tippen, 2488, 2580
Abbrechen, 2490, 2582
Ablauf, 2488, 2580
Absteuern, 2489, 2581
Arbeitsbereichsgrenze, 2490, 2582
Daten im Kanal-DB, 2489, 2581
Easy Motion Control, 2795
Überwachungen, 2490, 2582
Voraussetzung, 2488, 2580
TM_CTRL_DQ-Signal, 1295
TO_PositioningAxis, 1386
Tolerierbare Lebenszeichenausfälle, 2780
Tor
Hardware-, 203
internes, 203
Software-, 203
Torfunktion
abbrechende, 206
unterbrechende, 206
Torfunktion auswählen, 185
Torsteuerung, 21, 25, 51, 75, 76, 93, 149
bei Taktsynchronität, 206
Torsteuerung auswählen, 186, 187, 195, 197, 198, 200
Torstoppfunktion, 204
Transformation-OB, 2322
Transformationsbereich, 2316
Tripod
3D, 2307
3D mit Orientierung, 2311

U

Überlastungsbereich, 2776
Überwachung
Stillstandgeschwindigkeit, 2482, 2574
Zielbereich, 2482, 2574
Überwachung der 5 V-Gebersignale, 209
Überwachungen, 2482, 2574
Tippen, 2490, 2582
Überwachungszeit, 2482, 2483, 2523, 2574, 2575, 2621
Umkehrnocken, 1731, 1745, 1885
Umkehrschalter, 2492, 2584
Umschaltdifferenz, 2467, 2519, 2617
Umschaltdifferenz minus, 2520, 2618
Umschaltdifferenz plus, 2520, 2618
Umschaltpunkt, 2467
Unterbrechende Torfunktion, 206
Unterer Grenzwert, 243, 365, 426

V

Variablen des Technologieobjekts Drehzahlachse, 2085
Variablen des Technologieobjekts Externer Geber, 2144
Variablen des Technologieobjekts Gleichlaufachse, 2102
Variablen des Technologieobjekts Kurvenscheibe, 2177
Variablen des Technologieobjekts Messtaster, 2158
Variablen des Technologieobjekts Nocken, 2165
Variablen des Technologieobjekts Nockenspur, 2171
Variablen des Technologieobjekts Positionierachse, 1544, 1641, 2102
Verdrahtungstest
Easy Motion Control, 2837
Verfahrbereich, 2316
Verfahrbereichsbegrenzung, 1712, 1714, 1715, 1876
Vergleichswert, 40, 98, 118, 153, 163, 233, 312
Digitaleingabemodul, 78
Positionserfassung, 118, 163
Zählen, 98, 153
Verhalten bei CPU/Master-STOP
1STEP 5V, 2756
Verhalten bei CPU-STOP, 132, 1318, 1335
Technologiemodul, 1211
Verhalten der Digitalausgänge
bei Messbetriebsarten, 244
bei taktsynchronem Betrieb, 235
bei Zählbetriebsarten, 233, 312
Einschalten bei Erreichen des Vergleichswerts für Impulsdauer rückwärts, 238, 315
Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts, 237, 315
Einschalten bei Erreichen des Vergleichswerts für Impulsdauer vorwärts/rückwärts, 237, 314
Einschalten bei Zählerstand <= Vergleichswert, 237, 314
Einschalten bei Zählerstand >= Vergleichswert, 236, 314
Kein Vergleich, 236, 313
Randbedingungen, 240, 316
Schalten an Vergleichswerten, 239
Voreinstellung, 235, 279
Verhaltens eines Schrittmotors, 2776
Verstellung
dynamische, 459
Verzögerungszeit, 1199

Vierfachauswertung, 211
Voraussetzung
Bezugspunkt setzen, 2506, 2601
Fliegendes Istwert setzen, 2596
Geberdaten, 2511, 2609
Istwert setzen, 2503, 2596
Kantenerfassung, 2603
Längemessung, 2603
Nullpunktverschiebung, 2599
Referenzpunktfahrt, 2492, 2584
Schleifenfahrt, 2507, 2605
Schrittmaßfahrt, 2498, 2590
Tippen, 2488, 2580
Vorhalteweg, 460
Vorhaltezeit, 459
Vorhersagbarkeit, 1195

W

Wahrheitstabelle
Betriebsart Ein-/Ausschaltverzögerung, 1260
Betriebsart Frequenzausgabe, 1269
Betriebsart Gleichstrommotor, 1276
Betriebsart Impulsausgabe (Einzelimpuls), 1231
Betriebsart Impulskette, 1250
Betriebsart Impulsweitenmodulation (PWM), 1239
WCS, 2232
Wegerfassung
mit Motion Control, 61
mit SSI-Absolutwertgeber, 22
Wegfall
Synchronistation, 2600
Wegfall der Synchronisation, 480, 557
Wegmessung
Easy Motion Control, 2839
Wegnocken, 453, 454
Weltkoordinatensystem, 2232
Werkzeugarbeitspunkt, 2226, 2233
Werkzeugkoordinatensystem, 2233
Werkzeugzonen, 2336
Werte
vergleichen, 658

Z

Zählbereich, 221, 301
Zählen
einmaliges, 186
endlos, 185
periodisches, 187
Vergleichswert, 98, 153
Zähler setzen, 227, 306
mit dem Digitaleingang 1I2, 307
mit dem Digitaleingang I2, 229
mit der Nullmarke, 230, 309
mit einem externen Signal, 228, 306
Zählfunktionen, 20, 70, 71, 75
Zählgrenzen, 20, 23, 70, 71, 75, 92, 148, 180
Zählnockenspur, 457
Zeiten messen
zwischen zwei Flanken, 232
Zeitliches Verhalten
Standard-Technologie, 1194
Time-based IO, 1194
Zeitnocken, 453, 454
Einfluss von Funktionen, 474, 551
Schalten, 474, 552
Schaltverhalten, 474, 551
Ziel, 2467
Zielbereich, 2467, 2521, 2619
Überwachung, 2482, 2574
Zieleinlauf, 2482, 2575
Zonen
Arbeitsraumzonen, 2335
Arbeitszonen, 2335
Flanschzonen, 2337
Kinematikzonen, 2336
Meldezonen, 2335
Untereintrag, 2332, 2335
Werkzeugzonen, 2336
Zonengeometrie, 2337
Zonengeometrie, 2337
Zonenüberwachung, 2332
Zusätzlicher Geberistwert, 2785
Zweifachauswertung, 211
Zykluszeit, 1194
Zylindrischer Roboter
3D, 2298
3D mit Orientierung, 2302

SIEMENS

Online- und Diagnosefunktionen nutzen

Programmier- und Bedienhandbuch

Erreichbare Teilnehmer anzeigen 1

Gerätekonfiguration online ändern 2

Geräte online verbinden 3

Sicherung einer S7-CPU erstellen 4

PG/PC-Schnittstellen konfigurieren 5

Trace- und Logikanalysatorfunktion nutzen 6

Fernverbindung mit TeleService aufbauen 7

Geräte simulieren mit S7-PLCSIM 8

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Erreichbare Teilnehmer anzeigen....9

2 Gerätekonfiguration online ändern....11

3 Geräte online verbinden....13

3.1 Allgemeine Information zum Online-Modus....13

3.2 Ansicht im Online-Modus....15

3.3 Online-Verbindungsdaten voreinstellen....17

3.4 Online-Verbindung herstellen oder ändern....18

3.5 Online-Verbindung trennen....21

3.6 Online verbinden mit mehreren Geräten....22

3.7 Online-Verbindungen von mehreren Geräten trennen....23

4 Sicherung einer S7-CPU erstellen....25

4.1 Sicherungsmöglichkeiten für S7-CPUs....25

4.2 S7-300 und S7-400 CPUs sichern....27

4.2.1 Erstellen einer Sicherung für ein Gerät....27

4.2.2 Software und Hardware-Konfiguration eines Geräts wiederherstellen....28

4.2.3 Sicherung der Projektierung eines Geräts....29

4.3 S7-1200 und S7-1500 CPUs sichern....30

4.3.1 Erstellen einer Sicherung für ein Gerät....30

4.3.2 Sicherung der Projektierung eines Geräts....31

4.3.3 Projektierung eines Geräts wiederherstellen....32

5 PG/PC-Schnittstellen konfigurieren....35

5.1 Online-Zugänge....35

5.2 Grundlagen zur Parametrierung der PG/PC-Schnittstelle....37

5.3 Schnittstellen anzeigen oder verbergen....38

5.4 Eigenschaften einer Schnittstelle anzeigen und ändern....39

5.5 Schnittstellen hinzufügen....40

5.6 Ethernet-Schnittstelle parametrieren....41

5.6.1 Parametrieren der Industrial Ethernet-Schnittstelle....41

5.6.2 Betriebssystemparameter anzeigen....42

5.6.3 PG/PC-Schnittstelle mit einem Subnetz verbinden....42

5.6.4 Ethernet-Schnittstelle parametrieren....43

5.6.5 Nutzung von temporären IP-Adressen....44

5.6.6 Temporäre IP-Adressen verwalten....44

5.6.7 TCP/IP-Konfiguration zurücksetzen....45

5.7 MPI- und PROFIBUS-Schnittstelle parametrieren....46

5.7.1 Parametrieren der MPI- und PROFIBUS-Schnittstelle....46

5.7.2 MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren....47
5.7.3 MPI-Schnittstelle parametrieren....48
5.7.4 PROFIBUS-Schnittstelle parametrieren....50
5.7.5 Übersicht der Busparameter für PROFIBUS....52
5.7.6 MPI- oder PROFIBUS Konfiguration zurücksetzen....54

6 Trace- und Logikanalysatorfunktion nutzen....55

6.1 Vorwort....55
6.1.1 Security-Hinweise....56

6.2 Beschreibung....58
6.2.1 Unterstützte Hardware....58
6.2.2 Messwertaufzeichnung mit der Tracefunktion....58
6.2.3 Tracekonfiguration, Aufzeichnung, Trace im Gerät und Messung....60
6.2.4 Datenablage....61

6.3 Software-Oberfläche....63
6.3.1 Projektnavigator....64
6.3.1.1 Oberfläche - Projektnavigation Ordner "Traces"....64
6.3.1.2 Oberfläche - Projektnavigation Ordner "Messungen"....66
6.3.1.3 Oberfläche - Projektnavigation Ordner "Messungen im Gerät (Speicherkarte)"....67
6.3.1.4 Oberfläche - Projektnavigation Ordner "Überlagerte Messungen"....68
6.3.2 Arbeitsbereich....69
6.3.2.1 Oberfläche - Funktionsleiste des Trace....69
6.3.2.2 Oberfläche - Register Konfiguration....71
6.3.2.3 Oberfläche - Register Diagramm....71
6.3.2.4 Oberfläche - Register Signalauswahl (Überlagerte Messungen)....84
6.3.3 Task Card Trace....85
6.3.3.1 Oberfläche - Palette Mess-Cursor....85
6.3.3.2 Oberfläche - Palette Momentaufnahmen....87
6.3.4 Inspektorfenster....88
6.3.4.1 Oberfläche - Inspektorfenster....88

6.4 Bedienen....90
6.4.1 Schnelleinstieg....90
6.4.2 Anwenden der Tracefunktion - Übersicht....94
6.4.3 Projektnavigation....94
6.4.3.1 Trace anlegen....94
6.4.3.2 Konfiguration anzeigen....95
6.4.3.3 Diagramm anzeigen....95
6.4.3.4 Überlagerte Messung anlegen....96
6.4.3.5 Strukturierung der Objekte in Gruppen....96
6.4.4 Arbeitsbereich - Allgemein....97
6.4.4.1 Tracekonfiguration in das Gerät übertragen....97
6.4.4.2 Trace im Gerät aktivieren / deaktivieren....97
6.4.4.3 Aufzeichnung anzeigen....98
6.4.4.4 Laufende Aufzeichnung analysieren....99
6.4.4.5 Messungen im Projekt speichern....99
6.4.4.6 Messungen exportieren und importieren....100
6.4.4.7 Tracekonfiguration aus dem Gerät in das Projekt übertragen....101
6.4.4.8 Traces im Gerät löschen....101
6.4.5 Arbeitsbereich - Register Konfiguration....102
6.4.5.1 Trace konfigurieren....102
6.4.6 Arbeitsbereich - Register Diagramm....102

6.4.6.1 Verwendung des Kurvendiagramms....102
6.4.6.2 Verwendung der Signaltabelle....104
6.4.6.3 Verwendung der Signalgruppe in der Signaltabelle....105
6.4.6.4 Beobachten von schnellen Signalen....107
6.4.6.5 Aufzeichnungen vergleichen (überlagerte Messungen)....108
6.4.6.6 Messungen exakt ausrichten (überlagerte Messungen)....109
6.4.6.7 Drucken einer Aufzeichnung....110

6.5 Geräte....111
6.5.1 S7-1200/1500 CPUs....111
6.5.1.1 Erfassbare Variablen....111
6.5.1.2 Lebensdauer der Tracekonfiguration und aufgezeichneten Werte im Gerät....112
6.5.1.3 Aufzeichnungsebenen....112
6.5.1.4 Mengengerüst....113
6.5.1.5 CPU-Belastung durch Traceaufzeichnung....113
6.5.1.6 Software-Oberfläche der Konfiguration....113
6.5.1.7 Konfigurieren....123

7 Fernverbindung mit TeleService aufbauen....129

7.1 Grundlagen für das Arbeiten mit TeleService....129
7.1.1 Einführung in TeleService....129
7.1.2 Funktionalität von TeleService....130
7.1.3 Telefonbuch bei TeleService....130

7.2 Arbeiten mit dem Telefonbuch....132
7.2.1 Grundlagen zum Arbeiten mit dem Telefonbuch....132
7.2.2 Aufbau des Telefonbuchs....133
7.2.3 Symbole im Telefonbuch....134
7.2.4 Telefonbuch verwalten....134
7.2.4.1 Telefonbuch öffnen....134
7.2.4.2 Zeilen ins Telefonbuch einfügen....134
7.2.4.3 Spalten im Telefonbuch ein- und ausblenden....135
7.2.4.4 Telefonbuch speichern....135
7.2.4.5 Telefonbuchdaten importieren....135
7.2.4.6 Telefonbuchdaten exportieren....137
7.2.4.7 Telefonbuch drucken....138
7.2.4.8 Wählregeln definieren....138

7.3 Fernverbindung als Wählverbindung....140
7.3.1 Grundlagen zum Aufbauen einer Wählverbindung....140
7.3.2 Telefonnetze und Modems....141
7.3.2.1 Zulässige Telefonnetze und Modems....141
7.3.2.2 Installieren des lokalen Modems....142
7.3.2.3 Anschließen und Parametrieren des entfernten Modems....143
7.3.3 Zugriffsschutz bei Wählverbindungen....144
7.3.3.1 Informationen zum Zugriffsschutz....144
7.3.3.2 Rückrufmöglichkeiten in TeleService....145
7.3.3.3 Informationen zu den Schutzstufen....145
7.3.3.4 Zugriffsschutz und Rückrufnummer für den TS Adapter einrichten....146
7.3.3.5 Rückruf in TeleService durchführen....147
7.3.4 TS Adapter MPI....149
7.3.4.1 Kurzbeschreibung des TS Adapter MPI....149
7.3.4.2 Funktionsweise des TS Adapter MPI....149
7.3.4.3 TS Adapter MPI im Direktanschluss betreiben....150

7.3.4.4 TS Adapter MPI im Modemanschluss betreiben....151
7.3.4.5 Parametriermöglichkeiten für den TS Adapter MPI....152
7.3.4.6 TS Adapter MPI parametrieren....153
7.3.4.7 Standard-Parametrierung für TS Adapter MPI wieder herstellen....154
7.3.4.8 Exportieren von Adapterparametern....155
7.3.4.9 Importieren von Adapterparametern....156
7.3.5 TS Adapter IE....157
7.3.5.1 Kurzbeschreibung TS Adapter IE....157
7.3.5.2 Funktionsweise des TS Adapters IE....158
7.3.5.3 Anschlussarten....158
7.3.5.4 Parametriermöglichkeiten für den TS Adapter IE....160
7.3.5.5 TS Adapter IE parametrieren....161
7.3.6 Aufbauen einer Wählverbindung zu einer entfernten Anlage....163
7.3.6.1 Wählverbindung aufbauen....163
7.3.6.2 Wählverbindung trennen....164

7.4 Fernverbindung als VPN-Verbindung....165
7.4.1 Grundlagen zum Aufbauen einer VPN-Verbindung....165
7.4.2 Grundlagen zu CA-Zertifikaten....166
7.4.3 CA-Zertifikat für VPN-Verbindung installieren....168
7.4.4 CA-Zertifikat für VPN-Verbindung löschen....171
7.4.5 Aufbauen einer VPN-Verbindung zu einer entfernten Anlage....171
7.4.5.1 VPN-Verbindung aufbauen....171
7.4.5.2 VPN-Verbindung trennen....173
7.4.6 TS Adapter IE Advanced....173
7.4.6.1 Kurzbeschreibung TS Adapter IE Advanced....173
7.4.6.2 Anschlussarten....174
7.4.6.3 Parametriermöglichkeiten für den TS Adapter IE Advanced....175
7.4.6.4 TS Adapter IE Advanced parametrieren....176

7.5 CPU-gesteuerte Fernverbindungen bei TeleService....178
7.5.1 Übersicht über CPU-gesteuerte Fernverbindungen....178
7.5.2 Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung)....179
7.5.2.1 Zugriff von entfernten Anlagen auf ein PG/PC....179
7.5.2.2 Voraussetzungen für den Verbindungsaufbau....179
7.5.3 Datenaustausch zwischen entfernten Anlagen (AS-AS-Fernkopplung)....180
7.5.3.1 Grundlagen zur AS-AS-Fernkopplung....180
7.5.3.2 Hardware- und Softwarevoraussetzungen für AS-AS-Fernkopplung....181
7.5.4 Senden einer SMS von einer Anlage....182
7.5.4.1 Voraussetzungen für das Senden einer SMS....182
7.5.5 Senden einer E-Mail von einer Anlage....184
7.5.5.1 Voraussetzungen für das Senden einer E-Mail....184

7.6 Hinweise zur Fehlersuche....186
7.6.1 Allgemeine Hinweise für die Fehlersuche bei Modem-Problemen....186
7.6.2 Aufzeichnen einer Protokolldatei für das Modem....186
7.6.3 Wählverbindung zum TS Adapter wird nicht aufgebaut....187
7.6.4 Wählverbindung vom TS Adapter wird nicht aufgebaut....189
7.6.5 Modemverbindung wird unterbrochen....190
7.6.6 Checkliste bei Schwierigkeiten mit dem Modem....190
7.6.7 Meldungen des Modems....191
7.6.8 Mögliche Fehlermeldungen bei VPN-Verbindungen....192

8 Geräte simulieren mit S7-PLCSIM....195

8.1 Simulation von Geräten....195

8.2 Simulation starten....196

Index....197

Erreichbare Teilnehmer anzeigen 1

Erreichbare Teilnehmer sind alle angeschalteten Geräte, die mit einer Schnittstelle des PG/PC verbunden sind. Insofern können Sie auch Geräte anzeigen, die Sie mit den aktuell installierten Produkten nur eingeschränkt oder gar nicht projektieren können.

Um die Anzeige bei großen Netzwerken mit vielen Netzteilnehmern zu beschleunigen, werden bei PROFINET-Geräten die PROFINET-Gerätenamen in leicht angepasster Form angezeigt. Unicode-spezifische Zeichen innerhalb der PROFINET-Gerätenamen werden nach dem Punicode-Verfahren (entsprechend RFC 3492) in ASCII-kompatible Zeichen umgewandelt.

Erreichbare Teilnehmer in der Projektnavigation anzeigen

Um die erreichbaren Teilnehmer an einer Schnittstelle des PG/PC anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Online-Zugänge" in der Projektnavigation.
2. Klicken Sie auf das Pfeilsymbol links neben der Schnittstelle, um alle Objekte sichtbar zu machen, die unterhalb der Schnittstelle angeordnet sind.
3. Doppelklicken Sie auf "Erreichbare Teilnehmer aktualisieren" unterhalb der Schnittstelle. Sie sehen den Fortschritt des Suchvorgangs in der Statuszeile. Wenn Sie das gewünschte Gerät vor Abschluss des Suchvorgangs gefunden haben, können Sie den Suchvorgang abbrechen. Klicken Sie dazu auf das Kreuz rechts neben der Fortschrittsanzeige.
   Die Netzteilnehmer, die über die Suche gefunden wurden, werden unterhalb der Schnittstelle angezeigt.
4. Optional: Selektieren Sie die Schnittstelle in der Projektnavigation. Öffnen Sie anschließend das Übersichtsfenster in der Detailsicht, um weitere Details zu den jeweiligen Teilnehmern anzuzeigen. Weitere Details sind z. B. der Gerätetyp, die MAC-Adresse oder der projektierte Gerätename.
   Die detaillierte Anzeige der Teilnehmer im Übersichtsfenster kann bei vielen Netzteilnehmern etwas Zeit in Anspruch nehmen, da jedes einzelne Gerät die Details übermitteln muss.

Erreichbare Teilnehmer in einer Tabelle anzeigen

Um die erreichbaren Teilnehmer in einer übersichtlichen Tabelle außerhalb der Projektnavigation anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online" den Befehl "Erreichbare Teilnehmer".
   Der Dialog "Erreichbare Teilnehmer" wird geöffnet.
2. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" den Typ der Schnittstelle.
3. Wählen Sie die gewünschte Schnittstelle des PG/PC aus der Klappliste "PG/PC-Schnittstelle".

4. Klicken Sie auf die Schaltfläche "Suche starten".
   Die erreichbaren Teilnehmer an der gewählten Schnittstelle des PG/PC werden in der Tabelle angezeigt. In der Grafik wird die Verbindungslinie zwischen PG/PC und dem Netzwerk geschlossen und grün dargestellt. Wenn an einer Schnittstelle keine Geräte erreichbar sind, dann wird die Verbindungslinie unterbrochen dargestellt.
5. Um zu einem Gerät in der Projektnavigation zu springen, wählen Sie das Gerät aus der Liste der erreichbaren Teilnehmer und klicken auf die Schaltfläche "Anzeigen".
   Das selektierte Gerät wird in der Projektnavigation markiert.

Zusätzliche Informationen zu den Geräten in der Projektnavigation anzeigen

Um zusätzliche Informationen zu den einzelnen erreichbaren Teilnehmern in der Projektnavigation anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Pfeilsymbol links von einem der erreichbaren Teilnehmer in der Projektnavigation.
   Bei bekannten Geräten werden alle online verfügbaren Daten, z. B. Bausteine oder Systemdaten, angezeigt. Wenn für ein Gerät über das Kontextmenü weitere Bearbeitungsmöglichkeiten bestehen, dann wird das Gerät in schwarzer Schrift angezeigt. Auf die grau hinterlegten Objekte, z. B. Bausteine, haben Sie nur Lesezugriff..

Zusätzliche Informationen zu den Geräten in der Detailansicht anzeigen

Um zusätzliche Informationen zu den Geräten in der Detailansicht des Übersichtsfensters anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Ordner "Online-Zugänge" im Knoten der Netzwerkkarte doppelt auf den Eintrag "Erreichbare Teilnehmer aktualisieren".
   Alle im Netzwerk erreichbaren Teilnehmer werden unterhalb der Schnittstelle aufgelistet.
2. Klicken Sie doppelt auf den Eintrag "Weitere Informationen anzeigen".
   Weitere Informationen werden in der Detailansicht des Übersichtsfensters angezeigt.

Siehe auch

Gerätekonfiguration online ändern (Seite 11)

Online-Verbindungsdaten voreinstellen (Seite 17)

Gerätekonfiguration online ändern

2

Manche Geräte, vorzugsweise in kleinen Hardware-Aufbauten, können Sie direkt online auf dem Gerät parametrieren. Sie müssen dazu kein Projekt anlegen und offline Daten vorhalten. So können Sie unkompliziert und schnell die Gerätekonfiguration ändern. Sie müssen die Hardware-Konfiguration nicht übersetzen und keinen Download durchführen. In Abhängigkeit vom Gerät werden entweder alle Änderungen sofort aktiv oder nach einer Bestätigung auf das Gerät geschrieben.

Voraussetzung

- Das Gerät muss die Online-Parametrierung unterstützen. Ob Ihre jeweiligen Geräte diese Funktion unterstützen, erfahren Sie im Gerätehandbuch.
- Das Gerät muss mit dem PG/PC verbunden sein und in der Liste der erreichbaren Teilnehmer verfügbar sein.

Vorgehen

Um die Gerätekonfiguration online zu ändern, gehen Sie folgendermaßen vor:

1. Lassen Sie sich die erreichbaren Teilnehmer an der Schnittstelle anzeigen, über die das Gerät verbunden ist. Wie Sie die erreichbaren Teilnehmer anzeigen, erfahren Sie im Kapitel "Erreichbare Teilnehmer anzeigen (Seite 9)".
2. Klappen Sie das Gerät auf, um die unterlagerten Elemente anzuzeigen.
3. Doppelklicken Sie auf den Eintrag "Gerät parametrieren".
   Im Arbeitsbereich wird eine Konfigurationsseite für das Gerät geöffnet.
4. Nehmen Sie alle notwendigen Einstellungen vor.
   Bei manchen Geräten werden die geänderten Einstellungen sofort wirksam.
5. Optional in Abhängigkeit vom Gerät: Klicken Sie auf die Schaltfläche "Zum Gerät übertragen".
   Die Einstellungen werden auf das Gerät übertragen.

Geräte online verbinden 3

3.1 Allgemeine Information zum Online-Modus

Online-Modus

Im Online-Modus besteht eine Online-Verbindung zwischen Ihrem PG/PC und einem oder mehreren Geräten.

Eine Online-Verbindung zwischen PG/PC und Gerät ist beispielsweise für folgende Tätigkeiten erforderlich:

- Testen von Anwenderprogrammen
- Anzeigen und Ändern des Betriebszustands der CPU
- Anzeigen und Einstellen von Uhrzeit und Datum der CPU
- Anzeigen des Baugruppenzustands
- Bausteine vergleichen
- Hardware diagnostizieren

Um eine Online-Verbindung herstellen zu können, müssen PG/PC und Gerät physikalisch oder über eine Fernverbindung miteinander verbunden sein. Alternativ unterstützen einige Geräte einen Simulationsmodus. In diesem Fall wird über die virtuelle Schnittstelle PLCSIM eine Verbindung zum Gerät simuliert.

Nach dem Herstellen einer Verbindung können Sie in der Online- und Diagnosesicht oder über die Task Card "Online-Tools" auf die Daten des Geräts zugreifen. Außerdem wird der aktuelle Online-Status eines Geräts rechts neben dem Gerät in der Projektnavigation symbolisch angezeigt. Die Bedeutung der einzelnen Statussymbole entnehmen Sie dem jeweiligen Tooltip.

Hinweis

Einige Online-Funktionen sind vom Installationsumfang der Software abhängig oder davon, ob ein Projekt geöffnet ist.

Standby oder Ruhezustand des PG/PC

Wird der PG/PC bei einer bestehenden Online-Verbindung in den Standby-Modus oder in den Ruhezustand versetzt, werden alle Online-Verbindungen getrennt. Nach dem Aufwachen des PG/PC aus dem Ruhemodus werden die Online-Verbindungen nicht automatisch wiederhergestellt.

Beachten Sie in diesem Zusammenhang, dass durch das plötzliche Trennen der Online-Verbindung Daten verloren gehen können oder ein angeschlossenes Gerät unterbricht den Programmablauf.

LED-Blinktest durchführen

In vielen Online-Dialogen können Sie einen LED-Blinktest durchführen, wenn das online verbundene Gerät diese Funktion unterstützt. Aktivieren Sie das Optionskästchen "LED blinken", dann blinkt eine LED am gerade ausgewählten Teilnehmer. Diese Funktion ist beispielsweise hilfreich, wenn Sie sich nicht sicher sind, welches Gerät im Hardware-Aufbau dem gerade in der Software gewählten Teilnehmer entspricht.

Beachten Sie weitere Hinweise und mögliche Einschränkungen zum LED-Blinktest in der jeweiligen Gerätedokumentation.

Siehe auch

Ansicht im Online-Modus (Seite 15)

3.2 Ansicht im Online-Modus

Online-Anzeigen

Nach erfolgreichem Aufbau der Online-Verbindung ändert sich die Bedienoberfläche. Falls ein Gerät nicht erreichbar ist, wird dies symbolisch angezeigt. Das folgende Bild zeigt ein online verbundenes Gerät und die dazugehörige Bedienoberfläche:

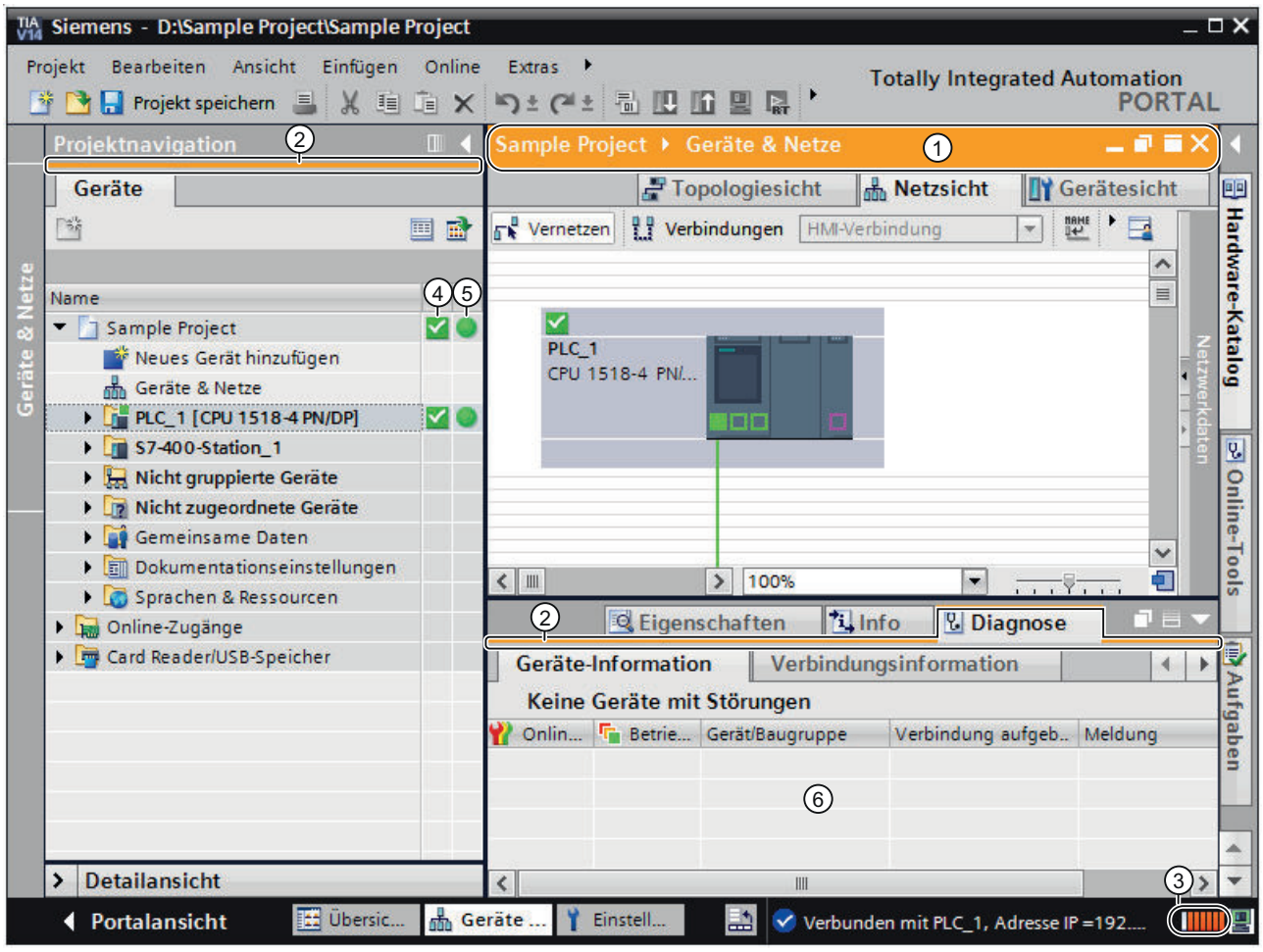

① Die Titelleiste des aktiven Fensters erhält einen orangefarbenen Hintergrund, sobald mindestens eines der Geräte, die aktuell im Editor angezeigt werden, erfolgreich online geschaltet wurde. Wenn ein oder mehrere Geräte nicht erreichbar sind, dann wird in der Titelleiste des Editors ein Symbol für eine unterbrochene Verbindung angezeigt.

② Die Titelleisten der inaktiven Fenster der zugehörigen Station erhalten am unteren Rand einen orangefarbenen Strich.

③ Am rechten Rand der Statuszeile erscheint ein orangefarbener, pulsierender Balken. Ist die Verbindung zwar zu Stande gekommen, aber fehlerhaft, wird statt des Balkens ein Symbol für eine unterbrochene Verbindung angezeigt. Im Inspektorfenster unter "Diagnose" erhalten Sie weitere Informationen zum Fehlerbild. Zusätzlich zeigt Ihnen die Statuszeile an, wenn die Online-Verbindung zur SIMATIC Hardware über den TIA Portal Cloud Connector erfolgt.

④ Die linke Spalte in der Projektnavigation zeigt die Diagnosestatus für Hardware-Objekte im Online-Modus durch Symbole an.

⑤ Die rechte Spalte in der Projektnavigation zeigt die Vergleichsstatus für Software-Objekte im Online-Modus durch Symbole an. Es wird automatisch ein Vergleich zwischen dem Online- und dem Offline-Zustand durchgeführt. Unterschiede zwischen den Online- und Offline-Objekten werden symbolisch angezeigt.

⑥ Im Inspektorfenster wird der Bereich "Diagnose > Geräte-Information" in den Vordergrund gebracht.

Abriss der Online-Verbindung

Der Online-Modus mit seiner zugehörigen Anzeige bleibt so lange erhalten, wie noch mindestens ein Gerät online verbunden ist. Reißt bei einem oder mehreren Geräten die Online-Verbindung ab, bleibt das TIA Portal im Online-Modus. Erst wenn zu keinem Gerät mehr eine Online-Verbindung besteht, wechselt die Darstellung des TIA Portals in den Offline-Modus.

Online-Verbindung über den TIA Portal Cloud Connector

Wenn Sie das TIA Portal in einer virtuellen Umgebung nutzen, haben Sie die Möglichkeit, über den TIA Portal Cloud Connector eine Online-Verbindung zu SIMATIC Hardware herzustellen, die an einem Benutzergerät angeschlossen ist. Erfolgt die Online-Verbindung über den TIA Portal Cloud Connector, wird in der Statuszeile das folgende Symbol angezeigt:

Eine Anleitung zur Nutzung des TIA Portals in einer virtuellen Umgebung (Private Cloud) finden Sie auf dem Installationsdatenträger im Verzeichnis "Documents\Readme \<Sprachverzeichnis>". Öffnen Sie hier das PDF-Dokument "TIAPortalCloudConnectorHowTo<Sprachkennung>.pdf".

Siehe auch

Allgemeine Information zum Online-Modus (Seite 13)

3.3 Online-Verbindungsdaten voreinstellen

Wenn Sie bevorzugt eine bestimmte Netzwerkschnittstelle Ihres PG/PC verwenden, um eine Online-Verbindung herzustellen, dann können Sie diese voreinstellen. Der angegebene Verbindungsweg ist anschließend in den Dialogen zur Online-Verbindung bereits ausgewählt.

Vorgehen

Um einen Standardverbindungsweg festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
   Die Einstellungen des TIA Portals werden geöffnet.
2. Wählen Sie in der Bereichsnavigation den Eintrag "Online- und Diagnose".
3. Legen Sie im Abschnitt "Voreingestellter Verbindungsweg für den Online-Zugriff" den Typ der PG/PC-Schnittstelle sowie die Schnittstelle selbst fest.
4. Aktivieren Sie das Optionskästchen "Voreingestellten Verbindungsweg für den Online-Zugriff verwenden".

Siehe auch

Online-Verbindung herstellen oder ändern (Seite 18)

Erreichbare Teilnehmer anzeigen (Seite 9)

Online-Zugänge (Seite 35)

3.4 Online-Verbindung herstellen oder ändern

Um die Online-Funktionen eines Geräts zu nutzen, legen Sie beim ersten Verbindungsversuch den Verbindungsweg fest.

Voraussetzung

Mindestens eine PG/PC-Schnittstelle ist installiert, die physikalisch mit einem Gerät verbunden ist, z. B. mit einem Ethernet-Kabel. Alternativ ist das Erstellen virtueller Online-Verbindungen über PLCSIM möglich.

Vorgehen

Um eine Online-Verbindung herzustellen oder zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein oder mehrere Geräte für die Online-Verbindung.
2. Wählen Sie im Menü "Online" den Befehl "Online verbinden".
   - Falls das Gerät bereits online verbunden war, wird die Online-Verbindung automatisch über den zuvor festgelegten Verbindungsweg aufgebaut.
   - Falls vorher noch keine Verbindung bestanden hat oder wenn die Geräteadresse nicht im selben Subnetz des PG/PC-Gerätes liegt, wird der Dialog "Online verbinden" geöffnet.

   Wenn Sie einen festgelegten Verbindungsweg ändern möchten, wählen Sie stattdessen im Menü "Online" den Befehl "Erweitert online verbinden". Der Dialog "Online verbinden" wird in diesem Fall erneut geöffnet.
3. Wählen Sie den Verbindungsweg:
   - Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" den Typ der Schnittstelle.
   - Wählen Sie die PG/PC-Schnittstelle aus der Klappliste "PG/PC-Schnittstelle".
     Die Auswahl entfällt, wenn Sie bereits in den Einstellungen eine PG/PC-Schnittstelle festgelegt haben.
   - Wählen Sie aus der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle oder das Subnetz für die Verbindung aus. Wählen Sie eine direkte Verbindung, falls kein S7-Gateway zwischengeschaltet ist. Wählen Sie das passende Subnetz für die Verbindung mit dem PG/PC aus, falls das Gerät über ein S7-Gateway erreichbar ist. Wählen Sie den Eintrag "Versuche alle Schnittstellen", falls das Gerät über mehrere Schnittstellen verfügt und Sie nicht wissen, über welche der Schnittstellen das Gerät mit dem PG/PC verbunden ist.
     Wenn Sie ein MPI- oder Profibus-Subnetz gewählt haben, werden zu diesem Zeitpunkt die parametrierten Busparameter in die PG/PC-Schnittstelle übernommen.
   - Falls das Gerät über ein Gateway erreichbar ist, wählen Sie aus der Klappliste "1. Gateway" das Gateway aus, das die betroffenen Subnetze miteinander verbindet.

4. Optional: Falls über den gewählten Verbindungsweg keine Geräte erreichbar sind, so wird die Verbindungslinie zwischen PG/PC und dem Gerät unterbrochen dargestellt. Dieser Fall tritt zum Beispiel dann ein, wenn das Gerät über einen NAT-Router mit dem Netz verbunden ist.
   Um alle angeschalteten und mit einer Schnittstelle des PG/PC verbundenen Geräte anzuzeigen, wählen Sie aus der Klappliste den Eintrag "Erreichbare Teilnehmer anzeigen" und klicken erneut auf "Suche starten".
5. Optional: Um das Gerät trotz eines unterschiedlichen Subnetzes zu erreichen, haben Sie zwei Möglichkeiten. Nutzen Sie beide Möglichkeiten in administrierten Netzen nur nach Rücksprache mit dem Administrator.
   - Vergeben Sie unter Windows eine statische IP-Adresse an der PG/PC-Schnittstelle.
   - Beim Herstellen der Online-Verbindung wird der Schnittstelle des PG/PC temporär eine weitere IP-Adresse zugewiesen, die sich im selben Subnetz wie das Gerät befindet. Klicken Sie auf "Verbinden" und bestätigen Sie die anschließende Abfrage.
6. Klicken Sie auf die Schaltfläche "Suche starten".
   Geräte, die über den eingestellten Verbindungsweg erreichbar sind, werden in der Tabelle der Zielgeräte angezeigt. Die Verbindungslinie in der Grafik auf der linken Seite wird geschlossen dargestellt.
7. Optional: Führen Sie einen Blinktest durch. Aktivieren Sie in der Grafik auf der linken Seite das Optionskästchen "LED blinken". Mit dem Blinktest kontrollieren Sie, ob Sie den richtigen Teilnehmer ausgewählt haben. Der Blinktest wird nicht von allen Geräten unterstützt.
8. Selektieren Sie in der Tabelle das Zielgerät und bestätigen Sie die Auswahl mit "Verbinden". Die Online-Verbindung wird zum ausgewählten Zielgerät aufgebaut.

Ergebnis

Nach dem Herstellen der Online-Verbindung verfärben sich die Titelleisten der Editoren orange. Außerdem wird in der Titelleiste eines Editors und in der Statuszeile jeweils eine orange Fortschrittsanzeige angezeigt. Die Statussymbole in der Projektnavigation zeigen den Unterschied zwischen online und offline vorhandenen Objekten an.

Der eingestellte Verbindungsweg wird für zukünftige Verbindungsversuche gespeichert.

Bei automatischer Vergabe der PG/PC-Adresse erscheint der Dialog wieder, wenn der Rechner neu gestartet wurde. Die Verbindung muss in diesem Fall erneut hergestellt werden.

Hinweis

Aktionen rückgängig machen

Nachdem Sie eine Online-Verbindung hergestellt haben, können Sie keine Aktionen rückgängig machen.

Hinweis

Direkte Verbindungen oder Subnetze verwenden

Eine Online-Verbindung zwischen PG/PC und angeschlossenen Geräten ist entweder über eine direkte Verbindung oder über ein Subnetz möglich.

Ein gleichzeitige Verbindung der Geräte über verschiedene Subnetze ist nicht möglich, wenn diese über dieselbe Schnittstelle physikalisch verbunden sind. Gleichzeitige Verbindungen zu verschiedenen Subnetzen sind nur durch geroutete Verbindungen möglich.

Eine gleichzeitige Verbindung von Geräten über ein Subnetz und über Direktverbindungen über dieselbe Schnittstelle ist nicht möglich.

Siehe auch

Online-Verbindungsdaten voreinstellen (Seite 17)

Online-Verbindung trennen (Seite 21)

Online verbinden mit mehreren Geräten (Seite 22)

Ansicht im Online-Modus (Seite 15)

Nutzung von temporären IP-Adressen (Seite 44)

3.5 Online-Verbindung trennen

Vorgehen

Um die bestehende Online-Verbindung zu trennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Gerät in der Projektnavigation, für das Sie die Online-Verbindung trennen möchten.
2. Wählen Sie im Menü "Online" den Befehl "Online-Verbindung trennen".
   Die Online-Verbindung wird getrennt.

Hinweis

Aktionen rückgängig machen

Nachdem Sie eine Online-Verbindung getrennt haben, können Sie keine Aktionen rückgängig machen.

Siehe auch

Online-Verbindung herstellen oder ändern (Seite 18)

3.6 Online verbinden mit mehreren Geräten

Sie können eine Online-Verbindung zu mehreren Geräten auf einmal herstellen, ohne die einzelnen Geräte vorher in der Projektnavigation oder in der Netzsicht selektieren zu müssen.

Voraussetzung

- Es darf kein Gerät selektiert sein
- Es ist mindestens eine PG/PC-Schnittstelle installiert, die physikalisch mit einem Gerät verbunden ist, z. B. mit einem Ethernet-Kabel. Alternativ ist das Aufbauen einer virtuellen Online-Verbindung über PLCSIM oder einer Fernverbindung möglich.

Vorgehen

Um eine Online-Verbindung zu mehreren Geräten gleichzeitig herzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Projekt in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Online verbinden".
   Der Dialog "Gerät für den Aufbau der Online-Verbindung auswählen" mit einer Tabelle aller verfügbaren Geräte wird geöffnet.
3. Selektieren Sie in der Spalte "Online verbinden" die Geräte, zu denen Sie eine Online-Verbindung aufbauen möchten.
4. Klicken Sie auf die Schaltfläche "Online verbinden".

Ergebnis

Eine Verbindung zu allen ausgewählten Geräten wird ohne weitere Nachfrage hergestellt, wenn vorher bereits mindestens einmal eine Verbindung zu den gewählten Geräten aufgebaut war. Bestand zuvor noch keine Online-Verbindung, wird der Dialog "Online verbinden" geöffnet. Konfigurieren Sie in diesem Fall zunächst die Online-Verbindung, wie im Kapitel "Online verbinden und Online-Verbindung trennen (Seite 18)" beschrieben.

Siehe auch

Online-Verbindung herstellen oder ändern (Seite 18)

Nutzung von temporären IP-Adressen (Seite 44)

3.7 Online-Verbindungen von mehreren Geräten trennen

Sie können die bestehenden Online-Verbindungen zu mehreren Geräten auf einmal trennen, ohne die einzelnen Geräte vorher in der Projektnavigation oder in der Netzsicht selektieren zu müssen.

Voraussetzung

- Es ist kein Gerät selektiert.
- Für mindestens ein Gerät besteht eine Online-Verbindung.

Vorgehen

Um die Online-Verbindungen zu mehreren Geräten gleichzeitig zu trennen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online" den Befehl "Online-Verbindung trennen".
   Der Dialog "Geräte auswählen" mit einer Tabelle aller verfügbaren Geräte wird geöffnet.
2. Selektieren Sie in der Spalte "Online-Verbindung trennen" die Geräte, zu denen Sie die Online-Verbindung trennen möchten.
3. Klicken Sie auf die Schaltfläche "Online-Verbindung trennen".

Ergebnis

Die Online-Verbindung zu allen ausgewählten Geräten wird getrennt.

Sicherung einer S7-CPU erstellen 4

4.1 Sicherungsmöglichkeiten für S7-CPUs

Im Laufe der Zeit werden Sie an Ihrer Automatisierungsanlage etliche Änderungen vornehmen, z. B. neue Geräte hinzufügen, vorhandene Geräte austauschen oder das Anwenderprogramm anpassen. Sollten Änderungen zu unerwünschtem Verhalten führen, können Sie den früheren Stand der Automatisierungsanlage wiederherstellen. Um nach dem Austausch einzelner Geräte nahtlos weiterzuarbeiten, können Sie vorhandene Programme und Werte übernehmen. Die CPUs bieten für die Sicherung sowie die Wiederherstellung der Hardware-Konfiguration und der Software verschiedene Möglichkeiten.

Sicherungsmöglichkeiten

Die folgende Tabelle gibt einen Überblick über die Sicherungs- und Wiederherstellungsmöglichkeiten von S7-CPUs:

	Momentaufnahme der Beobachtungswerte	Laden von Gerät (Software)	Laden des Geräts als neue Station (Hardware und Software)	Sicherung von Online-Gerät laden
Anwendungsfall	Wiederherstellen eines bestimmten Zustands eines Datenbausteins. Die Aktualwerte von Datenbausteinen werden inklusive des Zeitstempels in das Projekt übernommen.	Laden von Bausteinen, die auf einer CPU vorhanden sind, in das Projekt	Laden der Hardware-Konfiguration und der Software von einem Gerät in das Projekt.	Erstellen einer vollständigen Sicherung einer CPU als Wiederherstellungspunkt. Die Sicherungskopie ist konsistent und weder änderbar noch kann sie geöffnet werden.
Voraussetzung	Die CPU ist bereits in einem Projekt angelegt. Die Datenbausteine müssen online und offline identisch sein.	Die CPU ist im Projekt angelegt.	Das Gerät ist im Hardware-Katalog des TIA-Portals verfügbar. Eventuell notwendige HSPs oder GSD-Dateien sind installiert.	-
Möglich im Betriebszustand	RUN, STOP	RUN, STOP	RUN, STOP	STOP
Möglich für F-CPUs	Ja	Ja	Ja	Ja
Sicherung editierbar	Ja	Ja	Ja	Nein

Sicherungsinhalte

Die folgende Tabelle zeigt, welche Daten sich mit welchen Möglichkeiten laden und sichern lassen:

	Momentaufnahme der Beobachtungswerte	Laden von Gerät (Software)	Laden des Geräts als neue Station (Hardware und Software)	Sicherung von Online-Gerät laden
Aktualwerte der Datenbausteine	Momentaufnahme möglich	Laden möglich	Laden möglich	Sicherung möglich
Software-Bausteine	-	Laden möglich	Laden möglich	Sicherung möglich
PLC-Variablen (Variablen- und Konstantennamen)	-	Laden bei S7-1200 und S7-1500 CPUs möglich	Laden bei S7-1200 und S7-1500 CPUs möglich	Sicherung möglich
Technologieobjekte	-	Laden möglich	Laden möglich	Sicherung möglich
Hardware-Konfiguration	-	-	Laden möglich	Sicherung möglich
Beobachtungstabellen (Webserver)	-	-	Laden nicht möglich	Sicherung möglich
Lokaldaten, Merker, Zeiten, Zähler und Prozessabbild	Momentaufnahme nicht möglich	Laden nicht möglich	Laden nicht möglich	Sicherung möglich
Archive und Rezepte (PLC)	-	-	-	Sicherung möglich
Allgemeine Daten auf der SIMATIC Memory Card, z. B. Hilfe zu Programmbausteinen oder GSD-Dateien	-	-	-	Sicherung möglich

Besonderheiten bei der Sicherung von Aktualwerten

Die Sicherungsart "Sicherung von Online-Gerät laden" sichert die Aktualwerte der als remanent eingestellten Variablen. Um die Konsistenz der remanenten Daten sicherzustellen, deaktivieren Sie während der Sicherung alle Schreibzugriffe auf remanente Daten.

Aktualwerte der nicht-remanenten Daten werden bei einem Betriebszustandsübergang von STOP nach RUN zurück auf ihre Startwerte gesetzt. Bei einer Sicherung der CPU werden von nicht-remanenten Daten lediglich die Startwerte gesichert.

Siehe auch

Erstellen einer Sicherung für ein Gerät (Seite 27)

Erstellen einer Sicherung für ein Gerät (Seite 30)

4.2 S7-300 und S7-400 CPUs sichern

4.2.1 Erstellen einer Sicherung für ein Gerät

Software und Hardware-Konfiguration einer S7-300/400 CPU sichern

Wenn Sie bereits eine Projektierung auf eine S7-300/400 CPU geladen haben, kann es sinnvoll sein, diese zu sichern. Möglicherweise haben Sie die Projektierung geändert und möchten die neue Konfiguration testen. Bevor Sie die neue Projektierung auf die CPU laden, erstellen Sie eine Sicherung des aktuellen Gerätezustands und können so zu einem späteren Zeitpunkt die aktuelle Konfiguration wiederherstellen. Die Sicherung erfolgt mit den Aktualwerten der CPU. Bei S7-400 CPUs mit Failsafe-Funktion werden die Initialwerte gesichert.

Sie können beliebig viele Sicherungen anlegen und so unterschiedliche Projektierungen für eine CPU vorhalten. Die Sicherungen werden mit dem Namen der CPU sowie Uhrzeit und Datum der Sicherung benannt. Die Sicherung finden Sie in der Projektnavigation unterhalb der CPU im Ordner "Online-Sicherungen".

Das folgende Bild zeigt eine S7-319 CPU, von der zwei Sicherungen erstellt wurden:

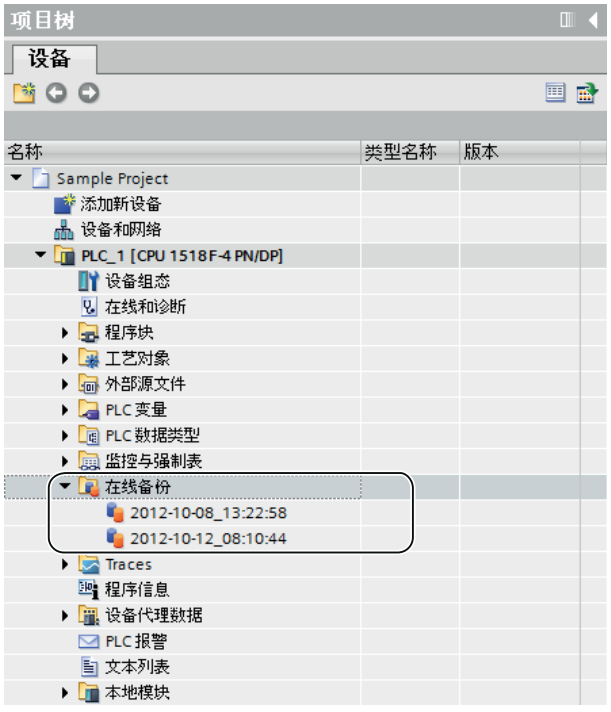

Siehe auch

Sicherungsmöglichkeiten für S7-CPUs (Seite 25)

4.2.2 Software und Hardware-Konfiguration eines Geräts wiederherstellen

Wenn Sie die Projektierung eines Geräts zu einem früheren Zeitpunkt gesichert haben, können Sie die Sicherung wieder auf das Gerät übertragen. Die gespeicherte Konfiguration wird dann auf dem Gerät wiederhergestellt.

Voraussetzung

Sie müssen das Gerät projektiert und eine Sicherung des Geräts im Projekt gespeichert haben.

Vorgehen

Um einen älteren Software- und Hardware-Stand auf einem Gerät wiederherzustellen, gehen Sie folgendermaßen vor:

1. Klappen Sie den Ordner des Geräts in der Projektnavigation auf, um die unterlagerten Objekte anzuzeigen.
2. Öffnen Sie den Ordner "Online-Sicherungen".
3. Selektieren Sie die Sicherung, die Sie wiederherstellen möchten.
4. Wählen Sie im Menü "Online" den Befehl "Laden in Gerät".
   - Falls Sie bereits eine Online-Verbindung hergestellt hatten, wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet und Sie müssen zunächst die Schnittstellen auswählen, über die Sie die Online-Verbindung zum Gerät herstellen möchten.
     Siehe auch: Online verbinden und Online-Verbindung trennen
5. Kontrollieren Sie die Meldungen im Dialog "Vorschau laden" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

   Hinweis

   Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen.

6. Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
7. Klicken Sie auf die Schaltfläche "Laden".
   Die Sicherung wird auf das Gerät übertragen und das Gerät wiederhergestellt.
   Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
8. Klicken Sie auf die Schaltfläche "Fertig stellen".

4.2.3 Sicherung der Projektierung eines Geräts

Sie können im TIA Portal die Projektierung einer S7-300/400 CPU sichern. So können Sie gefahrlos eine neue Projektierung auf ein Gerät laden und testen. Bei Bedarf können Sie die ursprüngliche Konfiguration der CPU wiederherstellen.

Voraussetzung

- Die CPU ist bereits im Projekt angelegt.
- Die CPU ist online verbunden sein. Besteht noch keine Online-Verbindung, so wird während der Sicherung eine Online-Verbindung hergestellt.

Vorgehen

Um eine Sicherung der aktuellen Projektierung einer CPU zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Sicherung von Online-Gerät laden".

Ergebnis:

Eine Sicherung der gesamten Hardware-Konfiguration und der Software wird angelegt. Die Sicherung wird in der Projektnavigation im Ordner "Name der CPU > Online-Sicherungen" abgelegt. Die Sicherung trägt den Namen der CPU mit der Uhrzeit und dem Datum der Sicherung. Sie können die Sicherung umbenennen, jedoch keine Änderungen an den Inhalten der Sicherung vornehmen.

4.3 S7-1200 und S7-1500 CPUs sichern

4.3.1 Erstellen einer Sicherung für ein Gerät

Software und Hardware-Konfiguration einer CPU sichern

Wenn Sie bereits eine Projektierung auf eine S7-1200 oder S7-1500 CPU geladen haben, kann es sinnvoll sein, die Projektierung zu sichern. Möglicherweise haben Sie die Projektierung geändert und möchten die neue Konfiguration testen. Bevor Sie die neue Projektierung auf die CPU laden, erstellen Sie eine Sicherung des aktuellen Gerätezustands. Zu einem späteren Zeitpunkt können Sie die aktuelle Konfiguration wiederherstellen.

Sie können beliebig viele Sicherungen anlegen und so unterschiedliche Projektierungen für eine CPU vorhalten. Die Sicherungen werden mit dem Namen der CPU sowie Uhrzeit und Datum der Sicherung benannt. Die Sicherung finden Sie in der Projektnavigation unterhalb der CPU im Ordner "Online-Sicherungen".

Das folgende Bild zeigt eine S7-1500 CPU, von der zwei Sicherungen erstellt wurden:

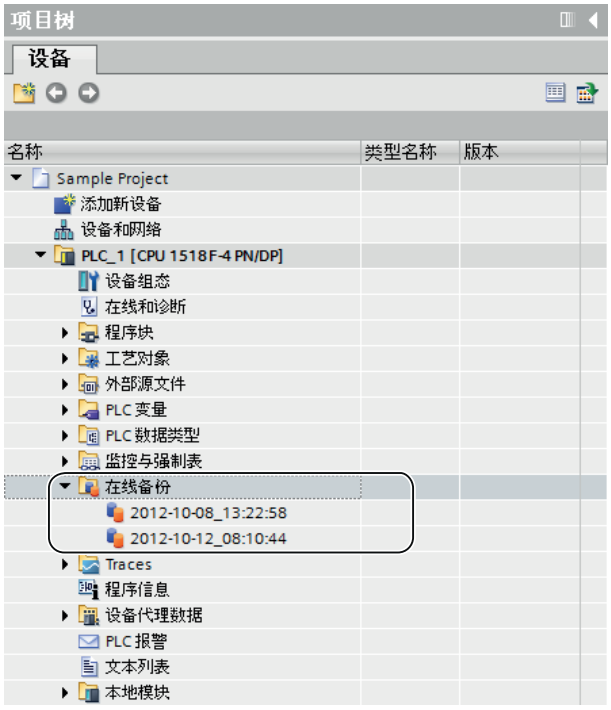

Umfang der Sicherung

Die Sicherung umfasst alle Daten, die nötig sind, um einen bestimmten Konfigurationsstand einer CPU wiederherzustellen. Folgende Daten werden beispielsweise gesichert:

- Die Inhalte der Memory Card
- Remanente Speicherbereiche, z. B. von Datenbausteinen, Zählern und Merkern
- Weitere remanente Speicherinhalte, wie z. B. IP-Adressparameter

Die Sicherung erfolgt mit den Aktualwerten der CPU. Einträge im Diagnosepuffer sind nicht Bestandteil der Sicherung. Bei einer S7-1500 CPU wird die aktuelle Uhrzeit nicht gesichert.

Siehe auch

Sicherungsmöglichkeiten für S7-CPUs (Seite 25)

4.3.2 Sicherung der Projektierung eines Geräts

Sie können im TIA Portal die Projektierung einer CPU sichern. So können Sie gefahrlos eine neue Projektierung auf ein Gerät laden und testen. Bei Bedarf können Sie die ursprüngliche Konfiguration der CPU wiederherstellen.

Die CPU geht während des Erstellens einer Sicherung in STOP. Wenn für die CPU eine Zugriffsstufe konfiguriert ist, benötigen Sie das Passwort für Lesezugriff auf die CPU.

Voraussetzung

- Die CPU ist bereits im Projekt angelegt.
- Das Gerät ist direkt über die PROFINET-Schnittstelle der CPU mit dem PG/PC verbunden. PROFIBUS-Schnittstellen der CPU sowie Schnittstellen von CMs/CPs werden nicht unterstützt.
- Die CPU ist online verbunden. Besteht noch keine Online-Verbindung, so wird während der Sicherung eine Online-Verbindung hergestellt.
- Die CPU befndet sich im Betriebszustand "STOP".

Vorgehen

Um eine Sicherung der aktuellen Projektierung einer CPU zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Sicherung von Online-Gerät laden". Gegebenenfalls müssen Sie das Passwort für lesenden Zugriff auf die CPU eingeben und bestätigen, dass die CPU in den Betriebszustand "STOP" versetzt wird.

Ergebnis:

Eine Sicherung der gesamten Hardware-Konfiguration und der Software wird angelegt. Die Sicherung wird in der Projektnavigation im Ordner "Name der CPU > Online-Sicherungen" abgelegt. Die Sicherung trägt den Namen der CPU mit der Uhrzeit und dem Datum der Sicherung. Sie können die Sicherung umbenennen, jedoch keine Änderungen an den Inhalten der Sicherung vornehmen.

Im Diagnosepuffer der CPU wird für jeden Sicherungsvorgang ein Eintrag erstellt.

Siehe auch

Projektierung eines Geräts wiederherstellen (Seite 32)

4.3.3 Projektierung eines Geräts wiederherstellen

Wenn Sie die Projektierung eines Geräts zu einem früheren Zeitpunkt gesichert haben, können Sie die Sicherung wieder auf das Gerät übertragen. Die gespeicherte Konfiguration wird dann auf dem Gerät wiederhergestellt.

Die CPU geht während des Ladens einer Sicherung in die CPU in STOP. Wenn für die CPU eine Zugriffstufe konfiguriert ist, benötigen Sie das Passwort für Schreibzugriff auf die CPU.

> ⚠ WARNUNG
>
> **Laden von Sicherungen mit unbekanntem Inhalt**
>
> Wenn Sie bei laufendem Anlagenbetrieb die vorgeschlagenen Aktionen beim Laden aktivieren, können bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden entstehen.
>
> Stellen Sie sicher, dass die Sicherungsinhalte keine Projektierung enthalten, die zu unvorgesehenem Verhalten der Anlage führen.

Voraussetzung

- Das Gerät ist projektiert und eine Sicherung des Geräts ist im Projekt gespeichert.
- Das Gerät ist über direkt über die PROFINET-Schnittstelle der CPU mit dem PG/PC verbunden.
- Die CPU befndet sich im Betriebszustand "STOP".
- Sie verfügen über das Passwort für Lesezugriff auf die CPU, falls eine Zugriffsstufe konfiguriert wurde.

Vorgehen

Um einen älteren Software- und Hardware-Stand auf einem Gerät wiederherzustellen, gehen Sie folgendermaßen vor:

1. Klappen Sie den Ordner des Geräts in der Projektnavigation auf, um die unterlagerten Objekte anzuzeigen.
2. Öffnen Sie den Ordner "Online-Sicherungen".
3. Selektieren Sie die Sicherung, die Sie wiederherstellen möchten.
4. Wählen Sie im Menü "Online" den Befehl "Laden in Gerät".
   - Falls Sie bereits eine Online-Verbindung hergestellt hatten, wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
   - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet und Sie müssen zunächst die Schnittstellen auswählen, über die Sie die Online-Verbindung zum Gerät herstellen möchten.
     Siehe auch: AUTOHOTSPOT
5. Kontrollieren Sie die Meldungen im Dialog "Vorschau laden" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
6. Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
7. Klicken Sie auf die Schaltfläche "Laden".
   Die Sicherung wird auf das Gerät übertragen und das Gerät wiederhergestellt.
   Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
8. Klicken Sie auf die Schaltfläche "Fertig stellen".
   Gegebenenfalls müssen Sie das Passwort für lesenden Zugriff auf die CPU eingeben und bestätigen, dass die CPU in den Betriebszustand "STOP" versetzt wird.
   Die Inhalte der Sicherung werden auf der CPU wiederhergestellt. Anschließend wird die CPU neu gestartet.

PG/PC-Schnittstellen konfigurieren 5

5.1 Online-Zugänge

Online-Zugang des Projekts

Im Ordner "Online-Zugänge" in der Projektnavigation finden Sie alle aktiven Schnittstellen Ihres PG/PC. An jedem Schnittstellensymbol erhalten Sie Informationen zum jeweiligen Status. Sie können sich die erreichbaren Teilnehmer anzeigen lassen und über das Kontextmenü die Eigenschaften einer Schnittstelle anzeigen und ändern.

Das folgende Bild zeigt den Ordner "Online-Zugänge" in der Projektnavigation:

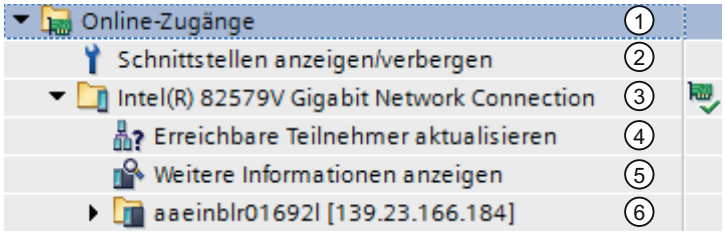

① Ordner "Online-Zugänge" in der Projektnavigation

Im Ordner "Online-Zugänge" sind sämtliche im PG/PC installierten Schnittstellen sichtbar.

② Schnittstellen anzeigen/verbergen

Mit der Funktion "Schnittstellen anzeigen/verbergen" blenden Sie einzelne Schnittstellen ein oder aus.

③ Statusanzeige für die Schnittstellen

Der aktuelle Status einer Schnittstelle wird durch ein Symbol rechts neben dem Namen angezeigt. Die Bedeutung der Symbole entnehmen Sie dem Tooltip.

④ Aktualisieren der Liste der erreichbaren Teilnehmer.

Diese Funktion ist für jede Hardware-Schnittstelle des PG/PC verfügbar. Software-Schnittstellen, wie z. B. eine Fernverbindung, verfügen nicht über diese Funktion.

⑤ Weitere Informationen anzeigen

Diese Funktion ist für jede Hardware-Schnittstelle des PG/PC verfügbar, nachdem Sie die Funktion "Erreichbare Teilnehmer aktualisieren" ausgeführt haben. Die weiteren Informationen finden Sie in der Detailansicht des Übersichtsfensters.

⑥ Geräte, die über die jeweilige Schnittstelle mit dem PG/PC verbunden sind

Der Typ des jeweiligen Geräts sowie dessen Status werden mit Hilfe des voranstehenden Symbols angezeigt.

Erreichbare Teilnehmer anzeigen oder aktualisieren

Sie haben folgende Möglichkeiten, um sich die online erreichbaren Teilnehmer an Ihrem PG/PC anzeigen zu lassen:

- Anzeige der erreichbaren Teilnehmer an einer einzelnen Schnittstelle des PG/PC in der Projektnavigation. In der Projektnavigation können außerdem zusätzliche Informationen zu den einzelnen erreichbaren Teilnehmern angezeigt werden.
- Anzeige der erreichbaren Teilnehmer aller Schnittstellen in einer Liste.

Siehe auch:AUTOHOTSPOT

Übersicht der Symbole für erreichbare Teilnehmer

Die erreichbaren Teilnehmer werden mit einem Symbol entsprechend ihres Typs und ihres Zustandes gekennzeichnet. Im Folgenden sehen Sie eine Übersicht über alle Symbole und deren Bedeutung.

	Symbol für nicht identifizierte Baugruppen Dieses Symbol wird immer dann angezeigt, wenn die Identifikation einer Baugruppe noch nicht vollständig abgeschlossen ist, oder wenn die Identifikation einer Baugruppe nicht erfolgreich war, beispielsweise weil die erforderlichen Online-Daten nicht gelesen werden konnten.
	Symbol für folgende Gerätetypen: • PLCs • SIMOCODE pro-Geräte • IE/PB Links • CPs von PC-Systemen • SCALANCE Kopfbaugruppen • S7-300 und S7-400 CPs • PROFINET IO-Devices und PROFINET CPs • SCALANCE-Baugruppen und Netzübergänge, die nicht identifiziert werden konnten
	PROFINET IO-Devices, Encoder, Switchgear, Sensoren und Identifikationssysteme, die durch ähnliche Geräte ersetzt wurden, weil diese nicht identifiziert werden konnten
	Symbol für folgende Gerätetypen: • HMI-Geräte • PROFINET IO-Devices vom Typ HMI, wenn diese nicht identifiziert werden konnten und deshalb durch ein ähnliches Gerät ersetzt wurden
	PROFINET IO-Devices vom Typ Antrieb, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden
	PROFINET IO-Devices vom Typ Development Kit und Netzkomponenten, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden
	PROFINET IO-Devices vom Typ TeleService Adapter, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden

Siehe auch

Eigenschaften einer Schnittstelle anzeigen und ändern (Seite 39)

5.2 Grundlagen zur Parametrierung der PG/PC-Schnittstelle

Verbindungsmöglichkeiten zu Zielsystemen

Wenn die Geräte des Projekts über unterschiedliche Subnetze verbunden sind, müssen Sie der PG/PC-Schnittstelle jeweils den passenden Netzzugang zuweisen, um Online-Verbindungen zu den Zielsystemen aufbauen zu können. Folgende Schnittstellen werden automatisch unterstützt:

- MPI
- PROFIBUS
- Industrial Ethernet (ISO und TCP/IP)

Die Schnittstellen lassen sich in unterschiedlichem Umfang parametrieren. Welche Möglichkeiten zur Parametrierung bestehen, erfahren Sie in den folgenden Kapiteln.

Hinweis

Beachten Sie, dass Änderungen an Schnittstellenparametern direkten Einfluss auf das Betriebssystem und den PG/PC haben. Insofern sind manche Parameter nur mit ausreichend Nutzerrechten änderbar.

Siehe auch

Parametrieren der Industrial Ethernet-Schnittstelle (Seite 41)

Parametrieren der MPI- und PROFIBUS-Schnittstelle (Seite 46)

5.3 Schnittstellen anzeigen oder verbergen

Um die Übersichtlichkeit zu verbessern, lassen sich einzelne Schnittstellen des PG/PC in der Projektnavigation verbergen und bei Bedarf wieder anzeigen.

Vorgehen

Um einzelne Schnittstellen anzuzeigen oder zu verbergen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Online-Zugänge" in der Projektnavigation.
2. Doppelklicken Sie auf das Symbol "Schnittstellen anzeigen/verbergen".
   Der Dialog "Schnittstellen anzeigen/verbergen" wird geöffnet.
3. Aktivieren oder deaktivieren Sie die gewünschten Schnittstellen in der Spalte "In der Projektnavigation anzeigen".
4. Klicken Sie auf die Schaltfläche "Anwenden".
   Die Änderungen werden übernommen. Die Ansicht der Schnittstellen im Ordner "Online-Zugänge" wird aktualisiert.

5.4 Eigenschaften einer Schnittstelle anzeigen und ändern

Einführung

Für jede Schnittstelle können Sie Eigenschaften anzeigen und gegebenenfalls ändern, z. B. Netztyp, Adresse und Status.

Vorgehen

Um die Eigenschaften zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation unter "Online-Zugänge" mit der rechten Maustaste auf die gewünschte Schnittstelle.
2. Wählen Sie aus dem Kontextmenü den Befehl "Eigenschaften".
   Der Dialog mit den Eigenschaften der Schnittstelle wird geöffnet. Auf der linken Seite des Dialoges finden Sie die Bereichsnavigation. Unter den einzelnen Einträgen in der Bereichsnavigation können Sie die aktuellen Parameter anschauen und bei Bedarf ändern.

5.5 Schnittstellen hinzufügen

Sie haben die Möglichkeit nach der Installation des TIA Portals weitere Schnittstellen zu installieren.

Vorgehen

Um eine Schnittstelle nachträglich zu installieren und im TIA Portal hinzuzufügen, gehen Sie folgendermaßen vor:

1. Installieren oder aktualisieren Sie die Treiber im Betriebssystem, nachdem Sie die Schnittstellen hardwareseitig installiert haben.
2. Schließen Sie das TIA Portal, falls es noch geöffnet ist.
3. Öffnen Sie die Windows-Systemsteuerung.
4. Öffnen Sie den Eintrag "PG/PC-Schnittstelle einstellen" in der Systemsteuerung.
   Der Dialog "PG/PC-Schnittstelle einstellen" wird geöffnet.
5. Nehmen Sie gegebenenfalls Änderungen an der Schnittstellenkonfiguration vor und bestätigen Sie die Änderungen mit "OK". Klicken Sie auf jeden Fall auf die Schaltfläche "OK", auch wenn Sie keine Änderungen vorgenommen haben.
6. Starten Sie das TIA Portal neu.

Ergebnis

Die neu installierte Schnittstelle wird nun in der Projektnavigation unter dem Ordner "Online-Zugänge" angezeigt.

5.6 Ethernet-Schnittstelle parametrieren

5.6.1 Parametrieren der Industrial Ethernet-Schnittstelle

Möglichkeiten zur Parametrierung der Industrial Ethernet-Schnittstelle

Für die Parametrierung der Industrial Ethernet-Schnittstelle haben Sie folgende Möglichkeiten:

- Vom Betriebssystem bestimmte Parameter
  Die Industrial Ethernet-Schnittstelle besitzt Parameter, die im Betriebssystem eingestellt werden und für alle angeschlossenen Geräte gültig sind. Diese Parameter werden nur angezeigt, können aber in den Netzwerkeinstellungen des Betriebssystems geändert werden.
- In der Software einstellbare Parameter

Hinweis

Beachten Sie, dass Änderungen an Schnittstellenparametern direkten Einfluss auf das Betriebssystem und den PG/PC haben. Insofern sind manche Parameter nur mit ausreichend Nutzerrechten änderbar.

Parameter der Industrial Ethernet-Schnittstelle

Die folgende Tabelle enthält einen Überblick, welche Parameter der Industrial Ethernet-Schnittstelle vom Betriebssystem vorgegeben sind und welche vom Nutzer geändert werden können.

Nicht änderbare Parameter	Einstellbare Parameter
MAC-Adresse	Fast Acknowledge am IE-PG-Zugang sowie für TCP/IP
DHCP-Server aktiviert/deaktiviert	Timeout am IE-PG-Zugang sowie für TCP/IP
APIPA aktiviert/deaktiviert	LLDP
IP-Adresse	Zusätzliche, temporäre IP-Adressen für die Netzwerkkarte
Subnetzmaske	-
DNS-Adressen	-
DHCP-Adressen	-

Siehe auch

Grundlagen zur Parametrierung der PG/PC-Schnittstelle (Seite 37)

Betriebssystemparameter anzeigen (Seite 42)

PG/PC-Schnittstelle mit einem Subnetz verbinden (Seite 42)

Ethernet-Schnittstelle parametrieren (Seite 43)

Nutzung von temporären IP-Adressen (Seite 44)

Temporäre IP-Adressen verwalten (Seite 44)

5.6.2 Betriebssystemparameter anzeigen

Die Ethernet-Schnittstelle ist Teil des Betriebssystems. Die Parameter der Netzwerkkarte können daher komplett in den Netzwerkeinstellungen des Betriebssystems angepasst werden.

Sie können sich die folgenden Parameter in der Software anzeigen lassen:

- Physikalische Adresse der Netzwerkkarte
- Vergabe der IP-Adresse durch einen DHCP-Server aktiviert oder deaktiviert
- Vergabe einer privaten IP-Adresse durch das Betriebssystem aktiviert oder deaktiviert
- Aktuell fest zugewiesene IP-Adresse
- Zugeordnete Subnetzmaske
- DNS-Adressen
- DHCP-Adressen

Wie sie die Parameter ändern können, entnehmen Sie bitte der Dokumentation des Betriebssystems oder der Netzwerkkarte.

Aktuelle Parameter der Ethernet-Schnittstelle anzeigen

Um sich die aktuellen Parameter der Ethernet-Schnittstelle anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie in der Bereichsnavigation "Konfigurationen > Industrial Ethernet".

Siehe auch

Ethernet-Schnittstelle parametrieren (Seite 43)

5.6.3 PG/PC-Schnittstelle mit einem Subnetz verbinden

Wenn Sie mehrere Subnetze angelegt haben, können Sie festlegen, mit welchem Subnetz die Ethernet-Schnittstelle verbunden werden soll.

Vorgehen

Um das Subnetz zu wählen, mit dem die Ethernet-Schnittstelle verbunden ist, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem Sie die Ethernet-Schnittstelle des PG/PC verbinden möchten.
4. Schließen Sie den Dialog mit "OK".

5.6.4 Ethernet-Schnittstelle parametrieren

Einige das Netzwerkprotokoll betreffende Parameter können Sie direkt innerhalb der Software anpassen.

Voraussetzung

Sie müssen über ausreichend Benutzerrechte verfügen.

Siehe auch: Einfluss von Benutzerrechten.

Vorgehen

Um Parameter zu ändern, die das Netzwerkprotokoll betreffen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > IE-PG-Zugang", um für die Netzwerkverwaltung relevante Protokolleinstellungen anzupassen.
   - Aktivieren Sie das Optionskästchen "Fast Acknowledge", um schnellere Reaktionszeiten bei kleineren Netzwerkpaketen zu erreichen.
   - Wählen Sie aus der Klappliste "Timeout" die Zeit aus, die bis zur Erkennung eines Netzwerkteilnehmers maximal vergehen darf.
4. Um das LLDP-Protokoll zur genaueren Bestimmung der Netzwerktopologie zu aktivieren, aktivieren Sie unter "Konfigurationen > LLDP" das Optionskästchen "LLDP aktiv".
5. Wählen Sie "Konfigurationen > TCP/IP", um das TCP/IP-Protokoll für den Netzwerkverkehr während des Betriebs anzupassen.
   - Aktivieren Sie das Optionskästchen "Fast Acknowledge", um schnellere Reaktionszeiten bei kleineren Netzwerkpaketen zu erreichen.
   - Wählen Sie aus der Klappliste "Timeout" die Zeit aus, die bei der Kommunikation mit einem Netzwerkteilnehmer maximal vergehen darf, bis es zu einer Zeitüberschreitung kommt.

Siehe auch

Betriebssystemparameter anzeigen (Seite 42)

5.6.5 Nutzung von temporären IP-Adressen

Hinzufügen einer temporären IP-Adresse

Befindet sich die IP-Adresse eines Geräts in einem anderen Subnetz als die IP-Adresse der Netzwerkkarte, müssen Sie der Netzwerkkarte zunächst eine zusätzliche IP-Adresse mit derselben Subnetzadresse wie der des Geräts zuweisen. Erst danach ist eine Kommunikation zwischen dem Gerät und dem PG/PC möglich.

Das Zuweisen einer zusätzlichen, temporären IP-Adresse wird gegebenenfalls automatisch vorgeschlagen, wenn Sie eine Online-Aktion durchführen möchten und sich die gegenwärtige IP-Adresse des PG/PC noch nicht im richtigen Subnetz befindet.

Eine temporär zugewiesene IP-Adresse ist bis zum nächsten Neustart des PG/PC gültig oder bis sie manuell gelöscht wird. Siehe auch: Temporäre IP-Adresse verwalten (Seite 44)

Hinweis

Sie müssen über ausreichend Benutzerrechte verfügen, wenn Sie eine temporäre IP-Adresse zuweisen möchten.

Siehe auch: Einfluss von Benutzerrechten

5.6.6 Temporäre IP-Adressen verwalten

Befindet sich die IP-Adresse eines Geräts in einem anderen Subnetz als die aktuell fest zugewiesene IP-Adresse der Netzwerkkarte, wird der Netzwerkkarte temporär eine passende IP-Adresse aus dem Subnetz des Geräts zugewiesen.

Sie können sich alle temporär zugewiesenen Adressen anzeigen lassen und diese löschen. Beachten Sie, dass IP-Adressen, welche Sie im Betriebssystem manuell zugewiesen haben, nicht im TIA Portal angezeigt werden.

Voraussetzung

Für das Löschen sind ausreichend Benutzerrechte nötig.

Vorgehen

Um temporär zugewiesene Adressen anzuzeigen und zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > IE-PG-Zugang".
   Eine Tabelle mit den zugewiesenen IP-Adressen wird angezeigt.
4. Klicken Sie auf die Schaltfläche "Lösche projektspezifische IP-Adressen", um alle IP-Adressen auf einmal zu löschen.

5.6.7 TCP/IP-Konfiguration zurücksetzen

Haben Sie Änderungen an den TCP/IP-Protokolleinstellungen vorgenommen, können Sie diese auf die Standardwerte zurücksetzen.

Vorgehen

Um die Standwerte für die TCP/IP-Konfiguration wiederherzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > TCP/IP".
4. Klicken Sie auf die Schaltfläche "Standard", um alle Einstellungen zurückzusetzen.

5.7 MPI- und PROFIBUS-Schnittstelle parametrieren

5.7.1 Parametrieren der MPI- und PROFIBUS-Schnittstelle

Möglichkeiten zur Parametrierung der MPI- und PROFIBUS-Schnittstelle

Die folgenden Möglichkeiten stehen zur Parametrierung der MPI- und PROFIBUS-Schnittstelle zur Verfügung:

- Automatische Konfiguration: Sie können automatisch erkennen lassen, ob ein Gerät über PROFIBUS oder MPI mit der PG/PC-Schnittstelle verbunden ist.
- Auswahl einer voreingestellten Konfiguration für PROFIBUS- oder MPI, die nachträglich angepasst werden kann.

Geräte- und netzbezogene Parametriermöglichkeiten für MPI und PROFIBUS

Sie können bei MPI- und PROFIBUS-Schnittstellen gerätebezogene und netzbezogene Parameter einstellen. Gerätebezogene Parameter sind lokale Einstellungen der Schnittstelle. Netzbezogene Parameter dagegen müssen Geräte übergreifend übereinstimmen.

Veränderbare Parameter an der MPI-Schnittstelle

Für die MPI-Schnittstelle können Sie die folgenden voreingestellten Parameter anpassen:

Gerätebezogene Parameter	Netzbezogene Parameter
Ist einziger Master	Höchste Adresse
Eigene Adresse	Transferrate
Timeout	

Veränderbare Parameter an der PROFIBUS-Schnittstelle

Für die PROFIBUS-Schnittstelle können Sie die folgenden voreingestellten Parameter anpassen:

Gerätebezogene Parameter	Netzbezogene Parameter
Ist einziger Master	Höchste Adresse
Eigene Adresse	Transferrate
Timeout	Profil
	Busparameter
	Anzahl der Master am Bus
	Anzahl der Slaves am Bus

Hinweis

Verhalten bei einem zugewiesenen Subnetz

Wenn Sie der Schnittstelle ein Subnetz aus dem Projekt fest zuweisen, dann haben die Subnetzeinstellungen des Projekts Vorrang. In diesem Fall sind die Einstellungen für die jeweilige Schnittstelle nicht änderbar.

Siehe auch

Grundlagen zur Parametrierung der PG/PC-Schnittstelle (Seite 37)

5.7.2 MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren

Automatische Erkennung der Busparameter einrichten

Wenn Sie eine Schnittstelle mit automatischer Erkennung der Busparameter wählen, z. B. CP5611 (Auto), dann können Sie den PG/PC an MPI oder PROFIBUS anschließen, ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minute entstehen.

Voraussetzung

- Am Bus sind Master angeschlossen, die zyklisch Busparameter verteilen.
- Bei PROFIBUS-Netzen ist das zyklische Verteilen der Busparameter aktiviert.
- Die Schnittstelle ist keinem Subnetz zugeordnet. Wenn Sie in den Eigenschaften der Schnittstelle ein Subnetz zuordnen, dann haben die Einstellungen für das Subnetz im Projekt Vorrang. In diesem Fall sind die Einstellungen zur automatischen Schnittstellenkonfiguration nicht änderbar.

Vorgehen

Um die automatische Erkennung der Busparameter zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Konfigurationen > Aktive Konfiguration" die Einstellung "Automatische Protokollerkennung".
4. Wählen Sie unter "Konfigurationen > Auto-Konfiguration > Lokale Einstellungen" aus der Klappliste "Eigene Adresse" die Adresse der PG/PC-Schnittstelle aus.
5. Wenn Sie anschließend die aktuellen Buseinstellungen anzeigen möchten, klicken Sie auf die Schaltfläche "Netzwerkerkennung".

Siehe auch

MPI-Schnittstelle parametrieren (Seite 48)

PROFIBUS-Schnittstelle parametrieren (Seite 50)

5.7.3 MPI-Schnittstelle parametrieren

Parameter der MPI-Schnittstelle ändern

Für die MPI-Schnittstelle lassen sich netzbezogene Parameter und Busparameter anpassen. Zunächst sollten Sie eine Voreinstellung wählen und diese bei Bedarf genauer anpassen.

Voraussetzung

Die Schnittstelle ist keinem Subnetz zugeordnet. Wenn Sie in den Eigenschaften der Schnittstelle ein Subnetz zuordnen, dann haben die Einstellungen für das Subnetz im Projekt Vorrang. In diesem Fall lässt sich die Schnittstellenkonfiguration nicht ändern.

MPI-Schnittstelle mit Voreinstellungen parametrieren

Um die Parameter der MPI-Schnittelle anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem die Schnittstelle verbunden werden soll.
4. Wählen Sie unter "Allgemein > Konfigurationen" eine Voreinstellung für die geräte- und netzbezogenen Parameter. Die Voreinstellungen sind für die meisten Konfigurationen geeignet. Wählen Sie eine der folgenden Einstellungen:
   - Automatische Protokollerkennung
     Sie können das PG an MPI bzw. PROFIBUS anschließen, ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minute entstehen. Voraussetzung für die automatische Erkennung ist, dass am Bus Master angeschlossen sind, die zyklisch Busparameter verteilen. Bei PROFIBUS-Subnetzen darf das zyklische Verteilen der Busparameter nicht ausgeschaltet sein (voreingestellte PROFIBUS-Netzeinstellung).
   - MPI
     Das Übertragungsprotokoll "MPI" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.
   - PROFIBUS
     Das Übertragungsprotokoll "PROFIBUS" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.

Voreingestellte Parameter ändern

Um die Voreinstellungen an Ihre Bedürfnisse anzupassen, ändern Sie die entsprechenden Parameter bei Bedarf unter "Konfigurationen > MPI" ab.

Folgende gerätebezogenen Parameter können Sie einstellen:

- Ist einziger Master
  Hiermit wird eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz deaktiviert, weil das PG/PC der einzige Master am Bus ist.
  - Aktivieren Sie diese Option nur, wenn Sie ausschließlich Slaves an Ihr PG oder Ihren PC angeschlossen haben.
  - Wenn das Optionskästchen "Ist einziger Master" aktiviert ist, ist die Ermittlung des direkt angeschlossenen Gerätes über das Fenster "Erreichbare Teilnehmer" nicht möglich.
- Eigene Adresse
  Diese Angabe bezieht sich auf das PG beziehungsweise den PC, auf dem Sie die Parametrierung der Schnittstelle aufgerufen haben. Stellen Sie hier die lokale Teilnehmeradresse Ihres PG/PC ein.
  - Die Adresse muss netzweit eindeutig sein.
  - Unter dieser Adresse wird der PG/PC im MPI-Netz angesprochen.
- Überprüfen
  Hiermit aktivieren Sie eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz. Der Treiber prüft, ob die eigene Adresse bereits durch einen anderen Teilnehmer belegt ist. Hierbei werden neben den aktiven auch die passiven Teilnehmer berücksichtigt. Der Treiber hört dabei am PROFIBUS mit. Bei der automatischen Überprüfung dauert das Ankoppeln des PG/PC an das Netz länger. Um das Überprüfen zu nutzen, muss der Treiber die Funktion unterstützen. Weiterhin darf die Option "Ist einziger Master" nicht aktiviert sein.
- Timeout
  Erhöhen Sie den Wert der Verbindungsüberwachungszeit, wenn Sie z. B. Probleme mit hohen Antwortzeiten im Netz haben.

Folgende netzbezogenen Parameter können Sie einstellen:

- Höchste Adresse:
  Wählen Sie hier die projektierte höchste Teilnehmeradresse. Stellen Sie sicher, dass bei allen Teilnehmern eines MPI-Netzes die gleiche höchste Teilnehmeradresse eingestellt ist.
- Transferrate:
  Wählen Sie hier die Übertragungsgeschwindigkeit, die im MPI-Netz verwendet werden soll.

Siehe auch

MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren (Seite 47)

5.7.4 PROFIBUS-Schnittstelle parametrieren

Parameter der PROFIBUS-Schnittstelle ändern

Für die PROFIBUS-Schnittstelle lassen sich netzbezogene Parameter und Busparameter genauer anpassen. Zunächst sollten Sie eine Voreinstellung wählen und diese bei Bedarf genauer anpassen.

Voraussetzung

Die Schnittstelle ist keinem Subnetz zugeordnet. Wenn Sie in den Eigenschaften der Schnittstelle ein Subnetz zuordnen, dann haben die Einstellungen für das Subnetz im Projekt Vorrang. In diesem Fall lässt sich die Schnittstellenkonfiguration nicht ändern.

PROFIBUS-Schnittstelle mit Voreinstellungen parametrieren

Um die Parameter der PROFIBUS-Schnittelle anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem die Schnittstelle verbunden werden soll.
4. Wählen Sie unter "Allgemein > Konfiguration" eine Voreinstellung für die geräte- und netzbezogenen Parameter. Die Voreinstellungen sind für die meisten Konfigurationen geeignet. Wählen Sie eine der folgenden Einstellungen:
   - Automatische Protokollerkennung
     Sie können das PG an MPI bzw. PROFIBUS anschließen, ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minute entstehen. Voraussetzung für die automatische Erkennung ist, dass am Bus Master angeschlossen sind, die zyklisch Busparameter verteilen. Bei PROFIBUS-Subnetzen darf das zyklische Verteilen der Busparameter nicht ausgeschaltet sein (voreingestellte PROFIBUS-Netzeinstellung).
   - MPI
     Das Übertragungsprotokoll "MPI" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.
   - PROFIBUS
     Das Übertragungsprotokoll "PROFIBUS" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.

Voreingestellte Parameter ändern

Um die Voreinstellungen an Ihre Bedürfnisse anzupassen, ändern Sie die entsprechenden Parameter bei Bedarf unter "Konfigurationen > PROFIBUS" ab.

Folgende gerätebezogenen Parameter können Sie einstellen:

- Ist einziger Master
  Hiermit wird eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz deaktiviert, weil das PG/PC der einzige Master am Bus ist.
  - Aktivieren Sie diese Option nur, wenn Sie ausschließlich Slaves an Ihr PG oder Ihren PC angeschlossen haben.
  - Wenn das Optionskästchen "Ist einziger Master" aktiviert ist, ist die Ermittlung des direkt angeschlossenen Gerätes über das Fenster "Erreichbare Teilnehmer" nicht möglich.
- Eigene Adresse
  Diese Angabe bezieht sich auf das PG beziehungsweise den PC, auf dem Sie die Parametrierung der Schnittstelle aufgerufen haben. Stellen Sie hier die lokale Teilnehmeradresse Ihres PG/PC ein.
  - Die Adresse muss netzweit eindeutig sein.
  - Unter dieser Adresse wird der PG/PC im PROFIBUS-Netz angesprochen.
- Überprüfen
  Hiermit aktivieren Sie eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz. Der Treiber prüft, ob die eigene Adresse bereits durch einen anderen Teilnehmer belegt ist. Hierbei werden neben den aktiven auch die passiven Teilnehmer berücksichtigt. Der Treiber hört dabei am PROFIBUS mit. Bei der automatischen Überprüfung dauert das Ankoppeln des PG/PC an das Netz länger. Um das Überprüfen zu nutzen, muss der Treiber die Funktion unterstützen. Weiterhin darf die Option "Ist einziger Master" nicht aktiviert sein.
- Timeout
  Erhöhen Sie den Wert der Verbindungsüberwachungszeit, wenn Sie z. B. Probleme mit hohen Antwortzeiten im Netz haben.

Folgende netzbezogenen Parameter können Sie einstellen:

- Höchste Adresse:
  Wählen Sie hier die projektierte höchste Teilnehmeradresse. Stellen Sie sicher, dass bei allen Teilnehmern eines PROFIBUS-Netzes die gleiche höchste Teilnehmeradresse eingestellt ist.
- Transferrate:
  Wählen Sie hier die Übertragungsgeschwindigkeit, die im PROFIBUS-Netz verwendet werden soll.
- Profil:
  Sie können zwischen vier Alternativen für die Einstellungen für den PROFIBUS wählen. "DP", "Standard" und "Universal (DP/FMS)" sind vordefinierte Einstellungen, die Sie nicht ändern können. Mit der Auswahl von "Benutzerdefiniert" können Sie die Busparameter selbst anpassen.
  - Wenn Sie "Benutzerdefiniert" gewählt haben, wechseln Sie in der Bereichsnavigation auf "Konfigurationen > PROFIBUS > Busparameter".
  - Wenn Sie eine der Voreinstellungen (DP, Standard oder Universal (DP/FMS)) ausgewählt haben, sollten Sie unter "Konfigurationen > PROFIBUS > Busparameter > Zusätzliche Parameter" das Optionskästchen "Berücksichtigen" aktivieren. Sie können dann die Anzahl der Master und Slaves am Bus einstellen. Die Busparameter werden auf diese Weise genauer berechnet und eventuelle Busstörungen können ausgeschlossen werden. Bei einem benutzerdefinierten Profil kann die Option nicht aktiviert werden.

Siehe auch

Übersicht der Busparameter für PROFIBUS (Seite 52)

MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren (Seite 47)

5.7.5 Übersicht der Busparameter für PROFIBUS

Einführung

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Sie sollten die voreingestellten Werte deshalb nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind.

Die Busparameter sind in Abhängigkeit vom Busprofil einstellbar oder nicht einstellbar. Es werden immer die Offline-Werte der Busparameter angezeigt, auch wenn Sie online mit dem Zielsystem verbunden sind.

Die angezeigten Parameter gelten für das gesamte PROFIBUS-Subnetz.

Bedeutung der einzelnen Parameter

- Tslot: Warte-auf-Empfangszeit (slot time)
  Die Warte-auf-Empfang-Zeit (slot-time) legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten.
- Max. Tsdr: maximale Protokollbearbeitungszeit (max. station delay responder)
  Die maximale Protokollbearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer das Protokoll bearbeitet haben muss.
- Min. Tsdr: minimale Protokollbearbeitungszeit (min. station delay responder)
  Die minimale Protokollbearbeitungszeit legt fest, welche Zeit der antwortende Teilnehmer mindestens benötigt, um das Protokoll zu bearbeiten.
- Tset: Auslösezeit (setup-Time)
  Die Auslösezeit ist die Zeit, die zwischen dem Empfang eines Datentelegramms und der Reaktion darauf verstreichen darf.
- Tqui: Modulator-Ausklingzeit (Quiet-Time for Modulator)
  Die Modular-Ausklingzeit ist die Zeit, die für das Umstellen von Senden auf Empfangen benötigt wird.
- GAP-Faktor: GAP-Aktualisierungsfaktor (GAP-Faktor)
  Der GAP-Aktualisierungsfaktor legt fest, nach wie viel Token-Umläufen ein hinzugekommener Teilnehmer in den Tokenring aufgenommen wird.
- Retry Limit: maximale Anzahl von Aufrufwiederholungen (retry limit)
  Mit dem Parameter wird festgelegt, wie viele Versuche maximal unternommen werden, um einen Teilnehmer zu erreichen.
- Trdy: Bereitschaftszeit (ready-time)
  Die Bereitschaftszeit ist die Zeit für Quittung oder Antwort.
- Tid1: Ruhezustandszeit 1 (idle-time 1)
  Die Ruhezustandszeit 1 legt die Verzögerungszeit nach Empfang einer Antwort fest.
- Tid2: Ruhezustandszeit 2 (idle-time 2)
  Die Ruhezustandszeit 2 legt die Verzögerungszeit nach Senden eines Aufrufs ohne Antwort fest.
- Ttr: Soll-Token-Umlaufzeit (Target-Rotation-Time)
  Die Soll-Token-Umlaufzeit ist die maximal zur Verfügung gestellte Zeit für einen Token-Umlauf. In dieser Zeit erhalten alle aktiven Teilnehmer (Master) einmal das Senderecht (Token). Die Differenz zwischen Soll-Token-Umlaufzeit und tatsächlicher Token-Umlaufzeit bestimmt, wie viel Zeit den Mastern für das Senden von Datentelegrammen an die Slaves übrig bleibt.
  Wählen Sie als minimale Target-Rotation-Time (Ttr) = 5000 mal HSA (Highest Station Address).
- Ansprechüberwachung: Ansprechüberwachungszeit (Watchdog)
  Die Ansprechüberwachung legt fest, nach welcher Zeit ein Teilnehmer angesprochen werden muss.
  Wählen Sie als minimale Ansprechüberwachungszeit = 6250 mal HSA.

Hinweis

Wenn Sie ein benutzerdefiniertes Busprofil erstellen wollen, dann beachten Sie, dass die minimale Target-Rotation-Time (Ttr) 5000 mal HSA (höchste PROFIBUS-Adresse) betragen sollte. Außerdem sollte die minimale Ansprechüberwachung (Watchdog) 6250 mal HSA betragen.

Siehe auch

PROFIBUS-Schnittstelle parametrieren (Seite 50)

5.7.6 MPI- oder PROFIBUS Konfiguration zurücksetzen

Haben Sie Änderungen an den MPI- oder PROFIBUS-Protokolleinstellungen vorgenommen, können Sie diese auf die Standardwerte zurücksetzen.

Voraussetzung

Die Schnittstelle ist keinem Subnetz zugeordnet. Wenn Sie in den Eigenschaften der Schnittstelle ein Subnetz zuordnen, dann haben die Einstellungen für das Subnetz im Projekt Vorrang. In diesem Fall lässt sich die Schnittstellenkonfiguration nicht auf die Standardwerte zurücksetzen.

Vorgehen

Um die Standardwerte für die MPI- oder PROFIBUS-Konfiguration wiederherzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die MPI/PROFIBUS-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".
   Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > MPI" oder "Konfigurationen > PROFIBUS", abhängig davon, welche Schnittstelleneigenschaften Sie zurücksetzen möchten.
4. Klicken Sie auf die Schaltfläche "Standard", um alle Einstellungen zurückzusetzen.

Trace- und Logikanalysatorfunktion nutzen 6

6.1 Vorwort

Zweck der Dokumentation

In dieser Dokumentation werden die Diagnosemöglichkeiten mit der Trace- und Logikanalysatorfunktion beschrieben. Abhängig vom eingesetzten Gerät können sich die Aufzeichnungsmöglichkeiten unterscheiden.

Erforderliche Grundkenntnisse

Zum Verständnis der Dokumentation sind die folgenden Kenntnisse erforderlich:

- Allgemeine Kenntnisse auf dem Gebiet der Automatisierungstechnik
- Kenntnisse über die Verwendung von Windows-Computern
- S7-1200/1500 CPUs, ET 200SP, ET 200Pro
  - Kenntnisse des Industrieautomatisierungssystems SIMATIC
  - Kenntnisse im Umgang mit STEP 7
- SINAMICS Antriebe
  - Kenntnisse im Umgang mit dem Antrieb
- SIRIUS SIMOCODE pro, SIRIUS Sanftstarter 3RW
  - Kenntnisse im Umgang mit diesen Systemen

Gültigkeitsbereich der Dokumentation

Diese Dokumentation gilt für alle Produkte der Produktfamilie S7-1200, S7-1500, S7-1500 Software Controller, ET 200SP, CPU 1516(F)pro-2 PN, SINAMICS Antriebe, SIRIUS SIMOCODE pro und SIRIUS Sanftstarter 3RW ab TIA-Portal V15.

Konventionen

Die vorliegende Dokumentation enthält Abbildungen zu den beschriebenen Geräten. Die Abbildungen können vom gelieferten Gerät in Einzelheiten abweichen.

Beachten Sie auch die folgendermaßen gekennzeichneten Hinweise:

Hinweis

Ein Hinweis enthält wichtige Informationen zum in der Dokumentation beschriebenen Produkt, zur Handhabung des Produkts oder zu dem Teil der Dokumentation, auf den besonders aufmerksam gemacht werden soll.

"mySupport"

Mit "mySupport", Ihrem persönlichen Arbeitsbereich, machen Sie das Beste aus Ihrem Industry Online Support.

In "mySupport" können Sie Filter, Favoriten und Tags ablegen, CAx-Daten anfordern und sich im Bereich Dokumentation Ihre persönliche Bibliothek zusammenstellen. Des Weiteren sind in Support-Anfragen Ihre Daten bereits vorausgefüllt und Sie können sich jederzeit einen Überblick über Ihre laufenden Anfragen verschaffen.

Um die volle Funktionalität von "mySupport" zu nutzen, müssen Sie sich einmalig registrieren.

Sie finden "mySupport" im Internet (https://support.industry.siemens.com/My/ww/de/).

"mySupport" - Dokumentation

In "mySupport" haben Sie im Bereich Dokumentation die Möglichkeit ganze Handbücher oder nur Teile daraus zu Ihrem eigenen Handbuch zu kombinieren.
Sie können das Handbuch als PDF-Datei oder in einem nachbearbeitbaren Format exportieren.

Sie finden "mySupport" - Dokumentation im Internet (http://support.industry.siemens.com/My/ww/de/documentation).

Weitere Unterstützung

- Das Angebot an technischer Dokumentation für die einzelnen SIMATIC Produkte und Automatisierungssysteme finden Sie im Internet (http://www.siemens.com/simatic-tech-doku-portal).
- Den Online-Katalog und das Online-Bestellsystem finden Sie im Internet (https://mall.industry.siemens.com).

Security-Hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies

notwendig ist und entsprechende Schutzmaßnahmen (z.B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Industrial Security finden Sie unter (https://www.siemens.com/industrialsecurity).

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter (https://www.siemens.com/industrialsecurity).

6.2 Beschreibung

6.2.1 Unterstützte Hardware

Wenn ein Gerät die Trace- und Logikanalysatorfunktion unterstützt, wird in der Projektnavigation unter dem Gerät "Traces" zur Auswahl angeboten.

Folgende Geräte (Seite 111) unterstützen die Trace- und Logikanalysatorfunktion:

- SIMATIC S7-1200 CPUs (ab Firmwarestand V4.0)
- SIMATIC S7-1500, ET 200SP und CPU 1516pro-2 PN CPUs
- SIMATIC S7-1500 Software Controller
- SINAMICS Antriebe, die in Startdrive unterstützt werden
- SIRIUS SIMOCODE pro (mit Simocode ES)
- SIRIUS Sanftstarter 3RW (mit Soft Starter ES)

6.2.2 Messwertaufzeichnung mit der Tracefunktion

Einführung

Die Trace- und Logikanalysatorfunktion rufen Sie im Projektnavigator (Seite 63) durch Doppelklick auf einen Eintrag im Systemordner "Traces" auf. Zusätzlich können über die Diagnoseschnittstelle des Webservers die Messungen auf der Speicherkarte ausgelesen und angezeigt werden.

Mit der Trace- und Logikanalysatorfunktion zeichnen Sie Variablen eines Geräts auf und werten die Aufzeichnungen aus. Variablen sind z. B. Antriebsparameter oder System- und Anwendervariablen einer CPU. Die Anzahl der Traces im Gerät ist hardwareabhängig.

Die Aufzeichnungen werden auf dem Gerät gespeichert und können bei Bedarf mit dem Projektierungssystem (ES) ausgelesen und dauerhaft gespeichert werden. Somit eignet sich die Trace- und Logikanalysatorfunktion zum Beobachten hochdynamischer Vorgänge. Die aufgezeichneten Werte werden bei erneuter Aktivierung der Aufzeichnung überschrieben.

Die Trace- und Logikanalysatorfunktionen werden auch in den Inbetriebnahmeeditoren von Technologieobjekten (z. B. Achssteuertafel) verwendet. Aktive Aufzeichnungen von der Achssteuertafel werden als Traces im Gerät im Systemordner "Traces" angezeigt. Im Kurvendiagramm der Achssteuertafel bzw. von PID können Aufzeichnungen über Kontextmenübefehl zu den Messungen hinzugefügt werden.

Abhängig vom eingesetzten Gerät (Seite 111) können sich die Aufzeichnungsmöglichkeiten unterscheiden.

Im Kapitel Bedienen finden Sie einen Schnelleinstieg (Seite 90) für das Arbeiten mit der Trace- und Logikanalysatorfunktion.

Das folgende Bild zeigt die Funktionsweise von "Traces":

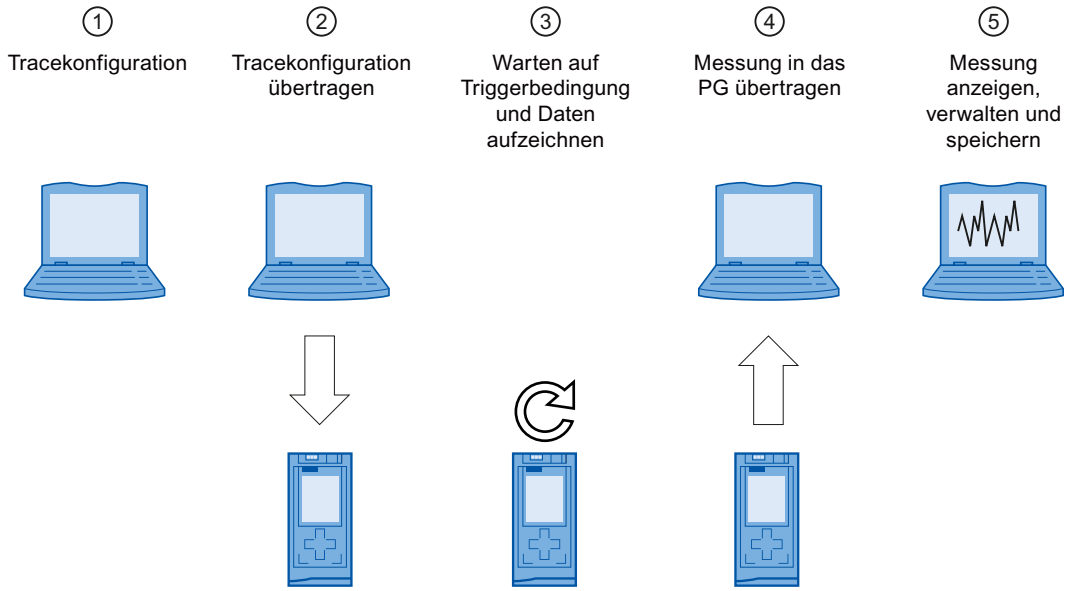

① Tracekonfiguration auf dem Programmiergerät (PG) im TIA-Portal

In der Tracekonfiguration legen Sie die aufzuzeichnenden Signale, die Dauer der Aufzeichnung und die Triggerbedingung fest. Die Tracekonfiguration ist geräteabhängig und beim jeweiligen Gerät (Seite 111) beschrieben.

② Übertragen der Tracekonfiguration vom PG auf das Gerät

Die vollständige Tracekonfiguration übertragen (Seite 97) Sie bei bestehender Online-Verbindung in das Gerät.

③ Warten auf die Aufzeichnung

Ist die Tracekonfiguration im Gerät aktiviert (Seite 97), dann erfolgt die Aufzeichnung unabhängig vom PG. Sobald die Triggerbedingung erfüllt ist, wird die Aufzeichnung gestartet.

④ Übertragen der Messung vom Gerät auf das PG

Das Speichern der Messung im Projekt (Seite 99) legt die Messung im geöffneten Projekt des TIA-Portals ab. Speichern der Messung ist unabhängig vom Zeitpunkt der Messung und kann auch jederzeit nach Abschluss der Aufzeichnung erfolgen.

⑤ Messung auswerten, verwalten und speichern

Für die Auswertung der Messung stehen vielfältige Möglichkeiten im Kurvendiagramm und in der Signaltabelle (Seite 98) zur Verfügung. Verschiedene Darstellungsformen sind möglich, z. B. für Binärsignale eine Bit-Darstellung.
Signalverläufe aus unterschiedlichen Messungen können als überlagerte Messung zusammengestellt und miteinander verglichen werden.
Messungen können auch als Datei exportiert und importiert werden.
Mit dem Speichern des Projekts (Seite 99) im TIA-Portal werden auch die in das Projekt übertragenen Messungen gespeichert.

Weitere Features

Folgende Features werden ab TIA-Portal V14 unterstützt:

- Speicherung von Tracekonfigurationen und Messungen in Bibliotheken
- Multiuser-Unterstützung für Tracekonfigurationen und Messungen (geräteabhängig)
- Globale Suche im Projekt

6.2.3 Tracekonfiguration, Aufzeichnung, Trace im Gerät und Messung

Dieses Kapitel erklärt die Bedeutung und Zusammenhänge der Begriffe Tracekonfiguration, Aufzeichnung, Trace im Gerät und Messung.

Tracekonfiguration

In der Tracekonfiguration nehmen Sie folgende Einstellungen vor:

- Aufzuzeichnende Signale
- Aufzeichnungsbedingungen
  - Abtastung
  - Trigger
  - Messungen im Gerät (Speicherkarte)

Aufzeichnung

Eine Aufzeichnung findet im Gerät statt. Zu jeder Tracekonfiguration im Gerät gibt es nur eine Aufzeichnung. Wenn eine Aufzeichnung neu gestartet wird, dann wird die alte Aufzeichnung überschrieben.

Eine Aufzeichnung im Gerät ist nicht remanent (geht nach Aus-/Einschalten des Geräts verloren), kann aber als Messung dauerhaft im Projekt gespeichert werden.

Trace im Gerät

Ein Trace im Gerät besteht aus einer Tracekonfiguration und optional aus einer Aufzeichnung. Die maximale Anzahl Traces im Gerät ist geräteabhängig.

Die Tracekonfiguration ist im Gerät remanent hinterlegt. Geräteabhängig kann die remanente Speicherung der Tracekonfiguration auch konfigurierbar sein, wie z. B. bei S120.

Messung

Eine Messung besteht aus einer Tracekonfiguration und aus einer Aufzeichnung, sofern aufgezeichnete Daten vorhanden sind. Jeder Trace im Gerät kann im Projekt als Messung gespeichert werden.

Die Aufzeichnung einer Messung kann offline betrachtet werden.

Die Konfiguration einer Messung kann durch Drag & Drop in den Ordner "Traces" übernommen werden.

Messungen im Gerät (Speicherkarte)

Im Ordner sind Messungen abgelegt, die im Gerät (z. B. auf der Speicherkarte) gespeichert sind. Diese Messungen sind remanent und können nur vom Anwender gelöscht werden.

Die Messungen im Gerät können durch Drag & Drop in den Ordner Messungen übernommen und damit im Projekt als Messung gespeichert werden.

Tracekonfiguration mit namensgleichem Trace im Gerät

Üblicherweise existiert zu einem Trace im Gerät eine namensgleiche Tracekonfiguration im Projekt. Bei bestehender Online-Verbindung wird dieser Trace mit dem Symbol in der Projektnavigation dargestellt.

Siehe auch Oberfläche - Projektnavigation Ordner "Traces" (Seite 64).

Überlagerte Messung

Die überlagerte Messung ermöglicht den Vergleich und die Analyse von Signalen verschiedener Messungen miteinander.

Die Messungen können zueinander synchronisiert und überlagert dargestellt werden.

6.2.4 Datenablage

Die Funktionsleiste des Trace und das Kurvendiagramm ermöglichen unter anderem das Übertragen der Tracekonfiguration und das Betrachten der Aufzeichnung.

Das folgende Bild zeigt schematisch die Datenablage:

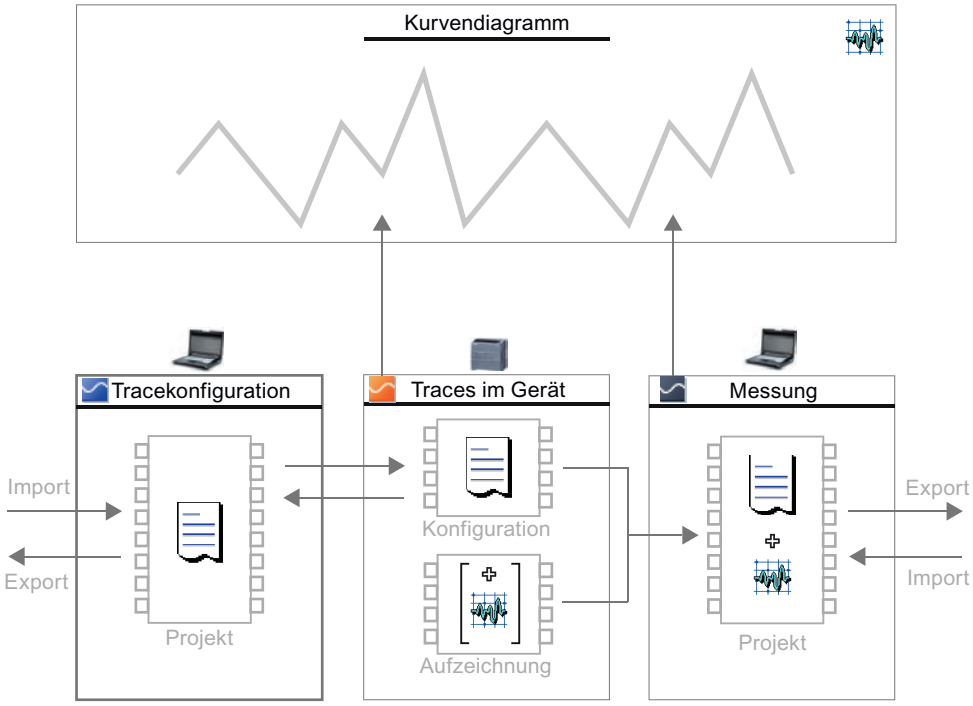

Hinweis

Tracekonfiguration und Messung speichern

Die Tracekonfiguration und Messung speichern Sie mit dem Projekt im TIA-Portal.

Wenn Sie das Projekt ohne Speichern schließen, werden die Tracekonfigurationen und die in das Projekt übertragenen Messungen verworfen. Der Editor der Traces kann bis zum Schließen des Projekts ohne Datenverlust geschlossen und wieder geöffnet werden.

6.3 Software-Oberfläche

Anzeigebereiche

Die Benutzeroberfläche der Trace- und Logikanalysatorfunktion setzt sich aus mehreren Bereichen zusammen.

Das folgende Bild zeigt exemplarisch die Aufteilung der Oberfläche im TIA-Portal:

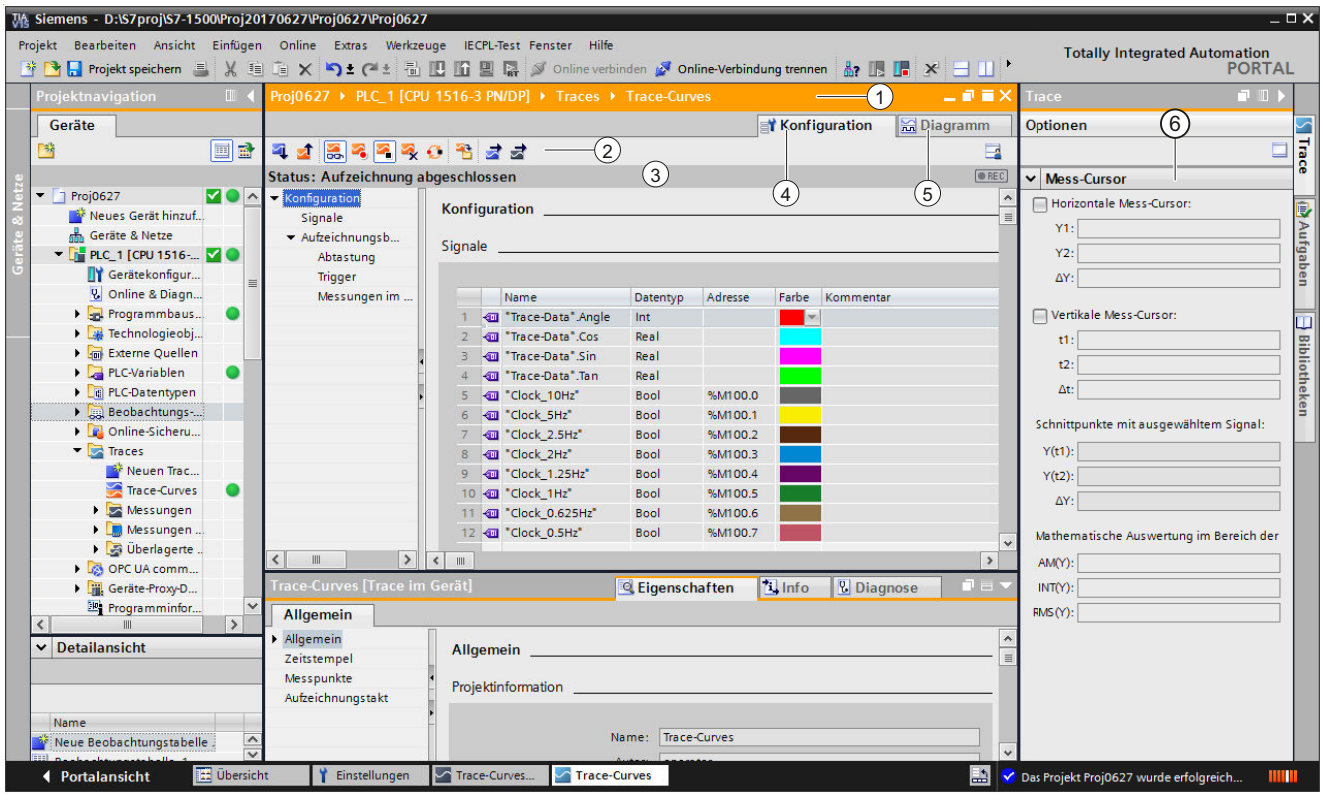

Projektnavigator Verwalten und Anlegen der Traces und Messungen direkt in der Projektnavigation und über Kontextmenübefehle.
Arbeitsbereich

	① Titelleiste des Arbeitsbereichs Zeigt an, zu welchem Gerät die aktuelle Darstellung gehört.
	② Funktionsleiste des Trace Schaltflächen zum Verwalten der Traces im Projekt und Gerät: • Aktivieren/Deaktivieren von Traces im Gerät • Löschen von Traces im Gerät • Übertragen von Tracekonfigurationen und Messungen zwischen Gerät und Projekt • Exportieren von Tracekonfigurationen und Messungen • Umschalten zwischen Offline- und Online-Anzeige
	③ Statusanzeige des Trace Anzeige des aktuellen Status der Aufzeichnung.
	④ Register Konfiguration Gerätespezifische Konfiguration der Aufzeichnungsdauer, Triggerbedingung und Signalauswahl. Siehe gerätespezifische Beschreibungen (Seite 111).
	⑤ Register Diagramm Anzeige der aufgezeichneten Werte als Kurvendiagramm und der Signale von der dargestellten Messung.
	Register Signalauswahl Anzeige aller Signale, die in den überlagerten Messungen enthalten sind.
Task Card "Trace" Anzeige der Mess-Cursordaten mit mathematischer Auswertung ⑥ und Momentaufnahmen.	
Inspektorfenster Anzeige allgemeiner Informationen zu der Tracekonfiguration.	

6.3.1 Projektnavigator

6.3.1.1 Oberfläche - Projektnavigation Ordner "Traces"

Im Ordner "Traces" werden Tracekonfigurationen und Traces im Gerät dargestellt.

Die Unterordner von "Traces" sind in den nachfolgenden Kapiteln beschrieben.

Mit Doppelklick auf einen Trace öffnen Sie das entsprechende Register "Diagramm" oder "Konfiguration" im Arbeitsbereich.

Symbole im Ordner "Traces"

Die folgende Tabelle erklärt die Symbole im Ordner "Traces":

Symbol	Beschreibung
	Tracekonfiguration hinzufügen Mit Doppelklick auf das Symbol fügen Sie eine neue Tracekonfiguration hinzu.
	Tracekonfiguration (offline) Mit Doppelklick auf das Symbol öffnen Sie das Register "Konfiguration".

Symbol	Beschreibung
	Trace im Gerät (online) Das Symbol wird nur angezeigt, wenn zu dem Trace im Gerät keine namensgleiche Offline-Tracekonfiguration existiert. Mit Doppelklick auf das Symbol öffnen Sie das Register "Diagramm".
	Tracekonfiguration mit namensgleichem Trace im Gerät Bei deaktivierter Schaltfläche wird die Tracekonfiguration aus dem Projekt angezeigt. Der Trace entspricht einer Tracekonfiguration. Bei aktivierter Schaltfläche wird die Tracekonfiguration aus dem Gerät angezeigt. Der Trace entspricht einem Trace im Gerät. Mit Doppelklick auf das Symbol öffnen Sie das Register "Diagramm" des Traces im Gerät.

Status

Bei bestehender Online-Verbindung wird der Status in der rechten Spalte der Projektnavigation angezeigt. Der Status wird auch als Tooltip über dem jeweiligen Symbol angezeigt.

Die folgende Tabelle zeigt die Bedeutung der Symbole:

Symbol	Beschreibung
	Online- und Offline-Konfiguration sind identisch
	Online- und Offline-Konfiguration sind verschieden
	Konfiguration existiert nur online

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle für den Systemordner "Traces":

Kontextmenübefehl	Beschreibung
"Neue Gruppe hinzufügen"	Fügt einen neuen Ordner ein.
"Neuen Trace hinzufügen"	Fügt eine neue Tracekonfiguration hinzu und öffnet das Register Konfiguration.
"Tracekonfiguration importieren"	Importiert eine Tracekonfiguration aus einer Datei.

Die folgende Tabelle zeigt die Kontextmenübefehle für Tracekonfigurationen und Traces im Gerät / :

Kontextmenübefehl	Tracekonfiguration	Trace im Gerät	Beschreibung
"Kopieren"	x	-	Kopiert die Tracekonfiguration oder den Trace im Gerät in die Zwischenablage.
"Einfügen"	x	-	Fügt eine Tracekonfiguration oder Messung aus der Zwischenablage ein.
"Löschen"	x	x	Löscht die selektierten Objekte aus der Projektnavigation bzw. aus dem Gerät.

Kontextmenübefehl	Tracekonfiguration	Trace im Gerät	Beschreibung
"Umbenennen"	x	-	Schaltet das selektierte Objekt in den Editier-Modus.
"Tracekonfiguration exportieren"	x	-	Exportiert eine Tracekonfiguration als Datei mit der Dateierweiterung "*.ttcfgx" oder einen Trace im Gerät mit der Dateierweiterung "*.ttrecx". Die Dateierweiterungen "*.ttcfg" und "*.ttrec" werden aus Kompatibilitätsgründen zu TIA-Portal V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.

Das Kopieren der Tracekonfiguration ist bei gleicher Gerätefamilie auch geräteübergreifend möglich.

Das Selektieren mehrerer Objekte ist möglich.

6.3.1.2 Oberfläche - Projektnavigation Ordner "Messungen"

Im Ordner "Messungen" werden die gespeicherten Messungen dargestellt.

Symbole im Ordner "Messungen"

Die folgende Tabelle erklärt die Symbole im Ordner "Messungen":

Symbol	Beschreibung
	Messung (offline) Mit Doppelklick auf das Symbol öffnen Sie das Register "Diagramm". Die Konfiguration einer Messung kann durch Drag & Drop in den Ordner "Traces" übernommen werden.

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle für den Systemordner "Messungen":

Kontextmenübefehl	Beschreibung
"Neue Gruppe hinzufügen"	Fügt einen neuen Ordner ein.
"Messung importieren"	Importiert eine Messung aus einer Datei mit der Dateierweiterung "*.ttrecx". Der Import ist geräteunabhängig. Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.

Die folgende Tabelle zeigt die Kontextmenübefehle für Messungen:

Kontextmenübefehl	Beschreibung
"Kopieren"	Kopiert die Tracekonfiguration der selektierten Objekte in die Zwischenablage.
"Einfügen"	Fügt eine Messung aus der Zwischenablage ein.
"Löschen"	Löscht die selektierten Objekte aus der Projektnavigation bzw. aus dem Gerät.

Kontextmenübe-fehl	Beschreibung
"Umbenennen"	Schaltet das selektierte Objekt in den Editier-Modus.
"Neue überlagerte Messung erstellen"	Erstellt mit den selektierten Messungen eine neue überlagerte Messung.
"Messung exportie-ren"	Exportiert eine Messung mit der zuletzt gespeicherten Standardansicht Die Messung wird mit der Dateierweiterung "*.ttrecx" oder "*.csv" gespeichert. Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.

Das Kopieren von Messungen ist unabhängig von der Gerätefamilie auch geräteübergreifend möglich.

Das Selektieren mehrerer Objekte ist möglich.

6.3.1.3 Oberfläche - Projektnavigation Ordner "Messungen im Gerät (Speicherkarte)"

Im Ordner werden auf Speicherkarte vorhandene Messungen dargestellt. Der Ordner wird nur bei einer Online-Verbindung zum Gerät angezeigt.

Ziehen Sie per Drag & Drop darin enthaltene Ordner oder Messungen in den Systemordner "Messungen". Damit werden die Messungen in das Projekt übernommen.

Symbole im Ordner "Traces"

Die folgende Tabelle erklärt die Symbole im Systemordner:

Symbol		Beschreibung
		Automatisch generierter Ordner mit Angabe des Aktivierungszeitpunkts der Aufzeichnung Der Name des Ordners kann nicht geändert werden.
		Messung im Gerät Mit Doppelklick auf das Symbol öffnen Sie das Register "Diagramm". Der Zeitstempel im Namen zeigt das Auftreten des Triggerereignisses an.

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle für den Ordner:

Kontextmenübe-fehl	Beschreibung
"Kopieren"	Kopiert die selektierten Objekte in die Zwischenablage.
"Löschen"	Löscht die selektierten Objekte aus der Projektnavigation und aus dem Gerät.

Die folgende Tabelle zeigt die Kontextmenübefehle für Messungen:

Kontextmenübe-fehl	Beschreibung
"Öffnen"	Öffnet die Messung im Register "Diagramm".
"Kopieren"	Kopiert die selektierten Objekte in die Zwischenablage.
"Löschen"	Löscht die selektierten Objekte aus der Projektnavigation und aus dem Gerät.

Kontextmenübe-fehl	Beschreibung
"Messung expor-tieren"	Exportiert eine Messung als Datei mit der Dateierweiterung "*.ttrecx" oder "*.csv". Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.
"Eigenschaften"	Zeigt die allgemeinen Eigenschaften der Messung (Seite 88) an.

Das Selektieren mehrerer Objekte ist möglich.

6.3.1.4 Oberfläche - Projektnavigation Ordner "Überlagerte Messungen"

Im Systemordner "Überlagerte Messungen" werden die konfigurierten überlagerten Messungen dargestellt.

Symbole im Ordner "Überlagerte Messungen"

Die folgende Tabelle erklärt die Symbole im Systemordner "Überlagerte Messungen":

Symbol	Beschreibung
	Neue überlagerte Messung hinzufügen Mit Doppelklick auf das Symbol fügen Sie eine neue überlagerte Messung hinzu und öffnen das Register "Diagramm".
	Überlagerte Messung Mit Doppelklick auf das Symbol öffnen Sie das Register "Diagramm".

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle für den Systemordner "Überlagerte Messungen" oder einen darin enthaltenen Gruppenordner:

Kontextmenübe-fehl	Beschreibung
"Neue Gruppe hin-zufügen"	Fügt einen neuen Ordner ein.
"Neue überlagerte Messung hinzufü-gen"	Fügt eine neue überlagerte Messung hinzu und öffnen das Register "Diagramm".
"Überlagerte Mes-sung importieren"	Importiert eine überlagerte Messung aus einer Datei mit der Dateierweiterung "*.ttcbmx"

Die folgende Tabelle zeigt die Kontextmenübefehle für überlagerte Messungen:

Kontextmenübefehl	Beschreibung
"Öffnen"	Öffnet die ausgewählten überlagerten Messungen im Arbeitsbereich.
"Messung importieren"	Importiert eine Messung aus einer Datei mit der Dateierweiterung "*.ttrecx" Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.

Kontextmenübefehl	Beschreibung
"Überlagerte Messung exportieren"	Exportiert eine überlagerte Messung Die überlagerte Messung wird mit der Dateierweiterung "*.ttcbmx" oder "*.csv" gespeichert. Das Format "*.ttcbmx" kann auch wieder importiert werden.
"Kopieren"	Kopiert die selektierten Objekte in die Zwischenablage.
"Einfügen"	Fügt Messungen, Messungen aus Traces im Gerät oder aus einer überlagerten Messung aus der Zwischenablage ein. Das Einfügen mehrerer Objekte aus der Zwischenablage ist möglich, wenn alle vom gleichen Typ sind.
"Löschen"	Löscht die selektierten Objekte aus der Projektnavigation bzw. aus dem Gerät.
"Umbenennen"	Schaltet das selektierte Objekt in den Editiermodus.
"Eigenschaften"	Zeigt die allgemeinen Eigenschaften der überlagerten Messung an.

Das Kopieren von überlagerten Messungen ist auch geräteübergreifend möglich.

Das Selektieren mehrerer Objekte ist möglich.

6.3.2 Arbeitsbereich

6.3.2.1 Oberfläche - Funktionsleiste des Trace

Über Schaltflächen stehen Werkzeuge zum Handling der Traces zur Verfügung.

Die folgende Tabelle zeigt die Funktionen der Schaltflächen:

Symbol	Beschreibung
	Übertragen der ausgewählten Tracekonfiguration in das Gerät Die ausgewählte Tracekonfiguration wird in das Gerät übertragen.
	Übertragen der ausgewählten Tracekonfiguration vom Gerät Die ausgewählte Tracekonfiguration wird vom Gerät in das Projekt übertragen.
	Beobachten ein-/ausschalten Wechsel der Anzeige zwischen Online und Offline. **Hinweis** Wenn das Beobachten und gleichzeitig die automatische Skalierung aktiviert sind, können keine Aktionen mehr mit der Schaltfläche "Rückgängig" rückgängig gemacht werden. **Hinweis** Beim erstmaligen Start eines Trace im Gerät wird die Anzeige im Kurvendiagramm auf automatische Skalierung voreingestellt. Beachten Sie bei einem wiederholten Start der Aufzeichnung, dass veränderte Einstellungen der Skalierung beibehalten werden. Aktivieren Sie zum Beobachten der Aufzeichnung ggf. die automatische Skalierung manuell wieder.

Symbol	Beschreibung
	Aktivieren der Aufzeichnung Bei Wiederholung der Aufzeichnung eines Trace im Gerät bleiben für die Anzeige relevante Einstellungen (Kurvendiagramm und Signaltabelle) auch für die neue Aufzeichnung erhalten. **Hinweis** Wenn eine Aufzeichnung neu gestartet wird, gehen die bisher aufgezeichneten Werte verloren. Um die aufgezeichneten Werte zu sichern, speichern Sie die Messung im Projekt (Seite 99), bevor Sie die Aufzeichnung erneut aktivieren.
	Deaktivieren der Aufzeichnung
	Trace im Gerät löschen Löscht den ausgewählten Trace aus dem Gerät.
	Aufzeichnung automatisch wiederholen Nachdem eine Aufzeichnung beendet ist wird die Aufzeichnung automatisch wieder aktiviert. Die Anzeige des Kurvendiagramms wird nach dem Beenden der Aufzeichnung aktualisiert. Diese Art der Aktualisierung eignet sich besonders zum Beobachten von schnellen Signalen (Seite 107).
	Übertragen der ausgewählten Messung aus dem Gerät in das Projekt Die Messung wird im Systemordner "Messungen" hinzugefügt. **Hinweis** Um die Daten des Trace im Gerät als Messung speichern zu können, muss diese zuvor einmalig im Kurvendiagramm angezeigt werden. Mit der Anzeige werden die Daten der Aufzeichnung aus dem Gerät geladen. Warten Sie ggf. die Aktualisierung der Anzeige ab, da immer nur die angezeigten Daten in das Projekt übertragen werden.
	Tracekonfiguration exportieren Exportiert eine Tracekonfiguration als Datei mit der Dateierweiterung "*.ttcfgx". Die Dateierweiterung "*.ttcfg" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.
	Tracekonfiguration erstellen Erstellt aus der Messung eine neue Tracekonfiguration.
	Messung exportieren mit den Einstellungen der aktuellen Ansicht Exportiert eine Messung als Datei mit der Dateierweiterung "*.ttrecx" oder "*.csv". Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.
	Messung importieren (nur bei überlagerten Messungen) Importiert eine Messung aus einer Datei mit der Dateierweiterung "*.ttrecx". Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.
	Überlagerte Messung exportieren (nur bei überlagerten Messungen) Die überlagerte Messung wird mit der Dateierweiterung "*.ttcbmx" oder "*.csv" gespeichert. Das Format "*.ttcbmx" kann auch wieder importiert werden.
	Auswahl einer Messung (nur bei überlagerten Messungen) Die Klappliste enthält die importierten Messungen. Wählen Sie zur Anzeige der Konfiguration die gewünschte Messung aus.

6.3.2.2 Oberfläche - Register Konfiguration

Oberfläche - Konfiguration

Die Tracekonfiguration ist geräteabhängig und beim jeweiligen Gerät (Seite 111) beschrieben.

6.3.2.3 Oberfläche - Register Diagramm

Oberfläche - Kurvendiagramm

Das Kurvendiagramm zeigt die ausgewählten Signale einer Aufzeichnung an. Analoge Signale werden im oberen Kurvendiagramm dargestellt. Binärsignale werden im unteren Diagramm als Bitspur dargestellt. Die Darstellung der Signale passen Sie in der Signaltabelle (Seite 76) und mit der Funktionsleiste des Kurvendiagramms an.

Einstellmöglichkeiten und Anzeigen im Kurvendiagramm

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

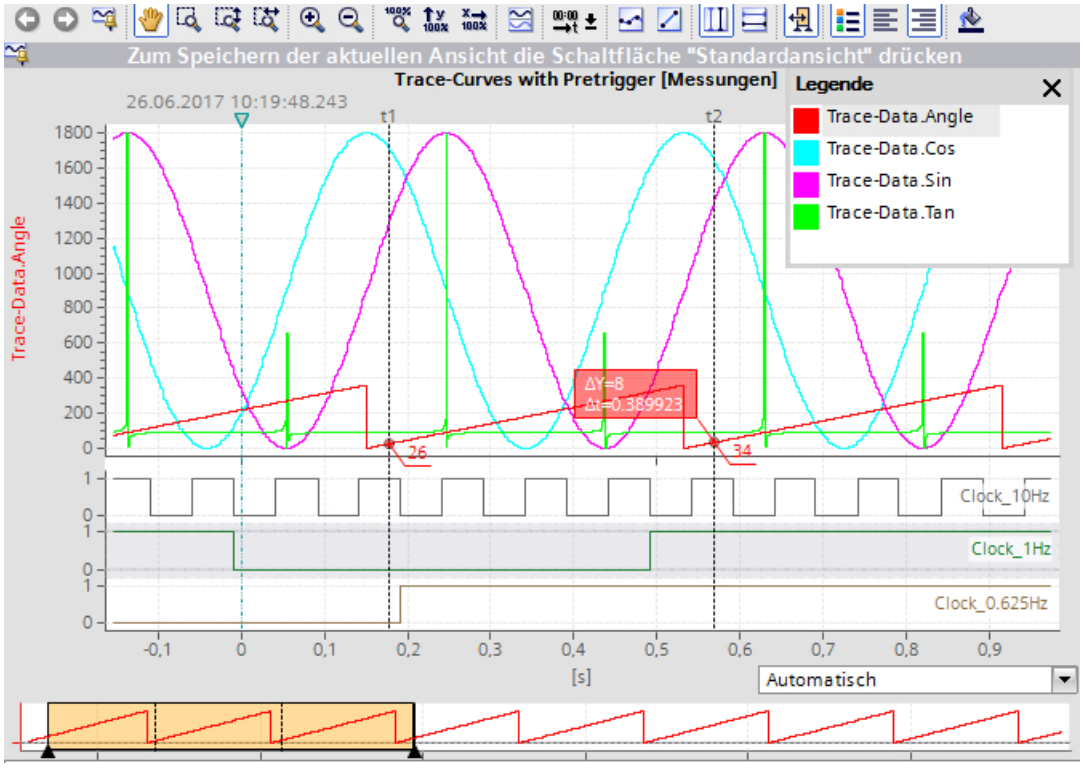

Die Skala im Diagramm gilt für das angewählte (grau hinterlegte) Signal in der Legende. Die Legende kann mit der Maus verschoben und in der Größe angepasst werden.

Das Symbol ▽ zeigt mit einer vertikalen Linie den Triggerzeitpunkt mit der Triggerzeit vom Gerät an.

Bei der Einstellung "Zeit" für die Zeitachse steht unterhalb des Kurvendiagramms eine Klappliste zur Auswahl der Einheit zur Verfügung. Die Einstellung "Automatisch" passt die Einheit entsprechend dem dargestellten Zeitbereich automatisch an.

Hinweis

Nicht interpretierbare Datentypen

Manche Datentypen benötigen ein definiertes Format, z. B. der S7-Datentyp `LTime_of_Day`. Liegt dieses Format nicht vor, dann wird der Datentyp als INT interpretiert.

Funktionen mit dem Mausrad

Die folgende Tabelle zeigt, welche Funktionen mit dem Mausrad im Kurvendiagramm möglich sind:

Funktion des Mausrads	Beschreibung
Verschieben des Kurvendiagramms in vertikaler Richtung	Drehen des Mausrads verschiebt die Anzeige im oberen Kurvendiagramm nach oben oder nach unten. Wenn die Signale in Spuren angeordnet sind, wird die Anzeige der Gruppe verschoben, die sich unterhalb des Cursors befindet. Der Mauszeiger muss über dem Kurvendiagramm mit den analogen Signalen positioniert sein.
Verschieben des Kurvendiagramms in horizontaler Richtung	Drehen des Mausrads bei gedrückter Taste <Shift> verschiebt die Anzeige im Kurvendiagramm nach links oder nach rechts. Der Mauszeiger muss über dem Kurvendiagramm positioniert sein.
Zoom in und Zoom out	Drehen des Mausrads bei gedrückter Taste <Strg> vergrößert oder verkleinert die Anzeige im Kurvendiagramm. Ausgangspunkt der Vergrößerung bzw. Verkleinerung ist die Position des Mauszeigers. Die Werteachse des unteren Kurvendiagramms (Bitspuren) wird nicht beeinflusst. Der Mauszeiger muss über dem Kurvendiagramm positioniert sein.

Funktionen mit der Tastatur

Die folgende Tabelle zeigt, welche Tastaturbefehle bei Fokus auf dem Kurvendiagramm möglich:

Tastenkombination	Beschreibung
Selektieren eines Mess-Cursors	
<Strg+Shift+1>	Der vertikale Mess-Cursor t1 wird selektiert bzw. deselektiert.
<Strg+Shift+2>	Der vertikale Mess-Cursor t2 wird selektiert bzw. deselektiert.
<Strg+Shift+3>	Der horizontale Mess-Cursor Y1 wird selektiert bzw. deselektiert.
<Strg+Shift+4>	Der horizontale Mess-Cursor Y2 wird selektiert bzw. deselektiert.
<Tab>	Der nächste Mess-Cursor wird selektiert.
Positionieren eines vertikalen Mess-Cursors	

Tastenkombination	Beschreibung
<Links>, <Rechts>	Der selektierte Mess-Cursor wird bei der Einheit "Messpunkte" um einen Messpunkt von dem im Vordergrund befindlichen Signal verschoben. Bei der Einheit "Zeit" erfolgt die Verschiebung um ein Pixel.
<Shift+Links>, <Shift+Rechts>	Der selektierte Mess-Cursor wird bei der Einheit "Messpunkte" um 10 Messpunkte von dem im Vordergrund befindlichen Signal verschoben. Bei der Einheit "Zeit" erfolgt die Verschiebung um 10 Pixel.
Positionieren eines horizontalen Mess-Cursors	
<Auf>, <Ab>	Der selektierte Mess-Cursor wird um ein Pixel entlang der Werteachse verschoben.
<Shift+Auf>, <Shift+Ab>	Der selektierte Mess-Cursor wird um 10 Pixel entlang der Werteachse verschoben.
Anzeigen der vertikalen Mess-Cursor	
<Strg+Leertaste>	Die vertikalen Mess-Cursor werden ein- bzw. ausgeblendet.
<Strg+Shift+Leertaste>	Die vertikalen Mess-Cursor werden ggf. eingeblendet und für die aktuelle Ansicht zentriert.
Ändern der Ansicht	
<Leertaste>	Ansicht verschieben
<Strg+0>	100% Ansicht im geöffneten Editor einstellen
<Strg++>	Zoom in mit 10% anwenden
<Strg+->	Zoom out mit 10% anwenden

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle im Kurvendiagramm:

Kontextmenübefehl	Beschreibung
"Diagramm als Bild speichern unter"	Exportiert die aktuelle Anzeige in einem Grafikformat, z. B. als Bitmap.
"Bild in Zwischenablage kopieren"	Kopiert die aktuelle Anzeige in die Zwischenablage.
"Mess-Cursor zentrieren"	Positioniert die aktivierten Mess-Cursor an einer zentralen Stelle der aktuellen Anzeige.
"Zu Messungen hinzufügen" (nur Achssteuertafel und PID)	Fügt die dargestellte Aufzeichnung im Systemordner "Messungen" hinzu.
"Automatische Bitspurenhöhe"	Passt die Höhe der Bitspuren automatisch an und legt damit die Größe des unteren Kurvendiagramms fest. Die Einstellung wird automatisch deaktiviert, sobald Sie die Platzaufteilung zwischen den Kurvendiagrammen manuell verändern. **Hinweis** Sie können die vertikale Platzaufteilung zwischen dem oberen und unteren Kurvendiagramm verändern. Ziehen Sie dazu mit der Maus die Zeitachse des oberen Kurvendiagramms nach oben oder nach unten.

Funktionsleiste des Kurvendiagramms

Über Schaltflächen stehen Werkzeuge zur Anpassung der Darstellung zur Verfügung.

Die folgende Tabelle zeigt die Funktionen der Schaltflächen:

Symbol	Funktion	Beschreibung
	Zoom rückgängig	Rückgängig machen der zuletzt ausgeführten Zoomfunktion. Wenn mehrere Zoomfunktionen ausgeführt wurden, können diese schrittweise rückgängig gemacht werden.
	Zoom wiederholen	Wiederholen der zuletzt rückgängig gemachten Zoomfunktion. Wenn mehrere Zoomfunktionen rückgängig gemacht wurden, können diese schrittweise wiederholt werden.
	Standardansicht	Aktuelle Ansicht als Standard für diese Aufzeichnung verwenden. Wenn die Traceaufzeichnung später erneut angezeigt wird, dann wird die Standardansicht wieder hergestellt.
	Ansicht verschieben	Verschieben der Anzeige mit gedrückter Maustaste.
	Zoomauswahl	Auswahl eines beliebigen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.
	Zoomauswahl vertikal	Auswahl eines vertikalen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.
	Zoomauswahl horizontal	Auswahl eines horizontalen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.
	Zoom in	Vergrößern der Anzeige. Die Bereiche der Zeitachse und Werteachse werden mit jedem Klick auf die Schaltfläche verkleinert. Die Kurven werden größer dargestellt.
	Zoom out	Verkleinern der Anzeige. Die Bereiche der Zeitachse und Werteachse werden mit jedem Klick auf die Schaltfläche vergrößert. Die Kurven werden kleiner dargestellt.
	Alles anzeigen	Skaliert die Anzeige der vorhandenen Daten so, dass der gesamte Zeitbereich und alle Werte angezeigt werden.
	Automatische Skalierung der Werteachse	Skalierung der Anzeige, dass für den aktuell dargestellten Zeitbereich alle Werte angezeigt werden. Das relative Skalierungsverhältnis zwischen den Signalen kann sich ändern. **Hinweis** Die automatische Skalierung der Werteachse wird bei Aktivierung einer Zoomfunktion der Werteachse angehalten. Mit dieser Schaltfläche aktivieren Sie die automatischen Anpassungen an die minimalen/maximalen Werte wieder.

Symbol	Funktion	Beschreibung
	Gesamten Zeitbereich anzeigen	Skalierung der Anzeige, dass für den gesamten Zeitbereich die Werte im aktuell dargestellten Wertebereich angezeigt werden. Der angezeigte Wertebereich ändert sich nur dann, wenn auch "Alle Werte anzeigen" aktiviert ist. **Hinweis** Die automatische Skalierung der Zeitachse wird bei Aktivierung einer Zoomfunktion der Zeitachse angehalten. Mit dieser Schaltfläche aktivieren Sie die automatischen Anpassungen des Zeitbereichs wieder.
	In Spuren anordnen	Aktivierung bzw. Deaktivierung der Spuranordnung. Bei aktivierter Spuranordnung werden die Signale untereinander mit den jeweiligen Werteachsen angeordnet. Signalgruppen werden in der gleichen Spur angezeigt. Diese Einstellung hat keine Auswirkung auf die Anzeige der Bitspuren.
	Einheitenumschaltung der Zeitachse	Umschalten der Einheit der Zeitachse Folgende Einheiten sind einstellbar: • Messpunkte • Zeit (Relative Zeit bezogen auf den Triggerzeitpunkt) • Zeitstempel der Messpunkte
	Messpunkte anzeigen	Die Messpunkte werden als kleine Kreise auf den Kurven angezeigt.
	Interpolierte Darstellung	Lineare Interpolation zwischen zwei aufeinanderfolgenden Messpunkten bei Gleitpunktzahlen Bei nicht aktivierter linearer Interpolation (Standard) wird die Verbindung zwischen den Messpunkten in Stufen gezeichnet.
	Vertikale Mess-Cursor anzeigen	Anzeige der vertikalen Mess-Cursor. Die vertikale Position der beiden Mess-Cursor kann mit der Maus verschoben werden. In der Signaltabelle werden entsprechend der Position die zugehörigen Messwerte und die Differenz der Mess-Cursor angezeigt. Blenden Sie in der Task Card Trace die Palette "Mess-Cursor" (Seite 85) ein, um weitere Informationen anzuzeigen. Nutzen Sie auch die Cursortasten. Folgende Aktionen sind bei den vertikalen Mess-Cursoren mit den Cursortasten möglich: • Selektieren • Positionieren • Mess-Cursor ein- oder ausblenden • Mess-Cursor zentrieren

Symbol	Funktion	Beschreibung
	Horizontale Mess-Cursor anzeigen	Anzeige der horizontalen Mess-Cursor. Die horizontale Position der beiden Mess-Cursor kann mit der Maus verschoben werden. Blenden Sie in der Task Card Trace die Palette "Mess-Cursor" (Seite 85) ein, um die Werte anzuzeigen oder die Mess-Cursor über Positionseingabe neu zu positionieren. Nutzen Sie auch die Cursortasten. Folgende Aktionen sind bei den horizontalen Mess-Cursoren mit den Cursortasten möglich: • Selektieren • Positionieren
	Zeitbereichsanzeige	Ein- bzw. Ausblenden der Zeitbereichsanzeige unterhalb des Kurvendiagramms Die Zeitbereichsanzeige stellt anhand eines selektierten Signals den Anzeigebereich im Kurvendiagramm als gelbe Fläche dar. Die gelbe Fläche kann mit der Maus verschoben und an den Begrenzungen horizontal in der Größe verändert werden. 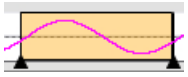
	Legende anzeigen	Ein- bzw. Ausblenden der Legende im Kurvendiagramm und der Bitspur-Bezeichnungen.
	Legende linksbündig anzeigen	Anzeige der Legende und der Bitspur-Bezeichnungen auf der linken Seite des Kurvendiagramms.
	Legende rechtsbündig anzeigen	Anzeige der Legende und der Bitspur-Bezeichnungen auf der rechten Seite des Kurvendiagramms.
	Hintergrundfarbe wechseln	Umschalten zwischen verschiedenen Hintergrundfarben.

Oberfläche - Signaltabelle

Die Signaltabelle listet die Signale der ausgewählten Messung auf und bietet Einstellmöglichkeiten für einige Eigenschaften. Wenn Aufzeichnungsdaten von Traces im Gerät angezeigt werden und die Einstellungen in der Signaltabelle verändert werden, bleiben diese Einstellungen nur so lange erhalten, bis in den Offline-Modus gewechselt wird. Beim Hinzufügen des Trace im Gerät zu den Messungen werden die aktuellen Einstellungen der Signaltabelle in der Messung gespeichert.

Die Sortierung der Signale ist per Drag & Drop möglich. Die Bits eines Signals können innerhalb eines Signals umsortiert werden.

Einstellmöglichkeiten und Anzeigen in der Signaltabelle

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

		Si...	Name	Datentyp	Anzeigeformat	Adresse	Formel	Farbe	Signalgruppe	Min. Y-Skala	Max. Y-Skala	Y(t1)	Y(t2)	ΔY	Einheit	Kommentar
1		$0	"Trace-Data".Angle	Int	Dez +/-				Temp	0	1800	78	67	-11		
2		$1	"Trace-Data".Cos	Real	Gleitpunkt				Temp	0	1800	20....	39....	18....		
3		$2	"Trace-Data".Sin	Real	Gleitpunkt					-99.99971	99.99997	97....	92....	-5....		

Die folgende Tabelle zeigt die Einstellungen und Anzeigen der aufgezeichneten Signale:

Spalte	Beschreibung
Signal- oder Fehler-Symbol	
	Signal
	Signal Failsafe
	Signal aus einem Datenbaustein
	Signal aus einem Datenbaustein Failsafe
	Berechnetes Signal (Formel)
	Fehler in der Formel des berechneten Signals
	Auswahl für die Anzeige im Kurvendiagramm - Maximal können 16 Signale ausgewählt werden. Der Punkt zeigt an, dass für das Signal in der Bitauswahl mindestens ein Bit zur Anzeige als Bitspur ausgewählt ist.
"Signalreferenz"	Automatisch generierte Nummer des Signals Über die Signalreferenz kann in den Formeln auf das Signal zugegriffen werden.
"Name"	Anzeige des Signalnamen Ein Klick auf den Namen eines angezeigten Signals aktualisiert die Skala im Kurvendiagramm. In der letzten Zeile ohne Signal-Symbol ist die Eingabe eines Namen für ein berechnetes Signal möglich. Mit Eingabe des Namen wird das berechnete Signal angelegt.
"Messung"	Anzeige der Messung (nur im Register "Diagramm" bei überlagerten Messungen) Zeigt den Namen der Messung an, zu der das Signal gehört.
	Bitauswahl öffnen Für folgende Datentypen können auch einzelne Bits zur Anzeige als Bitspur im unteren Kurvendiagramm ausgewählt werden: • Byte, Word, DWord, LWord • SInt, USInt, Int, UInt, DInt, UDInt, LInt, ULInt Beispiel einer geöffneten Bitauswahl für den Datentyp DWord: $0 "Tag_1" Byte Dez %MB10 $1 "Tag_2" DWord Dez %MD20 $1.x0 Bool Bin %M23.0 $1.x1 Bool Bin %M23.1 $1.x2 Bool Bin %M23.2 Wählen Sie durch Klick auf das Symbol das jeweilige Bit für die Anzeige ab oder an.
"Datentyp"	Anzeige des Datentyps

Spalte		Beschreibung
"Anzeigeformat"		Anzeigeformats des Signals Die für das Signal unterstützten Anzeigeformate werden zur Auswahl angeboten. Bei "Voreinstellung" wird ein für den Datentyp passendes Anzeigeformat eingestellt.
"Adresse"		Anzeige der Adresse des Signals Bei optimierten / typsicheren Variablen bleibt das Feld leer.
"Formel"		Anzeige bzw. Eingabe einer Formel Eine Formel kann mathematische Funktionen mit Zahlen und Signalen enthalten. Nutzen Sie den Formeleditor zum komfortablen Erstellen von Formeln.
		Aufruf des Formeleditors bei berechneten Signalen Klicken Sie auf das Symbol, um den Formeleditor aufzurufen.
"Farbe"		Anzeige und Einstellmöglichkeit für die Farbe des Signals
"Signalgruppe"		Anzeige bzw. Eingabe des Signalgruppennamens für eine Signalgruppe Bei allen Signalen einer Signalgruppe werden die Y-Skalen gleich skaliert. Geben Sie bei den Signalen einen identischen Signalgruppennamen ein, die gleich skaliert werden sollen. Signale können durch Löschen des Signalgruppennamens aus der Signalgruppe entfernt werden. Die Signalgruppen können über die Funktion "Aktuelle Ansicht als Standard verwenden" (Schaltfläche) gespeichert werden. **Hinweise** Binärsignale können nicht gruppiert werden. Gruppieren Sie beim Anzeigeformat Hex nur Signale, deren Format bezüglich des Vorzeichens für die Anzeige kompatibel ist.
	Graues Feld für Kettensymbol	Bewegen Sie den Mauszeiger über das graue Feld bzw. das Kettensymbol (oder), um das Signal einer Signalgruppe hinzuzufügen bzw. aus der Signalgruppe zu löschen. Ein Klick auf das Kettensymbol fügt das Signal einer Signalgruppe hinzu bzw. erstellt eine neue Signalgruppe. Ein Klick auf das Kettensymbol entfernt das Signal aus der Signalgruppe. Für ein markiertes Signal mit Signalgruppe zeigt das Kettensymbol alle Signale der gleichen Signalgruppe an.
	Eingabefeld	Das Eingabefeld zeigt den Signalgruppennamen an. Alternativ zum Kettensymbol können Sie in diesem Feld durch Texteingabe einen Gruppennamen zuordnen oder löschen.
"Min. Y-Skala"		Anzeige bzw. Eingabe des minimalen Werts für die Skalierung des Signals
"Max. Y-Skala"		Anzeige bzw. Eingabe des maximalen Werts für die Skalierung des Signals
"Y(t1)"		Anzeige des Werts an der Position des ersten Mess-Cursors
"Y(t2)"		Anzeige des Werts an der Position des zweiten Mess-Cursors
"ΔY"		Anzeige der Wertdifferenz zwischen dem ersten und dem zweiten Mess-Cursor
"Einheit"		Anzeige der Einheit (z. B. bei einheitenbehafteten Werten aus Technologieobjekten)
"Kommentar"		Anzeige und Eingabemöglichkeit eines Kommentars zu dem Signal

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Signaltabelle:

Kontextmenübefehl	Beschreibung
"Neues·berechnetes·Signal·einfügen"	Fügt oberhalb ein neues berechnetes Signal in die Tabelle ein
"Formel bearbeiten"	Öffnet den Formeleditor für das berechnete Signal
"Ausschneiden"	Nicht anwählbar
"Kopieren"	Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.
"Einfügen"	Nicht anwählbar
"Löschen"	Nicht anwählbar
"Umbenennen"	Nicht anwählbar
"Anzeigeformat"	Ermöglicht die Umschaltung des Anzeigeformats Die für das Signal unterstützten Anzeigeformate werden zur Auswahl angeboten.
"Signal(e) einblenden"	Blendet die selektierten Signale im Kurvendiagramm ein.
"Signal(e) ausblenden"	Blendet die selektierten Signale im Kurvendiagramm aus.

Siehe auch

Verwendung der Signaltabelle (Seite 104)

Verwendung der Signalgruppe in der Signaltabelle (Seite 105)

Oberfläche - Formeleditor

Der Formeleditor bietet für die Analyse von Signalen verschiedene mathematische Funktionen an. Öffnen Sie den Editor in der Signaltabelle über die Schaltfläche .

Parametriermöglichkeiten und Anzeigen im Formeleditor

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

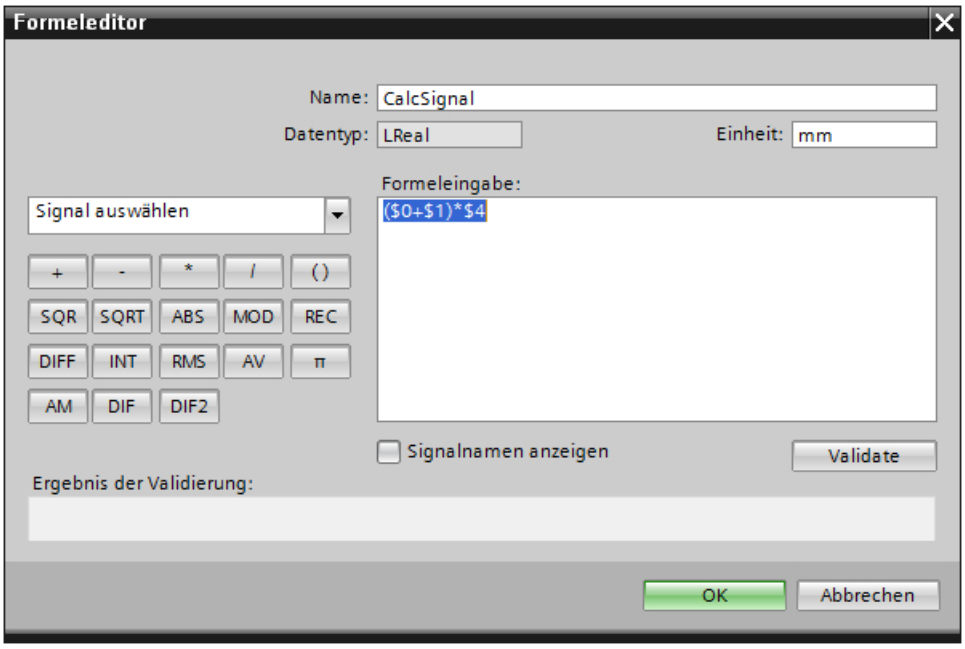

Bild 6-1 Formeleditor

Die folgende Tabelle zeigt die Parametriermöglichkeiten und Anzeigen des Formeleditors:

Feld/Schaltfläche		Beschreibung
"Name"		Anzeige und Eingabe der Bezeichnung für die erstellte Formel Der Name muss eindeutig sein und darf nur Zeichen enthalten, die auch in Windows-Dateinamen zulässig sind.
"Datentyp"		Anzeige des Datentyps der Formel Der Datentyp ist mit einer Gleitpunktzahl vom Typ LREAL vorbelegt und nicht änderbar.
"Einheit"		Anzeige und Eingabe einer Einheit Frei eingebbare anwenderdefinierte Einheit.
Klappliste mit Signalen		Auswahl der Signale Die Klappliste enthält die Signale der Signaltabelle und fügt ein angewähltes Signal in die Formel ein.
"Formeleingabe"		Textfeld zur Anzeige und Eingabe der Formel Erstellen Sie eine Formel durch Eingabe in dieses Textfeld oder mit Hilfe der Schaltflächen für die mathematischen Funktionen. Im Textfeld können Signale über die Signalreferenz mit vorangestelltem $-Zeichen oder über den Namen in doppeltem Hochkomma in der Formel referenziert werden. Eine gemischte Eingabe ist möglich. Bits aus einer Bitauswahl (z. B. unterhalb des Datentyps INT) sind in der Formel nicht zulässig.
Mathematische Funktionen		
	+	Addition

Feld/Schaltfläche		Beschreibung
	-	Subtraktion
	*	Multiplikation
	/	Division
	()	Klammern Gruppieren von Ausdrücken
	SQR	Quadrat
	SQRT	Wurzel
	ABS	Absolutwert Berechnet den Betrag einer Zahl. **Beispiele** `ABS(5)` → 5 `ABS(-3)` → 3 `ABS(-3.14)` → 3.14
	MOD	Modulo Berechnet den Restwert einer Division **Beispiele** `MOD(5,3)` → 2 `MOD(3.14,3)` → 0.14
	REC	Reziprokwert (1/x)
	DIFF [1)]	Numerische Differenziation **Beispiele** Formel: `DIFF($0,SAMPLETIME)`
	INT [1)]	Numerische Integration **Beispiele** Formel: `INT($0,SAMPLETIME)`
	RMS [1)]	Quadratisches Mittel Das quadratische Mittel wird gebildet, indem die Quadrate alle Messwerte addiert und durch die Anzahl Messwerte dividiert werden. Daraus die Quadratwurzel ergibt das quadratische Mittel. **Beispiele** Formel: `RMS($0,SAMPLETIME)`
	AV	Mittelwertfilter 1. bis 5. Ordnung Fehlt die Angabe einer Ordnung, dann wird ein Mittelwertfilter 1. Ordnung ausgeführt. **Beispiele** `AV($0,1)` → Mittelwertfilter 1. Ordnung `AV($0,5)` → Mittelwertfilter 5. Ordnung
	π	Mathematische Konstante PI
	AM	Arithmetisches Mittel Das arithmetische Mittel ist ein gleitender Mittelwert über fünf Messpunkte.

Feld/Schaltfläche		Beschreibung
	DIF	Einfache Differenzbildung mit Mittelwertfilter 1. bis 5. Ordnung Fehlt die Angabe einer Ordnung, dann wird eine einfache Differenzbildung mit Filter 1. Ordnung ausgeführt. **Beispiele** `DIF($0,1)` → Einfache Differenzbildung mit Filter 1. Ordnung `DIF($0,5)` → Einfache Differenzbildung mit Filter 5. Ordnung `DIF($0)` → Einfache Differenzbildung mit Filter 1. Ordnung **Beispiel: Aus einem Geschwindigkeitssignal einen Beschleunigungsverlauf berechnen** `$0`: Geschwindigkeitssignal in Meter pro Sekunde Zykluszeit der äquidistanten Geschwindigkeitsaufzeichnung: 1 ms Formel: `DIF($0,1)/0.001` Einheit: m/s^2
	DIF2	Doppelte·Differenzbildung·mit·Mittelwertfilter·1.·bis·5. Ordnung Fehlt die Angabe einer Ordnung, dann wird eine doppelte Differenzbildung mit Filter 1. Ordnung ausgeführt. **Beispiele** `DIF2($0,1)` → Doppelte Differenzbildung mit Filter 1. Ordnung `DIF2($0,5)` → Doppelte Differenzbildung mit Filter 5. Ordnung `DIF2($0)` → Doppelte Differenzbildung mit Filter 1. Ordnung **Beispiel: Aus einem Positionsverlauf einen Beschleunigungsverlauf berechnen** `$0`: Positionsverlauf in Meter Zykluszeit der äquidistanten Positionsaufzeichnung: 1 ms Formel: `DIF2($0,1)/SQR(0.001)` Einheit: m/s^2
"Signalnamen anzeigen"		Anzeige der Signalnamen Bei aktiviertem Optionskästchen werden in der Formel statt den Signalreferenzen die Signalnamen angezeigt.
"Validieren"		Prüfen der Formel auf Gültigkeit
"Ergebnis der Validierung"		Ergebnis der Validierung Zeigt das Ergebnis der Validierung an und weist auf einen Fehler und die Fehlerstelle hin.
"OK"		Übernehmen der Eingaben im Formeleditor
"Abbrechen"		Verwerfen der Eingaben im Formeleditor

1) Die Konstante `SAMPLETIME` steht nur bei äquidistanten Aufzeichnungstakten zur Verfügung. Zeiteinheit für `SAMPLETIME` ist immer µs.

Oberfläche - Messungen (Überlagerte Messungen)

Das Register Messungen zeigt die einzelnen Messungen an und bietet u. a. Einstellmöglichkeiten für die Synchronisation.

Einstellmöglichkeiten und Anzeigen im Register Messungen

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

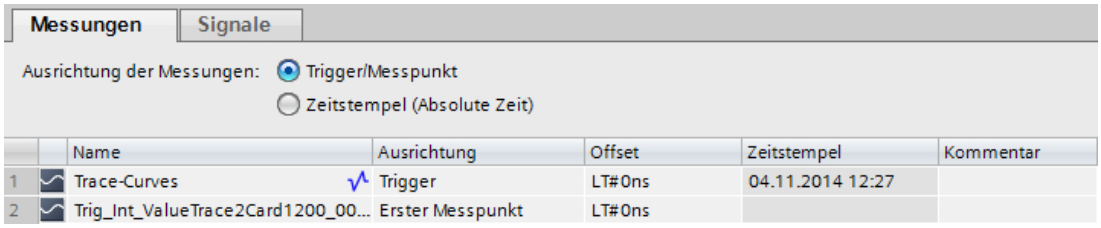

Die folgende Tabelle zeigt die Einstellungen und Anzeigen zu den Messungen:

Spalte		Beschreibung
Ausrichtung der Messungen		
	"Trigger/ Messpunkt"	Ausrichtung der Messungen nach dem Trigger bzw. Messpunkt Der individuelle Nullpunkt der Messung wird in der Tabelle unter der Spalte "Ausrichtung" festgelegt.
	"Zeitstempel (Absolute Zeit)"	Ausrichtung der Messungen nach ihrem Zeitstempel Die Signale werden nach der Zeit aus dem absoluten Zeitstempel ausgerichtet.
Tabellenspalten		
		Statische Anzeige des Messungs-Symbols
	"Name"	Anzeige und Änderungsmöglichkeit des Messungsnamen Der Name muss eindeutig sein und kann geändert werden.
	"Ausrichtung"	Ausrichtung der Messung (Nur bei aktiviertem Optionskästchen "Trigger/Messpunkt" einstellbar) Legt den individuellen Nullpunkt einer Messung fest. Alle Signale der Messung werden auf diesen Nullpunkt bezogen angezeigt. Folgende Einstellungen sind möglich: • Trigger • Erster Messpunkt nach dem Triggerereignis • Erster Messpunkt • Letzter Messpunkt
	"Offset"	Offset bezogen auf die Zeitachse Verschiebt die Messung um den angegebenen Offset auf der Zeitachse nach links oder rechts. Der Offset kann auch über die Zwischenablage aus dem Wert ΔX der Mess-Cursor in die Zelle übertragen werden. Siehe Messungen exakt ausrichten (überlagerte Messungen) (Seite 109).
	"Zeitstempel"	Anzeige des Triggerzeitpunkts
	"Kommentar"	Anzeige und Eingabemöglichkeit eines Kommentars zu dem Signal

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Signaltabelle:

Kontextmenübefehl	Beschreibung
"Ausschneiden"	Nicht anwählbar
"Kopieren"	Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.
"Einfügen"	Nicht anwählbar
"Löschen"	Nicht anwählbar
"Umbenennen"	Schaltet die selektierte Zelle in den Editier-Modus.
"Messung importieren"	Importiert eine Messung aus einer Datei z. B. mit der Dateierweiterung "*.ttrecx". Der Import ist geräteunabhängig und eignet sich z. B. auch zum Vergleich von Messungen einer PLC mit den Messungen eines Antriebsgeräts. Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.
"Messung exportieren"	Exportiert eine Messung als Datei mit der Dateierweiterung "*.ttrecx" oder "*.csv". Die Dateierweiterung "*.ttrec" wird aus Kompatibilitätsgründen zu V12 unterstützt, enthält jedoch keine Informationen über die Gerätefamilie.

6.3.2.4 Oberfläche - Register Signalauswahl (Überlagerte Messungen)

Oberfläche - Signalauswahl (Überlagerte Messungen)

Das Register Signalauswahl zeigt die Signale aller Messungen an und ermöglicht eine Vorauswahl von Signalen, die in der Signaltabelle des Diagramms dargestellt werden.

Einstellmöglichkeiten und Anzeigen im Register Signalauswahl

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

		Verfü...	Messung	Name	Datentyp	Adresse	Kommentar
1		☑	Trace-Curves	"Trace-Data".Angle	Int		
2		☑	Trace-Curves	"Clock_1Hz"	Bool	%M100.5	
3		☐	Trace-Curves	"Clock_1.25Hz"	Bool	%M100.4	
4		☐	Trace-Curves	"Clock_2Hz"	Bool	%M100.3	
5		☑	Trace-Curves	"Clock_2.5Hz"	Bool	%M100.2	
6		☑	Trace-Curves	"Clock_5Hz"	Bool	%M100.1	

Die folgende Tabelle zeigt die Einstellungen und Anzeigen der Tabelle:

Spalte	Beschreibung
	Statische Anzeige des Signal-Symbols
"Verfügbar im Diagramm"	Auswahl für die Anzeige im Kurvendiagramm Bei aktivierter Auswahl wird das Signal in die Signaltabelle des Kurvendiagramms übernommen.
"Messung"	Anzeige der Messung, zu der das Signal gehört
"Name"	Anzeige des Signalnamen
"Datentyp"	Anzeige des Datentyps
"Adresse"	Anzeige der Adresse (nicht bei symbolischen Variablen)
"Kommentar"	Anzeige eines Kommentars zu dem Signal

Weitere Informationen zu den einzelnen Einstellungen finden Sie in Oberfläche - Signaltabelle (Seite 76).

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Signalauswahl:

Kontextmenübefehl	Beschreibung
"Kopieren"	Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.
"Auswahl in der Signaltabelle anzeigen"	Die ausgewählten Signale werden in der Signaltabelle angezeigt und sind im Diagramm verfügbar.
"Auswahl aus der Signaltabelle entfernen"	Die entfernten Signale sind nicht im Diagramm verfügbar.

Das Selektieren mehrerer Objekte ist möglich.

6.3.3 Task Card Trace

6.3.3.1 Oberfläche - Palette Mess-Cursor

Die Palette "Mess-Cursor" zeigt die Position der Mess-Cursor im Kurvendiagramm an und die Werte an den Schnittpunkten.

Einstellmöglichkeiten und Anzeigen der Palette "Mess-Cursor"

Das folgende Bild zeigt die Palette "Mess-Cursor":

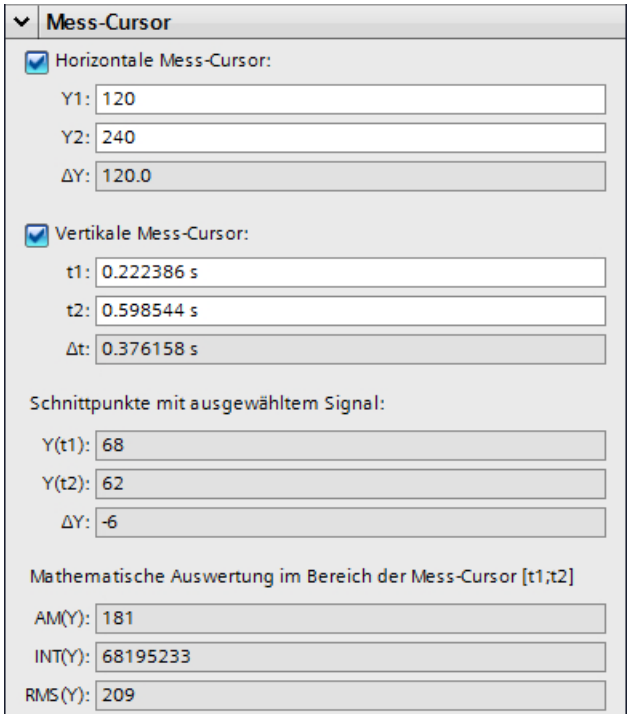

Die folgende Tabelle beschreibt die Einstellungen und Anzeigen:

Einstellung/Anzeige		Beschreibung
Horizontale Mess-Cursor		
	Y1	Position des ersten Mess-Cursors Der Wert gibt die Position bezogen auf die Skala des aktuell angewählten Signals an. Optional zum Verschieben mit der Maus können Sie in diesem Eingabefeld eine neue Position des Mess-Cursors vorgeben.
	Y2	Position des zweiten Mess-Cursors Der Wert gibt die Position bezogen auf die Skala des aktuell angewählten Signals an. Optional zum Verschieben mit der Maus können Sie in diesem Eingabefeld eine neue Position des Mess-Cursors vorgeben.
	ΔY	Anzeige der Positionsdifferenz zwischen dem ersten und dem zweiten Mess-Cursor
Vertikale Mess-Cursor		

Einstellung/Anzeige		Beschreibung
	t1	Position des ersten Mess-Cursors Optional zum Verschieben mit der Maus können Sie in diesem Eingabefeld eine neue Position des Mess-Cursors vorgeben.
	t2	Position des zweiten Mess-Cursors Optional zum Verschieben mit der Maus können Sie in diesem Eingabefeld eine neue Position des Mess-Cursors vorgeben.
	Δt	Anzeige der Positionsdifferenz zwischen dem ersten und dem zweiten Mess-Cursor
Schnittpunkte mit ausgewähltem Signal		
	Y(t1)	Anzeige des Werts an der Position des ersten Mess-Cursors
	Y(t2)	Anzeige des Werts an der Position des zweiten Mess-Cursors
	ΔY	Anzeige der Wertdifferenz zwischen dem ersten und dem zweiten Mess-Cursor
Mathematische·Auswertung·im·Bereich·der·Mess-Cursor·[t1;t2] für das ausgewählte Signal		
	AM(Y)	Mittelwert Der arithmetische Mittelwert wird für den Bereich zwischen den vertikalen Mess-Cursoren berechnet.
	INT(Y)	Integral Das Integral wird für den Bereich zwischen den vertikalen Mess-Cursoren berechnet.
	RMS(Y)	Effektivwert Der Effektivwert wird für den Bereich zwischen den vertikalen Mess-Cursoren berechnet.

Siehe auch

Oberfläche - Kurvendiagramm (Seite 71)

6.3.3.2 Oberfläche - Palette Momentaufnahmen

Die Palette "Momentaufnahmen" ermöglicht das Speichern und Wiederherstellen von verschiedenen Ansichten einer Messung.

Von einer gerade aktuellen Ansicht im Register "Diagramm" wird eine Momentaufnahme erstellt. Die Momentaufnahmen werden in der Messung mit dem Projekt gespeichert.

Einstellmöglichkeiten und Anzeigen der Palette "Momentaufnahmen"

Das folgende Bild zeigt die Palette "Momentaufnahmen":

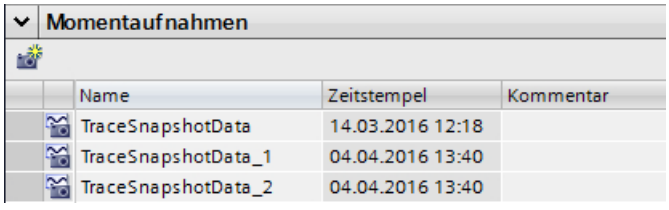

Die folgende Tabelle zeigt die Funktionen der Schaltflächen:

Symbol	Beschreibung
	Momentaufnahme der aktuellen Ansicht erstellen Speichert die aktuelle Ansicht in Register "Diagramm" als Momentaufnahme.

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

Spalte	Beschreibung
	Statische Anzeige des Symbols der Momentaufnahme
"Name"	Anzeige und Änderungsmöglichkeit des Namen
"Zeitstempel"	Anzeige der Erstellungszeit der Momentaufnahme
"Kommentar"	Anzeige und Eingabemöglichkeit eines Kommentars

Das Selektieren und Löschen mehrerer Zeilen ist möglich.

Mit Doppelklick auf eine Zeile wird die Messung mit der gespeicherten Ansicht im Register "Diagramm" geöffnet.

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle:

Kontextmenübefehl	Beschreibung
"Momentaufnahme wiederherstellen"	Zeigt im Register "Diagramm" die Messung mit der gespeicherten Ansicht an.
"Löschen"	Löscht die Momentaufnahme
"Umbenennen"	Schaltet die Zelle in den Editiermodus

Das Selektieren und Löschen mehrerer Zeilen ist möglich.

6.3.4 Inspektorfenster

6.3.4.1 Oberfläche - Inspektorfenster

Das Inspektorfenster zeigt allgemeine Informationen zum Trace an.

Für Messungen stehen weitere Informationen zur Verfügung:

- Bereich Zeitstempel
  Die Verfügbarkeit der Zeitstempel ist abhängig von den konfigurierten Aufzeichnungsbedingungen.
- Bereich Messpunkte
- Bereich Zykluszeit
  Bei äquidistanten Aufzeichnungen wird die Zeitdauer zwischen zwei Messpunkten angezeigt.
  Diese Zeit kann z. B. im Formeleditor zur Analyse genutzt werden.

Beispiel Zeitstempel

Das folgende Bild zeigt beispielhaft die Zeitstempel einer Messung:

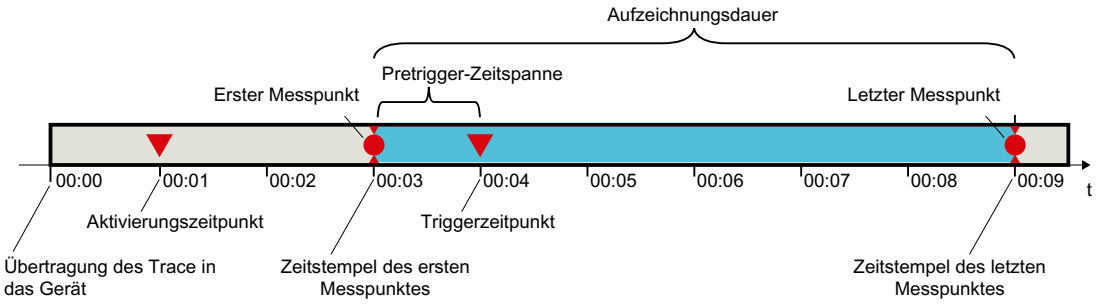

Hinweis

Analyse von Messungen mit sporadisch auftretender Aufzeichnungsbedingung

Berücksichtigen Sie für die Auswertung Ihrer Messungen, dass die Aufzeichnungsbedingung zwischen dem Aktivierungszeitpunkt und dem Triggerzeitpunkt nicht erfüllt gewesen sein kann.

6.4 Bedienen

6.4.1 Schnelleinstieg

Diese Beschreibung zeigt beispielhaft für die S7-1500 CPU die Schritte für eine Aufzeichnung. Die gezeigten Einstellungen können sich geräteabhängig unterscheiden.

Voraussetzung

Ein Gerät ist projektiert, das die Trace- und Logikanalysatorfunktion unterstützt.

Trace anlegen

Das folgende Bild zeigt die Projektnavigation mit dem Systemordner "Traces" unter dem Gerät:

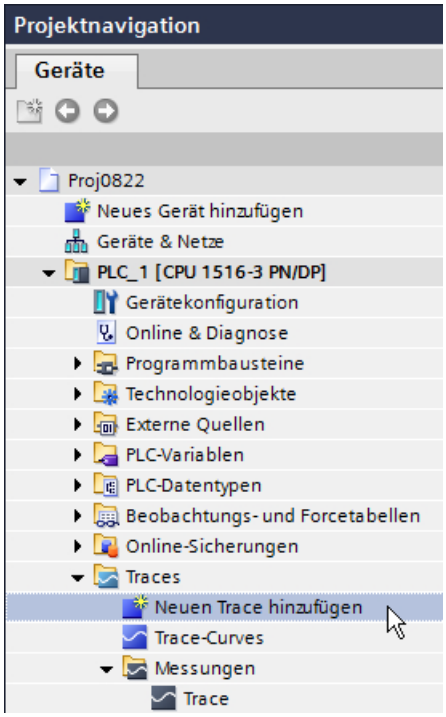

Vorgehen:

1. Doppelklicken Sie auf den Eintrag "Neuen Trace hinzufügen".
   Eine neue Tracekonfiguration wird angelegt und das Register "Konfiguration" im Arbeitsbereich geöffnet.
2. Passen Sie den Namen der Tracekonfiguration durch Klick auf den Text an.

Signale auswählen

Das folgende Bild zeigt die Konfiguration der Signale:

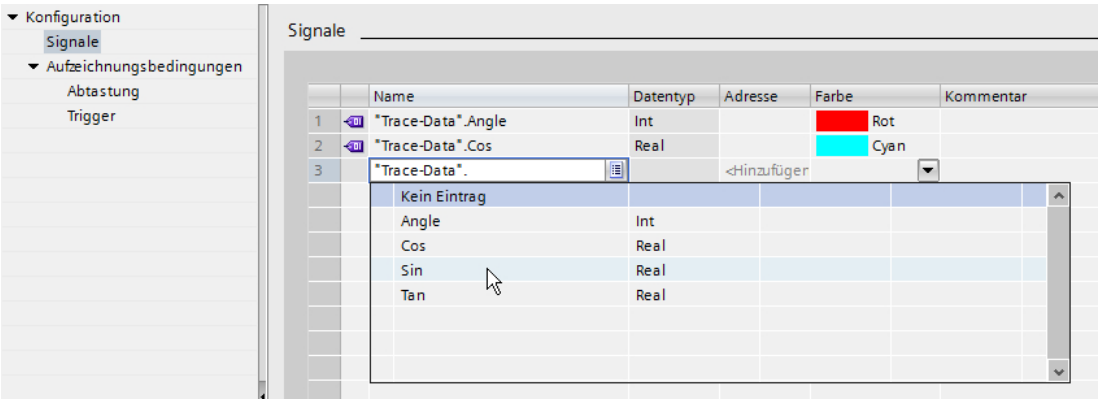

Vorgehen:

1. Wählen Sie im Bereich "Signale" die aufzuzeichnenden Signale aus.
   Oder:
2. Ziehen Sie z. B. aus einer Variablentabelle ein oder mehrere Signale per Drag & Drop in die Signaltabelle.

Aufzeichnungstakt konfigurieren

Das folgende Bild zeigt die Konfiguration der Abtastung:

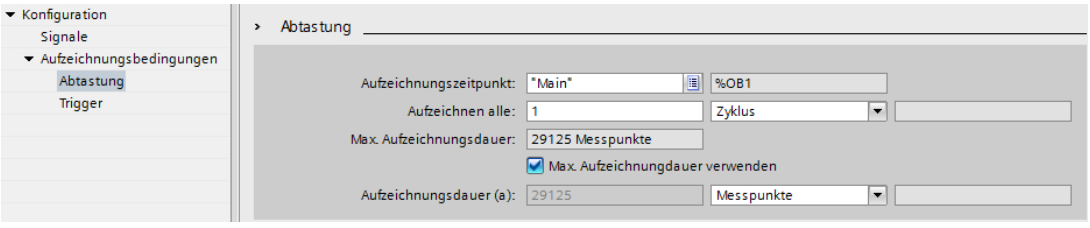

Vorgehen:

1. Konfigurieren Sie die Abtastung.

Trigger konfigurieren

Das folgende Bild zeigt die Konfiguration des Triggers:

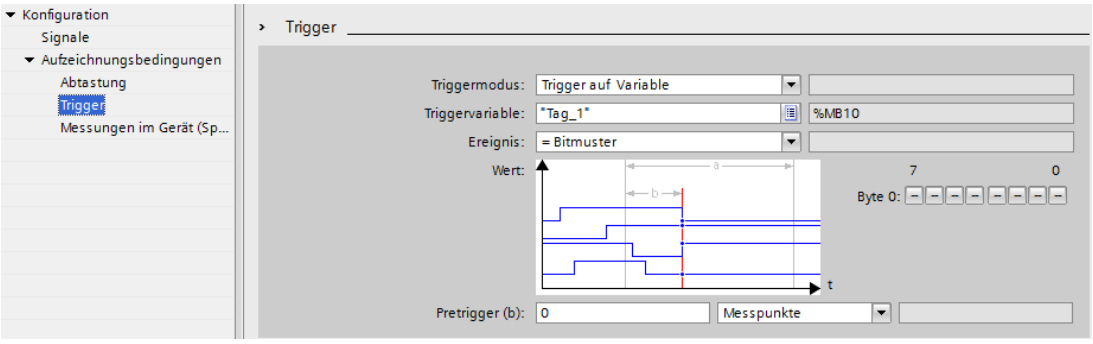

Vorgehen:

1. Konfigurieren Sie den Triggermodus und die Bedingung für den gewählten Trigger.

Tracekonfiguration in das Gerät übertragen

Vorgehen:

1. Übertragen Sie mit der Schaltfläche die Tracekonfiguration in das Gerät.
   Folgende Funktionen werden ausgeführt:
   – Eine Online-Verbindung wird zum Gerät hergestellt.
   – Die Tracekonfiguration wird in das Gerät übertragen.
   – Das Beobachten wird aktiviert.
   – Die Anzeige wechselt zum Register "Diagramm".

Aufzeichnung aktivieren

Vorgehen:

1. Klicken auf die Schaltfläche .

Aufzeichnung anzeigen

Das folgende Bild zeigt das Kurvendiagramm mit einer Aufzeichnung:

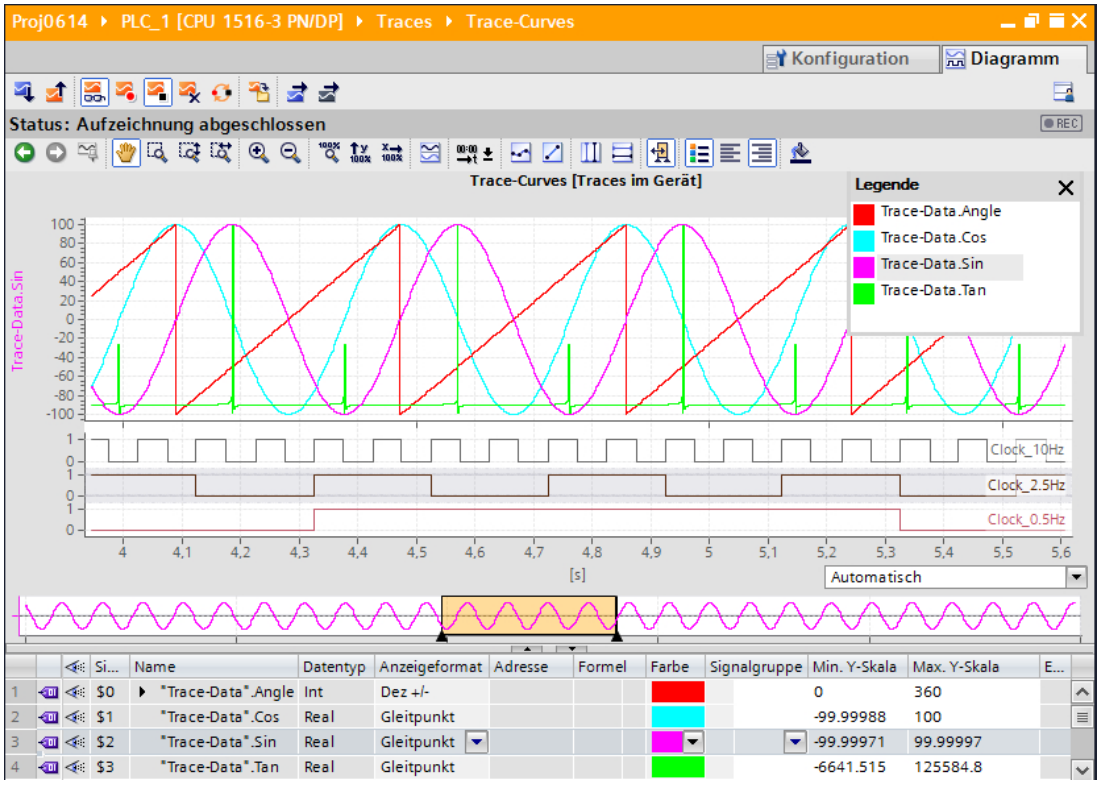

Vorgehen:

1. Warten Sie, bis in der Statusanzeige des Trace der Status "Aufzeichnung läuft" oder "Aufzeichnung abgeschlossen" angezeigt wird.
2. Wechseln Sie in das Register "Diagramm"
3. Klicken Sie in der Signaltabelle auf das Symbol ▸ eines Signals.
   Es werden die einzelnen Bits des Signals für die Anzeige als Bitspur angeboten.
4. Wählen Sie in der Signaltabelle mit dem Symbol die einzelnen Signale und Bits für die Darstellung an oder ab.

Messung im Projekt speichern

Vorgehen:

1. Übertragen Sie mit der Schaltfläche die Messung in das Projekt.
   Die Messung wird in der Projektnavigation unter dem Systemordner "Messungen" angezeigt.

Siehe auch

Oberfläche - Funktionsleiste des Trace (Seite 69)

6.4.2 Anwenden der Tracefunktion - Übersicht

Voraussetzung

Im TIA-Portal ist ein Gerät projektiert, das die Trace- und Logikanalysatorfunktion unterstützt und zu dem eine Online-Verbindung besteht.

Vorgehen

Die folgende Tabelle zeigt eine Handlungsübersicht mit den typischen Schritten beim Arbeiten mit der Trace- und Logikanalysatorfunktion.

Schritt	Beschreibung
1	Trace anlegen (Seite 94)
2	Trace konfigurieren (Seite 102)
3	Tracekonfiguration in das Gerät übertragen (Seite 97)
4	Trace im Gerät aktivieren / deaktivieren (Seite 97)
5	Aufzeichnung beobachten (Seite 98)
6	Messungen im Projekt speichern (Seite 99)
7	Aufzeichnung anzeigen (Seite 98)
8	Laufende Aufzeichnung analysieren (Seite 99)
9	Aufzeichnungen vergleichen (überlagerte Messungen) (Seite 108)

Siehe auch

Konfiguration anzeigen (Seite 95)

6.4.3 Projektnavigation

6.4.3.1 Trace anlegen

Im Projektnavigator können Traces in Form von Tracekonfigurationen angelegt werden.

Die folgende Anleitung beschreibt, wie Sie eine Tracekonfiguration unter dem Systemordner "Traces" anlegen.

Vorgehen

Um eine Tracekonfiguration anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Eintrag "Neuen Trace hinzufügen".

Eine neue Tracekonfiguration wird angelegt.

6.4.3.2 Konfiguration anzeigen

Voraussetzung

Im Systemordner "Traces" ist eine Tracekonfiguration, ein Trace im Gerät, eine Messung oder eine überlagerte Messung vorhanden.

Vorgehen

Um eine Tracekonfiguration anzuzeigen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf das entsprechende Symbol der Tracekonfiguration, eines Trace im Gerät, einer Messung oder einer überlagerten Messung. Im Arbeitsbereich wird das Register "Konfiguration" oder "Diagramm" geöffnet.
2. Klicken Sie zur Anzeige ggf. auf das Register "Konfiguration".

Hinweis

Schreibschutz

Von einem Trace im Gerät und in allen Messungen werden die Konfigurationsdaten schreibgeschützt dargestellt.

Siehe auch

Oberfläche - Projektnavigation Ordner "Traces" (Seite 64)

6.4.3.3 Diagramm anzeigen

Voraussetzung

Im Systemordner "Traces" ist ein Trace im Gerät, eine Messung oder eine überlagerte Messung vorhanden.

Vorgehen

Um ein Diagramm anzuzeigen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf das entsprechende Symbol eines Traces im Gerät, einer Messung oder einer überlagerten Messung.
   Im Arbeitsbereich wird das Register "Konfiguration" oder "Diagramm" geöffnet.
2. Klicken Sie zur Anzeige ggf. auf das Register "Diagramm".

Siehe auch

Oberfläche - Projektnavigation Ordner "Traces" (Seite 64)

6.4.3.4 Überlagerte Messung anlegen

Im Projektnavigator können überlagerte Messungen mit einer Vergleichsfunktionalität für verschiedene Messungen angelegt werden.

Die folgende Anleitung beschreibt, wie Sie eine überlagerte Messung unter dem Systemordner "Überlagerte Messungen" anlegen.

Voraussetzung

Ein Gerät ist projektiert, das die Trace- und Logikanalysatorfunktion unterstützt.

Vorgehen

Um eine überlagerte Messung anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Systemordner "Messungen" eine oder mehrere Messungen.
2. Ziehen Sie per Drag & Drop die Messungen auf den Systemordner "Überlagerte Messungen".

Eine neue überlagerte Messung wird angelegt. Diese enthält Kopien der selektierten Messungen.

6.4.3.5 Strukturierung der Objekte in Gruppen

Im Projektnavigator können Sie in Systemordnern Gruppen anlegen. Nutzen Sie diese Möglichkeit zum Strukturieren der Ansicht bei vielen Objekten.

Die folgende Anleitung beschreibt beispielhaft für Messungen, wie Sie die Messungen in Gruppen zusammenfassen. Die gleiche Funktionalität steht bei den Systemordnern Traces und überlagerten Messungen zur Verfügung.

Hinweis

Anzeige von Traces im Gerät auch in Gruppen möglich

Eine Tracekonfiguration in einer Gruppe mit einem namensgleichen Trace im Gerät wird unter der Gruppe dargestellt. Es werden somit nicht zwingend alle Traces im Gerät auf erster Ebene im Ordner "Traces" angezeigt.

Voraussetzung

Im Systemordner "Messungen" sind Messungen vorhanden.

Vorgehen

Um Messungen in Gruppen zu strukturieren, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklicken auf den Systemordner "Messungen" den Kontextmenübefehl "Neue Gruppe hinzufügen" aus.
   Ein neuer Gruppenordner wird angelegt.
2. Vergeben Sie der neu angelegten Gruppe einen aussagekräftigen Namen.
3. Wiederholen Sie Schritt 1, bis alle gewünschten Gruppen angelegt sind.
   (Auch das Anlegen von Untergruppen (Gruppen in Gruppen) ist möglich.)
4. Ziehen Sie per Drag & Drop die entsprechenden Messungen auf die angelegten Gruppenordner.

6.4.4 Arbeitsbereich - Allgemein

6.4.4.1 Tracekonfiguration in das Gerät übertragen

Voraussetzung

- Eine gültige Tracekonfiguration befindet sich im Systemordner "Traces".
- Die maximale Anzahl Traces im Gerät ist noch nicht erreicht.

Vorgehen

Um eine Tracekonfiguration in das Gerät zu übertragen, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine gültige Tracekonfiguration im Arbeitsbereich.
2. Klicken Sie auf die Schaltfläche .

Ergebnis

Die Tracekonfiguration wird in das Gerät übertragen.

6.4.4.2 Trace im Gerät aktivieren / deaktivieren

Voraussetzung

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace befindet sich im Gerät.
- Der Trace im Gerät wird im Arbeitsbereich angezeigt.
- Die Schaltfläche ist zum Beobachten für den angezeigten Trace aktiviert.

Trace im Gerät aktivieren

Um die Aufzeichnung für einen Trace im Gerät zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche .
   Der Trace im Gerät wird aktiviert und startet die Aufzeichnung entsprechend der konfigurierten Triggerbedingung. Die Triggerbedingung ist gerätespezifisch und im Kapitel Konfigurieren unter dem jeweiligen Gerät (Seite 111) beschrieben.
   Der aktuelle Zustand der Aufzeichnung wird in der Statusanzeige des Trace angezeigt.

   Hinweis

   Wenn eine Aufzeichnung neu gestartet wird, gehen die bisher aufgezeichneten Werte verloren.

   Um die aufgezeichneten Werte zu sichern, speichern Sie die Messung im Projekt (Seite 99), bevor Sie die Aufzeichnung erneut aktivieren.

Trace im Gerät deaktivieren

Um einen aktivierten Trace im Gerät zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche .
   Der Trace im Gerät wird deaktiviert.
   Die Statusanzeige des Trace wechselt auf "inaktiv".

6.4.4.3 Aufzeichnung anzeigen

Voraussetzung

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace mit Aufzeichnung befindet sich im Gerät.

Oder:

- Eine Messung befindet sich im Systemordner "Messungen".

Vorgehen

Um die Aufzeichnung anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie einen Trace im Gerät aus.
2. Doppelklicken Sie auf den ausgewählten Trace.
3. Aktivieren Sie ggf. die Schaltfläche zum Beobachten.

Oder:

1. Wählen Sie im Systemordner "Messungen" eine Messung aus.
2. Doppelklicken Sie auf die ausgewählte Messung.

Ergebnis

Die Aufzeichnung wird im Register "Diagramm" angezeigt.

Siehe auch

Oberfläche - Projektnavigation Ordner "Traces" (Seite 64)

6.4.4.4 Laufende Aufzeichnung analysieren

Voraussetzungen

- Eine laufende Aufzeichnung wird im Register "Diagramm" angezeigt

Speichern der aktuell aufgezeichneten Daten als Messung

Um einen bestimmten Zeitbereich einer laufenden Aufzeichnung zu analysieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche .
   Die bisher aufgezeichneten Daten werden zu den Messungen hinzugefügt.
   Die aktuelle Aufzeichnung wird dadurch nicht beeinflusst und läuft ohne Unterbrechung weiter.

Analysieren der Messung

Um die gespeicherte Messung anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Systemordner "Messungen" die gerade gespeicherte Messung mit Doppelklick
   Das Register "Diagramm" der Messung wird im Arbeitsbereich geöffnet.

Siehe auch

Aufzeichnung anzeigen (Seite 98)

Arbeitsbereich - Register Diagramm (Seite 102)

6.4.4.5 Messungen im Projekt speichern

Voraussetzung

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace mit Aufzeichnung befindet sich im Gerät.
- Die Daten des Trace im Gerät müssen mindestens einmal im Kurvendiagramm angezeigt worden sein. Für die Anzeige werden die Daten der Aufzeichnung aus dem Gerät geladen.

Vorgehen

Um eine Aufzeichnung im Projekt zu speichern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Trace im Gerät mit den aufgezeichneten Daten.
2. Stellen Sie ggf. durch Aktivierung der Schaltfläche sicher, dass die aktuellen Daten aus dem Gerät geladen werden.
3. Warten Sie nach Aktivierung der Schaltfläche , bis alle Daten geladen wurden und angezeigt werden.
4. Klicken Sie auf die Schaltfläche .
   Die Messung wird im Systemordner "Messungen" hinzugefügt.
5. Speichern Sie das Projekt im TIA-Portal.

Hinweis

Messungen erstellen

Von einem Trace im Gerät kann jederzeit eine Messung erstellt werden.

Nutzen Sie diese Funktionalität, um z. B. während einer Aufzeichnung die bis dahin aufgezeichneten Daten zu speichern und dann als statische Messung zu analysieren.

6.4.4.6 Messungen exportieren und importieren

Voraussetzung

Für den Export befindet sich mindestens eine Messung im Systemordner "Messungen".

Messungen exportieren

Um eine Messung zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklick auf eine Messung aus dem Systemordner "Messungen" den Kontextmenübefehl "Messung exportieren".
2. Wählen Sie einen Ordner, Dateinamen und Dateityp zum Speichern der Messung.
3. Klicken Sie auf die Schaltfläche "Speichern".

Messungen importieren

Um eine Messung zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklick auf den Systemordner "Messungen" den Kontextmenübefehl "Messung importieren".
2. Wählen Sie die zu importierende Datei z. B. vom Dateityp "*.ttrecx" mit der Messung.
3. Klicken Sie auf die Schaltfläche "Öffnen".
   Die importierte Messung wird mit dem Dateinamen im Systemordner "Messungen" angezeigt.

Hinweis

Tracekonfigurationen exportieren und importieren

Die gleiche Funktionalität steht für den Export und Import von Tracekonfigurationen zur Verfügung.

6.4.4.7 Tracekonfiguration aus dem Gerät in das Projekt übertragen

Voraussetzung

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace befindet sich im Gerät.

Vorgehen

Um eine Tracekonfiguration in das Projekt zu übertragen, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen Trace im Gerät.
2. Aktivieren Sie ggf. die Schaltfläche zum Beobachten.
3. Klicken Sie auf die Schaltfläche zum Übertragen der Tracekonfiguration aus dem Gerät.

Ergebnis

Die Konfiguration wird als neue Tracekonfiguration in den Systemordner "Traces" übernommen.

Eine gleichnamige Tracekonfiguration im Systemordner wird überschrieben.

6.4.4.8 Traces im Gerät löschen

Voraussetzung

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace befindet sich im Gerät.

Vorgehen

Um einen Trace im Gerät zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen Trace im Gerät.
2. Aktivieren Sie ggf. die Schaltfläche zum Beobachten.

3. Klicken Sie auf die Schaltfläche .
   Ein Dialog mit Aufforderung zum Bestätigen wird geöffnet.
4. Bestätigen Sie den Dialog zum Löschen.

Oder

1. Selektieren Sie in der Projektnavigation ein oder mehrere Traces im Gerät / .
2. Drücken Sie <Entf>, um die Traces im Gerät zu löschen.
   Ein Dialog mit Aufforderung zum Bestätigen wird geöffnet.
3. Wählen Sie ggf. eine Option zum Löschen aus und bestätigen Sie den Dialog zum Löschen.

6.4.5 Arbeitsbereich - Register Konfiguration

6.4.5.1 Trace konfigurieren

Voraussetzung

Das Register "Konfiguration" im Arbeitsbereich ist geöffnet.

Trace konfigurieren

Bei der Konfiguration legen Sie die Aufzeichnungs- und Triggerbedingungen fest und wählen Sie die aufzuzeichnenden Signale aus.

Siehe Kapitel Konfigurieren unter dem jeweiligen Gerät (Seite 111).

Hinweis

Tracekonfiguration speichern

Die Tracekonfiguration speichern Sie mit dem Projekt im TIA-Portal.

Wenn Sie das Projekt ohne Speichern schließen, wird die Konfiguration verworfen.

Siehe auch

Konfiguration anzeigen (Seite 95)

6.4.6 Arbeitsbereich - Register Diagramm

6.4.6.1 Verwendung des Kurvendiagramms

Das Kurvendiagramm zeigt die in der Signaltabelle ausgewählten Signale einer Aufzeichnung an.

Der Darstellungsbereich kann beliebig gezoomt werden. Mit Hilfe von Mess-Cursorn können einzelne Werte zur Anzeige in der Signaltabelle ausgewählt werden.

Die folgenden Handlungsanleitungen beschreiben beispielhaft die Verwendung des Kurvendiagramms und den Einsatz der Mess-Cursor.

Voraussetzungen

- Ein Trace im Gerät oder eine Messung ist zur Anzeige ausgewählt.
- Bei einem Trace im Gerät ist die Schaltfläche zum Beobachten aktiviert.
- Das Register "Diagramm" im Arbeitsbereich ist geöffnet.

Beobachten einer laufenden Aufzeichnung

Um alle Daten einer laufenden Aufzeichnung anzuzeigen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie "Alles anzeigen" über die Schaltfläche .

Der gesamte Zeitbereich und alle Werte der laufenden Aufzeichnung werden angezeigt.

Um ein konstantes Zeitfenster einer laufenden Aufzeichnung anzuzeigen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie "Alles anzeigen" über die Schaltfläche .
2. Wählen Sie den gewünschten Zeitbereich über die Schaltfläche aus.

Die Kurvenanzeige wird bei gleichbleibender Skalierung des Zeitbereichs aktualisiert.

Auswertung eines bestimmten Zeitpunkts einer Aufzeichnung

Um die Werte für einen bestimmten Messpunkt anzuzeigen, gehen Sie folgendermaßen vor:

1. Blenden Sie die vertikalen Mess-Cursor über die Schaltfläche ein.
2. Verschieben Sie einen Mess-Cursor mit der Maus an die gewünschte Position der Aufzeichnung.
   Die Werte der Signale werden in der Signaltabelle und in der Palette "Mess-Cursor" der TaskCard "Trace" angezeigt.

Auswertung der Differenz zwischen zwei Messpunkten

Um die Differenz anzuzeigen, gehen Sie folgendermaßen vor:

1. Blenden Sie die vertikalen Mess-Cursor über die Schaltfläche ein.
2. Verschieben Sie die beiden Mess-Cursor mit der Maus an die gewünschten Messpunkte der Aufzeichnung.
   Die Werte der Signale und die Differenz werden in der Signaltabelle und in der Palette "Mess-Cursor" der TaskCard "Trace" angezeigt.

Horizontale Mess-Cursor verwenden

Um das Erreichen eines bestimmten Werts zu prüfen, gehen Sie folgendermaßen vor:

1. Blenden Sie die horizontalen Mess-Cursor über die Schaltfläche ein.
2. Verschieben Sie einen Mess-Cursor mit der Maus auf den gewünschten Wert der Aufzeichnung.
   Die Werte der Mess-Cursor für das gewählte Signal werden in der Palette "Mess-Cursor" der TaskCard "Trace" angezeigt.

Angezeigten Zeitbereich verschieben

Um den angezeigten Zeitbereich zu verschieben, gehen Sie folgendermaßen vor:

1. Wählen Sie einen Zeitbereich über die Schaltfläche aus.
2. Schieben Sie die Kurve durch Drehen des Mausrads bei gedrückter Taste <Shift> in den gewünschten Zeitbereich.

Ein Signal in den Vordergrund bringen

1. Blenden Sie die Legende über die Schaltfläche ein.
2. Klicken Sie auf ein Signal in der Legende.

Oder:

1. Klicken Sie auf ein Signal im Kurvendiagramm.

Das Signal wird im Vordergrund dargestellt und in der Signaltabelle hervorgehoben/selektiert. Die Werteachse wird für das gewählte Signal aktualisiert.

Siehe auch

Diagramm anzeigen (Seite 95)

Oberfläche - Kurvendiagramm (Seite 71)

Oberfläche - Signaltabelle (Seite 76)

6.4.6.2 Verwendung der Signaltabelle

Die Signaltabelle zeigt die Signale von einem Trace im Gerät oder einer Messung an. Bei einer überlagerten Messung werden die in der Signalauswahl vorselektierten Signale angezeigt. In der Tabelle können Sie einzelne Signale für die Anzeige ein- oder ausblenden und Eigenschaften für die Anzeige anpassen.
Für einige Datentypen können auch einzelne Bits ausgewählt und als Bitspur angezeigt werden.

Die folgende Handlungsanleitung beschreibt die Bedienung der Signaltabelle.

Voraussetzungen

- Ein Trace im Gerät oder eine Messung ist im Register "Diagramm" geöffnet.
- Bei einem Trace im Gerät ist die Schaltfläche zum Beobachten aktiviert.
- Für die Anzeige einzelner Bits als Bitspur:
  Mindestens ein aufgezeichnetes Signal unterstützt die Anzeige als Bitspur.

Einzelne Signale ein- oder ausblenden und die Farbe ändern

Um die Anzeige Ihren Anforderungen anzupassen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Spalte auf das Symbol des jeweiligen Signals, um es für die Anzeige ab- oder anzuwählen.
2. Klicken Sie für das jeweilige Signal in die Spalte "Farbe" und wählen Sie eine Farbe aus.
   Die voreingestellte Farbe des Signals wird geändert.

Ein Signal in den Vordergrund bringen

1. Wählen Sie in der Signaltabelle die Zeile des Signals an.
   Die Y-Skala des Signals wird angezeigt.
   Die Kurve des Signals wird im Kurvendiagramm in den Vordergrund gebracht.

Einzelne Bits für die Anzeige als Bitspur auswählen

Um einzelne Bits als Bitspur im unteren Kurvendiagramm anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Signaltabelle auf das Symbol eines Signals.
2. Klicken Sie in der geöffneten Bitauswahl des Signals auf das Symbol .
   Die Bits werden für die Anzeige ab- bzw. angewählt.

Siehe auch

Aufzeichnung anzeigen (Seite 98)

6.4.6.3 Verwendung der Signalgruppe in der Signaltabelle

Mit der Signalgruppe können einzelne Signale gleich skaliert werden, was z. B. einen besseren Vergleich der Kurvenverläufe ermöglicht.

Binärsignale können nicht gruppiert werden.

Die folgenden Handlungsanleitungen beschreiben die Anwendung der Signalgruppe.

Hinweis

Speichern der Signalgruppen

Die Signalgruppen können für jede Messung individuell über die Funktion "Aktuelle Ansicht als Standard verwenden" (Schaltfläche) gespeichert werden.

Ohne Speichern der Signalgruppen und des Projekts gehen die angelegten Signalgruppen mit dem Schließen des Registers "Diagramm" verloren.

Voraussetzungen

- Ein Trace im Gerät oder eine Messung wird angezeigt.
- Bei einem Trace im Gerät ist die Schaltfläche zum Beobachten aktiviert.
- Das Register "Diagramm" im Arbeitsbereich ist geöffnet.
- Es befinden sich mindestens zwei Signale in der Signaltabelle, die nicht vom Typ BOOL sind.

Signale einer Signalgruppe zuordnen

Um eine Signalgruppe anzulegen und Signale dieser Gruppe zuzuordnen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Zeile oder Zelle des gewünschten Signals in der Signaltabelle an.
2. Klicken Sie auf das graue Feld in der Spalte "Signalgruppe"
   Das Kettensymbol wird im grauen Feld angezeigt und der Name für die Signalgruppe vorbelegt: Group
3. Klicken Sie auf das graue Feld von weiteren Signalen, die dieser Signalgruppe zugeordnet werden sollen.

Oder:

1. Klicken Sie für ein zu gruppierendes Signal in das Textfeld der Spalte "Signalgruppe".
2. Geben Sie einen Namen für die Gruppe ein.
3. Geben Sie bei weiteren Signalen im jeweiligen Textfeld den gleichen Gruppennamen ein oder wählen den Gruppennamen über die Klappliste aus.

Die Y-Skalen der gruppierten Signale werden mit den Werten des zuerst gewählten Signals skaliert. Änderungen eines Skalenwerts wirken sich immer auf die gesamte Gruppe aus.

Signale aus einer Signalgruppe entfernen

Um die Zuordnung eines Signals zu einer Signalgruppe zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie für das gewünschte Signal in der Spalte "Signalgruppe" auf das Kettensymbol .

Oder:

1. Klicken Sie für das gewünschte Signal in der Spalte "Signalgruppe" auf das Textfeld.
2. Drücken Sie die Taste <Entf>.

Oder:

1. Wählen Sie für mehrere Signale über die Tasten <Shift> und <Strg> das jeweilige Textfeld in der Spalte "Signalgruppe" aus.
2. Drücken Sie die Taste <Entf>.

Die Signale werden aus der Signalgruppe entfernt bzw. die Signalgruppe gelöscht.

6.4.6.4 Beobachten von schnellen Signalen

Voraussetzungen

- Es besteht eine Online-Verbindung zum Gerät.
- Ein Trace befindet sich im Gerät.

Vorgehen

Um den Verlauf eines schnellen Signals permanent zu beobachten, gehen Sie folgendermaßen vor:

1. Wählen Sie einen Trace im Gerät aus.
2. Doppelklicken Sie auf den ausgewählten Trace.
3. Aktivieren Sie die Schaltfläche zum Beobachten.
4. Aktivieren Sie die Schaltfläche zum automatischen Wiederholen der Aufzeichnung.

Ergebnis

Die Aufzeichnung wird nach jedem Aufzeichnungsende automatisch wieder aktiviert. Die Anzeige im Kurvendiagramm ist vergleichbar mit der Anzeige eines Oszilloskops.

6.4.6.5 Aufzeichnungen vergleichen (überlagerte Messungen)

Voraussetzung

- Eine überlagerte Messung ist angelegt oder wird implizit durch Ziehen der Messungen auf den Systemordner "Überlagerte Messungen" angelegt.
  Siehe auch Überlagerte Messung anlegen (Seite 96).

Messungen zum Vergleichen hinzufügen

Um Messungen zu vergleichen, fügen Sie die zu vergleichenden Messungen der überlagerten Messung hinzu. Gehen Sie dabei folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation per Drag & Drop aus dem Systemordner "Messungen" eine oder mehrere Messungen auf das Symbol der überlagerten Messung .

Oder:

1. Importieren Sie über den Kontextmenübefehl "Messung importieren" gespeicherte Messungen.

Eine Kopie der Messungen wird der überlagerten Messung hinzugefügt.

Hinweis

Änderungen der Einstellungen von Messungen innerhalb der überlagerten Messung wirken sich nicht auf die ursprünglichen Messungen aus. Die ursprünglichen Messungen bleiben unverändert.

Signale der Messungen für die Signaltabelle auswählen

Um die Signale für die Signaltabelle im Register "Diagramm" auszuwählen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf das Symbol der überlagerten Messung .
   Im Arbeitsbereich werden die Register der überlagerten Messung angezeigt.
2. Klicken Sie im Arbeitsbereich auf das Register "Signalauswahl".
   Die Signale aller Messungen werden in der Tabelle angezeigt.
3. Aktivieren bzw. deaktivieren Sie die Optionskästchen der Signale, die in der Signaltabelle sichtbar bzw. nicht sichtbar sein sollen.

Die aktivierten Signale werden in der Signaltabelle des Registers "Diagramm" angezeigt.

Verwendung der Signaltabelle

Gehen Sie zum Öffnen und Verwenden der Signaltabelle folgendermaßen vor:

1. Klicken Sie im Arbeitsbereich auf das Register "Diagramm".
2. Klicken Sie innerhalb des Registers "Diagramm" auf das Register "Signale".
3. Verwenden Sie die Signaltabelle wie unter Verwendung der Signaltabelle (Seite 104) beschrieben.

Messungen ausrichten

Um die Zeitachse der Messungen für den Vergleich auszurichten, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb des Registers "Diagramm" auf das Register "Messungen".
2. Wählen Sie die Ausrichtung der Messungen über die Optionskästchen aus.
3. Stellen Sie die Ausrichtung und ggf. einen Offset für die Ausrichtung der einzelnen Messungen ein.

Die Messungen werden an der Zeitachse entsprechend zueinander ausgerichtet.

(Die exakte Ausrichtung von zwei Messungen ist im folgenden Kapitel beschrieben.)

Verwendung des Kurvendiagramms

Gehen Sie zum Öffnen und Verwenden des Kurvendiagramms folgendermaßen vor:

1. Klicken Sie im Arbeitsbereich auf das Register "Diagramm".
2. Verwenden Sie das Kurvendiagramm wie unter Verwendung des Kurvendiagramms (Seite 102) beschrieben.

Siehe auch

Messungen exakt ausrichten (überlagerte Messungen) (Seite 109)

6.4.6.6 Messungen exakt ausrichten (überlagerte Messungen)

Voraussetzung

- Eine überlagerte Messung ist angelegt.
- Messungen zum Vergleichen sind der überlagerten Messung hinzugefügt.
- Signale der Messungen für die Signaltabelle sind ausgewählt.
- Das Register "Diagramm" der überlagerten Messung ist im Arbeitsbereich geöffnet.

Messungen exakt ausrichten mit Positionsdifferenz ΔX

Um die Zeitachse von zwei Messungen exakt auszurichten, gehen Sie folgendermaßen vor:

1. Blenden Sie die vertikalen Mess-Cursor über die Schaltfläche ein.
2. Vergrößern Sie den Zeitbereich z. B. mit der Schaltfläche so weit, bis Sie den ersten Mess-Cursor exakt auf den gewünschten Referenzpunkt der ersten Messung positionieren können.
3. Verschieben Sie den ersten Mess-Cursor mit der Maus an die gewünschte Position.
4. Suchen Sie den Referenzpunkt der zweiten Messung z. B. durch Umschalten auf "Alles anzeigen" mit der Schaltfläche .
5. Vergrößern Sie den Zeitbereich z. B. mit der Schaltfläche so weit, bis Sie den zweiten Mess-Cursor exakt auf den gewünschten Referenzpunkt der zweiten Messung positionieren können.
6. Verschieben Sie den zweiten Mess-Cursor mit der Maus an die gewünschte Position.
7. Öffnen Sie die Task Card "Trace".
8. Wählen Sie in der Palette "Mess-Cursor" den Wert der Positionsdifferenz ΔX aus.
9. Kopieren Sie den Wert in die Zwischenablage.
10. Fügen Sie den Wert der Zwischenablage in die Zelle Offset der ersten oder zweiten Messung ein.

Die beiden Messungen sind exakt an den gewünschten Messpunkten zueinander ausgerichtet.

Hinweis

Beachten Sie beim Einfügen der Positionsdifferenz als Offset, dass Sie ggf. das Vorzeichen anpassen müssen.

6.4.6.7 Drucken einer Aufzeichnung

Das Kurvendiagramm unterstützt das Speichern der Anzeige als Grafik und das Kopieren der Anzeige in die Zwischenablage. Nutzen Sie diese Funktionen (Seite 71) auch zum Drucken.

6.5 Geräte

6.5.1 S7-1200/1500 CPUs

6.5.1.1 Erfassbare Variablen

Geräteabhängige Erfassung von Variablen

Die folgende Aufzählung zeigt, aus welchen Operandenbereichen Variablen aufgezeichnet werden können:

- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Merker
- Datenbausteine

Datentypen

Eine Auswahl an elementaren und zusammengesetzten Datentypen kann aufgezeichnet werden. Die Verfügbarkeit der einzelnen Datentypen hängt vom eingesetzten Gerät ab.

Weitere Informationen finden Sie in der Hilfe unter "Übersicht über die gültigen Datentypen".

Die folgende Tabelle listet die unterstützten Datentypen auf:

Datentypen	Hinweis
Binärzahlen	
BOOL	-
Bitfolgen	
BYTE	-
WORD	-
DWORD	-
LWORD [1)]	Symbolischer Name erforderlich
Ganzzahlen	
SINT	-
USINT	-
INT	-
UINT	-
DINT	-
UDINT	-
LINT [1)]	Symbolischer Name erforderlich
ULINT [1)]	Symbolischer Name erforderlich
Gleitpunktzahlen	
REAL	-

Datentypen	Hinweis
LREAL	Symbolischer Name erforderlich
Zeiten	
TIME	-
LTIME [1)]	-
Datum und Uhrzeit	
DATE	-
TOD	-
LTOD [1)]	-
LDT [1)]	-

1) Wird nicht von S7-1200 unterstützt.

6.5.1.2 Lebensdauer der Tracekonfiguration und aufgezeichneten Werte im Gerät

Tracekonfigurationen im Gerät bleiben bei NETZ-AUS erhalten. Nach dem Wiederanlauf der CPU wird die Aufzeichnung neu aktiviert.

Aufgezeichnete Werte gehen durch den Wiederanlauf verloren.

Hinweis

Laden einer Konfiguration in das Gerät im Betriebszustand "STOP"

Beachten Sie nach dem Laden einer Konfiguration im Betriebszustand "STOP", dass Sie die Traces im Gerät prüfen und ggf. neu aktivieren oder neu übertragen müssen.

Hinweis

Bei Änderungen von Triggervariablen, die Auswirkungen auf die Adresse haben, muss auch die Tracekonfiguration neu in das Gerät übertragen werden.

Das ist z. B. der Fall, wenn ein Datenbaustein verkürzt oder verlängert wird oder der Datentyp geändert wird.

6.5.1.3 Aufzeichnungsebenen

Alle Ablaufebenen können für den Aufzeichnungstakt verwendet werden. Die zyklischen Ablaufebenen werden über die Schaltfläche zur Auswahl angeboten. Bei nicht periodischen Aufzeichnungsebenen ist die Aufzeichnungsdauer undefiniert.

Hinweis

Die Messwerte werden jeweils am Ende des OBs nach der Bearbeitung des Anwenderprogramms aufgezeichnet.

6.5.1.4 Mengengerüst

Die folgende Tabelle zeigt das maximale Mengengerüst, das mit der Trace- und Logikanalysatorfunktion aufgezeichnet werden kann:

Gerät	Maximale Anzahl Traces im Gerät	Maximale Anzahl Signale pro Tracekonfiguration
S7-1200 (ab Firmwarestand V4.0)	2	16
S7-1500, ET 200SP, CPU 1516pro-2 PN, S7-1500 Software Controller	mindestens 4 (abhängig vom CPU-Typ)	16

Beispiel CPU 1516-3 PN/DP

- Maximal 3854 Messpunkte bei 16 Signalen von PLC-Variablen des Datentyps DWORD
- Maximal 21844 Messpunkte bei 16 Signalen von PLC-Variablen des Datentyps BOOL
- Maximal 58250 Messpunkte bei einem Signal von einer PLC-Variable des Datentyps BOOL

Weitere Informationen dazu finden Sie auch als FAQ unter der Beitrags-ID 102781176 (http://support.automation.siemens.com/WW/view/de/102781176).

6.5.1.5 CPU-Belastung durch Traceaufzeichnung

Eine Traceaufzeichnung erhöht die Laufzeit der jeweiligen Aufzeichnungsebene, die bei hoher Auslastung der CPU zu einem Ablaufebenen-Überlauf führen kann.

Abhilfe bei einem Ablaufebenen-Überlauf:

- **Tracekonfiguration ändern**
  1) Konfigurieren Sie weniger Variablen und Signale.
  2) Erhöhen Sie dann die Anzahl Variablen und Signale schrittweise bis zur maximalen Signalanzahl ohne Ablaufebenen-Überlauf.
- **Langsamere Aufzeichnungsebene wählen**

6.5.1.6 Software-Oberfläche der Konfiguration

Aufbau der Oberfläche

Anzeigebereiche im Register "Konfiguration" des Arbeitsbereichs

Die Einstellmöglichkeiten unterscheiden sich in Abhängigkeit vom konfigurierten Gerät.

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

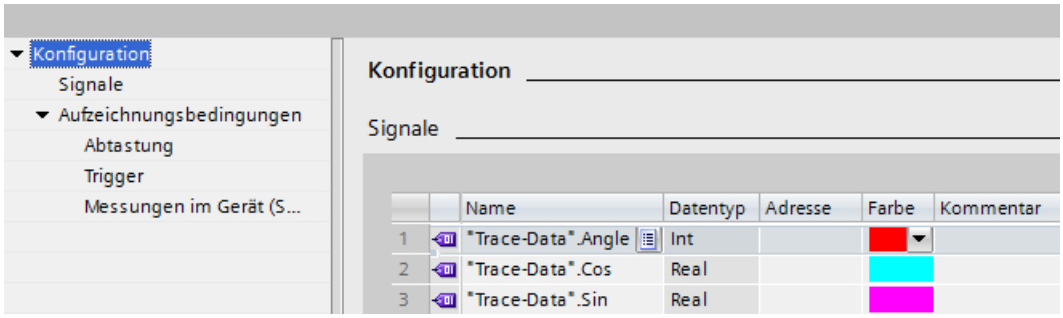

Die Bereichsnavigation bietet folgende Einträge zur Auswahl an:

- Konfiguration
  - Signale (Seite 114)
  - Aufzeichnungsbedingungen (Seite 115)

Eigenschaften einer Tracekonfiguration anzeigen und ändern

Ein Trace wird in der Projektnavigation ausgewählt und im Register "Konfiguration" angezeigt.

Die Tracekonfiguration kann nur offline geändert werden und wird online schreibgeschützt dargestellt.

Oberfläche - Signale

Der Bereich "Signale" zeigt eine Tabelle, in der die aufzuzeichnenden Signale für die angewählte Tracekonfiguration konfiguriert werden.

Signale können auch per Drag & Drop in die Tabelle eingefügt werden.
Die Sortierung der Signale ist per Drag & Drop möglich.

Einstellmöglichkeiten und Anzeigen unter "Signale"

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

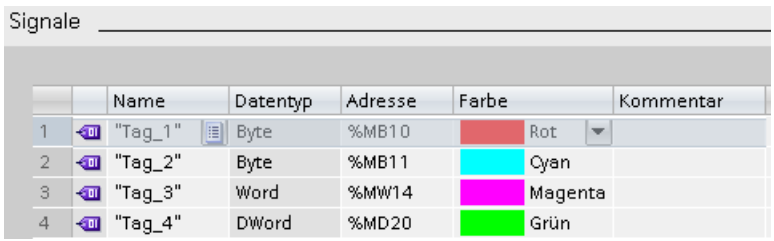

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

Spalte	Symbol	Beschreibung
		Anzeige des Signalsymbols für ein ausgewähltes Signal
"Name"	-	Eingabefeld für die Bezeichnung oder Adresse des Signals **Beispiele:** • "Datenbaustein_1".Druck • M0.0 • DB1.DBW3
-		Schaltfläche zum Öffnen der Signalauswahl-Tabelle Die Schaltfläche wird bei Auswahl der Tabellenzeile eingeblendet. Mit Klick auf das Symbol öffnen Sie eine Tabelle, welche mögliche Signale zur Auswahl anbietet. Das ausgewählte Signal wird im Eingabefeld angezeigt.
"Datentyp"	-	Textfeld mit der Anzeige des Datentyps für das Signal
"Adresse"	-	Eingabefeld für die Adresse des Signals Bei optimierten / typsicheren Variablen bleibt das Feld leer.
"Farbe"	-	Textfeld zur Anzeige und Auswahl der Farbe Klicken Sie auf die Signalfarbe, um den Farbauswahldialog zu öffnen.
"Kommentar"		Eingabefeld für einen Kommentar zum Signal

Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle:

Kontextmenübefehl	Beschreibung
"Ausschneiden"	Nicht anwählbar
"Kopieren"	Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.
"Einfügen"	Fügt den Inhalt der Zwischenablage ab der selektierten Zeile ein. Der bestehende Inhalt wird überschrieben.
"Löschen"	Löscht die selektierten Zeilen aus der Tabelle oder löscht den Inhalt der selektierten Zelle.
"Umbenennen"	Schaltet die selektierte Zelle in den Editier-Modus.

Aufzeichnungsbedingungen

Oberfläche - Aufzeichnungsbedingungen

Der Bereich "Aufzeichnungsbedingungen" zeigt für die ausgewählte Tracekonfiguration die Triggerbedingung an und in welchem Takt, wie schnell und wie lange aufgezeichnet wird. Die Konfiguration ist möglich, wenn die Tracekonfiguration im Offline-Modus bzw. im Online-Modus bei deaktivierter Beobachtung angezeigt wird.

Einstellmöglichkeiten und Anzeigen unter "Aufzeichnungsbedingungen"

Das folgende Bild zeigt exemplarisch die Einstellungen für die Abtastung im TIA-Portal:

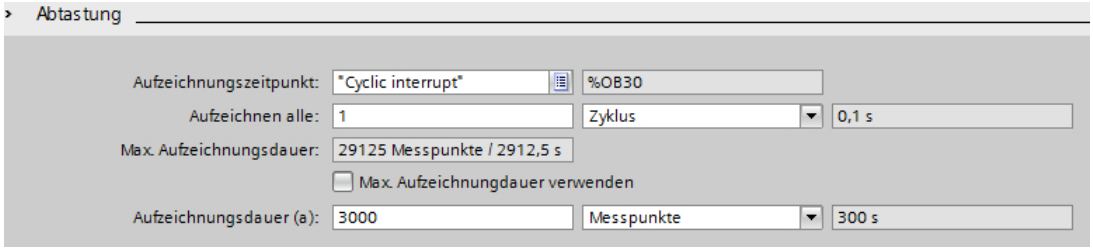

Die folgende Tabelle erklärt die Einstellungen und Anzeigen:

Einstellung/Anzeige		Beschreibung
"Aufzeichnungszeitpunkt"		
	Eingabefeld Aufzeichnungsebene	Auswahl des Aufzeichnungszeitpunkts Siehe Aufzeichnungsebenen (Seite 112)
	Textfeld Adresse des OB	Detailinfos zum ausgewählten Aufzeichnungszeitpunkt
"Aufzeichnen alle"		
	Eingabefeld Untersetzung	Eingabe der Untersetzung bezogen auf den Untersetzungsfaktor und die Einheit
	Klappliste Untersetzungsfaktor	Auswahl der Einheit des Untersetzungsfaktors Folgende Einstellungen sind möglich: • "Zyklus" • "s" (Einstellmöglichkeit hängt von der gewählten Aufzeichnungsebene unter "Aufzeichnungszeitpunkt" ab)
	Textfeld Abtastzeit	Anzeige der Abtastzeit unter Berücksichtigung der konfigurierten Untersetzung und der gewählten Einheit (nur bei äquidistanten OBs)
"Max. Aufzeichnungsdauer"		
	Textfeld Max. Aufzeichnungsdauer	Anzeige der berechneten maximalen Aufzeichnungsdauer Die "Max. Aufzeichnungsdauer" ist davon abhängig, wie viele Signale aufgezeichnet werden und welchen Datentyp diese Signale haben.
	"Max. Aufzeichnungsdauer verwenden"	Setzen der Aufzeichnungsdauer auf die maximale Aufzeichnungsdauer Mit Aktivierung des Optionskästchens wird die Aufzeichnungsdauer auf die maximal mögliche Aufzeichnungsdauer gesetzt. Die eingestellte Untersetzung im Eingabefeld "Aufzeichnen alle" wird berücksichtigt. Die Aufzeichnungsdauer wird auch angepasst, wenn weitere Signale hinzugefügt werden. Weitere Informationen dazu finden Sie auch als FAQ unter der Beitrags-ID 102781176 (http://support.automation.siemens.com/WW/view/de/102781176).
"Aufzeichnungsdauer"		
	Eingabefeld Aufzeichnungsdauer	Eingabe der Aufzeichnungsdauer bezogen auf die ausgewählte Einheit Wenn das Optionskästchen "Aufzeichnungsdauer = Max. Aufzeichnungsdauer" aktiviert ist, werden Eingaben durch den in "Max. Aufzeichnungsdauer" angezeigten Wert überschrieben.

Einstellung/Anzeige		Beschreibung
	Klappliste Einheit	Auswahl der Einheit für die Aufzeichnungsdauer Folgende Einstellungen sind möglich: • "Messpunkte" Es werden maximal die unter Aufzeichnungsdauer parametrierten Messpunkte aufgezeichnet. • "s" (Einstellmöglichkeit hängt von der gewählten Aufzeichnungsebene unter "Aufzeichnungszeitpunkt" ab)
	Textfeld Berechnete Aufzeichnungsdauer	Anzeige der berechneten Aufzeichnungsdauer (nur bei äquidistanten OBs)

Das folgende Bild zeigt exemplarisch die Einstellungen für den Trigger im TIA-Portal:

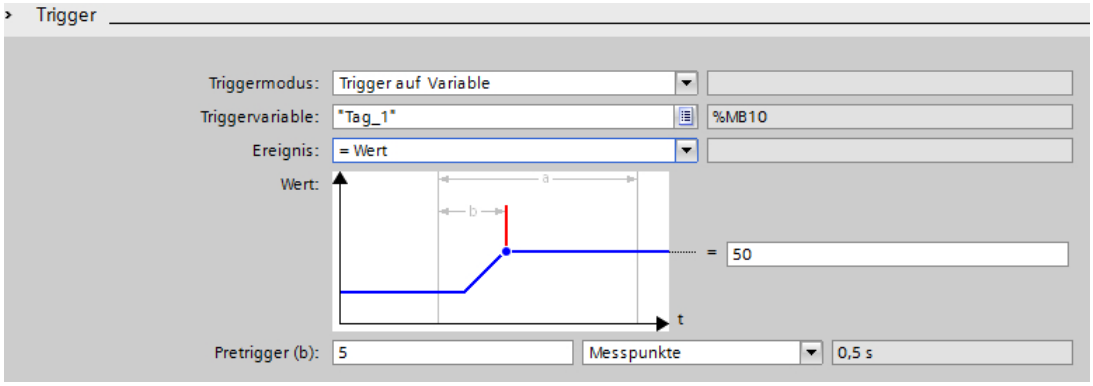

Die folgende Tabelle erklärt die Einstellungen und Anzeigen:

Einstellung/Anzeige		Beschreibung
"Triggermodus"		Auswahl des Triggermodus
	Klappliste Triggermodus	Folgende Einstellungen sind möglich: • "Aufzeichnung sofort" Die Aufzeichnung erfolgt sofort nach der Aktivierung im Gerät. • "Trigger auf Variable" Die Aufzeichnung erfolgt, sobald der Trace im Gerät aktiviert ist und die konfigurierte Bedingung des Triggers erfüllt ist.
	Textfeld	-
"Triggervariable"		Mit der "Triggervariable" legen Sie ein Signal fest, das zum Triggern der Aufzeichnung verwendet wird.
	Eingabefeld Triggervariable	Eingabe eines Signals **Beispiele:** • "Datenbaustein_1".Temperatur • M0.0 • DB1.DBW3 Siehe auch Datentypen für die Triggervariable (Seite 120).

Einstellung/Anzeige		Beschreibung
		Öffnen der Signalauswahl-Tabelle Mit Klick auf das Symbol öffnen Sie eine Tabelle, welche mögliche Signale zur Auswahl als Triggervariable anbietet. Das ausgewählte Signal wird im Eingabefeld angezeigt.
	Textfeld Adresse Triggervariable	Anzeige der Adresse der Triggervariable Bei rein symbolischen Signalen bleibt das Feld leer.
"Ereignis"		Entsprechend dem Datentyp der Triggervariable werden die auf diese Triggervariable anwendbaren Ereignisse zur Auswahl angeboten. Die Konfiguration des Ereignisses ist nur möglich, wenn ein gültiges Signal als Triggervariable eingetragen ist.
	Klappliste Triggerereignisse	Ereignisauswahl, auf das die Triggervariable geprüft wird Die Einträge der Klappliste sind im Kapitel Triggerereignis (Seite 120)beschrieben.
	Textfeld	-
"Wert"		Konfiguration des gewählten Ereignisses Die Konfigurationsmöglichkeiten unterscheiden sich in Abhängigkeit vom Format der Triggervariable und dem gewählten Ereignis. Siehe Triggerereignis (Seite 120).
"Pretrigger"		Mit dem "Pretrigger" definieren Sie die Anzahl der Messpunkte, die bereits vor der Erfüllung der eigentlichen Triggerbedingung aufgezeichnet werden. Tritt das Triggerereignis sofort oder kurz nach dem Aktivieren der Aufzeichnung ein, kann sich eine kürzerer Aufzeichnungsdauer ergeben. Beispiele für "Aufzeichnungsdauer (a)" = 20 Messpunkte und "Pretrigger (b)" = 5 Messpunkte: • Fall 1: Triggerereignis tritt 50 Messpunkte nach Aktivierung der Aufzeichnung ein Tatsächliche Aufzeichnungsdauer (a) = 20 Messpunkte • Fall 2: Triggerereignis tritt 2 Messpunkte nach Aktivierung der Aufzeichnung ein Tatsächliche Aufzeichnungsdauer (a) = 17 Messpunkte
	Eingabefeld Dauer	Eingabe der Dauer bezogen auf die Auswahl in der Klappliste.
	Klappliste Einheit	Auswahl der Einheit Folgende Einstellungen sind möglich: • "Messpunkte" • "s" (Einstellmöglichkeit hängt von der gewählten Aufzeichnungsebene unter "Aufzeichnungszeitpunkt" ab)
	Textfeld Resultierende Pretrigger-Dauer	Anzeige der berechneten "Pretrigger"-Dauer (nur bei Aufzeichnung in äquidistanten OBs)

Das folgende Bild zeigt exemplarisch die Einstellungen für das Speichern von Messungen im Gerät:

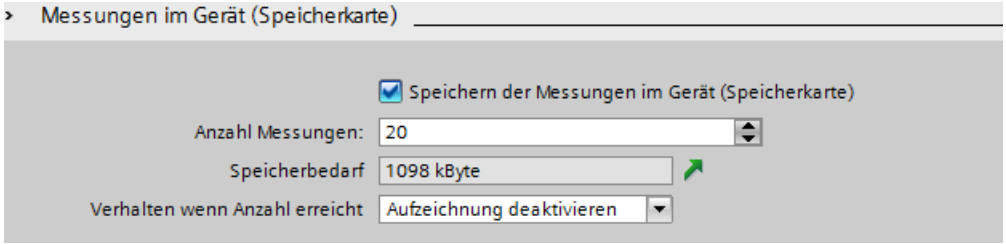

Hinweis

Verfügbarer Speicher im Gerät (Speicherkarte)

Der Speicher im Gerät (Speicherkarte) wird zum Teil von systemrelevanten Funktionen benutzt oder dafür reserviert.

Für das Speichern von Messungen kann daher nicht der gesamte Speicher genutzt werden.

Weitere Angaben entnehmen Sie dem Funktionshandbuch Struktur und Verwendung des CPU-Speichers (https://support.industry.siemens.com/cs/de/de/view/59193101).

Hinweis

Speicherbedarf bei Neustart

Nach einem Neustart des Geräts werden maximal so viele Messungen im Gerät gespeichert, wie unter "Anzahl Messungen" parametriert ist.

Beachten Sie bei wiederholtem Neustart, dass bereits gespeicherte Messungen nicht überschrieben werden und wieder erneut die parametrierte "Anzahl Messungen" im Gerät gespeichert werden.

Die folgende Tabelle erklärt die Einstellungen und Anzeigen:

Einstellung/Anzeige	Beschreibung
"Speichern der Messungen im Gerät (Speicherkarte)"	Messung automatisch wiederholen und remanent im Gerät ablegen Diese Einstellung ist nur bei Triggermodus "Trigger auf Variable" möglich. Die Messungen werden auf der "Primary" Speicherkarte abgelegt. Für Traces, bei denen das Speichern im Gerät (Speicherkarte) aktiviert ist, steht die Funktion zum automatischen Wiederholen der Aufzeichnung nicht zur Verfügung. **Hinweis** Es werden nur abgeschlossene Messungen im Gerät gespeichert. Eine vom Anwender deaktivierte Aufzeichnung wird nicht im Gerät gespeichert. **Diese Funktion steht mit folgenden Firmwareständen zur Verfügung:** • S7-1200 ab V4.2 • S7-1500 ab V2.0 Die Funktion wird von der CPU S7-1500 R/H nicht unterstützt.
"Anzahl Messungen"	Eingabe der Anzahl Messungen, die auf der Karte gespeichert werden sollen
"Speicherbedarf"	Anzeige des voraussichtlichen Speicherbedarfs für alle Messungen

Einstellung/Anzeige	Beschreibung
	Speicherauslastung anzeigen Zeigt das Register mit der Speicherauslastung an.
"Verhalten wenn Anzahl erreicht"	Auswahl des Verhaltens nach erreichter "Anzahl Messungen" Folgende Einstellungen sind möglich: • "Aufzeichnung deaktivieren" Die Messungen werden so lange wiederholt, bis die parametrierte "Anzahl Messungen auf der Karte" erreicht sind. • "Älteste Aufzeichnung überschreiben" Der Messungen werden in einem Ringpuffer gespeichert und so lange wiederholt, bis der Anwender die Aufzeichnung deaktiviert. Sobald die Anzahl Messungen die parametrierte "Anzahl Messungen auf der Karte" überschreitet, wird jeweils die älteste Messung auf der Karte überschrieben. **Hinweis** Beachten Sie, dass häufig wiederholte Schreibvorgänge die Karte beschädigen können.

Datentypen für die Triggervariable

Die folgende Tabelle zeigt die unterstützten Datentypen für die Triggervariable:

Speicherbedarf und Format der Zahl	Datentyp
1-Byte	BOOL
8-Bit-Ganzzahlen	SINT, USINT, BYTE
16-Bit-Ganzzahlen	INT, UINT, WORD, DATE
32-Bit-Ganzzahlen	DINT, UDINT, DWORD, TIME, TOD
64-Bit-Ganzzahlen [1)]	LINT, ULINT, LWORD, LTIME, LTOD, LDT
32-Bit-Gleitpunktzahlen	REAL
64-Bit-Gleitpunktzahlen	LREAL

1) Wird nicht von S7-1200 unterstützt.

Triggerereignis

In Abhängigkeit von der Auswahl in der Klappliste unterscheiden sich die weiteren Einstellungen zu "Ereignis".

Nachfolgend sind die einzelnen Ereignisse beschrieben.

"=TRUE"

Unterstützte Datentypen: Bit (Seite 120)

Der Start der Aufzeichnung erfolgt bei dem Zustand TRUE des Triggers.

"=FALSE"

Unterstützte Datentypen: Bit (Seite 120)

Der Start der Aufzeichnung erfolgt bei dem Zustand FALSE des Triggers.

"Steigende Flanke"

Unterstützte Datentypen: Bit (Seite 120)

Der Start der Aufzeichnung erfolgt, wenn der Trigger den Zustand von FALSE auf TRUE wechselt.
Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

"Steigendes Signal"

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 120) (keine Zeiten, Datum und Uhrzeit)

Der Start der Aufzeichnung erfolgt, wenn der aufsteigende Wert des Triggers den zu diesem Ereignis konfigurierten Wert erreicht oder überschreitet.
Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

"Fallende Flanke"

Unterstützte Datentypen: Bit (Seite 120)

Der Start der Aufzeichnung erfolgt, wenn der Trigger den Zustand von TRUE auf FALSE wechselt.
Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

"Fallendes Signal"

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 120) (keine Zeiten, Datum und Uhrzeit)

Der Start der Aufzeichnung erfolgt, wenn der absteigende Wert des Triggers den zu diesem Ereignis konfigurierten Wert erreicht oder unterschreitet.
Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

"Im Bereich"

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 120)

Der Start der Aufzeichnung erfolgt, sobald sich der Wert des Triggers in dem zu diesem Ereignis konfigurierten Wertebereich befindet.

"Außerhalb des Bereichs"

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 120)

Der Start der Aufzeichnung erfolgt, sobald sich der Wert des Triggers außerhalb des zu diesem Ereignis konfigurierten Wertebereichs befindet.

"Wertänderung"

Alle Datentypen werden unterstützt.

Der Wert wird ab Aktivierung der Aufzeichnung auf Änderung geprüft. Der Start der Aufzeichnung erfolgt, wenn sich der Wert des Triggers ändert.

Dieses Triggerereignis wird ab V13 SP1 unterstützt. Ältere Versionen des TIA-Portals können den Trigger nicht interpretieren. Berücksichtigen Sie, dass in diesem Fall kein expliziter Hinweis ausgegeben wird. Das kann z. B. vorkommen, wenn der Trace aus einer CPU in ein TIA-Portal kleiner V13 SP1 übertragen wird oder eine Tracekonfiguration importiert wird.

"= Wert"

Unterstützte Datentypen: Ganzzahlen (Seite 120)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers gleich dem zu diesem Ereignis konfigurierten Wert ist.

"<> Wert"

Unterstützte Datentypen: Ganzzahlen (Seite 120)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers ungleich dem zu diesem Ereignis konfigurierten Wert ist.

"= Bitmuster"

Unterstützte Datentypen: Ganzzahlen (Seite 120) (keine Zeiten, Datum und Uhrzeit)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers mit dem zu diesem Ereignis konfigurierten Bitmuster übereinstimmt.

Das folgende Bild zeigt die Einstellmöglichkeiten für ein "Bitmuster":

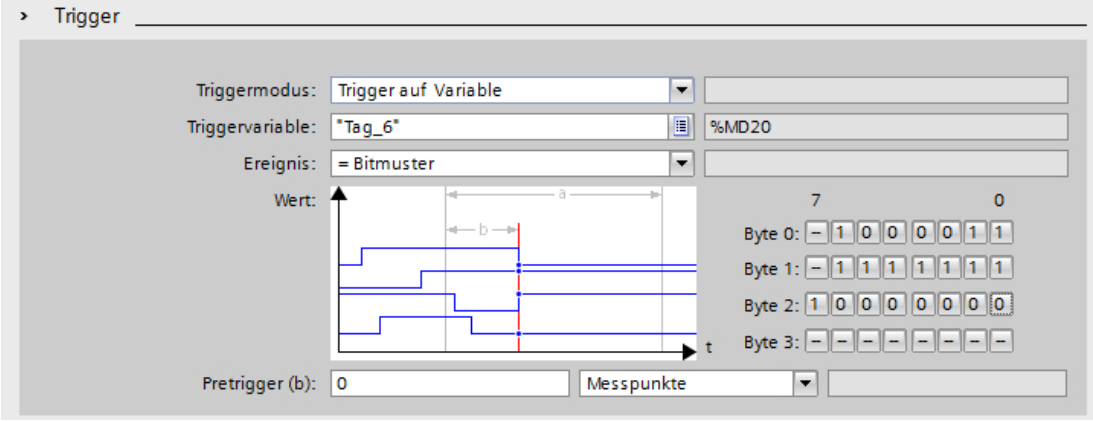

Durch Klick auf die jeweilige Schaltfläche kann zwischen den Symbolen umgeschaltet werden.

Die folgende Tabelle zeigt die Symbole:

Symbol	Beschreibung
–	Bit wird nicht ausgewertet
0	Bit wird auf FALSE geprüft
1	Bit wird auf TRUE geprüft

"<> Bitmuster"

Unterstützte Datentypen: Ganzzahlen (Seite 120) (keine Zeiten, Datum und Uhrzeit)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers nicht mit dem zu diesem Ereignis konfigurierten Bitmuster übereinstimmt.

Siehe auch

Triggerbedingungen konfigurieren (Seite 125)

Erfassbare Variablen (Seite 111)

6.5.1.7 Konfigurieren

Trace konfigurieren - Übersicht

Die Konfiguration der Aufzeichnungsbedingungen und aufzuzeichnenden Signale ist gerätespezifisch.

Voraussetzung

Eine Tracekonfiguration ist angelegt und im Arbeitsbereich im Register "Konfiguration" geöffnet.

Vorgehen

Die folgende Tabelle zeigt die Vorgehensweise zum Konfigurieren.

Schritt	Beschreibung
1	Dokumentation der Konfiguration (optional) Geben Sie im Inspektorfenster einen Kommentar und einen Autor zu der Konfiguration ein.
2	Signale auswählen (Seite 124) Wählen Sie im Bereich "Signale" die aufzuzeichnenden Signale aus.
3	AUTOHOTSPOT Wählen Sie im Bereich "Aufzeichnungsbedingungen" einen Aufzeichnungszeitpunkt, einen Takt und die Dauer aus.
4	Triggerbedingungen konfigurieren (Seite 125) Legen Sie im Bereich "Aufzeichnungsbedingungen" fest, ob die Aufzeichnung sofort oder in Abhängigkeit von einer Triggerbedingung erfolgt.
5	Messung im Gerät (Speicherkarte) konfigurieren (Seite 126) Legen Sie im Bereich "Aufzeichnungsbedingungen" fest, ob die Messung im Gerät (Speicherkarte) gespeichert werden soll.

Signale auswählen

Voraussetzung

- Eine Tracekonfiguration ist angelegt und geöffnet.
- Der Bereich "Signale" ist im Register "Konfiguration" geöffnet.

Vorgehen

Um die aufzuzeichnenden Signale zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie ein Signal aus. Folgende Möglichkeiten stehen zur Verfügung:
   – Klicken Sie in der Spalte "Name" auf die Schaltfläche und wählen Sie eine Variable aus.
   – Geben Sie in der Spalte "Name" die symbolische Bezeichnung der Variable in die Zelle ein.
   – Geben Sie in der Spalte "Adresse" direkt die Adresse ein.
   – Ziehen Sie ein Signal per Drag & Drop in die Tabelle.
2. Klicken Sie in die Spalte "Farbe" und wählen Sie eine Farbe für die Darstellung des Signals aus.
3. Klicken Sie in die Spalte "Kommentar" und geben Sie einen Kommentar zu dem Signal ein.
4. Wiederholen Sie das Vorgehen ab Schritt 1, bis alle aufzuzeichnenden Signale in der Tabelle eingetragen sind.

Aufzeichnungstakt und Dauer konfigurieren

Voraussetzung

- Eine Tracekonfiguration ist angelegt und geöffnet.
- Der Bereich "Aufzeichnungsbedingungen" ist im Register "Konfiguration" geöffnet.

Vorgehen

Um den Takt und die Dauer einer Aufzeichnung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche für den Aufzeichnungszeitpunkt.
2. Wählen Sie einen OB für den Aufzeichnungszeitpunkt (Seite 112) aus.
3. Wählen Sie in der Klappliste für "Aufzeichnen alle" eine Einheit für den Untersetzungsfaktor aus.
4. Geben Sie in das Eingabefeld für "Aufzeichnen alle" den Faktor für die Untersetzung ein.
5. Wählen Sie in der Klappliste für "Aufzeichnungsdauer" eine Einheit aus.
6. Legen Sie die Aufzeichnungsdauer fest.
   Folgende Möglichkeiten stehen zur Verfügung:
   - Geben Sie in das Eingabefeld für "Aufzeichnungsdauer" einen Wert für die Dauer ein.
   - Aktivieren Sie das Optionskästchen "Max. Aufzeichnungsdauer verwenden"

Triggerbedingungen konfigurieren

Voraussetzung

- Eine Tracekonfiguration ist angelegt und geöffnet.
- Der Bereich "Aufzeichnungsbedingungen" ist im Register "Konfiguration" geöffnet.

Triggerbedingung "Sofort aufzeichnen"

Um die Aufzeichnung sofort zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste für "Triggermodus" den Eintrag "Sofort aufzeichnen" aus.
   Die Eingabefelder für die Triggervariable werden ausgeblendet.

Triggerbedingung "Trigger auf Variable"

Um die Aufzeichnung in Abhängigkeit einer Bedingung zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste für "Triggermodus" den Eintrag "Trigger auf Variable" aus.
2. Wählen Sie eine Triggervariable aus. Folgende Möglichkeiten stehen zur Verfügung:
   – Klicken Sie auf die Schaltfläche für die Triggervariable und wählen Sie eine Variable aus.
   – Geben Sie direkt die Adresse oder die symbolische Bezeichnung der Variable in das Eingabefeld für die Triggervariable ein.

   Eine Klappliste mit Ereignissen und Eingabefeldern wird angezeigt. Die Anzeige ist abhängig vom Datentyp der Variable.
3. Konfigurieren Sie das Ereignis
4. Wählen Sie in der Klappliste für "Pretrigger" eine Einheit für den Pretrigger aus.
5. Um bereits eine Zeitspanne vor dem Triggerereignis aufzuzeichnen, geben Sie in das Eingabefeld für den Pretrigger einen Wert größer 0 ein.

Hinweis

Zyklische Prüfung der Triggerbedingung

Die Triggerbedingung wird unabhängig von der Einstellung in "Aufzeichnen alle" in jedem Zyklus überprüft. Zur sicheren Erkennung des Triggers muss das Triggersignal mindestens einen vollen Zyklus lang anstehen.

Messung im Gerät (Speicherkarte) konfigurieren

Voraussetzung

- Eine Tracekonfiguration ist angelegt und geöffnet.
- Der Bereich "Aufzeichnungsbedingungen" ist im Register "Konfiguration" geöffnet.
- Der Triggermodus "Trigger auf Variable" ist eingestellt.
- Die Firmware des Geräts unterstützt die Aufzeichnung einer Messung im Gerät.

Vorgehen

Um die Messung im Gerät (auf der Speicherkarte) zu speichern, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen "Speichern der Messungen im Gerät (Speicherkarte)".
2. Geben Sie in das Eingabefeld "Anzahl Messungen" die Anzahl der Messungen ein, die auf der Karte gespeichert werden sollen.
3. Stellen Sie in der Klappliste "Verhalten wenn Anzahl erreicht" das gewünschte Verhalten nach erreichter "Anzahl Messungen" ein.

Hinweis

Keine Auswertung des Triggers während der Speicherung

Solange die Aufzeichnung gespeichert wird, kann kein neuer Trigger ausgewertet werden.

Fernverbindung mit TeleService aufbauen 7

7.1 Grundlagen für das Arbeiten mit TeleService

7.1.1 Einführung in TeleService

Einführung

Mit TeleService wird Ihre Steuerung telekommunikationsfähig. Sie können dezentrale Anlagen über Fernverbindungen zentral verwalten, steuern und überwachen.

Funktionsumfang

Mit TeleService können Sie den Funktionsumfang des TIA Portals über ein Telefonnetz oder eine Internetverbindung nutzen, indem Sie eine Fernverbindung zu einer entfernten Anlage aufbauen. Über die Online-Verbindung ist es möglich, auch eine entfernte Anlage wie gewohnt mit dem TIA Portal zu bearbeiten.

Vorteile

Durch den Einsatz von TeleService ergeben sich für Sie folgende Vorteile:

- Sie können auch entfernte Anlagenteile gut erreichen und in ein Gesamtsystem einbeziehen.
- Bei Fehlern in einer entfernten Anlage können Sie schnelle Hilfe und Unterstützung bieten, ohne selbst vor Ort sein zu müssen.
- Sie können Ihre Ressourcen effektiv einsetzen.
- Kosten werden beträchtlich reduziert.
- Stillstandszeiten von Anlagen können wesentlich verringert werden.
- Die Wirtschaftlichkeit Ihrer Anlage steigt.

Siehe auch

Funktionalität von TeleService (Seite 130)

7.1.2 Funktionalität von TeleService

Einsatzmöglichkeiten von TeleService

TeleService bietet Ihnen folgende Einsatzmöglichkeiten:

- **Zugriff auf entfernte Anlagen (Fernwartung):**
  Sie können dezentrale Anlagen über Fernverbindungen zentral verwalten, steuern und überwachen.
  Dies ist möglich mit einer CPU S7-300/400, einer CPU S7-1200 und einer CPU S7-1500 und jeweils einem TS Adapter MPI oder einem TS Adapter IE.
- **Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung):**
  Mit Hilfe von PRODAVE MPI V5.0 und höher können Sie eine Fernverbindung zu einer entfernten Anlage aufbauen und mit der Kommunikationsanweisung "PG_DIAL" können Sie eine Fernverbindung von einer entfernten Anlage aufbauen.
  Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Datenaustausch zwischen Anlagen (AS-AS-Fernkopplung):**
  Mithilfe der Kommunikationsanweisung "AS_DIAL" können zwei Automatisierungssysteme über das Telefonnetz Prozessdaten miteinander austauschen.
  Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Senden einer SMS von einer Anlage:**
  Mithilfe der Kommunikationsanweisung "SMS_SEND" kann ein Automatisierungssystem über ein GSM-Funkmodem eine Nachricht (SMS) versenden.
  Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Senden einer E-Mail von einer Anlage**
  Mithilfe der folgenden Kommunikationsanweisungen und einem TS Adapter IE kann ein Automatisierungssystem auch eine E-Mail verschicken.
  - S7-300/400er CPUs (CPU S7-31x-2PN/DP bzw. CPU 41x-3PN/D) verwenden die Anweisung "AS_MAIL"
  - S7-1200er CPUs verwenden die Anweisung "TM_MAIL"
  - S7-1500er CPUs verwenden die Anweisung "TMAIL_C"

7.1.3 Telefonbuch bei TeleService

Einführung

Durch einen Doppelklick auf den Ordner "Telefonbuch" in der Projektnavigation öffnen Sie den Telefonbucheditor, der das Telefonbuch von TeleService anzeigt.

Jede Version des TIA Portals verfügt über ein eigenes "globales Telefonbuch". Wird in einer neuen Version des TIA Portals ein globales Telefonbuch aus einer älteren Version des TIA Portals gefunden, so werden sie einmalig gefragt, ob Sie dieses Telefonbuch importieren möchten.

Dies hat den Vorteil, dass Ihnen auch in der neuen Version des TIA Portals die Anlagendaten aus der vorherigen Version zur Verfügung stehen.

Eigenschaften des globalen Telefonbuchs

Das globale Telefonbuch dient in TeleService zum Verwalten der spezifischen Anlagedaten, die zum Aufbauen einer Fernverbindung benötigt werden.

Wenn Sie das Telefonbuch zum ersten Mal öffnen, wird ein leeres Telefonbuch mit allen vorhandenen Spalten angezeigt, ansonsten wird das zuletzt bearbeitete Telefonbuch angezeigt.

In einem Telefonbuch können Sie eine beliebige Anzahl von Anlagen eintragen. Anlagen enthalten die Daten, die Sie für den Aufbau einer Fernverbindung benötigen, z. B. den Namen und den Standort des Teilnehmers sowie die anzuwählende Telefonnummer mit den landesspezifischen Details. Bei VPN-Verbindungen können Sie an Stelle der Telefonnummer eine IP-Adresse bzw. einen DNS-Namen eintragen.

Die für den Verbindungsaufbau verwendeten TS Adapter sind farblich unterschiedlich dargestellt, abhängig davon, ob ein TS Adapter MPI oder ein TS Adapter IE für den Verbindungsaufbau eingesetzt wird.

7.2 Arbeiten mit dem Telefonbuch

7.2.1 Grundlagen zum Arbeiten mit dem Telefonbuch

Arbeiten mit dem Telefonbuch

Folgende Möglichkeiten stehen Ihnen für das Arbeiten mit einem Telefonbuch zur Verfügung:

- Telefonbuch öffnen
- Telefonbuch speichern
- Telefonbuchdaten importieren
- Telefonbuchdaten exportieren
- Telefonbuchdaten drucken
- Telefonbuchdaten zum Aufbau einer Fernverbindung verwenden

Diese Funktionen können Sie komfortabel über die in der Funktionsleiste des Telefonbuchs angezeigten Schaltflächen durchführen.

Hinweis

Zugriff auf Telefonbücher

Das Telefonbuch wird benutzerspezifisch in TeleService hinterlegt. Es ist jedoch nicht möglich, gleichzeitig mit mehr als einer Instanz des TIA Portals auf das globale Telefonbuch zuzugreifen.

Siehe auch

Telefonbuch öffnen (Seite 134)

Telefonbuch speichern (Seite 135)

Telefonbuchdaten exportieren (Seite 137)

Telefonbuch drucken (Seite 138)

Aufbau des Telefonbuchs (Seite 133)

7.2.2 Aufbau des Telefonbuchs

Einführung

In TeleService dient ein globales Telefonbuch zum Verwalten der Daten, die Sie für den Aufbau einer Fernverbindung benötigen. Wenn Sie die Verbindungsdaten einmalig erstellt und im Telefonbuch hinterlegt haben, können Sie immer wieder darauf zugreifen, um eine Fernverbindung aufzubauen.

Aufbau des Telefonbuches

Das in TeleService integrierte globale Telefonbuch enthält folgende Spalten:

Spaltenname	Erläuterung
Anlagenname	Tragen Sie hier den gewünschten Namen für Ihre Anlage ein.
Adapter-Typ	Wählen Sie aus der Klappliste den verwendeten Adapter-Typ aus: TS Adapter MPI oder TS Adapter IE.
Verbindungs-Typ	Wählen Sie den gewünschten Verbindungs-Typ aus: Wählverbindung oder VPN-Verbindung.
Ortskennzahl	Tragen Sie hier die gewünschte Ortskennzahl ein. Diese Spalte ist nur aktiv bei Wählverbindungen. Bei VPN-Verbindungen ist diese Spalte nicht editierbar.
Telefonnummer/Remote-Adresse	Tragen Sie hier die gewünschten Verbindungsdaten für den Aufbau der Fernverbindung ein. Dies kann für Wählverbindungen eine Telefonnummer sein und für VPN-Verbindungen ein DNS-Name oder eine IP-Adresse.
Fingerabdruck	Tragen Sie hier den zugehörigen Fingerabdruck für den Aufbau einer VPN-Verbindung zum TS Adapter IE Advanced ein.
Land	Tragen Sie die zugehörige Landeskennzahl ein. Diese Spalte ist nur aktiv bei Wählverbindungen. Bei VPN-Verbindungen ist diese Spalte nicht editierbar.
Benutzername	Tragen Sie den Benutzernamen ein, unter dem Sie angemeldet sind.
Kennwort	Tragen Sie hier das zum Benutzernamen gehörige Kennwort ein.
Gruppe	Tragen Sie hier die zugehörige Gruppe ein, falls Sie eine Gruppierung vorgenommen haben.
Firma	Tragen Sie die Firma ein, die kontaktiert werden soll.
Abteilung	Tragen Sie die zugehörige Abteilung ein.
Straße	Tragen Sie die Straße ein.
Ort	Tragen Sie den Ort ein, zu dem die Fernverbindung aufgebaut werden soll.
Kommentar	Tragen Sie hier, falls gewünscht, einen Kommentar ein.

Spalten ein- oder ausblenden

Einzelne Spalten können Sie ein- oder ausblenden. Selektieren Sie dazu die gewünschte Spaltenüberschrift und rufen Sie über die rechte Maustaste das Kontextmenü auf.

7.2.3 Symbole im Telefonbuch

Bedeutung der Symbole in TeleService

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole in TeleService:

Symbol	Bedeutung
	Öffnet das globale Telefonbuch
	Importiert ein Telefonbuch
	Exportiert ein Telefonbuch
	Baut eine Fernverbindung auf
	Trennt die aktive Fernverbindung
	Baut eine Fernverbindung auf oder trennt diese
	Zeigt im Telefonbuch die Verbindung zu einem TS Adapter IE an
	Zeigt im Telefonbuch die Verbindung zu einem TS Adapter MPI an
	Fügt im Telefonbuch eine neue Zeile hinzu
	Fügt im Telefonbuch eine neue Zeile ein

7.2.4 Telefonbuch verwalten

7.2.4.1 Telefonbuch öffnen

Öffnen von Telefonbüchern

Um das Telefonbuch zu öffnen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter "Online Zugänge" > "TeleService" auf den Ordner "Telefonbuch".
2. Das Telefonbuch wird geöffnet, sodass Sie die gewünschten Anlagedaten eintragen bzw. nachbearbeiten können.

7.2.4.2 Zeilen ins Telefonbuch einfügen

Zeilen ins Telefonbuch einfügen

Um eine neue Zeile ins Telefonbuch einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie eine neue Zeile einfügen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Zeile einfügen".

Ergebnis

Vor der markierten Zeile wird eine neue Zeile ins Telefonbuch eingefügt.

7.2.4.3 Spalten im Telefonbuch ein- und ausblenden

Ein und Ausblenden von Spalten

Um Spalten im Telefonbuch ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Spalten ein-/ausblenden".
   Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

Ergebnis

Die jeweiligen Spalten werden bei der Anzeige des Telefonbuchs ein- oder ausgeblendet.

7.2.4.4 Telefonbuch speichern

Speichern von Telefonbüchern

Beim Verlassen des Telefonbuch-Editors oder beim Verlassen des TIA Portals werden Sie gefragt, ob Sie das globale Telefonbuch speichern wollen.

Zum Speichern des Telefonbuchs klicken Sie auf "Ja".

7.2.4.5 Telefonbuchdaten importieren

Einführung

Es besteht die Möglichkeit, die Telefonbuchdaten von einer externen Datei oder aus einer älteren Version des TIA Portals zu importieren.

Voraussetzung

Sie haben bereits eine importfähige Telefonbuchdatei angelegt.

Sie haben mit einer älteren Version des TIA Portals bereits ein Telefonbuch angelegt.

Telefonbuchdaten aus Telefonbuchdatei importieren

Um Telefonbuchdaten aus einer Telefonbuchdatei zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" unter "Online-Zugänge" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Importieren".
4. Bestätigen Sie die Abfrage, ob der aktuelle Stand des Arbeitstelefonbuchs gespeichert werden soll, ggf. mit "Ja" und wählen Sie im nachfolgenden Dialog aus, wohin das Telefonbuch gespeichert werden soll.
5. Falls Sie den aktuellen Stand des Arbeitstelefonbuchs nicht speichern wollen, so wählen Sie bei der Abfrage "nein". Wählen Sie im nachfolgenden Dialog aus, welche Telefonbuchdatei in das aktuelle Telefonbuch gespeichert werden soll.
6. Verlassen Sie den Dialog mit "OK".

Ergebnis

Die importierten Telefonbuchdaten werden im globalen Telefonbuch angezeigt.

Hinweis

Wählregeln definieren

Damit die Spalten "Ortskennzahl" und "Land" im importierten Telefonbuch für Wählverbindungen editierbar sind, müssen Wählregeln definiert sein.

Um Wählregeln zu definieren, folgen Sie dem im Verlauf angezeigten Link.

Telefonbuchdaten aus älterer Version des TIA Portals importieren

Um Telefonbuchdaten aus einer älteren Version des TIA Portals zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" unter "Online-Zugänge" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Importieren".
4. Bestätigen Sie die Abfrage, ob der aktuelle Stand des Arbeitstelefonbuchs gespeichert werden soll, ggf. mit "Ja" und wählen Sie im nachfolgenden Dialog aus, wohin das Telefonbuch gespeichert werden soll.
5. Falls Sie den aktuellen Stand des Arbeitstelefonbuchs nicht speichern wollen, so wählen Sie bei der Abfrage "nein".
6. Tragen Sie im nachfolgenden Dialog folgende Pfadangabe ein: "%appdata%\siemens \automation\TeleService\GlobalTeleServicePhoneBook.tel".
7. Verlassen Sie den Dialog mit "OK".

Ergebnis

Die aus der älteren Version des TIA Portals importierten Telefonbuchdaten werden im globalen Telefonbuch angezeigt.

Hinweis

Wählregeln definieren

Damit die Spalten "Ortskennzahl" und "Land" im importierten Telefonbuch für Wählverbindungen editierbar sind, müssen Wählregeln definiert sein.

Um Wählregeln zu definieren, folgen Sie dem im Verlauf angezeigten Link.

Siehe auch

Wählregeln definieren (Seite 138)

7.2.4.6 Telefonbuchdaten exportieren

Einführung

Es besteht die Möglichkeit, Telefonbuchdaten in eine externe Datei zu exportieren.

Voraussetzung

Sie haben unter TeleService bereits ein Telefonbuch mit den entsprechenden Anlagendaten erstellt.

Vorgehen

Um die Telefonbuchdaten zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Exportieren".
4. Wählen Sie im nachfolgenden Dialog aus, wohin das aktuelle Telefonbuch exportiert werden soll.
5. Verlassen Sie den Dialog mit "OK".

Ergebnis

Die exportierten Telefonbuchdaten werden in der angegebenen Exportdatei gespeichert.

7.2.4.7 Telefonbuch drucken

Drucken von Telefonbüchern

Sie können sich alle Daten eines Telefonbuchs oder nur einen bestimmten Teil eines Telefonbuchs ausdrucken lassen.

Gehen Sie folgendermaßen vor:

1. Öffnen Sie das Telefonbuch.
2. Wählen Sie den Befehl **Projekt > Drucken** oder klicken Sie auf die entsprechende Schaltfläche in der Funktionsleiste. Daraufhin öffnet sich das Dialogfeld "Drucken".
3. Geben Sie hier an, ob Sie das gesamte Telefonbuch oder nur einen Teil des Telefonbuchs drucken möchten, und stellen Sie alle weiteren Optionen ein.
4. Starten Sie den Druckvorgang mit "OK".

Ergebnis

Die Daten des Telefonbuchs werden auf dem voreingestellten Drucker ausgegeben. Umfasst der Ausdruck mehrere Seiten, wird am rechten unteren Blattrand nach der Seitennummer eine Folgeseitenkennung ausgegeben. Die letzte Seite des Ausdrucks enthält keine Folgeseitenkennung.

7.2.4.8 Wählregeln definieren

Vorgehensweise

Um standortspezifische Wählregeln zu definieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im TIA Portal den Ordner "Online-Zugänge" und selektieren Sie den Ordner "TeleService".
2. Rufen Sie über das Kontextmenü die Eigenschaften von "TeleService" auf.
3. Öffnen Sie im nachfolgenden Dialog das Register "Erweiterte Einstellungen" und aktivieren Sie die Option "Wählregeln benutzen".
4. Klicken Sie auf die Schaltfläche "Anpassen".
5. Tragen Sie im nachfolgenden Dialog die gewünschten Standortinformationen für den jeweiligen Verbindungsaufbau ein.
   - Geben Sie bei neuen Standorten das Land bzw. die Region und die zugehörige Ortskennzahl an.
   - Stellen Sie, falls erforderlich, die richtige Netzkennzahl und Amtskennziffer für Ortsgespräche/Ferngespräche ein.
   - Selektieren Sie das gewünschte Wählverfahren für den Standort.

6. Verlassen Sie diesen Dialog, indem Sie auf "OK" klicken.
7. Wählen Sie im nachfolgenden Dialog den Standort aus, von dem aus gewählt werden soll, und klicken Sie auf "OK".
8. Überprüfen Sie im nachfolgenden TeleService-Dialog, dass unter "Standort" der gewünschte Standort eingetragen ist.
9. Bestätigen Sie Ihre Eingaben, indem Sie auf "OK" klicken.

Hinweis

Modembetrieb an Amtsleitung oder Nebenstelle

Wenn Sie Ihr Modem an einer Amtsleitung (Hauptanschluss) betreiben, brauchen Sie keine Amtskennziffer anzugeben. Die Felder der Amtskennziffer für Ortsgespräche und für Ferngespräche müssen leer sein.
Wenn Sie Ihr Modem an einer Nebenstelle betreiben, sollten Sie die Amtskennziffer eintragen, die gewählt werden muss, um eine Amtsleitung zu erhalten.

Beispiel für die Verwendung von Wählregeln

Das nachfolgende Beispiel zeigt die Verwendung von standortspezifischen Wählregeln für die Stadt Karlsruhe in Deutschland.

- Sie tragen im Telefonbuch von TeleService für eine Anlage die Rufnummer "1234567" ein, ohne Vorwahl und ohne Landeskennzeichen.
  Ergebnis:
  Es wird immer die Rufnummer "1234567" angewählt, unabhängig vom Standort des Anrufers.
- Sie tragen im Telefonbuch von TeleService zusätzlich zur Rufnummer "1234567" noch die standortspezifischen Wahlparameter für die Vorwahl von Karlsruhe "0721" und das Landeskennzeichen "+49" für Deutschland ein.
  Ergebnis:
  Aus Karlsruhe wird die Telefonnummer "1234567" angewählt.
  Aus einer anderen Stadt innerhalb Deutschlands wird die Telefonnummer "07211234567" angewählt.
  Aus einem anderen Land wird die Telefonnummer "00497211234567" angewählt.

Hinweis

Anzeige im Telefonbuch

Im Telefonbuch von TeleService sind die Spalten "Ortskennzahl" und "Land" nur dann editierbar, wenn Sie die Wählregeln wie oben beschrieben definiert haben.

7.3 Fernverbindung als Wählverbindung

7.3.1 Grundlagen zum Aufbauen einer Wählverbindung

Einsetzen eines TS Adapters für Wählverbindungen

Für den Aufbau einer Fernverbindung als Wählverbindung mit TeleService ist der Einsatz eines TS Adapters erforderlich.

Der TS Adapter dient dazu, ein Automatisierungssystem über ein Modem an ein Telefonnetz anzuschließen, um es so für den Einsatz von TeleService vorzubereiten. Hierzu umfasst der TS Adapter einen Parameterspeicher, in dem ein Parametersatz für den TeleService-Betrieb gespeichert wird.

Durch die Funktion "Exportieren von Adapterparametern" können verschiedene Parametersätze in externen Dateien gespeichert und durch die Funktion "Importieren von Adapterparametern" wieder in den TS Adapter geladen werden.

Aufbau der Fernverbindung als Wählverbindung

Sie haben die Auswahl zwischen verschiedenen TS Adaptern, die jeweils unterschiedliche Funktionalität und Anschlussmöglichkeiten bieten.

Das folgende Bild zeigt zwei mögliche Konfigurationen, um mit dem TS Adapter IE eine Wählverbindung zu einer Anlage aufzubauen.

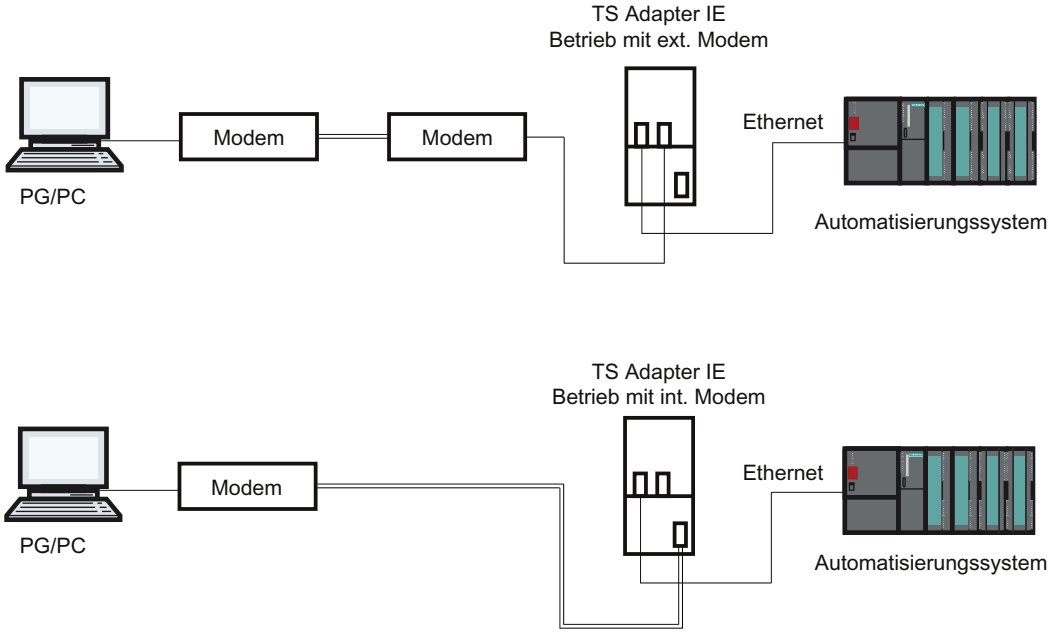

Übersicht der einsetzbaren TS Adapter:

Den TS Adapter gibt es in den folgenden Ausführungen:

- TS Adapter II (auch als "TS Adapter MPI" bezeichnet)
- TS Adapter IE Standard (auch als "TS Adapter IE" bezeichnet)
- TS Adapter IE Basic (auch als "TS Adapter IE" bezeichnet)

Bezeichnung "TS Adapter"

Nachfolgend steht die Bezeichnung "TS Adapter" für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter II" oder "TS Adapter IE Standard".

Hinweis

Weitere Informationen zum von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

Siehe auch

Kurzbeschreibung des TS Adapter MPI (Seite 149)

Kurzbeschreibung TS Adapter IE (Seite 157)

Exportieren von Adapterparametern (Seite 155)

Importieren von Adapterparametern (Seite 156)

7.3.2 Telefonnetze und Modems

7.3.2.1 Zulässige Telefonnetze und Modems

Einsetzbare Telefonnetze

TeleService ist mit digitalen Netzen (ISDN), analogen Netzen und Funknetzen (mit GSM-Technologie) einsetzbar. Unterstützt wird in dieser Version eine Fernverbindung zu einem TS Adapter.

Modem-Unterstützung

TeleService wurde Modem-unabhängig realisiert. Somit können alle handelsüblichen Modems, die in der Systemsteuerung von Windows installierbar und dort als Modem sichtbar sind, auch von TeleService genutzt werden.

Die Auswahl der Modemtypen wird in erster Linie durch die vorhandene Hardware des PG/PC und das einzusetzende Telefonnetz bestimmt.

Folgende Modemtypen/Medien werden unterstützt:

- Modems (externe Modems an COM-Schnittstelle, interne Modems und PCMCIA-Karten)
- Externer ISDN-Adapter an COM-Schnittstelle bzw. USB-Schnittstelle
- Interner ISDN-Adapter mit virtueller COM-Schnittstelle (z. B. AVM CAPI-Port)
- externe ISDN-Modems (ISDN-Adapter mit integrierter Analog-Modem-Funktionalität) an COM-Schnittstelle bzw. USB-Schnittstelle
- Funknetz-Modems mit GSM-Technologie, PCMCIA-Adapterkarte bzw. Datenkabel und Mobiltelefon

Netzübergänge

Netzübergänge zwischen den verschiedenen Telefonnetzen sind prinzipiell möglich. Fernverbindungen von einem ISDN-Adapter zu einem Analog-Modem und umgekehrt funktionieren nur mit speziellen ISDN-Telefon-Adaptern.

Leistungsfähigkeit in Telefonnetzen

Der Datendurchsatz einer Fernverbindung hängt von dem eingesetzten Modem, dem verwendeten Telefonnetz und von der Qualität der Telefonleitung ab.

7.3.2.2 Installieren des lokalen Modems

Einführung

Haben Sie unter Ihrem Betriebssystem bereits ein Modem für die Datenübertragung installiert, so können Sie dieses Modem auch für TeleService verwenden.

Ist dem Betriebssystem noch kein Modem bekannt, müssen Sie ein Modem installieren, um eine Fernverbindung mit TeleService aufbauen zu können.

Vorgehen

Gehen Sie folgendermaßen vor:

1. Stellen Sie sicher, dass Ihr PG/PC und das Modem ausgeschaltet sind.
2. Schließen Sie das externe Modem physikalisch an eine COM- oder USB-Schnittstelle Ihres PG/PC an. Sie können auch ein internes Modem bzw. eine PCMCIA-Karte entsprechend den Angaben des Herstellers einbauen.
3. Schalten Sie nun zuerst das Modem und anschließend das PG bzw. den PC ein.

Ergebnis

Plug&Play-fähige Modems werden vom Betriebssystem automatisch erkannt und installiert. Sie werden mit Hilfe von Dialogen durch den Installationsvorgang geführt.

Hinweis

Nicht Plug&Play-fähige Modems

Wenn Ihr Modem beim Einschalten nicht automatisch erkannt wird, so müssen Sie es über die Systemsteuerung selbst installieren.

Beachten Sie dabei die Hinweise in der zu Ihrem Modem mitgelieferten Dokumentation.

7.3.2.3 Anschließen und Parametrieren des entfernten Modems

Einführung

Eine Voraussetzung für das Arbeiten mit TeleService ist, dass an der entfernten Anlage ebenfalls ein Modem angeschlossen ist. Dieses Modem wird als "entferntes Modem" bezeichnet.

Parametrieren des entfernten Modems

Das Modem erhält alle zum Betrieb erforderlichen Parameter von dem angeschlossenen TS Adapter. Dies sind u. a. die Daten zur Initialisierung des Modems und die Einstellungen zur seriellen Übertragung zwischen TS Adapter und Modem.

Die für das entfernte Modem benötigten Daten werden bereits beim Parametrieren des TS Adapters festgelegt.

Abhängig vom eingesetzten TS Adapter kann es sich um ein internes oder externes Modem handeln.

Vorgehen zum Anschließen eines TS Adapter mit internem Modem

1. Schalten Sie den TS Adapter aus.
2. Schließen Sie den TS Adapter an das Automatisierungssystem an.
3. Schließen Sie den TS Adapter an den Telefonanschluss an.
4. Schalten Sie den TS Adapter ein.

Vorgehen zum Anschließen eines TS Adapter mit externem Modem

1. Schalten Sie das Modem aus.
2. Schließen Sie den TS Adapter an das Automatisierungssystem an.
3. Schließen Sie den TS Adapter mit einem Modemkabel an das Modem an.
4. Schließen Sie das Modem an den Telefonanschluss an.

5. Schalten Sie das Modem ein.
6. Schalten Sie den TS Adapter ein.

Hinweis

Beachten Sie die folgenden Hinweise zum Parametrieren des entfernten Modems:

- Die im TS Adapter voreingestellten Parameter für das Modem und die serielle Schnittstelle sollten in den meisten Fällen einen erfolgreichen Betrieb gewährleisten, sodass nur in seltenen Ausnahmefällen eine Umparametrierung erforderlich ist.
- Eine Umparametrierung des TS Adapter ist nur dann nötig, wenn keine Modemverbindung zu Stande kommt oder Werkseinstellungen angepasst oder optimiert werden sollen.
- Der TS Adapter kann entweder über eine Direktverbindung oder eine Fernverbindung umparametriert werden.

7.3.3 Zugriffsschutz bei Wählverbindungen

7.3.3.1 Informationen zum Zugriffsschutz

Einführung

Beim Parametrieren Ihres TS Adapters können Sie sowohl den Zugriff auf die Parametrierung des TS Adapters selbst als auch den Zugriff auf entfernte Anlagen schützen.

Gültigkeit des Zugriffsschutzes

Der Zugriffsschutz besteht beim TS Adapter MPI ausschließlich für Fernverbindungen, im Direktanschluss kann man jederzeit auf die Parametrierung des TS Adapters zugreifen.

Beim TS Adapter IE besteht der Zugriffsschutz auch im Direktanschluss.

Informationen zum Zugriffsschutz

Bei Auslieferung des TS Adapters MPI ist werkseitig kein Zugriffsschutz aktiviert. Beim TS Adapter IE gibt es ein Default-Passwort.

Der erste Benutzer, der den Adapter parametriert, kann also den Zugriffsschutz aktivieren, indem er das Kennwort für einen Benutzer und/oder eine Rückrufnummer definiert.

Es handelt sich um einen mehrstufigen Zugriffsschutz mit mehreren Benutzern, jeweils mit oder ohne Administratorrechte. Beim TS Adapter MPI gibt es nur einen Administrator und maximal zwei Benutzer.

Nur ein Administrator darf andere Benutzer einrichten, deren Einstellungen ändern und diese bei Bedarf auch wieder löschen. Melden Sie sich als Benutzer ohne Administratorrechte an, so können Sie nur das eigene Kennwort und die eigene Rückrufnummer ändern. Beim TS Adapter MPI kann man im Direktanschluss jedoch uneingeschränkt auf die Parametrierung des TS Adapters zugreifen.

Vorteile

Der Zugriffsschutz bietet Ihnen folgende Vorteile:

- Ein unberechtigter Zugriff betriebsfremder Personen ist nahezu ausgeschlossen.
- Der Anlagenbetreiber trägt den Hauptteil der Telefongebühren.

7.3.3.2 Rückrufmöglichkeiten in TeleService

Rückrufvarianten

Die Kosten einer Telefonverbindung trägt normalerweise der Anrufer, d. h. derjenige, der die Wählverbindung aufbaut.

TeleService kann jedoch auch so genutzt werden, dass nach einer kurzen Erstverbindung die Modemverbindung erneut in umgekehrter Richtung, d. h. vom TS Adapter ausgehend, aufgebaut wird (Rückruf). In diesem Fall trägt der Betreiber der Anlage die Kosten für den Rückruf.

TeleService unterscheidet zwei Rückrufvarianten:

1. Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer.
2. Rückruf zu einer im TS Adapter hinterlegten Rufnummer.

7.3.3.3 Informationen zu den Schutzstufen

Einführung

Bei Einrichten des Zugriffsschutzes für den Zugriff durch TeleService auf den TS Adapter gibt es zwei Schutzstufen, die mit unterschiedlichen Optionen verbunden sind.

Optionen für den Zugriffsschutz

Zugriffsschutzstufe 1:

Der TS Adapter ist durch den Benutzernamen und das Kennwort geschützt. Sie können auf den TS Adapter über beliebige Telefonanschlüsse zugreifen und optional beim Verbindungsaufbau eine beliebige Rückrufnummer angeben.

Zugriffsschutzstufe 2:

Der TS Adapter ist durch den Benutzernamen, das Kennwort und die Rückrufnummer geschützt. Sie können auf den TS Adapter nur über einen Telefonanschluss pro Benutzer zugreifen.

Die nachfolgende Tabelle zeigt die o. g. Bedingungen für die unterschiedlichen Schutzstufen:

Zugriffsschutzstufe	Administrator-/Benutzer-Kennwort	Rückrufnummer
1	eintragen	nicht eintragen
2	eintragen	eintragen

Anmelden am TS Adapter

Beim Anmelden am TS Adapter geben Sie, nachdem Sie den Zugriffsschutz eingerichtet haben, den Benutzernamen, das dazugehörige Kennwort und optional eine Rückrufnummer ein:

Zugriffsschutzstufe	Administrator-/Benutzer-Kennwort	Rückrufnummer
1	eintragen	nicht eintragen oder beliebige Rückrufnummer
2	eintragen	nicht eintragen

Haben Sie beim Verbindungsaufbau eine Rückrufnummer eingetragen (Zugriffsschutzstufe 1), oder eine Rückrufnummer im TS Adapter hinterlegt (Zugriffsschutzstufe 2), so wird die Modemverbindung getrennt und der TS Adapter ruft die angegebene Nummer zurück.

7.3.3.4 Zugriffsschutz und Rückrufnummer für den TS Adapter einrichten

Einführung

In TeleService können Sie beim Parametrieren des TS Adapters einen Zugriffsschutz und eine Rückrufnummer für die Parametrierung des Adapters und für die Verbindung zur entfernten Anlage einrichten. Nachfolgend ist die Parametrierung für einen TS Adapter MPI beschrieben. Die Parametrierung eines TS Adapters IE erfolgt analog. Die genaue Vorgehensweise ist in der Webhilfe dieses Adapters beschrieben.

Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird unter "Erreichbare Teilnehmer" in der Projektnavigation angezeigt.

Vorgehen

Um den Zugriffsschutz für den TS Adapter einzurichten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Befehl "TS Adapter MPI parametrieren".
2. Öffnen Sie das Register "Zugriffsschutz".
3. Geben Sie hier für Ihren Benutzernamen ein Kennwort und/oder eine Nummer ein, die das Modem nach dem Anmelden zurückrufen soll.
   - Sind Sie als Administrator angemeldet, so können Sie alle Einstellungen für Administratoren und Benutzer ändern sowie Benutzer einrichten und löschen.
   - Sind Sie als Benutzer angemeldet, können Sie nur Ihre eigenen Einstellungen (Kennwort und Rückrufnummer) ändern.
4. Bestätigen Sie alle Eingaben vor dem Verlassen des Dialoges mit "OK".
5. Drücken Sie die Schaltfläche "Ja" zum Bestätigen der nachfolgenden Abfrage.

Ergebnis

Die Parametrierung für den Zugriffsschutz und die Rückrufnummer wird im nichtflüchtigen Speicher des TS Adapters MPI gespeichert.

Hinweis

Beim Einrichten des Zugriffsschutzes ist unbedingt zu beachten:

- Um die Rückruf-Funktionalität zu gewährleisten, sollten Sie sicherstellen, dass die Einstellungen im Register "Modem" den Gegebenheiten an der Anlage entsprechen.
- Wenn Sie unter dem Benutzer "ADMIN" eine falsche Rückrufnummer eintragen, so können Sie den TS Adapter MPI über eine Fernverbindung nicht mehr erreichen!
- Testen Sie die Rückrufnummer, bevor Sie diese unter dem Benutzer "ADMIN" eintragen, durch einen Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer (Zugriffsschutzstufe 1).

7.3.3.5 Rückruf in TeleService durchführen

Rückrufmöglichkeiten

In TeleService besteht die Möglichkeit, zwei unterschiedliche Rückrufvarianten einzurichten.

Folgende Rückrufmöglichkeiten stehen zur Verfügung:

- Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer.
- Rückruf zu einer im TS Adapter hinterlegten Rufnummer

Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer

1. Klicken Sie in der Projektnavigation des TIA Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste "Adapter-Typ" den von Ihnen verwendeten Adapter-Typ aus.
5. Wählen Sie unter "Verbindungs-Typ" die "Wählverbindung" aus, sofern nicht bereits angewählt.
6. Wählen Sie unter "Lokale Einstellungen" das von Ihnen verwendete Modem aus.
7. Tragen Sie die anzuwählende Telefonnummer in das entsprechende Eingabefeld ein oder öffnen Sie das Telefonbuch, indem Sie auf die dahinterliegende Schaltfläche klicken und übernehmen Sie die gewünschte Telefonnummer aus dem Telefonbuch.
8. Tragen Sie den Benutzernamen und das zugehörige Kennwort des TS Adapters ein.
9. Falls Sie einen "Verbindungsaufbau mit Rückruf" wünschen, selektieren Sie das entsprechende Optionsfeld.

10. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben. Unter "Status" wird angezeigt, ob bereits eine Fernverbindung besteht oder nicht.
11. Tragen Sie ggf. im nachfolgenden Dialog die gewünschte Rückrufnummer ein.

Ergebnis

Die Fernverbindung zur gewünschten Anlage mit Rückruf wird aufgebaut.

In der Projektnavigation wird die verbundene Anlage mit dem entsprechenden Symbol angezeigt.

Hinweis

Diese Vorgehensweise bietet sich an, wenn die Kosten für die Modemverbindung auf Anlagenseite auflaufen sollen und wenn die eigentliche Rückrufnummer variabel gehandhabt werden soll, d. h. wenn der Rückruf nicht immer an denselben Empfänger geleitet werden soll. Dies ist vor allem auch für solche Anwender interessant, die mobil sind.

Rückruf zu einer im TS Adapter hinterlegten Rufnummer

1. Parametrieren Sie die gewünschte Rückrufnummer im TS Adapter.
2. Bauen Sie eine Verbindung zum TS Adapter auf, wie oben beschrieben und beachten Sie dabei folgende Besonderheiten:
   - Tragen Sie den Benutzernamen und das Kennwort ein, für die die Rückrufnummer im TS Adapter parametriert ist.
   - Das Optionsfeld "Verbindungsaufbau mit Rückruf" brauchen Sie nicht zu selektieren, da die Rückrufnummer dem TS Adapter bereits bekannt ist.

Ergebnis

Der Rückruf zu einer im TS Adapter hinterlegten Rufnummer ist eingerichtet. Wenn eine Fernverbindung aufgebaut wird, erfolgt der Rückruf von der entfernten Anlage.

Hinweis

Diese Vorgehensweise stellt die höchste Stufe des Zugriffsschutzes dar. Sie birgt jedoch die Gefahr, dass bei einer im TS Adapter falsch hinterlegten Rückrufnummer der TS Adapter nicht mehr über eine Modemverbindung erreicht werden kann und nur durch eine Umparametrierung vor Ort wieder in Betrieb genommen werden kann.

7.3.4 TS Adapter MPI

7.3.4.1 Kurzbeschreibung des TS Adapter MPI

TS Adapter MPI:

Die Bezeichnung "TS Adapter MPI" ist ein Sammelbegriff für alle TS Adapter, die über eine MPI/DP-Schnittstelle verfügen.

Den TS Adapter MPI gibt es in folgenden Ausführungen:

- als TS Adapter I (kann nicht über das TIA Portal parametriert werden)
- als TS Adapter II

Die nachfolgende Tabelle gibt eine Kurzbeschreibung über die jeweilige Funktionalität. Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

TS Adapter II:	
Direktanschluss	Der Anschluss erfolgt über den Universal Serial Bus (USB). Die Firmware ist austauschbar. Das Modem ist integriert oder auch als externes Modem anschließbar. Der TS Adapter II schaltet automatisch zwischen den Modems um. Solange kein eingeschaltetes externes Modem angeschlossen ist, benutzt er das interne Modem.
Es gibt zwei Varianten	• Mit internem Analog-Modem. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden. • Mit internem ISDN-Adapter. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden

Verwendung der Bezeichnung "TS Adapter"

Bei TeleService steht die Bezeichnung "TS Adapter" als Generalisierung für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung eines TS Adapters Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter II", "TS Adapter IE Standard" oder "TS Adapter IE Basic".

7.3.4.2 Funktionsweise des TS Adapter MPI

Funktionsweise des TS Adapters MPI

Je nach Konfiguration verbindet der TS Adapter MPI die serielle Schnittstelle oder die USB-Schnittstelle Ihres PG/PC (Direktanschluss) oder die serielle Schnittstelle eines Modems (Modemanschluss) mit dem MPI-/PROFIBUS-Netz Ihres Automatisierungssystems.

Der TS Adapter MPI verfügt über einen nichtflüchtigen Speicher. In diesem Speicher sind Parameter für folgende Funktionen hinterlegt:

- das MPI/PROFIBUS-Netz (Netz-Parameter)
- den Betrieb des eingesetzten Modems
- die serielle Schnittstelle zum Modem
- den Zugriffsschutz

Standard-Parametrierung

Der TS Adapter wird mit einer Standard-Parametrierung ausgeliefert. Die Parameter können im Rahmen einer Parametriersitzung eingestellt und in den nichtflüchtigen Speicher des TS Adapters gespeichert werden.

In der Konfiguration "Direktanschluss" werden vom TS Adapter nur die Netz-Parameter für den Zugang zum MPI-/PROFIBUS-Netz verwendet.

In der Konfiguration "Modemanschluss" werden alle auf dem TS Adapter abgelegten Parameter aktiviert.

Hinweis

Weitere Hinweise zur Parametrierung Ihres TS Adapters entnehmen Sie bitte der zum TS Adapter mitgelieferten Dokumentation.

7.3.4.3 TS Adapter MPI im Direktanschluss betreiben

Direktanschluss mit TS Adapter MPI

Der Direktanschluss dient zum Parametrieren des TS Adapters MPI. In der gleichen Konfiguration können Sie auch im TIA Portal online gehen und somit die parametrierten MPI-/PROFIBUS-Parameter auf Busverträglichkeit überprüfen. Damit ist (wie mit einem PC-Adapter) ein Zugang zu SIMATIC S7/C7-Systemen über die MPI-/DP-Schnittstelle möglich, ohne dass ein PG/PC-Steckplatz durch eine MPI-/PROFIBUS-Baugruppe belegt wird.

Im Direktanschluss ist kein Zugriffsschutz für den TS Adapter aktiv. Somit kann die Parametrierung des TS Adapters ohne Weiteres verändert werden, z. B. durch das Importieren von Adapterparametern.

Hinweis

Anzeige des TS Adapters MPI im TIA Portal

Sobald Sie einen TS Adapter MPI über die USB-Schnittstelle mit dem PG/PC verbunden haben, wird im TIA Portal innerhalb der Projektnavigation der Ordner "TS Adapter" angezeigt.

Wenn Sie den Ordner öffnen, können Sie den angeschlossenen TS Adapter MPI über die nachfolgenden Dialoge wie gewünscht parametrieren.

Aufbau des Direktanschlusses für TS Adapter MPI

Beim Direktanschluss besteht über den TS Adapter MPI eine direkte Verbindung zwischen dem PG bzw. PC, auf dem TeleService installiert ist, und dem Automatisierungssystem. Ein Modem ist nicht erforderlich.

Das nachfolgende Bild zeigt den Aufbau des TS Adapters MPI im Direktanschluss.

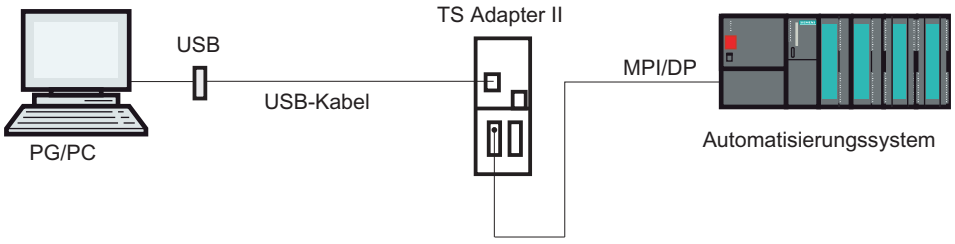

7.3.4.4 TS Adapter MPI im Modemanschluss betreiben

Einführung zum Modemanschluss mit TS Adapter MPI

Diese Konfiguration ermöglicht es Ihnen, sich in eine entfernte Anlage einzuwählen. Dazu bauen Sie mit TeleService über ein Telefonnetz eine Fernverbindung zu einer entfernten Anlage auf. Über die aufgebaute Modem-Verbindung können Sie dann die angewählte Anlage wie gewohnt mit dem TIA-Portal bearbeiten.

Aufbau des Modemanschlusses mit TS Adapter MPI

Die Verbindung zwischen dem PG bzw. PC, auf dem TeleService installiert ist, und dem Automatisierungssystem, an dessen MPI-/DP-Schnittstelle der TS Adapter MPI aufgesteckt ist, wird über eine Modemstrecke aufgebaut.

Die Konfiguration schließt somit das PG bzw. den PC über das Telefonnetz und den TS Adapter MPI an die MPI-/DP-Schnittstelle des Automatisierungssystems an.

Das nachfolgende Bild zeigt den Aufbau im Modemanschluss.

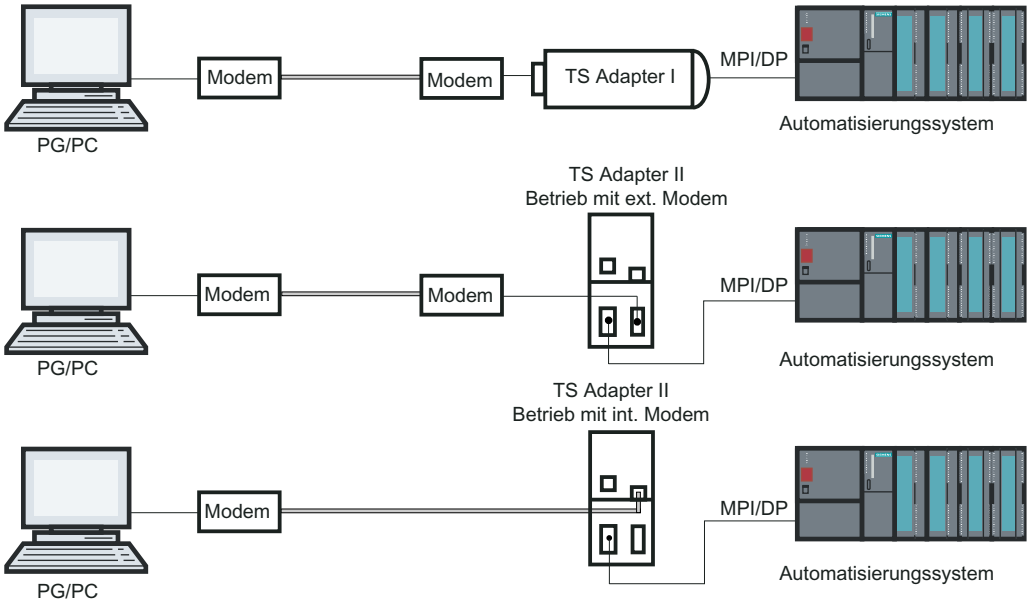

Hinweis

Parallelbetrieb zwischen Direkt- und Modemanschluss

Der TS Adapter II hat zwei Anschlüsse für die Kommunikation mit dem PG/PC, die beide gleichzeitig angeschlossen sein können. Sie verbinden die USB-Schnittstelle mit dem PG/PC und gleichzeitig die Modemschnittstelle mit dem Telefonnetz.

In dieser Konfiguration können Sie entweder den Direkt- oder den Modemanschluss nutzen.

Ein Parallelbetrieb ist **nicht** möglich!

7.3.4.5 Parametriermöglichkeiten für den TS Adapter MPI

Wissenswertes zum Parametrieren des TS Adapter MPI

Der TS Adapter MPI kann sowohl im Direktanschluss als auch über eine bestehende Fernverbindung parametriert werden.

Folgende Parametriermöglichkeiten stehen zur Verfügung:

- Neuparametrierung (Seite 153)
- Wiederherstellen der Standard-Parametrierung (Seite 154)
- Importieren von Adapterparametern (Seite 156)
- Exportieren von Adapterparametern (Seite 155)
- Zugriffsschutz einrichten (Seite 146)

Parametrierung durchführen

Parametrieren Sie Ihren TS Adapter unter Berücksichtigung der zum TS Adapter mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

Hinweis

Beachten Sie beim Parametrieren des TS Adapters MPI:

- Wenn Sie bei bestehender Fernverbindung die aktuellen Parameter ändern, so besteht die Gefahr, dass anschließend mit den geänderten Parametern keine Modemverbindung mehr aufgebaut werden kann. In diesem Fall kann der TS Adapter MPI nur noch im Direktanschluss parametriert werden.
- Das bedeutet, dass die Parametrierung mit einem PG/PC am Ort der Anlage vorgenommen werden muss oder dass der TS Adapter MPI zum Parametrieren zum Standort des lokalen PG/PC gebracht werden muss.

Positive Quittierung

Bei einer Parametrierung werden die Daten in den nichtflüchtigen Speicher des TS Adapters MPI geschrieben. Erst nachdem alle Vorkehrungen getroffen sind, dass die Parameteränderung richtig durchgeführt wurde und somit einen Spannungsausfall überlebt, wird der Parametriervorgang positiv quittiert.

Änderungen werden beim TS Adapter MPI wie folgt wirksam:

- Die seriellen Parameter, die Modem-Parameter und die Parameter für den Zugriffsschutz werden nach dem Abbau der Fernverbindung aktiviert.
- Die geänderten Netz-Parameter werden sofort aktiviert.

7.3.4.6 TS Adapter MPI parametrieren

Einführung

Sie können den TS Adapter MPI sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung im Modemanschluss parametrieren.

Nachfolgend wird die Vorgehensweise für das Parametrieren beschrieben.

Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und der Ordner "TS Adapter" wird unter "Online-Zugänge" in der Projektnavigation angezeigt.

Vorgehen

Um den TS Adapter MPI im Direktanschluss zu parametrieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Ordner "Online Zugänge".
2. Öffnen Sie den gewünschten Ordner:
   - Bei einer Direktverbindung den Ordner "TS Adapter".
   - Bei einer bestehenden Fernverbindung den Ordner "TeleService" und dann den Ordner mit dem gewünschten Anlagennamen.
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters MPI gespeichert. Die Parametrierung ist damit abgeschlossen.

7.3.4.7 Standard-Parametrierung für TS Adapter MPI wieder herstellen

Einführung

Für den TS Adapter MPI besteht die Möglichkeit, die beim Auslieferungszustand vorhandene Standardparametrierung wieder herzustellen.

Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online-Zugänge" im Ordner "TeleService" angezeigt.

Vorgehen

Um die Standardparametrierung für den TS Adapter MPI wieder herzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie unter "Allgemein" auf die Schaltfläche "Zurücksetzen".
5. Bestätigen Sie Ihre Eingaben mit "OK".

Ergebnis

Die Standardparametrierung, mit der der TS Adapter MPI ausgeliefert wurde, ist wieder hergestellt.

Siehe auch

Parametriermöglichkeiten für den TS Adapter MPI (Seite 152)

7.3.4.8 Exportieren von Adapterparametern

Einführung

Es besteht die Möglichkeit, die Parametrierung eines TS Adapters MPI in eine externe Datei zu exportieren. Die in dieser Datei abgelegte Parametrierung können Sie wiederum in beliebig viele TS Adapter MPI importieren.

Dies bietet sich beispielsweise an, wenn Sie mehrere TS Adapter MPI identisch parametrieren möchten oder wenn Sie den Parametersatz speichern, dokumentieren oder verteilen möchten.

Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online-Zugänge" im Ordner "TeleService" angezeigt.

Vorgehen

Um die Adapterparameter eines TS Adapters MPI zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie auf die Schaltfläche "Exportieren".
5. Wählen Sie im nachfolgenden Fenster die Datei aus, in die Sie die Parametrierung des TS Adapters MPI exportieren möchten.
6. Bestätigen Sie mit "Speichern".

Ergebnis

Die Parameter des TS Adapters MPI werden in der angegebenen Datei (*.tap) gespeichert. Damit ist der Export der Adapterparameter abgeschlossen.

7.3.4.9 Importieren von Adapterparametern

Einführung

Es besteht die Möglichkeit, die Parametrierung eines TS Adapters MPI aus einer zuvor angelegten Export-Datei (*.tap) zu importieren.

Die in dieser Datei abgelegte Parametrierung können Sie in beliebig viele TS Adapter importieren. Dies bietet sich beispielsweise an, wenn Sie mehrere TS Adapter MPI identisch parametrieren möchten.

Das Importieren können Sie lokal im Direktanschluss oder über eine aufgebaute Fernverbindung im Modemanschluss vornehmen.

Der TS Adapter übernimmt die Parameter der ausgewählten Export-Datei und arbeitet mit diesen Einstellungen weiter. Dabei gehen alle bis dahin vorgenommenen und nicht in einer Export-Datei gespeicherten Einstellungen verloren.

Bevor Sie die Datei importieren, sichern Sie die aktuelle Parametrierung in einer Export-Datei. Wenn notwendig, können Sie die Parametrierung wieder herstellen.

Voraussetzung

- Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online-Zugänge" im Ordner "TeleService" angezeigt.

Vorgehen

Um die Adapterparameter eines TS Adapters MPI zu importieren gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie auf die Schaltfläche "Importieren".
5. Wählen Sie nun die Datei aus, in die Sie die Parametrierung des TS Adapters MPI importieren möchten.
6. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

Ergebnis

Die ausgewählten Parameter werden im nichtflüchtigen Speicher des TS Adapters MPI gespeichert. Der Import der Adapterparameter ist damit abgeschlossen.

7.3.5 TS Adapter IE

7.3.5.1 Kurzbeschreibung TS Adapter IE

TS Adapter IE

Die Bezeichnung "TS Adapter IE" ist ein Sammelbegriff für alle TS Adapter, die über eine Ethernet-Schnittstelle verfügen.

Den TS Adapter IE gibt es in folgenden Ausführungen:

- als TS Adapter IE Standard
- als TS Adapter IE Basic

Die nachfolgenden Tabellen geben eine Kurzbeschreibung über die jeweilige Funktionalität. Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

TS Adapter IE Standard:
Direktanschluss über Industrial Ethernet (IE). Firmwareupdate möglich. Modem integriert oder auch als externes Modem anschließbar. Keine automatische Modemumschaltung wie beim TS Adapter II. Die Parametrierung erfolgt über ein Web-Interface.
Es gibt 2 Varianten:
• Mit internem Analog-Modem. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden.
• Mit internem ISDN-Adapter. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden.

TS Adapter IE Basic:
Direktanschluss über Industrial Ethernet (IE). Firmwareupdate möglich. Steckbare Module. Die Parametrierung erfolgt über ein Web-Interface.
Es gibt 4 Varianten:
• TS Adapter IE Basic MODEM: Grundgerät TS Adapter IE Basic mit TS Module MODEM für den Betrieb am analogen Telefonnetz.
• TS Adapter IE Basic ISDN: Grundgerät TS Adapter IE Basic mit TS Module ISDN für den Betrieb an ISDN-Telefonanlagen.
• TS Adapter IE Basic GSM: Grundgerät TS Adapter IE Basic mit TS Module GSM für den Betrieb am GSM-Funknetz.
• TS Adapter IE Basic RS232: Grundgerät TS Adapter IE Basic mit TS Module RS232 zum Anschluss eines externen Modems.

Verwendung der Bezeichnung "TS Adapter"

In der Online-Hilfe zu TeleService steht die Bezeichnung "TS Adapter" als Generalisierung für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung eines TS Adapters Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter I", "TS Adapter II", "TS Adapter IE Standard" oder "TS Adapter IE Basic".

7.3.5.2 Funktionsweise des TS Adapters IE

Funktionsweise des TS Adapters IE

Der TS Adapter IE verbindet das Telefonnetz oder die serielle Schnittstelle eines Modems mit dem Industrial Ethernet Ihres Automatisierungssystems.

Der TS Adapter IE verfügt über einen nichtflüchtigen Speicher. In diesem Speicher sind Parameter für folgende Funktionen hinterlegt:

- den Betrieb des eingesetzten Modems
- die serielle Schnittstelle zum Modem
- den Zugriffsschutz

Standard-Parametrierung

Der TS Adapter IE wird mit einer Standard-Parametrierung ausgeliefert. Die Parameter können im Rahmen einer Parametriersitzung eingestellt und in den nichtflüchtigen Speicher des TS Adapters gespeichert werden.

Hinweis

Weitere Hinweise zur Parametrierung Ihres TS Adapters entnehmen Sie bitte der zum TS Adapter mitgelieferten Dokumentation.

7.3.5.3 Anschlussarten

Anschlussarten des TS Adapter IE Basic

Die folgenden Bilder zeigen die möglichen Anschlussarten des TS Adapter IE Basic.

Direktanschluss

Im Direktanschluss am PG/PC können Sie den TS Adapter IE Basic über Ethernet parametrieren.

Hinweis

Der Betrieb des TS Adapter IE Basic ohne ein TS Modul ist unzulässig.

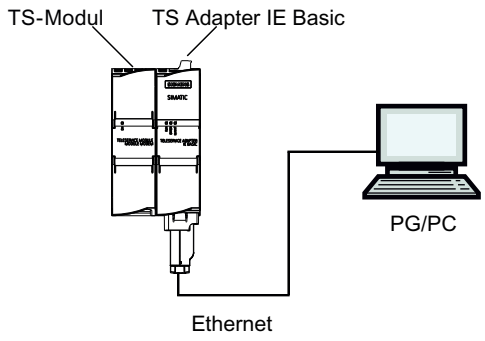

Bild 7-1 Direktanschluss

Anschluss an das Telefonnetz

Zum direkten Anschluss an das Telefonnetz müssen Sie den TS Adapter IE Basic zusammen mit einem der folgenden TS Module betreiben:

- TS Module Modem
- TS Module ISDN

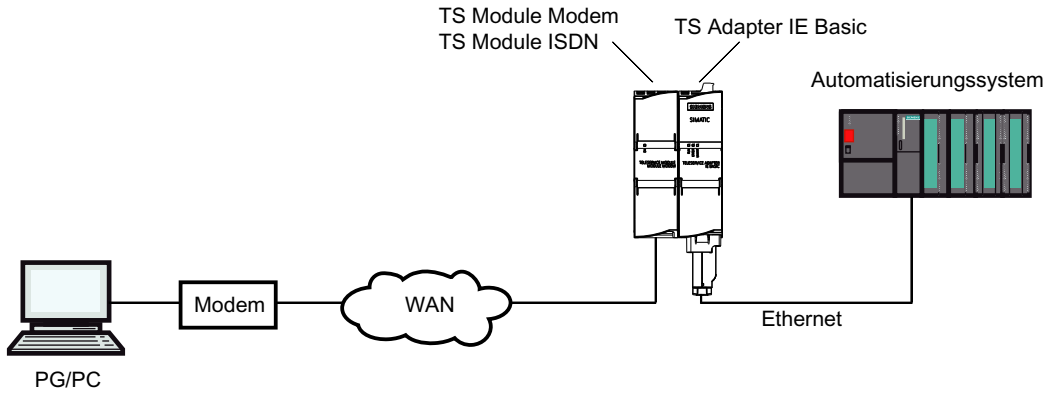

Bild 7-2 Direkter Anschluss an das Telefonnetz

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

Anschluss an das GSM-Netz

Zum Anschluss an das GSM-Netz müssen Sie den TS Adapter IE Basic zusammen mit diesem TS Module betreiben:

- TS Module GSM

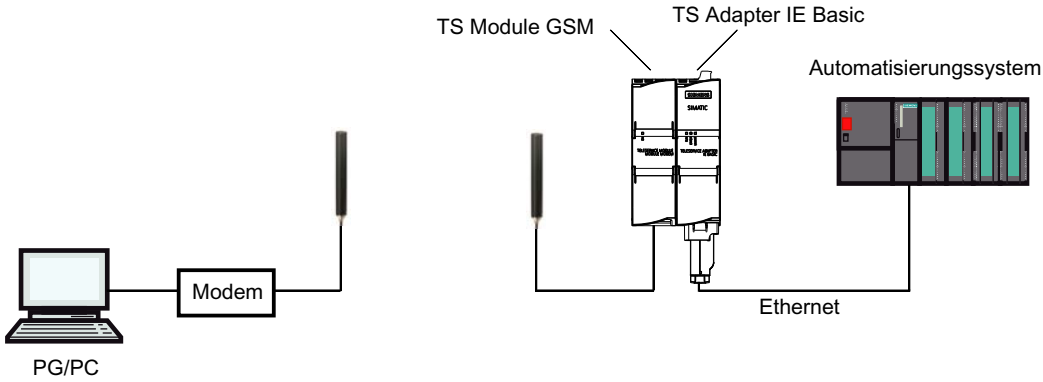

Bild 7-3 Anschluss an das GSM-Netz

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

Anschluss an das Telefonnetz über ein externes Modem

Zum Anschluss an ein externes Modem müssen Sie den TS Adapter IE Basic zusammen mit diesem Modul betreiben:

- TS Module RS232

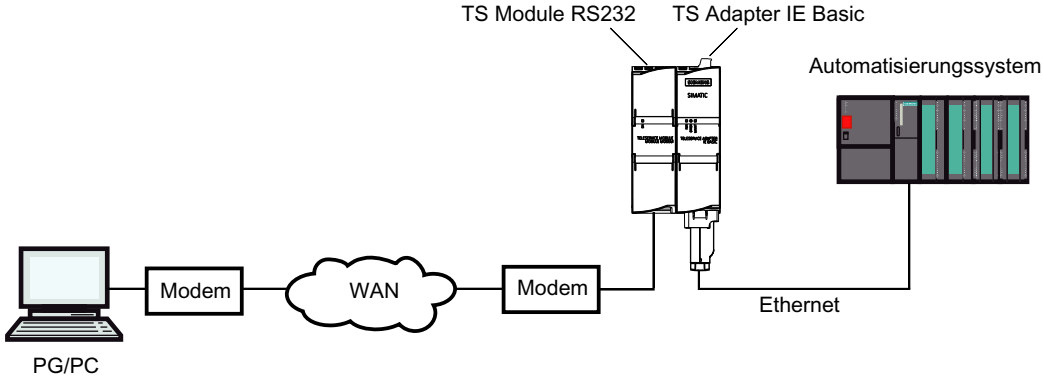

Bild 7-4 Anschluss an ein externes Modem

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

7.3.5.4 Parametriermöglichkeiten für den TS Adapter IE

Wissenswertes zum Parametrieren des TS Adapter IE

Der TS Adapter IE wird über ein Web-Interface parametriert.

Für die Parametrierung des TS Adapter IE steht Ihnen eine dem Web-Interface zugeordnete Webhilfe zur Verfügung.

Folgende Parametriermöglichkeiten stehen zur Verfügung:

- Neuparametrierung
- Wiederherstellen der Standard-Parametrierung
- Importieren von Adapterparametern
- Exportieren von Adapterparametern

Hinweis

Parametrierung durchführen

Parametrieren Sie Ihren TS Adapter unter Berücksichtigung der zum TS Adapter mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

7.3.5.5 TS Adapter IE parametrieren

Einführung

Sie können den TS Adapter IE sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung im Modemanschluss parametrieren.

Nachfolgend werden beide Parametriermöglichkeiten beschrieben.

Spezifische Details zur Parametrierung des TS Adapters IE entnehmen Sie bitte der Dokumentation zum TS Adapter IE.

Parametrieren des TS Adapter IE im Direktanschluss

Voraussetzung

Es besteht eine LAN-Verbindung zu Ihrem TS Adapter IE.

Der TS Adapter IE Basic ist an die Stromversorgung angeschlossen.

Vorgehen

Um den TS Adapter IE zu parametrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Online Zugänge".
2. Doppelklicken Sie hier auf die Ethernet-Schnittstelle Ihres Rechners.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer anzeigen". Daraufhin wird der TS Adapter IE angezeigt.

4. Doppelklicken Sie auf den Ordner <TS Adapter IE> und anschließend auf "Online und Diagnose" und weisen Sie dem TS Adapter IE in den nachfolgenden Dialogen die gewünschte IP-Adresse zu. Beachten Sie dabei, dass die IP-Adresse der Schnittstellenkarte des PG/PC im selben Subnetz liegt, wie die IP-Adresse, die Sie für den TS Adapter IE vergeben.
5. Aktualisieren Sie die Ansicht in der Projektnavigation für die "Erreichbaren Teilnehmer", damit der TS Adapter IE mit der neu zugewiesenen IP-Adresse angezeigt wird.
6. Öffnen Sie den Ordner <TS Adapter IE> in der Teilnehmerliste.
7. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugeordnete Web-Interface zum Parametrieren des TS Adapters IE.
8. Führen Sie das "Logon" für das Web-Interface durch.
9. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
10. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE gespeichert. Die Parametrierung ist damit abgeschlossen.

Parametrieren des TS Adapter IE über Fernverbindung

Voraussetzung

Es besteht eine aufgebaute Fernverbindung zu einem TS Adapter IE .

Vorgehen

1. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Online Zugänge".
2. Öffnen Sie auf den Ordner "TeleService" und danach den gewünschten Anlagenordner.
3. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugehörige Web-Interface zum Parametrieren des TS Adapters IE. Das "Logon" für das Web-Interface wird automatisch mit den Login-Daten der Fernverbindung durchgeführt.
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE gespeichert. Die Parametrierung ist damit abgeschlossen.

7.3.6 Aufbauen einer Wählverbindung zu einer entfernten Anlage

7.3.6.1 Wählverbindung aufbauen

Einführung zum Aufbauen einer Fernverbindung als Wählverbindung

Eine Wählverbindung kommt zu Stande, wenn Sie sich mit TeleService über ein Telefonnetz in eine entfernte Anlage einwählen. Hierzu ist das PG/der PC mit TeleService über ein lokales Modem an das Telefonnetz angeschlossen. Am anderen Ende ist das Automatisierungssystem über einen parametrierten TS Adapter und ein Modem mit der Telefonleitung verbunden.

Voraussetzungen

Ein lokales Modem ist installiert und parametriert.

In der entfernten Anlage befindet sich ein TS Adapter.

Ein entferntes Modem ist installiert und parametriert.

Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation des TIA Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste "Adapter-Typ" den von Ihnen verwendeten Adapter-Typ aus.
5. Wählen Sie unter "Verbindungs-Typ" die "Wählverbindung" aus.
6. Wählen Sie unter "Lokale Einstellungen" das von Ihnen verwendete Modem aus.
7. Tragen Sie die anzuwählende Telefonnummer in das entsprechende Eingabefeld ein oder öffnen Sie das Telefonbuch, indem Sie auf die dahinterliegende Schaltfläche klicken und übernehmen Sie die gewünschte Telefonnummer aus dem Telefonbuch.
8. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
9. Falls Sie einen "Verbindungsaufbau mit Rückruf" wünschen, selektieren Sie das entsprechende Optionsfeld.
10. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben. Unter "Status" wird angezeigt, ob bereits eine Fernverbindung besteht oder nicht.

Ergebnis

Die Wählverbindung zur gewünschten Anlage wird aufgebaut.

Sobald die Wählverbindung zu Stande gekommen ist, wird der Dialog geschlossen. In der Statuszeile des TIA Portals erscheint die Meldung: "Fernverbindung ist aufgebaut". Sie können

nun die Fernverbindung mit dem TIA Portal nutzen und mit dem Automatisierungssystem kommunizieren.

Verbindungsaufbau nicht möglich

Ist kein Verbindungsaufbau möglich, so versuchen Sie bitte, die Fehlerursache mithilfe der "Hinweise zur Fehlersuche" zu ermitteln.

Trennen der Verbindung

Haben Sie die Bearbeitung der entfernten Anlage abgeschlossen, so beenden Sie die Fernverbindung in der Projektnavigation durch Doppelklicken auf den Eintrag "Fernverbindung aufbauen/trennen".

Durch Beenden des TIA Portals beenden Sie ebenfalls die Fernverbindung.

7.3.6.2 Wählverbindung trennen

Trennen einer aktiven Wählverbindung

Hinweis

Trennen der Wählverbindung

Bevor Sie die Fernverbindung trennen, sollten Sie im TIA Portal offline schalten.

Gehen Sie vor wie folgt:

1. Doppelklicken Sie im TIA Portal auf den Eintrag "Fernverbindung aufbauen/trennen".
2. Bestätigen Sie die Abfrage im nachfolgenden Dialog mit "Ja".

Ergebnis

Die Verbindung wird getrennt.

7.4 Fernverbindung als VPN-Verbindung

7.4.1 Grundlagen zum Aufbauen einer VPN-Verbindung

Einsetzen eines TS Adapters IE Advanced für VPN-Verbindungen

Für den Aufbau einer VPN-Verbindung mit TeleService ist der Einsatz eines TS Adapter IE Advanced erforderlich.

Definition VPN

VPN steht für "Virtual Private Network". Damit ist ein geschlossenes privates Rechnernetz gemeint, das auf einer öffentlichen Netzwerk-Infrastruktur basiert.

VPN dient dazu, räumlich getrennte Teilnehmer eines privaten Netzes mit einer Anschlussstelle in einem anderen Netz möglichst sicher zu verbinden.

Ein VPN-Teilnehmer erhält über eine verschlüsselte Verbindung direkten Zugriff auf die Anschlussstelle im anderen Netz.

VPN-Verbindungen

Beim Aufbau einer VPN-Verbindung mit TeleService wird zur eindeutigen Identifizierung des TS Adapters IE Advanced ein CA-Zertifikat und ein daraus generierter eindeutiger Fingerabdruck verwendet. Durch die VPN-Verbindung wird eine abhör- und manipulationssichere Kommunikation zwischen dem Remote-PC und dem TS Adapter ermöglicht.

Wenn Sie eine VPN-Verbindung zu mehreren entfernten Netzwerken aufbauen möchten, benötigen Sie für jedes Netzwerk einen eigenen TS Adapter IE Advanced als Schnittstelle. Die über den TS Adapter verbundenen Netzwerke können intern identische IP-Adressen verwenden. Lediglich die externen IP-Adressen (WAN) der jeweiligen TS Adapter müssen sich unterscheiden.

Es ist immer nur eine VPN-Verbindung zu einem TS Adapter gleichzeitig möglich.

Hinweis

Authentifizierung

Beachten Sie bitte, dass das CA-Zertifikat ausschließlich den TS Adapter IE Advanced authentifiziert.

Ein Benutzer wird, wie bei Wählverbindungen auch, über das Login (Benutzername und Passwort) authentifiziert.

Verwenden Sie daher nur sichere Passwörter für das Login.

Siehe hierzu auch

- VPN-Verbindung aufbauen (Seite 171)
- VPN-Verbindung trennen (Seite 173)

7.4.2 Grundlagen zu CA-Zertifikaten

Einführung

Um mit TeleService eine sichere VPN-Verbindung aufbauen zu können, ist es erforderlich, dass Sie bei der Konfiguration des TS Adapter IE Advanced ein CA-Zertifikat mit einem eindeutigen Fingerabdruck generieren.

Dieses CA-Zertifikat können Sie wie folgt auf jedem PC installieren, den Zugriff auf den TS Adapter IE Advanced erhalten soll:

- Über den automatischen Download im Verbindungsdialog von TeleService, indem Sie dort den zum CA-Zertifikat zugehörigen Fingerabdruck eintragen
- Über manuelle Installation mit Hilfe der Microsoft® Management Console.

Definition CA-Zertifikat

Ein CA-Zertifikat ist ein durch eine Zertifizierungsstelle ("certificate authority" oder "certification authority", nachfolgend "CA" genannt) herausgegebenes digitales Zertifikat. Im Fall des TS Adapter IE Advanced werden selbst-signierte Zertifikate verwendet, die Zertifizierungsstelle ist in diesem Fall der TS Adapter IE Advanced selbst.

Vom CA-Zertifikat werden die Zertifikate für "SSTP" (Secure Socket Tunneling Protocol) und "HTTPS" (Hypertext Transfer Protocol Secure) abgeleitet.

CA-Zertifikate enthalten einen "Schlüssel“ und Zusatzinformationen, die zur Authentifizierung sowie zur Verschlüsselung und Entschlüsselung vertraulicher Daten dienen. Als Zusatzinformationen sind zum Beispiel Gültigkeitsdauer, Verweise auf Zertifikatsperrlisten, etc. enthalten, die durch die CA mit in das Zertifikat eingebracht werden.

Verwendung von CA-Zertifikaten bei TeleService

Um den TS Adapter IE Advanced als Verbindungspartner gegenüber dem Remote-PC eindeutig zu identifizieren, wird vom TS Adapter ein CA-Zertifikat mit einem eindeutigen Fingerabdruck generiert.

Für den Aufbau einer VPN-Verbindung ist es zwingend erforderlich, dieses CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC zu hinterlegen. Wenn Sie den Web-Server im

Direktanschluss aufrufen, so wird bei fehlendem CA-Zertifikat eine Sicherheitswarnung angezeigt, die akzeptiert werden kann.

Hinweis

Hantierung von CA-Zertifikaten

Die Hantierung von CA-Zertifikaten erfordert spezielle Kenntnisse des Betriebssystems und sollte nur von entsprechend geschultem Personal durchgeführt werden!

Zum Verwalten von CA-Zertifikaten benötigen Sie Administrator-Rechte.

Definition Fingerabdruck

Der Fingerabdruck ist ein 20 Bytes langer hexadezimaler Ausdruck. Er stellt einen eindeutigen Wert für ein CA-Zertifikat dar und wird dazu verwendet, ein bestimmtes CA-Zertifikat zu identifizieren.

Der Fingerabdruck wird dynamisch mit dem SHA1-Algorithmus berechnet und ist physisch im CA-Zertifikat nicht vorhanden.

Verwendung des Fingerabdrucks bei TeleService

Das CA-Zertifikat dient dazu, den TS Adapter IE Advanced eindeutig als Verbindungspartner zu identifizieren. Bei jeder CA-Zertifikatsgenerierung vom TS Adapter IE Advanced wird automatisch ein 20 Bytes langer, eindeutiger Fingerabdruck dieses CA-Zertifikats generiert. Er wird bei der Zertifikatsgenerierung vom TS Adapter IE Advanced automatisch berechnet. Jedes Zertifikat besitzt einen spezifischen, eindeutigen Fingerabdruck. Dieser Fingerabdruck muss an Ihren Rechner über einen sicheren Weg übergeben werden, z. B. per Telefon oder durch eine verschlüsselte Mail. Diesen Fingerabdruck müssen Sie beim Aufbau einer VPN-Verbindung mit TeleService im Verbindungsdialog eintragen, sofern das CA-Zertifikat nicht bereits auf Ihrem PC im Windows-Zertifikatsspeicher hinterlegt ist.

Den zu Ihrem TS Adapter IE Advanced zugehörigen Fingerabdruck finden Sie im Web-Interface zum TS Adapter IE Advanced. Doppelklicken Sie zum Öffnen des Web-Interface auf den Befehl "TS Adapter IE parametrieren" in der Teilnehmerliste im TIA Portal. Führen Sie das Web-Login durch, um den Fingerabdruck im Register "Sicherheit > Zertifikate" anzuzeigen.

Download des CA-Zertifikats beim Verbindungsaufbau

Beim Verbindungsaufbau überprüft TeleService, ob ein passendes CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC installiert ist. Wird ein passendes CA-Zertifikat gefunden, so wird die VPN-Verbindung als SSTP-Verbindung (Secure Socket Tunneling Protocol) aufgebaut.

Wird kein passendes CA-Zertifikat gefunden, so wird zunächst das CA-Zertifikat vom entsprechenden TS Adapter IE Advanced geladen. Dabei wird der TS Adapter IE Advanced über die im Verbindungsdialog eingetragene Remote-Adresse aufgerufen. Ist dieser Download des CA-Zertifikats erfolgreich, so wird der Fingerabdruck des heruntergeladenen CA-Zertifikats errechnet und mit dem im Verbindungsdialog eingetragenen Fingerabdruck verglichen. Bei Übereinstimmung beider Fingerabdrücke öffnet sich ein Dialog und Sie werden gefragt, ob das CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC gespeichert werden soll. Für das Speichern des CA-Zertifikates benötigen Sie Administrator-Rechte.

Anschließend wird die VPN-Verbindung aufgebaut.

Siehe hierzu auch

- CA-Zertifikat für VPN-Verbindung installieren (Seite 168)
- CA-Zertifikat für VPN-Verbindung löschen (Seite 171)

7.4.3 CA-Zertifikat für VPN-Verbindung installieren

Installieren von CA-Zertifikaten

Um mit TeleService eine sichere VPN-Verbindung zwischen Ihrem PG/PC und einer entfernten Anlage aufzubauen, benötigen Sie ein vom TS Adapter IE Advanced erstelltes gültiges CA-Zertifikat. Dieses muss im Windows-Zertifikatsspeicher Ihres PG/PC hinterlegt sein.

Die Installation eines CA-Zertifikates kann über einen automatischen Download oder über manuelle Installation erfolgen.

Die Verwaltung der CA-Zertifikate erfolgt über die Microsoft® Management Console.

Hinweis

Hantierung von CA-Zertifikaten

Die Hantierung von CA-Zertifikaten erfordert spezielle Kenntnisse des Betriebssystems und sollte nur von entsprechend geschultem Personal durchgeführt werden!

Zum Verwalten von CA-Zertifikaten benötigen Sie Administrator-Rechte!

Voraussetzung

Auf Ihrem Rechner ist noch kein CA-Zertifikat im Windows-Zertifikatsspeicher installiert.

Installation von CA-Zertifikaten über automatischen Download

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Übertragen Sie den zum CA-Zertifikat zugehörigen Fingerabdruck über einen "sicheren Weg", z. B. per Telefon oder über eine verschlüsselte Mail, vom TS Adapter IE Advanced auf Ihren Rechner.
   Den zu Ihrem TS Adapter IE Advanced zugehörigen Fingerabdruck finden Sie im Web-Interface zum TS Adapter IE Advanced. Doppelklicken Sie zum Öffnen des Web-Interface auf den Befehl "TS Adapter IE parametrieren" in der Teilnehmerliste im TIA Portal. Führen Sie das Web-Login durch, um den Fingerabdruck im Register "Sicherheit > Zertifikate" anzuzeigen.

3. Klicken Sie in der Projektnavigation des TIA Portals auf den Ordner "Online-Zugänge".
4. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
5. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
6. Wählen Sie in der Klappliste als Adapter-Typ den "TS Adapter IE" aus.
7. Wählen Sie als Verbindungs-Typ "VPN" aus.
8. Tragen Sie die IP-Adresse bzw. den DNS-Namen für den zu kontaktierenden TS Adapter IE Advanced in das entsprechende Eingabefeld ein. Alternativ können Sie evtl. bereits vorhandene Daten aus dem Telefonbuch übernehmen, indem Sie auf die dahinterliegende Schaltfläche klicken.
9. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
10. Kopieren Sie den im Web-Interface zum TS Adapter IE Advanced angezeigten Fingerabdruck in die Spalte "Fingerabdruck".
11. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben.
12. Da noch kein CA-Zertifikat auf Ihrem Rechner installiert war, wird kein gültiges CA-Zertifikat gefunden.
    Daher wird eine "normale" Verbindung (keine VPN-Verbindung) aufgebaut und das benötigte Zertifikat wird vom TS Adapter IE Advanced zu Ihrem Rechner in den Arbeitsspeicher heruntergeladen. Danach wird der Fingerabdruck berechnet (Algorithmus SHA1) und mit dem im Verbindungsdialog eingetragenen Fingerabdruck verglichen. Bei Übereinstimmung beider Fingerabdrücke öffnet sich ein Dialog und Sie werden gefragt, ob das CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC gespeichert werden soll.
13. Bestätigen Sie das Speichern des CA-Zertifikats.

Hinweis

Neustart von STEP 7 nach Erneuern von CA-Zertifikat erforderlich

Nachdem das CA-Zertifikat auf dem TS Adapter IE Advanced erneuert wurde, muss STEP 7 neu gestartet werden, bevor die Fernverbindung mit Hilfe des neuen Fingerabdrucks aufgebaut werden kann. Das CA-Zertifikat kann sonst unter Umständen nicht im Windows-Zertifikatsspeicher gespeichert werden und die Fernverbindung kann nicht aufgebaut werden.

Ergebnis

Die VPN-Verbindung zum gewünschten TS Adapter IE Advanced wird aufgebaut. Sobald die Verbindung aufgebaut wurde, wird der Dialog geschlossen.

Manuelle Installation von CA-Zertifikaten

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Öffnen Sie die Windows-Zertifikatsverwaltung auf Ihrem PG/PC mit Hilfe der Microsoft® Management Console.
   Klicken Sie dazu auf "Start", tragen Sie im Suchfeld "mmc" ein und drücken Sie die EINGABETASTE.
   Die Konsole öffnet sich.
3. Klicken Sie im Menü "Datei" auf den Befehl "Snap-In hinzufügen/entfernen...".
   Das Dialogfeld "Snap-In hinzufügen oder entfernen" öffnet sich.
4. Doppelklicken Sie in der Liste "Snap-In" auf "Zertifikate" und wählen Sie im anschließenden Dialog "Computerkonto" aus.
5. Wählen Sie in folgenden Dialog "Lokaler Computer" aus und klicken Sie auf "Fertig stellen" und auf "OK".
   Der Konsolenstamm öffnet sich und zeigt den Ordner "Zertifikate (Lokaler Computer)" an.
6. Öffnen Sie den angezeigten Ordner "Zertifikate (Lokaler Computer)" und klicken Sie auf "Vertrauenswürdige Stammzertifizierungsstellen".
7. Klicken Sie auf den Ordner "Zertifikate" und rufen Sie über das Kontextmenü den Befehl "Alle Aufgaben" > "Importieren..." auf.
8. Beachten Sie die im Dialog "Zertifikat - Import-Assistent" angezeigten Informationen und klicken Sie auf "Weiter".
9. Klicken Sie im nachfolgenden Dialog auf "Durchsuchen ..." und wählen Sie das gewünschte CA-Zertifikat aus.
10. Klicken Sie anschließend zwei Mal auf "Weiter" und anschließend auf "Fertig stellen", um das CA-Zertifikat zu installieren.

Ergebnis

Das ausgewählte CA-Zertifikat wird an der angegebenen Stelle im Windows-Zertifikatsspeicher installiert.

Hinweis

Weiterführende Informationen ...

... zur Installation von CA-Zertifikaten finden Sie über die Taste "F1" in der Online-Hilfe zu Ihrem Betriebssystem.

7.4.4 CA-Zertifikat für VPN-Verbindung löschen

Löschen von CA-Zertifikaten

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Öffnen Sie die Windows-Zertifikatsverwaltung auf Ihrem PG/PC mit Hilfe der Microsoft® Management Console.
   Klicken Sie dazu auf "Start", tragen Sie im Suchfeld "mmc" ein und drücken Sie die EINGABETASTE.
   Die Konsole öffnet sich.
3. Klicken Sie im Menü "Datei" auf "Snap-In hinzufügen/entfernen..."
   Das Dialogfeld zur Snap-In-Auswahl öffnet sich.
4. Doppelklicken Sie in der Liste "Snap-In" auf "Zertifikate" und wählen Sie im anschließenden Dialog "Computerkonto" aus.
5. Wählen Sie in folgenden Dialog "Lokaler Computer" aus und klicken Sie auf "Fertig stellen" und auf "OK".
   Der Konsolenstamm öffnet sich und zeigt den Ordner "Zertifikate (Lokaler Computer)" an.
6. Öffnen Sie den angezeigten Ordner "Zertifikate (Lokaler Computer)" und klicken Sie auf "Vertrauenswürdige Stammzertifizierungsstellen".
7. Öffnen Sie den Ordner "Zertifikate", wählen Sie das gewünschte CA-Zertifikat aus und klicken Sie im Kontextmenü auf "Löschen".
8. Bestätigen Sie die nachfolgende Abfrage mit "Ja".

Ergebnis

Das ausgewählte CA-Zertifikat wird aus der Liste der verfügbaren Zertifikate gelöscht.

7.4.5 Aufbauen einer VPN-Verbindung zu einer entfernten Anlage

7.4.5.1 VPN-Verbindung aufbauen

Einführung zum Aufbauen einer VPN-Verbindung

Eine VPN-Verbindung kommt zu Stande, wenn Sie sich mit TeleService über das Internet in eine entfernte Anlage verbinden.

Hierzu ist auf der einen Seite Ihr PG/PC mit installiertem TIA Portal mit dem Internet verbunden. An der anderen Seite ist das Automatisierungssystem über die WAN-Schnittstelle (Wide Area Network) des parametrierten TS Adapters IE Advanced mit dem Internet verbunden.

Voraussetzung

Ihr PG/PC ist mit dem Internet verbunden.

In der entfernten Anlage befindet sich ein TS Adapter IE Advanced.

Der TS Adapter IE Advanced ist parametriert und mit dem Internet verbunden.

Das zur Identifizierung des TS Adapters benötigte CA-Zertifikat wurde erzeugt und ist im Windows-Zertifikatsspeicher Ihres PG/PC installiert.

Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation des TIA Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste als "Adapter-Typ" den "TS Adapter IE" aus.
5. Tragen Sie als "Verbindungs-Typ" "VPN" ein.
6. Tragen Sie die IP-Adresse bzw. den DNS-Namen für den zu kontaktierenden TS Adapter IE Advanced in das entsprechende Eingabefeld ein. Alternativ können Sie evtl. bereits vorhandene Daten aus dem Telefonbuch übernehmen, indem Sie auf die dahinterliegende Schaltfläche klicken.
7. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
8. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte VPN-Verbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau der Fernverbindung benötigten Parameter eingetragen haben.

Ergebnis

Die VPN-Verbindung zur gewünschten Anlage wird aufgebaut. Im "Status" wird der Fortschritt des Verbindungsaufbaus angezeigt. Sobald die VPN-Verbindung zu Stande gekommen ist, wird der Dialog geschlossen. In der Statuszeile des TIA Portals erscheint die Meldung: "Fernverbindung ist aufgebaut". Sie können nun die Fernverbindung mit dem TIA Portal nutzen und mit dem Automatisierungssystem kommunizieren.

Verbindungsaufbau nicht möglich

Ist kein Verbindungsaufbau möglich, so versuchen Sie bitte, die Fehlerursache mithilfe der "Hinweise zur Fehlersuche" zu ermitteln.

Hinweis

Regeln für IP-Adressen

- Verwenden Sie ausschließlich IP-Adressen, die im Anlagennetz noch nicht vergeben sind.
- Falls die für den TS Adapter IE Advanced parametrierte IP-Adresse im Anlagennetz bereits vergeben ist, so ist der TS Adapter IE Advanced nur über seine MAC-Adresse ansprechbar.

Siehe auch

CA-Zertifikat für VPN-Verbindung installieren (Seite 168)

7.4.5.2 VPN-Verbindung trennen

Trennen einer aktiven VPN-Verbindung

Hinweis
Trennen der VPN-Verbindung
Bevor Sie die VPN-Verbindung trennen, sollten Sie im TIA Portal offline schalten.

Gehen Sie vor wie folgt:

1. Doppelklicken Sie im TIA Portal auf den Eintrag "Fernverbindung aufbauen/trennen".
2. Bestätigen Sie die Abfrage im nachfolgenden Dialog mit "Ja".

Ergebnis

Die VPN-Verbindung wird getrennt.

7.4.6 TS Adapter IE Advanced

7.4.6.1 Kurzbeschreibung TS Adapter IE Advanced

TS Adapter IE Advanced

Der TS Adapter IE Advanced hat folgende Eigenschaften:

- Direktanschluss über Industrial Ethernet (IE), 2 Ports
- WAN-Schnittstelle (Wide Area Network) für VPN-Verbindungen
- Firmwareupdate möglich
- Steckbare Module
- Die Parametrierung erfolgt über ein Web-Interface.

Hinweis

Weitere Informationen zum TS Adapter IE Advanced

Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

7.4.6.2 Anschlussarten

Anschlussarten des TS Adapter IE Advanced

Die folgenden Bilder zeigen die möglichen Anschlussarten für den TS Adapter IE Advanced.

Direktanschluss

Im Direktanschluss am PG/PC können Sie den TS Adapter IE Advanced über Ethernet parametrieren.

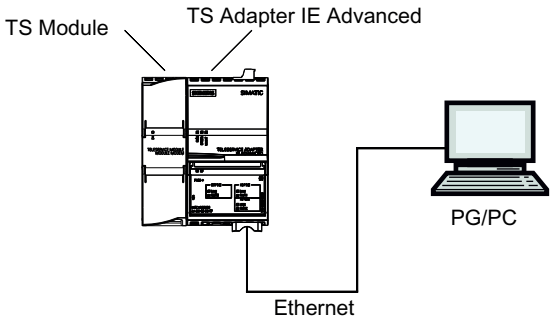

Bild 7-5 TS Adapter IE Advanced - Direktanschluss

Anschluss an das Internet (DSL-Modem/Router)

Zum Anschluss an das Internet müssen Sie das DSL-Modem/Router am WAN-Anschluss des TS Adapter IE Advanced betreiben.

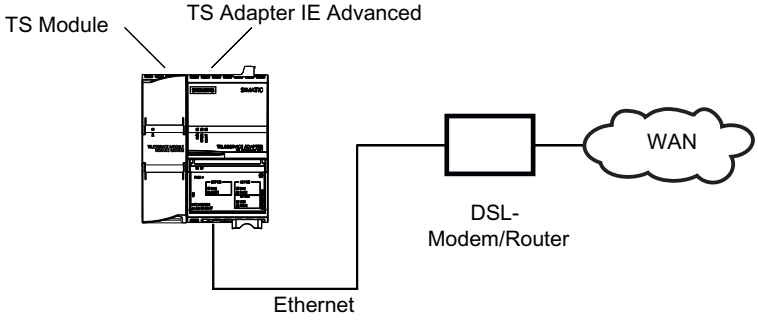

Bild 7-6 TS Adapter IE Advanced - Anschluss Internet

Anschluss an das Firmennetz (Intranet)

Zum Anschluss an das Intranet müssen Sie das Anlagennetz (Ethernet) an einem der beiden LAN-Anschlüsse des TS Adapter IE Advanced betreiben.

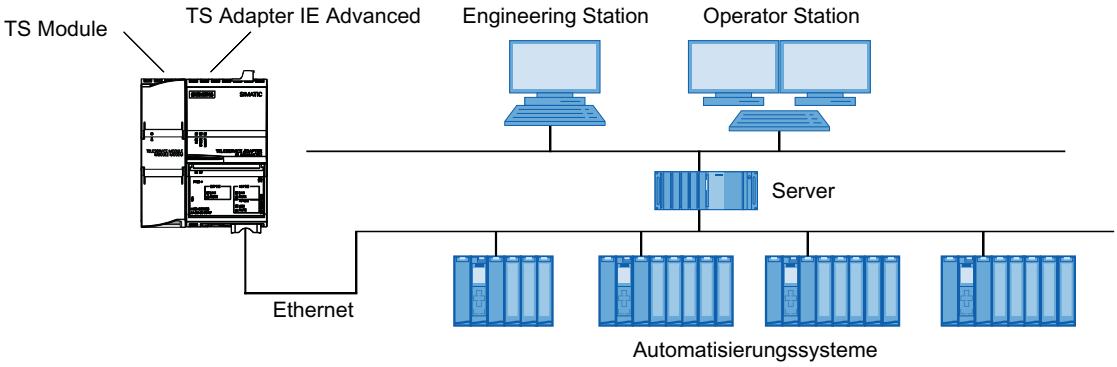

Bild 7-7 TS Adapter IE Advanced - Anschluss Firmennetz (Intranet)

Weitere Informationen

Weitere Informationen zu den TS Modulen finden Sie in der zu Ihrem TS Adapter IE Advanced mitgelieferten Dokumentation.

7.4.6.3 Parametriermöglichkeiten für den TS Adapter IE Advanced

Grundlagen zum Parametrieren des TS Adapter IE

Der TS Adapter IE Advanced wird über ein Web-Interface parametriert.

Für die Parametrierung des TS Adapter IE Advanced steht Ihnen eine dem Web-Interface zugeordnete Webhilfe zur Verfügung.

Folgende Parametriermöglichkeiten stehen z. B. zur Verfügung:

- Neuparametrierung
- Wiederherstellen der Standard-Parametrierung
- Importieren von Adapterparametern
- Exportieren von Adapterparametern

Hinweis

Parametrierung durchführen

Parametrieren Sie Ihren TS Adapter IE Advanced unter Berücksichtigung der mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

7.4.6.4 TS Adapter IE Advanced parametrieren

Einführung

Sie können den TS Adapter IE Advanced sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung parametrieren.

Nachfolgend werden beide Parametriermöglichkeiten beschrieben.

Spezifische Details zur Parametrierung des TS Adapters IE Advanced entnehmen Sie bitte der Dokumentation zum TS Adapter IE Advanced.

Parametrieren des TS Adapter IE Advanced im Direktanschluss

Voraussetzung

Es besteht eine LAN-Verbindung zu Ihrem TS Adapter IE Advanced.

Vorgehen

Um den TS Adapter IE Advanced zu parametrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Online Zugänge".
2. Doppelklicken Sie hier auf die Ethernet-Schnittstelle Ihres Rechners.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer anzeigen". Daraufhin wird der TS Adapter IE Advanced angezeigt.
4. Doppelklicken Sie auf den Ordner "TS Adapter IE Advanced" und anschließend auf "Online und Diagnose" und weisen Sie dem TS Adapter in den nachfolgenden Dialogen die gewünschte IP-Adresse zu. Beachten Sie dabei, dass die IP-Adresse der Schnittstellenkarte des PG/PC im selben Subnetz liegt, wie die IP-Adresse, die Sie für den TS Adapter IE Advanced vergeben.
5. Aktualisieren Sie die Ansicht in der Projektnavigation für die "Erreichbaren Teilnehmer", damit der TS Adapter IE Advanced mit der neu zugewiesenen IP-Adresse angezeigt wird.
6. Öffnen Sie den Ordner "TS Adapter IE Advanced" in der Teilnehmerliste.
7. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugeordnete Web-Interface zum Parametrieren des TS Adapters.
8. Führen Sie das "Logon" für das Web-Interface durch.
9. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
10. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE Advanced gespeichert. Die Parametrierung ist damit abgeschlossen.

Parametrieren des TS Adapter IE Advanced über Fernverbindung

Voraussetzung

Es besteht eine aufgebaute Fernverbindung zu einem TS Adapter IE Advanced.

Vorgehen

1. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Online Zugänge".
2. Öffnen Sie den Ordner "TeleService" und danach den gewünschten Anlagenordner.
3. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugehörige Web-Interface zum Parametrieren des TS Adapters IE. Das "Logon" für das Web-Interface wird automatisch mit den Login-Daten der Fernverbindung durchgeführt.
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE Advanced gespeichert. Die Parametrierung ist damit abgeschlossen.

7.5 CPU-gesteuerte Fernverbindungen bei TeleService

7.5.1 Übersicht über CPU-gesteuerte Fernverbindungen

Einführung

TeleService bietet Ihnen, abhängig von der jeweils verwendeten CPU, unterschiedliche Möglichkeiten für den Aufbau einer Fernverbindung. Die Initiative für den Verbindungsaufbau geht dabei jeweils von der CPU aus. Für die einzelnen Verbindungsmöglichkeiten werden die nachfolgend genannten Kommunikationsanweisungen genutzt.

Möglichkeiten zum Verbindungsaufbau mit S7-300/400er CPUs

Folgende Kommunikationsanweisungen stehen zur Verfügung:

- Kommunikationsanweisung "PG_DIAL": Fernverbindung zu PG/PC aufbauen
- Kommunikationsanweisung "SMS_SEND": SMS-Nachricht senden
- Kommunikationsanweisung "AS_DIAL": Fernverbindung zum AS aufbauen
- Kommunikationsanweisung "AS_MAIL": E-Mail übertragen

Möglichkeiten zum Verbindungsaufbau mit S7-1200er CPUs

Folgende Kommunikationsanweisung steht zur Verfügung:

- Kommunikationsanweisung "TM_MAIL": E-Mail übertragen

Möglichkeiten zum Verbindungsaufbau mit S7-1500er CPUs

Folgende Kommunikationsanweisung steht zur Verfügung:

- Kommunikationsanweisung "TMAIL_C": E-Mail übertragen

Hinweis

Beschreibung der einzelnen Kommunikationsanweisungen

Weiterführende Informationen zu den verwendbaren Kommunikationsanweisungen finden Sie im Informationssystem des TIA Portals im Verzeichnis "Referenzen > Kommunikation > TeleService".

Siehe auch

Parametriermöglichkeiten für den TS Adapter IE (Seite 160)

7.5.2 Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung)

7.5.2.1 Zugriff von entfernten Anlagen auf ein PG/PC

Einführung

Sie können mit der Applikation TeleService und einem TS Adapter MPI eine Fernverbindung zu einer entfernten Anlage aufbauen und so mit dieser Anlage kommunizieren. Die Initiative zum Aufbau der Fernverbindung geht hierzu vom PG/PC aus.

Häufig treten jedoch Ereignisse auf einer entfernten Anlage auf, die einen schnellen Eingriff erfordern. In diesem Fall ist es dem Automatisierungssystem möglich, bei Auftreten eines asynchronen Ereignisses den Aufbau einer Fernverbindung zu einem PG/PC zu initiieren.

Das folgende Bild zeigt eine grafische Darstellung der Komponenten, die für den Verbindungsaufbau von einer Anlage zu einem PG/PC erforderlich sind.

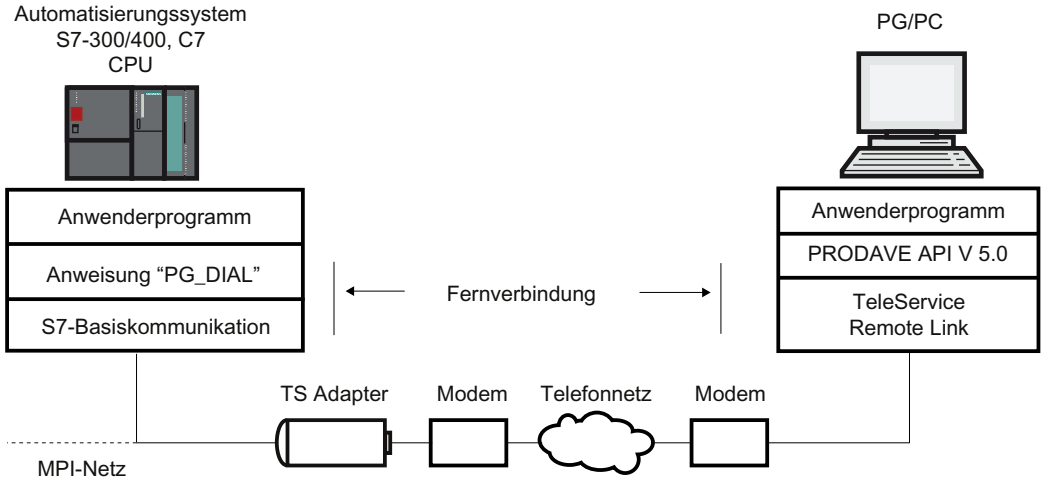

Bild 7-8 Funktion der Kommunikationsanweisung "PG_DIAL"

7.5.2.2 Voraussetzungen für den Verbindungsaufbau

Einführung

Soll eine entfernte Anlage eine Fernverbindung zu einem PG/PC aufbauen können, müssen bestimmte Hardware- und Software-Voraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

Hardware-Voraussetzungen:

Zum Aufbauen einer Fernverbindung von einer entfernten Anlage zu einem PG/PC wird keine weitere Hardware benötigt als die, die auch für den Zugriff des PG/PC auf die entfernte Anlage erforderlich ist.

Zum Aufbauen der Verbindung ruft Ihr Anwenderprogramm die Kommunikationsanweisung "PG_DIAL" auf. Diese ist auf einer S7-300 oder S7-400 CPU ablauffähig, auf der die S7-Basiskommunikation realisiert ist.

Es muss ein TS Adapter I mit einer Version größer oder gleich V5.0 oder ein TS Adapter II verwendet werden.

Software-Voraussetzungen auf Anlagen-Seite:

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "PG_DIAL" enthalten, die beim Installieren des TIA Portals mit installiert wird. Sie finden die mitinstallierten Kommunikationsanweisungen in der Task Card des Bausteineditors im Ordner "Kommunikation > TeleService".

Soll eine entfernte Anlage eine Fernverbindung zu einem PG/PC aufbauen, muss das Anwenderprogramm der Anlage die Kommunikationsanweisung "PG_DIAL" aufrufen.

Software-Voraussetzungen auf PG/PC-Seite:

Auf dem PG/PC benötigen Sie ein Software-Produkt, das in Verbindung mit TeleService auf den Anruf einer entfernten Anlage wartet, diesen erkennt und Ihr Anwenderprogramm informiert.

7.5.3 Datenaustausch zwischen entfernten Anlagen (AS-AS-Fernkopplung)

7.5.3.1 Grundlagen zur AS-AS-Fernkopplung

Einführung

Mithilfe der AS-AS-Fernkopplung können zwei Automatisierungssysteme über das Telefonnetz Prozessdaten miteinander austauschen.

Voraussetzung

Die Kommunikationsanweisung "AS_DIAL" steht Ihnen nur dann zur Verfügung, wenn Sie eine CPU aus der Familie S7-300/400 verwenden.

Definition: Lokales und entferntes Automatisierungssystem

- Das Automatisierungssystem, von dem die Initiative zum Aufbau der Fernverbindung ausgeht, wird als **lokal** bezeichnet.
- Das Automatisierungssystem, zu dem die Fernverbindung aufgebaut werden soll, wird als **entfernt** bezeichnet.

Datenaustausch über AS-AS-Fernkopplung

Der Datenaustausch erfolgt über spezifische Kommunikationsanweisungen für nicht projektierte S7-Verbindungen. Für den Aufbau einer Fernverbindung zum Automatisierungssystem nutzen Sie die Kommunikationsanweisung "AS_DIAL".

Weiterführende Informationen zum Verbindungsaufbau finden Sie im Informationssystem im Verzeichnis "Referenzen > Kommunikation > TeleService".

Das folgende Bild zeigt eine grafische Darstellung der Komponenten, die für den Verbindungsaufbau von einem lokalen zu einem entfernten Automatisierungssystem erforderlich sind.

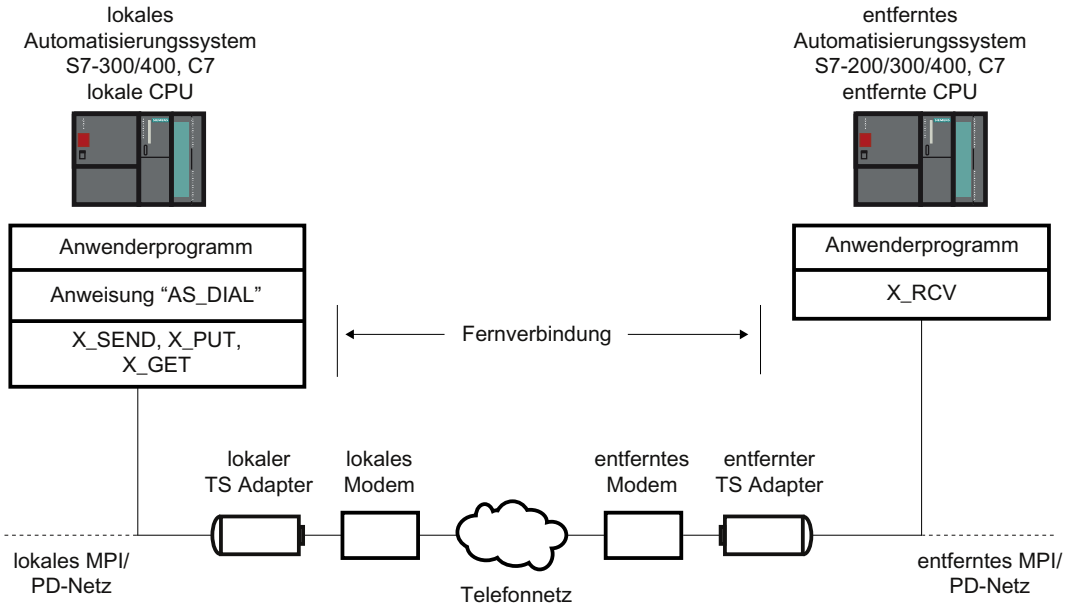

Bild 7-9 Datenaustausch über AS-AS-Fernkopplung

7.5.3.2 Hardware- und Softwarevoraussetzungen für AS-AS-Fernkopplung

Einführung

Soll ein lokales Automatisierungssystem eine Fernverbindung zu einem entfernten Automatisierungssystem aufbauen können, müssen bestimmte Hardware- und Software-Voraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

Hardware-Voraussetzungen

Zur Übertragung von Prozessdaten von einem lokalen zu einem entfernten Automatisierungssystem wird keine weitere Hardware benötigt als die, die auch für den Zugriff des PG/PC auf das jeweilige Automatisierungssystem erforderlich ist.

Zum Auf- und Abbauen der Fernverbindung ruft Ihr das Anwenderprogramm des TIA Portals der lokalen CPU eine Kommunikationsanweisung auf. Diese Kommunikationsanweisung ist auf einer S7-300/400-CPU oder auch C7-CPU ablauffähig. Diese Kommunikationsanweisung

setzt voraus, dass auf der CPU die S7-Basiskommunikation realisiert ist. Ebenso muss die entfernte CPU die S7-Basiskommunikation unterstützen.

Es muss ein TS Adapter I mit einer Version größer oder gleich V5.1 oder ein TS Adapter II verwendet werden.

Software-Voraussetzungen

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "AS_DIAL" enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA Portals aufgenommen wird. Zum Auf- und Abbauen einer Fernverbindung von einem lokalen Automatisierungssystem zu einem entfernten Automatisierungssystem ist im Anwenderprogramm des TIA Portals der lokalen CPU die Kommunikationsanweisung "AS_DIAL" aufzurufen.

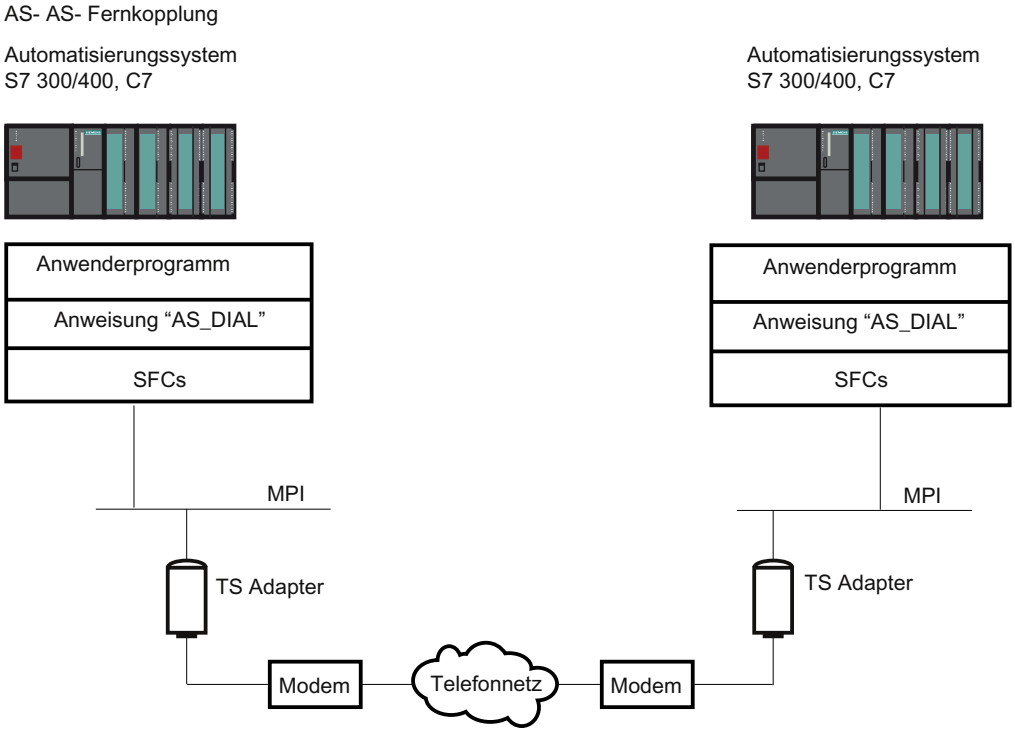

Bild 7-10 Hardware- und Softwarevoraussetzungen für AS-AS-Fernkopplung

7.5.4 Senden einer SMS von einer Anlage

7.5.4.1 Voraussetzungen für das Senden einer SMS

Einführung

Soll eine Anlage eine SMS senden, so müssen bestimmte Hardware- und Software-Voraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

Hardware-Voraussetzungen

Zum Senden einer SMS von einer Anlage benötigen Sie ein GSM-Funkmodem sowie einen TS Adapter MPI.

Es muss ein TS Adapter I mit einer Version größer oder gleich V5.2 oder ein TS Adapter II verwendet werden.

Software-Voraussetzungen auf Anlagen-Seite

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "SMS_SEND" enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA Portals aufgenommen wird. Soll eine Anlage eine SMS senden, muss das Anwenderprogramm der Anlage die Kommunikationsanweisung "SMS_SEND" aufrufen.

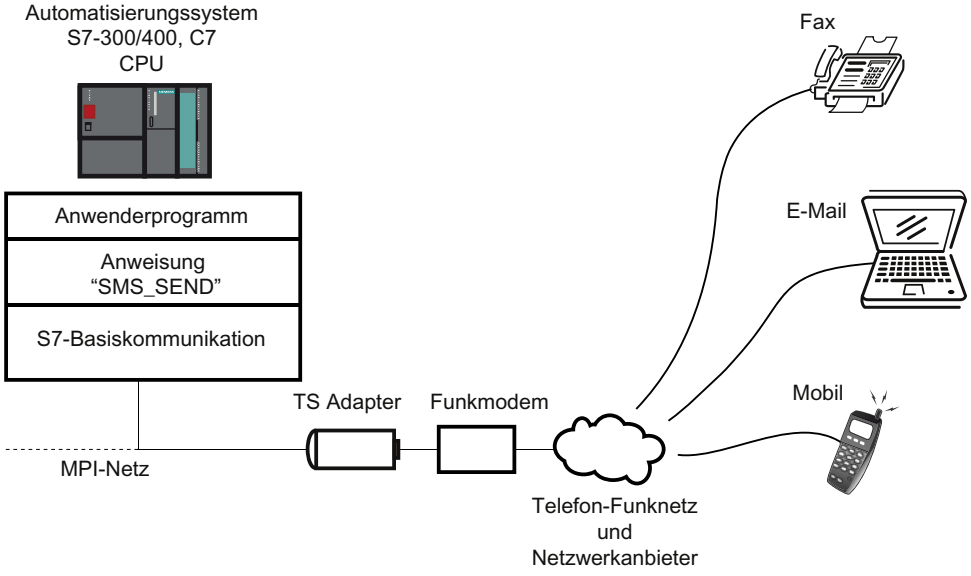

Bild 7-11 Funktionsweise der Kommunikationsanweisung "SMS_SEND"

Hinweis

Über Zusatzdienste des Mobilfunkanbieters kann eine SMS ggf. nicht nur an ein Mobiltelefon, sondern auch an eine E-Mail-Adresse oder an ein Fax-Gerät gesendet werden.

7.5.5 Senden einer E-Mail von einer Anlage

7.5.5.1 Voraussetzungen für das Senden einer E-Mail

Einführung

Soll eine Anlage eine E-Mail senden, so müssen folgende Hard- und Software-Voraussetzungen erfüllt sein:

Hardware-Voraussetzungen

Zum Senden einer E-Mail von einer Anlage benötigen Sie einen TS Adapter IE und eine der nachfolgenden genannten CPUs:

- eine CPU 31x-2 PN/DP ab Firmwarestand V2.5
- eine CPU 41x-3 PN/DP
- eine CPU aus der Familie S7-1200
- eine CPU aus der Familie S7-1500

Software-Voraussetzungen auf Anlagen-Seite

Im Lieferumfang von TeleService sind CPU-abhängig verschiedene Kommunikationsanweisungen enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA Portals aufgenommen werden.

Soll eine Anlage eine E-Mail senden, so muss das Anwenderprogramm der Anlage die entsprechende CPU-abhängige Kommunikationsanweisung aufrufen.

Nachfolgende Kommunikationsanweisungen stehen zum Senden einer E-Mail zur Verfügung:

- S7-300/400er CPU: Verwendet die Kommunikationsanweisung "AS_MAIL": E-Mail übertragen
- S7-1200er CPU V2.x und V3.x: Verwendet die Kommunikationsanweisung "TM_MAIL": E-Mail übertragen
- S7-1200er CPU >V4.0: Verwendet die Kommunikationsanweisung "TMAIL_C": E-Mail übertragen
- S7-1500er CPU: Verwendet die Kommunikationsanweisung "TMAIL_C": E-Mail übertragen

Die jeweilige Kommunikationsanweisung überträgt mittels Simple Mail Transfer Protocol (SMTP) mit dem Authentifizierungsverfahren "LOGIN" eine E-Mail von einer CPU zu einem Mail-Server. Bei diesem SMTP-Verfahren werden die Daten unverschlüsselt übertragen.

Das nachfolgende Bild zeigt ein Beispiel mit der Kommunikationsanweisung "AS_MAIL":

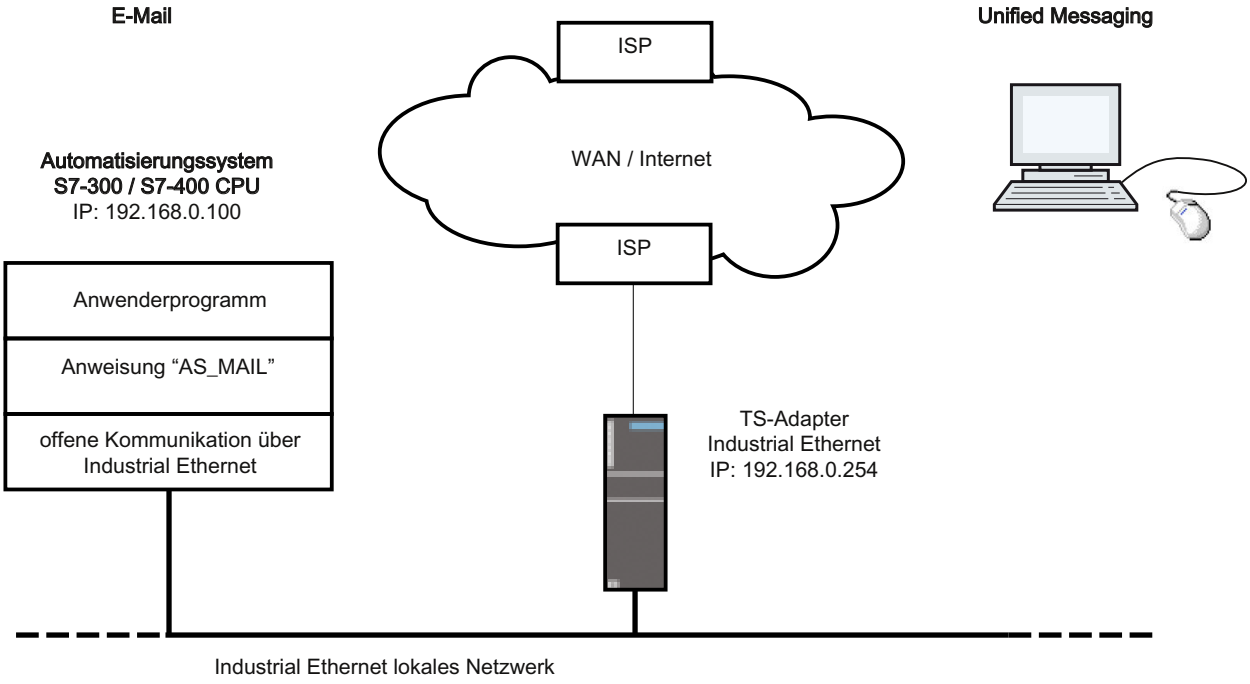

Darüber hinaus muss bei der Konfiguration der CPU, auf der die Kommunikationsanweisung "AS_MAIL" abläuft, für die Ethernet-Schnittstelle die Eigenschaft "Netzübergang / Router verwenden" eingestellt werden. (Zu finden in der Gerätekonfiguration unter Ethernet-Adressen und dort unter IP-Protokoll,) Als "Adresse" ist dabei die IP-Adresse der Ethernet-Schnittstelle des TS Adapters IE anzugeben.

Hinweis

Weiterführende Hinweise hierzu finden Sie in der Task Card im Ordner "Kommunikationsanweisungen" unter TeleService.

7.6 Hinweise zur Fehlersuche

7.6.1 Allgemeine Hinweise für die Fehlersuche bei Modem-Problemen

Einführung

Die nachfolgenden Hinweise erleichtern es Ihnen, die Ursachen für eventuell auftretende Modem-Probleme zu finden und zu beheben:

1. Schalten Sie das "Aufzeichnen einer Protokolldatei" zum Datenverkehr zwischen PG/PC und Modem ein. Die Einträge in dieser Datei können wertvolle Hinweise bei der Ermittlung von Fehlerursachen liefern.
2. Schalten Sie an Ihrem lokalen Modem den Lautsprecher ein. Wählen Sie eine gut hörbare Lautstärke.
   Sie können dann hören, ob:
   – ein Freizeichen auf dem Anschluss vorhanden ist.
   – das angerufene Modem besetzt ist.
   – das angerufene Modem den Anruf entgegennimmt.

Häufige Modem-Probleme

Zu den häufigsten Modem-Problemen gehören Probleme bei der Modemverbindung:

- Modemverbindung wird nicht aufgebaut.
- Modemverbindung wird unterbrochen.

In den folgenden Topics finden Sie Tabellen, in denen mögliche Ursachen beschrieben und Hinweise zur Behebung des jeweiligen Fehlerfalls angegeben sind.

Siehe auch

Wählverbindung zum TS Adapter wird nicht aufgebaut (Seite 187)

Wählverbindung vom TS Adapter wird nicht aufgebaut (Seite 189)

Modemverbindung wird unterbrochen (Seite 190)

Meldungen des Modems (Seite 191)

Aufzeichnen einer Protokolldatei für das Modem (Seite 186)

7.6.2 Aufzeichnen einer Protokolldatei für das Modem

Einführung

Um mögliche Fehlerursachen bei einem Modem leichter lokalisieren zu können, empfiehlt es sich, eine Protokolldatei aufzeichnen zu lassen.

Vorgehensweise:

Gehen Sie vor wie folgt:

1. Aktivieren Sie über die Systemsteuerung, Option "Telefon- und Modemoptionen", den Eigenschaftsdialog des verwendeten Modems.
2. Überprüfen Sie die Einstellungen zur Option "Protokoll" im Register "Diagnose" und ändern Sie, falls notwendig, die Einstellungen zur Protokolldatei, sodass diese aufgezeichnet wird.

Ergebnis:

In die Protokolldatei werden die Aktivitäten zwischen PG/PC und Modem eingetragen. Bei Problemen mit dem Verbindungsaufbau können Sie die Aufzeichnungen in der Protokolldatei auswerten, um Hinweise auf die Fehlerursache zu erhalten.

7.6.3 Wählverbindung zum TS Adapter wird nicht aufgebaut

Wählverbindung zum TS Adapter wird nicht aufgebaut

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls keine Fernverbindung zum TS Adapter aufgebaut werden kann.

Möglich Ursache	Überprüfung/Behebung
Verkabelung fehlerhaft	• Sind alle Verbindungskabel korrekt angeschlossen? • Sind Verbindungsstecker gelockert?
Wahlparameter für Haupt- und Nebenanschluss falsch eingestellt	• Passen die eingestellten Eigenschaften und Wahlparameter Ihres Modems zum Telefonanschluss an Hauptanschluss oder Nebenstelle? • Wenn Sie Ihr Modem an einer Amtsleitung (Hauptanschluss) betreiben, dürfen Sie im Dialogfeld "Wahlparameter" keine Amtskennzahl angeben. Die Felder der Amtskennzahl für Ortsgespräche und für Ferngespräche müssen leer sein.
Wahlverfahren falsch eingestellt	• Ist im Dialogfeld zu den Wahlparametern Ihres Modems das korrekte Wahlverfahren (Ton/Impuls) eingestellt? • Überprüfen Sie mit einem angeschlossenen Telefon den Anschluss, an dem Sie das Modem betreiben möchten. Bei Impulswahl sollten im Telefon Knackgeräusche während des Wählens hörbar sein, bei Tonwahl dagegen unterschiedlich hohe Töne. Stellen Sie das entsprechende Wahlverfahren bei den Wahlparametern des Modems ein.

Möglich Ursache	Überprüfung/Behebung
Wahlsperre aktiv	• Die Wahlsperre ist eine länderspezifisch vorgeschriebene Modem-Eigenschaft, die je nach Modem nach einem oder mehreren fehlgeschlagenen Versuchen eines Verbindungsaufbaus wirksam wird. • Reagiert Ihr Modem nach einigen vergeblichen Wahlversuchen nicht mehr, kann es sein, dass die Wahlsperre aktiv ist. Nach dem Wählbefehl werden zwar noch Zeichen zum Modem geschickt, aber das Modem startet den Wahlvorgang nicht. Der Treiber bekommt eine allgemeine Fehlermeldung. • Informieren Sie sich in der Modem-Dokumentation, wie bei Ihrem Modem die Wahlsperre realisiert ist. • Lassen Sie die Protokolldatei (Seite 186) (modemlog.txt) anlegen, in der die Aktivitäten zwischen PG/PC und Modem aufgezeichnet werden. Prüfen Sie dann, ob die Datei einen durch eine Wahlsperre verursachten Eintrag (z. B. DELAYED) enthält.
Telefonanschluss defekt oder belegt	• Schließen Sie ein Telefon an und prüfen Sie, ob auf diesem Anschluss ein Freizeichen zu hören ist. • Ist ein analoges Telefon am gleichen Anschluss angeschlossen, so muss dieses aufgelegt sein. Wenn bereits eine Telefonverbindung besteht, kann auf diesem Anschluss keine zusätzliche Modemverbindung hergestellt werden.
Serielle Parameter falsch eingestellt	• Sind im Register "Einstellungen" der Modemeigenschaften die korrekten Werte eingetragen (8 Datenbits, keine Parität, 1 Stoppbit)? Ist im Register "Allgemein" der Modemeigenschaften die richtige COM-Schnittstelle eingestellt?
Initialisierungsstring des TS Adapter passt nicht zum Modem.	• Informieren Sie sich über die Anforderungen an den Modem-Initialisierungsstring und stellen Sie diesen gemäß den Anforderungen ein. Vorgehensweise zum Parametrieren des TS Adapter IE (Seite 158)
Einstellung der Fehlerkorrektur zwischen Modem am TS Adapter und Modem am PC/PG sind nicht verträglich.	• Passen Sie die Modem-Einstellungen an. Wissenswertes zum Parametrieren des TS Adapter MPI (Seite 152) Wiederherstellen der Standard-Parametrierung eines TS Adapter MPI (Seite 154) Vorgehensweise zum Parametrieren des TS Adapters IE (Seite 153)

7.6.4 Wählverbindung vom TS Adapter wird nicht aufgebaut

Rückruf vom TS Adapter erfolgt nicht

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls kein Rückruf vom TS Adapter erfolgt.

Mögliche Ursache	Überprüfung / Behebung
Standort- oder Rufeinstellungen im TS Adapter sind fehlerhaft	Überprüfen Sie die Parametrierung des TS Adapter: • Sind Wahlverfahren und Amtskennzahl entsprechend ihrem Telefonanschluss eingestellt? • Unterstützt das Modem am TS Adapter die für die Amtskennzahl parametrierten Zeichen? • Ist bei einer Nebenstelle "Vor dem Wählen auf Freizeichen warten" deaktiviert?
Initialisierung des Modems nicht ausreichend	Überprüfen Sie den String zur Initialisierung des Modems: • Vielleicht benötigt das Modem zum Aufbauen einer Fernverbindung eine erweiterte Initialisierung. • Eigenschaften des Modem-Initialisierungsstrings beim TS Adapter MPI
Rückrufnummer stimmt nicht	Überprüfen Sie die Parametrierung der von Ihnen vergebenen Rückrufnummer.

Anruf vom TS Adapter MPI erfolgt nicht

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls kein Anruf vom TS Adapter MPI erfolgt.

Mögliche Ursache	Überprüfung / Behebung
Rufnummer stimmt nicht	Wird der Kommunikationsanweisung "PG_DIAL" die gewünschte Rufnummer übergeben?
Parametrierung des TS Adapter MPI ist inkorrekt	Überprüfen Sie die Parametrierung des TS Adapter MPI: • Sind Wahlverfahren und Amtskennzahl entsprechend Ihrem Telefonanschluss eingestellt? • Unterstützt das Modem am TS Adapter MPI die für die Amtskennzahl parametrierten Zeichen? • Ist bei einer Nebenstelle "Vor dem Wählen auf Freizeichen warten" deaktiviert?

7.6.5 Modemverbindung wird unterbrochen

Modemverbindung wird unterbrochen

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls die Modemverbindung unterbrochen wird.

Mögliche Ursache:	Überprüfung / Behebung:
Gebührenimpulse in der Leitung	Wenn Sie bei der Telefongesellschaft einen Gebührentakt beantragt haben, werden Gebührenimpulse erzeugt. Diese können dazu führen, dass das Modem das Trägersignal nicht mehr erkennt und abschaltet. • Stellen Sie eine längere Warte- bzw. Abschaltzeit am Modem ein. • Lassen Sie von der Telefongesellschaft den Gebührentakt abschalten.
Abschirmung	• Sind die verwendeten Verbindungskabel ausreichend abgeschirmt? • Stellen Sie sicher, dass die Modem-Zuleitungen nicht neben Netzkabeln verlaufen und möglichst großen Abstand von Netzteilen und Bildschirmen haben.
Protokoll-Timeout	• Stellen Sie feste Überwachungszeiten ein.
Automatischer Verbindungsabbau	• Schalten Sie die Option aus, dass sich eine bestehende Verbindung nach einer bestimmten Zeit ohne Datenübertragung selbst abschaltet ("Trennen nach Leerlauf von ...").
Datenflusskontrolle ausgeschaltet	• Klicken Sie auf die Schaltfläche "Erweitert" im Register "Einstellungen" der Modemeigenschaften und aktivieren Sie im angezeigten Dialogfeld folgende Optionen (falls angeboten und noch nicht eingestellt): – Datenflusskontrolle – Hardware (RTS/CTS) – Datenkomprimierung – Fehlerkontrolle
Initialisierungsstring des TS Adapter passt nicht zum Modem	• Stellen Sie den Modem-Initialisierungsstring gemäß den Anforderungen ein. Siehe auch: Parametriermöglichkeiten für den TS Adapter IE (Seite 160)

Siehe auch

Parametriermöglichkeiten für den TS Adapter MPI (Seite 152)

7.6.6 Checkliste bei Schwierigkeiten mit dem Modem

Einführung

Die folgende Liste soll Ihnen bei Problemen mit dem Modem helfen, die mögliche Ursache zu finden. Wie und in welchen Dialogfeldern Sie die genannten Einstellungen definieren, erfahren Sie in den unten aufgeführten Hilfethemen.

Die Modemverbindung kommt nicht zustande:

- Prüfen Sie die Verkabelung und die Anschlüsse.
- Prüfen Sie, ob das Wahlverfahren (Ton/Impuls) korrekt eingestellt ist.
- Reagiert Ihr Modem nach einigen vergeblichen Wahlversuchen nicht mehr, kann es sein, dass eine Wahlsperre aktiv ist. Informieren Sie sich, wie bei Ihrem Modem die Wahlsperre implementiert ist.
- Betreiben Sie Ihr Modem an einem Hauptanschluss oder an einer Nebenstelle? Richten Sie die Eigenschaften und Wahlparameter des Modems entsprechend ein.
- Aktivieren Sie in den erweiterten Eigenschaften die Option, eine Protokolldatei aufzuzeichnen. Daraufhin wird der nächste Verbindungsaufbau in einer Datei im Windows-Verzeichnis protokolliert.
- Achten Sie darauf, dass die benutzten ISDN-TAs mit demselben B- und D-Kanal-Protokoll arbeiten.

Die Modemverbindung wird getrennt:

- Gebührenimpulse können sich negativ auf eine Verbindung auswirken. Lassen Sie die Impulse von Ihrem Telekommunikationsanbieter abschalten.
- Stellen Sie feste Überwachungszeiten ein.
- Schalten Sie die Option aus, dass sich eine bestehende Verbindung nach einer bestimmten Zeit ohne Datenübertragung (Leerlauf) selbst abschaltet.
- Vergewissern Sie sich, dass Sie RTS/CTS für die Datenflusskontrolle aktiviert haben.

7.6.7 Meldungen des Modems

Informationen in der Protokolldatei

Die Meldungen des Modems werden in einer Protokolldatei eingetragen, sofern Sie die Aufzeichnung aktiviert haben.

Folgende Informationen sind in der Protokolldatei enthalten:

Meldung:	mögliche Ursache:	Behebung:
NO DIALTONE	Eventuell wird auf dieser Leitung momentan ein Telefongespräch geführt.	• Wiederholen Sie den Vorgang, wenn das Telefongespräch auf dieser Leitung beendet ist.
NO CARRIER	Der angewählte Teilnehmer ist nicht bereit, ist kein Modem oder kann mit der eingestellten Betriebsart keine Verbindung herstellen.	• Prüfen Sie die Rufnummern und die Einstellungen.
BUSY	Der angewählte Teilnehmer ist besetzt.	• Versuchen Sie es später nochmals.
DELAYED: ...	Wahlsperre	• Informieren Sie sich in der Dokumentation zu Ihrem Modem, wie bei Ihrem Modem die Wahlsperre realisiert ist und entfernen Sie diese, falls gewünscht.

7.6.8 Mögliche Fehlermeldungen bei VPN-Verbindungen

VPN-Verbindung zum TS Adapter IE Advanced wird nicht aufgebaut

Die folgende Tabelle zeigt mögliche Fehler und deren Behebung auf, falls keine VPN-Verbindung zum TS Adapter IE Advanced aufgebaut werden kann.

Mögliche Fehler	Überprüfung/Behebung
Im Browser erscheint bei Aufruf des Web-Interface die Fehlermeldung "Die Webseite kann nicht angezeigt werden".	• Hinweis: Dieses Problem tritt dann auf, wenn im Windows-Zertifikatspeicher ein CA-Zertifikat vorhanden ist, das den gleichen Namen wie das neu erzeugte CA-Zertifikat hat. • Falls bereits ein CA-Zertifikat für diesen TS Adapter im Windows-Zertifikatsspeicher installiert ist und Sie auf dem TS Adapter ein neues CA-Zertifikat generiert haben, so entfernen Sie das alte CA-Zertifikat aus dem Windows-Zertifikatsspeicher und installieren Sie das neu generierte CA-Zertifikat. • siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 168) • siehe auch: CA-Zertifikat für VPN-Verbindung löschen (Seite 171)
Fehlermeldung: "Es ist ein unerwarteter Fehler in der Applikation aufgetreten".	• Überprüfen Sie, ob in der Benutzerdatenbank des TS Adapters IE Advanced weitere Benutzer angelegt sind. • Der Benutzer "Administrator" kann keine Fernverbindung aufbauen.
Fehlermeldung: "Es wurde kein passendes CA-Zertifikat im Windows-Zertifikatsspeicher gefunden".	• Nutzen Sie den automatischen Zertifikatsdownload. Tragen Sie dazu den Fingerabdruck in das entsprechende Feld im Verbindungsdialog ein. • siehe auch: VPN-Verbindung aufbauen (Seite 171) Oder • Exportieren Sie das CA-Zertifikat aus dem Web-Interface des TS Adapter IE Advanced und installieren Sie es manuell in den Windows-Zertifikatsspeicher. • siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 168)
Fehlermeldung: "Die angegebene Remote-Adresse passt nicht zur Remote-Adresse des TS Adapters".	• Sie müssen für den Verbindungsaufbau die Remote-Adresse verwenden, die Sie im TS Adapter IE Advanced angegeben haben. • Es ist nicht möglich, bei eingetragenem DNS-Namen die IP-Adresse (oder umgekehrt) zu verwenden.
Fehlermeldung: "Die Signatur des Zertifikats konnte nicht überprüft werden".	• Falls Sie auf dem TS Adapter IE Advanced ein neues CA-Zertifikat generiert haben, müssen Sie das alte CA-Zertifikat aus dem Windows-Zertifikatsspeicher löschen und das neue installieren. • siehe auch: CA-Zertifikat für VPN-Verbindung löschen (Seite 171) • siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 168)
Fehlermeldung: "Protokollfehler"	• Überprüfen Sie, ob in der Konfiguration des TS Adapter IE Advanced die IP-Adresse des Service-PC im Anlagennetz gesetzt ist (Parameter > Anlagennetz).

Mögliche Fehler	Überprüfung/Behebung
Fehlermeldung: "Beim Verbindungsaufbau trat ein Fehler auf."	• Überprüfen Sie, ob die LAN-IP-Adresse des TS Adapter IE Advanced und die IP-Adresse einer Ihrer Netzwerkkarten im selben Subnetz liegen. Deaktivieren Sie in diesem Fall diese Netzwerkkarte, bevor Sie die Fernverbindung aufbauen.
Das Versenden einer E-Mail funktioniert nicht.	• Überprüfen Sie, ob ein Netzwerk/Internetverbindung am WAN-Port besteht. • Überprüfen Sie, ob der Mailserver erreichbar ist. • Überprüfen Sie, ob im TS Adapter IE Advanced ausgehende Verbindungen erlaubt sind. • Überprüfen Sie, ob der SMTP-Port in der Firewall für ausgehende Verbindungen geöffnet ist.

Geräte simulieren mit S7-PLCSIM 8

8.1 Simulation von Geräten

Einführung

Mit dem TIA Portal können Sie die Hardware und Software des Projekts in einer simulierten Umgebung ausführen und testen. Die Simulation wird direkt auf dem PG/PC ausgeführt. Daher ist keine zusätzliche Hardware nötig.

Die Simulationssoftware bietet eine grafische Bedienoberfläche zum Beobachten und Ändern der Projektierung. Sie unterscheidet sich je nach aktuell selektiertem Gerät.

Einbindung in das TIA Portal

Die Simulationssoftware ist vollständig in das TIA Portal integriert, wird aber nur von bestimmten Geräten unterstützt. Daher ist die Schaltfläche zum Aufruf der Simulationssoftware nur bedienbar, wenn das selektierte Gerät die Simulation unterstützt.

Die Simulationssoftware zu manchen Geräten benötigt eine eigene virtuelle Schnittstelle, um mit den simulierten Geräten zu kommunizieren. Die virtuelle Schnittstelle finden Sie in der Projektnavigation unter dem Eintrag "Online-Zugänge" neben den physikalischen Schnittstellen des PG/PC.

Weiterführende Hilfe zur Simulationssoftware bekommen Sie nach dem Öffnen der Software über einen gesonderten Link.

Siehe auch

Simulation starten (Seite 196)

8.2 Simulation starten

Einige Geräte lassen sich mit einer zusätzlichen Software simulieren. Sie müssen daher nicht tatsächlich über die Geräte verfügen, können aber trotzdem Ihr Projekt umfangreich testen.

Vorgehen

Um die Simulations-Software zu starten, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Gerät, das Sie simulieren möchten, z. B. in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Simulation > Starten".
   Die Simulations-Software wird aufgerufen.

Siehe auch

Simulation von Geräten (Seite 195)

Index

A

Allgemeine Hinweise für die Fehlersuche, 186
Anschließen eines TS Adapter mit externem Modem, 143
Anschließen eines TS Adapter mit internem Modem, 143
Aufbauen einer Fernverbindung, 163
    Vorgehen, 163
Aufzeichnen einer Protokolldatei für das Modem, 186
Automatisierungssystem
    entfernt, 180
    lokal, 180

B

Backup von online Gerät laden, 29, 31
Backup wiederherstellen, 28, 33
Baugruppen simulieren, 195
Bezeichnung
    TS Adapter, 157
    TS Adapter, 141, 149

C

CA-Zertifikat, 166
CA-Zertifikat installieren, 168
CA-Zertifikat löschen, 171
Checkliste bei Schwierigkeiten mit dem Modem, 190
CPU-Belastung durch Trace, 113

D

Datenaustausch über AS-AS-Fernkopplung
    Kommunikationsanweisung "AS_DIAL", 181

E

Einsetzen eines TS Adapters für TeleService, 140, 165
Ethernet-Schnittstelle
    Parameter anzeigen, 42
Ethernet-Schnittstelle parametrieren, 43
    Änderbare Parameter, 41
    Hinzufügen einer temporären IP-Adresse, 44
    Mit Subnetz verbinden, 43
    Möglichkeiten zur Parametrierung, 41
    Temporäre IP-Adressen löschen, 45

F

Fernverbindung
    als CPU-gesteuerte Fernverbindung, 178
    als VPN-Verbindung, 165
    als Wählverbindung, 140
Fernverbindung vom TS Adapter wird nicht aufgebaut, 189
Fernverbindung zum TS Adapter wird nicht aufgebaut, 187
Fernverbindungsaufbau nicht möglich, 164

G

Gerät wiederherstellen, 28, 33
Geräte simulieren, 195
Gerätekonfiguration online ändern, 11

H

Hardware simulieren, 195
Hardware-Voraussetzungen
    Kommunikationsanweisung "AS_MAIL", 184
    Kommunikationsanweisung "PG_DIAL", 179
    Kommunikationsanweisung "SMS_SEND", 183
Hardware-Voraussetzungen für AS-AS-Fernkopplung
    Kommunikationsanweisung "AS_DIAL", 181
Hinweise zum Parametrieren eines entfernten Modems, 144

I

Installieren des lokalen Modems, 142

K

Kommunikationsanweisung
    "AS_DIAL", 180, 181
    "AS_MAIL", 184
    "PG_DIAL", 179, 180
    "SMS_SEND", 182
Konfiguration online ändern, 11

L

Logikanalysatorfunktion, 58

M

Meldungen des Modems
  Protokolldatei, 191
Messungen
  Projektnavigator, 66
Modem
  entferntes, 143
  lokales, 142
  Probleme, 186
Modemverbindung kommt nicht zustande, 191
Modemverbindung wird getrennt, 191
Modemverbindung wird unterbrochen, 190
  Ursachen und Behebung, 190
MPI/PROFIBUS Einstellungen zurücksetzen, 54

N

Netzzugang
  Eigenschaften öffnen, 39
Nicht Plug&Play-fähige Modems, 143

O

O, 13
Öffnen von Telefonbüchern, 134
Online
  Darstellung des Online-Modus, 19
  Erreichbare Teilnehmer anzeigen, 9
  Verbindung herstellen, 18
  Verbindung trennen, 21
  Verbindungsweg ändern, 18
Online verbinden, 18
  mehrere Geräte verbinden, 22, 23
Online-Modus, 13
  Anzeige des Online-Modus, 15
  Online verbinden, 18, 21
  Standby oder Ruhezustand des PG/PC, 13
Online-Verbindung, 13
  Standardverbindungsweg festlegen, 17
Online-Zugang, 35
Online-Zustand anzeigen, 19

P

Parametrieren des entfernten Modems, 143
PG/PC-Schnittstelle
  Einstellbare Parameter an der MPI-Schnittstelle, 46
  Einstellbare Parameter an der PROFIBUS-Schnittstelle, 46
  online verbinden, 21
PG/PC-Schnittstelle parametrieren, 37
  Automatische Erkennung der Busparameter, 47
  MPI-Schnittstelle parametrieren, 48
  PROFIBUS-Schnittstelle parametrieren, 50
Protokolldatei
  Meldungen des Modems, 191

R

Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer, 147
Rückrufmöglichkeiten in TeleService, 147

S

Schnittstelle
  hinzufügen, 40
Schnittstellen anzeigen/verbergen, 38
Sicherung von online Gerät laden, 29, 31
Sicherung wiederherstellen, 28, 33
Simulation starten, 196
Software simulieren, 195
Software-Voraussetzungen auf Anlagen-Seite
  Kommunikationsanweisung "AS_MAIL", 184
  Kommunikationsanweisung "PG_DIAL", 180
  Kommunikationsanweisung "SMS_SEND", 183
Software-Voraussetzungen auf PG/PC-Seite
  Kommunikationsanweisung "PG_DIAL", 180
Software-Voraussetzungen für AS-AS-Fernkopplung
  Kommunikationsanweisung "AS_DIAL", 182
Speichern von Telefonbüchern, 135
Symbole
  in TeleService, 134

T

TCP/IP Einstellungen zurücksetzen, 45
Telefonbuch bei TeleService, 131
TeleService
  Arbeiten mit dem Telefonbuch, 132

AS_DIAL, 180
Aufbau des Telefonbuches, 133
Bedeutung der Symbole, 134
Definieren von Wählregeln, 138
Eigenschaften des Telefonbuchs, 131
Funktionalität, 130
Kommunikationsanweisung:"AS_DIAL", 130
Kommunikationsanweisung:"AS_MAIL", 130
Kommunikationsanweisung:"PG_DIAL", 130
Kommunikationsanweisung:"SMS_SEND", 130
Kommunikationsanweisung:"TM_MAIL", 130
Kommunikationsanweisung:"TMAIL_C", 130
Leistungsfähigkeit in Telefonnetzen, 142
Modemtypen/Medien, 142
Modem-Unterstützung, 141
Möglichkeiten zum Verbindungsaufbau:S7-1200er CPUs, 178
Möglichkeiten zum Verbindungsaufbau:S7-1500er CPUs, 178
Möglichkeiten zum Verbindungsaufbau:S7-300/400er CPUs, 178
Netzübergänge, 142
Rückrufvarianten, 145
Spalten im Telefonbuch ein- oder ausblenden, 135
Telefonbuch, 131
Telefonbuch drucken, 138
Telefonbuch exportieren, 137
Telefonbuch importieren, 135
Telefonbuch öffnen, 134
Telefonbuch speichern, 135
Verbindung zur AS aufbauen, 180
Wählregeln definieren, 138
Zeilen ins Telefonbuch einfügen, 135
Zugriff auf Telefonbücher, 132
Trace, 58, (Triggerzeitpunkt)
Abtastung, 91
Aufzeichnung, 60, 98
Aufzeichnungsbedingungen, 115, 120
Aufzeichnungsdauer, 125
Aufzeichnungsebenen, 112
Aufzeichnungstakt, 91, 125
Bitspur, 71
CPU-Belastung, 113
Datenablage, 61
Diagramm anzeigen, 95
drucken, 110
erfassbare Variablen, 111
Inspektorfenster, 88
Kurvendiagramm, 71, 102
Laufende Aufzeichnung analysieren, 99
Lebensdauer der Werte, 112
Mathematische Funktionen, 79
Mausrad, 72
Mengengerüst, 113
Mess-Cursor, 102
Messung, 60, 61, 100
Messungen im Gerät (Speicherkarte), 61, 67
Messungen im Gerät speichern, 119, 126
Oberfläche, 63, 88, 113
Palette Mess-Cursor, 85
Palette Momentaufnahmen, 87
Pretrigger, 118
Projektnavigation in Gruppen strukturieren, 96
Projektnavigator, 64
Schnelle Signale beobachten, 107
Schnelleinstieg, 90
Signale, 114, 124
Signalgruppe, 106
Signaltabelle, 76, 105
Status, 65
Tastaturbedienung, 72
Trace im Gerät, 60
Tracekonfiguration, 60, 61, 97, 101, 102, 124
Tracekonfiguration anlegen, 94
Tracekonfiguration anzeigen, 95
Tracekonfiguration speichern, 102
Trigger, 125
Triggermodus, 117
Triggervariable, 117, 120
Überlagerte Messung, 61, 68, 82, 108
Überlagerte Messung anlegen, 96
Überlagerte Messungen ausrichten, 110
Untersetzung, 116
unterstützte Geräte, 58
Zeitstempel, 88
Trace S7-1200/1500, 111
Tracefunktion, 58
Trennen der Fernverbindung, 164
TS Adapter IE
Funktionsweise, 158
Kurzbeschreibung, 157
parametrieren, 161, 176
Standard-Parametrierung, 158
TS Adapter IE Advanced
Anschlussarten, 174
Kurzbeschreibung, 173
parametrieren, 176
Parametriermöglichkeiten, 175
TS Adapter MPI
Aufbau des Direktanschlusses, 151
Aufbau des Modemanschlusses, 151
Direktanschluss, 150
Funktionsweise, 149

Kurzbeschreibung, 149
Modemanschluss, 151
parametrieren, 153
Parametriermöglichkeiten, 152
Parametrierung durchführen, 153
Standard-Parametrierung wiederherstellen, 154
TS Adapter MPI
Standard-Parametrierung, 150
TS Adapter IE Basic
Anschluss an das GSM-Netz, 160
Anschluss an das Telefonnetz, 159
Anschluss an das Telefonnetz über ein externes Modem, 160
Anschlussarten, 158
TS Adapter MPI
Adapterparameter exportieren, 155
Adapterparameter importieren, 156
Zugriffschutz einrichten, 146

U

Übersicht der einsetzbaren TS Adapter, 141

V

Verbindungsaufbau von einer entfernten Anlage ("PG_DIAL"), 179
Voraussetzungen für das Senden einer E-Mail, 184
Voraussetzungen für das Senden einer SMS, 182
Voraussetzungen zum Aufbau einer Fernverbindung, 163
Voraussetzungen zum Aufbau einer VPN-Verbindung, 172
Vorteile durch den Einsatz von TeleService, 129
VPN-Verbindung
aufbauen, 171
CA-Zertifikat installieren, 168
CA-Zertifikat löschen, 171
trennen, 173
VPN-Verbindung wird nicht aufgebaut
Überprüfung und Behebung, 192

W

Wählregeln in Telefonbüchern, 138
Wählverbindung
aufbauen, 163
trennen, 164
Wissenswertes zum Parametrieren des TS Adapter MPI, 152

Z

Zugriff von entfernten Anlagen auf ein PG/PC, 179
Zugriffsschutz
Gültigkeit, 144
Vorteile, 145
werkseitig, 144
Zugriffschutzstufe 1, 145
Zugriffschutzstufe 2, 145

SIEMENS

Energiedaten verarbeiten

Programmierhandbuch

Grundlagen zu SIMATIC Energy Suite 1

Energiedaten liefern 2

Energiedaten verarbeiten 3

Energiedaten puffern und archivieren 4

Energieprogramm generieren 5

Energiedaten visualisieren und auswerten 6

Energy Suite-Bausteine A

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Grundlagen zu SIMATIC Energy Suite....9
1.1 Energiemanagement mit SIMATIC Energy Suite....9
1.2 Funktionsweise und Komponenten von SIMATIC Energy Suite....11
1.3 Handlungsübersicht....15
1.4 Leistungsmerkmale und Konventionen....18
2 Energiedaten liefern....21
2.1 Wissenswertes zu Energiedaten....22
2.2 Energiedaten von Messhardware....24
2.3 Energiedaten aus dem Prozessabbild oder Merker-Bereich der S7-CPU....27
2.4 Energiedaten aus Datenbausteinen....29
3 Energiedaten verarbeiten....31
3.1 Energieobjekte und Energieprogramm....31
3.2 Energieobjekttabelle anlegen....33
3.3 Energieobjekte zu Energieobjekttabellen hinzufügen....34
3.4 Eigenschaften der Energieobjekte einstellen....37
3.4.1 Energieobjekt parametrieren....37
3.4.2 Grundeinstellungen....38
3.4.3 Projektinformation....39
3.4.4 Identification & Maintenance....39
3.4.5 Freigabe der Energieobjekte für Energieprogramm ändern....40
3.4.6 Energiedatenquelle....41
3.4.6.1 Energiedatenquelle zuweisen....41
3.4.6.2 Energiedatenquelle....42
3.4.6.3 Energiedatentyp....43
3.4.7 Periodische Energiewerte....46
3.4.7.1 Energiewerte für periodische Archivierung festlegen....46
3.4.7.2 Periodische Energiedaten....47
3.4.8 Verarbeitung....49
3.4.8.1 Allgemeine Verarbeitungseinstellungen....49
4 Energiedaten puffern und archivieren....51
4.1 Archivierungsperioden definieren....51
4.2 Archivierungsperioden....53
4.3 Archivierung auf S7-CPU (SIMATIC Memory Card) festlegen....55
4.4 Archivierung und Pufferung (S7-CPU)....56
4.5 Beispiel für Archivierung (S7-CPU)....58
4.6 Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen....59

4.7 Archivierung und Pufferung (WinCC Professional)....61
4.8 Beispiele für Archivierung und Pufferung (WinCC Professional)....65
5 Energieprogramm generieren....67
5.1 Allgemeine Einstellungen für das Energieprogramm festlegen....67
5.2 Allgemeine Einstellungen....68
5.3 Runtime-Lizenzen für Energy Suite....69
5.4 Energieprogramm generieren....70
5.5 Struktur des Energieprogramms....72
5.5.1 Aufbau des Energieprogramms....72
5.5.2 EnS - block types....73
5.5.3 EnS - energy objects....75
5.5.4 EnS - general program blocks....76
5.5.5 Energy Suite - data types....79
5.5.6 Treiberbausteine für Messhardware....81
5.5.6.1 Aufbau des Energieprogramms für Messhardware....81
5.5.6.2 Energiedatenverarbeitung nach dem "EVA"-Prinzip....81
5.5.6.3 Bereich: Eingabe....83
5.5.6.4 Bereich: Verarbeitung....83
5.5.6.5 Bereich: Ausgabe....85
5.6 PLC-Programm übersetzen und laden....87
5.7 Energieprogramm testen....88
5.8 PLC-Programm ohne Installation von Energy Suite bearbeiten....89
5.9 PLC-Programm ohne Messhardware-Bibliothek mit Treiberbausteinen....91
6 Energiedaten visualisieren und auswerten....93
6.1 Handlungsübersicht für Archivierung und Visualisierung mit WinCC Professional....93
6.2 Energiewerte auf PC-Station mit WinCC Professional....95
6.3 Energiewerte für die Archivierung in WinCC Professional anzeigen....96
6.4 Übersicht der Energiedaten....98
6.5 Archivierungstruktur für Energiewerte erzeugen....99
6.6 Variablenarchiv und Segmentierung für Energy Suite....102
6.7 Energiedaten archivieren....103
6.8 Energiedaten visualisieren mit WinCC Online Trend View....104
6.9 Energiedaten auswerten mit Energy Suite Export Tool....105
A Energy Suite-Bausteine....111
A.1 Übersicht der Energy Suite-Bausteine....111
A.2 Funktionsbausteine (FBs)....116
A.2.1 EnS_ArchiveDataLog: Energiewerte in Data Logs schreiben....116
A.2.1.1 Beschreibung EnS_ArchiveDataLog....116
A.2.1.2 Parameter EnS_ArchiveDataLog....117
A.2.1.3 Parameter Status....119

A.2.1.4 Meldungen EnS_ArchiveDataLog....121
A.2.2 EnS_ArchiveHMI: Energiewerte in Variablenarchive schreiben....124
A.2.2.1 Beschreibung EnS_ArchiveHMI....124
A.2.2.2 WinCC Professional-Komponente....126
A.2.2.3 Parameter EnS_ArchiveHMI....127
A.2.2.4 Parameter Status....128
A.2.2.5 Meldungen EnS_ArchiveHMI....129
A.2.3 EnS_BufferRead: Energiewerte aus Datenbausteinen lesen....131
A.2.3.1 Beschreibung EnS_BufferRead....131
A.2.3.2 Parameter - EnS_BufferRead....132
A.2.3.3 Parameter Status....133
A.2.4 EnS_BufferWrite: Energiewerte in Datenbausteine schreiben....133
A.2.4.1 Beschreibung EnS_BufferWrite....133
A.2.4.2 Parameter EnS_BufferWrite....134
A.2.4.3 Parameter Status....135
A.2.5 EnS_CalcEnergyData: Energiewerte berechnen....136
A.2.5.1 Beschreibung EnS_CalcEnergyData....136
A.2.5.2 Parameter EnS_CalcEnergyData....140
A.2.5.3 Parameter Status....140
A.2.6 EnS_DrvAnalogValue: Leistungswerte vom Typ REAL erfassen....142
A.2.6.1 Beschreibung EnS_DrvAnalogValue....142
A.2.6.2 Parameter EnS_DrvAnalogValue....142
A.2.6.3 Parameter Status....143
A.2.6.4 Meldungen EnS_DrvAnalogValue....143
A.2.7 EnS_DrvCounterLInt: Energiezählwerte vom Typ LINT erfassen....144
A.2.7.1 Beschreibung EnS_DrvCounterLInt....144
A.2.7.2 Parameter EnS_DrvCounterLInt....144
A.2.7.3 Parameter Status....145
A.2.7.4 Meldungen EnS_DrvCounterLInt....145
A.2.8 EnS_DrvCounterLReal: Enegiezählwerte vom Typ LREAL erfassen....146
A.2.8.1 Beschreibung EnS_DrvCounterLReal....146
A.2.8.2 Parameter EnS_DrvCounterLReal....146
A.2.8.3 Parameter Status....147
A.2.8.4 Meldungen EnS_DrvCounterLReal....147
A.2.9 EnS_DrvPulse: Impulssignale erfassen....148
A.2.9.1 Beschreibung EnS_DrvPulse....148
A.2.9.2 Parameter EnS_DrvPulse....148
A.2.9.3 Parameter Status....149
A.2.9.4 Meldungen EnS_DrvPulse....149
A.2.10 EnS_EnergyDataAdv: Erweiterte Energiewerte bereitstellen....150
A.2.10.1 Beschreibung EnS_EnergyDataAdv....150
A.2.10.2 Parameter EnS_EnergyDataAdv....150
A.2.10.3 Parameter Status....151
A.2.11 EnS_EnergyDataAdvMinMax: Erweiterte Minimal- und Maximalwerte bereitstellen....151
A.2.11.1 Beschreibung EnS_EnergyDataAdvMinMax....151
A.2.11.2 Parameter EnS_EnergyDataAdvMinMax....152
A.2.11.3 Parameter Status....152
A.2.12 EnS_EnergyDataBasic: Metadaten und Energiewerte bereitstellen....152
A.2.12.1 Beschreibung EnS_EnergyDataBasic....152
A.2.12.2 Parameter EnS_EnergyDataBasic....153
A.2.12.3 Parameter Status....154
A.2.12.4 Meldungen EnS_EnergyDataBasic....154

A.2.13 EnS_Organization: Organisationsdaten bereitstellen....156
A.2.13.1 Beschreibung EnS_Organization....156
A.2.13.2 Parameter EnS_Organization....157
A.2.14 EnS_TimeSync: Zeiten synchronisieren....157
A.2.14.1 Beschreibung EnS_TimeSync....157
A.2.14.2 Parameter EnS_TimeSync....160
A.2.14.3 Parameter Status....161

A.3 Datenbausteine (DBs)....164
A.3.1 EnS_BufferDB....164
A.3.1.1 Beschreibung EnS_BufferDB....164
A.3.1.2 Aufbau EnS_BufferDB....164
A.3.2 EnS_BufferWorkDB....165
A.3.2.1 Beschreibung EnS_BufferWorkDB....165
A.3.2.2 Aufbau EnS_BufferWorkDB....165

A.4 Anwenderdefinierte Datentypen (UDTs)....166
A.4.1 EnS_typeAnalogValue....166
A.4.1.1 Beschreibung EnS_typeAnalogValue....166
A.4.1.2 Aufbau EnS_typeAnalogValue....166
A.4.2 EnS_typeBufferReadCtrl....166
A.4.2.1 Beschreibung EnS_typeBufferReadCtrl....166
A.4.2.2 Aufbau EnS_typeBufferReadCtrl....166
A.4.3 EnS_typeBufferReadStat....167
A.4.3.1 Beschreibung EnS_typeBufferReadStat....167
A.4.3.2 Aufbau EnS_typeBufferReadStat....167
A.4.4 EnS_typeBufferWork....167
A.4.4.1 Beschreibung EnS_typeBufferWork....167
A.4.4.2 Aufbau EnS_typeBufferWork....167
A.4.5 EnS_typeBufferWriteCtrl....168
A.4.5.1 Beschreibung EnS_typeBufferWriteCtrl....168
A.4.5.2 Aufbau - EnS_typeBufferWriteCtrl....169
A.4.6 EnS_typeBufferWriteStat....169
A.4.6.1 Beschreibung EnS_typeBufferWriteStat....169
A.4.6.2 Aufbau EnS_typeBufferWriteStat....169
A.4.7 EnS_typeEnergyAdv....169
A.4.7.1 Beschreibung EnS_typeEnergyAdv....169
A.4.7.2 Aufbau EnS_typeEnergyAdv....170
A.4.8 EnS_typeEnergyBasic....171
A.4.8.1 Beschreibung EnS_typeEnergyBasic....171
A.4.8.2 Aufbau EnS_typeEnergyBasic....171
A.4.9 EnS_typeEnergyCounter....171
A.4.9.1 Beschreibung EnS_typeEnergyCounter....171
A.4.9.2 Aufbau EnS_typeEnergyCounter....172
A.4.10 EnS_typeEnergyMeta....172
A.4.10.1 Beschreibung EnS_typeEnergyMeta....172
A.4.10.2 Aufbau EnS_typeEnergyMeta....172
A.4.11 EnS_typeHMIWriteCtrl....175
A.4.11.1 Beschreibung EnS_typeHMIWriteCtrl....175
A.4.11.2 Aufbau EnS_typeHMIWriteCtrl....175
A.4.12 EnS_typeHMIWriteStat....176
A.4.12.1 Beschreibung EnS_typeHMIWriteStat....176
A.4.12.2 Aufbau EnS_typeHMIWriteStat....176

A.4.13 EnS_typeOrgData....176
A.4.13.1 Beschreibung EnS_typeOrgData....176
A.4.13.2 Aufbau EnS_typeOrgData....177
A.4.14 EnS_typeSync....177
A.4.14.1 Beschreibung EnS_typeSync....177
A.4.14.2 Aufbau EnS_typeSync....177
A.4.15 EnS_typeValue....178
A.4.15.1 Beschreibung EnS_typeValue....178
A.4.15.2 Aufbau EnS_typeValue....178

A.5 Definition der Enumerationen....179
A.5.1 Enumeration für den Verbindungstyp des Messgeräts....179
A.5.2 Enumeration für Einheiten....179

Grundlagen zu SIMATIC Energy Suite 1

1.1 Energiemanagement mit SIMATIC Energy Suite

Energiemanagement

Global steigende Energiekosten sowie gesetzliche Vorgaben zur Erreichung der weltweiten Klimaziele motivieren auch Industriebetriebe immer mehr, sich mit der Energiedatenerfassung sowie dem Energiemanagement zu beschäftigen. Der erste Schritt zum nachhaltigen Energiemanagement ist, die wesentlichen Energieflüsse und Energieverbraucher eines Unternehmens oder einer Produktionsanlage zu identifizieren und deren Verbrauchsverhalten zu analysieren. Dies wird als "Energietransparenz" bezeichnet.

Auf Grundlage der erfassten Energiedaten werden Maßnahmen zur Verbesserung der Energieeffizienz festgelegt. Anschließend werden die Maßnahmen in der technisch und ökonomisch effektivsten Reihenfolge umgesetzt. Die Reihenfolge der Umsetzung wird maßgeblich durch die Faktoren "Einsparpotenzial" und 'Umsetzungsaufwand' bestimmt. So können beispielsweise Maßnahmen mit mittlerem Einsparpotenzial und geringem Aufwand gegenüber Maßnahmen mit hohem Potenzial, aber hohem Aufwand vorgezogen werden.

Ein weiterer Aspekt sind gesetzliche Vorgaben. Diese erfordern eine regelmäßige Überwachung und Berichterstattung von Emissionen. Mithilfe der erfassten Energiedaten kann unternehmensweit Energietransparenz geschaffen und eine lückenlose Energie- und Stoffbilanzierung erstellt werden.

Neben der Erfassung von Energiedaten stellen deren Visualisierung, Archivierung, Kennzahlenberechnung und das Energieberichtswesen weitere wichtige Elemente des Energiemanagements dar.

SIMATIC Energy Suite in STEP 7 und WinCC (TIA Portal)

Mit den Komponenten von SIMATIC Energy Suite, nachfolgend Energy Suite genannt, integrieren Sie zentrale Aufgaben des Energiemanagements bereits auf der Ebene Ihres Automatisierungssystems. Die Konfiguration erfolgt einfach und komfortabel direkt im TIA Portal (STEP 7 und WinCC).

Energy Suite erfasst, archiviert und stellt die Energiedaten in der CPU standardisiert bereit. Sie können die Energiedaten anschließend visualisieren und in Ihrem Automatisierungsprogramm verwenden (z. B. zur Überwachung).

Für die Visualisierung der erfassten Energiedaten in WinCC Professional oder in WinCC Basic, Comfort bzw. Advanced stehen im Siemens Industry Online Support Vorlagenbilder und Bildelemente bereit.

Zusätzlich können Sie die Energiedaten in SIMATIC Energy Manager PRO weiterverarbeiten:

- zur Kennzahlenberechnung, z. B. Energiekosten pro produziertem Stück, CO_2-Ausstoß pro Jahr
- zur komfortablen Erstellung von Energieberichten, z. B. um den Energieeinkauf zu optimieren

Weiterverarbeitung von Energiedaten mit SIMATIC Energy Manager PRO

Konfigurationsinformationen von Energy Suite können Sie direkt in SIMATIC Energy Manager PRO übernehmen und ersparen sich dadurch zusätzliche Projektierungsaufwände. SIMATIC Energy Manager PRO, nachfolgend Energy Manager PRO genannt, bietet vorwiegend Funktionen zur kaufmännischen und unternehmensweiten Betrachtung des Energieverbrauchs. Um den Energieverbrauch optimieren zu können, müssen die maßgeblichen Energieflüsse über alle Energiemedien des Unternehmens transparent sein.

Mit Energy Manager PRO erstellen Sie z. B. Energie- und Stoffbilanzen oder Berechnungen von Kennzahlen. Mithilfe der Kennzahlen (englisch: Key Performance Indicator (KPI)) bewerten und vergleichen Sie unterschiedliche Prozesse und Anlagen energetisch. Das Bewerten und Vergleichen wird auch als "Benchmarking" bezeichnet. Die Zuordnung der Energiekosten zu den tatsächlichen Verursachern führt zu Kostentransparenz und zur Sensibilisierung der Mitarbeiter in Bezug auf den Energieverbrauch von Maschinen, Prozessen und Anlagen.

Weitere informationen zu Energy Manager PRO finden Sie im Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/de/ps/14711).

1.2 Funktionsweise und Komponenten von SIMATIC Energy Suite

EVA-Prinzip

Die Energiedatenverarbeitung lässt sich nach dem "EVA"-Prinzip in "Eingabe", "Verarbeitung" und "Ausgabe" unterteilen:

- Eingabe: Bereitstellung der Eingangsmessdaten für die Verarbeitung, z. B. durch Messhardware oder Berechnungen innerhalb der S7-CPU
- Verarbeitung: Erfassung, Berechnung und Bereitstellung der Energiewerte für weitere Systeme durch Energieobjekte in der S7-CPU
- Ausgabe: Visualisierung und Archivierung der Energiewerte auf angeschlossenen HMI- oder SCADA-Systemen

Übersicht

Das folgende Bild zeigt den Zusammenhang der einzelnen Energy Suite-Komponenten und die Verarbeitungsschritte:

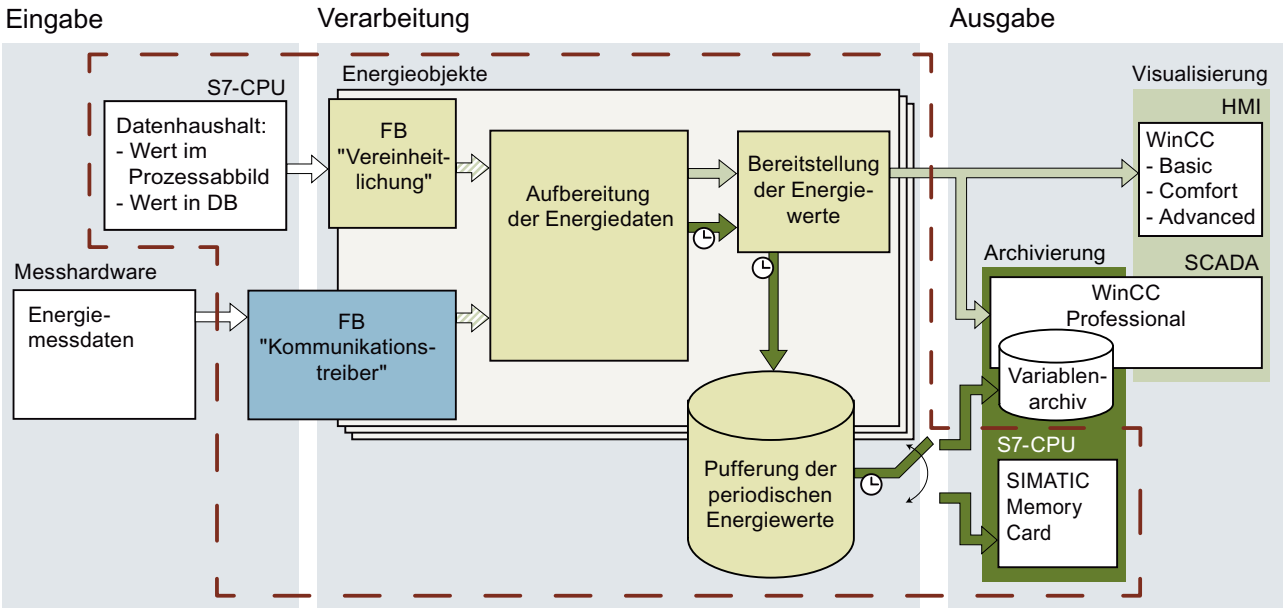

Eingangsmessdaten

Energierohdaten

Aufbereitete Energiemomentanwerte

Aufbereitete periodische Energiewerte mit Zeitstempel

Energy Suite-Bausteine (Präfix EnS)

Treiberbausteine für Messhardware (Präfix EnSL)

FB Funktionsbaustein

Prozess auf einer S7-CPU

Komponenten

Die folgende Tabelle zeigt die Komponenten für das Energiedaten-Management mit Energy Suite:

Bereich	Komponente/ Verarbeitungsschritt	Beschreibung
Eingabe - Energiedaten liefern (Seite 21)	S7-CPU	Im STEP 7-Projekt (TIA Portal) konfigurierte S7-1500 CPU Liefert die Eingangsmessdaten an den Funktionsbaustein (FB) "Vereinheitlichung"
	Messhardware	Von Energy Suite unterstützte messfähige Hardware Liefert die Eingangsmessdaten an den Funktionsbaustein (FB) "Kommunikationstreiber"

Bereich	Komponente/ Verarbeitungsschritt	Beschreibung
Verarbeitung - Energiedaten verarbeiten (Seite 31)	FB "Vereinheitlichung"	Beinhaltet die Energy Suite-Bausteine mit Treiberfunktionalität zur Erfassung, Vorverarbeitung und Harmonisierung von Eingangsmessdaten zu Energierohdaten Liefert die Energierohdaten zum Verarbeitungsschritt "Aufbereitung der Energiedaten" Folgende Energy Suite-Bausteine werden verwendet: • EnS_DrvAnalogValue (Seite 142) • EnS_DrvCounterLInt (Seite 144) • EnS_DrvCounterLReal (Seite 146) • EnS_DrvPulse (Seite 148)
	FB "Kommunikationstreiber"	Beinhaltet die Funktionsbausteine mit Treiberfunktionalität (Treiberbausteine) für die unterstützte Messhardware Die Treiberbausteine für die Messhardware werden in Form einer Energy Support Library (EnSL) (Seite 24) über die Produkt-DVD SIMATIC Energy Suite bereitgestellt und müssen separat zusätzlich installiert werden (sind nicht Bestandteil der Energy Suite-Bibliothek). Liefert die Energierohdaten zum Verarbeitungsschritt "Aufbereitung der Energiedaten"
	Energieobjekte	Ein Energieobjekt beschreibt in Energy Suite allgemein eine Messstelle (z. B. ein Messgerät oder einen Analogwert). Beim Engineering stellt das Energieobjekt die zentrale Komponente dar, an der alle Einstellungen vorgenommen werden (z. B. die Zuweisung der Archivierungsperiode und der Energiedatenquelle sowie Einstellungen für die Verarbeitung der Energierohdaten).
	Aufbereitung der Energiedaten	Dient zur Berechnung, Zeitsynchronisation und Organisation der Energierohdaten (z. B. Verdichtung und periodische Zeitstempelung der Energiewerte) Übergibt die aufbereiteten Energiemomentanwerte und die aufbereiteten periodischen Energiewerte mit Zeitstempel an den Verarbeitungsschritt "Bereitstellung der Energiewerte" Folgende Energy Suite-Bausteine werden verwendet: • EnS_CalcEnergyData (Seite 136) • EnS_TimeSync (Seite 157) • EnS_Organization (Seite 156)
	Bereitstellung der Energiewerte	Bereitstellung von: • Aufbereitete Energiemomentanwerte an "Visualisierung - HMI" und "Visualisierung - SCADA" • Aufbereitete periodische Energiewerte mit Zeitstempel an "Pufferung der periodischen Energiewerte" Folgende Energy Suite-Bausteine werden verwendet: • EnS_EnergyDataBasic (Seite 152) • Ens_EnergyDataAdv (Seite 150) • Ens_EnergyDataAdvMinMax (Seite 151)
	Pufferung der periodischen Energiewerte	Pufferung der aufbereiteten periodischen Energiewerte mit Zeitstempel von mehreren Energieobjekten Liefert die Energiewerte an: • Archivierung - WinCC Professional (Variablenarchiv) oder

Bereich	Komponente/ Verarbeitungsschritt	Beschreibung
		• Archivierung - S7-CPUs (SIMATIC Memory Card) Folgende Datenbausteine (DBs) von Energy Suite werden verwendet: • EnS_BufferDB (Seite 164) • EnS_BufferWorkDB (Seite 165) • EnS_BufferRead (Seite 131) • EnS_BufferWrite (Seite 133)
Ausgabe - Energiedaten visualisieren und auswerten (Seite 93)	Visualisierung - HMI	Visualisierung der Energiemomentanwerte auf SIMATIC Comfort Panel oder PC mit WinCC Applikationen (Basic, Comfort oder Advanced)
	Visualisierung - SCADA	Visualisierung der Energiemomentanwerte oder der gepufferten periodischen Energiewerte auf dem SCADA-System WinCC Professional
	Archivierung - WinCC Professional (Variablenarchiv)	Archivierung gepufferter periodischer Energiewerte im Variablenarchiv des SCADA-Systems WinCC Professional Folgender Energy Suite-Baustein wird verwendet: • EnS_ArchiveHMI (Seite 124)
	Archivierung - S7-CPUs (SIMATIC Memory Card)	Archivierung gepufferter periodischer Energiewerte als Data Logs (CSV-Dateien) auf in S7-1500 CPU gesteckter SIMATIC Memory Card Folgender Energy Suite-Baustein wird verwendet: • EnS_ArchiveDataLog (Seite 116)

1.3 Handlungsübersicht

Voraussetzung

Um die folgenden Projektierungsschritte durchzuführen, müssen in STEP 7 (TIA Portal) folgende Voraussetzungen erfüllt sein:

- Das Optionspaket "SIMATIC Energy Suite" ist installiert.
  Weitere Informationen hierzu finden Sie in der Installationsanleitung zu Energy Suite auf der DVD.
- Das STEP 7-Projekt mit einer S7-1500 CPU ist angelegt und konfiguriert.

Bei Einsatz von Messhardware als Energiedatenquelle zusätzlich:

- Die Energy Support Library der zum Einsatz kommenden Messhardware ist installiert.
  Weitere Informationen hierzu finden Sie in der Produktdokumentation zur Messhardware.
- Die Messhardware ist im STEP 7-Projekt angelegt und konfiguriert.
  Weitere Informationen hierzu finden Sie in der Produktdokumentation zur Messhardware.

Für die Visulalisierung bzw. die Archivierung in Variablenarchiven zusätzlich:

- Die PC-Station mit WinCC Professional bzw. WinCC Basic, Comfort oder Advanced ist im STEP 7-Projekt angelegt und konfiguriert.

Projektierungsschritte

Die folgende Tabelle zeigt eine Übersicht der wichtigsten Projektierungsschritte für das Energiedaten-Management mit den Energy Suite-Komponenten:

Thema	Schritt		Beschreibung
Energiedaten liefern	1	Energiedaten liefern (Seite 21)	Angeschlossene Messhardware und/oder PLC-Variablen (z. B. Werte aus Datenbausteinen und/oder aus dem Prozessabbild) anlegen und projektieren, damit sie später als Energiedatenquelle für die Energieobjekte parametrierbar sind
Energiedaten verarbeiten	2.1	Energieobjekttabelle anlegen (Seite 33)	Energieobjekttabellen dienen zum Anlegen und Verwalten der Energieobjekte. Mehrere Energieobjekttabellen ermöglichen die Abbildung und Strukturierung der Energieobjekte, z. B. entsprechend der Messhierachie Ihrer Produktionsanlage.
	2.2	Energieobjekte zu Energieobjekttabellen hinzufügen (Seite 34)	Um die Energiedaten aus einer Energiedatenquelle zu erfassen und zu verarbeiten, legen Sie Energieobjekte in den Energieobjekttabellen an.
	2.3	Eigenschaften der Energieobjekte einstellen (Seite 37)	Jedes Energieobjekt stellt einen Messstelle Ihrer Produktionsanlage dar und muss entsprechend parametriert werden: • Energiedatenquelle zuweisen • Verarbeitung der Eingangsmessdaten festlegen • Energiewerte für Archivierung aktivieren und Archivierungsperiode zuweisen • Freigabe der Energieobjekte für das Energieprogramm prüfen

Thema	Schritt		Beschreibung
Energiedaten puffern und archivieren	3.1	Archivierungsperioden definieren (Seite 51)	An zentraler Stelle verwalten Sie die periodischen Zeitabschnitte (Archivierungsperioden), in denen die Energierohdaten zu einem aufbereiteten periodischen Energiewert mit Zeitstempel berechnet und archiviert werden sollen. Jedem Energieobjekt kann eine gewünschte Archivierungsperiode zugewiesen werden.
	3.2	Pufferung der periodischen Energiewerte definieren (Seite 61)	Für die Pufferung der periodischen Energiewerte legen Sie die Größe der Datenbausteine (Puffer-DBs) fest. Die Anzahl der Puffer-DBs ergibt sich aus der Anzahl der angelegten Energieobjekttabellen (für jede Energieobjekttabelle ein Puffer-DB).
	3.3	Archivierung auf S7-CPU (SIMATIC Memory Card) festlegen (Seite 55)	Für die Archivierung der periodischen Energiewerte auf einer in der S7-CPU gesteckten SIMATIC Memory Card legen Sie den Namen, die Anzahl und die Größe der Data Logs fest. In den Data Logs sind die Energiewerte enthalten.
	3.4	Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)	Für die Archivierung der periodischen Energiewerte im Variablenarchiv von WinCC Professional aktivieren Sie das entsprechende Optionskästchen in den Energieprogrammeinstellungen von Energy Suite. Die Einrichtung der Variablenarchive von WinCC Professional muss dafür in den Einstellungen von WinCC Professional erfolgen.
Energieprogramm generieren	4.1	Allgemeine Einstellungen für das Energieprogramm festlegen (Seite 67)	Vor der Generierung des Energieprogramms stellen Sie die Anzahl der azyklischen parallelen Kommunikationskanäle und die Anlaufzeit der Bausteine im Energieprogramm ein. Zusätzlich definieren Sie die Meldungsunterdrückung.
	4.2	Freigabe der Energieobjekte für das Energieprogramm prüfen und ggf. anpassen (Seite 40)	Spätestens vor dem Generieren des Energieprogramms entscheiden Sie, welche Energieobjekte (und damit welche Messstelle) im Energieprogramm berücksichtigt werden sollen. Die Aktivierung/Deaktivierung der Energieobjekte können Sie komfortabel in den Energieobjekttabellen vornehmen.
	4.3	Energieprogramm generieren (Seite 70)	Die Generierung des PLC-Programms für die Energiedatenerfassung und -verarbeitung (kurz: Energieprogramm) starten Sie aus der Projektnavigation. Während des Generiervorgangs werden automatisch die notwendigen Bausteine und Datentypen aus der Energy Suite-Bibliothek und der Messhardware-Bibliothek im Energieprogramm verwendet und entsprechend der projektierten Energieobjekte miteinander verschaltet. Das generierte Energieprogramm wird unterhalb der S7-CPU im Ordner "Programmbausteine" in dem Unterordner "Energy Suite - program" angelegt.
	4.4	PLC-Programm übersetzen und laden (Seite 87)	Nach der Generierung des Energieprogramms übersetzen Sie das gesamte PLC-Programm und laden es auf das Zielsystem S7-1500 CPU.
	4.5	Energieprogramm testen (Seite 88)	Um Ihre Einstellungen zu überprüfen, öffnen Sie im Energieprogramm den betreffenden Datenaustein des jeweiligen Energieobjekts und beobachten dessen Online-Werte über die Funktion "Alle beobachten".

Thema	Schritt		Beschreibung
Energiedaten visualisieren und auswerten (Seite 93)	5.1	PC-Station mit WinCC Professional konfigurieren	Für das SCADA-System benötigen Sie in Ihrem Projekt eine PC-Station mit der SIMATIC HMI Applikation "WinCC RT Professional".
	5.2	Verbindung zum SCADA-System anlegen	Um den Datentransfer zwischen der S7-1500 CPU und dem SCADA-System zu ermöglichen, projektieren Sie in STEP 7 (TIA Portal) eine HMI-Verbindung zur PC-Station mit WinCC Professional.
	5.3	Auf Engineering System für Archivierung projektierte Energiewerte auf PC-Station einlesen und anzeigen (Seite 96)	Auf der PC-Station mit WinCC Professional bietet Energy Suite eine Übersicht über alle Energieobjekte und den periodischen Energiewerten an, die Sie im Engineering System zur Archivierung im Variablenarchiv von WinCC Professional projektiert haben. Über die Funktion "Aktualisieren" erfolgt ein Abgleich mit den Engineering Daten der von der PC-Station mit WinCC Professional erreichbaren S7-1500 CPUs.
	5.4	SCADA-System projektieren (Seite 99)	Projektierung der PC-Station mit WinCC Professional: • Für den Datentransfer zum SCADA-System erzeugen Sie die HMI-Variablen für die Datenbausteine zur Pufferung (Ordner "HMI-Variablen"). • Für die Archivierung generieren Sie das Variablenarchiv mit den Archivvariablen (Energiewerte) und richten die Archivsegmente für die Archivierung (Seite 102) der periodischen Energiewerte ein (Ordner "Archive"). • Für die Visualisierung der erfassten Energiedaten im Runtime können Sie Vorlagenbilder und Bildelemente aus dem Siemens Industry Online Support in die WinCC Applikation einbinden (Ordner "Bilder").
	5.5	PC-Station mit WinCC Professional übersetzen und laden	Anschließend übersetzen Sie die Projektierung für die gesamte PC-Station und laden sie auf das SCADA-System.
	5.6	PLC-Programm und SCADA-System starten (Runtime)	Nachdem sowohl WinCC RT Professionell auf dem SCADA-System als auch die PLC-Programme auf den S7-1500 CPUs gestartet sind, ist das Archivieren betriebsbereit. Die gewählten periodischen Energiewerte werden automatisch archiviert. Das Archivieren wird stetig ausgeführt, solange Runtime aktiv ist.

1.4 Leistungsmerkmale und Konventionen

Kompabilität von Energy Suite mit CPU und FW

Die folgende Tabelle zeigt, welche CPU mit welcher Energy Suite-Version kompatibel ist.

CPU / ET 200SP CPU / ET 200SP Open Controller	FW	Energy Suite-Version
S7-1500 CPU	≥ V2.0	V14.0 V14.0 SP1 V15.0 V15.1

Leistungsmerkmale von Energy Suite

Die folgende Tabelle zeigt die Leistungsmerkmale von Energy Suite:

Merkmal	Wert
Maximale Anzahl der lizenzierbaren Energieobjekte pro S7-1500 CPU	500
Maximale Anzahl der Energieobjekttabellen pro S7-1500 CPU	10
Maximale Anzahl S7-1500 CPUs für geichzeitige Archivierung	10
Maximale Anzahl gleichzeitig archivierbarer Energieobjekte (WinCC Professional) Empfehlung: verteilt auf maximal 10 S7-1500 CPUs	2000
Minimal einstellbare Archivierungsperiode	1 min
Wertebereich einer Archivierungsperiode	Ganzzahliger Teil einer Stunde / eines Tages z. B.: • 1 Minute • 5 Minuten • 15 Minuten • 12 Stunden Ungültig sind z. B.: • 7 Minuten • 11 Minuten • 5 Stunden
Maximal einstellbare Archivierungsperiode	24 h
Typische Archivierungsperiode für elektrische Größen	15 min
Typische Archivierungsperiode für nichtelektrische Größen	60 min

Namenskonventionen

Die folgende Tabelle zeigt die Namenskonventionen für das Engineering mit Energy Suite:

Objekt	Wertebereich für Namen	Gültigkeitsbereich
PLC	• Nicht erlaubte Zeichen: – : ? " ' \ * % – <Leerzeichen> – <Zeilenvorschub> – <Zeilenumbruch>	• Die Namenskonvention der PLC gilt bei der Archivierung der Energiedaten der Energieobjekte auf PC-Station mit WinCC Professional.
Energieobjekt	• 1 - 32 Zeichen • Nicht erlaubte Zeichen: – : ? ' * " \ $ % § ° – <Leerzeichen> – <Zeilenvorschub> – <Zeilenumbruch>	• Projekt-weit eindeutig bei Archivierungstyp "WinCC Professional (Variablenarchiv)"
Energieobjekttabelle	• 1 - 32 Zeichen • Nicht erlaubte Zeichen: – <Zeilenvorschub> – <Zeilenumbruch>	• Projekt-weit eindeutig bei Archivierungstyp "WinCC Professional (Variablenarchiv)"
Archivierungsperiode	• 1 - 32 Zeichen • Nicht erlaubte Zeichen: – : ? ' * " \ % § ° – <Leerzeichen> – <Zeilenvorschub> – <Zeilenumbruch>	-
Data Log	• 1 - 29 Zeichen • Erlaubte Zeichen: – 0 ... 9 – a ... z – A ... Z – - – _	-

Hinweis

Eindeutige Namen

Die Namen der Objekte müssen mindestens CPU-weit eindeutig sein. Einen Namen, der bereits für einen Baustein, eine PLC-Variable oder eine Konstante innerhalb der CPU vergeben ist, können Sie nicht für ein neues Objekt verwenden. Groß- und Kleinschreibung wird bei der Prüfung der Eindeutigkeit nicht unterschieden.

Energiedaten liefern 2

Energiedatenquellen liefern die Energiedaten zur Verarbeitung (Erfassung, Aufarbeitung, Bereitstellung) an das Energieobjekt (Seite 31). Folgende Datenquellen sind als Energiedatenquelle eines Energieobjekts parametrierbar:

- Eingangsmessdaten von angeschlossener Messhardware (Seite 24)
- Energiedaten aus dem Prozessabbild oder dem Merker-Bereich der CPU (PLC-Variablen) (Seite 27)
- Energiedaten aus Datenbausteinen (DB-Variablen) (Seite 29)

Der Datenaustausch der periodischen Energiedaten (Impulssignal, Energiezählwert, Leistungswert), aus denen die Basisenergiedaten berechnet werden, erfolgt zyklisch über das Prozessabbild. Die Messhardware liefert Energiedaten an Energieobjekte unter Verwendung zyklischer und azyklischer Kommunikation. Erweiterten Energiedaten, z. B. Spannungs-, Strom- oder Frequenzwert, werden in einem Datensatz von der Messhardware an das Energieobjekt übergeben. Dabei ist die maximale Anzahl der parallelen azyklischen Kommunikationskanäle auf 10 begrenzt.

Die Zuweisung der Energiedatenquellen für das Energieprogramm nehmen Sie bei der Parametrierung der Energieobjekte vor. Jedem Energieobjekt wird eine Energiedatenquelle zugewiesen.

2.1 Wissenswertes zu Energiedaten

Es werden folgende Energiedaten unterschieden:

- Basisenergiedaten
- Erweiterte Energiedaten

Basisenergiedaten

Basisenergiedaten (englisch: Basic) beinhalten folgende Energiedaten:

- Periodisch berechneter Energieverbrauch
- Durchschnittliche Leistung pro Archivierungsperiode
- Aktuelle Werte für Energieverbrauch und Leistungsaufnahme

Basisenergiedaten sind abrechnungsrelevante Energiedaten, die zur Darstellung der Energieflüsse in einer Anlage notwendig sind.

Basisenergiedaten werden vorwiegend mittels zyklischer Kommunikation über das Prozessabbild übertragen.

Erweiterte Energiedaten

Erweiterte Energiedaten (englisch: Advanced) werden nur aus der Messhardware gelesen, die diese Daten bereitstellen können. Erweiterte Energiedaten beinhalten folgende Energiedaten:

- Phasenspezifische Strom-, Spannungs-, Leistungs- und Leistungsfaktorwerte
- Gesamt-Leistungs-, Gesamt-Energie- und Gesamt-Leistungsfaktorwerte
- Frequenzwerte

Ob und in welchem Umfang die erweiterten Energiedaten bereitgestellt werden, hängt von der jeweiligen Messhardware ab. Zusätzlich zu dem Eingangsmesswert stellt die Messhardware bzw. der Treiberbaustein auch die **Minimal- und Maximalwerte** der erweiterten Energiedaten bereit, z. B. für die Ausgabe in einem Bildbaustein in WinCC Professional.

Erweiterte Energiedaten werden mittels azyklischer Kommunikation in einem Datensatz übertragen.

Zyklische Kommunikation

Zyklische Kommunikation nutzen Sie für abrechnungsrelevante Energiedaten (z. B. Leistung, Energiezähler), die für Überwachungs- und Berechnungsfunktionen in der CPU verarbeitet werden. Diese Daten werden schnell und zuverlässig zyklisch über das Prozessabbild eingelesen.

Weitere Informationen zum Prozessabbild finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter dem Stichwort "Prozessabbild".

Azyklische Kommunikation

Azyklische Kommunikation nutzen Sie für visualisierungsrelevante Energiedaten (z. B. Strom, Spannung), die an einem SIMATIC Comfort Panel oder PC dargestellt werden. Diese Daten werden z. B. für die Visualisierung bei Bildaufschlag für die Dauer eines Bildaufschlags kommuniziert.

Dabei werden die Energiedaten in einem Datensatz mit der Anweisung "RDREC" aus dem entsprechenden Modul ausgelesen. Weitere Informationen zur Anweisung "RDREC" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter dem Stichwort "RDREC".

Die Treiberbausteine für die Messhardware rufen die Anweisung "RDREC" für die azyklische Kommunikation intern auf.

Hinweis

Nutzen Sie die beiden Kommunikationsarten je nach Anwendungsfall. Achten Sie darauf, dass die Kommunikationskanäle nicht überlastet werden.

2.2 Energiedaten von Messhardware

Es gibt eine Vielzahl von Geräten, die Energiedaten liefern. Neben den Messgeräten (z. B. ET 200SP AI EnergyMeter 480VAC ST, SENTRON PAC), deren einzige Aufgabe das Messen von Energie ist, stellen viele Automatisierungsgeräte (z. B. Frequenzumrichter oder Schutz- und Schaltgeräte) Energiedaten zusätzlich zu ihren eigentlichen Automatisierungsfunktionen (z. B. Drehzahlregelung) bereit. Im Folgenden werden diese Geräte als Messhardware bezeichnet.

Übersicht der Messhardware für Energy Suite

Die folgende Tabelle zeigt beispielhaft einige Geräte als Messhardware, die von SIMATIC Energy Suite unterstützt werden. Für diesen Zweck stehen für die Messhardware über eine Energy Support Library (EnSL) spezifische Funktionsbausteine (FBs) mit Treiberfunktionalität zur Verfügung.

Messhardware	Beschreibung
AI EnergyMeter 480VAC ST	Analogeingabemodul für das dezentrale Peripheriesystem SIMATIC ET 200SP zur Messung relevanter Netzparameter im Spannungsbereich bis maximal AC 480V Leiter-Leiter-Spannung
AI EnergyMeter 480VAC HF	Analogeingabemodul für das dezentrale Peripheriesystem SIMATIC ET 200SP zur Messung relevanter Netzparameter im Spannungsbereich bis maximal AC 480V Leiter-Leiter Spannung. Erweiterte Einsatzfälle und Funktionalität; Strommessung mittels Rogowski-Spulen oder Strom- und Spannungswandler. Das AI EnergyMeter 480VAC HF gibt es in zwei Ausprägungen: CT und RC.
SENTRON PAC3200/PAC4200	Multifunktions-Messgerät zur Messung und Anzeige relevanter Netzparameter in der Niederspannungs-Energieverteilung
SENTRON 3VA (8er ETU)	Kompaktleistungsschalter für die Niederspannungs-Energieverteilung zum Schutz von z. B. Produktionsanlagen vor elektrischen Schäden
SINAMICS AC/AC	Umrichter für einfache bis mittlere Anforderungen zum wirtschaftlichen und sparsamen Betrieb von z. B. Pumpen, Lüftern oder Kompressoren
SIMOCODE pro V PN	Motorschutz- und Steuergerät für Motormanagement-Aufgaben wie z. B umfassende Schutz-, Überwachungs-, Sicherheits- und Steuerfunktionen zwischen Motorabzweig und Automatisierungssystem
Motorstarter ET 200SP/M200D	Motorstarter zum Schalten und Schützen von Drehstrom-Asynchronmotoren und einphasigen Wechselstrommotoren
Sanftstarter 3RW44	Sanftstarter zum Begrenzen des Anlaufstroms und des Anlaufmoments. So lassen sich sowohl mechanische Belastungen als auch Netzspannungseinbrüche zuverlässig vermeiden.
Sanftstarter 3RW55	Neue Generation von Sanftstartern zum Begrenzen des Anlaufstroms und des Anlaufmoments. So lassen sich sowohl mechanische Belastungen als auch Netzspannungseinbrüche zuverlässig vermeiden.

Gerätekonfiguration der Messhardware in STEP 7 (TIA Portal)

Die Gerätekonfiguration der Messhardware erfolgt in STEP 7 (TIA Portal). Um die Messhardware zur Hardware-Konfiguration Ihres Projekts hinzuzufügen, gibt es folgende Möglichkeiten:

- Gerät direkt aus dem Hardware-Katalog in die Netz- bzw. Topologiesicht übernehmen
- Gerät, das nicht im Hardware-Katalog vorhanden ist, über eine **GSD-Datei** (Gerätestammdaten-Datei) oder ein **HSP** (Hardware Support Packages) hinzufügen

Falls Sie eine Messhardware konfigurieren möchten, die nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller bereitgestellte GSD-Datei bzw. HSP zuerst installieren. Über GSD-Dateien bzw. HSP installierte Messhardware wird im Hardware-Katalog angezeigt und kann anschließend selektiert und im Projekt konfiguriert werden.

Weitere Informationen zur Konfiguration eines Geräts über GSD-Datei bzw. HSP finden Sie in der Online-Hilfe des TIA Portal Informationssystems.

Treiberbausteine der Messhardware für STEP 7 (TIA Portal)

Die Energiedaten (Eingangsmessdaten) der Messhardware werden auf unterschiedliche Weise bereitgestellt (z. B. Unterschiede im Datenformat oder in der Datensatzadresse).

Zusätzlich zu den Energy Suite-Bausteinen stehen für die Messhardware Hardware-spezifische Funktionsbausteine (FBs) mit Treiberfunktionalität über die **Energy Support Library (EnSL)** zur Verfügung. Diese Hardware-spezifischen Funktionsbausteine (im folgenden Treiberbausteine genannt) bilden die Schnittstelle zwischen der speziellen Messhardware und Ihrem Automatisierungssystem.

Mithilfe der Treiberbausteine werden die Energiedaten aus der Messhardware ausgelesen und vereinheitlicht dem Automatisierungssystem bereitgestellt. Auf diese Weise werden die Energierohdaten einfacher und schneller Ihrem PLC-Programm und Energy Suite zugeführt. Durch die Energie Suite-Bausteine werden die Energierohdaten anschließend aufbereitet und verarbeitet.

Energy Support Library (EnSL)

Nehmen Sie die Installation der Energy Support Library entsprechend der Installationsanleitung zur Messhardware vor.

Nach der erfolgreichen Installation der Energy Support Library:

- ist die Messhardware-Bibliothek mit den Treiberbausteinen in STEP 7 (TIA Portal) als globale Bibliothek verfügbar. Die Treiberbausteine können im PLC-Programm verwendet werden.
- kann die Messhardware bei der Auswahl der Energiedatenquelle einem Energieobjekt zugewiesen werden.

Weitere Informationen zur Verwendung von globalen Bibliotheken finden Sie in der Online-Hilfe des TIA Portal Informationssystems.

Art der Kommunikation

Die CPU kann die Energiedaten aus der Messhardware auf zwei Arten auslesen: mit zyklischer oder azyklischer Kommunikation. Die zyklischen Energiedaten werden über das Prozessabbild und die azyklischen Energiedaten in einem Datensatz von der Messhardware an Energy Suite übergeben.

Die Art der Kommunikation zwischen der CPU und der Messhardware hängt von der verwendeten Messhardware und deren Treiberbausteinen ab. Normalerweise werden die Basisenergiedaten mittels zyklischer Kommunikation, die erweiterten Energiedaten mittels azyklischer Kommunikation übertragen:

- Zyklische Kommunikation nutzen Sie für abrechnungsrelevante Energiedaten (z. B. Leistung, Energiezähler), die für Überwachungs- und Berechnungsfunktionen in der CPU verarbeitet werden.
- Azyklische Kommunikation nutzen Sie für visualsierungsrelevante Energiedaten (z. B. Strom, Spannung), die an einem SIMATIC Comfort Panel oder PC dargestellt werden.

Weitere informationen zur Kommunikation zwischen der CPU und der verwendeten Messhardware finden Sie in der Dokumentation der jeweiligen Messhardware.

Hinweis

Achten Sie beim Einsatz mehrerer Messhardware auf die Auslastung der Kommunikationskanäle, die von der verwendeten CPU unterstützt werden.

Siehe auch

Energiedatenquelle zuweisen (Seite 41)

2.3 Energiedaten aus dem Prozessabbild oder Merker-Bereich der S7-CPU

PLC-Variablen

PLC-Variablen können als Energiedatenquelle für Energieobjekte genutzt werden. Dazu legen Sie neue PLC-Variablen an oder verwenden die im PLC-Programm bereits vorhandenen PLC-Variablen. Je nach gewünschtem Energiedatentyp (Impulssignal, Leistungswert oder Energiezählwert) weisen Sie der angelegten PLC-Variable einen entsprechenden Datentyp zu. Den Datentyp der PLC-Variable vergeben Sie bei der Deklaration in der PLC-Variablentabelle. Die Energiedatentypen weisen Sie den Energiedatenquellen bei der Projektierung des Energieobjekts zu.

Beispiele

- Ein Messgerät, das nicht von der Energy Support Library unterstützt wird, liefert einen Leistungswert und einen Energiezählwert. Diese Werte weisen Sie den PLC-Variablen zu.
- 10 Lüftermotoren besitzen jeweils eine Nennleistung von 1,5 kW. Die Motoren werden durch eine PLC gesteuert. In der PLC wird anhand der Schaltzustände und der Nennleistung der theoretische Verbrauch abgeleitet. Wenn z. B. 5 Motoren eingeschaltet sind, dann ergibt sich eine Leistungsaufnahme von 7,5 kW. Diese Berechnung erfolgt in der PLC und kann der PLC-Variable zugewiesen werden.

Übersicht der relevanten Datentypen für PLC-Variablen

Die folgende Tabelle zeigt die relevanten Datentypen für PLC-Variablen zur Energiedatenerfassung:

Datentyp	Beschreibung
BOOL	Dient der Erfassung eines Impulssignals Aus einem erfassten binären Impulssignal wird mithilfe des Normierungsfaktors der resultierende Energieverbrauch berechnet. Beispiel: Bei einem Normierungsfaktor von 2 und der physikalischen Einheit "Wh" ergeben 3 Impulssignale einen Energieverbrauch von 6 Wh.
REAL, LREAL	Dient der Erfassung eines Leistungswerts Der erfasste Leistungswert wird normiert und der resultierende Energieverbrauch berechnet. Beispiel: Eine durchschnittliche Leistungsaufnahme von 180 kW über 10 Sekunden entpricht einem Energieverbrauch von 0,5 kWh bzw. 500 Wh.
SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT	Dient der Erfassung eines Energiezählwerts Aus der Änderung des absoluten Energiezählwerts wird mit Hilfe des Normierungsfaktors der Energieverbrauch und der Momentanwert der Leistungsaufnahme berechnet. Beispiel: Der Zählerstand eines Energiezählers verändert sich in 10 Sekunden von 6.734.392 Wh auf 6.734.414 Wh. Dies entspricht einem Energieverbrauch von 22 Wh und einem Momentanwert der Leistungsaufnahme von 7,92 kW.
REAL, LREAL	

PLC-Variablen für Energiedatenerfassung deklarieren

Um die PLC-Variablen für die Energiedatenerfassung zu deklarieren, legen Sie die PLC-Variablen in einer PLC-Variablentabelle nach der standardmäßigen Vorgehensweise an. Je nach Energiedatentyp weisen Sie den PLC-Variablen den entsprechenden Datentyp zu.

Weitere Informationen zur Vorgehensweise finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Variablen in der PLC-Variablentabelle deklarieren".

Siehe auch

Energiedatenquelle zuweisen (Seite 41)

2.4 Energiedaten aus Datenbausteinen

DB-Variablen

DB-Variablen können als Energiedatenquelle für Energieobjekte genutzt werden. Dazu legen Sie neue DB-Variablen in Datenbausteinen an oder verwenden die im PLC-Programm bereits vorhandenen DB-Variablen.

Je nach gewünschtem Energiedatentyp (Impulssignal, Leistungswert oder Energiezählwert) weisen Sie der angelegten DB-Variable den entsprechenden Datentyp zu. Den Datentyp einer DB-Variable oder eines Elements vergeben Sie in der Deklarationstabelle für Datenbausteine. Die Energiedatentypen weisen Sie den Energiedatenquellen bei der Projektierung des Energieobjekts zu.

Übersicht der Datentypen für DB-Variablen

Die folgende Tabelle zeigt die verwendbaren Datentypen für DB-Variablen zur Energiedatenerfassung:

Datentyp	Beschreibung
BOOL	Dient der Erfassung eines Impulssignals Aus einem erfassten binären Impulssignal wird mithilfe des Normierungsfaktors der resultierende analoge Energiewert berechnet. Beispiel: Bei einem Normierungsfaktor von 2 und der physikalischen Einheit "Wh" ergeben 3 Impulssignale einen Energieverbrauch von 6 Wh.
REAL, LREAL	Dient der Erfassung eines Leistungswerts Der erfasste Leistungswert wird normiert und der resultierende Energiewert berechnet. Beispiel: Eine Leistungsaufnahme von 180 kW über 10 Sekunden entpricht einem Energieverbrauch von 0,5 kWh bzw. 500 Wh.
SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT	Dient der Erfassung eines Energiezählwerts Aus der Änderung des absoluten Energiezählwerts wird mit Hilfe des Normierungsfaktors der Energieverbrauch und die aktuelle mittlere Leistungsaufnahme berechnet. Beispiel: Der Zählerstand eines Energiezählers verändert sich in 10 Sekunden von 6.734.392 Wh auf 6.734.414 Wh. Dies entspricht einem Energieverbrauch von 22 Wh und einer mittleren Leistungsaufnahme von 7,92 kW.
REAL, LREAL	

DB-Variablen für Energiedatenerfassung deklarieren

Um die DB-Variablen für die Energiedatenerfassung zu deklarieren, legen Sie entsprechende Datenbausteine an und definieren ihre Struktur nach der standardmäßigen Vorgehensweise.

Weitere Informationen zur Vorgehensweise finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Eigenschaften von Variablen in Datenbausteinen bearbeiten".

Siehe auch

Energiedatenquelle zuweisen (Seite 41)

Energiedaten verarbeiten 3

3.1 Energieobjekte und Energieprogramm

Für die Verarbeitung der Energierohdaten definieren Sie Energieobjekte in STEP 7 (TIA Portal) in der Projektnavigation unterhalb der S7-1500 CPU. Die Energieobjekte dienen der verarbeitungsspezifischen Parametrierung der Energierohdaten und der anschließenden Generierung eines Energieprogramms.

Nach der erfolgreichen Installation von SIMATIC Energy Suite gibt es hierfür den neuen Ordner "Energieobjekte" in der Projektnavigation. Das folgende Bild zeigt den Ordner "Energieobjekte" in der Projektnavigation.

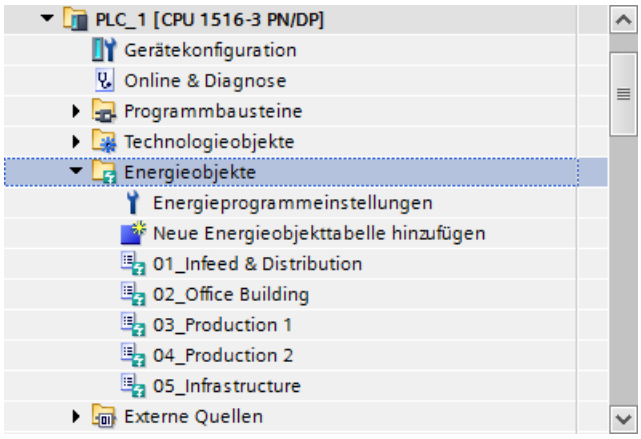

Energieobjekte

Jedes Energieobjekt stellt eine Messstelle Ihrer Produktionsanlage dar. Über die Parametrierung des Energieobjekts definieren Sie die Verarbeitung der Eingangsmessdaten zu auswertbaren Energiewerten.

Zur Festlegung der periodischen Berechnung von Energiewerten weisen Sie jedem Energieobjekt eine Archivierungsperiode zu. Damit definieren Sie in welchen Zeitabständen die erfassten Energierohdaten zu einem periodischen Energiewert verdichtet und mit ihrem Zeitstempel archiviert werden. Häufig wird ein elektrischer Energiewert in 15 min-Perioden und ein nichtelektrischer Energiewert (z. B. Gas) in 60 min-Perioden archiviert. Sie können mehrere Archivierungsperioden unabhängig von den Energieobjekten zentral erstellen und verwalten.

Energieobjekte legen Sie an und verwalten Sie in Energieobjekttabellen. Mehrere Energieobjekttabellen ermöglichen die Strukurierung der Energieobjekte z. B. entsprechend der Hierarchie der Messstellen in Ihrer Produktionsanlage.

Weitere Informationen zu der möglichen Anzahl von Energieobjekten und Energieobjekttabellen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".

Energieprogramm

Auf Basis der parametrierten Energieobjekte und der Archivierungsperioden, wird ein PLC-Programm zur Erfassung, Verarbeitung und Archivierung von Energiedaten generiert - im Folgenden als Energieprogramm bezeichnet.

Bei der Generierung werden Bausteine aus der mitgelieferten Energy Suite-Bibliothek sowie aus der Messhardware-Bibliothek (Energy Support Library) automatisch erzeugt und entsprechend der Konfiguration der Energieobjekte verschaltet. Vor dem Start der Generierung des Energieprogramms haben Sie die Möglichkeit über die Freigabe der Energieobjekte, diejenigen Messstellen auszuwählen, die in dem Energieprogramm berücksichtigt werden sollen.

Nach der Generierung ist das Energieprogramm in der Projektnavigation unterhalb der CPU im Ordner "Programmbausteine" > "Energy Suite - program" mit allen notwendigen Bausteinen angelegt.

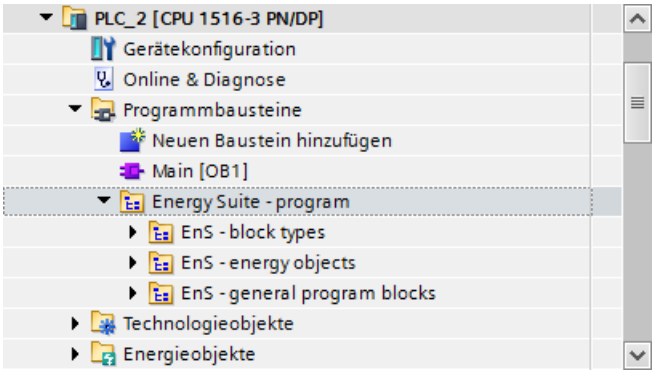

Regeln zur Programmierung

Hinweis

Das Präfix "EnS_" ist ein reserviertes Präfix für Energy Suite.

Stellen Sie vor der Generierung des Energieprogramms sicher, dass das Präfix "EnS_" nicht in Ihrem bestehenden PLC-Programm (Anwenderbausteinen, Variablen, UDTs oder Technologieobjekten) verwendet wird. Damit vermeiden Sie ein versehentliches Überschreiben durch die Generierung des Energieprogramms.

Hinweis

Der Ordner "Energy Suite - program" ist ein reservierter Bereich für Energy Suite.

Stellen Sie vor der Generierung des Energieprogramms sicher, dass sich im Ordner "Energy Suite - program" keine Teile Ihres bestehenden PLC-Programms befinden. Damit vermeiden Sie ein versehentliches Überschreiben durch die Generierung des Energieprogramms.

3.2 Energieobjekttabelle anlegen

Zuerst legen Sie Energieobjekttabellen unterhalb der CPU im Ordner "Energieobjekte" an. Überlegen Sie sich vorher, wie Sie die Energieobjekte über mehrere Energieobjekttabellen strukturieren möchten. Die Anzahl der Energieobjekttabellen hat Auswirkungen auf die maximale Pufferdauer (Seite 61).

Weitere Informationen zu der möglichen Anzahl der Energieobjekttabellen und zu Namenskonventionen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".

Voraussetzung

- Eine S7-1500 CPU ist im Projekt angelegt.

Vorgehen

Um eine Energieobjekttabelle anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die S7-1500 CPU.
2. Öffnen Sie unterhalb der S7-1500 CPU den Ordner "Energieobjekte".
3. Doppelklicken Sie auf das Objekt "Neue Energieobjekttabelle hinzufügen".
   Eine neue Energieobjekttabelle wird direkt im Ordner "Energieobjekte" angelegt.
4. Wählen Sie im Kontextmenü der neuen Energieobjekttabelle den Befehl "Umbenennen" und geben Sie den gewünschten Namen entsprechend den Namenskonventionen (Seite 18) ein.

Legen Sie bei Bedarf in gleicher Weise weitere Energieobjekttabellen an. Setzen Sie ggf. das Anlegen von Energieobjekttabellen unterhalb von anderen S7-1500 CPUs fort.

3.3 Energieobjekte zu Energieobjekttabellen hinzufügen

Um die Energiedaten zu verarbeiten, legen Sie für die Messstellen Ihrer Produktionsanlage Energieobjekte in den Energieobjekttabellen an. Sie können die Energieobjekte über mehrere Energieobjekttabellen strukturieren.

Weitere Informationen zu der möglichen Anzahl der Energieobjekte und zu Namenskonventionen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".

Voraussetzung

- Eine Energieobjekttabelle im Ordner "Energieobjekte" unterhalb der S7-1500 CPU ist angelegt.

Vorgehen

Um ein Energieobjekt anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation unterhalb der S7-1500 CPU den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in welcher Sie das Energieobjekt angelegen möchten.
   Die Energieobjekttabelle wird im Arbeitsbereich geöffnet.
3. Doppelklicken Sie in der Energieobjekttabelle in der Spalte "Name" auf "<Hinzufügen>".
   Ein neues Energieobjekt wird in der Energieobjekttabelle angelegt.

Legen Sie in gleicher Weise weitere Energieobjekte für alle Energieobjekttabellen an. Setzen Sie ggf. das Anlegen von Energieobjekten unterhalb von anderen S7-1500 CPU fort.

Ergebnis

Jede Zeile einer Energieobjekttabelle repräsentiert ein Energieobjekt. Die wichtigsten Eigenschaften der Energieobjekte können Sie direkt in der Energieobjekttabelle übersichtlich sehen und ggf ändern. Dazu gehören:

- Name des Energieobjekts
- Energiedatenquelle des Energieobjekts
- Freigabestatus des Energieobjekts für das Energieprogramm
- Konfiguration der spezifischen Eigenschaften des Energieobjekts (Verarbeitungseigenschaften)
- Archivierungsperiode des Energieobjekts

Das folgende Bild zeigt eine Energieobjekttabelle im Arbeitsbereich von STEP 7 (TIA Portal):

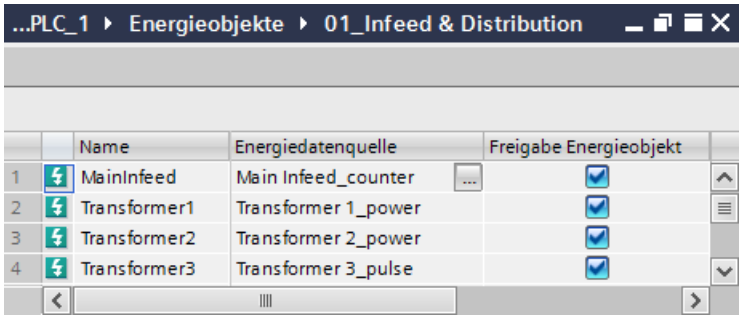

Bei Selektion eines Energieobjekts in der Energieobjekttabelle im Arbeitsbereich werden im Inspektorfenster von STEP 7 (TIA Portal) die spezifischen Eigenschaften des Energieobjekts angezeigt. Die Einstellung dieser Eigenschaften ist in den nachfolgenden Abschnitten beschrieben.

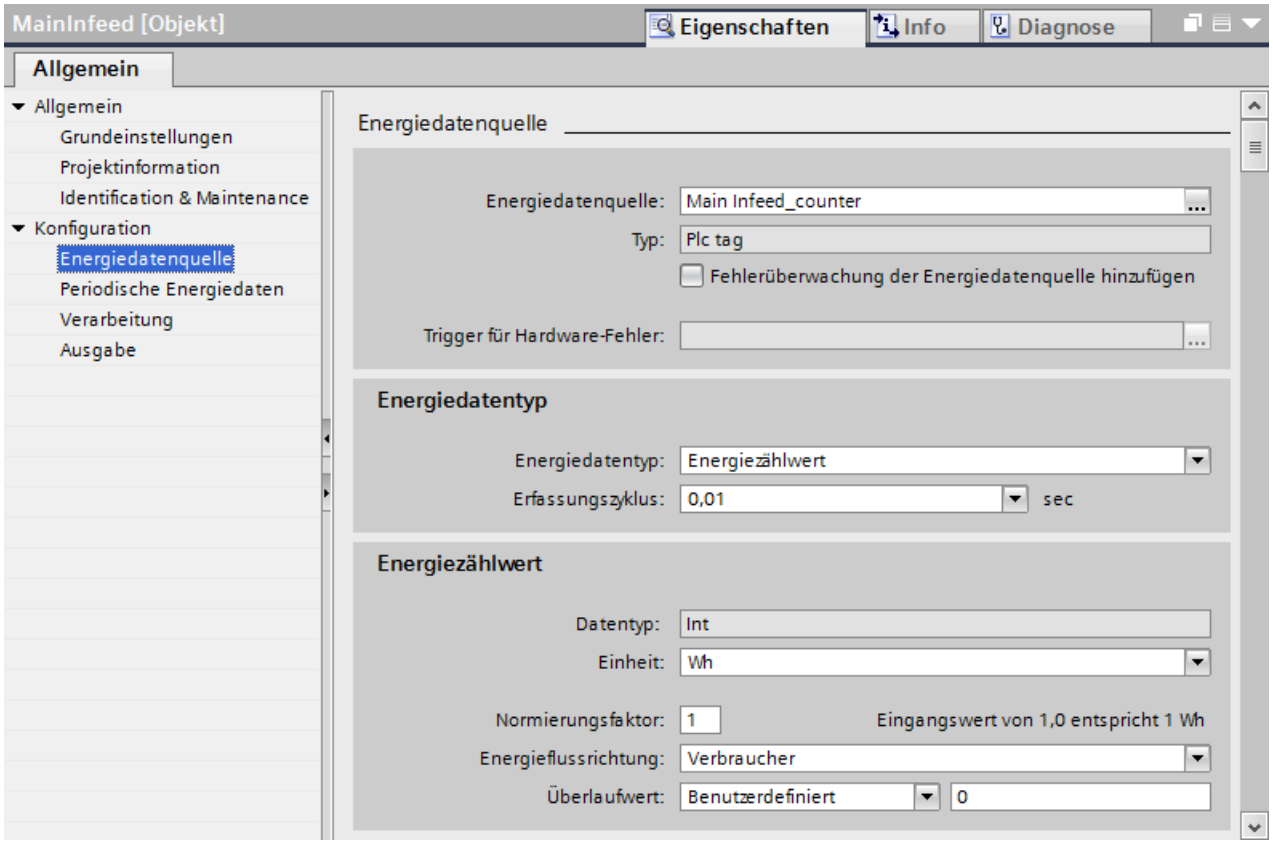

Kopieren von Energieobjekten

Energieobjekte können über das Kontextmenü zwischen den Energieobjekttabellen kopiert werden. Prüfen Sie nach dem Kopiervorgang die Eigenschaften des Energieobjekts und passen Sie bei Bedarf die kopierten Einstellungen an.

Siehe auch

Energieobjekt parametrieren (Seite 37)
Freigabe der Energieobjekte für Energieprogramm ändern (Seite 40)

3.4 Eigenschaften der Energieobjekte einstellen

3.4.1 Energieobjekt parametrieren

Jedes Energieobjekt stellt eine Messstelle Ihrer Produktionsanlage dar und wird von Ihnen entsprechend parametriert. Die Parametrierung eines Energieobjekts nehmen Sie im Inspektorfenster "Eigenschaften" vor.

Hinweis

Alternativ können Sie die wichtigsten Parameter des Energieobjekts in der Energieobjekttabelle anzeigen oder verbergen (über das Kontextmenü des Tabellenkopfs). In der so individuell zusammengestellten Energieobjekttabelle können die Einstellungen der Energieobjekte übersichtlich dargestellt und ebenfalls verändert werden.

Für jedes Energieobjekt legen Sie fest:

- Datenquelle der Energierohdaten
- Parameter für die Erfassung und Verarbeitung der Energiedaten
- Parameter für Archivierung der Energiedaten (z. B. Archivierungsperiode)
- Freigabe des Energieobjekts für das Energieprogramm

Auf Basis der parametrierten Eigenschaften wird das Energieprogramm generiert.

Voraussetzung

- Ein Energieobjekt ist in der Energieobjekttabelle angelegt.

Vorgehen

Um das Inspektorfenster "Eigenschaften" eines Energieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in der das Energieobjekt angelegt ist.
   Die Energieobjekttabelle mit den Energieobjekten wird geöffnet.
3. Klicken Sie auf das gewünschte Energieobjekt in der Energieobjekttabelle.
   Im Inspektorfenster "Eigenschaften" werden alle Parameter des Energieobjekts angezeigt.

Die Parametrierung des Energieobjekts ist in den folgenden Abschnitten detailliert beschrieben.

Siehe auch

Grundeinstellungen (Seite 38)

Energiedatenquelle (Seite 41)

Periodische Energiewerte (Seite 46)

Verarbeitung (Seite 49)

3.4.2 Grundeinstellungen

Die folgende Tabelle zeigt die Grundeinstellungen eines Energieobjekts:

Einstellung	Beschreibung
Name	CPU-weit eindeutiger Name des Energieobjekts. Der Name erscheint in der Energieobjekttabelle und ist nach individuellen Anforderungen änderbar. Gültige Namenslänge: 1 - 32 Zeichen Nicht erlaubt sind folgende Zeichen: ? ' * " \ % § ° Weitere Informationen zu Namenskonventionen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".
Kommentar	Optionale Kurzbeschreibung zum Energieobjekt, z. B. der Einsatzzweck oder die Datenquelle des Energieobjekts
Freigabe Energieobjekt	• Bei aktiviertem Optionskästchen ist das Energieobjekt für die Generierung des Energieprogramms freigegeben. • Bei deaktiviertem Optionskästchen wird das Energieobjekt im Energieprogramm nicht verwendet. Wenn die Anzahl der Runtime-Lizenzen für Energy Suite (Seite 69) kleiner ist als die Anzahl der freigegebenen Energieobjekte, wird die Generierung des Energieprogramms abgebrochen und eine Fehlermeldung angezeigt. Nicht freigegebene Energieobjekte werden bei der Lizenz-Zählung nicht berücksichtigt.
Erstellt am	Das Erstelldatum entspricht dem Zeitpunkt, an dem das Energieobjekt in der Energieobjekttabelle hinzugefügt wurde. Das Erstelldatum ist nicht änderbar.
Zuletzt geändert am	Das Änderungsdatum entspricht dem Zeitpunkt, an dem das Energieobjekt zuletzt geändert wurde. Bei jeder Änderung der Eigenschaften des Energieobjekts wird das Änderungsdatum aktualisiert. Das Änderungsdatum ist nicht änderbar.

Siehe auch

Energieobjekt parametrieren (Seite 37)

Freigabe der Energieobjekte für Energieprogramm ändern (Seite 40)

3.4.3 Projektinformation

Die folgende Tabelle zeigt die Einstellungen des Energieobjekts unter "Projektinformation":

Einstellung	Beschreibung
Energieobjektkennung	Die Energieobjektkennung wird automatisch beim Anlegen des Energieobjekts vergeben. Die Energieobjektkennung wird zur Adressierung und Identifizierung des Energieobjekts intern verwendet. Die Energieobjektkennung ist nicht änderbar.
Autor	Name des Bearbeiters des Energieobjekts. Der Name ist nach individuellen Anforderungen änderbar. Standardmäßig ist der im Betriebssystem angemeldete Benutzer eingestellt.

Siehe auch

Energieobjekt parametrieren (Seite 37)

3.4.4 Identification & Maintenance

Die folgende Tabelle zeigt die Einstellungen des Energieobjekts unter "Identification & Maintenance":

Einstellung	Beschreibung
Anlagenkennzeichen	Das Anlagenkennzeichen definiert die Messstelle, die dem Energieobjekt zugewiesen ist, innerhalb der prozesstechnischen Anlage eindeutig.
Ortskennzeichen	Das Ortskennzeichen definiert die genaue Lage der Messstelle, die dem Energieobjekt zugewiesen ist, innerhalb der prozesstechnischen Anlage.
Einbaudatum	Datum, an dem die Messstelle, die dem Energieobjekt zugewiesen ist, in der Anlage installiert wurde. Das Datum ist aus dem Kalenderdialogfeld auswählbar.
Zusatzinformation	Zusätzliche Informationen zur Messstelle, die dem Energieobjekt zugewiesen ist, z. B. Einsatzzweck der Messhardware.

Siehe auch

Energieobjekt parametrieren (Seite 37)

3.4.5 Freigabe der Energieobjekte für Energieprogramm ändern

Sie können die Berücksichtigung von einzelnen Energieobjekten für die Generierung des Energieprogramms ändern. Nach dem Hinzufügen eines neuen Energieobjekts ist die Freigabe aktiviert (Default-Einstellung). Für bestimmte Anwendungen kann es sinnvoll sein, diese Einstellung abzuschalten (z. B. zu Testzwecken oder zur optimalen Ausnutzung von Runtime-Lizenzen).

Hinweis

Nicht freigegebene Energieobjekte werden bei der Runtime-Lizenzierung (Seite 69) nicht berücksichtigt.

Dafür haben Sie folgende Möglichkeiten:

- Im Inspektorfenster: beim Anlegen und Parametrieren des Energieobjekts
- In der Energieobjekttabelle: zentral und übersichtlich für alle zugehörigen Energieobjekte

Vorgehen

Um die Freigabe des Energieobjekts für das Energieprogramms im Inspektorfenster zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in der das Energieobjekt angelegt ist.
   Die Energieobjekttabelle mit den angelegten Energieobjekten wird geöffnet.
3. Klicken Sie auf das gewünschte Energieobjekt in der Energieobjekttabelle.
   Im Inspektorfenster "Eigenschaften" werden alle Parameter des Energieobjekts angezeigt.
4. Aktivieren/Deaktivieren Sie bei den Grundeinstellungen das Optionskästchen "Freigabe Energieobjekt".
   - Aktiviert: Das Energieobjekt ist für die Generierung des Energieprogramms freigegeben.
   - Deaktiviert: Das Energieobjekt wird für die Generierung des Energieprogramms nicht verwendet.

Diese Einstellung des Energieobjekts wird innerhalb der Energieobjekttabelle in der Spalte "Freigabe Energieobjekt" ebenfalls aktualisiert. Alternativ können Sie in der Energieobjekttabelle das Optionskästchen aktivieren/deaktivieren.

3.4.6 Energiedatenquelle

3.4.6.1 Energiedatenquelle zuweisen

Jedes Energieobjekt benötigt zum Erstellen eines Energieprogramms eine Energiedatenquelle, die Energiedaten bereitstellt. Als Energiedatenquellen für die Energieobjekte sind folgende Quellen nutzbar:

- Eingangsmessdaten von angeschlossener Messhardware (Seite 24)
- Energiedaten aus dem Prozessabbild oder Merker-Bereich der CPU (PLC-Variablen) (Seite 27)
- Energiedaten aus Datenbausteinen (DB-Variablen) (Seite 29)

Für das Zuweisen der Energiedatenquelle haben Sie folgende Möglichkeiten:

- Im Inspektorfenster: beim Anlegen und Parametrieren des Energieobjekts
- In der Energieobjekttabelle: zentral und übersichtlich für alle zugehörigen Energieobjekte

Voraussetzung

- Die PLC-Variablen und DB-Variablen sind für die Energiedatenerfassung deklariert.
- Die Messhardware für die Energiedatenerfassung ist unterhalb der S7-1500 CPU konfiguriert.
- Die Energy Support Library mit den Treiberbausteinen zur Messhardware ist installiert.

Im Inspektorfenster

Um einem Energieobjekt eine Energiedatenquelle zuzuweisen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in welcher das Energieobjekt angelegt ist. Die Energieobjekttabelle mit den angelegten Energieobjekten wird geöffnet.
3. Klicken Sie auf das gewünschte Energieobjekt in der Energieobjekttabelle.
   Im Inspektorfenster "Eigenschaften" werden die Eigenschaften des Energieobjekts angezeigt.
4. Wählen Sie in der Bereichsnavigation den Eintrag "Konfiguration > Energiedatenquelle".
5. Klicken Sie rechts auf das Feld "Energiedatenquelle".
   Der Auswahldialog mit den CPU-weit deklarierten PLC-/DB-Variablen bzw. mit den Modulen der verfügbaren Messhardware wird geöffnet.
6. Navigieren Sie im Auswahldialog zur gewünschten Energiedatenquelle.
7. Übernehmen Sie die Auswahl durch Klicken auf das grüne Häkchen im Auswahldialog.

Abhängig von der zugewiesenen Energiedatenquelle werden im Bereich "Energiedatentyp" die auswählbaren Energiedatentypen aktualisiert. Im Inspektorfenster können Sie die weiteren Einstellungen zum Energiedatentyp (Seite 43) vornehmen.

Die zugewiesene Energiedatenquelle wird innerhalb der Energieobjekttabelle in der Spalte "Energiedatenquelle" ebenfalls aktualisiert.

In Energieobjekttabelle

Um ggfs. für mehrere Energieobjekte die Energiedatenquelle zuzuweisen oder zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in welcher die Energieobjekte angelegt sind.
   Die Energieobjekttabelle mit den angelegten Energieobjekten wird geöffnet.
3. Klicken Sie in der Spalte "Energiedatenquelle" auf das Tabellenfeld in der Zeile des Energieobjekts.
   Der Auswahldialog mit den CPU-weit deklarierten PLC-/DB-Variablen bzw. mit den Modulen der verfügbaren Messhardware wird geöffnet.
4. Navigieren Sie im Auswahldialog zur gewünschten Energiedatenquelle.
5. Übernehmen Sie die Auswahl durch Klicken auf das grüne Häkchen im Auswahldialog.

Die zugewiesene Energiedatenquelle wird im Inspektorfenster "Eigenschaften" ebenfalls aktualisiert. Im Inspektorfenster können Sie die weiteren Einstellungen zum Energiedatentyp (Seite 43) vornehmen.

3.4.6.2 Energiedatenquelle

Die folgende Tabelle zeigt die Einstellungen des Energieobjekts unter "Energiedatenquelle":

Einstellung	Beschreibung
Energiedatenquelle	Als Energiedatenquelle sind auswählbar: • Messhardware-Modul mit den Eingangsmessdaten der Messstell oder • PLC-Variable mit den Werten aus dem Prozessabbild oder DB Die Energiedatenquellen sind vor der Parameterierung der Energieobjekte anzulegen und zu projektieren. Danach stehen sie im Auswahldialog für die Energiedatenquelle zur Verfügung.
Typ	Je nach ausgewählter Energiedatenquelle für das Energieobjekt, wird der entsprechende Typ der Energiedatenquelle angezeigt. Der Typ der Energiedatenquelle ist nicht änderbar.

Einstellung	Beschreibung
Fehlerüberwachung der Energiedatenquelle hinzufügen	Fehlerüberwachung für Energiedatenquellen, die keine integrierte Fehlerüberwachung besitzen (z. B. bei analogen Energiewerten). Hierfür programmieren Sie die Auswertung des Hardware-Fehlers im PLC-Programm. Wenn eine Störung am Eingangssignal der Energiedatenquelle auftritt, dann wird der Trigger für Hardware-Fehler gesetzt. Konventionen für das Signal: • Mindestdauer von 2 * Wert an der Einstellung "Erfassungszyklus" • Steigende Flanke: kommender Fehler • High-Pegel: Fehler liegt an • Fallende Flanke: gehender Fehler • Low-Pegel: kein Fehler Es werden nur Fehlerereignisse erfasst, die mindestens für die Dauer des Erfassungszyklus anstehen. Weitere informationen zum Erfassungszyklus finden Sie im Abschnitt Energiedatentyp (Seite 43). Bei aktiviertem Optionskästchen ist die Variable für den Trigger für Hardware-Fehler auswählbar.
Trigger für Hardware-Fehler	Auswahl der Variable für den Trigger für Hardware-Fehler Wenn ein ungültiger Energiewert aus der Energiedatenquelle erfasst wird, die keine integrierte Fehlerüberwachung besitzt, dann wird die ausgewählte Variable vom Datentyp "BOOL" gesetzt und die Fehlermeldung "Hardware-Fehler" ausgegeben. Die Auswertung der Variable für den Trigger für Hardware-Fehler muss entsprechend der Konventionen für das Signal programmiert werden, um bei der fallenden Flanke einen kommenden Fehler zu diagnostizieren. Die Konventionen für das Signal finden Sie in der Beschreibung der Einstellung "Fehlerüberwachung der Energiedatenquelle hinzufügen". Beispiel: • Ein Messgerät, das von der Energy Support Library nicht unterstützt wird, liefert einen analogen Leistungswert. • Zusätzlich stellt das Messgerät einen binären Ausgang zur Verfügung, der bei Störung eine fallende Flanke ausgibt. Nur bei aktiviertem Optionskästchen "Fehlerüberwachung der Energiedatenquelle hinzufügen" auswählbar.

Siehe auch

Energiedatenquelle zuweisen (Seite 41)

3.4.6.3 Energiedatentyp

Abhängig von der Energiedatenquelle des Energieobjekts werden folgende Energiedatentypen verarbeitet.

Energiedatentyp

Die folgende Tabelle zeigt die verfügbaren Energiedatentypen mit entsprechenden Einheiten und Beispielen:

Einstellung	Einheit	Beispiel
Impulssignal	Wh l hl m³ m³ [n] kg t pcs	Bei einem Normierungsfaktor von 0,5 entsprechen sechs Pulse 3 Wh.
Energiezählwert	Wh l hl m³ m³ [n] kg t pcs	Der Zählerstand eines Energiezählers verändert sich in zehn Sekunden von 6.734.392 Wh auf 6.734.414 Wh. Die Differenz entspricht einem Energieverbrauch von 22 Wh und einer durchschnittlichen Leistungsaufnahme von 7,92 kW. Das Zurücksetzen des Energiezählwerts während einer laufenden Archivierungsperiode führt zur Berechnung von fehlerhaften Energiewerten für diese Archivierungsperiode. Für einen gültigen Energiezählwert gilt: • Energiezählwert < Überlaufwert
Leistungswert	W l/h hl/h m³/h m³/ h [n]kg/h t/h pcs/h	Eine durchschnittliche Leistungsaufnahme von 180 kW über zehn Sekunden entspricht einem Energieverbrauch von 0,5 kWh bzw. 500 Wh.

Erfassungszyklus

Pro Erfassungszyklus werden die Treiberbausteine für die Erfassung der Energiedaten einmal aufgerufen.

Folgende Erfassungszyklen sind auswählbar:

- 0,01 s
- 0,02 s
- 0,05 s
- 0,1 s
- 0,2 s
- 0,3 s
- 0,5 s

- 1 s
- 2 s

Standardmäßig sind folgende Werte eingestellt:

- Impulssignal: 0,01 s
- Energiezählwert: 1 s
- Leistungswert: 0,1 s

Einstellungen für ausgewählten Energiedatentyp

Die folgende Tabelle zeigt die weiteren Einstellungen für den ausgewählten Energiedatentyp:

Einstellung	Beschreibung
Datentyp	Nur bei Energiedatentyp "Energiezählwert" sichtbar Der Datentyp des Energiezählwerts wird abhängig von der ausgewählten Energiedatenquelle angezeigt, die den Energiezählwert liefert. Der Datentyp ist nicht änderbar.
Einheit	Physikalische Einheit des Energiewerts • Einheiten für Impulssignal: Wh, l, hl, m³, m³ [n], kg, t, pcs • Einheiten für Energiezählwert: Wh, l, hl, m³, m³ [n], kg, t, pcs • Einheiten für Leistungswert: W, l/h, hl/h, m³/h, m³/h [n], kg/h, t/h, pcs/h
Normierungsfaktor	Gibt an, mit welchem Wert der erfasste Energiewert multipliziert wird, um diesen Energiewert in ausgewählte Einheit umzurechnen (z. B. "W" bzw. "Wh" oder "m³" bzw. "m³/h"). Standardmäßig ist 1,0 voreingestellt. Zulässiger Wertebereich: 0 – 9999999 Beispiel: Ein Messgerät liefert einen Leistungswert von 3,7 mit der Einheit "kWh". Einstellungen hierfür in Energy Suite: • Einheit: Wh • Normierungsfaktor: 1000 Energy Suite verarbeitet den Wert zu 3700 Wh.
Energieflussrichtung	Parametrierung der Energieflussrichtung • Verbraucher Die Energie wird aus dem Stromnetz zum Verbraucher (z. B. elektrische Maschine im Motorbetrieb) bezogen. • Erzeuger Die Energie wird vom Erzeuger (z. B. elektrische Maschine im Generatorbetrieb) in das Stromnetz eingespeist.

Einstellung	Beschreibung
Überlaufwert	Nur für den Energiedatentyp "Energiezählwert" auswählbar Gibt an, bei welchem Wert der Energiezählwert auf 0 zurückgesetzt wird. Standardmäßig ist 1000000 (1*10^6) voreingestellt. Einstellbare Werte: • 1000000 (1*10^6) • 1000000000 (1*10^9) • 1000000000000 (1*10^12) • Benutzerdefiniert
Benutzerdefinierter Überlaufwert	Nur bei Überlaufwert = "Benutzerdefiniert" parametrierbar Manuelle Eingabe des Überlaufwerts, bei dem der Energiezählwert auf 0 zurückgesetzt wird. Zulässiger Wertebereich: 1 - 1.0e+15

Siehe auch

Energiedatenquelle zuweisen (Seite 41)

3.4.7 Periodische Energiewerte

3.4.7.1 Energiewerte für periodische Archivierung festlegen

Die erfassten Energiedaten werden nach der Verarbeitung als periodische Energiewerte zusammen mit ihren Zeitstempel archiviert. Je nach ausgewähltem Energiedatentyp sind unterschiedliche Energiewerte für die Archivierung einstellbar.

Beim Parametrieren des Energieobjekts wählen Sie hierfür die periodischen Energiewerte und die Archivierungsperiode im Inspektorfenster aus.

Voraussetzung

- Die Achivierungsperioden mit verschiedenen Zeitwerten sind an zentraler Stelle bei den Einstellungen für das Energieprogramm angelegt.
  Weitere Informationen hierzu finden Sie im Abschnitt "Archivierungsperioden definieren (Seite 51)".

Vorgehen

Um für ein Energieobjekt die periodischen Energiewerte für die Archivierung festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf die Energieobjekttabelle, in der das Energieobjekt angelegt ist.
   Die Energieobjekttabelle mit den angelegten Energieobjekten wird geöffnet.

3. Klicken Sie auf das gewünschte Energieobjekt in der Energieobjekttabelle.
   Im Inspektorfenster "Eigenschaften" werden die Eigenschaften des Energieobjekts angezeigt.
4. Wählen Sie in der Bereichsnavigation den Eintrag "Konfiguration > Periodische Energiedaten".
5. Aktivieren Sie die Optionskästchen zu den gewünschten Energiewerten (Seite 47):
   – Leistungsaufnahme
   – Energieverbrauch
   – Energiezählwert (nur bei Energiedatentyp Energiezählwert aktivierbar)
6. Klicken Sie rechts auf das Feld "Archivierungsperiode".
   Die Tabelle mit den für das Energieprogramm zentral angelegten Archivierungsperioden (Seite 51) wird geöffnet. Wenn zuvor keine eigenen Archivierungsperioden angelegt worden, stehen hier defaultmäßig nur zwei Archivierungsperioden (15 min, 60 min) zur Auswahl.
7. Klicken Sie auf die gewünschte Archivierungsperiode und übernehmen Sie die Auswahl durch Klicken auf das grüne Häkchen im Auswahldialog.

3.4.7.2 Periodische Energiedaten

Die erfassten Energiedaten werden nach der Verarbeitung als aufbereitete periodische Energiewerte zusammen mit ihren Zeitstempel archiviert. Je nach ausgewähltem Energiedatentyp stehen unterschiedliche Energiewerte für die Archivierung zur Verfügung.

Periodische Energiewerte

Die folgende Tabelle zeigt die periodischen Energiewerte für die Archivierung:

Einstellung	Beschreibung
Leistungsaufnahme	Durchschnittliche Leistungsaufnahme pro Archivierungsperiode. Archivierung bei folgenden ausgewählten Energiedatentypen möglich: • Impulssignal • Energiezählwert • Leistungswert
Energieverbrauch	Energieverbrauch pro Archivierungsperiode. Archivierung bei folgenden ausgewählten Energiedatentypen möglich: • Impulssignal • Energiezählwert • Leistungswert
Energiezählwert	Energiezählwert pro Archivierungsperiode. Archivierung bei folgenden ausgewählten Energiedatentypen möglich: • Energiezählwert

Zuweisung der Archivierungsperiode

Durch die Auswahl der Archivierungsperiode wird dem Energieobjekt und damit auch dem ausgewählten Energiewert der periodische Zeitabschnitt für die Berechnung und die Archivierung zugewiesen. Neue Archivierungsperioden legen Sie bei den zentralen Einstellungen für das Energieprogramm (Seite 51) fest. Jedem Energieobjekt weisen Sie genau eine Archivierungsperiode zu.

Hinweis

Mehrfachverwendung der gleichen Messhardware als Energiedatenquelle vermeiden

Achten Sie darauf, dass die gleiche Messhardware nur einmal als Energiedatenquelle im Energieprogramm verwendet wird, um die Kommunikationskanäle nicht zu überlasten.

Beispiel für Zuweisung mehrerer Archivierungsperioden

Für die Archivierung z. B. des Energieverbrauchs mit einer 15 min- und einer 60 min-Archivierungsperiode, legen Sie für jede Archivierungsperiode jeweils ein Energieobjekt an.

1. Für das Energieobjekt mit der 15 min-Archivierungsperiode wählen Sie die Messhardware als Energiedatenquelle und aktivieren die Archivierung des Energieverbrauchs. Anschließend generieren Sie das Energieprogramm.
2. Für das Energieobjekt mit der 60 min-Archivierungsperiode wählen Sie den Ausgangsparameter "energy" des Treiberbausteins der Messhardware als Energiedatenquelle. Anschließend aktivieren Sie die Archivierung des Energieverbrauchs für das Energieobjekt und generieren das Energieprogramm erneut.

So wird der Energieverbrauch einer Messstelle in zwei Archivierungsperioden archiviert und die Messhardware nur einmal als Energiedatenquelle im Energieprogramm verwendet.

Siehe auch

Energiewerte für periodische Archivierung festlegen (Seite 46)

3.4.8 Verarbeitung

3.4.8.1 Allgemeine Verarbeitungseinstellungen

Die folgende Tabelle zeigt die Einstellungen des Energieobjekts unter "Verarbeitung":

Einstellung	Beschreibung
Allgemeine Verarbeitungseinstellungen	
Interner Verarbeitungszyklus	Pro internen Verarbeitungszyklus werden die für die Archivierung aktivierten Energiedaten einmal berechnet und aktualisiert. Folgende Zeitdauer für den internen Verarbeitungszyklus ist auswählbar: • 0,1 s • 0,2 s • 0,5 s • 1 s Der interne Verarbeitungszyklus kann für eine schnellere Aktualisierung der Energiedaten verändert werden, z. B. für: • Visualisierung der Energiedaten • Weitere Verarbeitung der Energiedaten durch das Anwenderprogramm (z. B. Lastmanagement) Standardmäßig ist 1 s eingestellt.
Berechnung des Momentanwerts der Leistungsaufnahme	
Berechnungsintervall	Dient zur Berechnung des Momentanwerts der Leistungsaufnahme Der Momentanwert der Leistungsaufnahme wird aus dem Energieverbrauch pro Berechnungsintervall berechnet. Der berechnete Momentanwert entspricht dem Durchschnittswert über die Dauer des Berechnungsintervalls. Nur editierbar, wenn Energiedatentyp "Energiezählwert" oder "Impulssignal" ausgewählt ist. Folgende Zeitdauer für das Berechnungsintervall ist auswählbar: • 1 s • 2 s • 3 s • 4 s • 5 s • 6 s • 10 s • 15 s • 20 s • 30 s • 60 s Standardmäßig sind 10 s eingestellt.

Einstellung	Beschreibung
Berechnungsintervall verlängern	Die dynamische Verlängerung des Berechnungsintervalls ermöglicht die Berechnung des Momentanwerts der Leistungsaufnahme mit zusätzlicher Erfassungszeit. Beispiel: Bei konstanter Leistungsaufnahme wird alle 4 s ein Impuls ausgegeben. Beim standardmäßig eingestellten Berechnungsintervall von 10 s werden • im ersten Berechnungsintervall 2 Impulse gezählt, statt 2 1/2 • im zweiten Berechnungsintervall 3 Impulse gezählt Bei einer Verlängerung des Berechnungsintervalls von 5 s werden • im ersten Berechnungsintervall 3 Impulse gezählt • im zweiten Berechnungsintervall 3 Impulse gezählt Das zweite Berechnungsintervall beginnt nach 12 s ab Beginn des ersten Berechnungsintervalls.
Verlängerung des Berechnungsintervalls	Die Berechnung des Momentanwerts der Leistungsaufnahme startet, nachdem eine der folgenden Voraussetzungen erfüllt ist: • Der Energiezählwert hat sich geändert. • Mindestens die Dauer der Verlängerung des Berechnungsintervalls ist abgelaufen. • Der Verbraucher wurde ausgeschaltet. Folgende Zeitdauer für die Verlängerung des Berechnungsintervalls ist auswählbar: • 1 s • 2 s • 3 s • 4 s • 5 s • 6 s • 10 s • 15 s • 20 s • 30 s • 60 s

Energiedaten puffern und archivieren 4

4.1 Archivierungsperioden definieren

Archivierungsperioden

Archivierungsperioden sind periodische Zeitabschnitte, in denen die Energierohdaten des jeweiligen Energieobjekts in der CPU zu einem periodischen Energiewert mit Zeitstempel aufbereitet und anschließend - je nach Einstellung - in der CPU (SIMATIC Memory Card) oder in WinCC Professional (Variablenarchiv) archiviert werden (z. B. alle 15 Minuten).

Damit anlagenweite Energiewerte aus allen Energieobjekten zur selben Zeit archiviert werden, kann die konfigurierte Zeitdauer der Archivierungsperiode mit einem internen oder externen Zeitgeber synchronisiert werden:

- Standardmäßig wird die interne CPU-Uhrzeit als Zeitgeber für alle Bausteine des Energieprogramms verwendet. Hierfür synchronisieren Sie die CPU-Uhrzeit mit einem externen Zeitserver (z. B. mit NTP-Verfahren).
  Weitere Informationen hierzu finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter dem Stichwort "Uhrzeitsynchronisation (S7-1500)".
- Um eine Synchronisation mit einem externen Zeitgeber (z. B. Synchronisationssignal eines Versorgungsunternehmens) vorzunehmen, aktivieren Sie den externen Impuls und konfigurieren dessen Impulsquelle.

Die verschiedenen Archivierungsperioden legen Sie an zentraler Stelle in den Einstellungen für das Energieprogramm an. Bei der Konfiguration der Energieobjekte weisen Sie anschließend jedem einzelnen Energieobjekt die jeweilige Archivierungsperiode zu.

Weitere Informationen zu den Namenskonventionen von Archivierungsperioden finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".

UTC-Zeit

Hinweis

Stellen Sie die Uhrzeit der CPU auf die UTC-Zeit ein, um z. B. die Zeitverschiebung beim Wechsel der Sommer- und Winterzeit zu vermeiden.

Die Einstellung der Uhrzeit der CPU entspricht der standardmäßigen Vorgehensweise und erfolgt daher außerhalb von Energy Suite - in der Online- und Diagnosesicht der CPU.

Weitere Informationen hierzu finden Sie in der Online-Hilfe des TIA Portal Informationssystems im Abschnitt "Die Uhrzeit einer CPU ermitteln und einstellen".

Vorgehen

Um neue Archivierungsperioden im Dialog der zentralen Verwaltung anzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf das Objekt "Energieprogrammeinstellungen".
   Der Dialog mit den Einstellungen für das Energieprogramms wird im Arbeitsbereich geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Archivierungsperioden".
   Die Tabelle mit der zentralen Verwaltung der Archivierungsperioden wird geöffnet. Jede Zeile der Tabelle repräsentiert eine Archivierungsperiode. Wenn zuvor noch keine eigenen Archivierungsperioden angelegt worden sind, stehen in der Tabelle defaultmäßig zwei Archivierungsperioden (15 min, 60 min) zur Verfügung.
4. Nur für 24h-Archivierungsperioden: Stellen Sie über das Optionskästchen "Zeitzonen-Einstellungen von regionalen Einstellungen des lokalen Engineering-PCs anwenden" die Zeitzone (Differenz zur UTC-Zeit) für die Synchronisation ein:
   - Aktiviert: Die Zeitzone des lokalen Engineering-PCs wird verwendet.
   - Deaktiviert: Die Zeitzone aus der nachfolgenden Klappliste wird verwendet. Wählen Sie hierzu die Zeitzone aus der Klappliste aus.
5. Um eine weitere Archivierungsperiode anzulegen, doppelklicken Sie in der Tabelle in der Spalte "Name" auf "<Hinzufügen>".
   Eine neue Archivierungsperiode wird in der Tabelle angelegt.
6. Legen Sie in gleicher Weise ggf. weitere Archivierungsperioden an.
7. Legen Sie die Einstellungen für jede Archivierungsperiode fest:
   - Name des Archivierungsperiode
   - Zeitdauer der Archivierungsperiode
   - Einheit zu Zeitdauer der Archivierungsperiode
   - Synchronisation mit externem Impuls (Zeitgeber) und Auswahl der Impulsquelle

Die Bedeutung der Einstellungen ist im Abschnitt "Archivierungsperioden (Seite 53)" detailliert beschrieben.

4.2 Archivierungsperioden

Die verschiedenen Archivierungsperioden legen Sie an zentraler Stelle in den Einstellungen für das Energieprogramm an. Bei der Konfiguration der Energieobjekte weisen Sie anschließend jedem einzelnen Energieobjekt die jeweilige Archivierungsperiode zu.

Standardmäßig sind zwei Archivierungsperioden (15 min, 60 min) mit internem Zeitgeber für die Synchronisation voreingestellt.

Zeitzone

Die folgende Tabelle zeigt die Zeitzonen-Einstellungen für die Synchronisation unter "Archivierungsperioden". Die Zeitzonen-Einstellungen gelten nur für die Synchronisation der Archivierungsperioden mit einer Zeitdauer von 24 h.

Einstellung	Beschreibung
Zeitzonen-Einstellungen von regionalen Einstellungen des lokalen Engineering-PCs anwenden	• Aktiviert: Die 24 h-Archivierungsperioden werden anhand der Zeitzonen-Einstellungen von den regionalen Einstellungen des lokalen Engineering-PCs synchronisiert. • Deaktiviert: Auswahl der gewünschten Zeitzone für die Synchronisation der 24 h-Archivierungsperioden aus der Klappliste. Standardmäßig deaktiviert.
Zeitzone	Zeitzone (Differenz zur UTC-Zeit) Archivierungsperioden mit einer Zeitdauer von 24 h werden um 0:00:00:000 UTC-Zeit synchronisiert. Bei einer Verschiebung zwischen UTC-Uhrzeit und der regionalen Uhrzeit kann die Synchronisation der 24 h-Archivierungsperiode anhand der ausgewählten Zeitzone angepasst werden. Die Zeitzone ist nur bei deaktivierten Optionskästchen "Zeitzonen-Einstellungen von regionalen Einstellungen des lokalen Engineering-PCs anwenden" auswählbar.

Archivierungsperioden

Die folgende Tabelle zeigt die Einstellungen für neue Archivierungsperioden in der zentralen Verwaltung:

Einstellung	Beschreibung
Name	Standardmäßig sind zwei Archivierungsperioden (15 min, 60 min) mit internem Zeitgeber für die Synchronisation voreingestellt. Werden andere Werte für die Zeitdauer benötigt, dann legen Sie weitere Archivierungsperioden an. Der Name der Archivierungsperiode ist CPU-weit eindeutig und ist nach individuellen Anforderungen änderbar. Empfehlung: Verwenden Sie im Namen die Zeitdauer der Archivierungsperiode (z. B. UserPeriod_15min) Weitere Informationen zu Namenskonventionen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".
Zeitdauer	Zeitdauer, an deren Ende die Energiewerte periodisch berechnet und archiviert werden. Anhand der Zeitdauer werden die Zeitstempel erzeugt. Beispiel: Bei einer 15 min-Archivierungsperiode werden Zeitstempel im 15 min-Raster erzeugt (beginnend zur vollen Stunde). Die aufbereiteten periodischen Energiewerte werden zusammen mit dem jeweiligen Zeitstempel archiviert: • als CSV-Datei CPU-intern auf der SIMATIC Memory Card oder • im Variablenarchiv des SCADA-Systems WinCC Professional Bei neu hinzugefügten Archivierungsperioden sind standardmäßig 15 min voreingestellt.
Einheit	Einheit für die Zeitdauer, in der ein periodischer Energiewert berechnet und archiviert wird. Folgende Einheiten sind einstellbar: • min = Minuten • h = Stunden
Externer Impuls	Ein externer Impuls triggert den Taktgeber für die Synchronisation der Archivierungsperiode. • Bei aktiviertem Optionskästchen weisen Sie in der Spalte "Impulsquelle" dem externen Impuls das Synchronisationssignal zu (z. B. von einem Energieversorgungsunternehmen). • Bei deaktiviertem Optionskästchen triggert die interne CPU-Uhrzeit die Synchronisation der Archivierungsperiode. Die interne CPU-Uhrzeit wird für die Bildung des Zeitstempels mit einem externen Zeitserver (z. B. mit NTP-Verfahren) auf UTC-Zeit synchronisiert. Standardmäßig deaktiviert.
Impulsquelle	Als Impulsquelle ist eine Variable vom Datentyp "BOOL" auswählbar. Die Variable muss dem Synchronisationssignal, z. B. des Energieversorgungsunternehmens, zugewiesen sein. Nur editierbar, wenn der externe Impuls aktiviert ist.

Siehe auch

Archivierungsperioden definieren (Seite 51)

4.3 Archivierung auf S7-CPU (SIMATIC Memory Card) festlegen

Die aufbereiteten periodischen Energiewerte mit Zeitstempel werden in Datenbausteinen zur Pufferung zwischengespeichert, bevor Sie auf der S7-1500 CPU (auf gesteckter SIMATIC Memory Card) archiviert werden.

Die Anzahl der Datenbausteine für die Pufferung entspricht der Anzahl der Energieobjekttabellen.

Anschließend werden die Energiewerte in Data Logs (CSV-Dateien) auf der S7-1500 CPU (SIMATIC Memory Card) gespeichert. Pro Archivierungsperiode wird ein Datensatz in das Data Log geschrieben. Ein Datensatz beinhaltet einen Energiewert mit Zeitstempel und Status.

Allgemeine Informationen zu Data Logs finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Data Logging - Übersicht".

Für die Archivierung auf der Memory Card legen Sie die Anzahl und die Größe der Data Log-Dateien fest.

Voraussetzung

- Die aufbereiteten periodischen Energiewerte mit Zeitstempel (Leistungsaufnahme, Energieverbrauch, Energiezählwert) sind für die Archivierung aktiviert.
  Weitere Informationen hierzu finden Sie im Abschnitt "Energiewerte für periodische Archivierung festlegen (Seite 46)".

Vorgehen

Um die notwendigen Einstellungen für die Archivierung auf der S7-1500 CPU festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf das Objekt "Energieprogrammeinstellungen".
   Der Dialog mit den Einstellungen für das Energieprogramm wird im Arbeitsbereich geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Archivierung und Pufferung".
4. Aktivieren Sie im Bereich "Archivierung" das Optionsfeld "PLC (SIMATIC Memory Card)".
   - Unterhalb des Optionsfeldes werden die Parameter zum Data Log eingeblendet.
   - Im Bereich "Pufferung" werden die Anzahl und die Größe der Datenbausteine für die Pufferung angezeigt.
5. Legen Sie im Bereich "Archivierung" die Einstellungen zu den Data Logs (Datenarchive im CSV-Fomat) zum Übertragen auf die Memory Card fest:
   - Name des Data Logs
   - Anzahl der Data Logs
   - Größe des Data Logs
6. Prüfen Sie im Bereich "Pufferung" die angezeigten Parameter für die Größe und den Speicherbereich der Datenbausteine.

Die Bedeutung der Einstellungen ist im Abschnitt "Archivierung und Pufferung (S7-CPU) (Seite 56)" detailliert beschrieben.

4.4 Archivierung und Pufferung (S7-CPU)

Archivierung

Die aufbereiteten periodischen Energiewerte mit Zeitstempel werden in Data Logs (CSV-Dateien) auf der S7-1500 CPU (SIMATIC Memory Card) archiviert. Allgemeine Informationen zu Data Logs und zur Struktur der CSV-Dateien finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Data Logging - Übersicht".

Für die Archivierung auf der SIMATIC Memory Card ist die Anzahl und die Größe der Data Log-Dateien relevant.

Die folgende Tabelle zeigt die Einstellungen im Bereich "Archivierung" für den Archivierungstyp "PLC (SIMATIC Memory Card)":

Einstellung	Beschreibung
Name des Data Logs	Der Name des Data Logs wird für Dateinamen der CSV-Datei (Comma separated values) verwendet. Der Dateiname besteht aus dem Namen (als Präfix) und dem Zeitstempel des ersten Energiewerts im Data Log. Bei S7-1500 CPUs bestehen für den Namen des Data Logs folgende Einschränkungen: • Zulässige Namenslänge: 1 - 29 Zeichen • Erlaubt sind folgende Zeichen: 0 ... 9, a ... z, A ... Z, "_", "-" Weitere Informationen zu Namenskonventionen finden Sie im Abschnitt "Leistungsmerkmale und Konventionen (Seite 18)".
Anzahl der Data Logs	Die Anzahl der Data Logs wird durch die Größe der SIMATIC Memory Card begrenzt. Wenn die eingestellte Anzahl von Data Logs erreicht ist, wird das älteste Data Log gelöscht und ein neues Data Log angelegt. Gültige Anzahl der Data Logs: 2 - 10 Standardmäßig ist 5 eingestellt.
Größe des Data Logs	Wenn ein Data Log die eingegebene Größe erreicht, wird ein neues Data Log angelegt. Die maximale Größe des Data Logs ist vom System auf 1024 MB begrenzt. Für ein Data Log wird der erforderliche Speicherplatz auf der SIMATIC Memory Card über folgende Formel berechnet: 45 Byte + (Anzahl der Datensätze * 190 Byte) Ein Datensatz beinhaltet einen Energiewert mit Zeitstempel und Status. Empfehlung: Stellen Sie den Wert für die Einstellung "Größe des Data Logs" nicht größer als 180 MB ein, um einen Datenverlust bei der Weiterverwendung der CSV-Datei z. B. mit "Microsoft Excel" zu vermeiden. Standardmäßig sind 10 MB eingestellt.

Pufferung

Die periodischen Energiewerte werden zusammen mit ihrem Zeitstempeln in den Datenbausteinen zur Pufferung zwischengespeichert, bevor Sie auf der S7-1500 CPU (SIMATIC Memory Card) archiviert werden. Die Zwischenspeicherung vermeidet Datenverluste - z. B. für den Fall, dass die SIMATIC Memory Card voll ist, weil die parametrierte Anzahl der Data Logs erreicht ist.

Die folgende Tabelle zeigt die Einstellungen im Bereich "Pufferung" für den Archivierungstyp "PLC (SIMATIC Memory Card)":

Einstellung	Beschreibung
Anzahl der Datenbausteine	Die Anzahl der Datenbausteine entspricht der Anzahl der angelegten Energieobjekttabellen. Für jede Energieobjekttabelle wird ein Datenbaustein zur Pufferung benötigt. Die Datenbausteine werden bei der Generierung des Energieprogramms automatisch angelegt.
Größe der Datenbausteine	Durch größere Datenbausteine haben Sie die Möglichkeit den Pufferzeitraum der Energiewerte auf der CPU zu verlängern. Standardmäßig sind 10 KB eingestellt.
Belegter Speicher	Produkt aus Anzahl multipliziert mit Größe der Datenbausteine

Siehe auch

Beispiel für Archivierung (S7-CPU) (Seite 58)

Archivierung auf S7-CPU (SIMATIC Memory Card) festlegen (Seite 55)

4.5 Beispiel für Archivierung (S7-CPU)

Der erforderliche Speicherplatz auf der SIMATIC Memory Card für ein Data Log wird über folgende Formel berechnet:

45 Byte + (Anzahl der Datensätze * 190 Byte)

Die maximale Größe eines Data Logs ist systembedingt auf 1 GB begrenzt. Somit können 5.500.000 Datensätze eines Energieobjekts, bei dem die Archivierung der Leistungsaufnahme, Energieverbrauchs und Energiezählwerts aktiviert ist, in einem Data Log archiviert werden.

Das entspricht einer Archivierung über 19097 Tage (52 Jahre) für ein Energieobjekt oder 95 Tage für 200 Energieobjekte mit derselben Parametrierung.

4.6 Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen

Die aufbereiteten periodischen Energiewerte mit Zeitstempel werden in Datenbausteinen zur Pufferung zwischengespeichert, bevor sie auf dem SCADA-System WinCC Professional archiviert werden. Anschließend werden die Energiewerte in das Variablenarchiv auf dem SCADA-System WinCC Professional übertragen. Diese Art der gepufferten Kommunikation ermöglicht eine gesicherte Archivierung und damit eine deutliche Erhöhung der Qualität und Konsistenz der abrechnungsrelevanten Energiedaten.

Die Anzahl der Datenbausteine für die Pufferung entspricht der Anzahl der Energieobjekttabellen.

Hinweis

WinCC Professional (Variablenarchiv)

In Energy Suite aktivieren Sie über ein entsprechendes Optionskästchen, dass die Archivierung der gepufferten Energiewerte in Variablenarchiven von WinCC Professional erfolgen soll.

Die Einrichtung der Variablenarchive von WinCC Professional erfolgt in den Einstellungen von WinCC Professional.

Voraussetzung

- Die gewünschten periodischen Energiewerte (Leistungsaufnahme, Energieverbrauch und/oder Energiezählwert) sind für die Archivierung aktiviert.
  Weitere Informationen hierzu finden Sie im Abschnitt "Energiewerte für periodische Archivierung festlegen (Seite 46)".

Vorgehen

Um die notwendigen Einstellungen für die Archivierung auf der PC-Station im Variablenarchiv von WinCC Professional festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf das Objekt "Energieprogrammeinstellungen".
   Der Dialog mit den Einstellungen für das Energieprogramms wird im Arbeitsbereich geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Archivierung und Pufferung".

4. Aktivieren Sie im Bereich "Archivierung" das Optionsfeld "WinCC Professional (Variablenarchiv)".
   - Unterhalb des Optionsfeldes erscheint ein Hinweis, dass für diesen Archivierungstyp die Einstellung des Variablenarchivs in WinCC Professional zu erfolgen hat. Weitere Informationen hierzu finden Sie im Abschnitt "Archivierungstruktur für Energiewerte erzeugen (Seite 99)".
   - Über das Optionskästchen "Erweiterte Kommunikationseinstellungen (zwischen PLC und WinCC)" können Sie optional weitere Einstellungen für die Überprüfung der Datenübertragung und der Verbindung zwischen PLC und WinCC Professional festlegen.
   - Im Bereich "Pufferung" werden die Anzahl und die Größe der Datenbausteine für die Pufferung angezeigt.
     Das Ausgabefeld "Anzahl der Datenbausteine" zeigt die Anzahl der Datenbausteine an. Diese Anzahl entspricht der Anzahl der angelegten Energieobjektabellen und ist nicht änderbar.
5. Geben Sie im Bereich "Pufferung" die Größe der Datenbausteine für die Pufferung an. Im Ausgabefeld "Belegter Speicher" wird das Produkt aus der parametrierten Anzahl multipliziert mit der Größe der Datenbausteine angezeigt.

Die Bedeutung der Einstellungen ist im Abschnitt "Archivierung und Pufferung (WinCC Professional) (Seite 61)" detailliert beschrieben.

4.7 Archivierung und Pufferung (WinCC Professional)

Archivierung

Die aufbereiteten periodischen Energiewerte mit Zeitstempel werden in Variablenarchiven auf dem SCADA-System WinCC Professional archiviert.

Hinweis

WinCC Professional (Variablenarchiv)

Die Konfiguration der Variablenarchive von WinCC Professional erfolgt in den Einstellungen von WinCC Professional.

Weitere Informationen hierzu finden Sie im Abschnitt "Archivierungstruktur für Energiewerte erzeugen (Seite 99)".

Die folgende Tabelle zeigt die erweiterten Kommunikationseinstellungen (zwischen PLC und WinCC) bei dem ausgewählten Archivierungstyp "WinCC Professional (Variablenarchiv)".

⚠ VORSICHT

Erweiterte Kommunikationseinstellungen

Die standardmäßig eingestellten erweiterten Kommunikationseinstellungen brauchen Sie nicht zu ändern.

Wenden Sie sich bei eventuellen Rückfragen zur Archivierung an Service & Support.

Einstellung	Beschreibung
Erweiterte Kommunikations-einstellungen (zwischen PLC und WinCC)	Nur aktivierbar beim Archivierungstyp "WinCC Professional (Variablen-archiv)" Bei aktiviertem Optionskästchen sind die Einstellungen unter folgenden Bereichen parametrierbar: • Überwachung der Verbindung • Handshake für Datenübertragung Standardmäßig deaktiviert.
Überwachung der Verbindung	

Einstellung	Beschreibung
Überwachungszeit der Verbindung	Überwachungszeit der Verbindung für Watchdog Die Verbindung zwischen der CPU und WinCC Professional wird mit dem Watchdog-Zähler folgendermaßen überwacht: • Wenn die Verbindung besteht, dann quittiert die Energy Suite-Komponente in WinCC Professional die Anfrage zum Vergleich des Watchdog-Zählers. • Wenn keine Quittierung innerhalb der Überwachungszeit für Watchdog erfolgt, ist die Verbindung unterbrochen und eine Fehlermeldung wird ausgegeben. Folgende Zeitdauer für die Überwachungszeit der Verbindung ist auswählbar: • 1 s • 2 s • 5 s • 10 s • 20 s Standardmäßig sind 5 s eingestellt.
Handshake für Datenübertragung	

Einstellung	Beschreibung
Überwachungszeit	Überwachungszeit für das Schreiben der Energiedaten in das Variablenarchiv Handshake für Datenübertragung erfolgt unter folgenden Bedingungen: • Energy Suite-Komponente quittiert innerhalb der Überwachungszeit die Anfrage für das Schreiben der Energiedaten in das Variablenarchiv von WinCC Professional. • Kein Fehler der Überwachungszeit für Watchdog ist aufgetreten. Folgende Zeitdauer für die Überwachungszeit ist auswählbar: • 10 s • 20 s • 30 s • 60 s • 120 s • 180 s • 240 s • 300 s • 600 s • 900 s Standardmäßig sind 30 s eingestellt.
Wartezeit	Wartezeit für die Wiederholung der Anfrage zum Schreiben der Energiedaten in das Variablenarchiv Die Anfrage wird so lange wiederholt, bis folgende Bedingungen erfüllt sind: • Wartezeit ist abgelaufen. • Energy Suite-Komponente quittiert die Anfrage. Folgende Zeitdauer für die Wartezeit ist auswählbar: • 10 s • 20 s • 30 s • 60 s • 90 s • 120 s Standardmäßig sind 60 s eingestellt.

Hinweis

Ausreichende Überwachungszeit

Passen Sie die Überwachungszeit der Menge der Energieobjekte an. Um so mehr Energieobjekte projektiert sind, um so größer muss die Überwachungszeit sein.

Pufferung

Die periodischen Energiewerte werden zusammen mit dazugehörigen Zeitstempeln in den Datenbausteinen (DBs) zur Pufferung zwischengespeichert, bevor sie im Variablenarchiv von WinCC Professional archiviert werden. Die Zwischenspeicherung vermeidet Datenverluste - z. B. für den Fall, dass die Verbindung zu WinCC Professional unterbrochen ist.

Für die Dauer der Pufferung der periodischen Energiewerte ist die Anzahl und die Größe der Datenbausteine relevant. Für die maximale Dauer der Pufferung ist die Anzahl der zu archivierenden Energiewerte und die eingestellte Archivierungsperiode relevant.

Die folgende Tabelle zeigt die Einstellungen im Bereich "Pufferung" für den Archivierungstyp "WinCC Professional (Variablenarchiv)":

Einstellung	Beschreibung
Anzahl der Datenbausteine	Die Anzahl der Datenbausteine entspricht der Anzahl der angelegten Energieobjekttabellen. Für jede Energieobjekttabelle wird ein Datenbaustein zur Pufferung benötigt. Die Datenbausteine werden bei der Generierung des Energieprogramms automatisch angelegt.
Größe der Datenbausteine	Durch größere Datenbausteine haben Sie die Möglichkeit den Pufferzeitraum der Energiewerte auf der CPU zu verlängern, z. B. bei Kommunikationsausfall zum SCADA-System WinCC Professional. Standardmäßig sind 10 KB eingestellt.
Belegter Speicher	Produkt aus Anzahl multipliziert mit Größe der Datenbausteine

Siehe auch

Beispiele für Archivierung und Pufferung (WinCC Professional) (Seite 65)

Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)

4.8 Beispiele für Archivierung und Pufferung (WinCC Professional)

Beispiele für die Pufferung

Die folgenden Beispiele zeigen den Zusammenhang zwischen der Anzahl der Energieobjekttabellen und der Dauer der Pufferung - abhängig von den CPU. Die Beispiele zeigen die Berechnung mit einer und zwei Energieobjekttabellen.

- Die Dauer der Pufferung ist von der Anzahl der Energieobjekttabellen abhängig, weil für jede Energieobjekttabelle genau ein Datenbaustein zur Pufferung angelegt wird.
- Die Zuordnung einzelner Energieobjekte zur Energieobjekttabelle hat ebenfalls Auswirkung auf die maximale Pufferdauer.

Die folgende Tabelle zeigt die maximale Größe des Datenbausteins zur Pufferung und die Richtwerte für die Dauer der Pufferung eines Datensatzes (z. B. Leistungsaufnahme) bei einer 15 min-Archivierungsperiode, in verschiedenen S7-1500 CPUs. Die Größe eines Archivwerts inkl. Zeitstempel beträgt 150 Byte.

CPU	Maximale Größe des DBs [MB]	Dauer der Pufferung [d]
CPU 1511	1	72
CPU 1516	5	375
CPU 1518	10	750
CPU 1510SP	0,75	52
CPU 1512SP	1	72

Beispiel mit einer Energieobjekttabelle

Die Dauer der Pufferung wird anhand folgender Rahmenbedingungen berechnet:

- Verwendete CPU: CPU 1518
- Maximale Größe des Datenbausteins zur Pufferung: 10 MB
- Anzahl der Energieobjekttabellen: 1
- Anzahl der Energieobjekte: 50
- Zu archivierende Energiedaten pro Energieobjekt:
  - Leistungsaufnahme
  - Energieverbrauch
- Dauer der Archivierungsperiode: 15 min
- Dauer der Pufferung: 7 1/2 Tage

Die Dauer der Pufferung wird folgendermaßen berechnet:

750 Tage / 2 Energiedaten / 50 Energieobjekte = 7 1/2 Tage

Beispiel mit zwei Energieobjekttabellen

Die Dauer der Pufferung wird anhand folgender Rahmenbedingungen berechnet:

- Verwendete CPU: CPU 1518
- Maximale Größe des Datenbausteins zur Pufferung: 10 MB
- Anzahl der Energieobjekttabellen: 2
  - Pro Energieobjekttabelle: 25 Energieobjekte
- Anzahl der Energieobjekte: 50
- Zu archivierende Energiedaten pro Energieobjekt:
  - Leistungsaufnahme
  - Energieverbrauch
- Dauer der Archivierungsperiode: 15 min
- Dauer der Pufferung: 15 Tage

Die Dauer der Pufferung wird folgendermaßen berechnet:

1500 Tage / 2 Energiedaten / 50 Energieobjekte = 15 Tage

Die Dauer der Pufferung ist direkt von der Anzahl der Energieobjekttabellen abhängig, weil für jede Energieobjekttabelle genau ein Datenbaustein zur Pufferung angelegt wird.

Energieprogramm generieren 5

5.1 Allgemeine Einstellungen für das Energieprogramm festlegen

Nachdem die Konfiguration für die Energieobjekte und die Einstellungen für Pufferung und Archivierung abgeschlossen sind, legen Sie bei Bedarf für das Energieprogramm vor dem Generieren folgende allgemeine Eigenschaften fest:

- Freigabe der Meldungsunterdrückung
- Anlaufzeit der Energy Suite-Bausteine

Vorgehen

Um den Dialog für die allgemeinen Einstellungen des Energieprogramms zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Energieobjekte".
2. Doppelklicken Sie auf das Objekt "Energieprogrammeinstellungen".
   Der Dialog "Allgemein" mit den Einstellungen für das Energieprogramm wird im Arbeitsbereich geöffnet.
3. Legen Sie die gewünschten Einstellungen im Dialog "Allgemein" fest. Die Bedeutung der Parameter ist im nachfolgenden Abschnitt "Allgemeine Einstellungen (Seite 68)" detailliert beschrieben.

Siehe auch

Archivierungsperioden definieren (Seite 51)

Archivierung auf S7-CPU (SIMATIC Memory Card) festlegen (Seite 55)

Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)

5.2 Allgemeine Einstellungen

Die folgende Tabelle zeigt die Einstellungen des Energieprogramms unter "Allgemein":

Einstellung	Beschreibung
Meldungsunterdrückung	Auswahl der Variable (Datentyp "BOOL") zur übergeordneten Meldungsunterdrückung für das Energieprogramm. Damit werden z. B. bei Wiederanlauf der CPU nach einem Spanungsausfall die Meldungen von Bausteinen unterdrückt.
Anlaufzeit	Zeitdauer, in der die Energy Suite-Bausteine nach einem Wiederanlauf der CPU deaktiviert bleiben. Standardmäßig sind 10000 ms eingestellt. Beispiel: Mit der Anlaufzeit wird verhindert, dass Feldgeräte, die sich noch im Hochfahren befinden, Fehler verursachen bzw. Fehler wegen des Hochfahrens gemeldet werden. Wenn die Initialisierung der Feldgeräte abgeschlossen ist (Anlauf beendet), sind die Feldgeräte funktionsbereit. Die Einstellung der Anlaufzeit ist von der Anzahl der an der CPU angeschlossenen Geräte abhängig. Je mehr Geräte am PROFINET-Strang vorhanden sind, desto größer muss die Anlaufzeit eingestellt sein.

Siehe auch

Allgemeine Einstellungen für das Energieprogramm festlegen (Seite 67)

Archivierungsperioden (Seite 53)

Archivierung und Pufferung (S7-CPU) (Seite 56)

Archivierung und Pufferung (WinCC Professional) (Seite 61)

5.3 Runtime-Lizenzen für Energy Suite

Energy Suite benötigt Runtime-Lizenzen auf der CPU in Abhängigkeit von der Anzahl der projektierten Energieobjekte.

Folgende Runtime-Lizenzen sind für Energy Suite einstellbar:

- Lizenztyp "5 Energieobjekte"
- Lizenztyp "10 Energieobjekte"

Die vorhandenen Runtime-Lizenzen für Energy Suite werden Ihnen bei der CPU-Projektierung im Inspektorfenster "Eigenschaften" angezeigt unter:

Runtime-Lizenzen > Energy Suite > Runtime-Lizenzen > Anzahl erworbener Lizenzen

---

Hinweis

Runtime-Lizenzen

Bitte informieren Sie sich vor der Generierung des Energieprogramms, ob genügend Runtime-Lizenzen auf Ihrem Engineering System vorhanden sind. Deaktivierte Energieobjekte werden in einer Runtime-Lizenz nicht berücksichtigt.

Fehlende oder unzureichende Runtime-Lizenzen führen zum Abbruch der Programmgenerierung.

Beispiel

Das folgende Bild zeigt die Konfiguration der Runtime-Lizenzen für folgendes Beispiel:

Wenn Sie 32 Energieobjekte im Energieprogramm verwenden, dann benötigen Sie:

- 1 * Lizenztyp "5 Energieobjekte" und 3 * Lizenztyp "10 Energieobjekte"

oder

- 7 * Lizenztyp "5 Energieobjekte"

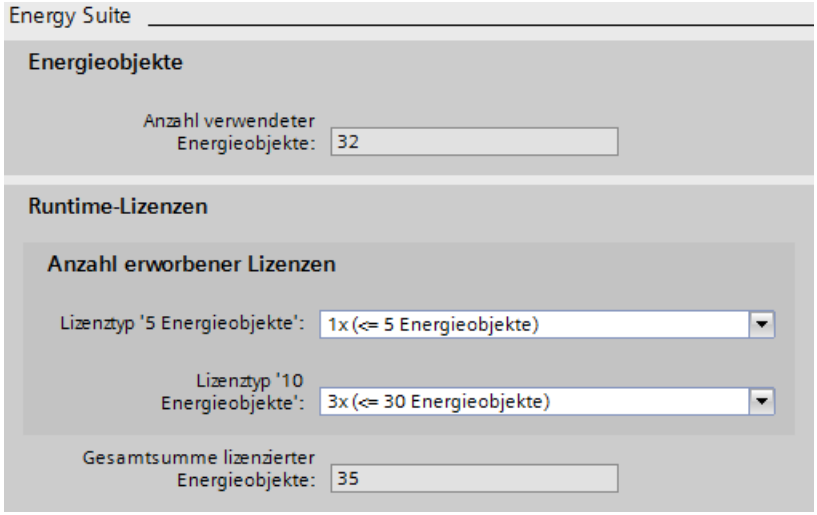

5.4 Energieprogramm generieren

Nach der vollständigen Konfiguration der Energieobjekte und der Einstellungen für das Energieprogramm generieren Sie auf Basis dieser Vorgaben das Energieprogramm.

Voraussetzung

- Alle Energieobjekte, die berücksichtigt werden sollen, sind für das Energieprogramm freigegeben.
- Die für das Energieprogramm freigegebenen Energieobjekte sind vollständig konfiguriert.
- Die Einstellungen für das Energieprogramm sind festgelegt.
- Die Anzahl der benötigten Runtime-Lizenzen für Energy Suite ist ausreichend. Weitere Informationen hierzu finden Sie im Abschnitt "Runtime-Lizenzen für Energy Suite (Seite 69)".
- Die Namensstrukturen·im·SCADA-System·überschreiten·nicht die maximal zulässige·Länge·des·CPU-Namens von 88 Zeichen.

Vorgehen

Um das Energieprogramm zu generieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die S7-1500 CPU.
2. Klicken Sie unterhalb der S7-1500 CPU auf das Objekt "Energieobjekte" und wählen Sie im Kontextmenü den Befehl für die Generierung. Sie können drei Optionen wählen:
   - Energieprogramm generieren > Basisenergiedaten
   - Energieprogramm generieren > Basis- und erweiterte Energiedaten
   - Energieprogramm generieren > Basis- und erweiterte Energiedaten (mit Minimal-/ Maximalwerte)

   Der Generator überprüft die Vollständigkeit und die Plausibilität der Konfiguration der Energieobjekte und startet die Generierung des Energieprogramms. Im Inspektorfenster werden unter "Info > Energy Suite" Informationen und Ereignisse während der Generierung des Energieprogramms angezeigt.
3. Über die Funktion "Gehe zu“ können Sie zur Stelle in der Konfiguration springen, die für das jeweilige Ereignis relevant ist.
4. Passen Sie ggf. die Konfiguration an und starten Sie die Generierung des Energieprogramms erneut.

Ergebnis

Nach einer fehlerfreien und vollständigen Konfiguration der Energieobjekte werden Instanzen von den notwendigen Bausteinen aus der Energy Suite-Bibliothek und der Messhardware-Bibliothek im Ordner "Programmbausteine" unterhalb der S7-1500 CPU angelegt. Zusätzlich werden im Ordner "Programmbausteine" weitere erforderliche Bausteine z. B. Organisationsbausteine und Datenbausteine ergänzt und die notwendigen Verschaltungen hergestellt.

Im Ordner "PLC-Datentypen" unterhalb der S7-1500 CPU werden Instanzen der Datentypen aus der Energy Suite-Bibliothek und der Messhardware-Bibliothek angelegt.

Die Generierung des Energieprogramms erfolgt für alle aktivierten Energieobjekte (unabhängig von der Aufteilung in den Energieobjekttabellen).

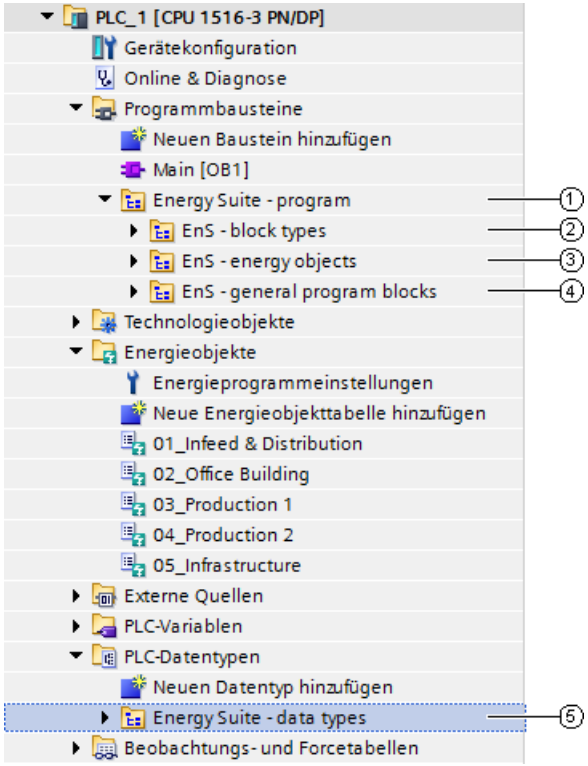

	Objekt	Beschreibung
①	Energy Suite - program	Neuer Ordner für Energieprogramm
②	EnS - block types (Seite 73)	Ordner mit den Energy Suite-Bausteinen und Messhardware-Treiberbausteinen (Funktionsbausteine)
③	EnS - energy objects (Seite 75)	Bausteine für Energieobjekte (Datenbausteine)
④	EnS - general program blocks (Seite 76)	Ordner für weitere Bausteine (Organisationsbausteine, Funktionen, Puffer-DBs, usw.)
⑤	Energy Suite - data types (Seite 79)	Ordner mit den Energy Suite-Datentypen und Messhardware-Datentypen

Siehe auch

Freigabe der Energieobjekte für Energieprogramm ändern (Seite 40)

Allgemeine Einstellungen für das Energieprogramm festlegen (Seite 67)

Wissenswertes zu Energiedaten (Seite 22)

5.5 Struktur des Energieprogramms

5.5.1 Aufbau des Energieprogramms

Bei der Generierung des Energieprogramms werden automatisch folgende Ordner unterhalb der S7-1500 CPU angelegt:

- Ordner "Energy Suite - program" unter "Programmbausteine" mit den Unterordnern
  - EnS - block types (Seite 73)
  - EnS - energy objects (Seite 75)
  - EnS - general program blocks (Seite 76)
- Ordner "Energy Suite - data types (Seite 79)" unter "PLC-Datentypen"

Hinweis

Bausteine des Energieprogramms

SIMATIC Energy Suite generiert für Sie automatisch alle notwendigen Bausteine und Datentypen und deren Verschaltung auf Basis der konfigurierten Energieobjekte und der Energieprogrammeinstellungen.

Für die Standard-Anwendung mit Energy Suite brauchen Sie diese Bausteine nicht weiter anpassen.

⚠ VORSICHT

Manuelle Änderungen an Bausteinen des Energieprogramms

Nehmen Sie nach dem Generieren des Energieprogramms manuelle Änderungen an den Bausteinen vor, dann sind Sie für die Fehlerfreiheit der Bausteine und des Energieprogramms selbst verantwortlich.

⚠ VORSICHT

Verwendung von Energy Suite-Bausteinen in anderen PLC-Programmen

Benutzen Sie die Energy Suite-Bausteine aus der globalen Bibliothek in anderen PLC-Programmen, dann sind Sie für die fehlerfreie Anwendung der Energy Suite-Bausteine selbst verantwortlich.

Siehe auch

Energieobjekte und Energieprogramm (Seite 31)

5.5.2 EnS - block types

Die Struktur Ihres Energieprogramms ist von folgenden Konfigurationen abhängig:

- Energiedatenquellen (Seite 21)
- freigegebenen Energieobjekten (Seite 37)
- Energieprogrammeinstellungen (Seite 67)

Das folgende Bild zeigt die prinzipielle Funktionsweise der Energy Suite-Bausteine im Energieprogramm für Energiedaten aus dem Prozessabbild der CPU (PLC-Variablen) oder aus Datenbausteinen (DB-Variablen).

Bei Eingangsmessdaten von angeschlossener Messhardware kommen die Hardware-spezifischen Funktionsbausteine (FBs) mit Treiberfunktionalität hinzu. Weitere Informationen hierzu finden Sie im Abschnitt "Aufbau des Energieprogramms für Messhardware (Seite 81)".

Funktionsweise der Energy Suite-Bausteine

Der Platzhalter "xxx“ im Bild bedeutet, dass es verschiedene Typen von diesem Baustein gibt, z. B. für "EnS_Drvxxx":

- EnS_DrvPulse
- EnS_DrvCounterLInt

Weitere Typen finden Sie in der Tabelle "EnS - block types" nach dem Bild.

Das folgende Bild zeigt die Funktionsweise der Energy Suite-Bausteine:

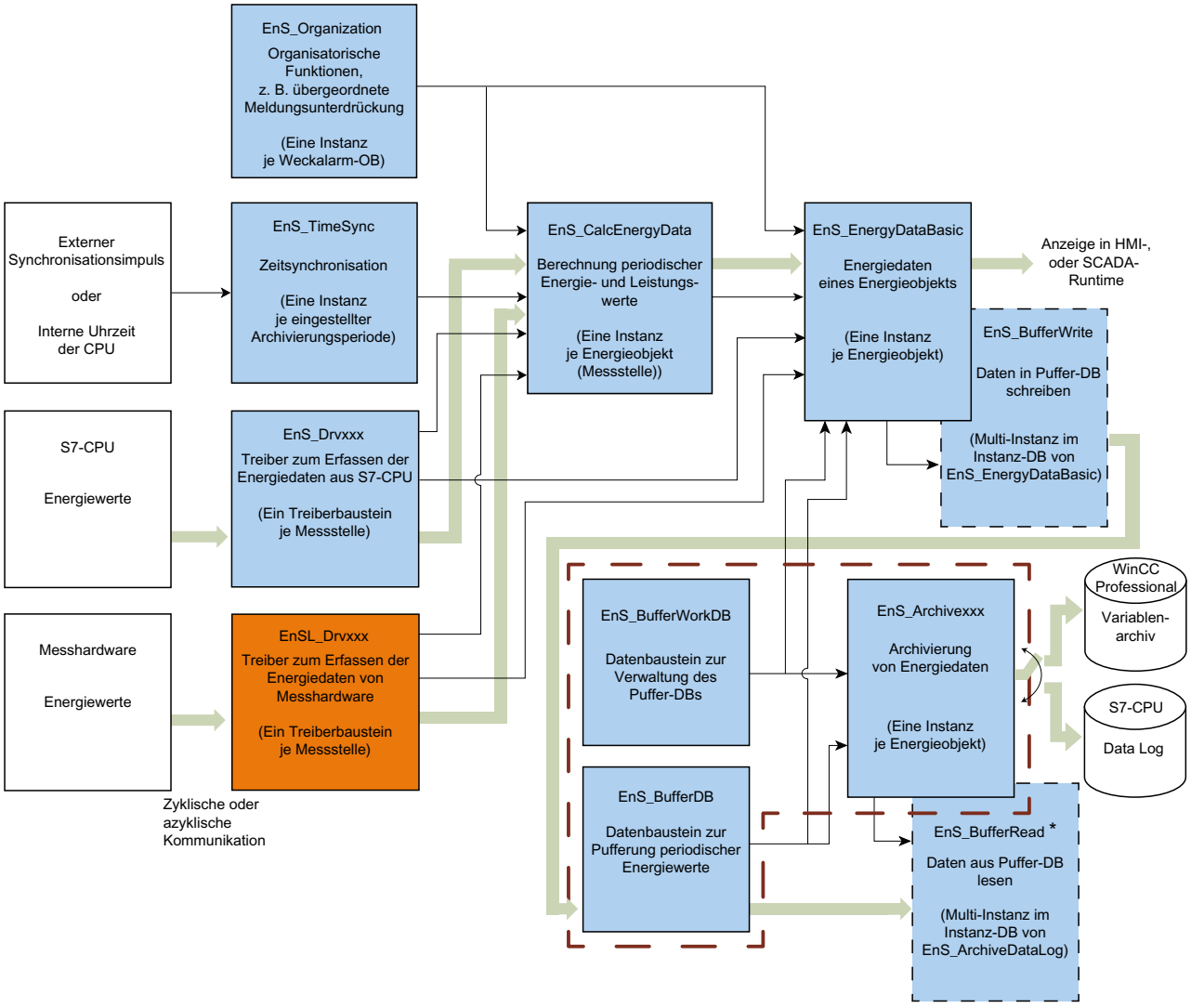

* EnS_BufferRead wird bei der Archivierung in WinCC Professional nicht aufgerufen

Energiedatenfluss

Parametrierdaten

Intern verwendete Bausteine

Einer Energieobjekttabelle zugehörigen Bausteine

EnS - block types

Der Ordner "EnS - block types" enthält nach der Programmgenerierung die Instanzen aller Funktionsbausteine aus der Energy Suite-Bibliothek. Bei Messhardware als Energiedatenquelle enthält der Ordner "EnS - block types" zusätzlich einen Unterordner "S7-1500 blocks" mit den Instanzen der zugehörigen Treiberbausteine aus der Messhardware-Bibliothek (Seite 81).

Funktionsbaustein	Beschreibung	Beispiel in STEP 7 (TIA Portal)
EnS_ArchiveDataLog (Seite 116)	Archiviert Energiedaten in Data Logs (CSV-Datei auf der SIMATIC Memory Card der CPU)	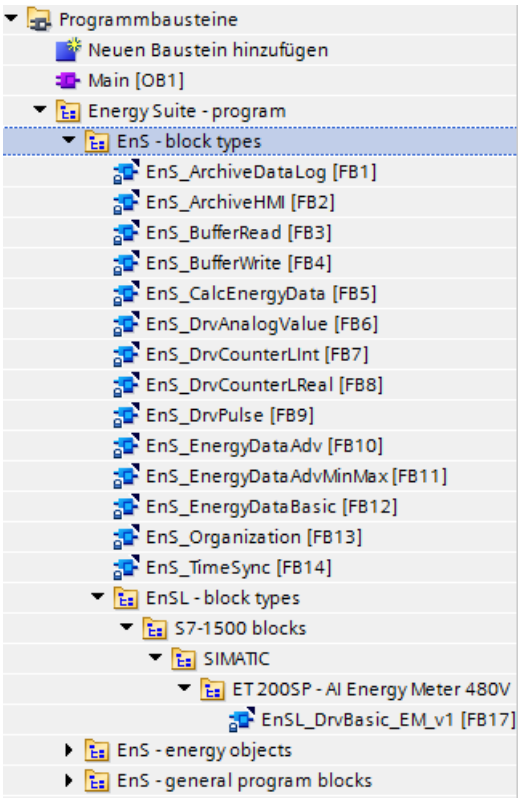
EnS_ArchiveHMI (Seite 124)	Archiviert Energiedaten in Variablenarchiven von WinCC Professional	
EnS_BufferRead (Seite 131)	Liest Daten aus dem Puffer-DB (EnS_BufferDB (Seite 164)). Wird intern verwendet	
EnS_BufferWrite (Seite 133)	Schreibt Daten in den Puffer-DB (EnS_BufferDB (Seite 164)). Wird intern verwendet	
EnS_CalcEnergyData (Seite 136)	Berechnet periodisch die verbrauchte Energie und durchschnittliche Leistungsaufnahme	
EnS_DrvAnalogValue (Seite 142)	Treiberbaustein für analogen Eingang (z. B. Leistungswert, Durchflusswert)	
EnS_DrvCounterLInt (Seite 144)	Treiberbaustein für Zählerwerte vom Typ LInt	
EnS_DrvCounterLReal (Seite 146)	Treiberbaustein für Zählerwerte vom Typ LReal	
EnS_DrvPulse (Seite 148)	Treiberbaustein für einen Impulseingang	
EnS_EnergyDataAdv (Seite 150)	Erweiterte Energiedaten eines Geräts (z. B. phasenspezifische Strom-, Spannungswerte)	
EnS_EnergyDataAdvMinMax (Seite 151)	Minimal- und Maximalwerte der erweiterten Energiedaten eines Geräts	
EnS_EnergyDataBasic (Seite 152)	Basisenergiedaten eines Geräts (z. B. Leistungs-, Energiezählwert)	
EnS_Organization (Seite 156)	Enthält allgemeine Daten und Parameter zur zentralen Programmorganisation	
EnS_TimeSync (Seite 157)	Synchronisiert die Zeiten für die Berechnung der Energiedaten	

5.5.3 EnS - energy objects

Der Ordner "EnS - energy objects" enthält nach der Programmgenerierung für jedes Energieobjekt einen Unterordner mit dem Namen des Energieobjekts "EO - <Name des Energieobjekts>".

Der Ordner "EO - <Name des Energieobjekts>" enthält die Instanz-Datenbausteine

- zu den Funktionsbausteinen, die für das Energieobjekt im Energieprogramm benötigt werden
- mit den periodischen Energiewerten des Energieobjektes, die für die Pufferung und Archivierung bereitstehen

EO - <Name des Energieobjekts>

Der Ordner "EnS_<Name des Energieobjekts>" kann in Abhängigkeit von der Konfiguration des Energieobjekts folgende Datenbausteine enthalten:

Datenbaustein	Beschreibung	Beispiel in STEP 7 (TIA Portal)
<Name des Energieobjekts>	Datenbaustein mit den Informationen zur Messstelle und den Energiewerten, die für die Archivierung bereitstehen	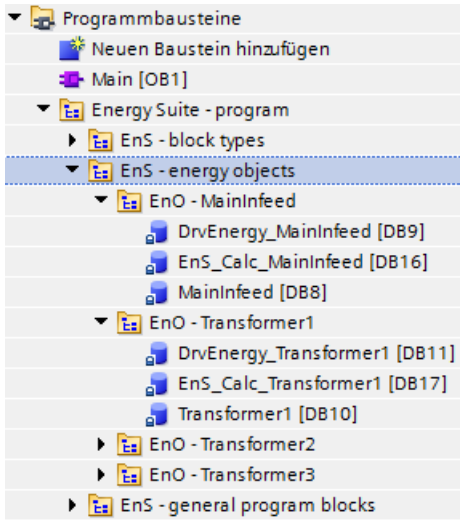
DrvEnergy_<Name des Energieobjekts>	Instanz-DB des FB "EnS_DrvPulse" Beispiel: Von einem Energieobjekt mit dem Namen "PulseGenerator" wird der Instanz-DB "DrvEnergy_PulseGenerator" angelegt.	
EnS_Calc_<Name des Energieobjekts>	Instanz-DB des FB "EnS_CalcEnergyData" Beispiel: Von einem Energieobjekt mit dem Namen "PulseGenerator" wird der Instanz-DB "EnS_Calc_PulseGenerator" angelegt.	

5.5.4 EnS - general program blocks

Der Ordner "EnS - general program blocks" enthält nach der Programmgenerierung weitere für das Energieprogramm notwendige Bausteine, wie Organisationsbausteine, Datenbausteinen und Funktionen (FCs). Die Datenbausteine für die Pufferung der Energiedaten sind ebenfalls als Instanzen aus der Energy Suite-Bibliothek in diesem Ordner angelegt.

EnS - general program blocks

Der Ordner "EnS - general program blocks" enthält nach der Programmgenerierung folgende Bausteine:

Baustein	Beschreibung	Beispiel in STEP 7 (TIA Portal)
Organisationsbaustein		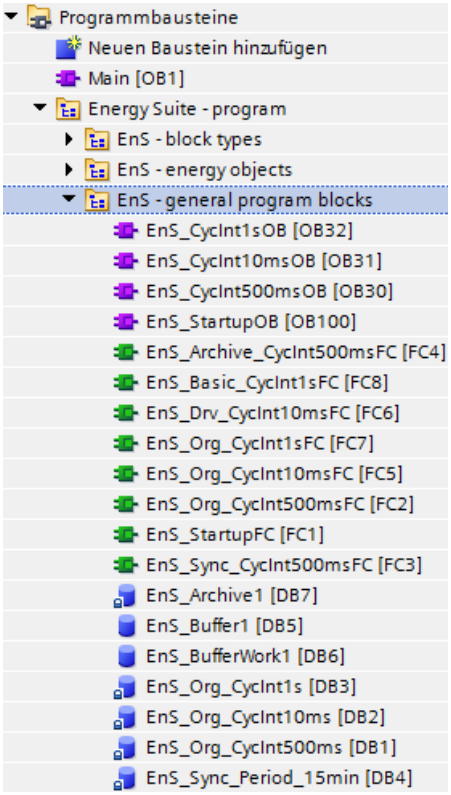
EnS_CycInt<NNNcc>OB	Drei Weckalarm-OBs mit verschiedenen Zeittakten. "NNN" und "cc" sind Platzhalter für den Zeittakt und die Einheit. Beispiel: Bei drei Zeittakten von 100 ms, 1000 ms, 10000 ms werden folgende Weckalarm-OBs angelegt: • "EnS_CycInt100msOB" • "EnS_CycInt1sOB" • "EnS_CycInt10sOB"	
EnS_StartupOB	Anlauf-OB 100	
Funktion		
EnS_Archive_CycInt<NNNcc>	Ein FC für die Archivierung der Energiedaten im zyklischen Weckalarm-OB. "NNN" und "cc" sind Platzhalter für die Zeitdauer und die Einheit. Beispiel: Bei einer Zyklusdauer von 100 ms wird der FC "EnS_Archive_CycInt100ms" angelegt.	
EnS_Basic_CycInt<NNNcc>	Ein FC für die Basisenergiedaten der Energieobjekte im zyklischen Weckalarm-OB. "NNN" und "cc" sind Platzhalter für die Zeitdauer und die Einheit. Beispiel: Bei einer Zyklusdauer von 100 ms wird der FC "EnS_Basic_CycInt100ms" angelegt.	
EnS_Drv_CycInt<NNNcc>	Ein FC für die Treiberbausteine der Energieobjekte im zyklischen Weckalarm-OB. "NNN" und "cc" sind Platzhalter für die Zeitdauer und die Einheit. Beispiel: Bei einer Zyklusdauer von 100 ms wird der FC "EnS_Drv_CycInt100ms" angelegt.	
EnS_Org_CycInt<NNNcc>FC	Drei FCs für die zentrale Organisation des Energieprogramms, die in Weckalarm-OBs und Anlauf-OB aufgerufen werden. "NNN" und "cc" sind Platzhalter für die Zyklusdauer und die Einheit. Beispiel: Bei drei Zyklusdauern von 1 s, 100 ms, 10 s werden folgende FCs angelegt: • "EnS_Org_CycInt1sFC" • "EnS_ Org_CycInt100msFC" • "EnS_ Org_CycInt10sFC"	
EnS_StartupFC	FC, der im Anlauf-OB aufgerufen wird.	
EnS_Sync_CycInt<NNNcc>	Ein FC für die Zeitsynchronisation. "NNN" und "cc" sind Platzhalter für die Zeitdauer und die Einheit.	

Baustein	Beschreibung	Beispiel in STEP 7 (TIA Portal)
	Beispiel: Bei einer Zyklusdauer von 100 ms wird der FC "EnS_Sync_CycInt100ms" angelegt.	
Datenbaustein		
EnS_Archive<N>	Instanz-DB des FB "EnS_ArchiveDataLog". "N" ist ein Platzhalter für die Nummer des Puffers. Beispiel: Bei einer Puffer-Nummer "2" wird der Instanz-DB "EnS_Archive2" angelegt.	
EnS_Buffer<N> (Seite 164)	Datenbaustein zur Pufferung von Energiedatensätzen. "N" ist ein Platzhalter für die Nummer des Puffers. Beispiel: Bei einer Puffer-Nummer "2" wird der Puffer-DB "EnS_Buffer2" angelegt.	
EnS_BufferWork<N> (Seite 165)	Datenbaustein für Kontroll- und Statusinformationen des Puffers. "N" ist ein Platzhalter für die Nummer des Puffers. Beispiel: Bei einer Puffer-Nummer "2" wird der Puffer-DB "EnS_BufferWork2" angelegt.	
EnS_Org_CycInt<NNNcc>	Instanz-DB des FB "EnS_Organization". "NNN" und "cc" sind Platzhalter für die Zyklusdauer und die Einheit. Beispiel: Bei einer Zyklusdauer von 10 ms wird ein Instanz-DB "EnS_Org_CycInt10ms" angelegt.	
EnS_Sync_Period_<NNNcc>	Instanz-DB des FB "EnS_TimeSync". "NNN" und "cc" sind Platzhalter für die Synchronisationsperiode und die Einheit. Beispiel: Bei einer Synchronisationsperiode von 30 Minuten wird der Instanz-DB "EnS_Time30m" angelegt.	

5.5.5 Energy Suite - data types

Die Energy Suite-Bibliothek enthält die für das Energieprogramm notwendigen Datentypen.

Energy Suite - data types

Der Ordner "Energy Suite - data types" enthält nach der Programmgenerierung die Instanzen folgender Datentypen aus der Energy Suite-Bibliothek.

Datentyp	Beschreibung	Beispiel in STEP 7 (TIA Portal)
EnS_typeAnalogValue (Seite 166)	Datentyp für analoge Prozesswerte	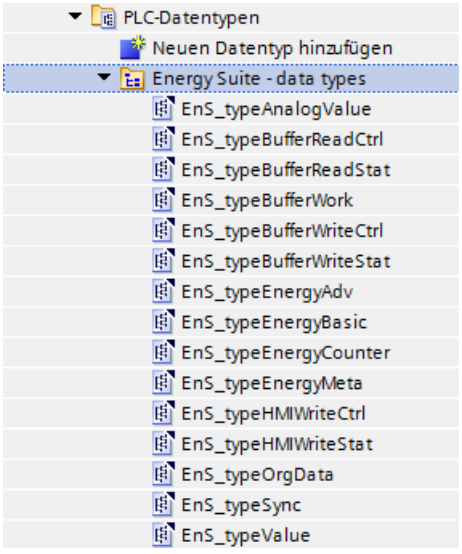
EnS_typeBufferReadCtrl (Seite 166)	Datentyp für Kontrollinformationen des Puffers für das Lesen von Daten	
EnS_typeBufferReadStat (Seite 167)	Datentyp für Daten und Statusinformationen des Puffers für das Lesen von Daten	
EnS_typeBufferWork (Seite 167)	Datentyp für Kontroll- und Statusinformationen des Puffers	
EnS_typeBufferWriteCtrl (Seite 168)	Datentyp für Kontrollinformationen des Puffers für das Schreiben von Daten	
EnS_typeBufferWriteStat (Seite 169)	Datentyp für Statusinformationen des Puffers für das Schreiben von Daten	
EnS_typeEnergyAdv (Seite 169)	Datentyp für erweiterte Energiedaten	
EnS_typeEnergyBasic (Seite 171)	Datentyp für Basisenergiedaten	
EnS_typeEnergyCounter (Seite 171)	Datentyp für Energiezählwert	
EnS_typeEnergyMeta (Seite 172)	Datentyp für Metadaten der Messstelle (Informationen zur Konfiguration des Energieobjekts)	
EnS_typeHMIWriteCtrl (Seite 175)	Datentyp für Kontrollinformationen des Puffers für das Schreiben von Energiedaten in das Variablenarchiv von WinCC Professional	
EnS_typeHMIWriteStat (Seite 176)	Datentyp für Statusinformationen des Puffers für das Schreiben von Energiedaten in das Variablenarchiv von WinCC Professional	
EnS_typeOrgData (Seite 176)	Datentyp für Organisationsdaten	
EnS_typeSync (Seite 177)	Datentyp für die Zeitsynchronisation	
EnS_typeValue (Seite 178)	Datentyp für Ausgangswerte	

5.5.6 Treiberbausteine für Messhardware

5.5.6.1 Aufbau des Energieprogramms für Messhardware

Treiberbausteine der Messshardware für STEP 7 (TIA Portal)

Die Energiedaten (Eingangsmessdaten) der Messhardware werden auf unterschiedliche Weise bereitgestellt (z. B. Unterschiede im Datenformat oder in der Datensatzadresse). Diese Unterschiede erschweren den Zugriff und die Weiterverarbeitung der Energiedaten von Messhardware im PLC-Programm.

Für diesen Zweck stehen für die Messhardware über eine Energy Support Library (EnSL) **zusätzlich** zu den Energy Suite-Bausteinen Hardware-spezifische Funktionsbausteine (FBs) mit Treiberfunktionalität zu Verfügung. Diese Hardware-spezifischen Funktionsbausteine (im Folgenden Treiberbausteine genannt) bilden die Schnittstelle zwischen der jeweiligen Messhardware und Ihrem Automatisierungssystem.

Mithilfe der Treiberbausteine werden die Energiedaten aus der Messhardware ausgelesen und vereinheitlicht dem Automatisierungssystem bereitgestellt. Auf diese Weise werden die Energierohdaten einfacher und schneller Ihrem PLC-Programm und Energy Suite zugeführt. Durch die Komponenten von Energy Suite werden die Energierohdaten anschließend aufbereitet und verarbeitet.

Zusammenspiel der Energy Suite-Bausteine und der Treiberbausteine für Messhardware

Bei der Generierung des Energieprogramms werden (abhängig von den ausgewählten Energiedatenquellen) automatisch die Instanzen der spezifischen Treiberbausteine für die Messhardware im Ordner "EnS - block types" des Energieprogramms angelegt und verschaltet.

In der vorliegenden Dokumentation ist in den folgenden Abschnitten das **prinzipielle Zusammenspiel** der Energy Suite-Bausteine und der spezifischen Treiberbausteine für die Messhardware im Energieprogramm beschrieben.

Dokumentation der spezifischen Treiberbausteine für die Messhardware

Eine **detaillierte Beschreibung** zur Funktionsweise und zu den Bausteinparametern der Treiberbausteine finden Sie in der Dokumentation zur Messhardware. Die Dokumentation zur Messhardware ist Bestandteil der Energy Support Library (EnSL).

5.5.6.2 Energiedatenverarbeitung nach dem "EVA"-Prinzip

Die Energiedatenverarbeitung lässt sich nach dem "EVA"-Prinzip in "Eingabe", "Verarbeitung" und "Ausgabe" unterteilen.

Funktionsweise der Funktionsbausteine

Das folgende Bild zeigt die allgemeine Funktionsweise der Treiberbausteine für die Messhardware nach dem "EVA"-Prinzip:

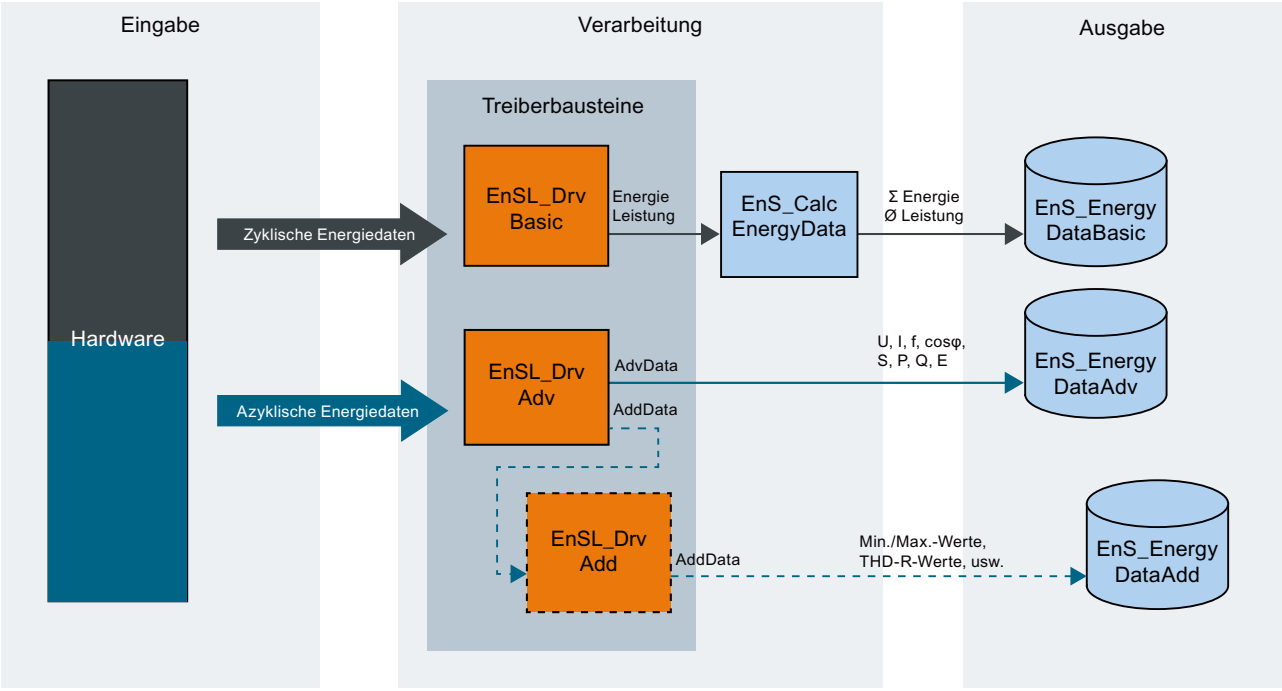

- Energiedaten mittels zyklischer Kommunikation (Prozessabbild)
- Energiedaten mittels azyklischer Kommunikation (Datensatz lesen)
- Treiberbausteine für Messhardware
- Bausteine von SIMATIC Energy Suite
- Erforderliche Bausteine
- Optionale Bausteine

Die folgende Tabelle beschreibt die einzelnen Bereiche des Bildes:

Bereich	Beschreibung
Eingabe	Enthält die Messhardware (z. B. Messgerät), welche die Eingangsmessdaten für die Verarbeitung bereitstellt
Verarbeitung	Erfassung, Berechnung, Zeitstempelung und Bereitstellung der Energiewerte für weitere Systeme durch die jeweiligen Funktionsbausteine
Ausgabe	Bereitstellung der periodischen Energiewerte für verschiedene Systeme zur Visualisierung, Archivierung oder weiteren Auswertung (z. B. SIMATIC Energy Manager PRO)

5.5.6.3 Bereich: Eingabe

Im Bereich "Eingabe" ist die Messhardware enthalten. Die Treiberbausteine lesen die Energiemessdaten aus der Messhardware. Dies erfolgt mithilfe zyklischer Kommunikation (Prozessabbild) und/oder azyklischer Kommunikation (Datensatz lesen).

Das folgende Bild zeigt die "Eingabe" nach dem "EVA"-Prinzip:

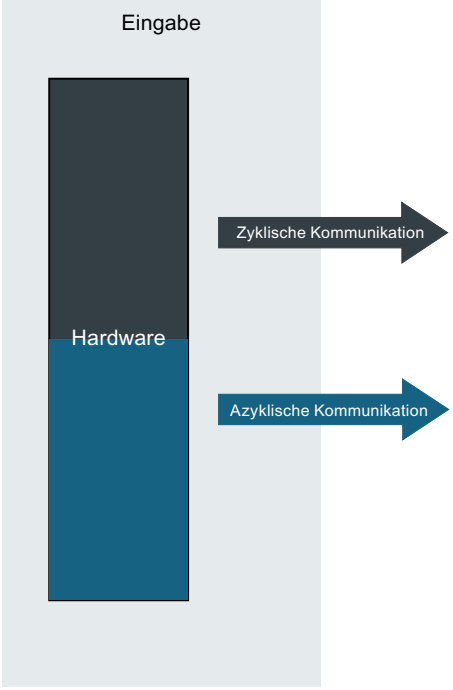

Energiedaten mithilfe zyklischer Kommunikation (Prozessabbild)

Energiedaten mithilfe azyklischer Kommunikation (Datensatz lesen)

Die genaue Beschreibung der Messhardware finden Sie in der Dokumentation des jeweiligen Messgeräts.

5.5.6.4 Bereich: Verarbeitung

Im Bereich "Verarbeitung" werden die Energiedaten von den jeweiligen Treiberbausteinen erfasst, normiert und zu einfach weiterverarbeitbaren Energierohdaten aufbereitet.

Die "Basic"-, "Adv"-, "Add"-Bausteine befinden sich in dem Bereich "Treiberbausteine". Der Treiberbaustein "EnSL_DrvBasic" übergibt die aktuellen Basisenergiedaten an den Energy Suite-Funktionsbaustein "EnS_CalcEnergyData (Seite 136)". Der Funktionsbaustein "EnS_CalcEnergyData" berechnet den Energieverbrauch und die durchschnittliche Leistung über die individuell konfigurierte Archivierungsperiode (z. B. 15 min).

Die genaue Beschreibung der Treiberbausteine finden Sie in der Dokumentation der jeweiligen Messhardware.

Das folgende Bild zeigt beispielhaft die "Verarbeitung" nach den "EVA"-Prinzip. Die Verschaltung der Parameter kann je nach verwendeter Messhardware variieren.

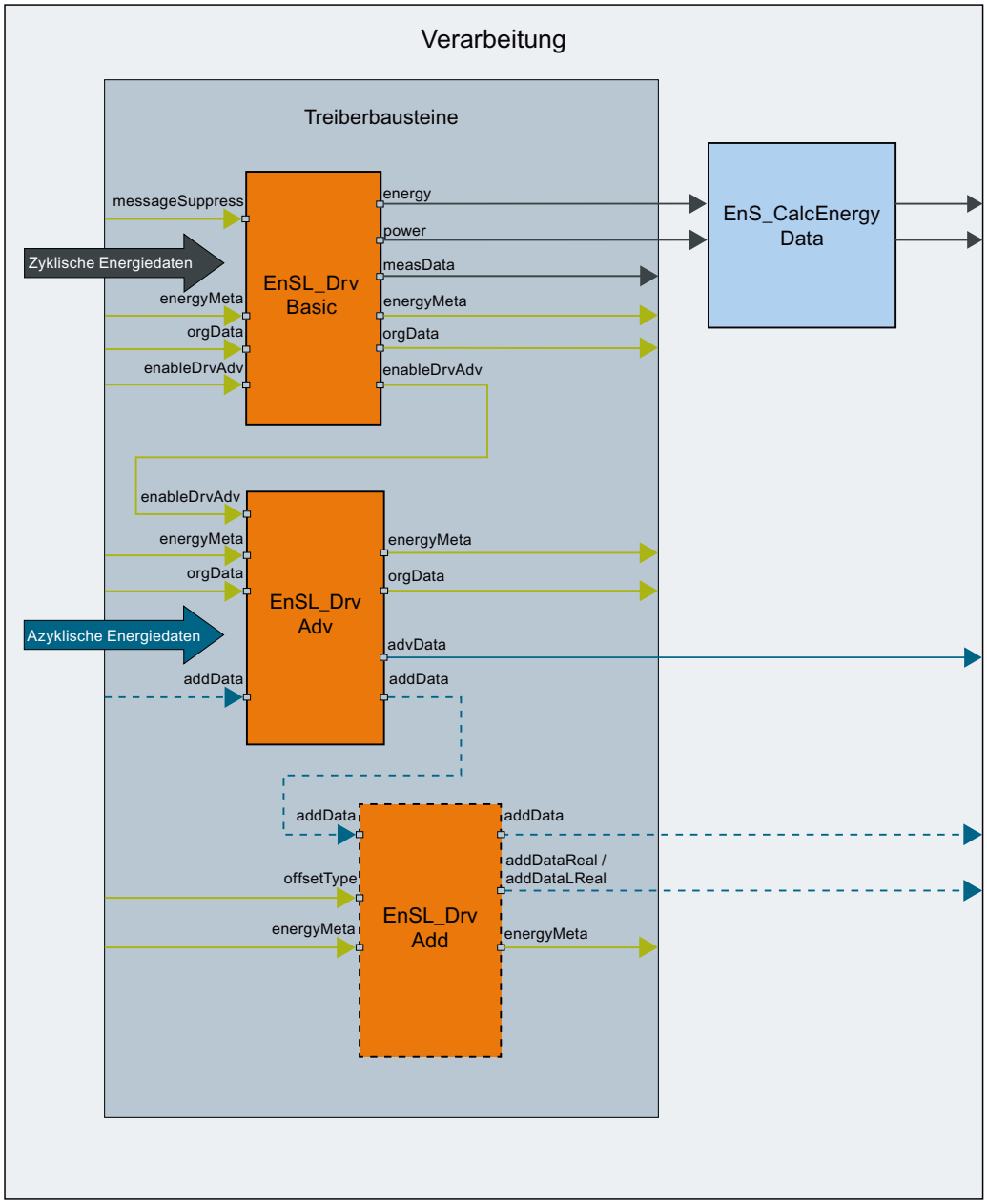

Zyklische Energiedaten

Azyklische Energiedaten

Treiberbausteine für Messhardware

Bausteine von SIMATIC Energy Suite

Erforderliche Bausteine

Optionale Bausteine

Die folgende Tabelle beschreibt die Komponenten des Bildes:

Komponente	Beschreibung
EnSL_DrvBasic	Treiberbaustein für Basisenergiedaten Liest vorwiegend mittels zyklischer Kommunikation die Basisenergiedaten (englisch: Basic) aus der Messhardware aus, z. B. Energiezähler, Leistung
EnSL_DrvAdv	Treiberbaustein für erweiterte Energiedaten Liest mittels azyklischer Kommunikation einen Datensatz mit erweiterten Energiedaten (englisch: Advanced) aus der Messhardware aus, z. B. Spannung, Strom
EnSL_DrvAdd	Treiberbaustein für zusätzliche Energiedaten Liest mittels azyklischer Kommunikation einen Datensatz mit zusätzlichen Energiedaten (englisch: Additional) aus der Messhardware aus, z. B. harmonische Oberschwingung
EnS_CalcEnergyData	Energy Suite-Baustein zur Weiterverarbeitung (periodische Berechnung, Zeitstempelung, etc.) der Energiedaten. Weitere Informationen hierzu finden Sie im Abschnitt "EnS_CalcEnergyData (Seite 136)".
Zyklische Energiedaten	Vorwiegend Basisenergiedaten, die mithilfe zyklischer Kommunikation (Prozessabbild) vom Treiberbaustein "EnS_DrvBasic" aus der Messhardware ausgelesen werden.
Azyklische Energiedaten	Vorwiegend erweiterte und zusätzliche Energiedaten, die mithilfe azyklischer Kommunikation (Datensatz lesen) von den Treiberbausteinen "EnSL_DrvAdv" und "EnSL_DrvAdd" ausgelesen werden.
messageSuppress	Meldungsunterdrückung
energyMeta	Messstellen-spezifische Daten Weitere Informationen hierzu finden Sie im Abschnitt "EnS_typeEnergyMeta (Seite 172)".
orgData	Organisationsdaten Weitere Informationen hierzu finden Sie im Abschnitt "EnS_typeOrgData (Seite 176)".
enableDrvAdv	Verarbeitungsfreigabe des Treiberbausteins "EnSL_DrvAdv"
energy	Aktueller Wert des Wirkenergiezählers
power	Aktueller Wert der Wirkleistung
measValues	Zyklische Messdaten, die an Treiberbaustein "EnSL_DrvAdv" übergeben werden
advData	Erweiterter Messdatensatz
addData	Zusätzliche Energiedaten
addDataLReal / addDataReal	Aktueller LREAL- / REAL-Datensatz

Siehe auch

Wissenswertes zu Energiedaten (Seite 22)

5.5.6.5 Bereich: Ausgabe

In der "Ausgabe" werden die Energiedaten nach der Erfassung und Verarbeitung gesammelt. Die "Ausgabe" dient als Schnittstelle zur Visualisierung mit SIMATIC WinCC oder anderen Systemen (z. B. SIMATIC Energy Manager PRO).

Das folgende Bild zeigt die "Ausgabe" nach dem "EVA"-Prinzip:

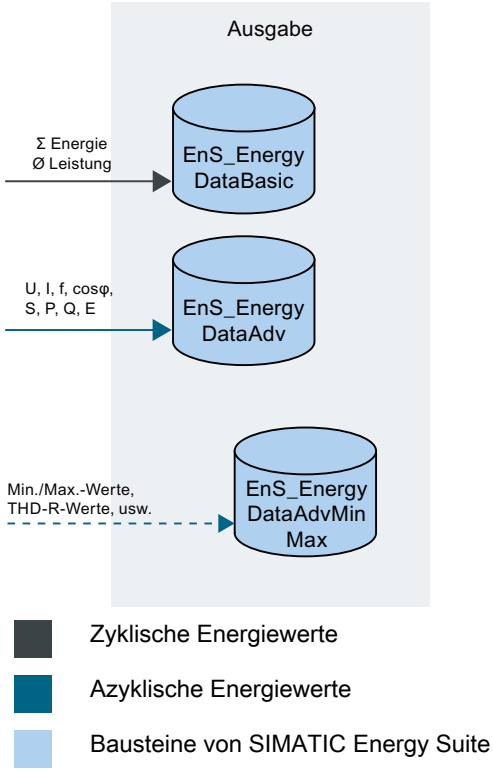

Die folgende Tabelle beschreibt die Komponenten des Bildes:

Funktionsbaustein	Beschreibung
EnS_EnergyDataBasic	Energy Suite-Baustein zur Bereitstellung der verarbeiteten Basisenergie- und Metadaten. Die zyklischen Basisenergiedaten werden von dem Energy Suite-Baustein "EnS_CalcEnergyData" zu folgenden Werten verrechnet: • Energieverbrauch über eine Periode (z. B. 15 min) • Durchschnittliche Leistung über eine Periode (z. B. 15 min) Weitere Informationen finden Sie unter "EnS_EnergyDataBasic: Metadaten und Energiewerte bereitstellen (Seite 152)".
EnS_EnergyDataAdv	Energy Suite-Baustein zur Bereitstellung eines Datensatzes mit erweiterten Energiewerten Erweiterte Energiewerte (azyklisch): • Spannung, Strom, cos φ • Schein-, Wirk-, Blindleistung, usw. Weitere Informationen finden Sie unter "EnS_EnergyDataAdv: Erweiterte Energiewerte bereitstellen (Seite 150)".
EnS_EnergyDataAdvMinMax	Energy Suite-Baustein zur Bereitstellung eines Datensatzes mit Minimal- und Maximalwerten der erweiterten Energiewerte Minimal- und Maximalwerte der erweiterten Energiewerte: • Spannung, Strom, cos φ • Schein-, Wirk-, Blindleistung Weitere Informationen finden Sie unter "EnS_EnergyDataAdvMinMax: Erweiterte Minimal- und Maximalwerte bereitstellen (Seite 151)".

5.6 PLC-Programm übersetzen und laden

Nach der Generierung des Energieprogramms übersetzen Sie das gesamte PLC-Programm und laden es auf das Zielsystem S7-1500 CPU.

Es gibt mehrere Varianten für das Übersetzen und Laden von Projektdaten in die CPU. Nachfolgend ist eine Variante beschrieben. Weitere Informationen hierzu finden Sie in der Online-Hilfe des TIA Portal Informationssystems.

Voraussetzung

- Das Energieprogramm wurde vollständig und fehlerfrei generiert.
- Das PLC-Programm ist ebenfalls vollständig und fehlerfrei.

Übersetzen

Um das gesamte PLC-Programm (inkl. des Energieprogramms) zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die S7-1500 CPU.
2. Klicken Sie unterhalb der S7-1500 CPU auf das Objekt "Programmbausteine" und wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)". Das PLC-Progamm wird übersetzt. Ob das Übersetzen erfolgreich durchgeführt wurde, können Sie im Inspektorfenster unter "Info > Übersetzen" kontrollieren.
3. Korrigieren Sie ggf. die Ursachen möglicher Übersetzungsfehler und starten Sie die Übersetzung erneut.

Laden

Um das gesamte PLC-Programm in die CPU zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die S7-1500 CPU.
2. Klicken Sie unterhalb der S7-1500 CPU auf das Objekt "Programmbausteine" und wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen)".
3. Folgen Sie den Anweisungen des Assistenten für den Ladevorgang. Wählen Sie ggf. die Schnittstelle für die Online-Verbindung zum Zielsystem aus.
   Die Online-Verbindung zur CPU wird aufgebaut.
4. Klicken Sie auf die Schaltfläche "Laden" und anschließend auf die Schaltfläche "Fertig stellen".

War der Ladevorgang erfolgreich, ist das gesamte PLC-Programm (inkl. des Energieprogramms) in die CPU geladen. War der Ladevorgang nicht erfolgreich, dann informieren Sie sich in der Online-Hilfe des TIA Portal Informationssystems, welche weiteren Aktionen erforderlich sind.

5.7 Energieprogramm testen

Mit den Standardfunktionen von STEP 7 (TIA-Portal) können Sie die Erfassung und Verarbeitung der Energiewerte testen.

Voraussetzung

- Das Energieprogramm wurde vollständig und fehlerfrei generiert.
- Das PLC-Programm wurde erfolgreich übersetzt und auf die S7-1500 CPU geladen.
- Eine Online-Verbindung zu S7-1500 CPU besteht.
- Die Programmbearbeitung ist aktiv (CPU im "RUN").

Vorgehen

Um das Energieprogramm zu testen, gehen Sie folgendermaßen vor:

1. Öffnen Sie aus dem Ordner "EO - <Name des Energieobjekts>" den Datenbaustein zum Energieobjekt: <Name des Energieobjekts> [DBx].
   Der Inhalt des Datenbausteins mit den Metadaten und Energiedaten des Energieobjekts wird im Arbeitsbereich geöffnet.
2. Starten Sie die Beobachtung durch Klicken auf das Symbol "Alle beobachten".
   Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an. Unter der Struktur "energyData" können Sie die aktuellen periodischen Energiewerte verfolgen und ggf. steuern.

Verhalten nach Neugenerierung des Energieprogramms

Bei der Neugenerierung des Energieprogramms weist Energy Suite folgendes Verhalten auf:

1. Das gesamte Energieprogramm wird gelöscht.
2. Das Energieprogramm inklusive Änderungen wird neu angelegt.
   Alle unveränderten Energieobjekte werden exakt wie zuvor angelegt.
3. Die Adressen (symbolische Namen) der DBs werden geändert, wenn sich folgende Einstellungen ändern:
   - Name des Energieobjekts (Instanz-DB von EnS_Drvxxx, EnS_CalcEnergyData, EnS_EnergyDataBasic)
   - Name der verwendeten Messhardware in der Hardware-Konfiguration (Instanz-DB von EnSL_DrvBasicxxx)
   - Name der Archivierungsperiode (Instanz-DB von EnS_TimeSync)

Hinweis

Die DB-Nummern werden von STEP 7 (TIA Portal) automatisch vergeben.

5.8 PLC-Programm ohne Installation von Energy Suite bearbeiten

STEP 7 (TIA Portal)-Projekte, die konfigurierte Energieobjekte und Anteile eines Energieprogramms haben, können auch auf STEP 7 (TIA Portal)-Rechnern geöffnet und bearbeitet werden, die keine Installation von Energy Suite haben.

Damit haben Sie z. B. die Möglichkeit, die Gerätekonfiguration oder das restliche PLC-Programm weiter zu bearbeiten. Die vorhandene Projektierung der Energieobjekte und das generierte Energieprogramm bleiben dabei unverändert in dem Projekt bestehen.

Verhalten auf Engineering System ohne Energy Suite-Installation

Beim Öffnen eines solchen STEP 7 (TIA Portal)-Projektes auf einem Engineering System ohne Energy Suite-Installation erscheint ein Dialog mit dem Hinweis, dass keine Installation von Energy Suite vorhanden ist. Nach dem Bestätigen des Dialogs können Sie das Projekt verändern (ausgenommen die Energy Suite-Objekte), übersetzen und auf die Zielsysteme (S7-1500 CPU, PC-Station) laden.

Mögliche Anwendungsfälle

Die folgende Bild und die nachfolgende Tabelle zeigen die möglichen Anwendungsfälle für die Projektbearbeitung ohne Installation von Energy Suite:

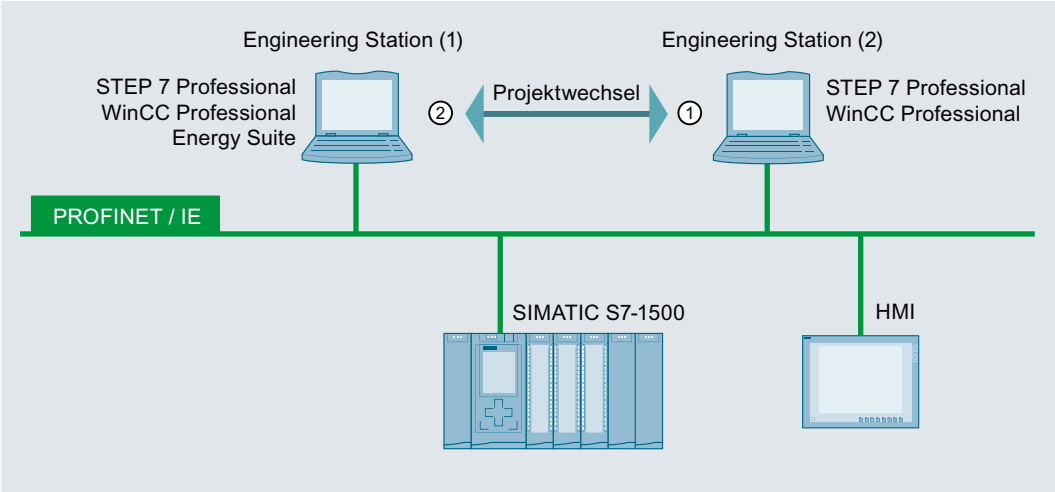

Anwendungsfall	Neu erstelltes/verändertes Projekt auf Engineering Station (1) mit	... kann weiter bearbeitet werden auf Engineering Station (2) mit	Mögliche Bearbeitung
①	• STEP 7 Professional • WinCC Professional • Energy Suite mit mindestens einem Energieobjekt	• STEP 7 Professional	Öffnen und Bearbeiten eines STEP 7 (TIA Portal)-Projekts auf Engineering Station (2), das eine Energy Suite-Konfiguration aber keine Energy Suite-Installation enthält. Die Energy Suite-Konfiguration bleibt unverändert.
		• STEP 7 Professional • WinCC Professional	
Anwendungsfall	**Auf Engineering Station (2) weiter berarbeitetes Projekt mit**	**... kann weiter bearbeitet werden auf Engineering Station (1) mit**	**Mögliche Bearbeitung**

②	• STEP 7 Professional mit unveränderter Energy Suite-Konfiguration • STEP 7 Professional • WinCC Professional mit unveränderter Energy Suite-Konfiguration	• STEP 7 Professional • WinCC Professional • Energy Suite	Öffnen und Bearbeiten eines STEP 7 (TIA Portal)-Projekts auf Engineering Station (1), das eine Energy Suite-Installation enthält, nachdem es vorher auf einem Engineering Station (2) ohne Energy Suite-Installation verändert wurde. Hinweis: Wurde eine PLC (S7-1500 CPU) mit konfigurierten Energieobjekten vorher komplett gelöscht, dann sind auch die Energy Suite-Objekte zu dieser S7-1500 CPU im Projekt nicht mehr vorhanden.

5.9 PLC-Programm ohne Messhardware-Bibliothek mit Treiberbausteinen

STEP 7 (TIA Portal)-Projekte, die konfigurierte Energieobjekte und Anteile eines Energieprogramms zu Messhardware haben, können auch auf STEP 7 (TIA Portal)-Rechnern geöffnet und bearbeitet werden, die keine Installation der Energy Support Library haben.

Damit haben Sie z. B. die Möglichkeit, die Gerätekonfiguration oder das restliche PLC-Programm weiter zu bearbeiten. Die vorhandene Projektierung der Energieobjekte und das generierte Energieprogramm zur Messhardware bleiben dabei unverändert in dem Projekt bestehen.

Verhalten auf Engineering System ohne Energy Support Library

Beim Konfigurieren der Energieobjekte wird bei der Zuweisung der Energiedatenquelle die Messhardware zur nicht mehr vorhandenen Energy Support Library (z. B. nach einer Deinstallation der Energy Support Library) in der Liste der möglichen Energiedatenquellen nicht mehr angezeigt.

Eine bereits zugewiesene Messhardware als Energiedatenquelle zur nicht mehr vorhandenen Energy Support Library verhindert die Neugenerierung des Energieprogramms. Im Auswahlfeld der Energiedatenquelle wird der fehlende Verweis angezeigt. Eine Neugenerierung des Energieprogramms wird abgebrochen, solange der fehlende Verweis nicht behoben ist (z. B. durch Wechsel der Energiedatenquelle oder durch Installation der Energy Support Library).

Das vom Energieprogramm nicht betroffene restliche PLC-Programm können Sie weiterhin ändern, übersetzen und auf die Zielsysteme (S7-1500 CPU, PC-Station) laden.

Energiedaten visualisieren und auswerten 6

6.1 Handlungsübersicht für Archivierung und Visualisierung mit WinCC Professional

Voraussetzung

Um die folgenden Projektierungsschritte durchzuführen, müssen in STEP 7 (TIA Portal) - in der Projektnavigation unterhalb der S7-1500 CPU - folgende Voraussetzungen erfüllt sein:

- Bei jeder CPU, die Energiedaten bereit stellt, ist der Archivierungstyp "WinCC Professional (Variablenarchiv)" aktiviert.
  Weitere Informationen hierzu finden Sie im Abschnitt "Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)".
- Das Energieprogramm wurde vollständig und fehlerfrei generiert.
- Das PLC-Programm wurde erfolgreich übersetzt und auf die S7-1500 CPU geladen.

Projektierungsschritte

Die folgende Tabelle zeigt eine Übersicht der wichtigsten Projektierungsschritte für die Visualisierung und Archivierung der periodischen Energiedaten mit dem SCADA-System WinCC Professional.

Diese Schritte werden in STEP 7 (TIA Portal) in der Projektnavigation unterhalb der PC-Station ausgeführt.

Thema	Schritt		Beschreibung
Energiedaten visualisieren und auswerten	1.1	PC-Station mit WinCC Professional konfigurieren	Für das SCADA-System benötigen Sie in Ihrem Projekt eine PC-Station mit der SIMATIC HMI Applikation "WinCC RT Professional".
	1.2	Verbindung zum SCADA-System anlegen	Um den Datentransfer zwischen der S7-1500 CPU und dem SCADA-System zu ermöglichen, projektieren Sie in STEP 7 (TIA Portal) eine HMI-Verbindung zur PC-Station mit WinCC Professional.
	1.3	Auf Engineering System für Archivierung projektierte Energiewerte auf PC-Station einlesen und anzeigen (Seite 96)	Auf der PC-Station mit WinCC Professional bietet Energy Suite eine Übersicht über alle Energieobjekte und den periodischen Energiewerten an, die Sie im Engineering System zur Archivierung im Variablenarchiv von WinCC Professional projektiert haben. Über die Funktion "Aktualisieren" erfolgt ein Abgleich mit den Engineering Daten der von der PC-Station mit WinCC Professional erreichbaren S7-1500 CPUs.
	1.4	SCADA-System projektieren (Seite 99)	Projektierung der PC-Station mit WinCC Professional: • Für den Datentransfer zum SCADA-System erzeugen Sie die HMI-Variablen für die Datenbausteine zur Pufferung (Ordner "HMI-Variablen"). • Für die Archivierung generieren Sie das Variablenarchiv mit den Archivvariablen (Energiewerte) und richten die Archivsegmente für die Archivierung (Seite 102) der periodischen Energiewerte ein (Ordner "Archive"). • Für die Visualisierung der erfassten Energiedaten im Runtime können Sie Vorlagenbilder und Bildelemente aus dem Siemens Industry Online Support in die WinCC Applikation einbinden (Ordner "Bilder"). • Für die Visualisierung der erfassten Energiedaten im Runtime können Sie auch die automatisch installierten Bildbausteine aus der Energy Suite-Bibliothek verwenden.
	1.5	PC-Station mit WinCC Professional übersetzen und laden	Anschließend übersetzen Sie die Projektierung für die gesamte PC-Station und laden sie auf das SCADA-System.
	1.6	PLC-Programm und SCADA-System starten (Runtime)	Nachdem sowohl WinCC RT Professionell auf dem SCADA-System als auch die PLC-Programme auf den S7-1500 CPUs gestartet sind, ist das Archivieren betriebsbereit. Die gewählten periodischen Energiewerte werden automatisch archiviert. Das Archivieren wird stetig ausgeführt, solange Runtime aktiv ist.

6.2 Energiewerte auf PC-Station mit WinCC Professional

Die auf der S7-1500 CPU aufbereiteten periodischen Energiewerte können in das Variablenarchiv von WinCC Professional zur Langzeitarchivierung übertragen werden. Dies ermöglicht z. B.:

- aus Energiewerten Berichte zu erzeugen
- die Energiewerte innerhalb der WinCC Professional-Umgebung zu visualisieren
- die Energiewerte an SIMATIC Energy Manager PRO zu weiteren Verarbeitung in der Management-Ebene zu übertragen

Auf der PC-Station mit WinCC Professional bietet Energy Suite eine Übersicht über alle Energieobjekte und den periodischen Energiewerte an, die Sie zur Archivierung im Variablenarchiv von WinCC Professional aktiviert haben.

Nach der erfolgreichen Installation von SIMATIC Energy Suite gibt es hierfür den neuen Ordner "Energiedaten" in der Projektnavigation unterhalb der PC-Station mit WinCC Professional.

Das folgende Bild zeigt den Ordner "Energiedaten" in der Projektnavigation.

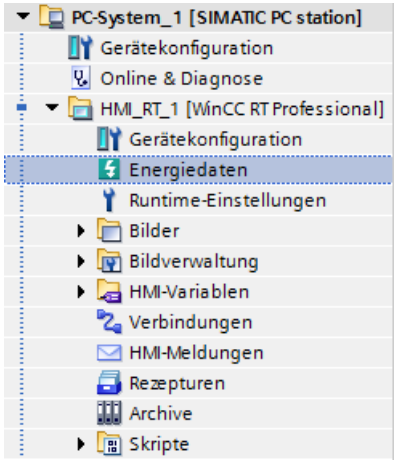

Siehe auch

Energiewerte für die Archivierung in WinCC Professional anzeigen (Seite 96)

6.3 Energiewerte für die Archivierung in WinCC Professional anzeigen

Sie möchten eine Übersicht über alle Energieobjekte und deren periodischen Energiewerte erhalten, die Sie zur Archivierung im Variablenarchiv von WinCC Professional projektiert haben.

Voraussetzung

- Bei jeder CPU, die Energiedaten bereit stellt, ist der Archivierungstyp "WinCC Professional (Variablenarchiv)" aktiviert.
  Weitere Informationen hierzu finden Sie im Abschnitt "Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)".
- Eine PC-Station mit der SIMATIC HMI Applikation "WinCC RT Professional" ist in Ihrem Projekt konfiguriert.
- Es besteht eine gültige HMI-Verbindung zwischen S7-1500 CPU und der PC-Station mit WinCCProfessional.
  Weitere Informationen hierzu finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "HMI-Verbindungen projektieren > Neue Verbindung grafisch anlegen".

Vorgehen

Um die Energiewerte für die Protokollierung und Archivierung anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die PC-Station.
2. Öffnen Sie unterhalb der PC-Station die Applikation "WinCC RT Professional".
   Das Objekt "Energiedaten" ist unterhalb der Applikation sichtbar.
3. Doppelklicken Sie auf das Objekt "Energiedaten".
   Eine Tabelle mit allen Energieobjekten der S7-1500 CPUs des gesamten Projekts wird im Arbeitsbereich geöffnet. Sie haben damit einen Überblick über alle verfügbaren Energieobjekte und deren im Variablenarchiv von WinCC Professional zu archivierenden periodischen Energiewerte.
4. Klicken Sie auf das Symbol "Aktualisieren".
   Die angezeigten Daten in der Tabelle werden neu eingelesen. Es erfolgt ein Abgleich mit den Engineering Daten der von dieser PC-Station mit WinCC Professional erreichbaren S7-1500 CPUs.

Ergebnis

Es wird zu jedem Energieobjekt die Freigabe für das Energieprogramm und die Projektierung folgender Energiewerte für die Archivierung angezeigt:

- Leistungsaufnahme
- Energieverbrauch
- Energiezählwert

Die Bedeutung der Anzeigen ist im Abschnitt "Übersicht der Energiedaten (Seite 98)" detailliert beschrieben.

Hinweis

Mengengerüst

- Es können maximal 2000 Energieobjekte mit WinCC Professional archiviert werden.
- Es können die Energieobjekte von maximal 10 S7-1500 CPUs archiviert werden.

6.4 Übersicht der Energiedaten

Die Tabelle "Energiedaten" in der Applikation "WinCC RT Professional" gibt eine Übersicht über die parametrierten Energieobjekte und deren im Variablenarchiv von WinCC Professional zu archivierenden periodischen Energiewerte.

Hinweis

Anpassung der periodischen Energiewerte

In WinCC Professional werden die periodischen Energiewerte nur angezeigt. Um die Einstellungen der periodischen Energiewerte zu ändern, müssen Sie zur Parametrierung der Energieobjekte und zu den Energieprogrammeinstellungen unterhalb der S7-1500 CPUs wechseln.

Die folgende Tabelle zeigt die Einstellungen für die Erfassung und Archivierung der Energiewerte im Variablenarchiv von WinCC Professional:

Einstellung	Beschreibung
Name des Energieobjekts	Zeigt den Namen des Energieobjekts an. Der Name ist nur bei der Parametrierung des Energieobjekts unterhalb der S7-1500 CPU änderbar.
Name der CPU	Zeigt den Namen der S7-1500 CPU an, der das Energieobjekt zugewiesen ist
Freigabe Energieobjekt	Zeigt den Freigabestatus des Energieobjekts für die Generierung des Energieprogramms an. Die Freigabe aktivieren Sie bei der Parametrierung des Energieobjekts unterhalb der S7-1500 CPU.
Leistungsaufnahme	Zeigt an, ob die periodischen Energiewerte der Leistungsaufnahme für die Archivierung aktiviert sind
Energieverbrauch	Zeigt an, ob die periodischen Energiewerte des Energieverbrauchs für die Archivierung aktiviert sind
Energiezählwert	Zeigt an, ob die periodischen Energiewerte des Energiezählwerts für die Archivierung aktiviert sind
Zeitdauer der Archivierungsperiode	Zeigt die Zeitdauer der Archivierungsperiode an. Die Zeitdauer ist nur in den Energieprogrammeinstellungen unterhalb der S7-1500 CPU änderbar.
Name der Archivierungsperiode	Zeigt den Namen der Archivierungsperiode an. Der Name ist nur in den Energieprogrammeinstellungen unterhalb der S7-1500 CPU änderbar.

Hinweis

Mengengerüst

- Es können maximal 2000 Energieobjekte mit WinCC Professional archiviert werden.
- Es können die Energieobjekte von maximal 10 S7-1500 CPUs archiviert werden.

Siehe auch

Energiewerte für die Archivierung in WinCC Professional anzeigen (Seite 96)

Energiewerte auf PC-Station mit WinCC Professional (Seite 95)

6.5 Archivierungstruktur für Energiewerte erzeugen

Nach dem Abgleich der Energiedaten des Engineering Systems mit den Energiedaten auf der PC-Station mit WinCC Professional erzeugen Sie die HMI-Variablen und das Variablenarchiv mit den Archivvariablen für die zu archivierenden Energiewerte. Dieser Generiervorgang erfolgt automatisch über ein Schaltfläche - eine nachträgliche manuelle Anpassung ist nicht erforderlich.

Die generierten Daten können anschließend übersetzt und auf das SCADA-System geladen werden bzw. stehen für die Visualisierung in Runtime zur Verfügung.

Voraussetzung

- Bei jeder CPU, die Energiedaten bereit stellt, ist der Archivierungstyp "WinCC Professional (Variablenarchiv)" aktiviert.
  Weitere Informationen hierzu finden Sie im Abschnitt "Archivierung auf PC-Station mit WinCC Professional (Variablenarchiv) festlegen (Seite 59)".
- Eine PC-Station mit der SIMATIC HMI Applikation "WinCC RT Professional" ist in Ihrem Projekt konfiguriert.
- Es besteht eine gültige HMI-Verbindung zwischen S7-1500 CPU und PC-Station mit WinCCProfessional.
  Weitere Informationen hierzu finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "HMI-Verbindungen projektieren > Neue Verbindung grafisch anlegen".

Vorgehen

Um die Archivierungsstruktur für die Energiewerte zu generieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die PC-Station.
2. Öffnen Sie unterhalb der PC-Station die Applikation "WinCC RT Professional".
   Das Objekt "Energiedaten" ist unterhalb der Applikation sichtbar.
3. Doppelklicken Sie auf das Objekt "Energiedaten".
   Die Tabelle mit allen Energieobjekten und deren zu archivierenden periodischen Energiewerte, struktruriert nach den S7-1500 CPUs des gesamten Projekts, wird im Arbeitsbereich geöffnet.
4. Klicken Sie auf das Symbol "Variablenarchiv anlegen".

Ergebnis: HMI-Variablen erzeugt

Im Ordner "HMI-Variablen" in der Projektnavigation unterhalb der PC-Station wird automatisch eine neuer Ordner "Energy Suite - HMI-Variablen" angelegt mit allen notwendigen Variablen für die Datenbausteine zur Pufferung.

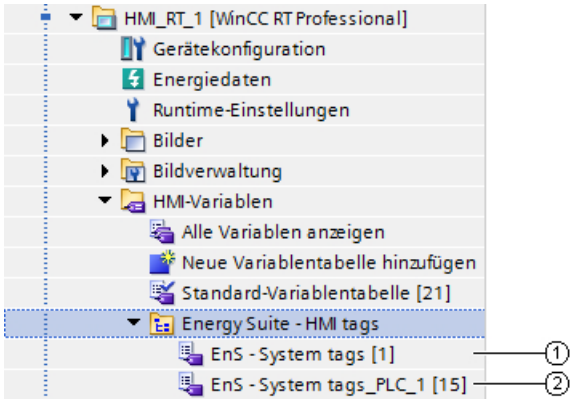

	Objekt	Beschreibung
①	EnS - System tags	Interne Variable @EnS_SystemTag für die Verbindung zum Variablenarchiv
②	EnS - System tags_<Name der CPU>	Für jeden Datenbaustein n (n = 1, 2, 3, ...) zur Pufferung (und damit zu jeder Energieobjekttabelle) werden folgende HMI-Variablen angelegt: • @EnS_Archive<n>_request_<Name der CPU> mit dem Datentyp EnS_typeHMIWriteCtrl • @EnS_Archive<n>_statusWrite_<Name der CPU> mit dem Datentyp EnS_typeHMIWriteStat • @EnS_Buffer<n>_<Name der CPU> mit dem Datentyp EnS_typeValue

Die zugehörige Tabelle im Arbeitsbereich zeigt die angelegten HMI-Variablen und Ihre HMI-Verbindung zur S7-1500 CPU. Die HMI-Variablen sind nicht veränderbar.

Ergebnis: Variablenarchiv mit Archivvariablen

Im Arbeitsbereich zum Ordner "Archive" wird automatisch ein neues Variablenarchiv mit dem Namen "EnS_EnergyArchive" mit Default-Einstellungen angelegt. Die Einstellungen "Gesperrt", "Manuelle Eingaben erlaubt" können geprüft und bei Bedarf geändert werden.

Im Bereich "Archivvariablen" ist für jeden zu archivierenden Energiewert aller projektierten Energieobjekte eine Archivvariable mit dem Namen des Energieobjektes und den Erweiterungen _energy, _power bzw. _energyCounter angelegt. Die generierten Archivvariablen sind schreibgeschützt und mit der internen Variable "@EnS_SystemTag" verbunden.

In den Spalte "Kommentar für Runtime" sind Parameterwerte angelegt, die von SIMATIC Energy Manager PRO zu weiteren Verarbeitung interpretiert werden können.

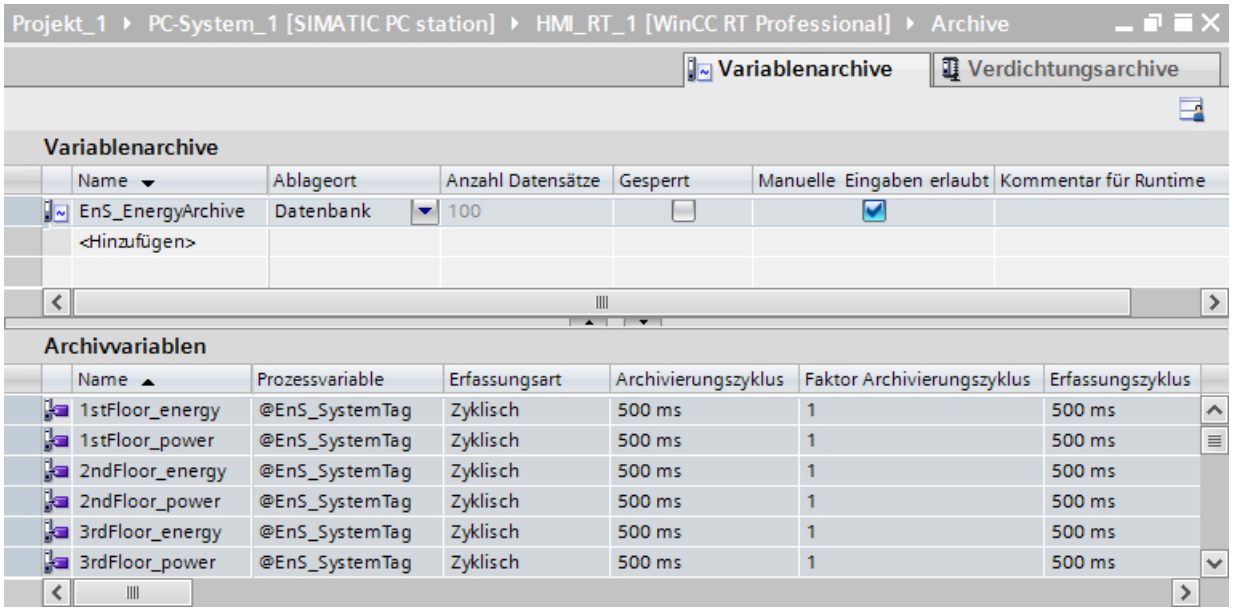

Zusätzlich wird die Energy Suite-Komponente in die Anlaufliste von WinCC Runtime eingetragen unter:

Runtime-Einstellungen > Service > Zusätzliche Tasks/Anwendungen

Die generierten Daten können übersetzt und auf das SCADA-System geladen werden bzw. stehen für die Visualisierung in Runtime zur Verfügung.

Vor dem Übersetzen überprüfen Sie noch die Einstellungen der Archivsegmente des Variablenarchivs (Seite 102) in den Runtime-Einstellungen der PC-Station.

6.6 Variablenarchiv und Segmentierung für Energy Suite

Das auf der PC-Station mit WinCC Professional automatisch angelegte Variablenarchiv ermöglicht die Langzeitarchivierung der zeitgestempelten Energiewerte, so dass die archivierten historischen Daten für empirische statistische Auswertungen genutzt werden können.

In den Runtime-Einstellungen unterhalb der PC-Station mit WinCC Professional haben Sie zusätzlich die Möglichkeit das Variablenarchiv zu segmentieren und somit seine Größe zu definieren.

Das Einrichten der Archivsegmente für die Archivierung der periodischen Energiewerte nehmen Sie im Arbeitsbereich vor unter:

Runtime-Einstellungen > Archivierung > Archivsegmente

Empfehlung für Segmentierung des Variablenarchivs

In einem segmentierten Umlaufarchiv werden mehrere gleich große Archivsegmente erstellt, die nacheinander mit periodischen Energiedaten (Prozesswerten) gefüllt werden. Wenn alle Segmente vollständig gefüllt sind, wird das älteste Segment neu beschrieben und die darin enthaltenen Energiedaten gehen verloren.

Für die Langzeitarchivierung wird nur auf die Energiedaten zugegriffen, die in dem Zeitraum liegen, der alle Segmente umfasst. Standardmäßig ist der Zeitraum auf 1 Woche eingestellt.

Für die durchgängige Langzeitarchivierung der Energiedaten gelten folgende Energy Suite-Empfehlungen:

- Zeitraum über alle Segmente: 2 Jahre
- Maximale Größe aller Segmente: 24 GB
- Zeitraum, den ein Einzelsegment umfasst: 1 Monat
- Maximale Größe eines Segments: 1 GB
- Backup: aktiviert

Zusätzlich stellen Sie den Pfad für das Backup auf einem Backup-Server ein.

6.7 Energiedaten archivieren

Nachdem sowohl WinCC RT Professionell als auch das PLC-Programm gestartet sind, ist das Archivieren betriebsbereit. Die periodischen Energiewerte werden automatisch archiviert. Das Archivieren wird stetig ausgeführt, solange Runtime aktiv ist.

Voraussetzung

- Das Projekt ist übersetzt und auf die Zielsysteme (CPUs und PC-Station mit WinCC Professional) geladen.
- Die Online-Verbindungen zu den S7-1500 CPUs bestehen.
- Die Programmbearbeitung auf den S7-1500 CPUs ist aktiv (CPU im "RUN").
- PC-Station mit WinCC Professional befindet sich in Runtime ("Runtime starten").

Ergebnis

Die archivierten Daten umfassen:

- Von der CPU erzeugten Zeitstempel
- Parameter der Energieobjekte, z. B.:
  - Name des Energieobjekts
  - Energieobjektkennung
  - oberer/unterer Grenze
  - Name des Variablenarchivs
- Periodische Energiewerte über eine Archivierungsdauer:
  - Energiezählwert
  - Energieverbrauch
  - Leistungsaufnahme
- Variablenattribute (z. B. Qualitiy Code)

Table view

Time column	EO1_Power	EO1_Energy	EO2_Power	EO2_Energy
6/16/2016 1:02:00 PM	111.11100769043 m.	1.8518500328064 m.	222.222015380859 m.	3.70370006561279 m.
6/16/2016 1:03:00 PM	111.11100769043 m.	1.8518500328064 m.	222.222015380859 m.	3.70370006561279 m.
6/16/2016 1:04:00 PM	111.11100769043 m.	1.8518500328064 m.	222.222015380859 m.	3.70370006561279 m.
6/16/2016 1:05:00 PM	111.11100769043 m.	1.8518500328064 m.	222.222015380859 m.	3.70370006561279 m.
6/16/2016 1:06:00 PM	111.11100769043 m.	1.8518500328064 m.	222.222015380859 m.	3.70370006561279 m.

6.8 Energiedaten visualisieren mit WinCC Online Trend View

Den Verlauf der periodischen Energiewerte können Sie verfolgen

- grafisch über WinCC Online Trend View (mit Verbindung zum Variablenarchiv)
- numerisch über das Energy Suite Export Tool (Seite 105) (gehört zum Installationsumfang von Energy Suite)

Visualisierung mit WinCC Online Trend View

Mit WinCC Online Trend View können die in den Variablenarchiven archivierten periodischen Energiewerte in Runtime als Kurve dargestellt werden. Die grafische Darstellung der Energiewerte können Sie selbst projektieren. Dazu legen Sie Kurvenfenster an und verbinden die Werte in der Kurve mit den Archivvariablen der gewünschten Energiewerte.

Weitere Informationen hierzu finden Sie im WinCC Informationssystem unter dem Stichwort "WinCC Online Trend View".

Das folgende Bild zeigt ein Beispiel für die Anzeige von Energiewerten in WinCC Online Trend View:

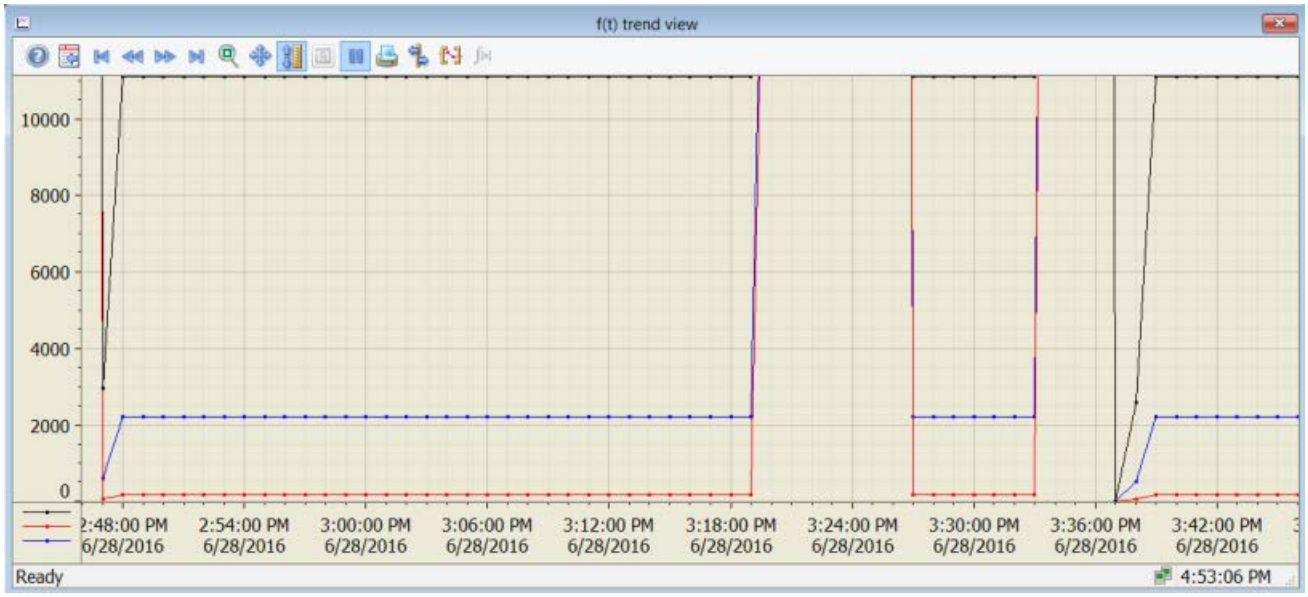

6.9 Energiedaten auswerten mit Energy Suite Export Tool

Das Energy Suite Export Tool unterstützt Sie bei der Auswertung der erfassten Energiewerte im Microsoft Excel-Format.

Das Energy Suite Export Tool ist Bestandteil von SIMATIC Energy Suite Runtime und wird mit diesem zusammen auf der PC-Station mit WinCC Professional installiert. Weitere Informationen hierzu finden Sie in der Installationsanleitung zu Energy Suite auf der DVD.

Mit dem Energy Suite Export Tool können Sie die in den Variablenarchiven archivierten periodischen Energiewerte in Runtime in eine externe Datei exportieren. Die so exportierten Daten können Sie anschließend in Microsoft Excel öffnen und auswerten.

Zusätzlich können Sie mit dem Energy Suite Export Tool Kostenstellenberichte mit den von Ihnen angelegten Tarifsätzen generieren. Diese Kostenstellenberichte ordnen Sie Ihren Kostenstellen zu.

Hinweis

Um einen Kostenstellenbericht zu generieren, müssen Sie mindestens einen Tarifsatz und eine Kostenstelle anlegen.

Basierend auf den definierten Tarifsätzen und Kostenstellen, zeigt Ihnen der Kostenstellenbericht den Energieverbrauch und die daraus resultierenden Energiekosten an. Den Kostenstellenbericht können Sie in Microsoft Excel öffnen und auswerten.

Mögliches Exportformat:

- .xlsx
- Kompatibel zu Microsoft Excel Version 2010 und höher

Mögliche Berichtserstellung:

- Manueller Bericht
- Automatischer Bericht

Energy Suite Export Tool starten und manuellen Bericht erstellen

1. Wählen Sie den Menübefehl Start > Siemens Automation > Options and Tools > SIMATIC Energy Suite Runtime Toolbox > SIMATIC Energy Suite Export Tool.
   Das Energy Suite Export Tool wird geöffnet.
2. Prüfen Sie den Namen des WinCC Servers:
   - Wenn WinCC Professional in Runtime ist, dann ist der Server-Name (Name des lokalen Runtime-PCs) automatisch eintragen.
   - Andernfalls können Sie über den Menübefehl Einstellungen > WinCC Server die Server-Verbindung manuell einstellen.
3. Wählen Sie die Oberflächensprache (Deutsch, Englisch) über den Menübefehl Einstellungen > Sprache.

4. Konfigurieren Sie die Vorlage für die Berichte über das Menü Bericht. Folgende Einstellungen sind z. B. möglich:
   - Berichtsart: Archivmesswerte-Export oder Kostenstellen-Bericht
   - Ausführung als "Manuell"
   - Berichtsname
   - Dateiname (setzt sich zusammen aus dem Namen des Berichts, Datum und Uhrzeit der Berichtserstellung)
   - Ablagepfad
   - Tarifsatz (für Kostenstellen-Bericht)
   - Einheit (wird automatisch entsprechend Tarifsatz festgelegt)
   - Variablenarchiv und Archivvariablen für die Auswertung (für Archivmesswerte-Export)
   - Kostenstelle (für Kostenstellen-Bericht)
   - Zeitraum, den der Bericht auswertet (Berichtszeitraum)

   Hinweis

   Die Einheit für Kubikmeter m^3 wird im Energy Suite Export Tool angezeigt als "m3".

5. Starten Sie den Export nach Microsoft Excel über den Menübefehl Bericht > Erzeugen oder klicken Sie auf das Symbol .
   Dieser Schritt setzt eine Verbindung zum WinCC Server mit WinCC Professional in Runtime voraus.

Ergebnis

Das folgende Bild zeigt ein Beispiel für die Anzeige von Energiewerten über das Energy Suite Export Tool:

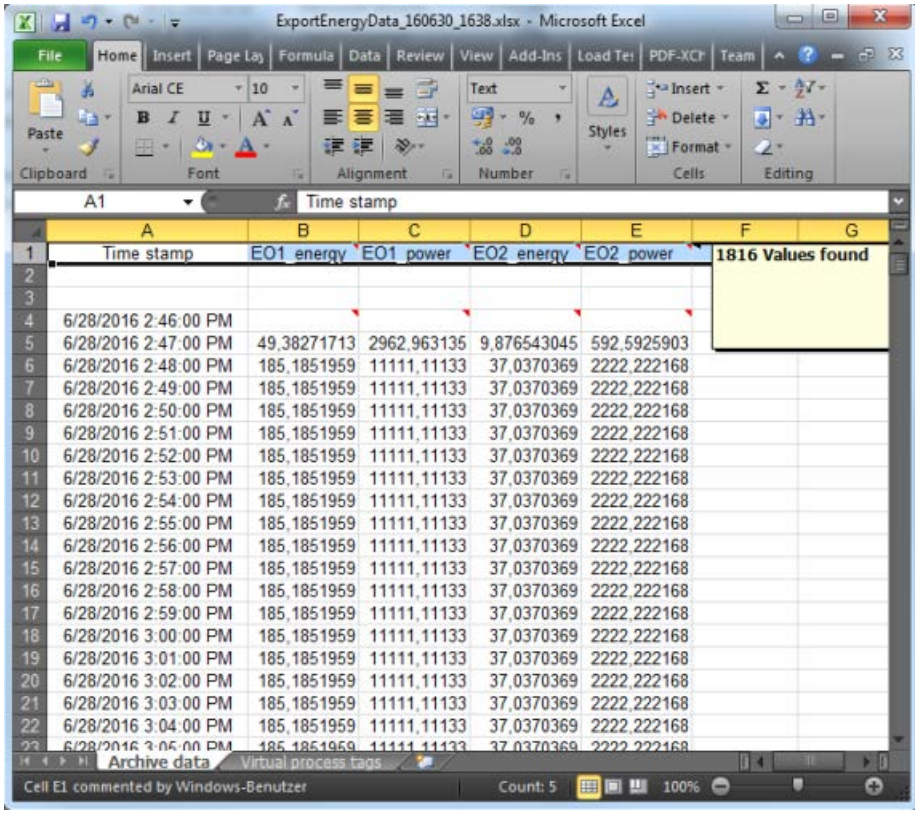

Time stamp	EO1_energy	EO1_power	EO2_energy	EO2_power
6/28/2016 2:46:00 PM				
6/28/2016 2:47:00 PM	49,38271713	2962,963135	9,876543045	592,5925903
6/28/2016 2:48:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:49:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:50:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:51:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:52:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:53:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:54:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:55:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:56:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:57:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:58:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 2:59:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:00:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:01:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:02:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:03:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:04:00 PM	185,1851959	11111,11133	37,0370369	2222,222168
6/28/2016 3:05:00 PM	185,1851959	11111,11133	37,0370369	2222,222168

Energy Suite Export Tool starten und automatischen Bericht erstellen

1. Wählen Sie den Menübefehl Start > SiemensAutomation > Options and Tools > SIMATIC Energy Suite Runtime Toolbox > SIMATIC Energy Suite Export Tool.
   Das Energy Suite Export Tool wird geöffnet.
2. Prüfen Sie den Namen des WinCC Servers:
   – Wenn WinCC Professional in Runtime ist, dann ist der Server-Name (Name des lokalen Runtime-PCs) automatisch eintragen.
   – Andernfalls können Sie über den Menübefehl Einstellungen > WinCC Server die Server-Verbindung manuell einstellen.
3. Wählen Sie die Oberflächensprache (Deutsch, Englisch) über den Menübefehl Einstellungen > Sprache.
4. Speichern Sie die Spracheinstellungen über den Menübefehl Datei > Einstellungen speichern.

5. Konfigurieren Sie die Vorlage für die Berichte über das Menü Bericht. Die folgende Tabelle zeigt alle möglichen Einstellungen. Die letzte Tabellenspalte zeigt die konkreten Werte für folgende Beispiele:
   – Beispiel 1: Automatischer Archivmesswerte-Bericht soll jeden Montag um 12:15 AM für die abgelaufene Woche, für den Zeitraum des Urlaubs vom 17.07.2017 bis 18.08.2017 erstellt werden.
   – Beispiel 2: Automatischer Archivmesswerte-Bericht soll am 2. jeden Monats um 09:00 AM für den letzten Monat, dauerhaft erstellt werden.

Parameter	Beschreibung	Wertebereich	Werte für Beispiel
Type (Art)	Art des Berichts	Archivmesswerte-Export Kostenstellen-Bericht	• Archivmesswerte-Export
Trigger (Ausführung)	Ausführung des Berichts	• Manuell • Automatisch	• Automatisch
Name (Name)	Name des Berichts	-	Für Beispiel 1: • WeeklyReport Für Beispiel 2: • MonthlyReport
File name (Dateiname)	Dateiname des Berichts, setzt sich zusammen aus dem Namen des Berichts, Datum und Uhrzeit der Berichtserstellung.	-	Für Beispiel 1: • WeeklyReport_170717_1215.xls Für Beispiel 2: • MonthlyReport_170702_0900.xls
Path (Pfad)	Ablageort des erstellten Berichts	-	• C:\Users\Administrator\Desktop\
Tariff set (Tarifsatz)	Auswahl des Tarifsatzes	-	Für Beispiel 1: - Für Beispiel 2: -
Unit (Einheit)	Einheit Energieverbrauch (automatisch aus Tarifsatz)	-	Für Beispiel 1: - Für Beispiel 2: -
Cost centers (Kostenstellen)	Auswahl der Kostenstelle	-	Für Beispiel 1: - Für Beispiel 2: -
Report cycle (Berichtszyklus)	Zyklus, in dem der Bericht erstellt wird	• Täglich • Wöchentlich • Monatlich • Jährlich	Für Beispiel 1: • Wöchentlich Für Beispiel 2: • Monatlich

Parameter	Beschreibung	Wertebereich	Werte für Beispiel
Time of day (Tageszeit)	Zeitpunkt, an dem der Bericht erstellt wird	• 12 h-Format • 24 h-Format	Für Beispiel 1: • 12:15:00 AM Für Beispiel 2: • 09:00:00 AM
Week day (Wochentag)	Wochentag, an dem der Bericht erstellt wird. Nur auswählbar, wenn Berichtszyklus = Wöchentlich.	Montag - Sonntag	Für Beispiel 1: • Montag Für Beispiel 2: -
Day (Tag)	Tag, an dem der Bericht erstellt wird. Nur auswählbar, wenn Berichtszyklus = Monatlich oder Jährlich.	1 - 28	Für Beispiel 1: - Für Beispiel 2: • 2
Month (Monat)	Monat, in dem der Bericht erstellt wird. Nur auswählbar, wenn Berichtszyklus = Jährlich.	Januar - Dezember	Für Beispiel 1: - Für Beispiel 2: -
Time period (Zeitbereich)	Zeitbereich, den der Bericht auswertet	Bei Berichtszyklus "Täglich": • Anfang, Ende • Letzter Tag • Abgelaufener Tag (00:00 - 24:00) Bei Berichtszyklus "Wöchentlich": • Letzte Woche • Abgelaufene Woche (Montag - Sonntag) Bei Berichtszyklus "Monatlich": • Letzter Monat • Abgelaufener Monat (1. - Ende des Monats) Bei Berichtszyklus "Jährlich": • Letztes Jahr • Abgelaufenes Jahr (Januar - Dezember)	Für Beispiel 1: • Abgelaufene Woche (Montag - Sonntag) Für Beispiel 2: • Letzter Monat
Valid period (Gültigkeitszeitraum)	Gesamter Zeitraum, für den die Berichtserstellung erlaubt und gültig ist	• Anfang: Aktuelles Datum (standardmäßig eingestellt) • Ende: 12/31/2099 12:00:00 AM (standardmäßig eingestellt)	Für Beispiel 1: • Anfang: 07/15/2017 12:15:00 AM Ende: 08/15/2017 12:15:00 AM Für Beispiel 2: • Standardmäßig eingestellte Werte

Hinweis

Die Einheit für Kubikmeter m^3 wird im Energy Suite Export Tool angezeigt als "m3".

6. Wenn die Vorlage für den Bericht komplett konfiguriert ist, klicken Sie auf "Fertigstellen". Ein Task wird automatisch zu Windows Task Scheduler hinzugefügt und in dem konfigurierten Berichtszyklus ausgeführt. Die Berichte werden in dem konfigurierten Ablagepfad abgelegt.

Tarifsätze anlegen

Um Tarifsätze anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Menübefehl "Einstellungen > Tarifsätze".
   Der Dialog "Tarifsätze verwalten" wird geöffnet.
2. Um einen neuen Tarifsatz anzulegen, klicken Sie in der Funktionsleiste auf das Symbol "Tarifsatz hinzufügen".
   Der Dialog "Tarifsatz hinzufügen" wird geöffnet.
3. Konfigurieren Sie die Vorlage für Tarifsätze. Folgende Einstellungen sind z. B. möglich:
   - Vorlagenname des Tarifsatzes
   - Einheit Preis pro Energieverbrauch
   - Zuordnung der Tarife zu Zeitbereichen und Wochentagen
   - Preis pro Energieverbrauch
   - Ausblenden von Feiertagen

   Folgende Einträge sind erforderlich:
   - Vorlagenname
   - Pro Wochentag eine Tarifauswahl
   - Tarif muss pro Wochentag kompletten Tag (Zeitbereich) abdecken
4. Klicken Sie auf die Schaltfläche "Hinzufügen".
   Der Tarifsatz wird der Liste im Dialog "Tarifsätze verwalten" hinzugefügt.

Verhalten im Fehlerfall

Hinweis

Bei unerwartetem Verhalten (z. B. WinCC Professional Datenbank nicht gestartet) werden die Fehlermeldungen in der Datei "Siemens.Simatic.EnergyMgmt.WinCCprof.DataExportTool.txt" gespeichert unter "C:\Program Files (x86)\Siemens\Automation\SCADA-RT_V11\WinCC\bin\config".

Energy Suite-Bausteine

A

A.1 Übersicht der Energy Suite-Bausteine

Hinweis

Beschreibung der Energy Suite-Bausteine für Experten

In den nachfolgenden Abschnitten finden Sie zu jedem Baustein die detaillierte Beschreibung zur Funktionsweise und zu den Bausteinparametern.

Für die Standard-Anwendung mit SIMATIC Energy Suite benötigen Sie diese Informationen nicht. SIMATIC Energy Suite generiert für Sie automatisch alle notwendigen Bausteine und deren Verschaltung auf Basis der konfigurierten Energieobjekte und der Energieprogrammeinstellungen.

⚠ VORSICHT

Manuelle Änderungen an Bausteinen des Energieprogramms

Nehmen Sie nach dem Generieren des Energieprogramms manuelle Änderungen an den Bausteinen und deren Verschaltung vor, dann sind Sie für die Fehlerfreiheit der Bausteine und des Energieprogramms selbst verantwortlich.

⚠ VORSICHT

Verwendung von Energy Suite-Bausteinen in anderen PLC-Programmen

Benutzen Sie die Energy Suite-Bausteine aus der globalen Bibliothek in anderen PLC-Programmen, dann sind Sie für die fehlerfreie Anwendung der Energy Suite-Bausteine selbst verantwortlich.

Dieser Abschnitt gibt einen Überblick über alle verfügbaren Energy Suite-Bausteine. Die Bausteine sind nach der Installation von SIMATIC Energy Suite unter "Globale Bibliotheken" in STEP 7 (TIA Portal) sichtbar:

- Funktionsbausteine (Seite 116)
- Datenbausteine (Seite 164)
- Datentypen (Seite 166) inkl. Enumerationen (Seite 179)

Funktionsweise der Energy Suite-Bausteine

Das folgende Bild zeigt die prinzipielle Funktionsweise der Energy Suite-Bausteine im Energieprogramm für Energiedaten aus dem Prozessabbild der CPU (PLC-Variablen) oder aus Datenbausteinen (DB-Variablen).

Der Platzhalter "xxx“ im Bild bedeutet, dass es verschiedene Typen von diesem Baustein gibt, z. B. für "EnS_Drvxxx":

- EnS_DrvPulse
- EnS_DrvCounterLInt

Weitere Typen finden Sie in der Tabelle "Funktionsbausteine" nach dem folgenden Bild.

Das folgende Bild zeigt die Funktionsweise der Energy Suite-Bausteine:

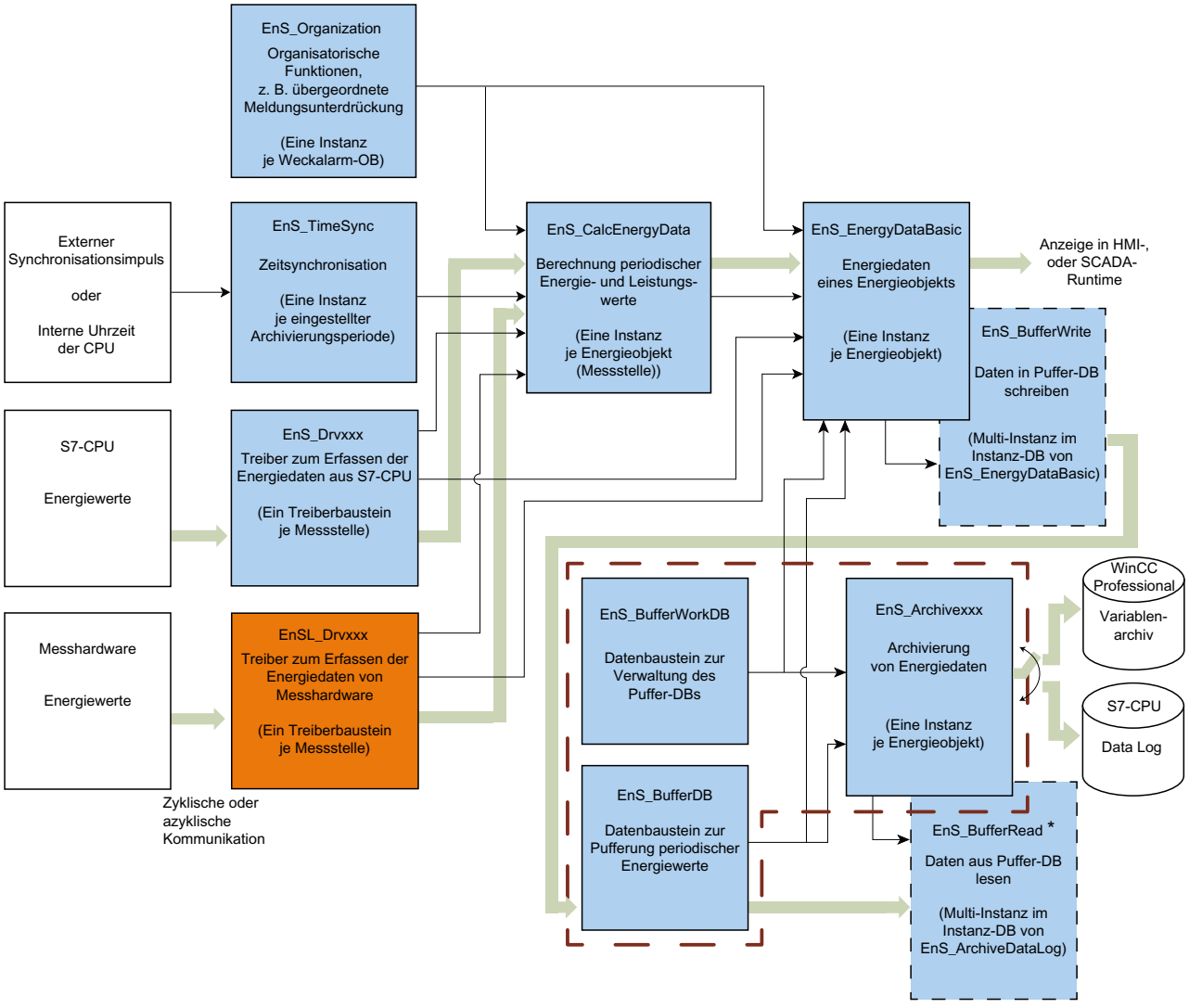

* EnS_BufferRead wird bei der Archivierung in WinCC Professional nicht aufgerufen

Energiedatenfluss

Parametrierdaten

Intern verwendete Bausteine

Einer Energieobjekttabelle zugehörigen Bausteine

Funktionsbausteine

Name	Funktion
EnS_ArchiveDataLog (Seite 116)	Archiviert Energiedaten in Data Logs (CSV-Datei auf der SIMATIC Memory Card der CPU)
EnS_ArchiveHMI (Seite 124)	Archiviert Energiedaten in Variablenarchiven von WinCC Professional
EnS_BufferRead (Seite 131)	Liest Daten aus dem Puffer-DB (EnS_BufferDB (Seite 164)). Wird intern verwendet
EnS_BufferWrite (Seite 133)	Schreibt Daten in den Puffer-DB (EnS_BufferDB (Seite 164)). Wird intern verwendet
EnS_CalcEnergyData (Seite 136)	Berechnet periodisch die verbrauchte Energie und durchschnittliche Leistungsaufnahme
EnS_DrvAnalogValue (Seite 142)	Treiberbaustein für analogen Eingang (z. B. Leistungswert, Durchflusswert)
EnS_DrvCounterLInt (Seite 144)	Treiberbaustein für Zählerwerte vom Typ LINT
EnS_DrvCounterLReal (Seite 146)	Treiberbaustein für Zählerwerte vom Typ LREAL
EnS_DrvPulse (Seite 148)	Treiberbaustein für einen Impulseingang
EnS_EnergyDataAdv (Seite 150)	Erweiterte Energiedaten eines Geräts (z. B. phasenspezifische Strom-, Spannungswerte)
EnS_EnergyDataAdvMinMax (Seite 151)	Minimal- und Maximalwerte der erweiterten Energiedaten eines Geräts
EnS_EnergyDataBasic (Seite 152)	Basisenergiedaten eines Geräts (z. B. Leistungs-, Energiezählwert)
EnS_Organization (Seite 156)	Enthält allgemeine Daten und Parameter zur zentralen Programmorganisation
EnS_TimeSync (Seite 157)	Synchronisiert die Zeiten für die Berechnung der Energiedaten

Datenbausteine

Name	Funktion
EnS_BufferDB (Seite 164) *	Datenbaustein zur Pufferung von Energiedatensätzen
EnS_BufferWorkDB (Seite 165) *	Datenbaustein für Kontroll- und Statusinformationen des Puffers

* Der Name kann geändert werden

Anwenderdefinierte Datentypen

Name	Funktion	Wird verwendet von
EnS_typeAnalogValue (Seite 166)	Datentyp für analoge Prozesswerte	EnS_CalcEnergyData EnS_DrvAnalogValue
EnS_typeBufferReadCtrl (Seite 166)	Datentyp für Kontrollinformationen des Puffers für das Lesen von Daten	EnS_BufferRead EnS_typeBufferWork
EnS_typeBufferReadStat (Seite 167)	Datentyp für Daten und Statusinformationen des Puffers für das Lesen von Daten	EnS_typeBufferWork

Name	Funktion	Wird verwendet von
EnS_typeBufferWork (Seite 167)	Datentyp für Kontroll- und Statusinformationen des Puffers	EnS_BufferRead EnS_BufferWrite EnS_ArchiveDataLog EnS_EnergyDataBasic EnS_BufferWorkDB EnS_ArchiveHMI
EnS_typeBufferWriteCtrl (Seite 168)	Datentyp für Kontrollinformationen des Puffers für das Schreiben von Daten	EnS_BufferWrite EnS_typeBufferWork
EnS_typeBufferWriteStat (Seite 169)	Datentyp für Statusinformationen des Puffers für das Schreiben von Daten	EnS_typeBufferWork
EnS_typeEnergyAdv (Seite 169)	Datentyp für erweiterte Energiedaten	EnS_EnergyDataAdv
EnS_typeEnergyBasic (Seite 171)	Datentyp für Basisenergiedaten	EnS_CalcEnergyData EnS_EnergyDataBasic
EnS_typeEnergyCounter (Seite 171)	Datentyp für Energiezählwert	EnS_CalcEnergyData EnS_DrvCounterLInt EnS_DrvCounterLReal EnS_DrvPulse
EnS_typeEnergyMeta (Seite 172)	Datentyp für Metadaten der Messstelle (Informationen zur Konfiguration des Energieobjekts)	EnS_CalcEnergyData EnS_BufferWrite EnS_DrvAnalogValue EnS_DrvCounterLInt EnS_DrvCounterLReal EnS_DrvPulse EnS_EnergyDataAdv EnS_EnergyDataBasic
EnS_typeHMIWriteCtrl (Seite 175)	Datentyp für Kontrollinformationen des Puffers für das Schreiben von Energiedaten in das Variablenarchiv von WinCC Professional	EnS_ArchiveHMI
EnS_typeHMIWriteStat (Seite 176)	Datentyp für Statusinformationen des Puffers für das Schreiben von Energiedaten in das Variablenarchiv von WinCC Professional	EnS_ArchiveHMI
EnS_typeOrgData (Seite 176)	Datentyp für Organisationsdaten	EnS_Organization EnS_TimeSync EnS_DrvAnalogValue EnS_DrvCounterLInt EnS_DrvCounterLReal EnS_DrvPulse EnS_ArchiveDataLog EnS_EnergyDataAdv EnS_EnergyDataBasic EnS_ArchiveHMI

Name	Funktion	Wird verwendet von
EnS_typeSync (Seite 177)	Datentyp für die Zeitsynchronisation	EnS_TimeSync EnS_CalcEnergyData
EnS_typeValue (Seite 178)	Datentyp für Ausgangswerte	EnS_BufferDB EnS_BufferRead EnS_ArchiveDataLog

A.2 Funktionsbausteine (FBs)

A.2.1 EnS_ArchiveDataLog: Energiewerte in Data Logs schreiben

A.2.1.1 Beschreibung EnS_ArchiveDataLog

Beschreibung

Der Funktionsbaustein "EnS_ArchiveDataLog" schreibt periodische Energiedaten aus dem Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)" (Parameter "bufferDB") in "Data Logs" auf die SIMATIC Memory Card der CPU. Das Schreiben erfolgt nur, wenn die Parameter "archiveEnergy", "archivePower", "archiveEnergyCounter" in den entsprechenden Metadaten (Seite 172) auf "TRUE" gesetzt sind. Die Daten werden auf der SIMATIC Memory Card als CSV-Datei gespeichert.

Um Zugriffskonflikte auf die SIMATIC Memory Card der CPU zu vermeiden, verwenden Sie je CPU nur eine Instanz des Funktionsbausteins.

Die folgende Tabelle zeigt den Aufbau einer CSV-Datei.

Spalte	Name	Beschreibung
1	SeqNo	Sequenzielle Nummer (Vorgabe durch das System)
2	Tagname	Variablenname
3	Value	Archivierter Wert
4	TimeStamp	Zeitstempel des archivierten Werts
5	Status	Status des archivierten Werts

Der Name der archivierten Daten wird zusammengesetzt aus <DataLogName> +<YYYYMMDD_hhmm (Zeitstempel des Erstellungszeitstempels im UTC-Zeitformat)>

Beispiel: DataLog_20141117_2315

Die Größe des Data Logs ist vom System auf 1 GB begrenzt. Die Anzahl der Datensätze muss beim Parameter "noRecords" entsprechend gesetzt werden (siehe folgendes Berechnungsbeispiel).

Beispiel: Der erforderliche Speicherplatz auf der Memory Card für ein Data Log wird über folgende Formel berechnet: 45 Byte + (Anzahl der Datensätze * 190 Byte).

Die Anzahl der Data Logs werden vom Parameter "noDataLogs" definiert. Die Größe der Memory Card begrenzt die Anzahl der verfügbaren Data Logs (gültige Anzahl: 1 - 100).

Schreiben der Data Logs

Die Archivierung der Energiedaten wird ausgeführt, wenn der Puffer-DB neue Daten enthält (bufferWork.noItems > 0). Der Funktionsbaustein liest anschließend die neuen Daten mit dem Funktionsbaustein "EnS_BufferRead (Seite 131)" und schreibt sie in das Data Log.

Hinweis

Abhängig von der verwendeten CPU, der verwendeten Memory Card und der Position der Daten auf der Memory Card kann die Schreibgeschwindigkeit variieren. Die Anzahl der Zyklen, die zum Schreiben eines Energiedatensatzes benötigt werden, ist variabel. Es kann nicht mehr als 1 Datensatz pro Zyklus geschrieben werden.

Anzahl der Data Logs

Wenn ein Data Log voll ist, wird der Ausgang "full" für einen Zyklus gesetzt und eine Meldung ausgegeben.

Wenn die maximale Anzahl von Data Logs ("noDataLogs", gültige Anzahl: 2-10) erreicht ist und auch das letzte Data Log voll ist, dann wird das älteste Data Log gelöscht und ein neues Data Log angelegt. Wenn ein Data Log gelöscht oder neu angelegt wird, dann wird eine Meldung erzeugt. Die Meldung sehen Sie z. B. auf einem SCADA-System.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 119)" enthält weitere Fehlerinformationen. Die Fehlerinformationen des untergeordneten Funktionsbausteins „EnS_BufferRead (Seite 131)“ werden zum Parameter "status" hinzugefügt. Die Fehlerinformation wird im Parameter "status" angezeigt, bis der Fehler beseitigt ist. Erst dann wird die Information des nächsten anstehenden Fehlers angezeigt.

A.2.1.2 Parameter EnS_ArchiveDataLog

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_ArchiveDataLog":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	R *
dataLogName	Input	STRING[32]	-	Name des Data Logs	-
noRecords	Input	UDINT	10000	Anzahl der Datensätze im Data Log	-
noDataLogs	Input	UINT	5	Anzahl der Data Logs	-
bufferDB	Input	DB_ANY	-	Zeiger auf Pufferrdatenbaustein	-
sD	Input	VARIANT	-	Begleitwert für Alarmmeldungen	-
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung akti-viert	-

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	R *
full	Output	BOOL	FALSE	TRUE = Data Log ist voll	-
busy	Output	BOOL	FALSE	TRUE = Bearbeitung ist aktiv	-
done	Output	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet	-
error	Output	BOOL	FALSE	TRUE = Fehler	-
status	Output	WORD	-	Fehlerstatusinformation (Seite 119)	-
statusCreateFct	Output	WORD	-	Fehlerstatusinformation der Anweisung DataLogCreate	-
statusOpenFct	Output	WORD	-	Fehlerstatusinformation der Anweisung DataLogOpen	-
statusWriteFct	Output	WORD	-	Fehlerstatusinformation der Anweisung DataLogWrite	-
statusDeleteFct	Output	WORD	-	Fehlerstatusinformation der Anweisung DataLogDelete	-
statusCloseFct	Output	WORD	-	Fehlerstatusinformation der Anweisung DataLogClose	-
dataLogNa-meAct	Output	STRING[48]	-	Name des aktiven Data Logs	-
dataLogName-Last	Output	STRING[48]	-	Name des letzten Data Logs	-
dataLogName-Del	Output	STRING[48]	-	Name des gelöschten Data Logs	-
errorAlarm1	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm1	-
statusAlarm1	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm1	-
alarmState1	Output	BYTE	-	Alarmstatus von Program_Alarm1	-
errorAlarm2	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm2	-
statusAlarm2	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm2	-
alarmState2	Output	BYTE	-	Alarmstatus von Program_Alarm2	-
errorAlarm3	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm3	-
statusAlarm3	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm3	-
alarmState3	Output	BYTE	-	Alarmstatus von Program_Alarm3	-
errorAlarm4	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm4	-
statusAlarm4	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm4	-
alarmState4	Output	BYTE	-	Alarmstatus von Program_Alarm4	-
errorAlarm5	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm5	-
statusAlarm5	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm5	-
alarmState5	Output	BYTE	-	Alarmstatus von Program_Alarm5	-
errorAlarm6	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm6	-
statusAlarm6	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm6	-
alarmState6	Output	BYTE	-	Alarmstatus von Program_Alarm6	-
id	InOut	DWORD	0	ID des aktuellen Data Logs	X

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	R *
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten	-
bufferWork	InOut	EnS_typeBuffer-Work (Seite 167)	-	Pufferkontroll- und Statusinformation	-

* Die markierten Parameter erhalten das Systemattribut "Retain" und werden im remanenten Speicherbereich der CPU abgelegt.

A.2.1.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" des Funktionsbausteins "EnS_ArchiveDataLog" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8511	Fehler beim Lesen aus dem Puffer	Archivierungsfehler	bufferWork.readStat.statusReadFct	Das Lesen wird so lange wiederholt, bis es erfolgreich ist. **Fehlerursache**: Falscher Datenbaustein ist angebunden oder Datenbaustein ist nicht vorhanden. **Vorgehen**: Korrigieren Sie das Programm.
8601	Fehler beim Anlegen des Data Logs	Fehler beim Erstellen	statusCreateFct	Der Fehler des SFC "DataLogCreate" wird in den Parameter "statusCreateFct" kopiert. Das Erstellen wird solange wiederholt, bis es erfolgreich ist. **Fehlerursachen**: • Der Name des Data Logs ist ungültig • Kein Speicherplatz **Vorgehen:** • Vergeben Sie für das Data Log einen gültigen Namen. • Weitere Informationen zu den gültigen Namen finden Sie im Abschnitt "Archivierung und Pufferung (S7-CPU) (Seite 56)". • Passen Sie die Speicherkapazität der verwendeten SIMATIC Memory Card an. • Entsperren Sie den Riegel auf der verwendeten SIMATIC Memory Card.

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8602	Fehler beim Öffnen des Data Logs	Fehler beim Öffnen	statusOpenFct	Das Öffnen des Data Logs wird so lange wiederholt, bis es erfolgreich ist. Fehlerursachen: • Data Log ist nicht vorhanden • Data Log ist bereits geöffnet Vorgehen: • Schliessen Sie das Data Log vor dem Öffnen. • Achten sie darauf, dass das Data Log bereits erstellt ist. • Entsperren Sie den Riegel auf der verwendeten SMATIC Memory Card.
8603	Fehler beim Schreiben des Data Logs	Fehler beim Schreiben	statusWriteFct	Das Schreiben in das Data Log wird solange wiederholt, bis es erfolgreich ist. Fehlerursachen: • Data Log ist nicht vorhanden • Kein Speicherplatz Vorgehen: • Achten Sie darauf, dass das Data Log bereits erstellt ist. • Entsperren Sie den Riegel auf der verwendeten SIMATIC Memory Card.
8604	Fehler beim Löschen des Data Logs	Fehler beim Löschen	statusDeleteFct	Die Data Log-Datei ist nicht gelöscht Fehlerursachen: • Data Log ist nicht vorhanden • Data Log ist schreibgeschützt • Kein Speicherplatz Vorgehen: • Schliessen Sie die Data Log-Datei. • Passen Sie die Speicherkapazität der verwendeten SIMATIC Memory Card an. • Entsperren sie den Riegel auf der verwendeten SIMATIC Memory Card.
8605	Fehler beim Schließen des Data Logs	Fehler beim Schließen	statusCloseFct	Das Schließen des Data Logs wird so lange wiederholt, bis es erfolgreich ist. Fehlerursachen: • Data Log ist nicht vorhanden • Data Log ist bereits geschlossen Vorgehen: • Öffnen Sie das Data Log vor dem Schließen. • Achten Sie darauf, dass das Data Log bereits erstellt ist. • Entsperren Sie den Riegel auf der verwendeten SIMATIC Memory Card.

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8611	Maximale Anzahl von Datensätzen zu hoch	Parametrier-fehler	noRecords	Die Größe des Data Logs ist vom system auf 1 GB begrenzt. Die Größe der SIMATIC Memory Card begrenzt die Anzahl der verfügbaren Data Logs. Fehlerursachen: • Die maximale Anzahl von Datensätzen ist überschritten. Vorgehen: • Passen Sie die Speicherkapazität der verwendeten SIMATIC Memory Card an (z. B. durch Löschen der alten Data Logs).
8612	Anzahl von Data Logs ungültig (<2 oder >10)	Parametrier-fehler	noDataLogs	Die Anzahl der Data Logs entspricht der Einstellung "Anzahl der Data Logs" im Bereich "Archivierung" für den Archivierungstyp "PLC (SIMATIC Memory Card)". Fehlerursachen: • Die Anzahl der Data Logs liegt außerhalb des Gültigkeitsbereichs. Vorgehen: • Geben Sie eine gültige Anzahl der Data Logs unter "CPU > Energieobjekte > Energieprogrammeinstellungen > Archivierung und Pufferung > Anzahl der Data Logs" ein.

Die zusätzlichen Meldetexte (Spalte "Beschreibung") sind in der Textliste "EnS_Alarms" des Meldebausteins "Program_Alarm2" beschrieben und werden mit den Fehlercodes in der Meldung ausgegeben.

Siehe auch

Beschreibung EnS_BufferRead (Seite 131)

A.2.1.4 Meldungen EnS_ArchiveDataLog

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Für jede Meldung ist eine Multiinstanz vom Datentyp "Program_Alarm" unter den Variablen "Static" des Funktionsbausteins "EnS_ArchiveDataLog" angelegt. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen, die der Funktionsbaustein "EnS_ArchiveDataLog" erzeugt:

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm1	Parametrierfehler	Mit Quittierung
Program_Alarm2	Archivierungsfehler: <Textlist: EnS_Alarms:#status>	Mit Quittierung
Program_Alarm3	Energie Log voll: <Tag:#dataLogNameLast >	Mit Quittierung
Program_Alarm4	Energie Log erzeugt: <Tag:#dataLogNameAct >	Mit Quittierung

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm5	Energie Log gelöscht: <Tag:#datalogNameDel >	Mit Quittierung
Program_Alarm6	Datenverlust von <Tag:#bufferWork.noDeletedItems" Datensätzen*	Mit Quittierung

* Der Parameter "bufferWork.noDeletedItems" enthält die Anzahl von Datensätzen, die bei Pufferüberlauf überschrieben werden.

Die zusätzlichen Meldetexte für "Program_Alarm2" sind in der Textliste "EnS_Alarms" beschrieben. Die Fehlercodes (status) werden im Meldetext ausgegeben und enthalten zusätzlichen Meldetext des jeweiligen Fehlercodes.

Die folgende Tabelle zeigt die Fehlercodes mit zusätzlichen Meldetexten des Meldebausteins "Program_Alarm2":

Meldebaustein	Fehlercode (W#16#...)	Meldetext	Parameter	Funktionsbaustein
Program_Alarm2	8511	Fehler beim Lesen aus dem Puffer	status	EnS_BufferRead (Seite 131)
	8601	Fehler beim Anlegen des Data Logs		EnS_ArchiveData-Log
	8602	Fehler beim Öffnen des Data Logs		
	8603	Fehler beim Schreiben des Data Logs		
	8604	Fehler beim Löschen des Data Logs		
	8605	Fehler beim Schließen des Data Logs		

Die Begleitwerte für die Meldungen sind folgendermaßen zugewiesen:

Meldebaustein	Begleitwert	Parameter	Beschreibung
Program_Alarm1	1	sD	Frei
	2	-	Nicht verwendet
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet

Meldebaustein	Begleitwert	Parameter	Beschreibung
Program_Alarm2	1	status	Fehlerstatusinformation
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm3	1	dataLogNameLast	Name des vollen Data Logs
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm4	1	dataLogNameAct	Name des erzeugten Data Logs
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm5	1	dataLogNameDel	Name des gelöschten Data Logs
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet

Meldebaustein	Begleitwert	Parameter	Beschreibung
Program_Alarm6	1	bufferWork.noDeletedI-tems	Anzahl der gelöschten Datensätze bei Pufferüberlauf
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet

Die Meldungen sind änderbar.

Der Eingangsparameter "SD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen.

Die Meldungen können mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.2 EnS_ArchiveHMI: Energiewerte in Variablenarchive schreiben

A.2.2.1 Beschreibung EnS_ArchiveHMI

Beschreibung

Der Funktionsbaustein (FB) "EnS_ArchiveHMI" schreibt periodische Energiedaten aus dem Datenbaustein (DB) zur Pufferung "EnS_BufferDB (Seite 164)" in das Variablenarchiv von WinCC Professional. Dazu fragt der FB "EnS_ArchiveHMI" zyklisch den DB zur Pufferung "EnS_BufferDB" auf vorhandene Datensätze mit Energiedaten ab. Bei vorhandenen Datensätzen sendet der Funktionsbaustein (FB) "EnS_ArchiveHMI" einen Auftrag an die Energy Suite WinCC Professional-Komponente, um die Energiedaten aus dem DB zur Pufferung auszulesen und in das Variablenarchiv zu schreiben.

Schreiben der Variablenarchive

Die Archivierung der Energiedaten in Variablenarchive erfolgt nur, wenn folgende Bedingungen erfüllt sind:

- Der DB zur Pufferung enthält neue Datensätze (bufferWork.noItems > 0).
- Der DB zur Pufferung ist verfügbar (bufferWork.busy = FALSE).
- Die HMI-Verbindung zu WinCC Professional ist aufgebaut.

Wenn die Bedingungen erfüllt sind, wird das Lesen von Daten aus dem DB zur Pufferung aktiv. Der FB "EnS_ArchiveHMI" sendet einen Auftrag zur Datenübertragung an WinCC

Professional. Der Auftrag am Parameter "request" vom Datentyp "EnS_typeHMIWriteCtrl (Seite 175)" enthält Lese-, Schreibzeiger und die Anzahl der Datensätze, die archiviert werden.

Anschließend liest die Energy Suite WinCC Professional-Komponente die Daten aus dem DB zur Pufferung und schreibt diese in das Variablenarchiv. Der Parameter "statusWrite" vom Datentyp "EnS_typeHMIWriteStat (Seite 176)" enthält die Anzahl geschriebener Datensätze und den aktualisierten Lesezeiger. Während des Schreibvorgangs aktualisiert die Energy Suite WinCC Professional-Komponente die Anzahl geschriebener Datensätze und den Schreibzeiger nach jedem geschriebenen Datensatz.

Hinweis

Einzelne Instanzen von "EnS_BufferWrite (Seite 133)", "EnS_BufferRead (Seite 131)" oder WinCC-Skript können auf den Datenbaustein zur Pufferung zugreifen. Nur ein Lesezugriff auf den Datenbaustein ist zur selben Zeit möglich.
Die Pufferkontroll- und Statusinformationen des globalen DB "EnS_BufferWorkDB (Seite 165)" sind am Parameter "bufferWork" bereitgestellt.

Zeitüberwachung

Die Verbindung zwischen der CPU und WinCC Professional wird mithilfe des Watchdog-Zählers überwacht. Der FB "EnS_ArchiveHMI" setzt den Watchdog-Zähler am Parameter "request.watchdog". Der Inhalt des Parameters "request.watchdog" wird in den Parameter "statusWrite.watchdog" kopiert.

- Wenn die Energy Suite-Komponente in WinCC Professional die Zeitüberwachung quittiert, dann inkrementiert der FB "EnS_ArchiveHMI" den Watchdog-Zähler.
- Wenn keine Quittierung innerhalb der Überwachungszeit erfolgt, wird der Fehlercode "16#8606" am Parameter "status" ausgegeben. Zusätzlich wird ein aktiver Auftrag zurückgesetzt und kein weiterer Auftrag gesendet, bis der Watchdog-Fehler beseitigt ist. Nach Beseitigung des Fehlers wird eine Meldung mit Informationen über den Datenverlust erzeugt.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 128)" enthält weitere Fehlerinformationen.

Wenn die Energy Suite WinCC Professional-Komponente auf den Auftrag des FB "EnS_ArchiveHMI" während der Überwachungszeit nicht antwortet und kein Watchdog-Fehler aufgetreten ist, wird der Fehlercode "16#8605" am Parameter "status (Seite 128)" ausgegeben. Nach der abgelaufenen Wartezeit und einer erfolgreichen Bearbeitung des Auftrags durch die Energy Suite WinCC Professional-Komponente, wird der Auftrag erneut gesendet.

A.2.2.2 WinCC Professional-Komponente

Beschreibung

Die Energy Suite WinCC Professional-Komponente schreibt Energiedaten in das Variablenarchiv von WinCC Professional.

Meldungen

Wenn Fehler auftreten, erzeugt die WinCC Professional-Komponente Systemmeldungen in WinCC Professional.

Die folgende Tabelle zeigt die Meldungen:

Meldetext	Meldeklasse
Archivierungsfehler <Text list: EnS_Alarms:#status>	System
Archivierung wird fortgesetzt	System
Möglicher Datenverlust	System

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, wird der Fehlercode mit dem zusätzlichen Meldetext in der Systemmeldung von WinCC Professional ausgegeben. Der zusätzliche Meldetext ist in der folgenden Tabelle unter "Beschreibung" angezeigt.

Desweiteren werden die Fehlercodes mit zusätzlichen Meldetexten am Parameter "statusWrite.status"/"statusHMI" der Instanz-DB des Funktionsbausteins "EnS_ArchiveHMI" in der CPU ausgegeben.

Die folgende Tabelle zeigt Fehlercodes der WinCC Professional-Komponente:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter
8621	Verbindung unterbrochen	Fehler der Zeitüberwachung	statusWrite.status
8622	Zugriff auf das Variablenarchiv verweigert	Fehler beim Schreiben	statusWrite.status
8623	Schreiben der Energiedaten in das Variablenarchiv abgebrochen	Fehler beim Schreiben	statusWrite.status
8624	Möglicher Datenverlust	Datenfehler	statusWrite.status
8625	Zugriff auf den Datenbaustein zur Pufferung abgebrochen	Datenfehler	statusWrite.status
8631	Variable nicht konfiguriert	Parametrierungsfehler	statusWrite.status

A.2.2.3 Parameter EnS_ArchiveHMI

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_ArchiveHMI":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	HMI *
statusWrite	Input	EnS_typeHMIWriteStat (Seite 176)	-	Statusinformation für Schreiben der Energiewerte in Variablenarchiv	-
monitoringTime	Input	TIME	T#5s	Überwachungszeit für Schreiben der Energiewerte in Variablenarchiv	-
monitoringTime-Watchdog	Input	TIME	T#10s	Überwachungszeit für Watchdog	-
waitTime	Input	TIME	T#1m	Wartezeit für die Wiederholung der Anfrage zum Schreiben der Energiedaten in Variablenarchiv	-
bufferDB	Input	DB_ANY	-	Zeiger auf Datenbaustein zur Pufferung	-
sD	Input	VARIANT	-	Begleitwert für Alarmmeldungen	-
messageSuppress	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert	-
request	Output	EnS_typeHMIWriteCtrl (Seite 175)	-	Auftrag zum Schreiben der Energiedaten in Variablenarchiv	X
busy	Output	BOOL	FALSE	TRUE = Schreiben der Energiedaten in Variablenarchiv ist aktiv	-
done	Output	BOOL	FALSE	TRUE = Schreiben der Energiedaten in Variablenarchiv war erfolgreich	-
error	Output	BOOL	FALSE	TRUE = Fehler	-
status	Output	WORD	-	Fehlerstatusinformation (Seite 128)	-
statusHMI	Output	WORD	-	Fehlerstatusinformation	-
errorAlarm1	Output	BOOL	FALSE	TRUE = Fehler "Program_Alarm1"	-
statusAlarm1	Output	WORD	-	Fehlerstatusinformation "Program_Alarm1"	-
alarmState1	Output	BYTE	-	Alarmstatus von "Program_Alarm1"	-
errorAlarm2	Output	BOOL	FALSE	TRUE = Fehler "Program_Alarm2"	-
statusAlarm2	Output	WORD	-	Fehlerstatusinformation "Program_Alarm2"	-
alarmState2	Output	BYTE	-	Alarmstatus von "Program_Alarm2"	-
errorAlarm3	Output	BOOL	FALSE	TRUE = Fehler "Program_Alarm3"	-
statusAlarm3	Output	WORD	-	Fehlerstatusinformation "Program_Alarm3"	-
alarmState3	Output	BYTE	-	Alarmstatus von "Program_Alarm3"	-
orgData	Output	EnS_typeOrgData (Seite 176)	-	Organisationsdaten	-
bufferWork	Output	EnS_typeBufferWork (Seite 167)	-	Pufferkontroll- und Statusinformation	-

A.2.2.4 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8603	Fehler beim Schreiben der Energiedaten in das Variablenarchiv	Fehler beim Schreiben	statusHMI	Interner Fehler **Fehlerursachen:** • HMI-Verbindung zu der S7-CPU ist unterbrochen • Data Log ist nicht vorhanden • Der Name des Data Logs ist ungültig **Vorgehen**: • Bei dem internen Fehler besteht kein Handlungsbedarf.
8606	Zeitüberschreitung beim Schreiben der Energiedaten in das Variablenarchiv	Fehler beim Schreiben	statusWrite.busy / statusWrite.done / statusWrite-error	Die Zeitdauer innerhalb derer die Energiedaten aus der S7-CPU in das Variablenarchiv übertragen werden, ist abgelaufen. **Fehlerursachen:** • Energy Suite SCADA-Komponente reagiert nicht auf die Anfrage zum Schreiben der Energiedaten innerhalb der Überwachungszeit. • HMI-Verbindung zu der S7-CPU ist unterbrochen • Datenverlust
8607	Watchdog-Fehler	Fehler beim Schreiben	statusWrite.error	Interner Fehler **Fehlerursachen**: Quittierung von der Energy Suite SCADA-Komponente ist nicht inerhalb der Überwachungszeit erfolgt. **Vorgehen**: • Bei dem internen Fehler besteht kein Handlungsbedarf.
8611	Überwachungszeit < 0 oder > 1 Stunde	Parametrierfehler	monitoringTime	Interner Fehler **Fehlerursachen:** • Überwachungszeit ist ungültig • Watchdog-Fehler ist nicht aufgetreten **Vorgehen:** • Bei dem internen Fehler besteht kein Handlungsbedarf.

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8612	Überwachungszeit für Watchdog < 0 oder > 1 Stunde	Parametrier-fehler	monitoringTime-Watchdog	Interner Fehler **Fehlerursachen:** • Überwachungszeit für Watchdog ist ungültig **Vorgehen:** • Bei dem internen Fehler besteht kein Handlungsbedarf.
8613	Wartezeit < 0 oder > 1 Stunde	Paramietrier-fehler	waitTime	Interner Fehler **Fehlerursachen:** • Wartezeit für die Wiederholung der Anfrage zum Schreiben der Energiedaten in das Variablenarchiv ist ungültig **Vorgehen:** • Bei dem internen Fehler besteht kein Handlungsbedarf.

Die zusätzlichen Meldetexte (Spalte "Beschreibung") sind in der Textliste "EnS_Alarms" des Meldebausteins "Program_Alarm2" beschrieben und werden mit den Fehlercodes in der Meldung ausgegeben.

A.2.2.5 Meldungen EnS_ArchiveHMI

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Für jede Meldung ist eine Multiinstanz vom Datentyp "Program_Alarm" unter den Variablen "Static" des Funktionsbausteins "EnS_ArchiveHMI" angelegt. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_ArchiveHMI":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm1	Parametrierfehler	Mit Quittierung
Program_Alarm2	Archivierungsfehler <Text list: EnS_Alarms:#status>	Mit Quittierung
Program_Alarm3	Datenverlust von <Tag:#bufferWork.noDeletedItems" Daten-sätzen *	Mit Quittierung

* Der Parameter "bufferWork.noDeletedItems" enthält die Anzahl der Datensätze, die beim Pufferüberlauf überschrieben wurden. Bei jedem Pufferüberlauf wird der Datenbaustein zur Pufferung geleert (bufferWork.noItems = 0).

Die zusätzlichen Meldetexte des Meldebausteins "Program_Alarm2" sind in der Textliste "EnS_Alarms" beschrieben. Die Fehlercodes (Status) werden im Meldetext ausgegeben und enthalten zusätzlichen Meldetext des jeweiligen Fehlercodes.

Die folgende Tabelle zeigt die Fehlercodes mit zusätzlichen Meldetexten des Meldebausteins "Program_Alarm2":

Meldebaustein	Fehlercode (W#16#...)*	Meldetext	Parameter	Funktionsbaustein
Program_Alarm2	8606	Zeitüberschreitung für Schreiben der Energiedaten in Variablenarchiv	status	EnS_ArchiveHMI
	8607	Verbindung zum Variablenarchiv verloren		EnS_ArchiveHMI

* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Anzeigeformate im Programmstatus umstellen".

Die folgende Tabelle zeigt die Meldebausteine mit zugewiesenen Begleitwerten:

Meldebaustein	Begleitwert	Parameter	Beschreibung
Program_Alarm1	1	sD	Frei
	2	-	Nicht verwendet
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm2	1	status	Fehlerstatusinformation
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm3	1	bufferWork.noDeletedItems	Anzahl der überschriebenen Datensätze beim Pufferüberlauf
	2	sD	Nicht verwendet
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet

Meldebaustein	Begleitwert	Parameter	Beschreibung
	9	-	Nicht verwendet
	10	-	Nicht verwendet

Die Meldungen sind änderbar.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen.

Die Meldungen können mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.3 EnS_BufferRead: Energiewerte aus Datenbausteinen lesen

A.2.3.1 Beschreibung EnS_BufferRead

Beschreibung

Der Funktionsbaustein "EnS_BufferRead" liest Daten vom Datenbaustein "EnS_BufferDB (Seite 164)".

Der Funktionsbaustein "EnS_BufferRead" wird intern vom Funktionsbaustein "EnS_ArchiveDataLog (Seite 116)" aufgerufen.

Das folgende Bild zeigt die Beschaltung der Parameter:

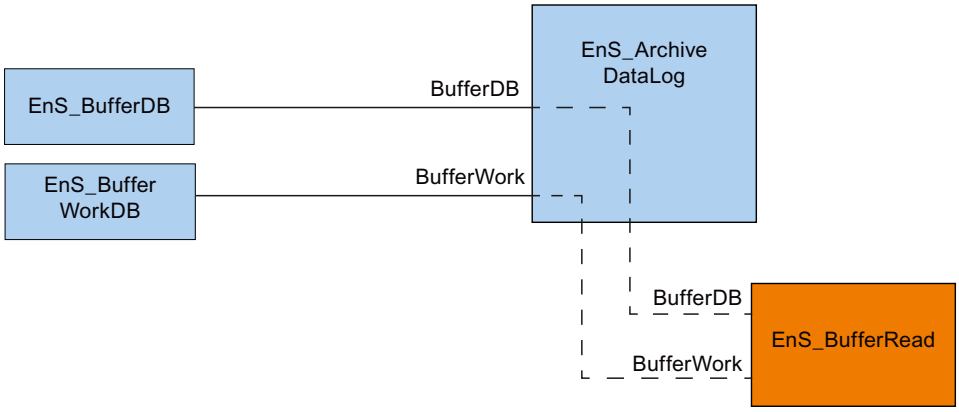

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 133)" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten. Meldungen werden vom aufrufenden Funktionsbaustein erzeugt.

Siehe auch

EnS_BufferWorkDB (Seite 165)

A.2.3.2 Parameter - EnS_BufferRead

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_BufferRead":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
bufferDB	Input	DB_ANY	-	Verweis zum Datenbaustein zur Pufferung
data	Output	EnS_typeValue (Seite 178)	-	Daten aus dem Puffer lesen
busy	Output	BOOL	FALSE	TRUE = Bearbeitung ist aktiv
done	Output	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 133)
bufferWork	InOut	EnS_typeBuffer-Work (Seite 167)	-	Pufferkontroll- und Statusinformation
request	InOut	EnS_typeBuffer-ReadCtrl (Seite 166)	-	Pufferkontrollinformation

A.2.3.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8511	Fehler beim Lesen aus dem Puffer	Fehler beim Lesen	bufferWork.readStat.statusReadFct	Das Lesen wird so lange wiederholt, bis es erfolgreich ist. Fehlerursachen: • Falscher Datenbaustein ist angebunden oder Datenbaustein ist nicht vorhanden. Vorgehen: • Korrigieren Sie das Programm.
8512	Datenverlust *		bufferWork.noDeletedItems	Fehlerursachen: • Datenverlust nach Pufferüberlauf. Ein Pufferüberlauf tritt auf, wenn der Puffer nicht geleert werden kann, weil z. B. die Verbindung zu WinCC (bei Archivierung in WinCC) unterbrochen ist. Vorgehen: • Stellen Sie die Verbindung zu WinCC her. • Passen Sie die Größe des Bausteins zur Pufferung an.

* Der Parameter "bufferWork.noDeletedItems" beinhaltet Datensätze, die bei Pufferüberlauf überschrieben werden. Die Anzahl der gelöschten Datensätze ist Teil des Alarms, der vom FB "EnS_ArchiveDataLog (Seite 116)" ausgegeben wird.

A.2.4 EnS_BufferWrite: Energiewerte in Datenbausteine schreiben

A.2.4.1 Beschreibung EnS_BufferWrite

Beschreibung

Der Funktionsbaustein "EnS_BufferWrite" schreibt Daten (periodische Energiedaten) in den Datenbaustein "EnS_BufferDB (Seite 164)".

Der Funktionsbaustein "EnS_BufferWrite" wird intern vom Baustein "EnS_EnergyDataBasic (Seite 152)" aufgerufen.

Für die Zugriffkontrolle ist "EnS_BufferWrite" mit "EnS_BufferWorkDB (Seite 165)" verknüpft (Parameter "bufferWork").

Das folgende Bild zeigt die Beschaltung der Parameter:

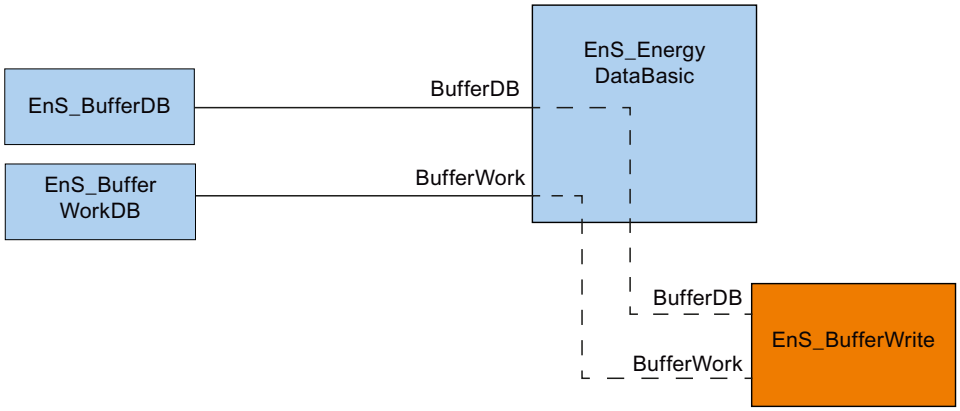

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 135)" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten. Meldungen werden vom aufrufenden Funktionsbaustein erzeugt.

A.2.4.2 Parameter EnS_BufferWrite

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_BufferWrite":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
bufferDB	Input	DB_ANY	-	Verweis zum Datenbaustein zur Pufferung
busy	Output	BOOL	FALSE	TRUE = Bearbeitung ist aktiv
done	Output	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet
error	Ouput	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 135)
bufferWork	InOut	EnS_typeBuffer-Work (Seite 167)	-	Pufferkontroll- und Statusinformation
request	InOut	EnS_typeBufferWri-teCtrl (Seite 168)	-	Pufferkontrollinformation
energyMeta	InOut	EnS_typeEnergy-Meta (Seite 172)	-	Messstellenspezifische Daten

A.2.4.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8501	Fehler beim Schreiben in den Puffer	Fehler beim Schreiben	bufferWork.writeStat.statusWriteFct	Das Schreiben wir solange wiederholt, bis es erfolgreich ist. Fehlerursache: • Falscher Datenbaustein ist angebunden oder der Datenbaustein ist nicht vorhanden. Vorgehen: • Korrigieren Sie das Programm.
8502	Alarm: Puffer bei 100 %, Datenverlust *		bufferWork.noItems / bufferWork.maxItem	Fehlerursache: • Datenverlust nach Pufferüberlauf • Ein Pufferüberlauf tritt auf, wenn der Puffer nicht geleert werden kann, weil z. B. die Verbindung zu WinCC (bei Archivierung in WinCC) unterbrochen ist. Vorgehen: • Verbindung zu WinCC herstellen (Archivierung bei WinCC) oder Fehler (z.B. Speicher voll) auf SIMATIC Memory Card beseitigen.
8503	Warnung: Pufferfüllstand über 80 %, bevorstehender Datenverlust		bufferWork.noItems / bufferWork.maxItem	Fehlerursache: • Die Anzahl der ARRAY-Elemente im Datenbaustein ist >= 0,8 * maximale Anzahl der ARRAY-Elemente. Vorgehen: • Erhöhen Sie in den Eigenschaften des Datenbausteins die ARRAY-Grenzen.
8504	Warnung: Pufferfüllstand über x %, bevorstehender Datenverlust **		bufferWork.noItems / bufferWork.maxItems	Fehlerursache: • Die Anzahl der ARRAY-Elemente im Datenbaustein ist < 0,8 * maximale Anzahl der ARRAY-Elemente. Vorgehen: • Erhöhen Sie in den Eigenschaften des Datenbausteins die ARRAY-Grenzen.

* Der Parameter "bufferWork.noDeletedItems" beinhaltet Datensätze, die bei Pufferüberlauf überschrieben werden. Die Anzahl der gelöschten Datensätze ist Teil des Alarms, der vom FB "EnS_ArchiveDataLog (Seite 116)" ausgegeben wird.

** Der Warnbereich wird unter "bufferWork.bufferLimit" konfiguriert.

A.2.5 EnS_CalcEnergyData: Energiewerte berechnen

A.2.5.1 Beschreibung EnS_CalcEnergyData

Beschreibung

Der Funktionsbaustein "EnS_CalcenergyBasic" berechnet den Energieverbrauch und die durchschnittliche Leistung über eine individuell konfigurierte Archivierungsperiode (z. B. 15 min).

Den Baustein können Sie auf folgende Arten beschalten:

- Leistungseingang
- Energieeingang
- Energie- und Leistungseingang

Bausteinstruktur bei verbundenem Leistungseingang

Die folgende Grafik zeigt die Bausteinstruktur bei verbundenem Leistungseingang:

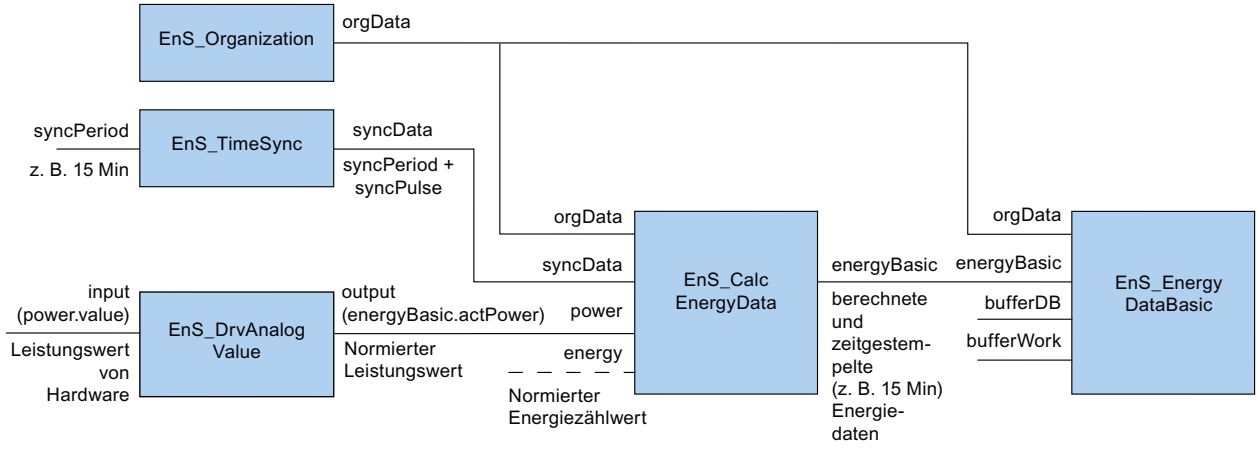

Aktuelle Leistung

"power.value" wird dem Leistungsausgang "energyBasic.actPower" übergeben.

Eine Totbereich-Funktion kann für den berechneten Leistungswert konfiguriert werden (Parameter "zeroCut“). Beträgt der Leistungswert weniger als der konfigurierte Wert, wird der berechnete Leistungswert auf 0 gesetzt. Deaktivieren Sie die Totbereich-Funktion indem Sie "zeroCut“ den Wert 0 zuweisen.

Aktuelle Energie

Bei Beginn einer neuen Archivierungsperiode wird der Energiewert "actEnergy" und die abgelaufene Zeit der Archivierungsperiode "syncData.syncPeriod" zurückgesetzt.

Mithilfe des aktuellen Leistungswerts wird der aktuelle Energieverbrauch "energyBasic.actEnergy“ während der Archivierungsperiode berechnet.

Der berechnete Energiewert wird zyklisch zu dem aktuellen Energiewert "energyBasic.actEnergy" addiert.

Energiezählwert

"energyBasic.actEnergyCounter" und "energyBasic.energyCounter" werden auf 0 gesetzt.

Bausteinstruktur bei verbundenem Energieeingang

Die folgende Grafik zeigt die Bausteinstruktur bei verbundenem Energieeingang.

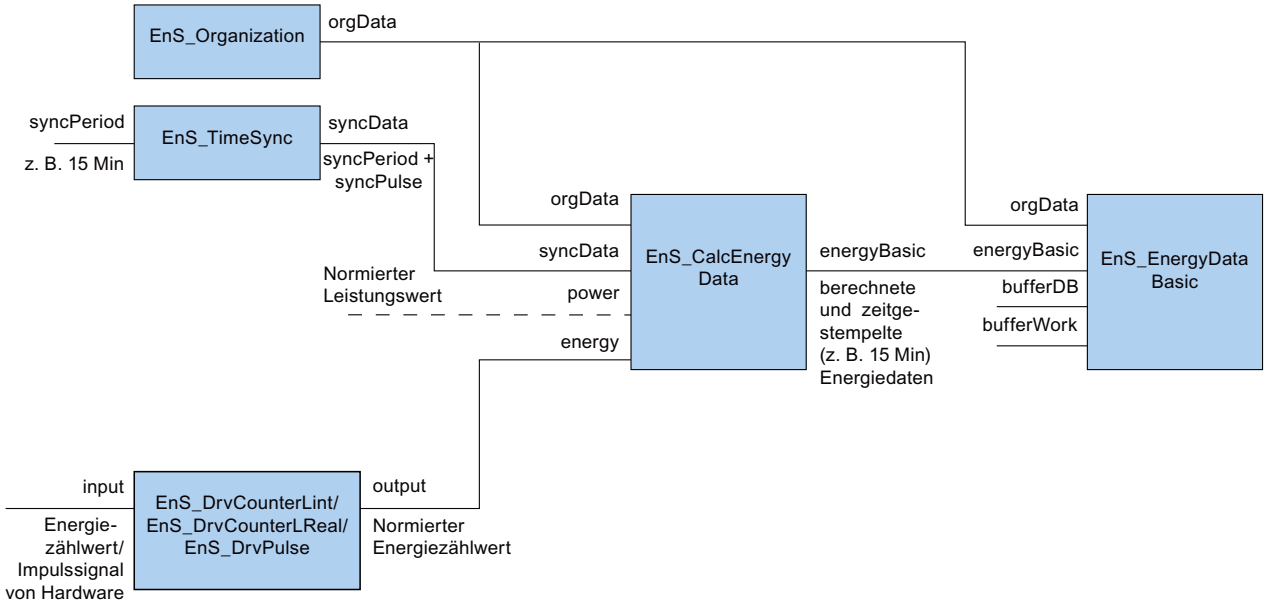

Aktuelle Energie

Der aktuelle Energieverbrauch "energyBasic.actEnergy“ wird berechnet, indem der aktuelle Energiezählwert "energy.value“ vom vorherigen Zählwert abgezogen wird.

Bei Beginn einer neuen Archivierungsperiode wird der aktuelle Energiewert "energyBasic.ectEnergy“ und die abgelaufene Zeit der Archivierungsperiode "syncData.syncPeriod“ zurückgesetzt.

Aktuelle Leistung

Die aktuelle Leistung "energyBasic.actPower" wird am Ende des Erfassungsintervalls ("baseAcqTime" + "addAcqTime") über den Energieverbrauch berechnet.

Erfassungsintervall:

Das Erfassungsintervall für die Leistungsberechnung wird über die Parameter "baseAcqTime" und "addAcqTime" definiert. "baseAcqTime" legt das minimale Erfassungsintervall fest. "addAcqTime" legt das zusätzliche Erfassungsintervall fest.

Zwischen Beginn und Ende des Erfassungsintervalls wird die Differenz des Zählerstandes ermittelt.

Um die aktuelle Leistung zu berechnen, wartet der Baustein mindestens das Standarderfassungsintervall "baseAcqTime". Die Berechnung der aktuellen Leistung "energyBasic.actPower" startet, nachdem eine der Voraussetzungen erfüllt ist:

- Der Zählwert hat sich geändert.
- Das zusätzliche Erfassungsintervall "addAcqTime" ist abgelaufen.
- Der Verbraucher wurde ausgeschaltet. In diesem Fall wird der Parameter "energy.consumerStatus" auf FALSE gesetzt.
  Diese Voraussetzung wird nur ausgewertet, wenn der Verbraucherstatus verfügbar ist ("energyMeta.consumerStatusEnable" = TRUE)

Energiezählwert

Der aktuelle Energiezählwert "energy.value" wird am Ende der Archivierungsperiode an "energyBasic.actEnergyCounter" und "energyBasic.energyCounter" kopiert.

Bausteinstruktur bei verbundenem Energie- und Leistungseingang

Die folgende Grafik zeigt die Bausteinstruktur bei verbundenem Energie- und Leistungseingang:

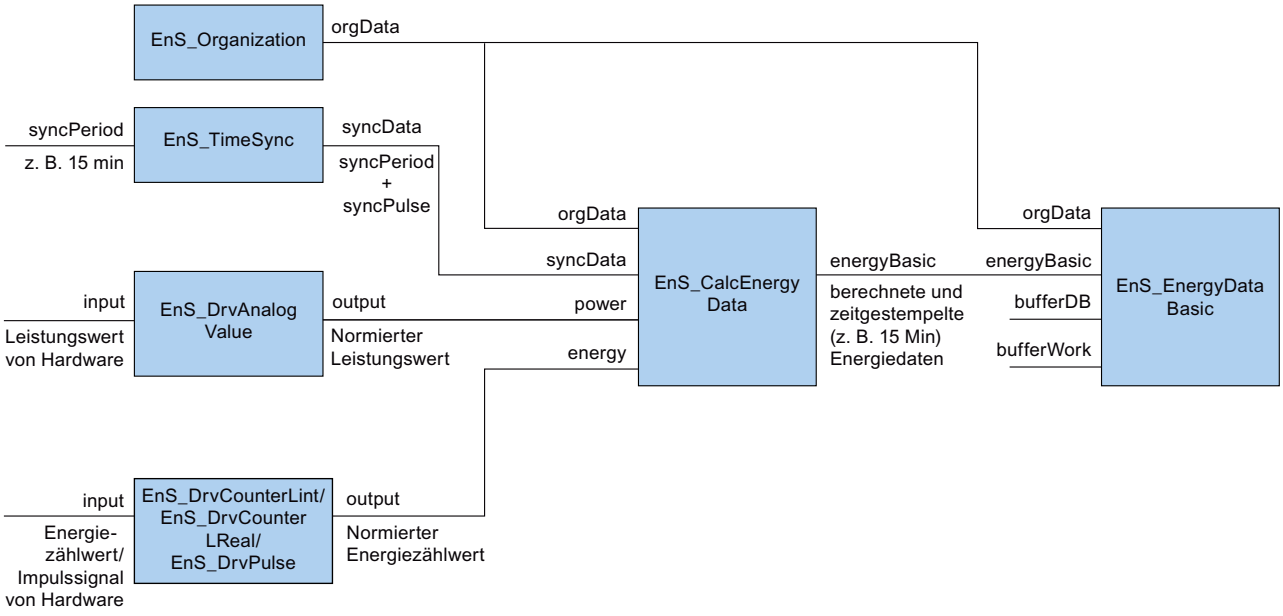

Aktuelle Leistung

"power.value" wird dem Leistungsausgang "energyBasic.actPower" übergeben.

Eine Totbereich-Funktion kann für den berechneten Leistungswert konfiguriert werden (Parameter „zeroCut“). Beträgt der Leistungswert weniger als der konfigurierte Wert wird der berechnete Leistungswert auf 0 gesetzt. Deaktivieren Sie die Totbereich-Funktion indem Sie „zeroCut“ den Wert 0 zuweisen.

Aktuelle Energie

Der aktuelle Energieverbrauch "energyBasic.actEnergy" wird berechnet, indem vom Anfangszählwert der aktuelle Zählwert "energy.value" abgezogen wird.

Bei Beginn der neuen Archivierungsperiode wird der aktuelle Energiewert "energyBasic.actEnergy" und die abgelaufene Zeit der Archivierunsgperiode "syncData.syncPeriod" zurückgesetzt.

Energiezählwert

Der aktuelle Energiezählwert "energy.value" wird am Ende der Archivierungsperiode an "energyBasic.actEnergyCounter" und "energyBasic.energyCounter" kopiert.

Durchschnittliche Leistung / verbrauchte Energie

Am Ende der Archivierungsperiode "syncData.syncPeriod" wird die durchschnittliche Leistungsaufnahme "energyBasic.power" aus der verbrauchten Energie "energyBasic.energy" berechnet.

Status

Die Berechnungen werden nur ausgeführt, wenn der Eingangswert gültig ist.

Der Status des Eingangswerts ist in den Datenstrukturen "energy" ("energy.status") und "power" ("power.status") beschrieben.

Wenn der Eingangswert ungültig ist, werden die entsprechenden Ausgangsparameter auf 16#FFFF_FFFF gesetzt. Für nachgeschaltete Bausteine ist der Status für weitere Protokollierung und Auswertungen verfügbar.

Aktuelle Werte mit dem Status 16#8301 sind gültig. Die Werte sind jedoch nicht geeignet zur Berechnung der verbrauchten Energie sowie der durchschnittlichen Leistungsaufnahme in der aktuellen Archivierungsperiode (z. B. 15 Minuten Periode). Die Werte für die durchschnittliche aktuelle Leistungsaufnahme und die verbrauchte Energie werden mit dem Status 18#8001 gekennzeichnet.

Anlauf

Im Anlauf werden alle Werte zurückgesetzt und die Zeiten neu gestartet.

Hinweis

Während einer unvollständigen Archivierungsperiode werden die Werte auf den Status "16#8301" (fehlerhafte Eingangswerte während der Archivierungsperiode) gesetzt.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 140)" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten.

A.2.5.2 Parameter EnS_CalcEnergyData

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_CalcEnergyData":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
power	Input	EnS_typeAnalogValue (Seite 166)	-	Eingangsdaten Leistung
energy	Input	EnS_typeEnergyCounter (Seite 171)	-	Eingangsdaten Energie
energyBasic	Output	EnS_typeEnergyBasic (Seite 171)	-	Basisenergiedaten
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 140)
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
syncData	InOut	EnS_typeSync (Seite 177)	-	Synchronisationsdaten
energyMeta	InOut	EnS_typeEnergyMeta (Seite 172)	-	Messstellenspezifische Daten

A.2.5.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8301	Fehlerhafte Eingangswerte während der Archivierungsperiode	Hardware-Fehler	energy.status / power.status	Interner Fehler **Fehlerursache:** • Fehler ist während Archivierungsperiode aufgetreten und ist bereits behoben. **Vorgehen**: • Bei dem internen Fehler besteht kein Handlungsbedarf.
8311	Verschaltungsfehler Berechnung	Verbindungsfehler	energy.status / power.status	**Fehlerursache:** • Die Parameter "energy" und "power" sind nicht verschaltet. **Vorgehen:** • Programmieren Sie die Verschaltung.

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8312	Berechnungsintervall "energyMeta.baseAcqTime" < 0	Parametrierfehler	energyMeta.baseAcqTime	**Fehlerursache:** • Die Dauer des Berechnungsintervalls ist ungültig. **Vorgehen:** • Verwenden Sie für die Dauer des Berechnungsintervalls gültige Werte: 0 - 3600
8313	Verlängerung des Berechnungsintervalls "energyMeta.addAcqTime" < 0		energyMeta.baseAcqTime / energyMeta.addAcqTime	**Fehlerursache:** • Die Dauer der Verlängerung des Berechnungsintervalls ist ungültig. **Vorgehen:** • Verwenden Sie für die Dauer der Verlängerung des Berechnungsintervalls gültige Werte: 0 - 3600
8314	Begrenzung für Nullpunkt "energyMeta.zeroCut" < 0.0		energyMeta.zeroCut	**Fehlerursache**: • Der Wert der Begrenzung für den Nullpunkt ist ungültig. **Vorgehen:** • Verwenden Sie für den Wert der Begrenzung für den Nullpunkt gültige Werte: 0.0 - 3.402823e+38.
8315	"energy.value" >= "energyMeta.overflowCntValue"		energy.value / energyMeta.overflowCntValue	**Fehlerursache:** • Der Energiezählwert am Parameter "energy.value" ist größer oder gleich dem Überlaufwert am Parameter "energyMeta.overflowCntValue". • Der Parameter "overflowCntValue" ist nicht richtig konfiguriert (zu klein). **Vorgehen:** • Korrigieren Sie die Konfiguration. • Vergrößern Sie den Überlaufwert.
8316	Normierungsfaktor "energyMeta.normFactorOut" ≤ 0.0 oder > 9999.0	Parametrierfehler	energyMeta.normFactorOut	**Fehlerursache**: • Der Normierungsfaktor am Parameter "energyMeta.normFactorOut" ist ungültig. **Vorgehen**: • Verwenden Sie für den Wert des Normierungsfaktors gültige Werte: 0.0 - 9999.0.

A.2.6 EnS_DrvAnalogValue: Leistungswerte vom Typ REAL erfassen

A.2.6.1 Beschreibung EnS_DrvAnalogValue

Beschreibung

Mit dem Funktionsbaustein "EnS_DrvAnalogValue" erfassen Sie einen Leistungswert. Das Eingangssignal wird normiert.

Der Ausgangswert wird in der Struktur "Output" an den Funktionsbaustein "EnS_CalcEnergyData (Seite 136)" übergeben.

Wird elektrische Energie gemessen, besitzt der ausgegebene Wert die Einheit "Wh". Wird nicht elektrische Meßgröße wie z. B. der Durchfluss gemessen, besitzt der ausgegebene Wert z. B. die Einheit "m³/h".

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 143)" enthält weitere Fehlerinformationen.

A.2.6.2 Parameter EnS_DrvAnalogValue

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_DrvAnalogValue":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
input	Input	REAL	0.0	Analoges Eingangssignal
normFactor	Input	REAL	1.0	Normierungsfaktor
hWFault	Input	BOOL	FALSE	TRUE = Hardware-Fehler des Eingangssignals
sD	Input	VARIANT	-	Begleitwert für Alarmmeldung
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert
output	Output	EnS_typeAnalogVa-lue (Seite 166)	-	Normiertes Ausgangssignal
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 143)
errorAlarm	Output	BOOL	FALSE	TRUE = Fehler des Meldebausteins "Pro-gram_Alarm"
statusAlarm	Output	WORD	-	Fehlerstatusinformation des Meldebausteins "Program_Alarm"
alarmState	Output	BYTE	-	Alarmstatus des Meldebausteins "Pro-gram_Alarm"

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
energyMeta	InOut	EnS_typeEnergy-Meta (Seite 172)	-	Messstellenspezifische Daten

A.2.6.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8001	Hardware-Fehler des Messgeräts	Hardware-Fehler	hWFault	**Fehlerursachen:** • Fehlerhafter Anschluss des Messgeräts • Fehlerhafte Adresse **Vorgehen:** • Überprüfen Sie den Anschluss des Messgeräts bzw. korrigieren Sie die Adresse.
8011	Normierungsfaktor <= 0.0 oder > 9999999.0	Parametrier-fehler	normFactor	**Fehlerursachen**: • Ungültiger Wert an der Einstellung "Normierungsfaktor" **Vorgehen**: • Geben Sie einen gültigen Wert (0.0 - 9999999.0) an der Einstellung "Normierungsfaktor" ein.

A.2.6.4 Meldungen EnS_DrvAnalogValue

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_DrvAnalogValue":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm	Hardware-Fehler	mit Quittierung

Die Meldung ist änderbar.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen sollen.

Die Meldung kann mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.7 EnS_DrvCounterLInt: Energiezählwerte vom Typ LINT erfassen

A.2.7.1 Beschreibung EnS_DrvCounterLInt

Beschreibung

Mit dem Funktionsbaustein "EnS_DrvCounterLInt" erfassen Sie einen ganzzahligen Energiezählerwert vom Typ LInt (64 Bit). Das Eingangssignal wird in einen normierten Energiezählwert konvertiert.

Der Ausgangswert wird vom Funktionsbaustein "EnS_CalcEnergyData (Seite 136)" als Energiezählwert verarbeitet.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 145)" enthält weitere Fehlerinformationen.

A.2.7.2 Parameter EnS_DrvCounterLInt

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_DrvCounterLInt":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
input	Input	LINT	0.0	Eingangszählwert
consumerStatus	Input	BOOL	FALSE	Verbraucherstatus-Bit: TRUE = Ein, FALSE = Aus
hWFault	Input	BOOL	FALSE	TRUE = Hardware-Fehler des Eingangssignals
sD	Input	VARIANT	-	Begleitwert für Alarmmeldung
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert
output	Output	EnS_typeEnergy-Counter (Seite 171)	-	Ausgangszählwert
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 145)
errorAlarm	Output	BOOL	FALSE	TRUE = Fehler des Meldebausteins "Pro-gram_Alarm"
statusAlarm	Output	WORD	-	Fehlerstatusinformation des Meldebausteins "Program_Alarm"
alarmState	Output	BYTE	-	Alarmstatus des Meldebausteins "Pro-gram_Alarm"

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
energyMeta	InOut	EnS_typeEnergy-Meta (Seite 172)	-	Messstellenspezifische Daten

A.2.7.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8001	Hardware-Fehler des Eingangsgeräts	Hardware-Fehler	hWFault	**Fehlerursachen:** • fehlerhafter Anschluss des Messgeräts • fehlerhafte Adresse **Vorgehen:** • Überprüfen Sie den Anschluss des Messgeräts
8011	Normierungsfaktor <= 0.0 oder > 9999999.0	Parametrier-fehler	energyMe-ta.normFactorIn	**Fehlerursachen:** • Ungültiger Wert an der Einstellung "Normierungsfaktor" **Vorgehen**: • Geben Sie einen gültigen Wert (0.0 - 9999999.0) an der Einstellung "Normierungsfaktor" ein.
8012	Zählerüberlaufwert < 1.0 oder > 1.0e+21		energyMe-ta.overflowCnt-Value	**Fehlerursachen:** • Ungültiger Wert an der Einstellung "Überlaufwert" **Vorgehen**: • Geben Sie einen gültigen Wert (1.0 - 1.0e+21) an der benutzerdefinierten Einstellung "Überlaufwert" ein.

A.2.7.4 Meldungen EnS_DrvCounterLInt

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_DrvCounterLInt":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm	Hardware-Fehler	Mit Quittierung

Die Meldung ist änderbar.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen sollen.

Die Meldung kann mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.8 EnS_DrvCounterLReal: Enegiezählwerte vom Typ LREAL erfassen

A.2.8.1 Beschreibung EnS_DrvCounterLReal

Beschreibung

Mit dem Funktionsbaustein "EnS_DrvCounterLInt" erfassen Sie einen Gleitpunkt-Energiezählwert vom Typ LREAL (64 Bit). Das Eingangssignal wird in einen normierten Energiewert konvertiert.

Der Ausgangswert wird vom Funktionsbaustein "EnS_CalcEnergyData (Seite 136)" als Energiezählwert verarbeitet.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 147)" enthält weitere Fehlerinformationen.

A.2.8.2 Parameter EnS_DrvCounterLReal

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_DrvCounterLReal":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
input	Input	LREAL	0.0	Eingangszählwert
consumerStatus	Input	BOOL	FALSE	Anwender Status-Bit: 1 = On, 0 = Off
hWFault	Input	BOOL	FALSE	FALSE = Hardware-Fehler des Eingangssignals
sD	Input	VARIANT	-	Begleitwert für Alarmmeldung
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert
output	Output	EnS_typeEnergy-Counter (Seite 171)	-	Ausgangszählwerte
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 147)
errorAlarm	Output	BOOL	FALSE	TRUE = Fehler des Meldebausteins "Pro-gram_Alarm"

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
statusAlarm	Output	WORD	-	Fehlerstatusinformation des Meldebausteins "Program_Alarm"
alarmState	Output	BYTE	-	Alarmstatus des Meldebausteins "Program_Alarm"
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
energyMeta	InOut	EnS_typeEnergyMeta (Seite 172)	-	Messstellenspezifische Daten

A.2.8.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8001	Hardware-Fehler des Messgeräts	Hardware-Fehler	hWFault	**Fehlerursachen:** • Fehlerhafter Anschluss des Messgeräts • Fehlerhafte Adresse **Vorgehen:** • Überprüfen Sie den Anschluss des Messgeräts
8011	Normierungsfaktor <= 0.0 oder > 9999.0	Parametrierfehler	energyMeta.normFactorIn	**Fehlerursachen**: • Ungültiger Wert an der Einstellung "Normierungsfaktor" **Vorgehen**: • Geben Sie einen gültigen Wert (0.0 - 9999.0) an der Einstellung "Normierungsfaktor" ein.
8012	Zählerüberlaufwert < 1.0 oder > 1.0e+21		energyMeta.overflowCntValue	**Fehlerursachen:** • Ungültiger Wert an der Einstellung "Überlaufwert" **Vorgehen**: • Geben Sie einen gültigen Wert (1.0 - 1.0e+21) an der benutzerdefinierten Einstellung "Überlaufwert" ein.

A.2.8.4 Meldungen EnS_DrvCounterLReal

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_DrvCounterLReal":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm	Hardware-Fehler	Mit Quittierung

Die Meldung ist änderbar.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen.

Die Meldung kann mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Oganization (Seite 156)" unterdrückt werden.

A.2.9 EnS_DrvPulse: Impulssignale erfassen

A.2.9.1 Beschreibung EnS_DrvPulse

Beschreibung

Der Funktionsbaustein "EnS_DrvPulse" addiert die erfassten Impulse, verrechnet sie mit dem konfigurierten Normierungsfaktor und berechnet einen Energiezählerwert vom Typ LREAL. Der Ausgangswert wird von "EnS_CalcEnergyData (Seite 136)" als normierter Energiezählerwert verarbeitet.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 149)" enthält weitere Fehlerinformationen.

A.2.9.2 Parameter EnS_DrvPulse

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_DrvPulse":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
input	Input	BOOL	FALSE	Impulseingang
consumerStatus	Input	BOOL	FALSE	Anwender Status-Bit: TRUE = Ein, FALSE = Aus
hWFault	Input	BOOL	FALSE	TRUE = Hardware-Fehler des Eingangssignals
sD	Input	VARIANT	-	Begleitwert für Alarmmeldung
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert
output	Output	EnS_typeEnergy-Counter (Seite 171)	-	Ausgangszählwerte
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 149)
errorAlarm	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm
statusAlarm	Output	WORD	-	Fehlerstatusinformation des Meldebausteins "Program_Alarm"

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
alarmState	Output	BYTE	-	Alarmstatus des Meldebausteins "Program_Alarm"
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
energyMeta	InOut	EnS_typeEnergyMeta (Seite 172)	-	Messstellenspezifische Daten

A.2.9.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8001	Hardware-Fehler des Messgeräts	Hardware-Fehler	hWFault	**Fehlerursachen:** • Fehlerhafter Anschluss des Messgeräts • Fehlerhafte Adresse **Vorgehen:** • Überprüfen Sie den Anschluss des Messgeräts bzw. korrigieren Sie die Adresse.
8011	Normierungsfaktor <= 0.0 oder > 9999999.0	Parametrierfehler	energyMeta.normFactorIn	**Fehlerursachen**: • Ungültiger Wert an der Einstellung "Normierungsfaktor" **Vorgehen**: • Geben Sie einen gültigen Wert (0.0 - 9999999.0) an der Einstellung "Normierungsfaktor" ein.
8012	Zählerüberlaufwert < 1.0 oder > 1.0e+21		energyMeta.overflowCntValue	**Fehlerursachen:** • Ungültiger Wert an der Einstellung "Überlaufwert" **Vorgehen**: • Geben Sie einen gültigen Wert (1.0 - 1.0e+21) an der benutzerdefinierten Einstellung "Überlaufwert" ein.

A.2.9.4 Meldungen EnS_DrvPulse

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_DrvPulse":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm	Hardware-Fehler	Mit Quittierung

Die Meldung kann vom Benutzer geändert werden.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen sollen.

Die Meldung kann mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.10 EnS_EnergyDataAdv: Erweiterte Energiewerte bereitstellen

A.2.10.1 Beschreibung EnS_EnergyDataAdv

Beschreibung

Der Funktionsbaustein "EnS_EnergyDataAdv“ stellt von einer Messstelle gesammelt einen erweiterten Energiedatensatz zur Verfügung. Der Funktionsbaustein dient als zentrale Schnittstelle zu WinCC Professional oder anderen Programmen (z. B. SIMATIC Energy Manager Pro).

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 151)" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten.

A.2.10.2 Parameter EnS_EnergyDataAdv

Die folgende Tabelle zeigt die Bausteinparameter des Bausteins "EnS_EnergyDataAdv":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	HMI *
advData	Input	EnS_typeEnergyAdv (Seite 169)	-	Erweiterter Messdatensatz	X
error	Output	BOOL	-	TRUE = Fehler	-
status	Output	WORD	-	Fehlerstatusinformation (Seite 151)	-
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten	-
energyMeta	InOut	EnS_typeEnergyMeta (Seite 172)	-	Messstellenspezifische Daten	-

* Die markierten Parameter erhalten das Systemattribut "Erreichbar für HMI" und "Sichtbar in HMI". Sie werden zur Bedien- und Beobachtungsfunktion im HMI-Gerät verwendet, z. B. um einen aktuellen Wert in einem Bildbaustein anzuzeigen.

A.2.10.3 Parameter Status

Der Ausgangsparameter "error" wird gesetzt, wenn in einem untergeordneten Funktionsbaustein ein Fehler auftritt.

Der Parameter "status" enthält zusätzliche Fehlerinformationen. Die Fehlerinformation wird im Parameter "status" angezeigt, bis der Fehler beseitigt ist. Erst dann wird die Information des nächsten anstehenden Fehlers angezeigt.

A.2.11 EnS_EnergyDataAdvMinMax: Erweiterte Minimal- und Maximalwerte bereitstellen

A.2.11.1 Beschreibung EnS_EnergyDataAdvMinMax

Beschreibung

Der Funktionsbaustein "EnS_EnergyDataAdvMinMax" stellt Minimal- und Maximalwerte der erweiterten Energiedaten für die Ausgabe z. B. in WinCC Professional bereit.

Anlauf

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Die Fehlerinformationen des untergeordneten Funktionsbausteins (z. B. von Messhardware) werden am Ausgang "status (Seite 152)" ausgegeben. Die Fehlerinformation wird im Parameter "status" angezeigt, bis der Fehler beseitigt ist. Erst dann wird die Information des nächsten anstehenden Fehlers angezeigt.

Meldungen

Der Baustein besitzt kein Meldeverhalten.

A.2.11.2 Parameter EnS_EnergyDataAdvMinMax

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_EnergyDataAdvMinMax":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	HMI *
advDataMin	Input	EnS_typeEnergyAdv (Seite 169)	-	Datensatz mit Minimalwerten der erweiterten Energiewerte	X
advDataMax	Input	EnS_typeEnergyAdv (Seite 169)	-	Datensatz mit Maximalwerten der erweiterten Energiewerte	X
error	Output	BOOL	-	TRUE = Fehler	
status	Output	WORD	-	Fehlerstatusinformation (Seite 152)	
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten	
energyMeta	InOut	EnS_typeEnergyMeta (Seite 172)	-	Messstellenspezifische Daten	

* Die markierten Parameter erhalten das Systemattribut "Erreichbar für HMI" und "Sichtbar in HMI". Sie werden zur Bedien- und Beobachtungsfunktion im HMI-Gerät verwendet, z. B. um einen aktuellen Wert im Bildbaustein anzuzeigen.

A.2.11.3 Parameter Status

Der Ausgangsparameter "error" wird gesetzt, wenn in einem untergeordneten Funktionsbaustein ein Fehler auftritt.

Der Parameter "status" enthält zusätzliche Fehlerinformationen. Die Fehlerinformation wird im Parameter "status" angezeigt, bis der Fehler beseitigt ist. Erst dann wird die Information des nächsten anstehenden Fehlers angezeigt.

A.2.12 EnS_EnergyDataBasic: Metadaten und Energiewerte bereitstellen

A.2.12.1 Beschreibung EnS_EnergyDataBasic

Beschreibung

Der Funktionsbaustein "EnS_EnergyDataBasic“ stellt von einer Messstelle gesammelte Meta- und Energiedaten zur Verfügung. Der Funktionsbaustein dient als zentrale Schnittstelle zu WinCC Professional oder anderen Programmen (z. B. SIMATIC Energy Manager Pro).

Metadaten

Die Metadaten am Parameter "energyMeta" vom Datentyp "EnS_typeEnergyMeta (Seite 172)" enthalten allgemeine Informationen über das Energieobjekt bzw. die Messhardware.

Anlaufen

Der Baustein besitzt kein Anlaufverhalten.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 154)" enthält weitere Fehlerinformationen.

A.2.12.2 Parameter EnS_EnergyDataBasic

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_EnergyDataBasic":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	HMI *
energyMeta	Input	EnS_typeEnergy-Meta (Seite 172)	-	Messstellenspezifische Daten	X
energyBasic	Input	EnS_typeEnergyBa-sic (Seite 171)	-	Basisenergiedaten	X
bufferDB	Input	DB_ANY	-	Zeiger auf den Datenbaustein zur Puf-ferung	-
sD	Input	VARIANT	-	Begleitwert für Alarmmeldungen	-
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung akti-viert	X
error	Output	BOOL	FALSE	TRUE = Fehler	-
status	Output	WORD	-	Fehlerstatusinformation (Seite 154)	-
status1	Output	WORD	-	Fehlerstatusinformation 1	-
status2	Output	WORD	-	Fehlerstatusinformation 2	-
status3	Output	WORD	-	Fehlerstatusinformation 3	-
status4	Output	WORD	-	Fehlerstatusinformation 4	-
status5	Output	WORD	-	Fehlerstatusinformation 5	-
errorAlarm1	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm1	-
statusAlarm1	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm1	-
alarmState1	Output	BYTE	-	Alarmstatus Program_Alarm1	-
errorAlarm2	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm2	-
statusAlarm2	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm2	-
alarmState2	Output	BYTE	-	Alarmstatus Program_Alarm2	-
errorAlarm3	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm3	-
statusAlarm3	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm3	-
alarmState3	Output	BYTE	-	Alarmstatus Program_Alarm3	-
errorAlarm4	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm4	-
statusAlarm4	Output	WORD	-	Fehlerstatusinformation Pro-gram_Alarm4	-
alarmState4	Output	BYTE	-	Alarmstatus Program_Alarm4	-

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung	HMI *
errorAlarm5	Output	BOOL	FALSE	TRUE = Fehler Program_Alarm5	-
statusAlarm5	Output	WORD	-	Fehlerstatusinformation Program_Alarm5	-
alarmState5	Output	BYTE	-	Alarmstatus Program_Alarm5	-
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten	-
bufferWork	InOut	EnS_typeBuffer-Work (Seite 167)	-	Pufferkontroll- und Statusinformation	-

* Die markierten Parameter erhalten das Systemattribut "Erreichbar für HMI" und "Sichtbar in HMI". Sie werden zur Bedien- und Beobachtungsfunktion im HMI-Gerät verwendet, z. B. um einen aktuellen Wert im Bildbaustein anzuzeigen.

A.2.12.3 Parameter Status

Die Fehlerinformationen der untergeordneten Funktionsbausteine (z. B. „EnS_TimeSync“) werden zum Ausgangsparameter "status1", "status5" hinzugefügt. Die Fehlerinformation wird im Parameter "status" angezeigt, bis der Fehler beseitigt ist. Erst dann wird die Information des nächsten anstehenden Fehlers angezeigt.

A.2.12.4 Meldungen EnS_EnergyDataBasic

Die Meldungen werden über die Anweisung "Program_Alarm" generiert. Weitere Informationen zu "Program_Alarm" finden Sie in der Online-Hilfe des TIA Portal Informationssystems unter "Erweiterte Anweisungen > Meldungen".

Die folgende Tabelle zeigt die Meldungen des Funktionsbausteins "EnS_EnergyDataBasic":

Meldebaustein	Meldetext	Meldeklasse
Program_Alarm1	Parametrierfehler <Text list: EnS_Alarms:#status1>	Mit Quittierung
Program_Alarm2	<Text list: EnS_Alarms:#status2>	Mit Quittierung
Program_Alarm3	Archivierungsfehler <Text list: EnS_Alarms:#status3>.	Mit Quittierung
Program_Alarm4	Reserviert (Warnung für Grenzwertverletzung)	Mit Quittierung
Program_Alarm5	Reserviert (Warnung für Grenzwertverletzung)	Mit Quittierung

Die zusätzlichen Meldetexte sind in der Textliste "EnS_Alarms" beschrieben. Die Ausgangsparameter "status1" bis "status5" beinhaltet den Wert für die Fehlerinformation des FB "EnS_EnergyDataBasic" und dessen untergeordnete FBs.

Meldebaustein	Fehlercode (W#16#...)	Meldetext	Parameter	Funktionsbaustein
Program_Alarm1	801x	Hardware	status1	EnS_Drvxxx
	821x	Zeitsynchronisation		EnS_TimeSync (Seite 157)
	831x	Berechnung		EnS_CalcEnergy-Data (Seite 136)
	8311	Verschaltungsfehler Berechnung		

Meldebaustein	Fehlercode (W#16#...)	Meldetext	Parameter	Funktionsbaustein
Program_Alarm2	800x	Hardware-Fehler	status2	EnS_Drvxxx
	8201	Externes Synchronisationssignal verloren		EnS_TimeSync (Seite 157)
	8202	Externes Synchronisationssignal außerhalb gültigem Bereich		
	8203	Zeitsprung Synchronisationszeit		
ProgramAlarm3	8501	Fehler beim Schreiben in den Puffer	status3	EnS_BufferWrite (Seite 133)
	8502	Alarm: Puffer bei 100 %, Datenverlust		
	8503	Warnung: Püfferfüllstand über 80 %, bevorstehender Datenverlust		
	8504	Warnung: Pufferfüllstand über x %, bevorstehender Datenverlust		
	8512	Möglicher Datenverlust		EnS_BufferRead (Seite 131)

Die Begleitwerte sind folgendermaßen zugewiesen:

Meldebaustein	Begleitwert	Parameter	Bedeutung
Program_Alarm1	1	status1	Fehlerstatusinformation 1
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet
Program_Alarm2	1	status2	Fehlerstatusinformation 2
	2	sD	Frei
	3	-	Nicht verwendet
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet

Meldebaustein	Begleitwert	Parameter	Bedeutung
Program_Alarm3	1	status3	Fehlerstatusinformation 3
	2	Reserviert*	Reserviert
	3	sD	Frei
	4	-	Nicht verwendet
	5	-	Nicht verwendet
	6	-	Nicht verwendet
	7	-	Nicht verwendet
	8	-	Nicht verwendet
	9	-	Nicht verwendet
	10	-	Nicht verwendet

* Die Begleitwerte sind für eine zukünftige Nutzung reserviert.

Die Meldung ist änderbar.

Der Eingangsparameter "sD" wird als Meldebegleitwert verwendet. Sie können den Parameter mit weiteren Informationen verschalten, die in der Meldung erscheinen.

Die Meldung kann mit "messageSuppress" oder übergeordnet mit "orgData.messageSuppress" des Funktionsbausteins "EnS_Organization (Seite 156)" unterdrückt werden.

A.2.13 EnS_Organization: Organisationsdaten bereitstellen

A.2.13.1 Beschreibung EnS_Organization

Beschreibung

Der Funktionsbaustein "EnS_Organization" stellt zentral Organisationsdaten zur Verfügung, auf die von anderen Funktionsbausteinen zugegriffen wird. Der Baustein muss von jedem Weckalarm-OB einmal aufgerufen werden.

Der Baustein verwaltet folgende Daten:

- Start-up Flag ("orgData.startup")
  Das Start-Up-Flag ist ein Parameter vom Typ BOOL. Der Parameter liegt am Ausgangsparameter "orgData.startup“ des Funktionsbausteines "EnS_Organization" an. Der Funktionsbaustein meldet über diesen Parameter, ob sich das Programm in der Anlaufphase befindet. Die Anzahl der Zyklen für die Anlaufphase legen Sie am Eingangsparameter "runUpCyc“ des Funktionsbausteins fest. Eine Anlaufphase von zehn Zyklen ist voreingestellt.
- Meldungsunterdrückung ("orgData.messageSuppress")
  Mit dem Parameter für die Meldungsunterdrückung legen Sie fest, ob Meldungen aus dem Anwenderprogramm der CPU unterdrückt werden.
- Aktueller Zeitstempel (UTC) der CPU
  Der aktuelle Zeitstempel entspricht der Systemzeit der CPU.

Anlauf

Der Baustein setzt den Ausgangsparameter "orgData.startup" für die Anzahl der konfigurierten Anlaufzyklen "runUpCyc" und unterdrückt die Meldungen über den Parameter "orgData.messageSuppress".

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten.

A.2.13.2 Parameter EnS_Organization

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteins "EnS_Organization":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
runUpCyc	Input	UINT	10	Anzahl der Anlaufzyklen
messageSup-press	Input	BOOL	FALSE	TRUE = Meldungsunterdrückung aktiviert
orgData	Output	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation

A.2.14 EnS_TimeSync: Zeiten synchronisieren

A.2.14.1 Beschreibung EnS_TimeSync

Beschreibung

Der Baustein dient als zentraler Taktgeber zur Synchronisation für den Baustein zur Energieerfassung "EnS_CalcEnergyData (Seite 136)".

Der Takt kann folgendermaßen festgelegt werden:

Der Taktgeber "syncData.syncPulse" wird durch einen externen Synchronisationsimpuls "extSyncPulse", z. B. eines Energieversorgungsunternehmens (EVU), oder die interne CPU-Uhrzeit einmal pro Archivierungsperiode getriggert. Der Baustein wertet de steigende Flanke des Synchronisationsimpulses aus. Der Zeitstempel für den Synchronisationsimpuls wird am Parameter "syncData.syncTimeStamp" auf den nächsten ganzen Zeitwert auf- oder abgerundet, z. B. 15 Minuten.

Der Synchronisationsimpuls wird nur innerhalb eines festgelegten Gültigkeitsbereichs erfasst. Den Gültigkeitsbereich legen Sie über den Bausteinparameter "validTimeRange" fest. Für verfrühte Synchronisationsimpulse ist eine Toleranz vorgesehen. Verfrühte Synchronisationsimpulse werden auch dann noch als gültig betrachtet, wenn diese maximal einen CPU-Zyklus zu früh erfasst werden. Verfrühte Synchronisationsimpulse sind von der Zykluszeit des Weckalarm-OBs abhängig, in dem der Baustein "EnS_TimeSync" aufgerufen wird.

Wenn der Weckalarm-OB einmal pro Sekunde aufgerufen wird, dann muss der externe Synchronisationsimpuls frühestens bei der Sekunde 59 gesetzt werden.

Wenn der Weckalarm-OB einmal pro zwei Sekunden aufgerufen wird, dann muss der externe Synchronisationsimpuls frühestens bei der Sekunde 58 gesetzt werden.

Wenn der externe Synchronisationsimpuls eines EVU zu früh gesetzt wird, dann wird der Synchronisationsimpuls des Bausteins nicht gesetzt und der Fehlerstatus 16#8202 gesetzt.

Ungültige Impulse werden mit dem Status "16#8202" gekennzeichnet und mit einer Meldung quittiert.

- Externe Synchronisation "extEnable" = TRUE
  Beispiele:
  - Externer Synchronisationsimpuls "extSyncPulse" zu früh gesetzt:
    Die Zeitdauer zwischen dem aktuellen CPU-Zeitstempel und der nächsten Periode ist kleiner als das Berechnungsintervall des Funktionsbausteins. In diesem Fall wird der Synchronisationszeitstempel auf den nächsten vollen Wert der Periode aufgerundet.
    Berechnungsintervall: 1 s
    Periode: 15 min
    Aktuelle CPU-Uhrzeit: 6:59:59:500
    Synchronisationszeitstempel: 7:00:00:000
  - Externer Synchronisationsimpuls "extSyncPulse" zu spät gesetzt:
    Die Zeitdauer zwischen dem aktuellen CPU-Zeistempel und der letzten Periode ist kleiner als der Gültigkeitsbereich "validTimeRange". In diesem Fall wird der Synchronisationszeitstempel auf den letzten vollen Wert der Periode abgerundet.
    Berechnungsintervall: 1 s
    Periode: 15 min
    Gültigkeitsbereich: 10 s
    Aktuelle CPU-Uhrzeit 7:00:00:001
    Synchronisationszeitstempel: 7:00:00:000
  - Externer Synchronisationsimpuls "extSyncPulse" ist außerhalb des Gültigkeitsbereichs:
    Die Zeitdauer zwischen dem aktuellen CPU-Zeistempel und der letzten Periode ist größer als der Gültigkeitsbereich "validTimeRange". In diesem Fall wird der Synchronisationszeitstempel verworfen und eine Fehlermeldung (Status = 16#8202) angezeigt.
    Berechnungsintervall: 1 s
    Periode: 15 min
    Gültigkeitsbereich: 10 s
    Aktuelle CPU-Uhrzeit: 7:00:20:000
    Synchronisationszeitstempel: Fehlermeldung (Status = 16#8202)
- Festlegung durch Parameter "syncPeriod"
  Ist die externe Synchronisation deaktiviert "extEnable" = FALSE, beinhaltet der Parameter "syncPeriod" die Periodenzeit für die Synchronisation, z. B. 15 Minuten.
- Die Archivierung der Energiedaten verhält sich bei einem fehlenden Synchronisationsimpuls folgendermaßen:
  - Die Energiedaten für die Archivierungsperiode werden berechnet, bis der Synchronisationsimpuls gesetzt wird.
  - Die Leistungswerte werden über die komplette Archivierungsperiode, ab dem letzten Synchronisationsimpuls gemittelt.
  - Die Energiezählwerte werden über die komplette Archivierungsperiode, ab dem letzten Synchronisationsimpuls akkumuliert.

Zeitstempel werden als UTC Standardweltzeit ausgegeben.

Sie können mehrere Instanzen dieses Bausteins anlegen und somit mehrere Archivierungsperioden realisieren. Üblich sind Perioden von 15 Minuten für elektrische Energie und 60 Minuten für nicht elektrische Medien.

Anlauf

Im Anlauf werden die intern ablaufenden Zeiten neu gestartet.

Verhalten im Fehlerfall

Wenn ein Fehler auftritt, dann wird der Ausgangsparameter "error" gesetzt. Der Parameter "status (Seite 161)" enthält weitere Fehlerinformationen.

Meldungen

Der Baustein besitzt kein Meldeverhalten.

A.2.14.2 Parameter EnS_TimeSync

Die folgende Tabelle zeigt die Bausteinparameter des Funktionsbausteines "EnS_TimeSync":

Parameter	Deklaration	Datentyp	Vorbelegung	Beschreibung
extEnable	Input	BOOL	FALSE	TRUE = Freigabe für externe Synchronisation
extSyncPulse	Input	BOOL	FALSE	Externer Synchronisationsimpuls
syncPeriod	Input	TIME	T#15m	Archivierungsperiode
pulseTime	Input	TIME	T#2s	Dauer des Synchronisationsimpulses
validTimeRange	Input	TIME	T#10s	Gültigkeitsbereich für den externen Synchronisationsimpuls
monitoringTimeExtSync	Input	TIME	T#2s	Reserviert
modeExtSync	Input	BOOL	FALSE	Reserviert
offsetUTCToLocalTime	Input	TIME	T#0s	Verschiebung der UTC Standardweltzeit von der lokalen Zeit
syncData	Output	EnS_typeSync (Seite 177)	-	Synchronisationsdaten
curTimeStamp	Output	DT	-	Aktueller Zeitstempel bei Bausteinaufruf
error	Output	BOOL	FALSE	TRUE = Fehler
status	Output	WORD	-	Fehlerstatusinformation (Seite 161)
orgData	InOut	EnS_typeOrgData (Seite 176)	-	Organisationsdaten
resetErrorExtSync	InOut	BOOL	FALSE	Reserviert

A.2.14.3 Parameter Status

Die folgende Tabelle zeigt die Fehlercodes, die bei aufgetretenen Fehlern am Ausgangsparameter "status" ausgegeben werden:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8201	Der externe Synchronisationsimpupls extSyncPulse wurde während der Archivierungsperiode syncPeriod verloren.	Hardware-Fehler	extSyncPulse	**Fehlerursachen:** • Externer Synchronisationsimpuls verloren **Vorgehen:** • Prüfen Sie die HW-Adresse des Impulses. • Verwenden Sie bei Unzuverlässigkeit des externen Impulses die interne Zeitsynchronisation (CPU-Uhrzeit als Zeitgeber).
8202	Der externe Synchronisationsimpuls extSyncPulse wurde nicht innerhalb des Gültigkeitsbereichs validTimeRange gesetzt.		extSyncPulse / validTimeRange	**Fehlerursachen:** • Externer Synchronisationsimpuls wurde außerhalb des Gültigkeitsbereichs gesetzt. Standardmäßig ist der Gültigkeitsbereich auf 10 s eingestellt. **Vorgehen:** • Verwenden Sie bei Unzuverlässigkeit des externen Impulses die interne Zeitsynchronisation (CPU-Uhrzeit als Zeitgeber).
8203	Der Zeitsprung des aktuellen Zeitstempels ist größer als die Zeitdauer der Archivierungsperiode syncPeriod.	Zeitfehler	curTimeStamp / syncPeriod	Interner Fehler **Fehlerursachen**: • Der Zeitsprung des aktuellen Zeitstempels ist größer als die Zeitdauer der Archivierungsperiode. **Vorgehen:** • Verwenden Sie bei Unzuverlässigkeit des externen Impulses die interne Zeitsynchronisation (CPU-Uhrzeit als Zeitgeber).

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
8211	Die Zeitdauer der Archivierungsperiode syncPeriod ist kleiner als: • Zeitdauer des Synchronisation simpulses pulseTime oder • 1 Minute	Parametrierfehler	syncPeriod / pulseTime	Fehlerursachen: • Die Zeitdauer des Synchronisationsimpulses ist größer als die Zeitdauer der Archivierungsperiode oder größer als 1 Minute. Vorgehen: • Verwenden Sie bei Unzuverlässigkeit des externen Impulses die interne Zeitsynchronisation (CPU-Uhrzeit als Zeitgeber).
8212	Die Zeitdauer der Archivierungsperiode syncPeriod ist größer als ein Tag (24 Stunden).		syncPeriod	Fehlerursachen: • Die Zeitdauer der Archivierungsperiode syncPeriod ist größer als ein Tag (24 Stunden). Vorgehen: • Geben Sie einen gültigen Wert (1 min - 24 h) an der Einstellung für die Zeitdauer der Archivierungsperiode ein.
8213	Die Zeitdauer der Archivierungsperiode syncPeriod ist kein ganzzahliger Wert.		syncPeriod	Fehlerursachen: • Die Zeitdauer der Archivierungsperiode syncPeriod ist kein ganzzahliger Wert. Vorgehen: • Geben Sie einen gültigen Wert (ganze Zahl) an der Einstellung für die Zeitdauer der Archivierungsperiode ein.
8214	Die Zeitdauer der Archivierungsperiode syncPeriod ist kein ganzzahliger Teil einer Stunde / eines Tages. Gültig sind z. B.: • 1 Minute • 5 Minuten • 15 Minuten • 12 Stunden Ungültig sind z. B.: • 7 Minuten • 11 Minuten • 5 Stunden		syncPeriod	Fehlerursachen: • Die Zeitdauer der Archivierungsperiode syncPeriod ist kein ganzzahliger Teil einer Stunde / eines Tages. Vorgehen: • Geben Sie einen gültigen Wert an der Einstellung für die Zeitdauer der Archivierungsperiode ein.
8215	Die Verschiebung der Standardweltzeit UTC zu der lokalen Zeit ist kein Vielfaches von 15 Minuten oder kleiner als 15 Minuten.		extSyncPulse	Fehlerursachen: • Die Verschiebung der Standardweltzeit UTC zu der lokalen Zeit ist kein Vielfaches von 15 Minuten oder kleiner als 15 Minuten. Vorgehen:

Fehlercode (W#16#...)	Beschreibung	Fehlerart	Wird kopiert in Parameter	Abhilfe
	Gültig sind z. B.: • 1 Stunde • 4 Stunden • 30 Minuten Ungültig sind z. B.: • 10 Minuten • 25 Minuten			• Geben Sie einen gültigen Wert für die Verschiebung der Standardweltzeit UTC zu der lokalen Zeit ein.

A.3 Datenbausteine (DBs)

A.3.1 EnS_BufferDB

A.3.1.1 Beschreibung EnS_BufferDB

Beschreibung

Der Globaldatenbaustein "EnS_BufferDB" beinhaltet die periodisch erfassten Energiedaten in einem Array. Die Energiedaten werden vom Funktionsbaustein "EnS_EnergyDataBasic (Seite 152)" in den Datenbaustein "EnS_BufferDB" geschrieben. Dazu wird der Funktionsbaustein "EnS_BufferWrite (Seite 133)" vom Funktionsbaustein "EnS_EnergyDataBasic" intern aufgerufen.

Das Lesen der Energiedaten aus dem Datenbaustein "EnS_BufferDB" erfolgt durch den Funktionsbaustein "EnS_ArchiveDataLog (Seite 116)" oder die Energy Suite SCADA-Komponente. Dazu wird der Funktionsbaustein "EnS_BufferRead (Seite 131)" vom Funktionsbaustein "EnS_ArchiveDataLog" intern aufgerufen.

Die Größe des Arrays können Sie entsprechend Ihren Anforderungen und der Speicherkapazität der CPU anpassen. Dazu stellen Sie in den Eigenschaften des Datenbausteins "EnS_BufferDB" die gewünschte Anzahl der Array-Elemente ein. Im Baustein "EnS_BufferWorkDB (Seite 165)" stellen Sie außerdem eine Reihe weiterer Parameter ein, z. B. eine Füllstandwarnung.

Jedes Array-Element entspricht dem Wert des Typs "EnS_typeValue (Seite 178)". Darin sind weitere vier Werte enthaltene

- Variablenname
- Wert
- Zeitstempel
- Status

Sie können mehrere Datenbausteine zur Pufferung zu einer CPU hinzufügen, solange die CPU genügend freien Speicher zur Verfügung hat. Zu einem Paar gehören immer zwei Datenbausteine: "EnS_BufferDB" und "EnS_BufferWorkDB".

A.3.1.2 Aufbau EnS_BufferDB

Die folgende Tabelle zeigt den Aufbau des Datenbausteins "EnS_BufferDB":

Element	Datentyp	Beschreibung
EnS_BufferDB	ARRAY[0..n] of "EnS_typeValue"	n+1 Elemente berechneter Datenwerte

A.3.2 EnS_BufferWorkDB

A.3.2.1 Beschreibung EnS_BufferWorkDB

Beschreibung

Der Datenbaustein "EnS_BufferWorkDB" ist ein Globaldatenbaustein vom Typ "EnS_typeBufferWork (Seite 167)". Der Datenbaustein beinhaltet die Kontroll- und Statusinformation des Puffer-DBs "EnS_BufferDB (Seite 164)". Die Werte des Datenbausteins "EnS_BufferWorkDB" werden intern von den Funktionsbausteinen "EnS_BufferRead (Seite 131)" und "EnS_BufferWrite (Seite 133)" für die Koordinierung der Lese- und Schreibzugriffe auf den Puffer-DB "EnS_BufferDB" ausgewertet und beschrieben.

A.3.2.2 Aufbau EnS_BufferWorkDB

Die folgende Tabelle zeigt den Aufbau des Datenbausteins "EnS_BufferWorkDB":

Element	Datentyp	Beschreibung
EnS_BufferWorkDB	EnS_typeBufferWork (Seite 167)	Pufferkontroll- und Statusinformation

A.4 Anwenderdefinierte Datentypen (UDTs)

A.4.1 EnS_typeAnalogValue

A.4.1.1 Beschreibung EnS_typeAnalogValue

Beschreibung

Der PLC-Datentyp "EnS_typeAnalogValue" beschreibt die Schnittstelle zwischen dem Funktionsbaustein für analoge Eingänge "EnS_DrvAnalogValue (Seite 142)" und dem Funktionsbaustein für periodische Berechnung von Energiedaten "EnS_CalcEnergyData (Seite 136)".

A.4.1.2 Aufbau EnS_typeAnalogValue

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeAnalogValue":

Parameter	Datentyp	Vorbelegung	Beschreibung
value	REAL	0.0	Prozesswert
status	WORD	16#FFFF	Status des Prozesswerts

A.4.2 EnS_typeBufferReadCtrl

A.4.2.1 Beschreibung EnS_typeBufferReadCtrl

Beschreibung

Der PLC-Datentyp "EnS_typeBufferReadCtrl" beinhaltet die erforderlichen Kontrollinformationen, um Daten aus dem Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)" auszulesen.

A.4.2.2 Aufbau EnS_typeBufferReadCtrl

Die folgende Tabelle zeigt den Parameter des PLC-Datentyps "EnS_typeBufferReadCtrl":

Parameter	Datentyp	Vorbelegung	Beschreibung
req	BOOL	FALSE	Anforderung um Daten aus dem Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)" zu lesen

A.4.3 EnS_typeBufferReadStat

A.4.3.1 Beschreibung EnS_typeBufferReadStat

Beschreibung

Der PLC-Datentyp "EnS_typeBufferReadStat" beinhaltet die zu lesenden Daten und die Statusinformationen für das Lesen vom Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)".

A.4.3.2 Aufbau EnS_typeBufferReadStat

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeBufferReadStat":

Parameter	Datentyp	Vorbelegung	Beschreibung
data	EnS_typeValue (Seite 178)	-	Aus dem Puffer gelesene Daten
busy	BOOL	FALSE	TRUE = Bearbeitung ist aktiv
done	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet
error	BOOL	FALSE	TRUE = Fehler beim Lesen
status	WORD	16#0000	Status des Puffers
statusReadFct	WORD	16#0000	Status von "ReadFromArrayDB"

A.4.4 EnS_typeBufferWork

A.4.4.1 Beschreibung EnS_typeBufferWork

Beschreibung

Der PLC-Datentyp "EnS_typeBufferWork" beinhaltet alle Kontroll- und Statusinformationen für das Schreiben und Lesen von Daten in den Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)".

A.4.4.2 Aufbau EnS_typeBufferWork

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeBufferWork":

Parameter	Datentyp	Vobelegung	Beschreibung
readCtrl	EnS_typeBufferReadCtrl (Seite 166)	-	Kontrollinformationen des Puffers für das Lesen von Daten
readStat	EnS_typeBufferReadStat (Seite 167)	-	Daten und Statusinformationen des Puffers für das Lesen von Daten

Parameter	Datentyp	Vobelegung	Beschreibung
writeCtrl	EnS_typeBufferWriteCtrl (Seite 168)	-	Kontrollinformationen des Puffers für das Schreiben von Daten
writeStat	EnS_typeBufferWriteStat (Seite 169)	-	Statusinformationen des Puffers für das Schreiben von Daten
busy	BOOL	FALSE	TRUE = Bearbeitung ist aktiv
done	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet
error	BOOL	FALSE	TRUE = Fehler bei der Datenverarbeitung
status	WORD	16#0000	Status des Puffers
maxItem	UDINT	0	Maximale Anzahl von Datensätzen des "EnS_BufferDB"
maxItemRead	UDINT	0	Reserviert
maxCycleWrite	UDINT	30	Maximale Anzahl der Zyklen zum Schreiben der Daten
cycleWrite	UINT	0	Anzahl der Zyklen zum Schreiben der Daten
noItems	UDINT	0	Anzahl geschriebener Datensätze des "EnS_BufferDB"
noDeletedItems	UDINT	0	Anzahl gelöschter Datensätze des "EnS_BufferDB" bei Pufferüberlauf
writeItem	DINT	0	Schreibzeiger auf den Datensatz
readItem	DINT	0	Lesezeiger auf den Datensatz
bufferLimit	UINT	20	Konfigurierbares Pufferlimit für Warnungen in [%] (Gültigkeitsbereich 1-79). Ab einem Füllstand des Puffers von 80 % wird standardmäßig ein Fehlerstatus ausgegeben (siehe auch "EnS_BufferWrite (Seite 133)").

A.4.5 EnS_typeBufferWriteCtrl

A.4.5.1 Beschreibung EnS_typeBufferWriteCtrl

Beschreibung

Der PLC-Datentyp "EnS_typeBufferWriteCtrl" beinhaltet die periodischen Energiedaten, die in den Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)" geschrieben werden sollen, und die erforderlichen Kontrollinformationen (z. B. Abfrage), um Daten zu schreiben.

A.4.5.2 Aufbau - EnS_typeBufferWriteCtrl

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeBufferWriteCtrl":

Parameter	Datentyp	Vorbelegung	Beschreibung
req	BOOL	FALSE	Anforderung um Daten in den Puffer zu schreiben.
data	EnS_typeValue (Seite 178)	-	Periodische Energiedaten, die in den Puffer geschrieben werden.

A.4.6 EnS_typeBufferWriteStat

A.4.6.1 Beschreibung EnS_typeBufferWriteStat

Beschreibung

Der PLC-Datentyp "EnS_typeBufferWriteStat" beinhaltet Statusinformationen für das Schreiben von periodischen Energiedaten in Datenbaustein zur Pufferung "EnS_BufferDB (Seite 164)".

A.4.6.2 Aufbau EnS_typeBufferWriteStat

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeBufferWriteStat":

Parameter	Datentyp	Vorbelegung	Beschreibung
busy	BOOL	FALSE	TRUE = Bearbeitung ist aktiv
done	BOOL	FALSE	TRUE = Daten erfolgreich verarbeitet
error	BOOL	FALSE	TRUE = Fehler bei der Datenverarbeitung
status	WORD	16#0000	Status des Puffers
statusWriteFct	WORD	16#0000	Schreibstatus

A.4.7 EnS_typeEnergyAdv

A.4.7.1 Beschreibung EnS_typeEnergyAdv

Beschreibung

Der PLC-Datentyp "EnS_typeEnergyAdv" beinhaltet erweiterte Energiedaten, die von Messgeräten ausgelesen und für eine CPU oder ein HMI-System bereitgestellt werden.

A.4.7.2 Aufbau EnS_typeEnergyAdv

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeEnergyAdv":

Parameter	Datentyp	Vorbelegung	Beschreibung
voltage1N	REAL	0.0	Aktuelle Spannung zwischen Phase 1 und N
voltage2N	REAL	0.0	Aktuelle Spannung zwischen Phase 2 und N
voltage3N	REAL	0.0	Aktuelle Spannung zwischen Phase 3 und N
voltage12	REAL	0.0	Aktuelle Spannung zwischen Phase 1 und 2
voltage23	REAL	0.0	Aktuelle Spannung zwischen Phase 2 und 3
voltage31	REAL	0.0	Aktuelle Spannung zwischen Phase 3 und 1
current1	REAL	0.0	Aktueller Strom auf Phase 1
current2	REAL	0.0	Aktueller Strom auf Phase 2
current3	REAL	0.0	Aktueller Strom auf Phase 3
frequency	REAL	0.0	Aktuelle Frequenz
powerFactor1	REAL	0.0	Aktueller Leistungsfaktor der Phase 1
powerFactor2	REAL	0.0	Aktueller Leistungsfaktor der Phase 2
powerFactor3	REAL	0.0	Aktueller Leistungsfaktor der Phase 3
totalPowerFactor	REAL	0.0	Aktueller Gesamt-Leistungsfaktor
appPower1	REAL	0.0	Aktuelle Scheinleistung der Phase 1
appPower2	REAL	0.0	Aktuelle Scheinleistung der Phase 2
appPower3	REAL	0.0	Aktuelle Scheinleistung der Phase 3
totalAppPower	REAL	0.0	Aktuelle Gesamt-Scheinleistung
actPower1	REAL	0.0	Aktuelle Wirkleistung der Phase 1
actPower2	REAL	0.0	Aktuelle Wirkleistung der Phase 2
actPower3	REAL	0.0	Aktuelle Wirkleistung der Phase 3
totalActPower	REAL	0.0	Aktuelle Gesamt-Wirkleistung
reactPower1	REAL	0.0	Aktuelle Blindleistung der Phase 1
reactPower2	REAL	0.0	Aktuelle Blindleistung der Phase 2
reactPower3	REAL	0.0	Aktuelle Blindleistung der Phase 3
totalReactPower	REAL	0.0	Aktuelle Gesamt-Blindleistung
totalAppEnergy	LREAL	0.0	Aktuelle Gesamt-Scheinenergie
totalActEnergy	LREAL	0.0	Aktuelle Gesamt-Wirkenergie
totalReactEnergy	LREAL	0.0	Aktuelle Gesamt-Blindenergie
status	WORD	16#FFFF	Status der erweiterten Daten

A.4.8 EnS_typeEnergyBasic

A.4.8.1 Beschreibung EnS_typeEnergyBasic

Beschreibung

Der PLC-Datentyp "EnS_typeEnergyBasic" beinhaltet Basisenergiedaten, die für eine CPU, ein HMI-System und die Archivierung bereitgestellt werden, z. B.:

- Periodisch berechneter Energieverbrauch
- Durchschnittliche Leistung über die Periode
- Aktuelle Werte für Energieverbrauch und Leistungsaufnahme

A.4.8.2 Aufbau EnS_typeEnergyBasic

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeEnergyBasic":

Parameter	Datentyp	Vorbelegung	Beschreibung
actEnergy	REAL	0.0	Aktueller Energiewert
actPower	REAL	0.0	Aktueller Leistungswert
actEnergyCounter	LREAL	0.0	Aktueller Energiezählwert
actStatus	WORD	16#FFFF	Status der aktuellen Werte
energy	REAL	0.0	Kumulierter Energiewert am Ende der Archivierungsperiode
power	REAL	0.0	Durchschnittlicher Leistungswert am Ende der Archivierungsperiode
energyCounter	LREAL	0.0	Energiezählerwert am Ende der Archivierungsperiode
status	WORD	16#FFFF	Status der kumulierten Energie und Durchschnittsleistung
syncData	EnS_typeSync (Seite 177)	-	Synchronisationsdaten

A.4.9 EnS_typeEnergyCounter

A.4.9.1 Beschreibung EnS_typeEnergyCounter

Beschreibung

Der PLC-Datentyp "EnS_typeEnergyCounter" beinhaltet eine Schnittstelle zwischen dem Treiberfunktionsbaustein für Energiezählwerte "EnS_DrvCounterLInt (Seite 144)", "EnS_DrvCounterLReal (Seite 146)" und dem Funktionsbaustein für die periodische Berechnung von Energiedaten "EnS_CalcEnergyData (Seite 136)".

A.4.9.2 Aufbau EnS_typeEnergyCounter

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeEnergyCounter":

Parameter	Datentyp	Vorbelegung	Beschreibung
value	LREAL	0.0	Energiezählwert
normValue	LREAL	0.0	Normierter Energiezählwert
consumerStatus	BOOL	FALSE	Verbraucherstatus: TRUE = Ein, FALSE = Aus
status	WORD	16#FFFF	Status des Energiewerts

A.4.10 EnS_typeEnergyMeta

A.4.10.1 Beschreibung EnS_typeEnergyMeta

Beschreibung

Der PLC-Datentyp "EnS_typeEnergyMeta" enthält Informationen zur Messstelle. Die Informationen dienen der Anzeige in der CPU oder einem HMI-System.

A.4.10.2 Aufbau EnS_typeEnergyMeta

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeEnergyMeta":

Parameter	Datentyp	Vorbelegung	Wertebereich	Beschreibung	HMI [1)]
name	STRING[32]	" "	-	Name des Energieobjekts	X
plantIdentCode	STRING[128]	" "	-	Anlagenkennzeichen	-
localIdentCode	STRING[128]	" "	-	Ortskennzeichen	-
techUnitId	DWORD	0	0 = Nicht konfiguriert	ID für technologische Einheit	-
deviceId	DWORD	0	0 = ungültig 16#0000_0001 - 16#FFFF_FFFF	Geräte-ID des Messgeräts	X
parentDeviceId	DWORD	0	0 = Nicht konfiguriert	Geräte-ID des übergeordneten Messgeräts (Hierarchiebeziehung)	X
id	ARRAY[0..9] of HW_IO	0	0 = Nicht konfiguriert	Hardware-IDs des Messgerätes in der Hardware-Konfiguration	-
systemParam	ARRAY[0..6] of LREAL	0	-	Systemparameter für die interne Verwendung	-
connection-Type[2)]	USINT	0	0 - 16 (siehe [2)])	Verbindungstyp des Messgerätes	-

Parameter	Datentyp	Vorbelegung	Wertebereich	Beschreibung	HMI [1)]
inputType	USINT	0	0 = ungültig 1 = Impulssignal 2 = Energiezählwert 3 = Leistungswert 4 = Energiezählwert und Leistungswert	Art des Eingangssignals	X
mode	USINT	0	0 = ungültig 1 = Verbraucher 2 = Erzeuger	Parametrierung der Energieflussrichtung	X
accuracyVolt-Conv	REAL	0.0	0.0 = Genauigkeits-einstellung nicht aktiv Gültigkeitsbereich: 0.0 - 100.0	Wandlungsfehler des Spannungswandlers in [%]	-
accuracyCurr-Conv	REAL	0.0	0.0 = Genauigkeits-einstellung nicht aktiv Gültigkeitsbereich: 0.0 - 100.0	Wandlungsfehler des aktuellen Stromwandlers in [%]	-
accuracyDe-vice	REAL	0.0	0.0 = Genauigkeits-einstellung nicht aktiv Gültigkeitsbereich: 0.0 - 100.0	Messfehler des Messgeräts in [%]	-
accuracyAdd	REAL	0.0	0.0 = Genauigkeits-einstellung nicht aktiv Gültigkeitsbereich: 0.0 - 100.0	Zusätzlicher Fehlerfaktor in [%]	-
correctionFac-tor	REAL	0.0	0.0 = ungültig Gültigkeitsbereich: 0.001 - 1d	Korrekturfaktor	-
syncPeriod	REAL	900.0	60.0 - 86400.0 (= 1min -1d)	Archivierungsperiode in [s]	X
overflowCntVa-lue	LREAL	0.0	1.0 - 1.0e+21	Überlaufwert des Zählers	X
normFactorIn	REAL	1.0	0.0 - 9999.0	Normierungsfaktor für Eingangswerte (selber Normierungsfaktor für Energie- und Leistungswerte)	X
normFactorOut	REAL	0.001	0.0 - 9999.0 (0.001 = Output entspricht kWh)	Normierungsfaktor für Ausgangswerte	X
zeroCut	REAL	0.1	0.0 - 3.402823e+38	Grenze für Nullpunkt bei der Energieberechnung	X
baseAcqTime	REAL	10.0	0.0 - 3600.0 (1 h) siehe auch EnS_CalcEnergy-Data (Seite 136)	Berechnungsintervall für den Momentanwert der Leistungsaufnahme in [s]	X

Parameter	Datentyp	Vorbelegung	Wertebereich	Beschreibung	HMI [1]
addAcqTime	REAL	10.0	0.0 - 3600.0 (=1h) siehe auch EnS_CalcEnergy-Data (Seite 136)	Dynamische Verlängerung des Berechnungsintervalls für den Momentanwert der Leistungsaufnahme in [s]	X
consumerStatusEnable	BOOL	FALSE	TRUE = ein FALSE = aus	Verbraucher-/Erzeugerstatus ist aktiv	-
coordinationEnable	BOOL	FALSE	-	TRUE = Koordination der azyklischen Kommunikationsdienste ist aktiviert	-
cyclicRd	BOOL	FALSE	-	TRUE = zyklisches Lesen der azyklischen Energiedaten ist aktiviert	-
cyclicTime	REAL	60.0	0.0 - 3600.0 (1 h)	Zeitdauer für das zyklische Lesen der azyklsichen Energiedaten in [s]	-
unitActEnergy[3]	UINT	0	siehe [3]	Einheit für Wirkenergie	-
unitActPower[3]	UINT	0	siehe [3]	Einheit für Wirkleistung	-
unitAppEnergy[3]	UINT	0	siehe [3]	Einheit für Scheinenergie	-
unitAppPower[3]	UINT	0	siehe [3]	Einheit für Scheinleistung	-
unitReactEnergy[3]	UINT	0	siehe [3]	Einheit für Blindenergie	-
unitReactPower[3]	UINT	0	siehe [3]	Einheit für Blindleistung	-
unitPowerFactor[3]	UINT	0	siehe [3]	Einheit für Leistungsfaktor	-
unitCurrent[3]	UINT	0	siehe [3]	Einheit für Strom	-
unitVoltage[3]	UINT	0	siehe [3]	Einheit für Spannung	-
unitFrequency[3]	UINT	0	siehe [3]	Einheit für Frequenz	-
unitAmpUnbalCurrent[3]	UINT	0	siehe [3]	Einheit für Amplitudensymmetrie des Stroms	-
unitAmpUnbalVoltage[3]	UINT	0	siehe [3]	Einheit der Amplitudenunsymmetrie der Spannung	-
unitAddValue[3]	ARRAY[0..19]	0	siehe [3]	Reserviert	-
typeAddValue	ARRAY[0..19]	0	0 - 16 (siehe Input-Type)	Reserviert	-
archiveEnergy	BOOL	FALSE	-	FALSE = Archivierung des Energiewerts deaktiviert, TRUE = Archivierung des Energiewerts aktiviert	-
archivePower	BOOL	FALSE	-	FALSE = Archivierung des Leistungswerts deaktiviert, TRUE = Archivierung des Leistungswerts aktiviert	-
archiveEnergyCounter	BOOL	FALSE	-	FALSE = Archivierung des Energiezählerwerts deaktiviert, TRUE = Archivierung des Energiezählerwerts aktiviert	-
archiveAddValue	ARRAY[0..19]	0	-	Reserviert	-

[1] Die markierten Parameter erhalten das Systemattribut "Erreichbar für HMI" und "Sichtbar in HMI". Sie werden zur Bedien- und Beobachtungsfunktion im HMI-Gerät verwendet, z. B. um einen aktuellen Wert in einem Bildbaustein anzuzeigen.

[2] Enumeration für den Verbindungstyp des Messgeräts (Seite 179)

[3] Enumeration für Einheiten (Seite 179)

Siehe auch

Parameter Status (Seite 135)

A.4.11 EnS_typeHMIWriteCtrl

A.4.11.1 Beschreibung EnS_typeHMIWriteCtrl

Der PLC-Datentyp EnS_typeHMIWriteCtrl beinhaltet Kotrollinformationen für das Schreiben der Energiedaten in das SCADA-System.

A.4.11.2 Aufbau EnS_typeHMIWriteCtrl

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeHMIWriteCtrl":

Parameter	Datentyp	Vorbelegung	Beschreibung	HMI *
req	BOOL	FALSE	Auftrag zum Schreiben der Energiedaten in das SCADA-System	X
maxItem	UDINT	0	Maximale Anzahl der Datensätze im Puffer	X
noItems	UDINT	0	Anzahl der geschriebenen Datensätze in das SCADA-System	X
noDeletedItems	UDINT	0	Anzahl gelöschter Datensätze bei Pufferüberlauf	X
writeItems	DINT	0	Schreibzeiger auf den Datensatz im Puffer	X
readItems	DINT	0	Lesezeiger auf den Datensatz im Puffer	X
watchdog	UINT	0	Watchdog-Zähler	X

* Den markierten Parametern sind die Systemparameter "Erreichbar für HMI" und "Sichtbar für HMI" zugeordnet. Die Parameter werden zum Datenaustausch zu der Energy Suite SCADA-Komponente verwendet.

A.4.12 EnS_typeHMIWriteStat

A.4.12.1 Beschreibung EnS_typeHMIWriteStat

Der PLC-Datentyp EnS_typeHMIWriteStat beinhaltet Statusinformationen für das Schreiben der Energiedaten in das Variablenarchiv des SCADA-Systems.

A.4.12.2 Aufbau EnS_typeHMIWriteStat

Die folgende Tabelle zeigt die Parameter der PLC-Datentyps "EnS_typeHMIWriteStat":

Parameter	Datentyp	Vorbelegung	Beschreibung	HMI *
busy	BOOL	FALSE	TRUE = WinCC-Skript: Bearbeitung läuft	X
done	BOOL	FALSE	TRUE = Energiedaten sind erfolgreich in SCADA-System geschrieben	X
error	BOOL	FALSE	TRUE = Fehler ist beim Schreiben der Energiedaten aufgetreten	X
status	WORD	16#0000	Status des WinCC-Sripts	X
noItems	UDINT	0	Anzahl geschriebener Datensätze	X
readItem	DINT	0	Lesezeiger auf den Datensatz im Puffer	X
writeHMIItem	DINT	0	Schreibzeiger auf den Datensatz im Variablenarchiv	X
watchdog	UINT	0	Watchdog-Zähler	X

* Den markierten Parametern sind die Systemparameter "Erreichbar für HMI" und "Sichtbar für HMI" zugeordnet. Die Parameter werden zum Datenaustausch zu der Energy Suite SCADA-Komponente verwendet.

A.4.13 EnS_typeOrgData

A.4.13.1 Beschreibung EnS_typeOrgData

Beschreibung

Der PLC-Datentyp "EnS_typeOrgData" beinhaltet übergeordnete Daten und Einstellungen, auf die alle Bausteine zugreifen. Somit können die Informationen übergeordnet an die Bausteine weitergeleitet werden, z. B. Meldesperre.

A.4.13.2 Aufbau EnS_typeOrgData

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeOrgData":

Parameter	Datentyp	Vorbelegung	Beschreibung
startup	BOOL	TRUE	TRUE = Anlauf ist aktiv
messageSuppress	BOOL	FALSE	TRUE = Meldungsunterdrückung
currTimeStamp	DT	-	Aktueller Zeitstempel (UTC-Format)

A.4.14 EnS_typeSync

A.4.14.1 Beschreibung EnS_typeSync

Beschreibung

Der PLC-Datentyp "EnS_typeSync" beinhaltet alle Informationen über die Archivierungsperiode einschließlich des Synchronisationsimpulses.

A.4.14.2 Aufbau EnS_typeSync

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeSync":

Parameter	Datentyp	Vorbelegung	Beschreibung	HMI *
syncPeriod	REAL	900.0	Archivierungsperiode in [s]	X
syncTime	REAL	0.0	Verbleibende Zeit der Archivierungsperiode in [s]	X
syncPulse	BOOL	FALSE	Synchronisationsimpuls	-
syncTimeStamp	DT	DT#1990-01-01-00:00:00.0	Zeitstempel des Synchronisationsimpulses	X
error	BOOL	FALSE	1 = Fehler	-
status	WORD	16#0000	Fehlerstatusinformation	-

* Den markierten Parameter werden die Systemparameter "Erreichbar für HMI" und "Sichtbar für HMI". Sie können zur Bedien- und Beobachtungsfunktion im HMI verwendet werden.

A.4.15 EnS_typeValue

A.4.15.1 Beschreibung EnS_typeValue

Beschreibung

Der PLC-Datentyp "EnS_typeValue" beinhaltet alle Informationen von periodisch erfassten Energiedaten, die archiviert werden sollen, z. B. im Variablenarchiv. Der globale Datenbaustein "EnS_BufferDB (Seite 164)" besteht aus einem ARRAY des PLC-Datentyps "EnS_typeValue".

A.4.15.2 Aufbau EnS_typeValue

Die folgende Tabelle zeigt die Parameter des PLC-Datentyps "EnS_typeValue":

Parameter	Datentyp	Vorbelegung	Beschreibung	HMI *
tagname	STRING[64]	" "	Variablenname	X
value	LREAL	0.0	Periodischer Energiewert	X
timeStamp	DT	DTL#1990-01-01-00:00:00.0	Zeitstempel des periodischen Energiewerts	X
status	WORD	16#0000	Status des periodischen Energiewerts	X

A.5 Definition der Enumerationen

A.5.1 Enumeration für den Verbindungstyp des Messgeräts

Die folgende Tabelle zeigt die Enumeration für den Verbindungstyp des Messgeräts:

Nummer	Verbindungstyp
0	Nicht definiert
1	1 Phase
2	1 Phase - 3 Phase (symmetrisch)
3	3 Phasen (symmetrisch)
4	3 Phasen + N (asymmetrisch) ohne Spannungswandler / mit 3 Stromwandlern 3P4W
5	3 Phasen + N (asymmetrisch) mit Spannungswandler / mit 3 Stromwandlern 3P4W
6	3 Phasen + N (symmetrisch) ohne Spannungswandler / mit 1 Stromwandler 3P4WB
7	3 Phasen + N (symmetrisch) mit Spannungswandler / mit 1 Stromwandler 3P4WB
8	3 Phasen (asymmetrisch) ohne Spannungswandler / mit 3 Stromwandlern 3P3W
9	3 Phasen (asymmetrisch) mit Spannungswandler / mit 3 Stromwandlern 3P3W
10	3 Phasen (asymmetrisch) ohne Spannungswandler / mit 3 Stromwandlern 3P3W
11	3 Phasen (asymmetrisch) mit Spannungswandler / mit 2 Stromwandlern 3P3W
12	3 Phasen (symmetrisch) ohne Spannungswandler / mit 1 Stromwandler 3P3WB
13	3 Phasen (symmetrisch) mit Spannungswandler / mit 1 Stromwandler 3P3WB
14	2 Phasen + N (asymmetrisch) ohne Spannungswandler / mit 2 Stromwandlern 3P4W
15	1 Phase + N ohne Spannungswandler / mit 1 Stromwandler 1P2W
16	3 Phasen + N (asymmetrisch) mit Spannungswandler / mit 3 Stromwandlern 3P3W

A.5.2 Enumeration für Einheiten

Die folgende Tabelle zeigt die Enumeration für die Einheiten:

Nummer	Einheit
Elektrische Einheiten	
0	Nicht definiert
1209	A
1210	kA
1240	V
1242	kV
1241	MV
1186	W
1190	kW
1189	MW

Nummer	Einheit
1188	GW
1175	Wh
1179	kWh
1178	MWh
1177	GWh
1342	%
1077	Hz
Nichtelektrische Einheiten	
1034	m³
1573	Nm³
1038	l
1041	hl
1088	kg
1092	t
32800	pcs
1598	m³/h
1590	Nm³/h
1353	l/h
1635	hl/h
1324	kg/h
1328	t/h
32801	pcs/h

SIEMENS

Team Engineering einsetzen

Programmier- und Bedienhandbuch

Übersicht über das Arbeiten mit Team Engineering 1

Multiuser Engineering einsetzen 2

Multiuser Commissioning einsetzen 3

Projekte mit PLC-Synchronisierung in Betrieb nehmen 4

Daten mit Inter Project Engineering (IPE) austauschen 5

10/2018

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

℗ 10/2018 Änderungen vorbehalten

Copyright © Siemens AG 2018.
Alle Rechte vorbehalten

Inhaltsverzeichnis

1 Übersicht über das Arbeiten mit Team Engineering....7

2 Multiuser Engineering einsetzen....13

2.1 Einführung in Multiuser Engineering....13

2.2 Übersicht über die Arbeitsweise mit Multiuser Engineering....16

2.3 Grundlagen zum Multiuser-Serverkonzept....21

2.4 Voraussetzungen für das Arbeiten mit Multiuser Engineering....24

2.5 Unterstützte Objekte für Multiuser Engineering....27

2.6 Einstellungen für das Arbeiten mit Multiuser Engineering....29

2.7 Verbesserung der Funktionalität von Multiuser Engineering....32

2.8 Anforderungen an Multiuser-Projekte....35

2.9 Upgrade von Multiuser-Projekten....37

2.10 Offline Arbeiten mit Multiuser Engineering....38

2.11 Bedienelemente bei Multiuser Engineering....40
2.11.1 Multiuser-Symbole in der Bedienoberfläche....40
2.11.2 Multiuser-Markierungen in der Bedienoberfläche....41
2.11.3 Multiuser-Banner in der Bedienoberfläche....44
2.11.4 Funktionalität des Multiuser-Editors....45

2.12 Multiuser-Server installieren und deinstallieren....49
2.12.1 Hinweise zur Installation des Multiuser-Servers....49
2.12.2 Hinweise zur Kompatibilität des Multiuser-Servers....51
2.12.3 Multiuser-Server installieren....52
2.12.4 Erneute Installation des Multiuser-Servers ohne Datenverlust durchführen....54
2.12.5 Multiuser-Server deinstallieren....55

2.13 Multiuser-Server konfigurieren und verwalten....57
2.13.1 Möglichkeiten zum Konfigurieren und Verwalten des Multiuser-Servers....57
2.13.2 Grafische Tools verwenden....58
2.13.2.1 Einführung zu den grafischen Multiuser-Tools....58
2.13.2.2 Multiuser-Server mit grafischem Configuration Tool konfigurieren....60
2.13.2.3 Multiuser-Server mit grafischem Administration Tool verwalten....65
2.13.2.4 Neue Serververbindung hinzufügen und löschen....67
2.13.2.5 Benutzerverwaltung für Multiuser-Server und Projekte festlegen....70
2.13.2.6 Warnhinweise verwalten....73
2.13.2.7 Multiuser-Projekte und lokale Sessions verwalten....74
2.13.2.8 Historie für Projekte anzeigen und Revisionen verwalten....77
2.13.2.9 Zwischen Engineering- und Commissioning-Modus umschalten....81
2.13.2.10 Oberflächensprache auswählen....83
2.13.2.11 Bestätigungen anzeigen....83
2.13.2.12 Zertifikate verwalten....85
2.13.3 Kommandozeilen-Tools verwenden....86

2.13.3.1 Einführung zu den Multiuser-Kommandozeilen-Tools....86
2.13.3.2 Multiuser-Server mit den Administrative Tools konfigurieren....88
2.13.3.3 Befehle zur Konfiguration des Multiuser-Servers....90
2.13.3.4 Multiuser-Server mit den Power Tools verwalten....92
2.13.3.5 Befehle zur Verwaltung des Multiuser-Servers....93
2.13.4 Zertifikat für Multiuser-Server bestätigen....100
2.13.5 Zugangsdaten für Multiuser-Server eingeben....101
2.13.6 Serverdaten auf anderen Multiuser-Server übertragen....102
2.13.7 Error-Log für Multiuser-Server auslesen....103
2.13.8 Error-Log für Multiuser-Server als Zip-Datei verpacken....104
2.13.9 Hinweise zum Fehlerverhalten des Multiuser-Servers....105
2.13.10 Unterstützte Netzwerk-Konfigurationen für den Multiuser-Server....106
2.13.10.1 Voraussetzungen für eine erfolgreiche Client-Server-Kommunikation....106
2.13.10.2 Netzwerk mit dediziertem Multiuser-Server....108
2.13.10.3 Netzwerk mit temporärem Multiuser-Server....109
2.13.10.4 Netzwerk in einer Windows-Domäne....112
2.13.10.5 Netzwerk in einer Windows-Domäne mit Internet-Zugriff über VPN....113
2.13.10.6 Netzwerk in einer Windows-Domäne mit Internet-Zugriff über Proxy....116

2.14 Mit dem Multiuser-Server arbeiten....120
2.14.1 Einführung zum Multiuser-Server....120
2.14.2 Multiuser-Netzwerkprofile einstellen....122
2.14.3 Neue Multiuser-Serververbindung hinzufügen....123
2.14.4 Multiuser-Serververbindung anzeigen....124
2.14.5 Multiuser-Serververbindung bearbeiten....125
2.14.6 Multiuser-Serververbindung wieder herstellen....126
2.14.7 Multiuser-Serververbindung löschen....127

2.15 Mit dem lokalen Multiuser-Server arbeiten....129
2.15.1 Einführung zum lokalen Multiuser-Server....129
2.15.2 Lokalen Multiuser-Server konfigurieren....130
2.15.3 Lokalen Multiuser-Server anzeigen....131
2.15.4 Lokalen Multiuser-Server starten....132
2.15.5 Lokalen Multiuser-Server beenden....133

2.16 Multiuser-Serverprojekte erstellen und verwalten....134
2.16.1 Einführung zum Arbeiten in Multiuser-Serverprojekten....134
2.16.2 Neues Multiuser-Serverprojekt hinzufügen....135

2.17 In der lokalen Session arbeiten....137
2.17.1 Lokale Session erstellen und verwalten....137
2.17.1.1 Lokale Session erstellen....137
2.17.1.2 Lokale Session öffnen....138
2.17.1.3 Lokale Session speichern....141
2.17.1.4 Lokale Session schließen....142
2.17.1.5 Lokale Session löschen....143
2.17.1.6 Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen....144
2.17.1.7 Lokale Session als Single-User-Projekt exportieren....145
2.17.2 Objekte in der lokalen Session bearbeiten....147
2.17.2.1 Einführung zum Arbeiten in lokalen Sessions....147
2.17.2.2 In der lokalen Session offline arbeiten....149
2.17.2.3 Regeln für das Markieren von Objekten....151
2.17.2.4 Objekte für die Bearbeitung automatisch oder manuell markieren....153
2.17.2.5 Mögliche Konflikte beim Automarkieren....156

2.17.2.6 Markierung für Objekte zurücknehmen....157
2.17.2.7 Verwendungsinfo für markierte Objekte anzeigen....158
2.17.2.8 Lokale Session aktualisieren....159
2.17.2.9 Lokale Session nach dem Aktualisieren wieder herstellen....161
2.17.2.10 Objekte ins Serverprojekt einchecken....162
2.17.2.11 Mögliche Konflikte beim Einchecken....165
2.17.2.12 Mögliche Fehlermeldungen beim Einchecken....168
2.17.2.13 Objekte in der lokalen Session löschen....170

2.18 In der Serverprojekt-Ansicht arbeiten....173
2.18.1 Einführung zum Arbeiten in der Serverprojekt-Ansicht....173
2.18.2 Serverprojekt-Ansicht öffnen und schließen....175
2.18.3 Objekte in der Serverprojekt-Ansicht bearbeiten....177
2.18.4 Objekte in der Serverprojekt-Ansicht umbenennen....179

3 Multiuser Commissioning einsetzen....181

3.1 Einführung in Multiuser Commissioning....181

3.2 Übersicht über die Arbeitsweise mit Multiuser Commissioning....183

3.3 Einstellungen für das Arbeiten mit Multiuser Commissioning....186

3.4 Regeln für das Arbeiten mit Multiuser Commissioning....188

3.5 Commissioning-Modus ein- und ausschalten....190

4 Projekte mit PLC-Synchronisierung in Betrieb nehmen....193

4.1 Grundlagen zur PLC-Synchronisierung....193

4.2 Voraussetzungen für die PLC-Synchronisiserung....197

4.3 Vorgehensweise bei der PLC-Synchronisierung....199

4.4 Regeln für die PLC-Synchronisierung....203

5 Daten mit Inter Project Engineering (IPE) austauschen....209

5.1 Grundlagen zu Inter Project Engineering (IPE)....209

5.2 Voraussetzungen für Inter Project Engineering (IPE)....213

5.3 Übersicht für das Arbeiten mit Inter Project Engineering (IPE)....214

5.4 Geräte-Proxy-Daten im Quellprojekt anlegen....217

5.5 IPE-Datei mithilfe des "Geräte-Proxy-Daten" erzeugen....218

5.6 Mit IPE Steuerungsdaten aus anderen Projekten verwenden....219
5.6.1 Steuerungsdaten aus anderen Projekten im Bediengerät verwenden....219
5.6.1.1 Grundlagen zu Steuerungsdaten....219
5.6.1.2 Geräte-Proxy über IPE-Datei initialisieren....219
5.6.1.3 Geräte-Proxy über Projektdatei initialisieren....225
5.6.2 Kommmunikation mit Geräte-Proxys....230
5.6.2.1 Grundlagen zur Kommunikation mit Geräte-Proxys....230
5.6.2.2 Direkte Verbindung projektieren....232
5.6.2.3 Geroutete Verbindung projektieren....234
5.6.3 Integriertes Projektieren mit WinCC und SIMATIC Manager....237
5.6.3.1 Grundlagen zur Projektierung mit WinCC und SIMATIC Manager....237
5.6.3.2 Geräte-Proxy über STEP 7 Projektdatei initialisieren....238
5.6.3.3 Netzwerkparameter anpassen....239

5.6.3.4 Inkonsistente Daten überprüfen....244
5.6.3.5 Variablen und Verbindungen anpassen....245
5.6.3.6 Geräte-Proxy über STEP 7 Projektdatei aktualisieren....245
5.6.3.7 Variablen aus dem SIMATIC Manager einbinden....246
5.6.3.8 Systemfunktion "Direkttasten" konfigurieren....247

Index....251

Übersicht über das Arbeiten mit Team Engineering 1

Einführung

Für das Bearbeiten Ihrer Automatisierungsaufgaben stehen Ihnen im TIA Portal unterschiedliche Funktionalitäten zur Verfügung, die alle unter dem Oberbegriff Team Engineering zusammengefasst sind.

Mit Team Engineering ist ein komfortables Arbeiten in einem Team möglich.

Dies hat den Vorteil, dass Sie mit mehreren Benutzern gemeinsam und gleichzeitig in Projekten arbeiten können. Dadurch werden die Projektierungszeiten für Ihre Projekte wesentlich verkürzt. Sie können Ihre Anlage schneller in Betrieb nehmen und damit auch schneller produktiv arbeiten.

Für folgende Entwicklungsphasen stehen Ihnen verschiedene Funktionen im Rahmen von Team Engineering zu Verfügung:

- Für die Engineering-Phase
- Für die Commissioning-Phase

In der Engineering-Phase entwickeln Sie Ihre Projekte zusammen in einem Team. In dieser Phase wird eine sinnvolle und für das gemeinsame Arbeiten geeignete Projektstruktur erstellt, die alle Anforderungen an das Projekt berücksichtigt. Dazu gehört z. B. eine bereits vollständig projektierte Hardware-Konfiguration und eine sinnvolle Aufteilung des Projekts in einzeln bearbeitbare Programmteile oder Software Units enthält.

In der Commissioning-Phase nehmen Sie Ihr Projekt gemeinsam und parallel in Betrieb. Sie laden Ihre Projektstände auf die CPU und werden dabei von den jeweiligen Team Engineering-Funktionalitäten unterstützt.

Funktionalitäten von Team Engineering

Im TIA Portal gibt es verschiedene Funktionen und Möglichkeiten, gemeinsam und parallel in einem Team zusammen zu arbeiten:

- Multiuser Engineering und Multiuser Commissioning
- Software Units
- PLC-Synchronisierung
- Inter Project Engineering (IPE)

Nachfolgend finden Sie die verschiedenen Möglichkeiten für das Arbeiten mit allen Funktionalitäten von Team Engineering aufgelistet.

Multiuser Engineering und Multiuser Commissioning

Mit Multiuser Engineering können Sie Multiuser-Projekte gemeinsam und gleichzeitig in mehreren lokalen Sessions bearbeiten. Sie erstellen Ihre Projekte offline und laden diese zunächst in den Multiuser-Server. Der Multiuser-Server ermöglicht eine komfortable Projektsynchronisierung und das Durchführen von lokalen Tests.

Mit Multiuser Commissioning nehmen Sie Ihre fertig projektierten Projekte auf komfortable Weise im Team in Betrieb. Auch in der Commissioning-Phase wird nach jedem Laden in die CPU das Multiuser-Serverprojekt aktualisiert und eine neue Revision für das Serverprojekt angelegt.

Dies hat folgende Vorteile:

Beim Arbeiten in den lokalen Sessions erfolgt die Synchronisierung des Multiuser-Projekts über das Serverprojekt.

Der Stand des Projekts in der CPU und auf dem Multiuser-Server sind nach dem Laden in die CPU jederzeit konsistent.

Siehe hierzu auch:

Einführung in Multiuser Engineering (Seite 13)

Einführung in Multiuser Commissioning (Seite 181)

Kombinierte Anwendungsmöglichkeiten:

Die Funktionalität von Software Units und von IPE können auch im Umfeld von Multiuser Engineering und Multiuser Commissioning eingesetzt werden.

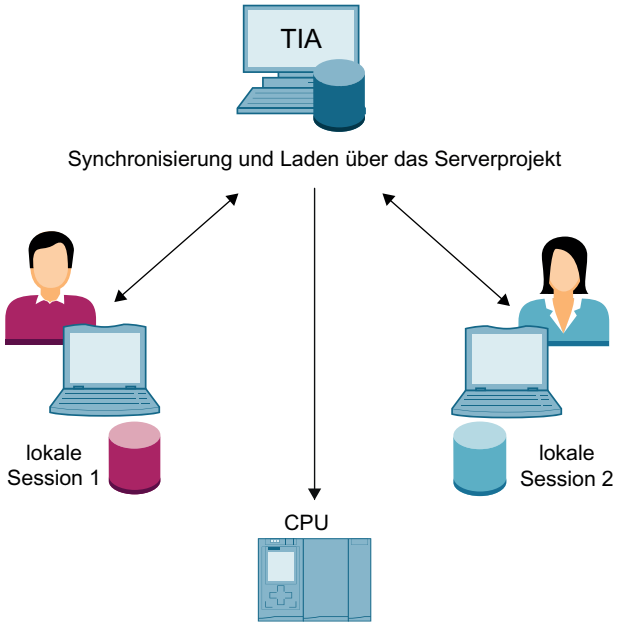

PLC-Synchronisierung

Als Basis für die PLC-Synchronisierung dient ein gemeinsames "Master-Projekt", das bereits die vollständig projektierte Hardware-Konfiguration mit allen benötigten Programmobjekten enthält. Dieses Projekt wird in die gemeinsam genutzte CPU geladen und anschließend mittels Projektkopien an bis zu fünf beteiligte Engineering Systeme verteilt. Die Projektkopien des Master-Projekts können unabhängig voneinander gleichzeitig und parallel offline bearbeitet werden. Bestimmte Online-Funktionen können, abhängig vom Firmewarestand der verwendeten CPU, gleichzeitig und parallel von mehreren Engineering Systemen auf der gemeinsam genutzten CPU ausgeführt werden, wie z. B.: das Beobachten und Steuern von Bausteinen auf der CPU und Tracefunktionen. Es kann immer nur ein ES in die CPU laden. Beim Laden werden die Änderungen der anderen Bearbeiter in einem Dialog für die "Softwaresynchronisierung" angezeigt und - soweit möglich - automatisch synchronisiert. Für konkurrierende Änderungen ist eine manuelle Synchronisation erforderlich. Bei der manuellen Synchronisation Ihrer Änderungen werden Sie durch den Vergleichseditor unterstützt. Siehe hierzu auch: Grundlagen zur PLC-Synchronisierung (Seite 193)	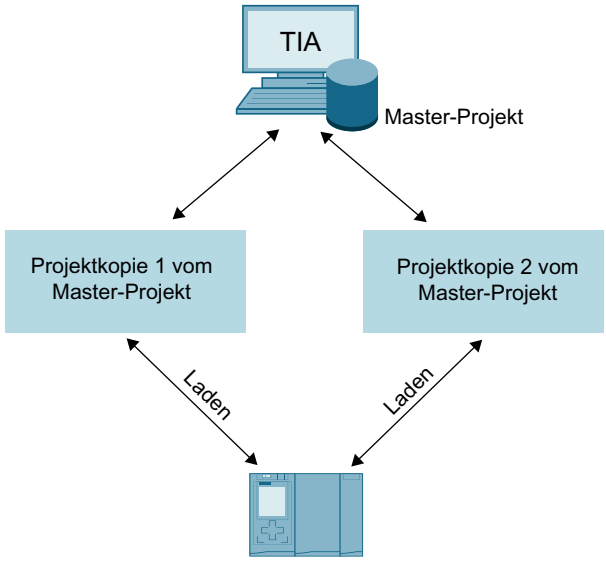
Software Units	

Sie strukturieren Ihr Anwenderprogramm mit Hilfe von Software Units. So können z.B. alle benötigten Programmierelemente für "Motor_1" in einer Software Unit zusammengefasst werden.

Auch hier dient ein dient ein gemeinsames "Master-Projekt" als Basis, das bereits die vollständig projektierte Hardware-Konfiguration mit allen benötigten Software Units enthält.

Dieses Projekt wird in die gemeinsam genutzte CPU geladen und anschließend mittels Projektkopien an bis zu fünf beteiligte Engineering Systeme verteilt, die dann jeweils ausschließlich die Ihnen zugewiesenen Software Units bearbeiten.

Jede Software Unit verfügt über eigene unterlagerte Programmierelemente, wie z. B. eigene Variablentabellen, so dass ein unabhängiges Arbeiten in den einzelnen Units gewährleistet ist.

Die Software Units können unabhängig voneinander gleichzeitig und parallel offline bearbeitet und - unter Einhaltung bestimmter Voraussetzungen - auch ohne Synchronisierungsbedarf in die CPU geladen werden.

Die Integration in das Masterprojekt erfolgt am Ende der Commissioning-Phase, indem jedes ES die von ihm bearbeiteten Software Units wieder in das Masterprojekt zurückkopiert.

Siehe hierzu auch:

AUTOHOTSPOT

Kombinierte Anwendungsmöglichkeiten:

Die Funktionalität von Software Units kann auch mit IPE und mit Multiuser Engineering und Commissioning eingesetzt werden.

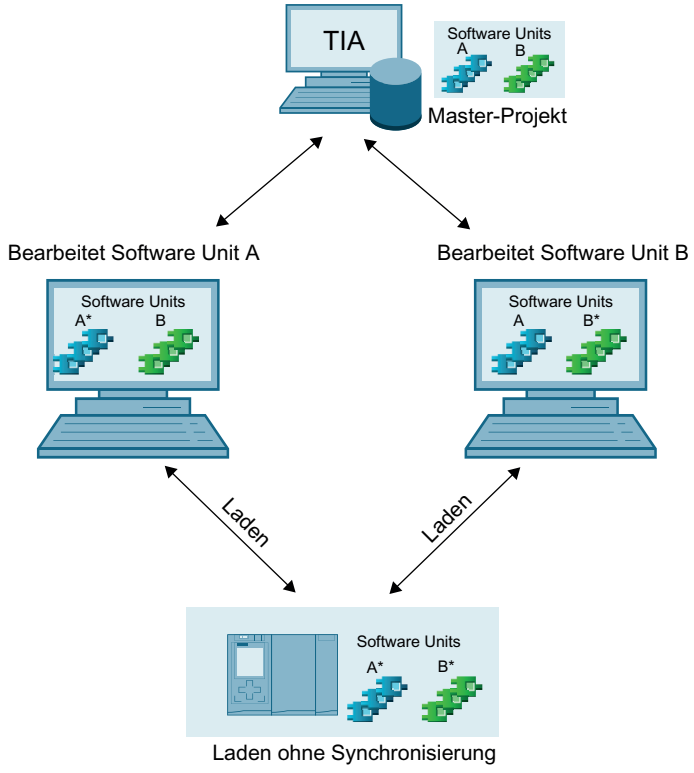

Inter Project Engineering (IPE)	
Mit der Funktionalität von Inter Project Engineering, nachfolgend IPE genannt, können Sie in einem Quellprojekt vorhandene Steuerungsdaten mit Hilfe eines Geräte-Proxy projektübergreifend austauschen. Diese Daten können Sie in anderen Projekte, z. B. für die Visualisierung in HMI, übernehmen und dort zur weiteren Projektierung verwenden. Inter Project Engineering ermöglicht Ihnen somit auf einfache Art und Weise projektübergreifend Steuerungsdaten von einer PLC-Programmierung für eine HMI-Projektierung zur Verfügung zu stellen. Siehe hierzu auch: AUTOHOTSPOT **Kombinierte Anwendungsmöglichkeiten:** Die Funktionalität von IPE kann sowohl mit der PLC-Snchronisierung als auch mit Software Units und mit Multiuser Engineering und Commissioning eingesetzt werden.	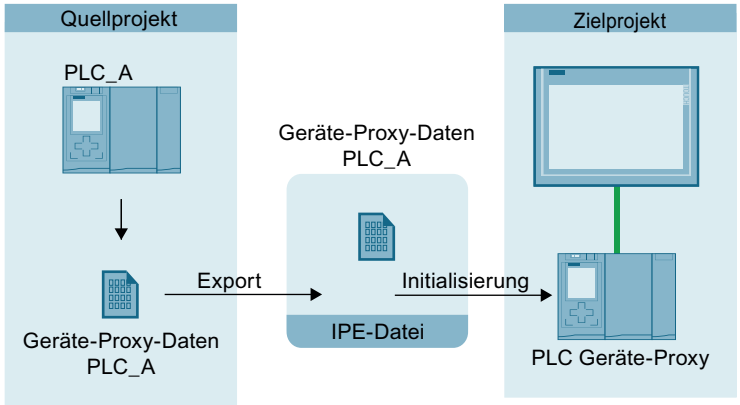

Siehe auch

Einführung in Multiuser Engineering (Seite 13)

Einführung in Multiuser Commissioning (Seite 181)

Grundlagen zur PLC-Synchronisierung (Seite 193)

Multiuser Engineering einsetzen 2

2.1 Einführung in Multiuser Engineering

Einführung

Im TIA Portal haben Sie die Möglichkeit, mit Multiuser Engineering zu arbeiten.

Dies hat den Vorteil, dass Sie mit mehreren Benutzern gemeinsam und gleichzeitig in Multiuser-Projekten arbeiten können. Dadurch werden die Projektierungszeiten für Ihre Projekte wesentlich verkürzt. Sie können Ihre Anlage schneller in Betrieb nehmen und produktiv damit arbeiten.

Sie können in Multiuser Engineering auch die anderen Funktionalitäten von Team Engineering nutzen, wie das gemeinsame Arbeiten auf einer CPU und den Datenaustausch mit Inter Project Engineering (IPE).

Und so funktioniert Multiuser Engineering im TIA Portal:

1. Die Projektverwaltung liegt auf einem lokalen oder externen Server.
2. Verschiedene Anwender arbeiten in lokalen Sessions, basierend auf den vom Server verwalteten Projekten.
3. Das Arbeiten in lokalen Sessions ist unabhängig voneinander.
4. Die Änderungen aus den lokalen Sessions werden durch das Einchecken ins Serverprojekt übertragen.
5. Eingecheckte Änderungen von anderen Bearbeitern werden angezeigt und können komfortabel übernommen werden.

Hinweis

Effizientes Arbeiten mit Multiuser Engineering

Beim parallelen Arbeiten in mehreren lokalen Sessions, sollten Sie auftretende Fehler frühzeitig beheben, damit das verteilte Arbeiten an einem gemeinsamen Projekt effektiv bleibt.

Daher ist es empfehlenswert, die anstehenden Änderungen vor dem Einchecken im Multiuser-Editor zuerst zu übersetzen. Dazu stehen Ihnen mehrere Übersetzungsoptionen zur Verfügung. Falls Übersetzungsfehler angezeigt werden, sollten Sie diese vor dem Einchecken beseitigen.

Durch das Übersetzen der anstehenden Änderungen vor dem Einchecken stellen Sie sicher, dass Sie ein fehlerfreies Projekt einchecken und an andere Bearbeiter übergeben.

Für das gleichzeitige Arbeiten in einem Multiuser-Projekt stehen Ihnen verschiedene Möglichkeiten zur Verfügung, wie in der nachfolgenden Tabelle gezeigt:

Anwendungsszenarien für Multiuser Engineering	
Szenario 1: Mehrere Projekteure arbeiten *geräteübergreifend* an unterschiedlichen Objekten. Dabei bearbeitet jeder Projekteur die ihm zugewiesenen Objekte auf verschiedenen Geräten.	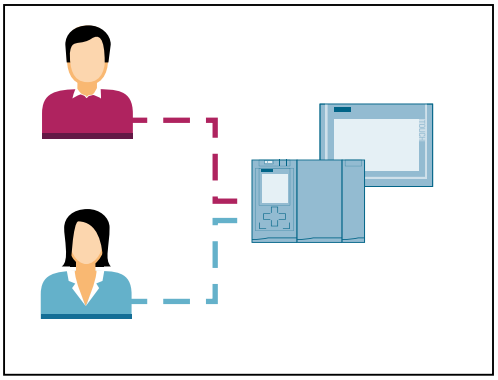
Szenario 2: Mehrere Projekteure arbeiten *geräteorientiert* an unterschiedlichen Objekten. Dabei bearbeitet jeder Projekteur die ihm zugewiesenen Objekte auf nur einem Gerät.	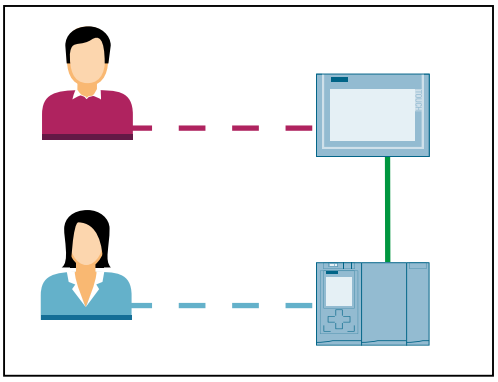
Szenario 3: Mehrere Projekteure arbeiten *technologieorientiert* an unterschiedlichen Objekten. Dabei bearbeitet jeder Projekteur definierte technologische Objekte (z. B. Motor_1 oder Motor_2) auf allen vorhandenen Geräten.	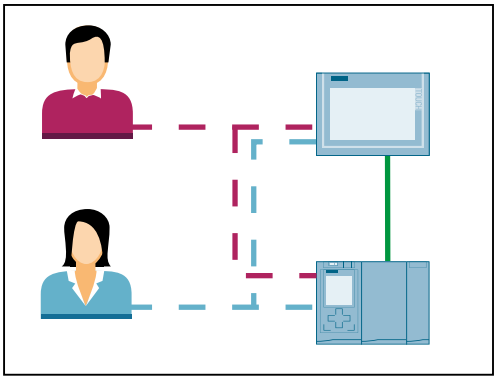

Durch das parallele Bearbeiten von verschiedenen Objekten, wie z. B. unterschiedlichen Variablen, Bausteinen, PLC-Variablen, Alarmen, Bildern, Text- und Grafiklisten, Technologieobjekten usw. in einem oder in mehreren Geräten innerhalb eines Multiuser-Projektes, können Sie Ihre Projektierungszeiten effektiver gestalten und wesentlich verkürzen.

Siehe auch

Übersicht über die Arbeitsweise mit Multiuser Engineering (Seite 16)
Grundlagen zum Multiuser-Serverkonzept (Seite 21)
Voraussetzungen für das Arbeiten mit Multiuser Engineering (Seite 24)
Unterstützte Objekte für Multiuser Engineering (Seite 27)
Einführung in Multiuser Commissioning (Seite 181)

2.2 Übersicht über die Arbeitsweise mit Multiuser Engineering

Einführung

Wenn Sie mit mehreren Benutzern gemeinsam und gleichzeitig in Multiuser-Projekten arbeiten, sollten bestimmte Vorgehensweisen eingehalten werden.

Nachfolgend erhalten Sie einen Überblick über die generelle Arbeitsweise mit Multiuser Engineering.

Anwendungsbeispiel

Ein Anwendungsbeispiel für das Arbeiten mit Multiuser Engineering finden Sie unter folgendem Link:

https://support.industry.siemens.com/cs/ww/de/view/109740141 (https://support.industry.siemens.com/cs/ww/de/view/109740141)

Regeln für das Arbeiten mit Multiuser Engineering

Beachten Sie für das Arbeiten in Multiuser-Projekten unbedingt die nachfolgenden Regeln:

- Die zu bearbeitenden Projekte sollten bereits die gesamte Hardware-Konfiguration, die benötigten Bausteine, Verbindungen und globalen Einstellungen enthalten, bevor sie als Multiuser-Projekt verwendet werden können.
  Weiterführende Informationen dazu finden Sie unter: Anforderungen an Multiuser-Projekte (Seite 35)
- Vermeiden Sie das gleichzeitige Arbeiten an denselben Objekten durch rechtzeitige Absprachen mit den anderen Bearbeitern.
- Alle globalen Einstellungen, wie z. B. Änderungen innerhalb der Gerätekonfiguration, müssen Sie in der Serverprojekt-Ansicht (also im zentralen Serverprojekt) vornehmen.
- In der Serverprojekt-Ansicht können alle Objekte eines Multiuser-Projektes bearbeitet werden. Sie können auch die Objekte bearbeiten, die die Funktionalität von Multiuser Engineering nicht unterstützen.

Übersicht über die Arbeitsweise mit Multiuser Engineering

Um die Funktionalität von Multiuser Engineering zu nutzen, führen Sie folgende Schritte durch:

Aktion	Beispiel
1. Multiuser-Projekt auf dem Server anlegen Sie installieren einen Multiuser-Server und laden ein Single-User-Projekt auf den verwendeten Multiuser-Server hoch. • Das Projekt sollte so strukturiert sein, dass es die Anforderungen an Multiuser-Projekte erfüllt. **Ergebnis:** Durch das Hochladen auf den Multiuser-Server erzeugen Sie aus dem Single-User-Projekt ein Multiuser-Serverprojekt.	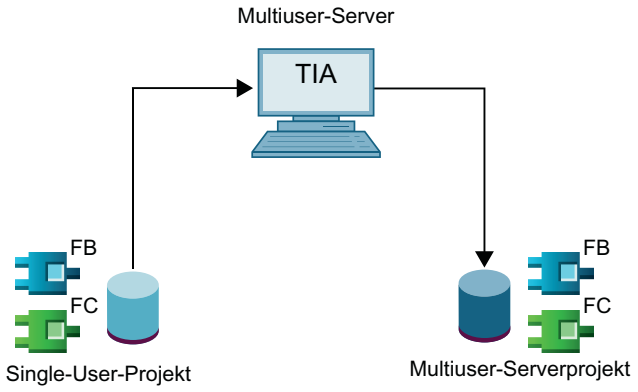
2. Lokale Session für "Bearbeiter 1" anlegen Sie erstellen eine "lokale Session" als Kopie des Serverprojektes auf Ihrem Rechner. • Für jeden Bearbeiter werden in seiner lokalen Session die Objekte, die er bearbeitet, automatisch mit blauen Fähnchen markiert. • In den verschiedenen lokalen Sessions können unterschiedliche Objekte gleichzeitig durch mehrere Bearbeiter bearbeitet werden. **Ergebnis:** In der eigenen lokalen Session bearbeitete Objekte werden mit einem blauen Fähnchen gekennzeichnet. Die Fähnchen werden auch in der Projektnavigation und in den Editoren des TIA Portals angezeigt.	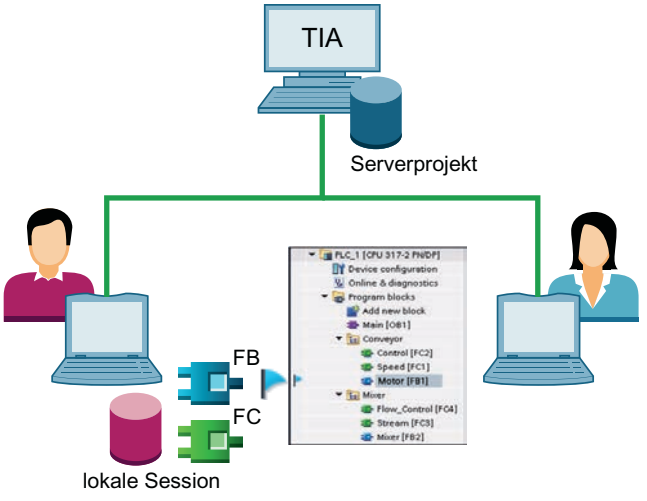

Aktion	Beispiel
3. Lokale Session für "Bearbeiter 2" anlegen Sie erstellen eine zweite lokale Session, damit auch der zweite Bearbeiter seine Aufgabe in einer eigenen lokalen Session bearbeiten kann. • Auch beim zweiten Bearbeiter werden in seiner lokalen Session die Objekte, die er bearbeitet, automatisch mit blauen Fähnchen markiert. **Ergebnis:** In der eigenen lokalen Session markierte Objekte werden mit einem blauen Fähnchen gekennzeichnet. In einer anderen lokalen Session markierte Objekte werden in der eigenen lokalen Session automatisch mit einem gelben Fähnchen gekennzeichnet. Jeder Bearbeiter sieht also, welche Objekte von ihm selbst oder von anderen bearbeitet werden.	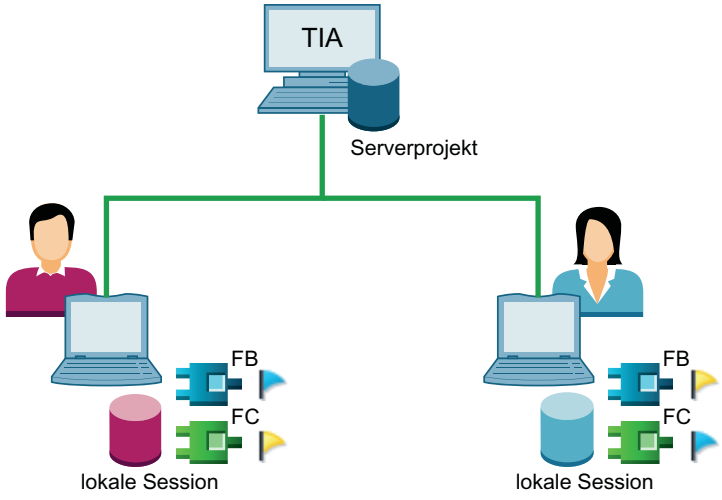
4. Einchecken der Änderungen durch "Bearbeiter 1". Sobald das Arbeiten in der lokalen Session einen gewissen Arbeitsstand erreicht hat, checken Sie die Änderungen ins Multiuser-Serverprojekt ein. • Durch das Einchecken werden die Änderungen an den markierten Objekten ins Multiuser-Serverprojekt übernommen. • Nach dem Einchecken wird dem "Bearbeiter 2" in seiner lokalen Session angezeigt, dass ein Objekt "veraltet" ist. **Ergebnis:** Die Änderungen von "Bearbeiter 1" werden ins Serverprojekt transferiert. "Bearbeiter 2" kann seine lokale Session mit den neuen Serverdaten aktualisieren, falls gewünscht.	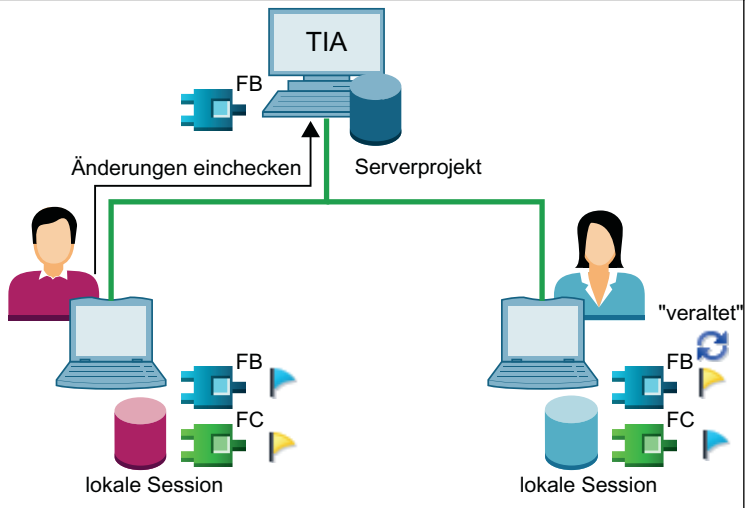

Aktion	Beispiel
5. "Bearbeiter 2" aktualisiert seine lokale Session "Bearbeiter 2" bekommt in seiner lokalen Session angezeigt, dass ein Objekt in einem neueren Stand im Serverprojekt vorhanden ist und aktualisiert seine lokale Session. • Durch das Aktualisieren wird der neuere Stand vom Serverprojekt übernommen und in der eigenen lokalen Session eingefügt. • Alle nicht markierten Objekte werden durch den aktuellen Serverstand ersetzt, alle markierten Objekte bleiben erhalten. **Ergebnis:** Die Änderungen vom Serverprojekt werden in die lokale Session von "Bearbeiter 2" transferiert.	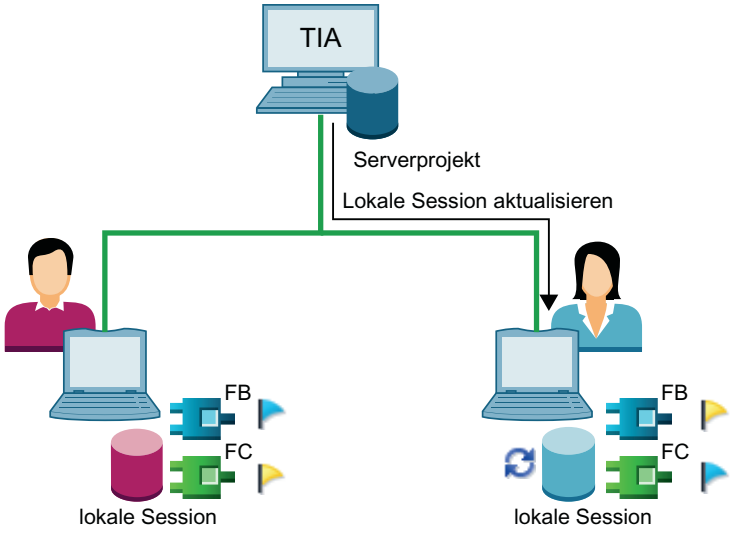
6. "Bearbeiter 2" checkt Änderungen ins Serverprojekt ein "Bearbeiter 2" hat seine Änderungen in der lokalen Session fertig gestellt und checkt seine blau markierten Objekte ins Multiuser-Serverprojekt ein. • Durch das Einchecken werden die Änderungen an den markierten Objekten ins Multiuser-Serverprojekt übernommen. • Nach dem Einchecken wird dem "Bearbeiter 1" in seiner lokalen Session angezeigt, dass ein Objekt "veraltet" ist. **Ergebnis:** Die Änderungen von "Bearbeiter 2" werden ins Serverprojekt transferiert. "Bearbeiter 1" kann seine lokale Session mit den neuen Serverdaten aktualisieren, falls gewünscht.	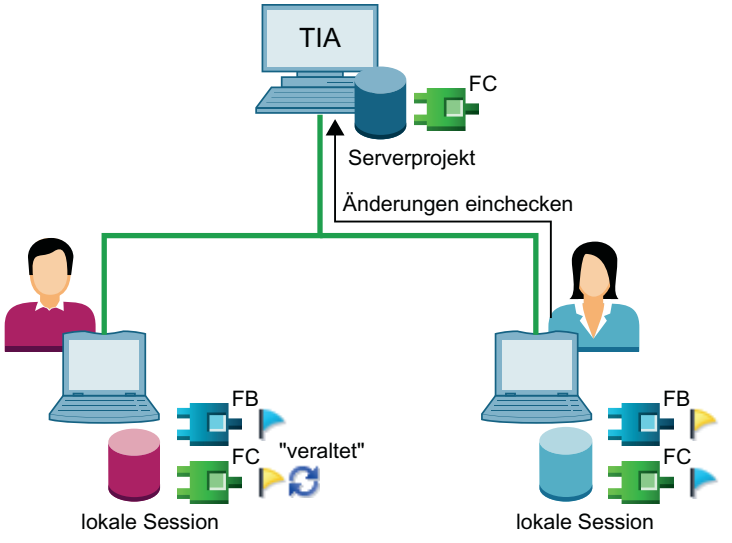

Aktion	Beispiel
7. Bearbeitung ist abgeschlossen Die Bearbeitung der unterschiedlichen Projektierungsaufgaben in den lokalen Sessions ist abgeschlossen. **Ergebnis:** Alle Änderungen sind ins Serverprojekt eingecheckt. Die lokalen Sessions und das Serverprojekt sind synchronisiert. In den lokalen Sessions sind keine Markierungen mehr vorhanden. Das Multiuser-Serverprojekt kann als Single-User-Projekt gespeichert und wie gewohnt im TIA Portal weiter verwendet werden. **Hinweis:** Falls gewünscht, können Sie lokale Sessions als Single-User-Projekt exportieren und wie gewohnt im TIA Portal weiter damit arbeiten.	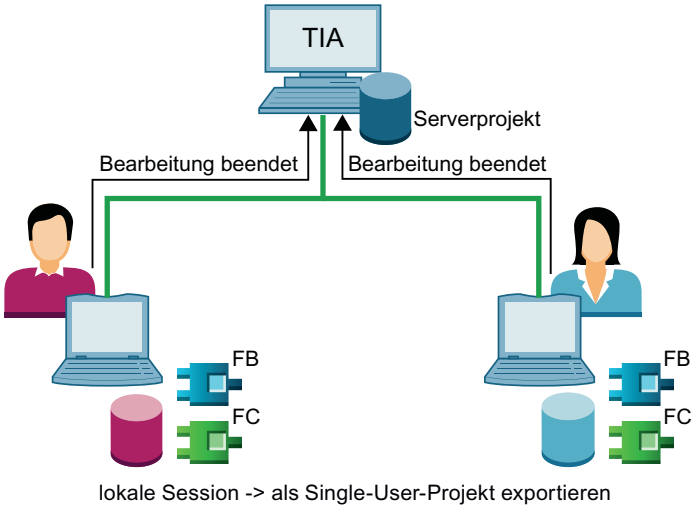

Siehe auch

Voraussetzungen für das Arbeiten mit Multiuser Engineering (Seite 24)
Grundlagen zum Multiuser-Serverkonzept (Seite 21)
Unterstützte Objekte für Multiuser Engineering (Seite 27)
Anforderungen an Multiuser-Projekte (Seite 35)
Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.3 Grundlagen zum Multiuser-Serverkonzept

Einführung

Um mit Multiuser Engineering arbeiten zu können, benötigen Sie einen Multiuser-Server, der Ihre Multiuser-Projekte und die lokalen Sessions verwaltet.

Jede Version des TIA Portals beinhaltet einen versionsspezifischen Multiuser-Server mit entsprechender Funktionalität.

Mit Hilfe des Multiuser-Servers können Sie:

- Neue Multiuser-Serverprojekte anlegen, in denen mehrere Anwender parallel arbeiten können.
- Bereits vorhandene Serverprojekte anzeigen und verwalten.
- Lokale Sessions erzeugen und verwalten.

Kompatibilität des Multiuser-Servers mit dem TIA Portal

Es können verschiedene Versionen des Multiuser-Servers nebeneinander auf einem Gerät installiert werden und gleichzeitig laufen.

Die Kommunikation zwischen dem Multiuser-Server und dem TIA Portal wird für verschiedene Anwendungsfälle unterstützt.

Siehe hierzu: Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

Voraussetzung

Um die Funktionalität des Multiuser-Servers nutzen zu können, müssen Sie die passende Software für den Multiuser-Server installiert haben oder Sie nutzen den per Voreinstellung mitinstallierten lokalen Multiuser-Server.

Mögliche Server-Konstellationen

Im TIA Portal stehen Ihnen verschiedene Server-Konstellationen für die Funktionalität von Multiuser Engineering zur Verfügung:

1. Der "Lokale Multiuser-Server", der im TIA Portal automatisch mitinstalliert wird.
2. Der so genannte "Temporäre Multiuser-Server", bei dem die Serverfunktionalität auf einem Arbeitsplatzrechner realisiert wird.
3. Der Multiuser-Server für eine langfristige Zusammenarbeit, der als dezidierter externer Server realisiert wird.

In der nachfolgenden Tabelle finden Sie eine Übersicht über das Arbeiten mit den möglichen Server-Konstellationen.

Übersicht der möglichen Server-Konstellationen für Multiuser Engineering	
1. Multiuser-Server als dezidierter Server • Der Multiuser-Server kann mit dem TIA Portal oder standalone installiert werden. • Diese Konfiguration ist die üblicherweise verwendete "Büro-Netzwerk-Konfiguration". • Der Multiuser-Server läuft auf einem eigenen Rechner. • Diese Server-Konstellation ist durchgängig erreichbar und ermöglicht standardmäßig ein durchgängiges Arbeiten "rund um die Uhr". • Der Multiuser-Server muss installiert und konfiguriert werden.	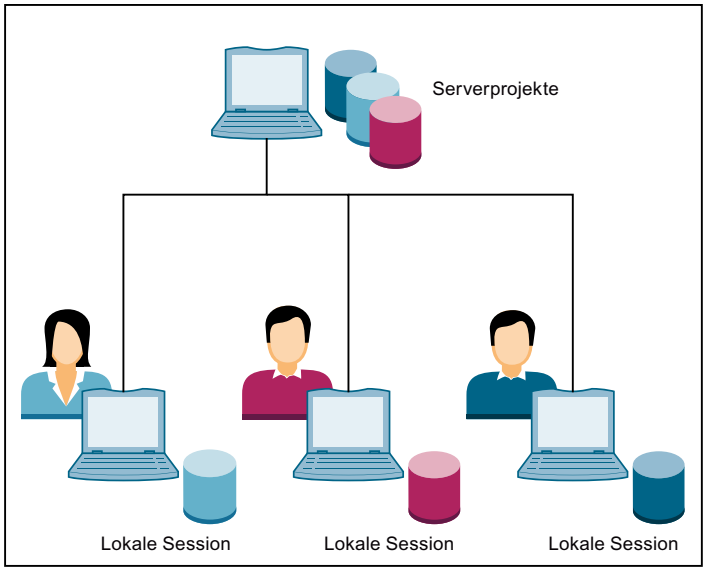
2. Temporärer Multiuser-Server: • Ein Bearbeiter des Multiuser-Projektes kann die Server-Funktion auf seinem Rechner übernehmen. • Diese Konfiguration ist ebenfalls eine mögliche "Büro-Konfiguration". • Der temporäre Multiuser-Server muss installiert und konfiguriert werden.	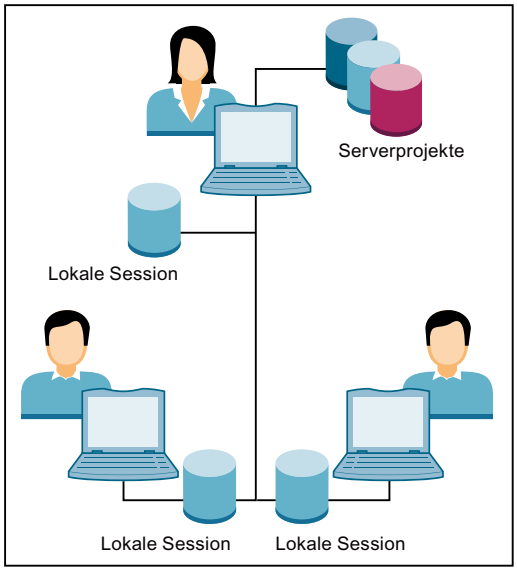

Übersicht der möglichen Server-Konstellationen für Multiuser Engineering	
3. Lokaler Multiuser-Server: • Der lokale Multiuser-Server läuft auf dem Rechner des jeweiligen Bearbeiters. • Er ist ohne weitere Installation und Konfiguration sofort verfügbar, da er mit dem TIA Portal mitinstalliert wird. • Es erfolgt keine Multiuser-Lizenzabfrage für den lokalen Multiuser-Server. • Auf den lokalen Multiuser-Server ist kein Zugriff über ein Netzwerk von anderen Rechnern aus möglich. • Der lokale Multiuser-Server ist zum Kennenlernen und Testen der Funktionalität von Multiuser Engineering geeignet, da keine Server-Installation und keine Konfiguration notwendig ist. • Der lokale Multiuser-Server kann auch für eine eigene Projektverwaltung mit "Rollback"-Funktion eingesetzt werden.	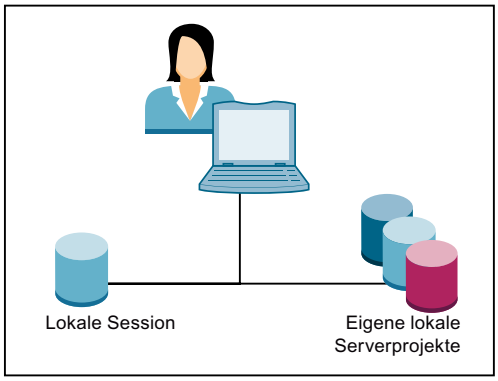

Siehe auch

Übersicht über die Arbeitsweise mit Multiuser Engineering (Seite 16)
Voraussetzungen für das Arbeiten mit Multiuser Engineering (Seite 24)
Unterstützte Objekte für Multiuser Engineering (Seite 27)
Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

2.4 Voraussetzungen für das Arbeiten mit Multiuser Engineering

Systemvoraussetzungen

Für das Arbeiten mit Multiuser Engineering gelten die gleichen Hard- und Software-Voraussetzungen wie für das Arbeiten mit dem TIA Portal.

Informieren Sie sich über die geltenden Bestimmungen für die von Ihnen installierten Produkte unter "Installation" > "Systemvoraussetzungen für die Installation".

Hinweis

Identische Software-Installation als Voraussetzung für das Arbeiten mit Multiuser Engineering

Um ohne Einschränkungen mit Multiuser Engineering arbeiten zu können, ist es wichtig, dass auf allen verwendeten Engineering Systemen die gleichen Software-Produkte des TIA Portals mit identischen Ausgabeständen installiert sind.

Hinweis

Zugriff auf den Multiuser-Server

Benutzer können nur dann Zugriff auf den Multiuser-Server erhalten, wenn diese **nicht** in der lokalen Administrator-Gruppe des Multiuser-Servers eingetragen sind.

Hinweis

Zugriff auf das Multiuser-Serverprojekt

Benutzer können nur dann Zugriff auf das Multiuser-Serverprojekt erhalten, wenn diese in der Benutzerverwaltung des Multiuser-Servers zumindest als "Member" eingetragen sind.

Multiuser-Server

Um mit der vollen Funktionalität von Multiuser Engineering arbeiten zu können, benötigen Sie einen installierten Multiuser-Server.

Für die Installation, Konfiguration und Verwaltung des Multiuser-Servers benötigen Sie teilweise Administrator-Rechte.

Informationen zur Kompatibilität des Multiuser-Servers finden Sie unter:Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

Die Konfiguration und Verwaltung des Multiuser-Servers wird über die mitgelieferten Multiuser-Tools ausgeführt.

Dazu stehen Ihnen zwei alternative Vorgehensweisen zur Verfügung:

- Verwendung der grafische Bedienoberfläche
- Verwendung der Multiuser Administrative Tools und Power Tools, die über die Kommandozeile ausgeführt werden.

Sie können die Konfiguration und Verwaltung des Multiuser-Servers sowohl über eine komfortable Bedienoberfläche als auch über die Kommandozeilen-Befehle durchführen. Beide Vorgehensweisen führen zu einem identischen Ergebnis.

Eine ausführliche Beschreibung zur Vorgehensweise finden Sie unter "Multiuser-Server installieren (Seite 49)" und unter "Multiuser-Server konfigurieren und verwalten (Seite 57)"

Multiuser-Netzwerkprofile

Sie haben die Möglichkeit, in den Einstellungen zum TIA Portal die gewünschte Leistungsfähigkeit für das von Ihnen verwendete Netzwerkprofil einzustellen.

Wählen Sie dazu im TIA Portal unter "Extras" > "Einstellungen" > "Multiuser" > "Multiuser-Netzwerkprofile" das gewünschte Profil aus der Klappliste aus.

Eine ausführliche Beschreibung der Netzwerkprofile finden Sie unter: Multiuser-Netzwerkprofile einstellen (Seite 122)

Lizenzierung für Multiuser Engineering

Für das Arbeiten mit Multiuser Engineering benötigen Sie nur dann eine versionsgebundene Lizenz (Floating License) und einen zugehörigen License Key, wenn Sie mit einem Multiuser-Server arbeiten.

Solange Sie den nur mit eingeschränkter Funktionalität verfügbaren lokalen Multiuser-Server nutzen, benötigen Sie keine zusätzliche Lizenz zum installierten TIA Portal.

Sie können den License Key entweder bei der Installation des TIA Portals gleich mitinstallieren oder nach der Installation mit dem Automation License Manager übertragen. Multiuser Engineering prüft das Vorhandensein eines gültigen License Keys, sobald Sie eine lokale Session öffnen, um Objekte zu bearbeiten.

Wird ein gültiger License Key gefunden, so wird die lokale Session geöffnet und Sie können ohne Einschränkungen in der lokalen Session arbeiten.

Wird kein gültiger License Key gefunden, erhalten Sie eine Fehlermeldung.

Sie können übergangsweise eine "Trial"-Lizenz aktivieren, um in einer lokalen Session arbeiten zu können.

Nach Ablauf der "Trial"-Lizenz benötigen Sie eine gültige Multiuser-Lizenz, um in lokalen Sessions arbeiten zu können.

Für folgende Aktionen im Kontext von Multiuser Engineering wird keine Lizenz benötigt:

- Multiuser-Server konfigurieren und starten
- Multiuser-Projektverwaltung öffnen
- Projekte zum Multiuser-Server hinzufügen
- Lokale Sessions erstellen
- Lokale Sessions löschen
- Lokale Sessions als Single-User-Projekte exportieren

Hinweis

Umgang mit Lizenzen

Beachten Sie unbedingt die allgemein gültigen Hinweise zum Umgang mit Lizenzen im TIA Portal. Sie finden diese Informationen im Kapitel "Installation" bei den jeweiligen Produkten im Informationssystem beschrieben.

Archivieren und Dearchivieren von Projekten

Das Archivieren und Dearchivieren von Lokalen Sessions wird von Multiuser Engineering nicht unterstützt.

Das Archivieren oder Dearchivieren von Multiuser-Projekten führt zu ungültigen lokalen Sessions, da die Anbindung an den Multiuser-Server durch diese Aktionen unterbrochen wird.

Lokale Session als Referenzprojekt öffnen

Beim Arbeiten mit Multiuser Engineering ist es nicht möglich, eine lokale Session als Referenzprojekt zu öffnen. Damit soll vermieden werden, dass eine lokale Session gleichzeitig mehrfach geöffnet und dadurch ungültig wird.

Länge der Pfad- und Dateinamen bei Multiuser Engineering

Haben die Pfad- und Dateinamen insgesamt mehr als 259 Zeichen, erhalten Sie beim Arbeiten mit Multiuser Engineering eine Fehlermeldung. Sie werden darauf hingewiesen, dass die verwendeten Namen zu lang sind und die gewünschte Aktion aus diesem Grund nicht durchgeführt werden kann. Kürzen Sie in einem solchen Fall die Pfad- bzw. Dateinamen, um die gewünschte Aktion durchführen zu können.

Siehe auch

Multiuser-Netzwerkprofile einstellen (Seite 122)

Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

2.5 Unterstützte Objekte für Multiuser Engineering

Einführung

Die Funktionalität von Multiuser Engineering steht Ihnen im TIA Portal für die nachfolgend aufgelisteten Objekte zur Verfügung.

Objekte, die diese Funktionalität noch nicht unterstützen, können innerhalb der lokalen Session nicht markiert werden.

Deshalb werden sie beim Einchecken **nicht** ins Serverprojekt übernommen. Die Änderungen an solchen Objekten werden nach dem Einchecken und Aktualisieren der lokalen Session wieder mit den Inhalten des Serverprojekts überschrieben und gehen verloren.

Solche nicht markierbaren Objekte können Sie in der Serverprojekt-Ansicht bearbeiten.

Auch Versionsänderungen an Anweisungen sollten Sie in der Serverprojekt-Ansicht durchführen.

Das Arbeiten mit der Funktionalität von Team Engineering, wie z. B. das Arbeiten auf einer gemeinsamen CPU und der Datenaustausch über Inter Project Engineering (IPE), ist in Multiuser Engineering ebenfalls möglich.

Von STEP 7 unterstützte Objekte für Multiuser Engineering

Die nachfolgend genannten Objekte für das Produkt STEP 7 werden unterstützt:

- Bausteine:
  OB, FB, FC, Globale DBs und Instanz-DBs
- Software-Units:
  Enthaltene Bausteine wie z.B. OB, FB, FC, Globale DBs und Instanz-DBs.
  Relationen können nur in der Serverprojekt-Ansicht bearbeitet werden.
- PLC-Datentypen und Konstanten:
  Mit Ausnahme der vom System automatisch generierten Systemdatentypen.
- Variablen:
  Mit Ausnahme von Systemvariablen, die vom System selbst erstellt und gelöscht werden.
- Alarme:
  Mit Ausnahme von Systemalarmen.
  Programmalarme können nur zusammen mit dem zugehörigen FB oder DB für die Bearbeitung markiert werden.
- Alarm-Klassen:
  Mit Ausnahme von System-Alarmklassen.
- Textlisten:
  Mit Ausnahme von Systemtextlisten.
- Beobachtungs- und Forcetabellen
- Bibliothekselemente der Projektbibliotheken

- Technologieobjekte:
  Mit Ausnahme der Technologieobjekte für Motion Control auf S7-1500 CPUs.
- Trace-Funktionalität:
  Mit Ausnahme von Online-Konfigurationen.

Von WinCC unterstützte Objekte für Multiuser Engineering

Die nachfolgend genannten Objekte für das Produkt WinCC werden unterstützt:

- Bilder:
  Mit Ausnahme von Globalem Bild, Übersichtsbild, Slide-in-Bildern und globalen Filtern der Meldeanzeige.
- Skripte:
  Mit Ausnahme der Header-Datei "GlobalDefinitions.h".
- Variablen:
  Mit Ausnahme von Systemvariablen, Rezepturvariablen, Archivvariablen und vom System generierten Rohdatenvariablen.
- Meldungen:
  Mit Ausnahme von Systemmeldungen, Steuerungsmeldungen, vordefinierten System-Meldeklassen (auch "Safety Warnings" für KTP700F, KTP900F), System-Klassengruppen und System-Meldegruppen sowie Meldeblöcken.
- Text-und Grafiklisten und C-Textlisten-Einträge

Von OPC UA unterstütze Objekte für Multiuser Engineering

Die nachfolgend genannten Objekte für die OPC UA-Kommunikation werden unterstützt:

- Server-Schnittstellen und Client-Schnittstellen

ACHTUNG
Bearbeiten von nicht unterstützen Objekten in der lokalen Session Nicht unterstützte Objekte können innerhalb einer lokalen Session zwar bearbeitet aber nicht markiert werden. Änderungen an nicht unterstützten Objekten werden beim Einchecken **nicht** ins Serverprojekt übernommen! Änderungen an solchen Objekten gehen nach dem Einchecken und nach einem Refresh in der lokalen Session verloren, da sie wieder mit den Inhalten des Serverprojektes überschrieben werden. Sie können solche nicht unterstützen Objekte jedoch wie gewohnt in der Serverprojekt-Ansicht bearbeiten.

2.6 Einstellungen für das Arbeiten mit Multiuser Engineering

Einführung

Für das Arbeiten mit Multiuser Engineering können Sie im TIA Portal unter "Extras > Einstellungen" im Register "Multiuser" folgende multiuser-spezifische Einstellungen vornehmen:

- Allgemein
- Multiuser-Serververbindung
- Multiuser-Netzwerkprofile und Kompression
- Suche in Projekt
- Speichereinstellungen
- Übersetzungseinstellungen

Die einzelnen Einstellmöglichkeiten sind nachfolgend beschrieben.

Allgemein

In der Einstellung "Allgemein" können Sie auswählen, ob die in der lokalen Session geöffneten Editoren beim Öffnen der Serverprojekt-Ansicht verborgen werden sollen.

Diese Option ist per Voreinstellung deaktiviert.

Multiuser-Serververbindung

Dieser Dialog ermöglicht das Hinzufügen neuer Serververbindungen für einen ausgewählten Multiuser-Server.

- Klicken Sie auf die Schaltfläche "Serververbindung hinzufügen", um diese Verbindung zu erstellen.
- Tragen Sie im nachfolgenden Dialog unter "Server-Name" den Server-Namen sowie den zugehörigen Host und Port ein.
- Wählen Sie das gewünschte Übertragungsprotokoll für die Serververbindung aus. Per Voreinstellung ist "http" als Übertragungsprotokoll ausgewählt.
- Klicken Sie auf die Schaltfläche "Hinzufügen".

Die neue Multiuser-Serververbindung wird hinzugefügt und angezeigt.

Siehe hierzu auch Einführung zum Multiuser-Server (Seite 120).

Multiuser-Netzwerkprofile und Kompression

Sie haben die Möglichkeit, aus der Klappliste das gewünschte Netzwerkprofil auszuwählen, das zur Leistungsfähigkeit des von Ihnen verwendeten Netzwerks passt.

Das ausgewählte Multiuser-Netzwerkprofil bestimmt die Zeit, bis es zu einer Zeitüberschreitungsmeldung kommt.

Eine ausführliche Beschreibung der einzelnen Netzwerkprofile finden Sie unter Multiuser-Netzwerkprofile einstellen (Seite 122).

Falls als Netzwerkprofil "Mittel" oder "Langsam" eingestellt ist, wird die Komprimierung für die Datenübertragung aktiviert. Bei WLAN oder bei mittleren Datenübertragungsgeschwindigkeiten (Fast Ethernet) wird durch die Komprimierung eine verbesserte Übertragungszeit erreicht.

Für das Netzwerkprofil "Schnell" wird keine Komprimierung durchgeführt.

Hinweis

Die Umstellung des Netzwerkprofils wird erst nach einem Neustart des Multiuser-Clients oder des Multiuser-Servers wirksam.

Suche in Projekt

Damit die globale Suche auch in lokalen Sessions funktioniert, müssen die Daten für die Suche indiziert werden. Diese Option ist per Voreinstellung aktiviert, damit die Indizierung für die Suche immer während des Eincheckens durchgeführt wird.

Dies ist die empfohlene Einstellung.

Wenn Sie diese Option deaktivieren, wird die Indizierung nicht während des Eincheckens, sondern zu anderen Zeitpunkten durchgeführt.

Die Indizierungsdaten werden dann zunächst als Datei an den Multiuser-Server übergeben. Eine Suche in der lokalen Session ist jedoch erst nach einer durchgeführten Indizierung möglich. Falls beim Einchecken keine Indizierung durchgeführt wurde, kann dies etwas längere Wartezeiten bei der Suche ergeben und zur Verlängerung der Eincheck-Zeiten bei anderen Bearbeitern führen, die diese Option aktiviert haben.

Speichereinstellungen

Hier können Sie festlegen, wohin Ihre lokalen Sessions und Ihre Multiuser-Serverprojekte gespeichert werden sollen.

- Momentan verwendeter Speicherort
  Diese Option ist per Voreinstellung aktiviert.
- Festlegen eines Speicherorts per Voreinstellung
  Wählen Sie hier jeweils den gewünschten Speicherort aus, indem Sie an die gewünschte Stelle navigieren. Klicken Sie dazu auf die Schaltfläche "Durchsuchen". Ihre Einstellungen werden als Voreinstellung übernommen.

Übersetzungseinstellungen

Legen Sie hier die gewünschten Übersetzungseinstellungen für Ihre lokalen Sessions fest. Die hier vorgenommenen Einstellungen werden als Voreinstellungen im Multiuser-Editor übernommen und können dort nicht mehr verändert werden.

Wählen Sie die gewünschten Einstellungen aus.

- Übersetzen vor dem Einchecken
  Diese Option ist per Voreinstellung aktiviert und bedeutet, dass vor dem Einchecken ein Übersetzungsvorgang durchgeführt wird.
- Übersetzungseinstellungen:
  Diese Einstellung legt fest, in welchem Umfang die Software-Änderungen übersetzt werden. Sie haben dabei folgende Optionen.
  - Markierte Objekte (nur Änderungen)
    Es werden die Änderungen an den markierten Objekten für das ausgewählte Gerät zusammen mit den gerätelokalen Objekten übersetzt.
  - Gerätesoftware (nur Änderungen)
    Es werden die Änderungen an den markierten Objekten zusammen mit dem auf den zugeordneten Geräten vorhandenen Softwarestand übersetzt.
    Beispiel: Sie haben mehrere Objekte markiert und geändert, die zu mehr als einem Gerät (z. B. zu CPU_1 und zu CPU_2) gehören. Beim Übersetzen werden diese markierten Objekte zusammen mit den Softwareständen der beiden CPUs übersetzt.
- Übersetzungseinstellungen im Multiuser-Editor deaktivieren
  Wenn Sie diese Option aktivieren, können die Übersetzungseinstellungen im Multiuser-Editor nicht mehr verändert werden.
  Es gelten dann die von Ihnen hier vorgenommenen Einstellungen als Voreinstellung.

Siehe auch

Multiuser-Netzwerkprofile einstellen (Seite 122)

Einführung zum Multiuser-Server (Seite 120)

2.7 Verbesserung der Funktionalität von Multiuser Engineering

Einführung

Im TIA Portal haben Sie die Möglichkeit, im Rahmen von Multiuser Engineering mit mehreren Benutzern gemeinsam und gleichzeitig in Multiuser-Projekten arbeiten zu können.

Die Funktionalität von Multiuser Engineering wurde wie nachfolgend beschrieben weiter verbessert.

Funktion "Automarkieren"

- Ab dem TIA Portal V15 werden die mit Multiuser Engineering bearbeiteten Objekte automatisch markiert. Die Funktion „Automarkieren“ kennzeichnet die in den lokalen Sessions neu erstellten oder geänderten Objekte automatisch für das Einchecken, indem sie die entsprechenden Multiuser-Markierungen setzt. Dabei ist es unerheblich, ob die Änderungen anwenderseitig oder systemseitig vorgenommen werden.
- Auch gelöschte Objekte werden automatisch markiert, damit diese Objekte nach dem Einchecken auch im Serverprojekt gelöscht werden können.
- Neben dem Automarkieren haben Sie nach wie vor die Möglichkeit, Objekte beim Bearbeiten manuell zu markieren.
- Es bestehen beim Arbeiten mit Multiuser Engineering keine funktionalen Unterschiede zwischen automatisch markierten und manuell markierten Objekten.
- Die Funktion „Automarkieren“ erkennt alle vom jeweiligen Bearbeiter vorgenommenen Änderungen an markierbaren Objekten und unterstützt Sie durch das automatische Markieren dieser Objekte dabei, Inkonsistenzen beim gemeinsamen Arbeiten mit Multiuser Engineering zu vermeiden.
  Siehe hierzu: Objekte für die Bearbeitung automatisch oder manuell markieren (Seite 153)

Offline arbeiten mit Multiuser Engineering

- Ab dem TIA Portal V15 können Sie beim Arbeiten mit Multiuser Engineering zwischen "Online- und Offline-Modus" wechseln. Das heißt, dass Sie auch ohne aktive Serververbindung zu einem Multiuser-Server in einer lokalen Session weiterarbeiten können.
- Unabhängig davon, ob die Serververbindung während des Arbeitens in der lokalen Session unterbrochen wird oder ob Sie sich von vorneherein für das "Offline arbeiten" entscheiden, können Sie nun an Ihren Multiuser-Projekten auch ohne aktive Serververbindung - mit einigen Einschränkungen - weiterarbeiten.
  Siehe hierzu: Offline Arbeiten mit Multiuser Engineering (Seite 38)

Verbesserungen in den Multiuser-Editoren

- Die Multiuser-Editoren wurden durch Kommentar-Funktionen erweitert zur Dokumentation von Art und Umfang der durchgeführten Änderungen.
- Beim Einchecken steht ein Filter für das Anzeigen von Objekten mit "Konflikten" zur Verfügung.
  Siehe hierzu: Funktionalität des Multiuser-Editors (Seite 45)

Kompatibilität des Multiuser-Servers

- Die Multiuser-Server aus verschiedenen Versionen sind nebeneinander einsetzbar.
- Der aktuelle Multiuser-Server unterstützt auch ältere TIA Portal-Projekte, mit entsprechendem Funktionsumfang.
- Die jeweils für eine neue Version neu hinzugefügte Serverfunktionalität ist nur dann verfügbar, wenn auch die neueste TIA Portal-Version installiert ist und die neueste Projekt-Version eingesetzt wird.
- Arbeiten Sie noch mit einem älteren TIA Portal oder mit einem älteren Multiuser-Server, so ist nur die in der installierten Version enthaltene Funktionalität verfügbar.
- Siehe hierzu: Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

Verbesserungen beim Multiuser "Configuration Tool" und "Administration Tool"

- Ab dem Multiuser-Server V15 stehen die externen Multiuser-Tools für die Konfiguration und Administration des Multiuser-Servers jetzt allen Sprachen des TIA Portals zur Verfügung.
- Der Multiuser-Server enthält eine erweiterte Projekthistorie mit Wiederherstellungsfunktion. Dies ermöglich die Nachvollziehbarkeit des Projektverlaufs am Multiuser-Server. Die Meilensteine eines Projekts können kommentiert und gespeichert werden. Die Projekthistorie ist für eine erweiterte Auswertung exportierbar.
  Siehe hierzu: Grafische Tools verwenden (Seite 58)

Benutzerverwaltung im TIA Portal mit Multiuser Engineering

- Wenn Sie ein Single-User-Projekt, das mithilfe der Benutzerverwaltung geschützt ist, auf einen Multiuser-Server hochladen, ist das daraus entstehende Multiuser-Projekt ebenfalls geschützt. Die im Projekt enthaltenen Benutzer und Benutzergruppen bleiben mit ihren Passwörtern bestehen und können sich damit in den lokalen Sessions des Multiuser-Projekts anmelden, wenn sie über die entsprechenden Funktionsrechte verfügen.
- Wenn Sie eine zuvor ungeschützte lokale Session mithilfe der Benutzerverwaltung schützen, wird dieser Schutz nach dem Einchecken und Aktualisieren der lokalen Session wieder entfernt, da das Multiuser-Projekt auf dem Server dadurch nicht geschützt wird. Dies gilt auch, falls mehrere Bearbeiter ihre lokalen Sessions schützen.
- Wenn Sie das Multiuser-Projekt auf dem Server mithilfe der Benutzerverwaltung schützen, werden auch die von diesem Multiuser-Projekt erstellten lokalen Sessions geschützt, sobald diese aktualisiert werden.
- Falls während der Bearbeitung Ihrer ungeschützten lokalen Session das Multiuser-Projekt auf dem Server mithilfe der Benutzerverwaltung geschützt wird und Ihrem Benutzerkonto nicht das Funktionsrecht "Projekt lesend/schreibend öffnen" zugewiesen wurde, können Sie keine Änderungen an Ihrer lokalen Session einchecken und Ihre lokale Session auch nicht aktualisieren.
  Weitere Informationen zur Benutzerverwaltung im TIA Portal finden Sie im Kapitel "Benutzerverwaltung verwenden".

Siehe auch

Objekte für die Bearbeitung automatisch oder manuell markieren (Seite 153)
Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)
Funktionalität des Multiuser-Editors (Seite 45)
Grafische Tools verwenden (Seite 58)

2.8 Anforderungen an Multiuser-Projekte

Einführung

Um gleichzeitig und parallel mit Multiuser Engineering arbeiten zu können, ist es sehr wichtig, dass das als Arbeitsgrundlage verwendete Multiuser-Serverprojekt gut strukturiert ist.

Anforderungen an Multiuser-Serverprojekte

Für das Erstellen von multiuserfähigen Serverprojekten starten Sie mit einem Single-User-Projekt, das die folgenden Anforderungen erfüllt:

- Das Projekt sollte bereits die Hardware-Konfiguration mit allen benötigten Verbindungen enthalten.
- Unterteilen Sie das Anwenderprogramm in voneinander unabhängige Programmteile.
- Erstellen Sie eine sinnvolle, technologisch orientierte Projektstruktur mit Ordnern und Gruppen für die von den einzelnen Benutzern zu bearbeitenden Objekte.
- Achten Sie bei der Strukturierung des Projektes darauf, dass mehrere Bearbeiter gleichzeitig und exklusiv mit den unterschiedlichen Projektverzeichnissen arbeiten können. Beispiel: Ein Benutzer bearbeitet die Bausteine in Gruppe 1, ein anderer Benutzer bearbeitet gleichzeitig die Bausteine in Gruppe 2.
- Nutzen Sie einen Main-OB und pro Programmteil einen zentralen FB oder FC, der die Funktionen des Teilprogramms aufruft.
- Legen Sie nach Möglichkeit für jede Gruppe eine eigene PLC-Variablentabelle an.
- Definieren Sie schon vorab im Single-User-Projekt alle benötigten Projektsprachen, die von den beteiligten Engineering-Systemen verwendet werden sollen.
- Legen Sie möglichst schon alle Verbindungen an, die im Multiuser-Projekt bearbeitet werden sollen.
- Definieren Sie die Bildschirmauflösung und legen Sie die Größe der HMI-Bilder fest.
- Nutzen Sie globale Datenbausteine für das Speichern der Daten für die einzelnen Programmteile, keine Merker.
- Laden Sie ein nach diesen Regeln erstelltes Single-User-Projekt auf einen Multiuser-Server, damit wird es zu einem Multiuser-Projekt.
- Bearbeiten Sie definierte Objekte des Multiuser-Projekts parallel mit mehreren Benutzern in den lokalen Sessions. Achten Sie beim Anlegen von neuen Objekten innerhalb der lokalen Sessions darauf, dass unterschiedliche symbolische Namen verwendet werden.

Beispiel für ein multiuserfähiges Serverprojekt

Die nachfolgende Grafik zeigt ein Beispiel für ein multiuserfähiges Serverprojekt mit guter Strukturierung, das ein paralleles Bearbeiten einzelner Objekte im Kontext von Multiuser Engineering erlaubt.

In jedem Programmteil befindet sich ein "Main-FB", der die jeweils untergeordneten Funktionen für diesen Programmteil aufruft.

Die von den einzelnen Engineering Systemen zu bearbeitenden Programmteile sind unterteilt in:

- Programmteil 1: "Conveyer"
- Programmteil 2: "Drill"
- Programmteil 3: "FurtherProgrammPart"

Im Programmteil "Conveyer" ruft der "ConveyerMainFB" die "ConveyerStartupFC"-Funktion auf.

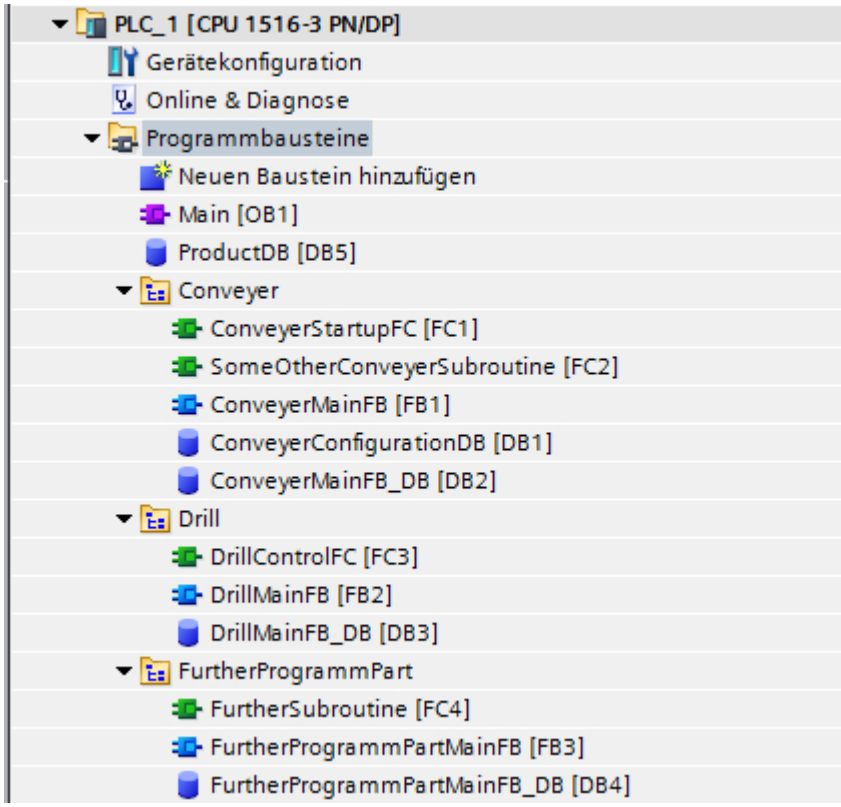

2.9 Upgrade von Multiuser-Projekten

Einführung

Sobald Sie versuchen, eine lokale Session, die mit einem älteren Stand des TIA Portals erstellt wurde, mit einem neueren Stand des TIA Portals zu öffnen, erhalten Sie eine entsprechende Nachricht. Um ältere Multiuser-Projekte und lokale Sessions mit der aktuellen Version des TIA-Portals nutzen zu können, müssen diese Projekte zuerst aktualisiert werden.

Upgrade von Multiuser-Projekten

Um ältere Multiuser-Projekte in der aktuellen Version des TIA Portals verwenden zu können, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session mit dem älteren Stand des TIA Portals, mit dem diese lokale Session erstellt wurde.
2. Stellen Sie sicher, dass alle Änderungen aus allen lokalen Sessions im zugehörigen Multiuser-Serverprojekt gespeichert sind.
3. Exportieren Sie das Multiuser-Serverprojekt mit der Erstellversion als Single-User-Projekt.
4. Öffnen Sie dieses Projekt mit der neuen TIA Portal-Version und führen Sie das vorgeschlagene Projekt-Upgrade aus.
5. Fügen Sie das aktualisierte Projekt als neues Multiuser-Serverprojekt zum aktuellen Multiuser-Server hinzu.
6. Erstellen Sie neue lokale Sessions für das aktuelle Multiuser-Serverprojekt.

Ergebnis

Das aktualisierte Multiuser-Serverprojekt kann wie gewohnt mit Multiuser Engineering bearbeitet werden.

2.10 Offline Arbeiten mit Multiuser Engineering

Einführung

Ab dem TIA Portal V15 können Sie auch "offline" mit Multiuser Engineering arbeiten.

Das heißt, dass Sie im Offline-Modus auch ohne eine aktive Serververbindung zu einem Multiuser-Server in einer lokalen Session mit den unten genannten Einschränkungen weiterarbeiten können.

Anwendungsfälle für das "Offline arbeiten"

Folgende Anwendungsfälle für das Arbeiten ohne Serververbindung sind möglich:

- **Eine Serververbindung kann beim Öffnen der lokalen Session nicht aufgebaut werden.**
  - Ein Dialog mit einer entsprechenden Meldung wird angezeigt.
  - Wenn Sie auf "OK" klicken, wird die lokale Session ohne Serververbindung geöffnet und Sie können offline weiterarbeiten.
- **Eine bestehende Serververbindung wird unterbrochen, während Sie in einer lokalen Session arbeiten.**
  - Ein Dialog mit einer entsprechenden Meldung wird angezeigt.
  - Wenn Sie auf "Wiederherstellen" klicken, können Sie offline weiterarbeiten.
- **Eine bestehende Serververbindung wird unterbrochen, während die Serverprojekt-Ansicht geöffnet ist.**
  - Ein Dialog mit einer entsprechenden Meldung wird angezeigt, der Ihnen verschiedene Optionen anbietet.
  - Wenn Sie auf "Wiederherstellen" klicken, wird versucht, die Serververbindung wieder herzustellen.
  - Wenn Sie auf "Verwerfen" klicken, werden Ihre in der Serverprojekt-Ansicht vorgenommenen Änderungen verworfen und die lokale Session geöffnet.
  - Da keine Serververbindung beim Öffnen der lokalen Session vorhanden ist, wird ein Dialog mit einer entsprechenden Meldung angezeigt.
  - Wenn Sie auf "Wiederherstellen" klicken, können Sie offline weiterarbeiten.
- **Sie haben von sich von vorneherein für das "Offline arbeiten" entschieden und diese Option im TIA Portal unter "Projekt > Multiuser > Offline arbeiten" angewählt.**

Einschränkungen beim "Offline arbeiten"

Folgende Einschränkungen sind beim Arbeiten im Offline-Modus vorhanden:

- Es werden Ihnen nur die eigenen Markierungen in Ihrer lokalen Session angezeigt.
- Markierungen von anderen Anwendern, also gelbe und rote Fähnchen, sowie Markierungen für veraltete Objekte können ohne Serververbindung nicht angezeigt werden.
- Das Symbol für den Serverstatus in der Multiuser-Funktionsleiste zeigt an, dass Sie offline arbeiten und keine Serververbindung vorhanden ist.

- Sie können Ihre Änderungen nicht ins Serverprojekt einchecken.
- Sie können Ihre lokale Session nicht aktualisieren.

Umschalten zwischen "Online- und Offline-Arbeiten"

Mit dem Befehl "Projekt > Multiuser > Offline arbeiten" können Sie zwischen dem Online- und dem Offline-Modus beim Arbeiten mit Multiuser Engineering umschalten.

Als Voreinstellung ist das "Online arbeiten" festgelegt. Wenn Sie auf das Kontrollkästchen vor dem Befehl "Offline arbeiten" klicken, wird in den Offline-Modus umgeschaltet. In der lokalen Session ändert sich das Symbol für die aktive Serververbindung von einem grünen Kreis in einen hellgrauen Kreis. Wenn Sie "Offline arbeiten" wieder abwählen, wird die Serververbindung wieder hergestellt. Bei erfolgreicher Serververbindung wird beim Serverstatus wieder der grüne Kreis angezeigt.

Die jeweilige Einstellung wird zusammen mit der lokalen Session gespeichert.

Wenn Sie in Ihrer lokalen Session im "Offline-Modus" gearbeitet haben und in den "Online-Modus" wechseln, wird in der geöffneten lokalen Session zuerst überprüft, ob eventuelle Markierungskonflikte vorliegen. Das Ergebnis der Überprüfung wird durch eine entsprechende Meldung angezeigt.

Werden Konflikte gefunden, so werden Sie aufgefordert, den Änderungseditor zu öffnen und sich dort mit der Filtereinstellung "Alle Konflikte anzeigen" die vorhandenen Konflikte anzeigen zu lassen.

Beheben Sie die bestehenden Konflikte und arbeiten Sie dann wie gewohnt online in Ihrer lokalen Session weiter.

2.11 Bedienelemente bei Multiuser Engineering

2.11.1 Multiuser-Symbole in der Bedienoberfläche

Einführung

Wenn Sie mit Multiuser Engineering arbeiten und eine lokale Session geöffnet haben, sehen Sie multiuser-spezifische Symbole in der Bedienoberfläche des TIA Portals.

Multiuser-Symbole in der Funktionsleiste

Die Bedienoberfläche von Multiuser Engineering zeigt bei einer geöffneten lokalen Session in der Funktionsleiste folgende Symbole:

Symbol	Bedeutung
	Einchecken Wenn Sie auf dieses Symbol klicken, wird der Multiuser-Editor in der Änderungsansicht geöffnet. In der Änderungsansicht werden alle für das Einchecken markierten Objekte angezeigt und Sie können die gewünschten Objekte auswählen und einchecken.
	Aktualisieren Wenn Sie auf dieses Symbol klicken, wird der Multiuser-Editor in der Ansicht "Aktualisieren" geöffnet. In dieser Ansicht können Sie Ihre lokale Session mit den neuen Inhalten des Serverprojekts aktualisieren, falls gewünscht.
	Serverprojekt öffnen/schließen Wenn Sie auf dieses Symbol klicken, wird das zu Ihrer lokalen Session gehörende Multiuser-Serverprojekt zum Bearbeiten geöffnet. Sie können Ihre Änderungen dann direkt im Serverprojekt durchführen.
	Commissioning-Modus ist nicht aktiv Dieses Symbol zeigt an, dass der Commissioning-Modus **nicht** aktiviert ist und Sie im Engineering-Modus arbeiten. Das Ein- und Ausschalten des "Commissioning-Modus" wird im Administration Tool durchgeführt. Das Einstellen des Modus gilt immer für alle lokalen Sessions eines Serverprojektes.
	Commissioning-Modus ist aktiv Dieses Symbol zeigt an, dass Sie im Commissioning-Modus arbeiten. Das Ein- und Ausschalten des "Commissioning-Modus" wird im Administration Tool durchgeführt. Das Einstellen des Modus gilt immer für alle lokalen Sessions eines Serverprojektes.

Symbol	Bedeutung
	Serverstatus "verbunden" Dieses Symbol zeigt an, dass der zugehörige Multiuser-Server verfügbar ist.
	Serverstatus "gesperrt" Dieses Symbol zeigt an, dass der zugehörige Multiuser-Server gerade von einem Benutzer verwendet wird und deshalb momentan gesperrt ist. Der Server wird auch als gesperrt angezeigt, wenn ein Benutzer in der Serverprojekt-Ansicht arbeitet. Wenn Sie den Mauszeiger auf diesem Symbol platzieren, erhalten Sie zusätzliche Informationen zur Sperre.
	Serverstatus "nicht verbunden" Dieses Symbol zeigt an, dass der zugehörige Multiuser-Server nicht verbunden ist.
	Serverstatus "Offline arbeiten" Dieses Symbol zeigt an, dass keine Verbindung zum Multiuser-Server besteht, da Sie die Funktion "Offline arbeiten" aktiviert haben. Deaktivieren Sie das Kontrollkästchen unter "Projekt > Multiuser > Offline arbeiten", um wieder eine Verbindung mit dem Multiuser-Server herzustellen.
	Dieses Overlay-Symbol zeigt an, dass auf dem Multiuser-Server ein neuerer Stand des Serverprojekts vorhanden ist.

Symbole für die Anzeige von Multiuser-Projekten

Folgende Symbole werden für Multiuser-Projekte und lokale Sessions in der Bedienoberfläche angezeigt:

Symbol	Bedeutung
	Multiuser-Server Dieses Symbol kennzeichnet einen Multiuser-Server.
	Multiuser-Projekt Dieses Symbol kennzeichnet ein Multiuser-Projekt, das zu einem Multiuser-Server gehört.
	Lokale Session Dieses Symbol kennzeichnet eine lokale Session, die zu einem Multiuser-Projekt gehört, das einem Multiuser-Server zugeordnet ist.

2.11.2 Multiuser-Markierungen in der Bedienoberfläche

Einführung

Beim Arbeiten mit Multiuser Engineering werden in der Bedienoberfläche des TIA Portals multiuserspezifische Symbole angezeigt, die Ihnen das parallele Arbeiten mit mehreren Benutzern erleichtern sollen. Durch eine eindeutige, nutzerspezifische Visualisierung soll die Zusammenarbeit innerhalb von Multiuser-Projekten und in den lokalen Sessions effektiver und transparenter gestaltet werden.

Markieren von Objekten

Um ein Objekt im Rahmen von Multiuser Engineering bearbeiten zu können, muss es zuerst "markiert" werden. Nur Objekte, die für das Einchecken markiert wurden, können nach der Bearbeitung in das Serverprojekt transferiert werden.

Markierungen können nur für "markierbare Objekte" hinzugefügt werden. Objekte, die in der lokalen Session nicht bearbeitet werden können, können auch nicht markiert werden.

Solche Objekte können jedoch in der Serverprojekt-Ansicht bearbeitet werden.

Die nachfolgend beschriebenen Symbole zeigen Ihnen an, welche Objekte von Ihnen selbst oder auch von anderen Bearbeitern markiert wurden.

Das Markieren von Objekten kann manuell erfolgen, indem Sie auf das graue Fähnchen vor dem jeweiligen Objekt klicken, oder automatisch Wird ein noch nicht manuell markiertes Objekt innerhalb einer lokalen Session in einem Editor geöffnet, so wird es über die Funktion "Automarkieren" automatisch markiert..

Symbole für Markierungen bei Multiuser Engineering

Folgende Symbole für Markierungen werden bei Multiuser Engineering verwendet:

Symbol	Bedeutung
	Markierbares Objekt Das Objekt kann durch Anklicken für das Einchecken markiert werden.
	Markiertes Objekt Das Objekt wurde in der eigenen lokalen Session für das Einchecken markiert.
	Markiertes Objekt Das Objekt wurde in einer anderen lokalen Session oder in der Serverprojekt-Ansicht für das Einchecken markiert.
	Markiertes Objekt mit Konflikt Das Objekt wurde in mehreren lokalen Sessions oder in einer lokalen Session und in der Serverprojekt-Ansicht zum Einchecken markiert und verursacht einen Konflikt. Ein Objekt sollte innerhalb des gesamten Multiuser-Arbeitsbereichs (innerhalb der lokalen Session und innerhalb der Serverprojekt-Ansicht) nur einmal für das Einchecken markiert sein! Ein rot markiertes Objekt kann trotzdem eingecheckt werden. Allerdings kann das Einchecken evtl. zu ungewollten Überschreibungen im Serverprojekt führen, da immer nur der zuletzt eingecheckte Stand im Serverprojekt übernommen wird.
	Objekt ist veraltet Wird eines der o.g. Symbole zusätzlich mit diesem Overlay gekennzeichnet, so entspricht das Objekt nicht mehr dem neuesten Stand des Serverprojektes und sollte unbedingt aktualisiert werden. Falls das Objekt ohne Aktualisierung eingecheckt wird, wird die Bearbeitung des anderen Benutzers überschrieben und damit wieder rückgängig gemacht. Das Aktualisieren stellt sicher, dass alle Bearbeiter eines Multiuser-Projektes einen konsistenten Stand haben.

Hinweis

Objekte ohne Fähnchen

Beachten Sie bitte, dass Objekte ohne Fähnchen nicht in der lokalen Session sondern ausschließlich in der Serverprojekt-Ansicht bearbeitet werden können.

Anzeige von Multiuser-Markierungen im TIA Portal

Wenn Sie mit Multiuser Engineering innerhalb der lokalen Session oder innerhalb des Serverprojektes arbeiten, werden die multiuser-spezifischen Markierungen in folgenden Ansichten des TIA Portals in allen Editoren dargestellt:

- in der Projektnavigation
- in der Detailansicht
- im Übersichtsfenster unter "Details"
- in den Bibliotheksansichten

In allen Tabellendarstellungen innerhalb der Projektnavigation gibt es beim Arbeiten mit Multiuser Engineering eine zusätzliche erste Tabellenspalte. Diese erste Tabellenspalte ist für die oben beschriebenen multiuser-spezifischen Markierungen vorgesehen.

Die nachfolgende Grafik zeigt beispielhaft mehrere markierte Objekte in der Projektnavigation:

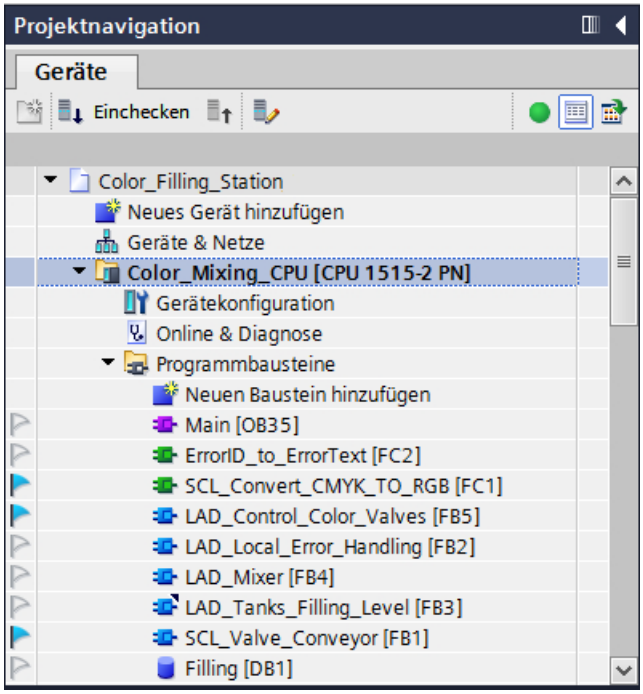

2.11.3 Multiuser-Banner in der Bedienoberfläche

Einführung

Beim Arbeiten mit Multiuser Engineering werden Sie durch die Anzeige von multiuserspezifischen Bannertexten innerhalb von Editoren unterstützt.

Bannertexte enthalten Hinweise und Warnungen für die weitere Vorgehensweise, die Ihnen das Arbeiten erleichtern sollen.

Bannertexte können auf- und zugeklappt werden, indem Sie auf den gelben Kreis mit dem Ausrufungszeichen klicken.

Beispiel für einen Multiuser-Bannertext

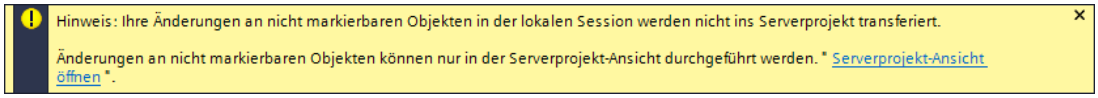

Funktionalität der Multiuser-Bannertexte

Bannertexte werden sowohl beim Arbeiten in der lokalen Session als auch beim Arbeiten in der Serverprojekt-Ansicht angezeigt.

Sobald Sie zusätzliche Informationen zum Arbeiten mit Multiuser Engineering benötigen, erscheint ein gelb hinterlegter Bannertext.

Die Multiuser-Bannertexte haben folgende Funktion:

- Informationshinweise für die Bearbeitung des geöffneten Objektes:
  - Bannertexte weisen z. B. darauf hin, dass das geöffnete Objekt noch nicht für das Einchecken markiert wurde und Änderungen deshalb nicht ins Serverprojekt übernommen werden können.
  - Sie weisen darauf hin, dass dieses spezifische Objekt nicht in der lokalen Session, sondern nur in der Serverprojekt-Ansicht bearbeitet werden kann.
- Links erleichtern die Ausführung der weiterhin benötigten Arbeitsschritte.
  - Über die Links können Sie z. B. auf komfortable Art und Weise die Markierung für das geöffnete Objekt durchführen oder die Serverprojekt-Ansicht öffnen, um die Bearbeitung dort fortzusetzen.

Sobald Sie die benötigten Arbeitsschritte innerhalb der einzelnen Editoren durchgeführt haben, werden die Bannertexte im Editor teilweise wieder entfernt und Sie können wie gewohnt weiter arbeiten.

2.11.4 Funktionalität des Multiuser-Editors

Einführung

Sie können den Multiuser-Editor über die Schaltflächen "Einchecken" und "Lokale Session aktualisieren" in der Funktionsleiste von Multiuser Engineering aufrufen.

Der Multiuser-Editor verfügt über zwei verschiedene Ansichten, abhängig davon, über welche Schaltfläche in der Multiuser-Funktionsleiste der Editor gestartet wurde:

- Ansicht "Einchecken": wird durch Klicken auf die Schaltfläche "Einchecken" in der Funktionsleiste der Projektnavigation geöffnet.
- Ansicht "Aktualisieren": wird durch Klicken auf die Schaltfläche "Lokale Session aktualisieren" in der Funktionsleiste der Projektnavigation geöffnet.
  Die Ansicht "Aktualisieren" kann immer geöffnet werden, auch wenn in der lokalen Session keine Objekte enthalten sind, die aktualisiert werden können.

Der Aufbau des Multiuser-Editors ist in beiden Ansichten identisch, es werden jedoch unterschiedliche Schaltflächen und Optionen angeboten.

Hinweis

Anzeigen im Multiuser-Editor

Wenn Sie den Multiuser-Editor geöffnet haben und gleichzeitig Änderungen in Ihrer lokalen Session vornehmen, werden diese Änderungen im geöffneten Editor erst nach einem "Aktualisieren" sichtbar. Klicken Sie dazu auf die Schaltfläche "Aktualisieren" (kreisförmige Pfeile) im geöffneten Editor.

Multiuser-Editor in der Ansicht "Einchecken"

Die nachfolgende Grafik zeigt ein Beispiel für den Multiuser-Editor in der Ansicht "Einchecken":

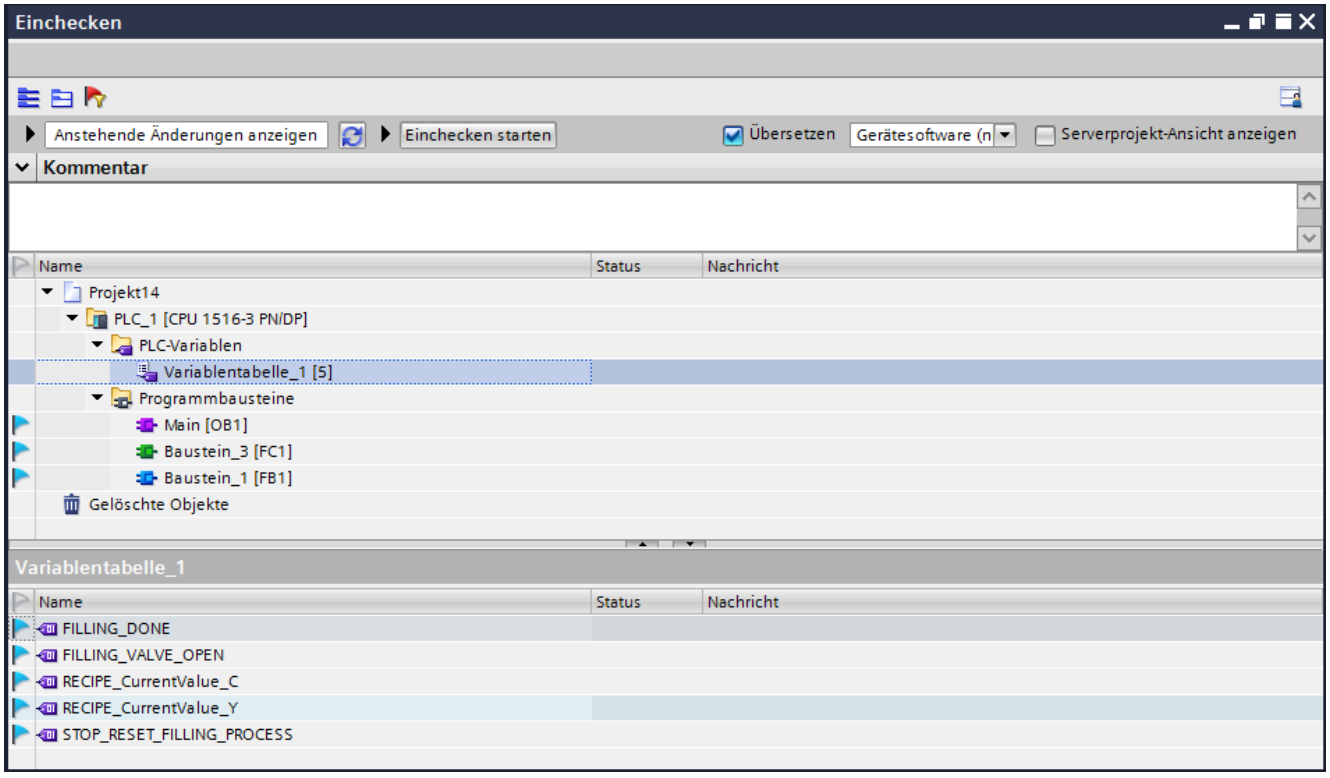

Multiuser-Editor in der Ansicht "Aktualisieren"

Die nachfolgende Grafik zeigt ein Beispiel für den Multiuser-Editor in der Ansicht "Aktualisieren":

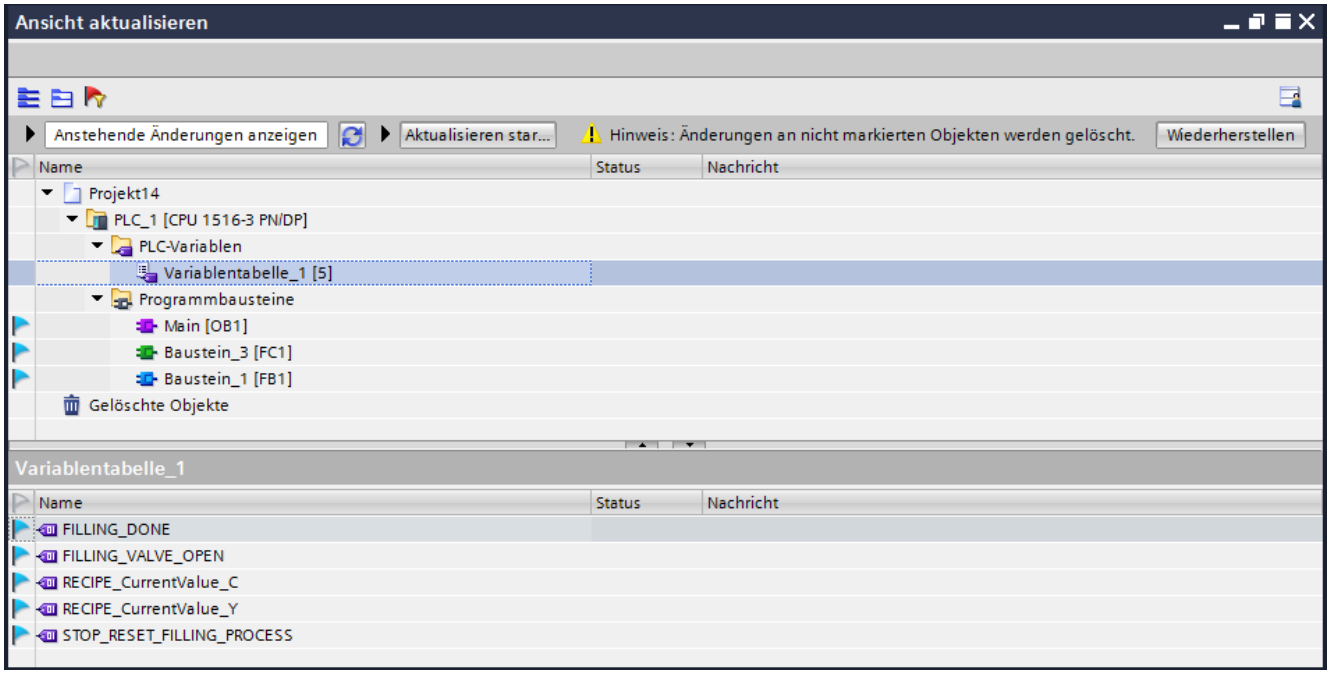

Aufbau des Multiuser-Editors

Der Multiuser-Editor hat folgenden Aufbau:

1. Funktionsleiste:

Zeigt folgende Schaltflächen an:

- Auf der rechten Seite die Schaltfläche zum Speichern der Layout-Einstellungen.
- Auf der linken Seite die Schaltflächen "Alles zuklappen", "Alles aufklappen" und "Konflikte anzeigen".
  Wenn Sie auf "Konflikte anzeigen" klicken, werden Ihnen im Editor alle Markierungen mit "Konflikt" (rotes Fähnchen) angezeigt.

2. Schaltflächen-Bereich:

Zeigt die multiuser-spezifischen Schaltflächen für das Einchecken und das Aktualisieren an.

3. Kommentarzeile:

Die Kommentarzeile ist nur in der Ansicht "Einchecken" vorhanden. Hier können Sie einen Kommentar eintragen, um die in der lokalen Session vorgenommenen Änderungen zu dokumentieren.

4. Arbeitsbereich:

Zeigt die Änderungsansicht vor dem Einchecken bzw. Aktualisieren an.

- Zeigt alle in der lokalen Session markierten Objekte für das Einchecken und Aktualisieren an, wie diese auch in der Projektnavigation dargestellt werden.
- Zeigt die Statusmeldungen mit den Resultaten für das Einchecken und für das Aktualisieren an.
- Der Ordner "PLC_Variablen" kann geöffnet werden.
  Durch Anklicken der Variablentabelle werden im unteren Fenster in der Detailansicht die für das Einchecken markierten Variablen angezeigt..
- Der Ordner "Bibliothekselemente" wird nur angezeigt, falls Bibliothekselemente für das Einchecken markiert wurden.
  Wird der Ordner selektiert, werden in der Detail-Anzeige die markierten Bibliothekselemente angezeigt.
- Der Ordner "Gelöschte Objekte" wird nur angezeigt, falls in der lokalen Session Objekte gelöscht wurden.
  Wird der Ordner selektiert, werden in der Detail-Anzeige alle Objekte angezeigt, die in der lokalen Session gelöscht wurden.

5. Detail-Anzeige:

Zeigt Informationen zu untergeordneten Elementen (hier zu den markierten PLC-Variablen) und Statusmeldungen an.

Hinweis

Vorgehensweisen zum Arbeiten mit dem Multiuser-Editor

Die Vorgehensweisen für das Arbeiten mit dem Multiuser-Editor werden in den nachfolgenden Kapiteln ausführlich beschrieben.

2.12 Multiuser-Server installieren und deinstallieren

2.12.1 Hinweise zur Installation des Multiuser-Servers

Einführung

Die Installation des Multiuser-Servers ermöglicht Ihnen, die Funktionalität des Multiuser Engineering zu nutzen.

Sie können den Multiuser-Server wie folgt installieren:

- Zusammen mit der Installation des TIA Portals.
- Als Standalone-Installation über einen eigenständigen Installationsprozess ohne TIA Portal.

Hinweis

Für die Serverinstallation werden Administratorrechte benötigt.

Installation des Multiuser-Servers mit dem TIA Portal

Die Installation des Multiuser-Servers erfolgt zusammen mit der Installation folgender TIA Portal-Produkte:

- SIMATIC STEP 7 Basic
- SIMATIC STEP 7 Professional
- SIMATIC WinCC Basic
- SIMATIC WinCC Comfort/Advanced
- SIMATIC WinCC Professional

Der Multiuser-Server ist in der Produktauswahl teilweise per Voreinstellung aktiviert und wird dann mitinstalliert. Überprüfen Sie die Einstellung während der Installation des TIA Portals. Sie finden die jeweilige Voreinstellung für den Multiuser Server während des Setups bei der Produktauswahl im Verzeichnis "Options". Wählen Sie hier durch das Setzen oder entfernen des Häkchens aus, ob der Multiuser-Server während der Installation des TIA Portals mitinstalliert werden soll oder nicht.

Installation des Multiuser-Servers als Standalone-Installation

Die Installation des Multiuser-Servers kann auch als Standalone-Installation durchgeführt werden.

Eine Beschreibung zur automatisierten Installation finden Sie auch auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Gleiche TIA Portal-Versionen bei der Installation

Achten Sie bei der Installation von TIA Portal-Produkten auf den verschiedenen Clients darauf, dass alle Teilnehmer für die Installation die gleichen Stände der Software-Produkte verwenden.

Wenn Sie also z. B. STEP 7 Professional, WinCCAdvanced und Safety sowie den Multiuser Server mit der Version Vx.y installiert haben, müssen alle anderen Teilnehmer von Team Engineering ebenfalls diese Software in identischen Versionen installiert haben.

Die Installation von Service Packs und Updates muss für alle Produkte zu einem gemeinsamen Zeitpunkt erfolgen.

Installationspfad

Verwenden Sie keine Unicode-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

Installation eines neueren Multiuser-Servers

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Virenscanner

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscanner verhindern dies. Daher wird empfohlen, während der Installation des Multiuser-Servers Virenscanner zu deaktivieren und erst danach wieder zu aktivieren.

Unterstützte Virtualisierungsplattformen

Sie können den Multiuser-Server ebenso wie das TIA Portal auch in einer virtuellen Maschine installieren.

Verwenden Sie dazu eine der folgenden Virtualisierungsplattformen in der angegebenen oder einer neueren Version:

- VMware vSphere Hypervisor (ESXi) 6.5
- VMware Workstation 12.5.7
- VMware Player 12.5.7
- Microsoft Hyper-V Server 2016

Für die Installation einer Visualisierungsplattform gelten die gleichen Voraussetzungen wie für die Installation des TIA Portals.

2.12.2 Hinweise zur Kompatibilität des Multiuser-Servers

Einführung

Um mit Multiuser Engineering arbeiten zu können, benötigen Sie einen Multiuser-Server, der Ihre Multiuser-Projekte und die lokalen Sessions verwaltet.

Im TIA Portal können verschiedene Versionen des Multiuser-Servers nebeneinander auf einem Gerät installiert werden und gleichzeitig laufen.

Kompatibilität des Multiuser-Servers mit dem TIA Portal

Die Kommunikation zwischen dem Multiuser-Server und dem TIA Portal wird für folgende Anwendungsfälle unterstützt:

Multiuser-Server-Version	TIA Portal-Projektversion
V15.1	V15.1, V15 und V14
V15	V15 und V14
V14	V14

Hinweis

Hinweise zur Kompatibilität des Multiuser-Servers

Die für V15.1 neu hinzugefügte Serverfunktionalität ist nur dann verfügbar, wenn auch das TIA Portal V15.1 installiert ist.

Arbeiten Sie noch mit einem TIA Portal <V15.1 oder mit einem Multiuser-Server <V15.1, so ist nur die in der jeweiligen Produkt-Version enthaltene Funktionalität verfügbar.

Ablageverzeichnis des Multiuser-Servers

Jede TIA Portal-Version wird mit einer entsprechenden Multiuser-Serverversion ausgeliefert. Sie haben die Möglichkeit, ein neues Ablageverzeichnis für den neuen Multiuser-Server zu definieren, oder ein bestehendes Ablageverzeichnis eines älteren Multiuser-Servers, in dem Sie z.B. die V14-Projekte gespeichert haben, auch für die neue Serverversion V15 weiter zu verwenden.

Dafür gelten folgende Regeln:

- Um unterschiedliche Multiuser-Server parallel und unabhängig voneinander zu verwenden, müssen Sie für die unterschiedlichen Multiuser-Server jeweils unterschiedliche Ports und Ablageverzeichnisse bei der Serverinstallation für den neuen Server konfigurieren.
- Um ein älteres Serververzeichnis auch als Ablage für den neu installierten Multiuser-Server weiterhin zu verwenden, tragen Sie das bestehende Verzeichnis bei der Konfiguration des neuen Multiuser-Servers ein. Der Ablagepfad wird bei der Installation des Multiuser-Servers überprüft und Sie erhalten eine entsprechende Meldung, dass dieses Ablageverzeichnis bereits für einen anderen Multiuser-Server verwendet wird.
  Danach haben Sie folgende Optionen:
  - Sie können die Installation abbrechen und ein neues Ablageverzeichnis für den Multiuser-Server eintragen.
  - Oder Sie führen ein Upgrade des Ablageverzeichnisses für den Multiuser-Server durch.

  Das hochgerüstete Verzeichnis wird durch das Upgrade jedoch für ältere Serverversionen ungültig und kann mit dem älteren Multiuser-Server nicht mehr gestartet werden.
  Das Upgrade des Ablageverzeichnisses hat keinen Einfluss auf die im Verzeichnis gespeicherten TIA Portal-Projekte.
  Bevor das Upgrade durchgeführt wird, wird eine interne Sicherung durchgeführt, so dass ein fehlerhaft durchgeführtes Upgrade rückgängig gemacht werden kann.
  Treten während des Upgrades keine Fehler auf, wird die Funktion "Rückgängig machen" nicht angeboten.

Hinweis

Ablageverzeichnis des Multiuser-Servers

Das Ablageverzeichnis des Multiuser-Servers ist während der Installation des Servers gesperrt. Dies soll verhindern, dass ein Ablageverzeichnis geändert wird, auf das gerade aktiv zugegriffen wird.

Siehe auch

Multiuser-Server mit den Administrative Tools konfigurieren (Seite 88)

Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

Grafische Tools verwenden (Seite 58)

Kommandozeilen-Tools verwenden (Seite 86)

2.12.3 Multiuser-Server installieren

Einleitung

Der Multiuser-Server kann im Rahmen der Installation des TIA Portals oder auch standalone installiert werden. Wird der Multiuser-Server mit dem TIA Portal mitinstalliert, so wird immer auch die dem TIA Portal entsprechende Version des Multiuser-Servers installiert.

Eine Beschreibung zur automatisierten Installation finden Sie auch auf der Produkt-DVD im Verzeichnis "Documents\Readme\<Sprachverzeichnis>".

Hinweis

Neuinstallation vor Deinstallation

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Regeln für die Installation

Auf einem Computer können mehrere Versionen des Multiuser-Servers installiert sein, die unterschiedliche Projektversionen des TIA Portals nutzen können.

Nutzbare Projektversionen:

Multiuser-Server-Version	TIA Portal-Projektversion
Server V15.1	V15.1, V15 und V14
Server V15	V15 und V14
Server V14	V14

Beachten Sie jedoch, dass jede Version des TIA Portals und des Multiuser-Servers jeweils nur über die in dieser Version enthaltene Funktionalität verfügt.

Multiuser Engineering führt kein Upgrade der Projektversionen durch.

Voraussetzungen für die Installation

Für die Installation des Multiuser-Servers gelten die gleichen Voraussetzungen wie für die Installation des TIA Portals.

Multiuser-Server mit dem TIA Portal installieren

Für die Installation gehen Sie folgendermaßen vor:

1. Beachten Sie die Voraussetzungen für die Installation des ausgewählten Softwarepakets.
2. Starten Sie das Setup-Programm des gewünschten Produkts durch Einlegen des Installationsdatenträgers in das entsprechende Laufwerk.
3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Vergewissern Sie sich, dass der Multiuser-Server in der Produktauswahl angewählt ist und folgen Sie den Hinweisen in den Setup-Dialogen.

Ergebnis

Der Multiuser-Server wurde auf Ihrem Computer zusammen mit dem TIA Portal installiert.

Multiuser-Server standalone installieren

Für die Installation gehen Sie folgendermaßen vor:

1. Beachten Sie die Voraussetzungen für die Installation des Multiuser-Servers.
2. Starten Sie den Setup für die Standalone-Installation des Multiuser-Servers durch Doppelklick auf die selbstextrahierende Exe-Datei "TIA_Portal_Multiuser_Server_V<x.y>.exe". Sie finden diese Datei im "Support"-Verzeichnis auf der jeweiligen Produkt-DVD.
3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Folgen Sie den Anweisungen in den Setup-Dialogen.

Ergebnis

Der Multiuser-Server wurde auf Ihrem Computer installiert.

2.12.4 Erneute Installation des Multiuser-Servers ohne Datenverlust durchführen

Einleitung

Nach dem Löschen eines bereits installierten Multiuser-Servers können Sie diesen ohne Datenverlust erneut installieren, solange die Verzeichnisstruktur mit den vorher verwendeten Daten noch vorhanden und für den neu installierten Multiuser-Server erreichbar ist.

Regeln für die Installation

Auf einem Computer können mehrere Versionen des Multiuser-Servers und des TIA Portals installiert sein, es ist jedoch immer nur eine Version des Multiuser-Servers mit der entsprechenden Version des TIA Portals ablauffähig.

Siehe hierzu: Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

Installation eines neueren Multiuser-Servers

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Voraussetzungen für die Installation

Für die Installation des Multiuser-Servers V<x.y> gelten die gleichen Voraussetzungen wie für die Installation des TIA Portals V<x.y>.

Multiuser-Server erneut standalone installieren

Für die Installation gehen Sie vor wie folgt:

1. Beachten Sie die Voraussetzungen für die Installation des ausgewählten Softwarepakets.
2. Starten Sie den Setup für die Standalone-Installation des Multiuser-Servers durch Doppelklick auf die selbstextrahierende Exe-Datei "TIA_Portal_Multiuser_Server_V<x.y>.exe", die im "Support"-Verzeichnis auf der jeweiligen Produkt-DVD liegt.
3. Wählen Sie die gewünschten Einstellungen aus und klicken Sie auf die Schaltfläche "Installieren".
4. Folgen Sie den Anweisungen in den Setup-Dialogen.

Ergebnis

Auf Ihrem Computer wurde der Multiuser-Server installiert.

Siehe auch

Hinweise zur Kompatibilität des Multiuser-Servers (Seite 51)

2.12.5 Multiuser-Server deinstallieren

Einleitung

Der Multiuser-Server kann über die Systemsteuerung deinstalliert werden.

Hinweis

Neuinstallation vor Deinstallation

Beachten Sie, dass Sie **zuerst den neueren Multiuser-Server installieren** und danach erst die ältere Version des Multiuser-Servers deinstallieren.

Mit dieser Vorgehensweise stellen Sie sicher, dass die für den Server vorhandenen Benutzerberechtigungen auch für den neuen Server übernommen werden können. Werden alle vorhandenen Multiuser-Server vor der Neuinstallation komplett gelöscht, gehen auch die Benutzereinstellungen verloren und müssen neu erstellt werden.

Voraussetzungen für die Deinstallation

Auf Ihrem Computer ist eine Multiuser-Serverinstallation vorhanden.

Multiuser-Server deinstallieren

Für die Deinstallation gehen Sie vor wie folgt:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie in der Systemsteuerung auf den Eintrag "Software".
   Der Dialog "Software" wird geöffnet.
3. Wählen Sie im Dialog "Software" den Eintrag für den "Multiuser-Server" aus und klicken Sie auf die Schaltfläche "Entfernen".
   Es erscheint eine Sicherheitsabfrage.
4. Bestätigen Sie die Sicherheitsabfrage, indem Sie auf die Schaltfläche "Deinstallieren" klicken.

Ergebnis

Der Multiuser-Server wurde auf ihrem Computer deinstalliert.

2.13 Multiuser-Server konfigurieren und verwalten

2.13.1 Möglichkeiten zum Konfigurieren und Verwalten des Multiuser-Servers

Einführung

Für die Konfiguration und Verwaltung des Multiuser-Servers stehen Ihnen alternativ folgende Möglichkeiten zur Verfügung:

- Verwenden der Grafischen Tools (Configuration Tool und Administration Tool).
- Verwenden der Kommandozeilen-Tools (Multiuser Administrative Tools und Multiuser Power Tools).

Mit beiden Vorgehensweisen können Sie den Multiuser-Server konfigurieren und verwalten.

Die grafischen Tools sind durch die grafische Benutzeroberfläche komfortabler zu bedienen. Die Kommandozeilen-Tools haben einen etwas erweiterten Funktionsumfang bezüglich des Error-Logs. Hier können Sie die Error-Logs als Zip-Datei verpacken und versenden.

Sie haben die Möglichkeit, die von Ihnen bevorzugte Vorgehensweise anzuwenden.

Für die Konfiguration des Multiuser-Servers werden unabhängig von der ausgewählten Vorgehensweise Administratorrechte benötigt.

Für Verwaltungsaufgaben werden teilweise Administratorrechte (bei der Benutzerverwaltung) und Nutzerrechte benötigt.

Hinweis

Verfügbare Sprachen

- Die Tools zur Konfiguration und Verwaltung des Multiuser-Servers stehen für den Multiuser-Server V14 nur in Englischer Sprache zur Verfügung.
- Ab dem Multiuser-Server V15 stehen die Tools in den vom TIA-Portal unterstützten Oberflächensprachen zur Verfügung.

Konfiguration und Verwaltung mit den grafischen Tools

Das grafische Tool bietet Ihnen auf sehr komfortable Art und Weise die Möglichkeit, den Multiuser-Server zu konfigurieren und zu verwalten.

Sie sehen auf einen Blick alle Einstellmöglichkeiten und können diese durch Auswahl der entsprechenden Optionen aktivieren.

Die genauen Vorgehensweisen finden Sie unter: Grafische Tools verwenden (Seite 58)

Konfiguration und Verwaltung mit den Kommandozeilen-Tools

Die Konfiguration des Multiuser-Servers wird mit den Administrative Tools durchgeführt.

Für die Verwaltung des Multiuser-Servers werden die Power Tools eingesetzt.

Die Power Tools stehen Ihnen auch innerhalb des TIA Portals zur Verfügung, um Verwaltungsaufgaben auf dem Multiuser-Server durchzuführen.

Die genaue Vorgehensweise finden Sie unter: Kommandozeilen-Tools verwenden (Seite 86)

Hinweis

Arbeiten mit den Kommandozeilen-Tools

Für das Arbeiten mit den Kommandozeilen-Tools sollten Sie bereits Erfahrung im Umgang mit Eingaben in die Kommandozeile haben.

Falls Sie damit keine Erfahrungen haben, sollten Sie die komfortablere grafische Benutzeroberfläche für die Konfiguration und Verwaltung des Multiuser-Servers nutzen.

Siehe auch

Kommandozeilen-Tools verwenden (Seite 86)

Grafische Tools verwenden (Seite 58)

2.13.2 Grafische Tools verwenden

2.13.2.1 Einführung zu den grafischen Multiuser-Tools

Einführung

Der Multiuser-Server kann mithilfe der grafischen Tools auf komfortable Weise konfiguriert und verwaltet werden.

Für die Konfiguration des Multiuser-Servers verwenden Sie das "Configuration Tool".

Für die Administration verwenden Sie das "Administration Tool".

Für die Konfiguration des Multiuser-Servers benötigen Sie teilweise Administratorrechte.

Hinweis

Verfügbare Sprachen

- Die Tools zur Konfiguration und Verwaltung des Multiuser-Servers stehen für den Multiuser-Server V14 nur in Englischer Sprache zur Verfügung.
- Ab dem Multiuser-Server V15 stehen die Tools in den vom TIA-Portal unterstützten Oberflächensprachen zur Verfügung.

Voraussetzung

Der Multiuser-Server ist installiert.

Schritte für die Konfiguration des Multiuser-Servers mit dem "Configuration Tool"

Um den Multiuser-Server zu konfigurieren, sind folgende Schritte notwendig:

- Rufen Sie das grafische Tool für die Multiuser-Server-Konfiguration mit Administrator-Rechten auf.
- Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
- Klicken Sie auf den Eintrag "TIA Portal Multiuser" > Multiuser Server V<x.y> Configuration", um das Configuration Tool zu öffnen.
- Wählen Sie in der schwarz hinterlegten Fußzeile die gewünschte Sprache aus.
- Tragen Sie die gewünschten Konfigurationsdaten ein und klicken Sie auf "Service installieren".

Die genauen Vorgehensweisen finden Sie unter: Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

Schritte für die Verwaltung des Multiuser-Servers mit dem "Administration Tool"

Um den Multiuser-Server zu verwalten, sind folgende Schritte notwendig:

- Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
- Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
- Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
- Im Register "Administration" fügen Sie eine Verbindung zum gewünschten Multiuser-Server hinzu und tragen Sie die Daten für die Verwaltung des Servers ein.
- Im Register "Einstellungen" wählen Sie die Spracheinstellungen aus und verwalten Bestätigungen und Zertifikate.

Die genauen Vorgehensweisen finden Sie unter: Multiuser-Server mit grafischem Administration Tool verwalten (Seite 65)

Siehe auch

Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

Multiuser-Server mit grafischem Administration Tool verwalten (Seite 65)

2.13.2.2 Multiuser-Server mit grafischem Configuration Tool konfigurieren

Einführung

Für die Konfiguration des Multiuser-Servers verwenden Sie das grafische "Configuration Tool".

Voraussetzung

Der Multiuser-Server ist installiert.

Funktionalität des Configuration Tools

Folgende Funktionen sind mit dem "Configuration Tool" ausführbar:

Bereich "Serverstatus"

- Enthält beim erstmaligen Öffnen des Configuration Tools die in der Grafik angezeigten Inhalte und ist nicht editierbar.
- Nachdem die Konfiguration erfolgreich durchgeführt wurde, kann der Server über die Schaltflächen in diesem Bereich deinstalliert, gestartet und gestoppt werden.

Bereich "Verbindungseinstellungen"

- Ermöglicht die Auswahl der Verbindungsdaten.

Bereich "Sicherheit"

- Bietet bei einer ausgewählten https-Verbindung die Möglichkeit, ein neues Zertifikat zu erzeugen oder ein bereits vorhandenes Zertifikat zu nutzen.

Bereich "Speicher"

- Ermöglicht die Auswahl des Speicherortes für Multiuser-Projekte.
- Ermöglich die Auswahl, wie viele Projektversionen als "Rollback" zur Verfügung stehen.

Hinweis

Speicherplatz auf dem Multiuser-Server

Für die Projektverwaltung benötigen Sie auf dem Multiuser-Server genügend freien Speicherplatz, abhängig von der Anzahl der gespeicherten Projekte, deren Projektgrößen und der Anzahl der vorgehaltenen Projektrevisionen.

Grundsätzlich sollte der bereitgestellte Speicherplatz mindestens das fünffache der zu erwartenden Projektgröße betragen.

Bei Nutzung einer erhöhten Anzahl von Projektrevisionen muss auch der zur Verfügung stehende freie Speicherplatz entsprechend erhöht werden. Die Anzahl der vorgehaltenen Projektrevisionen legen Sie bei der Konfiguration des Multiuser-Servers fest.

Generell sollten Sie ausreichend Speicherkapazität zur Verfügung stellen und auch die Meldungen des Betriebssystems bei abnehmendem Speicherplatz beachten bzw. auswerten.

Hinweis

Arbeiten mit Multiuser Commissioning

Beim Arbeiten mit Multiuser Commissioning wird nach jedem Laden in die CPU automatisch auch eine neue Revision des Serverprojekts erstellt.

Um den Speicherplatzbedarf zu reduzieren, sollten Sie die Anzahl der Revisionen auf ein Tagesmaximum reduzieren und nur die wichtigen Revisionen behalten, z. B. wenn ein Projekt-Meilenstein erreicht wurde.

Beachten Sie auch, dass nach dem Erreichen der maximalen eingestellten Anzahl der Revisionen die ältesten Revisionen automatisch gelöscht werden.

Die Anzahl der zu speichernden Projektrevisionen legen Sie bei der Konfiguration des Multiuser-Servers im Configuration Tool fest. Die maximale Anzahl beträgt 100 Revisionen.

Oberfläche des Configuration Tools

Die Oberfläche des Configuration Tools zeigt beim erstmaligen Öffnen folgenden Inhalt:

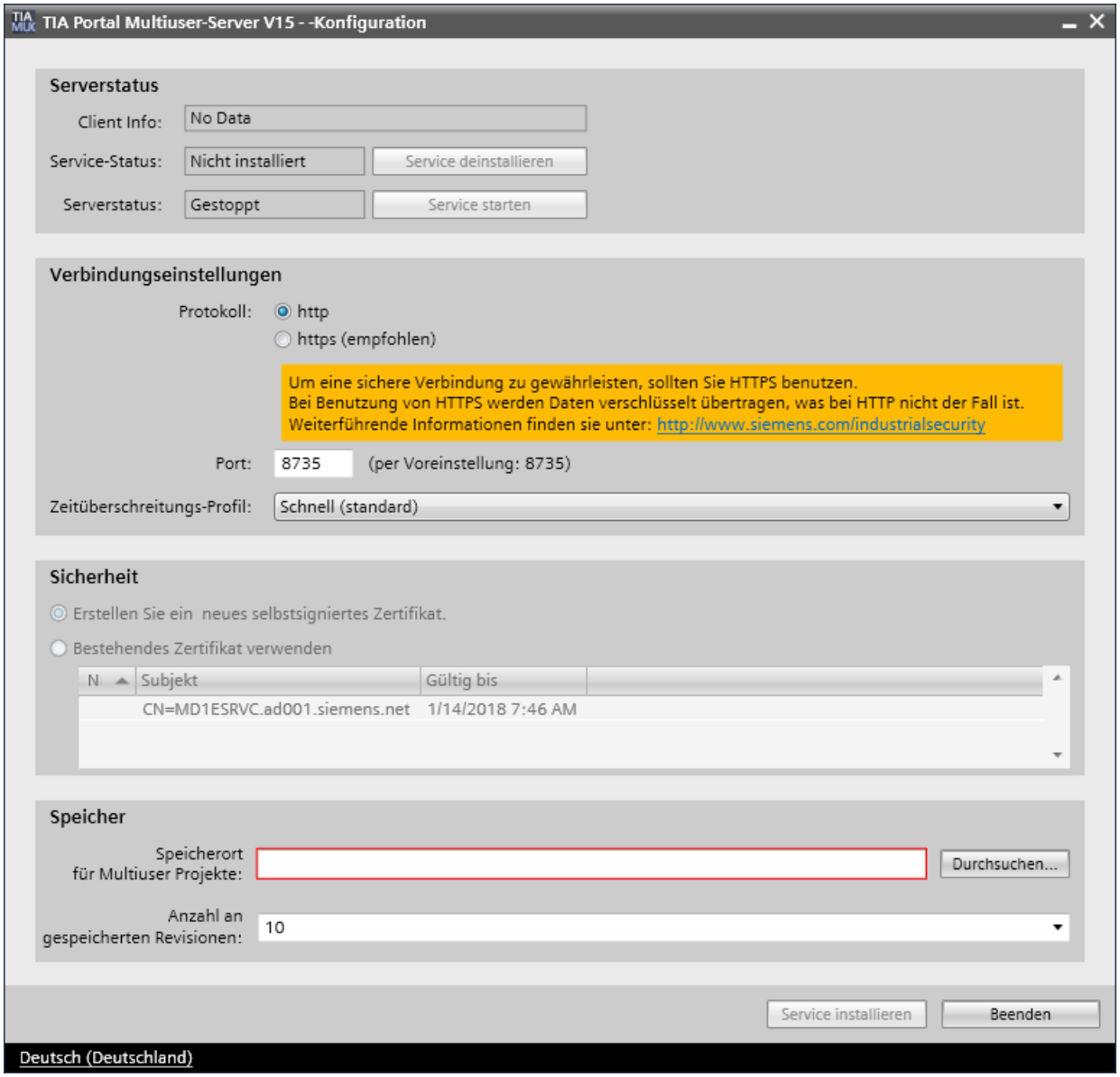

Konfiguration des Multiuser-Servers mit dem "Configuration Tool"

Um den Multiuser-Server zu konfigurieren, gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Konfiguration mit Administrator-Rechten auf.
   Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Configuration", um das Configuration Tool zu öffnen.
   Daraufhin öffnet sich die Benutzeroberfläche des Configuration Tools.
2. Wählen Sie im Bereich "Verbindungseinstellungen" das gewünschte Protokoll aus.
   Als Voreinstellung ist "http" aktiviert. Der gelb hinterlegte Text wird ausgeblendet, sobald Sie "https" anklicken.
   Mit "https" wird die Kommunikation verschlüsselt und ein Zertifikat angefordert. Die Kommunikation hat damit eine höhere Sicherheitsstufe als die unverschlüsselte "http"-Kommunikation.
3. Tragen Sie den gewünschten Port für die Serververbindung ein, falls Sie die Voreinstellung nicht verwenden möchten.
4. Wählen Sie aus der Klappliste das gewünschte "Zeitüberschreitungs-Profil" für den Server aus.
   Damit legen Sie das Zeitüberschreitungsverhalten für den Server fest.
   Sie haben die Möglichkeit zwischen drei Profilen zu wählen: "Schnell", "Mittel" und "Langsam", wobei "Schnell" voreingestellt ist.
5. Bereich "Sicherheit": Wenn Sie "https" als Protokoll ausgewählt haben, ist der Bereich "Sicherheit" aktiviert.
   Wählen Sie die gewünschte Option für das Zertifikat aus.
   Sie haben die Möglichkeit, ein neues Zertifikat zu erstellen oder ein bereits auf Ihrem Rechner vorhandenes Zertifikat zu verwenden.
   Falls Sie die erste Option aktivieren, wird nach dem erfolgreichen Beenden der Konfiguration ein neues Zertifikat erstellt.
   Falls Sie die zweite Option aktivieren, können Sie das gewünschte Zertifikat aus der Liste der angezeigten Zertifikate durch Anklicken auswählen.
6. Legen Sie unter "Speicher" den gewünschten Speicherort für Multiuser-Projekte fest. Sie können den Pfad direkt eintragen oder über "Durchsuchen" an die gewünschte Stelle navigieren.
7. Wählen Sie aus der nachfolgenden Klappliste die Anzahl der zu speichernden Projektversionen aus.
   Als Voreinstellung werden zehn Projektversionen gespeichert.
   Sie können eine Zahl zwischen "1" und "Unbegrenzt" für die zu speichernden Projektversionen wählen.
8. Klicken Sie auf "Service installieren", um die Konfiguration abzuschließen.

Ergebnis

Eine Meldung informiert Sie darüber, dass die Konfiguration des Multiuser-Servers erfolgreich durchgeführt wurde.

Im Bereich "Serverstatus" finden Sie unter "Client Info" die URL für den Multiuser-Server und den Fingerprint für das erzeugte Zertifikat.

Beispiel: https://XY12345.ww00X.siemens.net:8735/

C35F81B179062BA92A3EAE4DBF18B0DEA9F50D8A

Wenn Sie von Clientseite aus auf den Server zugreifen, sollten Sie immer das Zertifikat überprüfen, um eine sichere Kommunikation mit dem Multiuser-Server zu gewährleisten. Die Konfiguration des Multiuser-Servers ist damit abgeschlossen.

Fehlerbehebung im Configuration Tool

Falls Warnungen oder Fehler aufgetreten sind, finden Sie spezifische Informationen dazu im ServerConfiguration.log.

Gehen Sie vor wie folgt:

1. Klicken Sie in dem gelb hinterlegten Bereich auf "Log öffnen", um das ServerConfiguration.log zu öffnen.
2. Beheben Sie die im Logfile angezeigten Fehler und Warnungen.
3. Starten Sie das Configuration Tool erneut, falls es nicht mehr geöffnet ist.
4. Klicken Sie auf die Schaltfläche "Service installieren".

Ergebnis

Eine Meldung informiert Sie darüber, dass die Konfiguration des Multiuser-Servers erfolgreich durchgeführt wurde.

Die Konfiguration des Multiuser-Servers ist damit abgeschlossen.

Hinweis

Änderungen an bereits vorhandenen Eingaben

Um Änderungen an bereits vorhandenen Eingaben im Configuration Tool zu übernehmen, müssen Sie auf die Schaltfläche "Übernehmen" klicken.

2.13.2.3 Multiuser-Server mit grafischem Administration Tool verwalten

Einführung

Für die Verwaltung des Multiuser-Servers verwenden Sie das grafische "Administration Tool".

Hinweis

Administration Tool

- Für das Arbeiten mit dem Administration Tool benötigen Sie für Administratorrechte.
- Das Administration Tool steht Ihnen auch außerhalb des Multiuser-Servers zur Verfügung.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und zumindest eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Oberfläche des Administration Tools

Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.

- Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
- Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.

Die Benutzeroberfläche des Administration Tools sieht aus wie folgt:

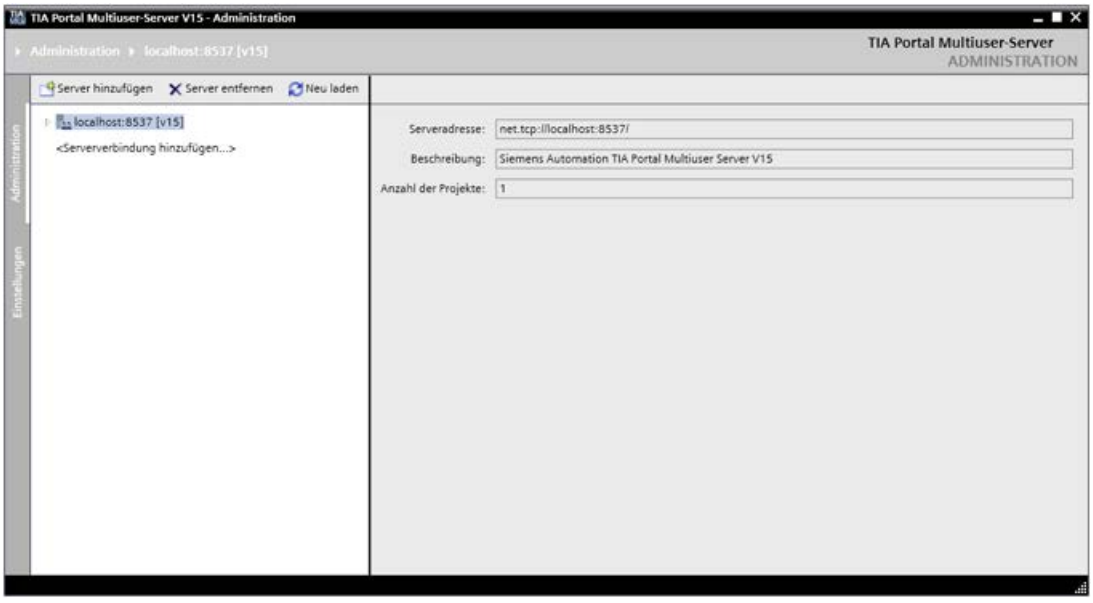

Funktionalität des Administration Tools

Auf der linken Seite des Tools sehen Sie zwei Register, die es ermöglichen, zwischen den "Einstellungen" und der "Administration" umzuschalten.

Folgende Funktionen sind mit dem "Administration Tool" ausführbar:

Unter "Administration" finden Sie eine Auflistung der konfigurierten Serververbindungen, sobald Sie über "Serververbindung hinzufügen" eine Verbindung zum Multiuser-Server eingerichtet haben. Auf der rechten Seite erhalten Sie in einer Detailansicht zusätzliche Informationen zu den links angewählten Objekten.

Beide Seiten enthalten eine eigene Funktionsleiste, um die angezeigten Verwaltungsfunktionen auszuführen.

Die Ansicht kann umgeschaltet werden, indem Sie auf die Schaltfläche "Administration" bzw. auf die Schaltfläche "Einstellungen" klicken.

Im Register "Administration" können Sie:

- Server hinzufügen und löschen
- Serverbindungen hinzufügen und löschen
- Daten aktualisieren
- Benutzerverwaltung für die Serververbindungen festlegen.
- Vorhandene Multiuser-Projekte und Sessions verwalten.
  - Benutzerrollen für Projekte und Sessions festlegen: ("Manager", "Contributor", "Member")
  - Projekte: Sperren, entsperren und löschen
  - Sessions: Anzeigen und löschen
- Commissioning-Modus für Projekte ein- und ausschalten

- "Historie" anzeigen: Projekthistorie mit allen Daten und Anzeige der gespeicherten Projektrevisionen
- "Historie" exportieren: Die Projekthistorie mit den ausgewählten Daten wird als XML-Datei exportiert.
- Revisionen verwalten.
- Revisions-Hinweise hinzufügen.

Im Register "Einstellungen" können Sie;

- Oberflächensprache auswählen
- Bestätigungen auswählen
- Zertifikate verwalten

Hinweis

Schaltfläche "Neu laden"

Durch Klicken auf diese Schaltfläche wird die Anzeige für das selektierte Objekt im Administration Tool aktualisiert.

Die genauen Vorgehensweisen für das Arbeiten mit dem Administration Tool sind nachfolgend beschrieben.

Siehe auch

Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

2.13.2.4 Neue Serververbindung hinzufügen und löschen

Einführung

Mit dem grafischen "Administration"-Tool können Sie neue Multiuser-Serververbindungen hinzufügen und löschen.

Diese Befehle können sowohl über die Befehle in der Funktionsleiste als auch über das Kontextmenü ausgeführt werden.

Hinweis

Administration Tool

Für das Arbeiten mit dem Administration Tool benötigen Sie Administratorrechte.

Voraussetzung

Der Multiuser-Server ist installiert.

Der Multiuser-Server wurde bereits konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Neue Serververbindung hinzufügen

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf "Serververbindung hinzufügen".

3. Tragen Sie im folgenden Dialog die gewünschten Verbindungsdaten für Ihre Serververbindung ein, wie in der Grafik beispielhaft gezeigt.

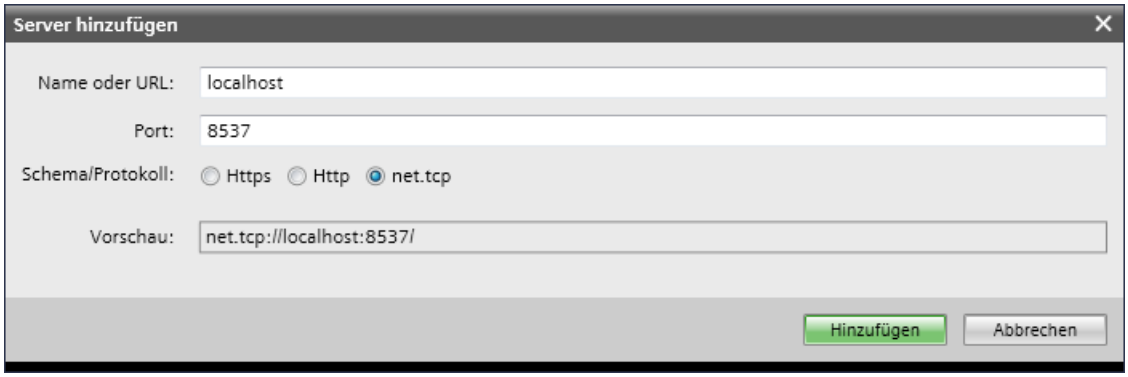

Hinweis

Verbindungsdaten für den lokalen Multiuser Server

Falls Sie eine Verbindung zum lokalen Multiuser Server aufbauen möchten, müssen Sie diesen zuerst starten.

Der lokale Multiuser Server wird automatisch gestartet, sobald Sie im TIA Portal eine lokale Session öffnen.

Alternativ können Sie den lokalen Multiuser Server auch starten, indem Sie im Installationsverzeichnis des TIA Portals unter "bin" auf die Datei "Siemens.Automation.Portal.LocalServer.exe" doppelklicken.

Die Verbindungsdaten für den lokalen Multiuser-Server finden Sie nach dem Starten des angezeigten Kommandofenster.

4. Klicken Sie auf "Hinzufügen", um die Serververbindung hinzuzufügen.
5. Klicken Sie auf das Dreieck von der neuen Serververbindung, um die Verbindung zu aktivieren und die Projektliste anzuzeigen.
6. Falls Sie eine "https"-Verbindung nutzen, werden Sie aufgefordert, die Gültigkeit des angezeigten Zertifikates zu bestätigen.
   Das verwendete Verbindungsprotokoll haben Sie bei der Konfiguration des Multiuser-Servers unter "Verbindungseinstellungen" bereits festgelegt.
   Vergleichen Sie den Fingerprint des angezeigten Zertifikats mit dem bei der Konfiguration erzeugten Zertifikat.
   Klicken Sie auf "OK", wenn beide Fingerprint übereinstimmen.
7. Falls die von Ihnen verwendeten Login-Daten beim Multiuser-Server nicht bekannt sind, werden Sie aufgefordert, alternative Verbindungsdaten für den Zugriff auf den Server einzugeben.
   Tragen Sie im angezeigten Dialog unter alternative Login-Daten ein, die dem Multiuser-Server bekannt sind.

Ergebnis

Die neue Serververbindung wird im linken Bereich des Administration Tools angezeigt.

Rechts sehen Sie die Anzahl der vorhandenen Multiuser-Projekte.

Wenn Sie im linken Bereich die neue Serververbindung "aufklappen", sehen Sie alle vorhandenen Projekte aufgelistet und weitere Einstellmöglichkeiten.

Vorhandene Serververbindung löschen

Gehen Sie vor wie folgt:

1. Selektieren Sie im Administration Tool auf der linken Seite die gewünschte Serververbindung.
2. Klicken Sie auf die Schaltfläche "Server entfernen" oder rufen Sie den Befehl "Remove" über das Kontextmenü auf.
3. Bestätigen Sie die folgende Abfrage mit "Ja".

Ergebnis

Die ausgewählte Serververbindung wird gelöscht.

Erreichbarkeit des Multiuser-Servers

Falls der Multiuser-Server nicht erreichbar ist, erhalten Sie eine entsprechende Fehlermeldung:

Gehen Sie in diesem Fall vor wie folgt:

1. Überprüfen Sie die angezeigten Verbindungsdaten.
2. Stellen Sie sicher, dass der ausgewählte Server gestartet ist.
3. Löschen Sie die ungültige Serververbindung, indem Sie diese anklicken und auf "Server entfernen" klicken.
4. Bestätigen Sie die folgende Abfrage mit "Ja".
5. Klicken Sie auf "Serververbindung hinzufügen" und erstellen Sie eine neue Serververbindung zu einem gestarteten Multiuser-Server mit gültigen Verbindungsdaten.

Ergebnis

Die neue Serververbindung mit gültigen Verbindungsdaten wird angezeigt.

2.13.2.5 Benutzerverwaltung für Multiuser-Server und Projekte festlegen

Einführung

Mit dem grafischen "Administration Tool" legen Sie unterhalb des Projektes in der "Benutzerverwaltung" die Rechte für das Arbeiten mit Multiuser-Serververbindungen, mit Multiuser-Projekten und mit Lokalen Sessions fest. Das Rechtekonzept basiert auf den Windows-Zugriffsrechten für Ordner und Dateien.

Sie finden den Eintrag "Benutzerverwaltung" im Administration Tool sowohl unterhalb jeder Serververbindung als auch unterhalb jedes Projektes und jeder lokalen Session. Abhängig davon, wo Sie das "Benutzerverwaltung" aufrufen, legen Sie die entsprechenden Berechtigungen fest.

Mit dem "Benutzerverwaltung" legen Sie explizite personengebundene Berechtigungen für folgende Rollen fest:

- Manager (Vollzugriff)
- Contributor (Teilzugriff, kein Löschen)
- Member (Lese-Berechtigung)

Hinweis

Benutzerverwaltung

Bitte achten sie darauf, dass durch die Vergabe von Nutzerrechten für ein Projekt **nicht** automatisch Benutzerrechte für den gesamten Multiuser-Server eingerichtet werden.

Für jede Rolle können über die Schaltflächen folgende Funktionen ausgeführt werden:

- "Hinzufügen": Fügt neue Benutzer mit den entsprechenden Berechtigungen hinzu.
- "Ausgewählten Benutzer entfernen": Entfernt den ausgewählten Benutzer
- "Ausgewählten Benutzer hochstufen": Verschiebt den ausgewählten Benutzer nach oben in die nächsthöhere Berechtigungsgruppe.
- "Ausgewählten Benutzer zurückstufen": Verschiebt den ausgewählten Benutzer nach unten in die nächstniedrigere Berechtigungsgruppe.

Die Rechte, die für die Serververbindungen festgelegt sind, werden auf die Berechtigungen für Projekte und lokale Sessions vererbt.

Hinweis

Administration Tool

- Für das Arbeiten mit der "Benutzerverwaltung" benötigen Sie Administratorrechte.
- Nach dem Ändern von Berechtigungen muss das Projekt neu gestartet werden.

Voraussetzung

Der Multiuser-Server ist installiert.

Der Multiuser-Server wurde bereits konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Zugriffsrechte vergeben

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Doppelklicken Sie unterhalb des gewünschten Eintrags auf der linken Seite unter "Benutzerverwaltung".
   - Wenn Sie unterhalb eines Servers auf "Benutzerverwaltung" klicken, vergeben Sie die Rechte für das Arbeiten mit dem Server.
   - Wenn Sie unterhalb eines Projekts auf "Benutzerverwaltung" doppelklicken, vergeben Sie die Rechte für das Arbeiten mit dem Projekt.
3. Das "Benutzerverwaltung öffnet sich und zeigt auf der rechten Seite die möglichen Einstellungen an. Sie sehen bereits, welche Rechte für die zugehörige Serververbindung vergeben wurden. Diese Rechte werden für die Projekte vererbt.

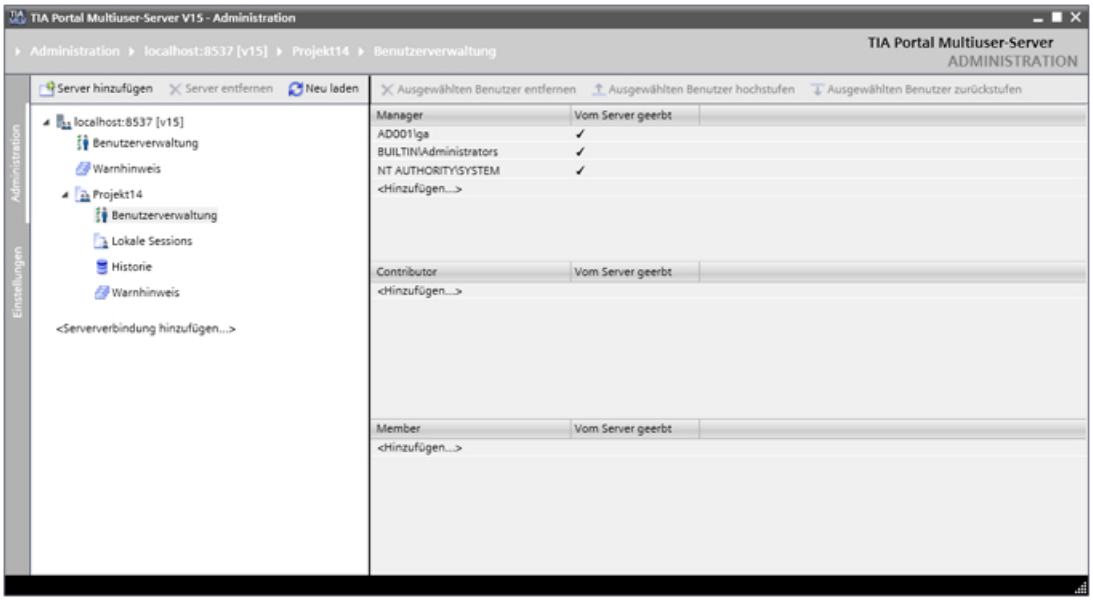

4. Klicken Sie auf "Hinzufügen", um für die angezeigten Rollen neue Benutzer hinzuzufügen.
5. Tragen Sie die benötigten Daten in den nachfolgenden Windows-Dialog ein und klicken Sie auf "OK".
6. Über die Schaltflächen "Ausgewählten Benutzer hochstufen" und "Ausgewählten Benutzer zurückstufen" können Sie Benutzer nach unten und oben in andere Berechtigungsgruppen verschieben.
7. Über die Schaltfläche "Ausgewählten Benutzer entfernen" können Sie den ausgewählten Benutzer wieder entfernen.
8. Vergeben Sie die gewünschten Rechte und Rollen über die "Benutzerverwaltung" sowohl für den Server als auch für die unterlagerten Projekte und Sessions.

2.13.2.6 Warnhinweise verwalten

Einführung

Mit dem grafischen "Administration Tool" können Sie im Register "Administration" > "Warnhinweis" vorhandene Warnhinweise für den Multiuser-Server anzeigen und verwalten. Mit Hilfe von Warnhinweisen können Informationen vom Server an den Administrator weitergegeben werden.

Sobald hinter dem Eintrag "Warnhinweis" eine Zahl in Klammern angezeigt wird, ist diese Anzahl von Warnhinweisen vorhanden.

Der Eintrag "Warnhinweis" ist sowohl unterhalb des Servers als auch unterhalb der einzelnen Projekte vorhanden.

Folgende Warnhinweise werden angezeigt:

- Ein Serveralarm, sobald weniger als 3 GB freier Speicher auf dem Laufwerk vorhanden sind, auf dem das Verzeichnis liegt.
- Ein Projektalarm für jede lokale Session, die mehr als 5 Revisionen zurück liegt.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Zertifikate verwalten

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf der linken Seite im Register "Administration" auf "Warnhinweis".
   Warnhinweise sind sowohl für den Multiuser-Server als auch für die einzelnen Projekte vorhanden.
   Alle für den ausgewählten Multiuser-Server vorhandenen Warnhinweise werden angezeigt.
3. Falls Sie einen Warnhinweis nicht mehr benötigen und löschen möchten, selektieren Sie diesen Warnhinweis und klicken Sie dann auf die Schaltfläche "Ausgewählten Warnhinweis löschen".

Ergebnis: Der selektierte Warnhinweis wird aus der Liste gelöscht.

2.13.2.7 Multiuser-Projekte und lokale Sessions verwalten

Einführung

Mit dem "Administration Tool" können Sie die zu einem Projekt angelegten Sessions verwalten.

Unter "Sessions" sehen Sie die zum ausgewählten Multiuser-Projekt vorhandenen lokalen Sessions.

Sie finden den Eintrag "Sessions" im "Administration Tool" unterhalb jedes Projektes.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Projektdetails anzeigen und Projekte verwalten

Gehen Sie vor wie folgt:

1. Rufen Sie das "Administration Tool" für auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf das gewünschte Projekt, damit auf der rechten Seite die Projektdetails angezeigt werden.
   Folgende Informationen zu den Projekten werden angezeigt:
   "Projektname": Zeigt den Projektnamen
   "Status": Zeigt den Projektstatus an ("gesperrt" oder "entsperrt").
   "Betriebsart": Zeigt an, ob der Engineering-Modus oder der Commissioning-Modus ausgewählt wurde.
   "Erstellt von / Datum": Bearbeiter, der das Projekt erstellt hat und Erstelldatum für das Projekt.
   "Zuletzt bearbeitet von / Datum": Bearbeiter, der das Projekt zuletzt geändert hat und Änderungsdatum für das Projekt.
   "Anzahl der lokalen Sessions": Anzahl der lokalen Sessions, die zu diesem Projekt gehören.
   "Kommentar": Anzeige des Projektkommentars, falls vorhanden.

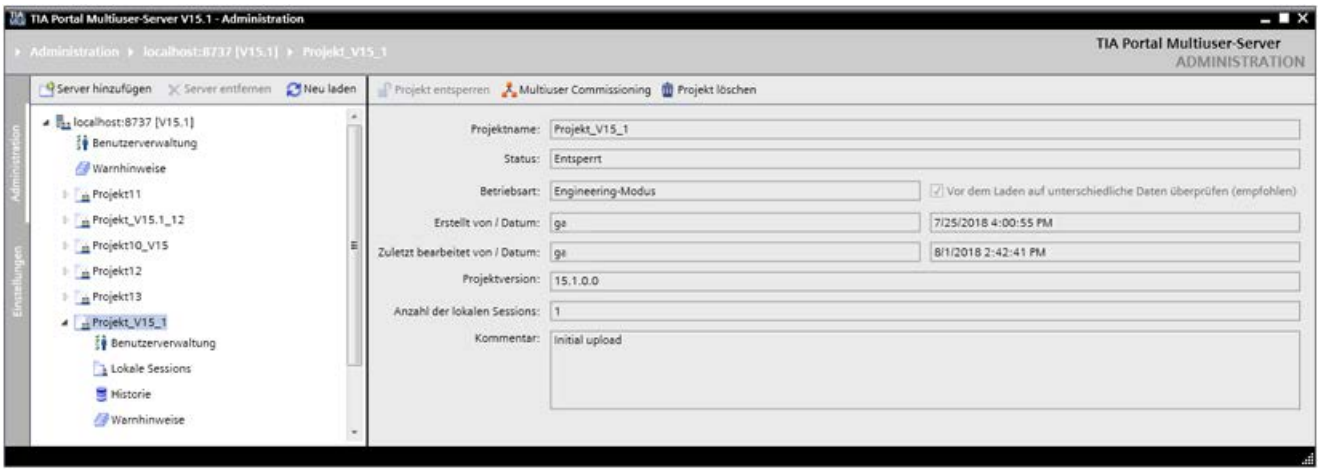

3. Um ein Projekt zu löschen, selektieren Sie das entsprechende Projekt und klicken Sie danach auf die Schaltfläche "Projekt löschen".
   Projekte können nur dann gelöscht werden, wenn keine lokalen Sessions zu diesen Projekten vorhanden sind.
4. Bestätigen Sie die nachfolgende Abfrage, um das Projekt zu löschen.
5. Um ein gesperrtes Projekt zu entsperren, selektieren Sie das entsprechende Projekt und klicken Sie danach auf die Schaltfläche "Projekt entsperren".
6. Bestätigen Sie die nachfolgende Abfrage, um das Projekt zu entsperren.

7. Klicken Sie auf die Schaltfläche "Multiuser Commissioning", um in den "Commissioning Modus" umzuschalten. Die Schaltfläche wird daraufhin aktiv und blau hinterlegt angezeigt. Gleichzeitig mit dem Einschalten des Commissioning-Modus wird die Option "Vor dem Laden auf unterschiedliche Daten überprüfen (empfohlen)" aktiviert.
8. Klicken Sie erneut auf die Schaltfläche "Multiuser Commissioning", um den den "Commissioning Modus" zu beenden und wieder in den "Engineering Modus" umzuschalten.

Lokale Sessions anzeigen und löschen

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   – Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   – Klicken Sie auf den Eintrag "TIA Portal Multiuser Server V<x.y> - Administration", um das Administration Tool zu öffnen.
2. Klicken Sie unterhalb des gewünschten Multiuser-Projekts auf der linken Seite auf "Lokale Sessions".
3. Folgende Informationen zu den vorhandenen lokalen Sessions werden auf der rechten Seite im "Administration Tool" angezeigt.
   "Id": Zählt die vorhandenen Sessions hoch.
   "Lokaler Pfad": Zeigt den eingestellten lokalen Speicherplatz für die Sessions an.
   "Computername": Zeigt den Namen des Computers an, auf dem die Session liegt.
   "Zuletzt verwendet von": Zeigt den Namen des letzten Bearbeiters an.
   "Projektrevision": Zeigt an, zu welcher Projektrevision die jeweilige Session gehört.

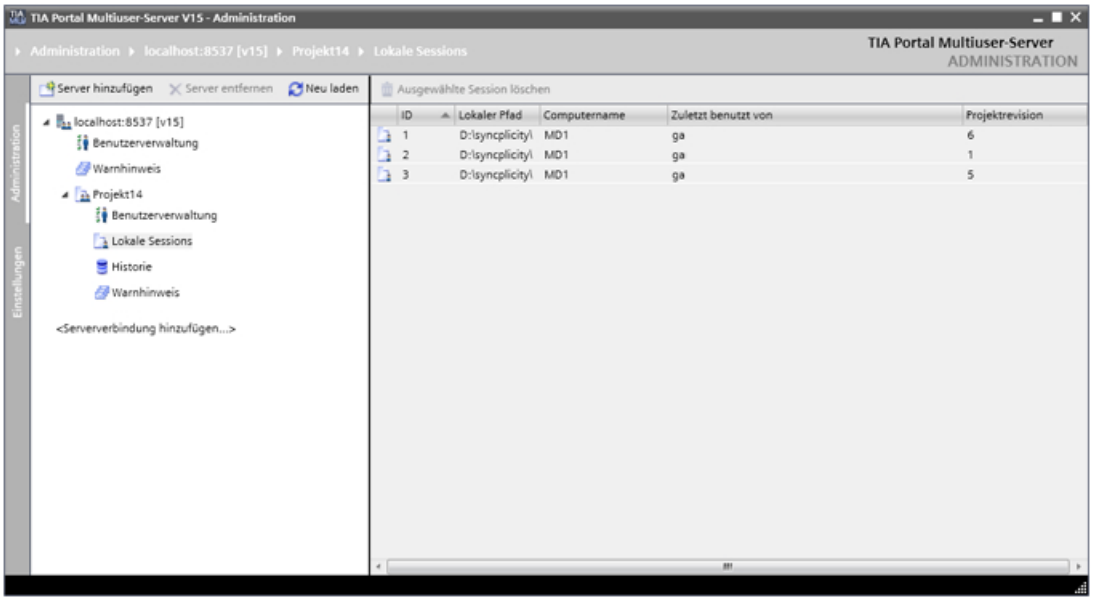

4. Selektieren Sie die gewünschte Session und klicken Sie auf die Schaltfläche "Ausgewählte Session löschen", um diese zu löschen.
5. Klicken Sie im folgenden Dialog auf "OK", um die ausgewählte Session zu löschen.

Hinweis

Anzahl der Projektrevisionen

Die Anzahl der vorgehaltenen Projektrevisionen legen Sie bei der Konfiguration des Multiuser-Servers fest.

Siehe hierzu: Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

Siehe auch

Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

2.13.2.8 Historie für Projekte anzeigen und Revisionen verwalten

Einführung

Mit dem grafischen "Administration Tool" können Sie die Historie zu den angelegten Projekten anzeigen und die vorhandenen Projektrevisionen verwalten.

Folgende Funktionen stehen zur Verfügung:

- Zur ausgewählten Revision zurückrollen
- Ausgewählte Revision exportieren
- Historie exportieren
- Revision sichern
- Details anzeigen
- Daten der Revision löschen (über Kontextmenü)

Sie finden den Eintrag "Historie" im "Administration Tool" unterhalb jedes Projektes.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Historie anzeigen

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie unterhalb des gewünschten Multiuser-Projektes auf der linken Seite auf "Historie".
   Die Historie zeigt alle vorhandenen Projektrevisionen an.
   Folgende Informationen zu den vorhandenen Revisionen werden auf der rechten Seite im "Administration Tool" angezeigt.
   "Verfügbarkeit": Zeigt an, ob die Session auf dem Server noch verfügbar ist.
   "Revisionsnummer": Zeigt die Revisionsnummer an.
   "Computername": Zeigt den Namen des Computers an, auf dem die Revision erstellt wurde.
   "Erstellt von": Zeigt den Namen des Erstellers an.
   "Erstelldatum": Zeigt das Erstelldatum der Revision an.
   "Kommentar": Zeigt den Kommentar an, falls vorhanden.
   "Hinweise": Zeigt die Anzahl der Hinweise zur Revision an, falls vorhanden. Die Hinweise können in der Detail-Sicht angezeigt werden.

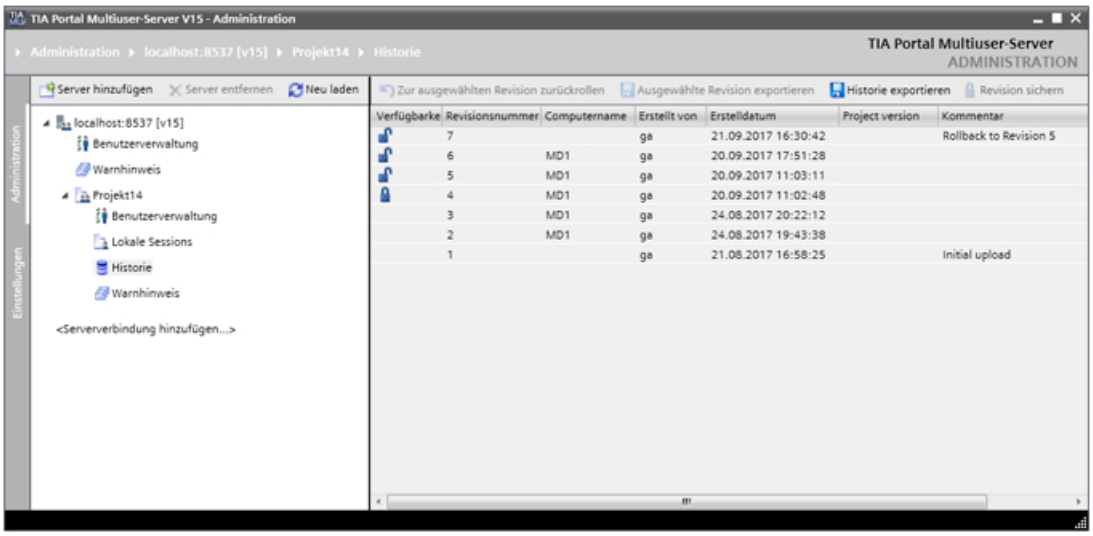

3. Über die Schaltfläche "Zur ausgewählten Revision zurückrollen" können Sie eine selektierte Session auf die ausgewählte Revision zurückrollen.
   Danach wird eine neue Zeile angelegt, die die neue Revision anzeigt, mit dem Kommentar "Zurückgerollt zur Revision <Nummer>".
4. Um eine Revision zu exportieren, klicken Sie auf die Schaltfläche "Ausgewählte Revision exportieren".
   Wählen Sie im folgenden Dialog den gewünschten Ablagepfad aus und klicken Sie auf "OK".

5. Um die Projekthistorie als XML-Datei zu exportieren, wählen Sie die gewünschten Revisionen aus und klicken Sie auf die Schaltfläche "Historie exportieren".
   Tragen Sie im nachfolgenden Dialog die benötigten Daten ein und klicken Sie auf "Speichern".
6. Um eine Revision zu sichern, klicken Sie auf die Schaltfläche "Revision sichern".
   Tragen Sie im folgenden Dialog einen Hinweis zu der zu sichernden Revision ein und klicken Sie auf "Revision sichern".

Daten der Revision löschen

Um Speicherplatz zu sparen, können Sie die Daten für nicht mehr benötigte Revisionen löschen.

Gehen Sie vor wie folgt:

1. Selektieren Sie die gewünschte Revision oder mehrere Revisionen und klicken Sie im Kontextmenü auf den Befehl "Daten der Revision löschen".
2. Bestätigen Sie die Abfrage im nachfolgenden Dialog mit "OK".

Ergebnis:

Die Daten der ausgewählten Revision werden gelöscht. Alle relevanten Informationen zur Änderungshistorie der Revision bleiben erhalten.

Die Revision wird unter "Verfügbarkeit" nicht länger angezeigt und auf diese Revision kann nicht mehr zurückgerollt werden.

Hinweis

Die Daten für folgende Revisionen können nicht gelöscht werden:

- Für Revisionen, die in der Spalte "Verfügbarkeit" ohne Symbol angezeigt werden. Auf diese Revisionen kann nicht mehr zugegriffen werden.
- Für Revisionen, die die Kennzeichnung "Retain" haben. Diese Revisionen haben Sie bereits als "gesichert" gekennzeichnet.
- Für die letzte vorhandene Revision zu einem Projekt.

Falls Sie versuchen, die Daten zu einer der o.g. Revisionen zu löschen, erhalten Sie eine Fehlermeldung.

Details zu einer Revision anzeigen

Innerhalb der Historie zu den Projektversionen können Sie über die Schaltfläche " Details anzeigen" weitere Informationen zur ausgewählten Revision in einem Detailfenster anzeigen.

- Im Register "Details" können Sie Hinweise hinzufügen.
  Klicken Sie dazu auf den Link und tragen Sie den gewünschten Hinweis im nachfolgenden Dialog ein.
  Klicken Sie auf die Schaltfläche "Hinweis hinzufügen", um den Hinweis zu übernehmen.
- Im Register "Geänderte Objekte" sehen Sie alle im ausgewählten Projekt vorgenommenen Änderungen.

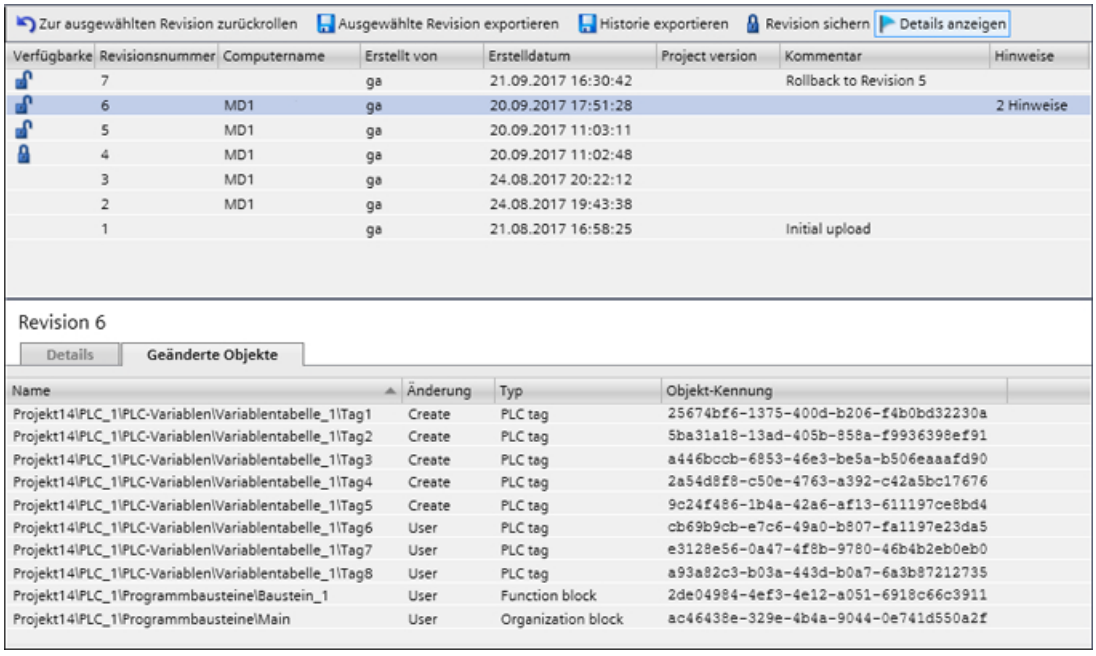

Verfügbarke	Revisionsnummer	Computername	Erstellt von	Erstelldatum	Project version	Kommentar	Hinweise
	7		ga	21.09.2017 16:30:42		Rollback to Revision 5	
	6	MD1	ga	20.09.2017 17:51:28			2 Hinweise
	5	MD1	ga	20.09.2017 11:03:11			
	4	MD1	ga	20.09.2017 11:02:48			
	3	MD1	ga	24.08.2017 20:22:12			
	2	MD1	ga	24.08.2017 19:43:38			
	1		ga	21.08.2017 16:58:25		Initial upload	

Name	Änderung	Typ	Objekt-Kennung
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag1	Create	PLC tag	25674bf6-1375-400d-b206-f4b0bd32230a
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag2	Create	PLC tag	5ba31a18-13ad-405b-858a-f9936398ef91
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag3	Create	PLC tag	a446bccb-6853-46e3-be5a-b506eaaafd90
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag4	Create	PLC tag	2a54d8f8-c50e-4763-a392-c42a5bc17676
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag5	Create	PLC tag	9c24f486-1b4a-42a6-af13-611197ce8bd4
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag6	User	PLC tag	cb69b9cb-e7c6-49a0-b807-fa1197e23da5
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag7	User	PLC tag	e3128e56-0a47-4f8b-9780-46b4b2eb0eb0
Projekt14\PLC_1\PLC-Variablen\Variablentabelle_1\Tag8	User	PLC tag	a93a82c3-b03a-443d-b0a7-6a3b87212735
Projekt14\PLC_1\Programmbausteine\Baustein_1	User	Function block	2de04984-4ef3-4e12-a051-6918c66c3911
Projekt14\PLC_1\Programmbausteine\Main	User	Organization block	ac46438e-329e-4b4a-9044-0e741d550a2f

Revision sichern

Revisionen können folgende Zustände haben:

- Verfügbar
  Dies bedeutet, dass die Daten dieser Revision momentan auf dem Server verfügbar sind. Diese Daten können jedoch automatisch gelöscht werden, sobald die maximale Anzahl von Revisionen auf dem Server erreicht wird.
  In der Spalte "Verfügbarkeit" wird als Symbol ein geöffnetes Schloss angezeigt.
- Gesichert
  Dies bedeutet, dass die Daten dieser Revision gesichert wurden und die Revision auf dem Server erhalten bleibt. Diese Revision zählt nicht mit bei der Anzahlt der maximal verfügbaren Revisionen.
  In der Spalte "Verfügbarkeit" wird als Symbol ein geschlossenes Schloss angezeigt.
- Nicht mehr verfügbar
  Dies bedeutet, dass die Daten dieser Revision bereits gelöscht wurden und die Revision nicht länger auf dem Server verfügbar ist.
  In diesem Fall wird die Revision zwar noch aufgelistet, in der Spalte "Verfügbarkeit" wird jedoch kein Symbol angezeigt.

Um zu verhindern, dass Revisionen automatisch gelöscht werden, besteht die Möglichkeit, erhaltenswerte Revisionen zu sichern.

Gehen Sie vor wie folgt:

1. Wählen Sie die zu sichernde Revision an, die noch auf dem Server verfügbar sein muss, und klicken Sie auf die Schaltfläche "Revision sichern".
2. Tragen Sie falls gewünscht einen Kommentar zur Revision ein und bestätigen Sie die Abfrage für das Sichern der ausgewählten Revision.

2.13.2.9 Zwischen Engineering- und Commissioning-Modus umschalten

Einführung

Mit dem grafischen "Administration Tool" können Sie zwischen dem Engineering- und dem Commissioning-Modus umschalten. Das Einstellen des Modus gilt immer für alle lokalen Sessions eines Serverprojektes. Es können immer nur alle lokalen Sessions entweder im Engineering-Modus oder im Commissioning-Modus sein. Der Engineering-Modus ist per Voreinstellung eingestellt.

Wenn Sie in den Commissioning-Modus umschalten, wird automatisch auch die Schaltfläche "Vor·dem·Laden·auf·unterschiedliche·Daten·überprüfen·(empfohlen)" aktiviert. Dies ist eine empfohlene Einstellung für das Arbeiten mit Multiuser Commissioning. Die damit eingestellte Überprüfung stellt sicher, dass keine vom Serverprojekt abweichenden Daten in der CPU geladen werden. Dies kann z. B. der Fall sein, wenn aus einer vorherigen lokalen Session ins Serverprojekt eingecheckt wurde, ohne auf die CPU zu laden.

Wenn es Abweichungen gibt zwischen den Daten, die im Serverprojekt vorhanden sind und den Daten, die auf die CPU geladen werden, wird ein Dialog angezeigt, der Sie über die weiteren Vorgehensweisen informiert.

Unterschiedliche Daten können in der Serverprojekt-Ansicht bereinigt werden.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Commissioning-Modus ein- und ausschalten

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Unter "Administration" finden Sie eine Auflistung der konfigurierten Serververbindungen, sobald Sie über "Serververbindung hinzufügen" eine Verbindung zum Multiuser-Server eingerichtet haben. Auf der rechten Seite erhalten Sie in einer Detailansicht zusätzliche Informationen zu den links angewählten Objekten.
3. Wählen Sie auf der linken Seite den Server aus und selektieren Sie das Projekt, für das Sie den Commissioning-Modus einstellen möchten.
4. Klicken Sie auf der rechten Seite in der Kopfzeile auf die Schaltfläche "Multiuser Commissioning", um den Commissioning-Modus zu aktivieren.
   - Sobald die Schaltfläche aktiv ist, wird sie blau hinterlegt angezeigt und das auf der linken Seite ausgewählte Projekt erhält das Commissioning-Symbol als Overlay.
   - Mit Commissioning-Modus wird auch die Option "Vor·dem·Laden·auf·unterschiedliche·Daten·überprüfen·(empfohlen)" aktiviert.

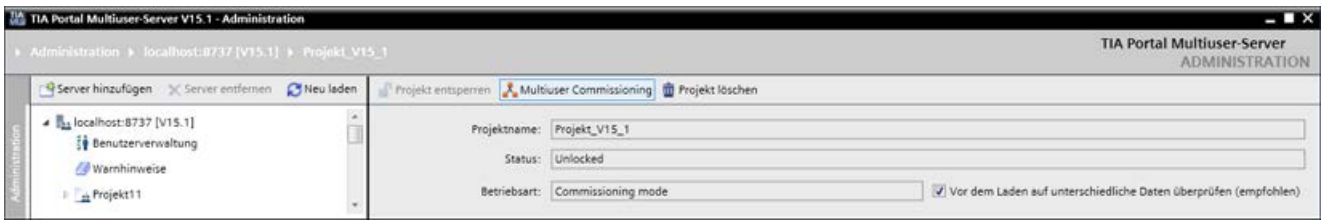

Um den "Commissioning-Modus" wieder zu beenden, klicken Sie erneut auf die Schaltfläche "Multiuser Commissioning" in der Funktionsleiste.

- Sobald die Schaltfläche deaktiviert ist, wird sie ohne farbige Hinterlegung dargestellt und für das auf der linken Seite ausgewählte Projekt wird kein Overlay-Smbol angezeigt.
- Mit Commissioning-Modus wird auch die Schaltfläche "Vor·dem·Laden·auf·unterschiedliche·Daten·überprüfen·(empfohlen)" deaktiviert.

Ergebnis

Der "Commissioning-Modus" wird durch das Klicken auf die Schaltfläche "Multiuser Commissioning" ein- und bei erneutem Klicken wieder ausgeschaltet.

2.13.2.10 Oberflächensprache auswählen

Einführung

Mit dem grafischen "Administration Tool" können Sie im Register "Einstellungen" > "Allgemein" die gewünschte Oberflächensprache auswählen.

Es stehen alle sechs Sprachen des TIA Portals zur Verfügung.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Sprache umschalten

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf der linken Seite auf das Register "Einstellungen" > "Allgemein". Wählen Sie im Bereich "Sprache" aus der Klappliste die gewünschte Oberflächensprache aus.

Ergebnis: Die Oberflächentexte werden in der ausgewählten Sprache angezeigt.

2.13.2.11 Bestätigungen anzeigen

Einführung

Mit dem grafischen "Administration Tool" können Sie im Register "Einstellungen" > "Allgemein" auswählen, für welche Aktionen beim Arbeiten mit Multiuser Engineering Bestätigungen eingeschaltet werden sollen.

Soll eine der nachfolgend ausgewählten Aktionen durchgeführt werden, muss diese Aktion vor der Ausführung durch eine Rückfrage in Form eines Dialogs nochmal bestätigt werden.

Bestätigungen für das Arbeiten in folgenden Bereichen können ausgewählt werden:

- Benutzerverwaltung
- Serververbindung löschen
- Lokale Session löschen
- Projekt entsperren
- Revision zu einem nicht leeren Ordner exportieren
- Benachrichtigung löschen
- Revision veröffentlichen

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Bestätigung auswählen

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf der linken Seite auf das Register "Einstellungen" > "Allgemein".
3. Klicken Sie unter "Bestätigungen" die gewünschten Aktionen an.
   - Über die Schaltfläche "Alle" können Sie Bestätigungen für alle Einträge aktivieren.
   - Über die Schaltfläche "Keine" können Sie Bestätigungen für alle Einträge deaktivieren.

Ergebnis: Sobald eine der ausgewählten Aktionen durchgeführt wird, muss diese Aktion vor der Ausführung durch eine Rückfrage in Form eines Dialogs nochmal bestätigt werden.

2.13.2.12 Zertifikate verwalten

Einführung

Mit dem grafischen "Administration Tool" können Sie im Register "Einstellungen" > "Zertifikate" vorhandene Zertifikate für den Multiuser-Server anzeigen und verwalten. Sobald Sie "https" für die Serververbindung ausgewählt haben, wird ein Zertifikat erstellt.

Nicht mehr benötigte Zertifkate können gelöscht werden, sobald sie nicht mehr verwendet werden. Wird ein Zertifikat in einer aktiven Serververbindung gerade verwendet, und Sie versuchen es zu löschen, erhalten Sie eine entsprechende Information.

Voraussetzung

Der Multiuser-Server ist installiert und konfiguriert.

Sie haben im TIA Portal bereits Projekte als Multiuser-Projekte angelegt und mindestens eine lokale Sessions erstellt.

Hinweis

Multiuser-Projekte und lokale Sessions

Falls Sie im TIA Portal noch keine Multiuser-Projekte und lokale Sessions auf dem Multiuser-Server angelegt haben, wird Ihnen bei der Administration des Multiuser-Servers keine diesbezügliche Information angezeigt.

Zertifikate verwalten

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Klicken Sie auf der linken Seite auf das Register "Einstellungen" > "Zertifikate".
   Alle für den ausgewählten Multiuser-Server vorhandenen Zertifikate werden mit den zugehörgen Daten angezeigt.
3. Falls Sie ein Zertifikat nicht mehr benötigen und löschen möchten, selektieren Sie dieses Zertifikat und klicken Sie dann auf die Schaltfläche "Ausgewählte Zertifikatsbestätigung entfernen".

Ergebnis: Das selektierte Zertifikat wird aus der Liste gelöscht. Wenn das Zertifikat gerade in einer aktiven Serververbindung verwendet wird, erhalten Sie die Meldung, dass das Zertifikat erst nach dem nächsten Neustart gelöscht wird.

2.13.3 Kommandozeilen-Tools verwenden

2.13.3.1 Einführung zu den Multiuser-Kommandozeilen-Tools

Einführung

Der Multiuser-Server kann mithilfe der Multiuser Administrative Tools und der Multiuser Power Tools konfiguriert und verwaltet werden.

Für die Konfiguration des Multiuser-Servers mit den Administrative Tools werden Administratorrechte benötigt.

Für Verwaltungsaufgaben mit den Power Tools werden Nutzerrechte benötigt.

Die Multiuser Power Tools stehen auch im TIA Portal zur Verfügung, um Verwaltungsaufgaben auf dem Multiuser-Server durchzuführen.

Voraussetzung

Der Multiuser-Server ist installiert.

Schritte für die Konfiguration des Multiuser-Servers mit den Administrative Tools

Die Server-Installation und -Verwaltung erfolgt durch einen Kommandozeilen-Befehl.

Um den Multiuser-Server zu konfigurieren, sind folgende Schritte notwendig:

- Rufen Sie die Multiuser Server Administrative Tools mit Administrator-Rechten auf.
- Die komplette Installation erfolgt mit dem Befehl:
  `musrv install -r d:\TIA\SRV -p 5080 -b https`
- Das Anlegen des Projekt-Verzeichnisses (Repository) erfolgt automatisch.
- Die Festlegung von Protokoll und Port-Adresse sind in diesem Befehl bereits enthalten, ebenso wie die integrierte Zertifikatverwaltung für eine gesicherte HTTPS Kommunikation.
- Das Erstellen neuer Zertifikate und die Auswahl vorhandener Zertifikate erfolgt mit Windows-Mitteln.
- Weitere Befehle zur Konfiguration des Multiuser-Servers finden Sie hier: Befehle zur Konfiguration des Multiuser-Servers (Seite 90)

Hinweis

Freischaltung des Firewall-Ports

Für den Zugriff auf den Multiuser-Server wird automatisch der angegebene Port in der Firewall freigeschaltet.

Verwaltung des Multiuser-Servers mit den Power Tools

Um den Multiuser-Server zu verwalten, sind nach der Installation des Servers folgende Schritte notwendig:

- Für den Zugriff auf den Multiuser-Server muss ggf. ein Port in der Firewall freigeschaltet sein.
- Für die Verwaltung des Multiuser-Servers muss die Multiuser-Verwaltung oder die Multiuser Server Power Tools aufgerufen werden.
- Weitere Befehle zur Verwaltung des Multiuser-Servers finden Sie hier: Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

Benutzer- und Rechteverwaltung

Die Benutzer- und Rechteverwaltung für den Multiuser-Server basiert auf der Benutzerverwaltung für die Windows-Benutzerkonten.

- Eine projektgranulare Rechtevergabe ist möglich.
- Die Verwendung von Gruppenrichtlinien ermöglicht eine einfache, schnelle und übersichtliche Rechtevergabe.

Projektverwaltung

Die Projektverwaltung ermöglicht eine schnelle Übersicht über vorhandene Projekte und zugehörige lokale Sessions.

Multiuser-Projekte und lokale Sessions können komfortabel verwaltet werden.

Die Projektverwaltung ist automatisierbar durch Skripte (Batch-Files).

Fehlermeldungen

Falls eine Aktion nicht erfolgreich ausgeführt werden kann, erhalten Sie eine entsprechende Fehlermeldung.

Die Fehlerbeschreibungen finden Sie in den entsprechenden Tools, indem Sie die Hilfe zu den Tools aufrufen:

Hilfeaufruf zu den Administrative Tools:

- Öffnen Sie die Administrative Tools wie folgt:
- Kopieren Sie den Befehl "musrv" in die Kommandozeile und drücken Sie die "Return-Taste".

Hilfeaufruf zu den Power Tools:

- Öffnen Sie die Power Tools wie folgt:
- Kopieren Sie den Befehl "mupt" in die Kommandozeile und drücken Sie die "Return-Taste".

Siehe auch

Befehle zur Konfiguration des Multiuser-Servers (Seite 90)

Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

2.13.3.2 Multiuser-Server mit den Administrative Tools konfigurieren

Einführung

Bevor der installierte externe Multiuser-Server im Rahmen von Multiuser Engineering genutzt werden kann, muss der Server konfiguriert werden.

Wenn Sie die Konfiguration des Multiuser-Servers über die Multiuser Administrative Tools durchführen, nutzen Sie die nachfolgend beschriebenen Befehle, die Sie über die Kommandozeile ausführen.

Folgende Aktionen werden bei der Eingabe des unten beschriebenen "Install"-Befehls automatisiert durchgeführt:

- Der Windows-Dienst für den Multiuser-Server wird installiert.
- Eine Firewall-Freischaltung für den angegebenen Port über eine TCP-Verbindung wird hinzugefügt.
- Die Performance-Überwachung zur Beobachtung des Multiuser-Servers wird installiert.
- Das als "Repository" bezeichnete Ablageverzeichnis wird erstellt, falls es noch nicht vorhanden ist.
- Das "Binding" wird definiert, ob Sie "http" oder "https" (empfohlen) als Transfer-Protokoll verwenden.
- Der "Fingerprint" zur Bestätigung eines gültigen Zertifikats wird an den Server übergeben.

Anschließend kann der Multiuser-Server eingesetzt werden.

Hinweis

Für die Konfiguration des Multiuser-Servers benötigen Sie Administratorrechte.

Voraussetzung

Der Multiuser-Server ist installiert.

Hinweis

Die Befehle sowie die zugehörigen Hilfen und Informationstexte innerhalb der Kommandozeile sind ausschließlich in Englisch verfügbar.

Vorgehensweise

Um den Multiuser-Server zu konfigurieren, gehen Sie vor wie folgt:

1. Rufen Sie das Tool für die Multiuser-Server-Konfiguration mit Administrator-Rechten auf. Doppelklicken Sie dazu im Installationsverzeichnis des TIA Portals unter "bin" auf die Datei "start-musrv.bat".
2. Kopieren Sie folgenden Befehl zur Serverkonfiguration mit den gewünschten Parametern in die Kommandozeile und drücken Sie die "Return-Taste".
   Befehl:
   ```
   => musrv install -r <repository-path> -p <port> -b <binding> -h <certificate-hash>
   ```
   Beispiel für die Eingabe:
   ```
   musrv install -r c:\TIA\SERVER -p 5080 -b https -h <certificate-hash>
   ```
3. Falls der Befehl "musrv install" nicht erfolgreich ausgeführt werden konnte, erhalten Sie eine Fehlermeldung", zu der Sie sich eine Erklärung anzeigen lassen können.
   Nachdem diese Befehle erfolgreich ausgeführt wurden, ist der Multiuser-Server konfiguriert.

Konfigurationsbeispiel

Falls Sie keine Parameter eintragen, fragt Sie das System nach den benötigten Parametern.

Befehl	Beschreibung
"Repository-path" definieren Im obigen Beispiel verwendet: `-r c:\TIA\SERVER`	Tragen Sie den Namen und den Pfad für das Verzeichnis (Repository) ein, in dem der Server die Daten ablegen soll. Falls Sie ein neues Verzeichnis eingeben, das noch nicht vorhanden ist, wird das Verzeichnis automatisch vom System erstellt. **Wichtiger Hinweis:** Dieses Verzeichnis und die Datenablage bleibt auch nach dem Löschen des Servers erhalten. Falls Sie den Multiuser-Server nach dem Löschen neu installieren und ein vorhandenes Verzeichnis als "Repository" eintragen, stehen Ihnen diese Daten auch im neuen Server uneingeschränkt zur Verfügung.
"Port" definieren Im obigen Beispiel verwendet: `-p 5080`	Geben Sie den Port ein, den Sie für die Kommunikation mit dem Multiuser-Server verwenden möchten.
"Binding" definieren Im obigen Beispiel verwendet: `-b https`	Definieren Sie, ob Sie "https" (empfohlen) oder "http" als Transferprotokoll verwenden möchten. Mit "https" wird die Kommunikation verschlüsselt und ein Zertifikat angefordert. Die Kommunikation hat damit eine höheren Sicherheitsstufe als die unverschlüsselte "http"-Kommunikation.
"Certificate hash" = Fingerprint für das verwendete Zertifikat	Tragen Sie den Fingerprint für das von Ihnen erzeugte oder verwendete Zertifikat für eine sichere Serverkommunikation ein. Weitere Informationen zum Erzeugen eines Zertifikats finden Sie in der Hilfe zum verwendeten Windows-Betriebssystem.
"Run service" = Dienst starten Beispiel: `musrv start`	Damit starten Sie den Windows-Dienst für den Multiuser-Server.

Weitere Befehle zur Konfiguration des Multiuser-Servers finden Sie hier: Befehle zur Konfiguration des Multiuser-Servers (Seite 90)

Siehe auch

Befehle zur Konfiguration des Multiuser-Servers (Seite 90)

2.13.3.3 Befehle zur Konfiguration des Multiuser-Servers

Einführung

Befehle zur Konfiguration des Multiuser-Servers stehen Ihnen über die Multiuser Administrative Tools zur Verfügung.

Beachten Sie die grundsätzliche Vorgehensweise zur Konfiguration des Multiuser-Servers: Multiuser-Server mit den Administrative Tools konfigurieren (Seite 88)

Hinweis

Die Befehle sowie die zugehörigen Hilfen und Informationstexte innerhalb der Kommandozeile sind ausschließlich in Englisch verfügbar.

Befehle zur Konfiguration des Multiuser-Servers

Die nachfolgend aufgelisteten Befehle stehen zur Verfügung:

Befehl	Beschreibung	Mögliche Eingaben
"Install"	Installiert den Multiuser-Server und ermöglicht die Kommunikation mit dem Server. Sie finden ein Konfigurationsbeispiel unter: Multiuser-Server mit den Administrative Tools konfigurieren (Seite 88)	**Parameter:** • -r, --repository: Verzeichnis, in das die Serverdaten gespeichert werden sollen. • -p, --port: Port-Nr. für den Serverzugang • -b, --binding: Festlegen des Protokolls für die Kommunikation (http oder https) • -d, --debug: (Optional): Definiert, ob Runtime-Debug-Informationen vom Server angezeigt werden. Hinweis: Wenn Sie die Option "-d" verwenden, werden keine Debug-Logs angezeigt. • -h, --certhash: Fingerprint des Zertifikats, das für einen sicheren Datentransfer verwendet werden soll (wird nur für https benötigt). • -t, --timeout: (Optional): Definition des verwendeten Timeout-Profils ("fast", "middle" oder "slow"). Default: "fast". • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage • -c, --count: Maximale Anzahl (1 bis 10) der gespeicherten Projekt-Versionen. **Beispiel:** `musrv install -r <repository-path> -p <port> -b <binding> -h <certificate-hash>`
"Config"	Zeigt die aktuelle Konfiguration des Multiuser-Servers an und ermöglicht optionale dynamische Konfigurations-Parameter.	**Parameter:** • -t, --timeout: (Optional): Definition des verwendeten Timeout-Profils ("fast", "middle" oder "slow"). • -c, --count: Maximale Anzahl (1 bis 10) der gespeicherten Projekt-Versionen. • -d, --debug: (Optional): Definiert, ob Runtime-Debug-Informationen vom Server angezeigt werden. Hinweis: Wenn Sie die Option "-d" verwenden, werden keine Debug-Logs angezeigt. **Beispiel:** `musrv config -d`
"Start"	Startet den Multiuser-Server.	**Parameter:** • keine Parameter verfügbar **Beispiel:** `musrv start`

Befehl	Beschreibung	Mögliche Eingaben
"Stopp"	Stoppt den Multiuser-Server.	**Parameter:** • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `musrv stopp`
"Uninstall"	Deinstalliert den Multiuser-Service. Die Deinstallation kann auch mit den Windows-Mechanismen durchgeführt werden.	**Parameter:** • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `musrv uninstall`

Siehe auch

Multiuser-Server mit den Administrative Tools konfigurieren (Seite 88)

2.13.3.4 Multiuser-Server mit den Power Tools verwalten

Einführung

Um im Rahmen von Team Engineering Serverprojekte und lokale Sessions auf dem Multiuser-Server verwalten zu können, stehen Ihnen die nachfolgend beschriebenen Befehle zur Verfügung, die mit Hilfe der Power Tools über die Kommandozeile ausgeführt werden können.

Die Verwaltung des Multiuser-Servers über die Power Tools kann von einem beliebigen Rechner aus mit installiertem TIA Portal und installierten Power Tools erfolgen.

Ihre Eingaben werden in einem "Sitzungsgedächtnis" gemerkt und stehen beim nächsten Öffnen der Power Tools zur Verfügung.

Hinweis

Um den Multiuser-Server zu verwalten, benötigen Sie Nutzerrechte.

Voraussetzung

Der Multiuser-Server ist installiert und in der Windows-Firewall ist ein Port freigeschaltet, so dass der Multiuser-Server erreichbar ist.

Vorgehensweise

Um den Multiuser-Server zu verwalten, gehen Sie vor wie folgt.

1. Rufen Sie das Power Tool für die Multiuser-Server-Verwaltung mit Administrator-Rechten auf. Doppelklicken Sie dazu im Installationsverzeichnis des TIA Portals unter "bin" auf die Datei "start-mupt.bat".
2. Kopieren Sie den Befehl "mupt" in die Kommandozeile und drücken Sie die "Return-Taste". Die Hilfe zu den Multiuser Power Tools wird angezeigt.
3. Tragen Sie den gewünschten Befehl zur Serververwaltung in die Kommandozeile ein und drücken Sie die "Return-Taste".
   Die zur Verfügung stehenden Befehle finden Sie beschrieben unter: Befehle zum Verwalten des Multiuser-Servers (Seite 93)
4. Falls der Befehl nicht erfolgreich ausgeführt werden konnte, erhalten Sie eine Fehlermeldung.

Siehe auch

Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

2.13.3.5 Befehle zur Verwaltung des Multiuser-Servers

Einführung

Befehle zur Verwaltung des Multiuser-Servers stehen Ihnen über die Multiuser Power Tools zur Verfügung.

Hinweis

Die Befehle sowie die zugehörigen Hilfen und Informationstexte innerhalb der Kommandozeile sind ausschließlich in Englisch verfügbar.

"User names" und "Role names"

"User names":

Alle verwendeten "User names" müssen das nachfolgend genannte Format berücksichtigen:

- <domain-or-machine-name>\<account>, e.g ww004\adgrfl0 or MD17E4MC\adgrfl0.
- Falls der <domain-or-machine-name> nicht vergeben ist, wird Windows selbständig eine entsprechende Domäne für den Account vorschlagen.

"Role namens":

Folgende "Role names" mit den entsprechenden Zugriffsrechten sind vorhanden:

- Member (only read access)
- Contributor (read & write access)
- Manager (read and write access and role management).

Befehle zur Verwaltung des Multiuser-Servers

Die nachfolgend aufgelisteten Befehle stehen zur Verfügung:

Befehl	Beschreibung	Mögliche Eingaben
"Display project"	Listet alle auf dem aktuellen Multiuser-Server vorhandenen Projekte auf.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse **Beispiel:** `%InstallDirectory% > mupt project show -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "show-projects -?"
"Delete project"	Löscht spezifizierte Projekte auf dem Multiuser-Server.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Definition des zu löschenden Projektes • -f --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `%InstallDirectory% > mupt project remove -s http://192.168.0.100:8057/ -p MyProject` **Hilfeaufruf:** • "delete-project -?"
"Display sessions"	Listet alle zum aktuellen Serverprojekt gehörenden lokalen Sessions auf.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Auswahl des Projektes **Beispiel:** `%InstallDirectory% > mupt session show -s http://192.168.0.100:8057/ -p MyProject` **Hilfeaufruf:** • "show-sessions -?"
"Delete session"	Löscht die ausgewählte lokale Session auf dem Multiuser-Server	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Auswahl des Projektes • -i, --id: Session-Identifikationsnummer • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `%InstallDirectory% > mupt session remove -s http://192.168.0.100:8057/ -p MyProject -i 15` **Hilfeaufruf:** • "delete-session -?"

Befehl	Beschreibung	Mögliche Eingaben
"Remove project lock"	Entfernt die Sperre für das Multiuser-Projekt.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Auswahl des Projektes • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `%InstallDirectory% > mupt project unlock -s http://192.168.0.100:8057/ -p MyProject` **Hilfeaufruf:** • "remove-projectlock -?"
"Show check-in history"	Zeigt die Eincheck-Historie für den ausgewählten Server und die ausgewählte lokale Session an.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Auswahl des Projektes • -i, --id: Session-Identifikationsnummer • -l, --page: Ergebnisanzeige Seite für Seite • -d, --display: Ergebnisanzeige in unterschiedlicher Form: USAGE: -d [cascade/short/tab] **Beispiel:** `%InstallDirectory% > mupt checkin show -s http://192.168.0.100:8057/ -p MyProject` **Hilfeaufruf:** • "show-checkinhistory -?"
"Show server access rights"	Zeigt die definierten Benutzerrechte für den ausgewählten Server an.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse **Beispiel:** `%InstallDirectory% > mupt user show -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "show-serveraccessrights -?"
"Modify server access rights"	Ändert die Zugriffsrechte für den ausgewählten Server.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -r, --role: Auswahl der Rolle • -u, --user: Anwendername, nach dem gefiltert werden soll. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `%InstallDirectory% > mupt user add -u user1 -r Manager -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "modify-serveraccessrights -?"

Befehl	Beschreibung	Mögliche Eingaben
"Remove server access rights"	Entfernt die Zugriffsrechte für den ausgewählten Server.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -u, --user: Anwendername, nach dem gefiltert werden soll. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel:** `%InstallDirectory% > mupt user remove -u user1 -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "modify-serveraccessrights -?"
"Show project access rights"	Zeigt die definierten Benutzerrechte für das ausgewählte Projekt.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname, zu dem die Benutzerrechte aufgelistet werden sollen. **Beispiel**: `%InstallDirectory% > mupt user show -s http://192.169.0.100:8057/ -p MYProject` **Hilfeaufruf:** • "show-projectaccessrights -?"
"Modify project access rights"	Ändert die Benutzerrechte für das ausgewählte Projekt.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname, zu dem die Benutzerrechte aufgelistet werden sollen. • -r, --role: Auswahl der Rolle • -u, --user: Anwendername, nach dem gefiltert werden soll. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel**: `%InstallDirectory% > mupt user add -u user2, user3 -r Member -p MyProject -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "modify-projectaccessrights -?"

Befehl	Beschreibung	Mögliche Eingaben
"Remove project access rights"	Entfernt die Zugriffsrechte des ausgewählten Benutzers für das ausgewählte Projekt.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -u, --user: Anwendername, nach dem gefiltert werden soll. • -p, --project: Projektname, zu dem die Benutzerrechte aufgelistet werden sollen. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel**: `%InstallDirectory% > mupt user remove -u user3 -p MyProject -s http://192.168.0.100:8057/` **Hilfeaufruf:** • "remove-serveraccessrights -?"
"Export project to single user"	Exportiert das Multiuser-Projekt als Single-User-Projekt zu dem angegebenen Pfad.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname, zu dem die Benutzerrechte aufgelistet werden sollen. • -t, --path: Auswahl des Ablagepfades **Beispiel**: `%InstallDirectory% > mupt project export -s http://192.168.0.100:8057/ -p MyProject -t D:\MyDirectory\` **Hilfeaufruf:** • "export-projectassingleuser -?"

Befehl	Beschreibung	Mögliche Eingaben
"Export of server project"	Exportiert ein vorhandenes Multiuser-Serverprojekt zu dem Zielordner im angegebenen Pfad.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname des Projekts, das exportiert werden soll. Wird kein Projektname angegeben, so werden alle im angegenbenen Server-Repository vorhandenen Projekte exportiert. • -t, --path: Auswahl des Zielordners, zu dem das Projekt exportiert werden soll. Der Ordner muss leer sein. • -m,--metadata: (Optional) Das Multiuser-Serverprojekt wird vollständig mit der gesamten Historie und allen vorhandenen Revisionen exportiert. Die Voreinstellung ist: "metadata" = false. Wenn metadata nicht spezifiziert ist, wird die ausgewählte Revision (-v, --revision) exportiert. • -v, --revision: (Optional) Definiert die Revisionsnummer, die exportiert werden soll. Wenn "metadata" aktiv ist, wird dieser Wert ignoriert. Per Voreinstellung wird immer die letzte Revision exportiert. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel**: `%InstallDirectory% > mupt project export --server net.tcp://localhost:8735 --project (ProjectName) --path (targetPath) --metadata (true/false) --revision(revisionNumber) --force` **Hilfeaufruf:** • "export project -?"
"Import of server project"	Importiert ein vorhandenes Multiuser-Serverprojekt vom angegebenen Pfad zum angegenen Multiuser-Server-Repository. **Voraussetzung:** Nur Benutzer mit der Rolle "Contributor" oder "Manager" dürfen diese Aktion durchführen. **Regeln:** • Es kann immer nur ein Serverprojekt gleichzeitig importiert werden. • Der Name des zu importierenden Serverprojekts wird auch dem Zielordner zugeordnet.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -t, --path: Auswahl des Ablagepfades, von dem das Projekt importiert werden soll. **Beispiel**: `%InstallDirectory% > mupt project import --server net.tcp://localhost:8735 --path (sourcePath)` **Hilfeaufruf:** • "import project -?"

Befehl	Beschreibung	Mögliche Eingaben
"Export of server project sessions"	Exportiert die lokalen Sessions eines Multiuser-Serverprojekts zum angegebenen Pfad.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname des Serverprojekts, zu dem die zugehörigen lokalen Sessions exportiert werden sollen. Wird kein Projektname angegeben, so werden alle im angegenbenen Server-Repository vorhandenen Projekte exportiert. • -t, --path: Auswahl des Zielordners, zu dem die lokalen Sessions des Serverprojekts exportiert werden sollen. Der Ordner muss leer sein. • -i--id: (Optional) ID der lokalen Session, die exportiert werden soll. **Beispiel:** `%InstallDirectory% > mupt session export --server net.tcp://localhost:8735 --path (targetPath)--id (idNum)` **Hilfeaufruf:** • "session export -?"
"Import of server project sessions"	Importiert die exportierten lokalen Sessions zu einem ausgewählten Multiuser-Serverprojekt. **Voraussetzung:** Nur Benutzer mit der Rolle "Contributor" oder "Manager" dürfen diese Aktion durchführen. **Regel**: Es werden nur die lokalen Sessions importiert, die dem zugehörigen Serverprojekt bekannt sind.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname des Serverprojekts, zu dem die zugehörigen lokalen Sessions importiert werden sollen. • -t, --path: Auswahl des Ablagepfades, von dem die lokalen Sessions importiert werden sollen. • -a, --alias: (Optional) Alias-Name des Zielservers, falls verfügbar. **Beispiel:** `%InstallDirectory% > mupt session import --server net.tcp://localhost:8735 --path (sourcePath)--alias (serverAlias)`

Befehl, um den Server auf eine ältere Revision zurückzusetzen

Der nachfolgend aufgelistete Befehl kann nur ausführt werden, wenn der Server nicht gesperrt ist.

"Project rollback"	Setzt den Server auf eine vorhandene ältere Revision zurück.	**Parameter:** • -s, --server: Multiuser-Server URI-Adresse • -p, --project: Projektname, zu dem die Benutzerrechte aufgelistet werden sollen. • -v, --revision: Revisionsnummer des Projektes, auf das zurückgesetzt werden soll. • -f, --force: (Optional): Aktiviert den Befehl ohne weitere Rückfrage **Beispiel**: `%InstallDirectory% > mupt project rollback -s http://192.168.0.100:8057/ -p MyProject -v 132`

2.13.4 Zertifikat für Multiuser-Server bestätigen

Einführung

Um eine sichere Verbindung mit einem installierten Multiuser-Server herstellen zu können, müssen Sie beim Arbeiten mit Multiuser Engineering zunächst das SSL-Server-Zertifikat festlegen. Sobald Sie nach der Server-Installation zum ersten Mal in einer lokalen Session arbeiten, werden Sie aufgefordert, das angezeigte Zertifikat zu bestätigen.

Wenn ein bestehendes Zertifikat abgelaufen ist, werden Sie erneut zur Bestätigung aufgefordert.

Für die Verwaltung der SSL-Server-Zertifikate werden die von Windows bereitgestellten Mechanismen verwendet.

Falls Sie das SSL-Server-Zertifikat nicht bestätigen, kann keine Verbindung zum Multiuser-Server aufgebaut werden.

Voraussetzung

Sie haben das TIA Portal und die benötigte Software für den Multiuser-Server installiert.

Der Multiuser-Server muss mithilfe der Multiuser-Tools konfiguriert und gestartet sein.

Sie arbeiten zum ersten Mal in der lokalen Session.

Zertifikat für Multiuser-Server bestätigen

Um das Zertifikat für den Multiuser-Server zu bestätigen, gehen Sie vor wie folgt:

1. Beim erstmaligen Arbeiten mit einem verbundenen Multiuser-Server, z. B. beim Öffnen einer lokalen Session, wird durch das Betriebssystem automatisch die Bestätigung des Server-Zertifikates für den Multiuser-Server angefordert.
2. Überprüfen Sie, ob das angezeigte Zertifikat vertrauenswürdig ist. Falls ja, bestätigen Sie im angezeigten Dialog das Zertifikat für den "TIA_Multiuser_Server", indem Sie auf "OK" klicken.

Ergebnis

Der Dialog wird geschlossen. Das Zertifikat für den Multiuser-Server wird als vertrauenswürdig angesehen und permanent gespeichert.

2.13.5 Zugangsdaten für Multiuser-Server eingeben

Einführung

Wenn Sie mit Multiuser Engineering arbeiten und einen Multiuser-Server installiert haben, müssen Sie sich beim Multiuser-Server anmelden, falls eine Authentifizierung benötigt wird.

In diesem Fall werden Sie vom Multiuser-Server aufgefordert, sich anzumelden. Ein Dialog wird angezeigt, in dem Sie die Zugangsdaten eingeben können.

Für die Anmeldung haben Sie zwei Möglichkeiten:

- Anmeldung mit Ihrem aktuellen Windows-Benutzernamen und Passwort
- Anmeldung mit den Zugangsdaten eines anderen Benutzers

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Der Multiuser-Server muss mit Hilfe der Power Tools konfiguriert und gestartet sein.

Zugangsdaten für Multiuser-Server eingeben

Um die Benutzerdaten einzugeben, gehen Sie vor wie folgt:

1. Wählen Sie im angezeigten Dialog die gewünschte Option aus, indem Sie diese anklicken.
   - Beim Anmelden mit Ihrem aktuellen Login wird der verwendete Benutzername bereits vorbelegt und ist nicht editierbar.
   - Alternativ können Sie sich mit den Zugangsdaten eines anderen Benutzers anmelden.
2. Wählen Sie dazu die zweite Option aus und tragen Sie für "Benutzername" und "Passwort" die gewünschten Daten ein.
3. Klicken Sie auf "OK".

Ergebnis

Bei erfolgreicher Authentifizierung wird der Dialog geschlossen und die Verbindung zum Multiuser-Server wird aufgebaut.

2.13.6 Serverdaten auf anderen Multiuser-Server übertragen

Einführung

Wenn Sie mit Multiuser Engineering arbeiten und einen Multiuser-Server installiert haben, können Sie bestehende Serverdaten auf einen anderen Multiuser-Server ohne Datenverlust übertragen.

Die Daten können auf dem gleichen Computer oder auf einen anderen Computer übertragen werden.

Voraussetzung

Sie haben das TIA Portal und mindestens einen Multiuser-Server installiert.

Der Multiuser-Server muss konfiguriert und gestartet sein.

Multiuser-Serverdaten auf anderen Multiuser-Server übertragen

Um die Serverdaten eines Multiuser-Servers auf einen anderen Multiuser-Server zu übertragen, gehen Sie vor wie folgt:

1. Stoppen Sie den aktiven Multiuser-Server.
   - Rufen Sie das grafische ConfigurationTool für die Multiuser-Server-Konfiguration mit Administrator-Rechten auf.
   - Öffnen Sie das Startmenü mit dem Befehl "Start" > "Alle Programme" > "Siemens Automation".
   - Klicken Sie auf den Eintrag "Multiuser Server V<x.y> Configuration". Daraufhin öffnet sich die Benutzeroberfläche des Configuration Tools.
   - Klicken Sie im Bereich "Serverstatus" auf die Schaltfläche "Stop Service".
2. Deinstallieren Sie den "alten" Multiuser-Server.
   Siehe hierzu: Multiuser-Server installieren und deinstallieren (Seite 49)
3. Kopieren Sie das gesamte Ablageverzeichnis (Repository) mit den bestehenden Serverdaten an den neuen Ablageort, z. B. auf einen anderen Computer.
4. Installieren Sie den Multiuser-Server auf dem neuen Computer.
   Siehe hierzu: Multiuser-Server installieren und deinstallieren (Seite 49)

5. Konfigurieren Sie das neue Ablageverzeichnis für den neuen Multiuser-Server.
   Siehe hierzu: Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)
6. Starten Sie den neuen Multiuser-Server. Klicken Sie dazu im Configuration Tool im Bereich "Serverstatus" auf die Schaltfläche "Start Service".

Ergebnis

Die Serverdaten wurden ohne Datenverlust auf den neuen Multiuser-Server übertragen.

Siehe auch

Multiuser-Server mit grafischem Configuration Tool konfigurieren (Seite 60)

Multiuser-Server installieren und deinstallieren (Seite 49)

2.13.7 Error-Log für Multiuser-Server auslesen

Einführung

Um die Multiuser-Serverfunktionalität zu überprüfen, gibt es die Möglichkeit, die Error-Logs auszulesen. Die Error-Logs geben Ihnen zusätzliche Hinweise zu den in der Oberfläche angezeigten Fehlermeldungen, wenn Serverfehler auftreten.

Vom Server wird automatisiert für jeden Tag eine separate Log-Datei angelegt. Die Log-Dateien werden nicht automatisch gelöscht.

Error-Logs für einen Multiuser-Server auslesen

Um das Error-Log auszulesen, gehen Sie vor wie folgt:

1. Öffnen Sie in Ihrer Ablage das Verzeichnis für den gewünschten Multiuser-Server.
2. Wählen Sie unter dem Pfad <Repository>\logs die gewünschte Log-Datei aus.
3. Öffnen Sie die gewünschte Log-Datei, um den Inhalt zu überprüfen

Error-Logs für ein Multiuser-Projekt auslesen

Um das Error-Log auszulesen, gehen Sie vor wie folgt:

1. Öffnen Sie in Ihrer Ablage das Verzeichnis für den gewünschten Multiuser-Server.
2. Wählen Sie unter dem Pfad <Repository>\proj\<Projektname>\logs die gewünschte Log-Datei aus.
3. Öffnen Sie die gewünschte Log-Datei, um den Inhalt zu überprüfen.

Hinweis

Auslesen des Error-Logs

Beachten Sie, dass kein Error-Log ausgelesen werden kann, wenn Sie im Debug-Modus arbeiten.

Überprüfen bzw. ändern Sie die Server-Konfiguration in der Kommandozeile über die Administrative Tools mit dem Befehl "musrv config".

Achten Sie darauf, dass die Option "-d" nicht aktiviert ist.

Windows-Ereignisanzeige auslesen

Die Ereignisanzeige ist ein Tool, mit dem detaillierte Informationen zu wichtigen Ereignissen auf dem Computer angezeigt werden. Die Ereignisanzeige kann bei der Behandlung von Problemen und Fehlern hilfreich sein.

Folgende sicherheitsrelevante Ereignisse bezüglich des Multiuser-Servers erhalten ebenfalls einen Eintrag in der Windows-Ereignisanzeige:

- Falls ein Anwender den Multiuser-Server starten möchte, der dem Multiuser-Server nicht "bekannt" ist, also beim Server nicht als Anwender angemeldet ist.
- Falls ein Anwender sich mit einem falschen Passwort beim Multiuser-Server anmelden möchte.

Um die Ereignisanzeige auszulesen, gehen Sie vor wie folgt:

1. Klicken Sie auf "Start" > "Systemsteuerung" > "System und Sicherheit" > "Verwaltung". Alternativ klicken Sie auf "Start" und tragen Sie im Suchfeld "Ereignisanzeige" ein.
2. Öffnen Sie die Ereignisanzeige durch Doppelklick.
3. Klicken Sie unter "Windows-Protokolle" auf "Sicherheit", um die protokollierten sicherheitsrelevanten Ereignisse zum Multiuser-Server anzuzeigen.

Ergebnis

Die mitprotokollierten sicherheitsrelevanten Ereignisse zum Multiuser-Server werden in einer Liste aufgeführt.

2.13.8 Error-Log für Multiuser-Server als Zip-Datei verpacken

Einführung

Um die Multiuser-Serverfunktionalität zu überprüfen, gibt es die Möglichkeit, die Error-Logs auszulesen. Die Error-Logs geben Ihnen zusätzliche Hinweise zu den in der Oberfläche angezeigten Fehlermeldungen, wenn Serverfehler auftreten.

Sie haben die Möglichkeit, alle vorhandenen Error-Logs in einer Zip-Datei zu verpacken, um diese bei Bedarf z. B. an den Customer Support weiterzuleiten.

Dazu müssen Sie sich als Administrator direkt beim beim Server einloggen. Es ist nicht möglich, diese Aktion über das TIA Portal oder über die Administrative und Power Tools auszuführen.

Error-Logs als Zip-Datei verpacken

Um eine Zip-Datei für die Error-Logs zu erstellen, gehen Sie vor wie folgt:

1. Melden Sie sich beim Multiuser-Server als Administrator an.
2. Führen Sie den Befehl "musrv ziplogs" aus.
3. Alternativ können Sie den Befehl mit folgenden Optionen ausführen:
   – -n, --name: Eingabe des Ablagepfads für die Zip-Datei
   – -f, --force: Überschreibt evtl. schon vorhandene Zip-Dateien ohne Rückfrage.

Ergebnis

Die Zip-Datei mit allen vorhandenen Error-Logs zum Multiuser-Server wird erstellt.

2.13.9 Hinweise zum Fehlerverhalten des Multiuser-Servers

Einführung

Nachfolgend wird das Verhalten des Multiuser-Servers nach dem Auftreten eines Fehlers beschrieben.

Sollte ein Fehler auftreten, erhalten Sie eine entsprechende Fehlermeldung.

Verhaltensweise des Servers nach dem Auftreten eines Fehlers

Der Multiuser-Server zeigt folgende Verhaltensweisen nach dem Auftreten eines Fehlers:

Projektsperre:

- Die Information, welches Projekt gesperrt ist, geht möglicherweise nach einem Serverfehler verloren.
- Diese Information steht nach einem Neustart des Servers nicht mehr zur Verfügung.

Fehler, die beim "Laden von Gerät" oder beim "Laden in Gerät" auftreten:

- Das Aktualisieren der lokalen Session kann nicht durchgeführt werden. Die Aktion wird mit einer Fehlermeldung abgebrochen.
- Das Einchecken kann nicht durchgeführt werden. Die Aktion wird mit einer Fehlermeldung abgebrochen.
- Eventuell bereits geladenen Daten werden vom Server wieder entfernt, der Server hat den gleichen Status wie vor Beginn der fehlerhaften Aktion.

Hinweis

Manueller Neustart des Servers

Nach dem Auftreten eines Fehlers muss der Server manuell neu gestartet werden.

Ein automatischer Start des Servers erfolgt erst nach dem nächsten Neustart des Rechners, auf dem der Server läuft.

2.13.10 Unterstützte Netzwerk-Konfigurationen für den Multiuser-Server

2.13.10.1 Voraussetzungen für eine erfolgreiche Client-Server-Kommunikation

Einführung

Nachfolgend finden Sie einen Überblick über die Voraussetzungen für eine erfolgreiche Client-Server-Kommunikation im Kontext von Multiuser Engineering zu:

- Netzwerk-Kommunikation
- Zertifikate-Verwaltung
- Authentifizierung

Voraussetzungen für die Netzwerk-Kommunikation

Das verwendete Netzwerk muss eine HTTPS-Kommunikation zwischen Client und Server ermöglichen.

Sie haben zwei Möglichkeiten, dies umzusetzen:

- Direkte TCP-Kommunikation zwischen Client und Multiuser-Server über den freigeschalteten Server-Port.
  - Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen.
  - Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server.
  - Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing).
  - Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren.
- Indirekte Kommunikation über einen HTTPS-Proxy-Server.
  - Ein HTTPS-Proxy-Server ist verfügbar.
  - Der HTTPS-Proxy-Server ist auf der Client-Seite im Windows-System konfiguriert und für die Kommunikation freigeschaltet.

Voraussetzungen für die Zertifikate-Verwaltung

Um eine sichere Kommunikation aufzubauen, sollten Sie "HTTPS" verwenden. Mit "HTTPS" wird die Kommunikation verschlüsselt und ein Zertifikat angefordert. Die Kommunikation hat damit eine höhere Sicherheitsstufe als die unverschlüsselte "HTTP"-Kommunikation.

Um eine HTTPS-Kommunikation aufzubauen, muss der Multiuser-Server so konfiguriert werden, dass er ein gültiges HTTPS-Zertifikat für die Datenverschlüsselung verwendet.

- Das im Windows-Zertifikatsspeicher hinterlegte Zertifikat wird auf dem Multiuser-Server verwendet.
- Das Zertifikat muss aktuell gültig sein und für die Verwendung mit "http.sys" registriert sein.
- Der Fingerprint des Zertifikats muss an den Client übergeben werden.

Voraussetzungen für die Authentifizierung

Für die Authentifizierung der Kommunikation mit dem Multiuser-Server werden die Windows-Mechanismen verwendet.

Multiuser Engineering unterstützt sowohl die Authentifizierung über Domänen-Accounts als auch die Authentifizierung über lokale Windows-Accounts.

- Authentifizierung mit Domänen-Account
  - Sowohl der Client als auch der Multiuser-Server sind Mitglieder innerhalb derselben Windows-Domäne.
  - Ein gültiger Anwender-Account für diese Domäne wird verwendet.
  - Sowohl der Client als auch der Multiuser-Server können eine Netzwerk-Kommunikation zum Domänen-Server aufbauen.
  - Die Anforderungen für das Verwenden einer sicheren https-Kommunikation müssen erfüllt sein.
- Authentifizierung mit lokalem Account
  - Für die Authentifizierung muss ein lokaler Account auf dem Multiuser-Server verwendet werden.
  - Die Anforderungen für das Verwenden einer sicheren HTTPS-Kommunikation müssen erfüllt sein.

Hinweis

NTLM-Authentifizierung auf dem Multiuser-Server

Zur Authentifizierung auf dem Multiuser-Server wird NTLM verwendet. Wir empfehlen aus Sicherheitsgründen, das standardmäßig seit Windows 2000 aktivierte NTLM V2 zu verwenden und nicht NTLM V1 zu aktivieren.

NTLM V2 verfügt über ein sichereres Anmeldeverfahren als NTLM V1.

Hinweis

In den nachfolgenden Kapiteln finden Sie weitere Informationen zu unterstützten Netzwerk-Konfigurationen für den Multiuser-Server.

Siehe auch

Netzwerk mit dediziertem Multiuser-Server (Seite 108)

Netzwerk mit temporärem Multiuser-Server (Seite 109)

Netzwerk in einer Windows-Domäne (Seite 112)

Netzwerk in einer Windows-Domäne mit Internet-Zugriff über VPN (Seite 113)

Netzwerk in einer Windows-Domäne mit Internet-Zugriff über Proxy (Seite 116)

2.13.10.2 Netzwerk mit dediziertem Multiuser-Server

Einführung

Nachfolgend finden Sie Informationen zu einer Server-Konfiguration mit einem dedizierten Multiuser-Server in einem Büro-Netzwerk.

Die Grafik zeigt den möglichen Aufbau einer solchen Konfiguration:

Büro-Netzwerk mit dediziertem Multiuser-Server

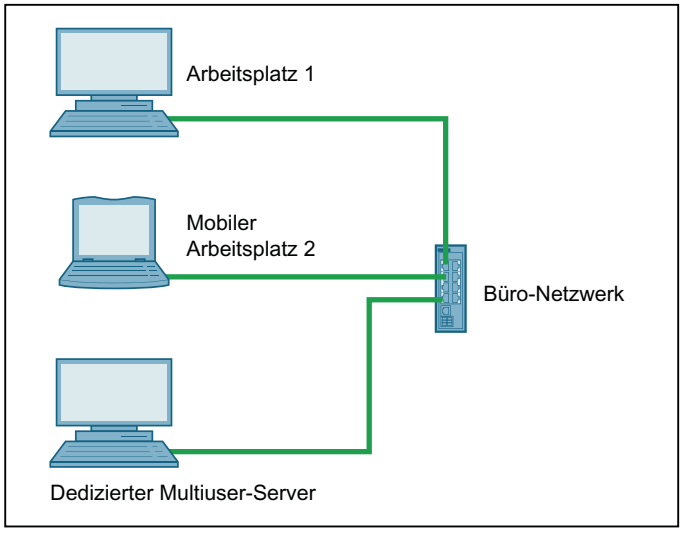

Beschreibung der Netzwerkkonfiguration mit einem dedizierten Multiuser-Server

Die Netzwerkkonfiguration sollte folgende Voraussetzungen erfüllen:

Konfiguration	Beschreibung
Netzwerk-Umgebung	Aufbau: • Verschiedene Windows-Arbeitsplätze, auf denen das TIA Portal installiert ist. • Ein dedizierter Windows-Server, der als Multiuser-Server installiert ist. • Die Arbeitsplätze und der Server befinden sich im selben Netzwerk.
Netzwerk	Voraussetzungen: • Es besteht eine direkte TCP-Verbindung zwischen den Arbeitsplätzen und dem Multiuser-Server über den ausgewählten Port. • Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen. • Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server. • Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing). • Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren. • Entweder ist kein Proxy-Server eingerichtet oder es existiert eine lokale Umgehung. Dazu sollte in den Windows-Einstellungen für ein lokales Netzwerk die Option "Proxyserver für lokale Adressen umgehen" aktiviert sein.
Authentifizierung	User Accounts und Windows-Arbeitsgruppen: • Diese müssen lokal auf dem Multiuser-Server angelegt werden. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Name und Passwort für die Authentifizierung werden auf allen Arbeitsplätzen abgefragt. • Für die Authentifizierung können nur lokale Accounts des Server-Systems genutzt werden. Hinweis: Zur Abfrage von Namen und Passwort kann die Windows-Authentifizierung verwendet werden. Dazu müssen Sie lokale Accounts anlegen, die identisch sind zu den Accounts, die auf dem Multiuser-Server vorhanden sind. Diese Vorgehensweise ermöglicht es, dass der Server einen lokalen Account mit identischen Login-Daten zu einem bestehenden Server-Account automatisch authentifiziert. Einschränkung: Anwender-Accounts und Gruppen können nicht zentral verwaltet werden, dazu werden lokale Anwender und Gruppen benötigt. Diese Konfiguration ist deshalb nicht geeignet für Enterprise-Netzwerke.

2.13.10.3 Netzwerk mit temporärem Multiuser-Server

Einführung

Nachfolgend finden Sie Informationen zu einer Server-Konfiguration mit einem temporären Multiuser-Server in einem Büro-Netzwerk.

Die Grafik zeigt den möglichen Aufbau einer solchen Konfiguration:

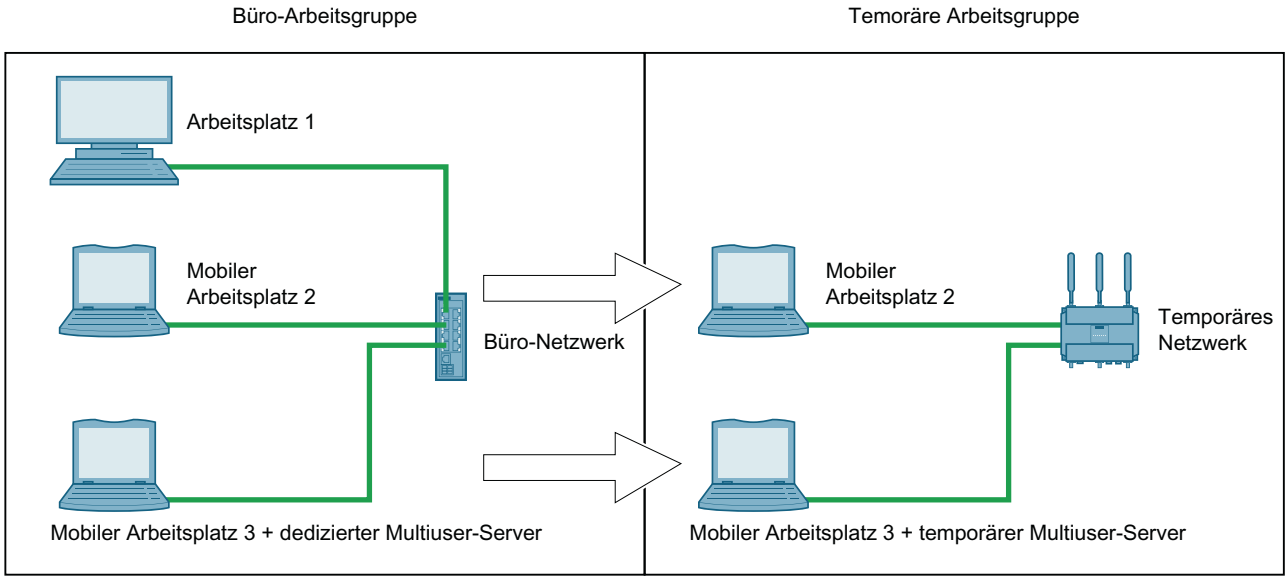

Beschreibung der Netzwerkkonfiguration mit einem temporären Multiuser-Server

Die Netzwerkkonfiguration sollte folgende Voraussetzungen erfüllen.

Konfiguration	Beschreibung
Netzwerk-Umgebung	Aufbau: • Verschiedene Windows-Arbeitsplätze, auf denen das TIA Portal installiert ist. • Ein temporärer Windows-Server, der als Multiuser-Server auf einem der Arbeitsplätze installiert ist. Möglicherweise wird die Büroumgebung verlassen, z. B. um eine Inbetriebnahme in einer Fabrik vorzunehmen. • Alle Arbeitsplätze und der temporäre Multiuser-Server befinden sich im selben Netzwerk. • Mit dem temporären Multiuser-Server kann die Büroumgebung verlassen und trotzdem weitergearbeitet werden.
Netzwerk	Voraussetzungen: • Es besteht eine direkte TCP-Verbindung zwischen den Arbeitsplätzen und dem Multiuser-Server über den ausgewählten Port. • Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen. • Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server. • Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing). • Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren. • Entweder ist kein Proxy-Server eingerichtet oder es existiert eine lokale Umgehung. Dazu sollte in den Windows-Einstellungen für ein lokales Netzwerk die Option "Proxyserver für lokale Adressen umgehen" aktiviert sein.
Authentifizierung	User Accounts und Windows-Arbeitsgruppen: • Diese müssen lokal auf dem Multiuser-Server angelegt werden. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Name und Passwort für die Authentifizierung werden auf allen Arbeitsplätzen abgefragt. • Für die Authentifizierung können nur lokale Accounts des Server-Systems genutzt werden. Hinweis: Zur Abfrage von Namen und Passwort kann die Windows-Authentifizierung verwendet werden. Dazu müssen Sie lokale Accounts anlegen, die identisch sind zu den Accounts, die auf dem Multiuser-Server vorhanden sind. Diese Vorgehensweise ermöglicht es, dass der Server einen lokalen Account mit identischen Login-Daten zu einem bestehenden Server-Account automatisch authentifiziert. Einschränkung: Anwender-Accounts und Gruppen können nicht zentral verwaltet werden, dazu werden lokale Gruppen benötigt. Diese Konfiguration ist deshalb nicht geeignet für Enterprise-Netzwerke.

2.13.10.4 Netzwerk in einer Windows-Domäne

Einführung

Nachfolgend finden Sie Informationen zu einer Server-Konfiguration mit einem dedizierten Multiuser-Server in einer Windows-Domäne.

Die Grafik zeigt den möglichen Aufbau einer solchen Konfiguration:

Netzwerk innerhalb einer Windows-Domäne

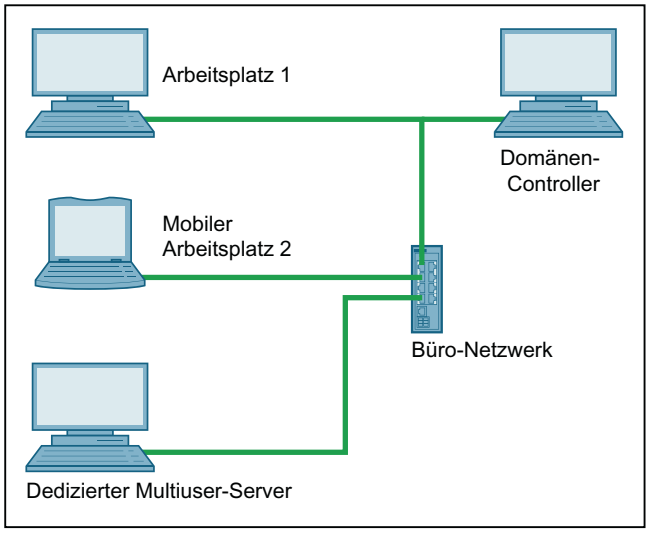

Beschreibung der Netzwerkkonfiguration in einer Windows-Domäne

Die Netzwerkkonfiguration sollte folgende Voraussetzungen erfüllen.

Konfiguration	Beschreibung
Netzwerk-Umgebung	Aufbau: • Alle Windows-Arbeitsplätze, auf denen das TIA Portal installiert ist, sind Mitglied in derselben Domäne. • Ein dedizierter Windows-Server als Multiuser-Server befindet sich ebenfalls in derselben Domäne. • Alle Arbeitsplätze und der Multiuser-Server nutzen dasselbe LAN (Local Area Network). • Alle Arbeitsplätze und der Multiuser-Server können sich mit dem Domänen-Controller verbinden.
Netzwerk	Voraussetzungen: • Es besteht eine direkte TCP-Verbindung zwischen den Arbeitsplätzen und dem Multiuser-Server über den ausgewählten Port. • Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen. • Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server. • Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing). • Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren. • Entweder ist kein Proxy-Server eingerichtet oder es existiert eine lokale Umgehung. Dazu sollte in den Windows-Einstellungen für ein lokales Netzwerk die Option "Proxyserver für lokale Adressen umgehen" aktiviert sein.
Authentifizierung	User Accounts und Windows-Arbeitsgruppen: • Diese müssen auf dem Domänen-Controller angelegt sein. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Anwender an Arbeitsplätzen, auf denen das TIA Portal läuft, werden automatisch authentifiziert mit Ihren Domänen-Login-Daten. Hinweis: Auf dem Multiuser-Server können Sie sich auch mit anderen gültigen Login-Daten anmelden. Einschränkung: Diese Konfiguration ist nicht geeignet für temporäre Server-Konfigurationen, da diese ebenfalls Zugriff auf den Domänen-Controller benötigen.

2.13.10.5 Netzwerk in einer Windows-Domäne mit Internet-Zugriff über VPN

Einführung

Nachfolgend finden Sie Informationen zu einer Server-Konfiguration mit einem dedizierten Multiuser-Server in einer Windows-Domäne mit VPN (Virtual Private Network).

Die Grafik zeigt den möglichen Aufbau einer solchen Konfiguration:

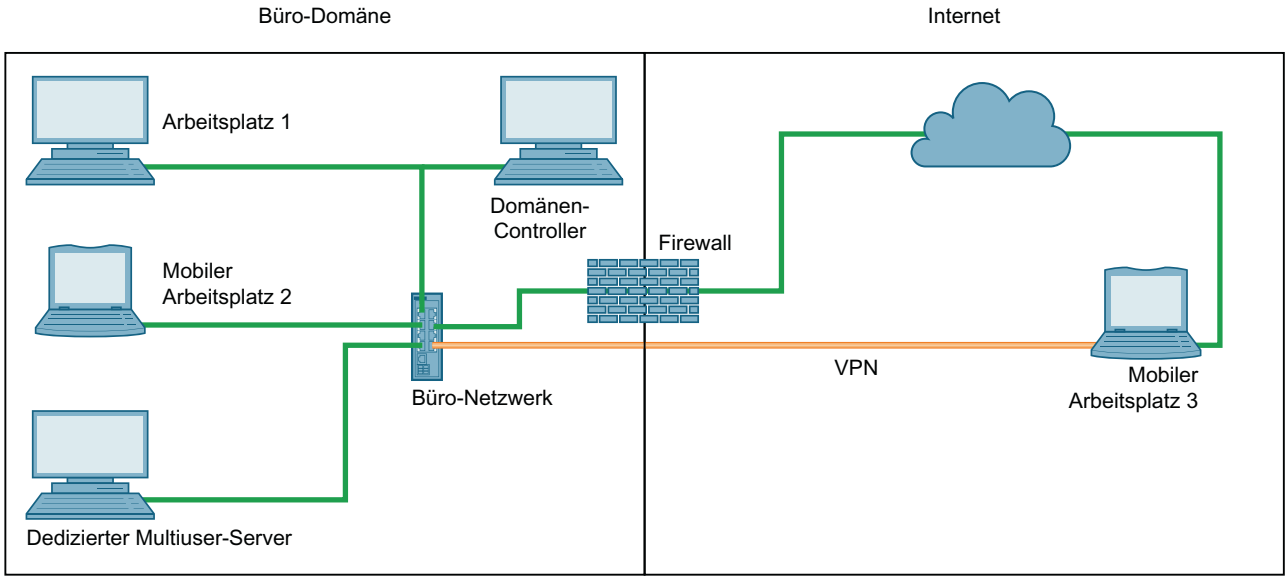
Büro-Domäne
Internet
Arbeitsplatz 1
Domänen-
Controller
Firewall
Mobiler
Arbeitsplatz 2
Büro-Netzwerk
VPN
Mobiler
Arbeitsplatz 3
Dedizierter Multiuser-Server

Beschreibung der Netzwerkkonfiguration in einer Windows-Domäne mit VPN

Die Netzwerkkonfiguration sollte folgende Voraussetzungen erfüllen.

Konfiguration	Beschreibung
Netzwerk-Umgebung	Aufbau: • Alle Windows-Arbeitsplätze, auf denen das TIA Portal installiert ist, sind Mitglied in derselben Domäne. • Ein dedizierter Windows-Server als Multiuser-Server befindet sich ebenfalls in derselben Domäne. • Alle Arbeitsplätze und der Multiuser-Server nutzen dasselbe LAN (Local Area Network). • Sowohl die Arbeitsplätze als auch der Multiuser-Server haben Zugriff auf den Domänen-Controller. • Zusätzlich können sich Arbeitsplätze über VPN zum LAN verbinden.
Netzwerk	Voraussetzungen: • Es besteht eine direkte TCP-Verbindung über den ausgewählten Port zwischen den Arbeitsplätzen und dem Multiuser-Server. • Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen. • Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server. • Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing). • Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren. • Entweder ist kein Proxy-Server eingerichtet oder es existiert eine lokale Umgehung. Dazu sollte in den Windows-Einstellungen für ein lokales Netzwerk die Option "Proxyserver für lokale Adressen umgehen" aktiviert sein. • Aus Sicherheitsgründen empfehlen wir die Benutzung von VPN-Lösungen, wie z. B. "Routing and Remote Access Service". • Aus Sicherheitsgründen wird das Einrichten einer Firewall dringend empfohlen. Es ist nicht notwendig, für den Multiuser-Server einen Port in der Firewall zu öffnen, wenn Sie VPN benutzen.

Konfiguration	Beschreibung
Authentifizierung	Die Authentifizierung erfolgt abhänging von der gewählten Konfiguration. **User Accounts und Windows-Arbeitsgruppen bei Konfiguration als Büro-Arbeitsgruppe:** • Diese müssen lokal auf dem Multiuser-Server angelegt werden. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Name und Passwort für die Authentifizierung werden auf allen Arbeitsplätzen abgefragt. • Für die Authentifizierung können nur lokale Accounts des Server-Systems genutzt werden. **Hinweis:** Zur Abfrage von Namen und Passwort kann die Windows-Authentifizierung verwendet werden. Dazu müssen Sie lokale Accounts anlegen, die identisch sind zu den Accounts, die auf dem Multiuser-Server vorhanden sind. Diese Vorgehensweise ermöglicht es, dass der Server einen lokalen Account mit identischen Login-Daten zu einem bestehenden Server-Account automatisch authentifiziert. **Einschränkung:** Anwender-Accounts und Gruppen können nicht zentral verwaltet werden, dazu werden lokale Anwender und Gruppen benötigt. Diese Konfiguration ist deshalb nicht geeignet für Enterprise-Netzwerke. **User Accounts und Windows-Arbeitsgruppen bei Konfiguration als Windows-Domäne:** • Diese müssen auf dem Domänen-Controller angelegt sein. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Anwender an Arbeitsplätzen, auf denen das TIA Portal läuft, werden automatisch authentifiziert mit Ihren Domänen-Login-Daten. **Hinweis:** Auf dem Multiuser-Server können Sie sich auch mit anderen gültigen Login-Daten anmelden. **Einschränkung:** Diese Konfiguration ist nicht geeignet für temporäre Server-Konfigurationen, da diese ebenfalls Zugriff auf den Domänen-Controller benötigen. Die Kommunikation mit einem Remote-Rechner über VPN zeigt für alle Client-Server-Anwendungen typische Performance-Einschränkungen. Der Multiuser-Server ist davon ebenfalls betroffen.

2.13.10.6 Netzwerk in einer Windows-Domäne mit Internet-Zugriff über Proxy

Einführung

Nachfolgend finden Sie Informationen zu einer Server-Konfiguration mit einem dedizierten Multiuser-Server in einer Windows-Domäne mit Intranet/Internet-Zugriff über einen Proxy.

Die Grafik zeigt den möglichen Aufbau einer solchen Konfiguration:

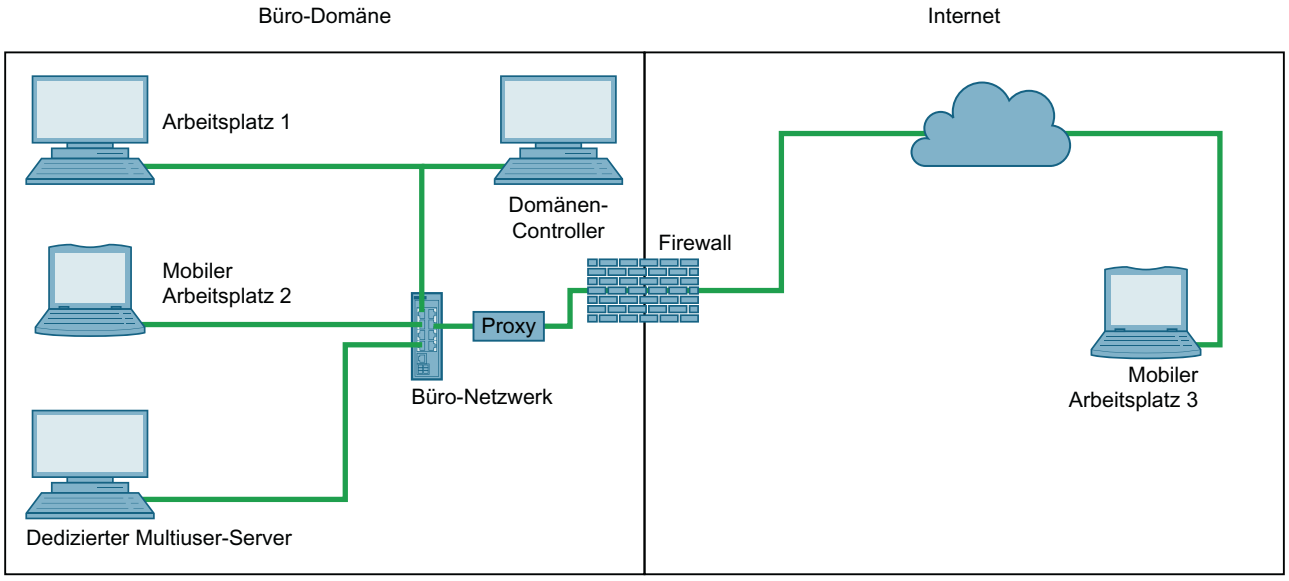
Büro-Domäne
Internet
Arbeitsplatz 1
Domänen-
Controller
Firewall
Mobiler
Arbeitsplatz 2
Proxy
Büro-Netzwerk
Mobiler
Arbeitsplatz 3
Dedizierter Multiuser-Server

Beschreibung der Netzwerkkonfiguration in einer Windows-Domäne mit Proxy

Die Netzwerkkonfiguration sollte folgende Voraussetzungen erfüllen.

Netzwerk-Umgebung	Aufbau: • Alle Windows-Arbeitsplätze, auf denen das TIA Portal installiert ist, sind Mitglied in derselben Domäne. • Ein dedizierter Windows-Server als Multiuser-Server befindet sich ebenfalls in derselben Domäne. • Alle Arbeitsplätze und der Multiuser-Server nutzen dasselbe LAN (Local Area Network). • Sowohl die Arbeitsplätze als auch der Multiuser-Server haben Zugriff auf den Domänen-Controller. • Zusätzlich können sich Arbeitsplätze außerhalb des LAN über einen Proxy-Server zum LAN verbinden.
Netzwerk	Voraussetzungen: • Es besteht eine direkte TCP-Verbindung zwischen den Arbeitsplätzen und dem Multiuser-Server über den ausgewählten Port. • Keine ausgehende Firewall-Einstellung blockiert die Kommunikation auf den Arbeitsplätzen. • Keine eingehende Firewall-Einstellung blockiert die Kommunikation auf dem Server. • Sowohl der Multiuser-Server als auch alle Arbeitsplätze sind im gleichen IP-Netz (ohne Routing). • Falls ein Router dazwischengeschaltet ist, darf der Router die Kommunikation über den ausgewählten Server-Port nicht blockieren. • Entweder ist kein Proxy-Server eingerichtet oder es existiert eine lokale Umgehung. Dazu sollte in den Windows-Einstellungen für ein lokales Netzwerk die Option "Proxyserver für lokale Adressen umgehen" aktiviert sein. • Aus Sicherheitsgründen wird nicht empfohlen, den Mulituser-Server im Internet zur Verfügung zu stellen. • Aus Sicherheitsgründen wird das Einrichten einer Firewall dringend empfohlen, die eingehende Verbindungen auf den ausgewählten Port akzeptiert. • Der Proxy-Server muss HTTPS-Kommunikation akzeptieren und auf dem aussgewählten Port zum Multiuser-Server weiterleiten.

Authentifizierung	Die Authentifizierung erfolgt abhänging von der gewählten Konfiguration. User Accounts und Windows-Arbeitsgruppen bei Konfiguration als Büro-Arbeitsgruppe: • Diese müssen lokal auf dem Multiuser-Server angelegt werden. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Name und Passwort für die Authentifizierung werden auf allen Arbeitsplätzen abgefragt. • Für die Authentifizierung können nur lokale Accounts des Server-Systems genutzt werden. Hinweis: Zur Abfrage von Namen und Passwort kann die Windows-Authentifizierung verwendet werden. Dazu müssen Sie lokale Accounts anlegen, die identisch sind zu den Accounts, die auf dem Multiuser-Server vorhanden sind. Diese Vorgehensweise ermöglicht es, dass der Server einen lokalen Account mit identischen Login-Daten zu einem bestehenden Server-Account automatisch authentifiziert. Einschränkung: Anwender-Accounts und Gruppen können nicht zentral verwaltet werden, dazu werden lokale Anwender und Gruppen benötigt. Diese Konfiguration ist deshalb nicht geeignet für Enterprise-Netzwerke. User Accounts und Windows-Arbeitsgruppen bei Konfiguration als Windows-Domäne: • Diese müssen auf dem Domänen-Controller angelegt sein. • Diese müssen entsprechend den Multiuser-Rechten und Rollen für Server und Projekte konfiguriert werden. • Anwender an Arbeitsplätzen, auf denen das TIA Portal läuft, werden automatisch authentifiziert mit Ihren Domänen-Login-Daten. Hinweis: Auf dem Multiuser-Server können Sie sich auch mit anderen gültigen Login-Daten anmelden. Einschränkung: Diese Konfiguration ist nicht geeignet für temporäre Server-Konfigurationen, da diese ebenfalls Zugriff auf den Domänen-Controller benötigen. Die Kommunikation mit einem Remote-Rechner über einen Proxy zeigt für alle Client-Server-Anwendungen typische Performance-Einschränkungen. Der Multiuser-Server ist davon ebenfalls betroffen.
Port	Hinweis: Teilweise ist es nur möglich, für eine https-Verbindung auf einem Proxy-Server den Port "443" zu verwenden. Ein anderer Port als "443" für eine https-Verbindug kann auf einigen Proxy-Servern nicht eingestellt werden. In diesem Fall ist es notwendig, den Multiuser-Server ebenfalls auf Port "443" zu installieren.

2.14 Mit dem Multiuser-Server arbeiten

2.14.1 Einführung zum Multiuser-Server

Einführung

Um mit der vollen Funktionalität von Multiuser Engineering arbeiten zu können, benötigen Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet.

Der Multiuser-Server muss zunächst installiert und mithilfe der Multiuser-Tools konfiguriert und gestartet werden.

Danach können die Verbindungen zum Multiuser-Server im TIA Portal über den Befehl "Extras" > "Einstellungen" > "Multiuser" projektiert und verwaltet werden.

Sie können im TIA Portal folgende Aktionen für die Serververbindungen durchführen:

- Neue Multiuser-Serververbindungen hinzufügen.
- Vorhandene Multiuser-Serververbindungen anzeigen, bearbeiten und löschen.

Eigenschaften des Multiuser-Servers

Der Multiuser-Server hat folgende Eigenschaften:

- Zur Installation und Konfiguration des Multiuser-Servers werden Administratorrechte benötigt.
- Die Installation des Servers benötigt Einstellungen in der Windows-Firewall und ggf. ein Zertifikat.
- Der Multiuser-Server wird mit den mitgelieferten Multiuser-Tools von einem Administrator konfiguriert und verwaltet.
- Er steht rechteabhängig allen Bearbeitern des Multiuser-Projektes zur Verfügung und kann auch von außen erreicht werden.
- Der Multiuser-Server kann außerhalb des TIA Portals gestartet und verwaltet werden.
- Der als Name vergebene "Alias" für den Multiuser-Server dient als Indentifikator und darf nachträglich nicht mehr verändert werden.

Microsoft System-Account für Multiuser-Server

Bei der Installation des Multiuser-Servers wird ein Microsoft System-Account für den Multiuser-Server erzeugt. Für diesen Account tauscht Microsoft zyklisch das Passwort aus.

Falls das Passwort für diesen Account abgelaufen ist, muss das Passwort erneuert werden und der Multiuser-Server danach erneut gestartet werden.

Mengengerüst für den Multiuser-Server

Für das effektive Arbeiten mit Multiuser Engineering können Sie maximal 100 Multiuser-Serververbindungen anlegen.

Wird diese Anzahl erreicht, so erhalten Sie eine Nachricht, dass keine neue Serververbindung mehr angelegt werden kann. Nach dem Löschen von nicht mehr benötigten Serververbindungen können wieder neue Verbindungen bis zur Maximalanzahl hinzugefügt werden.

Folgendes Mengengerüst für einen Multiuser-Server steht hardwareabhängig zur Verfügung:

Vorhandene Projekte und Revisionen auf dem Multiuser-Server	Standard Multiuser-Server (Anwendung als temporärer Multiuser-Server) Hardware: Field PG M5 Advanced (oder entsprechend) Intel® Core™ i5-6440EQ bis zu 3.4 GHz·,16 GB RAM (32 GB für große Projekte)· SSD mit 50 GB freiem Plattenspeicher Netzwerk 1 GBit 15.6" Full HD Display (1920 x 1080)
Maximale Anzahl der Serverprojekte pro Server:	1000
Maximale Anzahl der aktiven Projekte pro Server:	100
Maximale Anzahl der aktiven Arbeitsbereiche je Projekt	64
Maximalte Anzahl der parallelen Ladevorgänge pro Server	10
Maximale Anzahl der gespeicherten Revisionen pro Server	unbegrenzt, Anzahl ist parametrierbar Voreinstellung = 10 Beachten Sie, dass ausreichend Speicherplatz für das Speichern der Revisionen verfügbar ist.

Hinweis

Server-Hardware

Beim Einsatz leistungsfähigerer Server-Hardware sind auch höhere Mengengerüste möglich.

Siehe auch

Multiuser-Serververbindung anzeigen (Seite 124)

Multiuser-Serververbindung bearbeiten (Seite 125)

Multiuser-Serververbindung löschen (Seite 127)

Multiuser-Netzwerkprofile einstellen (Seite 122)

2.14.2 Multiuser-Netzwerkprofile einstellen

Einführung

Beim Arbeiten mit Multiuser Engineering haben Sie die Möglichkeit, zwischen drei unterschiedlichen Netzwerkprofilen zu wählen, abhängig von der Leistungsfähigkeit Ihres eigenen Netzwerkes.

Das ausgewählte Netzwerkprofil bestimmt die Zeit, bis es zu einer Zeitüberschreitungsmeldung kommt.

Zur Verfügung stehende Multiuser-Netzwerkprofile

Folgende Netzwerkprofile stehen Ihnen beim Arbeiten mit dem Multiuser-Server zur Verfügung:

Netzwerkprofil	Einsatzbereich
"Schnell"	• Büroumgebung • Gigabit-Ethernet • Dauerhafte und meist störungsfreie Netzwerkumgebung
"Mittel"	• Büroumgebung, kleine Arbeitsgruppe mit festem Standort • Schnelles Ethernet, WLAN, VPN-Verbindungen • Gute Netzwerkumgebung, Störungen eher selten
"Langsam"	• Inbetriebnahme, wechselnde Arbeitsstätten • WLAN-Verbindung • Störungen möglich (Maschinen, elektrische Anlagen)

Multiuser-Netzwerkprofile und Kompression

Sie haben die Möglichkeit, aus der Klappliste das gewünschte Netzwerkprofil auszuwählen, das zur Leistungsfähigkeit des von Ihnen verwendeten Netzwerks passt.

Das ausgewählte Multiuser-Netzwerkprofil bestimmt die Zeit, bis es zu einer Zeitüberschreitungsmeldung kommt..

Falls als Netzwerkprofil "Mittel" oder "Langsam" eingestellt ist, wird die Komprimierung für die Datenübertragung aktiviert.

Für das Netzwerkprofil "Schnell" wird keine Komprimierung durchgeführt.

Hinweis

Die Umstellung des Netzwerkprofils wird erst nach einem Neustart des Multiuser-Clients oder des Multiuser-Servers wirksam.

Multiuser-Netzwerkprofile einstellen

Um die Netzwerkprofile im TIA Portal einzustellen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras" > "Einstellungen" > "Multiuser".
   In den nachfolgenden angezeigten Einstellungen werden die möglichen Optionen für die Netzwerkprofile angezeigt.
2. Wählen Sie aus der Klappliste bei "Multiuser-Netzwerkprofile" das gewünschte Profil aus.
3. Starten Sie entweder den Multiuser-Client oder den Multiuser-Server neu, damit die Änderung des Netzwerkprofils wirksam wird.

Ergebnis

Das ausgewählte Profil wird aktiviert.

Abhängig vom ausgewählten Profil wird die Kompression aktiviert.

Hinweis

Beim Umstellen wird das Zeitüberschreitungsverhalten des aktuellen Netzwerkprofils nicht beeinträchtigt.

2.14.3 Neue Multiuser-Serververbindung hinzufügen

Einführung

Wenn Sie mit Multiuser Engineering arbeiten und einen Multiuser-Server installiert haben, können Sie im TIA Portal bis zu 100 Serververbindungen zum Multiuser-Server hinzufügen.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Der Multiuser-Server muss mithilfe der Configuration und Administration Tools konfiguriert und gestartet sein.

Neue Multiuser-Serververbindung hinzufügen

Um eine neue Multiuser-Serververbindung hinzuzufügen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Das Register "Multiuser-Serververbindungen" wird geöffnet und alle vorhandenen Serververbindungen werden angezeigt.
   Wenn noch keine Multiuser-Serververbindung projektiert wurde, ist nur der Eintrag für den "lokalen Multiuser-Server" vorhanden.
2. Klicken Sie im folgenden Editor in der Spalte "Server-Name" auf den Befehl "Serververbindung hinzufügen".
   Falls die maximale Anzahl von 100 Serververbindungen bereits erreicht ist, erhalten Sie eine Nachricht, dass Sie zuerst eine der vorhandenen Serververbindungen löschen müssen, um eine neue hinzufügen zu können.
   Falls das Hinzufügen einer neuen Serververbindung möglich ist, wird der Dialog "Neue Multiuser-Serververbindung hinzufügen" geöffnet.
3. Tragen Sie im Feld "Server-Name" den Namen für den Multiuser-Server ein.
   Für den Namen und die Pfad stehen Ihnen maximal 256 Zeichen zur Verfügung. Erlaubt sind Buchstaben, Zeichen, Ziffern und Kommata.
   Falls der eingegebene Name bereits verwendet wird oder der Name und der Pfad mehr als insgesamt 256 Zeichen lang ist, erhalten Sie einen Hinweis.
4. Wählen Sie das gewünschte Übertragungsprotokoll (https oder http) für den Multiuser-Server aus. Per Voreinstellung ist das Protokoll "http" angewählt.
5. Tragen Sie den gewünschten Host und Port für die Serververbindung ein.
   Falls der eingegebene Host und/oder Port bereits verwendet werden, erhalten Sie einen Hinweis.
6. Klicken Sie auf die Schaltfläche "Hinzufügen".

Ergebnis

Die neue Multiuser-Serververbindung wird hinzugefügt und unter "Extras > Einstellungen > Multiuser" angezeigt.

2.14.4 Multiuser-Serververbindung anzeigen

Einführung

Wenn Sie mit Multiuser Engineering arbeiten, können Sie sich im TIA Portal bestehende Serververbindungen zum Multiuser-Server in den Einstellungen anzeigen lassen.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Der Multiuser-Server muss mithilfe der Multiuser-Tools konfiguriert und gestartet sein.

Multiuser-Serververbindung anzeigen

Um die vorhandenen Multiuser-Serververbiundungen anzuzeigen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Im nachfolgenden Editor werden in der Spalte "Server-Name" die vorhandenen Multiuser-Serververbindungen angezeigt.
2. Klicken Sie auf die Spaltenüberschriften "Server-Name" oder "Host", um ggf. den Inhalt zu sortieren, falls Sie nach einer bestimmten Serververbindung suchen.

Ergebnis

Die vorhandenen Multiuser-Serververbindungen werden im Editor unter "Extras > Einstellungen > Multiuser" angezeigt.

2.14.5 Multiuser-Serververbindung bearbeiten

Einführung

Wenn Sie mit Multiuser Engineering arbeiten, können Sie in den Einstellungen vom TIA Portal bestehende Serververbindungen zum Multiuser-Server bearbeiten und neue Serververbindungen hinzufügen.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Der Multiuser-Server muss mithilfe der Multiuser-Tools konfiguriert und gestartet sein.

Multiuser-Serververbindung bearbeiten

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Im nachfolgenden Dialog werden in der Spalte "Server-Name" alle vorhandenen Multiuser-Serververbindungen aufgelistet.

   Hinweis

   Es ist nicht erlaubt, eine Multiuser-Serververbindung zu bearbeiten, die in einer aktuell geöffneten lokalen Session verwendet wird!

2. Klicken Sie ggf. auf die Spaltenüberschriften "Server-Name" oder "Host", um den Inhalt zu sortieren.
3. Doppelklicken Sie auf die Serververbindung, die Sie bearbeiten möchten.
   Ergebnis: Der Dialog "Serververbindung bearbeiten" wird geöffnet.

4. Tragen Sie hier die gewünschten Änderungen ein.
   – Das Feld "Server-Name" ist nicht editierbar.
   – Der Eintrag im Feld "Host" kann bearbeitet werden.
   – Der Eintrag im Feld "Port" kann bearbeitet werden.
5. Wählen Sie als Übertragungsprotokoll möglichst "https" aus, um einen sicheren Datenaustausch mit Verschlüsselung und Zertifikat zu erhalten.
6. Klicken Sie auf die Schaltfläche "OK", damit die Änderungen wirksam werden.

Ergebnis

Ihre Änderungen für die ausgewählte Multiuser-Serververbindung werden übernommen und unter "Extras > Einstellungen > Multiuser" angezeigt.

2.14.6 Multiuser-Serververbindung wieder herstellen

Einführung

Wenn Sie mit Multiuser Engineering arbeiten, kann die bestehende Serververbindung zum Multiuser-Server unterbrochen werden.

Ist dies der Fall, so erhalten Sie nach ca. 30 Sekunden eine entsprechende Fehlermeldung.

Sie haben unterschiedliche Optionen, abhängig davon, ob Sie in der lokalen Session oder in der Serverprojekt-Ansicht arbeiten.

Beim Arbeiten in der lokalen Session haben Sie folgende Optionen:

- Serververbindung wieder herstellen
- Speichern und lokale Session schließen
- Lokale Session ohne Speichern schließen

Beim Arbeiten in der Serverprojekt-Ansicht haben Sie folgende Optionen:

- Serververbindung wiederherstellen
- Änderungen verwerfen

Mögliche Vorgehensweisen in der lokalen Session

Zum Wiederherstellen einer Multiuser-Serververbindung, gehen Sie vor wie folgt:

1. Klicken Sie im angezeigten Dialog auf die Schaltfläche "Wiederherstellen".
   Wenn die Serververbindung wiederhergestellt werden kann, schließt der Dialog und Sie können ohne Einschränkung weiterarbeiten.
2. Klicken Sie auf "Speichern und lokale Session schließen", wenn Sie im Moment nicht weiterarbeiten, Ihre Änderungen jedoch speichern möchten.
3. Klicken Sie auf "lokale Session schließen", wenn Sie die lokale Session ohne Speichern verlassen möchten.

Mögliche Vorgehensweisen in der Serverprojekt-Ansicht

Zum Wiederherstellen einer Multiuser-Serververbindung, gehen Sie vor wie folgt:

1. Klicken Sie im angezeigten Dialog auf die Schaltfläche "Wiederherstellen".
   Wenn die Serververbindung wiederhergestellt werden kann, schließt der Dialog und Sie können ohne Einschränkung weiterarbeiten.
2. Klicken Sie auf "Änderungen verwerfen", wenn Sie die Serverprojekt-Ansicht ohne Speichern verlassen und Ihre eventuell vorgenommenen Änderungen verwerfen möchten. Daraufhin wird die Serverprojekt-Ansicht geschlossen.

2.14.7 Multiuser-Serververbindung löschen

Einführung

Wenn Sie mit Multiuser Engineering arbeiten, können Sie in den Einstellungen vom TIA Portal bestehende Serververbindungen zum Multiuser-Server löschen.

Sie können im TIA Portal bis zu 100 Multiuser-Serververbindungen anlegen, bearbeiten und löschen.

Hinweis

Beachten Sie, dass Sie eine Serververbindung nur dann löschen können, wenn vorher alle bestehenden lokalen Sessions zu dieser Serververbindung geschlossen wurden.

Beachten Sie, dass eine Verbindung von einer lokalen Session zu einer gelöschten Multiuser-Serververbindung nicht mehr möglich ist!

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Der Multiuser-Server muss mit Hilfe der Multiuser-Tools konfiguriert und gestartet sein.

Multiuser-Serververbindung löschen

Um einen vorhandenen Multiuser-Serververbindung zu löschen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Im nachfolgenden Editor in der Spalte "Server-Name" werden alle vorhandenen Multiuser-Serververbindungen aufgelistet.
2. Klicken Sie ggf. auf die Spaltenüberschriften "Server-Name" oder "Host", um den Inhalt zu sortieren.

3. Selektieren Sie einen oder mehrere Serververbindungen, die Sie löschen möchten, und klicken Sie im Kontextmenü auf den Befehl "Löschen".
   Der Dialog "Entfernen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "Ja", um die nachfolgende Meldung zu bestätigen und die ausgewählte Serververbindung endgültig zu löschen.

Ergebnis

Die selektierte Multiuser-Serververbindung ist gelöscht und wird unter "Extras > Einstellungen > Multiuser" nicht mehr angezeigt.

2.15 Mit dem lokalen Multiuser-Server arbeiten

2.15.1 Einführung zum lokalen Multiuser-Server

Einführung

Um mit Multiuser Engineering arbeiten zu können, benötigen Sie einen Multiuser-Server, der Ihre Multiuser-Projekte verwaltet.

Ein lokaler Multiuser-Server wird bei der Installation des TIA Portals automatisch immer mitinstalliert und kann nicht gelöscht werden. Er benötigt keine separate Software-Installation.

Der lokale Multiuser-Server wird in der Liste der Multiuser-Server als erster Eintrag in allen Sprachen unter dem nicht veränderbaren Namen "Local Multiuser Server" angezeigt und kann über den Befehl "Extras" > "Einstellungen" > "Multiuser" aufgerufen werden.

Vorteile des lokalen Multiuser-Servers

Der lokale Multiuser-Server bietet Ihnen folgende Vorteile:

- Sie können Multiuser Engineering ausprobieren, ohne eine vorherige Installation eines dezidierten Multiuser-Servers durchzuführen.
- Sie können Ihren Arbeitsfortschritt auf dem lokalen Multiuser-Server zwischenspeichern.

Voraussetzung

Um die Funktionalität des lokalen Multiuser-Servers nutzen zu können, muss das TIA Portal installiert sein.

Eigenschaften des lokalen Multiuser-Servers

Im Gegensatz zu einem dezidierten Multiuser-Server verfügt der lokale Multiuser-Server nur über eine eingeschränkte Funktionalität.

Der lokale Multiuser-Server hat folgende Eigenschaften:

- Er steht jeweils nur einem Anwender auf seinem Rechner zur Verfügung und kann nicht gelöscht werden.
- Er kann nur innerhalb des TIA Portals gestartet werden.
- Er kann nur von dem Rechner aus aufgerufen werden, auf dem auch die lokale Session läuft, die diesem lokalen Multiuser-Server zugeordnet ist.
- Er unterstützt keine Verwaltungsaufgaben.
- Er benötigt keine Anpassungen in den Einstellungen für die Firewall und keine Zertifikate.
- Nur der Port ist editierbar, Server-Name und Host können nicht verändert werden.

Siehe auch

Lokalen Multiuser-Server konfigurieren (Seite 130)

Lokalen Multiuser-Server anzeigen (Seite 131)

Lokalen Multiuser-Server starten (Seite 132)

Lokalen Multiuser-Server beenden (Seite 133)

2.15.2 Lokalen Multiuser-Server konfigurieren

Einführung

Ein lokaler Multiuser-Server wird bei der Installation des TIA Portals defaultmäßig mitinstalliert.

Er wird in der Liste der Multiuser-Server als erster Eintrag angezeigt und kann über den Befehl "Extras > Einstellungen > Multiuser" unter "Multiuser-Serververbindung" aufgerufen werden.

Hinweis

Es ist nicht erlaubt, den lokalen Multiuser-Server zu konfigurieren, während dieser mit einer geöffneten lokalen Session verbunden ist.

Falls bereits eine lokale Session angelegt wurde und der lokale Multiuser-Server erst danach umkonfiguriert wird, muss der Server zunächst gestoppt und anschließend erneut gestartet werden, damit die vorhandene lokale Session nicht ungültig wird.

Voraussetzung

Sie haben das TIA Portal installiert.

Lokalen Multiuser-Server konfigurieren

Um den lokalen Multiuser-Server zu konfigurieren, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Unter "Multiuser-Serververbindung" werden in der Spalte "Server-Name" alle vorhandenen Multiuser-Server angezeigt.
2. Klicken Sie ggf. auf die Spaltenüberschriften "Server-Name" oder "Host", um den Inhalt zu sortieren.
3. Positionieren Sie den Mauszeiger in der entsprechenden Zeile und rufen Sie über das Kontextmenü den Befehl "Serververbindung bearbeiten" auf.
   - Die Felder "Server-Name" und "Host" sind für den lokalen Multiuser-Server nicht editierbar.
   - Der Eintrag im Feld "Port" kann bearbeitet werden. Tragen Sie hier den gewünschten Port ein.
4. Klicken Sie auf die Schaltfläche "OK", damit die Änderungen wirksam werden.

Ergebnis: Ihre Änderungen für den lokalen Multiuser-Server werden übernommen und unter "Extras > Einstellungen > Multiuser" unter "Multiuser-Serververbindung" angezeigt.

Hinweis

Der lokale Multiuser-Server kann nicht gelöscht werden.

Der lokale Multiuser-Server wird in allen Sprachen mit dem nicht veränderbaren Server-Namen "Local Multiuser Server" angezeigt.

Siehe auch

Lokalen Multiuser-Server beenden (Seite 133)

2.15.3 Lokalen Multiuser-Server anzeigen

Einführung

Ein lokaler Multiuser-Server wird bei der Installation des TIA Portals per Voreinstellung mitinstalliert.

Voraussetzung

Sie haben das TIA Portal installiert.

Multiuser-Server anzeigen

Um den lokalen Multiuser-Server anzuzeigen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Extras > Einstellungen > Multiuser".
   Der lokale Multiuser-Server wird unter "Multiuser-Serververbindung" in der Spalte "Server-Name" an erster Stelle angezeigt.
   Nachfolgend werden die von Ihnen angelegten dezidierten Multiuser-Server aufgelistet.
   Falls der lokale Multiuser-Server nicht an erster Stelle angezeigt wird, klicken Sie auf die Spalte "Server-Name", um die angezeigten Einträge zu sortieren.

Hinweis

Der lokale Multiuser-Server kann nicht gelöscht werden.

Der lokale Multiuser-Server wird in allen Sprachen mit dem nicht veränderbaren Server-Namen "Local Multiuser Server" angezeigt.

2.15.4 Lokalen Multiuser-Server starten

Einführung

Ein lokaler Multiuser-Server wird bei der Installation des TIA Portals per Voreinstellung mitinstalliert.

Er wird in der Liste der Multiuser-Server als erster Eintrag angezeigt und kann über den Befehl "Extras > Einstellungen > Multiuser" unter "Multiuser-Serververbindung" aufgerufen werden.

Lokalen Multiuser-Server starten

Um den lokalen Multiuser-Server zu starten, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Der Dialog "Multiuser-Serverprojekte verwalten" wird geöffnet.
2. Wählen Sie im Feld "Server auswählen" den lokalen Multiuser-Server aus der Klappliste aus. Er wird unter dem Namen "Local Multiuser Server" angezeigt.
   - Falls bereits Serverprojekte vorhanden sind, werden diese aufgelistet. In den einzelnen Spalten erhalten Sie detaillierte Informationen über das jeweilige Projekt bezüglich Pfad und Änderungsdatum. Falls gewünscht, können Sie die Anordnung der Spalten verändern.
   - Falls der lokale Multiuser-Server noch nicht gestartet wurde, werden Sie gefragt, ob Sie diesen nun starten möchten.
   - Sollte keine Verbindung zum ausgewählten Server möglich sein, erhalten Sie eine Information.
3. Bestätigen Sie die nachfolgende Anfrage, ob der lokale Multiuser-Server auf Ihrem Rechner gestartet werden soll, mit "Ja".

Ergebnis

Die Verbindung zum lokalen Multiuser-Server wird aufgebaut. Eine Kommandofenster wird geöffnet und zeigt an, dass der Server nun aktiviert ist.

Hinweis

Durch das Schließen des TIA Portals wird die Verbindung zum lokalen Multiuser-Server zwar beendet, dieser wird jedoch nicht automatisch gestoppt.

Um den lokalen Multiuser-Server zu stoppen, muss die angezeigte Kommandozeile geschlossen werden.

Dem lokalen Multiuser-Server zugeordnete lokale Session öffnen

Sobald Sie eine lokale Session öffnen, die dem lokalen Multiuser-Server zugeordnet ist und der lokale Multiuser-Server noch nicht gestartet wurde, werden Sie gefragt, ob Sie die lokalen Multiuser-Server starten wollen.

Beantworten Sie diese Frage mit "Ja", so wird der lokale Multiuser-Server gestartet und die lokale Session zum Bearbeiten geöffnet.

Bei "Nein" brechen Sie den Vorgang ab.

Hinweis

Beachten Sie, dass Sie eine Serververbindung nur dann löschen können, wenn vorher alle bestehenden lokalen Sessions zu dieser Serververbindung gelöscht wurden.

Beachten Sie, dass eine Verbindung von einer bestehenden lokalen Session zu einem gelöschten Multiuser-Serverprojekt nicht mehr möglich ist.

2.15.5 Lokalen Multiuser-Server beenden

Einführung

Beim Öffnen einer lokalen Session, die dem lokalen lokalen Multiuser-Server zugeordnet ist, erfolgt die Abfrage, ob der lokale Multiuser-Server gestartet werden soll. Wird die lokale Session beendet oder das TIA Portal geschlossen, so wird der lokale Multiuser-Server nicht automatisch mit beendet.

Der lokale Multiuser-Server muss explizit beendet werden.

Voraussetzung

Der lokale Multiuser-Server wurde gestartet.

Vorgehensweise

Um den lokalen Multiuser-Server zu beenden, gehen Sie vor wie folgt:

1. Beenden Sie den lokalen Multiuser-Server, indem Sie die angezeigte Kommandozeile schließen.

2.16 Multiuser-Serverprojekte erstellen und verwalten

2.16.1 Einführung zum Arbeiten in Multiuser-Serverprojekten

Einführung

Nachdem der Multiuser-Server installiert und mit Hilfe der Multiuser-Tools konfiguriert und gestartet wurde, können Sie im TIA Portal die Multiuser-Serverprojekte verwalten.

Multiuser-Serverprojekte verwalten

Wenn Sie den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten" ausführen, werden im nachfolgenden Dialog alle bestehenden Serverprojekte und lokalen Sessions zu den vorhandenen Serververbindungen angezeigt.

Falls Sie noch kein Multiuser-Serverprojekt hinzugefügt haben, ist die Liste leer.

Falls es schon Serverprojekte gibt, werden diese angezeigt. Über die Klappliste im oberen Teil des Dialogs können Sie die gewünschte Serververbindung auswählen, zu der dann alle bestehenden Serverprojekte und lokalen Sessions aufgelistet werden.

Über die einzelnen Schaltflächen können Sie folgende Aktionen durchführen:

- "Server auswählen": Fügt eine neue Serververbindungen hinzu.
- "Aktualisieren": Aktualisiert die Anzeige im Dialog.
- "Serverprojekte verwalten": Öffnet in den Einstellungen die Anzeige für die Multiuser-Serverprojekte.
- "Projekt zum Server hinzufügen": Fügt ein neues Serverprojekt zur ausgewählten Serververbindung hinzu.
- "Neue lokale Session erstellen": Erstellt eine neue lokale Session zum ausgewählten Serverprojekt.
- "Öffnen": Öffnet die ausgewählte lokale Session.
- "Schließen": Schließt den Dialog.

Hinweis

Serverprojekte anzeigen

Es können nur dann Serverprojekte angezeigt werden, wenn eine Verbindung zum Multiuser-Server besteht und dieser gestartet wurde.

Falls keine Verbindung aufgebaut werden konnte, erhalten Sie einen Hinweis.

Hinweis

Serverprojekte löschen

Um Inkonsistenzen in der Projektverwaltung zu vermeiden, dürfen Serverprojekte nicht "händisch" über den Explorer im Ablageverzeichnis gelöscht werden.

Serverprojekte müssen immer im Kontext von Multiuser Engineering über die Multiuser-Tools gelöscht werden.

Bevor ein Serverprojekt gelöscht werden kann, müssen alle zugehörigen lokalen Sessions gelöscht sein.

Befehle im Kontextmenü

Folgende Aktionen können im Dialog "Multiuser-Serverprojekte verwalten" über das Kontextmenü ausgeführt werden:

- Projekt zum Multiuser-Server hinzufügen.
- Neue lokale Session erstellen.
- Lokale Session löschen.
- Lokale Session als Single-User-Projekt exportieren.

Siehe auch

Neues Multiuser-Serverprojekt hinzufügen (Seite 135)

2.16.2 Neues Multiuser-Serverprojekt hinzufügen

Einleitung

Beim Arbeiten mit Multiuser Engineering muss zunächst ein Multiuser-Serverprojekt erstellt werden. Ein Multiuser-Serverprojekt entsteht dadurch, dass ein Standard-Projekt (Single-User-Projekt) zum bereits installierten Multiuser-Server hinzugefügt wird, um dieses Projekt dann als Multiuser-Serverprojekt nutzen zu können.

Hinweis

Multiuser-Serverprojekte

Beachten Sie die unter Anforderungen an Multiuser-Projekte (Seite 35) beschriebenen Voraussetzungen, die ein multiuserfähiges Projekt erfüllen muss.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Vorgehen

Um ein Multiuser-Serverprojekt zu erstellen, gehen Sie vor wie folgt:

1. Klicken Sie im TIA Portal auf den Menübefehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
2. Wählen Sie im folgenden Dialog aus der Klappliste die gewünschte Serververbindung aus. Falls Sie den "Local Multiuser Server" auswählen, werden Sie gefragt, ob Sie den lokalen Multiuser-Server starten möchten.
   Klicken Sie in diesem Fall auf "Ja".
3. Klicken Sie auf die Schaltfläche "Projekt zum Server hinzufügen".
4. Stellen Sie bei Bedarf im nachfolgenden Dialog den gewünschten Quellpfad zum Single-User-Projekt ein oder navigieren Sie über die "..."-Schaltfläche zum gewünschten Verzeichnis.
5. Wählen Sie das Projekt aus, das Sie als Multiuser-Serverprojekt hinzufügen möchten.
6. Tragen Sie unter "Details" eventuell einen Kommentar zum Multiuser-Serverprojekt ein. Die Informationen unter "Serverprojekt-Name" und "Veröffentlich von" sind immer vorbelegt und können nicht editiert werden.
7. Die Schaltfläche "Lokale Session erstellen" ist per Voreinstellung deaktiviert. Aktivieren Sie diese Schaltfläche, falls eine lokale Session für das neu hinzugefügte Multiuser-Serverprojekt erstellt werden soll.
8. Klicken Sie auf die Schaltfläche "Hinzufügen".
   Falls das ausgewählte Projekt bereits als Multiuser-Serverprojekt vorhanden ist oder gerade von einem anderen Bearbeiter geöffnet ist, erhalten Sie eine Information.
9. Der Dialog "Lokale Session erstellen" öffnet sich. Hie können Sie die Details zur lokalen Session eintragen. Aktivieren Sie auf die Schaltfläche "Lokale Session öffnen" und klicken Sie auf "Erstellen", um eine lokale Session zu erstellen.

Ergebnis

Das ausgewählte Single-User-Projekt wird als neues Multiuser-Serverprojekt hinzugefügt und unter "Projekt > Multiuser > Multiuser-Serverprojekte verwalten" angezeigt.

Falls die Option "Lokale Session erstellen" aktiviert war, wird eine neue lokale Session erstellt und geöffnet.

Siehe auch

Anforderungen an Multiuser-Projekte (Seite 35)

2.17 In der lokalen Session arbeiten

2.17.1 Lokale Session erstellen und verwalten

2.17.1.1 Lokale Session erstellen

Einleitung

Um mit Multiuser Engineering arbeiten zu können, müssen Sie eine lokale Session erstellen. Die lokale Session ist immer einem spezifischen Multiuser-Serverprojekt zugeordnet.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt angelegt.

Vorgehensweise

Um eine lokale Session zu erstellen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Der Dialog "Multiuser-Serverprojekte verwalten" öffnet sich.
2. Wählen Sie aus der Klappliste den gewünschten Multiuser-Server aus.
3. Wählen Sie aus der angezeigten Liste das gewünschte Projekt aus und doppelklicken Sie auf die Schaltfläche "Neue lokale Session erstellen".
   Der Dialog "Lokale Session erstellen" wird geöffnet.
4. Kontrollieren Sie im Dialog "Lokale Session erstellen" Ihre Auswahl bezüglich des Multiuser-Servers und des Multiuser-Serveprojekts.
   Die angezeigte Auswahl kann in diesem Dialog nicht verändert werden.
   Falls die angezeigte Auswahl Ihren Wünschen nicht entspricht, klicken Sie auf die Schaltfläche "Abbrechen" und beginnen wieder bei Punkt 1.

5. Fügen Sie die Detailinformationen zur lokalen Session ein:
   - Tragen Sie einen gültigen Pfad für die Ablage der lokalen Session ein oder nutzen Sie die entsprechende Schaltfläche, um zur gewünschten Stelle zu navigieren.
   - Vergeben Sie einen Namen für das Verzeichnis, in dem die lokale Session gespeichert werden soll. Das Verzeichnis wird implizit erstellt, falls es noch nicht vorhanden ist.

     Hinweis

     Beachten Sie, dass der Autor für die lokale Session bereits per Voreinstellung eingetragen wurde und nicht bearbeitet werden kann.

   - Die Schaltfläche "Lokale Session öffnen" ist per Voreinstellung aktiviert und bewirkt, dass die neue lokale Session geöffnet wird, sobald sie erstellt wurde. Deaktivieren Sie diese Schaltfläche, falls gewünscht.
6. Klicken Sie auf die Schaltfläche "Erstellen", um die Aktion durchzuführen.
   Eine Meldung zeigt an, dass die lokale Session erstellt wird.

Ergebnis

Die neu erstellte lokale Session wird geöffnet und in der Projektnavigation angezeigt.

Siehe auch

Lokale Session öffnen (Seite 138)

Lokale Session speichern (Seite 141)

Lokale Session schließen (Seite 142)

Lokale Session löschen (Seite 143)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.17.1.2 Lokale Session öffnen

Einführung

Sie haben verschiedene Möglichkeiten, eine lokale Session zu öffnen:

- Im TIA Portal über "Zuletzt verwendete Projekte"
- Im TIA Portal über den Befehl "Projekte > Multiuser > Multiuser-Serverprojekte verwalten".
- Bei geschlossenem TIA Portal mit dem Windows Explorer

Beim Öffnen einer lokalen Session, die dem lokalen Multiuser-Server zugeordnet ist, werden Sie gefragt, ob Sie den lokalen Multiuser-Server starten möchten, falls dieser noch nicht gestartet wurde. Klicken sie auf "Ja", um den lokalen Multiuser-Server zu starten und die ausgewählte lokale Session zu öffnen.

Hinweis

Hinweise zum Öffnen von lokalen Sessions:

- Eine umbenannte oder gelöschte lokale Session kann nicht mehr geöffnet werden.
- Eine lokale Session kann nicht als Referenzprojekt geöffnet werden.

Beim Öffnen einer lokalen Session, die einem der o. g. Kriterien entspricht, erhalten Sie einen Hinweis, dass das Öffnen nicht möglich ist. In diesem Fall werden Sie gefragt, ob Sie diese lokale Session als Single-User-Projekt exportieren möchten.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt.

Lokale Session öffnen

Um eine kürzlich verwendete lokale Session zu öffnen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Öffnen".
   Der Dialog "Projekt öffnen" mit einer Liste der kürzlich verwendeten Projekte und Sessions wird angezeigt.
   Die in der ersten Spalte angezeigten Symbole dienen zur Unterscheidung von Single-User- und Multiuser-Projekten.
2. Wählen Sie die gewünschte lokale Session aus und klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

Die ausgewählte lokale Session wird geöffnet.

Lokale Session über das zugehörige Serverprojekt öffnen

Um eine lokale Session über das zugehörige Serverprojekt zu öffnen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Der Dialog "Multiuser-Serverprojekte verwalten" öffnet sich.
2. Wählen Sie aus der Klappliste den gewünschten Multiuser-Server aus.
3. Wählen Sie aus der angezeigten Liste die gewünschte lokale Session aus und klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

Alle geöffneten Dialoge werden geschlossen und die ausgewählte lokale Session wird geöffnet.

Lokale Session über den Windows Explorer öffnen

Um eine lokale Session über den Windows Explorer zu öffnen, gehen Sie vor wie folgt:

1. Navigieren Sie bei geschlossenem TIA Portal im Windows-Explorer zu dem Verzeichnis, in dem Ihre lokalen Sessions abgelegt sind.
2. Wählen Sie die gewünschte lokale Session mit der Dateikennung "*.als<version>" aus und öffnen Sie diese mit Doppelklick.

Ergebnis

Das TIA Portal wird gestartet und die ausgewählte lokale Session wird geöffnet.

Hinweis

Das Schließen einer lokalen Session erfolgt wie beim Single-User-Projekt über den Befehl "Projekt > Schließen".

Lokale Session von einem anderen Ablageort aus öffnen

Um eine lokale Session zu öffnen, die zwischenzeitlich auf einen anderen Ablageort verschoben wurde, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Öffnen".
   Der Dialog "Projekt öffnen" mit einer Liste der kürzlich verwendeten Projekte und Sessions wird angezeigt.
2. Wählen Sie die gewünschte lokale Session aus und klicken Sie auf die Schaltfläche "Öffnen".
   Sie erhalten die Information, dass Sie versuchen, eine lokale Session von einem anderen Ablageort aus zu öffnen.
3. Bestätigen Sie die angezeigte Abfrage mit "OK", falls Sie die Aktion durchführen wollen.

Ergebnis

Die ausgewählte lokale Session wird vom neuen Ablageort aus geöffnet.

Die Informationen bezüglich des neuen Ablageortes werden im zugehörigen Server gespeichert.

Lokale Session öffnen ohne Serververbindung

Eine lokale Session kann auch geöffnet werden, wenn der zugehörige Server gerade nicht verfügbar ist. In diesem Fall erhalten Sie beim Öffnen der lokalen Session eine Meldung, dass Sie in der lokalen Session offline weiter arbeiten können mit folgenden Einschränkungen:

- Sie können die Änderungen der anderen Bearbeiter in Ihrer lokalen Session nicht sehen. Es werden keine gelbe oder rote Fähnchen angezeigt und es werden auch keine "veralteten" Objekte angezeigt.
- Sie können ihre Änderungen erst dann ins Serverprojekt einchecken, wenn die Serververbindung wieder hergestellt ist.

Wenn Sie diese Meldung mit "OK" bestätigen, können Sie in Ihrer lokalen Session auch ohne Serververbindung weiterarbeiten.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session speichern (Seite 141)

Lokale Session schließen (Seite 142)

Lokale Session löschen (Seite 143)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.17.1.3 Lokale Session speichern

Einführung

Sie haben folgende Möglichkeiten, eine lokale Session zu speichern:

- als "lokale Session" mit der Dateikennung "*.als"
- nach einem Export als "Single-User-Projekt" mit der Dateikennung "*.ap".

Wenn Sie eine lokale Session als Single-User-Projekt exportieren und speichern, können Sie dieses Projekt wie gewohnt ohne Multiuser-Funktionalität weiterbearbeiten.

Während des Arbeitens in der lokalen Session speichern Sie diese wie gewohnt als lokale Session, um im Kontext von Multiuser Engineering weiter zu arbeiten.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt.

Vorgehensweise

Um eine lokale Session zu speichern, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session.
2. Führen Sie die gewünschten Änderungen durch.
3. Wählen Sie im TIA Portal den Befehl "Projekt > Speichern".

Ergebnis

Die lokale Session wird im zugehörigen Multiuser-Server mit der Dateikennung "*.als" gespeichert.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session öffnen (Seite 138)

Lokale Session schließen (Seite 142)

Lokale Session löschen (Seite 143)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.17.1.4 Lokale Session schließen

Einführung

Das Schließen einer lokalen Session erfolgt wie beim Single-User-Projekt über den Befehl "Projekt > Schließen".

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt, die geöffnet ist.

Vorgehensweise

Um eine lokale Session zu schließen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Schließen".
2. Beantworten Sie die nachfolgende Abfrage mit "Ja", falls Sie Ihre Änderungen speichen möchten.

Ergebnis

Die lokale Session wird geschlossen.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session öffnen (Seite 138)

Lokale Session speichern (Seite 141)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.17.1.5 Lokale Session löschen

Einführung

Um Inkonsistenzen in Multiuser-Projekten zu vermeiden, kann das Löschen einer lokalen Session **nicht** wie gewohnt über den Befehl "Projekt > Projekt löschen..." erfolgen.

Lokale Sessions können nur im Kontext der Multiuser-Funktionalität und in Abhängigkeit vom zugehörigen Multiuser-Serverprojekt gelöscht werden.

Hinweis

Falls Sie eine lokale Session manuell mit dem Windows Explorer aus dem Ablageverzeichnis gelöscht haben, wird diese im Dialog "Lokale Sessions" auch weiterhin noch angezeigt, kann aber nicht mehr geöffnet werden.

Löschen Sie diese lokale Session dann explizit nochmal im Multiuser-Kontext, wie nachfolgend in der Vorgehensweise beschrieben.

Im Multiuser-Server vorhandene lokale Sessions können auch im Rahmen der Serververwaltung über die Multiuser-Tools gelöscht werden. Siehe hierzu:Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt.

Die zu löschende lokale Session darf nicht geöffnet sein.

Vorgehensweise

Um eine lokale Session im TIA Portal zu löschen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten...".
   Der Dialog "Multiuser-Projekte verwalten" mit einer Liste der zum ausgewählten Server vorhandenen Sessions wird angezeigt.
2. Wählen Sie den gewünschten Server aus der Klappliste aus.
3. Wählen Sie das gewünschte Serverprojekt aus der Liste aus und klicken Sie auf die entsprechende lokale Session, die Sie löschen wollen.

4. Wählen Sie im Kontextmenü den Befehl "Löschen".

Hinweis

Falls eine lokale Session geöffnet ist, kann sie nicht gelöscht werden, der Befehl zum Löschen ist dann nicht aktiv.

Schließen Sie in diesem Fall zuerst die lokale Session und klicken Sie erneut auf die Schaltfläche "Löschen".

5. Bestätigen Sie die nachfolgende Abfrage zum Löschen mit "Ja".

Ergebnis

Die ausgewählte lokale Session wird gelöscht und aus der Liste der vorhandenen lokalen Sessions entfernt.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session öffnen (Seite 138)

Lokale Session speichern (Seite 141)

Lokale Session schließen (Seite 142)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

2.17.1.6 Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen

Einführung

Sie haben die Möglichkeit, alle zu einem Multiuser-Serverprojekt gehörenden lokalen Sessions im TIA Portal anzuzeigen.

Voraussetzung

Sie haben das TIA Portal und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt.

Alle lokalen Sessions anzeigen

Um alle zu einem Serverprojekt gehörenden lokalen Sessions anzuzeigen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Der Dialog "Multiuser-Serverprojekte verwalten" öffnet sich und zeigt alle vorhandenen Serververbindungen zum Multiuser-Server an.
2. Wählen Sie die gewünschte Serververbindung aus der Klappliste aus.
   Alle zu dieser Serververbindung vorhandenen Serverprojekte werden aufgelistet.
3. Wählen Sie das gewünschte Serverprojekt aus und klappen Sie es auf, um alle zugehörigen lokalen Sessions anzuzeigen.
4. Falls Sie eine lokale Session öffnen möchten, wählen Sie die gewünschte Session aus und klicken Sie auf die Schaltfläche "Öffnen".

Ergebnis

Alle zum ausgewählten Serverprojekt vorhandenen lokalen Sessions werden angezeigt.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session öffnen (Seite 138)

Lokale Session speichern (Seite 141)

Lokale Session schließen (Seite 142)

Lokale Session löschen (Seite 143)

Lokale Session als Single-User-Projekt exportieren (Seite 145)

2.17.1.7 Lokale Session als Single-User-Projekt exportieren

Einführung

Beim Arbeiten mit Multiuser Engineering haben Sie die Möglichkeit, eine lokale Session als Single-User-Projekt zu exportieren.

Dies ist dann von Vorteil, wenn eine lokale Session momentan nicht geöffnet werden kann, z. B. weil der Multiuser-Server nicht verfügbar ist.

Es kann auch sein, dass eine lokale Session ungültig wird und deshalb nicht mehr geöffnet werden kann.

Eine lokale Session wird aus folgenden Gründen ungültig:

- Wenn die lokale Session umbenannt wurde.
- Wenn die lokale Session auf einen anderen Ablageort verschoben wurde.
- Wenn die Einstellungen für den zugehörigen Server geändert wurden und der Server dadurch nicht mehr verfügbar ist.

Wenn eine lokale Session beim Öffnen als ungültig erkannt wird, erhalten Sie eine Abfrage, ob Sie einen Export als Single-User-Projekt durchführen möchten.

Damit können Sie die Arbeitsergebnisse aus der lokalen Session in einem Single-User-Projekt sichern und weiterhin verwenden.

Sie können die Änderungen aus der exportierten lokalen Session über Kopieren wieder in Ihr Multiuser-Serverprojekt übernehmen.

Sie haben auch die Möglichkeit, eine lokale Session über die Multiuser Power Tools zu exportieren.

Siehe hierzu: Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

Lokale Session als Single-User-Projekt exportieren

Um eine lokale Session als Single-User-Projekt zu exportieren, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Multiuser > Multiuser-Serverprojekte verwalten".
   Der Dialog "Multiuser-Serverprojekte verwalten" öffnet sich und zeigt alle vorhandenen Serververbindungen zum Multiuser-Server an.
2. Wählen Sie die gewünschte Serververbindung aus der Klappliste aus.
   Alle zu dieser Serververbindung vorhandenen Serverprojekte werden aufgelistet.
3. Wählen Sie das gewünschte Serverprojekt aus und klappen Sie es auf, um alle zugehörigen lokalen Sessions anzuzeigen.
4. Wählen Sie die lokale Session aus und klicken Sie im Kontextmenü auf den Befehl "Als Single-User-Projekt exportieren".
   Der Dialog "Als Single-User-Projekt exportieren" wird geöffnet. Der Name der lokalen Session und der Name der Export-Datei sind voreingestellt und können nicht verändert werden.
5. Tragen Sie den gewünschten Export-Pfad ein oder navigieren Sie zum gewünschten Ablageort und klicken Sie auf "Exportieren".
   Falls das Zielverzeichnis nicht leer ist, erhalten Sie einen entsprechenden Hinweis. Wählen Sie ein neues, leeres Zielverzeichnis aus und klicken Sie erneut auf "Exportieren".

Ergebnis

Die lokale Session wird als Single-User-Projekt exportiert und unter dem eingestellten Ablagepfad gespeichert.

Sie erhalten eine Meldung, dass der Export erfolgreich war.

Wenn sie auf "OK" klicken, wird der Export-Dialog geschlossen und die Serverprojekt-Liste wird wieder angezeigt.

Siehe auch

Lokale Session erstellen (Seite 137)

Lokale Session öffnen (Seite 138)

Lokale Session speichern (Seite 141)

Lokale Session schließen (Seite 142)

Lokale Session löschen (Seite 143)

Alle zu einem Serverprojekt gehörenden lokalen Sessions anzeigen (Seite 144)

Befehle zur Verwaltung des Multiuser-Servers (Seite 93)

2.17.2 Objekte in der lokalen Session bearbeiten

2.17.2.1 Einführung zum Arbeiten in lokalen Sessions

Einführung

Um im Rahmen von Multiuser Engineering gleichzeitig mit mehreren Anwendern in einem Multiuser-Projekt arbeiten zu können, muss jeder Bearbeiter eine eigene lokale Session anlegen.

In einer oder mehreren lokalen Sessions kann der jeweilige Bearbeiter seine Änderungen einfügen und diese anschließend ins Multiuser-Serverprojekt einchecken und damit veröffentlichen.

Nach dem Einchecken stehen die Änderungen aus der lokalen Session wieder allen Bearbeitern im Serverprojekt zur Verfügung. Diese können weiter verwendet und in nachfolgenden lokalen Sessions nochmals modifiziert werden.

Vorgehensweise für das Arbeiten in lokalen Sessions

Um als Multiuser die Ihnen zugeteilten Engineeringaufgaben in einer lokalen Session bearbeiten zu können, sind folgende Schritte notwendig:

- Sie haben das TIA Portal und einen Multiuser-Server installiert.
- Alternativ können Sie den per Voreinstellung installierten lokalen Multiuser-Server verwenden.
- Sie haben ein ablauffähiges und für das Arbeiten mit Multiuser Engineering vorbereitetes Projekt angelegt.
- Sie haben das vorbereitete Projekt auf den Multiuser-Server geladen und damit ein Multiuser-Projekt erstellt.
- Sie haben die Aufgabenstellung und die Mengengerüste für das Arbeiten in den lokalen Sessions mit den anderen Bearbeitern abgestimmt.
- Sie haben für jede Engineering-Aufgabe eine lokale Session angelegt.

Regeln für das Arbeiten in lokalen Sessions

Für das Arbeiten in lokalen Sessions gelten folgende Regeln:

- Eine lokale Session ist immer einem definierten Multiuser-Server und einem definierten Multiuser-Serverprojekt zugeordnet.
- Jede lokale Session enthält unterschiedliche Aufgabenstellungen für unterschiedliche Bearbeiter, die einem Multiuser-Serverprojekt zugeordnet sind.
- Eine lokale Session hat beim Speichern die Dateikennung "*.als" und darf im Windows Explorer nicht umbenannt werden.
- Während eine lokale Session geöffnet ist, dürfen die zugehörige Multiuser-Serververbindung und das zugehörige Multiuser-Serverprojekt nicht umbenannt, verschoben oder gelöscht werden.
- Lokale Sessions können nicht mehr geöffnet und bearbeitet werden, wenn der zugehörige Multiuser-Server oder das zugehörige Multiuser-Serverprojekt gelöscht wurde.
- Eine lokale Session kann ab der V15 vom TIA Portal auch geöffnet werden, wenn keine Serververbindung besteht. Nach dem Bestätigen einer entsprechenden Meldung können Sie danach offline, d. h. ohne Serververbindung, weiterarbeiten.
- Eine lokale Session kann nicht als Referenzprojekt geöffnet werden.
- Eine lokale Session kann als Single-User-Projekt exportiert und ohne Multiuser-Funktionalität weiterbearbeitet werden.
- Eine lokale Session kann nicht archiviert oder dearchiviert werden. Diese Aktionen sind möglich, sobald die lokale Session wieder als Single-User-Projekt bearbeitet wird.
- Das Arbeiten mit externen Tools ist in der lokalen Session nicht möglich. Änderungen durch externe Tools in lokalen Sessions gehen nach dem Einchecken und Aktualisieren der lokalen Session verloren. Externe Tools, die Änderungen an der Dateistruktur eines Multiuser-Projektes vornehmen, dürfen nur in der Serverprojekt-Ansicht eingesetzt werden.
- Das Ändern von anwenderdefinierter Dokumentation ist in der lokalen Session nicht möglich. Änderungen an vorhandener anwenderdefinierter Dokumentation gehen nach dem Einchecken und Aktualisieren der lokalen Session verloren. Das Ändern von anwenderdefinierter Dokumentation ist nur in der Serverprojekt-Ansicht möglich.

Hinweis

Umbenennen von Objekten in lokalen Sessions

Objekte sollten ausschließlich in der Serverprojekt-Ansicht umbenannt werden, da dies **keine** Inkonsistenzen im Serverprojekt verursacht.

Beim Umbenennen von Objekten in der lokalen Session müssen Sie dafür sorgen, dass alle von der Umbenennung betroffenen Objekte und deren Verwendungsstellen ebenfalls markiert und in das Serverprojekt eingecheckt werden. Ansonsten erhalten Sie durch die unvollständige Umbenennung verursachte Übersetzungsfehler und ein inkonsistentes Serverprojekt.

Vor dem Einchecken der Verwendungsstellen des umbenannten Objekts sollten Sie darauf achten, beim Markieren keine Konflikte (rote Fähnchen) zu erzeugen und auch keine veralteten Objekte zu markieren, da Sie sonst beim Einchecken die Arbeit eines anderen Benutzers überschreiben.

Hinweis

Bearbeiten von Bibliotheksobjekten in lokalen Sessions

Bibliotheksobjekte sollten ausschließlich in der Serverprojekt-Ansicht bearbeitet werden, da dies **keine** Inkonsistenzen im Serverprojekt verursacht.

Bei Versionsänderungen an Bibliotheksobjekten in der lokalen Session, z. B. beim Anlegen neuer Versionen von Typ-Instanz-Objekten, müssen Sie dafür sorgen, dass alle betroffenen Objekte mit einer Versionsänderung markiert wurden und in das Serverprojekt eingecheckt werden.

Ansonsten erhalten Sie durch das Anlegen neuer Versionen verursachte Übersetzungsfehler und ein inkonsistentes Serverprojekt.

Hinweis

Die genauen Vorgehensweisen zum Arbeiten mit lokalen Sessions finden Sie in den nachfolgenden Kapiteln beschrieben.

Siehe auch

Verwendungsinfo für markierte Objekte anzeigen (Seite 158)

Markierung für Objekte zurücknehmen (Seite 157)

Objekte ins Serverprojekt einchecken (Seite 162)

Mögliche Konflikte beim Einchecken (Seite 165)

Mögliche Fehlermeldungen beim Einchecken (Seite 168)

2.17.2.2 In der lokalen Session offline arbeiten

Einführung

Beim Öffnen einer lokalen Session wird automatisch die Verbindung zum zugehörigen Multiuser-Server aufgebaut. Bisher konnten Sie im Umfeld von Multiuser Engineering nur bei einer aktiven Serververbindung in der lokalen Session arbeiten.

Ab dem TIA Portal V15 haben Sie die Möglichkeit, auch ohne Serververbindung "offline" in einer lokalen Session zu arbeiten.

Kann beim Öffnen der lokalen Session keine Serververbindung aufgebaut werden, erscheint ein Dialog mit einer entsprechenden Fehlermeldung.

Sie werden zunächst gefragt, ob Sie die lokale Session als Single-User-Projekt exportieren möchten. Wenn Sie diese Option mit "OK" bestätigen, erscheint ein Folgedialog, der Sie darüber informiert, dass keine Verbindung zum Server aufgebaut werden konnte.

Wenn Sie diesem Dialog ebenfalls mit "OK" bestätigen, können Sie auch ohne Serververbindung in der lokalen Session arbeiten, mit den nachfolgend aufgeführten Einschränkungen:

- Markierungen von anderen Anwendern sind in Ihrer lokalen Offline-Session nicht sichtbar, es werden also keine gelben oder roten Fähnchen und auch keine veralteten Objekte angezeigt.
- Änderungen, die Sie in der lokalen Offline-Session vornehmen, können erst dann ins Serverprojekt eingecheckt werden, wenn wieder eine Serververbindung vorhanden ist.

Hinweis

Hinweise zum Öffnen von lokalen Sessions:

- Eine umbenannte oder gelöschte lokale Session kann nicht mehr geöffnet werden.
- Eine lokale Session kann nicht als Referenzprojekt geöffnet werden.

Beim Öffnen einer lokalen Session, die einem der o. g. Kriterien entspricht, erhalten Sie einen Hinweis, dass das Öffnen nicht möglich ist. In diesem Fall werden Sie gefragt, ob Sie diese lokale Session als Single-User-Projekt exportieren möchten.

Voraussetzung

Sie haben das TIA Portal ab V15 und einen Multiuser-Server installiert.

Sie haben ein ablauffähiges Multiuser-Serverprojekt und eine gültige lokale Session angelegt.

Lokale Session öffnen und offline bearbeiten

Um eine lokale Session zu öffnen, gehen Sie vor wie folgt:

1. Wählen Sie im TIA Portal den Befehl "Projekt > Öffnen".
   Der Dialog "Projekt öffnen" mit einer Liste der kürzlich verwendeten Projekte und Sessions wird angezeigt.
   Die in der ersten Spalte angezeigten Symbole dienen zur Unterscheidung von Single-User- und Multiuser-Projekten.
2. Wählen Sie die gewünschte lokale Session aus und klicken Sie auf die Schaltfläche "Öffnen".
3. Wenn keine Serververbindung zum zugehörigen Multiuser-Server aufgebaut werden kann, erhalten Sie eine entsprechende Meldung. Sie werden gefragt, ob Sie die lokale Session als Single-User-Projekt exportieren möchten.
4. Klicken Sie auf "OK".
5. Es erscheint ein zweiter Dialog, der aussagt, dass keine Serververbindung aufgebaut werden konnte. Wenn Sie diesen Dialog mit "OK" bestätigen, können Sie mit den genannten Einschränkungen auch ohne Serververbingung in Ihrer lokalen Session weiterarbeiten.

Ergebnis

Die ausgewählte lokale Session wird offline geöffnet.

Sie sehen keine Markierungen von anderen Anwendern.

Sie können Ihre Änderungen erst einchecken, wenn wieder eine Serververbindung vorhanden ist.

2.17.2.3 Regeln für das Markieren von Objekten

Einführung

Damit Objekte im Rahmen von Multiuser Engineering parallel bearbeitet werden können, müssen die zu bearbeitenden Objekte in den lokalen Session markiert werden.

Es gibt zwei gleichwertige Möglichkeiten für das Markieren von Objekten:

- Ab dem TIA Portal V15 übernimmt die Funktion "Automarkieren" das automatische Markieren der zur Bearbeitung zugelassenen Objekte. Das bedeutet, dass jedes in der lokalen Session zulässige "markierbare" Objekt, das Sie erstellen, bearbeiten oder löschen, automatisch markiert wird.
- Daneben können Sie Objekte innerhalb der lokalen Session auch manuell markieren:
  - Über den Kontextmenübefehl "Objekte für das Einchecken markieren".
  - Durch Anklicken der angezeigten grauen Fähnchen in der Projektnavigation und in den einzelnen Editoren des TIA Portals.

  Für das manuelle Markieren von Objekten sind auch Mehrfachselektionen über das Kontextmenü möglich. Voraussetzung für das Markieren ist, dass zumindest ein markierbares Objekt in der Selektion enthalten ist. Falls kein markierbares Objekt in der Mehrfachselektion enthalten ist, sind die Kontextmenübefehle eventuell nicht aktiv.

Es gibt einige Bearbeitungsschritte (z. B. Änderungen an der HW-Konfiguration), die nur in der Serverprojekt-Ansicht durchgeführt werden dürfen.

Regeln für das Markieren von Objekten

Folgende Regeln gelten für das Markieren:

- Die Markierungen und die Markierungsspalte in der Projektnavigation und in den einzelnen Editoren werden nur im Kontext von Multiuser Engineering angezeigt.
- Markierbare Objekte werden durch die Anzeige eines grauen Fähnchens in der Projektnavigation und den Editoren gekennzeichnet.
- Es können nur solche Objekte markiert werden, die mit Multiuser Engineering bearbeitet werden können.
- Hat ein Objekt **kein** vorangestelltes graues Fähnchen, ist dieses Objekt nicht markierbar und sollte in der lokalen Session nicht bearbeitet werden, da die Änderungen nicht ins Serverprojekt eingecheckt werden können. Bearbeiten Sie solche nicht markierbaren Objekte in der Serverprojekt-Ansicht.
- Bestimmte übergeordnete Verzeichnisse können selbst nicht markiert werden, aber die einzelnen Objekte in diesem Ordner sind markierbar.
  Beispiel: "PLC-Variablentabelle": Die PLC-Variablentabelle selbst ist nicht markierbar, die einzelnen Variablen in der geöffneten Variablentabelle können jedoch markiert werden.

- Das automatische Markieren wird vom System beim Bearbeiten eines markierbaren Objektes durchgeführt.
- Das manuelle Markieren wird durch Anklicken des grauen Fähnchens oder über das Kontextmenü durchgeführt. Das Fähnchen wird farbig dargestellt, sobald ein Objekt markiert ist.
- Die Farben der Fähnchen geben zusätzliche Informationen zur Markierung:

Symbol	Bedeutung
	Objekt kann zum Einchecken markiert werden.
	Objekt ist in der eigenen lokalen Session markiert.
	Objekt ist in einer anderen lokalen Session markiert, die zum gleichen Multiuser-Serverprojekt gehört.
	Konflikt: Das Objekt wurde in mehreren lokalen Sessions gleichzeitig markiert.

Besonderheiten beim automatischen Markieren von Objekten

Folgende Besonderheiten sind beim Automarkieren zu beachten:

Beim Arbeiten in den verschiedenen Editoren des TIA Portals werden Änderungen teilweise auch in unter- oder übergeordneten Ordnern Objekte automatisch markiert. Nachfolgend sind einige dieser Aktionen beispielhaft aufgelistet.

- Das Automarkieren von Objekten im Umfeld von Multiuser Engineering funktioniert sowohl in den geöffneten Editoren als auch beim Arbeiten in der Projektnavigation.
  - Wird ein Objekt in der Projektnavigation umbenannt, so wird das neue, umbenannte Objekt automatisch markiert.
  - Wird ein Objekt in der Projektnavigation innerhalb des selben Gerätes in einen anderen Ordner verschoben, so wird das verschobene Objekt automatisch markiert.
  - Wird jedoch ein nicht markierbarer Ordner (z. B. die PLC-Variablentabelle) in der Projektnavigation an einen anderen Ort verschoben, so werden die unterlagerten Objekte **nicht** automatisch markiert.
- Beim Arbeiten in der Projektbibliothek werden alle neu erstellten Objekte automatisch markiert. Wird allerdings ein kompletter Bibliotheks-Ordner in der Projektnaviation an einen anderen Ort verschoben, so werden die darin enthaltenen Typen **nicht** automatisch markiert.
- Beim Arbeiten mit Projekttexten werden die markierbaren Objekte, zu denen die Projekttexte gehören, automatisch markiert.
  - Beispiel: Bei Änderungen an einem Baustein-Kommentar wird der zugehörige Baustein automatisch markiert, damit die vorgenommenen Änderungen beim nächsten Einchecken mit ins Serverprojekt übernommen werden.

- Beim Arbeiten mit PLC-Meldetextlisten werden alle Änderungen an Texten automatisch markiert.
  - Wird ein neuer Eintrag in einer Textliste erstellt, so wird die zugehörige Textliste automatisch markiert.
  - Wird ein bestehender Text in einer Textliste bearbeitet, so wird die zugehörige Textliste automatisch markiert.
- Beim Bearbeiten von PLC-Überwachungen & -Meldungen werden alle Änderungen automatisch markiert.
  - Wird ein neuer Eintrag hinzugefügt oder werden bestehende Einträge verändert, so wird sowohl der geänderte Eintrag als auch der zugehörige Baustein automatisch markiert.
- Beim Bearbeiten von HMI-Text- und Grafiklisten werden alle Änderungen automatisch markiert. Die Markierungen werden allerdings nur im übergeordneten Editor angezeigt.
- Beim Bearbeiten von HMI-Variablen sind nur die übergeordneten Variablen markierbar.
  - Bei Änderungen an den unterlagerten Variablen werden automatisch die zugehörigen übergeordneten Variablen markiert.
  - Bei Änderungen im Alarm-Editor werden die zu diesen Variablen zugehörigen Alarme automarkiert und nicht die Variablen selbst.
- Beim Bearbeiten von HMI-Screens und HMI-Scripts sind einige Besonderheiten bezüglich des Automarkierens zu beachten, da nicht alle dort durchgeführten Aktionen zu einem Automarkieren von Objekten führen.
  - Bei Änderungen im Scripteditor wird das Automarkieren nur dann durchgeführt, wenn die ausgeführte Aktion über "Undo" rückgängig gemacht werden kann.
  - Bei Änderungen an einzelnen Elementen eines Sreens wird der zugehörige Screen automarkiert.
  - Bei Änderungen in der globalen Filterkonfiguration in den Eigenschaften eines Screens werden diese **nicht** automarkiert, da die diese Konfiguration nicht zu einem Screen zugehörig ist.
- Beim Bearbeiten von Trace - und Logikanalysatorfunktion gibt es folgende Aktionen, die **nicht** automarkiert werden:
  - Das Löschen des Beispiel-OBs (OB30) aus dem Programmierordner.
  - Das Ändern der Zykluszeit im OB 30.

2.17.2.4 Objekte für die Bearbeitung automatisch oder manuell markieren

Einführung

Damit Objekte im Rahmen von Multiuser Engineering parallel bearbeitet werden können, müssen die zu bearbeitenden Objekte in den lokalen Session markiert sein.

Es gibt zwei gleichwertige Möglichkeiten für das Markieren von Objekten:

- Die Funktion "Automarkieren", die die in der lokalen Session bearbeiteten Objekte automatisch markiert.
  Sobald Sie ein Objekt in der lokalen Session in einem Editor des TIA Portals öffnen, kopieren, einfügen oder löschen, wird dieses Objekt automatisch für das Einchecken markiert.
  Dies gilt auch für Objekte, die während der Bearbeitung durch den Anwender unterlagert vom System erzeugt werden (z. B. Systembausteine usw.).
- Das manuelle Markieren von Objekten ist nach wie vor möglich:
  - Über den Kontextmenübefehl "Objekte für das Einchecken markieren".
  - Durch Anklicken der angezeigten grauen Fähnchen in der Projektnavigation und in den einzelnen Editoren des TIA Portals.

Hinweis

Änderungen an nicht markierbaren Objekten

Änderungen an nicht markierbaren Objekten - das sind Objekte, die kein graues Fähnchen haben - sind in der lokalen Session zwar möglich, werden jedoch **nicht** ins Serverprojekt übernommen.

Diese Änderungen gehen nach einem Aktualisieren der lokalen Session verloren, da sie wieder mit dem Inhalt des Serverprojekts überschrieben werden!

Sobald Sie ein nicht markierbares Objekt bearbeiten wollen, wird im geöffneten Editor ein Bannertext angezeigt. In diesem Bannertext erhalten Sie Informationen zur weiteren Vorgehensweise.

Vorgehensweise zum automatischen Markieren von Objekten

Um Objekte automatische in einer lokalen Session zu markieren, gehen Sie vor wie folgt:

1. Öffnen Sie die gewünschte lokale Session.
2. Bearbeiten Sie gewünschte Objekt, z. B. indem Sie es in einem Editor öffnen, kopieren und einfügen oder löschen.

Ergebnis

Das Objekt wird beim Bearbeiten und auch beim Löschen automatisch markiert und das entsprechende blaue oder rote Fähnchen wird in der eigenen lokalen Session gesetzt.

Wird ein blaues Fähnchen angezeigt, kann das Objekt in der lokalen Session bearbeitet werden.

Wird ein rotes Fähnchen angezeigt, liegt ein Konflikt vor, der durch Absprache zwischen den Bearbeitern gelöst werden muss.

Falls ein Automarkieren-Konflikt beim Kopieren und Einfügen von Objekten auftritt, erhalten Sie eine Fehlermeldung. Teilweise wird die weitere Vorgehensweise über einen Dialog gesteuert, in dem Sie die wünschte Option auswählen können.

Vorgehensweise zum manuellen Markieren von Objekten

Um Objekte in einer lokalen Session zu markieren, gehen Sie vor wie folgt:

1. Öffnen Sie die gewünschte lokale Session und klicken Sie auf das angezeigte graue Fähnchen, um ein Objekt für das Einchecken zu markieren.
2. Alternativ können Sie auch das gewünschte Objekt selektieren und über den Kontextmenübefehl "Objekt für das Einchecken markieren" die Markierung vornehmen.
3. Markieren Sie alle Objekte in Ihrer lokalen Session, die Sie aktuell bearbeiten möchten. Multiselektionen für das Markieren sind möglich. Nach dem Markieren können Sie die Objekte wie gewohnt bearbeiten und speichern.

Hinweis

Multiselektionen

Sobald in der lokalen Session >=50 Objekte angewählt werden, sind beide Kontextmenübefehle gleichzeitig aktiv, sowohl der Befehl für das Markieren der Objekte als auch der Befehl für das Zurücknehmen der Markierungen.

Ergebnis

Die markierten Objekte werden mit einem blauen oder roten Fähnchen angezeigt.

Wird ein blaues Fähnchen angezeigt, kann das Objekt in der lokalen Session bearbeitet werden.

Wird ein rotes Fähnchen angezeigt, liegt ein Konflikt vor, der durch Absprache zwischen den Bearbeitern gelöst werden muss.

Hinweis

Markierungen mit Konflikt (rotes Fähnchen)

Ein Markierungskonflikt muss durch Absprache zwischen den Bearbeitern geklärt werden und eine Reihenfolge für das Bearbeiten der markierten Objekte festgelegt werden. Um den Konflikt zu lösen, muss ein Bearbeiter seine Markierung zurücknehmen.

Siehe auch

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

Markierung für Objekte zurücknehmen (Seite 157)

Verwendungsinfo für markierte Objekte anzeigen (Seite 158)

2.17.2.5 Mögliche Konflikte beim Automarkieren

Einführung

Damit Objekte im Rahmen von Multiuser Engineering parallel bearbeitet werden können, müssen die zu bearbeitenden Objekte in den lokalen Session markiert sein. Beim Automarkieren von Objekten in der lokalen Session können folgende mögliche Konflikte auftreten:

- Ein automatisch markiertes Objekt kann "veraltet" sein.
  In diesem Fall muss das Objekt unbedingt vor dem Einchecken aktualisiert werden, damit die auf dem Multiuser-Server vorhandenen Änderungen übernommen werden können.
- Ein automatisch markiertes Objekt kann durch andere Anwender bereits markiert gewesen sein. In diesem Fall wird beim Automarkieren ein rotes Fähnchen angezeigt und der Konflikt muss durch Absprache zwischen den Bearbeitern gelöst werden.

Hinweis

Markierungen mit Konflikt (rotes Fähnchen)

Ein Markierungskonflikt muss durch Absprache zwischen den Bearbeitern geklärt werden und eine Reihenfolge für das Bearbeiten der markierten Objekte festgelegt werden. Um den Konflikt zu lösen, muss ein Bearbeiter seine Markierung zurücknehmen.

Anzeige von Konflikten beim Automarkieren

Falls ein Automarkieren-Konflikt auftritt, erhalten Sie im Inspektorfenster im Register "Info" eine Fehlermeldung.

Diese Fehlermeldung informiert Sie darüber, bei welcher Aktion der Konflikt aufgetreten ist und um welche Art von Konflikt es sich handelt.

Teilweise wird die weitere Vorgehensweise über einen Dialog gesteuert, in dem Sie die gewünschte Option auswählen können.

Beispiel für eine Fehlermeldung beim Automarkieren:

Die Operation "Neues Netzwerk einfügen" führt zu einem Automark-Konflikt, da eines der betroffenen automarkierten Objekte veraltet ist.

Beheben von Konflikten beim Automarkieren

Alle beim Automarkieren möglicherweise auftretende Konflikte müssen vor dem Einchecken unbedingt behoben werden, da ansonsten die Gefahr von Datenverlust besteht. Beim Einchecken von veralteten Objekten und von "mit Konflikt markierten" Objekten gehen Änderungen verloren oder werden überschrieben. Gehen Sie vor wie folgt:

1. Wird ein Objekt beim Automarkieren als "veraltet" gekennzeichnet, so müssen Sie dieses Objekt in der lokalen Session vor dem Einchecken und bevor Sie es weiter bearbeiten aktualisieren, damit die auf dem Multiuser-Server vorhandenen Änderungen übernommen werden können.
2. Wird ein Objekt beim Automarkieren mit einem roten Fähnchen markiert, so wird dieses Objekt bereits durch einen anderen Bearbeiter in einer anderen lokalen Session bearbeitet. Lösen Sie diesen Konflikt, indem ein Bearbeiter seine Markierung manuell zurücknimmt.

2.17.2.6 Markierung für Objekte zurücknehmen

Einführung

Damit Objekte im Rahmen von Multiuser Engineering bearbeitet werden können, müssen diese Objekte in den lokalen Sessions entweder automatisch oder manuell markiert werden. Die gesetzten Markierungen für das Einchecken können auch wieder zurückgenommen werden.

Dies ist vor allem in einem Konfliktfall (angezeigt durch ein rotes Fähnchen) notwendig, wenn mehrere Bearbeiter in ihrer jeweiligen lokalen Session ein identisches Objekt zum Einchecken markiert haben.

Hinweis

Markierungen mit Konflikt (rotes Fähnchen)

Ein Markierungskonflikt muss durch Absprache zwischen den Bearbeitern geklärt werden und eine Reihenfolge für das Bearbeiten der markierten Objekte festgelegt werden. Um den Konflikt zu lösen, muss ein Bearbeiter seine Markierung zurücknehmen.

Vorgehensweise

Um die Markierung von Objekten in einer lokalen Session zurückzunehmen, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session und klicken Sie auf das angezeigte blaue oder rote Fähnchen, um die Markierung für ein Objekt zurücknehmen.
2. Alternativ können Sie auch das gewünschte Objekt selektieren und die Markierung zurücknehmen über den Kontextmenübefehl "Markierung zurücknehmen".

Ergebnis

Die nicht mehr markierten Objekte und die als Konflikt markierten Objekte werden wie folgt dargestellt:

Symbol vorher	Aktion	Symbol nachher
	Markierung zurücknehmen	
	Markierung zurücknehmen	

Siehe auch

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

Objekte für die Bearbeitung automatisch oder manuell markieren (Seite 153)

Verwendungsinfo für markierte Objekte anzeigen (Seite 158)

2.17.2.7 Verwendungsinfo für markierte Objekte anzeigen

Einführung

Damit Objekte im Rahmen von Multiuser Engineering parallel bearbeitet werden können, muss jeder Bearbeiter in seiner lokalen Session die Objekte markieren, die er bearbeiten möchte.

Um das mehrfache Markieren von Objekten und dadurch entstehende Markierungskonflikte zu vermeiden, erhalten Sie folgende Informationen zu markierten Objekten:

- Benutzer: zeigt an, wer das Objekt markiert hat
- Rechner: zeigt an, auf welchem Rechner das Objekt markiert wurde.
- Ablageort: zeigt an, wo das Objekt markiert wurde, in einer lokalen Session oder in der Serverprojekt-Ansicht.

Diese Informationen erleichtern es Ihnen, die weitere Vorgehensweise für markierte oder mit Konflikt markierte Objekte mit den anderen Benutzern abzusprechen.

Verwendungsinfo anzeigen

Um die Verwendungsinfo für markierte Objekte anzuzeigen, gehen Sie vor wie folgt:

1. Selektieren Sie ein gelb oder rot markiertes Objekt in ihrer lokalen Session oder in der Serverprojekt-Ansicht.
2. Rufen Sie über das Kontextmenü den Befehl "Verwendungsinfo" auf.

Ergebnis

Der Dialog "Verwendungsinfo" wird geöffnet und zeigt die o.g. Informationen zum markierten Objekt an.

Hinweis

Markierungen mit Konflikt (rotes Fähnchen)

Ein Markierungskonflikt muss durch Absprache zwischen den Bearbeitern geklärt werden und eine Reihenfolge für das Bearbeiten der markierten Objekte festgelegt werden. Um den Konflikt zu lösen, muss ein Bearbeiter seine Markierung zurücknehmen.

Siehe auch

Multiuser-Markierungen in der Bedienoberfläche (Seite 41)

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

Objekte für die Bearbeitung automatisch oder manuell markieren (Seite 153)

Markierung für Objekte zurücknehmen (Seite 157)

2.17.2.8 Lokale Session aktualisieren

Einführung

Damit Objekte im Rahmen von Multiuser Engineering parallel bearbeitet werden können,müssen die in der lokalen Session bearbeiteten Objekte automatisch oder manuell markiert sein. Wenn Objekte im Serverprojekt einen neueren Stand haben als in der lokalen Session, werden diese Objekte als "veraltet" gekennzeichnet.

Diese Objekte können in der lokalen Session aktualisiert werden, indem Sie auf das Symbol "Lokale Session aktualisieren" in der Multiuser-Funktionsleiste klicken.

Möglicherweise haben andere Benutzer neue Objekte im Serverprojekt angelegt oder Objekte gelöscht, die in Ihrer lokalen Session momentan noch nicht aktuell angezeigt werden.

Diese neuen Objekte werden nach einem Aktualisieren ebenfalls in Ihrer lokalen Session angezeigt.

Beim Aktualisieren von "veralteten" Objekten werden die Inhalte in der lokalen Session mit den aktuelleren Inhalten des Serverprojekts überschrieben.

Hinweis

Lokale Session aktualisieren

- Die Schaltfläche zum Aktualisieren der lokalen Session öffnet die Ansicht "Aktualisieren", auch wenn es in Ihrer lokalen Session keine zu aktualisierenden Objekte gibt. Wenn Sie auf die Schaltfläche "Aktualisieren starten" klicken, wird die lokale Session aktualisiert.
- Beim Aktualisieren werden in Ihrer lokalen Session alle Änderungen an **nicht markierten Objekten** mit den Inhalten des Serverprojekts überschrieben. Die markierten Objekte bleiben unverändert erhalten.
- Das Aktualisieren der lokalen Session können Sie bei Bedarf wieder rückgängig machen, indem Sie auf die Schaltfläche "Wiederherstellen" klicken. Danach wird Ihre lokale Session mit den Inhalten vor der Aktualisierung wiederhergestellt.

Vorgehensweise zum Aktualisieren

Um Objekte in einer lokalen Session zu aktualisieren, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session, die zu aktualisierende Objekte enthält.
2. Klicken Sie auf die Schaltfläche "Lokale Session aktualisieren" in der Multiuser-Funktionsleiste.
   Daraufhin öffnet sich der Multiuser-Editor in der Aktualisierungs-Ansicht und zeigt alle in Ihrer lokalen Session markierten Objekte an. Die von Ihnen markierten Objekte werden beim Aktualisieren nicht überschrieben. Lediglich nicht markierte Objekte werden mit den Inhalten des Serverprojekts synchronisiert.
3. Klicken Sie auf die Schaltfläche "Aktualisieren starten", um das Aktualisieren durchzuführen.
   Sie werden über den Fortschritt der Aktualisierung informiert. Falls Fehler auftreten, erhalten Sie eine Fehlermeldung.
4. Sie erhalten die Meldung, dass das Aktualisieren erfolgreich beendet wurde. Klicken Sie auf "Ok".

Ergebnis

Alle nicht markierten Objekte in Ihrer lokalen Session wurden aktualisiert.

Die in Ihrer lokalen Session markierten Objekte wurden **nicht** mit dem Serverstand synchronisiert.

Hinweis

Online-Offline-Unterschiede nach dem Aktualisieren

Nach einem Laden ins Gerät werden Online-Informationen zu den geladenen Objekten in der lokalen Session im TIA Portal gespeichert. Dies ermöglicht einen Vergleich zwischen der Online- und Offline-Version. Da diese Online-Informationen aber nicht auf den Server eingecheckt werden, stehen diese Informationen in der lokalen Session nach einem Aktualisieren nicht mehr zur Verfügung. Arbeitet man nach dem Aktualisieren dann Online auf dem Gerät, kann das TIA Portal aufgrund der fehlenden Informationen keinen Vergleich durchführen. Die Objekte werden mit Online-Offline-Unterschieden angezeigt, obwohl die Inhalte der Objekte identisch sind.

Um dies zu beheben, sollte nach einem Aktualisieren der lokalen Session ein erneutes und vollständiges Laden ins Gerät durchgeführt werden.

2.17.2.9 Lokale Session nach dem Aktualisieren wieder herstellen

Einführung

In der lokalen Session können nicht markierte Objekte, die im Serverprojekt einen neueren Stand haben, aktualisiert werden. Durch das Aktualisieren werden die nicht markierten Objekte in der lokalen Session mit den aktuelleren Inhalten des Serverprojekts überschrieben.

Das Aktualisieren der lokalen Session können Sie bei Bedarf wieder rückgängig machen. Klicken Sie dazu nach dem Aktualisieren im Multiuser-Editor auf die Schaltfläche "Wiederherstellen". Danach wird Ihre lokale Session mit den Inhalten vor der Aktualisierung wiederhergestellt.

Hinweis

Schaltfläche "Wiederherstellen"

Die Schaltfläche "Wiederherstellen" ist nur nach dem Durchführen einer Aktualisierung aktiv, so lange ein Wiederherstellen der Inhalte der lokalen Session mit dem letzten Stand möglich ist.

Die Schaltfläche "Wiederherstellen" ist inaktiv, wenn kein Wiederherstellen der lokalen Session durchgeführt werden kann, z. B. weil keine Aktualisierung durchgeführt wurde.

Das Schließen und erneute Öffnen des Editors hat keine Auswirkungen auf die Schaltfläche "Wiederherstellen". Falls eine Aktualisierung vorliegt, die rückgängig gemacht werden kann, ist die Schaltfläche auch nach dem Schließen und erneuten Öffnen des Editors aktiv.

Vorgehensweise zum Wiederherstellen der lokalen Session

Um die lokale Session nach dem Aktualisieren wieder herzustellen, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session, für die Sie eine Aktualisierung durchgeführt haben, falls Sie diese zwischenzeitlich geschlossen haben.
   Falls Sie die Aktualisierungs-Ansicht noch geöffnet haben, starten Sie mit Schritt 3.
2. Klicken Sie auf die Schaltfläche "Lokale Session aktualisieren" in der Multiuser-Funktionsleiste, um den Multiuser-Editor in der Aktualisierungs-Ansicht zu öffnen.
3. Klicken Sie auf die Schaltfläche "Wiederherstellen", um den Stand der lokalen Session vor dem letzten Aktualisieren wieder herzustellen.
4. Klicken Sie im nachfolgenden Dialog bei der Abfrage auf "Ja".
5. Beim Auftreten von Fehlern erhalten Sie eine Fehlermeldung, die Sie bestätigen müssen.
6. Bei erfolgreicher Wiederherstellung erhalten Sie eine Meldung. Klicken Sie auf "Ok".

Ergebnis

Ihre lokale Session wurde mit dem Stand vor der letzten Aktualisierung wieder hergestellt.

Siehe auch

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

2.17.2.10 Objekte ins Serverprojekt einchecken

Einführung

Nach dem Bearbeiten von markierten Objekten in der lokalen Session können die Änderungen ins Serverprojekt eingecheckt werden.

Für das Einchecken gelten folgende Regeln:

- Beim Einchecken ins Serverprojekt werden nur die Änderungen an markierten Objekten übernommen.
- Falls Sie nicht markierbare Objekte in der lokalen Session bearbeitet haben, werden diese Änderungen beim Einchecken **nicht** übernommen.
- Beim Aktualisieren Ihrer lokalen Session nach dem Einchecken werden Änderungen an nicht markierbaren Objekten mit den Inhalten des Serverprojekts überschrieben.
- Falls beim Einchecken Fehler auftreten, erhalten Sie eine Fehlermeldung mit weiteren Informationen. Beheben Sie die aufgetretenen Fehler wie vorgegeben entweder in der lokalen Session oder in der Serverprojekt-Ansicht. Starten Sie danach das Einchecken erneut.
- Beim Einchecken und beim Bearbeiten von Objekten in der Serverprojekt-Ansicht arbeitet der jeweilige Benutzer exklusiv auf dem Multiuser-Server. Deshalb wird der Server für alle anderen Benutzer gesperrt. Der Serverstatus wird während der exklusiven Bearbeitung in der Funktionsleiste als "nicht verfügbar" gekennzeichnet. Das Anlegen einer neuen lokalen Session und das Aktualisieren von lokalen Sessions ist trotzdem möglich.

Optionen für das Einchecken

Übersetzen

Sie haben die Möglichkeit, die in der lokalen Session vorgenommenen Änderungen vor dem Einchecken zu übersetzen.

Hinweis

Beim Einchecken angezeigte Optionen für das Übersetzen

Die beim Einchecken angezeigten Optionen für das Übersetzen sind abhängig von den im TIA Portal unter "Extras > Einstellungen > Multiuser" bei "Übersetzungseinstellungen" vorgenommenen Einstellungen.

Aufgrund dieser Einstellungen können die Übersetzungsoptionen im Multiuser-Editor eventuell deaktiviert sein.

Übersetzungsart

Die Übersetzungsart legt fest, in welchem Umfang die Software-Änderungen übersetzt werden. Sie können folgende Optionen aus der Klappliste auswählen:

- Markierte Objekte (nur Änderungen)
  Es werden die Änderungen an den markierten Objekten für das ausgewählte Gerät zusammen mit den gerätelokalen Objekten übersetzt.
- Gerätesoftware (nur Änderungen)
  Es werden die Änderungen an den markierten Objekten zusammen mit dem auf den zugeordneten Geräten vorhandenen Softwarestand übersetzt.
  Beispiel: Sie haben mehrere Objekte markiert und geändert, die zu mehr als einem Gerät (z. B. zu CPU_1 und zu CPU_2) gehören. Beim Übersetzen werden diese markierten Objekte zusammen mit den Softwareständen der beiden CPUs übersetzt.

Serverprojekt-Ansicht anzeigen

Durch Aktivieren dieser Option haben Sie die Möglichkeit, sich Ihre Änderungen an den markierten Objekten vor dem Einchecken nochmal in der Serverprojekt-Ansicht zusammen mit den aktuellen Inhalten des Serverprojekts anzeigen zu lassen.

In der Serverprojekt-Ansicht können Sie alle Objekte markieren und bearbeiten, sowie vorhandene Markierungen zurücknehmen. Danach können Sie entweder Ihre Änderungen im Serverprojekt speichern oder Ihre Änderungen verwerfen und die Serverprojekt-Ansicht ohne weitere Aktion verlassen.

Voraussetzung

Die Verbindung zum Multiuser-Server ist vorhanden und der Server ist verfügbar.

Sie haben die in der lokalen Session bearbeiteten Objekte für das Einchecken markiert.

Vorgehensweise beim Einchecken von Änderungen

Um die Änderungen aus der lokalen Session ins Serverprojekt einzuchecken, gehen Sie vor wie folgt:

1. Öffnen Sie die lokale Session, die Sie einchecken möchten.
2. Klicken Sie auf die Schaltfläche "Einchecken" in der Multiuser-Funktionsleiste.
   Daraufhin öffnet sich der Multiuser-Editor in der Ansicht "Einchecken" und zeigt alle in der lokalen Session markierten Objekte an, die beim Einchecken ins Serverprojekt übernommen werden.
3. Klappen Sie die angezeigten Ordner auf und überprüfen Sie die angezeigten und für das Einchecken markierten Objekte.
4. Klicken Sie in der Funktionsleiste auf die Schaltfläche mit dem roten Fähnchen, um sich eventuelle Konflikte anzeigen zu lassen.
   Vorhandene Konflikte werden angezeigt. Beheben Sie die angezeigten Konflikte vor dem Einchecken, um Datenverlust oder ungewolltes Überschreiben zu vermeiden.
   Falls keine Konflikte vorhanden sind, klicken Sie erneut auf diese Schaltfläche, um zur vorherigen Ansicht "Einchecken" zurück zu schalten.
5. Wählen Sie die gewünschten Optionen für das Einchecken aus.
   - "Übersetzen": Wählen Sie aus der Klappliste die gewünschte Einstellung aus.
   - "Serverprojekt-Ansicht öffnen": Bei Anklicken dieser Option wird vor dem Einchecken die Serverprojekt-Ansicht in der Projektnavigation geöffnet.
6. Tragen Sie einen Kommentar zur Änderungshistorie ein, um die von Ihnen iin der lokalen Session vorgenommenen Änderungen zu dokumentieren.
7. Klicken Sie nach der Überprüfung auf die Schaltfläche "Einchecken starten", um die angezeigten Objekte in das Multiuser-Serverprojekt einzuchecken.
   - Falls Sie die Serverprojekt-Ansicht geöffnet haben, können Sie Ihre Änderungen über die Schaltfläche "Änderungen speichern" einchecken. Wenn Sie auf die Schaltfläche "Änderungen verwerfen" klicken, werden Ihre Änderungen verworfen und nicht ins Serverprojekt übernommen.
   - Falls Sie die Markierungen in der lokalen Session auch nach dem Einchecken behalten möchten, müssen Sie die Option "Markierungen für Objekte behalten" aktivieren.
   - Falls Sie nur Teile Ihrer Änderungen markiert und eingecheckt haben, sollten Sie **kein Aktualisieren** durchführen, um Ihre Änderungen an den nicht markierten Objekten zu erhalten.
8. Wenn das Einchecken erfolgreich war, erhalten Sie abschließend eine Meldung. Wenn Sie diese Meldung mit "OK" bestätigen, wird ihe lokale Session mit dem Inhalt des Serverprojekts überschrieben. Falls Sie Ihre Änderungen an den markierten Objekten behalten möchten, klicken Sie in der Meldung auf die Schaltfläche "Lokale Session erhalten".
9. Falls Sie Ihre Markierungen in der lokalen Session behalten möchten, aktivieren Sie das Kontrollkästchen "Markierungen erhalten".
10. Wenn das Einchecken nicht erfolgreich war und wegen Fehlern abgebrochen wurde, erhalten Sie eine entsprechende Fehlermeldung. Beheben Sie die genannten Fehler in Ihrer lokalen Session oder in der Serverprojekt-Ansicht. Starten Sie danach das Einchecken erneut.

Ergebnis

Sie erhalten eine Meldung, dass das Einchecken erfolgreich beendet wurde.

Ihre lokale Session wird nach dem erfolgreichen Einchecken mit den Inhalten des Serverprojekts aktualisiert und geöffnet.

- Die lokale Session enthält nach der Aktualisierung alle Änderungen, die zwischenzeitlich im Serverprojekt eingecheckt wurden.
- Falls neue Objekte im Serverprojekt hinzugefügt wurden, sind diese Objekte nun auch in Ihrer lokalen Session sichtbar.
- Falls Objekte im Serverprojekt gelöscht wurden, sind diese Objekte nun in Ihrer lokalen Session nicht mehr vorhanden.
- Ihre Markierungen in der lokalen Session werden automatisch gelöscht, falls Sie die Option "Markierung für Objekte behalten" nicht aktiviert hatten.
- Die Serververbindung wird nach dem Einchecken wieder frei gegeben. Der Serverstatus wechselt in der Anzeige innerhalb der Funktionsleiste von "belegt" auf "verfügbar".

Siehe auch

Mögliche Konflikte beim Einchecken (Seite 165)

Mögliche Fehlermeldungen beim Einchecken (Seite 168)

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

Einführung zum Arbeiten in der Serverprojekt-Ansicht (Seite 173)

Objekte in der Serverprojekt-Ansicht bearbeiten (Seite 177)

2.17.2.11 Mögliche Konflikte beim Einchecken

Einführung

Beim Einchecken der Änderungen aus der lokalen Session in das zugehörige Multiuser-Serverprojekt können Fehler auftreten, die ein erfolgreiches Einchecken verhindern.

Sie haben die Möglichkeit, sich vorhandene Fehler von dem Einchecken im Multiuser-Editor anzeigen zu lassen.

Klicken Sie dazu im Multiuser-Editor in der Kopfzeile auf das rote Fähnchen und alle in der lokalen Session vorhandenen Konflikte werden Ihnen angezeigt. Sind keine Konflikte vorhanden, wird eine entsprechende Nachricht angezeigt.

Falls Sie trotz bestehenden Konflikten Ihre Änderungen aus der lokalen Session ins Serverprojekt einchecken möchten, müssen Sie dies explizit in einem Dialog bestätigen, da diese Vorgehensweise möglicherweise zu Datenverlust führen kann.

Die nachfolgend genannten Fehlerkonstellationen können durch entsprechende Absprachen zwischen den Bearbeitern und durch Beachtung der empfohlenen Vorgehensweisen leicht vermieden werden.

Mögliche Namenskonflikte beim Einchecken

Namenskonflikte beim Einchecken treten immer dann auf, wenn in unterschiedlichen lokalen Sessions identische Namen für unterschiedliche Objekte verwendet werden.

Beispiel:

- Bearbeiter 1 legt in seiner lokalen Session die Variable "Motor_1" an.
- Bearbeiter 2 legt in seiner lokalen Session ebenfalls eine Variable "Motor_1" an.

Ergebnis:

Bearbeiter 1 kann seine Änderungen ohne Fehler einchecken.

Bearbeiter 2 kann seine Änderungen **nicht** einchecken. Er bekommt eine Fehlermeldung wegen eines Namenskonflikts und das Einchecken wird abgebrochen. Der Fehler muss in der lokalen Session behoben werden.

Mögliche Konflikte beim Anlegen von Geräten in der lokalen Session

Werden in einer lokalen Session durch einen Bearbeiter neue Geräte mit markierten Objekten hinzugefügt, z. B. eine neue CPU mit Bausteinen angelegt, führt dies beim Einchecken zu einem Konflikt.

Beispiel:

- Bearbeiter 1 legt in seiner lokalen Session eine neue CPU mit zwei Funktionsbausteinen an.
- Bearbeiter 1 markiert die Bausteine ordnungsgemäß, bevor er diese bearbeitet.

Ergebnis:

Bearbeiter 1 bekommt beim Einchecken eine Fehlermeldung, da die lokal neu angelegte CPU im Serverprojekt nicht gefunden wird. Der Transfer von Geräten wird momentan bei Multiuser Engineering nicht unterstützt. In diesem Fall wird das Einchecken abgebrochen und die Serverprojekt-Ansicht wird geöffnet. Bearbeiter 1 kann seine neu angelegten Objekte direkt in das Serverprojekt kopieren. Die Bearbeitung geht also nicht verloren.

Mögliche Konflikte beim Umbenennen von Objekten in der lokalen Session

Objekte sollten ausschließlich in der Serverprojekt-Ansicht umbenannt werden, da dies **keine** Inkonsistenzen im Serverprojekt verursacht.

Beim Umbenennen von Objekten in der lokalen Session müssen Sie dafür sorgen, dass alle von der Umbenennung betroffenen Objekte und deren Verwendungsstellen ebenfalls markiert und in das Serverprojekt eingecheckt werden. Ansonsten erhalten Sie durch die unvollständige Umbenennung verursachte Übersetzungsfehler und ein inkonsistentes Serverprojekt.

Vor dem Einchecken der Verwendungsstellen des umbenannten Objekts sollten Sie darauf achten, beim Markieren keine Konflikte (rote Fähnchen) zu erzeugen und auch keine veralteten Objekte zu markieren, da Sie sonst beim Einchecken die Arbeit eines anderen Benutzers überschreiben.

Mögliche Konflikte bei Änderung des Gerätetyps in der lokalen Session

Wird der Typ eines Gerätes in einer lokalen Session geändert, wird also z. B. eine "CPU_1" vom Typ S7-15xx getauscht oder umbenannt, so werden die in der "alten" CPU markierten Objekte danach in der lokalen Session eventuell nicht mehr angezeigt.

Mögliche Konflikte beim Löschen von Geräten im Serverprojekt

Werden im Serverprojekt vorhandene Geräte mit den enthaltenen Objekten gelöscht, kann dies zu einem Transferfehler führen.

Beispiel:

- Bearbeiter 1 arbeitet in seiner lokalen Session an markierten Objekten, die zur "CPU_1" gehören.
- Bearbeiter 2 löscht in der Zwischenzeit die "CPU_1" in der Serverprojekt-Ansicht und speichert seine Änderungen im Serverprojekt.

Ergebnis:

Bearbeiter 1 bekommt beim Einchecken eine Fehlermeldung, da die "CPU_1" im Serverprojekt nicht mehr vorhanden ist. In diesem Fall kann Bearbeiter 1 die von ihm bearbeiteten Objekte in der Serverprojekt-Ansicht einfügen. Entweder kopiert er die "CPU_1" komplett ins Serverprojekt oder er fügt seine bearbeiteten Bausteine in eine andere, im Serverprojekt bereits vorhandene CPU ein.

Mögliche Szenarien beim Einchecken von Objekten, die im Serverprojekt gelöscht sind

Beim Einchecken von Objekten, die im Serverprojekt bereits gelöscht sind, werden diese Objekte wieder erstellt.

Beispiel:

- Anwender 1 bearbeitet in seiner lokalen Session die markierte Variable "Tag_1", die zu einer Gruppe gehört, die eine Variablentabelle mit mehreren Variablen enthält.
- Anwender 2 löscht diese Gruppe mit der gesamten Variablentabelle in der Serverprojekt-Ansicht und speichert seine Änderungen im Serverprojekt.

Ergebnis:

Anwender 1 ignoriert die Anzeige, dass die Variable "Tag_1" veraltet ist, arbeitet weiter und checkt seine Änderungen ein. Im Serverprojekt wird die Gruppe mit der Variablentabelle wieder erstellt, die jetzt jedoch nur noch die Variable "Tag_1" enthält.

Umbenennen von Objekten

Das Umbenennen von Objekten in der lokalen Session sollte vermieden werden, da es Namenskonflikte beim Einchecken erzeugt. Falls Objekte umbenannt werden müssen, sollte dies direkt in der Serverprojekt-Ansicht durchgeführt werden. Gleichzeitig darf keine zu diesem Serverprojekt zugehörige lokale Session mit markierten Objekten vorhanden sein.

Unterschiedliche Projektsprachen in Serverprojekt und lokaler Session

Wird eine Projektsprache in einer lokalen Session geändert, so gehen diese Änderungen nach dem Einchecken und nach dem Aktualisieren in der lokalen Session verloren.

Wird eine Projektsprache nachträglich in der Serverprojekt-Ansicht gelöscht, so existiert diese Projektsprache nicht länger im Serverprojekt.

Änderungen an den Projektsprachen sollten immer nur direkt in der Serverprojekt-Ansicht durchgeführt werden. Beachten Sie, dass keine lokalen Sessions mit Markierungen zu diesem Serverprojekt bearbeitet werden oder geöffnet sind.

Siehe auch

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

Serverprojekt-Ansicht öffnen und schließen (Seite 175)

2.17.2.12 Mögliche Fehlermeldungen beim Einchecken

Einführung

Beim Einchecken der markierten Änderungen aus der lokalen Session in das Multiuser-Serverprojekt können die nachfolgend beschriebenen Fehlermeldungen auftreten:

- Transferfehler
- Übersetzungsfehler

Nachfolgend wird die Vorgehensweise beim Auftreten der genannten Fehlermeldungen beschrieben.

Transferfehler

Beim Auftreten von Transferfehlern beim Einchecken werden automatisch folgende Aktionen durchgeführt:

- Eine Fehlermeldung informiert Sie über die Art des aufgetretenen Fehlers und die weitere Vorgehensweise. Nach dem Bestätigen wird diese Meldung geschlossen.
- Bei einem kritischen Transferfehler wird der Zustand vor Beginn des Eincheckens sowohl in der lokalen Session als auch im Serverprojekt wieder hergestellt. Es gehen keine Änderungen verloren. Beheben Sie den Fehler in Ihrer lokalen Session.
- Bei einem in der Serverprojekt-Ansicht behebbaren Transferfehler wird die Serverprojekt-Ansicht geöffnet. Beheben Sie den Fehler in der Serverprojekt-Ansicht.
- Im Multiuser-Editor werden in der Ansicht "Einchecken" die fehlerhaften Objekte angezeigt und gekennzeichnet.
- Der für das Einchecken gesperrte Multiuser-Server wird wieder frei gegeben, falls die Fehlerbehebung in der lokalen Session durchgeführt wird.
- Der Multiuser-Server bleibt während der Fehlerbehebung in der Serverprojekt-Ansicht weiterhin gesperrt. Beachten Sie, dass während dieser Zeit kein anderer Benutzer seine Änderungen einchecken kann.

Transferfehler beheben

Nach dem Auftreten eines Transferfehlers haben Sie folgende Möglichkeiten für die weitere Vorgehensweise:

1. Beheben Sie die angezeigten Fehler in Ihrer lokalen Session oder in der Serverprojekt-Ansicht.
2. Starten Sie danach das Einchecken erneut, damit ein fehlerfreies Projekt eingecheckt werden kann.

Übersetzungsfehler

Ist beim Einchecken die Option "Übersetzen" aktiviert, wird nach dem Transfer der markierten Änderungen ins Serverprojekt ein Übersetzungsvorgang mit der ausgewählten Option durchgeführt. Folgende Übersetzungsarten stehen zur Verfügung:

- Markierte Objekte (nur Änderungen)
  Es werden die Änderungen an den markierten Objekten für das ausgewählte Gerät zusammen mit den gerätelokalen Objekten übersetzt.
- Gerätesoftware (nur Änderungen)
  Es werden die Änderungen an den markierten Objekten zusammen mit dem auf den zugeordneten Geräten vorhandenen Softwarestand übersetzt.

Hinweis

Beim Einchecken angezeigte Optionen für das Übersetzen

Die beim Einchecken angezeigten Optionen für das Übersetzen sind abhängig von den im TIA Portal unter "Extras > Einstellungen > Multiuser" bei "Übersetzungseinstellungen" vorgenommenen Einstellungen.

Aufgrund dieser Einstellungen können die Übersetzungsoptionen im Multiuser-Editor eventuell deaktiviert sein.

Bei einem erfolgreichen Übersetzungsvorgang werden die Änderungen aus der lokalen Session automatisch ins Serverprojekt transferiert.

Beim Auftreten eines Übersetzungsfehlers werden folgende Aktionen durchgeführt:

- Die Fortschrittsanzeige für das Einchecken wird geschlossen und das Einchecken wird angehalten.
- Im Multiuser-Editor werden in der Ansicht "Einchecken" die fehlerhaften Objekte angezeigt und gekennzeichnet.
- Der Multiuser-Server bleibt während der Fehlerbehebung in der Serverprojekt-Ansicht weiterhin gesperrt. Beachten Sie bitte, dass während dieser Zeit kein anderer Benutzer seine Änderungen einchecken kann.

Übersetzungsfehler beheben

Nach dem Auftreten eines Übersetzungsfehlers haben Sie folgende Möglichkeiten für die weitere Vorgehensweise:

1. Beheben Sie die angezeigten Fehler in Ihrer lokalen Session oder in der Serverprojekt-Ansicht.
2. Starten Sie danach das Einchecken mit Übersetzen erneut, damit ein fehlerfreies Projekt eingecheckt werden kann.

Übersetzungsfehler ignorieren

Sie haben auch die Möglichkeit, die angezeigten Übersetzungsfehler zu ignorieren und ein fehlerhaftes Projekt einzuchecken.

ACHTUNG
Einchecken mit Übersetzungsfehlern Diese Vorgehensweise ist zwar möglich, wird jedoch nicht empfohlen! Um eine effektive Weiterarbeit für alle beteiligten Bearbeiter zu ermöglichen, sollten Sie möglichst nur fehlerfreie Projekte auf dem Multiuser-Server einchecken.

Siehe auch

Einführung zum Arbeiten in lokalen Sessions (Seite 147)

2.17.2.13 Objekte in der lokalen Session löschen

Einführung

Beim Arbeiten mit Multiuser Engineering können Sie Objekte in der lokalen Session löschen.

Für das Löschen von Objekten gelten folgende Regeln:

- Gelöschte Objekte werden in der lokalen Session nicht mehr angezeigt.
- Über die Funktion "rückgängig machen" kann die Aktion "Löschen" in der lokalen Session wieder rückgängig gemacht werden.
- Gelöschte Objekte werden vom System automatisch markiert und im Multiuser-Editor im Ordner "Gelöschte Objekte" angezeigt.

- Wenn Sie die Markierung für ein gelöschtes Objekt im Multiuser-Editor wieder zurücknehmen, wird das gelöschte Objekt **nicht** ins Serverprojekt eingecheckt. Das heißt, dass das in der lokalen Session gelöschte Objekt im Serverprojekt **nicht** gelöscht wird. In der lokalen Session wird das zunächst gelöschte, aber nicht eingecheckte Objekt, erst wieder nach dem nächsten Aktualisieren angezeigt.
- Beim Löschen von nicht markierbaren Objekten in der lokalen Session sind diese Objekte nach dem Aktualisieren in der lokalen Session wieder vorhanden. Beim Einchecken werden nur markierte Objekte transferiert. Da die gelöschten Objekte nicht markierbar waren, wird das Löschen nicht ins Serverprojekt übernommen. Beim nächsten Aktualisieren der lokalen Session werden diese Objekte wieder in die lokale Session eingefügt.

Voraussetzung

Die Verbindung zum Multiuser-Server ist vorhanden und der Server ist verfügbar.

Vorgehensweise beim Löschen von Objekten

Um Objekte in der lokalen Session zu löschen, gehen Sie vor wie folgt:

1. Öffnen Sie die gewünschte lokale Session.
2. Löschen Sie das gewünschte markierte Objekt oder löschen Sie ein nicht markiertes, aber markierbares Objekt.
3. Klicken Sie auf die Schaltfläche "Einchecken" in der Multiuser-Funktionsleiste.
   Daraufhin öffnet sich der Multiuser-Editor in der Ansicht "Einchecken" und zeigt alle in der lokalen Session markierten Objekte an, die beim Einchecken ins Serverprojekt übernommen werden. Im Ordner "gelöschte Objekte" werden die in der lokalen Session gelöschten Objekte angezeigt.
4. Überprüfen Sie die angezeigten und für das Löschen markierten Objekte. Sie können die Markierungen wieder zurücknehmen, wenn Sie bereits markierte Objekte doch nicht löschen möchten.
5. Wählen Sie die gewünschten Optionen für das Einchecken aus:
   - "Übersetzen": Bei Anklicken dieser Option werden Ihre Änderungen mit der Option "Software, nur Änderungen" vor dem Einchecken übersetzt. Dies bedeutet, dass die von Ihnen vorgenommenen Software-Änderungen mit allen davon abhängigen Objekten im Serverprojekt übersetzt werden.
   - "Serverprojekt-Ansicht öffnen": Bei Anklicken dieser Option wird vor dem Einchecken die Serverprojekt-Ansicht in der Projektnavigation geöffnet. Die Serverprojekt-Ansicht zeigt Ihnen vorab den Inhalt des aktuellen Serverprojekts mit Ihren zum Einchecken markierten Änderungen an. In der Serverprojekt-Ansicht können Sie alle Objekte markieren und bearbeiten, sowie vorhandene Markierungen zurücknehmen.
6. Klicken Sie auf die Schaltfläche "Einchecken", um die angezeigten Objekte in das Multiuser-Serverprojekt einzuchecken.
   Falls Sie die Markierungen in der lokalen Session auch nach dem Einchecken behalten möchten, müssen Sie die Option "Markierungen behalten" aktivieren.
7. Wenn das Einchecken nicht erfolgreich war und wegen Fehlern abgebrochen wurde, erhalten Sie entsprechende Fehlermeldungen. Beheben Sie die Fehler in Ihrer lokalen Session oder in der Serverprojekt-Ansicht. Starten Sie danach das Einchecken erneut.

Ergebnis

Ihre lokale Session wird nach dem erfolgreichen Einchecken mit den Inhalten des Serverprojekts aktualisiert und geöffnet.

- Die lokale Session enthält nach der Aktualisierung alle Änderungen, die zwischenzeitlich im Serverprojekt eingecheckt wurden.
- Falls neue Objekte im Serverprojekt hinzugefügt wurden, sind diese Objekte nun auch in Ihrer lokalen Session sichtbar.
- Die im Serverprojekt gelöschten Objekte sind nun in Ihrer lokalen Session nicht mehr vorhanden.
- Ihre Markierungen in der lokalen Session werden automatisch gelöscht, falls Sie die Option "Markierung für Objekte erhalten" nicht aktiviert hatten.
- Die Serververbindung wird nach dem Einchecken wieder frei gegeben. Der Serverstatus wechselt in der Anzeige innerhalb der Funktionsleiste von "belegt" auf "verfügbar".

2.18 In der Serverprojekt-Ansicht arbeiten

2.18.1 Einführung zum Arbeiten in der Serverprojekt-Ansicht

Einführung

Sie haben bei Multiuser Engineering zusätzlich zum Arbeiten in lokalen Sessions auch die Möglichkeit, direkt in der Serverprojekt-Ansicht zu arbeiten.

Die Serverprojekt-Ansicht kann über das Symbol in der Multiuser-Funktionsleiste zum Bearbeiten geöffnet werden.

Aufbau der Serverprojekt-Ansicht

Die Serverprojekt-Ansicht zeigt den aktuellen Inhalt des Serverprojektes in der Projektnavigation unterhalb der lokalen Session an und wird gelb hinterlegt dargestellt.

Die folgende Grafik zeigt ein Beispiel für ein in der Serverprojekt-Ansicht geöffnetes Multiuser-Projekt:

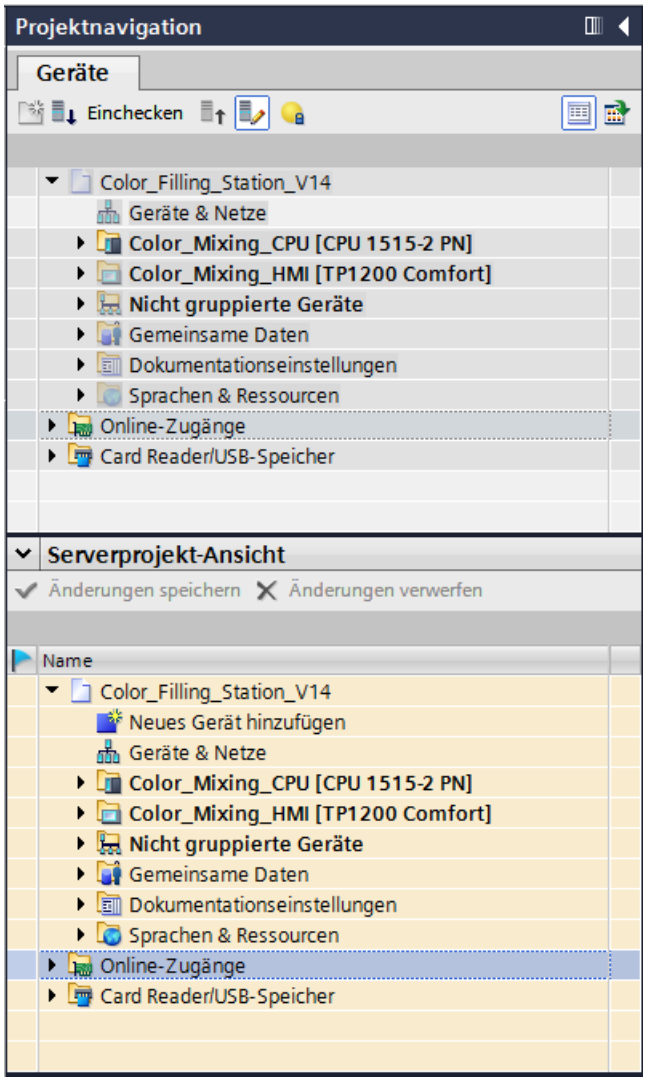

Aktionen in der Serverprojekt-Ansicht

Folgende Aktionen können Sie in der Serverprojekt-Ansicht durchführen:

- Die Serverprojekt-Ansicht ermöglicht das Arbeiten direkt im Serverprojekt.
- In der Serverprojekt-Ansicht können Sie alle vorhandenen Objekte bearbeiten, auch solche, die in den lokalen Sessions nicht bearbeitet werden dürfen.
- In der Serverprojekt-Ansicht können Sie mit der Funktionalität von Inter Project Engineering (IPE) arbeiten und Daten zwischen CPU und HMI über einen PLC-Proxy austauschen.
- Das gemeinsame Arbeiten auf einer CPU im Rahmen einer gemeinsamen Inbetriebnahme wird ebenfalls unterstützt.
- Das Hochladen von Bausteinen oder größeren Programmteilen aus einer CPU sollte nur in der Serverprojekt-Ansicht durchgeführt werden.

- In der Serverprojekt-Ansicht können Sie auch Änderungen in der Projektierung mit Hilfe von externen Tools vornehmen. Beim Arbeiten in den lokalen Sessions wird dies nicht unterstützt.
- Änderungen an anwenderdefinierter Dokumentaion, wie z. B. das Hinzufügen neuer Anwendertexte oder neuer HMI-Scripte, sollte ebenfalls nur in der Serverprojekt-Ansicht durchgeführt werden.
- Die Objekte, die bearbeitet werden, müssen auch in der Serverprojekt-Ansicht zum Bearbeiten markiert werden, damit die anderen Bearbeiter sehen, dass diese Objekte geändert werden.
- Über die Schaltflächen "Änderungen speichern" und "Änderungen verwerfen" können Sie die in der Serverprojekt-Ansicht vorgenommenen Änderungen speichern oder verwerfen.
- Mit dem Speichern werden die vorgenommenen Änderungen direkt im Serverprojekt gespeichert.

Hinweis

Einschränkungen beim Arbeiten in der Serverprojekt-Ansicht

Sobald die Serverprojekt-Ansicht geöffnet ist, gelten folgende Bedingungen:

- Die lokale Session ist für das Bearbeiten gesperrt.
- Der Server ist für die anderen Bearbeiter gesperrt.
- Das Symbol für den Serverstatus wird in der Funktionsleiste in Gelb dargestellt.
- Falls beim Arbeiten in der Serverprojekt-Ansicht die von Ihnen vorgenommenen Änderungen nicht gespeichert werden können, sollten Sie zusätzlich noch ein anderes Objekt markieren und wieder abwählen, damit die Schaltfläche "Änderungen speichern" aktiviert wird und das Speichern möglich ist.

Siehe auch

Serverprojekt-Ansicht öffnen und schließen (Seite 175)

Objekte in der Serverprojekt-Ansicht bearbeiten (Seite 177)

2.18.2 Serverprojekt-Ansicht öffnen und schließen

Einführung

Die Serverprojekt-Ansicht ermöglicht Ihnen das Bearbeiten aller Objekte, die im Serverprojekt vorhanden sind.

Es gibt folgende Möglichkeiten, die Serverprojekt-Ansicht zu öffnen:

- über das Symbol in der Multiuser-Funktionsleiste
- über den in den Bannertexten angezeigten Link
- über die Schaltfläche im Multiuser-Editor in der Ansicht "Einchecken"

Wenn die Serverprojekt-Ansicht nicht geöffnet werden kann, weil der Server momentan durch einen anderen Bearbeiter verwendet wird und gesperrt ist, erhalten Sie eine Meldung.

Angezeigte Inhalte in der Serverprojekt-Ansicht

Die Serverprojekt-Ansicht zeigt immer den aktuellen Stand des Multiuser-Serverprojektes an.

Wird die Serverprojekt-Ansicht im Multiuser-Editor in der Ansicht "Einchecken" geöffnet, so zeigt sie zusätzlich zum aktuellen Inhalt des Serverprojektes die in der lokalen Session vorgenommenen Änderungen an. Sie gibt also bereits eine Vorschau darauf, wie das Serverprojekt nach dem Einchecken der in der lokalen Session vorgenommenen Änderungen aussieht.

Öffnen der Serverprojekt-Ansicht

Um die Serverprojekt-Ansicht zu öffnen, gehen Sie vor wie fogt:

1. Öffnen Sie die gewünschte lokale Session.
2. Klicken Sie in der Multiuser-Funktionsleiste auf das Symbol "Serverprojekt-Ansicht öffnen/schließen".
   Oder Sie klicken in der Multiuser-Funktionsleiste auf "Einchecken" und im daraufhin geöffneten Multiuser-Editor auf die Schaltfläche "Serverprojekt-Ansicht vor dem Einchecken öffnen".
   Alternativ können Sie auch auf den Link in einem angezeigten Bannertext klicken.

Schließen der Serverprojekt-Ansicht

Um die Serverprojekt-Ansicht zu schließen, gehen Sie vor wie folgt:

1. Klicken Sie erneut auf das Symbol "Serverprojekt-Ansicht öffnen/schließen" in der Multiuser-Funktionsleiste.
2. Wenn Sie Änderungen in der Serverprojekt-Ansicht vorgenommen haben, erfolgt eine Abfrage, ob Sie die Änderungen speichern oder verwerfen möchten.
3. Klicken Sie auf "Ja", um Ihre Änderungen zu speichern oder klicken Sie auf "Nein", um Ihre Änderungen zu verwerfen.
   Alternativ können Sie auch vor dem Schließen auf die entsprechenden Schaltfläche in der Serverprojekt-Ansicht klicken, um Ihre Änderungen zu speichern oder zu verwerfen.

Ergebnis

Die Serverprojekt-Ansicht wird geschlossen.

Die lokale Session wird wieder zum Bearbeiten geöffnet.

Der Server wird wieder freigegeben, der Serverstatus wird in grün angezeigt.

Siehe auch

Einführung zum Arbeiten in der Serverprojekt-Ansicht (Seite 173)

Objekte in der Serverprojekt-Ansicht bearbeiten (Seite 177)

2.18.3 Objekte in der Serverprojekt-Ansicht bearbeiten

Einführung

Sie können im Kontext von Multiuser Engineering Objekte auch direkt in der Serverprojekt-Ansicht bearbeiten.

Die Serverprojekt-Ansicht ermöglicht Ihnen, alle im Serverprojekt enthaltenen Objekte zu bearbeiten, auch solche Objekte, die in den lokalen Sessions nicht bearbeitet werden dürfen.

Das Arbeiten in der Serverprojekt-Ansicht ist ein exklusives Arbeiten direkt im Multiuser-Serverprojekt. Während des Arbeitens in der Serverprojekt-Ansicht ist der Multiuser-Server für die anderen Bearbeiter gesperrt. Die zugehörige lokale Session ist während des Arbeitens in der Serverprojekt-Ansicht nicht editierbar.

Auch in der Serverprojekt-Ansicht müssen die Objekte markiert sein, die Sie einchecken möchten. Damit stellen Sie sicher, dass andere Bearbeiter in ihrer lokalen Session sehen, dass Sie diese Objekte gerade bearbeiten. Markierungskonflikte können dadurch vermieden werden.

Auch in der Serverprojekt-Ansicht können Sie die Funktion "Automarkieren" verwenden oder alternativ die zu bearbeitenden Objekte manuell markieren.

Hinweis

Markieren von identischen Objekten in der lokalen Session und in der Serverprojekt-Ansicht

Beachten Sie, dass Sie einen Markierungskonflikt mit einem roten Fähnchen angezeigt bekommen, wenn Sie ein Objekt mehrfach markieren, auch wenn beide Markierungen zum selben Benutzer gehören.

Bearbeiten von Objekten in der Serverprojekt-Ansicht

Folgende Aktionen können Sie beispielsweise in der Serverprojekt-Ansicht durchführen:

- Hinzufügen neuer Objekte, die im Serverprojekt bisher noch nicht vorhanden waren.
- Hardwarekonfiguration und Verbindungen hinzufügen oder ändern.
- Versionsänderungen an den verwendeten Anweisungen durchführen.
- Mit Inter Project Engineering (IPE) arbeiten.
- Bearbeiten aller Objekte, unabhängig davon, ob diese markierbar sind oder nicht.
- Umbenennen, Verschieben und Löschen von Objekten.
- Eine Online-Verbindung zu einer CPU herstellen und laden oder hochladen von Objekten.
- Alle von Multiuser Engineering unterstützten Funktionen des TIA Portals ausführen.

Vorgehensweise zum Bearbeiten von Objekten in der Serverprojekt-Ansicht

Um Objekte in der Serverprojekt-Ansicht zu bearbeiten, gehen Sie vor wie folgt:

1. Öffnen Sie die gewünschte lokale Session.
2. Klicken Sie in der Multiuser-Funktionsleiste auf das Symbol "Serverprojekt-Ansicht öffnen/schließen".
   Die Serverprojekt-Ansicht wird daraufhin unterhalb der lokalen Session in der Projektnavigation geöffnet und gelb hinterlegt dargestellt.
   Beim erneuten Klicken auf dieses Symbol wird die Serverprojekt-Ansicht wieder geschlossen.
3. Markieren Sie in der Serverprojekt-Ansicht die Objekte, die Sie bearbeiten möchten, indem Sie auf die vorangestellten Fähnchen klicken oder nutzen Sie die Funktion "Automarkieren", indem Sie die Objekte direkt im Editor öffnen und bearbeiten.
4. Führen Sie die gewünschten Änderungen durch.
5. Speichern Sie Ihre Änderungen, indem Sie auf die Schaltfläche "Änderungen speichern" klicken.
   Oder klicken Sie auf die Schaltfläche "Änderungen verwerfen", wenn Sie Ihre Änderungen nicht ins Serverprojekt übernehmen möchten.
6. Tragen Sie einen Kommentar zur Änderungshistorie ein, um Ihre in der Serverprojekt-Ansicht vorgenommenen Änderungen zu dokumentieren.
7. Falls Sie die Änderungen aus der Serverprojekt-Ansicht in die lokale Session übernehmen wollen, klicken Sie nach dem Speichern der Änderungen in der lokalen Session auf die Schaltfläche "Lokale Session aktualisieren".

Ergebnis

Die in der Serverprojekt-Ansicht vorgenommenen Änderungen werden im Serverprojekt gespeichert.

- Die Serverprojekt-Ansicht wird geschlossen und die lokale Session wird wieder zum Bearbeiten frei gegeben.
- Die Markierungen in der Serverprojekt-Ansicht werden zurückgenommen.
- Die lokale Session wird **nicht** automatisch aktualisiert mit den neuen Inhalten des Serverprojekts. Das Symbol für den Server zeigt an, dass eine neuere Version des Serverprojektes vorhanden ist.

Serverprojekt-Ansicht ist geöffnet und Serververbindung wird unterbrochen

Wenn Sie die Serverprojekt-Ansicht bereits geöffnet haben und danach die Verbindung zum Multiuser-Server unterbrochen wird, erhalten Sie eine entsprechende Fehlermeldung.

Sie haben dann folgende Optionen:

- Klicken Sie auf die Schaltfläche "Erneut verbinden", um eine neue Verbindung zum zugehörigen Multiuser-Server herzustellen.
- Oder klicken Sie auf die Schaltfläche "Verwerfen", um die in der Serverprojekt-Ansicht vorgenommenen Änderungen zu verwerfen.

Falls es nicht möglich ist, die unterborchene Serververbindung wieder herzustellen und Sie Ihre Änderungen verwerfen, wird die lokale Session mit einer Fehlermeldung offline geöffnet. Die lokale Session kann auch geöffnet werden, wenn der zugehörige Server gerade nicht verfügbar ist.

Sie können in der lokalen Session offline weiter arbeiten mit folgenden Einschränkungen:

- Markierungen von anderen Anwendern sind in Ihrer lokalen Offline-Session nicht sichtbar, es werden also keine gelben oder roten Fähnchen und auch keine veralteten Objekte angezeigt.
- Änderungen, die Sie in der lokalen Offline-Session vornehmen, können erst dann ins Serverprojekt eingecheckt werden, wenn wieder eine Serververbindung vorhanden ist.

Siehe auch

Einführung zum Arbeiten in der Serverprojekt-Ansicht (Seite 173)

Serverprojekt-Ansicht öffnen und schließen (Seite 175)

2.18.4 Objekte in der Serverprojekt-Ansicht umbenennen

Einführung

Beim Arbeiten mit Multiuser Engineering können Objekte umbenannt werden.

Regeln für das Umbenennen von Objekten

Folgende Regeln gelten für das Umbennen von Objekten:

- Das Umbenennen von Objekten in der lokalen Session sollte vermieden werden, da es zu Namenskonflikten beim Einchecken führen kann.
- Das Umbenennen von Objekten sollte immer in der Serverprojekt-Ansicht durchgeführt werden.
- Beim Umbenennen von Objekten in der Serverprojekt-Ansicht dürfen die zu umbenennenden Objekte keine Markierungen innerhalb von lokalen Sessions haben.
- Durch das Umbenennen von Objekten kann es eventuell zu Namenskonflikten kommen. Stellen Sie sicher, dass Sie beim Umbenennen im Projekt bereits verwendete Namen nicht nochmals vergeben.
- Konflikte, die beim Umbenennen auftreten, werden nicht automatisch behoben, um unbeabsichtigte Referenzen von Objekten zu vermeiden.
- Das Umbenennen von Geräten erzeugt für die Objekte innerhalb eines Geräts keine Konflikte, da Geräte nicht durch ihren Namen sondern durch ihre ID identifiziert werden.
- Das Umbenennen von Geräten erzeugt jedoch Konflikte bei eventuell vorhandenen Referenzen zwischen Geräten (PLC und HMI).
- Nach dem Umbenennen von Objekten sollten die anderen Bearbeiter ein Aktualisieren Ihrer lokalen Sessions durchführen.

Voraussetzung

Die Verbindung zum Multiuser-Server ist vorhanden und der Server ist verfügbar.

Es ist keine andere lokale Session zu diesem Serverprojekt geöffnet.

Objekte umbenennen

Um Objekte in der Serverprojekt-Ansicht umzubenennen, gehen Sie vor wie folgt:

1. Öffnen Sie die gewünschte lokale Session.
2. Klicken Sie in der Multiuser-Funktionsleiste auf das Symbol "Serverprojekt-Ansicht öffnen/schließen".
   Die Serverprojekt-Ansicht wird daraufhin unterhalb der lokalen Session in der Projektnavigation geöffnet und gelb hinterlegt dargestellt.
   Beim erneuten Klicken auf dieses Symbol wird die Serverprojekt-Ansicht wieder geschlossen.
3. Markieren Sie die gewünschten Objekte in der Serverprojekt-Ansicht und benennen Sie diese um.
4. Klicken Sie auf die Schaltfläche "Änderungen speichern", um die Änderungen in das Multiuser-Serverprojekt zu übernehmen.

Ergebnis

Die in der Serverprojekt-Ansicht vorgenommenen Änderungen werden im Serverprojekt gespeichert.

- Die Serverprojekt-Ansicht wird geschlossen und die lokale Session wird wieder zum Bearbeiten frei gegeben.
- Die Markierungen in der Serverprojekt-Ansicht werden zurückgenommen.
- Die lokale Session wird **nicht** automatisch aktualisiert mit den neuen Inhalten des Serverprojekts.

Multiuser Commissioning einsetzen 3

3.1 Einführung in Multiuser Commissioning

Einführung

Im TIA Portal haben Sie ab V15.1 die Möglichkeit, im Rahmen des Produktes "Multiuser Engineering" in der Inbetriebnahmephase mit Multiuser Commissioning zu arbeiten.

Nachdem Sie mit Multiuser Engineering mit mehreren Benutzern gemeinsam und gleichzeitig die Multiuser-Projekte erstellt und bearbeitet haben, können Sie die Inbetriebnahme mit Multiuser Commissioning genauso komfortabel und ebenfalls im Team durchführen.

Dadurch werden die Inbetriebnahmezeiten für Ihre Projekte wesentlich verkürzt. Sie können Ihre Anlage schneller nutzen und produktiv damit arbeiten.

Arbeiten mit Multiuser Commissioning

Und so funktioniert Multiuser Commissioning im TIA Portal:

1. Zuerst erstellen und bearbeiten Sie Ihre Projekte in der Engineering-Phase mit Multiuser Engineering.
   siehe hierzu: Übersicht über die Arbeitsweise mit Multiuser Engineering (Seite 16)
2. Sobald Ihre Projekte reif für die Inbetriebnahme sind, arbeiten Sie mit Multiuser Commissioning weiter.
   - Die Projektverwaltung liegt auch beim Commissioning auf einem lokalen oder externen Server.
   - Verschiedene Anwender arbeiten in lokalen Sessions, basierend auf den vom Server verwalteten Projekten.
   - Das Arbeiten in den lokalen Sessions ist auch bei der Inbetriebnahme unabhängig voneinander.
   - Die Änderungen aus den lokalen Sessions werden beim Laden immer zuerst in die gemeinsam genutzte CPU geladen und erst nach einem erfolgreichen Laden ins Serverprojekt übertragen.
   - Geladene Änderungen von anderen Bearbeitern werden angezeigt und können komfortabel übernommen werden.
   - Das Synchronisieren des Serverprojektes erfolgt bereits auf dem Multiuser-Server. Somit können immer konsistente Stände des Serverprojekts in die CPU geladen werden. Das mühsame Synchronisieren unterschiedlicher Projektstände auf der CPU entfällt damit.

Hinweis

Effizientes Arbeiten mit Multiuser Commissioning

Beim parallelen Arbeiten in mehreren lokalen Sessions, sollten Sie auftretende Fehler frühzeitig beheben, damit das verteilte Arbeiten an einem gemeinsamen Projekt effektiv bleibt.

Daher ist es empfehlenswert, die anstehenden Änderungen vor dem Laden und Einchecken im Multiuser-Editor zuerst zu übersetzen. Dazu stehen Ihnen mehrere Übersetzungsoptionen zur Verfügung. Falls Übersetzungsfehler angezeigt werden, sollten Sie diese vor dem Laden und Einchecken beseitigen.

Durch das Übersetzen der anstehenden Änderungen vor dem Einchecken stellen Sie sicher, dass Sie ein fehlerfreies Projekt einchecken und auf die CPU laden.

Durch das Laden eines Projektstandes, der sowohl im Serverprojekt als auch auf der CPU immer synchronisiert und damit konsistent ist, werden weniger Fehler an andere Bearbeiter weitergegeben.

Durch das parallele Bearbeiten von verschiedenen Objekten, wie z. B. unterschiedlichen Variablen, Bausteinen, PLC-Variablen, Alarmen, Bildern, Text- und Grafiklisten, Technologieobjekten usw. in einem oder in mehreren Geräten innerhalb eines Multiuser-Projektes, können Sie Ihre Inbetriebnahmezeiten effektiver gestalten und wesentlich verkürzen.

Siehe auch

Übersicht über die Arbeitsweise mit Multiuser Engineering (Seite 16)

3.2 Übersicht über die Arbeitsweise mit Multiuser Commissioning

Einführung

Wenn Sie mit mehreren Benutzern gemeinsam und gleichzeitig in Multiuser-Projekten während der Inbetriebnahmephase arbeiten, sollten bestimmte Vorgehensweisen eingehalten werden.

Nachfolgend erhalten Sie einen Überblick über die generelle Arbeitsweise mit Multiuser Commissioning.

Voraussetzungen für das Arbeiten mit Multiuser Commissioning

Folgende Schritte haben Sie bereits mit Multiuser Engineering in der Engineering-Phase durchgeführt:

1. Multiuser-Server installieren.
2. Geeignetes Multiuser-Projekt auf dem Server anlegen.
3. Lokale Sessions für alle Bearbeiter anlegen.
4. Schrittweises Erstellen der Automatisierungsaufgabe in den verschiedenen lokalen Sessions und Einchecken aller Änderungen ins Serverprojekt.
5. Aktualisieren der lokalen Sessions mit dem Serverprojekt.
6. Schrittweises Weiterarbeiten, bis das Multiuser-Projekt reif für die Inbetriebnahe ist und die Engineering-Phase beendet werden kann.

Arbeiten mit Multiuser Commissioning

Beachten Sie für das Arbeiten in Multiuser-Projekten unbedingt die nachfolgenden Hinweise:

- Die Projekte sollten bereits mit Multiuser Engineering bearbeitet worden sein und die gesamte Hardware-Konfiguration, die benötigten Bausteine, Verbindungen und globalen Einstellungen enthalten, bevor diese in der Inbetriebnahmephase mit Multiuser Commissioning verwendet werden können.
  Weiterführende Informationen dazu finden Sie unter: Anforderungen an Multiuser-Projekte (Seite 35)
- Vermeiden Sie das gleichzeitige Arbeiten an denselben Objekten durch rechtzeitige Absprachen mit den anderen Bearbeitern.
- Alle globalen Einstellungen, wie z. B. Änderungen innerhalb der Gerätekonfiguration, müssen Sie in der Serverprojekt-Ansicht (also im zentralen Serverprojekt) vornehmen.
- In der Serverprojekt-Ansicht können alle Objekte eines Multiuser-Projektes bearbeitet werden. Sie können auch die Objekte bearbeiten, die die Funktionalität von Multiuser Engineering nicht unterstützen.
- Alle ins Serverprojekt eingecheckten Änderungen aus den lokalen Sessions müssen kompiliert sein, um auf dem Serverprojekt immer einen konsistenten Stand zu halten.
- Nach dem Laden in die CPU erfolgt automatisch eine Aktualisierung des Multiuser-Serverprojekts. Nach jedem Laden wird eine neue Revision des Serverprojekts erstellt.

Übersicht über die Arbeitsweise mit Multiuser Commissioning

Um die Funktionalität von Multiuser Commissioning zu nutzen, führen Sie folgende Schritte durch:

Aktion	Beispiel
1. Bearbeitung in der Engineering-Phase ist abgeschlossen Die Bearbeitung der unterschiedlichen Projektierungsaufgaben in den lokalen Sessions ist abgeschlossen. **Ergebnis:** Alle Änderungen sind kompiliert und ins Serverprojekt eingecheckt. Die lokalen Sessions und das Serverprojekt sind synchronisiert. In den lokalen Sessions sind keine Markierungen mehr vorhanden. Das Multiuser-Serverprojekt wurde in die CPU geladen. Somit ist der Stand des Serverprojekts und der Stand des Multiuser-Projekts auf der CPU konsistent. **Hinweis:** Die Engineering-Phase ist damit beendet.	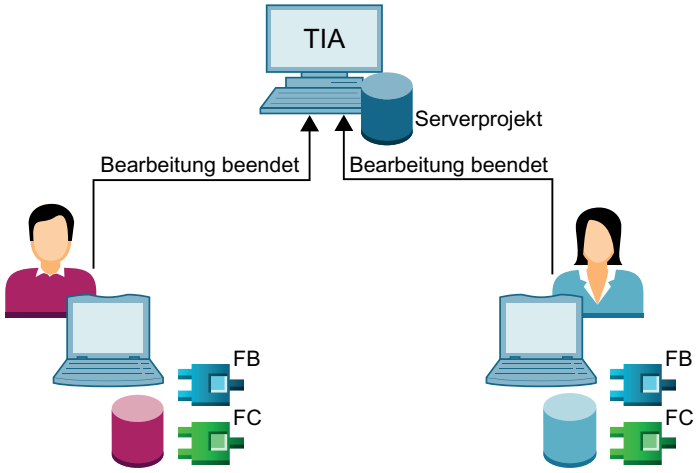 Das Multiuser-Projekt ist reif für die Inbetriebnahme. Die Engineering-Phase ist damit beendet.

Aktion	Beispiel
2. Arbeiten in der Commissioning-Phase Sie schalten nun im TIA Portal auf den "Commissioning-Modus" um und starten mit der Inbetriebnahmephase. Das Umschalten wird im Administration Tool vorgenommen. Die Vorgehensweise ist wie folgt: • Sie bearbeiten die lokale Session im Commissioning-Modus weiter wie gewohnt. • Alle Änderungen aus den lokalen Sessions werden kompiliert und ins Serverprojekt eingecheckt. • Nach jedem Laden in die CPU wird das Serverprojekt und die lokale Session automatisch aktualisiert. Änderungen anderer Benutzer sind dann in der lokalen Session sichtbar. • Nach jedem Laden in die CPU wird eine neue Revision des Serverprojekts angelegt. • Arbeiten Sie so lange in der Commissioning-Phase weiter, bis die Inbetriebnahme abgeschlossen ist. • Falls gewünscht, können Sie das Multiuser-Serverprojekt als Single-User-Projekt exportieren und wie gewohnt im TIA Portal weiter bearbeiten und verwalten. **Ergebnis:** Im Commissioning-Modus wird sichergestellt, dass das Online-Projekt in der CPU und das Offline-Projekt im Multiuser Server konsistent sind. Dies erleichtert die Inbetriebnahme, da kein unnötiger Synchronisierungsaufwand entsteht. Es werden beim Laden jedoch falls gewünscht auch Inkonsistenzen zugelassen, wenn Sie z. B. ohne vorheriges Einchecken direkt aus der lokalen Session in die CPU laden.	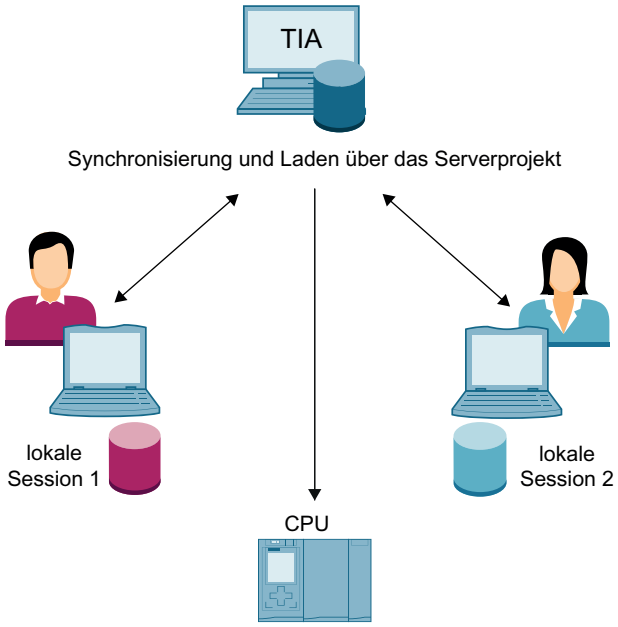

Siehe auch

Anforderungen an Multiuser-Projekte (Seite 35)

3.3 Einstellungen für das Arbeiten mit Multiuser Commissioning

Einführung

Für das Arbeiten mit Multiuser Commissioning gelten die allgemeinen Einstellungen für Multiuser Engineering ebenfalls. Sie können im TIA Portal unter "Extras > Einstellungen" im Register "Multiuser" folgende multiuser-spezifische Einstellungen vornehmen:

- Allgemein
- Multiuser-Serververbindung
- Multiuser-Netzwerkprofile und Kompression
- Suche in Projekt
- Speichereinstellungen
- Übersetzungseinstellungen

Siehe hierzu: Einstellungen für das Arbeiten mit Multiuser Engineering (Seite 29)

Einstellungen für Multiuser Commissioning

Ab dem TIA Portal V15.1 können Sie beim Arbeiten mit Multiuser Engineering zwischen dem Engineering-Modus und dem Commissioning-Modus umschalten.

Das Umschalten wird im Administration Tool durchgeführt.

Siehe hierzu: Multiuser-Server mit grafischem Administration Tool verwalten (Seite 65)

Öffnen Sie das Administration Tool und fügen Sie den verwendeten Server hinzu.

Wählen Sie das gewünschte Multiuser-Projekt aus.

In der Funktionsleiste des Administration Tools ist die Schaltfläche "Multiuser Commissioning" eingefügt.

Sobald diese Schaltfläche aktiviert ist, wird in den Commissioning-Modus umgeschaltet. Das Symbol für den Commissioning-Modus wird auch in der Projektnavigation des TIA Portals als Overlay-Symbol vor dem Projektnamen angezeigt.

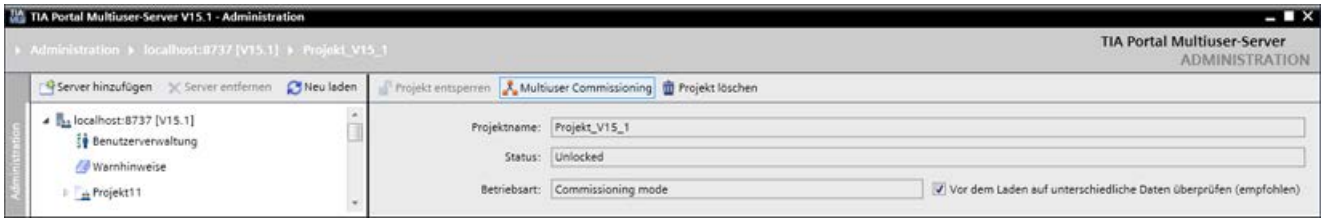

Im Commissioning-Modus wird nach dem Laden in die CPU immer das Serverprojekt aktualisiert.

Die Änderungen aus den lokalen Sessions werden vor dem Laden kompiliert und in die CPU geladen. Ist der Ladevorgang erfolgreich, so werden die Änderungen ins Serverprojekt transferiert. Auf dem Multiuser-Server wird dann jeweils eine neue Revision erzeugt und die lokale Session aktualisiert.

Falls der Multiuser-Server nicht erreichbar ist, erhalten Sie eine Fehlermeldung und der Commissioning-Modus wird beendet.

Sie haben dann die Möglichkeit, im Engineering-Modus aus der lokalen Session direkt in die CPU zu laden und ihre Änderungen nach dem erneuten Aufbau der Serververbindung ins Serverprojekt zu transferieren.

Beenden des Commissioning-Modus

Der Commissioning-Modus wird durch das Deaktivieren der Schaltfläche "Multiuser Commissioning" wieder beendet.

Klicken Sie dazu erneut auf die Schaltfläche Commissioning-Modus, um diese zu deaktivieren. Die deaktivierte Schaltfläche Commissioning-Modus bedeutet, dass der Engineering-Modus aktiv ist.

Im Engineering-Modus wird das kleine Overlay-Symbol beim Projektnamen nicht mehr angezeigt.

Automatisches Unterbrechen des Commissioning-Modus

In folgenden Fällen wird der Commissioning-Modus automatisch unterbrochen:

- Beim Auftreten von Fehlern beim Kompilieren der Änderungen.
- Beim Auftreten von Fehlern beim anschließenden Laden in die CPU.
- Beim Auftreten von Fehlern beim Transferieren der in die CPU geladenen Änderungen ins Serverprojekt.
- Beim Auftreten von Netzwerkunterbrechungen.
- Beim Offline arbeiten, wenn Sie ohne Verbindung zum Serverprojekt in der lokalen Session arbeiten.
- Wenn das Serverprojekt durch einen anderen Bearbeiter gesperrt ist.

Sie erhalten beim Auftreten von Fehlern eine entsprechende Fehlermeldung, die Sie über die Art des Fehlers und die mögliche Fehlerbehebung informiert.

3.4 Regeln für das Arbeiten mit Multiuser Commissioning

Einführung

Wenn Sie mit Multiuser Commissioning ein Projekt gemeinsam in Betrieb nehmen, ist es für eine erfolgreiche Zusammenarbeit sehr wichtig, dass alle Bearbeiter des Projektes folgende Regeln kennen.

Einstellen des Commissioning-Modus

Der Commissioning-Modus wird im Server Administration Tool eingestellt. Die Einstellung wird in der Regel vom Administrator vorgenommen und gilt für alle vorhandenen lokalen Sessions eines Serverprojektes. Dies bedeutet, dass entweder alle lokalen Sessions im Engineering-Modus arbeiten, oder dass alle lokalen Sessions im Commissioning-Modus arbeiten.

Siehe hierzu: Commissioning-Modus ein- und ausschalten (Seite 190)

Arbeiten mit dem Multiuser-Server im Commissioning-Modus

Sobald Sie im Commissioning-Modus das Laden in die CPU starten, wird zunächst versucht, eine Verbindung mit dem Multiuser-Server herzustellen, um die lokale Session auch im Multiuser-Serverprojekt zu aktualisieren. Kann der Multiuser-Server zu diesem Zeitpunkt nicht erreicht werden, öffnet sich ein Dialog und Sie erhalten eine entsprechende Nachricht.

Sie haben dann die Möglichkeit, durch Anklicken der Option im Dialog, in den Engineering-Modus zu wechseln und das Laden aus der lokalen Session in die CPU im Engineering-Modus durchzuführen.

Multiuser-Server ist nicht erreichbar im Commissioning-Modus

Wenn Sie im Commissioning-Modus arbeiten und der Multiuser-Server ist nicht mehr erreichbar, erhalten Sie eine entsprechende Meldung. Der sich öffnende Dialog bietet Ihnen folgende Möglichkeiten:

- Versuch, die Verbindung zum Multiuser-Server wieder herzustellen.
- Wechsel in den Engineering-Modus und Laden aus der lokalen Session in die CPU, ohne das Serverprojekt zu aktualisieren.
- Abbruch der Aktion.

Jede Option öffnet einen weiteren Dialog mit zusätzlichen Auswahl- und Abbruchmöglichkeiten.

"Offline" Arbeiten im Commissioning-Modus

Auch wenn Sie mit Multiuser Commissioning "Offline arbeiten", also ohne mit dem Multiuser-Server verbunden zu sein, können Sie aus der lokalen Session in die CPU laden.

In diesem Fall öffnet sich ebenfalls ein Dialog, in dem Sie in den Engineering-Modus wechseln und die lokale Session ohne Aktualisierung des Serverprojekts in die CPU laden können. Sie haben auch die Möglichkeit, den Vorgang abzubrechen.

Automatisches Unterbrechen des Commissioning-Modus

In folgenden Fällen wird der Commissioning-Modus automatisch vom System unterbrochen:

- Beim Auftreten von Fehlern beim Transferieren der Änderungen ins Serverprojekt.
- Beim Auftreten von Fehlern beim Kompilieren der Änderungen.
- Beim Auftreten von Netzwerkunterbrechungen.
- Beim Auftreten von Fehlern beim Laden in die CPU.
- Beim Offline arbeiten, wenn Sie ohne Verbindung zum Serverprojekt in der lokalen Session arbeiten.
- Wenn das Serverprojekt durch einen anderen Bearbeiter gesperrt ist.

Sie erhalten beim Auftreten von Fehlern eine entsprechende Fehlermeldung, die Sie über die Art des Fehlers und die mögliche Fehlerbehebung informiert.

3.5 Commissioning-Modus ein- und ausschalten

Einführung

Ab dem TIA Portal V15.1 können Sie beim Arbeiten mit Multiuser Engineering zwischen dem "Engineering-Modus" und dem "Commissioning-Modus" umschalten.

Der Engineering-Modus ist per Voreinstellung eingestellt.

Für das Arbeiten im "Commissioning-Modus" gelten die gleichen Voraussetzungen, wie für das Arbeiten im "Engineering-Modus".

Im "Engineering-Modus" bearbeiten Sie Ihre Multiuser-Projekte wie gewohnt und projektieren in den verschiedenen lokalen Sessions parallel Ihre Automatisierungsaufgabe.

Sobald Ihr Multiuser-Projekt reif für die Inbetriebnahme ist, schalten Sie in den "Commissioning-Modus" um.

Folgende Funktionalität ist im "Commissioning-Modus" verfügbar:

- Beim Laden werden die Änderungen aus den lokalen Sessions kompiliert und in die CPU geladen.
- Danach werden die Änderungen automatisch ins Serverprojekt transferiert und die lokale Session aktualisiert.
- Auf dem Server wird nach jedem Laden eine neue Revision erzeugt.
- Damit wird sicher gestellt, dass die Projektstände in der CPU und auf dem Multiuser-Server konsistent sind.

Hinweis

Falls der Multiuser-Server nicht erreichbar ist oder sonstige Feher eim Laden auftreten, erhalten Sie eine Fehlermeldung und der "Commissioning-Modus" wird beendet.

Sie haben dann die Möglichkeit, im "Engineering-Modus" aus der lokalen Session direkt in die CPU zu laden und ihre Änderungen nach dem erneuten Aufbau der Serververbindung ins Serverprojekt zu transferieren.

"Commissioning-Modus" ein- und ausschalten

Das Ein- und Ausschalten des "Commissioning-Modus" wird im Administration Tool durchgeführt. Das Einstellen des Modus gilt immer für alle lokalen Sessions eines Serverprojektes. Es können immer nur alle lokalen Sessions entweder im Engineering-Modus oder im Commissioning-Modus sein.

Siehe hierzu auch: Multiuser-Projekte und lokale Sessions verwalten (Seite 74)

Siehe hierzu auch: Multiuser-Server mit grafischem Administration Tool verwalten (Seite 65)

Gehen Sie vor wie folgt:

1. Rufen Sie das grafische Tool für die Multiuser-Server-Verwaltung auf.
   - Öffnen Sie das Startmenü, indem Sie auf den Befehl "Start" > "Alle Programme" > "Siemens Automation" klicken.
   - Klicken Sie auf den Eintrag "TIA Portal Multiuser" > "Multiuser Server V<x.y> Administration", um das Administration Tool zu öffnen.
2. Unter "Administration" finden Sie eine Auflistung der konfigurierten Serververbindungen, sobald Sie über "Serververbindung hinzufügen" eine Verbindung zum Multiuser-Server eingerichtet haben. Auf der rechten Seite erhalten Sie in einer Detailansicht zusätzliche Informationen zu den links angewählten Objekten.
3. Wählen Sie auf der linken Seite den Server aus und selektieren Sie das Projekt, für das Sie den Commissioning-Modus einstellen möchten.
4. Klicken Sie auf der rechten Seite in der Funktionsleiste auf die Schaltfläche "Multiuser Commissioning".
   - Sobald die Schaltfläche aktiv ist, wird sie blau hinterlegt angezeigt und das auf der linken Seite ausgewählte Projekt erhält das Commissioning-Symbol als Overlay.
   - Mit Commissioning-Modus wird auch die Option "Vor·dem·Laden·auf·unterschiedliche·Daten·überprüfen·(empfohlen)" aktiviert.

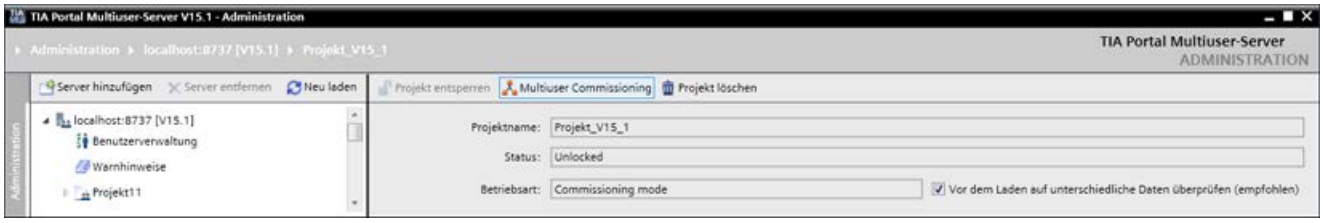

5. Um den "Commissioning-Modus" wieder zu beenden, klicken Sie erneut auf die Schaltfläche "Commissioning-Modus" in der Funktionsleiste.
   - Sobald die Schaltfläche deaktiviert ist, wird sie ohne farbige Hinterlegung dargestellt und für das auf der linken Seite ausgewählte Projekt wird kein Overlay-Smbol angezeigt.

Ergebnis

Der "Commissioning-Modus" wird durch das Klicken auf die Schaltfläche "Multiuser Commissioning" ein- und bei erneutem Klicken wieder ausgeschaltet.

Siehe auch

Multiuser-Server mit grafischem Administration Tool verwalten (Seite 65)

Multiuser-Projekte und lokale Sessions verwalten (Seite 74)

Projekte mit PLC-Synchronisierung in Betrieb nehmen 4

4.1 Grundlagen zur PLC-Synchronisierung

Einführung

Im Rahmen von Team Engineering haben Sie die Möglichkeit, Projekte mit Hilfe der PLC-Synchronisierung gemeinsam in Betrieb zu nehmen. Dabei können mehrere Bearbeiter parallel mit bis zu fünf Engineering Systemen (ES) gemeinsam auf eine CPU zugreifen.

Ein besonderer Vorteil ist es, dass in der Inbetriebnahmephase Teile eines Master-Projekts unabhängig voneinander gleichzeitig und parallel offline bearbeitet werden können. Beim Laden werden die Änderungen der anderen Bearbeiter in einem Dialog für die "Softwaresynchronisierung" angezeigt und - soweit möglich - automatisch synchronisiert.

Bestimmte Online-Funkionen können, abhängig vom Firmewarestand der verwendeten CPU, gleichzeitig und parallel von mehreren Engineering Systemen auf der gemeinsam genutzten CPU ausgeführt werden, wie z. B.:

- das Beobachten von Bausteinen auf der CPU
- das Steuern und Forcen von Bausteinen auf der CPU
- Tracefunktionen

Folgende Online-Funktionen können nicht gleichzeitig ausgeführt werden:

- Laden: Nur jeweils ein ES kann in die CPU laden.

Zusätzliche Informationen zum Thema Team Engineering finden Sie auch unter Service und Support im Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/de/82142829) .

Master-Projekt erstellen

Als Basis für die PLC-Synchronisierung im Rahmen der gemeinsamen Inbetriebnahme dient ein nach den vorgegebenen Regeln strukturiertes "Master-Projekt" (Seite 203), das bereits die vollständig projektierte Hardware-Konfiguration mit allen benötigten Variablen und Bausteinen enthält. Dieses Projekt wird in die gemeinsam genutzte CPU geladen und anschließend als "Master-Projekt" mittels Projektkopien an bis zu fünf beteiligte Engineering Systeme verteilt.

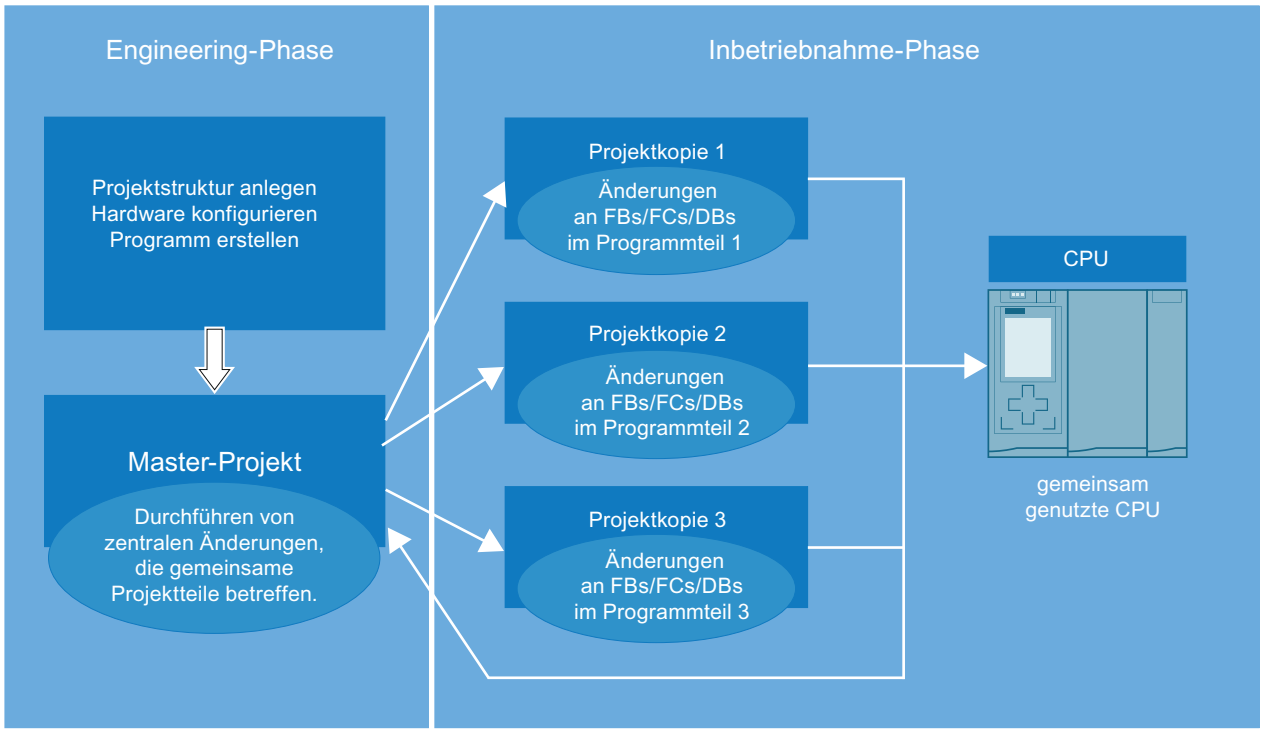

Arbeiten mit Projektkopien

Jedes Engineering System bearbeitet ausschließlich die ihm zugewiesenen Teile innerhalb der jeweiligen Projektkopie. Dies ist besonders wichtig, damit Konflikte beim Laden und ungewolltes Überschreiben der von anderen ES bereits modifizierten Bausteinen vermieden werden können.

Nach dem Bearbeiten lädt jedes ES seine Teile in die gemeinsam genutzte CPU. Beim Laden in die CPU werden Sie vom TIA Portal unterstützt. Im Dialog "Softwaresynchronisierung vor dem Laden in ein Gerät" (nachfolgend "Synchronisierungsdialog" genannt) werden durch einen Online-Offline-Vergleich die seit dem letzten Laden vorgenommenen Änderungen angezeigt. Hier erhalten Sie Vorschläge, wie Sie die von Ihnen bearbeiteten Offline-Daten beim Laden auf die CPU mit den dort vorhandenen Online-Daten synchronisieren können.

Sollte es so genannte "konkurrierende Änderungen" geben, so müssen diese manuell synchronisiert werden.

Konkurrierende Änderungen entstehen:

- Wenn verschiedene Bearbeiter in verschiedenen Projekt-Kopien parallel denselben Baustein bearbeiten.
- Wenn PLC-Variablen (E, A, M, T, Z) in den Projektkopien hinzugefügt, geändert oder gelöscht werden.
- Wenn in den Projektkopien Bausteine bearbeitet werden, die Beziehungen zur Hardware-Konfiguration haben.
- Wenn F-Bausteine in den Projektkopien bearbeitet werden.

Jeder Bearbeiter lädt seine Änderungen erst dann in die CPU, wenn alle Konflikte beseitigt sind.

Die Vorgehensweise beim Bearbeiten und Laden der Projektkopien kann von den einzelnen Engineering Systemen beliebig oft wiederholt werden, bis die PLC-Synchronisierung im Rahmen der gemeinsamen Inbetriebnahme abgeschlossen ist.

Projektkopien wieder in das Master-Projekt integrieren

Zum Abschluss der PLC-Synchronisierung ist es notwendig, die einzelnen Projektkopien wieder in das Master-Projekt zu integrieren. Damit sichern Sie auch die Daten, die ausschließlich in den Offline-Projekten vorhanden sind, also nicht auf die CPU geladen wurden. Dazu gehören z. B. die in den Projektkopien angelegten Projektsprachen und Textlisten.

Öffnen Sie dazu das Master-Projekt und jeweils eine Projektkopie als Referenzobjekt. Mit Hilfe des Vergleichseditors können Sie über einen Offline/Offline-Vergleich die Unterschiede zwischen beiden Projekten feststellen. Kopieren Sie die bearbeiteten Objekte aus der jeweiligen Projektkopie in das Master-Projekt. Führen Sie dies für alle vorhandenen Projektkopien durch.

Damit steht Ihnen zum Abschluss der gemeinsamen Inbetriebnahme über die PLC-Synchronisierung ein ablauffähiges Master-Projekt mit allen erzeugten Projektdaten zur Verfügung.

Die Integration der Projektkopien in das Master-Projekt ist auch dann notwendig, wenn Sie während der Inbetriebnahmephase feststellen, dass noch Erweiterungen oder Korrekturen an gemeinsam genutzten zentralen Elementen notwendig sind, wie z. B. eine Erweiterung der PLC-Variablentabelle oder das Hinzufügen einer neuen Hardware-Komponente.

Hinweise zur Kompatibilität

Für die PLC-Synchronisierung im Rahmen der gemeinsamen Inbetriebnahme gelten folgende Kompatibilitätsregeln:

- Wenn ein Engineering System mit einem TIA Portal V13 oder höher online auf der CPU ist und ein Laden durchführt, kann kein zweites ES, unabhängig von der Version, eine Online-Verbindung aufbauen.
- Wenn ein Engineering System mit einem TIA Portal < V13 online auf der CPU ist, kann kein zweites ES, unabhängig von der Version, eine Online-Verbindung aufbauen.

Team Engineering

Im Rahmen von Team Engineering haben Sie die Möglichkeit, Projekte gemeinsam in Betrieb zu nehmen. Dabei können mehrere Bearbeiter parallel und gleichzeitig mit mehreren Engineering Systemen (ES) gemeinsam auf eine CPU zugreifen.

PLC-Synchronisierung

Bei der PLC-Synchronisierung arbeiten mehrere Engineering Systeme im Rahmen von Team Engineering gemeinsam und parallel auf einer CPU.

Master-Projekt

Das Master-Projekt ist das Basisprojekt für die PLC-Synchronisierung. Es ist ein nach vorgegebenen Regeln strukturiertes Projekt, das bereits die vollständig projektierte Hardware-Konfiguration mit allen benötigten Variablen und Bausteinen enthält. Dieses Projekt wird in die gemeinsam genutzte CPU geladen und anschließend als "Master-Projekt" mittels Projektkopien an die beteiligten Engineering Systeme verteilt. Jedes Engineering System bearbeitet ausschließlich die ihm zugewiesenen Teile innerhalb der jeweiligen Projektkopie. Die Projektkopien werden anschließend wieder in das Master-Projekt integriert.

Projektkopie

Projektkopien werden beim Arbeiten mit Team Engineering aus strukturierten Master-Projekten erstellt und an die beteiligten Engineering Systeme zur Bearbeitung verteilt. Jedes Engineering System bearbeitet im Rahmen der PLC-Synchronisierung ausschließlich die ihm zugewiesenen Teile innerhalb der jeweiligen Projektkopie. Zum Abschluss der PLC-Synchronisierung werden die einzelnen Projektkopien wieder in das Master-Projekt integriert.

Siehe auch

Voraussetzungen für die PLC-Synchronisiserung (Seite 197)

Vorgehensweise bei der PLC-Synchronisierung (Seite 199)

4.2 Voraussetzungen für die PLC-Synchronisiserung

Software- und Hardwarevoraussetzungen

Um Projekte gemeinsam in Betrieb nehmen zu können, müssen die minimalen Software- und Hardwarevoraussetzungen für die Installation des TIA Portals V13 oder höher erfüllt sein.

Zusätzlich gibt es folgende Voraussetzungen:

Software:

- Sie haben das TIA Portal V13 oder höher und das Softwarepaket "SIMATIC STEP 7 Professional" auf den beteiligten Engineering Systemen installiert.
- Die Engineering Systeme müssen alle die gleiche Softwareversion installiert haben.
- Das Master-Projekt muss ein mit der V13 oder höher erstelltes Projekt sein, das auf die im Engineering System verwendete Version des TIA Portals aktualisiert wurde.
- Nach dem Umstieg von einer niedrigeren auf eine höhere TIA Portal-Version müssen alle beteiligten Engineering Systeme diese Version verwenden.
- Nach dem Umstieg von einer niedrigeren auf eine höhere TIA Portal-Version muss das Master-Projekt zentral angepasst und neue Projektkopien an die beteiligten Engineering Systeme übergeben werden.

Hardware:

- Sie verfügen über eine projektierte und konfigurierte CPU S7-1500 ab Firmwarestand V1.5.
- Die beteiligten Engineering Systeme können eine Online-Verbindung zu dieser CPU aufbauen.

Voraussetzungen für das gemeinsame, parallele Arbeiten

Um Projekte gemeinsam in Betrieb nehmen zu können, gibt es folgende Voraussetzungen:

- Sie haben ein Projekt mit vollständiger Hardware-Konfiguration und dem ausprogrammierten Anwenderprogramm erstellt, das in Betrieb genommen werden kann.
- Das Projekt wurde auf die CPU geladen und als "Master-Projekt" definiert.
- Mehrere Engineering Systeme haben Zugriff auf diese CPU und können eine Online-Verbindung zu dieser CPU aufbauen.
- Sie kennen die definierten Regeln und Vorgehensweisen für das gemeinsame Arbeiten auf einer CPU.

Hinweis zum Kompatibilitätsmodus

Die Funktionalität für die PLC-Synchronisierung steht im Kompatibilitätsmodus nicht zur Verfügung.

Siehe auch

Grundlagen zur PLC-Synchronisierung (Seite 193)
Vorgehensweise bei der PLC-Synchronisierung (Seite 199)
Regeln für die PLC-Synchronisierung (Seite 203)

4.3 Vorgehensweise bei der PLC-Synchronisierung

Einführung

Wenn Sie im Rahmen von Team Engineering ein Projekt gemeinsam in Betrieb nehmen, ist es sehr wichtig, dass sich alle Bearbeiter des Projektes an eine definierte Vorgehensweise halten.

Nur wenn die festgelegte Vorgehensweise eingehalten wird, ist sichergestellt, dass alle Änderungen und Korrekturen im Projekt automatisch synchronisiert und übernommen werden können und keine Änderungen einzelner Bearbeiter beim Laden ungewollt überschrieben werden oder bei konkurrierenden Änderungen sogar verloren gehen.

Vorgehensweise zum Erstellen des Master-Projektes

Das Master-Projekt wird in folgenden Schritten angelegt:

1. Erstellen Sie ein Master-Projekt, das bereits die komplette Projektstruktur enthält.
2. Konfigurieren Sie die Hardware für das Master-Projekt vollständig.
3. Definieren Sie eine Sprache als Projektsprache, die von allen beteiligten Engineering-Systemen ebenfalls ausschließlich verwendet werden muss.
4. Legen Sie alle Variablen und Bausteine an, die im Master-Projekt benötigt werden.
5. Erstellen Sie eigene Ordner und Gruppen für die von den einzelnen Engineering Systemen zu bearbeitenden Bausteine.
6. Erstellen Sie ein vollständig programmiertes und ablauffähiges Anwenderprogramm.
7. Laden Sie das Master-Projekt in die gemeinsam genutzte CPU.
8. Speichern Sie das Master-Projekt nach jedem Laden.

Beachten Sie hierzu auch die Regeln für die PLC-Synchronisierung (Seite 203)

Vorgehensweise bei der PLC-Synchronisierung

Die PLC-Synchronisierung wird in folgenden Schritten durchgeführt:

1. Laden Sie das Master-Projekt auf die gemeinsam genutzte CPU.
2. Erstellen Sie Kopien des Master-Projektes und verteilen Sie diese an die beteiligten Bearbeiter auf den zugehörigen Engineering Systemen.
3. Informieren Sie alle Bearbeiter darüber, wer welche Teile der Projektkopien bearbeiten und in die CPU laden darf.
4. Lassen Sie die Projektkopien in den einzelnen Engineering Systemen bearbeiten.
5. Nach dem Bearbeiten laden die einzelnen Engineering Systeme die Änderungen nacheinander in die CPU.

6. Beim Laden werden alle Änderungen über einen Online-Offline-Vergleich automatisch festgestellt. Im angezeigten Synchronisierungs-Dialog erhalten Sie Vorschläge für die Synchronisierung der geänderten Daten, soweit dies möglich ist. Möglicherweise müssen Sie zunächst Bausteine, die von anderen Bearbeitern geändert wurden, auf Ihr ES laden, bevor Sie Ihre Änderungen in die CPU laden können. Dies ist notwendig, um die Korrekturen der anderen Bearbeiter, die bereits in die CPU geladen wurden, in Ihrer Projektkopie zu aktualisieren.
   Folgende Synchronisierungsmöglichkeiten stehen zur Verfügung:
   - Baustein laden: Es gibt geänderte Bausteine auf der CPU, die in Ihrer Projektkopie aktualisiert werden müssen.
   - Baustein laden: Es gibt neue Bausteine auf der CPU, die in Ihre Projektkopie geladen werden müssen.
   - Bausteine mit konkurrierenden Änderungen: Eine systemunterstützte Synchronisierung ist in diesem Fall nicht möglich, der Konflikt muss manuell beseitigt werden.
7. Lösen Sie eventuell auftretende Konflikte auf Grund von konkurrierenden Änderungen oder Änderungen an zentralen Objekten manuell.
8. Laden Sie ihre Projektkopie in die CPU, wenn alle Konflikte beseitigt sind.
9. Speichern Sie Ihre Projektkopie nach jedem Laden.
10. Wiederholen Sie das Bearbeiten der Projektkopien und das Laden in die CPU beliebig oft, bis die PLC-Synchronisierung abgeschlossen ist.
11. Integrieren Sie die fertig bearbeiteten Projektkopien wieder in das Master-Projekt, um auch die nur offline vorhandenen Projektdaten zu sichern.

Beachten Sie hierzu auch die Regeln für die PLC-Synchronisierung (Seite 203).

Vorgehensweise zum manuellen Synchronisieren von konkurrierenden Änderungen

Das manuelle Synchronisieren von konkurrierenden Änderungen wird in folgenden Schritten durchgeführt:

1. Starten Sie den Vergleichseditor, um einen beim Laden in die CPU im Synchronisierungsdialog angezeigten Konflikt manuell zu beheben. Selektieren Sie dazu die gemeinsam genutzte CPU in der Projektnavigation und wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online". Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Beachten Sie dabei jedoch, dass Sie bei einer Synchronisierung immer nur Aktionen in eine Richtung ausführen können.
2. Wählen Sie zunächst die Aktion "Laden von Gerät" für alle Bausteine, die von anderen Bearbeitern auf der CPU geändert wurden, und die Sie übernehmen wollen.
3. Führen Sie gegebenenfalls einen Detailvergleich der Bausteine durch, um die Unterschiede zwischen Ihren Offline-Bausteinen und den auf der CPU geladenen Online-Bausteinen zu identifizieren.
4. Beheben Sie die konkurrierenden Änderungen an den Bausteinen manuell.
5. Laden Sie anschließend die betroffenen Bausteine mit dem Befehl "Ohne Synchronisierung fortfahren" in die CPU.
6. Speichern Sie Ihr Projekt bzw. Ihre Projektkopie nach jedem Laden.

Beachten Sie hierzu auch die Regeln für die PLC-Synchronisierung (Seite 203)

Hinweis

Um zu vermeiden, dass es zu erneuten Online-Änderungen kommt, während ein Engineering System gerade manuell synchronisiert, sollte während des manuellen Synchronisierens kein Laden auf die gemeinsam genutzte CPU durch ein anderes beteiligtes Engineering System vorgenommen werden.

Vorgehensweise zum Integrieren von Projektkopien in das Master-Projekt

Das Integrieren der Projektkopien in das Master-Projekt wird in folgenden Schritten durchgeführt:

1. Öffnen Sie das Master-Projekt und die zu integrierende Projektkopie als Referenzprojekt.
2. Kopieren Sie die von Ihnen bearbeiteten Teilprogramme aus der jeweiligen Projektkopie in das Master-Projekt und bestätigen Sie das Überschreiben der vorhandenen Objekte. Alternativ können Sie auch den Vergleichseditor verwenden, um Ihre Teilprogramme in das Master-Projekt zu übernehmen.
3. Speichern Sie das Master-Projekt und laden Sie es in die gemeinsam genutzte CPU mit dem Befehl "Ohne Synchronisierung fortfahren".
4. Speichern Sie das Master-Projekt nach jedem Laden.

Beachten Sie hierzu auch die Regeln für die PLC-Synchronisierung (Seite 203)

Vorgehensweise zur Modifizierung zentraler Objekte innerhalb des Master-Projektes

Die Modifizierung zentraler Objekte, die alle Programmteile betreffen, wird in folgenden Schritten durchgeführt:

1. Stoppen Sie zunächst die weitere Bearbeitung der Projektkopien.
2. Integrieren Sie nacheinander alle vorhandenen Projektkopien in das Master-Projekt, wie oben beschrieben.
3. Laden Sie das Master-Projekt in die gemeinsam genutzte CPU mit dem Befehl "Ohne Synchronisierung fortfahren".
4. Nehmen Sie die gewünschte Modifizierung an den gemeinsam genutzen Objekten wie z.B. innerhalb der Hardware-Konfiguration oder innerhalb der PLC-Variablentabelle vor. Für die Modifizierung anderer zentraler Objekte wie z.B. Technologieobjekte, F-Bausteine, OBs usw. ist die Vorgehensweise analog.
5. Laden Sie das Master-Projekt erneut in die CPU, wenn die Modifizierung abgeschlossen ist.
6. Speichern Sie das Master-Projekt nach jedem Laden.
7. Verteilen Sie die aktualisierten Projektkopien zur weiteren Bearbeitung an die betroffenen Engineering Systeme.

Beachten Sie hierzu auch die Regeln für die PLC-Synchronisierung (Seite 203)

Siehe auch

Grundlagen zur PLC-Synchronisierung (Seite 193)
Voraussetzungen für die PLC-Synchronisiserung (Seite 197)
Regeln für die PLC-Synchronisierung (Seite 203)

4.4 Regeln für die PLC-Synchronisierung

Einführung

Wenn Sie im Rahmen von Team Engineering ein Projekt gemeinsam über PLC-Synchronisierung in Betrieb nehmen, ist es für eine erfolgreiche Zusammenarbeit sehr wichtig, dass sich alle Bearbeiter des Projektes an vordefinierte Regeln halten.

Regeln für das Master-Projekt

Folgende Regeln sollten beachtet werden:

- Erstellen Sie ein Master-Projekt, das für die gemeinsame Bearbeitung geeignet ist.
- Unterteilen Sie das Anwenderprogramm in voneinander unabhängige Programmteile.
- Nutzen Sie "Gruppen", um die Programmteile auch visuell voneinander zu trennen.
- Nutzen Sie einen Main-OB und pro Programmteil einen zentralen FC, der die Funktionen des Teilprogramms aufruft.
- Legen Sie nach Möglichkeit für jedes Teil eine eigene PLC-Variablentabelle an.
- Legen Sie im Masterprojekt eine Projektsprache fest, die in den Projektkopien nicht verändert werden darf.
- Wenn Sie Daten zwischen den Programmteilen austauschen wollen, nutzen Sie dafür die Schnittstellenparameter von FCs und FBs (IN, OUT, INOUT) oder globale Datenbausteine.
- Nutzen Sie globale Datenbausteine für das Speichern der Daten für die einzelnen Programmteile, keine Merker.
- Vergeben Sie keine unterschiedlichen Namen für Bausteine mit identischen Adressen.
- Vergeben Sie keine identischen Namen für Bausteine mit unterschiedlichen Adressen.

Beispiel für die Programmstruktur im Master-Projekt

Nachfolgend sehen Sie ein Beispiel für ein teamfähig strukturiertes Master-Projekt, das für eine parallele Bearbeitung während der PLC-Synchronisierung geeignet ist.

Die von den einzelnen Engineering Systemen zu bearbeitenden Programmteile sind unterteilt in:

- Programmteil 1: "Conveyer"
- Programmteil 2: "Drill"
- Programmteil 3: "FurtherProgrammPart"

In jedem Programmteil befindet sich ein "Main-FB", der die jeweils untergeordneten Funktionen für diesen Programmteil aufruft.

Beispiel: Im Programmteil "Conveyer" ruft der "ConveyerMainFB" die "ConveyerStartupFC"-Funktion auf.

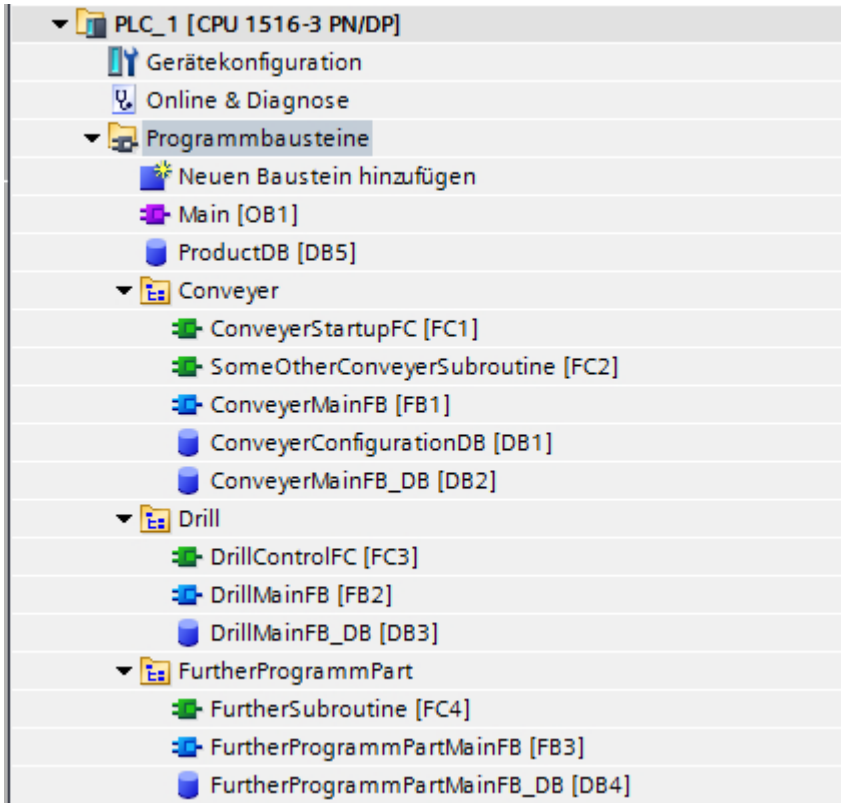

Durch eine solche Unterteilung innerhalb der Projektstruktur wird im Rahmen des Team Engineering die parallele Bearbeitung der einzelnen Programmteile und die automatische Synchronisierung der Änderungen während der gemeinsamen Inbetriebnahmephase ermöglicht.

Regeln für das gemeinsame Arbeiten auf einer CPU

Folgende Regeln für das gemeinsame Arbeiten sollten beachtet werden:

- Jeder Bearbeiter bearbeitet in seiner Projektkopie ausschließlich die ihm zugewiesenen Bausteine in den ihm zugewiesenen Gruppen.
- In den Projektkopien dürfen nur OBs, FBs, DBs, FCs und UDTs geändert werden, d.h. folgende Programmelemente dürfen in den Projektkopien **nicht** bearbeitet werden:
  - Hardware-Konfigurationen
  - PLC-Variablentabelle
  - Technologieobjekte
  - Textlisten und Projektsprache
  - F-Bausteine
- An Stelle von Merkern sollten globale Datenbausteine bei der Programmierung verwendet werden.

- An Stelle von SIMATIC-Zeiten und -Zählern sollten IEC-Zeiten und -Zähler bei der Programmierung verwendet werden.
- Die im Master-Projekt festgelegte Projektsprache darf in den Projektkopien nicht verändert werden.

Regeln für das Bearbeiten gemeinsam genutzter, zentraler Objekte

Folgende Regeln für gemeinsam genutzte zentrale Objekte sollten beachtet werden:

- Änderungen an gemeinsam genutzten zentralen Objekten dürfen ausschließlich im Master-Projekt vorgenommen werden.
  Zu diesen Objekten gehören:
  - Hardware-Konfigurationen
  - PLC-Variablentabelle
  - Technologieobjekte
  - Textlisten und Projektsprache
  - F-Bausteine
- Bevor die gemeinsam genutzten zentralen Objekte im Master-Projekt modizfiziert werden dürfen, müssen die unterschiedlichen Arbeitsstände innerhalb der Projektkopien zuerst wieder in das Master-Projekt integriert werden.
- Danach kann die Modifikation innerhalb des Master-Projektes vorgenommen werden und das Master-Projekt wieder in die gemeinsam genutzte CPU geladen werden.
- Wenn die Modifikation abgeschlossen ist, können neue Projektkopien erstellt und zur weiteren Bearbeitung an die Engineering Systeme verteilt werden.

Regeln für das Arbeiten mit Organisationsbausteinen

Folgende Regeln für das Bearbeiten von OBs sollten beachtet werden:

- OBs dürfen ausschließlich im Master-Projekt erstellt werden! Falls Sie in einer Projektkopie neue OBs erstellen, können diese **nicht** synchronisiert werden.
- OBs, die bereits im Master-Pojekt vorhanden sind und auch schon auf die CPU geladen wurden, können in den jeweiligen Projektkopien verändert werden. Beim Laden werden diese OBs automatisch synchronisiert.

Regeln für das Laden in die CPU

Folgende Regeln für das Laden sollten beachtet werden:

- Die Bearbeiter können die Projektkopien nacheinander in die CPU laden, es kann gleichzeitig immer nur ein Engineering System laden.
- Beim Laden von Bausteinen (Hardware darf nur im Masterprojekt geladen werden!) kann die Online-Verbindung der beteiligten Engineering Systeme bestehen bleiben.
- Führen Sie beim Laden die automatisch angebotene Sychronchronisierung über den Synchronisierungsdialog durch. Damit wird sichergestellt, dass Änderungen von anderen Bearbeitern nicht ungewollt überschrieben werden.

- Beheben Sie im Synchronisierungsdialog angezeigte Namenskonflikte dadurch, dass Sie Ihren Offline-Baustein vor dem Laden in die CPU umbenennen. Ein Namenskonflikt wird dann angezeigt, wenn Sie einen Baustein in die CPU laden wollen, der bereits mit identischem Namen auf der CPU vorhanden ist.
- Sogenannte "konkurrierende Änderungen" müssen manuell behoben werden. Konkurrierende Änderungen entstehen, wenn zwei Bearbeiter an zwei Engineering Systemen gleichzeitig denselben Baustein bearbeiten. Der erste Bearbeiter kann seine Änderungen noch ohne Probleme in die CPU laden. Will jedoch der zweite Bearbeiter den gleichen Baustein mit den von ihm vorgenommenen Änderungen ebenfalls in die CPU laden, so wird ihm im Synchronisierungsdialog ein Konflikt angezeigt, der nicht automatisch behoben werden kann. Beim Laden würden die vorherigen Änderungen an diesem Baustein durch Überschreiben wieder rückgängig gemacht werden. In diesem Fall muss von den Bearbeitern entschieden werden, welche Änderung übernommen und welche verworfen wird. Alternativ können Sie die Änderungen mit Hilfe des Detailvergleichs der Bausteine im Vergleichseditor manuell zusammenführen.
  In jedem Fall sollten derartige Änderungen durch eine gute Projektstrukturierung und entsprechende Absprachen unter den Bearbeitern unbedingt vermieden werden.
- Änderungen an zentralen Objekten werden im Synchronisierungsdialog angezeigt, können jedoch nicht automatisch synchronisert werden. Änderungen an zentralen Objekten und an der Hardware-Konfiguration müssen immer über das Master-Projekt vorgenommen werden.
- Speichern Sie Ihr Projekt bzw. Ihre Projektkopie nach jedem Laden.

Hinweis

Anlegen und anschließendes Löschen von Objekten in den Projektkopien

Beachten Sie, dass auch dann eine Software-Synchronisierung vor dem Laden notwendig wird, wenn Sie ein neues Objekt in einer Projektkopie anlegen und dieses anschließend gleich wieder löschen.

Das Anlegen eines Objektes bedingt eine Veränderung in der internen Datenhaltung, die durch das anschließende Löschen des Objektes nicht rückgängig gemacht wird. Deshalb besteht auch dann Synchronisierungsbedarf vor dem nächsten Laden, wenn das neu angelegte Objekt gleich wieder gelöscht wurde und keine Veränderung bei den vorhandenen Objekten innerhalb der Projektkopie sichtbar ist.

Regeln für Online-Funktionen

Folgende Regeln beim Nutzen von Online-Funktionen sollten beachtet werden:

Laden:

- Es kann immer nur ein Engineering System Daten auf die CPU laden.
- Während des Ladens können andere Engineering Systeme zusätzliche Online-Funktionen nutzen, wie z. B. Beobachten und Steuern.

Beobachten und Steuern:

- Bis zu zehn Engineering Systeme können gleichzeitig auf der CPU bis zu 16 verschiedene Bausteine beobachten und steuern (CPU 1517/1518). Weitere Details finden Sie in den technischen Daten der CPU.
- Ein spezifischer Code-Baustein kann gleichzeitig immer nur von einem Engineering System beobachtet und gesteuert werden.
- Weitere Engineering Systeme können jedoch gleichzeitig jeweils andere Code-Bausteine beobachten und steuern.

ACHTUNG
Gefahr durch paralleles Steuern eines identischen Operanden mit unterschiedlichen Steuerwerten in mehreren Beobachtungstabellen Beim Arbeiten mit mehreren Beobachtungstabellen ist darauf zu achten, identische Operanden nicht mehrfach mit unterschiedlichen Steuerwerten permanent zu steuern! Wenn ein identischer Operand in mehreren Beobachtungstabellen gleichzeitig mit unterschiedlichen Steuerwerten permanent gesteuert wird, so wird in allen Beobachtungstabellen als Beobachtungswert der zuletzt gesteuerte Wert angezeigt, da in diesem Fall immer der zuletzt zugewiesene Steuerwert gesteuert wird.

Forcen

- Es kann jeweils nur von einem Engineering System aus auf der CPU geforct werden.
- Das Engineering System, das das Forcen startet, hat den Forceauftrag exklusiv übernommen. Die anderen Engineering Systeme erhalten zwar die Information, dass ein Forceauftrag läuft, können auf diesen Auftrag jedoch nicht verändernd zugreifen. Über den Befehl "Geforcte Operanden aktualisieren" können die aktuell auf der CPU geforcten Operanden und Werte in der geöffneten Forcetabelle aktualisiert werden. In der geöffneten Forcetabelle werden alle geforcten Operanden mit den entsprechenden Werten aktualisiert. In der ersten Spalte wird ein rotes "F" eingeblendet, das anzeigt, welche Operanden geforct werden.
- Sobald das Engineering System, dem der Force-Auftrag gehört, die Online-Verbindung beendet, kann der Forceauftrag von einem anderen ES übernommen werden, das eine Online-Verbindung zur CPU aufbaut und den Befehl "Geforcte Operanden aktualisieren" ausgeführt hat. Damit werden die Schaltflächen "Alle Forcen"und "Forcen beenden" freigeschaltet und Sie können diese Funktionen ausführen.

Trace-Funktionen:

- Bis zu vier Engineering Systeme können gleichzeitig Trace-Funktionen durchführen.
- Das Engineering System, das das Tracen startet, hat den jeweiligen Traceauftrag exklusiv übernommen. Die anderen Engineering Systeme können diesen Auftrag im Trace-Editor zwar sehen, aber nicht darauf zugreifen.
- Sobald das Engineering System, dem der Trace-Auftrag gehört, den Trace-Editor beendet, kann der Traceauftrag von einem anderen ES übernommen werden, das den Trace-Editor wieder öffnet.

Testen mit Haltepunkten:

Sobald durch ein Engineering System einer CPU S7-1500 ab Firmwarestand V2.5 ein Haltepunkt aktiviert wird, gelten folgende Regeln:

- Das Aktivieren von Haltepunkten durch andere Engineering Systeme ist gesperrt.
- Das Aktivieren von zusätzlichen Haltepunkten auf dem ausführenden Engineering System ist erlaubt, auch wenn sich die CPU nach Erreichen eines Haltepunktes im Betriebszustand "HALT" befindet.
- Von einem anderen Engineering System kann kein "Laden in Gerät" durchgeführt werden und das Programm auf der CPU kann nicht weiter ausgeführt werden, da sich die CPU bei Erreichen eines Haltepunktes im Betriebszustand "HALT" befindet. Die Programmausführung wird erst dann fortgesetzt, wenn die CPU durch das ausführende Engineering System vom Betriebszustand "HALT" wieder in den Betriebszustand "RUN" gesetzt wird. Andere Engineering Systeme können den Betriebszustand nach Erreichen eines Haltepunktes nicht direkt nach "RUN" umschalten. Die CPU muss zunächst in den Betriebszustand "STOP" versetzt werden und kann erst danach in "RUN" geschaltet werden.
- Während bereits ein "Laden in Gerät" durchgeführt wird, können während des Ladevorgangs keine Haltepunkte aktiviert werden.
- Das "Laden in Gerät" ist auch im Betriebszustand "HALT" möglich, wenn keine Haltepunkte aktiviert sind. Während des Ladens wird die CPU dann in den Betriebszustand "STOP" gesetzt.

ACHTUNG
Haltepunkte aktivieren auf einer CPU S7-1500 ab Firmwarestand V2.5 Sobald Sie auf einer CPU S7-1500 ab V2.5 einen Haltepunkt aktivieren, erhalten Sie die Information, dass die Ausgangswerte für die angeschlossen Ausgangsbaugruppen in ihrem jeweiligen Zustand gehalten und (im Gegensatz zur SIMATIC S7-300/S7-400) weder zurückgesetzt noch auf die projektierten Ersatzwerte gesetzt werden. Darüber hinaus kann die SIMATIC S7-1500 in diesem Zustand nicht mehr auf Eingangswerte von Eingangsbaugruppen reagieren. Diese Information wird Ihnen in jeder Online-Sitzung nur einmal angezeigt.

Siehe auch

Grundlagen zur PLC-Synchronisierung (Seite 193)

Voraussetzungen für die PLC-Synchronisiserung (Seite 197)

Vorgehensweise bei der PLC-Synchronisierung (Seite 199)

Daten mit Inter Project Engineering (IPE) austauschen 5

5.1 Grundlagen zu Inter Project Engineering (IPE)

Einführung in Inter Project Engineering (IPE)

Mit der Funktionalität von Inter Project Engineering, nachfolgend IPE genannt, können Sie in einem Quellprojekt vorhandene Steuerungsdaten mit Hilfe eines Geräte-Proxy projektübergreifend austauschen.

Diese Daten können Sie in anderen Projekte, z. B. für die Visualisierung in HMI, übernehmen und dort zur weiteren Projektierung verwenden.

Inter Project Engineering ermöglicht Ihnen somit auf einfache Art und Weise projektübergreifend Steuerungsdaten von einer PLC-Programmierung für eine HMI-Projektierung zur Verfügung zu stellen.

Mit Hilfe des Objektes "Geräte-Proxy-Daten" führen Sie einen konsistenten und komfortablen projektübergreifenden Datenaustausch der Steuerungsdaten durch, ohne redundanten Projektierungssaufwand.

Dadurch wird es für den HMI-Projekteur möglich, an einem Projekt zu arbeiten, ohne dass die Hardware-Konfiguration in seinem Projekt vorhanden sein muss. Dadurch können die Arbeiten am PLC-Projekt und am HMI-Projekt parallel und zeitgleich durchgeführt werden.

Folgende Steuerungsdaten können Sie über den Geräte-Proxy zwischen den Projekten austauschen:

- Programmbausteine
- Technologieobjekte
- PLC-Variablen
- PLC-Meldungen

Folgende Daten werden beim Austausch automatisch mit übertragen:

- Schnittstellen der Steuerung
- Projektierte Kommunikationsprozessoren und -module

Durch diesen automatischen Datenaustausch wird sicher gestellt, dass zusätzlich zu den von Ihnen ausgewählten Daten auch die Schnittstellen und projektierten Kommunikationsprozessoren und -module mit übergeben werden. Diese Daten werden benötigt, um eine problemlose und konsistente Weiterarbeit in Ihrem Zielprojekt zu ermöglichen.

Projektübergreifender Datenaustausch

Für den projektübergreifenden Datenaustausch mit Inter Project Engineering haben Sie folgende Möglichkeiten:

- Steuerungsdaten über IPE-Datei austauschen
- Steuerungsdaten über Projektdatei austauschen

Steuerungsdaten über IPE-Datei austauschen

Das nachfolgende Bild zeigt den projektübergreifenden Datenaustausch über eine IPE-Datei.

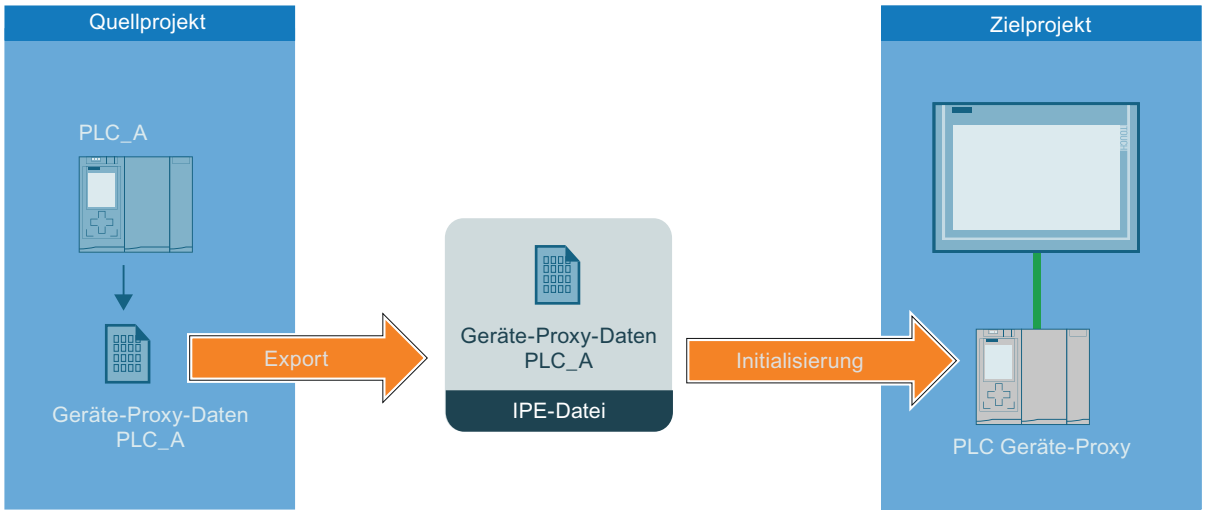

Die Steuerungsdaten der PLC aus dem Quellprojekt werden über das Objekt "Geräte-Proxy-Daten" an die PLC im Zielprojekt übergeben.

Steuerungsdaten, die über eine IPE-Datei ausgetauscht werden, müssen von TIA Portal-Projekten V13 oder höher stammen.

Steuerungsdaten über eine Projektdatei austauschen

Das nachfolgende Bild zeigt den projektübergreifenden Datenaustausch über eine Projektdatei.

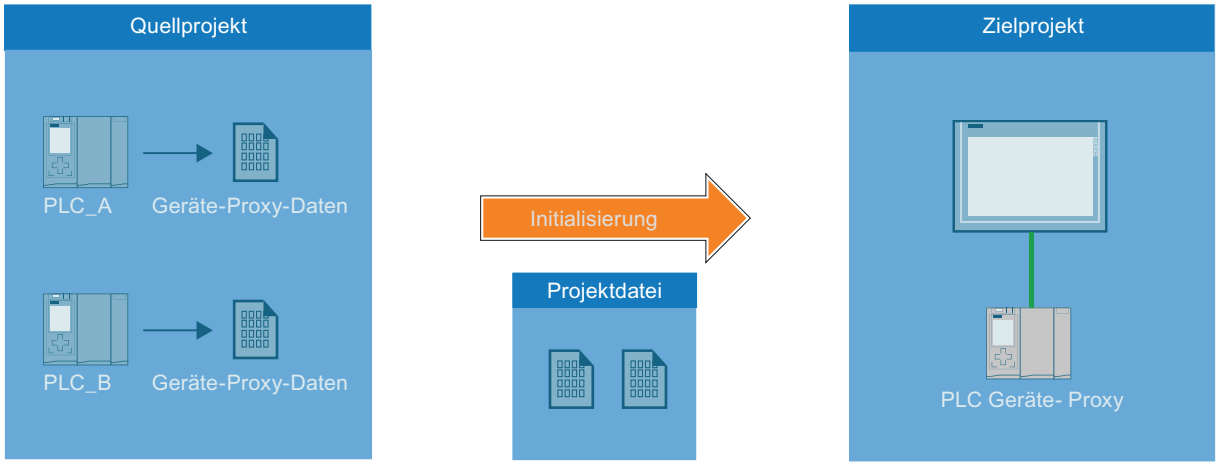

Die Steuerungsdaten der PLC aus dem Quellprojekt werden über eine Projektdatei an die PLC im Zielprojekt übergeben.

Sie können sowohl aus älteren TIA Portal-Projekten als auch aus STEP 7-Classic-Projekten, die nicht im TIA Portal integriert sind, Steuerungsdaten komfortabel in Ihre aktuellen TIA Portal-Projekte übernehmen und weiter verwenden.

Inter Project Engineering (IPE)

Mit der Funktionalität von Inter Project Engineering, auch "IPE" genannt, können Sie in einem Quellprojekt vorhandene Steuerungsdaten mit Hilfe des Objektes "Geräte-Proxy-Daten" projektübergreifend austauschen.

IPE

Mit der Funktionalität von Inter Project Engineering, auch "IPE" genannt, können Sie in einem Quellprojekt vorhandene Steuerungsdaten mit Hilfe des Objektes "Geräte-Proxy-Daten" projektübergreifend austauschen.

IPE-Datei

Die IPE-Datei enthält Steuerungsdaten der CPU aus einem Quellprojekt, die über das Objekt "Geräte-Proxy-Daten" an die CPU im Zielprojekt übergeben werden.

Projektdatei

Die Projektdatei übergibt Steuerungsdaten der CPU aus einem Quellprojekt an die CPU im Zielprojekt. Diese Steuerungsdaten können sowohl aus älteren TIA Portal-Projekten als auch aus STEP 7-Classic-Projekten, die nicht im TIA Portal integriert sind, Steuerungsdaten komfortabel in Ihre aktuellen TIA Portal-Projekte übernehmen und weiter verwenden.

Geräte-Proxy-Daten

Das Objekt "Geräte-Proxy-Daten" ermöglicht einen konsistenten projektübergreifenden Datenaustausch von Steuerungsdaten der CPU zwischen einem Quell- und einem Zielprojekt, ohne redundanten Projektierungssaufwand.

Projektübergreifender Datenaustausch mit IPE

Mit der Funktionalität von Inter Project Engineering (IPE) haben Sie folgende Möglichkeiten für einen konsistenten projektübergreifenden Datenaustausch:

- Steuerungsdaten über eine IPE-Datei austauschen
- Steuerungsdaten über eine Projektdatei austauschen

Beide Dateien übergeben Steuerungsdaten der CPU aus einem Quellprojekt an eine CPU im Zielprojekt.

Siehe auch

Voraussetzungen für Inter Project Engineering (IPE) (Seite 213)

Übersicht für das Arbeiten mit Inter Project Engineering (IPE) (Seite 214)

5.2 Voraussetzungen für Inter Project Engineering (IPE)

Software- und Hardwarevoraussetzungen

Damit Sie die Funktionalität von Inter Project Engineering nutzen können, müssen die minimalen Software- und Hardwarevoraussetzungen für die Installation des TIA Portals V13 oder höher erfüllt sein.

Zusätzlich gibt es folgende Voraussetzungen:

- Sie haben das TIA Portal V13 oder höher und das Softwarepaket "SIMATIC STEP 7 Professional" installiert.
- Sie verwenden eine CPU aus der Familie S7-1200/1500 oder S7-300/400.
- Die beteiligten Steuerungen müssen die gleiche Softwareversion installiert haben.

Voraussetzungen für IPE

Um Steuerungsdaten aus Projekten übernehmen zu können, gibt es folgende Voraussetzungen:

- Sie haben ein Projekt mit Hardware-Konfiguration und PLC-Daten erstellt, die in ein anderes Projekt übertragen werden können.
- Sie haben sicher gestellt, dass das Projekt in sich konsistent ist, bevor Sie die Steuerungsdaten über das Objekt "Geräe-Proxy-Daten" weitergeben.
- Sie kennen die definierten Vorgehensweisen für Inter Project Engineering.

Hinweise zum Kompatibilitätsmodus

Die Funktionalität für Inter Project Engineering steht im Kompatbilitätsmodus nicht zur Verfügung.

Siehe auch

Grundlagen zu Inter Project Engineering (IPE) (Seite 209)

Übersicht für das Arbeiten mit Inter Project Engineering (IPE) (Seite 214)

5.3 Übersicht für das Arbeiten mit Inter Project Engineering (IPE)

Steuerungsdaten projektübergreifend austauschen

Mit der Funktionalität von Inter Project Engineering können Sie bereits vorhandene Steuerungsdaten mit Hilfe eines "Geräte-Proxy" projektübergreifend austauschen und in einem anderen Projekt weiter verwenden. Der "Geräte-Proxy" ermöglich Ihnen einen konsistenten und komfortablen projektübergreifenden Datenaustausch ohne redundanten Projektierungsaufwand.

Vorgehensweise zum Austausch von Steuerungsdaten über eine IPE-Datei

Zum Datenaustausch über eine IPE-Datei befolgen Sie folgende Schritte:

1. Erstellen Sie mit dem TIA Portal ein Projekt mit allen benötigten Steuerungsdaten und einem lauffähigen Anwenderprogramm.
2. Konfigurieren Sie die für das Projekt benötigte Hardware.
3. Übersetzen Sie das Projekt, um die Konsistenz in Ihrem Projekt sicher zu stellen.
4. Erzeugen Sie in Ihrem Projekt unterhalb der gewünschten CPU das Objekt "Geräte-Proxy-Daten", indem Sie im Ordner "Geräte-Proxy-Daten" auf den Befehl "Geräte-Proxy-Daten hinzufügen" klicken.
   Ergebnis: Das Objekt "Geräte-Proxy-Daten" wurde erstellt.
5. Selektieren Sie den gewünschten Geräte-Proxy-Daten in der Projektnavigation und öffnen Sie das Objekt im Editor durch Doppelklick.
6. Tragen Sie im Editor unter "Allgemein" die gewünschten Daten für den Geräte-Proxy-Daten ein oder übernehmen Sie die Voreinstellung. Sie haben die Option, den Namen zu ändern und einen Kommentar für den Export der Steuerungsdaten einzutragen.
   Wählen Sie im Editor im Bereich "Inhalt definieren" aus, welche der vorhandenen Steuerungsdaten Sie als IPE-Datei exportieren möchten und klicken Sie auf den Befehl "Geräte-Proxy-Daten exportieren".
7. Tragen Sie in nachfolgenden Dialog den Namen und den Speicherort für die zu erstellende IPE-Datei ein und klicken Sie auf "Speichern". Sie erhalten eine Meldung, sobald der Export erfolgreich durchgeführt wurde.
   Ergebnis: Die ausgewählten Steuerungsdaten wurden als IPE-Datei exportiert.
8. Erzeugen Sie in Ihrem Zielprojekt einen Geräte-Proxy, um die in der IPE-Datei vorhandenen Steuerungsdaten importieren zu können. Klicken Sie dazu auf den Befehl "Neues Gerät hinzufügen" und wählen Sie im nachfolgenden Dialog den gewünschten Geräte-Proxy aus.
9. Klicken Sie auf den neu erstellten Geräte-Proxy und wählen Sie im Kontextmenü den Befehl "Geräte-Proxy initialisieren".
10. Wählen Sie die vorab erzeugte IPE-Datei aus, die für die Initialisierung verwendet werden soll und bestätigen Sie die Auswahl im nachfolgenden Dialog mit "OK". Daraufhin wird die Initialisierung gestartet.
    Ergebnis: Die in der IPE-Datei enthaltenen Steuerungsdaten aus dem Quellprojekt werden auf den Geräte-Proxy im Zielprojekt übertragen.

Nach erfolgreichem Austausch der Steuerungsdaten können Sie nun mit der Projektierung in Ihrem Zielprojekt fortfahren und z. B. PLC-Variablen mit HMI-Variablen verbinden. Bei Änderung der Steuerungsdaten im Quellprojekt können diese beliebig oft aktualisiert werden.

Vorgehensweise zum Austausch von Steuerungsdaten über eine Projekt-Datei

Beim Austausch von Steuerungsdaten über eine Projektdatei können Geräte-Proxy-Daten entweder im Quellprojekt bereits vorderfiniert sein oder auch vom Zielprojekt aus definiert und ausgewählt werden.

Zum Datenaustausch über eine Projekt-Datei befolgen Sie folgende Schritte:

1. Erstellen Sie mit dem TIA Portal ein Projekt mit allen benötigten Steuerungsdaten und einem lauffähigen Anwenderprogramm.
2. Konfigurieren Sie die für das Projekt benötigte Hardware.
3. Übersetzen Sie das Projekt, um die Konsistenz in Ihrem Projekt sicher zu stellen.
4. Öffnen Sie das TIA Portal-Projekt , aus dem Sie die Steuerungsdaten ins Zielprojekt übernehmen wollen.
5. Erzeugen Sie in Ihrem Projekt unterhalb der gewünschten CPU das Objekt "Geräte-Proxy-Daten", indem Sie im Ordner "Geräte-Proxy-Daten" auf den Befehl "Geräte-Proxy-Daten hinzufügen" klicken.
   Ergebnis: Das Objekt "Geräte-Proxy-Daten" wurde erstellt.
6. Tragen Sie im Editor unter "Allgemein" die gewünschten Daten für den Geräte-Proxy-Daten ein oder übernehmen Sie die Voreinstellung. Sie haben die Option, den Namen zu ändern und einen Kommentar für den Geräte-Proxy-Daten einzutragen.
7. Wählen Sie im Editor im Bereich "Inhalt definieren" aus, welche der vorhandenen Steuerungsdaten Sie durch den Geräte-Proxy-Daten zur Verfügung stellen möchten.
8. Klicken Sie auf "Speichern", um die Änderungen im Projekt zu sichern.
   Ergebnis: Sie haben die ausgewählten Steuerungsdaten erfolgreich in Ihrem Geräte-Proxy-Daten gespeichert.
9. Erzeugen Sie in Ihrem Zielprojekt einen Geräte-Proxy, um die in der Projektdatei vorhandenen Steuerungsdaten importieren zu können. Klicken Sie dazu auf den Befehl "Neues Gerät hinzufügen" und wählen Sie im nachfolgenden Dialog den gewünschten Geräte-Proxy aus.
10. Klicken Sie auf den neu erstellten Geräte-Proxy und wählen Sie im Kontextmenü den Befehl "Geräte-Proxy initialisieren".
11. Wählen Sie die vorab erzeugte Projektdatei mit dem vorbereiteten Geräte-Proxy-Daten aus, die für die Initialisierung verwendet werden sollen, und bestätigen Sie die Auswahl im nachfolgenden Dialog mit "OK". Daraufhin wird die Initialisierung gestartet.
    Ergebnis: Die in der Projektdatei enthaltenen Steuerungsdaten aus dem Quellprojekt werden auf den Geräte-Proxy im Zielprojekt übertragen.

Nach erfolgreichem Austausch der Steuerungsdaten können Sie nun mit der Projektierung in Ihrem Zielprojekt fortfahren und z. B. PLC-Variablen mit HMI-Variablen verbinden. Bei Änderung der Steuerungsdaten im Quellprojekt können diese beliebig oft aktualisiert werden.

Vorgehensweise zum Aktualisieren von bereits übertragenen Steuerungsdaten

Voraussetzungen für die Aktualisierung:

Für die Aktualisierung bereits übertragener Steuerungsdaten mit Hilfe des bereits angelegten Objektes "Geräte-Proxy-Daten" müssen folgende Voraussetzungen erfüllt sein:

- Sie verwenden das gleiche Projekt für die Aktualisierung, wie bei der vorherigen Übertragung der Steuerungsdaten.
- Sie verwenden für die Übertragung der Steuerungsdaten den bereits angelegten "Geräte-Proxy-Daten".
- Sie haben zwischenzeitlich keine Veränderung an der Hardware-Konfiguration und den Kommunikationsschnittstellen vorgenommen. Das Hinzufügen neuer Hardware-Komponenten ist erlaubt.

Vorgehensweise zum Aktualisieren:

1. Öffnen Sie das TIA Portal-Projekt, aus dem Sie die Steuerungsdaten im Zielprojekt aktualisieren wollen.
2. Doppelklicken Sie auf das bereits vorhandene Objekt "Geräte-Proxy-Daten" und wählen Sie unter Inhalt aus, welche Steuerungsdaten Sie für die Aktualisierung zur Verfügung stellen möchten.
3. Speichern Sie das Projekt bzw. exportieren Sie die IPE-Datei erneut.
4. Öffnen Sie im TIA Portal das gewünschte Zielprojekt.
5. Klicken Sie im Zielprojekt auf den bereits vorhandenen Geräte-Proxy und wählen Sie im Kontextmenü den Befehl "Geräte-Proxy aktualisieren" aus.
6. Wählen Sie die Quelle aus, die für die Aktualisierung verwendet werden soll, und bestätigen Sie die Auswahl.
7. Wählen Sie im nachfolgenden Dialog unter "Gerät" die vorher als Quell-CPU verwendete CPU aus, von der die Steuerungsdaten erneut importiert werden sollen.
8. Wählen Sie unter "Definierte Geräte-Proxy-Daten" das gewünschte Objekt aus und verlassen Sie den Dialog mit "OK".
9. Daraufhin wird die Aktualisierung gestartet und die ausgewählten Geräte-Proxy-Daten werden aus dem Quellprojekt auf den Geräte-Proxy im Zielprojekt übertragen.

Nach erfolgreicher Aktualisierung der Steuerungsdaten können Sie nun mit der Projektierung in Ihrem Zielprojekt fortfahren und z. B. PLC-Variablen mit HMI-Variablen verbinden. Bei Änderung der Steuerungsdaten im Quellprojekt können diese beliebig oft aktualisiert werden.

Siehe auch

Grundlagen zu Inter Project Engineering (IPE) (Seite 209)

Voraussetzungen für Inter Project Engineering (IPE) (Seite 213)

5.4 Geräte-Proxy-Daten im Quellprojekt anlegen

Einführung

Um Steuerungsdaten projektübergreifend austauschen zu können, benötigt man im Quellprojekt das Objekt "Geräte-Proxy-Daten", das die Steuerungsdaten der zugeörigen CPU als IPE-Datei übertragen kann.

"Geräte-Proxy-Daten"-Objekt im Quellprojekt anlegen

Um ein neues Objekt "Geräte-Proxy-Daten" anzulegen, gehen sie vor wie folgt:

1. Öffnen Sie den Ordner "Geräte-Proxy-Daten" unterhalb der CPU, zu der Sie einen "Geräte-Proxy-Daten" anlegen möchten.
2. Klicken Sie auf den Befehl "Neuen Geräte-Proxy-Daten hinzufügen".

Ergebnis

Das Objekt "Geräte-Proxy-Daten" wurde erstellt und wird in der Projektnavigation unterhalb der zugehörigen CPU angezeigt.

Der erzeugte "Geräte-Proxy-Daten" enthält die Steuerungsdaten der zugehörigen CPU.

5.5 IPE-Datei mithilfe des "Geräte-Proxy-Daten" erzeugen

Geräte-Proxy-Daten öffnen und Inhalt für IPE-Datei definieren

Um eine IPE-Datei mithilfe eines "Geräte-Proxy-Daten" zu erzeugen, gehen Sie vor wie folgt:

1. Öffnen Sie im Ordner "Geräte-Proxy-Daten" in der Projektnavigation unterhalb der jeweiligen CPU den gewünschten "Geräte-Proxy-Daten" durch Doppelklick.
2. Tragen Sie im Editor unter "Allgemein" die gewünschten Einstellungen für das Objekt "Geräte-Proxy-Daten" ein oder übernehmen Sie die Voreinstellung. Sie haben die Option, den Namen zu ändern und einen Kommentar für den Export der Steuerungsdaten einzutragen.
3. Wählen Sie im Editor im Bereich "Inhalt definieren" aus, welche der vorhandenen Steuerungsdaten der CPU Sie als IPE-Datei exportieren möchten und klicken Sie auf den Befehl "Geräte-Proxy-Daten exportieren".
4. Tragen Sie in nachfolgenden Dialog den Namen und den Speicherort für die zu erstellende IPE-Datei ein und klicken Sie auf "Speichern". Sie erhalten eine Meldung, sobald der Export erfolgreich durchgeführt wurde.

Ergebnis

Die ausgewählten Steuerungsdaten wurden als IPE-Datei exportiert.

5.6 Mit IPE Steuerungsdaten aus anderen Projekten verwenden

5.6.1 Steuerungsdaten aus anderen Projekten im Bediengerät verwenden

5.6.1.1 Grundlagen zu Steuerungsdaten

Einleitung

Im Quellprojekt der Steuerung können Sie auswählen, welche Steuerungsdaten Sie als Geräte-Proxy-Daten ablegen.

Dazu erzeugen Sie aus einer projektierten Steuerung Geräte-Proxy-Daten.

Welche Steuerungsdaten können ausgetauscht werden?

Über IPE können Sie folgende Steuerungsdaten austauschen:

- Datenbausteine
- Technologieobjekte
- PLC-Variablen
- PLC-Überwachungen und PLC-Meldungen

Folgende Informationen werden automatisch ausgetauscht:

- Schnittstellen der Steuerung
- Projektierte Kommunikationsprozessoren und Kommunikationsmodule an der Steuerung

5.6.1.2 Geräte-Proxy über IPE-Datei initialisieren

Initialisierung von Geräte-Proxys über IPE-Datei

Wie funktioniert die Initialisierung eines Geräte-Proxys über IPE-Datei?

Steuerungsdaten können Sie zwischen zwei oder mehr Projekten mithilfe von IPE-Dateien austauschen.

IPE-Dateien können Sie auf Grund des kleines Datenvolumens z. B. per E-Mail versenden.

Sie können pro Steuerung mehrere IPE-Dateien mit verschiedenen Geräte-Proxy-Daten erzeugen.

Im Zielprojekt kann ein Geräte-Proxy nur mit einem Geräte-Proxy-Datensatz gefüllt werden.

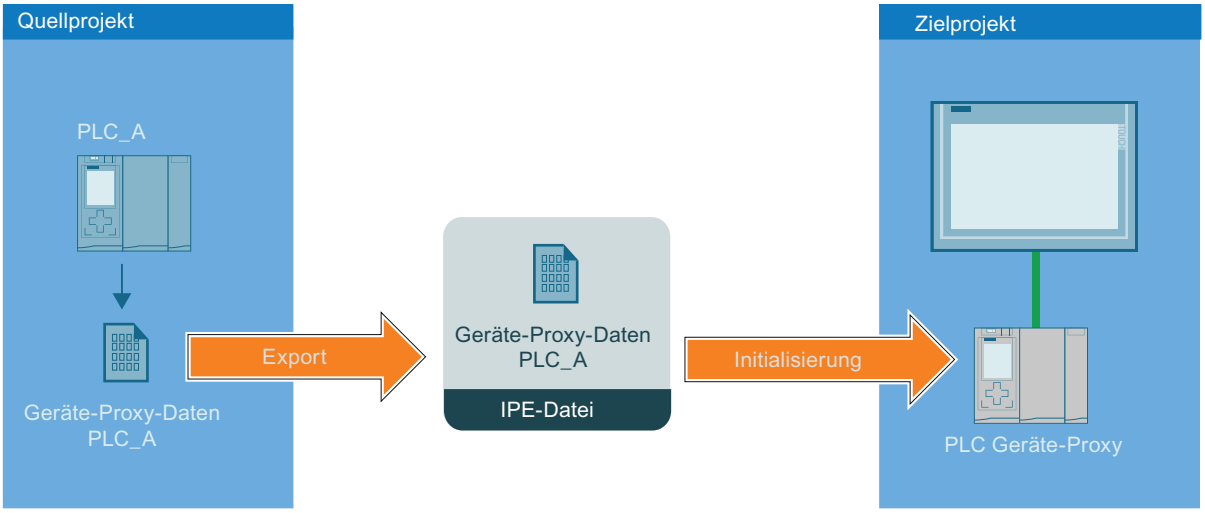

- Aus dem Quellprojekt wird eine IPE-Datei exportiert.
  Die IPE-Datei beinhaltet Geräte-Proxy-Daten der Steuerung "PLC_A".
- Im Zielprojekt wird ein Geräte-Proxy angelegt.
- Im Zielprojekt wird dann der Geräte-Proxy über die IPE-Datei mit Geräte-Proxy-Daten initialisiert.
- Nach der Initialisierung sind alle Geräte-Proxy-Daten aus der IPE-Datei im Geräte-Proxy enthalten.
- Bei Änderungen im Quellprojekt kann das Geräte-Proxy von PLC_A im Zielprojekt über die neu erzeugte IPE-Datei aktualisiert werden.

Geräte-Proxy über IPE-Datei initialisieren

Einleitung

Im TIA Portal initialisieren Sie Geräte-Proxys über einen Dialog am Geräte-Proxy.

In der IPE-Datei sind die Geräte-Proxy-Daten aus dem Quellprojekt enthalten.

Voraussetzung

IPE-Datei liegt vor.

Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation auf "Neues Gerät hinzufügen".
2. Wählen Sie unter "Controller" den Geräte-Proxy aus.

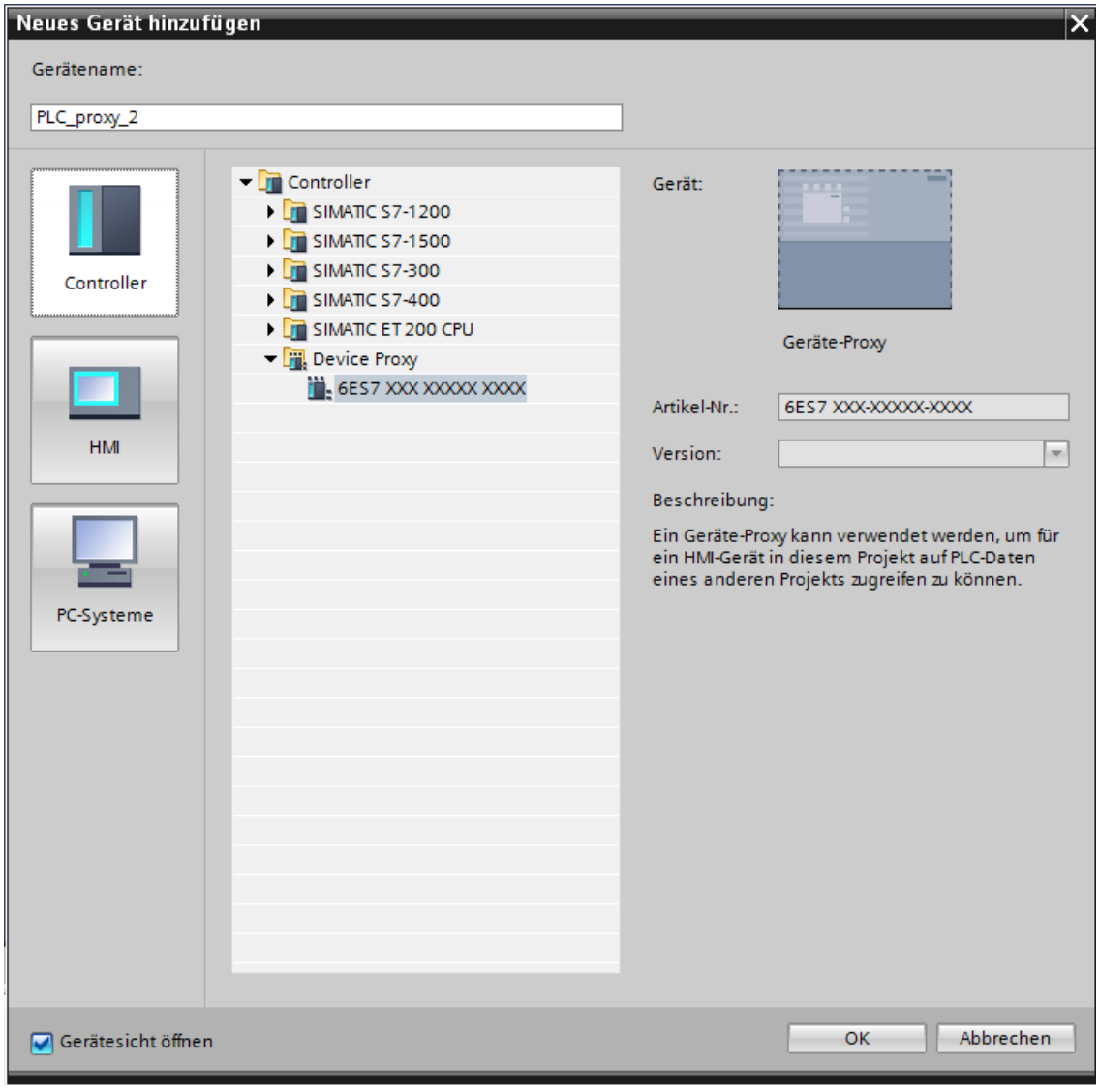

Im Editor "Geräte & Netze" wird ein neues Gerät angelegt.

3. Wählen Sie in der Projektnavigation den Geräte-Proxy aus.

4. Klicken Sie im Kontextmenü auf "Geräte-Proxy initialisieren".

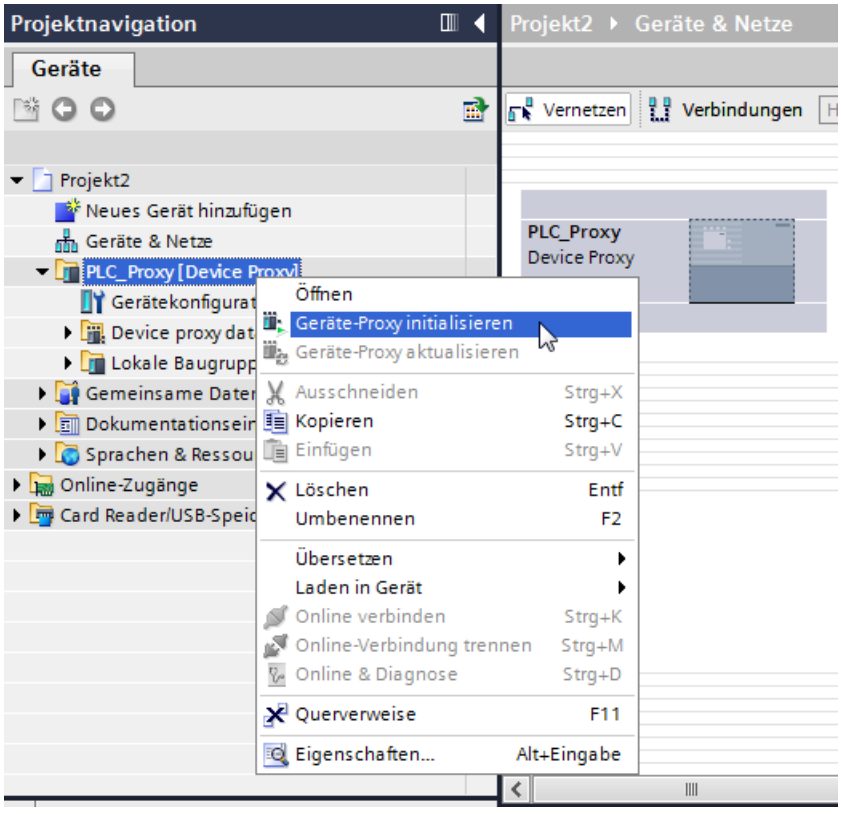

5. Wählen Sie die IPE-Datei aus und klicken Sie auf "Öffnen".
   Es öffnet sich der Dialog "Geräte-Proxy initialisieren".

Hinweis

Im Zielprojekt ist eine Auswahl der Geräte-Proxy-Daten vor der Initialisierung nicht möglich.

Es werden alle Geräte-Proxy-Daten übernommen, die in der IPE-Datei vorhanden sind.

Wenn Sie gezielt Geräte-Proxy-Daten vor einer Initialisierung auswählen wollen, dann initialisieren Sie das Geräte-Proxy über Projektdatei: Geräte-Proxy über Projektdatei initialisieren (Seite 225)

6. Klicken Sie auf "OK".
   Die Initialisierung des Geräte-Proxy wird gestartet.

Ergebnis

Nach erfolgreicher Initialisierung sind die Geräte-Proxy-Daten aus der IPE-Datei im Geräte-Proxy hinterlegt.

Sie können jetzt eine HMI-Verbindung mit dem Geräte-Proxy projektieren und z. B. PLC-Variablen aus dem Geräte-Proxy mit HMI-Variablen verbinden.

Geräte-Proxy über IPE-Datei aktualisieren

Einleitung

Wenn im Quellprojekt des Geräte-Proxys Änderungen an den Geräte-Proxy-Daten vorgenommen wurden, können Sie die Geräte-Proxy-Daten im Zielprojekt aktualisieren.

Voraussetzung

- Neue IPE-Datei mit den Geräte-Proxy-Daten aus dem Quellprojekt.
- Im Zielprojekt ist bereits ein Geräte-Proxy mit einer IPE-Datei initialisiert.

Vorgehensweise

1. Klicken Sie in der Projektnavigation auf den Geräte-Proxy .
2. Klicken Sie im Kontextmenü auf "Geräte-Proxy aktualisieren".

3. Wählen Sie die IPE-Datei aus.

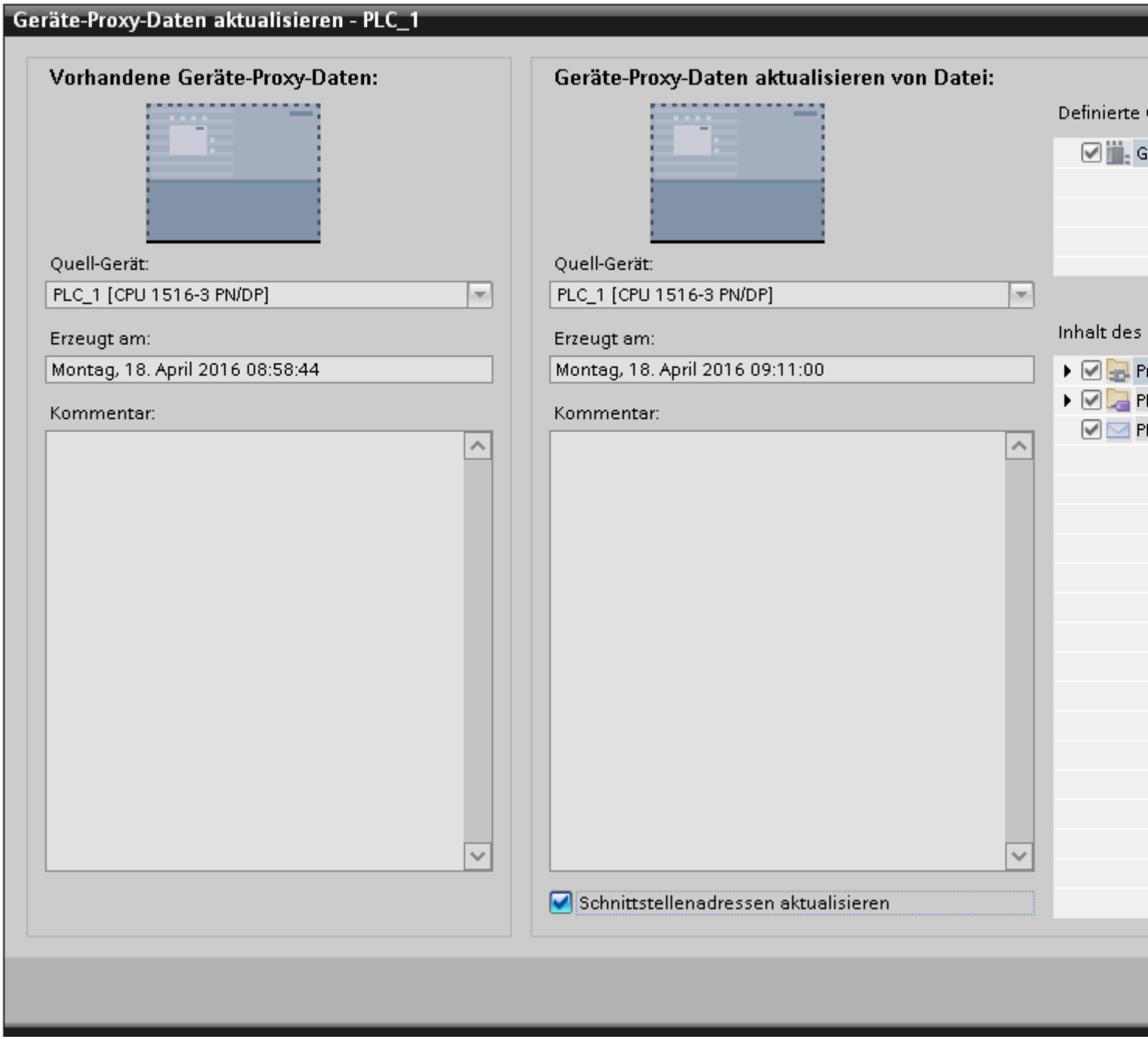

4. Wählen Sie aus, ob Sie die PROFINET- oder PROFIBUS-Adressen aktualisieren wollen.
5. Klicken Sie auf "OK".

5.6.1.3 Geräte-Proxy über Projektdatei initialisieren

Initialisierung von Geräte-Proxys über Projektdatei

Wie funktioniert eine Initialisierung über eine Projektdatei?

Steuerungsdaten können Sie über eine Projektdatei in Ihr TIA Portal-Projekt übernehmen:

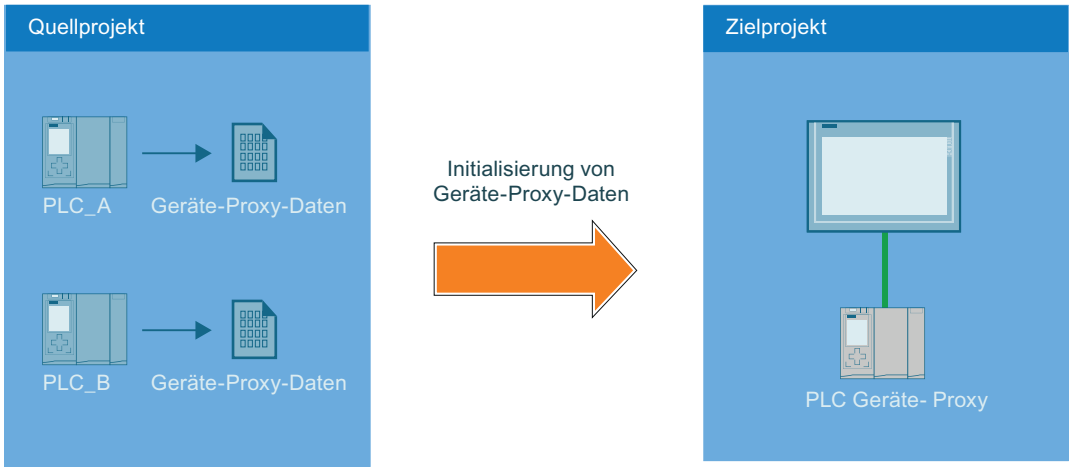

- Im Quellprojekt sind z.B. zwei Steuerungen vorhanden.
- Im Quellprojekt werden von beiden Steuerungen Geräte-Proxy-Datenobjekte angelegt.
- Das Quellprojekt wird abgespeichert.
- Im Zielprojekt wird ein Geräte-Proxy angelegt.
- Über den Eintrag zur Initialisierung des Geräte-Proxies im Geräte-Kontextmenü wird zuerst das Quellprojekt (*.ap13) ausgewählt.
- Zur Auswahl stehen die Steuerungen "PLC_A" und "PLC_B" und deren bereits angelegte Geräte-Proxy-Daten.
- Nach Auswahl von Geräte-Proxy-Daten der "PLC_A" werden die Daten im PLC_Proxy des Zielprojekts hinterlegt.
- Änderungen aus dem Quellprojekt können durch eine Aktualisierung über die Projektdatei übernommen werden.

Geräte-Proxy über Projektdatei initialisieren

Einleitung

Sie initialisieren Geräte-Proxys über eine Projektdatei.

Voraussetzung

Projektdatei (*.ap*) liegt vor.

Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation auf "Neues Gerät hinzufügen".
2. Wählen Sie unter "Controller" den Geräte-Proxy aus.

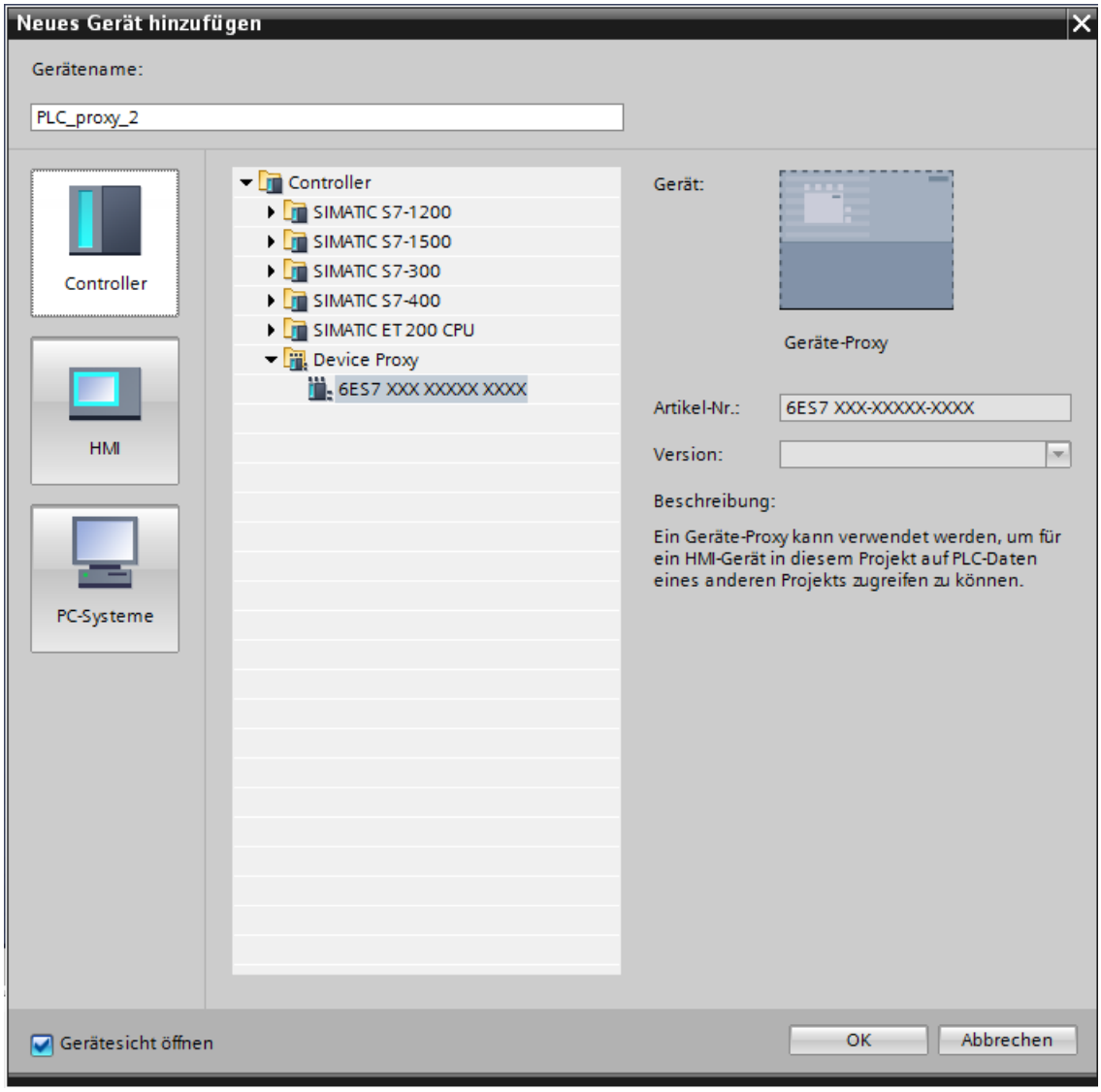

Im Editor "Geräte & Netze" wird ein neues Gerät angelegt.

3. Wählen Sie in der Projektnavigation den Geräte-Proxy aus.

4. Klicken Sie im Kontextmenü auf "Geräte-Proxy initialisieren".

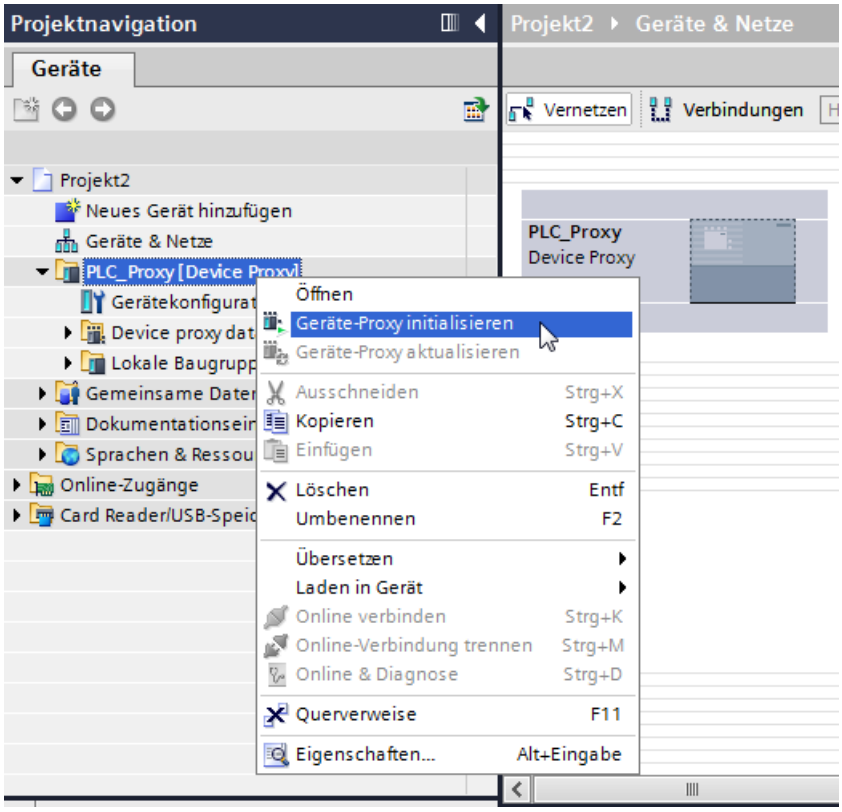

5. Wählen Sie die im Dialog "Geräte-Proxy-Daten Quelle öffnen" folgenden Eintrag aus: "TIA Portal Projekte (*.ap*)"
6. Wählen Sie eine Projekt-Datei aus und klicken Sie auf "Öffnen".
   Der Dialog "Geräte-Proxy initialisieren" öffnet sich.
7. Wählen Sie aus einer verfügbaren PLC ein bereits erstelltes Geräte-Proxy-Datenobjekt aus, um den Geräte-Proxy damit zu initialisieren.
8. Klicken Sie auf "OK".

Hinweis

Bei der Initialisierung des Geräte-Proxy wird automatisch geprüft, ob die ausgewählte Daten konsistent sind und in das Zielprojekt übernommen werden können. Wenn das Quellprojekt inkonsistente Daten enthält, kompilieren Sie Ihr Quellprojekt.

Ergebnis

Nach der Initialisierung sind die im ausgewählten Geräte-Proxy-Datenobjekt enthaltenen Steuerungsdaten aus der Projekt-Datei im Geräte-Proxy hinterlegt.

Sie können jetzt eine HMI-Verbindung mit dem Geräte-Proxy projektieren und z. B. PLC-Variablen aus dem Geräte-Proxy mit HMI-Variablen verbinden.

Initialisierung über STEP 7 V5.5 Projekte

Nähere Hinweise zum Importieren der Steuerungsdaten aus Projekten vor TIA Portal V11 finden Sie im Kapitel Integriertes Projektieren mit WinCC und Simatic Manager (Seite 237).

Weitere Informationen finden Sie im FAQ mit der Beitrags-ID: 73502293 (http://support.automation.siemens.com/WW/view/de/73502293)

Siehe auch

Integriertes Projektieren mit WinCC und SIMATIC Manager (Seite 237)

Geräte-Proxy über Projektdatei aktualisieren

Einleitung

Wenn im Quellprojekt des Geräte-Proxys Änderungen an den Steuerungsdaten vorgenommen wurde, können Sie den Geräte-Proxy in Ihrem TIA Portal-Projekt aktualisieren.

Voraussetzung

- Projektdatei wurde aus dem Quellprojekt des Geräte-Proxys erzeugt.
- Im Zielprojekt ist ein bereits initialisierter Geräte-Proxy vorhanden.

Vorgehensweise

1. Klicken Sie auf den Geräte-Proxy in der Projektnavigation.
2. Klicken Sie im Kontextmenü auf "Geräte-Proxy aktualisieren".

3. Wählen Sie die Projekt-Datei aus.
   Der Dialog "Geräte-Proxy aktualisieren" öffnet sich.

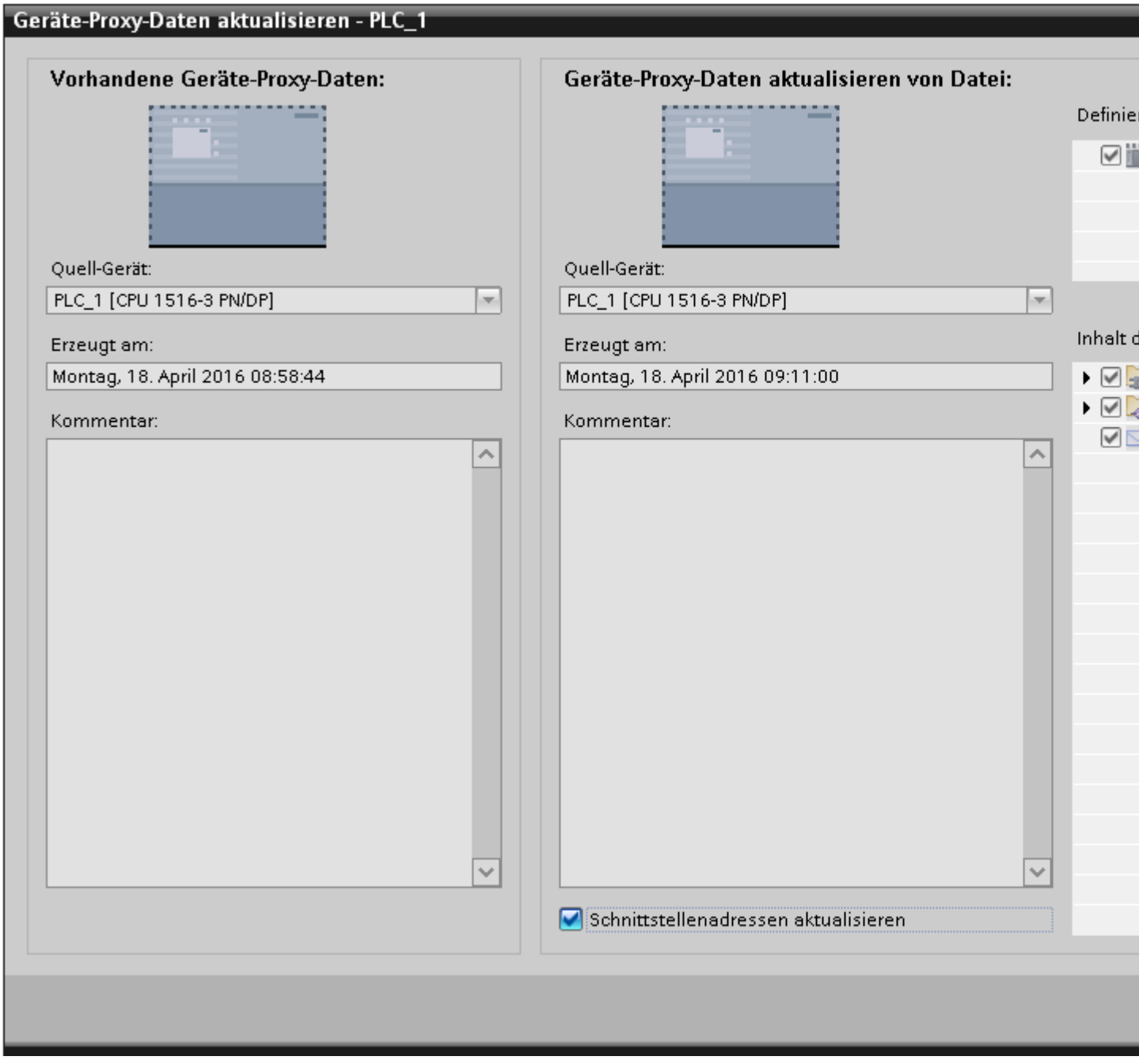

4. Wählen Sie ein Gerät aus.

Hinweis

Wenn Sie im Quellprojekt mehrere Geräte-Proxy-Datenobjekte angelegt haben, dann müssen Sie ein Geräte-Proxy-Datenobjekt auswählen.

Hinweis

Bereits initialisierte Geräte-Proxys können nicht mit Steuerungsdaten aus einer anderen Steuerung überschrieben werden.

5. Wählen Sie aus ob Sie die PROFINET- oder PROFIBUS-Adressen aktualisieren wollen.
6. Klicken Sie auf "OK".

5.6.2 Kommmunikation mit Geräte-Proxys

5.6.2.1 Grundlagen zur Kommunikation mit Geräte-Proxys

Einleitung

Datenaustausch zwischen dem Bediengerät in Ihrem Zielprojekt und vorhandenen Steuerung aus dem Quellprojekt wird im TIA-Portal mithilfe von Geräte-Proxys ermöglicht. Sie initialisieren einen Geräte-Proxy mit den Geräte-Proxy-Daten aus dem Quellprojekt und vernetzen das Bediengerät mit der Geräte-Proxy-PLC im Zielprojekt.

Das Bediengerät und die Geräte-Proxy-PLCs können direkt über ein Subnetz oder auch über mehrere Subnetze miteinander kommunizieren. Für die Kommunikation über mehrere Subnetzte projektieren Sie eine Geräte-Proxy-PLC als einen Router, der das Bediengerät und die Ziel-PLC verbindet.

Die Kommunikation von Verbindungspartner ist über folgende Verbindungen möglich:

- PROFINET
- PROFIBUS
- MPI

Netzwerkparameter einstellen

Im TIA Portal verbinden Sie das Bediengerät und die Geräte-Proxy-PLC im Editor „Geräte & Netze“.

Nachdem Sie ein HMI-Bediengerät mit der Geräte-Proxy-PLC vernetzt haben, passen Sie die Eigenschaften der Netzwerkparameter an. Sie notieren die

Netzwerkparameter aus dem Quellprojekt und tragen diese bei den Eigenschaften der Netzverbindung zwischen der Geräte-Proxy-PLC und dem HMI-Gerät ein.

Hinweis

IPE überträgt keine Daten zur Netzwerk-Konfiguration. Deshalb konfigurieren Sie die benötigten Netzwerke im Zielprojekt manuell so, dass die Netzwerk-Konfiguration im Zielprojekt zu der Konfiguration im Quellprojekt passt. Speziell die Subnetz-IDs der Geräte im Zielprojekt müssen zueinander und zu den IDs der Geräte im Quellprojekt passen. Andernfalls wird die Kommunikation zwischen dem Quellgerät und den HMI-Geräten in Runtime nicht möglich sein

Je nach Verbindung übertragen Sie folgende Parameter:

Verbindung	Parameter
Ethernet	S7-Subnetz-ID
MPI PROFIBUS	S7-Subnetz-ID Höchste Adresse Übertragungsgeschwindigkeit Busparameter

Kommunikation über S7-Routing

Um eine geroutete Verbindung zwischen den Verbindungspartnern herzustellen, muss ein Router zwischengeschaltet werden. Projektieren Sie sowohl den Router als auch die Ziel-PLC als einen Geräte-Proxy.

Der Router und die Ziel-PLC können mit einer IPE-Datei, TIA Portal-Projektdatei oder einer STEP 7 V5.5-Projektdatei initialisiert werden.

Die kommunikationsfähigen Baugruppen (CPUs oder CPs), die Netzübergänge zwischen den Subnetzen herstellen sollen, müssen "routing-fähig" sein.

Die S7-Routing-Einstellungen können über die betroffenen Schnittstellen-Eigenschaften bearbeitet werden.

Die S7-Routing-Verbindung projektieren Sie im Editor "Geräte & Netze".

Hinweis

Für eine geroutete Verbindung mit einem Verbindungspartner, der mit den Quellprojekt-Daten geladen wurde, muss auch im Quellprojekt eine geroutete Verbindung projektiert sein.

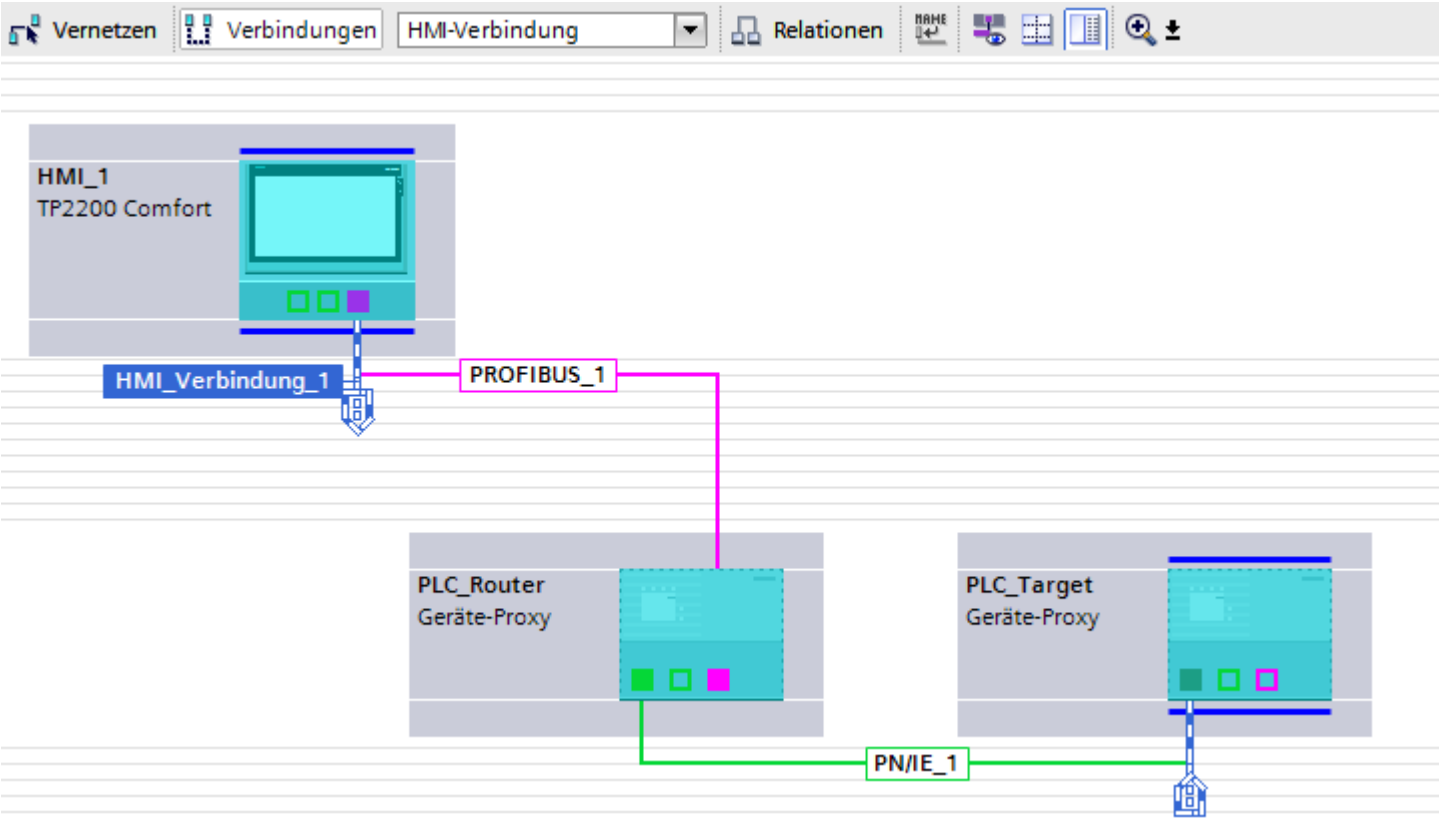

Der Routing-Weg wird in Runtime durch das System festgelegt und kann durch den Benutzer nicht beeinflusst werden. Während der Projektierung wird keine Information über eine fehlerhafte Verbindung ausgegeben werden. Alle relevanten Routing-Wege müssen im TIA-Portal genauso nachgebildet werden, wie sie im STEP 7-Quellprojekt projektiert waren.

5.6.2.2 Direkte Verbindung projektieren

Einleitung

Im Zielprojekt projektieren Sie eine direkte Verbindung zwischen einer Geräte-Proxy-PLC und einem Bediengerät im Editor „Geräte & Netze“.
Die nachfolgende Projektierung beschreibt ein Netz aus folgenden Kommunikationspartnern:

- HMI-Bediengerät
- Geräte-Proxy-PLC

Voraussetzungen

- Geräte-Proxy-Daten wurden aus dem Quellprojekt exportiert
- Ein HMI-Bediengerät im Zielprojekt ist angelegt
- Ein Geräte-Proxy ist angelegt

Vorgehensweise

1. Notieren Sie die Subnetz-ID der Verbindung im Quellprojekt.
   Bei einer MPI- oder PROFIBUS-Verbindung notieren Sie zusätzlich die höchste Adresse, die Übertragungsgeschwindigkeit und die Busparameter.
2.

   Initialisieren Sie den Geräte-Proxy im Zielprojekt mit den Geräte-Proxy-Daten aus Ihrem Quellprojekt.
3. Verbinden Sie die Kommunikationsschnittstellen des HMI-Bediengeräts und der Geräte-Proxy-PLC.
   Das entsprechende S7-Subnetz wird automatisch erzeugt.

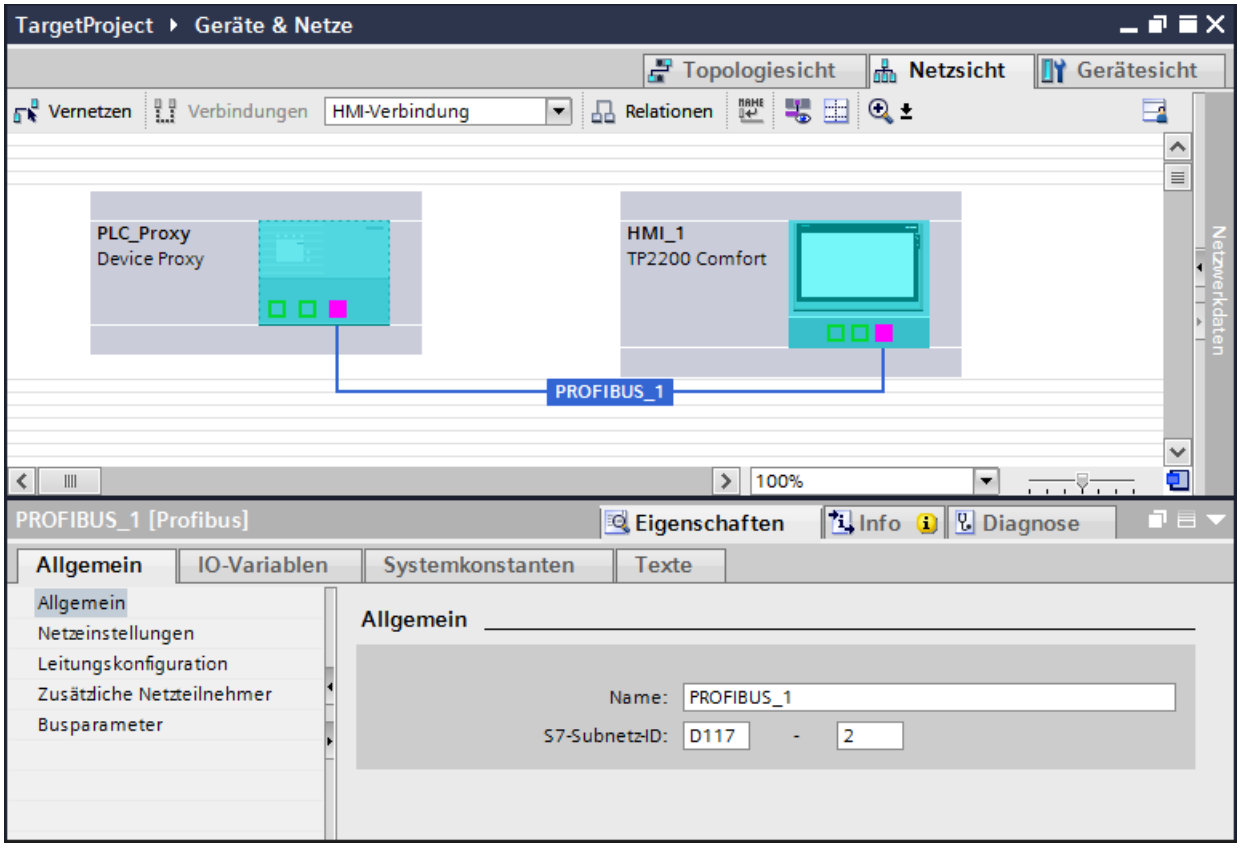

4. Übertragen Sie die Parameter, die Sie aus dem Quellprojekt notiert haben, in die Parameter des TIA-Projekts:
   - Die S7-Subnetz-ID tragen Sie unter "Eigenschaften > Allgemein" ein.
   - Die höchste Adresse und die Übertragungsgeschwindigkeit tragen Sie unter "Eigenschaften > Netzeinstellungen" ein.
   - Die Busparameter tragen Sie unter "Eigenschaften > Busparameter" ein.

5.6.2.3 Geroutete Verbindung projektieren

Einleitung

Das Bediengerät und die Geräte-Proxy-PLCs können über mehrere Subnetze miteinander kommunizieren. Für eine geroutete Verbindung projektieren sie eine Geräte-Proxy-PLC als Router zwischen dem Bediengerät und der Ziel-PLC.

Hinweis

Für eine geroutete Verbindung mit dem Verbindungspartner aus dem Quellprojekt muss auch in Quellprojekt eine geroutete Verbindung projektiert sein.

Die nachfolgende Projektierung beschreibt ein Netz aus folgenden Kommunikationspartnern:

- HMI-Bediengerät
- Geräte-Proxy PLC als Router
- Geräte-Proxy PLC als Ziel-PLC
  Sämtliche Subnetze, die im Routing zwischen dem HMI-Gerät und der Ziel-PLC involviert sind, müssen im Quellprojekt projektiert sein. Diese müssen im Zielprojekt identisch nachprojektiert werden.

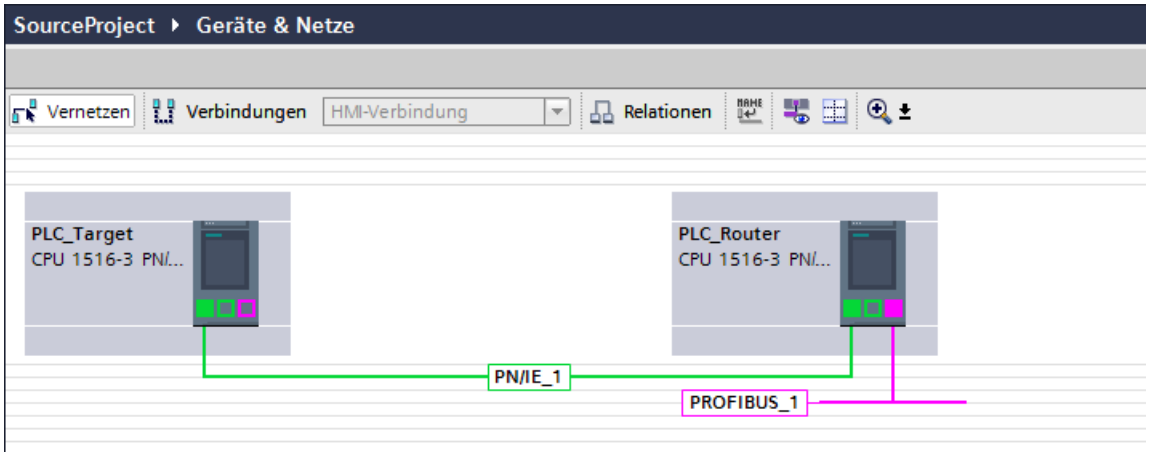

Voraussetzungen

- Geräte-Proxy-Daten wurden aus dem Quellprojekt exportiert
- HMI-Bediengerät ist angelegt
- Zwei Geräte-Proxys sind hinzugefügt

Vorgehensweise

1. Notieren Sie die Subnetz-IDs der Verbindungen im Quellprojekt.
   Bei einer PROFIBUS-Verbindung notieren Sie zusätzlich die höchste Adresse, die Übertragungsgeschwindigkeit und die Busparameter.
2. Initialisieren Sie den Quell-Proxy im Zielprojekt mit den Geräte-Proxy-Daten aus Ihrem Quellprojekt.
3. Erstellen Sie die Netzverbindung zwischen dem HMI-Bediengerät und dem Router sowie dem Router und der Ziel-PLC so, wie es im Quellprojekt projektiert war.
4. Klicken Sie auf die Schaltfläche "Verbindungen".
5. Verbinden Sie die Kommunikationsschnittstellen des HMI-Bediengeräts und des Routers. Router und Ziel-PLC werden automatisch verbunden.
6. Erstellen Sie eine HMI-Verbindung zwischen dem HMI-Gerät und der Ziel-PLC per Drag&Drop.
   Dialog "Mit Subnetz verbinden" wird geöffnet.

7. Wählen Sie "S7-geroutete Verbindung hinzufügen".
   Routing-Verbindung wird erstellt.

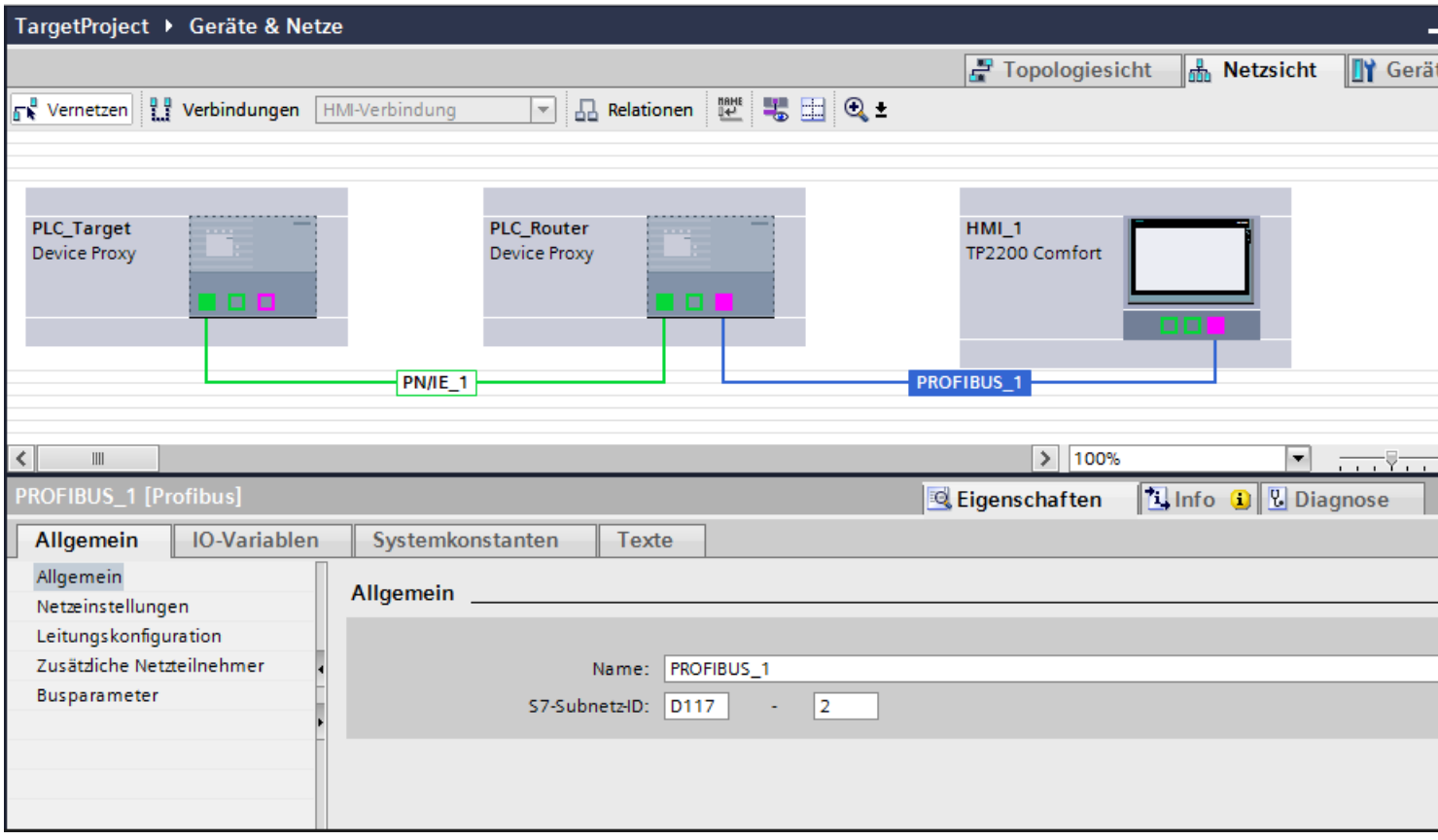

8. Übertragen Sie die Parameter, die Sie aus dem Quellprojekt notiert haben, in die Parameter des TIA-Projekts:
   - Die S7-Subnetz-ID tragen Sie unter "Eigenschaften > Allgemein" ein.
   - Die höchste Adresse und die Übertragungsgeschwindigkeit tragen Sie unter "Eigenschaften > Netzeinstellungen" ein.
   - Die Busparameter tragen Sie unter "Eigenschaften > Busparameter" ein.

Ergebnis

Die erstellte Verbindung sehen Sie im Editor "HMI-Verbindungen".

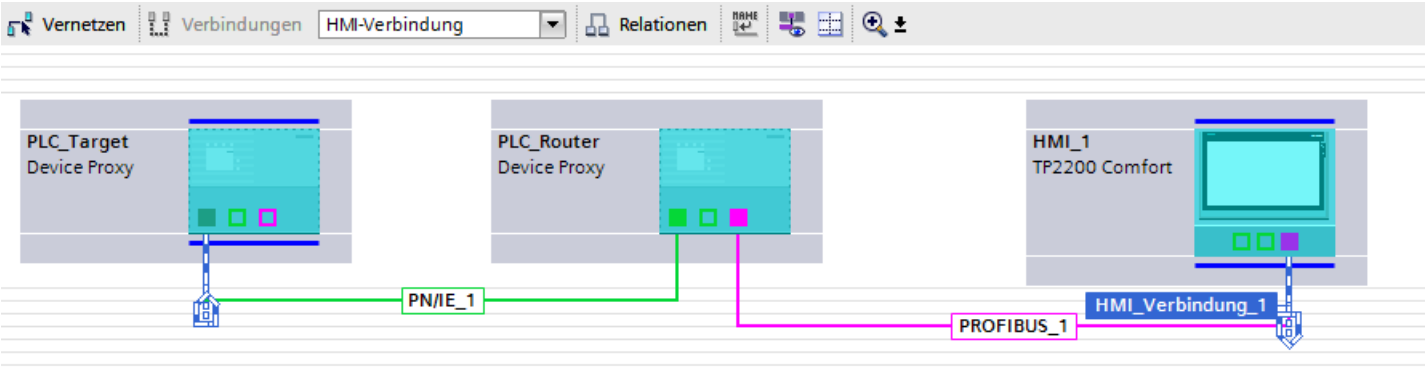

5.6.3 Integriertes Projektieren mit WinCC und SIMATIC Manager

5.6.3.1 Grundlagen zur Projektierung mit WinCC und SIMATIC Manager

Integriertes Projektieren mit WinCC und SIMATIC Manager

Das TIA Portal bietet die Möglichkeit SPS-Programme und die HMI-Projektierungen über eine einheitliche Softwareoberfläche zu erstellen.

In manchen Fällen ist es aus technischen oder kundenspezifischen Gründen erforderlich das SPS-Programm weiterhin mit der Software "STEP 7 V5.x" und die HMI-Projektierung mit der Software WinCC (TIA Portal)" zu erstellen.

Über die Geräte-Proxy PLC können Sie direkt auf die im SIMATIC Manager projektierten Steuerungen und deren aktuellen Daten zugreifen und diese in das WinCC (TIA Portal) integrieren.

Voraussetzung

- WinCC mit Version ab V13
- SIMATIC Manager STEP 7 mit Version ab V5.4 SP3
- Im SIMATIC Manager müssen alle notwendigen Software Optionspakete, die im STEP 7 Projekt verwendet werden, installiert sein.
- Das Projekt im SIMATIC Manager muss konsistent sein.

Unterstützte Steuerungen

- S7-300 Steuerung
- S7-300F Steuerung
- S7-400 Steuerung

- S7-400F Steuerung
- S7-300 T-Steuerung
- ET200 Steuerung (IM 151-x CPU)

Unterstützte Meldeverfahren

- Bitmeldeverfahren
- Analogmeldeverfahren
- Systemfehler Melden (SFM)
- Alarm_S, Alarm_SQ, Alarm_D, Alarm_DQ

5.6.3.2 Geräte-Proxy über STEP 7 Projektdatei initialisieren

Einleitung

Durch das Initialisieren wird eine Verbindung zu einer CPU innerhalb des SIMATIC Managers hergestellt. Nach erfolgreicher Initialisierung stehen die Variablen und Meldungen, die im SIMATIC Manager enthalten sind, auch im WinCC Projekt zur Verfügung.

Geräte-Proxy PLC initialisieren

1. Initialisieren Sie den Geräte-Proxy über die Projektdatei (*.s7p) Ihres STEP 7 Projekts. Im Dialog "Geräte-Proxy initialisieren" wählen Sie die CPU, Programmbausteine, Symbole und PLC-Meldungen für den Geräte-Proxy.
2. Nach der erfolgreichen Initialisierung werden die selektierten Daten in den Programm-Ordnern in der Projektnavigation angezeigt.

Hinweis

Die Daten werden in WinCC (TIA Portal) nicht bearbeitet. Es besteht nur lesender Zugriff auf die Variablen. Änderungen am STEP 7 Programm werden immer im SIMATIC Manager ausgeführt.

Hinweis

Unterstützung von H-PLCs

IPE Funktionalität im TIA Portal unterstützt auch die Verwendung von H-PLCs aus STEP 7 ab V5.5. Bei der Initialisierung des Geräte-Proxy ist nur die Master H-PLC relevant.

Anzeige des ursprünglichen Namens des Quellgeräts

1. Öffnen Sie die Gerätekonfiguration der Geräte-Proxy-PLC.
2. Wählen Sie "Gerätesicht".
3. Öffnen Sie "Eigenschaften > Allgemein > Allgemein > Geräte-Proxy-Information". Der ursprüngliche Name wird unter "Proxy-Quelle" angezeigt.

Netzverbindung zwischen HMI und SPS erstellen

Wie bei der Verwendung einer „Standard SPS“, projektieren Sie eine Netzwerkverbindung zwischen dem verwendetem HMI-Bediengerät und der Geräte Proxy-PLC.

1. Wählen Sie in Gerätekonfiguration "Netzsicht" aus.
2. Markieren Sie die Geräte-Proxy-PLC und tragen Sie unter "Eigenschaften" die gleichen Verbindungseigenschaften ein, welche das SPS-Programm im SIMATIC Manager besitzt.
3. Vernetzen und verbinden Sie die beiden Geräte.

Hinweis

Netzwerkverbindung und Adressen

Unabhängig davon, ob Sie eine Ethernet-Verbindung oder eine PROFIBUS-Verbindung erstellt haben, kontrollieren Sie die jeweiligen Adressen und Busparameter der verwendeten Baugruppen und passen Sie diese gegebenenfalls an. Wenn Sie PROFIBUS verwenden, dann beachten Sie die Zusatzinformationen unter Netzwerkparameter anpassen (Seite 239).

5.6.3.3 Netzwerkparameter anpassen

Einleitung

Wenn Sie ein HMI-Bediengerät mit der Geräte-Proxy-PLC vernetzt haben, dann müssen Sie die Eigenschaften der Parameter im STEP 7 5.x Projekt in den folgenden Fällen anpassen:

- Sie haben ein STEP 7 V5.x Projekt. Die in diesem Projekt vorhandenen und mit WinCC flexible erstellten HMI-Bediengeräte wurden nach WinCC (TIA Portal) migriert und mit der Geräte-Proxy-PLC vernetzt.
- Sie haben ein STEP 7 V5.x Projekt. HMI-Bediengeräte wurden nur mit WinCC (TIA Portal) projektiert und sind mit der Geräte-Proxy-PLC vernetzt.

Je nach Verbindung übertragen Sie folgende Parameter:

Verbindung	Parameter
Ethernet	S7-Subnetz-ID
MPI PROFIBUS	S7-Subnetz-ID Höchste Adresse Übertragungsgeschwindigkeit Busparameter

Sie notieren die Parameter im STEP 7-Projekt und tragen diese im TIA Portal bei den Eigenschaften der Netzverbindung ein.

Netzwerkparameter anpassen

1. Öffnen Sie im SIMATIC Manager die Hardwarekonfiguration der CPU.
2. Öffnen Sie die Objekteigenschaften der PROFIBUS-Schnittstelle der CPU.
3. Wählen Sie im Dialog „Eigenschaften MPI/DP“ unter „Allgemein“ die Schaltfläche „Eigenschaften…“ an.
   Der Dialog "Eigenschaften – PROFIBUS Schnittstelle" wird geöffnet.
4. Wählen Sie im Dialog „Eigenschaften – PROFIBUS Schnittstelle“ unter „Parameter“ die Schaltfläche „Eigenschaften…“ an.
   Der Dialog "Eigenschaften - PROFIBUS" wird geöffnet.

5. Notieren Sie die S7-Subnetz-ID.

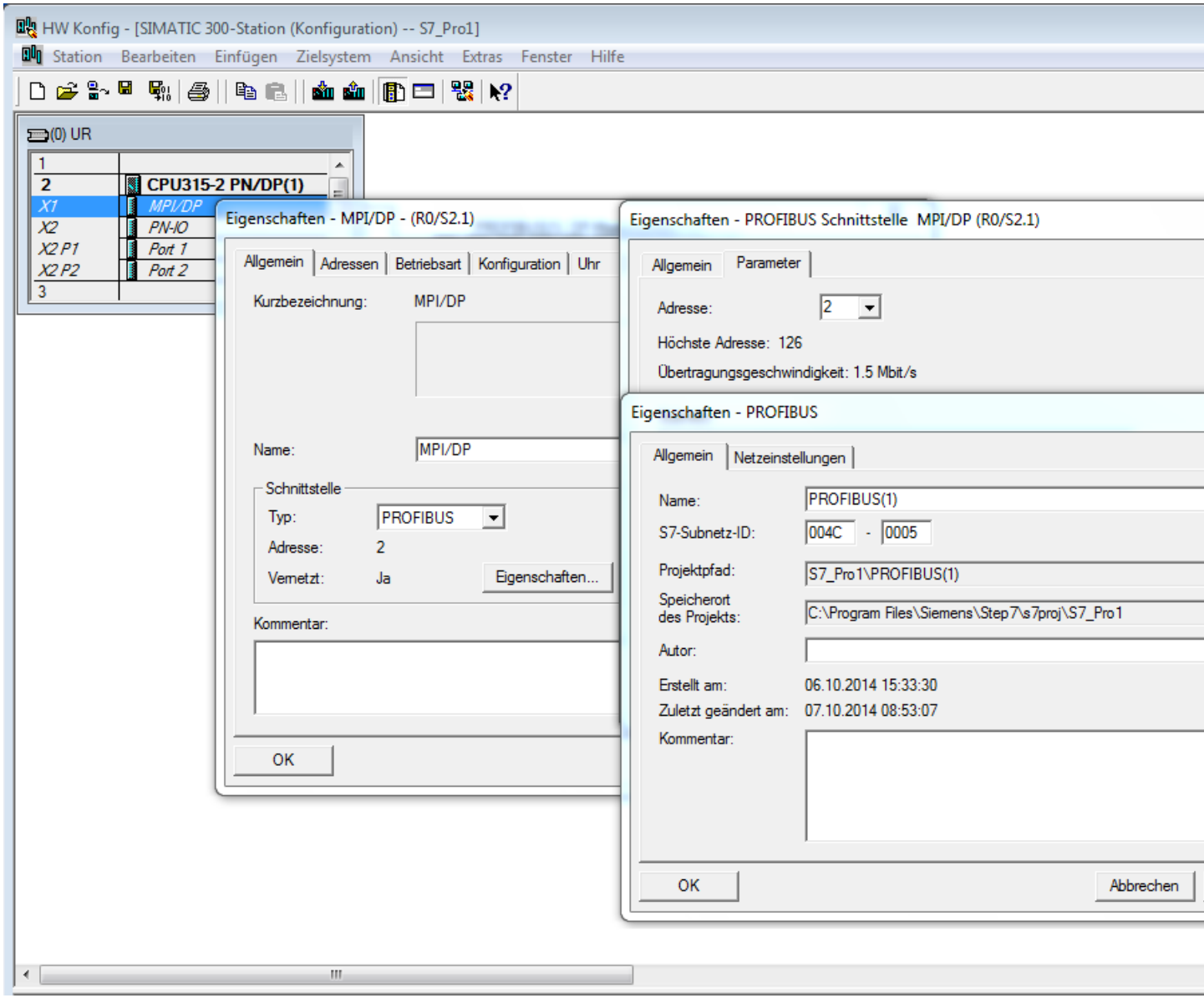

6. Wenn Sie die PROFIBUS-Verbindung anpassen, notieren Sie die Parameter "Höchste Adresse" und "Übertragungsgeschwindigkeit" im Dialogfeld "Netzeinstellungen".
7. Wenn Sie die PROFIBUS-Verbindung anpassen, wählen Sie im Fenster „Eigenschaften – PROFIBUS“ unter „Netzeinstellungen" die Schaltfläche "Busparameter" aus.
   Notieren Sie die dort eingetragenen Busparameter.
8. Wählen Sie im TIA Portal im Editor "Geräte & Netzte" die Netzverbindung zwischen der Geräte-Proxy-PLC und dem HMI-Bediengerät.

9. Wählen Sie unter "Eigenschaften > Allgemein > Netzeinstellungen" das Profil "Benutzerdefiniert".

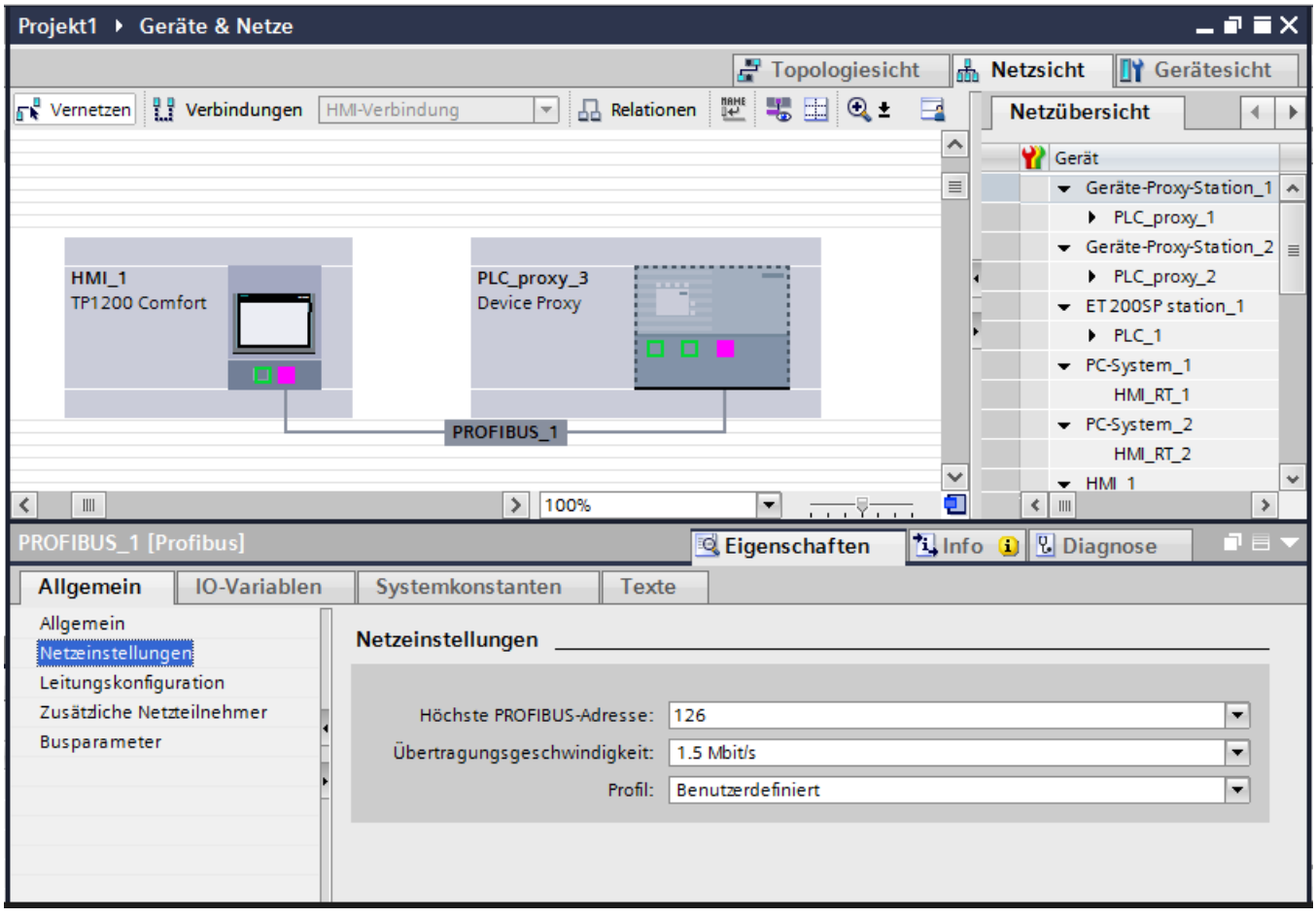

10. Übertragen Sie die Parameter, die Sie aus der STEP 7 Projektierung notiert haben, in die Parameter des TIA-Projekts:
    - Die S7-Subnetz-ID tragen Sie unter "Eigenschaften > Allgemein" ein.
    - Die höchste Adresse und die Übertragungsgeschwindigkeit tragen Sie unter "Eigenschaften > Netzeinstellungen" ein.
    - Die Busparameter tragen Sie unter "Eigenschaften > Busparameter" ein.

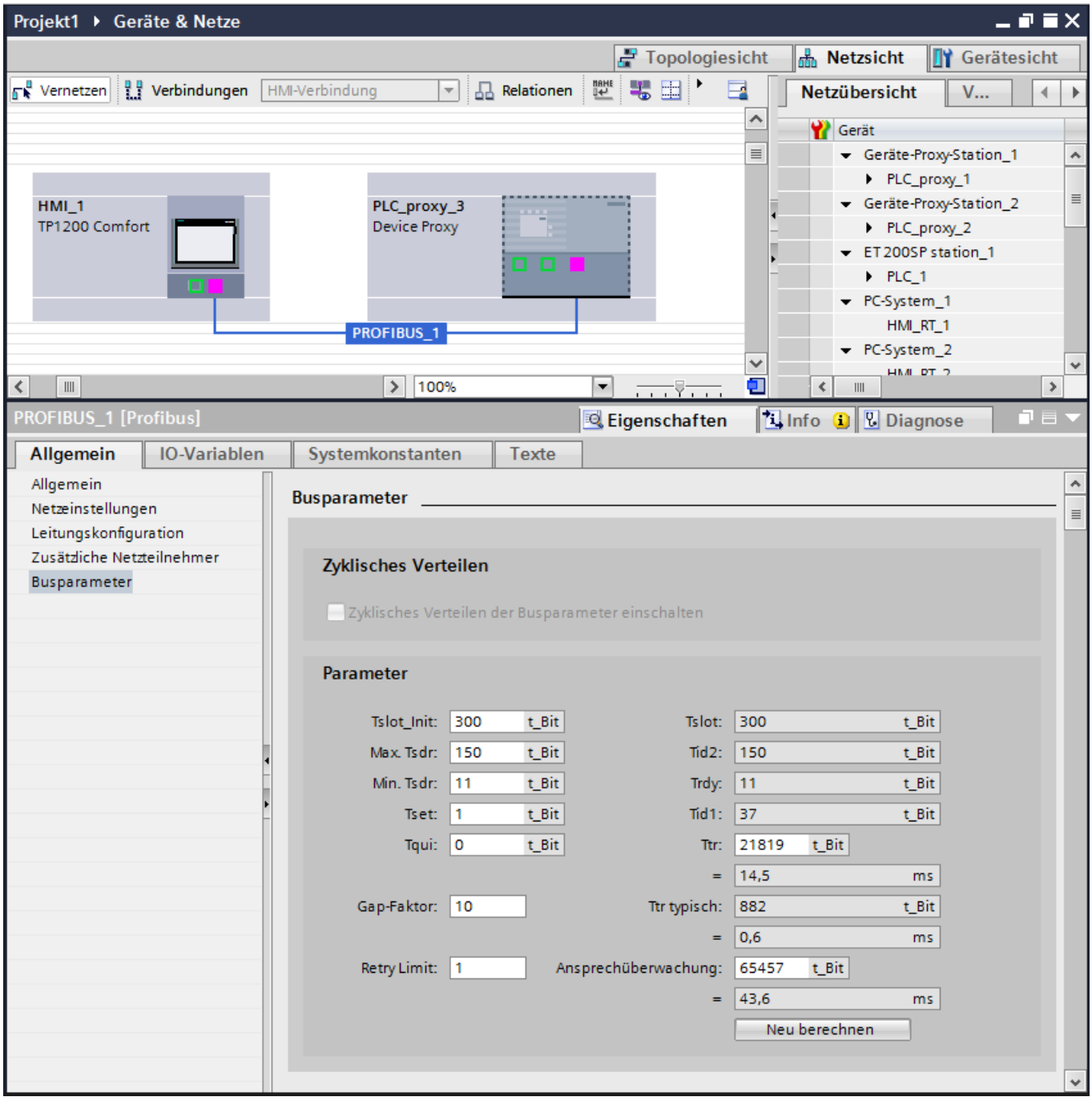
Projekt1 ▸ Geräte & Netze
Topologiesicht
Netzsicht
Gerätesicht
Vernetzen
Verbindungen
HMI-Verbindung
Relationen
Netzübersicht
V...
Gerät
Geräte-Proxy-Station_1
PLC_proxy_1
Geräte-Proxy-Station_2
PLC_proxy_2
ET 200SP station_1
PLC_1
PC-System_1
HMI_RT_1
PC-System_2
HMI_1
TP1200 Comfort
PLC_proxy_3
Device Proxy
PROFIBUS_1
100%
PROFIBUS_1 [Profibus]
Eigenschaften
Info
Diagnose
Allgemein
IO-Variablen
Systemkonstanten
Texte
Allgemein
Netzeinstellungen
Leitungskonfiguration
Zusätzliche Netzteilnehmer
Busparameter
Busparameter
Zyklisches Verteilen
Zyklisches Verteilen der Busparameter einschalten
Parameter
Tslot_Init: 300 t_Bit
Max. Tsdr: 150 t_Bit
Min. Tsdr: 11 t_Bit
Tset: 1 t_Bit
Tqui: 0 t_Bit
Gap-Faktor: 10
Retry Limit: 1
Tslot: 300 t_Bit
Tid2: 150 t_Bit
Trdy: 11 t_Bit
Tid1: 37 t_Bit
Ttr: 21819 t_Bit
= 14,5 ms
Ttr typisch: 882 t_Bit
= 0,6 ms
Ansprechüberwachung: 65457 t_Bit
= 43,6 ms
Neu berechnen

5.6.3.4 Inkonsistente Daten überprüfen

Einleitung

Bei der Initialisierung eines Geräte-Proxy wird automatisch geprüft, ob die ausgewählte Daten im Quellprojekt konsistent sind und in das Zielprojekt übernommen werden können. Wenn das Quellprojekt inkonsistente Daten enthält, reparieren Sie diese im Quellprojekt im SIMATIC Manager.

In STEP 7 V5.x stehen Ihnen folgende Werkzeuge zum Ermitteln und Korrigieren der inkonsistenten Daten:

- "Bausteinkonsistenz prüfen" zur Überprüfung von Datenbausteinen und Meldungen
- "Symbol Editor" zur Überprüfung der Symbole

Hinweis

Nähere Hinweise zu "Bausteinkonsistenz prüfen" und "Symbol Editor" finden Sie in der Dokumentation von STEP 7 V5.x.

Datenbausteine und Meldungen überprüfen

1. Öffnen Sie das Quellprojekt im S7 SIMATIC Manager.
2. Markieren Sie den Bausteinordner im Projektfenster.
3. Wählen Sie im Kontextmenü den Eintrag "Bausteinkonsistenz prüfen...".
   Editor "Bausteinkonsistenz prüfen" wird geöffnet.
4. Wählen Sie im Menü den Befehl "Programm > Übersetzen".
   Inkonsistenzen in den Bausteinen werden automatisch korrigiert und Objekte werden übersetzt.
5. Wenn eine Inkonsistenz nicht automatisch korrigiert werden kann, wird eine Fehlermeldung ausgegeben. Durch einen Doppelklick auf die Fehlermeldung gelangen Sie zum fehlerhaften Objekt. Korrigieren Sie die Inkonsistenzen manuell und speichern Sie anschließend das Projekt.

Symbole überprüfen

1. Öffnen Sie das Quellprojekt im S7 SIMATIC Manager.
2. Starten Sie den Symbol Editor durch einen Doppelklick auf die Symboltabelle.
   Symbol Editor öffnet sich.
3. Ungültige globale Symbole in der Tabelle sind rot markiert.
4. Reparieren oder löschen Sie die ungültigen Symbole.

5.6.3.5 Variablen und Verbindungen anpassen

Einleitung

Für den Verbindungsaufbau nach der Migration passen Sie den Verbindungsnamen an und synchronisieren Sie die Variablen.

Mit dem Vernetzen des HMI Bediengerätes mit der Geräte-Proxy PLC erstellt das System eine Verbindung und vergibt einen Verbindungsnamen. Falls bereits eine HMI-Verbindung vorlag, passen Sie den neuen Verbindungsnamen an.

Verbindungsnamen anpassen

1. Kopieren Sie den vorherigen Verbindungsnamen.
2. Löschen Sie die bereits vorhandene Verbindung.
3. Ersetzen Sie den neuen Verbindungsnamen durch den vorherigen Namen.

Symbolische Anbindung der Variablen

1. Rufen Sie den Variableneditor auf.
2. Markieren Sie alle Variablen.
3. Klicken Sie auf „Synchronisieren zum Abgleich mit der PLC-Variable“.
   Das Kontextfenster „Optionen für das Synchronisieren von WinCC-Variablen“ öffnet sich.
4. Wählen Sie in dem Fenster die Optionen „Datentyp und absolute Adresse übereinstimmt“ und „WinCC-Variablennamen durch den PLC-Variablennamen ersetzen“.
5. Betätigen Sie die Schaltfläche „Synchronisieren“.
   Die Synchronisierung der Variablen wird ausgeführt.

5.6.3.6 Geräte-Proxy über STEP 7 Projektdatei aktualisieren

Einleitung

Die Daten der Geräte-Proxy-PLC müssen immer dann aktualisiert werden, wenn Sie im SIMATIC Manager Änderungen vorgenommen haben, die das HMI Bediengerät betreffen. Dieses ist z. B. der Fall, wenn Adressbereiche in Datenbausteinen geändert oder erweitert wurden.

Geräte-Proxy-Daten aktualisieren

1. Aktualisieren Sie die Daten des Geräte-Proxy über die Projektdatei (*.s7p) Ihres STEP 7 Projekts.
2. Wählen Sie im Updatedialog die gleiche CPU als Datenquelle aus, die Sie auch bei der Initialisierung der Geräte-Proxy-PLC verwendet haben.
   Ein Update mit einer anderen HW Konfiguration ist nicht zulässig.

3. Über den Dialog haben Sie die Möglichkeit, neu erstellte Datenbausteine aus dem SIMATIC Manager zusätzlich mit anzuwählen.
   Es werden nur angewählte Objekte in das TIA Portal Projekt übernommen.

   Hinweis

   Alle Daten, die bei der vorherigen Initialisierung angewählt wurden, sind automatisch wieder mit angewählt. Diese Voreinstellung sollten Sie nur ändern, wenn Sie bewusst Daten aus der Geräte-Proxy-PLC entfernen wollen. Wurde z. B. der DB1 durch die Initialisierung in das TIA Portal Projekt eingebunden und Sie deaktivieren jetzt den DB1, dann wird bei der Aktualisierung der Daten der DB1 im TIA Portal Projekt gelöscht.

4. Bestätigen Sie die Auswahl über die Schaltfläche „OK“.
   Das Update der Variablen wird gestartet.
5. Durch die symbolische Anbindung der Variablen wurden die Adressen der Variablen automatisch angepasst.
   Damit ist das Update der Variablen abgeschlossen.

5.6.3.7 Variablen aus dem SIMATIC Manager einbinden

Einleitung

Mit dem Initialisieren der Geräte-Proxy PLC haben Sie die Variablen aus dem SIMATIC Manager in das WinCC Projekt eingebunden.

In WinCC fügen Sie Variablen der Geräte-Proxy PLC direkt in ein HMI-Bild oder in ein Bildobjekt ein.

Variable in die HMI-Projektierung einfügen

1. Öffnen Sie in der Projektnavigation den Ordner mit den Programmbausteinen der Geräte-Proxy PLC.
2. Selektieren Sie den Baustein, in dem sich die Variablen befinden.
3. In Detailansicht werden alle Variablen des selektierten Bausteins angezeigt.
4. Ziehen Sie die Variable per Drag&Drop aus der Detailansicht in den Arbeitsbereich.
   Ein E/A-Feld mit der ausgewählten Variable wird erstellt. Im Inspektorfenster nehmen Sie weitere Einstellungen vor.

Alternativ konfigurieren Sie ein E/A-Feld im HMI-Bild und ziehen Sie die Variable per Drag&Drop in das E/A-Feld.

Ergebnis

Die Variable aus einem Baustein der Geräte-Proxy PLC wird in das HMI-Projekt hinzugefügt und mit dem E/A-Feld verbunden.

5.6.3.8 Systemfunktion "Direkttasten" konfigurieren

Einleitung

Wenn Sie HMI Bediengeräte in der Hardwarekonfiguration des SIMATIC Managers einbinden wollen, die nur mit WinCC (TIA Portal) projektierbar sind, z. B. SIMATIC HMI Comfort Panels, dann benötigen Sie hierfür passende GSD/GSDML Dateien.

Die erforderlichen GSD/GSDM Dateien für WinCC finden Sie im FAQ Gemeinsames Projektieren mit WinCC (TIA Portal) und STEP 7 V5.x (http://support.automation.siemens.com/WW/view/de/73502293)

GSD Dateien installieren

1. Öffnen Sie die Hardwarekonfiguration im SIMATIC Manager.
2. Wählen Sie den Menübefehl „Extras > GSD-Dateien installieren…“.
   Dialog „GSD-Dateien installieren“ öffnet sich.
3. Wählen Sie über die Schaltfläche „Durchsuchen“ den Dateiordner an, in dem sich die GSD Dateien befinden.

   Hinweis

   PROFINET / PROFIBUS Ablagepfade im SIMATIC Manager

   Die GSDML Dateien für die HMI Bediengeräte finden Sie im Hardware Katalog unter „PROFINET IO > HMI > SIMATIC HMI > GSD > KP/x“.

   Die GSD Dateien für die HMI Bediengeräte finden Sie im Hardware Katalog unter „PROFIBUS DP > Weitere FELDGERÄTE > MMI > SIMATIC_HMI > HMI CP_x“.

4. Markieren Sie alle Dateien und betätigen Sie die Schaltfläche „Installieren“.

PROFINET-Verbindung: PROFINET IO Direkttasten

1. Notieren Sie die verwendeten Eigenschaften des zu ersetzenden Bediengeräts.
   Achten Sie besonders auf E/A-Adressen, Gerätenamen, Gerätenummer und Diagnose-Adressen, wenn sie ausgewertet werden.
2. Entfernen Sie die bestehende GSD-Datei.
3. Ziehen Sie den projektierten Bediengerätetyp per Drag&Drop aus dem Hardware Katalog auf den dargestellten PROFINET IO Bus.
4. Passen Sie die Eigenschaften, wie vorher notiert, an.

   Hinweis

   Achten Sie beim Gerätenamen auf die Regeln für Namen für PROFINET IO-Geräte. Verwenden Sie im Gerätenamen nur Kleinbuchstaben und keine Sonderzeichen.

5. Bestätigen Sie die Eingaben mit OK.

6. Speichern und übersetzen Sie die Konfiguration und übertragen Sie die Hardwarekonfiguration in die Steuerung.

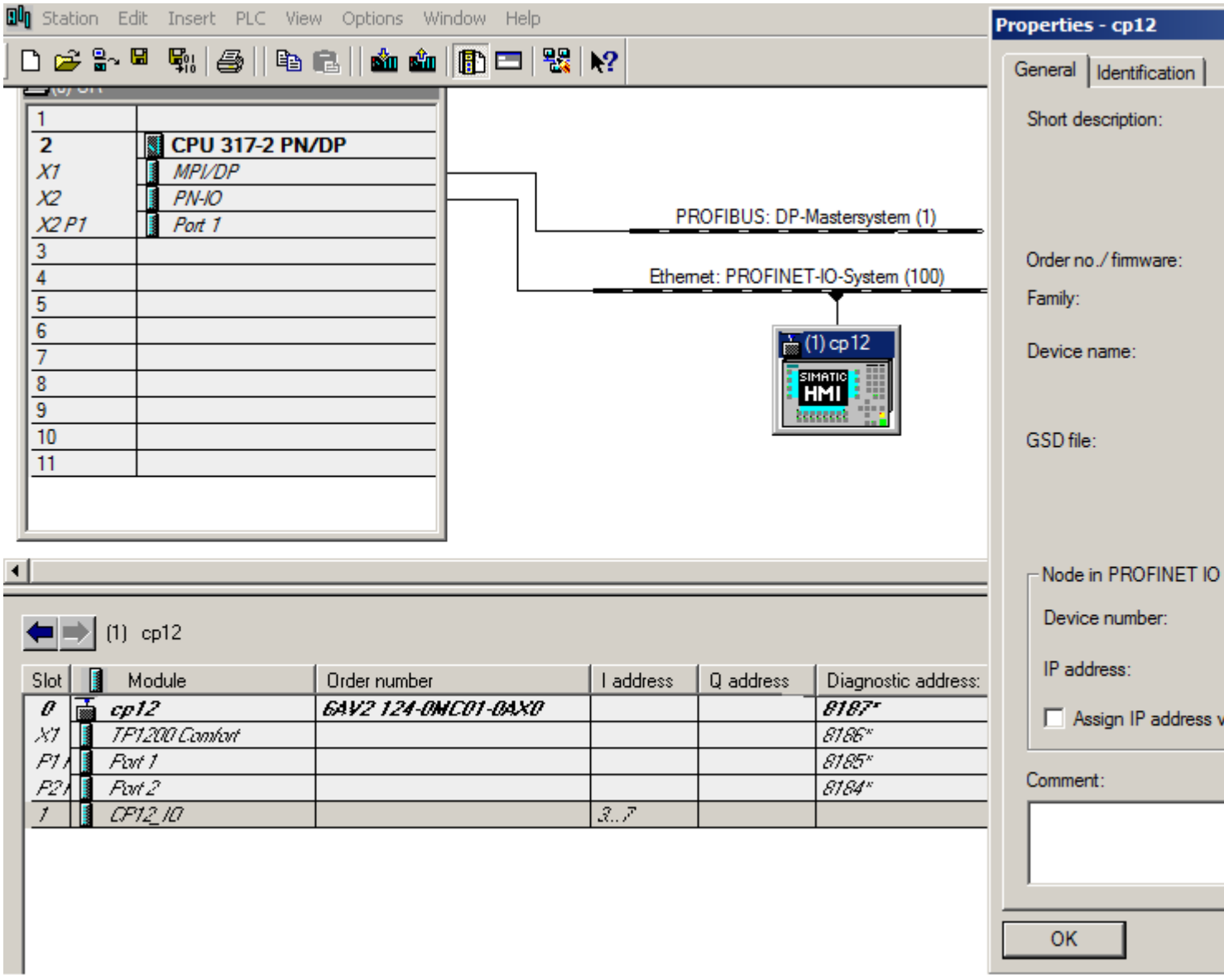

7. Öffnen Sie die Gerätekonfiguration des TP1200 Comfort und wechseln Sie in die „Gerätesicht“.
8. Tragen Sie den PROFINET-Gerätenamen in Eigenschaften des projektierten Bediengeräts in WinCC unter "Eigenschaften > Allgemein > PROFINET Schnittstelle X1 > Ethernet-Adressen" ein.
   Um den PROFINET-Gerätenamen zu vergeben, deaktivieren Sie die Option „PROFINET-Gerätename automatisch generieren“.

   Hinweis

   Der PROFINET- Name in der WinCC Projektierung muss mit dem im SIMATIC Manager hinterlegten PROFINET-Namen übereinstimmen.

PROFIBUS-Verbindung: PROFIBUS DP Direkttasten

Beachten Sie, dass die Direktasten im SIMATIC Manager konfiguriert werden.

Hinweis

Wenn von einem Panel Direkttasten zu mehreren CPUs projektiert werden, dann funktionieren die Direkttasten nur an einer CPU.

Die weiteren CPUs zeigen einen Bus-/Sammel-Fehler an.

1. Notieren Sie die verwendeten Eigenschaften des zu ersetzenden Bediengeräts.
   Achten Sie besonders auf E/A-Adressen, Gerätenamen und Diagnose-Adressen, wenn sie ausgewertet werden.
2. Entfernen Sie die bestehende GSD-Datei.
3. Ziehen Sie das GSD-Modul, in dem sich das projektierte Bediengerät befindet, per Drag&Drop aus dem Hardware Katalog auf den dargestellten PROFIBUS DP-Bus.
4. Passen Sie die Eigenschaften E/A-Adresse, Gerätename, PROFIBUS-Adresse und Diagnose-Adresse, wie vorher notiert, an.
5. Bestätigen Sie die Eingaben mit OK.

6. Speichern und übersetzen Sie die Konfiguration und übertragen Sie die Hardwarekonfiguration in die Steuerung.

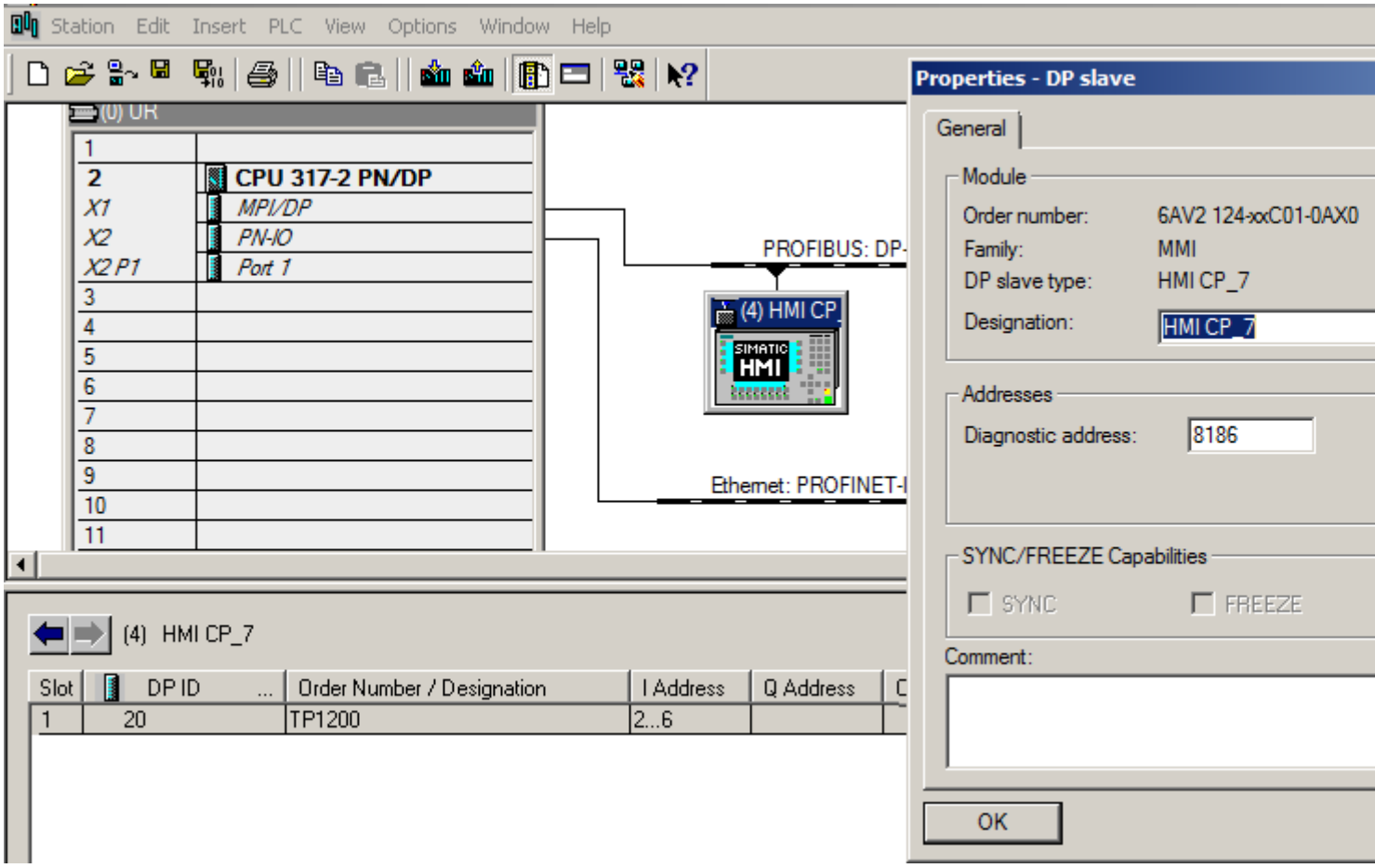

7. Öffnen Sie die Gerätekonfiguration des Bediengeräts, den sie einsetzten, und wechseln Sie in die „Gerätesicht“.
8. Vernetzen Sie die PROFIBUS-Schnittstelle in Eigenschaften des projektierten Bediengeräts unter "Eigenschaften > Allgemein > MPI/DP Schnittstelle X2 > PROFIBUS-Adresse > Schnittstelle vernetzen mit".
9. Geben Sie bei "Parameter" die PROFIBUS-Adresse vor.

Hinweis

Die PROFIBUS-Adresse in der WinCC Projektierung muss mit der im SIMATIC Manager hinterlegten PROFIBUS-Adresse übereinstimmen.

Index

A

aktualisieren
    Geräte-Proxy, 223, 228
    IPE-Daten, 223, 228
Anpassen
    Verbindungsname, 245
Anzeigen
    Programmnamen, 239
Arbeiten mit Multiuser Engineering, 13, 16, 32, 183
austauschen, 219

D

direkte Verbindung
    projektieren, 232

E

Einfügen
    Variable, 246
erstellen
    Netzverbindung, 239

G

Geräte-Proxy
    aktualisieren, 223, 228
    initialisieren, 225
geroutete Verbindung
    projektieren, 234

I

initialisieren
    IPE-Datei, 220
    Projektdatei, 225
Initialisiereung
    Projektdatei, 225
Initialisierung
    IPE-Datei, 219
Integriertes
    Projektieren, 237
Inter Project Engineering
    IPE-Datei mithilfe des "Geräte-Proxy-Daten" exportieren, 218
    Software- und Hardwarevoraussetzungen, 213
    Steuerungsdaten projektübergreifend austauschen, 214
    Voraussetzungen, 213
    Vorgehensweise Austausch von Steuerungsdaten über IPE-Datei, 214
    Vorgehensweise Austausch von Steuerungsdaten über Projekt-Datei, 215
    Vorgehensweise zur Aktualisierung bereits übertragener Steuerungsdaten, 216
Inter Project Engineering (IPE)
    Einführung, 209
    Steuerungsdaten über IPE-Datei austauschen, 210
    Steuerungsdaten über Projektdatei austauschen, 210
Inter Projekct Engineering
    Geräte-Proxy-Daten im Quellprojekt anlegen, 217
IPE
    Datei, 219, 220
    Einführung, 209
    Geräte-Proxy-Daten im Quellprojekt anlegen, 217
    Software- und Hardwarevoraussetzungen, 213
    Steuerungsdaten des "Geräte-Proxy-Daten" als IPE-Datei exportieren, 218
    Steuerungsdaten projektübergreifend austauschen, 214
IPE-Datei, 219
    initialisieren, 220
    Initialisierung, 219
IPE-Daten
    aktualisieren, 223, 228

K

Kommunikation
    Routing, 230
    S7-Routing, 234

M

Mit lokalen Sessions arbeiten, 147
Mit Multiuser Engineering arbeiten, 13, 16, 32, 183
Multiuser Commissioning
    Einführung, 181
Multiuser Engineering
    Deinstallation des Multiuser-Servers, 55
    Einführung, 13, 16, 32, 49, 183

Installation des Multiuser-Servers als Standalone-Installation, 49
Installation des Multiuser-Servers mit dem TIA Portal, 49
Installation des Multiuser-Servers starten, 52, 54
Regeln und Voraussetzung für die Installation, 53, 54

N

Netzverbindung
erstellen, 239

P

PLC-Daten, 219
Programmnamen
anzeigen, 239
Projektdatei
initialisieren, 225
Initialisierung, 225
Projektieren
direkte Verbindung, 232
Integriertes, 237
Proxy
Routing, 230

R

Regeln für das Arbeiten in lokalen Sessions, 148
Routing-Verbindung
projektieren, 230

S

S7-Routing
projektieren, 234
SIMATIC Manager, 237
Steuerungsdaten, 219
initialisieren, 220
Symbolische Anbindung
Variable, 245

T

Team Engineering
Arbeiten mit Projektkopien, 194
Beispiel für die Programmstruktur im Master-Projekt, 203
Einführung zur gemeinsamen Inbetriebnahme, 193
Hinweise zur Kompatibilität, 195
Master-Projekt erstellen, 194
Projektkopien wieder in das Master-Projekt integrieren, 195
Regeln für das Arbeiten mit Organisationsbausteinen, 205
Regeln für das Bearbeiten gemeinsam genutzter zentraler Objekte, 205
Regeln für das gemeinsame Arbeiten auf einer CPU, 204
Regeln für das Laden in die CPU, 205
Regeln für das Master-Projekt, 203
Regeln für Online-Funktionen, 206
Software- und Hardwarevoraussetzungen, 197
Voraussetzungen für das gemeinsame parallele Arbeiten, 197
Vorgehensweise bei der PLC-Synchronisierung, 199
Vorgehensweise zum Erstellen des Master-Projektes, 199
Vorgehensweise zum Integrieren von Projektkopien in das Master-Projekt, 201
Vorgehensweise zum manuellen Synchronisieren von konkurrierenden Änderungen, 200
Vorgehensweise zur Modifizierung zentraler Objekte innerhalb des Master-Projektes, 201

U

Übersicht über die Arbeitsweise mit Multiuser Commissioning, 181
Übersicht über die Arbeitsweise mit Multiuser Engineering, 13, 16, 32, 183
Unterstützte Objekte für Multiuser Engineering
Multiuser Engineering, (Unterstützte Objekte)

V

Variable
einfügen, 246
Symbolische Anbindung, 245
Verbindungsname
anpassen, 245
Vorgehensweise für das Arbeiten mit lokalen Sessions, 147